CIRCUIT AND ARCHITECTURE CO-DESIGN OF STT-RAM FOR HIGH PERFORMANCE AND LOW ENERGY

by

Xiuyuan Bi

Master of Science, New York University, 2010

Submitted to the Graduate Faculty of

the Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2016

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Xiuyuan Bi

It was defended on

November 11th 2016

and approved by

Hai (Helen) Li, Ph.D., Associate Professor, Department of Electrical and Computer Engineering

Yiran Chen, Ph.D., Associate Professor, Department of Electrical and Computer Engineering

William E. Stanchina, Ph.D., Professor, Department of Electrical and Computer Engineering

Minggui Sun, Ph.D., Professor, Department of Neurological Surgery

Samuel J. Dickerson, Assistant Professor, Department of Electrical and Computer Engineering

Dissertation Director: Hai (Helen) Li, Ph.D., Associate Professor, Department of Electrical and Computer Engineering

CIRCUIT AND ARCHITECTURE CO-DESIGN OF STT-RAM FOR HIGH PERFORMANCE AND LOW ENERGY

Xiuyuan Bi, PhD

University of Pittsburgh, 2016

Spin-Transfer Torque Random Access Memory (STT-RAM) has been proved a promising emerging nonvolatile memory technology suitable for many applications such as cache memory of CPU. Compared with other conventional memory technology, STT-RAM offers many attractive features such as nonvolatility, fast random access speed and extreme low leakage power.

However, STT-RAM is still facing many challenges. First of all, programming STT-RAM is a stochastic process due to random thermal fluctuations, so the write errors are hard to avoid. Secondly, the existing STT-RAM cell designs can be used for only single-port accesses, which limits the memory access bandwidth and constraints the system performance. Finally, while other memory technology supports multi-level cell (MLC) design to boost the storage density, adopting MLC to STT-RAM brings many disadvantages such as requirement for large transistor and low access speed. In this work, we proposed solutions on both circuit and architecture level to address these challenges.

For the write error issues, we proposed two probabilistic methods, namely write-verify-rewrite with adaptive period (WRAP) and verify-one-while-writing (VOW), for performance improvement and write failure reduction.

For dual-port solution, we propose the design methods to support dual-port accesses for STT-RAM. The area increment by introducing an additional port is reduced by leveraging

the shared source-line structure. Detailed analysis on the performance/reliability degradation caused by dual-port accesses is performed, and the corresponding design optimization is provided.

To unleash the potential of MLC STT-RAM cache, we proposed a new design through a cross-layer co-optimization. The memory cell structure integrated the reversed stacking of magnetic junction tunneling (MTJ) for a more balanced device and design trade-off. In architecture development, we presented an adaptive mode switching mechanism: based on application's memory access behavior, the MLC STT-RAM cache can dynamically change between low latency SLC mode and high capacity MLC mode.

Finally, we present a 4Kb test chip design which can support different types and sizes of MTJs. A configurable sensing solution is used in the test chip so that it can support wide range of MTJ resistance. Such test chip design can help to evaluate various type of MTJs in the future.

TABLE OF CONTENTS

1.0 INTRODUCTION 1

1.1 Challenges: 2

1.2 Contributions: 4

2.0 PRELIMINARY 6

2.1 STT-RAM Basics 6

2.2 Prior Art 7

3.0 PROBABILISTIC DESIGN METHOD FOR STT-RAM 10

3.1 Motivation 10

3.2 Write errors of STT-RAM 12

3.2.1 Stochastic Switching of MTJ 12

3.2.2 Asymmetric Switching Probability of STT-RAM 13

3.2.3 Block Error Rate (BLER) 13

3.3 Probabilistic STT-RAM Designs 14

3.3.1 WRAP 15

3.3.1.1 The optimal write pulse period τ_{opt} 15

3.3.1.2 The τ_{opt} configuration in WRAP 16

3.3.1.3 Overheads of WRAP 17

3.3.2 VOW 18

3.3.2.1 Design concept 19

3.3.2.2 Asymmetric SA w/ one-time precharge 19

3.3.3 Evaluation of Proposed scheme 20

3.4 Summary 24

4.0 DUAL-PORT CELL DESIGN FOR STT-RAM . . . 25
4.1 Motivations . . . 25
4.2 Dual-Port STT-RAM Design Challenges . . . 26
4.3 STT-RAM Design with Two Read/Write Ports . . . 29
4.3.1 Design Concept . . . 29
4.3.2 Reliability Analysis . . . 29
4.3.3 The Cell Configuration and The Operating Setup . . . 33
4.3.4 Layout Design . . . 34
4.4 STT-RAM Design with 1-Read/1-Write Port . . . 34
4.4.1 Design Concept . . . 34
4.4.2 Transistor Sizing and Operating Voltage . . . 35
4.4.3 Comparison of 2RW and 1R1W STT-RAM Designs . . . 36
4.5 Summary . . . 37
5.0 MLC STT-RAM DESIGNS . . . 39
5.1 Motivation . . . 39
5.2 Fundamentals of STT-RAM . . . 40
5.3 MLC STT-RAM Cell Design Exploration . . . 42
5.3.1 Design Challenges of Conventional MLC STT-RAM . . . 42
5.3.2 Exploring More MLC STT-RAM Cell Structures . . . 44
5.3.3 Observations and Motivation . . . 48
5.4 Application-aware Speed Enhancement Scheme . . . 49
5.4.1 Application-aware Speed Enhancement (ASE) . . . 50
5.4.1.1 Read Performance Improvement . . . 51
5.4.1.2 The Mode Switching Control . . . 51
5.4.2 Logic to Physical Mapping Strategies . . . 52
5.4.2.1 Direct Mapping . . . 52
5.4.2.2 Cell Split Mapping . . . 53
5.5 Optimization of Cache with Cell Split Mapping . . . 55
5.5.1 Intra-cell Swapping . . . 55
5.5.1.1 Write & Swap . . . 56

5.5.1.2 Read & Swap . . . 57
5.5.2 Migration Method . . . 57
5.5.2.1 Counter-based Migration . . . 57
5.5.2.2 Aggressive Migration . . . 57
5.5.3 Shifting Replacement Policy . . . 58
5.5.4 Tag Array Design Utilizing CSM . . . 59
5.6 Architectural Level Evaluation . . . 60
5.6.1 Experimental Setup . . . 60
5.6.2 The ASE MLC Cache . . . 62
5.6.3 The ASE Cache with CSM . . . 63
5.6.4 Data Migration Scheme Comparison . . . 64
5.6.4.1 Effectiveness of Data Migration . . . 64
5.6.4.2 Performance and Energy . . . 65
5.6.5 Sensitivity Study . . . 66
5.7 Summary . . . 68
6.0 A 4KB STT-RAM TEST CHIP SUPPORTING MULTIPLE TYPES OF MTJS . . . 69
6.1 Motivation . . . 69
6.2 Nano-ring Shaped MTJ (NR-MTJ) . . . 70
6.3 Design of 4K-bit Test Chip . . . 71
6.3.1 Memory Cell Design . . . 72
6.3.2 Write Circuitry . . . 73
6.3.2.1 Write driver . . . 73
6.3.2.2 Lever shifter . . . 75
6.3.3 Read Circuitry . . . 75
6.3.3.1 Adjustable reference cell . . . 75
6.3.3.2 Sense amplifier . . . 76
6.4 Experimental results . . . 77
6.4.1 Functionality Verification . . . 77
6.5 Summary . . . 79

7.0 CONCLUSIONS 81
BIBLIOGRAPHY 82

LIST OF TABLES

1 Comparison among different types of memory [1, 2, 3, 4] 2
2 Cache Access Latency Breakdowns 17
3 Write latency breakdown and read latency/energy of a 8MB STT-RAM L3 cache 20
4 Simulation Parameters 26
5 Worst-Case analysis of the 2RW cell. Transistor width=585nm; $V_{READ} = 0.14V$; $V_{BLN} = 0.50V$. 31
6 Worst-case access patterns for write operations in 1R1W. 36
7 Comparison between 2RW and 1R1W 38
8 The Key Design and Device Parameters 42
9 Margins between the driving current of MLC design with a 4.5F transistor and the required MTJ switching current 45
10 Configuration of CPU, L1 Cache and Main Memory 60
11 Different Configurations of STT-RAM L2 Cache 61
12 Specs of the fabricated STT-MRAM 80

LIST OF FIGURES

1 (a) MTJ in parallel and anti-parallel states; (b) 1T-1J STT-RAM cell. 6

2 (a) MTJ switching time distribution under different currents. The the vertical cursors represent the write failure probabilities at the corresponding write pulse period. (b) and (c) Biasing conditions when writing '0' and '1', respectively. 11

3 (a) BER0 and BER1. (b) Relation between $N_{0\to1}$ and BLER. 12

4 STT-RAM switching current distribution under process variations. 14

5 Average T_{total} varies with τ and $N_{0\to1}$. 16

6 The statistical data of Hamming weight and $N_{0\to1}$. 16

7 WRAP Scheme: (a) design diagram, (b) lookup table. 17

8 VOW scheme: (a) circuit diagram, (b) timeline. 18

9 (a) The design diagram of one time precharge asymmetric sense amplifier; (b) circuit simulation. 20

10 The error rates of Hamming and VOW at T=325K. 21

11 Average write latency comparison at T=325K. 23

12 Dynamic write energy of different schemes 23

13 Pulse period at Temperature Scenario (2) over Scenario(1). 23

14 (a) A typical dual-port SRAM. (b) A 4T-1J dual-port STT-RAM. 26

15 (a) When two cells with in a column is accessed by two ports. (b) Biasing condition for 4T-1J. 27

16 Transistor width vs write-1 current and switching time of (a) 4T-1J STT-RAM; (b) 2RW STT-RAM 28

17 2RW STT-RAM cell. 28

18 Illustration of how the access pattern affect the V_S. 30
19 Ideal and worst-case current of 2RW cell versus: (a) Transistor Width; (b) V_{BLN}; (c) V_{READ}. 32
20 (a) 2RW layout. (b) The directly tiled layout. (c) The optimized layout with shared diffusion. 33
21 Access pattern of 1R1W. 35
22 The minimum W_W and V_{READ} under different W_R. 37
23 (a) MTJ in parallel and anti-parallel states. (b) A single-level cell (SLC) STT-RAM design. 41
24 The serial MLC STT-RAM design. (a) The conventional structure; (b) two-step write operation; (c) two-step read operation. 41
25 Illustrations of switching current change when writing 1 to hard-bit in (a) the conventional MLC, and (b) SR-MLC. 43
26 The hard-bit write current provided in different MLC STT-RAM cell designs. (a) write-1 current; (b) write-0 current. F = Feature Size (32nm) 44
27 Other available MLC cell structures: (a) *Soft-bit reversed* (SR-MLC); (b) *Hard-bit reversed* (HR-MLC); (c) *Soft- and hard-bit reversed* (SHR-MLC). 45
28 The write time comparison among four types of MLC cells for (a) write-1 and (b) write-0 operations. F = Feature Size (32nm) 47
29 The write energy comparison among four types of MLC cells for (a) write-1 and (b) write-0 operations. F = Feature Size (32nm) 47
30 The sense margin when varying I_{read}/I_C, for (a) SR-MLC and SHR-MLC cells and (b) conventional MLC and HR-MLC designs. F = Feature Size (32nm) . 47
31 Miss rate statistic at different sets of a 4-way L2 cache for `h264ref`. 49
32 The set-based ASE scheme. 50
33 (a) The comparison of sense margins in MLC and SLC modes. (b) Sensing delay vs. sense margin. 51
34 The physical to logic mapping in MLC mode with (a) the direct mapping, or (b) the cell split mapping. 53

35 The timing sequence of data swapping execution enabled by a hard-way write opeation (a) or a hard-way read opeation (b). 56
36 Data migration flows. (a) Counter-based migration, and (b) Aggressive migration. 58
37 The shift-like line replacement. 59
38 MLC STT-RAM tag array design utilizing CSM. 59
39 IPC comparison among SLC, Conventional MLC and ASE (normalized to SLC). 63
40 Miss-rate comparison among SLC, Conventional MLC and ASE (normalized to SLC). 63
41 Total dynamic energy consumption among SLC, Conventional MLC and ASE (normalized to SLC). 64
42 IPC comparison after applying data mapping and migration policies (normalized to ASE). 65
43 Total dynamic energy consumption comparison after applying data mapping and migration policies (normalized to ASE). 66
44 The fraction of soft-way hits F_S. 66
45 (a) The average IPC of all 19 benchmarks (normalized to SLC). (b) The energy overhead caused by mode switching, normalized to M_{Th}=2048. 67
46 (a) The average IPC of all the 19 benchmarks, normalized to MLC(DM). (b) L2 Cache dynamic energy, normalized to MLC(DM). 67
47 (a) NR-MTJ structure. (b) NR-MTJ switching. 71
48 4K test STT-MRAM chip organization. 72
49 1T1J STT-MRAM cell. 73
50 Layouts of (a) STT-MRAM cell. (b) NR-MTJ. (b) E-MTJ. 73
51 (a) Bidirectional write driver. Driving abilities of 1.2V device and 2.5V device. 74
52 Schematic of (a) Level shifter. (b) Read circuit. 75
53 (a) Schematic of SA. (b) Layout of SA. 76
54 (a) Die photo and (b) Test chip layout. 77
55 Read signals (a) Readout signals. (b) Read control signals. 78
56 Relations between SA output and adjustable voltage VR and PTUNE. 79

57 Write current with pattern ‘0000’ and ‘0011’. 79

1.0 INTRODUCTION

The continuously increasing demand on system performance in recent years has greatly stimulated the development of Chip-Multiprocessor (CMP). By integrating multiple processing cores into one chip, the system performance can be dramatically increased without boosting the clock frequency, therefore high power efficiency can be acheived . As the number of cores in a CPU keep increasing, the bandwidth gap between CPU and off-chip main memory becomes more severe. As a result, on-chip cache memory that offers high capacity and fast accesses shows of paramount importance to fill this gap and improve the performance. In modern CPU, the area and power consumption of a processor chip is dramatically affected by the on-chip cache memory [5]. For many years, the on-chip cache memory is dominated by static random access memomry (SRAM) because of its high performance. However, as fabrication technology further scales down, SRAM suffers from large leakage power and degraded reliability which severely limits its future application [6].

In recent years, emerging nonvolatile memory technologies have been extensively studied. Examples include spin-transfer torque RAM (STT-RAM) [7][8][9], phase change memory (PCM) [10] and resistive memory (ReRAM) [11]. Because of their non-volatility (data can be kept without power supply), extremely low leakage power can be achieved. Table 1 compares the SRAM with the emerging nonvolatile memory. Among these technologies, STT-RAM is believed to have the greatest potential in developing the next generation on-chip cache memory [12][13][14] thanks to its high performance and good endurance. By storing the data as the relative magnetic direction of magnetic tunnel junction (MTJ), STT-RAM provides high density, fast access speed, zero standby power, as well as hardness to

Table 1: Comparison among different types of memory [1, 2, 3, 4]

	SRAM	STT-RAM	PCM	ReRAM
Nonvolatility	No	Yes	Yes	Yes
Cell Size	$> 100F^2$	$\sim 40F^2$	$8 - 16F^2$	$> 5F^2$
Read Latency	$<$ 10ns	$<$ 10ns	$<$ 48ns	$<$ 10ns
Write Latency	$<$ 10ns	12.5ns	40-150ns	$\sim$10ns
Dynamic energy	low	low	mid	low
Leakage Power	high	low	low	low
Endurance	$> 10^{15}$	10^{15}	10^8	10^5
Maturity	Product	Early Product	Early Product	Test Chip

radiation-injected soft-errors. Therefore, it has attracted much attention from both academia and industry world. In 2012, Everspin debuts first STT-RAM chip for high performance storage systems[15], indicating the commerialization of such memory technology.

1.1 CHALLENGES:

However, we found that there are three major challenges which limit the performance and reliability of STT-RAM based on-chip cache memory:

(1) **Write errors:** Programming STT-RAM is a stochastic process due to random thermal fluctuations. Conventional worst-case (corner) design with a fixed write pulse period cannot completely eliminate the write failures but maintain it at a low level by paying high cost in hardware complexity and system performance. Although Error correction code (ECC) can help reduce the error rates to some extent, it also introduce high latency and energy cost. Therefore, a better solution that is tailored for write errors of STT-RAM is needed.

(2) **Lack of dual-port functionality:** Dual-Port memory design is very common for SRAM based on-chip memory to improve the bandwidth and reduce the conflict. However, the existing STT-RAM cell designs can be used for only single-port accesses, which constraints the system performance. Directly apply the dual-port design method from SRAM to STT-RAM will cause the cell size increase significantly and unacceptable. A STT-RAM based dual-port design method needs to be proposed.

(3) **Poor adaptivity to Multilevel cell design:** The multi-level cell (MLC) design of STT-RAM that stores two or more bits in one cell potentially has higher storage capacity and faster system performance, attracting significant attention. However, the density improvement could be limited because of the large size of access transistor induced by high write current amplitude requirement and asymmetry of switching behavior. Moreover, the read and write accesses of existing MLC STT-RAM cache designs require two-step operation. The system level evaluation shows that the long access latency could amortize the performance speed brought by larger cache size, and even degrade the system performance for some applications. Hence, optimization solution needs to be studied to unleash the potential of MLC STT-RAM cache.

1.2 CONTRIBUTIONS:

To solve these issues that limit the performance of STT-RAM, we have made following contribution in this work.

First of all, we analyzed the root cause of the STT-RAM write errors, then we found that conventional deterministic method such as extending write pulse or using ECC bring too much performance and energy overhead when solving the write errors. Therefore, we have proposed two probabilistic design methods, Write-then-Read with Adaptive Period (WRAP) and Verify-One-while-Writing (VOW) to solve these issues. WRAP uses recursive write-then-verify solution to fully solve the write errors, the pulse width of each write is set dynamically according to data pattern and temperature to achieve the most optimized performance. VOW takes advantage of the asymmetry in STT-RAM write operation (*i.e.,*write-1 has larger error rate) and only verify write-1 operation. Although VOW cannot eliminate the write errors, it can maintain the write error to an extremely low range while achieving very high performance.

Secondly, we proved that in order to support dual-port access with acceptable cost, SRAM-like design method cannot be applied to STT-RAM design for the extremely large area overhead. By leveraging the shared source-line array structure [16][17], we propose a

STT-RAM design solution that supports dual-port accesses by paying a small cell area. In our design, each STT-RAM cell has two BLs and a memory array shares a single grounded SL. To meet the different access requirements of various applications, two types of designs are presented. In a *2RW* STT-RAM cell, both data access ports can support read and write operations. In contrast, *1R/1W* STT-RAM has one read-only port and one write-only port. Separating the read and write accesses reduces the size requirement of access transistor, therefore the even smaller cell area can be achieved. Furthermore, we analyze the reliability of the proposed structures and present the design and layout optimization techniques for density improvement.

Thirdly, we gave detailed analysis about the challenges on applying MLC design into STT-RAM. Accordingly, we introduce the reverse MTJ connection [18][19] that has been successfully utilized in SLC STT-RAM. The new device structure expands MLC cell design to four types, providing different design tradeoffs. Our investigation shows that the cell design with reverse MTJ connection results in the smallest area and continue the density advantage. We also propose an architectural solutions that are adaptive to application's requirement. An *application-aware speed enhancement* (ASE) mode which dynamically trades off the cache capacity and speed according to the behavior of applications is presented. On top of ASE, the *Cell Split Mapping* (CSM) method divides all the cache lines into soft-ways and hard-ways, and makes the soft-ways operated at fast read/write speed to further improve the speed.

Finally, we present a 4Kb test chip design which can support different types and sizes of MTJs. A configurable sensing solution is used in the test chip so that it can support wide range of MTJ resistance. Such test chip design can help to evaluate various type of MTJs in the future.

The remainder of the paper is organized as follows. Chapter 2 introduces the fundamental of STT-RAM and summarize prior arts. Chapter 3 starts with the write error issue of STT-RAM and then discusses the proposed solutions. Chapter 4 discusses the dual-port design techniques, and Chapter 5 analyze the MLC design challenges and the corresponding solutions. Chapter 6 discusses the 4Kb STT-RAM test chip that support different type MTJ.

2.0 PRELIMINARY

2.1 STT-RAM BASICS

The basic storage element in STT-RAM is magnetic tunneling junction (MTJ). Conceptually, an MTJ contains three layers as shown in Figure 1(a): two ferromagnetic layers are respectively named as reference layer and free layer, which are separated by an oxide barrier, *e.g.* MgO. The magnetization direction of the reference layer is fixed, but the magnetization direction of the free layer can be switched through a spin polarized current [7]. For example, a large current injected from the free layer to the reference layer can switch the magnetization direction of the free layer to be parallel to that of the reference layer, and vice versa. When the magnetization directions of the two ferromagnetic layers are *parallel* (P) or *anti-parallel* (AP), the MTJ demonstrates a low- or high-resistance state, representing logic '0' or '1', respectively.

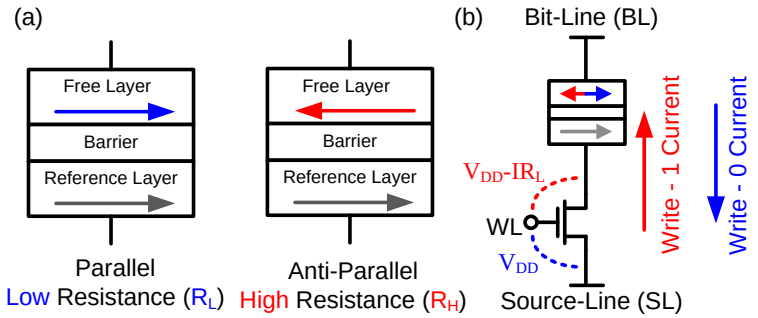

Figure 1: (a) MTJ in parallel and anti-parallel states; (b) 1T-1J STT-RAM cell.

Figure 1(b) illustrates the most popular STT-RAM cell structure consisting of one NMOS transistor and one MTJ (1T-1J) [7][8]. The NMOS transistor, named as the access transistor, connects to the MTJ's reference layer and controls the accessibility of the MTJ. Since there is only one set of WL, BL and SL, this cell structure can only be used for single-port memory design. The MTJ pillar has a very small area so the NMOS transistor determines the area of a STT-RAM cell. In other words, a small transistor is expected for high density. However, the MTJ switching performance strongly relies on the switching current [20]. Reducing transistor size reduces switching current through MTJ and hence degrades the write performance.

2.2 PRIOR ART

Following the progress in fabrication process development, utilizing STT-RAM as on-chip storage has emerged as an attractive topic in embedded system and computer architecture communities [7][8].

There were many circuit-level studies on process variation tolerance and write speed/energy improvement. For example, a corner-aware dynamic gate voltage scheme [21] was proposed to achieve constant current sensing under process variations. And a dual reference voltage sensing scheme [22] was invented to maintain high read yield under process variations while keeping acceptable read speed and energy. Using low threshold voltage device for select transistor has been investigated to improve the write margin [23]. The high-leakage of low threshold voltage devices was reduced by all-digital write driver. Farkhani *et al.* proposed a write-assist technique which applies a negative voltage to the bitline when propgramming logic 1 in order to balance the speeds of write-0 and write-1 operations [24].

Nearly all the previous works on STT-RAM focused on the single-port designs, such as the most popular 1T-1J STT-RAM [7][8]. The cell structure with two transistors (2T-1J)

have been presented by Chung [25]. However, the main motivation was to enhance the writability and array density. The two transistors are controlled by the same WL and hence the design still has only one port.

One major application of STT-RAM technology is on-chip cache so many architectural level solutions have been investigated. Dong, et. al. analyzed the possibility of integrating STT-RAM atop of a single-core microprocessor as on-chip cache to replace the SRAM technology [26]. Sun, et. al. proposed a 3D stacked STT-RAM cache layer on the top of Chip Multiprocessor (CMP) [12]. The slow write speed and high write energy of STT-RAM have been also addressed in many other researches. Zhou, et. al. proposed an early write termination scheme to eliminate the unnecessary writes to STT-RAM cells and save write energy [27]. A SRAM/STT-RAM hybrid cache hierarchy and its enhancements, such as write buffering and data migration were also proposed in [12, 13]. The long read-penalty issue when using STT-RAM as L1 cache were addressed by means of micro-architectural modifications along with code transformation [28]. Li *et al.* proposed retention-relaxed STT-RAM for L1 cache to improve the performance [29]. The data in retention-relaxed STT-RAM requires refresh, the overhead of which was reduced through re-arranging data layout at compile time. SRAM and STT-RAM hybrid cache structure to tradeoff system performance and energy consumption has been widely studied [30].

