Statistical Modeling for the Minimum Standby Supply Voltage of a Full SRAM Array

Jiajing Wang[1], Amith Singhee[2], Rob A. Rutenbar[2], and Benton H. Calhoun[1]

[1]University of Virginia
Charlottesville, VA 22904
{jjwang, bcalhoun}@virginia.edu

[2]Carnegie Mellon University
Pittsburg, PA 15213
{asinghee, rutenbar}@ece.cmu.edu

***Abstract*— This paper presents two fast and accurate methods to estimate the lower bound of supply voltage scaling for standby SRAM/Cache leakage power reduction of an SRAM array. The data retention voltage (DRV) defines the minimum supply voltage for a cell to hold its state. Within-die variation causes a statistical distribution of DRV for individual cells in a memory array, and cells far out the tail (i.e. >6σ) limit the array DRV for large memories. We present two statistical methods to estimate the tail of the DRV distribution. First, we develop a new statistical model based on the connection between DRV and Static Noise Margin (SNM). Second, we apply our Statistical Blockade tool to obtain fast Monte-Carlo simulation and a Generalized Pareto Distribution (GPD) model for comparison. Both the new model and the GPD model offer a high accuracy (<2% error) and a huge speed-up (>$10^4\times$ for 1G-b memory) over Monte-Carlo simulation. In addition, both models show a very close agreement with each other at the tail even beyond 7σ.**

I. INTRODUCTION

Standby leakage power can dominate the total power budget of memories or SoCs that dedicate increasingly large percentages of die area to memory. Supply voltage (V_{DD}) scaling is an effective approach for leakage power savings during SRAM/Cache standby mode [1]. Besides the direct effect of smaller voltage on power saving, V_{DD} scaling reduces both sub-threshold leakage current (due to drain induced barrier lowering (DIBL)) and gate leakage. Lowering V_{DD} as far as possible maximizes leakage power savings. However, lowering V_{DD} too far results in data loss. The data retention voltage (DRV) is the lower bound of the standby supply voltage that still preserves data in the bitcells [2].

Within-die device variation (i.e. mismatch) creates a statistical distribution of the DRV for the individual cells in a memory array. Monte-Carlo (M-C) simulation is a well-known existing approach that can provide the worst-case DRV for an SRAM array given the array size and the statistical parameters of the device variation. However, M-C simulations can be quite time-consuming for large arrays requiring long tail simulations (i.e. >6σ). Fig. 1 is the histogram of a 5k-point M-C simulation showing the DRV for SRAM bitcells that have normally-distributed within-die threshold voltage (V_T) variation. Since the DRV is not distributed normally, small Monte-Carlo simulations cannot be extrapolated using a normal distribution to model the tail.

This work was supported by the MARCO/DARPA Focus Research Center for Circuit and System Solutions (C2S2)

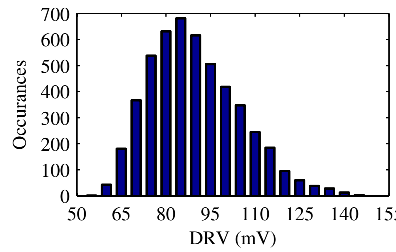

Fig. 1: The histogram of DRV from a 5k-point M-C simulation of bitcells having within-die V_T variation.

An alternative approach is to develop a theoretical model of the DRV distribution. Although [2] provides an analytical model for the DRV of an individual cell, a model of the DRV distribution is necessary for determining the DRV of a full SRAM array, because the worst case tail of the DRV distribution determines the lower bound of V_{DD} for the whole SRAM. In this paper, we propose a new statistical model for the DRV distribution that allows us to estimate the array-wide DRV for an SRAM array of arbitrary size. We base our method on the connection between DRV and static noise margin (SNM). We also show that the Statistical Blockade tool [3], which is designed to model the behavior on statistical tails, produces an accurate estimation for the tail of the DRV distribution.

