Accurate Estimation of Leakage Power Variability in Sub-Micrometer CMOS Circuits

Omid Assare*, Mahmoud Momtazpour*, and Maziar Goudarzi†‡

*Electrical Engineering Department, Sharif University of Technology, Tehran, Iran

†Computer Engineering Department, Sharif University of Technology, Tehran, Iran

‡School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran

omid@ee.sharif.edu, momtazpour@ee.sharif.edu, goudarzi@sharif.edu

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 also demonstrate the adjustability of this model to wider ranges of variation and its extendability to future technology scalings. We show that our framework imposes little timing penalty on the system design flow and is applicable to real design cases. The procedures explained in this paper are part of VAREX, an academic variability modeling framework for estimation of the effect of process variation on power consumption and performance of Multiprocessor SoCs.**

I. 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 information on the behavior of leakage power. Hence, it is essential to develop accurate variation-aware leakage power estimation methods to be used at all design levels. We are developing a variability modeling framework called VAREX [3] that estimates the distributions of power consumption and frequency of Multiprocessor SoCs. In this paper, we describe the models and procedures used by our framework to obtain leakage power distribution in the presence of process variation.

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, we are the first to simultaneously take into account major sources (V_{th} and L_{eff}) and types (intra-die and inter-die) of process variation and address 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.

- We propose a complete framework for estimation of full-chip leakage power distributions in presence of process variation. Our framework uses a Monte Carlo based approach, and is thus highly-accurate while the components of the framework are designed in a way that only a minimal timing penalty is imposed on the flow of system design.

A. 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 [12] and includes extensive complex computations which require long simulations times. The majority of the works have, however, focused on the Wilkinson's method as the most effective approach for approximating the sum of lognormal distribution functions [16], [15], [21], [14], [20]. These methods exchange the accuracy of Monte Carlo method with reduced estimation time and computation complexity. 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.

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) [12], intra-die (within-die) variations (i.e. the difference in process parameters of identically designed transistors within a die) [14], or the spatial correlation of process parameters among proximate cells [20], [12]. Most of the existing models do not take into account the simultaneous effect of threshold voltage and effective channel length on leakage current [13], [20]. Those that do, either assume a linear relation between these parameters [15], [14] or take an approach which is not extendable to future technology scalings [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 [14], [12], [21], [20].

B. Paper Overview

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 system level CAD tools as a part of variation-aware chip design flow. Fig. 1 shows an overview of our leakage estimation framework and when each stage is performed in relation to the traditional design steps. As illustrated in Fig. 1, *cell characterization* and *variation map generation* are performed in parallel to pre-P&R steps, imposing no additional timing penalty. After P&R, the position of the cells extracted from the final layout along with the variation maps reflecting within-die and die-to-die variations are fed into the leakage models of the characterization stage to derive the final leakage distribution.

The remaining of the paper is organized as follows. We begin, in Section II, 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 III describes our process variation model and the procedures used for Monte Carlo experiments to derive final leakage distributions. We present an evaluation of our framework in terms of accuracy and performance in Section IV and compare it to another well-known approach. Finally, in Section V, we conclude our discussion and point to the future directions for improving our work.

II. Cell Characterization

A. 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 [17]. As discussed in [19], subthreshold current of a CMOS
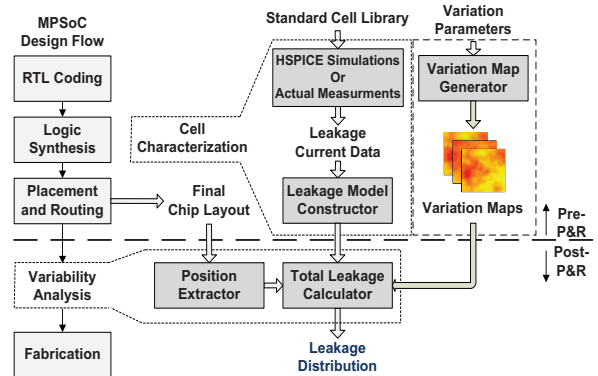

