VARIUS: A Model of Parameter Variation and Resulting Timing Errors for Microarchitects *

Radu Teodorescu, **Brian Greskamp**, **Jun Nakano**,
Smruti R. Sarangi, **Abhishek Tiwari** and **Josep Torrellas**
University of Illinois at Urbana-Champaign
http://iacoma.cs.uiuc.edu

ABSTRACT

As VLSI technology continues to scale, parameter variation is about to pose a major challenge to high-performance processor design. In particular, within-die variation is directly detrimental to a processor's frequency and leakage power.

To gain an understanding of this problem, this paper starts out by proposing a microarchitecture-aware model for parameter variation. It includes both random and systematic effects. It is partially calibrated with empirical data and uses a few intuitive parameters. Then, we extend the framework to model timing errors caused by parameter variation. This model yields the failure rate of microarchitectural blocks as a function of frequency and the amount of variation. With the combination of the variation model and the error model, we have *VARIUS*, a comprehensive model that is capable of producing detailed statistics of timing errors as a function of different process parameters and operating conditions. We propose possible applications of VARIUS to microarchitectural research.

1. Introduction

As high-performance processors move into 32 nm technologies and below, designers face the major roadblock of parameter variation — the deviation of process, voltage, and temperature (PVT [2]) values from nominal specifications. Variation makes designing processors harder because they have to work under a range of parameter values.

Variation is induced by several fundamental effects. Process variation is caused by the inability to precisely control the fabrication process at small-feature technologies. It is a combination of systematic effects [10, 23, 30] (e.g., lithographic lens aberrations) and random effects [1] (e.g., dopant density fluctuations). Voltage variations can be caused by *IR* drops in the supply distribution network or by *L dI/dt* noise under changing load. Temperature variation is caused by spatially- and temporally-varying factors. All of these variations are becoming more severe and harder to tolerate as technology scales to minute feature sizes.

Two key process parameters subject to variation are the transistor threshold voltage, V_{th}, and the effective length, L_{eff}. V_{th} is especially important because its variation has a substantial impact on two major properties of the processor, namely the frequency it attains and the leakage power it dissipates. Moreover, V_{th} is also a strong function of temperature, which increases its variability [31].

One of the most harmful effects of variation is that some sections of the chip are slower than others — either because their transistors are intrinsically slower or because high temperature or low supply voltage renders them so. As a result, circuits in these sections may be unable to propagate signals fast enough and may suffer timing errors. To avoid these errors, designers in upcoming technology generations may slow down the frequency of the processor or create overly conservative designs. It has been suggested that parameter variation may wipe out most of the potential gains provided by one technology generation [3].

An important first step to redress this trend is to understand how parameter variation affects timing errors in high-performance processors. Based on this, we could devise techniques to cope with the problem — hopefully recouping the full gains offered by every technology generation.

To address these problems, this paper proposes *VARIUS*, a novel microarchitecture-aware model for parameter variation and the resulting timing errors. VARIUS can be used by microarchitects in a variety of studies.

1.1. Contributions

A model for parameter variation. We propose a novel model that uses multivariate analysis to model parameter variation. To the best of our knowledge, we are the first to use a Spherical correlation structure for systematic variation with this model. This matches the empirical data obtained by Friedberg *et al.* [10] well. Moreover, our model takes into account temperature variation. Finally, it has only three parameters — all highly intuitive — and is easy to use.

A model for timing errors due to parameter variation. We propose a novel, comprehensive timing error model for microarchitectural structures in dies that suffer from process variation. This model is called *VATS*. It takes into account process parameters, the floorplan, and operating conditions like temperature. We model the error rate in logic structures, SRAM structures and combinations of both, and consider both systematic and random variation. Moreover, our model matches empirical data and can be simulated at high speed.

The rest of the paper is organized as follows. We present the model for parameter variation in Section 2 and show the implications on frequency and leakage power in Section 3. We present the model for timing errors in Section 4, discuss the implications of both models in Section 5, present related work in Section 6, and conclude in Section 7.

*This work was supported in part by the National Science Foundation under grants EIA-0072102, EIA-0103610, CHE-0121357, and CCR-0325603; DARPA under grant NBCH30390004; DOE under grant B347886; and gifts from IBM and Intel. Jun Nakano is now with IBM Japan. Smruti R. Sarangi is now with Synopsys, India.

2. Modeling Variation

Parameter variation can be broken down into two major components, namely die-to-die (D2D) and within-die (WID). Furthermore, WID variation can be divided into random and systematic components. Thus, variation in any parameter P, like V_{th} or L_{eff}, can be represented as follows:

$$\Delta P = \Delta P_{D2D} + \Delta P_{WID} = \Delta P_{D2D} + \Delta P_{rand} + \Delta P_{sys}$$

In this work, we focus on WID variation, but D2D variation is easily modeled: One needs only add a chip-wide offset to the V_{th} and L_{eff} parameters of every transistor on the die. For simplicity, we model the two components of WID process variation with normal distributions. This is an accepted approach [12, 29].

From a microarchitectural perspective, V_{th} and L_{eff} variation are of key importance: they directly affect a chip's leakage and frequency. The WID variation of these parameters is impacted by both systematic and random effects [1]. Limitations of the lithography and other manufacturing processes introduce systematic variations. Typically, such variations exhibit a spatial structure with a certain scale of parameter changes over the two-dimensional space [10, 23, 30]. On the other hand, a variety of materials effects, such as changes in the dopant density of the channel [1] and lithographic phenomena like line edge roughness [35], introduce random variations. Such random variations have a different profile for each transistor and are in effect noise superimposed on the systematic variation.

We treat random and systematic variation separately, since they arise from different physical phenomena. As described in [29], we assume that their effects are additive.

2.1. Systematic Variation

Systematic variation is characterized by a spatial correlation, meaning that adjacent areas on a chip have roughly the same systematic components. Such correlation can be characterized using different models. For example, [19, 29] use a quad tree model that recursively partitions the die into four parts. In this paper, we use a different method that models systematic variation using a multivariate [24] normal distribution with a specific correlation structure.

