DYNAMIC THERMAL MANAGEMENT OF MULTI CORE PROCESSORS USING CORE HOPPING

by

AVINASH RAGHU

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN AEROSPACE ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2010

ACKNOWLEDGEMENTS

At the outset, I am extremely grateful to my thesis advisor Prof. Dereje Agonafer, for his kind support and guidance and indefatigable enthusiasm en-route to this thesis. His timely words of encouragement and frequent, pertinent insights were instrumental in this piece of work attaining fruition. I would also like to express my gratitude to Dr. Haji-Sheikh and Dr. Seiichi Nomura for kindly consenting to be on my supervising committee.

I am also deeply indebted to my seniors and colleagues at the EMNSPC for their selfless help and constant encouragement which made all the time spent in the lab worthwhile. The times we shared will be moments to cherish forever.

Ms. Sally Thompson has been invaluable as an administrative liaison between myself and Dr. Agonafer, without whose timely reminders, most of the necessary formalities would have gone unnoticed. I would also like to thank the administrative staff at the Department of Mechanical and Aerospace Engineering for all the help they extended to me.

And of course, words are not enough to express the gratitude I feel towards my family – who have always supported me in every endeavor, no matter how trying the times or uncertain the path, and have had only positive words to express and spur me on in this effort. This work is theirs as much as it is mine.

April 19, 2010

ABSTRACT

DYNAMIC THERMAL MANAGEMENT OF
MULTI CORE PROCESSORS USING
CORE HOPPING

Avinash Raghu, M.S.

The University of Texas at Arlington, 2010

Supervising Professor: Dereje Agonafer

Multi core processors are the order of the day today. The demand for greater levels of performance and the consequent increase in design complexity have rendered the single core processor obsolete and resulted in more cores being put onto a single chip. Processors incorporating more than 4 cores on a chip are dubbed many core processors. This, while improving performance, has lead to increased power densities. Also, the power distribution across the die surface is not uniform. These factors together, result in heat being concentrated at specific points on the die surface, called 'hot spots'.

The increase in die temperature results in reduced performance and reliability and increased leakage currents and cooling costs. Existing cooling methodologies such as heat sinks and fans are hard pressed to alleviate the high temperatures associated with current generation multi core processors and with the inevitable increase in the number of cores in future, the feasibility and practicality of using these conventional techniques alone, comes into question. Dynamic Thermal Management Techniques such as Dynamic Voltage and Frequency Scaling are more or less successful in containing die temperature below a certain threshold but

involve some throttling to lower power consumption, resulting in performance degradation. Spreading of activity across a many-core chip is increasingly being looked upon as a way to contain chip temperatures without degrading performance.

In this study, an attempt to analyze the impact on temperature by performing power migration is made. It seeks to examine the possibility of increasing the number of active cores at any time, thereby further improving performance, while using power migration to maintain uniform power distribution and minimize overall temperature across the chip.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Figure Page

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 What is Packaging?

Electronic Packaging is the enclosing or interconnecting of microelectronic components, integrated circuit (IC) chips, and other passive devices, onto circuit cards and system level boards resulting in the production of the final electronic product or system. [1]

Electronic packaging plays an important role in the development of electronic products and serves a multitude of purposes such as : (i) Providing electrical connectivity, (ii) Facilitate improved handling and assembly, (iii) Heat dissipation, (iv) protection from environmental effects.[2]

Packaging is prevalent in various walks of life such as personal electronic devices, computers, telecommunications, medical electronics, aerospace systems and many others. [3].

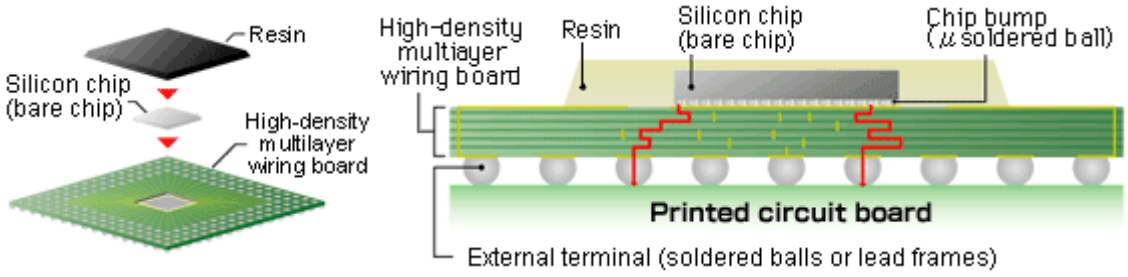


Figure 1.1 Illustration of an Electronic Package [2]

1.2 Levels in Packaging Hierarchy

Electronic packaging can be looked at predominantly at two levels : at the device or IC level and at the system level [3]. Explicitly, it can be classified into the following categories:

(i) Level 1: Device or IC Level

Packaging at this level involves the interconnection, powering, cooling and protection of the IC. It can be a single chip module or a multi chip module, comprising more than one chip. The packaging thus acts as an IC "carrier", or packaged ICs as they are called and are ready for assembly onto a system-level board, which leads to the next level in the packaging hierarchy.

(ii) Level 2: Printed Wiring Board

A typical system comprises of several active and passive components, in addition to packaged ICs, assembled and interconnected on a system-level board, also called a Printed Wiring Board (PWB). The assembly is done on both the top and underside of the board and the interconnection is achieved through conductor wiring, to form one interconnected system.

(iii) Level 3: System

Today's advancement in design complexity means that a single system level board or PWB may not accommodate all the components required to form a whole system. Multiple PWBs are themselves placed on a "motherboard" or backplane and interconnected through means of connectors and cables, creating a whole system.

This packaging hierarchy is illustrated in Figure 1.2[3]

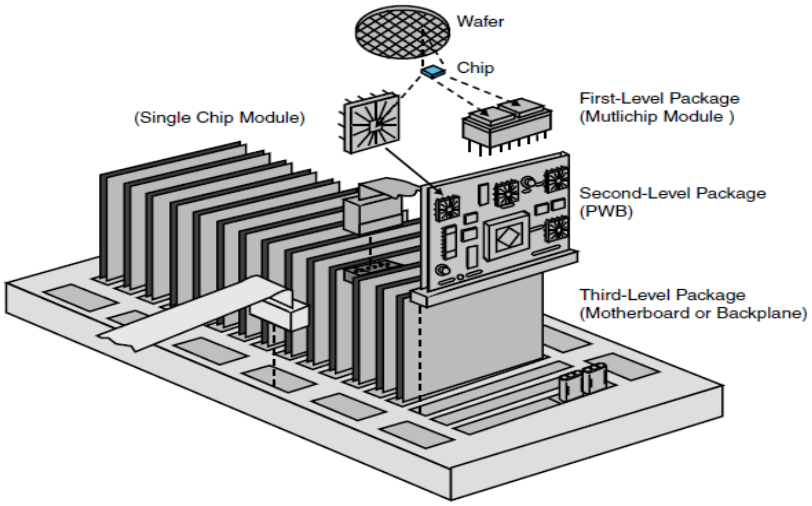


Figure 1.2 Packaging Hierarchy [3]

1.3 Types of IC Packages

IC packages vary in various aspects such as structure, material, fabrication technique, size, and thickness, number of I/O connections, electrical and thermal capability, reliability and cost. The choice of these parameters is contingent to the type of IC being packaged. In general, IC packages are classified into two categories: (i) through-hole and (ii) surface mount, based on the mounting methodology used to plant them onto a PWD. If inserted through holes on the board surface, the package is called a through-hole package, whereas if mounted on the surface of the PWB, it is called a surface mount package. Surface mount packages are more advantageous as both sides of the PWB can be used, thereby achieving higher packing density.

Figure 1.3 shows this classification[3].

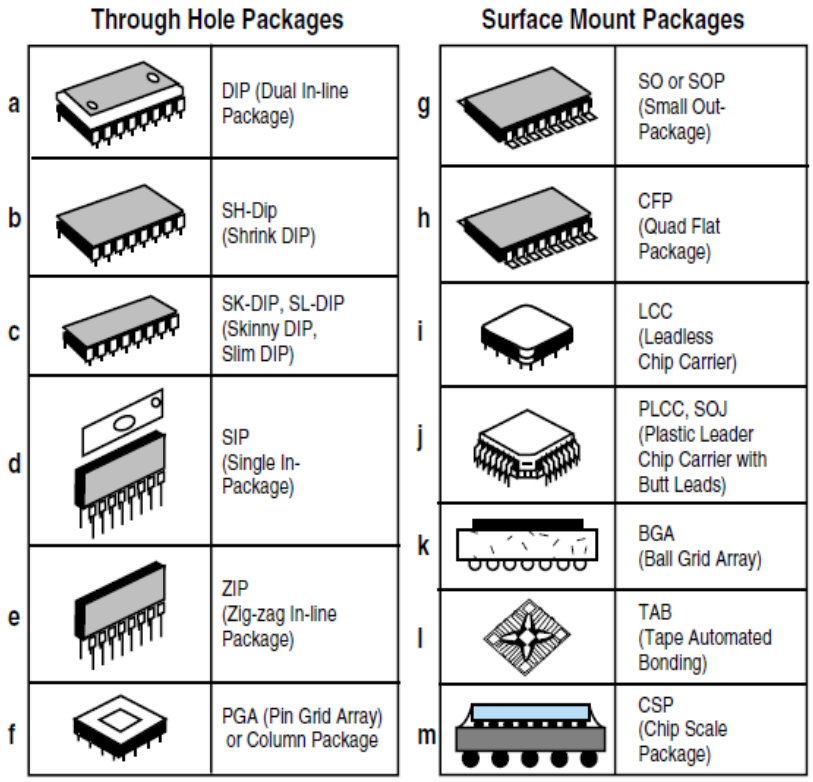


Figure 1.3 Types of IC Packages [3]

1.4 Single Chip Package

Since, this work deals with a single chip module, a brief insight into the evolution of single chip packages is provided. The evolution of single chip packages has taken three primary paths:

1. Peripheral:

Dual In Line Package (DIP)->Plastic Leader Chip Carrier (PLCC)->Quad-Flat Package (QFP)

DIP:

- First package invented by Bryant Rogers of Fairchild in the 1960s.
- Comes in ceramic or plastic versions, was one of the earliest industry standards.
- Applications include memory and logic controllers.
- Not favorable if space is at a premium.

