Efficient Statistical Analysis of Read Timing Failures in SRAM Circuits

Soner Yaldiz, Umut Arslan, Xin Li, Larry Pileggi
Electrical and Computer Engineering Department, Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
E-mail: {syaldiz, uarslan, xinli, pileggi}@ece.cmu.com

Abstract

A system-level statistical analysis methodology is described that captures the impact of inter- and intra-die process variations for read timing failures in SRAM circuit blocks. Unlike existing approaches that focus on cell-level performance metrics for isolated sub-components or ignore inter-die variability, the system-level performance is accurately predicted for the entire SRAM circuit that is impractical to analyze statistically via transistor-level Monte Carlo simulations. The accurate bounding of read timing failures using this methodology is validated with silicon measurements from a 64kb SRAM testchip in 90nm CMOS. We demonstrate the efficacy of this methodology for early-stage design exploration to specify redundancy, required sense amp offset, and other circuit choices as a function of memory size.

Keywords

SRAM, failure analysis, response surface modeling

1. Introduction

Static random access memory (SRAM) is widely used as on-chip cache for various embedded systems. The design and test of SRAM circuits have become increasingly difficult in nano-scale process technologies due to variability. Although high structural regularity in SRAM circuits reduces layout-induced systematic variations, the minimum-sized transistors used in SRAM cells suffer significantly from random variations (primarily random dopant fluctuations and line edge roughness [1]). Further exacerbating the variability problem is the increase in array size within embedded systems as technologies scale. With increasing array size, mismatch-related failure mechanisms tend to decrease the design quality considerably. For this reason, statistical analysis of SRAM circuit blocks is becoming an increasingly critical system design problem.

A particularly challenging problem is the system-level analysis of read timing failures. Firstly, a typical SRAM system contains millions of transistors, resulting in an extremely large problem size. This renders transistor-level Monte Carlo (MC) simulation for the entire SRAM circuit (that requires a read access to each cell for each random sample) impractical. Secondly, due to the numerous repeated blocks (e.g., SRAM cells, sense amplifier, etc.) on the same chip, system-level statistical analysis must take into account the variability in and the correlations among the repeated blocks. Even if there were no correlations among these blocks, the probability of a read timing failure in the whole SRAM cannot be estimated based on the failure probability of a single cell since multiple SRAM cells share the same peripheral circuitry. Finally, an SRAM system contains both digital (e.g., address decoder) and analog blocks (e.g., sense amplifier), which prevents us from directly applying existing statistical-timing algorithms for digital circuits to SRAM designs. The challenging problem is accurately modeling and analyzing such large-scale mixed-signal systems with large-scale parameter variations.

In this paper we describe a system-level statistical analysis methodology that is applicable to entire SRAM blocks. Our methodology simultaneously considers the variability in the SRAM bit-cells and other system components, such as the self-timing paths and sense amplifiers. The novelty of our approach is that the impact of both inter- and intra-die variability is taken into account for the entire system. This is accomplished with a response surface modeling framework that explicitly captures the correlations among (also known as the tracking between) the SRAM cells and the other system components. Our resulting models correlates well with results from statistical transistor–level analysis.

To demonstrate our approach, we apply our methodology to create system-level bounds for the read timing failure of a CMOS SRAM block as a function of memory size. We validate the accuracy of our methodology and bounds with comparison to silicon measurements from a 64kb SRAM testchip in 90nm CMOS. Results show that our framework can provide excellent bounds on the performance of the final block design. We further demonstrate how our methodology can be used for the early-stage design exploration that will be increasingly important with CMOS scaling. This includes demonstration of the read timing failure sensitivity with respect to memory size, self-timing circuits, sense amplifier input offset and memory redundancy.

The remainder of the paper is organized as follows: In the rest of this section, we discuss the related work and position our paper. In Section 2 we review the background on process variations and SRAM circuits. We describe the statistical modeling and analysis methodology in detail in Section 3. The efficacy of the proposed methodology is demonstrated by several design and numerical examples in Section 4. Finally, we conclude in Section 5.

