Kameswar Rao Vaddina

Thermal-Aware Networked Many-Core Systems

Turku Centre *for* Computer Science

TUCS Dissertations
No 175, May 2014

Thermal-Aware Networked Many-Core Systems

Kameswar Rao Vaddina

To be presented, with the permission of the Faculty of Mathematics and Natural Sciences of the University of Turku, for public criticism in Auditorium Beta on May 23rd, 2014, at 12 noon.

University of Turku
Department of Information Technology
20014 Turku, Finland

2014

Supervisors

Assoc. Prof Juha Plosila
Department of Information Technology
University of Turku
20014 Turku, Finland

Assoc. Prof Pasi Liljeberg
Department of Information Technology
University of Turku
20014 Turku, Finland

Reviewers

Prof. Gert Jervan
Department of Computer Engineering
Faculty of Information Technology
Tallinn University of Technology
Akadeemia tee 15a, 12618 Tallinn, Estonia

Assoc. Prof. Baris Taskin
Department of Electrical and Computer Engineering
Drexel University
3141 Chestnust Street
Philadelphia, PA 19104-2875, USA

Opponent

Prof. Peeter Ellervee
Department of Computer Engineering
Faculty of Information Technology
Tallinn University of Technology
Akadeemia tee 15a, 12618 Tallinn, Estonia

ISBN 978-952-12-3063-9
ISSN 1239-1883

The originality of this thesis has been checked in accordance with the University of Turku quality assurance system using the Turnitin Originality Check service.

Abstract

Advancements in IC processing technology has led to the innovation and growth happening in the consumer electronics sector and the evolution of the IT infrastructure supporting this exponential growth. One of the most difficult obstacles to this growth is the removal of large amount of heat generated by the processing and communicating nodes on the system. The scaling down of technology and the increase in power density is posing a direct and consequential effect on the rise in temperature. This has resulted in the increase in cooling budgets, and affects both the life-time reliability and performance of the system. Hence, reducing on-chip temperatures has become a major design concern for modern microprocessors.

This dissertation addresses the thermal challenges at different levels for both 2D planer and 3D stacked systems. It proposes a self-timed thermal monitoring strategy based on the liberal use of on-chip thermal sensors. This makes use of noise variation tolerant and leakage current based thermal sensing for monitoring purposes. In order to study thermal management issues from early design stages, accurate thermal modeling and analysis at design time is essential. In this regard, spatial temperature profile of the global *Cu* nanowire for on-chip interconnects has been analyzed. It presents a 3D thermal model of a multicore system in order to investigate the effects of hotspots and the placement of silicon die layers, on the thermal performance of a modern flip-chip package. For a 3D stacked system, the primary design goal is to maximise the performance within the given power and thermal envelopes. Hence, a thermally efficient routing strategy for 3D NoC-Bus hybrid architectures has been proposed to mitigate on-chip temperatures by herding most of the switching activity to the die which is closer to heat sink. Finally, an exploration of various thermal-aware placement approaches for both the 2D and 3D stacked systems has been presented. Various thermal models have been developed and thermal control metrics have been extracted. An efficient thermal-aware application mapping algorithm for a 2D NoC has been presented. It has been shown that the proposed mapping algorithm reduces the effective area reeling under high temperatures when compared to the state of the art.

Tiivistelmä

Integroitujen piirien valmistusteknologian edistys on johtanut kulutuselektroniikan innovaatioihin ja alan kasvuun sekä tätä eksponentiaalista kasvua tukevan IT-infrastruktuurin kehittymiseen. Yksi vaikeimmista kasvua haittaavista tekijöistä on elektroniikkajärjestelmän suorittaman laskennan ja kommunikaation tuottama lämpö ja sen poisto järjestelmästä. Piiriteknologian kehitys kohti pienempiä viivanleveyksiä ja tehotiheyden kasvu aiheuttavat lämpötilan nousua järjestelmissä. Tämä johtaa haastaviin jäähdytysteknisiin ratkaisuihin ja vaikuttaa sekä järjestelmän luotettavuuteen että suorituskykyyn. Lämpötilan alentamisesta on täten tullut tärkeä tekijä nykyaikaisten mikroprosessorisirujen suunnittelussa.

Tässä väitöskirjassa tarkastellaan planaaristen 2D-järjestelmien ja pinottujen 3D-järjestelmien lämmöntuoton haasteita eri tasoilla. Työssä esitetään itseajoittuva lämpötilan monitorointistrategia perustuen sirunsisäisten lämpötila-anturien vapaaseen käyttöön. Tämä strategia soveltaa kohinasietoista ja vuotovirtaperusteista lämpötilan ilmaisutekniikkaa monitorointitarkoituksiin. Tarkka terminen mallinnus ja suunnittelunaikainen analyysi ovat keskeisessä asemassa, kun pyritään tutkimaan lämmöntuoton hallintaan liittyviä kysymyksiä järjestelmän suunnitteluprosessin varhaisissa vaiheissa. Tähän liittyen työssä analysoidaan kuparipohjaisten sirunsisäisten johtimien lämpötilaprofiilia. Profiili esittää moniydinprosessorin kolmiuloitteisen lämpömallin, jonka avulla voidaan tutkia ns. kuumien pisteiden ja piisirukerrosten sijoittelun vaikutusta modernin "flip-chip"-tyyppisen monisirukotelon termiseen suorituskykyyn. Pinottuja 3D-piirejä suunniteltaessa ensisijainen tavoite on maksimoida suorituskyky siten, että tehonkulutus ja lämpötila pysyvät annettujen rajojen sisällä. Tätä silmälläpitäen väitöstyössä esitetään lämpötilaherkkä reititysalgoritmi verkkopiiri- ja väylärakenteet yhdistävälle 3D-hybridiarkkitehtuurille. Perusajatuksena on ohjata suurin osa tiedonsiirtoon liittyvästä piiriaktiviteetista sirulle, joka on lähimpänä jäähdytyselementtiä. Lopuksi väitöskirjassa tarkastellaan erilaisia lämpötilatietoisia sijoittelumenetelmiä 2D- ja 3D-järjestelmille. Tähän liittyen kehitetään useita lämpömalleja ja johdetaan lämpötilakontrollin mittareita. 2D-verkkopiirirakenteelle esitetään tehokas lämpötilatietoinen sovellusten hajautusalgoritmi. Työssä

osoitetaan, että verrattuna aikaisemmin esitettyihin ratkaisuihin tämä algoritmi pienentää korkeassa lämpötilassa olevan pinta-alan osuutta verkkopiirisirulla.

This thesis is dedicated to my parents Vaddina Narasimha Murthy and Vaddina Kalavathi.

Acknowledgements

"*Begin at the beginning,*" the King said, very gravely, "*and go on till you come to the end: then stop.*" - Lewis Carroll, Alice in Wonderland.

It was not an easy journey for me to embark on my doctoral adventure. Sometimes, it was riddled with despair and anguish, and sometimes with hope and delight. But overall, it was 'fun' and I enjoyed the ride. A lot of people supported, inspired, encouraged and influenced my research career in Finland. Many of them shaped my understanding, curated my thought process and helped me to evolve into a better human being. Thanks to them I am now here and waiting for life's next great adventure. As Alice would say, "*I can't go back to yesterday because I was a different person then*". But now, today, it is time for me to thank all the people who have helped me achieve this.

First and foremost, I would like to express my sincere gratitude to my supervisor and research director Adj. Prof. Juha Plosila for his guidance and encouragement during the PhD program. Without his high-level guidance and inspiration, this thesis would not have been possible. His great sense of humor has always kept me at ease in many technical discussions we had over the years. I would also like to thank Adj. Prof. Pasi Liljeberg for co-supervising my thesis and helping me to improve the quality of my research and the thesis text. Both of you have given me the liberty in choosing the topic and were flexible enough when I was moulding it to suit my research interests. Moreover, you have trusted me with the research and financially supported me throughout the years for which I am immensely grateful. This thesis work has been financially supported by the Academy of Finland, the Nokia Foundation, the Ulla Tuominen Foundation, Turku University Foundation and all the Finnish tax payers. I am infinitely indebted to all the financial supporters.

I would like to thank Prof. Gert Jervan from Department of Computer Engineering, Tallinn University of Technology and Assoc. Prof. Baris Taskin from Department of Electrical and Computer Engineering, Drexel University for their detailed reviews and constructive comments of this work. Their suggestions have greatly improved the quality and relevance of this work.

I addition, I would like to sincerely thank my fellow colleagues in the lab who have helped me stay sane during the years. Especially, Dr. Ethiopia Nigussie, Dr. Khalid Latif and Dr. Liang Guang. Incidentally, all of whom also happen to be my co-authors and have reviewed my thesis. I really cherish the time we spent together in the coffee room discussing everything under the sun. From research topic in question to various aspects of living in Finland to immigration issues, you were there all the time helping and guiding me along the process. I had some of the best time in Turku with you all and value your friendship forever.

I have had the great pleasure of working, authoring and co-authoring publications with some of the best minds in the lab. Our common passion for research has made us an ideal team to work together. Especially, I would like to thank Dr. Amir Mohammad Rahmani and Mohammad Fattah from UTU and Tamoghna Mitra (from Åbo Akademi University) for sharing their insights and understanding on varied subjects. Others from the lab with whom I have had pleasure to interact with on a daily basis, but was not fortunate enough to work together are Thomas Canhao Xu, Rajeev Kumar Kanth, Bo Yang, Masoud Daneshtalab, and Masoumeh Ebrahimi. Thank you all for making my stay in the lab a comfortable one and I learnt a great deal from each and every one of you. I would also like to thank immensely my tutor Dr. Teijo Lehtonen for helping me settle down in Turku during my initial days. Special mention and thanks should also go to Sami Nuuttila for answering many of my silly questions on Linux and for his immense help in setting up various tools and benchmarks. I would also like to thank TUCS secretary, Irmeli Laine and former graduate programme coordinator for TUCS, Satu Jääskeläinen for going out of their way and helping me to come to Finland so that I could pursue my passion.

All work and no play makes Jack a dull boy. The Indian community in Turku made sure that there is no dearth of social activities and gatherings. They were always there through think and thin and have showered me with their love, hospitality and amazing food. Shishir, when I said amazing food, I did not count you in. Thanks to the Indian community, Turku always felt like home for me.

I was very fortunate to be in the great company of friends who have varied interests, broad and liberal values, and yet have similar outlook towards life like me. I have enjoyed the long and endless conversations with them and remember that many times we had to "agree to disagree" just so that we can go home. I would like to express my sincere gratitude to Shishir Jaikishan, Anil Kumar Alla, Senthil Palani, Hariharan Dandapani and Pasi Kankaanpää. Shishir, Senthil and Hari are avid Cricket followers whose conversations I never followed. They all are great both on and off the field. Anil and me are co-founders of a startup which failed quickly. But then again, it was so much fun to be around you all and I enjoyed each

and every moment sharing and laughing with you. Special thanks to Pasi for all the great sailing trips (the best one is yet to come though) and for being a window into Finnish culture and society. Pasi, I have always told you otherwise, but I love to read those long e-mails you send. Sometimes!! Also, special thanks goes to my trekking buddies Bineet Panda, Anil Kanduri, Kartiek Kanduri, Tamoghna Mitra and Debanga Nandan. Thanks a lot Ponnuswamy Mohanasundaram for accompanying me for lunch everyday during the past year and a half and listening and sharing. Frankly guys, you all are irreplaceable.

I have to forcefully stop myself from extending this acknowledgment section lest it becomes a thesis in itself. But before I do that I have to thank the most important people of my life without whom I would not have been here. My parents have made several sacrifices in their adult life just so that their kids can have the best possible education. Despite facing several odds, they both believed in the power of knowledge and enlightenment which is bestowed by education, and education alone. I really hope that they have realized that they have succeeded in planting the seed of knowledge in me and have made me into a learner for life. I dedicate this thesis to both my parents who made this possible. I also greatly acknowledge the support of both my brother and sister during this entire process. I am truly thankful to them.

Finally, I would like to thank my wife Manjusha Kasu. Her constant support, encouragement and unwavering love is what made this thesis possible. Her smile alone used to light my days up. I would like to express my heartfelt thanks to her for being with me during the past 5 years which were the best years of my life.

Turku, May 2014
Kameswar Rao Vaddina.

Contents

1 Introduction **1**
- 1.1 Background and Motivation . . . 2
- 1.2 Temperature Issues with 3D Stacked Systems . . . 5
- 1.3 Thermal Control Optimization Strategies . . . 6
- 1.4 Thesis Objectives . . . 8
- 1.5 Thesis Contributions . . . 9
- 1.6 Research Publications . . . 10
- 1.7 Organization of Thesis . . . 12

2 Thermal Management Techniques for Microprocessors **15**
- 2.1 Power Management vs Thermal Management . . . 17
- 2.2 Classification of Temperature Control Mechanisms . . . 19
  - 2.2.1 Off-Chip Thermal Management Techniques . . . 20
  - 2.2.2 Design-Time Thermal Management Techniques . . . 20
  - 2.2.3 Dynamic Thermal Management Techniques . . . 21
- 2.3 Summary . . . 28

3 Self-Timed Thermal Sensing and Monitoring of Multicore Systems **29**
- 3.1 Introduction and Motivation . . . 29
- 3.2 Thermal sensing architecture . . . 31
- 3.3 Sensing interconnection network . . . 36
- 3.4 Noise and supply voltage variation analysis . . . 40
  - 3.4.1 Power supply noise (PSN) analysis . . . 41
  - 3.4.2 Input signal noise (ISN) analysis . . . 42
  - 3.4.3 Supply voltage variations . . . 42
- 3.5 Summary . . . 43

4 Thermal Modeling and Analysis **45**
- 4.1 Thermal analysis of on-chip interconnects in multicore systems 46
  - 4.1.1 Resistivity vs Temperature . . . 48
  - 4.1.2 Thermal Analysis of Links . . . 48
  - 4.1.3 Signal transmission methods . . . 50

4.1.4 Wide line vs narrow line 52
4.1.5 Summary 52
4.2 Thermal modeling and analysis of 3D stacked systems 54
4.2.1 Nomenclature 55
4.2.2 Flip-Chip package 55
4.2.3 Thermal modelling and Analysis 56
4.2.4 Simulation results 59
4.2.5 Summary 65

5 Thermally Efficient Inter-Layer Communication Scheme 67
5.1 Introduction to Hybrid NoC bus 3D architecture 67
5.2 Thermally efficient routing strategy for 3D NoC 68
5.3 Thermal model to evaluate the thermally efficient routing strategy for a 3D NoC 70
5.4 Simulation results and analysis 73
5.4.1 Thermally efficient routing for 3D NoC 73
5.5 Summary 75

6 Thermal-Aware Mapping 77
6.1 Thermal-Aware Placement in 2D and 3D Chip Systems . . . 77
6.1.1 Uniform power distribution 78
6.1.2 Thermal-aware placement for a 2D chip system 78
6.1.3 Thermal-aware placement for a 3D stacked chip systems 79
6.2 Thermal modeling and simplifications 82
6.2.1 Thermal modeling using Hotspot 82
6.2.2 Thermal modeling using COMSOL 82
6.3 Thermal analysis 84
6.3.1 Uniform power distribution case 84
6.3.2 Thermal-aware placement for a 2D chip system 84
6.3.3 Thermal-aware placement for a 3D stacked chip systems 86
6.4 Proposed temperature mitigation techniques 96
6.4.1 Thermal-aware mapping for 2D NoC 96
6.4.2 Thermal Modelling 100
6.5 Simulation results and analysis 102
6.5.1 Thermal-aware mapping for 2D NoC 102
6.6 Summary 103

7 Conclusions and Future Work 105
7.1 Summary and Conclusions 105
7.2 Future Work 108

List of Figures

1.1 Power consumption of a die as a function of temperature. It is a 15-mm Intel fabricated die in a 0.1μm technology and a supply voltage of 0.7V [1] 2
1.2 The vicious circle of power, temperature and leakage cycle . . 3
1.3 Increase in leakage power with technology scaling (IBS Electronics [2]) 4
1.4 Futuristic virtualization platform. 6

2.1 Future trends for static and dynamic power for both the logic and memory [3] 17
2.2 Future trends for switching and leakage power for both the logic and memory [3] 18
2.3 Classification of temperature control mechanisms 20
2.4 Classification of dynamic thermal management techniques and control algorithms 22
2.5 Classification of dynamic thermal management techniques in multicore architectures 23
2.6 A simple thermal equivalent circuit for a 3D stacked system in a flip-chip package 26

3.1 Thermal Sensing Circuit (TSC). 31
3.2 Self-timed handshaking protocol for the thermal sensing architecture. 32
3.3 Pulse Filter. 32
3.4 Timing diagram of the Thermal Sensing Circuit (TSC) at 27°C. 33
3.5 Timing diagram of the Thermal Sensing Circuit (TSC) at 60°C. 33
3.6 Leakage current based thermal sensor [4]. 34
3.7 Response of the sensor in the 27°C to 100°C range as simulated in 65nm technology. 35
3.8 The delay through the sensor, plotted against the temperature. 35
3.9 MUTEX and its timing diagram. 37
3.10 Self-timed signaling architecture for sensing interconnection network. 38

3.11 Propagation delay of the delay element vs temperature. . . . 39
3.12 No.of clock cycles vs temperature. 39
3.13 Encoding of temperature. 40
3.14 Output pulse-width of the thermal sensor with noisy power supply rails and input signal noise vs the one with the ideal power supply for different temperature values. 41

4.1 Conductor of length L. 49
4.2 Spatial temperature profile along the Cu nanowires with 400μm via separation. The dimensions and other material properties of the global interconnect used are for 65nm technology node from ST microelectronics [5]. 51
4.3 Temperature distribution along the total length of the conductor optimally divided into different segments and interspersed with vias. 51
4.4 Interleaving of repeaters. 52
4.5 Spatial temperature profile along the Cu nanowires with $2mm$ via separation. The dimensions and other material properties of the global interconnect used are for 65nm technology node from ST microelectronics [5]. 53
4.6 Connection between the interconnect segments with a group of vias. 53
4.7 Cross-Sectional view of a modern 3D Flip-Chip package with 2 stacked dies. 56
4.8 Thermal resistance measurements for both the dies in model-I at steady-state. 61
4.9 Thermal resistance measurements for both the dies in model-II at steady-state. 61
4.10 Maximum temperature on the processing and memory die for both models. 62
4.11 10% Maximum temperature on the processing and memory die for both models. 63
4.12 10% Thermal Resistance on the processing and memory die for both models. 63
4.13 Improvement required in heat sink thermal resistance for a 3D system (both models) whose memory layer is consuming 50% of the processing die power. It has been compared with a single die package system. 64

5.1 The proposed thermally efficient routing algorithm 70
5.2 Communication trace of encoder part of a H.264 video conference 74

5.3 Partition and core mapping of the video conference encoding application. 75
5.4 Steady-state grid level thermal maps for the die 1(layer 0) for both the normal routing and (a) and our thermal-aware hybrid routing (b). 76

6.1 Four different hotspot placement cases that were analyzed for a 2D chip system. 79
6.2 Cross-Sectional view of a modern 3D Flip-Chip package with 3 stacked dies. 79
6.3 Side view of the thermal model using COMSOL. 83
6.4 Front view of the thermal model using COMSOL. 83
6.5 Uniform power distribution does not lead to uniform temperature distribution on the silicon die in a Flip-Chip package. . 85
6.6 Thermal profiles of a) *FP_CENTER*, b) *FP_CORNER*, c) *FP_SIDE* and d) *FP_MIDDLE* cases of a 2D chip system. . . 86
6.7 Thermal profiles of all the 3-layers of a 3D stacked system when the worst case hotspot scenario occurs in a die which is i) BOTTOM:farther from the heat sink (a,b,c), ii) MIDDLE:equidistant from the heat sink and the heat spreader (d,e,f) iii) TOP:closer to the heat sink (g,h,i). 88
6.8 Coarse grained meshing of the thermal model. 89
6.9 Slice plot of the thermal model in the *Static* case. P = 200W, P_{die1}=P_{die2}=P_{die3}= 66.66W. 90
6.10 Subdomain plot of the thermal model in the *Static* case. P = 200W, P_{die1}=P_{die2}=P_{die3}= 66.66W. 90
6.11 Peak temperatures on all the three dies in the *Static* case. P = 200W, P_{die1}=P_{die2}=P_{die3}= 66.66W. 91
6.12 Peak temperatures on all the three dies in the *Adaptive* case. P = 200W, P_{die1} = 40W, P_{die2} = 60W, P_{die3} = 100W. 91
6.13 Peak temperatures on all the three dies in the *Adaptive_hotspot* case. P = 200W, P_{die1} = 40W, P_{die2} = 60W, P_{die3} = 100W, $P_{d_hotspot}$ = 100W/cm^2. 92
6.14 Interaction of two hotspots located on the same die (DIE-3). The plot is obtained by fixing the location of one hotspot at the center of the die and varying the location of the other. The distance 'd' in the plot is the distance between the centers of two hotspots. 95
6.15 Interaction of hotspots located in different vertically stacked layers. Each hotspot is located at the center of its die edge respectively. 95

6.16 Interaction of hotspots located in different vertically stacked layers, but distributed efficiently so that their thermal fields do not interact with each other. 96
6.17 An example task graph of an application consisting of 7 tasks. Hotspot tasks are depicted as concentric circles. 97
6.18 A 6×6 NoC depicting the blocks and dimensions of each tile. The dimensions are adopted from Intel's 65nm based 80-core processor [6]. 101
6.19 Latency vs throughput. 103
6.20 Thermal maps of a) Worst case b) TMB mapping case and c) Thermal-aware case 104

List of Tables

3.1 |PSN-ideal| and |ISN-ideal| values at different temperatures. Where PSN and ISN stands for power supply noise and input signal noise respectively. 42
3.2 Pulse width of the thermal sensor with supply voltage variations at 27°C and 100°C as compared to the one with no voltage variations. 43

4.1 Modelling parameters [7] [8] [9] [10]. 57

5.1 Power Consumption and Average Packet Latency 75
5.2 Layer temperature profile of the Hybrid Bus-NoC 3D Mesh-based system running the video conference application 75
5.3 Layer temperature profile of the proposed Hybrid Bus-NoC 3D Mesh-based system running the video conference application (thermal-aware hybrid routing) 76

6.1 Modelling parameters [7] [8] [9] [10]. 81
6.2 Maximum/peak, average and minimum temperatures in all the four placement cases of a 2D chip that is consuming a total power of 100W. 87
6.3 3D stacked system: Maximum/peak, average and minimum temperatures in all the three placement cases for a chip system that is consuming a total power of 300W. 87
6.4 Simulation run 1: Peak temperatures on all the three dies for all the three cases in a 200W system. 89
6.5 Simulation run 2: Peak temperatures on all the three dies for all the three cases in a 100W system. 92
6.6 Simulation run 3: Peak temperatures on all the three dies for all the three cases in a 600W system. 93
6.7 Table depicting the amount of chip area under a particular temperature. 103

List of Abbreviations

ACK	Acknowledgment
ADC	Analog-to-Digital Converter
ADI	Alternating Direction Implicit
APL	Average Packet Latency
AWMD	Average Weighted Manhattan Distance
BJTs	Bipolar Junction Transistors
CBGA	Ceramic Ball Grid Array
CMOS	Complementary Metal Oxide Semiconductor
CoC	Cloud on a Chip
Cu	Copper
DSD	Digital System Design
dTDMA	dynamic Time-Division Multiple Access
DTM	Dynamic Thermal Management
DVFS	Dynamic Voltage And Frequency Scaling
FCBGA	Flip-Chip Ball Grid Array
HAL	Hardware Abstraction Layer
HCI	Hot Carrier Injection
IC	Integrated Circuit
ILD	Interlayer Dielectric
ILM	Interlayer Material

ISN	Input Signal Noise
MPC	Model Predictive Control
MPSoC	MultiProcessor System-on-Chip
MTTF	Mean Time to Failure
NBTI	Negative Bias Temperature Instability
NoC	Network-on-Chip
PCM	Phase-Change Material
PE	Processing Element
PSSN	Power Supply Noise
PTAT	Proportional To Absolute Temperature
SoC	System-on-Chip
TCU	Thermal Control Unit
3D	Three-Dimensional
TIM	Thermal Interface Material
TMB	Tree-Model-Based
TSC	Thermal Sensing Circuit
TSV	Through-Silicon-Via
VMM	Virtual Machine Monitor
WLP	Wafer Level Packaging

Chapter 1

Introduction

In Greek mythology the Titan Prometheus is credited with stealing the heavenly fire from the gods and giving it to the humans thus enabling the process of progress and civilization for the entire humanity. This generous act brought the wrath of Zeus, king of the Olympian gods, who then sentenced Prometheus to eternal torment for his transgression before being freed by the hero Hercules. Great stories like these which are common to all of humanity follow the journey of a hero and usually culminate with the accomplishment of greater good for the entire mankind. For several millennia, humans made fire to generate heat and light which enabled them to cook food, stay warm and to keep nocturnal predators at bay. Despite sophisticated advances throughout human history in controlling and managing fire and heat, several challenges still remain. Especially, in the design of high performance electronic systems, the problem of heat is predominant and steps need to be taken in controlling and managing that heat. This thesis is a very small step in that direction.

Microprocessor chips are the building blocks of todays information world. Their performance has grown by over 1000-fold during the past 20 years which has been driven by both speed and energy scaling, as well as the advances in the microarchitecture design [11]. As was stated by Moore's law [12], the transistor density gains that were obtained by continuous scaling aided in the increased usage of microprocessor chips, from battery powered devices to data centers. At the same time, software applications are becoming more complex with every iteration and have a large impact on power and thermal maps of the system. Most of the energy consumed by the microprocessor is dissipated as heat, which could result in numerous undesirable effects, like performance degradation, reliability deterioration, high-energy costs, and physical damage leading to system failures. Hence, managing temperature at the chip, server and data center level has become one of the big concerns for the technology industry.

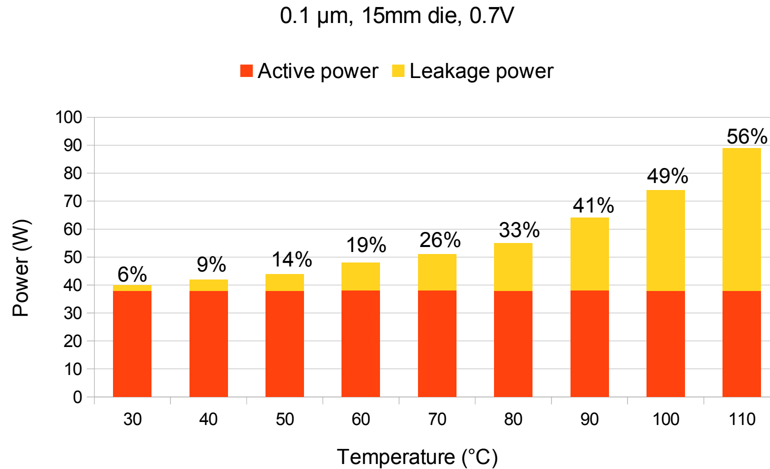

Figure 1.1: Power consumption of a die as a function of temperature. It is a 15-mm Intel fabricated die in a 0.1μm technology and a supply voltage of 0.7V [1]

1.1 Background and Motivation

As the technology scales down and power density increases, a lot of factors such as power dissipation, leakage, data activity, negative bias temperature instability (NBTI), hot carrier injection (HCI), and electro-migration contribute to higher temperatures, larger temperature cycles and increased thermal gradients all of which impact multiple failure mechanisms [7]. New improvements in process technology like the usage of low-k dielectrics and deep-trench isolation are also unfavorable for heat conduction [13]. The increase in temperature leads to increase in leakage power which in turn exacerbates an already serious thermal problem, thereby causing thermal runaway [14]. Fig. 1.1 depicts the significant increase in leakage power as a function of substrate temperature for a 15-mm die fabricated by Intel, at 0.1μm technology and a supply voltage of 0.7V [1].

This exponential dependance on temperature results in thermal runaway, which is sort of a vicious circle as shown in Fig. 1.2. This leads to significant drop in the performance of the system. Fallah et al. state that the leakage of transistors consume more than 40% of the total power consumption in 90nm process technology [15]. For still smaller nodes of technology the leakage power dominates the dynamic power consumption as shown in Fig. 1.3.

Also, the increase in temperature, increases interconnect delay due to the linear increase in electrical resistivity. These delay variations pose significant reliability problems with already dense interconnect structures. Joule self-heating, which is defined as the amount of heat generated when a maximum current of j_{max} passes through an interconnect wire, and delay variations

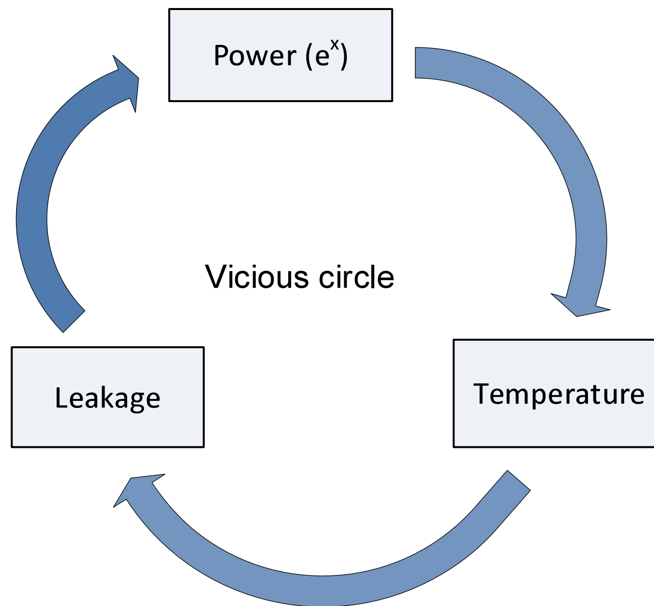

Figure 1.2: The vicious circle of power, temperature and leakage cycle

combined with the introduction of low-k dielectrics with low thermal conductivity increases the need for accurate thermal analysis and estimation of interconnect temperature. At the same time, it is not enough to just address thermal hot spots that might arise on the chip, as temperature gradients in both time and space determine the reliability of the system at moderate temperatures [16]. Instantaneous high temperature rises in the devices can cause catastrophic failure, as well as long-term degradation in the chip and package materials, both of which may eventually lead to system failure [17].

The ITRS report [7] projects that the power density for 14nm technology node will be greater than 100W/cm^2 and the junction-to-ambient thermal resistance will be less than 0.2°C. It is very important to keep the thermal resistance at bay as this may increase the package cost and thereby the overall cost of the product. Observation of the thermal contours of certain industrial chip shows that the temperature at the hotspots can really exceed 100°C [18]. Recent statistical analysis on component failures show that more than 50% of all integrated circuit failures are related to high temperatures. It means, that the aging process of the components increases with sustained high temperatures, thus leading to their failures [1].

If the heat cannot be transferred to the ambient at a rate equal to or greater than its generation, then junction temperatures will rise on the silicon die. As the junction temperatures increases, there will be a reduction in the mean time to failure (MTTF) [1]. Ambient temperatures vary a lot.

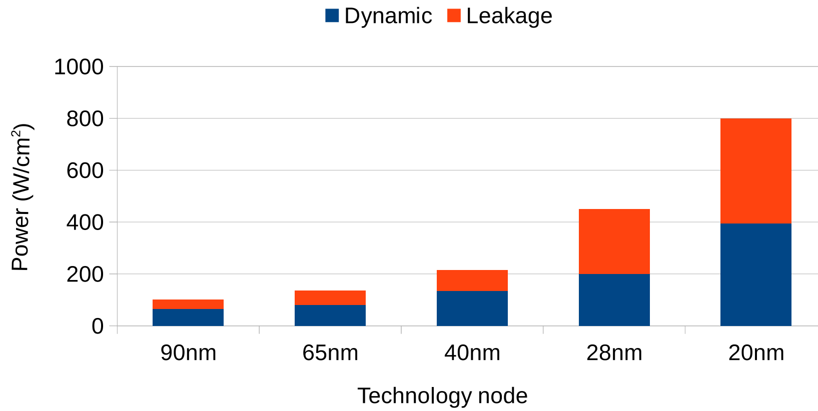

Figure 1.3: Increase in leakage power with technology scaling (IBS Electronics [2])

During the year 2013 in Turku, Finland, the temperature varied from -26°C to +30°C [19]. Also, depending on the industry segment under consideration, the ambient temperatures can be very harsh. For example in the automotive industry the temperature near the wheel ABS system can be greater than 150°C and the temperature near the exhaust system can be greater than 400°C. Similarly the temperature varies from engine oil, alternator to the interior of a car [20]. Hence it is important to consider detailed package models while performing thermal simulations.

The prevailing temperature crisis is a multi-scale problem. It can be seen at the chip/component level, server/board level, rack level and at the room level [21]. The temperature problem is worsened when the level of abstraction is scaled up from processor core to server, and from there to a data center, requiring equally expensive cooling solutions and increase in greenhouse carbon emissions.

The conventional strategy to increase performance of a microprocessor by increasing the frequency and taking advantage of innovations in process technology has hit the power wall [22]. To overcome the power wall, the semiconductor industry has started using multiple cores and parallelize execution to achieve the performance targets. With the increase in the number of cores on the chip we have now hit the thermal wall [21]. The total chip power is exceeding the thermal design power which is essentially the power that is dissipated by the chip through the package and to the ambient. This forces the designers and manufacturers of systems to only power part of the system resources at any point of time. It essentially means that most of the system is reeling under dark silicon. Dark silicon is a big problem for mobile

platforms, as there is very less scope to use complex and expensive cooling solutions in the light of the imposed functional and regulatory requirements by governmental agencies.

Futuristic virtualization platform

With the advent of cloud computing the systems of the future will become very complex with possibly thousands of cores running in parallel on a single silicon die. All of those cores could be tightly packed to form a data center on a chip which works on Cloud on a Chip (CoC) [23] paradigm. Virtualization platforms like the one shown in Fig. 1.4 can be an ideal solution for cloud computing. The hardware abstraction layer (HAL) is a small piece of software which interacts with the naked hardware and runs on top of it. Intel calls this hardware abstraction layer as Hypervisor, Microsoft calls it as Hyper-V and other vendors call it as Virtual Machine Monitor (VMM). There will be multiple operating systems running on the hardware simultaneously. Multiple users will be logged into those operating systems running multiple applications. The hardware abstraction layer provides access to the hardware resources and make them visible to the guest operating systems. The guest operating systems may not need to know the existence of other operating systems running in parallel. This increases the system robustness and stability. The time to deploy and debug new operating systems and applications without jeopardizing existing ones is a feature inherent to this technology. Such futuristic virtualization platforms would suffer from immense thermal challenges and needs dynamic thermal management techniques to be deployed.

1.2 Temperature Issues with 3D Stacked Systems

The processes required for stacking active device layers while preserving the intrinsic electrical characteristics of on-chip devices has been demonstrated by the industry [24]. On the other hand, several 3D chip design strategies that exploits the vertical dimension thereby facilitating heterogeneous integration technologies has also been demonstrated by the academia [25]. Therefore, three-dimensional (3D) integrated circuits have been proposed which would overcome the problems associated with the interconnects and the limits that are being posed by the traditional CMOS scaling [26]. At the same time, there exists several challenges in designing 3D integrated circuits. Some of them include the problem of placement, floorplanning and insertion of thermal-vias. Several techniques have been proposed to solve these issues [27] [28].

Thermal problems are also exacerbated with the transition from 2D chip systems to 3D stacked system [29]. 3D integrated circuits take advantage of

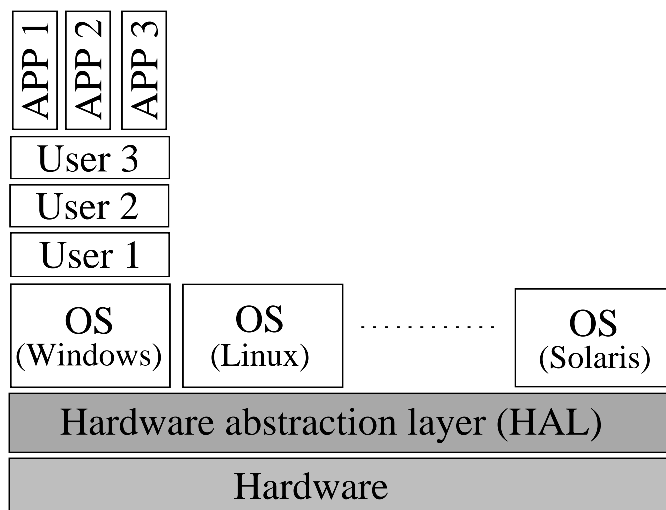

Figure 1.4: Futuristic virtualization platform.

dimensional scaling approach and are seen as a natural progression towards future large and complex systems. They increase device density, bandwidth and speed. On the other hand, due to increased integration, the amount of heat per unit footprint increases, resulting in higher on-chip temperatures and thereby degrading the performance and reliability of the system. In this case, heat sinks need to be very efficient in transferring the internally generated heat to the ambient. Most modern flip-chip devices [30] are designed to operate reliably with a junction temperature falling under a certain range. To ensure that the package can perform thermally well under this range a thermal model is simulated and tested. This thermal model can then be used to gauge the reliability of the package. This shortens the package development time and also provides an important analytical tool to evaluate its performance under different operating conditions.