The rest of the paper is organized as follows: Section II discusses new insight regarding the data dependency of DRV for a single cell and the connection between DRV and SNM. Section III gives the details of our statistical method to estimate the worst DRV based on SNM and compares our models with M-C simulation. Section IV presents approaches to improve DRV for memories in deeply scaled technologies. The conclusions are drawn in Section V.

II. DRV AND SNM

This section describes the connection between DRV and SNM for a 6T SRAM cell. The DRV is the minimum V_{DD} for retaining the cell data. Fig. 2 shows that V_{DD} scaling can cause failure in two ways. If the cell is balanced (symmetric), then its internal nodes Q and QB converge to a metastable point as a result of degraded gain, making the '0' and '1' states indistinguishable (Fig. 2a). In contrast, an imbalanced (asymmetric) cell will flip to its more stable state (Fig. 2b), causing it to have a higher DRV than the balanced one. This

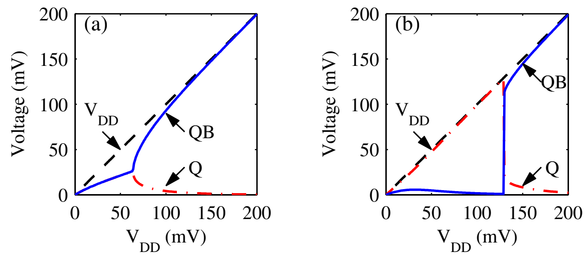

Fig. 2: DC sweep of (a) balanced and (b) imbalanced cells.

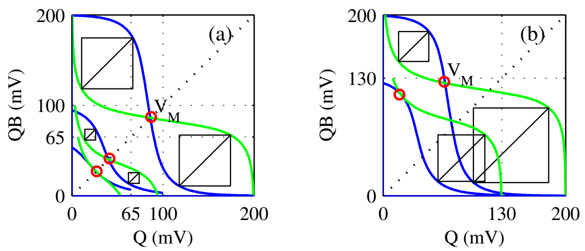

Fig. 3: VTCs of (a) balanced and (b) imbalanced cells with varying V_{DD}; V_M is the trip point of the VTCs.

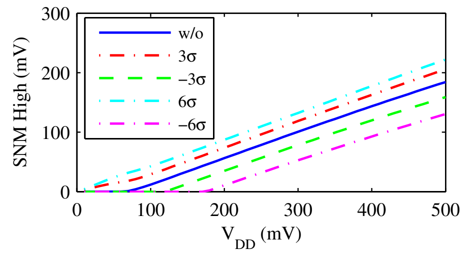

Fig. 4: SNM High versus V_{DD} with V_T mismatch in one transistor.

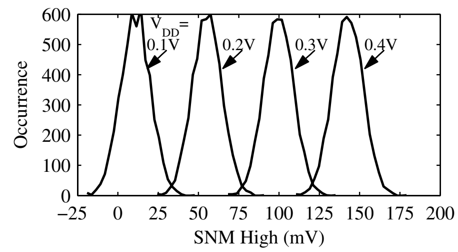

Fig. 5: The distribution of SNM High for a cell with mismatch has similar σ value across different V_{DD} values.

data dependence of the imbalanced cell can be better understood by examining SNM, the well-known measure of the maximum amount of voltage noise that a cell can tolerate [4]. Fig. 3 shows the butterfly curves that illustrate SNM (length of the side of the embedded square) for the two bitcells from Fig. 2. Fig. 3a shows that symmetry allows the cell to remain bistable to lower V_{DD}. Furthermore, it becomes clear that the DRV equals the supply voltage at which SNM is equal to zero in a noiseless system. As shown in Fig. 3a, both SNM High (upper-left square) and SNM Low (lower-right square) decrease symmetrically to zero, so the DRV is same regardless of the data stored in the cell. However, the stability of the imbalanced cell (and its DRV) strongly depends on the data pattern. For example, in Fig. 3b this particular imbalanced cell always has a larger SNM Low, and its SNM High decreases to zero at a lower V_{DD}. Therefore, this imbalanced cell is more sensitive to V_{DD} when Q='0', and its DRV is set by this worst-case data value.

