Circuit and CAD Techniques for Expanding the SRAM Design Space

---

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy (Electrical Engineering)

by

Jim Boley

May 2015

Abstract

As mobile devices become heavily energy constrained, the need for low power, energy efficient circuits has emerged. The application space varies from ultra low power devices such as body sensor networks (BSNs), to higher performance applications such as smart phones, tablets, and all other devices constrained by battery life. In order to reduce energy consumption and increase energy efficiency, voltage supplies are scaled down to take advantage of quadratic active energy savings. Static random access memory (SRAM) is a critical component in modern system on chips (SoCs), consuming large amounts of area and often on the critical timing path. SRAM is the most commonly used form of memory in cache designs due to its high speed and high density. In the past, conventional SRAM designs were able to take advantage of Moores law by simply reducing devices sizes and scaling down V_{DD}. This has become increasingly difficult as devices enter the nanoscale range due to increased device variability and leakage. SRAM devices are typically minimum sized, which further compounds this problem. The increase in both variation and leakage leads to reduced read and write margins, making it more difficult to design low power SRAMs that meet frequency and yield constraints. In addition, as the capacity of SRAM arrays continues to increase, the stability of the worst case bitcell degrades. Therefore it has become increasingly important to evaluate the effect of V_{DD} reduction on SRAM yield and performance.

The goal of this work is to push the memory design space beyond its conventional bounds. Typically the minimum supply voltage (V_{MIN}) of SRAMs is higher than that of conventional CMOS logic due to a higher sensitivity to device variation. In order to push SRAM designs

past this apparent brick wall, new knobs have been introduced such as alternative bitcells and read and write assist methods which improve the robustness of SRAMs in the presence of variability. These knobs introduce new trade-offs between energy, speed, area and yield which are difficult to evaluate because they are dependent on many factors such as technology node, bitcell architecture, and design constraints.

In this work, we first investigate the trade-offs in designing a subthreshold SRAM embedded in an ultra low power body sensor network. The result of this work is one of the first embedded subthreshold memories, capable of operation down to 0.35 volts. Next, we present a method for fast, accurate estimation of SRAM dynamic write V_{MIN}, which we will show provides a speedup of 112X over statistical blockade at a cost of only 3% average error. Furthermore, we will evaluate the combination of new bitcell circuit topologies and circuit assist methods at reducing SRAM read and write V_{MIN}. Next, we extend the functionality of an existing tool used for rapid design space exploration and optimization of SRAMs. The proposed extensions include: evaluation of read and write assist methods, support of multi-bank design evaluation, and yield evaluation. To combat the effects of process, voltage, and temperature (PVT) variation, we propose a tracking method using canary cells to regain energy lost through over-conservative guard-banding. Finally, we present a set of novel stack-based sense amplifier designs for reducing input-referred offset. The anticipated contribution of this research is a set of circuit methods and tools for pushing SRAM designs to lower operating voltages, increasing yields, and evaluating design trade-offs.

Approval Sheet

This dissertation is submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Electrical Engineering)

Jim Boley

Jim Boley

This dissertation has been read and approved by the Examining Committee:

Benton Calhoun

Benton Calhoun, Advisor

Joanne Dugan

Joanne Dugan, Committee Chair

Donald Brown

Donald Brown

John Lach

John Lach

Ronald Williams

Ronald Williams

Accepted for the School of Engineering and Applied Science:

James H. Aylor

James H. Aylor, Dean, School of Engineering and Applied Science

May 2015

Don't take life too seriously. You'll never get out alive.
-Van Wilder

Acknowledgements

Over the course of the past eight years, the University of Virginia has become home to me. The friends I've made and the memories we've shared will stick with me for the rest of my life. I've had the pleasure of meeting and working with some truly amazing people, and I give them all the credit for where I am today.

I would first like to thank my adviser, mentor, and friend Professor Ben Calhoun. His passion for both research and teaching are what inspired me to come to graduate school. The energy and excitement that he brings to work everyday is truly inspiring to the students he works with. Throughout the course of graduate school, Ben's high expectations have challenged me to become a better researcher and to hold myself to a higher standard. As a result it has made me a better person. Thanks for everything.

I'd like to thank all of my committee members: Professor Donald Brown, Professor Joanne Dugan, Professor John Lach and Professor Ron Williams for their time, advice, and insight throughout this process and for putting up with my obscenely long doodle polls.

I was very privileged to have worked closely with Dr. Randy Mann, Dr. Satya Nalam, Sudhanshu Khanna, Peter Beshay, Arijit Banerjee, Farah Yahya, Harsh Patel, and Ningxi Liu of the Bengroup SRAM team. Our conversations spurred many of the ideas presented in this thesis and I'm grateful for the insights, lessons, and knowledge you have imparted on me. I'd like to give a special thanks to Dr. Randy Mann for being extremely patient with me when I first entered grad school. Your knowledge and willingness to teach have been invaluable to my education.

A major factor in the positive experience that I had in grad school was the dynamic that I enjoyed in Bengroup. From my very first tapeout in the summer of 2006 with Yanqing Zhang, Yousef Shaksheer, Alicia Klinefelter, and Aatmesh Shrivastava I knew that even if the most important thing I contributed was late night Wendy's Frosties and a leaky sub-threshold memory that I was going to at least enjoy working together. As part of the BSN team, I spent most of my time making memories (pun intended) with Yousef and Alicia. They have been incredibly helpful and easy to work with and I look forward to continuing to work together after graduation. Outside of work I spent most of my time decompressing with a beer or Mario Kart (sometimes both) with Kyle Craig, Seyi Ayorinde, and Chris Lukas. Our early morning trips to the AFC (and ensuing arguments about "plate math") were often the highlight of my day. Overall Bengroup has been a blast and I want to thank you all for contributing positively to my experience: Dr. Jiajing Wang, Dr. Randy Mann, Dr. Satya Nalam, Sudhanshu Khanna, Joe Ryan, Yousef Shaksheer, Yanqing Zhang, Aatmesh Shrivastiva, Kyle Craig, Peter Beshay, Patricia Gonzalez, Divya Akella, Yu Huang, He Qi, Arijit Banerjee, Abhishek Roy, Farah Yahya, Chris Lukas, Harsh Patel, Ningxi Liu, and Terry Tigner.

I would like to thank ARM Inc. and my manager Vikas Chandra for the opportunity to intern during the summers of 2011 and 2012. My conversations with Vikas over lunch that first summer persuaded me to stick with the PhD program and I can't thank you enough for your guidance and advice.

Throughout my undergrad and graduate career I have been heavily involved in the ultimate frisbee community. I consider the friends that I have made through this sport to be family and I could not of made it this far without their support.

Last but not least I would like to thank my family. Your unwavering love and support throughout my time at UVA has meant the world to me and I want to thank you especially for always believing in me throughout this journey.

Contents

List of Tables

List of Figures

List of Acronyms

ASIC application specific integrated circuit

BIST built-in self test

BL bitline

BLB bitline bar

BSN body sensor network

CDF cumulative distribution function

CE characterization engine

CMOS complimentary metal oxide semiconductor

DRV data retention voltage

DMA direct memory access

DMEM data memory

ECC error correction code

FF fast-NMOS, fast-PMOS process corner

FIFO first in, first out

FoMs figures of merit

FS fast-NMOS, slow-PMOS process corner

IMEM instruction memory

IoT internet of things

I_{READ} average read current

ITRS Internation Roadmap for Semiconductors

LC level converter

MC Monte Carlo

MCU microcontroller

NMOS N-type metal oxide semiconductor

PDF probability density function

PMOS P-type metal oxide semiconductor

ROM read-only memory

SA sense amp

SER soft error rate

SF slow-NMOS, fast-PMOS process corner

SNM static noise margin

SoC system on a chip

SRAM static random-access memory

SS slow-NMOS, slow-PMOS process corner

Sub-V_T sub-threshold

T_{CRIT} critical wordline pulse width

TT typical-NMOS, typical-PMOS process corner

ULP ultra low power

V_{DD} supply voltage

V_{DS} drain to source voltage

V_{GS} gate to source voltage

ViPro virtual prototyper

VLSI very large scale integration

V_{MIN} minimum operation voltage at which memory can read and write at a target yield

V_T threshold voltage

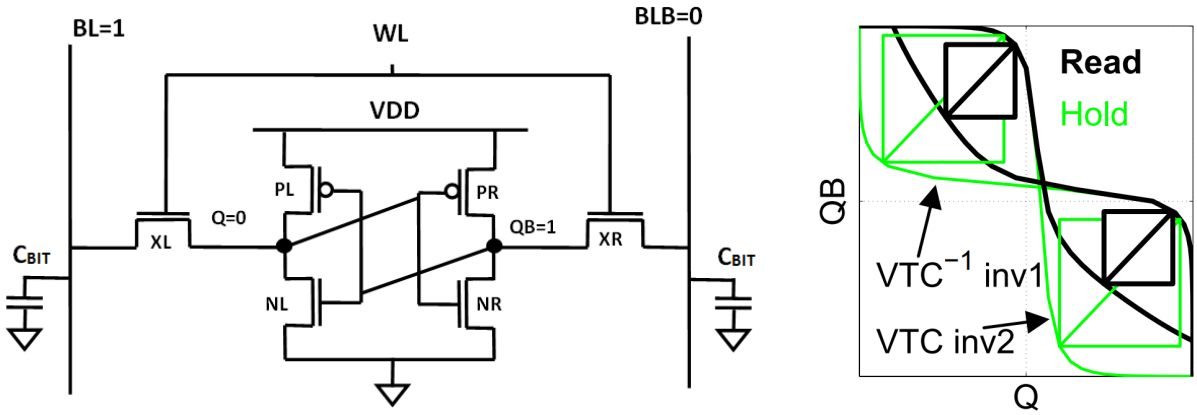
WL wordline

WM write margin

Chapter 1

Introduction

The miniaturization of devices as predicted by Moore's Law has resulted in an ever increasing role of electronics in our everyday lives. The reduction in size has enabled a variety of platforms ranging from high performance supercomputers down to highly mobile hand held devices. While technology scaling has led to an increase in mobility, it has also presented new design challenges due to increases in variability, leakage, and design complexity. Static random access memory (SRAM) is highly susceptible to these challenges due to its sub-minimum sized devices and ratioed design (Figure 1.1a). SRAM is the most commonly used form of embedded memory due to its high speed and density. It is a critical component in modern system on chips (SoCs), consuming up to 90% [10] of the total area on chip and is often on the critical timing path. Because of its large area, SRAM consumes a significant amount of power in the form of leakage. The most commonly employed method for reducing leakage is voltage scaling. In the past, the operating voltage of these memories has been easily scaled down with technology; however as devices sizes have entered the nanometer regime, voltage scaling has hit a fundamental wall due to reduced reliability. In order to break through this wall and continue scaling, SRAM designers must address and overcome the challenges outlined in the rest of this chapter.



(a) (b)

Figure 1.1: (a) Schematic of the conventional 6T SRAM bitcell (b) the length of the side of the largest square that can be fit inside the butterfly curve represents the static noise margin of the cell [1]

1.1 Reducing SRAM V_{MIN}

In order to increase energy efficiency, voltage supplies are scaled down to take advantage of quadratic energy savings ($E = C * V_{DD}^2$). In addition to reducing active energy, reducing V_{DD} also reduces leakage energy. This is especially important for SRAMs due to the fact that memories can contain millions of cells, the majority of which are held in standby mode (e.g. only consuming static current). The minimum operation voltage (V_{MIN}) is defined as the minimum voltage that the SRAM can operate without failures. The three main categories of SRAM failures are: read, write, and hold failures. These categories will be discussed in detail in the following subsections.

1.1.1 Read Static Noise Margin

The static noise margin is typically calculated using the butterfly curve technique (Figure 1.1b) first introduced by [11]. This metric is a measure of the amount of noise that a bitcell can tolerate before its data becomes corrupted. During a read operation, both of the bitlines are precharged high, and are held dynamically at V_{DD} (Figure 1.1a). Once the wordline (WL) pulses high (Figure 1.2b), the charge stored on the BL is discharged through XL and

NL. Because the bitline is shared with many cells (up to 512), the value of C_{BIT} is very large. This can cause the node at Q to rise above ground. In order to ensure that the voltage at this node does not rise above the switching threshold of the PR/NR inverter, the resistance of the XL transistor must be kept larger than that of the NL transistor. If the voltage rises above the threshold value of NR, this could result in the data being stored to flip values. This is prevented by sizing the pull-down and passgate using the following equations [12]:

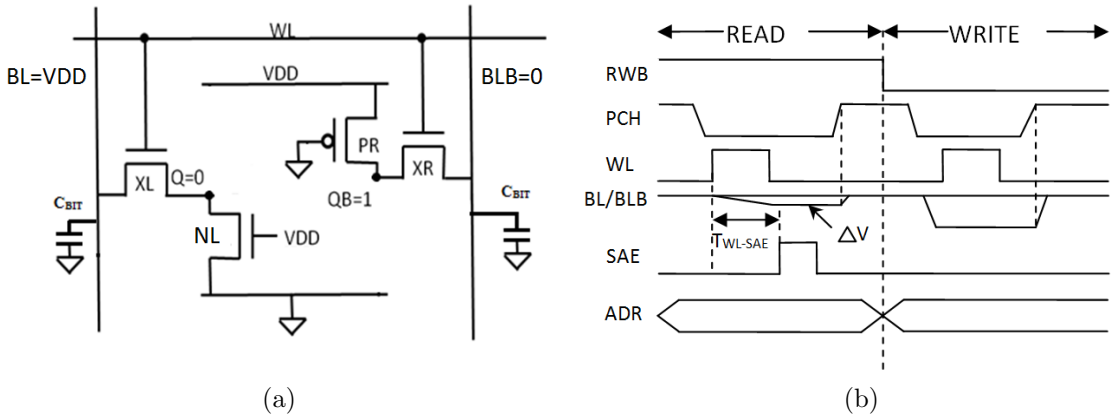
$$k_{n,XL}\left[\left(V_{DD}-\Delta V-V_{Tn}\right)V_{DSATn}-\frac{V_{DSATn}^2}{2}\right]=k_{n,NL}\left(V_{DD}-V_{Tn}\right)\Delta V-\frac{\Delta V^2}{2} \tag{1.1}$$

$$\Delta V=\frac{V_{DSATn}+CR(V_{DD}-V_{Tn}-\sqrt{V_{DSATn}^2(1+CR)+CR^2(V_{DD}-V_{Tn})^2}}{CR} \tag{1.2}$$

$$CR=\frac{W_{NL}/L_{NL}}{W_{XL}/L_{XL}} \tag{1.3}$$

As an example, if the threshold voltage of the NMOS transistor is 0.4 volts, than the cell ratio (CR) must be kept above 1.2 in order to ensure that the voltage of the Q node (ΔV) does not rise high enough to turn on the NR transistor. By sizing these devices properly, we can ensure that the bitcell remains stable during a read. However, as we can see from these equations, variation in the threshold voltage could cause the bitcell to become unstable. This type of ratioed design becomes even more unreliable in the sub-threshold region where the on current becomes exponentially dependent on V_T (equation 1.4). This exponential dependence of I_{SUBVT} on V_T is one of the biggest challenges in designing in the sub-threshold region.

$$I_{SUBVT}=I_0\frac{W}{L}exp\left(\frac{V_{GS}-V_T-\eta V_{DS}}{\eta V_{th}}\right)\left(1-exp\left(-\frac{V_{DS}}{V_{th}}\right)\right) \tag{1.4}$$



(a) (b)

Figure 1.2: (a) Equivalent circuit of the conventional 6T SRAM bitcell during a write operation (b) a typical SRAM read and write timing diagram

1.1.2 Write-Ability

During a write (Figure 1.2a and 1.2b), the bitlines are driven statically to V_{DD} and ground. In this example we are writing a 1 into the cell. Because we have sized the XL/NL ratio such that the Q node cannot rise high enough to flip the cell, the new value must be written in by pulling the QB node to ground. Once again we have a ratioed fight occurring, this time between the XR and PR transistors. In order to write a 0 into the bitcell, the QB node must be pulled low enough to turn on the PL transistor. Using a similar approach as in the previous section, we can set the currents of these two transistors equal to determine the minimum sizing of the pull up to pull down ratio to pull QB low enough to flip the cell [12]:

$$k_{n,XR}\left[\left(V_{DD}-V_{Tn}\right)V_Q-\frac{V_Q^2}{2}\right]=k_{n,PR}\left[\left(V_{DD}-|V_{Tp}|\right)V_{DSATp}-\frac{{V_{DSATp}}^2}{2}\right] \tag{1.5}$$

$$V_Q=V_{DD}-V_{Tn}-\sqrt{\left(V_{DD}-V_{Tn}\right)^2-2\frac{\mu_p}{\mu_n}PR\left[\left(V_{DD}-|V_{Tp}|\right)V_{DSATp}-\frac{V_{DSATp}^2}{2}\right]} \tag{1.6}$$

$$PR = \frac{W_{PR}/L_{PR}}{W_{XR}/L_{XR}} \tag{1.7}$$

What we find is that the pull up device should typically be kept minimum sized in order to improve write-ability. The downside to this is that the variability of this device will be larger due to the fact that it is minimum sized. The passgate is typically up sized to further reduce this ratio, and increase write-ability. The trade-off is that strengthening the passgate also reduces read stability, therefore the two metrics must be balanced in order to maintain functionality. As with read-stability, write-ability is reduced in sub-threshold due to the exponential dependence of the on current to threshold voltage variations.

To measure the static write margin, the bitcell is first set into a known state (holding '0' or holding '1'), and the BLs are driven to the opposite value. The WL is then swept from 0 to V_{DD}. The margin is defined as $V_{DD} - V_{WL}$, where V_{WL} is defined as the wordline voltage when the internal nodes flip [13]. A large positive margin means that the cell is easy to write, while a margin of ≤ 0 is equivalent to a static write failure. The downside to using this static metric for write margin is that it assumes an infinite WL pulse width and is therefore more optimistic compared to a dynamic metric. In addition, it does not take into account the transient behavior of the bitcell. A more accurate metric for measuring write ability is to measure the minimum WL pulse width required to flip the cell, known as T_{CRIT}. This metric will be discussed in detail in chapter 4.

1.1.3 Read Access Stability

Read access fails occur when the bitline differential developed before the sense amp enable (SAE) signal goes high is not large enough for the sense amp to correctly resolve to the correct value (Figure 1.3). This occurs due to variation in both the maximum current being sunk by the bitcell during a read (I_{READ}), and the sense amp offset voltage due to variation within the sense amp (V_{OS} or σ_{OFFSET}). I_{READ} sets the delay for the proper BL differential

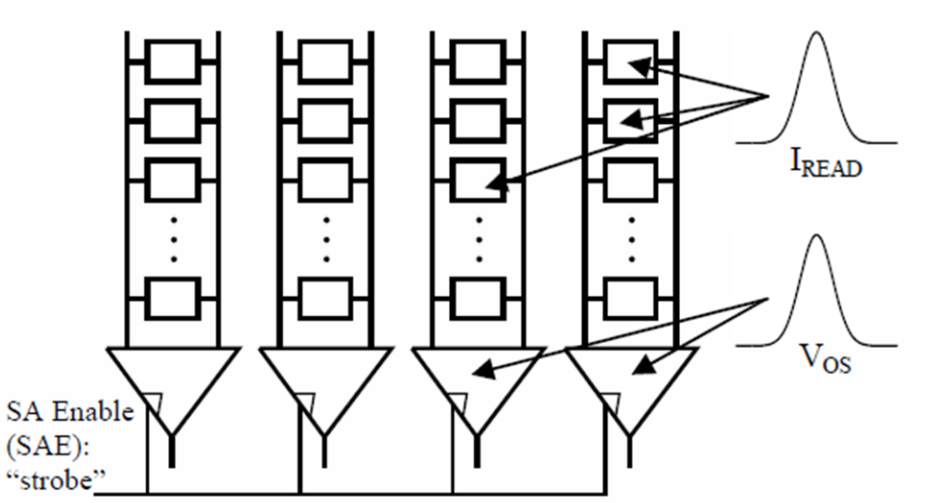


Figure 1.3: Read access fails occur due to variation in the read current and the built-in sense amp offset [2]

to develop and is typically normally distributed. σ_{OFFSET} determines the minimum BL differential required in order for the sense amp to resolve to the proper value. The sense amp offset is also normally distributed and typically has a mean of 0 mV. A read access failure is usually considered a performance failure, because the read failed to complete within the cycle time. It has been shown in [2] that 55% of the total read delay occurs in the development of the BL differential. Therefore it is important to minimize the delay between the WL and SAE signal (T_{WL-SAE}) without compromising yield.

Worst case analysis sets the value of T_{WL-SAE} by pairing the worst case bitcell with the worst case sense amp. However it is noted in [2] that the probability of this occurring in a large memory is actually very small. By using this pessimistic approximation, we are sacrificing performance as well as energy. The increase in energy is due to the fact that the WL pulse width is larger than it needs to be, resulting in more charge being dissipated from the bitlines. [2] instead uses order statistics to determine the bitcell/sense amp pairing that results in the worst case T_{WL-SAE}, resulting in a 9300x speed up over Monte Carlo simulations. This model will be revisited in chapter 5 for evaluating the trade-off between yield, performance, and energy.

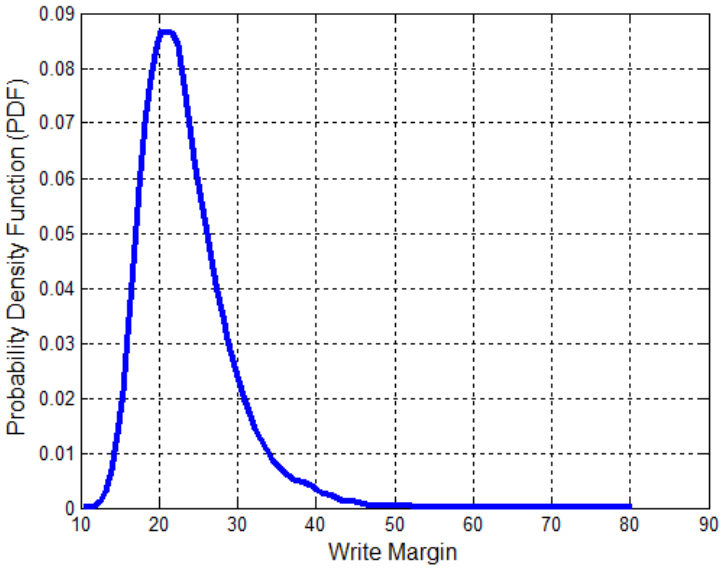


Figure 1.4: Curve fitting can lead to large errors if the data does not match a known distribution

1.2 Estimating Yield

Monte Carlo (MC) simulation is the gold standard for evaluating the effects of process variation on circuit performance and reliability. Because variation is a stochastic process, we use MC to calculate failure probabilities, but can not necessarily guarantee functionality. The difficulty with using MC for SRAMs is that memories can contain millions of bits; therefore the number of simulations needed for margining becomes prohibitively large. In addition, because we are only concerned about points lying in the tail region, Monte Carlo simulations are not efficient at identifying these points. A common approach to reducing simulation time is to run a relatively small number of samples and then fit the resulting distribution to the normal distribution. Once the μ and σ are known, the stability of the worst case bitcell can be identified. The problem with this approach is that it can only be applied to data sets that replicate a known distribution [14, 15]. As shown in Figure 1.4, not all data sets match a known distribution, which can lead to large errors in approximating the tail of the distribution. Therefore, we need some method for quickly and accurately estimating SRAM failure probabilities.

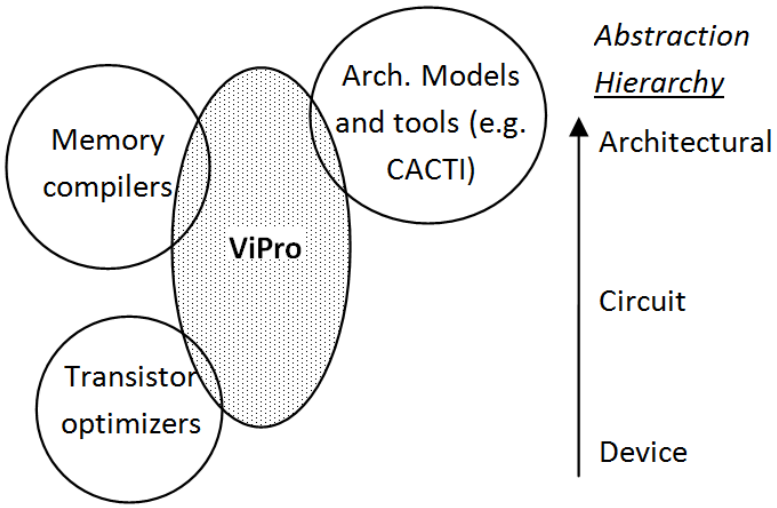


Figure 1.5: ViPro combines device, circuit, and architectural level models to generate optimal SRAM designs and evaluate the benefits of circuit innovations

1.3 Evaluating Design Decisions

The introduction of new circuit techniques such as read and write assist methods and new bitcell topologies creates a whole new set of trade-offs between speed, area, performance and reliability. These trade-offs are difficult to evaluate because they are dependent on many factors such as technology node, bitcell architecture, and design constraints. In addition, technology scaling has brought on a whole new set of challenges due to increases in memory capacity, process variation, interconnect delay, soft error susceptibility (SER), and leakage. Many circuit techniques have been proposed to address these challenges, however these solutions tend to address individual components. A change in any one of the key memory circuits or in the core cell technology will alter the optimal circuit topologies, partitioning, and architecture for the entire memory. We can no longer innovate in one portion of the memory while ignoring the effects our innovation could have on the overall memory and system design. Without the proper support structure and tools, it would be nearly impossible to re-design and re-optimize an entire memory by hand every time we try a new circuit, much less explore a technique's impact across different technologies and applications. Therefore there is a need for a tool flow which is capable of evaluating both circuit and architectural

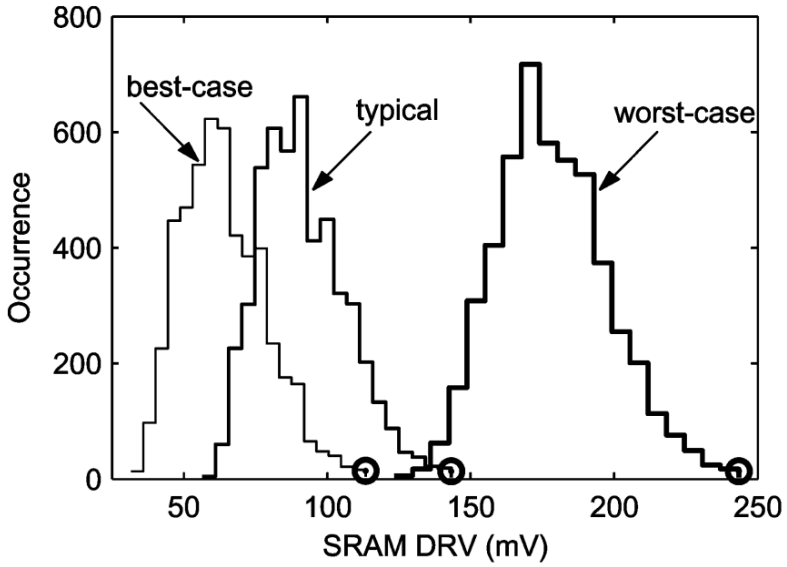


Figure 1.6: The worst case V_{MIN} is 100 mV higher than the average V_{MIN}, resulting in potential energy savings [3]

level design decisions at the system level. This is the goal of ViPro (Chapter 5): to combine the device, circuit, and architectural level models in order to generate optimal SRAM designs and evaluate the benefits of circuit innovations (Figure 1.5).

1.4 Adapating to Process, Voltage, and Temperature (PVT) Variations

One of the biggest challenges facing commercial SoC's is guaranteeing functionality across a wide range of process, voltage, and temperature variations. Designs are typically margined to ensure functionality under the worse case conditions, which typically means over-designing for the average case. Figure 1.6 shows the traditional method of guard-banding to consider the worst case scenario for setting the operating voltage at design time. This conservative approach ensures reliable operation across the worst PVT corners; however it also sacrifices potential energy savings because the full range of V_{MIN} is large when accounting for the worst case [16]. Because the circuit is not always operating in the worst case PVT corner, there is a

potential to regain some of this lost energy. If the circuit could adapt to changes in the PVT corner, instead of setting the operating voltage at design time, then the energy overhead of conservative guard-banding could be eliminated.

1.5 Dissertation Organization

This dissertation is organized as follows. Chapter 2 presents a novel asymmetric Schmitt Trigger (ST) bitcell which uses single ended reading to achieve higher read static noise margin (RSNM) compared to the 6T cell and the 10T ST bitcell [5] in simulation. The second contribution of this chapter is a comparison of different read and write assist methods and various sub-threshold bitcell topologies. We found that assist methods had a much stronger impact on reducing V_{MIN} compared to alternative bitcell topologies. In addition, we find that the bitcells proved to be write-limited in sub-threshold, a trend that has been shown to be especially true in newer technologies [17]. Using measurements from the test chip, we show which assist methods were more effective at reducing read and write V_{MIN}.

Chapter 3 presents a methodology for designing sub-threshold SRAM specifically for ultra low power body sensor networks (BSNs). In this chapter we highlight the challenges, and common pitfalls associated with ULP memory design. The final outcome of this chapter is a 2 kB and 4 kB sub-threshold SRAM embedded on an ultra low power body sensor SoC, capable of operation down to 0.35V.

In Chapter 4 we focus on modeling SRAM dynamic write V_{MIN}. We present a methodology based on sensitivity analysis that provides a total speedup compared with recursive staticsal blockade with only a small loss in accuracy. Using this method, we characterize SRAM dynamic V_{MIN} across a range of speed requirements and cache sizes. In addition, we compare the static write failure rates to the dynamic write failure probability predicted by the model to prove that the static metric is more optimistic and should not be used in high speed

applications. Finally, we use the model to compare a subset of write assist methods across a range of voltages.

Chapter 5 describes the development of Virtual Prototyping tool, ViPro. In this work, we extend the previous functionality of ViPro to consider the effect of V_T variation on the global figures of merit. Additionally, we enable the tool to consider die yield as a metric for evaluation.

Chapter 6 describes a canary based system for tracking process, voltage, and temperature variation in SRAM designs. We present a methodology for optimization using order statistics to maximize the energy savings of the system.

Chapter 7 focuses on reducing sense amp offset (σ_{OFFSET}) in order to provide savings in read energy and delay. The first proposed circuit uses a source coupled scheme which reduces σ_{OFFSET} and requires no area overhead. In addition, we present three novel sense amp designs which further reduce σ_{OFFSET} at iso-area compared to a traditional latch-based design.

Chapter 2

The Effects of Assist Methods on SRAM V_{MIN}

[1] As mobile devices become heavily energy constrained, the need for ultra low power circuits has emerged. In order to reduce energy consumption, voltage supplies are scaled down to take advantage of quadratic energy savings ($Energy = C * V_{DD}^2$). The sub-threshold region ($V_{DD} < V_T$) has been shown by [18] to minimize energy per operation. Sub-threshold systems require Static Random Access Memory for storing data at these low voltages. The problem with this is that while logic has been shown to easily scale into the sub-threshold region, the traditional 6T SRAM bitcell becomes unreliable at voltages below 700 mV due to process variations and decreased device drive strength [19]. SRAM devices are typically minimum sized, which further compounds this problem. As the capacity of SRAM arrays continues to increase, the stability (typically measured in terms of Static Noise Margin (SNM) [11]) of the worst case bitcell degrades. Therefore, in order for the minimum operating voltage (V_{MIN}) of SRAMs to enter the sub-threshold regime, more robust bitcell designs or assist methods must be used.

One possible solution to this problem is to design a more robust bitcell topology capable of

[1] This chapter is based on the published paper titled: Analyzing Sub-Threshold Bitcell Topologies and the Effects of Assist Methods on SRAM V_{MIN}" [JB2]

larger read and write margins. The downside to this strategy is that adding more transistors to the bitcell increases the total area of the array. The second strategy is to use various assist methods [20–29] to make the cell easier to read and write. This method also results in a smaller area overhead and may require multiple voltage sources. In this chapter we will analyze different bitcell topologies and assist methods to determine which is the most effective at reducing SRAM V_{MIN}.

2.1 Introduction of Sub-Threshold Bitcell Topologies

In a sub-threshold circuit, the supply voltage (V_{DD}) is set below the threshold voltage (V_T) of the transistors. This reduction in V_{DD} results in a quadratic reduction in switching power. In addition, it reduces leakage power, which is especially important for SRAMs that contain thousands or millions of bitcells. The main limitations of sub-threshold circuits are their sensitivity to variation and slow speed. In the sub-threshold region, transistor currents vary exponentially with V_T. This makes designing ratioed circuits such as SRAMs nearly impossible [30]. Another problem is that the I_{ON}/I_{OFF} current ratio is reduced, which can lead to read access failures on bitlines with excessive leakage. In order to combat these problems, new bitcell topologies have been introduced and are described below.

The 8T bitcell [4] shown in Figure 2.1 adds a two transistor read buffer to the conventional 6T bitcell in order to prevent the data from being disturbed during a read. In a normal read operation, the bitlines are precharged and the WL is pulsed high, causing the bitcell to discharge one of the bitlines. The problem with this is that if the node storing a 0 rises above the switching threshold of right inverter (Figure 2.1), then the cell could unintentionally flip. The 8T cell solves this problem by decoupling the data from the read operation; therefore the read SNM becomes the hold SNM. One weakness of this bitcell is that it still suffers from half-select instability, which occurs during a write when an unselected cell is read like a traditional 6T bitcell. Currently the best method to solve this problem in a bit interleaved

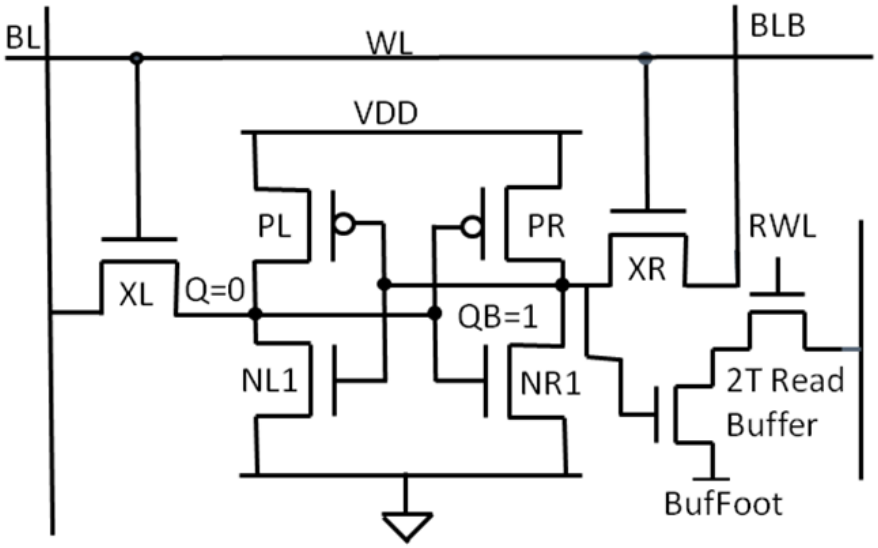


Figure 2.1: The 8T bitcell [4] introduces a two transistor read buffer which decouples the stored data from the read bitline during a read operation

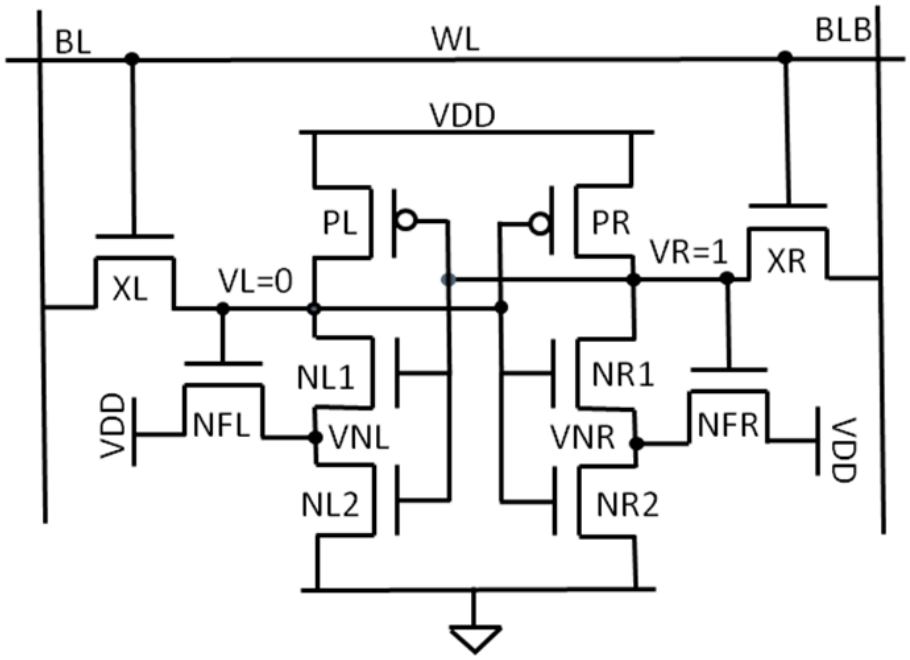


Figure 2.2: The 10T bitcell [5] uses Schmitt Trigger inverters to improve the stability of the cell during a read

architecture is by using a read before write scheme. In this method the entire row is read and then the data is written back into the unselected cells at the same time that new data is written to the selected cells.

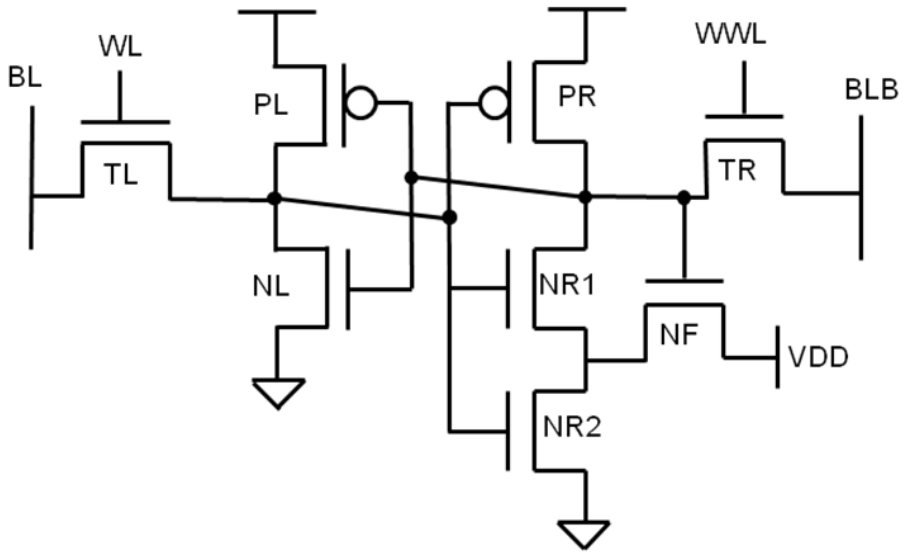
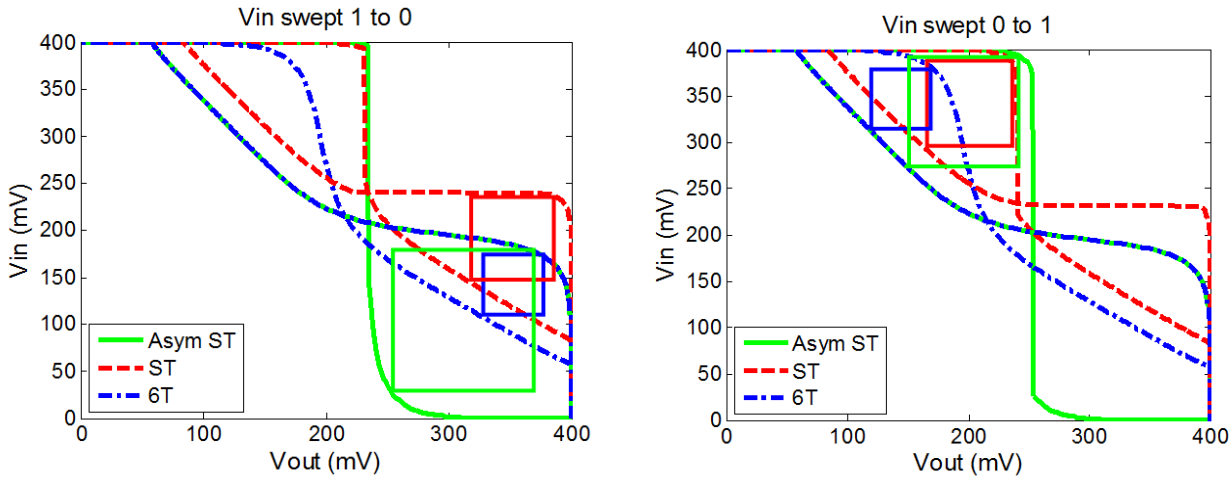


Figure 2.3: The 8T ST bitcell uses an asymmetric design to improve read margin without sacrificing write margin (as is the case for the asymmetric 5T cell [6])

The 10T bitcell [5] (Figure 2.2) uses Schmitt Trigger (ST) inverters to help improve the read static noise margin (RSNM). The NR2/NFR feedback transistors weaken the pull down network when VR is high, increasing the switching threshold of the right inverter. This means that the VL node would have to pull up much higher during a read in order to flip the cell, resulting in higher read stability. This bitcell has been shown by [5] to have 1.56X higher read SNM compared to the conventional 6T bitcell. The downside to this topology is that the four extra transistors result in a 33% area penalty compared to the 6T bitcell.

