

Thermal Measurements & Characterizations of Real Processors

Honors Thesis Submitted by

Shiqing, Poh

In partial fulfillment of the

Sc.B. In Electrical Engineering

Brown University

Prepared under the Direction of

Professor Sherief Reda

Professor William Patterson, Reader

Professor Jennifer Dworak, Reader

Table of Contents

Abstract	5
Chapter 1: Introduction	6
1.1 Thermal Characterization of Processors	6
1.2 Research Overview	7
Chapter 2: Previous Related Work	11
Chapter 3: Thermal Models & Methods	16
3.1 The HotSpot Tool	16
3.2 How It Works	16
3.3 Derivation of RC Model	18
3.4 Integrating Thermal Model in HotSpot	21
3.5 Validation of Model	22
3.6 Understanding Thermal Modeling during Design-time	23
3.7 Thermal Modeling in FinFETs	24
3.8 Validating RC Thermal Models for FinFETS	25
3.9 Background of CNTs	32
3.10 Extension of RC Model into CNT FinFET Device	35
Chapter 4: Thermal Measurements of Real Processors under Diff. Configs	39
4.1 Overview of Following Experiments	39
4.2 Thermal Measurement Setup	40
4.3 Experiment 1: Infrared Camera vs Thermal Sensors	43
4.4 Experiment 2: Oil Heat Sink vs Metal Heat Sink	51
Chapter 5: Workload Thermal Behavior in Oil Heat Sink	60
5.1 Overview of Experiments	60
5.2 Experimental Challenges	61
5.3 Experiment 1: Characterizing HotSpot Locus	62
5.4 Experiment 2: Statistics of Thermal Behaviors of Diff. Benchmarks	72
5.5 Background on Variogram	75
5.6 Experiment 3: Variogram of Various Benchmarks	78

Chapter 6: Conclusions	86
6.1 Conclusions of Experimental Results	86
6.2 Future Directions	88
References	89

Abstract

Effective design and evaluation of thermal management techniques require dependable forms of thermal characterization. However, the difficulty of runtime thermal characterization of a processor is contributed by a number of factors such as the inaccuracy of thermal sensors and inaccessibility of thermal data on a processor while using a thermal camera. Beginning with an understanding of thermal modeling using the HotSpot tool, I simulated RC thermal models of a FinFET device and validated the model with results. My work involves the thermal measurements of real running processors in an oil-sink setup using both an infrared camera and the processor thermal sensors and comparing the thermal behaviors via the two methods of measurement. I demonstrate that the on-chip thermal sensors correlate directly with IR imaging with slight systemic differences. This means that thermal sensors on which DTM techniques depend are very effective. In addition, I also compared the thermal behaviors of the processor in the oil sink and air sink configurations using the thermal sensors and found that the thermal behavior in the oil cooling configuration has a quicker long-term (gradual, low frequency variation) transient response and a slower short-term (high frequency variation) transient response than that of the air sink configuration. Finally, I will characterize the thermal behavior of various workloads with the infrared camera in an oil sink configuration, generating the hotspot loci, statistics and frequency domains of each benchmark and observe that there are some differences among different workloads.

Chapter 1: Introduction

1.1 Thermal Characterization of Processors

Elevated temperatures have a negative impact on processor performance. Leakage current constitutes a major portion of total power dissipation in processors and this leakage current varies exponentially with temperature. Additionally, higher operating temperature and increased power density can also decrease the reliability of logic and timing on the chip due to the increase in system parameter variability (temperature and voltage variations).

The challenge of temperature as a key limiter of processor performance has become especially real as device dimensions are scaled down aggressively to the nanometer range since a smaller gate length has a higher source-drain leakage, leading to a higher current density in interconnect lines and higher power consumption. Hence, a temperature-aware processor design has become a necessity for designers, leading to various thermal modeling tools such as HotSpot. These tools are used to characterize temporal and spatial thermal behaviors at design time, needed for developing efficient and effective thermal management techniques at the architectural level as well as at the dynamic level where the chip is adapting on the fly to changes in temperature.

However, while design time thermal models are very useful for a quick analysis and evaluation of thermal solutions, run time thermal characterization offers a more accurate and direct result of processors with actual workloads. It can also be used to validate the effectiveness of design-time thermal modeling tools. One common way of

measuring temperature of processors is using an infrared camera that can directly measure the photons levels emitted by the processor.

Unfortunately, the challenges of real time thermal characterization using the infrared camera lie in the innate inaccessibility of temperature on the processor due to the package and cooling configuration itself. While infrared (IR) thermal imaging has been increasingly popular among researchers, one challenge is to set up a unique cooling configuration requiring an IR transparent cooling fluid to make the processor temperature-accessible to the camera. Consequently, different thermal characteristics from a running processor under regular metal heat sink configuration would be observed.

This work will reveal the differences in run-time thermal characterizations of processors between a metal heat sink configuration and oil heat sink configuration. Moreover, I will also expose the workload thermal behavior of a processor under oil heat sink configuration, which would be useful for evaluating architectural thermal solutions and dynamic thermal management techniques.

1.2 Research Overview

Due to the inaccessible temperatures of the typical processor, it is difficult to exploit the accuracy of an infrared camera to capture the thermal data of the running processor under typical cooling configuration since the processor is masked from the camera lens by a metallic heat sink. On the other hand, to replace the metallic heat sink with a infrared transparent fluid heat sink would mean that the original configuration is

altered and thermal data collected would not accurately reflect typical run-time temperatures.

Dynamic Thermal Management (DTM) techniques that require run-time measurements of temperature rely on on-chip thermal sensors to engage in a reactive and adaptive process of dynamic frequency scaling, dynamic voltage scaling, fetch and clock gating, local toggling and computation migration.

To study the effectiveness of using an infrared camera to characterize thermal behavior of various SPEC CPU2006 benchmark workloads, we propose a three stage process.

1. Comparing the thermal results collected using an IR camera and using the on-chip thermal sensors on a processor under an oil heat sink configuration and comparing them.
2. Comparing the thermal results obtained from using the on-chip thermal sensors under the two different heat sink configurations.
3. Studying the thermal characterizations of different workload applications.

In the first stage, the relationship found between the infrared camera and the thermal sensors would enhance the effectiveness of using the on-chip thermal sensors as a relevant instrument to make thermal measurements and characterization for the purpose of effective DTM techniques.

In the second stage, by studying the differences and relationship between thermal measurements collected from two different heat sinks, we would be better able to develop DTM techniques while using the oil heat sink configuration, which is necessary for the utilization of the infrared camera.

To assist in the analysis of the different thermal results, the techniques used to model temperature and heat transfer, thermal models will be studied. This would give us the necessary tools required to understand the differences caused by the different heat sink configurations or the different thermal measuring instruments as set up above. Hence, by understanding the differences of the run-time thermal characterization of processor under typical metal heat sink configuration and run-time thermal characterization of processor under fluid heat sink configuration, we can justify the use of the oil heat sink apparatus in producing accurate thermal behaviors of different benchmarks.

The second part of this work involves the thermal characterization of actual benchmarks running on processor under an oil heat sink configuration, using an IR camera. The thermal data collected will be substantiated with spatial, temporal and statistical information valuable for a complete thermal characterization of these benchmark applications.

This report is organized as follows. I will first discuss previous related work in run-time thermal measurements of real processors. Second, I will proceed to analyze the technique used in the HotSpot tool to develop thermal models and extend this understanding by simulating a HotSpot thermal model of a Carbon Nanotube Transistor

(CNT) based processor and some research developments in this field. Third, I will show the thermal results of the two set of experiments - oil heat sink and metal heat sink, thermal sensors and IR camera, and provide a convincing analysis of the different thermal data collected under different conditions. Finally, I will show the thermal results of different benchmark applications running on a processor under an oil heat sink configuration, collected with an IR camera. These thermal data will be characterized in spatial, temporal and statistical terms.

Chapter 2: Previous Related Work

Run-time thermal characterizations of real processors are more accurate and useful than thermal results predicted by present tools that simulate a thermal model of the processor utilizing theoretical behaviors at the architectural level. One such tool is HotSpot, introduced in the paper, [1], which uses an equivalent circuit of thermal resistances and capacitances that correspond to microarchitecture blocks and essential aspects of the thermal package to model temperature based on available power densities, input configuration parameters and floorplan consisting of the processor and heat sink.

We would observe that other works such as [10] relying on the thermal RC model to simulate the self-heating of ultra-thin body SOI and FinFET devices which have unpredictable effects of confined dimensions and complicated geometries. [11] is the first work to propose a similar distributed thermal channel model for FinFET devices, which have promising electrostatic characteristics but also suffer from significant self heating. The model was validated using ANSYS and would be used to study the electro-thermal properties of multi-fin devices with both flared and rectangular devices. [13] extended this work to include derivations of different thermal resistances in the FinFET RC model which will be used in our work to validate our understanding of thermal models.

In our work, we would see a further extension of the HotSpot RC thermal model being used for a proposed Carbon Nanotube (CNT) transistor device. [7] focused more on the properties of CNT as a suitable semiconductor similar to Silicon and the possibility of using CNT for FETs. It described properties of CNTs that would make them more power efficient, smaller in size, more resistant to heating and degeneration. However, it also

discussed possible problems such as susceptibility to noise caused by electrical, thermal and chemical fluctuations. [8] explored further the use of CNTs in FETs and addressed several differences between CNTs and silicon which are important to the development of CNT based FETs. To support their work, a logic gate was fabricated using CNT and results were shown to prove their success. On a different note, [9] proposed a different application of the CNT, presenting a methodology to solve heating problems by using CNTs as “thermal interconnect” to transfer heat dissipated in a region of high activity such as ALU to regions of lower activity such as the cache. This is possible because CNTs have much higher thermal conductivity than typical heat spreader materials.

There have been many papers proposing experimental methods of IR thermal imaging such as [2], [3], [4] which are useful for the architecture community to validate processor power and thermal models. [2] and [3] discuss the advantages of 'observing the actual temperature and power behavior of proposed high performance systems' since 'without the measurement of real-time responses from the processor, the best efforts of the architecture community are reduced to best guesses and approximations when modeling the power and thermal behavior of proposed architectural designs. These two works focused on designing/ building an effective infrared measurement setup that simultaneously captures run-time power consumption, thermal characteristics and performance activity counters from modern processors. In addition, [2] included some previous failed setups that the group encountered. All these mistakes contributed to the aspects of their final successful setup.

[4] proposes an experimental technique 'which allows for spatial-resolved imaging of microprocessor power (SIMP)'. The work utilizes IR thermal imaging to derive the underlying power distribution by 'determining the temperature fields for each individual power source on the chip'. The discussion of the impact of power and temperature limitations of high performance CMOS chips shows the importance of temperature and power distributions for chip floor planning, layout, design and architecture and how the SIMP method could be used to directly measure the temperature and power fields as a function of workload and frequency.

While these works all propose a similar measurement setup involving IR thermal imaging and an IR transparent heat sink and show the importance of temperature distributions for architectural design and DTM techniques, our work studies the disparity of the experimental setup under IR transparent heat sink cooling configuration and real packaged processors. This disparity is caused by a different thermal characteristic of the cooling configurations – one is due to an IR transparent flowing fluid, one is due to a metal heat sink cooled by convection current of the air in the ambient surroundings. This leads to a very different thermal behavior of the processor when different workloads are running on it.

Similar work has been done in [5], which 'characterizes the differences between two cooling configurations - forced air flow over a copper heat sink and laminar oil flow over bare silicon'. Several attributes of the IR thermal imaging setup are discussed such as the thermal characteristics of the IR transparent heat sink, rate of flow of cooling solution and direction of flow that would result in the differences observed in the thermal

data of the processor under two different cooling configurations. It also makes use of HotSpot RC model to model the cooling configuration of the IR transparent heat sink using RC circuits.

Our work would validate this RC model and extend my analysis to the differences of the thermal data collected by the on-chip thermal sensors and the IR camera. Finally, with this understanding of the deviation of experimental measurements, I will collect thermal results and provide thermal characterizations of several benchmark workloads for the processor running under the oil cooling configuration.

[6] presents a similar experimental setup to 'measure and characterize the thermal behavior of real processors and their workloads'. It focuses on a theoretical approach of defining the attributes of the IR transparent cooling configuration to match the metal heat sink solution. It also studies the experimental process for producing accurate thermal results in an effort to address challenges and improve the simulation and modeling of thermal characterization of processor designs.

Our work will begin with comparing the thermal measurements of on-chip thermal sensors with IR measurements and then to acknowledge the disparity of results from an oil heat sink experimental setup and a typical metal heat sink setup using the RC thermal model. In [5], we only see simulations of thermal behaviors in HotSpot. Our work is novel in comparing 1) the thermal results of thermal sensors and IR camera and 2) the thermal measurements from processor in oil heat sink configuration and metal heat sink configuration on real processors instead of just simulations in HotSpot. Finally, we

will thermally characterize several benchmark workloads via IR thermal imaging displaying statistics, hotspot loci and variograms.

Chapter 3: Thermal Models & Methods

To understand thermal runtime behavior, we need to first understand heat transfer on a chip. This could be done by studying thermal modeling in design tools such as HotSpot.

3.1 The HotSpot Design Tool

HotSpot, as proposed in [1] is a design tool to model temperature and characterize temporal and spatial nonuniformities and application-dependent behavior. The tool is very useful for developing temperature-aware design and power management techniques that could directly target the spatial and temporal behavior of operating temperature.

By understanding the thermal modeling techniques used in HotSpot, we would be able to utilize the tool as a convenient means to account for the thermal behaviors in our experiments and understand the differences in thermal data obtained via different setups and cooling configuration. The HotSpot model is convenient because it draws a parallel of the RC Circuit to model temperature.

3.2 How It Works?

Hence, lumped values of thermal resistance and capacitance can be computed to represent heat flow among regions of chip and from each region to the thermal package based on this duality between electrical and thermal resistances and capacitances. This will be used to derive a dynamic compact model of the heat flow among the different architecture-level blocks within a microprocessor. In HotSpot, a new compact model is automatically generated for different microarchitectures based on the parameters and the

power trace it receives. It is also able to solve the thermal RC circuit differential equations quickly.

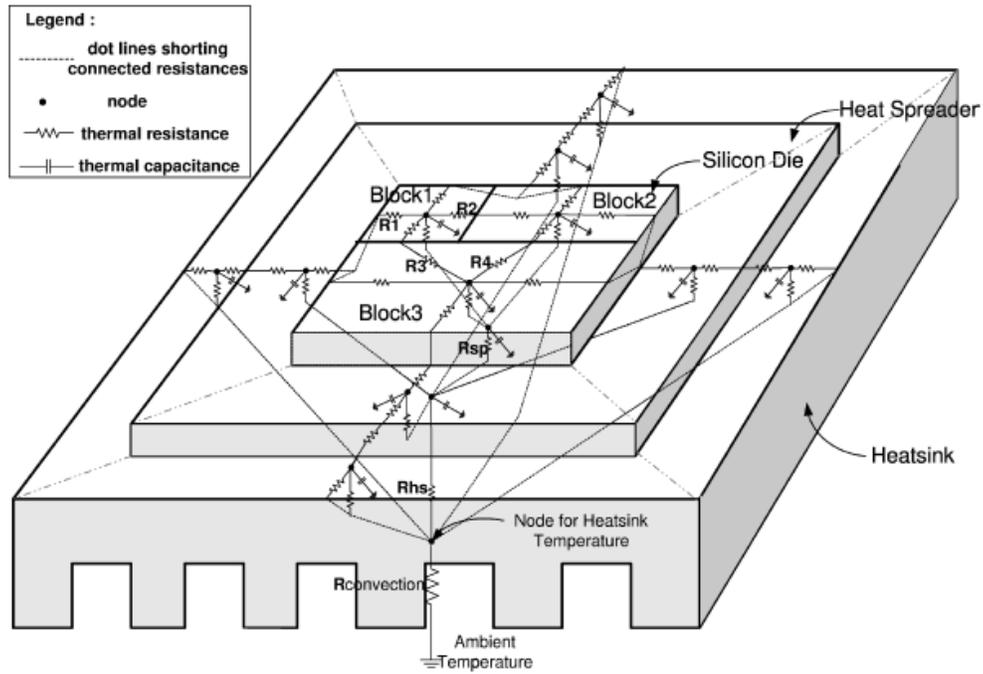


Figure 1: Example HotSpot RC model including heat sink and spreader, [1]

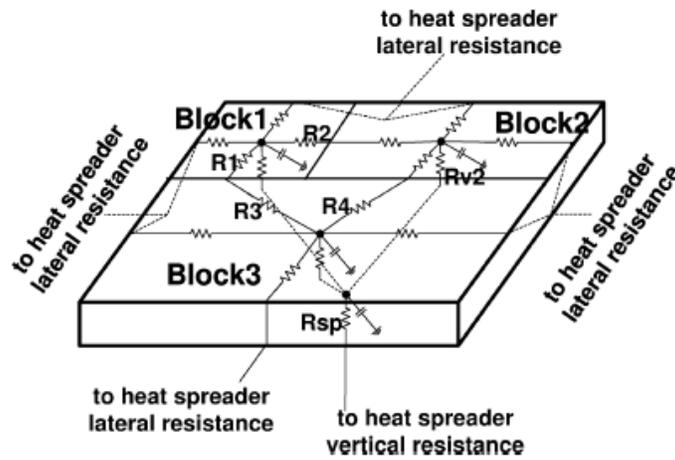


Figure 2: RC model of the die layer, [1]

The RC model consists of three vertical, conductive layers for the die, heat spreader and heat sink and a fourth vertical convective layer for the sink-to-air interface. These are either vertical models (heat flow from one layer to the next) or lateral models (heat diffusion between adjacent blocks within a layer and from the edge of one layer into the periphery of the next area). The die layer is divided into blocks that correspond to the microarchitectural blocks of interest and their floorplan. The sink is divided into five blocks: one for the area under the spreader and four trapezoids for the periphery. The convective heat transfer from the package to the air is represented by a single thermal resistance. The modeling tool assumes the spreader to be isothermal and neglect the small amount of heat flowing into the die's insulating ceramic cap and into the I/O pins. It also neglects interface material between the die, spreader and sink.

HotSpot dynamically generates the RC circuit when initialized with a configuration that consists of the blocks' layout and their areas. This circuit is then used in a dynamic architectural power/performance simulation by providing dynamic values for power density in each block as the values for the current sources. These current sources are at the nodes on each block. Power densities are obtained from the power/performance Wattch simulator.

3.3 Derivation of RC Model

The electrical-thermal duality proposed by [1] is based on the similarity of the Ohm's law to Fourier's law. Thermal resistance is proportional to the thickness of the

material and inversely proportional to the cross sectional area across which the heat is being transferred. Thermal capacitance is proportional to both thickness and area.

$$R = \frac{t}{k.A}$$

where R is the thermal resistance, t is the thickness of the material,

k is the thermal conductivity of the material per unit volume

and A is the cross sectional area across which heat is being transferred

$$C = c.t.A$$

where c is thermal capacitance per unit volume,

t is the thickness of the material,

and A is the cross sectional area across which heat is being transferred

Fourier's law can be described by the following equation.

$$Q = U\Delta T = \Delta T/R$$

where U is the conductance or reciprocal of resistance,

T is the temperature and Q is amount of heat

Hence we compare and see the similarity of Fourier's law and Ohm's law

Fourier's law	Ohm's law
$R = \Delta T/Q$	$R = V/I$

This leads to a similarity between the heat conduction equation and Kirchhoff Current Law. From the principle of conservation of energy, we obtained the following

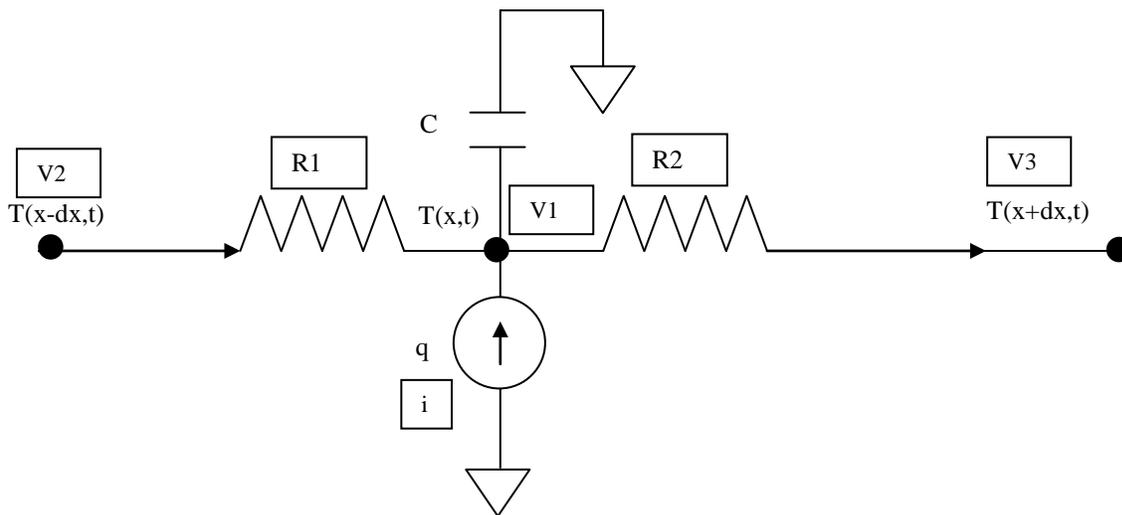
equation.

rate of heat conduction into control volume

+ rate of heat generation inside control volume

= rate of heat conduction out of control volume

+ rate of energy storage inside control volume



where q is rate of heat generation, c is thermal capacitance,

T is temperature at position x and time t

Boxed values refer to electrical equivalents

Using the circuit above and Fourier's law, I can derive the following equation.

$$kA \frac{T(x - dx, t) - T(x, t)}{\Delta x} + q\Delta x = kA \frac{T(x, t) - T(x + dx, t)}{\Delta x} + C\Delta x A \frac{\partial T}{\partial t}$$

where A is the cross sectional area perpendicular to heat transfer,

and k is thermal conductivity

The electrical version of the equation is given below.

$$\frac{V2 - V1}{R1} + i = \frac{V1 - V3}{R2} + C \frac{dV2}{dt}$$

Comparing the two above equations, we have demonstrated that the above heat conduction equation is very similar to Kirchhoff's current law where net current at a node is zero.

