Design and Analysis of Low-power SRAMs

by

Mohammad Sharifkhani

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2006

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

Mohammad Sharifkhani

I understand that my thesis may be made electronically available to the public.

Mohammad Sharifkhani

Abstract

The explosive growth of battery operated devices has made low-power design a priority in recent years. Moreover, embedded SRAM units have become an important block in modern SoCs. The increasing number of transistor count in the SRAM units and the surging leakage current of the MOS transistors in the scaled technologies have made the SRAM unit a power hungry block from both dynamic and static perspectives. Owing to high bitline voltage swing during write operation, the write power consumption is dominated the dynamic power consumption. The static power consumption is mainly due to the leakage current associated with the SRAM cells distributed in the array. Moreover, as supply voltage decreases to tackle the power consumption, the data stability of the SRAM cells have become a major concern in recent years.

To reduce the write power consumption, several schemes such as row based sense amplifying cell (SAC) and hierarchical bitline sense amplification (HBLSA) have been proposed. However, these schemes impose architectural limitations on the design in terms of the number of words on a row. Beside, the effectiveness of these methods is limited to the dynamic power consumption. Conventionally, reduction of the cell supply voltage and exploiting the body effect has been suggested to reduce the cell leakage current. However, variation of the supply voltage of the cell associates with a higher dynamic power consumption and reduced cell data stability. Conventionally qualified by Static Noise Margin (SNM), the ability of the cell to retain the data is reduced under a lower supply voltage conditions.

In this thesis, we revisit the concept of data stability from the dynamic perspective. A new criteria for the data stability of the SRAM cell is defined. The new criteria suggests that the access time and non-access time (recovery time) of the cell can influence the data stability in a SRAM cell. The speed vs. stability trade-off opens new opportunities for aggressive power reduction for low-power applications. Experimental results of a test chip

implemented in a $130nm$ CMOS technology confirmed the concept and opened a ground for introduction of a new operational mode for the SRAM cells.

We introduced a new architecture; Segmented Virtual Grounding (SVGND) to reduce the dynamic and static power reduction in SRAM units at the same time. Thanks to the new concept for the data stability in SRAM cells, we introduced the new operational mode of Accessed Retention Mode (AR-Mode) to the SRAM cell. In this mode, the accessed SRAM cell can retain the data, however, it does not discharge the bitline. The new architecture outperforms the recently reported low-power schemes in terms of dynamic power consumption, thanks to the exclusive discharge of the bitline and the cell virtual ground. In addition, the architecture reduces the leakage current significantly since it uses the back body biasing in both load and drive transistors.

A 40Kb SRAM unit based on SVGND architecture is implemented in a $130nm$ CMOS technology. Experimental results exhibit a remarkable static and dynamic power reduction compared to the conventional and previously reported low-power schemes as expect from the simulation results.

Acknowledgements

I would like to take this opportunity to express my extreme gratitude to my research supervisor Professor Manoj Sachdev. At many stages in the course of this research project I benefited from his advice, particularly so when exploring new ideas. His positive outlook and confidence in my research inspired me and gave me confidence.

A project of this nature, based on both experiment and theoretical work, is only possible with the help of many people. In particular, I would like to thank Dr. Applevich my mentor in the world of nonlinear time variant systems. Also, I would like to thank Arun Bagga and Pierce Chuang for helping me with the experiments.

The many hours that I spent at the school have been very stimulating and enriching thanks to the wonderful students I have been privileged to interact with. Particularly, I would like to thank Mohammad Maymandi, Andrei Pavlov, Nitin Mohan, David Rennie and Jahinuzzaman as well as Zhinian Shu and Nelson Lam for the great discussions that we had in the lab and for useful comments that improved my work significantly. Also, I would like to thank my tea-mates Amir Bayat, Nasser Lashgarian and Hamid Mohebbi.

I also would like to thank the great people of the Iranian Quran Session community. The divine inspiration of our Friday night Quran study session gave me the confidence and power to stand up against the difficulties that I faced in the course of my studies.

In addition, I would like to thank Mrs. Wendy Boles in grad office for her great effort to get me out of the bureaucracy when I needed her. And I appreciate Mr. Phil Regier for his constant help to keep me updated with my Cadence and Unix environment and his prompt helps when I was desperately looking for help.

At last, but not the least, I would like to appreciate my wife, Zohreh, for her support and patience and my son, Mahdi, for making home a place to release the stress of my hectic days.

Dedication

To my dear wife, Zohreh.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Motivation

This chapter sets the stage for the low-power embedded SRAM design. Section 1.1 briefly describes the importance of SRAM in current SoCs. Section 1.2 presents the place of SRAM among other blocks in a battery operated wireless SoC. Section 1.3 elaborates the motivation behind this research and challenges ahead of the low-power embedded SRAM design. Section 1.4 goes over the prior arts in low-power SRAM design and talks about their limitations. Section 1.5 outlines the contribution of this thesis. Section 1.6 summarizes the chapter.

1.1 Introduction

Developments in embedded memory technology have made large Dynamic Random Access Memories (DRAMs) and Static Random Access Memories (SRAMs) commonplace in today's System on Chips(SoCs.) Tradeoffs between large and small memories have made all sizes practical, enabling SoCs to resemble board-level systems more than ever. Large embedded memories give a SoC a number of benefits such as improved bandwidth and

considerable performance that can only be achieved through the use of embedded technologies. The possibility and success of including embedded DRAM and/or large SRAM blocks in a SoC depends mainly on manufacturability.

Implementation of embedded DRAMs with a unit cell as small as a single minimum size transistor and a single trench capacitor offers substantial benefits to a standard-CMOS based SoC. However, to be effectively beneficial in terms of area and power, a DRAM unit demands a manufacturing process with high-Vt low-leakage transistors as well as trench capacitors. This requirement increases the cost of this approach and limits the application of the embedded DRAMs to specialized SoCs requiring large embedded memory and operating at relatively low to medium speed.

On the other hand, embedded SRAMs are the prominent embedded memories used in today's SoCs. SRAM's integrability with standard CMOS technology gives it an ample opportunity to become the highest area consumer of many SoCs ranging from a high performance server processor to a an HDTV video processor as shown in Figure. 1.1. Unlike DRAMs, SRAMs do not require data refreshing mechanism. This is because an SRAM cell can store the data indefinitely as long as it is powered. This feature saves the complex and the area consuming data refreshing periphery circuits and makes medium size SRAM units a feasible choice for implementation in the standard CMOS process.

1.2 SRAM Application in Wireless Communication Devices

The integrability of embedded SRAMs have made it a prominent choice for the digital signal processors (DSPs) that operate along with the over-sampling analog to digital (A/D) and digital to analog (D/A) data converters. Over-sampling data converters are the most

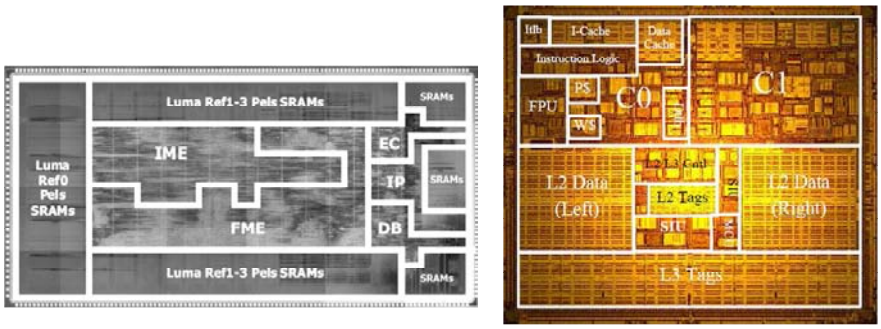


Figure 1.1 Two SoCs comprising SRAMs and cores presented in ISSCC'05: (a) A Video processor [1] and (b) a Sparc processor [2]

popular choices for the low-power applications where the accuracy of the conversion is a demanding requirement.

Amplitude and phase/frequency domain over-sampling data converters has been widely used in the wireless communication systems. These types of data converters are based on the noise shaping property of a closed loop system. Hence, the accuracy of the conversion is being traded off by sampling rate of the data converter. This feature makes the sampling frequency to be several times that of the Nyquist frequency of the signal bandwidth. In return, the accuracy of the samples is relaxed [5, 6, 7].

In the over-sampling data converters the quantization noise is shaped such that the noise is pushed out of the signal band width after its being digitized using a filter. After the signal is being digitized, it needs to be down-sampled (decimated) to the Nyquist frequency of the signal bandwidth. In order to avoid aliasing in the down-sampling process, out of band noise needs to be suppressed. This purpose is served by decimation filters.

A decimation filter is a digital filter with constant coefficients that processes the over-sampled coarsely quantized data. It provides more accurate data (i.e. higher number of

bits) since it attenuates the out of band quantization noise. Since out of band noise is weak enough after the filter, down-sampling can occur provided that the aliasing of the out of band noise does not change the signal to noise ratio in the signal bandwidth.

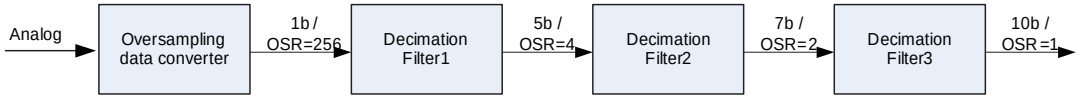


Figure 1.2 Decimation in an over-sampled data conversion

Decimation filters and down-sampling is usually done in a multi-stage fashion. For example for over-sampling ratio of 256, the down-sampling can be organized as 1/64 and 1/4 and 1/2. A decimation filter operates before each down-sampling process. Figure. 1.2 schematically depicts this idea.

In many cases, decimation filters are usually implemented as finite impulse response (FIR) filters or infinite impulse response (IIR) filters in a DSP [8]. This makes the DSP an inseparable part of an over-sampling data conversion front-ends. In many cases the DSP is designed such that it also performs other signal processing as well as coding/decoding which is demanded in a monolithic receiver. This consideration shares the resources on the chip and reduces the overall cost and power consumption.

The DSP requires a memory to store its temporary data. For example, to implement a decimation filter, each filter tap takes a certain memory location in an embedded memory. This requirement makes a memory block a key component in an integrated receiver (Figure 1.3). It is noteworthy that SRAMs are widely used in many other applications such as cache memories of multi-purpose processors.

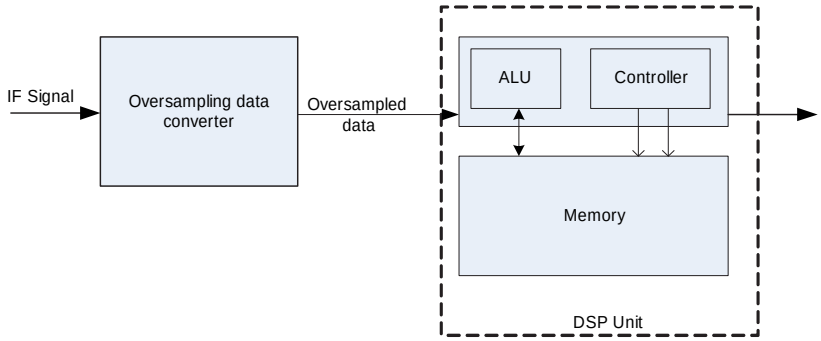


Figure 1.3 The role of memory in an over-sampled based receiver

1.3 Motivation

Over the past few years, with the explosive growth of battery operated devices such as wireless communication units, portable multi-media devices, and implantable bio-medical chips the demand for low-power integrated circuits has been significantly increased. According to International Technology Roadmap for Semiconductors (ITRS)-2003 [9], SRAM is going to take more than 60% of the SoCs in a near future. As the technology scales, the density of the transistors in the SRAM units increases substantially. Figure. 1.4 shows the trend of the transistor density according to [3]. This figure suggests that the majority of the transistors on a chip are going to sit in the SRAM unit.

As the technology scales the leakage current becomes a significant concern. According to [3], leakage current is one of the major challenges in standard CMOS SoCs:

> *"Scaling planar bulk CMOS will face significant challenges due to the high channel doping required, band-to-band tunneling across the junction and gate-induced drain leakage (GIDL), stochastic doping variations, and difficulty in adequately controlling short channel effects."*

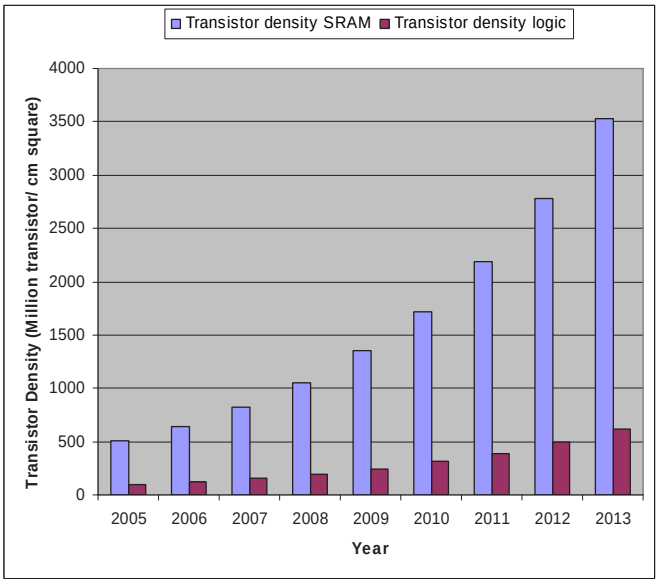


Figure 1.4 Transistor density trends: SRAM cell vs. four transistor logic with respect to year according to ITRS-2005 [3]

Of these, leakage current issue is especially important since it loses the low-power advantage of the CMOS circuits that we take for granted today. Figure. 1.5 shows the trend of the leakage current in the upcoming years predicted by ITRS. It can be seen that as the technology scales, the leakage current increases by several orders of magnitude. The modified predictions in later years portray an even higher increase. Exponentially coupled to temperature, the leakage current poses a serious threat for applications where there is a potential for high temperature operation. According to [10], the leakage current is the highest contributor to the standby power consumption of the Intel Pentium processors and there is an ongoing effort to restrain this current through device enhancement and circuit techniques. The high subthreshold leakage current has conventionally been dealt with to keep the overall leakage current within tolerable limits for high-performance chips. One common approach is to fabricate more than one type of transistor on the chip, including

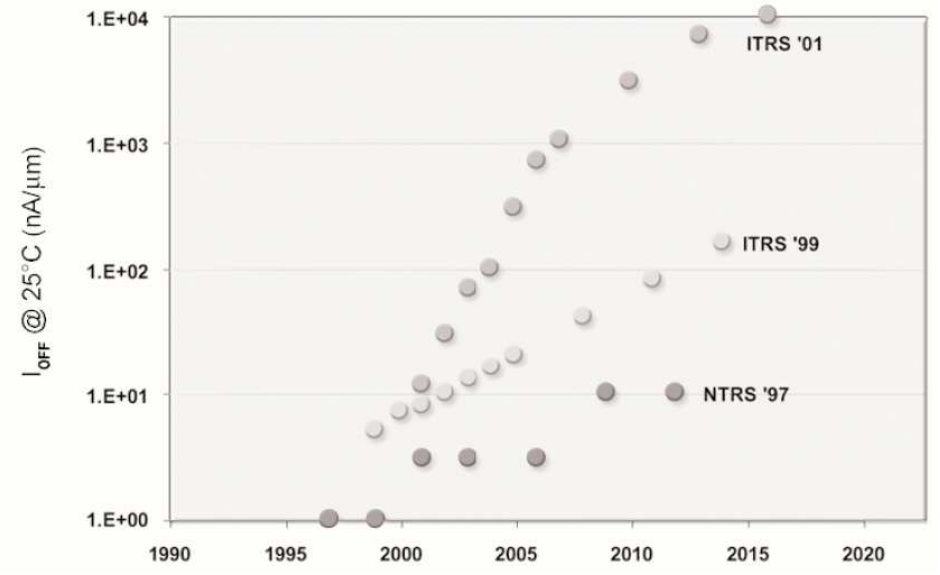


Figure 1.5 Trend of the leakage current in the standard CMOS technology according to ITRS-2005 [3]

the high-performance, low threshold voltage device described above, as well as other transistors(s) with a higher threshold voltage and larger area to reduce the leakage current. The high-performance device is used just in critical paths, and the low leakage devices are used everywhere else. This approach, however, has achieved limited success for the medium size embedded SRAM units because of the area overhead of the high-V_{th} transistors and the extra cost of the dual V_{th} process [11].

In addition to the static power consumption, the dynamic power consumption of the SRAM units is becoming an issue as the technology scales. This is particularly important on high density blocks where heavily capacitively loaded interconnects are located. Figure. 1.6 shows the trend of the minimum pitch of the Metal 1 layer as the CMOS technology scales. It can be seen that as the technology scales, the distance between the metal layers becomes shorter. Consequently, the capacitance of the interconnects increases thus influencing the

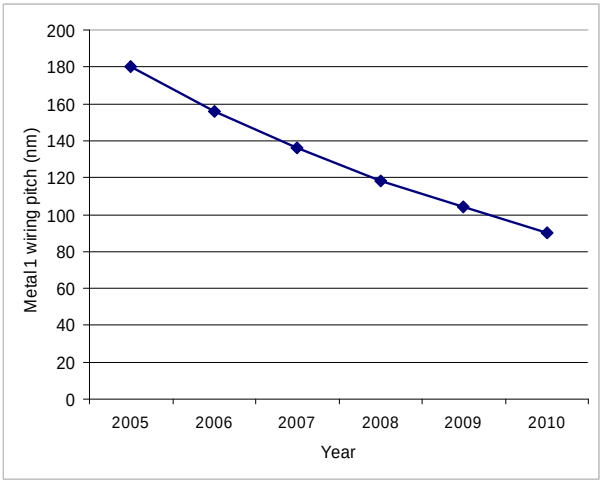


Figure 1.6 Trend of the minimum pitch for Metal 1 in the standard CMOS technology according to ITRS-2005 [3]

dynamic power consumption. This concern becomes especially important in high density blocks such as SRAM units where the interconnects are at their minimum distance from each other and are loaded with the capacitive load of a plurality of cells.

As we will see in the upcoming chapters, the reduction of the supply voltage is the most effective way in reducing both dynamic and static power consumption. Reduction of the supply voltage of the SRAM cells is known to have an adverse effect on the data stability of the SRAM cells. These cells are already under data stability problems as the technology scales. According to ITRS-2005:

> *"SRAM Difficulties with maintaining adequate noise margin and controlling key instabilities and soft error rate with scaling[are the most challenging issues for the upcoming generations of the CMOS process]."*

This assertion asks for innovative ways to address the emerging issues with regards to

embedded SRAM power consumption.

1.4 Previous Works

The demand for power reduction of the SRAM units have compelled many researchers toward innovative low-power circuits. Six transistor has been widely recognized as a suitable choice for low-power applications. Figure. 1.7 depicts a regular 6 transistor SRAM cell which holds one bit of data in an SRAM unit. It generally consists of a loop of two inverters and two access transistors. The NMOS transistors M3 and M4 are called drive transistors and are responsible for discharging the bitline during the read operation. Transistors M2 and M1 are referred to as access transistors. Once active they allow the internal nodes of the cell(*i.e.*, node A and B) to communicate with the bitlines. The gate of the access transistors is called wordline (WL). The PMOS transistors of M5 and M6 are called load transistors. Depending on the logic value stored in the cell, one of the internal nodes of the cell is at V_{DD} and the other one is at V_{SS}. The leakage current of the cell when

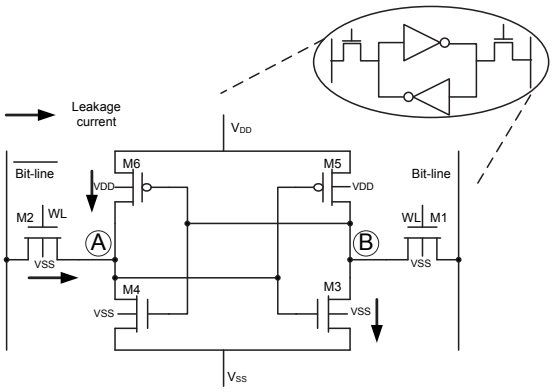


Figure 1.7 An SRAM cell

it is non-accessed is shown in the picture. The leakage current is primarily due to the subthreshold current and secondarily due to Gate Induced Drain Leakage (GIDL). In the subsequent chapters it will be shown that the leakage current has an inverse exponential relationship with the threshold voltage of the transistors.

To control the leakage, two main approaches have been suggested in recent years [12]: static and dynamic High-V_t transistors. Static high-V_t transistors have been used to reduce the leakage current where the dual V_t process is available. In this approach, there are two types of transistors. High-V_t transistors are used for drive transistors and Low-V_t transistors coupled with negative wordline voltage is used for the access transistors. The supply voltage of the cell is increased during the read operation to increase the drive of the current drive transistors. This approach requires a dual-V_t process which affects the production cost and the area of the cell [13]. On the other hand, variation of the supply voltage of the cells on the array affects the overall dynamic power consumption significantly. This effect poses a stalemate in dynamic and static power consumption trade-offs.

In the dynamic High-V_t approach, the threshold voltage of the transistors are increased by setting a negative voltage between the source and body [14]. This increase is implemented by increasing the source voltage of the drive transistors during the inactive mode. Once active, the source of the drive transistors are pulled down to the ground to establish a sufficient gate source drive voltage on the drive transistors. This variation is applied to an entire memory bank. In the upcoming chapters it will be shown that the variation of the source voltage of the drive transistors is coupled with a significant increase in the dynamic power consumption. Gate leakage reduction also become important as the technology scales and similar approach can be taken to reduce it [15].

Recently, the reduction of the dynamic power consumption of the SRAM units has received a significant attention. Particularly, the reduction of the write power consumption

of the SRAM unit is important since this operation consumes several times more energy compared to the read operation. High write energy consumption is due to the higher voltage swing on the bitlines in write operation. Recent methods and architectures for the reduction of the write power consumption will be discussed in detail in Chapter 4 after the basics of the SRAM architecture are covered in Chapter 2. In general, reduction of the voltage levels is the prominent way to reduce the write power consumption. This usually comes at the price of data stability in the cell.

1.5 Contributions and Outline of the Thesis

This thesis proposes a new architecture, which we refer to as Segmented Virtual Grounding, which addresses both dynamic and static power consumption. In addition, a new operational mode is introduced to the SRAM cell; Accessed Retention Mode (AR-Mode). By introduction of this mode, the bitlines are selectively discharged depending on weather they are selected or not. This technique reduces the dynamic power consumption.

In the proposed architecture, selective variation of the source voltage of the drive transistors breaks the deadlock between the standby power consumption and the dynamic power consumption. Since the reduction of the supply voltage of the cell affects the data stability, the concept of data stability is revisited to address data stability concerns. It is shown that dynamic data stability of the SRAM cell opens a broad opportunity for low-power SRAM design as well as design for test. Therefore, there are two main contributions for this thesis in addition to the measurement results that confirms these contributions:

- A low dynamic and static power architecture for SRAMs
- Defined the concept of dynamic data stability in SRAMs cells.

Next chapter discusses the operation of an SRAM unit. The architecture of the SRAM unit and different peripheral circuits that are involved in the operation of the unit is explained. Chapter 3 introduces the concept of dynamic data stability and its application in low-power SRAM design as well as in the test of the SRAM units. In Chapter 4 the proposed architecture of SVGND is demonstrated. A comparison is made against the recently reported low-power SRAMs. Chapter 5 explains the operation of an SRAM unit based on SVGND architecture that is implemented in a $130nm$ CMOS technology. Experimental results are also presented in the same chapter. Finally, chapter 6 concludes the thesis.

1.6 Summary

This chapter explained the importance of embedded SRAM units in current VLSI SoCs. It had been shown that the wide range of applications that need a low-power SRAM has compelled SRAM designers to come up with innovative circuits that reduce the power consumption of this unit. On the other hand, as the technology scales, low-power design becomes a more challenging task. The limitations of a number of the recently reported schemes for power reduction is identified. Finally the outline of the thesis is presented.

Chapter 2

CMOS SRAM: An overview

This chapter presents the basics of the CMOS SRAM design and operation. Section 2.1 explains the construction and operation of an SRAM cell including read and write operation. Section 2.2 discusses the conventional notion of data stability in an SRAM cell. Section 2.3 overlooks the conventional architectures of SRAMs and the peripheral blocks that are used in an SRAM unit. This section also sheds light on the timing issues. Section 2.4 demonstrates the power consumption of the SRAM unit from both static and dynamic perspective.

2.1 SRAM Cell

Memory cells are the key components of any SRAM unit. An SRAM cell can store one bit of data. An SRAM cell comprises two back-to-back connected inverters forming a latch and two access transistors. Access transistors serve for read and write access to the cell. An SRAM cell offers the following basic properties:

- **Retention**: An SRAM cell is able to retain the data indefinitely as long as it is

powered.

- **Read**: An SRAM cell is able to communicate its data.

  This operation does not affect the data *i.e.*, Read operation is non-destructive.

- **Write**: The data of an SRAM cell can be set to any binary value regardless of its original data.

A number of SRAM cell topologies have been reported in the past decade. Among these topologies, resistive load four-transistor (4T) cell, loadless 4T cell and six transistor (6T) SRAM cell have received attention in practice, owing to their symmetry in storing logic 'one' and logic 'zero'. [4]. The data retention in the 4T SRAM cells is ensured by the leakage current of the access transistors. Hence, they are not proper candidates for low-power applications. On the other hand, the data stability in a 6T SRAM cell is independent of the leakage current. Moreover, 6T configuration exhibits a significantly higher tolerance against noise which is an important benefit especially in the scaled technologies where the noise margins are shrinking. That is the main reason for the popularity of the 6T SRAM cell in low-power SRAM units instead of the 4T configurations.

As discussed in the previous chapter, a 6T SRAM cell consists of two cross-coupled CMOS inverters and two access transistors. The output (input) of the inverters construct the internal nodes of the cell. Once active, the access transistors facilitates the communication of the cell internal nodes with the input/output ports of the cell. The input/output ports of the cell are called bitlines (BL and $\overline{BL}$.) Bitlines are a shared data communications medium among the cells on the same column in an array of cells. Consequently, they have high capacitive loading. The read and write operations are conducted through the bitlines as we will see in the upcoming sections.

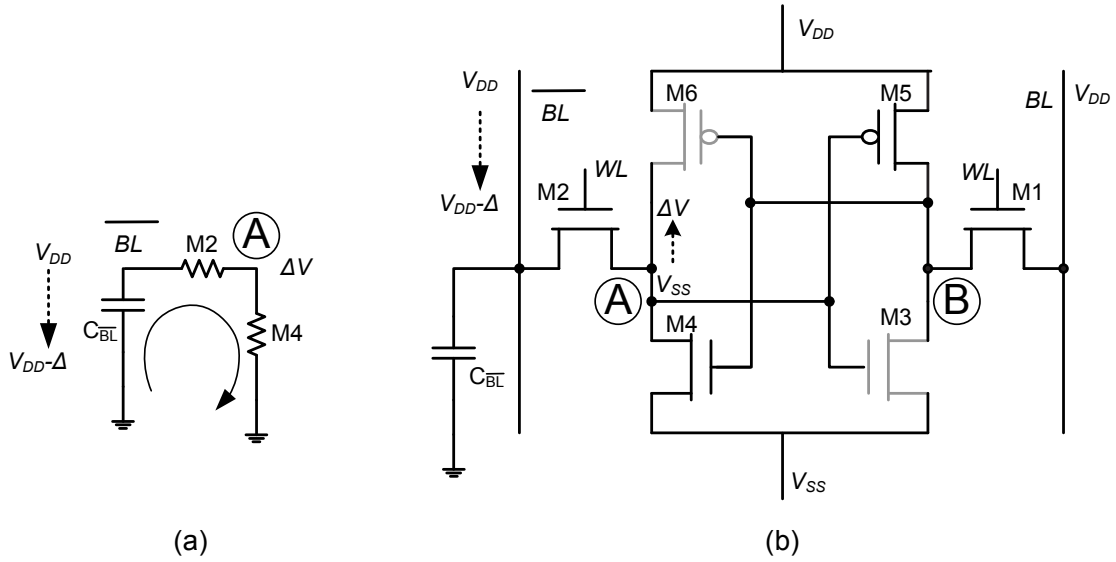


Figure 2.1 An SRAM cell during read operation:(a) linear model of transistors involved in bitline discharge (b) cell status during read operation

2.1.1 Read Operation

Figure. 2.1 (b) illustrates the operation of the cell during a read access. In this figure, node A carries a logic 'zero' and node B carries a logic 'one' before the cell is accessed. Thus, the gray transistors, M3 and M6, are 'off' while M4 and M5 are 'on' and compensate for the leakage current of M3 and M6. In conventional design, the bitlines are precharged to V_{DD} before the read operation begins.

Activation of the wordlines(WL), *i.e.*, the gate of the access transistors, initiates the read operation. As the wordlines go high, M2 goes to saturation region while M4 operates in triode region. Owing to the short-channel effect, the current associated with M2 has a linear relationship with the voltage of the node 'A' [4]. Hence, these transistors behave like a resistor in this operation. Therefore, M2 and M4 form a voltage divider and raise node 'A' voltage by ΔV. This voltage drives the input of the inverter M5-M3. To ensure

a non-destructive read operation ΔV is chosen such that it does not trigger the M5-M3 inverter and node B remains at V_{DD} over the entire cell access time. Having a constant voltage of V_{DD} at the gate of M4 warrants the constant resistivity assumption for M4 over the access time.

Figure. 2.1 (a) shows the linear model of the bitline discharge path. In this model the bitline capacitance of $C_{\overline{BL}}$ is precharged to V_{DD}. Upon the activation of M2, $C_{\overline{BL}}$ discharges through M2 and M4 and causes a voltage drop of Δ on $\overline{BL}$. Since the gate source voltage of M1 remains at zero volts(*i.e.*, $V_{gs1} = 0V$), C_{BL} can not discharge and remains at V_{DD}. The differential voltage between BL and $\overline{BL}$, Δ, is amplified using a sense amplifier to produce the regular logic levels. Clearly, a faster bitline discharge can be achieved by reducing the resistance in the discharge path. However, such improvements comes at the price of larger cell transistor sizes which is not recommended for high density SRAMs.

DC analysis of the operation of the cell transistors is conventionally adopted to ensure the stability of the cell during the read operation [4]. As it was mentioned before, a low enough ΔV ensures that the output of inverter M5-M3 remains constant at node B. To ensure a non-destructive read operation, the voltage level ΔV is controlled by the resistive ratio of M2 and M4. To assess the stability of the stored data during a read operation, cell ratio is defined as:

$$CR = \frac{W_4/L_4}{W_2/L_2} \tag{2.1}$$

where, W and L are the width and length of the corresponding MOS transistors, respectively. A higher cell ratio (a.k.a β) leads to a lower ΔV and results in a more stable read operation. The concept of data stability will be treated in the subsequent chapter in detail.

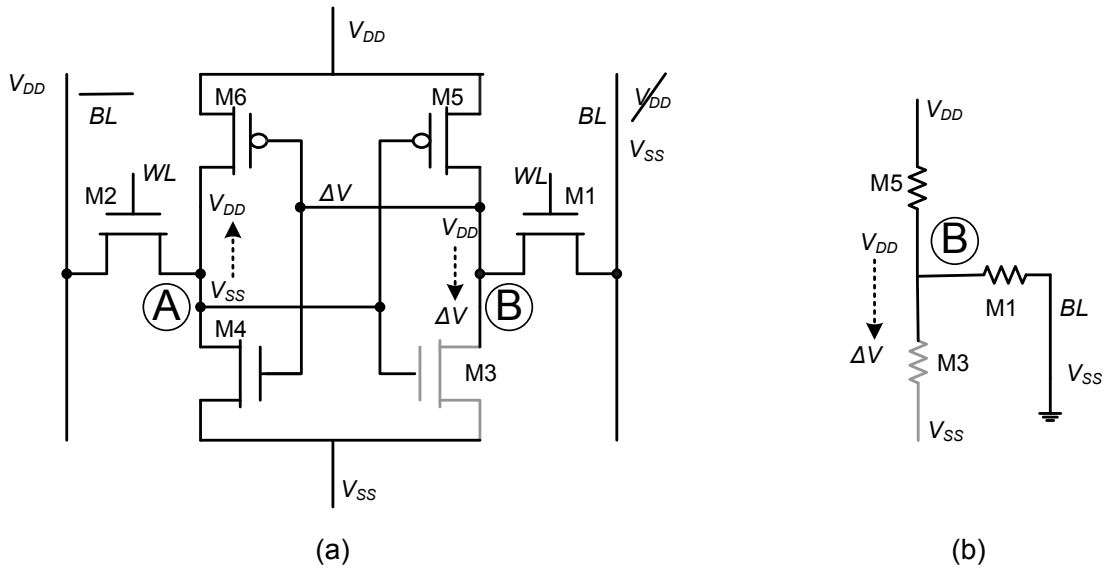


Figure 2.2 An SRAM cell during write operation:(a) linear model of transistors that initiate the write operation (b) cell status during write operation

2.1.2 Write Operation

Figure. 2.2(a) illustrates the operation of the cell in the write operation. In this figure the initial conditions of nodes A and B are V_{SS} and V_{DD}, respectively. Re-writing the old data to the cell is trivial so we concentrate on changing the data of the cell. In other words, the write operation is complete only if the voltage level on node A and B become V_{DD} and V_{SS}, respectively.

As it was mentioned in the previous subsection, for an appropriate CR, the activation of the wordline can not cause a sufficient voltage increase on node A to trigger the inverter M5-M3 if both bitlines are precharged to V_{DD}. Therefore, the write operation is conducted by reducing the bitline associated with node B, BL, to a sufficiently low voltage (*e.g.*, V_{SS}.) This operation forms a voltage divider comprising of M5 and M1 at the beginning of the operation. Pull-up ratio(PR) is defined as:

$$PR = \frac{W_5/L_5}{W_1/L_1} \tag{2.2}$$

to asses the voltage that appears at node B upon activation of the wordlines in write operation, ΔV. A sufficiently low ΔV triggers the inverter M6-M4 which results in charging up node A to V_{DD}. Since node A drives the inverter M5-M3, node B is pulled down to V_{SS} through M3 and M5 turns off. Hence, the logic state of the cell is changed. The wordline becomes inactive after the completion of the operation.

A successful write operation can be guaranteed by choosing a proper PR. A lower PR results in a lower ΔV, and a lower ΔV is associated with higher drive at the input of inverter M6-M4. In order to achieve a low PR a wider access transistor is desirable, however, increasing the width of the access transistor threatens the stability of the cell during the read operation by affecting CR. This calls for a trade-off between data-stability in the read operation and successfulness of the write operation.

It is noteworthy that for an SRAM cell, the desired type of operation can be set with the proper choice of the bitline voltage. However, this calls for additional periphery circuits such as bitline precharge circuits and write drivers to ensure proper bitline voltage setting before any operation.

