Importance Sampling Monte Carlo simulations for accurate estimation of SRAM yield

T.S. Doorn[1], E.J.W. ter Maten[1], J.A. Croon[1], A. Di Bucchianico[2], O. Wittich[2]
[1]NXP Semiconductors, Eindhoven, The Netherlands
[2]Eindhoven University of Technology, Eindhoven, The Netherlands

***Abstract*—Variability is an important aspect of SRAM cell design. Failure probabilities of $P_{fail} \leq 10^{-10}$ have to be estimated through statistical simulations. Accurate statistical techniques such as Importance Sampling Monte Carlo simulations are essential to accurately and efficiently estimate such low failure probabilities. This paper shows that a simple form of Importance Sampling is sufficient for simulating $P_{fail} \leq 10^{-10}$ for the SRAM parameters Static Noise Margin, Write Margin and Read Current. For the SNM, a new simple technique is proposed that allows extrapolating the SNM distribution based on a limited number of trials. For SRAM total leakage currents, it suffices to take the averages into account for designing SRAM cells and modules. A guideline is proposed to ensure bitline leakage currents do not compromise SRAM functionality.**

I. Introduction

Decades of scaling according to Moore's law have shrunk devices to such an extent that variability has become a serious issue at all levels of circuit design. The effects of variability are most noticeable in SRAM design, since SRAM cells use very small transistors. For this reason, statistics have long been part of SRAM cell design. Intra-die transistor Vt mismatch is still the main statistical parameter, although others are gaining importance. Downscaling of transistors leads to widened Vt-distributions (Figure 1 left). In addition, the amount of SRAM on large System-on-Chips (SoCs) continues to increase, causing the amount of variation that has to be taken into account to increase as well (Figure 1 right).

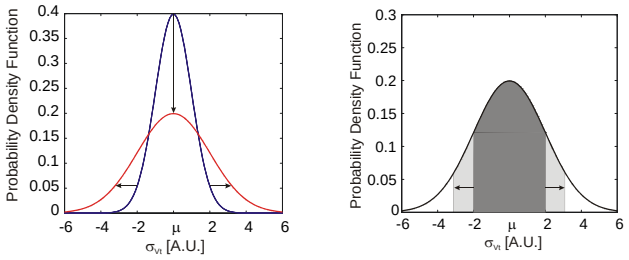

Figure 1: Increased variability leads to widening mismatch distributions (left). Increasing number of memory bits per SoC leads to a larger part of the mismatch distribution being taken into account in memory bitcell design (right).

On top of this, there is a clear trend towards voltage scalable systems [1]-[2], resulting in an increased demand for voltage scalable SRAM as well. At lower supply voltages, SRAMs are more susceptible to variability, leaving less design margin for the designer. Hence it is becoming increasingly hard to guarantee correct SRAM operation under all process, voltage and temperature conditions. This translates to very tough requirements on SRAM parameters like Static Noise Margin (SNM), Write Margin (WM) and read current (I_{read}).

SRAM yield should not be limited by design parameters. To guarantee no more than 0.1% yield loss for a 10Mb SRAM, a failure probability of $P_{fail} \leq 10^{-10}$ is taken into account in SRAM bitcell design for all relevant parameters. Provided the probability distribution is Gaussian, $P_{fail} \leq 10^{-10}$ corresponds to μ-6.4σ (with μ the mean and σ the standard deviation of the distribution). Using Monte-Carlo (MC) simulations, the 6.4σ limits of the SRAM parameter distributions are estimated. Accurate estimation of the relevant parameters at μ-6.4σ with plain Monte-Carlo takes billions of simulations and is too time consuming. Hence, a limited number of simulations is done (10^3-10^4), the μ and σ of the distribution are extracted and μ-6.4σ is determined by extrapolation. This technique is not always accurate, since the SNM distribution is not Gaussian at all [1] and the distribution I_{read} is not Gaussian in its tail.

This paper presents the use of the simplest form of Importance Sampling (IS) to drastically increase the accuracy of Monte-Carlo simulations. This technique was applied before in a complex adaptive fashion, requiring complex sampling algorithms and post-processing [3]. This paper presents a form of IS that requires less implementation effort. The applicability of the method is demonstrated by estimating the yield and probability distribution functions of SNM, WM and I_{read}. In the case of the SNM, a new method is presented for accurately estimating $P_{fail}=10^{-10}$ by extrapolation. For SRAM total leakage currents, it suffices to take the averages into account for designing SRAM cells and modules. A guideline is proposed to ensure bitline leakage currents do not compromise SRAM functionality.