1.3 Thermal Control Optimization Strategies

As the increase in temperature is directly related to power consumption of the chip, most of the power reduction techniques can be touted as modes to reduce temperature as well. But, in reality temperature reduction is a more complex problem. It is because, power is mainly an instantaneous effect, whereas increase in temperature is a long term process which is spread in both space and time. Temperature also depends on the physical properties of the chip layout and are not captured very well by just doing power reduction analysis (for example, metallization and thermal propagation) [31].

An excellent way to address temperature issues during design time is to ensure that circuit blocks are placed in such a way that they even out the

thermal profile: in other words, using temperature-aware placement. Simplistically speaking, if we spread high-power cells across the chip evenly, the temperature profile will be flat and we can avoid hot-spot related thermal issues. In reality though, thermal placement is a more complex problem, and a uniform distribution of power sources does not lead to uniform temperature [31].

It has been shown that for large multi-core 3D systems minimizing power is only part of the thermal optimization goal without sacrificing performance. The use of active liquid cooling is needed in future 3D servers and other temperature reduction techniques need to be explored for efficient thermal management [32]. It is clear that the cooling technology that gives the best results without sacrificing performance are always active (external cooling) options, but they cost power which can be quite high if they are not combined with other thermal optimizations [33] [32].

It should be noted that it is not completely possible to make a full categorization of thermal-control benefits for different thermal optimization techniques as they are dependent on various factors like how accurately the workloads are known, how tight the timing deadlines are, how close the utilization of the system is with respect to the maximum load, etc. As a general rule of thumb, the most effective thermal control optimization strategies which do not degrade performance are [33]:

1. At design time, we must correctly choose the architectural components and place them on the layout based on the expected application loads (memory access, computing power, etc.).

2. At run time, the longer we apply the operating system level corrections, the better results we get, but this implies that one has to have full knowledge of the possible workloads and arrival times of application jobs, which is a restrictive precondition.

Different thermal control mechanisms like dynamic voltage and frequency scaling (DVFS) [34], scaling threshold voltage (or body bias voltage), changing the workload, throttling traffic and routes [35] can be employed to great effect to improve the thermal performance and reliability of the overall system. There are several hardware mechanisms like hardware counters for cores and memories, integrated thermal sensors among others, which can be used to get an idea of temperature of the system under different workload conditions and which form the basis for broader thermal management of the system.

A more detailed and broad classification of the state-of-the-art temperature control strategies and their principles are described in chapter 2. There, we classify the strategies into off-chip and on-chip ones. Later we delve deep into the on-chip temperature control techniques which we further classify

into static (design time) thermal management and dynamic/runtime (adaptive) thermal management techniques. We present the context and perspective for the thesis and show where exactly our contributions fit in that larger context.

1.4 Thesis Objectives

Motivated by the aforementioned observations, this thesis targets: (1) Understanding and identifying the problem of increased temperature contributed by different components of the system by performing extensive thermal modeling and analysis based simulations; (2) developing novel thermal sensing and thermal management techniques. More precisely, the objectives of this thesis include:

1. To build a novel self-timed thermal sensing architecture which converts analog temperature information into digital form. The objective is also to make the sensing architecture more resilient towards various types of noise variations that may occur in the system and also prove that the system is robust enough under different operating temperatures.

2. To understand and identify the problem of temperature that is contributed by various system components such as interconnects, analyse the effect of packaging on the thermal performance of the system, and to tackle the thermal problems in the emerging 3D stacked systems.

3. To develop a thermally efficient interlayer communication algorithm for 3D stacked NoC architectures. The objective is to hybridize a proposed congestion-aware routing algorithm with other available algorithms and mitigate the thermal issues by herding most of the switching activity closer to the heatsink where most of the thermal conduction happens.

4. To develop an efficient thermal-aware application mapping algorithm for 2D planer NoC platforms. The objective of this multi-application mapping algorithm is to best place the blocks which are hotspot prone within a region dedicated for a particular application on the 2D NoC by using a set of extracted metrics obtained from extensive thermal modeling and analysis based simulations. How the presented placement algorithm can keep a balance between temperature on the chip and its performance while running applications is a major consideration.

1.5 Thesis Contributions

To reach the objectives that were set for this thesis, we have built several thermal models and carried out extensive simulations for 2D NoC and 3D stacked NoC architectures. We started our investigations by developing a novel thermal sensing circuit which can be used in a thermal sensing and monitoring infrastructure. Our later work on thermal modeling and analysis of interconnects and 3D stacked systems laid the ground work for a thermally efficient inter-layer communication scheme for 3D NoC systems and an efficient thermal-aware mapping algorithm for a 2D NoC system. The proposed thermal management scheme which relies on herding most of the switching activity to the die closer to the heat sink in a 3D stacked system can be combined with our novel thermal-aware mapping technique for additional thermal safety. This section summarises the main contributions of this thesis work. They are:

1. One of the most cost-effective and accurate temperature measurement technique is the use of thermal sensors in the system. We proposed a self-timed thermal monitoring strategy which is based on the use of thermal sensors. Since leakage currents are sensitive to temperature and increase with scaling, we propose the use of a leakage current based thermal sensing for monitoring purposes. We have implemented a novel thermal sensing circuit, which converts analog temperature information into digital form. We have also proposed a novel thermal sensing and monitoring interconnection network structure based on self-timed signaling, comprising of an encoder/transmitter and decoder/receiver. We have performed power supply noise, additive noise on sensor input signal and dynamic power supply voltage variation analysis on the thermal sensing circuit and show that it is robust enough under different operating temperatures.

2. The complexity of addressing the issue of temperature is such that, that one has to address it starting from earliest design stages of the system. The early design choices like the number and complexity of cores, types of materials and packaging used, dictate the temperature patterns of the system. As a result system designers have begun to study thermal management issues from the early design stages. In order to do so, accurate thermal modeling and analysis at design time is essential. We performed thermal analysis on interconnects and 3D stacked systems by building various thermal models. More specifically,

   (a) We analysed the spatial temperature profile of global *Cu* nanowire for on-chip interconnects. The impact of the temperature rise

along the interconnects has been analysed with two different signal transmission systems namely current-mode and voltage-mode signaling.

(b) A 3D thermal model of a multicore system is developed to investigate the effects of hotspot, and placement of silicon die layers, on the thermal performance of a modern flip-chip package. In this regard, both the steady-state and transient heat transfer analysis has been performed on the 3D flip-chip package. Two different thermal models were evaluated under different operating conditions. Through experimental simulations, we have found a model which has better thermal performance. The optimal placement solution is also provided based on the maximum temperature attained by the individual silicon dies. We have also provided the improvement that is required in the heat sink thermal resistance of a 3D system when compared to the single-die system.

3. One of the primary design goal of any high-performance system is the maximization of performance within the given power and thermal envelopes. If the system is a 3D stacked system, the temperature problem is all the more prominent due to the increased power density. Hence there is an urgent need for thermal management in 3D stacked systems. So, we proposed a thermally efficient routing strategy for 3D NoC-Bus hybrid architectures, which helps in mitigating the on-chip temperatures by herding most of the switching activity to the die which is closer to the heat sink. Our simulations with a real world benchmark show that there has been a decrease in the peak temperatures when compared to a typical stacked mesh 3D NoC architecture.

4. An exploration of various thermal-aware placement approaches for both the 2D and 3D stacked systems is presented. We have developed various thermal models which were used to investigate the effect of thermal-aware placement in 2D chip and 3D stacked systems. A set of metrics were developed which were used to propose an efficient thermal-aware application mapping algorithm for a 2D NoC. Our extensive steady-state simulations show that the proposed thermal-aware mapping algorithm reduces the effective area reeling under high temperatures when compared to the *Tree-Model-Based (TMB)* mapping and *Worst case* mapping.

1.6 Research Publications

The work presented in this thesis is based on and extended from the following peer-refereed journal/articles and peer-refereed conference proceedings. The

contributions of the author in the multi-authored publications has also been elucidated.

1. Kameswar Rao Vaddina, Amir-Mohammad Rahmani, Mohammad Fattah, Pasi Liljeberg, Juha Plosila, "Design space exploration of thermal-aware many-core systems", Journal of Systems Architecture, Volume 59, Issue 10, Part D, November 2013, Pages 1197-1213, ISSN 1383-7621. http://dx.doi.org/10.1016/j.sysarc.2013.08.007. [36]

   Author's contributions: The author contributed with a well-structured problem formulation, algorithm for thermally-efficient inter-layer communication scheme is developed in cooperation with Amir-Mohammad Rahmani, thermal-aware mapping algorithm has been developed in cooperation with Mohammad Fattah, performed thermal simulations and wrote most of the manuscript. System-level simulations for 2D NoC and 3D NoC are performed by Mohammad Fattah and Amir-Mohammad Rahmani respectively.

2. Kameswar Rao Vaddina, Pasi Liljeberg and Juha Plosila. "Exploration of Temperature-Aware Placement Approaches in 2D and 3D Stacked Systems." International Journal of Adaptive, Resilient and Autonomic Systems (IGI-Global IJARAS), Vol. 4, No. 3, pp 61-81, 2013. http://dx.doi.org/10.4018/jaras.2013070104. [37]

   Author's contributions: The author contributed with a well-structured problem formulation, performed thermal simulations and wrote the entire manuscript.

3. Rahmani, A-M., Khalid Latif, Kameswar Rao Vaddina, Pasi Liljeberg, Juha Plosila, and Hannu Tenhunen. "Congestion aware, fault tolerant, and thermally efficient inter-layer communication scheme for hybrid NoC-bus 3D architectures." In Networks on Chip (NoCS), 2011 Fifth IEEE/ACM International Symposium on, pp. 65-72. IEEE, 2011. [38]

   Author's contributions: The author contributed with a well-structured problem formulation, performed thermal simulations for the hybrid NoC-bus 3D architectures and wrote the manuscript.

4. Kameswar Rao Vaddina, Amir-Mohammad Rahmani, Khalid Latif, Pasi Liljeberg, Juha Plosila, "Thermal Analysis of Job Allocation and Scheduling Schemes for 3D Stacked NoC's." In Digital System Design (DSD), 2011 14th Euromicro Conference on, pp. 643-648. IEEE, Oulu, 2011. [39]

   Author's contributions: The author contributed with a well-structured problem formulation, performed thermal simulations and wrote the entire manuscript.

5. Kameswar Rao Vaddina, Tamoghna Mitra, Pasi Liljeberg, and Juha Plosila. "Thermal modelling of 3D multicore systems in a flip-chip package." In SOC Conference (SOCC), 2010 IEEE International, pp. 379-383. IEEE, 2010. [40]

   Author's contributions: The author contributed with a well-structured problem description, built thermal models and performed thermal simulations with Tamoghna Mitra, and wrote the entire manuscript.

6. Kameswar Rao Vaddina, Ethiopia Nigussie, Pasi Liljeberg, and Juha Plosila. "Self-timed thermal sensing and monitoring of multicore systems." In Design and Diagnostics of Electronic Circuits & Systems, 2009. DDECS'09. 12th International Symposium on, pp. 246-251. IEEE, 2009. [41]

   Author's contributions: The author contributed with a well-structured problem formulation, performed simulations and wrote the entire manuscript.

7. Kameswar Rao Vaddina, Pasi Liljeberg, and Juha Plosila. "Thermal analysis of on-chip interconnects in multicore systems." In NORCHIP, 2009, pp. 1-4. IEEE, 2009. [42]

   Author's contributions: The author contributed with a well-structured problem formulation, performed thermal simulations and wrote the entire manuscript.

1.7 Organization of Thesis

The rest of the thesis is organized into 7 chapters. In Chapter 2 we broadly classify the state-of-the-art temperature control techniques and provide their working principles and implementation details. Later, we introduce the concepts of Networks-on-Chip (NoC), 3D NoC's, and describe the problems of temperature associated with those systems and delve into the reasons as to why it is important to deal with those issues at all levels of system abstraction. This chapter puts the contributions of this thesis into context and perspective. Chapter 3 introduces to the self-timed thermal sensing and monitoring approach for multicore systems. It provides a novel thermal sensing architecture and proposes a unique sensing interconnection network. Various noise and supply voltage variation analysis has been detailed in this chapter. The Chapter 4 is divided into two important sections, one of which studies the thermal performance of on-chip interconnects in multicore systems and the other deals with thermal modeling and analysis of 3D stacked systems in a Flip-Chip package. Chapter 5 introduces hybrid

NoC-bus 3D architecture and proposes a thermally efficient routing strategy for 3D NoC's which helps in mitigating on-chip temperatures by routing most of the switching activity closer to the heat sink. Chapter 6 presents an exploration of thermal-aware placement approaches for 2D and 3D systems. It starts by arriving at various metrics which provide thermal guidance to circuit designers. Using the developed metrics a thermal-aware application mapping for a 2D NoC system has been developed and simulation results presented. Finally, Chapter 7 concludes this thesis and presents possible direction towards future research work on thermal-aware software programming.

Chapter 2

Thermal Management Techniques for Microprocessors

Temperature related challenges in modern microprocessor architectures has emerged as one of the key design constraints during the past several years. Higher on-chip temperatures are posing significant reliability concerns thereby causing thermal wear-outs of chips. Thermal wear-outs are the result of several aging mechanisms like electro-migration, hot-carrier injection, negative bias temperature instability (NBTI) and dielectric breakdown. These aging mechanisms are exponentially dependant on temperature. In the case of FPGAs, one of the most important wear out mechanisms are the failure of antifuses. Depending on the physics of the failure mechanism, additional stresses, such as elevated current or voltage, accelerates these failures [43].

Temperature induced errors in on-chip interconnects are also becoming a major cause for concern as the links become more susceptible to faults with the scaling down of technology. Defective links show unacceptably high resistance and therefore increase propagation delays [44]. As their resistivity drops with increasing temperature, the operating frequency of the chip degrades further. While most faults are temporary, about 20% of all errors are caused by permanent or intermittent (lasting up to several cycles) faults [44]. These faults occur because of manufacturing defects or run-time variations, such a multi-cycle delay failures during extended high temperature conditions or permanent faults caused by thermal runaway.

Given that supply voltage is not scaling commensurate to decrease in feature sizes, and the impending approach to the limits of possible air cooling, it is predicted that power densities will continue to rise even if micro-architectural complexity stops increasing [7]. Li et al [45] have concluded that for aggressive cooling solutions, reducing power density is at

least as important as reducing total power consumption. Whereas, for low-cost cooling solutions, reducing total power is more important, as raising power dissipation raises on-chip temperatures even if the power density remains constant [45]. Also, ambient temperatures can vary a lot and hence detailed package models are needed to understand the thermal behaviour of the system under various workloads. For example, depending on the field of application, electronic components are required to operate at different ambient temperatures. Many computing based and factory applications are required to operate at a maximum ambient temperature of 60°C and under natural convection and forced air-cooling [46]. Whereas, for automotive sector, the specified maximum ambient temperature under which the electronic components are supposed to operate for passenger compartment and for under the hood use is around 85°C and 105°C respectively [46]. Hence, any inadequate thermal management and control would lead to a complete system failure.

As discussed in the previous chapter, temperature is a more complex problem and hence the system designers should try to solve it at different levels of design flow, starting from the very early design stages. The early design choices like the number and complexity of cores, types of materials and packaging used, dictate the temperature profile of the system. Hence, in order to arrive at efficient thermal management techniques, accurate thermal modeling and analysis is warranted at design time. But, due to the involvement of complex equations of heat transfer, for thermal simulations and the heavy dependence of RC thermal time constant on environmental, material and packaging parameters, making accurate thermal simulations at design time is complex and time consuming process without the involvement of proper simulation tool chains [47] [48].

Power Consumption Trends for Portable and Stationary Systems

The power consumption and power efficiency requirements for mobile/portable systems is considerably different from the stationary systems which are always plugged into the power source. The power consumption trends for both of those systems are presented below.

Consumer Portable Devices

The current and future power consumption trends for mobile and portable platforms as given by the ITRS roadmap [3] is shown in Fig. 2.1. The figure depicts the total power consumption which is decomposed into static and dynamic power across both the logic and memory. The power consumption

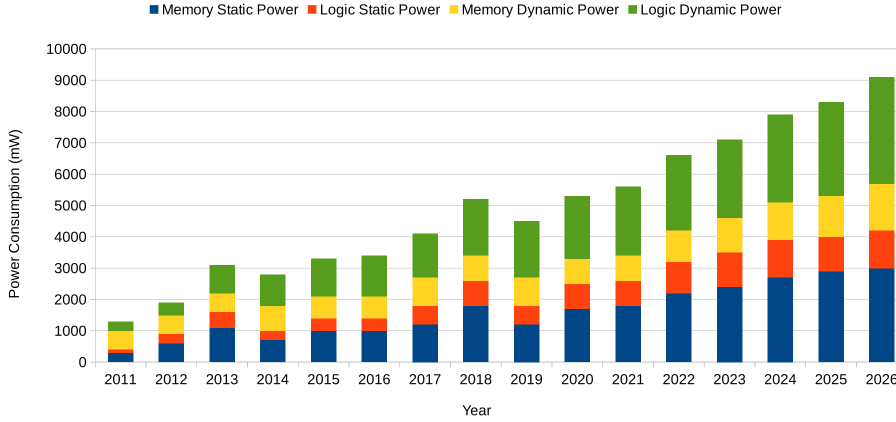

Figure 2.1: Future trends for static and dynamic power for both the logic and memory [3]

trends far exceed the power efficiency requirements [3]. This, combined with the global quest for greener and more energy efficient portable consumer products will lead to a more power-centric designs for the future.

Stationary Devices

The current and future power consumption trends for devices which are stationary and do not have any battery life issues, as given by the ITRS roadmap [3] is shown in Fig. 2.2. The figure depicts the total power consumption which is decomposed into switching and leakage power across both the logic and memory. A look at the trends leads us to believe that the huge increase in power consumption will result in increased chip packaging and cooling costs. At the same time, due to variability and temperature effects, the leakage power might be much greater than what is shown in the Fig. 2.2.

2.1 Power Management vs Thermal Management

A simple equation which is used to measure the operating temperature of a chip and one which gives the relationship between chip power and temperature can be represented by the following linear equation [1]:

$$T_{chip} = T_a + R_\theta . \frac{P_{tot}}{A} \tag{2.1}$$

where T_{chip} is the average silicon junction temperature, T_a is the ambient temperature, R_θ is the equivalent thermal resistance of the silicon (Si)

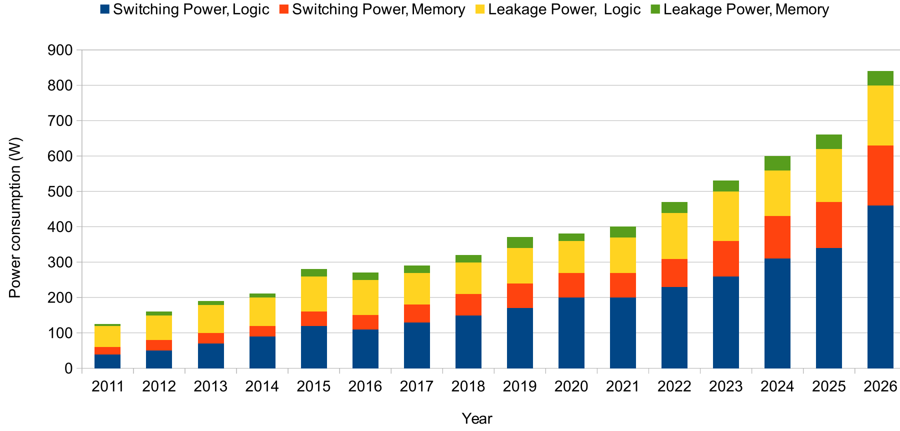

Figure 2.2: Future trends for switching and leakage power for both the logic and memory [3]

substrate, its package and heat sink (in $cm^2\ ^{\circ}C/W$), P_{tot} is the total power dissipation ($P_{dynamic} + P_{short-circuit} + P_{static}$) of the circuit and A is the total chip area in cm^2.

The temperature of the metal interconnect can be represented by the following self-heating equation [49]:

$$T_{metal} = T_{chip} + \Delta T_{self} \tag{2.2}$$

$$\Delta T_{self} = R_E I_{rms}^2 R_{\theta,self} \tag{2.3}$$

where ΔT_{self} is the temperature rise of the metal interconnect due to the flow of current (I_{rms}), R_E is the electrical resistance of the interconnect wire, and $R_{\theta,self}$ is the thermal impedance of the interconnect line to the substrate.

As the increase in temperature is directly related to the chip power dissipation, reducing it would in fact reduce power density and thus help control on-chip temperature issues. However, reducing power alone is not always an effective strategy and may indeed conflict with thermal management [48]. That means, power density increases when underused system components are turned off to reduce power and thereby concentrating more system activity in a smaller area. Conventional power saving techniques typically have very less impact on processor performance, as they try to take advantage of the under-utilization of the processor resources [50]. Whereas, thermal management is mainly a concern when the processor is very heavily used

and any power saving techniques could then hamper the performance of the system [48].

Conventional power management techniques which are used for energy efficiency, might have very limited impact on temperature, as they may target system units which might not be hot at all. Also, power is mainly an instantaneous effect, whereas increase in temperature is a long term process which is spread across in both space (due to the material mass of the silicon die) and time (microseconds or longer). Temperature also depends on the physical properties of the chip materials and also its dimensions and layout, which are not captured very well by just doing power reduction analysis. For example phenomenon like metallization and thermal propagation cannot be captured by using just the power analysis [31]. This means that power management will only affect on-chip temperature if the power reduction optimizations are applied for a sufficiently longer duration. For all the above reasons, power management techniques which are traditionally been used for energy efficiency concerns may not have sufficient impact on thermal related issues. This is attributed to the fact that the energy efficiency policies may be different for both the cases and may even sometimes potentially be in conflict with each other [48].

2.2 Classification of Temperature Control Mechanisms

In this chapter we summarize some of the important thermal management techniques which help improve the reliability of complex microprocessor systems. We have first classified the temperature control mechanisms and then delved deep into each one of those classifications. Kong et al. [48] have also presented a similar survey of recent thermal-aware micro-architecture techniques. When compared to their work, the way we classify temperature control mechanisms is different. But, like them, we also restrict our classification to only temperature related ones, thereby excluding studies which consider power (or energy).

The chip temperature control mechanisms are broadly classified into off-chip and on-chip strategies. The off-chip temperature control mechanism can be further classified as package/system level techniques and board level techniques. Whereas, the on-chip temperature control mechanism can be divided into static (or design time) and dynamic (or runtime) strategies as shown in Fig. 2.3. These techniques are described below.

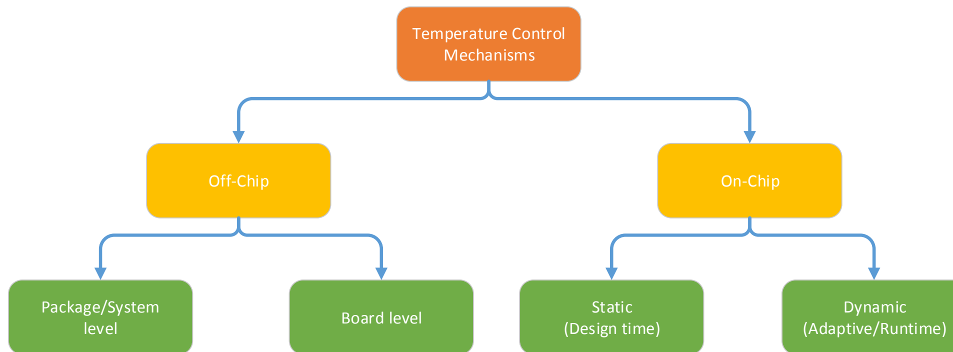

Figure 2.3: Classification of temperature control mechanisms

2.2.1 Off-Chip Thermal Management Techniques

Some older motherboards which use a thermistor placed inside a CPU socket, for thermal management have less accuracy in its thermal measurements. The thermal feedback from the thermistor is used to control the speed of the fan in order to keep the maximum temperature of the chip below a certain predefined threshold temperature value. Another package based off-chip thermal management involves the use of thermoelectric cooler which uses Peltier effect to create a heat flux between the die junction and the heatsink. Such a device pumps heat from the die to the heatsink at the expense of electrical energy [51] despite having drawbacks like added weight, space and expense of thermoelectric cooler and heatsink [1]. Another approach to design local temperature control involves the use of phase-change material (PCM), a substance with high heat of fusion properties [52]. The materials are designed to absorb the generated heat and thereby change its physical state. This method keeps the chip temperature constant although it only works for a limited amount of time. Encapsulation and added system complexity are the two main drawbacks of using phase-change materials for tackling thermal issues using off-chip strategies.

2.2.2 Design-Time Thermal Management Techniques

A large amount of research has been performed on thermal modeling and analysis of temperature in order to predict future thermal related issues at design time and find ways to minimize potential problems [18] [1] [53] [54]. As a result of those developed thermal models, a number of design time thermal-aware techniques have been proposed. They mainly fall under 3 different categories, namely, floorplanning based [18], routing based [55] and coding based techniques [56].

Designers have been using temperature-aware floorplanning techniques which help reduce peak temperatures by equalizing the temperature across the chip during the initial macrocell placement [57] [58] and help evaluate the temperature-performance trade-off early in the design stage. We have also explored design time floorplanning techniques in our work [36] which helped reduce the peak temperature and decreased the amount of chip area reeling under high temperatures. Other than the floorplanning based techniques, designers have studied the effect of substrate thermal gradients on the buffer insertion techniques [59] and concluded that the VLSI interconnects can be made more robust by widening traces and by doing buffer insertion and sizing [1]. Ting et al. [60] have demonstrated that via density strongly influences the spatial distribution of temperature as well as the maximum temperature rise in interconnects. They further show that the optimal spacing of dummy thermal vias in the higher metal layers impact the thermal characteristics of metal interconnects and helps reduce temperatures.

There are several design parameters like various packaging and on-chip electro-thermal parameters which when considered from the very early design stages in the physical synthesis process, allows for a thermally balanced placement of macrocells in order to minimize non-uniform on-chip thermal gradients. All of those parameters can be used by the system designers in a static thermal simulator which can be invoked to address the impact of temperature from the early design stages. Ajami et al. [61] claim that a RTL-to-GDSII system containing an embedded thermal analysis engine that is incorporated with various other components of the optimization flow is needed. Pedram et al. [1] argue that the end goal is to arrive at an EDA methodology, effective on-chip thermal-aware design flow, and integrated tool suite which includes thermal-aware optimization and analysis techniques in order to effectively tackle on-chip thermal gradients and hotspots at design time.

2.2.3 Dynamic Thermal Management Techniques

In this subsection we mainly describe different state-of-the-art adaptive thermal management techniques and control algorithms. That means, given a piece of working silicon, different dynamic thermal management techniques are used to manage on-chip thermal issues which will be explored in here. As power density increases, there is a predominance of localized thermal hotspots which move both in space and time. Dynamic thermal management techniques have been proposed which would address these issues using a class of both micro-architectural and software based solutions thereby having system wide thermal ramifications.

Dynamic thermal management techniques and control algorithms can be broadly classified into heat reduction techniques and heat distribution

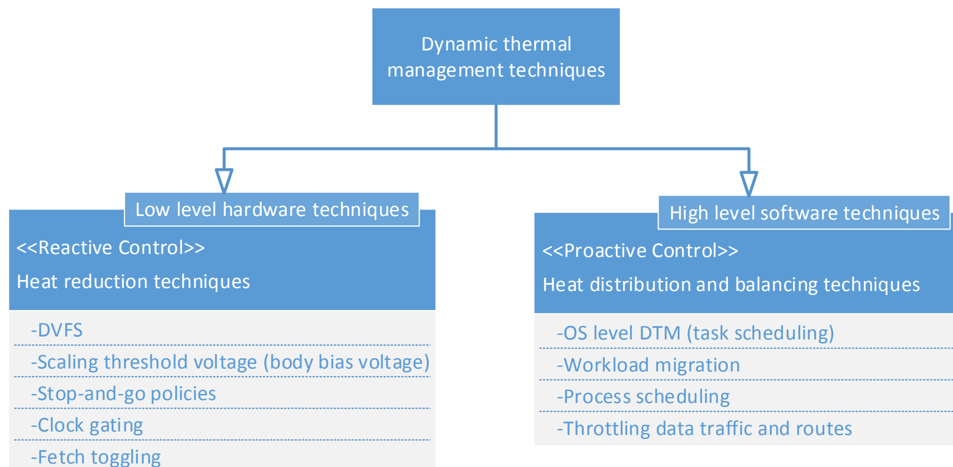

Figure 2.4: Classification of dynamic thermal management techniques and control algorithms

and balancing techniques [62] [34] as shown in Fig. 2.4. It is also possible to classify them as reactive vs proactive control techniques or even low-level hardware vs high-level software techniques. In the first category different power management techniques like dynamic voltage and frequency scaling (DVFS) [34] [63] [64] [65], scaling threshold voltage (or body bias voltage) [66], stop-and-go policies [62], fetch toggling [67] and clock gating [68] are included. Whereas in the second category workload migration [69], task/process scheduling (OS level DTM) [70] [71] [72], throttling data traffic and routes [73] [74] [75], are used in order to distribute and balance temperature across the chip.

DTM in Multicore Architectures

We have classified the dynamic thermal management techniques pertaining to multicore architectures as shown in Fig. 2.5. For multicore architectures which provide increased parallelism, it has been found that thread migration and DVFS techniques are the most promising methods to control temperature [62]. Donald et al. [34] have also come to the similar conclusions when they considered different parameters and schemes for DVFS (both local and global) and thread migration techniques (temperature based, counter based and power based). Li et al. [76] have built a parameterized, transient, thermal behavior models from computed thermal and power information at the architecture level. A feedback control loop which takes this information as input and adjust power and temperature as required. Kadin et al. [77] have developed a frequency planning technique which maximizes total performance of processors under various thermal constraints. Many

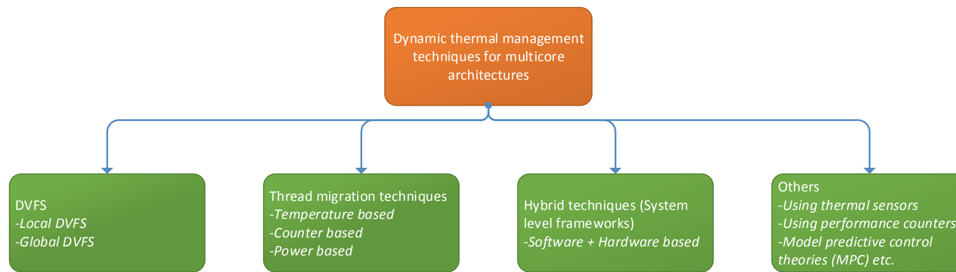

Figure 2.5: Classification of dynamic thermal management techniques in multicore architectures

of the DTM techniques rely on algorithms which assume that power can be assumed accurately at run-time. While others prescribe an oversimplified feedback based control mechanism which does not give any guarantees. Wang et al. [78] proposed a chip-level temperature-constraint power control algorithm which is based on Model Predictive Control (MPC) theory which can precisely control the power of a multicore chip to desired value while maintaining the temperature below a certain threshold. Their algorithm outperforms current state-of-the-art DTM techniques [79].

It is very important to have a broad overview and the trade-offs involved when using both the hardware based and software based DTM techniques. Hardware based DTM techniques have high execution time overhead and they are usually global in nature and ignore any application specific information. As Amit et al. [80] describe that in the case of thermal emergencies all the involved applications are equally penalized and all of them suffer an equal impact in performance. On the other hand software based DTM techniques like the OS level energy-aware process scheduling strategies [81] [82] are not very aggressive and have very low performance impact. These techniques are able to take application specific thermal behavior into account. In the following we describe operating system level and other software application level DTM techniques in more detail.

Operating System Level DTM

Other DTM techniques at the operating system level consider thermal-aware task scheduling [83]. Many of those DTM techniques take corrective measures only after the temperature reaches a certain predefined threshold value. That means, they react to their environment. Where as, Yeo et al. [84] have proposed a predictive DTM for multicore systems by modifying the task scheduler of Linux kernel. This allows to reduce the overall temperature with minimum performance overhead. Whereas, Coskun et al. [85] tried to investigate predictors for forecasting future temperature and workload

dynamics and thereby proposing proactive DTM techniques for multicore systems. All of these techniques bring forth two important and significant problems which needs to be addressed. More so because the accuracy of said thermal measurements directly impacts the performance of the system and the performance of the thermal management unit [86]. They are a) proper thermal sensing and modeling b) variation in power consumption based on software workload [79].

On-Chip CMOS Thermal Sensors

In order to properly sense the temperature, it is very important to have an accurate reading of on-die temperature. Since, a single temperature-diode on the die is not enough to get an accurate thermal profile of the entire chip, it is important to distribute temperature and leakage sensors throughout the chip. DTM techniques which rely on on-line thermal sensing and monitoring of temperature have to solve the problem of distribution/placement of thermal sensors on the die, the number of thermal sensors which can give an approximate thermal profile of the chip [87] [88] [89]. IBM's Power7 server processor has around 44 thermal sensors [90]. Whereas, Intel's SCC chip has around 96 thermal sensors [91] in total. Temperature measurement using thermal sensors suffer from various measurement inaccuracies, like the need for calibration, analog to digital conversion accuracy, proximity to thermal hotspot, granularity of the profiling or the sampling time among others.

A smart thermal sensor usually has different target specifications like cost, accuracy, resolution, supply voltage, supply current, speed and operating temperature range. All of these specifications vary depending on the application area. An important element for thermal sensing on silicon die is a transistor. Considering that the other alternatives like the resistors are difficult to manufacture accurately and the reference resistors are even more worse [92]. The integrated thermal sensor designers use of Bipolar Junction Transistors (BJTs) for thermal sensing, as their base-emitter voltage can be used to obtain the thermal voltage (kT/q), which is proportional to absolute temperature (PTAT) [93]. Both lateral and vertical substrate transistors can be used to implement these integrated thermal sensors. The problems associated with the BJT-based thermal sensors are that they require complicated calibration and also have non-linear dependency on temperature thereby requiring large area output interfaces. In order to eliminate these problems Poki et al. have introduced a time-to-digital converter based sensor [94] which does not involve the voltage/current analog-to-digital converter (ADC) or the bandgap reference. This sensor first generates a pulse with the width proportional to the measured temperature. A cyclic time-to-digital converter is used to convert the pulse into digital measurement [94]. These low-area and low-voltage thermal sensors have also been implemented on an FPGA [95].

Qikai et al. have designed a low overhead process variation tolerant temperature sensor with good sensitivity over a wide temperature range [96]. This temperature sensor uses a differential amplifier to minimize the temperature dependence of V_{th}. Michiel et al. presented a CMOS smart temperature sensor which achieves of only $\pm 0.1°$C (3σ) over the military range [97]. The errors caused by the readout circuitry have been reduced to 0.01°C level. Anton et al. presented a CMOS smart temperature sensor with digital output which only consumes around 7μW [98]. This extreme low power consumption is achieved by means of a facility which switches off the power supply after each sample. The temperature conversion to the digital domain is done by using the sigma-delta converter which makes less susceptible to digital interference. Pablo et al. have introduced a leakage based ultra low-power (1.05-65.5 nW at 5 samples/sec) and tiny (10250 μm^2) CMOS thermal sensor [99]. This sensor outperforms all the previous works and reduces both the area and the power consumption by more than 85%. It is also insensitive to spatial thermal gradients due to its smaller sensing part. At the same time, since this sensor has low power dissipation is considered to be very robust against self-heating issues. In Chapter 3, we have implemented a novel thermal sensing circuit in 65nm CMOS technology, which converts analog temperature information into digital form. Since the functionality and response of a circuit can be affected by the presence of disturbing noise sources on or off the chip, we have analysed the performance of our circuit under different noise conditions and found that it is robust enough. This noise analysis has not been done by the previous researchers and is novel to our work.

Thermal Modelling of On-Chip Networks

Accurate thermal models which can capture the thermal behaviour of the system and thermal packages with minimal architectural input parameters are needed at an early-design stage. Heat conduction across the chip and the package can be modelled after Fourier's law, which states that the rate of flow of heat through a surface is proportional to the negative temperature gradient across that surface. Skadron et al. have used the Fourier heat flow analysis and have presented a dynamic compact thermal model at the micro-architectural level in [100] and [101] for integrated circuit chip-package thermal analysis. Their simulation tool which is called *Hotspot* constructs a multi-layer lumped thermal RC network to model the heat dissipation path from the silicon die to the ambient [35]. Hotspot takes the floorplan of the silicon die and partitions it into functional blocks and connect the various blocks with the help of the thermal RC network. This thermal RC network is then solved in order to give the temperature at each node. The equivalent thermal circuit is constructed by taking into account that the heat flow in

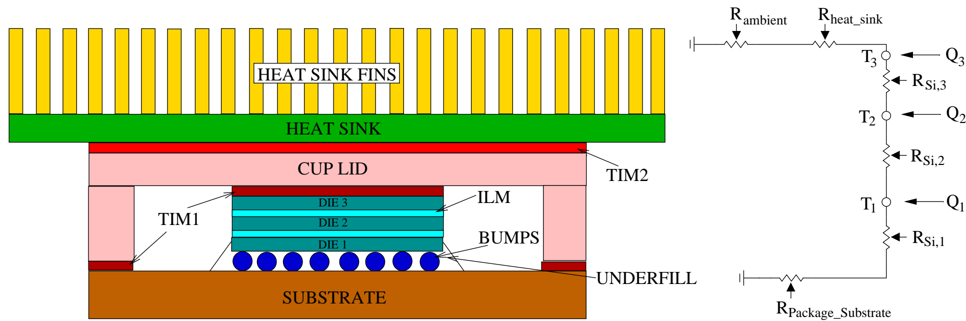

Figure 2.6: A simple thermal equivalent circuit for a 3D stacked system in a flip-chip package

the die is analogous to an electrical current, temperatures are analogous to voltages, heat sources are represented by constant current sources and the absolute thermal resistances are represented by resistors and thermal capacitances by capacitors. A simple thermal equivalent circuit for a 3D stacked system in a flip-chip package is shown in Fig. 2.6. In the figure, R is the thermal resistance, T is the temperature at that node and Q is the heat generated at that node.

