AN AREA-EFFICIENT, HIGH-PERFORMANCE, LOW-POWER MULTI-PORT CACHE MEMORY ARCHITECTURE

By

HASSAN BAJWA

A dissertation submitted to the Graduate Faculty in Engineering in partial fulfillment
of the requirements for the degree of Doctor of Philosophy,
The City University of New York

2007

UMI Number: 3283200

UMI Microform 3283200

Copyright 2007 by ProQuest Information and Learning Company.
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

© 2007

HASSAN M BAJWA

All Rights Reserved

This manuscript has been read and accepted for the Graduate Faculty in Engineering in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

Professor Xinghao Chen

Date

Chair of Examining Committee

Professor Mumtaz Kassir

Date

Executive Officer

Professor Mohamed M. Zahran

Professor Norman Scheinberg

Professor Ümit Uyar

Dr. Yi Tan

Supervision Committee

THE CITY UNIVERSITY OF NEWYORK

Abstract

AN AREA-EFFICIENT, HIGH-PERFORMANCE, LOW-POWER MULTI-PORT CACHE MEMORY ARCHITECTURE

By

Hassan Bajwa

Advisor: Professor Xinghao Chen

In recent years significant research activities have been geared toward reducing bottleneck caused by slow READ and WRITE operations of conventional memories. Ever increasing power-hungry applications and hardware have stretched present memory architecture and technology to the limit. Increasing cache sizes and the number of cache levels between microprocessors are no longer a practical solution as low cost as well as low access latency imposes a restriction on the growth of cache size. Dual-port memory technologies, implemented with explicit duplication of word and bit lines for each port, have been widely applied to multi-core processors in recent years. Since the silicon areas used by word and bit lines dominate entire memory area, duplicating the word and bit lines results in large silicon area and increases bitline discharge and power dissipation [1]. It is not uncommon to have dual-core processors or Systems-on- Chip (SOC) applications where memory occupy more then 50% of the chip area [2]. Recently, as technology scaled down in the sub-micron regime, power dissipation due to leakage energy has become an important

consideration in high performance microprocessor design. Leakage power dissipation, usually proportional to chip area has emerged as a dominant source of energy consumption [3, 4].

We have proposed a dynamically configured area-efficient memory architecture that can improve the performance of a traditional hardwired dual-port memory without explicitly duplicating the word and bit lines for the second port. In other words, the dual-ports in our design share the word and bit lines used in the traditional hardwired single-port memory. The new area-efficient dual-port memory technology employs a dynamically configured isolation circuit to divide a conventional memory into two virtually isolated blocks, thus allowing dual-port accesses simultaneously. These two virtually isolated memory blocks work just like two conventional single-port memories. Since no word and bit lines are explicitly added for the second port, the silicon size of the hardwired dual-port memory can be reduced almost to half, while allowing less power consumption and access latency at the same time. The dynamic isolation of a memory block into two virtually isolated blocks is realized, utilizing the real time memory access addresses on the two ports.

Acknowledgements

This thesis would not have been possible without the support, generosity and kindness of several people. My advisor and mentor, Professor Xinghao Chen played a critical role in guiding me throughout the course of my doctorate. I also wish to extend my gratitude and appreciation to my loving and supportive wife and my children who were always a constant source of inspiration. I would like to extend my sincerest appreciation to my parents and in-laws for their prayers and thoughtfulness.

Most of all I am thankful to God for allowing me to overcome obstacles and for making all this possible.

Table of Contents

ABSTRACT .... **IV**

ACKNOWLEDGEMENTS .... **VI**

LIST OF FIGURES .... **X**

LIST OF TABLES .... **XII**

CHAPTER 1 INTRODUCTION **1**

1.1 BACKGROUND 2

1.1.1 Area 2

1.1.2 Power 3

1.1.3 Wire or interconnect delay 4

1.2 DISSERTATION OBJECTIVE 4

1.3 ORGANIZATION 5

CHAPTER 2 OVERVIEW OF SRAM **7**

2.1 SRAM CELL 7

2.2 OPERATION OF SRAM CELLS 9

2.2.1 Read Operation of Conventional Single-Port SRAM Cell 9

2.2.2 Write Operation of Single-Port SRAM Cell 10

2.3 LIMITATION OF SINGLE-PORT MEMORY 10

2.4 DUAL-PORT MEMORY CELL 11

2.5 DUAL-PORT MEMORY OPERATIONS 12

2.5.1 Read / Read on same cell 12

2.5.2 Read / Write on same cell 14
2.5.3 Write / Write on same cells 14
2.5.4 Operations on two different cells 14
2.6 DUAL-PORT MEMORY APPLICATIONS 15
2.7 LIMITATION OF DUAL-PORT MEMORY 16
SUMMARY 16

CHAPTER 3 LEAKAGE CURRENT AND POWER DISSIPATION IN SRAM 17

3.1 SRAM LEAKAGE POWER 18
3.2 MEMORY CELL STATIC LEAKAGE CURRENT 19
3.3 LEAKAGE CURRENT IN IDLE MEMORY 20
3.4 LEAKAGE CURRENT DURING READ AND WRITE OPERATION 23
SUMMARY 24

CHAPTER 4 LITERATURE REVIEW 25

INTRODUCTION 25
4.1 CIRCUIT AND DEVICE APPROACHES TO REDUCE LEAKAGE CURRENT 25
4.1.1 Multi-Threshold CMOS 26
4.1.2 Dual-Threshold SRAM 27
4.1.3 Gated-Vdd SRAM 28
4.2 ARCHITECTURAL TECHNIQUES TO REDUCE LEAKAGE POWER 30
SUMMARY 31

CHAPTER 5 SRAM ORGANIZATION AND ARCHITECTURE 33

5.1 TERMINOLOGY 33
5.2 SRAM ARCHITECTURE 35
5.3 CACHE ADDRESSING 37
5.4 CACHE SIZING AND ASSOCIATIVITY 38
5.5 SRAM PARTITIONING 40
5.6 HIGH-PERFORMANCE MEMORIES TIMING AND DELAY ANALYSIS 41
5.7 PERFORMANCE AND DELAY ANALYSIS OF CACHE USING SUB-BANKING 44
SUMMARY 45

CHAPTER 6 DYNAMICALLY CONFIGURED MEMORY 46

6.1 DYNAMICALLY CONFIGURED MEMORY ARCHITECTURE 46
6.2 ISOLATION NODE PLACEMENT 49
6.3 DYNAMICALLY CONFIGURED CACHE ADDRESSING 55

6.4 DYNAMIC MEMORY PARTITIONING ALGORITHM 55
6.4.1 Fine-Grain DMP-1 56
6.4.2 Course-Grain DMP-2 59
6.5 CELL STRUCTURE 60
6.6 ADVANTAGES OF DYNAMICALLY CONFIGURED MEMORY 60
6.7 DYNAMIC MEMORY OPERATIONS 61
6.7.1 Read / Read on same Cell 61
6.7.2 Read / Write on same Cell 62
6.7.3 Write / Write on same Cells 62
6.7.4 Operations on Two Different Cells 63
6.8 DELAY ANALYSIS OF DYNAMICALLY CONFIGURED CACHE 63
6.8.1 Pre-Charge whole Memory 64
6.8.2 Precharging required Array or Bank 65
6.8.3 Pre-charge Sub-Array after Isolation Node is Setup 66
Summary 68

CHAPTER 7 DEVICE MODELING 69

INTRODUCTION 69
7.1 ADDITIONAL DEVICE MODELING 69
7.2 SIMULATION MODEL 71
7.2.1 Wire Capacitance and Resistance 71
7.2.2 MOS Device Modeling 73
SUMMARY 79

CHAPTER 8 ANALYTICAL AND SIMULATION RESULTS OF DMP IMPLEMENTATION 80

CHAPTER 9 CONCLUSION 86

REFERENCES 89

List of Figures

Figure 1. 6-Transistor SRAM Cell ... 8
Figure 2. 8-Transistors Dual-Port Memory Cell ... 13
Figure 3. Leakage Current Trends as Technology Scales from 180-to 32nm ... 19
Figure 4. Leakage Current in Single-Port SRAM Cell ... 21
Figure 5. Leakage Current in Dual-Port SRAM Cell ... 22
Figure 6. MTCMOS Circuit Structure ... 27
Figure 7. Single-Port SRAM with High-V_{TH} Transistors ... 28
Figure 8. Gated-Ground SRAM Cell ... 29
Figure 9. Block Diagram of SRAM ... 36
Figure 10. Hierarchical Memory Architecture ... 37
Figure 11. Block Diagram of Cache ... 38
Figure 12. A Cache Subsystem Organizations ... 40
Figure 13. Conventional Single-Port 2-Way Set-Associative Cache ... 42
Figure 14. SRAM Memory Operation ... 43
Figure 15. Memory Operation of SRAM With Sub-Banking ... 44
Figure 16. Placement of Isolation Nodes and Control Lines ... 47
Figure 17. Memory Operation Timing with DMP ... 48
Figure 18. Block Diagram of Dual-Port Memory with Dynamic Partitioning ... 50
Figure 19. ICL and Isolation Nodes Placement ... 52
Figure 20. Structural Configuration of Dual-Port Cache Architecture ... 53
Figure 21. A Cache Subsystem with DMP ... 54
Figure 23. Generic DMP Model ... 57
Figure 24. Generic DMP Model with Two Isolation Nodes ... 58
Figure 25. DMP with Single Isolation Node ... 59

Figure 26. Memory Operation Timing with DMP 64
Figure 27. Dynamically Configured SRAM with Bank Pre-charging....................... 66
Figure 28. Partial Sub-Array Precharging 67
Figure 29. Bit line Resistance as Technology Scales Down..................................... 73
Figure 30. R_{dsw} Trend as Technology Scales-Down form 90nm to 32nm................. 75
Figure 31.Technology Scaling Impact on MOS Devices and Transmission Gate Resistance 76
Figure 32. MOSFET Capacitive Model... 77
Figure 33. RC Model of Active Bit line with Isolation Node..................................... 78
Figure 34. Bit line Delay With/Without Isolation Nodes 82
Figure 35. Delay Analysis with DMP-1 83
Figure 36. Delay Analysis with DMP-2 84

List of Tables

Table 1. Wire Capacitance and Resistance as technology scales from 130 to 32nm . 72
Table 2. Device parameters extracted from PTM 74
Table 3. MOS Device Resistance 76
Table 4. Capacitance of MOS Devices in Sub-Micron Regime 78

Chapter 1 Introduction

Cache is a smaller, faster and expensive memory located closer to Central processing unit (CPU) and holds a fraction of total memory that the processor is most likely to need next. Despite the advances in integrated circuit design, overall system performance is seriously impeded by bottleneck due to memory latency. Most modern microprocessors use multi-level hierarchical large on-chip cache to bridge performance gap between processor and main memory, however larger on-chip cache results in larger die area and high fabrication cost [5]. Static random access memory (SRAM) also appears to be a major source of power dissipation as it contains highly capacitive bit line and is accessed frequently. In recent years, several architectural and circuit techniques [6-10, 15] have been developed to enhance performance, reduce area power consumption, and other design parameters. In this chapter we are going to present various design parameters and limitations of conventional cache architecture.

1.1 Background

Technology scaling has provided remarkable improvements in performance of electronic circuits. As feature size shrinks with every subsequent generation of technology, SRAM transistor density, on-chip cache size and leakage current increases. Submicron technologies impose new challenging design constraints, keeping memory cell smaller while designing low power large cache has proven to be extremely difficult [10].

1.1.1 Area

Ever increasing power-hungry applications and hardware have stretched present memory architecture and technology to the limit. Amount of real-estate devoted to on-memory is continuously increasing. With multi-core processor technologies (such as IBM Cell [20], AMD Opteron 2005 [22] , Intel Itanium®-2 [18] and Pentium®-D [26] that are capable of integrating two processor cores on the same die) moving into the mainstream applications, making cache memory highly accessible by multiple processors becomes a necessity, while being technologically challenging. Multi-port cache memory would provide the needed accessibility to multiple CPUs. Historically, multi-port memory designs have been implemented with dedicated word and bit lines for each port. Since the silicon areas used by word and bit lines almost cover the entire memory area [1], duplicating the word and bit lines results in multiplying the silicon areas used by a multi-port memory system with the number of ports.

An overview of state of the art multi-core processors reveals that level 2 (L2) cache

occupies more than half of the chip area in these processors [16-19]. International Technology Roadmap for Semiconductors 2001 predicted that cache will occupy 90% of chip area by 2013 [59]. Trends in recent System on-chip (SOC) designs have proven that memory design is as important as processor design in order to achieve performance [40, 41 and 42].

1.1.2 Power

Power along with performance has become a major design consideration in mobile and embedded computing products. Motivation for designing an efficient memory system stems from the fact that on-chip memory size is continuously increasing and, large cache memories have a significant impact on the overall system energy consumption. Sub-threshold leakage current (flowing from drain to source, even when the transistor is not operating) emerged as a dominant leakage current in high performance microprocessors [3,4] where L1 and L2 cache occupy majority of die area. As transistor size scales down, power dissipation becomes a serious problem that limits overall system performance [11, 12]. In high performance systems employing multi-core technologies, bit line leakage current can contribute to as much as 50% of overall cache memory leakage power [8-10, 63]. The trend of using multi-port cache in modern microprocessor technologies [10] has exuberated this further, as leakage current dissipation is roughly proportional to area of the circuit [17]

1.1.3 Wire or interconnect delay

Delay in memory is strongly dependent upon capacitance of the word and bit lines. As feature size shrinks by half, area of device is expected to be reduced to 1/4th. In nano meter design wires accounts for the majority of overall system delay and hence it is naive to assume that the capacitance is not a linear function of length. Resistive Capacitive (RC) delay is a major concern in submicron technologies and wire redesigning has to be done to keep the wire delay as small as possible. Several research efforts have been made to reduce bit line capacitance and thus improve the performance of SRAM [11, 12, and 54]. Intra memory interconnect (bit-lines and word-lines) dominates energy dissipation in traditional on-chip memories; increase in interconnect length due to growing cache size will adversely effect the performance of memory. Any technique that reduces interconnect length will also reduce latency and save power.

1.2 Dissertation Objective

This dissertation provides circuit and architectural solutions to reduce area overhead of SRAM caches and, thus, reduces power dissipation and latency in conventional SRAM architectures. Various circuit design approaches to lower leakage power and increase access speed of memories are studied and a novel architectural approach to reduce area and bit line latencies is presented.

This dissertation has achieved the mentioned objectives by the following steps:

- Presenting a design of a novel dynamically configured cache memory.

- Proposing an architecture that reduces leakage current and static power dissipation to half of traditional dual-port memories.
- Proposing an architecture that is capable of reducing cache size to half of traditional memory architecture.
- Presenting the device model and characteristics in sub-micron technology.
- Presenting analytical and simulation results.

1.3 Organization

This dissertation is organized as follows. Chapter 2 presents an overview of SRAM cell. Designs and operations of SRAM single and dual-port memory cells are explained. Limitations of conventional single-port and dual-port memory cells are explored in this chapter. In Chapter 3 we present an analytical model for static power dissipation in cache memories. We also explore existing architecture and circuit techniques to reduce leakage power in memory circuits are presented. In Chapter 4 we review several existing architectural and circuit techniques which reduce leakage current and bit line latency. Chapter 5 presents conventional cache architecture and organization. Cache associativity, memory partitioning and hierarchical memory architecture is detailed in this chapter. In the following chapter we l present newly proposed dynamically configured cache architecture and observe the advantages of this architecture in detail. Leakage power in conventional and the newly proposed dynamically configured memory is discussed in this chapter. Detailed analytical analysis of static power dissipation in newly proposed architecture is also presented in this chapter.

Accurate characterization of future technologies and device modeling at an early stage is extremely important in predicting behavior and performance of next generation devices. Chapter 7 presents the device model for the proposed dynamically configured memory using Berkeley Predictive technology model (BPTM). We present a device model for proposed dynamically configured memory. Device resistance as well as bit line resistance and capacitances are calculated as technology scales down beyond 100nm. Finally, the results are presented in Chapter 8 followed by conclusion in Chapter 9.

Chapter 2 Overview of SRAM

Introduction

More performance is expected from memories as multi-core processors capable of executing multiple processes/threads are implemented. Different memories with different capabilities are mostly designed for specific application. Because of the importance of fast on-chip memory, high speed SRAM is the first choice of designers in cache design. Though SRAM cells are four times the size of Dynamic random access memory (DRAM) [62], computer hierarchy uses a small amount of very fast SRAM for cache to hold most frequently accessed information. In this chapter we are going to focus on single SRAM cell design; we first describe a single-port SRAM cell design, its operation and limitations. We continue by describing a dual-port memory cell, present its operations and guide the reader through limitations of this dual-port memory cell.

2.1 SRAM Cell

A classical single-port SRAM cell employs 6-transistors, a word line (*WL*) and a pair

of bit lines. When the word line (*WL*) is selected, SRAM cell is connected to the pair of compliment bit lines (*BL* and $\overline{BL}$) via pair of pass transistors *T5* and *T6*. Based on conventional cross-coupled inverter design, SRAM cell (shown in Figure 1) is the most frequently implemented due to its low quiescent power and greater soft error resistance [1].

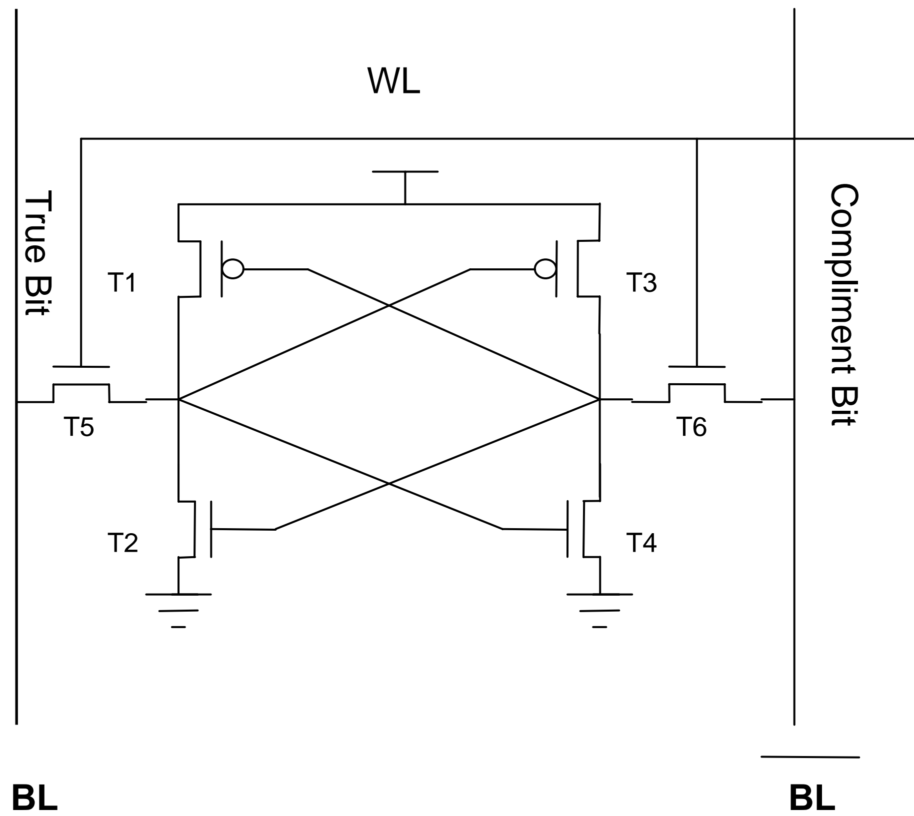


Figure 1. 6-Transistor SRAM Cell

2.2 Operation of SRAM Cells

The functions of the cells are to hold, read and write data. Same bit-lines pair is used for both read and write operations. It is possible to perform read and write operations using only one bit line, but it would require a full swing of logical value of data resulting in large transistor size and slow operations. The following subsections briefly discuss read and write operations of a memory cell.

2.2.1 Read Operation of Conventional Single-Port SRAM Cell

The read operation starts by address being input to the memory and clock rising high. For both the read and the write operations word line is selected high and access transistors are turned on. Before the read operation is performed on memory cell both bit lines are pre-charged high, the evaluation period then follows in which the word line goes high and the transfer devices (T5 and T6 in Figure 1) turns on. SRAM cell is then connected to the pair of compliment bit lines (BL and $\overline{BL}$ in Figure 1), which in turn reads the value stored in the cell. If a *0* is stored in the cell bit line is discharged and the compliment bit line remains high. If a *1* is stored in the cell, the bit line remains high and the complaint bit line is discharged. The difference between the bit and the compliment bit lines are fed into the sense amplifier to generate a valid output stored in the buffer. During the read operation the pre-charge circuit is turned off.

Delay in memory access is one of the major bottlenecks in embedded system performance and is generated from the following:

- Decoder: Decoding the address.

- Word line: Driving the specific word line high.
- Bit line: Placing the contents of cell on the bit line. (The longer the bit lines are more delay it cause).
- Sense Amplifier: Sensing the voltage at the bit line and driving the signal to data bus.

In most of the high performance memories, the bit line only swings a small amount in read operations. Delay in high performance memories can be controlled by reducing the size of the bit line and thus designing smaller memories and sense amplifiers that respond to a very small voltage swing. SRAM memories can restrict the bit line voltage swing to 10% of V_{DD} [25, 56].

2.2.2 Write Operation of Single-Port SRAM Cell

As mentioned earlier, in SRAM a differential bit line pair is attached to each cell. Before a write operation, these bit lines are pre-charged to the high state. The transfer devices are NFET transistors (T5 and T6 in Figure1) that can efficiently write a *0*. Thus write operation is always accomplished by forcing a *0* at either the bit or compliment bit line. To write a *1*, the compliment bit bar is forced to low and to write a *0,* the bit line is forced to low.

