# Design and Optimization of Resistive RAM-based Storage and Computing Systems

by

Manqing Mao

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2019 by the
Graduate Supervisory Committee:

Chaitali Chakrabarti, Chair
Shimeng Yu
Yu Cao
Umit Ogras

ARIZONA STATE UNIVERSITY

May 2019

# ABSTRACT

The Resistive Random Access Memory (ReRAM) is an emerging non-volatile memory technology because of its attractive attributes, including excellent scalability (< 10 nm), low programming voltage (< 3 V), fast switching speed (< 10 ns), high OFF/ON ratio (> 10), good endurance (up to $10^{12}$ cycles) and great compatibility with silicon CMOS technology [1]. However, ReRAM suffers from larger write latency, energy and reliability issue compared to Dynamic Random Access Memory (DRAM). To improve the energy-efficiency, latency-efficiency and reliability of ReRAM storage systems, a low cost cross-layer approach that spans device, circuit, architecture and system levels is proposed.

For 1T1R 2D ReRAM system, the effect of both retention and endurance errors on ReRAM reliability is considered. Proposed approach is to design circuit-level and architecture-level techniques to reduce raw Bit Error Rate significantly and then employ low cost Error Control Coding to achieve the desired lifetime.

For 1S1R 2D ReRAM system, a cross-point array with "multi-bit per access" per subarray is designed for high energy-efficiency and good reliability. The errors due to cell-level as well as array-level variations are analyzed and a low cost scheme to maintain reliability and latency with low energy consumption is proposed.

For 1S1R 3D ReRAM system, access schemes which activate multiple subarrays with multiple layers in a subarray are used to achieve high energy efficiency through activating fewer subarray, and good reliability is achieved through innovative data organization.

Finally, a novel ReRAM-based accelerator design is proposed to support multiple Convolutional Neural Networks (CNN) topologies including VGGNet, AlexNet and ResNet. The multi-tiled architecture consists of 9 processing elements per tile, where each tile

implements the dot product operation using ReRAM as computation unit. The processing elements operate in a systolic fashion, thereby maximizing input feature map reuse and minimizing interconnection cost. The system-level evaluation on several network benchmarks show that the proposed architecture can improve computation efficiency and energy efficiency compared to a state-of-the-art ReRAM-based accelerator.

DEDICATION

To all my families.

Especially my grandfather,

who left his fingerprint of grace on my life.

# ACKNOWLEDGMENTS

Conducting a PhD research is harder than I thought and more rewarding than I could have ever imagined. None of this would have been done without my advisor, Dr. Chaitali Chakrabarti, to whom I would like to pay special thankfulness, warmth and appreciation. She never stopped me; she assisted me at every point to accomplish my goal and encouraged me when I felt upset. With her endless support, I learned how to define a research problem, find a good solution to it, and finally publish the results. Dr. Chakrabarti is always eager to listen to the little problems and roadblocks and has infinite patience for research exploration. She is someone you will instantly love and never forget once you meet her.

I also want to thank all the rest of my committee members – Dr. Shimeng Yu, Dr. Yu Cao and Dr. Umit Ogras, whose invaluable guidance and advice turned my research into a success. Thanks to everyone in the ECEE department who helped me so much. Special thanks to Toni Mengert, the ever-patient and super-efficient graduate advisor I could ever imagine. I would also like to thank the National Science Foundation for funding my research.

To all my lab mates – Siyuan, Jian, Shunyao, Jingtao, Sizhe, Jingyi and Yan -- I have had the opportunity to study with, I want to say thank you for being such great company and inspiration in the past 6 years. I feel super lucky to join the lab and to be a part of you. I will never forget the many, many great moments of hanging out, sharing yummy food, tea and playing board games. You guys always stood by me during my struggles and my successes. Thank you all! I also want to thank my best friends – Yahuan and Jinzong, who always cooked the most delicious desserts in the world for me and ignited my life in the last 1 year. They were always there to cheer me up. That is true friendship!