Wang et al. have presented an efficient 3-D transient thermal simulator based on the full-chip layout using the alternating direction implicit (ADI) method, which instead of solving the 3-D problem, solves three one-dimensional problems in succession [102]. Clemens [103] presented two package thermal models (one for PFQP-style package and the other for BGA-style package). William et al [104] have presented a thermal modelling approach which is based on analytical solutions of heat-transfer equations. Their model is mainly focussed at the device level. A study conducted by Shang et al [35] on MIT Raw chip show that the on-chip networks have considerable impact (almost comparable to processing nodes) and contribute to the increase in the overall chip temperature. They have developed an architectural thermal model for on-chip networks that take into account the thermal impact of interconnects. Since, none of the mentioned thermal models work at different granularities (like circuit structures, standard cells, functional unit blocks, etc.) and do not work at different levels (like silicon surface, interconnect, package, etc.) Huang et al [105] have proposed compact thermal model which not only works at different granularity level, but also at different levels and can be easily integrated into existing CAD tools to achieve temperature-aware design. Later, they have extended this modelling methodology for early-stage VLSI design [106] and after that to build an accurate, pre-RTL temperature-aware design using a parameterized, geometric thermal model [107].

Impact of TSV's on Temperature Profile

3D chip stacking with TSV's has been identified as an effective way to achieve performance boost as well as better power performance [25]. However such solutions contribute to increased thermal profile of the systems. Bryan et al [108] has shown that a 3D floorplan of a high performance microprocessor from Intel (Pentium 4) has led to 15% increase in performance while lowering the power consumption by 15% with an apparently small 14%°C rise in peak temperature. This work assumes a face-to-face bonding and uses TSV's to connect the C4 I/O bumps to the active regions of the two dies. Jung et al [109] have mapped OpenSparc T2 chip into a 3D stacked system and developed design methodologies which resulted in 52.3% reduction in footprint, 25.5% reduction in wire length, 30.2% lower buffer call count and a 21.2% reduction in power compared to the 2D planer design. They make use of 2979 TSV's in their design for 3D placement. They have assumed the TSV diameter, height, resistance, and capacitance as 3μm, 25μm, 50mΩ and 30fF respectively. Zhang et al. [55] have proposed a temperature-aware 3D routing algorithm by inserting "thermal vias" and "thermal wires" to lower the effective thermal resistance of the material and reduce on-chip temperature. However, the TSV's are usually larger by the order of several tens of times when compared to logic gates and memory cells [110]. Therefore, their strategy reduces temperature at the expense of area [28] [111] [112] [113]. Hsu et al. [114] have proposed the use of an architecture with stacked signal TSV's with a two-stage TSV locating algorithm which reduces the temperature by 17% with only a 4% wiring overhead and 3% performance loss.

Software Application Level DTM

Software level thermal management can be used as an extension to low-level hardware based DTM techniques. Lee et al. [115] have used the hardware performance counters for cores and memories in order to provide a software based solution for runtime thermal sensing which can be used for thermal profiling of software applications. Meng et al. [116] have provided a software based framework which addresses both energy efficiency and thermal management in a unified way and it delivers around 40% energy reduction with negligible slowdown in the application. Similarly, Huang et al. [117] have also proposed a framework for dynamic energy efficiency and temperature management which maximizes energy savings without extending application execution times too much and to guarantee that the temperature remains below a certain threshold.

In this thesis work we have addressed a DTM strategy at design time by first exploring different thermal-aware placement approaches for both 2D and 3D stacked systems. We then proposed a static application mapping

algorithm which reduces the effective area reeling under high temperatures on the chip. We have also developed a thermally efficient routing strategy which works at run-time to reduce temperatures for a NoC-Bus hybrid architectures by herding most of the switching activity closer to the heat sink.

2.3 Summary

In this chapter we have briefly explained why power management techniques cannot substitute exclusive temperature management efforts. Later, we have broadly classified the temperature control mechanisms into off-chip and on-chip mechanisms. The off-chip mechanisms can be further classified as package/system level techniques and board level techniques. Whereas, the on-chip techniques can be classified into static (or design time) and dynamic (or runtime) strategies. We concluded by identifying where the following thesis fits in, into the larger scope of temperature management strategies.

Chapter 3

Self-Timed Thermal Sensing and Monitoring of Multicore Systems

As the number of cores increases thermal challenges increase, thereby degrading the performance and reliability of the system. We approach this challenge with a self-timed thermal monitoring method which is based on the use of thermal sensors. Since leakage currents are sensitive to temperature and increase with scaling, we propose the use of a leakage current based thermal sensing for monitoring purposes. In this work we have implemented a novel thermal sensing circuit in 65nm CMOS technology, which converts analog temperature information into digital form. We have also proposed a novel thermal sensing and monitoring interconnection network structure based on self-timed signaling, comprising of an encoder/transmitter and decoder/receiver. We have performed power supply noise, additive noise on sensor input signal and dynamic power supply voltage variation analysis on the thermal sensing circuit and show that it is robust enough under different operating temperatures.

3.1 Introduction and Motivation

Future generations of distributed on-chip systems would have system modules which are operated at optimal points by adaptively adjusting operating, manufacturing and environmental conditions leading to the design concept of "Always Optimal Design" [22]. For a given processing element function, activity factor and implementation instance there exists an optimal operating point which minimizes the energy performance space [118]. This optimal operating point is a function of variability of activity, process variations, dynamic environmental variations and temperature.

As the technology scales down and power density increases, a lot of factors like power dissipation, leakage, data activity and electro-migration contribute to higher temperatures, larger temperature cycles and increased thermal gradients all of which impact multiple failure mechanisms [7]. The increase in temperature leads to increase in leakage and thus forming a part of vicious circle leading to significant drop in performance of the distributed on-chip network. Hence, there is a great need to constantly keep the functional blocks at the optimal point by adaptively monitoring the thermal activity of such a distributed on-chip multicore system. To keep different functional blocks at optimal points we need to first identify the location of thermal hotspots and take corrective measures. Different thermal control mechanisms like dynamic voltage and frequency scaling (DVFS) [34], scaling threshold voltage (or body bias voltage), changing the workload, throttling traffic [35] and routes can be employed to great effect to improve the performance and reliability of the overall system.

In a multicore scenario power scales up with the increase in the number of cores. With power, thermal challenges become abound which needs to be addressed urgently. The increase in the on-chip temperature not only degrades the performance of the chip but its reliability will be called into question. Srinivasan et al. [119] have shown that the mean time to failure decreases with the increase in temperature. Hence, there is a great need to monitor the on-chip temperature accurately. Under or over estimation of the thermal profile of the chip leads to significant reliability issues.

There are different types of multicore architectures like homogeneous, heterogeneous and morphic architectures [7]. In this work we assume homogeneous architecture of the multicore system. The thermal behavior of such a homogeneous multicore system is not only application and architecture dependent but is also inherently distributed in nature. Thermal emergencies can occur at different locations on the multicore chip and often change dynamically as heat spreads from one block/core to another due to the differences in the temperature. A network of sensors that span the whole multicore system should be employed for accurate thermal modeling and profiling.

Due to the presence of multiple clock domains, high communications costs involved, prohibitive task of managing timing constraints [120] and centralized nature of our proposed thermal monitoring, where in being the highest decision making body, the central Thermal Control Unit (TCU) needs to sense the temperature at any time t. So, a thermal sensing approach which is based on self-timed signaling method is a more natural approach one than a synchronous one.

In this work we assume that there are N Thermal Sensing Circuits (TSC) operating in each core which sense the temperature and calibrate it into known digital form. The request to perform the thermal sensing operation

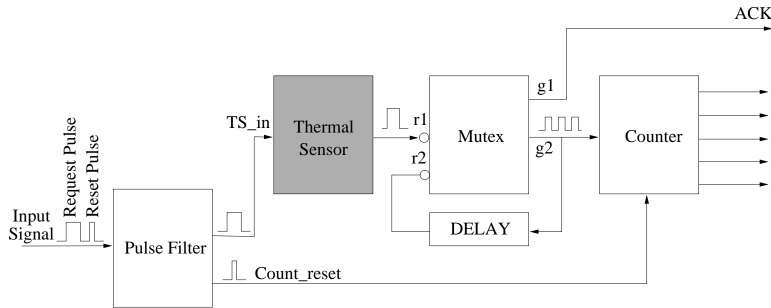

Figure 3.1: Thermal Sensing Circuit (TSC).

comes from the central TCU and when the thermal sensing operation is performed the acknowledgement signal is sent. The data transfer between the TCU unit and the TSC takes place over the self-timed interconnection network.

We will be talking about the thermal sensor, interfacing to the thermal sensor in Section 3.2 describing the thermal sensing architecture, sensing interconnection network in Section 3.3 and give noise analysis simulation results of the thermal sensing architecture in Section 3.4.

3.2 Thermal sensing architecture

In this section we provide a self-timed thermal sensing architecture which senses temperature and converts it into known digital form. The architecture consists of thermal sensor and its digital interface (consisting of pulse filter, MUTEX, delay element and a cycle counter). The block diagram of the thermal sensing architecture is shown in Fig. 3.1. It interacts with the external environment via an asynchronous protocol as shown in Fig. 3.2. The functionality of TSC as a whole and each block within is explained below.

The input signal which drives the whole TSC comes from the Thermal Control Unit (TCU) and consists of a train of two pulses, one a request pulse and the other a reset pulse. The request pulse is generated by a clock in the TCU, whose frequency is known and bounded by the minimum temperature that the sensor needs to measure. As and when the TCU decides to get the temperature profile of a particular hotspot within its domain area, it sends a request signal to the TSC. The TSC obliges by giving the temperature data in a digital format and then raises the acknowledgment signal. Then the TCU sends a reset pulse which resets the cycle counter in the TSC.

Pulse filter: The pulse filter separates the pulses (*request* and *reset*) on the input signal and latches them onto the output. Fig. 3.3 shows the implementation of the pulse filter and its timing diagram at 27°C and 60°C can be

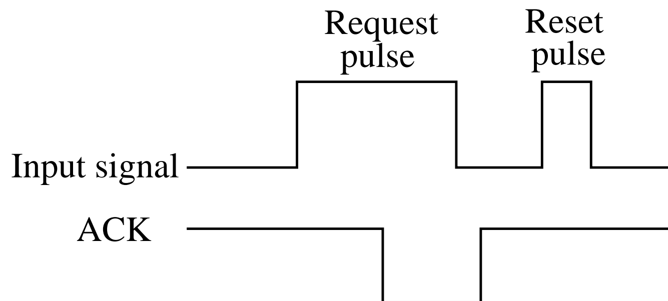

Figure 3.2: Self-timed handshaking protocol for the thermal sensing architecture.

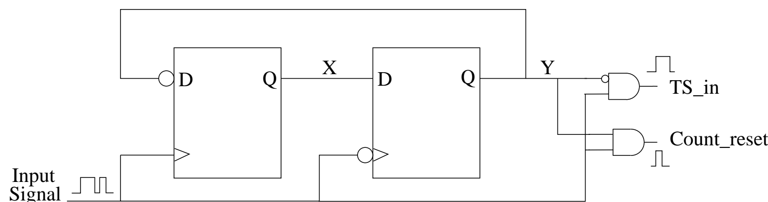

Figure 3.3: Pulse Filter.

traced to Fig. 3.4 and Fig. 3.5 respectively. The *request* pulse output (signal TS_in) goes to the input of the thermal sensor and the *reset* pulse output (signal count_reset) of the pulse filter is used to reset the cycle counter.

Thermal sensor: The thermal sensor is an integral and most crucial part of our thermal sensing architecture. Since, continuous scaling of CMOS technology into the nano domain increases the leakage currents which are sensitive to temperature variations, we propose to make use of these leakage currents in the design and implementation of thermal sensors. Ituero et al. [4] have proposed a leakage based on-chip thermal sensor in 0.35μm CMOS technology as shown in Fig. 3.6. We have simulated this sensor in 65nm CMOS technology from ST microelectronics. The input to the sensor is a pulse which is generated by a clock in the TCU, whose frequency is known and bounded by the minimum temperature that the sensor needs to measure. The output of the sensor is an analog signal whose pulse width varies with temperature. When temperature increases, the leakage currents of both NMOS and PMOS transistors increases. The PMOS leakage current charges the capacitor while the NMOS discharges it. By properly sizing these two transistors it is possible to control the amount of charging and discharging. In other words, the width of the output pulse. So, by calibrating the width of the pulse from the sensor, it is possible to determine the hotspot temperature.

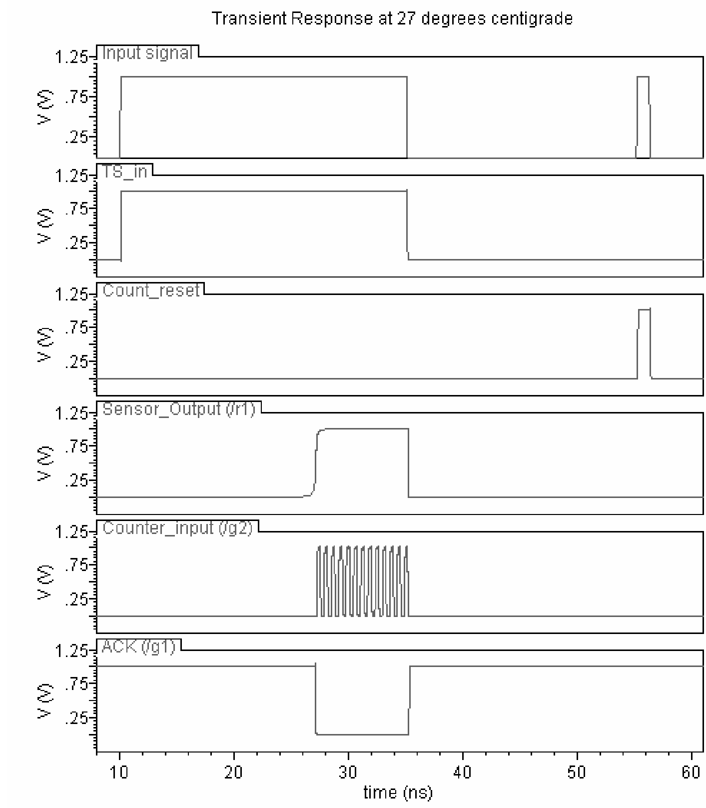

Figure 3.4: Timing diagram of the Thermal Sensing Circuit (TSC) at 27°C.

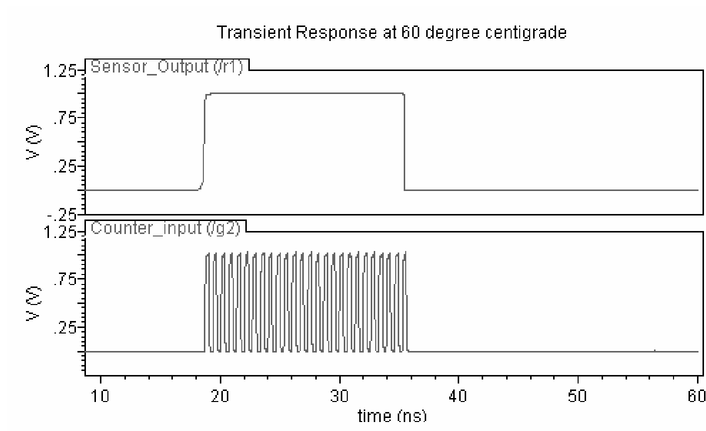

Figure 3.5: Timing diagram of the Thermal Sensing Circuit (TSC) at 60°C.

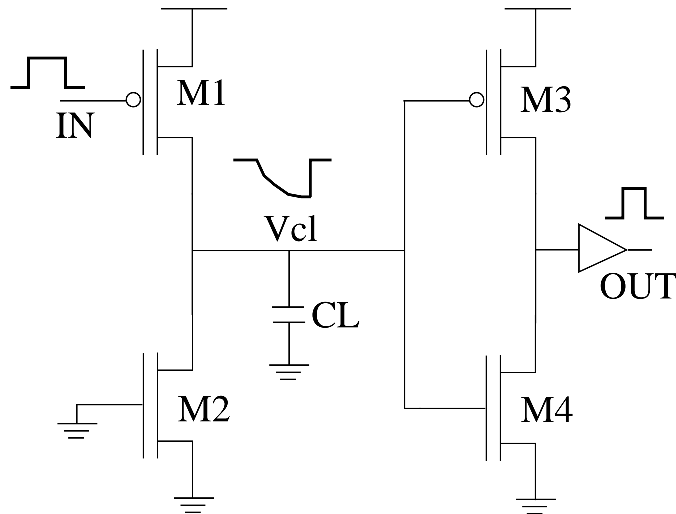

Figure 3.6: Leakage current based thermal sensor [4].

We have done several simulations so as to ascertain the functionality of the thermal sensor under different operating conditions. The simulated output pulse width and the delay of the sensor versus the temperature is shown in Fig. 3.7 and Fig. 3.8 respectively.

The output of the thermal sensor shown in Fig. 3.6 is the duration of a pulse which is in the analog domain. This pulse duration must be calibrated by converting it into a known digital format. This is done by using a digital interface circuit consisting of MUTEX, delay elements and a cycle counter.

MUTEX: The mutual exclusion element (or MUTEX, as it is called) is the basic building block of an arbiter. Fig. 3.9 shows the circuit schematic and timing diagram of the MUTEX element. It involves a bistable SR latch and a metastability filter. The requests *r1* and *r2* come from two independent sources. The role of the MUTEX is to pass those inputs to their corresponding outputs *g1* and *g2* in such a way that only one output is active at any given time. If there is only one request signal then the corresponding grant signal will be asserted. If one input request signal arrives well before the other, then the second signal will be blocked until the first request signal is de-asserted. When both the requests are active simultaneously then it passes only one of the pending requests and the selection is non-deterministic (arbitrary) between requests. In this case the circuit enters metastable state before arbitrarily settling down to either of the known stable states. We couple the MUTEX with a delay element to produce a clock for the cycle counter. As long as the input *r1* of the MUTEX is asserted (for the amount of pulse duration of the thermal sensor), the second input *r2* is latched onto the output *g2*. The output *g2* is coupled with a delay element and fed back

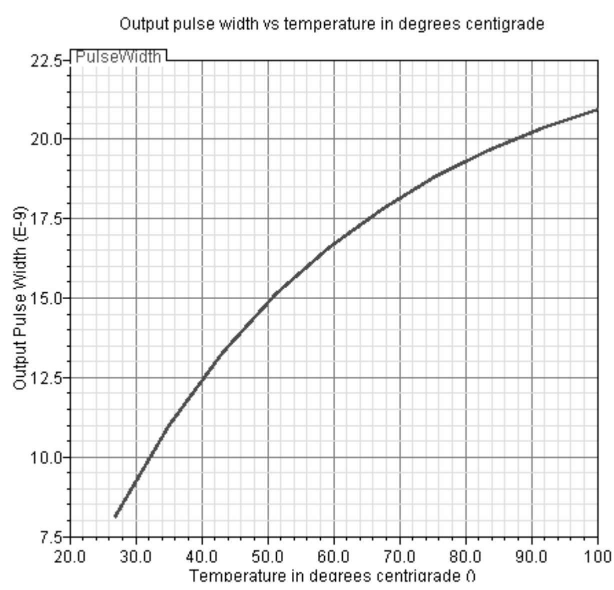

Figure 3.7: Response of the sensor in the 27°C to 100°C range as simulated in 65nm technology.

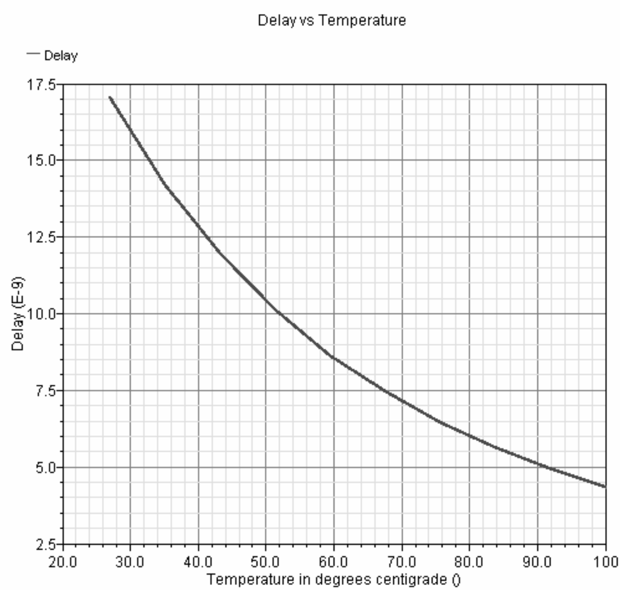

Figure 3.8: The delay through the sensor, plotted against the temperature.

to the input *r2*, thus forming an oscillator and creating a clock signal at the output *g2* whose pulse width is equal to the propagation delay of the delay element. When the input *r1* is de-asserted then the output *g1* is raised high, thus confirming the completion of the operation of producing the clock for the cycle counter. In Fig. 3.1, this output has been named as the acknowledgment (ACK) signal.

Delay element: Fig. 3.11 shows the linear variation in the propagation delay of the delay element with respect to the increase in temperature. For every 5°C raise in temperature, the propagation delay increases by 1.15ps. This sensitivity (propagation delay) of the delay element changes with respect to the temperature which in turn changes the time period of the generated clock. This change in the generated clock calls into question the accuracy of the counter and hence the thermal sensing in general. The increase or decrease in the propagation delay of the delay element (or the number of clock cycles at the input of the cycle counter) according to the increase or decrease in temperature variation can be taken care of by encoding. Fig. 3.13 shows the way we actually encode the temperature. The temperature range is divided into four different intervals according to the count value of the counter. So, when the number of clock cycles increases or decreases, it is already implicit in the encoding and does not have any effect on the accuracy of the thermal sensor.

Cycle counter: The cycle counter shown in Fig. 3.1 has an asynchronous reset. It measures the number of cycles which is nothing but the digital representation of duration of the pulse at the output of the thermal sensor. This result is communicated to the central TCU for suitable action. Fig. 3.12 shows the increase in the number of clock cycles being counted by the cycle counter at different temperatures.

Simulation results: The circuit shown in Fig. 3.1 has been simulated using 65nm technology from ST microelectronics under $Cadence^{TM}$ environment. Fig. 3.4 and Fig. 3.5 show the timing diagram of TSC at 27°C and 60°C respectively. It can also be seen from them that, there is an increase in the sensor output pulse width (and hence the number of clock cycles) at 60°C, compared to 27°C.

3.3 Sensing interconnection network

Considering a large distributed sensor network, communicating thermal data from the thermal sensing circuit (TSC) to the central Thermal Control Unit (TCU) is a challenging task. Since the TCU as a monitoring and decision making body, it needs to get information at any time t from the TSCs spread across the network, and a self-timed communication is a more natural approach than a synchronous one. The proposed sensing interconnection

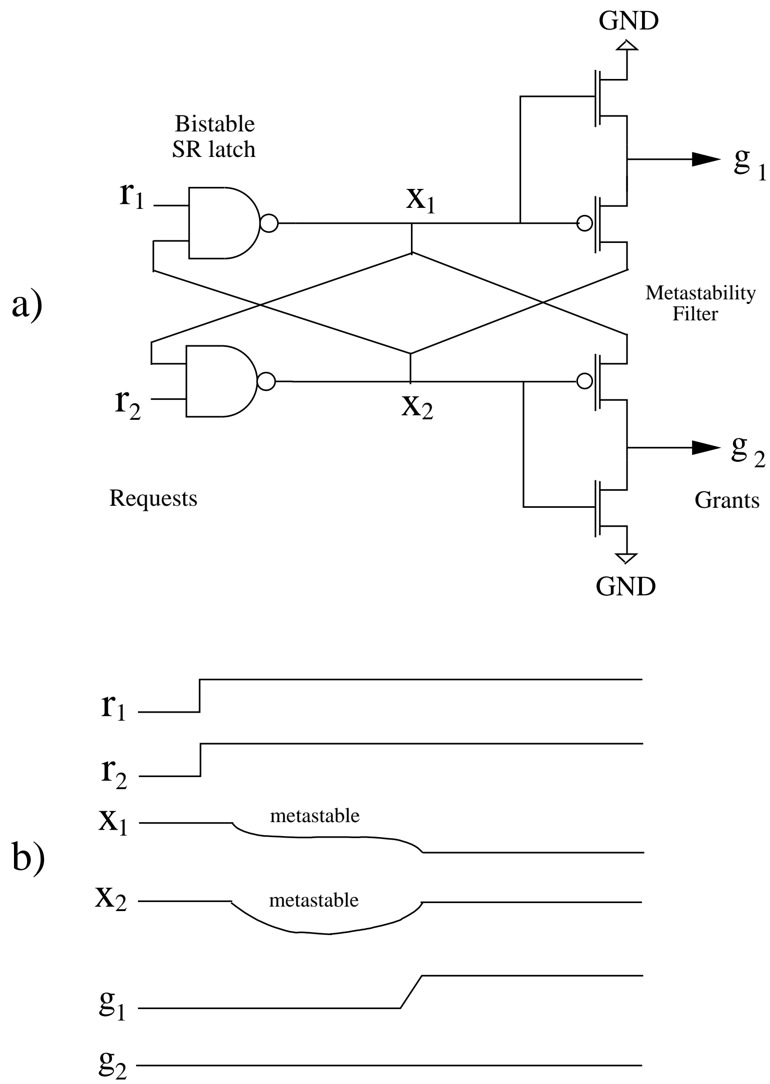

Figure 3.9: MUTEX and its timing diagram.

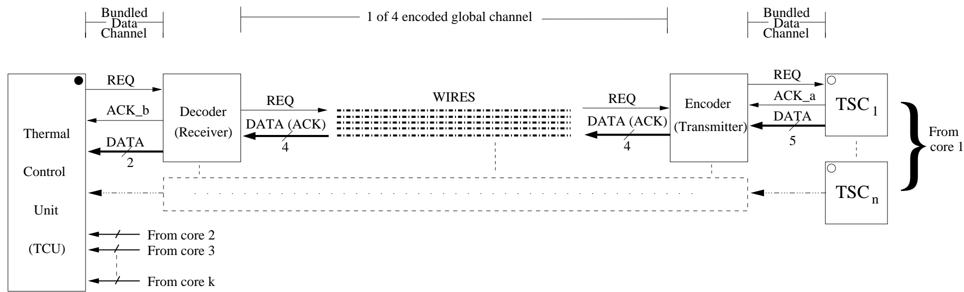

Figure 3.10: Self-timed signaling architecture for sensing interconnection network.

network is shown in Fig. 3.10. In self-timed communication via a pull-type channel, the receiving side initiates communication whenever it needs thermal data, and the sender acknowledges by sending thermal data.

The interconnect between TCU and TSC consists of a receiver (/decoder), global and local wires, and transmitter (/encoder) as shown in the Fig. 3.10. Whenever TCU needs thermal data, it initiates the communication by sending request to the decoder. The decoder in turn forwards the request to the encoder through global wires. The encoder outputs the request signal to be used as input to the TSC. As soon as the TSC gets request input it performs thermal sensing and outputs thermal data (i.e., number of clock cycles). The output of TSC is 5-bits in bundled-data encoding form. In the encoder this 5-bit thermal data is mapped to a symbol consisting of 2-bits depending on the sensed temperature as shown in Fig. 3.13. For example, if the sensed temperature is 45°C, the output of TSC_1 is '10110' (22 clock cycles). The encoder maps this 5-bit data to symbol '01'. The transmitter sends this symbol using four-phase 1-of-4 encoded global channel. In 1-of-4 encoded transmission a group of four wires is used to transmit two bits of information per symbol. A symbol is one of the two-bit codes 00, 01, 10, and 11 and it is transmitted through activity on one of the four wires. 1-of-4 encoded transmission is chosen due to its delay-insensitive feature because in such global data transfer signal propagation delay is unavoidable. In delay-insensitive communication in which the data validity or acceptance is transmitted implicitly within the data operates correctly regardless of the delay variations in the interconnecting wires. Besides being delay-insensitive, 1-of-4 encoding has more immunity against crosstalk effects as compared to the bundled-data encoding, because the likelihood of two adjacent wires switching at the same time is much smaller. Furthermore, it has smaller dynamic power consumption than the simpler delay-insensitive dual-rail encoding. Voltage-mode signaling with repeater insertion can be used, because communicating thermal data does not require advanced high-performance signaling schemes.

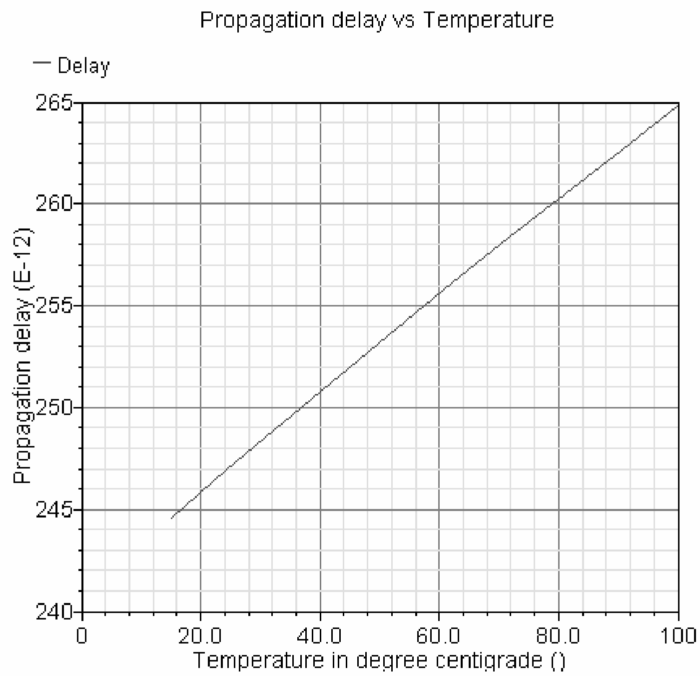

Figure 3.11: Propagation delay of the delay element vs temperature.

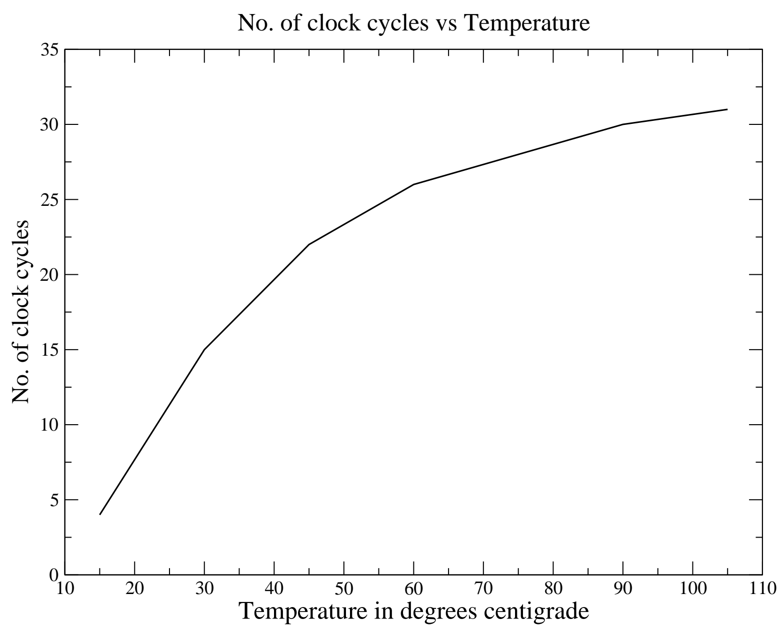

Figure 3.12: No.of clock cycles vs temperature.

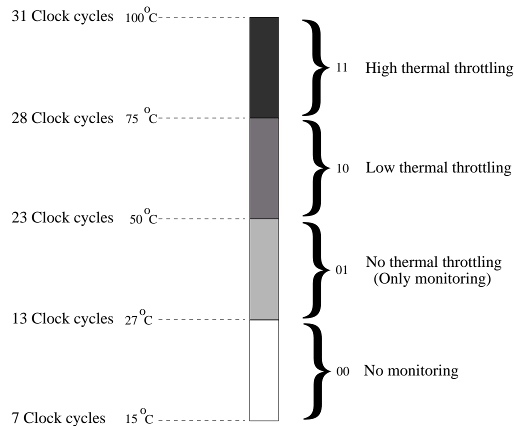

Figure 3.13: Encoding of temperature.

According to the experiments conducted by Puyan et al. [121], the automatic thermal throttling of Intel pentium 4 processor occurs at around 67°C. The emergency reset occurs at 135°C [122]. Assuming, that future generations of complex multicore systems would have to be dealt with, in a similar way where in for certain temperature range below the threshold, we do not do any sort of thermal throttling, when the temperature crosses certain threshold value we start the thermal throttling process and the emergency reset comes into play when the temperature reaches catastrophic proportions. Based on this approach we have divided the temperature range into four different intervals of no monitoring, monitoring, low thermal throttling and high thermal throttling as shown in Fig. 3.13. How the temperature ranges should be divided, actually depend on the technology being used, the kind of application that is running on the system and the thermal profile of the system. In this regard our approach can be modified to support more than 4 levels.

3.4 Noise and supply voltage variation analysis

The functionality and response of a circuit can be profoundly affected by the presence of disturbing noise sources on or off the chip. Also, scaling of technology coupled with the continuous reduction of supply and threshold voltages makes it difficult to manage such noise sources on large and complex systems. In this section we have simulated the circuit in Fig. 3.1 under different noisy environmental conditions. We analyse the performance of the circuit by verifying whether the noise has any impact on the output (pulse width) of the thermal sensor. We have performed power supply noise, input signal noise and dynamic voltage variation analysis as described below.

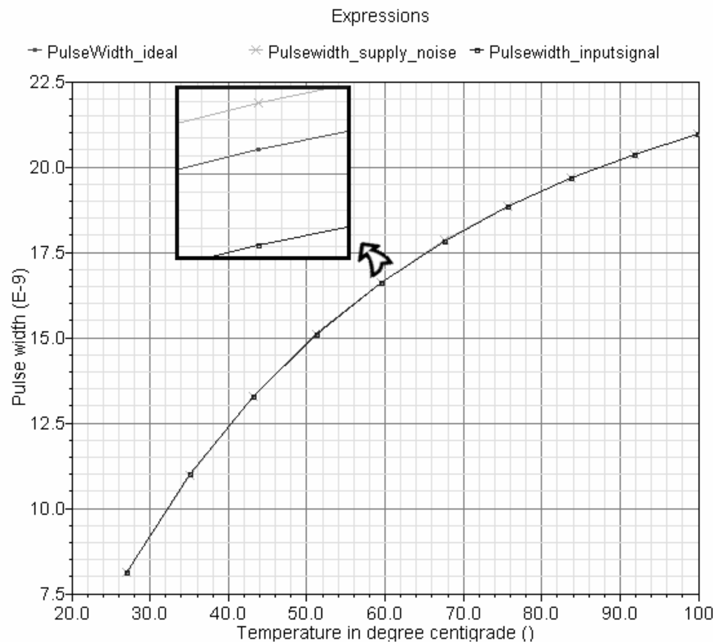

Figure 3.14: Output pulse-width of the thermal sensor with noisy power supply rails and input signal noise vs the one with the ideal power supply for different temperature values.

3.4.1 Power supply noise (PSN) analysis

Power supply noise, one of the largest sources of noise in a digital system, is mostly produced by simultaneous clock-induced switching of CMOS circuits which causes high peak current draws from the power source. In large and complex designs this noise is caused by the synchronous operation of the circuit. Such clock-induced switching forces gates and flip-flops to change their states at around the same time. Also, large bus and interconnect drivers have a significant total current draw when they switch simultaneously thus contributing enormously to the total system noise [123]. We have modelled and injected the noise whose assumed bandwidth is 15GHz and amplitude is 100mV (which is 10% of Vdd; where Vdd=1V) in the power rails of the circuit shown in Fig. 3.1. Fig. 3.14 shows that the pulse width of the thermal sensor with power supply noise, closely follows the one with no noise and their difference between them is just a few pico seconds (as also noted in Table 3.1). This means that the number of clock cycles remain unaffected with this level of power supply noise.