2.3 Limitation of Single-Port Memory

Memory latency is one of the major performance bottle-neck in modern

microprocessor design. Multi-core technologies and several data processing subsystems demand cache memory that allows parallel and high bandwidth access. As mentioned above single-port memory cells have only one port thus only one operation can be performed on a cell at a time. While an operation is being performed no other cell can be accessed from the block. Memory bandwidth can be increased by using effective caching techniques including memory bus sharing and implementation of cache coherence protocols [27], efficient readout and layout algorithms [1, 56], hierarchical cache memory [29,57], high-speed memory clock [30], and multiple ports to access memory partitions in parallel [1, 30]. Among these methods the dual-port memory architecture is the most extensively-used approach.

2.4 Dual-Port Memory Cell

Demand of on-chip shared memory that allows independent, parallel and high bandwidth access have risen significantly over the years. Dual-port memory cell is the most commonly used multi-port memory cell that provides required access to multi-processor applications. In addition to six transistors, a pair of bit-lines and a word line in conventional SRAM cell, a dual-port memory cell uses two additional pass transistors: an additional bit line pairs and a world line to provide much needed simultaneous access. Figure 2 shows the classic dual-port memory architecture, in which each SRAM cell is accessible by two ports with dedicated word and bit lines to each. The addition of the word and bit lines and access transistors *T7* and *T8* increase silicon area significantly.

Dual-port memories are implemented with explicit duplications of the word and bit

lines for each port (hence hardwired dual-port) or via software support on top of a single physical port (hence soft dual-port). Scalability of a multi-port cell is much trickier than a one port cell. Due to area overhead technologies like power5 [11] and Itanium [16] have almost half of the processor area consumed by cache.

2.5 Dual-Port Memory Operations

Dual-port memory is capable of performing two simultaneous operations in one clock cycle. As long as these operations are performed in different cells, dual-port memory can read and write data twice the speed of single-port memory cell without any further consideration. In dual-port memory it is possible that both ports access the same cell at the same time, which can make a dual-port memory cell unstable. To resolve this issue, pseudo multi-port memory utilizes high-speed clock to generate multiple memory accesses in a single clock cycle. Pseudo multi-port memory uses a clock pulse generated within the memory to allow same cell access in one clock cycle.

2.5.1 Read / Read on same cell

As in single-port memories in dual-port memories bit lines are pre-charged high before any operation is performed. The read operation starts by address being input to the memory and clock going high. Two world lines can be selected high and access transistors T5, T6, T7 and T8 are turned on which in turn reads the value stored in the cell shown in figure 2.

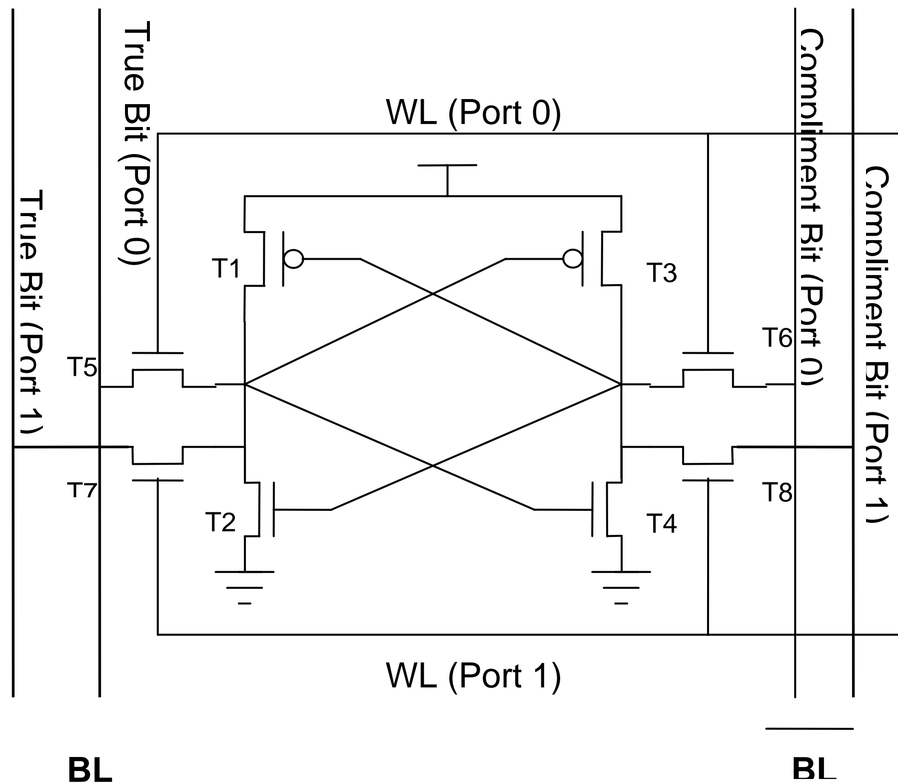


Figure 2. 8-Transistors Dual-Port Memory Cell

If a *0* is stored in the cell, the bit line is discharged and the compliment bit line remains high. If a *1* is stored in the cell, the bit line remains high and the compliment bit line is discharged. The difference between the bit and compliment bit lines are fed into the sense amplifier to generate a valid output stored in the buffer. During the read operation the pre-charge circuit is turned off.

When two read operations are performed on a cell at the same time this cell has to be able to sink twice the speed of single-port memory. The cell stability of multi-port memory cell is defined by the ratio of the strength of pull-down transistor divided by the combined strength of all transfer devices.

2.5.2 Read / Write on same cell

Since there is no way of knowing whether the data accessed by read operation has already been updated, read/write operation on the same cell is therefore extremely critical. It is not possible to know whether a data is valid when both "Read" and "Write" operations are performed on a cell in one clock cycle. As mentioned above, pseudo multi-port memories can use two clocks generated within the memory to perform this operation. With the first clock generated in the memory, the first operation is performed once this operation is completed, and then only the second operation is performed. These operations need to be defined carefully, because the invalid data will be read, if read is performed before data is written.

2.5.3 Write / Write on same cells

Write requests consist of new information to be written to a location specified by the address. If the location specified by the address is stored in the cache, a hit occurs, otherwise, a miss results and the request is forwarded to the next memory in the hierarchy. It is critical that only one write operation is performed on a specific location in one clock cycle, otherwise integrity of the data cannot be guaranteed.

2.5.4 Operations on two different cells

The biggest advantage of multi-port memory is multi-tasking. Memory speed can be enhanced by allowing multiple processors and threads to access cache simultaneously. Dual-port memory allows memory operation to be performed on any two memory

cell in a memory core simultaneously.

2.6 Dual-Port Memory Applications

Many microprocessors, multi-core technologies, as well as multi-media applications contain several data processing subsystems. Dual-port (as well as multi-port) memory architecture has been applied with instruction and data cache implementations in multi-core processors in recent years. These systems require on-chip shared cache that can provide independent, parallel and high bandwidth access. Dual and multi-port caches are capable of providing required accessibility by increasing the number of ports. Therefore, it doubles (in the case of the dual-port) or multiplies (in the case of multi-port) the speed of a single-port cache. IBM Power5® is a single-die silicon solution which contains two identical processor cores, each supporting two logical threads [19]. Power5® uses separate buses for READ and WRITE with L3 cache. Its L2 cache is a dual-port design and is implemented on the same die as three identical slices with separate controllers. Each processor core can independently access each controller. Intel Itanium®-2 uses a combination of dual-port and four-port caches to support the dual-core processors on the chip [16]. A 4-port data cache allows four M operations per cycle. Itanium®-2 L1 cache uses separate memory ports for different functions: Ports 0 and 1 support load operation and Ports 2 and 3 support integer operations. L2 cache is shared for data and instructions.

The new Intel Montecito dual-core dual-thread Itanium® processor uses 3 levels of cache [21]. The L1 cache uses two separate caches, one for instruction (L1I) and the

other for data (L1D). L1D allows two integer load and store with its dual-port at the same time, while L1I uses dual-ported tags and single data port to support simultaneous demand and pre-fetch accesses. The L2 cache also has dedicated L2I and L2D caches, with the tag array supporting four-port independent accesses in the same cycle.

2.7 Limitation of Dual-Port Memory

Multi-port memory cells are significantly larger then single-port cells. An eight transistors dual-port cell takes up to twice the area of six transistors single-port cell. This extra space is due to extra word lines and bit lines [1]. As the memory size grows larger, the high capacitance bit line and I/O lines also become larger. The longer the lines, the larger will be the swing and memory will not only become slower but will also consume more energy.

Summary

An overview of static CMOS RAM cells has been presented in this chapter. We observed that multi-port memory can easily be designed by adding a pair of pass transistors a duplicate bit lines and a word-line. Multi-port memory cells are significantly larger than single-port memory cell because as the number of port increases so does the word and bit lines that occupy majority of chip area. We will go over leakage current and power dissipation in the SRAM in the next chapter.

Chapter 3 Leakage Current and Power Dissipation in SRAM

Introduction

Sub-threshold leakage current flowing from drain to source, even when the transistor is not operating, emerged as a dominant leakage current in high performance microprocessors [3-4] where L1 and L2 cache occupy majority of the die area [16-20]. Large on-chip cache memories are needed as high performance multi-core technologies are becoming more popular. Technology scaling and increasing number of on-chip devices has worsened the already serious problem of power dissipation in high performance systems [7, 47, 49-52, 55].

In this chapter, we focus on present SRAM technologies and analyze leakage current in them. As technology scales down, the need of optimizing leakage power has become more significant. We also study the conventional cache organization and further estimate leakage current in SRAM.

3.1 SRAM Leakage Power

Historically, the primary source of power dissipation has been dynamic energy due to the charging/discharging of load capacitances as a device switches. As technology scales down in deep submicron dimensions, transistors gets smaller, gate oxide gets thinner, and as the result Metal-Oxide-Semiconductor Field-Effect Transistor (MOSFET) suffers from conventional short-channel effects [3, 4, 7-10, 13-15]. Channel length modulation (CLM), carrier velocity saturation, drain-induced-barrier lowering (DIBL), reverse short-channel (RSC) effect, and substrate current induced body effect (SCBE) are expected to play more significant role in total leakage power of future generation of processors [3,4]. In a Complementary Metal Oxide Semiconductor (CMOS) memory circuit half of the transistors will be off at any given time, and thus contribute towards memory leakage current.

Figure 3 shows how leakage current increases as the technology scales down from 180 to 32 nm. Scaling and power dissipation trends in future technologies will cause sub-threshold leakage current to become an increasingly large component of total system power dissipation [55]. Newly proposed architectural and circuit optimization techniques to reduce leakage current have been discussed later in this chapter. In the following sections we are going to analyze components of leakage current.

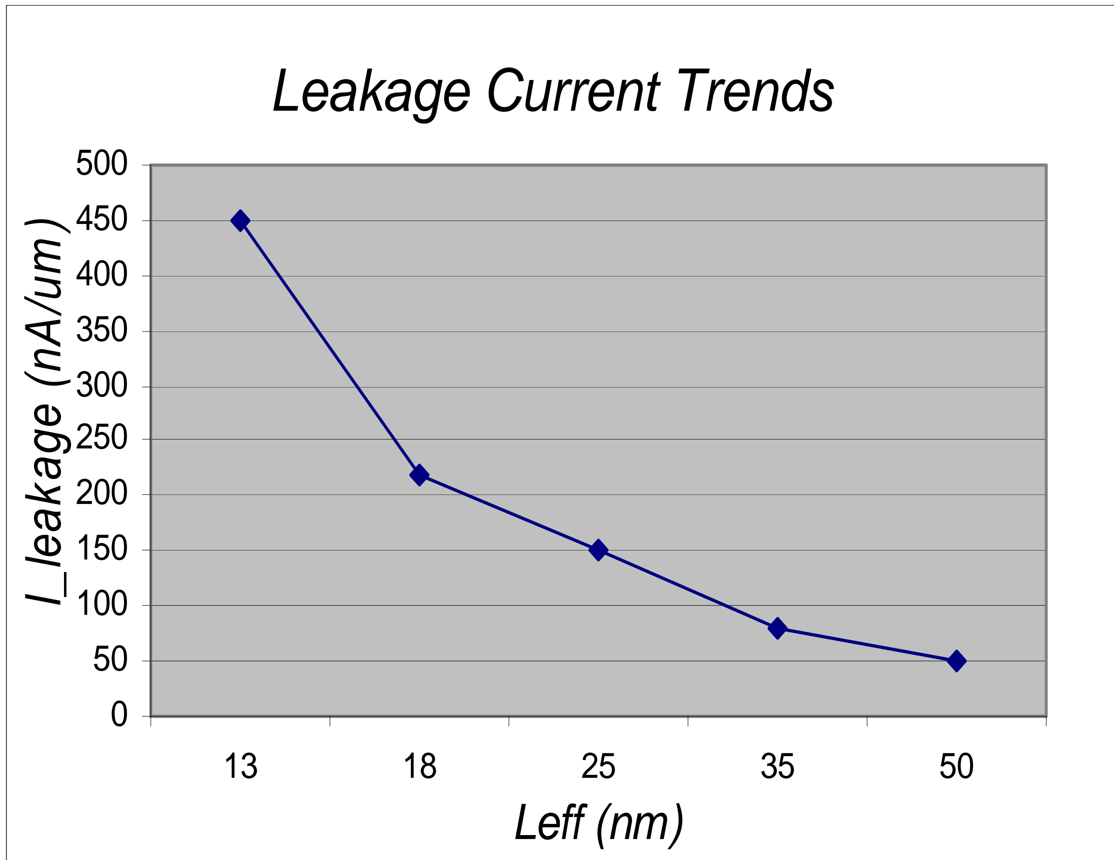


Figure 3. Leakage Current Trends as Technology Scales from 180-to 32nm

3.2 Memory Cell Static Leakage Current

Embedded SRAM suffers form cell leakage current and bit-line leakage current. Five basic components of cell leakage current are: reverse biased, sub- threshold, gate induced punch through, and gate tunneling leakage current. Sub-threshold leakage current is drain to source current of the transistor. When the gate source voltage is less then the threshold voltage (i.e, transistor is operating in weak inversion). In this region the main contributor to leakage current is defined by Berkeley BSIM model as

follows [24]

$$Idsub = Io\left[1 - \exp\left(-\frac{Vds}{Vt}\right)\right]\exp\left(\frac{Vgs - Vth - Voff}{nVt}\right) \tag{1}$$

where, V_{off} is empirically determined model parameter, $Vt = KT/q$ is a physical parameter proportional to temperature, V_{th} is the threshold voltage, and current $I_o = I_{so}'*W/L$ depends on transistor geometry. Leakage current through single transistor when it is off ($V_{gs} = 0$ and $V_{ds} = V_{cc}$) can be reduced either by increasing the threshold voltage or by changing the width of the transistor.

3.3 Leakage Current in Idle Memory

During the inactive state, when world-line is low and bit-lines are pre-charged high, one memory cell stores a high and a low value. In Figure 4, when a *0* is stored at node *A* transistors T1, T5 and T4 will dissipate leakage current, and when a *1* is stored, T2, T3 and T6 will dissipate leakage current. Leakage current in the memory cell can be expressed as:

$$Idcell = Idsub(T1) + Idsub(T5) + Idsub(T4) \tag{2}$$

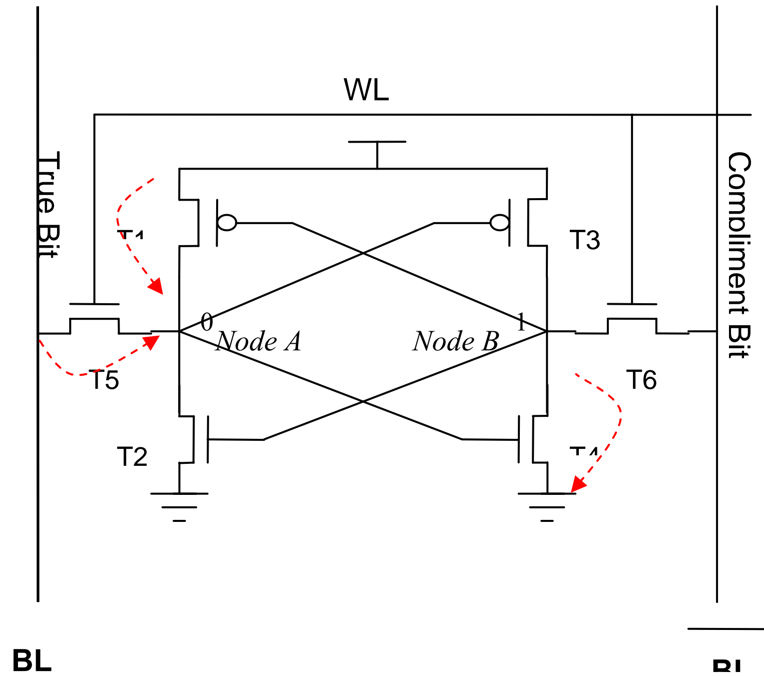


Figure 4. Leakage Current in Single-Port SRAM Cell

For a memory core with N rows and M column this leakage current is represented as:

$$Idcore = N * M\left(Idsub(T1) + Idsub(T5) + Idsub(T4)\right) \tag{3}$$

Figure 5 shows the classic dual-port memory architecture, in which each SRAM cell is accessible by two ports with dedicated word and bit lines to each port. The addition of the word , bit lines and access transistors *T7* and *T8* would not only double the silicon area as an approximation, but also increases the leakage current significantly.

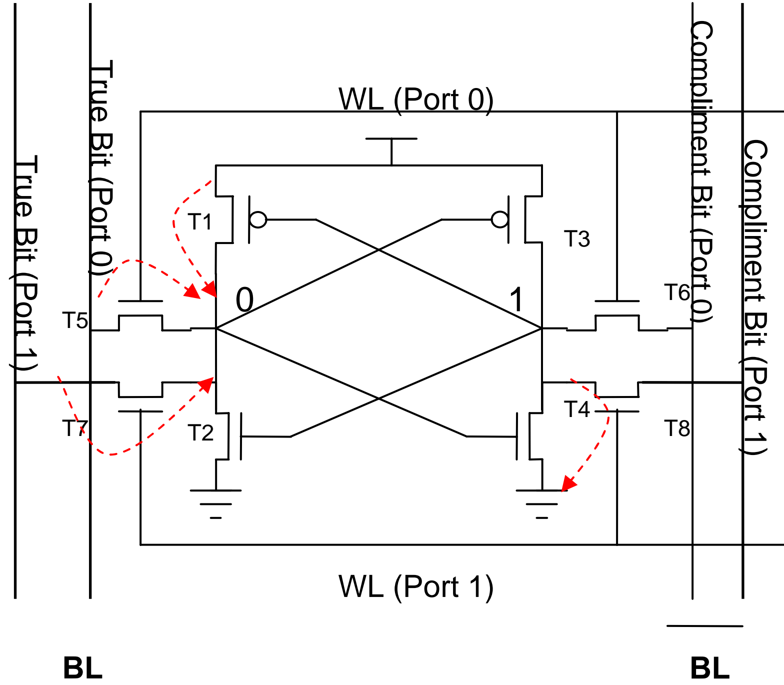


Figure 5. Leakage Current in Dual-Port SRAM Cell

In multi-core technologies, multi-port memory architecture has been applied to allow multiple simultaneous accesses to memory. However, sub-threshold leakage current and large amount of bit line discharge have emerged as an unwanted byproduct.

Leakage current for a Dual-port memory cell can be expressed as:

$$Idcell = Idsub(T1) + Idsub(T5) + Idsub(T4) + Idsub(T7) \tag{4}$$

For a memory core with N rows and M column this leakage current is represented as:

$$Idcore = N * M \begin{pmatrix} Idsub(T1) + Idsub(T5) + Idsub(T4) \\ + Idsub(T7) \end{pmatrix} \tag{5}$$

3.4 Leakage Current during Read and Write Operation

During read operation bit-lines are precharged high and one word-line is activated. Depending on the data stored in memory, one of the bit-lines discharges, which accounts for about 10% of V_{DD} in most high speed memories discharge [25]. For simplicity, let us ignore bit line discharge and assume that both bit-lines are charged to V_{DD} during the Read operations. In Figure 4, for single-port memory, pass transistors of T5 and T6 are turned on. When a *0* is stored at node *A*, only transistors T1 and T4 will dissipate leakage current. Leakage current in the memory cell during read and write operation is expressed as:

$$Idcell = Idsub(T1) + Idsub(T4) \tag{6}$$

In a memory core with *N* rows and *M* columns, *(N-1)* rows will have three transistors and one row will have only two transistors per cell that will dissipate leakage current. Memory array leakage current in read phase can be expressed as:

$$Idcore = (M)\{Idsub(T1) + Idsub(T4)\} + \Big[(N-1) * M\big(Idsub(T1) + Idsub(T5) + Idsub(T4)\big)\Big] \tag{7}$$

In high performance memories bit-lines are precharged high and *Write* operation is accomplished by forcing a 0 at either the bit or compliment bit line. Depending on data one of the bit-line pair is discharged. The leakage in write phase is the same as that for read operation.

Since leakage current in memory core is much larger than that of a column, collectively, we will use Equation (5) to express leakage current in a memory core.

Summary

This chapter explored analytical models for leakage power estimation in SRAM cell. The model considered leakage current in single and Dual-port memory cells. The model takes numbers of rows and column in consideration to estimate leakage current in memory core. We observed that in a single-port memory cell in an inactive state at-least three transistors are contributing towards sub-threshold leakage current. In a dual-port memory cell four transistors per cells contributes toward leakage current. We will compare leakage current dissipation in dual-port memory against leakage current in dynamically configured dual-port memory proposed later in this dissertation. The next chapter presents literature overview of the scope of this dissertation.

Chapter 4 Literature Review

Introduction

In the 90nm generation, the amount of embedded memory in system on-chip (SOC) has already exceeded 50% of the chip area. It is also expected that this percentage will further increase at the 65nm regime and beyond [63]. Static SRAM contains high capacitive buses that are accessed frequently, as the result latency and energy dissipation have become an important design constraint as technology scales down in sub micron regime. In this chapter we will outline various research areas that are relevant to reducing the sub-threshold leakage power in *static random access memory* (SRAM). Furthermore, we will discuss circuit as well as architectural techniques used to reduce leakage current.