We propose an 8T asymmetric Schmitt Trigger bitcell (Figure 2.3). This bitcell uses single-ended reading and asymmetric inverters, similar to the asymmetric 5T bitcell in [6] to improve read margin. By using an asymmetrical design, the trip point of the ST inverter is increased, resulting in higher read stability. Because the 5T bitcell has only one access transistor, write assist methods must be used when trying to write a 1 into the bitcell. The advantage that this design has over the 5T bitcell is that it is written like a traditional 6T bitcell, which eliminates the need for write assist methods. The WL is pulsed high during both a read and write, and the WWL is only pulsed high during a write. In simulation

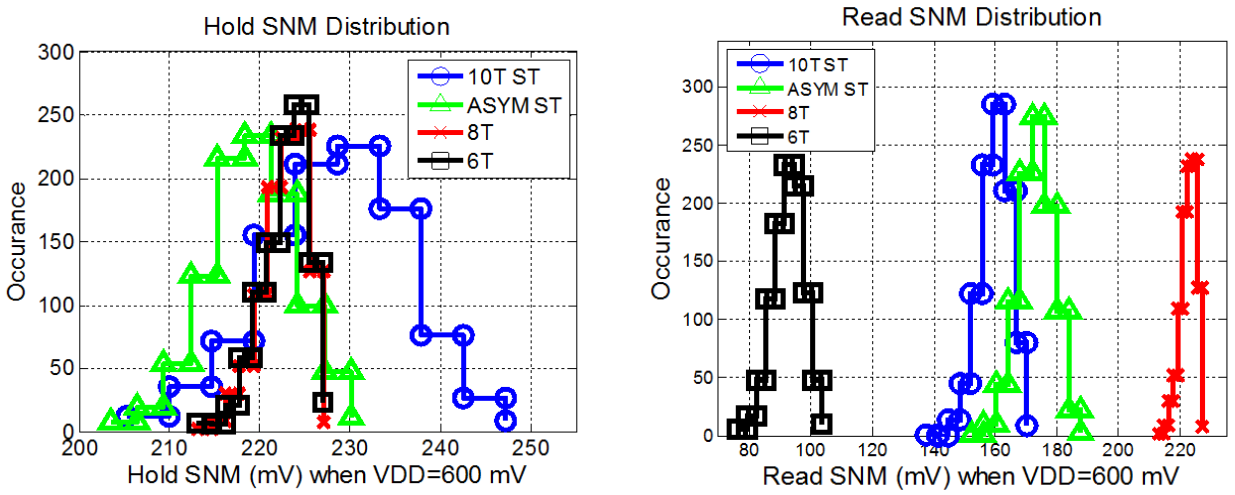


(a) (b)

Figure 2.4: Read butterfly curves for the asymmetric ST, ST, and 6T bitcells. Due to the asymmetric design of the cell, the 8T ST cell offers the highest read SNM

(Figure 2.4a) this bitcell achieves 86% higher RSNM than the 6T cell and 19% higher RSNM than the 10T ST bitcell without V_T variation.

In Figure 2.5a and 2.5b, we compare distributions of the read and hold static noise margins for each of the bitcells under the presence of V_T variation. The average hold static noise margin (HSNM) of the 6T and 8T bitcells is 222 mV, with the 10T ST slightly higher at 226 mV and the asymmetric ST slightly lower at 218 mV. However it is interesting to note that the standard deviation of the HSNM is 2.5 mV for 6T and 8T bitcells, 5.0 mV for the asymmetric ST, and 7.8 mV for the 10T ST bitcell. Therefore as the number of bitcells increases, the HSNM of the worst case bitcell in the 10T ST array will be lower compared to the other arrays. The average read static noise margin (RSNM) of the asymmetric ST is 88% higher than the 6T and 8% higher than the 10T ST. The 8T read distribution is the same as the hold distribution since the data is decoupled from the read operation. This assumes that the architecture of the 8T array does not interleave bits, or that a read before write scheme is implemented.



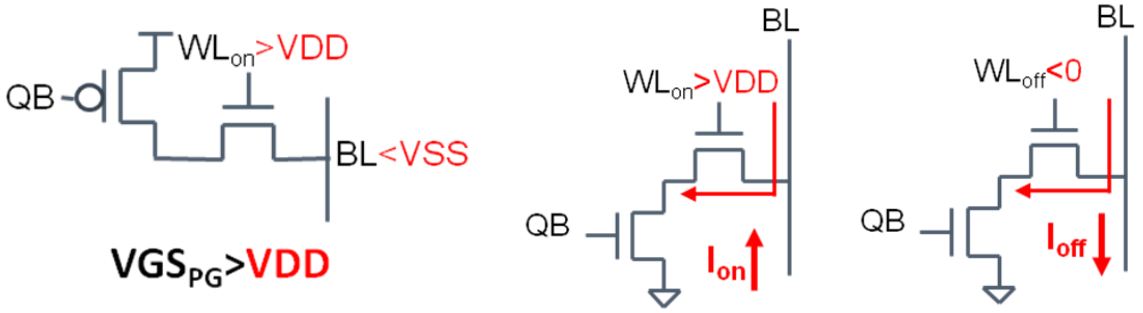
(a) (b)

Figure 2.5: (a) Hold and (b) Read static noise margin Monte Carlo simulation results. The 8T read and hold SNM are identical due to the 2T read buffer

2.2 Write Assist Methods

A write failure occurs when the value being stored in the bitcell is unable to be flipped. For example, to write the bitcell in Figure 2.1, the bitline (BL) is held high and BLB is held low. In order for the internal state to flip, pass-gate transistor XR must be able to pull node QB below the switching threshold of the left inverter. A ratioed fight is occurring between XR and PR, therefore transistor PR is usually made weak (by using a minimum sized device), to make writing easier. The downside to making the pull up transistor minimum sized is that it increases the V_T variation of this transistor.

The goal of write assist methods is to further weaken the pull-up transistor or strengthen the pass-gate transistor. There are several ways to accomplish this. The first is to increase the pass-gate to pull-up ratio by upsizing, however because we are operating in sub-threshold sizing is not an efficient knob. The second method is to collapse V_{DD}, which weakens the pull-up transistors by reducing their V_{GS} and V_{DS} [20, 25, 26]. The third and fourth methods involve strengthening the pass-gate transistors by either boosting the WL V_{DD} or reducing the BL V_{SS} [20–24, 27, 29]. These methods strengthen the pass-gate by increasing its V_{GS}.



(a) (b) (c)

Figure 2.6: (a) increasing the pass-gate V_{GS} allows for easier writing of the bitcell; (b-c) boosting the on current and reducing off current improves read access.

The downside to boosting the WL V_{DD} is that it reduces half selected cell stability. The weakness of reducing the BL V_{SS} is that it increases the BL swing, which increases the total write energy. This assist method can also lead to instability in cells sharing the same BL as the active cell if the BL is driven below the threshold voltage of the pass-gate transistor.

2.3 Read Assist Methods

Read failures can occurs in two ways. The first is that the bitcell is flipped during a read operation (referred to as read stability failure). This occurs when the XL and NL1 transistors (Figure 2.1) are sinking the large amount of charge from the highly capacitive BL, and the Q node rises above the trip point of the right inverter. In order to increase read stability, the pull-down transistor is made stronger than the pass-gate. This ensures that the voltage drop across NL1 (Figure 2.1) is not large enough to turn on PR. XL and NL1 form a resistive voltage divider during a read, so by upsizing NL1 we reduce its on resistance. This reduces the voltage rise seen on the Q node during a read. The second type of read failure occurs when the voltage difference between the BL and BLB is not large enough for the sense amp to determine the correct value (referred to as a read access failure). This happens in sub-threshold due to the BL leakage current in unaccessed cells causing the BL voltage to droop. Because the I_{ON}/I_{OFF} ratio is reduced in sub-threshold, it is feasible for the leakage

current through the unaccessed rows to pull the BL low at the same rate that the on current is pulling BLB low. This leakage current can be reduced by having less bitcells sharing the same bitline or by using one of the assist methods discussed below.

There are two goals involved in read assist methods. The first is to improve the stability of the cross-coupled inverters during the read by either raising the bitcell V_{DD} or reducing its V_{SS} [20, 21, 23–26]. While raising bitcell V_{DD} has been shown by [19] to result in larger gains in RSNM, the advantage of reducing the bitcell V_{SS} is that it significantly reduces read delay due to the body effect strengthening both the pull-down and pass-gate transistors [19]. The second goal is improve read access by increasing the read current (I_{ON}) and reducing the BL leakage in unaccessed cells (I_{OFF}). The read current can be increased by boosting the WL V_{DD} (Figure 2.6b). The downside here is that by strengthening the pass-gate, you reduce the stability of the cross-coupled inverters. In order to reduce bitline leakage current, the WL V_{SS} is reduced to a negative voltage (Figure 2.6c).

2.4 Chip Results

To compare bitcell topologies for subthreshold and to test assist features, we implemented a test chip that was fabricated in MITLL 180 nm FDSOI. This technology is specifically optimized for subthreshold operation by using an undoped channel to reduce capacitance and improve V_T control [31]. In addition, the gate spacer is widened and the source/drain extensions are removed which has only a small impact on I_{ON} due to low V_{DS} barrier. These optimizations result in a 50x reduction in energy-delay product compared to bulk silicon. As shown in Figure 2.7, the chip contains four SRAM arrays, with each array containing two four-Kb banks. The banks dimensions are 128 rows by two 16 bit words. The 6T and 8T cells are sized iso-area; the ST and asymmetric ST bitcells are also iso-area and suffer a 33% area penalty over the 6T and 8T bitcells. In order to easily test the read and write assist methods, peripheral and bitcell array voltages are controlled by separate supplies. The

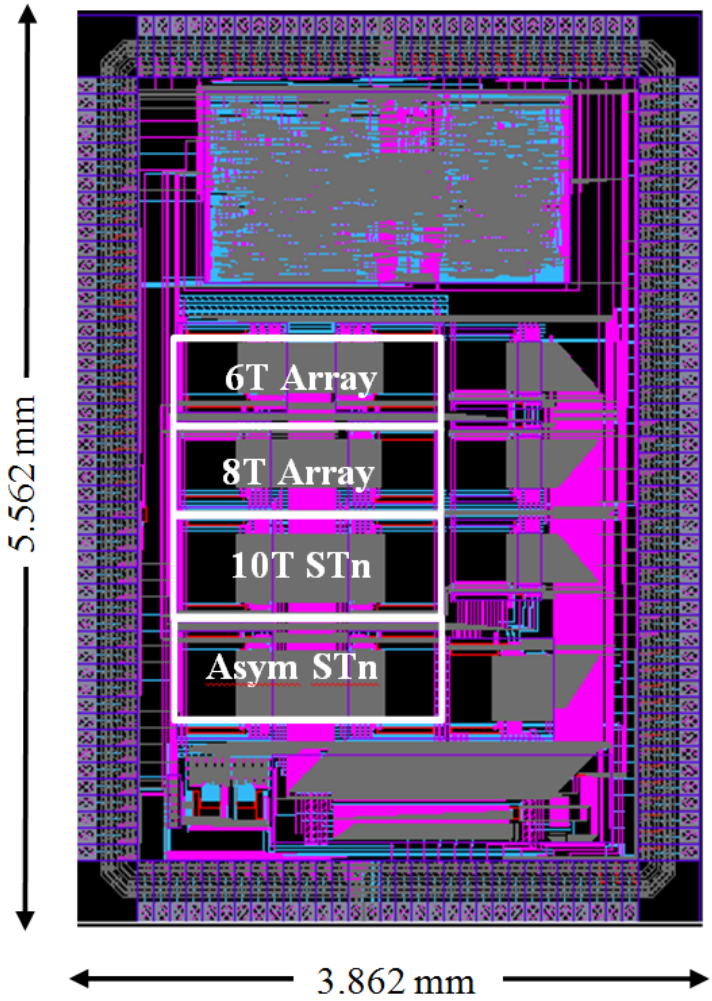


Figure 2.7: Schematic of the conventional latch based sense amp and the proposed modification

output pads used level converters to convert from sub-threshold to super-threshold in order to ensure that the data could be read by the Logic Analyzer. Because the main objective was reducing V_{MIN}, the chip was tested at 20 kHz to ensure that timing errors would not occur.

The test setup used a combination of Labview to control Keithley 2400 Source Meters and a Tektronix TLA7012 Logic Analyzer to handle the input and output signals. To determine the minimum data retention voltage (DRV), the memory is written with a known value, the voltage is dropped below nominal, then raised back to nominal and the data is read back out. The DRV is defined as the minimum voltage that the memory will retain the data. The second metric, write V_{MIN} is determined in a similar way. First a know value is written at

nominal V_{DD}, then the voltage is dropped and the opposite value is written. Next the voltage is raised back to nominal and read back out. To determine read V_{MIN}, a known value is written at nominal V_{DD}, then the voltage is dropped and the data is read back out. Each of the tests described above is an iterative process, with the voltage dropping lower at each step until it is close to ground.

Because the test chip was fabricated during the first run of a new technology (MITLL 180nm FDSOI), the yield was not ideal. We found full columns to be non-functional as well as a relatively high number of random bit failures. However, even with the non-ideal yield we were able to obtain some interesting results. The first result was that the SRAM proved to be write limited, meaning that the write V_{MIN} exceeded the read V_{MIN}. The best case write V_{MIN} at 80% yield was 620 mV, and the best case read V_{MIN} was 440 mV at 80% yield. This number was chosen because the yield of some of the arrays even at nominal voltage was below 90%. Therefore in order to capture the trends of the various assist methods, we chose to use a yield value of 80% in order to negate the effect of these outliers. The 8T bitcell offered the lowest read V_{MIN} which is surprisingly only 10% lower than the other three bitcells. This is interesting because in simulation, the RSNM of the asymmetric ST and 10T ST bitcells was much higher than the 6T bitcell. What we observed was that there seems to be a discrepancy between the SPICE models and silicon data. This is most likely due to the technology being relatively immature during its first fabrication run. As a result, it was difficult to compare bitcell topologies, which ended up producing very similar results in silicon. The cause of these discrepancies is not yet fully understood, and more research will be necessary to identify the source of error.

Although bitcell measurements yielded inconclusive results, we can still evaluate assist features. The results from the different write assist methods are shown in Figure 2.8a, 2.8b, and Table 2.1. Based on these figures, we conclude that BL V_{SS} reduction is the most effect method for reducing write V_{MIN}. This method outperforms the WL V_{DD} boost method across each of the bitcells. It is interesting to note that the 6T bitcell and Asymmetric ST bitcell

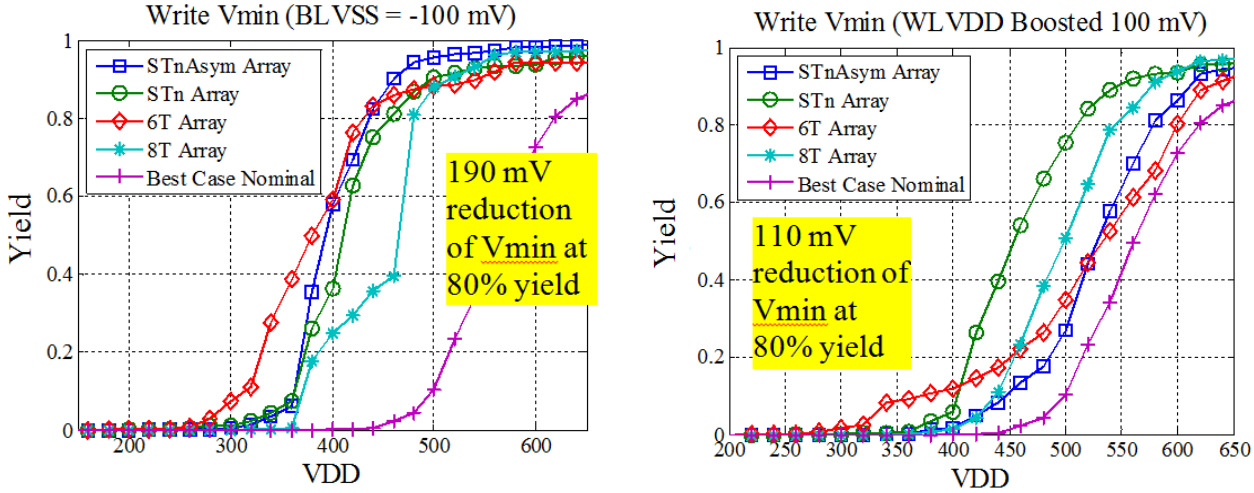


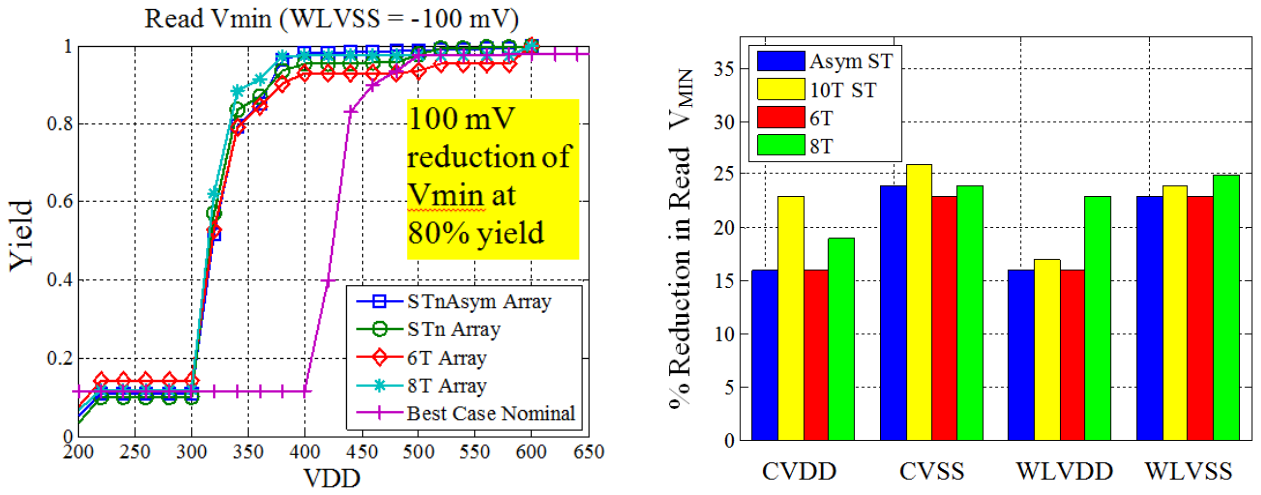
Figure 2.8: (a) effect of BL V_{SS} reduction on write V_{MIN}; (b) effect of WL V_{DD}boosting on write V_{MIN}; best case nominal refers to the bitcell with the lowest write V_{MIN} without the use of assist methods

Table 2.1: Percentage reduction in write V_{MIN} relative to write V_{MIN} without assist methods

Bitcell Type	**BL V_{SS} Reduction**	**WL V_{DD} Boost**
6T	30%	3%
8T	23%	12%
10T ST	27%	18%
Asymmetric ST	30%	7%

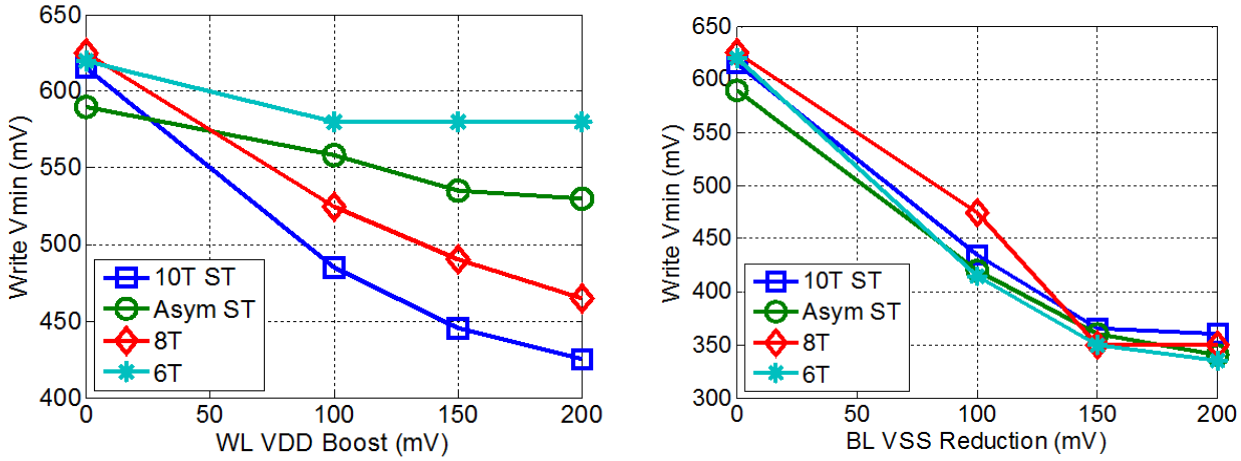
achieve the lowest write V_{MIN} at 430 mV, a reduction of 190 mV compared to the best case without assist methods.

As seen in Figure 2.9a, the WL V_{SS} reduction resulted in a 100 mV reduction in read V_{MIN} for each of the bitcells. The interesting trend with this plot is that each of the bitcells had almost identical read V_{MIN} values. This would suggest using a combination of the 6T bitcell and WL V_{SS} reduction is the most area efficient strategy for reducing read V_{MIN}. Based on the results from Figure 2.9b, reducing WL V_{SS} and bitcell V_{SS} consistently improved the read V_{MIN} for each of the bitcells. This suggests that bitline leakage was a major contributor



(a) (b)

Figure 2.9: (a) effect of WL V_{SS} reduction on read V_{MIN}; (b) comparison of read assist methods



(a) (b)

Figure 2.10: Effects of increasing the WL V_{DD}Boost (a) and BL V_{SS} Reduction (b) above 100 mV

to reduced read margin. It is also interesting to note that increasing the bitcell V_{DD} had the greatest impact on the 10T ST bitcell and WL V_{DD} boosting had the most positive effect on the 8T bitcell. Again, process features in the new technology most likely masked the effects of topological differences in the cells.

The results in Figure 2.10a and 2.10b show the effect of raising the assist voltage above 100 mV and are measured at a yield of 70% . As seen in Figure 2.10a, as the WL V_{DD} is

boosted up to 200 mV greater than nominal V_{DD}, the write V_{MIN} of the 10T ST and the 8T bitcells improve consistently. However, the 6T bitcell sees no improvement in V_{MIN} as the WL V_{DD} is boosted above 100 mV. Reducing the BL V_{SS} below -100 mV has a significant effect on reducing the write V_{MIN}. For the 8T bitcell, a reduction from -100 mV to -150 mV results in a 26% reduction in V_{MIN}. However, further reducing the BL V_{SS} to -200 mV does not have a significant effect on reducing V_{MIN}. Based on this data we conclude that using a combination of the 6T bitcell and negative BL V_{SS} is the most area efficient strategy for reducing write V_{MIN}.

2.5 Conclusions

In this chapter we present a novel asymmetric ST bitcell which uses single ended reading to achieve 86% higher RSNM than the 6T cell and 19% higher RSNM than the 10T ST bitcell in simulation. Although the asymmetrical ST and 10T ST bitcells offer improved read stability, silicon results in the first run of a 180 nm FDSOI process showed read V_{MIN} comparable to the 6T bitcell. Therefore it would be interesting to repeat this analysis in a more mature technology, to determine if the discrepancy was caused by the Spice models or by faults in the immature process. The second contribution of this chapter is a comparison of different read and write assist methods and various sub-threshold bitcell topologies. One important observation is that by choosing an effective assist method, the bitcell topology has a much less impact on V_{MIN}. Therefore the bitcell topology with less leakage and/or less area might be the optimum one for all the trade-offs. Another important observation is that sub-threshold bitcells proved to be write-limited, with unassisted write V_{MIN} 41% higher than read V_{MIN}. This trend has been shown by [17] to be especially true in newer technologies. In terms of write assist methods, the BL VSS reduction is the most effective, providing a 46% increase at -200 mV. Reducing WL V_{SS} or bitcell V_{SS} provided the largest reduction in read V_{MIN} of 26%. Based on our results, we conclude that using assist methods

as opposed to designing new bitcell topologies is more effective at reducing SRAM V_{MIN}.

Acknowledgments

We would like to thank MITLL for their help and support in the completion of this work.

Chapter 3

Subthreshold SRAM Design for a BSN

[1] Body sensor networks (BSNs) promise to provide significant benefits to the healthcare domain by enabling continuous monitoring and logging of patient bio-signal data, which can help medical personnel to diagnose, prevent, and respond to various illnesses such as diabetes, asthma, and heart attacks [32]. BSNs (Figure 3.1) consist of multiple nodes which are used to collect and transmit data to an aggregator, such as a smart phone. The basic functionality of the node is to sense a physical signal (such as temperature, heart rate, pressure, etc.), convert that signal into digital data, process the data on chip, and transmit the results back to the user. One of the greatest challenges in designing BSNs is supplying the node with sufficient energy over a long lifetime. A large battery increases the form factor of the node, making it unwearable or uncomfortable, while a small battery requires frequent changing and reduces wearer compliance. Another option is to use energy harvesting from ambient energy sources, such as thermal gradients or mechanical vibrations to provide potentially indefinite lifetime [32]. However, designing a node to operate solely on harvested energy

[1]This chapter is based on the published papers titled: "A Batteryless 19 μW MICS/ISM-Band Energy Harvesting Body Sensor Node SoC" [JB1], "A Batteryless 19μW MICS/ISM-Band Energy Harvesting Body Sensor Node SoC for ExG Applications" [JB3], and "A 6.45 μW Self-Powered IoT SoC with Integrated Energy-Harvesting Power Management and ULP Asymmetric Radios" [JB6]

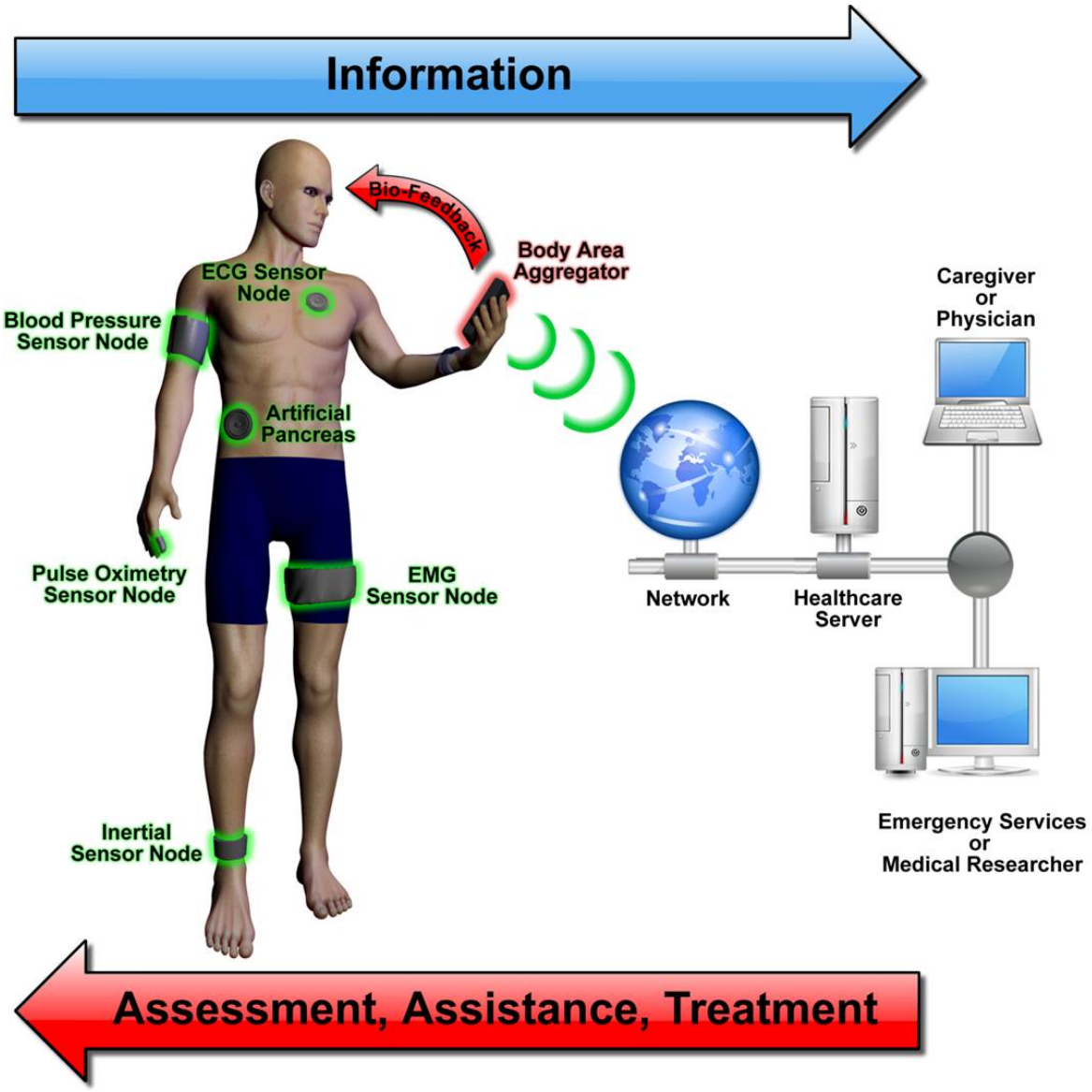


Figure 3.1: An example BSN contained multiple nodes and an aggregator. The basic functionality of each node is to collect and process physical signals and transmit to an aggregator [7].

requires ultra-low power (ULP) operation since the typical output of an energy harvester is in the 10's of μWs [33]. To ensure sustained operation of the node using harvest energy, on-node processing to reduce the amount of data transmitted, power management, and ultra-low power circuits are critical. Recently published BSNs have utilized subthreshold operation to keep overall system power less than 50W [34–37], making battery-less operation feasible.

For systems that rely on energy harvesting as in [35], an interruption of the power

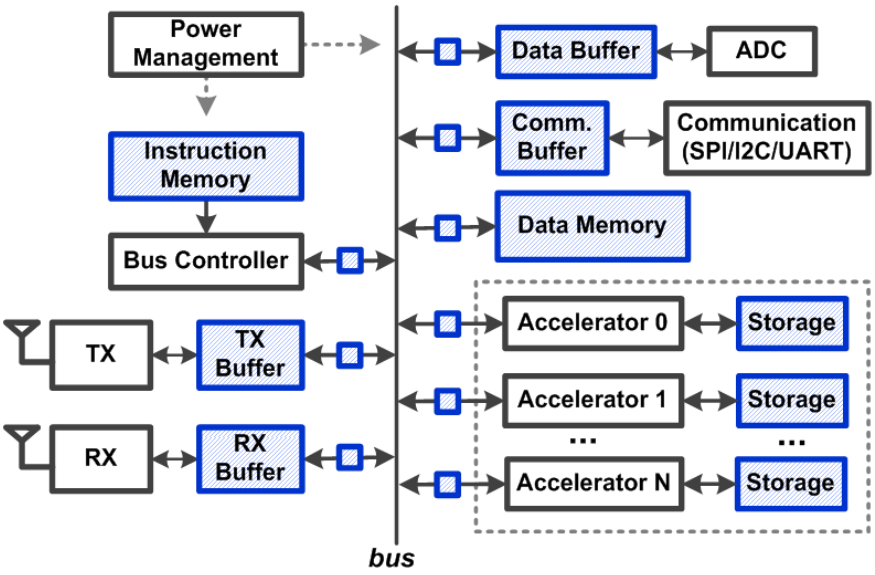


Figure 3.2: Typical block diagram for a wireless body sensor node SoC highlighting memory resources (shaded) [8]

source can cause complete system state loss due to the volatile SRAM-based on-chip storage, and this can result in the loss of vital patient data or chip instructions. Using on-chip power management as in [38] can reduce the probability of a complete system power loss by monitoring the available energy and turning off high-power blocks before a power outage. Similarly, BSN SoCs often have large sleep periods where most time is spent in data retention mode, making circuit leakage a primary concern. As technology continues to scale and operating voltages decrease into subthreshold, leakage dominates the power budget. This is especially critical for circuits that consume a large portion of the on-chip area such as SRAM memories used for program, data, and buffering. Recent BSN work such as [37] has shown that the dominant digital power consumer in a BSN can be the on-chip memories.

The amount of memory required for a biomedical system depends heavily on the target application. Making a flexible platform that can be used for various types of biosignal data acquisition and processing requires careful inspection of system components from a power and throughput perspective. Depending on the set of applications, the SoC might need to cater to programs with high compression ratios and low storage requirements while at other times

accommodating high throughput applications needing large amounts of storage. This creates a design challenge for flexible and ultra-low power (ULP) BSN design as power consumed by storage elements should scale with their use. A typical SoC platform block diagram is shown in Figure 3.2. This shows that most digital blocks on chip require buffering or storage, and its important to carefully evaluate the trade-offs between power, area, reliability, and integration complexity when choosing the type of memory used in each domain. In the rest of this chapter, we discuss the challenges of optimizing memory design from a system-level perspective, motivating the need for robust and low power storage. We then evaluate the challenges and design opportunities of designing SRAM-based memories for ULP SoCs.

3.1 System Level Memory Requirements

3.1.1 Storage Type Considerations

There are five general classes of storage shown in Figure 3.2 data memory, instruction memory, transceiver buffers, chip-to-chip communication buffers, and local block storage. The usage cases for each type of memory vary in terms of read/write frequency and capacity requirements, allowing for a variety of memory types and read/write optimizations.

Many recent BSN SoCs have relied on SRAMs to implement memory functionality on chip. SRAM macros are a common choice due to their density and energy efficiency for larger memory sizes typically used for data and instruction memories. To allow for ULP chip operation, most BSN SRAMs operate in the subthreshold regime, which reduces overall power but introduces challenges related to robustness and leakage. During power outage events, SRAM-based memories lose state, which can mean the loss of important medical data, chip state, and instructions. In this case, the chip must also be reprogrammed, which is inconvenient for longitudinally deployed systems.

An alternative to SRAM-based memories is using commercial, non-volatile memory (NVM) options such as Flash or EEPROM, but these require high read/write voltages and

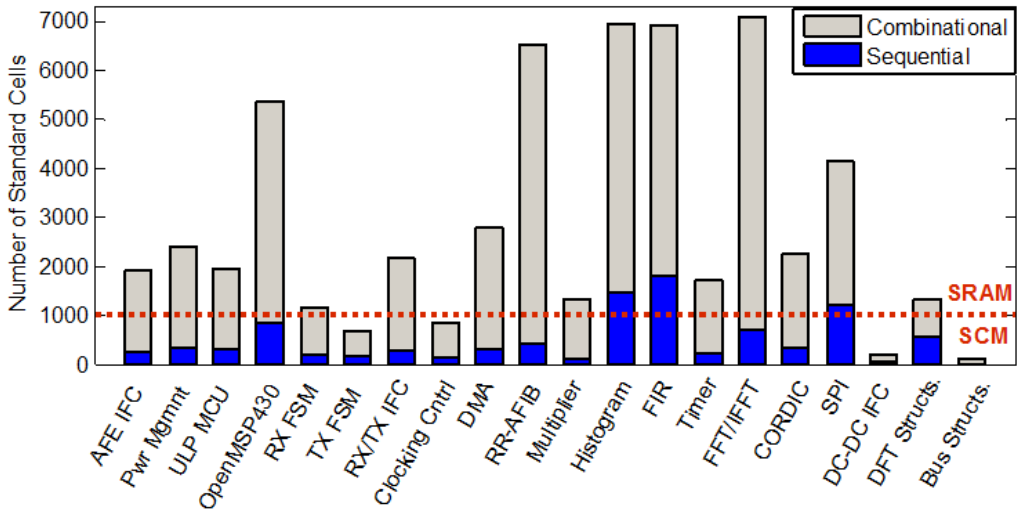


Figure 3.3: Breakdown of sequential and combinational elements for digital blocks in a BSN SoC [8]

have large peak current demands. Alternative NVM solutions exist that are promising for ULP designs but are either not yet commercially available or still require high read/write voltages. Examples such as FeRAM have enabled state-retentive ULP operation in recent designs [36, 39]. Another example, conductive bridging RAM (CBRAM) NVM, is targeted for ULP chips and can reduce write energy by 100x compared to Flash [40]. CBRAM also allows write operations at voltages down to 0.6V and read voltages down to 0.35V, making them compatible with existing BSN platforms. These NVMs were integrated on-die with an existing BSN platform, showing feasibility for ULP SoC integration [40].

Since all memories are not large (>1kb) on an SoC, standard cell-based memories (SCMs) synthesized using registers and latches must be considered for optimal energy and area efficiency for small-capacity memories. SCMs are easily integrated into digital blocks during synthesis without the need for extra power rings, reducing the overall area. Figure 3.3 shows the number of standard cells used for a set of BSN SoC blocks based on the system in [35]. In this example, sequential elements account for 17% of the total standard cell count and more than 40% of the digital chip area (not including SRAMs). Based on [41], blocks containing >1kb memory can benefit from SRAM-based storage, while blocks <1kb see power benefits

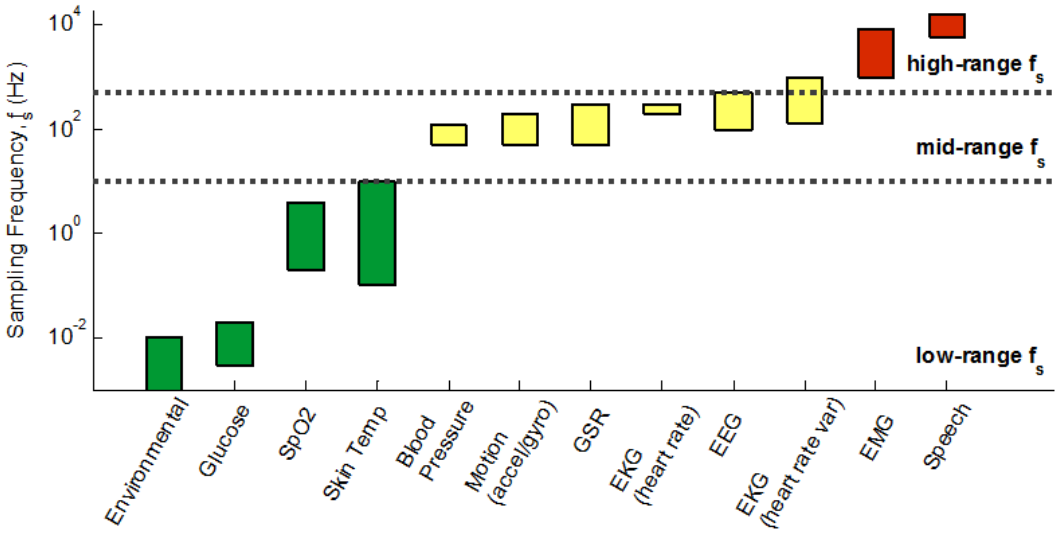


Figure 3.4: Body-worn platform sensing modalities/applications and their corresponding sampling rates [8]

using latch-based storage. The example in Figure 3.3 shows that some BSN accelerators with >1k sequential standard-cells can benefit from the integration of SRAM-based memories. This makes early design space exploration based on application requirements crucial to avoid excessively high power and area implementations.

3.1.2 Capacity Determination

Target applications in the biomedical space include a wide range of sensing modalities, sampling rates, processing requirements, and storage considerations. As shown in Figure 3.4, a majority of biomedical applications have sampling rates <1kHz, meaning new data samples are infrequent relative to common system clock rates used in systems operating in the near- or subthreshold regime (<1MHz). Minimum required memory capacities can be determined based on the data rate requirements between on-chip components. Along any processing path in the SoC shown in Figure 3.2, there exist data sources (e.g. sensor data from the ADC, data received over RX or SPI), data processing (e.g. FIR filters, CORDIC), and data sinks (e.g. TX, data memory). If the data rate of the source/processing unit, R_{SRC}, (in bits/s) is greater than the data rate of the processing/sink unit, R_{DEST}, then intermediate

buffering is required. The minimum amount of memory required, $N_{MinBuff}$, (in bits) to meet application constraints is dependent on the maximum continuous runtime of the program, t_{prog}, (in seconds). Compression that occurs during data processing eases the requirements on the intermediate buffer between the processing and sink units and reduces R_{SRC}. The final relationship for determining the minimum buffer size is shown in 3.1.

$$N_{MinBuff} = (R_{SRC} - R_{DEST})t_{prog}, R_{SRC} > R_{DEST} \tag{3.1}$$

Since wireless communication consumes the most power in BSN SoCs [41], minimizing the time that the transmitter or receiver is on is critical in energy-constrained systems. This can be accomplished using data encoding or compression methods to reduce packet sizes, but the maximum packet size (i.e. TX/RX buffer size) is determined by the available energy for processing. The maximum radio transmit and receive buffer sizes, $N_{RX/TX}$, (in bits) can be computed using an estimate for available system energy for communication, E_{avail}, radio startup energy, $E_{startup}$, and energy/bit of the radios, $Ebit_{RX/TX}$, shown in 3.2.

$$N_{RX/TX} = \frac{(E_{avail} - E_{statup})}{Ebit_{RX/TX}} \tag{3.2}$$

This can reduce the leakage overheads due to unnecessary memory resources.

