Deep Sub-Micron SRAM Design for Ultra-Low Leakage Standby Operation

Huifang Qin

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2007-74
http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-74.html

May 23, 2007

Copyright © 2007, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Deep Sub-Micron SRAM Design for Ultra-Low Leakage Standby Operation

by

Huifang Qin

B. Engr. (Tsinghua University) 2000

A dissertation submitted in partial satisfaction of the
requirements for the degree of

Doctor of Philosophy

in

Engineering – Electrical Engineering and Computer Sciences

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Jan M. Rabaey, Chair
Professor Borivoje Nikolic
Professor David R. Brillinger

Spring 2007

Deep Sub-Micron SRAM Design for Ultra-Low Leakage Standby Operation

Copyright © 2007

by

Huifang Qin

Abstract

Deep Sub-Micron SRAM Design for Ultra-Low Leakage Standby Operation

by

Huifang Qin

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Jan M. Rabaey, Chair

Suppressing the standby current in memories is critical in low-power design. By lowering the supply voltage (V_{DD}) to its standby limit, the data retention voltage (*DRV*), SRAM leakage power can be reduced substantially. The *DRV* theoretical limit is derived to be 52mV for a 90nm technology at room temperature. The *DRV* increases with transistor mismatches. Based on sub-threshold circuit analysis, a practical *DRV* model is developed and verified with measurement data from several test chips in 130nm and 90nm technologies. By reducing the standby V_{DD} of a 32K-bit 130nm industrial IP SRAM module to 490 mV (390 mV worst-case *DRV* + 100 mV electrical-noise guard-band), an 85% leakage power saving is measured, compared to the standby power at 1V.

Since the *DRV* is a strong function of both process and design parameters, the SRAM cell can be optimized to reduce *DRV*. It is shown that the body bias and device channel length are the most effective knobs in minimizing *DRV*. This is confirmed with measurement data from a 90nm SRAM test chip. Building on these results, feasibility of a 270mV standby V_{DD} is demonstrated for an optimized 4K-bit SRAM in a 90nm technology, leading to a 97% leakage power reduction. By dynamically configuring the

body bias during read and write operations, the active operation noise margins and data access speed are also improved according to simulation results.

Correcting the low-voltage retention errors with error correction code (ECC) provides another opportunity to further reduce the SRAM standby V_{DD}. To establish a power-per-bit metric, the SRAM leakage power is modeled as a function of the ECC parameters, *DRV* distribution and the standby V_{DD}. This power metric is optimized based on ECC theory to obtain fundamental bounds of the power saving enabled by error-tolerant design. Taking into account the practical design requirements, an error-tolerant SRAM design with a (31, 26) Hamming code is proposed introducing a further power reduction of 33%.

Both the circuit optimization and the error-tolerant architecture are implemented in a 90nm 26K-bit ultra-low leakage SRAM chip. Measurement result proves that the memory data can be reliably retained at a 255mV standby V_{DD}, with a 50X leakage power reduction. While the optimization also improves active SRAM operation, the only tradeoff is a 50% larger area caused by the larger channel length and ECC overhead.

In summary, this work is the first analytical investigation into the voltage limit of SRAM standby operation. The theoretical and practical *DRV* models provide insights to the future low-voltage SRAM designs. Besides the analytical study, we also develop two novel design solutions that aggressively reduce SRAM leakage. The error-tolerant SRAM standby scheme is the first time the ECC is used for memory power minimization.

Professor Jan M. Rabaey
Thesis Committee Chair

Contents

List of figures iii

List of tables v

1 Introduction 1

1.1 Existing Work 2

1.1.1 The novel SRAM cells 2

1.1.2 Dynamic biasing techniques 3

1.1.3 V_{DD}-gating techniques 4

1.2 Contribution 6

1.3 Thesis Organization 8

2 SRAM Data Retention Voltage (*DRV*) Analysis 9

2.1 *DRV* Definition 10

2.2 *DRV* Theoretical Lower Bound 11

2.3 *DRV* Model with Process Variations 15

2.3.1 DRV of a realistic SRAM cell 15

2.3.2 DRV sensitivity to variations 16

2.3.3 DRV model with variations 20

2.4 Low-Voltage SRAM Standby Stability Analysis 22

2.5 SRAM Standby Leakage Modeling 25

3 Measured SRAM *DRV* and its Evolution into the Future 27

3.1 *DRV* Measured in 130nm Technology 27

3.1.1 Dual supply design considerations 28

3.1.2 Test chip implementation 30

3.1.3 Measurement results 32

3.2 *DRV* Measured in 90nm Technology 37
3.2.1 Test chip design and implementation 37
3.2.2 Measurement results 39
3.3 *DRV* Scaling Trend 44
4 *DRV*-Aware SRAM Cell Design 46
4.1 *DRV* Design Model Based on the 90nm Technology Data 48
4.1.1 DRV design model 49
4.1.2 Model verification 53
4.2 *DRV*-Aware SRAM Cell Optimization Methodology 54
4.2.1 Worst case DRV minimization 54
4.2.2 Leakage power minimization 57
4.2.3 The optimization impact on active operation metrics 60
4.3 90nm SRAM *DRV*-Aware Design Optimization Summary 63
5 Error-Tolerant SRAM Design for Ultra-Low Power Standby 65
5.1 ECC Analysis for Low Voltage SRAM 65
5.1.1 Modeling the SRAM standby power 65
5.1.2 Power per useful bit bounds 67
5.1.3 Code implementation 68
5.2 An Implementation of Ultra-Low Leakage Error-Tolerant SRAM 69
5.2.1 Chip design 69
5.2.2 Measurement results 71
5.2.3 SER improvement 74
6 Conclusion 76
References 78

List of figures

Figure 2.1. Standard 6T SRAM cell structure. (a) 6T SRAM cell in standby (assuming V_1≈0 and V_2≈V_{DD}.). (b) Flip-flop representation of the same SRAM cell. 9
Figure 2.2. An illustration of SRAM inverter VTC deterioration under low-V_{DD}. The SRAM cell noise margin is zero at *DRV*. This VTC simulation assumes 3σ worst-case local mismatches among the SRAM cell transistors. 11
Figure 2.3. VTC of SRAM cell inverters under 3σ variation in *L* and V_{th}. (Solid lines: ideal case with no variation.) 17
Figure 2.4. *DRV* sensitivity to local and global variations. 19
Figure 2.5. SRAM cell *DRV* under process and temperature variations. 20
Figure 2.6. Flip-flop representation of SRAM cell with inserted static noise, V_n. 22
Figure 2.7. Static noise margin (*SNM*) as a function of V_{DD}. Slope of a first-order linear model agrees with simulation results. 24
Figure 3.1. SRAM low-voltage standby leakage suppression scheme. 28
Figure 3.2. A 130nm SRAM leakage-control test chip. 30
Figure 3.3. An SC converter optimized for 20μW output load 31
Figure 3.4. Waveform of *DRV* measurement. (a) *DRV* = 190mV in SRAM cell 1 with state "1", (b) *DRV* = 180mV in SRAM cell 2 with state "0". 32
Figure 3.5. Measured *DRV* distribution of a 32K-bit SRAM chip. 34
Figure 3.6. *DRV* spatial distribution of a 32K-bit SRAM chip. 35
Figure 3.7. Measured leakage current of a 32K-bit SRAM chip. 36
Figure 3.8. SRAM *DRV*-aware design optimization test chip in 90nm technology. 39
Figure 3.9. *DRV* sensitivity on design parameters (measured data from one chip). 40
Figure 3.10. *DRV* sensitivity to body bias (standard size array) 42
Figure 3.11. *DRV* sensitivity to *L* (zero body bias, standard *W/L* ratio) 42
Figure 3.12. *DRV* sensitivity to *W/L* sizing ratio (zero body bias, standard *L*) 43

Figure 3.13. *DRV* and V_{DD} scaling trend. 45
Figure 4.1. SRAM cell *DRV* minimization by improving data-retention *SNM*. 47
Figure 4.2. Approaching theoretical *DRV* limit with design approaches (90nm node) 48
Figure 4.3. SRAM cell design variables 50
Figure 4.4. Modeled *DRV* sensitivities to SRAM cell design parameters 52
Figure 4.5. DRV design model verification (standard size array) 54
Figure 4.6. Worst case *DRV* optimizations 56
Figure 4.7. Leakage power minimization with body bias 58
Figure 4.8. *DRV*-aware optimization for leakage saving 59
Figure 4.9. Measured *DRV* distributions before and after design optimization 60
Figure 4.10. *DRV* optimization impacts on active operation parameters 62
Figure 5.1. Minimizing SRAM standby V_{DD} with error correction 66
Figure 5.2. Upper and lower bounds on SRAM standby power per useful bit [32]. 68
Figure 5.3. Error-tolerant SRAM chip design diagram. 70
Figure 5.4. Ultra-low leakage SRAM chip in a 90nm industry technology 70
Figure 5.5. Measured *DRV* distributions from the ultra-low leakage SRAM chip 72
Figure 5.6. Measured SRAM leakage power savings 73
Figure 5.7. Leakage power savings with error-tolerant SRAM design 74
Figure 5.8. SER improvement with a (31, 26, 3) Hamming ECC 75

List of tables

Table 2.1. *DRV* model verification. 22
Table 3.1. Normalized array sizing of the 90nm design optimization test chip 38
Table 3.2. Summary of *DRV* sensitivity on design parameters 44
Table 4.1. *DRV* design model 50
Table 4.2. Prediction errors of the *DRV* design model 53
Table 4.3. Summary of *DRV*-aware SRAM optimization for a 90nm technology 64
Table 5.1. Measured worst-case *DRV* range among 24 chips...71

1 Introduction

CMOS technology scaling over the past four decades has been enabling higher integration capacity in VLSI designs. As device density increases, a larger fraction of chip area is devoted to the on-chip memory modules, because on-chip memory helps improve the micro-architectural performance of a microprocessor. Several of the latest processor designs showed that around 50% of the chip area was occupied by caches [1, 2]. On the lower end of performance spectrum, a recent implementation of a wireless sensor network-protocol processor used a 64KB SRAM module that consumed 40% of total chip area [3]. As a result of the large on-chip memories and a 5X leakage increase every technology generation [4], the memory leakage power has been increasing dramatically and becomes one of the main challenges in future system-on-a-chip (SoC) design. For example, 30% of the Alpha 21264 power consumption and 60% of the StrongARM power consumption are dissipated in cache and memory structures [5]. For mobile applications low standby power is crucial. A mobile device often operates in the standby mode. As a result, the standby leakage power has a large impact on the device battery life.

Memory leakage suppression is important for both high speed and low power SoC designs. While the analysis and techniques proposed in this paper are applicable in general, the focus of this work is to develop an effective scheme for SRAM leakage suppression in battery-powered mobile applications.

1.1 Existing Work

A large variety of circuit design techniques have been proposed to reduce the leakage power of SRAM cells and the memory peripheral circuits (decoding circuitry, I/O, etc). Previous work showed that leakage of the peripheral circuits can be effectively suppressed by turning off the leakage paths with switched source impedance (SSI) during idle period [6]. Our work focuses on the leakage control of SRAM core cell. The existing SRAM cell leakage reduction techniques include novel SRAM cell design [7, 8], dynamic-biasing [9-11], and V_{DD}-gating [13-17]. The following sections provide a detailed review of the existing techniques, and compare these to the approach we propose.

1.1.1 The novel SRAM cells

As the supply voltage (V_{DD}) scales down in each new technology generation, in recent years several new SRAM cell designs were proposed with a reduced leakage power. A 10-T SRAM cell in CMOS technology improves the read margin by buffering the stored data during a read access, and enhances the write margin with a floating V_{DD} during write operation [7]. The improved operation margins allow this cell to operate at a V_{DD} lower than 400mV. Memory operations at such a low voltage effectively reduce both the active and standby power. In another work, a 4-T FinFET-based SRAM cell used back-gated feedback design to boost the static noise margin (*SNM*) and reduce cell leakage [8]. In contrast to these approaches, this work focuses on improving the conventional 6-T structure CMOS SRAM cell for ultra-low power standby operation.

1.1.2 Dynamic biasing techniques

The dynamic-biasing techniques use dynamic control on transistor gate-source and substrate-source bias to enhance the driving strength of active operations and create low leakage paths during standby period [18]. For example, the driving source-line (DSL) scheme connects the source line of the cross-coupled inverters in an SRAM cell to a negative voltage V_{BB} during read cycle, and leaves the source line floating during write cycle. This bias configuration improves the speed of both the SRAM cell read and write operations. Therefore, high threshold (V_{th}) transistors can be used to reduce leakage, without compromising the active performance [9]. Another similar technique is the negative word-line driving (NWD) scheme. NWD uses low V_{th} access transistors with negative cut-off gate voltage and high V_{th} cross-coupled inverter pair with boosted gate voltage. The result is an improved access time and a reduced standby leakage [10]. The dynamic leakage cut-off (DLC) scheme applies reverse-biased PMOS and NMOS substrate voltages on non-selected SRAM cells [11].

At the current technology nodes (130nm and 90nm), the above dynamic-biasing schemes typically achieve 5-7X leakage power reduction. This power saving becomes less as the technology scales, because the worsening short-channel effects cause the reverse body bias effect on leakage suppression to diminish [12]. In order to design for a higher (>30X) and sustainable leakage power reduction, an SRAM designer needs to integrate multiple low-power design techniques, rather than using dynamic-biasing only.

By combining multiple leakage suppression schemes (e.g. reduced supply voltage, sizing optimization and dynamic-biasing), the low-leakage SRAM design presented in this work achieves a 50X leakage reduction ratio.

1.1.3 V_{DD}-gating techniques

The V_{DD}-gating techniques either gate-off the supply voltage of idle memory sections, or put less frequently used sections into a low-voltage standby mode. There are three types of leakage mechanisms in an SRAM cell: sub-threshold leakage, gate leakage and junction leakage. A lower V_{DD} reduces all of these leakages effectively. The reduction ratio in leakage power is even higher because both the supply voltage and leakage current are reduced.

An example of V_{DD}-gating is the Cache Decay technique, which gates off unused cache sections, and uses cache activity analysis to balance the leakage energy saving against the performance loss caused by extra cache misses. With adaptive timing policies in cache line gating, Cache Decay achieves 70% leakage power reduction at a performance penalty of less than 1% [13]. To further reduce leakage power for caches with large utilization ratio, the Drowsy Caches approach was proposed to allocate inactive cache lines to a low-power mode, where a low standby V_{DD} is used to reduce leakage. The Drowsy Caches design assumes that the standby V_{DD} is higher than the voltage level required for SRAM data-retention. Therefore the cache data are preserved during the drowsy standby mode. A leakage power reduction higher than 70% is reported in a drowsy data cache [14].

In recent years as the need of leakage reduction in high-utilization memory structures increases, there have been many research activities on low-voltage SRAM standby techniques. Most of the reported circuit techniques in this field focus on the design of sleep control circuits. For example, an array of dynamically-controlled sleep transistors was used to provide a finely programmable standby V_{DD} [15]. In another design, a self-decay circuit generates a periodical sleep pulse with an adaptive pulse period, which puts the SRAM array into a sleep mode more frequently at high leakage conditions (fast process, high temperature) and vice versa. The result is an optimized tradeoff between leakage power reduction and dynamic power overhead [16]. A recent work proposed an actively clamped sleep transistor design, which is capable of adaptively adjusting the level of standby V_{DD} based on the magnitude of leakage current. With this design the cache standby power is minimized under all conditions during the lifetime of a processor [17].

Although the above techniques can be very effective in enhancing the efficiency of low-voltage memory standby operation, an important parameter needed by all of these schemes is the value of SRAM standby V_{DD}. This is because a high standby V_{DD} preserves memory data but produces high leakage current, and a very low standby V_{DD} effectively reduces leakage power but does not guarantee a reliable data-retention. An optimal standby V_{DD} is needed to maximize the leakage power saving and satisfy the data preservation requirement at the same time. All of the existing circuit design techniques

either assumed that the value of this optimal standby voltage will be determined empirically, or did not address this issue.

1.2 Contribution

To determine the optimal standby V_{DD} of an SRAM, it is important to understand the voltage requirement for SRAM data retention. Based on an in-depth study of SRAM low-voltage data-retention behavior, this work defines the boundary condition of SRAM data retention voltage (*DRV*), and then derives both the theoretical and practical limits of *DRV* as functions of design and technology parameters. These *DRV* analysis and results provide insights to SRAM designers and facilitate the development of low power memory standby schemes. In addition to the analytical *DRV* study, we also develop two novel design techniques that aggressively reduce SRAM standby leakage: *DRV*-aware SRAM cell optimization and power-optimized error-correction scheme.

This work is the first analytical investigation into the minimum voltage required for SRAM standby operation. The only similar existing effort was an Intel study, which empirically characterized VCC_{min}, the minimum V_{DD} required for SRAM active operation (read and write) [19]. From their experiments it was found that the VCC_{min} vary temporally by up to 250mV over a time period of seconds. This variation was attributed to temporal variations in the device gate-leakage [19]. Because our work focuses on studying the minimum voltage requirement of standby instead of active operation, there are two major differences between our *DRV* study and the Intel VCC_{min} research. First of

all, due to the more stable closed-loop standby operation, the SRAM cell *DRV* is typically in the sub-threshold range, which is much lower than the VCC_{min} (550~750mV) for a 90nm CMOS process. Secondly, since gate leakage decreases exponentially with the supply voltage [20], the impact of gate leakage on sub-threshold range *DRV* is very small, and does not cause an observable temporal variation for *DRV*. Besides these differences, in this work we not only characterize the *DRV* empirically with measurement data, but also take an analytical approach to model the *DRV* as a function of design and process parameters. These analytical *DRV* models provide insights to both circuit and architecture level low-voltage SRAM design optimizations.

Built on the *DRV* analysis results, our next contribution is an ultra-low leakage SRAM designed for low-power mobile applications. At circuit-level, a *DRV*-aware SRAM cell optimization methodology was developed to minimize the SRAM cell standby leakage. At architecture-level, we designed an error-correction scheme to aggressively reduce the memory standby V_{DD} to a level below the highest *DRV* among all SRAM cells. This scheme uses an error correction code (ECC) to correct the retention errors caused by low-voltage standby operation. This is the first time the ECC data redundancy is used for SRAM power minimization.

As a design-verification, a 26kb ultra-low leakage SRAM module was implemented in an industrial 90nm technology. This error-tolerant SRAM design uses a (31, 26) Hamming ECC. The measurement results show a 330mV reduction in the worst-case *DRV* value and a 98% SRAM leakage power saving. The active operation noise margins

and read access speed are improved with dynamic bias control. The only penalty is a 50% area overhead.

1.3 Thesis Organization

The rest of this thesis is organized as following. Chapter 2 develops the analytical models of *DRV* under both ideal and practical conditions. To verify the *DRV* models, Chapter 3 presents the *DRV* data measured from two SRAM test chips implemented in 130nm and 90nm industrial technologies. Based on the *DRV* sensitivities data from the 90nm test chip, *DRV*-aware SRAM optimization methods in Chapter 4 minimize the SRAM cell leakage during low-voltage standby operation. Chapter 5 introduces an error-tolerant SRAM design that aggressive reduce the standby V_{DD} by correcting the low-voltage data-retention errors with an ECC. Chapter 6 concludes this work.

2 SRAM Data Retention Voltage (*DRV*) Analysis

The circuit structure of a 6T SRAM cell is shown in Figure 2.1(a). In a typical SRAM design, the bitline voltages are connected to V_{DD} during standby mode. To facilitate the *DRV* analysis, this cell can be represented by a flip-flop comprised of two inverters as shown in Figure 2.1(b) [21]. These inverters include access transistors M_5 and M_6. When V_{DD} is reduced to *DRV* during standby operation, all six transistors in the SRAM cell are in the sub-threshold region. Thus, the capability of SRAM data retention strongly depends on the sub-threshold current conduction behavior. In order to understand the low-voltage data preservation behavior of SRAM and the potential for leakage saving through minimizing the standby V_{DD}, analytical models of SRAM *DRV* under ideal and realistic conditions are developed in this section.