2.2 SRAM Cell Static Data Stability

Static data stability of the SRAM cell has been a prominent topic in the SRAM cell design. This is because it examines the SRAM cell for its ability to perform its main operation; to retain the data. The notion of static data stability is the foundation of the realization of binary computing using electrical devices such as BJT and MOS transistors for decades. Basically, this notion links the physical voltage levels at the input and the output of a

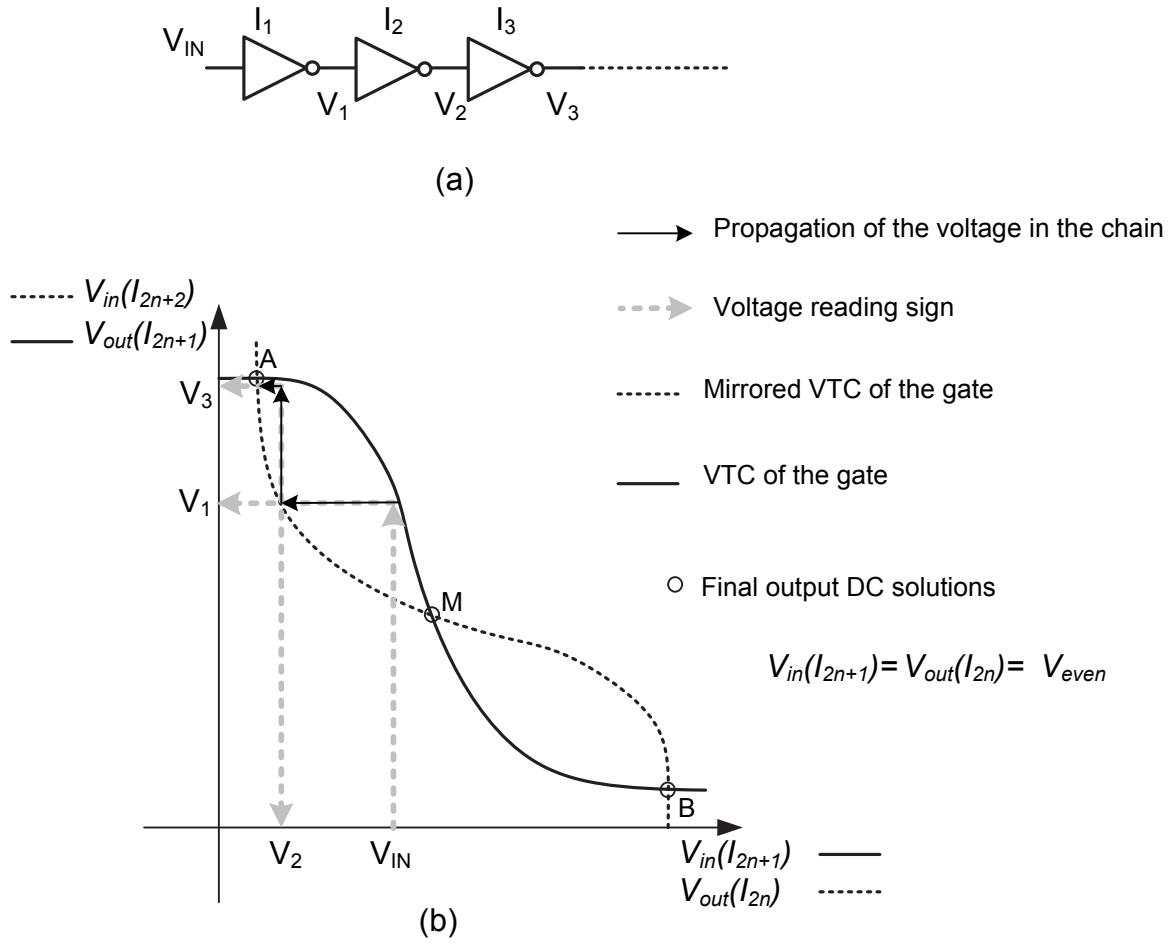


Figure 2.3 (a) Data stability in an infinitely long chain of logic gates and (b) Qualitative analysis of the gate chain behavior using VTC

gate (*e.g.*, an inverter) to the boolean logic states. A gate can offer a logic operation if for any arbitrary static input, the static output voltage of an infinitely long chain of the gate converges to one of the three unique voltage levels associated with the gate. Figure. 2.3(a) shows such a chain.

DC voltage transfer characteristic(VTC) of the gate has been widely used in verification of this criteria. The VTC of a gate is given by:

$V_{out} = h(V_{in})$

where V_{in} and V_{out} represent the DC input and output voltages of the gate. The verification of the criteria for operation as a logic component is conventionally done by drawing the gate's VTC and its mirror (See Figure. 2.3(b)). If VTC of the gate and its mirror coincide at three points(*i.e.*, A,B and M), then the output of the chain will be at V_A or V_B or V_M depending on V_{IN} [16]. The output of the chain is at V_M only if the input is exactly at the same voltage.

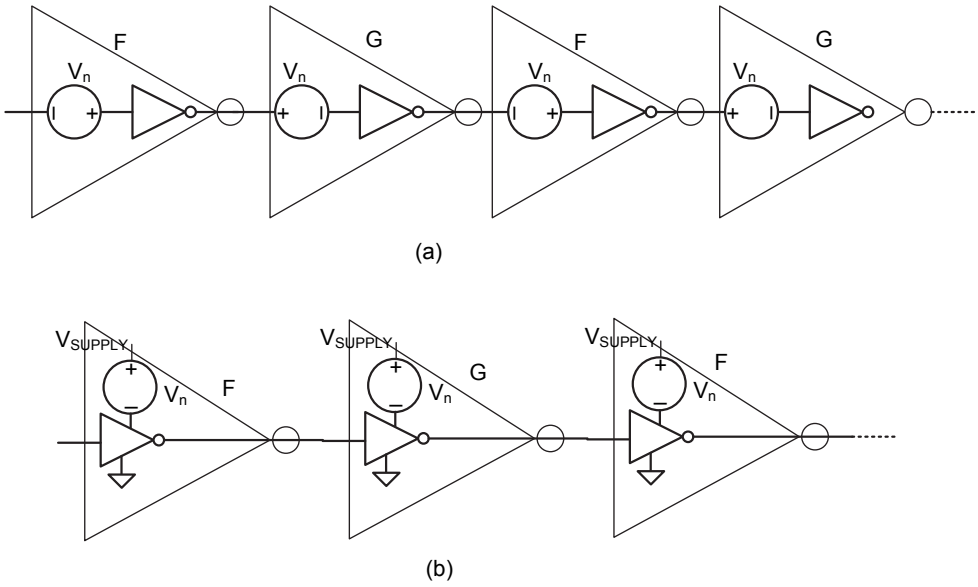


Figure 2.4 The schematic of the chain when the noise source affects the gate as a (a) series voltage source at the inputs and (b) supply voltage noise source

Worst case static noise margin has been defined as a numerical measure for the ability of creating logic states [17]. Worst case noise is defined as the DC disturbance which is adversely present in all logic gates in an infinitely long chain of gates. The noise source can be any DC noise source that affects the VTC of the gate (*e.g.*, series input voltage, supply voltage DC noise, ground voltage DC noise, etc.). Figure. 2.4(a) shows the chain

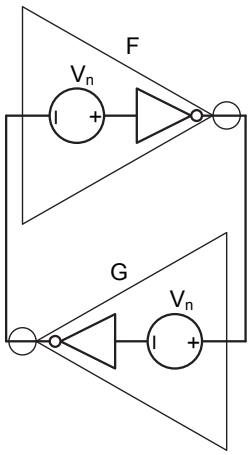


Figure 2.5 A loop can represent an infinitely long chain of gates

of gates and the static noise sources adversely connected at the inputs of the gates and Figure. 2.4(b) shows the same chain when the noise source is applied to the supply voltage of the cells. For any type of noise source there exist a worst case noise margin [18]. Moreover, it has been shown that the behavior of the infinitely long chain of the noisy gates can be investigated by analyzing a loop comprising the noisy gates of the chain [19]. Figure. 2.5 shows a circuit which is equivalent to the chain shown in Figure. 2.4 in terms of final DC operating points. This figure can also be used to explain the equivalence between the noise margin in a cross-coupled flip-flop, essentially an SRAM cell, and an infinitely long chain of logic gates as suggested in [20].

The noise margin for a noise source is defined as the amount of noise that if applied to the gates "with a little more noise the transfer characteristics have only one intersection" [18]. In other words, for any noise source the noise margin is the amount of noise that makes the cell violate the criteria of data stability: having three coincident points on VTC and its mirror. Clearly, in this definition both the noise source and the criteria have a static(*i.e.*, DC) nature.

The static noise margin for the *input series voltage noise source* has received a significant attention among different noise sources. That is because this noise source models the static circuit non-idealities such as threshold voltage variation of the MOS devices and mismatches. This type of noise margin is widely known as static noise margin (SNM). The SNM of an SRAM cell can be found using the well known "mirror and maximum square method" [20]. In this method, the VTC of the feedforward inverter and the the mirrored VTC of the feedback inverter is drawn to form a butterfly shape (See Figure. 2.6). The maximum square that can fit within the smaller wing of the butterfly curve represents the SNM of the cell. That is because a series noise source at the input of the feedforward gate moves the VTC horizontally to the left side by SNM volts and the same amount of noise at the input of the feedback inverter vertically moves up the VTC mirror by the same amount. Such movements closes the smallest wing of the butterfly and leaves only one coincident point between VTC and its mirror. Clearly, the two wings of the butterfly curev are identical if the feedback and feedforward VTCs are the same. However, if there is a mismatch between the feedback and feedforward VTC, then there is an asymmetry in the butterfly curves, making the sizes of maximum square different.

The notion of SNM has dominated the realm of SRAM cell design for forty years. Based on this measure, an SRAM cell is designed such that under the worst case operational mode (*i.e.*, read operation) there remains some noise margin [20, 4]. In the next chapter, this concept will be revisited and will be extended to another static noise margin for a dynamic stability criteria.

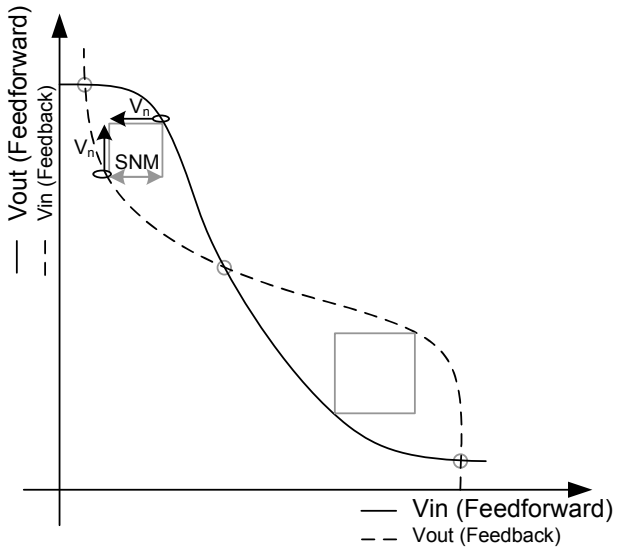


Figure 2.6 The concept of static noise margin (SNM) in an SRAM cell

2.3 Architecture of an SRAM Unit

The periphery blocks in an SRAM unit facilitate access to the cells for the read or write operation. In practice, multiple bits are accessed for the read or write operation at the same time. The group of bits that are accessed at the same time form a word. Depending on application, the word size, M, usually varies from a dozen bits to 64 bits. In regular SRAMs only one word is accessed at a time. The number of words that are accommodated in the unit specifies the length of address field, N. However, The total number of cells in an array can be calculated as $M \times N$.

An SRAM unit consists of several periphery blocks. An array accommodates the plurality of cells. A decoder decodes the binary encoded input address to indicate the physical location of the addressed cell(or word.) Sense amplifiers(SA) and write drivers interface with the bitlines to communicate with the cell in read and write operations, respectively. A timing control unit generates the proper timing signals for the activation of the wordline,

SA or write driver during the read or write operation, respectively.

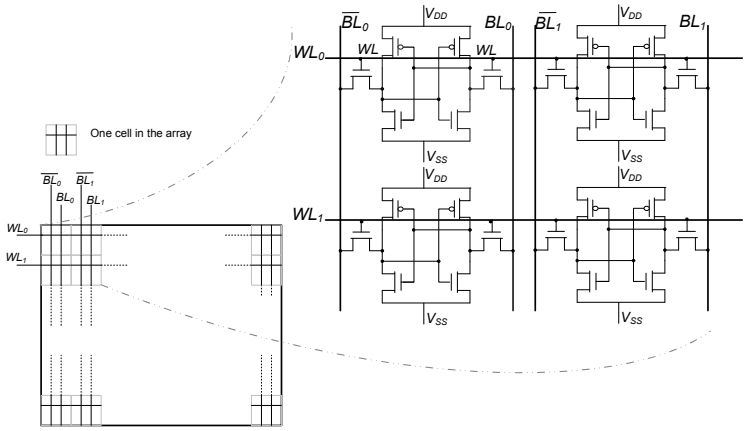


Figure 2.7 Construction of an array based on a plurality of SRAM cells

A plurality of cells organized beside each other form an array. The cells sitting on the same row share the same wordline. Cells on the same column share the same pair of bitlines. Figure. 2.7 illustrates the construction of an array and the associated bitlines and wordlines. Clearly, the numerous cells on the same bitline and the short distance between the neighboring columns impose a significant capacitive load on the bitlines. For every access, only one wordline is active in an array. Activation of the wordline causes all SRAM cells on the row to discharge their corresponding bitlines. Hence, all the bitlines are discharged as a result of wordline activation.

2.3.1 Row Decoder and Column Multiplexer

Multiple words are placed in one row in applications with usual word size($M < 128$). Different bits of the words on a row are interleaved to share periphery circuits such as SA, write driver and row decoder. Figure. 2.8 shows an array in which each row accommodates 2^n words and each word comprises M bits. The first bit of all the 2^n words on the same

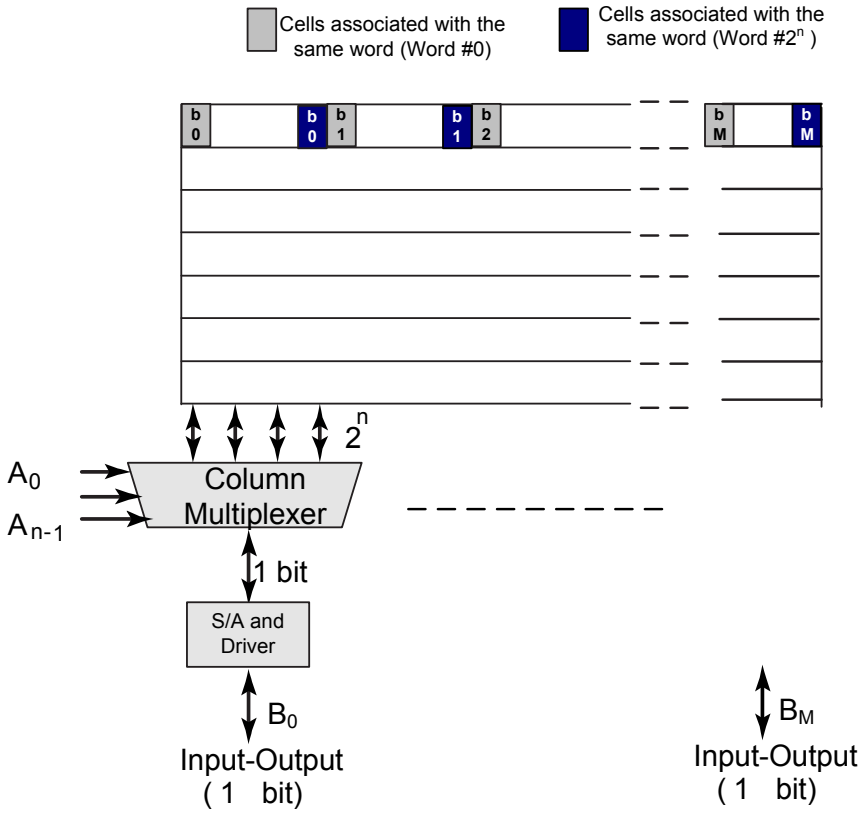


Figure 2.8 The concept of interleaving in an SRAM array

row, *i.e.*, b_0 of all 2^n words, are placed beside each other. The next bit of all words are placed at the neighboring set of cells. Hence, only one SA and write driver serves the first bit, B_0, of all the 2^n words on the same row. A column multiplexer* selects the bitline that is connected to the SA or write driver. It is clear that for a word size equal to M there are $M \times 2^n$ cells on the same row (*i.e.*, $M \times 2^n$ columns in the block).

The location of the accessed word is specified by the address inputs of the SRAM unit. Specified by $A_0 - A_{K-1}$, the address input is a binary encoded number with the bitwidth of $K = \log_2(N)$ bits. The address input is decoded to locate the word that is desired for an

*Some times the column multiplexer is referred to as column decoder.

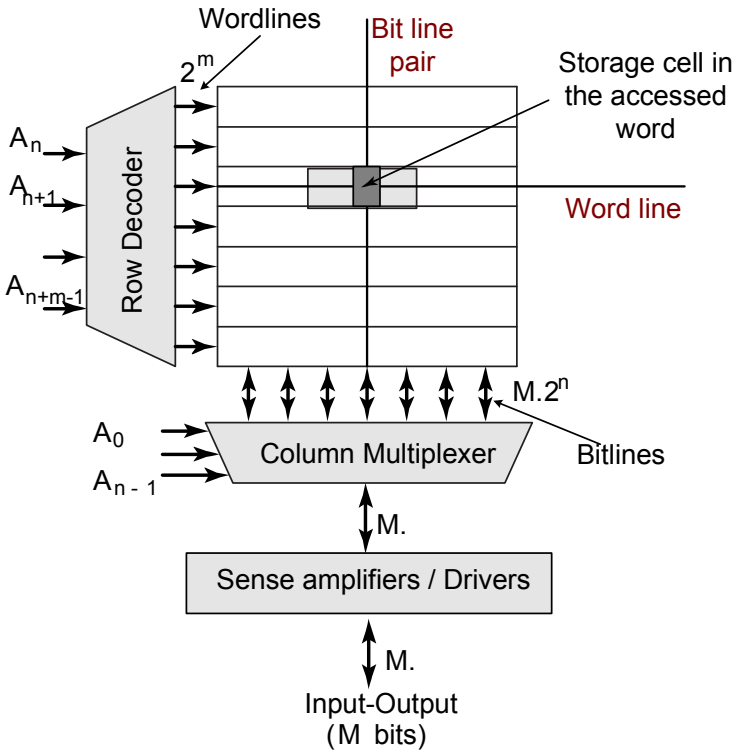


Figure 2.9 Utilization of a row decoder and a column multiplexer to activate the respective wordline and bitline according to the address

access. The word can be located by specifying the row on which the desired word is placed and the columns on which different cells corresponding to different bits of the same word are placed. Therefore, the K address bits are divided into column address bits $A_0 - A_{n-1}$ and row address bits $A_n - A_{n+m-1}$ where $K = n + m$. Figure 2.9 clarifies this technique. The column address bits drive the column multiplexer and specifies which word in the row is accessed. Driven by the row address bits, the row decoder specifies the row in which the accessed word is located by activating the wordline of that word.

A row decoder is usually based on two pre-decoders and a post-decoder as shown in Figure 2.10. It has been reported that equal division of the row address field between

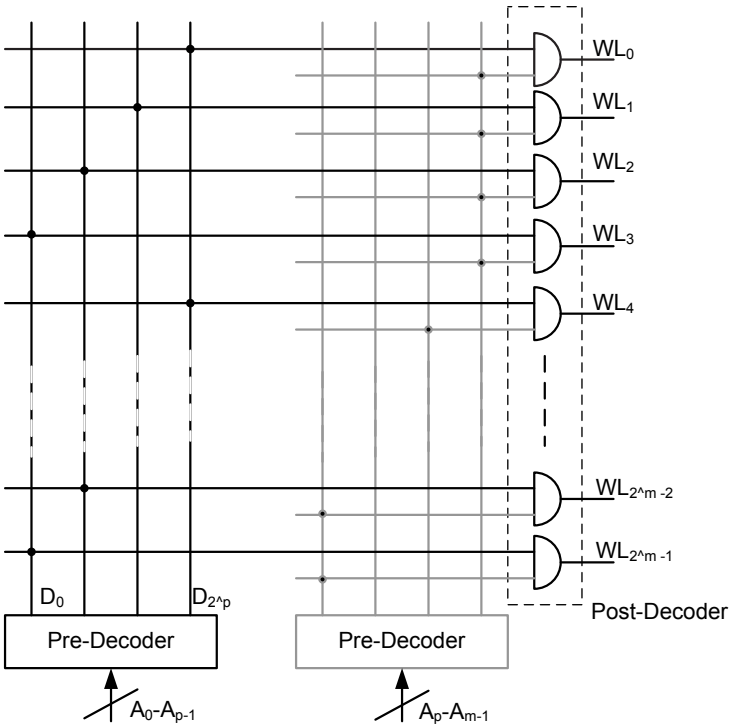


Figure 2.10 Implementation of the row decoder based on pre-decoders and a post-decoder

two equal pre-decoders exhibits shortest delay time [21]. For a given address, only one of the outputs of each pre-decoder is active. The post-decoder consists of a plurality of AND gates. The output of the AND gates drive the wordlines of each row. The AND gates are driven by the two pre-decoders such that the output of the AND gates produce all combinations of the pre-decoders' output. Basically, one input of every AND gate is driven by one of the pre-decoders whereas the other input of the AND gate is driven by the other pre-decoder. Therefore, for any address input, only one of the post-decoder's outputs will be active. It is noteworthy that, the length of an AND gate in the post-decoder that drive the wordline should match the length of the SRAM cells in the array. This limitation should be considered in the layout of such AND gates.

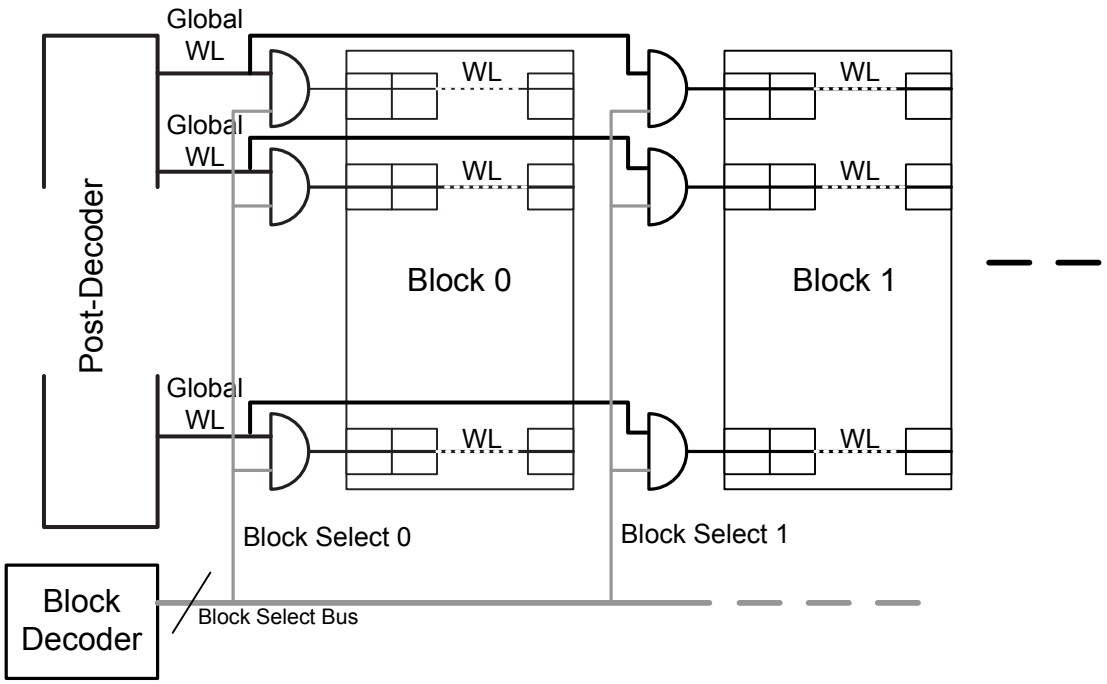


Figure 2.11 Divided wordline architecture for lower access delay and power consumption

Divided wordline configuration(DWL) has been devised for large address size SRAMs. Figure 2.11 shows this decoding scheme. In this scheme, the memory is partitioned into several blocks. The output of the row decoder is considered as a global wordline. Moreover, a local wordline effectively activates the access transistors of the cells in the selected row of the selected block. The local wordline of a row in a block is activated when both the global wordline and the block select signal are asserted. Since only one block is active at a time, DWL configuration reduces the time required to activate the wordline compared to the case where all words are placed in only one block. In addition, the power consumption is also reduced by reducing the number of cells on the same row which discharge their corresponding bitlines. However, this choice requires a block decoder which adds to the complexity and can cause an increase in the power consumption. Therefore, depending

on the size of the memory, multiple block solution can result in lower power consumption. Trade-offs and optimization of the decoders are investigated in [22].

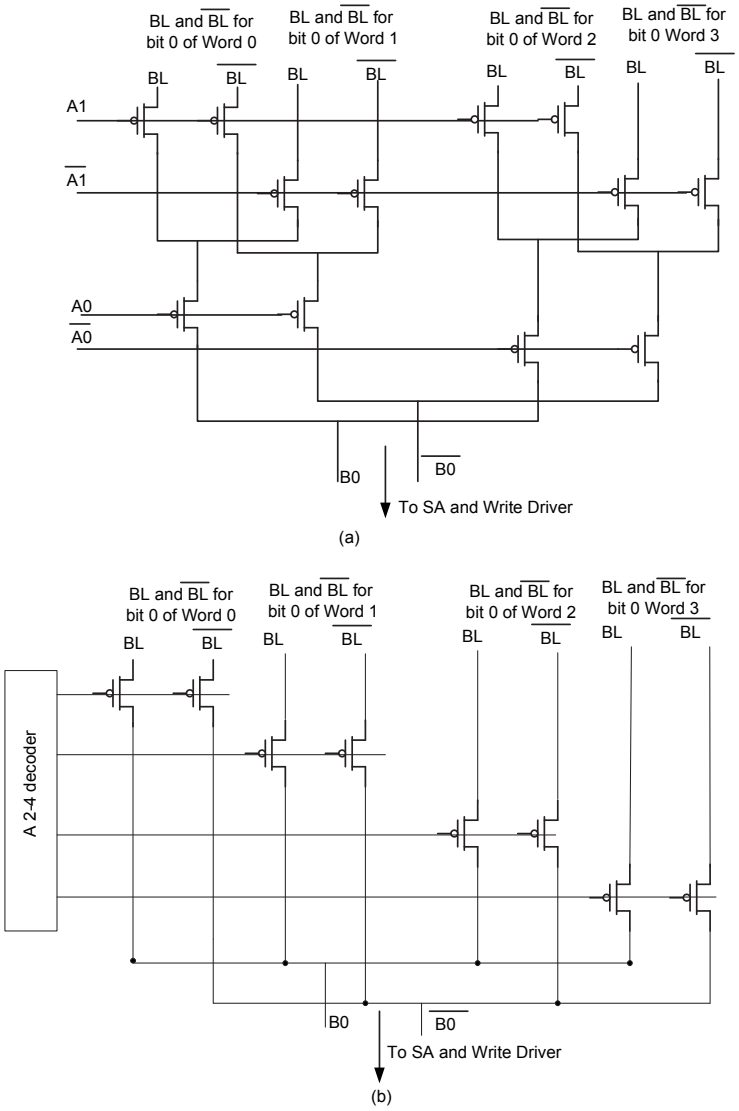


Figure 2.12 Implementation of a column multiplexer to access a single bit of the selected word

Unlike the row decoder which operates in the address decoding path, the column mul-

tiplexer(s) operate in the data path. In other words, column multiplexers direct the data associated with the desired word by selecting the corresponding bitlines. For every bit of a word, a 1 to 2^n column multiplexer is used corresponding to that bit of the word, assuming there are 2^n words on the same row(Refer to Figure. 2.8). Therefore, M such multiplexers effectively enables access to an M bit word on a row. The multiplexer can be implemented in two ways. Implemented in a differential form, a conventional tree multiplexer is shown in Figure. 2.12(a). One side of the multiplexer is connected to the BL and $\overline{BL}$ of different columns associated with the same bit of different words and the other side goes to the SA and the write driver. Since more than one transistors appear in the data path this configuration reduces the speed. The alternative scheme of using a pre-decoder and a number of pass gates is shown in Figure. 2.12(b). In this scheme, the pre-decoder activates the corresponding pair of pass gates and establishes the connection between the selected bitline (*i.e.*, selected word on the activated row) and the SA or write driver. In this case, the pre-decoder can be shared between all M multiplexers of the array. This sharing reduces the complexity of the pass gates at each bitline to only one transistor which gives rise to a higher speed and lowers the power consumption.

2.3.2 Sense Amplifier and Write Driver

Communication between an SRAM cell in the array and the outside world is conducted through a pair of bitlines. Bitlines have two important characteristics. First of all, bitlines conduct the information in a differential fashion. Second, bitlines are highly capacitive interconnects because of their being a long low metal layer and the numerous access transistors that loads them. Because of these characteristics, the voltage levels appear on the bitlines are limited to save power and delay time. A sense amplifier and a write driver interface between the pair of bitlines and the standard CMOS gates on the data I/O path

to ensure proper voltage level translation on either side.

The sense amplifier amplifies a small analog differential voltage developed on the bitlines in a read access. The amplification results in a full swing single ended digital output. Employment of SA reduces the size of the SRAM cell since the drive transistors does not need to fully discharge the bitlines. A SA needs to satisfy a few electrical requirements for proper operation in the SRAM unit. First, the required minimum differential voltage swing at the input of SA should be smaller than the minimum differential voltage that is developed over the bitlines by the SRAM cell. Second, the SA should be able to provide the output within the sense amplification time T_{sa} once it exercises minimum input differential voltage.

The area requirements for the SA depends on the SRAM array architecture. As it was mentioned before, architectures that use multiple words per row in a block can share a single sense amplifier among the multiplexed columns to fetch a single bit of a word. This sharing is possible since only one pair of bitlines associated with only one column(*i.e.*,

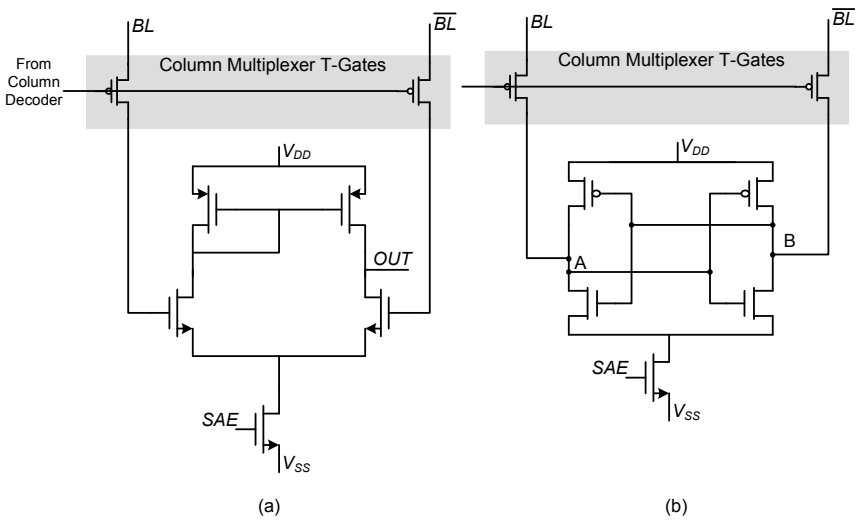


Figure 2.13 A linear sense amplifier (a) and a latch type sense amplifier (b)

one word) needs to be sensed and amplified to obtain the corresponding bit of the selected word. Hence, assuming M-bit words are placed in an array, M sense amplifiers are used for read operation. Therefore, assuming 2^n words reside on a row, the width of SA is limited by 2^n times the pitch of a column. It is clear that the pitch of a column is determined by the width of a cell.

SRAM sense amplifiers can be divided into two categories: Linear amplifiers and latch type amplifiers as shown in Figure 2.13. Here, a classical differential pair with current mirror active load can be used to amplify the small differential voltage over the bitlines. This amplifier has the benefit of rejecting the common mode voltage variation. However, it demands sufficient gate source voltage over drive $V_{gs} - V_{th}$ at its input transistor gates

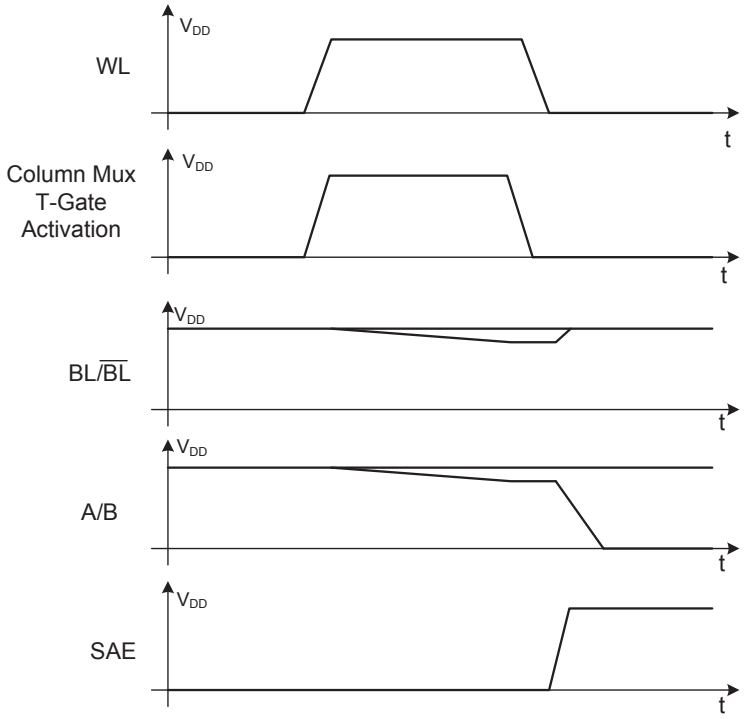


Figure 2.14 Timing in a read operation

for proper operation. On the other hand, a latch type sense amplifier is based on a cross-coupled inverter. Unlike a linear SA, a latch type SA requires an accurate timing for proper operation. This timing is shown in Figure 2.14. At the beginning, since sense amplifier enable(SAE) signal is inactive, both internal nodes of the SA are floating and precharged to V_{DD}. The bitlines are connected to the internal nodes of the SA through the multiplexer's pass gates while the cell discharges the bitline. Once sufficient differential voltage is developed over the bitlines and the internal nodes of the SA, the multiplexer's pass gates turns off to isolate the SA from the bitlines. The activation of the SAE signal concurs with the isolation of the SA from the bitline. Such a timing prevents from an unnecessary full swing discharging of the bitlines.

The initial condition of the bitlines determines the type of operation that the cell will undergo after its wordline becomes active. A precharge circuitry is employed for every column to ensure proper bitline voltage setting before each operation. The precharge circuitry sets both bitlines at the same voltage (*e.g.*, V_{DD}) before every operation. However, once the multiplexer becomes active, the precharge circuit is disabled to allow proper interaction between the bitlines and the SA or the write driver. In some cases a pair of NMOS is also used as a constant precharge load to speed up the bitline charge up especially after the write operation. Figure 2.15 shows the precharge circuitry.

Write driver has the duty of discharging the bitlines to a level below the write margin of the cell (*e.g.*, ground) quickly before or while the wordlines of the selected cell are active. Two typical write drivers are shown in Figure 2.16. The data input selects which bitline is discharged. The WE signal is turned on only when the write operation is intended. Otherwise, the WE isolates the bitlines from the write drivers. While both write drivers shown in Figure 2.16 offer the same function, Figure 2.16(b) is faster since it has less stacked transistors in its discharge path at the expense of complexity. Usually, the write

operation is not a speed limiting transaction and therefore, simpler configurations relaxing the layout requirements are preferred for the write driver.

2.3.3 Timing Control Unit

Deceptively simple in first look, yet every read and write operation in an SRAM unit relies on a detailed and properly timed procedure. This is because several blocks operate in series or parallel to perform a read or write operation. Timing control unit controls the

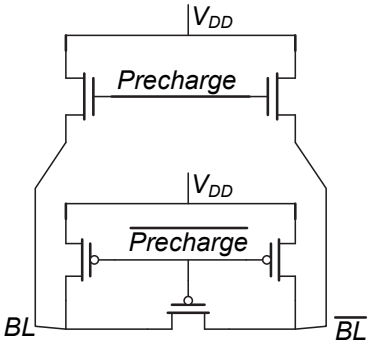


Figure 2.15 Precharge circuitry

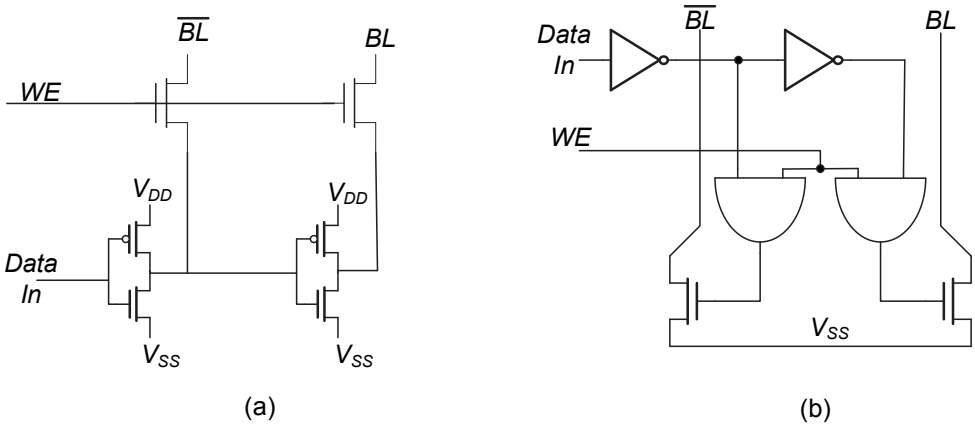


Figure 2.16 Two types of write drivers that offer write voltage of V_{ss}

procedure of these operations by providing the proper activation (and inactivation) signals for different blocks such as row-address decoders, column decoders, sense amplifiers, write drivers and precharge circuits.