4.1 Circuit and Device Approaches to Reduce Leakage Current

Most of the recent research activities in this area are geared towards the reduction of sub-threshold leakage current in on-chip cache. The exponential impact of higher

threshold voltage on leakage power leads to techniques such as *Dynamic V_t SRAM, Reduced-gate SRAM* (RG_SRAM), and *Multi-Threshold CMOS* (MTCMOS) [8, 35, and 38]. Among process and circuit level techniques, *Dual-Threshold Voltage* and *Gated V_{dd}* have been discussed in [8, 15, 34, 35, 36]. We will briefly describe these techniques in this section.

4.1.1 Multi-Threshold CMOS

The *multi-threshold CMOS* (MTCMOS) circuit technique was proposed to satisfy both the requirement of lowering the threshold voltage of transistors and reducing standby sub-threshold leakage current [23]. MTCOMS uses N-channel and P-channel MOSFETS with two different threshold voltages, it also uses *active* and *sleep* mode for efficient power management. High performance is achieved when low threshold voltage transistors are used in the logic circuit. Power dissipation is reduced by using High-V_t transistors (sleep transistors) to gate the power supply to the low-V_t logic block. Whenever high-V_t transistors are turned on, low-V_t logic block is connected to V_{dd} and V_{ss} as shown in Fig 6. MTCMOS technology cannot be applied effectively in the memory design since disconnecting the power supply can destroy the memory data [35].

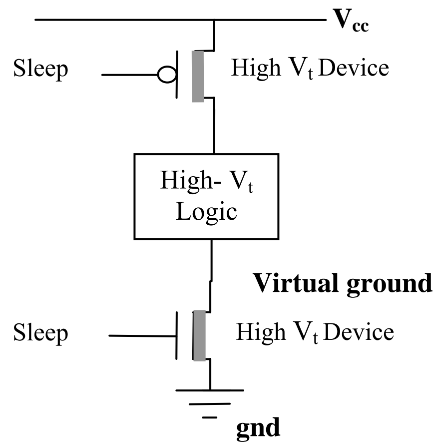


Figure 6. MTCMOS Circuit Structure

4.1.2 Dual-Threshold SRAM

Dual-Threshold voltage CMOS (Dual-V_{TH} CMOS) technique has been reported [15, 33 and 35] to reduce sub-threshold leakage current significantly. Dual threshold reduces leakage current by using low-V_t devices in the critical logical path (where high performance is desired). Since leakage current in SRAM contributes significantly towards total system power dissipation, dual-V_{TH} SRAM design uses high-Vt devices in the memory cell and low-V_{TH} devices in the peripheral circuit. In [42, 47] SRAM cell with different configuration of dual-V_{TH} transistors have been investigated. Fig.7 shows different design choices. High-V_{TH} reduced leakage current but had an adverse effect on the performance of memory. In the worst case scenario, when all transistors in a memory cell are high, V_{TH} cache memory delay can increase by 26% [42, 47].

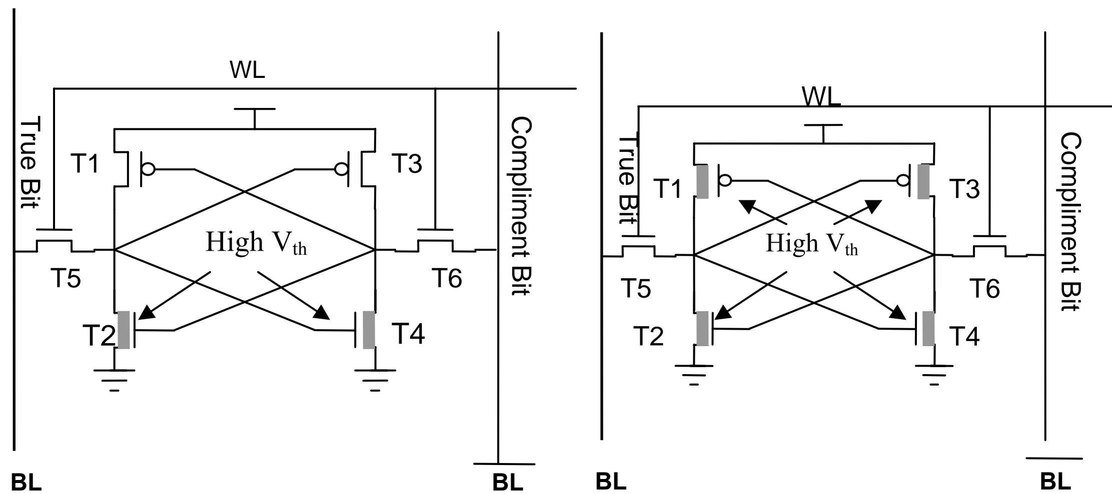


Figure 7. Single-Port SRAM with High-V_{TH} Transistors

4.1.3 Gated-Vdd SRAM

Sub-threshold leakage current and leakage energy dissipation increase exponentially with decreasing threshold voltage. *Gated-Vdd* or *Gated-ground* introduces an extra transistor to gate supply voltage (V_{dd} or the ground path) of SRAM cells as shown in Fig.8. The extra transistor is turned on in the used sections and turned off in the unused sections. To prevent leakage energy dissipation, Gated-V_{dd} uses the stacking effect of self reverse-biasing series-connected transistors. Gated-V_{dd} reduces leakage current significantly. When switched to low leakage mode, it can lose information stored in the memory cell, thus significant performance penalty may occur when higher level cache is accessed due to lost of data.

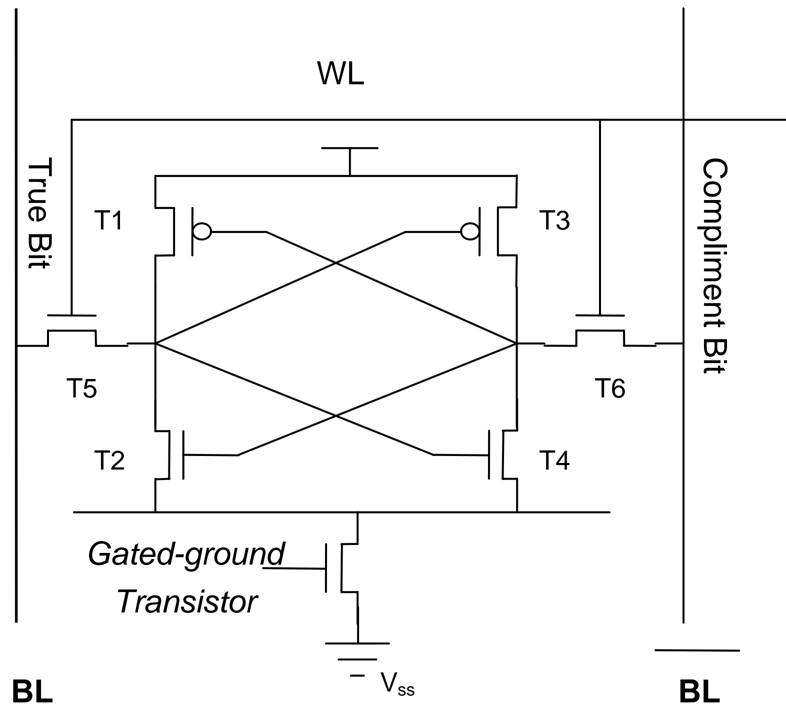


Figure 8. Gated-Ground SRAM Cell

Gated-V_{dd} transistor can be shared among multiple circuit blocks to amortize the overhead [8].

Among other approaches forward and reverse biasing techniques have also been used successfully to reduce the leakage power in SRAM. In these techniques, the threshold voltage of the transistors of each cache line is controlled separately by using forward or reverse biasing of the circuit. Forward biasing reduces the threshold voltages and thus increases the on-currents of the devices. Reverse bias voltages raises the threshold voltages by reducing the sub-threshold currents and saving power in the standby mode [46]. Dynamic V_t SRAM [34] reduces leakage current in cache memories by switching cache lines to high V_t, if the access has a small probability.

4.2 Architectural Techniques to Reduce Leakage Power

In recent years several techniques have been implemented to design energy efficient memories. Most approaches to reducing leakage power combine architectural techniques with circuit techniques [27 – 32]. Pre-charging as well as keeping high capacitive bit-lines pre-charged high, causes significant power dissipation and contributes heavily to total system power dissipations. Considerable leakage current reduction is achieved by putting the cache in low power or drowsy mode [27]. Sub-banks and bit line segmentation has also proven to be very effective in designing low power SRAMS.

Among other architectural and organizational techniques, bit line segmentation has proven to be very efficient in reducing the capacitance and power in large on-chip cache [28, 29]. Hierarchical divided bit line approach [28] divides bit line into sub-bit-lines resulting in that reduces bit line capacitance, which further reduces active power and access time. Yang *et. al.* [29] explored this architecture further and proposed Hierarchical Bit Lines with Local Sense-amplifier (HBLSA), that further reduces bit line capacitance and write voltage swing of bit line. Adding parallel bit lines in above architecture however increases memory area. Selective Cache Ways [30,44] approach also reduces power in memory design by divide large cache in sub-arrays. This approach saves power by enabling only a small portion of L2 cache at a time. With the sub-array partitioning in place on demand selective cache provides an ability to change the size of cache depending upon the application. The virtual cache memory presented in this approach is accomplished by introducing more ways within

sub-arrays using a local word line and repeater switches, cache size can be changed by combining sub arrays to make a larger cache.

One major source of energy dissipation is charging/discharging the whole bit-line. Rao et.al [31] used segmenters on bit-line to prevent waste of energy. Low power dissipation is achieved by allowing only part of bit lines to be pre-charged. In their work they studied power consumption by dividing memory into two four and eight segments, though this approach has added delays to critical path, yet it has reduced the length of bit lines for near cache rows there by reduced the capacitance and power efficiency of memory. Mostly proposed architectures have hardware overhead, increased area results in larger leakage energy.

The cache memory configurations employed in the above mentioned approaches use fixed bank size and duplicated word and bit lines (without providing dual- and multi-port accesses), hence, incur either moderate performance degradation or large area overhead.

Summary

Over the last decade there are many proposed circuit and architectural techniques aimed primarily at addressing leakage power in SRAM. New designs are continuously being developed to low power, reduce area and design high performance memories. This chapter presents overview of several circuit-level and architectural techniques for reducing the leakage power in deep submicron cache. Survey of design

and process techniques like Gated Vdd, RG_SRAM and MTCMOS for suppressing the gate leakage current by changing the threshold voltage of devices is presented.

Brief overview of organizational and architectural techniques like selective precharging, putting cache in drowsy mode, sub-banking and bit line isolation is also presented in this chapter. These architectural techniques reduce leakage current by activating only a part of cache or by changing the size of memory. In the following chapter we will look at architecture and organization of conventional cache in detail.

Chapter 5 SRAM Organization and Architecture

Introduction

Memory designs are continuously evolving when encountering new problems due to poor wire scaling and leakage current in large on-chip cache [45]. As Very-large-scale integration (VLSI) technology matured, research interests, performance and energy tradeoff points shifted. Since new technologies introduced new challenges, researchers responded with designs and circuits to compensate for such limitations. In this chapter we are going to present details of conventional cache architecture and evaluate some recently proposed designs to encounter limitation introduced due to device scaling.

5.1 Terminology

Since existing literature of cache has inconsistent terminologies, beginning of this chapter lists and defines the terminologies and expressions used in the following chapters.

Cache:

Cache is a small fast memory that holds the contents of s fraction of the overall memory.

Block:

The cache memory is organized in *blocks.* When data is transferred between caches or between cache and main memory, the entire block containing the desired address is transferred. In other words, this is the smallest unit of information that may be present in the cache.

Instruction and Data Cache:

Instruction Cache stores CPU instructions and *data cache* stores data for the running application.

Set:

Set is a collection of blocks that holds data.

Associativity:

Associativity is the number of blocks in a set. If the number of blocks is one, then the cache is called direct mapped cache.

Tag and Index:

In set associative, cache index is used to select the set and tags are used to select the block.

Tag RAM:

Tag RAM stores address bits. Tag RAM size depends upon the size of the memory. The data in the tag RAM determine whether a cache lookup results in a *hit* or a *miss.*

Data RAM:

The actual data is stored in a different part of the cache, called the data store. Tag RAM is of varying size, depending upon the size of memory.

Cache Hit and Misses:

If the location specified by the address is stored in the cache, then a cache *hit occurs,* otherwise a *miss* results and the request is forwarded to the next memory in the hierarchy.

5.2 SRAM Architecture

A single bit of memory cell, storing only one bit of data can only be accessed by using both row and column addresses. For easier addressing, memory arrays are organized as a logical matrix of 2^n rows by 2^m columns. *Thus a* 64KB memory cell would have 256 row and columns.

The row decoder activates one of the 2^n word lines, which connects the memory cells of that row to their respective bit lines. The column decoder sets a pair of column switches that connects one of 2^m bit line columns to the peripheral circuits. A simple block diagram of SRAM is shown in Figure 9. This architecture works well for the memory size of 64K to 256K [62]. However, as memory size increases capacitance and resistance of word *(rows)* and bit lines *(columns)* plays a significant role, thereby causing memories to become extremely slow.

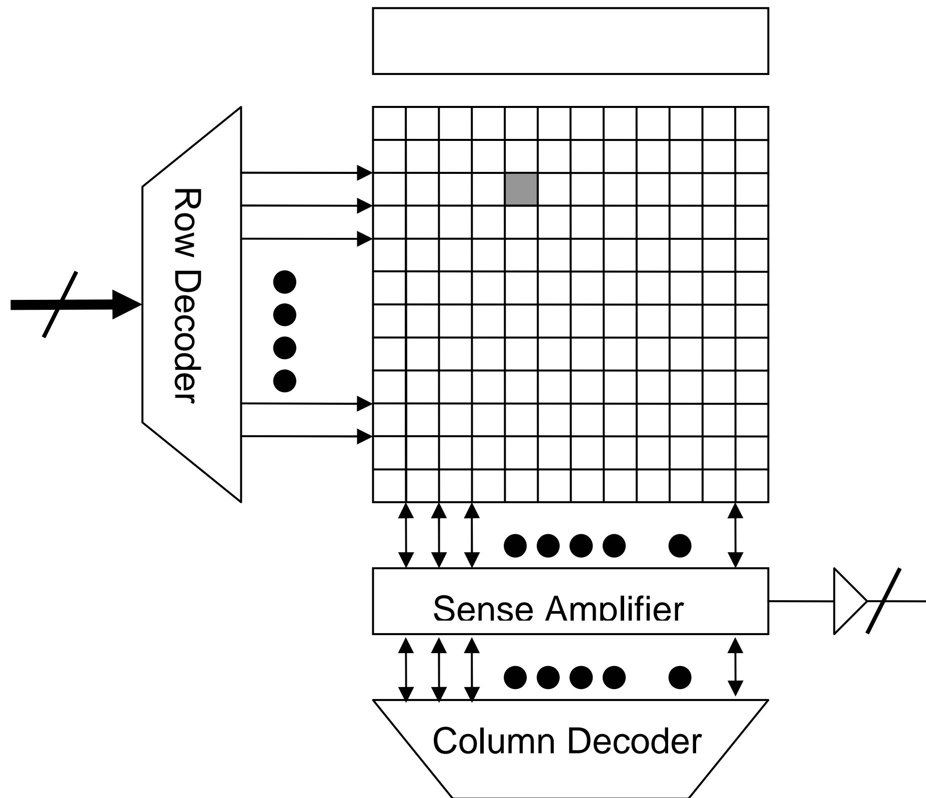


Figure 9. Block Diagram of SRAM

In response to increasing performance gap between high performance micro-processors and memory speed, the amount of chip area devoted for memory is continuously growing. In order to avoid speed degradation due to length resistance and capacitance of long wires, large memories are divided into small sub-arrays. Figure 10 shows an organization of simple data cache. An extra page select line can be used to select one page from several pages or one block from several blocks.

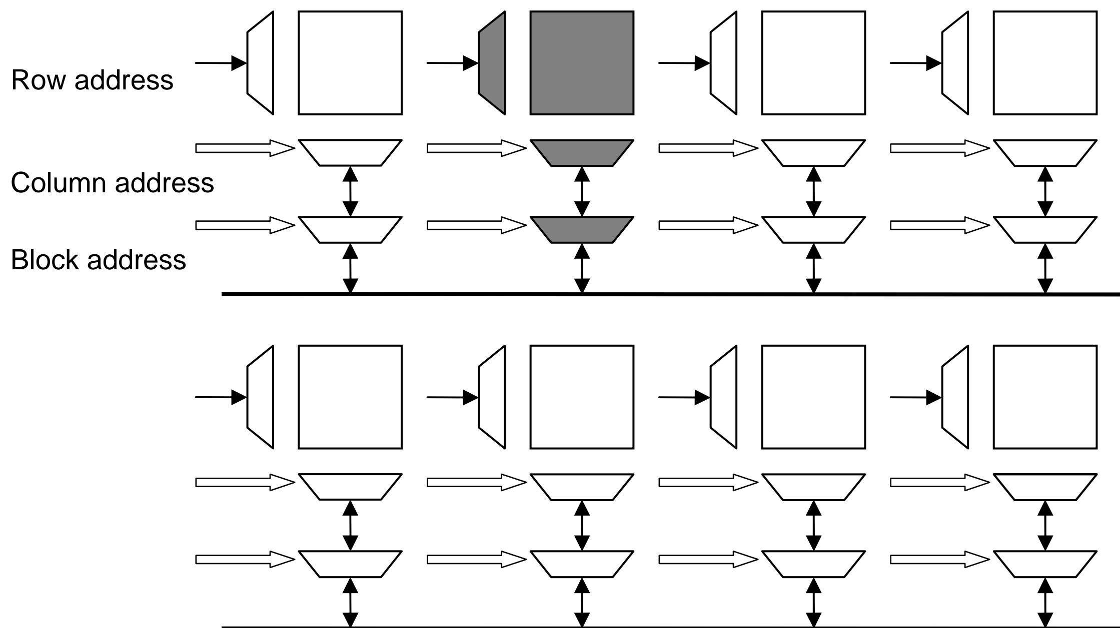


Figure 10. Hierarchical Memory Architecture

5.3 Cache Addressing

On-chip cache memory is divided into two independent banks, one for data called data bank and the other for address tags and status bits referred to as Tag Bank. Data bank stores data needed for program execution and tag is used to determine if cache data is indeed the data being referenced. Figure 11 shows a block diagram of a simple cache presented by [39]. A cache line shown below contains contents of different memory locations along with their addresses.

Memory address is also divided into three parts; tag, index and byte offset. Index bits are used to determine the cache line. A processor uses these bits to address higher order *tag bits* which are compared with the higher order address bits from the cache.

If it matches a hit occurs, otherwise the cache accesses the memory in the hierarchical structure.

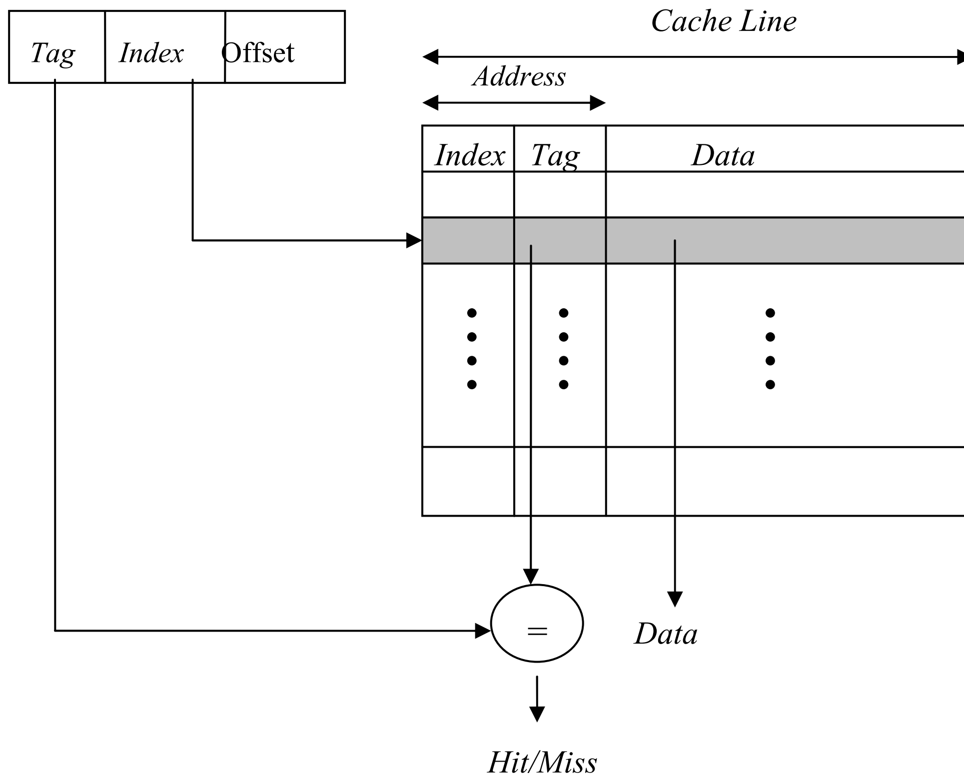


Figure 11. Block Diagram of Cache

5.4 Cache Sizing and Associativity

Cache is supported in direct mapped, set-associative or fully associative organizations, depending upon whether the block can be placed in only one location, a set of locations or anywhere in a cache. As mentioned earlier cache is a major source of power dissipation in embedded microprocessors. Associativity greatly affects the system's energy, direct mapped cache uses less power than set associative cache since only one tag and data array are accessed in one cycle. In this section we are going to

look at different cache organizations.

Direct mapped cache is the simplest cache organization, which requires only one comparator for matching the address requested by the processor. Since the data from the main memory can only be mapped to one location in the cache, *direct mapped cache* has a low hit rate. *Set associative cache* is the most commonly used cache in modern microprocessors. *N-way set associative cache* allows the block to be placed anywhere in a set of 'n' ways. Increase in associativity increases the hit rate by increasing the number of blocks and also comparators. A delay is caused because of comparison and selection of sets and blocks. *Fully associative cache*, also known as CAM (Content Address Memory) allows any main memory location to be mapped to the cache. In order to find a block in a fully associative cache, all memory location must be searched. To make this search practical, a fully associative cache checks all the tag addresses against the memory block address in parallel. Since comparator is associated to each cache line this architecture significantly increases the hardware cost [39].

