Full-Chip Power Density and Thermal Map Characterization for Commercial Microprocessors Under Heat Sink Cooling

Jinwei Zhang, *Student Member, IEEE*, Sheriff Sadiqbatcha, *Graduate Student Member, IEEE*, Michael O'Dea, Hussam Amrouch, *Member, IEEE*, and Sheldon X.-D. Tan, *Senior Member, IEEE*

***Abstract*—In this article, we address the problem of accurate full-chip power and thermal map estimation for commercial off-the-shelf multicore processors. Processors operating with heat sink cooling remains a challenging problem due to the difficulty in direct measurement. We first propose an accurate full-chip steady-state power density map estimation method for commercial multicore microprocessors. The new method consists of a few steps. First, 2-D spatial Laplace operation is performed on the measured thermal maps (images) without heat sink to obtain the so-called *raw power maps*. Then, a novel scheme is developed to generate the true power density maps from the raw power density maps. The new approach is based on thermal measurements of the processor with back-side cooling using an advanced infrared (IR) thermal imaging system. FEM thermal model constructed in COMSOL Multiphysics is used to validate the estimated power density maps and thermal conductivity. Later, this work creates a high-fidelity FEM thermal model with heat sink and reconstructs the full-chip thermal maps while the heat sink is on. Ensuring that power maps are similar under back cooling and heat sink cooling settings, the reconstructed thermal maps are verified by the matching between the on-chip thermal sensor readings and the corresponding elements of thermal maps. Experiments on an Intel i7-8650U 4-core processor with back cooling shows 96% similarity (2-D correlation) between the measured thermal maps and the thermal maps reconstructed from the estimated power maps, with 1.3 °C average absolute error. Under heat sink cooling, the average absolute error is 2.2 °C over a 56 °C temperature range and about 3.9% error between the computed and the real thermal maps at the sensor locations. Furthermore, the proposed power map estimation method achieves higher resolution and at least 100× speedup than a recently proposed state-of-art Blind Power Identification method.**

***Index Terms*—Cooling, finite element-based method (FEM) simulation, post-silicon, processor power estimation, sensor location, temperature estimation.**

I. Introduction

POWER, thermal and related reliability issues are among the major limiting factors for today's high performance multicore processors. This is especially true after the breakdown of the so-called Dennard scaling, since power density starts to increase as IC technology advances [1], [2]. To enhance reliability, researchers have proposed many power/thermal regulation or dynamic management methods, including clock gating, power gating, dynamic voltage and frequency scaling (DVFS), and task migration [3]–[6].

To obtain accurate on-chip temperature, we need to look at two important aspects of this problem: 1) the accurate estimation of the input power and 2) the accurate calculation of on-chip temperature from the thermal model and the input power.

Traditional power estimation methods focus on the functional unit (componentwise or corewise) power estimation based on the measured temperature and total power [7]–[10]. But those methods require understanding of the architecture details and functional units of each chip and many approaches are still *ad hoc*, involving manual turning. At the same time, postsilicon full-chip power (density) map estimation is also important for power verification and package design. Power map is a 2-D spatial distribution of heat dissipation in an IC chip. This problem was also coined as the *inverse thermal map to power map* problem as temperature can be more easily measured either directly or indirectly. Many approaches have been investigated in the past [11]–[16]. Most of the proposed methods tried to frame the problem as a nonlinear optimization problem (deterministically or statistically) once the thermal models are known. However, those methods do not work for general off-the-shelf commercial multicore processors where only core-level power can be obtained [16]. Many of those only work for specialized silicon, such as FPGAs [12]–[14], [16]. In addition, they suffer from high computing cost and measurement noise although some mitigation techniques have been proposed, such as using AC power [14]. Recently, a new heat source identification method based on the measured temperature and 2-D spatial Laplace transformation was proposed for general commercial multicore processors [17], but this is not enough for full-chip power or thermal characterization.

Once we know the power inputs, thermal models are needed to compute the temperature outputs. Many power-based thermal models have been proposed including equivalent thermal RC networks [18], [19], architecture level thermal modeling speed up [20], [21], finite difference-based methods, such as HotSpot [22], and finite element-based methods (FEMs) [23].

Most existing power modeling methods and related thermal models, however, do not work well for commercial multicore

Manuscript received August 18, 2020; revised November 25, 2020 and February 25, 2021; accepted May 18, 2021. Date of publication June 10, 2021; date of current version April 21, 2022. This work was supported in part by the National Science Foundation under Grant CCF-1816361, Grant CCF-2007135, and Grant OISE-1854276. This article was recommended by Associate Editor A. K. Coskun. *(Corresponding author: Jinwei Zhang.)*

Jinwei Zhang, Sheriff Sadiqbatcha, Michael O'Dea, and Sheldon X.-D. Tan are with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA 92521 USA (e-mail: jzhan319@ucr.edu).

Hussam Amrouch is with the Institute of Computer Architecture and Engineering, University of Stuttgart, 70049 Stuttgart, Germany.

Digital Object Identifier 10.1109/TCAD.2021.3088081

processors as mentioned before. It is even more challenging for modeling of commercial multicore processors running in the normal working environment with heat sink cooling as it is difficult to directly measure the temperature of the chip's surface. On the other hand, we notice that commercial multicore processors have many on-chip sensors, for instance, the Intel i7-8650U has one sensor for each core. One can leverage those sensor readings to validate the proposed thermal models. However, the exact locations of these temperature sensors are generally not known, neither provided by the processor's manufacturer.

To mitigate the aforementioned problems, in this work, first, we try to obtain the full-chip power density map from the measured thermal maps/images of the commercial multicore microprocessor when heat sink is removed. Second, we provide a new methodology to accurately estimate the thermal map and hot spots of commercial multicore processors running in the normal working environment with heat sink cooling. The obtained full-chip power and thermal maps under normal heat sink cooling can provide many insightful hot spot information, which cannot be obtained by physical sensors and will enable new applications for dynamic thermal/power/reliability management.

Our main contributions are as follows.

1) First, different than all the existing power estimation methods, the new method, based on the first principle of heat transfer, performs a much more efficient 2-D spatial Laplace operation on a given thermal map to obtain the so-called *raw* power density map. This consists of both positive and negative values due to the steady-state nature and boundary conditions of the microprocessors. We study two motivation cases to provide many insights into the relationship between raw power density maps and real power density maps. Our work is enabled by an advanced thermal measuring platform with a high-precision thermal camera and a cooling system installed on the back side of the CPU. This allows us to take explicit temperature images (thermal maps) of CPU die while the CPU is under load.
2) Then based on the total power of the microprocessor obtained using an online CPU monitoring tool, we develop a novel scheme to generate the true positive-only power density map from the measured raw power density map. At the same time, we develop a novel method to compute effective thermal conductivity of the microprocessor die, which is an important parameter for the subsequent thermal modeling.
3) To validate the power density map and the estimated actual effective thermal conductivity of the microprocessors, we construct a thermal model with COMSOL Multiphysics [24]. The model mimics the real experimental setup (without heat sink), with the same boundary conditions used in the IR imaging system. We use the thermal measurements when CPU is in idle status to determine the boundary conditions of the thermal simulation model. Then we use FEM method to compute the thermal map based on the estimated power density map to ensure the computed thermal maps match the measured thermal maps using the FEM method.
4) To develop the thermal models with heat sink cooling, we first try to identify the exact locations of on-chip sensors of commercial multicore processors based on the correlation analysis of measured thermal map traces and on-chip sensor readings for the first time.
5) Next, we construct the second thermal model with COMSOL Multiphysics that mimics the real set up of multicore processors with heat sinks under real working conditions. The model is validated by ensuring that the computed thermal maps using the estimated power density maps match the temperature values obtained from the real sensors of the chip with heat sink. We also manage to keep power maps consistent for both the back cooling and heat sink cooling to minimize the leakage impacts on power.
6) Numerical results show that the proposed power map estimation method is not only more than $100\times$ faster but also more fine-grained than the state-of-art blind power identification (BPI) method [16].

Experimental results on an Intel i7-8650U 4-core processor with back side cooling technique demonstrate 96% similarity (2-D correlation) between the measured thermal maps and the computed thermal maps, which are computed using the estimated power maps and accurately built FEM thermal model. Besides, based on the estimated power maps and the FEM thermal model modified with heat sink, the average absolute error between the computed thermal maps and the measured thermal maps without heat sink is around 1.3°C. Under real working conditions with heat sink obscured, the average absolute error is only 2.2°C over a 56°C dynamic temperature range and about 3.9% error between the computed thermal maps and the real thermal maps at the sensor locations.

This article is organized as follows. Section II reviews the existing relevant work. Section III shows the power modeling framework and IR thermography setup used in this study. Section IV presents the proposed power density map estimation method and the effective thermal conductivity estimation method. Section V details the proposed full-chip thermal map estimation method for commercial multicore processors operating under the normal condition with heat sink cooling. Section VI presents the experimental results and comparisons with the current state-of-art method. Section VII concludes this article.

II. Related Work

Postsilicon power modeling is concerned with finding the powers of functional blocks or power density maps of a whole chip under various workloads. A few existing works have proposed to estimate the component power and the total power of a *real* microprocessor [7]–[10]. One idea is to tune each component unit power until the summation matches with the total power that is measured experimentally [7], [8]. The main difficulty of those approaches, however, is that searching for component unit power values still remains an *ad hoc* approach, which almost always involves manual tuning. Wu *et al.* [9] tried to mitigate this problem by performing linear regression with K-means method to identify the unique power track patterns from the running programs. Dev *et al.* [10] framed the problem as constrained optimization problem once the thermal models are obtained from finite element simulation and measurement. Recently an recurrent-neural-network (RNN)-based approach has been proposed to quickly estimate the thermal and power hotspots based on the system performance metrics, such as Intel's performance counter monitor (IPCM) [17].

At the same time, many postsilicon full-chip power map estimation works have been proposed [11], [14], [16]. Most of those proposed methods tried to frame the *inverse thermal to power* problem as the nonlinear optimization problems as follows:

$$\min \|M \cdot \mathbf{p} - \mathbf{t}\|^2 \tag{1}$$

where M represents the steady-state power to temperature map matrix, which is dependent on the specific thermal models used. $\mathbf{p}$ is a vector that gives the power density at a set of discrete die locations. $\mathbf{t}$ is a vector of the measured or calculated temperatures at the same locations of corresponding power signals. M can be directly measured from the FPGAs [12]–[14], [16] or by using some approximation methods, such as the power blurring method [11], or by using predefined parameterized analytic forms for a special 3-D IC chip along with a parameter regression method [15]. Reda *et al.* [16] showed the power estimation for commercial multicore processors. However, it can only deliver corewise power information based on total power and corewise thermal sensor measurement.