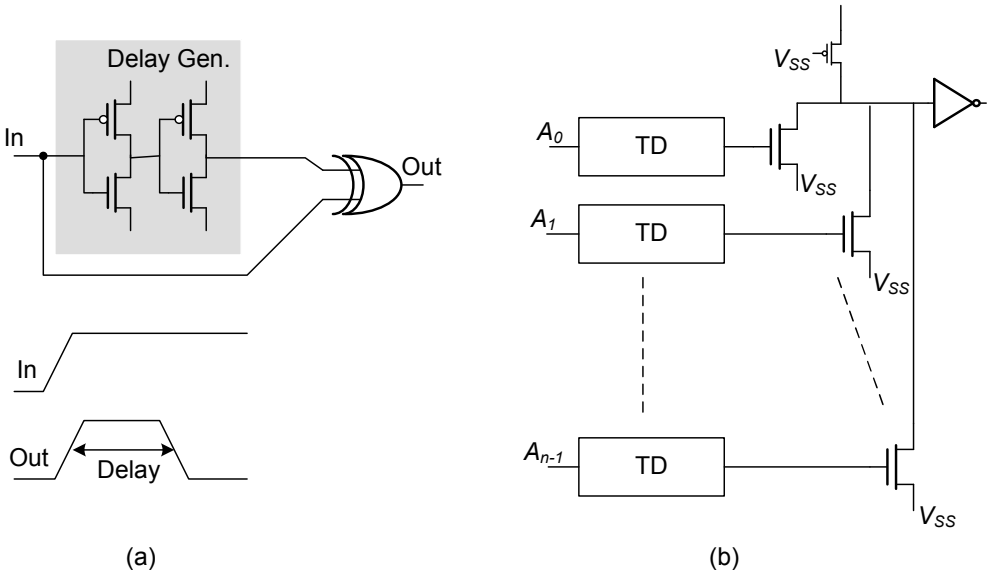


Figure 2.17 Address transition detector described in [4](a) Transition Detector (TD) for one input (b) ATD that is based on several TDs

Two types of interfacing methods have been proposed for the timing control block in the SRAM unit. A synchronous SRAM interface is mainly used for embedded SRAMs where a master clock is available. In this method, the operation of the timing block is initiated by an input clock and generates the internal signals according to the clock signal. The output of the SRAM unit is available at the edge of the next clock cycle. In other words, the time required for each SRAM operation does not exceed one clock cycle. An asynchronous SRAM interface, on the other hand, starts its operation upon detection of a change at its address inputs. It uses an Address Transition Detection circuitry (ATD) to detect such variations at its address input. Figure 2.17 shows a simplified ATD circuit. The asynchronous SRAM interface has long been used in the stand alone SRAM chips.

The internal control signals are generated once the interface circuit receives a read or write command. The generation of the internal control signals is an important and sometimes a challenging task in SRAM design. Proper generation of the internal control signals for different blocks improves the reliability, speed and power consumption of the unit. The procedure of a read or write operation consists of several phases. Figure 2.18 shows different phases in a read operation for a multi-block SRAM unit. Certain blocks or periphery circuits are activated (or inactivated) at the beginning or at the end of each phase. The timing control block estimates the actual delay time of each phase of the procedure (*e.g.*, decoding time, bitline discharge time, etc.) and issues the proper control signals for the subsequent blocks, accordingly. Basically, the timing control block is an asynchronous circuit that goes into several transitional states, stays there for a predetermined time before it goes to the next state, and ultimately arrives to the original state. It is clear that the timing control block starts its operation once the interface circuit initiates the loop operation.

The realization of the timing control unit is based on the state transition concept. Hence, a finite state machine (FSM) is used to implement the timing control unit. The FSM starts its operation upon receiving the activation signal which is issued by the interface circuitry. One possible circuit level implementation of the FSM is described in Chapter 5 where a case study of an SRAM unit is elaborated.

Owing to process variations in the scaled CMOS technology, estimation of the bitline discharge delay has received significant attention in the literature [23]. Two methods have been proposes to estimate this delay component:

- **Delay line**
- **Self-timed replica**

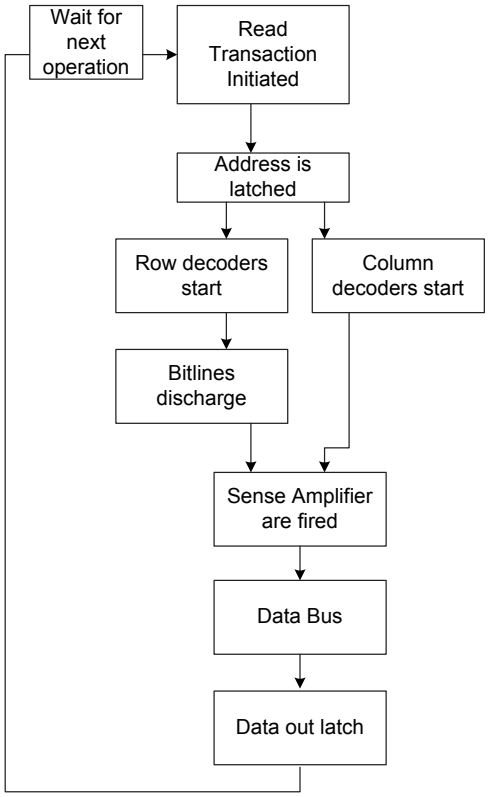


Figure 2.18 The procedure of a read operation

Figure 2.19 (a) and (b) shows the delay line and self-time replica method, respectively. In the delay line method, the bitline discharge time is mimicked in the timing loop by a series of delay elements constructing a delay line. The delay line is typically implemented by a series of inverters. Non-minimal length inverters are usually used in the delay line to realize the extended delay time associated with the bitline discharge.

Self-timed replica method offers a more accurate estimate of the bitline discharge delay in the loop and hence has dominated the SRAM designs over the past few years. A replica (dummy) column, similar to a real column, along with its sense amplifier(dummy sense amplifier) is used to generate the delay. The replica column mimics the capacitive load of

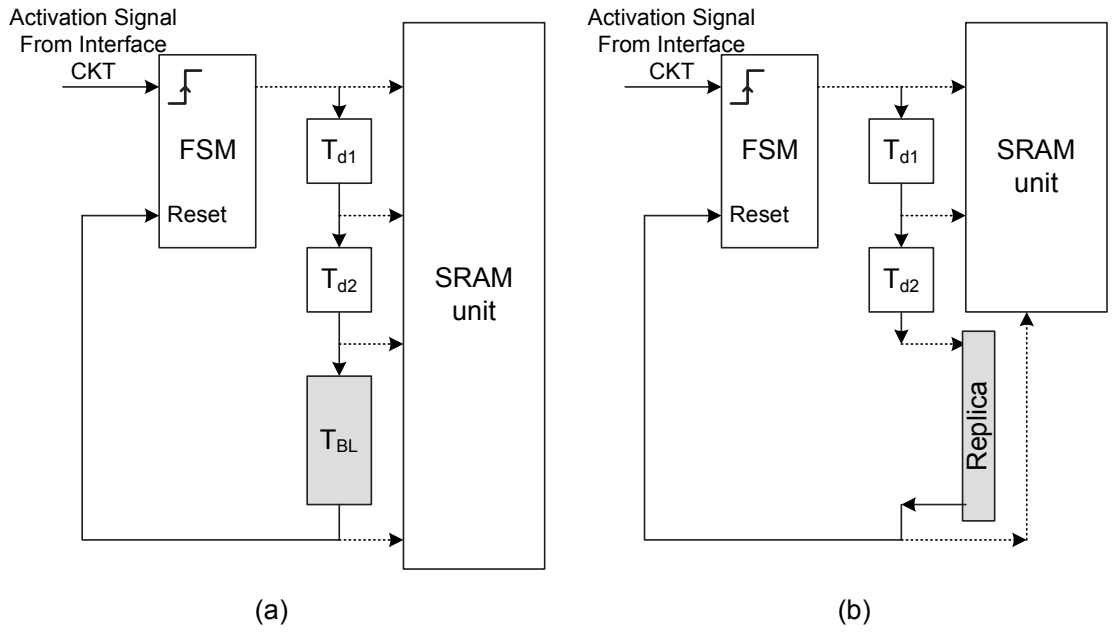


Figure 2.19 The timing loop using an FSM in an SRAM unit based on (a) delay line and (b) replica column

the real data path on the bitlines. Referred to as the discharging cell, a cell that is stuck at a known logic state is used in the replica column. The dummy sense amplifier does not have a sense enable signal and is only sensitive to the voltage that appears on the dummy bitline. In this method, the wordline associated with both discharging cell and the accessed cell become active at the same time. Once dummy bitline is discharged to the threshold voltage of the dummy sense amplifier, the signal for the deactivation of the wordline and activation of the SA is generated. It is clear that the threshold voltage of the dummy sense amplifier is designed to match the time required for the slowest SRAM cell to sufficiently discharge its bitline for proper sensing and amplification by the SA. Thus, the discharge of bitline stops at a predetermined time to prevent the excessive discharge of the bitlines. As we will see in Chapter 5, the power consumption overhead of the dummy bitline can be alleviated by modifying the replica column.

2.4 Power consumption in SRAMs

Identification of different components that contribute to the power consumption of an SRAM unit is critical to minimize the overall power consumption. The power consumption of the SRAM unit can be divided into two major terms; static and dynamic.

$$P_{Total} = P_{Static} + P_{Dynamic} \tag{2.3}$$

Static power consumption is the amount of power that is consumed by the unit to retain the data. Unlike many passive memory devices, an SRAM cell needs to be powered to keep the data. Although the amount of power that a cell consumes to retain the data is relatively small, when a plurality of cells are implemented, the total static power consumption can become significant. Static power consumption can be a major source of power consumption especially in large, low frequency SRAMs as well as SRAMs in scaled down technologies. The static power is also referred to as leakage power in digital circuit design since the static power consumption is due to the leakage current passing through the circuit when there is no activity:

$$P_{Static} = P_{Leakage} = I_{Leakage} \times V_{dd} \tag{2.4}$$

Dynamic power consumption of the SRAM unit is especially important when the speed of operation is high. The long interconnects with high capacitive loading require a significant amount of charge for their voltage variation. Owing to the interconnect's regular pattern and predictable switching activity factor, α, the power consumption associated with the interconnects that undergo a full swing voltage variation can be accurately calculated using the famous dynamic power consumption equation [4]:

$$P_{Dynamic} = \alpha f C_{interconnect} V_{dd}^2 \tag{2.5}$$

where f is the frequency of operation, $C_{interconnect}$ is the interconnect capacitance and V_{dd} is the supply voltage.

2.4.1 Static power consumption

The static power consumption in an SRAM unit is mainly due to the leakage current in SRAM cells. If several blocks are used, then the leakage current of the AND gates in the local row-decoders can become significant. However, for sufficiently large row sizes(*e.g.*, more than 32-bits) the leakage current of the post decoders can be neglected compared to the leakage current of the SRAM cells.

According to [12], among different leakage current mechanisms, the subthreshold leakage of the off transistors are the prominent source of the leakage current in an SRAM cell. The next important leakage current in the SRAM cells for currently available technologies is the gate induces drain leakage (GIDL) which is usually more than one order of magnitude smaller than the subthreshold leakage in the $130nm$ CMOS technology.

Figure 2.20 shows different components of the leakage current in an SRAM cell when

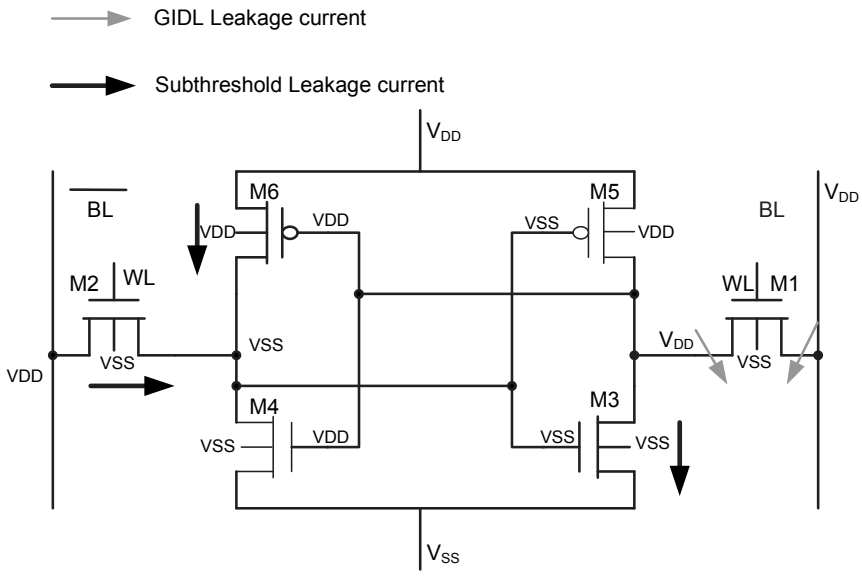


Figure 2.20 Leakage currents in a non-accessed cell

the cell is in the non-accessed mode (*i.e.*, access transistors are off.) The leakage current associated with the transistors that are off (*i.e.*, M1, M2, M3 and M6) constitutes the total leakage current. The subthreshold leakage current for any transistor can be expressed as:

$$I_s = I_0.e^{(V_{gs}-V_{th})/nV_T}(1 - e^{-V_{ds}/V_T}) \tag{2.6}$$

where $V_T = kT/q$ and $I_0 = \mu_0 C_{ox}(W_{eff}/L_{eff})V_T^2 e^{1.8}$. The gate source voltage, V_{gs}, of the transistors M3 and M6 are equal to zero because of the nature of the cross coupled loop. However, the subthreshold current of the access transistor M2 depends on the wordline voltage. Owing to a negative V_{gs}, the subthreshold leakage current of the transistor M1 is substantially smaller compared to the GIDL associated with that transistor. However, as long as the gate voltage does not go below the substrate voltage, the GIDL current associated with M1 is negligible in comparison to the subthreshold leakage currents of the other off transistors.

2.4.2 Dynamic power consumption

The read and write dynamic power consumption of an SRAM unit can be calculated by adding up the dynamic power consumption associated with different capacitive loads that is charged and discharged during the read and write operation, respectively. Clearly, the total dynamic power consumption is mainly dominated by the long interconnects which impose a large capacitive load to the signal paths in the SRAM unit. For example, the outputs of the pre-decoders at the row decoder which are loaded by the post decoder, block decoder's output, the wordlines and bitlines and the column decoder's output are considered as heavily loaded interconnects.

To simplify the calculations, the energy consumption associated with different nodes can be calculated, separately. Let's assume that a block comprises of R rows and $L = M \times 2^n$

columns, where M is the word size and 2^n is the number of words on the same row. Then capacitance associated with the bitlines, wordlines, row pre-decoders, column decoders and block decoders can be calculated as:

$$C_{BL} = R \times C_{BLcell} \tag{2.7}$$

$$C_{WL} = L \times C_{WLcell} \tag{2.8}$$

$$C_{RDEC} = R \times C_{VMetal} + P \times C_{AND} \tag{2.9}$$

$$C_{CDEC} = L \times C_{HMetal} + M \times (C_{MUX} + C_{Pre}) \tag{2.10}$$

$$C_{BDEC} = R \times (C_{AND} + C_{VMetal}), \tag{2.11}$$

respectively. Considering the figures shown in 2.3 the variables in the preceding equations are defined as:

- C_{BLcell}: Bitline capacitance associated with one cell
- C_{WLcell}: Wordline capacitance associated with one cell
- C_{VMetal}: Unit metal capacitance associated with the interconnecting wire at the output of the pre-decoder or block decoder that vertically runs in parallel to the block to feed the post-decoder
- C_{HMetal}: Unit metal capacitance associated with the interconnecting wire at the output of the column decoder that horizontally runs in parallel to the block to feed the multiplexer's pass gates and precharge circuits
- C_{AND}: Input capacitance of the P AND gates of the post decoder that are driven by the row predecoder

- C_{MUX}: Input capacitance of the column multiplexers
- C_{Pre}: Input capacitance of the column precharge circuitry.

The dynamic energy consumption associated with each capacitor during the read and write operation can be calculated provided that the voltage variation of these capacitors is known. Except for C_{BL} which has a different voltage variation during the read and write operation, the rest of the components have a similar energy consumption of:

$$E_i = C_i \times V_{DD}^2, \tag{2.12}$$

where C_i is one of the capacitors defined in equations 2.8- 2.11. For bitline energy consumption equation 2.12 is modified as:

$$E_{BL} = C_{BL} \times V_{discharge} \times V_{DD}, \tag{2.13}$$

where $V_{discharge}$ represents the amount of voltage variation of the bitlines in read or write operation. This value is about $100mV - 200mV$ in the read operation and close to V_{DD} in the write operation. Owing to the long, low-pitch interconnects and the numerous junction capacitors of the access transistors that are connected to the bitlines, C_{BL} is the largest capacitor among the previously mentioned capacitors that are mentioned. Moreover, the bitline voltage variation of V_{DD} during the write operation constitutes a prominent fraction of the dynamic energy consumed by the SRAM unit. This fact has compelled many designers to combat the write power consumption by reducing the bitline capacitance [24] or by reducing the bitline discharge voltage [25] in recent years.

2.5 Summary

This chapter discussed the conventional SRAM unit design and operation. The operation of an SRAM cell and design consideration of the cell was explained. The design consideration

includes the trade offs between speed and data stability in an SRAM cell. Conventional perception of data stability was presented. It was shown that the definition of SNM is based on static noise sources and static stability criteria.

The conventional architecture of an SRAM unit was explained. The implementation of different periphery blocks including row and column decoders, sense amplifiers and write drivers as well as timing block was discussed. The sources for power consumption in an SRAM unit was explained. Static and dynamic power consumption in an SRAM unit was detailed.

Chapter 3

SRAM Cell Data Stability: A Dynamic Perspective

This chapter discusses the concept of data stability in the SRAM cell from dynamic perspective. Section 3.1 gives an introduction about data stability. In Section 3.2, prior arts in assessing the data stability of the SRAM cell is explained. In Section 3.3, the SRAM cell is analyzed as a periodic time variant non-linear system. Section 3.4 describes the simulation method that verifies the dynamic d-stability of a cell. Section 3.5 and Section 3.6, is devoted to the application of the new concept in low-power SRAM design and test, respectively. The presented silicon results validate the theory.

3.1 Introduction

Six transistor static random access memory (SRAM) cell is one of the most widely used circuit elements in the current system-on-chip (SoC) solutions. Its integrability with CMOS

digital circuits, high speed, and robustness have made it the most popular choice in on chip memory implementations. The SRAM cell also has excellent operational features. Energized SRAM cell is able to hold two separate logic states indefinitely. It can communicate the logic state that it holds during the read operation. The read operation is non-destructive in nature. In addition, when a cell is accessed, for the same wordline voltage, the read or write operation is determined by the bitline voltage. The last property allows multiple words to be placed in a row which results in area and power efficient array designs.

However, the most important property of an SRAM cell is its ability to hold data under varying conditions. This property is commonly referred to as the *stability* of the cell. In this chapter we refer to this property as data stability (or d-stability in short) to avoid confusion with the concept of stability from its control point of view. Traditional approaches of data stability of SRAM cell are functions of static parameters such as supply voltage, threshold voltages, and static noise sources, etc. This chapter investigates the concept of SRAM cell data stability from the dynamic perspective where it is shown that the SRAM cell data stability is a function of static as well as timing parameters.

It will be shown that, the conventional static data stability criteria of holding two separate logic static states is expandable to two periodic solutions for a periodically accessed cell. These periodic solutions must remain in the region of convergence of each of the logic stable points of the non-accessed SRAM cell. The proposed definition of the dynamic data stability criteria introduces a new bound for the cell parameter variations and revises the notion of static noise margin (SNM). A simulation method for verification of the dynamic data stability criteria is presented. In addition, silicon measurement results in $130nm$ CMOS technology confirms the concept of dynamic data stability and designer's ability to trade timing and static parameters. It is shown that the low time constant due to the

subthreshold operation of the cell can be exploited to maintain data stability with proper choice of access and recovery time.

The impact of the new perception of d-stability is not limited to the design of new SRAM units. In light of the dynamic perception of data stability, conventional data stability test methods for the SRAM cells such as hammer test can be modified to reduce the risk of faults in the SRAM unit [26].

3.2 Background

Satisfaction of the d-stability criterion of a cell, has been the area of active research since late seventies [17, 18]. In [16], it is shown that the criterion for a cell to be able to hold two separate states for infinitely long time is that the feedforward and feedback transfer characteristic functions have three coincident points (See Figure 3.1).

It has been shown that there is a worst case *static* noise margin for any parameter (*e.g.*, supply voltage, ground voltage, series voltage etc.) that adversely affects the DC transfer curves. The notion of three point criterion and its implication on stability can further be explained with Figure 3.1 (a) and Figure 3.1 (b). In the former, added series noise sources of $V_{n1} = SNM1$ move both the curves adversely such that coincident points A and C converge. On the other hand, if noise sources are added in the supply path it requires the noise sources of the magnitude $V_{n2} = SNM2$ to converge points A and C rendering the cell unstable. It is apparent that $SNM1 \neq SNM2$. This chapter revisits the original three coincident point definition of the SRAM static d-stability and probes the concept of d-stability from the dynamic perspective such that an SRAM cell is dynamically accessed and non-accessed.

The SNM associated with input series voltage sources has received a considerable at-

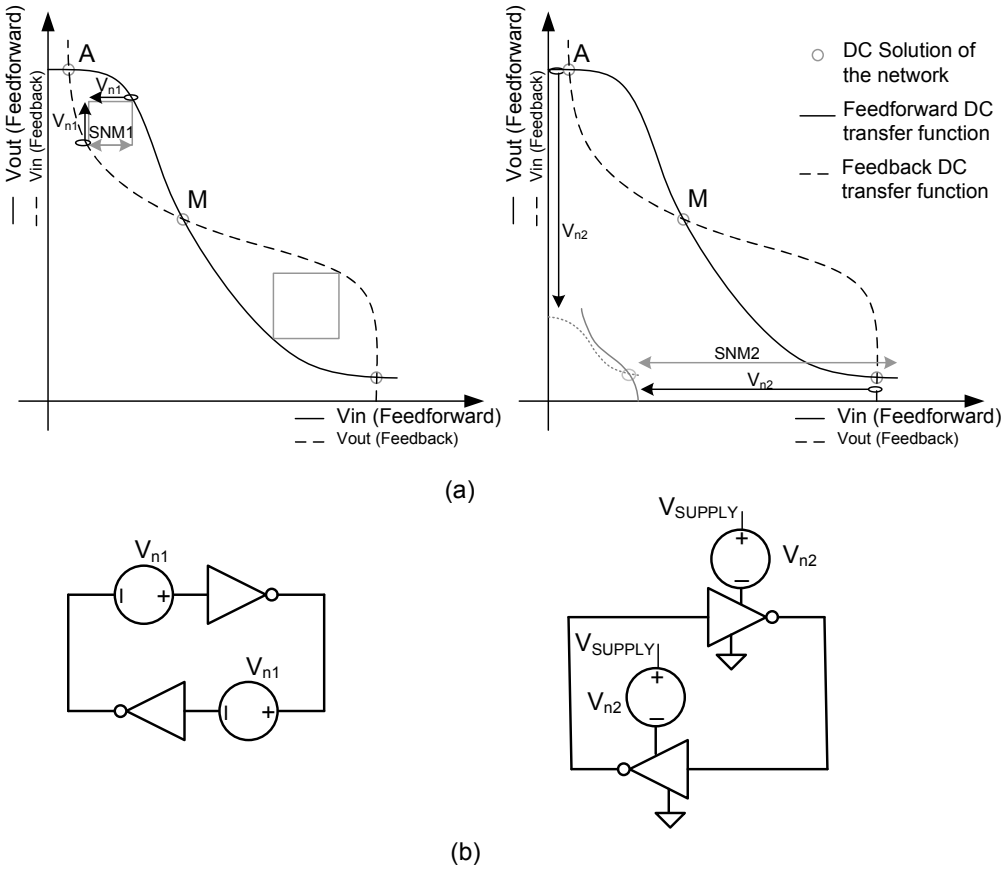


Figure 3.1 The SNM of an SRAM cell for input series voltage noise source and supply voltage noise source

tention owing to its ability to model the impact of transistor parameters on SRAM cell stability. These parameters include threshold voltage variation of the MOS transistors, conductivity and mismatch between MOS devices that construct the cell. The static noise margin associated with the input series voltage source is commonly referred to as the SNM in the literature and has dominated the perception of d-stability of a cell. Due to its static nature, SNM can be determined for an SRAM cell when it is either accessed or non-accessed. To ensure proper operation, the SNM of the cell when it is accessed has drawn

attention [20, 27] since it is lower compared to non-access situation. Static behavior of the *accessed* cell is the basis of the conventional SRAM cell transistor sizing methods [4, 28].

As we will see in the next chapter, dynamic perception of d-stability is relevant for new generations of SRAM cell and architecture were supply voltage is often manipulated to conserve power. Therefore, stability under dynamic circumstances becomes an issue. In addition, one can exploit the operational dynamics of the architecture to further reduce the leakage current and write power consumption while meeting the stability constraints.

3.3 SRAM Cell: A Dynamic System

3.3.1 Dynamic Data Stability

The basic cross-coupled SRAM cell is deceptively simple in appearance, yet attempts to analyze the dynamic behavior of such a nonlinear system have achieved only limited success [29]. As shown in Figure 3.2 when the access transistors are off (*i.e.*, the cell is non-accessed) the circuit has four nonlinear devices and two nonlinear capacitors. Since the capacitors do not result in a degenerated all C loop, voltages across them provide two state space variables assuming a constant power supply voltage. Therefore, a non-accessed cell can be described as a second order nonlinear time invariant system:

$$\frac{dV}{dt} = 1/C.I_r(V) \tag{3.1}$$

where $V = (v_1, v_2)$ is the vector defining the state of the system and $I_r = (i_{r1}, i_{r2})$ is the set of functions that describe the derivative of each state variable. In this equation, $1/C$ is a constant that balances the units of the two sides.

The cell can remain non-accessed for indefinitely long time and it must be able to retain the data. For this to happen nonlinear system such as the SRAM cell must have at least

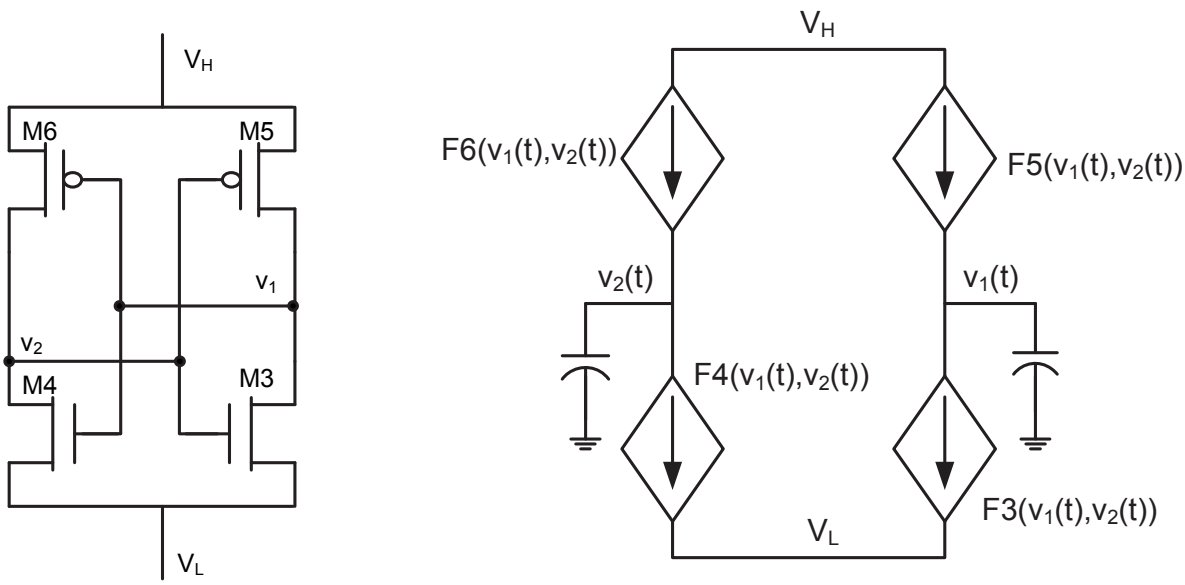


Figure 3.2 A non-accessed SRAM cell as a second order nonlinear circuit

two separate DC solutions. Practically, there must be at least three DC solutions for such a system [16]. These solutions are the separate coincident points on the DC characteristic transfer curves (See Figure 3.3) and evidently these solutions are the equilibrium points of the system described by (3.1).

Figure 3.3 shows the trajectories of the state variable in the two dimensional state space for the non-accessed cell for an arbitrary initial condition. Equilibrium points attract the state of the system towards themselves if the initial condition is in their region of attraction. Theoretically, if the cell remains non-accessed for an infinitely long time, the state of the cell will settle in one of the three stable points depending on its initial condition. In next Section, it will be shown that for MOS and BJT SRAM cells, stable points A and B are *uniformly asymptotically stable* (UAS) equilibrium points. For a symmetric cell, the regions of attraction of A and B are where $v_1 > v_2$ and $v_2 > v_1$, respectively. It can also

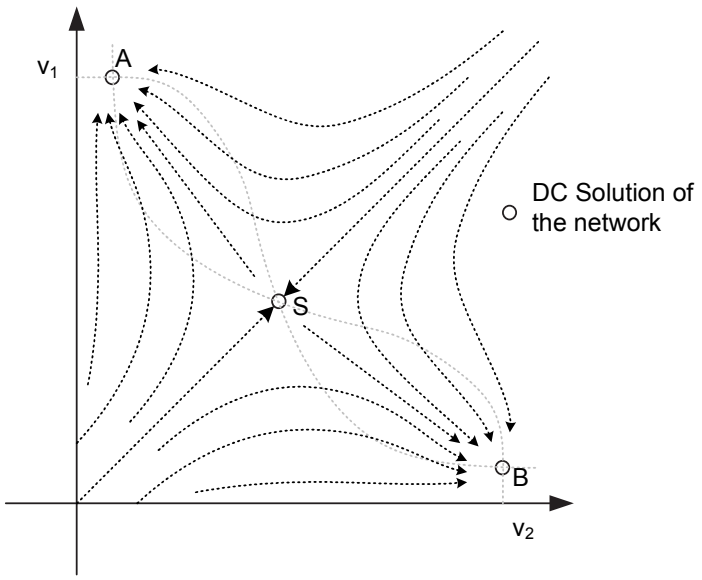


Figure 3.3 Trajectories of the state of the nonlinear system and its UAS points

be shown that the middle stable point, S, is a saddle stable point. A saddle stable point is a stable point whose region of attraction degenerates into a curve (metastable curve). The saddle point is also referred to as meta-stable point. The saddle point is the DC solution of the system only if the initial condition is on the metastable curve of the saddle point.

Figure 3.4 illustrates the analogy between the cell dynamic behavior in two dimensional space and the shadow of a ball on the plane when the ball is on a three dimensional surface that has a saddle shape. In this analogy, the two lowest points of the surface represent the two UAS points of the cell representing the logic states one and zero, respectively. If the initial location of the ball is on either side the saddle, the ball converges to the corresponding UAS point. On the other hand, if the ball is initially placed on the diagonal symmetric axis of the saddle, it ends up at the lowest point of the symmetric axis (S). Now, a small amount of perturbation can move the state variable to either side and push

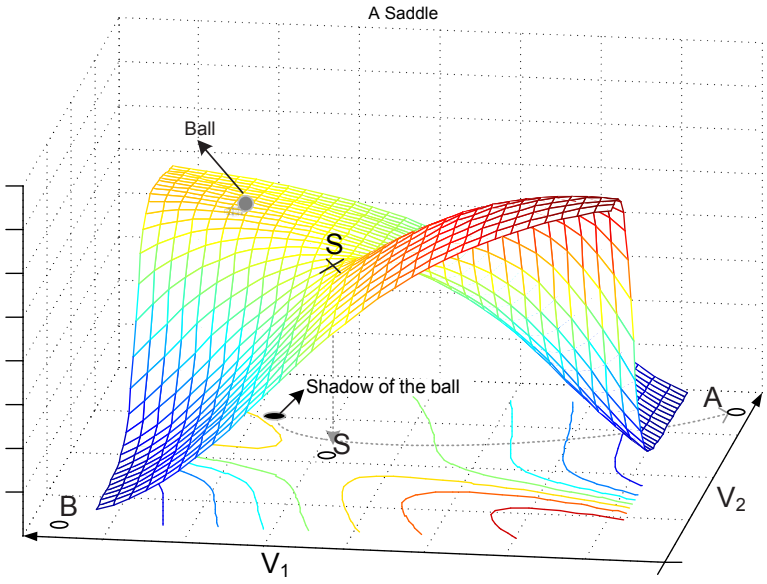


Figure 3.4 State dynamics of the system being analogous to the shadow of a ball on a saddle shaped surface

the state of the system to one of the UAS points.

A different set of nonlinear differential equations, $I_a(X)$ instead of $I_r(X)$ in (3.1) governs the cell dynamics when it is accessed. This change moves equilibrium points of the system. The new equilibrium points of the cell attract the state of the cell during the cell access time. Activation of the access transistors does not cause a discontinuity on the voltage of the capacitors since the access transistors have a finite admittance. Hence, the state of the system remains continuous. Unlike the non-accessed situation, in practice, the accessed mode does not last for infinitely long time for an SRAM cell. The access time of the cell (*i.e.*, the duration of time when the access transistors are on) is usually a design parameter, and a small fraction (*e.g.*, 15%) of the total access time in SRAMs [30, 31]. By definition, an SRAM cell is dynamically d-stable if and only if the state of the cell never leaves the region of attraction of the UAS equilibrium point associated with the initial

logic state during and after any number of access transactions. From here onward the cell access time is referred to as T_a and the time interval between the two consequent access and recovery time is referred to as T_r. If the cell is accessed for infinite number of times, it constitutes a second order periodic time variant (PTV) nonlinear system:

$$\frac{dV}{dt} = 1/C.I(V,t) \tag{3.2}$$

where $I = (i_1, i_2)$ is continuous and has a period of T $(= T_r + T_a)$ with respect to t. The solution of the system with initial condition $V = V_0$ at $t = t_0$ is denoted by $V(t, V_0, t_0)$.

The dynamic d-stability of the periodically accessed cell is the necessary condition for the d-stability of the cell in addition to the conventional static d-stability of the *non*-accessed cell [17]. A cell described by (3.2) is dynamically d-stable, under a periodic access if it provides at least two separate limit cycles, $\Phi_A(t)$ and $\Phi_B(t)$, within the regions of convergence of A and B, respectively where A and B are the UAS equilibrium points of the non-accessed cell. This condition guarantees that if the cell is accessed repeatedly for *any* number of times, the cell can still retain its original logic state. Also, it guarantees that the system returns to a specific trajectory if the system is perturbed by the thermal noise,*i.e.*, the system does not accumulate the noise over time when it is periodically accessed.

The basic behavior of the system (3.2) can be predicted based on the priori knowledge about the physical object that it describes. Since system (3.2) represents a dissipative system, there is at least one periodic solution for the system ([32], Chapter 2). In Appendix II, a theorem is proven which shows that a periodic system that is based on the alternation between two systems of I_a and I_r has the properties of a *convergent* system over region G, if for an initial condition V_0 at t_0 there exist a solution $V(t, V_0, t_0)$ that remains in G for any $t > t_0$ and both systems are monotonically asymptotically stable over G^*.

*For the special case of an SRAM cell, simulation results suggests that existence of a solution that

A convergent system has important properties; the system has a unique T-periodic solution $V = \Phi(t)$, this solution is Lyapunov-stable and it is the solution of the system for any initial condition within the region of attraction of $\Phi(t)$, as time approaches the infinity. In other words, depending on the number of regions G_i in the state space for which (3.2) satisfies the conditions of the theorem, there exist the same number of periodic limit cycle solutions for the system. Depending on in which region of attraction the initial condition lays, when time approaches the infinity, the solution of the system converges to the corresponding periodic trajectory which is the unique limit cycle associated with that region.

System (3.2) satisfies the d-stability critera, if it satisfies the condition of being convergent for two separate regions of G_A and G_B and these two regions are subsets of the regions of attraction of A and B, respectively where A and B are the two UAS equilibrium points corresponding to logic states in the non-accessed cell. This is because, each region G_i includes a unique attractive limit cycle solution, $\Phi_i(t)$. Practically, static d-stability of the accessed cell is a sufficient but not necessary condition for the d-stability of the cell. This argument will be explained using two examples.

