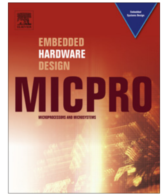
Microprocessors and Microsystems 37 (2013) 801–810

Contents lists available at SciVerse ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro



Leak-Gauge: A late-mode variability-aware leakage power estimation framework

Omid Assare [a,*], Mahmoud Momtazpour [a,b], Maziar Goudarzi [b]

[a] Department of Electrical Engineering, Sharif University of Technology, Tehran 11365-9363, Iran
[b] Energy Aware Systems Laboratory, Department of Computer Engineering, Sharif University of Technology, Tehran 11155-9517, Iran

ARTICLE INFO

Article history:
Available online 10 May 2013

Keywords:
Variability
Process variation
Leakage power

ABSTRACT

Leakage power has already become the major contributor to the total on-chip power consumption, rendering its estimation a necessary step in the IC design flow. The problem is further exacerbated with the increasing uncertainty in the manufacturing process known as process variability. We develop a method to estimate the variation of leakage power in the presence of both intra-die and inter-die process variability. Various complicating issues of leakage prediction such as spatial correlation of process parameters, the effect of different input states of gates on the leakage, and DIBL and stack effects are taken into account while we model the simultaneous variability of the two most critical process parameters, threshold voltage and effective channel length. Our subthreshold leakage current model is shown to fit closely on the HSPICE Monte Carlo simulation data with an average coefficient of determination (R^2) value of 0.9984 for all the cells of a standard library. We demonstrate the adjustability of this model to wider ranges of variation and its extendability to future technology scalings. We also present a complete framework for estimation of full-chip leakage power and show that our framework which we call Leak-Gauge, imposes little timing penalty on the system design flow and is applicable to real design cases.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

As the semiconductor technology scales into ever deeper regimes, power dissipation asserts its dominance as one of the most critical design metrics. The increase in dynamic power consumption is usually compensated for by down-scaling of the supply voltage. This scaling necessitates a decrease in transistor threshold voltages to maintain their switching speed and chip performance. As a result and due to its exponential dependence on threshold voltage, leakage power has emerged as the main contributor to the total on-chip power dissipation [1,2]. Leakage current is also exponentially dependent on transistor critical dimensions which are rapidly scaling down as the technology moves towards one-digit-nanometer nodes.

Also concomitant to the aggressive scaling of transistors is the increased variability in manufacturing process parameters, specifically transistor threshold voltage and critical dimensions such as effective channel length. The strong dependence of leakage power on these varying parameters renders it as an essentially random variable. Traditional worst case corner-based approach has proved too pessimistic, making it inevitable to move toward statistical estimation methods in order to obtain detailed and accurate information on the behavior of leakage power. Research efforts made in this direction in the past few years have produced fast and relatively accurate leakage estimation methods that can be used through the design flow. However, the lack of a late-in-the-design-process framework that could provide the levels of accuracy needed in a sign-off tool is more sensed as leakage power becomes more and more critical. Focusing on estimation accuracy, in this paper, we introduce Leak-Gauge, a leakage estimation framework aimed at the late stages of design flow.

Our main contributions may be summarized as follows:

- We develop a novel, variation-aware leakage current model by examining all major considerations at the transistor level in order to enable highly accurate characterization of standard cells. To the best of our knowledge, Leak-Gauge is the first framework that considers both the major sources (V_{th} and L_{eff}) and types (intra-die and inter-die) of process variation while addressing other complicating issues such as DIBL effect. The model is also highly flexible in terms of the level of the accuracy it provides and can be adjusted to handle further technology scalings by choosing the appropriate degree of approximation.

* Corresponding author. Present address: Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA 92093, USA. Tel.: +1 858 900 8137.

E-mail addresses: omid@ucsd.edu (O. Assare), momtazpour@ee.sharif.edu (M. Momtazpour), goudarzi@sharif.edu (M. Goudarzi).

0141-9331/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.micpro.2013.04.010

- We propose a complete framework for estimation of full-chip leakage power distributions in presence of process variation. While as a late-mode estimation framework, we primarily aim for accuracy and use a Monte Carlo based approach as opposed to popular statistical methods, the components of the framework are designed in a way that only a minimal timing penalty is imposed on the flow of system design. Moreover, the data- and task-parallelism present in our method which enable its implementation in parallel computing machines and its linear computation complexity make it scalable enough to be used for larger designs in future technologies.

Fig. 1 shows an overview of our leakage estimation framework and when each stage is performed in relation to the traditional design steps. Leak-Gauge consists of four components: *Cell Characterization*, *Variation Map Generation*, *Probability Calculation*, and *Leakage Estimation*. Cell Characterization is performed to develop variation-aware leakage models for standard library cells by actual measurements or HSPICE simulations. The exact formulations of cell leakage variation models and the procedures for obtaining them will come in Section 3. To capture the combined effect of both die-to-die and within-die variations, *variation maps* are generated in Variation Map Generation module. Separate maps are created for each process parameter, each map representing the variability in a process parameter across the surface of a sample die. Signal and input state probabilities of circuit nodes are computed in Probability Calculation step which can be performed right after circuit synthesis. More details on the generation of variation maps and the process parameter models used as well as signal probability calculation and its effect on leakage power will come in Section 4. Finally, Leakage Estimation stage (box number 4 in Fig. 1) takes four inputs: (1) cell leakage models produced by Cell Characterization module, (2) variation maps produced by Variation Map Generation module, (3) state probabilities produced in Probability Calculation step, and (4) the final layout of the chip extracted as a product of the P&R stage, and produces final leakage power distribution as the framework outputs.

This paper is an extension to the work presented in [4]. The extension includes:

- The effect of input combinations on the leakage power of individual CMOS cells as well as large circuits is investigated.
- A new module called Probability Calculator is added to the framework to account for this phenomenon. While this module introduces almost no added timing penalty, it greatly improves the accuracy and reliability of the framework.
- Extended experiments incorporating this new module have been performed and more detailed results are presented and discussed in the Experimental Results section.

The remainder of the paper is organized as follows. First, Section 2 provides a survey of the previous related work. We then continue, in Section 3, with constructing a model for analyzing the simultaneous effect of threshold voltage and effective channel length variations on leakage current of a cell, and explain how we use the model for characterization of standard library cells. Section 4 describes the probability calculator module, our process variation model, and the procedures used for Monte Carlo experiments to derive final leakage distributions. We present an evaluation of the accuracy and performance of our framework in Section 5 and compare it to another well-known approach. We conclude our discussion in Section 6.

2. Related work

The methods for estimating the effect of process variation on power consumption used in previous research can be divided into two categories. Monte Carlo based approach is often taken too strictly [18] and includes extensive complex computations which require long simulation times. The majority of the works have, however, focused on the Wilkinson's Method (WM) as the most effective approach for approximating the sum of lognormal distribution functions [22,21,28,20,27,9]. These methods exchange the accuracy of Monte Carlo method with reduced estimation time and computation complexity. A third approach, recently introduced in [10], makes statistical leakage estimation even faster by exploiting the spatial correlation property of leakage power while maintaining the same levels of accuracy as WM-based methods. We take a hybrid approach where we first develop a simple model for estimation of the leakage current of standard library cells and then use Monte Carlo experiments to obtain full-chip leakage power distribution. This results in a highly accurate method while keeping run-times at acceptable levels.

