Technique for Efficient Evaluation of SRAM Timing Failure

Masood Qazi, Mehul Tikekar, Lara Dolecek, Devavrat Shah, and Anantha P. Chandrakasan

***Abstract*—This brief presents a technique to evaluate the timing variation of static random access memory (SRAM). Specifically, a method called loop flattening, which reduces the evaluation of the timing statistics in the complex highly structured circuit to that of a single chain of component circuits, is justified. Then, to very quickly evaluate the timing delay of a single chain, a statistical method based on importance sampling augmented with targeted high-dimensional spherical sampling can be employed. The overall methodology has shown 650× or greater speedup over the nominal Monte Carlo approach with 10.5% accuracy in probability. Examples based on both the large-signal and small-signal SRAM read path are discussed, and a detailed comparison with state-of-the-art accelerated statistical simulation techniques is given.**

***Index Terms*—Cache memories, CMOS memory, process variation, random access memory, sense amplifier, static random access memory (SRAM).**

I. Introduction

Embedded static random access memory (SRAM) is a vital component of digital integrated circuits and often constitutes a dominant portion of the chip area [1]. Therefore, the specifications of embedded SRAM have significant implications on the overall chip cost, power, performance, and yield. Shown in Fig. 1(a) is a plot of reported cell areas in fully functional SRAM macros versus the technology node for the past few years. The cell area has scaled with the scaling of the critical feature size. Fig. 1(b) plots an unconventional metric—the number of SRAM bits per square millimeter of silicon in high-performance microprocessor chips—which reveals that reduced SRAM cell area does not readily translate into increased SRAM utilization.

This discrepancy in trends is due to a number of limitations of SRAM, all related to local variation: SRAM often needs a separate elevated power supply; excessive SRAM timing variation degrades performance; and uncertain aging effects show up first in the highly integrated and highly sensitive SRAM cells. As the overarching goal of this brief, we seek to increase the SRAM utilization by propagating the physical trend of shrinking cell area into the overall system-on-chip improvement. This goal can be achieved if designers have a way to quickly assess the impact of circuit solutions on the operating constraints (e.g., minimum VDD, frequency) to ultimately preserve the overall chip yield.

This brief focuses on read access yield because it has been observed in measurements that ac failures, manifested as too slow an access time from one or more addresses, are encountered before dc failures, manifested as the corruption of data at one or more addresses [2]. Therefore, dc stability (write and read margin) is necessary but not sufficient for yielding a memory chip. A significant degree of additional margin must be inserted to meet the performance requirements.

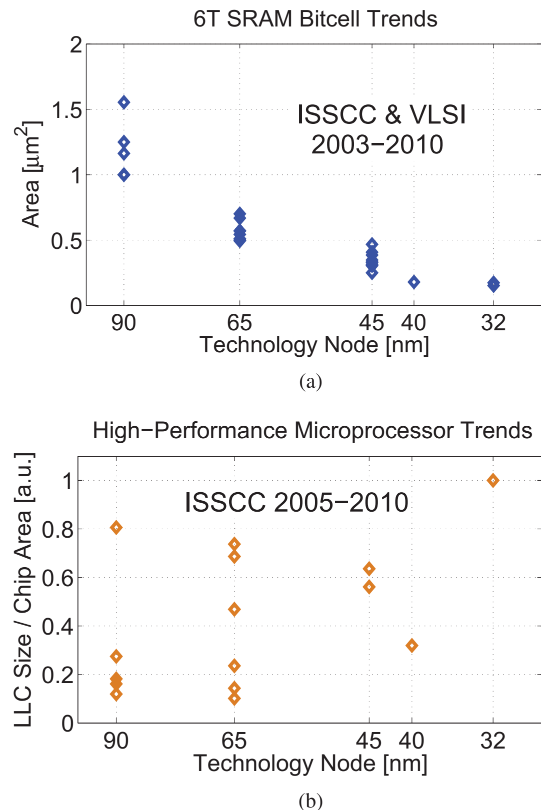

Fig. 1. (a) SRAM cell area scaling. (b) Utilization of SRAM in recent high-performance microprocessors.