Paek *et al.* [13] added some statistical spins into this problem by computing the maximum likelihood of power $\mathbf{p}$ given a condition of the thermal map $\mathbf{t}$. But it requires an accurate thermal model, i.e., using HotSpot [22] for simulation results to start with for the required accuracy. The author indeed tested the method on a real FPGA chip, but they only achieve 90.7% accuracy on average. This shows the difficulty in building accurate thermal models for real silicon chips.

In summary, first of all, the existing methods do not work well for off-the shelf commercial multicore processors as many of them only work for specialized silicon, such as FPGAs [12]–[14], [16] or special 3-D chips [15]. Second, they suffer from large computing costs as they try to solve nonlinear optimization problems shown in (1). Some of those methods also require special regulation items [11] or scaling or permutations for matrix M [16] during the optimization to enforce some physics laws, which will lead to more computational costs.

On the other hand, recent study show that the relative power density map can be easily obtained by 2-D spatial Laplace transformation of measured or calculated temperature maps based on the first principle of heat conduction [17], [25]. However, there exist several major differences between this work and the published work. First, this work targets a different set of problems: finding the true 2-D power density maps of multicore processors, validating the results via thermal measurements and applying the power maps for thermal map estimation with different cooling configurations and different workloads. The proposed techniques will bring much more useful applications.

Second, the previous two works simply applied Laplace method to obtain the power maps from the thermal maps. However, such power maps, called *raw power maps*, are not physical power density maps since the raw power maps have negative values as shown in Fig. 3. Furthermore, the prior works mainly identify a few major heat source locations by locating the local maxima from the raw power maps, whereas this work tries to estimate power density values ($\mathrm{W} \cdot \mathrm{mm}^{-2}$) quantitatively across the full chip. With the full-chip power map, one can further perform the full-chip thermal map estimation for different heat sink cooling configurations. To the best of our knowledge, this is a novel thermal modeling capability achieved for the first time.

III. Power Density Modeling Setup

In this section we briefly outline the framework of the proposed approach, thermal imaging system, and necessary data collection from the commercial multicore processor.

A. Power Density Modeling Framework

Power map (surface power density distribution) has tight relationship with the temperature distribution, the Laplace transform of temperature and the thermal conductivity. Our proposed approach involves two kinds of data. The first dataset is the thermal maps of CPU measured through a high-precision thermal camera, which senses the infrared emissions from CPU surface and transforms them into images of temperature distribution. The second is the total CPU power consumption over time, which can be obtained through the processor's performance counter monitor (PCM).

For real processors, as we do not know the exact power density distribution, to verify the estimated power density maps, we compare their corresponding thermal maps. The idea is to build a thermal simulation framework, which mimics the real experimental setup of the chip in the thermal imaging system with similar thermal boundary conditions and thermal structures. Hence, the verification flow of the real chip can be summarized as follows.

1) Obtain sufficient number of estimated power maps based on the proposed method. The experimental measurements should include an idle status, meaning CPU has extremely low power, which will be used to set boundary conditions.
2) Build an FEM thermal simulation model that mimics the real structure of the processor die in the thermal imaging system.
3) Substitute the estimated thermal conductivity κ as well as the estimated power map into thermal simulation model as parameters and inputs.
4) Examine similarities between the computed thermal maps and the experimentally measured thermal maps. Higher degree of identity indicates higher precision of power map estimation, *vice versa*.

Fig. 1(a) illustrates the *power modeling flow* of the proposed power density map estimation model and evaluates the accuracy of the estimations based on the ideal cases. Fig. 1(b) shows the *power inference flow* from data resources to the estimated results for real processors during the run-time.

B. Thermal Imaging System

High precision and resolution of thermal map measurements are critical to the estimation results. One thermal imaging method proposed in [26] maximized the explicitness of thermal maps by directly exposing the top surface of CPU die to the camera, while ensuring the CPU's normal thermal condition by cooling it from the back side of the motherboard. Massive heat generated from CPU flows downward through the motherboard into the cooling system, and is dissipated by the quickly circulating coolant.

This work adopts the thermal imaging method proposed in [26] where the coolant flow does not contact the chip directly. It uses a Peltier device (electrothermal devices) with soft thermal pads stacked together between the liquid pipe and the back side of motherboard. Further, we take advantage of a high-precision thermal camera installed closely over the CPU

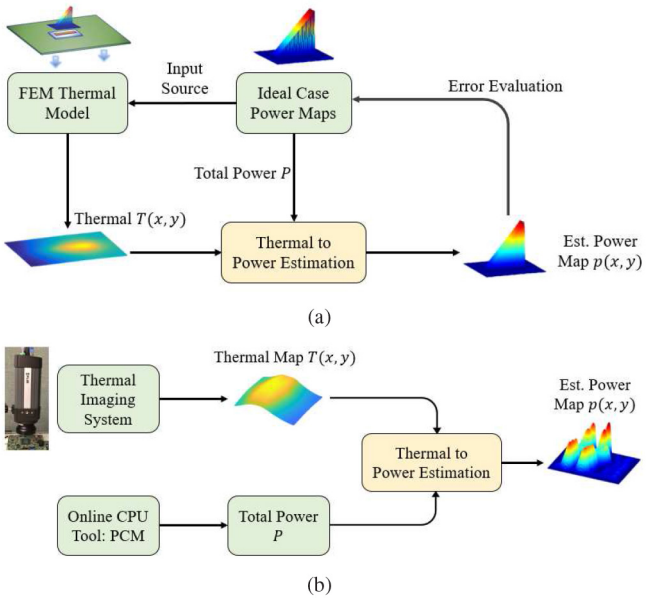


Fig. 1. Framework overview. (a) Power modeling flow. (b) Power inference flow.

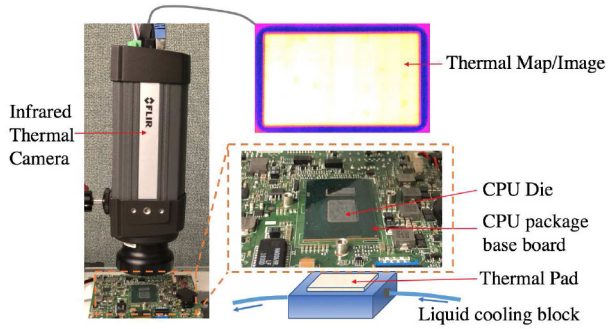


Fig. 2. Thermal imaging system.

die, as shown in Fig. 2. The model of our thermal camera is FLIR A325SC (240×320px images with 16 bit precision and 60-Hz capturing rate). Thanks to a close-up lens, the camera makes temperature difference 50 mK clearly visible within as small as 50 μm/px.

We remark that another thermal imaging system was proposed for power map estimation for commercial processors in [10]. The system uses a transparent silicon window over the surface of the chip and pumps the liquid oil to flow through the window to remove heat from the chip surface directly. After this, FEM method is used to model the setup and generate the power-to-thermal transfer matrix R. Such front-cooling techniques typically require more delicate setup, post imaging processing and more complicated thermal modeling as the oil-based liquid cooling affects the thermal images directly.

IV. New Power Density Map Estimation Method

In this section, we present our new method to estimate the full-chip power density from the real multicore processors. We start with a simple example, which leads to an important observation for the proposed power map estimation method. Then we will present the approach to compute the thermal conductivity of the real chip, which is a critical parameter for thermal modeling and validation.

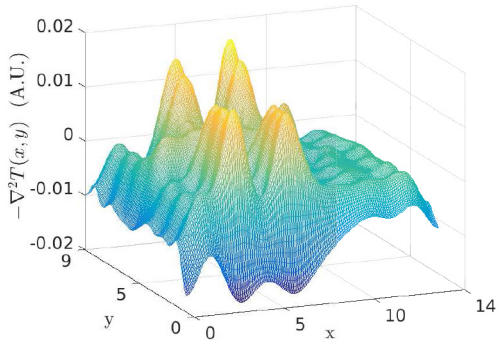


Fig. 3. Negative-Laplacian map (raw power map) example of experimental thermal measurements in 3-D view.

A. Proposed Power Density Map Based on Laplace Operation

Recently, Sadiqbatcha *et al.* [25] proposed an idea of identifying power sources from thermal maps using Laplace transformation. The work starts from the fundamental heat diffusion (2), which gives the relationship between temperature and heat generation

$$\rho C_P \frac{\partial T}{\partial t} - \nabla(\kappa \nabla T) = g \tag{2}$$

where T is temperature (K), ρ is the mass density of the material ($\mathrm{kg \cdot m^{-3}}$), C_P is the mass heat capacity ($\mathrm{J \cdot kg^{-1} \cdot K^{-1}}$), κ is the thermal conductivity ($\mathrm{W \cdot m^{-1} \cdot K^{-1}}$) and g is the spatial heat energy generation ($\mathrm{W \cdot m^{-3}}$).

When CPU runs into steady state, the transient term can be ignored and (2) can be simplified as

$$-\kappa \nabla^2 T = g_T(x,y) \tag{3}$$

where ∇^2 is the Laplace operator. From the simplified heat (3), we can see that the *negative spatial Laplacian* of the temperature distribution across the die is proportional to the spatial heat generation, i.e., the underlining heat-sources $g_T(x,y)$, called the *raw power map*.

This article distinguishes from the prior work in the way this article finds the true 2-D power density maps ($p(x,y)$, $\mathrm{W \cdot mm^{-2}}$) of multicore processors and validates the results via thermal measurements. Specifically, [25] simply applied Laplace method to obtain the power maps from the thermal maps. However, such power maps, called *raw power maps*, are not physical power density maps as they contain negative values. Negative values are clearly shown in Fig. 3. They cannot be explained by CPU power distribution since CPU power will never be negative. Furthermore, the prior work mainly identifies a few major heat source locations by locating the local maxima from the raw power maps without solving for the physical parameter of thermal conductivity κ. However, this article quantitatively estimates the power density values ($\mathrm{W \cdot mm^{-2}}$) across the full chip as well as the thermal conductivity κ of the chip die. This means we are able to learn the spatially continuous heat sources and their actual power densities.