Figure 3.5 shows the limit cycle trajectories of two different cells; cell A is a symmetric conventional cell for which DC transfer curves for both access and recovery systems have two UAS points and one saddle point. When the cell is accessed periodically there are three periodic solutions. If the initial condition of the cell is within the attraction region of anyone of the three periodic steady state solutions, the solution of the system will be attracted by that periodic solution. The region of attraction of the trajectory lays between the two saddle points is the diagonal symmetric axis of the saddle.

---

remains in G for $t > t_0$ is a sufficient condition to imply that the system is convergent in G, however, we could not found our theorem without the condition of monotonicity of the I_a and I_r over G

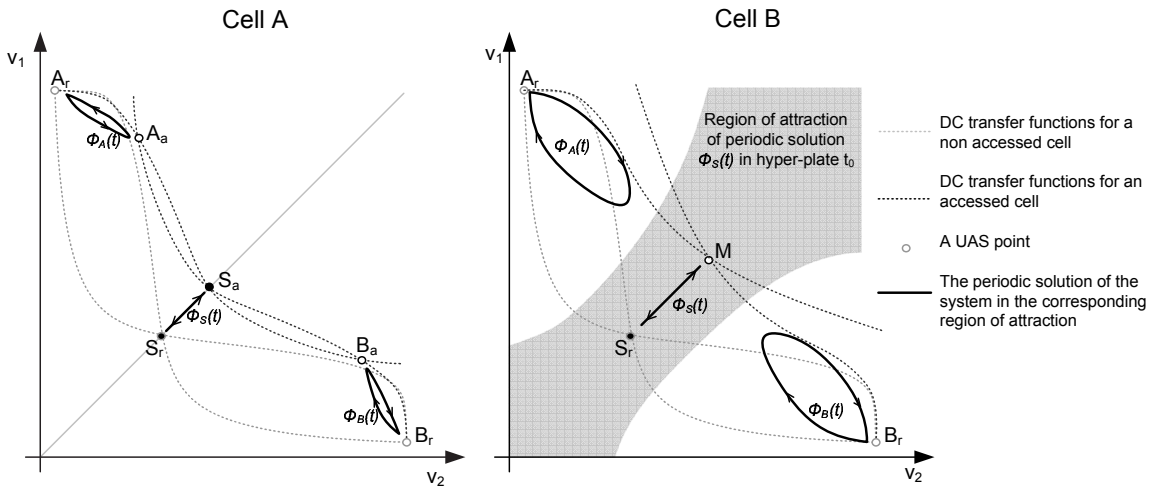


Figure 3.5 Trajectories of the limit cycles of the convergent system associated with an SRAM cell in the state-space for (a) a statically d-stable cell and (b) a statically d-unstable yet dynamically d-stable cell

In Figure 3.5, cell B has a similar dynamic behavior in the non-accessed (recovery) mode. However, this cell has only one UAS point in its access mode, hence statically d-unstable when accessed. One can imagine the dynamic of the state of the accessed cell as the XY-plane shadow of a ball on a bowl shaped surface with only one minima at the center instead of a saddle.

In practice, this situation may arise when access transistors are stronger than the drive transistors (*e.g.*, when the supply voltage of the cell is reduced while the wordline voltage remains the same.) Such a system can be dynamically d-stable provided the stability criteria is met. Hence, such a system can also have three limit cycles when it is periodically accessed. When the cell is accessed during T_a, the state of the system moves toward the only UAS equilibrium point of the accessed cell, M. If the access time is short enough, the state of the system remains in the region of attraction of the original state; A_r or B_r. When the cell returns to the non-accessed mode during T_r, it recovers to the original state.

The shape and regions of attraction of each trajectory depends on transistor sizes, T_a and T_r. In addition, the shape of the word-line activation waveform can also influence it. Cell B in Figure 3.5 is able to hold the data eventhough the accessed cell does not satisfy the static d-stability criteria provided the time constant of the accessed cell is too slow to pull out the state of the cell out of the region of attraction of the A_r and B_r during the access time. In other words, under a periodic access, the ability of holding a binary state depends on both accessed and non-accessed modes, as well as on their relative times T_a and T_r.

3.3.2 Noise Margins

The conventional measure of the static d-stability of an SRAM cell is based on the static noise margin (SNM). The SNM is defined as the maximum DC series voltage if adversely connected to inputs of the internal inverters of the accessed cell, the cell can still satisfy the static d-stability criteria. In other words, it has three separate DC solutions [18]. This value is represented in Volts and can be calculated by measuring the side of the biggest square that can fit in the DC transfer curves of the feedback and feedforward inverters (See Figure 3.1). Thanks to the static nature of the device imperfections (*e.g.*, mismatch, V_{th} variation, etc.), SNM has received a significant interest. It shows to what degree of imperfection can be tolerated if the imperfection is modeled as two equivalent series DC voltage sources adversely connected at inputs of both inverters in the cell. Therefore, SNM has a static nature in terms of both the type of the noise source and the criteria for d-stability.

Static noise sources also influence the dynamic d-stability of the cell. However, unlike the series DC voltage sources that have a direct and apparent effect on the number and location of the DC solutions, these sources have an indirect influence on the number of the periodic solutions. That is due to the fact that any DC noise source not only changes

the location and the number of the equilibrium points, but also varies the time constant of the loop. As suggested in [33], the reliability of the accessed mode depends on all random parameters which contribute to the dynamic behavior of a cell when it is accessed. Any static random parameter can have a corresponding $SNMD_P$ where $SNMD_P$ is the maximum tolerable noise margin associated with that parameter that keeps the dynamic d-stability criteria satisfied. For example, the maximum threshold voltage variation of access transistor that an SRAM cell can tolerate and still satisfies the dynamic d-stability criteria need to be considered in a design. Similarly, the maximum access time T_a for a T (the period of access transaction) that the cell can tolerate while satisfying the dynamic d-stability criteria need to be considered in a design. Although it does not have a direct mathematical link to the number and shape of the limit cycle trajectories, it is instructive to form $SNMD$ based on the conventional definition. The $SNMD$ can be defined as the the maximum series DC voltage source if adversely connected to inputs of internal inverters, there will be no limit cycle trajectory within the region of attraction of each logic state equilibrium points. Clearly, if $T_a = T$, then $SNMD = SNM$.

Broadening the concept of stability from static to dynamic perspective opens opportunities for new trade-offs between power consumption, d-stability of the cell and speed. As will be discussed in the next section, reduction of the supply voltage of the cell as well as reduction in write power consumption are possible. Moreover, if the speed is not a priority, for a fixed access time, a longer recovery time, T_r, provides a higher d-stability. Similarly, in low power applications, reducing the cell voltage reduces the cell power consumption at the cost of d-stability. However, now if the T_r is increased, one may reclaim part of the d-stability. This is due to the fact that the effective d-stability is the combination of the stability in access and non-accessed situation.

3.4 Simulation Technique for Data Stability Analysis

Complex transistor models together with the time variant nature of a periodically accessed cell makes the mathematical analysis of an SRAM cell a formidable task. However, the concept of the dynamic d-stability of an SRAM cell can be verified using computer based circuit simulators. Unlike static d-stability criteria that can be verified using a DC simulation, dynamic d-stability requires both time domain and AC simulations to characterize an SRAM cell.

As described in the previous section, a sufficient condition for dynamic d-stability is to have two regions of G_A and G_B within the region of attraction of the UAS equilibrium points of the respective logic states, A and B. The following conditions must be satisfied by in those two regions: 1) There exist at least one solution $V(t, V_0, t_0)$ that remains in G_i for all $t > t_0$. 2) Both I_r and I_a are monotonically asymptotically stable over G_i. It is noteworthy that in some cases there exist limit cycle trajectories that does not satisfy the latter condition yet are asymptotic. However, the theorem proved in the appendix can only guarantee the asymptotity for a solution only if it holds condition (2).

Two simulation steps are required to verify the two conditions for the dynamic d-stability of an SRAM cell. Verification of the first condition is convenient using a time domain simulation. If for an arbitrary initial condition of V_0 at an arbitrary phase of the periodic alternation t_0, the state of the cell does not leave its original region of attraction after sufficiently long simulation time(*i.e.*, sufficiently large number of consecutive accesses), the first condition is satisfied.

In general, finding out the region of attraction of a UAS point in a nonlinear system is not trivial. For nonlinear system analysis, often Jacobian matrix is formulated for linearization [34]. Therefore, we can form the Jacobian matrix around each point of the state space to find out the behavior of the trajectories at that point. If all eigenvalues

around a point are negative, it represents an exponentially, hence monotonically, converging trajectories around that point towards an equilibrium point.

For the SRAM cell shown in Figure 3.2, the Jacobian matrix of the retention system I_r around point $V_0 = (v_{01}, v_{02})$ can be constituted as:

$$\begin{aligned} \dot{v_1} &= 1/C.i_{r1}(v_{01}, v_{02}) + 1/C.g_{11}.(v_1 - v_{01}) + 1/C.g_{12}.(v_2 - v_{02}) \\ \dot{v_2} &= 1/C.i_{r2}(v_{01}, v_{02}) + 1/C.g_{21}.(v_1 - v_{01}) + 1/C.g_{22}.(v_2 - v_{02}) \end{aligned} \tag{3.3}$$

where

$$\begin{aligned} g_{11} &= \left.\frac{\partial i_{r1}(v_1, v_2)}{\partial v_1}\right|_{v_1=v_{01}, v_2=v_{02}} = -\left. g_{o1}\right|_{v_{01}, v_{02}}, \quad g_{12} = \left.\frac{\partial i_{r1}(v_1, v_2)}{\partial v_2}\right|_{v_1=v_{01}, v_2=v_{02}} = \left. g_{m1}\right|_{v_{01}, v_{02}} \\ g_{21} &= \left.\frac{\partial i_{r2}(v_1, v_2)}{\partial v_1}\right|_{v_1=v_{01}, v_2=v_{02}} = \left. g_{m2}\right|_{v_{01}, v_{02}}, \quad g_{22} = \left.\frac{\partial i_{r2}(v_1, v_2)}{\partial v_2}\right|_{v_1=v_{01}, v_2=v_{02}} = -\left. g_{o2}\right|_{v_{01}, v_{02}} \end{aligned} \tag{3.4}$$

where g_{o1} and g_{o2} represents the small signal output admittance of the inverter driving node 1 and 2, respectively, and g_{m1} and g_{m2} represents the small signal transconductance of the same inverter at the DC operating point of V_0. Transconductance, g_{m1}, can be written in terms of the small signal voltage gain of the same inverter $-A_{v1}$; $g_{m1} = -A_{v1}/r_{o1}$ where $r_{o1} = 1/g_{o1}$. Here, the negative sign represents the negative gain of the inverter. Similarly, $g_{m2} = -A_{v2}/r_{o2}$. Neglecting the voltage dependency of capacitances at both nodes, equation(3.3) can be written as:

$$\begin{pmatrix} \dot{v_1} \\ \dot{v_2} \end{pmatrix} = \begin{pmatrix} -1/\tau_1 & -A_{v1}/\tau_1 \\ -A_{v2}/\tau_2 & -1/\tau_2 \end{pmatrix} \times \begin{pmatrix} v_1 - v_{01} \\ v_2 - v_{02} \end{pmatrix} + \begin{pmatrix} 1/C.i_{r1}(v_{01}, v_{02}) \\ 1/C.i_{r2}(v_{01}, v_{02}) \end{pmatrix} \tag{3.5}$$

where $\tau_1 = r_{o1}.C$. It is easy to verify that 1) the system never provides complex eigenvalues and 2) eigenvalues are both negative if and only if $A_v = A_{v1}.A_{v2} < 1$. In other words, the

positive feedback loop of the two inverters behave similar to a monotonically stable system (*i.e.*, exponentially decaying) only over the region where the loop gain is less than one, regardless of the time constant of the loop.

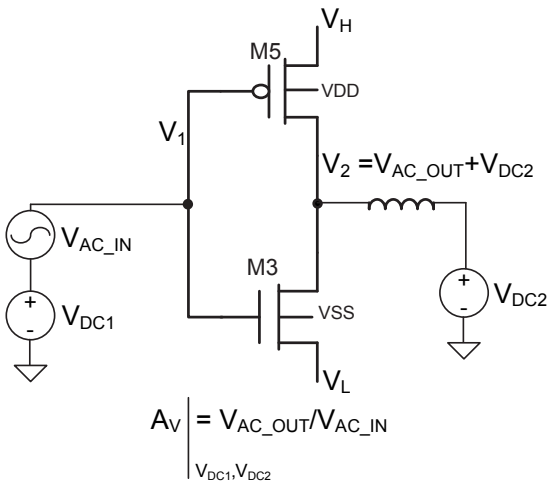


Figure 3.6 Proposed circuit for the derivation of the small signal gain of an inverter over the state space

Derivation of the small signal gain of an inverter at any input output DC operating point is possible using the circuit diagram shown in Figure 3.6. In this figure, V_H and V_L are the supply voltage and the ground voltage of the cell. The inductor sets the DC operating point at the output. If the inductance is chosen to be high enough, it decouples the DC source at the output of the inverter from influencing the AC analysis at high enough frequencies. Small signal gain of the inverter can be found over the entire state space by sweeping the DC voltage at the input and output of the inverter. For a typical inverter, this results in a saddle shape surface. This simulation can be conducted for both feedback and feedforward inverters to obtain the loop gain.

Figure 3.7 (a) shows the loop gain of an SRAM cell when the V_H and V_L of the cell are

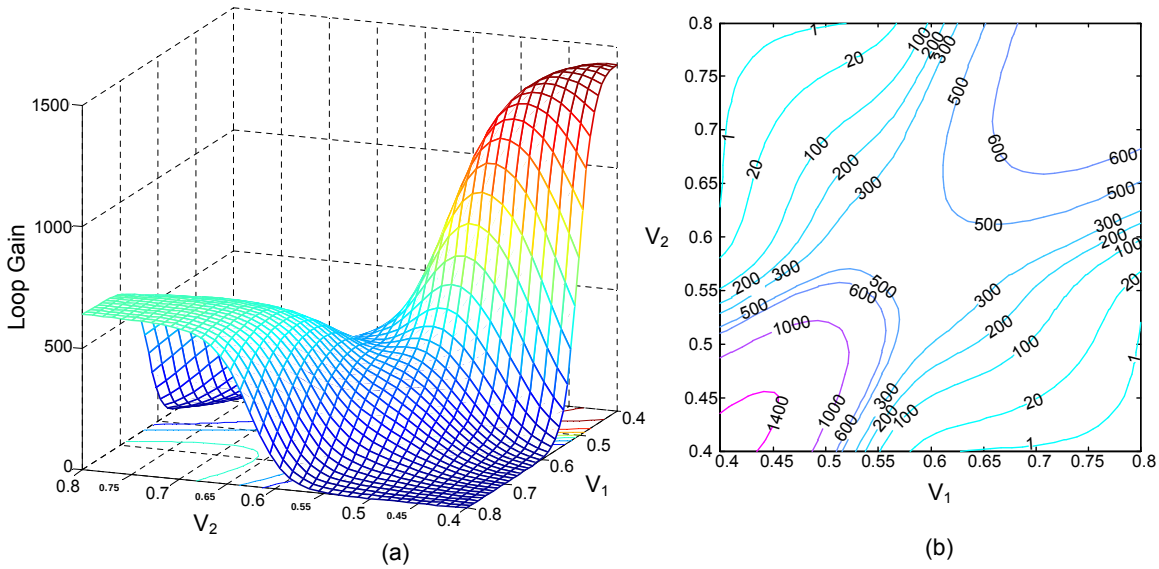


Figure 3.7 The simulated loop gain of a typical SRAM cell over the state space for a $130nm$ CMOS technology (a) and the contour of the gain in the state space (b)

at $800mV$ and $400mV$, respectively. Multiplication of the small signal gain of the feedback and feedforward inverter provides the loop gain at any point of the state space. Figure 3.7 (b) shows the contour of the loop gain in the state space. It can be seen that the areas around the equilibrium points associated with the logic states has a loop gain of less than unity. Hence these points are UAS as argued in Section I.

The same simulation procedure can be applied to the accessed system I_a. Clearly, the same areas with the gain of less than 1 in I_r will have a gain of less than 1 in I_a too since the only difference between the two systems is that the output of the internal inverters are more loaded in I_a which results in a lower gain. Hence, derivation of the loop gain for the retention system I_r is sufficient to find out which areas can maintain the monotonicity of the trajectories in both accessed and retention modes.

3.5 Dynamic Data Stability in Low-power Circuit Design

SRAM cell leakage reduction has been the focus of research in recent years [12]. The exponential relationship between the supply voltage, and the leakage current is well known. However, a straight forward reduction of the supply voltage in an SRAM cell has several shortcomings. A low-voltage cell, does not provide a sufficient gate-source overdrive voltage, V_{gs}, over the drive transistors to discharge the bitline when the cell is accessed for the read operation. On the other hand, clever circuit and architecture techniques may offer significant benefits in active and leakage power reduction. This issue has lead many SRAM designs to alternate the supply voltage of the cell when the cell undergoes different operational modes [35, 25, 36, 37].

Three operational modes have been conventionally used to describe the behavior of an SRAM cell: retention, read and write operations. If N words are organized in a row, for each read or write operation in the row, all N words are enabled through the activation of the corresponding wordline. However, $(N-1)$ words on the row are not selected. Cells in these words are expected to retain their data. Non-selected words are deemed to be in *Accessed Retention Mode* (AR-Mode). In a conventional design the non selected words in the given row are treated similar to those in the read operation (selected word). They discharge the corresponding bitlines. In other words, these non-selected $(N-1)$ words are causing dynamic power consumption eventhough their data is not read. Unfortunately, the supply voltage alternating schemes proposed in [25, 36] are no better in this context since AR-mode in them is treated the same way as the read operation. Needless to say they also result in extra dynamic power consumption and limits the effectiveness of the leakage reduction methods.

For low power operation, it is desirable that the dynamic power associated with the AR-mode words is minimized if not eliminated. Column based circuit techniques can be employed to keep the the AR-mode words in low supply potential so that the driver does not have the strength to discharge the bitline voltage [35, 37]. In other words, when the cells go to AR-mode, the supply voltage of the cell remains unchanged just leaving sufficient headroom for the d-stability of the cell. The application of this operational mode will be further explained when the cell is embedded in a low-power architecture that will be introduced in Chapter 4.

The d-stability of the cell in the AR-mode can be guaranteed using the extra headroom granted by the criteria for the dynamic d-stability of the SRAM cells. In other words, the actual extra d-stability noise margin, which is the result of the dynamic of the cell in both accessed and retention mode, allows the cell to operate properly under a significantly lower voltage. In fact, operation of the cell in the subthreshold region results in a slower time constant which improves the d-stability under a pre-determined short access time. This effect gives rise to the dynamic data stability of the cell that has been explained in the previous section. In other words, with proper choice of access and recovery time, a cell can tolerate lower supply voltages (lower SNM) and fulfills the duty of retaining the data. The additional supply voltage headroom can be budgeted to save write, read and static power consumption.

The following subsections, we will substantiate the concept of dynamic d-stability with simulation as well as measurement results. It is noteworthy that the d-stability in the read operation is a subset of d-stability in the AR-mode. In other words, for a given T_a and T, if the cell remains stable in AR-mode, it is also stable in the read operation. This is due to the fact that the drive transistors are relatively weaker in the AR-mode than in the read operation when the source of the drive transistors are grounded. The stronger

drive transistors offer a higher stability. This argument is verified by measurement results presented in the next two subsections.

3.5.1 AR-Mode stability simulation

Subthreshold operation of the CMOS transistors has received attention in recent years [38]. Such circuits impose a substantial delay time to the operation of the circuit due to the low current drive. Therefore, the time constant of the circuits that are based on subthreshold operation of MOS devices are significantly longer compared to the same devices operating in the linear mode.

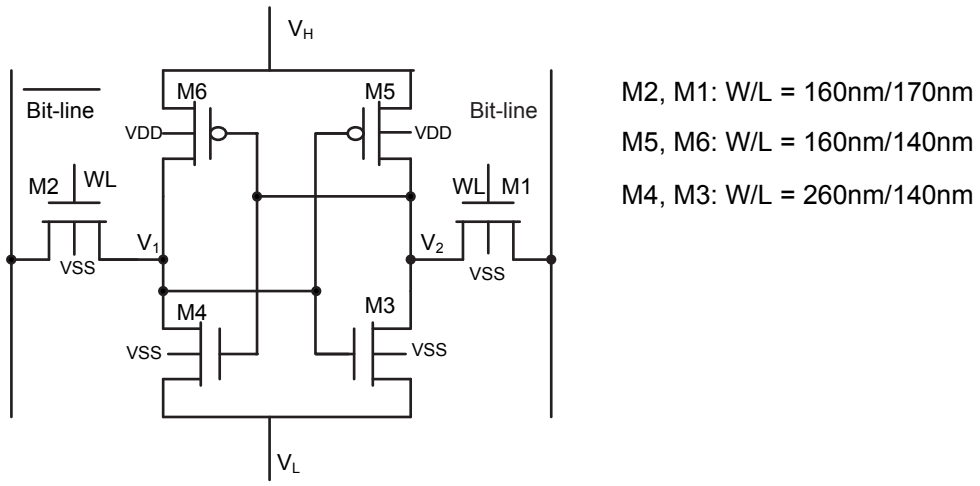


Figure 3.8 The voltage setting in an SRAM cell

Figure 3.8 , shows an SRAM cell in a $130nm$ CMOS technology with nominal supply voltage of 1.2V. In this figure, it is assumed that node 1 carries a logic '0' at V_L. The drive transistors are able to draw $50\mu A$ current from the bitline under the nominal voltage for the read operation. This current produces sufficient voltage drop on a capacitive bitline

with 256 rows in $2ns$ for the read operation. For AR-mode cells, the supply voltage of the cell is $0.6V$ ($V_H = 0.9V$ and $V_L = 0.3V$) while the wordline voltage in the access time is kept at $V_{WL} = 1.2V$. The Spice simulated waveforms shown in Figure 3.9 (a) illustrate the behavior of the cell when it goes to the AR-mode. The threshold voltage of the NMOS transistors are around $0.46V$. Node 1 raises up to $0.5V$ as a result of the voltage division between M1 and M3 when the cell is accessed. This voltage sets the V_{gs} of M4 at $200mV$ which is considerably below the threshold voltage of M4. Subthreshold operation of M4 delays the discharge of node 2. It takes more than $16ns$ for data in the cell to get corrupted. In other words, for a typical access time of $2ns$ the d-stability is ensured.

The d-stability in AR-mode can be further explained with the help of Figure 3.9 (b). This figure shows simulation results of a cell that is alternating between the AR-mode and the retention mode. In this example, the cell access time and cell recovery time are assumed to be equal and equal to $3ns$. It should be noted that the typical cell access time is considerably smaller than the cell recovery time in practice. It can be seen that the cell recovers the original logic state in the recovery time.

Figure 3.9 (c) shows the state-space trajectory of the state variable. The graph in the figure is extracted from the simulation results shown in Figure 3.9 (b). The limit-cycle trajectory attracts the solution associated with the initial condition and the state of the cell orbits on the trajectory under the periodic access. In simple terms, the trajectory shows that the cell is dynamically d-stable.

3.5.2 AR-Mode measurement

The SRAM cell shown in Figure 3.8 is implemented in a $130nm$ CMOS technology. Figure 3.10 shows the silicon micrograph and the layout of the measured test chip. The SRAM unit is designed such that the access time and recovery time, as well as voltage levels of

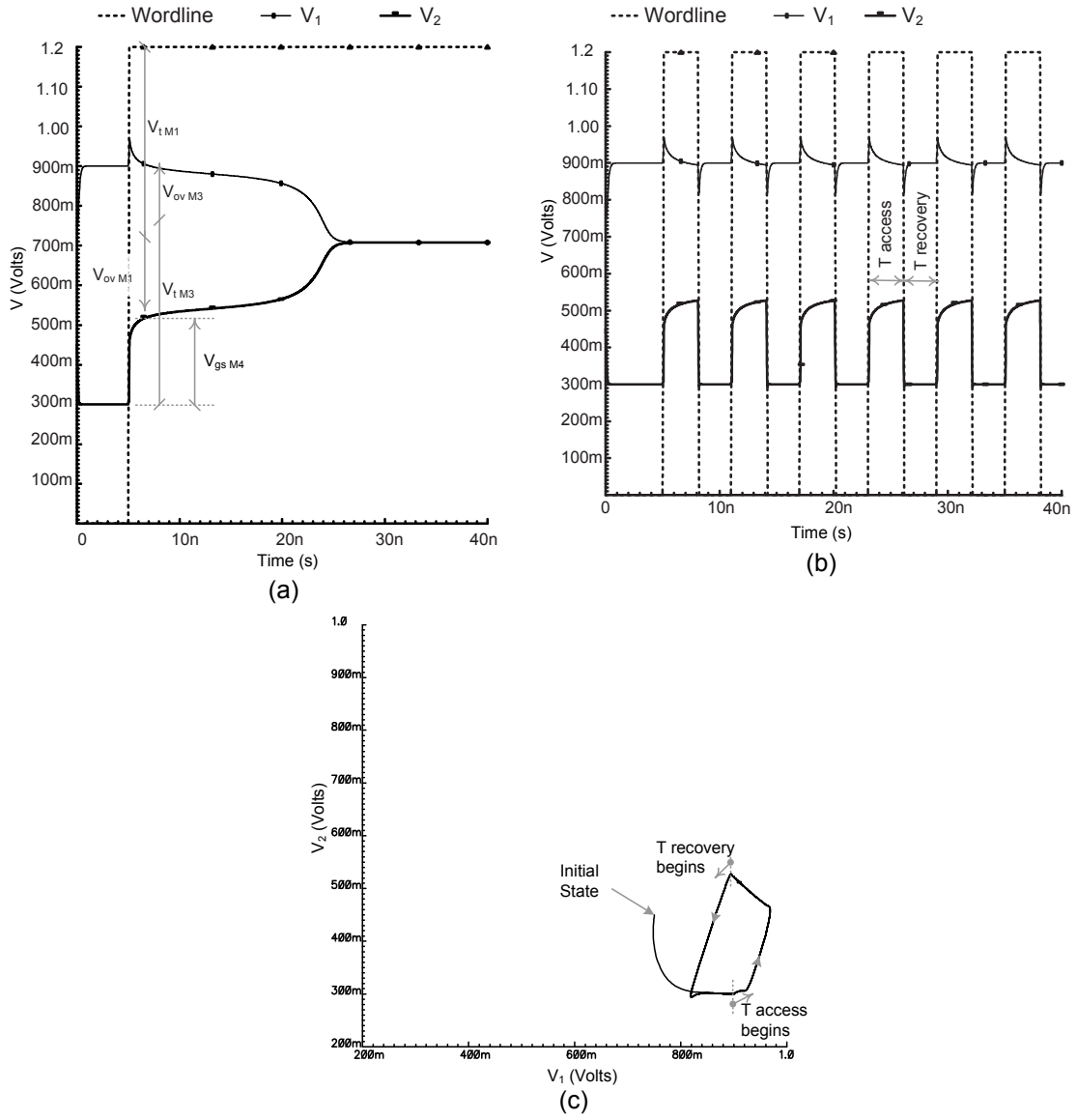
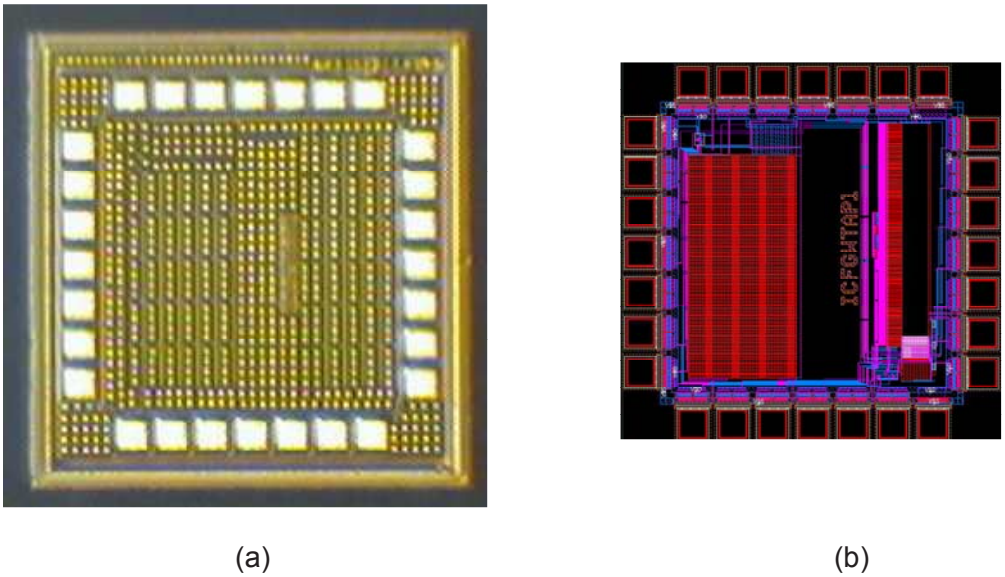


Figure 3.9 HSpice simulated waveforms of a cell in the accessed retention mode; (a) behavior of a cell when accessed for long time, (b) behavior of a cell when accessed periodically, (c) state space trajectory



(a) (b)

Figure 3.10 AR-Mode test chip: (a) silicon micrograph (b) layout

V_{WL}, V_H and V_L can be set accurately. Two measurement setups were devised to verify the concept of dynamic d-stability. In both setups, the cell is statically d-unstable when accessed and the internal transistors operate under subthreshold region. However, experimental observations show that the proper choice of access time and recovery time offers dynamic d-stability. Trade-offs between the design parameters are investigated in both setups.

In the first test setup, the voltage levels were kept constant as $V_{WL} = 1.012V, V_H = 0.826V, V_L = 0.404V$. The cell access time, T_a, and cell recovery time, T_r, are swept to examine under what conditions, the cell is stable. The measurement results are shown in Figure 3.11. Owing to the changes in the cell access and cell recovery time, the frequency of the access $T = T_a + T_r$ is also changed which is also depicted at various data points in the figure. The figure illustrates the trade-off between the cell access time and the cell recovery

time in order to to keep the cell d-stable. It is observed that the cell is d-unstable if the cell access time exceeds $53ns$ independent of the recovery time. For cell access times less than this value, however, an appropriate value of the cell recovery time can be established such that the cell is d-stable. In other words, for a particular operational frequency the reduction of the cell access time can result in a higher cell d-stability.

A closer inspection of the graph in this figure substantiates the concept of dynamic d-stability. Figure 3.11 (b) illustrates the same results in a different way. In this figure the X-axis represents the frequency of operation whereas the Y-axis shows the maximum tolerable duty cycle (T_a/T) that ensures the data stability. For high frequencies, the required recovery time T_r for d-stability is relatively constant. Therefore, as the frequency decreases (*i.e.*, larger T), the tolerable cell access time duty cycle increases. However, as it was mentioned before, the maximum access time that can offer d-stability is limited. Hence, any increase in T (*i.e.*, lower frequency) only increases T_r which results in the drop of access time duty cycle.

This experiment is consistent with the simulation results presented in Figure 3.9 (a) and (b) which represents the behavior of the cell operating in the subthreshold region. According to Figure 3.9 (b), the required recovery time for d-stability is small under nominal timing conditions. When the cell is accessed, the internal node voltages of the cell remain relatively constant. Therefore, T_a and hence T can be increased without adding corresponding increase in the T_r while maintaining the stability. Similarly, in Figure 3.11 (b) as T is increased, the cell is d-stable for larger duty cycle. However, as the absolute value of cell access time exceed certain value, shown in Figure 3.9 (a) the cell differential voltages converge requiring extremely large T_r. Therefore, the duty cycle becomes smaller in order to maintain the d-stability. The Figure 3.11 (b) depicts a similar behavior for frequencies below 40 MHz.

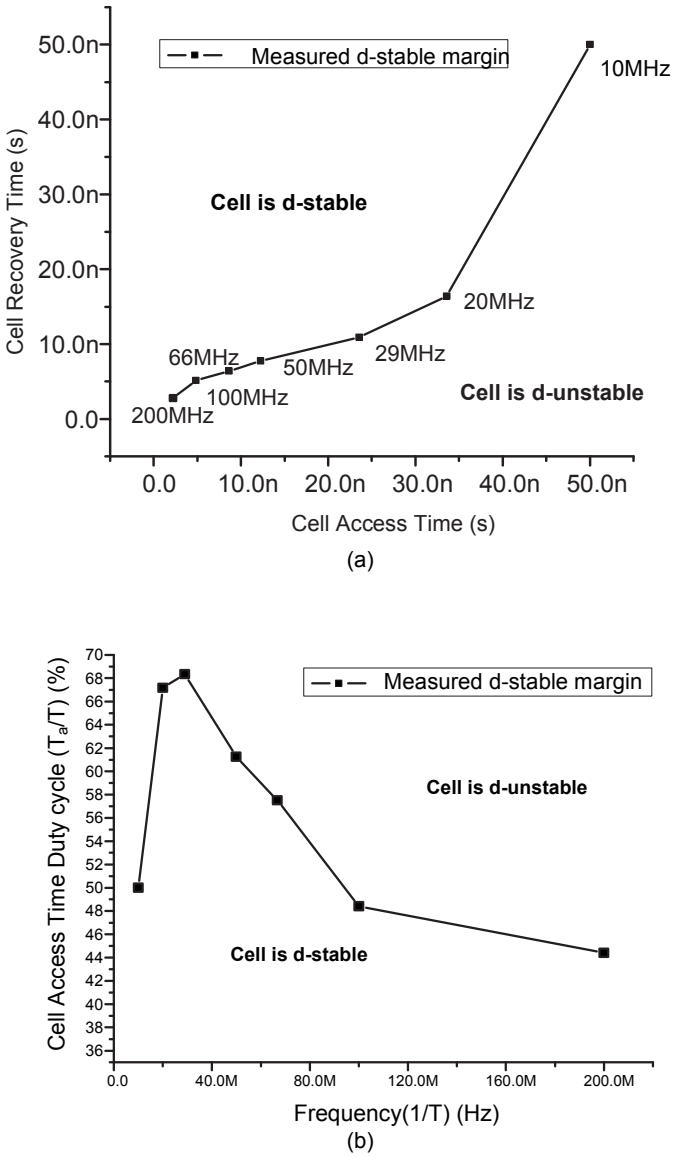


Figure 3.11 Measurement results indicating the trade-off between (a) the access time and recovery time to obtain data stability and (b) frequency of operation and the duty cycle

The second setup investigates the dependency of dynamic d-stability on the V_{WL} and V_L and the cell access time for a given frequency of 100MHz. Figure 3.12 shows the trade-off between wordline voltage V_{WL} and the cell access time for different values of V_L. It is clear as the value of V_L is reduced, the cell remains d-stable over a wider range of the cell access time and V_{WL}. Moreover, for a reduction of the cell access time for a given V_{WL}, improves the d-stability. This d-stability improvement now can be traded-off against a higher V_{WL} or lower threshold voltage for the access transistors. Conversely, for a given access time, the V_{WL} can be reduced for enhanced d-stability. The part of this enhancement can be traded to increase the cell access time which results in relaxed timing, lower active power consumption.

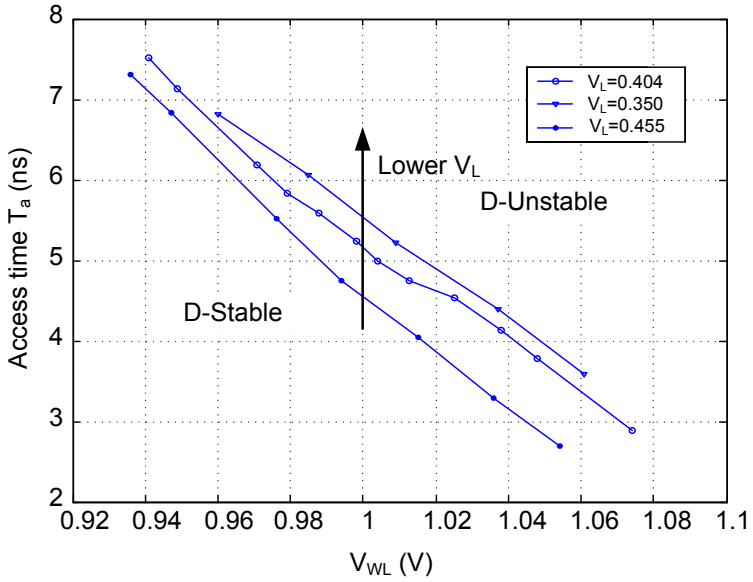


Figure 3.12 Measurement results indicating the trade-off between the access time and the wordline voltage for a cell being accessed at 100MHz

3.6 Dynamic Data Stability in SRAM Testing

Testing of the SRAM cells in an array for data stability has been the focus of active research owing to the rising density of this circuit in modern SoCs. Several conventional methods have been suggested in the literature for this purpose. The effectiveness of these methods which are based on static d-stability criteria can be improved in light of the proposed perception of d-stability.