II. Importance Sampling

Monte-Carlo analysis in circuit design normally assumes Gaussian distributed Vts of the transistors in the circuit. This results in many samples being drawn from around the average

of the distribution. The extreme Vts are responsible for the extremes in the distributions of the output parameters (SNM, WM, I_{read}, etc.). Therefore it makes sense to have more samples drawn from the tails of the Vt distributions. Using a Gaussian distribution with a higher standard deviation for the Vt is the simplest way to achieve this.

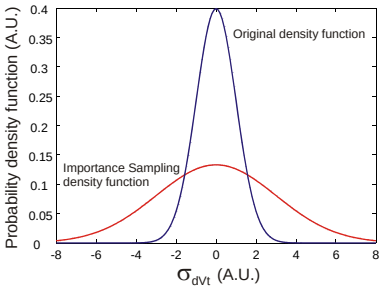

Figure 2: The principle of Importance Sampling. Using a density function with a higher standard deviation in Monte-Carlo analysis results in more samples being drawn from the extremes of the distribution.

From Figure 2 it is clear that using a wider Gaussian density function for Monte-Carlo sampling, indeed more samples are drawn from the extremes of the density. Using a wider Vt sampling distribution is a very practical choice, since no modifications to the circuit simulator are necessary. Using a wider density instead of the original distribution leads to distorted SNM, WM and I_{read} distributions. The correct density functions and distributions are obtained by a mathematical transformation based on the ratio of the original and IS distribution. The resulting distributions are now estimated over a much larger range compared to applying standard MC.

IS can be more formally described as follows. Suppose parameter x has a density $f(x)$. With IS, parameter x is sampled according to density $g(x)$. To compensate for sampling according to $g(x)$ instead of $f(x)$, the distribution function y, the sampled version of x, has to be multiplied by the ratio $f(x)/g(x)$. The sampled distribution function of parameter y is given by equations 1 and 2.

$$F^{MCIS}(y) = \frac{1}{N}\sum_{i=1}^{N} I_{\{x_i<y\}} \frac{f(x_i)}{g(x_i)} \qquad (1)$$

with

$$I_{\{x_i<y\}} = \begin{cases} 0 & x_i \geq y \\ 1 & x_i < y \end{cases} \qquad (2)$$

where N is the number of trials.

III. Application of IS to SRAM bit cell analysis

This section shows that with the same number of trials, IS MC can estimate much lower failure probabilities than is possible with normal MC. It is also shown that extrapolated MC can lead to over- or under-estimation of the $P_{fail} \leq 10^{-10}$ for the most important SRAM parameters: SNM, I_{read} and WM. Moreover, for the SNM, a new method allows estimating $P_{fail} \leq 10^{-10}$ using extrapolated MC with high accuracy.

A 65nm SRAM cell is simulated using PSP MOS transistor models. A supply voltage $V_{dd}=0.9V$ is used, to bring the cell closer to its operating limits. At this Vdd, the accuracy with which all parameters are determined becomes more important. The IS simulations use Gaussian distributions with a $\sigma = 3\sigma_{Vt}$ for the Vts of all transistors in the SRAM cell.

A. Static Noise Margin (SNM)

An SRAM cell has to be stable enough to be read without changing the data in the cell. The SNM is a measure for the read stability of the cell. The SNM is the amount of noise that can be imposed on the internal nodes of the SRAM cell before it changes its state. The SNM is determined by plotting the voltage transfer curve of one half of the SRAM cell together with the inverse of the voltage transfer curve of the other half of the cell. The sides of the largest squares that can be drawn inside the eyes are SNMh and SNMl (Figure 3). Both SNMh and SNMl have a Gaussian distribution. The minimum of SNMh and SNMl is traditionally defined as the SNM [4]. Since taking the minimum of SNMh and SNMl is a non-linear operation, the distribution of SNM is no longer Gaussian. Therefore using extrapolated MC to determine $P_{fail} \leq 10^{-10}$ does not yield accurate results.