Using STT-RAM for cache or register file designs in GPU has become a popular research topic recently. For example, a high-retention and low-retention mixed STT-RAM based last-level cache for GPU was proposed with a dynamic data migration scheme [31]. A hybrid register file design combining SRAM and STT-RAM technologies was proposed to leverage the wrap schedule on GPU with a wrap-aware write-back strategy [32]. Moroever, techniques that increases the parallelism of read/write access as well as reduces the number of repeated write access were investigated for better performance and energy of STT-RAM based register file [33].

Since MLC STT-RAM was presented [34][35], it gained a lot of attentions for density improvement. The MLC STT-RAM cache design in [36] utilizes a partially-protected scheme to improve the energy efficiency while achieving target reliability requirement. A rescheduling scheme was used to minimize the waiting time of issued wraps for MLC-based register

bank as presented in [37]. Jiang *et al.* investigated a line-paring method which divides the parallel MLC design into read-fast-write-slow and write-fast-read-slow regions [38]. Previous studies showed that of the two MLC STT-RAM cell structures, the parallel MLC [35] is more sensitive to process variations and has poor reliability. The series MLC structure [34] demonstrates overwhelming benefits in read and write reliability and great potential in commercial usage [39].

3.0 PROBABILISTIC DESIGN METHOD FOR STT-RAM

3.1 MOTIVATION

Unlike SRAM which suffers from radiation-injected soft-error, or DRAM which has low data retention time because of leakage, data stored in the nonvolatile STT-RAM cells can remain valid for a long time, *e.g.*, several years after the write succeeds. During the writing, however, the magnetization switching of a MTJ is a stochastic process influenced by random thermal fluctuations, which causes unpredictable intermittent errors.

The conventional memory design takes into account the worst-case (corner) fabrication and working conditions and utilizes error detection and correction techniques to dynamically address runtime cache operation failures. Such a *deterministic design methodology* is not sufficient in STT-RAM cache designs for several reasons: First, the worst-case guard-banding works effectively only for deterministic failures, *i.e.*, those induced by process variations. However, the write failures induced by the stochastic magnetization switching of MTJs are random and unpredictable. Second, the conventional *error correcting code* (ECC), such as Hamming Code, has limited correction capability due to the short access latency requirement, making it insufficient to protect an STT-RAM cache with a relatively high *bit error rate* (BER). Third, the design philosophy to cover the worst conditions in process and operation leads to overly-pessimistic design associated with high hardware and performance costs. This situation will be further aggravated as technology scales down.

In this chapter, we focused on reducing or even eliminate the inevitable write failures in STT-RAM caches with minimum hardware and performance costs.

As we shall show in Section 3.2.3, writing '1' into an STT-RAM cell is more vulnerable to fail than writing '0' because of the asymmetric MTJ switching property and the unbalanced

biasing conditions of the STT-RAM cell. Consequently, the write failure probability of a memory block (*block error rate*, or BLER) is dependent on its data pattern, say, the number of 1's. The *asymmetric switching property* is even more severe after including process and temperature variations.

We propose two probabilistic design techniques, namely, *Write-then-Read with Adaptive Period* (WRAP) and *Verify-One-while-Writing* (VOW). WRAP is an extension of read-verify-rewrite scheme, which has been adopted in previous works for write error elimination [40][41][42]. A long write pulse period determined by the corner condition usually is applied in read-verify-rewrite scheme. Instead, our proposed WRAP can adaptively adjust the write pulse period according to the Hamming weight of data to maximize performance and energy benefit. In VOW, only one write operation is conducted, which stops after all the 1's of the cache block have been successfully written. A long write pulse period rarely happens because (a) the $0 \rightarrow 1$ flipping bits usually occupy only a small portion of a cache line; and (b) most of writing 1's complete much earlier than the extreme case (or the tail of the MTJ switching time distribution). In VOW, the actual write failures come only from writing 0's and the probability of such failures is extremely low.

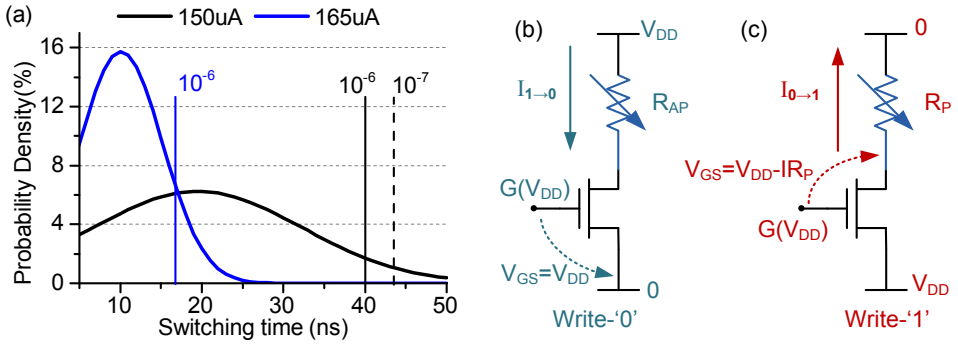

Figure 2: (a) MTJ switching time distribution under different currents. The the vertical cursors represent the write failure probabilities at the corresponding write pulse period. (b) and (c) Biasing conditions when writing '0' and '1', respectively.

3.2 WRITE ERRORS OF STT-RAM

In this chapter, we built a small test circuit using HSPICE at 45nm technology node. The PTM model [43] and the device data of 45nm × 90nm in-plane MTJ [44] were adopted. The NMOS transistor size is W = 360nm/L = 45nm, and the power supply V_{DD} is set to 1.0V.

3.2.1 Stochastic Switching of MTJ

The MTJ magnetization switching is a stochastic process influenced by the random thermal fluctuations. As a result, *the time for an MTJ device to complete magnetization switching is not fixed but changes every time*, even when the operating and environmental conditions remain the same.

Fig. 2(a) shows the distribution of the required switching time for "P→AP" of a 45nm × 90nm in-plane MTJ [44] under different switching current amplitudes. The average switching behavior and the switching variation are obtained by embedding Fokker-Planck equation of the switching time distribution into the LLG stochastic differential equation [20]. It can be observed that extending the write pulse period (*i.e.*, duration of write current) can increase the switching possibility and therefore reduce the write failure probability. Or, increasing the switching current can decrease both mean and variation of the switching time, which also helps to reduce write failures. However, this approach requires a larger NMOS transistor in STT-RAM cell, making memory density lower.

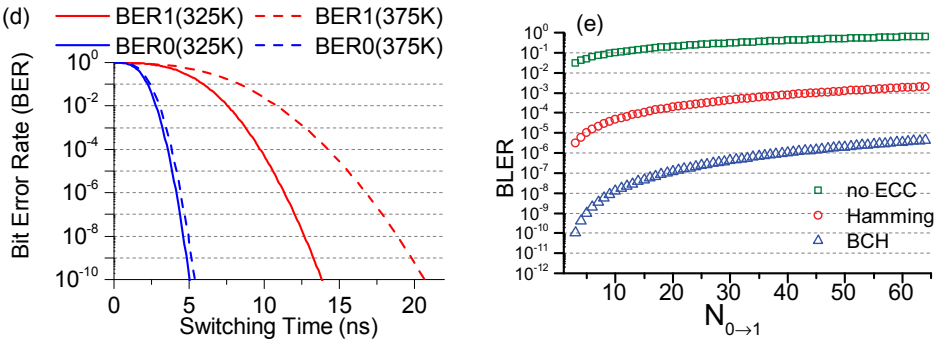

Figure 3: (a) BER0 and BER1. (b) Relation between $N_{0\to1}$ and BLER.

3.2.2 Asymmetric Switching Probability of STT-RAM

Device measurement results [45] showed that "P→AP" switching of a MTJ requires a higher electrical excitation than "AP→P" switching. This means writing '1' into a STT-RAM cell requires a longer write pulse and/or a bigger switching current than writing '0'. Moreover, the biasing conditions of STT-RAM cells during the two types of write operations are unbalanced as illustrated in Fig. 2(b) and (c). When writing '1' into a STT-RAM cell, the voltage drop across MTJ increases the potential at the source of NMOS transistor. The reduced V_{GS} and the body effect degrades the driving ability of NMOS transistor and hence the switching current through MTJ. The different switching time distributions of MTJ and the driving currents in writing '0' and '1' operations lead to ***the asymmetric switching probability of STT-RAM***. We define the probability that a STT-RAM bit fails to switch to '0' or '1' as *bit error rate 0* (**BER0**) or *bit error rate 1* (**BER1**), respectively.

Fig. 3(a) shows the BER0 and BER1 of the STT-RAM design used in this work when varying the write pulse period. The simulation results show that to obtain the same BER0 and BER1, writing '1' requires a longer write pulse than writing '0'. Increasing temperature, *e.g.*, from $325K$ to $375K$, degrades the transistor drivability and therefore the write current amplitude through MTJ. Accordingly, higher BER's are observed under high temperature. Comparably, BER1 is more sensitive to temperature change than BER0. Moreover, *process variations* (PVs) can aggravate the asymmetric switching. Fig. 4(a) shows the STT-RAM write current distribution based on 5,000 Monte-Carlo simulations. The PV parameter was obtained from [44]. PVs have a bigger impact on writing '1' than writing '0', and hence a bigger deviation can be observed. Increasing the temperature from $325K$ to $375K$ further exacerbates the current differences as shown in Fig. 4(b).

3.2.3 Block Error Rate (BLER)

We introduce *block error rate* (**BLER**) to denote the probability that a memory block cannot be corrected and is identified as erroneous after applying error correction technique, *e.g.*, ECC. Both the data pattern and the applied ECC algorithm can affect BLER, as demonstrated by the simulation of a 64-bit cache block in Fig. 3(b)

– Data pattern: Because of the asymmetric switching probability of STT-RAM cells, the more bits that switch from 0 to 1 (denoted as $N_{0\rightarrow1}$), the higher BLER is. Note that during a write operation, unsuccessful MTJ switching happens in only those memory bits that need to be switched, *i.e.*, 0→1 or 1→0. If the original value of an STT-RAM cell equals the new one, no error will be introduced.

– ECC: ECC has been widely utilized to protect the SRAM cells against soft errors. Although STT-RAM cells are not subject to soft errors, we can still leverage ECC to tolerate the intermittent write errors. Compared to Hamming code with single bit error correction, *Bose-Chaudhuri-Hocquenghem* (BCH) cyclic code that corrects 2-bit errors can dramatically reduce BLER. However, the associated hardware and performance overheads are much higher.

3.3 PROBABILISTIC STT-RAM DESIGNS

Applying conventional deterministic (corner) methodology in STT-RAM design to minimize intermittent write errors leads to large hardware and performance overheads. In this work, we propose two probabilistic design techniques, namely, *Write-verify-Rewrite with Adaptive Period* (**WRAP**) and *Verify-One-while-Writing* (**VOW**), to enhance STT-RAM cache performance while eliminating the write errors or maintaining the write errors at a practically negligible level.

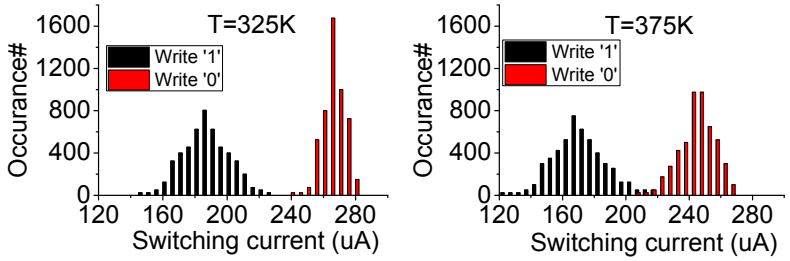

Figure 4: STT-RAM switching current distribution under process variations.

3.3.1 WRAP

Write-verify-rewrite is a straightforward approach to reduce write failures: after each write, the stored data is immediately read out and compared to the input data. If they do not match, a rewrite operation is performed. Such iteration is repeated till a successful write. In theory, this design can eliminate all the write errors in STT-RAM caches with up to infinite iterations. Previously Sun *et al.* presented a similar scheme by simply applying a fixed write pulse period to all the write iterations [42]. However, as we shall show below, each write has an optimal write pulse period τ_{opt} by trading-off the write pulse period and number of iteration. ***According to the write data, WRAP can adaptively employ*** τ_{opt} ***and hence improve the overall system performance.***

3.3.1.1 The optimal write pulse period τ_{opt} In write-verify-rewrite, the total latency of a successful write can be expressed as

$$\begin{aligned} T_{total} &= T_{peripheral} + (T_{write} + T_{verify}) \times N_{iter}, \\ T_{write} &= T_{charge} + \tau, \quad T_{verify} = T_{read} + T_{compare}. \end{aligned} \tag{3.1}$$

Here, N_{iter} is the total number of iterations. The latency of each write T_{write} includes the write pulse period τ and the driver charge latency T_{charge}. The verification overhead T_{verify} comes from reading out the data (T_{read}) and comparing with the input ($T_{compare}$). And $T_{peripheral}$ is the latency from the peripheral circuit such as H-tree routing and decoding.

Increasing τ results in a longer T_{write}, but a smaller N_{iter} due to the reduced BLER. Fig. 5 shows the average T_{total} when varying τ. The optimal write pulse period τ_{opt} inducing the shortest T_{total} exists and is significantly affected by $N_{0\to1}$: as $N_{0\to1}$ increases, τ_{opt} grows to compensate the increased BLER per write.

Tracing $N_{0\to1}$ for each write is costly: it need read out the original data stored in the cache block and compare it to the new data. The induced extra latency is too long to be compensated by the shortened τ_{opt}. We propose using Hamming Weight to estimate $N_{0\to1}$ of a write data in WRAP. Fig. 6 is a statistical analysis on the Hamming weight and the average $N_{0\to1}$ of an 8MB STT-RAM L3 cache for the selected benchmarks. A linear correlation between the Hamming weight and the average $N_{0\to1}$ can be observed.

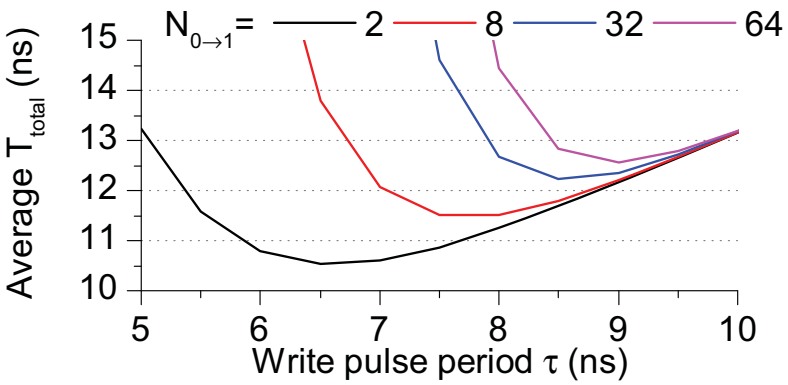

Figure 5: Average T_{total} varies with τ and $N_{0\to1}$.

3.3.1.2 The τ_{opt} configuration in WRAP Simulation results indicate that most writes have relatively small $N_{0\to1}$, say, < 20 for 64-bit cache sub-block. Recall that the change of τ_{opt} is more severe when Hamming weight is small. To reduce design complexity of WRAP, we divide the writing data into several groups based on Hamming weight range, for example, 0, 1, $2 \sim 7$, $8 \sim 31$, and $32 \sim 64$. Each group has one τ_{opt}.

Fig. 7(a) illustrates the design diagram of WRAP. The τ_{opt} configuration is affected by both process variations and temperature fluctuation. The impact of process variations is fixed and can be compensated by adding certain offsets based on post-silicon testing. In contrast, the runtime temperature influence varies dynamically and highly depends on the workload of the running program. On-chip temperature sensors [46] can be used to assist

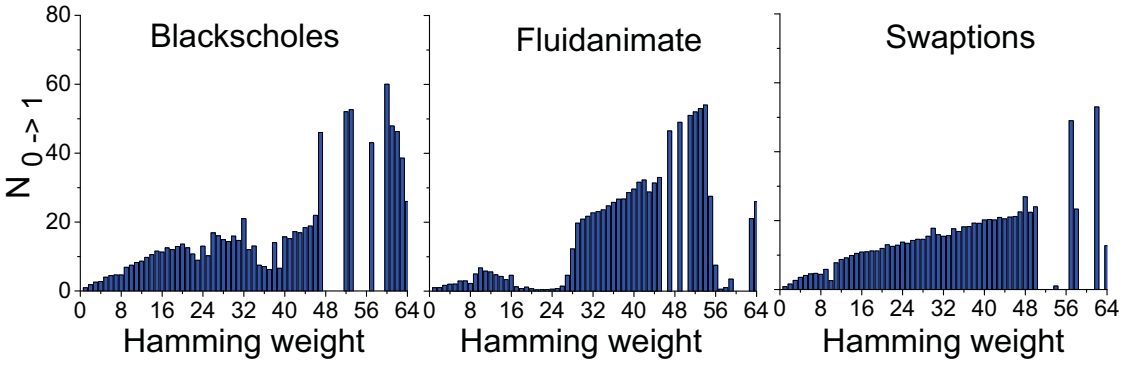

Figure 6: The statistical data of Hamming weight and $N_{0\to1}$.

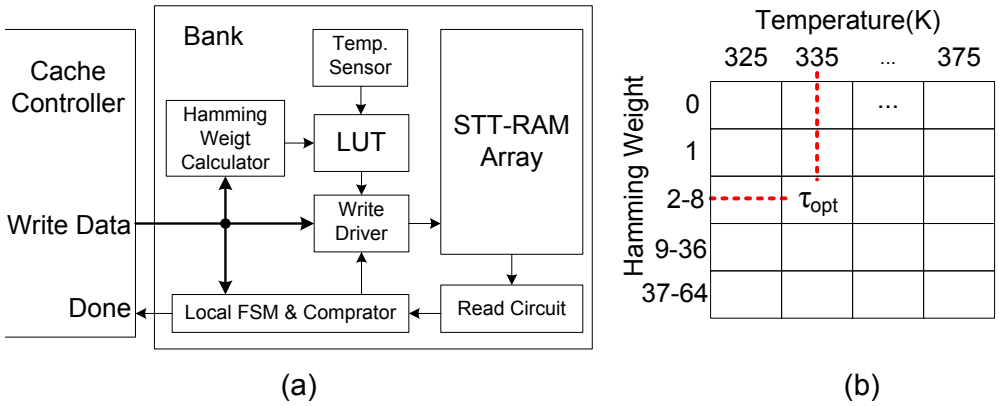

Figure 7: WRAP Scheme: (a) design diagram, (b) lookup table.

τ_{opt} selection. Accordingly, the τ_{opt} configuration of each cache bank is saved in a small 2-D lookup table as shown in Fig. 7(b). The two-dimensional indices are temperature and Hamming weight. Different banks could have different τ_{opt} tables to account for process variations.

3.3.1.3 Overheads of WRAP Different from the other write-verify-rewrite schemes, WRAP utilizes τ_{opt} based on Hamming weight [40][41][42]. Since τ_{opt} is used to terminate the write pulse, a write can start as usual. The latency induced by Hamming weight calculation and look-up table searching is hidden behind the write operation and will not introduce performance overhead.

The performance overhead due to extra data read and comparison in each write iteration is inevitable. Fortunately, the read occurs to the same cache line as the corresponding write.

Table 2: Cache Access Latency Breakdowns

One-Time (ps)	$T_{peripheral}$	H-tree	1101
		Pre-Decoder	163
		Row-Decoder	424
Iterative (ps)	T_{write}	Driver Charge	70
		Write Pulse	Adaptive
	T_{read}	SA Precharge	379
		SA Sensing	803
		Bitline & Mux	17
	$T_{compare}$	Comparator	200

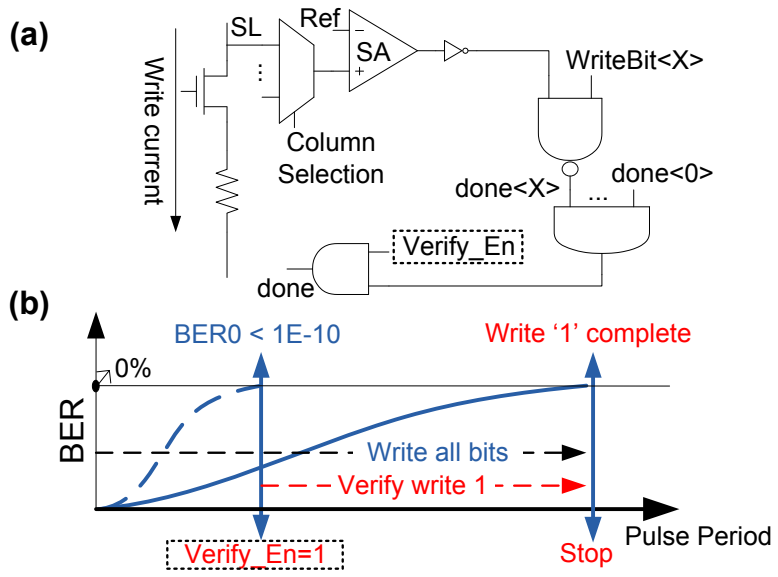

Figure 8: VOW scheme: (a) circuit diagram, (b) timeline.

A large portion of the peripheral circuitry latency in a regular cache read operation is not necessary. Table 2 summarize the cache access latency breakdowns based on NVSim [47]. Only 1.47ns extra delay is required to perform the read and comparison for each iteration.

The chance of rewrite is determined by BLER. Our simulation results in Section 3.3.3 shall show that by properly setting τ_{opt} to control the occurrence of rewrites, rewrite iteration will not degrade system performance much. In hardware implementation, a local *finite state machine* (FSM) is assigned in a cache bank. The FSM contains only three states: idle, write and verify. Our circuit experiment showed that τ_{opt} is always less than 16ns. A 5-bit counter at 2GHz frequency is sufficient to control the pulse period.

3.3.2 VOW

WRAP checks data after each write. Its performance can be further improved if we are able to perform the write and monitor the data in the cache line in parallel and immediately terminate the write pulse once all the bits are successfully written. However, there are two major implementation obstacles: (1) The current direction of '0' and '1' writes are opposite and the STT-RAM cell could be in high or low resistance states. Thus, four possible bitline voltages can be generated. Monitoring all the four possible bitline voltage changes requires

a very complex sensing scheme. (2) A conventional sense amplifier has a precharge before sensing. To monitor data change before a write is completed, we have to keep the loop of "precharge-then-sense", which will results in high dynamic energy overhead. In this work, we propose a *Verify-One-while-Writing* (VOW) scheme to overcome these two obstacles.

3.3.2.1 Design concept From statistical point of view, write-0 usually completes much earlier than write-1. In other words, during the same write pulse period, if all the write-1 bits have been successful, the probability of errant write-0 bits is extremely low. Based on this observation, VOW verifies only the write-1 bits to reduce the sensing complexity while maintaining the overall write failure rate within an acceptable level.

Fig. 8(a) illustrates the circuit diagram of VOW. An asymmetric sense amplifier is used to monitor the 0→1 switching of the STT-RAM cell. For a write-1 bit with WriteBit¡X¿=1, when the MTJ switches to '1', the corresponding done¡X¿ goes to '1'. Once all the write-1 bits finish successfully, the write operation stops. For a write-0 bit with WriteBit¡X¿=0, done¡X¿ is always'1'. Fig. 8(b) shows the timeline of a write operation in VOW scheme. We do not enable the verifying at the beginning in order to preserve a safe pulse period for write-0 bits, as illustrated by Verify_En = 1. In this work, we delay the verification to 5.0ns after initiating write operation to ensure BER0 $\leq 10^{-10}$. Further delaying verification offers negligible improvement on BER0, but performance penalty increases.

3.3.2.2 Asymmetric SA w/ one-time precharge The *sense amplifier* (SA) design in Fig. 9(a) is used to monitor the status of write-1 bits. Benefiting from the asymmetric structure, it requires only one time precharge to keep track of 0→1 switching. Fig. 9(b) shows the HSPICE simulation result. During precharging, OUT is pulled down to low and $\overline{\text{OUT}}$ is pulled up to high. At the beginning of a sensing stage, MTJ may remain at low resistance state. The voltage on IN is lower than that on Ref, and hence, OUT keeps low. After the MTJ switching to high resistance state, the voltage on IN becomes higher than Ref. The strong PMOS and NMOS force OUT and $\overline{\text{OUT}}$ to flip in less than 900ps. The dynamic power consumption of a sensing is about 24fJ. The delay from $\overline{\text{OUT}}$ to the done is less than 96ps.

