Copyright © 2013 American Scientific Publishers
All rights reserved
Printed in the United States of America

Journal of Low Power Electronics
Vol. 9, 1–11, 2013

Adaptive Thermal Monitoring of Deep-Submicron CMOS VLSI Circuits

Amir Zjajo*, Nick van der Meijs, and Rene van Leuken

Circuits and Systems Group, Delft University of Technology, Mekelweg 4, 2628 CD, Delft, The Netherlands

(Received: 12 July 2013; Accepted: 10 September 2013)

In integrated circuits accurate runtime sensing of on-chip temperature is required to establish efficient dynamic thermal management techniques. In this paper, we propose novel sensor allocation and placement algorithm and thermal sensing technique for indirect temperature estimation at arbitrary locations. As the experimental results indicate, the runtime thermal estimation method reduces temperature estimation errors by an order of magnitude.

Keywords: Integrated Circuits, Temperature Sensors, Simulation, Thermal Analysis, Thermal Management.

1. INTRODUCTION

The magnitude of thermal gradients and associated thermo-mechanical stress increase further as VLSI designs move into nanometer processes and multi-GHz frequencies.[1] Higher temperature increases the risk of damaging the devices and interconnects since major back-end and front-end reliability issues including electro-migration, time-dependent dielectric breakdown, and negative-bias temperature instability have strong dependence on temperature. Additionally, low power techniques such as dynamic power management, clock gating, voltage islands, dual V_{DD}/V_T and power gating may cause significant on-chip thermal gradients and local hot spots due to different clock/power gating activities and varying voltage scaling. As a consequence, continuous thermal monitoring is necessary to reduce thermal damage and increase reliability. Built-in temperature sensors predict excessive junction temperatures as well as the average temperature of a die within design specifications. In order to maximize the coverage, the thermal sensing devices are scattered across the entire chip to meet the high-level die temperature control requirements. This trend of multiple monitoring circuits is evident in recent processors such as the POWER5, CELL, Itanium, and Opteron processors.[1–4] The sensors are networked by an underlying infrastructure, which provides the bias currents to the sensing devices, collects measurements, and performs analog to digital signal conversion. Therefore, the supporting infrastructure is an on-chip element at a global scale, growing in complexity with each emerging processor design. It needs to span a large distance covering the entire processor core, networking an increasing number of devices. The temperature sensors for thermal monitoring of VLSI circuits should meet several requirements including compatibility with the target process with no additional fabrication steps, high accuracy, a small silicon area and low power consumption to reduce the error caused by self-heating. Temperature sensor based on time-to-digital-converter[5] is constrained by the large area and power overhead at the required sampling rate. Temperature sensor operating in the sub-threshold region[6] is prone to dynamic variations as thermal sensitivity increases by an order of magnitude when operating in sub-threshold.[7] Consequently, the majority of CMOS temperature sensors are based on the temperature characteristics of parasitic bipolar transistors.[8] Although modern temperature sensors achieve high level of accuracy,[9] the placement of these sensors in is constrained to areas where there is enough spatial slack. Additionally, underlying chip power density is highly random due to unpredictable workload, fabrication randomness and non-linear dependence between temperature and circuit parameters. Increasing the number of sensors could possible resolve this issue; nevertheless the cost of adding a large number of sensors is prohibitive. Moreover, even without considering the cost of added sensors, other limitations such as additional channels for routing and input/output may not allow placement of thermal sensors at the locations of interest.

*Author to whom correspondence should be addressed.
Email: amir.zjajo@ieee.org

 1546-1998/2013/9/001/011 doi:10.1166/jolpe.2013.1279

Several techniques have been proposed to solve the problem of tracking the entire thermal profile based on only a few limited sensor observations.[10–16] Among these techniques, the Kalman filter based methods are especially resourceful as such methods are capable of exploiting the statistical properties of power consumption along with sensor observations to estimate temperatures at all chip locations during runtime, while simultaneously retaining the possibility to incorporate associated sensor noise caused by fabrication variability, supply voltage fluctuation, cross coupling etc. However, existing Kalman filter based approaches imply a linear model ignoring the nonlinear temperature-circuit parameters dependency or employ a linear approximation of the system around the operating point at each time instant. These approximations, however, can introduce large errors in the true posterior mean and covariance of the transformed (Gaussian) random variable, which may lead to sub-optimal performance and sometimes divergence of the filter.

In this paper, with the unscented Kalman filter, we explicitly account for the nonlinear temperature-circuit parameters dependency. Since we are considering the spread of random variable, the technique tends to be more accurate than Taylor series linearization employed in existing Kalman filter based approaches. As the experimental results indicate, the runtime thermal estimation method reduces temperature estimation errors by an order of magnitude. Additionally, we propose a systematic optimization technique for thermal sensor allocation and placement based on the cutting plane method.

This paper is organized as follows: Section 2 focuses on the design of compact, low power temperature sensor[9] with high accuracy and wide temperature range. Section 3 introduces optimization algorithm for optimum temperature sensor placement. In Section 4, thermal conduction in integrated circuits and associated temperature estimation method is described. Section 5 elaborates experimental results. Finally, Section 6 provides a summary and the main conclusions.

2. TEMPERATURE SENSOR

To convert temperature to a digital value, a well-defined temperature-dependent signal and a temperature-independent reference signal are required. These quantities can be derived utilizing exponential characteristics of bipolar devices for both negative- and positive temperature coefficient.[17] For constant collector current, base-emitter voltage V_{be} of the bipolar transistors has negative temperature dependence around room temperature. This negative temperature dependence is cancelled by a proportional-to-absolute temperature dependence of the amplified difference of two base-emitter junctions. These junctions are biased at fixed but at unequal current densities resulting in the relation directly proportional to the absolute temperature. This proportionality is, however, rather small (0.1–0.25 mV/°C) and needs to be amplified to allow further signal processing.

2.1. Non-Idealities of Bipolar Transistor

In CMOS process, both lateral and vertical (substrate) *pnp* transistors can be used as temperature sensing devices. The lateral transistors, however, have low current gains and their exponential current voltage characteristic is limited to a narrow range of currents.[8] The substrate transistors have reasonable current gains and high output resistance, but their main limitation is the series base resistance, which can be high due to the large lateral dimensions between the base contact and the effective emitter region. To minimize errors due to this base resistance, we limited the maximum collector currents through the transistors to μA level. The slope of the base-emitter voltage V_{be} of the bipolar transistors depends on process parameters and the absolute value of the collector current. To obtain an overall accuracy of ± 1 °C a maximum spread of V_{be} is limited to 900 μV level and the maximum random voltage error in ΔV_{be} to 60 μV. This spread is PTAT in nature and can be mitigated by trimming, albeit at the expense of increased manufacturing costs. In this way, a single-point trim is enough to compensate for process spread. Since intra-batch spread is usually significantly less than inter-batch spread, batch calibration offers a cheaper alternative to individual trimming, at the expense of lower accuracy.