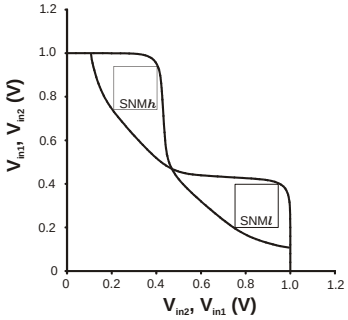

Figure 3: The butterfly curve of an SRAM cell, used to determine the SNM.

Figure 4 left, shows the cumulative distribution function (CDF) of the SNM, determined by a MC simulation using 50k trials, both for normal MC (solid) and IS MC (dotted). Normal MC can only simulate down to $P_{fail} \approx 10^{-5}$. Statistical noise becomes apparent below $P_{fail} \approx 10^{-4}$. Using the simple form of IS, $P_{fail} \leq 10^{-10}$ is easily simulated. The correspondence between normal MC and IS MC is very good down to $P_{fail} \approx 10^{-5}$. Figure 4 clearly shows that using extrapolated MC leads to overestimating the SNM at $P_{fail}=10^{-10}$.

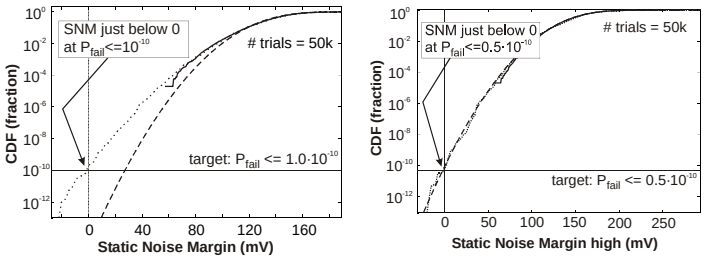

Figure 4: SNM (left) and SNM high (right) cumulative distribution function for extrapolated MC (dashed), normal MC (solid) and IS MC (dotted).

A new simple method is now presented to estimate the SNM by evaluating the distribution of only SNMh or SNMl. Figure 4 right shows the CDF of SNMh. The distribution of SNMh is a Gaussian distribution and extrapolation leads to a good estimate of SNMh at $P_{fail} \leq 10^{-10}$. The $P_{fail} \leq 10^{-10}$ limits for SNMh and SNM appear be to almost identical. At first sight, this is surprising, since the SNM and SNMh have different distributions. However, a small difference exists between SNM and SNMh/SNMl. The following describes how they are different.

The SNM is defined as the smaller value of *SNMh* and *SNMl*

$$SNM \equiv \min\left(SNMh, SNMl\right) \tag{3}$$

Also, using a probability rule,

$$P\left(A \cup B\right) = P\left(A\right) + P\left(B\right) - P\left(A \cap B\right) \tag{4}$$

We now apply equations (3) and (4) with $A=\{SNMh \leq a\}$ and $B=\{SNMl \leq a\}$. It is geometrically obvious from the butterfly curve that SNMh and SNMl cannot simultaneously be small. Therefore $P(A \cap B)=0$. Assuming that SNMh and SNMl are identically distributed, it follows for the values of interest for *a* that:

$$\begin{aligned} P\left(SNM \leq a\right) &= P\left(SNMh \leq a\right) + P\left(SNMl \leq a\right) \\ &= 2P\left(SNMh \leq a\right) \\ &= 2P\left(SNMl \leq a\right) \end{aligned} \tag{5}$$

A failure probability for SNMh of $P(SNMh \leq a)=0.5 \cdot 10^{-10}$ is required to get the same failure probability $P(SNM \leq a)=10^{-10}$. In the example shown in this paper, the difference between a for $P(SNMh \leq a)=0.5 \cdot 10^{-10}$ and $P(SNM \leq a)=1.0 \cdot 10^{-10}$ is only 1.2mV, which is within the statistical accuracy of IS MC. The extrapolated version of $P(SNMh<a)=0.5 \cdot 10^{-10}$ deviates from $P(SNM \leq a)=1.0 \cdot 10^{-10}$ by only 0.3mV. Effectively, using $P(SNMh<a)=0.5 \cdot 10^{-10}$ means extrapolating to μ-6.5σ. This analysis shows it is possible to use extrapolated MC as an accurate estimate of the far tail of the SNM distribution.

B. Read Current

The read current is a measure for the speed of the memory cell and is therefore an important parameter. Figure 5 shows the extrapolated MC, regular MC and IS MC distribution for the read current of an SRAM cell. Again, there is a good match between regular MC and IS MC, down to $P_{fail} \leq 10^{-4}$.

