Received March 26, 2014, accepted April 17, 2014, date of publication May 14, 2014, date of current version June 9, 2014.

Digital Object Identifier 10.1109/ACCESS.2014.2323233

Confronting the Variability Issues Affecting the Performance of Next-Generation SRAM Design to Optimize and Predict the Speed and Yield

JEREN SAMANDARI-RAD[1], MATTHEW GUTHAUS[2], (Senior Member, IEEE), AND RICHARD HUGHEY[2]

[1]Department of Electrical Engineering, University of California, Santa Cruz, CA 95064, USA
[2]Department of Computer Engineering, University of California, Santa Cruz, CA 95064, USA

Corresponding author: J. Samandari-Rad (jerensrad@soe.ucsc.edu)

This work was supported in part by the National Science Foundation under Grant CNS-1205493 and in part by the University of California, Santa Cruz, Open Access Fund.

ABSTRACT Effectively confronting device and circuit parameter variations to maintain or improve the design of high performance and energy efficient systems while satisfying historical standards for reliability and lower costs is increasingly challenging with the scaling of technology. In this paper, we develop methods for robust and resilient six-transistor-cell static random access memory (6T-SRAM) designs that mitigate the effects of device and circuit parameter variations. Our interdisciplinary effort involves: 1) using our own recently developed VAR-TX model [1] to illustrate the impact of interdie (also known as die-to-die, D2D) and intradie (also know as within-die, WID) process and operation variations—namely threshold voltage (*Vth*), gate length (*L*), and supply voltage (*Vdd*)—on future different 16-nm architectures and 2) using modified versions of other well-received models to illustrate the impact of variability due to temperature, negative bias temperature instability, aging, and so forth, on existing and next-generation technology nodes. Our goal in combining modeling techniques is to help minimize all major types of variability and to consequently predict and optimize speed and yield for the next generation 6T-SRAMs.

INDEX TERMS 6T-SRAM, 16-nm, access-time, aging, optimum architecture, reliability, type of variations, variability, yield.

I. INTRODUCTION

Reliability concerns due to technology scaling have been a major focus of researchers and designers for several technology nodes. Therefore, many new techniques for enhancing and optimizing reliability have emerged particularly within the last five to ten years. This paper expands on our recently published model, called VAR-TX [1], to make an interdisciplinary effort toward robust and resilient 6T-SRAM designs that mitigate the effects of device and circuit parameter variations in order to enhance system performance, energy efficiency, and reliability.

Our interdisciplinary effort involves using our own model VAR-TX (Sections IIB–IIC) along with popular existing models to mitigate not only the effects of process and operation variations covered by our own model (Sections VIA–VIF) but also the impact of other major types of variation such as temperature, NBTI, EM, soft error, etc., which are covered by existing models (Sections IVD–IVG), on the performance of next-generation SRAMs. Additionally, due to the crucial role of NBTI in the aging of devices and circuits, our interdisciplinary effort includes slightly modifying an existing model for older nodes introduced by Cao et al. [2] to show results and analysis from our model for the impact of supply voltage, temperature, and input control in static and dynamic operation on the delay degradation of next-generation SRAMs (Section VIG). Moreover, this paper reviews the progress in the community so far, discusses on-going research, and suggests future work, such as hardware and software collaboration (Section VII).

Athough this paper highlights the most prominent reliability concerns affecting both speed and power (dynamic, static, and leakage) for the sake of providing an elaborate coverage of variability sources, the focus of this paper is on the prediction and mitigation of the impact of the major type of variation affecting the performance and reliability of 6T-SRAM; to minimize the delay and delay variation; to predict and to optimize the speed and yield in upcoming 6T-SRAMs.

Design variability due to D2D and WID process variations has the potential to significantly reduce the maximum operating frequency and the effective yield of high-performance chips. This variability increases the access-time variance and mean of fabricated chips.

As device feature size shrinks, the impact of fabrication variability on product reliability, yield and cost is dramatically increased. Mismatched MOS transistors impact the performance more than ever because device dimensions and available signal swing are significantly reduced. As a main contributor to the overall performance of the system, memory subsystems are one of the components most vulnerable to the effects of variations (e.g., speed and power). Unwanted variation in SRAM may result in access-time and functional failures. Therefore, comprehensive investigation of feasible architectures and organizations is essential in confronting ever-growing scaling issues.

The performance and cost of a given on-chip SRAM requires investigating many alternatives. For example, one cannot compare two different SRAM organizations without comparing their access and cycle times, and chip area and power requirements.

Many modelling techniques have been proposed to minimize the impact of process variations in SRAM and cache, including chip-area models [3], [4], power/leakage models [3], [5]–[8], access-time models [9]–[11], and failure probability models [10]–[12]. Newer techniques can also be used to combat process variations such as adaptive body biasing (ABB) [13] or chip-by-chip resource resizing in various micro-architectural structures [14]. However, these either have inherent costs, must be applied with great caution, or require modification of the chip architecture. Such costly complications demonstrate the importance of inexpensive and early modelling to determine an optimal design that will allow the SRAM to be more tolerant of variations (Section III).

Previously [1], we proposed a novel hybrid analytical-empirical model that exhaustively computes and compares the sensitivity of different 6T-SRAM architectures to the variations in threshold voltage (*Vth*), gate length (*L*), and supply voltage (*Vdd*). This enables the user to select the architecture that gives the minimum delay and/or minimum delay variation while providing the maximum yield possible, for the given area and power constraints. In considering the sensitivity of the critical path to variations in both the overall architecture and within the individual devices, we not only added a new dimension to the path-based statistical timing analysis but also significantly improved upon the previous access-times models [10], [11], [15], [16]—which neither considered architectural sensitivity nor all three parameter variations. We continue this brief review of our recently published model [1] in Sections IIB–IIC.

The main contributions of this paper include:

1. We argue previously published works that suggest square SRAM always produce minimum delays. We show that minimum access-time and/or access-time variation can be obtained from a non-square SRAM (Section VIA).

2. We present the access-time variation calculated by our model for the future 16-nm node and compare it to 45-nm and 180-nm (Sections VIB–VIE, VIH) to show the larger impact of process variations in increasingly small devices and therefore help shed light on the challenges of future robust circuit design.

3. We show the impact of temperature on drain current ($I_{d,sat}$, $I_{d,triode}$ of MOS transistors), wire resistance (R), and frequency for 16-nm node (Sections IVB–IVC, VIF). Furthermore, we illustrate and discuss other important reliability and performance issues such as supply voltage (*Vdd*) fluctuations, static-noise margin (SNM) reduction, soft errors impact, Negative Bias Temperature Instability (NBTI), and more (Sections IVA, IVE, IVF, and IVG respectively).

4. Our novel simulation results and analysis for 16-nm 6T-SRAM (Sections VIA–VIH), and mitigation techniques (Sections IVA–IVG) provide the groundwork for their extension to other types of memory such as 8T-, 10T-, or multi-ported SRAM, cache and CAM.

5. Finally, we review progress in the community, discuss on-going research, and propose future work, such as hardware and software collaboration (Section VII).

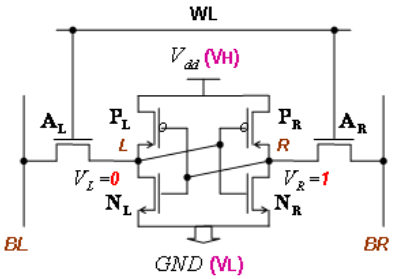

FIGURE 1. 6 transistor (6T) storage cell.

II. OVERVIEW OF SRAM AND OUR MODEL

A. SRAM OVERVIEW

The six-transistor-cell static random access memory (6T-SRAM) (Fig. 1) is the conventional choice for most on-chip memory designs. We do not change the circuit topology from the standard six-transistor cell because of previous studies discouraging this [17], [18]. With power applied, SRAM provides permanent data storage.

Cell design requires balancing several factors including speed, silicon area, and power/leakage consumption [6], [19], [20]. This is challenging due to conflicting goals including 1) minimizing the cell area to achieve high density memory, reduce power, and reduce the cost of the chip; 2) maintaining cell stability with minimum voltage to prevent yield loss due to data corruption; 3) good soft error immunity 4) high cell read current to minimize access time; 5) minimum word line pulse width to conserve power (by reducing bitline swing); 6) low leakage current, especially for battery operated systems [19].

For example, to maintain cell stability and good soft-error immunity while keeping access time short, one might specify

large transistor sizes [21], but large transistors occupy more area and result in increased leakage. Similarly, improving static noise margin (SNM) with smaller pass transistors can lead to a worse write margin [19]. Transistor sizing and circuit styles for 6T-SRAM components (decoders, sense amps, etc.)—and the interconnect sizing, buffers, and SRAM array partitioning—must all be balanced with considerations to the delay, area, and power consumption.

B. A BRIEF REVIEW OF OUR MODEL ASSUMPTIONS AND IMPLEMENTATION VARIATION

VAR-TX [1] is a novel hybrid analytical-empirical model for computing the delay distribution of access-time that considers both D2D and architecture-dependent, spatially-correlated WID variations. We proposed a model for D2D and WID device threshold voltage (*Vth*), length (L), and supply voltage (*Vdd*) variations and showed how the delay distribution can be efficiently computed using delay sensitivities. VAR-TX enables the user to predict the delay and delay variability in future 16-nm on-chip conventional 6T-SRAMs, given input specifications, area, power constraints, SRAM size and shape, number of columns, word-size, etc.

Here is the abstract review of the derivation process for our path-based approach to statistical timing analysis:

1. Compute the sensitivities and store them in tables.
2. Compute the D2D component of the path delay.
3. Express the WID component of the path delay variation as an analytical expression of the device parameter variation.
4. Combine the two components (D2D and WID) of the path delay variations to obtain the joint path delay distribution.
5. Optimize the delay through the examination of all feasible architectures to achieve maximum yield.

For our D2D modelling we exploited the property that for each parameter (*Vth*, L, *Vdd*), the corresponding gate delay in $D_{p,D2D}$ (critical path delay D_P due to D2D) shares a single random variable. Therefore, the D2D variation of D_P due to each parameter can be computed separately through enumeration of the distribution of *Vth*, L, and *Vdd* (Vth_{D2D}, L_{D2D}, and Vdd_{D2D}).

For WID variations ΔP_{WID}, there are both correlated (systematic) and random components. To capture this effect, we use the method introduced by Agarwal [15]. The SRAM area is divided into a multi-level quad-tree partitioning (Fig. 2). We chose six levels of quadrants (sufficient partitioning for our first order analysis) with the top quadrant the entire SRAM and the bottom quadrant the devices. For each quadrant, we generate a random variable according to a normal distribution.

Our hybrid analytical-empirical model was partially built on the empirical data collected from the results of numerous restricted simulations on SRAMs composed of the latest complex circuits. (The simulation is restricted to select critical paths for much shorter run-time). All circuits were designed at the transistor level, with each transistor in the circuit subject to random and spatially-correlated systematic fluctuations

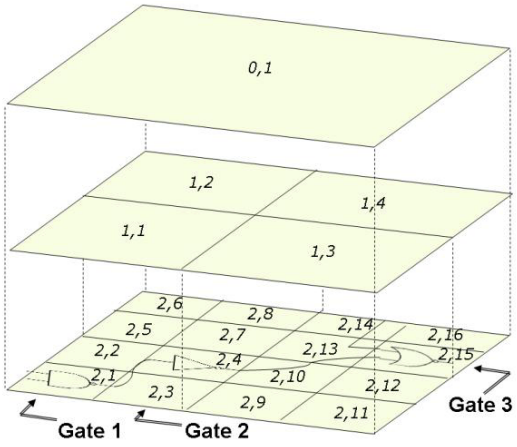

FIGURE 2. Spatial correlation modeling for WID variations (Based on Fig. 1 of Agarwal [15]). Only 3 levels of 6 quadrants are shown to avoid cluttering [1].

of *Vth*, L, and *Vdd*. Our model also includes *layout parasitics* (e.g., the resistance and capacitance of all the bitlines (wires) and wordlines (wires) in the 6T-cell array). To capture the effect of *Vdd* variation we first eliminated most of the supply voltage static IR drop ('IR-drop') and dynamic *Ldi/dt* variation (primarily from dynamic switching) by adding decoupling capacitors (at the cost of increased gate oxide area and consequently increased oxide leakage) in the empirically found crucial locations of our SRAM circuits [22]. After reducing the initial *Vdd* variation, we modeled the remaining *Vdd* variation in the same way we modeled the length, but with only half the variance of L, based on a prior investigation [22]. Such modeling makes sense logically as the initial/untreated *Vdd* variation for future nodes (i.e., 16-nm) could reach 10% [23], [24], with post-treated variation having both random and systematic components [25] (Section IVA).

Although labour-intensive (mainly during the data collection for sensitivities step), the construction of a hybrid analytical-empirical model such as this one takes a reasonable time on a small cluster (weeks, not months). The initial expensive sensitivity analysis characterization involved in the flow is compensated for by the time savings in the subsequent short run-times. While Hspice Monte-Carlo simulations for each of the many possible configurations of the actual large SRAM circuits can take days (which makes such alternatives comparatively quite expensive), VAR-TX carries out the same analysis in minutes. Despite the time savings, for the circuits we have chosen, our model produces delay estimates within 8% of Hspice results.

A total independent variation of 8.8% for the WID sigma of *Vth*, 4.4% for L, and 2% for *Vdd* were assumed for our variability analysis of 16-nm node. For D2D independent variance we assumed 4% for either of *Vth* and L, and 2% for *Vdd*, based on ITRS [23]. Our simulations are based on ASU Predictive Technology Models (PTM) [26]. Sixty different transistor models, each with a different value of V_{TH0}, were used to model *Vth* variations for our

SRAM circuits. To model gate length variations, we stipulated 20 different values of deviation from the standard minimum-size transistor length. Finally, we modelled *Vdd* variations using two extreme cases: the default supply voltage plus 1-sigma and the default *Vdd* less 1-sigma.

We verified the accuracy of our model assumptions through Monte Carlo simulations and validated our model optimization capability by comparing our access-time results with those obtained by Mukhopadhyay and VARIUS [1], [10], [11]. By considering more than one hundred different selected worst case critical paths spanning different regions in our quad-tree modeling that we found through experimental results, similar to the method introduced by Bowman [27], we were able to incorporate not just the correlated, but also the uncorrelated and partially correlated paths. We showed that our proposed approach produces very accurate results [1].

C. OUR MODEL OPTIMIZATION

In addition to computing the access time of a given SRAM system, VAR-TX performs exhaustive computations and comparisons based on the user entry (e.g., SRAM size, word-size) using its embedded library of lookup tables (constructed from the linearized device delays for different configurations) to provide the minimum-access-time architecture/organization that satisfies a given desired power and area requirement from the modelled alternatives. VAR-TX does this first phase of our optimization within thirty seconds, even for large SRAM circuits with nearly countless critical parameter fluctuations. VAR-TX also provides a measure of the expected variability in this minimum access-time.

Once the best architecture is found by our VAR-TX modeling methodology in the first phase of our optimization process, we additionally perform our sensitivity adjustments (such as bitline, interconnect width, and pitch optimization) in the second phase of our optimization process to further minimize the variability effects (including aging, such as EM and NBTI) and thus maximize the reliability of our design.

III. TYPE OF VARIATION

As node sizes decrease in order to achieve higher integration and lower cost, variation impacts are becoming more critical in the design of SRAM technologies. For example, in the 65-nm node, variation effects can be avoided by guard-banding the design and following recent advances in design development. However, in smaller geometries such as 22-nm or 16-nm, we are required to perform careful variation effect analysis by understanding the potential impacts of different types of variation on our design. A recent report by SOlido shows that 65% of engineers surveyed see variation effects as their top concern for analysis in the next few years [28].

There are three types of variation to consider:

Operational: environmental and loading variations. For example, the voltage of the power supply (V), temperature (T), and different loading conditions. In the design of SRAM, there may be different conditions in the actual implementation which can make them function differently than intended. The environmental fatigue phenomena—negative bias temperature instability (NBTI), hot carrier Injection (HCI), gate current shifts, shot noise, thermal noise, random telegraph noise (RTN)—are examples of temporal variations that could also be placed in this category.

Fabrication: global and local process variations (PV). Global variations have been historically analysed through corner-based models, but now, as nodes become smaller, the local effects are starting to be almost as important as the global effects. Therefore, they have to be considered as well when using Monte Carlo based tools.

Implementation: layout-based variation effects. Physical parasitic effects have been one of the design challenges during the last two decades. Similarly, power integrity connectivity effects for supply demand have increasingly become a concern in the last few years. More recently, the concern has been layout-dependent effects, which involve changes in electrical characteristics (such as *Vth* and effective length (L_{eff})) of specific devices depending on where they are placed within the SRAM.

Alternatively, the sources of hardware variation can be classified into four groups: 1) Manufacturing, 2) Environment, 3) Vendor, 4) Aging. These types of variations are explained by Gupta P. et al. in their recent work [29].

Designers often have to make a choice between running fewer simulations, which means there is less predictability in the quality of the design, or do more simulations, which run the risk of increased validation cost. This represents the fundamental challenge for today‘s designers: choosing between over- or under-designing.

IV. SENSITIVITY ANALYSIS OF VARIABILITY IN SRAM

In this section we provide the most prominent reliability-related design challenges we took into consideration during the process of making VAR-TX [1]. The purpose is to provide a glimpse of the challenges that future technology will need to successfully confront in order to create higher performance chips without sacrificing the reliability aspects of the manufactured product.

In this section we discuss the impacts of such crucial design challenges as supply voltage, temperature, wire delay variability, SNM, soft errors, and NBTI on the variability analysis of 6T-SRAM. In Section VI, we present our selected simulation results.