In order to closely study the relationship between the Laplace transform of temperature and the CPU power distribution, we build a simple ideal case in COMSOL Multiphysics heat transfer tool [24]. This structure contains a rectangular base whose geometric dimension is 10×15×0.5 mm, and a 4×4×0.5-mm heat source block embedded in the base, whose

total power is set at 3 W (0.1875 $\mathrm{W}\cdot\mathrm{mm}^{-2}$) and homogeneous in space [Fig. 4(a)]. The geometries can be flexible, we set it to approximately match the general size of CPU die and core. The κ of the material of the structure in this case is 400 $\mathrm{W}\cdot\mathrm{m}^{-1}\cdot\mathrm{K}^{-1}$. For the boundary conditions, a convective heat flux set at 1000 $\mathrm{W}\cdot\mathrm{m}^{-2}\mathrm{K}^{-1}$ is applied to the bottom surface. This convective heat flux mimics the heat dissipation through bottom surface. Ambient temperature is set to room temperature 297 K.

As shown in Fig. 4(a), it is obvious that the high-rising portion of negative-Laplacian map reflects the area of active power density. Furthermore, we observe that integration of negative-Laplacian map over all the area (pixels) is always zero, no matter how the power setting or geometry changes. The reason is that the thermal map we obtained comes from steady state of the CPU with specific thermal boundary conditions. This means that power generation and power dissipation are balanced in such equilibrium state. The negative power density value actually stands for more power dissipation than generation at the specific location due to thermal transfer and convection process at the boundaries. Where positive value means the opposite. For the very positive high-rising portion, which means the heat generation is significantly larger than the dissipation, typically indicates the hotspots of the chip.

In another example, we have an ideal linear heat source, whose power density increases linearly along the x-axis with total power 5 W. Fig. 4(b) illustrates the location of power source, power setting and its corresponding negative-Laplacian map. We can observe that the negative-Laplacian in such rectangular power region shows an important linear trend as well, while the surrounding region is negative.

Based on the observations from these examples, we can see that the positive part of the negative-Laplacian map are the region where most of the real power densities are located. In this two simple cases, they cover the 100% real power density distribution. As a result, we can just use the *positive part* of the negative-Laplacian map to represent the estimated power map. Though the actual values of power map in those region are yet to be determined, which will be answered in the following section by calculating the accurate thermal conductivity κ.

B. Estimation of Real Thermal Conductivity

Modern microprocessor die is usually as thin as 0.5 mm or below. Thus, thermal characteristics along z-axis can be viewed as homogeneous. Power density distribution is only important on the surface x-y plane.

In reality, heat density is a combination of CPU power and heat dissipation by heat sink. Assume the thickness of CPU die is Δz, $p(x, y)$ stands for surface power density ($\mathrm{W}\cdot\mathrm{mm}^{-2}$) at location (x, y) and $p_d(x, y)$ denotes heat dissipated locally. Heat density can be expressed as

$$g_T(x, y) = \frac{p(x, y) - p_d(x, y)}{\Delta z}. \tag{4}$$

Then for location (x, y), (3) can be rewritten as

$$-\kappa\nabla^2 T(x, y) = \frac{p(x, y) - p_d(x, y)}{\Delta z}. \tag{5}$$

Considering the entire chip, integrate both sides on the whole die area

$$-\kappa\int \nabla^2 T(x, y)dxdy = \int \frac{p(x, y) - p_d(x, y)}{\Delta z}dxdy. \tag{6}$$

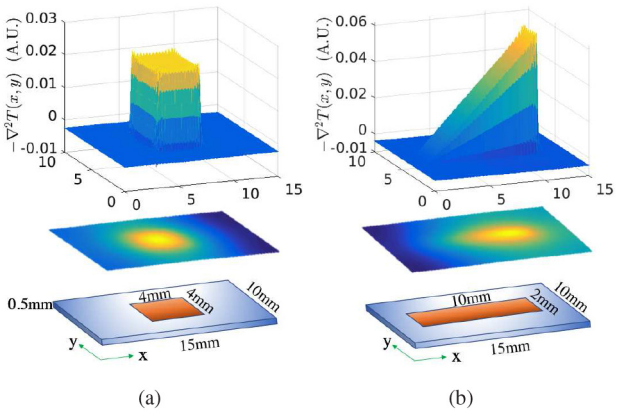


Fig. 4. Simple ideal cases. (a) Homogeneous heat source in orange region with total power 3 W. (b) Linear heat source in orange region (10×2 mm), with areal power density $0.05(x-2)$ $\mathrm{W}\cdot\mathrm{mm}^{-2}$, $x \in [2, 12]$ and total is 5 W.

Suppose P is total CPU power, and P_d is total heat dissipation (mainly through convective heat flux), (6) can be further written as

$$-\kappa\int \nabla^2 T(x, y)dxdy = \frac{P - P_d}{\Delta z}. \tag{7}$$

At steady state P should be equal to P_d. This infers the integration on the right hand side of (7) would give zero total heat, as CPU power is balanced with heat removal. It also implies the integrated Laplacian should be zero. In fact, this zero result has been observed both in our experiments and aforementioned simulation. Based on the discussion in the previous section, (7) can be approximated as

$$-\kappa\int_{S_P} \nabla^2 T(x, y)dxdy \approx \frac{P}{\Delta z} \tag{8}$$

$$\kappa \approx \frac{P/\Delta z}{-\int_{S_P} \nabla^2 T(x, y)dxdy} \tag{9}$$

where S_P indicates area where negative-Laplacian of temperature is positive. Since die thickness Δz is constant, once negative-Laplacian map is obtained from temperature image, the equivalent thermal conductivity κ can be obtained. It basically means that the proportional factor κ can be acquired from dividing total power by thickness and by areal integration of the positive parts of negative-Laplacian. Having this κ, CPU power density map becomes straightforward, which is expressed as

$$p(x, y) = \begin{cases} \kappa\Delta z\left[-\nabla^2 T(x, y)\right], & -\nabla^2 T(x, y) > 0 \\ 0, & -\nabla^2 T(x, y) \le 0. \end{cases} \tag{10}$$

Using the above equations to estimate the power map for the homogeneous heat source example and the linear heat source example, the results are shown in Fig. 5. Fig. 5(c) and (d) is the estimated power densities for the two cases, while Fig. 5(a) and (b) is the corresponding original power density maps. As we can see, some spikes exist at corners in the estimation results due to numerical noise.

To compare the similarity of the two power maps, we introduce *2-D correlation coefficient*, or simply *correlation* to evaluate the similarity between the real power map and the

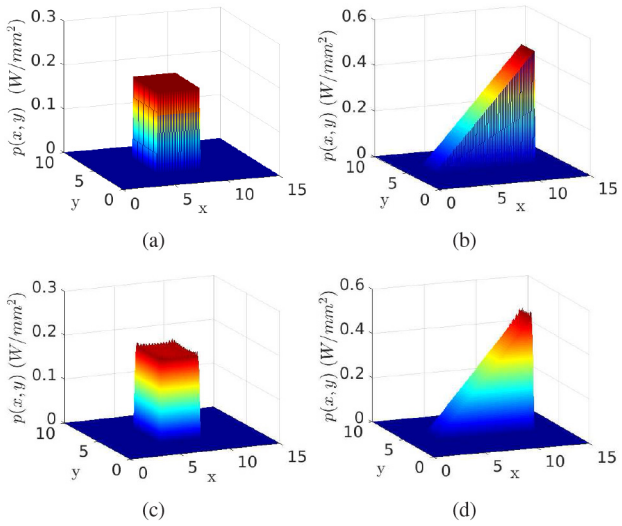


Fig. 5. Comparison between estimated power density maps and exact ones for simple ideal examples. (a) and (b) are original power density maps for homogeneous heat source and linear heat source, respectively; (c) and (d) are the corresponding estimated power density maps of (a) and (b).

estimated power map, which is defined as

$$r = \frac{\sum_m \sum_n \left(A_{mn} - \overline{A}\right)\left(B_{mn} - \overline{B}\right)}{\sqrt{\left(\sum_m \sum_n \left(A_{mn} - \overline{A}\right)^2\right)\left(\sum_m \sum_n \left(B_{mn} - \overline{B}\right)^2\right)}} \tag{11}$$

where $\overline{A}$ and $\overline{B}$ are mean of all entries in A and B, respectively. r is a scalar between 0 and 1, the more it approaches 1 the more they look alike. For the above two examples, the correlations of the first and second example are 0.977 and 0.973, respectively. In addition, RMSE of estimated power map on the active powered region is 0.005 $\mathrm{W \cdot mm^{-2}}$ and 0.015 $\mathrm{W \cdot mm^{-2}}$, respectively, for the two cases as well. As a result, we can see that the proposed power map estimation method is quite accurate.

The thermal conductivity κ of silicon is about 130 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$, copper is about 400 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$. Due to the mixture of silicon, copper and some other materials in real die, the overall κ could be somewhere around 130–400 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$. The material in the motivation examples in simulation has κ of 400 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$. In our case, the estimated κ by the proposed method is about 417 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$, about 5% error for the estimation.

One may wonder why the estimation error is 5% for κ based on those ideal simulation cases. There are several reasons. The first is the nonzero thickness of the chip hence the thermal distribution is not perfectly homogeneous along the vertical z-axis of chip die. The second reason is the nature of finite-element method, meaning elements are not infinitely fine on the object. We can see some glitches on the power map surface. The side surfaces are not absolutely vertical but with a small angle from the vertical plane. Ideally, the sides should be absolutely vertical. Third, a very small part of the low powered area that only has slight power density (e.g., smaller than 3% $\sim$ 5% of the average power) are computed as zero power because of ignoring the negative values in the raw power map. In reality, the above mentioned extremely low powered area may still show negative Laplacian values due to the heat diffusion behavior, such as vertical diffusion within the chip. To look at the error more closely, we repeat the trials with reduced thickness and the finest mesh structure in FEM in COMSOL for several ideal cases. The κ estimation error could be improved to within $\pm 1\%$ for ideal homogeneous heat sources (square or rectangle shaped, etc.), and 0%–3% for the ideal linear heat source case. Further from this observation, the error is workload dependent due to the aforementioned third reason. As we see, the error is different for ideal cases. One consideration is the linear heat source case has more errors happening at the border of heat source, where there is a small part of places at the border with positive power are still shown as negative values in the simulation setup and hence are zeroed out. In another words, the proposed κ approximation method is not perfect but is a reasonable estimation.