Since set associative cache is the most frequently used cache organization in modern high performance processors and multi core technologies, we will analyze it in detail and use it as an example to explain our newly proposed architecture. The Figure 12 below shows 8 way set associative cache. In the Figure 12, the structural configuration of a 264KB conventional single-port 8-way cache memory is illustrated. Eight way set associative cache divides memory into 8 identical banks, that have the

same set location. Tag comparisons from all banks are done simultaneously, if a hit occurs, the data is sent to output.

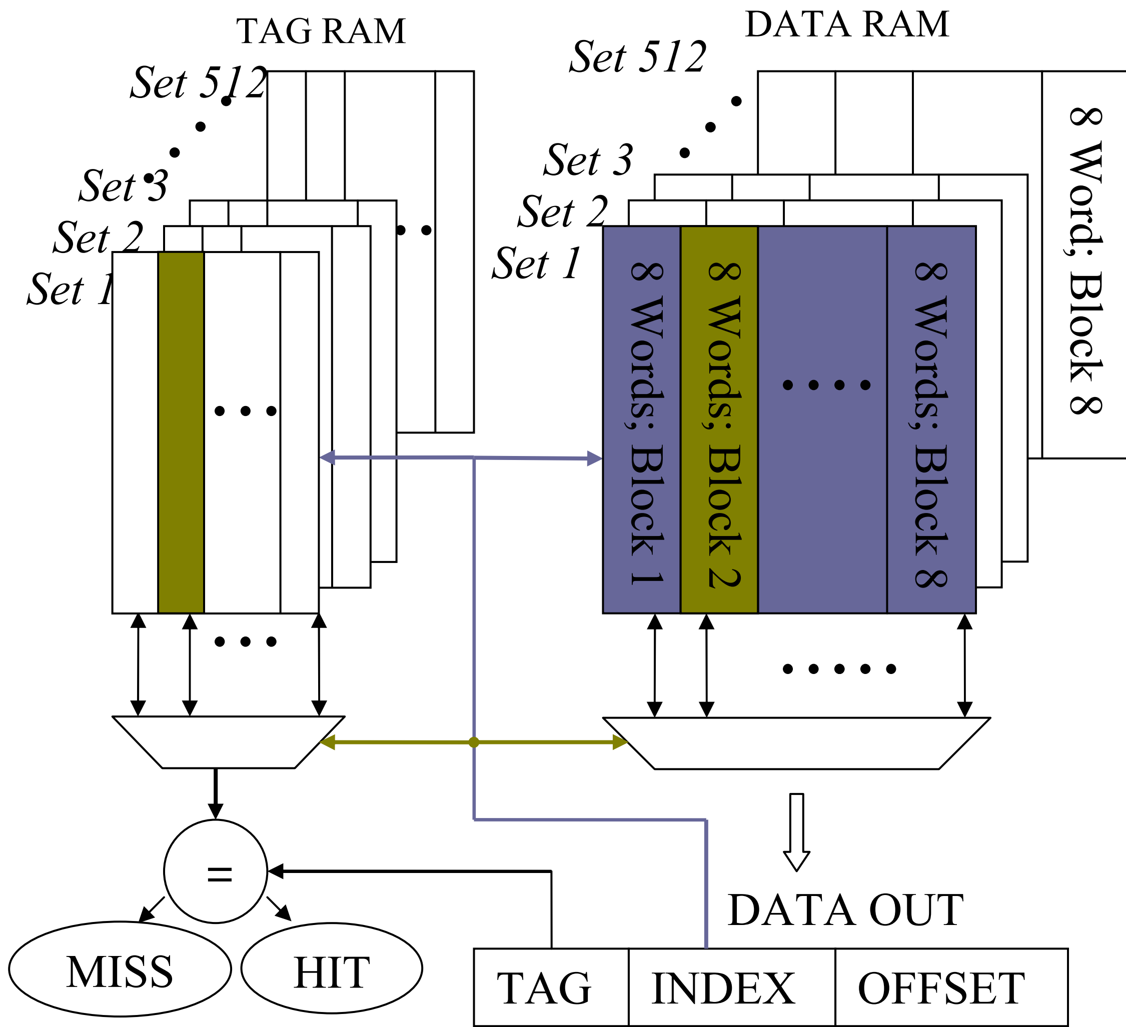


Figure 12. A Cache Subsystem Organizations

5.5 *SRAM Partitioning*

Delay and power of SRAM have been reduced over the years via innovations in the array organization and circuit design. Traditionally, to achieve high performance a

large SRAM arrays is divided into sub-blocks, this approach not only reduces both the bit-line and word-line capacitances, but also reduce power by enabling only a small portion (single way) of cache [30]. High performances SRAMs can have up to 16 sub-arrays, while low power memory typically has only one [30]. Each sub-array can be thought of as independent RAMs, except some sharing of parts of the decoder.

Set associative cache has been implemented in most of the modern microprocessors [28]. In an *m* way set associative cache, there are m tags and m ways, each cache access results in *m* tag comparison and the discharge of *m* data words. In case of a hit data from one of the discharged word-line is driven into CPU. Figure 13 shows a two way set associative cache that requires two comparators and 2-to-1 multiplexor. The comparator determines which element of the selected set matches the tag. The out put of the comparator is used to select the data from the multiplexer.

5.6 High-Performance Memories Timing and Delay Analysis

Bit-line pre-charging and discharging is one of the major sources of power dissipation in a conventional cache. To achieve low power dissipation, cache memory is divided into banks and sub-banks. Modeled after CACTI [23, 30], delay components in conventional memories are as follows:

- Sub-array address decoding incurs delay.
- Bit line pre-charge delay is introduced when all bit line in a bank are precharged high before any memory operation is performed.

- Word-line delay is due to long word-lines selection and long word-line going high.
- Bit line delay is introduced after word line goes high and pulls down one of the bit line, that determines the value stored in the cell (bit line delay).

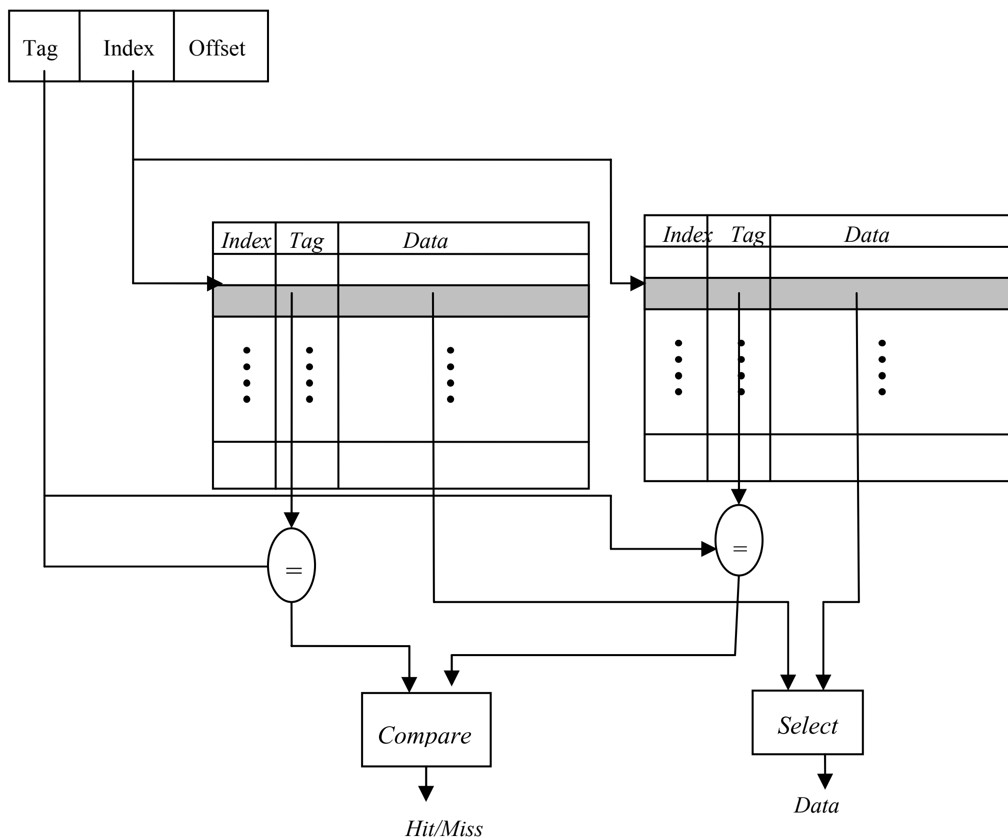


Figure 13. Conventional Single-Port 2-Way Set-Associative Cache

Figure 14 shows the timing diagram of a conventional high performance memory where bit line pre-charge is done first. Memory access time spans from the time necessary to select the SRAM cell to the time until a valid date is received after discharging the bit lines. After a read and write operation, the voltage on the bit-lines

must be equalized by pre-charging to the processor's supply voltage. To minimize delays, memory pre-charging is overlapped with address decoding and output drive [62].

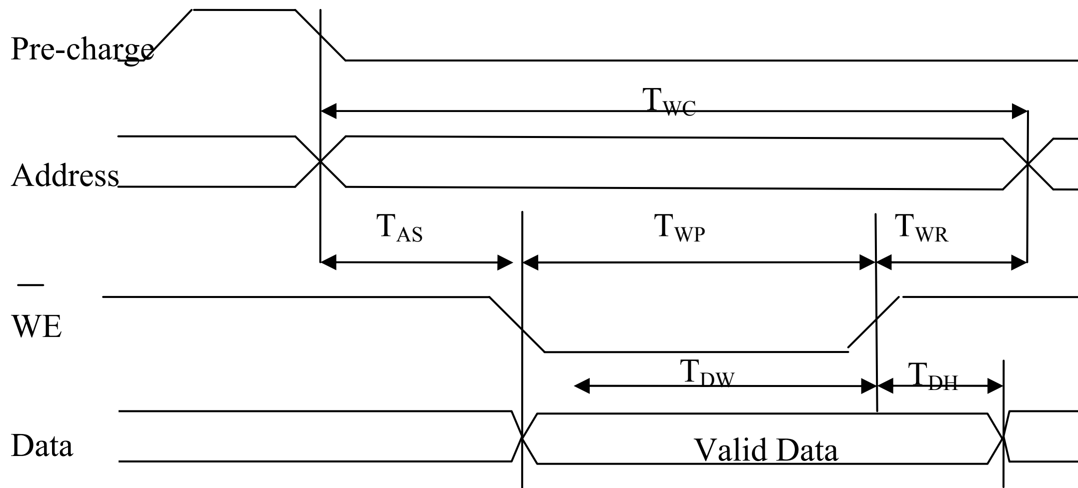


Figure 14. SRAM Memory Operation

T_{WC}: Write cycle time

T_{AW}: Address valid until rising write edge.

T_{AS}: Address setup time

T_{WP}: Write pulse width

T_{WR}: Write recovery time

T_{DW}: Data valid to write end

T_{DH}: Data hold time

Bitline pre-charge is a key source of energy dissipation in high performance memories [56]. In these memories all cache sub-arrays are kept pre-charge irrespective of where data is being accessed from. Memory access time consists of the following components.

5.7 Performance and Delay Analysis of Cache Using Sub-Banking

Selective precharging, i.e. pre-charging only the sub array that contains data, is frequently used to reduce power dissipation in large on-chip memories. Bit lines pre-charge devices are only turned on when a memory operation is needed to be performed. In order to save bit line energy further, only sub-arrays that contain the desired data are precharged. Part of the memory address is decoded to identify the sub-array. The delay introduced due to partial address decoding is identified as (T_{BA}). In these memories, first the address is provided then a bank is selected, followed by pre-charging and finally the necessary operation is performed. When there is no cache access all bit lines are isolated from the supply voltage.

Address setup time (T_{AS}) is

$$(T_{AS}) \geq (T_{BA}) + (T_p)$$

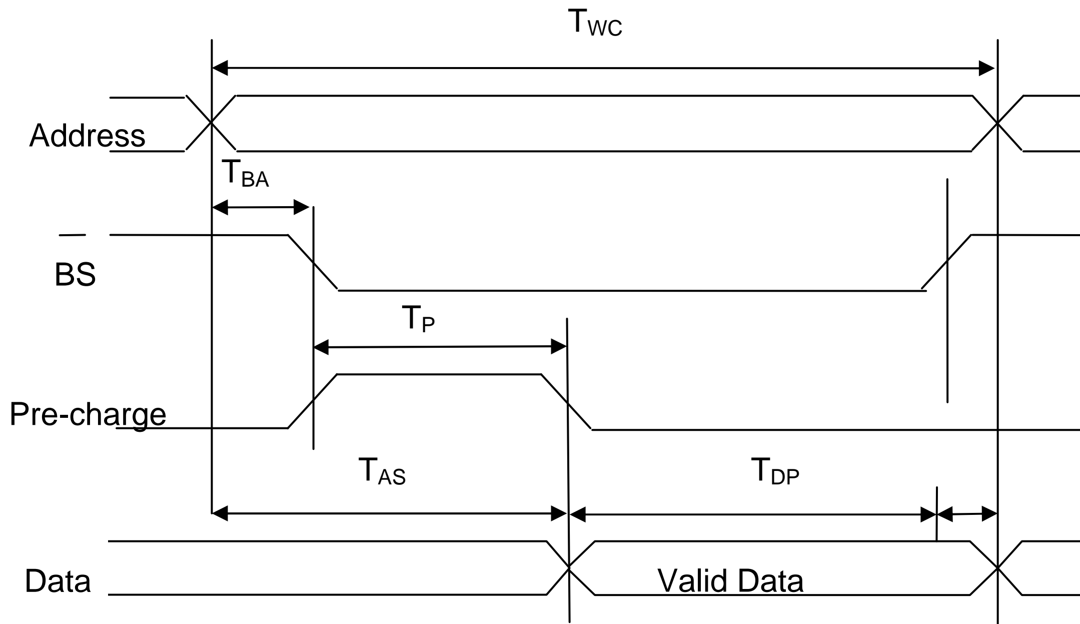


Figure 15. Memory Operation of SRAM With Sub-Banking

T_{WC}: Write cycle time

$T_{BA:}$ Bank address setup time

$T_{AS:}$ Address setup time

$T_{DW:}$ Data valid to write End

$T_{DP:}$ Valid data hold time

Summary

This chapter discussed detail implementation of a conventional SRAM array. Various SRAM architectures have previously been proposed to improve efficiency of cache. Cache sizing, addressing and associativity are discussed in detail. Taking set associative cache as an example, we presented timing analysis of high performance memory with and without selective precharging to reduce power dissipation. Delay components and timing analysis of conventional SRAM operation are also presented in detail. Various parameters and constraints of set associative cache in detail were explored as they are related to propose dynamically configured memory presented in the next chapter.

Chapter 6 Dynamically Configured Memory

Introduction

Multi-processors on single chip (CMP) and Simultaneous Multi-threading (SMT) have pushed memory performance to its maximum limit. Current trends lead toward using hardwired multi-port cache architecture, which employs dedicated word and bit line for each port and thus increases area overhead and power dissipation. In this chapter, we are presenting a newly developed dual-port memory architecture, which employs a technique that dynamically partitions a memory into two virtually independent sections. This technique allows dual-port accesses without duplicating the word and bit lines for the second port, hence, maximizing silicon area utilization.

6.1 Dynamically Configured Memory Architecture

Our investigation shows that a hardwired dual-port memory can be implemented without duplicating the word and bit lines. Newly proposed Dynamic Memory Partition (DMP) technique uses isolation nodes to partition a cache memory block into two virtually independent sections based on real-time access addresses of

multiple ports. The isolation nodes, controlled by the isolation control lines are interconnect switches which are placed on the bit lines between the access transistors on neighboring word lines.

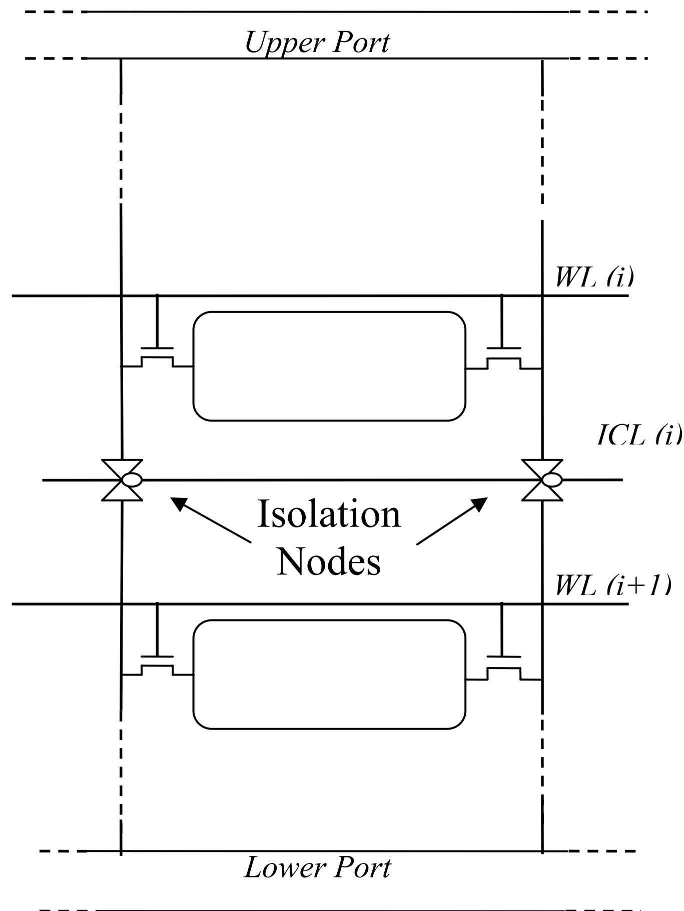


Figure 16. Placement of Isolation Nodes and Control Lines

Figure 16 shows placement of Isolation Control line (ICL) and isolation node on each of the bit lines to divide an SRAM block into the upper and lower sections, which are to be accessed by the upper and lower ports, respectively. When all isolation nodes are ON, both the upper and lower ports share the bit lines from each

end to access the same memory cells. With this new dual-port configuration, identification and setting of the isolation control line for dynamic partitioning (DP) must be carried out before accessing selected memory cells by dual-port memory operations. Figure 17 presents a delay analysis using a small circuit block to compare the addresses and calculate ICL delay. DMP operations are carried out in parallel with address setup and bit line pre-charge phases. There is no overall operation delay due to DMP.

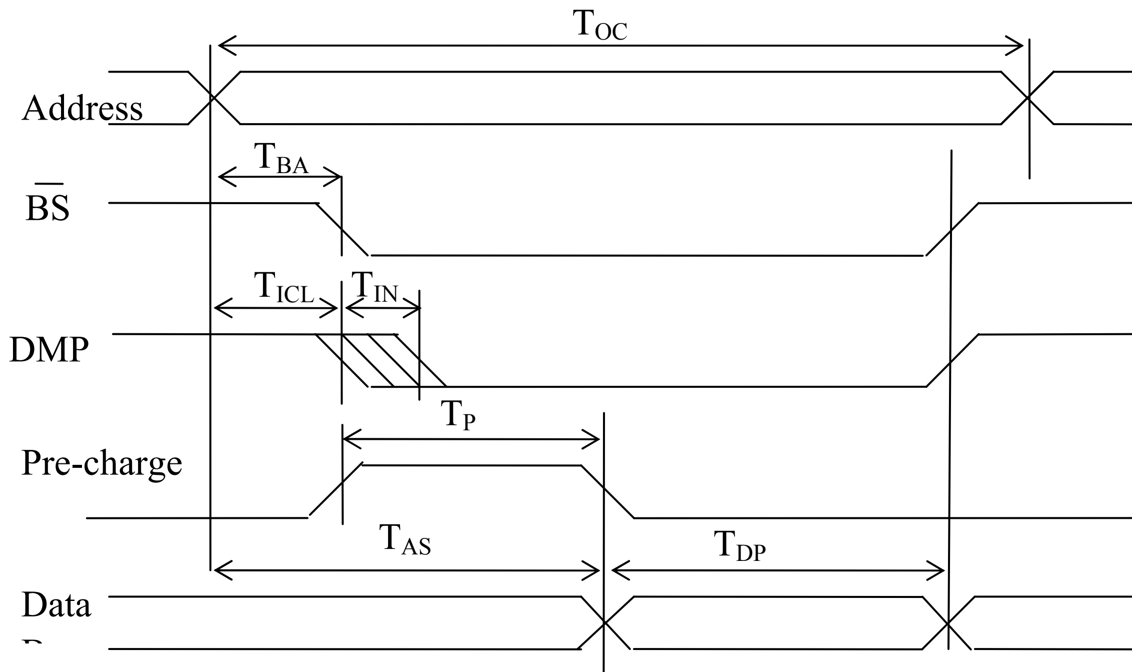


Figure 17. Memory Operation Timing with DMP

T_{OC}: Operation cycle time

$T_{AS:}$ Addr setup time

$T_{BA:}$ Bank address setup time

T_{DP}: Valid data

$T_{ICL:}$ ICL identification time

$T_{P:}$ Pre-charge time

In conventional high speed memories where all memory banks are pre-charged before any operation is performed, dynamic configuration of memory introduces delay. The only DP delay overhead is the setting of the isolation nodes of the selected isolation line. A detailed timing analysis of this will be presented later in this chapter. The use of isolation nodes would appear to increase the silicon area. In reality, this increase is insignificant, since silicon area used by memory is dominated by word and bit lines. Compared with the hardwired dual-port SRAM cell as shown in Figure 2, DMP can provide dual-port accesses without the need of the second pair of bit lines and effectively reduce leakage current, bit line latency and silicon area. Figure 18 shows a block diagram of newly proposed dual-port memory architecture.