3.2 SRAM Design Challenges For BSNs

In low performance applications, such as body sensor networks, node lifetime is the primary concern. To maximize node lifetime, it is important to operate at the minimum energy point, which typically lies in the subthreshold region [18]. While low voltage operation provides longer battery life, it also reduces the noise margins, particularly during read and write, due to reduced device drive strength and a higher sensitivity to V_T variation. Due to its ratioed design and minimum sized devices, the 6T SRAM bitcell is more susceptible to failure at low

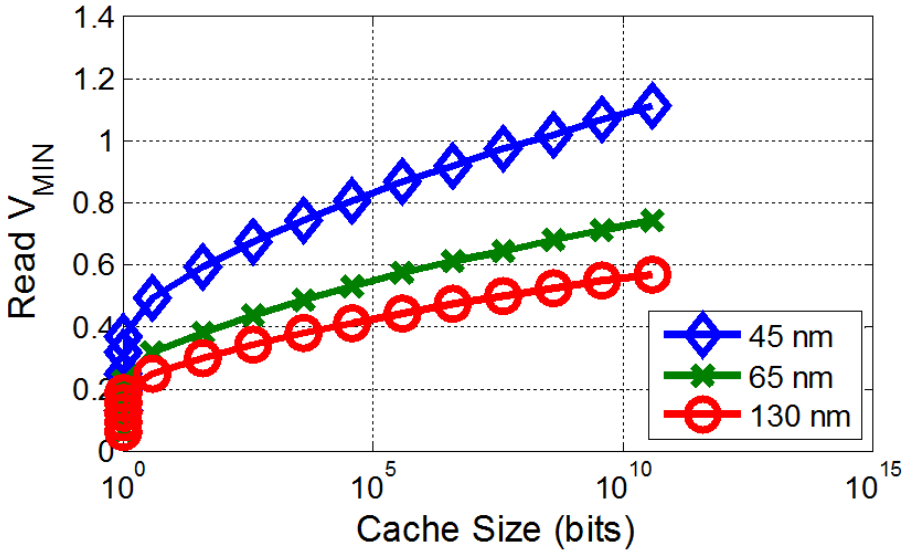


Figure 3.5: Read static V_{MIN} versus cache size across technology node

voltages than standard logic. Therefore, the two major types of designs of embedded SRAMs for body sensor nodes are the 6T bitcell at high V_{DD} (e.g. 1.2V) [42,43] and the 8T bitcell [4] at low V_{DD} (e.g. ~0.5V) [34,35,37,44]. Although many alternative bitcell topologies exist, e.g. [45], the 8T structure is most commonly used because it decouples the internal storage nodes from the bitlines (BLs) during the read operation and remains compact. In this section we highlight a few of the design challenges facing subthreshold SRAMs and the approaches taken to overcome these challenges.

Read Static Noise Margin

Read static noise margin measures the stability of the bitcell during a read operation. As VDD is scaled, this margin is reduced, and the probability of failure increases. Using the model from [15], we calculate the probability of a read upset failure across a range of supply voltages. Using the bit failure probability, we can calculate the minimum supply voltage (V_{MIN}) for a specified memory size that satisfies a given die yield. Figure 3.5 plots V_{MIN} versus cache size for the read operation to maintain a die yield of 95% (meaning 95% of dies have no failures during read). We can see from this figure that V_{MIN} increases as memory

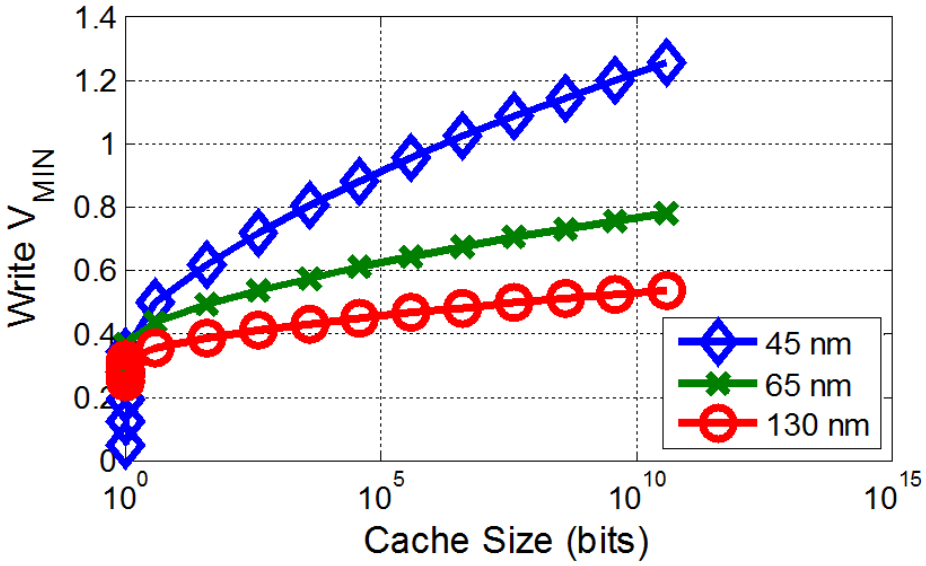


Figure 3.6: Write static V_{MIN} versus cache size across technology node

size increases. In addition, as feature size scales down, the minimum operating voltage for a fixed memory size increases, due to an increase in process variation. However, because BSNs typically operate at low clock speeds, leakage tends to dominate the power budget, so process scaling is less advantageous.

Static Write Margin

The write-ability of the cell is determined by the current ratio of the pass-gate to the pull-up device in the bitcell. In super-threshold, this ratio is set by upsizing the pass-gate device so that it is stronger than the pull-up device. This strategy does not work well in subthreshold due to the exponential dependence of I_{ON} on V_T. Static write margin is measured by setting bitline (BL) and bitline bar (BLB) to '0' and '1,' then sweeping the wordline (WL) from 0 to V_{DD}. The margin is defined as V_{DD} WL voltage when the Q/QB nodes flip. Once again using [15], we can measure the static write V_{MIN} required to meet a die yield of 95%. Figure 3.6 shows the write V_{MIN} as a function of memory size. Comparing the results of Figure 3.6 and Figure 3.5, we can see that the write V_{MIN} is typically slightly higher than the read V_{MIN} for the 45 nm and 65 nm nodes. The 130 nm design has a marginally lower write V_{MIN} due

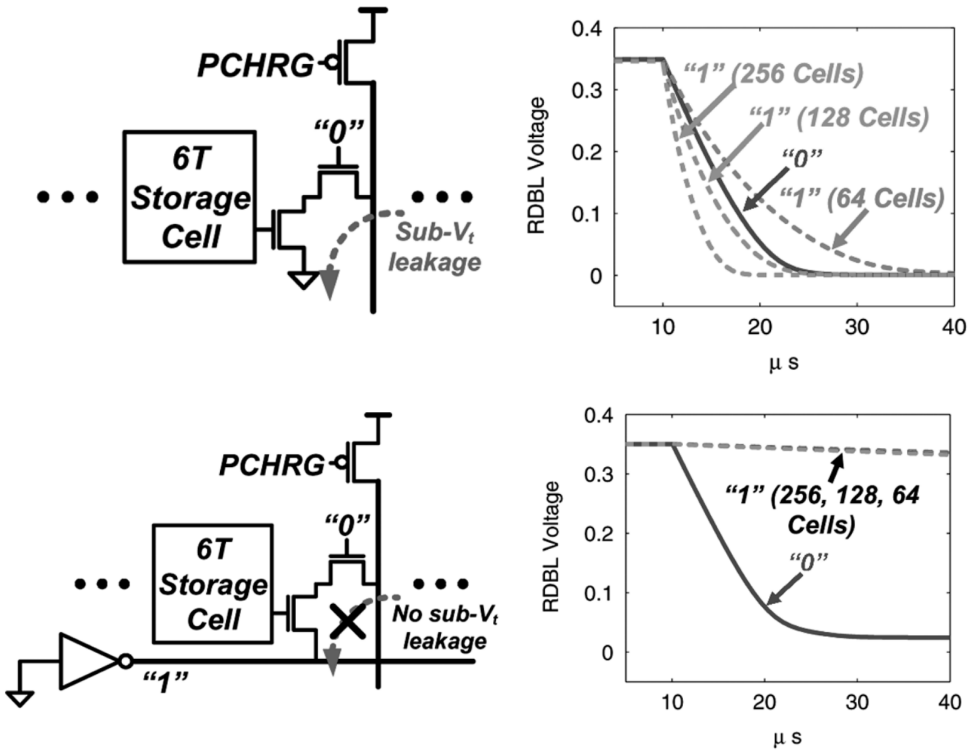


Figure 3.7: (top) Connecting the read buffer footer to ground causes the Read BL (RBL) to droop, while in (bottom) this leakage path is removed by driving the footer of unaccessed rows to V_{DD} [4]

to the fact that at the N-P ratio is heavily skewed at lower voltages in this process.

Read Access Stability

The 8T bitcell solves the problem of read upsets by adding a 2T buffer to isolate the internal storage nodes from bitlines. However, this introduces a new problem due to the single-ended design. Because we are operating in subthreshold, the I_{ON}/I_{OFF} current ratio is greatly reduced. This means that during a read, the value of I_{READ} could approach the total BL leakage current (number of cells per bitline * leakage per bitcell) in designs with a large number of cells per bitline. This can lead to read access stability failures where the leakage through the un-accessed cells pulls the read BL low while attempting to read a '1' (Figure 3.7). In [4,35], this is prevented by driving the footer of the 2T read buffer in the un-accessed rows to V_{DD} (Figure 3.7). This reduces the total leakage of the cache but also incurs an

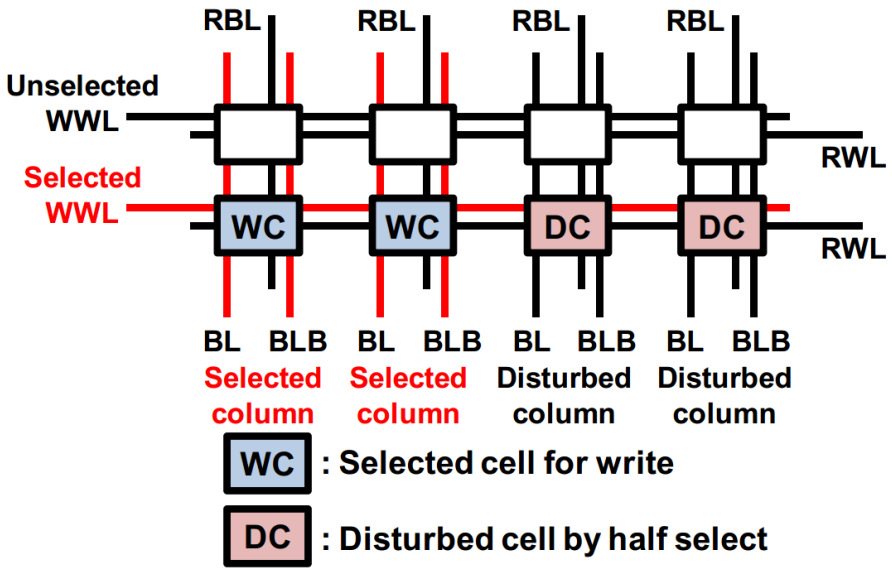


Figure 3.8: Half-select disturb during a write operation in bit interleaved designs

energy overhead to switch the footer of the accessed word during a read. [34] and [44] use a 4T read buffer (inverter + transmission gate) to hold the read BL high during a read '1'; however, this increases the leakage per bit when the memory is not in sleep mode. [37] uses a hierarchical bitline scheme to minimize the capacitance and leakage of the local bitline and improve read performance. The cost of this technique is higher layout area, as the global bitline must be routed to each of the local bitlines.

Half Select Instability

The 8T bitcell is immune to data instability during a read, however it suffers from half select instability during a write (Figure 3.8). During a write, the selected WL goes high to write data into the cell. In bit interleaved designs (e.g. designs with more than one data word per row), the write WL is shared by each word in the row. Therefore in unselected columns, the pass-gate devices turn on during a write, creating a read SNM disturb. If the read static noise margin of these cells is not sufficient, it could cause the cell to unintentionally flip states. Because the layout area of the bitcell is so small, it is impossible to route multiple local wordlines to each word in a row without increasing the cell size. Another solution is

to simply not use bit interleaving and place only a single word in each row. However this solution is not feasible in designs containing $> 1kb$. The third solution is to use a read before write scheme which will be described in detail in section 3.4.

3.3 Revision 1

The first version of the BSN chip required a 1.5 kB instruction SRAM / ROM and 4kB data SRAM. The instruction memory (I_{MEM}) was required for storing 12 bit instructions for execution by the digital power management (DPM) block and the PIC processor. It is programmed once during startup using a scan chain, then once the chip is deployed, the memory is only used for reading out instructions. The data memory (D_{MEM}) is used as a FIFO (First In, First Out). During signal acquisition, the digital data is streamed directly into the DMEM. Once the memory is full, the memory address is reset to 0 and old data is replaced with new data. When an atrial fibrillation (Afib) event is detected, the previous eight heart beat samples stored in the data memory are transmitted wirelessly from the radio.

3.3.1 Bitcell Design and Characterization

The first step in the design process is designing a reliable bitcell. The three metrics that we consider for reliability are: read static noise margin, write noise margin and read access stability.

Read Static Noise Margin (RSNM)

Monte Carlo simulation shows that the mean-4σ point (for RSNM) was 0.526 mV (Figure 3.9). 4σ was chosen because the maximum memory size is 4 kB, therefore the worst case read margin is approximately four standard deviations from the mean. With a margin this low, any noise source on the supply could potentially result in an accidental bit flip during a read. Therefore, to remedy this issue we decided to use the 8T bitcell [4], which eliminates

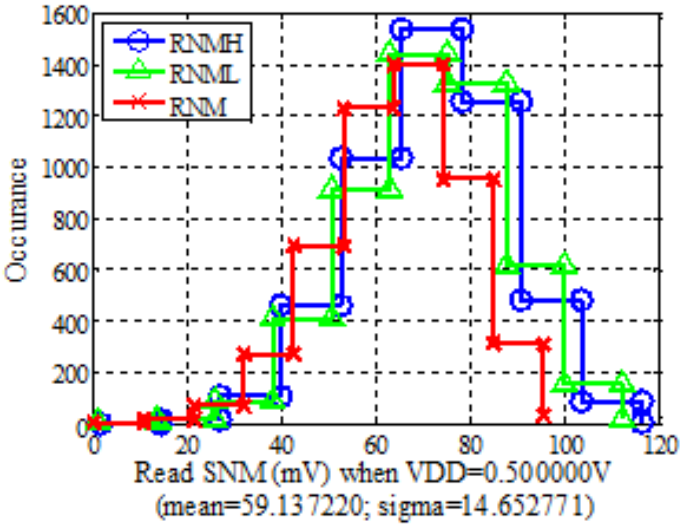
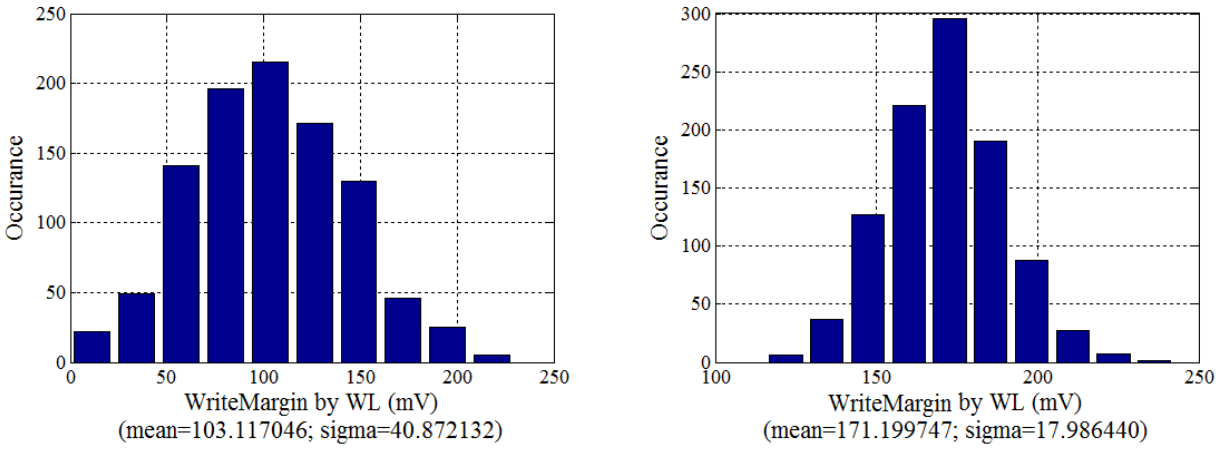


Figure 3.9: Read Static Noise Margin Distribution at 0.5V

the problem of read instability in designs that do not use bit interleaving. To eliminate the half-select instability that occurs during a write, a row buffer is used to store the eight words per row. A write only occurs when the row buffer is full and the entire row is then written. Since each row of the D_{MEM} contains eight 16-bit words, the memory is only written once every eight cycles. This control is managed by the direct memory access controller (DMA) which is a sub-threshold accelerator to interface the D_{MEM} with the rest of the SoC. We are able to use this approach due to the fact that the D_{MEM} is used as a FIFO (First-in, first-out), where each successive write increments the memory address by one. This same technique is used to write the I_{MEM}, however the control in this case is through the scan chain. During a read, both the instruction and data memories output the entire row, and the individual word is selected by the DPM (I_{MEM}) or the DMA (D_{MEM}). This type of design allows us to reduce the number of reads and writes to once every eight cycles, thus achieving close to an 8x energy savings (minus the overhead of additional buffers).



(a) (b)

Figure 3.10: Comparison of the write noise margin between the (a) high V_T cell and (b) regular V_T cell

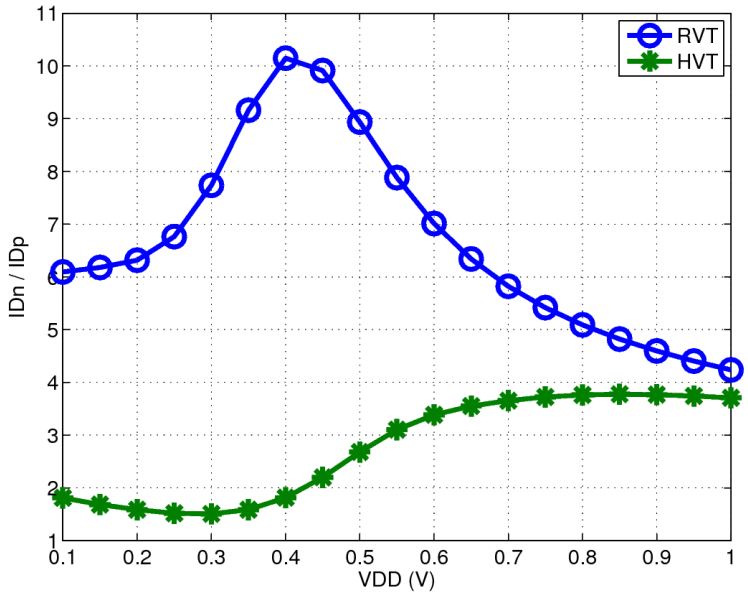


Figure 3.11: A comparison of the RVT and HVT I_N/I_P ratio across V_{DD}

Write Static Noise Margin (WSNM)

The next metric to consider is write noise margin. Because leakage is a major concern in SRAMs due to the large number of inactive bitcells, the ideal bitcell would use high V_T (HV_T) devices to reduce this wasted energy. However, Monte Carlo shows that the worst case static noise margin of the bitcell using HV_T devices was close to zero, meaning the bitcells

were failing to write at 500 mV (Figure 3.10a). Therefore, to ensure adequate write margins, we decided to use regular V_T (RV_T) devices (Figure 3.10b). Using these devices, we were able to achieve a worst case write margin of 100 mV ($\mu - 4 * \sigma$). We can see from these Monte Carlo results that the high V_T bitcell has a lower mean and a higher sigma. The lower mean can be explained by the plot in Figure 3.11. This figure plots the relative strength of the NMOS device compared to the PMOS (I_N/I_P). At 0.5V (the BSN digital voltage), the RV_T NMOS is 8.94X stronger than the PMOS, while the HV_T NMOS is only 2.68X stronger than the PMOS. As explained in Section 1.1.2, the NMOS pass-gate must be stronger than the PMOS pull-up device to write to the cell. Because the process is so highly skewed towards the NMOS at low voltages, the RV_T bitcell is able to achieve a 66% higher average write margin. In addition, the σ_{WM} is 56% lower in the RVT design compared to the HVT design. This is due to the fact that the threshold voltage of the RVT devices is ∼350 mV while the V_T of the HVT devices is ∼500 mV for the NMOS and ∼550 mV for the PMOS. This means that the HVT bitcell is operating in the sub-threshold region, while the RVT bitcell is operating in near threshold. As shown in equation 1.4, on current is exponentially dependent on V_T, which leads to the higher sensitivity to variation as shown by the HVT cell. The downside to using RV_T devices is that it increases the leakage current per bitcell by 24X.

Read Access Stability

The final metric to consider to ensure reliability is read access stability. Typically in super-threshold, read stability is determined by the minimum BL differential required for the sense amp to generate the proper output. However because speed is not an issue due to the five microsecond cycle time, no sense amp is required. Because the 8T bitcell has single ended reads, the output of the read BL (RBL) is fed directly into a standard buffer. The real concern for this design is that the leakage current from the unaccessed cells does not cause the RBL to droop when a '1' is being read (Figure 3.7). By reducing the number of bitcells per column, we can reduce the total leakage current, however this results in a larger number

of banks. Having more banks increases the total area due to increased redundancy of the periphery cells (WL drivers, BL drivers, output buffers). Another approach is to reduce the leakage from the unaccessed rows by precharging the footer voltage (Figure 3.7) to V_{DD}. [4] shows that this technique reduces the RBL leakage to almost zero. This technique does however introduce a new problem. Because the footer must be driven to V_{SS}, when a row is active, the driver of the footer must sink all of the current from each column (in this design there are 128 columns). By using a charge pump to boost the input gate voltage of this buffer to $2*V_{DD}$, we are able to achieve a 13.5X increase in on current. Even with this increase in current, we found that the maximum number of bitcells per row to ensure that the RBL pulled low within half of a cycle was 64.

3.4 Revision 2

To improve programmability, the second revision of the BSN design included an openMSP430 [46] on chip. In order interface the existing memory design with the openMSP430, the memory timing must be compatible. This meant that memory could no longer be treated as a FIFO, but must have the capability to act as a true random access memory. While this is not a problem for reading data, it does create a problem during the write operation. In the previous design, the write operation waited for each of the eight words in a single row to be written, and only then did it write a complete row. For this revision, the assumption that the word address would be incremented by one during each successive write is no longer valid. This means that the prior solution to the half-select problem described in Section 3.2 would have to be modified to ensure data stability during a write operation.

Three possible solutions for improving half-select instability are: using an alternative bitcell such as [5] to improve the RSNM, reducing the WL V_{DD} to improve RSNM [20–22, 47], or implementing a read before write scheme [48–50]. The downside to using an alternative bitcell such as [5] is that it introduces a large area overhead due to the additional bitcell

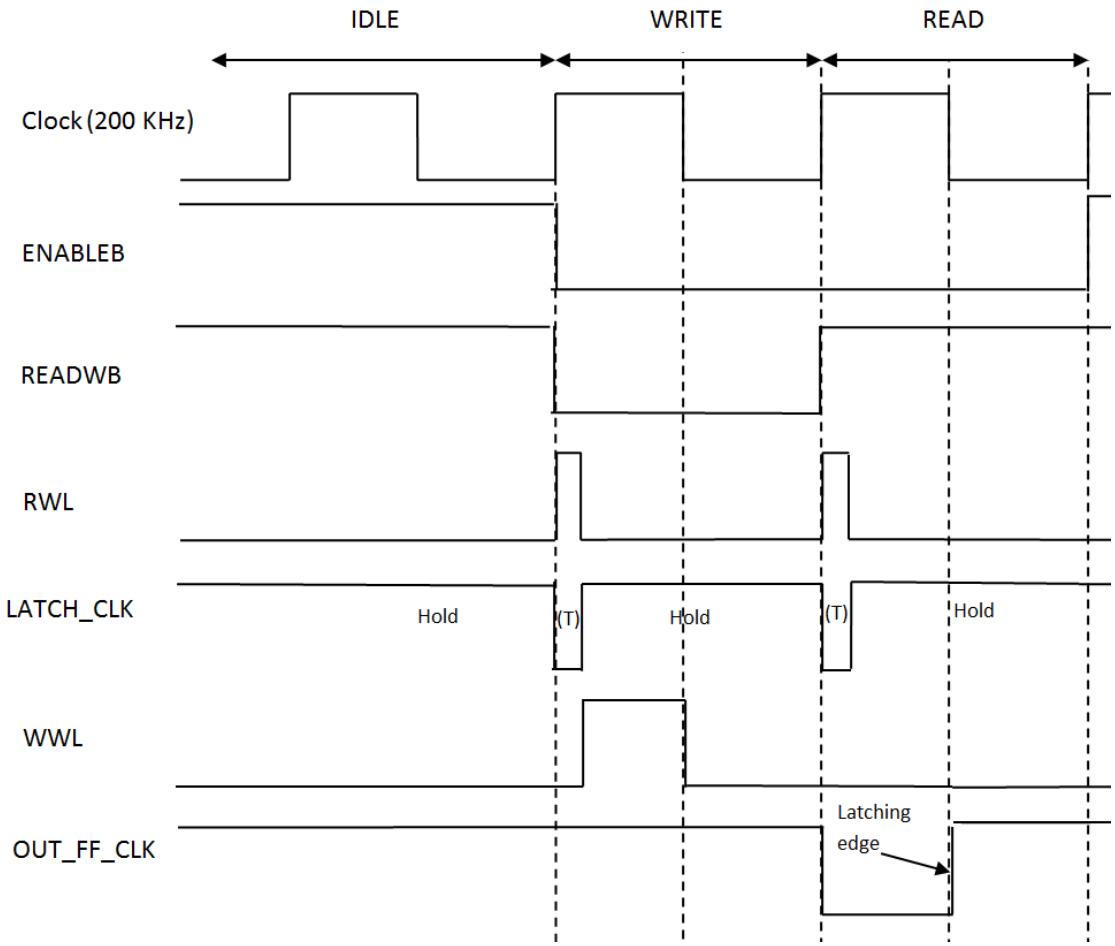


Figure 3.12: Memory timing diagram. During a read or write, the RWL is pulsed in the first half of the cycle. The read data is latched on the rising edge of the Latch Clock, and the write completes in the first half of the cycle.

devices. The additional devices also increase the leakage per bit. Simulation data from Revision 1 showed that leakage consumed 64.7% of the total memory energy during a read access. Therefore, reducing leakage energy per bit has a significant effect on the total memory energy. Lowering the WL V_{DD} reduces the noise injected into the cell during half-select. However, as shown in Section 3.3, the bitcell has a worst case write margin of 100 mV; therefore lowering the WL V_{DD} below 500 mV could create static write failures. The read before write scheme takes advantage of the 8T bitcell's ability to perform a read without disturbing the cell data. During a write, the active row is read and the data from the row is stored in a latch (Figure 3.12). This data is sent to the write BLs, along with the new data

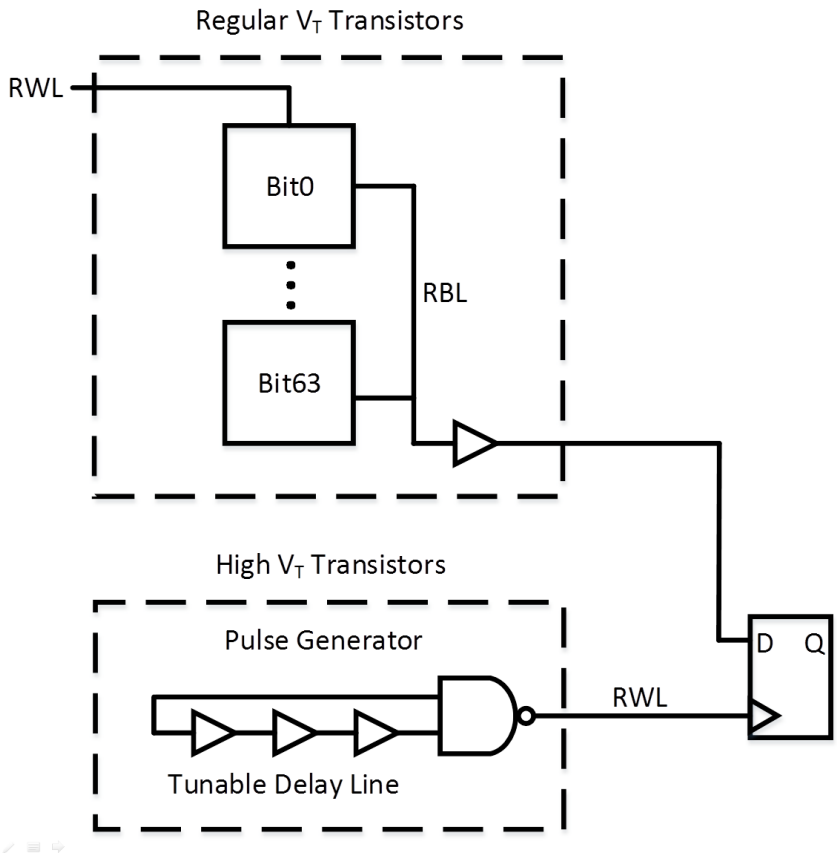


Figure 3.13: The pulse generator was designed using HV_T devices, while the memory core was designed using RV_T devices

word. The new data is written into the active column, while the old data is written back into the unselected columns. This scheme ensures that the data stored in half-selected columns is not disturbed at the cost of higher dynamic energy during a write.

Revision 2.1

In Revision 2.1, the width of the read WL (RWL) and Latch Clock (Figure 3.12) is set by a pulse generator circuit embedded within the memory (Figure 3.13). The pulse generator is tune-able from a range of 131 ns up to 1.17 μs, measured at the TT (Typical-NMOS, Typical-PMOS) process corner. During a read or a write, the pulse generator raises the RWL, allowing the bitcell to discharge the RBL to ground. The pulse width generator is designed using HV_T devices to reduce the number of buffering stages necessary to achieve the

target pulse width. As long as the read WL pulse width is greater than the worst case read delay, then the RBL is able to discharge to ground before the RWL is disabled, and before the output latch enters the hold state (Figure 3.12). The worst case read delay and largest tune-able pulse width are shown in Table 3.1. We can see from this table that at the TT and FF (Fast-NMOS, Fast-PMOS) corners that the output of the pulse width generator is 6.52x and 7.22x larger than the worst case read delay, respectively. Once the RWL goes low, the write occurs in the reminder of the first half of the clock cycle (while the Clock is high). The second half of the clock cycle is used to precharge the read BL before the next cycle. During a read, the data from the 128 bit output latch is latched into a 16 bit flip-flop. This ensures that the output of the memory is stable one half of a clock cycle before and after it is latched on the rising edge by the memory controller (ensuring setup and hold time are met).

Table 3.1: Worst case read delay, and largest pulse width generator output

	TT Corner	**FF Corner**
Read Delay	163 ns	25 ns
Pulse Width	1.17 μs	162 ns

After fabrication, the memory was unable to reliably read and write data at any voltage from 0.3V up to 1.0V. The bit yield across the die ranges from 0% up to 90%. The error causing this poor yield is a 'stuck at 1' error. Based on the schematic, the most likely cause of this problem is the RBL not discharging to ground before the RWL pulse is disabled. This was confirmed by sweeping the width of the RWL pulse. The pulse width has eight programmable settings, and test results show that the memory only begins to function at the two highest settings. Our assumption in designing the pulse width generator is that the regular and HV_T devices would track to the same process corner. However, based on Table 3.1, we can conclude that the HV_T devices are closer to the FF process corner, while the RV_T devices are TT.

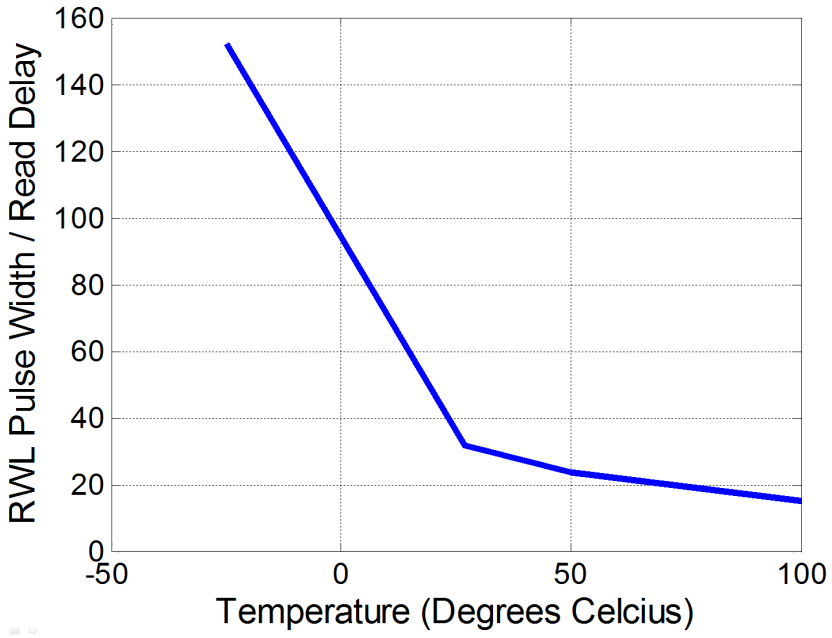


Figure 3.14: At low temperatures, the ratio of the RWL pulse width divided by the read delay increases

To test this theory, we ran a simulation of both devices across temperature to characterize the current ratio of the HV_T and RV_T devices. Results show that as the temperature is lowered, the $\frac{I_{Rvt}}{I_{Hvt}}$ ratio increases. This means that at low temperatures, the HV_T devices become weaker relative to the RV_T. This is important because in order for the memory to function properly, we need to increase the RWL pulse width, and decrease the read delay. Figure 3.14 shows that as the temperature is lowered, the ratio of the RWL pulse width divided by the read delay increases. This means that at low temperatures, we expect higher yields due to the increase in the RWL pulse width relative to the read delay. To test this in silicon, the memory was placed into a thermal chamber and the temperature was dropped to $-20\,^{\circ}\mathrm{C}$. On average, this increases the bit yield roughly 10% relative to the yield at room temperature. This confirms the theory that the HV_T devices are fast relative to the RV_T devices. This section highlights one of the major issues with designing muli-V_T systems: process variation may skew one set of devices relative to the other, causing issues with timing.

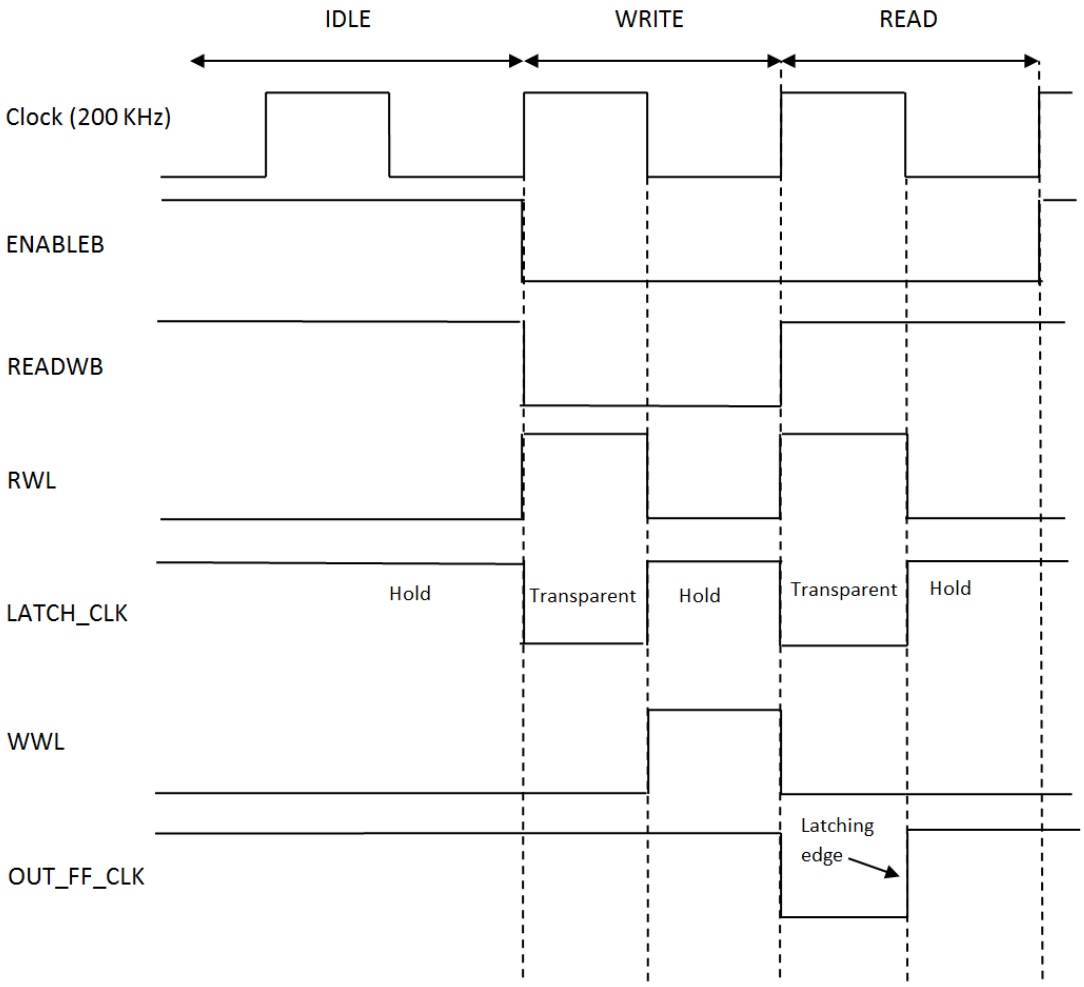


Figure 3.15: Memory timing diagram. During a read or write, the memory is read in the first half of the cycle. The read data is latched on the falling edge, and the write occurs in the second half of the cycle.

Revision 2.2

From Revision 2.1, we learned that using pulse generators is not a robust solution for implementing a read before write scheme in sub-threshold. In this revision, the pulse generator is replaced by the system clock. This allows us to directly control the length of the RWL pulse width by varying the system clock frequency. Figure 3.15 shows the new timing scheme. The advantage of this scheme is that both the read and write WL pulses are dependent on the clock period. Therefore, they can be easily tuned post-fabrication. The write and read precharge are performed in parallel in the second half of the cycle. This is

possible due to the fact that the read and write BLs are decoupled in the 8T bitcell. This creates a new problem on the rising edge of the clock when the new write data inputs and address bits arrive at the memory. To ensure that the new data isn't written into the address from the previous cycle, the write WL needs to pull low before the new data is accidentally written. This problem is addressed by inserting hold time buffers at the output of the column address and data input flip-flops.

The design was again fabricated in a commercial 130nm technology. Test results showed reliable (100% bit yield) operation down to 0.35V, with an average standby power of 0.871μW per kB. Table 3.2 presents a comparison of this work to existing BSN SoCs. We can see from the table that this work achieves one of the lowest operating voltages among the current state of the art without the use of peripheral assist methods.

Table 3.2: Comparison to existing BSN SoCs

	This Work	[34]	[44]	[37]	[42]	[43]
Application	EKG, EMG, EEG	-	Temp, Pressure	EKG	EKG, TIV	EKG
Technology	130 nm	180 nm	180 nm	130 nm	180 nm	180 nm
Supply Voltage	0.35-0.7 V	0.5 V	0.4-0.5 V	0.25-0.7V	1.2 V	1.2V
Memory Size	12 kB	496 B	5 kB	4 kB	20 kB	42 kB
Bitcell Type	8T	14T	10T	8T	6T	6T

3.5 Conclusions

In this chapter we have presented a sub-threshold SRAM embedded on a body sensor node SoC. We have motivated the need for SRAM on BSN SoCs as the optimal solution for large

amounts ($> 1kb$) of data buffering. In addition, we have presented the challenges in designing sub-threshold SRAMs for BSNs including read and write static noise margin, read access stability, half-select instability, and timing closure. In Revision 1, we presented a 1.5 kB instruction memory and a 4 kB data FIFO. Bitcell optimization included: selecting the the 8T cell [4] to prevent static read upsets, choosing the RV_T cell over the HV_T for improved write noise margins, and the use of a row buffer to prevent half-select disturbs. In Revision 2, we presented a truly random access embedded sub-threshold memory design. To meeting the timing requirements of the openMSP [46], the design employed a read-before-write scheme to ensure half-select stability. This design achieves an average standby power of 0.871μW per kB and reliable operation down to 0.35V in silicon, one of the lowest among existing state of the art solutions.

Chapter 4

Modeling SRAM Dynamic Write V_{MIN}

[1] Static Random Access Memory (SRAM) is a critical component of todays SoCs consuming large amounts of area and often setting the critical timing path. Technology scaling has allowed reductions in area, power, and delay. In order to continue this trend, the minimum operating voltage (V_{MIN}) of SRAMs must continue to scale down. This has become increasingly difficult as devices enter the nanoscale range due to increased device variability and leakage. SRAM devices are typically minimum sized, which further compounds this problem [51]. The increase in both variation and leakage leads to reduced read and write margins, making it more difficult to design low power SRAMs that meet frequency and yield constraints. In addition, as the capacity of SRAM arrays continues to increase, the stability of the worst case bitcell degrades. Therefore it has become increasingly important to accurately predict SRAM yield at a given supply voltage.