Therefore, we have verified the approach of estimating the power map in simulations. Moreover, the estimated power density maps sufficiently match the original power setting. We further note that the estimation error for κ can be smaller or equal to 5% from the simple ideal cases. However, it is difficult to know the true accuracy since we do not know the actual κ for the commercial chips.

C. Thermal Conductivity Estimation for Real Chip Die Area

Another important parameter for the thermal model is the thermal conductivity of the chip. Based on the power map model derived in Section IV-B, we show in this section how to estimate the equivalent thermal conductivity κ of the die from the measurement of thermal maps.

For our work, the total power of CPU is also needed. IPCM provides users a software interface that estimates the internal resource utilization of the latest Intel core processors. One metric of the PCM dataset is CPU energy consumption between two accesses. To ensure precision, power data has to be synchronized with the thermal maps. As mentioned in the system setup, if the capturing frequency of infrared camera is f, PCM data should be recorded in this same frequency. Suppose the CPU energy along discretized time points is series E, then total power $P = E/\Delta t$ is also a time series, and $\Delta t = 1/f$.

One thermal map is related to one raw power map, and it will result in one κ value. Different thermal maps may result in different κ values. A reasonable κ should be a constant despite different workloads and time. In this work, estimating κ accurately is important to the thermal characterization for that the FEM thermal model depends on the κ parameter. It is also important to power map estimations due to the proportional factor, as seen in (10). And the accuracy of κ depends on the accuracy of thermal imaging measurements. Therefore, we have analyzed a sufficient amount of imaging samples regarding various workloads to obtain an optimal κ expectation. In this work, we execute eight workloads of different kinds and capture over 14 000 thermal maps for each workload. The workloads are *gimp, aobench, phpbench, cachebench, tinymembench, build-gcc, compress-7zip* and *cyclictest*, respectively. The resulted κ with respect to different workloads along time line are plotted in Fig. 6. See from the traces, κ comes out quite constant and for half of the workloads and there is an obvious overlap, which is expected. On the other hand, it is observed that the mean κ of workloads *tinymembench*, *phpbench*, *aobench* and *build-gcc* are measured 160, 164, 165 and 180 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$, which have slightly large deviation compared to other workloads. In our

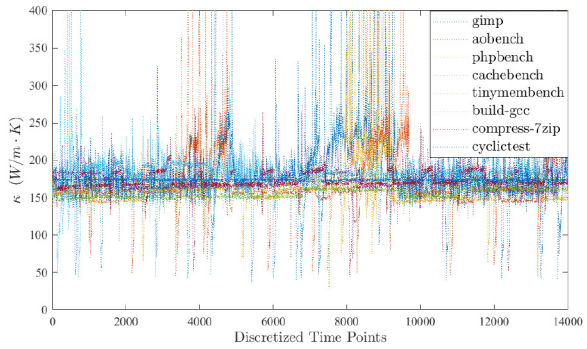


Fig. 6. Estimated thermal conductivity of die area with respect to multiple workloads in time domain.

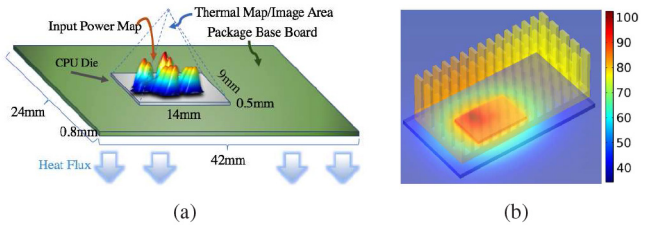


Fig. 7. (a) Thermal structure created to imitate the real experiment setup (without heat sink). (b) Transparent view of processor area when with heat sink mounted on.

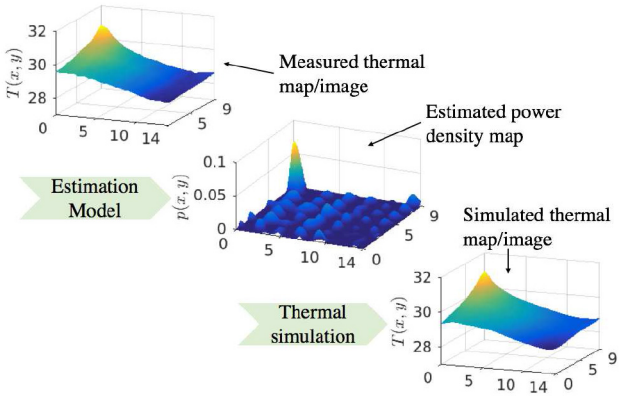


Fig. 8. Setting proper boundary conditions for thermal simulation model using measurements of CPU's idle status.

experiment, the workloads are selected such that: 1) they contain steady states during the execution since this work focus on the steady state power and thermal estimations and 2) they invoke enough power to generate measurable heat and infrared emissions so that the infrared camera captures the chip's spatial temperature efficiently. As seen in the results, the κ seems workload dependent. Possible reasons are that the top surface temperature does not perfectly follow the temperature inside the chip as the material is not perfectly homogeneous, as a result, the error is dependent on workloads and the thermal measurement error. Some glitches exist in κ due to CPU changing power levels thus not running in steady state. During those times, temperature transient term ($\rho C_P[\partial T/\partial t]$) cannot be ignored. Furthermore, the global arithmetic mean of κ is 174 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$. We define this κ as the estimated equivalent thermal conductivity of CPU die.

In theory, κ is related to the whole system (die + FR4 (Circuit Board) base + motherboard + cooling pads). However, the die has much greater thermal conductivity than the FR4 base below it, which allows us to separate the die out of the system when computing κ. For instance, if the FR4 base and motherboard were metal or silicon with the same κ as the die, then we cannot treat the chip as a thin piece anymore. The chip and boards should be treated as a homogenous object. Thanks to the fact that die has far greater κ than its base, we can approximate the $\kappa = 174\ \mathrm{W \cdot m^{-1} \cdot K^{-1}}$ as the κ of silicon die. However, the resulting κ is higher than that of the pure silicon. This could come from the fact the metal layers inside of the die and the metal protection covering the surface of the die increases the overall κ.

D. FEM Model Architecture to Imitate Real Experiment Setup

In this section, we present the FEM thermal model that computes (reconstructs) the thermal map for the test setup with bare chip and back cooling of the processor and a novel approach to validate the results.

Once we obtain the estimated power density maps and equivalent thermal conductivity κ of the CPU die, we then start to build a heat transfer structure that mimics the real experiment setup. Fig. 7(a) illustrates the structure created using COMSOL Multiphysics which matches the real CPU package geometry. Geometries of Intel i7-8650U are acquired from the open resource from WikiChip organization [27]. The CPU package dimension is 42×24×1.3 mm, thickness of base circuit board is 0.8 mm. There are two pieces of dies soldered on the base board, the CPU die, which is the object we will study, has dimension 14×9×0.5 mm. In our model, material of the CPU die and package base board are initialized with silicon and FR4, respectively. However, thermal conductivity of die part (silicon) is set to the computed κ, i.e., $\kappa = 174\ \mathrm{W \cdot m^{-1} \cdot K^{-1}}$ in our case. A convective heat flux is applied to the bottom surface of the package base board, which simulates the heat flow from CPU package through motherboard to the cooling system, as indicated by the arrows at bottom in Fig. 7(a). The convective heat flux rate and thermal conductivity of base board will be determined as boundary conditions in the next.

For the FEM thermal simulation, we also need to know the correct thermal boundary conditions of the die. One idea is to explore the idle status of CPU (its boundary conditions are the same of the CPU under other workloads) as it is easy to extract the power map in this status. Specifically, since the idle status has extremely low power, power map is preknown as almost zero, except for very few places that have slight power. Majority of CPU appears to be approaching ambient temperature, spatial temperature appears relatively flat. At the beginning, simulated thermal map according to the estimated idle power map has the same trend with the measured thermal map, whereas the amplitude and the range have a little discrepancy. This will guide us how to adjust the thermal conductivity of package base board and bottom convective heat flux rate such that the simulated thermal map matches the measured thermal map as much as possible for idle status, as shown in Fig. 8.

The boundary conditions for the FEM model are determined systematically and methodologically. Geometry dimensions are known and thermal conductivity of chip die has been calculated. We first feed the estimated idle power map to FEM

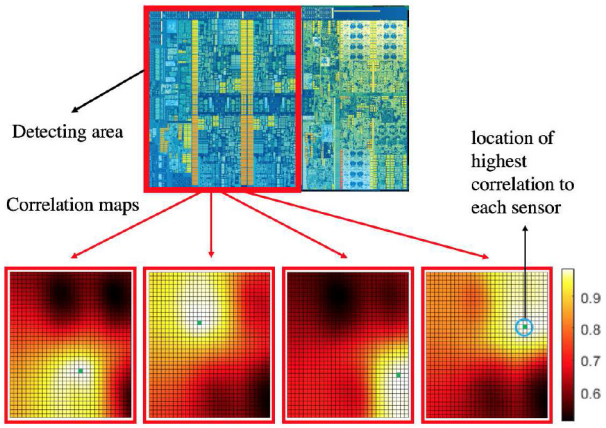


Fig. 9. Correlation between pixel blocks and sensor values of each core. In each correlation map the pixel block that has the highest correlation to the core sensor measurement is marked in green dots, which infers to the on-chip sensor location.

and manually adjust the temperature of the bottom interface of the base board at somewhere around 30 °C, making the FEM simulated thermal map to mirror the measured thermal map, as Fig. 8 shows. It needs a couple simple trials, once the simulated thermal map of the idle status mirrors the measured thermal map, temperature of base board bottom interface is then set and fixed. The idle power map has extremely low power across the chip only except for a corner area thus it is the best choice for setting the interface temperature. Then we prepare an estimated power map of an arbitrary active workload at an arbitrary steady state during runtime. We feed that power map to the FEM model and adjust the thermal conductivity of base board again, making the temperature range of the simulated thermal map equal to that range of measured thermal map. This is because the thermal conductivity has a scaling effect to the temperature range. Finally, adjust the heat flux rate at the interface such that the max temperature equal the measured max temperature. Note that this is one-time action for all, meaning these parameters are fixed in later tests for various workloads, including their various steady states of runtime. For the setup with heat sink, the physical parameters are computed in a similar way. The difference is the usage of (14) to compute the heat transfer coefficient or heat flux from heatsink to ambient. In our simulations we found that these one-time calibrated parameters perform well for all the data samples. We found that the convective heat flux rate is about 600 $\mathrm{W \cdot m^{-2} \cdot K^{-1}}$ and thermal conductivity of base board is about 6 $\mathrm{W \cdot m^{-1} \cdot K^{-1}}$.