2.2. Circuit Implementation

The proposed temperature sensor is illustrated in Figure 1. The right part of this circuit, comprising a voltage comparator, (transistors T_{13-21}) creates the output signal of the temperature sensor. The rest of this circuit consists of the temperature sensing-circuit, amplifier, and start-up. The input of the comparator consists of a differential source-coupled stage, followed by two amplifying stages and one digital inverter. To enable a certain temperature detection, voltage comparator require two signals with different temperature dependence; an increasing PTAT voltage V_{int} across the resistor network $N_T R$ (Fig. 2) and decreasing PTAT voltage V_{inr} at the comparator positive input generates temperature decisions (Fig. 3). The resistors are formed by $p+$ poly resistances, which have minimum process variation and temperature coefficient in the given foundry's CMOS process.

The (nominally) zero temperature coefficient is exploited for a temperature-independent bangap-reference generation. In a bandgap voltage reference (Fig. 4), an amplified version of ΔV_{be} is added to V_{be} to yield a temperature-independent reference voltage V_{ref}. The negative voltage-temperature gradient of the base-emitter junction of the transistor Q_1 is compensated by a PTAT voltage across the resistor R_1, thereby creating an almost constant reference voltage V_{ref}. The bandgap reference voltage is obtained at the output of the amplifier (rather than at its

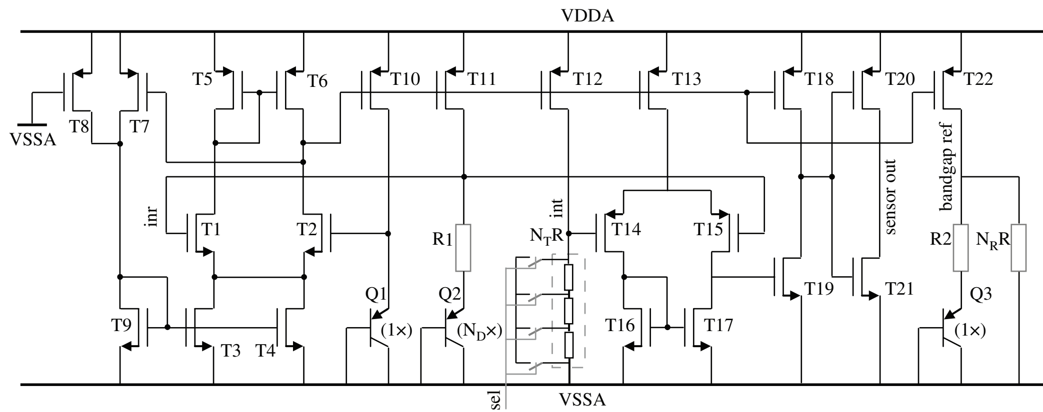

Fig. 1. Temperature sensor[10]-schematic view.

input). The PTAT voltage is firstly converted into current through transistor T_{22} and then summed up to lower reference voltage through resistor R_2. However, the curvature of $V_{\rm be}$ (of transistors $Q_{1\text{–}2}$) will also be present in the bandgap reference voltage.[17] For a current, which is independent of temperature, the curvature correction is in the same order of magnitude of mismatch. The first-order temperature compensation of bandgap reference voltage involves the cancellation of the temperature term by using the PTAT voltage. The second-order temperature compensation is curvature-compensated by adjusting the proportional-to-absolute temperature-type spread on $V_{\rm be}$ of a transistor Q_3 with adjustable resistors N_RR. In essence, based on the ratio of the resistors N_RR and R_2, the $V_{\rm be}$ of a junction with a constant current is subtracted with the $V_{\rm be}$ of a junction with the PTAT current. To accurately define this ratio, adjustable resistors N_RR are constructed of identical unit resistors. The amplifier ($T_{1\text{–}6}$) consists of a non-cascoded OTA with positive feedback to increase the loop-gain. The amplifier output voltage is relatively independent of the supply voltage as its open-loop gain is sufficiently high. Due to the asymmetries, the inaccuracy of the circuit is mainly determined by the offset and flicker noise of the amplifier, which directly adds to $\Delta V_{\rm be}$. Several dynamic compensation techniques such as auto-zeroing, chopping or dynamic element matching[18] might be employed to decrease offset and flicker noise. However, inherently, such techniques require very fast amplifier, whose noise is typically several order of magnitude larger and consumes considerably more power. Furthermore, chopping increases circuit complexity and adds switching noise due to e.g., charge dump and clock interference. Such characteristics make these techniques unsuitable for thermal monitoring of VLSI circuits. In this design, to lower the effect of offset to meet ± 1 °C accuracy, the systematic offset is minimized by adjusting transistor dimensions and bias current in the ratio, while the random offset is reduced by a symmetrical and compact layout. Additionally, the collector currents of bipolar transistors Q_1 and Q_2 are rationed by a pre-defined factor, e.g., transistors are multiple parallel connections of unit devices. The amplifier has sufficient gain to equalize its input voltages. Since these nodes are

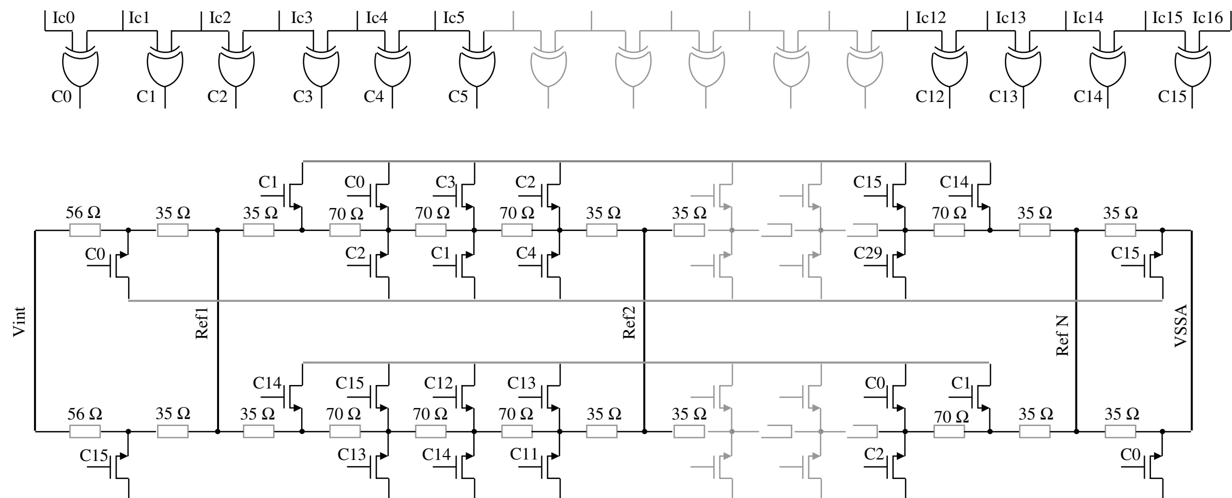

Fig. 2. N_TR resistive network—schematic view.

