Accurate Pre-RTL Temperature-Aware Design Using a Parameterized, Geometric Thermal Model

Wei Huang, *Member*, *IEEE*, Karthik Sankaranarayanan, Kevin Skadron, *Senior Member*, *IEEE*, Robert J. Ribando, and Mircea R. Stan, *Senior Member*, *IEEE*

Abstract—Preventing silicon chips from negative, even disastrous thermal hazards has become increasingly challenging these days; Considering thermal effects early in the design cycle is thus required. To achieve this, an accurate yet fast temperature model together with an early-stage, thermally optimized, design flow are needed. In this paper, we present an improved block-based compact thermal model (HotSpot 4.0) that automatically achieves good accuracy even under extreme conditions. The model has been extensively validated with detailed finite-element thermal simulation tools. We also show that properly modeling package components and applying the right boundary conditions are crucial to making full-chip thermal models like HotSpot accurately resemble what happens in the real world. Ignoring or oversimplifying package components can lead to inaccurate temperature estimations and potential thermal hazards that are costly to fix in later designs stages. Such a full-chip and package thermal model can then be incorporated into a thermally optimized design flow, where it acts as an efficient communication medium among computer architects, circuit designers, and package designers in early microprocessor design stages to achieve early and accurate design decisions and also faster design convergence. For example, the temperature-leakage interaction can be readily analyzed within such a design flow to predict potential thermal hazards such as thermal runaway. An example SoC design illustrates the importance of adopting such a thermally optimized design flow in early design stages.

Index Terms—Compact thermal model, early design stages, leakage, parameterized model, temperature, thermally optimized design flow.

1 Introduction

Because of the continued nonideal scaling of CMOS technology [1], managing on-chip temperatures, especially local hot spots, has become a major challenge. To deal with this thermal challenge, temperature-aware design in early stages, such as *microarchitecture design*, is especially important because the architecture definition fixes what subsequent design stages such as circuit implementation, packaging, etc., must accommodate and has the greatest impact on final design.

Temperature-aware design in early, pre-Register Transfer Level (RTL) design stages, in turn, requires a fast, yet accurate, architectural thermal model to explore large regions of the design space. Such a thermal model should be "by-construction" and parameterized, i.e., the model is constructed solely based on chip and package geometries and material properties, hence allowing a designer to explore potential design choices without the costly slow building of a prototype [2].

The complicated 3D heat transfer within both the silicon chip and the package, together with the closely coupled relationship between power (density) and temperature requires that such a thermal model be accurate even under extreme simulated conditions. While better accuracy in general means less computational efficiency, an early-stage, by-construction, full-chip thermal model can still achieve satisfactory accuracy by carefully correcting deficiencies in the model structure that lead to significant errors, without sacrificing the speed advantage from its compact nature. For example, in a microarchitecture floorplan, it is not uncommon to have functional blocks with relatively high aspect ratios. Modeling these high-aspect-ratio functional blocks as single nodes is less accurate than dividing them into a few more subblocks with aspect ratios close to unity, as we will see later in this paper. The flexibility in refining a functional block also validates the fact that the intuitive, parameterized, and by-construction modeling paradigm works well.

In addition to modeling the silicon chip, the early-stage compact thermal model should also properly model different package components. Ignoring or oversimplifying package components in a full-chip thermal model can lead to inaccurate temperature estimations and potential thermal hazards that are costly to fix in later design stages. For

- *W. Huang, K. Sankaranarayanan, and K. Skadron are with the Department of Computer Science, School of Engineering and Applied Science, University of Virginia, 151 Engineer's Way, Box 400740, Charlottesville, VA 22904-4740. E-mail: wh6p@virginia.edu, {ks4kk, skadron}@cs.virginia.edu.*
- *R.J. Ribando is with the Department of Mechanical and Aerospace Engineering, University of Virginia, 122 Engineer's Way, PO Box 400746, Charlottesville, VA 22904-4746. E-mail: rjr@virginia.edu.*
- *M.R. Stan is with the Department of Electrical and Computer Engineering, University of Virginia, 351 McCormick Rd., PO Box 400743, Charlottesville, VA 22904-0743. E-mail: mircea@virginia.edu.*

Manuscript received 26 Sept. 2007; revised 14 Feb. 2008 accepted 18 Mar. 2008; published online 4 Apr. 2008.
Recommended for acceptance by A. Gonzalez.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2007-09-0485.
Digital Object Identifier no. 10.1109/TC.2008.64.

0018-9340/08/$25.00 © 2008 IEEE Published by the IEEE Computer Society

example, the thermal interface material (TIM) is a thin layer bonding silicon chip and heat spreader. Due to its low thermal conductivity, TIM prevents effective heat spreading from silicon to the rest of the package and thus exacerbates localized heating within the die. Therefore, ignoring TIM or using the wrong TIM thickness in the model causes unrealistic silicon temperature estimates. Another example is the thermal boundary condition at the heatsink-air interface. Traditional thermal models usually assume an isothermal condition with a single thermal resistor connecting the heatsink surface to ambient air. In reality, a convective boundary condition is more appropriate as the heatsink surface is usually far from isothermal. Using the proper boundary condition can greatly improve the accuracy of the thermal model.

Consequently, an accurate full-chip and package compact thermal model can also act as a convenient medium for enhanced collaborations among circuit, architecture, and package designers. This implies a design flow leading to early design evaluations from a thermal point of view. If potential thermal hazards are discovered early in the design process, different design trade-offs can be carried out at the architecture level, the circuit level, and the package level in an efficient way. For example, it is well known that subthreshold leakage power is exponentially dependent on operating temperature. An accurate early-stage thermal model can efficiently close the temperature-leakage loop and warn of potential thermal disaster such as thermal runaway very early in the design process.

In this paper, we address the above topics and make the following contributions:

1. We identify sources of inaccuracies in a by-construction early-stage architecture-level thermal model and provide solutions to improve the accuracy under extreme conditions such as blocks with high aspect ratios and high power densities. We use the popular HotSpot thermal model [3] as the base case. All of the proposed solutions are implemented in the new HotSpot Version 4.0 [4].
2. We demonstrate the importance of modeling package components and using a proper thermal boundary condition, leading to a more useful *full-chip and package* thermal model that accurately resembles the temperature distribution in real processors and other IC designs.
3. We propose a thermally optimized design flow based on HotSpot 4.0 for early design stages. The design flow involves designers at all abstraction levels, who collaborate efficiently with the help of HotSpot and reach a thermally optimized design with faster design convergence and less design cost. We also show a potential leakage-induced thermal runaway example which demonstrates the importance of the proposed design flow.

This paper is organized as follows: Section 2 briefly introduces HotSpot, which is the thermal model we use for experiments and analysis throughout the paper. It also reviews other related work. Section 3 identifies the weakness of the generic by-construction modeling method and provides solutions to improve its accuracy. Section 4 shows the results of the proposed improvements. Following that, Section 5 proposes the thermally optimized design flow that can catch potential thermal hazards such as leakage-induced thermal runaway during early design stages efficiently. Section 6 summarizes the work.

2 Related Work

The HotSpot [3] thermal model is widely used by the computer architecture research community. To date, HotSpot seems to have been mostly used with existing architectural simulation infrastructures such as SimpleScalar[1] and Wattch [5], but it is designed as a portable library that can be used with a wide range of modeling infrastructures. HotSpot has a by-construction parameterized structure and is available online.[2]

HotSpot was first introduced only as a block-based model. Later on, a regular-grid-based HotSpot model [6] was also introduced. One major reason to develop the grid model was to achieve more accuracy by modeling lateral heat transfer paths in more detail than the block model. The irregular block model of HotSpot is suitable for fast thermal simulations with arbitrarily sized functional blocks. In contrast, the HotSpot grid model achieves more detailed temperature estimations at the cost of more computational overhead. The importance of having a grid-like thermal model was also discussed in [7].

There are numerous other existing chip level temperature models besides HotSpot. Among them, the most accurate models are the detailed finite-element models such as ANSYS,[3] FloWorks,[4] and FreeFEM3d,[5] which unfortunately are very computationally intensive and time consuming. There are also other thermal models dividing silicon into fine meshes and solving with fast solvers such as [8], [9] and the HotSpot grid model [10]. These models also achieve excellent accuracy while still incurring significant computational overhead compared to the *parameterized compact* thermal models such as the HotSpot block model [2], [3]. On the other hand, the compact thermal models trade off absolute accuracy with simpler structure and speed by constructing the model directly according to functional units of interest and physical properties of the chip. Therefore, they are well suited to the fast transient thermal simulations required in computer architecture research. This "by-construction" nature also makes the thermal model parameterized and allows designers to explore hypothetical designs easily without building prototypes.