Note that this FEM thermal model is built to validate the accuracy of estimated power maps. As mentioned before, since the exact power map cannot be directly measured, we take an indirect method to to validate it by comparing the measured thermal maps to the FEM computed thermal maps. The power map accuracy results are described in details in Section VI-A.

V. Full-Chip Thermal Modeling With Heat Sink Cooling

In this section, we present our accurate thermal model for real multicore processors with heat sink and the validation method. We first show how to identify the locations of physical thermal sensors of commercial off-the-shelf processors, whose sensor locations are usually not available publicly.

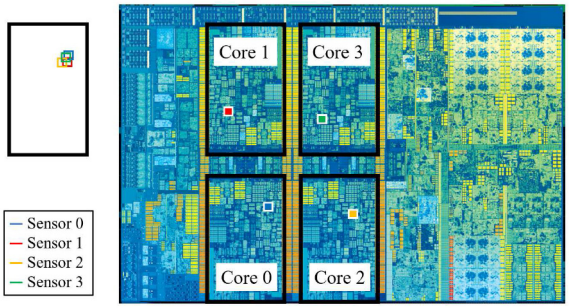


Fig. 10. Sensor locations of CPU cores are identified in colored squares on an Intel i7-8650U quad-core processor [27].

A. Identify Exact Locations of On-Chip Thermal Sensors

Typically, each core of the commercial multicore processor contains at least one thermal sensor. However, the exact locations of these sensors are not disclosed by the chip vendors or developers publicly.

The exact location of those thermal sensors are needed to be known in order to model the full-chip thermal map which is obscured by the heat sink. In this section, we present our novel method to identify the exact locations of those thermal sensors, which serves as a basis for the following sections.

Thermal sensor values are accessible in the runtime through online CPU tool PCM. In order to locate the embedded thermal sensors, one way is to measure the temperature image of the chip without heat sink and find the pixel location that matches the sensor value. However, there are two obstacles. The chip's top surface temperature measured by the thermal imaging system is lower than the internal temperature measured by the sensors. Besides, the difference between externally and internally measured temperatures is unknown.

To address this issues, we propose a correlation-based method to identify the exact sensor locations. First, we capture a series of temperature images of the chip without heat sink while it is running under workloads. Workloads activate different cores and heat up different places of the chip earlier or later, which enable all the sensors to have different temperature records along the timeline. To be more specific, if the recording time is sufficiently long, each different location of the chip will show distinct time curve of temperature measurements. Then we divide each of the measured temperature images into 5×5 pixel square blocks, let the average temperature of this square be the temperature of this location. One pixel is 50-μm wide thus the resolution is 250 × 250-μm block. This resolution is fine enough considering the chip size. We mark the location of each block and trace their temperature along the time from all the measured temperature images. After collecting the temperature series of each pixel block, each series is compared against the sensor values. Although externally measured temperatures are lower than sensor values, tendency of both variations will be the same, meaning the external temperature rises or falls as the internal temperature does. Therefore, the temperature series on the exact sensor location will have the highest correlation with the sensor values. Linear correlation is applied, which is defined as

$$\mathrm{corr}(T_m, T_s) = \frac{\mathrm{E}\left[\left(T_m - \overline{T_m}\right)\left(T_s - \overline{T_s}\right)\right]}{\sigma(T_m)\sigma(T_s)} \tag{12}$$

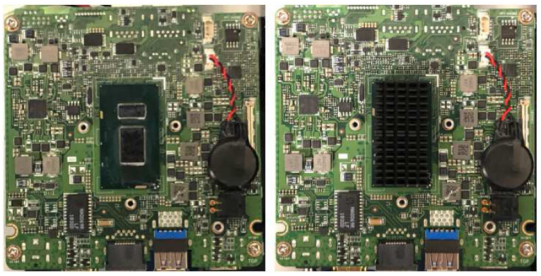


Fig. 11. Processor without heat sink (left) and with heat sink (right).

where T_m and T_s are the temperatures measured by thermal system and sensors, respectively. $\overline{T_m}$ and $\overline{T_s}$ denote the mean and $\sigma(T_m)$ and $\sigma(T_m)$ denote the standard deviations.

Correlation maps in Fig. 9 illustrate the correlation between the temperatures of pixel blocks and sensor values of each core. Pixel block having the highest correlation in each correlation map, marked as a green dot, indicates where the on-chip sensor location is identified. Furthermore, the identified sensor locations for all cores are illustrated in Fig. 10 as small colored squares, where the black rectangles outline the core regions. The chip layout is sourced from the open source Wikichip Organization [27]. When overlapping and aligning the cores we can see that the sensor locations identified in all the cores have quite good consistency.

B. FEM Thermal Modeling With Heat Sink

We have elaborated on the FEM architecture that imitates the bare chip and back cooling situation in Section IV-D. As we know, under real working situation, the back side liquid cooling is replaced by the top side heat sink cooling (either passive or active heat sink). The processor on device setup without and with heat sink is shown in Fig. 11. Fig. 7(b) illustrates the FEM structure of processor area in a 3-D transparent view when it is covered by a fin-shaped heat sink, where the dimensions of the heat sink strictly follow the object in the experiment. Compared to the previous one, this FEM model replaces the back side heat transfer with heat transfer through the fin-shaped heat sink.

We know that the computed thermal map cannot be compared against the exact thermal map since the chip is obscured by heat sink and not measurable through thermal imaging system. However, the results can still be validated in a sense if the computed temperatures on thermal sensor locations match the real on-chip sensor measurements. We compute the thermal maps by FEM-based thermal simulation using COMSOL Multiphysics. The FEM thermal model architecture basically consists of four major components to simulate the real device setup - motherboard, processor base board, processor die itself and the heat sink. We remark that the model can be customized to any setup, such as adding heat spreaders or using more sophisticated heat sinks.

Ambient temperature and the convective heat transfer rate of the heat sink to ambient are critical environment information for the FEM thermal model. To find out these parameters, we again include the temperature measurements of processor's idle status with heat sink in our analysis. The heat transfer per unit surface through convection can be expressed as

$$q = h_C A \mathrm{d}T \tag{13}$$

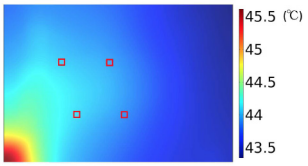


Fig. 12. Computed temperature of processor with heat sink under idle status. The small red boxes mark the on-chip thermal sensor locations.

where q is the heat transferred per unit time (W), which can be approximate as the processor power. A denotes the surface area of heat sink (m^2), h_C denotes the convective heat transfer coefficient ($W \cdot m^2 \cdot^\circ C^{-1}$) and dT is the temperature difference between the heat sink and ambient. We use the processor power and average sensor temperature when the processor is in idle status and under workloads to compute h_C according to

$$\begin{aligned} p_{\text{idle}} &= h_C A dT_{\text{idle}} \\ p_{\text{wkld}} &= h_C A dT_{\text{wkld}} \\ p_{\text{wkld}} - p_{\text{idle}} &= h_C A (T_{\text{wkld}} - T_{\text{idle}}). \end{aligned} \tag{14}$$

The average processor power when all cores are under workload p_{wkld} and in idle status p_{idle} are about 10.5W and 0.6W, and the corresponding average sensor temperature are $T_{\text{wkld}} = 100\ ^\circ\text{C}$ and $T_{\text{wkld}} = 44\ ^\circ\text{C}$, respectively. The surface of heat sink is measured as 60 cm^2. Hence, the convective heat transfer coefficient h_C is acquired as $29.5 W \cdot m^2 \cdot^\circ C^{-1}$. As before, the ambient T_A around the heat sink can be obtained using idle status by adjusting the ambient in FEM simulation such that the simulated sensor location values match the real sensor values for idle status. Fig. 12 shows the computed thermal map of idle status under heat sink, in which the computed sensor location temperatures match the real sensor measurements well when ambient is 33 °C (306K). Therefore, the FEM thermal model for the processor setup with heat sink mounted is properly built.

C. Ensuring the Same Power Density Maps for Both Coolings

One important aspect of our thermal modeling methodology is that we need to ensure that the power density maps obtained with back cooling and the one with heat sink cooling should be keep as the same as possible. We remark that this is required only for building the thermal models for the chip with heat sinks. Once the model is built, it can be used for different workloads with total different power maps.

This requires the processor to run a series of workloads without heat sink first, and then run the same workloads with heat sink on and with the same CPU core scheduling. Under this condition, we can claim the power distributions will remain the same for both with and without heat sink, even though the thermal sensor data varies.

In our study, single-threaded and mult*i*threaded workloads from Phoronix Test Suite are used. For the setup with and without heat sink, single-threaded workloads, such as *aobench* and *cachebench* are forced to run with the same core mapping by setting the workload's CPU affinity. Mult*i*threaded workloads, such as *compress-7zip* and *cyclictest* do not need forced core mapping since all cores are utilized and same scheduling. Thermal sensor values of some time-segments of workloads are shown in Figs. 13 and 14 for back-side cooling and heat

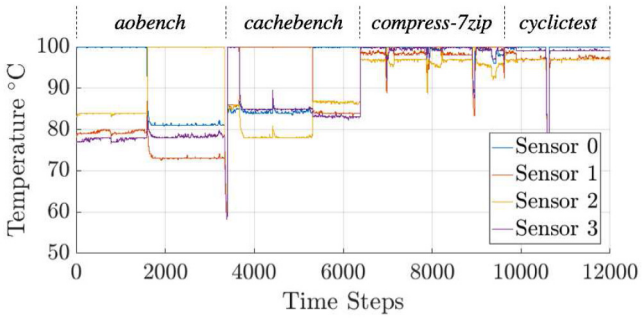


Fig. 13. Thermal sensor values when processor uses back-side liquid cooling with respect to workloads.