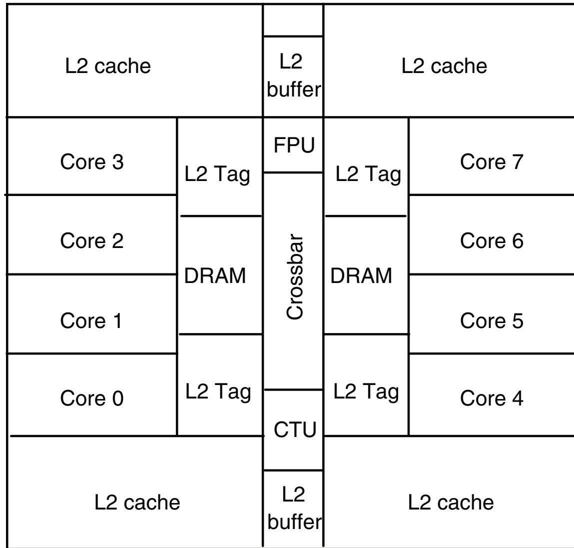

Fig. 3. UltraSparc T_1 architecture floorplan.

the same, the currents from these nodes to ground must be the same as well. The current through R_1 is therefore PTAT (this current is also flowing through the output transistor T_{22}). A start-up circuit consisting of transistors $T_{7–9}$ drives the circuit out of the degenerate bias point when the supply is tuned on. The diode-connected device T_9 provides a current path from the supply through T_7 to ground upon start-up.

The scan chain delivers a four-bit thermometer code for the selection of the resistor value $N_T R$. As illustrated in Figure 2, the nodes in between each resistor have different voltages depending on their proximity to V_{int}. By using thermometer decoding on the digital signal one specific node can be selected as the correct analog voltage. The number of resistor elements determines the resolution of the resistor-network; an n-bit network requires a ladder with 2^n resistors. The resistor-ladder network is inherently monotonic as long as the switching elements are designed correctly. Similarly, since no high-speed operation is required, parasitic capacitors at a tap point will not

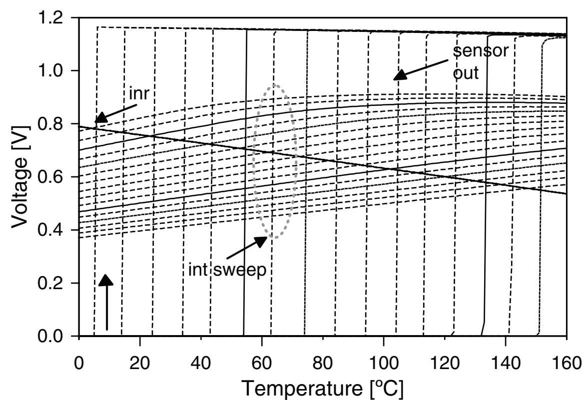

Fig. 4. Sixteen selection levels in the temperature sensor.

create significant voltage glitch. A limit on resistor-value is set by mismatches of individual resistors, which determine the overall accuracy of the generated reference voltages. Assuming that the resistor values are normally distributed with mean R and standard deviation σ_R, the maximum mismatch σ_R/R allowed for four-bit resolution is ≤ 17.6 percent.[19]

3. TEMPERATURE SENSOR PLACEMENT

Complex integrated circuits with large die area require multiple thermal sensors to capture temperatures at a wide range of locations as the unpredictability of a workload leads to continuous migration of hot spots, and within-die manufacturing variations lead to parameter variability that further conceal the locations of the thermal hot spots. However, the thermal sensors, together with their support circuitry and wiring, complicate the design process and increase the total die area and manufacturing costs. Given the limitations on the number of thermal sensors, it is necessary to optimally place them near potential hot spot locations. In Ref. [20], a clustering algorithm is described that computes the thermal sensor positions that best serve clusters of potential hot spot locations. In Ref. [21], an optimal sensor problem is computed as the unite-covering problem. In Ref. [22], the unknown temperature at a particular location is computed as a weighted combination of the known measurements at other locations. Nevertheless, these techniques may be ineffective if the accuracy or availability of sensors measurements is in question. The size of the grid improves the effectiveness of the sensor infrastructure in many cases; however, in others, the hotspots may simply be located such that even a sizable grid of sensors will be incapable of capturing the locations of significant thermal events. In Ref. [23], the maximum distance from the hotspot within which a sensor can be placed is based on the assumption that the temperature decays exponentially from a hotspot neglecting the effect of the location and power consumptions of other power sources on the temperature around a hotspot. In Ref. [24], a systematic technique for thermal sensor allocation and placement in microprocessors is introduced, which identifies an optimal physical location for each sensor such that steep thermal gradient is maximized. Nevertheless, this approach does not consider the accuracy of the sensors and does not guarantee the maximum error in the thermal sensor readings.

To minimize these errors, we have developed off-line optimization technique based on cutting plane method.[25] We find the optimum number of sensors and their locations such that there is at least one sensor in the observable set of each point of interest. This guarantees that the accuracy requirements are satisfied since any sensor placed on a grid cell in the observable set of a point of interest can sense temperature with the required accuracy. The optimization problem, given r iterations, is than formulated as to find a set of potential sensor points L that minimizes the error

E in the thermal sensor readings over the set of n points of interests Q in the design space Φ, such that there will be at least one sensor in observable set of hotspot given bound β:

$$
\begin{aligned}
L = \underset{Q \in \Phi(E)}{\arg\min}\ E(Q) \\
\text{subject to } E_r(Q_r) \geq 1-\beta \quad \forall Q \in \Phi(E_r)
\end{aligned} \tag{1}
$$

where $Q = \{q_1, q_2, \ldots, q_n\}$ and $E = \{e_1, e_2, \ldots, e_n\}$ are the set of n points of interests and the set of corresponding desired accuracies for these points, respectively. A set of potential sensor points $L = \{l_1, l_2, \ldots, l_k\}$ consists of all of the grid cells around the hotspots where a temperature sensor can be placed. Let $\Phi(E)$ be the compact set of all valid design variable vectors Q such that $E(Q) = E$. That Φ is assumed to be compact is, for all practical purposes, no real restriction when the problem has a finite minimum. If, as an approximation, we restrict $\Phi(E_r)$ to just the one-best derivation of E_r, then we obtain the structured perceptron algorithm.[26] As a consequence, given active constraints, (1) can be effectively solved by a sequence of minimizations of the feasible region with iteratively-generated low-dimensional subspaces.