6.2 Isolation Node Placement

The key challenge while designing dynamically configured multi-port memory is to design a mechanism that not only partitions the memory without incurring delay but also reduces power dissipation and area overhead. Hierarchical and segmented bit line approaches to reduce leakage current have been discussed earlier in chapter 3. We extended this idea of bit line segmentation earlier presented by Rao *et. al*. [31] and used it to provide dual-port access. Rao *et. al*. [31] proposed bit line segmentation by placing transmission gate on bit line. Their approach showed that, bit lines segmented

by 8 isolation nodes pose no significant performance degradation. With DMP, the placement of isolation nodes is of strategic importance. Placing isolation nodes between adjacent word lines provides the most flexibility for DMP, but, would be an overkill, if the applications at hand do not need to utilize such fine-grain DMP capability. In principle, isolation nodes are placed for every n word lines, where n is determined based on the statistical patterns of access addresses of targeted applications.

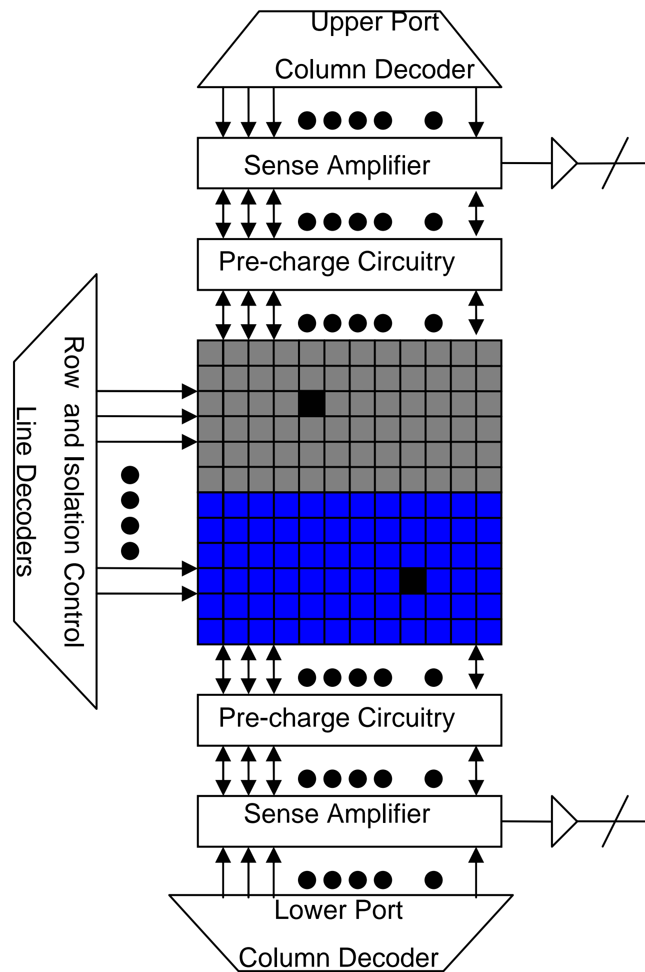


Figure 18. Block Diagram of Dual-Port Memory with Dynamic Partitioning

Figure 19 depicts the placement of ICLs and isolation nodes on a bit line pair. The bit lines access cells on 512 word lines, which are divided into 8 groups, each containing 8 sub-groups. Each group contains a dedicated local sense amplifier block and a pre-charge block. An ICL and two isolation nodes are placed between sub-groups, each containing cells on 8 word lines. When an ICL turns its isolation nodes OFF, the upper and lower ports can access desired cells from different sub-groups.

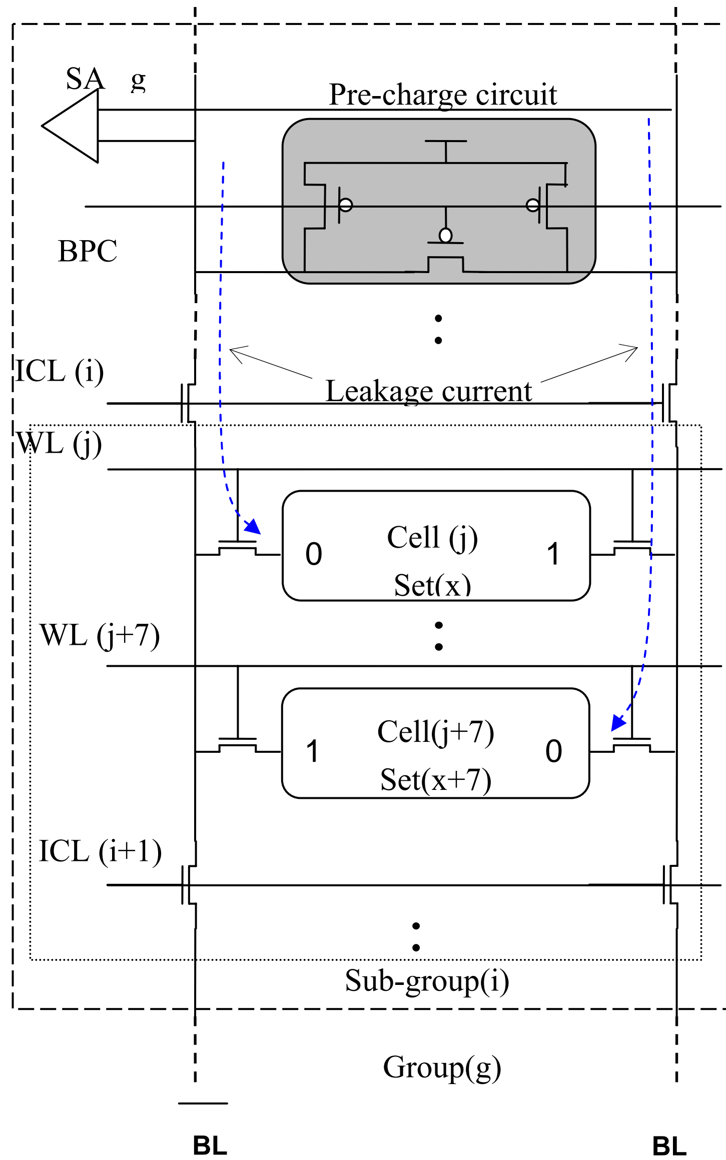


Figure 19. ICL and Isolation Nodes Placement

Each group's local sense amplifiers ensure that the group's bit line section accesses cells on 64 word lines separated by 8 isolation nodes which do not impose significant latency [31].

Figure 20 shows the horizontal view of a set-associative cache subsystem integrated with DMP for multi-port accesses. Compared with conventional memories, that allow only one set to be connected to the BLP during memory operation, placement of isolation nodes allows two sets to access bit line pairs. During memory operations, tags of all blocks in selected sets are compared with tags in the incoming addresses to determine if it is a hit.

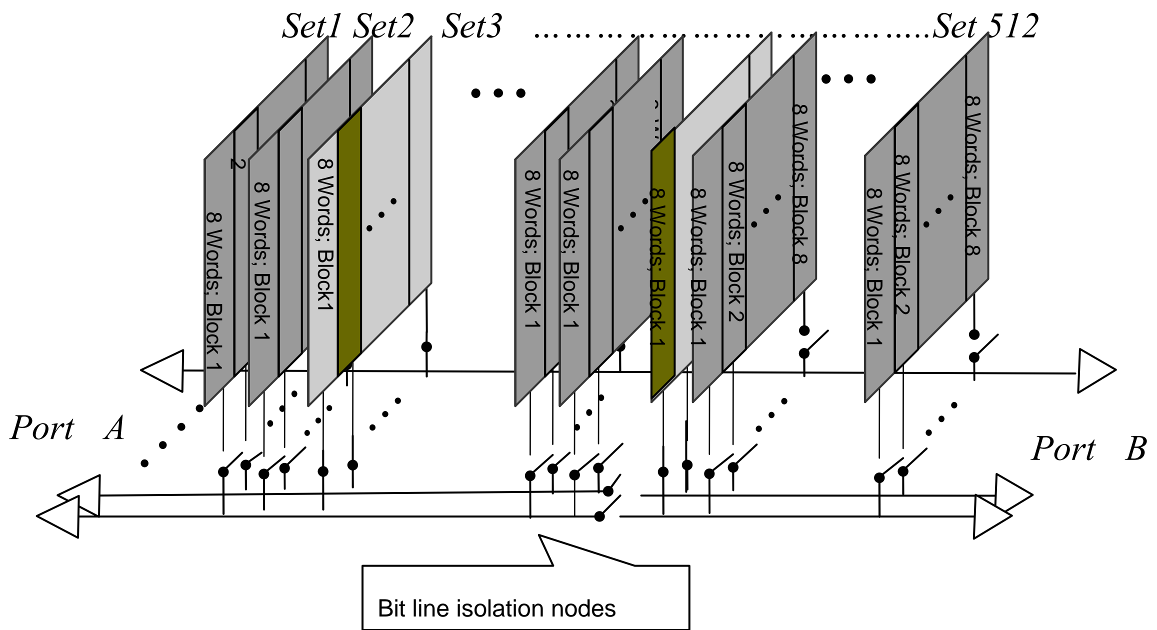


Figure 20. Structural Configuration of Dual-Port Cache Architecture

Thus a large cache memory can be divided into virtually independent sections. Figure 21 illustrates the structural configuration converting a conventional single-port 8-way cache memory into a dual-port cache with DMP without adding the bit lines for the second port.

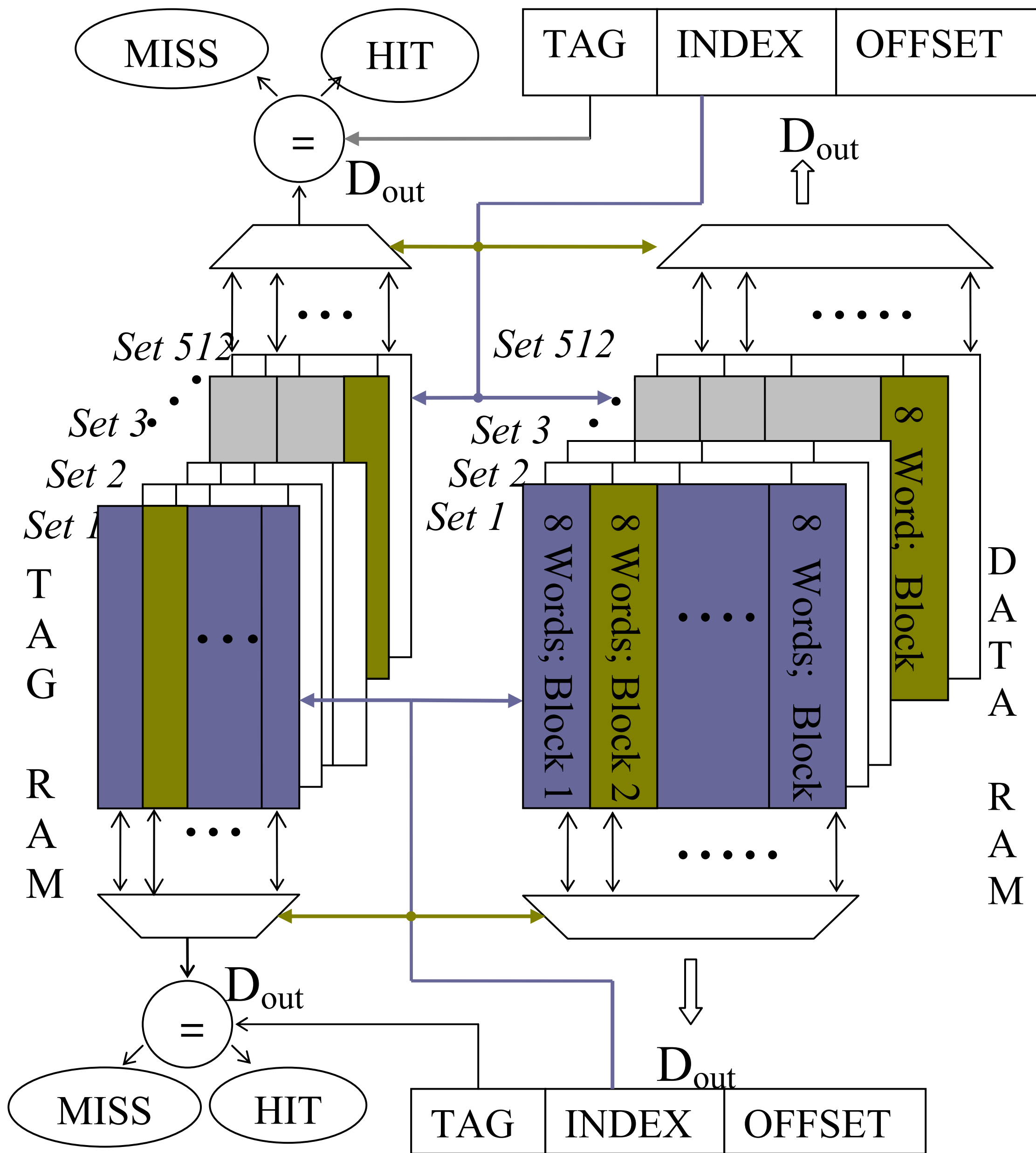


Figure 21. A Cache Subsystem with DMP

Accesses to cache memory via multiple ports may encounter potential conflicts known as coherence, that happen if access to the same cell is tried at the same time.

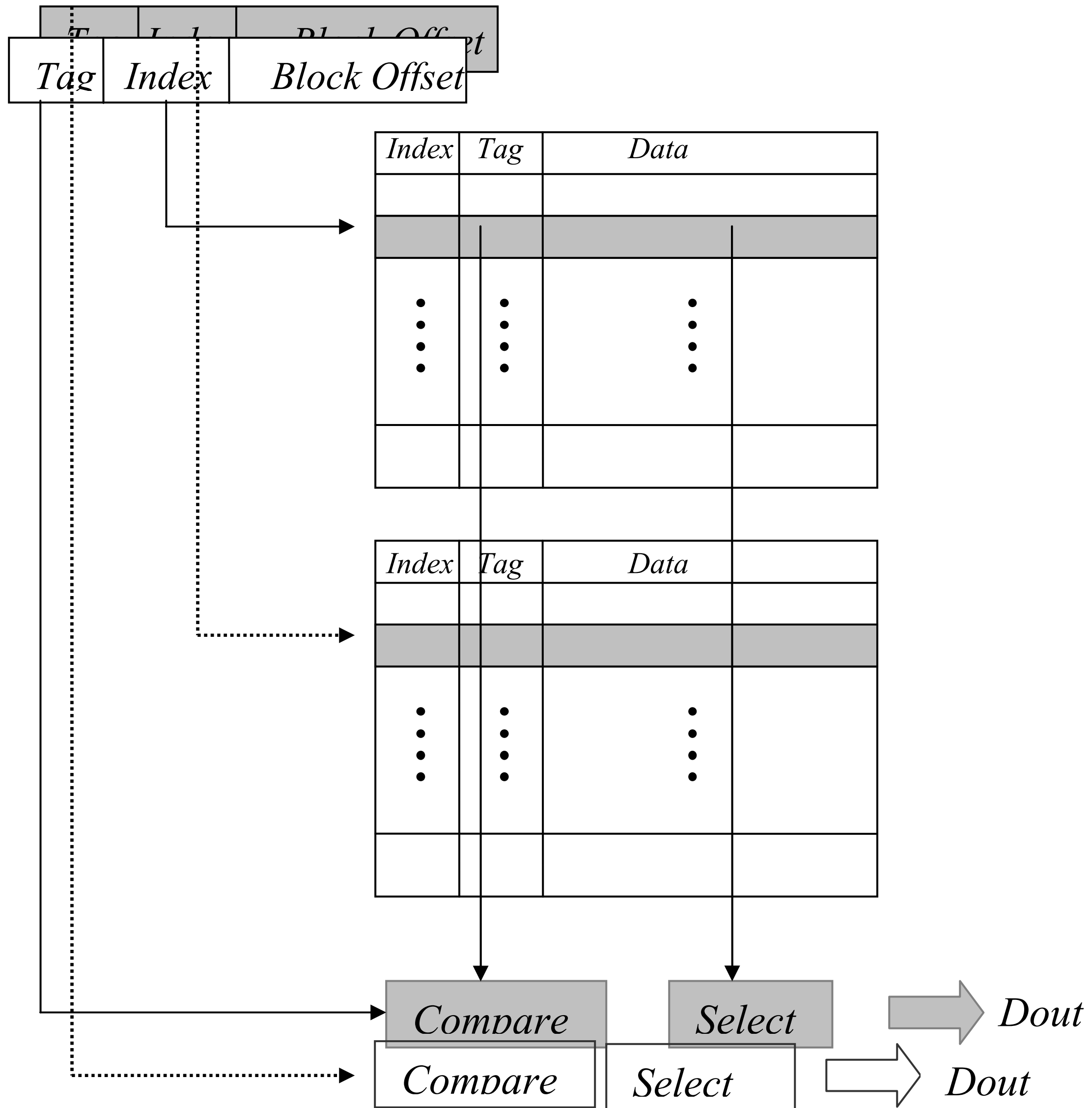
For example, a *READ* and *WRITE* operations via Ports X and Y to the same cells, may take place simultaneously. When this occurs, existing (hardwired) multi-port memory access conflict resolutions may require the READ-WRITE-READ sequence to be executed to ensure data integrity. With DMP, all existing multi-port memory access conflict resolutions still apply and no changes to these protocols are necessary.

6.3 Dynamically Configured Cache Addressing

Reconfigurable cache allows on-chip cache to be dynamically partitioned and reused for other processing activities. One challenge in designing reconfigurable cache is to dividing the memory in two virtual banks and efficiently addressing these banks. Figure 21 presents a block diagram for the organization of dynamically configured cache. The dynamically configured cache has the advantage of being simple and similar to the current cache organization and the mechanism of addressing does not need to be changed. Addressing scheme does not have to change since the functionality of the index, tag and offset bits remains the same. Critical issue of policy, for when to reconfigure is presented in the next section.

6.4 Dynamic Memory Partitioning Algorithm

Optimal cache partitioning not only provides dual-port access but also reduces delay and minimizes power dissipation. Efficiency of dynamically partitioned algorithm is determined by three factors, delay, and power dissipation. Two generic DMP algorithms are developed:



6.4.1 Fine-Grain DMP-1

Algorithm DMP-1 provides the optimal partitioning algorithm that minimizes bit line latency and power dissipation, with which two ICLs are turned off during partitioning: ILC(j) is turned off for the upper port to access WL(j), while ILC(i-1) is turned off for the lower port to access WL(i), where j < i. This approach ensures shortest active bit

lines for both the upper and lower ports, hence minimizing latency and power dissipation.

DMP-1 is pseudo-coded as follows:

```
addr (A) <1: n>; addr (B) <1: n>;
where addr (A) = i  > addr (B) = j;
if i = j +1 return ICL (j);
else return ICL (j) and ICL (i-1);
```

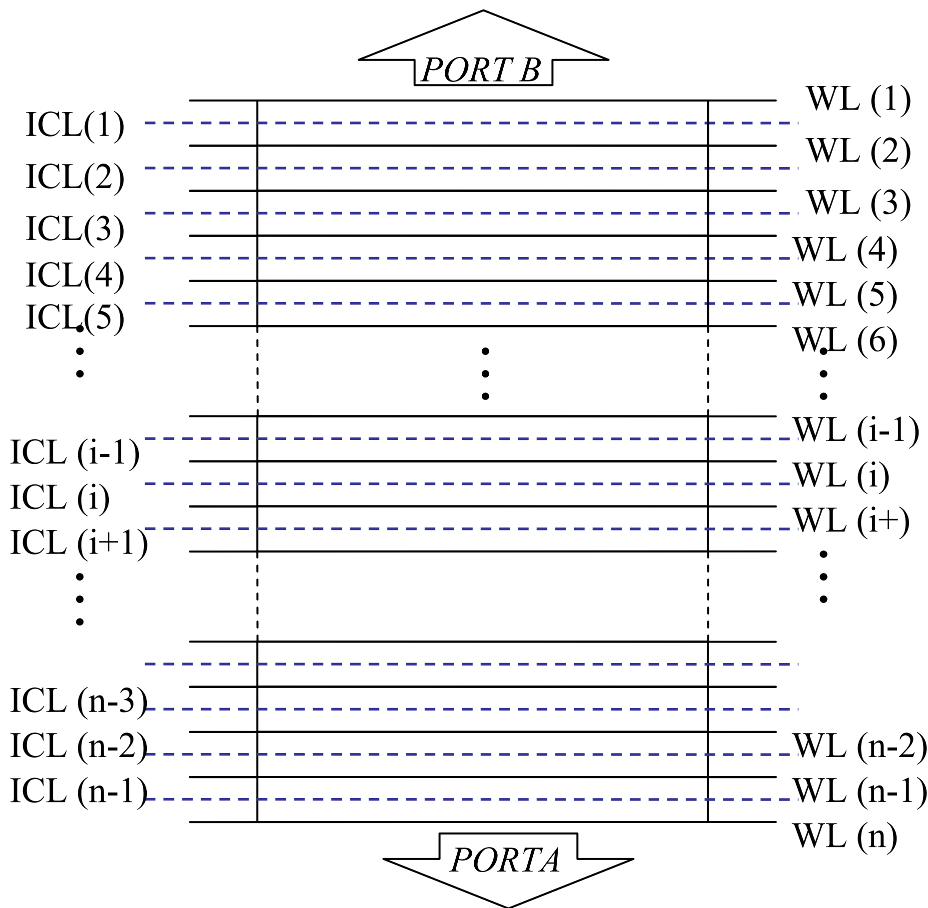


Figure 23. Generic DMP Model

Optimizing cache to minimize delay and power dissipation requires repartitioning after every memory reference. Figure 23 and 24 show generic DMP models before

and after isolation nodes are setup. Since world line 5 and word line (n-2) are accessed by the accessed by two different processing elements the adjacent isolation nodes *ICL(5)* and *ICL (n-3)* were activated. Size of bit lines are reduced and significant energy saving was archived by pre-charging and discharging only a fraction of the bit-line.