We divide a chip into N small rectangular cells. The value of the systematic component of V_{th} is assumed to be constant within one small cell. This is consistent with other work [29]. We also assume that the value of V_{th} for all the cells has a normal distribution with mean μ and standard deviation σ. Along with this, the values of V_{th} are spatially correlated.

To determine the spatial correlation, we make the following assumptions. First, we treat the distribution of V_{th} as isotropic and position-independent. This means that given two points $\vec{x}$ and $\vec{y}$ in the grid, the correlation between them depends only on the distance between $\vec{x}$ and $\vec{y}$, and not on the direction of the segment that goes from $\vec{x}$ to $\vec{y}$, or the position of $\vec{x}$ and $\vec{y}$ in the grid. We verify these assumptions by analyzing the empirical data obtained by Friedberg *et al.* [10] and using results from [29]. Nevertheless, we acknowledge the fact that in reality there are some anisotropic effects — for example in defects due to misalignment of the masks.

Given the assumptions of position independence and isotropy, the correlation function of $V_{\text{th}}(\vec{x})$ and $V_{\text{th}}(\vec{y})$ is expressible as $\rho(r)$, where $r = |\vec{x} - \vec{y}|$. By definition, $\rho(0) = 1$ (i.e., totally correlated). We also set $\rho(\infty) = 0$ (i.e., totally uncorrelated) because two infinitely separated points have independent V_{th} when we only consider WID variation.

To determine how $\rho(r)$ changes from $\rho(0) = 1$ to $\rho(\infty) = 0$ as r increases, we use the Spherical model [7, 14], which has the following form:

$$\rho(r) = \begin{cases} 1 - (3r/2\phi) + (r/\phi)^3/2 & \text{if (r} \leq \phi) \\ 0 & \text{if (r} > \phi) \end{cases} \quad (1)$$

This model is very similar to the correlation function experimentally measured by Friedberg *et al.* [10] for the WID variation of gate length. Our rationale for using this model is that gate length variation is the main determinant of systematic V_{th} variation.

Figure 1 shows the function $\rho(r)$. At a finite distance ϕ that we call *range*, the function converges to zero. Intuitively, this assumption implies that the V_{th} of a transistor is highly correlated to the V_{th} of those in its immediate vicinity. The correlation decreases linearly with distance at small distances. Then, it decreases more slowly. At distance ϕ, there is no longer any correlation between two transistors' V_{th}.

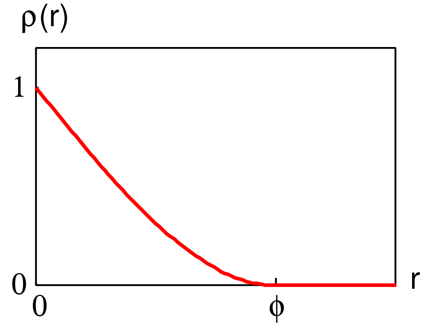

Figure 1. Correlation between the V_{th} at two points as a function of the distance r between them.

In this paper, we express ϕ as a fraction of the chip's width. A large ϕ implies that large sections of the chip are correlated with each other; the opposite is true for small ϕ. As an illustration, Figure 2 shows example systematic V_{th} variation maps for chips with $\phi = 0.1$ and $\phi = 0.5$. Both maps were generated by the geoR statistical package [26] of R [25]. In the $\phi = 0.5$ case, we discern large spatial features, whereas in the $\phi = 0.1$ one, the features are small. A distribution without any correlation ($\phi = 0$) appears as white noise.

Finally, to estimate the systematic component of L_{eff}, we proceed as follows. The ITRS report [13] tells us that the total σ/μ of L_{eff} is roughly half of that of V_{th}. Moreover, according to [4], the systematic component of L_{eff} is strongly correlated with the systematic component of V_{th}. Hence, we use the following equation to generate a value of the systematic component of L_{eff} given the value of the systematic component of V_{th}. Let L_{eff0} be the nominal value of the effective length and let V_{th0} be the nominal value of the threshold voltage. We use:

$$L_{\text{eff}} = L_{\text{eff0}} \left(1 + \frac{1}{2}(V_{\text{th}} - V_{\text{th0}})/V_{\text{th0}}\right) \quad (2)$$

2.2. Random Variation

The random variation occurs at a much finer granularity than the systematic variation; it occurs at the level of individual transistors, rather than at the level of millions of transistors. Hence, it is not possible to model random variation in the same explicit way as systematic variation — by simulating a grid where each cell has its own parameter values. Instead, random variation appears in the model analytically.

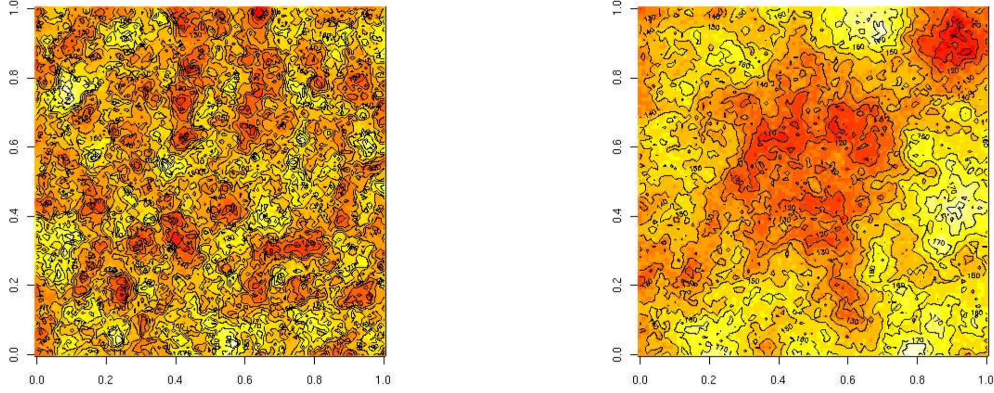

Figure 2. Systematic V_{th} variation maps for a chip with $\phi = 0.1$ (left) and $\phi = 0.5$ (right).

Random components of V_{th} and L_{eff} are normally distributed with a σ_{rand} and a zero mean.

2.3. Combining Variations

To combine systematic and random components, we use the following equations:

$$\mu_{total} = \mu_{rand} + \mu_{sys} \tag{3}$$

$$\sigma_{total} = \sqrt{\sigma^2_{rand} + \sigma^2_{sys}} \tag{4}$$

Since the random and systematic components are assumed to be normal distributions, the combination of them is also normal. This applies to both V_{th} and L_{eff}.