In general, the exact distributions of the relevant SRAM performance metrics are not known. As a consequence, any statistical simulation method unavoidably resorts to numerical solvers such as SPICE. Classical approaches such as the Monte Carlo method require too many iterations of such SPICE evaluations because of the circuit complexity and extremely low tolerable failure probabilities of individual components (10^{-8} and below). Thus, the primary challenges to any statistical simulation method are: 1) dealing with the structural complexity of the timing delay evaluation problem and 2) estimating timing delay statistics to a very high accuracy.

A. Prior Work

A lot of exciting recent work has made important progress toward the eventual goal of designing generically applicable efficient simulation methodologies for circuit performance evaluation. To begin with, in [3]–[7], the authors developed efficient sampling-based approaches that provide significant speedup over the Monte Carlo method. However, these works do not deal with the interconnection complexity, i.e., do not address challenge 1) stated in the previous section.

Other authors have addressed the issue of structural complexity. In [8], by modeling the bitline signal and the sense amplifier offset (and the timer circuit) with Gaussian distributions, the authors proposed a linearized model for the read path. As this model can be simulated in MATLAB, the SRAM structure can be emulated and the evaluation time can be significantly improved. Additional approaches such as [9] and [10] apply more sophisticated techniques involving Gumbel

Manuscript received November 20, 2011; revised April 12, 2012; accepted June 3, 2012. Date of publication September 10, 2012; date of current version July 22, 2013. This work was supported by the C2S2 Focus Center, one of six research centers funded under the Focus Center Research Program, a Semiconductor Research Corporation Entity.

M. Qazi, M. Tikekar, D. Shah, and A. P. Chandrakasan are with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: mqazi@mit.edu; mtikekar@mit.edu; devavrat@mit.edu; anantha@mtl.mit.edu).

L. Dolecek is with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095-1592 USA (e-mail: dolecek@ee.ucla.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2012.2212254

1063-8210/$31.00 © 2012 IEEE

distributions and sensitivity analysis but still do not incorporate a full-scale SPICE functionality check to directly evaluate extreme delay statistics, which is generally necessary to handle all possible operating scenarios (e.g., low-voltage operation).

B. Contributions

In this brief, we show how to overcome the two challenges for the timing delay analysis of SRAM by means of two proposed methods of loop flattening and spherical importance sampling (IS), respectively. These techniques were introduced in [11], and in this brief, we add: 1) a theoretical justification of loop flattening; 2) new evidence of the loop-flattening accuracy in the large signal SRAM read path under general conditions of non-Gaussian delays, multiple levels of nested sampling, and correlated fluctuations; 3) a detailed breakdown of the simulation cost of spherical IS; and 4) a quantitative comparison with other works regarding simulation cost versus failure probability level and dimensionality.

II. Loop Flattening for Timing Variation

In this section, we describe the problem of statistically analyzing the SRAM read path which contains circuit blocks repeated at different rates. Then we introduce and justify the loop-flattening approximation to enable the application of accelerated statistical simulation techniques. In the representative block diagram of an SRAM array of Fig. 2(a), there are highly repeated structures: memory cells, sense amplifiers, row decoders and drivers, and timing circuits. There are several distinct cascaded circuit stages, some of which may be correlated. The circuit is also big. A straightforward way to simulate this circuit is to take a complete schematic and address each location in simulation while noting the behavior for each address location. This method would cost too much computational resources, so a circuit designer turns to a simulation of a critical path by taking a representative member of each group of circuits and adding appropriate parasitic loading in parallel.

A statistical analysis of a memory-critical path requires additional insight into the architecture. For now, consider a single column of 256 memory cells as in Fig. 2(b) with $R = 256$. When the wordline goes high at time $t = 0$, the memory cell develops a differential signal (voltage difference between BLT and BLC), and when the enable signal goes high at time $t = T$, the sense amplifier resolves that differential signal to a logic-level output (voltage difference between SAT and SAC). One can represent the bitcell signal of cell i as $TX_i = T(\sigma_X \tilde{X}_i + \mu_X)$ and the sense amplifier offset as $Y = \sigma_Y \tilde{Y}$ [$\tilde{Y}$ and $\tilde{X}_i$ are $\mathcal{N}(0, 1)$]. The failure of this read operation is determined by the interaction of two random variables sampled at different rates. The probability P_f that this single-column memory fails for a given strobe timing T is the probability that the sense amplifier offset overpowers the bitcell signal for one or more paths in the column

$$P_f := \Pr\left(\bigcup_{i=1}^{R} \{Y - TX_i > 0,\}\right) \quad (1)$$

$$\leq R \cdot \Pr(Y - TX_1 > 0) =: P_u \quad (2)$$

where P_u is the conservative union-bound estimate of P_f.