Regarding transient thermal modeling, another previous work [11] approaches the topic analytically at a finer granularity—the transistor level. Since the size of a transistor is much smaller than the die thickness, silicon can be modeled as semi-infinite, which greatly simplifies the boundary conditions and makes an analytical transient heat transfer solution possible. With the semi-infinite silicon assumption, heat can be fully spread within silicon before

1. http://simplescalar.com.
2. http://lava.cs.virginia.edu/HotSpot/.
3. http://www/ansys.com.
4. http://www.solidworks.com/pages/products/cosmos/cosmosfloworks.html.
5. http://www.freefem.org/ff3d/.

reaching the back surface of the silicon substrate, leading to a smaller thermal resistance and also a shorter thermal time constant. On the contrary, the HotSpot model aims at granularities coarser than transistors and the block size or grid size are usually comparable to or greater than the die thickness, rendering the boundary conditions assumed in [11] not valid. With a finite silicon thickness, the heat generated from a block cannot be fully spread before reaching the back surface of the die, causing a larger thermal resistance and also a longer thermal time constant. This difference in silicon thermal time constant leads to slower transient temperature changes in HotSpot that models larger blocks and grid cells than the model in [11], which models tiny transistors.

So far, models such as the HotSpot block model have been successfully helping computer architects in their temperature-aware research. However, there is still room to improve their accuracy and usefulness further without introducing significant computational overhead. Recently, some accuracy concerns were raised regarding the HotSpot block model [12]. Noticeable and even significant errors were found under certain evaluation scenarios. All of these scenarios contain extreme configurations (e.g., functional blocks with very high aspect ratios) or uncommon designs (e.g., extremely high power densities). In this paper, we extend the discussions in [4] to analyze the sources of inaccuracies for the by-construction compact thermal modeling approach and provide solutions to improve the accuracy even under the aforementioned extreme conditions, which is an important improvement of our previous work [3], [10], [13].

Another important factor that greatly impacts the accuracy of chip-level thermal models is how accurately the thermal package components are modeled and how realistic the boundary conditions are applied are. In recent years, there have been a number of existing full-chip thermal models that provide detailed die temperature distributions, such as [14], [15], [16]. These models all have detailed temperature distribution information across the silicon die and can be solved efficiently. Unfortunately, a limitation of the above models is that the thermal package is oversimplified. For example, the TIM that greatly affects die temperature distribution is not included in the models. The bottom surfaces of the silicon substrate, the heat spreader, and the heatsink are all treated as isothermal, which significantly deviates from the real-world convective thermal boundary condition and introduces errors. On the other hand, properly modeling package components and their boundary conditions can significantly improve the model's accuracy and usefulness. Ignoring or oversimplifying the package components can lead to inaccurate temperature estimations and, hence, incorrect design decisions. In comparison, there are also several package-only compact thermal models [17], [18], [19]. These package models consist of simple networks of thermal resistances whose values are extracted by data-fitting from the results of accurate but time-consuming detailed numerical package thermal model simulations (e.g., using the finite element method). Therefore, they are not fully parameterized and cannot be easily used to explore new package designs. In addition, these package thermal models can provide only one or a few die-level temperatures, which is far from enough for fine-grained die-level designs. In this paper, we extend the discussions in [20] to show the importance of modeling both chip and package components in a thermally optimized design flow.

With the improved accuracy and the inclusion of package components, a parameterized compact thermal model can be a convenient communication medium among architects, circuit designers, and package designers. In this paper, we also outline a thermally optimized design flow for early design stages. With the proposed design flow, potential thermal hazards such as leakage-induced thermal runaway should be discovered as early in the design process as possible. With the help of a compact chip and package level thermal model, across-die temperature distribution can be estimated at design time, which permits thermally self-consistent leakage power calculations in an iterative manner, as shown in [21], [22]. This is illustrated by an example of potential thermal runaway for an SoC design.

3 Accuracy Improvements

This section identifies weaknesses that have come to light in an earlier HotSpot block model when used with extreme simulation parameters such as functional blocks with high aspect ratios, high power densities, etc. We show how to address these issues within the framework of the parameterized, by-construction paradigm. Solutions include further dividing blocks with high aspect ratio into smaller subblocks, applying a proper heatsink boundary condition, modeling package components that can cause significant error but have been neglected so far, and others. Experimental results regarding the improvements are shown in Section 4.

3.1 Aspect Ratio

First, when a functional block is approximated by only one node in the model, the associated lumped thermal resistors and capacitors cannot fully model the distributed nature of heat transfer. In particular, for *blocks with high aspect ratios* where the lateral heat transfer in one direction dominates the other direction, the resulting error can be more significant. This simply requires higher spatial resolution and the solution is to further divide these high-aspect-ratio blocks into subblocks with aspect ratios closer to unity. In Fig. 1a, a functional block with high lateral aspect ratio is represented by only one node. The four lumped lateral thermal resistors connected to that node are also shown. In Fig. 1b, this block is divided into several subblocks with close-to-unity aspect ratios. With this modification, the lateral heat transfer within the block is modeled by a finer network with greater fidelity.

3.2 Heatsink Boundary Condition

Different boundary condition assumptions lead to different temperature estimations. For example, at the heatsink-ambient interface, an isotherm condition is usually assumed in traditional thermal model approaches, whereas a more realistic boundary condition is a convective one, which leads to a nonisotherm temperature distribution at the

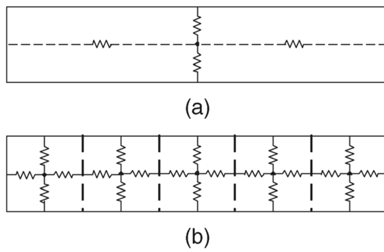

Fig. 1. A block with high aspect ratio. (a) Only one node represents the block for computational efficiency. (b) The block is divided into subblocks with aspect ratio close to unity. The lateral heat transfer paths are modeled with more detail but also with more computational complexity.

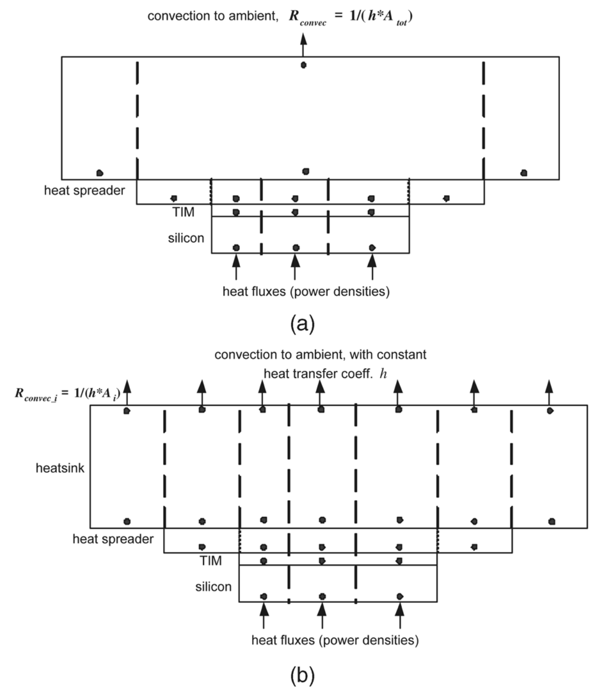

Fig. 2. (a) Simple thermal model with only one convection resistor from heatsink to ambient air, with top surface of heat spreader and heatsink both assumed to be isothermal. (b) An improved model structure. The center part of heatsink is modeled at the same level of detail as the silicon. The isotherm nodes are replaced with multiple nodes connected by different convection resistors.

heatsink surface. Therefore, this more realistic convective boundary condition should be adopted to further improve accuracy.