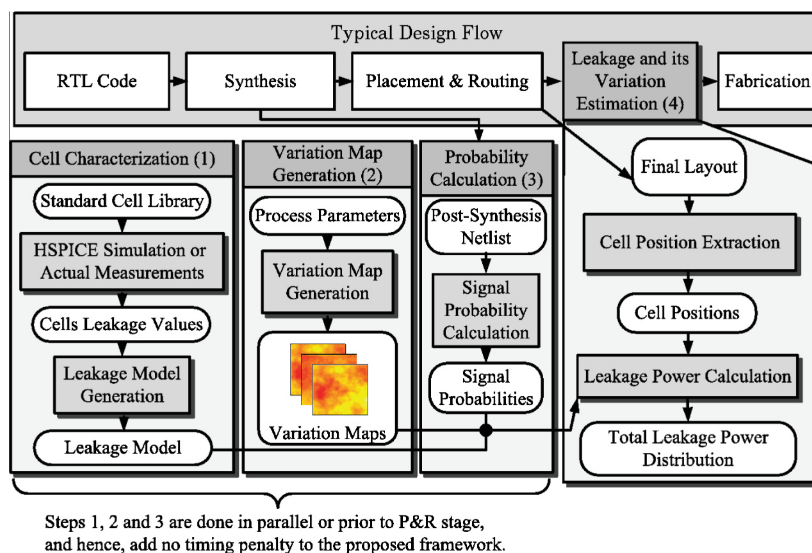


Fig. 1. Overview of our variability modeling framework.

In order to be satisfactorily accurate, a variation model must take various considerations into account. Several works on statistical power analysis and estimation have investigated these concerns. However, none have presented a comprehensive framework that could address all issues simultaneously, and are thus not accurate enough. Three distinct deficiencies could be observed in the existing estimation methods. A majority of them do not consider inter-die (die-to-die) variations (i.e. the difference in process parameters of identical chips within a wafer) [18], intra-die (within-die) variations (i.e. the difference in process parameters of identically designed transistors within a die) [20], or the spatial correlation of process parameters among proximate cells [27,18]. Most of the existing models do not take into account the simultaneous effect of threshold voltage and effective channel length on leakage current [19,27], or assume a linear relation between these parameters [21,20]. And finally, specific considerations of leakage current modeling such as the effect of the input state, or DIBL and stack effects are not taken into account in many frameworks [20,18,28,27].

As will be discussed in the following sections, our framework addresses all these issues and provides accurate estimations of leakage power distribution while it is fast enough to be included in CAD tools as a part of variation-aware chip design flow.

3. Cell characterization

3.1. Formulation

Subthreshold leakage current is considered as the main contributor to the total on-chip leakage current as rapid adaptation of high-k gate dielectric has effectively reduced gate leakage [23]. As discussed in [26], subthreshold current of a CMOS network is proportional to subthreshold current of one of its constructing transistors with the proportionality constant and the specific transistor being different for each set of inputs to the gate. Thus, we can model subthreshold leakage current of a cell with m transistors numbered from 1 to m, corresponding to its ith set of possible inputs as

$$I_{sub}(i) = a(i) \cdot I_{sub}^{N(i)}; \quad 1 \leqslant N(i) \leqslant m \tag{1}$$

where $a(i)$ and $N(i)$ are, respectively, the proportionality constant and the transistor number specific to the ith input set, and I_{sub}^{j} is the subthreshold current of the jth transistor. Although [26] does not consider process variability, its results may be extended to our discussion of process variation. Hence, to analyze the effect of process variation on the leakage current of a cell, it suffices to model the leakage current of one transistor in the presence of process variation for every set of inputs to the cell.

We, therefore, proceed by constructing a model for the subthreshold current of a transistor for analyzing the simultaneous variability in V_{th} and L_{eff}. The model will have the general form of $y = f(x_1,x_2)$ where y symbolizes the subthreshold current, $f(\cdot)$ represents its dependence on the variability in threshold voltage and effective channel length and x_1 and x_2 characterize this variability. The remainder of this section explains how we choose x_1, x_2, and the function $f(\cdot)$. For function $f(\cdot)$ to be derivable in a closed form, random variables x_1 and x_2 must meet the following two conditions:

- They must, jointly, include all variation sources that change threshold voltage and effective channel length.
- They must be statistically uncorrelated; i.e. a single variation source must not change both of them simultaneously.

According to BSIM4 device model [29], effective channel length of a transistor is modeled independent of threshold voltage. Therefore, we can choose $x_1 = L_{eff}$ without violating the second condition mentioned above. Moreover, x_1 will naturally include all the variation sources that change the effective channel length, working towards meeting the first condition. Threshold voltage, on the other hand, is not independent of channel length. Hence, we cannot assign it to x_2. We must differentiate between *channel-length-dependent* threshold voltage variations (variations in threshold voltage stemming from variations in channel length) and *channel-length-independent* variations (variations in threshold voltage that are not affected by channel length variations), and choose x_2 such that it includes all channel-length-independent variations, but no channel-length-dependent variations. This choice of x_2 along with our choice of $x_1 = L_{eff}$ will then satisfy both conditions mentioned above. Eq. (2) shows the BSIM4 model for threshold voltage.

$$V_{th} = VTH0 + \Delta V_{th}(SCE, DIBL) + \Delta V_{th}(Dopant) + \Delta V_{th}(Narrow_Width) \tag{2}$$

Here, $VTH0$ is the threshold voltage of the long channel device at zero substrate bias, $\Delta V_{th}(SCE,DIBL)$ is the change in threshold voltage due to short channel and DIBL effects, $\Delta V_{th}(Dopant)$ is the change in threshold voltage due to non-uniform substrate doping, and $\Delta V_{th}(Narrow_Width)$ is the change in threshold voltage due to the decrease in channel width.

Variations in threshold voltage can, therefore, be contributed to variations in the components of (2). We should, then, examine each component's dependency on channel length to identify it as either a channel-length-independent or channel-length-dependent variation source. $\Delta V_{th}(SCE,DIBL)$ varies exponentially with L_{eff} and is clearly a channel-length-dependent variation source. $\Delta V_{th}(Dopant)$ models the non-uniformity of dopant concentrations in both vertical and lateral directions.

$$\Delta V_{th}(Dopant) = \Delta V_{th}(Vertical) + \Delta V_{th}(Lateral) \tag{3}$$

$\Delta V_{th}(Vertical)$ is completely independent of channel length rendering it as a channel-length-independent variation source. Non-uniform doping concentration in lateral direction will, on the other hand, result in greater V_{th} roll-ups in short-channel devices resembling the L_{eff}-dependent nature of $\Delta V_{th}(Lateral)$. However, compared to the exponential dependency of $\Delta V_{th}(SCE,DIBL)$ on channel length, we can safely neglect the weak dependency of $\Delta V_{th}(Lateral)$ on L_{eff} and place this component in the channel-length-independent category as well. Therefore, while $\Delta V_{th}(Dopant)$ covers Random Dopant Fluctuations (RDF) as the main source of variation in threshold voltage, it does not include any channel-length-dependent variations. Finally, $\Delta V_{th}(Narrow_Width)$ changes threshold voltage in two ways as shown in (4).

