

Worst-Case Design and Margin for Embedded SRAM

Robert Aitken and Sachin Idgunji
ARM, Sunnyvale CA, USA
{rob.aitken, sachin.idgunji}@arm.com

Abstract

An important aspect of Design for Yield for embedded SRAM is identifying the expected worst case behavior in order to guarantee that sufficient design margin is present. Previously, this has involved multiple simulation corners and extreme test conditions. It is shown that statistical concerns and device variability now require a different approach, based on work in Extreme Value Theory. This method is used to develop a lower-bound for variability-related yield in memories.

1 Background and motivation

Device variability is becoming important in embedded memory design, and a fundamental question is how much margin is enough to ensure high quality and robust operation without over-constraining performance. For example, it is very unlikely that the “worst” bit cell is associated with the “worst” sense amplifier, making an absolute “worst-case” margin method overly conservative, but this assertion needs to be formalized and tested. Setting the margin places a lower bound on yield – devices whose parameters are no worse than the margin case will operate correctly. Devices on the other side are not checked, so the bound is not tight.

1.1 Design Margin and Embedded Memory

Embedded memory product designs are specified to meet operating standards across a range of defined variations that are derived from the expected operating variations and manufacturing variations. Common variables include process, voltage, temperature, threshold voltage (V_t), and offsets between matched sensitive circuits. The design space in which circuit operation is guaranteed is called the operating range. In order to be confident that a manufactured circuit will function properly across its operating range (where conditions will not be as ideal as they are in simulation), it is necessary to stress variations beyond the operating range. The difference between the design space defined by the stress variations and the operating variations is called “margin”.

When multiple copies of an object Z are present in a memory, it is important for margin purposes to cover the worst-case behavior of any instance of Z within the memory. For example, sense amplifier timing must accommodate the slowest bit cell (i.e. the cell with the lowest read current). The memory in turn must

accommodate the worst case combination of sense amp and bit cell variation, in the context of its own variation. The general issue is shown graphically in Figure 1.

For design attributes that vary statistically, the worst case value is itself covered by a distribution. Experimental evidence shows that read current has roughly a Gaussian (normal) distribution (see Figure 2, and also [6]). The mean of this distribution depends on global process parameters (inter-die variation, whether the chip is “fast” or “slow” relative to others), and on chip-level variation (intra-die variation; e.g. across-chip line-width variation). Margining is concerned with random (e.g. dopant) variation about this mean. Within each manufactured instance, individual bit cells can be treated as statistically independent of one another; that is, correlation between nearby cells, even adjacent cells, is minimal. Every manufactured memory instance will have a bit cell with lowest read current. After thousands of die have been produced, a distribution of “weakest cells” can be gathered. This distribution will have a mean, the “expected worst-case cell”, and a variance.

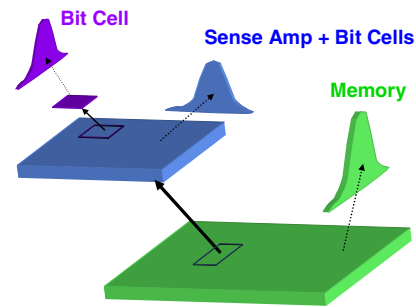


Figure 1 System of variation within a memory

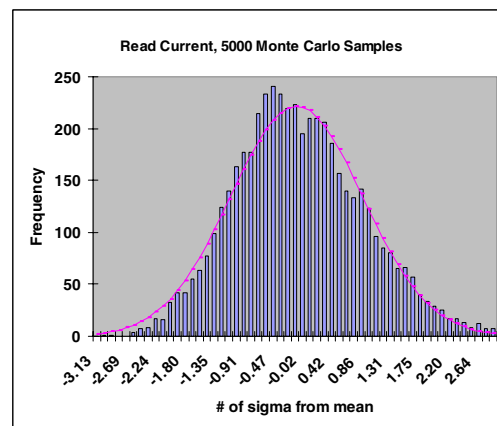


Figure 2 Read current variation, 65nm bit cell

This paper shows that setting effective margins for memories depends on carefully analyzing this

distribution of worst-case values. It will be shown that the distribution is not Gaussian, even if the underlying data are Gaussian, and that effective margin setting requires separate treatment of bit cells, sense amplifiers, and self-timing paths.