Fig. 2a shows the model structure in traditional thermal models such as HotSpot 3.1, in which the center part of the upper surface of the heat spreader is approximated to be isothermal and has only one node (each black dot is a node). The heatsink-ambient interface also has only one node. In the real case, these surfaces are not fully isothermal. Accuracy can therefore be improved by removing the isothermal nodes and modeling the heatsink at the same level of details as the silicon die. Furthermore, the convection interface between heatsink and ambient air can be modeled with multiple convection surfaces (hence, multiple nodes) with a constant heat transfer coefficient,

$$R_{\mathrm{convec}i} = \frac{1}{hA_i}, \tag{1}$$

where $R_{\mathrm{convec}i}$ is the convection thermal resistance for the $i\mathrm{th}$ subarea of the heatsink convection surface, h is the constant heat transfer coefficient, and A_i is the subarea. The resulting thermal model structure is shown in Fig. 2b. The heat transfer coefficient h in (1) can be found by solving h from $R_{tot} = 1/(hA_{tot})$ to make sure the equivalent total convection thermal resistance calculated using the total heatsink surface area (A_{tot}) is the same as the lumped isothermal sink-to-air thermal resistance (R_{tot}), which is usually specified in a heatsink's datasheet. Modeling the heatsink with more details introduces more computing overhead to the model. However, as long as the floorplan does not contain too many blocks, the overhead remains tolerable.

Similarly, a recent full-chip thermal model [23] also has added more nodes in the package components. Chaparro et al. [23] approximate the convective boundary condition by mapping and splitting heat spreader and heat sink into blocks according to the die floorplan, with each block in the package bigger than its silicon counterpart as a result of the bigger size of spreader and sink than the silicon die. This is a natural way to add more details in the package components and achieves reasonable accuracy. This package components splitting scheme is slightly different from HotSpot—HotSpot only divides the center parts of the spreader and the sink covered by the previous layer into the same number of blocks as the previous layer and uses four extra nodes for the remaining peripheral areas. The reason behind this is the fact that finite-element simulations (e.g., ANSYS) show that, for copper spreader and heat sink, since the heat spreading within copper is significantly better than silicon, the temperatures outside the center parts of the spreader and sink quickly drop to uniform values. Therefore, we find that it is more accurate to split the package into center nodes and peripheral nodes. For other types of spreaders and heat sinks, such as those with different thermal conductivities, phase-change spreaders, and microchannel spreaders and sinks, different schemes of modeling the package components may need to be developed on a case-to-case basis.

3.3 Including Thermal Interface Material

As mentioned in Section 2, ignoring or oversimplifying package components can introduce significant errors to the results of a thermal model. One package component, the TIM, is of particular interest. TIM is special because it has rather low thermal conductivity due to material limitation and economic reasons. Comparing with the thermal conductivity of silicon (about 100 W/m-K), typical TIM thermal conductivity is less than 10 W/m-K nowadays [24]. In addition, TIM is the layer usually between the silicon die and the heat spreader. Therefore, a low-conductivity TIM prevents efficient heat spreading within the silicon and exacerbates the on-chip local hot spot temperatures. Although TIM with better thermal conductivity is being developed, it will remain as a concern, at least for the near future.

3.4 Additional Improvements to Hotspot

Specific to HotSpot, the following additional sources of accuracy we identified and solutions are proposed here.

First, transient thermal responses can be inaccurate when high power density is applied to a block. In general, absolute transient accuracy is harder to achieve than static accuracy in HotSpot without introducing significant extra model complexity. This is due to the lumped structure of HotSpot and the distributed nature of actual transient thermal response. In HotSpot, scaling factors to thermal capacitors are used to match the thermal time constants between lumped and distributed systems. However, the scaling factors cannot guarantee a perfect match over the entire transient temperature response. The only way to achieve the ultimate transient accuracy is to use a very fine 3D mesh to model the system, which inevitably introduces significant computational overhead and is probably not suitable for architecture-level simulations. Here, we improve the transient accuracy of HotSpot by using *a constant 0.5 scaling factor* for lumped thermal capacitors. As will be shown in Section 4.1.3, using a constant 0.5 capacitance scaling factor in the model achieves fairly good accuracy with respect to ANSYS for most of the time scales. The reason behind the 0.5 scaling factor is that the time constant of a distributed resistor-capacitor circuit is half of that of a one-lumped resistor-capacitor stage [25].

Another source of inaccuracy in HotSpot comes from the fact that certain material properties, such as thermal conductivity and specific heat, are weakly temperature dependent. Approximating them with constant values thus introduces small errors. Although it is fairly straightforward to include this in HotSpot in the form of lookup tables, this is not the focus of this paper and is a topic for future work.

To accurately take the temperature-leakage dependency into consideration during an early design stage, HotSpot 4.0 is further extended to calculate the leakage power according to the updated temperature using the user's own leakage model or HotLeakage [26] and checking for convergence or thermal runaway.

4 Results of Accuracy Improvements

In this section, we present the experimental results of the effect of the abovementioned solutions to the accuracy concerns regarding a by-construction parameterized compact thermal model such as HotSpot. All of the improvements are implemented and verified in HotSpot Version 4.0. For better clarity, we isolate the results of the TIM's impact on temperature estimates to Section 4.2, while showing results of all the rest of the solutions combined in Section 4.1.

4.1 Chip and Boundary Condition Solutions

To evaluate the accuracy improvement, we use ANSYS as our primary reference finite-element model and FreeFEM3d as a secondary source. ANSYS allows users better control on the level of spatial discretization (mesh granularity) and the shape of the finite element (e.g., tetrahedral versus quadrilateral elements) so that greater accuracy can be achieved with smaller elements. In our ANSYS experiments, we use multiple meshing levels (e.g., 1-5 layers for silicon) and types of elements (e.g., tetrahedral versus quadrilateral elements with up to 20 nodes per element) and ensure that the results are consistent across them. The results of FreeFEM3d are either from repeating experiments in [12] or extracted directly from that in [12].

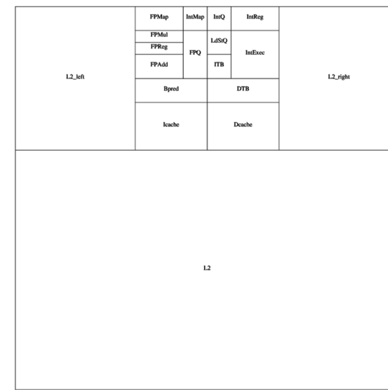

Fig. 3. EV6 floorplan, adapted from that in [3].

4.1.1 Alpha EV6 Steady-State Results

The package geometry used is similar to that in Fig. 2. For this experiment, the silicon die has 16 mm $\times$ 16 mm $\times$ 0.5 mm dimensions. The TIM layer has the same size as the die and is 0.1 mm thick. We also use two different TIM materials; one has a better conductivity of 7.5 W/m-K (good TIM) and the other has a worse thermal conductivity of 1.33 W/m-K (worse TIM).

The heat transfer coefficient at the top surface is 2,777.7 W/m^2-K, which is equivalent to a single lumped convection thermal resistance of 0.1 K/W. The floorplan is one that is similar to that of EV6. We slightly modify the coordinates of the functional blocks for alignment so that it is easier to build the model in ANSYS and FreeFEM3d. We use the same modified EV6 floorplan for HotSpot, ANSYS, and FreeFEM3d in this experiment. The floorplan is shown in Fig. 3.

Figs. 4a and 5a show the temperature estimations from ANSYS, FreeFEM3d (FF3d), HotSpot3.1,[6] and HotSpot4.0 for the good TIM and the worse TIM. To better illustrate the absolute errors of HotSpot block model, in Figs. 4b and 5b, we use ANSYS temperatures as the references and plot the errors of HotSpot4.0, HotSpot3.1, and FreeFEM3d (FF3d) with respect to the ANSYS for both TIM materials.