$$\Delta V_{th}(Narrow_Width) = \Delta V_{th}(Narrow_Width1) + \Delta V_{th}(Narrow_Width2) \tag{4}$$

As channel width increases, the depletion region underneath the fringing fields becomes comparable to the depletion layer formed from the vertical field, and this in turn increases the threshold voltage. Using BSIM4 notation, we call this $\Delta V_{th}(Narrow_Width1)$ and represent the change in threshold voltage due to the width effect for small channel lengths with $\Delta V_{th}(Narrow_Width2)$. According to [29], $\Delta V_{th}(Narrow_Width1)$ and $\Delta V_{th}(Narrow_Width2)$ are, respectively, completely independent and exponentially dependent on effective channel length. Thus, the two terms correspond to channel-length-independent and channel-length-dependent variations respectively.

To summarize our discussion of threshold voltage variation sources, we can rewrite (2) as

$$V_{th} = VTH0 + \Delta V_{th}(CLI) + \Delta V_{th}(CLD) \tag{5}$$

where

$$\Delta V_{th}(CLI) = \Delta V_{th}(Dopant) + \Delta V_{th}(Narrow_Width1) \tag{6}$$

$$\Delta V_{th}(CLD) = \Delta V_{th}(SCE, DIBL) + \Delta V_{th}(Narrow_Width2) \tag{7}$$

It is now clear that choosing $x_2 = \Delta V_{th}(CLI)$ together with our choice of $x_1 = L_{eff}$ satisfies both conditions mentioned for the independent variables of our model. We now proceed to determine the function $f(\cdot)$. To this end, we first express leakage current as functions of each model parameter (i.e. V_{th} and L_{eff}) separately. In other words, we first choose the two functions $f_1(x_1) = f(x_1, x_2 = c)$ and $f_2(x_2) = f(x_2 = c, x_2)$. Then, considering the fact that the final model must preserve the behavior expressed in these two functions in cases where one parameter is held fixed, we propose the leakage current model as a product of these two functions.

Based on BSIM4 device model [29], subthreshold leakage current of a transistor can be expressed as

$$I_{sub} = I_0 \cdot \left[exp\left(\frac{V_{gs} - V_{off} - V_{th}}{nV_T}\right)\right] \cdot \left[1 - exp\left(\frac{-V_{ds}}{V_T}\right)\right] \tag{8}$$

where I_0 is the nominal subthreshold current, V_T is the thermal voltage, V_{off} is the offset voltage which determines the channel current at $V_{gs} = 0$, and n is the subthreshold swing factor. Therefore, for a given set of voltages on its terminals, subthreshold current of a transistor as a function of V_{th} can be modeled as

$$I_{sub} = C_1 \cdot exp(C_2 \cdot V_{th}) \tag{9}$$

where

$$C_1 = I_0 \cdot exp\left(\frac{V_{gs} - V_{off}}{nV_T}\right) \cdot \left[1 - exp\left(\frac{-V_{ds}}{V_T}\right)\right] \tag{10}$$

$$C_2 = -\frac{1}{nV_T} \tag{11}$$

Substituting (5) in (9) gives

$$I_{sub,V} = C_1 \cdot exp[C_2 \cdot (VTH0 + \Delta V_{th}(CLI) + \Delta V_{th}(CLD))] \tag{12}$$

now, by considering constant L_{eff}, we obtain

$$I_{sub,V} = C_1' \cdot exp(C_2 \cdot \Delta V_{th}(CLI)) = C_1' \cdot exp\ (C_2 \cdot VTH) \tag{13}$$

where

$$C_1' = C_1 \cdot exp[C_2(VTH0 + \Delta V_{th}(CLD))] \tag{14}$$

and VTH is the simple notation we will use for channel-length-independent variation in threshold voltage in our final model and in the remaining of the paper. Since this model works in situations where L_{eff} is held fixed ($x_1 = c$), it may serve as our $f_2(x_2)$ function.

To find subthreshold leakage as a function of L_{eff}, we can substitute V_{th} in (9) with a function describing its dependency on L_{eff}. However, since it is not possible to analytically find such a function, we can approximate it with a polynomial function of L_{eff} to obtain

$$I_{sub,L} = q_0 \cdot exp\left(\sum_{i=1}^{n} q_i \cdot (L_{eff})^i\right) \tag{15}$$

where q_is are fitting parameters and n is the degree of approximation. Since this model works in situations where V_{th} is held fixed ($x_2 = c$), it may serve as our $f_1(x_1)$ function. Now that we have chosen $f_1(x_1)$ and $f_2(x_2)$, we propose to form the final model as $f(x_1, x_2) = f_1(x_1) \cdot f_2(x_2)$ which gives

$$I_{sub} = c \cdot exp\left(p_0 \cdot VTH + \sum_{i=1}^{n} p_i \cdot (L_{eff})^i\right) \tag{16}$$

where c, and p_is are coefficients to be determined based on actual leakage values. As mentioned earlier, note that although (16) is essentially a model for subthreshold current of a single transistor, we may use it to model the current of a cell as well. This is based on the results of [26] that the subthreshold current of a cell is proportional to the current of one of its constructing transistors. Therefore, we can fit (16) on the leakage values of a cell, merging the proportionality constant (expressed in our cell leakage model (1) as $a(i)$) with the coefficient c in (16). However, it should be noted that the coefficients of (16) must be chosen for each set of inputs to the cell separately. In other words, we will have a separate subthreshold current model for each input set.

3.2. Characterization

Note that n in (15) must be chosen based on the technology node under study. Fig. 2 illustrates the dependency of threshold voltage on effective channel length over a wide range of L_{eff} values shown in meter. It can be observed that while a quadratic or even a linear approximation function of L_{eff} to model V_{th} may be acceptable for 45 nm and above technologies, the increased scaling of transistor dimensions and wider variability range of future technologies make more accurate approximations necessary. As shown in Fig. 1, in this step, leakage current models are constructed for all the cells of a standard library. We propose guidelines as to how the coefficients of (16) should be determined. Being independent of the specific design, this step can be performed in parallel to, or even before the placement and routing, imposing no timing penalty on the chip design flow. Moreover, this step needs to be performed only once, and all the designs based on the characterized library may use its results.