The remainder of the paper is organized as follows: Section 2 covers margin and yield. Section 3 introduces extreme value theory and shows its applicability to the margin problem. Section 4 develops a statistical margining technique for memories in a bottom-up fashion, and section 5 gives areas for further research and conclusions.

2 Margin and Yield

Increased design margin generally translates into improved product yield, but the relationship is complex. Yield is composed of three components: random (defect-related) yield, systematic (design-dependent) yield, and parametric (process-related) yield. Electrical design margin translates into a lower bound on systematic and parametric yield, but does not affect defect-related yield. Most layout margins (adjustments for manufacturability) translate into improved random yield, although some are systematic and some parametric.

Examples of previous work in the area include circuit improvements for statistical robustness (e.g. [1]), replacing margins through Monte Carlo simulation [2], and improved margin sampling methods [3]. In this work, we retain the concept of margins, and emphasize a statistically valid approach to bounding yield loss with a margin-based design method.

2.1 Margin statistics

Because of the challenges in precisely quantifying yield, it is more useful to think of improved design margins in terms of the probability of variation beyond the margin region. If variation of a design parameter Z can be estimated assuming a normal probability distribution, and the margin limit for Z can be estimated as S standard deviations from its mean value, then the probability of a random instance of a design falling beyond that margin can be calculated using the standard normal cumulative distribution function.

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-\frac{x^2}{2}} dx \quad (1)$$

If Z is a single-sided attribute of a circuit replicated N times, the probability that no instance of Z is beyond a given margin limit S can be estimated using binomial probability (assuming that each instance is independent, making locality of the margined effect an important parameter to consider):

$$P(N, S) = (1 - \Phi(S))^N \quad (2)$$

An attribute Z margined at 3 sigma is expected to be outside its margin region in 0.135% of the manufactured instances. If there is 1 copy of Z per memory, the probability that it will be within margin is $1 - 0.00135$ or 99.865%. If there are 2 copies of Z , the probability drops to $(0.99865)^2$ or 99.73%. And so it goes: If there are 10 copies of Z per memory, the probability that all 10 will be within margin is $(0.99865)^{10}$ or 98.7%. With 100 copies this drops to 87.3% and with 1000 copies to 25.9%, and so on.

Intuitively it is clear that increasing the number of samples decreases the probability that all of the samples lie within a certain margin range. The inverse cumulative distribution function, $\Phi^{-1}(Prob)$, or quantile function, is useful for formalizing worst case calculations. There is no elementary primitive for this function, but a variety of numerical approximations for it exist (e.g. NORMSINV in Microsoft Excel). For example, $\Phi^{-1}(0.99) = 2.33$, so an attribute margined at 2.33 sigma has a 99% probability of being within its margin range. We can use equation (3) to calculate $R(N, p)$, which is the margin value needed for N copies of an attribute in order to ensure that the probability of all of the N being within the margin value is p .

$$R(N, p) = \Phi^{-1} \left(p^{\frac{1}{N}} \right) \quad (3)$$

For example, setting $N=1024$ and $p=0.75$ gives $R(N, p)$ of 3.45. Thus, an attribute that occurs 1024 times in a memory and which has been margined to 3.45 sigma has 75% probability that all 1024 copies within a given manufactured memory are within the margin range (and thus a 25% probability that at least one of the copies is outside the margin range). For $p=0.999$ (a more reasonable margin value than 75%), $R(N, p)$ is 4.76.