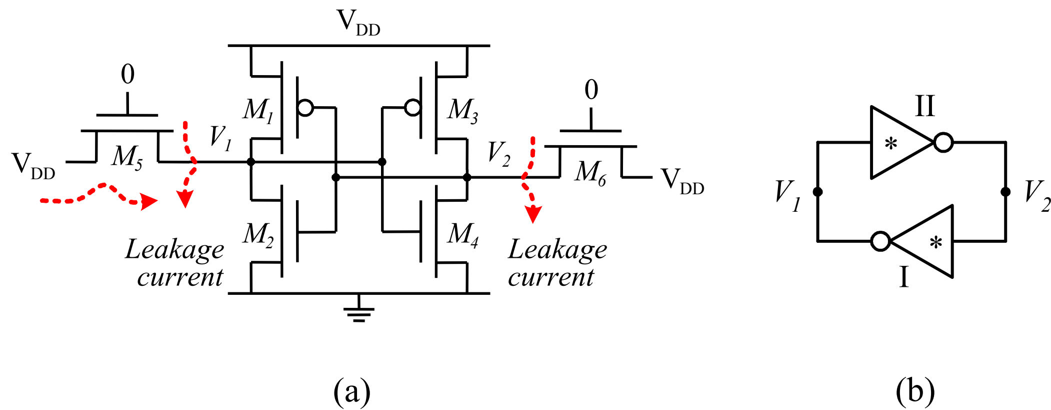


Figure 2.1. Standard 6T SRAM cell structure. (a) 6T SRAM cell in standby (assuming $V_1 \approx 0$ and $V_2 \approx V_{DD}$.). (b) Flip-flop representation of the same SRAM cell.

2.1 *DRV* Definition

As the minimum V_{DD} required for data preservation, *DRV* of an SRAM cell is a measure of its state-retention capability under very low voltage. In order to reliably preserve data in an SRAM cell, the cross-coupled inverters shown in Figure 2.1(b) must have a loop gain greater than one. The stability of an SRAM cell is also indicated by the static-noise margin (*SNM*) [21]. As shown in Figure 2.2, the *SNM* can be graphically represented as the largest square between the voltage transfer characteristic (VTC) curves of the internal inverters from Figure 2.1(b). When V_{DD} scales down to *DRV*, the VTC of the cross-coupled inverters degrade to such a level that the loop gain reduces to one and *SNM* of the SRAM cell falls to zero, as illustrated in Figure 2.2. Using the notations from Figure 2.1, this condition is given by:

$$\left.\frac{\partial V_1}{\partial V_2}\right|_{\text{inverter I}} \cdot \left.\frac{\partial V_2}{\partial V_1}\right|_{\text{inverter II}} = 1, \text{ when } \mathrm{V_{DD}} = \mathrm{DRV} \qquad (2.1)$$

If V_{DD} is reduced below the *DRV*, the inverter loop switches to the other biased state determined by the deteriorated inverter VTC curves, and loses the capability to hold the stored data. Note that the VTC shown in Figure 2.2 assumes the worst-case local mismatches among the SRAM cell transistors. That is also the condition for the worst-case *DRV* because the *SNM* deterioration increases with mismatches.

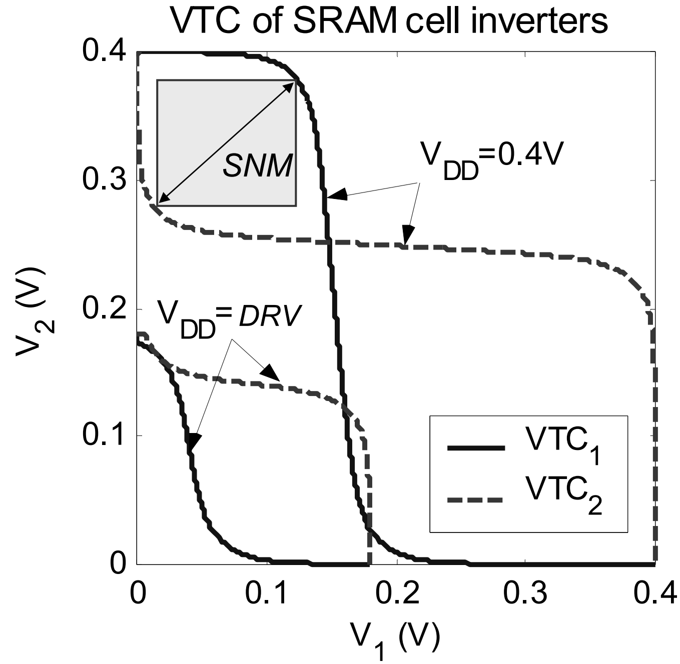


Figure 2.2. An illustration of SRAM inverter VTC deterioration under low-V_{DD}. The SRAM cell noise margin is zero at *DRV*. This VTC simulation assumes 3σ worst-case local mismatches among the SRAM cell transistors.

2.2 *DRV* Theoretical Lower Bound

Before investigating *DRV* in a realistic design environment, it is important to understand the fundamental limit of *DRV* in theory, assuming ideal process and design conditions. Such an understanding provides guidance to the optimization of design and technology in the long term.

Based on Eq. 2.1, the *DRV* of a SRAM cell can be determined by solving the sub-threshold VTC equations of the two internal data-holding inverters, since all the transistors conduct in weak inversion region when V_{DD} is around *DRV*. The derivation is presented below.

When an SRAM cell (Figure 2.1) is in standby mode, the currents in each internal inverter are balanced:

$$\text{Node V}_1: \quad I_1 + I_5 = I_2, \tag{2.2}$$

$$\text{Node V}_2: \quad I_3 + I_6 = I_4. \tag{2.3}$$

We may assume that the original state stored in SRAM cell is:

$$V_1 \approx 0 \text{ and } V_2 \approx V_{DD}. \tag{2.4}$$

In order to minimize mismatches and maximize the data-retention noise margin, in a theoretical *DRV* limit analysis we assume that the SRAM cell is manufactured with ideal process conditions, i.e., NMOS and PMOS have symmetrical V_{th} and sub-threshold slope factor, and there are no process variations. Furthermore, the leakage of access transistors (I_5 and I_6) is assumed to be totally eliminated by aggressive design optimization, e.g., reversed body bias on M_5 and M_6 during standby mode to increase V_{th}. Then, Eq. 2.2 and Eq. 2.3 simplifies to:

$$I_1 = I_2, \quad I_3 = I_4 \tag{2.5}$$

I_i is the sub-threshold current of the ith transistor (Figure 2.1). Assuming room-temperature standby operation, I_i can be considered as dominated by the drain-source leakage. This is because at sub-threshold V_{DD} and current technology (130nm and 90nm nodes) the gate leakage and other leakage mechanisms have minor contribution compared to the sub-threshold current. I_i is modeled as in [22]:

$$I_i = \beta_i I_0 \exp\left(\frac{-V_{th,i}}{n_i v_T}\right)\cdot \exp\left(\frac{V_{gs,i}}{n_i v_T}\right)\cdot\left(1-\exp\left(\frac{-V_{ds,i}}{v_T}\right)\right), \qquad (2.6)$$

where $v_T = kT/q$ is the thermal voltage, equal to 26mV when $T = 27°C$; β_i is the transistor (W/L) ratio; I_0 is the leakage current of a unit sized device at $V_{gs} = 0$ and $V_{ds} >> v_T$; T is the chip temperature; and n_i is the sub-threshold factor, (sub-threshold swing divided by 60mV at room temperature). If we further define:

$$I_{off,i} = \beta_i I_0 \exp\left(\frac{-V_{th,i}}{n_i v_T}\right), \qquad (2.7)$$

I_i can be expressed as:

$$I_i = I_{off,i}\cdot \exp\left(\frac{V_{gs,i}}{n_i v_T}\right)\cdot\left(1-\exp\left(\frac{-V_{ds,i}}{v_T}\right)\right). \qquad (2.8)$$

The V_{thi} in Eq. 2.7 can be accurately modeled as following, with the second and third terms representing the body bias effect and the drain-induced-barrier-lowering (DIBL) effect [23].

$$V_{th,i} = V_{thi,0} + \gamma_i\left(\sqrt{|-2\phi_i + V_{sb,i}|} - \sqrt{|-2\phi_i|}\right) - V_{ds,i}\cdot \exp(-\alpha l_i) \qquad (2.9)$$

Since all the SRAM cell transistors conduct in weak inversion region when V_{DD} is around DRV, the DIBL effect can be ignored in a DRV analysis.

Substituting these current models, which are functions of V_1, V_2, V_{DD}, T, and other technology parameters, Eq. 2.5 can be expanded into:

$$\exp\left(\frac{V_{DD}-V_2}{nv_T}\right)\cdot\left[1-\exp\left(-\frac{V_{DD}-V_1}{v_T}\right)\right]=\exp\left(\frac{V_2}{nv_T}\right)\cdot\left[1-\exp\left(-\frac{V_1}{v_T}\right)\right] \qquad (2.10)$$

$$\exp\left(\frac{V_{DD}-V_1}{nv_T}\right)\cdot\left[1-\exp\left(-\frac{V_{DD}-V_2}{v_T}\right)\right]=\exp\left(\frac{V_1}{nv_T}\right)\cdot\left[1-\exp\left(-\frac{V_2}{v_T}\right)\right], \qquad (2.11)$$

In Eq. 2.10 and Eq. 2.11, $I_{off,N} = I_{off,P}$ and $n_N = n_P$ are assumed based on the symmetry requirement to maximize the data-retention noise margin. The $I_{off,N} = I_{off,P}$ condition represents a balanced PMOS-to-NMOS (*P/N*) leakage strength ratio.

Then, by solving (V_1/ V_2) from Eq. 2.1 respectively and using the condition of Eq. 2.4, the theoretical limit of *DRV* is solved as:

$$DRV_{ideal} = 2v_T \ln(1+n) \qquad (2.12)$$

When $V_{DD} = DRV_{ideal}$, $V_1 = V_2 = DRV_{ideal}/2$. As a result, the SRAM cell loses the capability to differentiate store data. Note that by defining the condition of the loop gain equals to 1, Eq. 2.12 also holds for the minimum operation voltage of a single inverter.

For an ideal CMOS technology $n = 1$ (i.e., 60mV/dec as the swing), which provides DRV_{ideal} = 36mV. For a typical 90nm technology with $n = 1.5$, *DRV* goes up to 50mV. These results were confirmed with SPICE simulation result from an industrial 90nm technology.

Eq. 2.12 provides the theoretical bottom-line of *DRV* for CMOS-based SRAM design, no matter how well we can optimize the size or V_{th} of transistors. For a future transistor technology, if the sub-threshold swing could be reduced to 0 (i.e., $n = 0$), *DRV* could decrease to 0V. In a realistic CMOS technology, when the process or design parameters

deviate from the ideal condition, *DRV* increases to a value larger than that of Eq. 2.12. The *DRV* under a practical condition is discussed in the following section.

2.3 *DRV* Model with Process Variations

While the theoretical analysis showed <50mV values of *DRV* limit under ideal situation, in reality many imperfect conditions contribute to increases in *DRV*. In this section the *DRV* of a single SRAM cell is analyzed as a function of realistic design and process parameters.

2.3.1 DRV of a realistic SRAM cell

Based on Eq. 2.2, 2.3, and 2.4, the *DRV* of a realistic SRAM cell can be derived. In a typical standby condition, the leakages through the access transistor M_6 is negligible, because both the bitline voltage and the storage node voltage V_2 are at the same level of V_{DD}, creating a drain-to-source voltage of zero. Therefore Eq. 2.3 can be simplified to:

$$\text{Node V}_2: \quad I_3 = I_4. \tag{2.13}$$

Using the sub-threshold current model (Eq. 2.8), Eq. 2.2, 2.13 and 2.1 can be solved to derive the *DRV* and the corresponding V_1 and V_2. Due to the complexity of exponential functions, a general solution to the current and VTC equations requires numerical iterations. To avoid the iterations, we first estimate the initial value of *DRV*, i.e. $DRV^{(0)}$, using the approximations of Eq. 2.4:

$$DRV^{(0)} = \frac{kT/q}{n_2^{-1} + n_3^{-1}} \cdot \log\left[\left(n_3^{-1} + n_4^{-1}\right)\frac{I_{off,4}}{I_{off,2} I_{off,3}}\left(\frac{I_{off,5}}{n_2} + \frac{I_{off,1}}{\left(n_1^{-1} + n_2^{-1}\right)^{-1}}\right)\right] \tag{2.14}$$

Then, using $DRV^{(0)}$, a first approximation of V_1 and V_2 can be obtained:

$$V_1 = \frac{kT}{q} \cdot \frac{I_{off,1} + I_{off,5}}{I_{off,2}} \cdot \exp\left(\frac{-DRV^{(0)}}{n_2 \, kT/q} \right), \qquad (2.15)$$

$$V_2 = DRV^{(0)} - \frac{kT}{q} \cdot \frac{I_{off,4}}{I_{off,3}} \cdot \exp\left(\frac{-DRV^{(0)}}{n_3 \, kT/q} \right). \qquad (2.16)$$

With Eq. 2.15 and Eq. 2.16 available, we can refine the calculation of *DRV* and a final expression is obtained:

$$DRV = DRV^{(0)} + \left[\frac{V_1}{2} + \frac{\left(DRV^{(0)} - V_2 \right) \cdot n_2}{2} \right]. \qquad (2.17)$$

The above *DRV* formula only relies on the values of $I_{off,i}$ and n_i, which can be easily extracted from transistor characterizations, either by simulation or measurement. For a 130nm industrial technology we studied, $n = 1.25$ for both PMOS and NMOS.

2.3.2 DRV sensitivity to variations

As shown in the above equations, the *DRV* is a function of various process and design parameters. Later in Chapter 4 we will discuss about the impact of SRAM cell design parameters on *DRV* and design optimization methods that minimize *DRV*. Following sections discuss the other variation factors that impact the *DRV*, including both the global (systematic) and local (random) variations.

Process variation and temperature fluctuation are the main factors that cause degradations in circuit performance. For an SRAM cell, any local mismatch between the

two internal data-holding inverters has a strong impact on its *DRV*. As an example, Figure 2.3 shows the simulated SRAM inverter VTC under a 200mV V_{DD}, based on a standard industrial SRAM cell design. The solid lines show the ideal VTC without process variation, while the dashed lines are VTC with 3σ local variations in *L* and V_{th}. A major decrease in *SNM* is a clear result of the worst-case mismatch among transistors, as indicated by the small opening between VTC curves with variations. The impact of global variation on *SNM* is much smaller because both the VTC curves shift in the same direction.

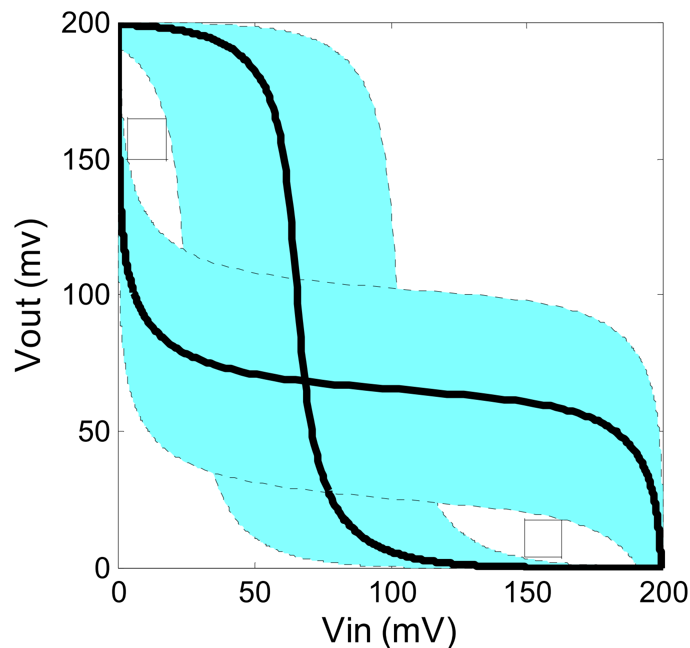


Figure 2.3. VTC of SRAM cell inverters under 3σ variation in *L* and V_{th}. (Solid lines: ideal case with no variation.)

Since the local process variation (mismatch) is the key factor that causes *SNM* degradation and DRV increase, we analyze the general process variation by first

identifying the global and local variation components, and then evaluating the impact of each variation component on DRV separately. Using V_{th} as an example, the local and global V_{th} variations in the PMOS pull-up transistors (M1 and M3) are defined as:

$$\Delta V_{th1,global} = \Delta V_{th3,global} = \frac{V_{th1} + V_{th3}}{2} - V_{thP} \qquad (2.18)$$

$$\Delta V_{th1,local} = \Delta V_{th3,local} = \left| \frac{V_{th1} - V_{th3}}{2} \right|,$$

where V_{thP} is the designed threshold value for these PMOS transistors. Note that the local variation value is always positive. This is because for two SRAM cell transistors in symmetrical positions, a local mismatch in either polarity between these two transistors causes the same increase in DRV.

Figure 2.4 plots the change in *DRV* value versus the magnitude of local and global variations in an SRAM cell (obtained from SPICE simulations). For each DRV data point, the same magnitude of process variation (in the unit of σ) is assumed for all six transistors. As the plot shows, the local variations result in substantial *DRV* increases. Based on a 130nm technology model, a 3σ local mismatch in V_{th} causes a 70mV increase in *DRV*, compared to the ideal case with perfect matching. At the same time, a global variation in V_{th} or *L* has a much weaker impact on *DRV*. This is because a global variation affects both inverters (Figure 2.1(b)) in the same direction and does not cause significant *SNM* degradation. This result was also indicated in Eq. 2.14, where the transistor local

variation (e.g. an I_{off} mismatch between M_2 and M_4) causes a substantial increase in *DRV* value and results in a reduced *SNM* (Figure 2.3).

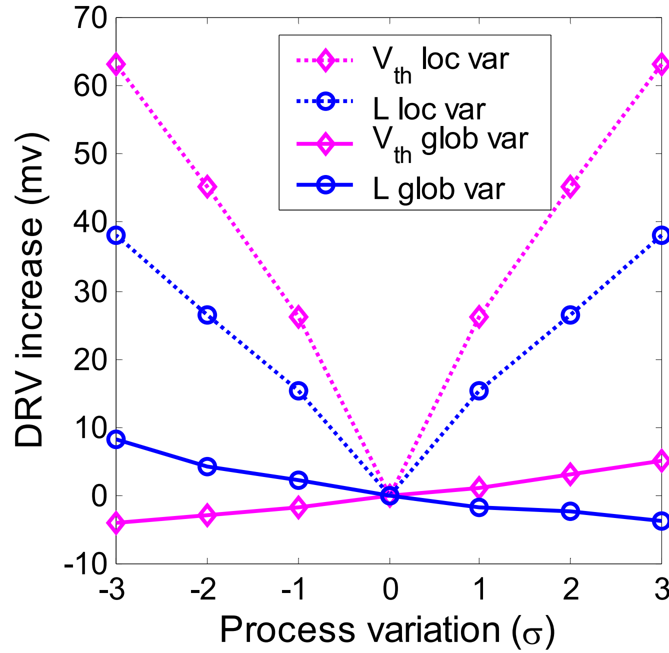


Figure 2.4. *DRV* sensitivity to local and global variations.

The temperature fluctuation is another global variable that has a fairly weak influence on *DRV* since it affects all the transistors in an SRAM cell uniformly. Simulation results in Figure 2.5 compare the impact of process and temperature variation on *DRV*. The *DRV* increases about 100mV in the presence of 3σ local mismatch in V_{th} and L, while the temperature impact is much smaller. When T changes from 27°C to 100°C, the *DRV* is only about 13mV higher.

Due to the small impact of global variation on the *DRV*, the rest of this work focus on analyzing the local mismatch in an SRAM cell. All the worst-case *DRV* analysis assume

a 3σ local mismatch around the typical parameters. (In a complete analysis, the worst-case *DRV* needs to be analyzed at each process corner that represents different global variation.)

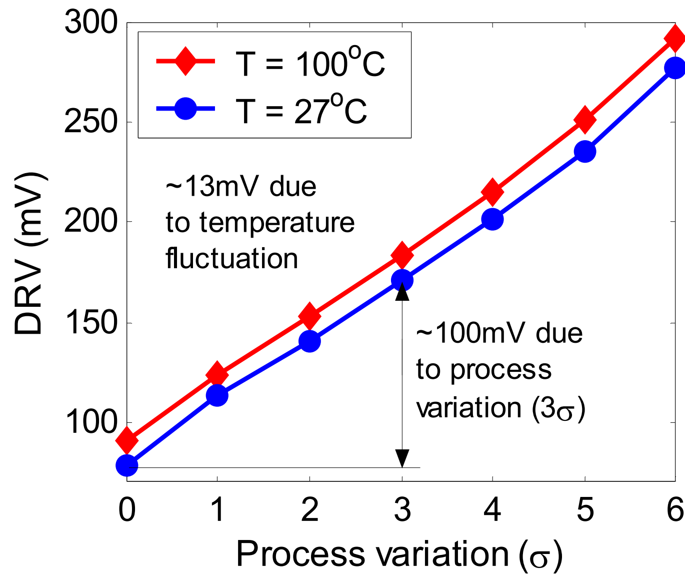


Figure 2.5. SRAM cell *DRV* under process and temperature variations.