3.1. Optimization Problem

To start the optimization problem, a design metric for global solution is initially selected, based on the priority given to the accuracy of the points of interests as opposed to the performance function. In the algorithm, we use a cutting plane method[25] to repeatedly recomputed optimum L with a precision of at least ε and add it to a working set D_r of derivations on which (1) is optimized. A new L is added to the working set only if $L > \varepsilon$; otherwise, the algorithm terminates, e.g., we are cutting out the half-space because we know that all such points have an objective value larger than ε, hence can not be optimal. The algorithm solves (1) restricted to D_r by sequential minimal optimization,[27] in which we repeatedly select a pair of derivatives of Q and optimize their dual (Lagrange) variables, required to find the local maxima and minima of the performance function. Although sequential minimal optimization algorithm is guaranteed to converge, we used the heuristics suggested by Ref. [28] to accelerate the rate of convergence and to select feasibility region: one must violate one of the conditions, and the other must allow the objective to be improved. At the end of sequence, we average all the weight vectors obtained at each iteration, just as in the averaged perceptron.

3.2. Parameter Update

To insure that the data is completely separable, we employ stochastic steepest gradient descent method to adapt the parameters. We map design variable vector Q to feature vectors $h(Q)$, together with a vector of feature weights w, which defines contribution of design variable in obtained yield. Updating feature weights is presented as a quadratic program

$$
\begin{aligned}
&\text{minimize } 1/2\eta \|w' - w\|^2 \\
&\text{subject to } E_r(Q_r) \geq 1-\beta \quad \forall Q \in \Phi(E_r)
\end{aligned} \tag{2}
$$

where η is a step size. The quadratic programming problem is solved incrementally, covering all the subsets of classes constructing the optimal separating hyperplane for the full data set. If no hyperplane can be found that can divide the *a priori* and *a posteriori* classes, with the modified maximum margin technique[29] we find a hyperplane that separates the training set with a minimal number of errors.

4. ADAPTIVE RUNTIME THERMAL TRACKING

4.1. Thermal Model

The thermal behavior of complex deep-submicron VLSI circuits is affected by various factors, such application dependent localized heating. In addition, process variations impact the total power consumption (by largely affecting the leakage component) and, hence, the temperature behavior of each chip, generating different thermal profiles. Power management techniques, such as local clock gating, further create a disparity in power densities among different regions on a chip. As a consequence, continuous thermal monitoring is necessary to reduce thermal damage and increase reliability. To model the thermal properties of the deep-submicron VLSI, we use an off-line temperature profile estimation methodology,[30] which has the capability to include layout geometry of individual circuit blocks in a chip. The model is composed by three types of layers: bulk silicon, active silicon and the heat-spreading copper layer. The chip is partitioned into a mesh according to the information provided by the layout geometry and power distribution map. Nominal power distribution (including switching and leakage power dissipation) for each functional unit according to its activity factor is assigned an initial value. Each functional unit in the floorplan is represented by one or more thermal cells of the silicon layer. Physical parameters such as thermal conductivity and heat transfer coefficient depend on specific packaging material properties and applied cooling techniques. Boundary conditions are determined by the operating environment. The simulator uses layout geometry, power distribution, boundary conditions, and physical thermal parameters as initial values to formulate the system of partial differential equations (PDEs), which are approximated into a system of ordinary differential equations (ODEs) with discontinuous Galerkin method.

The thermal model is slightly nonlinear since coefficients are temperature-dependent (relative error in the order of 0.16%).[31] To represent the thermal model using a linear, time invariant discrete-time system, the solution

of the differential equations modeling the heat flow inside the MPSoC has been linearized. In the sequel we assume that the kth temperature measurement is done at time t_k. The system can be represented with

$$T_{(k)} = AT_{(k-1)} + BP_{(k-1)} + u_{(k-1)}$$
$$S_{(k)} = CT_{(k)} \tag{3}$$

where at time k, $P_{(k)}$ is its input and $S_{(k)}$ is its output. The temperature value of each cell is the state $T \in R^{2n}$. The first n entries represent the cells composing the silicon floorplan and the remaining n entries model the copper layer. The input of the system $P \in R^p$ is the vector of power inputs (heat sources as function of time, wherever they exists). The output $S \in R^s$ is the temperature observed by the s on-chip thermal sensors placed in the silicon layer. Matrices A, B, C and vector u describe the system and model all geometric constraints among each entry of the state vector and its placement on the chip floorplan. Matrix $A \in R^{2n\times 2n}$ expresses the part of the temperature spreading process inside the chip that depends only on the current temperature profile of the cells determined by the circuit parameters. Matrix $B \in R^{2n\times p}$ expresses the temperature increase due to the input. The part of the system dynamic that is not controllable by the input vector such as fabrication variability, supply voltage fluctuation, cross coupling etc. is expressed by vector $u \in R^{2n}$. Matrix $C \in B^{s\times 2n}$, $B = \{0, 1\}$, represents a selection matrix that models the placement of a sensor on the silicon die identifies the sensor grid cells at which temperatures are observable. We are assuming that distinct measurements are coming from distinct sensors: C has only one nonzero element per row.

4.2. Temperature Estimation

Several on-line techniques have been proposed to solve the thermal tracking problem.[10–16] Among these techniques Kalman filter (KF) based methods generate thermal estimates for all chip locations while countering sensor noise and can be applied to real-time thermal tracking problems. The KF propagates the mean and covariance of the probability density function of the model state in an optimal (minimum mean square error) way in case of linear dynamic systems. However, as VLSI fabrication technology continues to scale down, leakage power can take up to 50% of the total chip power consumption.[32] Note that leakage has the nonlinear nature that increase exponentially with the chip temperature. As a consequence, the standard Kalman filter tends to under-estimate the actual chip temperature due to the assumed linear model. Consider (3) in corresponding discrete-time state space

$$\begin{aligned} T_{(k)} &= AT_{(k-1)} + B(P_{D(k-1)} + P_{L(k-1)}) + u_{(k-1)} \\ &= AT_{(k-1)} + B1/2\alpha C_L V_{\mathrm{DD}}^2 f + BK_1 T_{(k-1)}^2 \\ &\quad \times e^{K_2/T_{(k-1)}} + u_{(k-1)} = f(T_{(k-1)}) + u_{(k-1)} \\ S_{(k)} &= h(T_{(k)}) + z_{(k)} \end{aligned} \tag{4}$$