Because of the different rates of repetition, a proper Monte Carlo simulation on a critical path with one cell and one sense amplifier must sample variables in a nested for loop: for each sense amplifier, simulate over 256 cells and check for one or more failing paths, then sample a new sense amplifier and repeat over 256 new cell realizations and so on, as suggested in [8]. If one wishes to apply an accelerated statistical simulation to evaluate the failure of this circuit, the "for loop" sets an unfavorable lower bound on the number of simulations needed just to emulate the architecture.

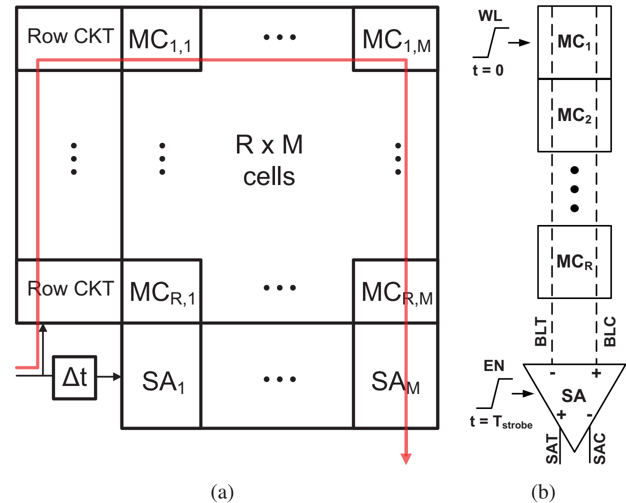

Fig. 2. (a) Representative SRAM array. (b) Simplified schematic of the small-signal read path.

We observed that the simple union-bound estimate P_u provides an accurate way to bypass this requirement. Just the path failure probability is simulated and the result is multiplied by R. The estimate is guaranteed to be conservative and in practice agrees very well at low levels of failure probability. In [11], this loop-flattening estimate was shown to be accurate for the small-signal SRAM read path. As in [8], the bitline signal development and sense amplifier offset were parametrized by Gaussian random variables—$\{\mu_X = 1$ mV/ps, $\sigma_X = 0.10 \times \mu_X$, $R = 256$, $\sigma_Y = 25$ mV$\}$ for (1). Specifically, the loop-flattening estimate was only 1.9% pessimistic in strobe timing for a modestly sized 512-KB memory (2048 memory columns) at a yield of 99%, and increased in accuracy at even lower levels of failure.

The schematic in Fig. 3(a) is the schematic tree of the large signal read path. For the case of cascaded random delays, we can also see the applicability of the loop-flattening estimate. This circuit is simulated in a production-quality 45-nm CMOS technology, where each shaded transistor (or gate input) exhibits local mismatch modeled as a fluctuation of its threshold voltage. Fig. 4 shows the Monte Carlo SPICE simulation result of this circuit for 8 cells per local bitline ($N_{\text{LBL}} = 8$) and 16 local evaluation networks ($N_{\text{SEG}} = 16$). In this picture, there is a delay Z_i ($1 \leq i \leq 256$) associated with each of the 256 cells on the column, and the probability of failure associated with a target delay t is

$$P_f := \Pr\left(\bigcup_{i=1}^{R} \{Z_i \geq t\}\right) \quad (3)$$

with $R = 256$. The solid curve gives the conventional nested Monte Carlo simulation result by sampling random variables in proportion to the rate of repetition of their associated transistors. The dashed curve gives the loop-flattening estimate in which a simple chain of representative transistors is simulated with all random variables sampled at the same rate. Even for this example, in which the delays are not perfectly normal and delay stages are correlated, the loop-flattening technique produces a tight estimate. The single solid black dot gives a specific example for how an IS simulation with an appropriately chosen mean shift can evaluate the loop flattening (dashed curve) with significant speedup, consuming approximately 1100 SPICE simulations in this example. The loop-flattening approximation suggests that this IS estimate in turn will match the result produced by a proper Monte Carlo simulation with nested sampling,