A. IMPACT OF SUPPLY VOLTAGE (Vdd)

As mentioned in Section IIB, the main causes of *Vdd* variation are static IR-drop and Ldi/dt drop due to dynamic switching.

The amount of DC change in the power supply voltage (IR-drop) is a function of the average value of the current that the circuit draws from the power supply network, which randomly varies temporally and spatially. As a result, the spatial variation of the IR-drop across the power distribution

network is usually considered unpredictable. Additionally, power-saving techniques such as clock gating and sleep transistor logic tend to increase the variability of spatial and temporal IR-drop distributions across the chip [25].

Although the resistance of the package is quite small, the inductance of the package leads is significant in both wire-bond and C4 bump arrays. This causes a voltage drop (the di/dt drop) at the pad locations due to the time-varying current drawn by the devices on the die. Therefore, the voltage at the device-level is the *Vdd* minus the IR-drop and Ldi/dt-drop [30]. Decoupling capacitance is inserted to eliminate most IR-drop and Ldi/dt variation.

Excessive voltage drops in the power grid reduce switching speeds and noise margins and inject noise which might lead to functional failures. High average current densities lead to undesirable wear on the metal wires due to electromigration (EM). Therefore, the challenge in the design of a power distribution network is achieving excellent voltage regulation at the consumption points while considering the wide fluctuations in power demand across the chip, as well as minimizing the area of the metal layers.

One mitigation technique [18] aims at effectively confronting this type of variability challenge through the collaboration of hardware with software. Although the resilient circuits ensure correct system operation within the presence of dynamic variations, a hardware-only solution is purely reactive. Allowing software, such as the operating system or some form of runtime layer, to monitor the recovery cycles in a resilient hardware might enable a more efficient system design by anticipating future events based on the workload. Recent experimental results [18] show that the occurrence of *Vdd* drops varies widely across the different programs. By giving software the capability to monitor recovery cycles in the resilient hardware, we can track the optimum F_{clk} setting for each workload, store these values, and then reuse this information during subsequent executions. In this way, software can predict the optimal F_{clk} setting based on previous measurements to enhance the performance and energy [18]. This is discussed further in Section VII.

B. IMPACT OF TEMPERATURE ON CURRENT (I_d)

An increase in temperature impacts the performance of SRAM. Delay and power suffer from increases in temperature due to the adverse impact of temperature on the drain current and interconnect resistance. Therefore, in our analysis of the temperature dependency of delay, power [31], and performance of SRAM, we consider both the change in the drain current of the transistors on the critical path and the change in the wire resistance of the bitlines and wordlines.

First, the analysis of the drain current and its dependency on temperature is the basis for the delay propagation (t_p) and leakage current (I_{leak}), both of which relate to the temperature-dependent parameters used in the drain current. While we have used drain current equations derived from short-channel devices [33] for our SRAM modeling and simulations, in this section we have adapted equations (1)-(2) derived from long-channel devices [33] for show-casing and simplicity. The drain currents for this section may also be derived from short-channel devices for controversially higher accuracy at the expense of higher complexity. Referring to the drain current equations below (Eqs. 1 and 2, which are sufficiently accurate for our first-order analysis), the temperature affects certain variables such as the mobility and threshold voltage—which determine I_d (drain current), $I_{d,sat}$ (drain current in saturation), and R_{eq} (equivalent resistance) of the transistors on the critical path. The two equations correspond to the saturation and triode modes, respectively:

$$I_{d,sat} = \mu Cox \frac{W}{L}(V_{gs} - V_{th})^2 \ (Saturation) \tag{1}$$

$$I_{d,triode} = \mu Cox \frac{W}{L}(V_{gs} - V_{th})V_{\min} - \frac{V_{\min}^2}{2} \ (Triode) \tag{2}$$

where μ is the charge-carrier effective mobility, Cox (which is equal to e_{ox}/t_{ox}) is the gate oxide capacitance per unit area, W is the gate width, L is the gate length, V_{gs} is the potential difference between the gate and source of a transistor, *Vth* is the threshold voltage, and $V_{\min}$ represents the potential difference between the drain and source of a transistor.

During the access operation (and read operation), the access transistor (A_L in Fig. 1) is in saturation mode and the pull-down transistor (N_L in Fig. 1) is in triode mode. A similar principal applies to the write operation, except that instead of the left access transistor A_L operating in the saturation mode, the right access transistor A_R operates in the triode mode while the pull-up transistor P_R operates in the saturation mode.

An increase in temperature decreases the mobility, μ, as shown in the following equation:

$$\mu(T) = \mu_0(T/T_0)^{\alpha_\mu} \tag{3}$$

Typical electron mobility for Si at room temperature (300K) is 1400 $cm^2/(V.s)$ and the hole mobility is around 450 $cm^2/(V.s)$.

Similarly, an increase in temperature leads to a decrease in the threshold voltage, as shown in the following equation:

$$V_{th}(T) = V_{th0} + \alpha_{Vth}(T - T_0) \tag{4}$$

The temperature dependence of mobility, threshold voltage and resistance along with their typical values for the parameters used in Eqs. (3) and (4) are summarized in Table 1 [32].

According to Eqs. (3) and (4), both the mobility (μ) and the threshold voltage (*Vth*) decrease with an increase in temperature. However, the decrease in μ is slightly larger than the decrease in *Vth*, comparatively. Looking back at Eqs. (1) and (2), we observe that the impact of a temperature increase on the drain current will not be dramatic, simply because the changes in *Vth*(T) and μ(T) are approximately equal and opposite in sign. The authors of VARIUS [11] express the partial cancellation of μ and *Vth* temperature dependency by illustrating the relation between these two parameters in the

TABLE 1. Temperature dependency of mobility, threshold voltage and resistance [32].

$$\mu(T) = \mu_0 (T/T_0)^{a_\mu}$$
$$V_{th}(T) = V_{th0} + \alpha_{Vth}(T - T_0)$$
$$R(T) = R_0[1 + \alpha_R (T - T_0)]$$

where T is the temperature, T_0 is the nominal temperature, μ_0 is the mobility at T_0, V_{th0} is the threshold voltage at T_0, R_0 is the resistance at T_0; α_μ, α_{Vth} and α_R are empirical terms named the mobility temperature exponent, threshold voltage temperature coefficient, and resistance temperature coefficient, respectively, where α_μ = -2, -1 mV/°C ≤ α_{Vth} ≤ -4 mV/°C, and α_R in Cu is 0.004.

toggling frequency equation below:

$$T_g \propto \frac{L_{eff} V}{\mu (V - V_{th})^{\alpha}} \tag{5}$$

where α is typically 1.3 and μ is the mobility of carriers ($\mu(T) \equiv T^{-1.5}$). As *Vth* decreases, $(V - Vth)$ increases and the gate becomes faster. As T increases, $V - V_{th}(T)$ increases, but μ(T) decreases [11]. The second factor dominates and, with higher T, the gate becomes slower, though not dramatically, especially for nodes of 45-nm or larger. The impact of temperature on T_g, however, is expected to be more pronounced for future technology nodes (i.e., 16-nm) when the operating temperature can vary from −30C to 175C (e.g., in automotive context) which will result in several tens of percent performance change and several orders of magnitude sleep power variation [23], [24].

$$t_p = \frac{t_{pHL} + t_{pLH}}{2} = 0.69 C_L \left(\frac{R_{eqn} + R_{eqp}}{2}\right) \tag{6}$$

This analytical equation assumes that the equivalent load-capacitance, C_L, is identical for both the high-to-low, t_{pHL}, and low-to-high, t_{pLH} transitions, with R_{eqn} and R_{eqp} representing the equivalent on-resistance of the NMOS and PMOS, respectively. Typically, the on-resistance of NMOS and PMOS are set to be approximately equal (through transistor sizing) so that they have identical propagation delays t_p for both rising and falling inputs.

C_L in Eq. (6) represents the total load capacitance, which is composed of input, diffusion and gate capacitances of the NMOS and PMOS transistors of the inverter [33]. C_L increases as the temperature increases mainly due to the junction capacitance (C_j), affecting the diffusion capacitances of the NMOS and PMOS transistors. C_L also increases due to K_{eqn} and/or K_{eqp}, but only slightly (K_{eqn} and K_{eqp} are multiplication factors for NMOS and PMOS, respectively, and relate the linearized capacitor to the value of the junction capacitance under the zero-bias condition). The detailed definitions/descriptions of the components of C_L as well as their associated residual dependency to temperature can be found in [31] and [33].

R_{eq} in Eq. (6) is related to the saturated drain current, I_{DSAT}—which, in turn, is related to μ and *Vth* (two temperature-dependent parameters)—through the following equations [33]:

$$R_{eq} = \frac{1}{\frac{V_{dd}}{2}} \int_{\frac{V_{dd}}{2}}^{V_{dd}} \frac{V}{I_{DSAT}(1 + \lambda V)} \partial V = \frac{3V_{dd}}{4I_{DSAT}} \left(1 - \frac{7}{9} \lambda V_{dd}\right) \tag{7}$$

$$with\ I_{DSAT} = \mu C_{ox} \frac{W}{L} \left[(V_{dd} - V_{th}) V_{DSAT} - \frac{V_{DSAT}^2}{2} \right] \tag{8}$$

where λ is an empirical parameter called the channel-length modulation. In general, λ is proportional to the inverse of the channel length. In shorter transistors (such as those used in 16-nm technology), the drain-junction depletion region presents a larger fraction of the channel, and the channel-modulation effect is more pronounced. V_{DSAT} is the saturation drain voltage.

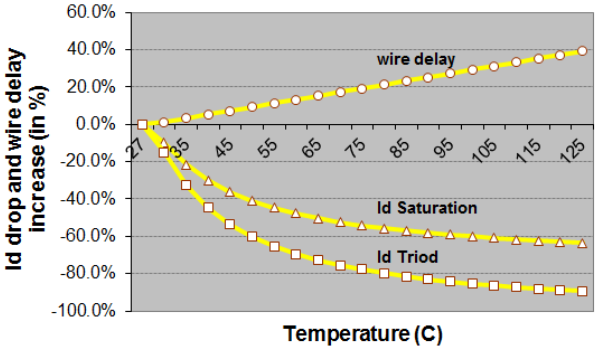

FIGURE 3. Drain Current and wire delay vs. temperature for 16-nm node.

By applying the equations presented in this section to 16-nm technology, we can observe the impact of temperature on $I_{d,sat}$, $I_{d,triode}$ (Fig. 3; wire delay equation from Table 1). The equation involving the temperature dependency of wire delay is given in Table 1, and will be discussed in the next section.

For the temperature range of 27 °C to 125 °C, the average I_d drop is less than 1% per °C, though is more than 2% per °C for 27 to 55 °C.

C. IMPACT OF TEMPERATURE ON INTERCONNECT

In addition to its impact on the drain current, an increase in temperature results in an increase in the resistivity (R) of a conductor of uniform cross-section length which, in turn, increases the delay (t) of the interconnects (widely known as Elmore delay):

$$t = \ln(2)\tau = 0.69\tau = 0.69 \times R \times C \tag{9}$$

where τ is the time constant; R is the resistance; and C is the capacitance.

Wire delay also depends on temperature due to temperature dependency of the resistivity component of wire [32]:

$$R(T) = R_0[1 + \alpha_R(T - T_0)] \tag{10}$$

where, α_R is the resistance temperature coefficient of the interconnect. The typical value of α_R in copper is 0.004 (Table 1).

Unsurprisingly, the wire delay has a linear relation to the wire resistivity and increases with temperature at an approximate rate of 0.4% per degree centigrade. Studies [32] show that the increase in temperature of interconnects used in commercial 65-nm technology is about 6.8 °C, and this translates into approximately a 2.72% increase in the interconnect delay. This means the impact of temperature on the delay of typical interconnects used in contemporary SRAM chips today is not significant, especially as compared to the impact of temperature on leakage current [11]. This impact, although currently insignificant, may become significant in 16-nm SRAMs experiencing accelerated aging due to wires and transistors wearing out faster due to higher temperatures. Degradation can be noticeable within a few milliseconds and may not saturate for several years [24]. In Section VI, we discuss how the combined drain current decrease and wire delay increase (due to temperature increase) impact the access-time and performance of 16-nm 64KB SRAM.

The extended version of process, voltage, temperature (PVT) variations and their reduction techniques are discussed in J. Samandari-Rad's and others' work [31], [34]. We extend our sensitivity analysis regarding the temperature (T), supply voltage (*Vdd*), and current (I_d) in Section IVG, where we discuss the impact of these parameters on aging of the existing and next generation CMOS designs.

D. PARAMETERS IMPACTING AGING

Studies have shown that all major aging effects exhibit a temperature dependency, as in Arrhenius' Law [35]. An aging effect λ_{EFF} has the property [34]:

$$\lambda_{EFF} \propto e^{\frac{-E_a}{kT}} \tag{11}$$

where E_a is the activation energy (which is specific for a certain aging process), $k$is Boltzmann's constant, and T is the temperature. For existing technology nodes (i.e., 22-nm), the critical aging effects include *negative bias temperature instability* (NBTI), *random telegraph noise* (RTN), *electromigration* (EM), *Time-Dependent Dielectric Breakdown* (TDDB), and *Hot Carrier Injection* (HCI).

While the exact mechanism causing NBTI is still a topic of active research, a common explanation is that high electric fields in the gate region cause the activation of *traps* in the gate material which when filled create a fixed charge that changes the surface potential and in turn causes the threshold voltage to shift [34]. NBTI primarily affects pMOS devices and exhibits itself on two time scales. The first is a short time constant (ns regime) phenomenon whereby a device under high gate voltage stress will exhibit a threshold voltage higher than normal for a short period of time, with subsequent return to normal after the stress is removed. The second is a slow and steady change in the threshold voltage over time as traps get permanently filled [34]. Fig. 4 shows a plot of threshold voltage increase after seven years of operation for a range of technologies.

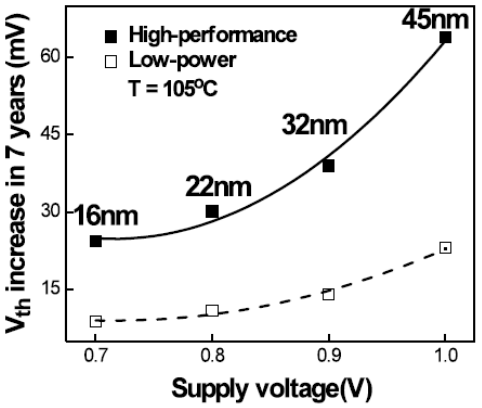

FIGURE 4. The prediction of V_{th} increase in 7 years due to NBTI [34].

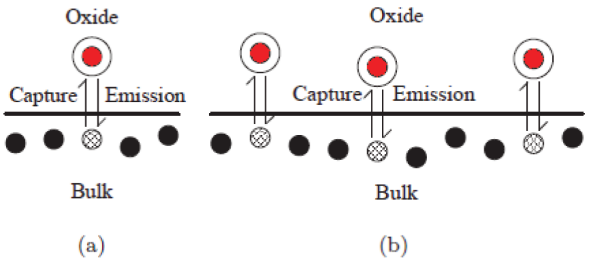

FIGURE 5. Capture/emission process of RTN [36]. (a) Single trap. (b) Multiple traps.

Fig. 5(a) shows the physics of RTN. The carrier (hatched circle) is occasionally captured by the trap (red hollow circle) in the oxide, and will be emitted back into the channel after a period of time. Multiple capture/emission events can occur at the same time (Fig. 5(b)).

In Eq. (12) [36] Cao et al. models the impact of RTN on digital circuits, where $V_{ov} = V_{gs} - V_{th}$ is the gate overdrive voltage and β and λ are fitted by experimental data. Using PTM device library, the above model that ΔVth of a 16-nm device can be as much as 130mV.

$$\Delta V_{th} = \frac{\beta - \lambda V_{ov}}{W \cdot L} \tag{12}$$

As briefly mentioned earlier, one of the challenges resulting from the decreasing chip feature size is increased current densities carried by interconnects, as confirmed by ITRS [23]. These large currents cause various reliability problems—amongst which electromigration (EM) is dominant. EM is caused by momentum transfer between the flowing electrons and copper atoms whose positions gradually move over time. Simply put, EM is caused by the erosion of metal interconnects through ion movement. Failures due to EM typically

manifest themselves as shorts (hillocks), opens or increased wire resistance (void) [37]. Fig. 6 illustrates an instance of such EM-related failures.

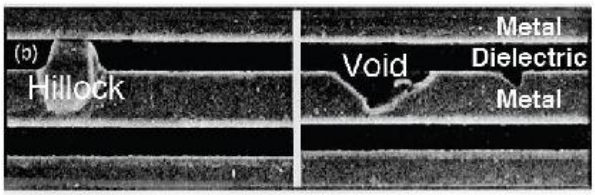

FIGURE 6. Hillocks and voids induced by electromigration with high current density in a Cu interconnect [92].

In addition, apart from the property expressed in Eq. (11), λ_{EM} is also affected by the thermal gradients on a chip. In this case the time dependency of electromigration becomes:

$$\lambda_{EM} \propto e^{\frac{-E_a}{k(T+\Delta T_{joule})}} \tag{13}$$

where ΔT_{joule} is the change in heat energy resulting from local power consumption and not from heat conducted from elsewhere in the chip.

An additional effect that shortens chip lifespans is *thermal cycling*—which induces stress through periodic heating and cooling—modelled through the Coffin-Manson equation [34], [38].

$$N = C(\frac{1}{\Delta T})^q \tag{14}$$

where N represents the expected number of cycles until a failure occurs, C is a material constant, ΔT is the change in temperature, and the exponent q is the experimentally determined Coffin-Mason exponent with $q \in$ [39], [40].