where $u_{(k-1)} \sim N(0, R_{(k-1)})$ is the Gaussian process noise, and $z_{(k)} \sim N(0, U(k))$ is the Gaussian sensor noise. For clarity, we subdivided power P into two components, dynamic power $P_{D(k-1)}$ and leakage power $P_{L(k-1)}$. While dynamic power consumption $P_{D(k-1)} = 1/2 \propto C_L V_{\mathrm{DD}}^2 f$, where C_L is switching capacitance, α is switching activity of output node, V_{DD} is supply voltage and f is the operation frequency of system, is weakly coupled with temperature variation, static power consumption is a strong function of temperature $P_{L(k-1)} = K_1 T_{(k-1)}^2 e^{(K_2/T_{(k-1)})}$,[33] where K_1 and K_2 are design/technology and fixed supply voltage constants, respectively. Due to unpredictability of workloads (power vector is unknown until runtime) and fabrication/environmental variabilities, the exact value of $T_{(k)}$ at runtime is difficult to predict. To elevate the issue, on-chip sensors provide an observation vector $S_{(k)}$, which is essentially a subset of $T_{(k)}$ plus sensor noise $z_{(k)}$. In (4), $h(\cdot)$ is a transformation function determined by the sensor placement. Due to the sensors power/area overheads, their number and placement are highly constrained. As a consequence, the problem of tracking the entire thermal profile (vector $T_{(k)}$) based on only a few limited sensor observations $S_{(k)}$ is rather complex.

To extend the model for the nonlinear leakage-temperature function $f(\cdot)$, the most common way of applying the KF is in the form of the extended Kalman filter (EKF). In the EKF, the probability density function is propagated through a linear approximation of the system around the operating point at each time instant. These approximations, however, can introduce large errors in the true posterior mean and covariance of the transformed (Gaussian) random variable, which may lead to sub-optimal performance and sometimes divergence of the filter. The advantage of EKF over the other non-linear filtering methods is its relative simplicity compared to its performance. However, as EKF is based on a local linear approximation, it will have limited applicability in thermal tracking problems with considerable nonlinearities. Also the filtering model is restricted in the sense that only Gaussian noise processes are allowed and thus the model cannot contain, for example, discrete valued random variables. The Gaussian restriction also prevents handling of hierarchical models or other models where significantly non-Gaussian distribution models would be needed. To overcome these limitations, we employ the unscented Kalman filter (UKF).[34, 35] The UKF is using the statistical linearization technique to linearize a nonlinear function of a random variable through linear regression between k data points drawn from *a priori* distribution of the random variable. Since we are considering the spread of random variable, the unscented transform is able to capture the higher order moments caused by the non-linear transform better than the EKF Taylor series based approximations.[34] The mean and covariance of the transformed ensemble can then be computed as the estimate of the nonlinear

transformation of the original distribution. The UKF outperforms the EKF in terms of prediction and estimation error, at an equal computational complexity for general state-space problems.[35] Additionally, the UKF can easily be extended to filter possible power estimation noises, restricting the influence of the high frequency component in power change on the modeling approach.

5. EXPERIMENTAL RESULTS

5.1. Experimental Setup

The chip architecture determines the complexity of processing versus storage versus communication elements and thus the thermal peak of these elements. A chip with complex processing elements (e.g., wide-issue, multi-threaded) will require larger storage elements (e.g., large multi-level caches, register files) as well as sophisticated communication elements (e.g., multi-level, wide buses, networks with wide link channels, deeply-pipelined routers and significant router buffering). On the other extreme, there are chip architectures where processing elements are single ALUs serviced by a few registers at ALU input/output ports, interconnected with simple single-stage routers with little buffering. Application characteristics dictate how these elements are utilized, and hence influencing the thermal profile of the chip. In this paper, as a platform for analyzing the absolute and relative thermal impact of all components of a chip, we use an architecture resembling UltraSparc $T1$ architecture[36] (Fig. 3). The experiments were executed on a 64-bit Linux server with two quad-core Intel Xeon 2.5 Ghz CPUs and 16 GB main memory. Values regarding thermal resistance, silicon thickness, and copper layer thickness have been derived from[36] and its floorplan and power/area distribution ratio of each element from Ref. [37], respectively. BasicMath application from the MiBench benchmark[38] is selected and run on datasets provided by Ref. [39]. Switching activities were obtained utilizing SimpleScalar.[40] The calculation was performed in a numerical computing environment.[41]

5.2. Temperature Sensor Performance

The stand-alone sensor occupies an area of 0.05 mm^2 operates within 1.0 V–1.8 V range and dissipates 11 μW. In the test silicon, four bits for a sixteen selection levels are chosen for the temperature settings, resulting in a temperature range from 0 °C–160 °C in steps of 9 °C, which is sufficient for thermal monitoring of VLSI circuits (Fig. 4). Simulated bandgap reference voltage versus temperature is illustrated in Figure 5. If more steps are required, a selection $N_T R$ can be easily extended with higher resolution resistive network. For the robustness, the circuit is completely balanced and matched both in the layout and in the bias conditions of devices, cancelling all disturbances and non-idealities to the first order. A summary of the sensor performance and comparison with recently published works is shown in Table I. Measurements have

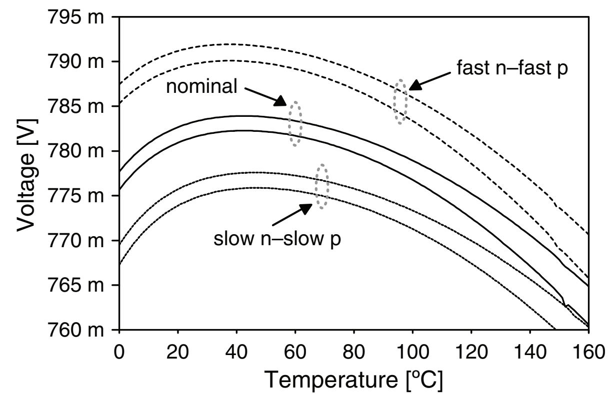

Fig. 5. Bandgap reference voltage: nominal, fast–fast and slow–slow process corners.

been performed on 45 samples from 2 different batches. All chips are functional in a temperature range between 0 °C and 160 °C. The average error at room temperature is around 0.5 °C, with a standard deviation of less than 0.4 °C, which matches the expected error of 0.4 °C within a batch. Non-linearity is approximately 0.4 °C from 0 °C to 160 °C. The intrinsic base-emitter voltage non-linearity in the bandgap reference is limited by compensation circuit. The measured noise level is lower than 0.05 °C.