There are several observations from Figs. 4 and 5:

1. HotSpot 4.0 in general has lower error than HotSpot 3.1. The improved accuracy is achieved by eliminating the isotherm nodes in package and dividing high-aspect-ratio blocks into subblocks with unit aspect ratios.
2. For the case of good TIM, HotSpot is even closer to ANSYS than FreeFEM3d! Furthermore, even HotSpot 3.1 does provide reasonably accurate temperature estimations. Since the package configuration with good TIM represents a realistic package for

6. HotSpot 3.1 is an earlier version that has TIM but does not include the other proposed solutions.

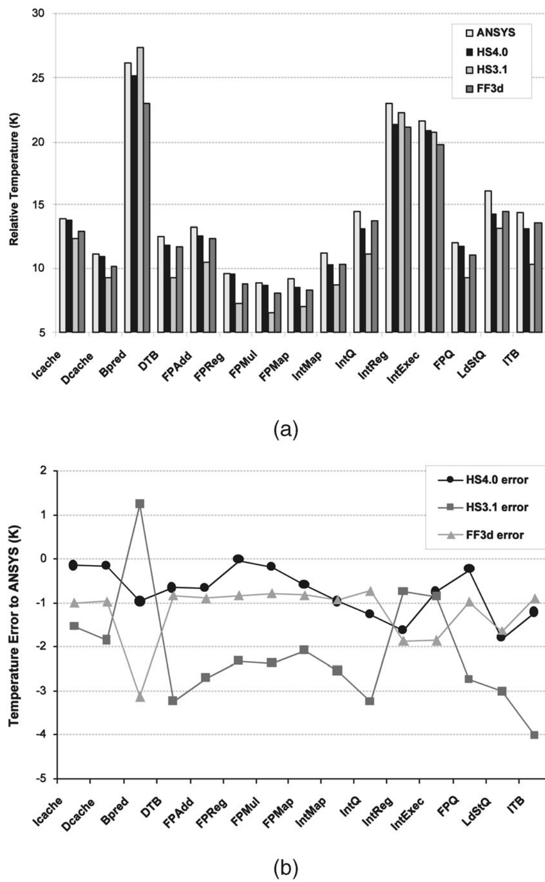

Fig. 4. (a) EV6 block relative temperatures with good thermal interface material. (b) EV6 block relative temperature errors with respect to ANSYS, with good thermal interface materials ($k_{TIM} = 7.5W/(m \cdot K)$).

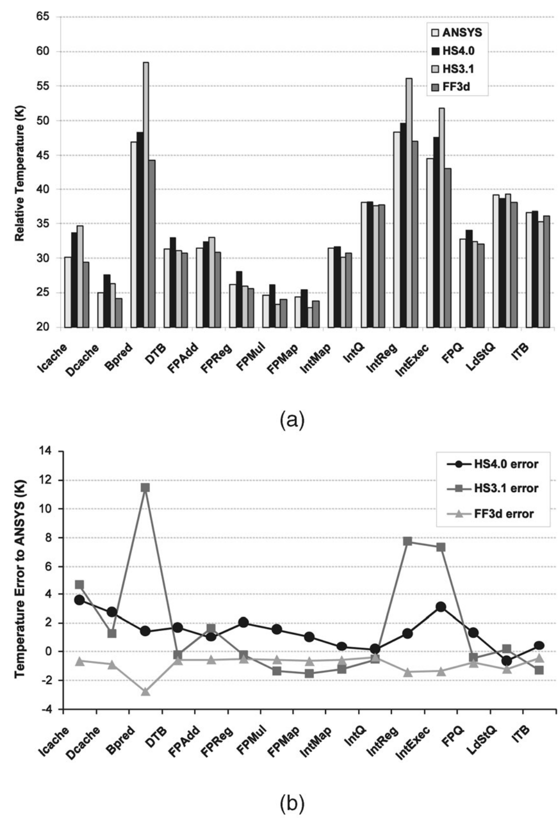

Fig. 5. (a) EV6 block relative temperatures with worse thermal interface materials. (b) EV6 block relative temperature errors with respect to ANSYS with worse thermal interface materials ($k_{TIM} = 1.33W/(m \cdot K)$).

modern high-performance microprocessors, *we can see that the original HotSpot 3.1 block model is already quite accurate under typical thermal simulation scenarios.*

3. For the case of worse TIM, HotSpot predicts hotter temperatures than both ANSYS and FreeFEM3d in most cases, but the percentage errors for hot units, e.g., BPred and IntReg, are 3.05 percent and 2.56 percent, respectively. Overall worst-case percentage error with worse TIM is 11.96 percent for I-Cache, which is a relatively cool unit.
4. There are noticeable differences between ANSYS and FreeFEM3d (FF3d) as well, both being detailed finite-element models.

4.1.2 Square Source Steady-State Results

A better experiment that helps to evaluate and explain the steady-state errors is to test a range of heat source sizes with the same power density. In this experiment, the silicon chip has a size of 21 mm $\times$ 21 mm $\times$ 0.5 mm and the dimensions of other package components are the same as Section 4.1.1. The center heat source size varies from 1 mm to 19 mm. The applied power density to the center block is set to a constant value of 1.66 $\mathrm{W/mm^2}$. Fig. 6a shows a floorplan with a 1 mm square heat source, together with its high aspect ratio neighbor blocks. Fig. 6b shows the same floorplan in which the high aspect ratio blocks are divided into square subblocks.

Figs. 7 and 8 show the comparisons among the HotSpot 3.1, HotSpot 4.0, ANSYS, and FreeFEM3d for different heat source sizes. We also plot the HotSpot 3.1 results with unity-aspect-ratio (sub)blocks (HS3.1-AR) to isolate the effect of each individual aforementioned modifications (i.e., unity aspect ratio and nonisothermal boundary condition). As can be seen, the HotSpot 4.0 block model is much more accurate than the earlier HotSpot 3.1 block model.

For a smaller heat source size (1 mm to 5 mm), the significant error of HotSpot 3.1 is caused by the extreme aspect ratio (10:1) of the four long and narrow blocks that are adjacent to the center small heat source block. In HotSpot 4.0, these long, narrow blocks are automatically

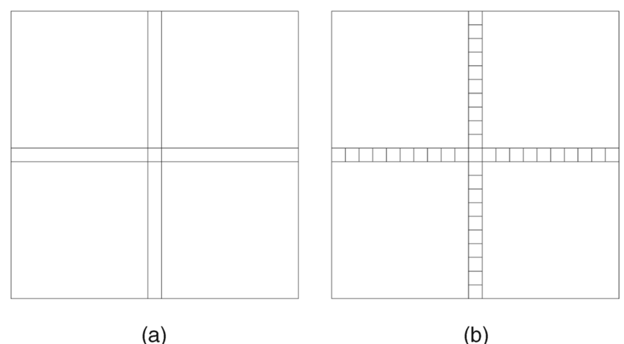

Fig. 6. (a) Floorplan with a 1 mm center square heat source dissipating 1.66 W. Notice the neighboring high aspect ratio blocks. (b) The neighboring high aspect ratio blocks are divided into square subblocks.

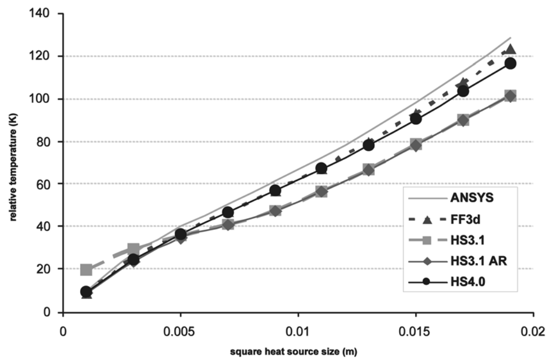

Fig. 7. Center temperature for different heat source sizes, with good thermal interface material $(k_{TIM} = 7.5W/(m \cdot K))$, power density is $1.66\ W/mm^2$.

divided into 10 subblocks with aspect ratios of 1:1; thus the accuracy is greatly improved (see the left part of the "HS3.1 AR" curves for small heat source sizes).

For a larger heat source size (e.g., 19 mm, which has 600 W of power), the significant error of HotSpot 3.1 is caused by the fact that the upper surfaces of the heat spreader and the heatsink are no longer close to being isothermal, so approximating them with single nodes yields significant errors. In HotSpot 4.0, the isothermal nodes are removed. Instead, we model the heatsink at the same level of detail as the silicon die and use a constant heat transfer coefficient $(\mathrm{h} = 2777.7\ \mathrm{W/m^2\text{-}K})$ for each subarea of the heatsink-ambient interface. This significantly improves the accuracy for large-size heat sources (see the significant improvement for larger heat source sizes from "HS3.1 AR" to "HS4.0").

Here again, by eliminating the isothermal nodes in package and dividing high-aspect-ratio blocks into subblocks with unit aspect ratios, the HotSpot block model greatly improves its accuracy.

4.1.3 Pulse Response for Bpred Unit in EV6 Floorplan

To evaluate the transient accuracy improvement of HotSpot 4.0, we performed an experiment with power pulses of different time scales.