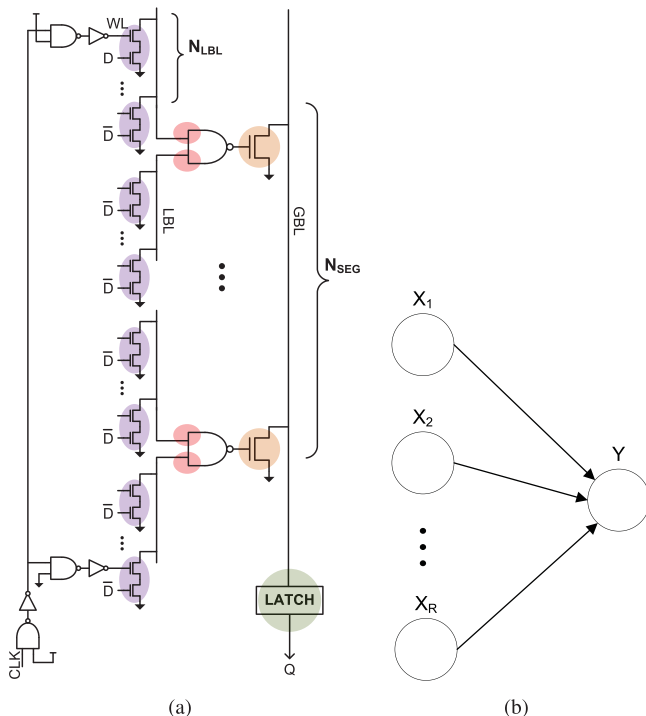

Fig. 3. (a) Schematic tree of the large signal read path. (b) Simple tree for analyzing loop flattening.

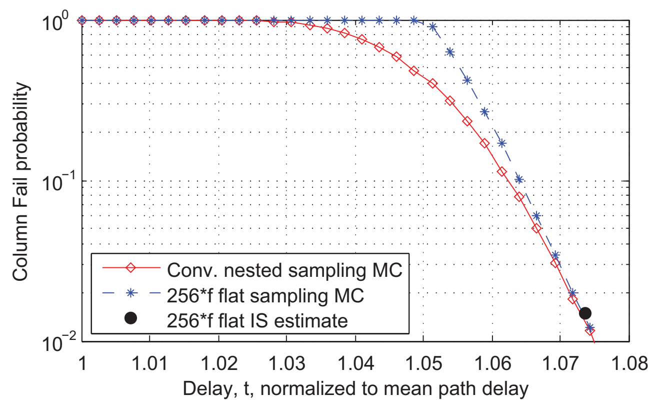

Fig. 4. SPICE simulation of the large-signal read path.

which requires 1.7 million SPICE simulations to observe the 1% failure probability of this single-column memory.

For the case of independent random additive delays, it can be shown analytically that the loop-flattening estimate converges to the true failure. Consider the simple tree in Fig. 3(b) where this time the random variables X_i and Y represent delays, and the overall delay of any path is $Z_i = X_i + Y$, associated with the node whose delay is X_i. Then, given a time t, the failure probability is defined as the probability that one or more paths in the tree exceed t as in (3). The proposed loop-flattening estimate treats these paths as independent at low probabilities and approximates the failure probability with the upper bound

$$P_u := R \cdot \Pr(Z_1 \geq t). \tag{4}$$

For X_i and Y independent normal random variables, the result in the Appendix shows that

$$\lim_{t\to\infty} \frac{P_u - P_f}{P_f} = 0. \tag{5}$$

A similar argument can be developed for the sense amplifier strobe timing case.

For finite t, the ratio in (5) represents the overestimate of the column failure probability as a fraction of the true failure probability ($P_u = P_f(1+\epsilon)$ with $\epsilon \to 0$). For a memory with M independent columns, the overall memory failure estimate from the loop-flattening approximation is

$$\begin{aligned} 1 - (1 - P_f(1+\epsilon))^M &\approx 1 - (1 - MP_f - M\epsilon P_f) \\ &= MP_f(1+\epsilon). \end{aligned}$$

Therefore, the failure of the overall memory is also overestimated by only $\epsilon \times 100\%$. For a variety of cases including numerical examples and a formal proof, the loop-flattening estimate has been justified. This approximation eliminates the constraint of nested sampling which would set an unfavorable lower bound on the required number of Monte Carlo simulations. Accelerated statistical methods such as those described in Section III can be directly applied to quickly determine the value of the loop-flattening estimate which produces an accurate and conservative value for the memory failure probability.