2.2 Application to memory margining

Consider a subsystem $M0$ within a memory consisting of a single sense amp and N bit cells. Suppose each bit cell has Gaussian-distributed mean read current μ and standard deviation σ . Each manufactured instance of $M0$ will include a bit cell that is weakest for that subsystem, in this case the bit cell with lowest read current. By (3), calculating $R(N, 0.5)$ gives the median value for this worst case value. For $N=1024$, this median value is 3.2. This means that half of manufactured instances of $M0$ will have a worst case cell at least 3.2 standard deviations below the mean, while half will have their worst case cell less than 3.2 standard deviations from the mean. If a full memory contains 128 copies of $M0$ (we will call this system

M1), we would expect 64 in each category for a typical instance of M1 (although clearly the actual values would vary for any given example of M1).

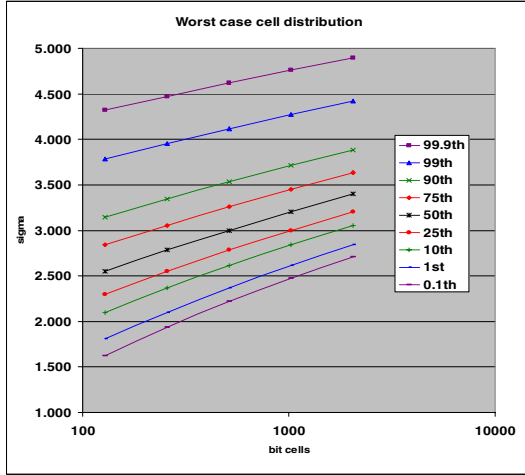


Figure 3 Graph of $R(N,p)$ for various p

Figure 3 shows various values for $R(N,p)$ plotted together. For example, the 50th percentile curve moves from 2.5 sigma with 128 cells to 3.40 sigma at 2048 cells. Reading vertically gives the value for a fixed N . For 512 cells, the 0.1th percentile is at 2.21 sigma, and the 99.9th percentile is at 4.62 sigma. Three observations are clear from the curves: First, increasing N results in higher values for the worst case distributions – consistent with the expectation that more cells should lead to a worst case value farther from the mean. Second, the distribution for a given N is skewed to the right – the 99th percentile is much further from the 50th percentile than the 1st, meaning that extremely bad worst case values are more likely than values close to the population mean μ . Finally, the distribution becomes tighter with larger N .

It turns out that the distributions implied by Figure 3 are known to statisticians, and in fact have their own branch of study devoted to them. The next section gives details.

3 Extreme Value Theory

There are many areas where it is desirable to understand the expected behavior of a worst-case event. In flood control, for example, it is useful to be able to predict a “100 year flood”, or the worst expected flood in 100 years. Other weather-related areas are monthly rainfall, temperature extremes, wind, and so on. In the insurance industry, it is useful to estimate worst-case payouts in the event of a disaster, and in finance it is helpful to predict worst-case random fluctuations in stock price. In each case, limited data is used to predict worst-case events.

There is a significant body of work in extreme value theory that can be drawn upon for memory margins. Some of the early work was conducted by Gumbel [4], who showed that it was possible to characterize the expected behavior of the maximum or minimum value of a sample from a continuous, invertible distribution. For Gaussian data, the resulting Gumbel distribution of the worst-case value of a sample is given by

$$G(x) = \frac{1}{s} \exp \left[\frac{-(x-u)}{s} - e^{\frac{-(x-u)}{s}} \right] \quad (4)$$

where u is the mode of the distribution and s is a scale

$$E(G(X)) = u + \gamma \cdot s \quad (5)$$

factor. The mean of the distribution is

$$stdev(G(X)) = \frac{\pi \cdot s}{\sqrt{6}} \quad (6)$$

(γ is Euler’s constant), and its standard deviation is

Below are some sample curves. It can be seen that varying u moves the peak of the distribution, and that s controls the width of the distribution.