2.4. Values for μ, σ and ϕ

For V_{th}, we set $\sigma/\mu = 9\%$. This is consistent with near-future technologies and includes both the systematic and random components. Moreover, according to empirical data gathered by [17], these two components are approximately equal for 32 nm technology. Hence, we assume that they have equal variances. Since both components are modeled as normal distributions, their standard deviations σ_{rand} and σ_{sys} are equal to $9\%/\sqrt{2} = 6.4\%$ of the mean. This value for the random component matches the empirical data of Keshavarzi *et al.* [18].

As explained before, we take the total σ/μ of L_{eff} to be half of that of V_{th}. Consequently, L_{eff}'s σ/μ is 4.5%. Furthermore, assuming again that the two components of variation are more or less equal, we have that σ_{rand} and σ_{sys} for L_{eff} are equal to $4.5\%/\sqrt{2} = 3.2\%$ of the mean.

To estimate ϕ, we note that Friedberg *et al.* [10] experimentally measured the correlation of gate length to be around half of the chip length. The rest of this paper also adopts $\phi = 0.5$, but depending on how ϕ scales with die size, larger values may be appropriate for smaller dies.

3. Impact on Chip-Level Behavior

To evaluate the impact of variation on a chip's behavior, we look at two key characteristics: chip leakage and chip frequency.

3.1. Leakage Power

Subthreshold leakage is the main source of leakage in current and future technologies, especially now that the accelerated adoption of high-k gate dielectric is set to reduce gate leakage 100-fold [5]. The following subthreshold leakage model is based on that of HotLeakage [36], itself a simplification of the full BSIM3 SPICE model:

$$I_{\text{leak}} \propto (kT/q)^2 e^{\frac{q(V_{off} - V_{th})}{(\eta k T)}} \tag{5}$$

where $V_{\text{th}} = Tc_1 + c_2$, k is Boltzmann's constant, and q the electron charge, while c_1, c_2, η and V_{off} are empirically determined parameters. We find the value for these parameters by fitting the leakage Equation 5 to experimental data for the 32 nm technology node obtained from SPICE simulations using the Predictive Technology Model [37].

In order to estimate the impact of different levels of V_{th} variance on the chip's leakage power, we take our V_{th} distribution and integrate Equation 5 over all the transistors in the chip. The result is the total leakage current in the chip. Let P_{leak} and I_{leak} be the chip leakage power and current under V_{th} variation, and P^0_{leak} and I^0_{leak} be the same parameters when there is no variation. The expected value of the ratio of post-variation and pre-variation leakage is:

$$P_{\text{leak}}/P^0_{\text{leak}} = I_{\text{leak}}/I^0_{\text{leak}} = e^{(q\sigma/\eta k T)^2/2} \tag{6}$$

which implies that the increase in the chip's leakage power and current due to V_{th} variation depends on the standard deviation σ of V_{th}. Figure 3 plots the relative power as a function of σ. It increases rapidly as σ goes up.

Another important factor affecting leakage power is temperature. Figure 4 shows how the relative leakage power changes as a function of temperature, for different threshold voltages at 100 $^{\text{o}}$C. Leakage power increases dramatically with temperature (3X from 50 $^{\text{o}}$C to 100 $^{\text{o}}$C). In addition, we observe that the leakage dependence on the threshold voltage is significant. For different V_{th} (different lines in Figure 4), the leakage changes significantly.

3.2. Chip Frequency

The delay of an inverter gate is given by the alpha-power model [27] as:

$$T_g \propto \frac{L_{\text{eff}} V}{\mu (V - V_{\text{th}})^{\alpha}} \tag{7}$$

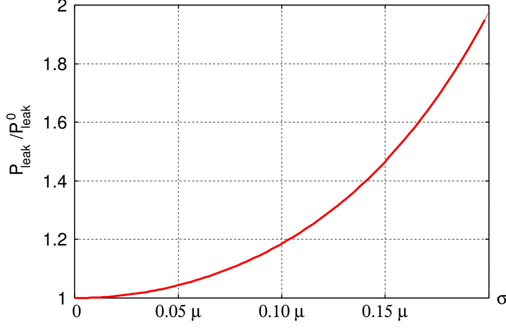

Figure 3. Relative leakage power in the chip as a function of V_{th}'s σ. V_{th0} is 0.150V at 100 °C.

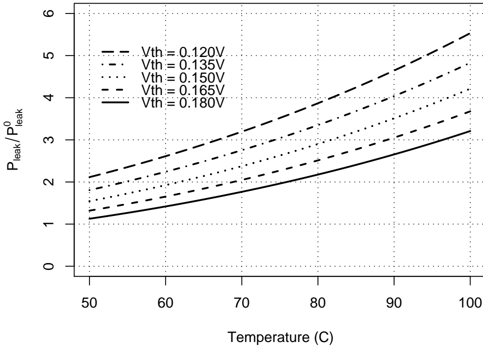

Figure 4. Relative leakage power versus temperature for different threshold voltages at 100 °C. We use V_{th0}=0.150V at 100 °C.

where α is typically 1.3 and μ is the mobility of carriers ($\mu(T) \propto T^{-1.5}$). As V_{th} decreases, $V - V_{\text{th}}$ increases and the gate becomes faster. As T increases, $V - V_{\text{th}}(T)$ increases, but $\mu(T)$ decreases [15]. The second factor dominates and, with higher T, the gate becomes slower. Figure 5 plots the dependence between relative switching frequency and temperature as dictated by Equation 7. We can see that the dependence is not very strong.

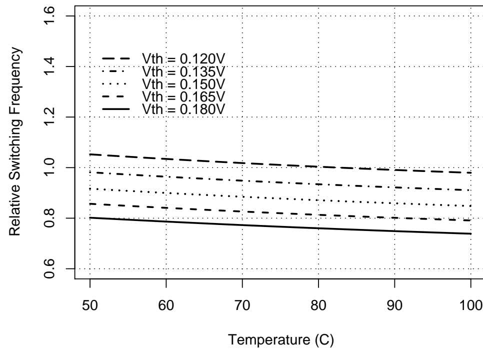

Figure 5. Relative switching frequency versus temperature for different threshold voltages at 100 °C. We use V_{th0}=0.150V at 100 °C.