2.3.3 DRV model with variations

The analytical *DRV*-modeling in Eqs. 2.14-2.17 is based on the leakage current of each individual SRAM cell transistor. Those equations capture the *DRV* sensitivities on process parameters (I_{off} and n), sizing β_i, and chip temperature (T). By generalizing the model over process variation factors, the *DRV* sensitivities on variations can be extracted from Eq. 2.17 with a first-order analysis:

$$
\begin{aligned}
DRV &= DRV_{matched} + \Delta DRV \\
&= DRV_{matched} + \sum_i a_i \frac{\Delta\beta_{i,local}}{\beta} + \sum_i b_i \Delta V_{thi,local} + c\Delta T
\end{aligned} \tag{2.19}
$$

where $DRV_{matched}$ is the data-retention voltage of a perfectly matched SRAM cell (i.e., no variations or with only global variation on all transistors) at room temperature; a_i, b_i, and c are fitting coefficients for each individual transistor. The $\Delta\beta_{i,local}$ and $\Delta V_{thi,local}$ terms in this model represent the local variation on individual transistors. ΔT is the overall chip temperature fluctuation. Since there is usually a small change in the $DRV_{matched}$ value caused by global process variation, this model focuses on capturing the impact of local process variation on the *DRV*. Considering an industrial SRAM standard cell design in a 130nm technology as an example, the model coefficients a_i's are extracted from SPICE simulations. Temperature coefficient c is extracted as 0.169mV/°C, which predicts an increase of 12.3mV in *DRV* when T rises from 27°C to 100°C.

The *DRV* predictions by Eq. 2.19 match well with SPICE simulations over a wide range of design parameters and their variations. This is illustrated in Table 2.1, which summarizes the simulation and modeling results. These results assume a 3σ worst-case local mismatch in V_{th} and L for all six transistors in the SRAM cell.

***DRV* Conditions**	**SPICE**	**Model**
Ideal (without variations)	77 mV	78 mV
With 3σ variation in V_{th} & L	170 mV	169 mV
200% PMOS sizing with 3σ V_{thv}& L variation	136 mV	138 mV
200% NMOS sizing with 3σ V_{th} & L variation	182 mV	180 mV
T at 100°C with 3σ V_{th} & L variation	183 mV	182 mV

Table 2.1. *DRV* model verification.

2.4 Low-Voltage SRAM Standby Stability Analysis

In order to reliably preserve data in an SRAM cell at a low-voltage standby mode, an adequate *SNM* is necessary. Usually a positive *SNM* is created by setting the SRAM standby V_{DD} at a level higher than the *DRV*. The difference between the standby V_{DD} and the *DRV* is called the standby guard band voltage. This section quantitatively evaluates SRAM cell *SNM* as a function of the standby guard band voltage.

The *SNM* of an SRAM cell can be calculated in many different ways: the maximum square between the normal and mirrored VTC, small-signal loop-gain, Jacobian of the Kirchoff equations, coinciding roots [24]. These methods are well researched and it has been shown that they are all equivalent [24]. Similar to [21], we take the loop-gain approach of analyzing the *SNM* as the maximum value of noise that can be tolerated by the flip-flop before changing states. As shown in Figure 2.6, two noise sources, V_n, are inserted to assure the worst-case noise scenario when the noise is present in both gates in the same way [21].

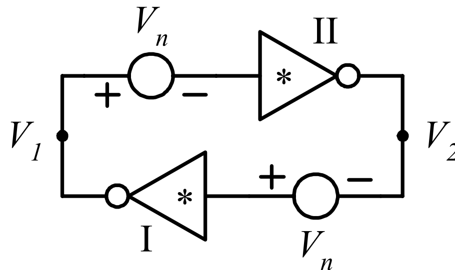


Figure 2.6. Flip-flop representation of SRAM cell with inserted static noise, V_n.

Following the methodology of *DRV* derivation in Section 2.1 but this time with inserted static noise V_n,

$$V_{\mathrm{GS2}} + V_{\mathrm{n}} = V_2 \tag{2.20}$$

$$V_{\mathrm{GS4}} - V_{\mathrm{n}} = V_1, \tag{2.21}$$

we obtain a zero-order approximation for *SNM* from the condition of marginal stability, that is the unity loop-gain. The maximum noise corresponding to the unity gain is given by:

$$\begin{aligned} SNM = \frac{2}{3}V_{DD} - \frac{n \cdot kT/q}{3} \cdot \ln\left(\frac{2I_{off,4}}{n \cdot I_{off,2} \cdot I_{off,3}}\right) \\ - \frac{n \cdot kT/q}{3} \cdot \ln\left(\frac{I_{off,5}}{n} + \frac{2I_{off,1}}{n} e^{\frac{SNM}{n \cdot kT/q}}\right) \end{aligned}, \tag{2.22}$$

where we assume a simplification of equal sub-threshold slope factor for NMOS and PMOS transistors, and adopt the approximations from Eq. 2.4 as well. The above formula does not have a closed-form solution, but can be solved iteratively. For each value of V_{DD}, the *SNM* value after five iterations is shown in Figure 2.7. This zero-order model closely predicts the slope of the SNM-V_{DD} line, compared to simulation data for cases under 3σ local mismatch and ideal case without variation.

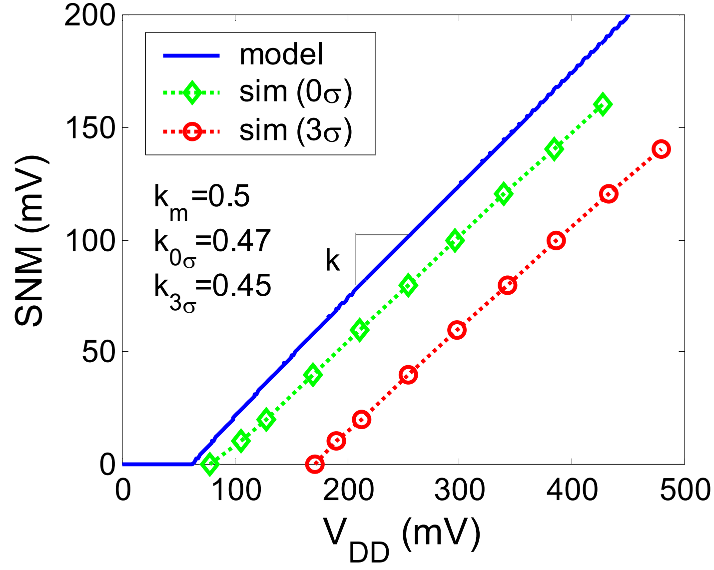


Figure 2.7. Static noise margin (*SNM*) as a function of V_{DD}. Slope of a first-order linear model agrees with simulation results.

Furthermore, from the linear relationship indicated in Figure 2.7 we can adopt simple linear macro-model given by:

$$SNM = k \cdot \left(V_{dd} - DRV\right), \qquad (2.23)$$

Further expansion of Eq. 2.22 and comparison with Eq. 2.23 yield the following approximation of the *k* factor:

$$k \approx \frac{2}{3+n}, \qquad (2.24)$$

where $I_{off,5}$ from Eq. 2.22 is neglected due to exponential nature of the other term under the logarithm. This approximation is valid for $SNM > nkT/q$. With $n = 1.25$, we obtain $k = 0.47$, which exactly matches the simulation data shown in Figure 2.7. The result in Eq. 2.24 means that a smaller sub-threshold factor is desirable for higher noise tolerance in standby mode.

This linear correlation of *SNM* and the standby guard band voltage facilitates the SRAM design for reliable data retention under low voltage. For example, in order to achieve a 50mV *SNM* under 3σ local process variations, the SRAM standby V_{DD} needs to be 100mV higher than the corresponding *DRV*.

2.5 SRAM Standby Leakage Modeling

Assuming that $V_1 \approx 0$ and $V_2 \approx V_{DD}$, the total leakage of an SRAM cell in the subthreshod standby mode can be calculated as:

$$\begin{aligned} I_{leak} &= I_1 + I_4 + I_5 \\ &\approx \left(\beta_1 I_0 \exp\left(\frac{-V_{th,1}}{n_1 v_T}\right) + \beta_4 I_0 \exp\left(\frac{-V_{th,4}}{n_4 v_T}\right) + \beta_5 I_0 \exp\left(\frac{-V_{th,5}}{n_5 v_T}\right)\right) \cdot \left(1 - \exp\left(\frac{-V_{DD}}{v_T}\right)\right) \\ &= \left(\beta_P I_0 \exp\left(\frac{-V_{th,P}}{n_P v_T}\right) + \beta_N I_0 \exp\left(\frac{-V_{th,N}}{n_N v_T}\right) + \beta_A I_0 \exp\left(\frac{-V_{th,A}}{n_N v_T}\right)\right) \cdot \left(1 - \exp\left(\frac{-V_{DD}}{v_T}\right)\right) \end{aligned} \quad , (2.25)$$

where β_P, β_N and β_A represent the sizing ratio of the pull-up PMOS transistor, pull-down NMOS transistor and the access transistor respectively.

Since I_{leak} is a function of V_{DD}, the leakage power P_{leak} can be calculated after the standby V_{DD} (V_{DD_STBY}) is determined:

$$P_{leak} = V_{DD_STBY} \cdot I_{leak}(V_{DD_STBY}) \quad (2.26)$$

In order to reliably preserve the memory data during the low-voltage standby mode, the minimum standby V_{DD} ($V_{DD_STBY_MIN}$) is

$$V_{DD_STBY_MIN} = DRV_{MAX} + V_{gb}, \quad (2.27)$$

where V_{gb} stands for the guard-band voltage, and DRV_{MAX} represents the highest DRV among all cells in an SRAM module.

3 Measured SRAM *DRV* and its Evolution into the Future

To obtain silicon verification of the *DRV* models derived in Chapter 2, two SRAM test chips were designed and implemented in 130nm and 90nm industrial technologies. The *DRV* measurement data from these test chips are presented in this chapter. SRAM leakage power savings by reducing the standby V_{DD} to the minimum data-retention level are discussed.

3.1 *DRV* Measured in 130nm Technology

To attain the first *DRV* measurement result and explore the potential of SRAM leakage suppression with ultra low standby V_{DD}, a 32K-bit SRAM test chip with dual rail standby control was implemented in an industrial 130nm technology. Designed for ultra low-power applications, this scheme puts the entire SRAM into a deep sleep mode during the system standby period. As shown in Figure 3.1, the SRAM supply rails are connected to the standard V_{DD} and the standby V_{DD} through two power switches. The test chip consists of a 32K-bit industrial IP SRAM module and a custom on-chip switch-capacitor (SC) converter that generates the standby V_{DD} with 85% conversion efficiency.

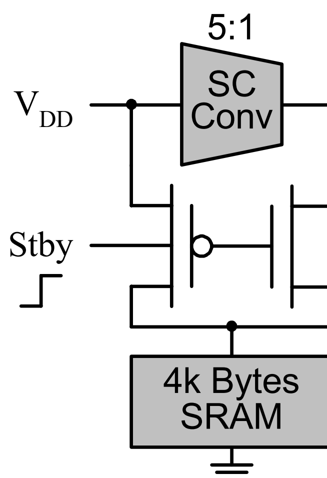


Figure 3.1. SRAM low-voltage standby leakage suppression scheme.

3.1.1 Dual supply design considerations

When designing for an ultra low standby V_{DD}, reliability of the SRAM data retention is a top concern. The major factors that may disturb the memory state are noises on the standby supply rail and radiation particles. During a low voltage standby mode, the power supply noise is mostly caused by the output voltage ripple of the SC converter. Therefore, an appropriate noise margin is needed in order to achieve the desired reliability. As analyzed in section 2.3, assigning a noise margin of 100mV results in about 50mV *SNM* in an SRAM cell. Since simulation shows that the peak-to-peak ripple on SC converter output is 20mV, a noise margin of 100mV provides a worst case *SNM* of 45mV, which is typical sufficient for SRAM cell state preservation.

Radiation particle poses another threat to reliable data storage. For a 130nm technology SRAM cell with about 1fF parasitic capacitance at the data storage node, the

critical charge ($Q_{critical}$) for a 1V V_{DD} is simulated to be approximately 3 fC. This is the minimum amount of storage-node charge injection that disrupts the state preserved in this cell. For a reduced V_{DD} at 100 mV above the DRV, $Q_{critical}$ is reduced to 0.5 fC. Considering the soft error hazard, a larger guard-band voltage or additional storage capacitances [25] may be needed. Use an error-correction scheme also helps reduce the soft error rate. The SRAM design with error correction is discussed in Chapter 5.

For a dual supply scheme, other design considerations include the active performance degradation due to the power switch resistance, memory wake up delay and the power penalty during operation mode transition. Designed for ultra low-power applications, the system requirements of this work are much more stringent on power than performance. In this context, the concern on delay overhead is not crucial. A 200μm wide PMOS power switch with 30Ω conducting resistance is used to connect the memory module to a 1V V_{DD} during active operations. With the same switch the memory wake up time is simulated to be within 10ns, which is typically a small fraction of the system cycle time in battery-operated applications [26].

The wake up power penalty incurred when switching from standby to the active mode determines the minimum standby time for this scheme if net standby power saving is to be achieved. This break-even time is an important system-design parameter, as it helps the power control algorithm to decide when a power-down would be beneficial. With the parasitic capacitance data attained from the technology process model, the minimum standby time in this dual-rail design is estimated to be several tens of microseconds, which is much shorter than the typical system idle time in a battery-supported system.

3.1.2 Test chip implementation

The 130nm SRAM test chip is shown in Figure 3.2. The two main components on this chip are a 32K-bit SRAM module and a SC converter. This memory is an industrial IP module with no modifications from its original design. As shown in Figure 3.3, a representative five-stage step down dc-dc SC converter topology is selected to implement the on-chip standby V_{DD} generator [27]. Compared to magnetic-based voltage regulators, an SC converter provides higher efficiency, smaller output current ripple, and easier on-chip integration for small loads in the microwatt range. The design challenge here resides in handling small output loads in the range of 10~20μW with high power efficiency. With such a small output load, the power loss incurred by short-circuit currents during phase switching becomes comparable to the output power and forms a significant portion of the total power loss.

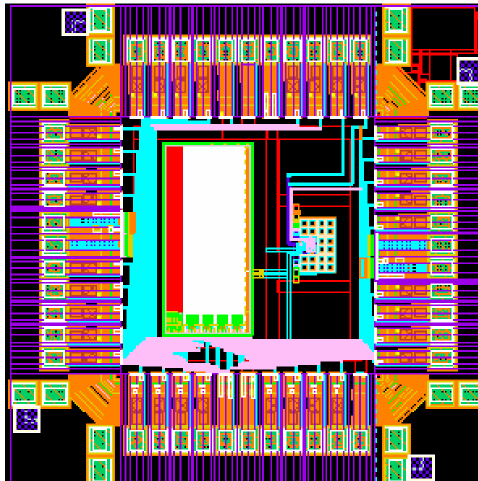
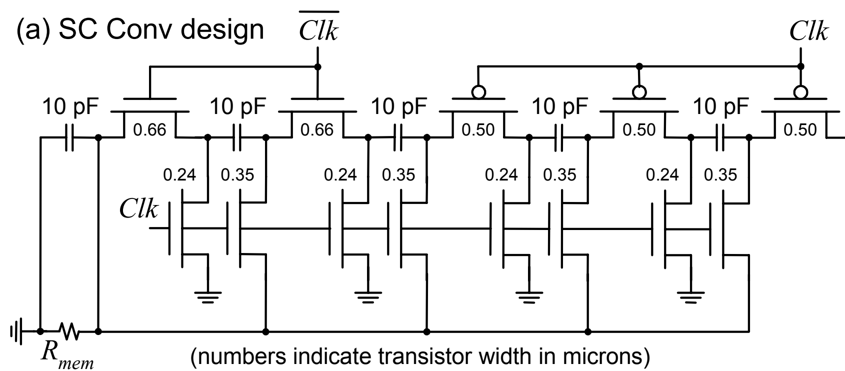
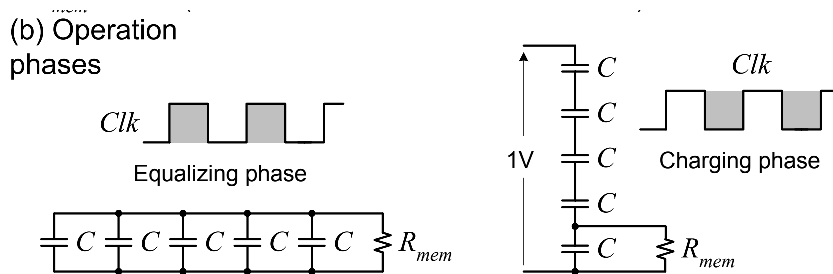


Figure 3.2. A 130nm SRAM leakage-control test chip.

To maximize the power efficiency, it is desirable to minimize both the switching voltage drop and short-circuit current, which have opposite dependence on device sizes. Hence the switching devices need to be carefully designed to balance these two requirements. For example, the NMOS / PMOS switch-type selection should maximize the device gate-source overdrive voltage at conducting mode, and minimize this overdrive voltage when the switch is turned off. With a careful design, Figure 3.3 shows the schematic of an optimized SC converter, with an 85% conversion efficiency at a 1V input and a 20μW output load.



(a) Schematic of the SC converter design



(b) Operation phases

Figure 3.3. An SC converter optimized for 20μW output load

3.1.3 Measurement results

The *DRV* is measured by monitoring the data retention capability of an SRAM cell with different values of standby V_{DD}, as demonstrated in Figure 3.4. With V_{DD} switching between active and standby modes, a specific state is written into the SRAM cell under test at the end of each active period (t_2), and then read out at the beginning of the next active period (t_1). Preservation of the assigned logic state is observed when standby V_{DD} is higher than *DRV* (top traces), while the state is lost when standby V_{DD} is below *DRV* (bottom traces).

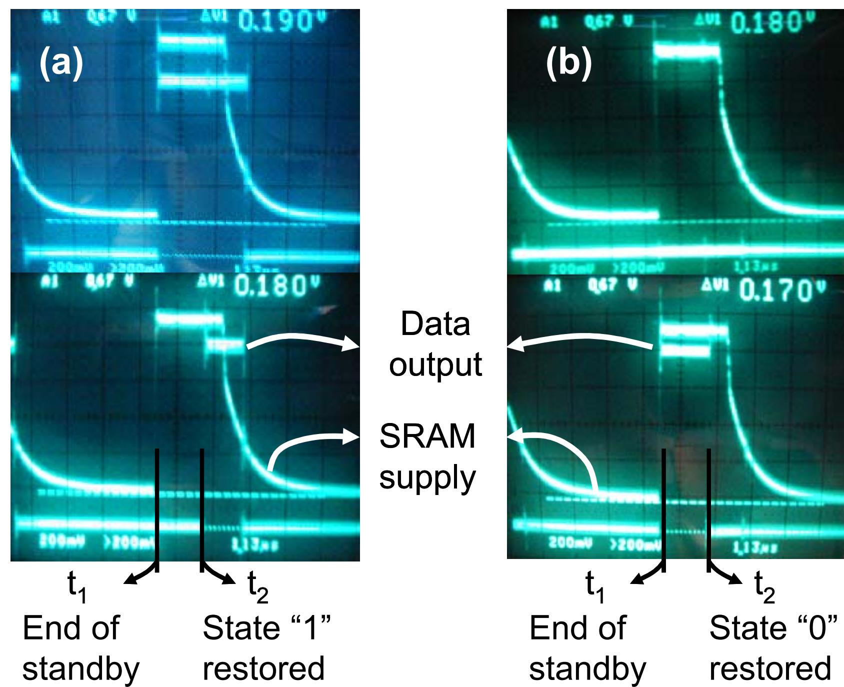


Figure 3.4. Waveform of *DRV* measurement. (a) *DRV* = 190mV in SRAM cell 1 with state "1", (b) *DRV* = 180mV in SRAM cell 2 with state "0".