3.4.2 Input signal noise (ISN) analysis

Noise from various sources including thermal vibrations of atoms (thermal noise) adds additively to the signals over the interconnects. They are a major impediment for the transmission of signals and power distribution through the interconnects due to self-heating caused by the flow of currents [124]. In our case, since the interconnect wires connecting the central Thermal Control Unit (TCU) and the Thermal Sensing Circuit (TSC) are affected by self-heating, there is a high likelihood for the input signal to get affected by noise. In this context we analyse the effect of noise on the input signal which comes from the TCU to TSC. The pulse filter which receives the input signal is a digital circuit and thus robust to noise. Fig. 3.14 shows that the output pulse width of the thermal sensor with input signal noise, closely follows the one with no noise and their difference is just a few pico seconds (as also noted in Table 3.1). This means that, like in the power supply noise analysis, even in this case the number of clock cycles remain unaffected with input supply noise.

Table 3.1: |PSN-ideal| and |ISN-ideal| values at different temperatures. Where PSN and ISN stands for power supply noise and input signal noise respectively.

	Difference in pulse width's	
Temperature	|PSN-ideal|	|ISN-ideal|
At 27°C	16ps	9.8ps
At 50°C	12.5ps	11ps
At 75°C	7.5ps	9.13ps
At 100°C	2.8ps	3.68ps

3.4.3 Supply voltage variations

Another major concern for designing high-performance multicore systems is the supply voltage variations. These variations occur when processor activity rapidly changes the current consumption over a very small period of time. Since, the subsystem which delivers power can have substantial parasitic inductance, this variation in the current causes a voltage ripple in the chip's main power supply rails [125]. If the ripple is above or below a certain tolerable range then there is high probability that the chip may even malfunction.

We have simulated both the positive and negative voltage variations (of about 100mV) on vdd and gnd rails. The results in Table 3.2 shows that the pulse width of the thermal sensor with all the supply voltage variations differ by about 800ps for 27°C and 50-80ps for 100°C. The capacitor shown in Fig. 3.6 takes longer or lesser time to charge depending on the voltage variations and hence noticeable difference at 27°C. But at 100°C the difference diminishes because of the exponential rise in the leakage currents. We have noticed that the number of clock cycles even in this case does not get affected by supply voltage variations.

Table 3.2: Pulse width of the thermal sensor with supply voltage variations at 27°C and 100°C as compared to the one with no voltage variations.

Temperature	Pulse width				
	vdd=1V; gnd=0V (ideal)	vdd=1.1V; gnd=0V	vdd=0.9V; gnd=0V	vdd=1V; gnd=-0.1V	vdd=1V; gnd=0.1V
At 27°C	8.12ns	8.961ns	7.325ns	8.956ns	7.338ns
At 100°C	20.95ns	21.03ns	20.9ns	21.01ns	20.91ns

3.5 Summary

A novel thermal self-timed sensing architecture has been presented, consisting of thermal sensor and its digital interface. This architecture has been simulated and verified in 65nm CMOS technology. Power supply noise, input signal noise and supply voltage variation analysis have been performed and it has been found that these does not have any effect on the accuracy of the sensing. A novel monitoring interconnection network based on self-timed signaling has been proposed which would serve as a foundation for further development of our work.

Chapter 4

Thermal Modeling and Analysis

As the technology scales down, power density increases which increases the on-chip temperature. This increase in on-chip temperature increases the cost of cooling solutions exponentially. Also, as many modern chips cannot simply be designed anymore for the worst case thermal profile, there arises a great need for thermal-aware design. Having a greater understanding of the techniques involved in thermal-aware design would help in controlling as well as reducing the thermal profile of the system. Temperature-aware run-time techniques help in regulating the operating temperature of the chip, thereby preventing thermal emergencies by tuning the processors run-time behaviour accordingly. In order to study and evaluate the efficacy of such techniques, requires a thermal model. Such a thermal model would aid in the analysis of architectural trade-offs and design-space explorations. In this chapter we would describe a) the thermal analysis of on-chip interconnects in multicore systems and b) Thermal modeling and analysis of 3D multicore systems in Flip-chip package systems.

1. Thermal analysis of on-chip interconnects in multicore systems: As the temperature increases, interconnect delay increases due to the linear increase in electrical resistivity. This degrades the performance and shortens the interconnects life time. Package reliability will also be severely affected by the resulting thermal hotspots, thus impacting the overall performance of multicore systems. We approach this challenge by proposing to use thermal management techniques with the help of architectural thermal model of a multicore system running on a network with interconnects spawning across it. In this regard we have analysed the spatial thermal profile of the global Cu nanowire for on-chip interconnects in 65nm CMOS technology from ST microelectronics. The impact of this temperature rise along the interconnects has

been analysed with two different signal transmission systems namely current-mode and voltage-mode signaling.

2. Thermal modeling of 3D multicore systems in a Flip-chip package: Three-dimensional (3D) technology offers greater device integration, reduced signal delay and reduced interconnect power. It also provides greater design flexibility by allowing heterogeneous integration. In this work, a 3D thermal model of a multicore system is developed to investigate the effects of hotspot, and placement of silicon die layers, on the thermal performance of a modern flip-chip package. In this regard, both the steady-state and transient heat transfer analysis has been performed on the 3D flip-chip package. Two different cases for the thermal model were evaluated under different operating conditions. The optimal placement solution is also provided based on the maximum temperature attained by the individual silicon dies. We have also provided the improvement that is required in the heat sink thermal resistance of a 3D system when compared to the single-die system.

4.1 Thermal analysis of on-chip interconnects in multicore systems

As technology scales down and power density increases, a lot of factors like power dissipation, leakage, data activity and electro-migration contribute to higher temperatures, larger temperature cycles and increased thermal gradients all of which impact multiple failure mechanisms [7]. This increase in temperature, increases interconnect delay due to the linear increase in electrical resistivity. These delay variations pose significant reliability problems with already dense interconnect structures. Joule self-heating, which is defined as the amount of heat generated when a maximum current of j_{max} passes through an interconnect wire, and delay variations combined with the introduction of low-k dielectrics with low thermal conductivity increases the need for accurate thermal analysis and estimation of interconnect temperature.

In a multicore scenario power scales up with the increase in the number of cores. With power, thermal challenges become abound which needs to be addressed urgently. The increase in the on-chip temperature not only degrades the performance of the chip but also its reliability will be called into question. Srinivasan et al. [119] have shown that the mean time to failure decreases with the increase in temperature. Hence, there is a great need to monitor the on-chip temperature accurately. Under or over estimation of the thermal profile of the chip leads to significant reliability issues.

Generally on-chip networks consume significant proportion of the total system power. MIT's 16 tile RAW processor's on-chip interconnection network consumes around 36% of total chip power, with each router dissipating 40% of the individual tile power [126]. So, the power consumed by on-chip interconnection networks is translated into heat which affects both the underlying silicon and metal layers. The interconnect temperature not only depends on the low-k dielectrics but also on the vias, which have much higher thermal conductivity and hence can serve as efficient heat dissipation paths [127]. In the following sections we have analysed the temperature rise on an interconnection link which incorporates the via effect and deduct several conclusions, some of which can be used to design circuits efficiently. As part of our desire to model a multicore system and deploy run-time thermal management techniques when ever a thermal emergency occurs, we have to model the thermal behavior of on-chip interconnection networks. In this regard we have proposed the thermal modeling of on-chip links in a multicore scenario.

As the number of cores increases thermal challenges increase, thereby degrading the performance and reliability of the system. In this work we assume that there are N Thermal Sensing Circuits (TSC) operating in each core which sense the temperature and convert it into known digital form. The request to perform the thermal sensing operation comes from the central Thermal Control Unit (TCU) and when the thermal sensing operation is performed the acknowledgement signal is sent. The data transfer between the TCU unit and the TSC takes place over the self-timed interconnection network.

We have discussed about self-timed thermal monitoring methodology in Chapter 3, which is based on the use of thermal sensors. Since leakage currents are sensitive to temperature and increase with scaling, we have proposed the use of a leakage current based thermal sensing for monitoring purposes. In this regard we have implemented a novel thermal sensing circuit in 65nm CMOS technology, which converts analog temperature information into digital form. We have also proposed a novel thermal sensing and monitoring interconnection network structure based on self-timed signaling, comprising of an encoder/transmitter and decoder/receiver. We have performed power supply noise, additive noise on sensor input signal and dynamic power supply voltage variation analysis on the thermal sensing circuit and shown that it is robust enough under different operating temperatures. This work of analysing the temperature profile of the interconnection network is a logical extension to our previous work.

We will be describing about the impact of temperature on the resistivity of the copper (Cu) nano wire in Section 4.1.1, introduce to the thermal model of interconnection link in Section 4.1.2 and analyse the impact of

temperature rise along the length of the interconnect with two different signal transmission mechanisms namely current-mode and voltage-mode in Section 4.1.3.

4.1.1 Resistivity vs Temperature

The electrical resistivity of Cu nano wires which is a quantitative measure of opposition to the flow of electrical current, increases with the increase in temperature. This dependence on temperature is usually explained with the help of a Bloch-Grüneisen formula [128] as follows.

$$\rho(T) = \rho(0) + \rho_{el-ph}(T),$$

$$\rho_{el-ph}(T) = \alpha_{el-ph} \left(\frac{T}{\Theta_R}\right)^n \int_0^{\Theta_R/T} \frac{x^n}{(e^x - 1)(1 - e^{-x})} dx \tag{4.1}$$

Where the temperature independent part $\rho(T)$ is the residual resistivity caused due to defect scattering, the temperature dependent part $\rho_{el-ph}(T)$ is the cause of electron-phonon interaction, n and α_{el-ph} are constants and Θ_R is the Debye temperature.

The electrical resistivity also increases with decreasing wire widths due to surface scattering and grain boundary scattering effects [129]. As the wire widths decreases from top level metal layer to the bottom level metal layer, increasing the electrical resistivity along the way and hence contributing to both the increase in the propagation delay time constant and temperature rise.

For a first order approximation, the effect of variation in ρ with temperature is found to be constant for normal operating conditions [127] and hence it is ignored in our calculations. Chen et al. [130] have reported that with the use of resistivity values at 100°C, the results represented would be commensurate with a chip operating at 100°C and the error in interconnect temperature prediction is under 4% for a case in which the chip substrate is at room temperature. Another source of error could be the use of bulk metal thermal conductivities instead of thin film thermal conductivities, as the values of conductivities depend on the thickness of the film. This is unavoidable due to the lack of any thin film values [130]. It has been reported that the error in the calculation of the interconnect temperature rise could be as high as 15% if bulk material conductivities are used [131].

4.1.2 Thermal Analysis of Links

In a typical surface mount packaging, like for example IBM's ceramic ball grid array package (CBGA), heat usually flows through the metal layers

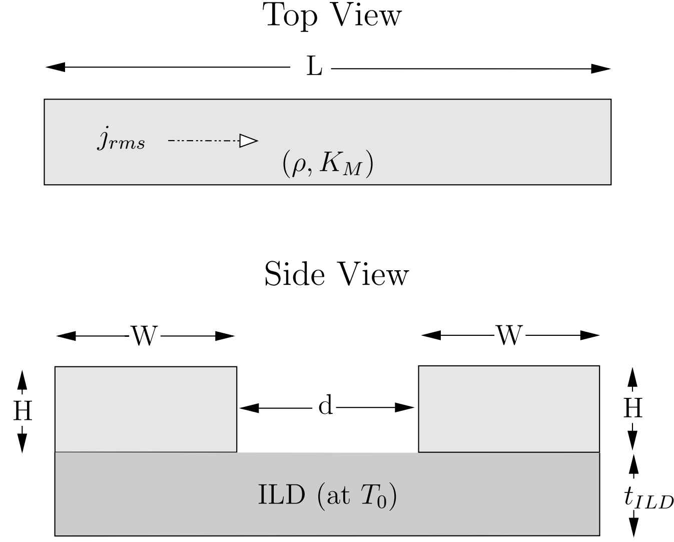

Figure 4.1: Conductor of length L.

to the heat sinks. The upper metal layers have long via separations when compared to the lower ones. Hence the temperature rise ΔT in those upper metal layers is much higher and is the main cause of concern from the thermal design perspective. So, we have confined our analysis in this chapter to those global interconnects.

Assuming, that a uniform root mean square current density of j_{rms} is flowing through a conductor of length L, width W and thickness H that has a resistivity of ρ and thermal conductivity of K_M and is separated from the underlying interlayer dielectric (ILD) of thickness t_{ILD} and thermal conductivity k_{ILD}. The link has two vias at both the ends and is connected to the underlying layer which is at a temperature of T_0. The temperature of the link is actually affected by other parallel and orthogonal metal conductors separated by a spacing of d. The top and side views of such a conductor is shown in Fig. 4.1 and the spatial temperature distribution along its length is given by the following equation [127]

$$T(x) = T_0 + \Delta T_{Max} \left(1 - \frac{cosh\left(\frac{x}{L_H}\right)}{sinh\left(\frac{L}{2L_H}\right)} \right) \tag{4.2}$$

$$\text{for } -\frac{L}{2} \leq x \leq \frac{L}{2}$$

where

$$\Delta T_{Max} = \frac{j_{rms}^2 \rho L_H^2}{k_M}$$

$$L_H = \left[\frac{K_M H t_{ILD}}{k_{ILD}} \left(\frac{1}{s} \right) \right]^{\frac{1}{2}}$$

$$and\ s = \left(\frac{w}{t_{ILD}} \left[\frac{1}{2} ln \left(\frac{w+d}{w} + \frac{t_{ILD} - \frac{d}{2}}{w+d} \right) \right] \right)^{-1}$$

Since the thermal conductivity of the vias is much higher when compared to the dielectrics, heat flows rapidly through the vias to the underlying layer. The thermal model [127] of the interconnect described in the above equation (2) incorporates the via effect and also takes the heat spreading factor (s) which is the one dimensional heat flow from the metal wire to the underlying layer into consideration.

Since, the hottest part of a typical global interconnect is the part where the via effect diminishes, we can deduct that the probability of hotspot lying in this area is higher. In the case where there are metal bus arrays crossing other metal bus arrays then the probability of hotspot lies at the intersection of those arrays.

4.1.3 Signal transmission methods

A multicore system running on a network is the most viable solution for on-chip communication that provides good scalability, which is achieved by keeping the length of the communication link constant and the signaling local between the routers. In this context higher throughput is achieved by using long-range high performance links between the routers and efficient signaling techniques to accompany them.

The signal transmission systems used in CMOS circuits can be broadly classified into two categories: voltage mode and current mode signaling. The main difference between the two transmission systems lies in the type of signal that is forced on to the interconnection link. Voltage mode uses voltage as the signal, whereas current mode uses current.

Voltage mode signaling

In the voltage mode, the voltage has to swing from rail to rail over the entire length of the wire, which leads to larger delays. This increase in delay is compensated by inserting repeaters at optimal locations [132] and splitting the long-range link into different segments. Each such segment is connected to the silicon through vias at either ends.

The length of a global interconnect link in a multicore system is around 2mm [133]. We have optimally divided the link into 5 different segments, each segment being 400μm in length. Fig. 4.2 shows the temperature rise along the segment of the global Cu nanowire with an average temperature

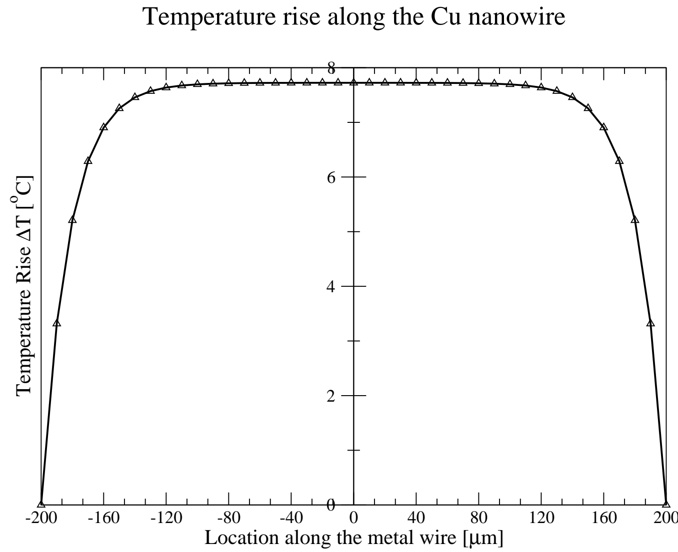

Figure 4.2: Spatial temperature profile along the Cu nanowires with 400μm via separation. The dimensions and other material properties of the global interconnect used are for 65nm technology node from ST microelectronics [5].

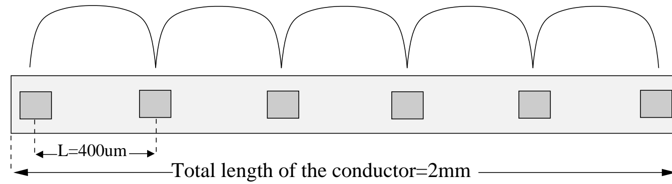

Figure 4.3: Temperature distribution along the total length of the conductor optimally divided into different segments and interspersed with vias.

rise of 6.85°C. The dimensions and other material properties used are for 65nm CMOS technology node from ST microelectronics [5]. Fig. 4.3 shows the temperature distribution along the total length of the global interconnect, optimally divided and interspersed with vias. In the case where the interconnection links form part of the parallel bus array, interleaving the repeaters as shown in the Fig. 4.4 will spread the heat flux uniformly and hence diminish the probability of a hotspot.

Current mode signaling

Current mode signaling on the other hand, when compared to voltage mode signaling, reduces the communication latency and gains high throughput without pipelining and/or using repeaters. This is achieved due to the low-

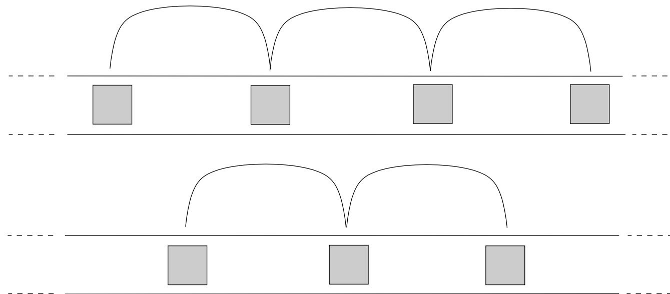

Figure 4.4: Interleaving of repeaters.

impedance termination at the receiver end which results in reduced signal swings without the need for separate voltage references. Also, this low-impedance termination shifts the dominant pole of the system, thus leading to smaller delays.

The temperature rise along the length of the global Cu nanowire without any repeaters/buffers in between is shown in Fig. 4.5. The average temperature rise is about 6.78°C using the current mode signal transmission system.

The power consumed by the on-chip interconnection network effects the temperature of both the metal layers and the silicon underneath. So, having low-power links in between the processing elements would decrease the temperature significantly.

4.1.4 Wide line vs narrow line

Usually, interconnects from one metal layer can be connected to interconnects from a different metal layer, by a group of vias as shown in the Fig. 4.6. In the case where a narrow line is connected to another narrow line or a wide line (Fig. 4.6(c) and Fig. 4.6(a) respectively) only a single via/contact actually fits along the width. But, two or more vias can be used to connect them in accordance with the design rules. In the case where a wide line is connected to another wide line as shown in Fig. 4.6(b), the maximum number of vias allowed along the width are used. Since, the equation for the thermal model (i.e., eq.(2)) has been derived with a single via at each of the ends of the conductor, we need to reevaluate it in light of fact that there could be multiple vias at either ends. This can be done by replacing those multiple vias with a single effective via and reevaluating the equation for the thermal model.

4.1.5 Summary

We have analysed the spatial temperature distribution on a global interconnect link in 65nm CMOS technology from ST microelectronics. It has been

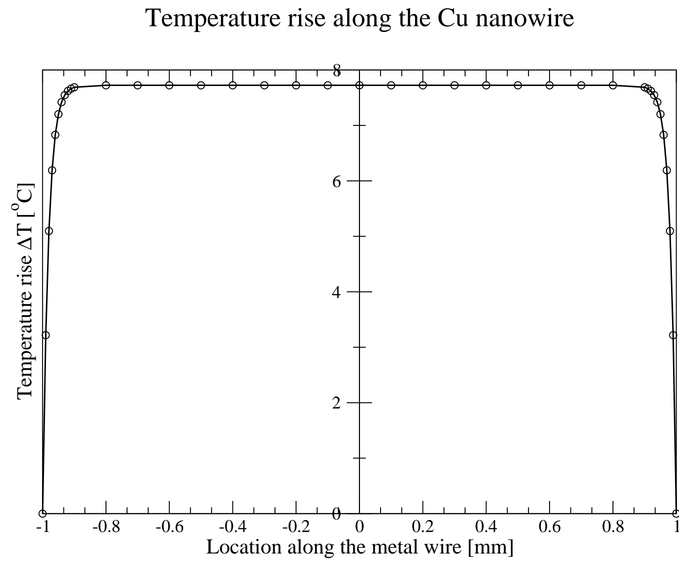

Figure 4.5: Spatial temperature profile along the Cu nanowires with $2mm$ via separation. The dimensions and other material properties of the global interconnect used are for 65nm technology node from ST microelectronics [5].

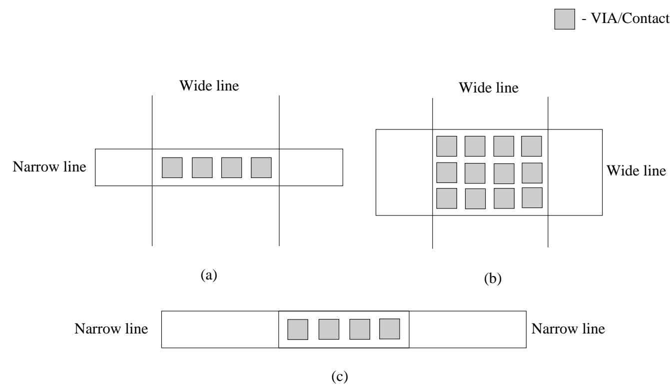

Figure 4.6: Connection between the interconnect segments with a group of vias.

found that the average temperature rise ΔT along the length of the conductor is around 6.8°C for a global interconnection link. The impact of this temperature rise has been analysed for both the voltage mode and current mode signaling.

4.2 Thermal modeling and analysis of 3D stacked systems

As technology scales down and power density increases, a lot of factors like power dissipation, leakage, data activity and electro-migration contribute to higher temperatures, larger temperature cycles and increased thermal gradients all of which impact multiple failure mechanisms [7]. This increase in temperature, increases interconnect delay due to the linear increase in electrical resistivity. These delay variations pose significant reliability problems with already dense interconnect structures. In order to overcome the problems associated with the interconnects and the limits posed by the traditional CMOS scaling, three-dimensional (3D) integrated circuits has been proposed. 3D integrated circuits take advantage of dimensional scaling approach and are seen as a natural progression towards future large and complex systems. They increase device density, bandwidth and speed. But on the other hand, due to increased integration, the amount of heat per unit footprint increases, resulting in higher on-chip temperatures and thereby degrading the performance and reliability of the system. In this case, heat sinks need to be very efficient in transferring the internally generated heat to the ambient. Although there is a dearth of design and layout tools for 3D technology, there is a significant amount of effort going on in that direction.

The ever expanding market for consumer electronics is driving innovation in packaging technology leading to newer packages which are smaller, more thermally efficient and cost effective at the same time. The technology related to wafer level packaging and 3D integration has recently outpaced ITRS roadmap forecasts [7]. One of the fastest growing packaging architectures is the wafer level packaging (WLP). It offers lower cost, improved electrical performance, lower power requirements and smaller size. Although several architectural variations are available, in this chapter we will be discussing only the flip-chip packaging. The ITRS report projects that the power density for 14nm technology node will be greater than 100 W/cm^2 and the junction-to-ambient thermal resistance will be less than 0.2°C. It is very important to keep the thermal resistance at bay as this may increase the package cost and the overall cost of the product.

Guoping et al. [9] [10] have done thermal modelling of multicore systems and have investigated the effects of CPU power level, local hotspot power density, hotspot location and hotspot size on its thermal performance. But

they stopped short of extending their work to 3D multicore systems. In the work depicted by Ankur et al., [134], they have proposed an analytical and numerical modelling of the thermal performance of three-Dimensional Circuits. In the following sections, we have chosen to model a 3D multi-core system in a modern flip-chip package which is used mostly for high-performance processors. We have started our study with thermal modelling of a multicore processor and have investigated the effects of hotspots and their locations on the thermal performance of the package. We then perform thermal modeling and analysis of 3D multicore systems in a Flip-Chip package.

Although some work has been done in the past regarding the 3D stacked IC's a comprehensive treatment of the subject was missing. Banerjee et al. [25] [135] have performed thermal analysis of 3D IC's using analytical modelling and numerical simulations. They did not simulate the 3-D IC's with TSV's. Instead they compared two alternative 3-D IC's with wafer bonding technologies. They are a) 3D IC's fabricated by wafer bonding using polymer adhesives and b) 3D IC's fabricated with wafer bonding using a thermocompression method. In this chapter we performed numerical simulations while considering the presence of TSV's between the dies. Thermal models are used to understand the limits of thermal feasibility of 3D stacked systems.

4.2.1 Nomenclature

h_{eff}	= Effective heat transfer coefficient of the heat sink base (W/m^2K)
K	= Thermal Conductivity (W/mK)
T_A	= Ambient Temperature (°C)
T_J	= Junction Temperature (°C)
R_{JA}	= Junction-to-Ambient thermal resistance (°C/W)
Q	= Power dissipation that produced the change in the junction temperature (W)

4.2.2 Flip-Chip package

Although IBM's Ball Grid Array packages have been in use since the 1970's, recent advances in packaging technology have lead to Flip-Chip Ball Grid Array (FCBGA) packages being extensively used. FCBGA allows for much higher pin count than the other package types by distributing the input-output signals through the entire die rather than being confined to the chip periphery. In an FCBGA the die is mounted upside-down (flipped) and connects to the package balls (lead-free solder bumps) via a package substrate.

The cross-sectional view of a modern 3D flip-chip package is shown in the Fig. 4.7 whose primary consideration will be its ability to transfer heat

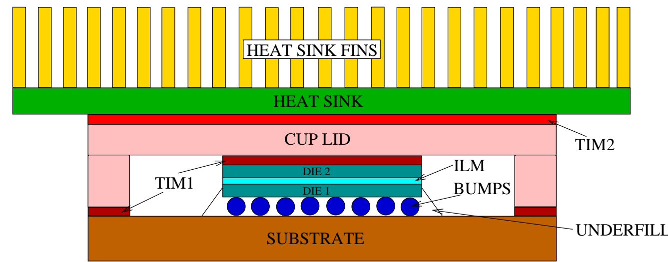

Figure 4.7: Cross-Sectional view of a modern 3D Flip-Chip package with 2 stacked dies.

from the silicon die to the ambient. Unlike the traditional wire-bonding technology, the electrical connection of a face-down (or flipped) integrated circuit onto the substrate is done with the help of conductive bumps on the chip bond pads. The conductive bumps are initially deposited on the top-side of the die during the fabrication process. It is then flipped over so that its top side faces down, and aligned with the matching pads on the substrate. The solder is then flown to complete the interconnection. The advantages of flip-chip interconnect include reduced signal inductance, power/ground inductance, and package footprint, along with higher signal density [17].

4.2.3 Thermal modelling and Analysis

The high operating temperature of a semiconductor device, caused by the combination of device power density and ambient conditions is an important reliability concern. Instantaneous high temperature rises in the devices can possibly cause catastrophic failure, as well as long-term degradation in the chip and package materials, both of which may eventually lead to system failure [17]. Most modern flip-chip devices are designed to operate reliably with a junction temperature falling under a certain range. To ensure that the package can perform well thermally under this range a thermal model is simulated and tested. This thermal model can then be used to gauge the reliability of the package. This shortens the package development time and also provides an important analytical tool to evaluate its performance under different operating conditions.

Table 4.1: Modelling parameters [7] [8] [9] [10].

MODEL CONFIGURATION	PARAMETERS	INPUT DATA
Boundary condition	T_{Amb} (°C)	25
	$h_{eff}(W/m^2K)$	840
Heat Sink Base [9]	Size (mm)	100x100
	t_{base} (mm)	5
TIM2	t_{TIM2} (mm)	0.1
	k_{TIM2} (W/mK)	3
Cup Lid (heat spreader)	Size (mm)	50x50
	t_{Lid} (mm)	2
	k_{Lid} (W/mK)	600
TIM1	t_{TIM1} (mm)	0.1
	k_{TIM1} (W/mK)	8
Silicon Die 1 and 2	Size (mm)	20x20
	t_{Die} (mm)	0.6
	k_{Die} (W/mK)	90
Interlayer Material	t_{ILM} (mm)	0.02
	k_{ILM} (W/mK)	4
Lead bumps and Underfill	k_{UF} (W/mK)	1
	t_{UF} (mm)	0.65
Substrate	Size (mm)	50x50
	t_{Sub} (mm)	1.44
	k_{Sub} (W/mK)	17
Boundary condition	$h_{Sub}(W/m^2K)$	10

We have developed a thermal model of the modern flip-chip package using a commercial tool called COMSOL. It is a finite element based multiphysics modelling and simulation software. Our simulations are based on

the heat transfer module of COMSOL multiphysics package. The size of the silicon die 1 and 2 is 20 mm × 20 mm × 0.6 mm which is being mounted on to the substrate of size 50 mm × 50 mm × 1.44 mm. The layers of silicon die are separated by an interlayer material whose thickness is around 0.02 mm. The cup lid which acts as the heat spreader and whose thermal conductivity is very high is placed on top of the silicon die. The thermal interface material (TIM1) which is some sort of a thermal grease and has very good adhesive properties is being used as the filler material in between the heat spreader and the silicon die. The heat sink base of size 100 mm × 100 mm × 5 mm is being used. A vapour chamber is used as the heat sink base and the detailed assumptions can be found in [10].

Instead of including the heat sink fins in our computational model, we have used an effective heat transfer coefficient (h_{eff}) as a boundary condition on the heat sink [9]. Other assumptions related to the geometry of the package and its components, material properties (like thermal conductivity, density and specific heat capacity) and the boundary conditions are taken from the literature [7] [8] [9] [10]. Some important model configuration parameters are represented in the tabular format as shown in Table 1. The parameter Q, which is the heat generated per unit volume is applied to the silicon die. The boundary condition for the substrate layer is assumed to be convective and the sides of the package are assumed to be adiabatic.

Modelling interlayer material

Three effective thermal conductivities are used for the lead solder bumps/-underfill layer, substrate layer and the interlayer material (ILM) respectively. The interlayer material in between the silicon dies is modelled as a homogeneous layer in our thermal model. We assumed a uniform through-silicon-via (TSV) distribution on the die and obtained the effective interlayer material resistivity based on the TSV density (d_{TSV}) values [8], where d_{TSV} is the ratio of total TSV's area overhead to the total layer area. Coskun et al. [8] have observed that even when the TSV density reaches 1-2%, the temperature profile of the silicon die is only limited by a few degrees, thus justifying the use of homogeneous TSV density in our thermal model. According to the current TSV technology [136], the diameter of each via is 10μm, and the spacing required around the TSV's is assumed to be around 10μm [8]. For our simulations we have assumed around 8 via's/mm^2, that is around 3200 vias spread across the 400 mm^2 area of the silicon die. Hence the TSV density is around 0.062% and the resistivity of the interlayer material is around 0.249 mK/W (i.e. thermal conductivity = 4.016 W/mK) [8].

Junction temperature and thermal resistance for a 3D system

The two most important thermal parameters for any semiconductor device are the junction temperature (T_J) and thermal resistance (R_{JX}). The junction temperature is usually the highest temperature on a silicon die, whereas the thermal resistance is quantified as the rate of heat transfer between two layers in a package. The junction-to-ambient thermal resistance (R_{JA}) which is a measure to evaluate the thermal performance of a flip-chip package is determined from equation (4.3).

$$R_{JA} = \frac{T_J - T_A}{Q} \tag{4.3}$$

The single-valued junction-to-ambient thermal resistance which has been used traditionally to describe the thermal characteristics of a silicon die is not sufficient enough to describe the thermal performance of a 3D system, due to the presence of multiple heat sources and multiple thermal resistances. Hence, Ankur et al. [134] have suggested a matrix representation for the junction-to-ambient thermal resistance. In this regard R_{ij} represents the temperature rise in the ith layer per unit heat dissipation in the jth layer. This is represented in the equation (4.4).

$$R_{ij} = \frac{\theta_i}{Q_j} \tag{4.4}$$

Where, θ_i is the temperature rise above ambient of the ith node and Q_j is the heat generated at the jth node. The equation (4.4) can be rewritten as follows.

$$R_{ij} = \frac{T_i - T_A}{Q_j} \tag{4.5}$$

Where, T_i is the junction temperature of the ith layer. So, for a simple two-die stack, where one layer is the processing layer (denoted by subscript 'p') and the other a memory layer (denoted by subscript 'm'), we have 4 different thermal resistance values namely $R_{pp},\ R_{pm},\ R_{mp}\ and\ R_{mm}$ and the junction-to-ambient thermal resistance can be represented as shown below.

$$R_{JA} = \begin{bmatrix} R_{pp} & R_{pm} \\ R_{mp} & R_{mm} \end{bmatrix}$$

4.2.4 Simulation results

We have built a generic two-die stack in a flip-chip package using COMSOL. The layer where the hotspot is generated is considered as a processing die and the other layer is considered as the memory die in our simulations. In the first instance (model-I) the processing die is placed near the substrate,

and the memory die is placed next to the heat spreader and the heat sink. In the second instance (model-II) the memory die is placed near the substrate and the processing die is placed near the heat spreader and sink. We have assumed that the total power consumed by both the processing layer and the memory layer is 100 W. Guoping Xu [9] has varied the size of the hotspot from 0.5 mm to 2 mm in his work related to the thermal modelling of multicore systems. In our work the power density of the hotspot which is being generated at the center of the multicore processing layer is fixed at 100 W/cm^2 and the dimensions are fixed at 1mm $\times$ 1mm $\times$ 0.6mm. We have performed both the steady-state and transient heat transfer analysis on the flip-chip package.

Steady-state heat transfer analysis

In the steady-state the heat generated by the memory and the processing layer is equal to the heat leaving the flip-chip package. During the measurements we have assumed that the power is gradually applied to the chip until the chip has reached the maximum working temperature (i.e. steady state). We have then measured thermal resistance which is the reluctance of the die to transfer heat when it reaches steady state. Fig. 4.8 and 4.9 show different thermal resistance plots for the dies in both the models. They are plotted against the memory power dissipated (as a percentage of processing power dissipation). It can be clearly seen that the overall thermal performance of model-II is much better than that of model-I. It can also be noted that, when both the layers consume equal amount of power then there is not much difference in the thermal resistance values in both the models. That is, the stacking order of the silicon dies does not influence the thermal resistance values.

When both the layers are consuming equal amount of power, then in model-II, it can be noted that there is no difference in the thermal resistance values of the processing and memory layers even though a hotspot is present in the processing layer. This shows that the heat sink is efficient in removing the heat generated by the hotspot, thereby maintaining constant thermal resistance values.

Fig. 4.10 shows the maximum temperature attained on the processing and the memory die for both models at steady state. The maximum temperature is plotted against the memory power dissipation (as a percentage of processing power dissipation). In the case where the memory die consumes around 10% of the processing die power, it can be observed that the difference in the maximum temperature of memory and processing die layers is around 4°C for model-I and 0.3°C for model-II. This goes on to say that the model-II is the optimized one which places the most heat generating layer, i.e. the processing layer near the heat sink for efficient heat transfer to the

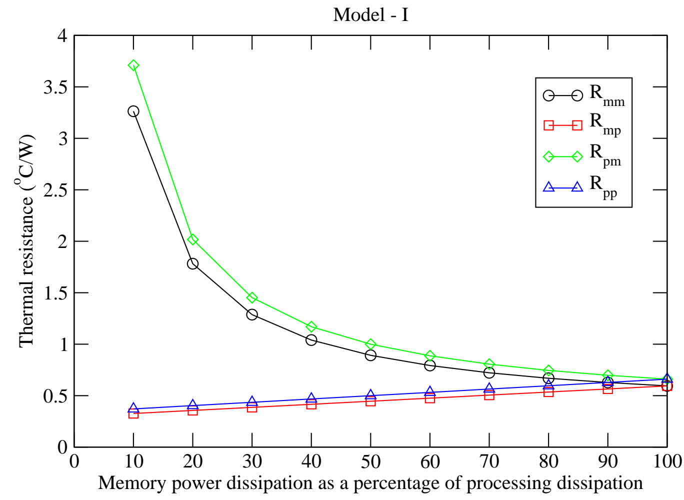

Figure 4.8: Thermal resistance measurements for both the dies in model-I at steady-state.

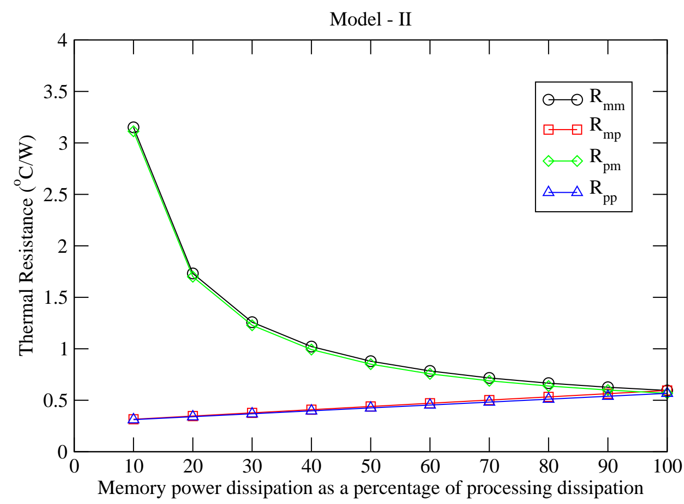

Figure 4.9: Thermal resistance measurements for both the dies in model-II at steady-state.