QFP:

- Has a higher pin count than DIP.
- Ceramic QFP provides resistance to high temperatures and humidity.

2. Area Array:

Ceramic and Plastic Pin Grid Array (PGA) -> Ball Grid Array (BGA)

PGA:

- First high volume PGA package was implemented in 1982.
- Was the primary package solution till the late 1990s.

BGA:

- Interconnects are formed by an array of solder balls located within the package boundaries.
- Consumes lesser space on the PWB.
- Is further classified into plastic (PBGA), ceramic (CBGA) and tape (TBGA), based on the substrate.

3. Flip Chip:

Ceramic flip chip to organic flip chip is explained in a bit more detailed manner in the following section.

1.5 Flip Chip Package

The flip chip process was first introduced for ceramic substrates as the Solid Logic Technology by IBM in 1962. It was later, in 1970, that IBM converted this to the Controlled Collapse Chip Connection, or C4, for ICs [3]. Flip chip technology is an advanced form of surface mount technology, in which bare semiconductor chips are turned upside down, and hence called flip chip (i.e., active face down), and bonded directly to a printed circuit board or chip carrier substrate. In contrast to wirebonding, which is a peripheral and time consuming bond technique, in which bonds are formed sequentially, the flip chip allows all I /Os to be connected simultaneously.

Flip chip interconnection is the connection of an integrated circuit chip to a carrier or substrate with the active face of the chip facing toward the substrate. Interconnection between the chip I /O and substrate is achieved using a bump structure on the chip and a bonding material, typically on the substrate, forming an electrical interconnection between the chip and the substrate. Flip chip bonding typically involves solder interconnections that make the electrical and mechanical connection between the chip and the carrier, although alternate material systems such as conductive adhesives can also be used.

1.6 Scope of this work

The objective of this work is Dynamic thermal management of a multicore processor. In particular, a technique known as "Core Hopping" is considered, in which the process "hops" from one core to another, thereby distributing the thermal load more equally and not compromising much on performance. Simply, put, an active core turns "off" and another, currently inactive, cooler core turns "on" at different intervals of time.

The thermal aspect of the problem alone is considered without going into the specifics of how core hopping actually takes place using the resources on an actual processor. The package under consideration is a flip chip package and the chip is discretized into 16 blocks of equal area, each representing a core. A heat flux is then assigned to each of the "cores" and

time and temperature based switching of power on the cores is performed to simulate core hopping and note the effect on temperature.

The simulations are performed using commercial CFD software and various cases such as one processor core functioning at a time, two cores operating simultaneously and 4 cores operating together, are considered for the time based switching case, in addition to temperature based switching.

En-route to explaining the problem and discussing the results, aspects of multi-core processors and thermal issues faced along with the thermal management techniques that exist are also presented.

Chapter 2 covers processor trends, the advent of multi-core processors, the shift to many-core processors and the thermal challenges arising thereof.

Chapter 3 discusses thermal management of multi-core and many core processors. Dynamic Thermal Management is dwelt upon in detail and the concept of Core Hopping is introduced.

Chapter 4 details the literature search that was performed with respect to the various aspects of the problem being dealt with.

Chapter 5 explains the designing and modeling of the set-up being used. It provides an insight into the approach adopted and the methodology followed in obtaining the model that has been used.

Chapter 6 presents the results and the associated discussion.

Chapter 7 gives the conclusion and future scope of this work.

CHAPTER 2

PROCESSOR TRENDS AND THERMAL CHALLENGES

2.1 The Transition from Single to Multi-Core

We live in an age of technology. From Laptops to PDAs, we demand the best - and most – in any electronic gadget. Increased competition for market share has resulted in various manufacturers packing in more and more applications into devices whose physical sizes are decreasing with every passing day.

Achieving this combination of versatility and compactness requires complex electronic design and invariably involves high power consumption. CMOS semiconductor technology scaling, limited performance of traditional instruction-level parallelism along with increased design complexity and power consumption have led to a reorientation of the microprocessor industry from a single, complex monolithic core to multiple cores on a single chip [5, 6]. Such multi-core chips are also called Chip Multiprocessors (CMPs).

Gordon Moore, co-founder of Intel, proposed a concept in his 1965 paper, that the numbers of transistors will double every year, a statement now known as Moore's Law [7]. Today's chips integrate billions of transistors on them [8]. The graphical representation for Moore's Law [9] and transistor integration capacity [8] are shown in Figure 2.1 and Figure 2.2 respectively

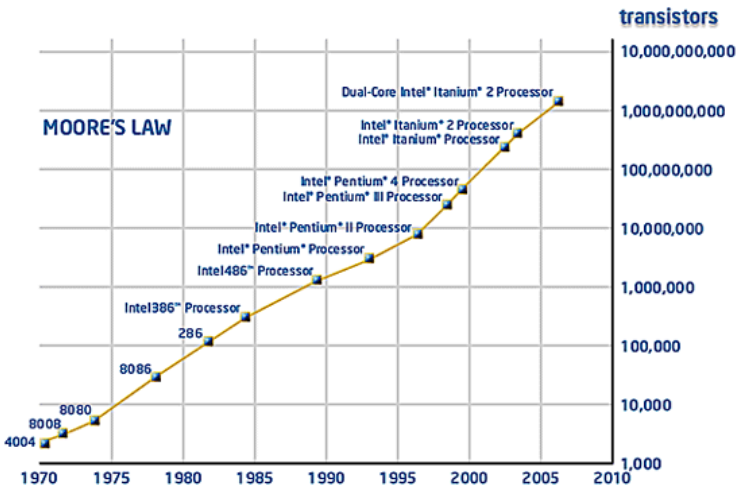


Figure 2.1 Moore's Law Projection [9]

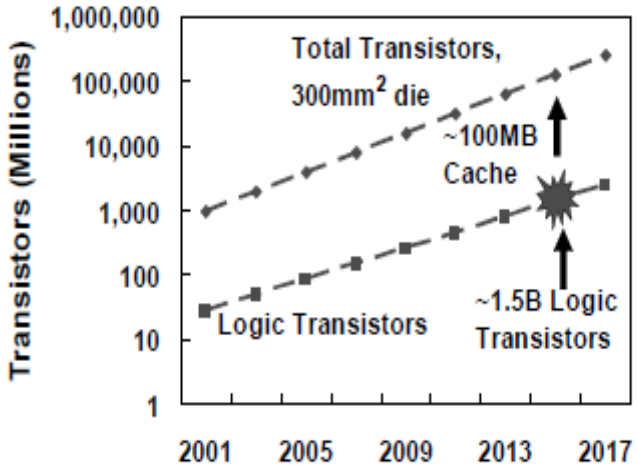


Figure 2.2 Transistor Integration Capacity [8]

CMPs are recognized in industry and academia alike as having the greatest potential for harnessing the billions of transistors now possible on a chip [6], so much so that they have now become a mainstay of the microprocessor industry today [10]. All major players in the microprocessor industry offer multi-core processors [6], for example, AMD with its Athlon TM 64 X2 dual core processor and Opteron 800 series processor for desktop PCs and servers respectively [11],Intel with its Core ™ 2 class of processors unveiled in 2006 [12] and IBM

starting with its POWER4 processor [13]. CMPs improve overall performance by exploiting fully, instruction-level parallelism and thread-level parallelism [6][10].

A multi-core architecture provides potentially linear performance improvement with complexity and power [8]. Two smaller cores instead of one large monolithic core provide up to a 70-80 % improvement in performance and also provide various other advantages such as selective operation, optimized supply voltage and frequency, computation and thermal load distribution and reduction in die temperature, all of which serve towards improving reliability [8].

2.2 From Multi-Core to Many-Core

Despite continued scaling of technology, leakage current and slow down in supply voltage scaling limit transistor performance [14]. Borkar [8] suggests that microarchitecture performance increase is dictated by Pollack's rule which states that performance increase is roughly proportional to square root of increase in complexity of the architecture. Thus, developing multi-core chips is not as trivial as replicating multiple complex cores on a die. Instead, integrating a large number of simpler, small cores, which albeit individually deliver lower performance, results in a higher overall throughput of the system as a whole.

Processor chips having more than four cores are termed as many-core processors. The diminishing return on performance and increasing area and power overhead incurred in scaling traditional single core processors means that processors nowadays incorporate multiple parallel processing cores on the same chip [15]. Every succeeding technology generation is predicted to hold double the number of cores on a silicon chip as the previous [8]. Current graphics processors already hold parallel processing units numbering in hundreds, such as NVIDIA with 240 parallel cores [16] and AMD with 800 processing units [17].