Using automated measurement with a logic analyzer, the *DRV* of all 32K SRAM cells in one test chip was measured. For each SRAM cell, we measure the *DRV* by constantly

reading a written state out of the cell after a period of time in a low voltage standby mode. The read and write operations are conducted at 1V V_{DD}, while the standby V_{DD} keeps reducing until the read-after-standby cell state becomes the opposite to the written-before-standby state. Due to the unsymmetrical inverter strengths, every SRAM cell with process variations has a predominant state, and always returns to this state when V_{DD} is lower than the *DRV*, no matter what the original cell state is. During *DRV* testing each cell is measured twice with pre-written states '1' and '0' respectively. The measurement that writes the cell predominant state always read out the same value even with zero standby V_{DD}, while the other measurement provides the *DRV* of this memory cell when the non-predominant state flips to the predominant state at low V_{DD}.

Figure 3.5 shows the distribution of the 32Kb measurement results. The *DRV* values range from 60mV to 390mV with the mean value around 122mV. Such a wide range of *DRV* variation reflects the existence of considerable process variations during fabrication. As a result, the long *DRV* tail at the higher end reduces the leakage reduction achievable by minimizing the SRAM standby V_{DD}. To lower the SRAM standby V_{DD} and to improve the leakage power saving, other design techniques need to be employed, including circuit optimization and error-correction scheme.

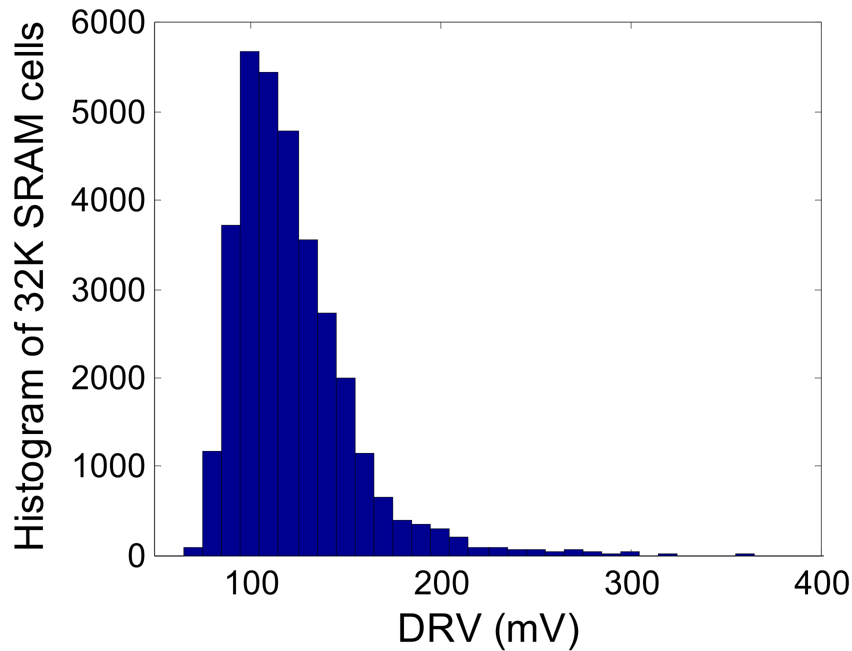


Figure 3.5. Measured *DRV* distribution of a 32K-bit SRAM chip.

The temperature dependency of *DRV* was also investigated experimentally. When the test chip was heated up to 100°C, a 10mV increase in *DRV* is observed. As evaluated in Section 2.3, our analytical *DRV* model not only predicts the ideal *DRV* values, but also fully captures the impact of process and temperature variations. Thus, it can serve as a convenient base for further design optimizations.

Furthermore, Figure 3.6 shows the spatial distribution plot of the *DRV* on the measured SRAM chip. From the plot, it can be observed that the on-chip *DRV* distribution is a combination of random within-die mismatches and systematic deviations on the boundaries of SRAM sub-array blocks. The pattern of the SRAM *DRV* spatial

distribution can be used when designing an error tolerant scheme for aggressive SRAM standby voltage reduction.

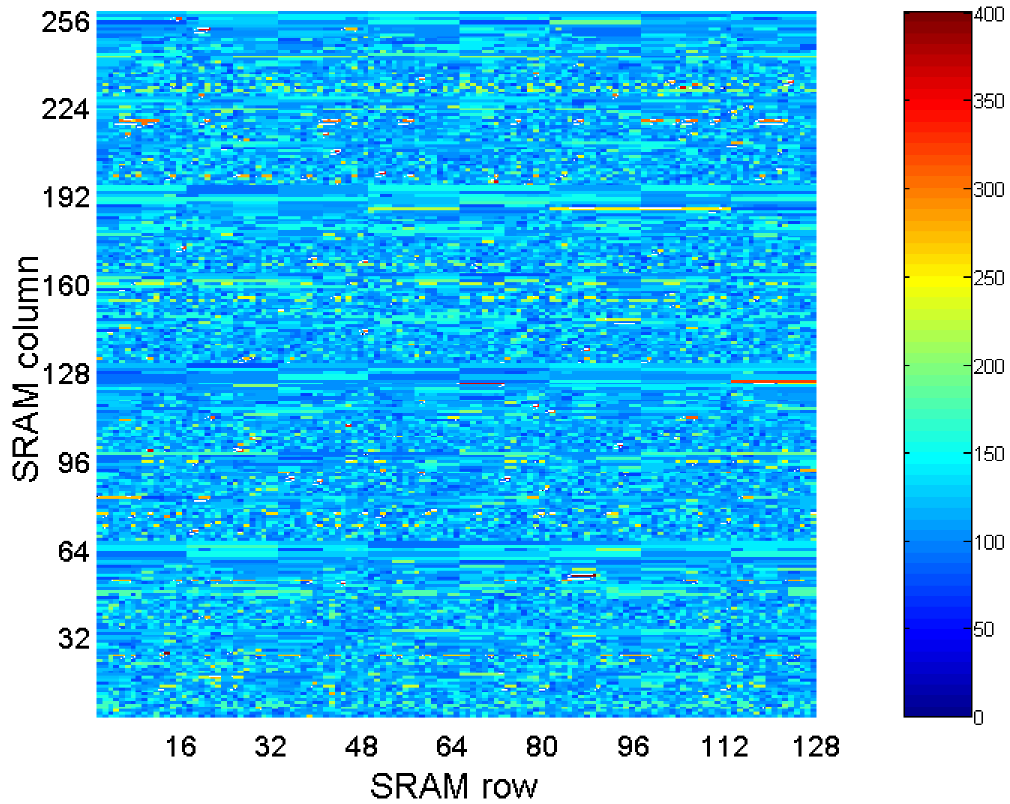


Figure 3.6. *DRV* spatial distribution of a 32K-bit SRAM chip.

Leakage measurement result of the 32K-bit SRAM is shown in Figure 3.7. The leakage current increases substantially with a high V_{DD}. This is caused by the DIBL effect in short channel transistors. Note that in the *DRV* anaysis of a typical SRAM cell, the DIBL effect modeled in Eq. 2.9 can be ignored because all the SRAM transistors operate in a weak-inversion mode. But when V_{DD} is significantly higher than the *DRV*, the DIBL effect causes a rapid increase in leakage current. This phenomenon reflects the

importance of low-voltage standby leakage control in future CMOS technologies, where the short-channel effect increases.

The shaded area in Figure 3.7 indicates the range of measured *DRV* (60-390mV). Although the memory states can be preserved at sub-400mV V_{DD}, adding an extra guard band of 100mV to the standby V_{DD} enhances the noise robustness of state preservation as discussed in section 3.1. With the resulting 490mV standby V_{DD}, SRAM leakage current can still be reduced by over 70%. Therefore the leakage power, as the product of V_{DD} and leakage current, is reduced by about 85% compared to 1V operation.

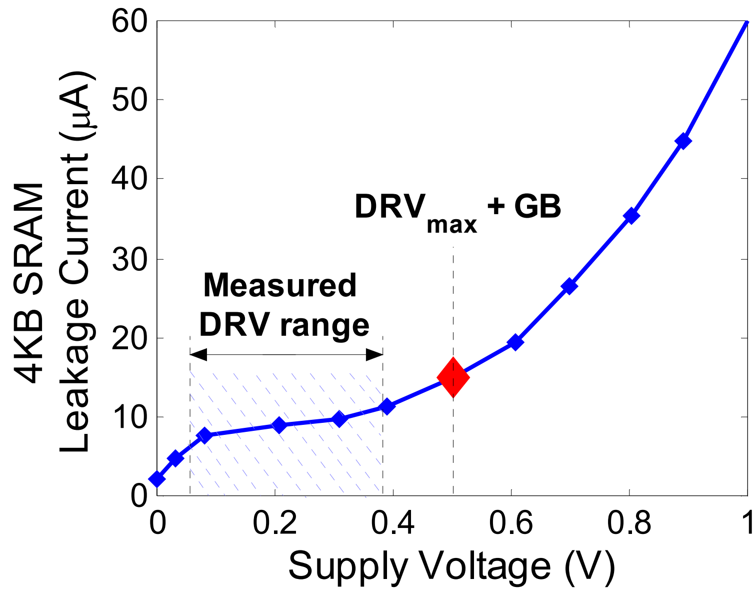


Figure 3.7. Measured leakage current of a 32K-bit SRAM chip.

The dual rail scheme is shown to be fully functional through the *DRV* measurements. With a 10MHz switch control signal, the SC converter generates the standby V_{DD} with less than 20mV peak-to-peak ripple. A wake up delay of 10ns is observed during mode

transition, while the time to enter sleep mode is around 10μs. The delay overhead in SRAM read operation is measured to be about 2X, which is reasonable for an ultra-low power application where the system clock period is typically 10 times the minimum operating cycle time of a low leakage SRAM.

3.2 *DRV* Measured in 90nm Technology

The *DRV* measurement result from our 130nm test chip showed that 85% of SRAM leakage power can be saved with reliable data retention by using a 490mV standby V_{DD}. An even higher leakage power reduction can be achieved by further lowering this standby V_{DD}, if the SRAM *DRV* can be effectively reduced.

According to the analytical *DRV* models of Eqs. 2.14-2.17, *DRV* is a function of the SRAM cell circuit parameters. Therefore the optimization of SRAM cell design can be used to minimize *DRV*. In order to explore the *DRV* sensitivities on circuit design parameters, an SRAM test chip in an industrial 90nm technology was designed.

3.2.1 Test chip design and implementation

This 90nm SRAM test chip is comprised of 64 memory arrays of differing cell sizing. Within each array there are 16-by-16 bits with the same sizing. The sizing variables include the channel length and *W/L* sizing ratio (β) of the access transistors (L_A, β_A), the pull-down NMOS transistors (L_N, β_N), and the pull-up PMOS transistors (L_P, β_P). Table 3.1 shows the design of array sizing, normalized to an industry standard SRAM cell. The A-arrays are a series of 25 memory arrays, from A1 to A25, with PMOS and NMOS channel lengths varied between 1 and 3 times of standard value. Similarly, the 25 B-

arrays use different sizing ratios for NMOS pull-down and PMOS pull-up transistors. While larger values were explored for most variables, a smaller β_N was preferred due to the strong strength of pull-down NMOS transistors in standard SRAM cell design. C-arrays test four configurations of access transistor sizing and channel length. The D-arrays mix sizing and channel length experiments on pull-down NMOS and pull-up PMOS transistors.

Array	Variable 1	Values	Variable 2	Values
A	L_P	1, 1.5, 2, 2.5, 3	L_N	1, 1.5, 2, 2.5, 3
B	β_P	1, 1.5, 2, 2.5, 3	β_N	1, 0.83, 0.66, 0.5, 0.33
C	L_A	2, 3	β_A	2, 3

Table 3.1. Normalized array sizing of the 90nm design optimization test chip

Besides sizing experiments, several other standby controls were implemented in this chip, including configurable NMOS and PMOS body bias voltages (V_{NB}, V_{PB}) and standby bitline voltage control. During standby mode, the bitlines can be connected to either V_{DD} or ground, or be left floating. A ground-switch on each array enables leakage measurement on a per-array basis. The design diagram and chip picture are shown in Figure 3.8.

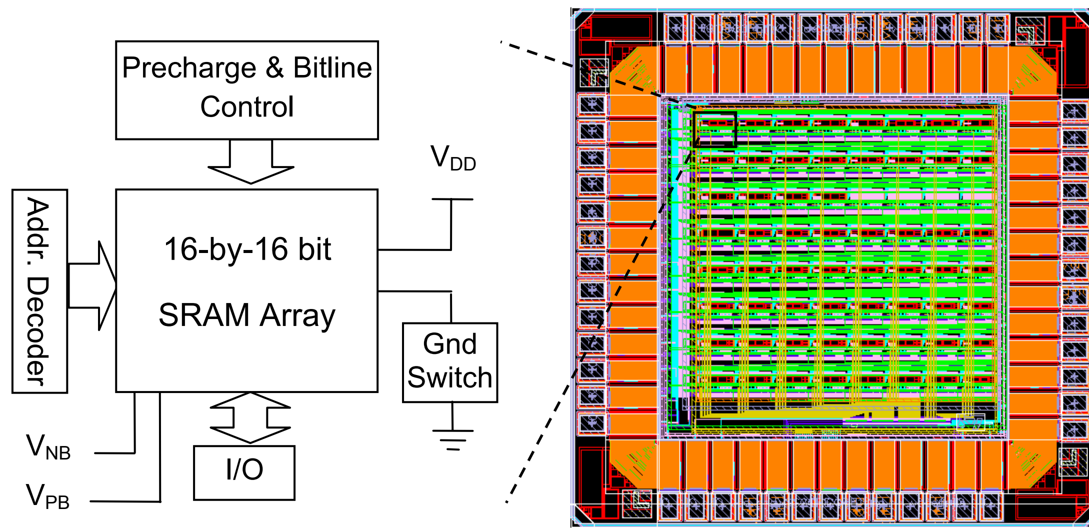


Figure 3.8. SRAM *DRV*-aware design optimization test chip in 90nm technology.

3.2.2 Measurement results

Indicated by color intensity, the measured *DRV* of one chip is shown in Figure 3.9. The average *DRV* of standard size arrays (A1 and B1) is around 140mV. The *DRV* decreases with larger channel length (A arrays), and increases when PMOS is sized between 2 and 3 times the standard size (B11 ~B20). For B21~B25 arrays when PMOS is 3X the standard size, strong pull-up strength causes instability or malfunction during write operation (at 1V). Such failures are indicated by the black spots in Figure 3.9.

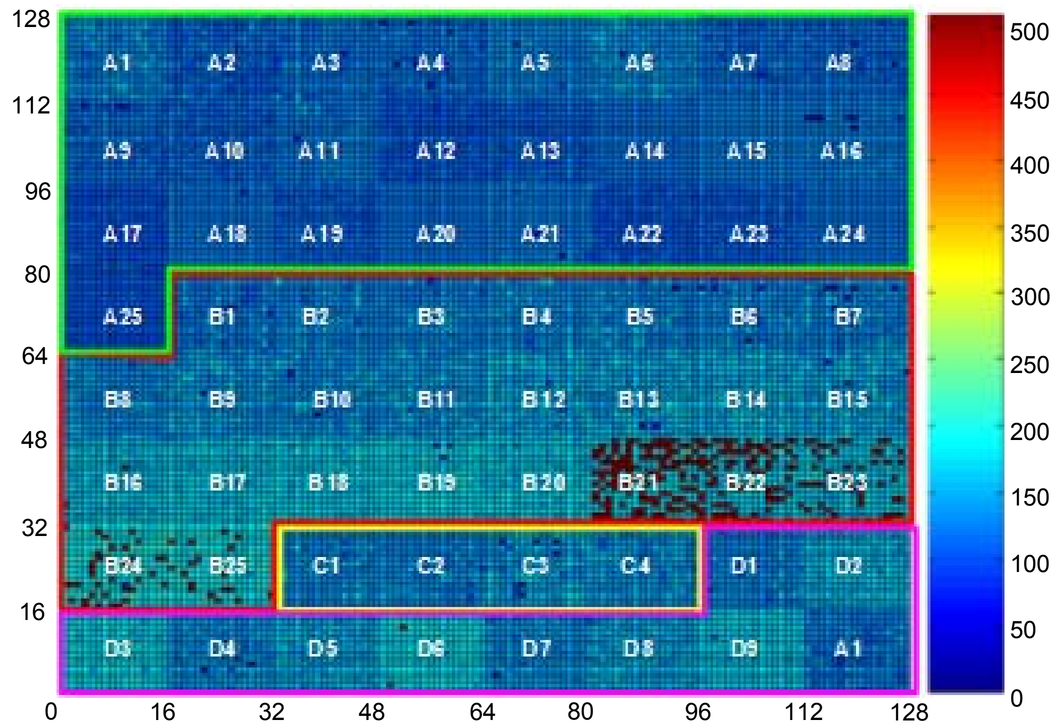


Figure 3.9. *DRV* sensitivity on design parameters (measured data from one chip).

In contrary to the high *DRV* sensitivity on pull-up PMOS and pull-down NMOS sizing, the sizing of the access transistors shows very small impact on *DRV* (C arrays). This is because of two factors. First the length of access transistor (L_A) in a standard SRAM cell is already larger than the minimum length. Therefore increasing L_A has a small effect on the access transistor leakage or the channel length process variation effect. Secondly, the access transistors have a relatively weak impact on *DRV* compared to the pull-up PMOS and pull-down NMOS transistors. The reasons include that only one access transistor contributes to the leakage balance of a standby SRAM cell (section 2.3.1), and that the access transistor gate-to-source voltage is lower than the other two types of transistors during very low voltage standby.

On the other hand, adjustments on the body bias voltage affect the leakage of all types of transistors exponentially, and cause a large impact on *DRV*. A quantitative analysis is shown in Figure 3.10 - Figure 3.12. These are the plots of the average *DRV* values measured from all SRAM arrays on 15 test chips. The goal is to show the impact of various design parameters on *DRV*.

Figure 3.10 shows that the *DRV* is a strong function of the body bias. For each V_{NB} there is an optimal V_{PB} that minimizes *DRV*, because balanced P/N ratio is a key factor in *DRV* optimization. For example, a –0.2V forward-biased V_{PB} produces the lowest *DRV* average value at zero V_{NB}. This indicates a weak P/N ratio in the standard SRAM cell sizing. On the other side, when V_{PB} is fixed the V_{NB} value has a two-fold impact on *DRV*, due to the existence of two different types of NMOS devices in an SRAM cell. A forward-biased V_{NB} leads to an even weaker P/N balance ratio, and at the same time significantly increases access transistor leakage. Both of these effects cause a *DRV* increase. As a result the average *DRV* increases from 140mV to 190mV with a 0.4mV forward-biased V_{NB}.

In summary, the measurement results in Figure 3.10 indicate that in order to minimize *DRV* mean value, reverse-biasing V_{NB} to suppress access transistor leakages and adjusting V_{PB} accordingly to achieve a balanced P/N strength ratio (zero V_{PB} in this design and technology) is a very effective method.

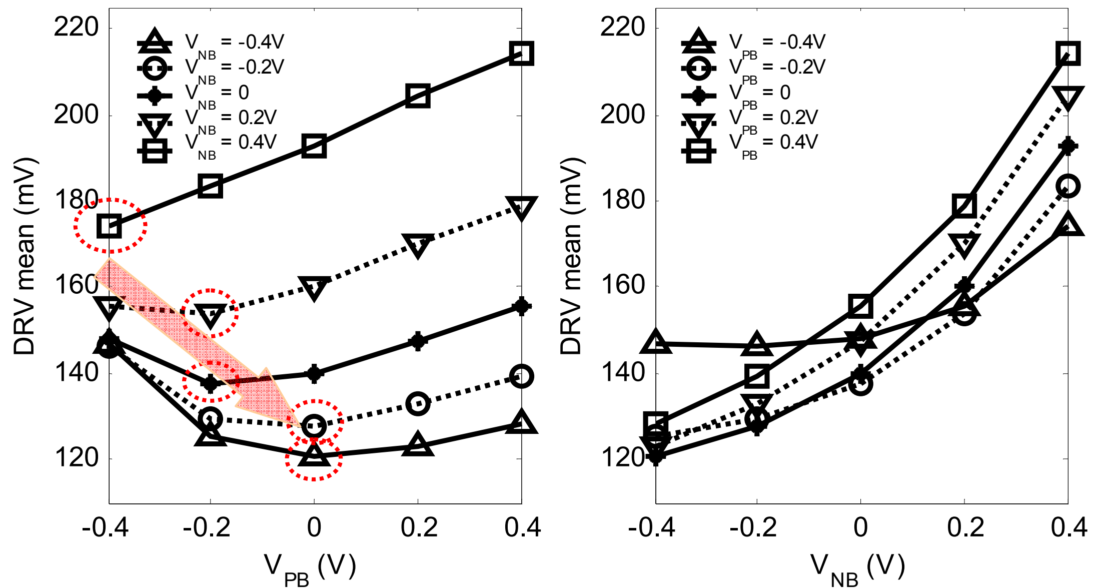


Figure 3.10. *DRV* sensitivity to body bias (standard size array)

Figure 3.11 shows that generally *DRV* reduces with larger channel length at fixed *W/L* ratio. The shape of *DRV* versus L_N curves is a result of the NMOS device characteristic in the 90nm technology this design used. For this device the threshold voltage and its variance are the lowest when L_N is at 1.5X the minimum length.