In Fig. 9, power pulses of 100 μs, 1 ms, and 10 ms are sequentially applied to the Branch Predictor (Bpred) block in the EV6 floorplan with a uniform power density of $2\ \mathrm{W/mm^2}$ to verify HotSpot 4.0's accuracy at different time scales. Notice that the time axis is in log scale. We compare HotSpot 4.0 and HotSpot 3.1 results with ANSYS. As can be seen, HotSpot 4.0 significantly improves transient accuracy for all time scales under this high-aspect-ratio and high-power-density extreme case.

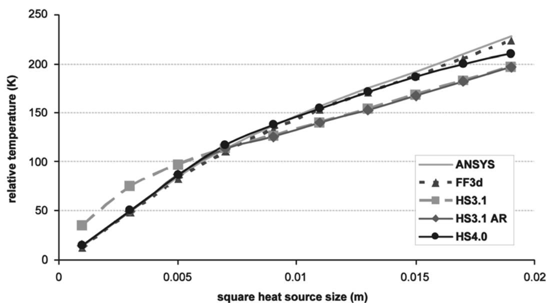

Fig. 8. Center temperature for different heat source sizes, with worse thermal interface material $(k_{TIM} = 1.33W/(m \cdot K))$, power density is $1.66\ W/mm^2$.

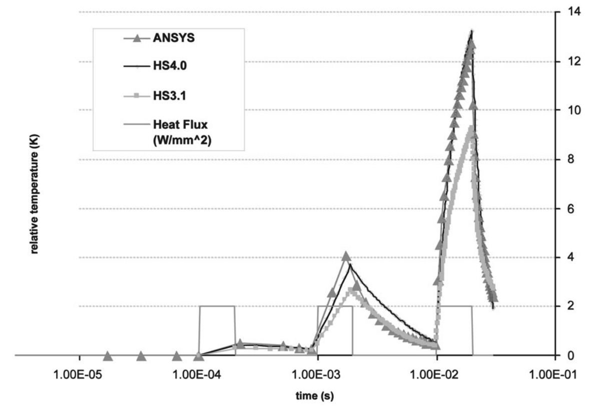

Fig. 9. Transient temperature response for different power pulse-widths applied to the branch predictor of EV6. Power density is $2\mathrm{W/mm^2}$ $(k_{TIM} = 7.5W/(m \cdot K))$.

We can see that, in addition to eliminating the isothermal nodes in the package and dividing high-aspect-ratio blocks into subblocks with unit aspect ratios, the HotSpot block model's transient accuracy is also improved by using a constant scaling factor of 0.5 to approximate the thermal time constant of the distributive nature of transient temperature evolvement. The scaling factor comes from the analogous electrical distributed RC circuit whose time constant is half of the one-ladder RC circuit [25].

Based on the above steady-state and transient experiments and comparisons among the HotSpot block model, ANSYS, and FreeFEM3d, we can see that the improved HotSpot 4.0 model is accurate as a by-construction compact thermal model for architecture-level and other early-stage design levels. The small inaccuracies come from the fact that the compact thermal model trades off accuracy to achieve greater model compactness.

4.2 TIM's Impact on Chip Temperature

Earlier in the paper, we mentioned that package components can greatly affect the temperature distribution across the silicon die. In this section, we show some example thermal analysis regarding one particular packaging component—TIM that bonds the silicon die to the heat spreader.

With the flexibility of the improved parameterized compact thermal model, we can easily investigate the thermal impacts of different TIM properties such as its thickness, void size, and attaching surface roughness in the early design stages and provide important insights for computer architects, circuit designers, and package designers.

We first show how the *thickness* of TIM affects silicon die temperature distribution. Fig. 10 plots the across-die temperature difference from the compact thermal model with different TIM thickness.

As can be observed in Fig. 10, thicker TIM results in poor heat spreading that leads to large temperature differences across the die. We can see that thick TIM can lead to very large die temperature difference across the die $(> 50^\circ\mathrm{C})$.

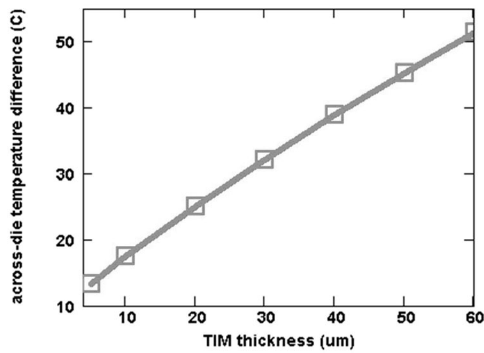

Fig. 10. The impact of TIM thickness on silicon die temperature difference [20].

Even with nominal TIM thickness, which is 20 μm for this design, the temperature difference across the die is still $24^{\circ}\mathrm{C}$. This means that the bottom surface of the die cannot be modeled as an isothermal surface. If the TIM is thick enough, the resulting extremely large temperature differences across the die may be disastrous to circuit performance and die/package reliability. Using a better heatsink will only lower the average silicon temperature but will not help to reduce the temperature difference. This analysis suggests that using the thinnest possible TIM is one of the key issues for package designers to consider. On the other hand, with the known TIM thickness that can be best assembled in a package with state-of-the-art packaging technology, it is the task of circuit designers and computer architects to design proper circuits and architectures to maintain the temperature difference across die within a manageable level.

As another example, Fig. 11 shows the relationship between the size of TIM *void* and the hot spot temperature. During the packaging process, it is almost unavoidable to leave voids or air bubbles in the TIM. In the compact thermal model, the void in TIM can be easily modeled by introducing higher vertical TIM thermal resistance to the grid cell where the void resides. Different sizes of the TIM void can be modeled by different sizes of the grid cell. For the simulations in Fig. 11, we put the TIM void right under the hottest grid cell, thus modeling the highest possible die temperature in the presence of a void with different sizes.

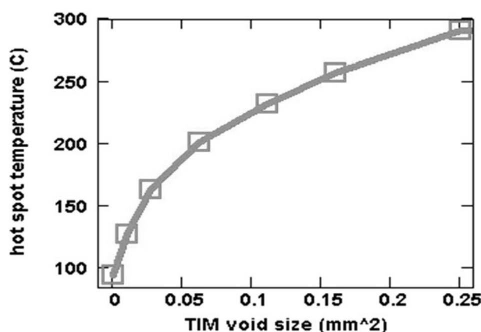

Fig. 11. The impact of the size of the void defect in TIM on the silicon die hottest temperature. Temperatures are normalized to the ideal case where there is no void defect in the TIM layer. TIM void sizes are with the unit of mm^2 [20].

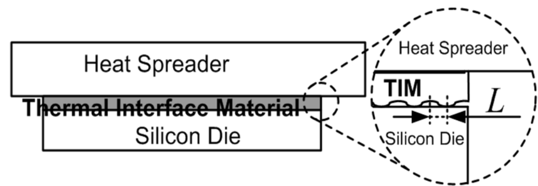

Fig. 12. Close-up view of the TIM/die attaching surface. Surface nonuniformity is indicated by L [20].

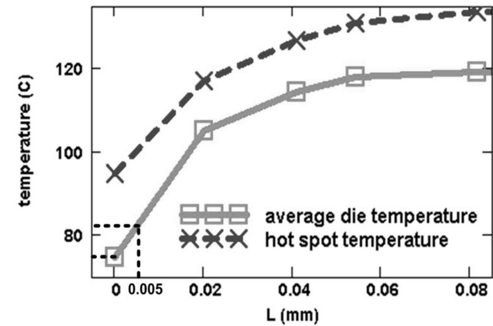

Fig. 13. Hottest die temperature and average die temperature versus the nonuniformity of the TIM attaching surface. The larger L is, the rougher the attaching surface is. L is defined in Fig. 12 [20].

As can be seen in Fig. 11, if the hot spot temperature of the design is $95^{\circ}\mathrm{C}$, a void or air bubble in the TIM with a size of 0.25 mm^2 can make the hot spot temperature drastically higher ($290^{\circ}\mathrm{C}$), which inevitably leads to a thermal runaway of the chip. Therefore, it is desirable to improve the packaging techniques to make the size of the TIM void as small as possible. Package designers usually have the expertise to know typical TIM void sizes for different packaging processes. They can include this information in the thermal model. By doing this, the thermal model is now able to provide the possible worst-case temperature regarding TIM void defects. The consequent architecture and circuit design decisions can thus avoid potential thermal hazards caused by the TIM void defects.