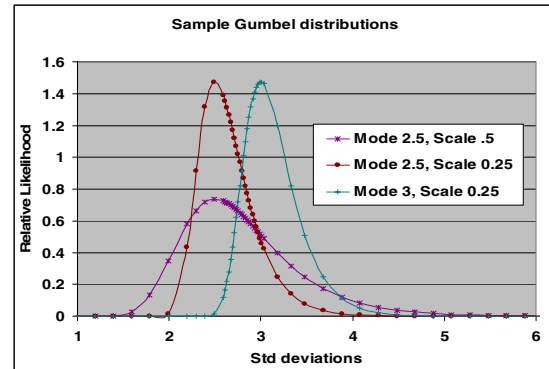


Figure 4 Sample Gumbel Distributions

4 Using Extreme Value Theory for Memory Margins

We consider this problem in three steps: First, we consider a key subsystem in a memory: the individual bit cells associated with a given sense amp. Next we generalize these results to the entire memory. Finally, we extend these results to multiple instances on a chip.

4.1 Subsystem results: Bit cells connected to a given sense amp

Figure 3 shows the distribution of worst-case bit cell read current values. Note that the distribution shown in the figure is very similar to the Gumbel

curves in Figure 4 – right skewed, variable width. In fact, the work of Gumbel [4] shows that this must be so, given the assumption that the underlying read current data is Gaussian. This is demonstrated by Figure 5, which shows the worst case value (measured in standard deviations from the mean) taken from 2048 randomly generated instances of 4096 bit cells.

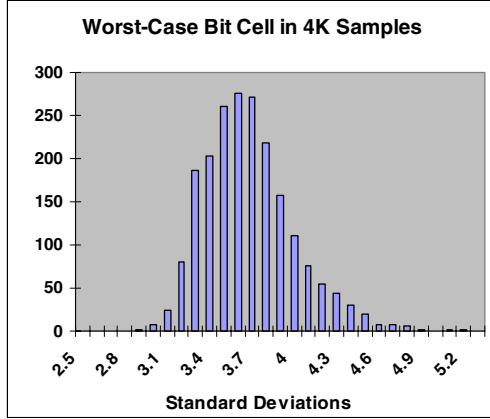


Figure 5 2048 instances of the worst case bit cell among 4096 random trials

Given this, we may now attempt to identify the correct Gumbel parameters for a given memory setup, using the procedure below, applied to a system M0 consisting of a single sense amp and N bit cells, where each bit cell has mean read current μ and standard deviation σ . Note that from equation (4) the cumulative distribution function of the Gumbel distribution is given by

$$CDFG(y) = \int_{-\infty}^y \frac{1}{s} \exp \left[\frac{-(x-u)}{s} - e^{\frac{-(x-u)}{s}} \right] dx \quad (7)$$

1. Begin with an estimate for a Gumbel distribution that matches the appropriate distribution shown in Figure 2 – the value for u should be slightly less than the 50th percentile point, and a good starting point for s is 0.25 (see figure 4).
2. Given an estimated u and s , a range of CDFG(y) values can be calculated. We use numerical integration with a tight distribution around the mode value u and a more relaxed distribution elsewhere. For example, with $u=3.22$ and $s=0.25$, the CDFG calculated for $y=3.40$ is 0.710.
3. Equation (3) allows us to calculate the cumulative distribution function R for a given probability p and sample number N . Thus, for each of the CDFG values arrived at in step 2, it is possible to calculate R . Continuing the previous example, $R(1024,0.710)=3.402$.

4. Repeat steps 2-3, adjusting u and s as necessary in order to minimize the least squares deviation of y and $R(N, CDFG(y))$ across the selected set y .

Using this approach results in the following Gumbel values for N of interest in margining the combination of sense amp and bit cells:

N	128	256	512	1024	2048	4096
mode	2.5	2.7	2.95	3.22	3.4	3.57
scale	0.28	0.28	0.27	0.25	0.23	0.21

Table 1 Gumbel parameters for sample size

These values produce the Gumbel distributions shown in the figure below.