In this section, the limitation of a conventional test method, namely hammer test, in detecting marginally d-stable cells will be elaborated. A test method that mitigates this limitation is also introduced [39].

3.6.1 Hammer Test Effectiveness

This section shows some cases in which hammer test fails to detect a cell with marginally dynamically d-stable. Technically, in such cases although the cell is able to retain the original data under normal conditions, they may not be considered as a reliable cell when it comes to high yield and quality products. Especially, when the MOS device characteristics changes over time because of temperature variations or fatigue the reliability of the cells in retaining the data becomes a concern [40].

Inverter Offset Voltage

Figure 3.13 shows a six transistor SRAM cell which suffers from an offset voltage between the internal inverters due to the excessive threshold voltage mismatch between the transistors. Figure 3.14 shows the feedback and feedforward DC transfer curves under an excessive offset voltage. It is evident that the offset voltage moves feedback and feedforward DC transfer curves adversely. For sufficiently high offset voltages, the accessed cell

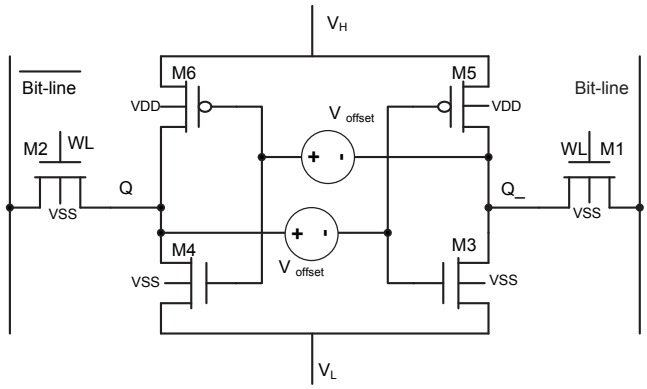


Figure 3.13 Six transistor cell with offset

does not satisfy the static data stability criteria (i.e, the number of intersect points reduces to one point.) However, the non accessed cell can still provide two DC states.

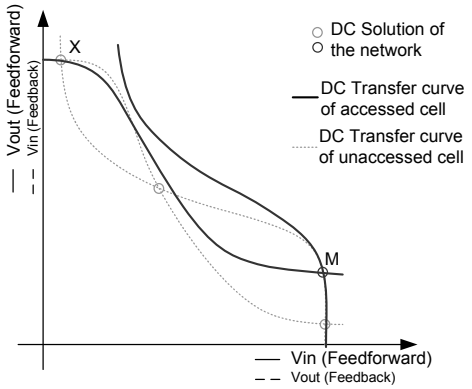


Figure 3.14 DC transfer curves of a statically data unstable accessed cell

In Figure 3.14, assume the non-accessed cell holds state X. When the cell is accessed the cell state is attracted by the only DC solution of the accessed cell, state M, with the initial condition of the cell at X. The attraction time toward M depends on the offset voltage. Figure 3.15 shows the behavior of the cell node voltages under different offset values. It

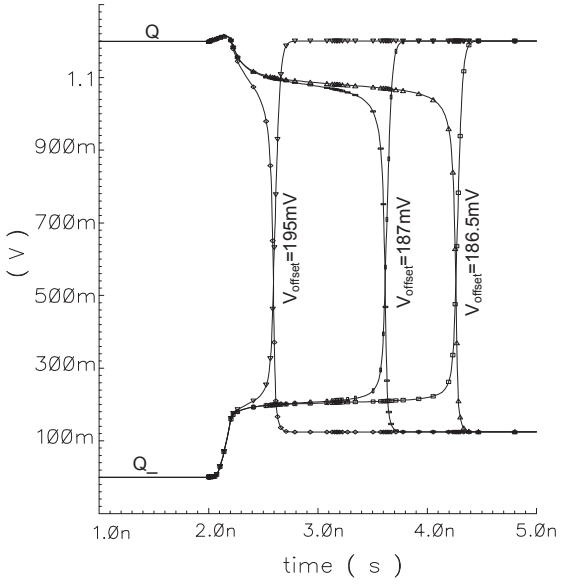


Figure 3.15 Spice simulated waveforms of the cell internal node voltages under different offset voltages

can be seen that when the offset voltage increases, the cell flips faster and a low offset value can extend the time required to flip the cell.

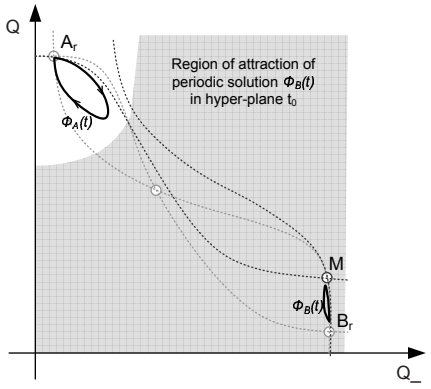


Figure 3.16 Trajectory of the state variable in the state space for the statically d-unstable cell

Hammer test, with a regular cell access time is not able to make the cell flip and detects the marginality. That is because the cell becomes dynamically d-stable and hides the fault; If the offset voltage is not high enough, the cell does not change the state during the cell access time and returns to the original state X when the access transistors are deactivate. Hence, the cell portrait itself as an un-faulty d-stable cell. Figure 3.16 shows the typical trajectories of the state space variables when the cell is accessed periodically. Figure 3.17 shows the waveforms associated with the internal nodes of the cell when the offset voltage is equal to 186mV. It can be seen that for cell access time equal to 1.5nS and retention time of 3.5nS, repetitive access is unable to make the cell flip and show the marginalities. Figure 3.18 shows the dependency of the flipping time on the offset values.

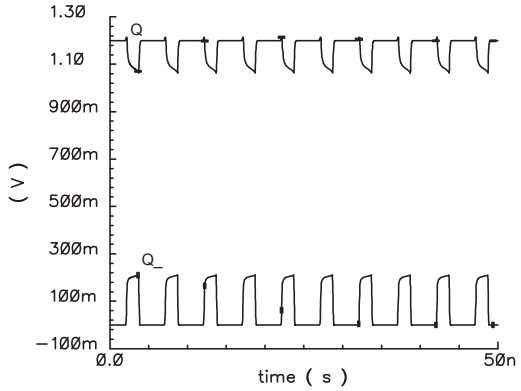


Figure 3.17 Spice simulated waveforms of the same cell goes under hammer test

Access Transistor Threshold Voltage Mismatch

As indicated in [41], threshold voltage variations due to the random doping concentration is an emerging issue in the scaled VLSI technologies. Significant reduction of the stability of the accessed cell is possible if the threshold voltage of the drive transistors increases

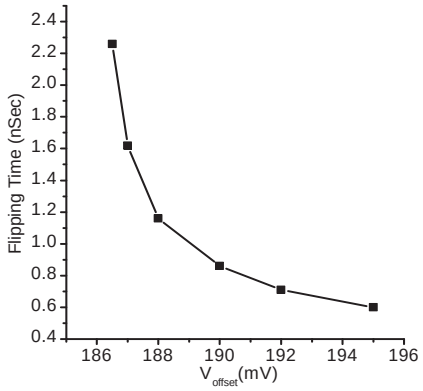


Figure 3.18 Flipping time dependency on access transistors threshold voltage mismatch

while the threshold voltage of the access transistor drops. In that case, failure in detecting marginally d-stable cells that was discussed in the previous sub-section can happen under lower inverter offset voltages. Combined with the inverter's offset voltage, access transistor's random threshold voltage variation increases the probability of the failure of the hammer test with lower inverter's offset voltage. The probability of the static data instability increases even more when the difference between the threshold voltages of the access transistors moves the DC transfer curves adversely.

Figure 3.19 demonstrates the dependency of the flipping time on the access transistors' threshold voltage mismatch during the activation of the access transistors. In order to mimic the said situation in which the access transistor becomes stronger the simulations were carried out by reducing the supply voltage by 12% and reducing the threshold voltage of the access transistors by 100mV compared to the nominal values. Reduction of the supply voltage weakens the drive transistors whereas reduction of the threshold voltage of the access transistor strengthens the access transistors. It can be seen that under a variety of threshold voltage mismatch between the access transistors and inverter's offset voltage

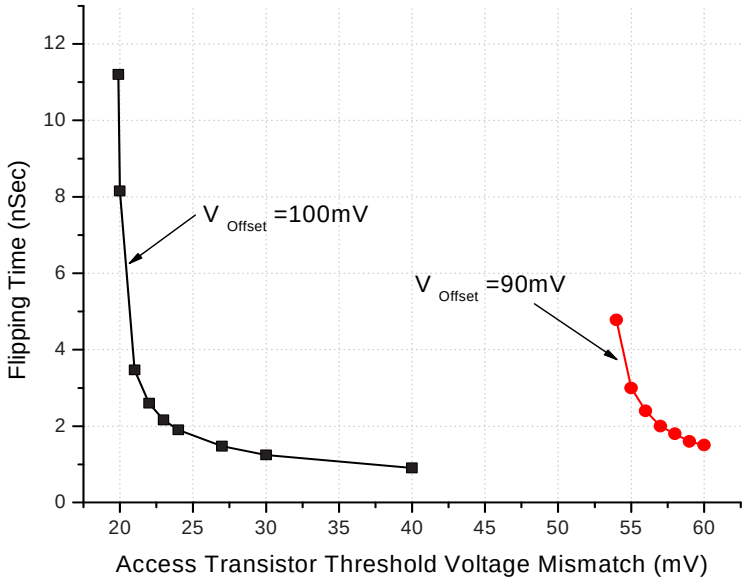


Figure 3.19 Flipping time dependency on access transistors threshold voltage mismatch

variation, the cell needs more than 1nS to flip and reveals the marginality.

Hammer test with access time less than the flipping time does not reveal the excessive mismatch, no matter how many times it repeats. That is because the convergence time towards the initial condition during the retention time is rather small and the cell recovers its original state quickly if it does not flip during the flipping time.

Gate Resistive Open Defects

Resistive open defects are typical faults that may occur in the scaled technologies. Some times these faults can directly affect the static behavior of the cell. For example when the resistive open defects appear at the drain of the drive or access transistors. Such faults can be detected using conventional tests. However, if a resistive open fault appears at the gate of the drive transistor, then it does not influence the static nature of the circuit as the

gate terminal is already open circuit. However, such defects prevent hammer test or any test that is based on conventional cell access time to identify undesired faults that cause marginal d-stability.

Figure 3.20 shows a six transistor cell with a resistive open defect at the gate of the drive transistor. The accessed cell is statically data unstable due to the excessive offset voltage between the internal inverters. As mentioned in the previous subsection, if the offset voltage is high enough the cell flips rather fast. In this case, however, the resistive defect at the gate of the drive transistors slows down the loop in settling to the DC solution of the accessed cell.

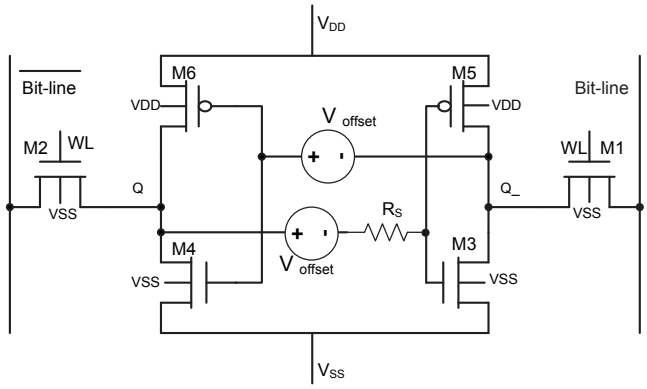


Figure 3.20 An SRAM cell that suffers from resistive open fault at the gate of the drive transistor

Flipping of a cell depends on the amount of resistance value. As suggested in [42], the resistive open defect at the gate of the transistor can reach up to a few mega Ohms. The RC time delay associated with the resistive open defect causes a balanced time lag for charge and discharge of the gate capacitance of the drive transistor. Therefore, the cell can be made to flip after a number of accesses. Figure 3.21 shows such a case. For a particular offset voltage and resistance, flipping of a cell depends on a number of parameters including

the duty cycle of the wordline activation and the number of accesses.

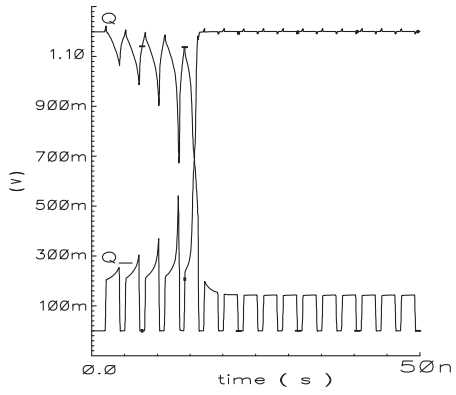


Figure 3.21 Hammer test is able to detect a faulty cell after several consecutive accesses

Hammer test can detect the defective cell if the cell flips after a finite time. However, this test fails to detect the failure if the state of the cell is not changed after a finite number of accesses. Figure 3.22 shows the flipping time dependency on the resistance value for a number of different offset voltages. The simulation results are for the constant duty cycle of 30%. It can be seen that the required number of access intervals for the detection of the faulty cells increases with the resistance value. Also, if the resistance exceeds certain amount, then the test is unable to detect the failure no matter how many times it is accessed.

3.6.2 Design For Test Technique

The number and shape of the periodic solution of the nonlinear periodically accessed cell depends on both V_a, and V_r functions as well as T_a and T_r. V_a, and V_r are both depend on the circuit characteristics and the faults in the accessed and retention modes. The state of a statically d-stable accessed cell, with a saddle shaped state-space characteristics for V_a,

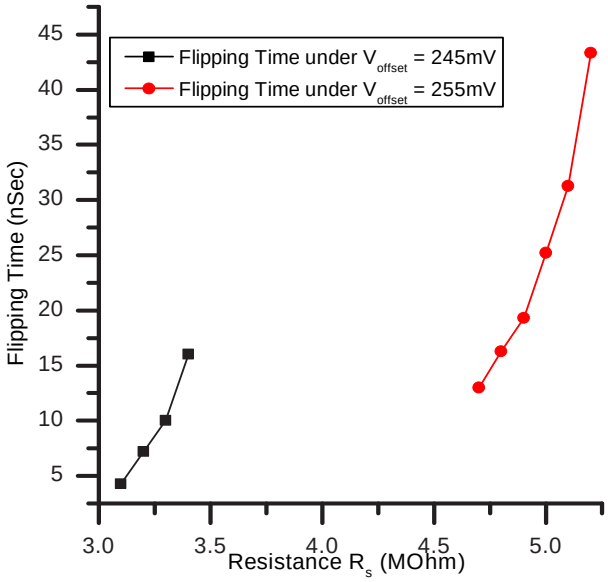


Figure 3.22 Dependency of flipping time on the series resistance for different offset voltages

will remain on the same side of the saddle associated with V_r, no matter how long the cell is accessed. Therefore, it is considered as an un-faulty cell. However, if the cell is statically d-instable it means that it only has one equilibrium point, M, in the state-space. The state of the cell is attracted to that point during the access time, no matter what the initial (i.e. non accessed) logic state is. Assuming M to be on the attraction region of the logic state opposite to the initial logic state, the cell will change its state during the access time, if the access time T_a is long enough. In other words, under long enough access time, the logic state of the cell leaves the region of convergence of the original logic state to end up at M. When the cell becomes non-accessed, the logic state of the cell is changed. Therefore the ultimate logic state of the cell is determined by the location of M rather than the original state of the cell. Hence, by increasing the access time, the marginalities of the accessed cell can be detected. The type and amount of fault determines the required access time to

move the state of the cell to M since the type and amount of fault affects V_a, the function that determines the dynamic behavior of the cell.

Simulation results confirm the idea as shown in Figure 3.23. The simulation setup is such that the cell is d-stable in the non-accessed mode and statically d-instable in the accessed mode and the DC solution of the accessed cell is in the opposite side of the initial condition of the cell. This is the case in the examples presented in subsection 3.6.1. It can be seen that for a constant R_s , the extension of the duty cycle of the cell access time increases the possibility of detecting a marginal cell by reducing the flipping time. Also, the minimum detectable R_s increases which increases the scope of the test toward higher R_s values. Also, as discussed in section 3.6.1, if the cell access time exceeds the flipping time of the cell it can detect the marginalities associated with excessive threshold voltage fluctuation of the access transistors and internal inverters. These faults, unlike the marginalities associated with R_s, produce significantly shorter recovery time during the non-accessed mode. Hence, if the access time does not exceed the flipping time, it allows a full recovery and prevents the detection of the statically d-instable cells. Therefore, repetitive access does not make a significant improvement in detection of such faults even with higher duty cycles.

It is evident that an increase in the cell access time can never cause a statically d-stable cell to flip during the access time. Therefore, marginal cells can be distinguished from the statically stable cells using this method without over reacting against the d-stable cells with minor degradation. This feature, distinguishes the proposed scheme from other schemes such as bitline/power supply voltage variation schemes which are subject to over reaction against statically d-stable cells.

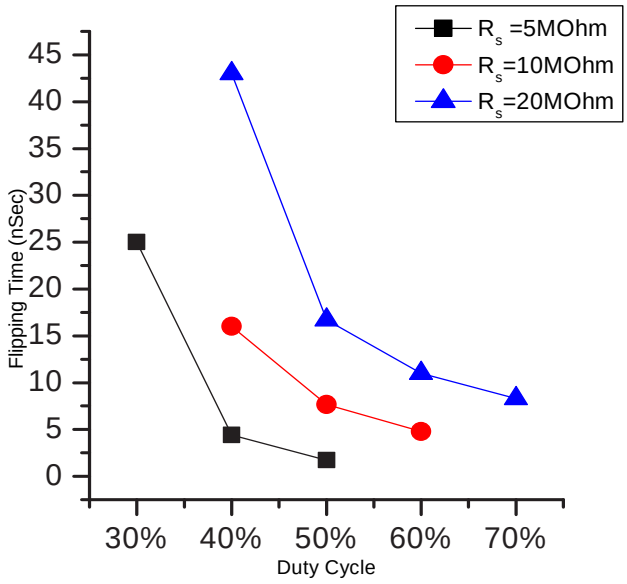


Figure 3.23 The flipping time dependency on the cell access time duty cycle for different Rs values

3.7 Summary

Traditional approaches of d-stability of SRAM cell are functions of static parameters such as V_{DD}, V_T, and static noise sources, etc. In this paper chapter, we introduced the concept of dynamic d-stability where it is shown that it is function of static as well as timing parameters. The dynamic behavior of the cell is especially important when the cell is accessed over a short period of time. It is shown that the state space equations associated with the SRAM cell can describe the dynamic behavior of the cell during access and retention conditions.

A new criteria of dynamic d-stability is introduced for SRAM cells. This criteria is based on the existence of at least two periodic solutions for a periodically accessed SRAM cell. These periodic solutions must remain in the region of convergence of each of the

UAS points of a non-accessed SRAM cell. It is shown that the conventional criteria of static d-stability is a limiting case of the proposed dynamic d-stability criteria. In addition, sufficient conditions for dynamic d-stability criteria are presented and a simulation method to verify these conditions is outlined.

For a given static noise source capable of violating the criteria of static d-stability, there exists a corresponding SNM. Similarly, for the proposed criterion for dynamic d-stability, we defined the associated static noise margin. For a static noise source, SNMD is defined as the amount of static noise that if applied to the cell, it violates the dynamic stability criteria once the cell is accessed periodically.

The concept of dynamic d-stability can be exploited in the design of low power SRAMs. The SRAM cell static and timing parameters can be traded to achieve low power, and other desirable features while maintaining the cell dynamic d-stability.In particular, it is shown that with the proper choice of the access time and recovery time the SRAM cell remains data stable even if it is data unstable from static perspective. This feature adds to the noise margin headroom of the cell and allows the further reduction of the supply voltage of the cell which results in corresponding reduction of the leakage current. Silicon measurement results in $130nm$ CMOS technology confirms the concept of dynamic d-stability and designer's ability to trade timing and static parameters. Moreover, it is shown that the low time constant due to the subthreshold operation of the cell can be exploited to maintain data stability with proper choice of access and recovery time.

Chapter 4

SVGND Architecture and Comparison

This chapter introduces the Segmented Virtual Grounding Architecture. Section 4.1 gives an introduction to the chapter. Section 4.2 gives a short review of SRAM low power techniques. Subsequently, in Section 4.3, the architecture of the memory unit and different cell operating modes with respective operational voltage settings are described. Section 4.4 compares the scheme with the recently reported schemes and finally section 4.5 concludes the chapter.

4.1 Introduction

Embedded memories are vital components of the system-on-chip (SoC) solutions. In general, embedded memories are implemented with static random access memories (SRAMs) owing to their speed, robustness, and ability to be integrated in a logic process. Embedded SRAMs occupy increasingly larger die area, and therefore, it influences various aspects of

the SoC such as power, area, and yield.

With increasing demand for battery operated applications, methods for reduction of the power consumption of the memory blocks have received significant interest. Six transistor SRAM cell is preferred for many applications because of its high speed and robustness. However, with the increased SRAM sizes, the leakage is becoming a growing concern. Besides, the leakage current increases with technology scaling, hence DC power minimization has become a priority and been addressed by innovative solutions [43, 25]. In addition, the power consumption of the write operation is high because of the high bitline swing requirement. To overcome this problem, several methods have been proposed in the literature [25, 35, 23].

A new scheme to reduce the power consumption of static random access memories (SRAM) is presented. It is shown that using segmented virtual grounding (SVGND), it is possible to reduce both dynamic and static power consumption. The leakage power of the cells is reduced by reducing the voltage drop over a cell. The dynamic power dissipation is also reduced by eliminating the power consumption due to the discharge of the non-desired neighboring bitlines. The effectiveness of this scheme is compared to recently reported low-power schemes. It is shown that unlike those schemes, SVGND can accommodate multiple words in one row; a significant improvement in soft error rate tolerance [44].

4.2 SRAM Power Reduction Techniques

There have been several attempts to reduce the leakage, write and read power for SRAMs. Itoh provided an excellent review of low power SRAM design strategies [43]. Virtual grounding is a well known technique to reduce the write power consumption. SAC scheme [25] and its predecessor [45] have reduced the write power consumption significantly by

floating the source line of the cells on the same row and reducing the required bitline variation to $V_{dd}/6$. This technique, however, drives the non-accessed neighboring cells on the shared source line to an unstable floating region during the write operation, destroying the data of the words located on the same row. This effect limits the application of this technique to high bitwidth applications in which one row represents only one word. A similar write power reduction method has been reported in [35] with the source line of the cells on the same column are shared instead of the source line of the rows, saving the power consumption of the neighboring bitlines. However, the power consumption due to the swing of the internal nodes of the cells in the accessed column increases the read power consumption significantly.

Reduction of the bitline precharge voltage reduces the power dissipation in both read and write operations. This method has been utilized in [46] at the expense of noise margin degradation in read and write operations and some design overhead. In this scheme, the bitline voltage swing is reduced to $V_{dd}/2$. Recently, another write power reduction scheme has been reported in [24] which is based on hierarchical bitline and local sense amplification. In this scheme the voltage swing of the bitlines for the write operation is reduced to $V_{dd}/10$ at the expense of two additional full swing control signals in the array that run in parallel to the wordline. In this scheme both read and write operations require switching of these signals. Aside from complexity and area overhead, having additional full-swing control signals increases the power consumption significantly, especially if more than one word is stored in a row. In addition, this configuration is destructive to the neighboring words in the same row in write operation. Therefore it can be categorized as a single word per row memories similar to [25].

Stand-by power consumption is another important source of power consumption in large scale CMOS SRAMs. Leakage current which is the main cause of the stand-by power

consumption increases as the technology scales. Kanda et. al described a sleep mode where the supply voltage of all non-accessed cells was decreased in order to reduce the leakage current [25].

This chapter presents a segmented virtual grounding (SVGND) scheme that is capable of saving power consumption in both dynamic and static operating regions. From dynamic power perspective, it saves the power consumption in both read and write operations. In this scheme, the virtual grounds of the cells in one segment is shared and controlled using an extra switch for that segment (see Figure. 4.1). The virtual ground of a segment is lowered to the ground only if one of the cells in the segment is accessed for the read operation. On the other hand, the cell retains the data while all internal transistors operate in the weak inversion region. Keeping the cells in the weak inversion region reduces the static power consumption by decreasing the leakage current. In this scheme, neighboring bitlines which are associated with neighboring words on the same row, are not discharged in every read or write operation. This schemes introduces a new state to the memory cells in addition to the conventional read, write and retention states. The new state is called "accessed retention mode". Using this scheme the voltage swing for write operation is reduced providing a low-power write operation.

4.3 SVGND Architecture and Operational Modes

This section describes the architecture of an SRAM unit based on SVGND scheme. First, the SRAM cell level circuit issues and operational modes are discussed. Next, the architecture of the SRAM unit in terms of segmentation and control signals is discussed and the operation of the unit under read and write operation is explained.

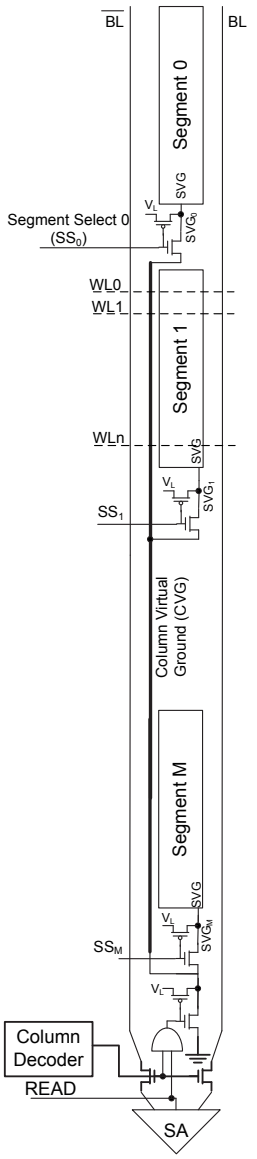


Figure 4.1 The schematic of a column based on Segmented Virtual Grounding (SVGND)

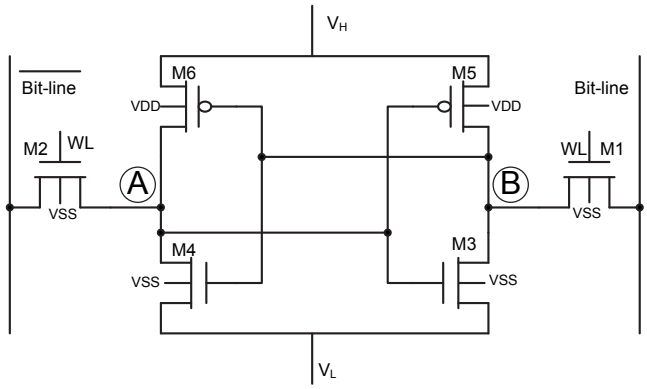


Figure 4.2 Nominal cell operational voltages in SVGND scheme

4.3.1 SRAM Cell Issues and Operational Modes

Figure 4.2 shows the voltage setting of a SRAM cell in data retention mode. The body of the PMOS load transistors are connected to V_{dd} the highest voltage available on the chip while the body of the NMOS drive transistors and access transistors are connected to the ground, V_{ss}. When a cell is not accessed, the wordline is connected to the ground, V_{ss}. The source of the PMOS transistors are connected to V_H, the high voltage of the cells; and the source of the drive transistors are at V_L, the virtual ground voltage of cells. In other words, V_H represents the logic '1' and V_L represents the logic '0' in a cell. Since V_H and V_L are not equal to the respective body voltage of transistors, all transistors are reverse body biased. Simulation results demonstrate that the threshold voltage, V_{th}, of transistors can be increased significantly with reverse body bias. Assuming the voltage across a cell, $V_H - V_L$, is close to V_{th} and is approximately one third of the nominal supply voltage, the V_{gs} over the gate source of the drive and load transistors is three times smaller than the V_{gs} of the same transistors in the conventional configuration. The higher threshold voltage due to the body effect decreases the cell leakage current significantly. The leakage

current of the access transistors is also reduced provided that the precharge voltage of the bitlines are at V_H which results in negative V_{gs} over both access transistors. According to [47], these reductions are due to the fact that the leakage current has an exponential relationship with V_{gs} and V_{th} as shown in Equation. 4.1:

$$I_s = I_0.e^{(V_{gs}-V_{th})/nV_T}(1 - e^{-V_{ds}/V_T}) \tag{4.1}$$

where $V_T = kT/q$ and $I_0 = \mu_0 C_{ox}(W_{eff}/L_{eff})V_T^2 e^{1.8}$. Figure 4.3 shows the relationship between overall leakage current and the cell's effective voltage when both V_H and V_L digress from mid rail voltage(*i.e.*, $V_{dd}/2$) towards V_{dd} and V_{ss}, respectively. The simulation results are based on a $130nm$ CMOS technology with nominal threshold voltages of 0.3V($V_{bs} = 0$) for typical cell transistor sizes at 25°C. It can be seen that the leakage current increases exponentially when the effective voltage over the cell exceeds one V_{th} of the transistors (*i.e.*, 0.4V when it is affected by body effect) as predicted in [11]. Therefore, the nominal voltage drop over the cell is chosen to be close to V_{th}, keeping the drive and load transistors in weak inversion region. The second curve in the figure depicts the static noise margin (SNM) with respect to the effective voltage across the cell. It is apparent that the SNM degrades gracefully, and at 400 mV it is approximately 220 mV for this cell which is adequate to retain the data in data retention mode.

The nominal wordline voltage that activates the access transistors is $V_{wl} = V_H + V_{th_a} - V_\Delta$, where V_{th_a} is the nominal access transistor's threshold voltage and V_Δ is a fraction of threshold voltage bigger than the SNM of the cell under retention mode. When the access transistors are activated, the cell goes to the *accessed retention mode*. In this mode, the source of the drive transistors are still connected to V_L. However, since the drive transistor does not have sufficient overdrive voltage, the access transistor will dominate in setting the internal voltages of the cell. Such a scenario may give rise to cell unstability. This

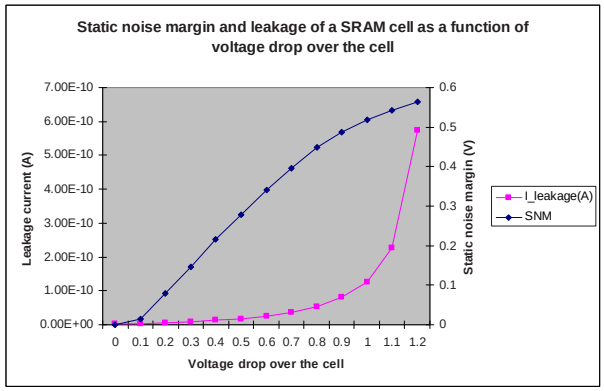


Figure 4.3 Leakage and SNM as functions of voltage across the cell

situation is taken care of by choosing appropriate V_{wl} as explained above.

The voltage setting of a cell when it goes to the accessed retention mode can be explained using Figure. 4.2. In this figure, node A carries a zero and node B carries a one. Since node B carries a one (*i.e.*, $V_B = V_H$) V_{gs} of M1 is less than V_{th_a} when the cell is accessed. Therefore, M1 remains off and has no effect on node B. Node A however, raises to $V_H - V_\Delta$ where V_{gs} of M2 is equal to V_{th_a} and M2 turns off. Therefore, all six transistors remain off when the cell goes to the accessed retention mode, leaving V_Δ voltage difference between nodes A and B. This is due to the fact that under nominal voltage setting the overdrive voltage of M4 is too small to compete with M2 when M2 is on($V_{gs} > V_{th}$). So M2 lifts up the voltage of node A until this voltage reaches to $V_H - V_\Delta$ where M2 turns off($V_{gs} \leq V_{th}$). It is known that as long as the initial differential voltage between node A and node B is higher than the offset voltage between the two internal inverters, the cell can recover its original state [29]. When the access transistors are deactivated, the V_{gs} of M4 is $V_H - V_L$ while V_{gs} of M3 is 0 which helps the node A to come back to V_L. The recovery time of the cell is directly related to the effective voltage across the cell ($V_H - V_L$). In

practice the duration of activation of the wordline is small and charge injection and clock feed through due to the activation of the access transistors are small and are compensated when the cell is deactivated. Obviously, a cell in the accessed retention mode will not discharge the bitline.

The dynamic data stability concept introduced in Chapter 3 ensures the cell data stability during the accessed retention mode. All six transistors go to the subthreshold region during the access time. If the cell is statically data unstable during the access time because of non-idealities, it means that there is only one DC solution for the state of the cell. However, the subthreshold operation of the transistors extends the time that is required for the state of the cell to reach to the DC solution. Therefore, the state of the cell does not move away from the region of attraction of the original logic equilibrium state before the cell goes non-accessed. This condition offers periodic solutions that suffice the data stability criteria and can be investigated for a cell using circuit simulations.

4.3.2 Segmented Architecture Implementation

The SVGND architecture was implemented using $130nm$ CMOS technology. The architecture of the SRAM array is based on segmentation of the memory cells in a column. Figure 4.4 shows a segment. In this figure, wordlines are removed to avoid cluttering. A segment is defined as a set of N cells on the same column with a shared segment virtual ground (SVG). A virtual ground switch connects the virtual ground of the segment to the virtual ground of the column (CVG). The CVG is a node shared between all virtual ground switches on the same column (see Figure. 4.1). If the virtual ground switch of a segment is activated by turning signal SS to high voltage, the virtual ground voltage of the segment is equal to the virtual ground voltage of the column; otherwise, the virtual ground of the segment keeps its nominal voltage, V_L. The logic of the virtual ground switch is an inverter

which drives the SVG node either to V_L or to the voltage of CVG node depending on the control signal, SS.

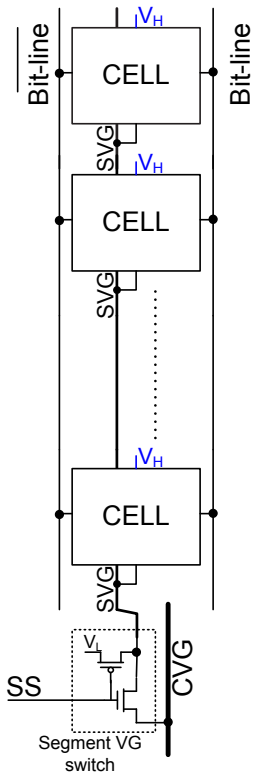


Figure 4.4 The architecture of one segment

Figure. 4.5 depicts an SRAM architecture with the SVGND scheme. In this figure, wordlines and PMOS transistors of the virtual ground switches are removed for clarity. The activating signal for the virtual ground switch or segment select signal (SS_i) is generated by the predecoder of the row decoder at no additional hardware cost. If a cell is accessed for either read or write operation, the (SS_i) signal of that segment is also active. On the other hand, the column virtual ground(CVG) is lowered to V_{SS} only if one of the cells on the column is to be *read*. CVG of other columns remains at V_L under read operation.

Lowering the CVG of a column allows the virtual ground of the accessed cell on that column to be lowered to V_{SS}. When the virtual ground of a cell is pulled down to the ground, the drive transistors become stronger because of both body effect elimination and V_{gs} soars. However, other cells on the same row and in the neighboring columns will go to the accessed retention mode.

4.3.3 Operational Modes

Figure. 4.7 shows the conceptual waveforms of different nodes under the read, write, and accessed retention operational modes. Wordline and SS signals come at the same time since both of them are generated through the same decoder. The SS signal controls the SVG node, while CVG of the column that is to be read is discharged in advance. This signal is generated by performing a logical AND between the read and the decoded column address. Signals A and B represent the internal nodes of the cell.