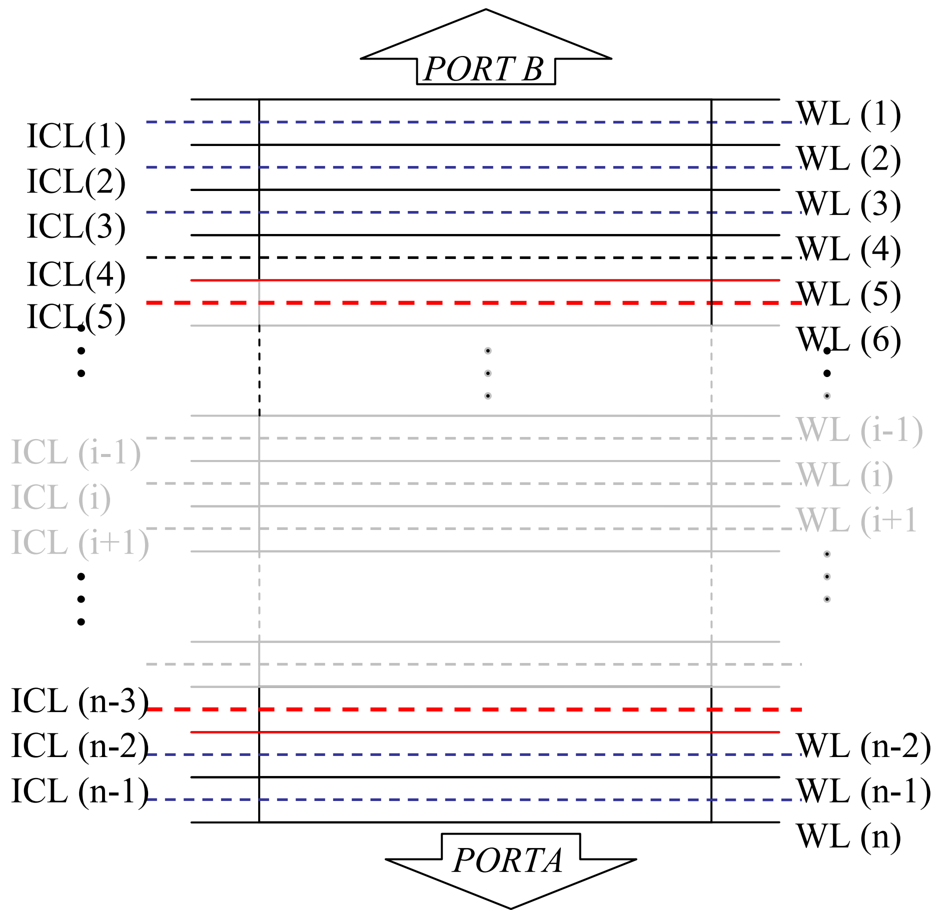


Figure 24. Generic DMP Model with Two Isolation Nodes.

6.4.2 Course-Grain DMP-2

In order to find optimal partitioning with minimum hardware overhead, we force partitioning to remain fixed for a maximum time period. When a previously selected isolation line can be used for a current configuration, no repartitioning is required. In coarse-grain DMP-2, an isolation node is placed on bit lines between every *n* word lines. DMP-2 minimizes the switching of isolation nodes by identifying whether or not a current partition can facilitate new accesses.

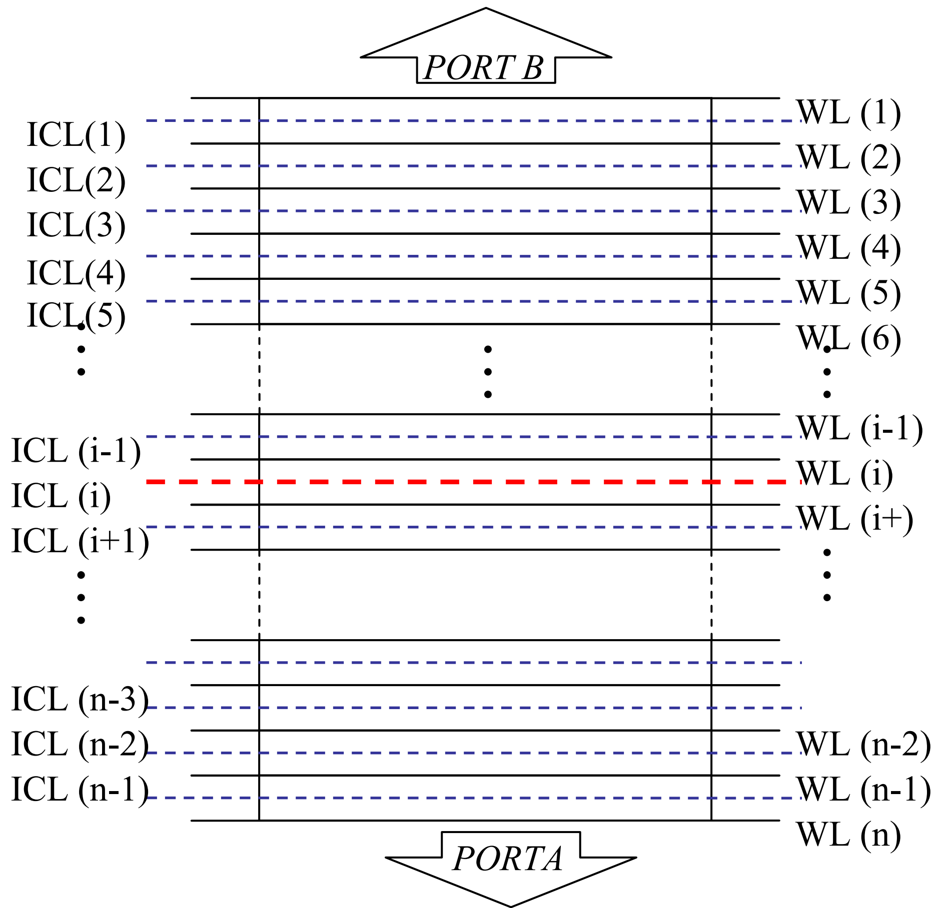


Figure 25. DMP with Single Isolation Node

The generic fine-grain DMP-2 algorithm is pseudo-coded as follows:

addr (A) <1:n>;

addr (B) <1:n>;

where addr (A) = i > addr (B) = j ;

k = current ICL;

if (j ≤ k < i) return NULL (no new DMP);

else return (j + (i-j)/2);

Figure 25 shows course grain DMP model. On average, isolation node will be activated in the center and bit line pre-charge and leakage currents are reduced to half the value of typical hardwired multi-port memory. Shorter active bit lines also means less latency.

6.5 Cell Structure

Dynamically configured memory uses simple single-port memory cells with six transistors along with a bit line pair and a word line. Like conventional dual-port memories, it uses differential read and write.

6.6 Advantages of Dynamically Configured Memory

Dynamically configured memory uses significantly less real-estate than the conventional dual-port memories. Even if dynamically configured memory is designed with external signals and an isolation node, two transistors and a pair of bit

lines are reduced per cell. Therefore, less interaction coupling charge is produced as the numbers of bit lines are reduced to half.

6.7 Dynamic Memory Operations

The proposed dynamically configured memory has all the functionalities of a conventional dual-port memory, at the same time it eliminates the need of redundant bit-lines. This will reduce the size of a conventional memory significantly. The reason behind this is that the data access ports are available at both ends of the bit line and the numbers of port are same as that of conventional dual-port memories. Dynamically configured memory operations have to be carefully defined. These are explained below.

6.7.1 Read / Read on same Cell

Present dual-port memory architecture requires redundant bit lines to read data at two different ports. In these memories, during a read operation from the same cell, two word lines corresponding to the same address are turned high in succession, while selecting a given row of the cell. Analyzing dual-port transistor of fig 2 we can see that if a read operation is performed on the same cell same charge will be present on both bit lines, if a *0* is stored in the cell both bit lines *True Bit Line 0* and *True Bit Line 1* will discharge using *T5* and *T6* as transfer devices.

Dynamically configured memory uses the same rail to read data at two different ports.

No isolation circuit and additional hardware is required while data is being read simultaneously. Beta ratio is critical as in the case of traditional dual-port memories. If two ports are accessing the same cell at the same time, the cell should be capable of sinking at twice the speed of conventional single-port memory. Like the conventional two port memories, the stability of the cell is much trickier.

6.7.2 Read / Write on same Cell

Simultaneous read and write on the same cell is typically similar to that of conventional multi-port memory. Dynamically configured memory also has to follow the same legal operations defined for conventional multi-port memory cell, isolation is not required. Like pseudo multi-port memories, dynamically configured memory also has to use different clock for both ports. This clock can be generated within the memory with the first clock, the first operation is performed. Only after this operation is completed, can the second operation be performed. These operations will be performed exactly as in a traditional memory.

6.7.3 Write / Write on same Cells

This operation is also the same as that in conventional dual-port memories. Isolation node does not need to be activated during write operation on the same cell. It is critical in dynamic memories that only one write operation can be performed on a single cell in one clock cycle. If two write operations are performed, then the legitimacy of data cannot be guaranteed.

6.7.4 Operations on Two Different Cells

Conventional dual-port memory cells use extra pair of transistors, a pair of bit lines and a word line to allow simultaneous accesses within one clock cycle. If the operations are performed on two different columns, then redundant hardware do not play any role and only one bit line or compliment bit line is activated during this operation. Statistically, most of the operations in dual-port memory are performed on different columns, thus redundant hardware is not utilized during these operations.

In the proposed dynamically configured memory architecture, we do not need extra hardware to ensure multitasking. Before performing any two simultaneous operations on two different cells in a block, the isolation circuit is activated and dynamically configured memory is divided into virtual memory blocks (Fig 15). Once isolation is active, all operations on each block are similar to those on conventional two port memories.

6.8 Delay Analysis of Dynamically Configured Cache

Dynamically configured cache provides dual/multi-port access by bit line segmentation. In addition to multi-port access, this architecture can significantly reduce power dissipation depending upon whether the isolation node is activated before or after bit lines are pre-charged. Pre-charging bit line after sub-array selection can significantly reduce power dissipation. However, on the downside this will degrade the performance of cache memory. In addition to this delay, an additional delay is also incurred due to isolation node setup. This can be minimized by overlapping with banks and bit line pre charging. In the rest of this section we will

investigate timing analysis and operations of dynamically configured memories.

6.8.1 Pre-Charge whole Memory

In conventional high speed memories, all memory banks are pre-charged before any operation is performed. Dynamically configured memories introduce isolation nodes and thus an additional isolation node setup time (T_{NS}). Isolation node setup time (T_{NS}) is a partial address decoders' delay.

An additional delay of ΔT is introduced before any address logic is applied to SRAM. This is to make sure that isolation is performed before any other operation in the memory. This approach does provide dual-port access but does not save power by selective sub-array selection. Timing components follow:

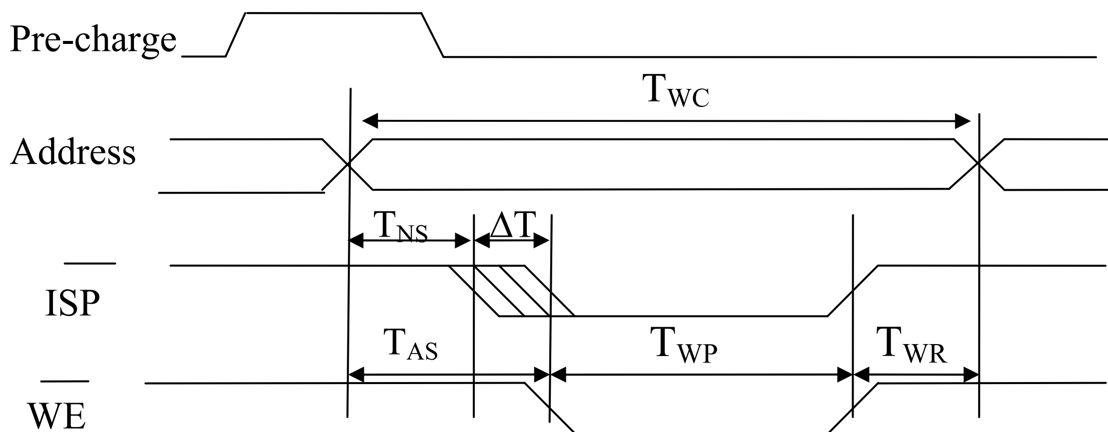


Figure 26. Memory Operation Timing with DMP

$T_{NS:}$ Isolation node setup time

$T_{AS:}$ Address setup time

$T_{CW:}$ Chip enables to end of write

ΔT: Induced delay before memory operations

T_{WP}: Write pulse width

$T_{WR:}$ Write recovery time

6.8.2 Precharging required Array or Bank

As mentioned above, sub-banking techniques reduce powers dissipation by precharging only sub-bank that contains the desired data. In addition to bank selection delay (T_{BA}), bit line precharging time (T_P) also incurs a delay. Since isolation control line setup time (T_{ICL}) is smaller then pre-charge time (T_p) and overlaps bank selection delay (T_{BA}), introduction of isolation node should not affect performance of conventional memories that uses selective precharging.

$$T_P + T_{BA} + \Delta T < T_{ICL}$$

$$T_{AS} = T_{NS}$$

Where, T_{BA} is a small 1-3 bit address decoding delay.

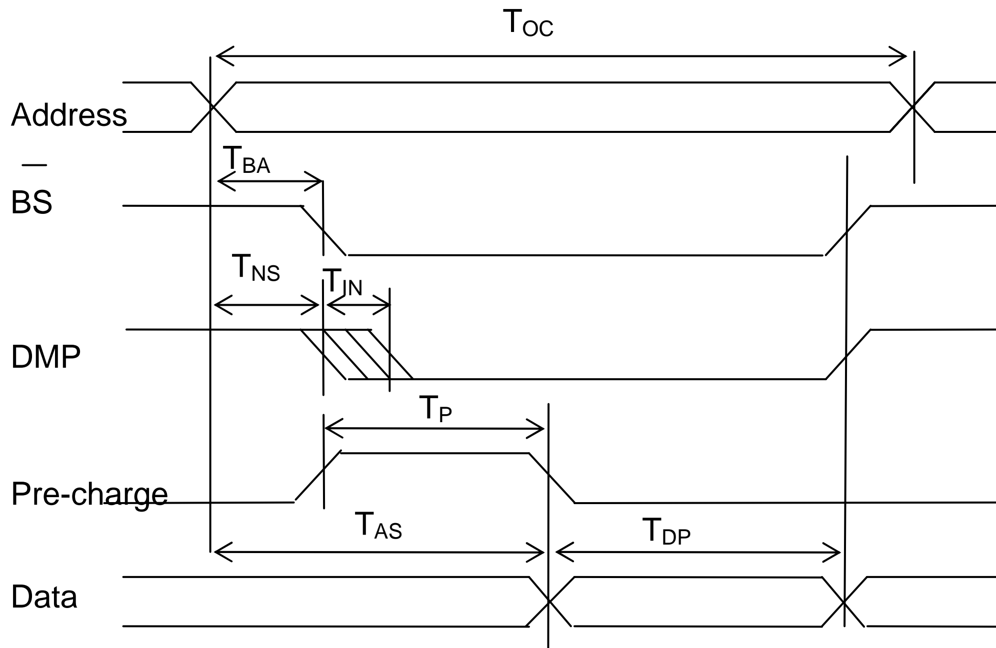


Figure 27. Dynamically Configured SRAM with Bank Pre-charging

$T_{OC:}$ Memory operation cycle time

$T_{BA:}$ Bank address decoding delay

T_{NS} Node setup time

$T_{P:}$ Precharging time delay (T_P).

$T_{AS:}$ Address setup time

T_{DP}: Valid data

6.8.3 Pre-charge Sub-Array after Isolation Node is Setup

We can further reduce power dissipation if we pre-charge our circuit after isolation node is setup. Moreover, we can reduce energy dissipation by pre-charging only the part of the sub-bank. Also, performance enhancement can be achieved by reducing bit line delays.

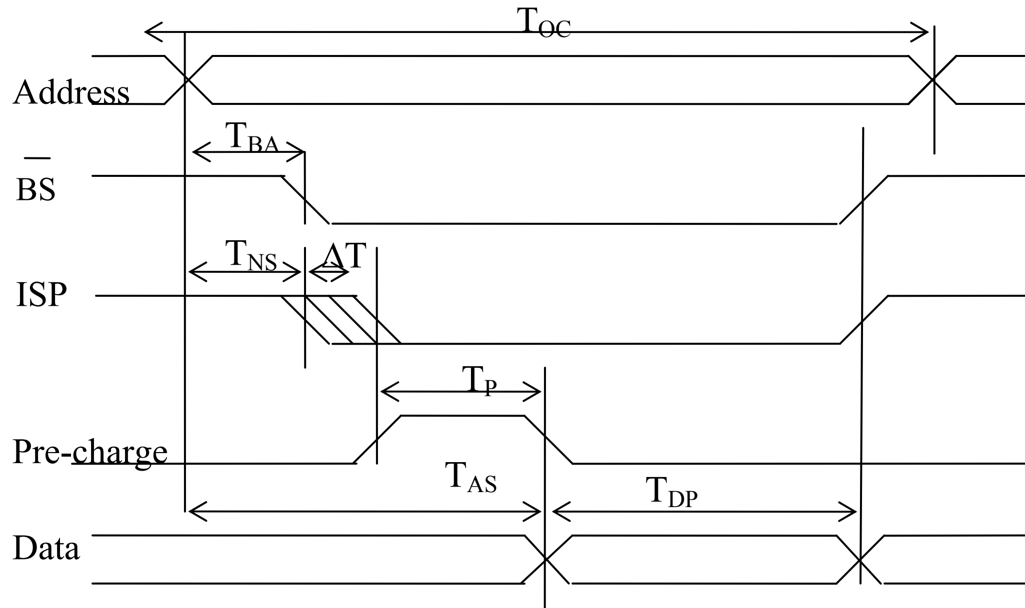


Figure 28. Partial Sub-Array Precharging

T_{OC}: Operation cycle time

T_{BA}: Bank address decoding delay

T_P: Precharging time delay (T_P).

T_{NS}: Isolation node setup time

T_{AS}: Address setup time

ΔT : Induced delay before memory operations

T_{DP}: Valid pulse width

This setup introduces a delay due to isolation node setup, which is followed by bit line precharging delay. Total address setup time is

$$T_{AS} = T_{NS} + T_P$$

$$T_{NS} \geq T_{BA}$$

Summary

Existing approaches presented in previous chapters have partitioned cache at the circuit and architectural level by enabling/disabling sub-banks from both the performance and energy viewpoints [1, 5]. In this chapter we proposed a dynamically configured cache that allows cache resizing by bit line segmentation. Proposed architecture is capable of providing multi-port access without duplicating bit lines. Dynamically configured cache memory uses isolation control lines (ICL) to allow on-chip cache to be partitioned and accessed by multiple processors and threads at the same time.

We also presented cache partitioning algorithm and reconfiguration policies for minimum bit line latency and power dissipation. Detailed timing analysis of memory operations in dynamically configured cache is also presented in this chapter.

Chapter 7 Device Modeling

Introduction

It is extremely important to sustain accuracy across process generations; general process scaling rules tend to fail in nano meter regime. Pre-silicon device modeling allows evaluation of circuits using simulators when fabrication is not economical. It is becoming critical in circuit design research to predict performance of devices before fabrication. A realistic device model enables the circuit designer to evaluate their circuits under realistic conditions [47]. In order to accurately predict characteristics of nano scale devices several physical effects have to be considered to model devices, accurately [48]. In this chapter we will present device model established using Zhao's [48] method.

7.1 Additional Device Modeling

As device scales down below 100 nm, several physical factors become significant. Such factors had little or no effect on previous performance [48]. Accurate MOSFET modeling is extremely critical in determining the characteristics of devices and

circuits. For our simulation purposes we have used predictive design model (PTM) developed by Zhao [48].

Berkeley Model 4 (BSIM 4) has proven to be too simple to predict behavior of nano-scale technologies. BSIM 4 ignored effects like quantum tunneling between the poly gate and inversion layer as oxide thickness is decreased. Berkeley's predictive model (BPTM) considered more physical parameters and provided a reasonable model for 180 to 45nm technologies. Shrinking parameters like effective gate length L_{eff}, oxide thickness T_{ox}, threshold voltage V_{th}, drain to source resistance R_{ds} and voltage V_{dd} , while keeping other parameter unchanged, has proven to be ineffective and error prone in nano meter regime [48,49]. As an example, scaling V_{th} not only requires the change of channel doping but also impacts physical parameters such as mobility μ_o and saturation velocity V_{sat}. Predictive model developed in [48] overcame these limitation and refined Berkeley predictive models [39] developed earlier. The accuracy of PTM has been verified, for I_{on}, it shows an error of less then 10%.

Out of more than 100 parameters in compact transistor model, only about ten are used by the PTM to determine the characteristics of the devices. These ten parameters include technology parameters, process parameters and also physical parameters [50]. Accurate modeling of these values is extremely important in developing a reasonably accurate SPICE model. Based on the data collected by literature, survey and from published industry data presented by Zhao [48], quantum tunneling E_{TO} scaling pace is slowing down over last few years. V_{dd} is scaling much sharper then V_{th}, which

remains almost constant to avoid increase in sub-threshold leakage current. N_{ch} is carefully evaluated in PTM, as it affects both the carrier mobility and saturation velocity V_{sat} of devices. Simulation model follows.

7.2 Simulation Model

We have used BSIM Predictive Technology Model [48] to calculate resistance and capacitance of MOS devices. Bit line delay, wire resistance and capacitance is an increasing source of concern in sub-micron cache design [51].

7.2.1 Wire Capacitance and Resistance

Wire capacitance is calculated according to formula presented by [61].

$$Cw = \frac{2\varepsilon_d . \varepsilon_o [1 + 2[T/W]^2}{T/W} + C_{fringe} \qquad (8)$$

Where εd is the dielectric constant, εo is the permittivity of free space. *Cfringe* is the fringing capacitance. As presented in [61] we assumed it to be equal to 0.04fF/μm in all technology nodes.

Wire resistnce is calculated by formula :

$$Rw = \frac{\rho}{WT} \qquad (9)$$

where ρ is resisotivity of copper and W and T are width and thickness respectively.

For simulation purpose we use *Rw* and *Cw*, as presented in Table 1 by [61].