The most common method for evaluating yield is through Monte Carlo (MC) simulations. However for very large arrays (i.e. 10 Mb) the number of simulations required to identify the worst case bitcell becomes prohibitively large. Because the majority of simulated samples

[1]This chapter is based on the published papers titled: Leveraging Sensitivity Analysis for Fast, Accurate Estimation of SRAM Dynamic Write V_{MIN}" [JB4] and "Modeling SRAM dynamic V_{MIN}" [JB5]

do no lie in the tail region, a full MC simulation is not an efficient method for estimating very small failure probabilities. A common approach to reducing simulation time is to run a relatively small number of samples and then fit the resulting distribution to the normal distribution. Once the μ and σ are known, the stability of the worst case bitcell can be identified. The problem with this approach is that it can only be applied to data sets that replicate a known distribution [14, 15]. However, it has been recognized that the dynamic write margin does not fit the normal distribution [15, 52]. The distribution resembles the long tail F-distribution, but does not match it exactly. Because the distribution does not closely match any known statistical distribution, it is difficult to model without full simulation of the tail region.

One approach to solve this problem is to develop purely analytical models as in [53, 54]. However these approaches are less accurate because approximations must be made to simplify the problem. [52] showed that these approximations can lead to errors in failure probability estimates of up to three orders of magnitude. Two methods that reduce MC run time by effectively simulating only points in tail region include importance sampling [55, 56] and statistical blockade [57, 58]. These techniques can be used to reduce simulation time by several orders of magnitude. However, because the calculation of the dynamic margin requires a much larger number of simulations than the static noise margin (SNM), these methods still require long simulation times. In this chapter, we present a methodology using sensitivity analysis to further reduce the time required to calculate dynamic write V_{MIN}.

4.1 Background

The dynamic noise margin is defined as the minimum pulse width required to write the cell, or T_{CRIT} [59–64]. The benefit of this metric is that it takes into account the transient behavior of the bitcell, which is not captured by static metrics. This metric has been shown by [62] to produce more accurate V_{MIN} estimations than static metrics, since static metrics

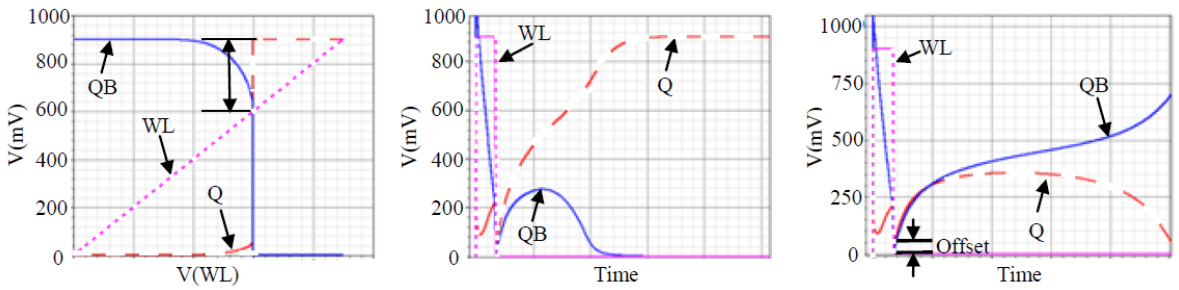


Figure 4.1: a) DC sweep of WL allows for the write margin to be calculated in a single simulation, b) successful write operation c) even with QB pulling below Q at the end of the WL pulse, the write is not successful

give optimistic write margins and pessimistic read margins, due to the infinite wordline (WL) pulse width. In this chapter we focus primarily on dynamic write-ability since the static metric results in optimistic yields and because it has been shown that write failure is more likely in newer technologies [17]. The downside to using transient simulations is that they are more time costly, especially when running large numbers of Monte Carlo samples to isolate the worst case bitcells. Whereas static margin can be calculated using a single simulation (Figure 4.10a), the calculation of T_{CRIT} requires a binary search. This takes on average ten to fifteen iterations to determine the critical pulse width with a high level of accuracy. Figure 4.10 b-c shows that in the presence of variation, pulling QB below Q doesnt guarantee a successful write.

In [14, 15] the author defines static V_{MIN} under the presence of variation. The V_{MIN} is defined as the point where the SNM becomes zero. The author uses the hold SNM to define the data retention voltage, the read SNM to define read V_{MIN}, and the WL sweep method to define write V_{MIN} [13]. To estimate the failure probability at a given supply voltage, each metric is simulated across a range of V_{DD}s. Each resulting distribution is then fitted to the normal distribution. As V_{DD} is reduced, the mean of the write distribution decreases and the standard deviation increases. Then using equations 4.1 and 4.2, the failure probability can be calculated for any V_{DD}. In equation 4.1, s is equal to the SNM which causes a failure, which in this case is just zero. μ_l and μ_h are defined as the SNM for writing a zero and writing a

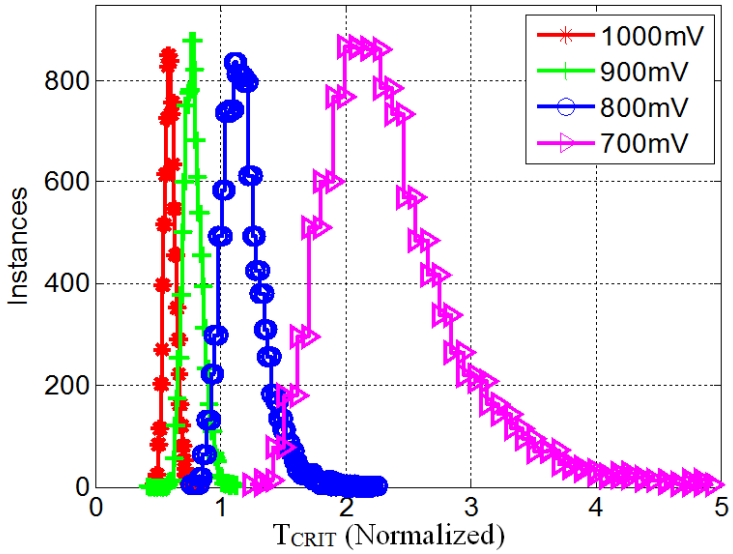


Figure 4.2: The distribution of T_{CRIT} does not fit a normal distribution

one. Equation 4.2 is a best fit line representing the value of μ and σ versus V_{DD}.

$$p_f = \frac{1}{2}erfc\left(\frac{\mu_h - s}{\sqrt{2}\sigma_0}\right) + \frac{1}{2}erfc\left(\frac{\mu_l - s}{\sqrt{2}\sigma_1}\right) - \frac{1}{4}erfc\left(\frac{\mu_h - s}{\sqrt{2}\sigma_0}\right) * \frac{1}{4}erfc\left(\frac{\mu_l - s}{\sqrt{2}\sigma_1}\right) \tag{4.1}$$

$$\mu = \mu_0 + a\left(v^2 - v_0^2\right) + b\left(v - v_0\right), \sigma = \sigma_0 + c\left(v - v_0\right) \tag{4.2}$$

The problem with this approach is that the dynamic margin is not normally distributed. From Figure 4.2, the shape of the T_{CRIT} distribution is long tailed, making the normal approximation inaccurate. In order to apply a similar method as in [14,15], a new distribution must be identified that fits the data. Using the curve fitting toolbox in MATLAB, we were able to determine that the T_{CRIT} distribution closely matches the Frechet distribution, whose probability density function is shown in (4.3), for $V_{DD} \leq 800$ mV.

$$f(x) = \frac{\alpha}{\beta}\left(\frac{\beta}{x - \mu}\right)^{\alpha+1} * exp\left[-\left(\frac{\beta}{x - \mu}\right)^{\alpha}\right] \tag{4.3}$$

The Frechet distribution is an extreme value distribution, commonly used to estimate the maxima of long sequences of random variables. The three parameters: μ, α and β represent

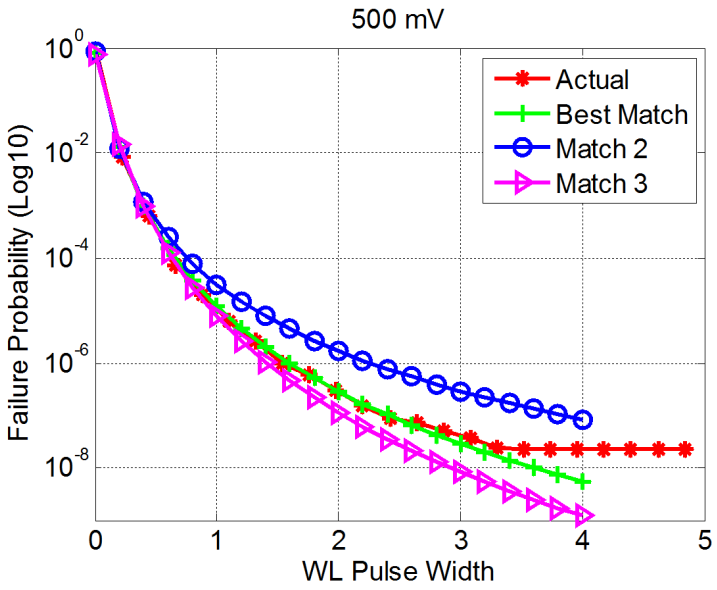


Figure 4.3: The three distributions match the MC data however, they do no match the tail of the distribution

the offset, shape, and scale respectively. When applying the fit to a sample size of 5K Monte Carlo points, we were able to closely match the Monte Carlo data. However, due to the shape of the distribution, the fit software calculates a large confidence interval for the three fitting parameters, resulting in errors modeling the tail region. Figure 4.3 shows three possible fit lines predicted by the curve fitting tool. The actual line represents data taken using importance sampling to estimate the extremely low failure probabilities. At a certain point, the probability of failure remains constant as the WL pulse width is increased. This is due to the fact that at 500 mV, the memory is approaching the static failure point, and therefore a wider WL pulse does not reduce the failure probability. This figure shows that it is not possible to extrapolate the tail of the distribution using only a small (5K) Monte Carlo sample.

Another method to determine the dynamic write margin of the worst case bitcell is recursive statistical blockade [58]. However, in order to accurately determine the dynamic margin using binary search, its takes an average of twelve simulations. Using this method, it would take over 894,000 simulations to identify the worst case write margin for a 100 Mb

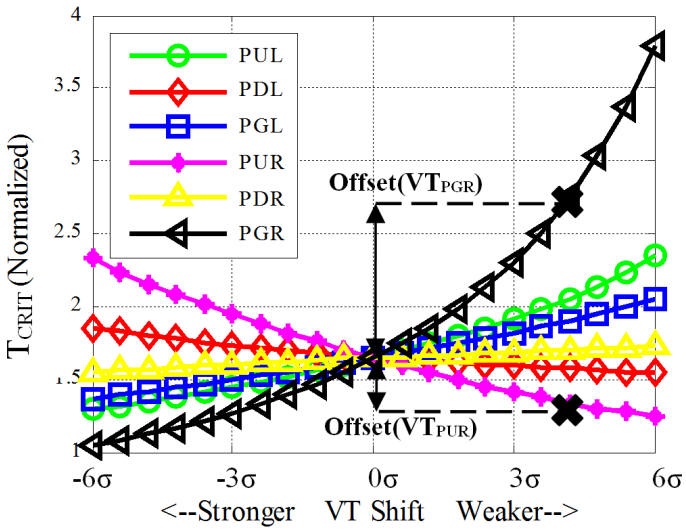


Figure 4.4: In order to characterize the bitcell, the VT of each transistor is swept independently

memory. In the next section we will describe a method using sensitivity analysis to accurately predict the worst case bitcell that requires less than 1,100 simulations.

4.2 Estimating Dynamic Write Margin (T_{CRIT})

In order to reduce the cost of running large numbers of transient Monte Carlo simulations, we propose using sensitivity analysis to quickly generate the T_{CRIT} distribution [65]. The first step in this method is to sweep the threshold voltages of each transistor to produce the plot shown in Figure 4.4. The PU, PD, and PG labels represent the pull-up, pull-down, and passgate transistors respectively. The left node of the bitcell is initially holding a 0 and the right node is initially holding a 1. The x-axis represents the V_T shift of each transistor ranging from -6σ to 6σ; the y-axis represents the resulting T_{CRIT} value. When sweeping the V_T of each transistor, all other transistors are left at nominal V_T. We then fit each curve to a third order polynomial. Once each of the curves has been fitted, the next step is to generate a V_T distribution for each of the six transistors (Figure 4.5). This is done by generating a normal distribution using the sigma values from the Spice model.

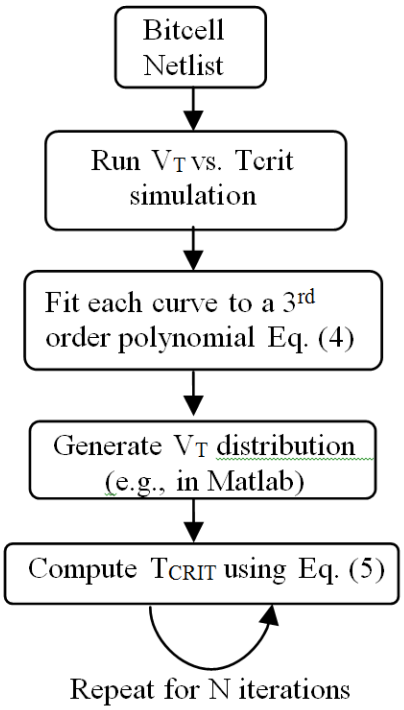


Figure 4.5: Flow chart of the proposed T_{CRIT} model

$$T_{CRIT-OFFSET} = aV_{T-Shift}^3 + bV_{T-Shift}^2 + cV_{T-Shift} + d \tag{4.4}$$

Next, the V_T offset of each transistor is plugged into (4.4), and the six offsets are then added to the nominal case to produce the T_{CRIT} prediction:

$$T_{CRIT} = T_{CRIT-NOM} + T_{CRIT-Offset-PUL} + ... + T_{CRIT-Offset-PGR} \tag{4.5}$$

This calculation is repeated N times depending on the desired sample size. Clearly, computing (4.5) is much faster than running the set of simulations required to find T_{CRIT} using Spice.

One assumption made by sensitivity analysis is that the V_T variation of each transistor has an independent effect on T_{CRIT}. In order to verify this, we repeat the V_T sweep on each transistor, adding Monte Carlo variation to the other five transistors. If the V_T variation of

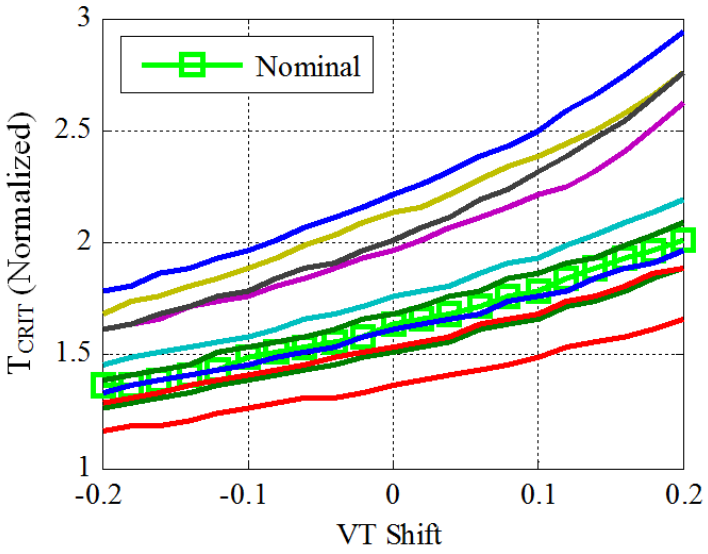


Figure 4.6: Transistor variation has a close to independent effect on T_{CRIT}. Each line represents a single Monte Carlo iteration

each transistor does have an independent effect on T_{CRIT}, then we would expect the shape of the V_T curve to remain the same in the presence of variation. As shown in Figure 4.6, the shape of the V_T shift vs. T_{CRIT} curves does not change significantly; the curves are simply shifted up or down from the nominal case. There is some slight overlap between the curves which leads to small errors in the predicted value.

In order to verify the accuracy of this methodology, we compared the margin of the worst case bitcell calculated by the model and using the recursive statistical blockade tool [58]. The accuracy of the model was tested for three memory sizes: 100 Kb, 10 Mb, and 100 Mb. The model was also tested across a range of V_{DD}s from 500 mV up to 1V. The results are shown in Table 4.1. We can see from the table that the worst case error is only 6.83%, while the average is 3.01%. A positive percentage error means that the model overestimated the T_{CRIT} value, resulting in slightly pessimistic margins.

The advantage of this method is that it greatly reduces simulation times while sacrificing very little accuracy compared to statistical blockade. This same technique can be applied to importance sampling to reduce the total run time. Simulating the V_T curves in Figure 4.4 requires approximately 18.8 minutes. Once these curves have been produced, random

Table 4.1: Percentage Error Across Memory Size

Voltage	100K	10M	100M
500 mV	6.83%	-4.25%	6.51%
600 mV	2.96%	-3.69%	5.61%
700 mV	-0.18%	-2.64%	4.75%
800 mV	0.83%	-0.7%	1.21%
900 mV	-4.5%	0.83%	1.43%
1V	-2.72%	-2.2%	-2.27%
Average	3.01%	2.39%	3.63%

samples are generated (e.g., by MATLAB) and applied to equation 4.5. The run time for the sensitivity analysis increases linearly with the number of samples. The total run time for a 100 Mb memory is only 32 minutes (Table 4.2). One disadvantage of the statistical blockade tool is that in order to determine the worst case write margin, two separate test cases must be run: writing a 0 and writing a 1. This means that two separate filters must be generated, as well as two separate sets of Monte Carlo simulations. The total number of simulations required for the recursive statistical blockade tool is 894,288, corresponding to a total CPU runtime of 60 hours.

In summary, our method provides a 112.5X speedup at the cost of an average loss in accuracy of 3.01% and a worst case loss of 6.83%.

4.3 Dynamic Write V_{MIN} Prediction

Write V_{MIN} is defined as the minimum operating voltage in which the write operation will succeed. We can define this minimum operating point using either static or dynamic metrics. Static write V_{MIN} is defined as the voltage that results in an SNM of zero, meaning that even if the WL is pulsed high for an infinite time period, the write operation will fail. Dynamic

Table 4.2: Total Run Time Comparison

	Statistical Blockade (Number of Simulations)	**Sensitivity Analysis (Run Time)**
Initial Simulation	24,000	18.8 min
100 Kb	107,904	0.72 s
10 M	531,096	72 s
100M	231,288	12 min
Total Simulations	894,288	-
Total Run Time	60 Hours	32 Minutes

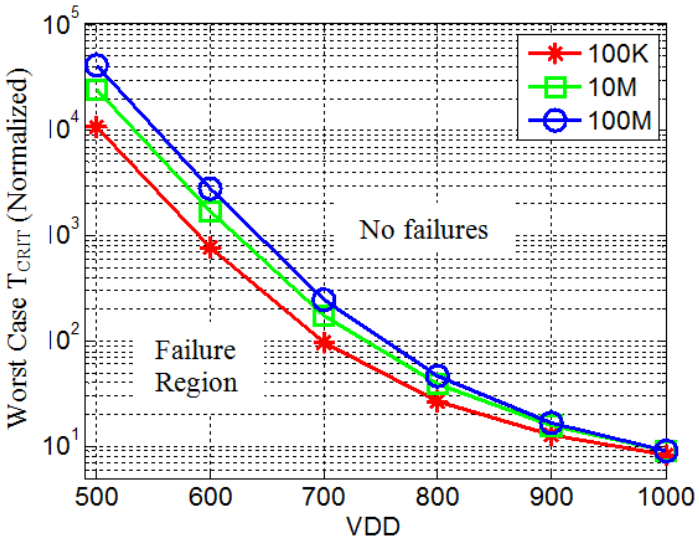


Figure 4.7: The lines represent the point of single failure while the region above represents no fail, and the region below represents multiple bit fails

write V_{MIN} is defined in [62] as the voltage in which the worst case T_{CRIT} value is larger than the word line pulse width. In order to calculate dynamic write V_{MIN} using our approach we can repeat the procedure described in Figure 4.5 for varying V_{DD}. In this example we chose six test points between 0.5V and 1V. The procedure can be repeated for different memory sizes, and the worst case dynamic write margin can be plotted versus V_{DD}.

In the plot on Figure 4.7, the individual lines represent different memory sizes in bits.

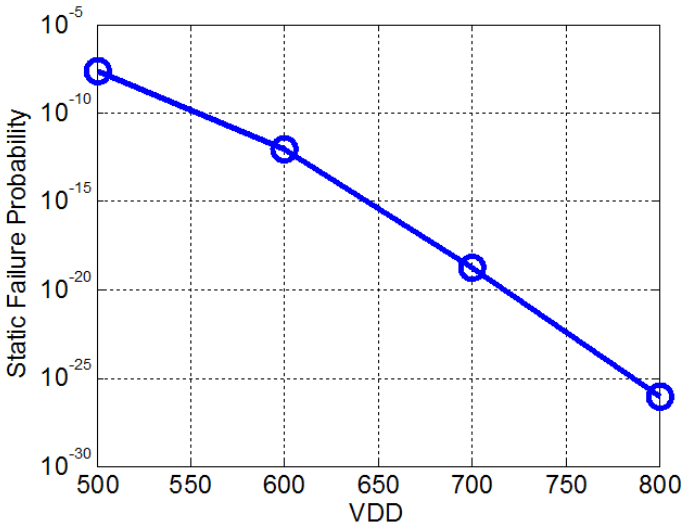


Figure 4.8: Static failure probablity versus V_{DD}

The curves represent the (T_{CRIT}, V_{DD}) point of the first single bit failure. Below the curve represents multiple bit failures and the region above the curve has no bit failures. By choosing a WL pulse width constraint and memory size, we are able to determine the minimum operating voltage necessary to ensure reliable operation. We can see that as the size of the memory increases, the critical word line pulse width for the worst case bitcell also increases. This effect becomes more pronounced as V_{DD} is scaled down. Generating these same results across V_{DD} using statistical blockade would take approximately 360 hours.

To show the importance of using dynamic write V_{MIN} as opposed to static, we have plotted the static failure probability versus V_{DD} in Figure 4.8. At 500 mV, the static failure probability is 2.57e-8, which means that in a 100 Mb SRAM cache, there will likely be two to three bitcells statically failing. At 600 mV, this failure probability decreases by over five orders of magnitude, meaning that at this operating voltage, it is statically unlikely that there will be any bitcells failing. As V_{DD} is raised the static failure probability continues to decrease. Clearly margining based on static failure probabilities leads to drastic overestimation of SRAM yield. While dynamic metrics allow for a much more accurate measure of V_{MIN} they are much more costly to simulate than static metrics. This is why having a method to accurately predict dynamic write V_{MIN} is so valuable.

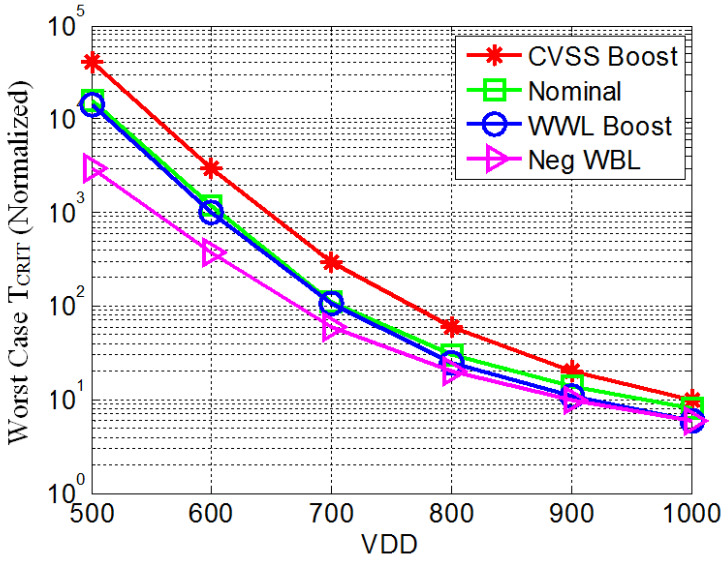


Figure 4.9: Measuring the effects of write assist methods on dynamic write V_{MIN}

4.4 Impact of Assists on Dynamic Write V_{MIN}

As SRAMs continue to scale, peripheral assist methods will be needed to allow for continued voltage scaling [22, 23, 27, 29, 66, 67]. Therefore it is important to determine which write assist methods provide the largest reductions in dynamic write V_{MIN}. Some assist features such as cell V_{SS} (CV_{SS}) boost aim to weaken the cross coupled inverters, by reducing the gate to source voltage of the PMOS device. However, we will show that this technique actually increases the worst case T_{CRIT} value. With write WL (WWL) boosting, we increase the gate voltage of the passgate transistor in order to improve the drive strength. The negative BL reduction technique also increases the drive strength of the passgate by dropping the voltage on the source. Using sensitivity analysis, we can quickly and accurately make predictions about which assist methods are the most effective across a range of V_{DD}s.

In Figure 4.9, the memory size is 1 Mb, and the ∇V for each assist method is 100 mV. We can see that the CV_{SS} boost technique actually increases the worst case T_{CRIT} value. This is due to the fact that the weakening of the cross coupled pair results in a longer time delay for the node initially holding a 0 to pull high. At high V_{DD}, the negative BL reduction and WWL boost have comparable effects on reducing T_{CRIT}, however as V_{DD} is reduced, the

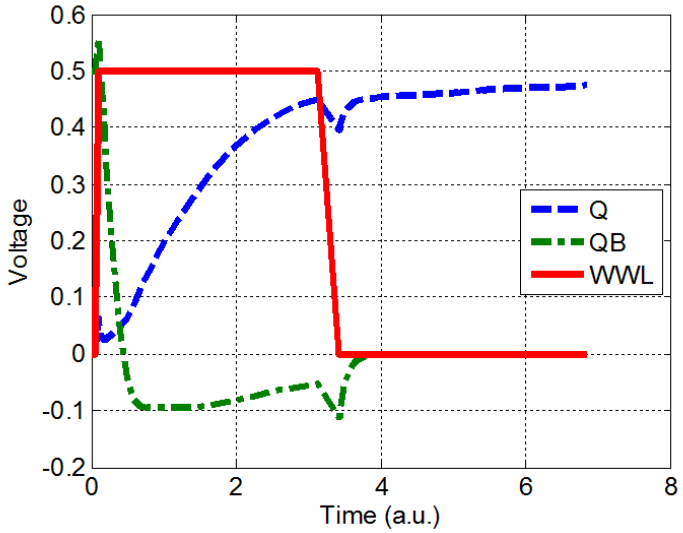


Figure 4.10: The negative BL reduction results in improved write times due to the QB node being pulled negative

negative BL technique provides much larger reductions in T_{CRIT}. This can be explained by the V_T curves in Figure 4.4. We can see from this plot that as the PUR transistor (initially on) gets stronger, the T_{CRIT} value increases as expected. However, as the PUL transistor (initially off) gets stronger, the T_{CRIT} value decreases. This second order effect is due to the fact that as the PUL transistor gets stronger, it is able to more quickly pull the internal node high, resulting in a slightly faster switching time. Because the negative BL technique is passing a stronger 0 (e.g. a negative voltage) into the bitcell, it is effectively strengthening the PUL transistor. Therefore the effect of negative BL is twofold: it strengthens the passgate transistor as well as the PUL transistor. This second effect is not seen with the WL boost technique because it is not passing a strong 0. Figure 4.10 shows that at lower V_{DD} (e.g. 500 mV), the QB node pulls low relatively quickly, while the majority of the write operation is spent waiting for the Q node to pull high. This explains why boosting the WWL has negligible effects on reducing T_{CRIT} as compared to negative BL at lower V_{DD}. These results were obtained using the analysis described in section 4.2, resulting in a total speedup of 672X over statistical blockade.

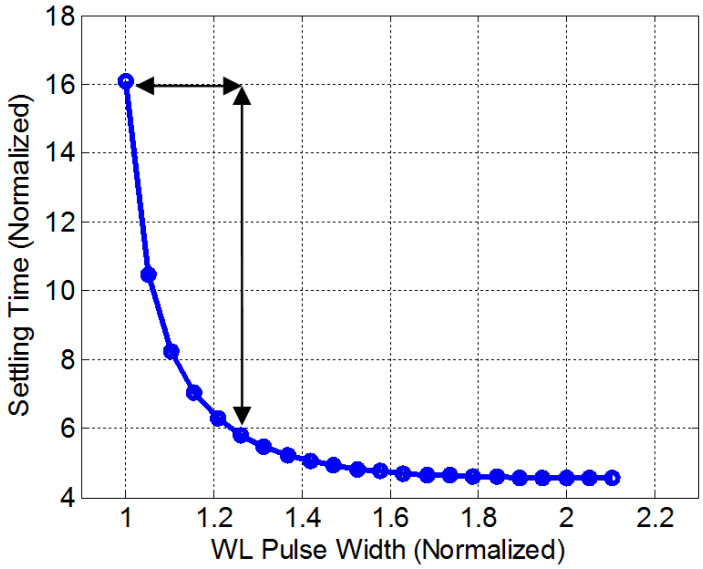


Figure 4.11: Relaxing the WL pulse width requirement reduces the overall cycle time

4.5 Dependence of Cycle time on T_{CRIT}

As the WL pulse width is scaled more aggressively, the time that it takes for the internal nodes of the bitcell to settle increases. It is important that these nodes settle before the following clock cycle because if there is a read immediately following a write, the stability of the cell could be compromised. In a less extreme case, if one of the internal nodes has not pulled up to V_{DD}, the read current of the cell will be reduced which could result in a dynamic read failure. In Figure 4.11, we quantify this trade-off by sweeping the WL pulse width and observing the effect on the settling time of the cell. In this experiment, we define the settling time as the length of time after the WL has gone high before the difference between the internal nodes is $0.9*V_{DD}$. What we observe is that as the WL pulse width is lengthened, the settling time is reduced until it eventually plateaus. This means that if we relax our WL pulse width requirement, we can reduce the overall cycle time. The downside to lengthening the WL pulse width is that it results in an increase in half-select energy due to a larger BL discharge. However, we can see from Figure 4.11 that in some cases a small change in the WL pulse width results in a large reduction in cycle time. In this example, lengthening the

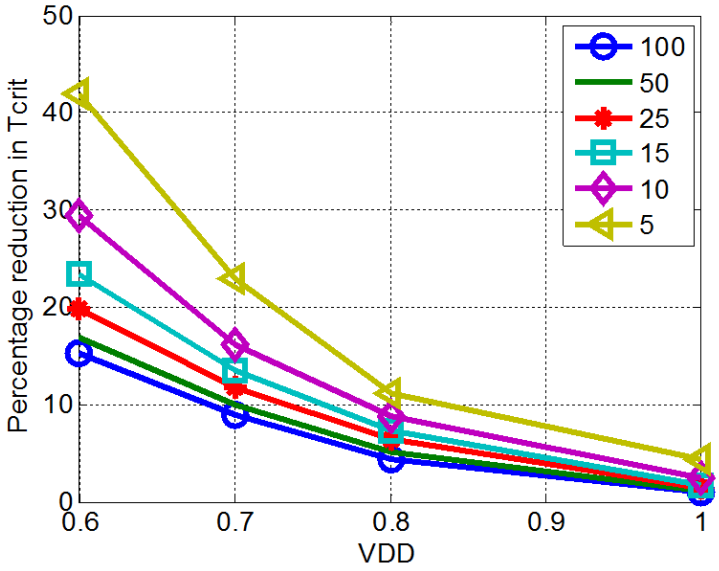


Figure 4.12: Percentage reduction in T_{CRIT} by using negative BL as opposed to WL boosting. Each line represents a different ratio of cycle time/WL pulse width.

WL pulse by 26% results in a 64% reduction in cycle time.

In section 4.4 we observed that negative BL reduction is more effective at reducing dynamic write V_{MIN} compared to WL boosting and that this effect becomes more pronounced at low supply voltages. In this section, we take a closer look at how this trend is affected by the cycle time. Our assumption was that the cycle time was always equal to 2*WL pulse width. Our constraint was that the internal nodes of the bitcell must settle by the end of the cycle. We can see from the plot in Figure 4.12 that once the ratio of cycle time to WL pulse width becomes less than 3, lengthening the WL any further has little effect on the settling time. This explains why the plot in Figure 4.9 shows that WL boosting results in only small reductions in T_{CRIT} across V_{DD}. However, negative BL reduction reduces the settling time of the cell by overdriving the gate of the PMOS devices, thus resulting in a reduction in T_{CRIT}. In order to verify this result was not simply a product of our experimental setup, we tested the two assist methods across a range of cycle times. The plot in Figure 4.12 shows the percentage reduction in T_{CRIT} gained by using 100 mV of negative BL assist versus 100 mV of WL boosting. In order to normalize the cycle time across V_{DD}, we took slices of data at various cycle time/T_{CRIT} ratios. We can see from this plot, that as cycle time is increased,

the percentage reduction in T_{CRIT} decreases. However, at low supply voltages we still see significant (i.e. 10%) reductions in T_{CRIT}. This is due to the fact that at low voltages, the rise time of the PMOS dominates the total settling time of the cell.

4.6 Conclusions

In this chapter, we have shown that modeling the tail of the dynamic write margin using a small Monte Carlo simulation is not effective due to the shape of its distribution. While the static noise margin has been shown in [15] to fit the normal distribution, the dynamic write margin follows a skewed long tailed distribution. We found that at V_{DD}s below 800 mV, the distribution fits the Frechet distribution, however the tail of the distribution can only be determined by full simulation due to poor confidence in the tail fit.

While statistical blockade is a good method for reducing simulation time, evaluating dynamic V_{MIN} still requires a large number of simulations. We introduced a method using sensitivity analysis that provides a speed up over statistical blockade of 112X with an average percentage error of 3%. This approach allows rapid assessment of dynamic write V_{MIN} and write assist features. In addition, we also determined that negative BL reduction has a greater effect on reducing T_{CRIT} than WL boosting. In addition we show that there exists a trade-off between the critical WL pulse width and cycle time. We observe that the advantage of using a negative BL assist versus WL boosting is reduced as cycle time increases; however this advantage is still significant at reduced supply voltages.

Acknowledgments

We would like to thank ARM for their support and for funding this work.

Chapter 5

Virtual Prototyper (ViPro)

[1] As technology continues to scale down, the design of reliable SRAMs has become more difficult due to increases in process variation, leakage, and interconnect delay. Many circuit techniques such as read and write assist methods [22, 23, 27, 29, 66, 67] and alternative bitcells [4–6] have been proposed to improve reliability; however, in order to fully understand the tradeoffs, we must consider their effect on the global Figures of merit (FoMs) (e.g. energy, performance, yield, and area) for the entire SRAM macro. A change in any one memory component can impact the optimal design at both the architectural level and circuit level. We cannot simply innovate in one portion of the memory while ignoring the effects our innovation could have on the overall memory and system design. While designing and optimizing a full SRAM macro provides the most accurate assessment of the global FoMs, it is not efficient to perform a full re-design and optimization each time a new circuit is added. In addition, this approach does not support design space exploration across the technology and application space. Thus, there is a need for a tool flow that supports rapid design space exploration and evaluation of SRAM designs in terms of the global FoMs.

In [68] the authors introduced a tool called ViPro (Virtual Prototyper) capable of performing early design space exploration by creating a virtual prototype of a complete

[1] This chapter is based on the submitted papers under reivew titled: "Virtual Prototyper (ViPro): An SRAM Design Tool for Yield Constrained Optimization" [JB8]

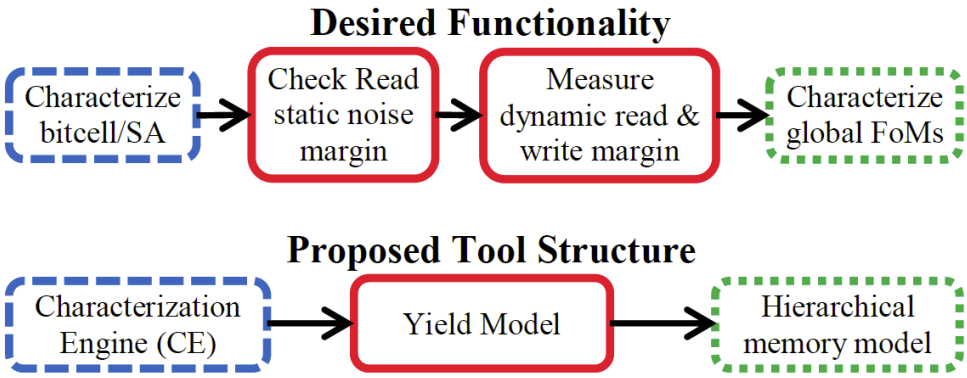


Figure 5.1: High level flow diagram of the desired functionality and tool structure

SRAM macro. The advantage of this tool is that it uses a hierarchical model that allows designers to describe circuits with varying levels of detail (e.g. from as simple as an energy and delay estimate up to a full netlist). Along with the hierarchical model, the tool also includes a characterization engine that can be used with any process technology that has defined SPICE/Spectre device models. This characterization engine includes an expandable library of templates for characterizing SRAM components in terms of energy and delay. While this tool provides a good first order estimate of energy and delay, it does not account for the effect of process variation on the global FoMs. Instead, it provides the energy and delay of the average case, with no variation present. The problem with this is that variation introduces large amounts of timing variation and noise margin spread into the design. Ignoring variation leads to optimistic energy and delay estimates and may also affect the optimal design configuration. In addition, the tool in [68] provides no information on yield, which is a critical metric in SRAM design. Therefore, in order to provide more accurate assessment of the global FoMs, there is a need for a tool flow that takes into account the effect of process variation on energy, delay, and yield.

In this chapter, we will expand the previous functionality of ViPro, by adding yield as a metric for optimization. Figure 5.1 shows a high-level overview of the desired functionality and proposed tool structure. In the proposed flow, the first step is to characterize the bitcell and sense amp (SA) in the presence of variation using a characterization engine (CE). The

next step is to check that the read SNM is robust, before moving on to the dynamic read and write margin. Once the dynamic margin has been measured by the yield model, this data will be plugged into a hierarchical memory model to measure the global FoMs. We will describe the proposed tool structure and flow in more detail in sections III and IV.

This proposed tool flow will not only improve the accuracy of energy and delay calculations for complete memories, but will also allow the tool to take into account the effects of process variation. In addition, it will allow for a tradeoff between yield and performance and energy. In order to accomplish this, we use a combination of simulation and modeling techniques from [2, 15] and Chapter 4 to determine the minimum WL pulse width required for both the read and write operation to meet a user specified die yield as well as the static read V_{MIN}. Plugging this information back into the existing SRAM model, we are able to calculate the energy and delay of a margined design, thus creating a new Pareto optimal curve for a given yield requirement. Previously in [68], the Pareto optimal design space was generated for the average case (not accounting for variation). In this chapter, we are able to generate Pareto optimal curves for margined designs at a user specified die yield.

5.1 Prior Art

There are many tools that support SRAM characterization, but none offer the same level of integrated process-circuit-sytem co-design as ViPro [68]. [69] presents a tool called CACTI for estimating the global FoMs including area, power, and access time. This tool is commonly used by computer architects to evaluate the FoMs across cache size. In [70], this tool is extended to include information about global wiring parasitic energy and delay for muli-bank designs. This enables architectural level design space exploration, however it does not support the evaluation of circuit level optimization. The analytical models used in [69, 70] make fixed assumptions about the circuits that comprise the SRAM design. In ViPro, these circuits can be altered by the user to evaluate their impact on the system level FoMs. In addition, CACTI

uses ITRS [71] (Internation Roadmap for Semiconductors) parameters for estimation of power and access times, which may not be accurate for advanced process nodes [68]. ViPro uses TASE [72] (Technology Agnostic Simulation Environment), which uses technology agnostic simulation templates for energy and delay calculations. This enables SRAM evaluation across any process technology (assuming a device model is available), and ensures SPICE level simulation accuracy. Finally, CACTI does not consider the effect of V_T variation on the global FoMs including die yield. In this chapter, we will extend the current ViPro tool [68] to include die yield as a metric for evaluation.

As shown in Figure 1.5, two other options for design evaluation are transistor level optimizers [73–75], and memory compilers [76]. While circuit optimizers are ideal for quickly evaluating the energy-delay trade-off of a single component, they are not feasible to use for architectural level design exploration. The strength of ViPro is that it can evaluate circuit level optimization on the global FoMs. This is important because it informs the designer of which circuit level knobs have the largest impact on system level performance. Memory compilers are useful for generating complete designs (schematics and layout), however they do not enable design space exploration.

5.2 Background: SRAM Yield Metrics

In this section we will provide a brief overview of the metrics used to quantify die yield. The definition of die yield that we will use throughout this chapter is the percentage of dies with no bit errors.