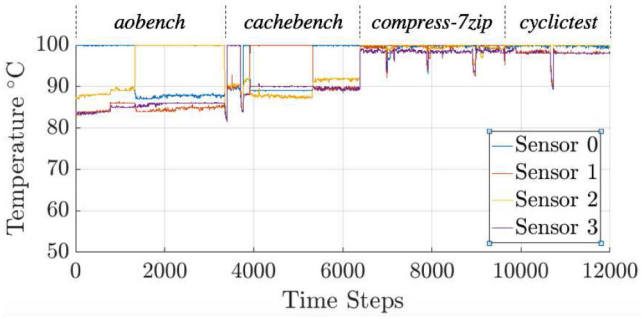


Fig. 14. Thermal sensor values when processor is mounted with heat sink with respect to workloads.

sink cooling, respectively. The relatively high (100 °C) or low (80 °C–90 °C) temperature indicates the corresponding core is in busy or idle working status. As we can see, for *aobench* and *cachebench* on both cooling setups the sensor of busy core reaches 100 °C, which is the thermal spec temperature, whereas the other three cores are about 10 °C cooler. And for *compress-7zip* and *cyclictest*, all sensor temperatures are near the maximum values.

Furthermore, we ensure both of the cooling techniques - with back-side cooling or with top-side heat sink have similar or close cooling capability, so that the total power is the same for both cooling solutions and CPU can work under similar thermal conditions in both scenarios. As a result, the amount of leakage current would also be similar or close in both scenarios. Our study shows that the leakage distribution difference will not significantly affect the consistency of power maps in the two cooling scenarios under this condition. DVFS conditions are also kept in nominal status, meaning no unexpected frequency kick-down by over heat (actually we only require the DVFS are same in both coolings). The corresponding total power consumptions of both cooling scenarios during the time line are plotted in Fig. 15, where we can see two power traces follow almost identical trend, despite very slight variations.

D. Application for Different Workloads

The proposed thermal model with heat sink can be applied to different workloads once it has been built as the thermal model is workload-independent in theory. The only thing is that one has to obtain the accurate power maps for the workloads first. For different workloads, one way is to go through the same power map characterization process using thermal imaging system as we discussed in this article and ensure that the back cooling has the same cooling capability as the heat sink cooking in the sense of total power. However, this is laborious as we need to ensure the running settings

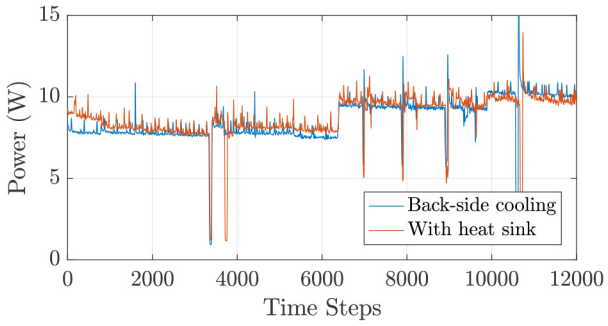


Fig. 15. Total processor power with back-side liquid cooling and with heat sink during the time line.

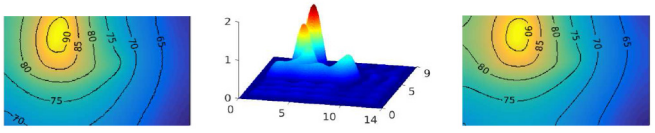


Fig. 16. Power map estimation result for *aobench* (ts = 10 750): (a) Experiment measured thermal map. (b) Estimated power density map in 3-D view. (c) FEM simulation reconstructed thermal map.

(task-to-core mapping, DVFS) are same. Estimating accurate power density map or map series for commercial multicore processors for different workloads is a difficult problem. The proposed thermal map to power map-based power map characterization can still be used as the tool to collect the data to train the machine-learning-based model-based on the real-time utilization metrics, such as on chip Intel's Performance Counting Monitor [28], DVFS setting and task-to-core assignment and scheduling. We believe this will bring huge online real-time thermal modeling and monitoring capability which is not available for today's commercial multicore processors. Recent studies show that one can built very accurate data-driven deep neural network model to map from IPCM to full-chip thermal maps in real-time [29] which is assisted by back cooling and infrared imaging system. As a result, we can extend that technique for full-chip power map estimation so that there is no need to go through the same power map characterization process tediously. Instead, we can perform real-time thermal map estimation for commercial multicore processors under heat sinks. But this can be our future work and is not the focus of this submission.

VI. Experimental Results and Discussions

In this section, we will present the results with the experiment of Intel i7-8650U in two stages, which has 4 CPU cores, and an integrated GPU. First, we present the estimated power map results for the 4-core processor and validate the estimated power density maps by finite element-based thermal simulation with no heat sink using COMSOL Multiphysics as discussed earlier [24]. Then we compare it against a recently proposed power map estimation method [16]. Second, with the estimated power maps, we presented the accurate thermal models for estimating full-chip thermal maps when the processor works in real situation under heat sinks.

A. Power Map Estimation and Comparison Results

We have first examined sufficiently large amount of data samples that relate to various steady states when with back side

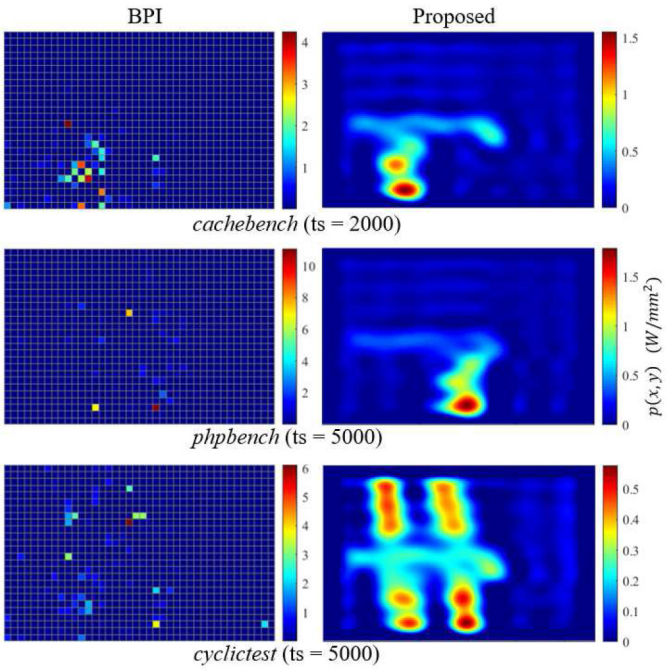


Fig. 17. Comparison of the estimated power maps between the BPI (27×41 resolution) and the proposed method for three benchmarks (*cachebench*, *phpbench*, and *cyclictest*) with respect to a specific time step during their full execution, respectively.

cooling. For most data samples, the estimated power maps are able to reconstruct the thermal maps that are almost identical to the measured thermal maps, with average similarities over 96%. Besides, the average absolute error for thermal maps reconstructed by estimated power maps and FEM model is 1.3 °C. Fig. 16 lists an example of the power map estimation results when running under workload *aobench* at the time step (ts) 10 750 during its execution. From left to right are the experiment measured thermal map, the estimated power map in 3-D view, and the thermal map reconstructed from FEM thermal simulation in the back-side cooling scenario.

As mentioned in the related works in Section II, Reda *et al.* [16] proposed a general blind power identification, called BPI method for power maps from thermal measurements. For fair comparison, we utilize the open source code of BPI implemented in MATLAB program [30] with our thermal measurements and compare it with the proposed method, which is also implemented in MATLAB. We choose arbitrary steady-state time steps of workloads (*cachebench*, *phpbench* and *cyclictest*) as the target of power map estimation. Fig. 17 left-hand side illustrates the power maps under 27 × 41 resolution estimated by the BPI method, and the right-hand side illustrates the power maps under 190 × 290 resolution estimated by our method. It is apparent that the power patterns of the proposed power maps actually match the powered blocks in the power maps estimated through BPI. The total integrated power over the power maps equal to each other as well. The powered blocks may have excessive power density because of pixilation. In one word, the proposed method provides a more fine-grained estimation than the BPI method.

Furthermore, the proposed method is more efficient than the BPI method, as shown in Table I. For BPI method to achieve high-resolution power maps, the size of required response

TABLE I
COMPUTATIONAL COST COMPARISON FOR POWER MAP ESTIMATION

	BPI			Proposed
Map Resolution (pixel by pixel)	13 × 20	19 × 29	27 × 41	190 × 290
Time Per Single Power Map (ms)	39	277	740	7.5

TABLE II
ESTIMATED POWER EXAMPLE FOR PROCESSOR COMPONENT (I7-8650U)

Component	Power	Component	Power
System Agent	0.70W	Ring/Interconnect	1.78W
GPU	0.59W	L3 cache	0.53W
Core#1	2.22W	Core#2	2.36W
Core#3	2.27W	Core#4	2.51W

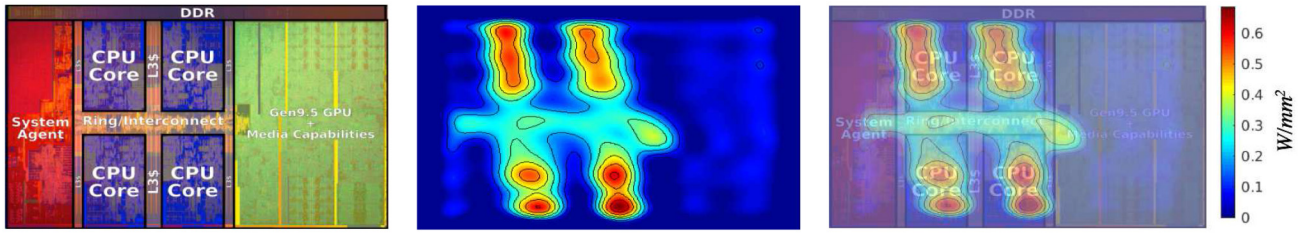
matrices will grow exponentially as the resolution increases. Actually achieving resolution beyond 27 × 41 becomes challenging due to excessive computational costs of the underlying optimization. Note that we used MATLAB tool to perform the computation on an Intel i7-8750H 2.2-GHz PC. As we can see, the proposed method is much more efficient even with higher resolution, which is essentially the resolution of the thermal image.

Fig. 18 further illustrates the power density distribution [Fig. 18(b)] projected on the processor die floor-plan [Fig. 18(a)] at an example steady state. Intel i7-8650U processor has a system agent (including an image processing unit, a display engine and an I/O bus), a GPU module on the side and four cores in the middle. As we observe, the four cores consume the major power, whereas the system agent and GPU module consume low power at such steady state [Fig. 18(c)]. We can see that power pattern aligns with the arrangement of cores and Ring/Interconnect quite well. We obtain the component power by power density integration for the component, which are presented in Table II.