Fig. 1. Overview of our variability modeling framework

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, considering equal probabilities for all possible input sets, we model subthreshold leakage current of a cell with n inputs and m transistors numbered from 1 to m as

$$I_{sub}^{cell} = \frac{1}{2^n} \sum_{i=1}^{2^n} a(i).I_{sub}^{N(i)} \; ; \; 1 \leq N(i) \leq m \qquad (1)$$

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 [19] does not consider process variability, its results may be extended to our discussion of process variability. 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, $f(.)$ represents its dependence on the variability in threshold voltage and effective channel length and x_1 and x_2 characterize this variability. The remaining of this section explains how we choose x_1, x_2, and the function $f(.)$. For function $f(.)$ 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 [22], 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. Equation (2) shows the BSIM4 model for threshold voltage.

$$
\begin{aligned}
V_{th} &= VTH0 + \Delta V_{th}(SCE, DIBL) + \Delta V_{th}(Dopant) \\
&+ \Delta V_{th}(Narrow\_Width)
\end{aligned} \tag{2}
$$

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).

$$
\begin{aligned}
\Delta V_{th}&(Narrow\_Width) \\
&= \Delta V_{th}(Narrow\_Width1) \\
&+ \Delta V_{th}(Narrow\_Width2)
\end{aligned} \tag{4}
$$

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 [22], $\Delta V_{th}(Narrow\_Width1)$ 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

$$
\begin{aligned}
\Delta V_{th}(CLI) &= \Delta V_{th}(Dopant) \\
&+ \Delta V_{th}(Narrow\_Width1)
\end{aligned} \tag{6}
$$

$$
\begin{aligned}
\Delta V_{th}(CLD) &= \Delta V_{th}(SCE, DIBL) \\
&+ \Delta V_{th}(Narrow\_Width2)
\end{aligned} \tag{7}
$$

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 $f(.)$. 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 $f_2(x_2) = f(x_1 = c, x_2)$. 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 [22], subthreshold leakage current of a transistor can be expressed as

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

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

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

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

Substituting (5) in (9) gives

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

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

$$
\begin{aligned}
I_{sub,V} \\
&= C_1' \cdot exp\ (C_2 \cdot \Delta V_{th}(CLI)) \\
&= C_1' \cdot exp\ (C_2 \cdot VTH) \qquad (13)
\end{aligned}
$$

where

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

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

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

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 ($x2 = c$), 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

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

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 [19] 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.

B. 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 45nm 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
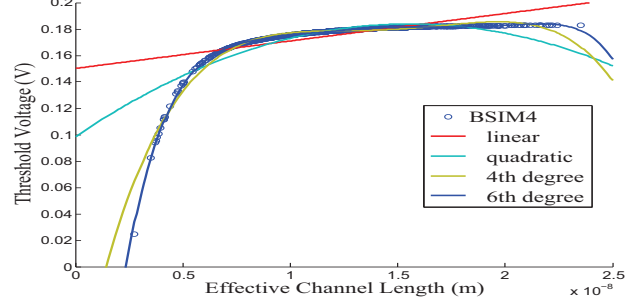

Fig. 2. The dependency of threshold voltage on effective channel length over a wide range of L_{eff} variations

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

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

$$
P = \begin{pmatrix}
p_0 \\
p_1 \\
. \\
. \\
. \\
p_n
\end{pmatrix}
$$

$$
I = \begin{pmatrix}
log(\frac{I_2}{I_1}) \\
log(\frac{I_3}{I_1}) \\
. \\
. \\
. \\
log(\frac{I_{n+2}}{I_1})
\end{pmatrix}
$$

Therefore, p_is can be calculated as

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

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

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

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.

III. VARIABILITY ANALYSIS

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. In this section, we explain the process variation model used in this work and describe the procedures used to obtain final leakage power distributions.

A. Process Variation Model

Similar to [5], [6], [7], 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 [8], 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 [22]. We use geoR [9], geostatistical analysis package of R [10] 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 [7], we use the spherical function (19) to incorporate spatial correlation of the effective channel length.

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

Here, $\rho(r)$ is the correlation function, r is the distance between two points on the chip, 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.