Read mode: In the read mode, the CVG signal is pulled down before the selected wordline, and SS signals are asserted. While the wordline is high the SVG is low, the selected SRAM cell discharges the appropriate bitline. During this interval, the voltage across the cell is V_H, therefore the cell is capable of discharging the bitline. A selective discharge of the SVG node to ground of only one segment in the array during the read operation prevents the discharge of both internal capacitances of the neighboring cells on the same row and the internal capacitances of the non accessed segments on the same column. Therefore, it saves a significant amount of power compared to the previously reported schemes ([45], [35]) that discharge the internal cell capacitance of the complete row or column during the read operation. The virtual ground capacitance including the internal capacitances of a cell that needs to be (dis)charged is estimated to be 3-4 times higher than the bitline capacitance per cell. This situation can be further explained with

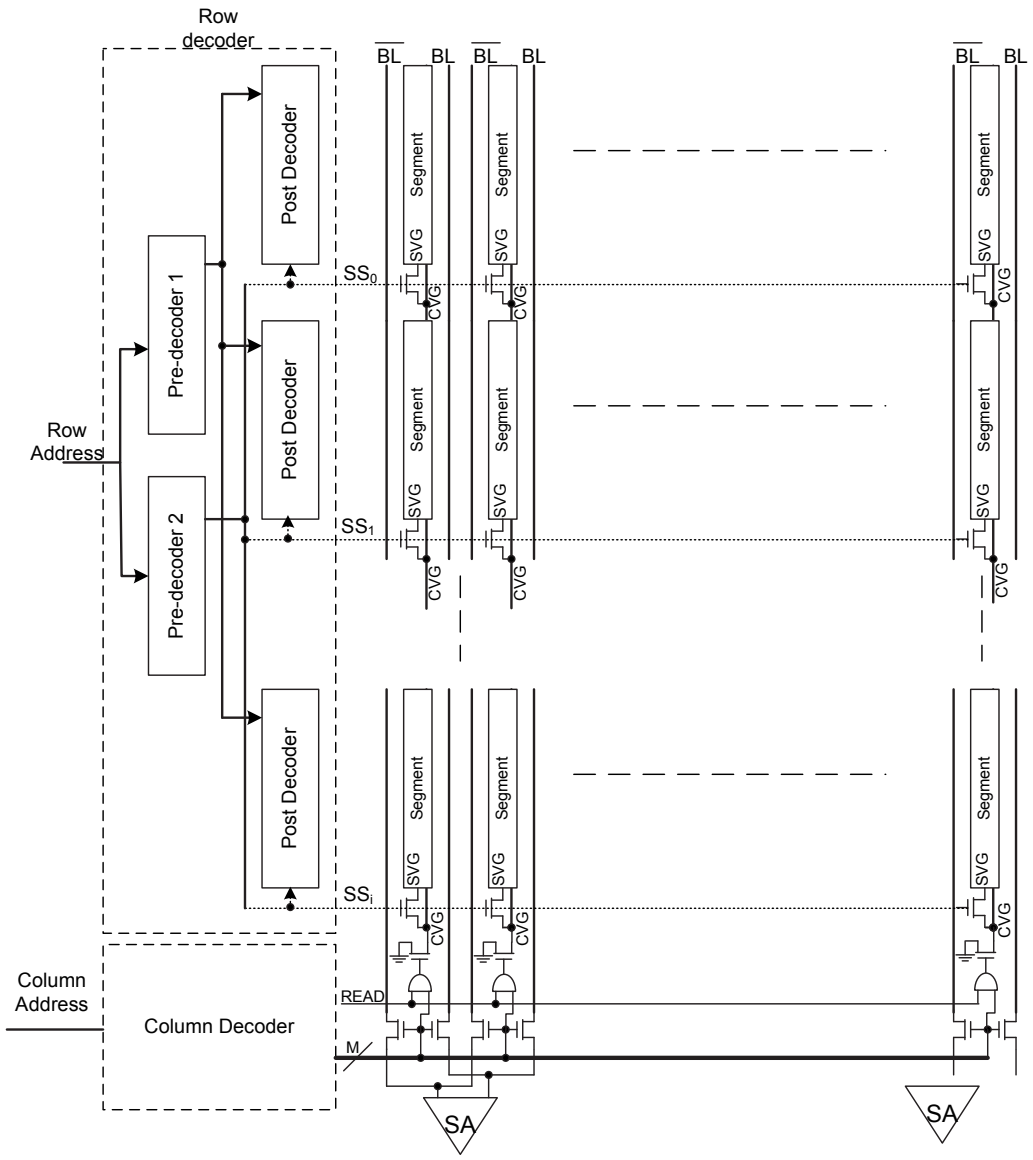


Figure 4.5 SVGND architecture of an SRAM

the help of Figure. 4.6 which depicts different capacitances affected by the change of virtual ground voltage of a cell. Therefore, a circuit technique that prevents (dis)charge of the

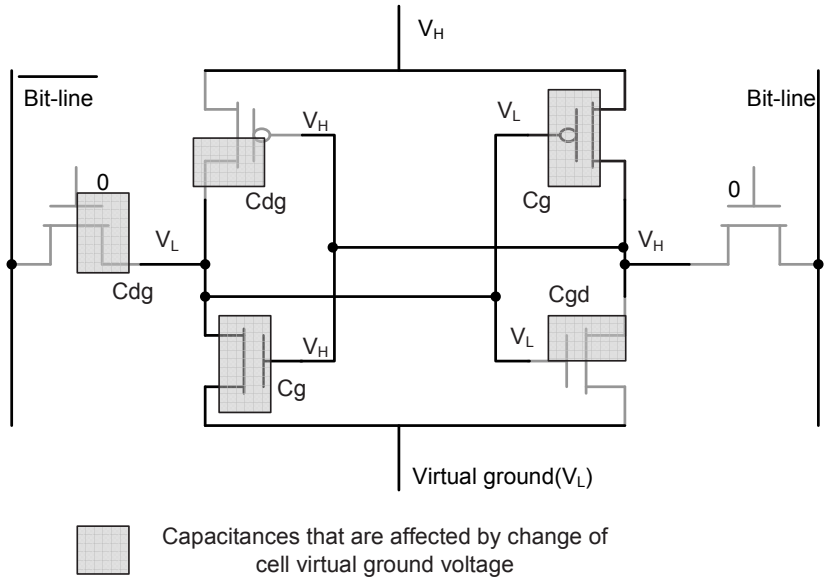


Figure 4.6 Internal capacitances of a cell affected by variation of the virtual ground voltage

virtual ground capacitance during the read operation is likely to save significant amount of power.

Accessed retention mode: The figure also exhibits the internal node voltages of a neighboring cell on the same row and different column which goes to the accessed retention mode with the assertion of wordline voltage. It can be seen that in this mode all transistors are in the weak inversion region,*i.e.*, $V_{gs} < V_{th}$.

When the wordline is activated to access a desired cell, the cells on the same row and non selected columns are kept in the nominal voltage condition. Therefore, these cells go to the accessed retention mode. As mentioned earlier, a cell can not discharge its corresponding bitline if it goes to the accessed retention mode. Thus, the power consumption is reduced compared to the power consumption of the recently reported low-power schemes of [25] and [24] in case they are to implement a multi-word per row architecture. In addition,

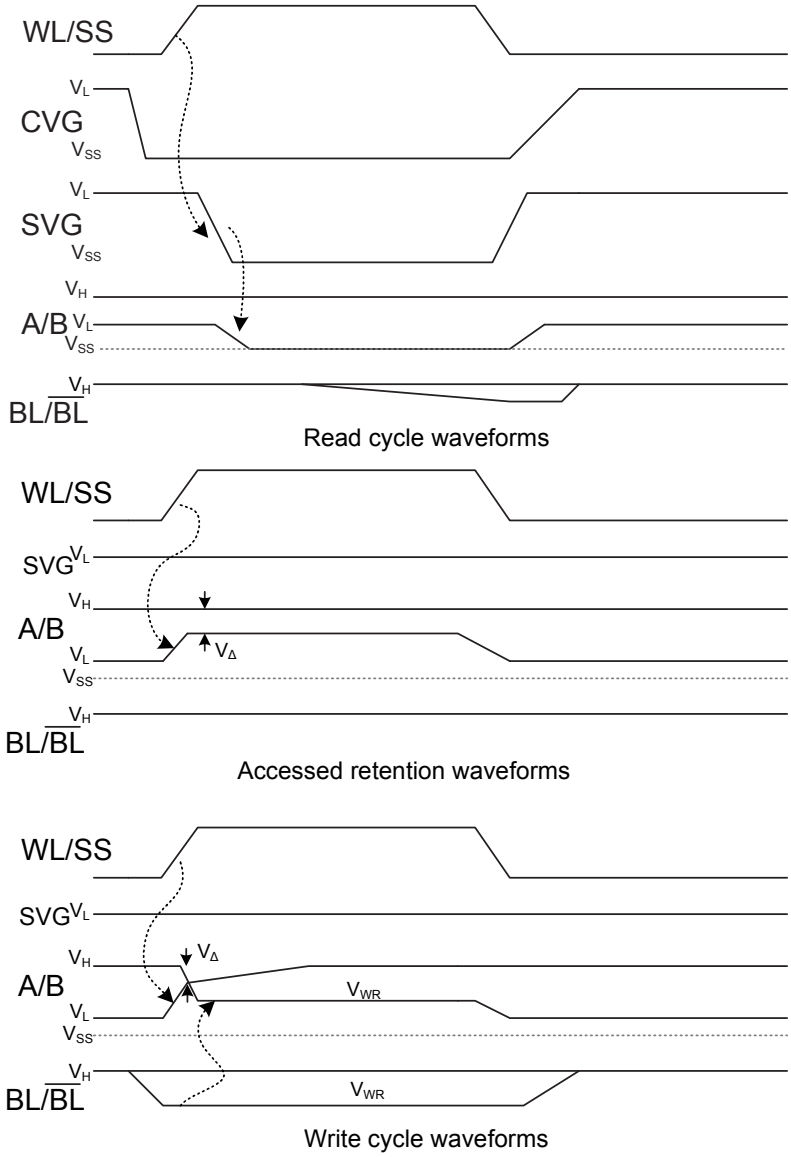


Figure 4.7 Time domain waveforms in read, write and accessed retention modes

SVGND requires fewer control signals running in parallel to the wordline on the array compared to [24] at the expense of a column virtual ground which is a low voltage swing

line at a low capacitive load higher level metal layer. Having a low-voltage, multi-word per row configuration has the additional benefit of higher tolerance to soft errors compared to low voltage, single word per row scenarios. Distributing the word over the row reduces the number of soft error related bit errors in a word by a single radiation event. In addition, serving multiple words in a row saves the power consumption of the block decoders that need to drive all the gates in the post decoder stage for a regular bit-width applications like [24].

Write mode: Low power write operation is another benefit of the scheme. Write operation can be explained based on Figure. 4.2. In write operation, corresponding bitline is discharged to V_{wr} before the wordline high signal is asserted. When a cell goes to the accessed retention mode, the voltage on node A is at $V_H - V_\Delta$ (logic '0'), and the voltage of node B is V_H (logic '1'). Under this condition both access transistors are off. To write on to the cell, it would be enough to pull node B to a voltage that is sufficiently below $V_H - V_\Delta$, called V_{wr}. Such a voltage can easily turn on the access transistors (*i.e.*, $V_{gs} > V_{th}$) and discharge node B to V_{wr} because the PMOS load transistor of M5 is in weak inversion region and can not overcome the access transistor M1. Theoretically, if the voltage difference between nodes A and B(*i.e.*, $V_H - V_\Delta - V_{wr}$) is large enough, the cell can recover the full voltage swing at its internal nodes after the access transistors are deactivated. Simulation results show that bitline swing as low as $V_{dd}/4$ from the precharged voltage of V_H can result in a successful write operation under worst case offset condition between the cell inverters. It is evident that unlike conventional write operation the neighboring bitlines are not discharged, this adds to the power saving capability of the scheme.

Figure. 4.8 illustrates the post layout simulated waveforms when the cell is accessed for read, write and accessed retention mode. In this figure, V_H and V_L are chosen to be 0.9V and 0.5V respectively. Since the wordline voltage of $V_{wl} = 1.2V$ is chosen, the resulting V_Δ

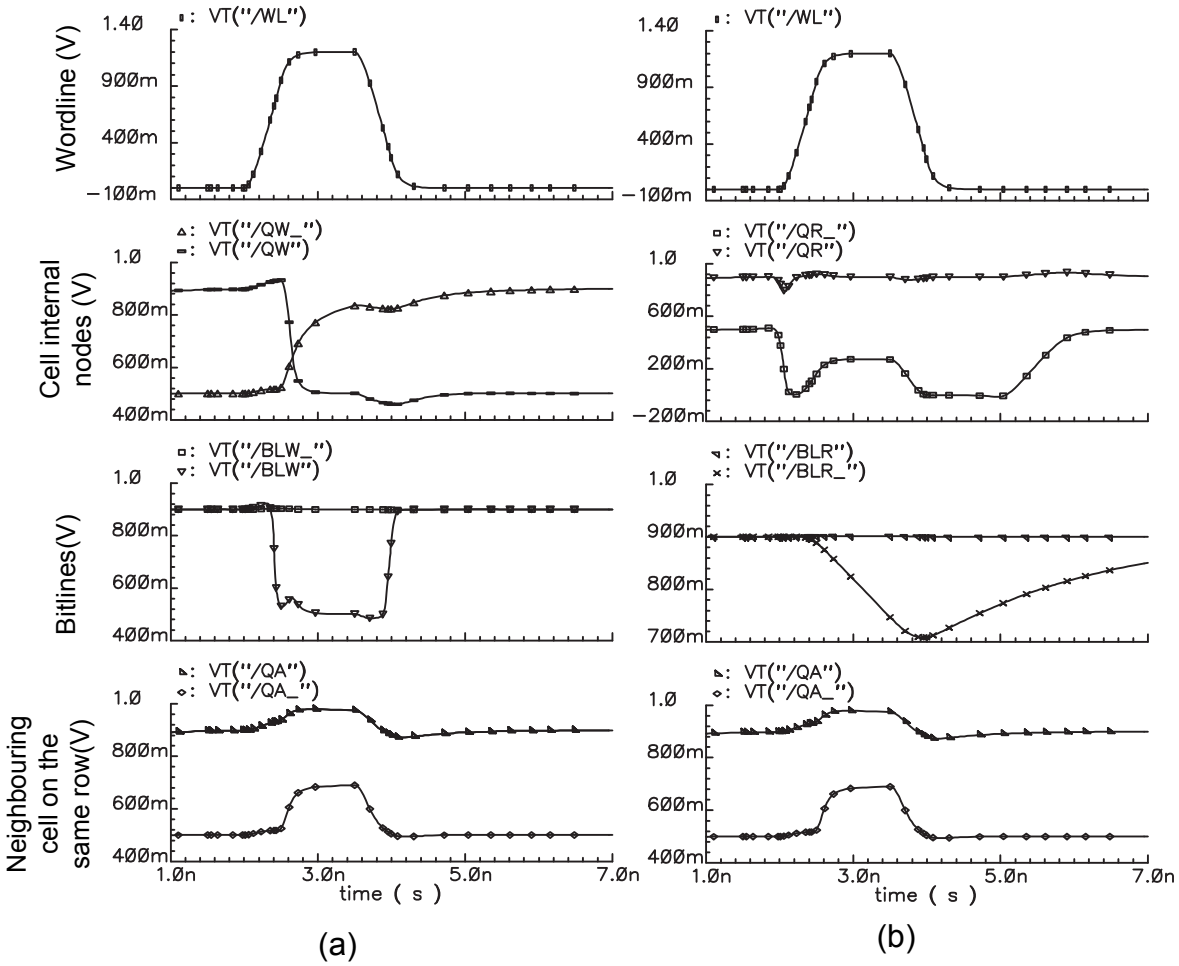


Figure 4.8 Spice simulations for write (a), read (b) operations and internal node voltages for an SRAM cell in accessed retention mode (both a and b; bottom waveform)

of 0.2V appears on the internal nodes when the cell goes to the accessed retention mode. In this figure, V_{wr} is 0.6V which is sufficient to flip the cell's state.

4.4 Comparison

There have been a number of low power SRAM schemes reported in the last several years [35], [46], [25], [24]. In particular, we are interested in comparing our work to that of recently reported works of Hierarchical Bitline Sense Amplifier (HBLSA) [24], and Sense Amplifier Cell (SAC) [25]. Kanda et. al. carried out an excellent comparison between their circuit technique with that of [35], [46]. Therefore, by comparing this work to that of Kanda, we are implicitly comparing our work to [35] and [46]. In addition, we also compare all these schemes with the conventional scheme (CONV) as a reference. The comparative analysis is carried out in terms of dynamic power consumption, and static power consumption as well as other design implementation aspects.

4.4.1 Dynamic power comparison

In this section, the array dynamic power consumption of the SVGND, HBLSA, SAC and conventional schemes are compared. The comparison is based on a $130nm$ CMOS technology with the nominal power supply voltage of 1.2V.

Dynamic power consumption in SRAMs is segregated in several parts specially if the size of the memory is big [24], [46]. Although the control unit including the dummy circuit, and the decoders take a large fraction of the overall dynamic power consumption, the power consumption of the array is still substantial. In medium size, non block oriented SRAMs, the array power consumption is the dominant component.

In this comparison it is assumed that a single array is used to implement a medium size SRAM unit. It is assumed that the unit has 2^W words with each word having B bits. It is further assumed that all three schemes are capable of implementing multiple words per row configuration, even though both [25] and [24] put severe implementation restrictions

because data of the complete row is destroyed during the write operation. It is evident that the array power dissipation of a full word read or write operation is equal to the power dissipation of a single bit operation times B. Therefore, the comparison can be made on *per bit* basis. It is assumed that M number of bits reside on a per bit row (*i.e.*, M words are interleaved on the same row). Typical transistor size for a conventional SRAM cell is chosen for all three schemes. This choice provides approximately the same overall speed for all schemes given the same bitline capacitive load. The parasitic capacitance of different nodes in this analysis are obtained using careful layout extraction in all three cases.

The per bit array energy consumption can be expressed as the sum of the per bit energy consumptions on the active nodes of the array in each of read and write operation:

$$E_{total} = E_{wl} + E_{bl} + E_{bl_a} + E_{control} + E_{cell} \tag{4.2}$$

All energy terms are in per bit unit in this equation. E_{wl} is the wordline energy consumption which is equal to $M.C_{wl}.V_{dd}^2$ where C_{wl} is the capacitance associated with the wordline capacitance of a memory cell. $E_{control}$ is the energy associate with the additional control signals activity; the signals that run in parallel to the wordline to activate a switch or a component on the array for different schemes. In SVGND and SAC, this term is equal to the energy consumption of the virtual ground switches (VGS). This switch is repeated for every column(*i.e.*, M switches for each desired bit) in SVGND whereas in SAC one ground switch is shared among a number of cells. The number of cells with a shared VGS in SAC is assumed to be equal to 8 which is equal to the number of cells in a segment, N, in SVGND. This choice makes the area overhead to be similar and approximately 6-8% for all low-power schemes. The energy consumption of the CVG in each read operation adds to the control energy dissipation in SVGND. The capacitance associated with CVG is rather small as it runs in a higher level metal layer and the voltage swing is only V_L;

$E_{CVG} = C_{CVG}V_L^2$. The control signals in HBLSA scheme consists of the access transistors switch of the local sense amplifiers and the sense enable signal for each local sense amplifiers that is repeated on every column. The voltage swing on these nodes are equal to V_{dd}. E_{cell} is the energy consumption due to the internal nodes (dis)charge of a memory cell (see Figure. 4.6). These capacitances are accessible through the virtual ground and are charged(or discharged) when the supply voltage of the cell varies. In SVGND, the number of cells that go under virtual ground voltage variation is equal to N in read operation. This is equal to M in SAC in read operation in which all cells in a row gain a high supply voltage. In HBLSA, M local sense amps with the same size of a memory cell are active during both read and write operations. The activity of local sense amplifiers involves its internal node variation. Internal node voltage variation is only V_L in SVGND whereas it is V_{dd} in HBLSA and $0.8V_{dd}$ in SAC.

The energy dissipation on the bitlines is an important source of overall energy consumption. E_{bl} is the energy consumption of the bitline on which the desired cell which is to be read or written on is located:

$$E_{bl} = R.C_{bl}.V_{pre}.V_{\delta} \tag{4.3}$$

In this equation, R is the number of rows, C_{bl} is the bitline capacitance per cell, V_{pre} is the precharge voltage of the bitline and V_{δ} is the voltage fluctuation in each operation. C_{bl} in HBLSA is 0.8 times the nominal C_{bl} because of the hierarchical bitline architecture. V_{pre} in SAC and SVGND is approximately equal to $V_{dd} - V_{th}$. E_{bl_a} is the energy consumption of the neighboring bitlines. This is equal to $M - 1$ times E_{bl} for SAC and HBLSA and zero for SVGND.

Per-bit array energy consumption is calculated for different array sizes. Figure. 4.9 compares the energy consumption of different arrays implemented using conventional, SAC,

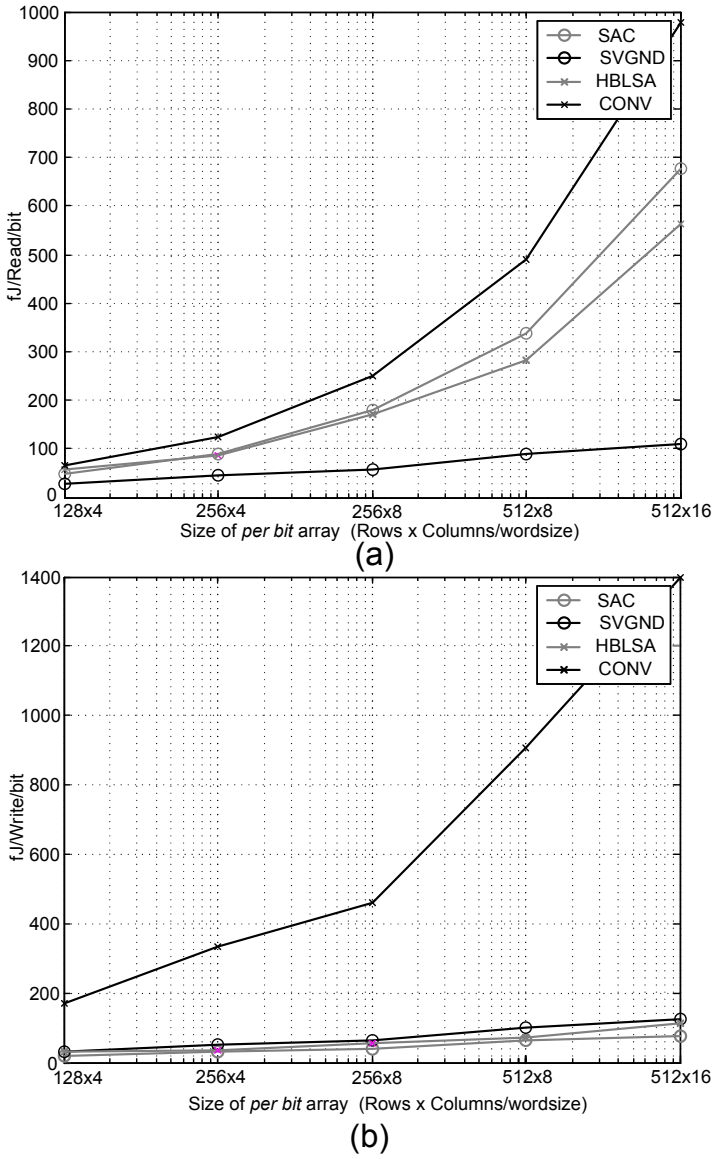


Figure 4.9 Power consumption comparison among different schemes for read (a) and write (b) operation.

HBLSA and SVGND schemes. X axis represents the size of the memory in per bit unit. For example, in this figure data point 256x4 represents 256 rows each with 4 interleaved

words. Therefore, energy computations can be broken down to per bit basis, *i.e.*, each interleaved bit. One can inspect the overall array size by multiplying the per bit size by the bitwidth, B. In each step the size of the memory is doubled by doubling the number of addresses to be served by the array. Doubling the size of the memory is realized by doubling the number of rows or columns on every step. For example in the first step the number of rows is equal to 128 and the number of, per-bit columns is 4 (which is equal to the number of words that are accommodated on the same row). In the second step the number of rows increased to 256. In the third step the number of columns is 8 and the number of rows is 256. In this figure, Y axis represents the energy per bit per operation. Therefore one can figure out the total power consumption by multiplying a Y axis value by the bitwidth (*e.g.*, B=32) and the frequency of operation.

It can be seen that the read energy consumption increases substantially as the size of memory increases in conventional, SAC and HBLSA. However, the SVGND, presents a lower energy consumption as the size of memory increases. For example in the third step, where the array consists of 256 rows and 8 words per row, SVGND reduces the conventional read energy consumption by 74%. This is because increasing the number of words on each row has minor effect on the read energy consumption in this scheme: it merely doubles the capacitive load of the wordline and SS wires. The energy consumption of these signals on per bit basis is minor compared to the energy consumption associated with the bitline discharge. All other schemes (Conventional, SAC and HBLSA), on the other hand, discharge non desired bitlines in read operation. In addition, the precharge bitline voltage , V_H, is lower in the SVGND scheme relative to V_{dd}-the precharge voltage in conventional and the HBLSA scheme.

The read energy efficiency of the scheme can be further justified by referring to equation 4.2. In SVGND the term E_{bla} which is $M-1$ times higher than E_{bl} is eliminated because

the non-desired columns are not discharged. Furthermore, the rest of the terms in that equation are small because they are associated with the short wires running horizontally in the per-bit array. Therefore, SVGND can reduce the energy consumption significantly by elimination of the most significant term of the equation E_{bla}. This power reduction becomes more clear as the number words per row increases.

Table 4.1 Different energy components(fJ) during read operation

Scheme	E_{wl}	E_{bl}	E_{cell}	$E_{control}$	E_{total}
SVGND	9.3888	19.9680	1.7434	12.67	56.06
SAC	9.3888	159.7440	6.9734	2.8080	178.9142
HBLSA	9.3888	130.7520	15.6902	15.1488	170.9798
CONV	9.3888	239.6160	0	0	249.0048

Table 4.1 represents different energy components described in Equation 4.2 during a read operation and compares read energy components in different schemes. This table takes a snap shot of the third data point (256x8) of Figure 4.9. In this table, CVG energy consumption in the SVGND is not explicitly mentioned in a column, since it is unique to this scheme. However, it is taken into account while calculating the total energy. The bitline energy consumption includes both the desired bitline power consumption and the non-selected bitline's power consumption. The E_{cell} requires some additional explanation. As mentioned before this energy segment includes the energy associated with the cell internal node voltage variations due to the voltage variation of the virtual ground of cells as well as that of the distributed sense amplifiers in HBLSA scheme. Since there is no voltage variation in the ground of the conventional scheme, this energy component is zero in that scheme. On the other hand, this component is relatively high in the HBLSA scheme due to the distributed local sense amplifiers in the array with full swing voltage variation

of the ground. It is evident that SVGND saves a substantial read energy consumption compared to all other schemes.

Table 4.2 Different energy components(fJ) during write operation

Scheme	E_{wl}	E_{bl}	E_{cell}	$E_{control}$	E_{total}
SVGND	9.3888	39.9360	0	12.6720	61.99
SAC	9.3888	26.6240	1.7434	0	37.7562
HBLSA	9.3888	16.3440	15.6902	15.1488	56.5718
CONV	9.3888	449.2800	0	0	458.6688

Figure. 4.9 illustrates the write energy consumption/bit for different array sizes. It is evident from the figure that compared to the conventional scheme, all other schemes are very energy efficient. In these schemes, the circuit techniques enable a low swing write operation which saves significant amount of write energy. Furthermore, SVGND allows elimination of the bitline power consumption associated with the non desired columns which adds to the benefit of this scheme.

Similar to the Table 4.1, Table 4.2 reveals the write power components in different schemes. This table takes a snap shot of the third data point (256x8) of Figure 4.9. All schemes has the same E_{wl} consumption because of the same wordline load. The bitline energy consumption differs significantly. The conventional scheme has the largest energy consumption due to full swing bitline voltage variation. The rest of the schemes allow low bitline voltage swing write operation, therefore, their respective E_{bl}s are significantly smaller. The HBLSA scheme offers the minimum E_{bl} because of lower V_{wr} and reduced C_{bl} due to bitline segmentation. In the SVGND scheme, the E_{cell} is insignificantly small since there is no internal voltage variations for non selected cells in the same row. Similarly, in SAC scheme, internal voltage variation is very small. In the HBLSA, all local sense

amplifiers in the desired row have a full swing operation therefore it has largest E_{cell}. The conventional scheme does not require any additional control signal, therefore, $E_{control}$ is zero. Similarly in the SAC scheme, during the write operation the control signal is inactive, therefore its $E_{control}$ is also zero. In the HBLSA scheme there are two control signals compared to one in the SVGND scheme, therefore its $E_{control}$ is relatively higher.

It can be seen that in write operation, SAC outperforms the other schemes. This is due to its rather fewer number of control signals, smaller voltage variation on bitlines and smaller precharge voltage of the bitlines. However, SAC can not accommodate more than one word in a row in practice. HBLSA and SVGND have slightly higher write energy consumption when comparison is made with respect to the conventional write operation. In particular, SVGND saves the write energy consumption by 84% whereas SAC provides 91% energy saving as predicted in [25].

For main stream applications, the number of write access is only half as many as read access [48]. But for some applications like ATM switching, the writes is as frequent as read operation.

A Numerical Example

This section elaborates the dynamic power efficiency of the SVGND scheme using a numerical example. This example compares the power consumption of the conventional and SVGND scheme. In this example we derive the column read energy consumption associated with one bit for both schemes. Evidently, in order to find the overall array power consumption, one can multiply the said value to the word size. The architectural information and voltage levels are assumed to be the same for both schemes to make a fair comparison.

In this example, the number of words in a row, M, is chosen to be equal to 4 while

the number of cells on a bitline, R, is 128. Therefore in order to access a single bit(cell) of a word, M bits(cells) are accessed. The bitline precharge voltage is $V_{pre} = 0.8V$ and there is a voltage swing of $V_\delta = 0.15V$ over the bitline(s) in the read access. Therefore, the column energy consumption associated with the conventional scheme can be calculated using 4.3:$E_{CONV} = M \times V_{pre} \times V_\delta \times R \times C_{bl}$. Note that in conventional scheme all M bitlines are discharged during the read operation. In the SVGND scheme, the number of cells in a segment, N, is 8. The V_L swing of $0.4V$ is applied to the cell internal node voltages as well as CVG of the to-be-read column. Therefore, the column energy consumption of SVGND scheme is:

$$E_{SVGND} = V_{pre} \times V_\delta \times R \times C_{bl} + V_L^2 \times R \times C_{CVG} + V_L^2 \times N \times C_{INT} \tag{4.4}$$

where C_{CVG} is the capacitive load imposed to the CVG by one cell and C_{INT} is the capacitive load of the internal nodes of a cell seen from SVG. In SVGND, in order to access one bit, only one bitline is discharged in addition to a CVG and the internal cell capacitance of one segment. C_{CVG} is 2.5 times smaller than C_{bl} since the former wire does not carry the capacitance of access transistors and also it is in a high metal layer with far neighboring parallel wires in contrast to the bitlines. On the other hand, as it was mentioned before, C_{INT} is 3 times bigger than C_{bl} because of the internal gate oxide capacitance (See figure 1). The effect of 8% bitline wires extension associated with SW is negligible as SW does not impose a junction capacitance to the bitlines and the additional interconnect capacitance is small(less than 4%). Dividing E_{SVGND} to E_{CONV} portraits the read power saving efficiency:

$$\frac{E_{SVGND}}{E_{CONV}} = 0.445 \tag{4.5}$$

Evidently, the write power saving is more since conventional scheme asks for full swing variation over the bitlines. Also, if the number of words in a row increases or the bitline

length is doubled, the power saving will be even more.

It is worth noting that, the interconnect metal capacitance in this example is based on an available three thin metal layer $130nm$ CMOS technology. Clearly, in state of the art CMOS processes with several metal layer a CVG at the highest metal layer will impose a very small parasitic cap at this node and increases the power efficiency of the scheme significantly.

4.4.2 Speed Consideration

The proposed scheme is primarily intended for low power applications, yet it has limited impact on the overall access time of the unit compared to the conventional scheme. The modest increase in the total access time can be explained due to the fact that the access time of the unit is the sum of the latency of several delay components. According to [30] the total access time can be expressed as:

$$T_{total} = T_{Buff} + T_{Decoder} + T_{Bitline} + T_{SenseAmp} + T_{DataBus} + T_{OutBuff} \tag{4.6}$$

where T_{Buff} is the delay time of the input address buffer, $T_{Decoder}$ is the delay of the decoder, $T_{Bitline}$ is the time needed for the discharge of the bitlines, $T_{SenseAmp}$ is the delay of the sense amplifier, $T_{DataBus}$ is the delay of the data bus and $T_{OutBuff}$ is the delay of the output latches.

Similar to SAC scheme, the SVGND only influences the $T_{Bitline}$ compared to the conventional scheme. Hence, for the same memory size with the same peripheral circuits, the rest of the terms remains constant for all three schemes. Simulation results with the same transistor sizes suggests that the delay time of $T_{Bitline}$ increases in SAC and SVGND compared to the conventional scheme because of the extra switches in the bitline discharge path for a constant bitline capacitive load.

Figure 4.10 illustrates the bitline discharge path of the conventional, SVGND and SAC during the read operation. In this figure, only the bitline that is discharged is shown since this is the only side that communicates the data over the bitline. $M1$ and $M2$ are the drive and access transistors, respectively. $M3$ is the switch that connects the virtual ground of the cell to the ground and CVG in SAC and SVGND schemes, respectively. Assuming the same sense amplifier are used, the bitline must be discharged by the same voltage in all three schemes. In SVGND, the CVG is discharged to the ground through $M4$. $M4$ is designed to be a large transistor and is shared for the whole column. In SAC, the same virtual ground switch(*i.e.*, same M3 size) is shared among four cells in a row, hence four transistors are discharged through this transistor. Therefore, the effective W/L associated with one cell is one fourth.

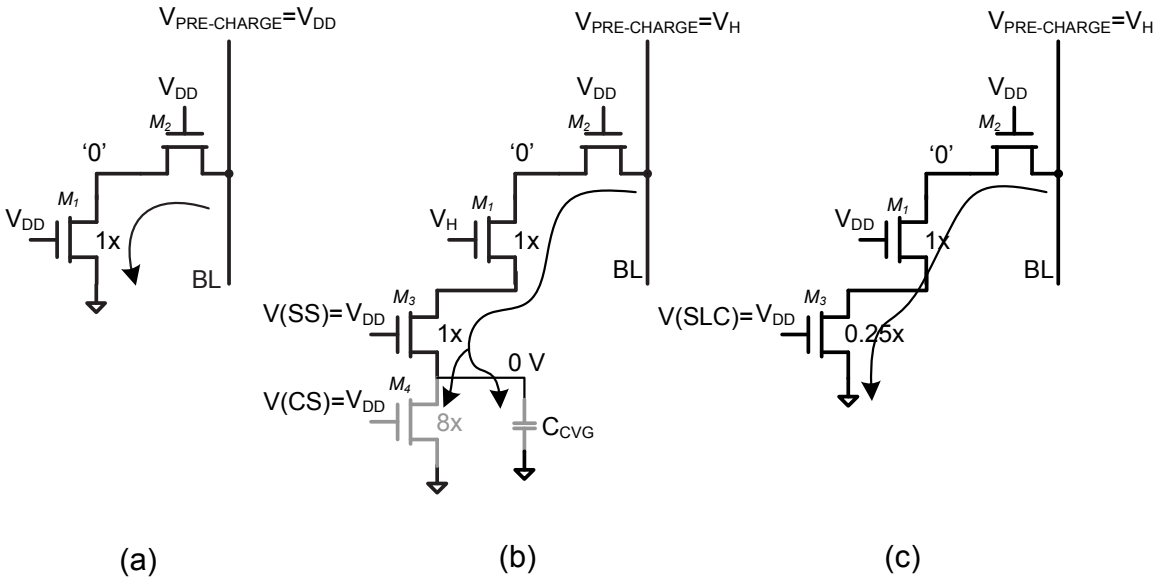


Figure 4.10 Bitline discharge path in read operation for (a) conventional SRAM, (b) SVGND and (c) SAC schemes

Figure 4.11 shows the breakdown of the access time in the said schemes. The bitline

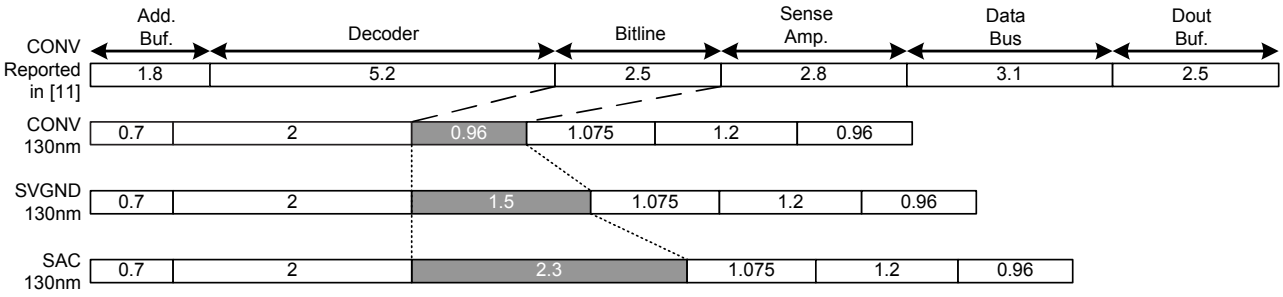


Figure 4.11 The access time breakdown for different schemes in nano Seconds

delay component of different schemes are based on circuit simulation results under the same bitline load and bitline voltage discharge of 200mV while the common terms are according to the data provided in [30]. It can be seen that the overall access time is increased by 7% and 19% compared to the conventional scheme in SVGND and SAC, respectively. Clearly, these values can be different if the delay components in the propagation path are different or the discharge path transistors (*e.g.*, $M3$, $M4$) are sized differently. It is noteworthy that according to [24], HBLSA is slower than SAC because of its two phase data fetching method.