1.1. Related Work

For statistical analysis of SRAM circuits, various methods have been proposed in the literature [2]-[6]. These methods either rely on transistor-level MC simulation [2]-[4] or MC simulation of response surface models of the building blocks [5]-[6]. In [4], the authors use data classifiers to filter out insignificant samples which are not likely to result in a failure, and thereby, reduce the required number of

simulations in transistor-level MC. In [2], the authors employ mixture importance sampling for the same purpose. They use a mixture of probability distribution functions, also called importance functions, to generate important random samples in the subset of the parameter space in which failures occur. Although both methods can be effective in the estimation of cell-level performance metrics such as static noise margin, write margin, etc, they can not be directly applied to read timing failures since this problem requires the consideration of variability in multiple peripheral blocks and the dimension of the process parameter space increases linearly with the memory size.

In [5] and [6], the authors employ response surface modeling techniques to overcome the dimensionality problem. In these methods, a response surface model (RSM) or a look-up table is first obtained for each block via circuit simulation. To estimate read timing failures, these regression models are used in an MC framework that offers significant speed up in runtime since evaluation of these models is faster than circuit simulation. The key drawback of these RSM-based approaches is that they consider only intra-die variation and ignore the inter-die variability that plays an important role in read timing failures. Inter-die variability needs to be modeled to capture the correlation, also referred as tracking, between the accessed SRAM cell and the self timing circuit. The RSM-based methodology we propose in this paper can efficiently capture this tracking between SRAM cells and the self-timing circuit paths.

2. Background

2.1. Process Variations

Process variations can be classified into two broad categories: *inter-die variations* and *intra-die variations*. Inter-die variations model the variations across different dies. Intra-die variations, which are also called within-die variations or on-chip variations, model the local variations within a single die and can be spatially correlated. In the state-of-the-art transistor models, inter/intra-die variations are modeled by two vectors of Normal random variables that we denote by ε_G and ε_{Li} respectively. ε_G represents correlated global variation between dies. ε_{Li} represents independent local variation in i^{th} transistor. We represent the union of ε_{Li} for each transistor i in a circuit by ε_L.

The statistical transistor models are used in a MC simulation to estimate performance distributions where the total simulation cost is:

$$\text{MC Cost} = (\text{Simulation time of a Sample}) \times (\text{Number of Samples}) \quad (1)$$

The simulation time of a sample depends on the circuit size. The number of samples depends on the desired estimation accuracy where the estimation variance is inversely proportional to the number of samples.

2.2. SRAM Architecture

The conventional 6-transistor SRAM cell that stores a single bit consists of two back-to-back inverters as shown in Fig.1. Nodes VL and VR retain the true and complemented value of the stored data. The storage nodes of multiple cells are connected via access transistors to bitline and its complement (BL and BR) forming a column as shown in Fig.2. The access transistors are controlled by the wordline (WL) signal.

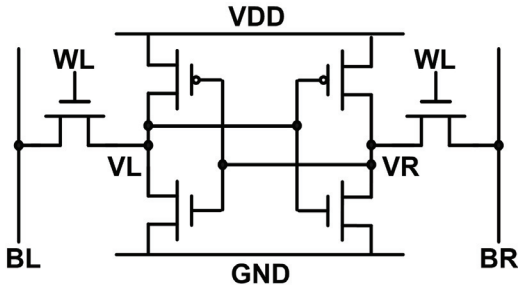

Figure 1: 6-transistor SRAM cell

To read from an SRAM cell, the access transistors are turned on under the control of the WL signal for a finite duration, and the target cell starts to discharge the pre-charged BL or BR depending on the stored data. To increase access speed and reduce power consumption, the differential voltage swing on BL and BR is designed to be less than the full logic level. This differential voltage is then converted to full logic level using a sense amplifier which is controlled by the sense enable (SE) signal. A sense amplifier can be shared between multiple columns to reduce area overhead through a column multiplexer. The delay between WL and SE, denoted by t_{WL2SAE}, controls the amount of voltage swing at the inputs of the sense amplifier, denoted by ΔV_{BL}. This delay is generated by a *self-timing circuit* which can be an inverter chain or a replica bitline circuit (RBL) [7]. RBL has the capability of tracking memory cell delay across inter-die and environmental variations; however, it is more susceptible to random device mismatches which have become more dominant in recent process technologies. Although we use an inverter chain as the self-timing path in this paper, we would like to note that our methodology can easily be applied to RBL circuits.