To ensure optimum parallel and serial performance, an asymmetric many-core architecture is preferred, consisting of numerous basic, simple cores handling parallel threads and one aggressive, complex, "primary" core dealing with serial computation [18]. Also, it is vital that the design is capable of dealing with variable amounts of parallelism to be applicable for

general purpose computing, for which reconfigurable chip multiprocessors facilitating dynamic combination of multiple simple cores into one primary core have been proposed [19][20].

2.3 Thermal Challenges in many-core processors

An important consequence of the technology scaling mentioned previously is the concurrent increase in power density. With the scaling of feature size into the sub-100 nm realm, difficulties in scaling transistor threshold voltage due to leakage current and reliability issues result in supply voltage scaling very slowly. This is termed non-ideal CMOS scaling [15]. Non-ideal CMOS technology scaling leads to an increase in power density from generation to generation [5]. The shrinking of chip geometry combined with transistor scaling in accordance with Moore's Law and increase in clock frequency and leakage current results in high power consumption and heat generation [6]. Also, transistor size scales faster than power per transistor [15].

Huang et al. [15] derive the expression for power consumed by a core with fixed architecture as

$$P_{n+1} = \left(\frac{Vdd_{n+1}}{Vdd_n}\right)^2 P_n \quad (1)$$

Where, V_{dd} is the supply voltage and n and n+1 denote technology generations.

The power density would scale from one generation to another as

$$PD_{n+1} = \left(\frac{1}{s}\right)^2 \left(\frac{Vdd_{n+1}}{Vdd_n}\right)^2 PD_n \quad (2)$$

Where, s is the technology feature size scaling factor.

The power dissipation of modern processors is increasing with the increase in clock frequency and transistor numbers needed for a given implementation [21]. The general trend suggests an increase in maximum power consumption by a processor by a factor of around 2X every four years [21]. This is as illustrated in Figure 2.3.

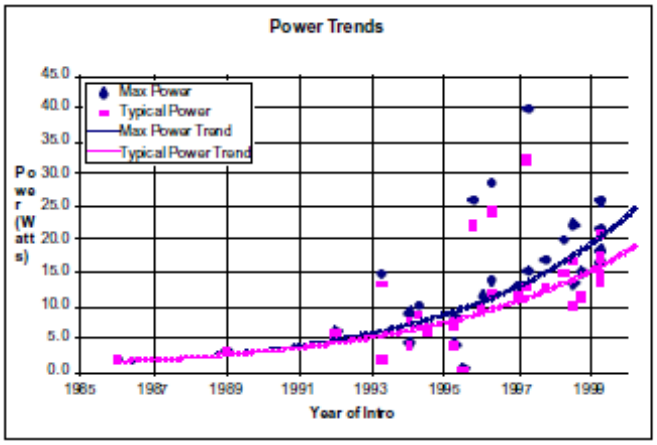


Figure 2.3 CPU Power Consumption Trend [21]

The advent of many-core processors only means an increase in power consumption and heat generated. Asymmetric many-cores, mentioned earlier, particularly pose a thermal problem as the more complex primary core will consume more power than the other simple cores, resulting in localized concentration of heat called hot spots [15]. Huang et al. [22] have shown that large cores are hotter than small cores with the same power density. Huang et al. [5] also point out that speed boosting techniques such as the "turbo mode" used in the Intel Nehalem processors, used to selectively increase the performance of a single core when other cores in a CMP are idle, increase power density and temperature.

The components on a chip not comprising the core, or "uncore" components such as the on-chip network, lower level caches and I/O pads and drivers also consume a significant amount of power [15]. As the number of cores keeps on increasing in a CMP, the activity in the "uncore" components and consequently, the power consumed by these components also increases.

Another crucial aspect of the thermal problem in CMPs is the issue of non-uniform power distribution. In addition to the creation of hot spots due to asymmetric design, Cho et al. [23] make the point that not all cores in a many-core processor will be functioning at all times.

Also, different locations of “active” cores at different times result in temporal and spatial non-uniformity in power consumption.

The reduction of dimensions in forthcoming silicon technologies means there will be significant thermal coupling between neighboring cores, which will aggravate the occurrence of hot spots in future nanoscale technologies [6].

Gunther et.al [21] also make the important observation that with increase in power, there is a non-linear relationship between cooling solutions and the associated cost, as is shown in Figure 2.4. This emphasizes the importance of containing power consumption within a certain limit.

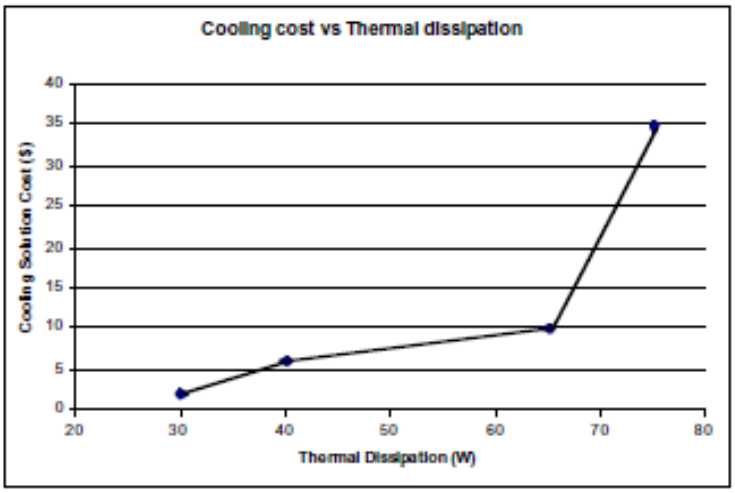


Figure 2.4 Cooling costs with increasing thermal dissipation [21]

CHAPTER 3

THERMAL MANAGEMENT OF MULTI-CORE PROCESSORS

3.1 Introduction

Increase in temperature results in reduced performance and reliability and increased leakage currents and cooling costs. Given the technology scaling trends and continuous increase in the number of cores and the consequent exponential increase in power density with Moore's Law, there is a pressing urgency to deal with thermal challenges [15][24].

Existing cooling methodologies have focused on mechanical techniques such as heat sinks and fans, which are hard pressed to alleviate the high temperatures associated with current generation multi-core processors. Techniques such as heat pipes and Peltier devices are also not feasible when confronted with today's rising processor numbers and temperatures.

[25] and [26] have dwelt at length on Dynamic Thermal Management.

The move towards billion transistor microprocessors means that the thermal challenges must be addressed at all design cycle levels. Increased power dissipation levels leads to increased system complexity owing to the cost and complexity of thermal packaging and the power distribution network required on the chip. [25] point out that beyond 40 W, any additional power dissipation will result in cost increase by more than $1 per W.

Also, there is growing disparity between the maximum possible power dissipation and the typical power dissipation. Wherein thus far, designs were geared towards dealing with the maximum power dissipation scenario, it is becoming increasingly apparent that techniques geared toward dealing with actual power dissipation are more favorable.

CMPs, albeit causing the latest severe thermal challenges, also offer a new and exciting scenario for DTM techniques.

3.2 Dynamic Thermal Management

Dynamic Thermal Management (DTM) encompasses various hardware and software techniques enabled at run-time to control a chip's operating temperature [25]. Current microprocessors are configured to slow down or even power down when a pre-determined temperature threshold is reached, thereby crippling performance [26]. The purpose of DTM is to provide inexpensive thermal management to reliably reduce power and maintain the chip below a safe operating temperature while effecting as little impact on performance as possible.

DTM enables design closer to actual power dissipation levels as opposed to unrealistic 'worst case' scenarios. Therefore, along with decreasing the cost of packaging, it also leverages techniques such as clock gating better in order to reduce average power. The core of any DTM system is how it responds to a thermal emergency; i.e. the technique it employs to ameliorate the thermal condition.

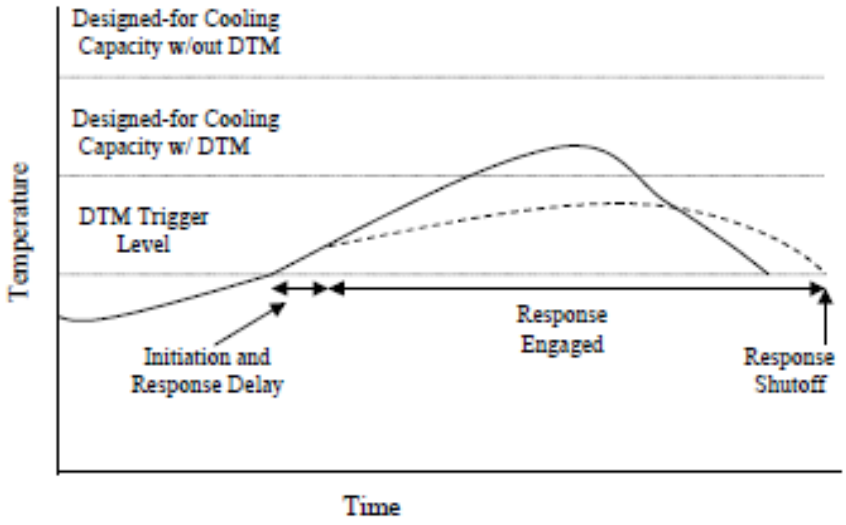


Figure 3.1 Overview of DTM Technique [25]

The plot of chip temperature versus time (in cycles) shown in Figure. 3.1 provides a graphical description of the concept of DTM. It illustrates that temperature can be brought down

if DTM is implemented. The DTM trigger level is the temperature at which DTM techniques are engaged. The two curves in the Figure represent some sequence of code being executed. The solid curve represents execution without employing DTM while the dashed curve represents execution with DTM implemented. Both curves are identical till the DTM trigger level is exceeded. After this, save for a small delay period while the response engages, the curves diverge. We see that without DTM, the chip reaches a much higher temperature whereas in the case of DTM being employed, the power dissipation is reduced and consequently, so is the chip temperature, which never exceeds the designed-for cooling capacity [25]. Eventually, the temperature does decrease in both cases and the DTM response is dis-engaged after a while with some performance delay with respect to the Non-DTM curve, which highlights the fact that, despite being effective in curbing chip temperatures, DTM does cause some degradation in performance.

