A Resilience Roadmap

(Invited Paper)

Sani R. Nassif
Austin Research Laboratory
IBM Corporation
Austin, TX 78758
nassif@us.ibm.com

Nikil Mehta
Department of Computer Science
California Institute of Technology
Pasadena, CA 91125
nikil@caltech.edu

Yu Cao
Department of Electrical Engineering
Arizona State University
Tempe, AZ 85287
yu.cao@asu.edu

***Abstract*—Technology scaling has an increasing impact on the resilience of CMOS circuits. This outcome is the result of (a) increasing sensitivity to various intrinsic and extrinsic noise sources as circuits shrink, and (b) a corresponding increase in parametric variability causing behavior similar to what would be expected with hard (topological) faults. This paper examines the issue of circuit resilience, then proposes and demonstrates a roadmap for evaluating fault rates starting at the 45nm and going down to the 12nm nodes. The complete infrastructure necessary to make these predictions is placed in the open source domain, with the hope that it will invigorate research in this area.**

I. Introduction

It is a well-known fact that integrated circuits can fail. Such failure, broadly, can result from: (a) manufacturing defects which change the topology of the circuit, i.e. via shorts or opens; (b) signal corruption resulting from internal or external noise which causes the circuit to misbehave; or (c) failure of the circuit to meet its specification in terms of frequency of operation, power, or similar metric. In this context, we say that a circuit is more resilient if it is able to tolerate increased levels of faults and noise.

Focusing on digital synchronous CMOS circuits, we can narrow the resilience issue to a more manageable discussion, since measures of the performance and correctness of such circuits are readily available and highly standardized. Consider a metric like logic functionality: we can clearly state that the logical output of a CMOS inverter must be the logic inverse of its input, or that a latch must maintain its output value until the next clock pulse. More complex metrics like timing or power are somewhat harder to express simply, but there is an abundance of checking tools that do an excellent job of determining correctness and identifying specific failures.

Traditionally, the difference between the so called "hard" and "soft" failures has been based on whether the fault causes a topological change in the circuit, which typically causes incorrect behavior under all conditions, versus merely changing the electrical parameters of various devices and thus causing faulty behavior. For example, a short or an open can cause a CMOS inverter to have its output "stuck at" a logic 1 or a 0, while a shift in a parameter such as gate oxide thickness has the more subtle effect of changing the time necessary for the inverter to change its state, i.e. the inverter delay.

However, under certain conditions excessive parametric variability can cause circuit behavior consistent with a permanent or hard fault. The canonical example of this case in present day (e.g. 65nm CMOS) technologies is static random access memory (SRAM), where the need for density leads to using the smallest devices possible. These small devices are especially susceptible to various scaling-related sources of variability like Random Dopant Fluctuations (RDF) and Line Edge Roughness (LER). For SRAM, excessive device variability can lead to scenarios where a particular SRAM bit cannot be read or written, or where reading one bit causes a neighboring bit to change value. This behavior has been recognized for some time now, and many researchers have examined the causes, developed detailed statistical analysis methodologies, and proposed solutions. In fact, the sophistication of current schemes for dealing with SRAM faults, like redundancy, parity checking, and error correction, are a testimony to the recognition of this problem.

In addition to the parametric variability mentioned above, we note that as circuits get smaller they become more sensitive to various forms of noise. As the so-called "critical charge" being held in a memory element reduces, the potential for an errant noise source impacting that charge increases.

Our goal in this paper is to show that continued technology scaling will cause the type of behavior currently observed in SRAM to become much more pervasive. This observation stands to reason since as technologies scale and devices shrink, what once was a small SRAM-sized transistor for one node (e.g. 65nm) will be quite similar in size to a nominally sized transistor two nodes later (e.g. 32nm). To demonstrate these trends in concrete terms, we will develop a resilience roadmap which focuses on predicting circuit resilience for future technologies. This roadmap will focus on the manner with which technology scaling impacts circuit resilience, holding constant the current circuit implementation styles and topologies. It is understood and expected that as these resilience problems become important, innovations at the device, circuit, and architecture levels will be occur. Our goal in this work is to develop a rational methodology for predicting when these types of innovations are going to be necessary.