B. Thermal Map Results Under Real Working Conditions

In this section, we will first present the reconstructed thermal map results from our FEM model created with COMSOL Multiphysics for this commercial multicore processor underneath the heat sink cooling. We have examined various steady states for both single-threaded and multithreaded workloads to compare the computed temperature at sensor locations under the heat sink against the real sensor measurements. Full-chip processor heat maps for certain typical power scenarios of workloads are illustrated in Fig. 19. Temperatures extracted from those computed heat maps are compared against real sensor measurements and listed in Table III. In the test, the estimation error varies with respect to the workloads. From the results listed in Table III we observed that *compress-7zip* has the best accuracy (0.5 °C average error), whereas *phpbench* has the largest error which is shifted up by about (3.3 °C) in such thermal steady state. In the time axis, the maximum absolute error for *phpbench* and *cyclictest* in the worst scenario is between 3∼4 °C.

To analyze the estimated temperature in the time axis, we utilize HotSpot [22] to compute the temperature at sensor locations. Power maps can be obtained from the thermal images



(a) (b) (c)

Fig. 18. (a) Intel i7-8650U processor die floor-plan [27]. (b) Estimated power density map. (c) Projection of power density map onto the processor die floor-plan.

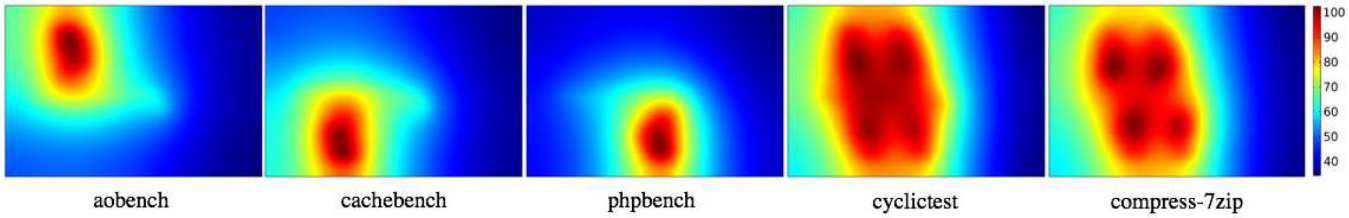


Fig. 19. Full-chip thermal maps of processor under real working conditions with heat sink mounted on with respect to various workloads.

TABLE III
FEM COMPUTED TEMPERATURE AT SENSOR LOCATIONS VERSUS REAL SENSOR VALUES WITH HEAT SINK

Workloads		Sensor0 location	Sensor1 location	Sensor2 location	Sensor3 location
idle	Real	43.8	44.3	44.9	42.2
	Computed	43.9	44.0	43.9	43.5
cachebench #1	Real	100.0	89.5	91.5	89.0
	Computed	100.6	89.3	90.2	88.5
cachebench #2	Real	89.0	100.0	89.0	90.3
	Computed	88.8	100.5	87.1	90.4
aobench	Real	88.2	86.0	100.0	86.0
	Computed	89.8	86.1	100.2	87.6
phpbench	Real	87.0	84.0	100.0	85.2
	Computed	90.7	86.7	103.6	88.4
cyclictest	Real	99.9	98.2	99.8	98.0
	Computed	101.6	101.5	100.8	101.6
compress-7zip	Real	99.9	99.5	98.3	100.0
	Computed	99.4	99.6	98.8	99.1

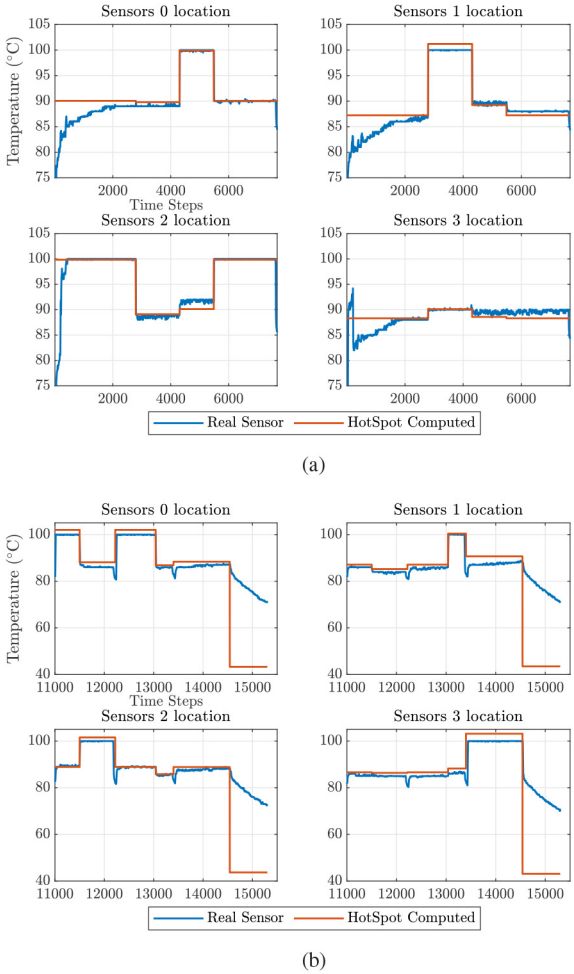


Fig. 20. Comparison between the HotSpot computed steady state temperature at sensor locations and the real sensor readings for (a) *cachebench* and (b) *phpbench* in a time duration.

in the time axis. We tried to perform the true transient analysis in HotSpot based on the power map stream we obtained. However, we found that the true transient analysis in HotSpot takes a prohibitive amount of time in our test cases and hence is not feasible. Reasons are the large map dimension and relatively long sampling interval. Specifically, HotSpot's typical sampling interval is 3.3 μs, whereas in this work has 0.016 ms (60 Hz). It takes about two minutes to compute one single time step even when shrinking the map area by 50 times. Changing the sampling time (3.3 μs) in HotSpot to match the real sampling time (0.016 ms) does not help the simulation because of the underlying mechanism of HotSpot. However, the steady state computation by HotSpot is much faster. As a result, we perform the *pseudo-transient* analysis in which the temperature series of consecutive steady states in time are computed instead. So in this work, we only present and compare the pseudo-transient analysis results.

For each steady state, power maps corresponding to that steady state are averaged to one map sample and fed into HotSpot. For instance, in Fig. 20(a) the time steps from 2792 to 4308 is one steady state and time steps from 4358 to 5488 is another steady state, and one averaged power map for each steady state is fed to HotSpot. Hence, we can form the temperature series of consecutive steady states in the time axis. One

detail is that we implement the 64 by 64 grid model in HotSpot to compute the thermal maps first and later interpolate them back to the original image dimension. Fig. 20 shows the temperature computations at the sensor locations compared to the real sensor readings under heat sink with respect to *cachebench* and *phpbench* in a certain time duration. *phpbench* is followed by idle from the time step 14 540. The maximum absolute error for steady states only for *cachebench* is 1.98 °C, and for *phpbench* is 3.7 °C. The average absolute errors over the time axis across four sensor locations are 0.78 °C and 2.5 °C for these two benchmarks, respectively. Note that *phpbench* has the worst error among the workloads we tested. The results from HotSpot and COMSOL are quite similar.

We remark that the error is calculated based on the measured temperature at the sensor locations because this is the only measured information we can have. But we believe the estimated errors based on the sensor locations are good error indicators for the entire thermal map estimation. As we observe, temperatures track the real trend for all the workloads quite well. One explanation of the error may be due to the leakage difference or the κ error as we discussed in the previous section.

Finally, the average absolute estimation error using COMSOL turns out 2.2 °C for the steady states across all sensor locations over all the workloads, in which only one averaged sample is counted for each steady state. In this way the error won't be biased by the length of idle status or other steady states. We also noticed that for some of the workloads the hottest spot is located away from the sensor locations and can be 3 °C higher than the nearest sensor measurement even though only 2-mm away. Such difference is heat-sink-related. Our measurements also show that when the device runs with back-side cooling and with the processor exposed to air, the underestimated hot spot temperature can be 6 °C–7 °C higher than the nearest sensor. The thermal spec power of the processor in our study is 15W, however, the difference between the hottest spot and sensor may reach a higher value as processor power goes higher. This observation raises the importance of complete heat maps rather than thermal sensor values of limited locations.

VII. Conclusion and Future Work

In this article, we have proposed accurate full chip steady-state power density and thermal map estimation methods for commercial multicore microprocessors operating under normal conditions with heat sink cooling. The proposed thermal to power map recovery method, based on the first principle of heat transfer, is very efficient and fast, which in contrast with existing nonlinear optimization-based methods. Once accurate power density map was estimated and validated with FEM thermal models for back cooling using the IR thermography setup, we proposed a method to build accurate thermal models for commercial processors under heat sink cooling. The methodology is validated by the on-chip sensor reading of the chip once their exact locations are estimated. Experimental results on a Intel i7-8650U 4-core processor with back side cooling have shown that average absolute error of the computed thermal maps compared to the measured thermal maps is around 1.3 °C. Furthermore, the computed thermal maps and the measured thermal maps have 96% similarity (2-D correlation). In addition, we have proposed a novel full-chip steady-state thermal map estimation and characterization approach for the commercial multicore processors when they run under heat sink cooling, which cannot be directly measured by thermal imaging systems. Additionally, experimental results show 2.2 °C average absolute error over a 56 degrees temperature range under heat sink, which indicates about 3.9% error between the computed heat maps and the real thermal maps. Furthermore, the proposed power map estimation method is at least 100× faster and higher resolution than a state-of-art BPI method.

Regarding the future work, the proposed method can actually handle transient powers if the input thermal maps become time series maps. The resulting power maps will be time series maps. For different workloads, we need to characterize them in the back-cooling situations via thermal imaging system as we did in this work first. Then we can compute their thermal maps. However, for the real-time and on-line applications, we cannot do the off-line thermal-imaging characterization for each individual workload. Machine-learning-based power map estimation can be applied, such as using deep neural networks (DNNs) as the methods demonstrated in our machine-learning-based full-chip thermal map estimation method [29], [31].

References

[1] H. Esmaeilzadeh, E. Blem, R. S. Amant, K. Sankaralingam, and D. Burger, "Dark silicon and the end of multicore scaling," *IEEE Micro*, vol. 32, no. 3, pp. 122–134, May/Jun. 2012.

[2] M. Taylor, "A landscape of the new dark silicon design regime," in *Proc. IEEE/ACM Int. Symp. Microarchit.*, vol. 33, Oct. 2013, pp. 8–19.