Lateral resistances must account for spreading resistance between blocks of different aspect ratios. Vertical resistance of the heat sink must account for constriction resistance from the heat sink base into the fins. Spreading and constriction resistances account for the increased heat flow from a small area to a large area and vice versa. The power densities are comparative to the current sources in the thermal RC model.

3.4 Integrating Thermal Model in HotSpot

At first, initialization information in the form of initial temperatures and floorplan must be passed to HotSpot. The initial steady state temperatures can be generated in the first place by running the HotSpot application once. Second, at runtime, the power dissipated in each block is averaged over a user specified interval and passed to HotSpot's RC solver which returns the newly computed temperature.

Systematically, the capacitances and resistances from the grid are derived from the geometric and thermal properties of the material. Using Kirchhoff's Current Law, the HotSpot software obtains the number of equations as the number of nodes. Initially, with

the steady state temperatures, capacitance has no effect on the current. Hence, simply solving the equations involving resistance values and power densities (thermal current sources) would give the initial temperatures (thermal voltages) as described in the first step in the above paragraph.

In the following time intervals, the initial or previous temperatures (thermal voltages) are used to solve for the RC equations which now involve differential equations of thermal voltages due to the transient temperatures (thermal voltages) changing over time. The Runge Kutta 4 method is used to solve for node temperatures over given time intervals.

This method is boundary and initial condition independent since the component values are derived only from material, physical and geometric values.

3.5 Validation of Model

Any source of localized, time dependent measurements of physical temperatures at a microarchitectural granularity could be used to validate our model. For example, [1] compared HotSpot with Floworks, a commercial, finite-element simulator of 3D fluid and heat flow for arbitrary geometries, materials and boundary conditions that performs full fluid dynamics calculations, including air flow. They verified that the two obtained similar steady state operating temperatures and transient response.

3.6 Understanding Thermal Modeling during Design-time

With inputs of the configuration file, floorplan and power trace of a particular SPEC benchmark, gcc, I generated outputs of the temperature trace file which contains the transient temperatures and the steady state file which contains the steady state temperatures. This is done on a DEC Alpha EV6 processor. After that, I use the steady state temperatures as initial temperatures for the next run. The final output is shown in the figure below.

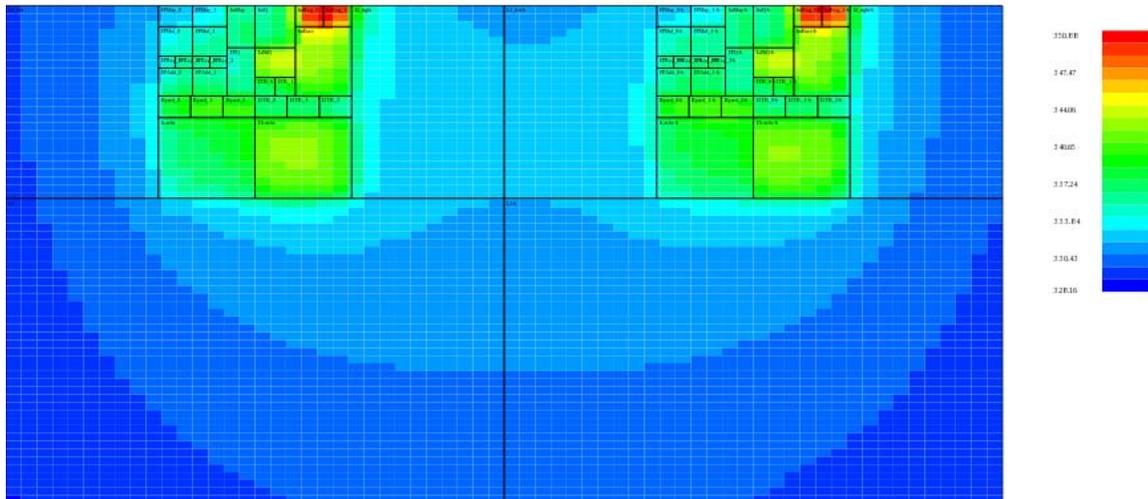


Figure 3: Generated HotSpot Thermal Map of Dual Core with gcc running on both cores

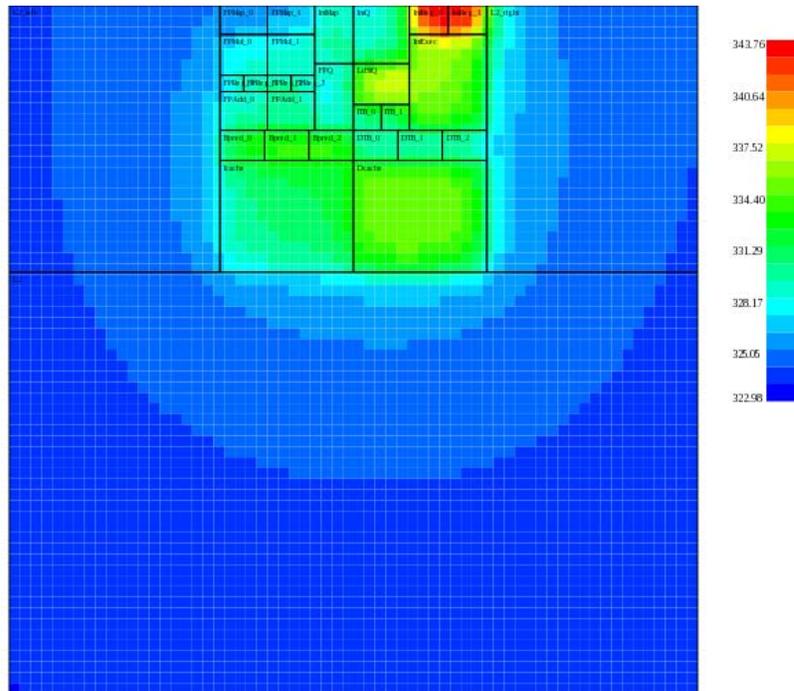


Figure 4: Generated HotSpot Thermal Map of Single Core with gcc

With the configuration parameters and floorplan, the equivalent RC network is designed and the values are determined by the power trace values with the thermal RC equations. From the RC networks, HotSpot is able to calculate the values of temperature for individual nodes in each block taking into consideration the size of the grid and the fact that the block acts as a low pass filter that attenuates spatial high-frequency variations.

3.7 Thermal Modeling of FinFETs

We also propose that the thermal model used in the HotSpot tool can also be used to thermally model the FinFET device, which is a quasi-planar double-gated device,

formed by creating a silicon fin which protrudes out of the wafer, wrapping a gate around the fin, and then doping the ends of the fin to form the source and drain' [11]. Figure 5 below shows how a multifin device looks like.

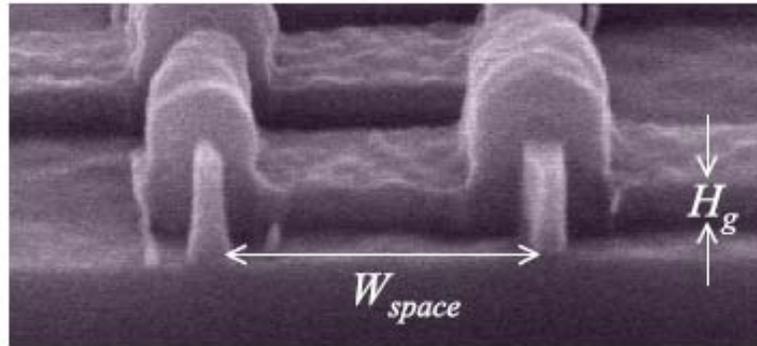


Figure 5: Image of Multi-fin Device, [11]

[10] and [11] introduced thermal models due to the self heating of the ultra- thin FinFETs. Both proposed models that use the similar thermal – electrical duality of resistance and capacitance as proposed by HotSpot.

3.8 Validating RC Thermal Model for Single-fin FinFET in [11]

To test my understanding of thermal modeling techniques, I will first validate the model proposed by [11] and [13] by calculating the values of the thermal voltages or temperatures using the resistance values in the thermal models and Table 1 as shown below for a single-fin FinFET. This is done using KCIRC. All the diagrams below are obtained from [11] and [13]. The first table you can see below contains the dimensions

and thermal conductivities of the FinFET. All these parameters could be found in the following diagrams.

Table 1: Model FinFET Dimensions and Thermal Conductivities, [11]

L_g	H_g	W_g	H_{fin}	W_{fin}	t_{ox}	L_{ext}	L_g	L_{sd}	W_{sd}
50nm	75nm	140nm	65nm	10nm	16Å	50nm	5nm	200nm	200nm
H_{sd}	L_{gp}	W_{gp}	W_{space}	R_{if}	k_g	k_{ch}	k_{ext}	k_{ox}	k_{sd}
65nm	200nm	200nm	100nm	$20E-9 \frac{m^2 \cdot K}{W}$	$45.3 \frac{W}{m \cdot K}$	$6.5 \frac{W}{m \cdot K}$	$13.0 \frac{W}{m \cdot K}$	$1.38 \frac{W}{m \cdot K}$	$13.0 \frac{W}{m \cdot K}$

In the model below, we will obtain the dimensions of H_{fin} , L_{gate}/L_g , t_{ox} and W_{fin} . H_g corresponds to the height of the gate which is the sum of H_{fin} and thickness of the oxide and gate at the top. We also note that W_{gate} is width of the gate which wraps around the fin but this value will not be useful to our calculations as we will see later.

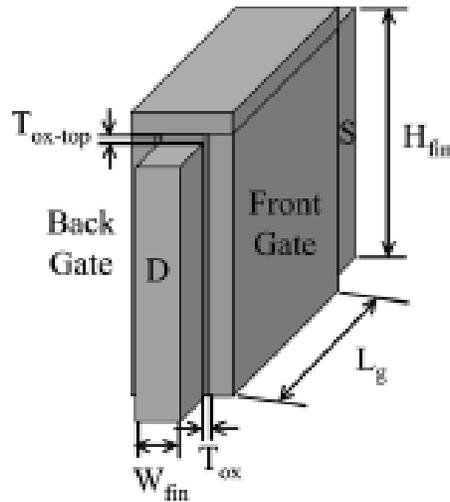


Figure 6: Single Fin FET Device, [13]

A more accurate illustration of how a FinFET device will look like is shown below with its three pads. Our verification of the RC thermal model will be based on this thermal model.

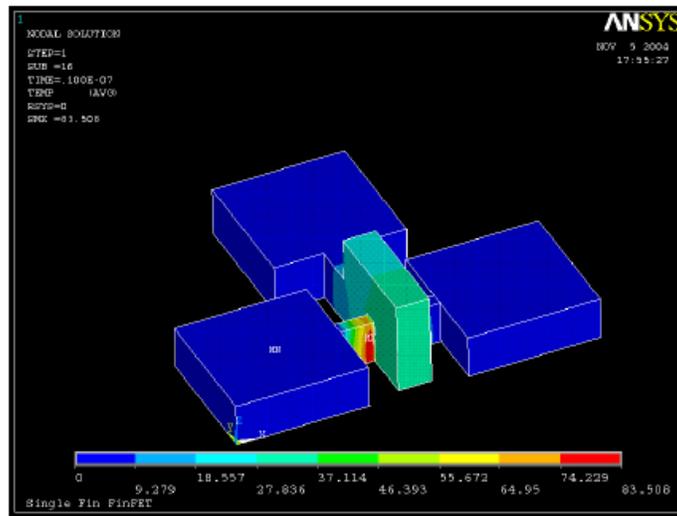


Figure 7: ANSYS Thermal Model of a FinFET, [11]

Flattening the above diagram will give us a top view of the thermal model as shown below. The diagram below gives us parameters such as W_{gp} , L_{gp} , L_{sd} , W_{sd} and L_{ext} . H_{sd} in the table above corresponds to the height of the pads which is observed below. One point to note is that L_q in the above table corresponds to the length between the heat generation regions, T_d to the drain edge of the gate, as explained in [10]. This is due to the fact that Monte Carlo simulation has shown that the heat generation region seems to extend a few nm into the drain. The reason that there is no thermal resistance modeled from T_g to T_s is because the heat generation region is at T_d and thus there is a

heat transfer between T_d and T_g . Another assumption we made is T_s is located at the source edge of the gate.

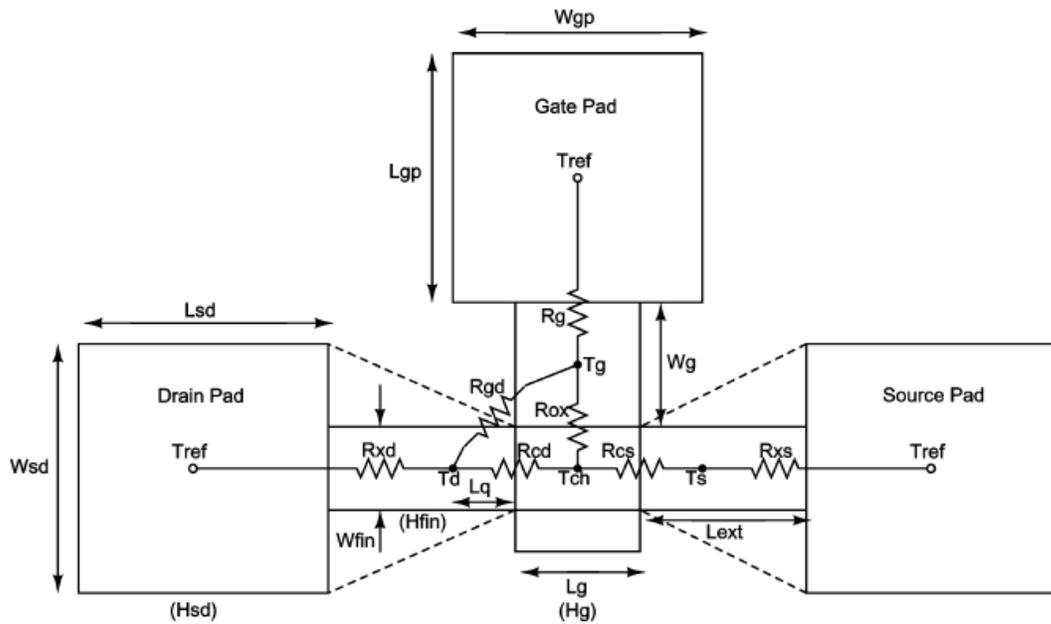


Figure 8: Top View of FinFET Layout with Equivalent Thermal Resistances, [13], (dotted lines rep. flared channel extensions)

Figure 9 below is the simplified version of the thermal model shown above.

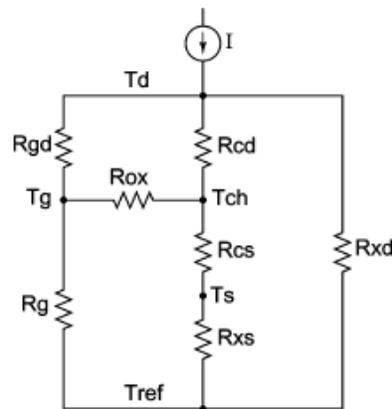


Figure 9: Simplified Circuit Diagram of Thermal Model above, [13]

The resistances in the above figure could be calculated using the values in the above table and the equations below. The equations below describe the thermal RC model at steady state temperatures based on the dimensions and thermal conductivities indicated above. In (7), the extra factored term on the right is due to the interface resistance which is more significant for thin layers of materials, oxide in this case, as explained in [10]. The below equations are provided and explained in [13].

$$R_{xs} = \frac{L_{ext}}{k_{ext} \cdot (W_{fin} \cdot H_{fin})} + \frac{L_{sd}}{k_{sd} \cdot (W_{fin} \cdot H_{fin} + L_{sd} \cdot H_{sd})} \quad \text{--- (1)}$$

$$R_{xd} = \frac{L_{ext} - L_q}{k_{ext} \cdot (W_{fin} \cdot H_{fin})} + \frac{L_{sd}}{k_{sd} \cdot (W_{fin} \cdot H_{fin} + L_{sd} \cdot H_{sd})} \quad \text{--- (2)}$$

$$R_{cd} = \frac{\frac{1}{2}L_g}{k_{ch} \cdot (W_{fin} \cdot H_{fin})} + \frac{L_q}{k_{ext} \cdot (W_{fin} \cdot H_{fin})} \quad \text{--- (3)}$$

$$R_{cd} = \frac{\frac{1}{2}L_g}{k_{ch} \cdot (W_{fin} \cdot H_{fin})} \quad \text{--- (4)}$$

$$R_{ox} = \frac{1}{L_g \cdot (W_{fin} + 2H_{fin})} \left(R_{if} + \frac{t_{ox}}{k_{ox}} \right) \quad \text{--- (5)}$$

$$R_g = \frac{W_g}{k_g \cdot (L_g \cdot H_g)} + \frac{L_{gp}}{k_g \cdot (L_g \cdot H_g + L_{gp} \cdot H_g)} \quad \text{--- (6)}$$

$$R_{gd} = \frac{L_q}{\frac{1}{2} \cdot k_{ox} \cdot W_{fin} \cdot H_g} \quad \text{--- (7)}$$

The values of the resistances are computed below.

$$R_{if} = 2 \times 10^{-8} \Omega$$

$$R_{xs} = 7044237.8 \Omega$$

$$R_{xd} = 6452521.8 \Omega$$

$$R_{cd} = 6508875.7 \Omega$$

$$R_{cs} = 5917159.8 \Omega$$

$$R_{ox} = 3022774.3 \Omega$$

$$R_g = 1059602.6 \Omega$$

$$R_{gd} = 9661835.7 \Omega$$

The thermal current source or equivalently power density is computed as follows.

$$\text{Real Voltage} = 0.9 \text{ V}$$

$$\text{Real Current} = 2295 \frac{\mu\text{A}}{\mu\text{m}} \text{ (per unit width)}$$

(Above values obtained from International Roadmap for Semiconductors (ITRS) guidelines)

$$W_{fin} = 10 \text{ nm}$$

$$\text{Power} = 0.9 \text{ V} \times (2295 \times 10^{-6}) \text{ A}/\mu\text{m} \times 0.01 \mu\text{m}$$

$$\text{Thermal Current, } I = 2.0655 \times 10^{-5} \text{ A}$$

Using KCIRC to model an identical circuit as the simplified circuit above, I found the values of the thermal voltages by substituting the above values of resistances into the circuit as shown below.

Table 3: Thermal Model Nodal temp. & ANSYS temp. in degrees Celsius, [11]

	$W_{fin} = 10nm$			$W_{fin} = 20nm$			$W_{fin} = 40nm$		
	ANSYS	Lumped	Dist.	ANSYS	Lumped	Dist.	ANSYS	Lumped	Dist.
T_d	59.57	96.66	60.42	62.57	79.99	59.26	63.05	67.40	58.35
T_{ch}	17.80	14.18	16.19	24.70	19.29	22.11	28.51	24.30	28.47
T_g	9.66	12.26	11.06	13.00	16.83	14.61	15.37	21.53	18.17
T_s	13.58	7.71	12.57	17.85	11.13	16.48	20.86	15.19	20.96

We can observe that the ANSYS values in [11] and [13] and the values obtained by our computations are similar, thereby verifying that the RC thermal model is accurate in determining the temperatures of a transistor computed by ANSYS which uses finite element analysis to find approximate solutions to the differential heat diffusion equations. The heat – electrical duality relationship is also therefore validated.

Now, we will extend the RC thermal model to CNT transistors.

3.9 Background of CNTs

The possibility of using CNTs as semiconductor devices is discussed in [7] which could solve the problems if decreasing size of metallic wires. Smaller wires lead to overheating and degeneration of wires. On the other hand, CNTs could cool as well as diamond or sapphire. One important characteristic of CNT is that it is semimetal. At the quantum level, electrons behave like waves and particles which can reinforce or cancel each other. Thus, only electrons of certain wavelengths can remain when the electrons spread around the nanotube circumference. This wavelength depends on the circumference of the nanotube. However, in a graphite sheet, only one particular electron

state gives the graphite almost all its conductivity. Thus, only nanotubes of certain circumference (one third) can include this Fermi point in their subset of allowed states, causing them to be truly metallic.

The rest of the nanotubes function as semiconductors which contain band gaps across which a voltage is required to power the electrons. These band gaps depend on the circumference. The smallest diameter nanotubes have very few states that are spaced far apart in energy. As the diameter increases, more and more states are allowed and the spacing between them shrinks.