4.4.3 Other design benefits

Leakage energy reduction is the main feature of this scheme. Regardless of the architecture, the leakage current of a cell is determined by the leakage currents of the transistors that are 'off', *i.e.*, $V_{gs} = 0$ [12]. SVGND reduces the leakage current by a factor of 15 compared to conventional and HBLSA using its voltage settings. A higher threshold voltage due to the body effect results in substantial reduction of the leakage current. It is also important to investigate the thermal aspect of leakage current in different architectures. In any architecture, according to equation 2.6, the leakage current has an exponential relationship with

the temperature. SVGND architecture has all cell transistors with high V_{th} in hibernation due to reverse body bias. The HBLSA and the conventional architectures do not have this feature, therefore, they exhibit higher leakage with respect to the temperature. Hence, SVGND offers a lower leakage current under realistic temperatures compared to HBLSA and conventional scheme.

Low voltage SRAM cells are generally susceptible to a higher soft error rates(SER) due to a lower $Q_{cirtical}$. Interleaving multiple words in a row reduces the risk of multiple soft errors in one word. This reduction is the result of the physical distribution of the different bits of a word. Reducing the number of erroneous bits in a word, relaxes the complexity of the FEC technique to recover the corrupted bit. This feature can be used to reduce the vulnerability of the cells against soft errors. As it is mentioned before, this feature is not possible in HBLSA and SAC schemes.

It is evident that decoding takes a significant portion of the overall energy consumption [46]. The energy consumption of a decoder almost doubles if an additional address bit is decoded. The SVGND scheme allows multiple words to be addressed in a single row thus providing greater flexibility of adding extra row or columns, or both in order to accommodate increased addresses. In a typical application where bitwidth is relatively small compared to the number of rows (e.g., 32 bit word and 256 addresses), adding a column decoder to double the address field is less energy consuming than doubling the number of rows or adding a block decoder (especially when the non-accessed bitlines are not discharged during wordline activation.) A column decoder in this case has a smaller capacitive load compared to the capacitive load of a row predecoder load. Therefore, the flexibility of distributing additional words in a row or in a column leads to a lower decoder energy consumption. Reduction of the area overhead of the block decoders and the number of leakage paths through the post decoders is an additional advantage of this scheme.

4.5 Summary

A novel low power SRAM architecture based on segmented virtual grounding scheme is introduced. The scheme offers low power, low energy consumption in both dynamic and static perspectives. In this scheme multiple words can fit in a row while keeping the power consumption of non selected bitlines low when a word is accessed. This feature allows multiple words to be placed in a row while keeping the power low.

The SVGND architecture is compared with recently published schemes. The comparison results show that an array based on SVGND scheme can save the read energy consumption by 74% relative to the conventional scheme and is most energy efficient.In addition, the SVGND scheme saves up to 84% of write operation energy compared to the conventional scheme. It has also been shown that the scheme is capable of reducing the cell static power consumption by several times by reducing the effective supply voltage of the cells.

Allowing more than one word per row reduces per word SER occurrences therefore relaxes the complexity of FEC techniques. Addressing more than one word per row also relaxes the power consumption of the address decoder which takes a substantial fraction of the overall power consumption.

Chapter 5

Case Study: A Low-power SRAM in 130 nm CMOS Technology

This chapter describes the design and implementation of a low-power SRAM unit in a CMOS technology. Section 5.1 gives an introduction. The configuration of the chip is described in section 5.2. The decoding process is treated in sections 5.3 and 5.4. Section 5.5 reveals the timing control unit that is used in the SRAM unit. Finally section 5.8 reports the silicon result.

5.1 Introduction

Realization of a low-power embedded SRAM unit necessitates the low-power design of both the overall architecture and the peripheral circuit in the unit. Conventional techniques such as dividing the entire memory unit into several blocks can be combined with the proposed SVGND technique to reduce the power consumption, aggressively. Combination of different low-power techniques asks for peripheral circuits that can operate

properly in the modified architecture. Therefore, conventional peripheral circuits should be adapted for the low- power architecture to ensure proper operation. For example, the conventional timing control unit and dummy (replica) column should be modified for the new architecture.

In this chapter the design and implementation of a 40Kb low-power SRAM in a 130nm technology will be described. The design is intended for a digital signal processor (DSP) chip for hearing aid applications. The CMOS process that is used for the implementation offers three thin metal layer made of copper and two aluminum thick metal layers. The memory accommodates 2048 words with the word size of 20 bits which results in the address field of 11 bits. Owing to the low speed nature of the application, the speed requirement for this application is about 10MHz, although the unit is designed such that it accommodates higher frequencies. Subsequent section will detail the design choices and the measurement results.

5.2 SRAM configuration

Figure 5.1 shows the top level block diagram of the implemented SRAM unit. The unit is divided into four blocks, where each block accommodates a quarter of the addresses. There are 512 words in each block. Each block comprises of 128 rows and each row comprises of 4 words. Given the word size of 20 bits, 80 cells are placed in each row. The cells associated with different bits of the four words on a row are interleaved in the same fashion that is described in Chapter 1. Therefore, twenty sense amplifiers (SA) and write drivers are required for each block. However, as we will see in the next section, the SAs (and write drivers) are shared between the top and bottom blocks to reduce power consumption, area and complexity.

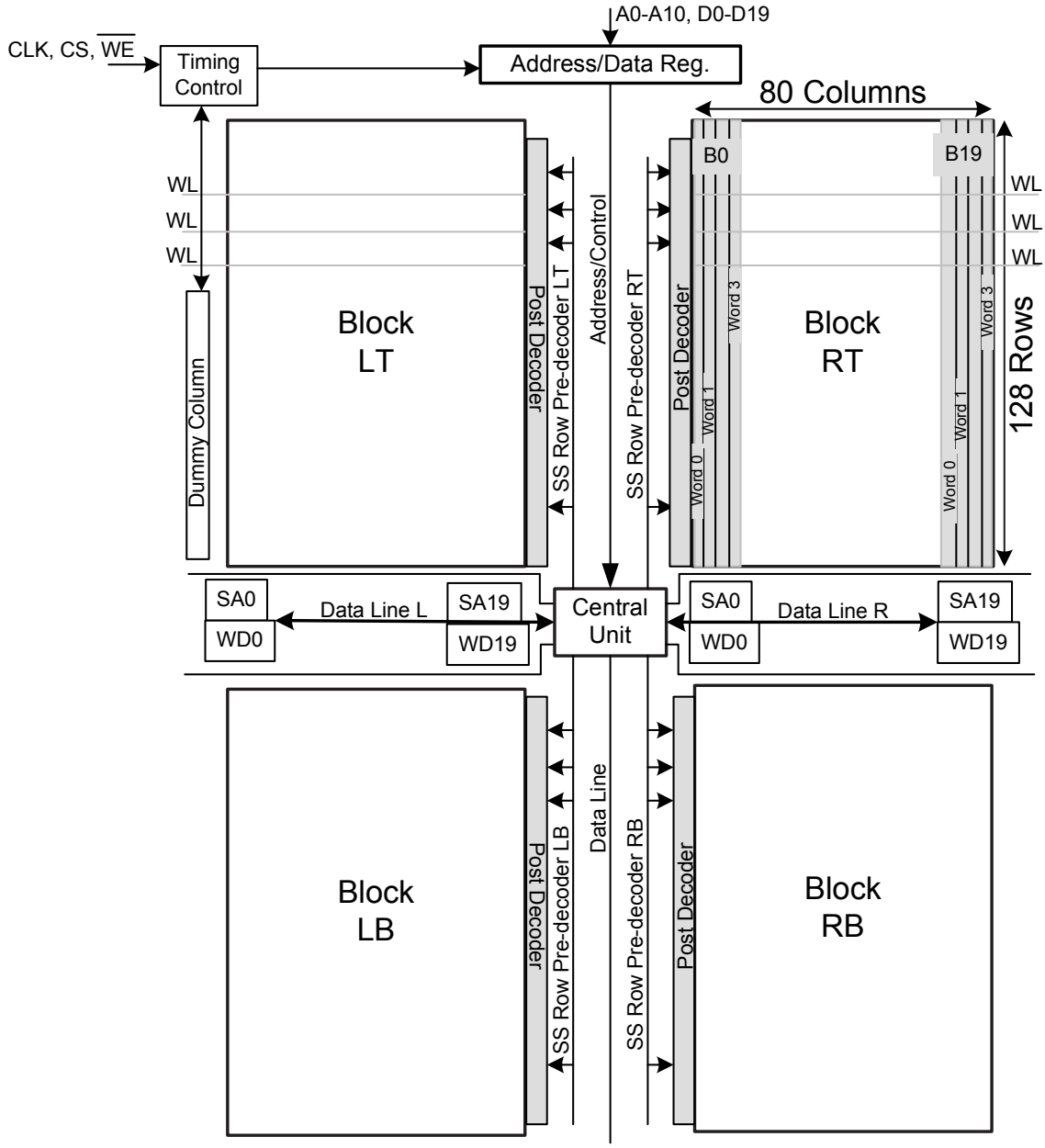


Figure 5.1 Top level block diagram of the implemented SRAM

Periphery units such as row and column decoder as well as timing unit is shared among the four blocks. In order to save power and increase the speed, a big fraction of the decoding process is implemented at a central unit which is physically located at the center of the SRAM unit. A timing control unit is used to control the read and write operations. A dummy column is used to mimic the delay time associated with the bitlines.

The SVGND architecture is utilized in every block to reduce the read, write and leakage power consumption. A segment comprises of $N = 8$ cells in a column with a shared virtual ground as described in the previous chapter. Therefore, 16 segments constitute the entire column which has 128 rows ($R = 128 = 16 \times N$.) The segment select (SS) signal associated with a segment will be active whenever one of the cells on the segment is accessed. Running in parallel to the wordline, SS is derived from one of the row pre-decoder's output which is called segment select(SS) row pre-decoder.

The four block configuration has several advantages over a single block configuration. Dividing the memory unit into the top and bottom blocks has the advantage of reducing the capacitive load associated with the bitlines and row predecoder's output which results in a low-power and high speed design. Dividing the memory into left and write blocks offers a 50% reduction in the capacitive load associated with the wordline, SS signal, the control signals for the sense amplifiers (SA) and write drivers as well as the capacitive load at the output of the column decoder.

Table 5.1 shows the address bit assignment of the unit. The most significant bits of the address input A10 and A9 specify the accessed block. A10 selects between the right or left blocks while A9 chooses between top or the bottom blocks. The least significant bits of the address, A0 and A1, are assigned to perform the column selection. A2-A4 specifies which cell in a segment is accessed. Therefore, it contains part of the row address information. Address bits of A5-A8 specifies the accessed segment. Therefore, altogether

Table 5.1 Address bits assignment

Address	Description
A0-A1	Column Address
A2-A4	Cell Address in a Segment
A5-A8	Segment Address
A9-A10	Block Address

the seven address bits of A2-A8 indicate the row in which the accessed word is located.

5.3 Row Decoders

Different parts of the input address are decoded at different physical locations in order to save area and power. Figure 5.2 shows the idea of distributing the decoding process in the unit. A 3 to 8 row pre-decoder is utilized to decode A2-A4 address bits. This decoder is physically located at the top of the SRAM unit beside the address register unit. The output of this row pre-decoder drives the post decoder AND gates of all four blocks. Therefore, every output of this predecoder runs vertically in parallel to all four blocks. As it was mentioned before, the output of this predecoder specifies which cell among the 8 cells of the selected segment is accessed.

The rest of the row decoding process is implemented at the central unit where access to the control signals of all blocks is available. The input address bits associated with all decoders in the central unit is delivered from the top where the address input registers are located. A 4 to 16 decoder is used as a segment select predecoder of the row decoder. This decoder decodes the address bits A5-A8. The output of this decoder specifies the segment that is accessed in a given block and activates the SS signal for that segment.

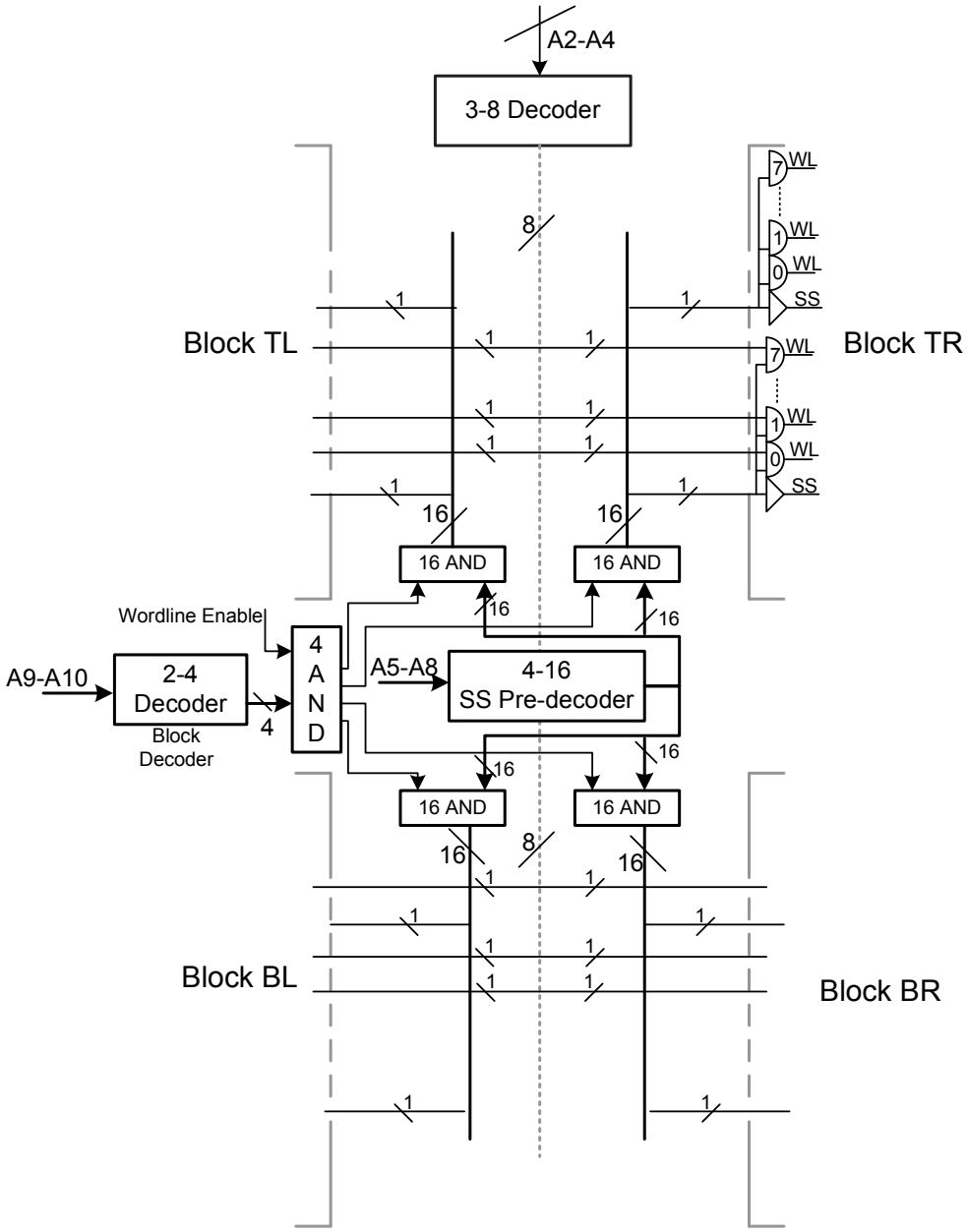


Figure 5.2 Top level block diagram of the implemented SRAM

The block select address bits, A10 and A9, mask the outputs of the segment select predecoder from the non-accessed blocks. The four outputs of the block decoder are combined with the 16 outputs of the segment select predecoder using 4×16 AND gates. The outputs of the AND gates generates four versions of the segment select predecoder output for the four blocks. The 16 segment select output of each block runs vertically in parallel to the corresponding block to drive the post decoder AND gates associated with the block. Moreover, the SS signals of each block is also derived from the same outputs.

Poor timing in activating and de-activating the decoder's output may cause coupling issues: It may cause two wordlines remain active at the same time. In order to control the timing of the wordline a 'wordline enable' signal controls the outputs of the block decoder before these outputs are combined with the outputs of the segment select predecoder. Utilization of the 'wordline enable' signal prevents coupling between two cells when two consecutive access on different rows happens. The operation and generation of this signal will be discussed in detail in the timing control section.

Combining the block address information with the segment select predecoder's output prevents unnecessary propagation of the decoded signals on the long interconnects of segment select predecoder's output. This operation stops the unnecessary activation of the wordlines, SS signals and bitlines of the non selected blocks. Reducing the length of the activated wordline and SS signals saves the power consumption and adds to the speed of operation.

5.4 Data Path Decoders

The SA and write drivers of the two top blocks are shared with the bottom blocks to save the space and complexity of the design. Hence, only 40 SAs (and write drivers) are used,

half of which is on the left side and the other half is on the right side. Figure 5.3 shows this idea. Each SA is shared among eight columns; four columns in the top block and four columns in the bottom block. Driven by the column decoder output, the pass gates of the column multiplexer connect the selected pair of bitlines to the corresponding SA or write driver.

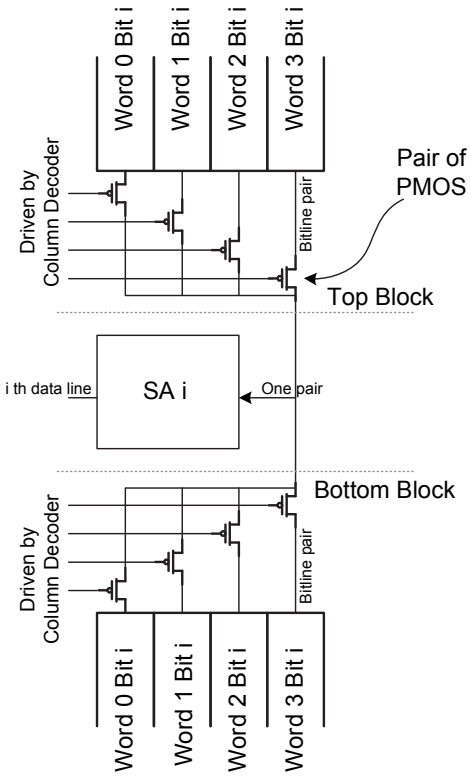


Figure 5.3 Organization of the column multiplexer transmission gates to share a SA between eight bitlines in the data path

Figure 5.4 shows the organization of the column decoder in the central unit. The output of the column decoder is also masked by A9 and A10 from the non-accessed blocks. A10 specifies which of the right or left blocks are selected by masking the column decoder's

output on the side that is not selected. This choice prevents the unnecessary propagation of the column decoder's output to the non selected side. A9, on the other hand, directs the column decoder's output to the top block or the bottom block. Therefore, for each block, the corresponding set of four column decoder outputs is generated. In a given transaction, only one set of the four sets of the column decoder output is active. Therefore, in a read operation only one SA per bit is operational and the SAs on the non-selected side (left or right) does not affect the voltage of the corresponding data line.

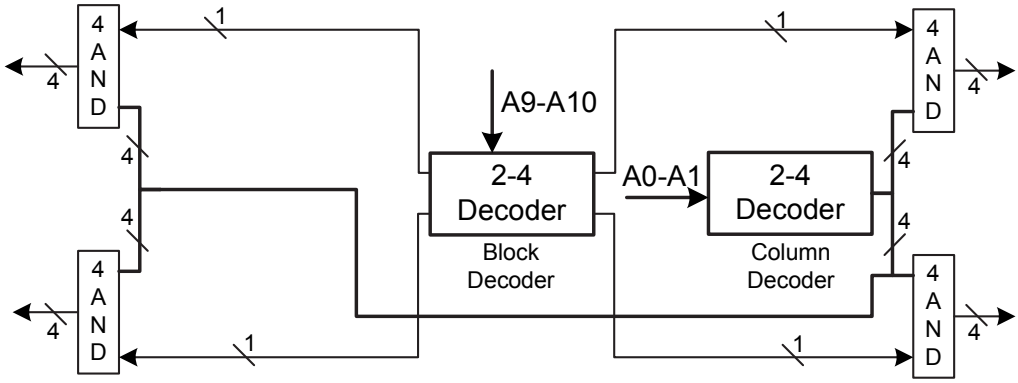


Figure 5.4 Organization of the column decoder in the central unit

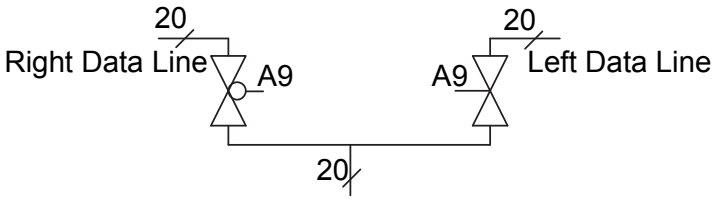


Figure 5.5 Data multiplexer enabling data propagation to left or right blocks

Figure 5.5 shows the data line path. Depending on A9, the left side or right side data line is directed to the SRAM output at the bottom of the unit. Two sets of 20 transmission

gates are implemented in the central unit for this purpose.

5.5 Timing Control Unit

Accurate timing is the key in proper operation of the SRAM. The proper operation of the unit depends on the parallel asynchronous operation of several periphery blocks including decoders, SAs, multiplexers, and the array itself. A timing control unit is designed to control the operation of different blocks in every operation. The timing control unit communicates with a dummy column which replicates the bitline delay time.

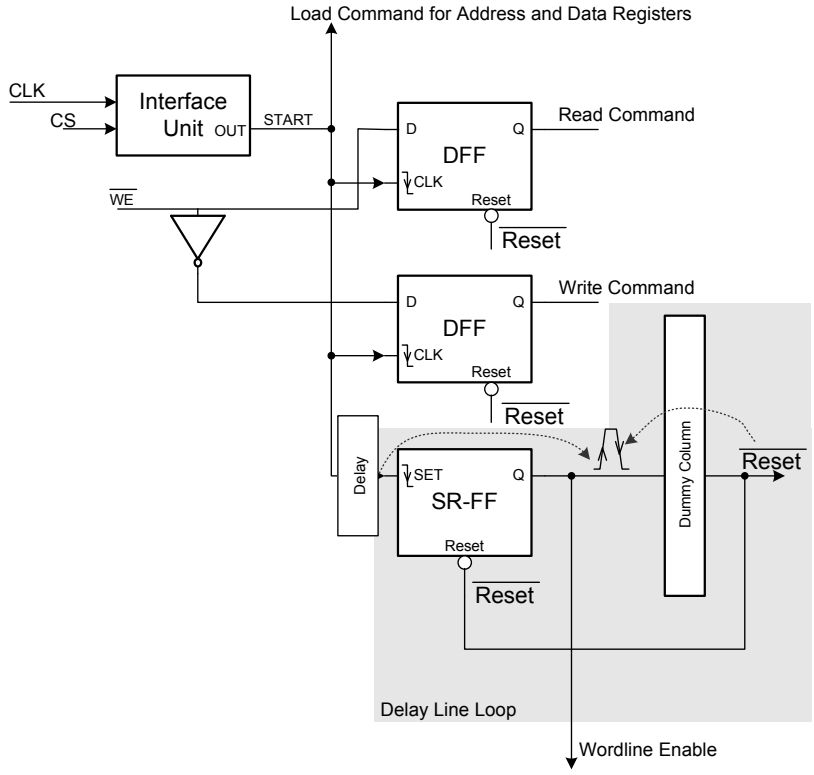


Figure 5.6 Block diagram of the timing unit

Figure 5.6 shows the block diagram of the timing control unit. Three control inputs trigger the control unit; chip select (CS), clock(CLK) and write enable bar ($\overline{WE}$). If CS is active when the unit observes the rising edge of the clock, it means that a read or write operation is intended. The type of the operation is specified by the $\overline{WE}$ signal. The timing control unit consists of an interface unit, a delay line loop and a number of edge triggered D flip-flops. The interface unit deals with CS and CLK and its output is normally high. This unit generates a negative edge at its output once it observes an activated CS at the rising edge of the clock. Figure 5.7 shows the schematic diagram of the interface unit. The interface unit consists of an edge triggered D flip-flop (DFF) and two inverters and a NAND gate. The two inverters delay time is equal to D-Q delay time of the DFF.

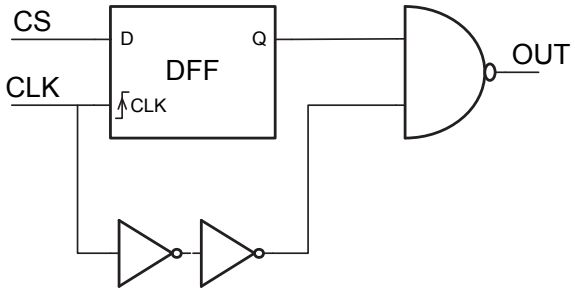


Figure 5.7 Realization of the interface unit of the timing unit

The timing control unit starts its operation when the interface unit generates a negative edge at its output. The output of the interface unit is called 'START' signal and is buffered to generate the load command for the address and data input registers. These registers are basically made of a set of edge triggered DFFs. The output of the address registers are buffered before they are connected to the corresponding decoders.

The negative edge that is generated at the output of the interface unit is also used for other blocks in the timing control unit. It is used as the clock signal for the DFFs that

save the read or write commands. These DFFs are equipped with a reset input which can be used to reset the output of the DFF once a read or write operation is completed. The activation of the write command signal associates with enabling the write drivers and the activation of the read command signal resets the SAs. When the operation is completed an active low $\overline{Reset}$ signal will be generated by the dummy unit to deactivate the read and write commands. As we will see in the next section, deactivation of the read command initiates the operation of the SA.

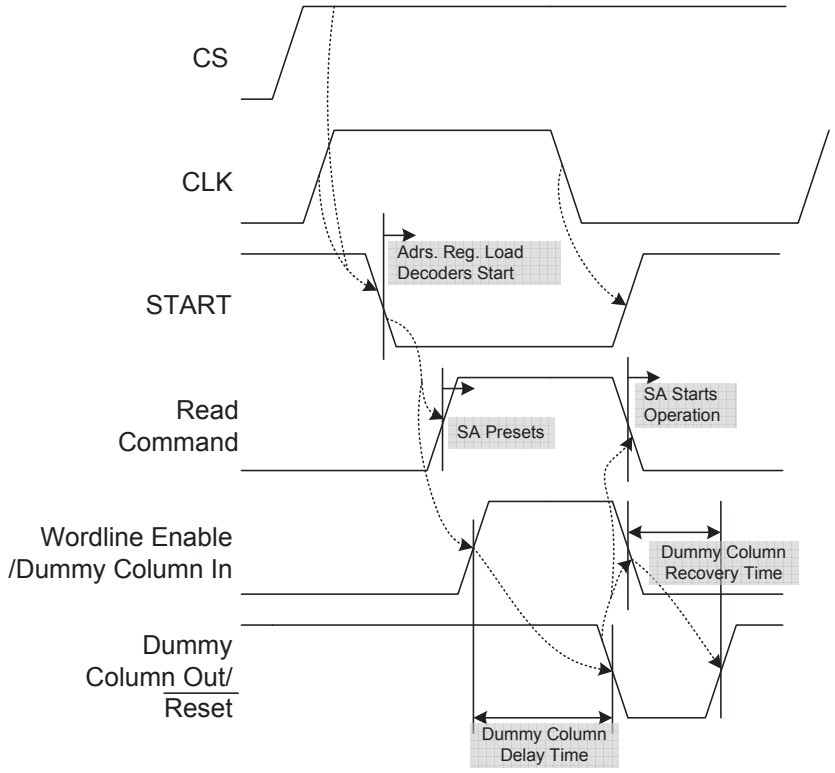


Figure 5.8 The conceptual waveforms in the timing control unit

Figure 5.8 shows the waveforms at different internal nodes of the timing control unit. A set-reset flip-flop (SR-FF) is used to control the wordline access time. The set input of

the SR-FF is a delayed version of the output of the interface unit output. A falling edge at the set input of the SR-FF causes the output of the SR-FF to become active. The output of the SR-FF is used to activate both the dummy column and the wordline in the array. The dummy column operates as an inverter with a delay time identical to the bitline delay. Hence, a logic low at the output of the dummy column indicates that the array bitline operation is complete and the wordline can be deactivated. Consequently, the output of the dummy column is considered as an active low $\overline{Reset}$ signal for the SR-FF. Therefore, when the output of the dummy column drops to logic 'zero' it deactivates the output of the SR-FF. Deactivation of the output of the SR-FF turns off the wordline in the array and presets the dummy output to logic 'one'. It is noteworthy that the activation and deactivation of the wordline in the array is controlled by the 'wordline enable' signal at the central unit (See Figure 5.2.)

The reset signal for the SR-FF (*i.e.*, the dummy column output) is also used to reset the read or write command DFFs. Resetting the read command DFF initiates the SA operation and resetting the write command decouples the write drivers from the bitlines. In this scheme the column decoder and read command are set before the wordline enable signal. This choice allows the column virtual grounds (CVG) of the accessed column to be discharged before the wordline and SS signal become actually active. In other words, the row predecoder operation and the CVG discharge is done concurrently. This concurrence increases the speed of operation.

Figure 5.9 shows the schematic diagram of the dummy unit. The dummy column architecture is identical to an array column since both use SVGND scheme. The transistor sizes in the dummy column are the same as the size of the transistors used in an array column. There are only two differences between a column in the array and the dummy column. The first difference is that the number of rows in the dummy column are one

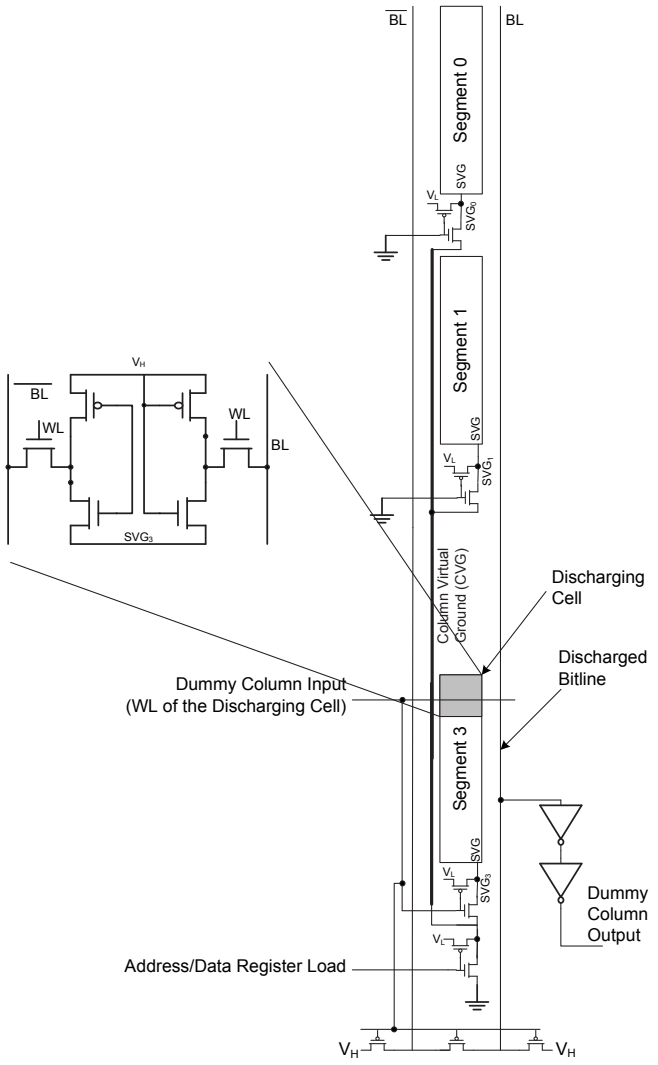


Figure 5.9 Block diagram of the dummy unit

fourth of the number of rows in an array column. Reduction of the number of rows in the dummy column reduces the bitline capacitance of the dummy column. This reduction magnifies the dummy column bitline voltage drop by four times with respect to the array bitline voltage drop. The dummy column bitline voltage drop is buffered to produce the

$\overline{Reset}$ signal.

There is another difference between the dummy column and an array column. In an array column the wordline activates a cell with a random logic value which can discharge any one of the two bitlines. However, the input of the dummy column turns on the wordline of a cell that stores a known logic value. Therefore, only a predetermined bitline is discharged. The cell that is responsible for discharging the bitline is called a discharging cell. The bitline discharge is detected using an inverter with a properly designed threshold voltage.

5.6 Sense Amplifier and Write Driver

Figure 5.10 shows the schematic of the SA that is used in the SRAM unit. The utilized SA has two important properties which distinguishes it from the conventional SAs that has been reported in Chapter 2. The first property is that it has a cross couple latch in its configuration which relaxes the gain requirement of the amplifier. Also, in this configuration the input and output are isolated from each other. Therefore, the SA can remain active and deliver the output without conflicting with the bitline voltage. This property is especially important because it relaxes the timing requirement of the SA enable signal. This feature allows bitlines to be precharged to their nominal voltages while the SA is providing the output associated with the initial value of the bitlines.

Figure 5.11 depicts the schematic of the write driver. The write driver consists of two transmission gates and two AND gates. Owing to low voltage swing write operation transmission gates are used instead of conventional NMOS transistors. Depending on the input logic value, the corresponding bitline is discharged to V_{WR} through the transmission gates. The transmission gates are sized such that they are strong enough to discharge

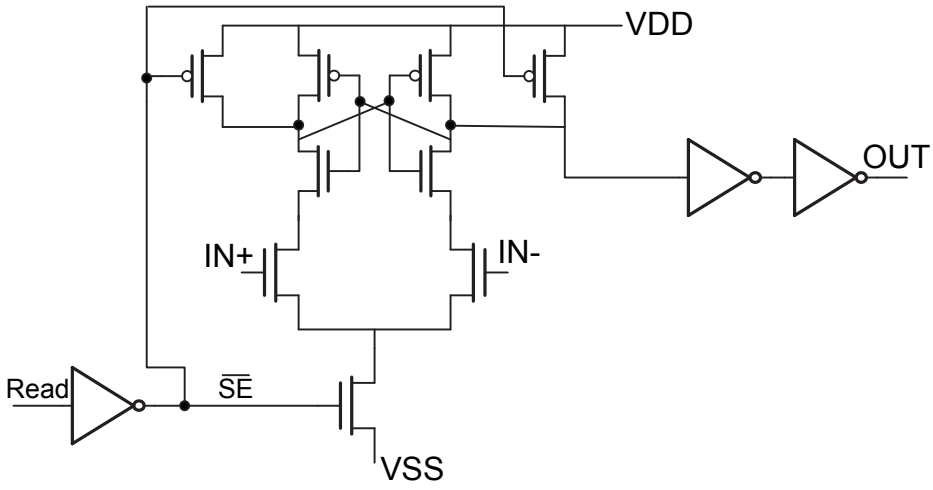


Figure 5.10 Schematic of the Sense Amplifier

the bitlines in a short time. The write driver is isolated from the bitlines when the write command signal is disabled.