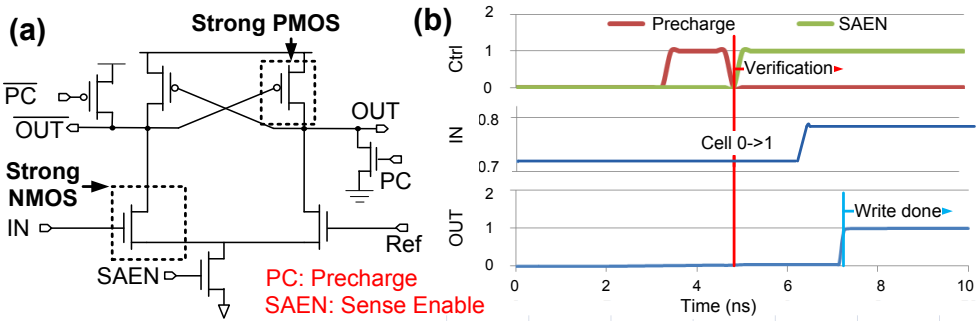

Figure 9: (a) The design diagram of one time precharge asymmetric sense amplifier; (b) circuit simulation.

3.3.3 Evaluation of Proposed scheme

We use NVSim [47] to simulate 8MB STT-RAM with 16 banks and 64-byte cache line under different write schemes. Eight benchmarks from Parsec [48] benchmark suite are selected to represent various data patterns. The baseline is the deterministic STT-RAM cache design with Hamming code, labeled as 'Hamming'. Considering the encoding/decoding complexity, a 64-byte block is divided into eight 64-bit sub-block, and (72, 64) Hamming code is applied. For the entire 64-byte block, the encoding and decoding latencies are 0.7ns and 1.1ns, respectively. The corresponding energy consumptions are 120pJ and 160pJ, respectively. WRAP and VOW scheme are also applied to the 64-bit sub-block level.

Table 3: Write latency breakdown and read latency/energy of a 8MB STT-RAM L3 cache

			Hamming	noECC	WRAP	VOW
Write	Pulse Period(ns)	325K	13.70	13.70	6.92[a,b]	6.47[a]
		375K	23.15	23.15	11.01[a,b]	7.85[a]
	Peripheral (ns)		2.03	1.76		
	ECC (ns)		0.7	0		
	Verify (ns)		0	0	1.54[a,b]	1.00
Read Latency (ns)			5.57	3.99		
Read Energy (nJ)			1.82	1.40		

[a] Average value obtained from simulation.
[b] Re-write probability is taken into account.

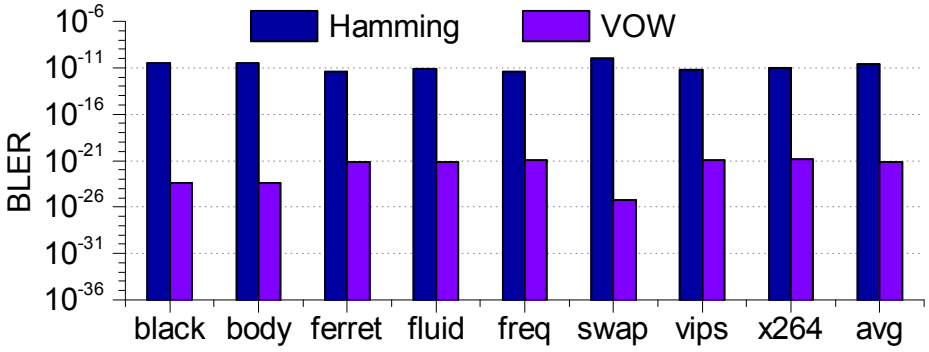

Figure 10: The error rates of Hamming and VOW at T=325K.

To explore the impact of process variations on the write current, the Monte-Carlo simulations described in Section 3.2.2 are performed. For the deterministic designs, we set the write current at -3σ corner. Accordingly, the write pulse period is 13.7ns to obtain BER1=10^{-6}.

To evaluate the write latency of WRAP and VOW schemes, we run Monte-Carlo simulations to estimate the write current distribution and assign write current to sub-arrays. Next, for each sub-array, we generate 100,000 switching times according to the switching time distribution under the given current. For a write access with WRAP in the simulation, we compare writing data to the orignal data and calculate $N_{0\to1}$, then a total $N_{0\to1}$ samples of pre-generated switching time is randomly selected to obtain the switching behavior of this write. If the maximum value among the selected samples is longer than τ_{opt} chosen by WRAP, which indicate some bits are not successfully switched, then a re-write is issued. VOW scheme also randomly selects $N_{0\to1}$ pre-generated switching time samples. Since the VOW only terminated the pulse when all the write-'1's finish, the maximum value of the samples (plus verification delay) is the write pulse period for the write operation.

We also implemented *Recursive Write-Read-Verify* scheme [42] (labeled as 'RWRV') for comparison. RWRV uses the fixed pulse period for each write, while our proposed WRAP can adaptively change the pulse period. The simulation method and overhead calculation of those two schemes are same.

Error Rate: Fig. 10 shows the error rate of 64-bit sub-block obtained in Hamming and VOW. In theory, RWRV and WRAP have BLER=0 by eliminating all the write errors. Note that during the error rate calculation, only the bits that need to be flipped (*i.e.*, $0 \to 1$ or

$1 \rightarrow 0$) are taken in to account. The average error rate of Hamming is $\sim 10^{-11}$, which means that Hamming code with 1-bit correction ability is not strong enough to recover the write errors in the conventional deterministic design. VOW can eliminate all the write-'1' errors. And thanks to the significantly low error rate of write-'0', the error rate of VOW scheme can be as low as 10^{-22} even without any ECC. Interestingly, the error rate of Hamming and VOW have opposite trends. For example, among all the benchmarks, `swaptions` has the highest error rate when using Hamming, but the lowest error rate under VOW. The error rate trend of Hamming follows the distribution of $N_{0\rightarrow1}$ as previously discussed in Section 3.2.3. On the contrary, VOW terminates the write pulse until all the write-1 bits complete. Statistically as $N_{0\rightarrow1}$ increases, its average pulse period prolongs, which in turn reduces the write-0 error.

Write latency and energy: The average write latency of different schemes are compared in Fig. 11. Hamming, as a deterministic design, has a fixed write pulse period determined by the switching time distribution of STT-RAM even though most of the switching processes will finish much earlier. Moreover, the latency overhead from ECC encoding further degrade its write performance. WRAP can dynamically adjust the write pulse according to the data pattern and hence the average write pulse is only $\sim$ 6.92ns, as summarized in Table 3. Even the verify process brings in $\sim$ 1.54ns extra overhead on average, the overall write latency is still about 40% less than Hamming. The rewrites happens only when the previous write fails, which on average causes $\sim$ 5% degradation on write performance. Note that the data in Table 3 and Fig. 11 already took the overhead caused by rewrite into account. Since WRAP doesn't need ECC, we build a 'noECC' scheme (for comparison only) which has the same bank size as WRAP and no ECC delay. As shown in Fig. 11, WRAP can still achieve 34% write latency reduction over 'noECC'. VOW has the best write performance because it terminates write pulse immediately when all the write-1 are finished. The average write latency of VOW is 52% and 47% shorter than Hamming and 'noECC', respectively.

Since our proposed WRAP and VOW have shorter write performance and don't need ECC, they can obtain lower write energy consumption than Hamming. As shown in Fig. 12, an average 26% or 29% write energy reduction can be achieved by WRAP or VOW, respectively. Even compared to noECC, WRAP and VOW can still gain 4% and 7% energy reduction, respectively.

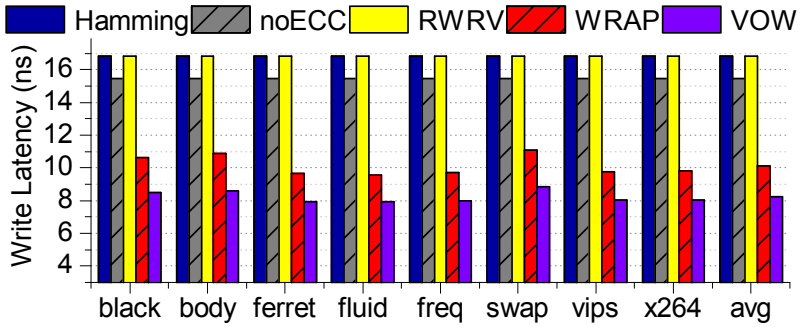

Figure 11: Average write latency comparison at T=325K.

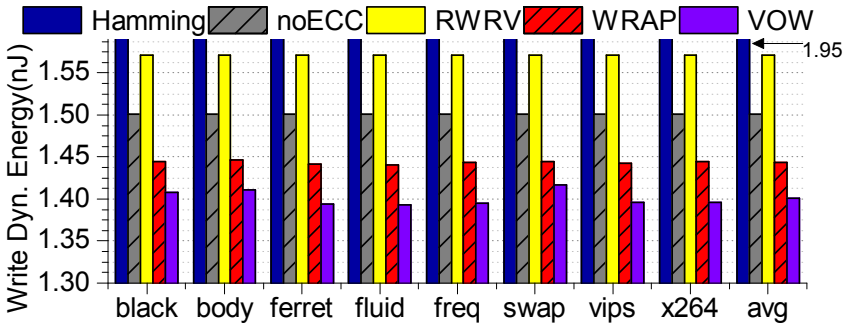

Figure 12: Dynamic write energy of different schemes

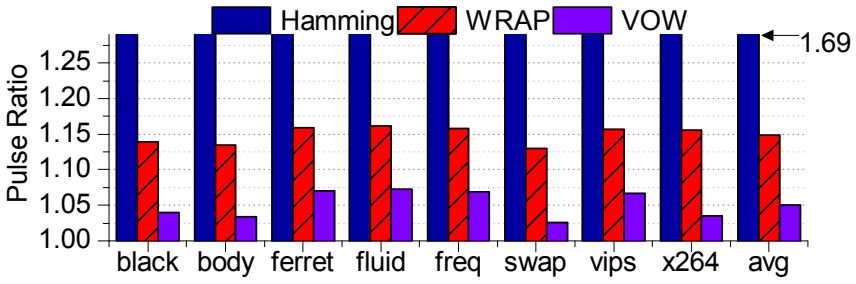

Figure 13: Pulse period at Temperature Scenario (2) over Scenario(1).

Read latency and energy: As shown in Table 3, the read latency of WRAP and VOW is 22% shorter than 'Hamming'. This is because these two probabilistic schemes has smaller bank area since no ECC is needed. For the same reason, the read energy of the WRAP and VOW are 17% less than Hamming.

Temperature Impact: To understand the impact of temperature variation on difference schemes, we conduct simulations under two scenarios: (1) all the 16 banks operate at 325K; (2) two banks work at 375K, four banks operate at 350K, and the temperature of the remaining banks is 325K.

After including temperature fluctuations, a deterministic design need further extend the write pulse to maintain BER requirement as shown in Table 3. This is because the write current drops at high temperature and the MTJ switching process has a larger variation, as previously discussed in Section 3.2. In contrast, WRAP can adaptively reflect the temperature impact on the optimal pulse period with assist of temperature sensors. Longer write pulse period is applied to only the hotspot. VOW, on the other hand, can automatically extend/shorten the pulse at high/low temperature until all the write-1 finish. The ratio of average write pulse width between scenario (1) and (2) is shown in Fig. 13. On average, the higher temperature in scenario (2) results in only 14% and 5% write pulse increments for WRAP and VOW, respectively. For comparison, the write pulse width increment for Hamming is 69%.

3.4 SUMMARY

In this chapter, we first investigate the stochastic switching behavior of MTJ's under the impacts of both process variations and temperature change. By exploiting the asymmetric switching property of STT-RAM cells, two probabilistic design techniques, WRAP and VOW, are proposed to enhance the performance while maintaining a very low write failure probability. The simulation results show that, compared to the conventional fixed pulse scheme protected by Hamming Code, WRAP can ensure zero write error with 40% of write latency reduction and 26% of energy saving. When an average write failure probability as low as 10^{-22} is acceptable, the VOW scheme can further increase the write latency reduction to 52% and energy saving to 29%.

4.0 DUAL-PORT CELL DESIGN FOR STT-RAM

4.1 MOTIVATIONS

With the development of CMP and SOC (System-On-Chip), the large instruction and data exchange among different memory hierarchies makes the memory accesses more and more frequent. Often a memory array receives multiple requests from one or many cores at the same time. The single-port memory which grants access to one request and stalls all the others can lead to significant performance degradation. Therefore, the dual-port or multi-port memory to reduce access conflicts and provide high memory bandwidth becomes a popular approach [49][50][51]. For example, Dual-Port SRAM is used as buffer memory in multimedia applications[52] or a data cache in a multi-core processor[53][54].

However, all the previous STT-RAM designs can support only single-port access [8][25]. For example, the popular one-transistor-one-MTJ cell structure contains only one set of word-line (WL), bit-line (BL), and source-line (SL), which makes dual-port access impossible. Considering the fact that writing to a STT-RAM cell takes longer time than programming a SRAM cell, the stall of the pending accesses of STT-RAM will become even more severe, especially when the port is occupied by write operations. Therefore, the dual-port or multi-port STT-RAM cell design is necessary to enhance the system performance.

In dual-port SRAM designs [49][55], the additional port access is implemented by adding two extra access transistors and one set of WL/BL to the six-transistor cell design. Unfortunately, as we shall show in Section 4.2, the same design method cannot be applied to STT-RAM design for the extremely large area overhead. Therefore, new design techniques for dual-port STT-RAM must be studied.

Table 4: Simulation Parameters

Technology[1]	65nm
VDD	1.2V
MTJ geometry	65nm × 130nm
R_P/R_{AP}	1.88/3.77kΩ
AP→P Switching Current[2]	112μA
P→AP Switching Current[2]	142μA

[1] The minimun channel length is 60nm.
[2] At 10ns switching time.

In this chapter, we use 65nm CMOS technology [56] with a 65nm × 130nm in-plane MTJ model calibrated against the experimental data [25]. The switching behavior of the MTJ is modeled based on the Landau-Lifshitz-Gilbert equation [20]. The detailed parameters are listed in Table 4.

4.2 DUAL-PORT STT-RAM DESIGN CHALLENGES

Figure 14(a) illustrates a typical SRAM design with two sets of read/write ports [49][55]. Compared to a single-port SRAM cell with six transistors, two more transistors (M1 and M2) associated with the wordline control (WLB) and the data access connections (BLB and $\overline{\text{BLB}}$) of the second port, are inserted.

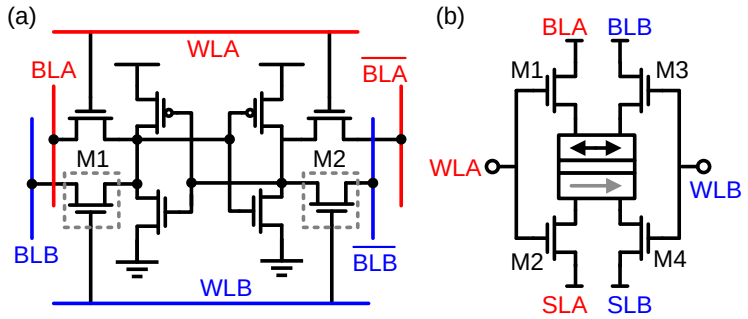

Figure 14: (a) A typical dual-port SRAM. (b) A 4T-1J dual-port STT-RAM.

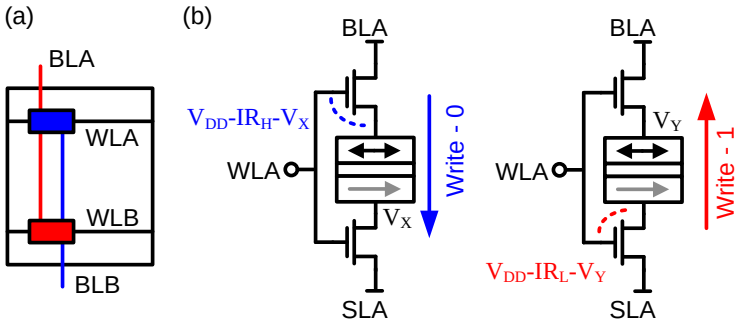

Figure 15: (a) When two cells with in a column is accessed by two ports. (b) Biasing condition for 4T-1J.

By following the same design concept, Figure 14(b) shows a dual-port STT-RAM cell with four transistors and one MTJ (4T-1J). Here, a duplicate pair of BL and SL provide the access through the second port. Compared to the single-port STT-RAM cell in Figure 1(b), three additional transistors (M1, M3, and M4) are needed for access control. Note that in a STT-RAM array, the BL and SL are usually shared by entire column. For single-port cells, only one memory cell within a column can be activated at a time. Therefore, one transistor at SL terminal is sufficient to control the accessibility to one cell per column. In contrast, a dual-port array may simultaneously access two cells within one column through Port-A and Port-B, respectively, as illustrated in Figure 15(a). Determined by the operation type and data pattern, the two concurrent accesses could have the different BL voltages. Thus, M1 and M3 are necessary to isolate BLA and BLB from each other.

Due to degraded biasing condition, the 4T-1J dual-port STT-RAM cell is functionally correct by paying significant area overhead compared to the 1T-1J single-port design. As shown in Figure 1(b), a conventional 1T-1J STT-RAM encounters V_{GS} degradation induced by the voltage drop on MTJ only in write-1 operations, which constrains the switching current through MTJ. The 4T-1J dual-port STT-RAM has a symmetric cell structure: along an access path, *e.g.*, from BLA to SLA, two transistors M1 and M2 are turned on and connected side by side of the MTJ. No matter in write-1 or write-0 operations, one of them suffers from V_{GS} degradation, as shown in Figure 15(b). In other words, the biasing condition

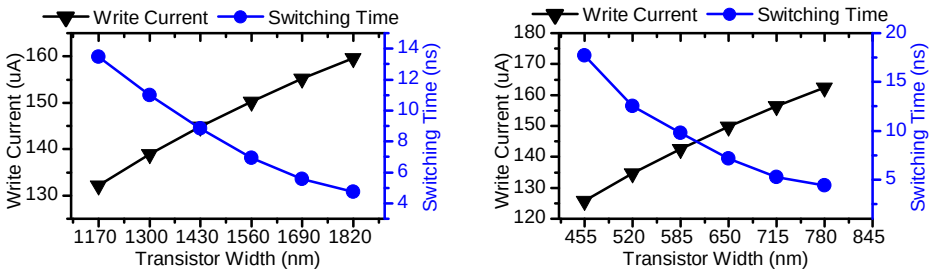

Figure 16: Transistor width vs write-1 current and switching time of (a) 4T-1J STT-RAM; (b) 2RW STT-RAM

of the access transistors in the 4T-1J cell is much worse than that of a 1T-1J design. We have to enlarge all the access transistors to provide sufficient MTJ switching current in write operations.

Figure 16(a) shows the relation between the write-1 current and the size of the access transistors in the 4T-1J design, assuming all the four transistors are of the same size. Here, the write-1 operation dominates the transistor size selection because of the asymmetric P $\rightarrow$ AP and AP $\rightarrow$ P switching currents of the MTJ device used in this work. To obtain the write time of 10ns, the access transistors' width is approximately 1400nm. Integrating four such large transistors into one memory cell leads to a cell area of 575F^2, which is even bigger than that of the dual-port SRAM design (*e.g.*, 233F^2 reported in [55]). It is not acceptable to adopt such a large STT-RAM design for on-chip applications.

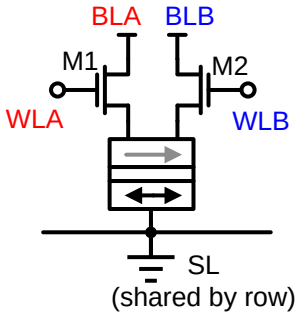

Figure 17: 2RW STT-RAM cell.

4.3 STT-RAM DESIGN WITH TWO READ/WRITE PORTS

4.3.1 Design Concept

Previously, the shared SL for single-port STT-RAM array has been proposed by Zhao *et. al.* to increase array density [16]. It also has been used to balance the write-0 and write-1 performance [17]. The basic design concept is that all the cells on the same row share the same SL, then all the SLs are connected together and grounded (GND/0V).

In this work, we propose to reduce the cell area of dual-port STT-RAM design by utilizing the shared SL structure. Figure 17 depicts the STT-RAM design with two read and write ports (2RW). Please note in [16], the grounded (0V) SL is connected to the transistor, but in the proposed 2RW design, the grounded SL is connected to the MTJ in order to support Dual-Port. The write-1 operation requires a switching current from SL (GND) to BL, so a negative voltage (V_{BLN}) need be applied to BL. Such V_{BLN} can be generated using level converter[57].

The 2RW cell design can significantly reduce the cell area compare to 4T-1J. First, since the SL is always connected to GND, isolating SLs of different memory cells is no longer necessary. The transistors used for SL access control in STT-RAM cell can be removed. Only two transistors M1 and M2 remain to enable/disable the access to Port-A and Port-B, respectively. Thus, the number of transistors reduces to half of the 4T-1J dual-port design. Second, the width of access transistors can greatly decrease because only one transistor exists along the current path between BL and SL. Figure 16(b) shows the relation between the write-1 current and the size of the access transistors in the proposed 2RW STT-RAM design. The required transistor width to achieve the 10ns write time is 585nm, which is only $\sim 42\%$ of the access transistor size of the 4T-1J STT-RAM cell. The area of a proposed 2RW cell is approximately 21% of the 4T-1J design.

4.3.2 Reliability Analysis

The voltage of the shared SL (V_S) in the single-port STT-RAM array may not be ideal 0V due to the existence of the parasitic resistance (R_S) [16]. Figure 18 illustrates the scenario.

When turning on WL and applying a certain voltage to BL, the variation on V_S exists and induces degradation on both read and write performance. For example, if V_S is higher than ideal 0V, the actual voltage drop across the BL and SL reduces. Consequently, the write-1 current becomes lower than the projected value obtained under the ideal condition. In read operations, a higher/lower V_S can decrease/increase the read-0/read-1 current. The reduced difference between read-0 and read-1 currents could result in more read errors. V_S variation can also leads to higher possibility of read disturbance, *i.e.*, unwanted '0'→'1' switch when reading a cell which stored '0' [8]. A negative V_S will increase the read-0 current (I_{R0}) and bring it closer to the P → AP switching current (I_{W1}).

For the proposed 2RW STT-RAM design, the impact of the V_S variation becomes even more severe. First, the V_S variation increases as the number of cells being accessed grows. When both ports access the cells on the same row as illustrated in Figure 18, the number of cells doubles compared to that of single-port STT-RAM array. So a larger V_S variation is expected. Moreover, we notice that in the single-port STT-RAM, the read operations have a lower V_S variation than the write operation. This is because the write requires a bigger voltage amplitude applied to BL ($|V_B|$) and the only port can perform either write or read access. However, for the 2RW STT-RAM design, it is possible that the read and write are conducted simultaneously through the two sets of ports. The interaction in between

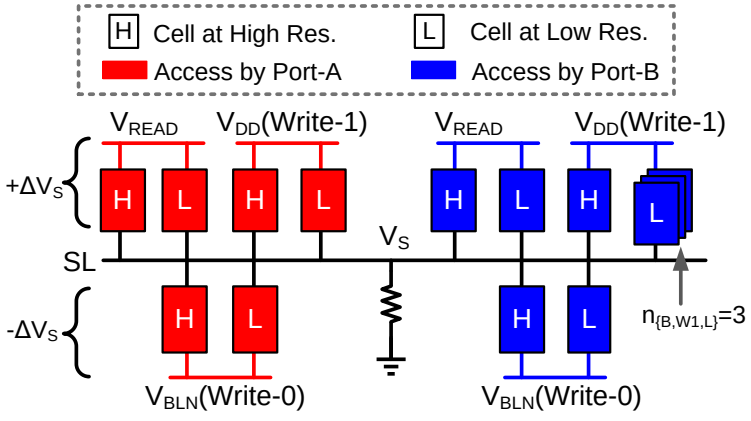

Figure 18: Illustration of how the access pattern affect the V_S.

Table 5: Worst-Case analysis of the 2RW cell. Transistor width=585nm; $V_{READ} = 0.14V$; $V_{BLN} = 0.50V$.

		Dual-Port Access		
	Ideal Current[1]	**Worst-Case Pattern**[2]	**Worst Current**	**Worst Vs**
Write-1	142μA	n{A,W1,L}=8,n{B,W1,L}=8	130μA	57.5mV
Write-0	112μA	n{A,W0,H}=1, n{A,W0,L}=7, n{B,W0,L}=8	96μA	−71.4mV
Read-1	29.0μA	n{A,R,H}=8, n{B,W0,L}=8	35.7μA	−32.1mV
Read-0	47.6μA	n{A,R,L}=8, n{B,W1,L}=8	34.9μA	37.7mV
I_{W1}-I_{R0}	94.4μA	n{A,R,L}=1, n{A,R,H}=7, n{B, W0, L}=8	83.7μA	−31.5mV

[1] $V_S = 0V$ for Ideal case.
[2] Unlisted n indicates the corresponding value is 0.

degrades the V_S variation of read operations. Third, the value of V_S is also affected by the MTJ resistance states of the cells being accessed. When the MTJ is at high resistance state, the V_S is more reluctant to be disturbed by V_B.