In all-digital temperature sensors,[5,45] the two-temperature-point calibration is required in every sensor; thus, calibration cost is very large in on-chip thermal sensing applications. A current-output temperature sensor[6] does not have a linear temperature reading and is sensitive to process variation, which requires more effort and cost for after-process calibration. Although the dual-DLL-based temperature sensor[44] only needs one-temperature-point calibration, it occupies large chip area with a high level of power consumption at a microwatt level. The sensors based on the temperature characteristics of parasitic bipolar transistors[42,43] offer high accuracy and small chip area. However, high power consumption in Ref. [42] and small temperature range in Ref. [43] make these realizations unsuitable for on-chip thermal monitoring.

5.3. Temperature Tracking

Based on (4), we simulated the thermal profile of the test processor for a total duration of 600 seconds (the simulation starts at room temperature). This is assumed to be the real chip temperature and is used to measure estimation accuracy. We examine the mean absolute error and the standard deviation of the error as the location of interest. These values are averaged over all the locations of interest. Results of the optimization algorithm in (1) are shown in Table II as measurement errors. Accuracy is limited due to the variable workload and placing restrictions such as additional channels for routing and input/output. We compare the accuracy of our UKF approach to that of the Kalman filter[10] and extended Kalman filter.[11] Due to the inaccuracy of its linear model, the standard Kalman

Table I. Summary of the temperature sensor performance and comparision with prior art.

	[5]	[6]	[42]	[43]	[44]	[45]	[This work]
Range (°C)	0∼100	10∼100	−55∼125	temp switch	0∼100	0∼100	0∼160
Supply voltage (V)	3.0∼3.8	5	2.5∼5.5	1.0∼1.8	1.2	1.0	1.0∼1.8
Inaccuracy (°C)	−0.7∼+0.9	±1	±0.1	±1.1	−1.8∼+2.3	±10.0	±0.9
Sensor type	Temp-to-pulse	Analog current	ΔV_{be}	ΔV_{be}	Duall-DLL	Temp-to-pulse	ΔV_{be}
Calibration	Two-points	–	One-point	–	One-point	Autocalibration	–
Power (μW)	490	300	247	13	12000	55	11
Area (mm^2)	0.175	0.023	0.16	0.03	0.16	0.01	0.05

Table II. Error statistics for limited number of sensors.

	Sensor placing		KF[10]		EKF[11]		UKF	
	Estimation errors (°C)		Estimation errors (°C)		Estimation errors (°C)		Estimation errors (°C)	
# Sensors	Error (μ)	Error (σ)	Error (μ)	Error (σ)	Error (μ)	Error (σ)	Error (μ)	Error (σ)
2	3.06	3.37	2.57	3.43	1.35	2.37	0.38	0.54
3	2.88	3.01	2.65	2.86	1.41	2.44	0.26	0.67
4	2.44	2.72	2.74	2.56	1.38	1.94	0.33	0.94
5	2.18	2.31	2.57	2.34	1.21	1.64	0.26	0.84
6	1.84	2.02	2.24	2.94	1.24	1.86	0.32	0.56

filter relies excessively on the accuracy of sensor input. The temperature estimates derived from the Kalman filter are non-anticipative in the sense that they are only conditional to sensor measurements obtained before and at the time step n. The EKF approximate the nonlinearities with linear or quadratic functions or explicitly approximate the filtering distributions by Gaussian distributions. In UKF, the unscented transform is used for approximating the evolution of Gaussian distribution in non-linear transforms. Figure 6 illustrate that the UKF method always keep track of the actual temperature with high accuracy. For clarity, we only depicted UKF tracking.

High precision of temperature tracking (within 0.4 °C for mean and 1.0 °C for standard deviation) for various cases, ranging from one to six sensors, respectively, placed at an arbitrary location around the hotspot, is shown in Table II. As expected, with increased number of sensors, the measurement error decreases. The UKF obtain almost identical

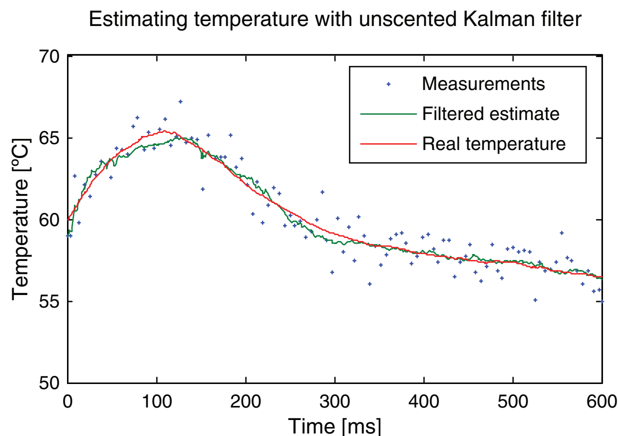

Fig. 6. Sensor measurements, actual and estimated temperatures.

accuracy significantly outperforming KF and EKF, especially when the number of sensors is small. Note that 1 °C accuracy translates to 2 W power savings.[46] The average error in Table III (across all chip locations) of each method is reported as we vary the sensor noise level. As we increase the noise level, the estimation accuracy generated by KF and EKF degrades more rapidly in contrast to UKF, which generate accurate thermal estimates (within 0.7 °C) under all circumstances. The improved performance of the UKF compared to the EKF is due to two factors, namely, the increased time-update accuracy and the improved covariance accuracy. In the UKF case, the covariance estimation is very accurate, which results in a different Kalman gains in the measurement-calibration equation and hence the efficiency of the measurement-calibration step. The EKF also formally requires the measurement model and dynamic model functions to be differentiable. Even when the Jacobian matrices exist and could be computed, the actual computation and programming of Jacobian matrices is error prone and hard to debug. On the other hand, UKF is not based on local linear approximation; UKF utilizes a bit further points in approximating the non-linearity. The computational load increases when moving from the EKF to the UKF if the Jacobians are computed analytically (the average runtime of EKF versus UKF is approximately 12 ms and 16 ms for one measurement, respectively). However, for higher order systems, the Jacobians for the EKF are computed using finite differences. In this case the computational load for the UKF is comparable to the EKF. Effectively, the EKF builds up an approximation to the expected Hessian by taking outer products of the gradient. The UKF, however, provide a more accurate estimate through direct approximation of the expectation of the Hessian. Note that another distinct advantage of the UKF

Table III. Error statistics for different noise settings.