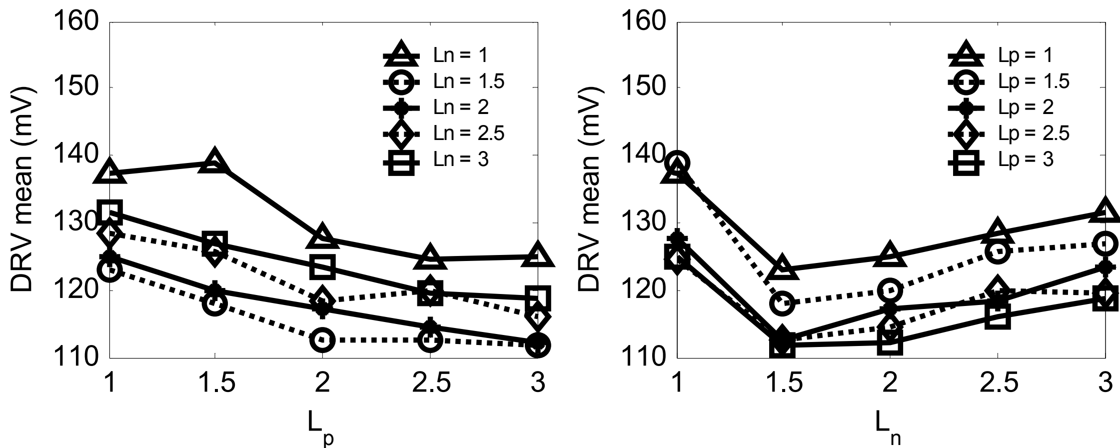


Figure 3.11. *DRV* sensitivity to *L* (zero body bias, standard *W/L* ratio)

Figure 3.12 shows that the device *W/L* sizing ratio has a relatively smaller impact on *DRV*, while the *DRV* improvement with a balanced P/N ratio can still be observed. For example, a 1.5X β_P minimizes the average *DRV*, because under very low V_{DD} a stronger PMOS holds the stored '1' state better against the pull-down NMOS leakage. But when β_P is further increased, *DRV* is higher again because the PMOS becomes too strong so that the store '0' state becomes unstable and causes the data-loss at low V_{DD}.

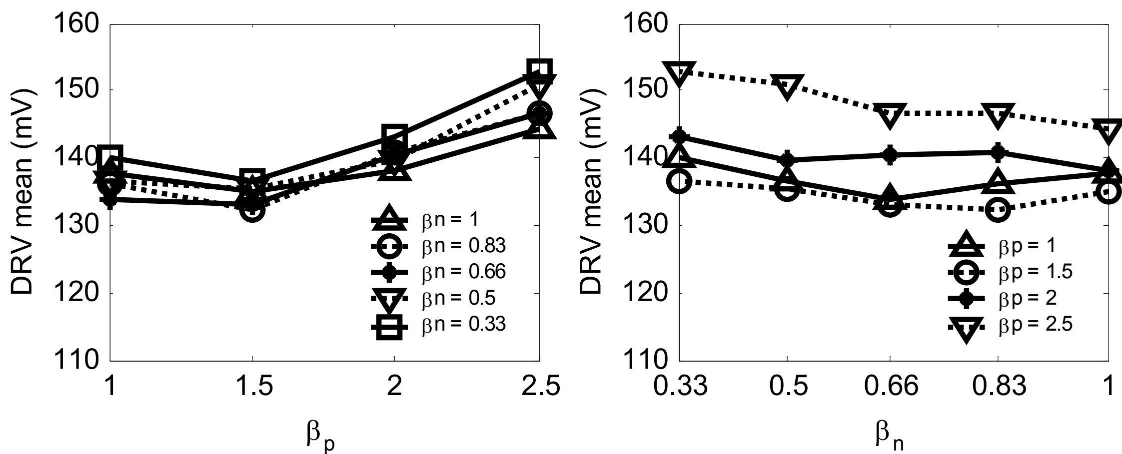


Figure 3.12. *DRV* sensitivity to *W/L* sizing ratio (zero body bias, standard *L*)

Finally, experiments showed that different standby bitline voltages cause less than 10mV difference in *DRV*. Similar to the small impact of the access transistor sizing on *DRV*, the bitline voltage has a weak influence on the access transistor leakage and the *DRV*. Temperature experiments show that the *DRV* increases at about 5mV/10˚C.

Design Parameter	Impact on *DRV*	Design Improvement for Lower *DRV*
Body Bias	High	Reverse-bias V_{NB} and adjust V_{PB} accordingly to achieve a balanced P/N leakage strength ratio
L_P , L_N	High	Use a larger L to reduce the process variation
β_P , β_N	Medium	No change (to avoid impact on active operations)
β_A, L_A and Bitline Voltage	Small (less than 10mV)	No change

Table 3.2. Summary of *DRV* sensitivity on design parameters

3.3 *DRV* Scaling Trend

Today exploring the SRAM ultra-low voltage data preservation is mainly for the interest of ultra-low power designs, but the technology scaling will soon bring up this topic to the memory designers for all-purpose applications. Scaling trends such as lower voltage, smaller device dimension and larger scale of integration all pose reliability hazards for memory design.

Our 130nm and 90nm *DRV* test chip results revealed the SRAM data-retention limits in the current-day technologies. A forward-looking investigation into the future technology nodes can be attained using SPICE simulations based on the Berkeley Predictive Technology Model (BPTM) [28]. The simulated results of *DRV* as a function of future technology dimensions are shown in Figure 3.13.

Due to the difficulty in predicting future technology process variations, an optimistic estimation is used in this simulation. The σ of device channel length variation is fixed at 10% of the mean value, and the σ of V_{th} variation is fixed at 10mV. The resulting SRAM

DRV scales in the contrary trend of V_{DD} reduction and approaches V_{DD} at sub-45nm nodes.

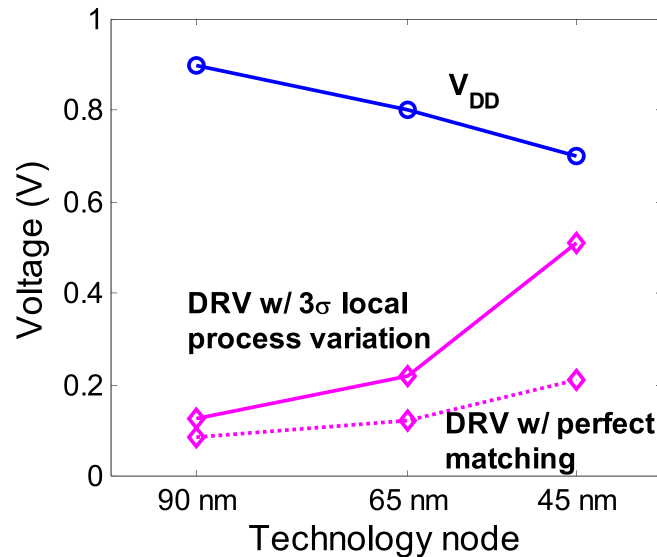


Figure 3.13. *DRV* and V_{DD} scaling trend.

The up-scaling trend of *DRV* is a result of both the dramatically increasing leakage current (which leads to degradation of I_{on}/I_{off}) and larger sensitivity of I_{off} to process variations at smaller technology dimensions. As a result of this *DRV* and V_{DD} scaling trend, severe reliability hazard of SRAM data preservation under the normal operation voltage is posed around 45nm technology node.

In order to meet the V_{DD} scaling and low power design requirements, effective design techniques are needed to reduce the SRAM *DRV*. Next in Chapter 4, we study the SRAM cell optimization as one of the solutions to *DRV* reduction.

4 *DRV*-Aware SRAM Cell Design

Since *DRV* is a function of the SRAM circuit parameters, a design optimization can be used to reduce *DRV*. At a fixed *SNM*, a lower *DRV* reduces the minimum standby V_{DD} and the leakage power. When the V_{DD} is fixed, a lower *DRV* improves the *SNM* and enhances the reliability of SRAM data retention. Traditionally, a standard SRAM cell is designed based on a performance-driven design methodology, which does not optimize the data retention reliability. For example, using a large NMOS pull-down device and a small PMOS pull-up device reduce data access delay, but cause a degraded *SNM* at low voltage. In order to gain a larger *SNM* and lower the *DRV*, the *P/N* strength ratio needs to be improved during the standby operation. In this chapter, we study the design techniques for a *DRV*-aware SRAM cell optimization.

Based on an industry standard SRAM design with realistic process variations, the methods to minimize *DRV* can be derived by analyzing the *SNM* during data-retention mode. In Figure 4.1 the solid lines show the VTC of a standard cell under *DRV* condition. The un-balanced VTC openings are caused by three reasons: a weak *P/N* strength ratio (P/N) that skews the VTC; process variations that further degrade both curves especially the one with a weaker PMOS; and the leakage through the access transistor that connects the state '0' to the bitline at V_{DD}. Therefore, to improve *SNM* and reduce *DRV*, the following methods can be used:

1) *Reduce process variation with larger channel length*

2) *Use a balanced P/N strength ratio during standby*

3) Suppress the access transistor leakage during standby

The *SNM* improvements after applying each of these techniques are shown in Figure 4.1. In a design practice, the *P/N* strength ratio and the access transistor leakage can be controlled with methods of body bias control and negative word line voltage during standby. The dynamic body bias control can also improve the active operation parameters (data access delay, read and write noise margins). This active operation improvement will be analyzed in section 4.2.3.

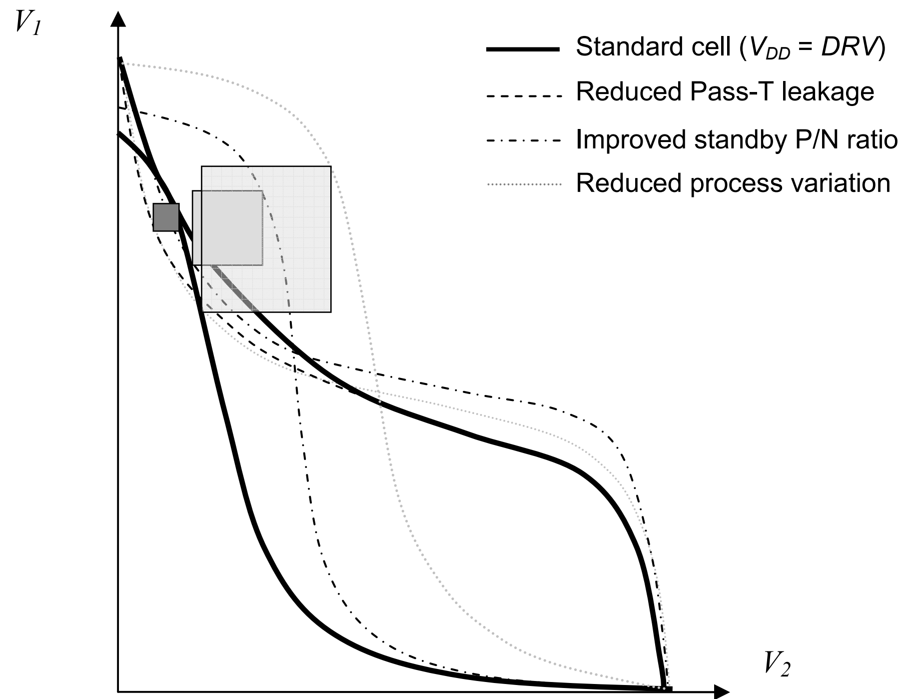


Figure 4.1. SRAM cell *DRV* minimization by improving data-retention *SNM*.

Figure 4.2 illustrates the quantitative *DRV* improvement from applying these design techniques. Simulated with an industrial 90nm technology model, the *DRV* of a standard-size SRAM cell is around 260mV, assuming a local mismatch in the V_{th} and L parameters

(local mismatch as defined in section 2.3.2). The magnitude of this local mismatch is 20% of the nominal parameter value. By reducing the process variations to zero, *DRV* can be lowered to 140mV. After using a balanced *P/N* ratio and suppressing the pass transistor leakages, the *DRV* approaches the technology theoretical limit of 50mV. In a practical design, such an effective optimization may not be possible. But understanding the factors that cause high *DRV* and knowing the corresponding *DRV*-aware design techniques help the designers build reliable SRAM for low-voltage standby operation.

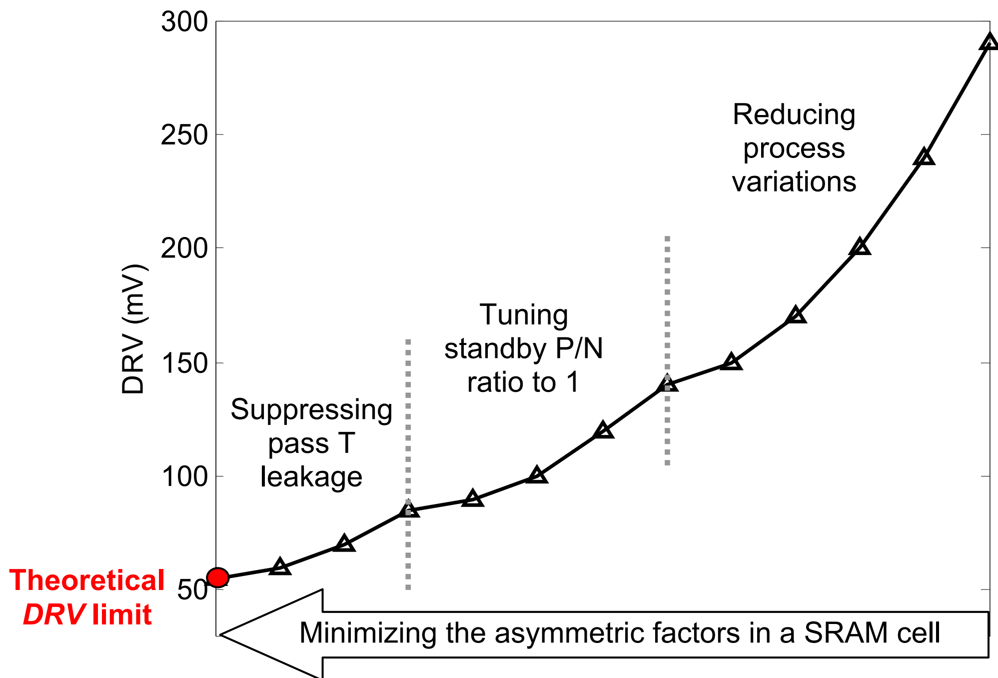


Figure 4.2. Approaching theoretical *DRV* limit with design approaches (90nm node)

4.1 *DRV* Design Model Based on the 90nm Technology Data

Based on the *DRV* sensitivity data attained from the 90nm test chip (presented in section 3.2), we can build a *DRV* design model that describes *DRV* sensitivity to the key

SRAM design parameters, including transistor sizing, channel length, and body bias. Such a design model provides an optimization tool to a *DRV*-aware SRAM design.

4.1.1 DRV design model

For the reader's convenience, the design parameters involved in an SRAM cell optimization are summarized in Figure 4.3. These design variables include the *W/L* sizing ratio and channel length of PMOS pull-up transistors (β_P, L_P), NMOS pull-down transistors (β_N, L_N) and NMOS access transistors (β_A, L_A), body bias voltages of PMOS and NMOS devices (V_{PB}, V_{NB}), and the bitline standby voltages (V_{BL}, $V_{BL_}$). Among these parameters, the access transistor sizing and bitline voltages have small impact on *DRV*, due to the reasons explained in section 3.2.2. Therefore our *DRV* design model focus on the other variables of larger impact on *DRV*, i.e. β_P, L_P, β_N, L_N, V_{PB} and V_{NB}.

In section 2.3, a *DRV* model based on process parameters of each individual SRAM cell transistor was developed as Eqs. 2.14-2.17. Using Eq. 2.9, I_i in Eq. 2.8 can be expressed in terms of design parameters as

$$i_k = a_k \times \frac{w_k}{l_k} \times \exp\left(a_2 \sqrt{\left|-2\phi_k + vsb_k\right|}\right) \times \exp(a_3 \exp\left(-\alpha l_k\right)) \qquad (4.1)$$

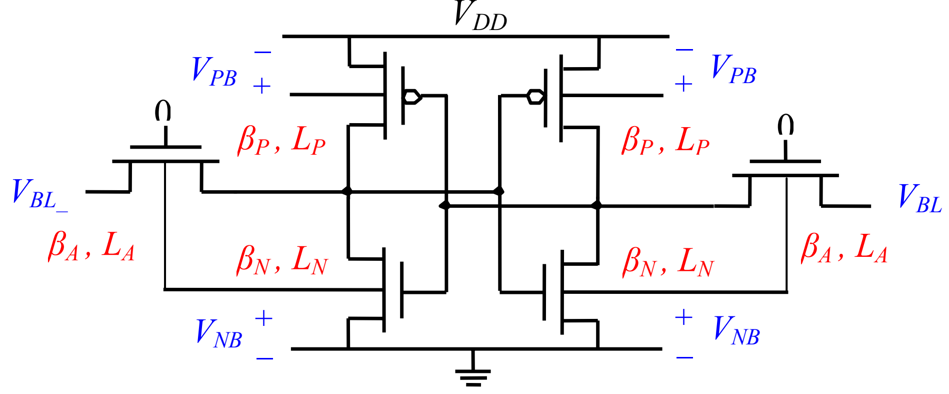


Figure 4.3. SRAM cell design variables

Using I_i (k=[1,5]) in in Eqs. 2.14-2.17, an expression for *DRV* in terms of the key design parameters is proposed in Eqs. 4.2-4.4. The model coefficients are summarized in Table 4.1. This model is general and scalable across all design parameters.

$$DRV = DRV^{(0)} + DRV^{(1)} \tag{4.2}$$

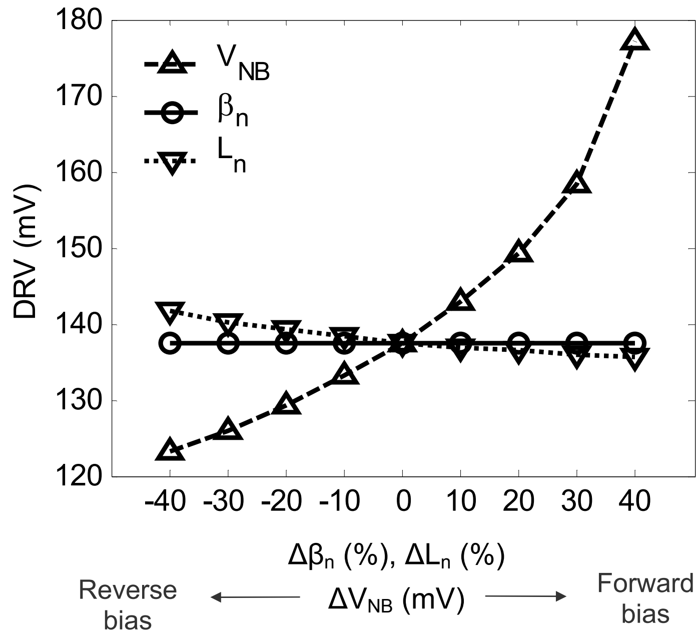
$$\begin{aligned} DRV^{(0)} &= a_1 \log\Big(\frac{1+\lambda_1/L_N}{\beta_P L_P + k_1} \cdot (a_2 \exp(\gamma_1\sqrt{|-2\phi_1 + V_{NB}|} - \gamma_2\sqrt{|-2\phi_2 - V_{PB}|}) + \\ &a_3 (1+\frac{\lambda_2}{L_P})(\beta_P L_P + k_2)\exp(\gamma_3\sqrt{|-2\phi_3 - V_{PB}|} - \gamma_4\sqrt{|-2\phi_4 + V_{NB}|}))\Big) \end{aligned} \tag{4.3}$$

$$\begin{aligned} DRV^{(1)} &= b_1 \log\Big(b_2 (1+\frac{\lambda_1}{L_P})\exp(\gamma_1\sqrt{|-2\phi_1 + V_{PB}|} - \gamma_2\sqrt{|-2\phi_2 - V_{NB}|}) + \\ &b_3 (1+\frac{\lambda_2}{L_N})\exp(\gamma_1\sqrt{|-2\phi_1 - V_{NB}|} - \gamma_2\sqrt{|-2\phi_2 - V_{PB}|})\Big) \end{aligned} \tag{4.4}$$