Working electronic devices have been built out of carbon nanotubes, like FETs. These devices have electrical characteristics similar to silicon devices and can switch reliably using much less power than silicon based device.

There have been many other works exploring the use of CNT as field-effect transistors (FETs) such as [8] or the use of CNT as “thermal interconnect” for on-die heat transfer in [9] as seen in figure 11. The high thermal conductivity, tensile strength, thermal stability and semi-metallic properties make it an ideal candidate for the above applications.

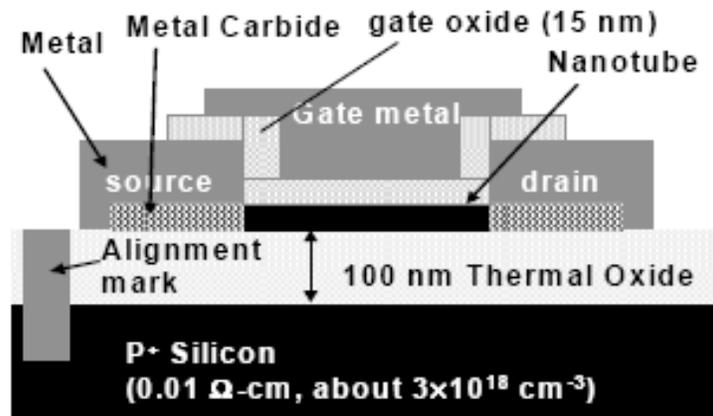


Figure 11: Side view of proposed CNFET, [8]

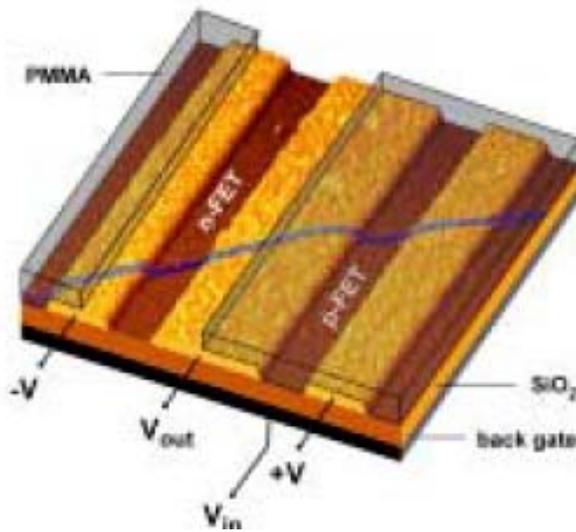


Figure 12: Top view of CNFET, [8]

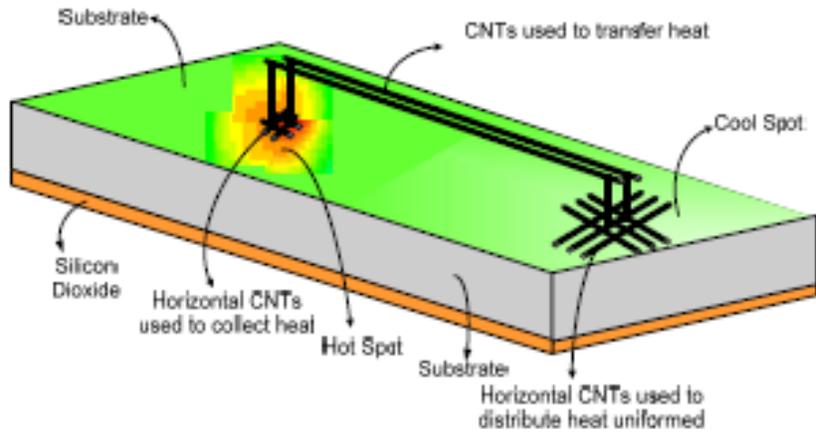


Figure 13: View of the proposed CNT-based heat interconnect, [9]

3.10 Extension of RC Model into CNT FinFET Device

We propose that a similar thermal model could be implemented on a FinFET device based on CNTs instead of silicon fins. The figure below from [12] shows the thermal conductivity for a (10, 10) single walled carbon nanotube converging to a value of $29 \text{ W cm}^{-1} \text{ K}^{-1}$. This will be used for the value of k_{ch} .

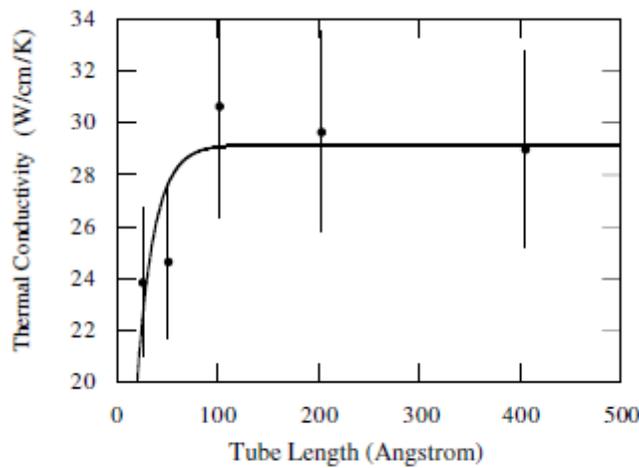


Figure 14: Convergence of Thermal Conductivity along CNT length, [12]

The following values from table 1 were modified to account for the difference in characteristics of the CNT.

$$k_{ch} = 2900 \text{ W m}^{-1} \text{ K}^{-1}$$

$$k_{ext} = 5800 \text{ W m}^{-1} \text{ K}^{-1}$$

$$H_{fin} = 10 \text{ nm}$$

$$H_g = 20 \text{ nm}$$

$$H_{sd} = 10 \text{ nm}$$

The values of the resistances for the CNT transistor thermal model were computed below.

$$R_{xs} = 7412214.2 \text{ } \Omega$$

$$R_{xd} = 7403593.5 \text{ } \Omega$$

$$R_{cd} = 94827.6 \text{ } \Omega$$

$$R_{cs} = 86206.9 \text{ } \Omega$$

$$R_{ox} = 14106280.2 \text{ } \Omega$$

$$R_g = 3973509.9 \text{ } \Omega$$

$$R_{gd} = 36231884 \text{ } \Omega$$

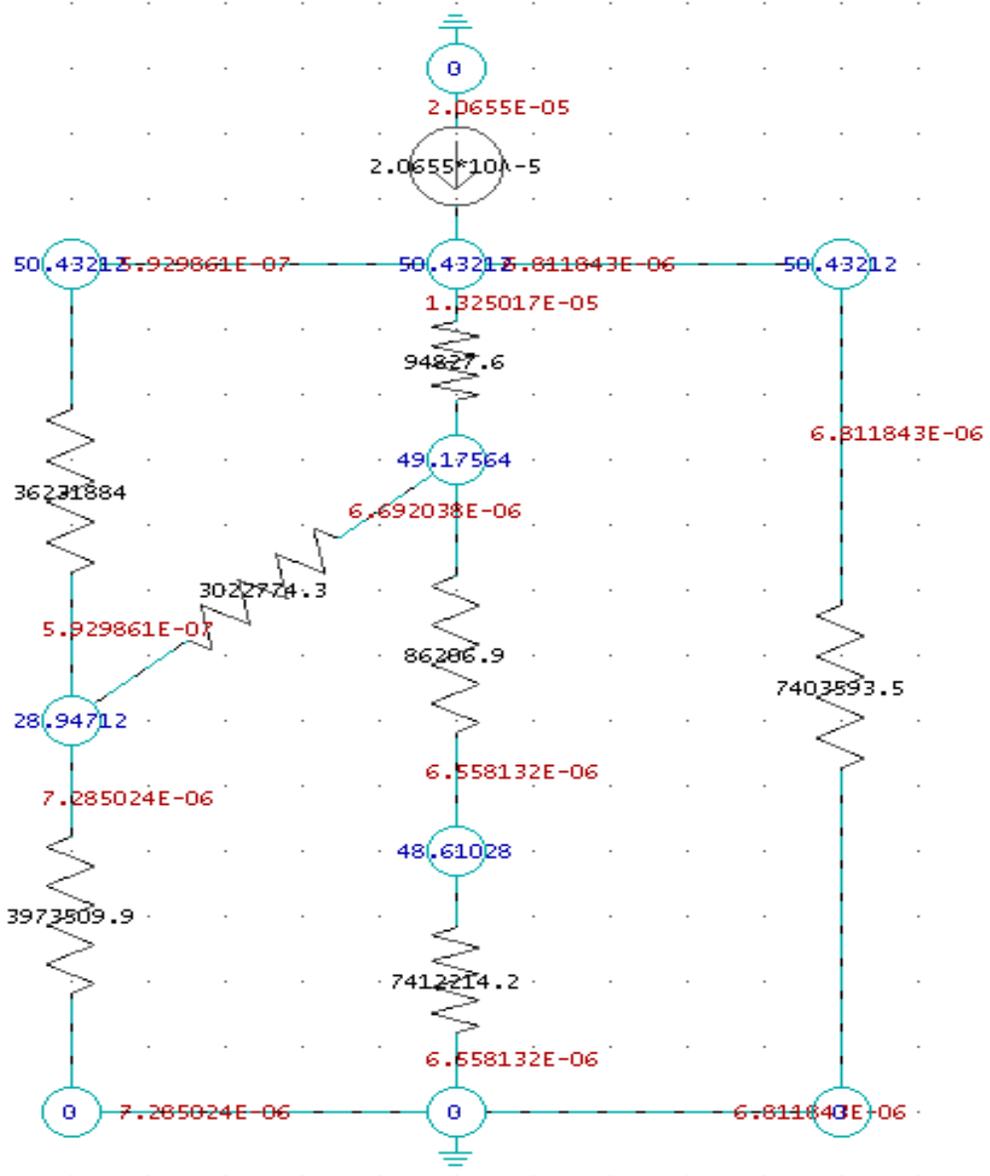


Figure 15: Screenshot of KCIRC Thermal Circuit of CNFET

Table 4: Comparison of CNFET and FinFET temperatures

Temperature at diff nodes	Obtained Results for CNFET/ degrees Celsius	Obtained Results for FinFET previously/ degrees Celsius
Td	50.43	61.06
Tg	28.95	10.01
Tch	49.18	22.60
Ts	48.61	12.28
Tref	0	0

The results of the values of temperature obtained above for the CNFET are slightly different from the FinFET. One observation made is that the range of temperatures is smaller for CNFET (0 to 50.43 degrees) compared to FinFET, (0 to 61.06 degrees). Hence, we immediately see the advantages of using Carbon nanotubes in a transistor.

Chapter 4: Thermal Measurements of Real Processors under Different Configurations

4.1 Overview

In this chapter, we will show results of two experiments we conducted: – 1. Compare the differences in thermal results collected with IR thermal imaging and on-chip thermal sensors. 2. Compare the thermal results collected by the on-chip thermal sensors for a running processor in two different cooling configurations – metal heat sink and oil heat sink. The metal heat sink cooling configuration refers to the processor in a typical setting such as a copper heat sink with forced air flow over the sink generated by a fan. The oil heat sink refers to an exposed processor without the metal heat sink but with a layer of oil flowing over it forced by a pump and cooled by an external thermoelectric cooler. The details of the setup will be given in the next section.

With an understanding of RC thermal models from the previous chapter, we will analyze the results of our experiments. The motivation behind these two experiments is to draw a relationship between using a thermal camera and using on-chip thermal sensor. This will help us determine the accuracy of measurements obtained by the on-chip thermal sensors as compared to the thermal camera. Subsequently, we can take into account the effect of using thermal sensors in run time DTM while using an accurate measuring tool like the thermal camera to design DTM techniques or validate design time thermal modeling tools. We perform this experiment using the oil heat sink and not the metal heat sink because IR imaging requires an IR transparent heat sink above the processor to be able to capture the thermal data. Similarly, in the second experiment, by

comparing results obtained from two different cooling configurations, we are better able to consider different important effects of run time thermal measurement setups which are entirely different environments from typical environments of running processors. Furthermore, we have to conduct this second experiment using only the thermal sensors because the IR camera is not able to capture thermal data through the metal sink. Hence, both sets of experiments will complement each other to fill up the missing link between thermal data collected by thermal sensors of typical running processor under metal sink to the thermal data collected by IR imaging in an oil cooling configuration.

In this chapter, I will explain the oil cooling and thermal camera setup, describe the two experiments, show their results and provide an analysis and comparison between the two cases in each experiment.

4.2 Thermal Measurement Setup

The general setup consists of a oil cooling setup in which infrared transparent mineral oil is pumped through a thermo-electric cooling system and then to the processor, flowing under a sapphire window. Both the oil and sapphire window are IR transparent. Hence, a window is present exposing the processor to the lens of the thermal camera. The thermal camera used is a FLIR SC5600 (3-5 μm spectral range, 30 μm spatial resolution, 100 Hz 6210 x 512 resolution). The software for the camera is Altair software by FLIR systems version 5.90.001. Figures 16 and 17 shows the experimental setup with the thermal camera and figure 18 shows a screenshot of the Altair software used.



Figure 16: Image of IR Camera and Motherboard

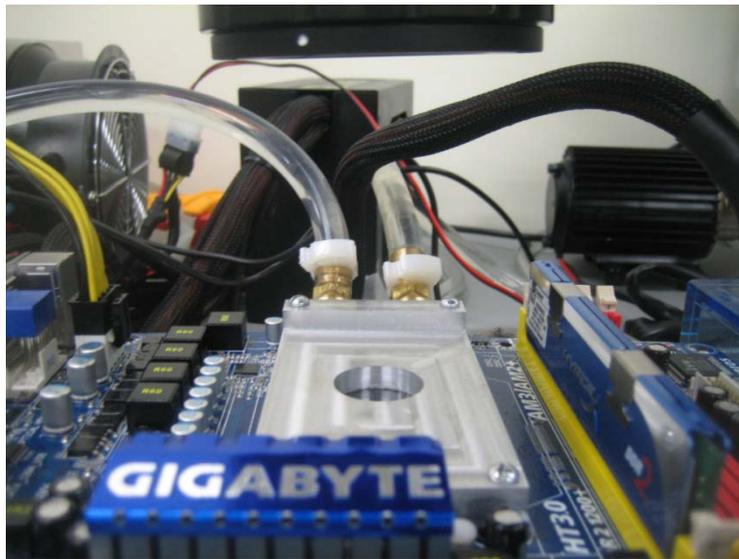


Figure 17: Close-up Image of processor concealed by sapphire window and oil tubes

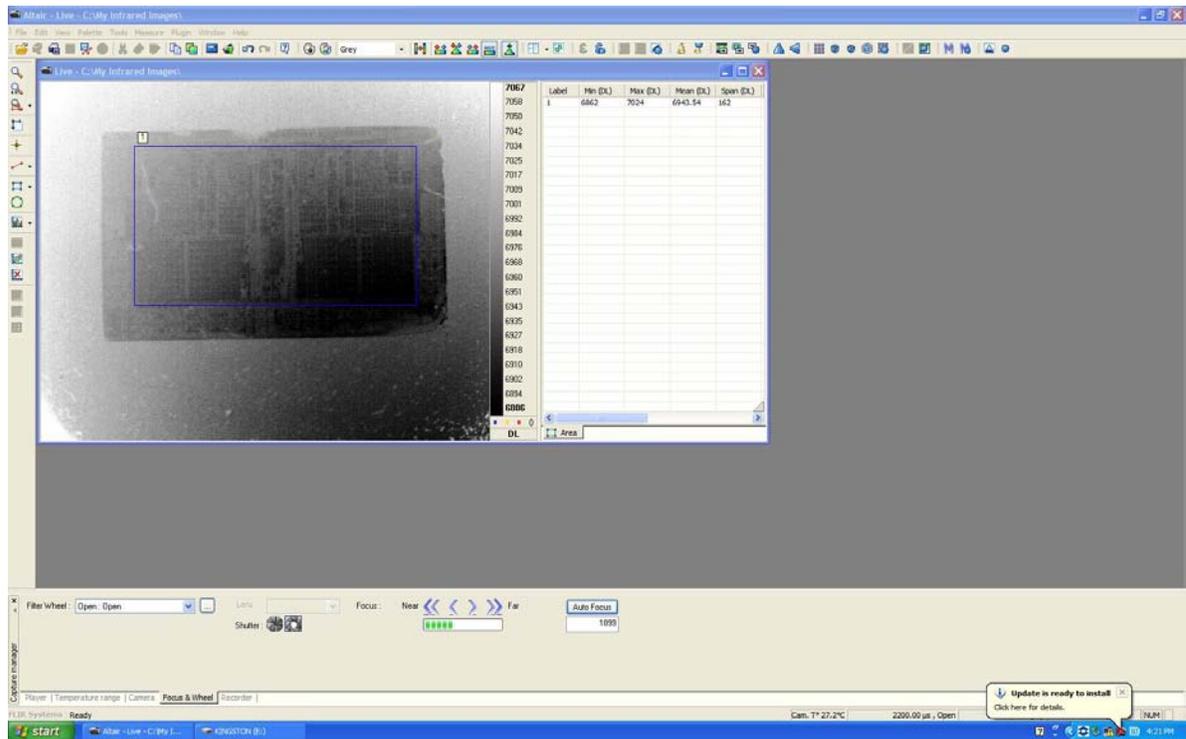


Figure 18: Screenshot of Altair Software with image of processor

The setup has progressed over the time of my research. At first, an open oil bath is used in which the motherboard is immersed partially. Oil is pumped via tubes into the space above the processor for cooling and exit via tubes into the oil bath. Another tube is used to suck the oil out of the oil bath. After a few improvements, the present setup (refer to earlier figures 16 and 17) consists of a closed reservoir of oil that forms a close loop that leaves the oil unexposed, making the setup cleaner and less easily contaminated. This helps with a cleaner window for thermal measurements as well as reduced the need for periodic replacement of oil when it is overexposed and contaminated. At this point, we would like to emphasize that we have no control over the oil flow rate or knowledge of the oil temperature.

The output data that would be processed by perlscript and MATLAB is in asc. file format and the radiometric data is shown in the table below.

Table 5: Radiometric data obtained from asc. thermal output file

Radiometric data :		
Calibration file		
Unit	DL (Digital Levels)	
Emissivity	1	
BackGround temperature	22	°C
Transmission	100	%
Distance	0.01	m
Atmosphere temperature	22	°C
Housing temperature	33.69	°C
Pixel size	14	µm
Pixel pitch	30	µm
Focal length	1	mm
Aperture	2	F/#
Cut on	3.7	µm
Cut off	4.8	µm

4.3 Experiment 1: Infrared Camera vs Thermal Sensors

We collected the thermal data for several SPEC CPU2006 benchmark applications using the IR camera and thermal sensors at the same time. Both instruments are set to the same sampling frequency to collect data at 40 Hz for the same amount of time for each SPEC CPU2006 benchmark. The thermal sensors are from lm-sensor package running on Linux operating system. As mentioned previously, this experiment

will be conducted solely on an oil heat sink cooling configuration since it can accommodate both IR imaging and thermal sensor measurements. These SPEC CPU2006 benchmarks are listed as follows.