Of the two other temperature-related effects impacting the circuit aging, TDDB [41] results in conductive paths due to the breakdown of the dielectric through the formation of traps caused by high electric fields and HCI [41] is caused when hot carriers in a source-drain current attain sufficient energy to be injected into gate oxide to form traps [34].

To keep the scope of this research in bound, of the aging parameters briefly discussed in this paper, we focus only on NBTI (by discussing it in further detail in Section IVG) since NBTI has become the most prominent of these phenomena and has received widespread attention in the community.

E. IMPACT OF STATIC-NOISE MARGIN (SNM)

The noise margin of a SRAM cell is defined as the minimum amount of DC noise required to flip the state of the cell [42]. The SNM of a cell is often used as a measure of the robustness of an SRAM cell against flipping [43]. It represents the resilience of the design in the event of a disturbance.

As CMOS technology continues to scale down, SNM decreases with successive technology generations [44]. Fig. 7 shows how nominal SNM changes with supply V_{CC} (namely *Vdd*) from 32-nm to 15-nm for junctionless (JL) fin field-effect transistors (FinFETs) 6T-SRAM. With increasing V_{CC},

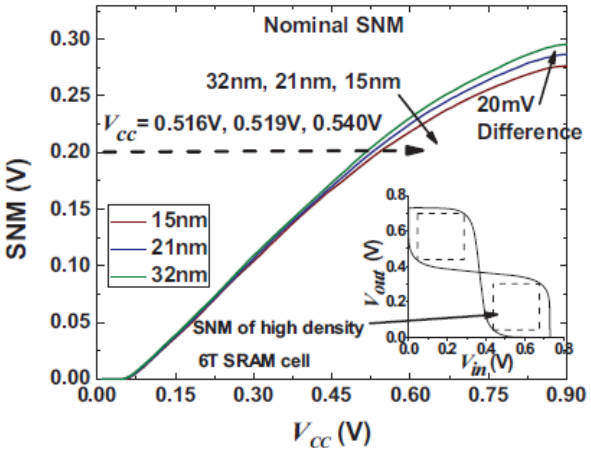

FIGURE 7. Nominal SNM as a function of working V_{CC} for high density design JL FinFET 6T-SRAM cells. Note that for successive technology nodes, SNM decreases when $V_{CC\,min}$ is held fixed. Conversely, $V_{CC\,min}$ increases when SNM is held fixed. The inset plot refers to higher doping levels (e.g., $ND = 3 \times 10^{19}$ cm^{-3}) [45].

the SNM diverges for different technologies with differences of up to 20 mV at $V_{CC} = 0.9$ V. As cell density increases, power consumption becomes a crucial consideration requiring reduction of V_{CC} to conserve both dynamic and leakage power. The minimum working supply voltage $V_{CC\,\min}$ is thus an important metric for judging the viability of a cell design [45]. In general, for a fixed SNM, $V_{CC\,\min}$ increases with scaling. Fig. 7 shows, for instance, how enforcing SNM of 0.2 V causes $V_{CC\,\min}$ to increase from 0.516 V at the 32-nm node to 0.540 V at 15-nm. In addition to SNM, static/dynamic read and write noise margins also affect $V_{CC\,\min}$. However, considering all such metrics would raise many more design issues outside the scope of this paper.

It is desirable to have a sufficiently large noise margin to ensure that flipping does not occur. However, an increase in SNM makes the cell difficult to write by increasing its data holding capability, which increases write failures. This means that, although the SNM can be increased by careful sizing, the cumulative/joint failure probability of SRAM is not reduced correspondingly.

For example, reducing the size of the access transistors (A_L and A_R in Fig. 1) improves the SNM [10], [43] and therefore, decreases read-failure probability. At the same time, the write-failure probability increases (Fig. 8(a)). Hence, the reduction in the sizes of access transistors that results in a maximum SNM does not necessarily correspond to a minimum-failure probability (Fig. 8(a)). Moreover, increasing the size of all the transistors in a cell by the same factor does not modify the SNM. However, an increase in the size of all the transistors in a cell considerably reduces its failure probability by reducing the standard deviation of the *Vth* variation (Fig. 8(b)) [10].

In short, Figs. 8(a) and 8(b) show that an increase in the SNM does not necessarily reduce the overall failure probability and an SNM-based analysis of the cell does not directly correspond to the memory failure probability and

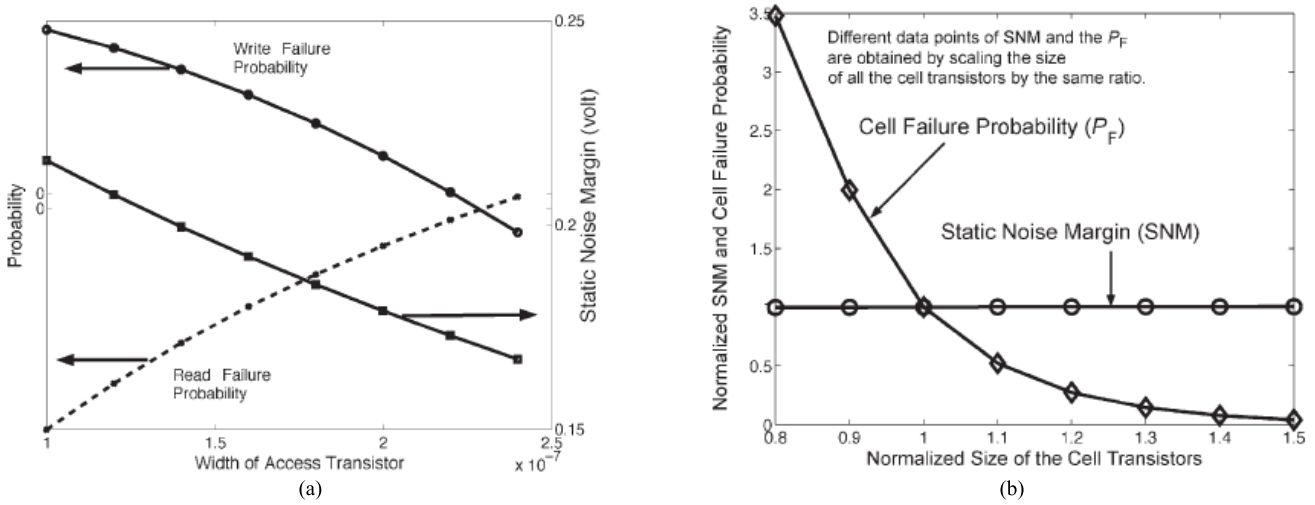

FIGURE 8. Variation of SNM and failure probability with (a) width of the access transistors; and (b) normalized cell area [10].

yield. Hence, a statistical analysis and design of the cells and memory architecture is necessary to ensure acceptable yield in the nano-meter regime. Our proposed model VAR-TX [1] and the simulation results presented in this paper provide such statistical analysis and design of the cells and memory architecture.

F. IMPACT OF SOFT ERRORS

Soft errors involve changes to data but not changes in the physical circuit [46]. Therefore, a soft error may be corrected by rewriting the data where it was lost with no adverse effects to the circuit. Soft errors can occur on transmission lines, in digital logic, analog circuits, magnetic storage, and elsewhere, but are most commonly known in memory, which may occupy more than 70% of the chip area [23], [47].

Mastipuram et al. posit that with continuous downscaling of CMOS technologies, memories and logic become more susceptible to radiation increases [48]. Radiation can come from atmospheric neutrons or on-chip radioactive impurities [34]. The increase in the susceptibility of smaller node memories and logic to radiation is linked to the fact that the soft error rate has an exponential relationship to the critical charge Q_{crit}, the minimum amount of charge that can flip a data value in a memory cell [49]. Since Q_{crit} is smaller in newer nodes due to having smaller sensitive depletion area and smaller supply voltage, the newer nodes have higher soft error rates, especially for SRAMs below 40-nm [50]. Aggressive voltage scaling greatly reduces each cell's capacitance increasing vulnerability to low energy alpha particles, or cosmic rays [51], [52]. SRAMs are more vulnerable to soft errors than logic, since memory cells lack transient masking mechanisms and they are much denser. The density makes SRAM cells much more susceptible to process-induced transistor variability, which strongly impacts Q_{crit} [34].

The adverse impact of soft errors on reliability in newer nodes is exacerbated by multiple-bit/multiple-cell upsets (MCU)—which results in an increased MCU occurrence with respect to the total number of upsets [53]. According to recent measurements [53], despite the fact that the soft error rate for a single memory cell or latch has decreased, the capacity on a chip has increased faster than the soft error rate change. For example, the transition from the 130-nm to the 65-nm technology node has reduced the soft error rate by about a factor of 2 in SRAMs. But at the same time, memory capacity has increased faster, resulting in an increase in the system soft error rate. Table 2 [50] shows the single event upset (SEU) rate per microprocessor in various technologies [51]. SEU rate (or soft error rate, SER) is the rate at which a device or system encounters or is predicted to encounter soft errors.

TABLE 2. Soft error rates in microprocessors [50].

Tech-nology (nm)	Relative SEU rate in FITs/kbit	Approx. Mbit per micro-processor	Relative uncorrected SEU rate per micro-processor (kFIT)
180	3.0	1.52	4.3
130	2.4	3.28	7.9
90	1.0	33.6	33.6
65	0.7	44.3	30.5
40	0.94	71.0	67.0

SEU rate is typically expressed as either number of failures-in-time (FIT), or mean time between failures (MTBF). The unit adopted for quantifying failures in time is called FIT, equivalent to 1 error per billion hours of device operation. So, typically, SEU rate is reported in FITs/kbit, equivalent to cell upset events per bit per million hours) [50].

Beyond the increase in SEU rate per microprocessor, power efficiency forces designers to reduce the voltage via sophisticated techniques like dynamic voltage scaling or near sub-threshold voltage which decreases and consequently increases the soft error rate [34].

Giving our highest priority to performance and reliability over power saving considerations, we used interleaving architecture in our design (that we used for our VAR-TX modelling) to reduce the probability of SEU and MCU occurrence. That is, instead of using dynamic voltage scaling or near sub-threshold voltage that could affect performance and reli-

ability, we used, among others, the interleaving architecture design (accessing non-adjacent cells at a time) for soft error occurrence reduction. The interleaving architecture allows the configuration of having the bits of each word of multiple interleaved words scattered with an equal distance from each other on each row, which has the benefit of higher tolerance to soft errors compared to low voltage, single or multiple non-interleaved words per row scenarios. Distributing the bits of a word over the row reduces the number of soft errors caused by a single radiation event [33], [54].

Overall, it is predicted that with existing technology, soft errors will increase in the upcoming nodes, reversing the long-term trend of the past. Similarly, multi-cell upsets will become much more frequent.

G. IMPACT OF NEGATIVE BIAS TEMPERATURE INSTABILITY (NBTI)

The rapid scaling of CMOS technology has resulted in new reliability concerns, such as negative bias temperature instability (NBTI) and non-conductive stress (NCS), among others [2], [55]–[58]. NBTI has become the primary limiting factor of circuit lifetime. As briefly explained in Section IVD, NBTI primarily affects pMOS devices, since they almost always operate with negative gate-to-source voltage; however, the very same mechanism also affects n-channel MOS (nMOS) transistors when biased in the accumulation regime, i.e., with a negative bias applied to the gate. NBTI manifests itself as an increase in the threshold voltage (*Vth*) and a consequent decrease in the drain current and transconductance (gm)—the ratio of the current change at the output port to the voltage change at the input port; $gm = \Delta I_{out}/\Delta V_{in}$. The degradation exhibits logarithmic dependence on time [55], [59].

NBTI degradation is frequency independent [60], [61] but increases with supply voltage (*Vdd*) and temperature [62]. Experimental data further indicate that NBTI worsens exponentially with thinner gate oxide and higher operating temperature [62]–[64]. In fact, since the gate oxide became thinner than 4 nm (as in nodes below 32-nm), NBTI has gradually become the dominant factor to limit circuit lifetime [2], [65]. If not appropriately provisioned for, the degradation due to NBTI may result in up to 50 mV shifts in the threshold voltage (*Vth*) throughout the lifetime of a circuit, which translates to more than a 20% degradation in the circuit speed, or in extreme cases, to a functional failure [62], [66]. Therefore, for nano-scale CMOS circuits, it is essential to develop design methods to understand, simulate, and minimize the degradation of circuit performance in the presence of NBTI, in order to ensure reliable circuit operation over a desired period of time.

With the introduction of High-K Metal gates, a new degradation mechanism, Positive Bias Temperature Instabilities (PBTI), has appeared. The PBTI affects the NMOS transistor when positively biased [67]. Since, in this particular case, no interface states are generated and 100% of the *Vth* degradation may be recovered, the impact of PBTI is not as severe as that of NBTI.

Traditionally, guardbanding has been used to protect against NBTI. For example, the operating frequency can be reduced or the supply voltage can be increased to offset the degradation over the lifetime of a design. Unfortunately, guardbanding incurs a throughput or power cost over the lifetime of a circuit, even though NBTI degradation does not fully accumulate until the end of the lifetime. As such, several dynamic, architecture-level approaches [68]–[71] have been proposed to mitigate NBTI degradation. Evaluation of architecture-level approaches to mitigate NBTI degradation is typically based on analytical degradation models, like Eq. (15) [72]:

$$\Delta V_{th} = A_{NBTI} \cdot \tau_{OX} \cdot \sqrt{C_{ox}(V_{dd} - V_{th})} \cdot e^{\frac{V_{dd}-V_{th}}{\tau_{ox}E_0} - \frac{E_a}{KT}} \cdot t_{stress}^{0.25} \quad (15)$$

where A_{NBTI} is a constant that depends on the aging rate, t_{ox} is oxide thickness, C_{ox} is gate capacitance per unit area, E_0, E_a, and K are fitting constants, and, t_{stress} is time.

Even though the above equation describes NBTI degradation over time at the device level, its accuracy to evaluate NBTI effect at the architecture-level may be limited, simply because it does not account for scenarios like dynamic voltage scaling, averaging effects across logic paths, and different activity and power management schemes [62]. To mitigate these shortcomings, many recent studies have proposed techniques to alleviate the impact of NBTI induced degradation, from the device and circuit-level [55], [65], [69], [70], [73], [74] to the architecture-level [68], [75]–[77].

At the device and circuit level, a closed form solution for ΔVth (Eq. 16) [55] has been proposed that includes the dependency of ΔVth (or aging) on dynamic voltage scaling *Vdd* (the missing factor in Eq. (15), where *Vdd* is not dynamic but fixed) to enable the designers estimate the aging for various technology nodes with higher accuracy. The model in Eq. (16) is the derivative of a differential equation given in [55] and [78], that accounts for trap energy ET, trap energy distribution f(ET), Fermi energy level EF, and Dynamic Voltage Scaling (DVS). Eq. (16) predicts the dependence of device degradation as a function of *Vdd*, t_{ox}, and temperature (T) under multiple *Vdd* in DVS operation—where A, B, and C are relatively constant.

$$\Delta V_{th} \sim K_1 \cdot \exp\left(\frac{-E_0}{kT}\right) \exp\left(\frac{BV_{dd}}{kTt_{ox}}\right) [A + \log(1 + Ct)] \quad (16)$$

The log(1+Ct') term models the stress/recovery phase behaviour to determine if the device undergoes stress or recovery when *Vdd* is changed. Given the value of *Vth* change at t=0 (ΔVth_0), stress time experienced by the device (t'), and the supply voltage to be operated (*Vdd'*), we can predict whether the degradation increases or recovers. Based on this BTI model, we can predict the *Vth* shift assuming that the device is stressed under *Vdd'* from time $t = 0$ to $t = t'$. The degradation increases further if:

$$\Delta V_{th0} < K_1 \cdot \exp\left(\frac{-E_0}{kT}\right) \exp\left(\frac{BV'_{dd}}{kTt_{ox}}\right) [A + \log(1 + Ct')] \quad (17)$$

Otherwise, the degradation recovers until it reaches equilibrium [55]. This condition at the boundary of supply voltage change allows the accurate aging prediction under DVS operation.

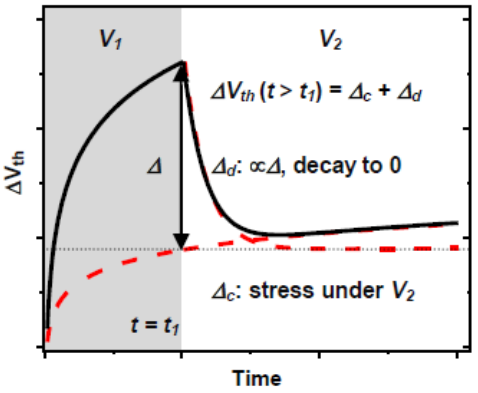

FIGURE 9. The *Vth* shift under DVS is non-monotonic, containing two parts: one for the constant stress (Δ_c) and the other for the dynamic aging (Δ_d) [79].

Fig. 9 illustrates *Vth* shift under DVS. When the stress voltage changes from V_1 to a lower V_2 at $t = t_1$, the behaviour of *Vth* degradation (aging) changes. Since the degradation is highly sensitive to the voltage (Eq. 16), dynamic voltage scaling (DVS) leads to different amounts of circuit aging. In this case, the traps (Fig. 5) emit some of the charge carriers, and the number of occupied traps reaches a new equilibrium [79].

Thus, ΔVth after t_1 has two parts:

$$\Delta V_{th}(t) = \Delta_c + \Delta_d \tag{18}$$

where Δ_c is the stress part under V_2 and Δ_d is the dynamic part before reaching the equilibrium [79].

When $V_2 < V_1$, Δ_d behaves as a recovery:

$$\Delta_d = \Delta \cdot [A + B\log(1 + C(t - t_1))] \tag{19}$$

where Δ is the difference between the aging under V_1 and that under V_2 at $t = t_1$, as shown in Fig. 9 [79].