The outline of the paper is as follows. We first detail our basic modeling assumptions in dealing with future technologies, their variability, and anticipated sources of noise. Next we explain our methodology in creating scaled versions of three canonical circuits: a CMOS inverter, latch, and SRAM

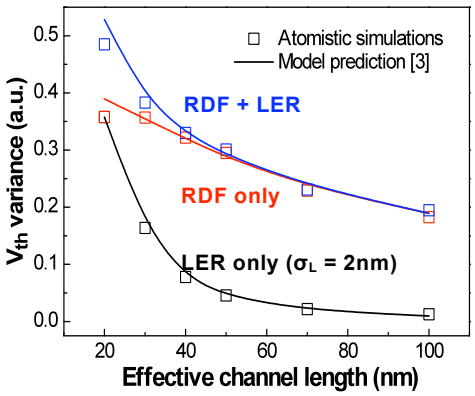

Fig. 1. LER induced variation becomes more pronounced as gate length scales below 20nm.

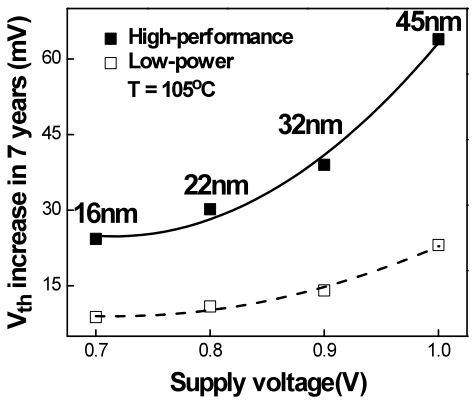

Fig. 2. The prediction of V_{th} increase in 7 years due to NBTI.

bit. We then show a simple statistical analysis methodology to measure the fail rates of these circuits and provide results for our three canonical circuits. Finally, we close with some ideas for possible applications of this roadmap, and conclude by providing the web location where all the associated models, scripts and documentation for this open roadmap can be found.

II. Predictive Technology Modeling

The prediction of future device characteristics is based on the Predictive Technology Model (PTM), from the 45nm node to the 12nm node [1]. PTM uses a set of physical equations that capture the essential behavior of charge and carrier transport, rather than the full set of BSIM models [2]. The electrostatic models emphasize the dependence of the threshold voltage (V_{th}) on physical aspects of the device like channel length (e.g. drain-induced barrier lowering -DIBL), channel doping, HALO implant, etc. The transport part of the model adopts the velocity saturation model with overshoot behavior [2]. Moreover, the impact of layout dependent stress effects is embedded into predictive models of the mobility and threshold voltage.

In addition to nominal PTM models, this work further integrates a suite of predictive models of random, systematic, and temporal variations. Variability and reliability effects usually manifest themselves as parameter fluctuations in a CMOS transistor, such as the channel length, gate oxide thickness, the threshold voltage. These fluctuations may be static, i.e. occurring during fabrication, or dynamic (e.g., aging effect), and they can be spatially or temporally distributed. The exact amount of the fluctuation further depends on layout and operational conditions. A generic modeling solution to abstract these effects into circuit simulation is to identify the key device parameters under the influence of variability, and build variational models for them on top of the existing standard device model. Such a model of handling variability, i.e. as an external module on top of a standard device model, affords great flexibility in terms of model customization; and offers a convenient approach to integration with nominal device model and circuit simulation tools.

As an example, the effects of random dopant fluctuations (RDF) and line-edge roughness (LER) can be captured as variations in the threshold voltage [3]. Both RDF and LER are intrinsic to the CMOS structure, and they are known to be two of the most important phenomena imposing the ultimate limits on technology scaling. They both stem from atomistic-level fluctuations, and are truly random in nature. As the device size scales down, the amount of this randomness is rapidly escalating. Based on the underlying physics and atomistic simulations, compact models of V_{th} variation under RDF and LER are developed in order to predict their impact on future circuit performance [3]. As shown in Fig. 1 [3], the effect of LER on V_{th} variation may be comparable to that by RDF starting at around the 22nm node, severely affecting the leakage and SRAM cell stability. Other fundamental variations covered in this modeling set include carrier mobility and the stress effect.