Table 1. Wire Capacitance and Resistance as technology scales from 130 to 32nm

Technology	Rw [Ω/μm]	Cw[fF/μm]
130nm	0.06	0.3
90 nm	0.12	0.22
65 nm	0.2	0.2
45 nm	0.44	0.2
32 nm	0.73	0.2

Though it is difficult to predict length and width of the wires in the earlier stages of design, we used a stick diagram and calculated the length of the wire to be 60λ per memory cell. Width of wire is set at λ per cell. Figure 29 shows calculated and predicted wire resistance for a group of eight cells as technology scales down from 90nm to 32nm.

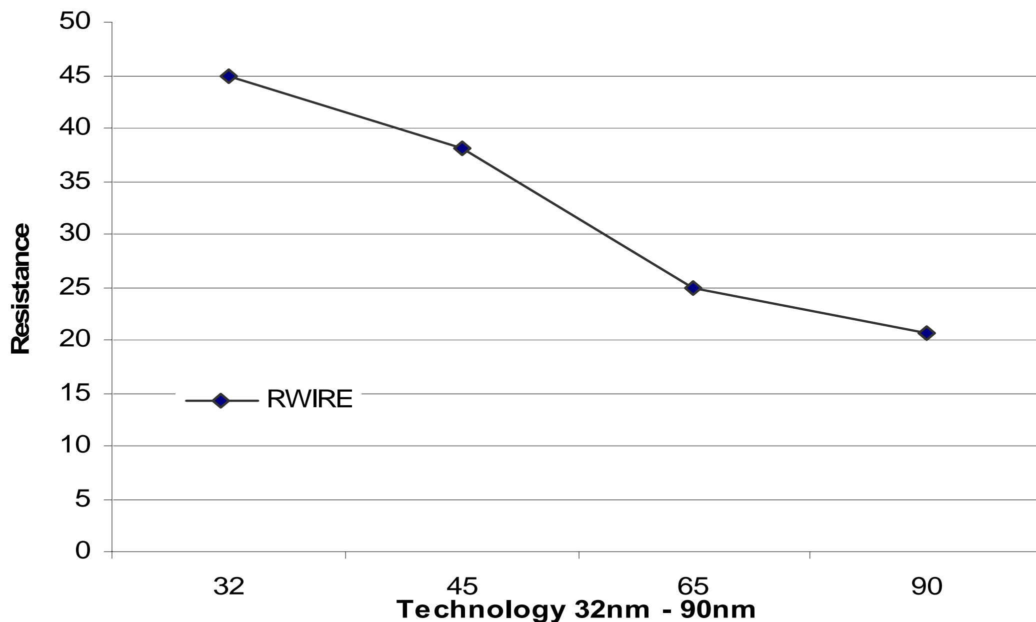


Figure 29. Bit line Resistance as Technology Scales Down

The key observation from figure 29 is that resistivity of wire will increase significantly as technology shrinks. While new and less resistive materials such as carbon nano tubes are being studied to replace copper wire, it is going to take more time before actual devices based on these studies become available. An alternate popular approach is to reduce bit line capacitance and resistance by wire resizing [40, 41].

7.2.2 MOS Device Modeling

The effective capacitance and resistance of MOS device can be calculated using BSIM predictive model. We used customized predictive model files available at http://www.eas.asu.edu/~ptm [60] to determine device parameters. Major device

parameters have been extracted from PTM published data [48], and are presented in table 2.

Table 2. Device parameters extracted from PTM

Tech. node (nm)	32	45	65	90
Vdd (V)	0.9	1	1.1	1.2
Toxe (nm)	1.65	1.75	1.85	2.05
Leff (nm)	12.6	17.5	24.5	35
Vth (V)	0.295	0.292	0.289	0.284
Rdsw (Ω/μm)	150	155	160	170

Source/drain parasitic resistance (R_{dsw}) is crucial for accurate prediction of device performance [48, 50]. PTM use I_V curve in linier region to extract values of $R_{dsw.}$ Figure 30 shows the wire resistance trends evaluated from [48, 60] as technology scales down from 32nm to 90nm.

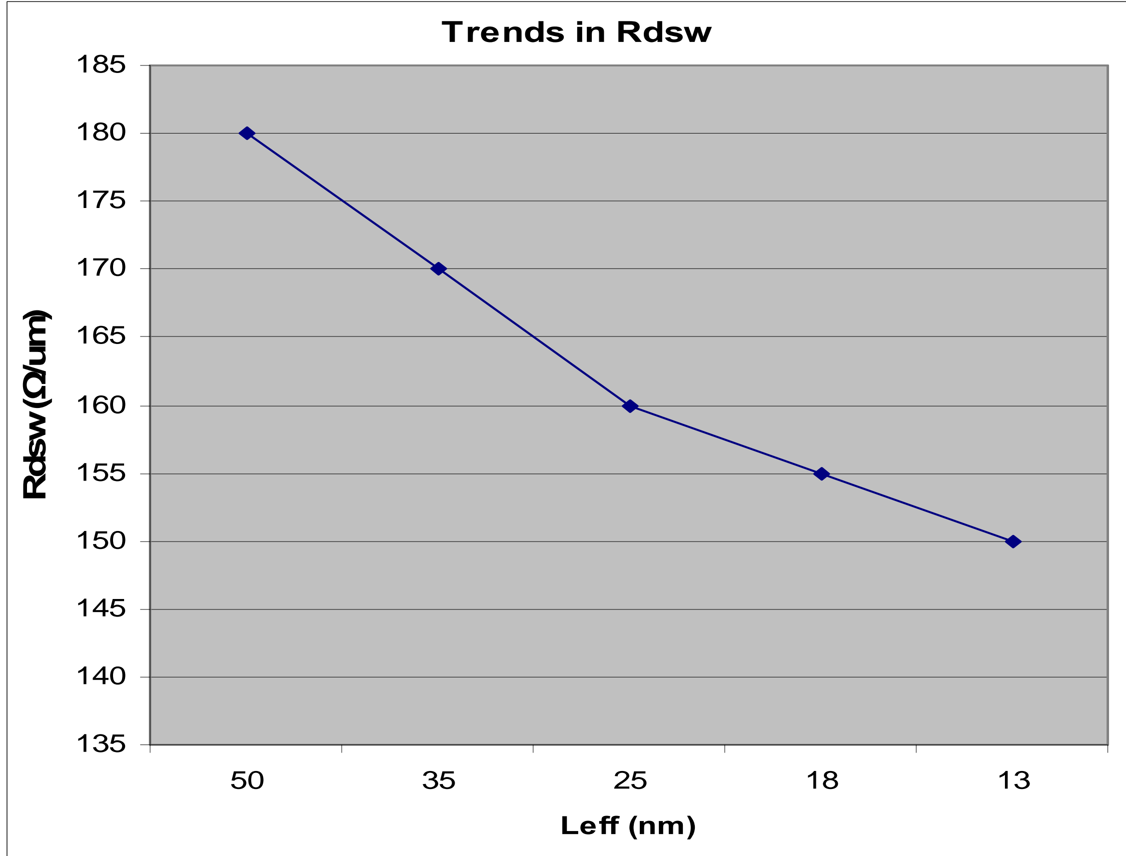


Figure 30. R_{dsw} Trend as Technology Scales-Down form 90nm to 32nm

BSIM 4 [24] models parasitic resistance as:

$$R_{ds} = \frac{\left\{R_{dsw\min} + R_{dsw}\left[\left(\Pr wb(\sqrt{\phi s - Vbseff}\right) - \sqrt{\phi s} + \frac{1}{1+PRWG.Vgseff}\right]\right\}}{\left(10^{6} W_{eff}\right)^{wr}} \qquad (10)$$

Where *Rdsw* is the resistance per unit width, *Wr* is a fitting parameter, *Prwb* and *Prwg* are the body bias and the gate bias coefficients, respectively. Using BSIM predictive model, we extracted parameters of equation 10 and calculated channel resistance of NMOS and PMOS devices and transmission gates that are used as isolation nodes.

Table 3. MOS Device Resistance

Tech (nm)	R_{NMOS} (Ohms)	R_{PMOS} (Ohms)	R-Tgate (Ohms)
32	33.75	58.5	21.33
45	36.25	43	19.66
65	41	31	17.65
90	45	25	16.7

Figure 31 shows transmission gate and MOS devices resistance generated with the BSIM Predictive Technology Model [48]. As technology scales down, device resistance decreases and the delay due to isolations nodes is likely to become smaller.

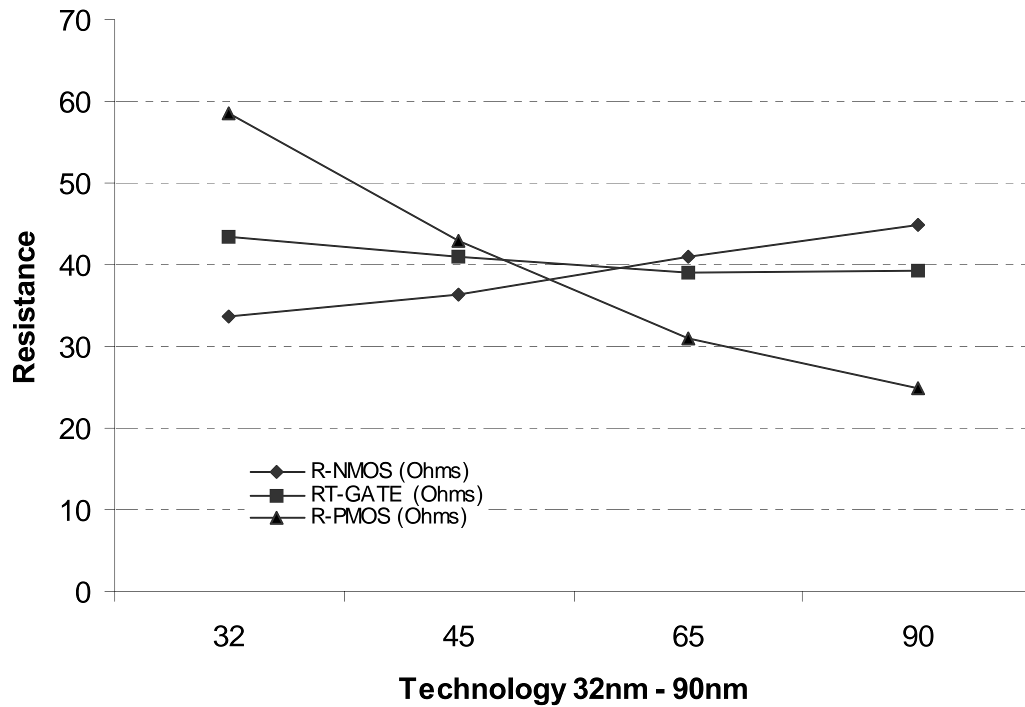


Figure 31. Technology Scaling Impact on MOS Devices and Transmission Gate Resistance

To model the capacitance of pass transistors and isolation nodes we used simple MOSFET capacitive model presented in [53] given in Figure 32.

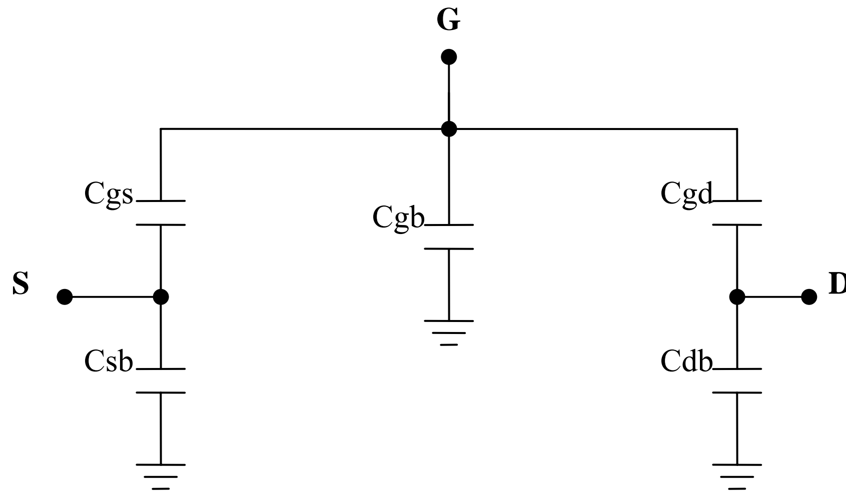


Figure 32. MOSFET Capacitive Model

The gate capacitance of a transistor consists of 2 components, the parasitic capacitance between gate and source and parasitic overlap capacitance.

The value of capacitance depends on whether the transistor is operating in cutoff, resistive or saturation mode. The geometry of transistor also affects the capacitance of devices. Detailed equations for estimating the drain and gate capacitances of transistors in different scenarios are adapted from [53], and are given below:-

$C_{GS} = C_{GCS} + C_{GSO}$;

$C_{GD} = C_{GCS} + C_{GDO}$;

In saturation mode $C_{GCS} = C_{GCS} = 2/3\ C_{ox}\ WL$. From [60] and using the prodictive model, we calculated the capacitance of MOS devices presented in table 4 below :

Table 4. Capacitance of MOS Devices in Sub-Micron Regime

Tech (nm)	cgdo	Leff (nm)	Tox (nm)	$C_{G\text{-}NMOS}$	C_{GD}	$C_{G\text{-}PMOS}$
32	8.50E-11	12.6	1	47.7E-18	8.50E-11	95.5 E-18
45	1.10E-10	17.5	1.1	84.8E-18	1.10E-10	169.6 E-18
65	1.50E-10	24.5	1.2	160E-18	1.50E-10	320 E-18
90	1.90E-10	35	1.4	300E-18	1.52 E-16	600 E-18

From Table 4 and figure 31, we observe that as technology scales down, MOS device parasitic capacitance and resistance decreases. The delay due to isolations nodes will become smaller.

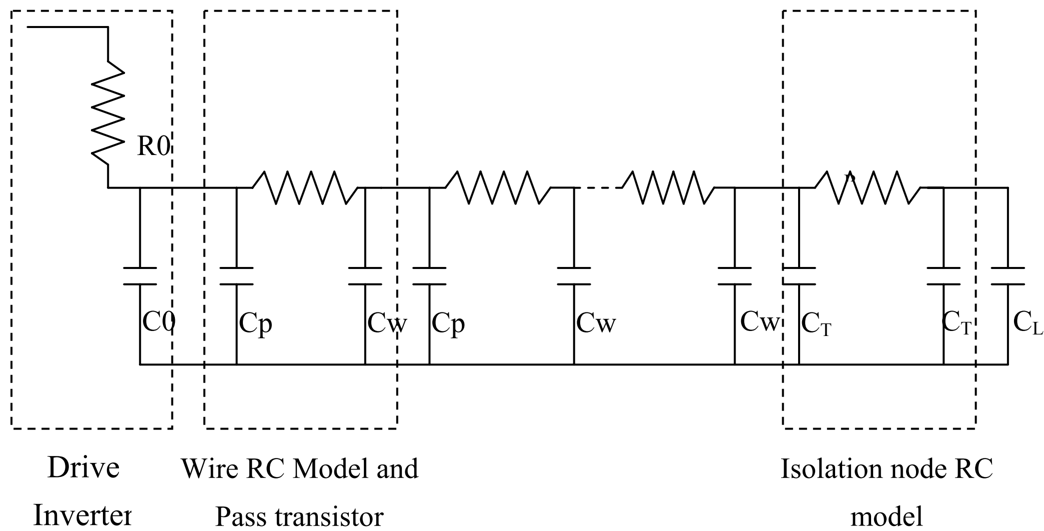


Figure 33. RC Model of Active Bit line with Isolation Node

Fig. 33 shows the circuit simulation model for the active bit lines with DMP. The isolation node (implemented as a transmission gate) is modeled by R_T and 2 C_T. With Wn/Wp =0.5, using 90nm technology, we calculated the capacitance and resistance for the transmission gate. For simulation purpose, the drive inverter of SRAM cell is replaced with equivalent R_O and C_O. C_L is the total load capacitance of the local sense amplifier structure.

Summary

As we delve deeper into the sub-micron region, the trend of using a large on-chip cache and logic is increasing. Accurate pre-silicon device modeling is extremely important to predict characteristics of nano-scale devices. In this chapter we used the predictive technology model (PTM) to characterize SRAM in submicron regime. PTM is used to calculate resistance and capacitance of MOS devices as technology scales from 90nm to 32nm. Bit line delay, wire resistance and capacitance are also calculated using BSIM and PTM models.

Chapter 8 Analytical and Simulation Results of DMP Implementation

Compared with conventional memory architecture, newly proposed dynamically configured cache reduce bit line pre-charge and leakage currents to half the value of typical hardwired multi-port memory. Area is reduced to half and shorter active bit lines means less latency. The individual areas where improvement has been achieved are described as follows.

Power dissipation: Major design objective in today's chip design has become the reduction in power dissipation rather than an increase in performance in large scale systems [45]. Cache memory often accounts for a large part of the total system power dissipation. This happens as the bit lines remain pre-charged even when not accessed. The proposed architecture reduces leakage power by using bit line isolation and selective pre-charging. Dynamically configured memory not only reduces leakage current by eliminating pass transistors in hardwired multi-port memory, but also

reduces the bit line leakage power by eliminating bit lines pair.

$$Idcore = N * M \begin{pmatrix} Idsub(T1) + Idsub(T5) + Idsub(T4) \\ + Idsub(T7) \end{pmatrix} \tag{11}$$

For a memory core with N rows and M columns the leakage current is reduced to less than half the value of hardwired dual-port memories. Equations 11 and 12 show leakage current in conventional and dynamically configured dual-port memories respectively.

$$Idcell = \frac{N}{2} * M\left(Idsub(T1) + Idsub(T5) + Idsub(T4)\right) \tag{12}$$

Area: Since the silicon area (or foot print) of multi-port cache memory is dominated by word and bit lines [5], reducing the number of bit lines can reduce the silicon area. For multi-port cache memory, the proposed DMP reduces the number of bit-lines to half, thus reducing the silicon area significantly.

Delay analysis: *A m*emory simulation model with DMP is presented in Figure 33. Each delay component is estimated individually and then combined into a form that calculates access time. Using 90nm technology data we estimated bit line delay by calculating resistance and capacitance in the drive circuit, wires, transmission gate and load. Cadence *SPECTRE* is used as simulation tool, resistance and capacitance

per cell is calculated to be 0.36 Ω and 0.6675 *fF*, respectively. Capacitance of pass transistor in cutoff mode is calculated to be 76 *fF*. Transmission gate resistance R_T and two ground capacitances C_T are calculated to be 39.2 Ω and 376 *fF*, respectively.

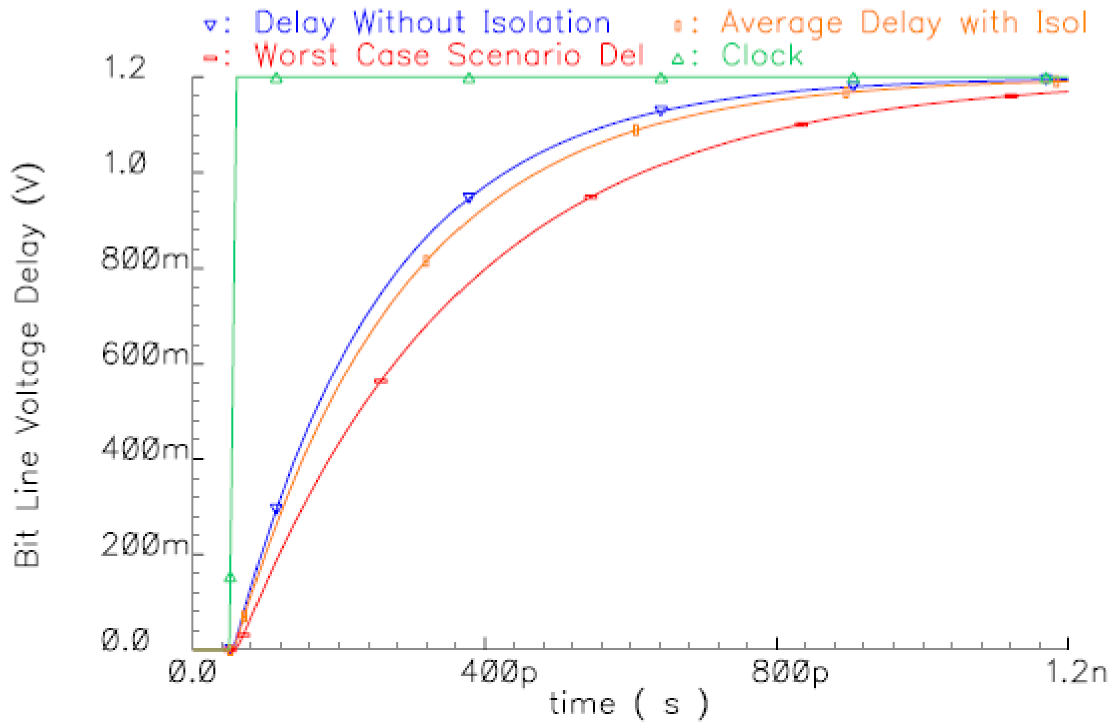


Figure 34. Bit line Delay With/Without Isolation Nodes

Figure 34 compares the bit line delay in a 64 word-line block, with and without isolation nodes. Each isolation node separates an 8 word-line sub-block from another. In the worst case scenario, the simultaneous access addresses are near either the top or the bottom of a cache block (hence one active bit line is much longer than the other), the delay of the longer active bit line is increased by about 200 *ps* compared to the hardwired bit line without the isolation nodes. When accesses are facilitated with a DMP in the center of the bit lines, the delay of the active bit lines for each section is

about 6% more then that of the traditional dual-port hardwired memory.