If $V_2 > V_1$, Δ_d behaves as an accelerated stress. Eventually the degradation converges to the stress under V_2 [79].

At the architecture level, techniques have been proposed to bias input vectors to mitigate aging [75], enhance throughput at the expense of aging in a multi-core environment [76], monitor and adapt to estimated processor lifetimes [71], [80], perform aging-aware scheduling [77], and apply voltage scaling [72] or power gating [68] to mitigate the effects of aging.

Gupta et al. [62] report that due to the underlying physical phenomena that cause NBTI, the degradation is front-loaded by nature. As illustrated in Fig. 10, this means that the rate of degradation is rapid in the early lifetime and slows down considerably under continued stress.

For our NBTI analysis, we adopt the method/results introduced by two of the most recent reputable works. The first [2], [55], [59], [79] relies on device-level analytical models and the second [62] utilizes its proposed flexible numerical

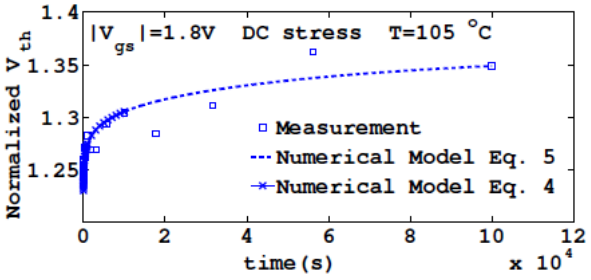

FIGURE 10. An NBTI model [62] vs. measurement data by W. Wang et al. [93].

model for NBTI degradation analysis. We use a combination of these two models for our NBTI analysis to estimate the impact of both device-level and architecture-level techniques on NBTI degradation accurately and efficiently. In Section VIG we illustrate and analyze the impact of NBTI on the performance of logic/SRAM circuits under various operating conditions, such as supply voltage, temperature, and input vectors, to show that given a circuit topology and input switching activity, it is possible to efficiently predict the degradation of circuit speed over a long period of time for the next generation nodes (i.e., 16-nm).

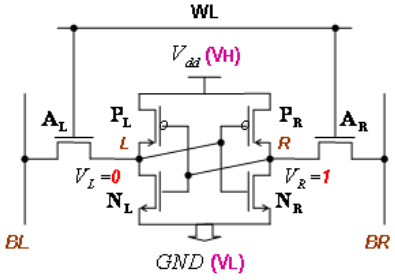

FIGURE 11. (Repeat of Fig. 1) 6 transistor (6T) storage cell.

V. FAILURE IN SRAM

The 6T-SRAM cell consists of two N-type access transistors and two cross-coupled CMOS inverters (Fig. 11). Large mismatches in transistor strengths due to scaling (or fluctuations in die electrical characteristics) can cause the cell to fail.

SRAM cell failures can be classified into four categories: *read, write, access-time,* and *hold* failures.

Read failure occurs when the cell state flips during a SRAM cell read. Because a voltage divider exists between BL (precharged at VDD) and GND, V_ℓ (the node L voltage, a 0 in Fig. 11) is raised from zero to V_{read} through A_L and N_L. If V_{read} exceeds the tripling voltage, V_{trip}, of the (N_R, P_R) inverter, the cell state flips—a read failure. Fluctuations in the *Vth* of A_L and N_L (or of the N_R/P_R *Vth*) lead to large variations in V_{read} (or V_{trip}, respectively) [17].

Write failure occurs when a memory cell does not register an input change correctly. Because a voltage divider exists between BR (at GND) and VDD, V_r transitions to V_{write} through A_R and P_R when a zero is written to node R in place

of an original one. The write fails if V_{write} is larger than the V_{trip} of the (N_L, P_L) inverter. *Vth* variations in A_R and P_R, and also in A_L and P_L—typically the smallest transistors in the cell—cause large variations in V_{write}. This V_{write} ambiguity means a high write-failure probability [17].

Access time failure: The access time of a cell (T_{access}) is the time required to develop a predefined voltage between BL and BR. When node L stores a zero, BL will discharge through A_L and N_L in a read operation. The A_L and N_L strengths influence the discharge speed. *Vth* variations in these transistors cause a spread in T_{access} [17]. If T_{access} exceeds the maximum tolerable limit ($T_{\lim it}$), an access time failure occurs.

Hold failure is the destruction of the cell content in standby mode with the application of a lower supply voltage VH (below 0.5V in our 16-nm node, primarily to reduce leakage in standby mode) [10].

Read, write, access-time, and hold failure probabilities are highly sensitive to *Vth* variation [17] and considerably sensitive to *L* and *Vdd* variations [10] and can be as high as 0.15 for 16-nm nodes [18]. If, in addition to process variations, other types of variations—such as lifetime variations**,** external noise, and intrinsic noise (all of which are defined in [18] and briefly discussed in the previous sections of this paper)—are taken into account, this cell failure probability can be even higher.

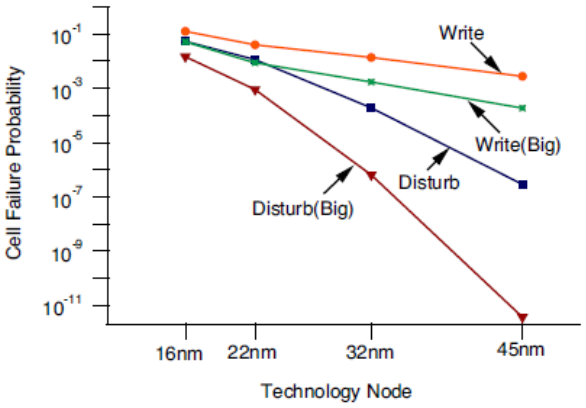

FIGURE 12. SRAM variability-induced failure rates for various technologies [18].

Fig. 12 predicts the impact of scaling (independent of other possible effects) on the failure rates for the SRAM. The x-axis shows the relevant technology nodes and the y-axis shows the estimated SRAM bit-cell failure rate due to manufacturing-induced variability.

The curves in Fig. 12 that are labelled Write and Disturb show the failure rates (due to write and read-disturb mechanisms) for a conventionally scaled SRAM bit cell, and the curves labeled Write(Big) and Disturb(Big) show failure rates due to the same mechanisms for an SRAM bit-cell that is approximately 40 percent larger in size. We can make the following two observations from Fig. 12 [18]:

1. SRAM failure rates will continue to be a problem and will require even more circuit and architectural innovations (perhaps beyond the existing redundancy and error correction techniques) to combat increasing manufacturing variability.

2. Enlarging the SRAM bit cell (reverse scaling) seems to be a moderately effective technique in controlling the impact of variability and may be used locally to create hardened portions of a design, but at a significant cost in layout density.

In addition to the reverse scaling remedy, a moderate increase in power supply has proven to be an important factor in significantly reducing circuit failure rates due to variability. Of course, this comes at the expense of additional power consumption, which is already a major factor for many types of designs. (i.e., 16-nm).

VI. RESULTS AND ANALYSIS

In this section, we present our simulation results to illustrate the impact of major variability concerns on the access time and performance of next-generation SRAM. First, we use our own model, VAR-TX, to show and analyze the impact of process, operation, and temperature variation (namely *Vth* affected by temperature) on access time. Then, we use our own slightly modified version of Cao's model [2] to show and analyze the impact of NBTI on the delay of 16-nm SRAM.

We used the mixed-signal Ultrasim simulator (MMSIM72-Ultrasim64, Cadence Inc.) to produce the results presented in this section.

A. ACCESS-TIME

We characterize our access-time results with the following terms:

ACS (ACcess time Squared): minimum access time for an SRAM where the optimal organization takes a square shape. ACS is always larger than or equal to ACI.

ACI (ACcess time Ideal): similar to ACS, but the optimal organization need not take a square shape.

ACavg (ACcess time average): the mean access time for a SRAM of any shape, as affected by the process variations.

ACH (ACcess time High): the slowest possible access time for a SRAM of any shape, as affected by the process variations.

ACL (ACcess-time Low): the fastest possible access time, and the opposite of ACH.

The two upper curves in Fig. 13 show two access time traces for the 16-nm technology. The trace with the sharp peak depicts ACS (upper dashed line); the more linear trace just below ACS shows ACI. The lower traces in the plot analytically break ACI down into its several components, such as bank select or precharge time. The large diamonds surrounding ACS are Hspice results. The triads of numbers (e.g., $\frac{8:64:128}{16:2:8}$) represent number of columns(8):word-size(64):number of rows(128) in the upper sets, and total number of banks($16 = 2 \times 8$):number of columns of banks(2):number of rows of banks(8), in the lower sets.

Several observations follow from Fig. 13:

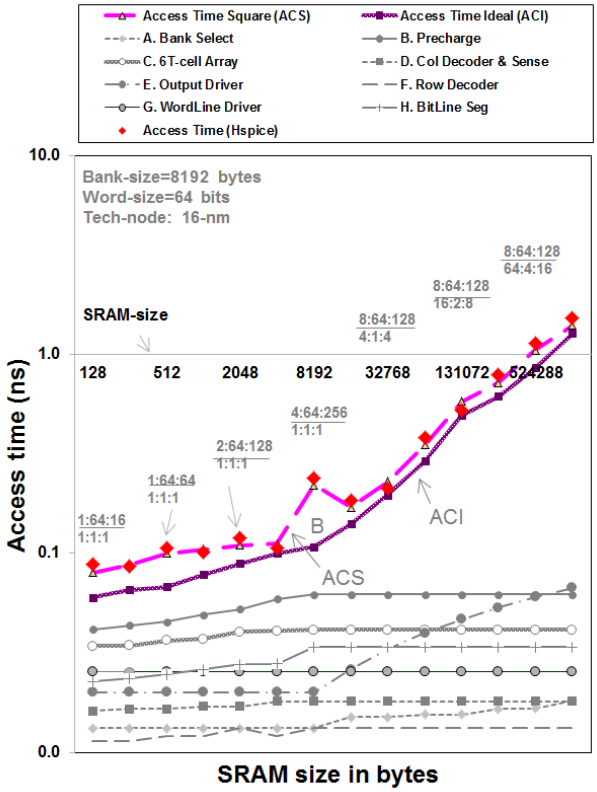

FIGURE 13. Access-time for "square" SRAM (ACS), Access-time for "non-square" SRAM (ACI), and ACI break-down traces.

Size and organization: SRAM delay is a function of SRAM size and organization.

Arguing square policy: Comparison of the ACS and ACI traces reveals that perfectly square SRAMs do not always produce minimum delays, especially for medium-size units. This finding contradicts previously published work [9], [33], [81] and common expectations that have found that the minimum delays are always produced by perfectly square SRAMs (and never by non-square SRAMs). However, our model shows that it is possible, in some cases, that the delay of a non-square SRAM can be shorter than the delay of a perfectly square SRAM of the same size. This is due to the fact that the previous studies base their assumptions heavily on the Elmore delay (delay due to the resistance and capacitance of wiring/routing), which is minimized with a perfectly square SRAM. Although intuitively correct, those studies do not take into account the cumulative effect of differently sized components of the SRAM. For example, the longer routing delay of non-square SRAM can be compensated for by making one of the SRAM components, such as the output driver, a bit bigger while making the row decoder a bit smaller.

This means it is possible to achieve a faster access time by selecting an optimum organization and architecture for the SRAM. If one compares the left side of the ACS and ACI traces in Fig. 13, it is apparent that the SRAM access time can be reduced by up to 31% by favoring one or more SRAM input specification over others. For example, word-size can be favored over the number of rows. This "favoritism" involves a negligible amount of extra area and cost for more sense-amps and flip-flops.

Various component delays: Precharge and SixTXArray component delays are much larger than the other component delays. Mitigating this is the fact that SRAM stability increases with sufficiently large pre-charging and discharging times. The large delay times for the Precharge and SixTXArray components effectively outweigh delays from the row decoder, column decoder, and wordline and bitline segmenting. SRAM delay variability tends to be partially obscured as well; this effect will be explained further below.

ACS approaching ACI due to banking: Whereas the left wings of the ACS and ACI traces differ for SRAM sizes up to 8KB (Point B), the right wings are nearly linear and almost overlap. Beyond Point B, the advantage of multiple banks kicks in, changing not only output driver behaviour, but also permitting bank arrangements that make ACS almost the equal of ACI. The nearly linear increase in both ACS and ACI beyond Point B is mostly due to the fact that the output driver changes from parallel to serial mode.

Benefits of intelligent balancing: Finally, we see that intelligent balancing of different SRAM components can provide two benefits. First, optimum delays for larger SRAMs can be reduced nearly to those for small and medium-sized SRAMs. And second, when large-SRAM delays increase with SRAM size in a near-linear fashion, delay and variability become more predictable.

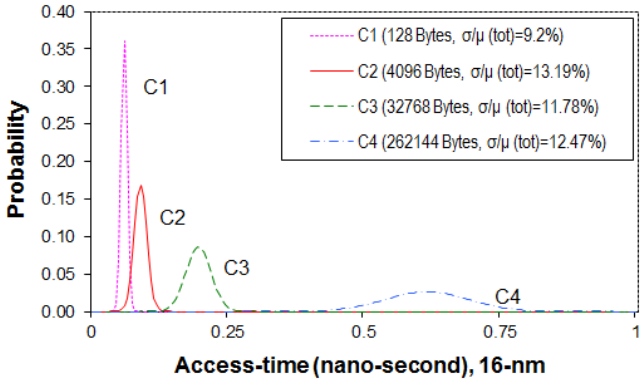

FIGURE 14. Cumulative distribution of access-time for 4 different SRAM sizes in 16-nm node.

B. CUMULATIVE Vth, L, AND Vdd VARIABILITY

As explained in Section IVB, access time is influenced by $I_{d,sat}$ and bitline capacitance. $I_{d,sat}$, in turn, is influenced by the process and operation parameters (*Vth*, *L*, and *Vdd*) and their variation. In that section, we saw that $I_{d,sat}$ has a quadratic dependence on the difference between the gate voltage and threshold voltage ($V_{gs} - V_{th}$) and a linear dependence on the channel length dimensions (W and L), oxide thickness (t_{ox}), and bitline capacitance. Fig. 14 shows the cumulative probability distribution of the access-time for four different SRAM sizes, with the assumed parameter variations presented in Section IIB (i.e., independent WID variations of 8.8% for *Vth*, 4.4% for *L*, and 2% for *Vdd* and independent

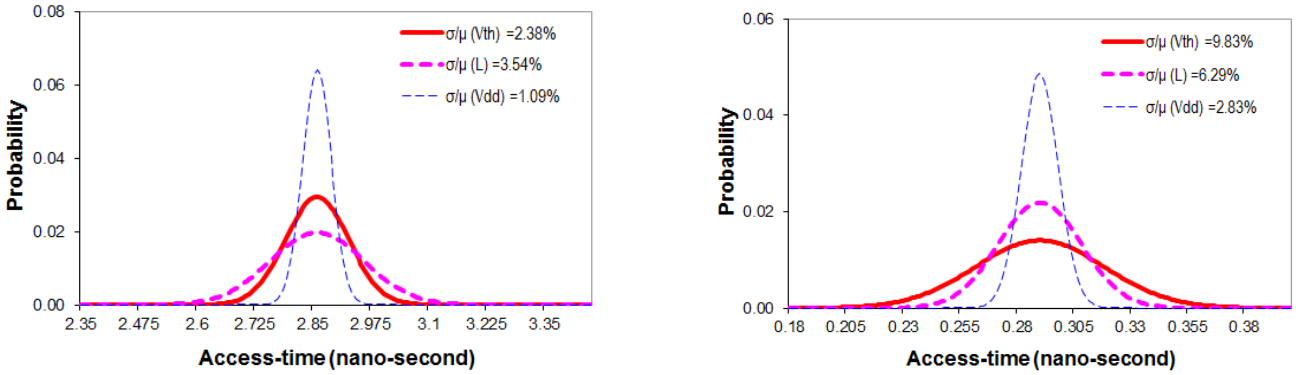

FIGURE 15. Individual Distribution of Access-time according to VAR-TX for (a) 180-nm 64KB SRAM and (b) 16-nm 64KB SRAM.

D2D variations of 4% for *Vth* and *L*, and 2% for *Vdd*).

Several observations follow from Fig. 14:

1. SRAM access-time variation is a function of SRAM size and organization.

2. The results show that the cumulative probability of access-time variation grows with an increase in row size. The lowest row width (curve C1, 4×64, showing the smallest $\sigma = 9.2\%$) displays a much lower cumulative probability than the largest row width (curve C2, 32×64, showing the largest $\sigma = 13.19\%$).

3. Comparing the different SRAM sizes, we can deduce that as area is increased, the cumulative probability of variation is only slightly larger. This holds not just for access-time but for power as well (not shown).

4. Comparing the PDF traces of access-time of 16-nm SRAMs with 45-nm and 180-nm SRAMs reveals that the variation of SRAM increases with technology scaling.

C. INDIVIDUAL Vth, L, & Vdd VARIATIONS

Two interesting observations follow from Table 3, which lists the individual impacts of transistor threshold voltage, transistor length, and supply voltage variations on the access-time variation for three different technology generations and several different SRAM sizes:

1. Variation of the access-time due to *Vth* variation is much larger for the newer nodes than for the older nodes. For example, whereas the variation of access-time due to *Vth* variation barely reaches 2.5% in 180-nm 64KB, it easily exceeds 7.5% and 9.8% in 45-nm 64KB and 16-nm 64KB, respectively. A similar trend in the 32-nm node is observed elsewhere [11].