	Sensor placing		KF[10]		EKF[11]		UKF	
	Estimation errors (°C)		Estimation errors (°C)		Estimation errors (°C)		Estimation errors (°C)	
# Sensors	Error (μ)	Error (σ)	Error (μ)	Error (σ)	Error (μ)	Error (σ)	Error (μ)	Error (σ)
2.5	2.53	2.86	2.94	3.36	1.46	2.27	0.26	0.57
5.0	4.94	4.47	3.26	4.24	1.85	2.64	0.33	0.69
7.5	6.63	5.55	5.68	5.97	2.11	2.86	0.47	0.66
10.0	7.86	8.13	6.33	7.65	2.69	3.26	0.46	0.73

occurs when either the architecture or error metric is such that differentiation with respect to the parameters is not easily derived as necessary in the EKF. The UKF effectively evaluates both the Jacobian and Hessian precisely through its sigma point propagation, without the need to perform any analytic differentiation.

6. CONCLUSION

Accurate temperature estimation is one of the foremost steps in the evaluation of successful high-performance system on chip designs. The feasibility of a high accuracy, adaptive temperature sensor has been verified by experimental measurements from the silicon prototype. The stand alone sensor occupies a small silicon area of 0.05 mm^2, operates at 1.0 V–1.8 V supply voltage in a temperature range from 0 °C–160 °C and dissipates only 11 μW. With proposed optimization technique based on cutting plane method, we find the optimum number of sensors and their locations such that there is at least one sensor in the observable set of each point of interest. Furthermore, to improve thermal management efficiency we present methodology based on unscented Kalman filter for accurate temperature estimation at all chip locations while simultaneously countering sensor noise. As the results indicate, the proposed method generates accurate thermal estimates (within 1.0 °C) under all examined circumstances. In comparison with KF and EKF, the UKF consistently achieves a better level of accuracy at limited costs.

References

1. ITRS, International Technology Roadmap for Semiconductors (**2011**).
2. J. Clabes, et al., Design and implementation of the POWER5 microprocessor, *Proceedings of IEEE International Solid-State Circuits Conference* (**2004**), pp. 56–57.
3. D. Pham, et al., The design and implementation of a first generation CELL processor, *Proceedings of IEEE International Solid-State Circuits Conference* (**2005**), pp. 184–185.
4. R. McGowen, et al., Power and temperature control on a 90 nm Itanium-family processor. *IEEE Journal of Solid-State Circuits* 41, 229 (**2006**).
5. J. Dorsey, et al., An integrated quad-core opteron processor, *Proceedings of IEEE International Solid-State Circuits Conference* (**2007**), pp. 102–103.
6. P. Chen, C. Chen, C. Tsai, and W. Lu, A time-to-digital-converter based CMOS smart temperature sensor. *IEEE Journal of Solid-State Circuits* 40, 1642 (**2005**).
7. V. Szekely, C. Marta, Z. Kohari, and M. Rencz, CMOS sensors for on-line thermal monitoring of VLSI circuits. *IEEE Transactions on VLSI Systems* 5, 270 (**1997**).
8. B. Datta and W. Burleson, Temperature effects on energy optimization in sub-threshold circuit design, *Proceedings of IEEE International Symposium on Quality Electronic Design* (**2009**), pp. 680–685.
9. G. C. M. Meijer, G. Wang, and F. Fruett, Temperature sensors and voltage references implemented in CMOS technology. *IEEE Sensors Journal* 1, 225 (**2001**).
10. A. Zjajo, N. van der Meijs, and R. van Leuken, A 11 μW 0 °C–160 °C temperature sensor in 90 nm CMOS for adaptive thermal monitoring of VLSI circuits, *Proceedings of IEEE International Symposium on Circuits and Systems* (**2012**), pp. 2007–2010.
11. Y. Zhang, A. Srivastava, and M. Zahran, Chip level thermal profile estimation using on-chip temperature sensors, *Proceedings of IEEE International Conference on Computer Design* (**2008**), pp. 432–437.
12. S. Sharifi, C.-C. Liu, and T. S. Rosing, Accurate temperature estimation for efficient thermal management, *Proceedings of IEEE International Symposium on Quality Electronic Design* (**2008**), pp. 137–142.
13. R. Cochran and S. Reda, Spectral techniques for high resolution thermal characterization with limited sensor data, *Proceedings of IEEE Design Automation Conference* (**2009**), pp. 478–483.
14. F. Zanini, D. Atienza, C. N. Jones, and G. De Micheli, Temperature sensor placement in thermal management systems for MPSoCs, *Proceedings of IEEE International Symposium on Circuits and Systems* (**2010**), pp. 1065-1068.
15. H. Jung and M. Pedram, A stochastic local hot spot alerting technique, *Proceedings of the IEEE Asia and South Pacific Design Automation Conference* (**2008**), pp. 468–473.
16. S. Sharifi and T. S. Rosing, Accurate direct and indirect on-chip temperature sensing for efficient dynamic thermal management. *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems* 29, 1586 (**2010**).
17. Y. Zhang and A. Srivastava, Adaptive and autonomous thermal tracking for high performance computing systems, *Proceedings of IEEE Design Automation Conference* (**2010**), pp. 68–73.
18. F. Fruett, G. C. M. Meijer, and A. Bakker, Minimization of the mechanical-stress-induced inaccuracy in bandgap voltage references. *IEEE Journal of Solid-State Circuits* 38, 1288 (**2003**).
19. A. Bakker and J. H. Huijsing, A low-cost high-accuracy CMOS smart temperature sensor, *Proceedings of IEEE European Solid-State Circuit Conference* (**1999**), pp. 302–305.
20. J. Doernberg, P. R. Gray, and D. A. Hodges, A 10-bit 5-M sample/s CMOS two-step flash ADC. *IEEE Journal of Solid-State Circuits* 24, 241 (**1989**).
21. S. O. Memik, R. Mukherjee, M. Ni, and J. Long, Optimizing thermal sensor allocation for microprocessors. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 27, 516 (**2008**).
22. B.-H. Lee and T. Kim, Optimal allocation and placement of thermal sensors for reconfigurable systems and its practical extension, *Proceedings of IEEE Asia and South Pacific Design Automation Conference* (**2008**), pp. 703–707.