These read current simulations were done on one side of the cell. Therefore, $P_{fail} \leq 0.5 \cdot 10^{-10}$ has to be targeted for the read current as well. The correspondence with the SNMh simulation is very good. The cells start flipping during a read action at almost exactly the same failure probability as where SNM=0mV.

These simulations show that extrapolated MC can result in serious underestimation of the read current. This can lead to over-design of the memory cell. To be able to accurately simulate the worst case read current as a result of mismatch, IS MC is required for simulating the read current. Extrapolated MC is by no means accurate enough.

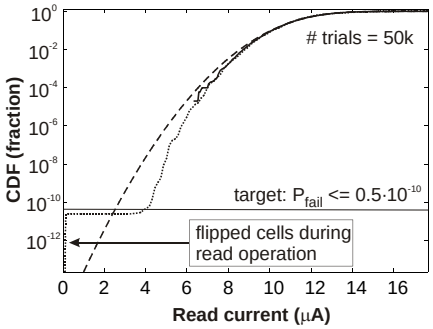

Figure 5: Read current Cumulative Distribution function of the extrapolated distribution (dashed), regular Monte-Carlo (solid) and IS Monte-Carlo (dotted).

C. Write Margin

An SRAM cell should not only be stable during read, it also has to be sufficiently instable to be written when desired. The write margin is a measure for the writeability of the SRAM cell. A cell is written by precharging one bitline to Vdd and discharging the other bitline to ground, with the wordlines at Vdd. The write margin can be defined as the highest acceptable voltage on this low bitline (Figure 6).

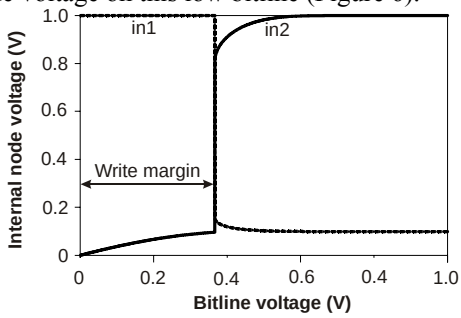

Figure 6: The internal node voltages of an SRAM cell versus the low bitline voltage. The write margin (WM) is defined as the highest bitline voltage at which de SRAM cell flips.

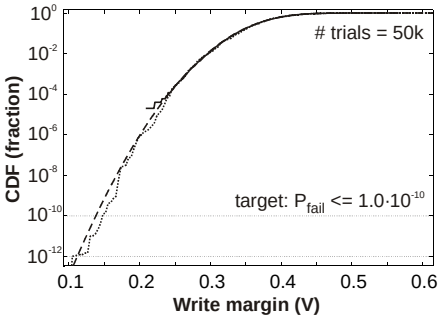

Figure 7: Write margin Cumulative Distribution function of the extrapolated distribution (dashed), regular Monte-Carlo (solid) and IS Monte-Carlo (dotted).

The distribution function of the write margin was also simulated using extrapolated MC, normal MC and IS MC (Figure 7). Again, a good match is obtained between normal

MC and IS MC. The WM is underestimated by about 10 mV, which is not a significant deviation. Therefore the far tail of the WM distribution can be estimated using extrapolated MC.

D. Leakage currents

Leakage currents can be divided into two important components: total leakage current and bitline leakage current. Total leakage current is important for the standby power consumption of the memory. This can be estimated by multiplying the average of the total cell leakage by the number of cells in the memory instance. The large number of cells in an SRAM results in a small variation on this estimate, making this method sufficiently accurate.

Bitline leakage is the sum of the leakage currents of the non-selected cells in the column being accessed. Too much bitline leakage current can result in a non-functional memory. During reading, one of the two bitlines of the column is discharged to develop sufficient differential voltage for the sense amp to detect. In a worst case situation, all non-accessed cells connected to the column being read are discharging the opposite bitline with their leakage currents. If the sum of the leakage currents is in the order of the worst-case read current, there is a risk of developing insufficient differential voltage on the bitlines and a read failure.