III. Spherical IS Cost Analysis

In this section, we introduce the Monte Carlo method and then we introduce the spherical IS method with the purpose of significantly reducing the number of SPICE simulations required. Suppose one is given a circuit that fails with probability p, and wishes to identify the value of this failure with a Monte Carlo simulation that produces an estimate $\hat{p}_{\text{MC}}$. Based on a Chernoff bound, the number of required Monte Carlo simulation runs is given by

$$N_{\text{MC}} > \frac{2\text{In}\frac{1}{\delta}}{p\epsilon^2} \tag{6}$$

where $\delta = \Pr(\hat{p}_{\text{MC}} < (1-\epsilon)p)$. In plain English, (6) says that with probability $1-\delta$, the Monte Carlo estimate $\hat{p}_{\text{MC}}$ will not underestimate the true failure p by more than $\epsilon \times 100\%$. For typical values of δ (0.01 to 0.1) and ϵ (0.05 to 0.3), this expression indicates $N_{\text{MC}} > 100/p$ to $1000/p$.

To accurately estimate low failure probabilities—10^{-6} to 10^{-10} for embedded SRAM—the standard Monte Carlo approach would require too many (10^8 to 10^{13}) SPICE simulations as seen from (6).

Fortunately, IS provides a way to speed up such a simulation, with a simple implementation (one needs only to provide an indicator of pass/fail from the SPICE simulation). This brief focuses on identifying the most likely region of failure in the parameter space, relying on the well-known notion of a worst case point [12]. The spherical IS approach defines a dominant point of failure as the point that minimizes the quadratic norm (in a parameter space normalized by the standard deviation along each coordinate). This point is then used as a mean shift in an IS simulation to quickly evaluate the circuit failure probability [5]. The detailed implementation of the method is described in [11] and it is used to evaluate the read timing failure of the small signal SRAM read path [Fig. 2(b)], comprised of a critical path netlist with 12 transistors exhibiting local mismatch variation.[1] Failure is evaluated from a transient SPICE simulation, checking for a polarity of sense amplifier differential output that is consistent with the bitcell data.

In step 1 of the method, the parameter space of threshold voltage fluctuation is sampled on concentric spherical shells, whose radii are adjusted until a sufficient number of failures are observed. For a radius that exhibits sufficient failures (1 to 250 for this work),

[1]Global variation from one chip to another is modeled as a common shift in the mean of threshold voltage distributions. This type of variation is also a concern (but does not dominate simulation cost). It must be separately treated after the impact of local variation is evaluated.

TABLE I
COMPARISON OF SIMULATION COST BETWEEN THIS BRIEF AND [5]

	This brief 12 dimensions 2-step spherical samples			[Dolecek 2008 ICCAD] 6 dimensions uniform exploration		
P	9.08×10^{-6}	1.33×10^{-7}	1.91×10^{-9}	4.9×10^{-3}	4.4×10^{-4}	3.0×10^{-6}
Step 1	500	1000	1000	-	-	-
Step 2	500	500	500	-	-	-
Total exploration	1000	1500	1500	1000	1000	2000
IS run	660	714	923	1000	2000	2000
Total	1660	2214	2423	2000	3000	4000

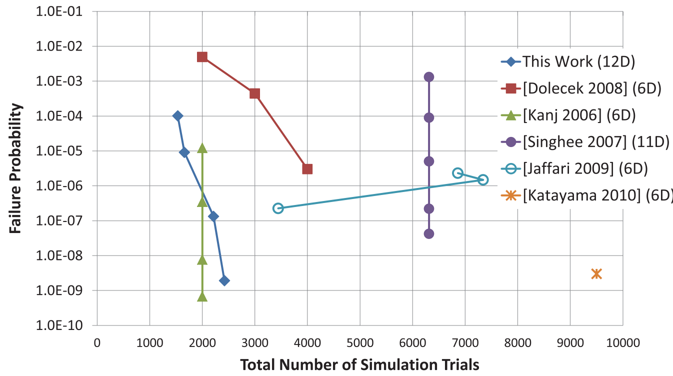

Fig. 5. Simulation cost comparison. Not shown is a point for evaluating a failure probability of 1.8×10^{-7} in 300 000 simulations in a 24-dimensional space by [6].

the failing parameters are averaged to produce an initial IS mean shift that is refined in step 2 of the method. In step 2, threshold voltage parameters are selected in the neighborhood of the current mean shift estimate and are evaluated by a SPICE simulation. The updating of the IS mean shift is designed to gravitate to points of minimum quadratic norm that reside in the failure region. Subsequent trails in step 2 sample within a shrinking neighborhood around the current IS mean shift to increase the resolution of parameter space exploration. Finally, step 3 of the method consists of a conventional IS run. As a consistent stopping criterion, empirically validated by accuracy against Monte Carlo and not to be mistaken for the overall variance of the process, the relative sample variance of the IS run is monitored until it falls to 0.1 [5].