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

B. Full-Chip Leakage Estimation

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.

IV. 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. 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 prove its easy application to larger designs.

A. 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 [20], 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

$$I_{sub} = c \, . \, exp \, (p_0 \, . \, VTH + p_1 \, . \, L_{eff} + p_2 \, . \, (L_{eff})^2) \quad (20)$$

TABLE III

TABLE III
COEFFICIENT OF DETERMINATION VALUES FOR THE 18 CELLS USED FOR THE EVALUATION OF OUR MODEL AND IMPLEMENTATION OF OUR
OPENRISC-BASED MPSoC

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

TABLE II
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

We used 18 most common cells of Nangate Open Cell Library[11], and ran 1000 simulations per cell per input set. Table I summarizes the process parameters corresponding to 45nm technology. We stored corresponding VTH and leakage current values obtained from the formulas in [22] and HSPICE simulations. Finally, we used a MATLAB surface fitting tool for the process of non-linear multivariate regression and determining goodness-of-fit measures. Fig. 3 and Table II show the results of the fitting process for a 2-input NAND cell as an example. In Table II, *SSE*, *R-square* and *RMSE* correspond to *Sum of Squares Due to Error*, *Coefficient of Determination*, and *Root Mean Squared Error* respectively. 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 III 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.

B. Performance Evaluation

We applied our framework to a real design to measure its imposed timing overhead. Note that the two components of *Cell Characterization* and *Variation Map Generation* 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 *Variability Analysis* stage (more specifically in *Total Leakage Calculator* block in Fig. 1). The experimented design was an OpenRISC [18] processor core. The core was synthesized by Synopsys Design Compiler using the 18 cells listed in Table III and placed and routed by Cadence Encounter. The resulting layout consisted of around
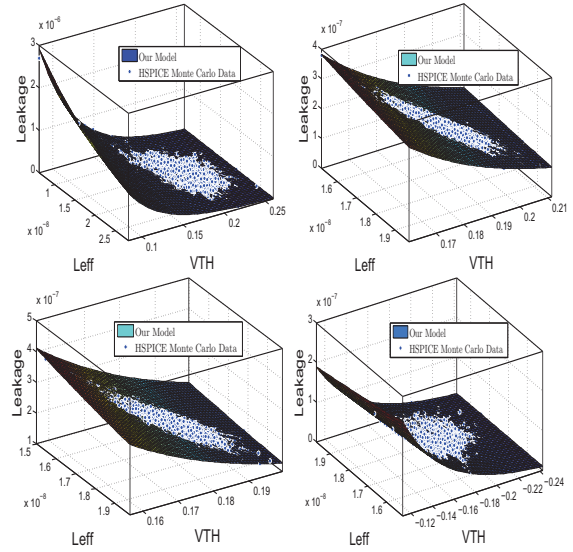

Fig. 3. Fitting results of our current leakage model on the HSPICE simulation data. The four figures, from top-left to bottom-right, correspond to 00, 01, 10, and 11 input sets

15k standard cells and occupied a die area of $200 \times 200 \mu m^2$. The same process variation values described in Table I were used and a set of 900 variation maps were produced using the method described in Section IV 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 1.72 seconds on a 1.73GHz Intel Core 2 Duo machine with a 2.00GB RAM and running a 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 1700 seconds (28 minutes) for a typical 15 million-cell design which demonstrates the practicality of its usage in real world applications.