Here, we use $n_{\{A/B,R/W1/W0,H/L\}}$ to represent the number of the cells under certain access pattern. The subscript A/B indicates Port-A or Port-B access. $R/W1/W0$ describes the operation modes, including read, write-1, or write-0. H/L represents the high or low resistance states of MTJ. For example, $n_{\{B,W1,L\}}$ is the number of the cells that are with low MTJ resistance and conducting write-1 operations through Port-B.

Without loss of generality, we studied the current through a 2RW STT-RAM cell when it is accessed through Port-A. Table 5 summarizes its worst-case current and the corresponding access patterns in read and write operations. In the experiment, we assume a SL is shared by 32 columns, and each port accesses only 8 cells by using column selection, which is very common to support set-associative cache. The R_S of such setup is set to 27.5Ω according to [16]. The worst-case scenario happens when all the 16 cells being accessed fall on the same row. For comparison purpose, the currents under the ideal condition when V_S is exactly 0V are also presented.

The simulation results show that write-1 and write-0 currents drop from 142μA and 112μA projected under the ideal condition to 130μA and 96μA in the worst scenario, respec-

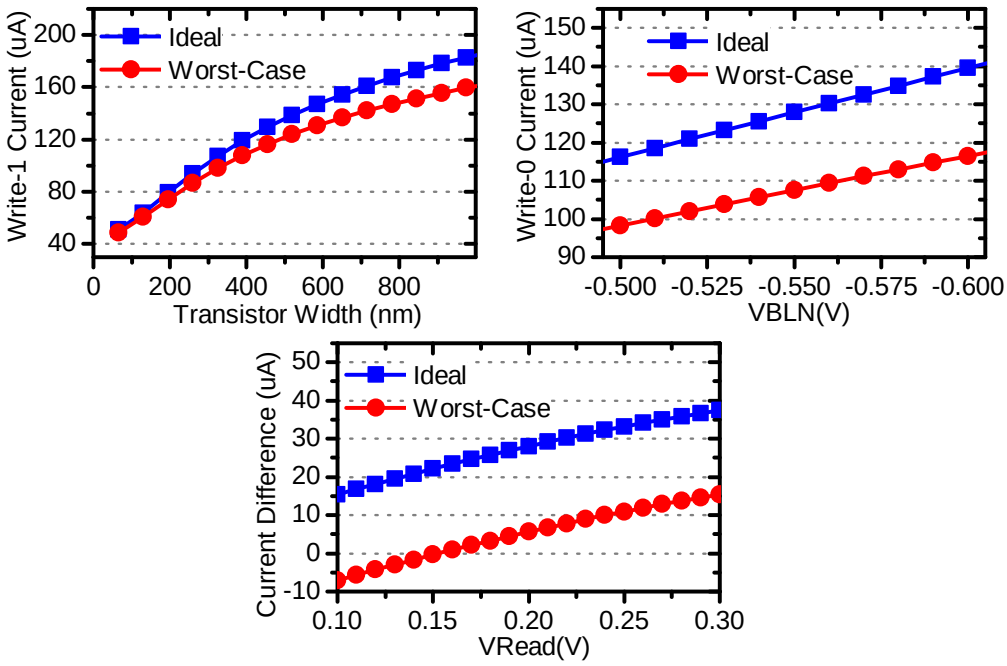

Figure 19: Ideal and worst-case current of 2RW cell versus: (a) Transistor Width; (b) V_{BLN}; (c) V_{READ}.

tively. The write current degradation means the design cannot meet the target of a 10ns switching time. The situation for read operation is even worse: ideally the read-0 current is 18.6μA more than the read-1 current. However, in the worst-case combination, the read-0 current turns to be less than that the read-1 current, which can result in read decision errors. One possibly way to solve this is to increase the read voltage (Section 4.3.3). The margin between read-0 current and P $\rightarrow$ AP switching current ($I_{W1} - I_{R0}$) reduces from 94.4μA to 83.7μA under worst-case, which indicate higher possibility of read disturbance. Please note the "worst-case" for the $I_{W1} - I_{R0}$ occurs when I_{R0} reach its highest value.

In Table 5, we also show the results when disabling Port-B, which is indeed equivalent to single-port access. The results show that the second set of access ports results in 8μA degradation on both write-1 and write-0 currents in the worst-case condition. The difference between read-0 and read-1 currents dramatically drops 17.3μA due to the interaction between read and write in dual-port accesses.

4.3.3 The Cell Configuration and The Operating Setup

Previously we demonstrate that the variation of V_S is exaggerated by the dual-port access, which must be considered when determining the access transistor size in cell design and setting up the operating conditions, *i.e.*, the read and write voltages.

For the given MTJ device in Table 4, the write-1 operation is critical in transistor size selection. To compensate the current degradation under the worst-case access pattern, we have further increase the transistor width. The simulation result in Figure 19(a) shows that to maintain the write-1 current at 142μA in the worst-case condition, the access transistor grows to 715nm in width.

The negative voltage (V_{BLN}) for write-0 operations also needs to be adjusted to compensate the impact of V_S variation. With the access transistor width of 715nm, Figure 19(b) shows that $|V_{BLN}|$ should increase to 0.58V to obtain the 112μA write-0 current in the worst-case condition. Figure 19(c) demonstrates the relation between the read voltage (V_{Read}) and the current difference in read-1 and read-0 operations. The negative value of current difference indicates that the read-0 produces a smaller current than the read-1, which will result in inevitable read decision error. Increasing V_{Read} can significantly improves the read current difference. On the other hand, the higher read-0 current can increase the chance of read disturbance.

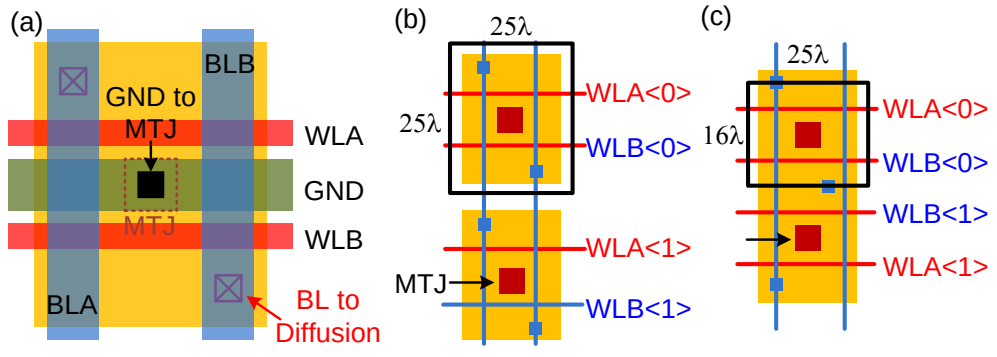

Figure 20: (a) 2RW layout. (b) The directly tiled layout. (c) The optimized layout with shared diffusion.

4.3.4 Layout Design

Figure 20(a) shows the layout of the proposed 2RW STT-RAM cell, where λ is half of the feature size (F). Based on the analysis in the previous section, the access transistor width is 715nm (11F). The two access transistors in one cell can share the diffusion area, which is connected to the MTJ.

Figure 20(b) shows that when directly tiling up the cells on a column, the diffusion area of two adjacent cells cannot be shared. Because WLA and WLB are driven from two separate decoders, WLB $< 0 >$ and WLA $< 1 >$ could be turned on at the same time. In such operation situation, sharing the diffusion area can results in current flowing through the two MTJs, which is not allowable. In contrast, we can safely share the diffusion area by vertically flipping the bottom cell as shown in Figure 20(c). The shared diffusion is controlled by WLB $< 0 >$ and WLB $< 1 >$. They are driven by the same decoder and won't be turned on simultaneously. As a result, the height of a memory cell greatly reduces from 25λ to 16λ.

The area of the optimized cell in Figure 20(c) is $100F^2$, which is about 42% of the area of a 2RW SRAM design ($233F^2$) reported in [55]. Comparing to the single-port 1T-1J cell which obtains same write performance with our MTJ parameter ($72F^2$) [58], the area overhead of introducing an additional port is about 39%.

4.4 STT-RAM DESIGN WITH 1-READ/1-WRITE PORT

4.4.1 Design Concept

Some dual-port SRAM designs restrict the port functionality [50][51]: one support read operations only and the other is for writes only. Such designs with 1-read/1-write port (1R1W) can alleviate the degradation of static noise margin, compared to 2RW design.

Similarly, the 1R1W design concept can be applied to the dual-port STT-RAM to reduce the impact of V_S variation. Figure 21 illustrates the access pattern when constraining the port functionality to 1R1W. Not like writes through two port aggravate the V_S variation in 2RW STT-RAM, a write in 1RW design can be accompanied to only a read through

the other port. Since the read voltage is much lower than the write voltage, V_S reduces compared to the 2RW case. Moreover, the positive V_{READ} tends to bring V_S to the positive direction, which actually improves the write-0 current strength. Therefore, the worst-case access patterns for the write operations in 1R1W STT-RAM is redefined as shown in Table 6. The patterns for read operations remain the same as the 2RW design in Table 5.

4.4.2 Transistor Sizing and Operating Voltage

Benefiting from the improved worst-case write current, the 1R1W design can shrink the transistor sizes to achieve the same write performance as the 2RW design. For example, if assuming the two access transistors are of the same dimension and setting V_{Read} to $0.24V$, our simulation shows that the transistors can reduce to 670nm.

Moreover, if utilizing the different sizes to the read access transistor (W_R) and the write access transistor (W_W), the design could be further reduced. On one hand, the increased resistance induced by a smaller W_R helps reduce the V_S variation, which in turn alleviates the sizing requirement for W_W. On the other hand, the smaller read access transistor degrades the read current difference I_{Rdiff}, which could lead to more read errors. To maintain I_{Rdiff} when decreasing transistor sizes, we can increase V_{READ}, which however exaggerates the V_S variation.

For a given W_R, we proposed the following design flow to obtain the minimum W_W and hence the most area-efficient configuration:

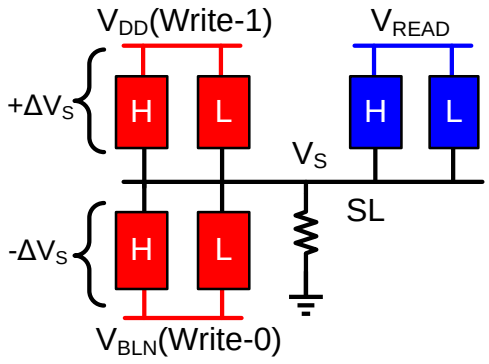

Figure 21: Access pattern of 1R1W.

Table 6: Worst-case access patterns for write operations in 1R1W.

	Worst-Case Pattern
Write-1	n{A,W1,L}=8, n{B,R,L}=8
Write-0	n{A,W0,H}=1, n{A,W0,L}=7, Port-B idle.

[1] Assuming Port-A is write only and Port-B is read only.

Step 0: Randomly choose a V_{READ}.

Step 1: With the given W_R and V_{READ}, sweep W_W till it meets the write-1 current target in the worst-case pattern.

Step 2: Find the V_{BLN} to achieve the write-0 current target under the worst-case configuration, when W_R, V_{READ} and W_W are fixed.

Step 3: Get the V_{Read} to achieve the I_{Rdiff} target for the given W_W, V_{BLN} and W_R.

Step 4: Repeat the iteration from Step 1 to Step 3 until W_W and V_{READ} converge to certain values.

Figure 22 shows the minimum W_W and the corresponding V_{READ} under different W_R. Here, we set the targeted I_{Rdiff} as 10μA and the write time as 10ns for both write-0 and write-1. The result shows that reducing W_R from 660nm to 540nm helps relax the sizing requirement of W_W due to the increased equivalent resistance of read access transistor. However, W_W starts to increase when further decreasing W_R because the the higher V_{READ} becomes the dominating factor. As the width of the cell layout is determined by W_W, the smallest 1R1W STT-RAM cell can be obtained when $W_W = 660$nm and $W_R = 540$nm. The corresponding V_{READ} and V_{BLN} are 0.27V and -0.53V, respectively.

4.4.3 Comparison of 2RW and 1R1W STT-RAM Designs

We compared the proposed 2RW and 1R1W STT-RAM designs by following the worst-case design methodology and the results are summarized in Table 7. Thanks to the smaller transistors, the cell area of a 1R1W STT-RAM cell is only 92.3% of that of the 2RW design.

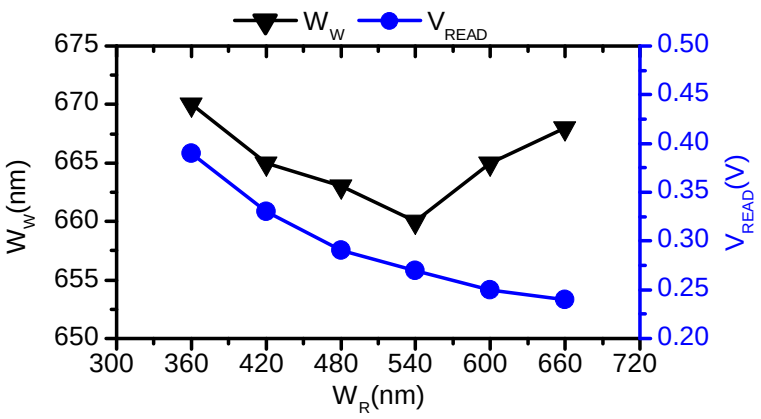

Figure 22: The minimum W_W and V_{READ} under different W_R.

The amplitude of V_{BLN} is smaller too. The reduced transistors and $|V_{BLN}|$ indicates that the 1R1W design has the less write current in the non-worst-case condition and hence consumes less write energy than the 2RW design. Interestingly, although the V_{READ} is higher for 1R1W, the worst-case difference between read-'0' current and P $\rightarrow$ AP switching current ($I_{W1} - I_{R0}$) is still improved, which indicates lower possibility of read disturbance. This is because smaller W_W and V_{BLN} reduce the V_S drift toward negative direction, which is the main reason for the excessive I_{R0}. In summary, 1R1W cell can achieve smaller area, less energy waste and smaller possibility of read disturbance, with the cost of restricted port functionality.

4.5 SUMMARY

In this chapter, we firstly propose the dual-port STT-RAM design, which can provide higher data bandwidth. We found that the dual-port design based on the conventional 1T-1J STT-RAM is not feasible because of the large cell area. Accordingly, we propose to leverage the shared SL design to simplify the cell structure and reduce the memory cell area. Two types

Table 7: Comparison between 2RW and 1R1W

	2RW	**1R1W**
Worst-case write time	10ns	
Worst-case I_{Rdiff}	10μA	
Transistor Width	both 715nm	$W_W = 660$nm $W_R = 540$nm
Cell Size	$100F^2$	$92.3F^2$
Area overhead over Single-Port STT	39%	28%
V_{READ}	0.24V	0.27V
I_{VBLN}	-0.58V	-0.53V
I_{W1}-I_{R0}	77.8μA	82.4μA

of the dual-port STT-RAM design, 2RW and 1R1W, are presented. Furthermore, the related design issues, including reliability, cell configuration, operating setup, and layout techniques, have been considered and discussed.

5.0 MLC STT-RAM DESIGNS

5.1 MOTIVATION

Compared to single-level cell (SLC) design, multilevel cell (MLC) that stores two or even more bits in one memory cell is more efficient in data storage density. The MLC design has been successfully adopted in Flash memory and PCM technologies by dividing the threshold voltage of Flash and the resistance range of PCM cell into multiple levels, respectively [59][60]. The use of MLC in STT-RAM cache design has also been investigated. For example, Chen *et al.* examined the read/write scheme and proposed a set remapping solution to extend its life time [61]. Zhang *et al.* compared series and parallel MLC STT-RAM designs, concluding that series MLC STT-RAM is more resilient to process variations [39]. Jiang *et al.* addressed the performance issue through line paring and line swapping methods particularly for parallel MLC STT-RAM design [38]. Nevertheless, a number of circuital and architectural challenges remain unsolved in MLC design, including the limited density benefit and the degraded performance induced by multi-step accesses.

An SLC STT-RAM cell is composed of an MTJ for data storage and an NMOS transistor for access control. Its area is mainly determined by the transistor size, the selection of which shall take many factors into consideration, including the MTJ resistance, the MTJ switching current requirement, the biasing condition of the transistor, *etc.* Unlike MLC PCM which obtains multiple logic bits by partitioning resistance range without changing the cell structure, MLC STT-RAM design need insert an extra MTJ pillar to represent the second logic bit. The change in cell structure greatly complicates the design trade-off and makes the use of the minimal-sized selective transistor very difficult. In fact, our evaluation shows that the conventional MLC structure [61][39][34] even is in danger of losing the density

competition to SLC design. We note that the reverse MTJ stacking has been successfully utilized in SLC STT-RAM [18][19]. In this work, we explore its use in MLC design. The new device structure expands the design space of MLC STT-RAM. Our simulations show that the new cell structure made of reverse MTJ connection can achieve the smallest area and continue the density advantage.

Besides the storage capacity, the access speed is another key metric in cache design. By nature, accessing an MLC design is slower than SLC, simply because its logic detection in a read operation requires two sensing stages and writing an MLC cell involves two-step programming. At the system level, the enlarged storage capacity and prolonged access latency of MLC STT-RAM have contradictory impacts on the overall system performance. The winner is determined by application's requirement. Those with large datasets benefit from the high cache capacity that reduces cache miss rate and costly accesses to main memory. In contrast, applications with small data sets may suffer from the long read and write latencies, performing even worse than the system integrated with SLC STT-RAM cache.

We observed that an MLC STT-RAM cache can support the SLC operation mode, which provides fast accesses but sacrifices half of its storage capacity. Based on it, an architectural level solution, named as application-aware speed enhancement (ASE), was proposed: according to application's memory access behavior, the MLC STT-RAM cache dynamically changes between the MLC mode with high capacity and the SLC mode that offers low access latency. Furthermore, we presented a cell split mapping (CSM) method, which divides a cache line into a fast and a low regions to reduce the mode switching cost. To fully take advantage of the proposed architecture solutions, new data migration policies that allocate frequently used data to fast regions were also studied.

5.2 FUNDAMENTALS OF STT-RAM

MLC STT-RAM is developed by integrating two MTJs into one single cell. For example, *parallel MLC STT-RAM* divides the free layer of an MTJ into a hard domain and a soft

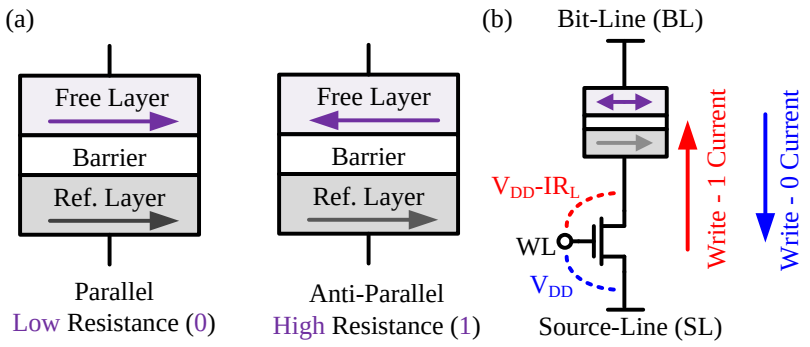

Figure 23: (a) MTJ in parallel and anti-parallel states. (b) A single-level cell (SLC) STT-RAM design.

domain to represent two logic bits [35]. This design demonstrates poor reliability due to its high sensitivity to process variations [39]. Instead, *series MLC STT-RAM* that stacks two MTJs in series is more feasible and has been widely accepted [34]. Its cell structure is illustrated in Fig. 24(a).

No matter in a parallel or serial MLC cell, the two MTJ pillars representing different logic bits have different areas. As shown in Fig. 24(a), we name the data stored in the small and big MTJs as *soft-bit* and *hard-bit*, respectively. Because both the resistance-area product (RA) and critical switching current density (J_C) remain constant in a given magnetic process, the soft-bit has a larger resistance value but requires a smaller switching current I_C than the hard-bit.

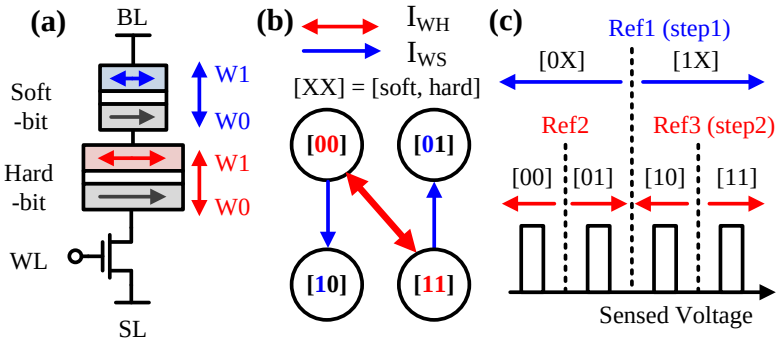

Figure 24: The serial MLC STT-RAM design. (a) The conventional structure; (b) two-step write operation; (c) two-step read operation.

Table 8: The Key Design and Device Parameters

RA ($\Omega \cdot \mu m^2$)	4.88	$J_{C,1\to0}$ (MA/cm^2)	2.13
Soft-bit area (nm^2)	32×64	$J_{C,0\to1}$ (MA/cm^2)	3.20
Hard-bit area (nm^2)	45×90	$I_{C,Soft,1\to0}$ (μA)	34.31
TMR	105%	$I_{C,Soft,0\to1}$ (μA)	51.47
V_{DD}(V)	1.2	$I_{C,Hard,1\to0}$ (μA)	67.86
Feature Size (nm)	32	$I_{C,Hard,0\to1}$ (μA)	101.80

Fig. 24(b) summarizes the write procedure of an MLC STT-RAM. Programming an MLC cell needs two stages. First, apply a current larger than the hard-bit critical current (*i.e.*, $I_{WH} > I_{C,Hard}$), which inevitably switches both the hard-bit and soft-bit. Then a smaller current that satisfies $I_{C,Soft} < I_{WS} < I_{C,Hard}$ is deployed to switch only the soft-bit. Reading data from an MLC STT-RAM requires two sensing steps too: first detect the soft-bit; then according to the value of the soft-bit, apply another reference voltage to detect the hard-bit data. The procedure is shown in Fig. 24(c).

In this work, we adopted 32nm PTM CMOS model [43] and the MTJ parameters from [62] for circuit analysis. The area ratio of the two MTJs is set to 2 in order to balance the difference of adjacent resistance states [39]. The key design and device parameters are summarized in Table 8.

5.3 MLC STT-RAM CELL DESIGN EXPLORATION

5.3.1 Design Challenges of Conventional MLC STT-RAM

Higher density is the major motivation to promote MLC design. In STT-RAM, the MTJ pillar is realized at the minimal allowable dimension to reduce the switching current requirement. Hence, the cell area is mainly determined by the selective transistor. On the one hand, a small transistor is preferred to improve data storage density. On the other hand, the

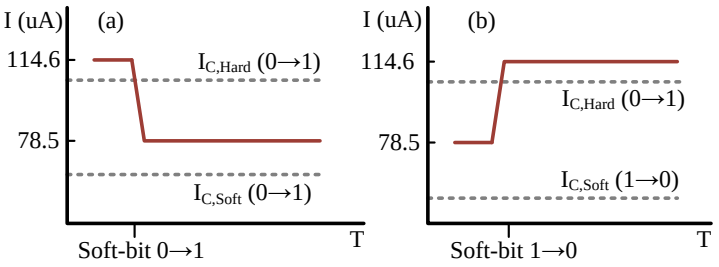

Figure 25: Illustrations of switching current change when writing 1 to hard-bit in (a) the conventional MLC, and (b) SR-MLC.

transistor must be large enough to provide sufficient current to switch MTJ during programming. In an MLC STT-RAM cell, an extra MTJ is introduced to stand for the second logic bit. The structural modification, however, exacerbates the size requirement of the selective transistor for the following two reasons:

(1) Increased switching current requirement. Two MTJs in an MLC cell must be in different areas in order to differentiate the two logic bits. The soft-bit uses the smallest pillar which is the same as that in SLC design. The hard-bit size increases properly [39]. Note that J_C is fixed and I_C increases proportionally with MTJ area. So $I_{C,Hard}$ for hard-bit programming is much bigger than $I_{C,Soft}$ required for soft-bit switching, as shown in Table 8.