[3] D. Brooks and M. Martonosi, "Dynamic thermal management for high-performance microprocessors," in *Proc. IEEE Int. Symp. High-Perform. Comput. Archit. (HPCA)*, Jan. 2001, pp. 171–182.

[4] V. Hanumaiah and S. Vrudhula, "Energy-efficient operation of multicore processors by DVFS, task migration, and active cooling," *IEEE Trans. Comput.*, vol. 63, no. 2, pp. 349–360, Feb. 2014.

[5] Z. Liu, S. X.-D. Tan, X. Huang, and H. Wang, "Task migrations for distributed thermal management considering transient effects," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 2, pp. 397–401, Feb. 2015.

[6] H. Wang *et al.*, "Hierarchical dynamic thermal management method for high-performance many-core microprocessors," *ACM Trans. Design Autom. Electron. Syst.*, vol. 22, no. 1, pp. 1–21, Jul. 2016.

[7] R. Joseph and M. Martonosi, "Run-time power estimation in high-performance microprocessors," in *Proc. Int. Symp. Low Power Electron. Design (ISLPED)*, 2001, pp. 135–140.

[8] C. Isci and M. Martonosi, "Runtime power monitoring in high-end processors: Methodology and empirical data," in *Proc. MICRO*, 2003, pp. 93–104.

[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 Autom. Electron. Syst.*, vol. 12, no. 3, pp. 1–29, 2007.

[10] K. Dev, A. N. Nowroz, and S. Reda, "Power mapping and modeling of multi-core processors," in *Proc. Int. Symp. Low Power Electron. Design (ISLPED)*, Sep. 2013, pp. 39–44.

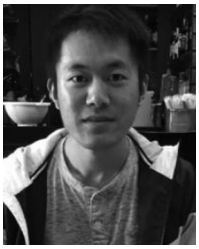
[11] X. Wang, S. Farsiu, P. Milanfar, and A. Shakouri, "Power trace: An efficient method for extracting the power dissipation profile in an IC chip from its temperature map," *IEEE Trans. Compon. Packag. Technol.*, vol. 32, no. 2, pp. 309–316, Jun. 2009.

[12] R. Cochran, A. N. Nowroz, and S. Reda, "Post-silicon power characterization using thermal infrared emissions," in *Proc. Int. Symp. Low Power Electron. Design (ISLPED)*, New York, NY, USA, 2010, pp. 331–336.

[13] S. Paek, W. Shin, J. Sim, and L.-S. Kim, "PowerField: A probabilistic approach for temperature-to-power conversion based on Markov random field theory," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 10, pp. 1509–1519, Oct. 2013.

[14] A. Nowroz, G. Woods, and S. Reda, "Power mapping of integrated circuits using AC-based thermography," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 8, pp. 1398–1409, Aug. 2013.

[15] F. Beneventi, A. Bartolini, P. Vivet, and L. Benini, "Thermal analysis and interpolation techniques for a logic+ WIDEIO stacked DRAM test chip," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 35, no. 4, pp. 623–636, Apr. 2016.
[16] S. Reda, K. Dev, and A. Belouchrani, "Blind identification of thermal models and power sources from thermal measurements," *IEEE Sensors J.*, vol. 18, no. 2, pp. 680–691, Jan. 2018.
[17] S. Sadiqbatcha, H. Zhao, H. Amrouch, J. Henkel, and S. X.-D. Tan, "Hot spot identification and system parameterized thermal modeling for multi-core processors through infrared thermal imaging," in *Proc. Design Autom. Test Europe Conf. Exhibition (DATE)*, Mar. 2019, pp. 48–53.
[18] C. Lasance, H. Vinke, H. Rosten, and K.-L. Weiner, "A novel approach for the thermal characterization of electronic parts," in *Proc. IEEE 11th Annu. Semicond. Thermal Meas. Manag. Symp.*, 1995, pp. 1–9.
[19] Y. C. Gerstenmaier and G. Wachutka, "Rigorous model and network for transient thermal problems," *Microelectron. J.*, vol. 33, pp. 719–725, Sep. 2002.
[20] P. Liu, H. Li, L. Jin, W. Wu, S. X.-D. Tan, and J. yang, "Fast thermal simulation for runtime temperature tracking and management," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 12, pp. 2882–2893, Dec. 2006.
[21] D. Li, S. X.-D. Tan, E. H. Pacheco, and M. Tirumala, "Architecture-level thermal characterization for multi-core microprocessors," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 10, pp. 1495–1507, Oct. 2009.
[22] 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 Integr. (VLSI) Syst.*, vol. 14, no. 5, pp. 501–513, May 2006.
[23] S. P. Gurrum, Y. K. Joshi, W. P. King, K. Ramakrishna, and M. Gall, "A compact approach to on-chip interconnect heat conduction modeling using the finite element method," *J. Electron. Packag.*, vol. 130, no. 3, pp. 1–8, Sep. 2008.
[24] COMSOL Multiphysics, *Heat Transfer Module User's Guide*, COMSOL, Stockholm, Sweden, 2014.
[25] S. Sadiqbatcha, J. Zhang, H. Zhao, H. Amrouch, J. Hankel, and S. X.-D. Tan, "Post-silicon heat-source identification and machine-learning-based thermal modeling using infrared thermal imaging," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 40, no. 4, pp. 694–707, Apr. 2021.
[26] H. Amrouch and J. Henkel, "Lucid infrared thermography of thermally-constrained processors," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design (ISLPED)*, Jul. 2015, pp. 347–352.
[27] WikiChip. *Core i7-8650U—Intel*. Accessed: Feb. 25, 2021. [Online]. Available: https://en.wikichip.org/wiki/intel/core_i7/i7-8650u
[28] Intel. *Intel Performance Counter Monitor (PCM)*. Accessed: Feb. 25, 2021. [Online]. Available: https://software.intel.com/en-us/articles/intel-performance-counter-monitor
[29] S. Sadiqbatcha, Y. Zhao, J. Zhang, H. Amrouch, J. Henkel, and S. X. D. Tan, "Machine learning based online full-chip heatmap estimation," in *Proc. 25th Asia South Pac. Design Autom. Conf. (ASP-DAC)*, 2020, pp. 229–234.
[30] S. Reda and A. Belouchrani. (2018). *Blind Identification of Power Sources in Processors*. [Online]. Available: https://github.com/scale-lab/BPI
[31] S. Sadiqbatcha, J. Zhang, H. Amrouch, and S. X.-D. Tan, "Real-time full-chip thermal tracking: A post-silicon, machine learning perspective," *IEEE Trans. Comput.*, early access, Jun. 2, 2021, doi: 10.1109/TC.2021.3086112.



Jinwei Zhang (Student Member, IEEE) received the B.S. degree in electrical and control engineering from Beijing Institute of Technology, Beijing, China, in 2014, and the M.S. degree in electrical engineering from Washington University in Saint Louis, St. Louis, MO, USA, in 2016. He is currently pursuing the Ph.D. degree with VLSI System and Computation Lab, Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA, USA.

His research interests are aligned with electronic design automation, system level dynamic thermal/reliability management, power modeling and optimization for VLSI, and commercial multicore processors with machine learning techniques.

Sheriff Sadiqbatcha (Graduate Student Member, IEEE) received the B.S. degree (most outstanding graduate of the year) in computer engineering from California State University at Bakersfield, Bakersfield, CA, USA, in 2016, and the M.S. degree (*magna cum laude*) in electrical engineering from the University of California at Riverside, Riverside, CA, USA, in 2019, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering.

He is an Associate Instructor with the Department of Electrical and Computer Engineering, University of California at Riverside. His research interests include presilicon reliability (electromigration), and applied machine-learning in the area of electronic design automation, post-silicon thermal, power, and reliability modeling, and control.

Michael O'Dea is currently pursuing the B.S. degree in computer engineering with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA, USA.

Hussam Amrouch (Member, IEEE) received the Ph.D. degree (*Summa cum laude in distinction*) from the Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany, in 2015.

He is a Junior-Professor heading the Chair of Semiconductor Test and Reliability within the Computer Science, Electrical Engineering Faculty, University of Stuttgart, Stuttgart, Germany, as well as a Research Group Leader with KIT. Then, he founded and led the Dependable Hardware Research Group with KIT. He has over 115 publications (including 45 journal papers) in multidisciplinary research areas across the entire computing stack, starting from semiconductor physics to circuit design all the way up to computer-aided design and computer architecture. His main research interests are design for reliability and testing from device physics to systems, machine learning for CAD, security, approximate computing, and emerging technologies with a special focus on ferroelectric device memories, and negative capacitance transistors.

Dr. Amrouch holds seven HiPEAC Paper Awards and three best paper nominations at top EDA conferences: DAC'16, DAC'17, and DATE'17 for his work on reliability. He currently serves as an Associate Editor of *Integration, the VLSI Journal*. He has served in the technical program committees of many major EDA conferences, such as DAC, ASP-DAC, and ICCAD and a reviewer in many top journals like T-ED, TCAS-I, TVLSI, TCAD, and TC.

Sheldon X.-D. Tan (Senior Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from Fudan University, Shanghai, China, in 1992 and 1995, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Iowa, Iowa City, IA, USA, in 1999.

He is a Professor with the Department of Electrical Engineering, University of California at Riverside, Riverside, CA, USA, where he is also a Cooperative Faculty Member with the Department of Computer Science and Engineering. He was a Visiting Professor with Kyoto University, Kyoto, Japan. He has published more than 320 technical papers and has coauthored six books on those areas. His research interests include machine and deep learning for VLSI reliability modeling and optimization at circuit and system levels, machine learning for circuit and thermal simulation, thermal modeling, optimization and dynamic thermal management for many-core processors, parallel computing and adiabatic and Ising computing based on GPU, and multicore systems.

Prof. Tan received the NSF CAREER Award in 2004, the Best Paper Awards from ICSICT'18, ASICON'17, ICCD'07, and DAC'09. He also received the Honorable Mention Best Paper Award from SMACD'18. He is serving as the TPC Chair for ASPDAC 2021 and the TPC Vice Chair for ASPDAC 2020. He is serving or served as an Editor in Chief for *Integration, the VLSI Journal* (Elsevier), an Associate Editor for three journals: IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, ACM TRANSACTION ON DESIGN AUTOMATION OF ELECTRONIC SYSTEMS, and *Microelectronics Reliability* (Elsevier). He is a JSPS Fellow from December 2017 to January 2018