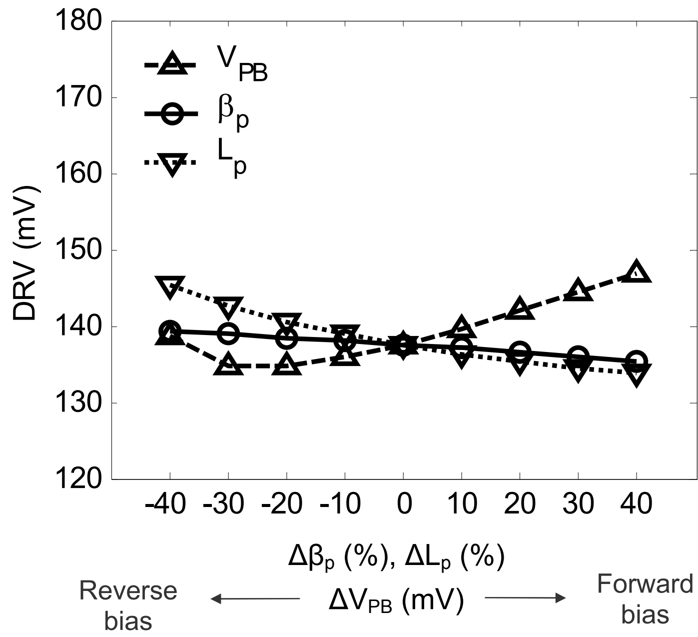
a_1	71.8	λ_1	0.0172	k_1	1	γ_1	1.0
a_2	2.52	λ_2	0.03	k_2	0.5	γ_2	1.0
a_3	6.17	ϕ_i	0.41	γ_4	0.3	γ_3	0.9
b_1	0.75	b_2	9.25	b_3	18.5		

Table 4.1. *DRV* design model

Figure 4.4 plots the modeled *DRV* sensitivities towards various design parameters. With an exponential influence on leakage, V_{PB} and V_{NB} have the largest impact on *DRV*. An NMOS reverse bias effectively reduces *DRV* because of a more balanced *P/N* ratio and a lower access transistor leakage. *DRV* decreases with stronger PMOS but increases again when the forward bias causes PMOS leakage to be stronger than NMOS with zero body bias. At a fixed *W/L* ratio, larger L_P and L_N reduce process variation and *DRV*. The *W/L* ratios, β_P and β_N alone have very little impact on *DRV*.



(a) *DRV* sensitivity to NMOS parameters



(b) *DRV* sensitivity to PMOS parameters

Figure 4.4. Modeled *DRV* sensitivities to SRAM cell design parameters

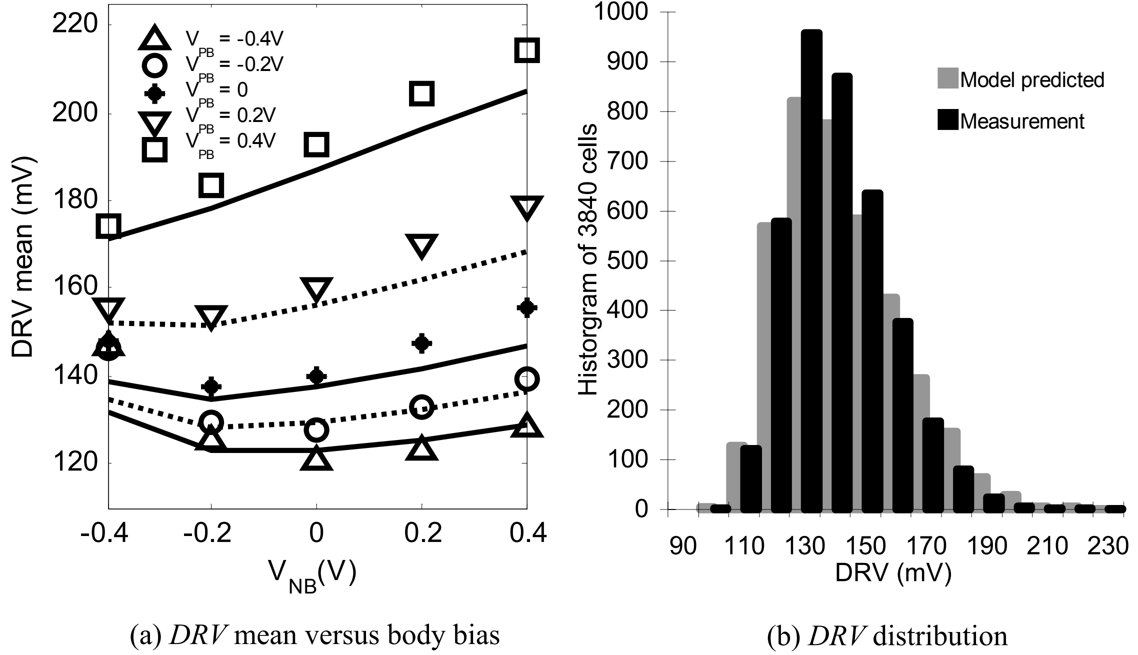
4.1.2 Model verification

The *DRV* design model can be verified by comparing both the model-predicted *DRV* values with the measurement data from the 90nm test chip (section 3.2). As shown in Table 4.2, the errors between predicted and measured *DRV* mean values are less than 5%.

Figure 4.5(a) plots the modeled *DRV* mean values over body bias, with the measurement data indicated by markers. Figure 4.5(b) shows the *DRV* distribution measured from 4K standard sized SRAM cells. Shown in the same figure is the distribution of 4K predicted *DRV* values generated from the *DRV* design model, assuming gaussian-distributed process variations. In both plots, the comparison between the measured and model predicted *DRV* data shows a close match. This *DRV* design model can be used to predict the highest *DRV* among all SRAM cells, when designing the minimum standby V_{DD} of a memory module.

DRV tuning over key design parameters	Average *DRV* (mV)	Average error btw predicted and measured *DRV* (mV)	Average Error / Average *DRV*
V_{sb} tuning	153.4028	6.6948	4.1642 %
L tuning	121.9328	5.5775	3.9742 %
W tuning	112.0048	3.5923	3.2073 %

Table 4.2. Prediction errors of the *DRV* design model



(a) *DRV* mean versus body bias (b) *DRV* distribution

Figure 4.5. DRV design model verification (standard size array)

4.2 *DRV*-Aware SRAM Cell Optimization Methodology

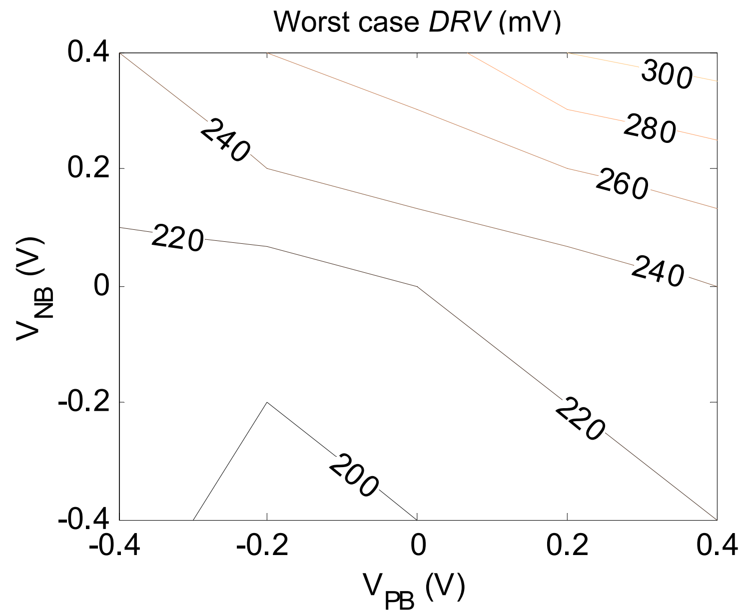
With supports from the *DRV* design model, an optimization analysis for *DRV* and SRAM leakage power reduction is discussed in this section.

4.2.1 Worst case DRV minimization

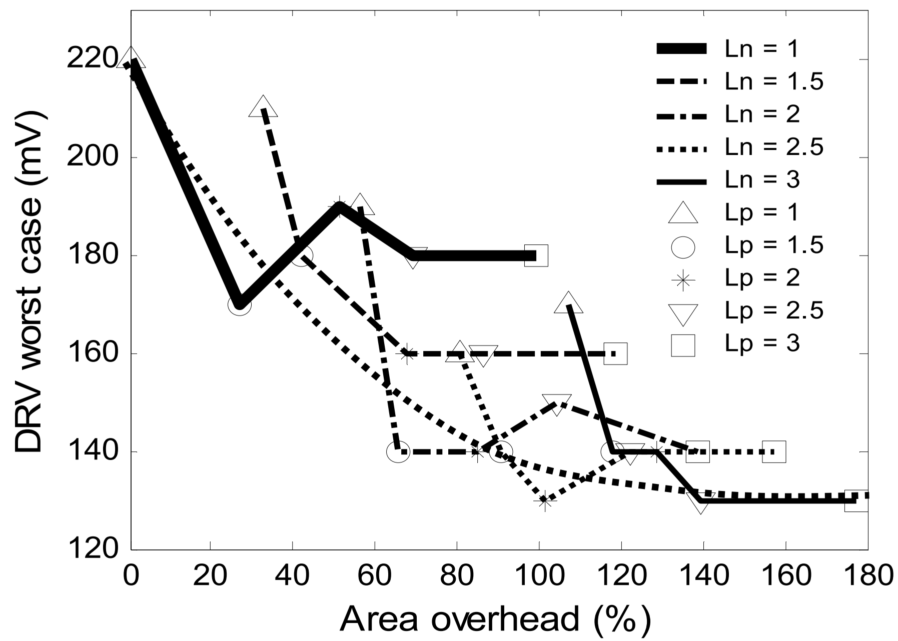
In order to preserve all data in an SRAM during a low-voltage standby operation, the minimum standby V_{DD} needs to be derived based on the worst case (w. c.) *DRV* value among all memory cells in one SRAM module. Using the *DRV* distribution in Figure 4.5(b) as an example, the w. c. *DRV* for this distribution is 220 mV.

The *DRV* design model predicts the SRAM *DRV* distribution given the memory size, SRAM cell design and magnitude of process variations. Therefore this model can be used to find a set of design parameter that minimizes the w. c. *DRV*. Based on model prediction, Figure 4.6 shows the w. c. *DRV* for an aggregate memory size of 4K bits (A1 array of 15 test chips with 256 bits in each array) over body bias and channel length. These predictions were confirmed with measurement data.

As shown in Figure 4.6(a), a reverse biased V_{NB} effectively reduces the w. c. *DRV* due to reduced access transistor leakage and improved *P/N* ratio. A forward biased V_{PB} also helps reduce the w. c. DRV due to the stronger PMOS and less variation in PMOS V_{th}. Figure 4.6(b) shows that a larger channel length lowers the w. c. *DRV* by reducing device mismatch, but involves a tradeoff with area overhead. A designer may select the optimal point to balance the *DRV* improvement with area constraint. For example, with a 50% larger channel length (30% extra area), a 50mV reduction in w. c. *DRV* and a 50% leakage power saving can be achieved. In a larger memory, the reduction in w. c. *DRV* is higher.



(a) Optimization with body bias



(b) Optimization with L

Figure 4.6. Worst case *DRV* optimizations

4.2.2 Leakage power minimization

The goal of a *DRV*-aware SRAM design is to reduce the memory leakage power. Based on the leakage current data measured from the 90nm test chip (presented in section 3.2), the following analysis shows the optimal leakage power saving produced by the SRAM design optimization.

While a reverse biased V_{NB} and a larger L reduce both *DRV* and the leakage current, forward biased V_{PB} minimizes *DRV* but increases leakage. To investigate the optimal bias scheme for leakage minimization, the leakage of each SRAM array on the 90nm test chip was measured under different bias conditions. During the leakage measurement, only the ground switch of the measured array is turned on, and all other 63 SRAM arrays are turned off. Typically, the leakages through the other turned-off ground switches add up to about 5 times the leakage being measured (through the array with its ground switch turned-on). By carefully estimating the turned-off resistances of each array and extracting the turned-on array leakage current from the measurement results, the individual array leakage current is attained.

Figure 4.7 shows the measured leakage power of a standard sized array at 100mV margin above the w. c. *DRV* at various body biases. Although reverse-biased V_{NB} and forward-biased V_{PB} minimize the w. c. *DRV*, reverse bias on both V_{NB} and V_{PB} minimizes the leakage power.

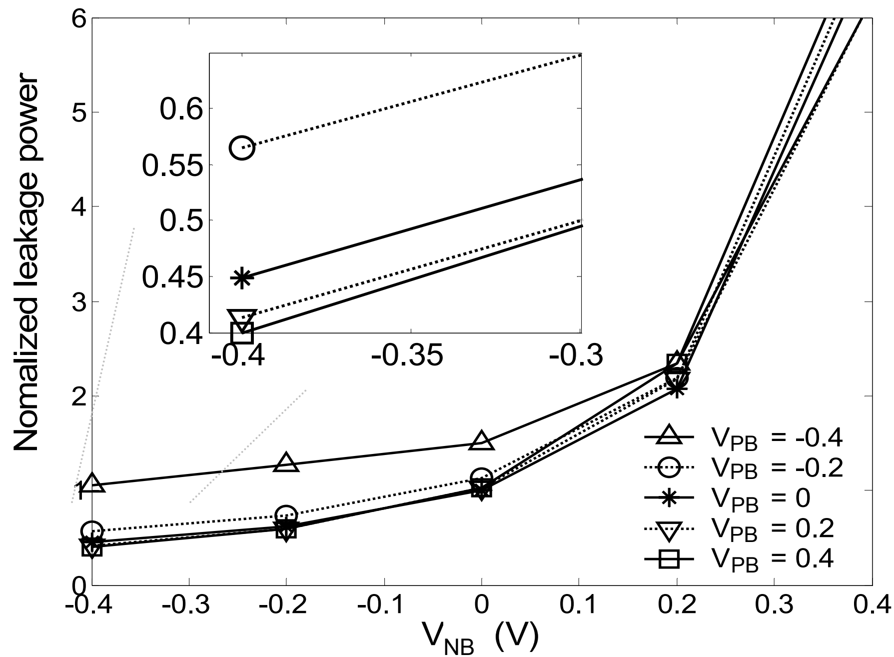


Figure 4.7. Leakage power minimization with body bias

As a summary, the SRAM leakage power minimization requires reverse body bias and larger channel length at a cost of area. *DRV* can be further reduced by forward biasing V_{PB} but with higher leakage power. The impact of larger channel length and body bias control on memory active operation metrics (performance, read and write reliability) will be analyzed in the next section.

The leakage savings by applying *DRV*-aware SRAM cell optimization methods are quantitatively shown in Figure 4.8, which plots the leakage measured from standard size array A1 and array A7 (with a 50% larger channel length in both L_P and L_N). For a standard SRAM design, the leakage power can be reduced by 10X from standby at 1V (A) to standby at 320mV (B), the w. c. *DRV* plus 100mV safety margin. With 400mV reverse body biases for both PMOS and NMOS (bias scheme I), the w. c. *DRV* does not

change but the leakage power reduces by 2X (C). Furthermore, by using a larger channel length in A7, the w. c. *DRV* is 50mV lower, and leakage power is reduced by another 2X (D). The overall standby power saving with optimized SRAM design at 270mV standby V_{DD} is 75% compared to standard cell standby at w. c. *DRV*, and 97% compared to standard cell standby at 1V.

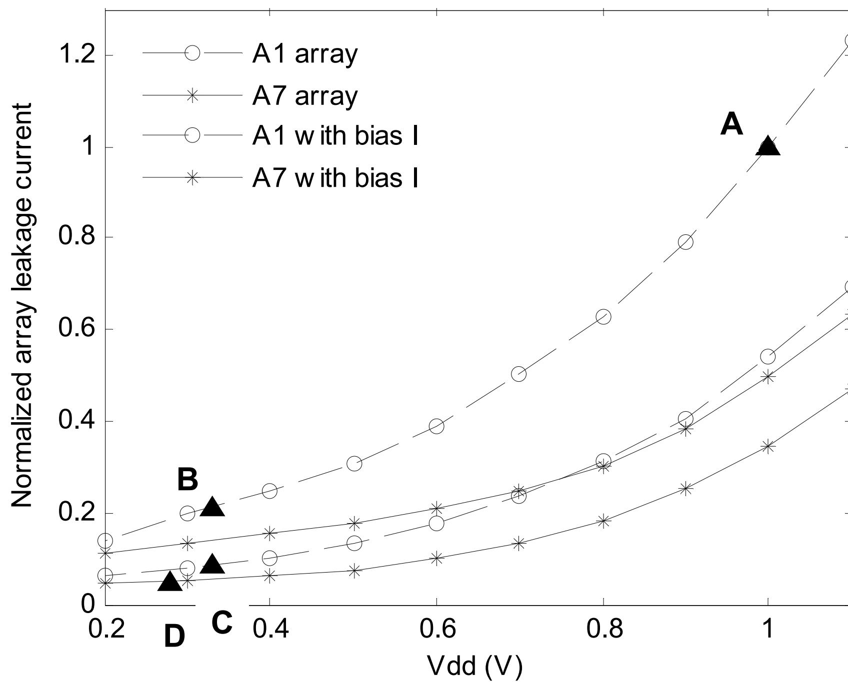


Figure 4.8. *DRV*-aware optimization for leakage saving

Finally, Figure 4.9 shows the cell optimization impact on *DRV* distribution (measured from the 90nm test chip). Compared to the standard array, *DRV* distribution of the leakage-optimized design (A7 array with 400mV reverse body bias) moves towards the lower end with narrower spread. The *DRV* mean value is 30mV lower and the w. c. *DRV* is 50mV lower. With an error correction scheme to correct the errors at the end of *DRV*

distribution, the minimum memory standby V_{DD} can be further reduced. This error-correction scheme design will be discuss in Chapter 5.

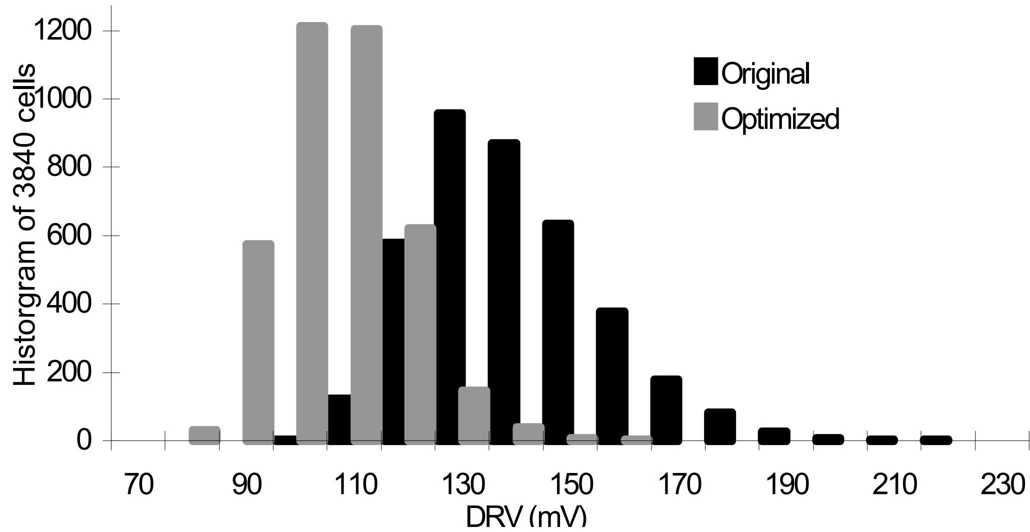


Figure 4.9. Measured *DRV* distributions before and after design optimization

4.2.3 The optimization impact on active operation metrics

Besides reduction in *DRV* and leakage power, the impact of *DRV*-aware SRAM cell optimization on active (read and write) SRAM operations are analyzed in this section. Based on a simulation analysis, Figure 4.10 shows the active noise margins and read delay with and without the *DRV*-aware design optimization. This analysis uses an industrial 90nm technology with 20% local variations in V_{th} and L. The read margin is defined as the maximum square between the inverter VTC curves during the read operation [21], and the write margin is defined in the latest proposed method [29]. The read access delay is characterized as the time it takes to discharge the bitline to 90% V_{DD} level, with an approximated bitline capacitive load of 5pF.