5.2.1 Static Metrics

The three static metrics used to quantify SRAM yield are: hold static noise margin (HSNM), read static noise margin (RSNM) and static write margin (WM). The HSNM and RSNM are measured using the traditional butterfly curve technique [11]. Static write margin is measured

by setting bitline (BL) and bitline bar (BLB) to 0 and 1, then sweeping the wordline (WL) from 0 to VV_{DD} [13]. The margin is defined as V_{DD} WL voltage when the Q/QB nodes flip. Because the distribution has been shown to closely match the Gaussian distribution, we can estimate read and hold static failures to very low failure probabilities using the method in [15]. We chose not to use static write metrics for reasons outlined in the following subsections.

5.2.2 Advantage of Using Dynamic Versus Static Metrics

Because static metrics assume an infinite WL pulse width, they result in pessimistic read margins and optimistic write margins [62]. Therefore, dynamic metrics [59–64] have been proposed to measure the effect of the WL pulse width on read stability, read access time, and write-ability. In our case, we will look specifically at read access time and write-ability, because we will show using [15] that the bitcell has a very low RSNM failure probability. Our justification for this decision is that if we can show that the cell is not susceptible to static read failures, then it will not suffer from dynamic read stability failures. The reason for this is that static failures assume an infinite WL pulse width; therefore this metric is pessimistic compared to dynamic read stability which assumes a finite noise disturbance.

5.2.3 Dynamic Write Margin

The dynamic write margin is defined as the minimum pulse width required to write the bitcell (write T_{CRIT}) [62]. We choose to use this metric since static metrics for write produce optimistic margins due to the assumption of an infinite WL pulse width. One downside to using transient simulations is that they take much longer to run. Additionally, it is shown in Chapter 4 that the dynamic write margin distribution does not exactly match any known distribution, making it difficult to accurately model the tail of the distribution. In Chapter 4, we proposed a fast, accurate method using sensitivity analysis to predict the worst case write T_{CRIT} for varying memory sizes. This method can be used to accurately predict the WL pulse width required to meet a specified bit failure probability (PF). Using the failure

probability along with the memory size, we can then calculate the probability of chip success as $P_{CHIP} = (1 - P_F)^n$, where n is memory size. Using the binomial distribution we can calculate die yield as the probability that more than k out of N chips pass:

$$CDF = P(X \leq k) = \Sigma_{i=0}^{k} \binom{N}{i} P_{CHIP}^{i} \left(1 - P_{CHIP}\right)^{N-i} \tag{5.1}$$

$$1 - CDF = P(X > k) \tag{5.2}$$

5.2.4 Read Access Time

Read access failures are defined as a failure to generate a large enough BL differential for the read operation to complete correctly. The development of the BL differential is one of the major timing bottlenecks for an SRAM and is determined by the BL capacitance and read current (I_{READ}). The amount of differential required for the read operation to perform successfully is determined by the sense amp offset (σ_{OFFSET}). In order to guard band for the worst case condition, it is common for designers to pair the worst-case σ_{OFFSET} with the worst-case I_{READ}. In reality, it is not likely that the worst-case bitcell of a large memory will be in the same column as the worst-case sense amp [2]. By using this overly-conservative method, designers sacrifice potential performance gains. In [2], the author uses order statistics to calculate the critical WL pulse width (read access T_{CRIT}) required to meet a specific die yield. [2] shows that matching worst case σ_{OFFSET} with the worst case I_{READ} for that column provides a more accurate model of read access yield. In addition, this model allows designers to make a direct tradeoff between performance and yield.

5.3 Proposed Tool Flow

The proposed tool flow is outlined in Figure 5.2. Throughout this section, we will describe the tool under the assumption that none of the yield model parameters have been specifically

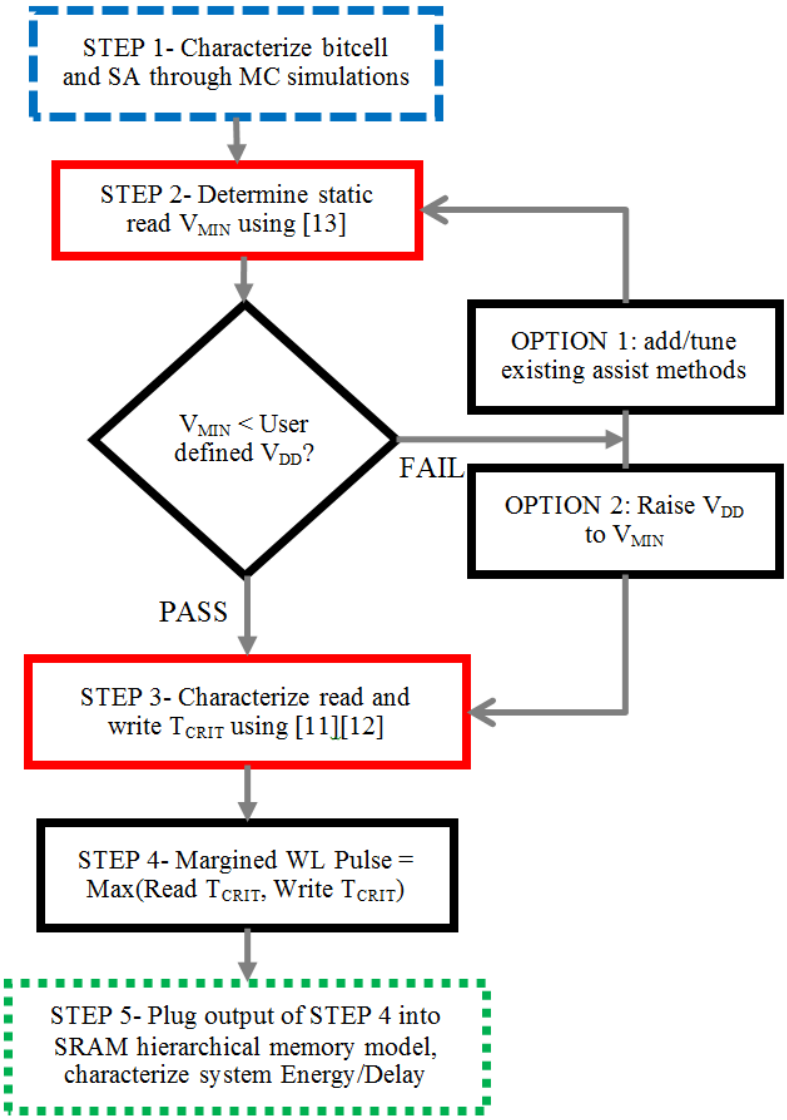


Figure 5.2: Flow diagram for performing yield constrained optimization

defined by the user.

5.3.1 Determining Static Read V_{MIN}

STEP 1 in our yield constrained optimization tool flow is to characterize the static noise margins of the bitcell. For each Monte Carlo iteration, we measure the noise margin for reading/writing a 0 and a 1 (SNM0 and SNM1). The resulting margin is the minimum of these two calculations. [15] found that SNM0 and SNM1 are normally distributed as well as

negatively correlated. Therefore the SNM distribution can be accurately modeled as a joint probability distribution function. Using the statistical model from [15] we can accurately predict static failure probabilities (PF) across a range of V_{DD}. Equations (5.1) and (5.2) can then be used to determine the PF that satisfies the given die yield constraint. STEP 2 then plugs this failure probability back into the model from [15], which tells us the minimum supply voltage (V_{MIN}) required to meet the yield constraint. If this supply voltage is less than the user defined supply voltage then the bitcell passes the read SNM yield test. If it is greater, then the user can either choose to add an assist method to improve read stability (OPTION 1) or boost the supply voltage up to V_{MIN} (OPTION 2).

5.3.2 Characterizing Read and Write T_{CRIT}

Once the bitcell has passed static noise margin tests, STEP 3 is to determine the critical WL pulse width (T_{CRIT}). In older technology generations, the read operation has typically set this pulse width, however it has been shown that newer technologies are more write limited due to the variation of the minimum sized PMOS device [17]. The critical read and write WL pulse width is determined in STEP 3 using the methods outlined in [2] and Chapter 4 respectively. In the case of write T_{CRIT}, this value is based solely on the bitcell sizing and the target yield. The read access T_{CRIT} is dependent on not only the bitcell sizing (which sets I_{READ}) and target yield, but also on the BL capacitance per bitcell, number of bitcells per column, number of columns per SA, the SA offset, and the total number of sense amps. Because the read access model is able to quickly estimate the read T_{CRIT} value for a given configuration, we chose to calculate this value for every possible SRAM configuration and store this data in the optimization engine.

5.3.3 Energy and Delay Characterization

After receiving the read and write pulse widths, the tool chooses to use the maximum of these two values in STEP 4. Our assumption is that the WL pulse width for reading and

writing is the same, so in order to meet the yield constraint we must chose the larger of the two pulse widths. In the previous implementation, the read and write WL pulse widths were optimized separately through simulation of the average case (no V_T variation). Once the WL pulse width has been set, STEP 5 plugs this value into the SRAM hierarchical memory model, which then calculates the energy and delay of the full macro.

5.4 Tool Structure

Figure 5.3 shows the complete structure of the yield-aware ViPro, which is similar to what was presented in [68]. The main contributions of this work are the yield model and tool flow for supporting yield constrained optimization. The hierarchical memory model has been implemented as a class structure in C++ and is used to describe the virtual prototype. The characterization engine (CE) contains technology agnostic templates for characterizing the energy and delay (E/D) of individual SRAM components as well as Monte Carlo (MC) templates for characterizing I_{READ}, sense amp offset, dynamic write margin, read SNM and hold SNM. Full descriptions of the hierarchical model and CE can be found in [68]; therefore we will only provide a brief overview below. The yield model contains the framework for the models described in the previous section for calculating read and write T_{CRIT}, as well as the static read V_{MIN}. The optimization engine contains both a brute force optimization algorithm as well as a simulated annealing algorithm for optimizing large design spaces. Because it is not a focus of this chapter, the simulated annealing algorithm will not be described in detail.

5.4.1 Hierarchical Memory Model

The two main functions of the hierarchical memory model are to store the virtual prototype and calculate the global FoMs. The class hierarchy has been implemented in C++ in order to interface with the optimization algorithms. The top level SRAM class contains the global parameters such as memory capacity, supply voltage, word size, bank partioning, and flags

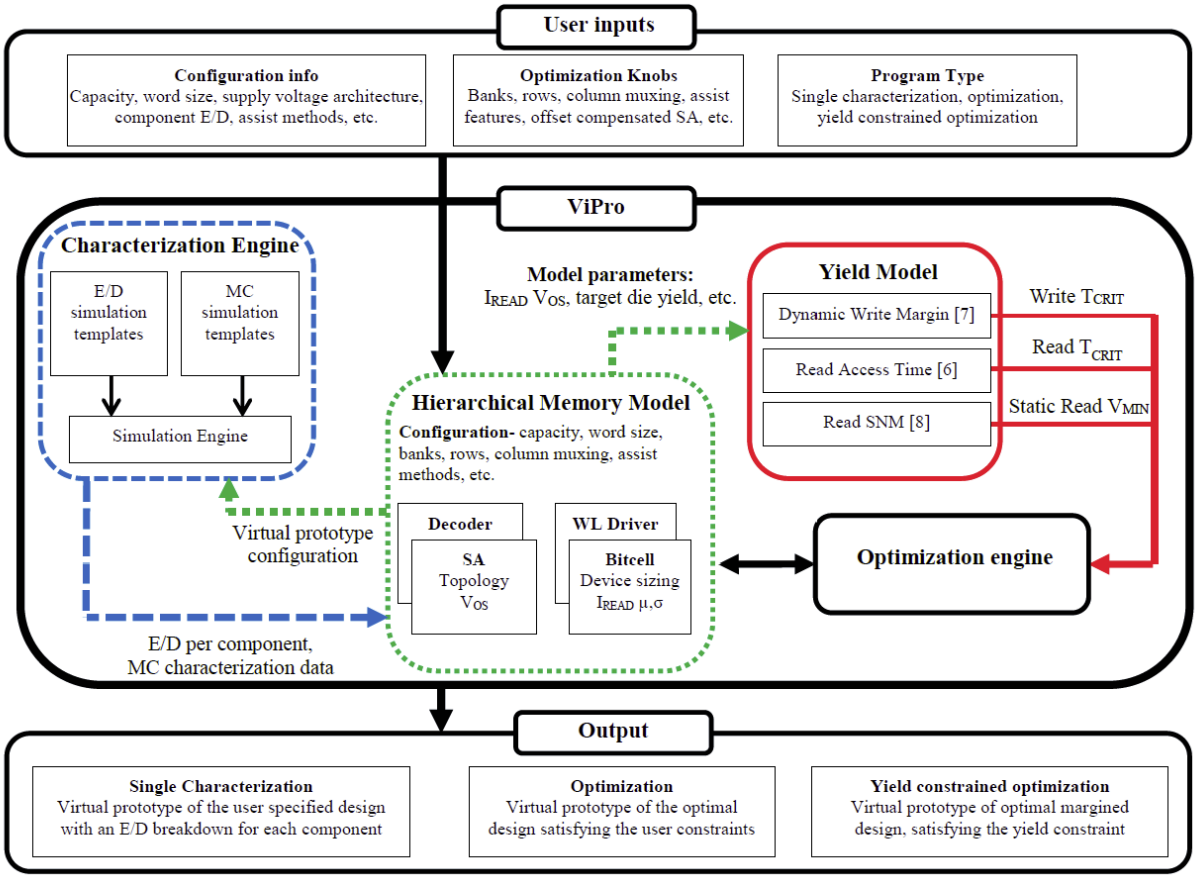


Figure 5.3: Block diagram showing the structure of ViPro. The yield modeling is the additional feature which takes its parameters from the SRAM model and outputs the critical read and write WL pulse widths, as well as the minimum supply voltage necessary to meet the yield constraint.

for assist methods. It also contains energy, yield, and delay constraints. These parameters and constraints can be set by either the user inputs or the optimization engine. The global parameters are inherited by each component class, which additionally contains its own local parameters. For example, the sense amp class has parameters such as offset, topology type, energy, delay, etc. which are specific to that component. Once the virtual prototype is defined, components with fully defined netlists are characterized by the CE, which returns the energy and delay of the component. Once all of the components have been defined, the model calculates the global FoMs and either reports back to the optimization engine or outputs the results to the user. Two advantages of using a hierarchical model are: that it is easily

extensible and it is able to capture the interactions between the various components.

5.4.2 Characterization Engine (CE)

The CE contains a library of simulation templates that can be used for characterizing a technology (e.g. I-V curves, leakage current, inverter delay), characterizing the energy and delay of an SRAM component, or characterizing yield specific parameters through MC simulation such as I_{READ}, σ_{OFFSET}, read and hold SNM, and dynamic write noise margin. The configuration and technology specific parameters must be passed to the CE by the memory model before the simulation can be executed. After simulation, the data is then sent back to the memory model and stored in the virtual prototype.

In this chapter, the characterization engine is used to calculate the average energy (not considering local mismatch) and the delay of the full macro. The CE contains full SPICE netlists of each SRAM component (e.g. bitcell, WL drivers, BL drivers, SA, etc.), including the output load. The energy and delay of each component is measured separately through simulation, including the leakage energy of each array. The read and write energy are measured in separate simulations, and the total energy, including local and global signal buffering, is calculated as the sum of each component. The energy and delay of the active bank have been verified against a full schematic in [77]. Results show that for a 512 row by 16 column array, the model is accurate to within 14% in terms of delay and within 19% in terms of energy. In order to account for the energy and delay of the global lines, the wires are modeled as an RC network using parameters from the technology design manual. The length of the wires is determined using the measured layout area of the bitcell. For designs with a large number (e.g. > 4) of banks, the global interconnect is routed in an H-Tree to reduce propagation delay. The characterization engine considers the bank dimensions when placing the banks, to reduce the total length of the global lines.

5.4.3 Yield Model

The yield model contains the framework for executing the models described in [2, 15] and Chapter 4. The parameters for each yield model are sent from the memory model. Each yield model takes as an input the target yield and memory size. The other model specific parameters are described in Table 5.1. The output of each model is shown in Figure 5.3. For the two dynamic metrics, the output is the minimum WL pulse width required to meet the specified die yield constraint. The output of the static read V_{MIN} model is the minimum operating voltage required to meet the die yield constraint. The addition of this model allows ViPro to create virtual prototypes that are specifically designed to meet a die yield constraint set by the user. This results in improved accuracy of energy and delay calculations for complete memories and also allows the tool to take into account the effects of process variation. In addition, it enables ViPro to make a trade-off between yield and energy and performance.

Table 5.1: A list of the input parameters for each of the yield models

Model Type	Parameters
Dynamic Write Margin (Chapter 4)	Sensitivity curves for each bitcell device
Read Access Margin [2]	$I_{READ}(\mu, \sigma), \sigma_{OFFSET}$, BL capacitance per bitcell, number of bitcells per column, number of columns per SA, total number of SA
Static Read V_{MIN} [15]	RSNM(μ, σ) across V_{DD}

5.5 Results from the Characterization Engine

In this section, we break down the global energy and delay of a fixed size (16 Kbits) SRAM design in order to show the architectural level trends. We look specifically at read delay

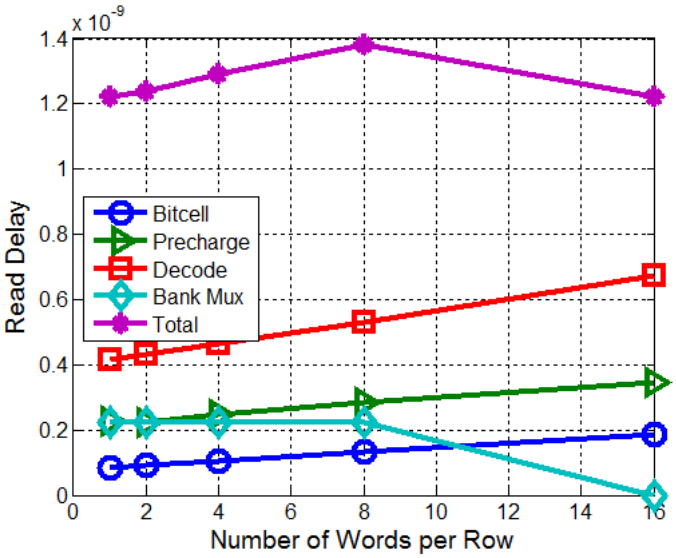


Figure 5.4: Read delay for a fixed number of rows (64), and an increasing number of words per row

because it is the worst case operation (e.g. the read delay > write delay, therefore it sets the cycle time of the memory). In addition, we look at the write energy because it dominates the total active energy. The ability to evaluate these trade-offs across any range of memory size and technology node are what makes ViPro such a powerful tool.

5.5.1 Read Delay

The plot in Figure 5.4 shows a breakdown in the delay by component for an increasing level of column muxing (e.g. number of words per row) and a fixed number of rows. As the number of words per row increases from 1 to 16, the number of banks decreases from 16 to 1. This results in a reduction in the wordline capacitance, resulting in lower WL driver delays. In addition, the load on the precharge driver decreases, resulting in reduced precharge delays. However, the one advantage that the single bank design has over the multibank design is the lack of a bank mux delay. By going to a multibank scheme, we pay this cost of increasing the logic depth, by adding another gate in the critical path. However, as seen from the plot, the 16 bank design achieves a slightly smaller read delay than the single bank design.

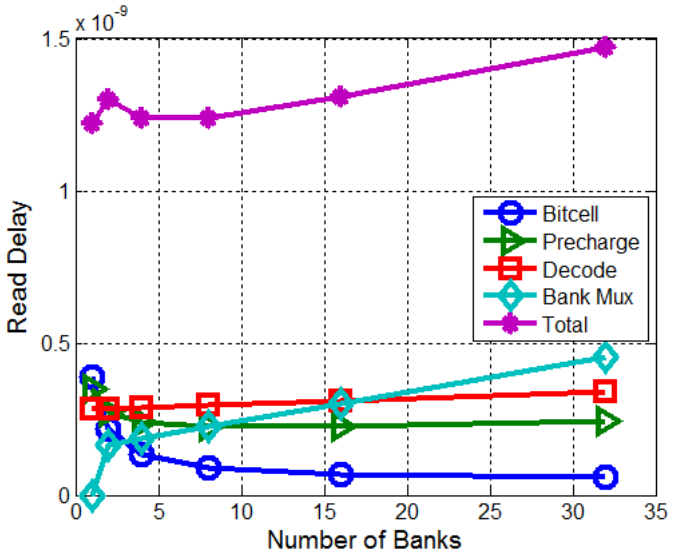


Figure 5.5: Read delay for a fixed level of column muxing (2) and an increasing number of banks

Figure 5.5 plots the read delay for each component as the number of banks increases. The level of column muxing remains constant, so as the number of banks increases, the number of rows decreases. The reduced number of rows results in lower BL capacitance which yields reduced bitcell delay and reduced precharge delay. However, the reduction in precharge delay does not have an effect on the total delay because it is less than the decoder and WL driver delay. Since these occur in parallel, the decoder delay sets the critical path. The decoder delay reduces as the number of rows decreases, due to a reduction in the logical stages within the decoder. The critical component for this test case is the bank mux. As the number of banks becomes large, the output capacitance on the bank mux (a tristate buffer) becomes larger due to the increasing wire length. This results in a sharp increase in the bank mux delay. However it is likely that re-optimizing the sizing on the output mux will result in improved read delay measurements.

5.5.2 Write Energy

Figure 5.6 shows the write energy breakdown by component for two words per row, with the number of banks increasing. As the number of banks increases, as does the number of

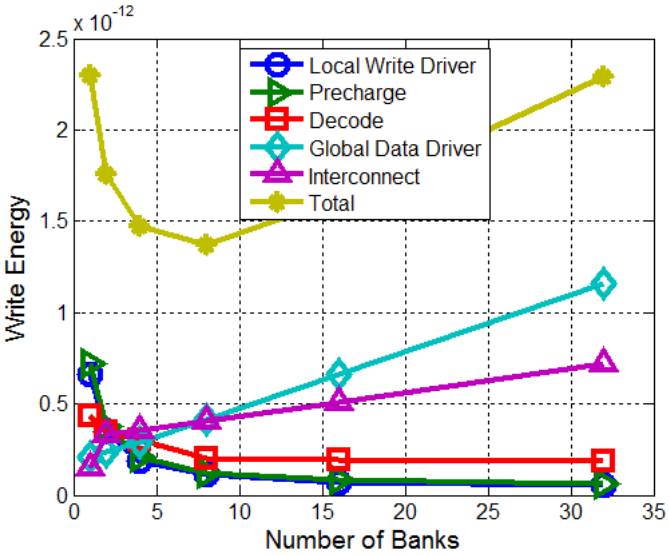


Figure 5.6: Write energy for a fixed number of words per row (2). This plot shows the trade-off between global and local interconnect energy

rows. This results in a reduction of the write driver and precharge energy, as the bitline capacitance is reduced. In addition, the decoder energy is reduced as the number of address bits decreases. The bitcell energy is not shown because it stays constant. This is due to the fact that bitcell energy is mostly dependent on the number of half selected cells discharging during the write operation. Because the number of banks increases, the length of the global interconnect increases, which in turn increases the total interconnect energy. The data driver plays a major role in determining the best case configuration. Because the interconnect capacitance increases, the energy dissipated by the flip flop transmitting the data to the local write bitlines increases. This shows that the tradeoff for reducing the length of the global interconnect lines is higher capacitance per bank. In order to optimize the global write energy, these combination of these two metrics needs to be minimized.

Figure 5.7 shows the write energy breakdown for a fixed number of rows and an increasing words per row. The interconnect energy decreases from 8 to 16 words per row (2 banks to 1 bank) because the because of the introduction of the bank select signals. It then decreases slightly because as the number of words per column decreases, the load on the local precharge signal also decreases. The same is true for the word line driver energy. Bitcell energy has the

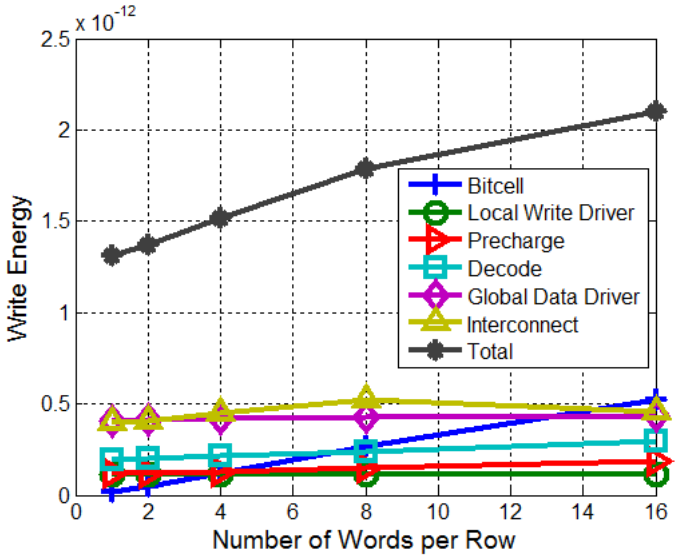
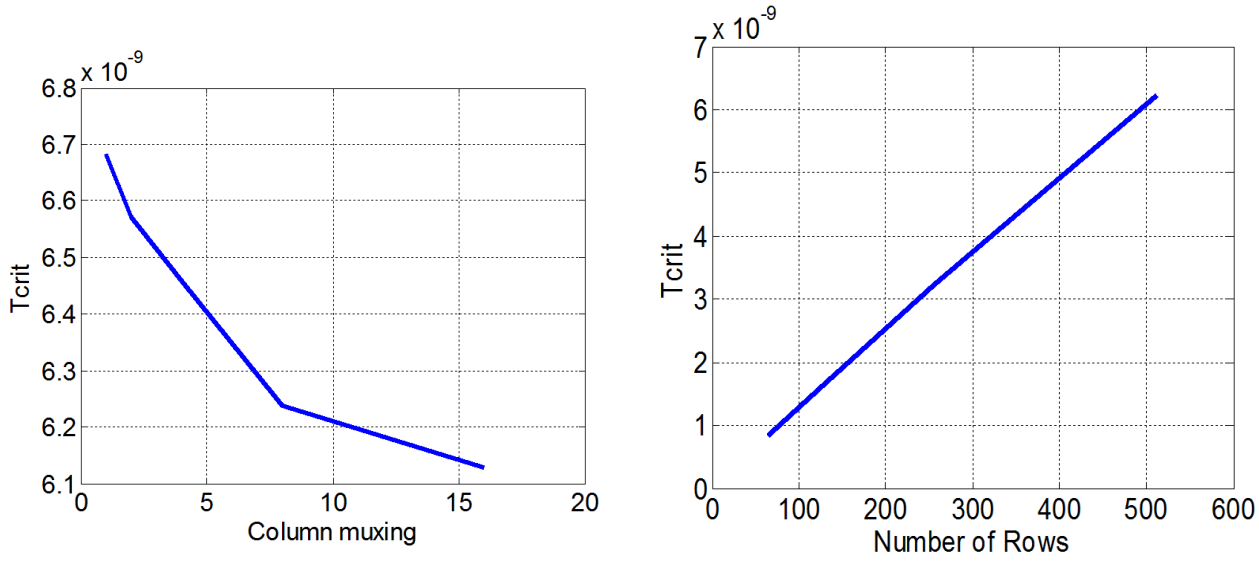


Figure 5.7: Write energy for a fixed number of rows, and in increasing number of words per row

largest influence on total energy. As the number of words per row increases, the number of half selected cells also increases. This means that more bitlines are being discharged through the bitcell, leading to much higher energy. This plot shows that the best way to reduce write energy is to reduce the level of column muxing.

5.6 Results from the Yield Model

This section shows results from several experiments that were performed using the yield model (STEP 3 from Figure 5.2). These results will highlight the trends for the critical read and write WL pulse width across memory size, temperature, and yield. Additionally we show the impact of WL boosting on read and write T_{CRIT}. For the write T_{CRIT} measurement, we define T_{CRIT} as the WL pulse width required for the difference between Q (initially 0) and QB (initially 1) to equal V_{DD}*0.9. This measurement is taken at time 2*(WL pulse width), which assumes a 50% duty cycle on the WL. This constraint was chosen arbitrarily and can be modified to account for different timing requirements.



(a) (b)

Figure 5.8: (a) performance gains using column muxing (b) performance performance gains through BL capacitance reduction

5.6.1 Column Muxing vs. BL Capacitance Reduction

[2] showed that increasing the number of bitcells per sense amp through column muxing caused more bitcell+SA failures to cluster in a single column, thus increasing die yield. In order to quantify this effect, we compared the benefits of column muxing vs. BL capacitance reduction. In Figure 5.8a, we measured the read access time of a 16 Kb memory with a fixed number of rows in Figure 5.8b with a fixed number of banks. The results show that moving from a design with no column muxing to a 16:1 mux produced an 8.2% reduction in read access T_{CRIT}. From Figure 5.8b we can see that the relationship between the number of cells per bitline and T_{CRIT} is approximately linear. This is expected since I_{READ} is dependent on the V_T variation of the bitcell and the BL capacitance increases linearly as more cells are added. By cutting the number of rows in half, the result is a nearly 50% reduction in T_{CRIT}. Therefore, while column muxing does result in performance gains, BL capacitance is a much more effective knob.

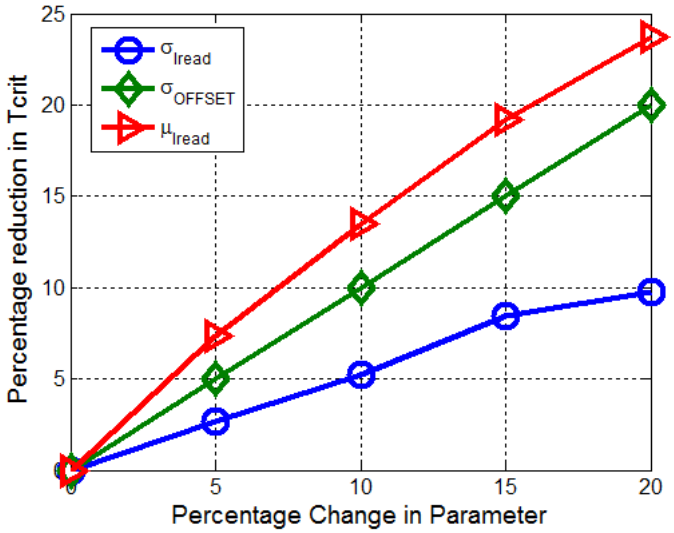


Figure 5.9: Comparing the sensitivity of read access time to the three statistical parameters: μ_{Iread}, σ_{Iread}, and σ_{OFFSET}

5.6.2 I_{READ} vs. Sense Amp Offset

In order to generate optimal SRAM designs, it is important to understand the effects that design parameters have on the global figures of merit. Figure 5.9 plots the percentage reduction in three measured design variables ($\mu_{Iread}, \sigma_{Iread}, \sigma_{OFFSET}$) and the corresponding percentage reduction in T_{CRIT}. These three parameters are measured in STEP 1 of Figure 5.2. In order to reduce read T_{CRIT}, σ_{Iread} and σ_{OFFSET} should be reduced. The most common way to reduce these parameters is to increase the device size, since V_T variation is proportional to the channel area. In the bitcell, σ_{Iread} could be reduced by upsizing the pull-down and pass-gate devices. Increasing μ_{Iread} results in reductions in read access time. This can be accomplished by using a read assist method such as WL overdrive to increase the on current of the pass-gate device. Figure 5.9 shows that increasing the average read current has the most pronounced effect on reducing read access time. It is interesting to note that reducing σ_{OFFSET} results in a directly proportional reduction in read access time (e.g. a 17% reduction in σ_{OFFSET} results in exactly a 17% reduction in T_{CRIT}).

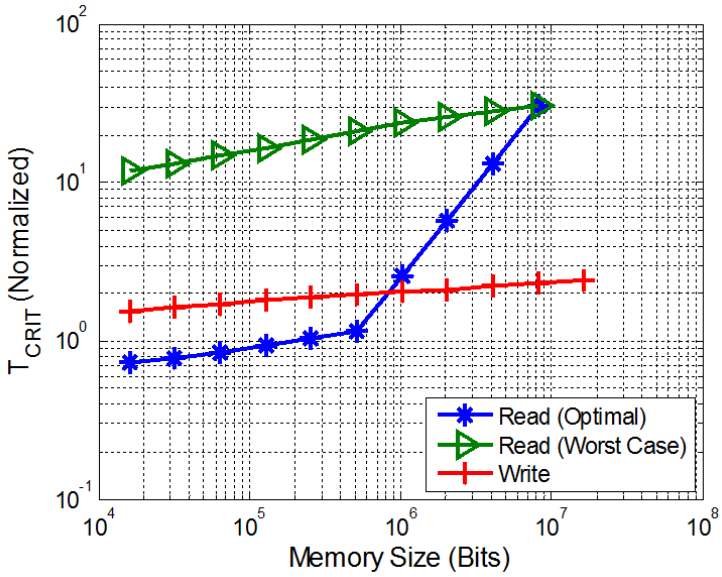


Figure 5.10: Read and write critical WL pulse width vs. memory size at a fixed die yield of 95%. In the case of read, the WL pulse width is shown for both the optimal and worst case macro configuration

5.6.3 Memory Size vs. T_{CRIT}

As memory size increases, the write bit failure rate must also decrease in order to maintain a constant target die yield. The write failure rate is only dependent on the WL pulse width, therefore we expect a steady increase in write T_{CRIT} as is shown in Figure 5.10. The read access time however is dependent on the total number of SAs, number of bitcells per sense amp, and BL capacitance. The assumptions that we make in Figure 5.10 are that we are limited to between 32 and 512 rows, column muxing up to 16 words per row, a maximum bank number of 32, and a target die yield of 95%. We can see from Figure 5.10 that for smaller designs (e.g. $<$ 1Mb), the optimal macro design offers an order of magnitude improvement in the read T_{CRIT} over the worst case design. In this experiment, the worst case design has the maximum BL length (512 cells per column) and the optimal design has minimum BL length, which varies with design size. As the memory size rises, the optimal read access time increases at an average rate of 9.37% per 2X increase in capacity. Once the capacity exceeds 500 Kb, the rate of change increases to 128% per 2X increase in capacity. This rapid

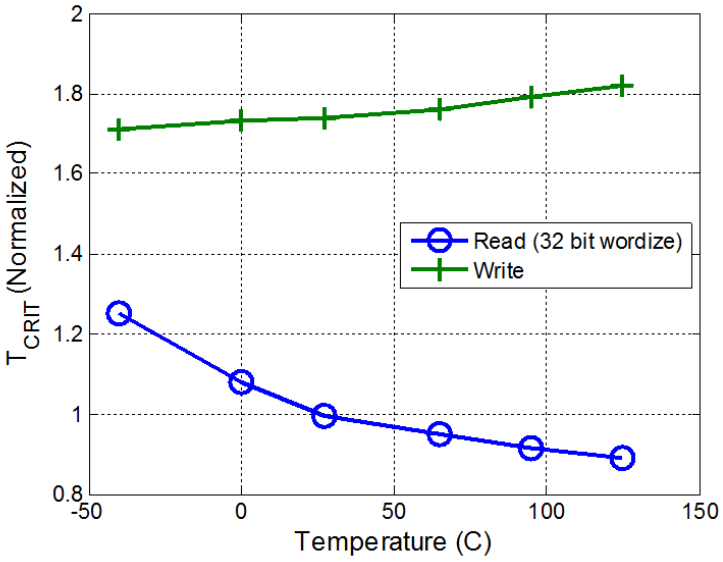


Figure 5.11: Read and write critical WL pulse width vs. temperature

increase in T_{CRIT} is caused by the bitline length doubling in each successive iteration due to the fact that the design has reached the maximum level of column muxing and number of banks. Figure 5.10 shows that designs with 64 or more rows are read-limited (read operation sets the WL pulse width), while designs with BL lengths of 32 are write-limited. Once the memory has reached the maximize size (8 Mb), due to the constraints we defined earlier in this section, the optimal design has the same configuration as the worst case design. Using the yield model, we are able to quickly calculate the read and write T_{CRIT} of every possible configuration, which enables this type of experiment.

5.6.4 Trends Across Temperature

Tracking T_{CRIT} across temperature is important because this information can be used to adaptively tune assist methods to prevent bit failures. Since assist methods consume extra energy, a mechanism for tracking failure probabilities could help reduce that cost. We can see in Figure 5.11 that as temperature increases, the write performance degrades, while read performance improves. One thing to note is that read performance is typically traded off for read stability, so at higher temperatures this stability is degraded. Therefore, implementing

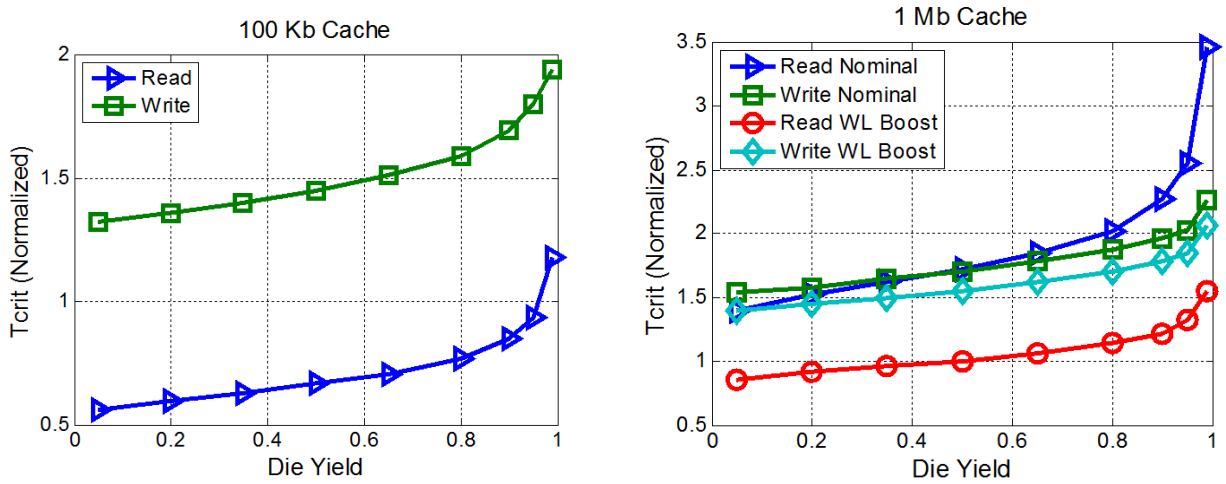


Figure 5.12: trade-off of read and write critical WL pulse width vs. die yield for a 100 Kb cache (a) and 1 Mb cache with and without WL boosting (b)

a write assist using a boosted WL at higher temperatures is not a good option since it will further degrade the read stability of half-selected cells. Instead, a negative BL scheme could be implemented since it does not affect the stability of unselected columns.

5.6.5 Yield vs. T_{CRIT}

In Figure 5.12a and 5.12b we characterize the trade-off between yield and performance. This plot shows that as we approach 99% chip yield, write and read T_{CRIT} rise dramatically. A reduction from 99% yield to 95% results in a 26.4 and 21.1 percent reduction in read T_{CRIT} in the 1Mb and 100 Kb cache respectively. However, since the 100 Kb cache is write limited, it only sees a 7.18% total reduction in T_{CRIT}. Moving down further to 80% chip yield results in a 41.5% reduction in T_{CRIT} for the 1 Mb cache and 18.0% for the 100 Kb cache. Once the curve hits 80% yield it levels off to an average of 3.47% reduction in T_{CRIT} for every 10% loss in yield. It is also interesting to note that at 45% and below, the 1 Mb cache is write limited, while above 45% it is read-limited. This trade-off between yield and performance is enabled by the addition of the yield model to the tool-flow.