Consider now a fixed temperature. Substituting Equation 2 into Equation 7 and factoring out constants with respect to V_{th} produces:

$$T_g \propto \frac{1 + V_{\text{th}}/V_{\text{th0}}}{(V - V_{\text{th}})^{\alpha}} \quad (8)$$

Empirically, we find that Equation 8 is nearly linear with respect to V_{th} for the parameter range of interest. Because V_{th} is normally distributed and a linear function of a normal variable is itself normal, T_g is approximately normal.

Assuming that every critical path in a processor consists of n_{cp} gates, and that a modern processor chip has thousands of critical paths, Bowman *et al.* [3] compute the probability distribution of the longest critical path delay in the chip ($\max\{T_{\text{cp}}\}$). Such path determines the processor frequency ($1/\max\{T_{\text{cp}}\}$). Using this approach, we find that the value of V_{th}'s σ affects the chip frequency.

Figure 6 shows the probability distribution of the chip frequency for different values of V_{th}'s σ. The frequency is given relative to a processor without V_{th} variation (F/F_0). The figure shows that, as σ increases, (i) the mean chip frequency decreases and (ii) the chip frequency distribution gets more spread out. In other words, given a batch of chips, as V_{th}'s σ increases, the mean frequency of the batch decreases and, at the same time, an individual chip's frequency deviates more from the mean.

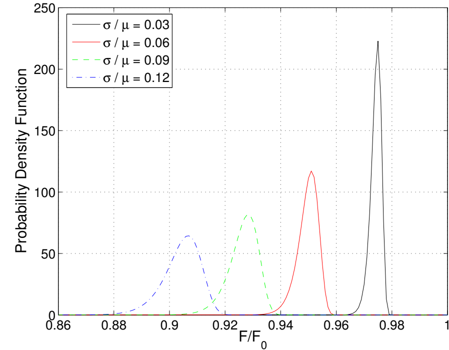

Figure 6. Probability distribution of the relative chip frequency as a function of V_{th}'s σ. We use V_{th0}=0.150V at 100 °C, 12 FO4s in the critical path, and 10,000 critical paths.

3.3. Summary of the Effects of Variation

We saw that V_{th}'s σ directly affects chip leakage and frequency. As σ increases, chip leakage increases rapidly, and chip frequency decreases in mean value and varies more. Therefore, V_{th} (and L_{eff}) variation is very detrimental.

4. VATS: Modeling Timing Errors

As we move to 32nm and below, designing processors for worst-case parameter values will be unacceptable. Instead, processors may need to be designed for closer to nominal-value parameters, inevitably resulting in some sections of the chip being too slow to meet the chip's frequency. In this case, the result will likely be timing faults due to variation-induced slow paths. In this section, we extend the parameter variation framework to model timing errors in processor pipelines due to parameter variation. We call the model VATS. In the following, we first describe our assumptions, then model errors in logic and in SRAM memory, and finally present an empirical validation of the model.

4.1. General Approach

A pipeline stage typically has a multitude of paths, each one with its own time slack — possibly dependent on the input data values. In our analysis, we make two simplifying assumptions.

Assumption 1: A path causes a timing fault if and only if it is exercised and its delay exceeds the clock period. Note that this fault definition does not account for any architectural masking effects. However, architectural vulnerability factors (AVFs) could be applied to model these masking effects if desired.

Assumption 2: A pipeline stage is tightly designed. This means that, in the absence of process variation, there is at least one path whose delay for a certain input data value equals the clock period.

In the following, path delay is normalized by expressing it as a fraction t_R of the pre-variation clock period t_0. Let us first examine the probability density function (pdf) of the normalized path delays in a pipeline stage. Figure 7(a) shows an example pdf before variation effects. The right tail abuts the $X = 1$ abscissa and there are no timing errors.

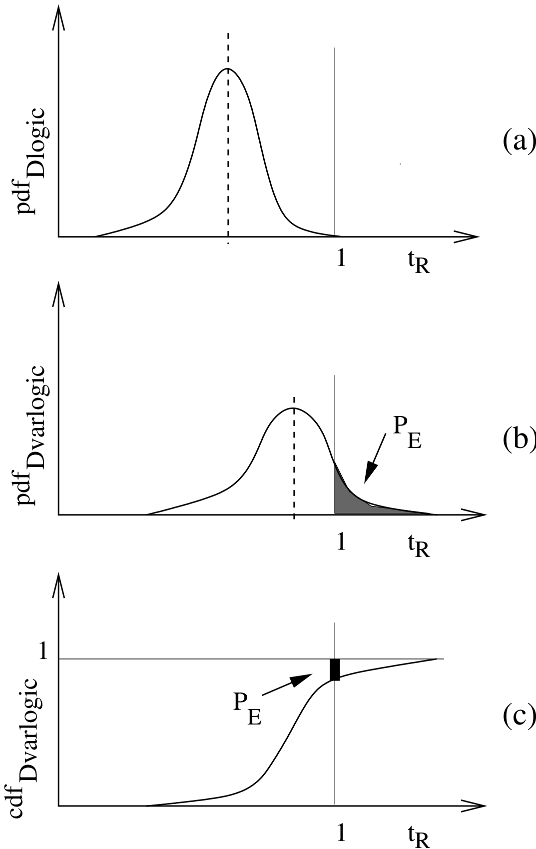

Figure 7. Example probability distributions.

As the pipeline stage paths suffer parameter variation, the pdf changes shape: the curve may change its average value and its spread (e.g., Figure 7(b)). All the paths that have become longer than 1 generate errors. Our model estimates the probability of error at a given clock period ($P_E(t_R)$) as the area of the shaded region in the figure. The same error probability can be obtained by generating the cumulative distribution function (cdf) of the distribution, and observing that:

$$P_E(t_R) = 1 - cdf(t_R). \tag{9}$$

For example, Figure 7(c) shows the cdf of Figure 7(b), and the thick segment is $P_E(t_R)$ at t_R=1. The cdf approach of Equation 9 allows for fast evaluation of the error probability at a variety of frequencies.