To determine the coefficients of (16), we only need $n + 2$ accurate leakage values, n being the degree of approximation as introduced in (15). These values can be obtained through either actual measurements or transistor-level simulations with $n + 2$ sets of doping concentration and channel length values preferably scattered across the desirable range of parameter variations. We will, then, have $n + 2$ ordered triples, (I_1, VTH_1, L_1) through $(I_{n+2}, VTH_{n+2}, L_{n+2})$, which when substituted in (16), will provide us with a system of $n + 2$ equations. Dividing all equations by one of them, say the first one, and taking log of both sides will result in a system of $n + 1$ linear equations ($I = AP$) where A, P, and I are defined as

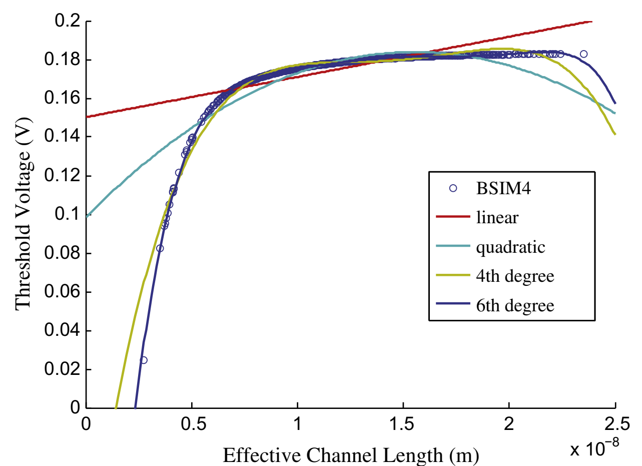


Fig. 2. The dependency of threshold voltage on effective channel length over a wide range of L_{eff} variations.

$$A = \begin{pmatrix} VTH_2 - VTH_1 & L_2 - L_1 & \dots & L_2^n - L_1^n \\ VTH_3 - VTH_1 & L_3 - L_1 & \dots & L_3^n - L_1^n \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ VTH_{n+2} - VTH_1 & L_{n+2} - L_1 & \dots & L_{n+2}^n - L_1^n \end{pmatrix}$$

$$P = \begin{pmatrix} p_0 \\ p_1 \\ . \\ . \\ . \\ p_n \end{pmatrix} I = \begin{pmatrix} log\left(\frac{I_2}{I_1}\right) \\ log\left(\frac{I_3}{I_1}\right) \\ . \\ . \\ . \\ log\left(\frac{I_{n+2}}{I_1}\right) \end{pmatrix}$$

Therefore, p_is can be calculated as

$$p_i = \frac{|A_i|}{|A|}, \quad 0 \leqslant i \leqslant n$$

where A_i is the matrix formed by replacing the ith column of A by the column vector I. Then, replacing these values in the first equation will give c as

$$c = \frac{I_1}{exp\left(p_0 \cdot VTH_1 + \sum_{i=1}^{n} p_i \cdot (L_{eff})^i\right)}$$

These coefficients, which are determined for all the cells in the standard library, are then used in the "Variability Analysis" stage which is explained next.

4. Variability analysis

In this section, we discuss the impact of input combinations on individual cell and full-chip leakage power, explain the process variation model used in this work, and describe the procedures used to obtain final leakage power distributions.

4.1. Input combination probabilities

Due to the well-known stack effect and its input pattern dependence, leakage power of CMOS cells vary considerably for different input sets [21]. Table 1 shows the maximum to minimum ratio of leakage power for a number of cells in a standard library [17]. Judging from these values, the leakage power of a CMOS circuit can be strongly dependent on the signal probability of its nodes. Therefore, it is essential for a leakage estimation framework to include an accurate signal probability approximation module. Fig. 3 shows the total leakage power of our simple OperRISC processor core, normalized to nominal, variation-free values (details in Section 5), for different input signal probability values. With this design as an example, it should be noted that although a majority of actual design cases are large enough that this input dependency phenomenon is averaged off, the wide spread of leakage values across input combinations and, hence, the possibility of considerable impact of input patterns cannot be overlooked.

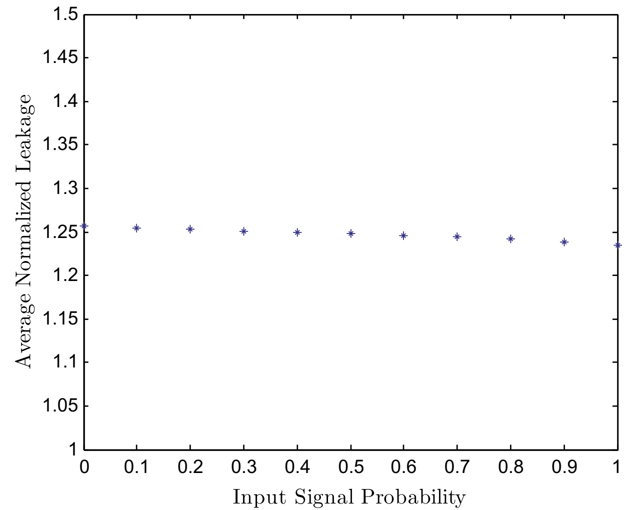


Fig. 3. Total leakage power (Normalized to Nominal, Variation-Free Values) of OpenRISC processor core for different input signal probabilities.

Input set probabilities can be computed using the signal probabilities of circuit nodes which may be obtained in a variety of ways. Examples are the algorithms presented in [12–14]. The choice of a specific algorithm must be made according to the design on which the framework is being applied. However, it should be noted that since only a circuit netlist (and not the circuit placement) is needed to obtain these probabilities, this step may be performed prior to the placement and routing stage and does not contribute to the timing overhead of the framework (see Fig. 1).

Table 1
Maximum to minimum ratio of leakage power for some of the cells in a standard library for different input sets.

Cell name	DFF_X1	INV_X4	AND2_X4
$\frac{MAX(Leakage)}{MIN(Leakage)}$	7.2	5.2	5.4
Cell name	INV_X2	INV_X1	INV_X8
$\frac{MAX(Leakage)}{MIN(Leakage)}$	7.5	7.3	4.0
Cell name	AND2_X1	AND2_X2	NAND2_X2
$\frac{MAX(Leakage)}{MIN(Leakage)}$	5.0	6.9	5.4

4.2. Process variation model

Similar to [6–8], we model the random and systematic components of both die-to-die (D2D) and within-die (WID) variations as zero-mean normal distributions. Therefore, each process parameter is modeled as a random variable as shown in (17) and (18) with four Gaussian independent components that correspond to systematic D2D variation, random D2D variation, systematic WID variation and random WID variation.

$$N_{dep} = n_0 + n_{sys_d2d} + n_{rnd_d2d} + n_{sys_wid} + n_{rnd_wid} \quad (17)$$
$$L_{eff} = l_0 + l_{sys_d2d} + l_{rnd_d2d} + l_{sys_wid} + l_{rnd_wid} \quad (18)$$

Here, n_0 and l_0 are nominal values of doping concentration and effective channel length and the remaining terms represent the four random variables named above. As in [11], we assume that each of the mentioned components contribute equally to the total variation. VTH values corresponding to each variation-included N_{dep} value are obtained using equations in [29]. We use geoR [15], geostatistical analysis package of R [16] to generate large numbers of different sets of WID and D2D *variation maps* for each of the process parameters while taking spatial correlation of the effective channel length into account. A variation map reflects the amount of variation in a process parameter across the surface of the die. Similar to [8], we use the spherical function (19) to incorporate spatial correlation of the effective channel length.

$$\rho(r) = \begin{cases} 1 - \frac{3r}{2\Phi} + \frac{r^3}{2\Phi^3} & r \leqslant \Phi \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

Here, $\rho(r)$ is the correlation function, r is the distance between two points on the die, and Φ is the *Range*, the distance between two points where they will be no longer correlated. We then divide the die area into regions of area A_{Region}. A_{Region} is chosen such that the correlation function $\rho(r)$ would not change considerably across each region.