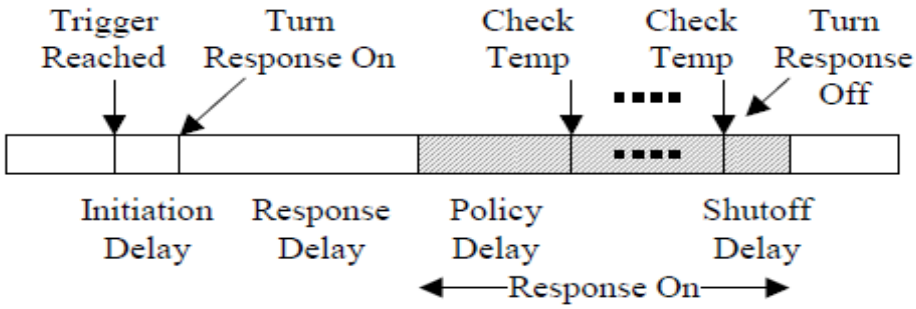


Figure 3.2 Mechanism of DTM [25]

The steps involved in the implementation of any DTM technique are illustrated in Figure.3.2. At a pre-set temperature, DTM is triggered. There is some initiation delay while the triggering event is interpreted. Then the DTM response begins. Depending upon the type of response, there is also some response delay and also, once the temperature has dipped below the emergency threshold level, there is a policy delay which refers to the number of cycles the system is designed to wait before deciding to turn off the DTM mechanism. This too involves a

delay period termed shutoff delay while the voltage and frequency are adjusted [25]. All these delays contribute to the performance degradation resulting from DTM.

Some of the DTM response mechanisms are frequency scaling, voltage scaling and throttling. [25] evaluated the DTM scheme based on (i) the number of cycles in which the thermal emergency threshold is exceeded and (ii) overall performance loss incurred.

DTM however does have a flip-side in that it is predominantly a trade-off between cutting power and hitting performance, so DTM techniques do invariably cause some performance degradation.

Micro-architectural temperature control techniques are primarily classified into 2 categories:

(i) Reducing the amount of energy dissipated.

(ii) Distributing the processor activity over the chip area.

These are detailed as follows.

3.2.1 Reducing the amount of energy dissipated

An example of this is Stop and Go [25], which is employed to bring down peak temperature. Whenever a critical temperature is reached, the core is 'stopped'. This is achieved by clock gating that particular core. This involves freezing all operations and turning off all signals. While this is done, processor state is maintained [24]. This however, only reduces dynamic power while leakage continues [26]. A more aggressive option would be to cut off supply voltage to the core while taking care to save architectural state prior to supply voltage cut off [26].

Dynamic power dissipated is given by [21]

$$P_d = CV^2f$$

Where C: Capacitance being switched per clock cycle

V: Voltage

f: switching frequency

Thus, as V changes, power consumption changes quadratically and as f changes, power consumption changes linearly. DVFS is effective in the case of high thermal resistance solutions wherein global heating is reduced, achieving faster cool down times. Of course, throttling down on voltage or clock frequency will imply some performance degradation.

[25] also mention Decode Throttling, which is a DTM technique used in the PowerPC G3 microprocessor, which restricts the flow of instructions to the processor core while relying on clock gating to reduce power dissipation and I-cache toggling, which involves disabling the instruction fetch unit and using the instruction fetch queue to feed the pipeline.

Conventional DVFS, relying on off-chip regulators, encounters temporal granularity and cost limitations when considering future multi core systems [27].

3.2.2 Distributing processor activity over the chip area

By constantly relocating activity, which is the source of heat, total temperature can be reduced as the average temperature for each particular part of the chip is reduced. Examples of this technique include Thread Migration or Core Hopping.

As most DTM techniques involve some performance degradation, with the advent of multiple cores on a chip, an emphasis on spreading the heat generating activity across the area of a many core chip as a means of mitigating thermal dissipation and non uniform power issues without compromising on performance, is gaining prominence [28].

[27] discusses thread motion (TM) in detail.

TM is a fine-grained power management scheme for use in CMPs that enables rapid thread movement to adapt the time varying computing needs of different applications running on simple homogenous cores with heterogeneous power-performance capabilities [27]. TM seeks to exploit the spatial slack offered by CMPs and requires a mechanism to transfer architectural state from one core to another [26].

Compared to conventional regulator based DVFS schemes, TM can be applied at much finer time intervals, thereby maximizing performance across different power budget constraints [27].

The goal of TM is to extend workload-based power management into the nanosecond realm by exploiting variations arising from micro-architectural events and, for a given power budget, gives up to 20 percent better performance than coarse-grained DVFS [27].

Different TM policies are mentioned in [26]. [27] mention some benefits and limitations of TM. The benefits include enabling fine grained tracking of program behavior that is not offered by fine grained DVFS thereby offering benefits sensitive to different types of workloads, increase in system throughput for different power envelopes, providing an intermediate voltage with only 2 VF settings thus cutting down on system costs. Limitations being that TM relies on rapid movement of applications between cores thereby constrained to systems featuring simple homogenous cores with relatively small amounts of architected state, targets power constrained multi core systems and incurs power overhead due to increased access to register files, I level caches and L2 cache.

TM is very effective when the temperature of a small number of cores needs to be changed since it acts on a per-core basis [26].

3.3 Core Hopping

Core Hopping is a coarser time scale application of TM. It was proposed in 2002 at Intel [29] as a means of reducing chip temperature by having threads jump around from core to core thereby distributing the heat. This enables key transistors to stay cooler, heat distribution is more uniform and overall performance improves [30].

Designing core hopping was made possible with the introduction of an 80 core chip called the Tera-Scale Teraflop Prototype at Intel [31].

[28] carried out core hopping on a dual core POWER5 processor and determined that, because of two cores sharing a common on-chip L2 cache, core hopping does not incur

expensive L2 cache misses and also the performance cost of warming up L1 cache is insignificant. Consequently, performance degradation was measured to be less than 3 percent.

CMP cores typically share caches for better hit ratios, thereby reducing migration penalty of a task from one core to another, rendering core hopping as a viable thermal mitigation technique.

CHAPTER 4

LITERATURE REVIEW

4.1 Introduction

The advent of multi core processor technology and the tremendous potential to harness performance and the overwhelming challenge of thermal management has been dwelt upon in academia and industry alike and spurred active research into the same.

A comprehensive literature search was performed not just pertaining to the problem at hand alone, but with a view towards gaining a broad perspective on trends in many core processor technologies, the challenges involved and the solutions examined and implemented. A detail of the literature review performed is described as follows, in that order.

4.2 Trend towards multi core and many core

Borkar [8] discusses the many core architecture in terms of performance, power and design. This work points out that Moore's Law continues with technology scaling, increasing transistor performance and integration to increase frequency and realize complex architectures. Also, the energy consumed per logic operation is reduced to limit power dissipation. The move from multi core to many core is not as trivial as integrating multiple cores on a die. Instead, it involves the integration of a lot of smaller cores that, individually, deliver lower performance than an individual complex core but have a higher throughput when taken together. The rule governing increase in performance due to use of microarchitectures is the Pollack's Rule which states that performance increase is roughly proportional to square root of increase in complexity. Pollack's rule is illustrated in Figure 4.1.

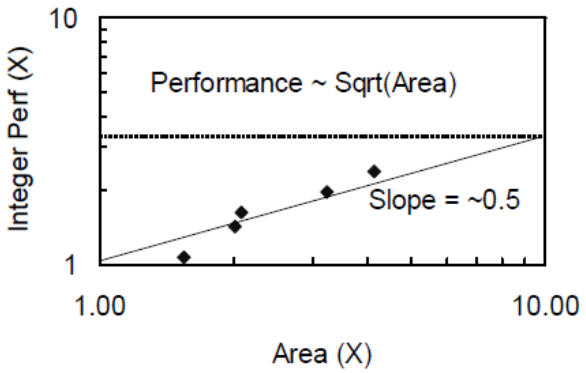


Figure 4.1 Pollack's Rule [8]

Applying Pollack's rule inversely, it can be stated that performance of a smaller core reduces as square root of size while power reduction is linear thereby tilting the balance in favor of power reduction over performance degradation. In fact, it is stated that two smaller processor cores, as opposed to a large monolithic core offers up to 70 to 80 percent more performance as compared to 40 percent for a single large core. The author also points out that multiprocessor provide several advantages such as, each processor could be turned on or off individually, thereby saving power, voltage and frequency can be optimized, load can be balanced among various cores and can achieve lower overall temperatures increasing reliability and efficiency.