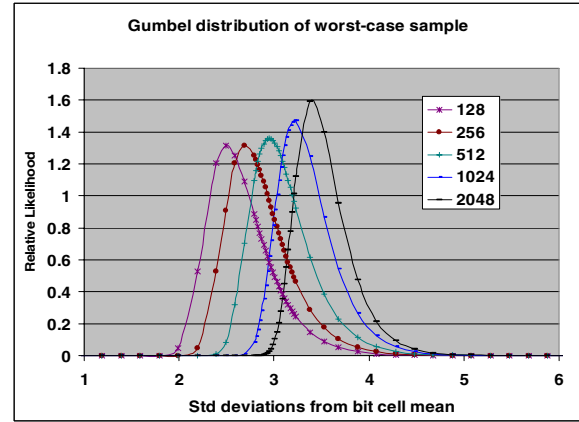


Figure 6 Gumbel distributions for sample sizes

Recall that these distributions show the probability distribution for the worst-case bit cell for sample sizes ranging from 128 to 2048. From these curves, it is quite clear that margining the bit cell and sense amp subsystem (M0) to 3 sigma of bit cell current variation is unacceptable for any of the memory configurations. Five sigma, on the other hand, appears to cover essentially all of the distributions. This covers M0, but now needs to be extended to a full memory.

4.2 Generalizing to a full memory

Suppose that a 512Kbit memory (system M1) consists of 512 copies of a subsystem (M0) of one sense amp and 1024 bit cells. How should this be margined? Intuitively, the bit cells should have more influence than the sense amps, since there are more of them, but how can this be quantified?

There are various possible approaches, but we will consider two representative methods: isolated and merged.

Method 1 (isolated):

In this approach, choose a high value for the CDF of the subsystem M0 (e.g. set up the sense amp to tolerate the worst case bit cell with probability

99.99%) and treat this as independent from system M1.

Method 2 (merged):

Attempt to merge effects of the subsystem M0 into the larger M1 by numerically combining the distributions; i.e. finding the worst case combination of bit cell and sense amp within the system, being aware that the worst case bit cell within the entire memory is unlikely to be associated with the worst case sense amp.

Each method has some associated issues. For method 1, there will be many “outside of margin” cases (e.g. a bit cell just outside the 99.99% threshold with a better than average sense amp) where it cannot be proven that the system will pass, but where in all likelihood it will. Thus the margin method will be unduly pessimistic. On the other hand, combining the distributions also forces a combined margin technique (e.g. a sense amp offset to compensate for bit cell variation) which requires that the relative contributions of both effects to be calculated prior to setting up the margin check. We have chosen to use Method 2 for this work, for reasons that will become clear shortly.

Before continuing, it is worthwhile to point out some general issues with extreme value theory: Attempting to predict behavior of tails of distributions, where there is always a lack of data, is inherently challenging and subject to error. In particular, the underlying Gaussian assumptions about sample distributions (e.g. central limit theorem) are known to apply only weakly in the tails. Because of the slightly non-Gaussian nature of read current, it is necessary to make some slight adjustments when calculating its extreme values of read current (directly using sample sigma underestimates actual read current for sigma values beyond about -3). There is a branch of extreme value theory devoted to this subject (see for example [5]), but this is beyond the scope of this work.

Consider systems M1 and M0 as described above. We are concerned with local variation within these systems (mainly implant), and, as with Monte Carlo SPICE simulation, we can assume that variations in bit cells and sense amps are independent. Since there are 1024 bit cells in M0, we can use a Gumbel distribution with $x=3.22$ and $s=0.25$ (see Table 2). Similarly, with 512 sense amps in system M1, we can use a Gumbel distribution with $x=2.95$ and $s=0.27$. We can plot the combined distribution in 3 dimensions as shown in Figure 6. The skewed nature of both distributions can be seen, as well as the combined probability, which shows that the most likely worst case combination is, as expected, a sense amp 2.95 sigma from the mean together with a bit cell 3.22 sigma from the mean.