Another important TIM property that affects the die temperature is the *surface roughness*, i.e., nonuniform TIM. In a real-life chip packaging process, the bottom surface of the die and the TIM's attaching surface cannot be perfectly smooth. As shown in Fig. 12, TIM is only attached to the die at the bumps of the TIM surface. This causes ineffective heat conduction and, hence, higher die temperature compared to the case where TIM and the die attach to each other perfectly. In order to investigate the impact of TIM nonuniformity to the die temperature, we change the thermal model of the TIM layer according to that in Fig. 12, where we simply model the nonuniformity of the TIM surface as tiny bumps with spacing $2L$. The size of each grid cell is set to L. Therefore, heat can only be conducted through the grid cells representing the touching bumps. Grid cells representing the valleys are essentially tiny voids that do not touch the die and have extremely low thermal conductivity. The value of L thus can be used as an indicator of the nonuniformity of the TIM surface—the surface is rougher when L is larger and vice versa. Fig. 13 is the model results showing the relationship between L (nonuniformity) and die temperatures, where $L = 0$ means

the TIM surface is perfectly uniform. As observed, even a tiny nonuniform TIM surface (e.g., $L = 5\mu\text{m}$) can significantly raise both the hottest and the average die temperature (by about 10 degrees). Package designers again usually have the specifications of the surface nonuniformities for different packaging processes. Without considering such package processing specifications, it is inevitable that a thermal model underestimates the die temperature and leads to designs that are not thermally optimized and designs with a higher probability of premature failures.

4.3 Trade-Offs of Using Hotspot 4.0

While HotSpot 4.0 has better accuracy under more extreme conditions, it also introduces more computational overhead than HotSpot 3.1 due to the increased number of nodes in both silicon and package components. The overhead is usually negligible when the number of blocks is relatively small. HotSpot 4.0 is extremely useful when there are very high-power density blocks with high aspect ratio (e.g., BPred and IntReg with low-conductivity TIM in Fig. 5) or the total power of the chip is extremely high (e.g., the right-end points in Figs. 7 and 8 corresponding to the 19 mm $\times$ 19 mm 600 W heat source). However, if the floorplan has tens and hundreds of blocks, HotSpot 3.1 may be a better trade-off between computation complexity and accuracy.

5 A Thermally Optimized Design Flow

As temperature management is more challenging as the result of the nonideal CMOS scaling, considering thermal issues early in the design process becomes imperative. Even though the recent trend toward many-core chips can, to some extent, alleviate localized heating due to a more uniform power distribution compared to traditional single- and dual-core designs, accurately modeling local temperature variation using HotSpot is still important due to the fact that the high-activity cores are usually surrounded by cool local caches; hence, local temperature distribution may still be far from uniform. In addition, wafer thickness also scales down, resulting in less efficient within-silicon heat spreading and possibly more prominent localized heating, not to mention multicore chips with heterogeneous cores that can vary significantly in terms of power consumption and temperature among different cores.

The HotSpot model is unique because it efficiently models both chip and package temperatures with satisfactory accuracy for any type of processor designs at any level of detail. This is the key to more effective collaborations among computer architects, circuit designers, and package designers. With the help of such an accurate full-chip and package compact thermal model, an early-stage thermally optimized design flow is proposed in this section to accurately predict potential thermal hazards and to achieve economical designs with faster design convergence.

5.1 The Design Flow

Fig. 14 illustrates the proposed prelayout design flow. As shown in Fig. 14, circuit designers first design basic blocks such as macros and each macro has a simulated dynamic power for a certain workload. It also has an estimated layout bounding box. Computer architects then assemble a

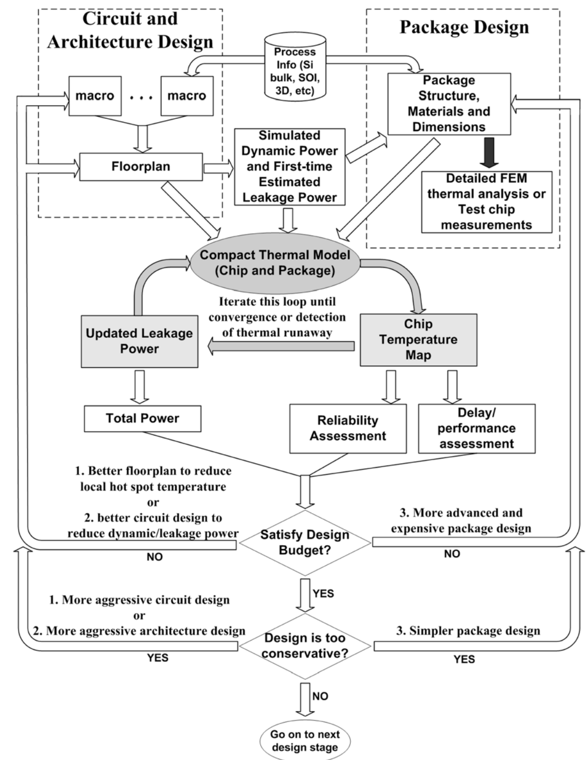

Fig. 14. A design flow showing the compact thermal model acts as a convenient medium for productive collaborations for designers at the circuit, architecture, and package levels [20].

preliminary microarchitecture-level floorplan. At this point, initial total power, including the rough estimation of leakage power, can be used for a package designer to propose a preliminary package design. All of the information about power, floorplan, and package is used to construct a compact thermal model that can perform thermally self-consistent leakage power calculations, as shown in the highlighted inner loop in Fig. 14. The resulting temperature map can then be utilized to perform temperature-critical reliability analysis (e.g., interconnect electromigration, gate-oxide breakdown, and package deformation) and temperature-related performance analysis (e.g., interconnect and device delay and power grid *IR* drop).

The results of all this analysis, together with the total power, are then compared to the design goals. If the goals are not satisfied, different trade-offs can be made—circuit designers may need to invent novel circuits with lower power dissipation, computer architects may have to think more about new architectures and different floorplans to better manage power and temperature, or package designers may need to propose more advanced, usually more expensive, packages. On the other hand, if the design goals are fully satisfied, we still need to check whether the design is too conservative and the design margin is too large for the application. We can then improve the conservative design by either introducing more aggressive circuit and/or architecture solutions to enhance performance or using simpler and cheaper packages to reduce the cost of the final product. These decisions and trade-offs can then be evaluated using the thermal analysis, again following the

same flow until an optimal design point is reached. Then, one can proceed to the physical design stage.

With the above design flow, the potential thermal hazards can be discovered and dealt with early and efficiently; thus, the design is optimized from a thermal point of view.

5.2 An SoC Design Example

To illustrate the importance of adopting such a thermally optimized design flow early in the design process, we show the thermal analysis together with the temperature-leakage loop for an SoC design. We use InCyte®, a novel commercialized early design estimation tool,[7] to reconstruct an SoC design based on the published 180 nm design data in [27]. This SoC design does not have an integrated heatsink due to its low power consumption. It uses natural convection from a metal covering lid, which acts as the heat spreader, as the cooling method.

We use HotSpot 4.0 for the thermally self-consistent leakage analysis of this SoC design. Because a heatsink is not present in the package, we apply the natural convection boundary condition at the surface of the thin lid that is attached to the silicon substrate. Notice that, without the improvement of directly modeling the convection boundary condition in HotSpot 4.0, it is impossible to accurately simulate such a scenario because, under natural convection, the package surface is obviously not isothermal.

We pick logic and memory modules similar to those in [27] from InCyte's incorporated IP libraries and come up with an early SoC design whose total power is almost identical to data reported in [27]. InCyte also outputs a preliminary floorplan for the design. Although the area of each block is also similar to the original design, the relative locations of different blocks are noticeably different. This is acceptable since InCyte is a tool for early stage design. In addition, notice that InCyte estimates leakage power of each block at a constant temperature. Following that, we use HotSpot to estimate chip temperature distribution and pick a proper package from InCyte's package library for this design based on data estimated from InCyte. If we assume that the on-chip highest temperature constraint is $85^{\circ}\mathrm{C}$ and the ambient temperature is $25^{\circ}\mathrm{C}$, we find that the thermal package needs to have a lumped thermal resistance of 18.2 K/W, which is common for standard SBGA packages, in order to keep the hot spot temperature below $85^{\circ}\mathrm{C}$. The estimated temperature map of this 180 nm design is shown in Fig. 15.