The SRAM read path setup has been discussed in detail in [11], and here we present a full cost breakdown of the method in Table I. Across a range of probabilities from 9×10^{-6} to 2×10^{-9}, the exploration cost of spherical IS varied from 1000 to 1500 simulation runs. The subsequent IS stage took 660–923 runs. Compared to the previous ISNM work [5], at half the dimensionality (6 instead of 12) and higher probability levels, the previous simulation method remains costlier. The improvement proposed in this brief comes from the two-step spherical exploration finding a better shift for the IS run. It is also worth highlighting that the local exploration in step 2 is computationally less costly than the directional search in step 1. The two-stage spherical search effectively handles the dimensionality of 12, considering that over 4000 simulations are needed just to check all the corners of a 12-D cube. With spherical IS, much fewer directions are examined while still identifying a suitable mean shift.

A general comparison across a variety of simulation methods [3], [5]–[7], [13] is presented in Fig. 5. The y-axis gives failure probability levels, and the horizontal axis gives the number of total circuit simulations (e.g., SPICE runs) to evaluate the failure probability. Also indicated is the dimensionality of the problem (number of random variables). All methods require less than 10 000 circuit evaluations to identify failure probabilities from 10^{-3} to 10^{-9}, and the relation between failure probability and number of simulation trials is much steeper than the Monte Carlo behavior of $\approx 100/p$. The spherical IS method compares favorably with other works. Indeed, Monte Carlo simulation is much less sensitive to dimensionality than the accelerated statistical evaluation techniques in Fig. 5, which all rely on some type of classification of the parameter space. Developing an accelerated simulation method for a higher dimensional parameter space (e.g., 50) will broaden the applicability of quick statistical simulation techniques.

IV. CONCLUSION

This brief discussed two techniques—loop flattening and spherical IS—and a method to synthesize them to reduce the statistical analysis of an SRAM block to an IS simulation of a chain of component circuits. The key challenge of searching for the most likely failure mechanism in a high-dimensionality (12 in this brief) parameter space is addressed by a two-stage process in which a coarse direction is obtained first, followed by a local sampling of increasing resolution. As future work, this method can be extended to the complete global row and column path of large embedded SRAM, as well as to other highly structured circuits such as adders, finite impulse response filters, and fast Fourier transform accelerators. Such highly symmetric multiplexing structures will become more prevalent in the ascent of multicore chip design.

APPENDIX
PROOF OF THE LOOP-FLATTENING APPROXIMATION

Here, (5) is shown for the case where X_i and Y are $\mathcal{N}(0, 1)$ and independent additive delays ($Z_i = X_i + Y$).[2] Recall that failure is given by (3) and the conservative loop-flattening estimate P_u is defined in (4).

By the union bound

$$P_f := \Pr\left(\max_{1\le i\le R} Z_i \ge t\right) = \Pr\left(\bigcup_{i=1}^{R} Z_i \ge t\right) \le \sum_{i=1}^{R} \Pr(Z_i \ge t) = R \cdot \Pr(Z_1 \ge t) =: P_u.$$

Then, by incorporating the pairwise intersection probabilities, we can introduce an optimistic (lower bound) estimate P_x as

$$P_f = \Pr\left(\bigcup_{i=1}^{R} Z_i \ge t\right) \ge \sum_{i=1}^{R} \Pr(Z_i \ge t) - \frac{1}{2}\sum_{\substack{(i,j)\\ i\ne j}}^{R} \Pr\left(Z_i \ge t \bigcap Z_j \ge t\right) \ge P_u - \frac{R(R-1)}{2}\Pr\left(Z_1 \ge t \bigcap Z_2 \ge t\right) =: P_x.$$

This implies

$$\frac{P_u - P_f}{P_f} \le \frac{P_u - P_x}{P_x} = \frac{1}{\frac{2}{R-1}\frac{\Pr(Z_1\ge t)}{\Pr(Z_1\ge t\bigcap Z_2\ge t)} - 1}. \quad (7)$$

[2]This proof assumes zero-mean standard normal variables. The same arguments can be applied for the general case of nonzero mean normal random variables of different variance and will hold for generally well-behaved distributions that decay sufficiently fast.