2. Scaling the technology from the older node (180-nm) to the newer nodes (16-nm and/or 45-nm) shifts the main contribution to the variation in access-time from *L* variation (about 3.6% in 180-nm 64KB) to *Vth* variation (about 9.83% in 16-nm 64KB). Scaling from the larger to the smaller technology impacts *Vth* variation drastically, while the variation of access-time due to *Vdd* variation increases a modest 5% or so. The change in the relative impact of parameter variation between the technology nodes is particular to each parameter, of course. Oxide thickness reduction accounts for most of the *Vth* change; lithography improvements that allow fabrication of smaller transistors at higher precision impact *L*; and reducing the supply voltage from ∼1.8V to ∼1.1V and to ∼0.9V affects *Vdd* variability.

In Fig. 15(a), we show the probability density function (PDF) of 180-nm ACI due to WID+D2D variation for each device parameter separately (*Vth*, *L*, and *Vdd*). The mean of each distribution is aligned at 0.29 ns. Comparing the three PDFs with each other, it is clear that ACI due to *Vdd* variation has the narrowest distribution, followed by *Vth* and *L*. This difference in the widths of the three PDF curves is a direct measure of the standard deviation, and therefore variability, of ACI due to the three parameters. The 3-sigma delay due to each of these three parameters follows the same pattern, meaning that *L* causes the worst deviation in access-time for the 180-nm node.

Fig. 15(b) shows a plot similar to Fig. 15(a), except that it is for the 16-nm node and it is the PDF due to *Vth* that has the largest width. The same pattern holds for the 45-nm case (not shown). This and other similar experimental results confirm that the performance-limiting parameter in newer nodes such as 45-nm and 16-nm is not *L*, but the *Vth*.

In other words, going from older nodes to newer nodes has swapped the magnitude variability role of *L* and *Vth*. Table 3 validates our preceding discussion regarding the change in the impact of *L* and *Vth* variability on access-time, due to technology scale down. However, the effect of *Vdd* variation on access-time in newer nodes shows comparatively little change.

D. WORDLINE vs. BITLINE VARIABILITY

Fig. 16 compares wordline 3σ corner-point delay variability to bitline 3σ cornerpoint delay variability for the 16-nm node.

To allow a comparison between the upper (the slowest possible access-time) and the lower (the fastest possible access-time) 3σ corner-points to ACI, both ACH and ACL traces are shown. The horizontal axis extends from minimum modelled wordlines and maximum modelled bitlines at left to the

TABLE 3. Individual parameter fluctuations.

SRAM size (bytes)	Organization (*ACI*)	Access-Time (*ACI*), 1-sigma (in %)								
		180-nm			45-nm			16-nm		
		Vth	*L*	*Vdd*	*Vth*	*L*	*Vdd*	*Vth*	*L*	*Vdd*
128	$\frac{4:64:4}{1:1:1}$	1.82	2.71	0.84	5.78	3.68	1.65	7.53	4.82	2.17
256	$\frac{8:64:4}{1:1:1}$	2.02	3.00	0.92	6.40	4.07	1.83	8.34	5.34	2.40
512	$\frac{8:64:8}{1:1:1}$	2.06	3.07	0.95	6.54	4.16	1.87	8.53	5.46	2.45
1024	$\frac{16:64:8}{1:1:1}$	2.29	3.41	1.05	7.27	4.63	2.08	9.47	6.06	2.73
2048	$\frac{16:64:16}{1:1:1}$	2.34	3.48	1.07	7.41	4.72	2.12	9.66	6.18	2.78
4096	$\frac{32:64:16}{1:1:1}$	2.61	3.89	1.20	8.28	5.27	2.37	10.8	6.91	3.11
8192	$\frac{8:64:128}{1:1:1}$	2.24	3.34	1.03	7.11	4.53	2.04	9.27	5.93	2.67
16384	$\frac{8:64:128}{2:1:2}$	2.29	3.41	1.05	7.25	4.62	2.08	9.46	6.05	2.72
32768	$\frac{8:64:128}{4:2:2}$	2.33	3.47	1.07	7.40	4.71	2.12	9.64	6.17	2.78
65536	$\frac{8:64:128}{8:2:4}$	2.38	3.54	1.09	7.54	4.80	2.16	9.83	6.29	2.83
131072	$\frac{8:64:128}{16:4:4}$	2.43	3.61	1.11	7.68	4.89	2.20	10.0	6.41	2.88
262144	$\frac{8:64:128}{32:4:8}$	2.47	3.67	1.13	7.83	4.98	2.24	10.2	6.53	2.94
524288	$\frac{8:64:128}{64:8:8}$	2.52	3.74	1.15	7.97	5.07	2.28	10.4	6.65	2.99
1048576	$\frac{8:64:128}{128:8:16}$	2.56	3.81	1.17	8.11	5.16	2.32	10.6	6.77	3.04

Comparing the ACI (ideal access-time) 1-sigma of 16-nm (incurred due to **individual** parameter fluctuations) with those of 180-nm and 45-nm for different SRAM-sizes.

reverse case of maximum wordlines and minimum bitlines at right. Several interesting observations follow from Fig. 16:

1. We see that delay variability in the large-bitline cases substantially exceeds delay variability in the large-wordline cases. One can control long-bitline variability to a degree, with well-chosen bitline segmenting. Control over long-wordline variability is harder to achieve since oxide thickness—and therefore *Vth*—varies across long wordlines, and is especially problematic at the extreme ends.

2. The decrease in SRAM delay variability with a larger number of wordlines comes at a price, however. When there are more than the optimal number of wordlines (more than 128 in our 6T-SRAM), access-time climbs steeply. This effect is less pronounced for the 45-nm and 180-nm nodes (not shown). The physical parameter most responsible for the variation difference between the nodes is the 50% reduction in

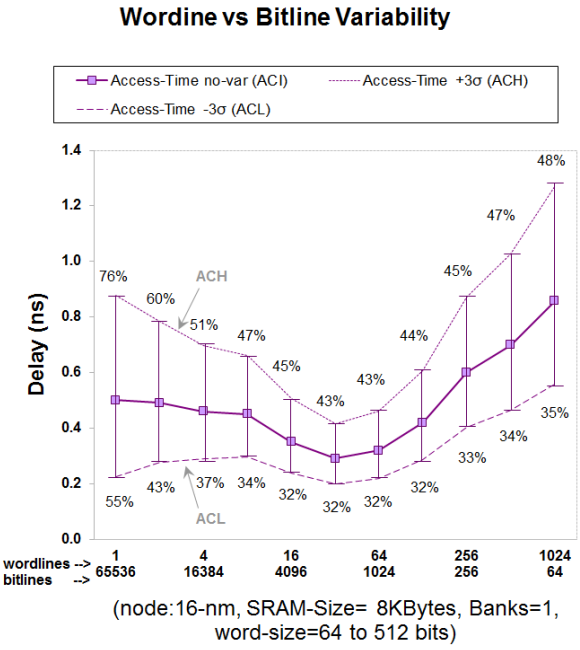

FIGURE 16. Wordline vs. Bitline 3σ corner-points (ACH and ACL) Variability of 16-nm SRAM.

oxide thickness in going from a larger node to a smaller node.

3. The upper 3σ variation ACH is always larger than its corresponding lower 3σ variation ACL. This means the access-time of SRAM is more likely to increase than decrease due to process variations.

4. The rise of the access-time on the extreme left-end of the ACI trace is due to the large number of columns (i.e., 128) used in non-optimum organizations (i.e., $\frac{128:512:1}{1:1:1}$). Similarly, the rise of the access-time on the extreme right-end of the ACI trace is due to the large number of wordlines (i.e., 1024) used in some other non-optimum organizations (i.e., $\frac{1:64:1024}{1:1:1}$).

5. Both bitline and wordline variability fall to a minimum in the middle of the plot, where optimum organizations (i.e., $\frac{32:64:32}{1:1:1}$) are found. This finding further validates our VAR-TX model.

E. BANK VARIABILITY

In general, the variability of large SRAMs declines when SRAMs are divided into smaller sized banks. Such decline in variability, however, comes at the expense of increased delay, power, and area in most cases.

Fig. 17 shows how access-time variability improves in the 16-nm 64KB 6T-SRAM when more banks are used. Similar, but smaller, improvements are seen for the 45-nm (32% less) and 180-nm (70% less) nodes as well [31]. There are two different sets of plots in Fig. 17, each set composed of three traces. One set (purple traces) represents an organization with a wordwidth of 64 bits and the other set (orange traces) represents an organization with a wordwidth of 1024 bits. In both sets, the ACI trace represents the ideal access-time with no variability, and the +sigma and −sigma traces represent the

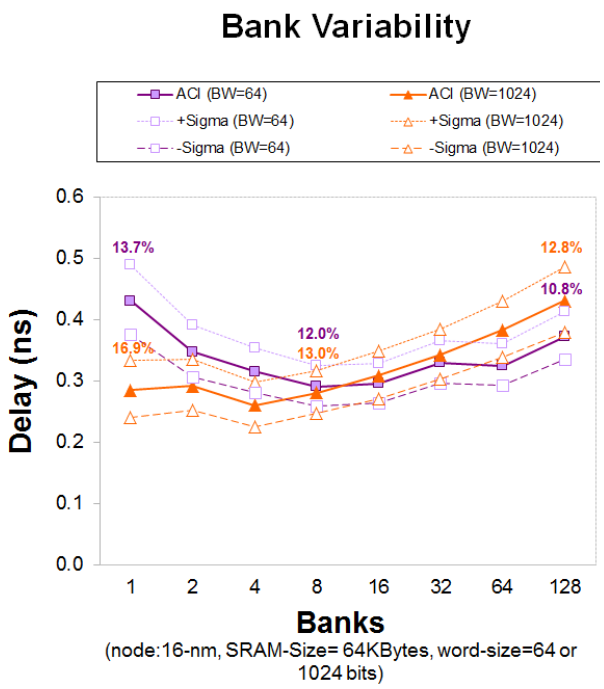

FIGURE 17. Bank Variability; Access-time variation vs. number of banks—illustrating 1-sigma corner points (+sigma, −sigma) variability of ACI (ideal access-time) for two different organizations (wordwidth=64bits and wordwidth=1024bits) for a 16-nm 64KB SRAM divided into 1 to 128 bank.

upper variation (slower) access-time and the lower variation (faster) access-time, respectively.

Looking at Fig. 17, we can observe that an increase in bank number from 1 to 128 leads to a decrease in ACI variation, as the upper variation of ACI ($+\sigma$ trace) decreases from 13.7% to 10.8% for BW=64 (purple) and from 16.9% to 12.8% for BW=1024 (orange).

This means SRAM variability decreases as the number of banks increases. The main reason SRAM variation ($\pm\sigma$) declines with a larger number of banks is related to the smaller number of bitlines used in smaller banks. A smaller number of bitlines means shorter wordlines. This, in turn, means more correlation and a smaller possibility of a mismatch between the 6T-cells on the same short wordlines, and also a smaller probability of variation between the transistors in those cells. Other factors, such as a smaller number of rows, smaller area, and smaller loading effects on the output bus used in smaller sized banks, are also among the reasons why SRAM variation ($\pm\sigma$) declines with a larger number of banks. However, the impact of these secondary factors on variability is not as much as that incurred by the bitlines. Our model's distance correlation incorporates the increase in the probability of oxide thickness variation as wordlines and bitlines increase in length.

The price for reducing overall SRAM variation by raising the number of banks (above the optimum number of banks) is a considerable area increase, a tolerable power increase, and a delay increase due to output bank loading. This trade-off phenomenon is shown in Fig. 17 and is illustrated by PDF curves in Figs. 18(a) and 18(b). Fig. 17 shows that the access-time for both cases starts and/or continues to increase as the 64kB SRAM is divided into more than 8 banks. Similarly, Fig. 18(a) and 18(b) show that the access-time (or the mean) of the curves x_{128} and y_{128} (having smaller sigma) is larger than the curves x_8 and y_8 (having moderately larger variation). Simply put, the price for ∼3% improvement in variability is about 28% decline in speed.

Designers should also be aware of potential SRAM yield declines when the number of banks increases. The latter effect stems from the increased hardware-failure probability with larger transistor numbers.

Table 4 summarizes the mean and standard deviation of the distribution of access-time (PDF) for the aforementioned two different organization sets (wordwidth=64bits and wordwidth=1024bits), as the number of banks is swept from 1 to 128. Looking at Table 4, we can see that the reduction in sigma generally corresponds to an increase in mean and vice versa.

Table 4 also shows that the variability of SRAMs using narrower wordwidth architecture (i.e., wordwidth=64 bits) is less than the variability of SRAMs using wider wordwidth architecture (i.e., wordwidth=1024 bits) for a same number of banks. For example, in the case of SRAM having only one bank, a sigma=13.7% for wordwidth=64 bits is less than a sigma=16.9% for wordwidth=1024 bits. Similarly, in case of SRAM having 32 banks, a sigma =10.8% for wordwidth=64 bits is less than a sigma=12.2% for wordwidth=1024 bits.

Finally, the results in Table 4 (as well as the comparison between the x_{128} and y_{128} traces in Fig. 18) reveal that both the mean and the sigma of the organizations with larger wordwidth (i.e., wordwidth=1024bits) are larger than the mean and sigma of the organizations with smaller wordwidth (i.e., wordwidth=64 bits). Such a difference, however, is less of a concern since the SRAM can typically be designed around the optimum architecture specifications. For example, the designer can pick a y_8 trace specification over a y_{128} trace specification when having a wordwidth of 1024 bits is desired.

All this means that, although overall variability decreases and overall reliability increases in SRAMs with larger bank numbers, the delay times soar and the yields decline to some degree. Luckily, such an increase in delay and decline in yield is generally acceptable in the optimally designed architecture cases, but this is not necessarily the case for non-optimally designed architectures. The best approach, therefore, is to design around the optimum architecture, where the access-time is close to the smallest possible delay and has a modest variation. Whether or not such a balance between delay and variation offered by optimal architecture design can be tolerated will depend on the individual application.

F. TEMPERATURE IMPACT ON RELATIVE SWITCHING FREQUENCY

Temperature variation is caused by spatially- and temporally-varying factors. These variations are becoming more severe

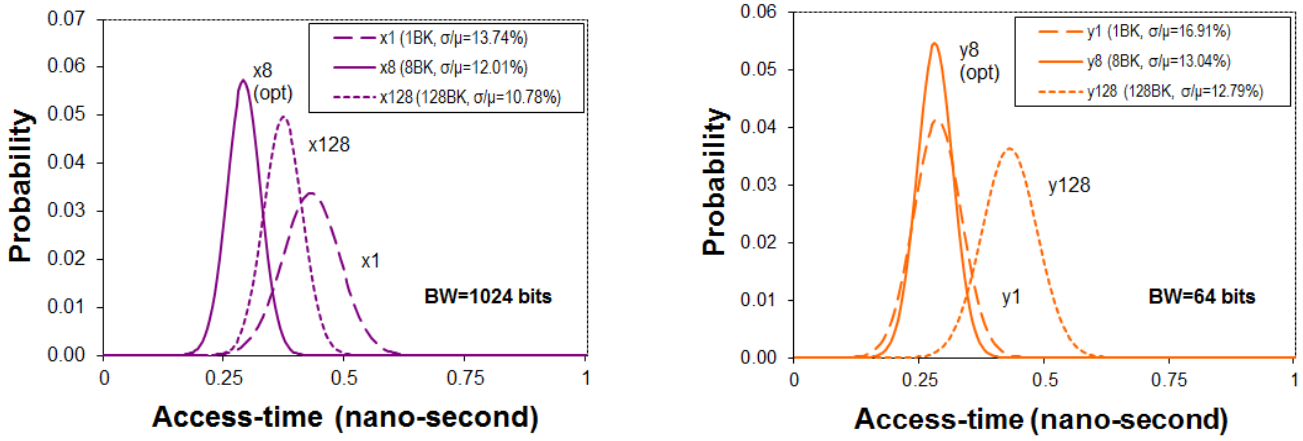

FIGURE 18. Bank Variability; illustrating the distribution of ACI (ideal access-time) for two different organizations—(a) BW=64 bits and (b) BW=1024 bits—for a 16-nm 64KB SRAM divided into 1 to 128 banks.

and harder to tolerate as technology scales to submicron feature sizes. As illustrated in Section VIC, of the three key process parameters subject to variation (*Vth*, L_{eff}, and *Vdd*), threshold voltage (*Vth*) is the most important because its variation has a substantial impact on two major properties of the SRAM/processor: the frequency it attains and the leakage power it dissipates. Moreover, *Vth* is also a strong function of temperature, which increases its variability [11].

TABLE 4. ACI for different bank numbers.

Number of Banks	Wordwidth = **64** bits (x curves)			Wordwidth = **1024** bits (y curves)		
	Organization	Mean (ns)	Std (%)	Organization	Mean (ns)	Std (%)
1	32 : 64 : 256 / 1 : 1 : 1	0.43	13.7	16 : 1024 : 32 / 1 : 1 : 1	0.28	16.9
2	32 : 64 : 128 / 2 : 1 : 2	0.35	12.6	8 : 1024 : 32 / 2 : 1 : 2	0.29	14.6
4	32 : 64 : 64 / 4 : 2 : 2	0.31	12.2	8 : 1024 : 16 / 4 : 2 : 2	0.26	14.5
8	32 : 64 : 32 / 8 : 2 : 4	0.29	12.0	4 : 1024 : 16 / 8 : 2 : 4	0.28	13.0
16	16 : 64 : 32 / 16 : 4 : 4	0.29	11.3	4 : 1024 : 8 / 16 : 4 : 4	0.31	13.1
32	8 : 64 : 32 / 32 : 4 : 8	0.33	10.8	2 : 1024 : 8 / 32 : 4 : 8	0.34	12.2
64	8 : 64 : 16 / 64 : 8 : 8	0.32	11.0	2 : 1024 : 4 / 64 : 8 : 8	0.38	12.4
128	4 : 64 : 16 / 128 : 8 : 16	0.37	10.8	2 : 1024 : 2 / 128 : 8 : 16	0.43	12.8

Analysis of Mean and standard deviation of Ideal Access-Time (ACI) for two different organizations, one with Wordwidth=64 bits and the other with Wordwidth=1024 bits, in 16-nm SRAMs of different bank numbers.