Because InCyte does not yet include the temperature dependency of leakage, whereas subthreshold leakage is exponentially dependent on temperature, we double-check to see whether the thermally self-consistent leakage power causes thermal problems to this 180 nm SoC design. Using HotSpot and the simplistic leakage model in [20] to iterate the temperature-leakage loop, as shown in Fig. 14, after convergence, we find that the final total leakage is only a negligible 546 μW for this design with the picked package. Therefore, the above temperature estimation is quite accurate without considering the temperature-leakage loop.

7. http://www.chipestimate.com/.

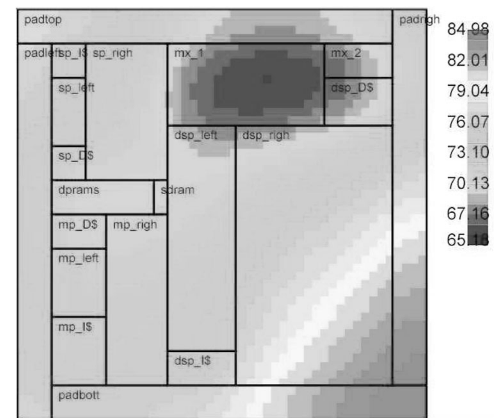

Fig. 15. Estimated temperature map of an SoC design at 180 nm technology, based on the data in [27] and InCyte. Temperatures are in $^{\circ}\mathrm{C}$.

However, if we redesign this SoC design in a 90 nm technology, there are two design possibilities: 1) We scale both the area and active power of each individual blocks and thus maintain the same function and complexity. This means that the total power of the entire design is also scaled accordingly; thus, the power density remains the same due to area scaling. Therefore, we can use a cheaper thermal package for less overall power consumption and keep the chip below the $85^{\circ}\mathrm{C}$ thermal constraint. 2) Since ITRS [1] projects that the die size and power remain the same, if not increasing, across different technology nodes, we can alternatively assume the total active power and chip area remain the same as those in the 180 nm design. Assuming a floorplan similar to that in 180 nm technology, this is equivalent to adding more parallelism (such as more processing cores and higher memory bandwidth) to the die and designing the chip for higher throughput by burning more power. In this case, with the same 18.2 K/W thermal package, after iterating the leakage-temperature loop, the hottest on-chip temperature exceeds the thermal threshold and eventually causes thermal runaway. The reason is twofold: 1) At 90 nm, a greater fraction of total power consumption is caused by leakage [1], and 2) the subthreshold leakage power's dependency on temperature is stronger at 90 nm than at 180 nm (see the leakage model coefficients shown in [28], [29]). The results are listed in Table 1.

The above SoC design example shows that it is crucial to incorporate thermal estimations (such as leakage-temperature dependence) early in the design process in order to locate potential thermal hazards that are too costly to fix in the later design stages. At this early design stage, possible solutions to the SoC design at 90 nm can be 1) circuit designers can choose IPs that have high-Vt transistors and use reverse body-bias or sleep transistors for noncritical paths to reduce leakage, 2) architects can consider using dynamic voltage and frequency scaling (DVFS), migrating computation [3], [28], more parallelism, and temperature-aware floorplanning techniques [30], etc. to reduce hot spot temperatures, and, alternatively, 3) package designers need to consider the possibility of adding a heatsink or a fan. Trade-offs among portability, cost, performance, and temperature have to be made in this case by following the

TABLE 1
As Technology Scales, Temperature Dependence of Subthreshold Leakage Power Becomes More Problematic

	active power	avg. temp rise	hottest temp rise	actual leakage	leakage error@const temp
180nm, orig. design	1.7W	30.77C	59.98C	546μW	116%
90nm, scaled pow&area	316mW	6.19C	25.42C	24mW	66%
90nm, same pow&area	1.7W	>35.08C	runaway	>277mW	>680%

Without early-stage thermally optimized design flow (Fig. 14), thermal runaway can happen even for low-power SoC designs.

design flow in Fig. 14. Unlike HotSpot 4.0, other existing thermal modeling approaches are either not accurate enough (e.g., neglecting package component details or using the wrong boundary conditions) or too time-consuming (e.g., detailed FEM) and, hence, are not suitable for such a design trade-off analysis early in the design process.

6 CONCLUSIONS

In this paper, we first present improvements to an efficient by-construction compact thermal model, like HotSpot, to make it accurate even under scenarios such as high aspect ratio blocks and high power density and to better model realistic convective boundary conditions for thermal package components. The accuracy improvements of both steady-state and transient temperatures are confirmed by comparing with finite-element models in ANSYS and FreeFEM3d. The importance of accurate considerations and modeling of package components also determines the accuracy of the die-level temperature estimations. Several examples are presented to illustrate the impact of TIM on die temperature distribution. With the improvements of the model structure and the proper inclusion of package components, thermal models such as HotSpot 4.0 can further act as a convenient communication medium for more efficient cooperations among computer architect, circuit designers, and package designers, thus achieving a thermally optimized design early in the design stages. The importance of adopting such an early stage thermally optimized design flow is illustrated by the detection of potential thermal runaway in the early stage analysis for a 90 nm SoC design.

ACKNOWLEDGMENTS

The authors would like to thank Pierre Michaud and Damien Fetis from IRISA/INRIA, France, for the interesting discussions and generous help on FreeFEM3d. They also thank Jeff Ng, Nozar Nozarian, and Miles McGowan from ChipEstimate Inc. for their help with InCyte. This work is funded by US National Science Foundation CRI Grant CNS-0551630 and has partial support from a MARCO IFC Grant.

REFERENCES

[1] The Int'l Technology Roadmap for Semiconductors (ITRS), 2003.
[2] W. Huang, R. Stan, and K. Skadron, "Parameterized Physical Compact Thermal Modeling," *IEEE Trans. Components and Packaging Technologies,* vol. 28, no. 4, pp. 615-622, Dec. 2005.
[3] K. Skadron, K. Sankaranarayanan, S. Velusamy, D. Tarjan, M.R. Stan, and W. Huang, "Temperature-Aware Microarchitecture: Modeling and Implementation," *ACM Trans. Architecture and Code Optimization,* vol. 1, no. 1, pp. 94-125, Mar. 2004.
[4] W. Huang, K. Sankaranarayanan, R.J. Ribando, M.R. Stan, and K. Skadron, "An Improved HotSpot Block-Based Thermal Model with Granularity Considerations," *Proc. Workshop Duplicating, Deconstructing, and Debunking in conjunction with the Int'l Symp. Computer Architecture,* June 2007.
[5] D. Brooks, V. Tiwari, and M. Martonosi, "Wattch: A Framework for Architectural-Level Power Analysis and Optimizations," *Proc. Int'l Symp. Computer Architecture,* pp. 83-94, June 2000.
[6] W. Huang, M.R. Stan, K. Skadron, K. Sankaranarayanan, and S. Ghosh, "Compact Thermal Modeling for Temperature-Aware Design," *Proc. 41st Design Automation Conf.,* pp. 878-883, June 2004.
[7] P. Chaparro, J. Gonzalez, G. Magklis, Q. Cai, and A. Gonzalez, "Understanding the Thermal Implications of Multicore Architectures," *IEEE Trans. Parallel and Distributed Systems,* vol. 18, no. 8, pp. 1055-1065, Aug. 2007.
[8] Y. Yang, Z.P. Gu, C. Zhu, R.P. Dick, and L. Shang, "ISAC: Integrated Space and Time Adaptive Chip-Package Thermal Analysis," *IEEE Trans. Computer-Aided Design,* vol. 26, no. 1, pp. 86-99, Jan. 2007.
[9] W. Wu, L. Jin, J. Yang, P. Liu, and S.X.-D. Tan, "Efficient Power Modeling and Software Thermal Sensing for Runtime Temperature Monitoring," *ACM Trans. Design Automation of Electronic Systems,* vol. 12, no. 3, Aug. 2007.
[10] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M.R. Stan, "HotSpot: A Compact Thermal Modeling Methodology for Early-Stage VLSI Design," *IEEE Trans. Very Large Scale Integration (VLSI) Systems,* vol. 14, no. 5, pp. 501-513, May 2006.
[11] N. Rinaldi, "On the Modeling of the Transient Thermal Behavior of Semiconductor Devices," *IEEE Trans. Electronic Devices,* vol. 48, no. 12, pp. 2796-2802, Dec. 2001.
[12] D. Fetis and P. Michaud, "An Evaluation of HotSpot-3.0 Block-Based Temperature Model," *Proc. Workshop Duplicating, Deconstructing, and Debunking in conjunction with Int'l Symp. Computer Architecture,* June 2006.
[13] M.R. Stan, K. Skadron, M. Barcella, W. Huang, K. Sankaranarayanan, and S. Velusamy, "HotSpot: A Dynamic Compact Thermal Model at the Processor-Architecture Level," *Microelectronics J.,* vol. 34, pp. 1153-1165, 2003.
[14] T.-Y. Wang and C.C.-P. Chen, "3-D Thermal-ADI: A Linear-Time Chip Level Transient Thermal Simulator," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems,* vol. 21, no. 12, pp. 1434-1445, Dec. 2002.
[15] H. Su, F. Liu, A. Devgan, E. Acar, and S. Nassif, "Full Chip Estimation Considering Power Supply and Temperature Variations," *Proc. Int'l Symp. Low Power Elec. Design,* pp. 78-83, Aug. 2003.
[16] P. Li, L. Pileggi, M. Asheghi, and R. Chandra, "Efficient Full-Chip Thermal Modeling and Analysis," *Proc. Int'l Conf. Computer-Aided Design,* 2004.
[17] C.J.M. Lasance, "Two Benchmarks to Facilitate the Study of Compact Thermal Modeling Phenomena," *IEEE Trans. Components and Packaging Technologies,* vol. 24, no. 4, pp. 559-565, Dec. 2001.
[18] M.-N. Sabry, "Compact Thermal Models for Electronic Systems," *IEEE Trans. Components and Packaging Technologies,* vol. 26, no. 1, pp. 179-185, Mar. 2003.
[19] E.G.T. Bosch, "Thermal Compact Models: An Alternative Approach," *IEEE Trans. Components and Packaging Technologies,* vol. 26, no. 1, pp. 173-178, Mar. 2003.
[20] W. Huang, E. Humenay, K. Skadron, and M. Stan, "The Need for a Full-Chip and Package Thermal Model for Thermally Optimized IC Designs," *Proc. Int'l Symp. Low Power Electronic Design,* pp. 245-250, Aug. 2005.