Besides static variations due to the fabrication process, this work incorporates the temporal degradation of CMOS devices, especially the effect of negative-bias-temperature-instability (NBTI). NBTI occurs in the P-Channel device and causes an increase in the magnitude of V_{th}. As the gate oxide becomes thinner than 4nm, NBTI-induced shift in V_{th} has become the dominant factor limiting device lifetime and circuit reliability [4]. The exact amount of V_{th} shift is a strong function of both device parameters and circuit operation conditions such as the switching activity. This work adopts NBTI models in [4], [5] to project the increase of V_{th} in typical high-performance and low-power applications. Figure 2 presents that prediction; these values are integrated into nominal PTM model files to assess the impact on circuit reliability.

In addition to intrinsic parameter variations, it is possible to also model the impact of extrinsic noise sources, such as soft errors due to particle strikes [6]. Due to space constraints, we will only show the general methodology for accomplishing this task, and defer a more complete treatment to a future publication. There are an abundance of extrinsic noise particles that can cause circuits to fail, depending on their environment: alpha particles, neutrons, heavy-ions, solar event protons, etc. So in the context of this roadmap, one can propose a

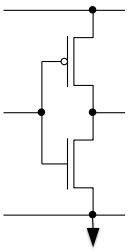

Fig. 3. A CMOS inverter.

framework for predicting the impact of such noise on circuit failure by modeling a *generic* particle strike as a triangular-shaped current pulse injected into a sensitive circuit node. The pulse height and width can be calibrated to that of a pulse induced by realistic high-energy particle [7], [8].

Temporal noise sources other than particle strikes, such as V_{dd} variations or intrinsic effects like shot, thermal, and random telegraph noise are not considered in this iteration of the roadmap and are left to future work and contributions from the greater research community.

Overall, predictive models of variability and reliability in this work emphasize intrinsic variations and temporal effects, since they are fundamental to the CMOS structure and have a far-reaching impact on future IC design, especially for devices with minimum feature sizes. By abstracting these effects into appropriate device parameters, it provides a solid and flexible basis for benchmarking the reliability of various circuits.

III. Basic Circuit Scaling

Our purpose in creating this roadmap is to show trends in failure rate as technology moves forward. In order to illustrate this in as unbiased a manner as possible, it is necessary to show the impact of technology scaling independent of any other changes or innovations that might occur to the basic circuits on which the roadmap is based. Thus it is necessary to develop a methodology for scaling these circuits across the various technologies in a realistic manner.

Consider the example of a simple CMOS inverter, made up of one N-Channel and one P-Channel MOSFET, and shown in Figure 3. We make the following assumptions:

- The length of the N and P-Channel devices are the same (see Table I).
- The supply voltage (V_{dd}) is set to whatever the technology roadmap recommends as the appropriate voltage for that technology (see Table I).
- The length is fixed to be the nominal channel length for the technology in question.
- The width of the N-Channel device is arbitrarily fixed at $8\times$ the length. This is a reasonable assumption for a typical mid-performance logic cell family.
- The ratio of the P-Channel to N-Channel widths is such that the rise and fall times of this inverter, when composed in a long chain of inverters, are equal.

The first part of our scaling methodology is to determine how the P to N-Channel width ratio scales. We do this by

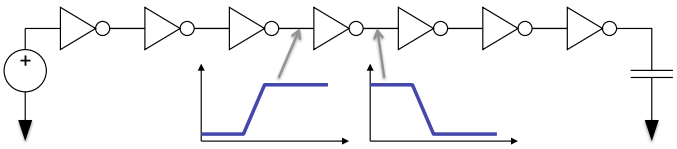

Fig. 4. A chain of CMOS inverters used to estimate the optimal P to N-Channel width ratio.

TABLE I
Characteristics and scaling results for technologies considered in this roadmap.