The simulated plots shown in Fig. 19 (for 16-nm) agree with the corresponding results in VARIUS [11], except that the declining slope of the relative switching frequency due to a temperature increase is almost twice as steep as for VARIUS (32-nm). Such a high rate in the relative switching frequency

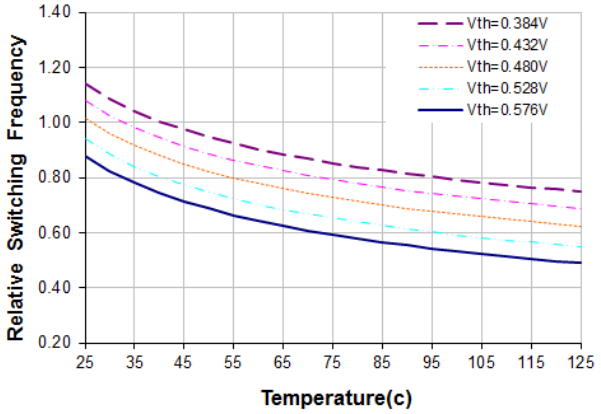

FIGURE 19. Relative switching frequency versus temperature for different threshold voltages. We use *Vth* = 0.480V at 27 °C (room temperature) as reference.

is most likely due to the existence of a larger drain current, larger wire resistance, and larger junction capacitance (C_j) inherent in the smaller 16-nm technology node modeling.

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 (i.e., 16-nm) 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 [27].

As we discussed in the first three sections, the important first step to redress this trend is to understand how parameter variation affects the timing errors in high-performance SRAMs and processors. Based on this, we can devise techniques to cope with the problem—hopefully recouping the full gains offered by every technology generation. Several of

our references (especially [31]) have attempted to accomplish this task by presenting several recent advanced techniques that can either remedy or minimize such adverse effects on the chip performance. One example is runtime adoption techniques that can be employed to balance the workload throughout the many core system and thus decreasing peak temperatures and gradients [34]. We incorporated most of these suggested techniques in the design for our simulations to produce both the data needed for our VAR-TX modelling and the plots presented in this section.

Here, we present two plots that illustrate the impact of temperature on frequency.

As discussed in Section IVB, the impact of temperature on delay is not as dramatic as the impact of temperature on leakage power. Fig. 19 illustrates the impact of temperature on the relative switching frequency for a 16-nm 64KB SRAM. As we can see in Fig. 19, the dependence of temperature on the relative switching frequency is not very strong. All five plots (with the middle one *Vth*=0.220V used as reference) follow a similar modest decreasing trend.

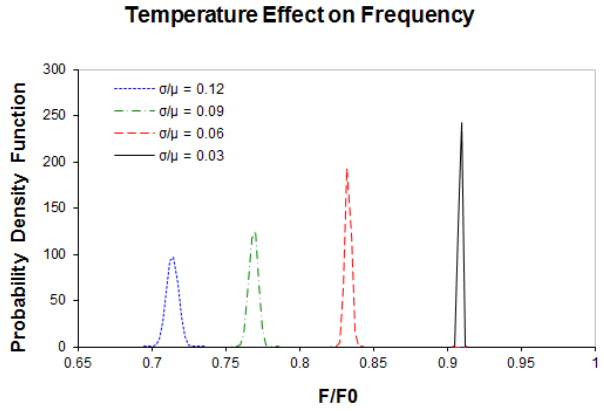

FIGURE 20. Probability distribution of the relative chip frequency as a function of *Vth*'s σ (varied due to temperature change). We use *Vth* = 0.480V at 27 °C, 27 gates in the critical path, and 2000 critical paths in our 16-nm 64KB SRAM design.

Fig. 20 shows the effect of temperature on the frequency of SRAM. Assuming that every critical path in an SRAM consists of n_{cp} gates and that a modern SRAM chip has hundreds of critical paths, Bowman et al. [27] compute the probability distribution of the longest critical path delay in the chip ($\max\{T_{cp}\}$). Such a path determines the SRAM frequency ($1/\max T_{cp}$). Using this approach, we find that the value of *Vth*'s σ (resulting from a variation in temperature) affects the chip frequency.

Fig. 20 shows the probability distribution of the chip frequency for four different values of *Vth*'s σ. A smaller σ represents a smaller variation in *Vth* due to smaller range of possible temperature changes (i.e., 27 °C to 50 °C) and a larger σ represents a larger variation in *Vth* due to larger range of possible temperature changes (i.e., 27 °C to 125 °C).

The frequency (F) is normalized to the case of an SRAM without *Vth* variation (F_0).

The PDF curves in Fig. 20 show that as σ increases the mean chip frequency decreases and the chip frequency distribution becomes more spread out. In other words, given a batch of chips, as the σ of *Vth* increases, the mean frequency of the batch decreases and at the same time, an individual chip's frequency deviates more from the mean.

G. NBTI SIMULATION RESULTS AND ANALYSIS

The analysis of NBTI is inherently more complicated than that of other traditional reliability issues, such as hot-carrier injection (HCI) [82], [83]. In addition to its dependency on supply voltage and temperature, NBTI exhibits the unique property of having distinct stress and recovery behavior during a circuit's dynamic operation, as illustrated in the following two subsections.

1) SUPPLY VOLTAGE AND TEMPERATURE DEPENDENCE OF NBTI

NBTI has strong dependence on *Vdd* and T [61], [84]. The nominal *Vdd* is assumed to be 0.7V and the nominal T is 80 °C. The data for the 16-nm *Vdd* and T profiles are extrapolated from data for an industrial 65-nm design provided by Wang et al. [2]. Based on the extrapolated data, the variations of *Vdd* and T for the whole chip are assumed to be within 10% for NBTI analysis. For the purpose of circuit timing analysis, we follow Cao's method [2] and select five representative operating conditions with different combinations of *Vdd* and T: high *Vdd* and high T (HH), low *Vdd* and low T (LL), high *Vdd* and low T (HL), low *Vdd* and high T (LH), and normal *Vdd* and normal T (NN). In order to analyze the temperature dependence in a wider range, we also include one more condition: low *Vdd* and room temperature (LL'). Using the formula, algorithm, and procedure outlined by Cao [2], we obtain the delay degradation for our different SRAM circuits after one year, five years, and ten years of stress—as illustrated in Table 5.

TABLE 5. NBTI dependency on supply voltage and temperature.

Circuit	1 Year (%)					
	NN	HH	LH	LL	HL	LL'
(arc N)	9.9	11.8	11.5	9.5	9.2	6.1
(arcO)	6.7	7.3	7.7	6.5	6.3	4.4

Circuit	5 Year (%)					
	NN	HH	LH	LL	HL	LL'
(arc N)	13.9	15.4	16.3	13.1	12.6	8.1
(arcO)	8.8	9.6	10.1	8.4	8.2	5.7

Circuit	10 Year (%)					
	NN	HH	LH	LL	HL	LL'
(arc N)	16.2	17.8	19.2	14.1	14.7	10.5
(arcO)	9.9	10.8	11.2	9.6	9.3	6.4

Simulation results for two 16-nm SRAM circuits: arcN (non-optimum, 4:64:256 / 1:1:1) and arcO (optimum, 64:64:16 / 1:1:1).

From Table 5, we make the following three important observations for dynamic circuit operation:

1. Temperature has a bigger impact on the degradation of circuit performance than the operating supply voltage. For instance, after ten years of stress, the delay degradation of circuit arc N is about 19.2% under the LH condition, while it is about 14.1% under the LL condition. The degradation difference caused by temperature is about 5%. If we further reduce T to room temperature, the delay degradation can be reduced to about 10.5%. Therefore, lowering the temperature is a very effective approach to minimize NBTI.

2. Within the allowed 10% voltage variation, tuning the operating *Vdd* does not show any advantage in reducing NBTI. For example, the delay degradation of circuit arcO is about 7.7% under the LH condition, while it is 7.3% under the HH condition. The degradation difference caused by voltage is only 0.4%.

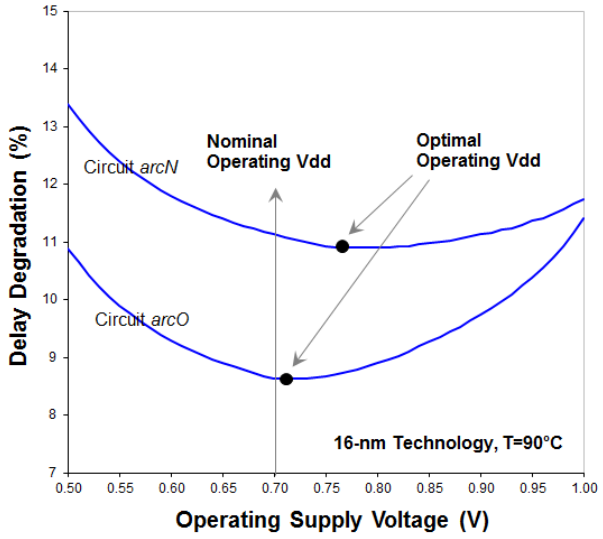

FIGURE 21. Optimal *Vdd* for minimum degradation of circuit performance for two different 16-nm SRAM architectures: optimal ($\frac{64:64:16}{1:1:1}$) and non-optimal ($\frac{4:64:256}{1:1:1}$).

3. Although lower operating *Vdd* is generally preferred to reduce the amount of circuit aging, this does not hold true for scaled CMOS design, as observed in our simulation results. On the contrary, lower operating voltage may lead to more circuit timing degradation for the 16-nm technology node, as shown in Fig. 21. Given the stress time, there exists an optimum operating *Vdd* that achieves the minimum amount of circuit delay degradation. When *Vdd* is lower than that value, circuit performance becomes increasingly sensitive to changes in *Vth*, and thus, the degradation rate climbs even though the absolute increase of *Vth* is smaller than that at higher *Vdd*. On the other hand, when *Vdd* is higher than that value, the amount of *Vth* increases exponentially, dominating the performance degradation. The exact value of the optimum operating *Vdd* also depends on the technology node and the circuit structure.

In summary, during dynamic operation, NBTI-induced degradation is relatively insensitive to supply voltage, but strongly dependent on temperature. In addition, there is an optimum supply voltage that leads to the minimum circuit performance degradation; the circuit degradation rate actually goes up if the supply voltage is lower than that optimum value. Since our simulation results agree with those of [2], we have confidence that the NBTI analysis presented in this section is valid.

2) INPUT CONTROL IN STATIC AND DYNAMIC OPERATION

In addition to the dependence on *Vdd* and T, NBTI has an optimum gate voltage, as well. For a pMOS, a gate bias at *Vdd* helps the recovery, while a gate bias at "0" stresses the transistor. A longer time spent in recovery (i.e., lower duty cycle) corresponds to smaller changes in *Vth* for the transistor. Because of this mechanism, NBTI is strongly affected by node activity. In standby mode, this implies a dependence on input patterns; during the dynamic operation, the duty cycle further impacts the relative time between stress and recovery [2].

Depending on the input patterns and duty cycle, over 75% of previous NBTI-induced degradation can be annealed by biasing the pMOS gate at the supply voltage (*Vdd*) [61], [84]. Therefore, the recovery phase and its dependence on node switching activity are critical to the analysis and design margining for the NBTI-induced degradation. The *Vth* change under dynamic conditions is dramatically different from that in the static mode. Because of the rapid annealing at the beginning stage of the recovery even a small recovery period (i.e., signal probability close to 1) greatly reduces the overall degradation by more than 50% of the static stress. This property is confirmed by silicon data [85], [86] and experimental results. Therefore, an accurate prediction of performance degradation should include not only *Vdd* and T, but also the switching activity of the node. These parameters are neither spatially nor temporally uniform, but vary significantly from gate to gate and over time due to the uncertainty in circuit topologies and operations. These non-uniformities need to be incorporated into the degradation analysis for both short-term and long-term predictions. Otherwise, a simple static analysis may provide an extremely pessimistic estimation, and consequently, result in drastic over-margining. So far, design and tool research are in the early stages of addressing emerging reliability needs [2]. The impact of static NBTI on the performance of combinational circuits was analyzed by P. Bipul et al. [87]. These authors demonstrate that by resizing the paths that are most sensitive to NBTI, it is possible to mitigate the increase of path delay of the entire circuit. They show, on average, an increase of 8.7% in circuit size is required for 70-nm technology. An algorithm for determining the amount of delay degradation of a circuit due to NBTI is provided by S. Kumar et al. [88].

Using the data flow and structure of the Framework introduced by W. Wang et al. [2] (Fig. 22), we estimated the delay degradation due to NBTI for our 16-nm design. The temporal degradation of our circuit performance had a dependency on both technology and design conditions. First, we made an accurate model of *Vth* degradation at the transistor level. For

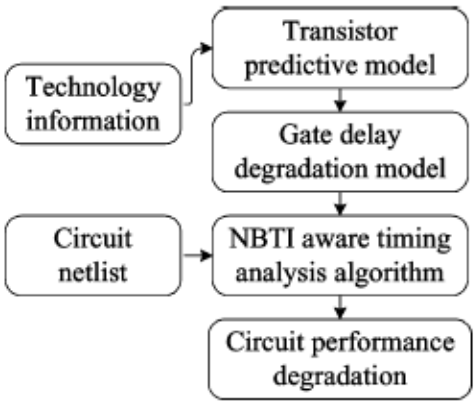

FIGURE 22. NBTI timing analysis framework [2].

NBTI, we used predictive transistor models to characterize the timing behaviour of various basic circuit building gates, such as NAND and NOR gates. An NBTI-aware library could then be built upon these predictive models. Given a circuit netlist, the new library further supported a timing analysis algorithm that is a simple and efficient way to calculate the circuit performance degradation. By including transistor-level modelling of other reliability mechanisms, such as HCI and NCS, the aforementioned framework is extendable to analyze other aging effects.

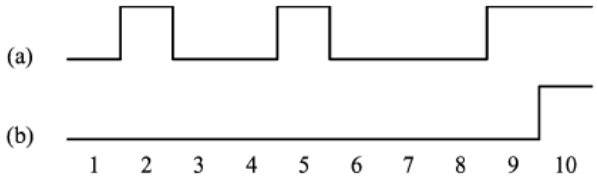

FIGURE 23. Random input sequence. (a) Normal case. (b) Extreme case [2].

Fig. 23(a) shows one single typical random input sequence within a ten-cycle period—in which there are n "0"s and $(10-n)$ "1"s. An extreme case of such a random sequence is shown in Fig. 23(b). This input vector has only 1 flip within ten cycles, i.e., α is equal to 0.9. This means that the stress time is much longer than the recovery time. Here, the term of $[\alpha/\min(\alpha, 1-\alpha)]$ is defined to capture how many cycles are spent in the stress phase. In the case of Fig. 23(b), this term is equal to 9.

The simulation results for nodes smaller than 70-nm (i.e., 65-nm and 45-nm) show that there is around an 8% delay degradation in combinational logic circuits after ten years of stress [2]. The simulation results for our 16-nm SRAM circuits show a higher delay degradation of around 10.3% after ten years of stress. The expected higher percentage of delay degradation for 16-nm, as compared to those of the larger technology nodes, is due to the smaller oxide thickness, stronger electric field, etc. used in the 16-nm node.

It is imperative to note that despite a considerable amount of work performed so far by many researchers such as W. Wang et al., an accurate and comprehensive understanding of NBTI is still not available to guide reliable design to minimize its impact. Consequently, the results shown in this section should only be considered rough estimates, rather than accurate predictions of circuit aging.

The analysis of our findings regarding the impact of multiple inputs and duty cycle on NBTI degradation is presented as parts A and B in the remainder of this subsection.

a: INPUT PATTERN DEPENDENCE (STATIC MODE)

For a circuit containing n inputs, each input signal can be either set to "1" or "0" during the standby mode. Thus, the circuit can have at most 2^n possible input patterns. Since NBTI has a strong dependence on the input pattern of the circuit, different input patterns will result in significantly different delay degradations. An input vector that results in the least delay degradation of the circuit is referred to as the best standby mode. Similarly, an input vector that results in the most delay degradation is referred to as the worst standby mode. Much like Wang et al. [2], we estimate the best and worst standby modes by sampling the circuit with 500 different input vectors. By biasing several selected SRAM circuits under the worst and the best standby modes, we compare their delay degradations for one, five, and ten year periods and record the results. Based on the results, we see that the delay degradation caused by NBTI can be greatly reduced by applying the optimal input pattern to the entire circuit in the standby mode. A typical example is circuit $\{\frac{8:64:128}{1:1:1}\}$. After ten years, the delay degradation for the worst standby mode is about 46%, while under the best standby mode it is about 11%. The delay degradation can change by more than a factor of 4 for different input patterns. Like NBTI, the leakage current of a circuit also has a strong dependence on the input pattern. Therefore, if the application in which the SRAM is used allows a set of pre-selected input patterns in the standby mode, both the temporal degradation caused by NBTI and the circuit leakage can be minimized. Again, this result is validated by its similarity to the results of Wang et al. [2].

b: DUTY CYCLE DEPENDENCE (DYNAMIC MODE)

For a circuit operating in the dynamic mode, the probability that each input can take a value of "1" or "0" can be any value between 0 and 1. For a given circuit with n inputs, α_i, i ε $\{1, \ldots, n\}$, is the duty cycle of input i. Like Wang et al. [2], we define one combination of $\{\alpha_1, \alpha_2, \ldots \alpha_n\}$ as one α set. Since for an n-input circuit, the number of distinct α sets can be infinite, we choose five typical values in order to analyze the impact of different α sets on the circuit performance: 0.1, 0.3, 0.5, 0.7, and 0.9 for each α_i. Thus, in the α sets, all α are set to either 0.1, 0.3, 0.5, 0.7, or 0.9.