4.3. Full-chip leakage estimation

After the characterization stage where leakage models for each input set of a cell are constructed using (16), we form the final leakage model of a cell with n input nodes as

$$I_{sub}^{cell} = \sum_{i=1}^{2^n} P_i \cdot I_{sub}(i) \tag{20}$$

where P_i is the probability of the occurrence of the ith input set, and $I_{sub}(i)$ is the leakage model of the ith input set as introduced in (1).

As illustrated in Fig. 1, this step is performed right after the placement-and-routing stage of the design when the positions of all the cells of the chip are determined. The total leakage power consumption of a chip is calculated using a summation of the leakage values of all its constructing cells. After the placement and routing stage, the positions of the cells can be extracted from the final layout of the design. For each variation map, the value of each transistor parameter may be determined for all the cells. These values may then be used to calculate the total leakage power of the chip. This process, when performed for all variation maps in a series of Monte Carlo experiments, provides us with an accurate estimation of leakage power consumption in presence of process variation.

5. Experimental results

In this section, we investigate the accuracy and timing performance of our framework and compare it to the widely used Wilkinson's Method as a representative of the related work. We evaluate our framework against transistor level HSPICE simulation data to verify its accuracy and apply it to a real design to examine its imposed timing overhead over the system design flow. We further present a complexity analysis of our framework to evaluate its scalability and practicality of its application to larger designs.

5.1. Accuracy evaluation

We evaluated our leakage current model by fitting it on leakage values generated by HSPICE simulation and analyzing the resulting goodness-of-fit measures. *Note that since we use no approximations for deriving final leakage power distributions from these models, our evaluation of the accuracy of them serves as the evaluation of our variability modeling framework as a whole*. In other words, assuming the accuracy of the process variation parameters and our leakage models, our approach to finding the leakage power distributions (Monte Carlo experiments of calculating leakage values for each variation map) closely resembles a practical scenario in which one would form a distribution of leakage power consumption of a set of chips by measuring their actual leakage currents after fabrication. Furthermore, our approach would nullify the inaccuracies inherent to measuring devices by calculating the full-chip leakage as a mathematical summation of cell leakages. With these arguments as the justifications for our evaluation method, we proceed to evaluate the accuracy of our leakage model.

Similar to [27], we set n in (16) to 2. We show that this is sufficient for our 45-nm library, but note that a higher degree of approximation may be needed for more scaled technologies. Our subthreshold current model would then be formed as

$$I_{sub} = c \cdot exp(p_0 \cdot VTH + p_1 \cdot L_{eff} + p_2 \cdot (L_{eff})^2) \tag{21}$$

We used 18 most common cells of Nangate Open Cell Library [17], and ran 1000 simulations per cell per input set. Table 2 summarizes the process parameters corresponding to 45 nm technology. We stored corresponding *VTH* and leakage current values obtained from the formulas in [29] and HSPICE simulations. Finally, we used MATLAB surface fitting tool for the process of non-linear multivariate regression and determining goodness-of-fit measures. Fig. 4 and Table 3 show the results of the fitting process for a 2-input NAND cell as an example. In Table 3, *SSE*, *R-square* and *RMSE* correspond to *Sum of Squares Due to Error*, *Coefficient of Determination*, and *Root Mean Squared Error* respectively. SSE measures the total deviation of the response values from the fit to the response values. A value closer to 0 indicates that the model has a smaller random error component, and that the fit will be more useful for prediction. R-square, the coefficient of determination, measures how successful the fit is in explaining the variation of the data. It can take on any value between 0 and 1, with a value closer to 1 indicating that a greater proportion of variance is accounted for by the model. For example, an R-square value of 0.8234 means that the fit explains 82.34% of the total variation in the data about the average. RMSE is known as the fit standard error and the standard error of the regression. It is an estimate of the standard deviation of the random component in the data. Just as with SSE, an RMSE value closer to 0 indicates a fit that is more useful for prediction [25]. As evident in the figures and the goodness-of-fit measures, the model fits closely onto the HSPICE simulated leakage values and is robust enough to account for DIBL and stack effects. Table 4 lists the corresponding coefficient of determination for each cell. The mean coefficient of determination (R^2) of our model for all the used cells in the standard library is 0.9984 with a minimum of 0.9953 at the worst case indicating the good accuracy of the model. As stated before, we perform Monte Carlo experiments to derive final leakage distributions by summing the leakage values calculated using these models, and there exists no other approximations or error sources in the framework. Hence, we have demonstrated the accuracy of our framework by verifying the accuracy of our cell leakage models.

5.2. Performance evaluation

We applied our framework to a real design to measure its imposed timing overhead. Note that the three components of *Cell Characterization*, *Variation Map Generation*, and *Probability Calculation* can be performed off-line and in parallel or prior to the design flow, and do not contribute to the timing overhead of our framework. The overhead is measured in the *Leakage Estimation* stage (more specifically in *Total Leakage Calculator* block in Fig. 1). The experimented design was an OpenRISC [24] processor core. The core was synthesized by Synopsys Design Compiler using the 18 cells listed in Table 4 and placed and routed by Cadence Encounter. The resulting layout consisted of around 15 k standard cells and occupied a die area of $200 \times 200\ \mu m^2$. Signal and input combination probabilities were computed using an algorithm based on [14]. The same process variation values described in Table 2 were used and a set of 900 variation maps were produced using the method described in Section 4.1 with a Φ value of 0.5 and A_{Region} value of $0.5 \times 0.5\ \mu m^2$. The timing overhead was a nominal 1.91 s

Table 2
Process parameters with respective mean and variance values.

Parameter	Mean (μ)	Variance ($3\sigma/\mu$) (%)
V_{th}	180 mV	30
L_{eff}	17.5 nm	15