Technology	L (nm)	V_{dd} (V)	P to N ratio	Pair Delay (ps)
45nm	45	1.0	1.02	18.94
32nm	32	0.9	0.96	15.96
22nm	22	0.8	0.91	13.40
16nm	16	0.7	0.81	10.36
12nm	12	0.65	0.84	9.49

composing the circuit shown in Figure 4 made up of a chain of seven CMOS inverters in series, and we examine the waveforms of the two inverters in the middle of the chain. We use the middle of the chain to minimize the effect of the input waveform and the terminating load on the waveforms in question. We measure the 20%–80% transition times for the two waveforms, one of which is rising and one falling. We then perform a binary search to find the value of P to N-Channel widths which results in the rise and fall times being equal, terminating when we have determined that ratio to within +/– 0.01. The number of iterations required to find the optimum ratio is typically about 10. For the technologies under consideration, the resulting ratios are shown in Table I.

Once the P to N-Channel width ratio is determined, the second challenge is scaling time (or frequency). This is needed for two distinct reasons:

1) The correct performance of any complex digital circuit such as a latch or an SRAM is determined in the context of that circuit producing the expected output within an appropriate time window. We need a way in which that time window scales consistently.
2) To make sure the circuits are operating in a realistic range, we often need to provide the appropriate output termination, i.e. output load. Such a load is often represented as a lumped capacitor, and the value of that capacitor would be expected to scale in a manner similar to time scaling.

We choose to determine this time scale by using the so-called "pair delay" of a fanout-of-four inverter. This is defined as the total delay (measured at 50% of the supply voltage) of a series pair of inverters, each of which has a total load equal to four copies of itself. To do this we simply create the circuit shown in Figure 5 and measure the pair delay directly. The results are shown in Table I.

Now that we have a methodology for model scaling and for circuit scaling, we can perform our roadmap study in the next sections.

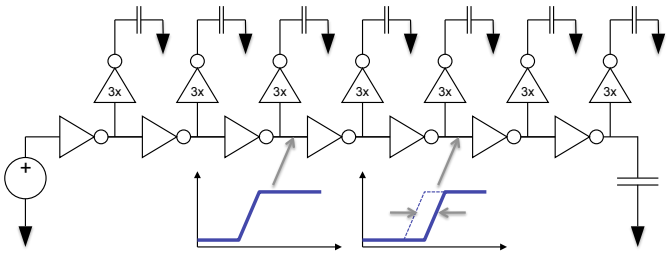

Fig. 5. An FO4 chain of CMOS inverters used to estimate the pair delay.

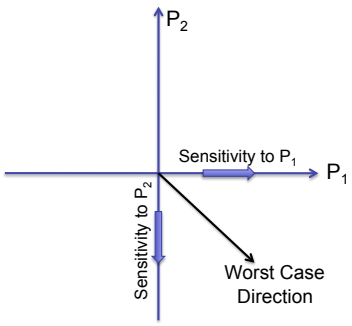

Fig. 6. Illustrating the determination of the worst case direction based on individual parameter sensitivities.

IV. Failure Study Methodology

We want to estimate the probability of failure of each of the three circuits selected for this study, a CMOS inverter, a latch, and an SRAM bit cell. First, we examine failures caused by manufacturing variability, represented by the fluctuations in parameter values of the underlying MOSFET devices. This is equivalent to asking: *if I manufactured a large number of copies of this circuit, what proportion would not be functional?* Stated in this manner, this probability of failure is directly related to the *yield* of the individual circuit of interest. In later sections we show how we also estimate the impact of NBTI and extrinsic noise on this failure probability.

In order to estimate failure probability we need a clear definition of what constitutes failure. Each circuit has multiple failure modes, so the definition of failure even for a single circuit can be quite complex. We choose to take a pragmatic approach to this problem, and to define a single failure metric for each of the three circuits of interest. Given our desire to make this roadmap and its underlying implementation available to others for enhancement and further study, we fully expect other researchers will enhance our simple failure model to produce a more complete and general one.

As a concrete example of this pragmatic approach we start with our simplest circuit, the CMOS inverter, and define failure as the point at which the inverter can no longer produce a zero on its output, i.e. the point at which the inverter appears to be *stuck at 1*. Conceptually, this happens if the P-Channel device has very high leakage, and if the N-Channel device is so weak that sinking the P-Channel leakage current causes its drain-to-source voltage to be larger than half the supply voltage. A more natural way of defining this fault is as the condition under which the falling delay of the CMOS inverter becomes infinite.