To improve the performance, memory is divided into banks where each bank contains N_{COL} columns and N_{ROW} rows of cells and a separate self-timing circuit. To decrease the yield loss of a memory system due to process variations or catastrophic defects a number of redundant rows and/or columns (N_{RED}) are included in each bank to replace the faulty rows and/or columns.

Fig.2 shows a simplified organization of critical blocks in read timing. In Fig.2, DRV represents a logic path which is used to match the delay between RWL and WL signals. Ideally, a sense amplifier provides a correct reading as long as there is a non-zero voltage difference at its inputs. In the presence of intra-die variability, the sense amplifier has an inherent and randomly distributed input offset voltage, denoted by V_{OFFSET}. To sense and read the stored data correctly ΔV_{BL} should be greater than V_{OFFSET} where ΔV_{BL} is also randomly distributed due to the variations in the self-timing circuit delay and the variations of the transistors discharging the bitline. A read timing failure occurs when ΔV_{BL} is less than V_{OFFSET}. We define *system-level read timing failure probability* (P_{FAIL}) as the probability that at least a

single read timing failure occurs in the whole memory. P_{FAIL} can be formulated as follows:

$$P_{FAIL} = 1 - \Pr\left\{ \bigcap_{i=1}^{N_{BANK}} \left(\bigcap_{j=1}^{N_{COL}/N_{MUX}} \left(\min_{k=1}^{N_{ROW} N_{MUX}} \left(\Delta V_{BL,ijk} \right) > V_{OFFSET,ij} \right) \right) \right\} \quad (2)$$

where N_{MUX} is the column multiplexer factor. In addition to read timing failure, self-timing path delay is also an important performance metric in SRAM design since it is a part of the total read delay. We next propose a methodology to estimate P_{FAIL} for a given SRAM design.

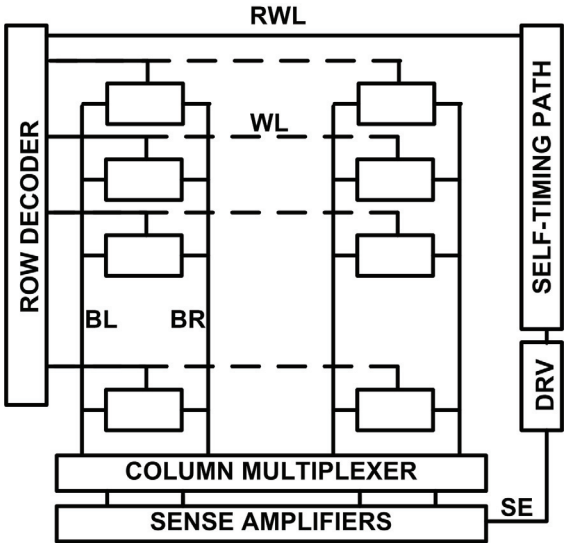

Figure 2: SRAM bank organization

3. Read Timing Failure Analysis

The most accurate way of estimating read timing failure probability is the MC simulation of the whole SRAM circuit. For each sample, every individual SRAM cell should be accessed to guarantee correct timing. Considering the fact that the memory size can be up to megabits, it is clear that Monte Carlo simulation is impractical. As it has been discussed in Section 1, the existing analysis techniques in the literature neglected inter-die variability, which plays a key role in timing failures by introducing correlation between building blocks. Even if there is no correlation among the memory blocks, the read timing failure probability cannot be estimated based on the failure probability of a single memory cell since a single self-timing path is shared by all memory cells in a bank.