To observe how the duty cycle affects the delay degradation of circuits over time, we apply the formula/methods outlined by Cao et al. [2]. We use three different α sets on two of our selected SRAM architectures (where I stands for $\{\frac{4:64:256}{1:1:1}\}$

and II stands for $\{\frac{64:64:16}{1:1:1}\}$. We observe that, within the same architecture, different α sets can result in very different timing degradation. For example, after one year of stress, the delay degradation of circuit I (the bottom curves, Fig. 24) with an input duty cycle of α set3 is nearly 2× larger than that with α set1. In addition, the difference in delay degradation (Δ) increases with time, i.e., $\Delta 2$ is much larger than $\Delta 1$. As mentioned previously, NBTI is clearly related to the gate bias due to its exponential dependence on the electrical field. Therefore, for a circuit operating in the dynamic mode, NBTI-induced degradation can be reduced by adjusting the input signal α such that it stays in the recovery state longer.

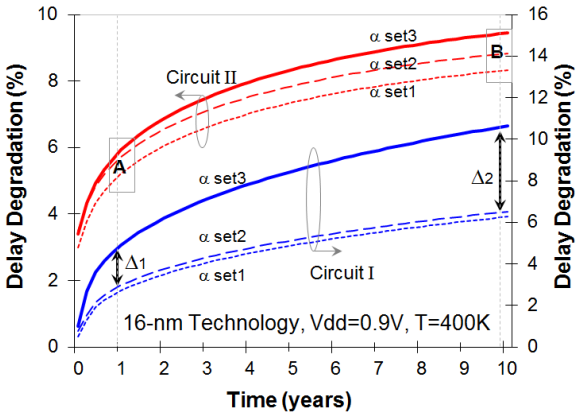

FIGURE 24. Delay degradation over time for various duty cycle sets of two sample circuits (circuit I and circuit II).

Fig. 24 illustrates that the delay degradation profile of the circuit after ten years has a much wider spread than the degradation after one year. That is, with increasing time, different α sets tend to generate diversified effects on the circuit degradation. In other words, several α sets might result in a similar circuit delay degradation in a short time period. However, in the long term, they can result in very different degradations.

Furthermore, our experiments (in accord with [2]) show that using a higher number of input α sets tends to generate a larger timing degradation and a wider path delay distribution in the long run. This is because different input αs result in very different ΔVth, which correspondingly leads to wide distribution of circuit path timing. Therefore, modulating node activities will be a very useful design tool to mitigate NBTI for dynamic operation. Modulating node activities, however, has its own limitations and comes with some disadvantages if used for critical paths in SRAM circuits. For example, the reduction of duty cycles on such signals as WRITE, WL (wordline pulse), or CLS (pulse for senseamp) can reduce the stability and robustness of SRAM.

In summary, circuit performance degradation due to NBTI is highly sensitive to input vectors. The difference in delay degradation could be up to 5× for various static and dynamic operations.

H. SRAM YIELD-ESTIMATION MODEL

The D2D and WID variations and, hence, failure probability (P_F) of SRAM is directly related to the yield of the memory chip [10]. To estimate the yield, we use Monte Carlo simulations for D2D distributions of *Vth*, *L*, and *Vdd* (assumed to be Gaussian) in our model. An embedded algorithm takes the result of the Monte Carlo simulation along with the given desired maximum power and area to determine the optimum yield. The algorithm discards the delays not meeting both of the required maximum allowable power and area and selects the smallest delay meeting both the given desired total power and area. For each D2D value of the parameters (say Vth_{D2D}, L_{D2D}, and Vdd_{D2D}) we estimate probability failure ($P_F = 1 - CDF$) considering the WID distribution of ΔVth, ΔL, ΔVdd, where CDF is the cumulative distribution function. Finally, the yield is defined as expressed by Mukhopadhyay [10].

$$Yield = 1 - \left(\frac{\sum_{D2D} P_F\left(Vth_{D2D}, Lgate_{D2D}, Vdd_{D2D}\right)}{N_{D2D}} \right) \quad (20)$$

where N_{D2D} is the total number of D2D Monte Carlo simulations (i.e., total number of chips). An increase in the WID variation (i.e., $\sigma_{Vth}, \sigma_L, \sigma_{Vdd}$) increases the memory-failure probability, thereby reducing the yield.

This means that without proper cell transistor sizing and careful choice of SRAM architecture, yield can suffer significantly. For example, using close to minimum size width for the pull down transistors of each 6T-cell can increase both the read delay and delay variation. Similarly, increasing the number of cells in a column increases capacitance and leakage current of bitlines and also increases the access-time, resulting in an increase in P_F, and therefore a decline in yield. Hence, for yield enhancement the cell configurations and the memory architecture need to be optimized, considering a given minimum area and power constraints. In this estimation, we have assumed a standard deviation of 4% for D2D distribution of *Vth* and *L*, and 2% for *Vdd*.

Table 6 shows the yield results for 16-nm 64KB SRAM, using the method expressed by Eq. (20). A similar trend holds for all other sizes of 16-nm SRAM—which is about 3% and 5% lower than the trend observed in 45-nm and 180-nm, respectively. To quantify the approximation error empirically, we compared the results obtained from our model with the empirical results obtained from our actual transistor-level SRAM circuits. The approximation error was below 8%.

TABLE 6. SRAM yield before and after optimization.

	Architecture	Area	Power	Tac-time	Yield
Initial Design (scaled from 50-nm) [10]	$\frac{8:64:256}{1:1:1}$	41 mm²	0.885 W	0.42 ns	57%
Empirically Optimized Designed SRAM	$\frac{64:64:16}{1:1:1}$	44 mm²	0.939 W	0.29 ns	93%

Unless new advancements are made, the estimated yields shown in Table 6 are expected to decline when the impact of other variability factors such as metal gate granularity (MGG), Bias Temperature Instability (BTI), Trap-Assisted Tunnelling (TAT)—causing gigantic random telegraph noise (RTN) [99]—and strained silicon effect [95] are included in Eq. (20). While these factors decrease the yield, the growing self-tracking and adaptive design techniques can help improve the yield by tracking and subsequently mitigating the factors responsible for the post-silicon variation.

VII. ONGOING AND FUTURE WORK

Variability has typically been addressed by process, device, and circuit designers (traditionally called hardware designers) and rarely by software designers. Recent studies show that many of hardware-related variability issues can be addressed by software-related techniques, which could lead to increased chip yields at lower costs [29], [89].

In the future, better integration and collaboration between hardware and software will be required. As technology trends force building toward typical-case designs, error detection and recovery mechanisms will become pervasive both in the microprocessor and system on chip designs. To sustain continued increases in performance, we must identify and develop new machine organizations that are capable of dynamically detecting and recovering from errors in the field across all layers of the computing stack, including computer architecture, system software, and applications. The development of such new machine organizations provides two benefits:

1. The performance inefficiencies that arise at each layer from maintaining strict abstraction between hardware and software are eliminated, and

2. Power and area overheads that arise from the use of circuit- and microarchitectural-level techniques that mitigate the various sources of failures (Section III) are eliminated as well. For example, clock gating in high-performance processors can cause unacceptable stresses on the power delivery network. Although the resulting inductive noise, or Ldi/dt, can be handled using mechanisms that reduce voltage swings caused by large dynamic current, those mechanisms incur their own set of problems and performance impacts [18].

Simply put, we must look for ways to build reliable systems from unreliable devices using cross-layer solutions. We need a result obtained through a reactive approach to the design of high-performance, low-cost processors that, as these authors [18] put it, resembles the children's game ''Whack-AMole'': see a problem, design a solution, optimize the solution, look for the next problem, and repeat.

One major study, called ''Expedition,'' has made some efforts to handle variability at higher layers of abstraction [29]. For instance, software schemes have been used to address voltage [90] or temperature variability [91].

The Expedition research attempts to holistically and proactively exploit hardware variability across multiple abstraction levels, as well as across different subsystems of computing platforms. The exploitation of such cross-layer techniques is planned to continue into visionary ones, like future on-chip systems that will, expectedly, evolve into complex Cyber-physical Systems-on-Chip technology [34].

The Expedition project is a part of a larger push for the Under-designed and Opportunistic (UnO) computing paradigm [29]. UnO machines make a paradigm shift away from a traditional ''crash-and-recover approach to errors'' towards an approach that makes proactive measurements and predicts parametric and functional deviations to ensure continued system operation and availability. This will preempt impact on software applications, rather than just reacting to failures (as is the case in fault-tolerant computing) or under-delivering along power/performance/reliability axes.

In other words, instead of guardbanding systems—which partially masks the presence of variability at the cost of over-design with less than optimal power and performance—Expedition encourages device manufacturers and designers to build variation-aware software stacks that may adapt and opportunistically exploit said variations to increase system performance and minimize power consumption. The major application of the proposed software stacks is the memory subsystem—which is one of the largest components in today's computing system, a main contributor to the overall power consumption of the system, and therefore one of the most vulnerable components to the effects of variations (e.g., power). Through their suggested memory management strategy, the authors of Expedition suggest opportunistically exploiting the hardware variations in on-chip and off-chip memory at the system level through the deployment of variation-aware software stacks. For the envisioned UnO machines challenges related to the sensing infrastructure, software interfaces, and modeled hardware, the interested reader can consult our references [24], [29].

VIII. CONCLUSION

In this paper, we confronted the most important variation and reliability issues impacting the performance of on-chip 6T-SRAMs of today, and especially that of the next generation node with a two-pronged approach:

1. Using our VAR-TX model

- We significantly expanded on our recently published model VAR-TX [1]—that considers both D2D and architecture-dependent, spatially-correlated WID variations of device threshold voltage (Vth), length (L), and supply voltage (Vdd)—by presenting a number of new experimental simulation results to help predict and therefore minimize the delay and delay variation.
- We provided a quick overview of 6T-SRAM cell design challenges and briefly reviewed our model assumptions and implementation.

- We classified the major type of variations and presented the main causes for failure.
- We presented a comprehensive analysis of selected simulation results regarding access-time to help predict and thereby minimize the impact of process, operation, and temperature variations (PVT) on SRAM variability.
- We showed that how selecting the optimal architecture can increase the yield in SRAM.
- We also showed that perfectly-square banks do not necessarily lead to minimum access-times.

2. Using others' models

- We presented several other new experimental simulation results (generated not by VAR-TX but by our slightly modified versions of some other well-received models to illustrate and mitigate the impact of NBTI (Negative Bias Temperature Instability) on the aging of 16-nm 6T-SRAMs.
- Additionally, we discussed several other key reliability issues, such as SNM (Static Noise Margin), IR-Drop, Ldi/dt, EM (electromigration), etc., and provided the corresponding mitigation techniques. While some of these topics were aimed at reducing the variability and increasing stability, reliability, and robustness of 6T-SRAM, some others intended to keep the associated power and energy in check while trying to increase the speed.
- Finally, we highlighted several active research projects —such as Hardware and Software Collaboration—that we believe are the prominent subject matters crucial for further future variability minimization and reliability improvements.

We tested the high accuracy of our simulation results (show to be within 8% of Hspice results) and validated them by comparing our results with Monte Carlo simulation and access-time method discussed by Mukhopadhyay and VARIUS [10], [11], using the method explained in our prior work [1].

Such a two-pronged approach helped us predict and therefore minimize the impact of all three different types of variations (namely Operational, Fabrication, and Implementation) on the performance of our 6T-SRAMs.

Researchers shall develop even more effective variation-tolerant techniques for random and systematic variations in order to make the future node designs more tolerant to those unwanted variations—which can result from the manufacturing process, aging, or operational conditions [18].

REFERENCES

[1] J. Samandari-Rad, M. R. Guthaus, and R. Hughey, "VAR-TX: A variability-aware SRAM model for predicting the optimum architecture to achieve minimum access-time for yield enhancement in nano-scaled CMOS," in *Proc. 13th ISQED*, Santa Clara, CA, USA, Mar. 2012, pp. 506–515.

[2] W. Wang, S. Yang, S. Bhardwaj, S. Vrudhula, T. Liu, and Y. Cao, "The impact of NBTI effect on combinational circuit: Modeling, simulation, and analysis," *IEEE Trans. Very Large Scale Integration (VLSI) Syst.*, vol. 18, no. 2, pp. 173–183, Feb. 2010.

[3] A. B. Kahng, L. Bin, P. Li-Shiuan, and K. Samadi, "ORION 2.0: A fast and accurate NoC power and area model for early-stage design space exploration," in *Proc. DATE Conf. Exhibit.*, Apr. 2009, pp. 423–428.

[4] F. Dubois, V. Catalano, M. Coppola, and F. Petrot, "Accurate on-chip router area modeling with Kriging methodology," in *Proc. IEEE/ACM ICCAD*, Nov. 2012, pp. 450–457.

[5] J. P. Kulkarni, K. Keejong, P. P. Sang, and K. Roy, "Process variation tolerant SRAM array for ultra low voltage applications," in *Proc. 45th ACM/IEEE DAC*, Jun. 2008, pp. 108–113.

[6] P. Gupta, A. B. Kahng, P. Sharma, and D. Sylvester, "Gate-length biasing for runtime-leakage control," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 8, pp. 1475–1485, Aug. 2006.

[7] X. Zheng *et al.*, "Modeling of power delivery into 3D chips on silicon interposer," in *Proc. IEEE 62nd ECTC*, May 2012, pp. 683–689.

[8] K. Kiyeong *et al.*, "Modeling and analysis of a power distribution network in TSV-based 3-D memory IC including P/G TSVs, on-chip decoupling capacitors, and silicon substrate effects," *IEEE Trans. Compon., Packag. Manuf. Technol.*, vol. 2, no. 12, pp. 2057–2070, Dec. 2012.

[9] S. J. E. Wilton and N. P. Jouppi, "CACTI: An enhanced cache access and cycle time model," *IEEE J. Solid-State Circuits*, vol. 31, no. 5, pp. 677–688, May 1996.

[10] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 24, no. 12, pp. 1859–1880, Dec. 2005.

[11] S. R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas, "VARIUS: A model of parameter variation and resulting timing errors for microarchitects," *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 1, pp. 3–13, Feb. 2008.

[12] N. Monnereau, F. Caignet, N. Nolhier, M. Bafleur, and D. Tremouilles, "Investigation of modeling system ESD failure and probability using IBIS ESD models," *IEEE Trans. Device Mater. Rel.*, vol. 12, no. 4, pp. 599–606, Dec. 2012.

[13] J. Tschanz *et al.*, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," *IEEE J. Solid-State Circuits*, vol. 2, no. 11, pp. 344–539, Feb. 2002.

[14] S. Borkar, "Microarchitecture and design challenges for gigascale integration," in *Proc. 37th Int. Symp. Microarchit.*, Dec. 2004, p. 3.

[15] A. Agarwal *et al.*, "Path-based statistical timing analysis considering inter and intra-die correlations," in *Proc. ACM/IEEE TAU*, Dec. 2002, pp. 16–21.

[16] X. Liang and D. Brooks, "Latency adaptation of multiported register files to mitigate the impact of process variations," in *Proc. Workshop ASGI*, 2006.

[17] L. D. Hung, M. Goshima, and S. Sakai, "SEVA: A soft-error-and variation-aware cache architecture," in *Proc. 12th PRDC*, Dec. 2006, pp. 47–54.

[18] V. J. Reddi, D. Z. Pan, S. R. Nassif, and K. A. Bowman, "Robust and resilient designs from the bottom-up: Technology, CAD, circuit, and system issues," in *Proc. 17th ASP-DAC*, Jan. 2012, pp. 7–16.

[19] B. Mohammad, M. Saint-Laurent, P. Bassett, and J. Abraham, "Cache design for low power and high yield," in *Proc. 9th ISQED*, Mar. 2008, pp. 103–107.

[20] B. S. Amrutur and M. A. Horowitz, "Speed and power scaling of SRAM's," *IEEE J. Solid-State Circuits*, vol. 35, no. 2, pp. 175–185, Feb. 2000.

[21] R. C. Baumann, "Soft errors in advanced semiconductor devices-part I: The three radiation sources," *IEEE Trans. Device Mater. Rel.*, vol. 1, no. 1, pp. 17–22, Mar. 2001.

[22] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Proc. 40th Annu. DAC/ACM*, Jun. 2003, pp. 338–342.

[23] (2012). *ITRS* [Online]. Available: http://public.itrs.net

[24] (2014, Feb. 12). *Variability Expedition* [Online]. Available: http://variability.org/research

[25] S. Kirolos, Y. Massoud, and Y. Ismail, "Power-supply-variation-aware timing analysis of synchronous systems," in *Proc. IEEE ISCAS*, May 2008, pp. 2418–2421.