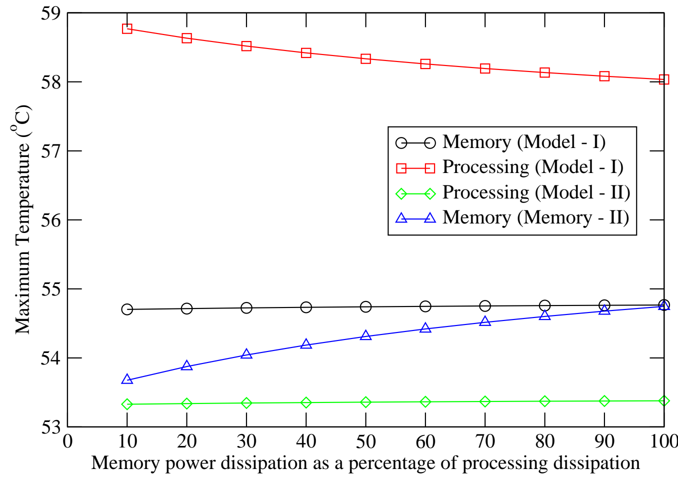

Figure 4.10: Maximum temperature on the processing and memory die for both models.
ambient.

Transient heat transfer analysis

The dalliance in reaching the steady state is measured in transient analysis, wherein the temperature responses are continually recorded within a short time interval for the given power consumption of the silicon dies. Transient analysis is necessary to observe the steady-state behaviour and also the thermal profile of different configurations that might change over time as the maximum temperature is reached.

Fig. 4.11 and Fig. 4.12 shows the maximum temperature and the thermal resistance curves plotted against time for both the models when the memory layer is consuming around 10% of the processing power consumption. It can be seen from those curves that the heat sinks of the two models are efficient enough to take the heat out of the system irrespective of the placement of the processing die. By the time steady-state is reached the processing cores of model-I is 6.5°C hotter than model-II. It can also be noted that the thermal resistance of the memory die (R_{mm}) in Model-I is lower by 0.45°C/W when compared to model-II, whereas the thermal resistance of the processing die (R_{pp}) in both the models is almost the same.

In order to find out the improvement that is required in the heat sink thermal resistance for a 3D system when compared to the single die system, a transient percentage reduction plot of the heat sink thermal resistance (R_{hs}) has been plotted as shown in Fig. 4.13. The single die package system whose power consumption is 100 W, and has a hotspot of 100 W/cm^2 power density at the center of the silicon die has been used for comparison purposes. The curves have been plotted for both the models and for different power

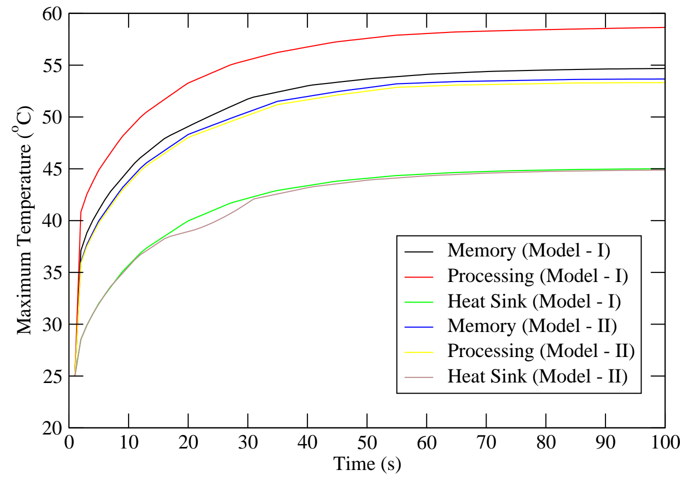

Figure 4.11: 10% Maximum temperature on the processing and memory die for both models.

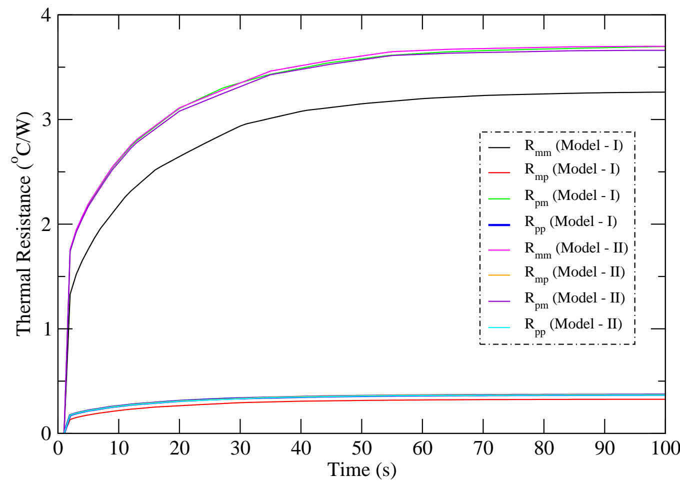

Figure 4.12: 10% Thermal Resistance on the processing and memory die for both models.

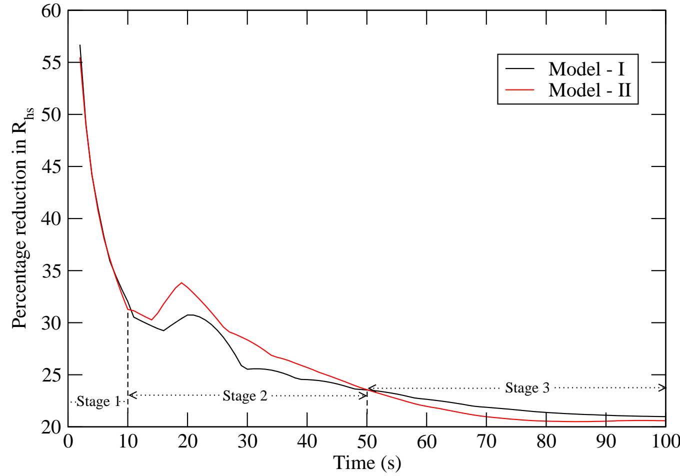

Figure 4.13: Improvement required in heat sink thermal resistance for a 3D system (both models) whose memory layer is consuming 50% of the processing die power. It has been compared with a single die package system.

consumption's of processing and memory layers. All those plots have showed some similarities in nature and hence could be easily segmented into three distinct durations or stages. In this chapter we have presented only one plot (Fig. 4.13) where in the memory die is consuming around 50% of the processing die power.

In the first stage the percentage reductions in R_{hs} is approximately the same for both the models, suggesting that the heat sink behaves identically for both the models for short durations of time.

In the second stage, when the maximum temperature on the heat sink starts to increase before attaining steady-state, model-I demands less reduction in the heat sink thermal resistance. This is due to the fact that the heat could not be transferred from the processing layer below the ILM to the heat sink. If the configuration of model-I tends to work in this stage, then instead of improving the heat sink one should concentrate on improving the effective thermal conductivity of the ILM layer.

In the third stage when both the models are attaining steady-state, they exhibit expected behaviour, as the configuration with the processor layer near the heat sink (model-II) behaves more efficiently. This is because the required reduction in thermal resistance is less. This plot not only shows the dependence on the stacking sequence but also shows that the observations should not be made strictly on the basis of the steady-state [134] analysis, as in some cases the chips might not reach steady state due to various dynamic thermal management techniques that are employed.

4.2.5 Summary

A thermal model of a 3D multicore system in a modern flip-chip package is developed in order to investigate the effects of hotspot, and placement of silicon die layers, on the thermal performance of a multicore system. We have used a finite-element based method to run our simulations. Both the steady-state and transient heat transfer analysis has been performed on the 3D flip-chip thermal model we built. Two different cases for the thermal model were evaluated under different operating conditions. We have found that in steady-state for the case where the memory layer dissipates around 10% of the power consumed by the processing core, an overall improvement of 0.6°C/W is obtained in the thermal resistance by placing the silicon layers optimally. For the same case, it has been observed that the difference in the maximum temperature of memory and processing die layers is around 4°C for model-I and 0.3°C for model-II. An improvement that is required in the heat sink thermal resistance for a 3D system when compared to a single-die system has been quantified.

Chapter 5

Thermally Efficient Inter-Layer Communication Scheme

The primary design goal of a high-performance system is the maximization of performance within the given power and thermal envelopes. The wire-length reductions in 3D stacked systems directly translate into both the power and performance improvements. Despite decreasing the power and latency of the system, 3D technology exacerbates thermal problems due to increase in power density.

In this chapter, we propose a thermally efficient routing strategy for 3D NoC-Bus Hybrid architectures, which mitigates on-chip temperatures by conducting most of the switching activity closer to the heat sink. Our simulations with a real world benchmark show that there has been a significant decrease in the peak temperatures when compared to a typical stacked mesh 3D NoC.

5.1 Introduction to Hybrid NoC bus 3D architecture

One of the popular 2D NoC architectures is the 2D Mesh. It consists of an interconnecting network of m×n switches connecting various IP blocks. A logical extension to this popular planer structure is the 3D Symmetric NoC which can be obtained by adding two additional physical ports to each router; one for Up and the other for Down [137]. Despite its simplicity, this architecture has two inherent problems. Firstly, it does not exploit the beneficial attribute in 3D chips which is negligible inter-wafer distance, because in this architecture both the inter-layer and intra-layer hops are

almost indistinguishable. Secondly, a considerably larger crossbar is required as a result of the two extra ports [138].

The stacked (Hybrid NoC-Bus) mesh architecture which is presented in [139] is a hybrid between the packet switched network and the bus architecture. It overcomes several 3D Symmetric NoC challenges by taking advantage of the short inter-layer distances (around 20μm) in 3D stacked systems [26]. It integrates the multiple layers of 2D mesh networks by connecting them with a bus spanning the entire vertical distance of the chip. As the inter-layer distance for 3D ICs is small, the bus length will also be smaller; approximately around $(n\text{-}1)*20\mu m$, where n is the number of layers. This makes the bus suitable for inter-layer communication in vertical direction. A six-port router is required instead of a seven port one for a typical 3D stacked Hybrid NoC-Bus architecture. Also, vertical communication is just one hop away to any destination layer. The dynamic Time-Division Multiple Access (dTDMA) bus [139] was used as a communication pillar. Due to one hop vertical communication and usage of router with one less port, this architecture is efficient in terms of both the power consumption and latency.

In [38] we proposed an efficient inter-layer communication scheme and routing algorithm which enables congestion-aware communication and improves the average packet latency (APL), power consumption and fault tolerance. We have further hybridized the proposed adaptive routing algorithm with available algorithms in order to mitigate the thermal issues by conducting the majority of the switching activities closer to the heat sink. The later part is described in this chapter.

5.2 Thermally efficient routing strategy for 3D NoC

In a stacked mesh 3D architecture, the thermal coupling of vertically aligned tiles is larger than the horizontally aligned tiles [136]. This is because the thickness of the silicon dies is much smaller than the lateral dimensions and hence the lateral heat flow is usually lower than the vertical heat flow. Also, having interface materials with lower thermal conductivities does contribute to this issue. The thermal impact of on-chip 3D NoCs are governed by various non-design issues like the ambient temperature, cooling solutions and the package solutions. In this chapter we assume that the size of the heat sink is fixed, the ambient temperature around the chip is constant and the velocity of air-flow is set [35]. We also assume that the application mapping is fixed and just focus on the routing based approach.

In a typical stacked 3D NoC, the maximum thermal conduction usually takes place from the die which is closer to the heat sink. The die closer to the heat sink also has lower junction temperature and thermal resistance.

In [140] [38], we proposed a congestion-aware routing algorithm called *AdaptiveZ* for vertical communication. In *AdaptiveZ* routing algorithm, the first bus pillar available on the way for the vertical communication is used. In this section, we hybridize the *AdaptiveZ* routing with other available algorithms to mitigate the thermal issues by herding most of the switching activities closer to the heat sink.

Definition 1: *LastZ - A 3D routing algorithm is LastZ-based if the intra-layer routing process is completed before the inter-layer routing. In other words, in a LastZ-based routing algorithm, when a node N_{source} sends a flit to a node $N_{destination}$, the flit will first travel along the X or Y direction (statically or adaptively) in N_{source} dimension until $Flit_{xy}$=$Pillar_{xy}$, then it will traverse the last hop in the Z direction.*

Assuming that the heat sink is at the top, the proposed routing algorithm shown in the Fig. 5.1 is described as follows:

1. AdaptiveZ routing: When the current node is located at the bottommost layer (farthest from heat sink), the packet that needs to be sent, first traverses adaptively upwards along the Z direction. It is then routed using a 2D routing algorithm (e.g. XY, YX).

2. A *LastZ*-based routing algorithm [141] (as defined in Definition 1): When the current node is located at the top layer (closest to the heat sink), the packet that needs to be sent, first traverses in the current layer using a 2D routing algorithm (e.g. XY, YX) and then moves downwards along the Z direction.

3. Hybrid routing: When the current node is other than the top or bottom layer, then depending on the location of the destination node relative to the current node, the routing is performed. That is, if the destination node is below the current node then a *LastZ*-based routing algorithm such as static XYZ or YXZ routing is performed. If the destination is above the current node then *AdaptiveZ* routing is performed.

Our thermally efficient routing algorithm is described in Algorithm 1. Since our algorithm is adaptive, it takes care of possible congestion that might arise due to excessive routing of packets onto the layer closer to the heat sink. Algorithm 1 is also a distributed routing algorithm and the routing decision is made at each router for every hop. So, the output shows the next hop which can be any one of the possible output ports (East, West, North, South, UP/Down). The proposed routing algorithm also offers negligible area overhead being at the same time thermally efficient. It is noteworthy that the routers located in the topmost and the bottom layers do not need the hybrid routing, hence they do not adversely affect the

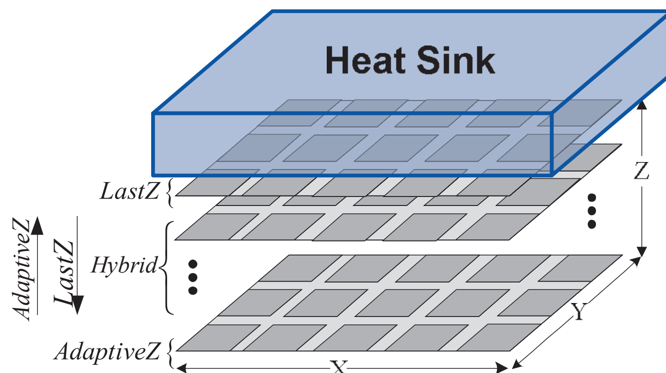

Figure 5.1: The proposed thermally efficient routing algorithm

area. It should be noted that although the communication is adaptive, it is deadlock free because of the usage of the available virtual channels.

In [75], Chao *et al.* proposed a traffic- and thermal-aware run-time thermal management scheme using a proactive upward routing to ensure thermal safety. Although their technique has potential to enhance the runtime thermal safety, there are some important drawbacks. Firstly, to migrate the communication power towards heatsink, they use a non-minimal path routing. They showed that, even if source and destination of a packet are located in adjacent layers, it may take too many vertical hops for the packet to reach the destination. But the fact is, non-minimal path routing naturally increases the zero load latency and has power overhead. Despite driving power of a vertical transfer is small, intermediate large 3D routers consume a considerable part of power budget. Secondly, the prediction-based routing algorithm imposes a large area overhead and extra TSVs because of required logic for traffic estimation, decision logic, information passing through layers, etc. Further, the window-based prediction mechanisms potentially have inefficiency due to the presence of a probability of misprediction. Our adaptive minimal routing mechanism which benefits from one-hop bus-based vertical communication, overcomes these issues using run-time congestion checking before sending a packet to layers closer to the heatsink.

5.3 Thermal model to evaluate the thermally efficient routing strategy for a 3D NoC

We have built a thermal model of a 3×3×3 NoC using HotSpot v.5.0 [100]. We have exploited Hotspot's grid model which is capable of modeling stacked 3D chips for our thermal simulations. We obtained the power trace file from our in-house cycle accurate NoC simulator which was implemented in HDL.

Algorithm 1 *Thermally Efficient Routing Algorithm*

Input: $(X_{current}, Y_{current}, Z_{current})$, $(X_{destination}, Y_{destination}, Z_{destination})$
Output: Next Hop (E, W, N, S, L, U/D)

1: **if** $(Z_{current} = Z_{destination})$\{*The current and destination nodes are located in the same layer*\} **then**
2: Use a 2D intra-layer routing algorithm;
3: **else if** $(Z_{current} > Z_{destination})$ \{*The destination node is below the current node and farther from the heat sink*\} **then**
4: Use a *LastZ*-based routing algorithm;
5: **else** \{*The destination node is above the current node and closer to the heat sink*\}
6: Use *AdaptiveZ* routing algorithm;
7: **end if**

We purposefully chose a smaller system so that our real time application can be easily mapped on to it.

Intel's 80-tile teraflops chip running the stencil kernel code (which solves for steady-state 2-D heat diffusion equation with periodic boundary conditions on left and right boundaries of a rectilinear grid, and prescribed temperature on top and bottom boundaries) gives an average performance of 1.0 TFLOPS at 4.27 GHz and 1.07 V supply with total chip power dissipation of 97 W. Also, the total power dissipation increases to 230 W at 1.35 V and 5.67 GHz operation, delivering 1.33 TFLOPS of average performance. At 4.27 GHz, measurements performed by Intel indicate that approximately 358K floating point operations take place achieving an overall peak performance of 73.3%. Intel also provides an estimated power breakdown at the tile and router levels, which is simulated at 4 GHz, 1.2 V supply and at 110°C [133].

Like [75], the tile geometry and power model has been adopted from Intel's 65nm based 80-core processor [6] [133]. The following assumptions regarding our power model are based on literature [75]. We have assumed that the power and temperature measurements in [6] and in [133] were obtained at a certain operating condition (i.e., at a certain traffic load/packet injection rate). We use the packet injection rate which doubles the performance metric, zero-load latency (the latency of the network when only one packet traverses through it) to calibrate our power model so that it matches approximately the power values measured by Intel [6]. The power of the router is modelled as a linear increasing function of traffic load. Whereas, the power of the processing element and the power of the local memory are linear function of the power of the router. It should be noted that for the sake of simplicity we do not model the power as a function of tempera-

ture which could lead to some underestimation of temperature profile when the temperature is high enough. The cumulative power and energy of each router during an interval is calculated by counting the number of operations in the router [75].

We have used realistic traffic patterns for our thermal analysis. For this an encoding part of video conference application with sub-applications of H.264 encoder, MP3 encoder and OFDM transmitter was used [142]. The video stream used for simulation purposes was 300×225 pixels in size with each pixel consisting of up to 24 bits. Thus, each video frame is composed of 1.62 Mbits and can be broken down into 8400 data packets with each data packet consisting of 7 flits (which includes the header flit as well). The data width is set to 64 bits. The application graph with 26 nodes is shown in Fig. 5.2. In this application, the Mem_In_Video component generates 8400 packets for one application cycle equivalent to one video frame. The frame rate for the video stream was 30 frames/second and the data rate for the video stream was 49336 kbps. We have modelled the application graph, mapping strategy, frame rate, buffer size, number of nodes, layers and generated packets, supply-voltage and clock frequency for the simulation of this application.

The application graph consists of processes and data flows; data is, however, organized in packets. Processes transform input data packets into output ones, whereas packet flows carry data from one process to another. A transaction represents the sending of one data packet by one source process to another, target process, or towards the system output. A packet flow is a tuple of two values (P, T). The first value 'P' represents the number of successive, same size transactions emitted by the same source, towards the same destination. The second value 'T' is a relative ordering number among the (packet) flows in one given system. For simulation purposes, all possible software procedures are already mapped within the hardware devices.The video conference application which is mapped onto a 3×3×3 3D-mesh NoC is shown in Fig. 5.3. This mapping is based on the mapping technique described in [143]. The central node (1, 1, 1) was used as a platform agent for monitoring purposes.

The sizes of the silicon die's 1, 2 and 3 are 4.5 mm × 6.0 mm × 0.15 mm. The convection capacitance and convection resistance of the heat sink are 140.4 J/K and 0.1 K/W respectively. We have modeled the interlayer material as described in subsection 4.2.3. For that, we have assumed around 8 via's/mm^2, that is, around 216 vias spread across the 27 mm^2 area of the silicon die. Hence the TSV density is around 0.062% and the resistivity of the interlayer material is 0.249 mK/W (i.e. thermal conductivity = 4.016 W/mK) [8]. We have used an interlayer material (ILM) whose thickness is 0.02 mm. Other parameters are left unchanged from Hotspot's configuration file.

5.4 Simulation results and analysis

5.4.1 Thermally efficient routing for 3D NoC

We study the impact of the proposed thermal-aware hybrid routing algorithm on the chip temperature of a 3×3×3 NoC-based system using the thermal model presented in subsection 6.4.2. To this end, we have imported the physical floorplan and the obtained power trace file to the thermal simulator, and estimate the temperature profile for each layer.

The results of the thermal simulations on normal Hybrid Bus-NoC 3D Mesh-based and the proposed Hybrid Bus-NoC 3D Mesh-based (hybrid routing) systems running the video conference encoding application (Fig. 5.2) are shown in Table 5.2 and Table 5.3, respectively. In these tables, we show the steady state minimum and peak temperatures of each layer. Layer 0 is considered to be the one which is farther from the heatsink in our thermal model. The comparison between these tables shows the effectiveness of temperature optimization of the proposed thermal-aware hybrid routing. As expected, moving from the traditional 3D NoC to the proposed 3D NoC causes the peak temperature of the chip to decrease. The significant importance of the proposed hybrid routing algorithm is the mitigation of hotspots. Hotspots can noticeably exacerbate performance and reduce the lifetime of the chip. The figures given in Table 5.2 and Table 5.3 show that the proposed technique improves the peak chip temperature with negligible performance degradation. The improvement is up to 4°C for the realistic application. For our system with not much interlayer communication, the reduction of 4°C is quite an achievement. Assuming that the mapping of task is predefined and the computation power cannot be migrated, the proposed hybrid routing offers a significant peak temperature improvement by only migrating the communication power.

Modern streaming applications, like MPEG, are one of the most popular applications for use on embedded systems. They are both computation and communication intensive, thereby emphasizing the need for thermally efficient routing and mapping strategies. Accordingly, as shown for an MPEG application, we achieved the temperature reduction of up to 4°C. The tasks of the MPEG application, which has been curated from the ground up to work on our 3D NoC platform, is mapped in such a way that most of the processing happens in the die closer to the heat sink. That is, we have minimized the interlayer communication to reduce the load on the bus because of the limited bandwidth it offers and placing the cores which communicate heavily with each other on the same die and closer to the heat sink. Other benchmarks like the EEMBC benchmark suite [144] can also be used, provided that they can be modelled appropriately into an application graph, can be properly allocated to the processing nodes and have enough compu-

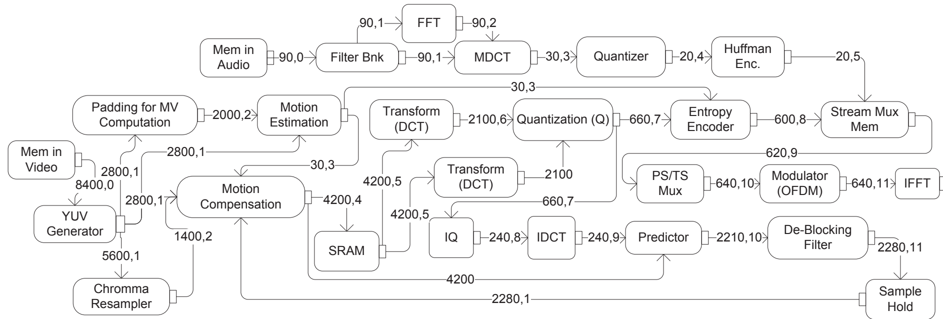

Figure 5.2: Communication trace of encoder part of a H.264 video conference

tation and communication information. Also, most of the available parallel benchmarks provide only communication centric information whereas our system requires both computation and communication information. The processing elements and other subsystems can make a significant difference in the evaluated performance of each benchmark, thereby making it hard to compare them without porting applications to our system. The future work would include porting other benchmarks to our platform and using them to evaluate and compare our thermally efficient routing strategy for 3D NoC systems.

Fig. 5.4 shows the steady state grid level thermal maps of the die 1 (Layer 0) for both the normal and our proposed adaptive routing approach. In the figure, each tile is comprised of a router (R) including a network interface, its attached PE (P) and memory (M), and the corresponding links. On this layer the efficacy of our proposed adaptive routing approach is seen quite clearly. In this figure, the temperature values are in the Kelvin scale. It can be observed that the drop in the maximum temperature in this layer with our proposed routing approach is around 4K.

To estimate the power consumption, we extended [145] the high-level NoC power simulator presented in [146] to support the 3D NoC architectures. The power is estimated for the interconnection network which includes NoC switches, bus arbiters, intermediate buffers, and interconnects. The thermal analysis and simulation results for the video conference encoding application is presented here. In addition, the average power consumption and APL of the proposed architecture using the thermally efficient hybrid routing are shown in Table 5.1. As predicted, although we are herding more traffic loads to the Layer 2 (closest to the heatsink) to mitigate the peak temperature, there is a negligible APL rise due to the routing adaptivity.

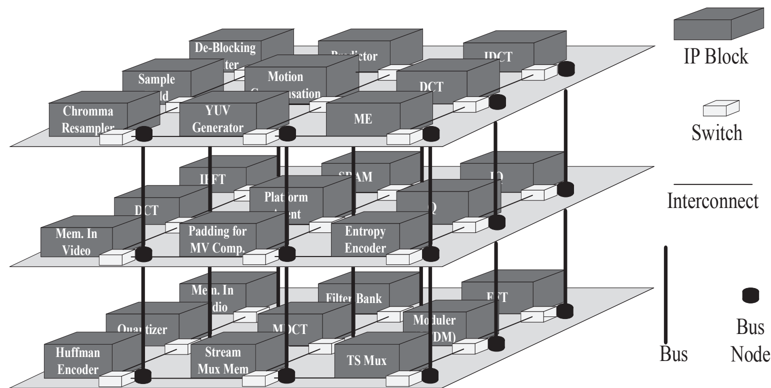

Figure 5.3: Partition and core mapping of the video conference encoding application.

Table 5.1: Power Consumption and Average Packet Latency

3D NoC architecture	**Power consumption (W)**	**Average Packet Latency (cycles)**
Hybrid Bus-NoC 3D Mesh	1.439	166
Proposed Hybrid Bus-NoC 3D Mesh (Hybrid routing)	1.428	168

Table 5.2: Layer temperature profile of the Hybrid Bus-NoC 3D Mesh-based system running the video conference application

Layer ID	**Peak Temperature (°C)**	**Min Temperature (°C)**
Layer 0	114.0	80.8
Layer 1	113.5	77.0
Layer 2	93.5	69.6

5.5 Summary

In this chapter, a thermally efficient routing strategy is introduced for 3D NoC-Bus Hybrid architecture which helps in mitigating on-chip temperatures. The routing, mitigates temperatures by conducting most of the switching activities closer to the heat sink. Our simulations for the proposed Hybrid routing with an integrated video conference application demonstrate

Table 5.3: Layer temperature profile of the proposed Hybrid Bus-NoC 3D Mesh-based system running the video conference application (thermal-aware hybrid routing)

Layer ID	**Peak Temperature (°C)**	**Min Temperature (°C)**
Layer 0	110.0	80.4
Layer 1	110.3	76.7
Layer 2	94.8	69.5

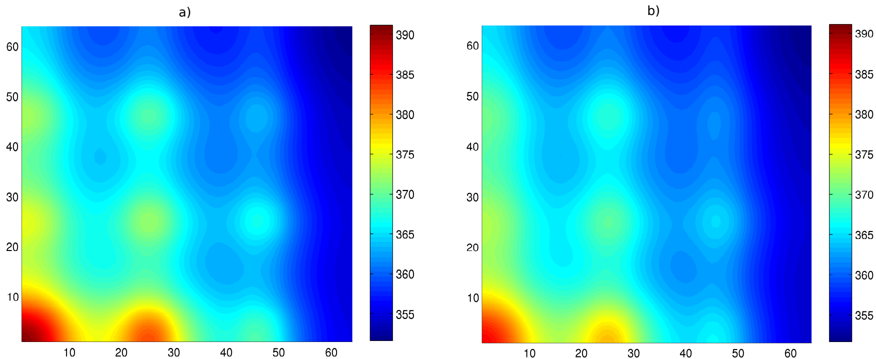

Figure 5.4: Steady-state grid level thermal maps for the die 1(layer 0) for both the normal routing and (a) and our thermal-aware hybrid routing (b).

peak temperature improvements compared to a typical stacked mesh 3D NoC.

Chapter 6

Thermal-Aware Mapping

In this chapter, we have presented an exploration of various thermal-aware placement approaches for both the 2D and 3D stacked systems. Various thermal models have been developed in order to investigate the effect of thermal-aware placement in 2D chip and 3D stacked systems. Using the developed metrics, we proposed an efficient thermal-aware application mapping for a 2D NoC. Steady-state simulations show that the proposed thermal-aware mapping algorithm reduces the effective chip area reeling under high temperatures when compared to the *Tree-Model-Based (TMB)* mapping and *Worst case* mapping. The proposed thermal-aware mapping algorithm considers the developed thermal metrics to map applications on the NoC architecture while maintaining system performance. Based on the location of the nodes, communication and computation of applications we find an efficient way of mapping applications. Also, the algorithm does not need any input temperature data. The aim of the proposed mapping algorithm is to give the designers, insights into thermal characteristics of the system.

6.1 Thermal-Aware Placement in 2D and 3D Chip Systems

Uneven distribution of high temperatures on the chip can lead to timing uncertainities, thereby decreasing the mean time to failure and increasing the problems associated with reliability of the system under consideration. Hence, accurate thermal modeling and analysis is needed in order to achieve thermal objectives whilst maintaining the performance of the system. Also, at the same time it is not completely possible to make a full categorization of thermal-control benefits for different thermal optimization techniques as they are dependent on various factors like how well the workloads are known, how tight the timing deadlines are, how close the utilization of the system is with respect to the maximum load etc. As a general rule of thumb, the

most effective thermal control optimization strategies which do not degrade performance are [33]:

1. At design time we must correctly choose the architectural components and place them on the layout based on the expected application loads (memory access, computing power, etc.).
2. At run-time, the longer we apply the operating system level corrections, the better results we get, but this implies that one has full knowledge of the possible workloads and arrival times.

In order to arrive at a thermal-aware mapping algorithm for 3D stacked systems which meets our performance criteria vis-a-vis throughput and energy, we try and identify different scenarios of placement for known thermally volatile blocks in a thermally optimal way. This is done to gain deep understanding of how the temperature of the system varies with the placement of hotspots at different places on the chip stack. In this work we have studied the following cases wherein we have analyzed the effect of the placement of hotspots has on the chip for both the two dimensional and three dimensional stacked systems. The study includes the evaluation of thermal profile of the said systems along with their maximum(peak), average and minimum temperatures.

6.1.1 Uniform power distribution

At a more abstract level one would assume that the temperature profile of a system would be balanced when the high power sources of the chip are distributed evenly to mitigate thermal hotspots. But in reality, the thermal-aware placement of the building blocks is more complex as we will see that uniform distribution of power does not necessarily yield uniform distribution of temperature.

6.1.2 Thermal-aware placement for a 2D chip system

The four different hotspot placement cases that were analyzed for a 2D chip system are as follows.

1. FP_CENTER: The 4 hotspots are placed at the center of a 4-core silicon die as shown in the Fig. 6.1(a).
2. FP_CORNER: The 4 hotspots are distributed in the four corners of the silicon die as shown in the Fig. 6.1(b).
3. FP_SIDE: The 4 hotspots are placed at the four sides of the silicon die as shown in the Fig. 6.1(c).

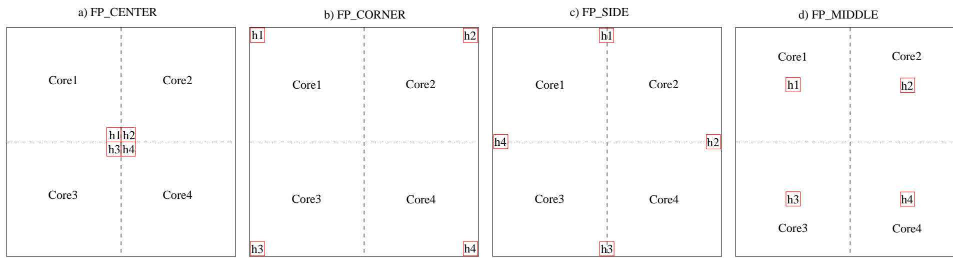

Figure 6.1: Four different hotspot placement cases that were analyzed for a 2D chip system.

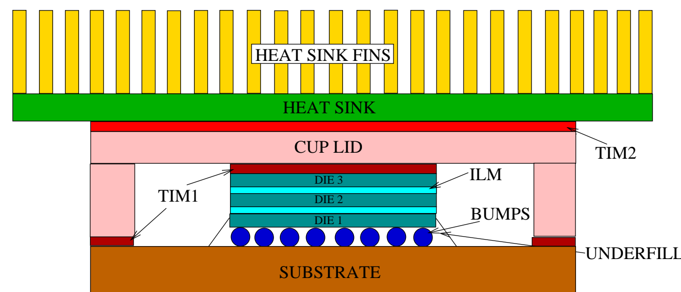

Figure 6.2: Cross-Sectional view of a modern 3D Flip-Chip package with 3 stacked dies.

4. FP_MIDDLE: The 4 hotspots are distributed at the center of each core on the silicon die at a radius of 5.65mm from the center of the die (or placed equidistantly at 8mm apart from each other) as shown in the Fig. 6.1(d).

6.1.3 Thermal-aware placement for a 3D stacked chip systems

In the case of 3D stacked systems, we have analyzed the temperature distribution on different layers of silicon dies in a 3D flip-chip package firstly by placing the thermal hotspots in each layer separately and secondly by placing the hotspots in multiple layers and observing their interaction. The cross-sectional view of a modern 3D flip-chip package is shown in the Fig. 6.2. The following are the thermal configurations that were evaluated.

1. Varying_Hotspots_acrossLayers: In this case we place all the 4 hotspots corresponding to four cores in either one of the 3 layers of the 3D stacked system and observe the corresponding thermal profile of the layers in the system. The hotspots are placed at the center of the sili-

con die as shown in Fig. 6.1(a) in order to test the worst-case thermal behaviour.

2. Varying_WorkloadConditions: In this case we have studied 3 different workload conditions by varying the power consumption of the layers for 3D stacked systems for their thermal behavior.

   (a) Static workload (*Static*): Assuming that the total system power is 200W, each individual die's consume around 66.66W, which is one third of the total power consumption. That is, all the dies in this 3D stacked chip setup consume equal amount of power.

   (b) Adaptive workload (*Adaptive*): In a typical 3D stacked system, the maximum thermal conduction usually takes place from the die which is closer to the heat sink. That particular die also has lower junction temperature and thermal resistance. In [75], Chao et al. have proposed a traffic- and thermal-aware run-time thermal management scheme using proactive routing towards the die closer to the heat sink in order to ensure thermal safety. Here, we analyze a simulation setup wherein we assume that, most of the switching activity is herded away to the die closer to the heat sink. By virtue of this switching activity in the die closer to the heat sink, it would be consuming more power when compared to the other two dies. In this thermal model we assume that DIE-3 (the die closer to heat sink) consumes around 40% more power compared to DIE-2 and around 60% more power compared to DIE-1. So, assuming that the total system power is 200W, then DIE-3 is consuming around 100W, DIE-2 around 60W and DIE-1 around 40W respectively.

   (c) Adaptive workload with a hotspot (*Adaptive_hotspot*): This thermal model is similar to the above adaptive workload (Adaptive) model. But, in here we analyze the effect of hotspot, which we assume gets created in the die closer to the heat sink due to high amount of switching activity happening.

3. Hotspots_inMultiplelayers: In this case the hotspots are placed at different locations in several layers and the interaction of their corresponding thermal fields is observed.

Table 6.1: Modelling parameters [7] [8] [9] [10].

MODEL CONFIGURATION	PARAMETERS	INPUT DATA
Boundary condition	T_{Amb} (°C)	25
	$h_{eff}(W/m^2K)$	840
Heat Sink Base [10]	Size (mm)	100x100
	t_{base} (mm)	5
TIM2	t_{TIM2} (mm)	0.1
	k_{TIM2} (W/mK)	3
Cup Lid (heat spreader)	Size (mm)	50x50
	t_{Lid} (mm)	2
	k_{Lid} (W/mK)	600
TIM1	t_{TIM1} (mm)	0.1
	k_{TIM1} (W/mK)	8
Silicon Die 1 and 2	Size (mm)	20x20
	t_{Die} (mm)	0.6
	k_{Die} (W/mK)	90
Interlayer Material	t_{ILM} (mm)	0.02
	k_{ILM} (W/mK)	4
Lead bumps and Underfill	k_{UF} (W/mK)	1
	t_{UF} (mm)	0.65
Substrate	Size (mm)	50x50
	t_{Sub} (mm)	1.44
	k_{Sub} (W/mK)	17
Boundary condition	$h_{Sub}(W/m^2K)$	10

6.2 Thermal modeling and simplifications

We have developed our thermal models using two different modeling tools. They are Hotspot v 5.0.2 [100] and a commercial tool called COMSOL [147]. We have used Hotspot for modeling and simulating structures for uniform power distribution case and thermal-aware placement of 2D and 3D stacked chip systems (except for the subcases of *Varying_WorkloadConditions* and *Hotspots_inMultiplelayers* where the tool COMSOL has been used). In the following we describe our thermal models in detail.