C. Comparison with Previous Methods

In this subsection, we will present a comparison of our framework with previously proposed methods in terms of accuracy and performance. Most statistical approaches to leakage power variability analysis use the well-known Wilkinson's Method as the most effective method for approximating 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. Our approach is fundamentally different, in that it forms the desired leakage distribution by point-by-
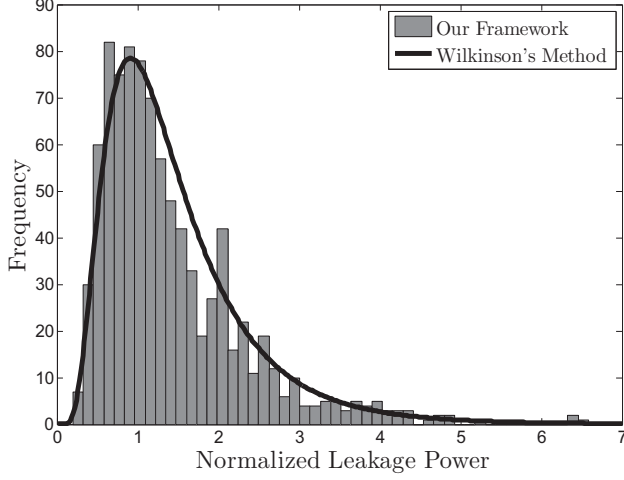
Fig. 4. Leakage Power Distribution of OpenRISC Processor Core Using Wilkinson's Method and Our Framework

point mathematical summation of cell leakage values. *Our Monte Carlo approach ensures that our method is superior in terms of accuracy assuming both methods use the same cell leakage models. 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. 4 shows the final normalized leakage distributions derived from our framework and Wilkinson's Method. While the mean and SD values estimated by Wilkinson's Method are relatively accurate, with respectively 2.1% and 3.4% 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 of our OpenRISC design estimated by both methods. Table IV shows these values and their difference for various leakage power constraints. The contraints in the first column are expressed in relation to the nominal, variation-free leakage power estimate. 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. 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).

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

$$
\begin{aligned}
\mu_1 &= E[e_1^Y + ... + e_n^Y] = e^{\mu_Z + \sigma_Z^2/2} \\
&= \sum_{i=1}^{n} e^{\mu_{Yi} + \sigma_{Yi}^2/2}
\end{aligned}
$$

$$
\begin{aligned}
\mu_2 &= E[(e_1^Y + ... + e_n^Y)^2] = e^{2\mu_Z + 2\sigma_Z^2} \\
&= \sum_{i=1}^{n} e^{2\mu_{Yi} + 2\sigma_{Yi}^2} \\
&\quad + 2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} e^{\mu_{Yi} + \mu_{Yj} + (\sigma_{Yi}^2 + \sigma_{Yj}^2 + 2\rho_{ij}\sigma_i\sigma_j)/2}
\end{aligned}
$$

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

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

$$\sigma_Z^2 = ln\mu_2 - 2ln\mu_1$$

This approach reduces the error in the mean and SD values of the approximated distribution. However, as indicated by our results in Table IV, the assumption that the final distribution follows a log-normal pattern 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.

In terms of performance, the computation complexity of our framework and Wilkinson's Method are $O(N)$ and $O(N^2)$ where N is the number of cells in the design. Of course, there have been several attempts to reduce this complexity. Chang and Sapatnekar [4] 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. Kim et. al. [21] 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 IV. As expected, the VCA approach's error rate increases with the amount of process variability [21].

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.

V. CONCLUSION AND FUTURE WORK

We presented a variation-aware leakage power estimation method which is used as a part of our variability modeling framework for analyzing the effect of process variation on power consumption and performance of Multiprocessor SoCs. We developed a subthreshold leakage current model

Constraint	Wilkinson's Method	Our Framework	Difference
1.1	41%	45%	4%
1.2	46%	53%	7%
1.3	52%	59%	7%
1.4	57%	65%	8%
1.5	62%	69%	7%
Average	-	-	6.6%

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 sub-micron 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 maintain high accuracy while keeping an acceptable timing penalty imposed by the variability analysis procedures. We plan to include a better estimation of a cell leakage using a more accurate prediction of its input state. This can be done by propagating state probabilities of the nodes through the entire circuit. Integration of other process parameters such as oxide thickness in our subthreshold leakage model is also straightforward and may be used to achieve more accurate estimations.