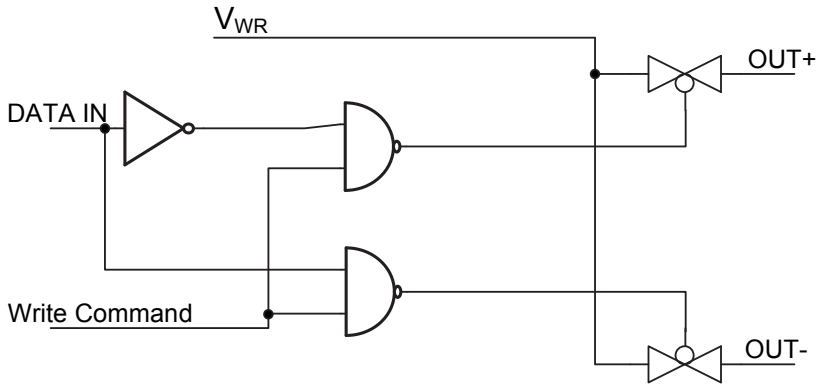


Figure 5.11 Write Driver

5.7 Layout and Silicon Micrograph

Figure 5.12 shows the implemented silicon micrograph. Unfortunately, due to the intensive auto-fill on higher metal layers the core design is masked from the view. The unit takes $410\mu m \times 860\mu m$ in a $130nm$ CMOS technology. Each block takes $150\mu m \times 400\mu m$. The top level layout of the chip is presented in Figure 5.13. The layout of the SVG switch in the array and the central unit is magnified in the layout. The size of the SVG switch is compared to the size of a cell in this figure. The SVG switch is shared among $N = 8$ cells. Therefore, the area overhead of the SVGND scheme is less than 8%. The figure also shows the location of the central unit in the layout.

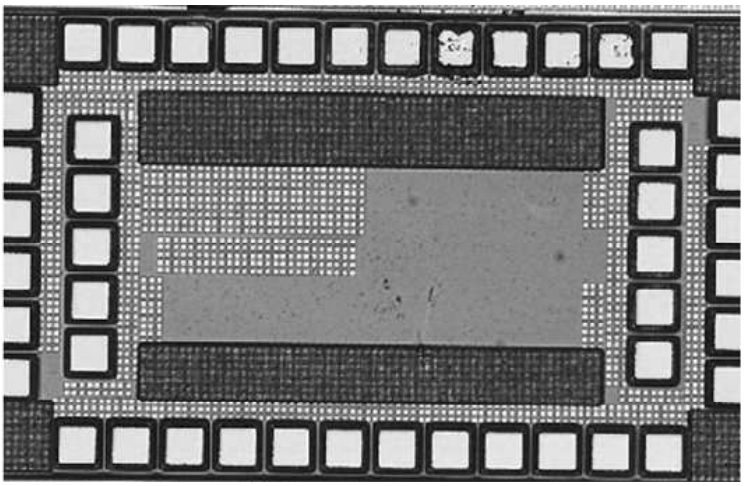


Figure 5.12 Silicon micrograph of the SRAM unit

Figure 5.14 shows the details of the central unit layout. The core unit layout includes the 4-16 segment select row predecoder. The output of this decoder feeds 4×16 AND gates associated with four blocks. Two 2-4 decoders are used to decode the column address and the block address in the core unit. The output of the column decoder output is used to

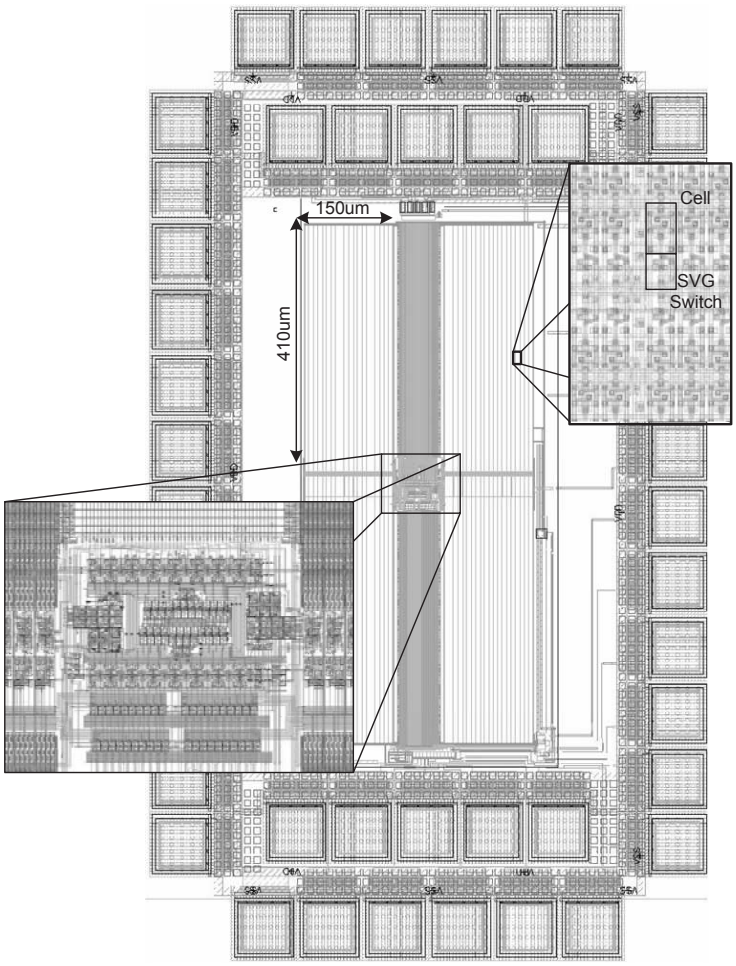


Figure 5.13 Top level layout of the SRAM unit

produce 4×4 column select signals. 8 AND gates on each side produce half of the column select signals. To combine the wordline enable signal with the output of the block decoder, 4 AND gates are used, two of which are in the top left corner and the other two are in the bottom right corner of the central unit. The output of each of these AND gates drive the inputs of 16 AND gates associated with the SS predecoder (See Figure 5.2.) It can be seen

that the low-power consumption associated with breaking down the SRAM unit into four blocks comes at the expense of area overhead of the central unit.

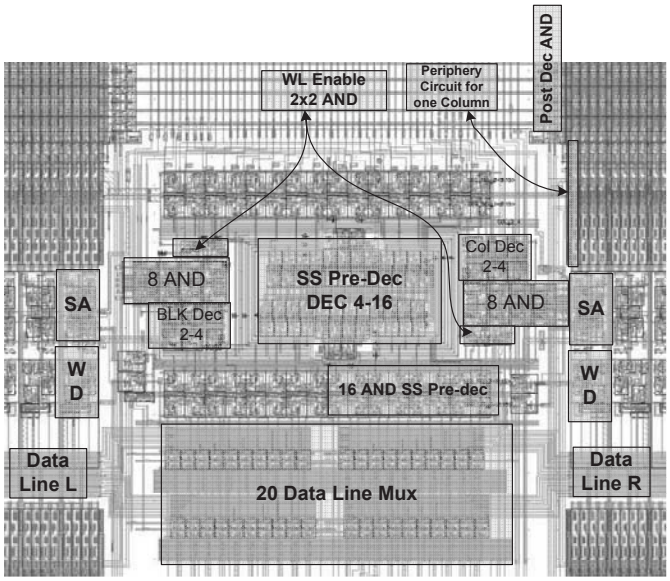


Figure 5.14 Layout of the core unit

A sense amplifier and a write driver is also shown in the Figure 5.14. It can be seen that each pair of SA and write driver (WD) is shared among four columns. The data lines are also shown on the two sides of the layout. The data line multiplexers at the bottom of the core unit conduct one of the two sides to the bottom of the SRAM unit.

5.8 Measurement Results and Comparison

Extensive silicon measurements were carried out to substantiate the simulation results and confirm the low-power of SVGND architecture. The chip was designed such that the power consumption of different components of the SRAM unit can be measures separately.

Table 5.2 Voltage levels in the test setup

Voltage source	Description	Level
V_{DD}	Supply for digital periphery, decoders and data lines	1.2V
V_{WL}	Wordline voltage	1.1V
V_{SS}	Segment Select Signal	1.2V
V_H	Cell high voltage	0.9V
V_L	Cell low voltage	0.5V
V_{WR}	Write voltage	0.5V

Five voltage regulators provide different voltage levels required for the operation of the chip. The power supply voltage of $1.2V$ is provided for the periphery blocks and decoders. The wordline voltage and voltage level of the segment select signal can also be controlled. Moreover, the high and low voltage of the cells, V_H and V_L, can be adjusted and the current passing through them can also be measured. In addition, the bitline voltage level for the write operation can also be set. As we mentioned before, the bitlines are precharged to V_H after each operation while column virtual ground (CVG) has a precharge voltage of V_L.

The total power consumption of the chips is obtained by summing up the power consumption associated with each voltage source. In order to find out the power consumption associated with each voltage source, the current that is sunk from the source is measured and multiplied to the voltage of the voltage source. Table 5.2 shows the typical voltage levels at different voltage sources that made the SRAM unit operate properly. We deliberately reduced write voltage down to $0.5V$ to match with V_L.

Except for the V_{DD} power consumption, the rest of the power associated with other supply voltages is dissipated in the array. The array power consumption is proportional to the word size, as discussed in Chapter 4. Power consumption of V_{DD}, on the other hand,

Table 5.3 Dynamic power consumption during read and write operation

Voltage source	**Read Power Consumption(W)**	**Write Power Consumption(W)**
V_{DD}	1.72e-04	1.72e-4
$V_{WL} + V_{SS}$	8.4e-6	8.4e-6
V_H	5.85e-5	1.26e-4
V_L	1.5e-5	0
Total	254e-6	306e-6

has direct relationship with the address size. It is worth noting that a separate supply voltage is used to drive the output buffers to distinguish the power consumption of the internal unit from the off-chip power consumption.

The leakage current of the SRAM unit is measured in the stand-by mode by measuring the current taken from the voltage sources of V_{DD} and V_H. The latter is associated with the cell leakage current while the former measures the leakage current of the periphery units (*e.g.*, , pre and post decoders, SA, etc.) The leakage current from the V_{DD} is $600nA$ and the leakage current from V_H is $1.12\mu A$ that is $27.3pA/cell$. This amount can be compared to the $530pA/cell$ in a conventional design.

Table 5.3 shows the read and write dynamic power consumption when the unit operates at $50MHz$. The table shows the power consumption associated with each supply voltage. In this experiment, the data line are charged and discharged in every cycle. The read and write power consumption is measured when the input address sweeps the entire memory unit and the outputs show valid values. The overall power consumption is measured to be $306\mu W$ and $254\mu W$ in write and read operation, respectively. These values are equal to 40% of write power consumption and 53% of read power consumption in a conventional design with the same blocking scheme.

Despite the intensive research on low-power SRAM architecture in the industry and academia in the past few years, no particular scheme have dominated the low-power SRAM design yet. Table 5.4 draws a comparison between the SVGND scheme and some of the low-power SRAMs that are reported in the past two years. In these reports, depending on the application, the address size, word size and the way the memory is accessed, a specific fraction of power consumption is the center of attention for power reduction. For example, in very large memories with medium word size for cache applications the leakage current reduction is the main focus. On the other hand, in high speed small size memories only write power consumption is considered for power reduction. It is worth noting that except for [10] and [49], the rest of the scheme can only accommodate a single word per row. The effect of this limitation on the decoding power consumption is substantial when the address size becomes large. In addition, the scheme reported in [50] requires 10 transistors per cell.

Figure 5.15 puts the reported values in perspective with respect to SVGND for write power consumption. In this figure, the reported write power consumption values are scaled with respect to frequency and supply voltage. For reports that only have the power consumption saving compared to the conventional scheme (*i.e.*, does not include the absolute value of power consumption) the power consumption of a memory with the reported size is estimated and multiplied to the amount of the reported power saving factor.

5.9 Summary

The implementation of a SVGND based 2048×20 bit SRAM unit was explained in this chapter. The unit was implemented in a $130nm$ CMOS technology and takes $0.352mm^2$. The implementation of the unit is based on four blocks each of which takes a quarter of

Table 5.4 Comparison between the SVGND and other schemes

Ref.	**Word size(b)**	**Adrs. size(b)**	**Supply voltage**	**Freq.**	**Leakage**	**Write power**	**Consideration**
Conv.	20	11	$1.2V$	$50MHz$	$530pA/Cell$	$781\mu W$	Same blocking
[25]	256	8	$1.5V$	$100MHz$	N/A	$13.4mW$	-
[24]	32	13	$2.5V$	$200MHz$	N/A	$28mW$	-
[10]	64	18	$1.4V$	$500MHz$	$20pA/Cell$	$480mW$	Overall *active* power is reported instead of write power
[50]	128	11	$0.6V$	$475kHz$	$32pA/Cell$	50%	Only *active* power reduction compared to conventional is reported
[49]	N/A	N/A	$1.2V$	$450MHz$	$15pA/Cell$	18%	Only *active* power reduction compared to conventional is reported
SVGND	20	11	$1.2V$	$50MHz$	$27pA/Cell$	$306\mu W$	-

the addresses. A central unit is located at the middle of the unit which is in charge of decoding blocks, columns and the predecoding of the row address line. The operation of the timing control unit and the sense amplifiers as well as write drivers is explained.

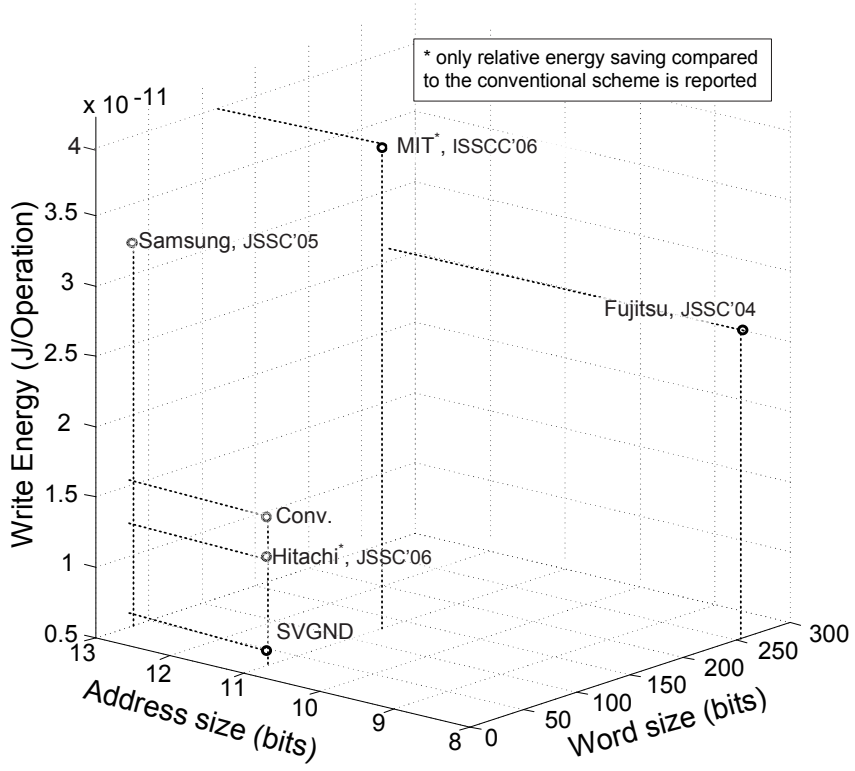


Figure 5.15 Write power consumption comparison

Silicon results shows a significant reduction in both static and dynamic power consumption when it is compared to the conventional scheme. Also, in comparison to the recently reported schemes, the SVGND offers a substantially less dynamic power consumption while it drops the leakage current into reasonable ranges.

Chapter 6

Discussion and Future Works

With growing demand for low-power SRAMs different sources of power consumption are the targets for the low-power design techniques. Increasing leakage current in the scaled technology demands low-power techniques that reduces the leakage current of the SRAM cell. In addition, the dynamic power consumption of the SRAM unit is significant because of the high capacitance of the long interconnecting wires. In particular, the write operation power consumption takes a significant fraction of the overall power consumption because of the high voltage swing nature over the bitlines. Therefore, techniques to reduce the dynamic power consumption and particularly write power consumption has received significant attention.

Conventional techniques for controlling the SRAM cell leakage current is usually based on varying the supply voltage of an entire block. In such schemes, the entire unit goes to a low-voltage low-leakage hibernation mode where the cells can retain the data. However, the block can not bear with a read operation in this mode because of the weakness of the drive transistors to communicate with the bitlines. Therefore, the supply voltage of the entire block has to be increased for a successful accessed operation. It has shown that variation

of the supply voltage of the block imposes a substantial dynamic power consumption to the SRAM unit.

Reduction of the bitline voltage swing has also been reported in the literature for the write power reduction. However, this technique involves turning off the supply voltage of an entire row which results in an architectural limitation. Specifically, in such a scheme, only one word can be placed on a row. This limitation not only increases the multiple soft error rates per word (SER) but also increases the complexity and the power consumption of the row decoding path. Nonetheless, the effectiveness of such schemes is limited to the write power consumption and does not reduce the read power consumption or the leakage.

It is shown that the proposed supply voltage variation can break the deadlock between the static and dynamic power consumption. In other words, if the supply voltage variation is only applied to the cells that are accessed for the read operation then the supply voltage variation does not impose a significant power consumption. This concept requires an additional operational mode to the conventional operational modes of the SRAM cell. The additional operational mode is called accessed retention mode (AR-mode) in which the accessed cell has to retain the data without any supply voltage variation.

In addition to the leakage reduction, introducing AR-mode differentiates the task of a cell that is accessed for the read operation and the task of a cell that is accessed only because a neighboring cell in the same row is read. The cell that is accessed for read operation discharges the bitline while the AR-mode cell leaves the bitline voltage of the non-accessed column unchanged. Therefore, introduction of the AR-mode reduces the dynamic power consumption, significantly.

The proposed supply voltage variation of the cells is realized in a segmented virtual grounding (SVGND) architecture. It is shown that the architecture saves both dynamic and static power consumption. The power reduction capability of the scheme is compared to the

state-of-the-art techniques. The SVGND outperforms all other schemes in read operation and it offers a substantial write power reduction which is comparable to the state-of-the-art. At the same time, the static power consumption is also reduced significantly.

The benefits of the SVGND scheme can be further appreciated as the technology scales. The bitline leakage current associated with the cells on the same column increases as the technology scales. Therefore, for the same amount of current drive, the voltage difference over the bitlines is reduced. This increases the requirement of the sanse amplifiers in the architectures that does not reduce the leakage currents of the cells on the column that is selected for read operation. SVGND, on the other hand, benefits from low-leakage operational mode of the cells on the to-be-read column.

The power consumption reduction of the low-voltage SRAMs comes at the price of poorer *static* data stability of the cell during the accessed retention mode. However, it is shown that the conventional static data stability analysis does not cover the entire scope of the data stability concept of a cell. The static data stability criteria is based on the static (DC) behavior of the accessed cell. However, in practice an SRAM cell alternates between accessed mode and the retention mode. Therefore, a dynamic stability criteria is introduced to analyze the stability of an SRAM cell. Supported by measurement results, the dynamic stability theory suggests that time can be traded-off with the data stability in an SRAM cell. This trade-off suggests that if the cell access time is short enough compared to the time constant of the cell, the cell is able to recover the original logic state and remains data stable. The subthreshold operation of the cells in the AR-mode in the SVGND scheme exploits this trade-off to maintain stability.

6.1 Future Works

The notion of static data stability and its associated measure, static noise margin (SNM), has dominated the realm of digital circuit design for four decades. Therefore, the dependency of SNM on scaling, process variation, faults, etc., has received a substantial attention in the literature. However, such treatments are based on the static behavior of the cell. The introduction of the dynamic data stability criteria demands a new level of research on the effect of process variation and faults on the stability of an SRAM cell. Similar to the case with the static data stability, theoretical bounds of different design parameters (*e.g.*, access time, recovery time, supply voltage, etc.) for maintaining the data stability should to be driven.

Supply voltage reduction associates with an increased soft error rate (SER). Therefore, novel efficient techniques needs to be investigated along with the feasibility of the conventional schemes to alleviate this problem. For example, using conventional forward error correction (FEC) methods, 6 extra bits can correct a single error in a 32 bit word. Limiting the number of errors in a word is viable in SVGND because of the possibility of implementation of multiple words in a row. This feature opens research areas for more efficient FEC methods for SER reduction.

Driven by the demand for the lowest power solution, SRAM optimization techniques have been adapted over the last decade. These techniques optimize the address break down in terms of number of blocks, rows and words per row for a given address and word size. However, all of these methods are based on conventional SRAM architecture where all non selected bitlines are discharged once the wordline of a row becomes active. With introduction of SVGND scheme the guideline for optimizing the address break down is changed and new optimization methods should be adapted.

Appendix A

Theorem on the convergent properties of periodic solutions

Assume the right side of the system:

$$\frac{dX}{dt} = F(X,t) \tag{A.1}$$

is defined as:

$$F(X,t) = \begin{cases} F_1(X), & kT \leq t < T_1 + kT \\ F_2(X), & T_1 + kT \leq t < (k+1)T \end{cases} \tag{A.2}$$

for $k = 0, 1, ...$ If $F_1(X)$ and $F_2(X)$ are uniformly monotonically asymptotically stable systems over region G and there exist a solution $\psi(t) = (t, X_0, t_0)$ that is bounded within G for $\forall t > t_0$, then $F(X,t)$ is convergent, meaning:

1. There exist a unique periodic solution,$\phi(t)$, with the period of $1/T$ in G.

2. The following relationship is held by any solution $X(t, X^*, t^*)$ that remains in G for all $t > t_0$:

$$\lim_{t\to\infty} \parallel X(t, X^*, t^*) - \phi(t) \parallel = 0. \tag{A.3}$$

Proof

Definition Let $X = X(t, X^*, t^*)$ be the solution of the system (A.1) with initial data $t = t^*, X = X^*$. Assume that the point X^* of the hyperplane $t = 0$ is such that the solution of $X(t, X^*, 0)$ can be extended to all $0 \leq t \leq T$. By associating the point $X(T, X^*, 0)$ with the point X^*, we obtain a transformation D of the section of the hyperplane $t = 0$ through which solutions are extended by integral multiples of the period pass into the hyperplane $t = 0$. It follows from the uniqueness and the theorem on integral continuity that D is one-to-one and continuous in both directions [32]. Also, it follows from the definition of D that the function $X(t, X^*, 0)$ is a harmonic oscillation if and only if X^* is stationary under D: In other words, $DX^* = X^*$. (The definition of D is well known and can also be found in Chapter 2 in [32].)

Lemma Transformation D associated with the periodic function of F is convergent. In other words,

$\forall X_0, X_0' \in G : \|X(t, X_0, t_0) - X(t, X_0', t_0)\| > \|X(t + T, X_0, t_0) - X(t + T, X_0', t_0)\|.$

Proof Without loss of generality let $t = nT + \Delta t, \Delta t < T_1$ then because of the monotonicity of F_1:

$\|X(nT + T_1, X_0, t_0) - X(nT + T_1, X_0', t_0)\| < \|X(t, X_0, t_0) - X(t, X_0', t_0)\|$

Because of the monotonicity of F_2 and considering the continuity of the X with respect to t we have:

$\|X((n + 1)T, X_0, t_0) - X((n + 1)T, X_0', t_0)\| < \|X(nT + T_1, X_0, t_0) - X(nT + T_1, X_0', t_0)\|$

Again since F_1 is monotonic, we have:

$\|X((n + 1)T + \Delta t, X_0, t_0) - X((n + 1)T + \Delta t, X_0', t_0)\| < \|X((n + 1)T, X_0, t_0) - X((n + 1)T, X_0', t_0)\|$

Which means, $\|DX_0 - DX_0'\| < \|X_0 - X_0'\|$. Q.E.D.

We are going to show that system (A.1) is asymptotic in G. In other words for every pair of initial conditions of $X_0, X_0' \in G$:

$\forall \epsilon, \exists t$ such that $\|X(t, X_0, t_0) - X(t, X_0', t_0)\| < \epsilon$.

Proof We assume the contrary:

$\alpha = \sup_{X_0, X_0' \in G} \|X(t, X_0, t_0) - X(t, X_0', t_0)\| > \epsilon$

Since G is a closed region therefore according to the lemma that we proved previously we have:

$\|D^{-1}X - D^{-1}X'\| > \|X - X'\| = \alpha$

But this contradicts the definition of α and this contradiction shows that X' converges to X where X and X' are the solutions of the system with initial conditions of X_0 and X_0' at t_0, respectively. This relationship implies that (A.1) is dissipative in G. Therefore, according to Theorem 2.2 in [32] it has harmonic solutions with the period kT and $(k+1)T$ for a sufficiently high natural number, k. These solutions coincide as $t \to \infty$ because of asymptotity. If a system has a solution with period kT and $(k+1)T$ it should have a solution with period T. Since this solution is within G and because all solution converge to each other, all solutions converge to the periodic solution with the period of T. Q.E.D.

Appendix B

Publications

- **M. Sharifkhani**, M. Sachdev, " Segmented Virtual Ground Architecture for Low-power Embedded SRAM", *Accepted with modification in IEEE Transactions on VLSI (IEEE T-VLSI).*
- **M. Sharifkhani**, M. Sachdev, " SRAM Cell Data Stability: A Dynamic Perspective", *Submitted to IEEE Journal of Solid State Circuits (IEEE JSSC), June 2006.*
- **M. Sharifkhani**, M. Sachdev, " A Phase-Domain Continuous-Time 2nd-Order $\Delta\Sigma$ Frequency Digitizer", *Proceedings of IEEE Custom Integrated Circuit Conference September 2006 (IEEE CICC'06).*
- **M. Sharifkhani**, M. Sachdev, " A Low-static and Dynamic Power SRAM Based on Segmented Virtual Grounding", *Submitted to IEEE International Solid-State Circuit Conference 2007 (IEEE ISSCC'07).*
- **M. Sharifkhani**, M. Sachdev, " A Phase-Domain Continuous-Time 2nd-Order $\Delta\Sigma$ for Frequency Digitization", *Presented at IEEE Symposium on Circuits and Systems, pp. 3434-3438, May 2006 (IEEE ISCAS'06).*

- **M. Sharifkhani**, S. M. Jahinnuzaman, M. Sachdev, " Dynamic Data Stability in SRAM Cells and its Implications on Data Stability Tests", *Proceedings of IEEE International Workshop on Memory Technology, Design, and Testing, pp. 55-61, 2006 (IEEE MTDT'06).*

- **M. Sharifkhani**, M. Sachdev, " A Low Power SRAM Architecture Based on Segmented Virtual Grounding", *Accepted at IEEE Symposium on International Symposium on Low Power Electronics and Design 2006 (IEEE ISLPED'06).*

- **M. Sharifkhani**, " A Frequency Digitizer Based on the Continuous Time Phase Domain Noise Shaping", *Proceedings of IEEE Symposium on Circuits and Systems, pp. 1060-3 ,2004 (IEEE ISCAS'04).*

References

[1] Y.-W. Huang, T.-C. Chen, C.-H. Tsai, C.-Y. Chen, T.-W. Chen, C.-S. Chen, C.-F. Shen, S.-Y. Ma, T.-C. W. andBing Yu Hsieh, H.-C. Fang, and L.-G. Chen, "A 1.3tops h.264/avc single-chip encoder for hdtv applications," *IEEE ISSCC Digest of Tech. Papers*, Feb. 2005.

[2] J. Hart, S. Y. Choe, L. Cheng, C. Chou, A. Dixit, K. Ho, J. Hsu, K. Lee, and J. Wu, "Implementation of a 4th-generation 1.8ghz dual-core sparc v9 microprocessor," *IEEE ISSCC Digest of Tech. Papers*, Feb. 2005.

[3] International technology roadmap for semiconductors - 2005 (ITRS-2005). [Online]. Available: http://public.itrs.net/Common/2005/

[4] J. Rabaey, A. Chandrakasan, and B. Nicolic, *Digital Integrated Circuits A Design Perspective*, 2nd ed. Prentice Hall, 2003.

[5] M. Sharifkhani, "A frequency digitizer based on the continuous time phase domain noise shaping," *Proceedings of International Symposium on Circuit and Systems (ISCAS)*, pp. 1060–1063, May 2004.

[6] M. Sharifkhani and M. Sachdev, "A phase-domain 2nd-order continuous time *delta − sigma*-modulator for frequency digitization," *Proceedings of International Symposium on Circuit and Systems (ISCAS)*, p. 3434, May 2006.

[7] ——, "A phase-domain continuous-time 2nd-order delta-sigma frequency digitizer," *Accepted for presentation at IEEE Custom Integrated Circiut Conference (CICC)*, Sept. 2006.

[8] S. R. Norsworthy, R. Schreier, and G. C. Temes, *Delta-Sigma Data Converters: Theory, Design, and Simulation.* IEEE Pres, 1997.

[9] International technology roadmap for semiconductors - 2003 (ITRS-2003). [Online]. Available: http://public.itrs.net/

[10] R. Islam, A. Brand, and D. Lippincott, "Low power SRAMs for handheld products," *IEEE Symposium on Lowpower Electronics and Design*, pp. 198–202, Oct. 2005.

[11] Y. Nakagome, M. Horiguchi, T. Kawahara, and K. Itoh, "Review and prospect of low-voltage RAM circuits," *IBM J. Res. and Dec.*, vol. 47, no. 5/6, pp. 525–552, Sept. 2003.

[12] S. Narenda and A. Chandrakasan, *Leakage in Nanometer CMOS Technology.* Springer-Verlag, 2006.

[13] C. Kim, J. Kim, S. Mukhopadhayay, and K. Roy, "A forward body bias low-loekage SRAM cache: Device and architecture considerations," *IEEE Symposium on Low-power Electronics and Design*, pp. 6–10, Oct. 2003.

[14] K. Osada, Y. Saitoh, E. Ibe, and K. Ishibashi, "16.7 fA/cell tunnel-leakage-suppressed 16 Mb SRAM for handling cosmic-ray-induced multi-errors," *IEEE ISSCC Digest of Tech. Papers*, p. 302, Feb. 2003.

[15] K. Nii, Y. Tsukamoto, T. Yoshizawa, S. Imaoka, Y. Yamagami, T. Suzuki, A. Shibayama, H. Makino, and S. Iwade, "A 90-nm low-power 32-kb embedded SRAM with gate leakage suppression circuit for mobile applications," *IEEE J. Solid-State Circuits*, vol. 39, no. 4, pp. 684–692, Apr. 2004.

[16] C. F. Hill, "Noise margin and noise immunity in logic circuits," *Microelectron Journal*, pp. 16–21, Apr. 1968.

[17] J. Lohstroh, "Static and dynamic noise margins of logic circuits," *IEEE J. Solid-State Circuits*, vol. SC-14, pp. 591–598, 1979.

[18] J. Lohstroh, E. Seevinck, and J. D. Groot, "Worst-case static noise margin criteria for logic circuits and their mathematical equivalence," *IEEE J. Solid-State Circuits*, vol. SC-18, pp. 803–807, 1983.

[19] C. Mead and L. Conway, *Introduction to VLSI systems.* Addison Wesley, 1980.

[20] E. Seevinck, F. List, and J. Lohstroh, "Static-noise margin analysis of MOS SRAM cells," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 748–754, 1987.

[21] B. S. Amrutur and M. A. Horowitz, "Fast low-power decoders for SRAM's," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1506–1515, 2001.

[22] R. J. Evans and P. D. Franzon, "Energy consumption modeling and optimization for SRAM's," *IEEE J. Solid-State Circuits*, vol. 30, pp. 571–579, June 1995.

[23] B. S. Amrutur and M. A. Horowitz, "A replica technique for wordline and sense control in low-power SRAM's," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1208–1219, 1998.

[24] B. Yang and L. Kim, "A low power SRAM using hierarchical bit-line and local sense amplifiers," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1366–1376, June 2005.

[25] K. Kanda, S. Hattori, and T. Sakurai, "90 % write power-saving SRAM using sense-amplifying memory cell," *IEEE J. Solid-State Circuits*, vol. 93, pp. 929–933, 2004.

[26] M. Sharifkhani and M. Sachdev, "Sram cell data stability: A dynamic perspective," *Submitted to IEEE Journal of Solid-State Circuits*, 2006.

[27] A. Bhavnagarwala, X. Tang, and J. Meindl, "The impact of intrinsic device fluctuations on CMOS SRAM cell stability," *IEEE J. Solid-State Circuits*, vol. 36, pp. 657–665, 2001.

[28] K. Itoh, *VLSI Memory Chip Design*. Springer-Verlag, 2001.

[29] H. Veendrick, "The behavior of flip-flops used as synchronizers and prediction of their failure rate," *IEEE J. Solid-State Circuits*, vol. SC-15, no. 2, pp. 169–176, Apr. 1980.

[30] M. Matsumiya *et al.*, "A 15-ns 16-Mb CMOS SRAM with interdigitated bit-line architecture," *IEEE J. Solid-State Circuits*, vol. 27, pp. 1497–1503, 1992.

[31] T. Wada, S. Rajan, and S. Przybylski, "An analytical access time model for on-chip cache memories," *IEEE J. Solid-State Circuits*, vol. 27, pp. 1147–1156, 1992.

[32] V. A. Pliss, *Nonlocal Problems of the theory of oscillations*. Academic Press, 1966.

[33] A. Agarwal, B. Paul, S. Mukhopadhyay, and K. Roy, "Process variation in embedded memories: Failure analysis and variation aware architecture," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1804–1813, 2005.

[34] H. K. Khalil, *Nonlinear Systems*, 2nd ed. Prentice Hall, 1996.

[35] N. Shibata, "A switched virtual-gnd level technique for fast and low power srams," *IEICE Trans. Electron.*, vol. E80-C, pp. 1598–1607, 1997.

[36] K. Zhang *et al.*, "A 3-GHz 70Mb SRAMin 65nm CMOS technology with integrated column based dynamic power supply," *IEEE ISSCC Digest of Tech. Papers*, pp. 474–475, Feb. 2005.

[37] M. Sharifkhani and M. Sachdev, "A low power SRAM architecture based on segmented virtual grounding," *Accepted for presentation at IEEE Symposium on Low-Power Electronics and Design (ISLPED)*, Oct. 2006.

[38] B. Calhoun, A. Wang, and A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1778–1785, 2005.

[39] M. Sharifkhani and M. Sachdev, "Dynamic data stability in SRAM cells and its implications on data stability tests," *Presented at IEEE International Workshop on Memory Technology, Design, and Testing 2006 (IEEE MTDT'06)*, 2006.

[40] A. Pavlov, M. Sachdev, and J. D. Gyvez, "Weak cell detection in deep-submicron srams: A programmable detection technique," *IEEE J. Solid-State Circuits*, pp. 2334 – 2343, Oct. 2006.

[41] H. Mahmoodi, S. Mukhopadhyay, and K. Roy, "Estimation of delay variations due to random-dopant fluctuations in nanoscale CMOS circuits," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1787–1796, Sept. 2005.

[42] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel, and M. Bastian, "Resistive-open defect injection in SRAM core-cell: analysis and comparison between 130nm and 90nm technologies," *Proceedings of 42nd Design Automation Conference*, pp. 857 – 862, 2005.

[43] K. Itoh, K. Sasaki, and Y. Nakagome, "Trends in low-power SRAM circuit technologies," *Proc. IEEE*, vol. 83, pp. 524–543, 1995.

[44] M. Sharifkhani and M. Sachdev, "Segmented virtual ground architecture for low-power embedded sram," *Accepted with modifications in IEEE Transactions on VLSI*, 2006.

[45] H. Mizuno and T. Nagano, "Driving source-line cell architecture for sub-1-v highspeed low-power applications," *IEEE J. Solid-State Circuits*, vol. 31, pp. 552–557, 1996.

[46] K. W. Mai *et al.*, "Low-power SRAM design using half-swing pulse-mode techniques," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1659–1671, 1998.

[47] R. Gu and M. Elmasry, "Power dissipation analysis and optimization of deep submicron CMOS digital circuits," *IEEE J. Solid-State Circuits*, vol. 31, pp. 707–713, May 1996.

[48] J. L. Hennessy and D. Patterson, *Computer Architecture a Quantitative Approach*. Morgan Kaufman, 1996.

[49] M. Yamaoka *et al.*, "90-nm process-variation adaptive embedded SRAM modules with power line floating write technique," *IEEE J. Solid-State Circuits*, vol. 41, no. 3, pp. 705–711, 2006.

[50] B. Calhoun and A. Chandrakasan, "A 256kb subthreshold sram in 65nm," *IEEE ISSCC Digest of Tech. Papers*, pp. 628–629, Feb. 2006.