(2) Aggravated asymmetry in write operation. As illustrated in Fig. 23(b), the current flows from SL to BL direction when writing logic 1 (*write-1*) to an SLC STT-RAM cell. The voltage drop on MTJ causes V_{GS} degradation and limits the drivability of the selective transistor. Comparably, *write-0* is easier and faster because $V_{GS} = V_{DD}$. Moreover, the required MTJ switching current in write-1 and write-0 operations are different, usually $J_{C,0\to1} > J_{C,1\to0}$ [8]. This scenario is called as *asymmetric writes*. MLC design with more MTJs stacking in series increases the overall resistance. Thus, V_{GS} degradation becomes worse and the current from SL to BL direction further reduces.

The conventional MLC STT-RAM design in Fig. 24(a) is mainly constrained by the "write-1 to hard-bit" operation. First, it requires the highest switching current ($I_{C,Hard,0\to1}$). Moreover, the selective transistor is under the weakest biasing condition and produces the

lowest driving current when the soft-bit is 1. Even the soft-bit originally stores 0, large I_{WH1} will quickly flip it to 1, bringing the design into the worst-case condition. The scenario is illustrated in Fig. 25(a).

During the following evaluation, the transistor of baseline SLC cell is set as 4.5F, which is sufficient to write logic 0 and 1 into an MTJ with an area of 32nm × 64nm. F represents the technology feature size, which is 32nm in this work. Further reducing the transistor size does not increase density because the layout design rules, *e.g.*, metal wire and via connection of BL and SL, start dominating the cell area [58].

We simulated the driving current when writing 1 or 0 to the hard-bit of an MLC STT-RAM under the worst-case conditions. As can be seen in Fig. 26, enlarging the selective transistor helps improve the driving current. However, the conventional MLC with a transistor of 9F (2× of that of SLC) cannot supply sufficient driving current to flip hard-bit to 1 ($I_{WH1} < I_{C,Hard,0\rightarrow1}$). Further increasing the transistor size results in an even lower data density than SLC STT-RAM cache, which is meaningless.

5.3.2 Exploring More MLC STT-RAM Cell Structures

The conventional MLC structure in Fig. 24(a) have two MTJs in regular connection. In fact, it is not the only possible cell structure. The free layer in MTJ can also be fabricated underneath the reference layer to form a reverse connection [18]. The reverse connection has

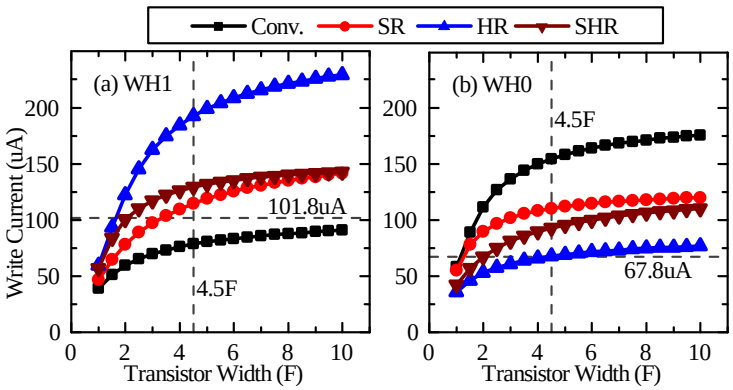

Figure 26: The hard-bit write current provided in different MLC STT-RAM cell designs. (a) write-1 current; (b) write-0 current. F = Feature Size (32nm)

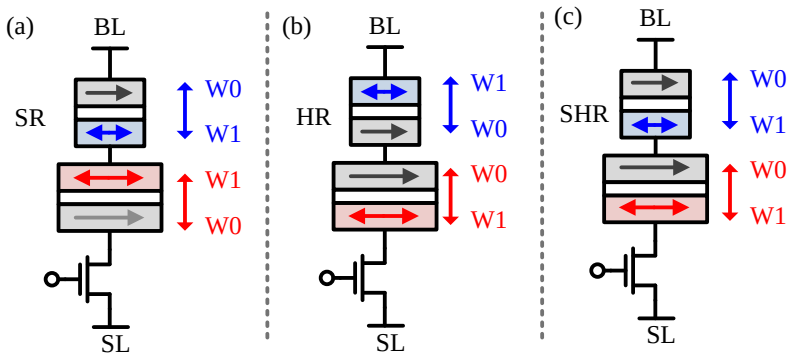

Figure 27: Other available MLC cell structures: (a) *Soft-bit reversed* (SR-MLC); (b) *Hard-bit reversed* (HR-MLC); (c) *Soft- and hard-bit reversed* (SHR-MLC).

been successfully utilized in SLC STT-RAM for cell area reduction [19]. Fig. 27 shows three new MLC STT-RAM cell designs. Based on the stacking connections of the soft- and hard-bits, we name these designs as soft-bit reversed (SR-MLC), hard-bit reversed (HR-MLC), and soft- and hard-bits reversed (SHR-MLC), respectively.

Since device characteristic is solely determined by material engineering, the change in MTJ connection does not affect the switching current requirement. So programming a hard-bit is still more difficult than its corresponding soft-bit. For comparison purpose, we simulated the driving currents when writing 1 or 0 to the hard-bit of these MLC designs under the worst-case conditions. The results are shown in Fig. 26.

It can be seen that reversing MTJ connection helps alleviate the asymmetry in write operations. For example, the worst-case condition of I_{WH1} in an SR-MLC cell is relaxed

Table 9: Margins between the driving current of MLC design with a 4.5F transistor and the required MTJ switching current

	ΔI_{WH1}	ΔI_{WH0}
Conv. MLC	$-23.37\mu A$	$86.65\mu A$
SR-MLC	$12.80\mu A$	$42.47\mu A$
HR-MLC	$90.58\mu A$	$0.23\mu A$
SHR-MLC	$27.29\mu A$	$25.05\mu A$

when the soft-bit is 0. This is because even the initial logic of the soft-bit is 1, writing 1 to the hard-bit will quickly switch the soft-bit to 0 and raise I_{WH1} up. The scenario is illustrated in Fig. 25(b). Compared to conventional MLC design, the worse-case I_{WH1} of SR-MLC grows much faster as the selective transistor size increases. As a trade-off, I_{WH0} is smaller than that of the conventional design, but still more than sufficient to conduct a successful write-0 to hard-bit.

HR-MLC reverses the hard-bit, resulting in the change of I_{WH1}'s direction from SL $\rightarrow$ BL to BL $\rightarrow$ SL. Therefore, when writing 1 to the hard-bit, the selective transistor does not suffer from V_{GS} degradation. The amplitude of I_{WH1} grows even higher than that of SR-HLC. However, its I_{WH0} degrades significantly and can barely exceed $I_{C,Hard,1\rightarrow0}$.

Table 9 summarizes the write current margins provided by four MLC designs over the required MTJ switching current, such as

$$\Delta I_{WH1} = I_{WH1} - I_{C,Hard,0\rightarrow1}, \text{ and}$$
$$\Delta I_{WH0} = I_{WH0} - I_{C,Hard,1\rightarrow0}.$$

The size of selective transistor is set to 4.5F, corresponding to twice data density over the baseline SLC. The result showed that all the three new MLC cell designs in Fig. 27 can supply sufficient driving currents under the worst-case write operation conditions. Among the four possible MLC STT-RAM cell structures, SHR-MLC achieves the most balanced current margins for both I_{WH1} and I_{WH0}.

Fig. 28 compares the write performance of all the four cells for both write-1 and write-0 operations. Because all the four types of cells have the same latency requirement on soft-bit writing, only the hard-bit write latency is presented in the figure. When setting the select transistor size at $4.5F$, the SHR-MLC design has the most balanced performance for both write-1 and write-0. The SR-MLC requires shorter write-0 latency than SHR-MLC, but its write-1 latency is much higher especially at smaller transistor size. Therefore the SHR-MLC provides the best overall write performance among all the four types of cell structures. The hard-bit write energy comparison among the four types of cells is presented in Fig. 29. For each design, the longer latency requirement of the write-1 and write-0 operations is adopted for the energy calculation. SHR-MLC demonstrates the lowest energy consumption mainly because it has the shortest overall write time.

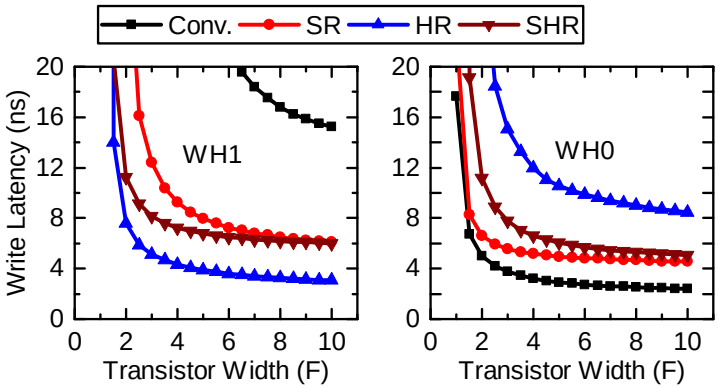

Figure 28: The write time comparison among four types of MLC cells for (a) write-1 and (b) write-0 operations. F = Feature Size (32nm)

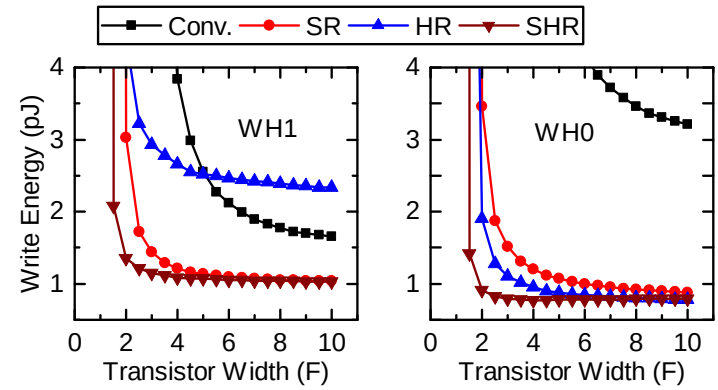

Figure 29: The write energy comparison among four types of MLC cells for (a) write-1 and (b) write-0 operations. F = Feature Size (32nm)

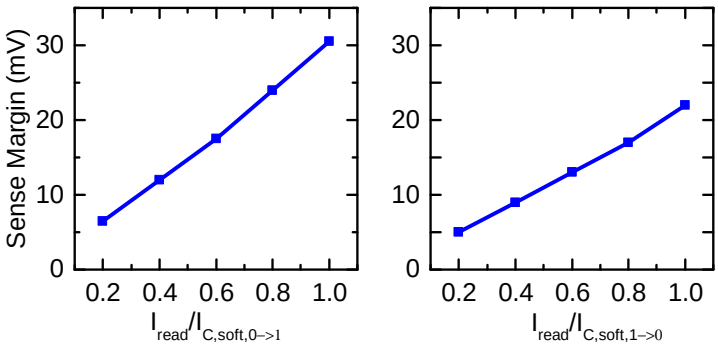

Figure 30: The sense margin when varying I_{read}/I_C, for (a) SR-MLC and SHR-MLC cells and (b) conventional MLC and HR-MLC designs. F = Feature Size (32nm)

Moreover, reversing the MTJ connection helps alleviate the read disturbance and therefore improve the read stability, with the MTJ parameters used in this work. For a MLC cell, the read disturbance mainly happens to the soft-bit because the current density received at the soft-bit is always twice larger than the current density of the hard-bit. When the soft-bit is reversed, the read current from BL to SL is along the direction of the soft-bit's write-1 operation. The required current to switch the soft-bit becomes $I_{C,Soft,0\rightarrow1} = 51.47\mu A$, which is much larger than $I_{C,Soft,1\rightarrow0} = 34.31\mu A$ in the conventional cell structure. It implies that in the design with reversed soft-bit is more resilient to read disturbance.

The read stability can also be affected by the amplitude of read current (or more precisely, the ratio of the read current and the critical switching current I_{read}/I_C), which in turn affects the sense margin. We calibrate the relation of the sense margin and I_{read}/I_C for four cell designs. According to cell structures, $I_C = I_{C,Soft,0\rightarrow1}$ for SR-MLC and SHR-MLC, while $I_C = I_{C,Soft,1\rightarrow0}$ for Conventional MLC and HR-MLC. The results in Fig. 30 show that at any given I_{read}/I_C, the SR-MLC and SHR-MLC designs have higher sense margins. In other words, under the same possibility of read disturbance, these two types of cells can tolerate high read current.

Based on previous analysis, the SHR-MLC cell provides best read/write performance and smallest cell area with the parameters in Table 9, and therefore, be adopted in the following discussion at architecture level. It is worthwhile to mention that this conclusion is not general, but determined by given device characteristics, including MTJ resistance, TMR, and critical switching current. Detailed analysis shall be performed based on given MTJ parameters before a choice of cell type is made.

5.3.3 Observations and Motivation

From system perspective, the major motivation of promoting the use of MLC STT-RAM cache is to increase the capacity and hence reduce the cache miss rate. Though the two-step read/write prolongs cache access latency, it is expected that the reduction in costly main memory accesses can amortize the impact and eventually enhance the overall system performance. However, it is not always the case.

5.4 APPLICATION-AWARE SPEED ENHANCEMENT SCHEME

First of all, the large variety of applications behave differently and demonstrate different data access patterns. Although some of them need occupy a large amount of data and demand big cache capacity, many others constrain data accesses within only a small data set that can fit into a limited cache space. For the latter cases, increasing cache capacity does not have a significant impact on the cache miss rate. Moreover, many applications show a streaming-like data access behavior: data fetched from lower level memory hierarchy will be accessed only once and then evicted. The cache miss rate in these applications is always relatively higher and usually independent of cache capacity.

Second, even within a single application, the usage of different cache sets could be very different. For example, Fig. 31 presents the set-based miss-rate of `h264ref` in a 4-way L2 cache. Many sets obtain a close-to-zero miss rate, implying that these locations unlikely benefit from capacity increase. Considering these factors, directly replacing SLC STT-RAM cache with MLC could result in system performance degradation for many applications. This has been observed in our simulations that shall be discussed in Section 5.6.

By observing the MLC STT-RAM design and read/write operation mechanism in Fig. 24, we found that a serial MLC can support SLC-like accesses: (1) reading data from a soft-bit needs only one sensing stage because the soft-bit itself determines if the total resistance falls into the lower-half or the higher-half range; (2) programming a soft-bit requires only a small current I_{WS} which does not affect the corresponding hard-bit. Based on the observations at circuit and architectural levels, in this work, we proposed an application-aware speed enhancement (ASE) scheme for MLC STT-RAM design.

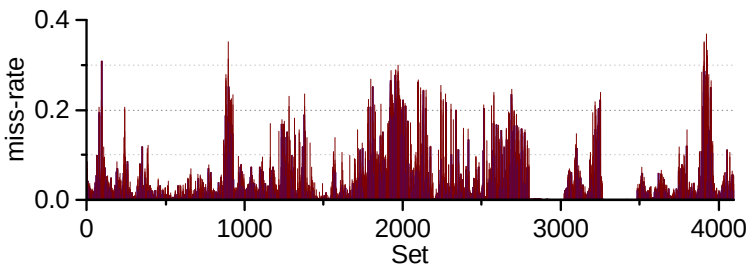

Figure 31: Miss rate statistic at different sets of a 4-way L2 cache for `h264ref`.

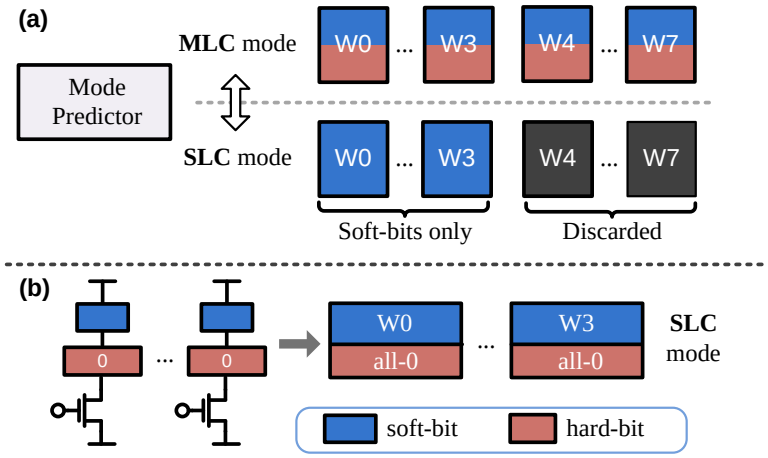

Figure 32: The set-based ASE scheme.

5.4.1 Application-aware Speed Enhancement (ASE)

Our approach enables two types of access modes for each cache set. In *MLC mode*, full storage capacity is provided but a read/write operation need two steps to complete. The cache set can also switch to *SLC mode*, in which only soft-bits will be read/written. Thus, an access can be completed quickly (in one step) though half of data storage capacity is sacrificed. According to the cache set accesses, ASE scheme dynamically switches between the two access mode.

Fig. 32 illustrates the utilization of the set-based ASE in an 8-way cache memory. Controlled by a *mode-predictor*, a cache set can stay at the MLC mode supporting 8-way accesses or change to the SLC mode at which only 4 ways are accessible. We chose to change the number of ways instead of sets, because the latter scheme requires to modify the word-line decoding circuitry which induces larger overheads in hardware and latency. In this example, ways W4-W7 are discarded when switching from MLC to SLC mode, while W0-W3 are always active in both mode. Such a set-based 8-/4-way configurations will be used in the following discussion.

5.4.1.1 Read Performance Improvement

We can further improve the read performance of SLC mode by reseting all the hard-bits to '0'. Note that an MLC cell can be at '00', '01', '10', or '11' states. Here, the first and second bits represent the data of soft- and hard-bits, respectively. As shown in Fig. 33(a), when detecting the soft-bit, the reference voltage shall be set to ref1. The sense margin defined as the difference between the reference voltage and bit-line voltage is SM1 = 8.5mV. Erasing the hard-bit to '0' reduces the possible data states to '00' and '10' only. We can shift the reference voltage to ref2 and improve the sense margin of soft-bit detection to SM2 = 18mV. Consequently, the sensing delay greatly reduces, as shown in Fig. 33(b).

5.4.1.2 The Mode Switching Control

The mode predictor is used to determine whether MLC or SLC mode shall be applied, based on cache access pattern. In implementing the set-based mode-predictor, we define the merit of MLC (M_{MLC}) and the merit of SLC (M_{SLC}) as:

$$M_{SLC} = (Hit_{W0-3}) \times (\text{Latency-Reduction}), \text{ and}$$
$$M_{MLC} = (\text{Avoidable-Miss}) \times (\text{Miss-Penalty}).$$

Here, Hit_{W0-3} counts the number of cache hits on ways W0-W3. Latency-Reduction represents the latency reduction of an access to these ways once switching it from MLC to SLC mode. In short, M_{SLC} denotes the accumulated latency reduction of switching to SLC mode.

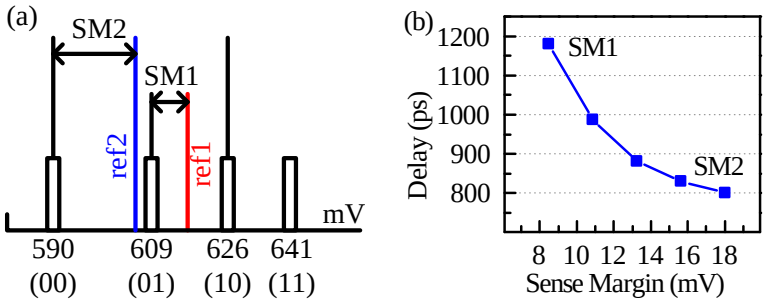

Figure 33: (a) The comparison of sense margins in MLC and SLC modes. (b) Sensing delay vs. sense margin.

M_{MLC} evaluates the latency reduction in MLC mode if that are not many lower-level memory accesses. Avoidable-Miss is the number of avoidable cache misses if changing from SLC to MLC mode. To accurately calculate Avoidable-Miss, the tag should always stay in 8-way configuration no matter what mode the set is in. Miss-Penalty is the access latency of the lower level memory hierarchy. Conceptually, the SLC mode leads to better system performance if M_{SLC} exceeds M_{MLC}, or vice versa.

The mode predictor is a saturation counter with a similar structure as [63]. It is incremented by Latency-Reduction when a hit occurs to W0-W3. If a hit falls on W4-W7, indicating that a miss can be avoided in MLC mode, the mode predictor is decremented by Miss-Penalty.

A set changes to the MLC mode when its mode predictor decreases to 0. Or, if the mode predictor reaches to preset threshold M_{Th}, the set switches to the SLC mode. At the moment, data on W4-W7 will be evicted to lower-level memory hierarchy, followed by resetting hard-bits to '0'. Considering the associated high latency cost, frequent mode changing is unaffordable and can be constrained by increasing M_{Th}. However, a very big M_{Th} could cause the mode changing to be lagged so that cache sets cannot adjust to the suitable mode in time. Therefore, M_{Th} as a key design parameter shall be carefully selected for the best performance. More discussion will be presented in Section 5.6.5.

5.4.2 Logic to Physical Mapping Strategies

The effective mapping of the logic data and physical cells is critical in the ASE scheme. It not only determines the performance but also affects the overhead induced by the SLC/MLC mode switching. In this work, we propose direct mapping and cell split mapping methods.

5.4.2.1 Direct Mapping A straight-forward way to utilize an MLC STT-RAM cache is directly mapping every N logic bits to $N/2$ MLC cells. For instance, as illustrated in Fig. 34(a), a cache line with 64-byte (512-bit) can be allocated to 256 MLC cells: half of the data bits are stored in the soft-bits and the other half are saved in the hard-bits. We name this logic data and physical cell mapping method as direct mapping (DM).

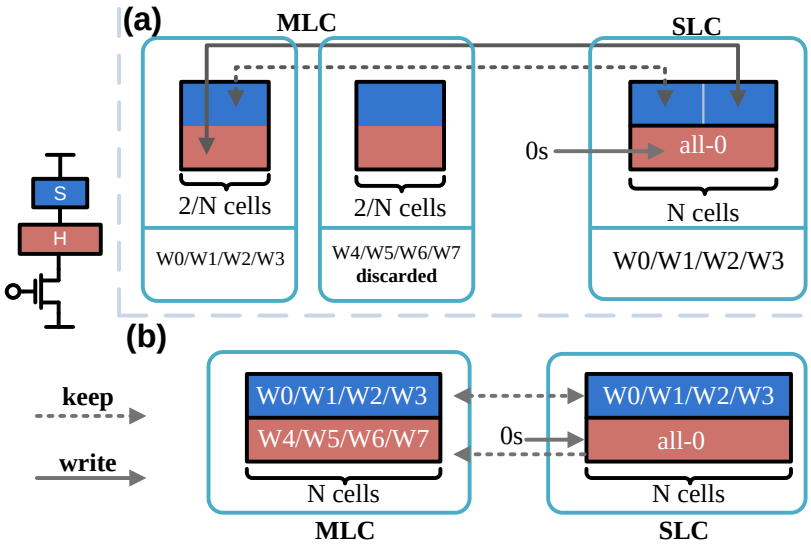

Figure 34: The physical to logic mapping in MLC mode with (a) the direct mapping, or (b) the cell split mapping.

For its simplicity, direct mapping has been naturally adopted in MLC STT-RAM cache designs [61]. Since a cache line contains both soft- and hard-bits, each read/write access need take two operation steps. Moreover, direct mapping incurs relatively high overhead during mode switching as illustrated in Fig. 34(a). When changing from the MLC to SLC mode, data stored in W4-W7 need to be read out and written to the lower-level memory before they can be discarded. Then W0-W3 need to be remapped, which introduces an extra round of read and write. When switching backward from the SLC to MLC mode, such a remapping need to be performed one more time.

5.4.2.2 Cell Split Mapping The direct mapping is not able to leverage the fast soft-bit access which requires only one-step operation. Moreover, mapping a cache line to both soft- and hard-bits will cause data reorganization whenever a mode switch occurs. To solve these issues, we propose a new cache line mapping method, named as cell split mapping (CSM).

Fig. 34(b) depict the cache architecture when adopting CSM. Half of the cache lines (W0-W3) are mapped to soft-bits, while W4-W7 are mapped to hard-bits. Recall that W0-W3 are also mapped to soft-bits in SLC mode, these ways remain unchanged during the

SLC/MLC mode switching. Also, W4-W7 can be activated without affecting the data in the corresponding W0-W3, minimizing the cost in SLC to MLC mode changing. When switching from MLC to SLC, the data stored in hard-ways shall be evicted into lower level memory hierarchy if they are marked dirty. So a read operation on hard-way and a write operation to lower level memory are needed. In general, CSM eliminates the data reorganization during mode switching and therefore greatly improves the efficiency of ASE. In the following discussion, we use "*soft-ways*" to represent W0-W3 which contains only soft-bits, and denote W4-W7 as"*hard-ways*".