ACKNOWLEDGEMENT

REFERENCES

[1] (2007) Semiconductor industry association, international technology roadmap for semiconductors. [Online]. Available: http://www.itrs.net/
[2] A. Chandrakasan, W. Bowhill, and F. Fox, Design of high-performance micro-processor circuits. IEEE press, 2001.
[3] O. Assare, H. Izady Rad, M. Momtazpour, E. Sanaei, and M. Goudarzi, "VAREX: A Post-P&R Variability Modeling Framework for Multiprocessor SoCs," IEEE/ACM ICCAD Workshop on Variability Modeling and Characterization (VMC), San Jose, CA, November 2011.
[4] H. Chang and S. S. Sapatnekar, Full-chip analysis of leakage power under process variations, including spatial correlations, in Proc. Des. Autom. Conf., 2005, pp. 523528.
[5] S. Chandra, K. Lahiri, A. Raghunathan, and S. Dey, "Considering process variations during system-level power analysis," Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED), pp. 342-345, Oct. 2006.
[6] R. Teodorescu and J. Torrellas, "Variation-aware application scheduling and power management for chip multiprocessors," 35th International Symposium on Computer Architecture (ISCA), pp. 363-374, June 2008.
[7] S. R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas, "VARIUS: A model of process variation and resulting timing errors for microarchitects," in IEEE Transactions on Semiconductor Manufacturing, February 2008.
[8] T. Karnik, S. Borkar, and V. De, "Probabilistic and variation-tolerant design: Key to continued Moores law," ACM/IEEE TAU Workshop on Timing Issues in the Specification and Synthesis of Digital Systems, Feb. 2004.
[9] P. Ribeiro and P. Diggle, The geoR Package [Online]. Available: http://www.est.ufpr.br/geoR

[10] R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2007.
[11] The NanGate 45nm Open Cell Library: An Open source standard cell library. [Online]. Available: http://www.nangate.com/
[12] S. Mukhopadhyay and K. Roy, "Modeling and estimation of total leakage current in nano-scaled CMOS devices considering the effect of parameter variation," in Proc. Int. Symp. Low Power Electron. Des., 2003, pp. 172-175.
[13] S. Narendra, V. De, S. Borkar, D. Antoniadis, and A. Chandrakasan, "Full-chip sub-threshold leakage power prediction model for sub-0.18 in CMOS," in Proc. Int. Symp. Low Power Electron. Des., 2002, pp. 19-23.
[14] A. Agarwal, K. Kunhyuk, and K. Roy, "Accurate estimation and modeling of total chip leakage considering inter- & intra-die process variations," in Proc. Int. Conf. Comput.-Aided Des., 2005, pp. 736-742.
[15] H. F. Dadgour, S. Lin, and K. Banerjee, "A Statistical Framework for Estimation of Full-Chip Leakage-Power Distribution Under Parameter Variations," IEEE Transactions on Electron Devices, Vol 54, No. 11, pp. 2930-2945, Nov. 2007.
[16] N. C. Beaulieu, A. A. Abu-Dayya, and P. J. McLane, "Comparison of methods of computing lognormal sum distributions and outages for digital wireless applications," in Proc. IEEE Int. Conf. Commun., 1994, pp. 1270-1275.
[17] R. Chau, S. Datta, M. Doczy, J. Kavalieros, and M. Metz. "Gate dielectric scaling for high-performance CMOS: from SiO2 to High- K," In Extended Abstracts of International Workshop on Gate Insulator (IWGI), Nov. 2003, pp. 124-126.
[18] OpenRISC 1200, [Online]. Available: http://opencores.org/openrisc,or1200.
[19] R. X. Gu and M. I. Elmasry: "Power Dissipation Analysis and Optimization of Deep Submicron CMOS Digital Circuits," IEEE J. Solid-State Circuits, May 1996, vol. 31, pp. 707-713
[20] R. Rao, A. Sirvastava, D. Blaauw, and D. Sylvester, "Statistical Analysis of Sub-threshold Leakage Current for VLSI Circuits", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, VOL. 12, NO. 2, February 2004.
[21] W. Kim, K. T. Do, and Y. H. Kim, "Statistical Leakage Estimation Based on Sequential Addition of Cell Leakage Currents," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 18, No. 4, Apr. 2010, pp. 602-615.
[22] W. Liu, K. M. Cao, X. Jin, X. Xi, and C. Hu, BSIM4.2.0 MOSFET Model - User Manual, http://www-device.eecs.berkeley.edu/bsim3/ bsim4.html.