The methodology we propose is able to capture both inter- and intra-die variability. This is achieved by fitting response surface models as explicit functions of inter-die variability parameters. We begin with fitting a quadratic response surface model for ΔV_{BL} as:

$$\Delta V_{BL} = x^T Ax + Bx + c \qquad x = [t_{WL2SAE}, \varepsilon'_G, \varepsilon'_L] \quad (3)$$

ε'_G in Eqn.3 is a subset of ε_G which has the dominant impact on ΔV_{BL} and t_{WL2SAE}. ε'_L represents dominant intra-die variability parameters in the accessed SRAM cell. These dominant variability parameters are determined by variable screening based on linear sensitivities. The data for model fitting is obtained by simulating a testbench circuit which contains a single SRAM column with the peripheral circuitry loading the bitlines. Consequently, ΔV_{BL} model inherently captures the leakage of the non-accessed cells on the bitlines.

For the inverter chain delay used for self-timing, we fit a quadratic model for inter-die variability and a lumped linear model for intra-die variability as follows:

$$t_{WL2SAE} = \left(y^T Ay + By + c \right) + d\varepsilon_{LUMPED} \qquad y = \varepsilon'_G \quad (4)$$

where ε_{LUMPED} is a zero-mean Normal random variable representing the effect of intra-die variability. Since the relative effect of intra-die variability on the inverter chain delay decreases with the increasing number of stages, a lumped model provides sufficient accuracy as will be shown in Section 4. Furthermore, a single testbench can be used to collect data for modeling inverter chains of varying number of stages. Similar to [6], we assume that the input offset of the sense amplifier has a Normal distribution. Through ε'_G in Eqn. 2 and 3, our analysis framework can accurately capture the tracking between the accessed SRAM cell and self-timing circuit in the presence of inter-die variability.

Once the response surface models are constructed, we use them in a MC simulation that is implemented in MATLAB. The pseudo-code of the analysis is shown in Fig. 3. The outer loop in line 2 samples inter-die variability parameters while the inner loop on line 3 samples intra-die variability parameters. For every bank, samples are generated for $N_{ROW}(N_{COL}+N_{RED})$ cells, N_{COL}/N_{MUX} sense amplifiers and a single self-timing circuit. Since response surface models are simulated to evaluate performance, the runtime of this flow is significantly lower than the MC simulation of the actual netlist. Line 6 shows how memory redundancy in each bank can be easily included in the analysis. Our approach is also applicable to the memories where redundancy is shared among banks. In the next section, we demonstrate how our methodology can be used during SRAM design.

```
1 Generate N_G samples for inter-die variability parameters
2 for i=1:N_G
3   for j=1:N_BANK
4     Generate samples for intra-die variability parameters
5     Evaluate ΔV_BL, t_WL2SAE and V_OFFSET models
6     Replace rows/columns with weak cells with redundant ones
7     If a read timing failure occurs, Failure(i) ←1
8 P_FAIL=mean(Failure)
```

Figure 3: Read timing failure analysis flow

4. Experiments

This section is organized in two subsections: In the first subsection, we validate the accuracy of our statistical analysis methodology with an SRAM design manufactured in 90nm process technology. In the second subsection, we demonstrate how our methodology can be used for design space exploration.

4.1. Accuracy Evaluation

To validate the accuracy of our methodology, we compared it with the measured results from a 64kb SRAM testchip manufactured in 90nm bulk CMOS technology [8]. The memory was organized in 256 rows and 256 columns

where four columns share a sense amplifier. The testchip had an external input pin through which t_{WL2SAE} can be controlled. We measured read timing failure probability for t_{WL2SAE} ranging from 60 to 140 picoseconds. Since only a few testchips were available to us for measurement, we only considered intra-die variability in our methodology. We approximated the inter-die variability parameters by comparing the simulated and the measured delay of a deep logic path.

The measured and the estimated bounds on the read timing failure probability are presented in Fig. 4. Our methodology is able to provide upper and lower bounds on the read timing failure even though the inter-die variability was not known precisely. Although there is a large gap for low t_{WL2SAE}, the gap decreases considerably for the higher t_{WL2SAE} values where the design is more likely to be centered.