Part 1: Integer Benchmarks

- 400.perlbench
- 401.bzip2
- 403.gcc
- 429.mcf
- 445.gobmk
- 456.hmmer
- 458.sjeng
- 462.libquantum
- 464.h264ref
- 471.omnetpp
- 473.astar
- 483.xalancbmk

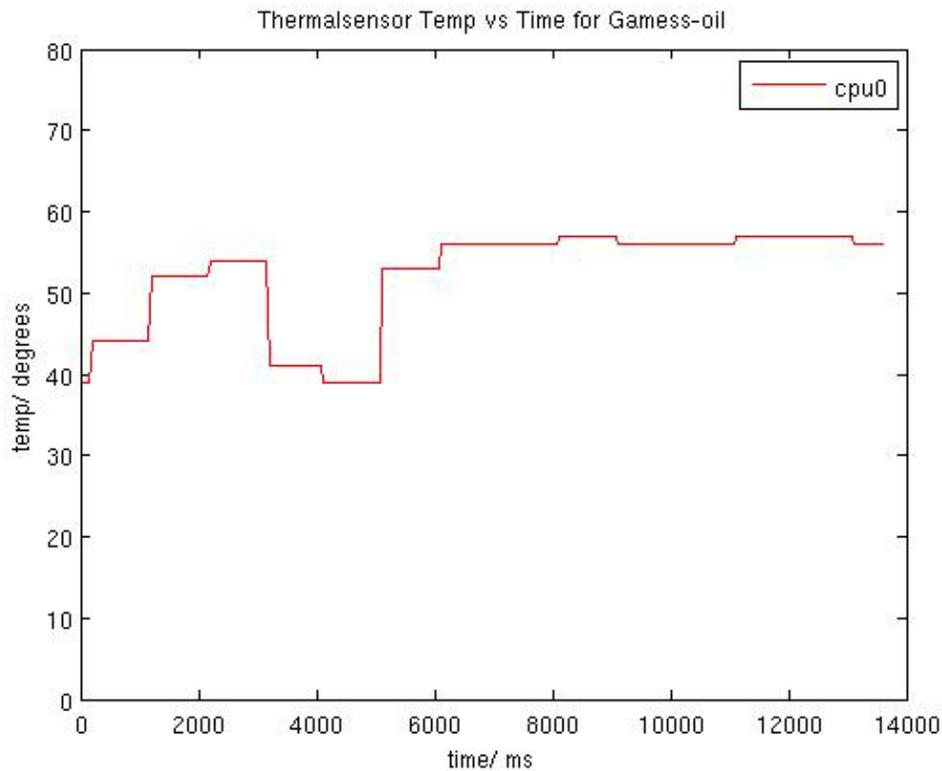
Part 2: Floating Point Benchmarks

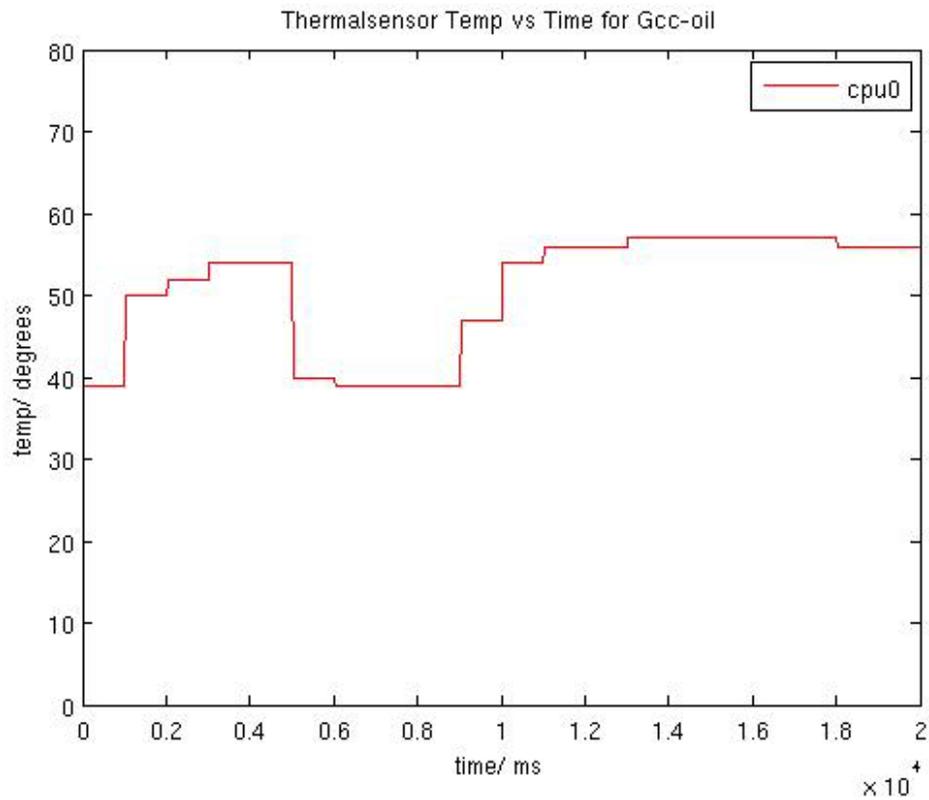
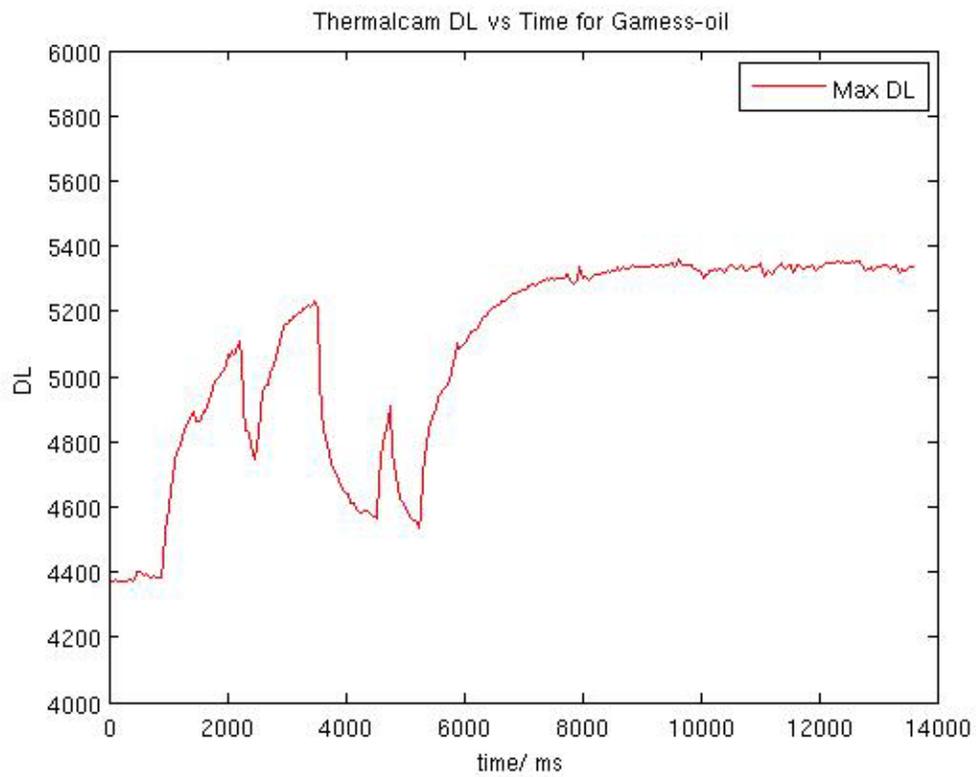
- 410.bwaves
- 416.gamess
- 433.milc
- 434.zeusmp
- 435.gromacs
- 436.cactusADM
- 437.leslie3d
- 444.namd
- 447.dealII
- 450.soplex
- 453.povray
- 454.calculix
- 459.GemsFDTD
- 465.tonto
- 470.lbm
- 481.wrf
- 482.sphinx3
- 999.speccrand

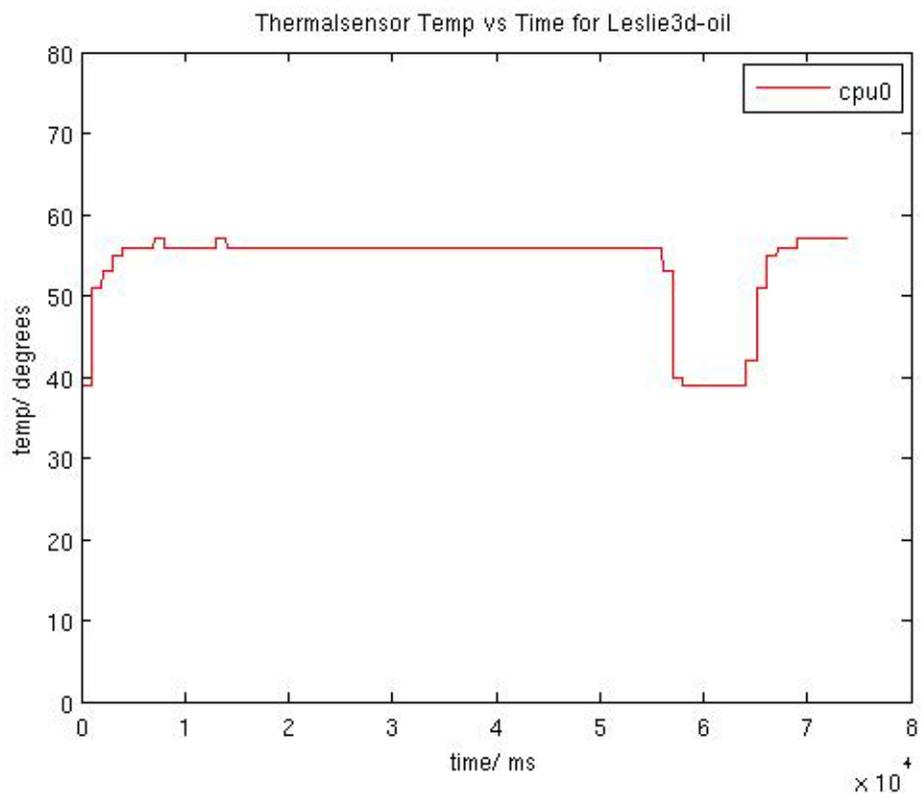
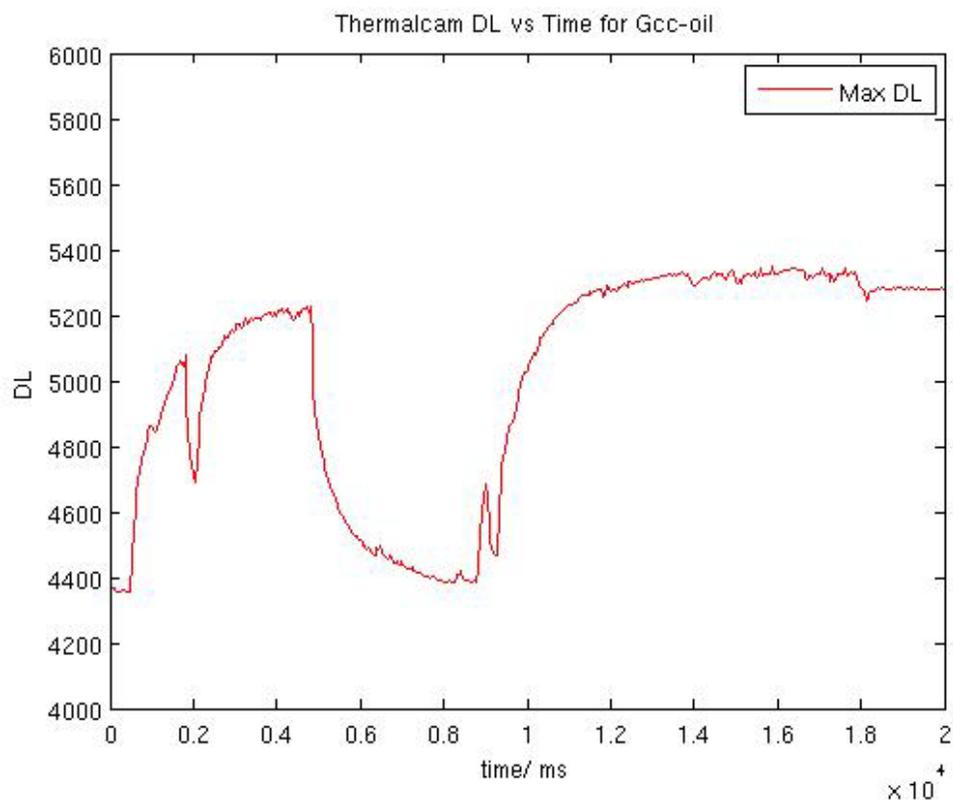
Only the maximum thermal data for each unit time will be recorded and graphed. The digital levels (DL) are captured instead of temperature with the thermal camera because the DL is proportional to the number of photons captured by the thermal camera lens (Capacitor accumulating charge proportional to the number of photons captured). This is due to the emissivity problem whereby the number of photons emitted is dependent upon the material and to obtain the right temperatures from the number of photons captured, calibration is needed. These benchmarks are run on an Intel Core 2 Duo 45nm processor at 1.6 GHz. Moreover, the operating system kernel and scripts are

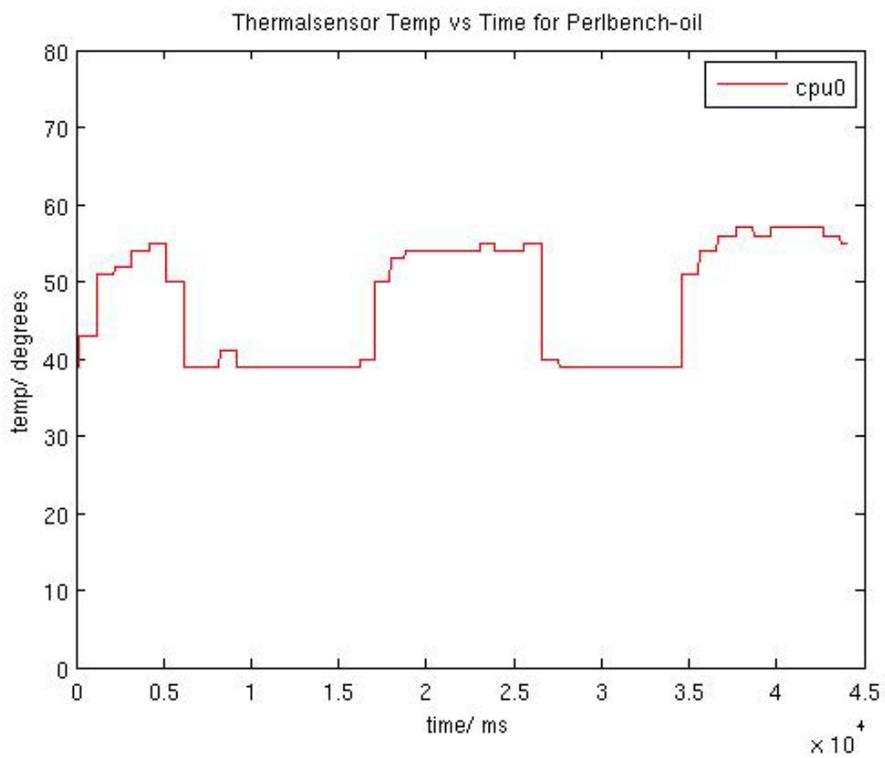
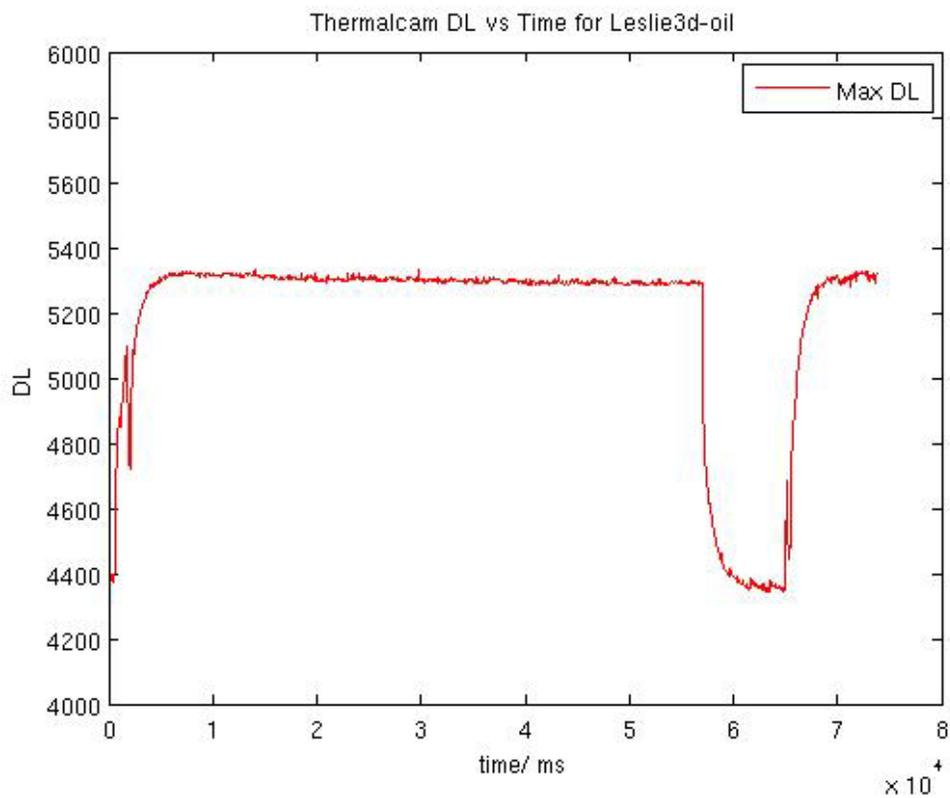
configured so that only one core is running and no switching of applications by operating system is allowed.

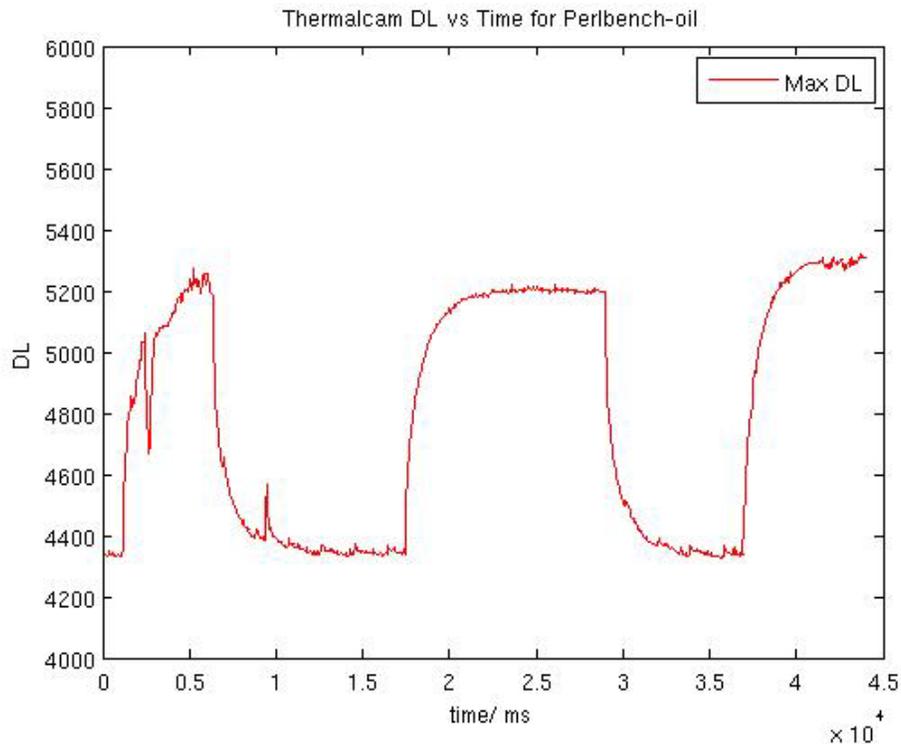
The results are plotted as graphs of temperature and digital levels against time and shown in the figures below for several benchmarks. For the thermal sensors, the graphs are of temperature values measured by the thermal sensors versus time and these graphs have vertical temperature axes in degrees Celsius and horizontal time axes in milliseconds. For the thermal camera, the graphs are of maximum values of DL every time frame versus time and these graphs have vertical DL axes and horizontal time axes.











My observations are as follows.

1. As observed from the above results, one can directly see the similarity between the thermal results collected by the thermal camera and the thermal sensors. The peaks and troughs of both graphs correspond closely with each other (second and third pair of graphs). Furthermore, the magnitude of the dips and rises seems to be linearly scaled between the temperature and DL graphs according to the proportionality factor of around 1 degree Celsius is to 50 DL. Since the graph plots only the maximum DL every frame captured by thermal camera while the thermal sensors are actually measuring the exact temperatures where they are placed, we can deduce that the thermal sensors on the chip are quite well placed

on the processor to capture the thermal variation, especially the increase and decrease in temperatures, of each workload. The differences in the first pair of graph is described and explained in the next point below.

2. There are some differences in the first pair of graphs with one missing peak and trough in the thermal sensor graph. This shows that the thermal sensor failed to capture certain variations caught by the thermal camera. Upon closer observation, we noticed that all the short-term peaks and troughs of about 400 DL are not captured in all the thermal sensor graphs, indicating a disability of the thermal sensor to capture variation of that magnitude and period. Perhaps these certain variations correspond to thermal variations on locations on the die that are not covered by the thermal sensors and thus have slipped through.

3. One can also observe that the plots for thermal sensors are more discrete with less variation than the thermal camera. This might be due to the thermal sensors being slower to respond to changes in temperatures compared to the thermal camera. Moreover, the ADC of thermal sensors has limited resolution which can also cause this lack of variation compared to the thermal camera. Because of this slowness, some peaks that occur over a short period of time such as in the graph showing the 'gamess' benchmark, are missing. It should also be noted that the camera captures the maximum DL everywhere across the chip but the sensors are placed at fixed locations which would inadvertently lead to a difference in measurements.

4.4 Experiment 2: Oil Heat Sink vs Metal Heat Sink

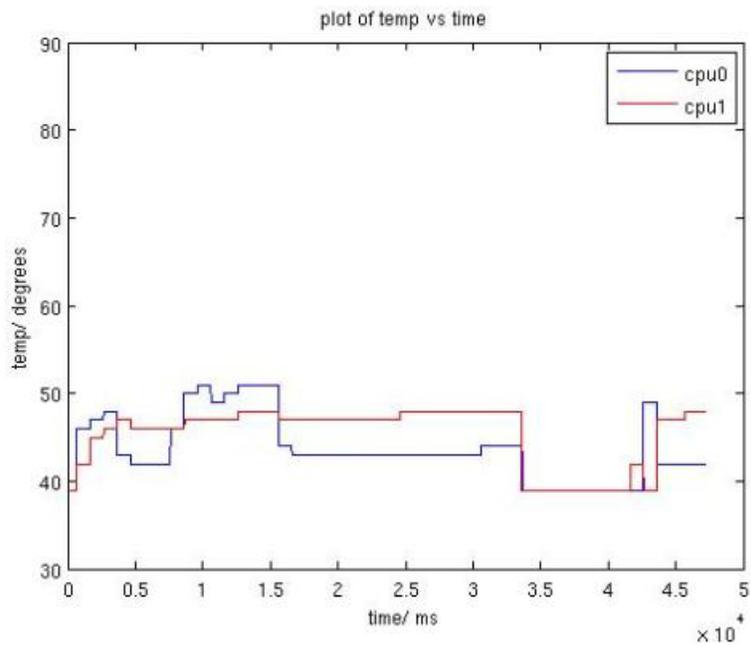
In this experiment, I will collect thermal data using the thermal sensors on the processor while it is running benchmark applications under two different cooling configurations – cooling by oil and cooling by typical metal heat sink. The oil cooling setup is the same as the previous experiment (refer to figures 16 and 17). Additionally, I will allow two workloads from SPEC CPU2006 benchmarks to run simultaneously to add variation to the experiment.