Huang et al [15] point out that one consequence of technology scaling has been the increased power per unit area, thereby making improving chip level cooling solutions a priority. In addition to power scaling of cores, they also mention the contribution of the "uncore" components such as the on chip network and lower level caches towards higher power consumption in the era of multicore processors. With the increase in number of cores, the activity and consequently, the thermal load on these uncore components will also increase.

The authors also examined the limits of various cooling solutions and concluded that air cooling alone will no longer be able to control the rising temperatures. With the advent of 3D stacking, air cooling scenarios are only going to get worse. They also mention that micro channel cooling appears to b a feasible methodology especially in the case of 3D stacking,

provided the technology develops in the near future to become practical and affordable. The authors also proposed looking into alternate technologies such as phase change cooling, spot cooling for localized heat sources and thermo-electric cooling.

Parkhurst et al [32] mention that process scaling has been the underlying enabler of this trend towards multicore processors starting in the early 90's. Starting with the 0.8 um process in 1992, process scaling has occurred at an interval of about 24 months enabling a consistent feature size reduction factor of 0.7.

The authors state that, along with dynamic power, static power also continues to increase due to transistor source to drain and gate leakage. This leakage too, increases with scaling and is poised to become a significant portion of the overall power budget. They also point out that power density is yet another exponential associated with feature size scaling.

Along with process scaling, the authors also describe parallelism as being instrumental in providing impetus to the movement towards multi core. Parallelism is looked at as one of the best ways to address power issues while maintaining performance as higher throughput can be achieved at lower operating voltage and frequency.

Both [15] and [32] mention heterogeneous cores in a multi core chip dedicated to running particular, possibly more intense applications. The heterogeneity may be in terms of core complexity and power level.

Huang et al [5] point out that asymmetric manycores particularly create a thermal disadvantage as the "primary" or more complex cores create localized hot spots due to higher power consumption. They also mention that, with the same power density, large cores are hotter than small cores. This is because the equivalent thermal resistance of smaller cores is reduced due to a thermo-spatial low pass filtering effect.

What makes the situation worse, according to Huang et al [5], is techniques such as Intel's "Turbo Mode" used to boost processing speed by selectively increasing supply voltage

and frequency to those cores that are active. This, albeit improving performance, further increases hot spot problems.

Janicki et al [6] also state that as feature sizes scale down from 65 nm to 15 nm, the thermal coupling between cores will increase.

4.3 Dynamic Thermal Management

[25], [26] and [24] discuss Dynamic Thermal Management (DTM) in general.

Brooks and Martonosi [25] explore the effects of hardware and software techniques and come up with areas that need to be looked into for research work in DTM. They illustrate the basic steps involved in the implementation of any generic DTM technique and state that the main design choices in implementing an effective DTM system are in the areas of selecting simple, effective triggers, identifying response mechanisms and coming up with a policy on turning the mechanisms off and on judiciously.

The same authors have come up with an architectural level power modeling tool called Wattch [33] which is utilized in their work.

Some of the trigger mechanisms mentioned are temperature sensors, on-chip activity counters and dynamic profiling. The authors define a “thermal Trigger” as the temperature at which the response mechanism is to be triggered so as to begin cooling the processor.

Some of the response mechanisms examined are clock frequency scaling, voltage and frequency scaling, decode throttling, speculation control and I-cache toggling, which restrict the flow of instructions to the processor core.

The number of cycles in which the thermal emergency threshold is exceeded, the overall performance loss and the total execution time are the metrics used to evaluate the DTM schemes. It is concluded that the trigger overhead of frequency and voltage scaling techniques is significantly higher than microarchitectural techniques and to eliminate the trigger overhead, the trigger mechanism must be integrated into the microarchitecture.

The authors conclude by saying that for effective DTM, choosing the correct trigger mechanism and its activation time proves significant.

Donald and Martonosi [24] explore various thermal management techniques that seek to exploit the distributed nature of multi core processors. They classify thermal management techniques in terms of throttling or migration policies. Using processor and power modeling and temperature calculating tools and performing transient analysis, various workloads are simulated. The goal is to maximize performance while remaining under a fixed temperature constraint.

It is determined that distributed DVFS results in more than a 2.5X increase in throughput over distributed stop and go and both counter based and thermal trend based migration reflect around 2X improvement over stop and go policies. It is therefore concluded that distributed DVFS in conjunction with a sensor based migration policy offers the best option for thermal control illustrating the benefit of hardware-software collaboration.

26 et al [26] present an in-depth study of different DTM techniques to be used in multicore processors.

They create a detailed thermal model of a 16 core chip multiprocessor and evaluate the performance of thermal management schemes under two conditions; one wherein the processor is running multiple independent applications and second, where the processor is handling two parallel applications, one comprising “cold” threads, the other comprising “hot” threads.

The thermal management schemes evaluated are Thread Migration (TM), DVFS and combined TM and DVFS schemes. The TM policies that are evaluated, are, in turn, classified as

(i) Rotation: threads are migrated sequentially.

(ii) Temperature Based: thread assignment based on temperature

(iii) Counter Based: thread assignment based on temperature difference

(iv) Power Based: Threads are measured with respect to power and assigned to cores based on temperature.

DVFS techniques in turn are classified as local or global based on application to individual cores or the whole chip.

The paper concludes that for expensive thermal solutions, with low thermal resistance, local DVFS combined with counter based TM offers the best option whereas in the case of cheap thermal solutions, local DVFS alone is the preferred option. It also states that TM and DVFS hold a promise of boosting performance while controlling temperature.

4.4 Thread Migration

Within DTM, it was desired to gain further knowledge regarding migration of activity among the various cores in a multi core processor, as a step towards studying core hopping. Some of the works dealing with Thread Migration or Activity Migration are mentioned as follows. Choi et al [28] investigated temporal and spatial hot spot mitigation schemes, in which an Operating System (OS) – level scheduler was used for a POWER5 processor. To demonstrate leveraging of spatial heat slack, single threaded programs were made to “hop” between the two cores after a fixed time interval on the POWER5 processor. It was noted that core hopping results in temperature changes of up to 5.5 degrees Celsius while causing less than 1 % average slowdown. Since the dual core POWER5 has a shared on-chip L2 cache, the performance degradation was measured to be less than 3 %. Thus it was concluded that, with shared caches reducing the migration penalty of a task from core to core, core hopping would be a viable thermal management technique.

Powell et al [34] proposed heat-and-run thread migration (HRTM) which migrates threads away from overheated cores to cooler cores, thereby improving performance and enabling uniform power distribution across the chip. HRTM thus effectively utilizes the spatial slack provided by a simultaneously multi-threaded chip multi processor (CMP). An analytical example was provided using two cores. TM was performed through the OS, again. The base

simulator used was again Wattch [25], along with HotSpot [35] for power density and temperature sensing. The authors stated that HRTM improves throughput over techniques such as stop-and-go by exploiting spatial slack and that, for a 4 core CMP running 5 threads, HRTM achieved on an average, 9 % higher throughput than stop-and-go and 6 % higher average throughput than dynamic voltage scaling.

Rangan et al [27] studied TM on a 16 core CMP system developed by Sun Microsystems featuring core clusters sharing a last level cache, each cluster containing 4 cores. An important conclusion reached by the authors is that TM achieves the effect of having multiple voltage-frequency (VF) domains with just two voltage levels. Performance benefits of up to 20 % are observed when compared to a static OS-driven DVFS scheme.

Shayesteh et al [36] explored the use of a thermally-triggered core-swapping technique on a dual micro-core architecture, which features a small, fast pipeline augmented with helper engines featuring the large structures factored out of the original core, resulting in a micro-core. The helper engines reduce the overhead of swapping by buffering core state during the swapping process. Core swapping is found to be able to maintain temperature below the threshold as helper engines rarely heat up to critical temperatures.

Heo et al [37] investigated reduction of power density using activity migration (AM) using a RC thermal model. AM is implemented by pingponging between two sites. It was shown that sustainable power dissipation can be increased by nearly a factor of two for a given junction temperature limit. Peak die temperature could be reduced by around 12 degree Celsius at the same clock frequency.

Cho et al [23] presented a proactive spatiotemporal power multiplexing method to achieve a lower peak temperature and a more uniform thermal field on the chip. The basic concept of spatiotemporal power multiplexing is to change the location of power dissipation after a fixed time interval or timeslice while maintaining the throughput during the redistribution. The

migration was performed for any number of active cores irrespective of the temperature or thermal load on each individual core. A core was turned “on” or “off” using clock gating.

A tile type 256 core processor is discretized into a 16 X 16 array. The total power of the chip is kept constant. For a given number of active cores, the locations of these active cores are changed after every fixed time interval resulting in a time varying spatial power map for the chip. The time varying spatial power maps are then coupled to the HotSpot [35] thermal simulator and transient thermal simulations are performed to study the spatiotemporal variation in the thermal field.

The migration policy was random. The maximum temperature, spatial and temporal temperature difference were computed and the effect of timeslice was studied.

It was found that, for a given number of active cores, a smaller timeslice results in reduced maximum temperature and spatiotemporal non-uniformity. It also helps to reduce core-to-core delay and leakage.