6.2.1 Thermal modeling using Hotspot

Architectural level thermal modeling tool called Hotspot v.5.0.2 [100] has been used for modeling and simulating our 2D and 3D stacked systems as well as the uniform power distribution case. It has been modified so that the thermal profiles of all the layers of silicon dies in the 3D stacked chip system can be obtained. Hotspots grid model has been exploited which is capable of modeling stacked 3D chips for our thermal simulations. The size of the silicon die (and die's in the case of a 3D stacked chip system) is kept at 16 mm × 16 mm × 0.15 mm. It has been mounted on to the substrate of size 30 mm × 30 mm × 1.0 mm. The heat sink base of size 60 mm × 60 mm × 6.9 mm has been used. The layers of the silicon dies are separated by an interlayer material whose thickness is around 0.02 mm and whose thermal conductivity is set to 4 W/mK. All the other parameters are left to Hotspot default values.

6.2.2 Thermal modeling using COMSOL

We have developed a thermal model of the modern flip-chip package using a commercial tool called COMSOL. It is a finite element based multiphysics modeling and simulation software. Our simulations are based on the heat transfer module of COMSOL multiphysics package. Fig. 6.3 and Fig. 6.4 show the side view and front view of the thermal model we have built. The size of the silicon die 1, 2 and 3 is 20 mm × 20 mm × 0.6 mm which is being mounted on to the substrate of size 50 mm × 50 mm × 1.44 mm. The layers of silicon die are separated by an interlayer material whose thickness is around 0.02 mm. The cup lid which acts as the heat spreader and whose thermal conductivity is very high is placed on top of the silicon die. The thermal interface material (TIM1) which is a kind of thermal grease and has very good adhesive properties, is being used as the filler material in between the heat spreader and the silicon die. The heat sink base of size 100 mm × 100 mm × 5 mm is being used. A vapour chamber is used as the heat sink base and the detailed assumptions can be found in [9]. Instead of including the heat sink fins in our computational model, we have used

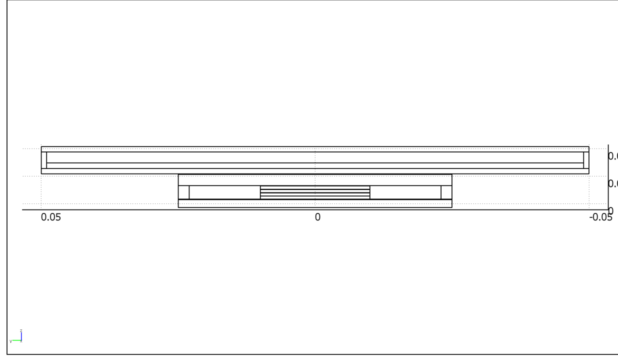

Figure 6.3: Side view of the thermal model using COMSOL.

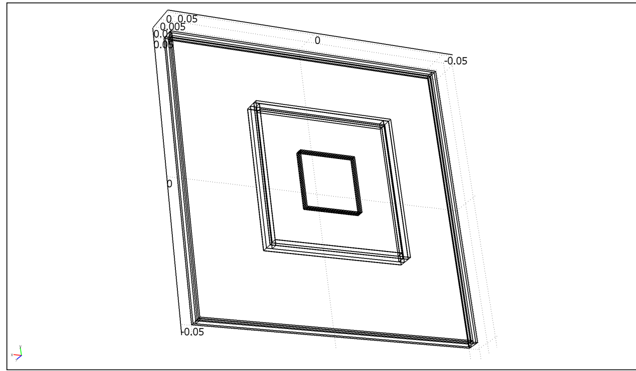

Figure 6.4: Front view of the thermal model using COMSOL.

an effective heat transfer coefficient (h_{eff}) as a boundary condition on the heat sink [10]. Other assumptions related to the geometry of the package and its components, material properties (like thermal conductivity, density and specific heat capacity) and the boundary conditions are taken from the literature [7] [8] [9] [10]. Some important model configuration parameters are represented in the tabular format as shown in Table 6.1. The parameter Q, which is the heat generated per unit volume is applied to the silicon die. The boundary condition for the substrate layer is assumed to be convective and the sides of the package are assumed to be adiabatic.

Modelling interlayer material

Three effective thermal conductivities are used for the lead solder bumps (or underfill layer), substrate layer and the interlayer material (ILM) respectively. The interlayer material in between the silicon dies is modelled as a homogeneous layer in our thermal model. Usually, the through-silicon-

via's (TSV's) have much lower thermal resistance than the silicon dies which helps immensely in heat conduction. We assumed a uniform TSV distribution on the die and obtained the effective interlayer material resistivity based on the TSV density (d_{TSV}) values [8], where d_{TSV} is the ratio of total TSV's area overhead to the total layer area. Coskun et al. [8] have observed that even when the TSV density reaches 1-2%, the temperature profile of the silicon die is only limited by a few degrees, thus justifying the use of homogeneous TSV density in our thermal model. According to the current TSV technology [136], the diameter of each via is 10μm, and the spacing required around the TSV's is assumed to be around 10μm [8]. For our experiments we have assumed around 8 via's/mm^2, that is around 3200 vias spread across the 400 mm^2 area of the silicon die. Hence the TSV density is around 0.062% and the resistivity of the interlayer material is around 0.249 mK/W (i.e. thermal conductivity = 4.016 W/mK) [8].

6.3 Thermal analysis

In this section we have performed various thermal analysis of the placement approaches we discussed in section 6.1 on the models that we have prepared in section 6.2.

6.3.1 Uniform power distribution case

In this part of the work we have performed thermal analysis of a chip which consumes 100W of power. The power is distributed uniformly throughout the chip. Thermal map of the silicon die under such a uniform power distribution is shown in Fig. 6.5. From the thermal map, it can been seen that the temperature is not uniform even as the power distributed on the chip is uniform. In this case we have noticed that the steady-state temperature variation between the maximum (63.93°C) and minimum (61°C) temperatures on the chip is approximately around 3°C. As the power density increases so does the temperature variation. In the case of a 200W system we have noticed that the steady-state temperature variation between the maximum and minimum temperatures is around 6°C. Hence, it can be seen that even if one is able to control power perfectly and manage to have a uniform distribution of power sources on the chip, the temperature will still be left unbalanced.

6.3.2 Thermal-aware placement for a 2D chip system

We have analyzed four different hotspot placement cases in a 2D chip system to arrive at a general solution towards optimal thermal-aware placement.

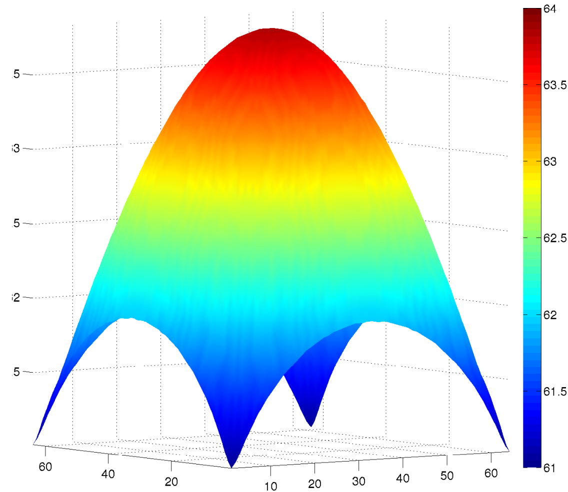

Figure 6.5: Uniform power distribution does not lead to uniform temperature distribution on the silicon die in a Flip-Chip package.

They are *FP_CENTER*, *FP_CORNER*, *FP_SIDE* and *FP_MIDDLE* as described in section 6.1 and as shown in Fig. 6.1. In all the four cases, the sizes of the hotspots *h1*, *h2*, *h3* and *h4* have been fixed at 1mm × 1mm × 0.15mm. The total power consumption of the chip is set to 100 W and the power density of the hotspot is fixed at 200 W/cm^2. The thermal profiles of all the four cases is shown in Fig. 6.6. The maximum/peak, average and minimum temperatures in all the four placement cases of a 2D chip thermal model is shown in Table 6.2. It can be seen that by placing the hotspots at equidistant from each other (in the case of *FP_MIDDLE* they are placed at 8mm apart from each other) a thermally efficient solution can be achieved. We have noticed that the peak temperature is reduced by about 5°C in this case by placing the thermally volatile blocks in an efficient way. Since, the heat transfer along the edges of the silicon is negligible, because it is proportional to the surface area and the air flow would be poor, we have noticed higher peak temperatures in both the *FP_CORNER* and *FP_SIDE* cases when compared to *FP_MIDDLE* case. Also, as will be seen in the subsequent sections that most of the heat will be conducted vertically and spread out via the underneath heat spreader.

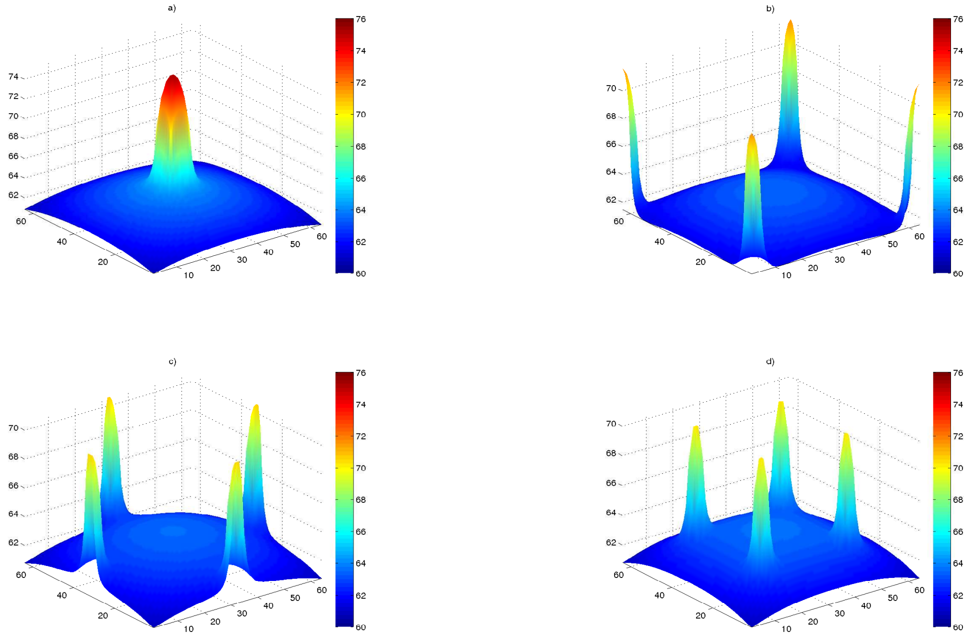

Figure 6.6: Thermal profiles of a) *FP_CENTER*, b) *FP_CORNER*, c) *FP_SIDE* and d) *FP_MIDDLE* cases of a 2D chip system.

6.3.3 Thermal-aware placement for a 3D stacked chip systems

In this case we have analyzed the thermal profiles of 3D stacked chip systems while varying the locations of hotspots in different layers and positions. Mainly, we have analyzed *Varying_Hotspots_acrossLayers*, *Varying_WorkloadConditions* and *Hotspots_inMultiplelayers* thermal configurations which are described in section 6.1. The following are the details.

Varying_Hotspots_acrossLayers

Here, we take the thermally worst case performer of a 2D system (i.e. *FP_MIDDLE*) and place it in either one of the three layers of the 3D stacked chip system and observe the corresponding thermal profile of the layers. That is, we would be observing the case where all the hotspots corresponding to 4 different cores are formed at the center of any of the dies in a 3D stacked system. The sizes of the hotspots has been fixed at 1mm $\times$ 1mm $\times$ 0.15mm and their power density is assumed to be 200W/cm^2. The total power consumption of each of the dies is set to 100W. The thermal profiles of all the 3-layers of the 3D stacked system when all the four hotspots are placed on a die which is either closer to the heat sink (*TOP*), equidistant

Table 6.2: Maximum/peak, average and minimum temperatures in all the four placement cases of a 2D chip that is consuming a total power of 100W.

100W	Temperatures (°C)		
	Max.	Avg.	Min.
FP_CENTER	75.09°C	62.75°C	60.79°C
FP_CORNER	71.60°C	62.57°C	61.47°C
FP_SIDE	70.56°C	62.63°C	60.67°C
FP_MIDDLE	70.25°C	62.70°C	60.81°C

Table 6.3: 3D stacked system: Maximum/peak, average and minimum temperatures in all the three placement cases for a chip system that is consuming a total power of 300W.

300W	Temperatures (°C)								
	Die 1			Die 2			Die 3		
	Max.	Avg.	Min.	Max.	Avg.	Min.	Max.	Avg.	Min.
BOTTOM	129.26°C	105.34°C	100.62°C	126.84°C	104.75°C	100.03°C	118.78°C	102.80°C	98.08°C
MIDDLE	121.28°C	105.34°C	100.78°C	121.11°C	104.75°C	100.16°C	120.54°C	102.80°C	98.08°C
TOP	115.07°C	105.33°C	100.94°C	114.70°C	104.75°C	100.32°C	113.48°C	102.80°C	98.25°C

from the heat sink and the spreader (*MIDDLE*), farther from the heat sink (*BOTTOM*) is shown in Fig. 6.7. The maximum/peak, average and minimum temperatures in all the 3-layers of the 3D stacked system in the three cases (TOP, MIDDLE and BOTTOM) are presented in Table. 6.3. It can be seen that when the hotspots are placed in the die which is closer to the heat sink (*TOP* case) a more uniform thermal profile has been observed in all the layers of the 3D chip system. The peak temperature of the chip stack has been reduced by about 14°C just by efficiently placing the thermally volatile dies closer to the heat sink. It can also be noted that the average temperature of each individual die is the same for all the three cases even though the maximal and minimal temperatures are different. This happens because, at steady-state the system attains thermal equilibrium by both lateral and vertical heat diffusion, thereby resulting in the same average temperatures.

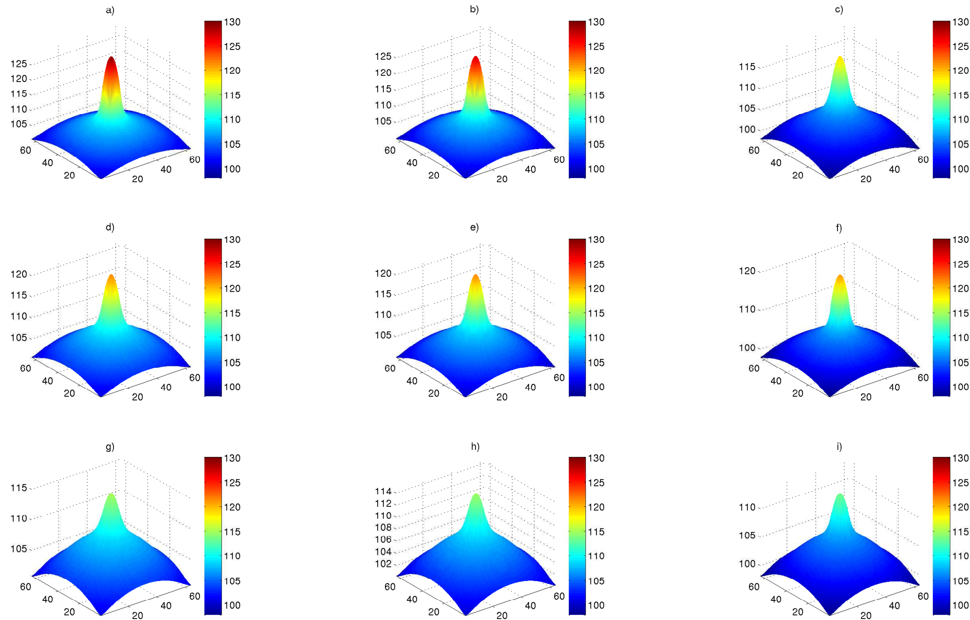

Figure 6.7: Thermal profiles of all the 3-layers of a 3D stacked system when the worst case hotspot scenario occurs in a die which is i) BOTTOM:farther from the heat sink (a,b,c), ii) MIDDLE:equidistant from the heat sink and the heat spreader (d,e,f) iii) TOP:closer to the heat sink (g,h,i).

Varying_WorkloadConditions

We have built a generic three-die stack in a flip-chip package using COMSOL and simulated three different scenarios (Static, Adaptive and Adaptive_hotspot) as described in Section 6.1. In the *Static* case all the 3 dies in the flip-chip package consume equal amount of power. In both the *Adaptive* and *Adaptive_hotspot* case DIE-3 consumes around 40% more power compared to DIE-2 and 60% more power compared to DIE-1. So, assuming that the total power consumption of the system is 200W, then in the *Static* case all the dies consume around 66.66W, whereas in both the *Adaptive* and *Adaptive_hotspot* cases DIE-3 consumes 100W, DIE-2 around 60W and DIE-1 around 40W respectively.

Due to high amount of switching activity happening in DIE-3 we assume that a hotspot gets created at the center of the die and analyze the thermal behaviour of the system in *Adaptive_hotspot* case. Guoping Xu [10] has varied the size of the hotspot from 0.5 mm to 2 mm in his work related to the thermal modeling of multicore systems. In our work the power density of the hotspot which is being generated at the center of DIE-3 in the case of *Adaptive_hotspot* is fixed at 100 W/cm^2 and the dimensions are fixed at 1mm x 1mm x 0.6mm. We have performed the steady-state heat transfer

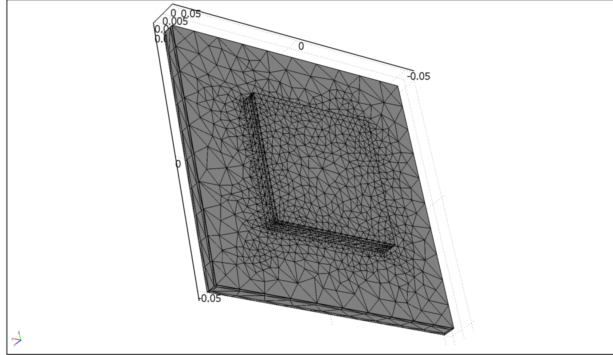

Figure 6.8: Coarse grained meshing of the thermal model.

analysis on the flip-chip package. In the steady-state the heat generated by the three dies is equal to the heat leaving the flip-chip package. During the measurements we have assumed that the power is gradually applied to the chip until the chip has reached the maximum working temperature (i.e. steady state).

For our thermal model we have used coarse grained meshing as shown in Fig. 6.8. Slice and subdomain plots of the simulated thermal model for the *Static* case in which the total system power consumption is 200W is shown in Fig. 6.9 and Fig. 6.10 respectively. For the sake of brevity we are not presenting the slice and subdomain plots for the rest of the cases. The peak temperatures on all the three dies for all the three cases at steady-state is shown in Fig. 6.11, 6.12 and 6.13 respectively and concisely tabulated in Table 6.4. The peak temperature curves are plotted along the X-axis of the dies. It can be observed from those curves that the temperature is maximum at the center of the die and decreases on the edges due to convection.

Table 6.4: Simulation run 1: Peak temperatures on all the three dies for all the three cases in a 200W system.

	Temperatures (°C)		
200W	Static	Adaptive	Adaptive_hotspot
DIE-3	75.6°C	75.6°C	78°C
DIE-2	79.6°C	79°C	79.8°C
DIE-1	82°C	80.5°C	81°C

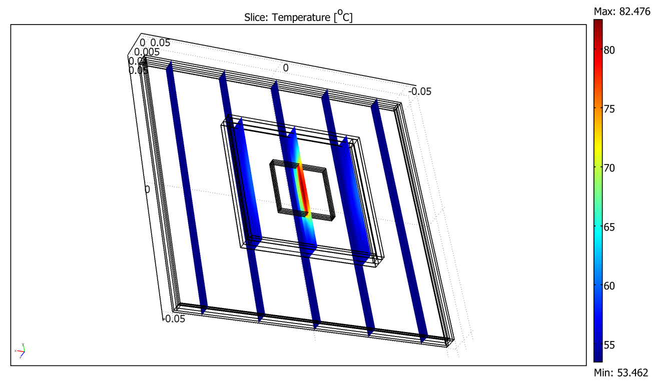

Figure 6.9: Slice plot of the thermal model in the *Static* case. P = 200W, P_{die1}=P_{die2}=P_{die3}= 66.66W.

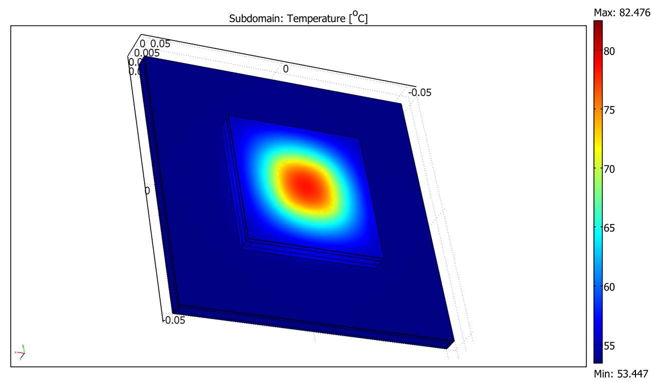

Figure 6.10: Subdomain plot of the thermal model in the *Static* case. P = 200W, P_{die1}=P_{die2}=P_{die3}= 66.66W.

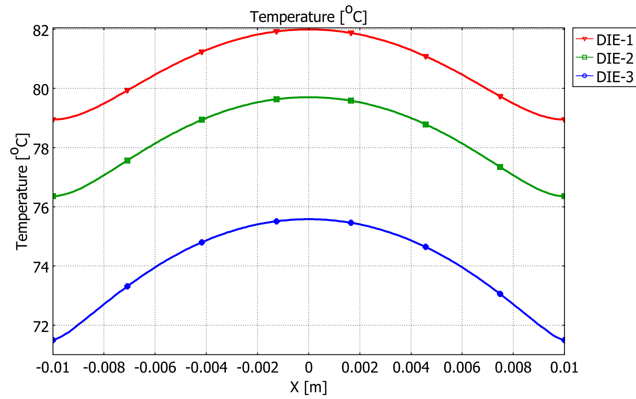

Figure 6.11: Peak temperatures on all the three dies in the *Static* case. P = 200W, P_{die1}=P_{die2}=P_{die3}= 66.66W.

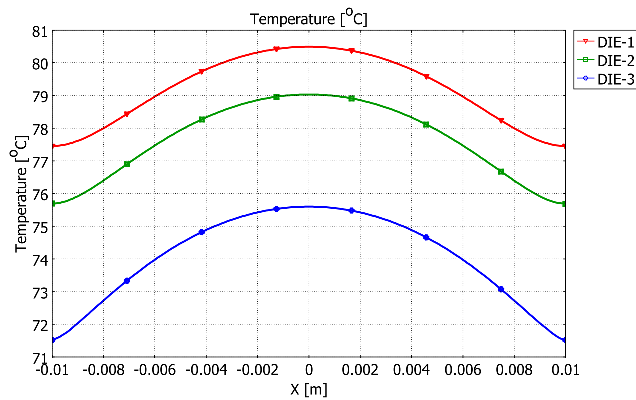

Figure 6.12: Peak temperatures on all the three dies in the *Adaptive* case. P = 200W, P_{die1} = 40W, P_{die2} = 60W, P_{die3} = 100W.

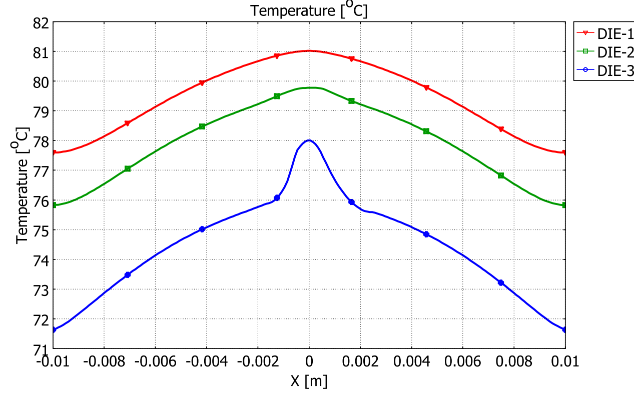

Figure 6.13: Peak temperatures on all the three dies in the *Adaptive_hotspot* case. P = 200W, P_{die1} = 40W, P_{die2} = 60W, P_{die3} = 100W, $P_{d_hotspot}$ = 100W/cm^2.

We have also concisely tabulated the peak temperatures at steady-state in all the three dies in cases where the total power consumption of the system is 100W and 600W. They are shown in Table 6.5 and Table 6.6 respectively. The hotspot parameters in the case where the total power consumption is 100W is the same as 200W system. But in the case of 600W system the hotspot power density is increased to 300W/cm^2 for our simulations.

Table 6.5: Simulation run 2: Peak temperatures on all the three dies for all the three cases in a 100W system.

	Temperatures (°C)		
100W	Static	Adaptive	Adaptive_hotspot
DIE-3	52.7°C	52.7°C	55°C
DIE-2	54.7°C	54.4°C	55.6°C
DIE-1	55.9°C	55.2°C	55.9°C

The Analysis: The following are the three different analysis we have performed.

Static case analysis In the *Static* case, since all the dies consume equal amount of power, generate equal amount of heat at the same time, have almost the same thermal resistance, the only possible direction towards which the heat can flow is the direction of heat sink and the ambient. The proxim-

Table 6.6: Simulation run 3: Peak temperatures on all the three dies for all the three cases in a 600W system.

600W	Temperatures (°C)		
	Static	Adaptive	Adaptive_hotspot
DIE-3	167°C	167°C	174.5°C
DIE-2	179.5°C	177.5°C	179.5°C
DIE-1	186.5°C	182°C	183°C

ity of DIE-3 to the heat sink makes it dissipate more heat than the other two dies. Hence it can be safely said, that the die which is closer to the heat sink (DIE-3) is the coolest, the die which is farther from the heat sink (DIE-1) is the hottest and the die which is sandwiched (DIE-2) has a temperature somewhere in between them. This phenomenon can be observed in all the three simulation runs we have conducted.

Adaptive case analysis In the *Adaptive* case it can be clearly seen that the peak temperature on DIE-3 is the same as the *Static* case despite it consuming around 33.3% more power. The DIE-3 in this case is consuming around 40% more power compared to DIE-2 and 60% more power than DIE-1. Despite dramatic power reductions on DIE-1 and DIE-2 and herding the tasks towards DIE-3, it can be seen that there is minimal impact on peak temperatures on the three dies at steady-state when compared to the *Static* case, where all the dies are consuming equal amount of power. This is because, since DIE-3 consumes more power it generates more heat when compared to the other two dies. Hence the direction of the flow of heat is not only towards the heat sink, but also towards the dies which are cooler compared to DIE-3 at any given time. So, by the time steady-state is actually reached the system attains thermal equilibrium by dissipating heat from the one generating more to the one generating less and to the ambient via the heat sink. Hence one does not notice the anticipated reduction in peak temperatures in DIE-2 and DIE-1. If we consider non-uniform power distribution of various on-chip components then we would notice that hearding most of the tasks onto the die closer to the heat sink would significantly improve the thermal profile of the system as can be seen in Section 5.2.

Adaptive_hotspot case analysis Since the *Adaptive* case does not have much reductions in peak temperatures when compared to the *Static* case, we have experimented further with the presence of a hotspot in DIE-3 which we

assume gets created due to excessive routing and herding of tasks towards it. Even then, we have noticed that the peak temperatures on DIE-2 and DIE-1 are not very much different from both the *Static* and *Adaptive*. On DIE-3 itself we have noticed a slight increase in peak temperature which in this case is the temperature of the hotspot.

Hotspots_inMultiplelayers

As is the case with typical chip stacks, it is not unusual for them to have more than one hotspot being active at the same time. Those hotspots could be active in the same die or in different dies simultaneously. Hence, exploring the interaction between those hotspots is of utmost importance and can lead to interesting conclusions. Fig. 6.14 shows the interaction of two hotspots on the same die (DIE-3). It has been obtained by fixing the hotspot at the center of the die and varying the location of the other. The variable 'd' in the plot is the distance between the centers of those two hotspots. For the sake of comparison and clarity, we have also included a temperature plot with a single hotspot at the center of the die in Fig. 6.14. In this study, we have modeled a 3D stacked system whose overall power consumption is 200W, with each die consuming around 66.66W. The two hotspots have the same dimensions of 1mm x 1mm x 0.6mm and their power density is fixed at 100 W/cm^2. It can be seen from the Fig. 6.14 that the maximum temperature on the die actually depends on the distance between the two hotspots. When the two hotspots are closer to each other (d = 2mm),there is an increase of about 0.5°C compared to the case where only a single hotspot is present. This value increases further as the hotspots come more closer to each other and culminates in achieving a temperature of 80.5°C (d = 0mm), which is almost 2.2°C more than the case with a single hotspot. As the two hotspots move away from each other there is very little thermal interaction between them and the peak temperature on the die is almost equal to the case when a single hotspot is present.

We have also studied two special cases in order to understand the interaction of hotspots in different vertical layers of the dies. In the first case, we have analyzed the interaction of hotspots, wherein each hotspot is located at the center of its die edge respectively. In the second case, we have analyzed the interaction of hotspots when they are spread evenly across different dies. That is, a hotspot is present at the center of the right most edge of DIE-1, center of the DIE-2 and at the left most edge of DIE-3 respectively. Comparing Fig. 6.15 and Fig. 6.16, it can be observed that the peak temperature on each die can be reduced by efficiently placing the thermally risky blocks far from each other, so that their corresponding thermal fields do not interact with each other. In this analysis, the maximum temperature on the hottest die (DIE-1) has been reduced from 85.5°C to 83.5°C.

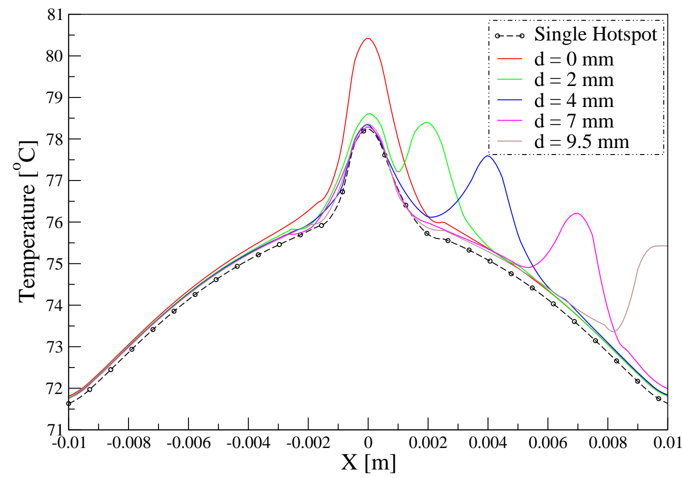

Figure 6.14: Interaction of two hotspots located on the same die (DIE-3). The plot is obtained by fixing the location of one hotspot at the center of the die and varying the location of the other. The distance 'd' in the plot is the distance between the centers of two hotspots.

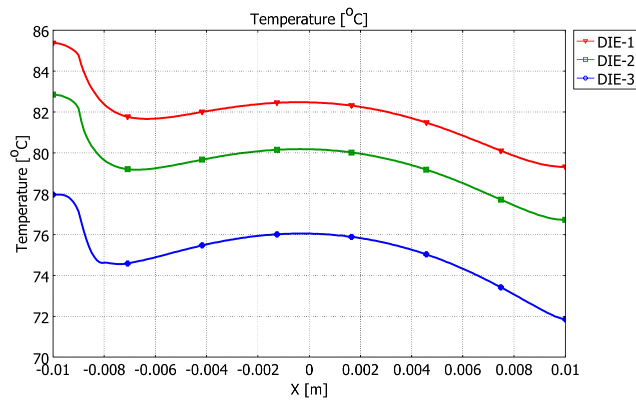

Figure 6.15: Interaction of hotspots located in different vertically stacked layers. Each hotspot is located at the center of its die edge respectively.

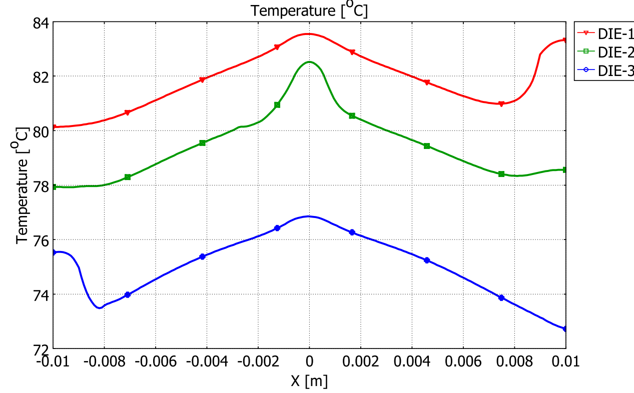

Figure 6.16: Interaction of hotspots located in different vertically stacked layers, but distributed efficiently so that their thermal fields do not interact with each other.

6.4 Proposed temperature mitigation techniques

Based on the thermal modeling and analysis in the previous sections and the developed metrics thereafter, we propose two temperature mitigation techniques in this section which forms the core of our work. The first one deals with a thermally efficient way to route data in a 3D network on a chip aptly titled "thermally efficient routing strategy" and the second one deals with an efficient thermal-aware application mapping for a 2D network on a chip. In the following subsection 6.4.1 we start off by discussing the proposed temperature-aware mapping technique for 2D planar NoC's and then proceed to further elaborate on the thermal model used to evaluate the technique in subsection 6.4.2.

6.4.1 Thermal-aware mapping for 2D NoC

In this Section we propose a three step mapping algorithm which takes placement of hot-spots into account in order to achieve a better thermal management. We assume a 2D homogenous multicore system where a set of applications (AP) are supposed to run simultaneously. Each application ($A_p \subset$ AP) is composed of several communicating tasks, modeled by a task graph $A_p = TG(T, HS, E)$. Each vertex $t_i \in T$ represents one task of the application A_p, while the edge $e_{i,j} \in E$ stands for a communication between the source task t_i, and the destination task t_j. Task graph of an application with 7 tasks is shown in Fig. 6.17. The amount of data transferred from a task t_i to t_j of edge $e_{i,j}$ is denoted as $w_{i,j}$, which is written on each edge. An application might have a set of hot-spot tasks $HS \subset T$. A task is defined to be hot-spot if it requires processing resources more than a predefined

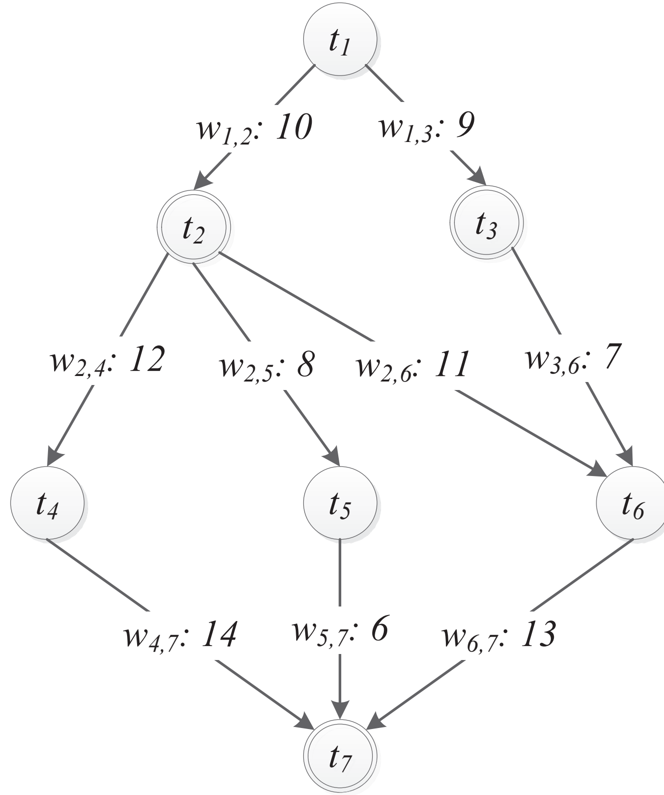

Figure 6.17: An example task graph of an application consisting of 7 tasks. Hotspot tasks are depicted as concentric circles.

threshold. Hot-spot tasks are represented by double circles in Fig. 6.17.

The proposed algorithm results in fewer hot spot area compared to other state-of the art works [148]. The three main steps of the algorithm are:

1. Application mapping or region selection: In this step, a near convex area of required number of nodes is dedicated to each application.

2. Hot-Spot placement: In this step, we determine the best placement of hot-spots within the application regions using the extracted metrics from Section 6.3. This has been explained in more detail below.

3. Task mapping: hot-spot tasks as well as other tasks of each application are mapped onto nodes of their specified region.

Region selection

The first step involves selecting a set of nodes (called region), onto which an application is mapped. A convex area which decreases the average distance

between allocated nodes is targeted for each application. Yang [148] et al has proposed an NAD-based region selection algorithm which has been used as the first step for our algorithm.