It should be noted that when V_{DD} is reduced to the DRV, all six transistors in the SRAM cell are in the sub-threshold region. The impact of different parameters on SNM for a sub-threshold bitcell was shown in [5]. Because of the natural connection between SNM and DRV, DRV has similar dependencies. These include a nearly linear dependence on temperature, a relatively weak dependence on sizing, and a strong dependence on local variation (which causes the imbalanced cell scenario) [5]. Thus, we can exploit our understanding of SNM for sub-threshold bitcells to develop a model for DRV distribution.

III. Modeling DRV for a Full SRAM Array

A. Proposed New Statistical DRV Model Based on SNM

Since the DRV occurs when SNM is equal to zero, our method is based on analyzing SNM then utilizing those results to model the DRV.

Fig. 4 shows that the change of SNM High (SNMh) with V_{DD} is almost linear before reaching the DRV point (i.e. SNM=0). Also, the slope of SNMh is approximately unchanged regardless of the V_T mismatch in the cell transistors. Therefore, we assume the first-order differential coefficient $\partial SNMh/\partial V_{DD}$ is a constant value 'k' which is independent of variations. We can approximate SNMh with the first order model (1) as in [2]

$$SNMh = k \cdot V_{DD} + c. \quad (1)$$

The coefficient 'k' is extracted from a simple dc-sweep SPICE simulation of a balanced cell. However, different mismatch values will change the offset value 'c' which implies that DRV does change with mismatch, so we need to quantify the impact of variation on SNM.

Monte-Carlo simulation with random independent V_T mismatch in all transistors indicates that both the SNM High and SNM Low are normally distributed [5]. Fig. 5 plots the SNM High distribution of the bitcell with varying V_{DD}. Although the distributions have different mean (μ) values, their standard deviation (σ) values are almost constant. This is reasonable because σ of the SNM High distribution is determined by the V_T mismatch, and changes in V_{DD} do not alter V_T mismatch. In addition, μ of SNM High at a certain V_{DD} is approximately equal to the ideal value without mismatch, which can be obtained using (1). Therefore, if the SNM High at a specific supply V_0 is a Gaussian with mean μ_0 and standard deviation σ_0, then the SNM High at supply voltage x is also a Gaussian with mean $\mu=\mu_0+k\cdot(x-V_0)$ and standard deviation $\sigma=\sigma_0$. We can extract μ_0 and σ_0 from a small-scale Monte-Carlo simulation (e.g. 1.5k to 5k points). Since the SNM Low distribution has a similar statistical characteristic, it can also be estimated by using the same Gaussian as SNM High.

The actual SNM is the minimum of SNM High and SNM Low, and its PDF and CDF can be approximated by the model in [5], which gives a good estimate of the tail of the SNM distribution. We expand on that model to calculate the probability that SNM is less than s at the supply voltage x, which can be expressed as

$$P(SNM \le s, V_{DD} = x) = erfc(-\frac{s-\mu}{\sqrt{2}\sigma}) - \frac{1}{4}\left(erfc(-\frac{s-\mu}{\sqrt{2}\sigma})\right)^2 \quad (2)$$

where erfc(·) is the complementary error function, and μ and σ are the mean and standard deviation of SNM High at that supply voltage *x*. Since the DRV occurs at SNM=0, the CDF of DRV is

$$F_{DRV}(x) = 1 - P(SNM \le 0, V_{DD} = x) \quad (3)$$

By substituting (2) and the previous expressions of μ and σ related to μ_0 and σ_0, we can get the final CDF model of the DRV distribution:

$$F_{DRV}(x) = 1 - erfc\left(\frac{\mu_0 + k(x - V_0)}{\sqrt{2}\sigma_0}\right) + \frac{1}{4}\left(erfc\left(\frac{\mu_0 + k(x - V_0)}{\sqrt{2}\sigma_0}\right)\right)^2 \quad (4)$$