Now that we have a circuit and a particular failure mode, we need to define what we call the *worst case direction*. To do so, we first need to define the input parameter space of interest, i.e. the sources of manufacturing variability that we will include in our analysis. For the CMOS inverter example, we select the following:

- Channel length deviation, which is assumed to follow a zero-mean normal distribution, and is used for all (both P and N-Channel) devices in the circuit.
- P-Channel threshold voltage deviation, which is assumed to follow a zero-mean normal distribution.
- N-Channel threshold voltage deviations, same as above.
- P-Channel mobility deviation, which is assumed to follow a zero-mean normal distribution.
- N-Channel mobility deviation, same as above.

Thus the input parameter space is of dimension 5 in this case. Since all the distributions are normal and have zero mean and are uncorrelated with each other, we can further simplify our representation of this parameter space by normalizing each parameter by its standard deviation (a practice called *standardization*). In this space the origin is the *nominal* point, and the hypersphere of unit radius one is one standard deviation away from the mean.

In this input space, the worst case direction is that direction along which the performance metric of interest degrades. We can estimate this worst case direction by performing a simple perturbation study, changing one parameter at a time, and determining the sensitivity of the performance to each parameter. In our implementation, we drastically simplify this problem by only using the *sign* of the sensitivity, rather than its value, and by assuming that the sensitivity is constant throughout the input space. Far more complex methodologies have been proposed by researchers over the years, a recent example is [9] and a more complete treatment can be found in [10]; we are confident that such researchers will enhance the accuracy of this roadmap by implementing more sophisticated methods for finding the failure probability. A simple two-dimensional example of this worst case direction determination is shown in Figure 6.

Once the worst case direction is set, we seek the point along that direction where the circuit first fails. We do this by performing a simple binary line-search along that direction, stopping when we have determined the distance to within 0.02 sigma (recall that the space is normalized so that all of our statistical values have zero mean and unit standard deviation). Denoting this distance by D_f, we define the probability of success ζ as being inside the hypersphere of radius D_f. We can compute ζ in N dimensional space as follows:

$$\zeta = (2\Phi(x) - 1)^N \tag{1}$$

Where Φ is the standard cumulative distribution function for

TABLE II
WORST CASE DIRECTION FOR A SIMPLE CMOS INVERTER.

L	PV_{th}	NV_{th}	P μ	N μ
+	-	+	+	-

TABLE III
ROADMAP PREDICTIONS FOR A SIMPLE CMOS INVERTER.

Technology	Distance D_f	Probability ζ
45nm	–	≈ 0
32nm	–	≈ 0
22nm	–	≈ 0
16nm	16.1	2.4e-58
12nm	13.1	1.2e-39

a normal distribution:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt = \frac{1}{2} \left[1 + erf \left(\frac{x}{\sqrt{2}} \right) \right] \quad (2)$$

Table II shows the worst case direction for the CMOS inverter and Table III shows the distance and failure probability as a function of technology. Note that for current technologies the distance from the origin is so far that the probability is essentially zero. A comforting fact since we expect a simple circuit like a CMOS inverter to be extraordinarily robust. A less comforting conclusion is the trend obvious in the results, which we will come back to in our conclusions.

A. CMOS Latch

We perform a similar failure analysis as described in the previous section for a conventional D type CMOS latch (Figure 7) taken from [11] (where it is referred to as being the DSTC latch). While the CMOS inverter has a single failure mode due to variability (failure to switch), a latch can fail in two different ways assuming stable inputs:

- Write Latency: Failure to propagate the $D \rightarrow Q$ value within a clock cycle.
- Hold: Failure to maintain the output Q value within a clock cycle.

For simplicity we only consider the write latency failure mode. To determine the time required for a write for a given technology, we calculate the minimal clock pulse width needed for correct operation at 45nm and scale that timing down using the pair delay ratio of 45nm to our target technology (Table I). As in the inverter case, we apply parameter variation in a direction such that it increases the delay of the $D \rightarrow Q$ write until such point that the write fails. Table IV shows the failure probability of the latch due to variation.