4.2. Timing Errors in Logic

We start by considering a pipeline stage of only logic. We represent the logic path delay in the absence of variation as a random variable D_{logic}, which is distributed in a way similar to Figure 7(a). Such delay is composed of both wire and gate delay. For simplicity, we assume that wire accounts for a fixed fraction k_w of total delay. This assumption has been made elsewhere [12]. Consequently, we can write:

$$\begin{aligned} D_{logic} &= D_{wire} + D_{gates} \\ D_{wire} &= k_w \, D_{logic} \\ D_{gates} &= (1 - k_w) \, D_{logic} \end{aligned} \tag{10}$$

We now consider the effects of variation. Since variation typically has a very small effect on wires, we only consider the variation of D_{gates}, which has a random and a systematic component. For each path, we divide the systematic variation component (ΔD_{gates_sys}) into two terms: (i) the average value of it for all the paths *in the stage* ($\overline{\Delta D_{gates_sys}}$) — which we call the stage systematic mean — and (ii) the rest ($\Delta D_{gates_sys} - \overline{\Delta D_{gates_sys}}$) — which we call intra-stage systematic deviation.

Given the high degree of spatial correlation in process (*P*) and temperature (*T*) variation, and the small size of a pipeline stage, the intra-stage systematic deviation is small. In Section 2.4, we suggested a value of ϕ equal to 0.5 (half of the chip length). On the other hand, the length of a pipeline stage is less than 0.1 of the length of, say, a typical 4-core chip. Therefore, given that the stage dimensions are significantly smaller than ϕ, the transistors in a pipeline stage have highly-correlated systematic V_{th} values and systematic L_{eff} values. Using Monte Carlo simulations with the parameters of Section 2.4, we find that the intra-stage systematic deviation of D_{gates} has a $\sigma_{intrasys} \approx 0.004 \times \mu$, while the variation of $\overline{\Delta D_{gates_sys}}$ across the pipeline stages of the processor has a $\sigma_{intersys} \approx 0.05 \times \mu$. Similarly, T varies much more across stages than within them.

The random component of D_{gates}'s variation is estimated from the fact that we model a path as n FO4 gates connected with short wires. Each gate's random component is independent. Consequently, for the whole n-gate path, D_{gates}'s σ_{rand} is $\sqrt{n} \times \sigma_{rand_D_{FO4}}$, where D_{FO4} is the delay of one FO4 gate. If we take $n = 12$ as representative of high-end processors, the overall variation is small. It can be shown that D_{gates}'s $\sigma_{rand} \approx 0.01 \times \mu$. Finally, *T* has no random component.

We can now generate the distribution of D_{logic} with variation (which we call $D_{varlogic}$ and show in Figure 7(b)) as follows. We model the contribution of $\overline{\Delta D_{gates_sys}}$ in the stage as a factor η that multiplies D_{gates}. This factor is the average increase in gate delay across all the paths in the stage due to systematic variation. Without variation, $\eta = 1$.

We model the contribution of the intra-stage systematic deviation and of the random variations as D_{extra}, a small additive normal delay perturbation. Since D_{extra} combines D_{gates}'s intra-stage systematic and random effects, $\sigma_{extra} = \sqrt{\sigma^2_{intrasys} + \sigma^2_{rand}}$. For our parameters, $\sigma_{extra} \approx 0.011 \times \mu$. Like η, D_{extra} should multiply D_{gates} as shown in Equation 11. However, to simplify the computation and because D_{logic} is clustered at values close to one, we prefer to approximate D_{extra} as an additive term as in Equation 12:

$$D_{varlogic} = (\eta + D_{extra})\, D_{gates} + D_{wire} \quad (11)$$

$$\approx (1-k_w)\,(\eta\, D_{logic} + D_{extra}) + k_w\, D_{logic} \quad (12)$$

Once we have the $D_{varlogic}$ distribution, we numerically integrate it to obtain its $cdf_{D_{varlogic}}$ (Figure 7(c)). Then, the estimated error rate P_E of the stage cycling with a relative clock period t_R is:

$$P_E(t_R) = 1 - cdf_{D_{varlogic}}(t_R) \quad (13)$$

4.2.1. How to use the model

To apply Equation 12, we must calculate k_w, η, D_{extra}, and D_{logic} for the prevailing variation conditions. To do this, we produce a gridded spatial map of process variation using the model in Section 2.1 and superimpose it on a high-performance processor floorplan. For each pipeline stage, we compute η from the pipeline stage's *T* and the systematic L_{eff} and V_{th} maps. Moreover, by subtracting the resulting mean delay of the stage from the individual delays in the grid points inside the stage, we produce the intra-stage systematic deviation. We combine the latter distribution with the effect of the random process variation to obtain the D_{extra} distribution. D_{extra} is assumed normal.

Ideally, we would obtain a per-stage k_w and D_{logic} through timing analysis of each stage. For our evaluation in this paper, we assume that the LF adder in [11] is representative of processor logic stages, and set $k_w = 0.35$ [12]. Additionally, we derive $pdf_{D_{logic}}$ using experimental data from Ernst *et al.* [9] — as explained in detail in [28]. With this data, we find that the resulting distribution of D_{logic} (normalized to the nominal access time in a chip with no variation) has $\sigma = 0.019$ and $\mu = 0.84$. For simplicity, we model D_{logic} as a normal distribution.

Strictly speaking, this $pdf_{D_{logic}}$ curve only applies to the circuit and conditions measured in [9]. To generate $pdf_{D_{logic}}$ for a different stage with a different technology and workload characteristics, one would need to use timing analysis tools on that particular stage. In practice, Section 4.5 shows empirical evidence that this method produces $pdf_{D_{logic}}$ curves that are usable under a range of conditions, not just those under which they were measured.

Finally, since D_{logic} and D_{extra} are normally distributed, $D_{varlogic}$ in Equation 12 is also normally distributed.