We also provide the inverse CDF of the DRV distribution:

$$F_{DRV}^{-1}(x) = \frac{1}{k}\left(\sqrt{2}\sigma_0 \cdot erfc^{-1}\left(2 - 2\sqrt{x}\right) - \mu_0\right) + V_0 \quad (5)$$

where $erfc^{-1}(\cdot)$ is the inverse function of erfc(·). Now we can get a fast estimate for the worst-case tail of the DRV distribution by using (5). Here is the procedure for applying our model:

Step 1. Extract 'k' from a short dc-sweep of SNM vs V_{DD}.
Step 2. Extract μ_0 and σ_0 from a 1.5k to 5k-point M-C simulation of SNM High at V_{DD}=V_0 (we will comment in Section III-C on how to select V_0).
Step 3. Use (4) to find P(DRV ≤ voltage *x*) *or*
Step 4. Use (5) to calculate the V_{DD} that is necessary to ensure that P(DRV ≤ V_{DD}) = *x*.

B. Statistical Blockade Tool and Statistical Tail Modeling

Monte-Carlo simulation of phenomena such as array-wide DRV can take huge amounts of time. As the memory size increases and samples from far out the tail are required, this simulation delay becomes untenable. Our new Statistical Blockade (SB) tool [3] improves upon traditional M-C for simulating rare events. To reduce simulation time, the Blockade tool classifies the possible M-C samples *prior* to simulation and selects only a subset of them that are likely to appear on the tail for simulation. After simulating this subset of points, the tool identifies the true tail points and uses them to fit a Generalized Pareto Distribution (GPD) model to the tail [3]. This statistical model allows estimation of events even farther out in the tail of the distribution of interest. In the next section, we show how we used the SB tool to verify the statistical DRV model and how the GPD model produced by the tool closely matches the actual DRV distribution.

C. Analyzing the Proposed DRV Models

We used an industrial 90nm technology to test the DRV models. For the new model, we calculated k=0.425 from a DC sweep simulation as in Step 1. We selected 100mV as V_0 and obtained the parameters μ_0=11.0mV and σ_0=9.3mV from a 5k-point M-C simulation as in Step 2. Fig. 6 shows the semilog plot of the probability that SNM≤0 obtained from (2) (with *s*=0) with varying V_{DD}. The curve confirms that reducing V_{DD} leads to the higher probability of negative SNM, i.e., lower reliability of the SRAM. The dashed lines in the plot show that the probability that the SNM is less than or equal to 0 is 10^{-5} with a 170mV supply. In other words, the DRV for a 100kb memory is 170mV. In addition, we can also get the probability trend for a memory that must tolerate a certain amount of noise by setting s>0 (e.g. 20mV), which allows us to redefine the DRV to an SNM of 20mV.

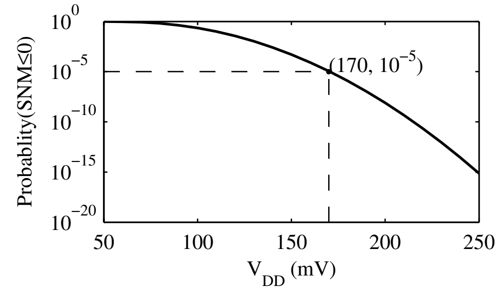

Fig. 6: Probability of SNM≤0 versus V_{DD}, using (2).

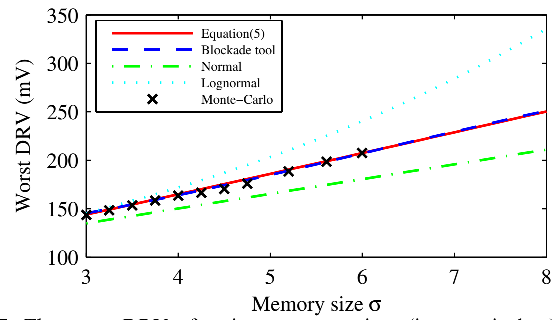

Fig. 7: The worst DRV of various memory sizes (in σ equivalent) from different approaches; our new model (5) and the GPD model from the Statistical Blockade tool [3] (lines coincident on the plot) closely track Monte-Carlo simulation *and* match farther out the tail.