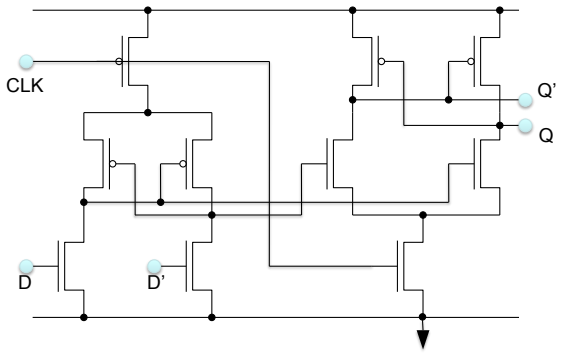

Fig. 7. Circuit diagram for latch used in roadmap.

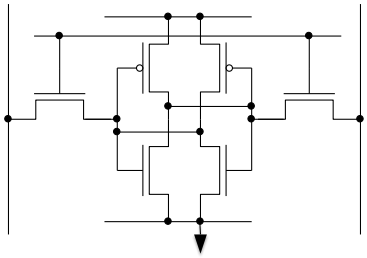

Fig. 8. Circuit diagram for standard 6T SRAM bit cell used in roadmap.

TABLE IV
ROADMAP PREDICTIONS FOR ALL THREE CIRCUITS.

Technology	Inverter ζ	Latch ζ	SRAM ζ
45nm	≈ 0	≈ 0	6.1e-13
32nm	≈ 0	1.8e-44	7.3e-09
22nm	≈ 0	5.5e-18	1.5e-06
16nm	2.4e-58	5.4e-10	5.5e-05
12nm	1.2e-39	3.6e-07	2.6e-04

B. CMOS SRAM

Our third and final circuit in this roadmap is one that is well known to already have high failure rates in current technologies, namely the SRAM six transistor bit cell shown in 8. Like the latch above, an SRAM can fail in many different ways:

- Readability: where an SRAM cell cannot be read within a specified cycle time.
- Writability: where an SRAM cell cannot be written within a specified cycle time.
- Stability: where reading an SRAM cell causes its neighbors to be disturbed sufficiently to change their content.

For simplicity we only consider the writability failure mode. To test the SRAM bit, we initialize it with a one, set the bit lines such that they will attempt to write a zero into the cell, pulse the word line, and measure the delay before the internal cell node takes on the new value. Like before, the timing is scaled using the pair delay ratio from Table I.

Because SRAM devices are small and therefore exhibit large threshold variations caused by random dopant fluctuations, we use a somewhat more complex parameter space for the SRAM. In addition to the length and mobility of the P and N-Channel devices, we consider *each* of the threshold voltages of the six transistors within the SRAM as separate and uncorrelated variables. This leads to a parameter space of dimension 9, but the same type of analysis is performed as before, i.e. finding the worst case direction, and then performing a search to find the failure point. The results are shown in Table IV.

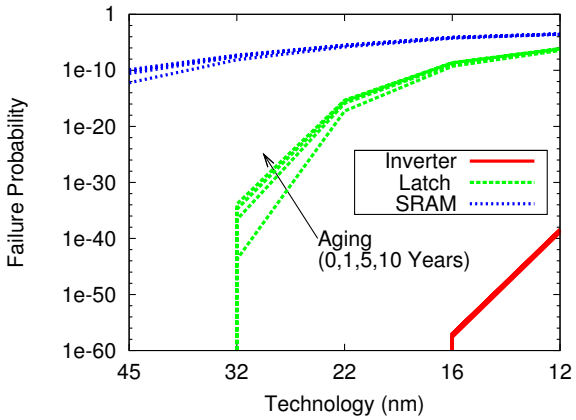

Fig. 9. Impact of NBTI on failure probability trends.

C. Aging

Figure 9 plots the impact of NBTI on failure probability for each circuit across technologies. Failure trends are plotted for manufacturing time failures and for 1, 5, and 10 year lifetime estimates. V_{th} shift due to NBTI over a desired time span is estimated assuming nominal V_{dd}, temperature, and 50% duty cycle (active operation and not sleep mode) using techniques from [12].