Note that the CSM induces non-uniform data access latencies, determined by both the operation type and data location. A cache hit on a soft-way, no matter it is a read or write operation, can be completed in one step, which is the same as an SLC operation. The accesses to hard-ways, however, is more costly and complex. First, reading data from a hard-way behaves the same as that in an MLC cache with directly mapping. While, when writing to a hard-way, the data in the corresponding soft-way shall be protected by following the sequence of reading the soft-way data, programing the hard-way, and restoring the soft-way data back. The write access latency of an hard-way can be denoted as

$$L_{CSM,W,H} = T_{per} + T_{RS} + T_{WH} + T_{WS}, \tag{5.1}$$

where T_{per} is the latency on peripheral circuitry such as the signal routing and address decoding components. T_{RS} is the sensing time to detect soft-way. T_{WH} and T_{WS} are the time to program hard-way and soft-way, respectively. Note that $L_{CSM,W,H}$ is longer than the write latency of an MLC cache with direct mapping which is

$$L_{DM,W} = T_{per} + T_{WH} + T_{WS}. \tag{5.2}$$

Fortunately, the extra read occurs to the same MLC cells as the original write, so T_{per} can be shared.

CSM shares some similarities with *line paring* for parallel MLC STT-RAM [38], which pairs two cache line in different banks into one group and re-organizes the data. However,

due to the complex characteristics of parallel MLC, the line pairing scheme divides a cache line into write-fast-read-slow and read-fast-write-slow forms, which cannot efficiently handle the data blocks requiring high-frequent read and write accesses. It cannot provide a natural support to the SLC mode as what is proposed in this work either.

5.5 OPTIMIZATION OF CACHE WITH CELL SPLIT MAPPING

In an MLC STT-RAM cache with CSM, soft-ways and hard-ways evenly split the capacity. Without any optimization, about half of the cache hits occur on the hard-ways and suffer from long access latency. In order to reduce the hits on hard-ways and maximize the usage of soft-ways, we propose an optimization methodology which includes the intra-cell swapping mechanism, the data migration method, the shifting replacement policy during a cache miss, and the associated tag array design. Details of the optimization method will be explained in this section.

5.5.1 Intra-cell Swapping

Data migration is very common in caches with non-uniform access latencies. It is usually performed by swapping data between fast and slow regions that are assigned to different physical locations or even implemented with different memory technologies, *e.g.*, between SRAM and STT-RAM [13][38]. However, data swapping in between usually introduces large overheads in latency and energy consumption.

In the proposed MLC STT-RAM cache, a soft-way and a hard-way in the same group of memory cells, *e.g.*, W0 and W4 in Fig. 34(b), are coupled. The data swapping between coupled ways, namely, *intra-cell swapping*, is natural and easy. So our design adopts only the intra-cell swapping to reduce the data migration overhead.

For example, if swapping W0 with a way belonging to other MLCs, say, W5, the latency of such an *inter-cell swapping* is

$$T_{inter} = T_{RS0} + T_{RS5} + T_{RH5} + T_{WS0} + T_{WH5} + T_{WS5}, \tag{5.3}$$

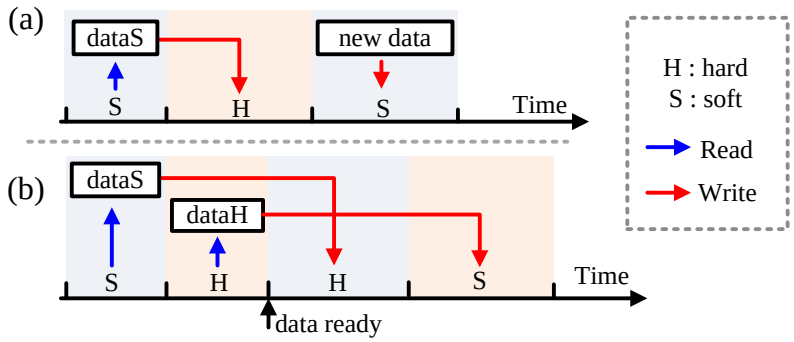

Figure 35: The timing sequence of data swapping execution enabled by a hard-way write opeation (a) or a hard-way read opeation (b).

where the suffix number 0/5 represents the way index. For comparison, the latency to complete an intra-cell swapping between W0 and W4 is much shorter, such as

$$T_{intra} = T_{RS} + T_{RH} + T_{WH} + T_{WS}, \tag{5.4}$$

where T_{RH} is the hard-bit sensing latency once the soft-bit is known. The benefit of constraining the data swapping within same MLCs is obvious by comparing T_{inter} and T_{intra}.

Executing data swapping when memory is idle can alleviate the impact on system performance but cannot avoid extra energy overhead. Instead, our approach tends to hide the swap operation into normal read/write accesses to hard-ways. Fig. 35 shows the timing diagram of data swapping enabled by a hard-way access, which can be a write or a read.

5.5.1.1 Write & Swap For a data swapping triggered by a hard-way write, we can move the data of its corresponding soft-way to the hard-way and allocate the new data to the soft-way. It is not necessary to read the hard-way which will be over-written by the incoming data. This operation is exactly the same as a normal hard-way write, with a latency summarized in Eq. (5.1). It does not induce extra latency or energy overhead.

5.5.1.2 Read & Swap Data swapping can also be initiated by a hard-way read. The soft-way is read-out first, followed by the hard-way read. Then the two data blocks are swapped and written into the hard-way and soft-way in sequence. Note that the read-out data can be used for further operation without waiting for the completeness of writes. So the swapping will not cause extra delay to this read access. Also, a great amount of energy cost of swapping such as decoding and sensing can be absorbed by the normal read access.

Although the inter-cell swapping can provide more flexible data migration and enhance the soft-way utilization, the big latency overhead cannot be completely hidden by normal operations. Our evaluation in Section 5.6 shall show that the intra-cell swapping together with simple data migration policy can allocate more than 90% of cache hits to soft-ways. Thus, we didn't adopt the inter-cell swapping between different MLC cells in this work.

5.5.2 Migration Method

Data migration is possible with the support of the swapping mechanism. Our objective is to move frequent-access data blocks to soft-ways that require only one step in read and write operations. Here, we propose two methods, namely, counter-based migration (CM) and aggressive migration (AM), to control the data movement between soft- and hard-ways.

5.5.2.1 Counter-based Migration A counter `Hcnt` is assigned to each pair of coupled soft- and hard-ways to track access frequency. When a hit occurs on an soft-/hard-way, `Hcnt` increases/decreases one. If `Hcnt` reaches a pre-set threshold (H_{Th}), indicating that more accesses hit the hard-way than the soft-way, we swap their data and reset `Hcnt` to 0. This flow is shown in Fig. 36(a). The overhead of CM mainly comes from the counters.

5.5.2.2 Aggressive Migration It is a simpler scheme without counters. Considering the fact that modern embedded processors usually utilize write-back L1 cache for energy reduction [64], a large portion of writes to L2 cache are caused by dirty line eviction from L1 cache. Many of these data could be sent back to L1 cache again. AM exploits this fact and triggers data swapping whenever a write hits on a hard-way, as shown in Fig. 36(b).

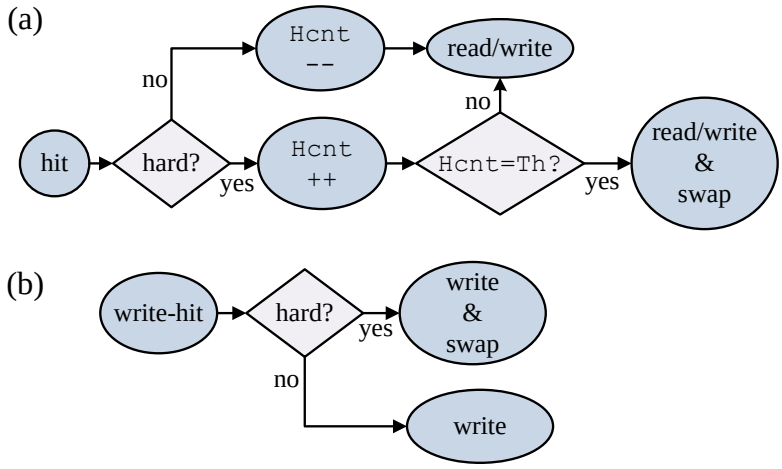

Figure 36: Data migration flows. (a) Counter-based migration, and (b) Aggressive migration.

It guarantees that the most recently written data always stay on soft-ways. AM will cause more data swaps than CM because every write-hit on hard-way triggers a swap. However, the swapping itself does not induce any overhead because it is totally hidden by write operation as previously discussed. Moreover, AM does not require counters or other complex logic so the area overhead is negligible.

5.5.3 Shifting Replacement Policy

When a cache miss occurs, an old cache line will be evicted and replaced with new data fetched from lower-level memory. The widely adopted replacement policy like LRU tends to choose the least recently used data as a candidate for replacement. While applying our proposed data migration method, such data is likely to be located on a hard-way. This causes potential harm on performance because the new data usually incurs more frequent accesses and should be placed in a soft-way that offers better access speed. Thus, we propose a shifting replacement policy which is a modified version of LRU, an example of which is illustrated in Fig. 37: if a hard-way (*e.g.*, W4) is chosen to be evicted when applying *least recently used* (LRU) replacement policy, instead of putting the new data directly into the hard-way, we locate it to the corresponding soft-way (*e.g.*, W0) meanwhile *shift* the data of

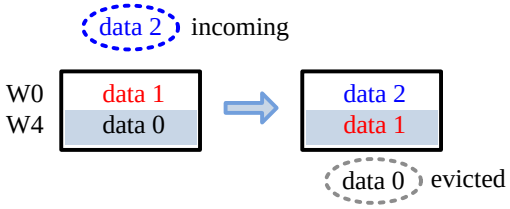

Figure 37: The shift-like line replacement.

W0 to W4. With the proposed replacement policy, the new data will always be placed in a soft-way which guarantees fast access. The latency of such a shifting replacement remains the same as a hard-way write as described in Eq. (5.1).

5.5.4 Tag Array Design Utilizing CSM

Due to the concern on system performance. previous MLC STT-RAM cache designs usually uses SLC to implement tag arrays. The major drawbacks of the approach are the large array area and the increased design complexity caused by different types of cell structures. Here, we propose to apply MLC in tag array. Besides the smaller design area that helps reduce the fabrication cost, another major advantage of the MLC-based tag array is having the same structure for both the tag and the data arrays. The compatibility in array design style eventually results in the design cost reduction through sharing read/write peripheral circuitry and easing the layout organization. Similar to data array design, we utilize CSM to reduce the tag search latency. An illustration is shown in Fig. 38, where the physical

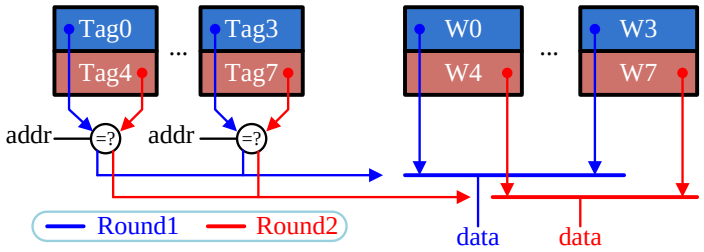

Figure 38: MLC STT-RAM tag array design utilizing CSM.

location of tag and data blocks present an one-to-one correspondence, *i.e.*, both Tag0 and W0 use soft-bits while Tag7 and W7 are located on hard-bits. Accordingly, a two-round tag searching method takes advantage of possible fast accesses of CSM. At the first round, only those tags located on soft-bits (*i.e.*, Tag0-Tag3 in Fig. 38) are read out and compared with the target address. If a match is found, the data on the corresponding way can be identified and the tag search is completed. Otherwise, the second round of search will be performed on the hard-bits (*i.e.*, Tag4-Tag7) and the remaining ways are searched. During the procedure, the read out data of the first round search shall be kept and will be used in reading the hard-bits in the second round. Thanks to the data migration methods that guarantee the majority of hits happen to soft-bits, most tag searches will only require one round with a latency equal to that of an SLC tag. Thus, the system performance after applying the new tag design is close to previous SLC tag design while the area can be greatly reduced.

5.6 ARCHITECTURAL LEVEL EVALUATION

5.6.1 Experimental Setup

We conducted the performance evaluation by using the cycle-accurate simulator MacSim [65]. Its built-in cache model was modified to implement our architecture level techniques. The baseline architecture setup is a Dual-Core embedded processor with two-level cache hierarchy, which is similar to Intel Atom [64]. The configuration details of CPU core and L1 cache are summarized in Table 10.

Table 10: Configuration of CPU, L1 Cache and Main Memory

CPU	1.86GHz, 2 Cores, in-order, 2-way issue
L1 Cache (SRAM)	16K+16K, 64B, 4-way, Write Back 1R+1W ports, 2 Cycles R/W
Main Memory	1GB, 400 Cycle, 31nJ/access[66]

Table 11: Different Configurations of STT-RAM L2 Cache

	SLC	**Conv-MLC**	**ASE MLC**
Cell Size	$36F^2$ (4.5F transistor) [58]		
Capacity (Byte)	2M	4M	4M
Configuration	64B line, 8-way, Write-Back 4 Banks 1 R/W port per bank		
Read Lat. (Cycles)	6	9	S: 6/7 H: 10
Write Lat. (Cycles)	23	42	S: 23, H: 45
Read Energy (nJ)	0.415	0.424	S: 0.424, H: 0.576
Write Energy (nJ)	0.876	1.859	S: 1.081, H: 2.650
Leakage (mW)	80.8		

SPEC CPU2006 benchmarks [67] were adopted in the architecture simulations. For each benchmark, we fast-forwarded 500 million instructions and then executed 1 billion instructions. The processor performance is measured by the *instruction per cycle* (IPC). In the work, we compared the following STT-RAM L2 cache designs:

- **SLC**: SLC STT-RAM cache;
- **Conv-MLC**: Conventional MLC STT-RAM cache;
- **ASE**: Our proposed ASE MLC STT-RAM cache design, using direct mapping method;
- **ASE+CSM**: The ASE cache with CSM;
- **ASE+CSM+CM**: The ASE cache with CSM, applying counter-based data migration;
- **ASE+CSM+AM**: The ASE cache with CSM, integrating aggressive migration.

Our proposed ASE MLC STT-RAM cache adopted the SHR-MLC cell structure in Section 5.3 that offers 2× data capacity than SLC cache. Both SLC and MLC cell utilized a $4.5F$ transistor. Further decreasing the transistor size does not reduce the actual cell size because the layout design rules start dominating the cell area [58]. The data-array of both

SLC and MLC caches is composed of sub-array with size of 1024×1024, and the bit-line latency of MLC is 2.696*ps* higher than that of SLC because of the small difference in resistance value. The CSM-based MLC tag array was used to all the CSM-related cache designs, otherwise SLC tag array was deployed.

By default, the `Hcnt` threshold (H_{Th}) is set to 32 and the threshold of mode-predictor (M_{Th}) is set to 1024. Table 11 summarizes the configurations of the STT-RAM L2 caches, where the latency and energy parameters were obtained by using NVsim [47]. The MTJ and CMOS technology parameters can refer Table 8.

5.6.2 The ASE MLC Cache

Fig. 39 compares the system performance when utilizing SLC, Conv-MLC, and ASE cache designs. The IPC performance was measured on 19 benchmarks and their arithmetic average is denoted as "*avg*". Compared to SLC, the average IPC of Conv-MLC improves 1.2%, while the effectiveness varies significantly by applications. The performance improvement (*e.g.*, `bzip2`) mainly comes from the miss-rate reduction, benefiting from the large capacity of MLC cache as shown in Fig. 40. For benchmarks that cannot take advantage of the larger cache capacity, the system performance degrades because of the two-step access of Conv-MLC. These benchmarks either demonstrate extremely low cache miss rates (*e.g.*, `gamess`) or merely reduce misses even cache capacity is enlarged (*e.g.*, `lbm`).

The ASE cache performs MLC/SLC mode switching dynamically by monitoring the cache miss-rate. It has a similar high IPC in `bzip2` as Conv-MLC, mainly due to the miss-rate reduction induced by enlarged capacity in MLC mode. For the benchmarks with few cache misses, *e.g.*, `gamess`, it stays at SLC mode that offers fast accesses. On average, the ASE cache improves performance by 3.4% and 2.1% compared to SLC and Conv-MLC, respectively. However, limited by the long access latency of conventional direct mapping in MLC mode, the performance gain of ASE is not significant.

Fig. 41 shows the normalized dynamic energy consumption on both STT-RAM L2 cache and main memory. SLC consumes the least dynamic energy on STT-RAM cache because both read and write operations can complete within one step. However, it has the highest

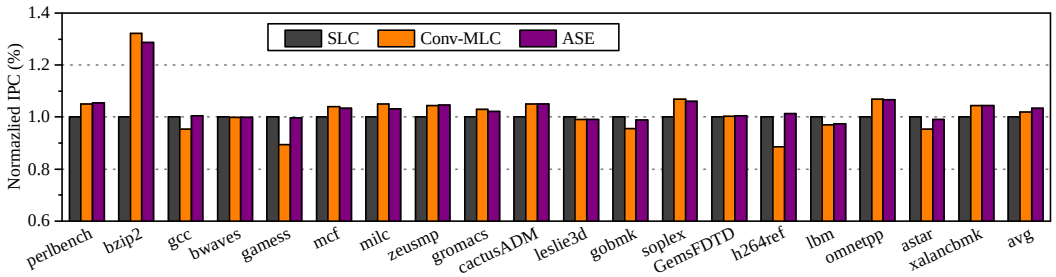

Figure 39: IPC comparison among SLC, Conventional MLC and ASE (normalized to SLC).

energy consumption on main memory among all the designs due to high cache miss rate. Conv-MLC increases L2 cache energy 55% because of the complex and long read/write operations. However, the overall energy reduces 3.3% on average, thanks to the doubled cache capacity and therefore reduced main memory accesses. ASE keeps the main memory energy benefits of MLC, and further reduce the energy on cache memory by 6.4% because the low energy cost during SLC mode.

5.6.3 The ASE Cache with CSM

Applying CSM to ASE not only accelerates the accesses to half of cache lines but also leverages the extra data capacity. In addition, CSM naturally supports the switching between SLC and MLC modes with minimal overhead. As shown in Fig. 42, all the benchmarks obtain

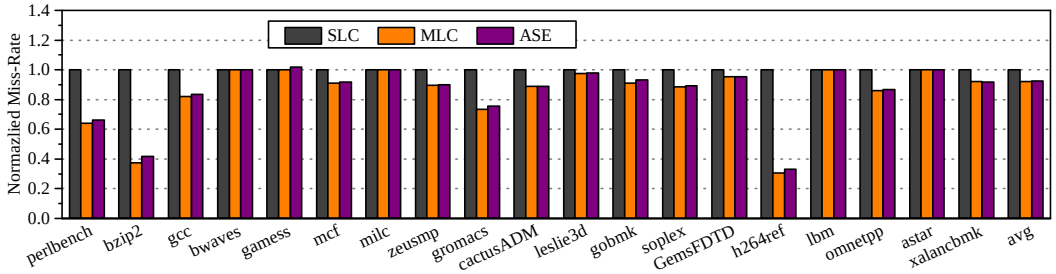

Figure 40: Miss-rate comparison among SLC, Conventional MLC and ASE (normalized to SLC).

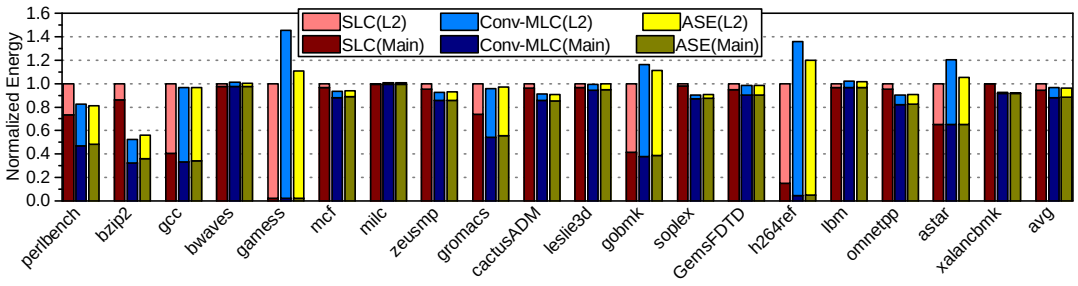

Figure 41: Total dynamic energy consumption among SLC, Conventional MLC and ASE (normalized to SLC).

performance enhancement after adopting CSM. Even without utilizing any data migration scheme, ASE+CSM obtains averagely 3.8% and 2.2% IPC performance improvements over Conv-MLC and ASE, respectively.

The energy consumption on main memory remains almost same when integrating CSM with ASE. This is because the change of mapping method does not affect much on the miss-rate. Energy on L2 cache reduces by 5.0% as shown in Fig. 43, since accessing soft-ways requires less energy than conventional mixed ways containing of both soft-bits and hard-bits. Also, unlike conventional mapping method, ASE+CSM does not need data remapping when switching between MLC and SLC modes. However, without specific data control, almost half of the accesses in MLC mode go to hard-ways. So the energy reduction over conventional direct mapping is not very significant.

5.6.4 Data Migration Scheme Comparison

5.6.4.1 Effectiveness of Data Migration The proposed data migration schemes attempt to move the cache lines with frequent accesses to soft-ways. The effectiveness can be evaluated by using soft-hit faction F_S defined as

$$F_S = (\#\text{hits-on-soft-ways})/(\#\text{total-hits}).$$

Fig. 44 compares F_S of different policies. Without applying any data migration, F_S is in the range between 50% and 60% for most benchmarks, with an average of 56%. Simply utilizing the shifting replacement policy (denoted as *shift*) increases F_S to 67%, because it always put

the recently fetched data to soft-ways. Not surprising that on average, the design adopting the shifting replacement policy and counter-based migration (*CM+shift*) obtains the highest F_S of 90.4% since it counts the occurrence of hits for both read and write accesses and move the frequently accessed lines to soft-ways. The aggressive migration moves a cache line to soft-way only when a write hits hard-way. The read-hits are ignored so some data swapping opportunity could be missed. The average F_S of *AM+shift* is 84%, which is still significantly higher than the design without any migration.

5.6.4.2 Performance and Energy After moving most of accesses to soft-ways, CM and WA migration policy obtained 4.4% and 3.1% performance improvement over CSM+ASE without data migration scheme, respectively. Compared with SLC or conventional MLC, the overall performance improvement of ASE+CSM+CM is 12.4% and 10.2%, respectively. The cache energy consumption of ASE+CSM+CM is 1.5% higher than ASE+CSM because of the data swapping overheard, but it's still 9.5% lower than a Conv-MLC cache design. ASE+CSM+AM shows slightly less IPC performance, but significant lower cache energy than ASE+CSM+CM. Compared to Conv-MLC, however, ASE+CSM+AM improves 8.8% in IPC and saves 26% of cache energy because the swapping of AM occurs with hard-way writes only. And "write & swap" does not incur latency and energy overhead (Section 5.5).

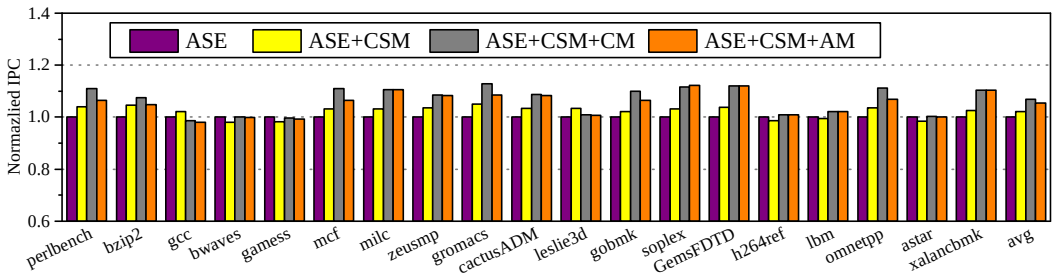

Figure 42: IPC comparison after applying data mapping and migration policies (normalized to ASE).