For a given-size memory, the worst DRV of the entire memory is actually determined by the failure probability constraint, (n+1)/m, where n is the number of erroneous bits that can be tolerated and m is the size of the memory in bits. For memories that can tolerate some bit errors (e.g. because they are correctable using redundancy or ECC), the worst DRV value will be lower and more leakage power saving can be achieved. For a fault-free memory (i.e. having no ECC), n=0. So the critical failure probability threshold is equal to 1/m. Here, we will use this fault-free memory as an example. However, it is easy to extend the use of our statistical model to apply to a fault-tolerant memory.

Fig. 7 shows the worst DRV for a fault-free memory calculated by our new statistical model (e.g. Step 4), and the GPD model generated by Statistical Blockade, and compares them with Monte-Carlo simulation. The size of the memory is represented by the corresponding sigma value (For example, 6σ stands for a ~1G-b memory). Results from Equation (5) closely track the M-C results with an average error of 1.3% out to 6σ. M-C points greater than 5σ were simulated using the selected points from the Blockade tool classifier, thus allowing dramatically reduced simulation time. The GPD model produced by the Blockade tool also closely matches the M-C data with an average error of 1.0%. In addition, the two models match each other even at the 7~8σ tail, which is too time-consuming for using filtered Monte-Carlo. This matching of independently derived models increases the confidence that they are correct. We also show the estimation from Normal and Log-normal models that were based on a 5k-point M-C simulation for comparison. The Normal model underestimates DRV while the Log-normal model overestimates it.

The primary advantage of our new statistical model is a significant speedup compared with M-C for large memories. Specifically, we replace a M-C simulation of m-bits with a much smaller simulation of only a few thousand bits. So for a 1G-b memory, our model provides a speed-up about 5 orders of magnitude. Fig. 8 shows that the average error of our model for estimating the tail of the DRV distribution is ≤3% for an M-C simulation of greater than 1.5k points in Step 2. Likewise, the sensitivity of our model to the parameters k, μ_0, σ_0 and V_0 is quite small. Fig. 9 shows that the absolute average error rate of our model over M-C is <6% even when those parameters vary. The voltage V_0 should be selected in the sub-threshold region, and Fig. 9(d) suggests that a choice of V_0 closer to the DRV of an ideal cell decreases error.

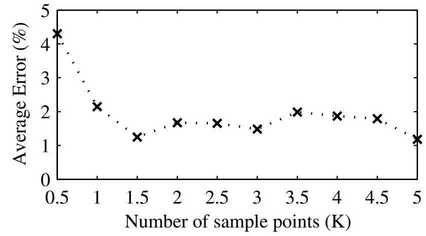

Fig. 8: Average error rate of our new model over M-C vs. the number of sample points for SNM High M-C simulation at V_0.

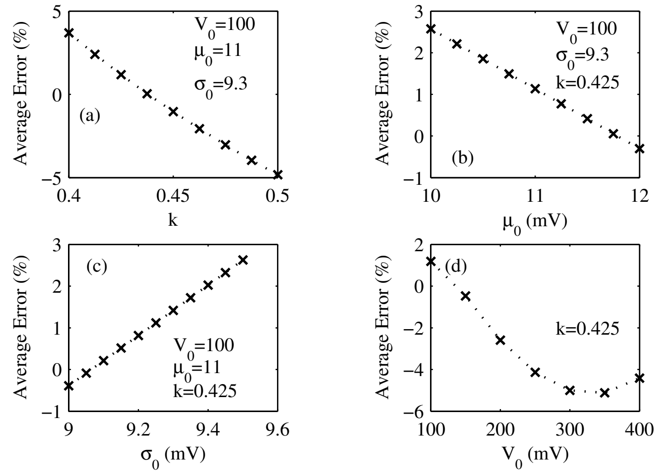

Fig. 9: Average error rate of our new model over M-C changes slightly with the altering of (a) k, (b) μ_0, (c) σ_0 and (d) V_0.