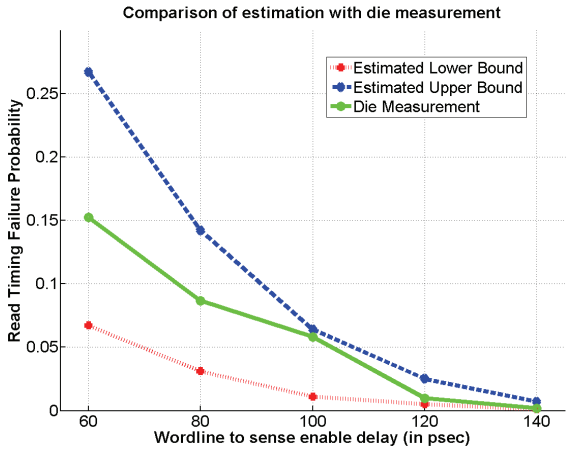

Figure 4: Estimation versus measurement

4.2. Design Space Exploration

In this section, we present three design experiments that were conducted using a commercial 65nm CMOS process and the corresponding 6-transistor SRAM cell provided by the foundry. It is assumed that there are 128 rows and 256 columns in a bank and four columns share a single sense amplifier. The memory size is varied by increasing the number of banks. We would like to note that the runtime of these experiments was on the order of hours. In Figs. 5-7, the error bars represent the 95% confidence interval of the estimated performance.

Table 1: Performance modeling summary

Performance Models	# of Active Variables	# of Circuit Simulations	Average Absolute Error
ΔV_{BL}	23	500	< 2 mV
t_{WL2SAE}	16	1250	< 1psec
V_{OFFSET}	1	1000	-

The simulation cost and the average absolute error of the response surface models are summarized in Table 1. Since we used quadratic models, the required number of simulations was on the order of the square of the number of active variables. The cost of $\boldsymbol{t_{WL2SAE}}$ and $\boldsymbol{V_{OFFSET}}$ includes 1000-sample Monte Carlo simulations to model intra-die variability. Table 1 shows that the response surface models are sufficiently accurate for timing analysis.

In the first example we used the framework for self-timing path design. We analyzed the read timing failure probability of twelve and fourteen-stage inverter chains that have two F04 delays in between. Fig. 5 compares the read timing failure probability with these inverter chains for varying memory size without redundancy. Fig. 5 confirms that a larger memory is more likely to have weak SRAM cells requiring a slower self-timing path.

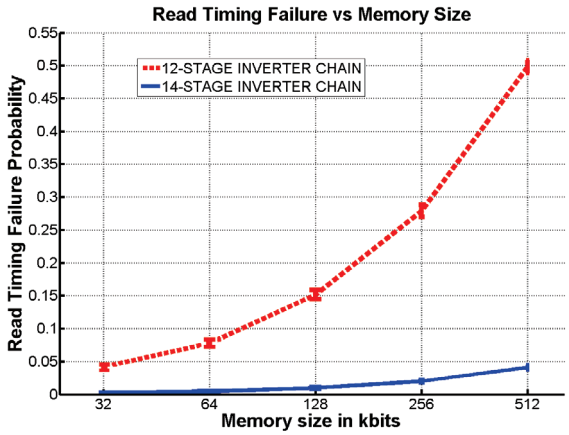

Figure 5: Read timing failure versus memory size

In the second example we used our framework to determine how memory redundancy can reduce read timing failures. For this purpose, we analyzed the read timing failure of a twelve-stage inverter chain where each bank had redundant columns. During this analysis, columns with the slowest SRAM cells are replaced with the redundant columns. Fig. 6 presents the read timing failure for three different memory sizes and for varying number of redundant columns per bank. Fig. 6 shows that 1.5% column redundancy can provide up to 25% reduction in the read timing failures. It also shows that redundancy has a diminishing return in failure reduction since read timing failures become limited by the input offset of the sense amplifiers.

In the last example we analyzed the sensitivity of read timing failure probability with respect to sense amplifier offset. We used a twelve-stage inverter chain as the self-timing path and assumed that the sense amplifier offset can be further minimized by transistor resizing or post-manufacturing configurability [9]. Fig. 7 presents read timing failure probability for three different memory sizes where the standard deviation of the sense amplifier offset is reduced by 5 to 15%. Fig. 7 shows that sense amplifier offset has a larger impact on read timing failure than redundancy although area and power overhead also needs to be considered before a final design decision can be made.