Delay analysis using DMP-1: Fine grain DMP partitioning not only provides dual-port access but also reduces delay and minimizes power dissipation by keeping active bit lines as small as possible. Simulation results of DMP-1 using eight isolation nodes on 64 word-line blocks are shown below:

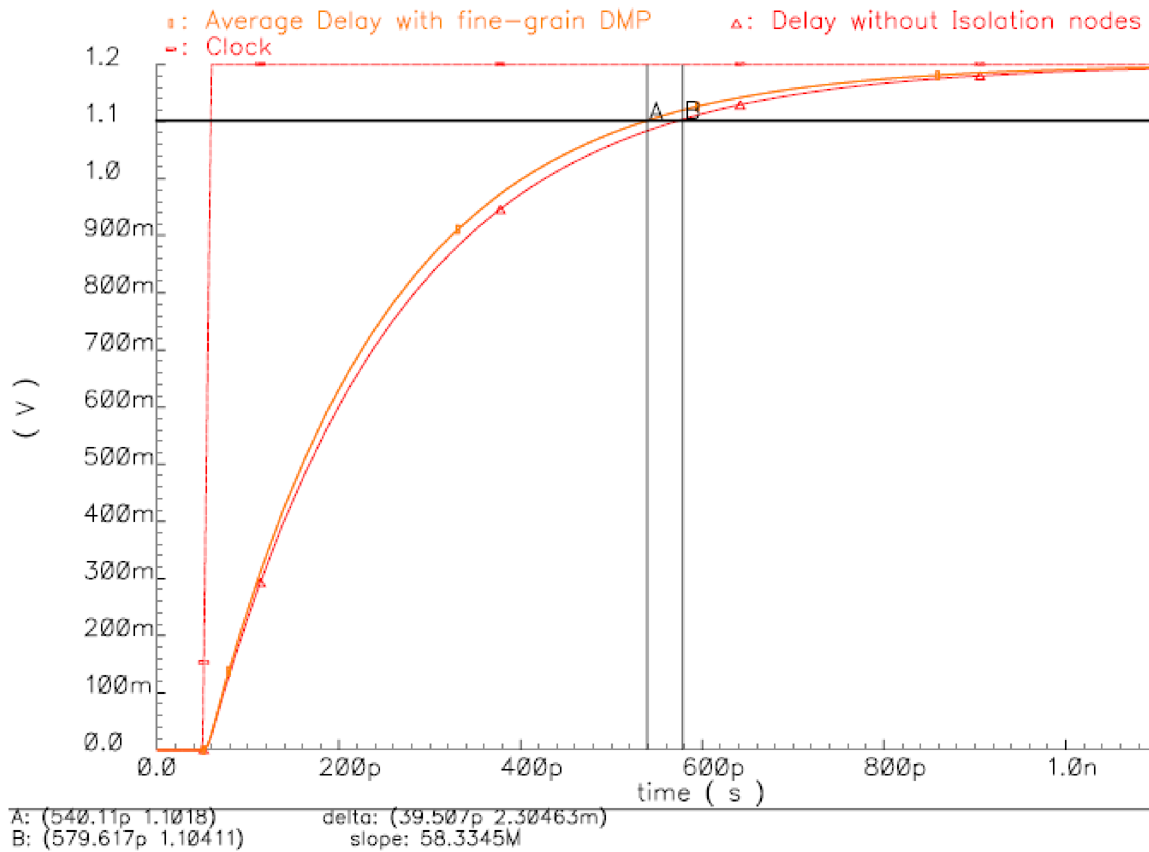


Figure 35. Delay Analysis with DMP-1

Compared with traditional memories, the proposed architecture access time is 40ps smaller. Thus a gain of approximately 7% is achieved by the newly proposed

architecture.

Delay analysis using DMP-2: Course grain DMP partitioning uses only one isolation node. To keep the overhead to a minimum, we force partitioning to remain fixed for a maximum time period. When a previously selected isolation line can be used for a current configuration, no repartitioning is required.

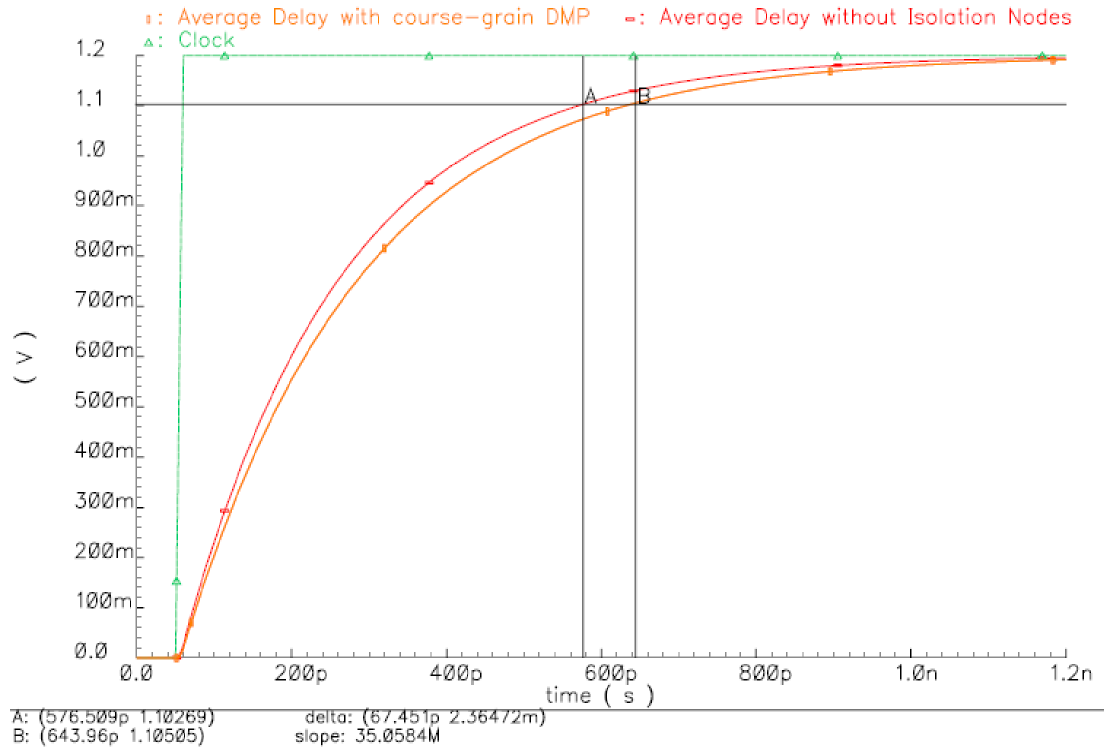


Figure 36. Delay Analysis with DMP-2

Figure 36 shows simulation results of course grain DMP algorithm. Compared with traditional memories, the proposed architecture access time is 60ps larger. This algorithm results in approximately 9% performance degradation.

Combined with local sense amplifiers and port multiplexing, the proposed DMP can support efficient multi-port cache architecture to reduce silicon footprint, wiring contention, power dissipation and bit line latency.

Chapter 9 Conclusion

We have proposed an area and energy-efficient multi-port memory architecture. It employs new DMP techniques, which uses isolation nodes and control lines to dynamically partition the bit-lines of a memory block into virtually-isolated sections, so that they can be accessed simultaneously and independently. Compared to the classic hardwired multi-port memory architecture, the new DMP reduces the area of the memory to almost half, consequently reducing power dissipation. It facilitates efficient designs with no significant impact on the timing of memory operations. It reduces the use of silicon area largely due to the elimination of additional bit lines.

One of the advantages of the proposed architecture is its ability to provide dual-port support without duplicating word and bit lines for the second port. The DMP concept can be applied to multi-port cache designs as well. A second advantage is the reduction of bit line precharge/leakage currents. DMP makes effective bit lines shorter that those in the conventional hardwired dual/multi-port memory and therefore, require less energy to keep the bit lines from being highly charged. As a result of the shorter bit lines, latency is also reduced, which is also facilitated by the

use of local sense amplifiers for each group. A third advantage is that, due to the elimination of the dedicated word and bit lines for the second and additional ports, the entire cache block uses less silicon resources with smaller foot prints, which also reduces interconnect routing contentions due to less complex wiring.

References

[1] R. D. Adams, "High Performance Memory Testing: Design Principles, Fault Modeling and Self-Test." Kluwer Academic Publishers, Boston, 2003.

[2] A. Agrawal, B. C. Poul, H. Mahmoodi, A. Datta and K. Roy, "A Process-Tolerant Cache Architecture for Improved Yield in Nano-scale Technologies." In the transactions on Very Large Scale Integration Systems, Vol. 13 No. 1, January 2005.

[3] S. Kim, N. Vijaykrishnan, M. Kandemir and M. J. Irwin, "Optimizing Leakage Energy Consumption in Cache Bit-lines." In The Journal of Design Automation for Embedded Systems, Vol. 9, No 1, pp. 5-18, Mar. 2004.

[4] M. Mamidipaka, K. Khouri, N. Dutt, and M. Abadir "Analytical Models for Leakage Power Estimation of Memory Array Structures" In International Conference on Hardware/Software and Co-design and System Synthesis (CODES+ISSS) pp. 146–151, 2004.

[5] J. Lee, W. Hong and S. Kim, "Design and evaluation of a selective compressed memory system" In International Conference on Computer Design pp.184 – 191, Oct. 1999.

[6] Y. CAI, M. Schmitz, A. Ejlali, B. Al- Hashmi, and S. Reddy, "Cache size selection for performance, energy and reliability of time –constrained systems." In IEEE Asia and South Pacific Conference on Design Automation,

pp 923-928, 2006.

[7] N. S. Kim, K. Flautner , D. Blaauw and T. Mudge, “Circuit and Micro-architectural techniques for reducing cache leakage power.” In IEEE Transaction on VLSI systems Vol. 12, No. 2, pp. 167-184, Feb. 2004.

[8] M. Powell, S. Yang, B. Falsafi, K. Roy, and T. Vijaykumar,, “Gated-Vdd A circuit technique to reduce leakage in deep-submicron cache memories.” In Proc. IEEE/ACM Int. Symposium on Low Power Electronics and Design, pp. 90-95, 2000.

[9] C. Zhang, F. Valid and W. Najajr, “A highly configurable cache architecture for Embedded System.” In the processing of 30th Annual International symposium on computer architecture, pp. 136 - 146 (ISCA ’03).

[10] X. Chen and H. Bajwa, “Energy-efficient dual-port cache architecture with improved performances.” In IEE Journal of Electronic Letters, Vol. 43, No. 1, pp. 12-14, Jan. 2007.

[11] S.-H. Yang and B. Falsafi, “Performance and Energy Trade-offs of Bit line Isolation in Nano-scale CMOS Caches.” Presented at the Workshop on Complexity-Effective Design (WCED) held in conjunction with the 30th International Symposium on Computer Architecture (ISCA-30), Jun. 2003.

[12] K. Nii, Y. Tsukamoto, T. Yoshizawa, S. Imaoka, Y. Yamagami, T. Suzuki, A. Shibayama, H. Makino and S. Iwade, “A 90nm Low Power 32-kB Embedded SRAM With Gate Leakage Suppression Circuit for Mobile Applications.” In IEEE Journal of Solid State Circuit, Vol. 39, No. 4, April 2004.

[13] A. Turier, L. B. Ammar and A. Amara, “Static power consumption management in CMOS memories” in IEEE International Symposium on Circuits and Systems, 2001, Vol. 4, pp. 506 – 509, May 2001.

[14] J.A. Butts and G.S. Sohi, “A static power model for Architects.” In the Proceeding of the 33rd Annual IEEE/ACM International Symposium on Micro-architecture (MICRO33), pp.191-201, Dec. 2000.

[15] B. Amelifard, F. Fallah, M. Pedram, “Reducing the sub-threshold and Gate-tunneling Leakage of SRAM cells using Dual-Vt and Dual-Tox Assessment.” In IEEE Proceedings of Design, Automation and Test, Vol. 1, pp. 1-6, 2006.

[16] S. D. Naffziger, G. Colon-Bonet, T. Fischer, R. Riedlinger, T. J. Sullivan and T. Grutkowski, “The implementation of the Itanium 2 microprocessor.” In IEEE Journal of Solid-State Circuits, Vol. 37, No. 11, pp. 1448 – 1460, Nov. 2002.

[17] R. S. Guindi and F. N. Najm, “Design Techniques for Gate-Leakage Reduction in CMOS Circuits.” In Fourth IEEE International Symposium on Quality Electronic Design, pp. 61-65, March 2003.

[18] C. McNairy, R. Bhatia, “Montecito: a dual-core, dual-thread Itanium processor.” In IEEE Micro, Vol. 25, No. 2, pp.10 – 20, 2005.

[19] R. Kalla, S. Balaram and J. M. Tendler, “IBM 5 chip: A dual-core multithreaded processor.” In IEEE Micro Vol. 24, Issue 2, pp.40 – 47, 2004.

[20] O.Takahashi, R.Cook, S.Cottier, S.H.Dong, B.Flachs, K.Hirairi, A.Kawasumi, H. Murakami, H. Noro, H. Oh, S. Onish, J. Pille, J. Silberman and S Yong, “The Circuit and Physical Design of the Synergistic Processor Element of a

CELL Processor." In Symposium on VLSI design of Technical Papers, pp. 20-23, 2005.

[21] C. McNairy and R. Bhatia, "Montecito: a dual-core, dual-thread Itanium processor" In IEEE Micro, Vol. 25, No. 2, pp.10 – 20, 2005.

[22] C.N. Keltcher, K.J. McGrath, A. Ahmed and P. Conway, "The AMD Opteron processor for multiprocessor servers" in IEEE Micro, Vol.23, No. 2 pp. 66 - 76, 2003.

[23] S. J. E. Wilton and N. P. Jouppi, "CACTI: an enhanced cache access and cycle time model" In IEEE Journal of Solid-State Circuits, Vol. 31, Issue 5, pp. 677 – 688.

[24] X. Xi, C. Hu, et al., BSIM4 Manual, UC Berkeley Device Group, 2005.

[25] S. Hattori and T. Sakurai, "90% Write Power Saving SRAM Using Sense-Amplifying Memory Cell." In IEEE Symposium on VLSI Circuits, pp. 46-47, 2002.

[26] M. Manusharow, A. Hasan, T. W. Chao, M. Guzy, "Dual Die Pentium D Package Technology Development" in Electronic Components and Technology Conference, 2006. pp. 303-309, 2006.

[27] K. Flautner, N.S. Kim, S Martin, D. Blaauw, T. Mudge, "Drowsy Cache: Simple Techniques for Reducing Leakage Power." In 29th Annual International Symposium on Computer Architecture, pp.148 - 157, May 2002.

[28] A. Karandikar and K.K. Parhi, "Low power SRAM design using hierarchical divided bit line approach." In Proc. Int. Computer Design: VLSI in Computers and Processors, pp. 82-88, 1998.

[29] B. D. Yang and L. S. Kim, "A low Power SRAM Using Hierarchical Bit Line and Local Sense Amplifier." In IEEE Journal of Solid State Circuits, Vol. 40, No 6, pp. 1366 – 1376 Jun. 2005.

[30] D. H. Albonesi, "Selective Cache Ways: On-Demand Cache Resource Allocation," in Proc. of the 32nd Annual International Symposium on Microarchitecture, pp. 248-259, Nov. 1999.

[31] R. Rao, J. Wence, D. Franklin, R. Amirtharajah and V. Akella, "Exploiting Non- Uniform Memory Access Pattern through Bit Line Segmentation.," presented at the Workshop on Memory Performance Issues, in conjunction with High Performance Computer Architecture (HPCA), Feb. 2006.

[32] K. Ghose, M.B. Kamble, "Reducing power in superscalar processor caches using sub-banking, multiple line buffers and bit-line segmentation," in Proc. of the International Symposium on Low Power Electronics and Design, pp.70–75, Aug. 1999.

[33] J. Kao, and A. Chandrakasan, "Dual-Threshold Voltage Techniques for Low-Power Digital Circuits," IEEE Journal of Solid-State Circuits, Vol 35, Page(s): 1009~1018, Jul 2002.

[34] C. H. Kim and K. Roy "Dynamic V/sub t/ SRAM: a leakage tolerant cache memory for low voltage microprocessors" In Proceedings of the 2002 International Symposium on Low Power Electronics and Design, pp: 251 - 254, ISLPED '02.

[35] J. T. Koa and A. P. Chandrakasan, "Dual threshold voltage techniques for Low-Power digital circuits," in IEEE Journal of solid state Circuits, Vol. 35,

No.7, pp: 1009-1018, Jul. 2000.

[36] S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, and J. Yamada, “1-V power supply high-speed digital circuit technology with Multithreshold-voltage CMOS,” in IEEE J. Solid-State Circuits, Vol. 30, pp. 847–854, Aug. 1995

[37] C. Thondapu, P. Elakkumanan, R. Sridhar, “RG-SRAM: a low gate leakage memory design.” In the Proc. of the IEEE Computer Society Annual Symposium on VLSI, pp. 295–296, 2005.

[38] G. Chandra, P. Kapur, K.C. Saraswat "Scaling trends for the on-chip power dissipation." In Proceedings of the IEEE Interconnect Technology Conference, pp: 170- 172, 2002.

[39] J. L. Hennessy and D. Patterson. “Computer Architecture: A Quantitative Approach,”2nd Edition Morgan Kaufmann Publishers, 1999.

[40] K. Itoh, K. Sasaki, and Y. Nakagome, "Trends in low-power RAM circuit technologies." In the Proceedings of IEEE, Vol 83, Issue: 4, pp 524-543, Apr 1995.

[41] R. Balasubramonian, D. Albonesi, A. Buyuktosunoglu, and S.Dwarkadas, "Memory hierarchy reconfiguration for energy and performance in general-purpose processor architectures" In the Proceedings of the 33rd annual ACM/IEEE international symposium on Micro-architecture, pp. 245 - 257, 2000.

[42] I. Fukushi, et al., “A Low-Power SRAM using Improved Charge Transfer Sense Amplifiers and A Dual-Vth CMOS Circuit Scheme,” IEEE Int.

Symposium on VLSI Circuits, Page(s): 142~145, Jun 1998.

[43] B. Luca, M. Alberto, P. Massimo, "Energy-aware design of embedded memories: A survey of technologies, architectures, and optimization techniques" In ACM Transactions on Embedded Computing Systems (TECS), Volume 2 , Issue 1, pp: 5 - 32, 2003.

[44] K. Itoh, et al., “A Deep Sub-V, Single Power-Supply SRAM Cell with Multi-VT, Boosted Storage Node and Dynamic load,” IEEE Int. Symp. on VLSI Circuits, pp. 132-133, Jun 1996.

[45] A. Turier, A. Ben and A. Amara, “Static power consumption management in CMOS memories”. In IEEE International Symposium of Circuits and Systems, 2001. Issue, 6-9 vol. 4, 2001.

[46] The impact of silicon nano wires technology on the design of single-work-function CMOS transistors and circuits. B. S. Amrutur and M. A. Horowitz, “Speed and power scaling of SRAM’s,” IEEE J. Solid-State Circuits, Vol. 35, No. 2, pp. 175–185, Feb. 2000.

[47] K. Arnim, E. Borinski, P. Seegebrecht, H. Fiedler, R. Brederlow, R. Thewes, J.Berthold, and C. Pacha, " Efficiency of body biasing in 90 nm CMOS for low power digital circuits" In Solid-State Circuits Conference, No. 21-23, pp. 175 - 178, 2004.

[48] A. Zeng, K. Rose, R. J. Gutmann, “Memory performance prediction for High-Performance Microprocessors at Deep Sub-micrometer Technologies” In IEEE Trans. on CAD of Integrated Circuits and Systems Vol. 25, pp. 1705-1718, 2006.

[49] W. Zhao , Y. Cao, "New Generation of Predictive Technology Model for Sub-45nm Design Exploration." In the Proceedings of the 7th International Symposium on Quality Electronic Design, pp. 585-590, Mar. 2006.

[50] K. Agawa, H. Hara, T. Takayanagi and T. Kurodo, "A bit line leakage compensation scheme in low voltage SRAMS" In Symposium on VLSI Circuits design of Technical Papers, pp. 70-71, 2000.

[51] Y. Cao, T. Sato, M Orshansky, D. Sylvester and C. Hu, "New Paradigm of Predictive MOSFET and Interconnect Modeling for Early Circuit Simulations." In IEEE conference on custom integrated Circuits, pp. 201-204, 2000.

[52] P. Kapur, J.P. McVittie, K.C. Saraswat, "Technology and reliability constrained future copper interconnects. I. Resistance modeling," in IEEE Transactions on Electron Devices, Vol. 4, pp. 590-597, 2002.

[53] J. Cong and Z. Pan "Wire width planning for interconnect performance optimization" In IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 21, No. 3 pp. 319-329, 2002.

[54] J. M Rabaey, A. chandrakasan, B. Nikolic "Digital integrated Circuits, A design prospective." Printed by Prentice Hall, 2003.

[55] S. Shi, and D. Z. Pan, "Wire sizing with scattering effect for nano-scale Interconnection" In IEEE 2006 conference on Asia South Pacific design automation, pp. 503 - 508, 2006.

[56] X Chen and L. Peh, "Leakage power modeling and Optimization in Interconnection Network" In IEEE Low Power Electronics and Design, 2003.

Vol.25, No. 27, pp. 90 – 95, 2003.

[57] Y. Choi and T. Kim: Memory layout techniques for variables utilizing efficient dram access modes in embedded system design. In Proc. of Design Automation Conference, pp. 881 – 886, 2003.

[58] N. Shibata, M. Watanabe, and Y. Tanabe, "A current Sensed high speed and low power First in First out Memory using a Word/bit line swapped port SRAM cell." In IEEE Journal of solid-state circuits, Vol. 37, No. 6, pp. 1-4, 2002.

[59] B. S. Amrutur and M. A. Horowitz, "Techniques to reduce power in fast wide memories." In IEEE Symposium on Low Power Electronics, 1994, Vol.10, No. 12, pp. 92 – 93, Oct 1994.

[60] http://www.itrs.net/

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

[62] C. Grecu, P.P. Pande, A. Ivanov and R. Saleh, "A scalable communication-centric SoC interconnect architecture.," in the Proc. of 5th International Symposium on Quality Electronic Design, pp. 343–348, 2004.

[63] Betty Prince, "High performance memories- New Architecture DRAMs and SRAMs Evolution and Functions." ISBN: 978-0-471-98610-2, John Wiley and Sons, 1999.

[64] P. Ranaganathy, S. Adve and N. P. Jouppi "Reconfigurable cache and their applications to Multi-Media Processing." In Proceedings of the 27th International Symposium on Computer Architecture pp. 214 - 224, 2000.