4.3. Timing Errors in SRAM Memory

To model variation-induced timing errors in SRAM memory, we build on the work of Mukhopadhyay *et al.* [22]. They consider *random* V_{th} variation only and describe four failures in the SRAM cell of Figure 8: Read failure, where the contents of a cell are destroyed after the cell is read; Write failure, where a write is unable to flip the cell; Hold failure, where a cell loses its state; and Access failure, where the time needed to access the cell is too long, leading to failure. The authors provide analytical equations for these failures, which show that, for the standard deviations of V_{th} considered here, Access failures dominate[1].

[1]While [22] only considers the random component of V_{th} variation, it can be shown that the previous statement also holds for the combination of random and systematic components, and for the variation of both V_{th} and L_{eff}.

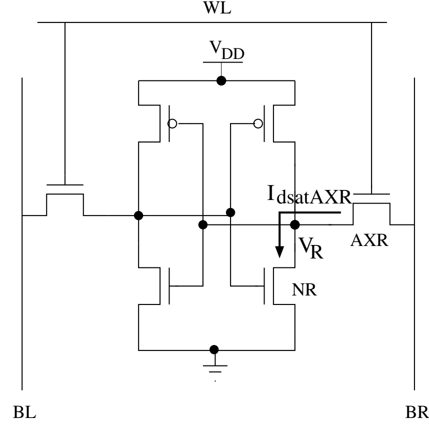

Figure 8. A 6-transistor SRAM cell.

Consequently, we focus on modeling Access errors only. According to [22], the cell access time under variation on a read is:

$$T_{varacc} \propto \frac{1}{I_{dsatAXR}} = h(V_{thAXR}, V_{thNR}, L_{AXR}, L_{NR}) \quad (14)$$

where V_{thAXR} and L_{AXR} are the V_{th} and L_{eff} of the AXR access transistor in Figure 8, and V_{thNR} and L_{NR} are the same parameters for the NR pull-down transistor in Figure 8.

To determine $I_{dsatAXR}$ and the h function, we use Kirchoff's current law to equate the currents through the AXR and NR transistors. The alpha-power model [27] is used for the individual transistor currents. It is then trivial to solve for the operating point and find $I_{dsatAXR}$ and, subsequently, h — as detailed in [28].

Using multivariable Taylor expansion [24] of h, the mean $\mu_{Tvaracc}$ and variance $\sigma_{Tvaracc}$ of T_{varacc} can be put as a function of the μ and σ of each of V_{thAXR}, V_{thNR}, L_{AXR}, and L_{NR}. T_{varacc} is assumed normally distributed.

In reality, an SRAM array access does not read only one cell at a time, but a line — e.g., 8-1024 cells. Consequently, we need to compute the distribution of the maximum access time of all the cells in a line. There is no exact analytical solution for the distribution of the maximum of n normally distributed variables, but we use an approximation from [6]. The resulting distribution has mean $\mu_{vararray}$ and standard deviation $\sigma_{vararray}$.

Finally, the access to the memory array itself takes only a fraction k of the whole pipeline cycle — the rest is taken by logic structures such as sense amplifiers, decoders, and comparators. Such logic delays are modeled according to Section 4.2. Consequently, the total path delay with variation D_{varmem} is the sum of the normal distributions of the delays in the line access and in the logic. It is distributed normally with:

$$\mu_{varmem} = k\, \mu_{vararray} + (1-k)\, \mu_{varlogic}$$

$$\sigma_{varmem} = \sqrt{k^2\, \sigma^2_{vararray} + (1-k)^2\, \sigma^2_{varlogic}}$$

Then, the estimated error rate of a memory stage cycling with a relative clock period t_R is:

$$P_E(t_R) = 1 - cdf_{D_{varmem}}(t_R) \quad (15)$$

4.4. Error Rate per Instruction

We model an n-stage pipeline as a series failure system, where each stage i can fail independently. Now, each stage has an activity factor α_i, which is the number of times that the average instruction exercises the stage. Hence, the error rate per instruction as a function of the relative clock period t_R is:

$$P_E(t_R) = \sum_{i=1}^{n} (\alpha_i \times P_{E_i}(t_R)) \quad (16)$$

4.5. Empirical Validation

To partially validate the model of timing errors, we use it to explain some error rate data obtained empirically elsewhere. We validate both the logic and memory model components. For the former, we use the curves obtained by Das *et al.* [8], who reduce the voltage (V) of the logic units in an Alpha-like pipeline and measure the error rate. They report curves for three different T: 45 ^{o}C, 65 ^{o}C, and 95 ^{o}C. Their curves are shown in solid pattern in Figure 9.

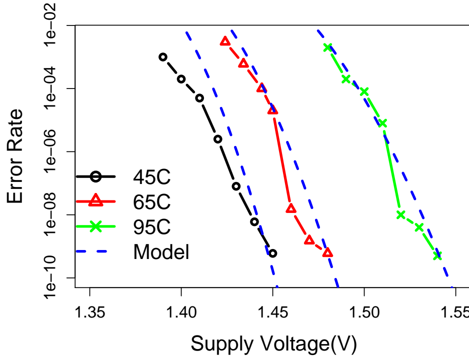

Figure 9. Validating the logic model.

To validate our model, we use the 65 ^{o}C curve to predict the other two curves. Specifically, we take the 65 ^{o}C curve and follow the procedure outlined in Section 4.2.1 (and detailed in [28]) to generate the distribution for D_{logic}. We then use D_{logic} to predict, say, the 95 ^{o}C curve as follows. First, we generate a large number of voltage values (V_i). For each V_i, we compute $\eta(V_i)$ at 95 ^{o}C using Equation 7, as the ratio of gate delay at V_i and gate delay at the minimum voltage in [8] for which no errors were detected. Since the data in [8] came from an older 180nm process with little process variation, $D_{extra} = 0$. Knowing the D_{logic} distribution, we use Equation 12 for each $\eta(V_i)$ to compute the $D_{varlogic}(V_i)$ distribution. We can then compute:

$$P(D_{varlogic}(V_i) > 1) = 1 - cdf_{D_{varlogic}(V_i)}(1) = P_E(V_i)$$

Finally, we plot the resulting pairs $(V_i, P_E(V_i))$. The resulting curves for 45 ^{o}C and 95 ^{o}C are shown in dashed lines in Figure 9. We also show the recomputed curve for 65 ^{o}C — it does not fully match the original one because we use a normal approximation for D_{logic} (Section 4.2.1). We see that the 45 ^{o}C and 95 ^{o}C curves track the experimental data closely. The small inaccuracy comes largely from the normal approximation of D_{logic}, which is assumed for simplicity.