This experiment poses more difficulties since separate runs must be conducted for different cooling configurations, unlike the previous experiment where the thermal data are collected by two instruments at the same time. I also took note to clear the cache after every run of the workload to ensure a fair experiment and comparison between the two cooling configurations. Furthermore, the times of each run of the same workload might differ from each other when run separately. This adds to the difficulty of the comparison.

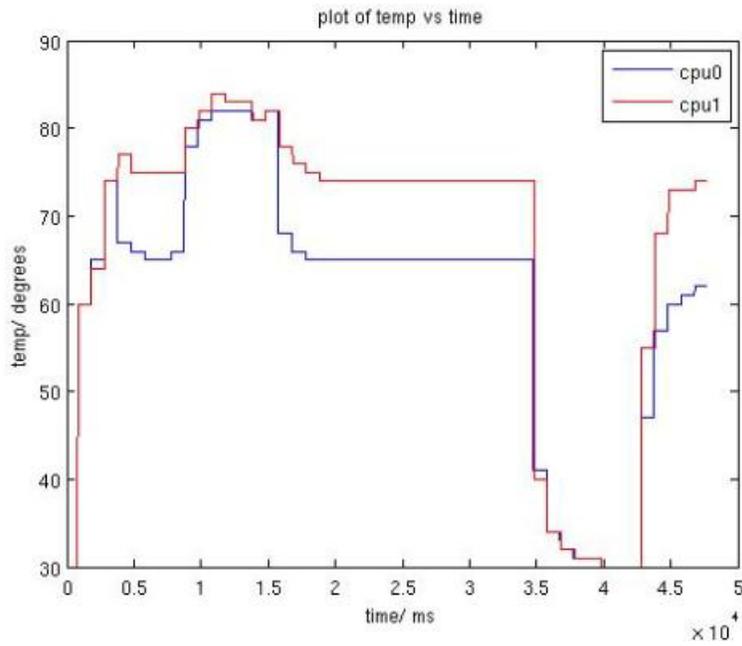
In order to minimize the room for error and ensure a fairer comparison, I ran the benchmarks in a strict order in a continuous fashion. This sequence is: perlbench-perlbench, leslie3d-leslie3d, perlbench-leslie3d, gcc-leslie3d, gcc-perlbench. I ran this sequence of workload pairs only once after booting up to ensure the cache is clear before the experiment. After that, I plotted the results of the thermal data collected on two cores on the same graph for each cooling configuration. I scaled the axes for every pair of graphs to be the same length for an easier comparison. When the time taken was different for the same workload ran on different cooling setups, I extrapolated the data by using the

same value of the final temperature so that both workload run times are the same. The figures below show the graphs of temperature versus time of different benchmark pairs collected by the thermal sensors.

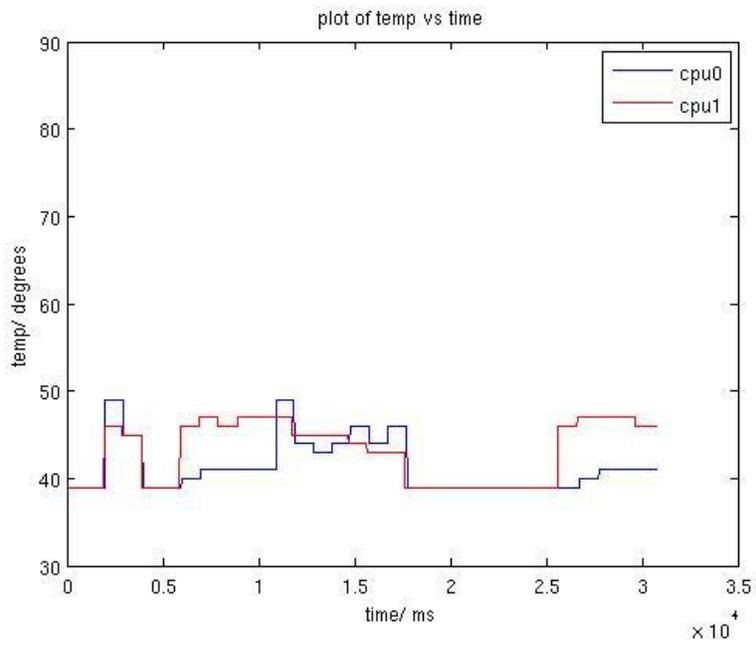
gcc-leslie3d (metal heat sink)



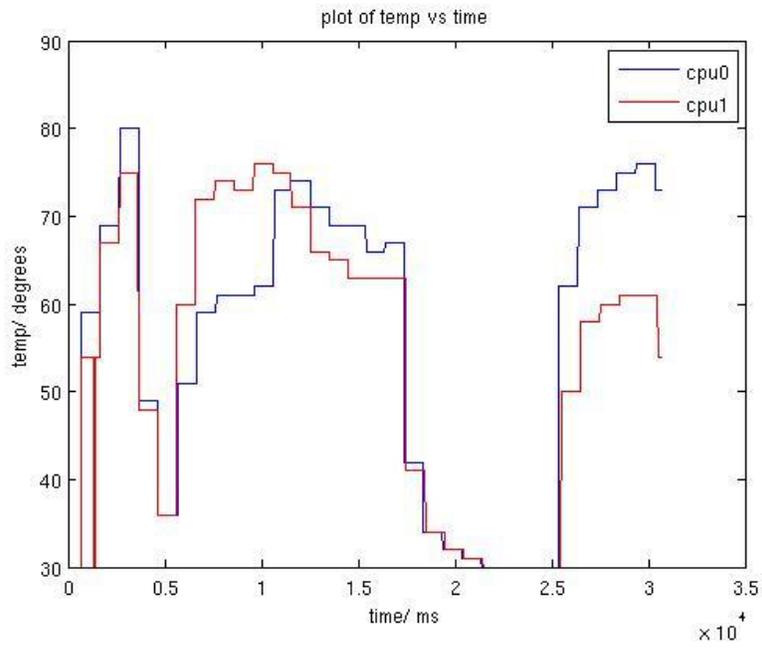
gcc-leslie3d (oil heat sink)



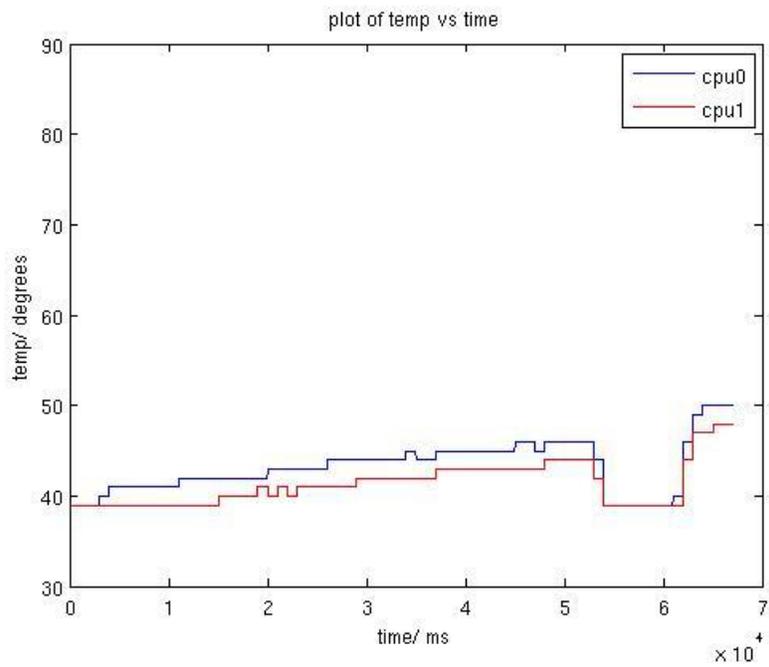
gcc-perlbench (metal heat sink)



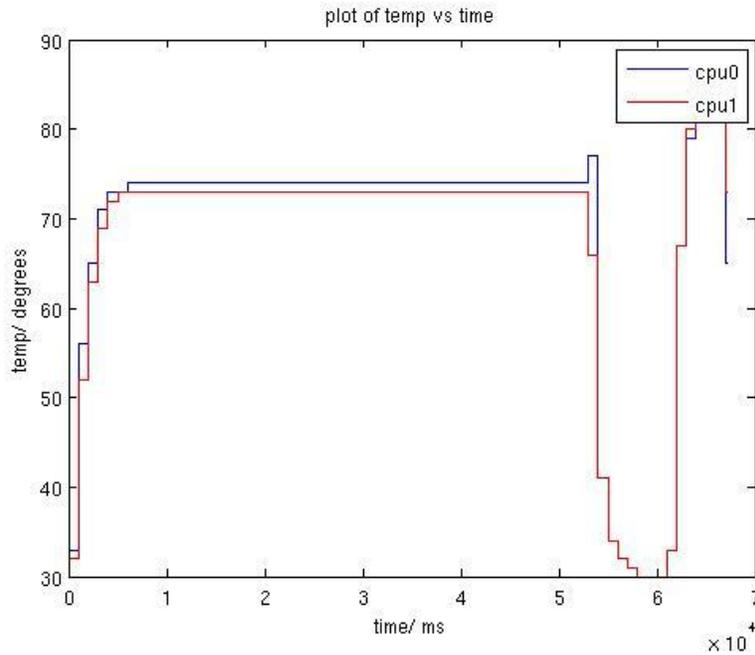
gcc-perlbench (oil heat sink)



leslie3d-leslie3d (metal heat sink)



leslie3d-leslie3d (oil heat sink)



From the above results, we made three observations.

1. We observed that the oil heat sink setup has a larger range of max temperature values than the metal heat sink setup. The range of temperature for the oil heat sink setup is about 30 to 80 degrees Celsius but the range for the metal heat sink setup is about 40 to 50 degrees Celsius. This could be explained by the more effective lateral heat spreading in the metal heat sink because of its higher thermal conductivity and presence of heat spreader, leading to a lower thermal resistance in the RC thermal model than the IR-transparent oil flow. The low thermal conductivity of the oil leads directly to a higher temperature gradient on the processor. This result is validated by [5] which modified the

HotSpot thermal modeling tool to model the oil flow over a bare silicon die. In the work, it also proposed that due to the low thermal conductivity of oil, heat from the silicon is vertically transferred to the oil at a very slow rate. This causes heat from that hotspot to either accumulate at that point or spread to its closest neighbors, raising the temperature of that spot. In the metal heat sink setup, higher thermal conductivity results in good vertical heat transfer from silicon to metal sink and also effective lateral heat spreading in the metal sink instead of silicon like the oil heat sink case. As a side note, the velocity of the oil is a factor in the thermal resistance of the oil as the velocity is directly proportional to the Reynolds number of the oil flow and from [5], we observed the relationship between Reynolds number and the thermal resistance of the oil described in the equations below.

$$R_{oil} = \frac{1}{h_L A_{chip}}$$

where h_L is the heat transfer coefficient and A_{chip} is entire silicon area

$$h_L = 0.664 \frac{k}{L} Re_L^{0.5} Pr^{1/3}$$

where k is the thermal conductivity of oil,

L is length of Silicon along flow direction,

Pr is the Prandtl Number of the fluid

and Re_L is the Reynolds Number of the flow

Hence, we can see that if there is an increase in oil flow rate (with a stronger pump), there will be a larger Reynolds number and a lower thermal resistance of the oil. This will facilitate a higher rate of heat transfer from Silicon to oil. However, this will not be the focus of our work presently.

2. Another observation is that the oil heat sink setup has a slower short term transient response than the metal heat sink setup when the temperatures are changing rapidly. This is seen from the longer time taken to reach a peak when the temperatures are fluctuating. This could also be observed from the rounded peaks and dips for the oil heat sink setup plots. The reason for such slow response when the power density switches rapidly is because of the low conductivity of the oil, which translates to a high thermal resistivity and a larger RC time constant. [5] explained that for a metal heat sink setup, a short term heat pulse will only increase the temperature of the silicon and not the metal heat sink because the metal has a much higher thermal capacitance than the silicon due to its large size. Thus, the temperature of the metal heat sink does not change visibly and thermal resistance of the metal heat sink is not applicable. The thermal time constant is given as below.

$$\tau_{metal\ sink} = R_{Silicon} C_{Silicon} \quad (1)$$

Due to the lower thermal capacitance of the oil compared to the silicon because of its thin conducting layer, the temperature does not change in the silicon but in the oil. Thus, the thermal capacitance of the oil is negligible. Because of the almost constant temperature of the silicon and the lower

resistance of the silicon compared to the oil, the thermal resistance of silicon is negligible. This gives us the thermal time constant below.

$$\tau_{oil\ sink} = R_{oil}C_{Silicon} \quad (2)$$

Comparing (1) and (2), since the thermal resistance of the oil is much higher (2 orders of magnitude) than the silicon, we see that the oil heat sink setup, (2), has a longer response time than the metal heat sink setup, (1) under the application of short term heat pulse.

3. My third observation is that it takes a longer time for the temperature to reach the peak for the metal heat sink than the oil heat sink when the maximum temperatures are steady for a long time. This could be seen in the last pair of graphs. This means that there is a faster long term transient response for the oil heat sink than the metal heat sink. This could be explained by the fact that the metal heat sink has a much larger thermal capacitance than the oil, leading to a longer thermal RC time constant during warmup. For the metal heat sink, since this is a long term transient response, the temperature of the metal heat sink is no longer constant and the thermal capacitance and resistance of the metal heat sink now represents the major factors contributing to the thermal time constant as shown below.

$$\tau_{metal\ sink} = R_{metal}C_{metal} \quad (3)$$

For the oil heat sink setup, the oil still has a lower thermal capacitance than the silicon and is negligible. Temperature changes more in the oil than the

silicon and the thermal resistance of oil is much larger so the resistance of the silicon is negligible. This gives the following time constant, identical to (2).

$$\tau_{oil\ sink} = R_{oil} C_{Silicon} \quad (4)$$

Comparing (3) and (4), the thermal resistance of oil could be seen as comparable to the metal since this resistance refers to the vertical thermal resistance from silicon to oil or metal. Thus, even though metal has lower thermal resistance than oil (better lateral heat spreading), R_{metal} and R_{oil} really refers to the vertical heat transfer between two different materials. Thus, with this in mind, we can see that $\tau_{metal\ sink} > \tau_{oil\ sink}$, since the thermal capacitance of the metal is much higher than the silicon due to its larger size.

The implications of these results and observations are that the many differences in the thermal behaviors of the processor when in different cooling configurations would necessarily affect the way DTM techniques are developed such as engagement duration or thermal thresholds. It is essential to take into account these differences in thermal behaviors while using the IR imaging setup cooled by oil so that the design of the DTM techniques would be efficient and free of thermal hazards. Alternatively, an easier solution would be to achieve a higher control of the oil heat sink apparatus such as the oil flow rate, temperature and direction such that a higher similarity between the two cooling configurations can be achieved.

Chapter 5: Workload Thermal Behavior in Oil Heat Sink

5.1 Overview of Experiments

Characterizing workload thermal behaviors are useful because it allows us to study various aspects of thermal behaviors which could later be used for developing DTM techniques. For example, determining the locations of hotspots for different workloads would enable us to place thermal sensors in a more appropriate fashion. Statistical data could be obtained from raw thermal data over time to give us important information of how different workloads differ from one another or how temperatures on different locations on the chip correlate to one another. All these results would give us a clearer idea in the design of DTM techniques.

In this chapter, I characterized the thermal behavior of various benchmark workloads running on the AMD Athlon II Dual Core processor in an oil heat sink cooling configuration. At this point, we note that based on results of previous experiments, the thermal measurements of the oil heat sink setup does not match the metal heat sink. The processor is set to run at 1.6 GHz. This will be done using the IR camera to capture the thermal data at a rate of 2 Hz over 90 seconds of the application. This rate is sufficient because enough variation of temperature occurs over the first 90 seconds to thermally characterize the benchmark workloads. The thermal data captured is not used directly for my analysis. Instead, the differential thermal data, which is the difference in the thermal data collected from the running application and the thermal data collected from the processor when the processor is switched off (but the oil is still flowing) would be used.

This ensures that temperature variation due to the specific ambient surroundings as well as the oil flow is accounted for in my thermal characterization.

Using this differential temperature data, the difference between the absolute frame and idle frame, I displayed the thermal characteristics of these different workloads through different forms.

1. The first is an image representation of the hotspot locus on the chip. This represents the spatial locations of all the temperatures above a set threshold level at any given time or frame of the thermal data captured.
2. The second set of characterizations is in the form of thermal data statistics, such as the mean, median, max and standard deviation.
3. The final set of characterizations is represented by means of a variogram. This final means of representation of thermal behavior will be discussed in a separate section below.

5.2 Experimental Challenges

One of the challenges in setting up an IR transparent oil cooling configuration is that the oil flow was not regular over different runs of the workloads. This is caused by the difficulty of regulating the oil flow, especially when the oil setup has a large open reservoir which could be affected by the ambient surroundings. Also, the cooling of the oil was done by fans initially which is not as effective as we wanted it to be. The oil pump is too weak to generate a high rate of oil flow. Over the course of my work, the setup was improved. As mentioned previously, the oil reservoir was made to be a closed

one, thereby reducing the sources of uncertain external factors that might affect the oil flow. The oil pump was changed to a better one and a thermoelectric cooler was used to replace the fan coolers. As a result, there was less turbulence or more regular flow during the course of the experiments.

Another challenge that I encountered is the errors in the thermal data collected. The thermal data in the form of asc. files contain missing values and terms that were the cause of many errors when I ran my perlscript. My solution was to design a data checking perlscript to test the thermal data for bugs and correct them manually.

There was also a problem for thermal data file size since collecting too much data would cause my data processing to slow down. I can set the duration of each run, the sampling frequency of the data in order to change the file size. I found that 90 seconds of workload runs at sampling frequencies of 2 Hz corresponds to a file size of around 120 MB which is manageable.

Finally, in the characterization of hotspot locus, there was a problem in representing the locus with lines. The motivation was to provide a graphical means of showing how the hotspots move about on the processor. However, the lines were too messy and I decided to represent just the locations in dots for a clearer view.

5.3 Experiment 1: Characterizing HotSpot Locus

In this experiment, the hotspot loci of the various workloads are characterized on a 2D representation of the processor. This is done by extracting all the thermal DL levels from every frame that are above a user set threshold. Together with its specific location

on the array of thermal data, I accurately reproduce these spots on a separate background so that we can observe the hotspots on the processor. The threshold is set such that we can correctly represent locations of high levels of activity while running the workload, thereby showing the different hotspot loci in different workloads. I conducted the experiments for both single and dual cores. When only one benchmark workload is run, only the left core will be used, otherwise both cores will be used.

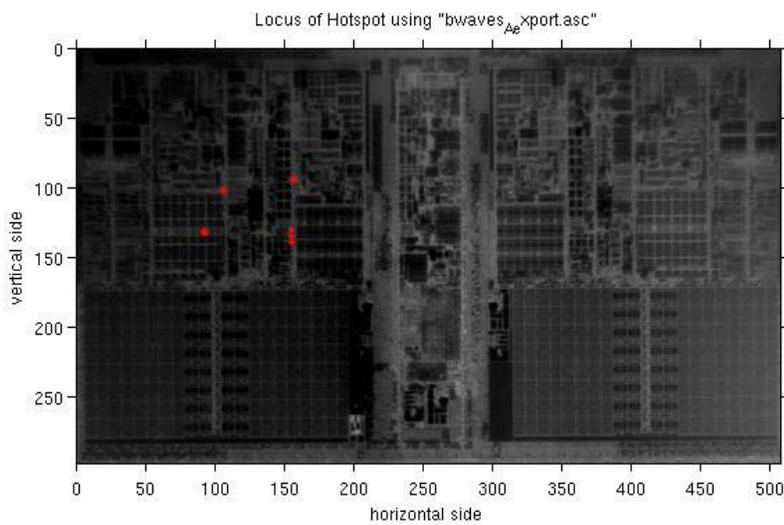
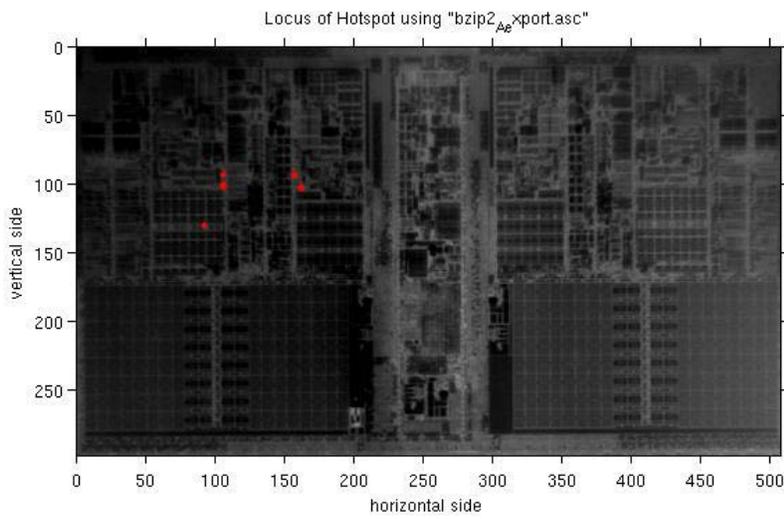
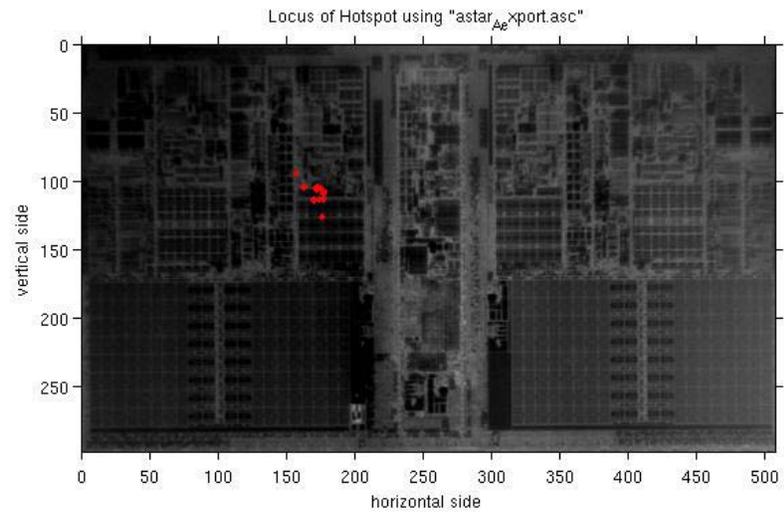
To ensure the uniformity of basis for comparison, I ran the benchmark workloads in 5 different batches. Each batch represents a single period of time when I turned the processor on to collect thermal data. This reduced human and systemic errors within each batch. I found the maximum thermal DL levels for each batch of workloads and set the threshold levels for the hotspot loci accordingly. Also, each batch has its own idle frame. I also made sure to kill every workload after every run. The following table shows the maximum DL for every workload.