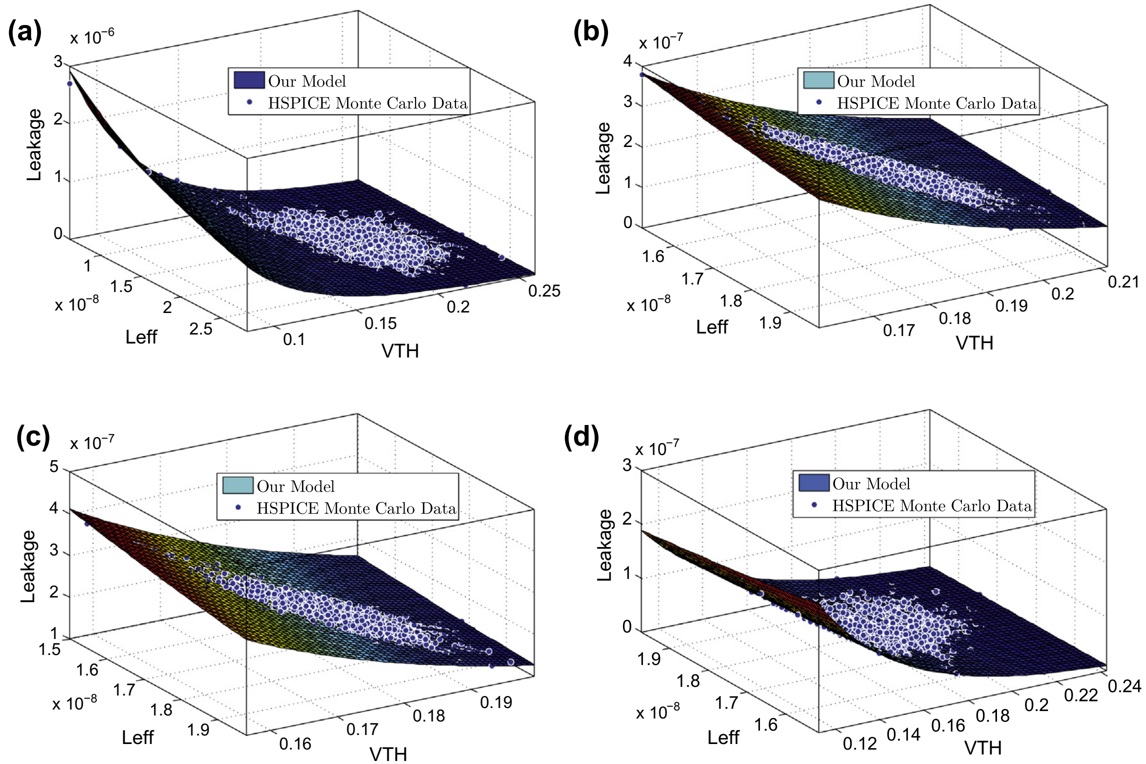


Fig. 4. Fitting results of our current leakage model on the HSPICE simulation data. Input sets: (a) 00, (b) 01, (c) 10, and (d) 11.

Table 3
Goodness-of-fit measures of our current leakage model for a 2-input NAND cell.

Input sets	SSE	R-square	RMSE
00	1.955e−16	0.9996	4.43e−10
01	3.78e−16	0.9999	6.161e−10
10	5.305e−16	0.9999	7.298e−10
11	1.74e−17	0.9999	1.322e−10

Table 4
Coefficient of determination values for the 18 cells used for the evaluation of our model and implementation of OpenRISC.

Cell Name	DFF_X1	INV_X4	CLKBUF_X1
R^2	0.9953	0.9988	0.9995
Cell Name	INV2_X2	INV2_X1	INV_X8
R^2	0.9996	0.9982	0.9984
Cell Name	AND2_X1	AND2_X2	NOR2_X2
R^2	0.9979	0.9983	0.9963
Cell Name	XOR2_X1	AND2_X4	NAND2_X2
R^2	0.9993	0.9992	0.9998
Cell Name	DLH_X1	NAND2_X1	MUX2_X1
R^2	0.9972	0.9997	0.9984
Cell Name	BUF_X1	DLL_X1	BUF_X2
R^2	0.9997	0.9972	0.9984

on a 1.73 GHz Intel Core 2 Duo machine with 2.00 GB RAM and running 32-bit Windows 7. Since the computation complexity of our method is $O(N)$ where N is the number of cells in the design, it can be inferred that our method would impose a timing penalty of around 1900 s (32 min) for a typical 15 million-cell design which demonstrates the practicality of its usage in real world applications.

5.3. Comparison with previous methods

At the heart of most statistical approaches to leakage power variability analysis lies the Wilkinson's Method, a well-known approach to approximate the sum of log-normals. This method is used to approximate the mean and standard deviation (SD) of the full-chip leakage power distribution using mean and SD values of individual standard cells. Most related works have focused on reducing the timing complexity of this method while trying to maintain its accuracy. Our approach is fundamentally different, in that it forms the desired leakage distribution by point-by-point mathematical summation of cell leakage values. In this subsection, we present a comparison of the accuracy and scalability of our framework with this method as a representative of related work. We show that Wilkinson's Method, even without timing optimizations which inevitably reduce its accuracy, is not accurate enough for variation-aware leakage estimation of aggressively scaled technologies.

5.3.1. Accuracy

Our Monte Carlo approach ensures that our method is superior in terms of accuracy assuming both methods use the same cell leakage models and input combination probabilities. This is because we use no approximations for calculating the final leakage distributions. Therefore, to examine the level of inaccuracy in Wilkinson's Method of summing log-normals, we implemented this method and used it to estimate the leakage power distribution of our OpenRISC design. Note that since we used the same cell leakage models as the input to Wilkinson's Method, the difference between its results and those obtained from our framework reflect the error introduced by Wilkinson's Method with respect to a purely Monte Carlo approach. Fig. 5 shows the final normalized leakage distributions derived from our framework and Wilkinson's Method for three input signal probability values of 0.25, 0.5, and 0.75. While the mean and SD values estimated by Wilkinson's Method are relatively