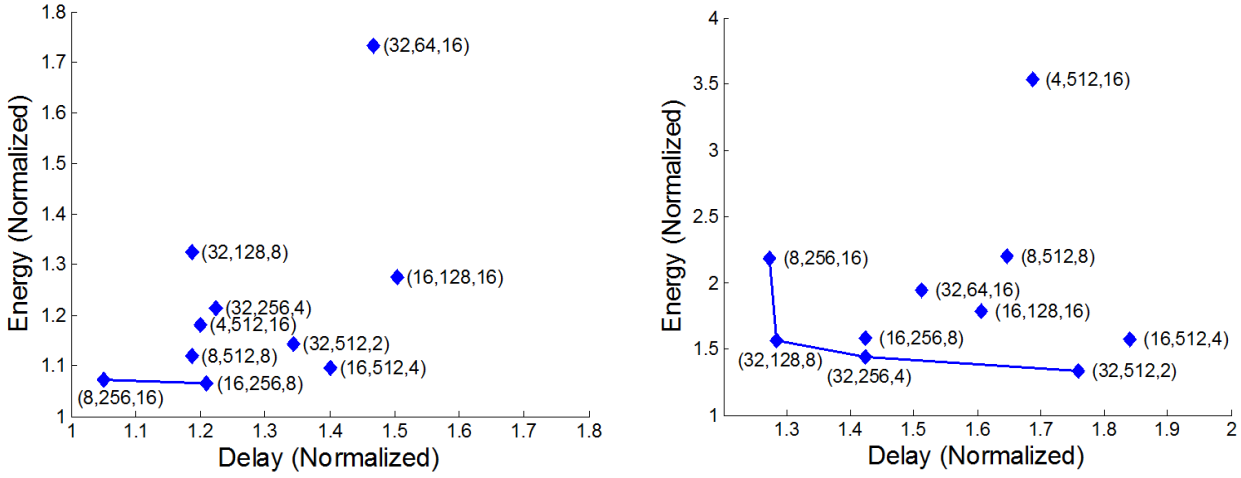


Figure 5.13: Results from ViPro for a 1 Mb memory. (a) average case, (b) 95% die yield. Annotation format- (number of banks, number of rows, words per row)

5.7 System Level Optimization

5.7.1 Average Case vs. Yield Constrained Optimization

In this section we compare the optimization results for the average case (not accounting for V_T variation) and the yield constrained case. As expected, when we account for V_T variation by margining the WL pulse width, both the energy and delay increase. Delay increases simply due to the fact that we are significantly lengthening the WL from the average case. The energy increases because a longer WL pulse width results in discharging the BLs to a lower voltage during a read. This also affects the write energy due to an increase in the half-select energy. Figure 5.13a shows the results from ViPro for a 1Mb SRAM, not accounting for V_T variation, while Figure 5.13b shows the same results 95% die yield. We can see from this Figure that the Pareto optimal points shift to higher energy and delay. Additionally, the difference in energy and delay between optimal and non-optimal points increases when constrained by die yield. In the average case, the global interconnect delay and energy dominates the system level energy and delay, consuming on average 40% of the system level energy and delay. Once variation is accounted for and the WL pulse width is lengthened,

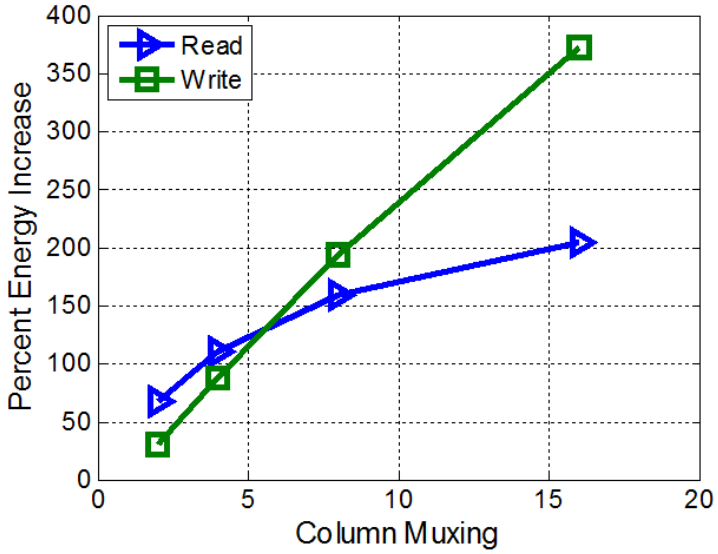


Figure 5.14: Percentage energy increase from the average design (no V_T variation) to the 95% die yield optimized design

energy increases 1.5-4x as shown in Figure 5.14, however the interconnect energy and delay remains the roughly the same due to the large size of the buffers. As shown in Figure 5.13b, designs that were initially dominated by interconnect energy and delay (and therefore not on the original Pareto curve) shift down onto the curve because the increase in energy and delay of the bank outweighs the overhead of the global interconnects.

Figure 5.14 shows the trends for percentage increase in macro level energy between the average design, with no V_T variation, and the 95% yield optimized design. Figure 5.14 assumes a fixed number of rows (512) with the number of banks and columns being swept. As the WL pulse width is lengthened, the amount of energy discharged by the half-selected cells increases. This increase in energy is multiplied by the number of half selected cells. Therefore in designs with a large number of words per row, the percentage increase in energy is higher than in designs with fewer words per row. The rate of increase for the read operation is less than write due to the fact that in the nominal design, the read and write WL pulse width are optimized separately. In the yield constrained case, we set the WL pulse width equal to the larger of the two T_{CRIT} values calculated by the yield model (Figure 5.2) in order to ensure that both operations meet the yield constraint. In the average case, the write WL

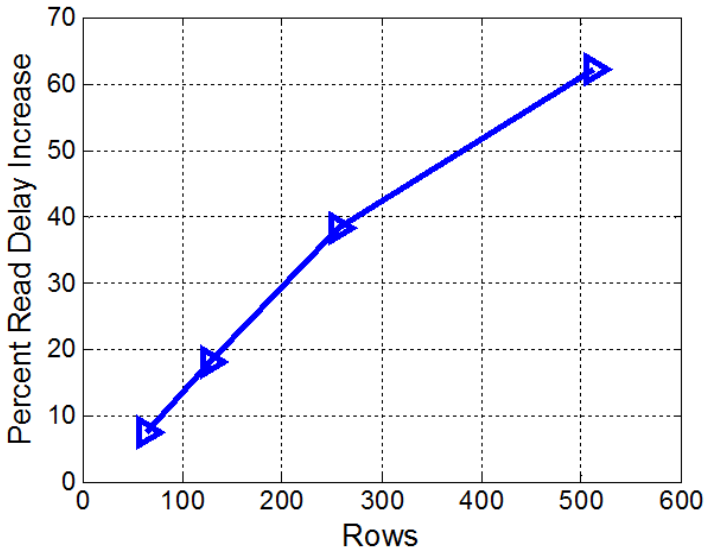


Figure 5.15: Percentage delay increase from the average design (no V_T variation) to the 95% die yield optimized design

pulse is considerably shorter because VT variation is not taken into account. Therefore the percentage increase in the write WL pulse width is much larger than the read WL pulse width. This larger increase in WL pulse width leads to a larger increase in half-select energy, which increases with the number of half selected columns.

Figure 5.15 assumes a fixed number of words per row (16) with the number of rows and banks being swept. The percentage increase in the read delay increases with the number of rows due to the fact that the WL pulse width accounts for a larger percentage of the total read delay in designs with large BL capacitances. Therefore, even though the WL pulse width approximately doubles when moving from 256 to 512 rows, the percentage read delay increase in the design with 512 rows is larger because the WL pulse width accounts for a larger percentage of the total read delay. The addition of yield as a constraint to the tool-flow allows for the effect of V_T variation on macro-level FoMs to be quantified as shown by these results.

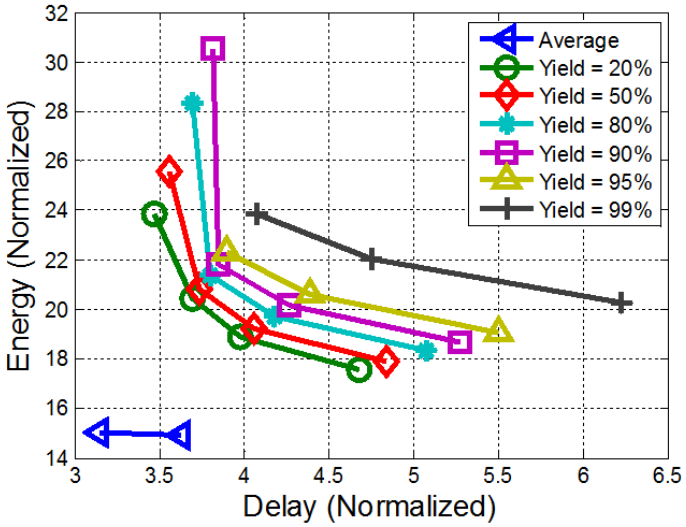


Figure 5.16: Plot of Pareto optimal points across varying die yields

5.7.2 Energy and Delay Pareto Curves Across Yield

In order to quantify the performance versus yield trade-off shown in Figure 5.12a and 5.12b, we ran the global optimization for the entire SRAM macro for varying target yields. Shown in Figure 5.16 are the Pareto optimal points for each case. The average and 95% die yield Pareto optimal curves are the same points from Figure 5.13b. This plot shows that as the target yield is reduced, both the energy and delay of the optimal design decrease. As the WL pulse width is reduced more aggressively, yield, delay, and energy are all reduced. We can see from this plot that even at only 20% die yield, the energy and delay of the optimal designs are still 27% and 16% higher than the average case.

5.7.3 Comparison of Designs with Assist Methods

Typically when reporting on assist methods, authors cite the energy overhead associated with adding new circuitry, such as a charge pump, to boost the WL above V_{DD}. In this section we compare the energy and delay of two 1Mb designs at a fixed die yield of 95%. One design uses a charge pump [27] to boost the WL 100 mV above V_{DD} and the second uses a standard WL driver at nominal V_{DD}. Figure 5.17 shows the Pareto optimal curves of the two SRAM

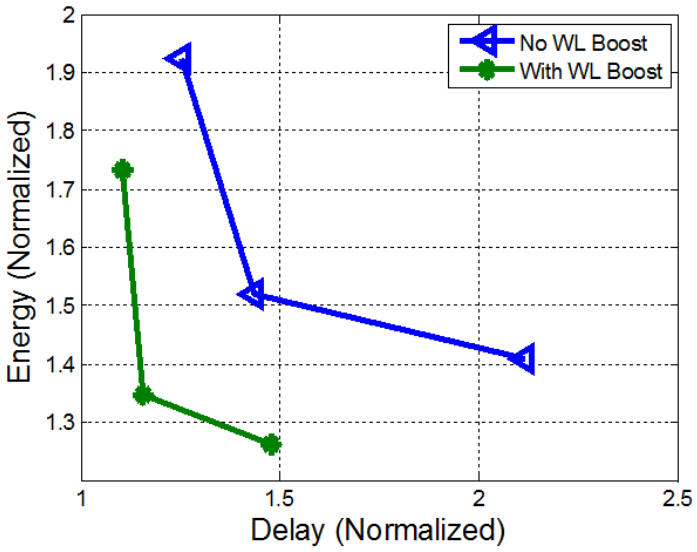


Figure 5.17: Comparison of the Pareto optimal points of a 1Mb design with a WL boosting scheme vs. no WL boosting at a die yield of 95%

macro level designs. It comes as no surprise that the addition of WL boosting reduces the delay of the Pareto optimal points; however, the results show that the total energy is also reduced.

WL boosting increases I_{READ}, which significantly reduces the length of the WL pulse width. When the critical WL pulse width is calculated and plugged back into the hierarchical SRAM model (steps 3-5 in Figure 5.2), the model measures the energy for the average case (no V_T variation). Simulation results show that using the boosted WL results in a 27% reduction in the average BL dissipation for read and for the half-selected cells during write. This means that, on average, less charge is being dissipated by the bitcells, and subsequently less energy is used to precharge the BLs back to V_{DD}. This result is counter-intuitive since normally we think of assist methods resulting in an increase in energy.

We can explain this result by the plot in Figure 5.18. In this simulation, the read delay was measured as the WL pulse width required to generate 150 mV of BL offset. We can see from this plot that as the level of WL boosting increases, both the and of the read delay distribution decrease. For the example shown in Figure 5.17, σ is reduced from 16 ps to only 10 ps. Thus, when margining the read WL pulse, the worst case is not as far out on the tail

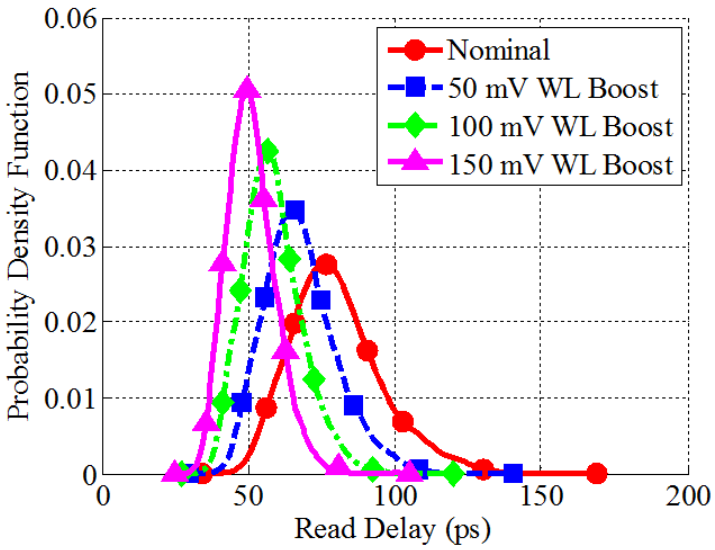


Figure 5.18: As the level of WL boosting increases, both the and of the read delay distribution decrease. This explains why WL boosting saves energy at the macro level in Figure 5.17

of the distribution from the average case. This leads to the overall energy savings shown in Figure 5.17. This example highlights the value of the yield-aware ViPro tool and its ability to provide helpful insight to designers.

5.8 Conclusions

In this chapter we have extended the capabilities of ViPro, first presented in [68] to include die yield as a constraint. The tool flow present in this chapter not only improves the accuracy of energy and delay calculations, but also allows the tool to take into account the effects of process variation. In addition, it allows for a trade-off between yield, performance, and energy. By using a combination of simulation and modeling techniques, we are able to create virtual prototypes that are margined to meet die yield requirements set by the user. The tool structure described in this paper allows for comparison across different array topologies, process technologies, and circuit choices including assist methods. Our results from the yield model show the trends of dynamic read and write margin across temperature, memory size, and die yield. Using this tool, we show that adding a wordline boosting scheme results in

an overall energy savings, despite the overhead of using a charge pump circuit, due to an improved read delay distribution.

Acknowledgments

We would like to thank the Semiconductor Research Corporation (SRC) for sponsoring this work.

Chapter 6

Canary-Based PVT Tracking System for Reducing Write V_{MIN}

As shown in Chapter 4, reducing write V_{MIN} has becoming increasingly challenging due to increases in process variation. Local mismatch due to random dopant fluctuation is one of the major contributors to SRAM reliability [78–80]. However, other sources of variation such as global process, voltage, and temperature (PVT) variation must also be accounted for. These sources are especially important in commercial designs, where die yield is a major concern. As shown in Figure 6.1, margining a design for the worst case condition leads to higher operating voltages relative to the typical case, leading to higher energy. In this example, margining for the worst case corner results in a 0.793V increase in V_{MIN} relative to the typical corner (TT 27 °C). Because the circuit is not always operating in the worst case PVT corner, there is a potential to regain some of this lost energy. If the design could adapt to changes in the PVT corner, instead of setting the operating voltage at design time, then the energy overhead of conservative guard-banding could be reduced.

Variation in operating temperatures effects the transistor current characteristics. In super-threshold, high temperatures reduce on currents (I_{ON}) leading to a reduction in performance [82]. The typical commercial operating temperature specification is 0 °C to

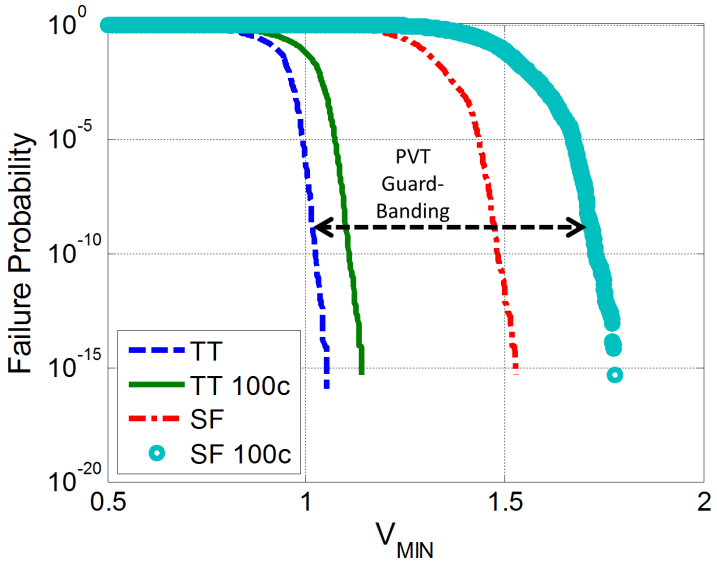


Figure 6.1: Accounting for the various sources of variation results in an increase in SRAM V_{MIN}

85 °C, while the industrial specification ranges from −40 °C to 100 °C [81]. In addition to temperature variation, designers must also account for variation in the power delivery network. Designs are typically margined to operate reliably under voltage variations up to +/-10%. Finally, designs are margined to operate in the worst case global process corner. In this chapter, we have chosen to focus on the write operation because it has been shown by [17] to set the minimum operating voltage in newer technologies. In addition, replica bitlines and self timed paths have been shown to reduce active energy by optimizing the read wordline pulse width separately from the write WL pulse width [83–88]. As explained in Section 1.1.2, the write-ability of the bitcell is determined by the current ratio of the NMOS pass-gate and PMOS pull-up devices. A high $\frac{I_{PG}}{I_{PU}}$ ratio leads to higher write-ability. This means that the SF (Slow-NMOS, Fast-PMOS) corner is the worst case for writing data into the bitcell.

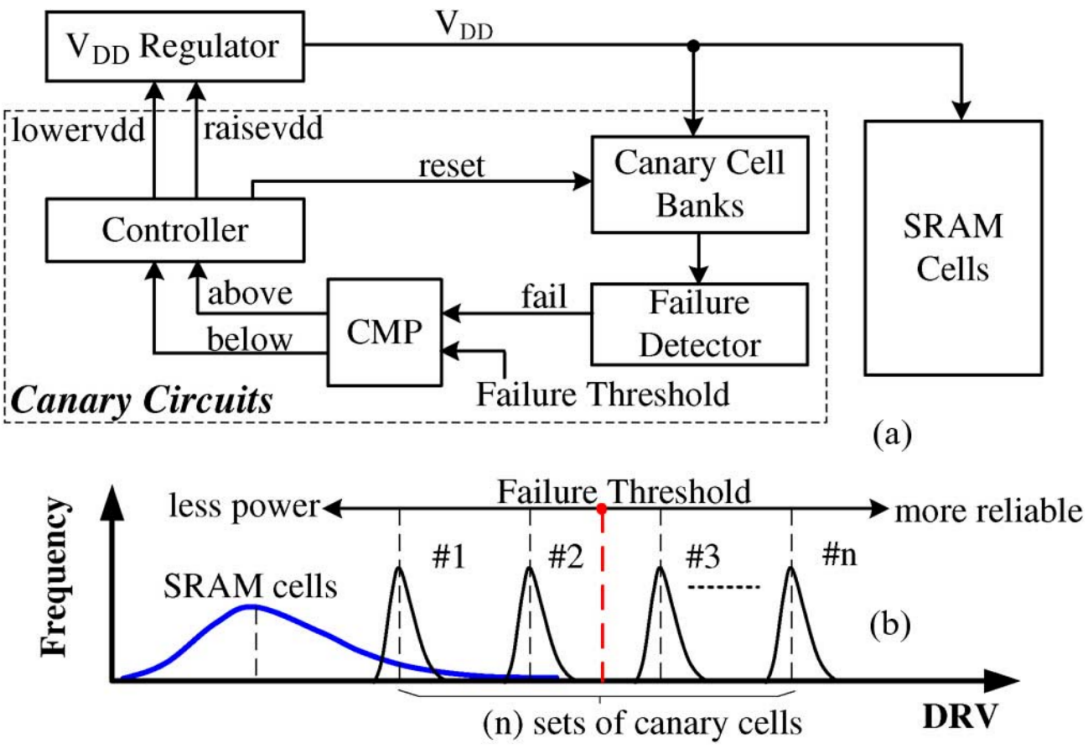


Figure 6.2: (a) The canary control scheme counts the number of failures, then adjusts the core voltage accordingly (b) using multiple sets of canaries allows for a tradeoff between power and reliability [9]

6.1 Prior Art

The idea of using canary structures to monitor the failure point of storage elements was first introduced by [89]. In this paper, the operating voltage of flip-flops is reducing during standby mode in order to reduce leakage power. The canary flip-flop is sized to fail at a voltage higher than the core flip-flops. During standby mode, the voltage is scaled down until a failure is detected in the canary array. This allows for closed loop voltage scaling, which enables the flip-flops to operate close to the optimal voltage. Because flip-flops are such a widely used storage element in synthesized logic, [90] was able to show a 20.5% energy reduction in a multicore processor using canary flip-flops.

In [3, 9, 16] the use of canary circuits has been extended to SRAMs. In this work, the author has implemented a closed control loop (Figure 6.2a) which detects the data retention

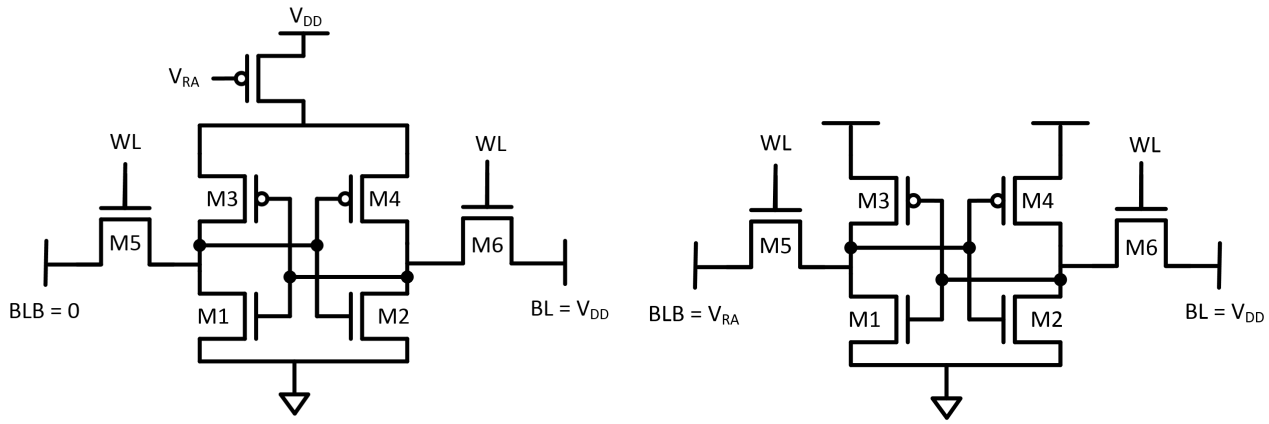


Figure 6.3: (a) Raising the gate voltage of the PMOS header creates a voltage drop between V_{DD} and the virtual rail of the canary cell (b) increasing V_{RA} weakens the pass-gate, thus increasing write V_{MIN}

voltage (DRV) of the canary array, then tunes the core voltage accordingly. DRV is the lowest possible voltage that the bitcell can operate while still retaining its data. In order to mimic the behavior of the core array as closely as possible, the canary cells use identical sizing and layout. The failure voltage is tuned using a PMOS header to set the canary virtual V_{DD} below the core V_{DD} (Figure 6.3a). By increasing the gate voltage of this header (thus reducing the voltage on the virtual rail), the failure voltage of the canary can be tuned across a wide range. This allows for multiple failure thresholds (Figure 6.2b) which enables a tradeoff between reliability and power. The author shows that the canary architecture can achieve up to a 5x reduction in leakage power compared to the conventional guard-band approach. However, the downside to this scheme is that the control loop is only active when the core array is in retention mode. When the core array is active (read or write mode) the voltage is scaled back to the guard-band voltage. Therefore, this scheme only addresses the problem of leakage power, not active power.

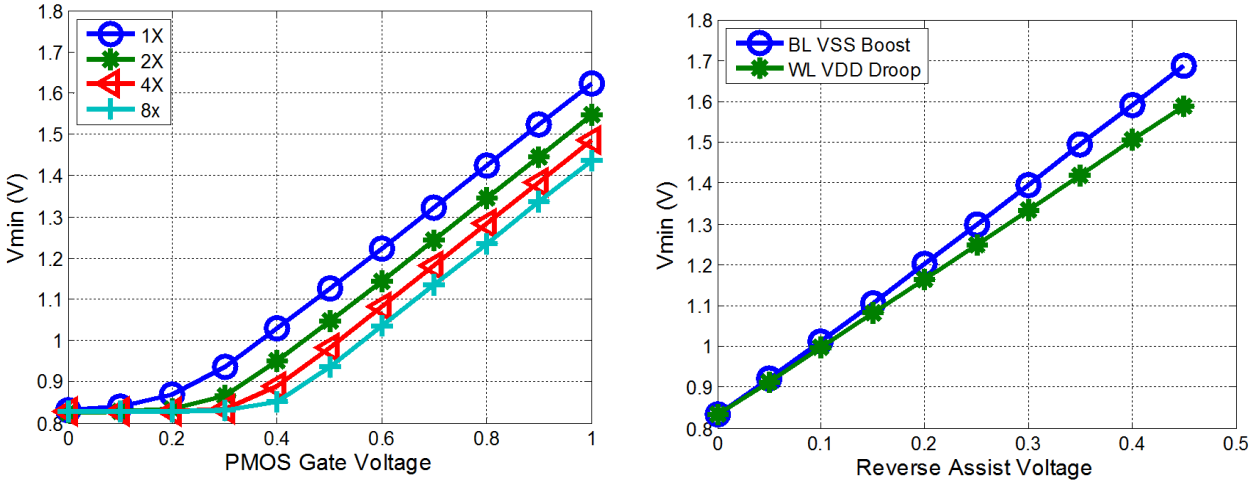
In [91], the author introduces a reverse assist canary circuit for tracking dynamic write V_{MIN}. As shown in Figure 6.3b, the canary cell uses a boosted bitline V_{SS} voltage to weaken the pass-gate device, thus increasing write V_{MIN}. In order to check for write failures, the canary cell is written and read during normal operation, which introduces an active power

overhead. The canary cell uses the same wordline pulse width and wordline driver as the core array in order to closely match the behavior of the core cells. The author shows that the canary architecture is able to reduce the energy per cycle by up to 51.5% at the best case corner. The downside to this work is that there is no analysis showing how many canary cells are necessary to minimize the active energy of the system. Because there is local mismatch present in the canary cells, the ability of the canary array to set the core array close to the optimal voltage is determined by the size of the canary array. However, we will show in this chapter that there is an optimal number of canary cells for a given core array size, that balances the overhead of adding additional canaries with the energy saved. In addition, we will show that using a PMOS header, (as first introduced in [16]) as opposed to the BL reverse assist method, allows the core V_{DD} to be tuned closer to its optimal V_{MIN}.

6.2 Comparison of Canary Types

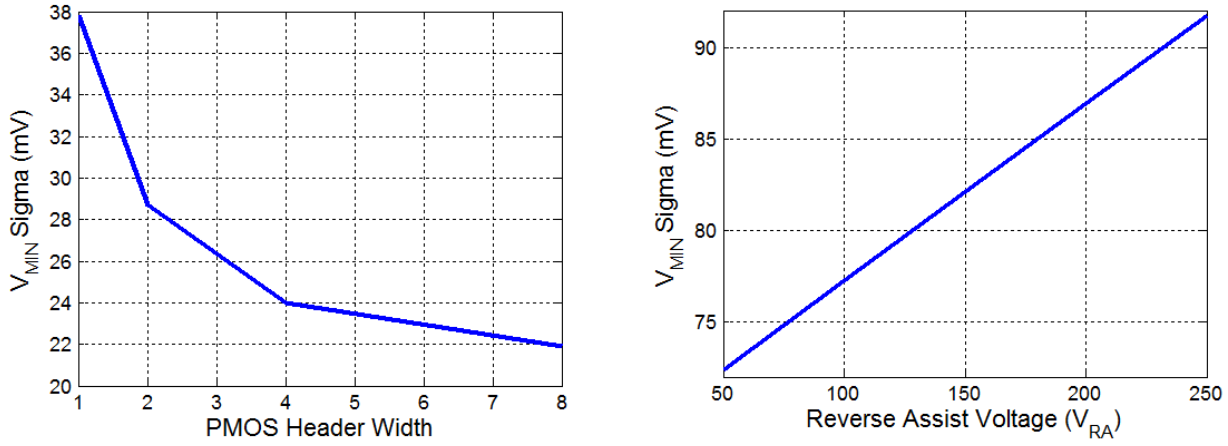
Figures 6.3a and 6.3b show the two options for canary cells. Each uses a different strategy to raise the write V_{MIN} relative to the core array. The circuit in Figure 6.3a creates a virtual V_{DD} lower than that of the core, which raises the average failure voltage of the canary cell. The canary in Figure 6.3b uses a boosted BL voltage to weaken the pass-gate transistor relative to the PMOS pull-up, thus reducing the write-ability of the cell. A third option is to under drive the WL voltage, which also reduces the strength of the NMOS pass-gate. We will use two metrics to compare the three cells options: tune-ability and robustness.

The first metric is important because the cell must be tune-able to fail at a wide range of voltages. By sweeping the reverse assist voltage (V_{RA}), the failure voltage of the canary cell changes. Figure 6.4a shows that as the width of the PMOS header is increased, the range of V_{MIN} decreases. However, even at 8x minimum size, the range of V_{MIN} is still 608 mV. We will show later in this chapter that this is more than a sufficient range to account for the local mismatch of the core array. The BL boost method achieves the highest range of 854



(a) (b)

Figure 6.4: Range of V_{MIN} for three canary types: (a) PMOS header, (b) WL droop and BL boost reverse assist



(a) (b)

Figure 6.5: Range of V_{MIN} for three canary types: (a) PMOS header, (b) WL droop and BL boost reverse assist

mV, while the WL droop method achieves a range of 756 mV. For these simulations, the WL pulse width is kept constant, and the failure voltage is located by performing a binary search.

Robustness is measured as the standard deviation of the canary array V_{MIN} distribution under the presence of local mismatch. In order to measure this, Monte Carlo simulations were run on the three canary cells, and the standard deviation was measured. Figure 6.5a shows

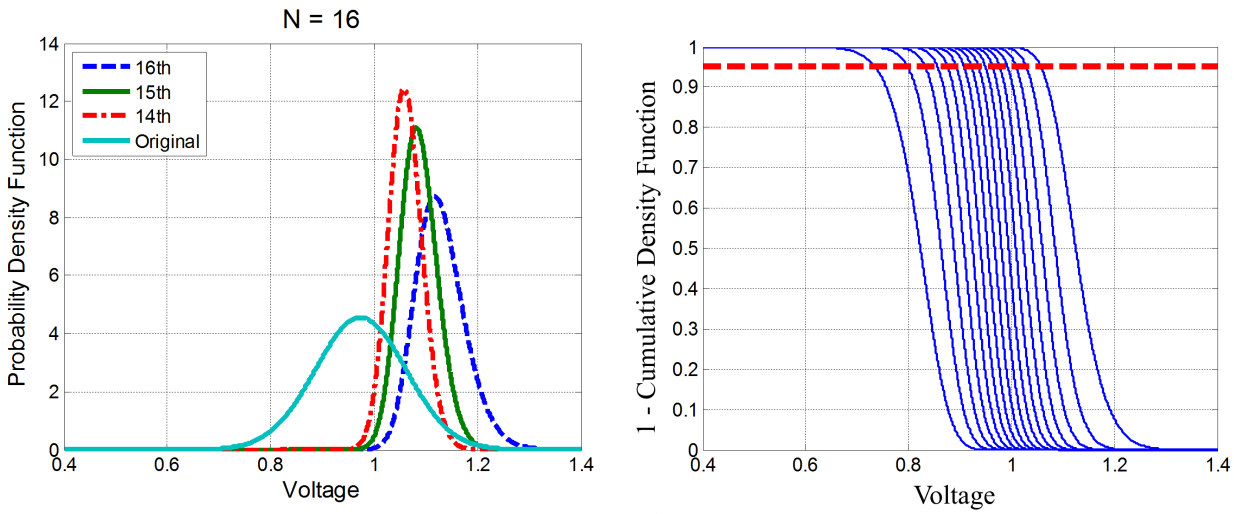
that as the PMOS header width is increased, the standard deviation decreases. This is due to the fact that V_T variation is inversely proportional to the channel area ($\sigma_{Vt} \sim \frac{1}{\sqrt{W * L}}$). At a width of 8x minimum, the PMOS header canary achieves a standard deviation as low as 21.9 mV. The standard deviation of the BL boost and WL drooop canary is close to identifcal, and is shown in Figure 6.5b. This figure shows that as the reverse assist voltage is increased, the σ_{Vmin} increases. The average sigma at $V_{RA} = 150mV$ is 82 mV, nearly 4x that of the PMOS header canary. This metric is important because decreasing sigma results in a tighter confidence interval, thus allowing the canary array to more closely approximate the optimal core V_{MIN}. Table 6.1 summarizes the results. Based on this table, the PMOS header canary type is the best options for tracking the core V_{MIN} because it has a sufficient range and is the most robust option.

Table 6.1: Comparison of the three canary cells

	PMOS Header (8x)	**WL V_{DD} Droop**	**BL V_{SS} Boost**
Tuneable Range	608 mV	756 mV	854 mV
Sigma	21.9 mV	82 mV (average)	82 mV (average)

6.3 Optimizing Canary Design using Order Statistics

Using the plot in Figure 6.1, we can measure the target V_{MIN} for the design. In this plot, the y-axis represents the write failure probability of a single bitcell at the given supply voltage. Using the binomial distribution (as in Chapter 5), we calculate the maximum failure probability needed to ensure a given die yield. From the plot in Figure 6.1 we then measure the target V_{MIN} at the typical corner, and design the canary array to accurately identify this point. Because the canary cell is identical to the core cell, it will be able to track PVT



(a) (b)

Figure 6.6: Example of an order statistic for N=16 (a) probability density function (b) cumulative distribution function

variation, thus allowing the core array voltage to be set to the optimal V_{MIN}. This target voltage is the point that we will design the canary system to track using order statistics.

Order statistics are used to characterize the kth smallest value of a set of samples. The equation for the probability density function (PDF) $z_{N:k}$ (where k represents the kth smallest element from a sample size of N) of a distribution with a PDF of f(x) and a cumulative distribution function (CDF) of F(x) is [92]:

$$z_{N:k}(x) = \frac{N!}{(k-1)!(N-k)!} [F(x)]^{k-1} [1 - F(x)]^{N-k} f(x) \tag{6.1}$$

In this case, f(x) and F(x) are the PDF and CDF of the canary V_{MIN} distribution, which follows a normal distribution. In the case of the canary design, $k = 1$ represents the canary with the lowest V_{MIN}, while $k = N$ represents the canary with the highest V_{MIN}. As the memory (core and canary) voltage is scaled down, we expect the canary $k = N$ to fail first, then $k = N - 1$, and so on. Figure 6.6a shows an example of a canary array of size 16. This figure shows the failure distribution of the three canaries with the highest V_{MIN}. Figure 6.6b shows the CDF of the canary distribution. The intersection of $Y = 0.95$ and each

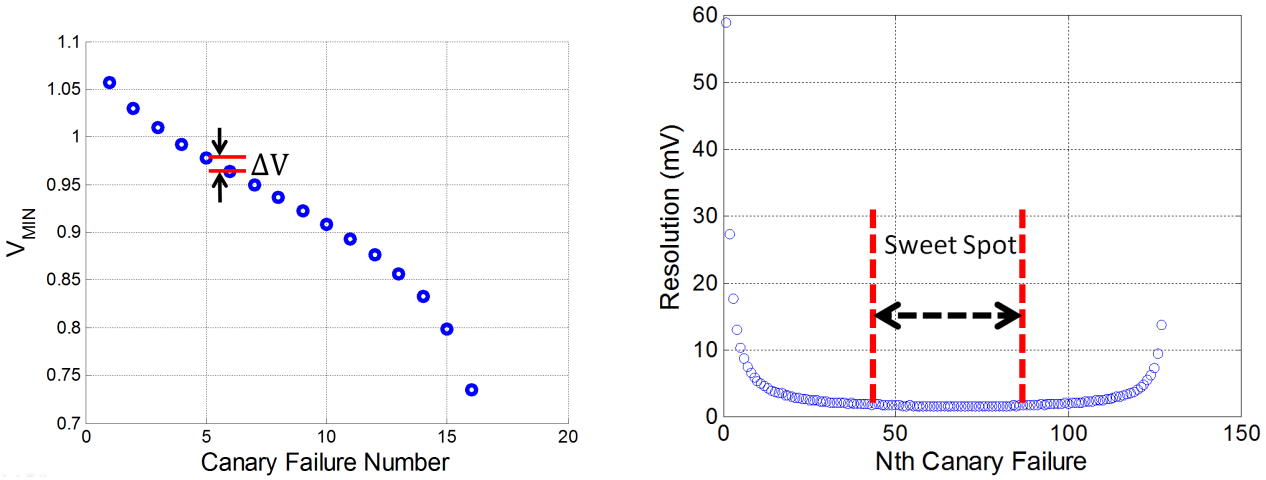
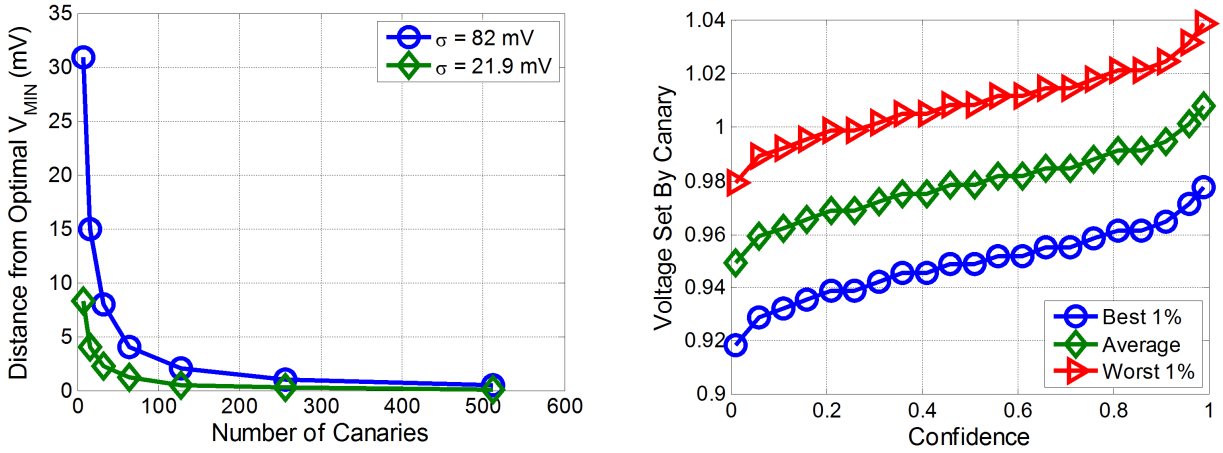


Figure 6.7: (a) shows the expected failure points at a confidence of 0.95 for a canary array of N=16. ΔV represents the resolution of the canary array (b) ΔV of a canary array N=128. The "sweet spot" occures between $k = \frac{N}{4}$ and $k = \frac{3N}{4}$

order statistic represents the voltage that each successive canary is expected to fail at, or above, with 95% confidence. We can see from Figure 6.6 that as k approaches N/2 the order statistic becomes more robust, meaning that the width of the confidence interval decreases. By designing the canary array using order statistics, we are able to more closely track the target core V_{MIN}.

Figure 6.7a plots the expected failure voltages at a confidence of 0.95 of the canary array from Figure 6.6b. The difference in voltage between successive failures (labeled ΔV) represents the resolution of the canary array. A smaller resolution means that the canary array is able to detect smaller changes in the supply voltage. Reducing the resolution allows the canary array to more closely track the optimal core V_{MIN}. This resolution depends on two factors: the number of canaries and the standard deviation of the canary V_{MIN} distribution. Figure 6.7b plots the resolution versus the canary failure number for a canary of size $N = 128$. We can see from this plot that the tail points have the worst resolution. This is evident in the Figure 6.6a where the worst canary, $k = 16$, has the largest standard deviation. The points with the smallest resolution occur in the "sweet spot" roughly between $k = \frac{N}{4}$ and $k = \frac{3N}{4}$.



(a) (b)

Figure 6.8: (a) plots the canary resolution versus the number of canaries for the PMOS header canary ($\sigma = 21.9mV$) and the BL reverse assist canary ($\sigma = 82mV$) (b) shows the tradeoff between confidence and target voltage

Therefore, we want to design our canary array such that the core target V_{MIN} aligns with this sweet spot, in order to minimize the core V_{DD}.

The plot in Figure 6.8a shows the tradeoff between resolution and canary array size. The resolution of the canary array represents the maximum voltage that the core array can be tuned from its optimal V_{MIN}. As the number of canaries increases from eight to sixty-four, this resolution decreases sharply. However, the net gains when increasing the canary array size greater than sixty-four quickly diminish. In addition, this plot shows that the PMOS header canary achieves on average 73.42% smaller resolution than the BL reverse assist canary at identical canary array sizes. Figure 6.8b shows the tradeoff between confidence and canary voltage. In this example, the target V_{DD} that the canary is tuned to set is 0.977V. Table 6.2 shows the minimum, average, and maximum expected core array V_{DD}. On average, the PMOS header canary achieves a 2.31% lower V_{MIN}. Because there is variation within the canary cells, each die will not be set to the same voltage. The plot in Figure 6.8b shows the average case as well as the best and worst 1% of dies. At a confidence of 0.99, the best 1% of dies are set to a voltage of 0.9777, which is above the target voltage of 0.977. As confidence is reduced, the number of canary failures that the system can tolerate increases, thus allowing

the core V_{DD} to scale down more aggressively. If designing to meet a confidence of 0.5, then the average energy will decrease, however the expected die yield is 50%. Using this analysis, the designer can easily set a target die yield of the canary system. For the rest of this chapter, we will target a die yield of 99%.

Table 6.2: Comparison of the target V_{DD} for two canary types

	PMOS Header (8x)	**BL V_{SS} Boost**
Minimum	0.9777V	0.9778V
Average	0.9853V	1.0081V
Maximum	0.9936V	1.039V

Figure 6.9 shows the target voltage set by the canary system versus the number of canary cells. In this example, the target core voltage was 0.977, the confidence is 0.99, and the canary type is the PMOS header. We can see that as the number of canaries increases, the average and worst 1% of die begin to approach the optimal V_{DD}. However, even with 512 canaries, there is still a small amount of variation between each die. Once again we see that as the number of canaries exceeds sixty-four, the overall voltage reduction of adding additional canaries degrades. In the next section, we will show that there exists an optimal number of canary cells for a given core capacity.