IV. DRV REDUCTION

With technology scaling, variation becomes more and more severe, which leads to higher DRV and degrades leakage power saving. Therefore, improving DRV is important, and this section describes general techniques for decreasing the DRV of an SRAM design.

Process P/N strength at the typical corner has a strong impact on SNM, and thus DRV, in the sub-threshold region where it is set by parameters like V_T instead of mobility [6]. The effect of P/N strength mismatch as well as global process variation can be reduced by body biasing. To improve DRV, we should move a given process (e.g. using adaptive body biasing [7]) towards being balanced. Using larger transistors in the bitcell can also improve DRV. The standard deviation of the threshold voltage is proportional to $(W_{eff}L_{eff})^{-1/2}$ [8], where W_{eff} and L_{eff} are the effective FET channel width length. Larger transistor sizes lead to a reduction of the spread of local threshold voltage variation and thus reduce the impact of mismatch on DRV. Bitline leakage can impact DRV significantly when mismatch makes the access transistor at the '0' side stronger. Bitline leakage reduction techniques, such as negative wordlines or floating bitlines, can be used with supply voltage scaling to reduce the impact of bitline leakage on DRV.

V. CONCLUSIONS

Local variation, or mismatch, has the largest impact on DRV and causes a spread of DRV for cells in the same SRAM array. The worst-case tail of the DRV distribution becomes the critical metric and sets the DRV for the whole memory. Based on the relationship between DRV and SNM, we proposed a statistical model to estimate the worst DRV value for an entire memory with a given size and error-tolerant ability. Our new model is accurate to within a few percent even out to 6σ compared with Monte-Carlo simulation. And it shows a close agreement with the GPD model from the Statistical Blockade tool at the tail out to 8σ. Furthermore, it replaces computationally costly Monte-Carlo runs with a single small-scale M-C simulation. It thus offers a 10^3~10^5× speedup compared with traditional Monte-Carlo simulation for a full memory with 1M~1G bits. The Statistical Blockade tool also produces an accurate statistical model of the DRV tail and offers a ~10^4× speed-up over Monte-Carlo for a 1G-b memory. For DRV tail estimation, our new model is about 10 times faster than the SB tool, which is a more generic approach for tail modeling.

REFERENCES

[1] R. Krishnamurthy, et al., "High-performance and low-power challenges for sub-70 nm microprocessor circuits," *CICC,* 2002.

[2] H. Qin, et. al., "SRAM leakage suppression by minimizing standby supply voltage," *ISQED*, 2004.

[3] A. Singhee and R. A. Rutenbar, "Statistical Blockade: A novel method for very fast Monte Carlo simulation of rare circuit events, and its application," *DATE*, 2007.

[4] E. Seevinck, F. List, and J. Lohstroh, "Static noise margin analysis of MOS SRAM cells," *JSSC*, vol. SC-22, no. 5, pp. 748-754, Oct. 1978.

[5] B. Calhoun and A. Chandrakasan, "Static noise margin variation for sub-threshold SRAM in 65-nm CMOS", *JSSC*, vol. 41, No. 7, pp. 1673-1679, July 2006.

[6] J. Ryan, J. Wang, and B. Calhoun, "Analyzing and modeling process balance for sub-threshold circuit design," *GLSVLSI*, 2007.

[7] J. W. Tschanz, J. T. Kao, et. al, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage", *JSSC,* vol. 37, No. 11, pp.1396-1402, Nov 2002.

[8] A. Asenov, et.al, "Simulation of intrinsic parameter fluctuations in decananometer and nanometer-Scale MOSFETs," *IEEE Trans. Electron Devices* 50, pp.1837-1852, 2003.