As shown in Figure 4.10(a), the write margin decreases with larger L, especially L_P. This is because larger L_P reduces the PMOS V_{th} and causes an increased difficulty writing state '0' into the SRAM cell inverter that originally holds state '1'. Such a situation can be improved by applying a 400mV reverse PMOS body bias and a 400V forward NMOS body bias (bias scheme II) during the write operation. By boosting the NMOS to PMOS strength ratio the write margin can be improved by 80~100mV, resulting in a higher reliability than in the original SRAM cell.

On the other hand, Figure 4.10(b) shows that the read margin improves with larger L by 5mV to 35mV, due to the reduced mismatch in the read access circuit path, formed by the access transistor and the pull-down NMOS device. Therefore the body bias control can be used to improve the other important SRAM cell design metric, the read performance. By applying 400mV forward body bias to both NMOS and PMOS (bias scheme III), the read margin is about 5mV lower than without body bias, but still 30mV higher than the original design. But the read delay is reduced by more than 10%, as shown in Figure 4.10(c).

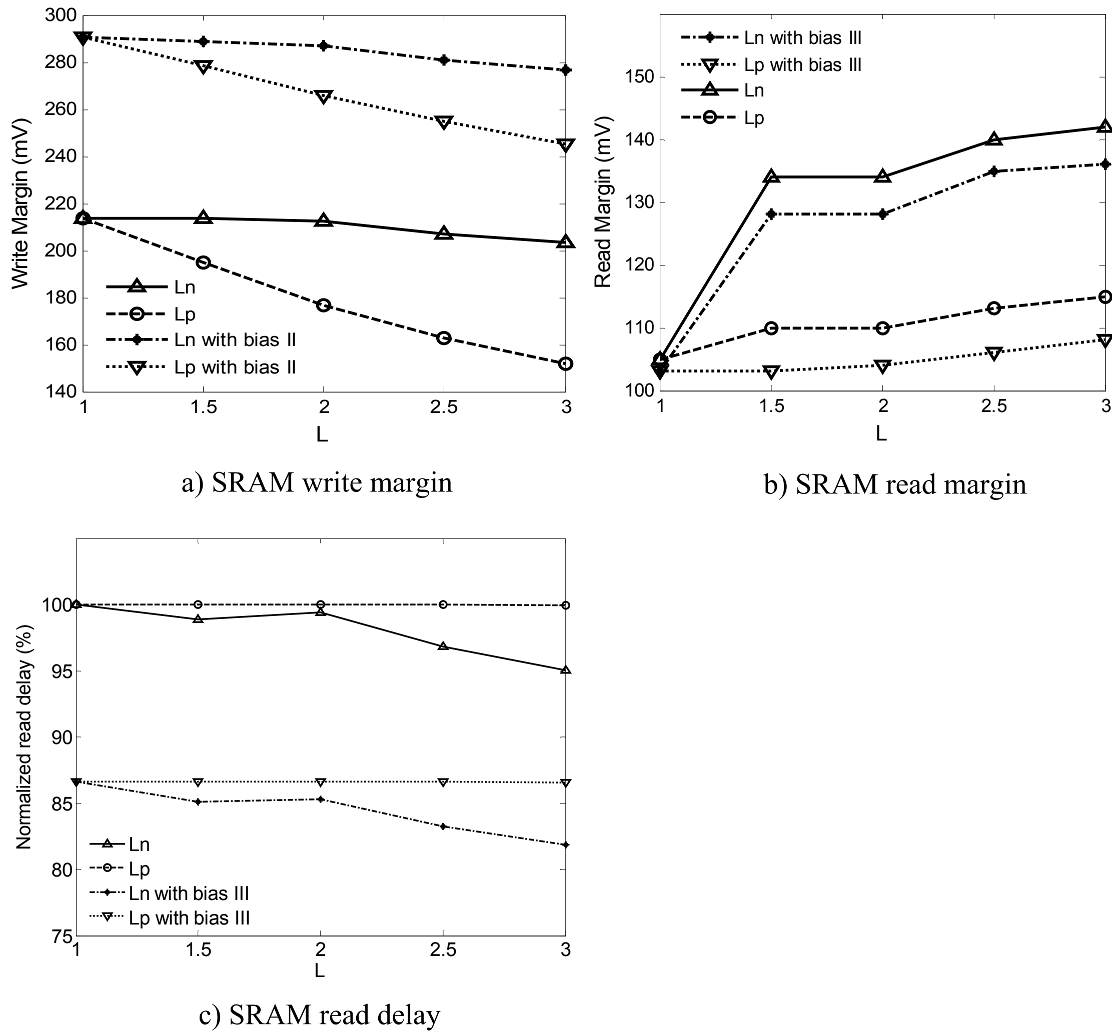


Figure 4.10. *DRV* optimization impacts on active operation parameters

As a summary, the *DRV* optimization techniques of applying body bias control and using larger channel length can be used effectively to improve both the active and standby operations. The critical tradeoff is between the optimized results and the area penalty (caused by larger channel length and the body bias control).

4.3 90nm SRAM *DRV*-Aware Design Optimization Summary

The key design parameters in SRAM cell optimization are *W/L* sizing ratio and channel length of PMOS pull-up transistors (β_P, L_P), NMOS pull-down transistors (β_N, L_N), as well as the body bias voltages of PMOS and NMOS devices (V_{PB}, V_{NB}).

With an exponential influence on leakage, the body bias (V_{PB}, V_{NB}) has the largest impact on the *DRV* average value. Optimization analysis based on the standard SRAM cell design and a 90nm technology found that a reverse biased V_{NB} and zero V_{PB} minimizes the *DRV*, due to the more balanced *P/N* standby strength ratio and lower access transistor leakage. However, when considering leakage suppression, reverse bias in both V_{PB}, and V_{NB} achieves the lowest leakage power even though the *DRV* is slightly higher than the optimum case.

At a fixed *W/L* ratio, larger channel length (L_P, L_N) effectively reduces process variation and the worst-case *DRV* value, at a tradeoff with area overhead. With 4K SRAM cells, 50% larger *L* (30% extra area) leads to 50mV reduction in w. c. *DRV* and 50% leakage power saving. In a larger memory, the reduction in w. c. *DRV* is going to be higher.

The *W/L* sizing ratio (β_P, β_N) has a relatively small impact on *DRV*. In order to avoid impact on active SRAM cell operation, these sizing ratios are not changed in a *DRV*-aware SRAM cell optimization. Analysis in simulation showed that using the same *DRV* optimization techniques (larger *L* and adjustable body bias control), the SRAM cell read speed and operation noise margins (read and write) can be improved. Therefore the

optimization benefits both active and standby operation. The critical tradeoff is with the area penalty caused by larger channel length and body bias control.

The 90nm test chip measurement data showed that the w. c. *DRV* of 4K SRAM cells is 220mV, with an average of 140mV (90nm industry technology). By optimizing the SRAM cell design, the w. c. *DRV* value can be effectively reduced by 50~100mV, resulting in 75% reduction in leakage power compared to the original SRAM (standby at un-optimized w. c. *DRV*).

Following table summarizes the proposed design improvements that minimize the DRV and SRAM leakage power.

Design Parameter	Minimize *DRV*	Minimize Leakage Power	Impact on Active Operation
Body Bias	Reverse-bias V_{NB}, and adjust V_{PB} accordingly to achieve a balanced P/N leakage strength ratio	Reverse-bias V_{NB} and V_{PB}	Improves write margin and read speed
L_P , L_N	Use a larger L to reduce the process variation	Use a larger L to reduce *DRV* and suppress leakage	Improves read margin

Table 4.3. Summary of *DRV*-aware SRAM optimization for a 90nm technology

5 Error-Tolerant SRAM Design for Ultra-Low Power Standby

As shown in the 90nm and 130nm test chip measurement data, process variations cause a large variation in the *DRV* values for different cells on the same SRAM chip. The traditional design method for reliable data-retention is to set the SRAM standby V_{DD} at a level above the w. c. *DRV* value. In contrast to this worst-case design approach, we propose an aggressive reduction of the standby V_{DD} below the worst case *DRV* to further reduce the leakage power. The SRAM data is reliably preserved by using ECC to correct the occasional data-retention errors due to standby operation below the worst-case voltage requirement.

5.1 ECC Analysis for Low Voltage SRAM

In the past, ECC has been used in many SRAM designs to enhance the data storage reliability against manufacturing defects and soft errors [30, 31, 32]. This work is different because the focus is on power versus redundancy tradeoff instead of reliability versus redundancy tradeoff. To optimize the ECC design for power minimization, we first build a model of the SRAM standby power as the optimization metric.

5.1.1 Modeling the SRAM standby power

Figure 5.1 shows the empirical distribution of the *DRV* values among 4K bits measured from the 90nm test chip. Setting the standby supply voltage v_S to be slightly larger than the w. c. *DRV* value DRV_{max} among all cells in an SRAM (here equal to

*190*mV) is a worst-case design approach. This method guarantees reliable data retention at the cost of a typically high per-cell (per-bit) leakage power given by $G(DRV_{max})^2$, where G is a constant (the leakage-current in the 100-200mV range is approximately linear in the supply voltage).

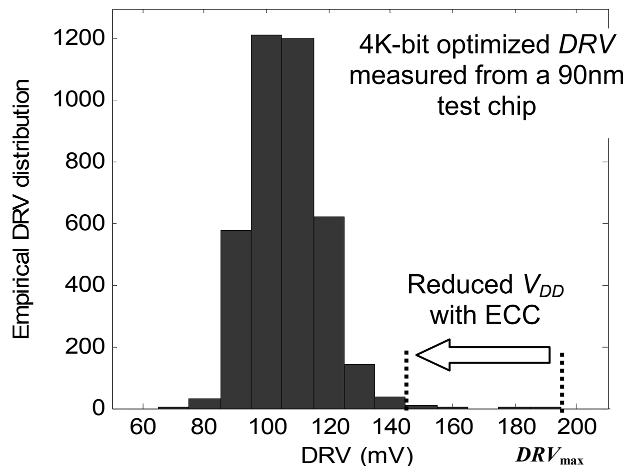


Figure 5.1. Minimizing SRAM standby V_{DD} with error correction

For an *N*-bit SRAM, reducing the standby supply voltage v_S to a level well below DRV_{max} reduces the leakage power but also renders $N \cdot r(v_S)$ cells unreliable, where $r(v_S)$ denotes the fraction of cells whose *DRV* is greater than v_S according to the empirical *DRV* distribution (Figure 5.1). This motivates the use of an (*n, k, d*) error correction code, which guarantees the recovery of a *k*-bit information word if there are no more than $t = \lfloor (d-1)/2 \rfloor$ errors in its stored *n*-bit coded representation. Based on this, the power per useful bit function of the aggressive low voltage standby scheme is defined as [33]:

$$P_\varepsilon(v_S) := \frac{n}{k} G \cdot {v_s}^2 + \frac{E_C}{kT_s}, \qquad (5.1)$$

where E_C is the average encoder-decoder computational energy over a codeword of k information bits, and T_S is the standby period between two error corrections. The ECC code needs to be able to correct all the $N \cdot r(v_S)$ data-retention errors. This function can be thought of as a cost-function to be minimized in our ECC optimization.

5.1.2 Power per useful bit bounds

To derive the theoretical bounds for SRAM standby power per useful bit, we first assume that the standby time, T_S is large, so that the second term in Eq. (5.1 approaches zero. ECC theory in

[33] shows that there exist linear codes with parameters satisfying

$$\frac{Gv_S^2}{1-h(r(v_S))} \leq \frac{Gv_S^2}{(k/n)} \leq \frac{Gv_S^2}{1-h(2r(v_S))}, \qquad (5.2)$$

where $h(x) = -x \log_2(x) - (1-x) \log_2(1-x)$ is the binary entropy function. This shows that there is an optimum value of v_S which minimizes the power per useful bit function. In addition, if v_S is selected to satisfy $(t/n) > r(v_S)$, then the percentage of words having more than t cells with DRV larger than v_S can be made arbitrarily small for all n large enough

[33]. A small number of redundant rows can be used to fix these errors.

By ignoring the ECC computation overhead given by (E_C/kT_S), the power per useful bit bounds can be plotted as a function of v_S in Figure 5.2. The figure illustrates the upper and the lower bounds on $P(v_S)$ as a function of $r(v_S)$, or the failure rate, as the supply voltage is changed. The minimum achievable upper bound is 40% lower than the leakage

power using a worst-case standby V_{DD}. The minimum value of capacity lower bound is 49% lower than the worst-case leakage power.

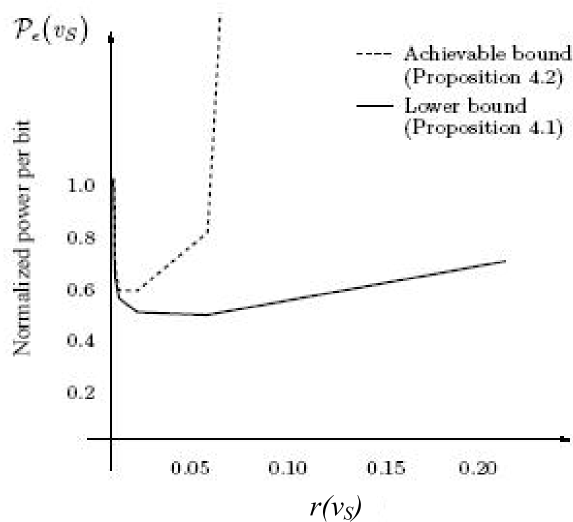


Figure 5.2. Upper and lower bounds on SRAM standby power per useful bit [33].

5.1.3 Code implementation

As predicted by Eq. 5.2, a large variety of error-correction codes can be used to produce positive SRAM leakage power reduction compared to the worst-case standby V_{DD} design. However, a certain type of codes, such as asymptotic codes or codes with very large block-length cost high energy consumption, require large-silicon area, and cause decoding latency (in the number of clock cycles). These design overheads increase the $P(v_S)$ metric by adding a large (E_C/kT_S) term, or degrades the system performance. These considerations motivate the study of low-complexity ECC designs. The (31, 26, 3) single bit-error correcting Hamming code proposed in [33] is a particularly attractive

design choice with 33-40% estimated power reduction for the test-chip distribution of Figure 5.1. This code has an estimated latency of 1-clock cycle (2ns). Implemented in an industrial 90nm technology, the estimated average encoding plus decoding energy is E_C = (0.93 + 2.32)pJ. The estimated leakage current at 200mV for 256 cells is 55.76nA. With these numbers, a standby duration $T_S \geq$ 100ms will achieve a power per useful bit reduction of 33%.

5.2 An Implementation of Ultra-Low Leakage Error-Tolerant SRAM

Chapter 4 and Section 5.1 introduced two design techniques for SRAM leakage power reduction at very low standby V_{DD}. In order to prove the effectiveness of these schemes, a 26k-bit ultra-low leakage error-tolerant SRAM design was implemented in an industrial 90nm technology. The chip design and measurement data are presented in this section.

5.2.1 Chip design

Figure 5.3 shows the error-tolerant SRAM design diagram. Based on an industry IP SRAM module, a 26k-bit SRAM was implemented with the *DRV*-aware leakage optimization design and a error-correction scheme (31, 26, 3) Hamming ECC. The SRAM cell optimization includes a 50% larger L in the pull-up PMOS and pull-down NMOS devices, and an adjustable body bias control for memory cell arrays. The configurable body bias control in this design did not involve an extra area overhead, since our industry-IP SRAM module used separate metal grid for body bias connections. By reconnecting this grid from the V_{DD} and ground contacts to the external V_{PB} and V_{NB} pins,

flexible body bias control was achieved. Such an optimized SRAM with ECC design has a total of 50% larger area than the original memory module.

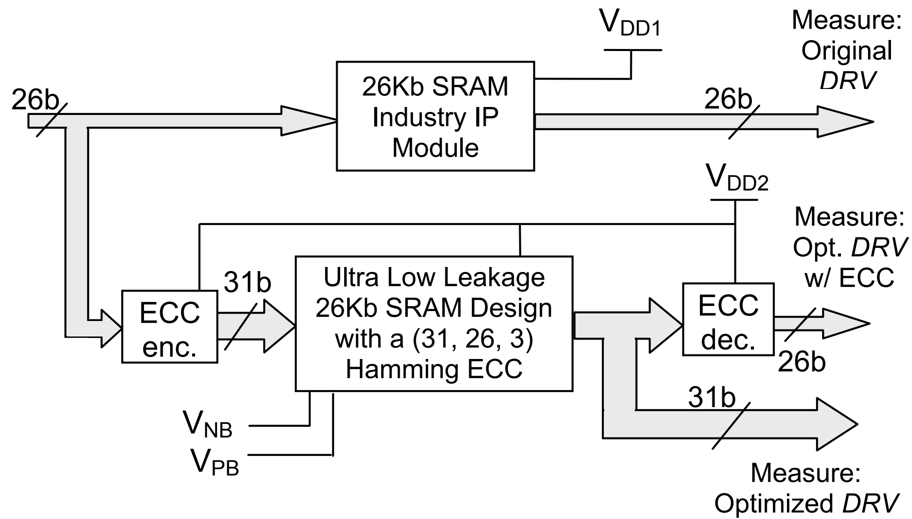


Figure 5.3. Error-tolerant SRAM chip design diagram.

The chip picture is shown in Figure 5.4. For comparison purpose, the chip also contains an original SRAM module with separate power supply.

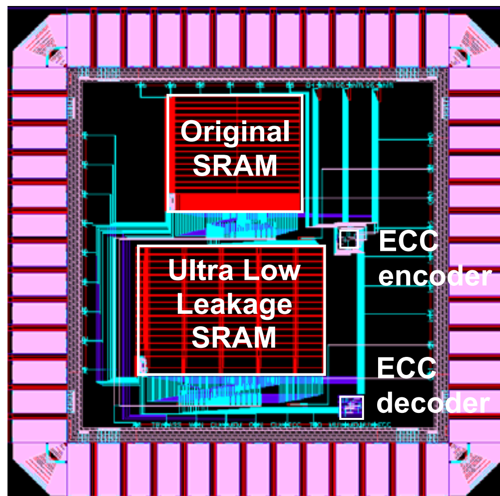


Figure 5.4. Ultra-low leakage SRAM chip in a 90nm industry technology

5.2.2 Measurement results

To investigate the optimization improvements on *DRV* and leakage power, the values of original *DRV*, optimized *DRV* and optimized *DRV* after error-correction were measured on 24 chips. Figure 5.5 shows the *DRV* data from one chip. Due to the large process variations, *DRV* of the original SRAM design ranges from 120mV to 550mV. With cell optimization the w. c. *DRV* is reduced to 220mV. After error correction, the highest *DRV* becomes 155mV, and the *DRV* distribution becomes much narrower than the original design.

The measured w. c. *DRV* values of 24 chips are shown in Table 5.1. Obviously, both the circuit optimization and the ECC scheme effectively reduced the range of *DRV* variation in the high end. As a result, the chip-level w. c. *DRV* value is lowered by 180-410mV among the chips measured.