6.4 Optimizing Energy Savings

In order to minimize energy, the overhead of adding additional canaries should not outweigh the energy savings from reducing the core V_{DD}. Unlike in [3, 9, 16], where the canary array only contributes to leakage energy, each additional canary contributes to both the leakage and active energy. Our assumption is that the canary array will be read and written during each cycle. As additional canaries are added, the WL and BL capacitance of the canary array will increase, resulting in a higher energy overhead. For this design, we found that reducing

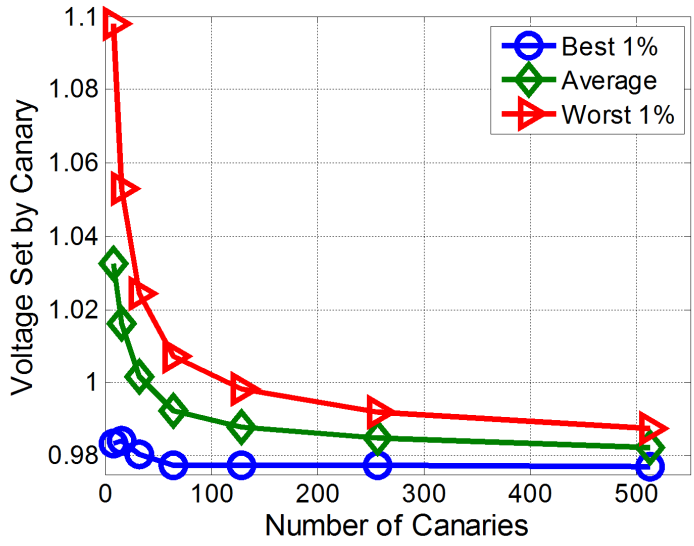


Figure 6.9: The expected core voltage versus the number of canary cells

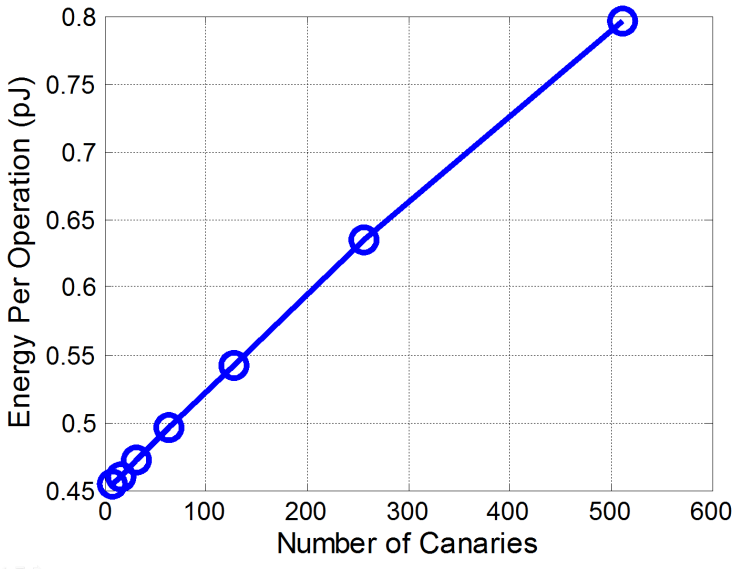


Figure 6.10: The energy overhead of the canary array increases linearly as the capacity increases

the WL capacitance by only using eight canary cells per row minimized the overall energy of the canary array. Adding additional canaries in this case means increasing the number of rows, and therefore the BL capacitance. Figure 6.10 shows the canary energy as a function of the canary array capacity. We see from this plot that the energy overhead increases linearly with the number of canaries.

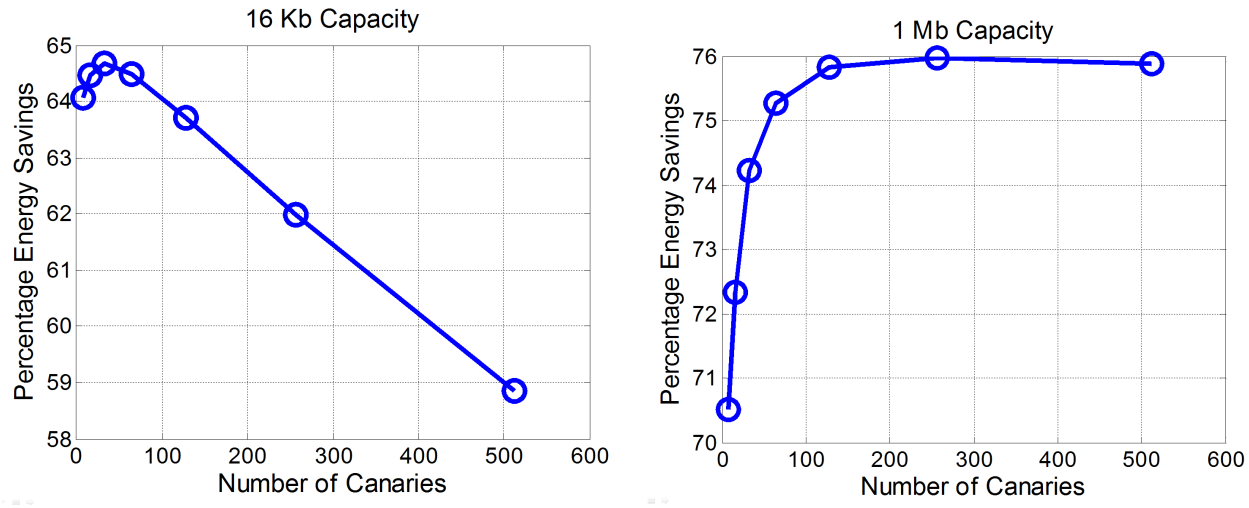


Figure 6.11: Characterizing the minimum energy point for a (a) 16 Kb memory and a (b) 1 Mb memory

In order to measure the system level energy savings, we simulated three core array sizes (16 Kb, 128 Kb, and 1 Mb) using the ViPro tool described in Chapter 5. The target core V_{DD} at the TT-27c corner was calculated using importance sampling [56]. Once the target V_{DD} is known, the average expected voltage of the core array was measured as shown in Figure 6.9. Using these voltages, the macro level energy was measured by ViPro (Chapter 5. By combining this data with the data from Figure 6.10, we can calculate the maximum percentage energy savings achieved using the canary system relative to the worst case corner (SF, 100 °C, +/-10% supply variation). Figure 6.11a shows that the maximum percentage energy savings for a 16 Kb memory occurs using a canary array size of 32. At canary arrays larger than 32, the overhead of adding additional canaries outweighs the energy saved by V_{DD} reduction. Both Figures 6.11a and 6.11b show the results using a PMOS header canary. Figure 6.11b shows that the maximum percentage energy savings for a 1 Mb core array occurs at a canary array size of 256. Figure 6.11 shows that as the capacity of the core array increases, the canary energy overhead relative to the system level energy is reduced.

Figure 6.12 plots the percentage energy savings for a 128 Kb using two types of canary cells: the PMOS header and reverse BL assist canary. On average, the PMOS header provides

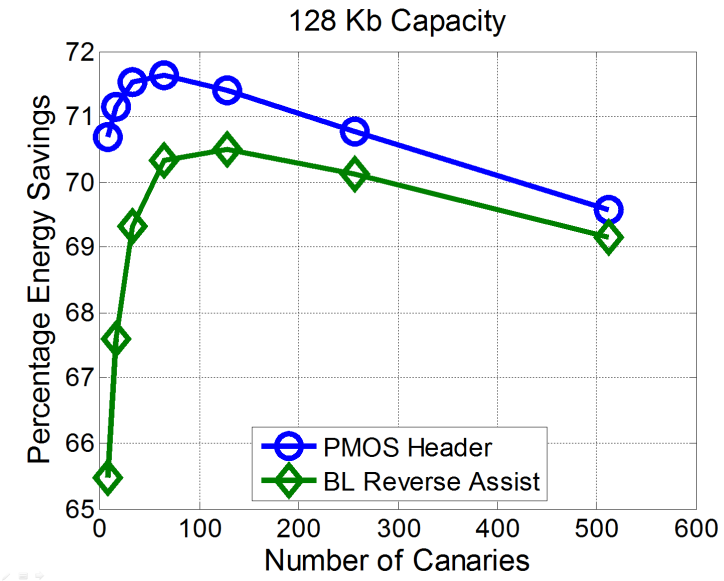


Figure 6.12: The PMOS header provides an overall energy savings of 4.0% over the reverse BL assist canary

a 4.0% lower system level energy relative to the reverse BL assist canary. While this isn't a huge net gain, the more interesting trend is that the optimal energy point occurs at a canary array size of 64 using the PMOS header, and 128 using the reverse BL assist technique. This is due to the fact that the PMOS header has a more robust (e.g. smaller standard deviation) V_{MIN} distribution relative to the reverse BL assist canary. This allows the PMOS header to reduce the core array voltage 22.8 mV lower than the reverse BL assist technique at a canary array size of 64. Improving robustness of the canary design allows the core array to be tuned closer to its optimal V_{MIN}.

In order to quantify the energy savings from using the canary system, we compared the minimum energy points from Figure 6.11 and 6.12 to the energy measured by ViPro by using conservative guard-banding at two corners. The first corner is the worst case corner for write: SF process corner, 100 °C, and +/-10% supply variation. Because it is possible to track the process corner using characterization sensors other than canary bitcells, we also compared the energy savings to the TT process corner, 100 °C, and +/-10% supply variation. The results are shown in Table 6.3. We can see from this table that the benefit of using canary

cells increaes as the memory size increases. This is due to the fact that the energy overhead of the canary array becomes smaller relative to the core energy as the core size increases.

Table 6.3: Energy savings using the canary system

	Energy Savings Compared to TT, 100 °C, +/-10% V_{DD}	**Energy Savings Compared to SF, 100 °C, +/-10% V_{DD}**
16 Kb	2.91%	64.67%
128 Kb	21.05%	71.63%
1 Mb	28.29%	75.97%

6.5 Conclusions

In this chapter, we presented a new approach for optimizing canary SRAM systems using order statistics. We show that using a PMOS header with a modulated gate voltage to set the canary V_{MIN} is more robust compared to boosting the BL V_{SS} or drooping the WL V_{DD}. Using order statistics, we show that there is a trade-off between the number of canaries and the voltage resolution of the canary array. In addition, we show that aligning the target core voltage with the canary "sweet spot" (between $k = \frac{N}{4}$ and $k = \frac{3N}{4}$) minimizes the resolution of the array, allowing the core voltage to be tuned closer to its optimal V_{MIN}. Next, we identified a minimal energy point (or maximum energy savings) when using canary systems to track write V_{MIN}. This point varies due to the fact that the energy overhead of the canary array becomes smaller relative to the core energy as the core size increases. Finally we show that using the canary system can save up to 75.97% energy in a 1 Mb compared to guard-banding for the worst case PVT corner.

Chapter 7

Sense Amplifier Designs for Reducing Offset

[1] As technology has scaled into the nanometer regime, designing high performance SRAMs has become more difficult due to increases in process variations, leakage, and memory capacity. One of the major failure mechanisms is read access stability [52, 54, 93]. To ensure that the correct data is sensed during a read operation, designers must account for variations in both I_{READ} and sense amp offset (σ_{OFFSET}) [2]. SRAM devices are typically minimum sized to maximize density, which makes them even more susceptible to process variation [51]. Increases in capacity have also lead to higher bitline (BL) densities, which causes the BL dissipation during a read to dominate the total read access time. Sense amplifiers are used to reduce the BL differential required for a successful read. This reduces both read energy and delay. However, these devices are also susceptible to threshold voltage variation because they are designed to be electrically symmetrical. Consequently, any variation in the cross coupled inverters can cause functional failures as well as reduced performance [94]. Therefore, to ensure reliable read operation at high performance, σ_{OFFSET} must be minimized (Figure 7.1).

[1] This chapter is based on the published paper titled: Stack Based Sense Amplifier Designs for Reducing Input-Referred Offset" [JB7]

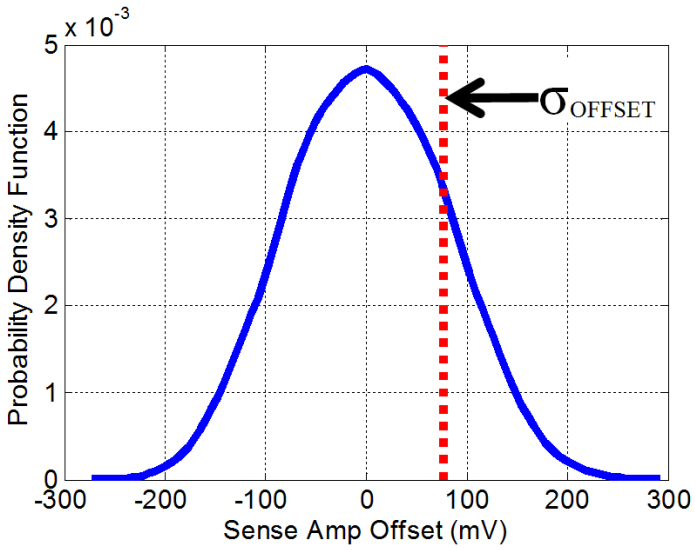


Figure 7.1: Reducing σ_{OFFSET} reduces read energy and delay

The most common method for reducing σ_{OFFSET} is to upsize the devices in the SA circuit. Because intra-die variation due to random dopant fluctuations is proportional to the size of the channel, upsizing reduces this source of local mismatch [95]. Other methods for counteracting sense amp offset include: redundancy, increasing the bitline differential before firing the sense amp, and error correction coding. These methods improve robustness, but can also degrade performance, bit density, power, and reliability [96]. Offset compensation techniques [96, 97] have also been used to perform post-silicon tuning, however this approach requires an area and control overhead. In this chapter, we present an offset reduction scheme that does not require major modifications to the existing conventional latch based sense amp, as well as three new sense amp topologies. Each of these topologies offers reduced σ_{OFFSET} relative to the conventional latch at the cost of more devices.

7.1 Methods for Reducing Sense Amp Offset

In this section we present three methods for reducing σ_{OFFSET}. Our approach to increase robustness is to improve the sensitivity of the sense amp to the voltage differential at the inputs. Figure 7.2 shows the schematic of a conventional voltage-mode sense amplifier. When

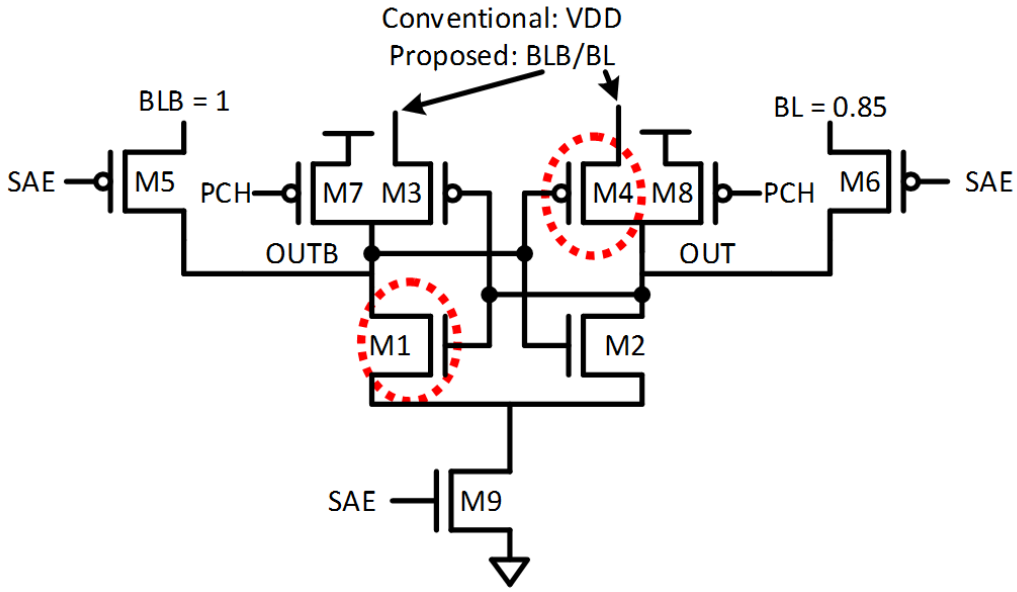


Figure 7.2: Schematic of the conventional latch based sense amp and the proposed modification

a voltage difference is supplied across the input of the SA, as shown in Figure 7.2, M1 is weakened relative to M2, due to a lower V_{GS}, resulting in OUT pulling down to 0. To improve the robustness of the design, we want to maximize I_{M2}/I_{M1} when BLB > BL and vice versa.

7.1.1 Source Coupled Scheme

To ensure that OUT is pulled to 0, M4 and M1 must be weakened relative to M3 and M2. The compensation scheme that we are proposing involves connecting the source of M4 directly to the BL and the source of M3 to BLB (Figure 7.2). This effectively weakens M4 relative to M3 by reducing its V_{GS} and V_{DS}. Because these PMOS devices are initially off ($V_{GS} = 0$) before the sense amp is fired, they do not draw current from the bitlines. During a read, the bitlines droop slightly below V_{DD}. However, once the sense amp is fired, if $BLB = V_{DD}$, then OUT is discharged to ground and OUTB is able to remain at V_{DD}. Therefore, in either case the output differential remains V_{DD}. To quantify the effectiveness of this scheme, we ran transient Monte Carlo simulations and compared σ_{OFFSET} of the compensated vs. uncompensated sense amps. The results in Figure 7.3 show that the compensation scheme

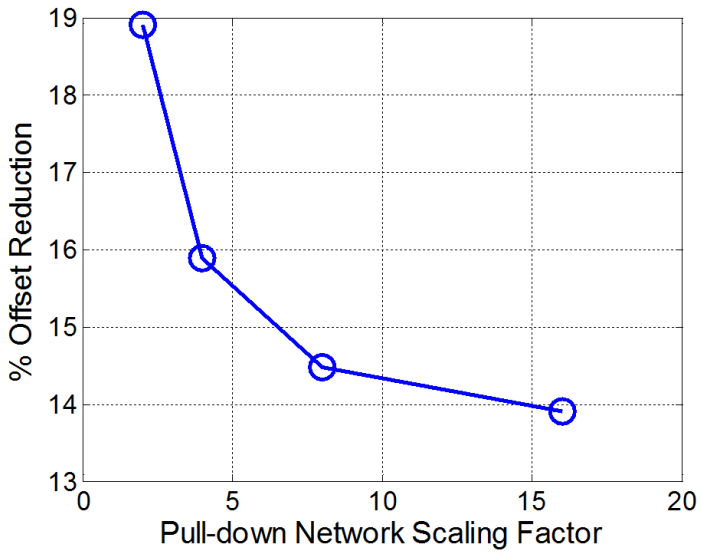


Figure 7.3: The offset compensation scheme provides up to a 19% reduction in sense amp offset. Scaling factor represents the scaling of the pull-down network

offers up to a 19% reduction in σ_{OFFSET} for a 2*minimum size pull down network (M1 and M2) sense amplifier. The advantage of this technique is that it provides a reduction in offset without adding additional devices, thus resulting in zero area overhead.

7.1.2 Schmitt Trigger Sense Amp

As described in the previous section, to reduce sense amp offset, we need to increase the strength of one of the cross coupled inverters relative to the other. Schmitt triggers are often used to improve the robustness of a standard inverter by modifying the switching threshold. [5] showed that using a Schmitt trigger in the pull down network of the standard 6T SRAM bitcell increases the read stability of the cell by increasing the switching threshold of the inverter during a 0 to 1 input transition. We can apply this same concept to the sense amplifier. After the sense amp is fired, M9 (Figure 7.2) begins to pull down the source of M1 and M2, causing the V_{GS} of these devices to rise. If the gate of M1 is at a lower voltage than the gate of M2 (e.g. $I_{M2} > I_{M1}$), then OUT is pulled low. Adding a Schmitt Trigger (ST) to the pull down network (Figure 7.4) increases the switching threshold during a 0 to 1 input

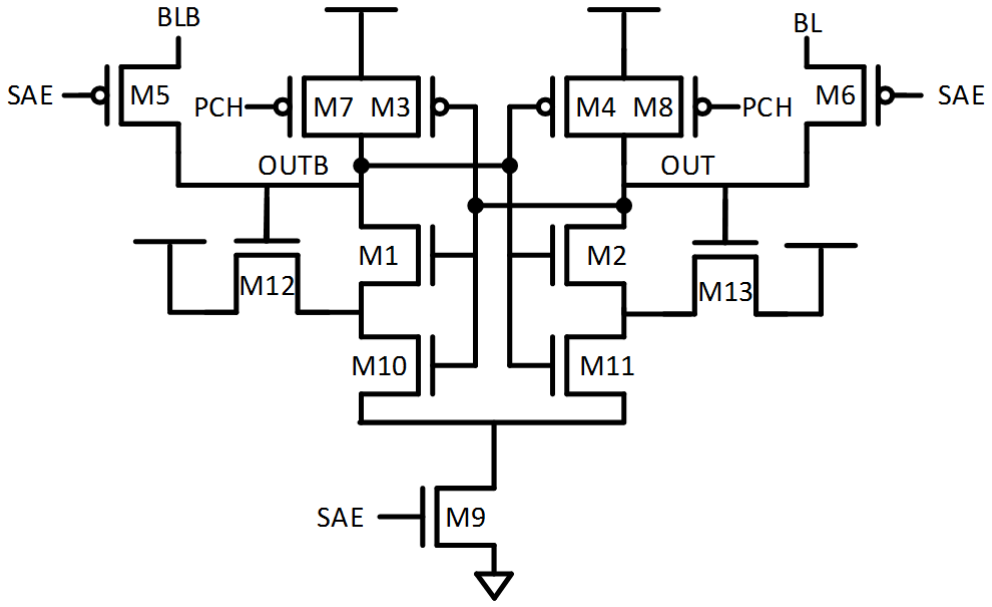


Figure 7.4: Adding a Schmitt trigger to the pull down network enhances the SAs sensitivity to small changes at the inputs

transition which we will show greatly weakens the inverter holding 1 relative to the inverter pulling down to 0.

Figure 7.5 shows the simulation waveforms of the conventional SA and the Schmitt trigger (STn) SA. The initial conditions are: $BL = 0.85V, BLB = 1V$. What we see from this plot is that before the SA is fired, $V_{GS-M2} - V_{GS-M1}$ of the conventional sense amp is 150 mV (the input differential), while the STn SA has a input voltage difference of 240.3 mV due to the feedback of the ST devices. Once the sense amp is enabled, the V_{GS} of both M1 and M2 begins to rise. However, in the case of the conventional sense amp, $V_{GS\text{-}M1}$ reaches a maximum value of 667 mV, while $V_{GS\text{-}M1}$ of the STn sense amp reaches only reaches 266 mV. The resulting current ratio $\frac{max(I_{M2})}{max(I_{M1})}$ is 2.98 for the conventional SA and 61.2 for the STn SA. Because $V_{GS\text{-}M1}$ of the STn SA never crosses the threshold voltage, it is unable to pull OUTB low, thus resulting in a higher tolerance to V_T variation, which we will show in section 7.2. The drawback of this topology is an increase in the layout area. Compared to the conventional latch-based sense amplifier, this design consumes 20% more area.

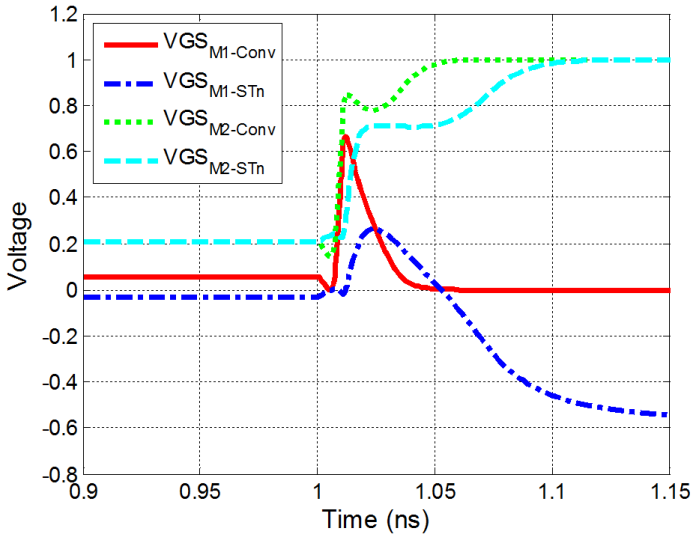


Figure 7.5: $V_{GS\text{-}M1}$ of the STn SA never rises above the threshold of the NMOS device, resulting in a larger current ratio between M2/M1

7.1.3 Stacked Sense Amp Topologies

The STn sense amp achieves a higher level of sensitivity and tolerance to V_T variation by increasing the switching threshold of the inverter. This can also be achieved by weakening the pull-down network using stacked NMOS devices as shown in Figure 7.6. This design also achieves high robustness to V_T variation, using the concept of Elmore delay. When sense amp is fired, M12 and M13 begin to discharge nodes SL2 and SR2 respectively. Assuming again that BL = 0.85V and BLB = 1V, M13 is able to discharge SR2 to ground more quickly than M12 can discharge SL2 because of its higher V_{GS}. This pattern continues up the stack until M2 is able to discharge OUT to ground. Because the devices in the right branch of the stack have a higher initial V_{GS}, they in turn have a lower on resistance. This leads to a lower overall RC delay (parasitic capacitance is constant for both branches because the devices are equal size), allowing the right branch to discharge more quickly. Simulation results show that using a three device stack as shown in Figure 7.6, $V_{GS\text{-}M1}$ has a maximum value of only 202 mV. Using a two device stack, (e.g. removing M12 and M13) achieves a maximum $V_{GS\text{-}M1}$ of 416 mV. The advantage that this design has over the STn SA is that adding stacks to the pull

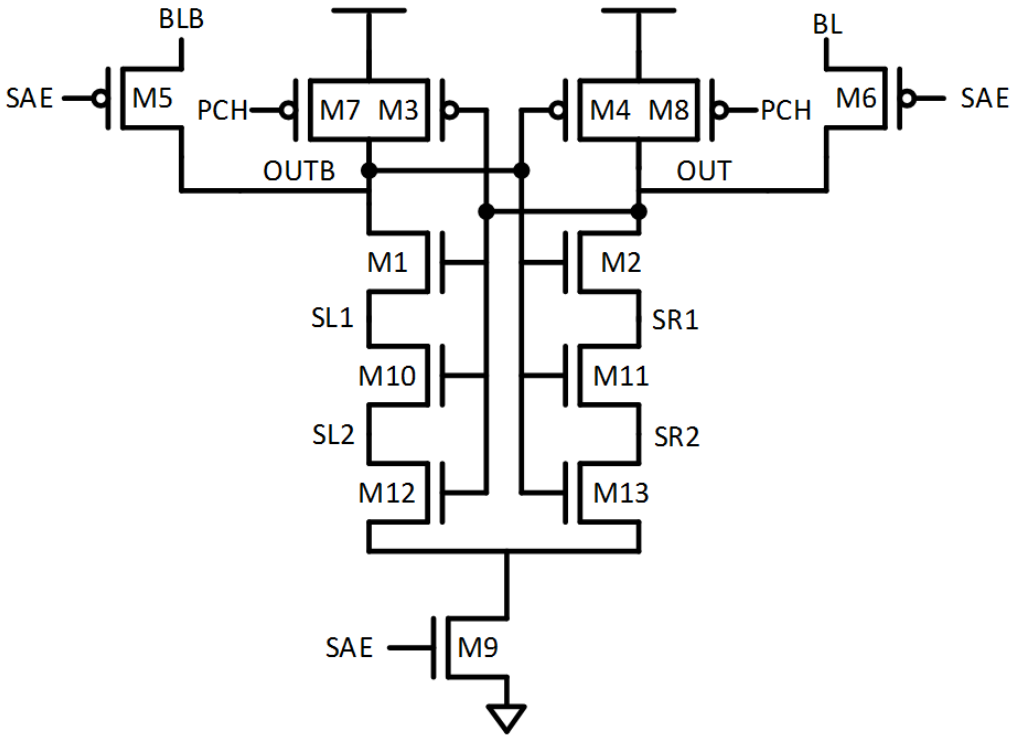


Figure 7.6: The use of stacked devices increases the switching threshold of the inverters

down network does not incur large overheads. If fingering is not used, then the two device stack requires no area overhead and the three device stack incurs only a 5% area overhead. The downside to this design is that the added devices increase the capacitive loading of OUT and OUTB, increasing the sense amp resolution time.

7.2 Evaluation of Sense Amp Topologies

In the previous section, we showed that the proposed sense amp designs provided a significantly higher sensitivity to the voltage differential across the inputs. In order to to evaluate the robustness of each sense amp to V_T variation, we used sensitivity analysis as a first order approximation. The plots in Figures 7.7a and 7.7b illustrate the effect of V_T variation on offset. These plots were generated by sweeping the threshold voltages of each of the devices in the cross coupled inverters. These plots show that variation in the PMOS pull-up devices

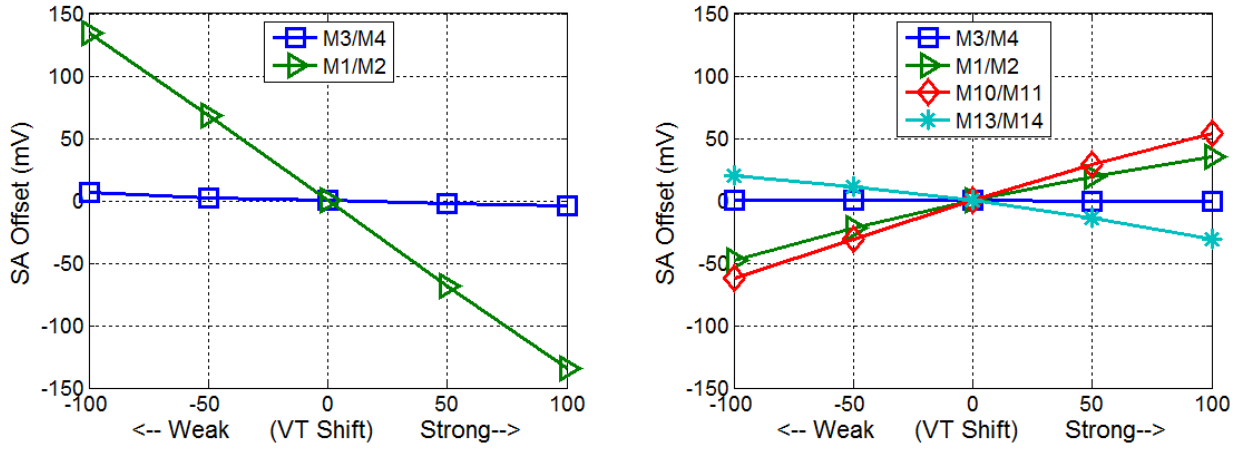


Figure 7.7: The additional devices in the STn sense amp reduce the sensitivity of the SA to fluctuations in V_T

(M3/M4 in Figure 7.2) has very little effect on SA offset. Therefore, upsizing these devices does not result in significant reductions in σ_{OFFSET}. In addition, the sensitivity to V_T variation of the pull-down devices (M1/M2 in Figure 7.2) is much higher in the conventional sense amp compared to the STn sense amp. In the conventional sense amp, a +/-100 mV shift in the threshold voltage of the pull-down devices results in an absolute offset of 134 mV. Meanwhile, the STn sense amp has a worst case offset of only 62 mV, due to the variation in M10 and M11 (Figure 7.4). It is interesting to note that devices M12 and M13 (Figure 7.4) have very little effect on offset, with a worst case magnitude of only 30 mV. Based on this data, we predict that the conventional sense amp will have a much higher σ_{OFFSET} relative to the proposed STn and stacked designs.

To evaluate the robustness of the four topologies presented in the previous section, we ran transient Monte Carlo simulations to measure σ_{OFFSET} across a range of device widths. We choose to only upsize the pull down devices, because sensitivity analysis showed that variation in the PMOS devices had very little impact on σ_{OFFSET}. Figure 7.8 shows that for device widths less than 8*minimum size, the three device stack has the lowest offset. Once the devices exceed this threshold however, the loading on the output nodes causes the sense

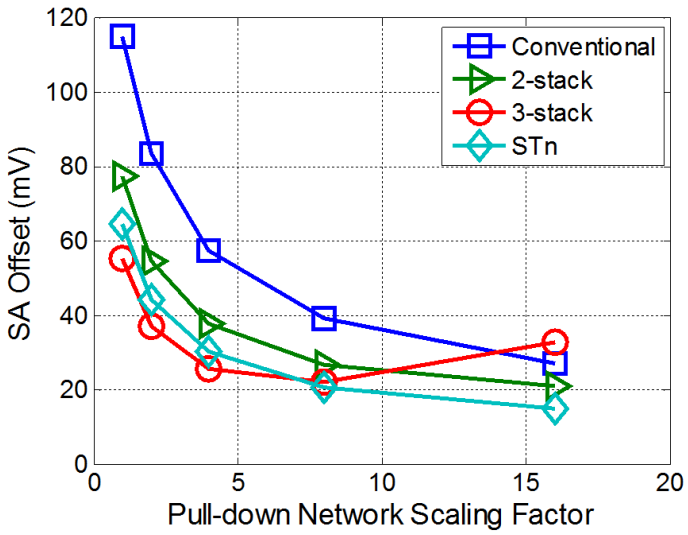


Figure 7.8: Sense amp offset (σ_{OFFSET}) vs. pull-down scaling factor

amp to function poorly. This was also the case in four device stacks and greater. In these cases, the time to discharge OUT/OUTB is much greater than the Elmore delay through the stacked devices. During resolution, both inverters near their switching threshold, which greatly increases the sensitivity of the SA to variations in the PMOS device. We can also see from this figure that at pull-down device widths greater than 8*minimum size, the STn SA has the lowest σ_{OFFSET}. At a scaling factor of 16 * minimum size, the STn sense amp achieves an offset of only 14 mV. Across the range of scaling factors, the STn sense amplifier consistently achieves an offset reduction of 50% relative to the conventional SA.

Figure 7.9 shows the resolution times for the various SA designs. The resolution time was measured assuming a 100 mV input differential and a single inverter buffering the output nodes. We can see from this figure that the conventional sense amplifier has the shortest resolution times. However, the sense amp resolution time represents only a small fraction of the total read access time. Therefore, we would expect that the reduction in BL development time, due to reduced σ_{OFFSET}, will result in an overall savings of both energy and delay. These savings are shown in section 7.3. We can see from Figures 7.8 and 7.10 that the effect of upsizing to reduce V_T variation begins to saturate once the devices reach 8*minimum size.

Because bit density is important, SRAM bitcells are typically scaled to minimum size

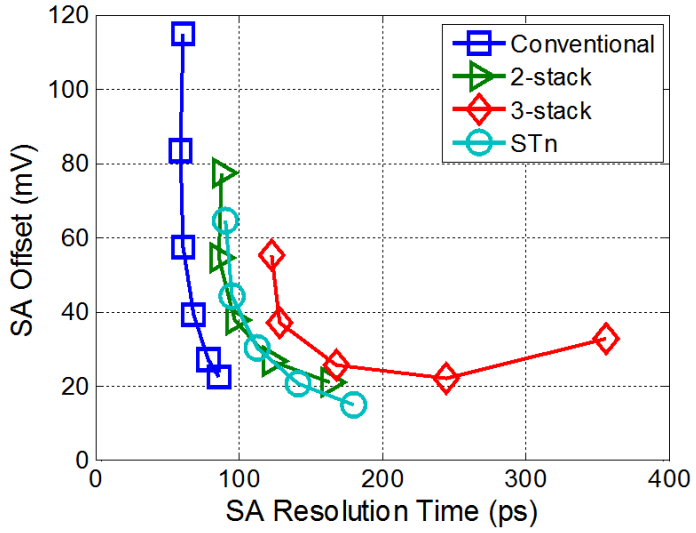


Figure 7.9: Adding capacitance to the output nodes increases resolution time

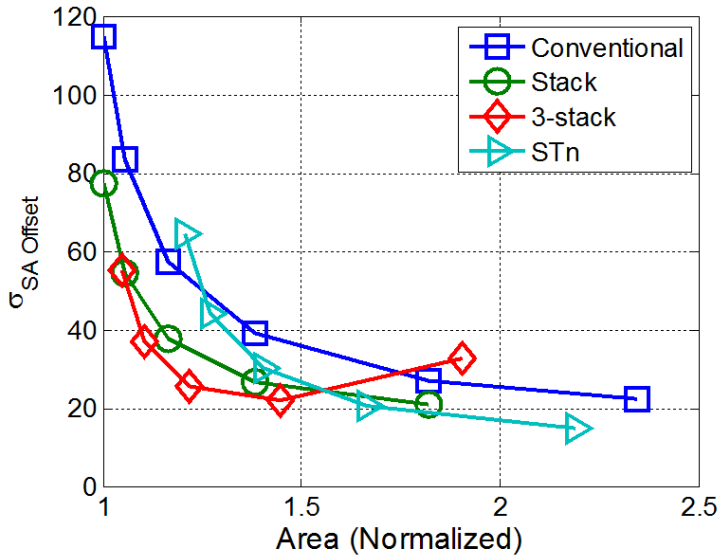


Figure 7.10: For smaller layout area, the stacked SAs are optimal, while the STn SA offers the lowest σ_{OFFSET} at the cost of higher area

using "pushed" rules to increase the density relative to standard logic. In addition, because SRAM arrays are very regular structures, it is important that the periphery logic (e.g. BL and WL drivers, column decoders, sense amps, etc.) is pitch matched to the bitcell array. This means that the height of the WL drivers (row periphery) much match the height of the bitcell, and the width of the column periphery must match the width of the bitcell.

However, because the bitcell area is so small, sense amps are typically shared by multiple columns, allowing the sense amp to be upsized relative to the bitcells. Increasing the number of columns sharing a single sense amp (e.g. increasing the number of words per row) allows the sense amp to be upsized further. Therefore, designs with a small number of words per row (e.g. $\leq$2) have more stringent area requirements. Figure 7.10 shows that for iso-layout area, the two device stack consistently produces a lower σ_{OFFSET}. For smaller layout areas, the three device stack offers the largest improvement in offset voltage of up to 48%, while the STn SA offers the lowest offset of the proposed sense amps at the highest layout area. Therefore in three device stack is optimal for designs with fewer words per row, while the STn SA is ideal for designs with a large number of words per row.

7.3 SRAM Macro Level Savings

To quantify the macro level energy and delay savings gained by reducing sense amp offset, we used the tool described in chapter 5. This tool accounts for both the sense amp delay and offset, which allows us to measure the full benefit of the new sense amp designs. Because BL discharge tends to dominant the total read time, we expect that a reduction in the sense amp offset will result in significant energy and delay savings. Figure 7.11 shows the Pareto optimal curves generated by ViPro for a 1Mb memory. In this example, we compare the three designs at a fixed NMOS device width (4x minimal size). We can see from this figure that the delay reduction achieved by reducing the sense amp offset outweighs the increased sense amp resolution time (shown in Figure 7.9). On average, the ST sense amp reduces energy and delay by 14.91% and 14.20%. The stacked sense amp achieved an average energy and delay savings of 10.75% and 10.07%. The area of the Schmitt trigger sense amp is 20% larger than the conventional sense amp. However, the benefits of the STn design are an increased sensitivity to the differential input which results in a more robust design in the presence of threshold voltage variation.

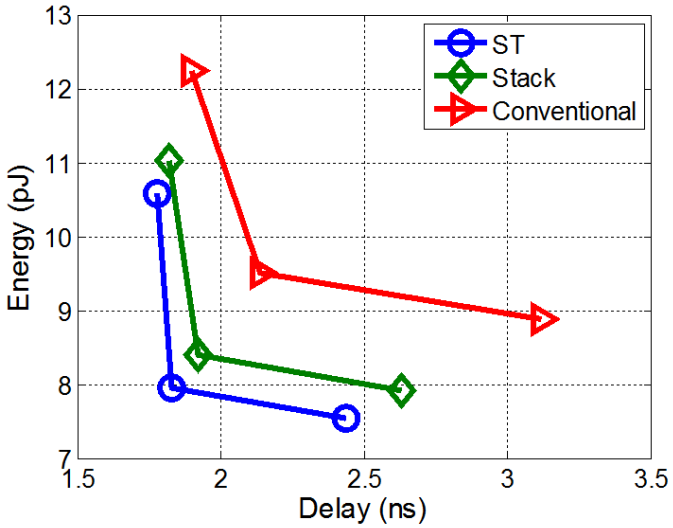
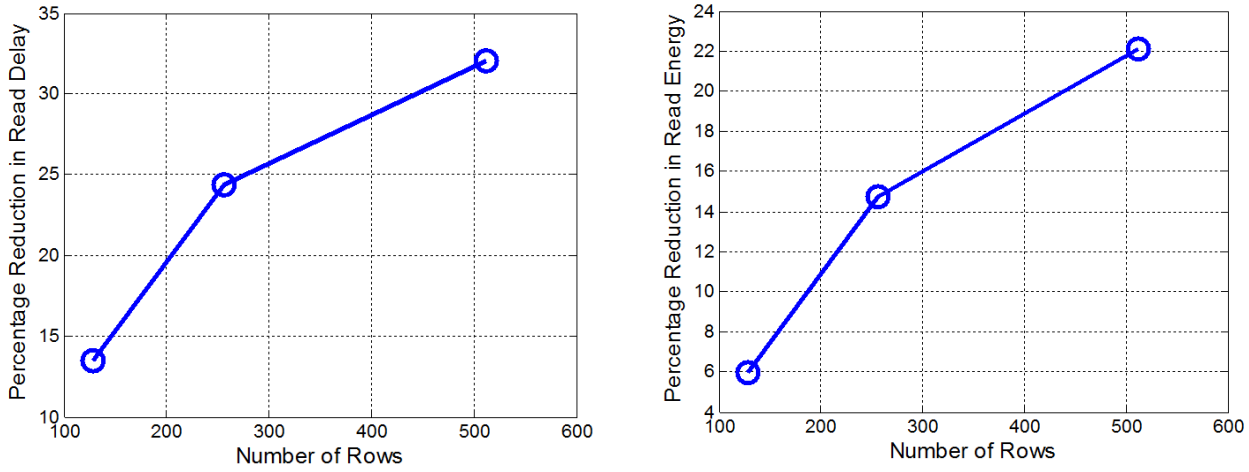


Figure 7.11: 1 Mb macro level energy and delay measurements calculated using ViPro



(a) (b)

Figure 7.12: As the number of rows increases, the macro level (a) delay and (b) energy savings provided by the STn SA relative to the convention SA increase

The energy and delay savings provided by reducing σ_{OFFSET} are largely dependent on the BL capacitance. Figure 7.12a shows that as the number of rows is increased from 128 to 512, the macro level delay savings provided by using the STn sense amp, as opposed to the conventional sense amp, increases by 2.4x. This is due to the fact that in designs with a large number of rows, the BL development time dominates the total delay. Therefore, even though the WL pulse width is reduced by 47% in each of the designs, the impact that this has at the

macro level level is more significant in designs with larger BL capacitance. The same trend is apparent in Figure 7.12b, which plots the energy savings versus the number of rows. The resulting energy savings is higher in designs with long BLs because C_{BL} represents a larger percentage of the total capacitance in these designs.