CHAPTER 5

DESIGN AND MODELING

5.1 Description of the Assembly

To implement the model being considered, a basic flip chip package was used. The model was obtained by scaling the model developed by Karajgikar et al [38] [39]. The set up consists of a die, a thermal interface material (TIM), a heat spreader, and another thermal interface material crowned by a heat sink on the top. In the current analysis, it is assumed that heat dissipation occurs only through the top of the package. An effective heat transfer coefficient of 1200 W/m^2K was defined at the top surface of the heat sink.

The dimensions and properties of the various components are listed in table 5.1 and a pictorial representation of the set-up is shown in Figure 5.1.

Table 5.1 Dimensions and properties of the components in the model

Component	Dimension (mm)	Thickness (mm)	Conductivity (W/mK)
Die	20 x 20	0.75	120
TIM I	20 x 20	0.025	50
Heat Spreader	52 x 52	1.8	390
TIM II	52 x 52	0.075	3
Heat Sink	85 x 85	6.35	390

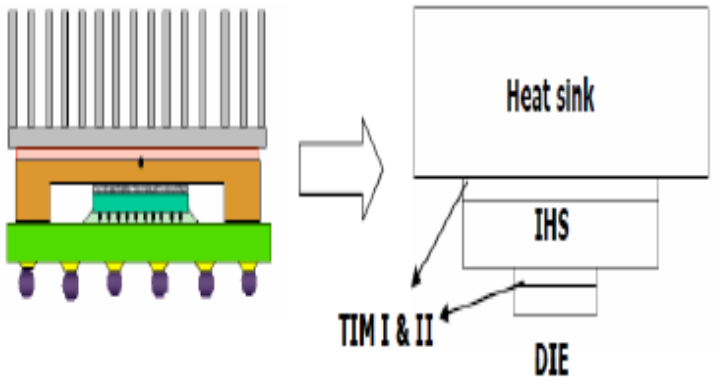


Figure 5.1 Set-Up of the model used [38]

5.2 Modeling Methodology

The above described set-up was modeled in Icepak 4.4.8. Blocks were created to represent the die, TIMs, heat spreader and heat sink. The dimensions were provided according to the table given in the previous section and also, the appropriate thermal properties were accorded to the blocks. The heat sink was modeled simply by a block instead of using an elaborate finned model. The excess area afforded by the fins was instead compensated for by using a higher value of the effective heat transfer coefficient. The surface of the die was discretized into 16 blocks of equal area, each representing a core and modeled by a heat source.

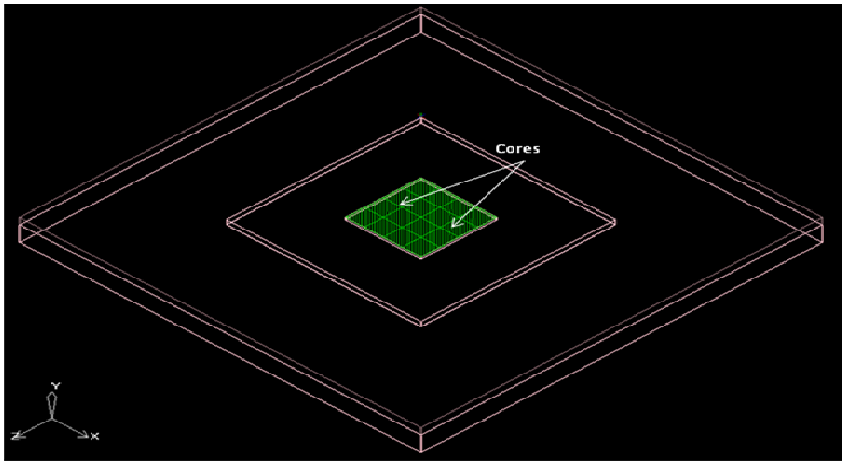


Figure 5.2 Isometric view of the Flip chip package

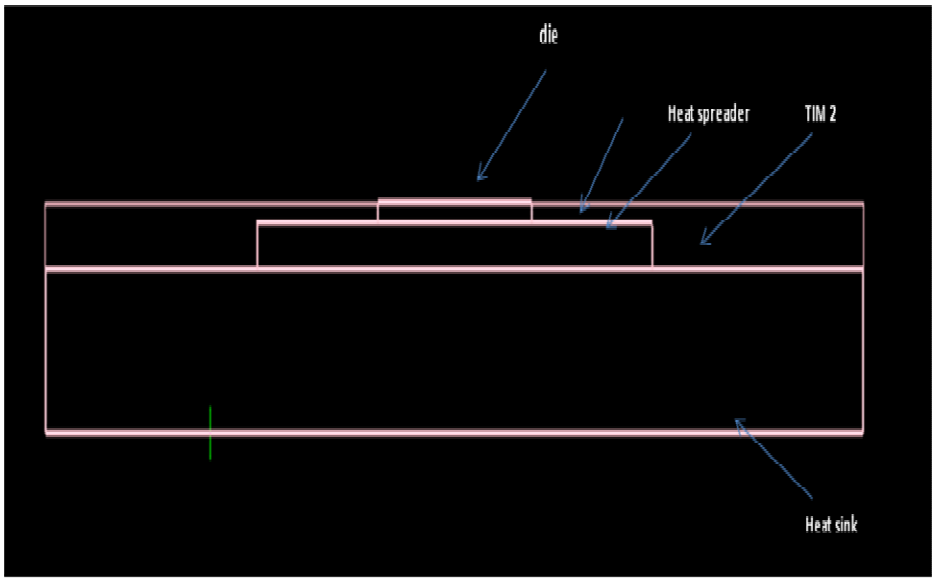


Figure 5.3 Side view of the Flip chip package

The heat load given on each core was 1 W/mm^2. The whole model is enclosed in a hollow block so as to make the problem a conduction problem. The model as created in Icepak is shown in Figures 5.2 and 5.3.

CHAPTER 6

RESULTS AND DISCUSSION

6.1 Introduction

Two cases were considered: Time-based hopping, in which the activity hop was performed after a fixed time interval and temperature based hopping, in which the activity hop was performed based on the temperature of the active core.

6.2 Results

6.2.1Time Based Hopping

Transient analysis was carried out for a multitude of cases involving the 16 core model generated using ICEPAK 4.4.8. Various combinations with the number of cores active at any time were tried out. The different cases tested were:

1. 1 core active at any time.
    A. Processors turned on in sequence.
    B. Processors turned on in a random sequence.
2. 2 cores active at any time.
    A. Adjacent processors working at the same time.
    B. Pair of processors turned on in a random sequence.
3. 4 cores active at any time.
    A. Scheme 1: Outer to inner sequence.
    B. Scheme 2: Random grouping.
4. Asymmetric Cores.
5. One core starting operation before another ends.

The objective was to make the activity hop from core to core and observe the effect of active core location on temperature. In each case, the simulations were run for 500 seconds.

The cases run and the temperature data obtained in each case are listed as follows, followed subsequently by a discussion of the observations made.

Case 1: One core active at a time

In this case, each core was turned on and off in sequence for a fixed time interval, such that at any point in time, there was only one core functioning, and consequently, dissipating heat. A core is turned on at the beginning of a ten second interval and then turned off at the end of ten seconds. Then, the next core is turned on and kept active for a period of ten seconds and so on. Once a full cycle of all 16 cores is completed, the cycle executes itself again and continues till the end of the simulation period, which as mentioned earlier, was 500 seconds.

A. Processors turned on in sequence

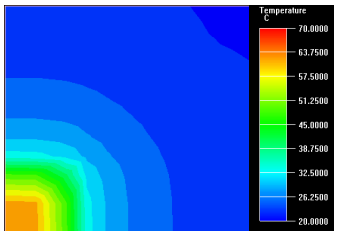
The cores are started up and shut down one after the other in sequence starting from core 1 to core 16. The active periods and the maximum temperature observed are presented in table 6.1.

Table 6.1 Single core activated: scheme 1

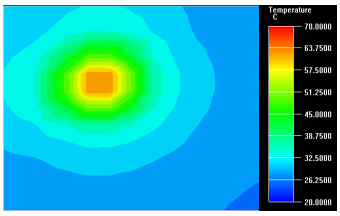
Core Active	Time-span for which core is ON (s)
1	0 - 10
2	10 - 20
3	20 - 30
4	30 - 40
5	40 - 50

Table 6.1 *Continued*

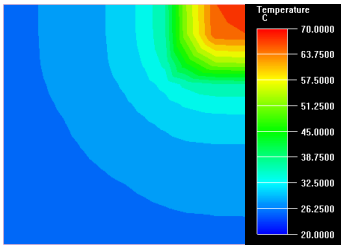
6	50 - 60
7	60 - 70
8	70 - 80
9	80 - 90
10	90 - 100
11	100 - 110
12	110 - 120
13	120 - 130
14	130 - 140
15	140 - 150
16	150 - 160
For Each Core : T_{ON} = 10 s T_{OFF}= 150 s	
Maximum Temperature Attained : 37.7 ^{0}C	



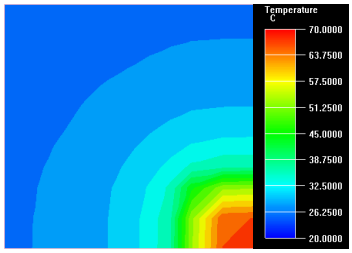
(a)



(b)



(c)



(d)

Figure 6.1 Simulation of Scheme 1 (a) t = 6 s (b) t = 158 s (c) t = 348 s (d) t = 500 s

B. Processors following a random sequence

In this case, although only one core is active at any point in time, the order of activation of the various cores is random. One such order is given below in table 6.2.