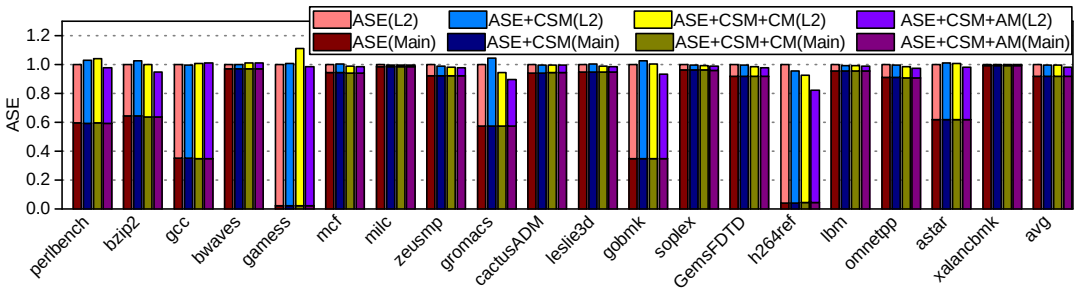

Figure 43: Total dynamic energy consumption comparison after applying data mapping and migration policies (normalized to ASE).

5.6.5 Sensitivity Study

A sweet spot of M_{Th} in terms of performance exists, as shown in Fig. 45(a). A small M_{Th} aggressively forces more cache sets to stay at SLC mode, resulting in high cache miss rate. A large M_{Th}, on the other hand, delays the switching to SLC mode even a cache set shows extreme low miss rate. Based on our exploration, M_{Th} of 1024 is optimal for average performance. When switching from MLC to SLC mode, the energy overhead associated with dirty data eviction and hard-way resetting shall be considered. Fig. 45(b) shows the relation of such energy overhead and M_{Th}. When M_{Th} decreases from 2048 to 512, the energy overhead increases 34%. Fortunately, the energy overhead caused by mode switching accounts for less than 1% of the total energy even decreasing M_{Th} to 512. So it does not affect much on the energy benefits of the proposed ASE MLC STT-RAM cache.

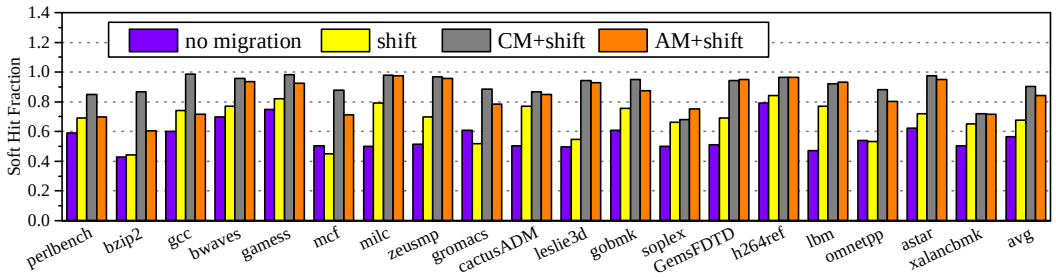

Figure 44: The fraction of soft-way hits F_S.

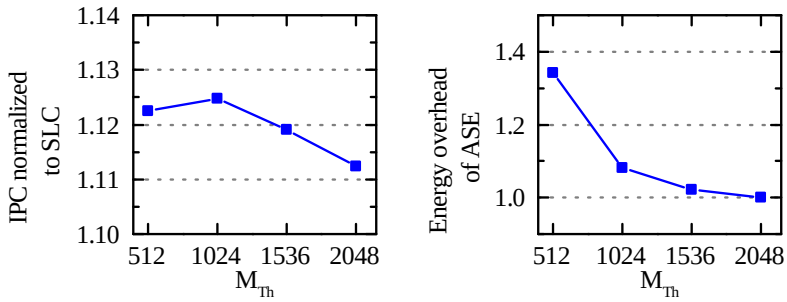

Figure 45: (a) The average IPC of all 19 benchmarks (normalized to SLC). (b) The energy overhead caused by mode switching, normalized to M_{Th}=2048.

The threshold of `Hcnt` (H_{Th}) is used to control the data swapping frequency in CM migration policy. Fig. 46(a) shows the trend of average IPC performance when varying H_{Th}. A smaller H_{Th} indicates easier cache lines swapping, assuring a quick response to the change of access patterns. So IPC increases quickly when H_{Th} decreases from 96 to 64. However, when H_{Th} further decreases from 64 to 16, IPC starts dropping because of too many "read & swap". Although "read & swap" does not delay the ongoing read operation, the extra write induced by data swapping might stall the following cache accesses. If H_{Th} is too small, the probability of such stalls increases quickly and hurts system performance. Moreover, the high occurrence of swapping increases the energy overhead. Fig. 46(b) demonstrates that the dynamic energy on the L2 cache increases significantly as H_{Th} decreases.

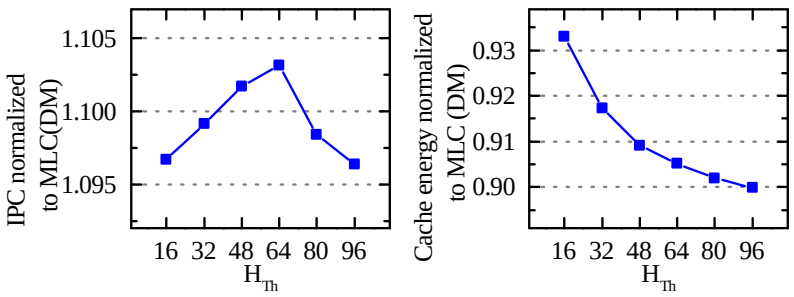

Figure 46: (a) The average IPC of all the 19 benchmarks, normalized to MLC(DM). (b) L2 Cache dynamic energy, normalized to MLC(DM).

5.7 SUMMARY

In this chapter, we studied the design challenges in implementing MLC STT-RAM as on-chip caches. Our analysis showed that the conventional design may not continue the density benefit as expected under scaled technology, but potentially degrade system performance. Accordingly, a cross-layer solution was proposed to address these design challenges. At the circuit level, we introduced the reversed MTJ connection to MLC STT-RAM cell design. Through proper device and design tradeoff, 2× capacity over SLC is promised. At the architectural level, the application-aware speed enhancement scheme was proposed which can adaptively adjust cache configuration to tradeoff capacity and speed. Moreover, the cell split mapping differences the fast-region and slow-regions in cache architecture and the according data migration methods allocate the frequently used data to fast-regions. Compared to conventional MLC STT-RAM cache design, the proposed MLC cache design can improve the system performance by 10.2% while reducing dynamic energy consumption on L2 cache by 9.5%.

6.0 A 4KB STT-RAM TEST CHIP SUPPORTING MULTIPLE TYPES OF MTJS

6.1 MOTIVATION

In recent years, many STT-MRAM test chips with capacity between 4Kb and 64Mb [68, 69, 15] have been successfully demonstrated by major semiconductor and data storage companies. For example, in November 2012, Everspin started shipping 64MB STT-MRAM in DDR3 DIMM format [15], announcing the commercialization of STT-MRAM. Croscus also unveiled the thermal-assisted STT-MRAM chip to store transaction data on smartphones and smartcards [70].

Although STT-MRAM has achieved some milestones of its deployment, there are still many concerns on the scalability of the technology, especially the current elliptical-shaped MTJ (E-MTJ). In particular, the shape anisotropy energy and the stray field energy of a nano-scale MTJ increase rapidly, leading to large coercivity and switching field energy [71]. These issues hinder the further reduction of the write current amplitude and hence, requiring a large-size access transistor to supply the needed write current. To simultaneously improve the programmability while also maintain thermal stability of STT-MRAM, it has been suggested to build a nano-ring shaped MTJ (NR-MTJ) whose magnetization directions can be directly controlled by the spin-polarized current and spin-transfer torque effect [72].

In this work, we fabricated a 4Kb test chip to validate the technology feasibility of STT-MRAM with NR-MTJs. The designed outer and inner diameters of the NR-MTJ are 200nm and 120nm, respectively. There are two operating voltages on the test chip, say, 2.5V and 1.2V: The 2.5V power serves as the write voltage of the STT-MRAM cells to provide sufficient write current to the MTJ. The access transistors and the write drivers,

hence, are also 2.5V devices. For energy and area concerns, the 1.2V power is supplied to all other circuit modules. The write control signals generated from 1.2V module is converted to 2.5V signals using level shifters. Testing results demonstrated successful read and write functionalities of our chip as well as distinctive write current reduction achieved by NR-MTJ w.r.t. E-MTJ, proving the theocratically-predicted electrical advantages of the STT-MRAM with NR-MTJs.

6.2 NANO-RING SHAPED MTJ (NR-MTJ)

For the MTJ structure with in-plane magnetization, the thermal stability Δ is greatly determined by the shape anisotropy (K_D). A specific elliptical shape is required to stabilize the magnetization along the long in-plane axis in order to maintain a thermal energy barrier for data storage. However, it has been pointed out by many prior arts that the E-MTJ suffers from several scalability issues, i.e., the large write current and the difficulty to maintain the elliptical shape in the scaled technology node. Hence, many new types of MTJ structures including NR-MTJ have been innovated to ensure lower write current amplitude and better manufacturing scalability.

Figure 47 shows the structure of NR-MTJ and its current-induced switching between two resistance states. The NR-MRJ has the same vertical stack structure as the E-MTJ. However, a hole is created in the center of the NR-MTJ to form a vortex structure free of magnetic poles in the magnetization [73]. As a result, the thermal stability is improved and the required write current density is reduced.

There are two *domain walls* (DWs) in the free layer and fixed layer within the ring width. And two semicircular domains are separated by these two DWs. The DW in free layer can move under an external magnetic field or STT current, but the DW in reference layer is pinned, which is similar to E-MTJ. The NR-MTJ is distinguished by two states: onion state and twisted state according to the positions of the two DWs in free layer. For onion state, the positions of the DWs is the same as that of reference layer, leaving NR-MTJ in low resistance state. When a write current is applied, the two DWs move toward each

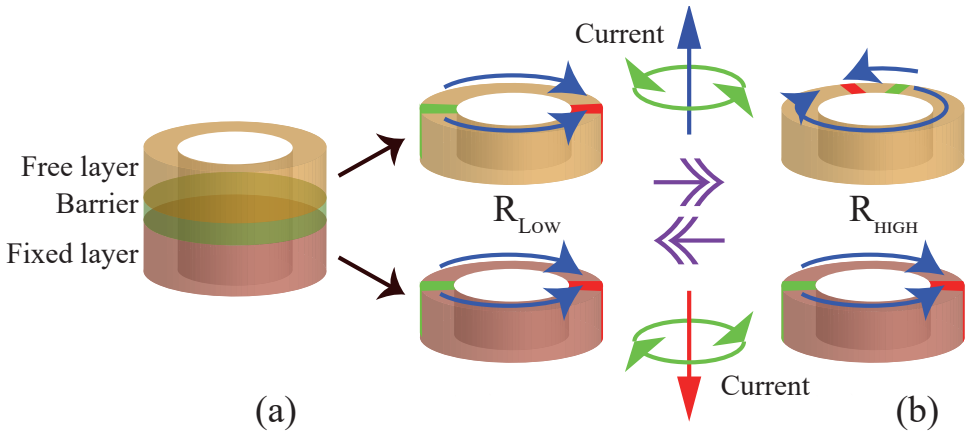

Figure 47: (a) NR-MTJ structure. (b) NR-MTJ switching.

other under the generated circulatory magnetic field, as depicted in Figure 47(b). For a thin nano-ring existing in the NR-MTJ, the two half vortices within free layer cannot annihilate but form the twisted states with two unequal domains. As a result, the NR-MTJ is in high resistance state. The shapes of the NR-MTJ and the center hole are usually selected to be round due to manufacturing robustness concern though they can be other shapes like ellipse or polygon.

Besides NR-MTJ, using MTJ with out-of-plane anisotropy (i.e., perpendicular-MTJ) or thermally-assisted mechanism can also help to reduce write current amplitude and improve thermal stability. However, these designs require either special materials or extra antiferromagnetic layer [74], incurring substantial extra fabrication cost and complexity.

6.3 DESIGN OF 4K-BIT TEST CHIP

We built a 4Kb STT-MRAM test chip with the developed NR-MTJ devices. Besides a memory array with the aforementioned 1T1J cell structure, the test chip also includes the WL decoder, the *write driver* (WD), the column multiplexer and decoder, the *sense amplifiers* (SA), and the timing control block, as illustrated in Figure 48.

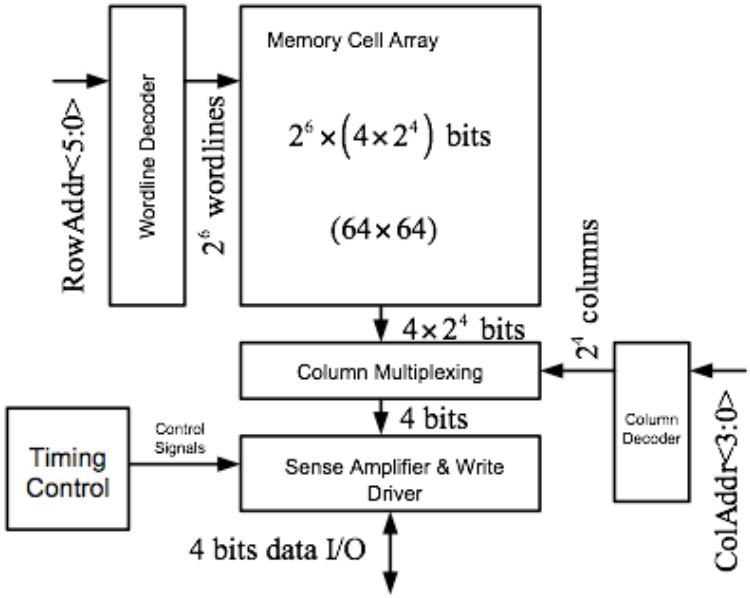

Figure 48: 4K test STT-MRAM chip organization.

There are 64 rows and 16 columns in the memory array to ensure an aspect ratio close to 1:1. The word length is 4-bit. For area efficiency consideration, 4 sets of SAs and WDs are shared among 16 columns via the column multiplexer. The control signals to SAs and WDs are generated from the timing control block.

6.3.1 Memory Cell Design

Figure 49 shows the "*one-transistor-one-MTJ* (1T1J)" STT-MRAM cell design [75] adopted in our STT-RAM design where one NMOS transistor is connected to the MTJ. The NMOS transistor whose gate connects *wordline* (WL) controls the access of the STT-MRAM cell and supplies the read and write current passing through the MTJ. The increase in the MTJ write current amplitude requires increasing the size of the NMOS transistor and hence, resulting in a larger STT-MRAM cell area.

During write operations of the STT-MRAM cell, proper voltage biases are applied to the *bitline* (BL) and the *sourceline* (SL) to control polarization of the write current. During read operations of the STT-MRAM cell, a predetermined read current is applied to the MTJ. The generated voltage on the BL is compared to a reference voltage, which is either generated from dummy cells or outside signals.

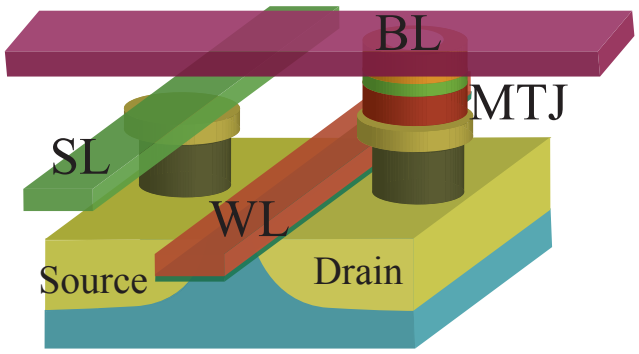

Figure 49: 1T1J STT-MRAM cell.

Figure 50(a) shows the layout of the 1T1J STT-MRAM cell with NR-MTJ. The width (W) and length (L) of the NMOS transistor, which is a 2.5V device, are 2μm and 280nm, respectively. The finger number of the NMOS transistor is 2, which keeps the aspect ratio of cell and thus the array around 1:1. The designed driving ability of the transistor is 530.41μA and 493.04μA for the MTJ switchings of "0→1" and "1→0", respectively. The layout of NR-MTJ is two nested squares whose side lengths are 200nm and 120nm, respectively. As a comparison, Figure 50(b) shows the layout of E-MTJ, which is a 175nm×75nm rectangle.

6.3.2 Write Circuitry

6.3.2.1 Write driver A bidirectional write current needs to be supplied by the write driver to switch the MTJ between two resistance states. As shown in Figure 51(a), both BL

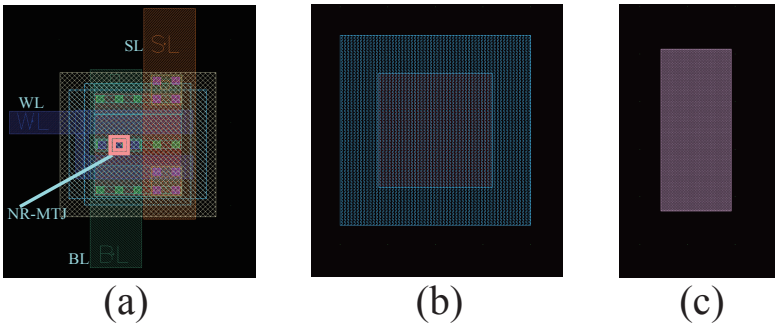

Figure 50: Layouts of (a) STT-MRAM cell. (b) NR-MTJ. (b) E-MTJ.

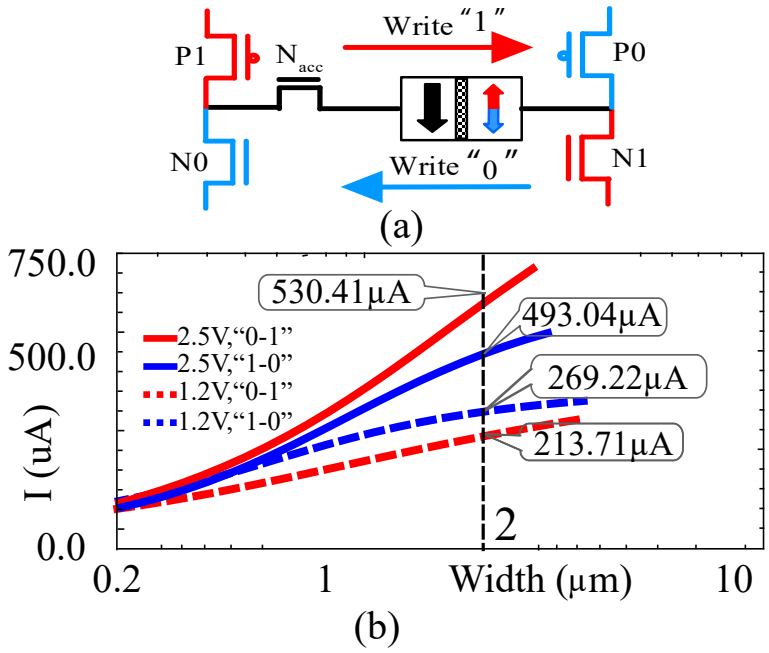

Figure 51: (a) Bidirectional write driver. Driving abilities of 1.2V device and 2.5V device.

and SL are connected to two sets of a source PMOS and a drain NMOS that are controlled separately. When writing '0', N0 and P0 are turned on and the write current I_0 flows through the MTJ from free layer to reference layer; When writing '1', N1 and P1 are turned on and the write current I_1 flows from the opposite direction. As discussed before, the voltage drop on the MTJ reduces the V_{GS} of the access transistor N_{acc} when writing '1', *usually* resulting in a lower write current than that in writing '0'.

We simulate the relation between the transistor width and the supplied write current at the switching's of "0→1" and "1→0", respectively, as shown in Figure 51(b). The results show that the normal 1.2V transistor cannot supply the needed write current at "0→1" switching, i.e., 494.87μA even when the width of N_{acc} raises to 2μm. Hence, we select 2.5V transistor in our design as the access device.

When using 1.2V access transistor, the supplied current in writing '1' is smaller than that in writing '0', which is consistent with the results in previous discussion. However, when using 2.5V access transistor, the supplied current in writing '1' becomes larger than that in writing '0'. The reason for this observation can be explained as follows: The driving ability

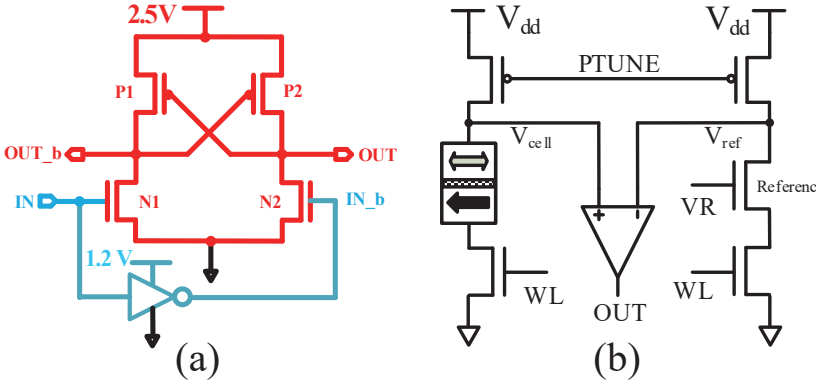

Figure 52: Schematic of (a) Level shifter. (b) Read circuit.

of the N_{acc} is affected by two factors: the V_{GS} reduction and the V_{DS} reduction, both of which are caused by the voltage drop on the NR-MTJ. When writing '1' in the 2.5V design, the initial resistance state of the NR-MTJ is low, the degradation of the N_{acc} driving ability incurred by V_{GS} reduction is compensated by the less reduction of V_{DS} compared to writing '0'. Nonetheless, the required transistor width is still very large, say, 2μm in our design due to the large MTJ size.

6.3.2.2 Lever shifter There are two sets of power supply on our test chip: 1) 1.2V for read circuitry and timing control; and 2) 2.5V for write circuitry. Level shifters are designed to translate the 1.2V control signals (IN/IN_b) generated from the timing control to 2.5V control signals (OUT/OUT_b) to the 2.5V write circuitry, as shown in Figure 52(a). The level shifter is mainly determined by the pull-down speed of the 2.5V transistors N1 and N2 under a 1.2V gate bias. In our chip testing, we found that the driving ability degradation of N1 and N2 caused by transistor threshold variation significantly affects the level shifter performance and consequently, the robustness of shifter operations. But it can be easily fixed by raising the 1.2V voltage to a slightly higher level.

6.3.3 Read Circuitry

6.3.3.1 Adjustable reference cell We adopted voltage-sensing in our read circuitry design, as shown in Figure 52(b). A carefully selected PMOS tuning voltage PTUNE is applied to the gates of the PMOS transistors in both the selected STT-MRAM cell and the

reference cell whose WLs are asserted to high. Voltage V_{cell} and V_{ref} are then generated on the BLs of the two cells accordingly. Here the reference cell is a NMOS transistor whose equivalent resistance is set to a value in between R_H and R_L by tuning its gate voltage VR. The generated V_{cell} and V_{ref} are sent to a SA for data readout. If the MTJ in the accessed STT-MRAM cell is in low (high) resistance, V_{cell} will be lower (higher) than V_{ref} and the SA outputs a logic '0' ('1'). In our design, one reference cell is shared by the whole row in the STT-MRAM array to minimize the incurred area overhead.

The raw sense margin (—V_{ref}-V_{cell}—) must be large enough to overcome signal noise and intrinsic input offset of the SA generated from device mismatch. In addition, the standard deviations of the two resistance states of the MTJ have been proven not equal [76]. Hence, the optimal reference voltage level is slightly lower than the middle between the two V_{cell}'s corresponding to the mean values of the two MTJ resistance states, and determined by the actual on-chip distribution of MTJ resistance along the same row. Our testing results show that adjusting the VR can significantly improve the readability of the test chip, as we shall show in Section 6.4.1.

6.3.3.2 Sense amplifier Figure 53(a) depicts the schematic of our SA design. Prior to read, port $\overline{\text{PC}}$ is asserted to ground, pre-charging $\overline{\text{OUT}}$ and OUT to Vdd. After that, sensed voltage (V_{cell}) is applied on port IN and the reference voltage V_{ref} is applied to port Ref. Then a sense enable signal SAN turns on transistor M_7, commencing discharging $\overline{\text{OUT}}$ and

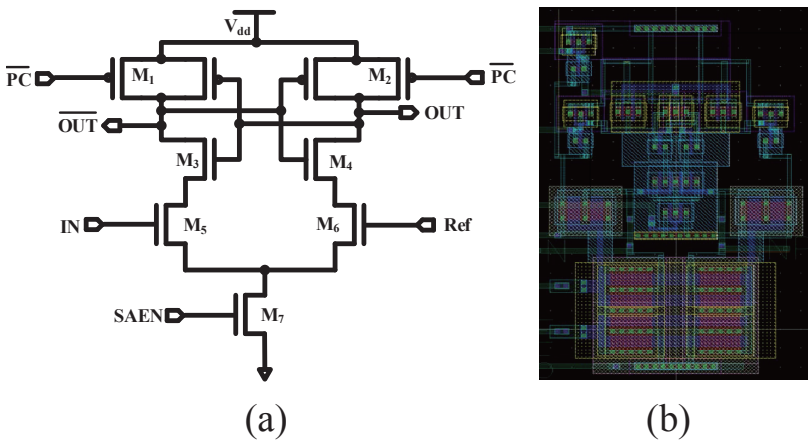

Figure 53: (a) Schematic of SA. (b) Layout of SA.