Margin methodologies can be considered graphically by looking on Figure 7 from above. Method 1 consists of drawing two lines, one for bit cells and one for sense amps, cutting the space into quadrants, as shown in Figure 8, set to (for example) 5 sigma for both bit cells and sense amps. Only the lower left quadrant is guaranteed to be covered by the margin methodology; these designs have both sense amp and bit cell within the margined range. Various approaches to method 2 are possible, but one involves converting bit cell current variation to a sense amp voltage offset via Monte Carlo simulation. For example, if expected bit line voltage separation when the sense amp is latched is 100mV and Monte Carlo simulation shows that a 3 sigma bit cell produces only 70mV of separation, then 1 sigma of bit cell read current variation is equivalent to 10mV of bit line differential. This method is represented graphically by an angled line, as shown in Figure 9. Regions to the left of the line are combinations of bit cell and sense amp within margin range.

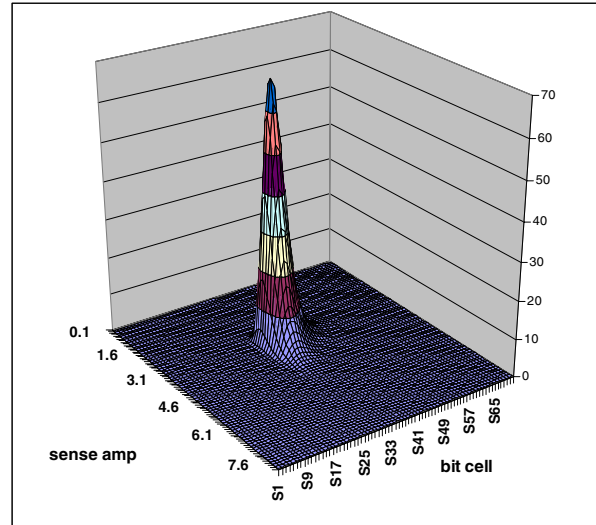


Figure 7 Combined Gumbel Distribution

Using numerical integration methods, the relative coverage of the margin approaches can be compared. Method 1 covers 99.8% of the total distribution (roughly 0.1% in each of the top left and lower right quadrants). Method 2 covers 99.9% of the total distribution, and is easier for a design to meet, because it passes closer to the peak of the distribution (a margin line that passes through the intersection of the two lines in Figure 8 requires an additional 0.5 sigma of bit cell offset in this case. For this reason, we have adopted Method 2 for margining at 65nm and beyond.

A complete memory, as in Figure 1, has an additional layer of complexity beyond what has been considered so far (e.g. self-timing path, clocking circuitry), which we denote as system M2. There is only one copy of M2, so a Gaussian distribution will

suffice for it. It is difficult to represent this effect graphically, but it too can be compensated for by adding offset to a sense amplifier.