Table 6.2 Single core activated: scheme 2

Core Active	Time-span for which core is ON (s)
1	0 - 10
4	10 - 20
16	20 - 30
13	30 - 40
5	40 - 50
8	50 - 60
12	60 - 70
9	70 - 80
2	80 - 90
15	90 - 100
7	100 - 110
14	110 - 120
6	120 - 130
11	130 - 140
3	140 - 150
10	150 - 160
For Each Core : T_{ON} = 10 s T_{OFF}= 150 s	
Maximum Temperature Attained : 37.7 ^{0}C	

Case 2: Two cores active at a time

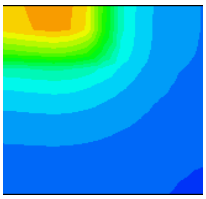
In this case, a pair of cores was turned on and off in sequence for a fixed time interval, such that at any point in time, there were two cores functioning, generating heat. A pair is turned on at the beginning of a ten second interval and then turned off at the end of ten seconds. Then, the next pair is turned on and kept active for a period of ten seconds and so on. Once a full cycle of all 8 pairs (16 cores) is completed, the cycle executes itself again and continues till the end of the simulation period, which as mentioned earlier, was 500 seconds.

A. Adjacent processors turned on

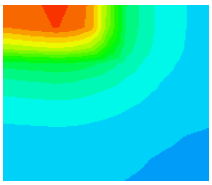
First, the case where the two active cores are adjacent cores is considered. The active periods and the maximum temperature observed are presented in table 6.3.

Table 6.3 Two cores activated: scheme 1

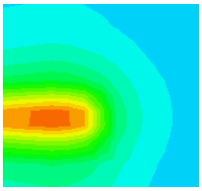
Cores Active	Time-span for which pair is ON (s)
1 and 2	0 - 10
15 and 16	10 - 20
3 and 4	20 - 30
9 and 10	30 - 40
7 and 8	40 - 50
13 and 14	50 - 60
5 and 6	60 - 70
11 and 12	70 - 80
For Each Core Pair : T_{ON} = 10 s T_{OFF} = 70 s	
Maximum Temperature Attained : 44.0 ^{0}C	



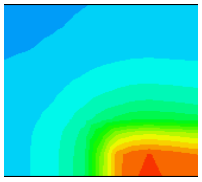
(a)



(b)



(c)



(d)

Figure 6.2 Simulation of Scheme 2 (a) t = 10 s (b) t = 168 s (c) t = 354 s (d) t = 500 s

B. Random pairs active

In this case, although a pair of cores is active at any point in time, the cores are deliberately chosen such that they are not neighboring cores and the selection and order of activation of the various pairs is random. One such pairing and order of activation is given below in table 6.4.

Table 6.4 Two cores activated: scheme 2

Cores Active	Time-span for which pair is ON (s)
1 and 16	0 - 10
4 and 13	10 - 20
6 and 12	20 - 30
2 and 8	30 - 40
9 and 15	40 - 50
5 and 11	50 - 60
7 and 14	60 - 70
3 and 10	70 - 80
For Each Core Pair : T_{ON} = 10 s T_{OFF} = 70 s	
Maximum Temperature Attained : 40.16 ^{0}C	

Case 3: Four cores active at a time

In this case, a set of four cores were turned on and off in sequence for a fixed time interval, such that at any point in time, there were two cores functioning, generating heat. A set is turned on at the beginning of a ten second interval and then turned off at the end of ten seconds. Then, the next set is turned on and kept active for a period of ten seconds and so on. Once a full cycle of all 4 sets (16 cores) is completed, the cycle executes itself once more and continues till the end of the simulation period, which as mentioned earlier, was 500 seconds.

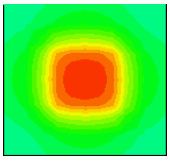
A. Scheme 1

The cores at the periphery, cores 1, 4, 16 and 13 comprise one set and are activated initially. The innermost cores - 6, 7, 10 and 11 comprise another set and are activated subsequently. The set comprising of cores 5, 8, 9 and 12 is then turned on and finally, cores 2, 3, 15 and 14 are activated.

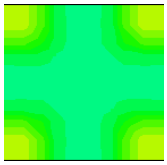
The active periods and the maximum temperature observed are presented in table 6.5.

Table 6.5 four cores activated: scheme 1

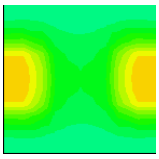
Cores Active	Time-span for which pair is ON (s)
1-4-16-13	0 - 10
6-7-10-11	10 - 20
5-8-12-9	20 - 30
2-3-15-14	30 - 40
For Each Core Set : T_{ON} = 10 s T_{OFF} = 30 s	
Maximum Temperature Attained : 55.07 ^{0}C	



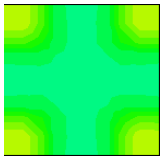
(a)



(b)



(c)



(d)

Figure 6.3 Simulation of Scheme 1(a) t = 124 s (b) t = 246 (c) t = 348 s (d) t = 482 s

B. Scheme 2

In this case, again, a set of four cores is active at any point in time, but the selection and order of activation of the various sets is random. One such grouping that has been considered to reduce global temperature and order of activation is given below in table 6.6.

Table 6.6 Four cores activated: scheme 2

Cores Active	Time-span for which pair is ON (s)
1 and 16	0 -10
4 and 13	10 -20
6 and 12	20 -30

Table 6.6 *Continued*

2 and 8	30 -40
9 and 15	40 -50
5 and 11	50 -60
7 and 14	60 -70
3 and 10	70 -80
For Each Core Set : T_{ON} = 10 s T_{OFF} = 30 s	
Maximum Temperature Attained : 40.16 ^{0}C	

Case 4: One core getting activated before another stops

As the title suggests, a case was also considered wherein what happens if while a core is processing an operation, another core needs to get powered on and how the sequence in which the cores get powered on may play a role in lowering on chip temperature.

Two such schemes were considered, which are mentioned as follows.

A. Scheme 1

This scheme entails a processor core in the immediate vicinity of an already active core starting up before the already active core has stopped functioning. The sequence of activation of cores and the temperature encountered thereof is mentioned in table 6.7.

Table 6.7 One core getting activated before another stops: Scheme 1

Core Active	Time-span for which core is ON (s)
1	0 – 10
5	5 – 15
2	10 - 20
6	15 - 25

Table 6.7 *Continued*

3	20 - 30
7	25 - 35
4	30 - 40
8	35 - 45
9	40 - 50
13	45 - 55
10	50 - 60
14	55 - 65
11	60 - 70
15	65 - 75
12	70 - 80
16	75 - 85
For Each Core : T_{ON} = 10 s T_{OFF} = 70 s	

B. Scheme 2

This scheme entails a processor core NOT in the immediate vicinity of an already active core starting up before the already active core has stopped functioning, to see the effect on temperature. The sequence of activation of cores and the temperature encountered thereof is mentioned in table 6.8.

Table 6.8 One core getting activated before another stops: Scheme 2

Core Active	Time-span for which core is ON (s)
1	0 - 10
4	5 - 15

Table 6.8 *Continued*

16	10 - 20
13	15 - 25
5	20 - 30
8	25 - 35
12	30 - 40
9	35 - 45
2	40 - 50
15	45 - 55
7	50 - 60
14	55 - 65
6	60 - 70
11	65 - 75
3	70 - 80
10	75 - 85
For Each Core : T_{ON} = 10 s T_{OFF} = 70 s	
Maximum Temperature Attained : 43 ^{0}C	

Case 5: Heterogeneous Cores

In addition to the above mentioned cases, the case of asymmetric many-core processors, wherein a few cores are larger, "primary" cores, consuming more power and running for a longer time than others which are smaller cores consuming less power, was also considered. As far as this particular model is concerned, two cores were made primary cores while the remaining were kept as smaller cores. The temperature obtained is given in table 6.9.

Table 6.9 Heterogeneous Cores

Core Active	Time-span for which core is ON (s)
1	0 – 20
14	20 – 40
For Each Core : T_{ON} = 20 s T_{OFF} = 20 s	
3	0 -10
9	10 -20
11	20 -30
5	30 -40
7	40 -50
13	50 -60
2	60 -70
8	70 -80
6	80 -90
4	90 -100
10	100 -110
12	110 -120
For Each Core : T_{ON} = 10 s T_{OFF} = 110 s	
Maximum Temperature Attained : 49.3^0C	

6.2.2 Temperature Based Hopping

The model generated in Icepak was imported into FLUENT 12.0.16. User Defined Functions were written to assign the boundary and switching conditions and were hooked to the respective zones in the model.

Tow test cases were examined, a dual core processor and a quad core processor. Only one core was kept active at any time.

Case 1: Dual Core

For the dual core case, initially core 1 is turned on and core 2 turned off. The temperature of core 1 is measured and if it crosses a certain set threshold, the power on core 1 is turned off and core 2 is turned on. Then, when core 1 goes below the temperature threshold, the operation is again switched back to core 1. A heat flux of $1e^6$ W/m^2 and an ambient temperature of 293 K were assumed.