We use the following well-known bound on the tail probability of the standard Gaussian (let $W \sim \mathcal{N}(0,1)$, $w > 0$:)

$$\frac{1}{\sqrt{2\pi}\, w}\left(1-\frac{1}{w^2}\right)e^{-\frac{w^2}{2}} < \Pr(W \geq w) \leq \frac{1}{2}e^{-\frac{w^2}{2}}. \quad (8)$$

Since $\mathrm{var}(Z_i) = 2$

$$\frac{1}{\sqrt{\pi}\, t}\left(1-\frac{2}{t^2}\right)e^{-\frac{t^2}{4}} < \Pr(Z_1 \geq t). \quad (9)$$

We now examine the intersection probability $\Pr(Z_1 \geq t \bigcap Z_2 \geq t)$. Conditioning on $Y = y$, the two events $Z_1 \geq t$ and $Z_2 \geq t$ are independent. Therefore the joint probability can be written as

$$\begin{aligned}\Pr(Z_1 \geq t \textstyle\bigcap Z_2 \geq t) &= \int_{-\infty}^{\infty} f_Y(y)\,[\Pr(X_1 + y \geq t)]^2\, dy \\ &\leq \Pr(Y \leq 0)\,[\Pr(X_1 \geq t)]^2 \\ &\quad + \int_0^t f_Y(y)\,[\Pr(X_1 + y \geq t)]^2\, dy \\ &\quad + \Pr(Y \geq t). \quad (10)\end{aligned}$$

The inequality above follows from partitioning the region of integration into $(-\infty, 0)$, $(0, t)$, and (t, ∞). For the first and third terms, the maximum value for $\Pr(X_1 + y \geq t)$ is substituted and $f_Y(y)$ is integrated. The first term is bounded by $(1/8)\exp -t^2$ and the third term is bounded by $(1/2)\exp -(t^2/2)$. The middle term can be bounded through bounds on the integrand

$$\begin{aligned}\int_0^t f_Y(y)\,[\Pr(X_1 + y \geq t)]^2\, dy &\leq \int_0^t \frac{1}{2}e^{-\frac{y^2}{2}}\frac{1}{4}e^{-(t-y)^2} \\ &\leq \frac{t}{8}\cdot \max_{y\in(0,t)} e^{-\frac{y^2}{2}}e^{-(t-y)^2}. \quad (11)\end{aligned}$$

From elementary calculus, the right-hand side bound evaluates to $(t/8)e^{-t^2/3}$. This bound on the middle term decays the slowest, and therefore the ratio $\Pr(Z_1 \geq t)/\Pr(Z_1 \geq t \bigcap Z_2 \geq t)$ grows at least as fast as $(1/t^2)e^{t^2/12}$, which grows arbitrarily large, causing the right-hand side in (7) to go to zero and in turn verifies the limit in (5).

References

[1] N. A. Kurd, S. Bhamidipati, C. Mozak, J. L. Miller, T. M. Wilson, M. Nemani, and M. Chowdhury, "Westmere: A family of 32 nm IA processors," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2010, pp. 96–97.

[2] J. Pille, C. Adams, T. Christensen, S. R. Cottier, S. Ehrenreich, F. Kono, D. Elson, O. Takahashi, S. Tokito, O. Torreiter, O. Wagner, and D. Wendel, "Implementation of the cell broadband engine™ in 65 nm SOI technology featuring dual power supply SRAM arrays supporting 6 GHz at 1.3 V," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 163–171, Jan. 2008.

[3] R. Kanj, R. Joshi, and S. Nassif, "Mixture importance sampling and its application to the analysis of SRAM designs in the presence of rare failure events," in *Proc. IEEE/ACM Design Autom. Conf.*, 2006, pp. 69–72.

[4] A. Singhee and R. A. Rutenbar, "Statistical blockade: Very fast statistical simulation and modeling of rare circuit events and its application to memory design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 8, pp. 1176–1189, Aug. 2009.