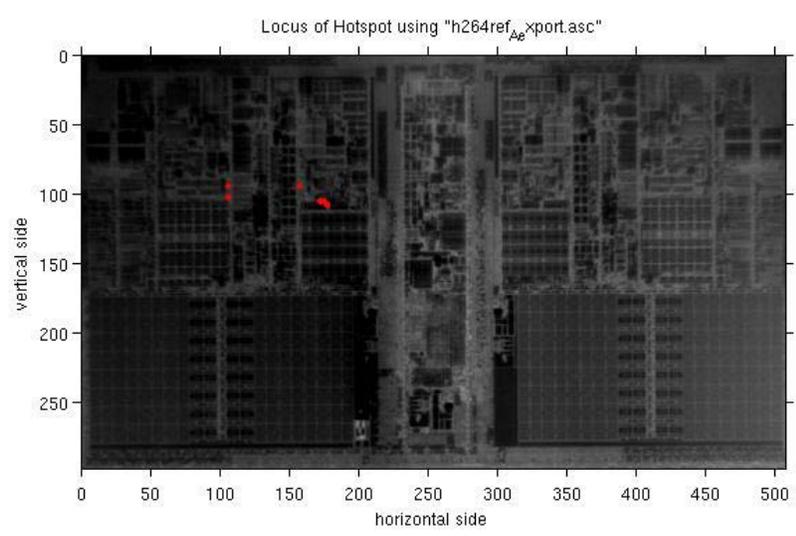
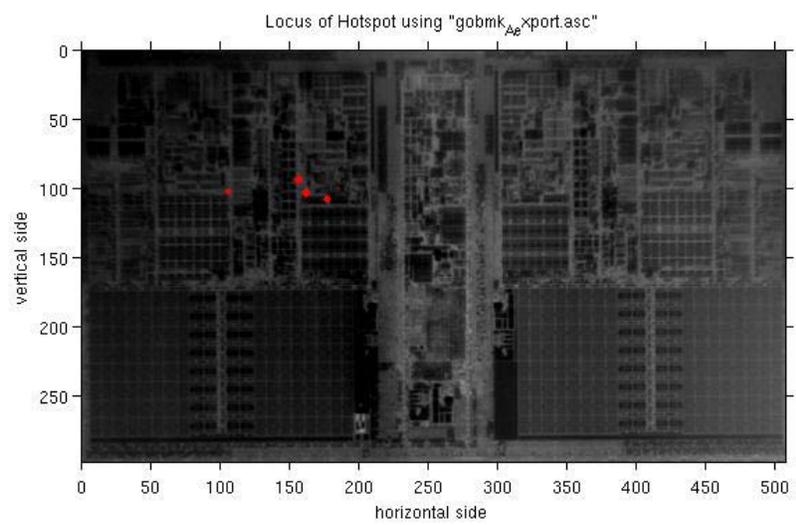
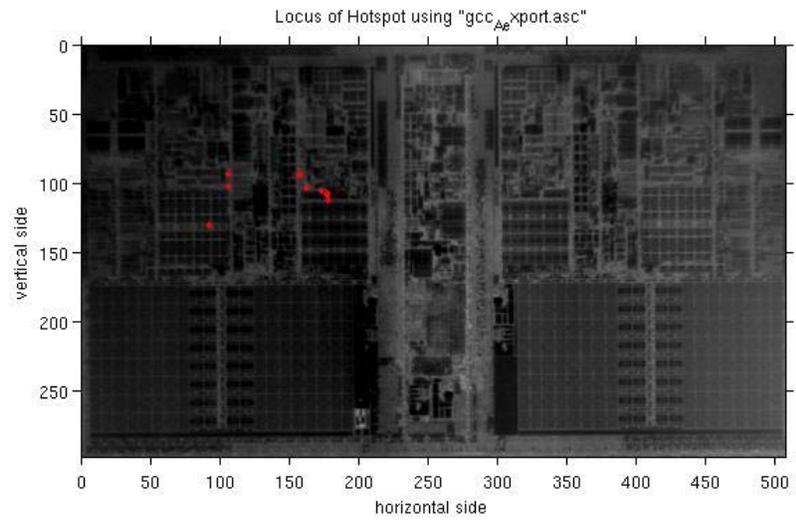
Table 6: Maximum DL for every workload

Benchmarks	Max Data
astar	4651
bwaves	4586
bzip2	4351
gcc	4534
gobmk	4506
h264ref	5159
hmmer	5017
libquantum	5182
mcf	3902
omnetpp	4331
perlbench	4229
sjeng	4597
xalancbmk	4261

cactusADM	3786
dealII	4501
gamess	4485
gromacs	3779
leslie3d	4192
milc	4230
namd	4440
soplex	3782
zeusmp	6821
calculix	5072
GemsFDTD	4068
lbm	3681
povray	4507
sphinx3	3915
tonto	4597
wrf	4173

From the above results, I found that the maximum DL levels seem to be pretty consistent across batches. One anomaly was for zeusmp workload where the thermal DL level was unusually high. Since it is the only one, I neglected its significance and set the threshold levels to be 4000 for those with maximum values above 4000 and 3500 for those below 4000. My results are as follows.





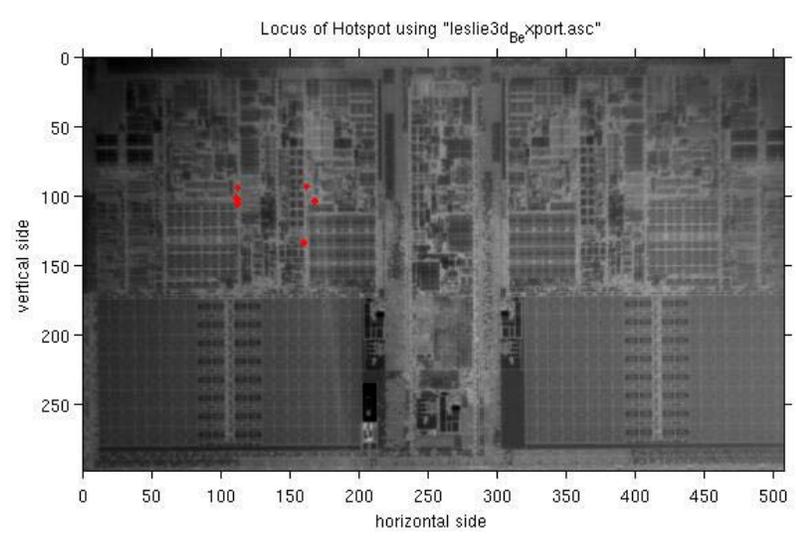
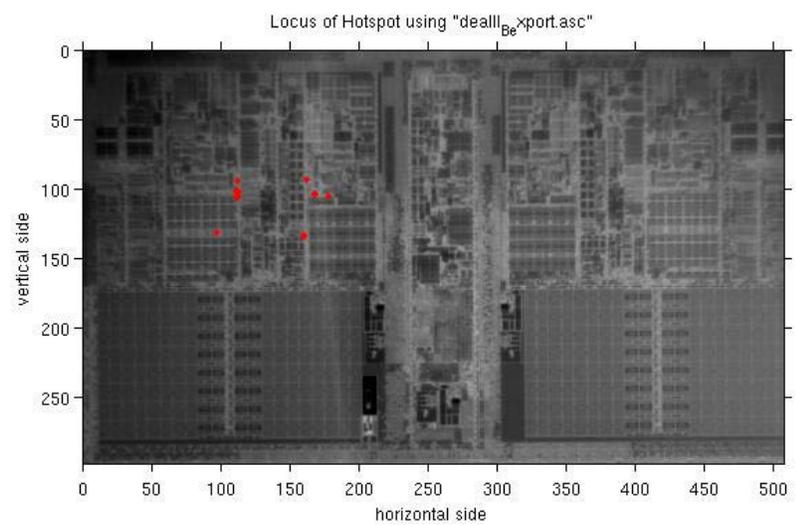
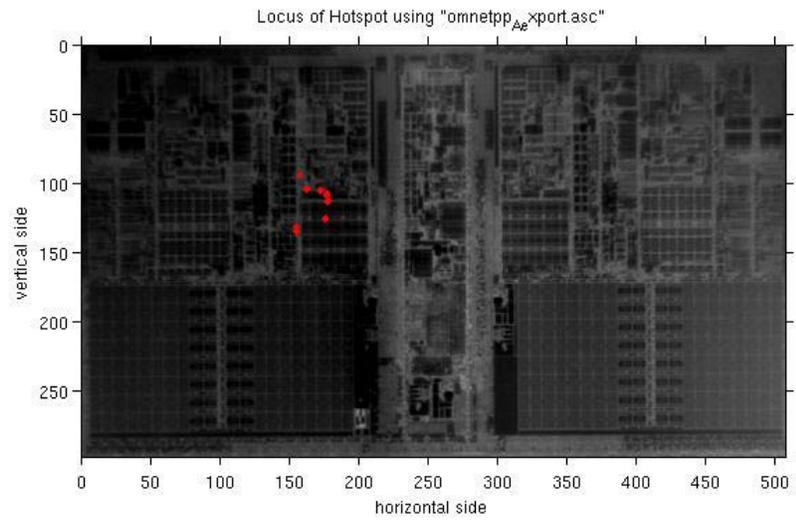
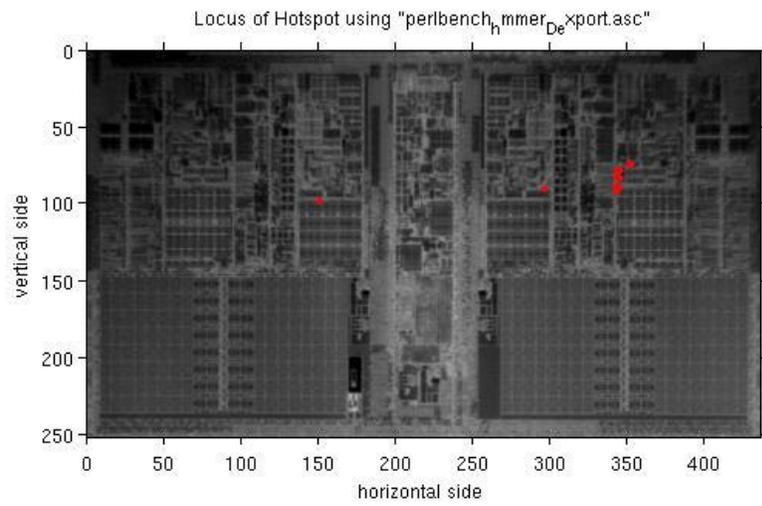
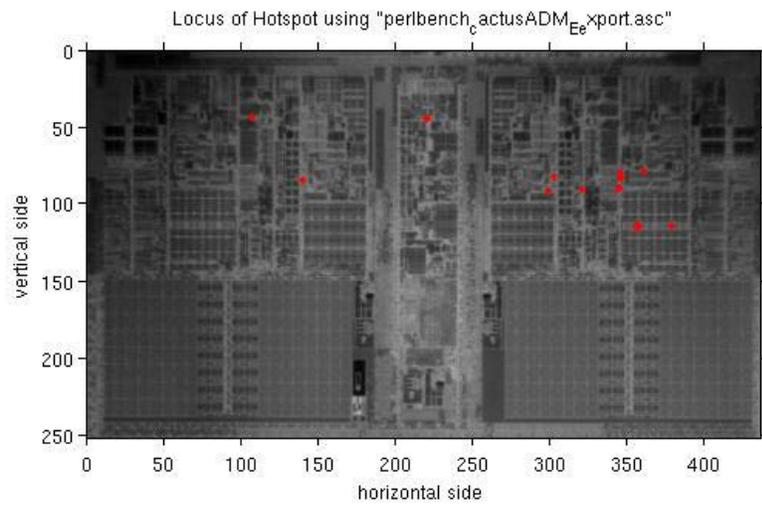
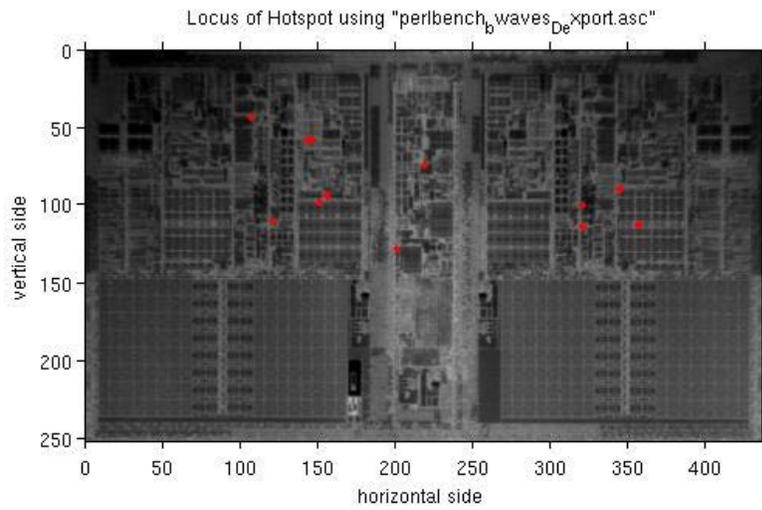
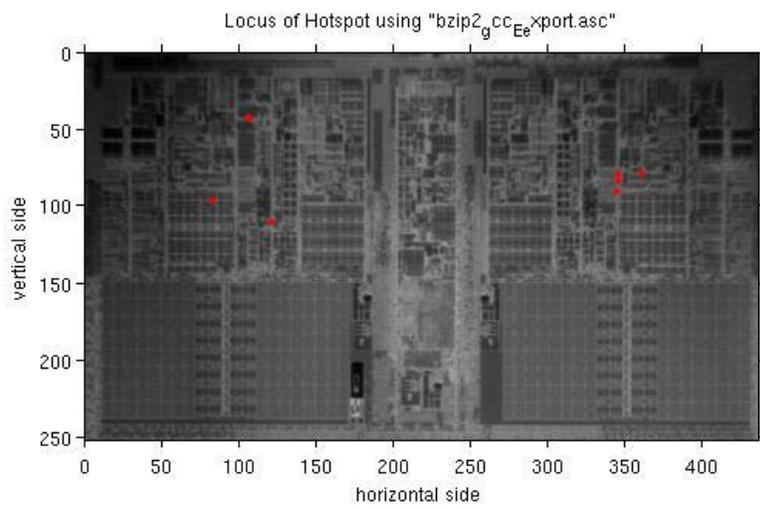
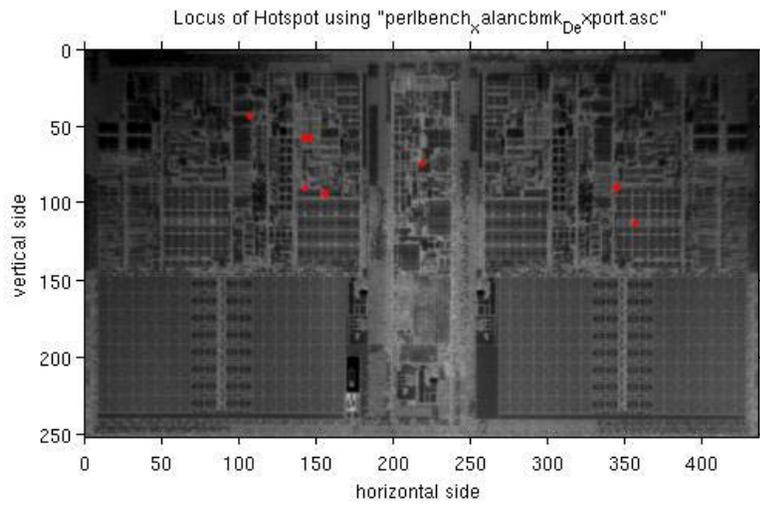
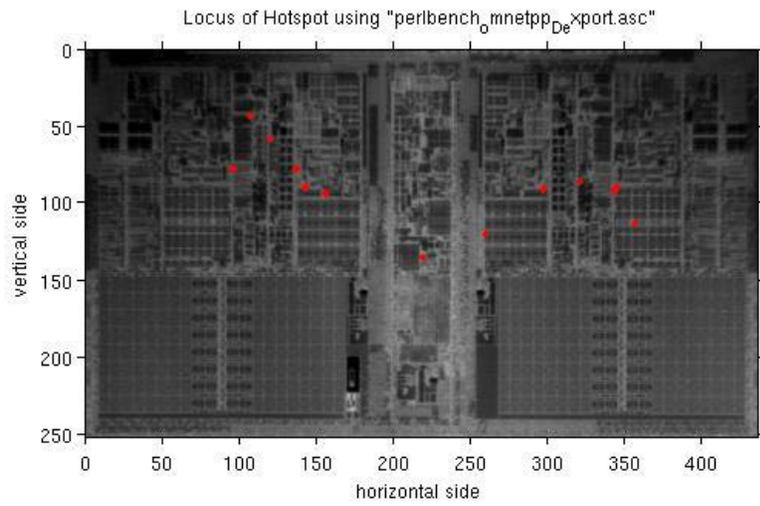
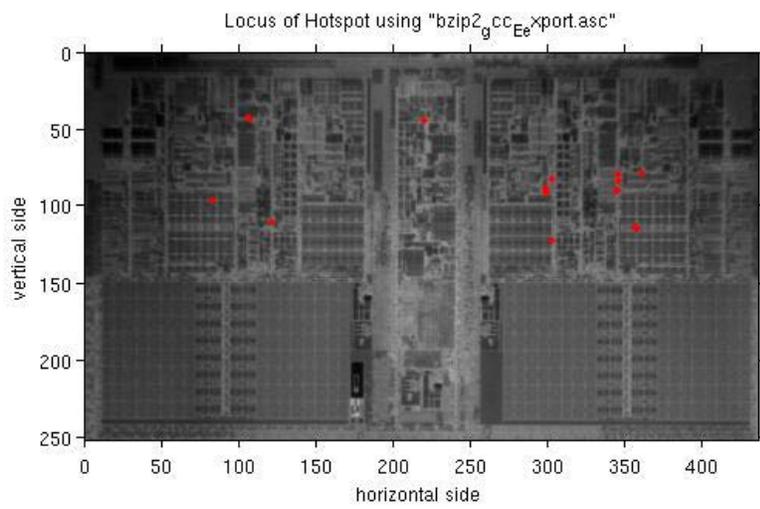
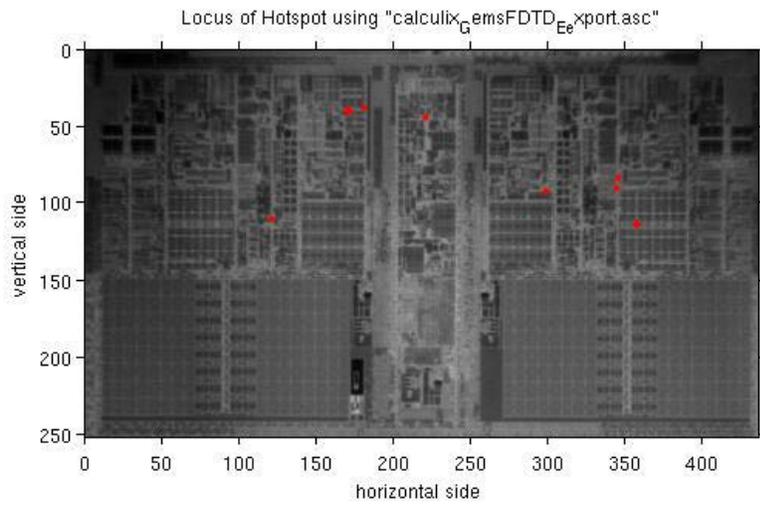
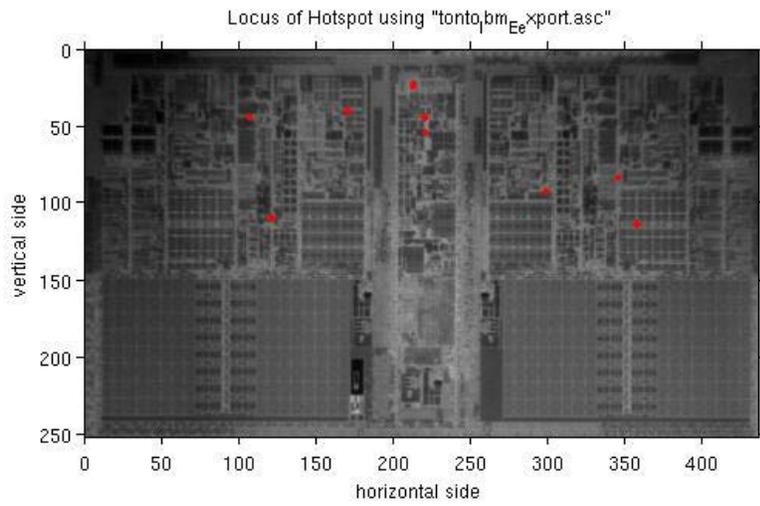


Table 7: Maximum DL for every workload pair

Benchmark Pairs	Max Data
perlbench - astar	6827
perlbench - bwaves	6626
perlbench - bzip2	6262
perlbench - gamess	6910
perlbench - gcc	6392
perlbench - gobmk	6558
perlbench - h264ref	6806
perlbench - hmmer	6846
perlbench - libquantum	6672
perlbench - mcf	6304
perlbench - omnetpp	6454
perlbench - sjeng	6547
perlbench - xalancbmk	6552
bzip2 - gcc	5076
cactusADM - leslie3d	4919
calculix - GemsFDTD	4893
gamess - milc	4908
hmmer - sjeng	4817
libquantum - h264ref	4981
mcf - gobmk	5005
namd - dealII	5016
omnetpp - astar	4942
perlbench - cactusADM	4861
perlbench - gromacs	4853
perlbench - leslie3d	4932
perlbench - milc	4862
perlbench - namd	5054
perlbench - zeusmp	4830
soplex - povray	5082
tonto - lbm	4835
xalancbmk - bwaves	4975
zeusmp - gromacs	4883







Based on the above results, we made some observations.