Two temperature thresholds were considered - one at 305 K and the other at 310 K. For the case of the temperature threshold being set as 305 K, hopping of power was observed from core 1 to core 2 as the temperature of core 1 exceeded 305 K. However, when the threshold was set as 310 K, no hopping was observed which implied that core 1 never reached the thermal threshold in order for the hopping to take place, thereby validating the code. The choice of 310 K as a threshold value was guided by observations that the temperature reached steady state at around 306 K.

Case 2: Quad Core

In the case of quad core processor, again, initially core 1 is turned on and all the other cores are kept turned off. Core 2 is turned on only if core 1 exceeds the temperature threshold. Similarly, core 3 is turned on only if both cores 1 and 2 are above their prescribed temperature thresholds and likewise for core 4, which is turned on only if cores 1, 2 and 3 are all above their prescribed thresholds.

The temperature threshold is chosen as 305 K and the ambient temperature is considered to be 293 K. However, this time, two different values of heat flux are considered, to be assigned to an active core in order to check the validity of the code.

In the first case, a heat flux of $1e^6$ W/m^2 is considered. It is observed that, despite four cores being present, hopping occurs only between cores 1 and 2. This is because, due to the comparatively low value of heat flux, core 1 cools down sufficiently after some time of the activity switching to core 2, thereby causing the activity to hop back to core 1. This results in the activation conditions for cores 3 and 4 to never be satisfied, due to which they are never activated.

However, when a heat flux of $5e^6$ W/m^2 is considered, we see hopping occurring among all four cores as the conditions for activation for each core is satisfied at some point of time.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Core Hopping was studied as a technique to cool a multi-core processor. Both time based and temperature based core hopping were looked into. Transient analysis was performed to see the effect of switching of activity from core to core on the overall temperature of the chip. Along the way, different cases were considered such as varying number of cores active simultaneously, varying locations of the active cores and heterogeneous cores, for the time based case and variations in temperature threshold and heat flux for the temperature based case and the temperature was observed.

For the time based case, it was noted that there was no marked variation in temperature for the case of a single core running but a change was observed in the case of dual and quad core processors, which lend credence to the technique of core hopping. It is also noted that the choice of sequence of core activation also plays an important role in reducing chip temperature.

In the case of temperature based hopping, it is observed that it is possible to maintain a multi core chip within a certain temperature threshold by continuously monitoring temperature on the active core and switching the operation if the temperature threshold is close to being violated.

7.2 Future Scope of work

Further analysis of temperature based core hopping is proposed to examine the myriad possibilities offered by core hopping, such as increasing the number of concurrently active cores, thereby increasing performance, and observing the consequent effect on temperature. Also, a study on hybrid temperature and time based core hopping technique is of interest.

REFERENCES

[1]. http://www.calce.umd.edu/general/published/books/descriptions/materials.html

[2]. http://www2.renesas.com/pkg/en/pk_01.html

[3]. Fundamentals of Microsystems Packaging - Rao Tummala, McGraw Hill 2000

[4]. http://www.electronics-cooling.com/2003/02/optical-measurement-of-flip-chip-package-warpage-and-its-effect-on-thermal-interfaces/

[5]. "Exploring the Thermal Impact on Manycore Processor Performance", Wei Huang, Kevin Skadron, Sudhanva Gurumurthi, Robert.J. Ribando and Mircea R. Stan, 26th IEEE SEMI-THERM symposium, 2010.

[6]. "Hot Spots and Core-to-Core thermal Coupling in Future Multi-Core Architectures", M.Janicki, J.H. Collet, A.Louri and A.Napieralski, 26th IEEE SEMI-THERM Symposium, 2010.

[7]. "Cramming more components onto integrated circuits", Electronics Magazine, May 1965.

[8]. "Thousand core chips - A Technology perspective", S.Borkar, DAC 2007.

[9]. http://web.sfc.keio.ac.jp/~rdv/keio/sfc/teaching/architecture/architecture-2007/lec01.html

[10]. "A Fast Scheme to Investigate Thermal Aware Scheduling policy for Multicore Processors", Liqiang He, Cha Narisu, 2009.

[11]. http:// www.amd.com Ret. 03/20/2010.

[12]. http:// www.intel.com Ret. 03/20/2010.

[13]. http://ixbtlabs.com

[14]. "Design Challenges of Technology Scaling", S.Borkar, Intel Corp., 1999.

[15]. "Interaction of Scaling Trends in Processor Architecture and Cooling", Wei Huang, Mircea R. Stan, Sudhanva Gurumurthi, Robert.J. Ribando and Kevin Skadron, 26th IEEE SEMI-THERM Symposium, 2010.

[16]. http:// www.nvidia.com Ret. 03/21/2010

[17]. http:// ati.amd.com

[18]. "Heterogeneous Chip Multiprocessing", R.Kumar, D.M. Tullsen, N.P.Jouppi, and P.Ranganathan, IEEE Computer 38(11):32-38, November 2005.

[19]. "Core Fusion: Accommodating Software Diversity in Chip Multiprocessors", E.Ipek, M.Kirman, N.Kirman and J.F. Martinez, In Proc. Of ISCA, 2007.

[20]. "Federation : Repurposing scalar cores for out-of-order instruction issue.", D.Tarjan, M.Boyer, and K.Skadron, In proc. Of DAC, 2008.

[21]. "Managing the Impact of Increasing Microprocessor Power Consumption", Stephen H. Gunther, Frank Binns, Douglas M. Carmean, Jonathan. C.Hall, Intel Technology Journal, 2001.

[22]. "Many Core Design from a Thermal Perspective", W.Huang, M.Stan, K.Sankaranarayanan, R.Ribando, and K. Skadron, In Proc. Of DAC 2008.

[23]. "Proactive Power Migration to reduce maximum value and spatiotemporal Non-Uniformity of On-Chip temperature distribution in homogenous many-core processors", M.Cho, N.Sathe, M.Gupta, S.Yalamanchilli, and S. Mukhopadhyay, In Proc. Of SEMITHERM 2010.

[24]. "Techniques for Multicore Thermal Management: Classification and New Exploration", J.Donald, and M.Martonosi, In Proc. Of ISCA 2006.

[25]. "Dynamic Thermal Management for High Performance Microprocessors", D.Brooks, and M.Martonosi, In Proc Of 7th International Symposium on Performance Computer Architecture 2001.

[26]. "Understanding the Thermal Implication of Multicore Architectures", P.Chaparro, and J.Gonzalez, IEEE 2007.

[27]. "Thread Motion: Fine-Grained Power Management for Multi-Core Systems", Krishna.K.Rangan, Gu-Yeon Wei, D.Brooks, In Proc Of 36th ISCA,2009.

[28]. "Thermal-aware Task Scheduling at the System Software Level", Jeonghwan Choi, Chen-Yong Cher, Hubertus Franke, Hendrik Hamann, Alan Weger, and Pradip Bose, In Proc Of International Symposium on Low Power Electronics and Design, 2007.

[29]. http://vision.pcvsconsole.com/?article=21

[30]. http://news.cnet.com/2100-1001-954456.html

[31]. http://www.eetimes.com/news/semi/showArticle.jhtml?articleID=196901229

[32]. "From Single Core to Multi-Core: Preparing for a New Exponential", J.Parkhurst, J.Darringer, and B.Grudmann, In Proc Of International Conference on Computer Aided Design, 2006.

[33]. "Wattch: A Framework for Architectural-Level Power Analysis and Optimizations", D.Brooks, V.Tiwary and M.Martonosi, International Symposium on Computer Architecture, June 2000.

[34]. "Heat-and-Run: Leveraging SMT and CMP to Manage Power Density Through the Operating System", M.Gomaa, M.D.Powell, and T.N.Vijaykumar, ACM SIGOPS Operating Systems Review,2004.

[35]. "Temperature-aware micro Architecture", K.Skadron, M.R.Stan, W.Huang, S.Veluswamy, K.Sankaranarayanan, D.Tarjan, International symposium on computer architecture-June 2003.

[36]. "Reducing the Latency and Area Cost of Core Swapping Through Shared Helper Engines", A.Shayesteh, E.Kursun, T.Sherwood, S.Sair, and G.Reinman, In Proc Of International Conference on Computer Design, 2005.

[37]. "Reducing Power Density Through Activity Migration", S.Heo, K.Barr, and K.Asanovic, In Proc Of International Symposium on Low Power Electronics and Design, 2003.

[38]. "Development of Numerical Model for Non Uniformly Powered Die to Improve both Thermal and Device Clock Performance", S.Karajgikar, D.Agonafer, K.Ghose, B.Sammakia and G.Refai-Ahmed, IPACK 2009.

[39]. "Effect of Relocation of Functional Units of a Nonuniformly powered microprocessor on thermal and device clock performance", S.Karajgikar, D.Agonafer, K.Ghose, B.Sammakia, C.Amon and G.Refai-Ahmed, IMECE 2009.

BIOGRAPHICAL INFORMATION

The author obtained his Bachelor's Degree in Electronics and Communication Engineering from Anna University, Chennai, India and commenced his Master's Degree Program in Aerospace Engineering at The University of Texas at Arlington in Fall 2007. He joined the EMNSPC team under Dr. Agonafer in Fall 2009. He has worked in the areas of Thermoelectric Coolers and cooling of Quantum Cascade Lasers, apart from focusing primarily on his research area, cooling of Multi Core Processors.