23. F. Liu, A general framework for spatial correlation modeling in VLSI design, *Proceedings of IEEE Design Automation Conference* (**2007**), pp. 817–822.
24. R. Mukherjee and S. O. Memik, Systematic temperature sensor allocation and placement for microprocessors, *Proceedings of IEEE Design Automation Conference* (**2006**), pp. 542–547.
25. K.-J. Lee and K. Skadron, Analytical model for sensor placement on microprocessors, *Proceedings of IEEE International Conference on Computer Design* (**2005**), pp. 24–27.
26. I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, Support vector machine learning for interdependent and structured output spaces, *Proceedings of International Conference on Machine Learning* (**2004**), pp. 1–8.
27. Y. Freund and R. E. Schapire, Large margin classification using the perceptron algorithm. *Machine Learning* 37, 277 (**1999**).
28. J. C. Platt, Fast training of support vector machines using sequential minimal optimization, Advances in Kernel Methods: Support Vector Learning, edited by B. Scholkopf, C. J. C. Burges, and A. J. Smola, MIT Press (**1998**), pp. 195–208.
29. B. Taskar, Learning structured prediction models: A large margin approach, Ph.D. Thesis, Stanford University (**2004**).
30. V. Franc and V. Hlavac, Multi-class support vector machine, *Proceedings of IEEE International Conference on Pattern Recognition* (**2002**), Vol. 2, pp. 236–239.
31. A. Zjajo, N. van der Meijs, and R. van Leuken, Thermal analysis of 3D integrated circuits based on discontinuous Galerkin finite element method, *Proceedings of IEEE International Symposium on Quality Electronic Design* (**2012**), pp. 117–122.
32. G. Paci, et al., Exploring temperature-aware design in low-power MPSoCs, *Proceedings of IEEE Design, Automation in Europe Conference* (**2006**), pp. 838–843.
33. N. Kim, et al., Leakage current: Moore's law meets static power. *IEEE Computer* 36, 68 (**2003**).
34. L. He, W. Liao, and M. Stan, System level leakage reduction considering the interdependence of temperature and leakage, *Proceedings of IEEE/ACM Design Automation Conference* (**2004**), pp. 12–17.
35. S. J. Julier and J. K. Uhlmann, Unscented filtering and nonlinear estimation, *Proceedings of IEEE* 92, 401 (**2004**).
36. R. van der Merwe and E. A. Wan, The square-root unscented Kalman filter for state and parameter-estimation, *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing* (**2001**), pp. 3461–3464.
37. A. Leon, K. Tam, J. Shin, D. Weisner, and F. Schumacher, A power efficient high-throughput 32-thread SPARC processor, *Proceedings of IEEE International Solid-State Circuits Conference* (**2006**), pp. 295–304.
38. A. K. Coskun, T. S. Rosing, and K. Whisnant, Temperature aware task scheduling in MPSoCs, *Proceedings of IEEE Design, Automation and Test in Eurupe Conference* (**2007**), pp. 1–6.
39. MiBench, http://www. eecs. umich.edu/mibench/.
40. G. Fursin, J. Cavazos, M. O'Boyle, and O. Temam, MiDataSets: Creating the conditions for a more realistic evaluation of iterative optimization, *Proceedings of International Conference on High-Performance and Embedded Architectures and Compilers* (**2007**), pp. 245–260.
41. SimpleScalar, http://www. simplescalar.com/.
42. MatLab, http://www. mathworks.com/.
43. A. P. Pertijs, K. A. A. Makinwa, and J. H. Huijsing, A CMOS smart temperature sensor with a 3σ inaccuracy of ± 0.1 °C from -55 °C to 125 °C. *IEEE Journal of Solid-State Circuits* 40, 2805 (**2005**).
44. D. Schinkel, R. P. de Boer, A. J. Annema, and A. J. M. van Tuijl, A 1-V 15 μW high-precision temperature switch, *Proceedings of IEEE European Solid-State Circuit Conference* (**2001**), pp. 77–80.
45. K. Woo, S. Meninger, T. Xanthopoulos, E. Crain, D. Ha, and D. Ham, Dual-DLL-based CMOS all-digital temperatrue sensor for microprocessor thermal monitoring, *Proceedings of IEEE International Solid-State Circuit Conference* (**2009**), pp. 68–70.
46. C.-C. Chung and C.-R. Yang, An all-digital smart temperature sensor with auto-calibration in 65 nm CMOS technology, *Proceedings of IEEE International Symposium on Circuits and Systems* (**2010**), pp. 4089–4092.
47. E. Rotem, J. Hermerding, C. Aviad, and C. Harel, Temperature measurement in the Intel Core Duo Processor, *Proceedings of IEEE International Workshop on Thermal Investigations of ICs* (**2006**), pp. 23–27.

Amir Zjajo

Amir Zjajo *received the M.Sc. and DIC degrees from the Imperial College London, London, U.K., in 2000 and the Ph.D. degree from Eindhoven University of Technology, Eindhoven, The Netherlands in 2010, all in electrical engineering. In 2000, he joined Philips Research Laboratories as a member of the research staff in the Mixed-Signal Circuits and Systems Group. From 2006 until 2009, he was with Corporate Research of NXP Semiconductors as a senior research scientist. In 2009, he joined Delft University of Technology as a Faculty member in the Circuit and Systems Group. Dr. Zjajo has published more than 60 papers in referenced journals and conference proceedings, and holds more then 10 US patents or patents pending. He is the author of the books Low-Voltage High-Resolution A/D Converters: Design and Calibration (Springer, 2011, Chinese translation, 2013) and Stochastic Process Variations in Deep-Submicron CMOS: Circuits and Algorithms (Springer, in press). He serves as a member of Technical Program Committee of IEEE Design, Automation and Test in Europe Conference, IEEE International Symposium on Circuits and Systems and IEEE International Mixed-Signal Circuits, Sensors and Systems Workshop. His research interests include mixed-signal circuit design, signal integrity and timing and yield optimization of VLSI.*

Nick van der Meijs

Nick van der Meijs *received M.Sc. and Ph.D. degrees from Delft University of Technology in 1985 and 1992 respectively. Currently, he is associate professor at Delft University of Technology in Delft, the Netherlands in the Circuits and Systems group of the Department of Micro Electronics and Computer Engineering. He has (co-)authored some 100 papers on various topics including design frameworks, interconnect optimization and parasitics modeling, and was one of the lead developers of the SPACE 2D and 3D parasitic layout to circuit extractor. He and his research group currently work both on modeling of parasitic effects in advanced integrated circuits and on circuit level design methods and tools for dealing with variability. He is a regular reviewer for various EDA and design methodology conferences and journals, and has served as topic chair on multiple at conferences. As a Director of Studies he is responsible for the content, organization and quality of the B.Sc. and M.Sc. curricula in Electrical Engineering and Computer Engineering at TU Delft.*

Rene van Leuken

Rene van Leuken *received the Ph.D. degree in electrical engineering from the Delft University of Technology in 1988. At the moment he is a professor at the Circuit and Systems group at the same institution. His current research interests include high level system design, design automation, system design optimization and DSP engines. Over the years he has been involved in many research projects: ESPRIT, FP6, FP7, JESSI, MEDEA, and recently in MEDEA+ and ENIAC/CATRENE projects. He is member of the PATMOS program committee and has published papers in all major conferences and workshops proceedings.*