Hotspot placement

After the mapping regions of all applications are selected in the first step using the NAD-based algorithm, hot-spot placement step is performed. In this step, applications are prioritized based on their ratio of hot-spot tasks, called $HSR_{A_p}=\frac{|HS|}{|T|}$. In the other words, the hot-spot placement is performed first for the application with the largest number of hot-spot tasks compared to its total number of tasks. To asses selected nodes (*HSN*), we use the equation (6.1) which is the sum of the distance between all pairs of the selected nodes:

$$Dist_{HSN} = \sum_{n_i \in HSN} \sum_{n_j \in HSN} Dist(n_i, n_j) \tag{6.1}$$

Where Dist(n_i, n_j) is the diametric distance between nodes n_i and n_j. To select the appropriate hotspot nodes within a region, these general rules are followed:

1. Avoid allocating hot-spot tasks to the neighboring nodes throughout the chip (both within the region, as well as the neighboring regions).
2. When the distance of two candidate HSNs are within 10% difference, the one which has less corner nodes is preferred. Fewer side nodes are preferred in case of equal corner nodes.

The pseudocode of the hot-spot placement step is shown in Algorithm 2. The applied randomness (line 6) is to decrease the problem size and achieve expected results in acceptable time. Using the proposed algorithm, the hot spot nodes will be placed well distributed over the chip while keeping them away from corners and sides as much as possible. The 10% threshold level of rule 2 is extracted with practical experiments.

Task Mapping

After the hotspot nodes are placed over the chip, task mapping process allocates nodes of each region to its application's tasks. The proposed mapping aims at minimizing the Average Weighted Manhattan Distance (*AWMD*) metric introduced in [149]. The *AWMD* for the mapping result of a given task t_{cur}, which is communicating with a set of already mapped tasks T_t, is defined in equation (6.2).

Algorithm 2 *Pseudocode algorithm for Hot-spot placement*

Input: Set of applications AP, set of selected regions R

Output: Hot-spot placement within each application region

```
1: Sort applications of AP in descending order of their HSR
2: HSN_sys ← ∅
3: for p = 1 → |AP| do
4:    #Neigbrs ← ∞
5:    Dist_best ← 0
6:    for several random selection of possible HSN within region R_Ap do
7:       if number of neighboring nodes exist in HSN's ∪ HSN_sys is ≤
         #Neigbrs then
8:          if (Dist_HSN > 1.1 Dist_best) or (Dist_HSN is within 10% of
            Dist_best and rule 2 is followed) then
9:             #Neigbrs ← number of neighboring nodes exist in HSN ∪
               HSN_sys
10:            HSN_best ← HSN
11:         end if
12:      end if
13:   end for
14:   HSN_Ap ← HSN_best
15:   HSN_sys ← HSN_sys ∪ HSN_best
16: end for
```

$$AWMD_{Map(t_{cur})} = \frac{\sum_{t_i \in T_t} MD(Map(t_{cur}), Map(t_i)) \times (w_{i,cur} + w_{cur,i})}{\sum_{t_i \in T_t} (w_{i,cur} + w_{cur,i})} \tag{6.2}$$

The *AWMD* for the final result of a mapping function is defined in equation (6.3).

$$AWMD_{Map(A_p)} = \frac{\sum_{\forall e_{i,j} \in E} w_{i,j} \times MD(Map(t_i), map(t_j))}{\sum w_{i,j}} \tag{6.3}$$

As shown in pseudo code of Algorithm 3, task mapping process considers several randomly selected possible mappings of hot-spot tasks (line 3). This is to decrease the required time in finding a solution. Within each possible mapping of hot-spot tasks, the algorithm selects the task (t_{cur}) which is communicating the most with already mapped tasks (line 6), and find the node which minimizes the *AWMD* of the task (line 7). This is repeated until all tasks of the application are mapped (line 5). Among the all resulted

Algorithm 3 *Task allocation in a region with hotspot nodes*

Input: Application *Ap(T, HS, E)*, and its corresponding region R_{Ap} and selected hotspot nodes HSN_{Ap}

Output: Mapping Result of application *Ap* within the region R_{Ap}

1: $AWMD_{best} \leftarrow \infty$
2: **for** several random selection of possible mappings of *HS* tasks onto *HSN* nodes **do**
3: *Unmapped* $\leftarrow$ *T* - *HS*
4: *Mapped* $\leftarrow$ *HS*
5: **while** *Unmapped* $\neq \emptyset$ **do**
6: $t_{cur} \leftarrow$ find the task with maximum communication with *Mapped* tasks
7: $Map(t_{cur}) \leftarrow$ find the node which minimizes the *AWMD* of task t_{cur}
8: *Mapped* $\leftarrow$ *Mapped* $\cup\ t_{cur}$
9: *Unmapped* $\leftarrow$ *Unmapped* - t_{cur}
10: **end while**
11: **if** $AWMD_{Map(Ap)} < AWMD_{best}$ **then**
12: *Result* $\leftarrow$ *Map(Ap)*
13: **end if**
14: **end for**

mappings, the one with minimum *AWMD* is chosen for the application (line 11).

6.4.2 Thermal Modelling

In this subsection we talk about the thermal models that were built in order to evaluate the two important contributions of this paper. We have developed the thermal models in both the cases using HotSpot v.5.0. [100]. Hotspot is an accurate and fast thermal model suitable for use in architectural studies and is based on an equivalent circuit of thermal resistances and capacitances that correspond to micro-architecture blocks and essential aspects of the thermal package. Hotspot takes a power trace file, floorplan file and a lot of other modifiable model configurations and parameters as its input.

Thermal model to evaluate thermal aware mapping algorithm

We have built a thermal model of a 6×6 NoC using Hotspot. The tile geometry has been adopted from Intel's 65nm based 80-core processor [6] and the details are shown in Fig. 6.18. The size of the silicon die is 9.0

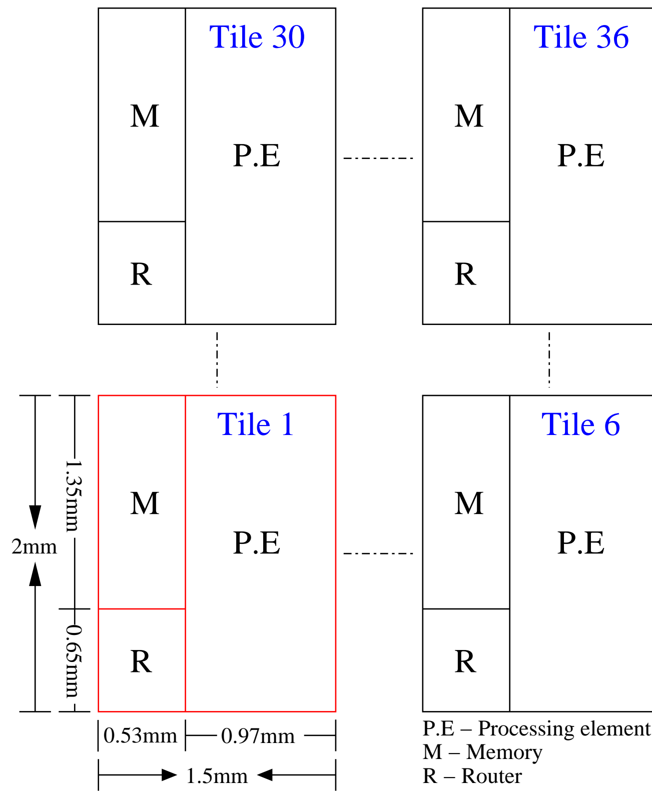

Figure 6.18: A 6×6 NoC depicting the blocks and dimensions of each tile. The dimensions are adopted from Intel's 65nm based 80-core processor [6].

mm × 12.0 mm × 0.15 mm. The total power consumption of the NoC has been set to 200W. In our model, each application has several tasks and is modelled with a task graph. The task graph includes the computation and communication information of the tasks belonging to an application. It describes the behaviour of the tasks in question during the application run. Using this information both the CPU usage statistics and the activity of the routers is computed with the help of our in-house cycle accurate multi-core simulator which uses a pruned version of Noxim [150] as its communication platform. The convection capacitance and convection resistance of the heat sink are 140.4 J/K and 0.1 K/W respectively. Other parameters are left unchanged from the Hotspot's configuration file.

6.5 Simulation results and analysis

6.5.1 Thermal-aware mapping for 2D NoC

We have compared our proposed thermal-aware mapping strategy with two other mapping algorithms (Tree-Model-Based mapping and Worst case mapping). Tree-Model-Based mapping (TMB) uses the work presented by [148] as its baseline which tries to map applications on regions while minimizing the communication cost regardless of thermal criteria. On the other hand the scenario where applications are mapped onto their regions while their hot-spots are placed together (where $Dist_{HSN}$ is minimized) in the center is also examined. This is referred as worst-case scenario in the results.

To study the practical impact of our hotspot placement considered application mapping on the system performance, a 6×6 mesh of processing elements are considered where four applications with sizes of 7, 8, 10 and 11 tasks are considered. As baseline, applications are mapped onto the system with heuristic proposed in [148] as well as our proposed algorithm. To extract packet latency, a inhouse developed cycle-accurate message-passing multi-core simulator is utilized.

Applications are mapped onto the platform using the aforesaid three mapping algorithms. Several simulations are run for each mapping over different packet injection rates and the average packet latency of the network is extracted. As can be seen in Fig. 6.19, all mapping algorithms cause almost the same latency for small injection rates, while the network saturation point varies over different mapping algorithms. Results show that in the performed placement analysis, our presented algorithm trades off around 5% reduction in saturation point (0.067 vs 0.065 Flit/Core/Clk) with balanced temperature on the chip. Our performed placement analysis shows, that the presented algorithm can keep a balance between temperature on the chip and performance of the running applications.

Fig. 6.20 shows the thermal maps of the silicon die for worst case, Tree-Model-Based (TMB) and thermal-aware mapping cases. Table 6.7 depicts the amount of chip area in mm^2 that is above a certain temperature for all the 3 cases we discussed. Comparing the *thermal-aware* mapping with that of the *TMB mapping*, it can be seen, that there has been a 31% reduction in the area of the chip whose temperature is above 76°C and around 80% reduction in the area of the chip whose temperature is above 77°C in the case of TMB mapping. Compared to the *worst case* mapping, our proposed thermal-aware mapping reduces the area of the chip which is above 76°C by 59% and the area of the chip which is above 77°C by about 94%. The proposed mapping technique explores a random set of all possible placements to find an efficient solution during the two steps hotspot placement and task mapping. This can be extended to exhaustive search and still re-

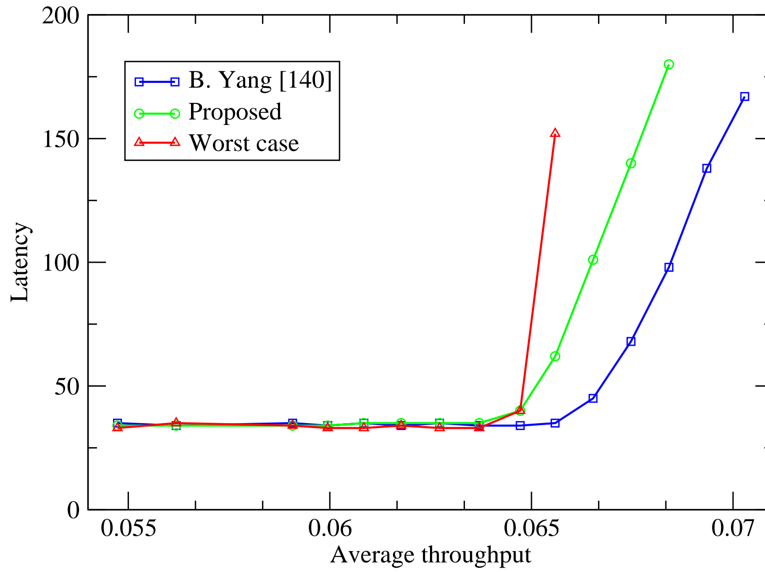

Figure 6.19: Latency vs throughput.

Table 6.7: Table depicting the amount of chip area under a particular temperature.

Temperature (°C)	Area (mm^2)			*% Reduction in area*	
	Worst	*TMB*	*Thermal*	*TMB_thermal*	*Worst_thermal*
> 76°C	28.89	17.13	11.81	31.07	59.12
> 76.5°C	25.20	11.02	6.64	39.71	73.64
> 77°C	20.40	6.30	1.26	79.91	93.79
> 77.5°C	14.55	1.92	0	100	100

main reasonable for the static mapping case, however, with the increase in the number of cores and imposed dynamic workloads, a more sophisticated heuristic is planned as future work.

6.6 Summary

In this chapter, We have presented an exploration of thermal-aware placement approaches for both the 2D and 3D stacked systems. Various thermal models have been developed in order to investigate the effect of uniform power distribution, thermal-aware placement in 2D chip and 3D stacked

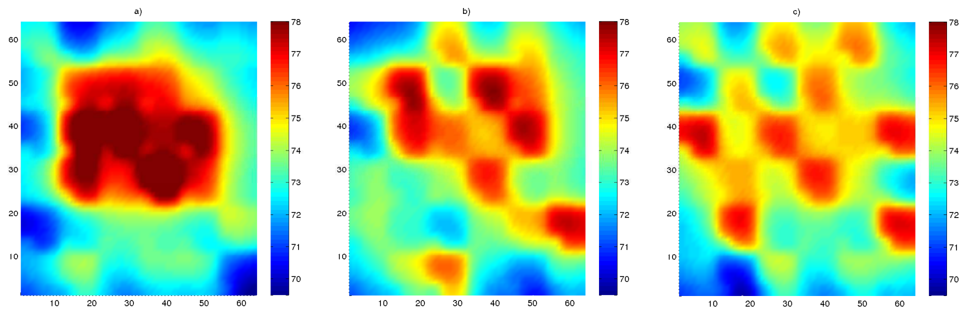

Figure 6.20: Thermal maps of a) Worst case b) TMB mapping case and c) Thermal-aware case

systems on the thermal performance of the system. The resulting metrics thus obtained from our parametric study provides thermal guidance for circuit designers on optimizing the die layout from the thermal perspective for various thermal parameters. Using the developed metrics, we proposed an efficient thermal-aware application mapping for a 2D NoC. Steady-state simulations show that the proposed thermal-aware mapping algorithm reduces the effective chip area reeling under high temperatures when compared to the *TMB* and *Worst case* mapping.

Chapter 7

Conclusions and Future Work

7.1 Summary and Conclusions

Changing technology and evolving consumer demand is paving way for smaller and faster electronic devices. With the continuous shrinking of manufacturing technology as predicted by the technology roadmap, it has now become possible to integrate hundreds and possibly thousands of processing cores on a single silicon die. Today's large systems will become one of the final building blocks of a much larger system all interconnected with processing elements, I/O devices and memories thereby increasing the complexity manifold. All of this complexity translates into higher on-chip power density, hotspots, higher operating temperatures, non-uniform thermal gradients and reduced semiconductor and system reliability. The thermal resistance of the chip increases if the heat cannot escape to the ambient at a fast enough rate, thereby leading to higher junction temperatures and lowering the devices mean time to failure (MTTF). That is, the devices would have high thermal wear-outs and very short lifetimes. Other important factors which affects system reliability are electro-migration (EM), oxide breakdown, negative bias temperature instability (NBTI), hot carrier injection and thermal cycling.

The thermal issues are all the more exacerbated by the advent of stacked three dimensional (3D) integrated systems, as both the power and temperature distribution become increasingly non-uniform compared to a 2D planar chip. Also, the heat in the 3D stacked system flows both in the lateral and in the vertical directions thereby contributing to the thermal gradients. Hence, it is important to address the aggravated thermal issues using both design time and run-time thermal-aware techniques, so that the system is not only thermally controlled, but also thermally balanced. This thesis mainly ex-

plored design time thermal efficient routing and mapping techniques.

On-chip thermal sensors are one of the most popular, accurate and cost-effective ways to obtain run-time thermal information of the processors. Recent processor trends indicate that, the use and number of on-chip thermal sensors will continue to grow. The run-time thermal information provided by the on-chip sensors can be used for various dynamic thermal management (DTM) strategies. In this context, we presented a self-timed thermal monitoring strategy which is based on the liberal use of on-chip thermal sensors. For this, we implemented a novel thermal sensing circuit, which converts analog temperature information into digital form, for further processing. We have proposed the use of leakage current based thermal sensing circuit for monitoring purposes, as leakage currents are found to be sensitive to temperature variations and increase with technology scaling. We have also presented a novel thermal sensing and monitoring interconnection network infrastructure which is based on self-timed signaling, and comprising of an encoder/transmitter and decoder/receiver. The presented sensing architecture has been made more resilient to various types of noises that may occur in the system. This is done by performing power supply noise, additive noise on sensor input signal and dynamic power supply voltage variation analysis on the thermal sensing circuit. This shows that the circuit is robust enough under different operating temperatures.

On-chip temperature has become such a complex issue, that it needs to be addressed from the earliest design stages of the system. The early design choices like the number and complexity of cores, types of materials and packaging being used, dictate the temperature patterns of the system. As a result, it has become mandatory for the system designers to study thermal management issues from the early design stages. For that, accurate thermal modeling and analysis at design time is essential. We started by a) performing thermal analysis on interconnects and packaging. b) Later, considering the emergence of 3D stacked systems, we built various thermal models and performed thermal analysis on them. Based on the thermal modeling and analysis study, it is observed that:

1. In order to understand and identify the challenges presented by the increase in temperature that is contributed by various system components like the global interconnects, it becomes necessary to perform accurate thermal modeling and analysis at design time. We have analysed the spatial temperature distribution on a global interconnect link in 65nm CMOS technology from ST microelectronics. It has been found that the average temperature rise ΔT along the length of the conductor is around 6.8°C for a global interconnection link. The impact of this temperature rise has been analysed for both the voltage mode and current mode signaling.

2. We developed a thermal model of a 3D multicore system in a flip-chip package and investigated the affects of hotspots and placement of silicon die layers on the thermal performance of the system. We evaluated two different thermal models (depending on whether the processing die or the memory die is placed near the substrate; Model-I and Model-II respectively). We performed both the steady-state and transient simulations of our system in a flip-chip package. It has been found that in steady-state for the case where the memory layer dissipates around 10% of the power consumed by the processing core, an overall improvement of 0.6°C/W is obtained in the thermal resistance by effectively placing the silicon layers. For the same case, it has been observed that the difference in the maximum temperature of memory and processing die layers is around 4°C for model-I and 0.3°C for model-II. We have also quantified the improvement that is required in the thermal resistance of the heat-sink for a 3D stacked system when compared to a single-die system.

However, the complexity of thermal behaviour increases with 3D integration due to the increase in power density. In this work we have introduced a thermally efficient routing strategy for 3D NoC-Bus architectures by hybridizing a proposed congestion-aware routing algorithm with other available algorithms. That is, we have used a congestion-aware routing algorithm called *AdaptiveZ* for vertical (inter-layer) communication. In *AdaptiveZ* the first available bus pillar on the way for vertical communication is used. We then hybridize the *AdaptiveZ* routing with other available algorithms (namely, *LastZ* and XY) in order to arrive at our thermally-efficient routing strategy. Our routing strategy helps in mitigating the on-chip temperatures by herding most of the switching activity to the die which is closer to the heat sink where most of the conduction to the ambient takes place. Our simulations with a real world benchmark demonstrate that there has been a decrease of 4°C in the peak temperatures when compared to a typical stacked mesh 3D NoC architecture.

In order to arrive at a thermally balanced system, it is pivotal that not only a thermally-efficient routing strategy which is run-time aware be used, but also the workload consisting of several applications be mapped in a more thermally efficient way at design time. In this thesis we have developed an efficient thermal-aware application mapping algorithm for 2D planar NoC platforms. We performed an exploration of various thermal-aware placement approaches for 2D and 3D stacked systems, by developing several thermal models and extracting thermal metrics which were used to investigate thermal-aware placement approaches. The algorithm follows three main steps: 1) Application mapping or region selection: in which a near convex area within the required number of nodes is dedicated to each

application 2) Hot-Spot prone block placement: where the best possible placement of hotspot prone blocks within the application regions is achieved using the extracted metrics 3) Task mapping: where the hotspot prone tasks and other tasks of each application are mapped onto the nodes within the specified regions. Our extensive steady-state simulations demonstrate that the proposed thermal-aware mapping algorithm reduces the effective chip area reeling under high temperatures when compared to the *Tree-Model-Based (TMB)* mapping and *Worst case* mapping.

7.2 Future Work

This thesis has explored the potential of achieving thermally efficient 2D and 3D systems by proposing effective routing based and mapping based algorithmic techniques. However, still further explorations are possible. Future work of this thesis can be classified into research directions which have both short term and long term implications.

Short Term Research Implications

Combining both the routing and mapping based techniques will result in additional thermal safety for the system. In the short term the works that were discussed in this thesis, mainly the thermal-aware routing and mapping can be integrated together and then further extended with new algorithms and new research directions. Firstly, the static 2D application mapping can be extended to dynamic application mapping and later to 3D stacked NoC's. This can then be integrated with our novel and thermally efficient inter-layer communication scheme for 3D NoC-Bus hybrid architectures in order to arrive at a run-time thermal management scheme which provides additional thermal safety for the system under consideration. The run-time thermal management scheme can be integrated further with other low-level, hardware based heat reduction techniques (like DVFS, clock gating, fetch toggling, stop-and-go policies) which will yield a more thermally balanced system that works optimally under varying workloads. Evaluating this combined methodology for a large 3D NoC based system and exploring it further on different network topologies would lead to interesting conclusions.

For 3D stacked NoC systems, further investigations are needed to find novel network architectures, topologies and protocols which when combined with future 3D integration technologies will result in thermally balanced systems. Although, there are some concerns towards the yield and area overhead, vis-a-vis the usage of thermal via's in 3D stacked systems, their usage needs to be further explored, as they are considered to be an effective means to drive out heat from the chip to the ambient. Also, the usage of other active thermal cooling mechanisms (like using liquid cooling) needs to

be further explored as they do not sacrifice performance. But, using liquid cooling trades off power for performance. Power is required for injecting liquids through chip capillaries using tiny motors, which may be quite a lot if they are not combined with other power and thermal optimizations.

Long Term Research Implications

Power consumption and heat dissipation of the current state-of-the-art microprocessors are becoming major limiting factors for their performance evolution. As their use increases in the design of battery powered devices and high performance computers, different power and thermal management strategies have been proposed and implemented to overcome their performance limitation. At the same time, software applications are becoming more complex with every iteration and have a large impact on power and thermal maps of the system. Until now, there has been less focus on studying how software applications can be used for thermal management and whether or not it is feasible to implement thermal-aware software applications. Apart from embedding Dynamic Thermal Management (DTM) cues into software applications, designing novel thermal optimization strategies for multiple abstraction layers and use cases needs to be further explored.

To address the temperature issues for heterogeneous architectures comprehensively, it is prudent to start with a framework for thermal estimation and management, where balanced thermal profile can be achieved at different levels of abstraction. It is known that more thermal benefits are to be obtained at higher levels of abstraction. Hence, in this context, it is required to create a framework for thermal estimation and management, where balanced thermal profile can be achieved at different levels of abstraction (core, server, data center level). At the same time the framework and techniques should not be architecture-dependent and should be applicable to any target architecture. The thermal framework should effectively address the following themes:

1. Thermal profiling of software applications using compilation based mechanism and middle-ware based mechanism.

2. Extracting an accurate thermal model after multiple data analysis runs and integrating it for an application set running on a target architecture with a higher-level abstraction model of temperature dissipated at core, server or data center level.

3. Designing of thermal optimization strategies for multiple abstraction layers (thermal-aware software applications by optimizing the source code of applications and compiler options, efficient thermal-aware load distribution, parallelization etc.).

In addition, two different ways of building thermal management frameworks can be further explored.

1. Exploration and implementation of dynamic compilation based framework for controlling energy, performance and temperature, and

2. The middle-ware based framework which takes into account the high-level thermal QoS requirements from the applications.

Bibliography

[1] Massoud Pedram and Shahin Nazarian. Thermal modeling, analysis, and management in vlsi circuits: principles and methods. *Proceedings of the IEEE*, 94(8):1487–1501, 2006.

[2] Ibs electronics.

[3] Semiconductor Industry Association et al. International technology roadmap for semiconductors (itrs), 2011 edition, system drivers, 2009.

[4] Pablo Ituero, José L Ayala, and Marisa Lopez-Vallejo. Leakage-based on-chip thermal sensor for cmos technology. In *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pages 3327–3330. IEEE, 2007.

[5] ST Microelectronics. *Design Rule Manual for 65nm Bulk CMOS process*, 2007.

[6] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar. A 5-ghz mesh interconnect for a teraflops processor. *Micro, IEEE*, 27(5):51–61, 2007.

[7] Semiconductor Industry Association et al. International technology roadmap for semiconductors (itrs), 2009 edition, 2009.

[8] A.K. Coskun, J.L. Ayala, D. Atienza, T.S. Rosing, and Y. Leblebici. Dynamic thermal management in 3d multicore architectures. In *Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE'09.*, pages 1410–1415. IEEE, 2009.

[9] Guoping Xu, Bruce Guenin, and Marlin Vogel. Extension of air cooling for high power processors. In *Thermal and Thermomechanical Phenomena in Electronic Systems, 2004. ITHERM'04. The Ninth Intersociety Conference on*, pages 186–193. IEEE, 2004.

[10] Guoping Xu. Thermal modeling of multi-core processors. In *Thermal and Thermomechanical Phenomena in Electronics Systems, 2006.*

ITHERM'06. The Tenth Intersociety Conference on, pages 96–100. IEEE, 2006.

[11] Shekhar Borkar and Andrew A Chien. The future of microprocessors. *Communications of the ACM*, 54(5):67–77, 2011.

[12] Gordon E Moore et al. Cramming more components onto integrated circuits, 1965.

[13] Edmund K. Cheng. Thermal design issues–impacting design to packaging. *Welcome to the latest edition of Future Fab International...*, page 43.

[14] Arman Vassighi and Manoj Sachdev. Thermal runaway in integrated circuits. *Device and Materials Reliability, IEEE Transactions on*, 6(2):300–305, 2006.

[15] Farzan Fallah and Massoud Pedram. Standby and active leakage current control and minimization in cmos vlsi circuits. *IEICE transactions on electronics*, 88(4):509–519, 2005.

[16] Clemens JM Lasance. Thermally driven reliability issues in microelectronic systems: status-quo and challenges. *Microelectronics Reliability*, 43(12):1969–1974, 2003.

[17] Texas Instruments. Flip chip ball grid array package reference guide. *Literature Number: SPRU811A*, 2005.

[18] Jeng-Liang Tsai, CC-P Chen, Guoqiang Chen, Brent Goplen, Haifeng Qian, Yong Zhan, Sung-Mo Kang, Martin DF Wong, and Sachin S Sapatnekar. Temperature-aware placement for socs. *Proceedings of the IEEE*, 94(8):1502–1518, 2006.

[19] Åbo Akademy University. Temperature graph over the last year. URL: http://at8.abo.fi/cgi-bin/en/Huge-T.

[20] Rajit Chandra. Full-chip transient temperature analysis, 2006. Invited talk, ROBUSPIC Workshop at ESSDERC'06 in Montreux.

[21] Luca Benini. Temperature awareness in digital systems design & management, 2013.

[22] Prof. J.M. Rabaey. Scaling the power wall. presentation given at Synopsys/Sun university reception, 2007.

[23] Prof. Hannu Tenhunen. personal communication, 2010.

[24] KW Guarini, AW Topol, M Ieong, R Yu, L Shi, MR Newport, DJ Frank, DV Singh, GM Cohen, SV Nitta, et al. Electrical integrity of state-of-the-art 0.13/spl mu/m soi cmos devices and circuits transferred for three-dimensional (3d) integrated circuit (ic) fabrication. In *Electron Devices Meeting, 2002. IEDM'02. International*, pages 943–945. IEEE, 2002.

[25] Kaustav Banerjee, Shukri J Souri, Pawan Kapur, and Krishna C Saraswat. 3-d ics: A novel chip design for improving deep-submicrometer interconnect performance and systems-on-chip integration. *Proceedings of the IEEE*, 89(5):602–633, 2001.

[26] AW Topol, DC La Tulipe, L Shi, DJ Frank, K Bernstein, SE Steen, A Kumar, GU Singco, AM Young, KW Guarini, et al. Three-dimensional integrated circuits. *IBM Journal of Research and Development*, 50(4.5):491–506, 2006.

[27] Brent Goplen and Sachin Sapatnekar. Efficient thermal placement of standard cells in 3d ics using a force directed approach. In *Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design*, page 86. IEEE Computer Society, 2003.

[28] Brent Goplen and Sachin Sapatnekar. Thermal via placement in 3d ics. In *Proceedings of the 2005 international symposium on Physical design*, pages 167–174. ACM, 2005.

[29] Sachin S Sapatnekar. Addressing thermal and power delivery bottlenecks in 3d circuits. In *Design Automation Conference, 2009. ASP-DAC 2009. Asia and South Pacific*, pages 423–428. IEEE, 2009.

[30] John H Lau. *Flip chip technologies*, volume 1. McGraw-Hill New York, 1996.

[31] David Atienza, Pablo G Del Valle, Giacomo Paci, Francesco Poletti, Luca Benini, Giovanni De Micheli, Jose M Mendias, and Roman Hermida. Hw-sw emulation framework for temperature-aware design in mpsocs. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 12(3):26, 2007.

[32] Ayse Coskun, Jie Meng, David Atienza, and Mohamed M Sabry. Attaining single-chip, high-performance computing through 3d systems with active cooling. *Micro, IEEE*, 31(4):63–75, 2011.

[33] Asst. Prof. David Atienza. personal communication, 2012.

[34] James Donald and Margaret Martonosi. Techniques for multicore thermal management: Classification and new exploration. *ACM SIGARCH Computer Architecture News*, 34(2):78–88, 2006.

[35] Li Shang, Li-Shiuan Peh, Amit Kumar, and Niraj K Jha. Thermal modeling, characterization and management of on-chip networks. In *Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture*, pages 67–78. IEEE Computer Society, 2004.

[36] Kameswar Rao Vaddina, Amir-Mohammad Rahmani, Mohammad Fattah, Pasi Liljeberg, and Juha Plosila. Design space exploration of thermal-aware many-core systems. *Journal of Systems Architecture*, 59(10):1197–1213, 2013.

[37] Kameswar Rao Vaddina, Pasi Liljeberg, and Juha Plosila. Exploration of temperature-aware placement approaches in 2d and 3d stacked systems. *International Journal of Adaptive, Resilient and Autonomic Systems (IJARAS)*, 4(3):61–81, 2013.

[38] A-M Rahmani, Khalid Latif, Kameswar Rao Vaddina, Pasi Liljeberg, Juha Plosila, and Hannu Tenhunen. Congestion aware, fault tolerant, and thermally efficient inter-layer communication scheme for hybrid noc-bus 3d architectures. In *Networks on Chip (NoCS), 2011 Fifth IEEE/ACM International Symposium on*, pages 65–72. IEEE, 2011.

[39] Kameswar Rao Vaddina, A Rahmani, Khalid Latif, Pasi Liljeberg, and Juha Plosila. Thermal analysis of job allocation and scheduling schemes for 3d stacked noc's. In *Digital System Design (DSD), 2011 14th Euromicro Conference on*, pages 643–648. IEEE, 2011.

[40] Kameswar Rao Vaddina, Tamoghna Mitra, Pasi Liljeberg, and Juha Plosila. Thermal modelling of 3d multicore systems in a flip-chip package. In *SOC Conference (SOCC), 2010 IEEE International*, pages 379–383. IEEE, 2010.

[41] Kameswar Rao Vaddina, Ethiopia Nigussie, Pasi Liljeberg, and Juha Plosila. Self-timed thermal sensing and monitoring of multicore systems. In *Design and Diagnostics of Electronic Circuits & Systems, 2009. DDECS'09. 12th International Symposium on*, pages 246–251. IEEE, 2009.

[42] Kameswar Rao Vaddina, Pasi Liljeberg, and Juha Plosila. Thermal analysis of on-chip interconnects in multicore systems. In *NORCHIP, 2009*, pages 1–4. IEEE, 2009.

[43] Michael John Sebastian Smith. *Application-Specific Integrated Circuits.* Addison-Wesley Professional, 1st edition, 2008.

[44] David Wolpert and Paul Ampadu. *Managing temperature effects in nanoscale adaptive systems*. Springer, 2012.

[45] Yingmin Li, Benjamin Lee, David Brooks, Zhigang Hu, and Kevin Skadron. Cmp design space exploration subject to physical constraints. In *High-Performance Computer Architecture, 2006. The Twelfth International Symposium on*, pages 17–28. IEEE, 2006.

[46] Mali Mahalingam. Thermal management in semiconductor device packaging. *Proceedings of the IEEE*, 73(9):1396–1404, 1985.

[47] Francisco Javier Mesa-Martinez, Ehsan K Ardestani, and Jose Renau. Characterizing processor thermal behavior. In *ACM SIGARCH Computer Architecture News*, volume 38, pages 193–204. ACM, 2010.

[48] Joonho Kong, Sung Woo Chung, and Kevin Skadron. Recent thermal management techniques for microprocessors. *ACM Computing Surveys (CSUR)*, 44(3):13, 2012.

[49] Kaustav Banerjee, Amit Mehrotra, Alberto Sangiovanni-Vincentelli, and Chenming Hu. On thermal effects in deep sub-micron vlsi interconnects. In *Proceedings of the 36th annual ACM/IEEE Design Automation Conference*, pages 885–891. ACM, 1999.

[50] Vasanth Venkatachalam and Michael Franz. Power reduction techniques for microprocessor systems. *ACM Computing Surveys (CSUR)*, 37(3):195–237, 2005.

[51] W Stubstad. The application of thermoelectric spot cooling to electronic equipment. *Product Engineering and Production, IRE Transactions on*, 5(4):22–29, 1961.

[52] D Pal and Y Joshi. Application of phase change materials for passive thermal control of plastic quad flat packages: a computational study. *Numerical Heat Transfer, Part A Applications*, 30(1):19–34, 1996.

[53] Amir H Ajami, Kaustav Banerjee, and Massoud Pedram. Modeling and analysis of nonuniform substrate temperature effects on global ulsi interconnects. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(6):849–861, 2005.

[54] Sheng Xu, Ibis Benito, and Wayne Burleson. Thermal impacts on noc interconnects. In *Networks-on-Chip, 2007. NOCS 2007. First International Symposium on*, pages 220–220. IEEE, 2007.

[55] Tianpei Zhang, Yong Zhan, and Sachin S Sapatnekar. Temperature-aware routing in 3d ics. In *Design Automation, 2006. Asia and South Pacific Conference on*, pages 6–pp. IEEE, 2006.

[56] Feng Wang, Michael De Bole, Xiaoxia Wu, Yuan Xie, Narayanan Vijaykrishnan, and Mary Jane Irwin. On-chip bus thermal analysis and optimisation. *Computers & Digital Techniques, IET*, 1(5):590–599, 2007.

[57] Karthik Sankaranarayanan, Sivakumar Velusamy, Mircea Stan, and Kevin Skadron. A case for thermal-aware floorplanning at the microarchitectural level. *Journal of Instruction-Level Parallelism*, 7(1):8–16, 2005.

[58] Yongkui Han, Israel Koren, and Csaba Andras Moritz. Temperature aware floorplanning. In *Workshop on Temperature Aware Computer Systems*, 2005.

[59] Amir H Ajami, Kaustav Banerjee, and Massoud Pedram. Analysis of substrate thermal gradient effects on optimal buffer insertion. In *Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design*, pages 44–48. IEEE Press, 2001.

[60] Ting-Yen Chiang, Kaustav Banerjee, and Krishna C Saraswat. Effect of via separation and low-k dielectric materials on the thermal characteristics of cu interconnects. In *Electron Devices Meeting, 2000. IEDM'00. Technical Digest. International*, pages 261–264. IEEE, 2000.

[61] Amir H Ajami. Thermal management takes center stage in ic design.

[62] Pedro Chaparro, José González, Grigorios Magklis, Cai Qiong, and Antonio González. Understanding the thermal implications of multicore architectures. *Parallel and Distributed Systems, IEEE Transactions on*, 18(8):1055–1065, 2007.

[63] James Tschanz, Nam Sung Kim, Saurabh Dighe, Jason Howard, Gregory Ruhl, S Vanga, Siva Narendra, Yatin Hoskote, Howard Wilson, Carol Lam, et al. Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 292–604. IEEE, 2007.

[64] David Wolpert and Paul Ampadu. Adaptive delay correction for runtime variation in dynamic voltage scaling systems. *Journal of Circuits, Systems, and Computers*, 17(06):1111–1128, 2008.

[65] Sebastian Herbert and Diana Marculescu. Variability-aware frequency scaling in multi-clock processors. In *Adaptive Techniques for Dynamic Processor Optimization*, pages 207–227. Springer, 2008.

[66] Le Yan, Jiong Luo, and Niraj K Jha. Joint dynamic voltage scaling and adaptive body biasing for heterogeneous distributed real-time embedded systems. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(7):1030–1041, 2005.

[67] Jayanth Srinivasan and Sarita V Adve. Predictive dynamic thermal management for multimedia applications. In *Proceedings of the 17th annual international conference on Supercomputing*, pages 109–120. ACM, 2003.