To validate the memory model, we use data from Karl *et al.* [16]. They take a 64KB SRAM with 32-bit lines and multiple, different-latency banks, and measure the error rate as they reduce the voltage (V). The resulting curve is shown in solid pattern in Figure 10. The different-latency banks explain the steps in the curve. To re-generate the curve, we use the method of Section 4.3. The resulting curve, shown in dashes in Figure 10 is very close to the original one. Overall, these two experiments give us confidence in our model.

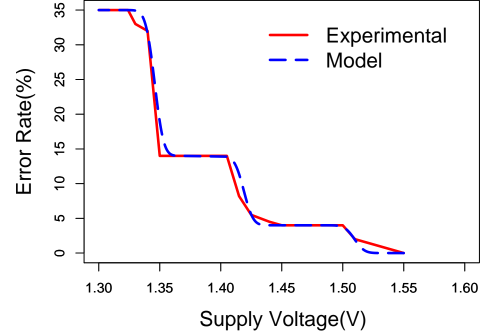

Figure 10. Validating the memory model.

5. Discussion

5.1. Applications of the Variation Model

Process variation introduces many problems. It is not possible to ignore variation unless we are overly conservative — in which case, performance suffers considerably. Hence, it is necessary to understand the implications of process variation. We list a number of potential studies that can be carried out using our variation model.

Find delay and leakage power for different microarchitectural structures. We can generate a large sample set of dies and find the average delay and leakage power of different microarchitectural structures while they are running real workloads. Our model will also take temperature effects into account.

Evaluate the effectiveness of delay and leakage reduction techniques. Using our variation model, we can evaluate the effectiveness of delay and leakage reduction techniques. We can then use the techniques that yield the maximum benefit for the largest number of chips.

Evaluate the effectiveness of architectural techniques to tolerate variation. There are some architectural techniques that help mitigate the effects of variation, such as pipeline adaptation [32], port switching [19], spare functional units [33] or using functional unit redundancy [34]. Due to the nature of process variation, not all chips can benefit from a given technique. However, we can still evaluate their average effectiveness using our computationally-inexpensive model.

5.2. Applications of the Error Model

Process variation renders traditional static timing analysis ineffective. Along with this, it forces us to deal with timing errors in microarchitectural structures. We can use our error model to achieve the

following goals.

Evaluate the error rates of different microarchitectural structures. We can evaluate the error rates for different clock frequencies for different microarchitectural structures. This will allow us to find the susceptibility of different units to timing errors.

Evaluate the effectiveness of error checkers and checkpointing mechanisms. Knowing the expected error rates, it is possible to make decisions about the nature of error checking and checkpointing. Depending on the error frequency, we may want to checkpoint different parts of the machine.

Study fault-tolerant circuit and architectural designs. We can find that some circuit techniques are amenable to a reduction in error rate, whereas others are not. Likewise, we can distinguish between different floorplans and different architectural designs based on their fault-tolerant capabilities.

5.3. Example Evaluation

As an example evaluation, we simulate a processor like the AMD Athlon 64 at 32nm technology. The pipeline has 12 stages. We divide the pipeline stages into those that are mostly logic and those that are mostly memory.

Figure 11 shows the error rate versus the frequency for the different pipeline stages. Frequency is normalized to that of a processor without process variation. The processor runs all the SpecInt and SpecFP applications in sequence. We can see that the error rate increases steeply as we increase the frequency — more and more paths fail to meet timing.

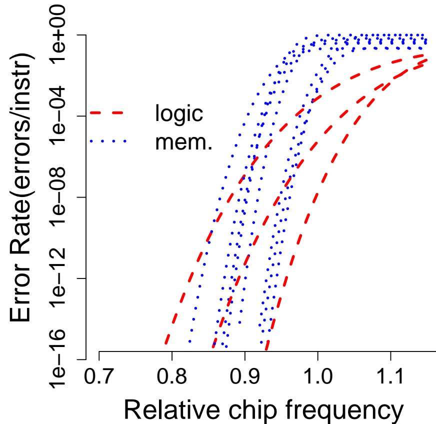

Figure 11. Error rate of the different pipeline stages for a range of normalized processor frequencies.

Our experiment enables two observations. First, we see that error rates rise very sharply. Indeed, in Figure 11, the y axis is plotted in log scale. The second observation is that the memory error curves are the steepest. This is because the critical paths in a memory stage are more homogeneous, and all have relatively similar delays.

6. Related Work

Stine *et al.* [30], Orshansky *et al.* [23], and Friedberg *et al.* [10] performed actual measurements of test chips to characterize gate length variation. They provide a lot of empirical data that we have used in our models. They observed that a significant portion of the WID variation is systematic. Cao and Clark [4] proposed a model that attributes V_{th} variation to gate length variation and studied the impact of spatial correlation on the delay of one critical path. For our V_{th} model, we mainly rely on Friedberg's data.

There are several approaches to model variation [29]. The first one includes multivariate normal methods like the one we use. One novelty of our work is that we propose a Spherical correlation matrix. In addition, we enhance the variation model with a timing error model. Two other approaches are quad-tree modeling [19] and regression-based approaches [12]. In the former, it is difficult to control aspects of the correlation structure and the distribution parameters. In regression-based approaches like [12], the model is deterministic. It models the distribution of the systematic component of just one die. It cannot be used to study a set of dies to find average statistics.

Mukhopadhyay *et al.* [22] proposed models for timing errors in SRAM memory due to random V_{th} variation. They considered several failure modes. We extended their model of Access time errors by (i) including systematic variation effects, (ii) considering variation in L_{eff} as well, and (iii) modeling the maximum access time of a *line* of SRAM, since [22] only modeled the access time of a single cell.

Memik *et al.* [20, 21] modeled errors in SRAM memory due to cross-talk noise as they overclock circuits. They used high degrees of overclocking — they doubled the nominal frequency and more. For the range of frequencies that we consider, such cross-talk errors are negligible. For very small feature-size technologies however, the situation may change.