7.4 Conclusion

We have presented methods for reducing the σ_{OFFSET} of a conventional latch based sense amplifier. The source coupled scheme requires no additional devices and provides up to a 19% reduction in σ_{OFFSET}. The topologies presented in sections 2.2 and 2.3 offer significant reductions in sense amp offset at the cost of adding additional devices. The two and three NMOS stack sense amp topologies offer up to a 48% improvement in σ_{OFFSET} for the smallest area overhead. Even though the proposed sense amps have longer resolution times, the savings in BL development time outweigh this overhead. The STn SA requires the largest layout area, but obtains a σ_{OFFSET} of only 14 mV, 34% lower than the minimum σ_{OFFSET} of the conventional latch based SA. Using the proposed sense amps, we have shown that at iso-device width, the STn and stacked topologies offer macro level energy and delay savings of 14% and 10% respectively in the pareto optimal designs.

Acknowledgments

We would like to thank DARPA and NVIDIA for sponsoring this work.

Chapter 8

Conclusions

Technology and voltage scaling have created many new design challenges for SRAMs moving forward. Increases in variability, leakage, and memory capacity, coupled with the reduction of the I_{ON}/I_{OFF} ratio have lead to reductions in functional margins and therefore yield. In addition, the introduction of new applications spaces, such as ultra-low power body sensor nodes (ULP BSNs), calls for further reductions in SRAM V_{MIN} into the sub-threshold region to improve node lifetimes. This dissertation contributes to expanding the SRAM design space in the following ways: evaluating the effect of assist methods on SRAM V_{MIN}, presenting a design methodology for embedded ULP BSN memories, modeling SRAM dynamic V_{MIN}, evaluating the effect of design decisions and V_T variation on the global figures of merit, implementing a canary system to track process, voltage, and temperature variation, and reducing sense amp offset using variation tolerant designs.

8.1 Summary of Contributions

Evaluating alternative bitcells and the effect of assist methods on SRAM V_{MIN}

- Introduced a novel asymmetric Schmitt Trigger bitcell which achieves 86% than the 6T cell and 19% higher RSNM than the 10T ST bitcell.

- Demonstrated that the addition of assist methods leads to larger reductions in V_{MIN} compared to alternative bitcells.
- Demonstrated that sub-threshold bitcells are write-limited, showing that write V_{MIN} was 41% higher than read V_{MIN}.
- Demonstrated BL V_{SS} reduction is the most effective write assist method for reducing write V_{MIN}.
- Demonstrated that WL V_{SS} reduction and bitcell V_{SS} result in the largest reduction of read V_{MIN}.

SRAM design for embedded ULP BSN memories

- Motivated the need for SRAM designs on ULP body sensor nodes for data storage >1kb.
- Presented a design method for implementing SRAM in ULP designs
- Implemented a 2 KB and 4KB instruction and data memory embedded on an ULP BSN capable of operation as low as 0.35V in silicon.
- Implemented a robust read before write timing scheme to prevent half-select instability.
- Highlighted the risk of using multi-V_T designs in sub-threshold.

Modeling SRAM dynamic V_{MIN}

- Demonstrated that modeling the tail of the dynamic write margin using a small Monte Carlo sample is not accurate due to the shape of the distribution.
- Introduced a method using sensitivity analysis that provides a speed up over statistical blockade of 112X with an average percentage error of 3%.
- Demonstrated how this method is used to rapidly predict dynamic write V_{MIN}.

- Demonstrated the tradeoff between WL pulse width and cycle time.
- Demonstrated that negative BL reduction has a greater effect on reducing dynamic write V_{MIN} compared to WL boosting.
- Demonstrated that the advantage of negative BL reduction is reduced as cycle time increases.

Evaluating the effect of design decisions and V_T variation on the global FoMs

- Implemented a tool flow to extend the functionality of ViPro to evaluate the effect of V_T variation on the global FoMs.
- Integrated existing yield models into the tool flow to create virtual prototypes margined to meet the user constrained die yield.
- Extended the functionality of ViPro to evaluate the trade-off between yield, performance, and energy.
- Evaluated the effect of temperature, memory size, and die yield on the critical read and write WL pulse widths.
- Demonstrated that implementing a WL boosting scheme results in an overall energy and delay savings, despite the energy overhead of the charge pump circuit due to an improved read delay distribution.

Canary-based PVT tracking system for reducing dynamic write V_{MIN}

- Demonstrated an optimization approach for canary SRAM systems using order statistics
- Demonstrated that using a PMOS header with a modulated gate voltage to set the canary V_{MIN} is more robust compared to boosting the BL V_{SS} or drooping the WL V_{DD}.

- Demonstrated the trade-off between the size of the canary array and its voltage resolution.
- Demonstrated that aligning the core target V_{MIN} with the canary "sweet spot" allows the core voltage to be tuned closer to its optimal V_{MIN}.
- Demonstrated that the minimum energy point of the canary system depends on the size of the core array
- Demonstrated a 75.97% energy savings in a 1 Mb core array using the canary system compared to guard-banding for the worst case PVT corner.

Stack based sense amplifier designs for reducing input referred offset

- Demonstrated a source coupled sense amplifier that requires no additional devices and provides up to a 19% reduction in σ_{OFFSET}
- Demonstrated two and three NMOS stack designs that offer up to a 48% reduction in σ_{OFFSET} for a small area overhead.
- Demonstrated a STn sense amp that obtains a σ_{OFFSET} of only 14 mV.
- Demonstrated that at iso-device width, the STn and stacked topologies offer macro level energy and delay savings of 14% and 10% respectively in pareto optimal designs.

8.2 Open Problems

This work has addressed many of the challenges in designing reliable SRAMs, however there are still many open questions. As the internet of things (IoT) begins to shift closer towards a reality, the need for ultra low power memory storage becomes greater. To minimize SRAM V_{MIN}, assist methods are commonly employed. However, the trade-off between energy, performance, area, and yield in the sub-threshold regime is still not fully understood. Further

analysis of existing solutions, as well as the combination of peripheral assist methods is needed to identify the optimal solution.

In Chapter 3, we present an embedded SRAM capable of operation down to 0.35 V. Even at 0.35 V, this design consumes roughly 75% of the total digital energy in the form of leakage. To reduce the leakage of this design, it should be implemented using high V_T devices. This presents new challenges due to reduced write margin and performance. The read performance will suffer due to reduced I_{ON} which requires either implementation of a read assist method, or redesign of the array to reduce BL capacitance. This same problem is apparent with the periphery, where driving large BL and WL capacitances could create timing problems. We have shown that the write margin of the HV_T bitcell also requires the use of a peripheral assist method to ensure reliability.

Sub-threshold SRAM implementations have been demonstrated in academic designs, however their use in commercial designs has not been studied. Because of their reduced operating margins, SRAMs are susceptible to soft errors in the form of single event upsets due to alpha particle stikes [98]. Therefore, the use of error correction codes (ECC) will significantly impact reliability. In addition, PVT variation has a more pronounced effect on sub-threshold designs due to their exponential dependence of V_T on drive strength. Therefore, conservative guard-banding for the worst case becomes much more costly in terms of energy and area. To ensure reliable operation in sub-threshold, memory circuits must be able to adapt to process, voltage, and temperature variation in real time.

Typically SRAMs are designed to account for the read and write margin of the worst case cell by accounting for local mismatch. This leads to over-margining in the majority of cells, which creates an opportunity for energy savings. To further reduce the energy of the core array, built-in self tests (BIST) and sensors such as the canary structure described in Chapter 6 must be introduced to track local mismatch. For example, instead of margining a design (e.g. by setting the WL pulse width) for the worst case cell in the entire array, a BIST could enable bank or row level granularity. By operating the cells closer to their critical WL

pulse width (T_{CRIT}), the effect of local mismatch could be greatly reduced.

The implementation of ViPro as described in Chapter 5 has uncovered many open questions. When margining for the worst case read and write margin, we noticed that designs with a high bitline capacitance tended to be read limited, while reducing the BL capacitance lead to write limited designs. In cases where the read and write critical WL pulse widths are very different, it would be interesting to perform an analysis on the potential energy savings of implementing designs with separate read and write WL pulse generators. This would prevent pulsing the WL for longer than is necessary, leading to reductions in BL dissipation. In addition, we found that for some memory sizes, having a large number of columns sharing the same sense amp resulted in improved die yields, while in other cases this was not true. An analysis on how to optimize the number of columns per sense amps across a range of memory sizes would be very useful for improving performance and yield. Finally, as stated earlier, error correction coding is often used for correcting soft errors due to alpha particle strikes, however it would be interesting to measure the potential energy savings of using ECC to correct hard errors. If ECC was able to correct one or multiple errors per word, then the memory could be more aggressively margined, leading to improved performance and energy. The trade-off is that you risk creating more errors than the ECC algorithm can correct which could potentially reduce bit yields.

8.3 Conclusion and Outlook

SRAM has been and will continue to be an important component of processor design as it typically comprises the majority of L1, L2, and L3 caches. As technology continues to scale, memory yield will remain a major challenge. The 8T bitcell has provided improved read stability, as well as the performance benefit of being able to perform a read and write in parallel. However, it still suffers from half-select instability, as well as write-ability concerns in new technology nodes. Many new bitcells have been proposed, but so far none of these designs

have seen commercial use due to their reduced bit density. Based on the work presented in this thesis, the 6T and 8T coupled with the use of read and write assist circuitry remains the most competitive all around solution. As technology continues to scale, the need to balance these assist methods to optimize energy, performance, area, and yield presents a major challenge. This challenge has long been apparent in sub-threshold memory design, but has recently become evident in highly scaled technology nodes.

In addition on yield concerns, SRAM leakage has recently become a major source of power consumption. To combat leakage concerns, many non-CMOS embedded solutions such as spin-tronics, resistive RAM, and phase change memory have been proposed as possible solutions. Currently SRAM remains as the most competitive option, as none of the emerging solutions have been able to match its performance. The biggest upside of the emerging solutions is the reduction in leakage and the ability to retain state after shut down. These technologies have the potential to provide a universal memory solution, which could have a profound effect on the field of electronics.

To continue to advance circuit and system design, new challenges must be addressed and solutions identified. The ability to evaluate new solutions and trade-offs has pushed the need for improved CAD and human interactions. Because today's systems are very complex, it is virtually impossible to perform optimization without the help of CAD tools. In Chapter 5 we presented ViPro, a tool for exploring the SRAM design space. This tool has proven to extremely valuable, as it was used throughout Chapters 6 and 7 to evaluate the effects of canary and sense amp design on the figures of merit. Many of the results and insights gained through the use of ViPro are non-intuitive, which makes it great tool for aiding design engineers. Because the tool uses hierarchical modeling, it can easily be extended to evaluate emerging memory technologies. In addition, the structure can be extended to any system where there is a need to model complex block to block interactions.

The internet of things has created a potentially huge market for ultra low power sensors. Because the processing power required by these types of sensors is relatively low and the need

for long node lifetimes is high, the sub-threshold region is ideal because it minimizes energy per operation. This has been known for many years, however very few, if any, commercial products have been released which take advantage of this operating region. Because of this, little is known about designing sub-threshold devices for mass commercial production. For the idea of the internet of things to become a reality, this design space must further be explored and understood.

Appendix A

Publications

JB1 F. Zhang, Y. Zhang, J. Silver, Y. Shakhsheer, M. Nagaraju, A. Klinefelter, J. Pandey, J. Boley, E. Carlson, A. Shrivastava, B. Otis, and B. H. Calhoun, "A Battery-less 19μW MICS/ISM-Band Energy Harvesting Body Area Sensor Node SoC," International Solid-State Circuits Conference, February 2012.

JB2 J. Boley, J. Wang, B. H. Calhoun, "Analyzing Sub-Threshold Bitcell Topologies and the Effects of Assist Methods on SRAM V_{MIN}." Journal of Low Power Electronics and Applications, 2012.

JB3 Y. Zhang, F. Zhang, Y. Shakhsheer, J. D. Silver, A. Klinefelter, M. Nagaraju, J. Boley, J. N. Pandey, A. Shrivastava, E. J. Carlson, A. Wood, B. H. Calhoun, and B. Otis, "A Batteryless 19μW MICS/ISM-Band Energy Harvesting Body Sensor Node SoC for ExG Applications," Journal of Solid State Circuits, vol. 48, issue 1, pp. 199-213, January 2013.

JB4 J. Boley, V. Chandra, R. Aitken, and B. H. Calhoun, "Leveraging Sensitivity Analysis for Fast, Accurate Estimation of SRAM Dynamic Write V_{MIN}," Design, Automation and Test in Europe, March 2013.

JB5 J. Boley, V. Chandra, R. Aitken, and B. H. Calhoun, "Modeling SRAM dynamic V_{MIN}," International Conference on IC Design and Technology, May 2014.

JB6 A. Klinefelter, N. Roberts, Y. Shakhsheer, P. Gonzalez, A. Shrivastava, A. Roy, K. Craig, M. Faisal, J. Boley, S. Oh, Y. Zhang, D. Akella, D. Wentzloff, and B. H. Calhoun, "A 6.45 μW Self-Powered IoT SoC with Integrated Energy-Harvesting Power Management and ULP Asymmetric Radios," International Solid-State Circuits Conference, February 2015.

JB7 J. Boley and B. H. Calhoun, "Stack Based Sense Amplifier Designs for Reducing Input-Referred Offset," International Symposium on Quality Electronic Design, March 2015.

JB8 J. Boley, P. Beshay, and B. H. Calhoun, "Virtual Prototyper (ViPro): An SRAM Design Tool for Yield Constrained Optimization," Transactions on Very Large Scale Integration Systems, (Under Review)

Patents

JB9 J. Boley, "Stack Based Sense Amplifier Designs for Reducing Input-Referred Offset," US Patent Application

Bibliography

[1] B.H. Calhoun and A.P. Chandrakasan. Static noise margin variation for sub-threshold sram in 65-nm cmos. *Solid-State Circuits, IEEE Journal of*, 41(7):1673–1679, July 2006.

[2] J.F. Ryan, S. Khanna, and B.H. Calhoun. An analytical model for performance yield of nanoscale sram accounting for the sense amplifier strobe signal. In *Low Power Electronics and Design (ISLPED) 2011 International Symposium on*, pages 297–302, Aug 2011.

[3] Jiajing Wang and B.H. Calhoun. Techniques to extend canary-based standby v_{DD} scaling for srams to 45 nm and beyond. *Solid-State Circuits, IEEE Journal of*, 43(11):2514–2523, Nov 2008.

[4] N. Verma and A.P. Chandrakasan. A 256 kb 65 nm 8t subthreshold sram employing sense-amplifier redundancy. *Solid-State Circuits, IEEE Journal of*, 43(1):141–149, 2008.

[5] J.P. Kulkarni, Keejong Kim, and K. Roy. A 160 mv robust schmitt trigger based subthreshold sram. *Solid-State Circuits, IEEE Journal of*, 42(10):2303–2313, Oct 2007.

[6] S. Nalam and B.H. Calhoun. Asymmetric sizing in a 45nm 5t sram to improve read stability over 6t. In *Custom Integrated Circuits Conference, 2009. CICC '09. IEEE*, pages 709–712, Sept 2009.

[7] M.A. Hanson, H.C. Powell, A.T. Barth, K. Ringgenberg, B.H. Calhoun, J.H. Aylor, and J. Lach. Body area sensor networks: Challenges and opportunities. *Computer*, 42(1):58–65, 2009.

[8] James Boley, Alicia Klinefelter, and B.H. Calhoun. Memory challenges and opportunities for body sensor node socs. *Unpublished Manuscript*, April 2014.

[9] Jiajing Wang, A. Hoefler, and B.H. Calhoun. An enhanced canary-based system with bist for sram standby power reduction. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 19(5):909–914, May 2011.

[10] Y. Nakagome, M. Horiguchi, T. Kawahara, and K. Itoh. Review and future prospects of low-voltage ram circuits. *IBM Journal of Research and Development*, 47(5.6):525–552, Sept 2003.

[11] E. Seevinck, F.J. List, and J. Lohstroh. Static-noise margin analysis of mos sram cells. *Solid-State Circuits, IEEE Journal of*, 22(5):748–754, Oct 1987.

[12] M Rabaey Jan, Chandrakasan Anantha, and N Borivoje. Digital integrated circuits–a design perspective, 2004.

[13] Zheng Guo, A. Carlson, Liang-Teck Pang, K. Duong, Tsu-Jae King Liu, and B. Nikolic. Large-scale read/write margin measurement in 45nm cmos sram arrays. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 42–43, June 2008.

[14] Jiajing Wang, A. Singhee, R.A. Rutenbar, and B.H. Calhoun. Statistical modeling for the minimum standby supply voltage of a full sram array. In *Solid State Circuits Conference, 2007. ESSCIRC 2007. 33rd European*, pages 400–403, Sept 2007.

[15] Jiajing Wang and B.H. Calhoun. Minimum supply voltage and yield estimation for large srams under parametric variations. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 19(11):2120–2125, Nov 2011.

[16] Jiajing Wang and B.H. Calhoun. Canary replica feedback for near-drv standby vdd scaling in a 90nm sram. In *Custom Integrated Circuits Conference, 2007. CICC '07. IEEE*, pages 29–32, Sept 2007.

[17] A. Bhavnagarwala, S. Kosonocky, C. Radens, K. Stawiasz, R. Mann, Qiuyi Ye, and Ken Chin. Fluctuation limits amp; scaling opportunities for cmos sram cells. In *Electron Devices Meeting, 2005. IEDM Technical Digest. IEEE International*, pages 659–662, Dec 2005.

[18] A. Wang, A.P. Chandrakasan, and S.V. Kosonocky. Optimal supply and threshold scaling for subthreshold cmos circuits. In *VLSI, 2002. Proceedings. IEEE Computer Society Annual Symposium on*, pages 5–9, 2002.

[19] R.W. Mann, S. Nalam, Jiajing Wang, and B.H. Calhoun. Limits of bias based assist methods in nano-scale 6t sram. In *Quality Electronic Design (ISQED), 2010 11th International Symposium on*, pages 1–8, March 2010.

[20] O. Hirabayashi, A. Kawasumi, A. Suzuki, Y. Takeyama, K. Kushida, T. Sasaki, A. Katayama, G. Fukano, Y. Fujimura, T. Nakazato, Y. Shizuki, N. Kushiyama, and T. Yabe. A process-variation-tolerant dual-power-supply sram with 0.179 um2 cell in 40nm cmos using level-programmable wordline driver. In *Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International*, pages 458–459,459a, Feb 2009.

[21] D.P. Wang, H.J. Liao, H. Yamauchi, Y.H. Chen, Y.L. Lin, S.H. Lin, D.C. Liu, H.C. Chang, and W. Hwang. A 45nm dual-port sram with write and read capability enhancement at low voltage. In *SOC Conference, 2007 IEEE International*, pages 211–214, Sept 2007.

[22] K. Nii, M. Yabuuchi, Y. Tsukamoto, S. Ohbayashi, Y. Oda, K. Usui, T. Kawamura, N. Tsuboi, T. Iwasaki, K. Hashimoto, H. Makino, and H. Shinohara. A 45-nm single-port and dual-port sram family with robust read/write stabilizing circuitry under dvfs environment. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 212–213, June 2008.

[23] Y.H. Chen, W.M. Chan, S.Y. Chou, H.J. Liao, H.Y. Pan, J.J. Wu, C.H. Lee, S.M. Yang, Y.C. Liu, and H. Yamauchi. A 0.6v 45nm adaptive dual-rail sram compiler circuit design for lower vddmin vlsis. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 210–211, June 2008.

[24] Yeonbae Chung and Seung-Ho Song. Implementation of low-voltage static {RAM} with enhanced data stability and circuit speed. *Microelectronics Journal*, 40(6):944 – 951, 2009.

[25] K. Zhang, U. Bhattacharya, Z. Chen, F. Hamzaoglu, D. Murray, N. Vallepalli, Y. Wang, B. Zheng, and M. Bohr. A 3-ghz 70mb sram in 65nm cmos technology with integrated column-based dynamic power supply. In *Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC. 2005 IEEE International*, pages 474–611 Vol. 1, Feb 2005.

[26] M. Yamaoka, K. Osada, and K. Ishibashi. 0.4-v logic-library-friendly sram array using rectangular-diffusion cell and delta-boosted-array voltage scheme. *Solid-State Circuits, IEEE Journal of*, 39(6):934–940, June 2004.

[27] Masaaki Iijima, Kayoko Seto, Masahiro Numa, Akira Tada, and Takashi Ipposhi. Low power sram with boost driver generating pulsed word line voltage for sub-1v operation. *Journal of Computers*, 3(5), 2008.

[28] H.S. Yang, R. Wong, R. Hasumi, Y. Gao, N-S Kim, D.H. Lee, S. Badrudduza, D. Nair, M. Ostermayr, H. Kang, H. Zhuang, J. Li, L. Kang, X. Chen, A. Thean, F. Arnaud, L. Zhuang, C. Schiller, D.P. Sun, Y.W. Teh, J. Wallner, Y. Takasu, K. Stein, S. Samavedam, D. Jaeger, C.V. Baiocco, M. Sherony, M. Khare, C. Lage, J. Pape, J. Sudijono, A.L. Steegen, and S. Stiffler. Scaling of 32nm low power sram with high-k metal gate. In *Electron Devices Meeting, 2008. IEDM 2008. IEEE International*, pages 1–4, Dec 2008.

[29] N. Shibata, H. Kiya, S. Kurita, H. Okamoto, M. Tan'no, and T. Douseki. A 0.5-v 25-mhz 1-mw 256-kb mtcmos/soi sram for solar-power-operated portable personal digital equipment - sure write operation by using step-down negatively overdriven bitline scheme. *Solid-State Circuits, IEEE Journal of*, 41(3):728–742, March 2006.

[30] B.H. Calhoun and A. Chandrakasan. A 256kb sub-threshold sram in 65nm cmos. In *Solid-State Circuits Conference, 2006. ISSCC 2006. Digest of Technical Papers. IEEE International*, pages 2592–2601, Feb 2006.

[31] S.A. Vitale, P.W. Wyatt, N. Checka, J. Kedzierski, and C.L. Keast. Fdsoi process technology for subthreshold-operation ultralow-power electronics. *Proceedings of the IEEE*, 98(2):333–342, Feb 2010.

[32] G. Z. Yang. *Body Sensor Networks*. Springer-Verlag, 2006.

[33] E.J. Carlson, K. Strunz, and B.P. Otis. A 20 mv input boost converter with efficient digital control for thermoelectric energy harvesting. *Solid-State Circuits, IEEE Journal of*, 45(4):741–750, 2010.

[34] Mingoo Seok, S. Hanson, Yu-Shiang Lin, Zhiyoong Foo, Daeyeon Kim, Yoonmyung Lee, N. Liu, D. Sylvester, and D. Blaauw. The phoenix processor: A 30pw platform for sensor applications. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 188–189, June 2008.

[35] Yanqing Zhang, Fan Zhang, Y. Shakhsheer, J.D. Silver, A. Klinefelter, M. Nagaraju, J. Boley, J. Pandey, A. Shrivastava, E.J. Carlson, A. Wood, B.H. Calhoun, and B.P. Otis. A batteryless 19 w mics/ism-band energy harvesting body sensor node soc for exg applications. *Solid-State Circuits, IEEE Journal of*, 48(1):199–213, 2013.

[36] M. Zwerg, A. Baumann, R. Kuhn, M. Arnold, R. Nerlich, M. Herzog, R. Ledwa, C. Sichert, V. Rzehak, P. Thanigai, and B.O. Eversmann. An 82 ua/mhz microcontroller with embedded feram for energy-harvesting applications. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, pages 334–336, Feb 2011.

[37] M. Khayatzadeh, Xiaoyang Zhang, Jun Tan, Wen-Sin Liew, and Yong Lian. A 0.7-v 17.4-/spl mu/w 3-lead wireless ecg soc. *Biomedical Circuits and Systems, IEEE Transactions on*, 7(5):583–592, Oct 2013.

[38] Y. Shakhsheer, Y. Zhang, B. Otis, and B.H. Calhoun. A custom processor for node and power management of a battery-less body sensor node in 130nm cmos. In *Custom Integrated Circuits Conference (CICC), 2012 IEEE*, pages 1–4, 2012.

[39] S.C. Bartling, S. Khanna, M.P. Clinton, S.R. Summerfelt, J.A. Rodriguez, and H.P. McAdams. An 8mhz 75 ua/mhz zero-leakage non-volatile logic-based cortex-m0 mcu soc exhibiting 100and sleep transitions. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, pages 432–433, Feb 2013.

[40] N. Gilbert, Yanqing Zhang, J. Dinh, B. Calhoun, and S. Hollmer. A 0.6v 8 pj/write non-volatile cbram macro embedded in a body sensor node for ultra low energy applications. In *VLSI Circuits (VLSIC), 2013 Symposium on*, pages C204–C205, June 2013.

[41] P. Meinerzhagen, C. Roth, and A. Burg. Towards generic low-power area-efficient standard cell based memory architectures. In *Circuits and Systems (MWSCAS), 2010 53rd IEEE International Midwest Symposium on*, pages 129–132, Aug 2010.

[42] Long Yan, Joonsung Bae, Seulki Lee, Taehwan Roh, Kiseok Song, and Hoi-Jun Yoo. A 3.9 mw 25-electrode reconfigured sensor for wearable cardiac monitoring system. *Solid-State Circuits, IEEE Journal of*, 46(1):353–364, Jan 2011.

[43] Hyejung Kim, Sunyoung Kim, N. Van Helleputte, A. Artes, M. Konijnenburg, J. Huisken, C. van Hoof, and R.F. Yazicioglu. A configurable and low-power mixed signal soc for portable ecg monitoring applications. *Biomedical Circuits and Systems, IEEE Transactions on*, 8(2):257–267, April 2014.

[44] G. Chen, M. Fojtik, Daeyeon Kim, D. Fick, Junsun Park, Mingoo Seok, Mao-Ter Chen, Zhiyoong Foo, D. Sylvester, and D. Blaauw. Millimeter-scale nearly perpetual sensor system with stacked battery and solar cells. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 288–289, Feb 2010.

[45] Ik Joon Chang, Jae-Joon Kim, Sang Phill Park, and K. Roy. A 32 kb 10t sub-threshold sram array with bit-interleaving and differential read scheme in 90 nm cmos. *Solid-State Circuits, IEEE Journal of*, 44(2):650–658, Feb 2009.

[46] OpenCores. openmsp430, 2012.

[47] S. Ohbayashi, M. Yabuuchi, K. Nii, Y. Tsukamoto, S. Imaoka, Y. Oda, T. Yoshihara, M. Igarashi, M. Takeuchi, H. Kawashima, Y. Yamaguchi, K. Tsukamoto, M. Inuishi, H. Makino, K. Ishibashi, and H. Shinohara. A 65-nm soc embedded 6t-sram designed for manufacturability with read and write operation stabilizing circuits. *Solid-State Circuits, IEEE Journal of*, 42(4):820–829, April 2007.

[48] Y. Morita, H. Fujiwara, H. Noguchi, Y. Iguchi, K. Nii, H. Kawaguchi, and M. Yoshimoto. An area-conscious low-voltage-oriented 8t-sram design under dvs environment. In *VLSI Circuits, 2007 IEEE Symposium on*, pages 256–257, June 2007.

[49] Tae-Hyoung Kim, J. Liu, J. Keane, and C.H. Kim. A high-density subthreshold sram with data-independent bitline leakage and virtual ground replica scheme. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 330–606, Feb 2007.

[50] Tae-Hyoung Kim, J. Liu, J. Keane, and C.H. Kim. Circuit techniques for ultra-low power subthreshold srams. In *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, pages 2574–2577, May 2008.

[51] A.J. Bhavnagarwala, X. Tang, and J.D. Meindl. The impact of intrinsic device fluctuations on cmos sram cell stability. *Solid-State Circuits, IEEE Journal of*, 36(4):658–665, Apr 2001.

[52] D.E. Khalil, M. Khellah, Nam-Sung Kim, Y. Ismail, T. Karnik, and V.K. De. Accurate estimation of sram dynamic stability. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 16(12):1639–1647, Dec 2008.

[53] Bin Zhang, A. Arapostathis, S. Nassif, and M. Orshansky. Analytical modeling of sram dynamic stability. In *Computer-Aided Design, 2006. ICCAD '06. IEEE/ACM International Conference on*, pages 315–322, Nov 2006.

[54] S. Mukhopadhyay, H. Mahmoodi, and K. Roy. Modeling of failure probability and statistical design of sram array for yield enhancement in nanoscaled cmos. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(12):1859–1880, Dec 2005.

[55] 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 *Design Automation Conference, 2006 43rd ACM/IEEE*, pages 69–72, 2006.

[56] T.S. Doorn, E.J.W. ter Maten, J.A. Croon, A. Di Bucchianico, and O. Wittich. Importance sampling monte carlo simulations for accurate estimation of sram yield. In *Solid-State Circuits Conference, 2008. ESSCIRC 2008. 34th European*, pages 230–233, Sept 2008.

[57] 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 *Design, Automation Test in Europe Conference Exhibition, 2007. DATE '07*, pages 1–6, April 2007.

[58] A. Singhee, Jiajing Wang, B.H. Calhoun, and R.A. Rutenbar. Recursive statistical blockade: An enhanced technique for rare event simulation with application to sram circuit design. In *VLSI Design, 2008. VLSID 2008. 21st International Conference on*, pages 131–136, Jan 2008.

[59] M. Sharifkhani and M. Sachdev. Sram cell stability: A dynamic perspective. *Solid-State Circuits, IEEE Journal of*, 44(2):609–619, Feb 2009.

[60] Wei Dong, Peng Li, and G.M. Huang. Sram dynamic stability: Theory, variability and analysis. In *Computer-Aided Design, 2008. ICCAD 2008. IEEE/ACM International Conference on*, pages 378–385, Nov 2008.

[61] Jiajing Wang, S. Nalam, and B.H. Calhoun. Analyzing static and dynamic write margin for nanometer srams. In *Low Power Electronics and Design (ISLPED), 2008 ACM/IEEE International Symposium on*, pages 129–134, Aug 2008.

[62] S. Nalam, V. Chandra, R.C. Aitken, and B.H. Calhoun. Dynamic write limited minimum operating voltage for nanoscale srams. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2011*, pages 1–6, March 2011.

[63] Seng Oon Toh, Zheng Guo, and B. Nikolic. Dynamic sram stability characterization in 45nm cmos. In *VLSI Circuits (VLSIC), 2010 IEEE Symposium on*, pages 35–36, June 2010.

[64] M. Yamaoka, K. Osada, and T. Kawahara. A cell-activation-time controlled sram for low-voltage operation in dvfs socs using dynamic stability analysis. In *Solid-State Circuits Conference, 2008. ESSCIRC 2008. 34th European*, pages 286–289, Sept 2008.

[65] Y. Tsukamoto, K. Nii, S. Imaoka, Y. Oda, S. Ohbayashi, T. Yoshizawa, H. Makino, K. Ishibashi, and H. Shinohara. Worst-case analysis to obtain stable read/write dc margin of high density 6t-sram-array with local vth variability. In *Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, pages 398–405, Nov 2005.

[66] Randy W. Mann, Jiajing Wang, Satyanand Nalam, Sudhanshu Khanna, Geordie Braceras, Harold Pilo, and Benton H. Calhoun. Impact of circuit assist methods on margin and performance in 6t {SRAM}. *Solid-State Electronics*, 54(11):1398 – 1407, 2010.

[67] V. Chandra, C. Pietrzyk, and R. Aitken. On the efficacy of write-assist techniques in low voltage nanoscale srams. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pages 345–350, March 2010.

[68] S. Nalam, M. Bhargava, Ken Mai, and B.H. Calhoun. Virtual prototyper (vipro): An early design space exploration and optimization tool for sram designers. In *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, pages 138–143, June 2010.

[69] S.J.E. Wilton and N.P. Jouppi. Cacti: an enhanced cache access and cycle time model. *Solid-State Circuits, IEEE Journal of*, 31(5):677–688, May 1996.

[70] N. Muralimanohar, R. Balasubramonian, and N. Jouppi. Optimizing nuca organizations and wiring alternatives for large caches with cacti 6.0. In *Microarchitecture, 2007. MICRO 2007. 40th Annual IEEE/ACM International Symposium on*, pages 3–14, Dec 2007.

[71] ITRS. Internation roadmap for semiconductors, 2013.

[72] S. Nalam, M. Bhargava, K. Ringgenberg, Ken Mai, and B.H. Calhoun. A technology-agnostic simulation environment (tase) for iterative custom ic design across processes. In *Computer Design, 2009. ICCD 2009. IEEE International Conference on*, pages 523–528, Oct 2009.

[73] Xiaoliang Bai, C. Visweswariah, P.N. Strenski, and D.J. Hathaway. Uncertainty-aware circuit optimization. In *Design Automation Conference, 2002. Proceedings. 39th*, pages 58–63, 2002.

[74] V. Sundararajan, S.S. Sapatnekar, and K.K. Parhi. Fast and exact transistor sizing based on iterative relaxation. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 21(5):568–581, May 2002.

[75] D. Patil, O. Azizi, M. Horowitz, R. Ho, and R. Ananthraman. Robust energy-efficient adder topologies. In *Computer Arithmetic, 2007. ARITH '07. 18th IEEE Symposium on*, pages 16–28, June 2007.

[76] K. Chakraborty, S. Kulkami, M. Bhattacharya, P. Mazumder, and A. Gupta. A physical design tool for built-in self-repairable rams. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 9(2):352–364, April 2001.

[77] Satyanand Nalam. *Circuit and CAD Solutions for Optimal SRAM Design in Nanoscale CMOS*. PhD thesis, University of Virginia, December 2011.

[78] Tomohisa Mizuno, J. Okumtura, and Akira Toriumi. Experimental study of threshold voltage fluctuation due to statistical variation of channel dopant number in mosfet's. *Electron Devices, IEEE Transactions on*, 41(11):2216–2221, Nov 1994.

[79] T. Fischer, C. Otte, D. Schmitt-Landsiedel, E. Amirante, A. Olbrich, P. Huber, M. Ostermayr, T. Nirschl, and J. Einfeld. A 1 mbit sram test structure to analyze local mismatch beyond 5 sigma variation. In *Microelectronic Test Structures, 2007. ICMTS '07. IEEE International Conference on*, pages 63–66, March 2007.

[80] M.J.M. Pelgrom, Aad C.J. Duinmaijer, and A.P.G. Welbers. Matching properties of mos transistors. *Solid-State Circuits, IEEE Journal of*, 24(5):1433–1439, Oct 1989.

[81] Altera. Altera, 2014.

[82] David Wolpert and Paul Ampadu. Temperature effects in semiconductors, 2012.

[83] B.S. Amrutur and M.A. Horowitz. A replica technique for wordline and sense control in low-power sram's. *Solid-State Circuits, IEEE Journal of*, 33(8):1208–1219, Aug 1998.

[84] M. Khellah, Yibin Ye, Nam Sung Kim, D. Somasekhar, G. Pandya, A. Farhang, K. Zhang, C. Webb, and V. De. Wordline amp; bitline pulsing schemes for improving sram cell stability in low-vcc 65nm cmos designs. In *VLSI Circuits, 2006. Digest of Technical Papers. 2006 Symposium on*, pages 9–10, 2006.

[85] Jianhui Wu, Jiafeng Zhu, YingCheng Xia, and Na Bai. A multiple-stage parallel replica-bitline delay addition technique for reducing timing variation of sram sense amplifiers. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, 61(4):264–268, April 2014.

[86] Y. Niki, A. Kawasumi, A. Suzuki, Y. Takeyama, O. Hirabayashi, K. Kushida, F. Tachibana, Y. Fujimura, and T. Yabe. A digitized replica bitline delay technique for random-variation-tolerant timing generation of sram sense amplifiers. In *Solid State Circuits Conference (A-SSCC), 2010 IEEE Asian*, pages 1–4, Nov 2010.

[87] Y. Niki, A. Kawasumi, A. Suzuki, Y. Takeyama, O. Hirabayashi, K. Kushida, F. Tachibana, Y. Fujimura, and T. Yabe. A digitized replica bitline delay technique for random-variation-tolerant timing generation of sram sense amplifiers. *Solid-State Circuits, IEEE Journal of*, 46(11):2545–2551, Nov 2011.

[88] Meng-Fan Chang, Shu-Meng Yang, Kuang-Ting Chen, Hung-Jen Liao, and R. Lee. Improving the speed and power of compilable sram using dual-mode self-timed technique. In *Memory Technology, Design and Testing, 2007. MTDT 2007. IEEE International Workshop on*, pages 57–60, Dec 2007.

[89] B.H. Calhoun and A.P. Chandrakasan. Standby power reduction using dynamic voltage scaling and canary flip-flop structures. *Solid-State Circuits, IEEE Journal of*, 39(9):1504–1511, Sept 2004.

[90] Y. Otsuka, T. Sato, T. Yoshiki, and T. Hayashida. Multicore energy reduction utilizing canary ff. In *Communications and Information Technologies (ISCIT), 2010 International Symposium on*, pages 922–927, Oct 2010.

[91] A. Banerjee, M.E. Sinangil, J. Poulton, C.T. Gray, and B.H. Calhoun. A reverse write assist circuit for sram dynamic write vmin tracking using canary srams. In *Quality Electronic Design (ISQED), 2014 15th International Symposium on*, pages 1–8, March 2014.

[92] H. N. Nagaraja H.A David. Temperature effects in semiconductors, 2003.

[93] M.H. Abu-Rahma, Ying Chen, Wing Sy, Wee Ling Ong, Leon Yeow Ting, Sei Seung Yoon, M. Han, and E. Terzioglu. Characterization of sram sense amplifier input offset for yield prediction in 28nm cmos. In *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, pages 1–4, Sept 2011.

[94] P. Elakkumanan, J.B. Kuang, K. Nowka, R. Sridhar, R. Kanj, and S. Nassif. Sram local bit line access failure analyses. In *Quality Electronic Design, 2006. ISQED '06. 7th International Symposium on*, pages 6 pp.–209, March 2006.

[95] S.R. Nassif. Modeling and analysis of manufacturing variations. In *Custom Integrated Circuits, 2001, IEEE Conference on.*, pages 223–228, 2001.

[96] M. Bhargava, M.P. McCartney, A. Hoefler, and Ken Mai. Low-overhead, digital offset compensated, sram sense amplifiers. In *Custom Integrated Circuits Conference, 2009. CICC '09. IEEE*, pages 705–708, Sept 2009.

[97] R. Singh and N. Bhat. An offset compensation technique for latch type sense amplifiers in high-speed low-power srams. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 12(6):652–657, June 2004.

[98] R. Baumann. The impact of technology scaling on soft error rate performance and limits to the efficacy of error correction. In *Electron Devices Meeting, 2002. IEDM '02. International*, pages 329–332, Dec 2002.

[99] B.H. Calhoun, J. Lach, J. Stankovic, D.D. Wentzloff, K. Whitehouse, A.T. Barth, J.K. Brown, Qiang Li, Seunghyun Oh, N.E. Roberts, and Yanqing Zhang. Body sensor networks: A holistic approach from silicon to users. *Proceedings of the IEEE*, 100(1):91–106, 2012.

[100] Fan Zhang, Yanqing Zhang, J. Silver, Y. Shakhsheer, M. Nagaraju, A. Klinefelter, J. Pandey, J. Boley, E. Carlson, A. Shrivastava, B. Otis, and B. Calhoun. A batteryless 19 uw mics/ism-band energy harvesting body area sensor node soc. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 298–300, 2012.