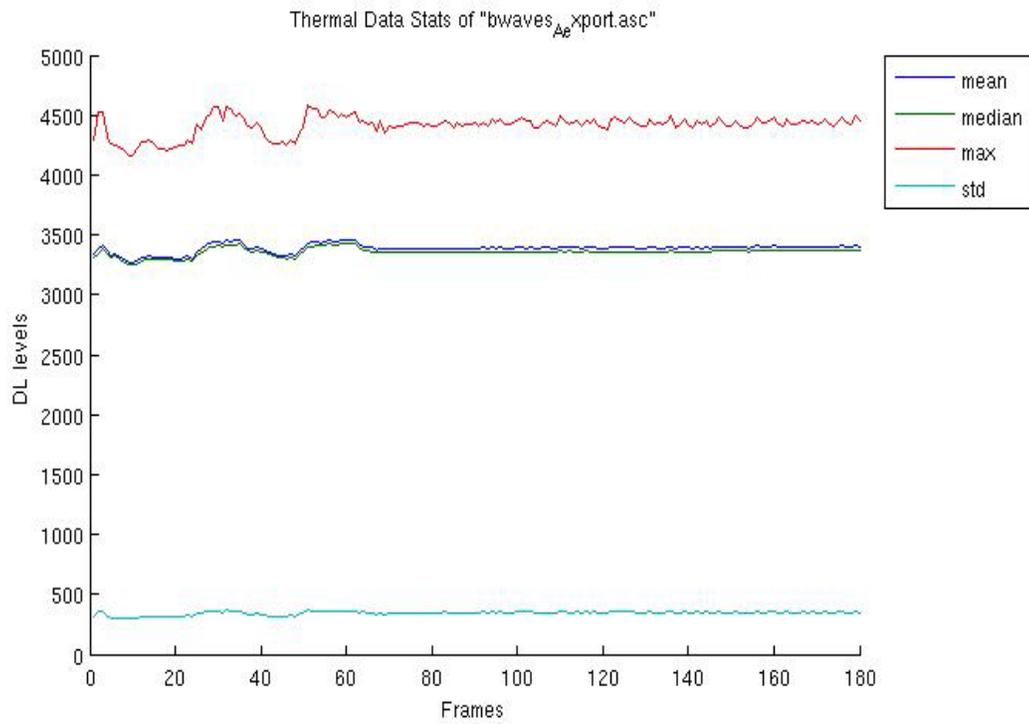
1. In almost all the locus plots, there are no hotspots in the cache or in the center column (memory controller) of the processor. This means that the hotspots are localized in the top left and right regions of the processor and the caches below and the center column remains cool, or experienced less thermal activity.
2. Also when there is an addition of one more workload to an existing workload, the hotspot locus of the initial workload is changed and hotspots appear on both halves of the processor.
3. One further observation is that most hotspots appear in particular locations, showing that there are certain spots on the chip that often experience hotspots

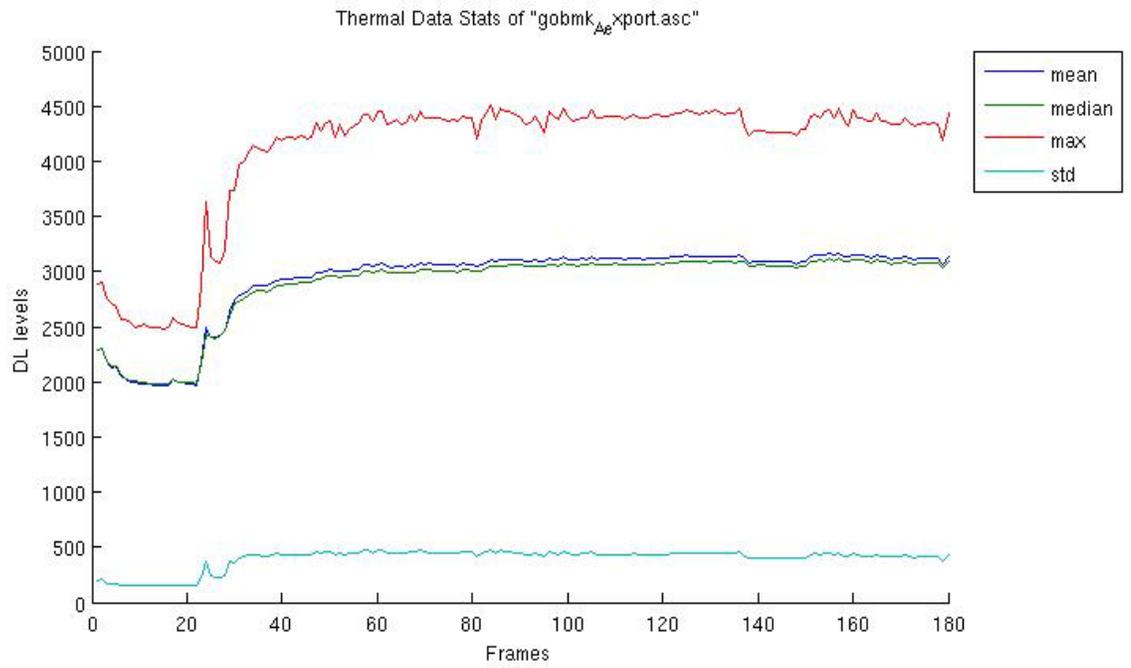
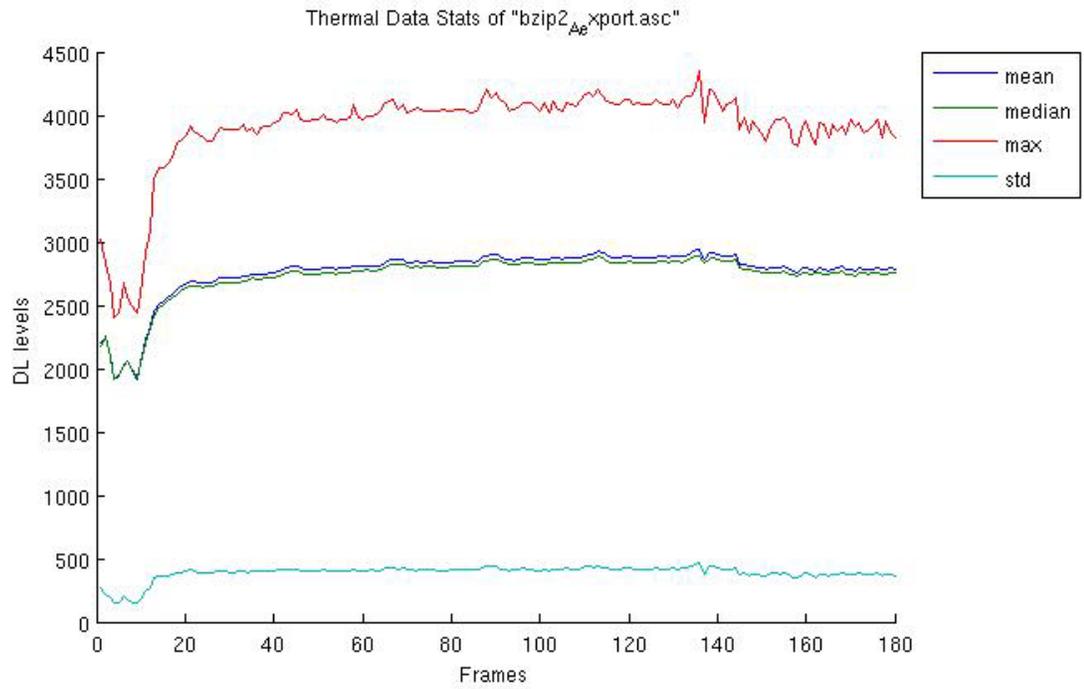
There are many uses of generating hotspot loci like we did. We could make use of these hotspot loci generated to better place thermal sensors by observing the ‘popular’ spots on the chip. We can also identify the areas of hotspots on the die layer corresponding to each workload or workload combinations to study the areas of the chip being affected by different benchmark applications. These hotspot loci could also be a means of evaluating the results of design time architectural thermal management techniques.

5.4 Experiment 2: Statistics of Thermal Behaviors of Different Benchmarks

This experiment will show the digital levels mean median, max and standard deviation of each workload and how these statistics vary across workloads.

I generated these results by transferring the thermal data into MATLAB and using the statistics functions to generate the following graphs. These graphs will show the maximum DL versus time over 180 frames at 2 Hz or 90 seconds.





Our observations are as follows.

1. I observed that the statistics of the thermal data of different workloads varies a lot among one another. Some have less variation in max DL such as leslie3d and bwaves after 70 frames or 35 seconds. Others were irregular and consist of peaks and dips such as bzip2 and gobmk.
2. The standard deviations are similar for different workloads at a value of below 500 DL.
3. The mean and median values usually reach a steady level ranging from 2500 to 3000 DL but the max DL has more variation than the mean and median. This indicates that the workloads have a more significant impact on localized temperatures rather than changing the temperatures by a similar extent across the entire die.
4. In most workloads, there is larger variation in DL for the initial 20 to 80 frames or 10 to 40 seconds before they reach a relatively stable level. This is perhaps due to the thermal variation encountered at the start of any workload execution.

5.5 Background on Variogram

In this section, I will discuss about the variogram and how it can be used to study the spatial thermal field of the various benchmark workloads. Very generally, a variogram shows the correlation of any two different data points of a particular distance, h from each other. Hence, given a set of data points, the variogram is found by finding the correlation of every possible pair of data points and forming a relation between the

correlation and the distance between the pair of data points. This is very useful for studying how thermal data collected from different benchmark workloads running, would correlate to one another, according to distance. This would give us a very good sense of spatial thermal behavior for each unique benchmark. For example, some workloads might have two hotspots that constantly correlate closely with each other throughout the runs while others might have hotspots that behave differently from each other at different times of the runs.

In probabilistic notion, the variogram is defined by the following equations.

$$2\gamma(h) = E[\{Z(u) - Z(u + h)\}^2] \quad \text{--- (3)}$$

$$2\gamma(h) = \frac{1}{N(h)} \sum_{N(h)} [\{Z(u) - Z(u + h)\}^2]$$

where h is the lag distance, or the distance between any two points in the data and Z represents the value of the data at the location u.

From the equation (3), we can observe that the variogram, $\gamma(h)$ varies proportionally with the expected value of the square of every pair of data points with a lag distance of h between them.

From (3),

$$2\gamma(h) = E[Z(u)^2] + E[Z(u + h)^2] - 2E[Z(u)Z(u + h)] \quad \text{--- (4)}$$

$$\mathbf{Var}(X_i) = E[X_i^2] - E[X_i]^2$$

From (4),

$$2\gamma(h) = \text{Var}(Z(u)) + E(Z(u))^2 + \text{Var}(Z(u+h)) + E(Z(u+h))^2 - 2E[Z(u)Z(u+h)] \quad \text{--- (5)}$$

$$\text{Cov}(X_i, Y_i) = E[X_i Y_i] - E[X_i]E[Y_i]$$

From (5),

$$2\gamma(h) = \text{Var}(Z(u)) + E(Z(u))^2 + \text{Var}(Z(u+h)) + E(Z(u+h))^2 - 2\text{Cov}(Z(u)Z(u+h)) - 2E(Z(u))E(Z(u+h)) \quad \text{--- (5)}$$

In a stationary field,

$$E(Z(u)) = E(Z(u+h))$$

$$\text{Var}(Z(u)) = \text{Var}(Z(u+h))$$

Hence, from (5),

$$\gamma(h) = \text{Var}(Z) - \text{Cov}(Z(u)Z(u+h)) \quad \text{--- (6)}$$

From (6), we can observe this relationship from an example of a variogram graph below which is generated by a software developed by ‘Surfer of Golden Software, Inc.’

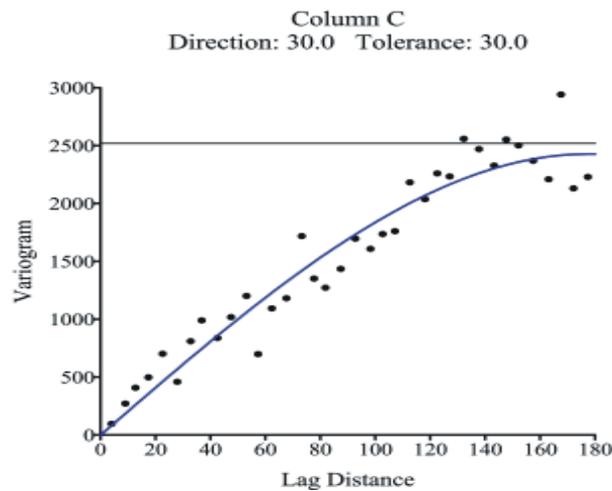


Figure 19: Example of Variogram graph by Surfer of Golden Software, Inc.’

The maximum level of the variogram corresponds to the variance of the data field, beyond which there is no correlation between any two points. Hence, it is reasonable that the longer the lag distance, the less correlated one point is to the other. The covariance is described in this graph by the difference between the highest line and the variogram. It decreases with an increasing lag distance, indicating that there is zero covariance at the lag distance when the variogram is equal to the variance of the data field.

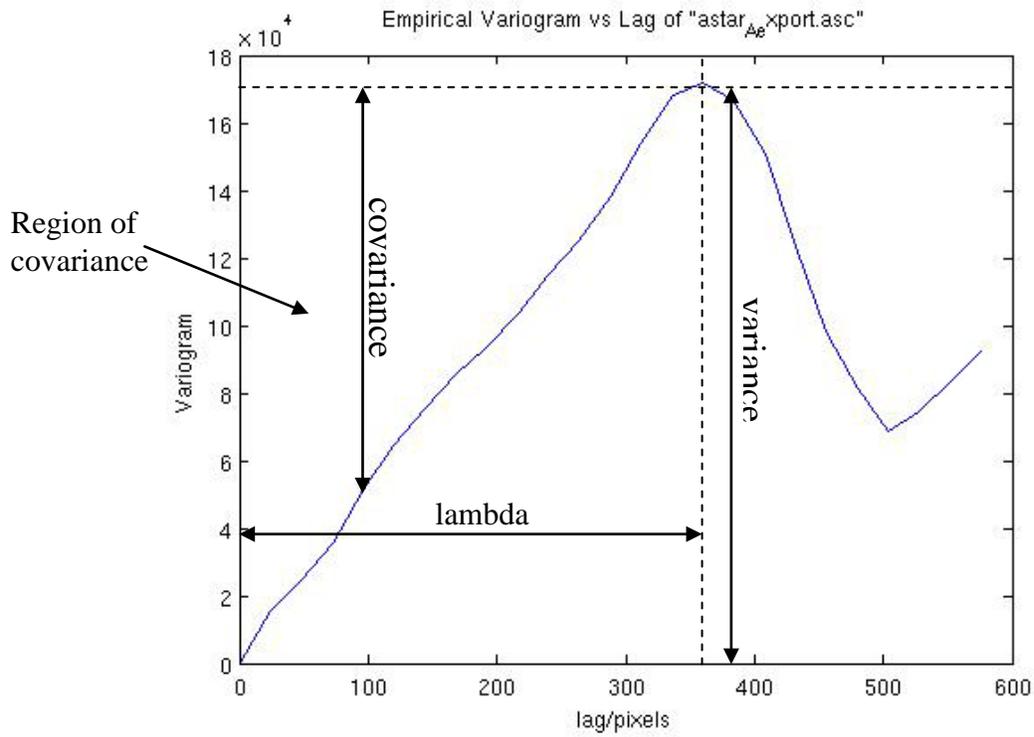
5.6 Experiment 3: Variogram of Various Benchmarks

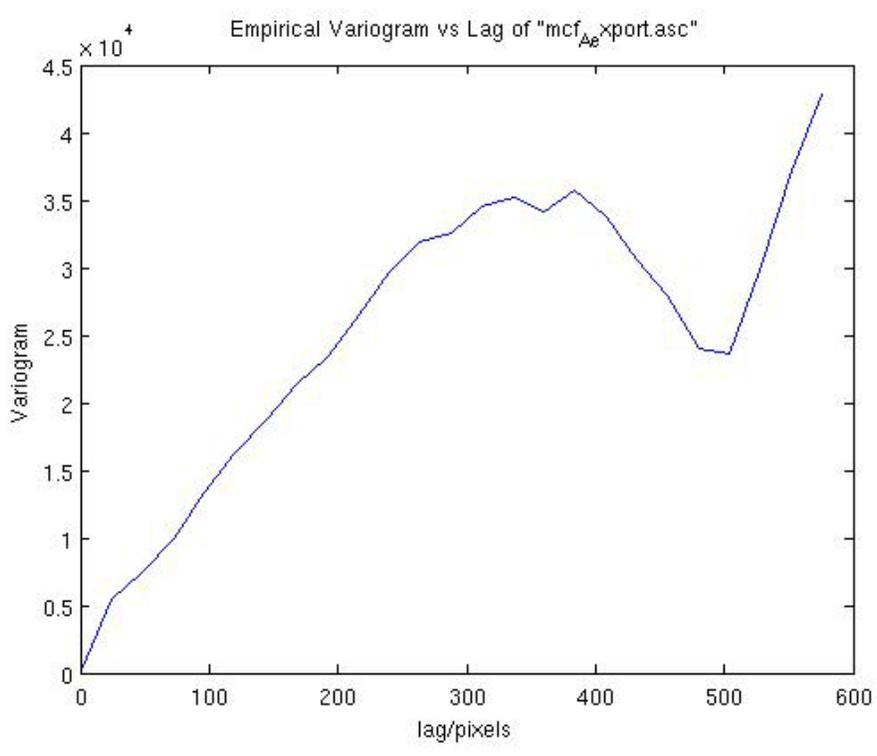
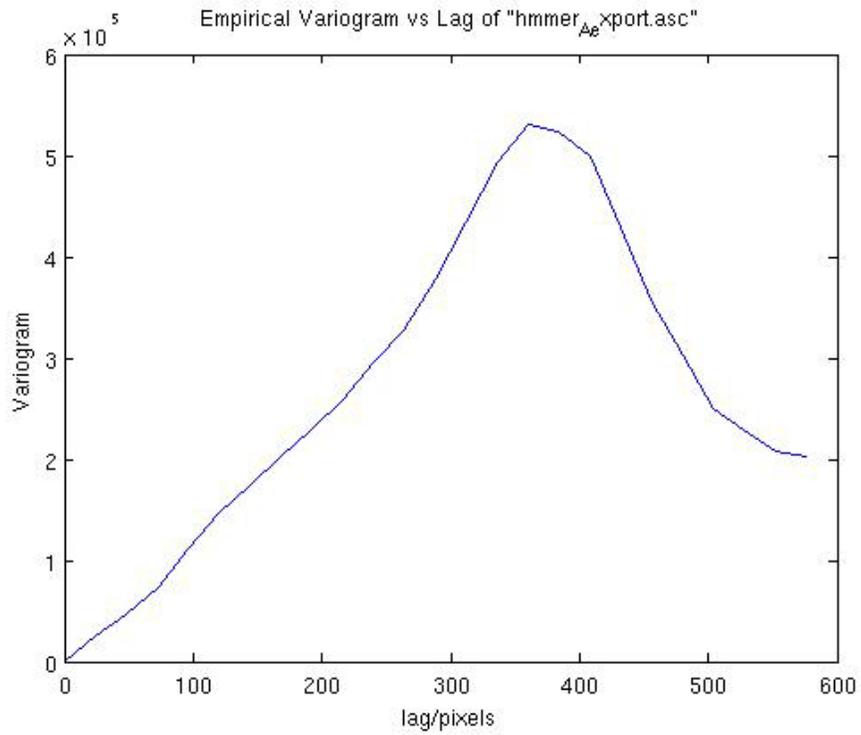
Since the thermal data I collect is too large, I have to define a sampling interval to downsample the thermal data to a more manageable size. I also have to define a lag interval, h for the variogram to collect enough thermal samples in ‘buckets’ at regular spatial intervals. The lag distance refers to the spatial Euclidian distance between any two data points in the data field. Each bucket would correspond to a certain value of lag distance, h and generates the variogram with the thermal samples collected in that bucket. In my experiments, I defined these two sampling and lag intervals to be 24 pixel intervals. The size of the pixel is 14 μm and the pixel pitch is 30 μm as could be seen in the radiometric data table (Table 5, page 41). Thus the distance of pixel interval or distance between adjacent pixels is 30 μm . I generated variograms for both single and double workloads running and tabulated the values of λ , the lag distance at which the variogram is equal to the variance, or distance between any two data points with the least covariance. As seen in the previous section, the equations are repeated below.