7. Conclusions

Parameter variation is the next big challenge for processor designers. To gain insight into this problem from a microarchitectural perspective, this paper made two contributions. First, it developed a novel model for process and temperature variation. The model uses three intuitive input parameters, and is computationally inexpensive.

Second, the paper extends the variation model with VATS, a novel model of timing errors due to parameter variation. The model is widely usable, as it applies to logic and SRAM units and is driven with intuitive parameters. The model has been partially validated with empirical data. The resulting combined model, called VARIUS, has been used to estimate timing error rates for pipeline stages in a processor with variation.

References

[1] S. Borkar, T. Karnik, and V. De. Design and reliability challenges in nanometer technologies. In *DAC*, 2004.

[2] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De. Parameter variations and impact on circuits and microarchitecture. In *DAC*, 2003.

[3] K. Bowman, S. Duvall, and J. Meindl. Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration. *JSSC*, 37(2), 2002.

[4] Y. Cao and L. Clark. Mapping statistical process variation toward circuit performance variability: An analytical approach. In *DAC*, 2005.

[5] Robert Chau, S. Datta, M. Doczy, J. Kavalieros, and M. Metz. Gate dielectric scaling for high-performance CMOS: from SiO2 to High-K. In *IWGI*, 2003.

[6] C. E. Clark. The greatest of a finite set of random variables. *Operations Research*, 9, 1961.

[7] N. Cressie. *Statistics for Spatial Data*. John Wiley & Sons, 1993.

[8] S. Das, S. Pant, D. Roberts, S. Lee, D. Blaauw, T. Austin, T. Mudge, and K. Flautner. A self-tuning DVS processor using delay-error detection and correction. In *VLSI*, 2005.

[9] D. Ernst, N. S. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner, and T. Mudge. Razor: A low-power pipeline based on circuit-level timing speculation. In *MICRO*, 2003.

[10] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos. Modeling within-die spatial correlation effects for process-design co-optimization. In *ISQED*, 2005.

[11] Z. Huang and M. C. Ercegovac. Effect of wire delay on the design of prefix adders in deep-submicron technology. In *ACSS*, 2000.

[12] E. Humenay, D. Tarjan, and K. Skadron. Impact of parameter variations on multicore chips. In *ASGI*, 2006.

[13] International Technology Roadmap for Semiconductors.

[14] A. Journel and C. Huijbregts. *Mining Geostatistics*. Academic Press, 1978.

[15] K. Kanda, K. Nose, H. Kawaguchi, and T. Sakura. Design impact of positive temperature dependence on drain current in sub-1-V CMOS VLSIs. *JSSC*, 36(10), 2001.

[16] E. Karl, D. Sylvester, and D. Blaauw. Timing error correction techniques for voltage-scalable on-chip memories. In *ISCAS*, 2005.

[17] T. Karnik, S. Borkar, and V. De. Probabilistic and variation-tolerant design: Key to continued Moore's law. In *TAU*, 2004.

[18] A. Keshavarzi, G. Schrom, S. Tang, S. Ma, K. Bowman, S. Tyagi, K. Zhang, T. Linton, N. Hakim, S. Duvall, J. Brews, and V. De. Measurements and modeling of intrinsic fluctuations in MOSFET threshold voltage. In *ISLPED*, 2005.

[19] X. Liang and D. Brooks. Latency adaptation of multiported register files to mitigate variations. In *ASGI*, 2006.

[20] A. Mallik and G. Memik. A case for clumsy packet processors. In *MICRO*, 2004.

[21] G. Memik, M. H. Chowdhury, A. Mallik, and Y. I. Ismail. Engineering over-clocking: Reliability-performance trade-offs for high-performance register files. In *DSN*, 2005.

[22] S. Mukhopadhyay, H. Mahmoodi, and K. Roy. Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS. *TCAD*, 24(12), 2005.

[23] M. Orshansky, L. Milor, and C. Hu. Characterization of spatial intrafield gate CD variability, its impact on circuit performance, and spatial mask-level correction. *TSM*, 17(1), 2004.

[24] A. Papoulis. *Probability, Random Variables and Stochastic Process*. McGrawHill, 2002.

[25] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2007.

[26] P.J. Ribeiro Jr. and P.J. Diggle. geoR: A package for geostatistical analysis. *R-NEWS*, 1(2), 2001.

[27] T. Sakurai and R. Newton. Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas. *JSSC*, 25(2), 1990.

[28] S. R. Sarangi, B. Greskamp, and J. Torrellas. A model for timing errors in processors with parameter variation. In *ISQED*, 2007.

[29] A. Srivastava, D. Sylvester, and D. Blaauw. *Statistical Analysis and Optimization for VLSI: Timing and Power*. Springer, 2005.

[30] B. Stine, D. Boning, and J. Chung. Analysis and decomposition of spatial variation in integrated circuit processes and devices. *TSM*, 10(1), 1997.

[31] Y. Taur and T. H. Ning. *Fundamentals of Modern VLSI Devices*. Cambridge University Press, 1998.

[32] A. Tiwari, S. R. Sarangi, and J. Torrellas. ReCycle: Pipeline adaptation to tolerate process variation. In *ISCA*, 2007.

[33] X. Vera, O. Unsal, and A. Gonzalez. X-pipe: An adapative resilient microarchitecture for parameter variations. In *ASGI*, 2006.

[34] D. Wu, G. Venkataraman, J. Hu, Q. Li, and R. Mahapatra. DiCER: distributed and cost-effective redundancy for variation tolerance. In *ICCAD*, 2005.

[35] S. Xiong, J. Bokor, Q. Xiang, P. Fisher, I. Dudley, P. Rao, H. Wang, and B. En. Is gate line edge roughness a first-order issue in affecting the performance of deep sub-micro bulk MOSFET devices? *TSM*, 17(3), 2004.

[36] Y. Zhang, D. Parikh, K. Sankaranarayanan, K. Skadron, and M. Stan. HotLeakage: A temperature-aware model of subthreshold and gate leakage for architects. Technical Report CS-2003-05, University of Virginia, 2003.

[37] W. Zhao and Y. Cao. New generation of predictive technology model for sub-45nm design exploration. In *ISQED*, 2006.