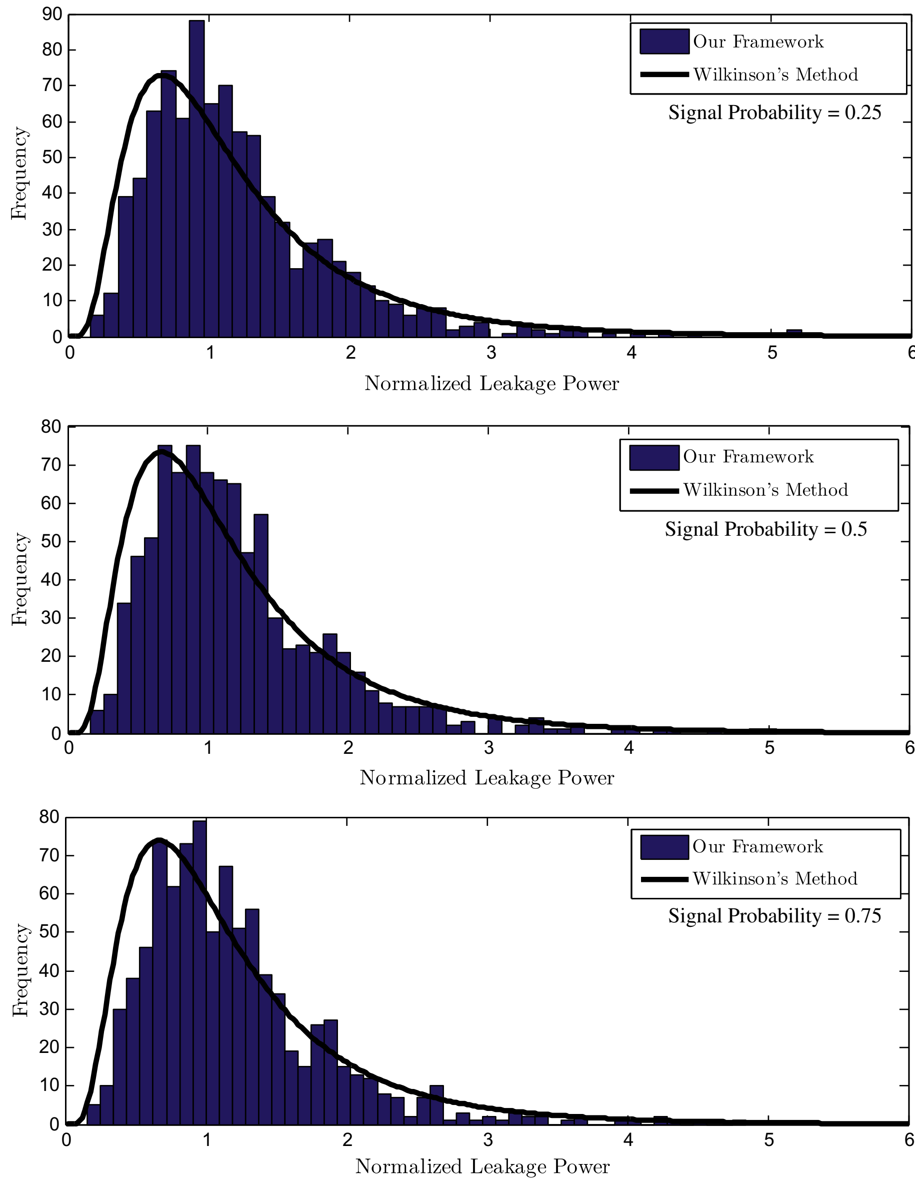


Fig. 5. Leakage power distribution of OpenRISC processor core using Wilkinson's method and our framework. From top to bottom, starting signal probability: 0.25, 0.5, 0.75.

accurate, with respectively 2.1% and 3.4% average errors with respect to our Monte Carlo based approach, the assumption that the full-chip leakage distribution follows a log-normal pattern introduces larger errors. As an example of possible errors in practical applications, we compared the leakage-yield (i.e. the percentage of chips consuming no more leakage power than a specified leakage power constraint) of our OpenRISC design estimated by both methods. Table 5 shows these values and their difference for various leakage power constraints. The constraints in the first column are expressed in relation to the nominal, variation-free leakage power estimate. WM, LG, and Error columns correspond, respectively, to the leakage yield predicted by Wilkinson's Method, leakage yield computed by our framework, and the error of the estimation performed by Wilkinson's Method with respect to our Monte Carlo approach. These results show that for all three input signal probability values, the Wilkinson's Method keeps unaccept-

Table 5
Leakage-yield values of Wilkinson's method and Leak-Gauge for Three different input signal probabilities.

Leakage constraints	Input signal probabilities								
	0.25			0.5			0.75		
	WM (%)	LG (%)	Error (%)	WM (%)	LG (%)	Error (%)	WM (%)	LG (%)	Error (%)
1.1	49	51	3.9	50	53	5.7	51	55	7.3
1.2	53	57	7.0	54	58	6.7	55	59	6.8
1.3	56	62	6.4	58	63	7.9	60	64	6.2
1.4	61	65	6.1	62	67	7.5	63	69	8.7
1.5	63	70	10	65	71	8.4	67	72	6.9
Average	–	–	6.7	–	–	7.2	–	–	7.2

able errors in approximating leakage-yield. This can be attributed to this method's simplifying assumption that the final distribution is a pure log-normal. As indicated by our results in Table 5, this assumption may be inaccurate especially in cases where there exists high levels of spatial correlation between process parameters. Our approach eliminates this possible inaccuracy by forming the final distribution using mathematical summations of cell leakage values.

5.3.2. Scalability

To approximate the distribution of the sum of log-normals, Wilkinson's Method assumes another log-normal as the resulting distribution and matches its first and second momentums with those of the distribution of the sum of log-normals as shown in the following equations.

$$e_1^Y + e_2^Y + \ldots + e_n^Y \approx e^Z$$

$$\mu_1 = E\left[e_1^Y + \ldots + e_n^Y\right] = e^{\mu_Z + \sigma_Z^2/2} = \sum_{i=1}^{n} e^{\mu_{Yi} + \sigma_{Yi}^2/2}$$

$$\mu_2 = E\left[\left(e_1^Y + \ldots + e_n^Y\right)^2\right] = e^{2\mu_Z + 2\sigma_Z^2}$$
$$= \sum_{i=1}^{n} e^{2\mu_{Yi} + 2\sigma_{Yi}^2} + 2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} e^{\mu_{Yi} + \mu_{Yj} + \left(\sigma_{Yi}^2 + \sigma_{Yj}^2 + 2\rho_{ij}\sigma_i\sigma_j\right)/2}$$

where ρ_{ij} is the correlation coefficient of Y_i and Y_j. Then, mean and standard deviation would be obtained as:

$$m_Z = 2ln\mu_1 - \frac{1}{2}ln\mu_2$$
$$\sigma_Z^2 = ln\mu_2 - 2ln\mu_1$$

It is easy to infer that the computation complexity of this method is $O(N^2)$ where N is the number of log-normals to be summed (i.e. the number of cells in our case) while our approach results in a complexity of $O(N)$. Of course, there have been several attempts to reduce this complexity. Chang and Sapatnekar [5] proposed a method to decrease the computation complexity of Wilkinson's Method to $O(n^2)$ where n is the number of grids used in variability analysis. It is obvious that n can be significantly smaller than the number of cells, but the quadratic complexity causes estimation times to increase dramatically for large, real world circuits. More recently, Kim et al. [28] have also proposed a method called VCA that has a computation complexity similar to ours ($O(N)$). However, their use of Wilkinson's Method keeps the possibility of unacceptable error rates as indicated by our results in Table 5. As expected, the VCA approach's error rate increases with the amount of process variability [28].

Another characteristic of Leak-Gauge that makes it increasingly scalable is the inherent task- and data-parallelism present in its different components that make it suitable to accelerate by running on multicore processors or cluster machines. Referring to Fig. 1, the cell characterization and variation map generation stages, while not contributing to the design time overheads, can be easily partitioned as parallel tasks to be performed on multiple processor cores. More importantly, the timing-critical phase of the framework which takes place in the leakage and variation estimation stage is highly data-parallel in nature. In fact, this stage can be viewed as a single summation reduction operation making the framework readily suitable in a parallel computing environment.

In summary, our framework has a lower or equal complexity compared to other leakage estimation methods while it is more accurate as it uses no approximations in forming the final distributions. Furthermore, our transistor level analysis of cell leakage power provides us with more accurate cell leakage models as shown in the previous subsection and renders our framework even more accurate.

6. Conclusion

We presented Leak-Gauge, a late-mode, variation-aware leakage power estimation method which is used as a part of our variability modeling framework [3]. We developed a subthreshold leakage current model and demonstrated its reliability through comparison of its predictions with HSPICE simulation data. We showed how our model could be adjusted for wide ranges of process variation and scaled transistor dimensions in future deep submicron regimes by choosing the degree of $V_{th} - L_{eff}$ dependency approximations. Through using a combination of Monte Carlo experiments and our developed leakage model, we could achieve high accuracy while maintaining an acceptable timing penalty imposed by the variability analysis procedures.