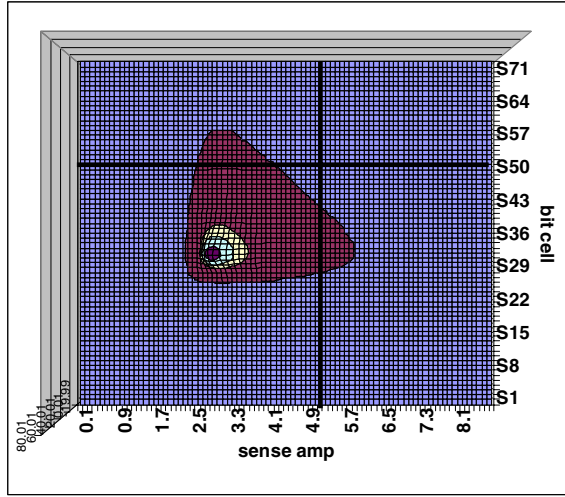


Figure 8 Margin Method 1

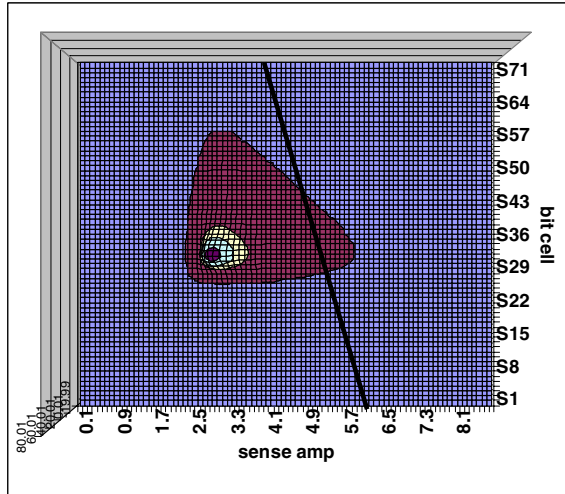


Figure 9 Margin Method 2

Extension to repairable memory: Note that the existence of repairable elements does not alter these calculations much. System M0 cannot in most architectures be improved by repair, but system M1 can. However, repair is primarily dedicated to defects, so the number of repairs that can be allocated to out-of-margin design is usually small. We are currently developing the mathematics for repair and will submit it for publication in the future.

4.3 Generalization to multiple memories

Since failure of a single memory can cause the entire chip to fail, the margin calculation is quite straightforward.

$$P_{margin}(\{m\}) = \prod_{all\ m} MAR(m) \quad (8)$$

$$Y_{var} \geq P_{margin}(\{m\}) \quad (9)$$

where each $MAR(m)$ is the probability that a given memory m is within margin. For a chip with 1 copy of the memory in the previous section, this probability is 99.9% under Method 2. This drops to 95% for 50 memories and to 37% for 1000 memories. The latter two numbers are significant. If 50 memories is typical, then this approach leads to a margin method that will cover all memories on a typical chip 95% of the time, so variability related yield will be at least 95%. This is not ideal, but perhaps adequate. On the other hand, if 1000 memories is typical, the majority of chips will contain at least one memory that is outside its margin range, and the lower bound on variability related yield is only 37%. This is clearly unacceptable. To achieve a 99% lower bound for yield on 1000 memories, each individual yield must be 99.999%, which is outside the shaded area in Figure 9. For this reason, it is vital to consider the entire chip when developing a margin method.

5 Conclusions and future work

We have shown a margin method based on extreme value theory that is able to accommodate multiple distributions of bit cell variation, sense amp variation, and memory variation across chip. We are currently extending the method to multi-port memories and to include redundancy and repair. These will be reported in future works.

6 References

- [1] B. Agrawal, F. Liu & S. Nassif, "Circuit Optimization Using Scale Based Sensitivities", *Proc Custom Int. Circ. Conf.*, pp. 635-638, 2006.
- [2] S. Mukhopadhyay, H. Mahmoodi, & K. Roy, "Statistical design and optimization of SRAM cell for yield enhancement", *Proc. ICCAD*, pp. 10-13, 2004.
- [3] R.N.. Kanj, R.V. Joshi, S.R. Nassif, "Mixture Importance Sampling and Its Application to the Analysis of SRAM Designs in the Presence of Rare Failure Events", *Proc. Design Aut. Conf.*, 5-3, 2006.
- [4] E.J. Gumbel, *Les valeurs extrêmes des distributions statistiques*, Ann. Inst. H. Poincaré, Vol 5, 115-158, 1935. (available online at numdam.org)
- [5] E. Castillo, *Extreme Value Theory in Engineering*, Academic Press, 1988.
- [6] D-M Kwai et al, "Detection of SRAM Cell Stability by Lowering Array Supply Voltage", *Proc. Asian Test Symp.*, pp. 268-271, 2000