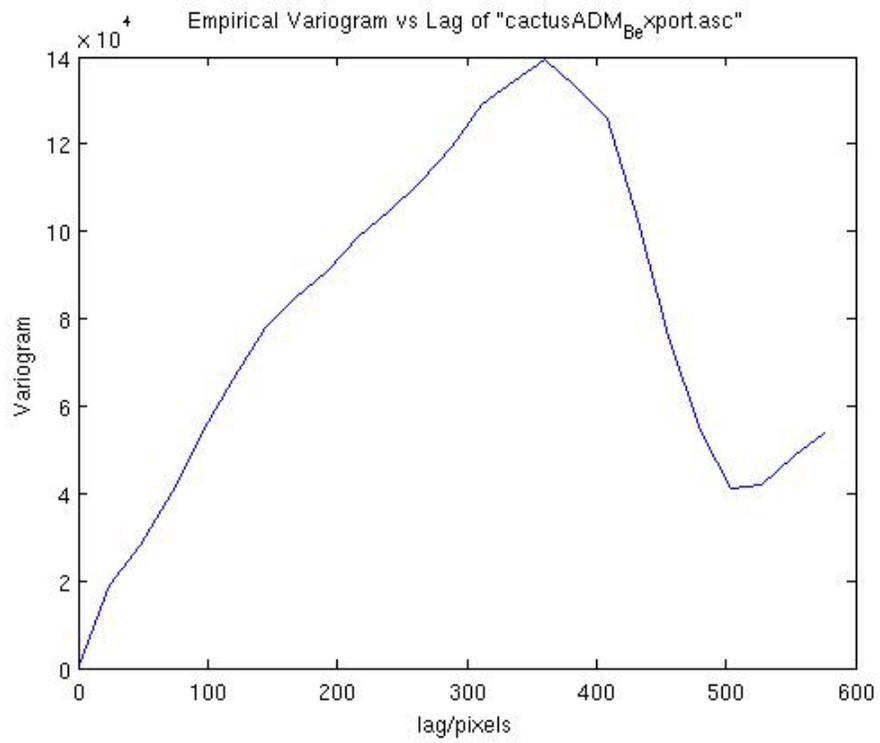
$$\gamma(h) = \text{Var}(Z) - \text{Cov}(Z(u)Z(u+h)) \text{ --- (6)}$$

where γ is the variogram, Z is field, u is the local coordinates and h is the lag distance

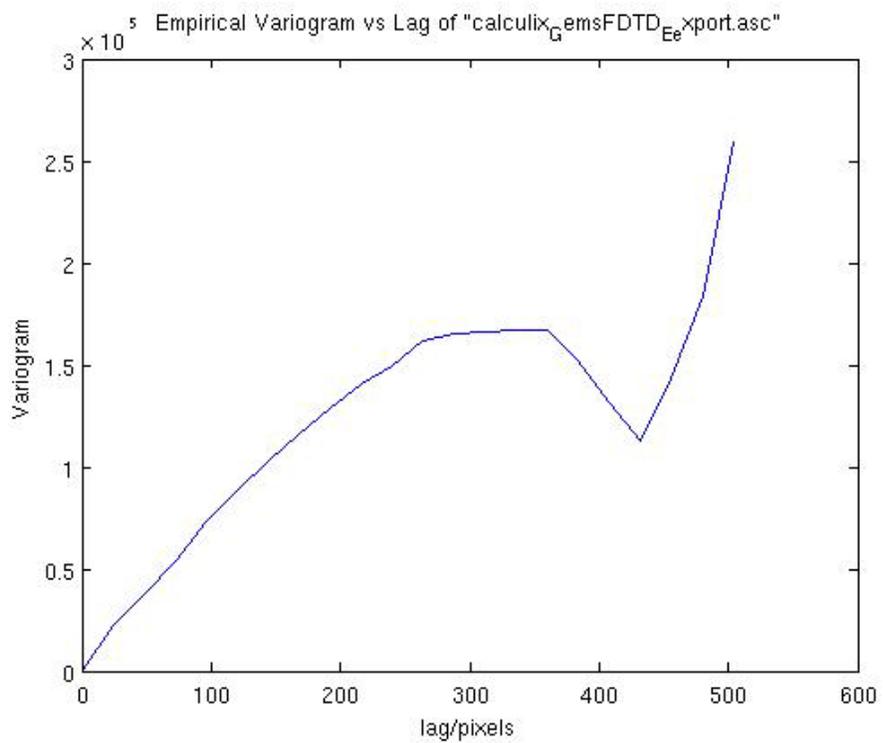
Single Workload

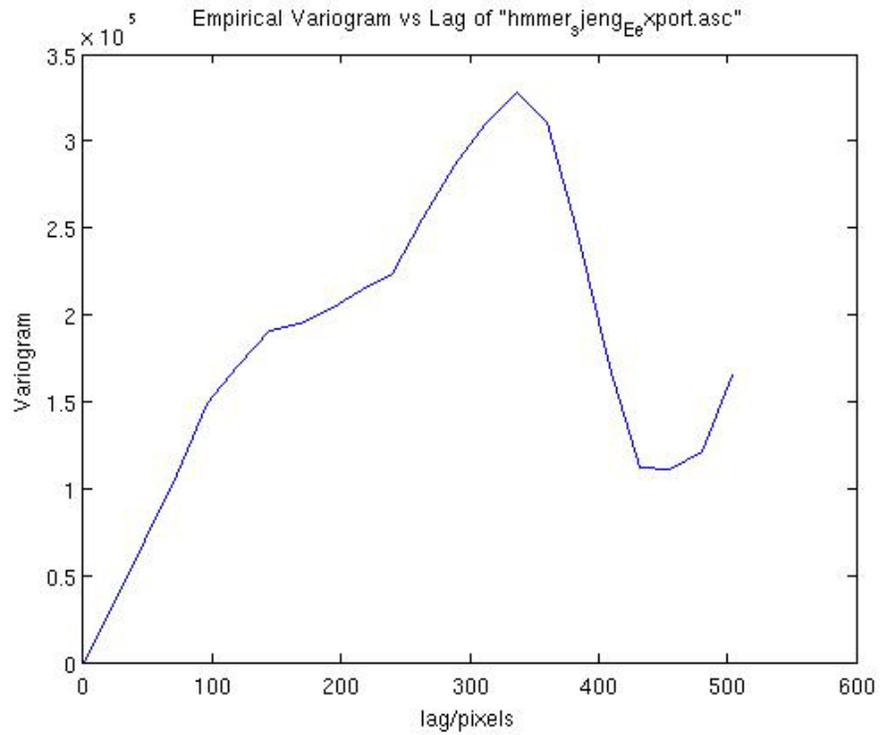
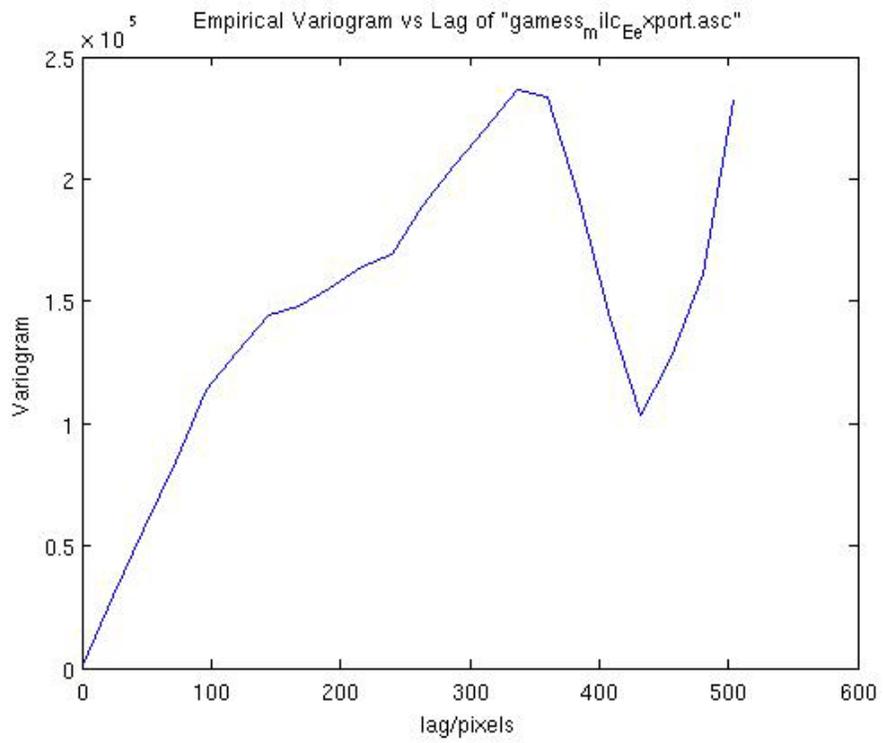


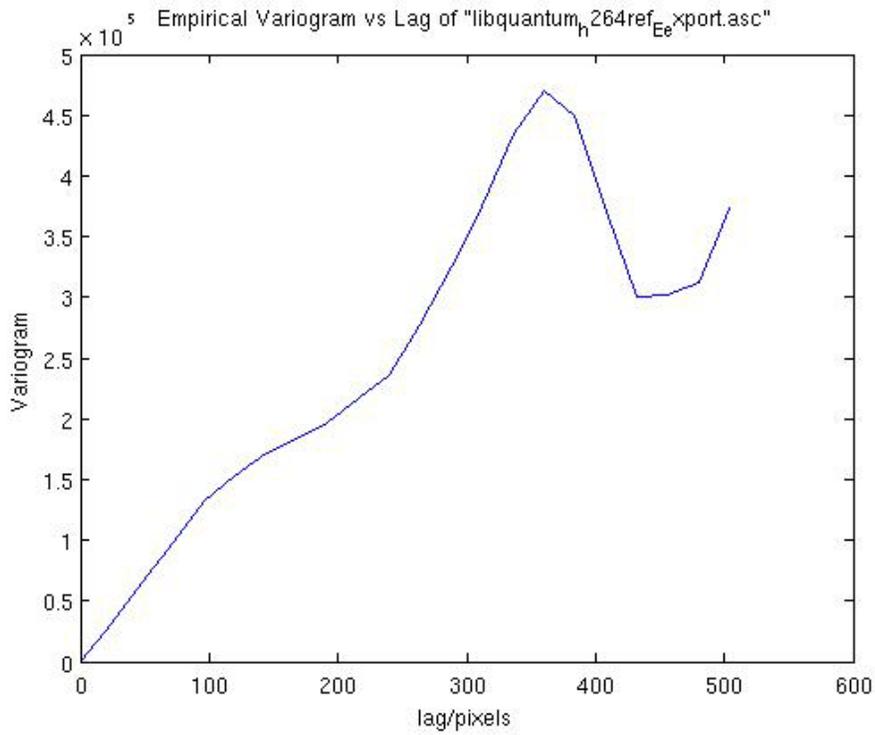




Two Workloads







My observations are tabulated below.

Table 8: Values of Lambda and Variance for every workload

SPEC CPU2006 Benchmarks Workloads	Lambda/ pixels (will be converted to mm)	Variance/ 10^5
astar	360	1.7
bwaves	360	2.1
bzip2	360	2.45
h264ref	360	5
hmmer	360	5.3
libquantum	360	3.8
mcf	380	0.35
omnetpp	360	1.8
perlbench	360	3.55
cactusADM	360	1.4
gamess	360	4.7
bzip2-gcc	360	2.55
calculix-GemsFDTD	360	1.7
games-milc	335	2.4
hmmer-sjeng	335	3.3
libquantum-h264ref	360	4.7
mcf-gobmk	360	10

My observations are listed below.

1. I observed that for single workloads, the processor shows similar thermal behavior in variogram. The value of the variogram rises with lag distance almost linearly to a peak before falling to almost half the rise and rising by a little again or not at all, near the edge of the processor. This means that as the distance between any pair of data points increases, the less correlated they are. This happens until the peak which represents the variance of the data field. The reason for the fall is due to the similarity of the temperatures at the boundaries where there is little activity at the edges. Hence, most of the temperatures there are similarly low in value and in that sense would be more correlated to each other.
2. For dual workloads, there is more variation in the shape of the variograms across the different pairs of workloads. One characteristic I observed in certain pairs is the slight presence of a second peak at about 50 % of the lag distance of λ , especially in the second and third graphs with two workloads. This could be explained by an increasing 'unrelatedness' or decreasing covariance between closer data points due to two concurrent workloads which are completely different and independent of each other. The presence of two workload activities on the same processor region occupies more space of activity than space of non activity in the single workload processor. Hence, it is expected that the region of covariance as represented

by the space above the variogram and below the variance level decreases in area due to a decrease in region of non activity on the processor, which has a high covariance.

3. Another observation is the variance in each graph is different, ranging from 0.3×10^5 to 5.5×10^5 . This shows that different workloads or workload combinations have different spatial thermal behaviors. The presence of two workloads instead of one does not seem to affect the value of variance in the thermal data field. Perhaps each combination exudes a spatial thermal behavior in terms of variance that is unique to itself.
4. I also observed that for all of the graphs the values of lambda or lag distance which corresponds to the variance of the field are almost similar. This implies that the lambda is independent of the type of workloads running on the processor or that the distance between any two points that are completed uncorrelated in temperature is always the same. This might be because there is a maximum distance on the processor where it is physically impossible for temperature at one point to affect temperature at the other point. Perhaps one of the factors that affect the value of lambda is the thermal conductivity of the material in use which can contribute to lateral heat spreading.

Chapter 6: Conclusions

6.1 Conclusions of Experimental Results

In the first experiment comparing thermal measurements from thermal sensors and thermal camera, we see a close correspondence in thermal readings from these two different measuring instruments and hence can conclude that the on-chip thermal sensors are relatively accurate. However, one can also observe that the readings for thermal sensors are more discrete with less variation than the thermal camera due to a slower response, limited resolution and rigid placements of the thermal sensors. Also, the thermal sensor failed to capture certain variations of particular period and magnitude that is caught by the thermal camera.

In the second experiment comparing thermal readings using IR imaging on two different cooling configurations, we first observed that the oil heat sink setup has a larger range of max temperature values than the metal heat sink setup. Secondly, the oil heat sink has a slower short term transient response but a faster long term transient response as compared to the metal heat sink setup which is due to a difference in thermal resistance and capacitance of the materials contributing to the cooling configurations.

In our thermal characterization involving hotspot locus of different workloads and workload combinations, we observed that the hotspots are localized in the top left and right regions of the processor and the caches below and the center column remains cool. Also when there is an addition of one more workload to an existing workload, the hotspot locus of the initial workload is changed and hotspots appear on both halves of the

processor. One further observation is that there are certain spots on the chip that often experience hotspots.

In the next thermal characterization involving statistical characterization of thermal behavior of various workloads, I observed that the statistics of the thermal data of different workloads varies a lot among one another. The standard deviations are similar for different workloads at a value of below 500 DL. The mean and median values usually reach a steady level ranging from 2500 to 3000 DL but the max DL has more variation than the mean and median indicating that the workloads have a more significant impact on localized temperatures rather than changing the temperatures by a similar extent across the entire die.

In our final characterization of thermal behaviors showing the variograms of different workloads, we observed that for single workloads, the processor shows similar thermal behavior in variogram. For dual workloads, there is more variation in the shape of the variograms across the different pairs of workloads and there is a second peak at about 50 % of the lag distance of λ , indicating a lower covariance between any two thermal data points in general. The variance in each graph is different, showing that different workloads or workload combinations have different spatial thermal behaviors. We also observed that for all of the graphs the values of λ or lag distance which corresponds to the variance of the field are almost similar, indicating that the lag distance of lowest covariance between any two points is independent of the type of workloads.

6.2 Future Directions

We observed the differences and similarities in thermal measurements collected by two different measuring instruments, thermal sensors and thermal camera, in our work. A further improvement of this work could be exploring the effect of different allocations of thermal sensors on thermal measurements collected. This will assist in the development of better techniques for allocation of sensors on the chip to capture hotspots. Furthermore, we would be able to achieve a better comparison by using the IR camera to obtain the real temperature of the processor instead of digital levels. This could be done by finding the relationship between DL and temperature of the processor and calibrating the IR camera accordingly.

We have also compared thermal measurements collected by the IR camera on the processor in an oil heat sink and a metal heat sink. However, there are other factors such as oil flow speed, oil flow direction and oil temperature that are worth considering. By adjusting these parameters, we can better match the cooling configuration in a typical metal heat sink and perform a more accurate thermal measurement and characterization of real processors under typical conditions.

Finally, in the thermal characterization of various benchmark workloads, a further improvement to our work could be generating the fourier transforms of the thermal data collected on different workloads and studying them. Moreover, we have allowed the processor to run at 1.6 and 2.8 GHz in our experiments in this work but have not really explored the effect of different frequencies on thermal characterizations of workloads. Thus, this could be also worth investigating.

References

- [1] K. Skadron et al., "Temperature-aware microarchitecture: Modeling and Implementation," *ACM Trans. Archit. Code Optim.*, vol. 1, 2004, p. 94-125.
- [2] F.J. Mesa-Martinez et al., "Measuring Performance, Power, and Temperature from real processors," *Proceedings of the 2007 workshop on Experimental computer science*, San Diego, California: ACM, 2007, article 16.
- [3] F.J. Mesa-Martinez, J. Nayfach-Battilana, and J. Renau, "Power Model Validation Through Thermal Measurements," *International Symposium on Computer Architecture*, 2007, p. 302 - 311
- [4] H. F. Hamann, A. Weger, J. A. Lacey, Z. Hu, P. Bose, E. Cohen and J. Wakil, "HotSpot Limited Microprocessor: Direct Temperature and Power Distribution Measurements," *IEEE Journal of Solid-State Circuits*, 2007, 42(1):56-65
- [5] W. Huang, K. Skadron, S. Gurumurthi, R. Ribando, and M. Stan, "Differentiating the Roles of IR Measurement and Simulation," *Proceedings of the 2009 international Symposium on Performance Analysis of Systems and Software*, 2009, p. 1-10
- [6] F.J. Mesa-Martinez, E. K. Ardestani, J. Renau, "Characterizing Processor Thermal Behavior," *Proceedings of the fifteenth edition of ASPLOS on Architectural support for programming languages and operating systems*, 2010, p. 193-204
- [7] P. G. Collins, P. Avouris, "Nanotube for Electronics," *Scientific American*, Dec 2000
- [8] R. Martel, V. Derycke, J. Appenzeller, S. Wind, Ph. Avouris, "Carbon Nanotube Field-Effect Transistors and Logic Circuits," *Proceedings of the 39th annual ACM Design Automation Conference*, 2002, p. 94-98
- [9] Z. Yu, S. Paul, S. Bhunia, "Towards Uniform Temperature Distribution in SOI Circuits Using CNT Based Interconnect," *Quality Electronics Design, 2008, 9th International Symposium, Proceedings of the 9th international symposium on Quality Electronic Design*, 2008, p. 861-866
- [10] E. Pop, R. Dutton, and K. Goodson. "Thermal Analysis of Ultra-Thin Body Device Scaling." *IEEE International Electron Devices Meeting*, 2003, p. 36.6.1-4,

- [11] B. Swahn, S. Hassoun, "Thermal and Sensitivity analysis of Multifin Devices," *Proceedings of 12th International Workshop on Thermal investigations of ICs* Nice, Côte d'Azur, France, 2006, p. 229-234
- [12] J. W. Che, T. Cagin, W. A. Goddard III, "Thermal Conductivity of Carbon Nanotubes," *2000 Nanotechnology 11 65, Vol 11 No. 2*, 2000, p. 65-69
- [13] B. Swahn, S. Hassoun, "Electro-Thermal Analysis of Multi-Fin Devices," *Volume 16, Issue 7*, 2008, p. 816-829