Acknowledgement

This work has been supported in part by the Institute for Research in Fundamental Sciences (IPM) under Grant No. CS1390-4-10.

References

[1] Semiconductor Industry Association, International Technology Roadmap for Semiconductors, 2007. <http://www.itrs.net/>.
[2] A. Chandrakasan, W. Bowhill, F. Fox, Design of High-Performance Micro-Processor Circuits, IEEE Press, 2001.
[3] O. Assare, H. Izady Rad, M. Momtazpour, E. Sanaei, M. Goudarzi, VAREX: a post-P& R variability modeling framework for multiprocessor SoCs, in: IEEE/ACM ICCAD Workshop on Variability Modeling and Characterization (VMC), November 2011.
[4] O. Assare, M. Momtazpour, M. Goudarzi, Accurate estimation of leakage power variability in sub-micrometer CMOS circuits, in: 15th Euromicro Conference on Digital System Design, September, 2012.
[5] H. Chang, S.S. Sapatnekar, Full-chip analysis of leakage power under process variations, including spatial correlations, in: Proceedings of Design Automation Conference, 2005, pp. 523–528.
[6] S. Chandra, K. Lahiri, A. Raghunathan, S. Dey, Considering process variations during system-level power analysis, in: Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED), October 2006, pp. 342–345.
[7] R. Teodorescu, J. Torrellas, Variation-aware application scheduling and power management for chip multiprocessors, in: 35th International Symposium on Computer Architecture (ISCA), June 2008, pp. 363–374.
[8] S.R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, J. Torrellas, VARIUS: a model of process variation and resulting timing errors for microarchitects, IEEE Transactions on Semiconductor Manufacturing 21 (2008) 3–13.
[9] Ye Zuochang, Yu Zhiping, An efficient algorithm for modeling spatially-correlated process variation in statistical full-chip leakage analysis, in: Digest of Technical Papers of International Conference on Computer-Aided Design (ICCAD), November 2009, pp. 295–301.
[10] R. Shen, S. Tan, H. Wang, J. Xiong, Fast statistical full-chip leakage analysis for nanometer VLSI systems, ACMTransactions on Design Automation of Electronic Systems (TODAES) 17 (51) (2012) (issue 4).
[11] T. Karnik, S. Borkar, V. De, Probabilistic and variation-tolerant design: key to continued Moore's law, in: ACM/IEEE TAU Workshop on Timing Issues in the Specification and Synthesis of Digital Systems, February 2004.
[12] G. Asadi, M.B. Tahoori, An accurate SER estimation method based on propagation probability, Proceedings of Design, Automation and Test in Europe (DATE) 1 (2005) 306–307.
[13] R. Marculescu, D. Marculescu, M. Pedram, Efficient power estimation for highly correlated input streams, in: 32nd Design Automation Conference (DAC), 1995, pp. 628–634.
[14] J. Dunoyer, N. Abdallah, P.B. Sabet, A symbolic simulation approach in resolving signals' correlation, in: Proceedings of the 29th Annual Simulation Symposium, April 1996, pp. 203–211.
[15] P. Ribeiro, P. Diggle, The geoR Package. <http://www.est.ufpr.br/geoR>.
[16] R Development Core Team. R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2007.
[17] The NanGate 45 nm Open Cell Library: An Open Source Standard Cell Library. <http://www.nangate.com/>.
[18] S. Mukhopadhyay, K. Roy, Modeling and estimation of total leakage current in nano-scaled CMOS devices considering the effect of parameter variation, in: Proceedings of International Symposioum on Low Power Electroninc Design, 2003, pp. 172–175.

[19] S. Narendra, V. De, S. Borkar, D. Antoniadis, A. Chandrakasan, Full-chip subthreshold leakage power prediction model for sub-0.18 in CMOS, in: Proceedings of International Symposioum on Low Power Electroninc Design, 2002, pp. 19–23.
[20] A. Agarwal, K. Kunhyuk, K. Roy, Accurate estimation and modeling of total chip leakage considering inter- & intra-die process variations, in: Proceedings of International Conference on Computer-Aided Design, 2005, pp. 736–742.
[21] H.F. Dadgour, S. Lin, K. Banerjee, A statistical framework for estimation of full-chip leakage-power distribution under parameter variations, IEEE Transactions on Electron Devices 54 (11) (2007) 2930–2945.
[22] N.C. Beaulieu, A.A. Abu-Dayya, P.J. McLane, Comparison of methods of computing lognormal sum distributions and outages for digital wireless applications, in: Proceedings of IEEE International Conference on Communications, 1994, pp. 1270–1275.
[23] R. Chau, S. Datta, M. Doczy, J. Kavalieros, M. Metz, Gate dielectric scaling for high-performance CMOS: from SiO2 to High-K, Extended Abstracts of International Workshop on Gate Insulator (IWGI) (2003) 124–126.
[24] OpenRISC 1200 Processor Core. <http://opencores.org/openrisc.or1200>.
[25] MATLAB Product Documentation. <http://www.mathworks.com/>.
[26] R.X. Gu, M.I. Elmasry, Power dissipation analysis and optimization of deep submicron CMOS digital circuits, IEEE Journal of Solid-State Circuits 31 (1996) 707–713.
[27] R. Rao, A. Sirvastava, D. Blaauw, D. Sylvester, Statistical analysis of subthreshold leakage current for VLSI circuits, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 12 (2) (2004).
[28] W. Kim, K.T. Do, Y.H. Kim, Statistical leakage estimation based on sequential addition of cell leakage currents, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 18 (4) (2010) 602–615.
[29] W. Liu, K.M. Cao, X. Jin, X. Xi, C. Hu, BSIM4.2.0 MOSFET Model-User Manual. <http://www-device.eecs.berkeley.edu/bsim3/bsim4.html>.

Omid Assare is pursuing a Ph.D. degree in Computer Science and Engineering at the University of California, San Diego. His research focuses on analysis and modeling of variability in deep submicron technologies. He has a B.Sc. in Electrical Engineering from Sharif University of Technology.

Mahmoud Momtazpour received his B.Sc., M.Sc. and Ph.D. degrees in Electrical Engineering from Sharif University of Technology in 2005, 2007 and 2012 respectively. He is now a postdoctoral research fellow in the same university. His research interests include system-level performance and power optimization for MPSoCs and data centers.

Maziar Goudarzi is an Assistant Professor at the Department of Computer Engineering, Sharif University of Technology, Tehran, Iran. He received the B.Sc., M.Sc., and Ph.D. degrees in Computer Engineering from Sharif University of Technology in 1996, 1998, and 2005, respectively. Before joining Sharif University of Technology as a faculty member in September 2009, he was a Research Associate Professor at Kyushu University, Japan from 2006 to 2008, and then a member of research staff at University College Cork, Ireland in 2009. His current research interests include green computing, hardware software codesign, and reconfigurable computing. Dr. Goudarzi has won two best paper awards, published several papers in reputable conferences and journals, and served as member of technical program committees of a number of IEEE, ACM, and IFIP conferences including ICCD, ASP-DAC, ISQED, ASQED, and IEDEC among others.