	Original *DRV*	Optimized *DRV*	Optimized *DRV* w/ ECC
W.C. Min.	320 mV	170 mV	140 mV
W.C. Max.	570 mV	220 mV	160 mV

Table 5.1. Measured worst-case *DRV* range among 24 chips

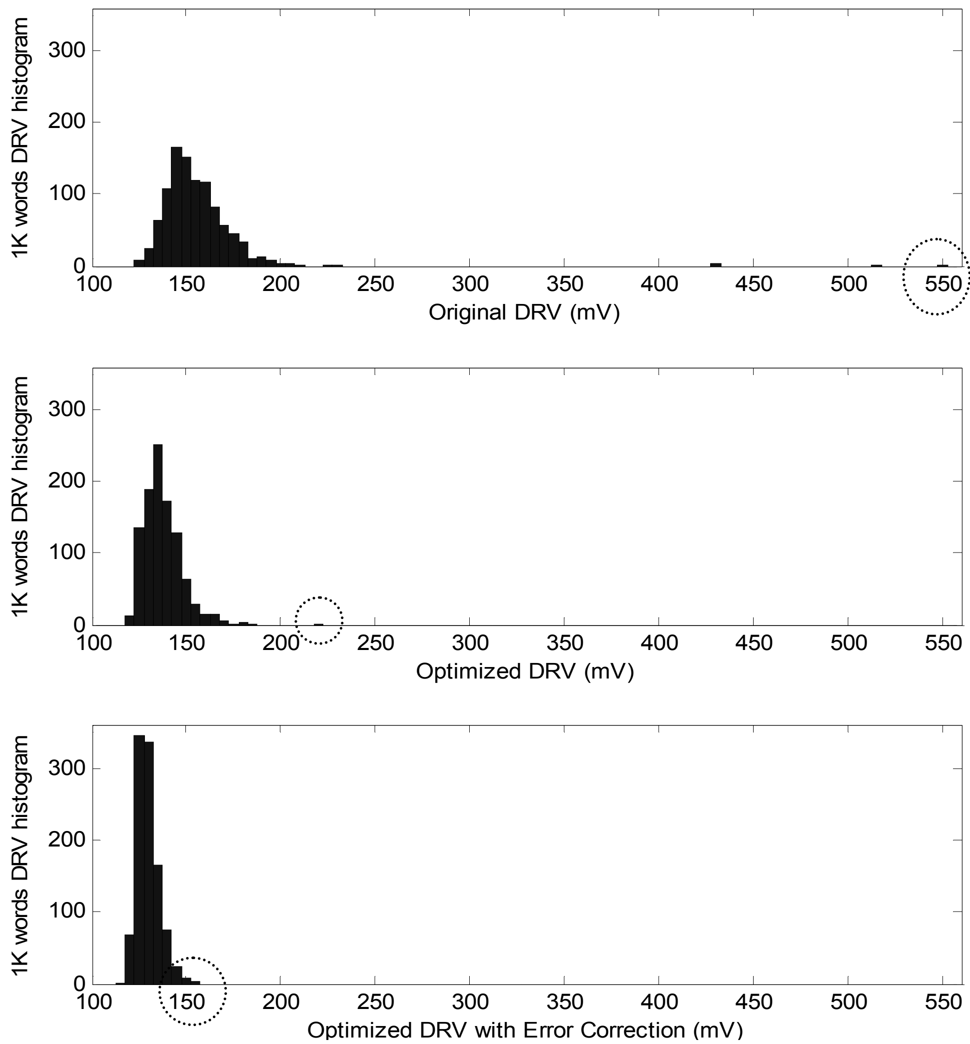
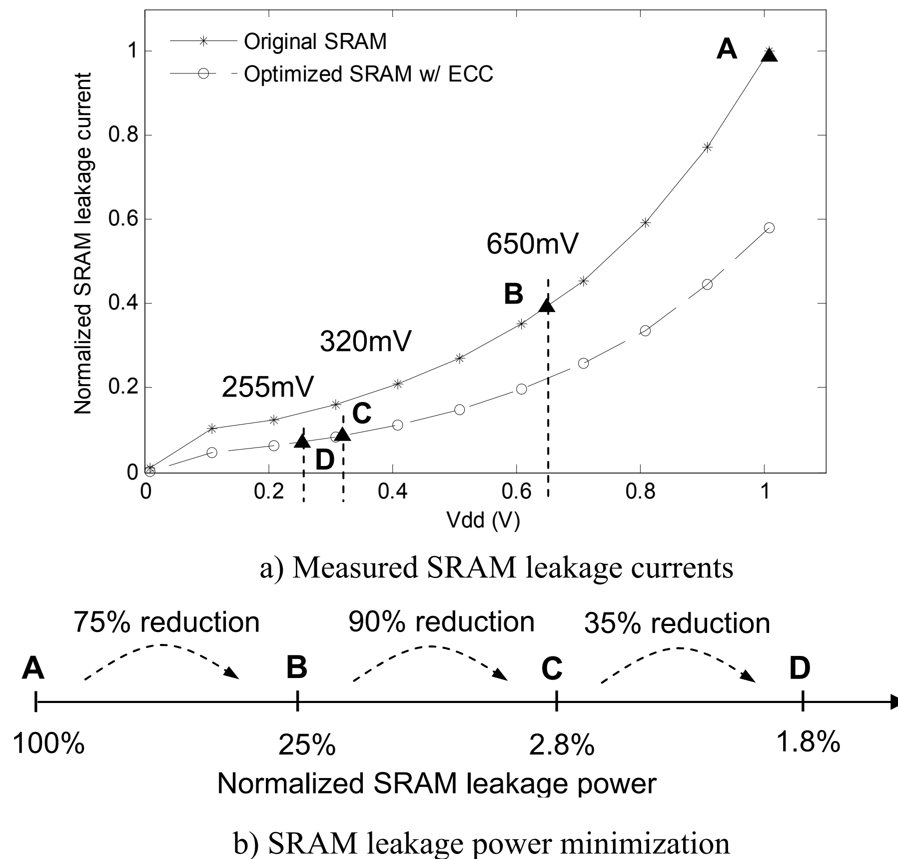


Figure 5.5. Measured *DRV* distributions from the ultra-low leakage SRAM chip

The reduced w. c. *DRV* values lead to a lower SRAM standby V_{DD} and a larger leakage power reduction. Figure 5.6(a) shows the measured leakage currents of both the original and optimized SRAM modules. Since both larger *L* and reverse body bias effectively reduces SRAM cell leakage, the total leakage of the error-tolerant SRAM is 40% lower than the original memory, despite the additional ECC parity bits overhead. Figure 5.6(b) quantitatively illustrates the leakage power savings with ultra-low voltage

standby operation. Based on the *DRV* distributions, the minimum SRAM standby V_{DD} can be determined by adding 100mV electrical noise margin to the highest *DRV*. Compared to the standard standby at 1V V_{DD} (A), the leakage power can be reduced by 75% at 650mV (B) V_{DD} of un-optimized SRAM design. The optimized SRAM achieves a standby V_{DD} of 320mV (C), and consumes 2.8% of the original leakage power. Finally, the optimized memory with ECC lowers the standby V_{DD} to 255mV (D), and reduces the leakage power by another 35%.



a) Measured SRAM leakage currents

b) SRAM leakage power minimization

Figure 5.6. Measured SRAM leakage power savings

In Figure 5.7, the power savings for different test-chips are plotted. While the maximum possible power saving using optimal bounded-distance ECC ranges from 24% to 53%, the power saving achieved by the Hamming code implemented in design is from 12% to 48%, and closely tracks the optimum savings. On average, the Hamming code achieves 76% of the optimum code power saving.

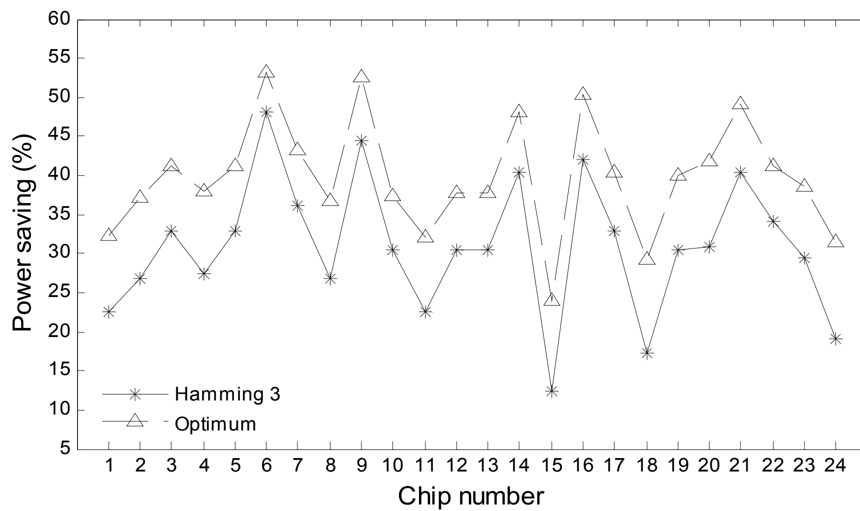


Figure 5.7. Leakage power savings with error-tolerant SRAM design

5.2.3 SER improvement

Although the ECC design in this work was introduced for power-optimization, it also improves the memory reliability towards soft errors as in the traditional memory redundancy schemes. Based on the soft error rate (SER) analysis method elaborated in [32], the SER improvement of a (31, 26, 3) Hamming ECC is shown in Figure 5.8. This theoretical analysis assumes a random occurrence of soft errors. The SRAM SER after the Hamming ECC correction is calculated given each value of SER without ECC.

In an error-tolerant SRAM designed for ultra-low standby power, the data-retention errors cause a slightly reduced effectiveness of soft error protection. For example, if the ECC needs to correct data-retention errors in 10% of the total SRAM code words, only 90% of the SRAM array is protected by ECC towards the soft errors. It is an interesting research topic to analyze how to design a memory ECC to optimize the soft error protection during an error-tolerant low-voltage standby. That is beyond the scope of this dissertation work but may be a future project in low-voltage SRAM design.

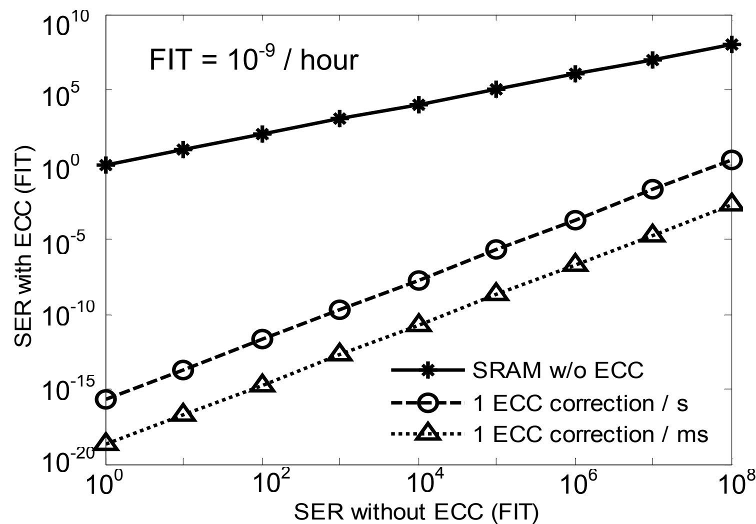


Figure 5.8. SER improvement with a (31, 26, 3) Hamming ECC

6 Conclusion

In order to maximize the SRAM leakage power saving with reliable data retention, this work explores the limit of SRAM data preservation under ultra-low standby V_{DD}. An analytical model of the SRAM *DRV* is developed and verified with measurement results from a 130nm SRAM test chip. An industrial IP SRAM module with high-V_{th} process is shown to be capable of sub-400mV standby data preservation. With additional 100mV guard band to account for power supply noises, leakage power saving of more about 85% can be achieved under a 490mV standby V_{DD}, compared to the typical standby power at 1V V_{DD}.

The *DRV* is observed to be a strong function of process variation and also SRAM cell design parameters. Therefore a *DRV*-aware SRAM cell optimization methodology was developed to further reduce *DRV* and improve the leakage power saving. It was shown that the body bias and device channel length parameters are the most effective design knobs in minimizing *DRV*. By using a standby-mode reverse body biasing and larger channel length, *DRV* and the SRAM cell leakage power can be significantly reduced. A *DRV* design model was developed based on the experiment data from a 90nm SRAM test chip. With these circuit optimizations, a leakage power saving of 97% under 270mV standby V_{DD} was demonstrated for a memory size of 4K cells.

On architecture level, an aggressive SRAM standby V_{DD} reduction scheme with an error-tolerant design was proposed and analyzed. By modeling and optimizing the SRAM leakage power as a function of the ECC parameter, *DRV* distribution and the standby V_{DD},

we established fundamental bound for the reduction of standby power based on the ECC theory. A practical error-tolerant SRAM design using a (31, 26, 3) Hamming ECC was proposed with a predicted power reduction of 33%, compared to the conventional design using a standby V_{DD} higher than the worst-case DRV among all SRAM cells.

By integrating the circuit optimization and error-tolerant architecture in a 90nm 26kb ultra-low leakage SRAM chip, it has been shown that the SRAM data can be reliably retained at a 255mV standby V_{DD}, with a 50X leakage power reduction. While the optimization techniques also improve active SRAM operation, the tradeoff is a 50% larger area. Additionally, the error correction scheme improves the memory soft error resilience.

References

[1] F. Ricci, L. T. Clark, T. Beatty, W. Yu, A. Bashmakov, S. Demmons, E. Fox, J. Miller, M. Biyani, J. Haigh, "A 1.5 GHz 90 nm embedded microprocessor core," *Digest of Technical Papers. Symposium on VLSI Circuits,* pp. 12-15, June 2005.

[2] S. Naffziger, B. Stackhouse, T. Grutkowski, D. Josephson, J. Desai, E. Alon, M. Horowitz, "The implementation of a 2-core, multi-threaded itanium family processor," *Journal of Solid-State Circuits,* vol. 41, issue 1, pp. 197-209, Jan. 2006.

[3] M. Sheets, F. Burghardt, T. Karalar, J. Ammer, Y. Chee, J. Rabaey, "A Power-Managed Protocol Processor for Wireless Sensor Networks," *Digest of Technical Papers. Symposium on VLSI Circuits,* pp. 212-213, June, 2006.

[4] S. Borkar, "Design challenges of technology scaling," *IEEE Micro*, vol. 19, issue 4, pp. 23-29, Jul-Aug 1999.

[5] S. Manne, A. Klauser, and D. Grunwald, "Pipeline Gating: Speculation Control for Energy Reduction, " *International Symposium on Computer Architecture*, pp. 132-141, July 1998.

[6] M. Horiguchi, T. Sakata, and K. Itoh, "Switched-source-impedance CMOS circuit for low standby subthreshold current giga-scale LSI's," *IEEE Journal of Solid-State Circuits*, vol. 28, issue 11, pp. 1131-1135, Nov. 1993.

[7] B. H. Calhoun, A. Chandrakasan, "A 256kb sub-threshold SRAM in 65nm CMOS," *IEEE International Solid-State Circuits Conference*, pp. 628, Feb 2005.

[8] Z. Guo, S. Balasubramanian, R. Zlatanovici, T. J. King, B. Nikolic, "FinFET-based SRAM design," *International Symposium on Low Power Electronics and Design*, pp. 2-7, Aug. 2005.

[9] H. Mizuno and T. Nagano, "Driving source-line (DSL) cell architecture for sub-1-V High-speed low-power applications," *Digest of Technical Papers. Symposium on VLSI Circuits*, pp. 25–26, June 1995.

[10] K. Itoh, A. R. Fridi, A. Bellaouar, and M. I. Elmasry, "A deep sub-V, single power-supply. SRAM cell with multi-Vt, boosted storage node and dynamic load," *Digest of Technical Papers. Symposium on VLSI Circuits,* pp. 132–133, June 1996.

[11] H. Kawaguchi, Y. Iataka, and T. Sakurai, "Dynamic Leakage Cut-off Scheme for Low-Voltage SRAM's," *Digest of Technical Papers, Symposium on VLSI Circuits*, pp.140-141, June 1998.

[12] A. Keshavarzi, S. Ma, S. Narendra, B. Bloechel, K. Mistry, T. Ghani, S. Borkar, and V. De, "Effectiveness of Reverse Body Bias for Leakage Control in Scaled Dual Vt CMOS ICs," *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED)*, Huntington Beach, CA, August 2001, pp. 207–212.

[13] S. Kaxiras, Z. Hu, and M. Martonosi, "Cache decay: Exploiting generational behavior to reduce cache leakage power," *International Symposium on Computer Architecture*, pp. 240-251, Jun-Jul 2001.

[14] K. Flautner et al, "Drowsy caches: simple techniques for reducing leakage power," *International Symposium on Computer Architecture*, pp. 148-157, May 2002.

[15] K. Zhang, et al., "SRAM design on 65-nm CMOS technology with dynamic sleep transistor for leakage reduction," *IEEE Journal of Solid-State Circuits*, vol. 40, issue 4, pp. 895-901, April 2005.

[16] C. H. Kim, J. Kim, I. Chang, and K. Roy, "PVT-Aware leakage reduction for on-die caches with improved read stability", *IEEE Journal of Solid-State Circuits*, vol. 41, no. 1, pp. 170-178, Jan. 2006.

[17] M. Khellah, D. Somasekhar, Y. Ye, N. S. Kim, J. Howard, G. Ruhl, M. Sunna, J. Tschanz, N. Borkar, F. Hamzaoglu, G. Pandya, A. Farhang, K. Zhang, V. De, "A 256-Kb Dual-V_{CC} SRAM Building Block in 65-nm CMOS Process With Actively Clamped Sleep Transistor," *IEEE Journal of Solid-State Circuits*, vol. 42, issue 1, pp. 233-242, Jan. 2007.

[18] K. Itoh, "Low Voltage Memories for Power-Aware Systems," *International Symposium on Low Power Electronics and Design*, pp. 1-6, Aug. 2002.

[19] M. Agostinelli et al., "Erratic fluctuations of SRAM cache vmin at the 90nm process technology node," *IEEE International Electron Devices Meeting*, pp. 655–658, Dec 2005.

[20] K. M. Cao et al., "BSIM4 gate leakage model including sourcedrain partition," *IEEE International Electron Devices Meeting*, pp. 815–818, Dec 2000.

[21] E. Seevinck, F. J. List, and J. Lohstroh, "Static-noise margin analysis of MOS SRAM cells," *IEEE Journal of Solid-State Circuits*, vol. SC-22, No. 5, pp. 748-754, Oct. 1987.

[22] J. Rabaey, A. Chandrakasan, and B. Nikolic, "Digital Integrated Circuits: A Design Perspective," 2nd edition, Prentice-Hall 2002.

[23] Bin Yu; Wen-Chin Lee; Chenming Hu, "Modeling Short-channel Effects Of Cmosfet's Taking Account For Channel-engineering, Defect-enhanced-diffusion And Gate-depletion," *VLSI Technology, Systems, and Applications*, *1997 International Symposium on,* June 3-5, 1997 Page(s):298 - 302

[24] J. Lohstroh, E. Seevinck, and J.D. Groot, "Worst-Case Static Noise Margin Criteria for Logic Circuits and Their Mathematical Equivalence," *IEEE Journal of Solid-State Circuits*, vol. SC-18, no. 6, pp. 803-807, Dec 1983.

[25] C. Lage et al., "Soft error rate and stored charge requirements in advanced high-density SRAMs," *IEEE International Electron Devices Meeting*, pp. 821-824, Dec.1993.

[26] M.J.Ammer, et. al., "A Low-energy chip-set for wireless intercom," *Design Automation Conference,* Jun. 2003.

[27] K.D.T. Ngo and R. Webster, "Steady-state analysis and design of a switched-capacitor DC-DC converter," *Power Electronics Specialists Conference*, pp. 378-385, Jun-Jul 1992.

[28] http://www.eas.asu.edu/~ptm

[29] K. Takeda, H. Ikeda, Y. Hagihara, M. Nomura and H. Kobatake, "Redefinition of write margin for next-generation SRAM and write-margin monitoring circuit", *International Solid-State Circuits Conference*, pp. 630-631, Feb 2006.

[30] C. Heegard and A. E. Gamal, "On the capacity of computer memory with defects," *IEEE Transaction on Information Theory*, vol. 29, no. 5, pp. 731–739, Sep 1983.

[31] C. W. Slayman, "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations," *IEEE Transaction on Reliability*, vol. 5, no. 3, pp. 397–404, Sept 2005.

[32] T. Mano, J. Yamada, J. Inoue, and S. Nakajima, "Circuit techniques for a VLSI memory," *IEEE Journal of Solid-State Circuits*, pp. 463-469, Oct. 1983.

[33] A. Kumar, et al., "Fundamental bounds on power reduction during SRAM standby data-retention", in press, *IEEE International Symposium on Circuits and Systems,* 2007.