[26] (2011). *ASU Predictive Technology Model (PTM)* [Online]. Available: http://ptm.asu.edu/
[27] K. A. Bowman, S. G. Duvall, and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE J. Solid-State Circuits*, vol. 37, no. 2, pp. 183–190, Feb. 2002.
[28] Solido Design Automation, Inc., *Solido Design Releases Worldwide Variation-Aware Custom IC design Survey Results*. San Jose, CA, USA: Solido Design, Jan. 2010.
[29] P. Gupta *et al.*, "Underdesigned and opportunistic computing in presence of hardware variability," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 8–23, Jan. 2013.
[30] D. Blaauw, S. Pant, R. Chaudhry, and R. Panda, "Design and analysis of power supply networks," in *Electronic Design Automation for Integrated Circuits Handbook*. Boca Raton, FL, USA: CRC Press, 2005.
[31] J. Samandari-Rad, "Design and analysis of robust variability-aware SRAM to predict optimal access-time to achieve yield enhancement in future nano-scaled CMOS," Ph.D. dissertation, Dept. Elect. Eng., Univ. California Santa Cruz, Santa Cruz, CA, USA, Dec. 2012.
[32] D. Wolpert, F. Bo, and P. Ampadu, "Temperature-aware delay borrowing for energy-efficient low-voltage link design," in *Proc. 4th ACM/IEEE Int. Symp. NOCS*, May 2010, pp. 107–114.
[33] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2008.
[34] J. Henkel *et al.*, "Reliable on-chip systems in the nano-era: Lessons learnt and future trends," in *Proc. 50th ACM/EDAC/IEEE DAC*, Jun. 2013, pp. 1–10.
[35] M. White, "Microelectronics reliability: Physics-of-failure based modeling and lifetime evaluation," NASA Electronic Parts and Packaging (NEPP) Program, Office of Safety and Mission Assurance, JPL Publication, Pasadena, CA, USA, Tech. Rep. 08–05, Feb. 2008.
[36] H. Luo, Y. Wang, Y. Cao, Y. Xie, Y. Ma, and H. Yang, "Temporal performance degradation under RTN: Evaluation and mitigation for nanoscale circuits," in *Proc. IEEE Comput. Soc. Annu. Symp. ISVLSI*, Aug. 2012, pp. 183–188.
[37] Z. Guan, M. Marek-Sadowska, and S. Nassif, "SRAM bit-line electromigration mechanism and its prevention scheme," in *Proc. 14th ISQED*, Mar. 2013, pp. 286–293.
[38] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "The case for lifetime reliability-aware microprocessors," in *Proc. 31st Annu. Int. Symp. Comput. Archit.*, Jun. 2004, pp. 276–287.
[39] (2009). *International Technology Roadmap for Semiconductors (ITRS)* [Online]. Available: http://www.itrs.net/
[40] C. Tuck-Boon, R. S. Ghaida, and P. Gupta, "Electrical modeling of lithographic imperfections," in *Proc. 23rd Int. Conf. VLSID*, Jan. 2010, pp. 423–428.
[41] J. B. Bernstein, M. Gurfinkel, X. Li, J. Walters, Y. Shapira, and M. Talmor, "Electronic circuit reliability modeling," *Microelectron. Rel.*, vol. 46, no. 12, pp. 1957–1979, Feb. 2006.
[42] K. Agarwal and S. Nassif, "Statistical analysis of SRAM cell stability," in *Proc. 43rd ACM/IEEE DAC*, San Francisco, CA, USA, Jul. 2006, pp. 57–62.
[43] A. J. Bhavnagarwala, X. Tang, and J. D. Meindl, "The impact of intrinsic device fluctuations on CMOS SRAM cell stability," *IEEE J. Solid-State Circuits*, vol. 36, no. 4, pp. 658–665, Apr. 2001.
[44] B. H. Calhoun *et al.*, "Digital circuit design challenges and opportunities in the era of nanoscale CMOS," *Proc. IEEE*, vol. 96, no. 1, pp. 343–365, Feb. 2008.
[45] S. Wang, G. Leung, A. Pan, C. O. Chui, and P. Gupta, "Evaluation of digital circuit-level variability in inversion-mode and junctionless FinFET technologies," *IEEE Trans. Electron Devices*, vol. 60, no. 7, pp. 2186–2193, Jul. 2013.
[46] A. Dinaburg, "Bitsquatting: DNS hijacking without exploitation," Raytheon Company, Waltham, MA, USA, Tech. Rep. 2011-307, Aug. 2011.
[47] F. J. Kurdahi, A. Eltawil, Y. Kang, S. Cheng, and A. Khajeh, "Low-power multimedia system design by aggressive voltage scaling," *IEEE Trans. Very Large Scale Integration (VLSI) Syst.*, vol. 18, no. 5, pp. 852–856, May 2010.
[48] R. Mastipuram, "Soft errors impact on system reliability," in *Proc. EDN*, Sep. 2004, pp. 69–74.
[49] S. M. Jahinuzzaman, M. Sharifkhani, and M. Sachdev, "An analytical model for soft error critical charge of nanometric SRAMs," *IEEE Trans. Very Large Scale Integration (VLSI) Syst.*, vol. 17, no. 9, pp. 1187–1195, Sep. 2009.
[50] A. Dixit and A. Wood, "The impact of new technology on soft error rates," in *Proc. IEEE IRPS*, Apr. 2011, pp. 5B.4.1–5B.4.7.
[51] N. Dutt, P. Gupta, A. Nicolau, L. A. D. Bathen, and M. Gottscho, "Variability-aware memory management for nanoscale computing," in *Proc. 18th ASP-DAC*, Jan. 2013, pp. 125–132.
[52] K. Lee, A. Shrivastava, L. Issenin, N. Dutt, and N. Venkatasubramanian, "Mitigating soft error failures for multimedia applications by selective data protection," in *Proc. Int. Conf. CASES*, Oct. 2006, pp. 411–420.
[53] J. L. Autran *et al.*, "Altitude and underground real-time SER characterization of CMOS 65 nm SRAM," in *Proc. Eur. Conf. RADECS*, Sep. 2008, pp. 519–524.
[54] D. A. Patterson and J. L. Hennessy, "Large and fast: Exploiting memory hierarchy," in *Computer Organization & Design*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann, 1998, ch. 7, sec. 2, pp. 560–562.
[55] J. B. Velamala, K. Sutaria, T. Sato, and Y. Cao, "Physics matters: Statistical aging prediction under trapping/detrapping," in *Proc. 49th ACM/EDAC/IEEE DAC*, Jun. 2012, pp. 139–144.
[56] H. Puchner and L. Hinh, "NBTI reliability analysis for a 90 nm CMOS technology," in *Proc. 34th ESSDERC*, Sep. 2004, pp. 257–260.
[57] S. Mahapatra, P. Bharath Kumar, and M. A. Alam, "Investigation and modeling of interface and bulk trap generation during negative bias temperature instability of p-MOSFETS," *IEEE Trans. Electron Devices*, vol. 51, no. 9, pp. 1371–1379, Sep. 2004.
[58] B. C. Paul, K. Kunhyuk, H. Kufluoglu, M. A. Alam, and K. Roy, "Impact of NBTI on the temporal performance degradation of digital circuits," *IEEE Electron Device Lett.*, vol. 26, no. 8, pp. 560–562, Aug. 2005.
[59] D. K. Schroder and J. A. Babcock, "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *J. Appl. Phys.*, vol. 94, no. 1, pp. 1–18, Jul. 2003.
[60] M. A. Alam, "A critical examination of the mechanics of dynamic NBTI for PMOSFEts," in *IEEE IEDM Tech. Dig.*, Dec. 2003, pp. 14.4.1–14.4.4.
[61] R. Vattikonda, W. Wenping, and Y. Cao, "Modeling and minimization of pMOS NBTI effect for robust nanometer design," in *Proc. 43rd ACM/IEEE DAC*, Jul. 2006, pp. 1047–1052.
[62] C. Tuck-Boon, J. Sartori, P. Gupta, and R. Kumar, "On the efficacy of NBTI mitigation techniques," in *Proc. DATE Conf. Exhibit.*, Mar. 2011, pp. 1–6.
[63] S. Mahapatra, D. Saha, D. Varghese, and P. B. Kumar, "On the generation and recovery of interface traps in MOSFETs subjected to NBTI, FN, and HCI stress," *IEEE Trans. Electron Devices*, vol. 53, no. 6, pp. 1583–1592, Jul. 2006.
[64] A. T. Krishnan *et al.*, "Material dependence of hydrogen diffusion: Implication for NBTI degradation," in *IEEE IEDM Tech. Dig.*, Dec. 2005, pp. 688–691.
[65] K. Kunhyuk, S. P. Park, K. Roy, and M. A. Alam, "Estimation of statistical variation in temporal NBTI degradation and its impact on lifetime circuit performance," in *Proc. IEEE/ACM ICCAD*, Nov. 2007, pp. 730–734.
[66] S. Borkar, "Electronics beyond nano-scale CMOS," in *Proc. 43rd ACM/IEEE DAC*, Jul. 2006, pp. 807–808.
[67] M. A. Alam and S. Mahapatra, "A comprehensive model of pMOS NBTI degradation," *Microelectron. Rel.*, vol. 45, no. 1, pp. 71–81, 2005.
[68] A. Calimera, E. Macii, and M. Poncino, "NBTI-aware power gating for concurrent leakage and aging optimization," in *Proc. 14th ACM/IEEE ISLPED*, Aug. 2009, pp. 127–132.
[69] C. Xiaoming, Y. Wang, Y. Cao, Y. Ma, and Y. Huazhong, "Variation-aware supply voltage assignment for minimizing circuit degradation and leakage," in *Proc. 14th ACM/IEEE ISLPED*, Aug. 2009, pp. 39–44.
[70] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "Adaptive techniques for overcoming performance degradation due to aging in digital circuits," in *Proc. ASP-DAC*, Jan. 2009, pp. 284–289.
[71] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "Lifetime reliability: Toward an architectural solution," *IEEE Micro*, vol. 25, no. 3, pp. 70–80, May 2005.
[72] A. Tiwari and J. Torrellas, "Facelift: Hiding and slowing down aging in multicores," in *Proc. 41st IEEE/ACM Int. Symp. Microarchit.*, Nov. 2008, pp. 129–140.
[73] W. Wang *et al.*, "The impact of NBTI on the performance of combinational and sequential circuits," in *Proc. 44th ACM/IEEE DAC*, Jun. 2007, pp. 364–369.

[74] Y. Wang *et al.*, "On the efficacy of input vector control to mitigate NBTI effects and leakage power," in *Proc. ISQED*, Mar. 2009, pp. 19–26.
[75] J. Abella, X. Vera, and A. Gonzalez, "Penelope: The NBTI-aware processor," in *Proc. 40th Annu. IEEE/ACM Int. Symp. Microarchit.*, Dec. 2007, pp. 85–96.
[76] U. R. Karpuzcu, B. Greskamp, and J. Torrellas, "The BubbleWrap many-core: Popping cores for sequential acceleration," in *Proc. 42nd Annu. IEEE/ACM Int. Symp. Microarchit.*, Dec. 2009, pp. 447–458.
[77] T. Siddiqua and S. Gurumurthi, "A multi-level approach to reduce the impact of NBTI on processor functional units," in *Proc. 20th Symp. GLSVLSI*, May 2010, pp. 67–72.
[78] R. da Silva and G. Wirth, "Logarithmic behavior of the degradation dynamics of metal-oxide-semiconductor devices," *J. Statist. Mech.*, vol. 2010, no. 6, pp. 1–12, Apr. 2010.
[79] J. B. Velamala, K. Sutaria, H. Shimizu, H. Awano, T. Sato, and Y. Cao, "Statistical aging under dynamic voltage scaling: A logarithmic model approach," in *Proc. IEEE CICC*, Sep. 2012, pp. 1–4.
[80] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "The case for lifetime reliability-aware microprocessors," in *Proc. 31st Annu. Int. Symp. Comput. Archit.*, Jun. 2004, pp. 276–287.
[81] T. Wada, S. Rajan, and S. A. Przybylski, "An analytical access time model for on-chip cache memories," *IEEE J. Solid-State Circuits*, vol. 27, no. 8, pp. 1147–1156, Aug. 1992.
[82] J. Keane and C. H. Kim. (2011, Apr.). Transistor aging. *IEEE Spectrum* [Online]. Available: http://spectrum.ieee.org/semiconductors/processors/transistor-aging
[83] W. A. Tisdale, K. J. Williams, B. A. Timp, D. J. Norris, E. S. Aydil, and X. Y. Zhu, "Hot-electron transfer from semiconductor nanocrystals," *Science*, vol. 328, no. 5985, pp. 1543–1547, Jun. 2010.
[84] S. Bhardwaj, W. Wang, R. Vattikonda, Y. Cao, and S. Vrudhula, "Predictive modeling of the NBTI effect for reliable design," in *Proc. IEEE CICC*, Sep. 2006, pp. 189–192.
[85] T. Grasser *et al.*, "Simultaneous extraction of recoverable and permannent components contributing to bias-temperature instability," in *Proc. IEEE IEDM*, Dec. 2007, pp. 801–804.
[86] V. Huard *et al.*, "New characterization and modeling approach for NBTI degradation from transistor to product level," in *Proc. IEEE IEDM*, Dec. 2007, pp. 797–800.
[87] B. C. Paul, K. Kang, H. Kufluoglu, M. A. Alam, and K. Roy, "Temporal performance degradation under NBTI: Estimation and design for improved reliability of nanoscale circuits," in *Proc. DATE*, Mar. 2006, vol. 1, pp. 1–6.
[88] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "An analytical model for negative bias temperature instability," in *Proc. IEEE/ACM ICCAD*, Nov. 2006, pp. 493–496.
[89] K. Jeong, A. B. Kahng, and K. Samadi, "Impact of guardband reduction on design outcomes: A quantitative approach," *IEEE Trans. Semicond. Manuf.*, vol. 22, no. 4, pp. 552–565, Nov. 2009.
[90] V. J. Reddi, M. S. Gupta, M. D. Smith, W. Gu-Yeon, D. Brooks, and S. Campanoni, "Software-assisted hardware reliability: Abstracting circuit-level challenges to the software stack," in *Proc. 46th ACM/IEEE DAC*, Jul. 2009, pp. 788–793.
[91] J. Choi, C. Chen-Yong, H. Franke, H. Hamann, A. Weger, and P. Bose, "Thermal-aware task scheduling at the system software level," in *Proc. ACM/IEEE ISLPED*, Aug. 2007, pp. 213–218.
[92] K. Kyung-Hoae, "Comparison study of future on-chip interconnects for high performance VLSI applications," Ph.D. dissertation, Dept. Elect. Eng., Stanford Univ., Stanford, CA, USA, Mar. 2011.
[93] W. Wang, V. Reddy, B. Yang, V. Balakrishnan, S. Krishnan, and Y. Cao, "Statistical prediction of circuit aging under process variations," in *Proc. IEEE CICC*, Sep. 2008, pp. 13–16.
[94] L. Gerrer *et al.*, "Interplay between statistical reliability and variability: A comprehensive transistor-to-circuit simulation technology," in *Proc. IEEE IRPS*, Apr. 2013, pp. 3A.2.1–3A.2.5.
[95] A. Kumari and S. Kumar, "Analysis of nanoscale strained-Si/SiGe MOSFETs including source/drain series resistance through a multi-iterative technique," in *Proc. 27th Int. Very Large Scale Integration (VLSI) Design, 13th Int. Conf. Embedded Syst.*, Jan. 2014, pp. 427–432.

JEREN SAMANDARI-RAD received the B.S. degree in electrical engineering and computer science from the University of California at Berkeley, Berkeley, CA, USA, in 2000, the M.S. degree in electrical engineering from California Polytechnic State University, San Luis Obispo, CA, USA, in 2005, and the Ph.D. degree in electrical engineering from the University of California at Santa Cruz (UCSC), Santa Cruz, CA, USA, in 2012.

He is currently a Post-Doctoral Researcher and Staff Member at UCSC. His most recent work titled *VAR-TX: A Variability-Aware SRAM Model for Predicting the Optimum Architecture to Achieve Minimum Access-Time for Yield Enhancement in Nanoscaled CMOS*, co-authored by Prof. Hughey and Prof. Guthaus, was published in the International Symposium on Quality Electronic Design in 2012. Prior to starting his Ph.D. program, he was a Hardware Design Engineer with the Department of Research and Development, Real Time Solutions, Inc., Napa, CA, USA. He was also the CEO of his own Ragbar Electric Company for five years prior to 1995. His research interests are in high-speed low-power analog and digital circuits, and static random access memory.

Dr. Samandari-Rad was a recipient of a patent for the instant electrical water heater, the Certificate of Appreciation from Real Time Solutions, Inc., and several academic awards from the colleges he has attended in the USA.

MATTHEW GUTHAUS (M'06–SM'10) received the B.S.E. degree in computer engineering, the M.S.E. degree, and the Ph.D. degree in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, in 1998, 2000, and 2006, respectively. He is currently an Associate Professor with the Department of Computer Engineering, University of California at Santa Cruz (UCSC), Santa Cruz, CA, USA. He is the Director of the UCSC Summer Undergraduate Research Fellowship in IT, a national science foundation sponsored research experience for undergraduates site. His research interests are in low-power computing, including applications in mobile health systems such as new circuits, architectures, and sensors along with their application to mobile and clinical health systems. He is also a Senior Member of ACM and a member of IFIP Working Group 10.5. He was a recipient of the 2011 NSF CAREER Award and the 2010 ACM SIGDA Distinguished Service Award.

RICHARD HUGHEY is the Vice Provost and the Dean of Undergraduate Education with the University of California at Santa Cruz (UCSC), Santa Cruz, CA, USA, where he is also a Professor of Computer Engineering and of Biomolecular Engineering. Previously, he served as the Co-Director of the Advanced Studies Laboratories, a partnership laboratory jointly run by UCSC and the NASA Ames Research Center. He is the co-creator of the SAM sequence analysis package. He has served as the Chair of Computer Engineering and the Vice Chair of Biomolecular Engineering. He has led and co-led the development of a multitude of programs, including undergraduate programs in robotics engineering, bioengineering, network and digital technology, and bioinformatics, graduate programs or concentrations in bioinformatics, biomedical science and engineering, and robotics and control. His research interests include bioinformatics, hidden Markov models, computer architecture, and parallel processing. He was a recipient of the UCSC Chancellor's Achievement Award for diversity in recognition of his work with the Baskin School of Engineering in 2008. He received the B.A. degree in mathematics and the B.S. degree in engineering from Swarthmore College, Swarthmore, PA, USA, and the M.Sc. and Ph.D. degrees in computer science from Brown University, Providence, RI, USA.