V. Conclusions and Future Work

Our attempt in assembling this roadmap is to sensitize our research community to the types of resilience problems that are expected in the near future as technology scaling continues. To this end, we intend to make the models, methodologies, analysis scripts, and results freely available for other researchers to examine, improve, correct, and extend. The authors do not claim to have produced the last word on this subject, but rather the first. Throughout the paper we pointed out places where, for the sake of simplicity and expediency, we consciously made approximations and took short-cuts in order to get this initial roadmap started. We indeed look forward to seeing others take up this challenge, and to continued lively dialogue in this area.

Some specific study directions that we recognize would be of interest:

- Impact of power supply, which is already known to be highly detrimental to SRAM. Many of the resilience issues raised in this paper are often dealt with by increasing power consumption, but such a solution is not sustainable.
- Impact of operating temperature, especially in connection with aging mechanisms.
- Impact of all types of extrinsic noise particles.
- Extending the roadmap to deal with intrinsic noise sources (i.e. noise generated within the circuit and devices themselves).
- Comparative study of different types of circuits, e.g. taking a family of latch implementations and comparing their resilience (somewhat like what was done in [11] for performance).

All the material associated with this roadmap can be found at the web site for the predictive technology models: http://ptm.asu.edu/

Acknowledgment

The authors want to gratefully acknowledge the contributions of Juan-Antonio Carballo, Chris Wilkerson, and Larry Wissel to the development of this roadmap; and thank Andre DeHon, Heather Quinn and Helmut Graeb for comments on this paper.

This material is based upon work supported by the National Science Foundation under Grant No. 0637190 to the Computing Research Association. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Computing Research Association or the National Science Foundation.

References

[1] "Predictive technology model (ptm)," available at http://www.eas.asu.edu/p̃tm.

[2] W. Zhao and Y. Cao, "New generation of predictive technology modeling for sub-45nm early design exploration," *IEEE Trans. Electron Devices*, vol. 53, no. 11, pp. 2816–2823, Nov 2006.

[3] Y. Ye, F. Liu, S. Nassif, and Y. Cao, "Statistical modeling and simulation of threshold variation under dopant fluctuations and line-edge roughness," in *Design Automation Conference*, 2008, pp. 900–905.

[4] W. Wang, V. Reddy, A. T. Krishnan, R. Vattikonda, S. Krishnan, and Y. Cao, "Compact modeling and simulation of circuit reliability for 65nm cmos technology," *IEEE Trans. on Device and Materials Reliability*, vol. 7, no. 4, pp. 509–517, Dec 2007.

[5] R. Zheng, J. Velamala, V. Reddy, V. Balakrishnan, E. Mintarno, S. Mitra, S. Krishnan, and Y. Cao, "Circuit aging prediction for low-power operation," in *Custom Integrated Circuits Conference*, 2009, pp. 427–430.

[6] A. KleinOsowski, E. H. Cannon, P. Oldiges, , and L. Wissel, "Circuit design and modeling for soft errors," *IBM Journal of Research and Technology*, vol. 52, no. 3, pp. 255–263, May 2008.

[7] P. Shivakumar, M. Kistler, S. W. Keckler, D. Burger, and L. Alvisi, "Modeling the effect of technology trends on the soft error rate of combinational logic," in *International Conference on Dependable Systems and Networks*, 2002, pp. 389–398.

[8] M. Zhang and N. R. Shanbhag, "A soft error rate analysis (sera) methodology," in *International Conference on Computer-Aided Design*, 2004, pp. 111–118.

[9] H. Zhang, T. Chen, M. Ting, and X. Li, "Efficient design-specific worst-case corner extraction for integrated circuits," in *Design Automation Conference*, 2009, pp. 386–389.

[10] H. Graeb, *Analog Design Centering and Sizing*. Springer Verlag, 2007.

[11] V. Stojanovic and V. G. Oklobdzija, "Comparative analysis of master-slave latches and flip-flops for high-performance and low-power systems," *IEEE Journal of Solid State Circuits*, vol. 34, no. 4, pp. 536–548, Apr 1999.

[12] R. Vattikonda, W. Wang, and Y. Cao, "Modeling and minimization of pmos nbti effect for robust nanometer design," in *Design Automation Conference*, 2006, pp. 1047–1052.