[21] K. Banerjee, S.C. Lin, A. Keshavarzi, and V. De, “A Self-Consistent Junction Temperature Estimation Methodology for Nanometer Scale ICs with Implications for Performance and Thermal Management,” *Proc. Int'l Electron Devices Meeting,* pp. 3671-3674, 2003.
[22] L. He, W. Liao, and M.R. Stan, “System Level Leakage Reduction Considering the Interdependence of Temperature and Leakage,” *Proc. 41st Design Automation Conf.,* pp. 12-17, June 2004.
[23] P. Chaparro, J. Gonzalez, and A. Gonzalez, “Thermal-Effective Clustered Microarchitecture,” *Proc. First Workshop Temperature-Aware Computer Systems,* June 2004.
[24] E.C. Samson, S.V. Machiroutu, J.-Y. Chang, I. Santos, J. Hermerding, A. Dani, R. Prasher, and D.W. Song, “Interface Material Selection and a Thermal Management Technique in Second-Generation Platforms Built on Intel Centrino Mobile Technology,” *Intel Technology J.,* vol. 9, no. 1, Feb. 2005.
[25] H.B. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI.* Addison-Wesley, 1990.
[26] Y. Zhang, D. Parikh, K. Sankaranarayanan, K. Skadron, and M. Stan, “HotLeakage: A Temperature-Aware Model of Subthreshold and Gate Leakage for Architects,” Technical Report CS-2003-05, Computer Science Dept., Univ. of Virginia, 2003.
[27] H. Stolberg, S. Moch, L. Friebe, A. Dehnhardt, M. Kulaczewski, M. Berekovic, and P. Pirsch, “An SoC with Two Multimedia DSPs and a RISC Core for Video Compression Applications,” *Digest of Papers, IEEE Int'l Solid-State Circuits Conf.,* Feb. 2004.
[28] S. Heo, K. Barr, and K. Asanovic, “Reducing Power Density through Activity Migration,” *Proc. Int'l Symp. Low Power Electronics and Design,* pp. 217-222, Aug. 2003.
[29] J. Srinivasan, S.V. Adve, P. Bose, and J.A. Rivers, “The Impact of Technology Scaling on Lifetime Reliability,” *Proc. Int'l Conf. Dependable Systems and Networks,* June 2004.
[30] K. Sankaranarayanan, S. Velusamy, M.R. Stan, and K. Skadron, “A Case for Thermal-Aware Floorplanning at the Microarchitectural Level,” *The J. Instruction-Level Parallelism,* vol. 7, Oct. 2005.

Wei Huang received the BE degree in electrical engineering from the University of Science and Technology of China and the PhD degree in electrical engineering from the University of Virginia. He is currently with the Computer Science Department at the University of Virginia as a postdoctoral researcher. His research interests include VLSI circuits and computer architecture with considerations on thermal, power, variability, and reliability issues. He is a member of the IEEE.

Karthik Sankaranarayanan received the BE degree in computer science and engineering from Anna University, Chennai, India, in 2000 and the MS degree from the University of Virginia, Charlottesville, in 2003. He is currently working toward the PhD degree at the University of Virginia. He is a member of the LAVA Laboratory at the University of Virginia. His research interests include computer architecture in general and thermal and power-aware microarchitectures in particular.

Kevin Skadron received the BS and BA degrees in electrical and computer engineering and economics from Rice University and the PhD degree in computer science from Princeton University. He is the cofounder and the associate editor-in-chief of the *IEEE Computer Architecture Letters*. He is an associate professor in the Department of Computer Science at the University of Virginia. His research interests focus on physical design challenges and programming models for multicore/manycore architectures, including graphics architectures. He is a member of the Eta Kappa Nu and Omicron Delta Epsilon and he is a senior member of the ACM, the IEEE, the IEEE Computer Society, and the IEEE Circuits and Systems Society.

Robert J. Ribando received all of his degrees from Cornell University. He is an associate professor in the Department of Mechanical and Aerospace Engineering at the University of Virginia. Prior to coming to Virginia, he was on the research staff of the Advanced Reactors Safety Section at Oak Ridge National Laboratory. His research and teaching interests include computational fluid dynamics and heat transfer and the graphical display of quantitative information. The applications have included nuclear reactor heat transfer, strongly rotating flows, biomedical flows, turbomachinery flows, etc. From 1992 to 1995, he held the Lucien Carr III Chair in Engineering Education, a temporary position intended to recognize and encourage the use of technology in instruction. He is currently writing a textbook and CD applying modern computational methods and visualization to the study of heat transfer.

Mircea R. Stan received the diploma in electronics and communications from “Politehnica” University in Bucharest, Romania, in 1984 and the MS and PhD degrees in electrical and computer engineering from the University of Massachusetts, Amherst, in 1994 and 1996. Since 1996, he has been with the Department of Electrical and Computer Engineering at the University of Virginia, where he is currently an associate professor. He is teaching and doing research on high-performance low-power VLSI, temperature-aware circuits and architecture, embedded systems, and nanoelectronics. He has more than eight years of industrial experience, has been a visiting faculty member at the University of California, Berkeley, in 2004-2005, IBM in 2000, and Intel in 2002 and 1999. He received the NSF CAREER award in 1997 and was a coauthor of the papers which received the best paper awards at GLSVLSI 2006, ISCA 2003, and SHAMAN 2002. He is the chair of the VLSI Systems and Applications Technical Committee (VSA-TC) of IEEE CAS, general chair for ISLPED 2006 and for GLSVLSI 2004, technical program chair for NanoNets 2007 and ISLPED 2005, and a member of the technical committees of numerous conferences. Since 2004, he has been an associate editor for the *IEEE Transactions on Circuits and Systems Systems I* and, from 2001 to 2003, an associate editor for the *IEEE Transactions on VLSI Systems*. He ws also a guest editor for the *Computer* special issue on power-aware computing in December 2003 and a distinguished lecturer for the IEEE Solid-State Circuits Society (SSCS) in 2007-2008 and the IEEE Circuits and Systems (CAS) Society in 2004-2005. He is a senior member of the IEEE and a member of the ACM, the IET (former IEE), and also of Eta Kappa Nu, Phi Kappa Phi, and Sigma Xi.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.