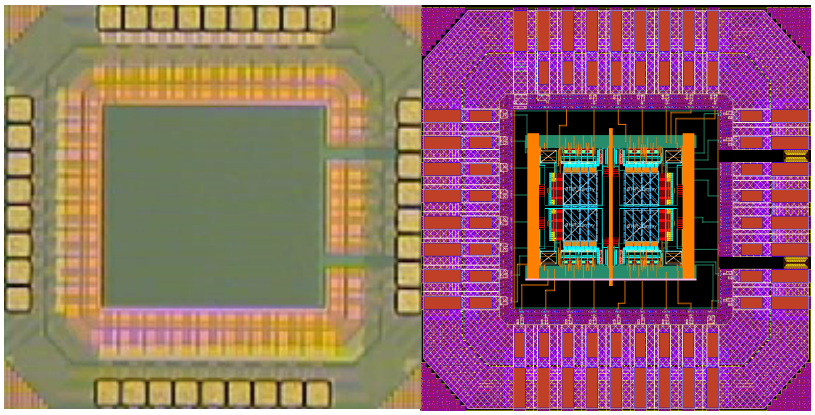

Figure 54: (a) Die photo and (b) Test chip layout.

OUT. If V_{cell} is larger than V_{ref}, for example, the left branch of the SA discharges more quickly than the right branch. As a result, $\overline{\text{OUT}}$ will be grounded and OUT will be pulled up to V_{dd}.

To accelerate the discharging speed of the branches of the SA, M_5, M_6 and M_7 are especially sized up, as shown in Figure 53(b). The large transistor size also helps to mitigate the impact of the device mismatch between the two branches and improve the sensing reliability.

6.4 EXPERIMENTAL RESULTS

Our 4Kb STT-MRAM with NR-MTJ is fabricated with 65nm technology in a leading foundry in Asia. The preparation of the magnetic devices is performed by a third party. Figure 54 shows die photo and layout of the test chip.

6.4.1 Functionality Verification

Figure 55 shows the measured results of the readout data and the critical timing signals, i.e., PC (pre-charge), SAE (SA enable), and CLK.

As aforementioned, VR and PTUNE can be adjusted to overcome the impact of process variations. Figure 56 shows the measured relations between the readout result and these two voltages. Here, Y axis refers to the decimal value of 4-bit readout data between 0 (b0000)and

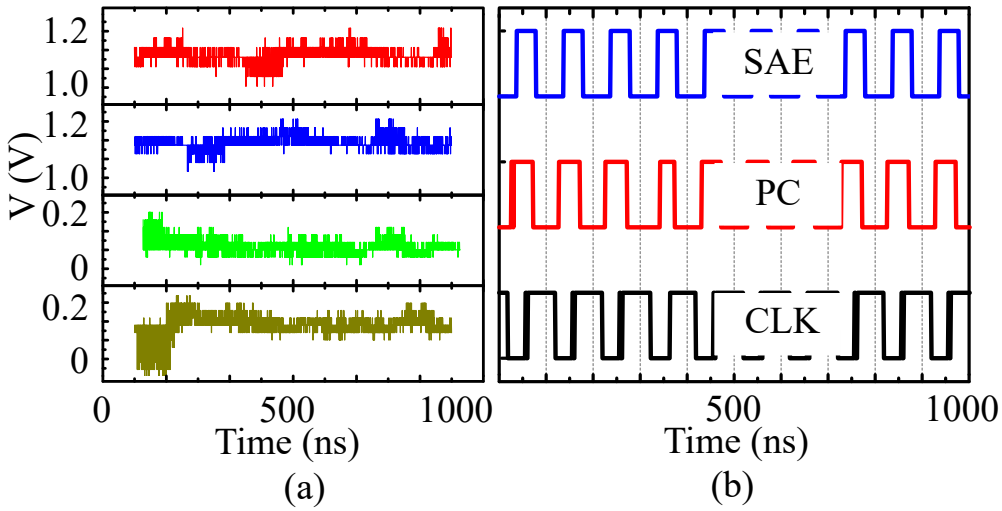

Figure 55: Read signals (a) Readout signals. (b) Read control signals.

15 (b1111). X axis refers to the illustrated first eight rows of the memory array. We repeat sampling the eight 4-bit words and plot red points based on occurrence frequencies of the readout value: the deeper the red point is, the more frequently the corresponding output appears.

A low (high) VR results in reading the memory words as '0000 (1111)', which is consistent with our design expectation due to the generated high (low) reference level out of the functional range. The best configurations of VR and PTUNE are 1.15 V and 0.4 V, respectively, where the obtained output is identical to the pre-programmed data. It proved the functions of the adjustable reference cell and the SA, as well as the write circuitry. Some observed bits errors may be due to process variations or bit failures in the array. A detailed debugging on the bit errors is still ongoing.

Figure 57 shows the writing current while programming '0000' and '0011' into the STT-MRAM array. As expected, writing '1' consumes a larger current than writing '0'. It can be observed that the measured writing current drops slightly as the row index increases because of the longer routing path and hence the larger resistance shown at the output of the write driver.

Some important design specs are summarized in Table 12.

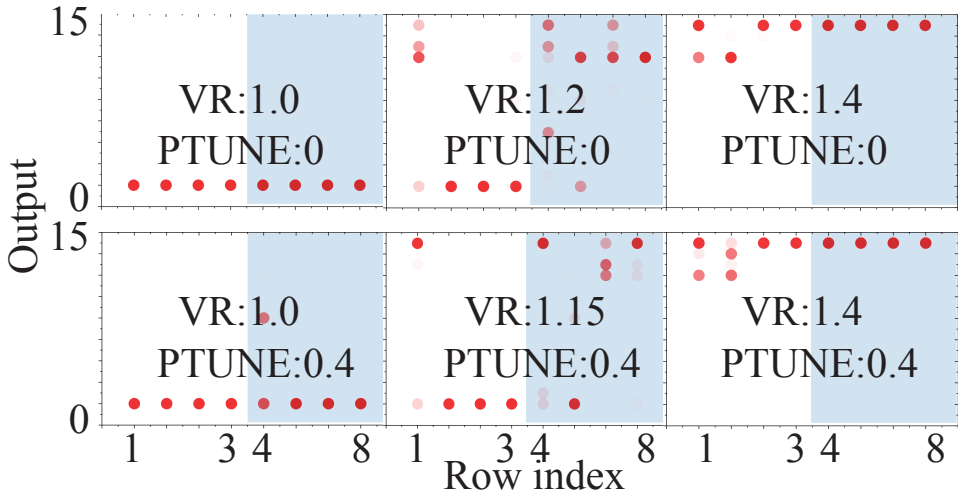

Figure 56: Relations between SA output and adjustable voltage VR and PTUNE.

6.5 SUMMARY

In this work, we design and fabricate the first 4Kb STT-MRAM test chip with NR-MTJ using 65nm technology. We also develop a novel fabrication process of NR-MTJ with outer and inner diameters of 200nm and 120nm, respectively, and demonstrate MRAM chip under commercial manufacturing facility. The testing results validate successful read and write functionalities of the chip, and show substantial write current reduction of NR-MTJ compared with conventional E-MTJ.

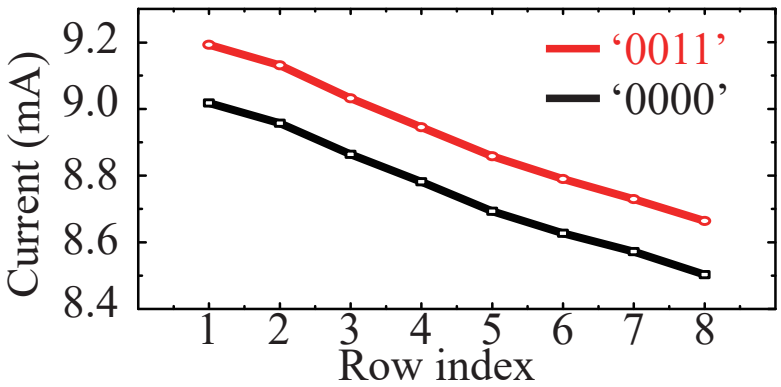

Figure 57: Write current with pattern '0000' and '0011'.

Table 12: Specs of the fabricated STT-MRAM

Organiztion	Operating Frequency	R/W power
1024 word×4	10MHz/5MHz	8.41mW/28.75mW*
Cell size (μm)	Array size (μm)	Chip size (μm)
1.82×1.76	85.88×78.40	970.28×900.33**

* Write power is pattern dependent and 28.75mW is the peak power with '1111' programming.

** There are four memory cores in a single chip. A design dimension of a memory core is 169.88×167.75 (μm^2)

7.0 CONCLUSIONS

In this work, we have proposed and evaluated several circuit and architectural level optimization method for STT-RAM. First of all, a probabilistic design method is proposed to reduce the write error of STT-RAM while maintaining a low performance and energy overhead. Secondly, we proposed different types of STT-RAM cell that support dual-port access and evaluated their performance and area cost. Moreover, we investigate the new types of Cell design for MLC STT-RAM and proposed architecture level solution to reduce the access latency and energy cost of MLC STT-RAM. Finally, we built a test chip that can be reconfigurable and support different type of MTJs. We have proved with circuit and architecture level co-optimization, reliability, functionality and storage density can be significantly improved over the conventional design, and STT-RAM can be truly adopted as a great candidate for universal memory.

BIBLIOGRAPHY

[1] M. H. Kryder and C. S. Kim, "After hard drives - what comes next?" *IEEE Transactions on Magnetics*, vol. 45, no. 10, pp. 3406–3413, Oct 2009.

[2] D. L. Lewis and H. H. S. Lee, "Architectural evaluation of 3d stacked rram caches," in *3D System Integration, 2009. 3DIC 2009. IEEE International Conference on*, Sept 2009, pp. 1–4.

[3] J. C. Mogul, E. Argollo, M. Shah, and P. Faraboschi, "Operating system support for nvm+dram hybrid main memory," in *Proceedings of the 12th Conference on Hot Topics in Operating Systems*, ser. HotOS'09, 2009, pp. 14–14.

[4] M. K. Qureshi, V. Srinivasan, and J. A. Rivers, "Scalable high performance main memory system using phase-change memory technology," in *Proceedings of the 36th Annual International Symposium on Computer Architecture*, ser. ISCA '09, 2009, pp. 24–33.

[5] B. M. Rogers, A. Krishna, G. B. Bell *et al.*, "Scaling the bandwidth wall: Challenges in and avenues for cmp scaling," in *ACM International Symposium on Computer Architecture (ISCA)*, 2009, pp. 371–382.

[6] "The International Technology Roadmap for Semiconductors," http://www.itrs.net, 2015.

[7] M. Hosomi *et al.*, "A novel nonvolatile memory with spin torque transfer magnetization switching: Spin-RAM," in *International Electron Devices Meeting (IEDM)*, 2005, pp. 459–462.

[8] T. Kawahara *et al.*, "2 Mb SPRAM (Spin-Transfer Torque RAM) with bit-by-bit bidirectional current write and parallelizing-direction current read," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 43, no. 1, pp. 109–120, 2008.

[9] T. Kawahara, "Scalable Spin-Transfer Torque RAM technology for normally-off computing," *IEEE Design Test of Computers*, vol. 28, no. 1, pp. 52–63, 2011.

[10] H.-S. Wong, S. Raoux, S. Kim *et al.*, "Phase change memory," *Proceedings of the IEEE*, vol. 98, no. 12, pp. 2201–2227, Dec 2010.

[11] B. Govoreanu, G. Kar, Y. Chen *et al.*, "10x10nm2 hf/hfox crossbar resistive ram with excellent performance, reliability and low-energy operation," in *IEEE International Electron Devices Meeting (IEDM)*, Dec 2011, pp. 31.6.1–31.6.4.

[12] G. Sun *et al.*, "A novel architecture of the 3D stacked MRAM L2 cache for CMPs," in *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2009, pp. 239–249.

[13] X. Wu *et al.*, "Hybrid Cache Architecture with Disparate Memory Technologies," in *international symposium on Computer architecture (ISCA)*, 2009, pp. 34–45.

[14] W. Xu *et al.*, "Design of Last-Level On-Chip Cache Using Spin-Torque Transfer RAM (STT-RAM)," *IEEE Transactions on Very Large Scale Integration Systems (TVLSI)*, vol. 19, no. 3, pp. 483 –493, 2011.

[15] [Online]. Available: http://www.everspin.com/spinTorqueMRAM.php

[16] B. Zhao *et al.*, "Architecting a common-source-line array for bipolar non-volatile memory devices," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2012, pp. 1451–1454.

[17] D. Lee *et al.*, "High-performance low-energy STT MRAM based on balanced write scheme," in *ACM/IEEE international symposium on Low power electronics and design (ISLPED)*, 2012, pp. 9–14.

[18] L. Liu *et al.*, "Spin-torque switching with the giant spin hall effect of tantalum," *Science*, vol. 336, no. 6081, pp. 555–558, 2012.

[19] C. Lin *et al.*, "45nm low power CMOS logic compatible embedded STT MRAM utilizing a reverse-connection 1t/1mtj cell," in *IEEE International Electron Devices Meeting (IEDM)*, 2009, pp. 1–4.

[20] X. Wang *et al.*, "Thermal fluctuation effects on spin torque induced switching: mean and variations," *Journal of Applied Physics*, vol. 103, no. 3, p. 034507, 2008.

[21] S. Choi, T. Na, J. Kim *et al.*, "Corner-aware dynamic gate voltage scheme to achieve high read yield in stt-ram," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. PP, no. 99, pp. 1–10, 2016.

[22] T. Na, J. Kim, B. Song *et al.*, "An offset-tolerant dual-reference-voltage sensing scheme for deep submicrometer stt-ram," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 4, pp. 1361–1370, 2016.

[23] F. B. Yahya, M. M. Mansour, J. Tschanz *et al.*, "Designing low-vth stt-ram for write energy reduction in scaled technologies," in *International Symposium on Quality Electronic Design*, 2015, pp. 5–9.

[24] H. Farkhani, A. Peiravi, and F. Moradi, "Low-energy write operation for 1t-1mtj stt-ram bitcells with negative bitline technique," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 4, pp. 1593–1597, 2016.

[25] S. Chung *et al.*, "Fully integrated 54nm STT-RAM with the smallest bit cell dimension for high density memory application," in *IEEE International Electron Devices Meeting (IEDM)*, 2010, pp. 12.7.1–12.7.4.

[26] X. Dong *et al.*, "Circuit and microarchitecture evaluation of 3D stacking magnetic RAM (MRAM) as a universal memory replacement," in *ACM/IEEE Design Automation Conference (DAC)*, 2008, pp. 554–559.

[27] P. Zhou *et al.*, "Energy reduction for stt-ram using early write termination," in *Proceedings of the International Conference on Computer-Aided Design (ICCAD)*, 2009, pp. 264–268.

[28] M. P. Komalan, C. Tenllado, J. I. G. Perez *et al.*, "System level exploration of a stt-mram based level 1 data-cache," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2015, pp. 1311–1316.

[29] Q. Li, Y. He, J. Li *et al.*, "Compiler-assisted refresh minimization for volatile stt-ram cache," *IEEE Transactions on Computers*, vol. 64, no. 8, pp. 2169–2181, 2015.

[30] W.-K. Cheng, Y.-H. Ciou, and P.-Y. Shen, "Architecture and data migration methodology for l1 cache design with hybrid sram and volatile stt-ram configuration," *Microprocessors and Microsystems*, vol. 42, pp. 191 – 199, 2016.

[31] M. H. Samavatian, M. Arjomand, R. Bashizade *et al.*, "Architecting the last-level cache for gpus using stt-ram technology," *ACM Transactions on Design Automation of Electronic Systems*, vol. 20, no. 4, pp. 55:1–55:24, 2015.

[32] G. Li, X. Chen, G. Sun *et al.*, "A stt-ram-based low-power hybrid register file for gpgpus," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015, pp. 1–6.

[33] J. Wang and Y. Xie, "A write-aware sttram-based register file architecture for gpgpu," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 12, no. 1, pp. 6:1–6:12, 2015.

[34] T. Ishigaki *et al.*, "A multi-level-cell spin-transfer torque memory with series-stacked magnetotunnel junctions," in *IEEE Symposium on VLSI Technology (VLSIT)*, 2010, pp. 47–48.

[35] X. Lou *et al.*, "Demonstration of multilevel cell spin transfer switching in mgo magnetic tunnel junctions," *Applied Physics Letters*, vol. 93, no. 24, p. 242502, 2008.

[36] F. Sampaio, M. Shafique, B. Zatt *et al.*, "Approximation-aware multi-level cells stt-ram cache architecture," in *International Conference on Compilers, Architecture and Synthesis for Embedded Systems*, 2015, pp. 79–88.

[37] X. Liu, M. Mao, X. Bi *et al.*, "An efficient stt-ram-based register file in gpu architectures," in *Asia and South Pacific Design Automation Conference*, 2015, pp. 490–495.

[38] L. Jiang *et al.*, "Constructing large and fast multi-level cell STT-MRAM based cache for embedded processors," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2012, pp. 907–912.

[39] Y. Zhang *et al.*, "Multi-level cell STT-RAM: Is it realistic or just a dream?" in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2012, pp. 526–532.

[40] E. Ipek *et al.*, "Dynamically replicated memory: building reliable systems from nanoscale resistive memories," in *Proceedings of the fifteenth edition of ASPLOS on Architectural support for programming languages and operating systems*, 2010, pp. 3–14.

[41] S. Schechter *et al.*, "Use ecp, not ecc, for hard failures in resistive memories," in *Proceedings of the 37th Annual International Symposium on Computer Architecture(ISCA)*, 2010, pp. 141–152.

[42] H. Sun *et al.*, "Design techniques to improve the device write margin for mram-based cache memory," in *GLSVLSI*, 2011, pp. 97–102.

[43] W. Zhao *et al.*, "New generation of Predictive Technology Model for sub-45 nm early design exploration," *IEEE Transactions on Electron Devices*, vol. 53, no. 11, pp. 2816–2823, 2006.

[44] Y. Zhang *et al.*, "Stt-ram cell design optimization for persistent and non-persistent error rate reduction: A statistical design view," in *International Conference on Computer Aided Design (ICCAD)*, 2011, pp. 471–477.

[45] J. C. Slonczewski, "Currents, torques, and polarization factors in magnetic tunnel junctions," *Phys. Rev. B*, vol. 71, no. 2, p. 024411, 2005.

[46] X. Bi *et al.*, "Spintronic memristor based temperature sensor design with cmos current reference," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2012, pp. 1301 –1306.

[47] X. Dong *et al.*, "Nvsim: A circuit-level performance, energy, and area model for emerging nonvolatile memory," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 31, no. 7, pp. 994–1007, 2012.

[48] C. Bienia, "Benchmarking modern multiprocessors," Ph.D. dissertation, Princeton University, January 2011.

[49] K. Nii *et al.*, "A 90nm dual-port SRAM with 2.04um^2 8t-thin cell using dynamically-controlled column bias scheme," in *IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 1, 2004, pp. 508–543.

[50] T. Suzuki *et al.*, "A stable 2-port SRAM cell design against simultaneously read/write-disturbed accesses," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 43, no. 9, pp. 2109–2119, 2008.

[51] S. Ishikura *et al.*, "A 45 nm 2-port 8T-SRAM using hierarchical replica bitline technique with immunity from simultaneous r/w access issues," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 43, no. 4, pp. 938–945, 2008.

[52] J. Kim *et al.*, "The v2.0+edr bluetooth soc architecture for multimedia," *IEEE Transactions on Consumer Electronics (TCE)*, vol. 52, no. 2, pp. 436–444, 2006.

[53] T. Shiota *et al.*, "A 51.2 gops 1.0 gb/s-dma single-chip multi-processor integrating quadruple 8-way vliw processors," in *IEEE International Solid-State Circuits Conference*, vol. 1, 2005, pp. 194–593.

[54] M. Nakajima *et al.*, "Homogenous dual-processor core with shared l1 cache for mobile multimedia soc," in *IEEE Symposium on VLSI Circuits (VLSIC)*, 2007, pp. 216–217.

[55] K. Nii *et al.*, "Synchronous ultra-high-density 2RW dual-port 8T-SRAM with circumvention of simultaneous common-row-access," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 44, no. 3, pp. 977–986, 2009.

[56] "SMIC 65nm logic low leakage & RF Cadence PDK," http://service.smics.com, 2011.

[57] F. Ishihara *et al.*, "Level conversion for dual-supply systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems (TVLSI)*, vol. 12, no. 2, pp. 185–195, 2004.

[58] S. Gupta *et al.*, "Layout-aware optimization of STT MRAMs," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2012, pp. 1455–1458.

[59] Y.-M. Chang, Y.-H. Chang, T.-W. Kuo, Y.-C. Li, and H.-P. Li, "Achieving slc performance with mlc flash memory," in *Proceedings of the 52nd Annual Design Automation Conference*. ACM, 2015, p. 192.

[60] J. Cheon, I. Lee, C. Ahn *et al.*, "Non-resistance metric based read scheme for multi-level pcram in 25 nm technology," in *IEEE Custom Integrated Circuits Conference (CICC)*. IEEE, 2015, pp. 1–4.

[61] Y. Chen, W.-F. Wong, H. Li *et al.*, "On-chip caches built on multilevel spin-transfer torque ram cells and its optimizations," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 9, no. 2, pp. 16:1–16:22, 2013.

[62] R. Dorrance *et al.*, "Scalability and design-space analysis of a 1t-1mtj memory cell for stt-rams," *IEEE Transactions on Electron Devices*, vol. 59, no. 4, pp. 878–887, 2012.

[63] A. R. Alameldeen *et al.*, "Adaptive cache compression for high-performance processors," in *IEEE Annual International Symposium on Computer Architecture (ISCA)*, 2004, pp. 212–223.

[64] INTEL, 2014. [Online]. Available: http://ark.intel.com/products/family/29035

[65] Macsim, 2014. [Online]. Available: http://comparch.gatech.edu/hparch/software.html

[66] A. N. Udipi *et al.*, "Rethinking dram design and organization for energy-constrained multi-cores," in *ACM International Symposium on Computer Architecture (ISCA)*, 2010, pp. 175–186.

[67] C. D. Spradling, "Spec cpu2006 benchmark tools," *SIGARCH Computer Architecture News*, no. 1, pp. 130–134.

[68] N. Edel, D. Tuteja, E. Miller, and S. Brandt, "Mramfs: a compressing file system for non-volatile ram," in *International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*, 2004, pp. 596–603.

[69] M. Durlam *et al.*, "A 1-mbit mram based on 1t1mtj bit cell integrated with copper interconnects," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 5, pp. 769–773, 2003.

[70] http://www.crocus-technology.com.

[71] X. Han *et al.*, "A novel design and fabrication of magnetic random access memory based on nano-ring-type magnetic tunneljunctions," *Journal of Materials Sciences and Technology*, vol. 23, no. 03, p. 304, 2007.

[72] X. F. Han, Z. C. Wen, and H. X. Wei, "Nanoring magnetic tunnel junction and its application in magnetic random access memory demo devices with spin-polarized current switching," *Journal of Applied Physics*, vol. 103, no. 7, 2008.

[73] F. Q. Zhu, G. W. Chern, O. Tchernyshyov, X. C. Zhu, J. G. Zhu, and C. L. Chien, "Magnetic bistability and controllable reversal of asymmetric ferromagnetic nanorings," *Phys. Rev. Lett.*, vol. 96, p. 027205, 2006. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.96.027205

[74] Z. Diao *et al.*, "Spin-transfer torque switching in magnetic tunnel junctions and spin-transfer torque random access memory," *Journal of Physics: Condensed Matter*, vol. 19, no. 16, p. 165209, 2007.

[75] W. Reohr *et al.*, "Memories of tomorrow," *Circuits and Devices Magazine, IEEE*, vol. 18, no. 5, pp. 17–27, 2002.

[76] Y. Zhang, X. Wang, H. Li, and Y. Chen, "Stt-ram cell optimization considering mtj and cmos variations," *IEEE Transactions on Magnetics*, vol. 47, no. 10, pp. 2962–2965, 2011.