[68] David Brooks and Margaret Martonosi. Dynamic thermal management for high-performance microprocessors. In *High-Performance Computer Architecture, 2001. HPCA. The Seventh International Symposium on*, pages 171–182. IEEE, 2001.

[69] Ratnesh K Sharma, Cullen E Bash, Chandrakant D Patel, Richard J Friedrich, and Jeffrey S Chase. Balance of power: Dynamic thermal management for internet data centers. *Internet Computing, IEEE*, 9(1):42–49, 2005.

[70] Ayse Kivilcim Coskun, Tajana Simunic Rosing, and Keith Whisnant. Temperature aware task scheduling in mpsocs. In *Proceedings of the conference on Design, automation and test in Europe*, pages 1659–1664. EDA Consortium, 2007.

[71] Jun Yang, Xiuyi Zhou, Marek Chrobak, Youtao Zhang, and Lingling Jin. Dynamic thermal management through task scheduling. In *Performance Analysis of Systems and software, 2008. ISPASS 2008. IEEE International Symposium on*, pages 191–201. IEEE, 2008.

[72] Jeonghwan Choi, Chen-Yong Cher, Hubertus Franke, Henrdrik Hamann, Alan Weger, and Pradip Bose. Thermal-aware task scheduling at the system software level. In *Proceedings of the 2007 international symposium on Low power electronics and design*, pages 213–218. ACM, 2007.

[73] Li Shang, Li-Shiuan Peh, and Niraj K Jha. Dynamic voltage scaling with links for power optimization of interconnection networks. In *High-Performance Computer Architecture, 2003. HPCA-9 2003. Proceedings. The Ninth International Symposium on*, pages 91–102. IEEE, 2003.

[74] Yuan Xie and Wei-Lun Hung. Temperature-aware task allocation and scheduling for embedded multiprocessor systems-on-chip (mpsoc) design. *Journal of VLSI signal processing systems for signal, image and video technology*, 45(3):177–189, 2006.

[75] C.H. Chao, K.Y. Jheng, H.Y. Wang, J.C. Wu, and A.Y. Wu. Traffic- and thermal-aware run-time thermal management scheme for 3d noc systems. In *Networks-on-Chip (NOCS), 2010 Fourth ACM/IEEE International Symposium on*, pages 223–230. IEEE, 2010.

[76] Duo Li, Sheldon X-D Tan, Eduardo H Pacheco, and Murli Tirumala. Parameterized transient thermal behavioral modeling for chip multiprocessors. In *Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design*, pages 611–617. IEEE Press, 2008.

[77] Michael Kadin and Sherief Reda. Frequency planning for multi-core processors under thermal constraints. In *Low Power Electronics and Design (ISLPED), 2008 ACM/IEEE International Symposium on*, pages 213–216. IEEE, 2008.

[78] Yefu Wang, Kai Ma, and Xiaorui Wang. Temperature-constrained power control for chip multiprocessors with online model estimation. In *ACM SIGARCH Computer Architecture News*, volume 37, pages 314–324. ACM, 2009.

[79] Marius Marcu. Power–thermal profiling of software applications. *Microelectronics Journal*, 42(4):601–608, 2011.

[80] Amit Kumar, Li Shang, Li-Shiuan Peh, and Niraj K Jha. System-level dynamic thermal management for high-performance microprocessors. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 27(1):96–108, 2008.

[81] Xiliang Zhong and Cheng-Zhong Xu. Energy-aware modeling and scheduling of real-time tasks for dynamic voltage scaling. In *Real-Time Systems Symposium, 2005. RTSS 2005. 26th IEEE International*, pages 10–pp. IEEE, 2005.

[82] Peng Yang, Chun Wong, Paul Marchal, Francky Catthoor, Dirk Desmet, Diederik Verkest, and Rudy Lauwereins. Energy-aware runtime scheduling for embedded-multiprocessor socs. *IEEE Design & Test of Computers*, 18(5):46–58, 2001.

[83] Eren Kursun, Chen-Yong Cher, Alper Buyuktosunoglu, and Pradip Bose. Investigating the effects of task scheduling on thermal behavior. In *Third Workshop on Temperature-Aware Computer Systems (TACS'06)*, 2006.

[84] Inchoon Yeo, Chih Chun Liu, and Eun Jung Kim. Predictive dynamic thermal management for multicore systems. In *Proceedings of the 45th annual Design Automation Conference*, pages 734–739. ACM, 2008.

[85] Ayse Kivilcim Coskun, Tajana Simunic Rosing, and Kenny C Gross. Utilizing predictors for efficient thermal management in multiprocessor socs. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 28(10):1503–1516, 2009.

[86] Efraim Rotem, J Hermerding, A Cohen, and H Cain. Temperature measurement in the intel® coretm duo processor. 2006.

[87] Jieyi Long, Seda Ogrenci Memik, Gokhan Memik, and Rajarshi Mukherjee. Thermal monitoring mechanisms for chip multiprocessors. *ACM Transactions on Architecture and Code Optimization (TACO)*, 5(2):9, 2008.

[88] Rajarshi Mukherjee and Seda Ogrenci Memik. Systematic temperature sensor allocation and placement for microprocessors. In *Proceedings of the 43rd annual Design Automation Conference*, pages 542–547. ACM, 2006.

[89] Francesco Zanini, David Atienza, Colin N Jones, and Giovanni De Micheli. Temperature sensor placement in thermal management systems for mpsocs. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 1065–1068. IEEE, 2010.

[90] Ron Kalla, Balaram Sinharoy, William J Starke, and Michael Floyd. Power7: Ibm's next-generation server processor. *Micro, IEEE*, 30(2):7–15, 2010.

[91] M Sadri, Andrea Bartolini, and Luca Benini. Single-chip cloud computer thermal model. In *Thermal Investigations of ICs and Systems (THERMINIC), 2011 17th International Workshop on*, pages 1–6. IEEE, 2011.

[92] Anton Bakker. Cmos smart temperature sensors-an overview. In *Sensors, 2002. Proceedings of IEEE*, volume 2, pages 1423–1427. IEEE, 2002.

[93] Gerard CM Meijer, Guijie Wang, and Fabiano Fruett. Temperature sensors and voltage references implemented in cmos technology. *IEEE Sensors Journal, Vol. 1*, (3):225–234, 2001.

[94] Poki Chen, Chun-Chi Chen, Chin-Chung Tsai, and Wen-Fu Lu. A time-to-digital-converter-based cmos smart temperature sensor. *Solid-State Circuits, IEEE Journal of*, 40(8):1642–1648, 2005.

[95] Poki Chen, Mon-Chau Shie, Zhi-Yuan Zheng, Zi-Fan Zheng, and Chun-Yan Chu. A fully digital time-domain smart temperature sensor realized with 140 fpga logic elements. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 54(12):2661–2668, 2007.

[96] Qikai Chen, Mesut Meterelliyoz, and Kaushik Roy. A cmos thermal sensor and its applications in temperature adaptive design. In *Proceedings of the 7th International Symposium on Quality Electronic Design*, pages 243–248. IEEE Computer Society, 2006.

[97] Michiel AP Pertijs, Kofi AA Makinwa, and Johan H Huijsing. A cmos smart temperature sensor with a 3σ inaccuracy of$\pm$0.1 c from-55 c to 125 c. *Solid-State Circuits, IEEE Journal of*, 40(12):2805–2815, 2005.

[98] Anton Bakker and Johan H Huijsing. Micropower cmos temperature sensor with digital output. *Solid-State Circuits, IEEE Journal of*, 31(7):933–937, 1996.

[99] Pablo Ituero, José L Ayala, and Marisa Lopez-Vallejo. A nanowatt smart temperature sensor for dynamic thermal management. *Sensors Journal, IEEE*, 8(12):2036–2043, 2008.

[100] K. Skadron, M.R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan. Temperature-aware microarchitecture. In *ACM SIGARCH Computer Architecture News*, volume 31, pages 2–13. ACM, 2003.

[101] Kevin Skadron, Mircea R. Stan, Wei Huang, Sivakumar Velusamy, Karthik Sankaranarayanan, and David Tarjan. Temperature-aware microarchitecture: Extended discussion and results. In *In Proceedings of the 30th Annual International Symposium on Computer Architecture*, pages 2–13, 2003.

[102] Ting-Yuan Wang and CC Chen. 3-d thermal-adi: A linear-time chip level transient thermal simulator. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 21(12):1434–1445, 2002.

[103] Clemens JM Lasance. Two benchmarks to facilitate the study of compact thermal modeling phenomena. *Components and Packaging Technologies, IEEE Transactions on*, 24(4):559–565, 2001.

[104] William Batty, Carlos E Christoffersen, Alexander B Yakovlev, John F Whitaker, Amir Mortazawi, Ayman Al-Zayed, Mete Ozkar, Sean C

Ortiz, Ronald M Reano, Kyoung Yang, et al. Global coupled em-electrical-thermal simulation and experimental validation for a spatial power combining mmic array. *Microwave Theory and Techniques, IEEE Transactions on*, 50(12):2820–2833, 2002.

[105] Wei Huang, Mircea R Stan, Kevin Skadron, Karthik Sankaranarayanan, Shougata Ghosh, and Sivakumar Velusam. Compact thermal modeling for temperature-aware design. In *Proceedings of the 41st annual Design Automation Conference*, pages 878–883. ACM, 2004.

[106] Wei Huang, Shougata Ghosh, Sivakumar Velusamy, Karthik Sankaranarayanan, Kevin Skadron, and Mircea R Stan. Hotspot: A compact thermal modeling methodology for early-stage vlsi design. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 14(5):501–513, 2006.

[107] Wei Huang, Karthik Sankaranarayanan, Kevin Skadron, Robert J Ribando, and Mircea R Stan. Accurate, pre-rtl temperature-aware design using a parameterized, geometric thermal model. *Computers, IEEE Transactions on*, 57(9):1277–1288, 2008.

[108] Bryan Black, Murali Annavaram, Ned Brekelbaum, John DeVale, Lei Jiang, Gabriel H Loh, Don McCauley, Pat Morrow, Donald W Nelson, Daniel Pantuso, et al. Die stacking (3d) microarchitecture. In *Microarchitecture, 2006. MICRO-39. 39th Annual IEEE/ACM International Symposium on*, pages 469–479. IEEE, 2006.

[109] Moongon Jung, Taigon Song, Yang Wan, Young-Joon Lee, Debabrata Mohapatra, Hong Wang, Greg Taylor, Devang Jariwala, Vijay Pitchumani, Patrick Morrow, et al. How to reduce power in 3d ic designs: A case study with opensparc t2 core. In *Custom Integrated Circuits Conference (CICC), 2013 IEEE*, pages 1–4. IEEE, 2013.

[110] Mohit Pathak, Young-Joon Lee, Thomas Moon, and Sung Kyu Lim. Through-silicon-via management during 3d physical design: When to add and how many? In *Proceedings of the International Conference on Computer-Aided Design*, pages 387–394. IEEE Press, 2010.

[111] Jason Cong and Yan Zhang. Thermal via planning for 3-d ics. In *Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, pages 745–752. IEEE, 2005.

[112] Xin Li, Yuchun Ma, Xianlong Hong, Sheqin Dong, and Jason Cong. Lp based white space redistribution for thermal via planning and performance optimization in 3d ics. In *Design Automation Conference,*

2008. ASPDAC 2008. Asia and South Pacific, pages 209–212. IEEE, 2008.

[113] Santhosh Onkaraiah and Chuan Seng Tan. Mitigating heat dissipation and thermo-mechanical stress challenges in 3-d ic using thermal through silicon via (ttsv). In *Electronic Components and Technology Conference (ECTC), 2010 Proceedings 60th*, pages 411–416. IEEE, 2010.

[114] Po-Yang Hsu, Hsien-Te Chen, and TingTing Hwang. Stacking signal tsv for thermal dissipation in global routing for 3d ic. In *Design Automation Conference (ASP-DAC), 2013 18th Asia and South Pacific*, pages 699–704, Jan 2013.

[115] K-J Lee and Kevin Skadron. Using performance counters for runtime temperature sensing in high-performance processors. In *Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International*, pages 8–pp. IEEE, 2005.

[116] Ke Meng, Russ Joseph, Robert P Dick, and Li Shang. Multi-optimization power management for chip multiprocessors. In *Proceedings of the 17th international conference on Parallel architectures and compilation techniques*, pages 177–186. ACM, 2008.

[117] Michael Huang, Jose Renau, Seung-Moon Yoo, and Josep Torrellas. The design of deetm: a framework for dynamic energy efficiency and temperature management. *Journal of Instruction-Level Parallelism*, 3:1–31, 2002.

[118] Alice Wang and Anantha Chandrakasan. A 180-mv subthreshold fft processor using a minimum energy design methodology. *Solid-State Circuits, IEEE Journal of*, 40(1):310–319, 2005.

[119] Jayanth Srinivasan, Sarita V Adve, Pradip Bose, and Jude A Rivers. The case for lifetime reliability-aware microprocessors. In *ACM SIGARCH Computer Architecture News*, volume 32, page 276. IEEE Computer Society, 2004.

[120] Narinder Pal Singh. A design methodology for self-time systems. 1981.

[121] Puyan Dadvar and Kevin Skadron. Potential thermal security risks. In *Semiconductor Thermal Measurement and Management Symposium, 2005 IEEE Twenty First Annual IEEE*, pages 229–234. IEEE, 2005.

[122] Intel Corp. Mobile intel pentium 4 processor-m: Datasheet, 2003.

[123] Pasi Liljeberg. *On self-timed communication architectures for network-on-chip*. PhD thesis, University of Turku, nov 2005.

[124] Mohamed A Elgamel and Magdy A Bayoumi. *Interconnect noise optimization in nanometer technologies*. Springer Science+ Business Media, 2006.

[125] Meeta S Gupta, Jarod L Oatley, Russ Joseph, Gu-Yeon Wei, and David M Brooks. Understanding voltage variations in chip multiprocessors using a distributed power-delivery network. In *Design, Automation & Test in Europe Conference & Exhibition, 2007. DATE'07*, pages 1–6. IEEE, 2007.

[126] Hangsheng Wang, Li-Shiuan Peh, and Sharad Malik. Power-driven design of router microarchitectures in on-chip networks. In *Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture*, page 105. IEEE Computer Society, 2003.

[127] Ting-Yen Chiang, Kaustav Banerjee, and Krishna C Saraswat. Analytical thermal model for multilevel vlsi interconnects incorporating via effect. *Electron Device Letters, IEEE*, 23(1):31–33, 2002.

[128] Aveek Bid, Achyut Bora, and AK Raychaudhuri. Temperature dependence of the resistance of metallic nanowires of diameter $\geq$ 15nm: Applicability of bloch-grüneisen theorem. *Physical Review B*, 74(3):035426, 2006.

[129] Qiaojian Huang, Carmen M Lilley, Matthias Bode, and Ralu S Divan. Electrical properties of cu nanowires. In *Nanotechnology, 2008. NANO'08. 8th IEEE Conference on*, pages 549–552. IEEE, 2008.

[130] Danqing Chen, Erhong Li, Elyse Rosenbaum, and Sung-Mo Kang. Interconnect thermal modeling for accurate simulation of circuit timing and reliability. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 19(2):197–205, 2000.

[131] Sven Rzepka, Kaustav Banerjee, Ekkehard Meusel, and Chenming Hu. Characterization of self-heating in advanced vlsi interconnect lines based on thermal finite element simulation. *Components, Packaging, and Manufacturing Technology, Part A, IEEE Transactions on*, 21(3):406–411, 1998.

[132] Dinesh Pamunuwa and Hannu Tenhunen. Repeater insertion to minimise delay in coupled interconnects. In *VLSI Design, 2001. Fourteenth International Conference on*, pages 513–517. IEEE, 2001.

[133] Sriram R Vangal, Jason Howard, Gregory Ruhl, Saurabh Dighe, Howard Wilson, James Tschanz, David Finan, Arvind Singh, Tiju Jacob, Shailendra Jain, et al. An 80-tile sub-100-w teraflops processor in 65-nm cmos. *Solid-State Circuits, IEEE Journal of*, 43(1):29–41, 2008.

[134] Ankur Jain, Robert E Jones, Ritwik Chatterjee, and Scott Pozder. Analytical and numerical modeling of the thermal performance of three-dimensional integrated circuits. *Components and Packaging Technologies, IEEE Transactions on*, 33(1):56–63, 2010.

[135] Sungjun Im and Kaustav Banerjee. Full chip thermal analysis of planar (2-d) and vertically integrated (3-d) high performance ics. In *Electron Devices Meeting, 2000. IEDM'00. Technical Digest. International*, pages 727–730. IEEE, 2000.

[136] Changyun Zhu, Zhenyu Gu, Li Shang, Robert P Dick, and Russ Joseph. Three-dimensional chip-multiprocessor run-time thermal management. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 27(8):1479–1492, 2008.

[137] Luca P Carloni, Partha Pande, and Yuan Xie. Networks-on-chip in emerging interconnect paradigms: Advantages and challenges. In *Proceedings of the 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip*, pages 93–102. IEEE Computer Society, 2009.

[138] Jongman Kim, Chrysostomos Nicopoulos, Dongkook Park, Reetuparna Das, Yuan Xie, Vijaykrishnan Narayanan, Mazin S Yousif, and Chita R Das. A novel dimensionally-decomposed router for on-chip communication in 3 d architectures. *ACM SIGARCH Computer Architecture News*, 35(2):138–149, 2007.

[139] Feihui Li, Chrysostomos Nicopoulos, Thomas Richardson, Yuan Xie, Vijaykrishnan Narayanan, and Mahmut Kandemir. Design and management of 3d chip multiprocessors using network-in-memory. *ACM SIGARCH Computer Architecture News*, 34(2):130–141, 2006.

[140] A.M. Rahmani, K. Latif, P. Liljeberg, J. Plosila, and H. Tenhunen. A stacked mesh 3d noc architecture enabling congestion-aware and reliable inter-layer communication. In *Parallel, Distributed and Network-Based Processing (PDP), 2011 19th Euromicro International Conference on*, pages 423–430. IEEE, 2011.

[141] A.M. Rahmani, P. Liljeberg, J. Plosila, and H. Tenhunen. Exploring a low-cost and power-efficient hybridization technique for 3d noc-bus

hybrid architecture using lastz-based routing algorithms. *Journal of Low Power Electronics*, 8(4):403–414, 2012.

[142] Khalid Latif, A Rahmani, Kameswar Rao Vaddina, Tiberiu Seceleanu, Pasi Liljeberg, and Hannu Tenhunen. Enhancing performance of noc-based architectures using heuristic virtual-channel sharing approach. In *Computer Software and Applications Conference (COMPSAC), 2011 IEEE 35th Annual*, pages 442–447. IEEE, 2011.

[143] Khalid Latif, A-M Rahmani, Tiberiu Seceleanu, and Hannu Tenhunen. Power-and performance-aware ip mapping for noc-based mpsoc platforms. In *Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on*, pages 758–761. IEEE, 2010.

[144] Embedded Microprocessor Benchmark Consortium et al. EEMBC benchmark suite, 2009.

[145] A-M Rahmani, Pasi Liljeberg, Juha Plosila, and Hannu Tenhunen. Bbvc-3d-noc: An efficient 3d noc architecture using bidirectional bisynchronous vertical channels. In *VLSI (ISVLSI), 2010 IEEE Computer Society Annual Symposium on*, pages 452–453. IEEE, 2010.

[146] Guilherme Guindani, Cezar Reinbrecht, Thiago Raupp, Ney Calazans, and Fernando Gehm Moraes. Noc power estimation at the rtl abstraction level. In *Symposium on VLSI, 2008. ISVLSI'08. IEEE Computer Society Annual*, pages 475–478. IEEE, 2008.

[147] COMSOL Multiphysics. Comsol. *Inc., Burlington, MA, www. comsol. com*, 1994.

[148] B. Yang, L. Guang, T. Säntti, and J. Plosila. Mapping multiple applications with unbounded and bounded number of cores on many-core networks-on-chip. *Microprocessors and Microsystems*, 2012.

[149] M. Fattah, M. Ramirez, M. Daneshtalab, P. Liljeberg, and J. Plosila. Cona: Dynamic application mapping for congestion reduction in many-core systems. In *Computer Design (ICCD), 2012 IEEE 30th International Conference on*, pages 364–370. IEEE, 2012.

[150] F. Fazzino, M. Palesi, and D. Patti. Noxim: Network-on-chip simulator. *URL: http://sourceforge.net/projects/noxim [24.06. 2008]*, 2008.

Turku Centre for Computer Science
TUCS Dissertations

1. **Marjo Lipponen**, On Primitive Solutions of the Post Correspondence Problem
2. **Timo Käkölä**, Dual Information Systems in Hyperknowledge Organizations
3. **Ville Leppänen**, Studies on the Realization of PRAM
4. **Cunsheng Ding**, Cryptographic Counter Generators
5. **Sami Viitanen**, Some New Global Optimization Algorithms
6. **Tapio Salakoski**, Representative Classification of Protein Structures
7. **Thomas Långbacka**, An Interactive Environment Supporting the Development of Formally Correct Programs
8. **Thomas Finne**, A Decision Support System for Improving Information Security
9. **Valeria Mihalache**, Cooperation, Communication, Control. Investigations on Grammar Systems.
10. **Marina Waldén**, Formal Reasoning About Distributed Algorithms
11. **Tero Laihonen**, Estimates on the Covering Radius When the Dual Distance is Known
12. **Lucian Ilie**, Decision Problems on Orders of Words
13. **Jukkapekka Hekanaho**, An Evolutionary Approach to Concept Learning
14. **Jouni Järvinen**, Knowledge Representation and Rough Sets
15. **Tomi Pasanen**, In-Place Algorithms for Sorting Problems
16. **Mika Johnsson**, Operational and Tactical Level Optimization in Printed Circuit Board Assembly
17. **Mats Aspnäs**, Multiprocessor Architecture and Programming: The Hathi-2 System
18. **Anna Mikhajlova**, Ensuring Correctness of Object and Component Systems
19. **Vesa Torvinen**, Construction and Evaluation of the Labour Game Method
20. **Jorma Boberg**, Cluster Analysis. A Mathematical Approach with Applications to Protein Structures
21. **Leonid Mikhajlov**, Software Reuse Mechanisms and Techniques: Safety Versus Flexibility
22. **Timo Kaukoranta**, Iterative and Hierarchical Methods for Codebook Generation in Vector Quantization
23. **Gábor Magyar**, On Solution Approaches for Some Industrially Motivated Combinatorial Optimization Problems
24. **Linas Laibinis**, Mechanised Formal Reasoning About Modular Programs
25. **Shuhua Liu**, Improving Executive Support in Strategic Scanning with Software Agent Systems
26. **Jaakko Järvi**, New Techniques in Generic Programming – C++ is more Intentional than Intended
27. **Jan-Christian Lehtinen**, Reproducing Kernel Splines in the Analysis of Medical Data
28. **Martin Büchi**, Safe Language Mechanisms for Modularization and Concurrency
29. **Elena Troubitsyna**, Stepwise Development of Dependable Systems
30. **Janne Näppi**, Computer-Assisted Diagnosis of Breast Calcifications
31. **Jianming Liang**, Dynamic Chest Images Analysis
32. **Tiberiu Seceleanu**, Systematic Design of Synchronous Digital Circuits
33. **Tero Aittokallio**, Characterization and Modelling of the Cardiorespiratory System in Sleep-Disordered Breathing
34. **Ivan Porres**, Modeling and Analyzing Software Behavior in UML
35. **Mauno Rönkkö**, Stepwise Development of Hybrid Systems
36. **Jouni Smed**, Production Planning in Printed Circuit Board Assembly
37. **Vesa Halava**, The Post Correspondence Problem for Market Morphisms
38. **Ion Petre**, Commutation Problems on Sets of Words and Formal Power Series
39. **Vladimir Kvassov**, Information Technology and the Productivity of Managerial Work
40. **Frank Tétard**, Managers, Fragmentation of Working Time, and Information Systems

41. **Jan Manuch**, Defect Theorems and Infinite Words
42. **Kalle Ranto**, Z_4-Goethals Codes, Decoding and Designs
43. **Arto Lepistö**, On Relations Between Local and Global Periodicity
44. **Mika Hirvensalo**, Studies on Boolean Functions Related to Quantum Computing
45. **Pentti Virtanen**, Measuring and Improving Component-Based Software Development
46. **Adekunle Okunoye**, Knowledge Management and Global Diversity – A Framework to Support Organisations in Developing Countries
47. **Antonina Kloptchenko**, Text Mining Based on the Prototype Matching Method
48. **Juha Kivijärvi**, Optimization Methods for Clustering
49. **Rimvydas Rukšėnas**, Formal Development of Concurrent Components
50. **Dirk Nowotka**, Periodicity and Unbordered Factors of Words
51. **Attila Gyenesei**, Discovering Frequent Fuzzy Patterns in Relations of Quantitative Attributes
52. **Petteri Kaitovaara**, Packaging of IT Services – Conceptual and Empirical Studies
53. **Petri Rosendahl**, Niho Type Cross-Correlation Functions and Related Equations
54. **Péter Majlender**, A Normative Approach to Possibility Theory and Soft Decision Support
55. **Seppo Virtanen**, A Framework for Rapid Design and Evaluation of Protocol Processors
56. **Tomas Eklund**, The Self-Organizing Map in Financial Benchmarking
57. **Mikael Collan**, Giga-Investments: Modelling the Valuation of Very Large Industrial Real Investments
58. **Dag Björklund**, A Kernel Language for Unified Code Synthesis
59. **Shengnan Han**, Understanding User Adoption of Mobile Technology: Focusing on Physicians in Finland
60. **Irina Georgescu**, Rational Choice and Revealed Preference: A Fuzzy Approach
61. **Ping Yan**, Limit Cycles for Generalized Liénard-Type and Lotka-Volterra Systems
62. **Joonas Lehtinen**, Coding of Wavelet-Transformed Images
63. **Tommi Meskanen,** On the NTRU Cryptosystem
64. **Saeed Salehi**, Varieties of Tree Languages
65. **Jukka Arvo**, Efficient Algorithms for Hardware-Accelerated Shadow Computation
66. **Mika Hirvikorpi**, On the Tactical Level Production Planning in Flexible Manufacturing Systems
67. **Adrian Costea**, Computational Intelligence Methods for Quantitative Data Mining
68. **Cristina Seceleanu**, A Methodology for Constructing Correct Reactive Systems
69. **Luigia Petre**, Modeling with Action Systems
70. **Lu Yan**, Systematic Design of Ubiquitous Systems
71. **Mehran Gomari**, On the Generalization Ability of Bayesian Neural Networks
72. **Ville Harkke**, Knowledge Freedom for Medical Professionals – An Evaluation Study of a Mobile Information System for Physicians in Finland
73. **Marius Cosmin Codrea**, Pattern Analysis of Chlorophyll Fluorescence Signals
74. **Aiying Rong**, Cogeneration Planning Under the Deregulated Power Market and Emissions Trading Scheme
75. **Chihab BenMoussa**, Supporting the Sales Force through Mobile Information and Communication Technologies: Focusing on the Pharmaceutical Sales Force
76. **Jussi Salmi**, Improving Data Analysis in Proteomics
77. **Orieta Celiku**, Mechanized Reasoning for Dually-Nondeterministic and Probabilistic Programs
78. **Kaj-Mikael Björk**, Supply Chain Efficiency with Some Forest Industry Improvements
79. **Viorel Preoteasa**, Program Variables – The Core of Mechanical Reasoning about Imperative Programs
80. **Jonne Poikonen**, Absolute Value Extraction and Order Statistic Filtering for a Mixed-Mode Array Image Processor
81. **Luka Milovanov**, Agile Software Development in an Academic Environment
82. **Francisco Augusto Alcaraz Garcia**, Real Options, Default Risk and Soft Applications
83. **Kai K. Kimppa**, Problems with the Justification of Intellectual Property Rights in Relation to Software and Other Digitally Distributable Media
84. **Dragoş Truşcan**, Model Driven Development of Programmable Architectures
85. **Eugen Czeizler**, The Inverse Neighborhood Problem and Applications of Welch Sets in Automata Theory

86. **Sanna Ranto**, Identifying and Locating-Dominating Codes in Binary Hamming Spaces
87. **Tuomas Hakkarainen**, On the Computation of the Class Numbers of Real Abelian Fields
88. **Elena Czeizler**, Intricacies of Word Equations
89. **Marcus Alanen**, A Metamodeling Framework for Software Engineering
90. **Filip Ginter**, Towards Information Extraction in the Biomedical Domain: Methods and Resources
91. **Jarkko Paavola**, Signature Ensembles and Receiver Structures for Oversaturated Synchronous DS-CDMA Systems
92. **Arho Virkki**, The Human Respiratory System: Modelling, Analysis and Control
93. **Olli Luoma**, Efficient Methods for Storing and Querying XML Data with Relational Databases
94. **Dubravka Ilić**, Formal Reasoning about Dependability in Model-Driven Development
95. **Kim Solin**, Abstract Algebra of Program Refinement
96. **Tomi Westerlund**, Time Aware Modelling and Analysis of Systems-on-Chip
97. **Kalle Saari**, On the Frequency and Periodicity of Infinite Words
98. **Tomi Kärki**, Similarity Relations on Words: Relational Codes and Periods
99. **Markus M. Mäkelä**, Essays on Software Product Development: A Strategic Management Viewpoint
100. **Roope Vehkalahti**, Class Field Theoretic Methods in the Design of Lattice Signal Constellations
101. **Anne-Maria Ernvall-Hytönen**, On Short Exponential Sums Involving Fourier Coefficients of Holomorphic Cusp Forms
102. **Chang Li**, Parallelism and Complexity in Gene Assembly
103. **Tapio Pahikkala**, New Kernel Functions and Learning Methods for Text and Data Mining
104. **Denis Shestakov**, Search Interfaces on the Web: Querying and Characterizing
105. **Sampo Pyysalo**, A Dependency Parsing Approach to Biomedical Text Mining
106. **Anna Sell**, Mobile Digital Calendars in Knowledge Work
107. **Dorina Marghescu**, Evaluating Multidimensional Visualization Techniques in Data Mining Tasks
108. **Tero Säntti**, A Co-Processor Approach for Efficient Java Execution in Embedded Systems
109. **Kari Salonen**, Setup Optimization in High-Mix Surface Mount PCB Assembly
110. **Pontus Boström**, Formal Design and Verification of Systems Using Domain-Specific Languages
111. **Camilla J. Hollanti**, Order-Theoretic Mehtods for Space-Time Coding: Symmetric and Asymmetric Designs
112. **Heidi Himmanen**, On Transmission System Design for Wireless Broadcasting
113. **Sébastien Lafond**, Simulation of Embedded Systems for Energy Consumption Estimation
114. **Evgeni Tsivtsivadze**, Learning Preferences with Kernel-Based Methods
115. **Petri Salmela**, On Commutation and Conjugacy of Rational Languages and the Fixed Point Method
116. **Siamak Taati**, Conservation Laws in Cellular Automata
117. **Vladimir Rogojin**, Gene Assembly in Stichotrichous Ciliates: Elementary Operations, Parallelism and Computation
118. **Alexey Dudkov**, Chip and Signature Interleaving in DS CDMA Systems
119. **Janne Savela**, Role of Selected Spectral Attributes in the Perception of Synthetic Vowels
120. **Kristian Nybom**, Low-Density Parity-Check Codes for Wireless Datacast Networks
121. **Johanna Tuominen**, Formal Power Analysis of Systems-on-Chip
122. **Teijo Lehtonen**, On Fault Tolerance Methods for Networks-on-Chip
123. **Eeva Suvitie**, On Inner Products Involving Holomorphic Cusp Forms and Maass Forms
124. **Linda Mannila**, Teaching Mathematics and Programming – New Approaches with Empirical Evaluation
125. **Hanna Suominen**, Machine Learning and Clinical Text: Supporting Health Information Flow
126. **Tuomo Saarni**, Segmental Durations of Speech
127. **Johannes Eriksson**, Tool-Supported Invariant-Based Programming

128. **Tero Jokela**, Design and Analysis of Forward Error Control Coding and Signaling for Guaranteeing QoS in Wireless Broadcast Systems
129. **Ville Lukkarila**, On Undecidable Dynamical Properties of Reversible One-Dimensional Cellular Automata
130. **Qaisar Ahmad Malik**, Combining Model-Based Testing and Stepwise Formal Development
131. **Mikko-Jussi Laakso**, Promoting Programming Learning: Engagement, Automatic Assessment with Immediate Feedback in Visualizations
132. **Riikka Vuokko**, A Practice Perspective on Organizational Implementation of Information Technology
133. **Jeanette Heidenberg**, Towards Increased Productivity and Quality in Software Development Using Agile, Lean and Collaborative Approaches
134. **Yong Liu**, Solving the Puzzle of Mobile Learning Adoption
135. **Stina Ojala**, Towards an Integrative Information Society: Studies on Individuality in Speech and Sign
136. **Matteo Brunelli**, Some Advances in Mathematical Models for Preference Relations
137. **Ville Junnila**, On Identifying and Locating-Dominating Codes
138. **Andrzej Mizera**, Methods for Construction and Analysis of Computational Models in Systems Biology. Applications to the Modelling of the Heat Shock Response and the Self-Assembly of Intermediate Filaments.
139. **Csaba Ráduly-Baka**, Algorithmic Solutions for Combinatorial Problems in Resource Management of Manufacturing Environments
140. **Jari Kyngäs**, Solving Challenging Real-World Scheduling Problems
141. **Arho Suominen**, Notes on Emerging Technologies
142. **József Mezei**, A Quantitative View on Fuzzy Numbers
143. **Marta Olszewska**, On the Impact of Rigorous Approaches on the Quality of Development
144. **Antti Airola**, Kernel-Based Ranking: Methods for Learning and Performace Estimation
145. **Aleksi Saarela**, Word Equations and Related Topics: Independence, Decidability and Characterizations
146. **Lasse Bergroth**, Kahden merkkijonon pisimmän yhteisen alijonon ongelma ja sen ratkaiseminen
147. **Thomas Canhao Xu**, Hardware/Software Co-Design for Multicore Architectures
148. **Tuomas Mäkilä**, Software Development Process Modeling – Developers Perspective to Contemporary Modeling Techniques
149. **Shahrokh Nikou**, Opening the Black-Box of IT Artifacts: Looking into Mobile Service Characteristics and Individual Perception
150. **Alessandro Buoni**, Fraud Detection in the Banking Sector: A Multi-Agent Approach
151. **Mats Neovius**, Trustworthy Context Dependency in Ubiquitous Systems
152. **Fredrik Degerlund**, Scheduling of Guarded Command Based Models
153. **Amir-Mohammad Rahmani-Sane**, Exploration and Design of Power-Efficient Networked Many-Core Systems
154. **Ville Rantala**, On Dynamic Monitoring Methods for Networks-on-Chip
155. **Mikko Pelto**, On Identifying and Locating-Dominating Codes in the Infinite King Grid
156. **Anton Tarasyuk**, Formal Development and Quantitative Verification of Dependable Systems
157. **Muhammad Mohsin Saleemi**, Towards Combining Interactive Mobile TV and Smart Spaces: Architectures, Tools and Application Development
158. **Tommi J. M. Lehtinen**, Numbers and Languages
159. **Peter Sarlin**, Mapping Financial Stability
160. **Alexander Wei Yin**, On Energy Efficient Computing Platforms
161. **Mikołaj Olszewski**, Scaling Up Stepwise Feature Introduction to Construction of Large Software Systems
162. **Maryam Kamali**, Reusable Formal Architectures for Networked Systems
163. **Zhiyuan Yao**, Visual Customer Segmentation and Behavior Analysis – A SOM-Based Approach
164. **Timo Jolivet**, Combinatorics of Pisot Substitutions
165. **Rajeev Kumar Kanth**, Analysis and Life Cycle Assessment of Printed Antennas for Sustainable Wireless Systems
166. **Khalid Latif**, Design Space Exploration for MPSoC Architectures

167. **Bo Yang**, Towards Optimal Application Mapping for Energy-Efficient Many-Core Platforms
168. **Ali Hanzala Khan**, Consistency of UML Based Designs Using Ontology Reasoners
169. **Sonja Leskinen**, m-Equine: IS Support for the Horse Industry
170. **Fareed Ahmed Jokhio**, Video Transcoding in a Distributed Cloud Computing Environment
171. **Moazzam Fareed Niazi**, A Model-Based Development and Verification Framework for Distributed System-on-Chip Architecture
172. **Mari Huova**, Combinatorics on Words: New Aspects on Avoidability, Defect Effect, Equations and Palindromes
173. **Ville Timonen**, Scalable Algorithms for Height Field Illumination
174. **Henri Korvela**, Virtual Communities – A Virtual Treasure Trove for End-User Developers
175. **Kameswar Rao Waddina**, Thermal-Aware Networked Many-Core Systems

Turku Centre *for* Computer Science

Joukahaisenkatu 3-5 B, 20520 Turku, Finland | www. tucs.fi

University of Turku
Faculty of Mathematics and Natural Sciences
- Department of Information Technology
- Department of Mathematics and Statistics

Turku School of Economics
- Institute of Information Systems Science

Åbo Akademi University
Division for Natural Sciences and Technology
- Department of Information Technologies

ISBN 978-952-12-3063-9
ISSN 1239-1883

Kameswar Rao Vaddina

Thermal-Aware Networked Many-Core Systems

Kameswar Rao Vaddina

Thermal-Aware Networked Many-Core Systems