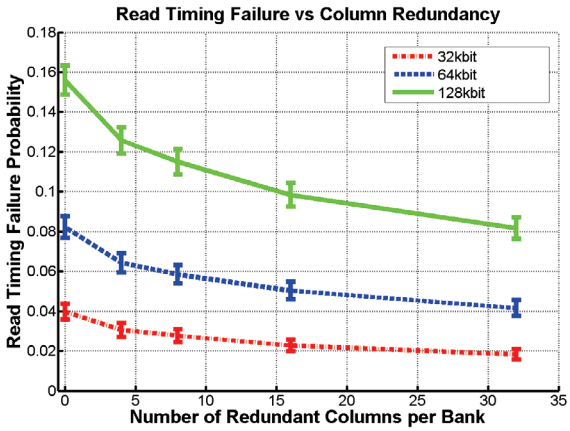

Figure 6: Read timing failure versus redundancy

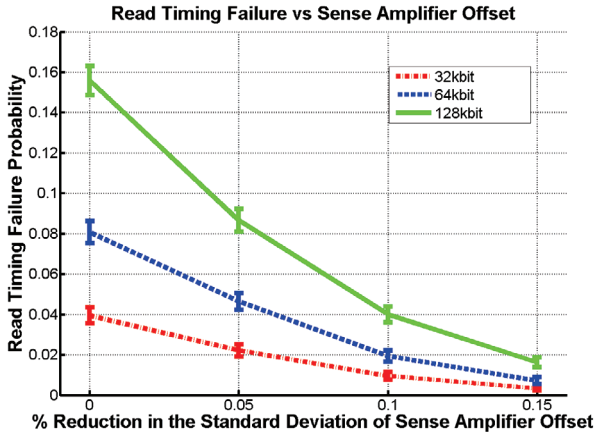

Figure 7: Read timing failure versus sense amplifier offset

5. Conclusion

In this paper we presented a system-level statistical analysis methodology for SRAM circuits that can be used for both early-stage design exploration and final design sign-off. We demonstrated how this methodology can be used to predict read timing failure probability in SRAM designs for which transistor-level Monte Carlo simulation is impractical due to prohibitive computational cost. As a part of future work, we intend to build a more comprehensive analysis framework that can analyze both read timing and read stability failures simultaneously.

6. Acknowledgments

The authors acknowledge the support of the Focus Center for Circuit & System Solutions (C2S2), one of five research centers funded under the Focus Center Research Program, a Semiconductor Research Corporation Program.

7. References

[1] Semiconductor Industry Associate, International Technology Roadmap for Semiconductors, 2005.

[2] 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," *IEEE DAC,* pp. 69-72, 2006.

[3] R. Aitken and S. Idgunji, "Worst-case design and margin for embedded SRAM," *IEEE DATE*, pp.1-6, 2007.

[4] A. Singhee and R. Rutenbar, "Statistical blockade: a novel method for very fast Monte Carlo simulation of rare circuit events, and its application," *IEEE DATE*, pp. 1379-1384, 2007.

[5] S. Mukhopadhyay, H. Mahmoodi and K. Roy, "Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS," *IEEE TCAD*, vol. 24, no. 12, pp. 1859- 1880, Dec. 2005.

[6] Houle, R.M., "Simple statistical analysis techniques to determine optimum sense amp set times", *Solid-State Circuits, IEEE Journal of*, vol. 43 no. 8, pp. 1816-1825, Aug. 2008

[7] Amrutur, B.S. and Horowitz, M.A., "A replica technique for wordline and sense control in low-power SRAM's", *Solid-State Circuits, IEEE Journal of*, vol.33, Iss.8, pp. 1208-1219, Aug. 1998.

[8] U. Arslan, M. P. McCartney, M. Bhargava, X. Li, K. Mai, L. T. Pileggi, "Variation-tolerant SRAM sense-amplifier timing using configurable replica bitlines", *IEEE Custom Integrated Circuits Conference*, Sept. 2008

[9] L. Pileggi, G. Keskin, X. Li, K. Mai, J. Proesel, "Mismatch analysis and statistical design at 65nm and below", *IEEE Custom Integrated Circuits Conference*, Sept. 2008