[5] L. Dolecek, M. Qazi, D. Shah, and A. Chandrakasan, "Breaking the simulation barrier: SRAM evaluation through norm minimization," in *Proc. Int. Conf. IEEE Comput.-Aided Design*, Nov. 2008, pp. 322–329.

[6] K. Katayama, S. Hagiwara, H. Tsutsui, H. Ochi, and T. Sato, "Sequential importance sampling for low-probability and high-dimensional SRAM yield analysis," in *Proc. IEEE/ACM Comput.-Aided Design Int. Conf.*, Nov. 2010, pp. 703–708.

[7] J. Jaffari and M. Anis, "Adaptive sampling for efficient failure probability analysis of SRAM cells," in *Proc. Int. Conf. Comput.-Aided Design*, 2009, pp. 623–630.

[8] M. H. Abu-Rahma, K. Chowdhury, J. Wang, Z. Chen, S. S. Yoon, and M. Anis, "A methodology for statistical estimation of read access yield in SRAMs," in *Proc. IEEE/ACM Design Autom. Conf. 45th*, Jun. 2008, pp. 205–210.

[9] R. Aitken and S. Idgunji, "Worst-case design and margin for embedded SRAM," in *Proc. Design Autom. Test Europe Conf. Exhib.*, 2007, pp. 1–6.

[10] P. Zuber, P. Dobrovolny, and M. Miranda, "A holistic approach for statistical SRAM analysis," in *Proc. Conf. Design Autom. Test Eur.*, 2010, pp. 717–722.

[11] M. Qazi, M. Tikekar, L. Dolecek, D. Shah, and A. P. Chandrakasan, "Loop flattening & spherical sampling: Highly efficient model reduction techniques for SRAM yield analysis," in *Proc. Design Autom. Test Eur. Conf. Exhib.*, Mar. 2010, pp. 801–806.

[12] K. J. Antreich and H. E. Graeb, "Circuit optimization driven by worst-case distances," in *IEEE Comput.-Aided Design Dig. Tech. Papers Int. Conf.*, Nov. 1991, pp. 166–169.

[13] A. Singhee and R. A. Rutenbar, "Statistical blockade: A novel method for very fast monte carlo simulation of rare circuit events, and its application," in *Proc. Design Autom. Test Eur. Conf. Exhib.*, 2007, pp. 1–6.

Crosstalk-Canceling Multimode Interconnect Using Transmitter Encoding

HoonSeok Kim, Chanyoun Won, and Paul D. Franzon

***Abstract*— A new implementation approach to cancel crosstalk using modal decomposition on a multiconductor transmission bundle is presented. The proposed approach requires a CODEC only at the transmitter, not at both the transmitter and receiver. This gives potential for more flexibility, lower power, better scaling, and ease of implementation. A circuit is presented along with the simulation results.**

***Index Terms*— Crosstalk cancellation, encoding, modal decomposition, multiconductor transmission, multimode interconnect.**

I. Introduction

As system bandwidth scales, there is a need for a wide-bandwidth, high-density, and high-link-per-pin channel. Density is often limited by the requirement to separate lines in order to control crosstalk noise. Several schemes to control crosstalk noise have been proposed in the past. To suppress crosstalk-induced level distortion, phase-alternating transmitter is suggested [15], but this scheme is effective only when voltage noise is weak and the propagation-delay differences among links are ignorable. Code division multiple access coding scheme was introduced for eliminating voltage noises among links [6]. However, this scheme still suffers from crosstalk-induced timing distortion. To reduce crosstalk-induced jitter, on the other hand, a timing-adjustable transmitter scheme is suggested in [7] and the usage of nonuniform channel is suggested in [8]. However, these schemes do not resolve the voltage noise.

Another approach is to use modal decomposition to formulate a CODEC that eliminates noise [1], [13]. Choi *et. al*. [1] refer

Manuscript received December 10, 2011; revised April 6, 2012; accepted July 21, 2012. Date of publication February 5, 2013; date of current version July 22, 2013. This work was supported in part by customized funding and member specific research programs from the Semiconductor Research Corporation, Intel.

The authors are with North Carolina State University, Raleigh, NC 27606 USA (e-mail: hkim7@ncsu.edu; cwon@ncsu.edu; paulf@ncsu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2012.2213281

1063-8210/$31.00 © 2013 IEEE