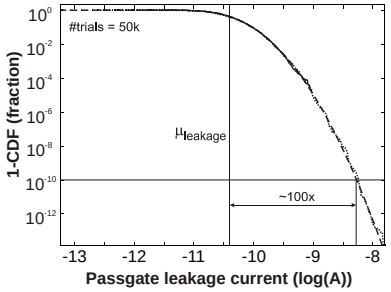

Figure 8: 1-CDF of the logarithm of the Passgate leakage current: extrapolated MC (dashed), regular MC (solid) and IS MC(dotted).

Short columns with fewer cells have lower bitline leakage currents than longer columns. Hence, if a memory with long columns can handle the worst case bitline leakage, a smaller instance of that memory with shorter columns can also handle the bitline leakage.

Figure 8 shows the logarithm of the passgate leakage current. Since the leakage current depends exponentially on the transistor Vt, the distribution of the logarithm is excellently Gaussian. The probability of a passgate leakage current that is 100x higher than the average is approximately $P(I_{leak,pg} \geq 100\ I_{leak,pg,\mu}) \approx 10^{-10}$ for this cell, meaning this is a very rare event. Hence it is safe to assume only one cell has worst-case leakage and all other cells have an average leakage current. Equation 6 is proposed as a guideline to ensure bitline leakage does not compromise SRAM functionality.

$$I_{read,wc} \geq x \cdot \left(I_{leak,pg,6.4\sigma} + (L-2) I_{leak,pg,\mu} \right) \qquad (6)$$

where $I_{read,wc}$ is the worst case read current, L is the maximum number of cells in a column and x is a margin factor at the discretion of the designer.

IV. CONCLUSION

Continuous scaling according to Moore's law and an increasing number of bits used in SRAM memories strongly increase the need for incorporating statistical information into the design of SRAM bit cells. To guarantee sufficient yield for a 10 Mb SRAM, failure probabilities of $P_{fail} \leq 10^{-10}$ are required, probabilities found in the far tails of the parameter distributions. Accurate statistical techniques are a must to be able to simulate such failure probabilities.

In this paper it is shown that accurate statistical DC SRAM cell simulations are possible using a relatively simple statistical technique like Importance Sampling (IS) Monte Carlo (MC) with widened Vt distributions. The technique has been successfully applied to accurately estimate the distributions of Static Noise Margin (SNM), Write Margin (WM) and read current I_{read}.

For the SNM, it is shown that extrapolation of normal MC simulations overestimates the yield. In addition to the benefit of IS MC simulations, it has been shown that extrapolation of the Gaussian distributions of the individual eyes yields results in accurate yield estimation. The results of the latter method are in agreement with IS MC simulations.

The read current distribution deviates strongly from a Gaussian distribution and its distribution can therefore not be extrapolated. The use of extrapolated distributions would result in a pessimistic I_{read} and could thus lead to over-design of the memory cell and/or memory architecture. Importance Sampling or a technique with similar statistical accuracy is required to make correct decisions in the design process.

The WM can be estimated with extrapolated Gaussian distributions. Although a small difference of the WM at $P_{fail} \leq 10^{-10}$ is observed between extrapolated MC and IS MC, this difference is not significant.

To determine the SRAM total leakage currents the average current per cell is multiple by the number of cell in the instance. A guideline is proposed to guarantee that bitline leakage currents do not compromise SRAM functionality.

V. ACKNOWLEDGEMENTS

Roelof Salters, Patrick van de Steeg, Jwalant Mishra and Dick Klaassen (all NXP Semiconductors) are acknowledged for many fruitful discussions.

REFERENCES

[1] B.H. Calhoun, A.P. Chandrakasan, "Static Noise Margin Variation for sub-threshold SRAM in 65-nm CMOS", IEEE Journal of solid-state circuits, vol. 41, no. 7, pp. 1673-1679, July 2006

[2] H. Onoda et al., "0.7V SRAM technology with stress-enhanced dopant segregated Schottky (DSS) source/drain transistors for 32 nm node", Symp. on VLSI Technology digest of technical papers, pp. 76-77, 2007

[3] R. Kanj, R. Joshi, S. Nassif, "Mixture importance sampling and its application to the analysis of SRAM designs in the presence of rare failure events," Design Automation Conference, pp. 69-72, July 2006.

[4] E. Seevinck, F.J. List, J. Lohstroh, "Static Noise Margin Analysis of MOS SRAM cells", IEEE Journal of solid state circuits, vol. 22, no. 5, pp. 748-754, October 1987.