I would like to express my deepest gratitude to my Mom, Dad, sister and family members, without whom I am a nobody. They not only assisted me financially but also extended their love and support morally and emotionally. Thanks to my Mom who was always the first one unconditionally supporting me and my Dad who spent sleepless nights helping me edit papers. My sister has been my best friend all my life and I love her dearly and thank her for all her advice and support. I am eternally grateful to my dear grandparents, who took in an extra mouth to feed when they did not have to. Sincerest thanks to my grandfather, although no longer with me, taught me about discipline, love, manners, respect and so much more that has helped me to succeed. He is always in my heart and in my mind.

Finally, I would like to show my gratitude to Feng, my husband who is also my best DOTA2 friend. From taking care of me in daily life, to giving me study advice, to carrying me in the game, to keeping the chocolate out of my desk so I could study, he played a very important role in my PhD research. Together we both learned a lot about study, about life and strengthened our commitment to each other to live life to the fullest. Thank you so much, dear!

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

# CHAPTER 1

# INTRODUCTION

The increasing gap between processor and memory speeds has rendered design of memory systems an increasingly important part of computer-system design. Memory-hierarchy parameters affect system performance significantly more than processor parameters (e.g. they are responsible for 2× - 10× changes in execution time, as opposed to 2% - 10%), making it essential for memory designers to improve the performance of memory system.

Semiconductor memories can be divided into two major categories: Random Access Memories (RAM), and Read Only Memories (ROM). RAM loses its content when power supply is turned off. Examples include static random access memory (SRAM) and dynamic random access memory (DRAM). Such memories typically have very low latency and are used as primary storage. On the other hand, ROM virtually holds data forever but it cannot be altered. Examples include FLASH memory, electrically erasable programmable READ-only memory (EEPROM). A third category lies in between, Non-Volatile Memories (NVM), whose content can be electrically altered but it is also preserved when power supply is switched off. These are more flexible than the original ROM. Examples include Phase Change RAM (PRAM), magnetic RAM (MRAM) and resistive RAM (ReRAM). Since these memories have large access time, they are typically used in high levels of memory hierarchy. However, recently, new types of nonvolatile memories, such as spin torque transfer RAM (STT-RAM) and ReRAM have been shown to have timing performance that is comparable to traditional volatile memory and thus have the potential to be used at low levels of memory hierarchy.

The different types of nonvolatile memory have very different data storage mechanisms. STT-MRAM relies on difference in resistance between the parallel configuration (logic state

'1') and anti-parallel configuration (logic state '0') of mutual magnetic layers in a thin tunneling insulator layer. PCRAM relies on chalcogenide material to switch between the crystalline phase (logic state '1') and the amorphous phase (logic state '0'), while ReRAM relies on the formation (logic state '1') and the rupture (logic state '0') of conductive filaments in the insulator between two electrodes. Due to the different underlying physics, the device characteristics of these different NVM technologies are also different. Table 1.1 compares the typical device characteristics between the emerging memory technologies and the mainstream memory technologies [1]. In general, nonvolatile memories have higher cell density, but they also have higher latency, as shown in Table 1.1. Since higher memory layers require larger storage sizes and have low access frequency, use of nonvolatile memories in main memory or hard disk is cost effective. For instance, compared to SRAM, STT-MRAM has the advantage of smaller cell area, while maintaining low programming voltage. It has fast write/read speed and long endurance, making it attractive for embedded memory on chip, e.g., the last-level cache [2]. FLASH memories are dominant in the storage market due to their high storage density and low storage cost per cell. ReRAM has lower programming voltage and faster write/read speed compared to FLASH. So ReRAM is expected to replace the NOR FLASH for code storage and more ambitiously to replace NAND FLASH as data storage [3]. Compared to other emerging memory technologies, ReRAM has > 10× reduction in the write current over Phase Change Memory (PCM), and > 5× reduction in the cell area over Spin Torque Transfer (STT) magnetic RAM [1]. Compared to Dynamic Random Access Memory (DRAM), ReRAM has lower standby power due to its non-volatility, making it a viable technology to replace DRAM in main memory systems. Thus, in this thesis, we focus on the optimization of ReRAM-based storage and computing systems.

The basic ReRAM cell is a two-terminal variable resistor that operates in a high resistance state (HRS) or OFF-state and a low resistance state (LRS) or ON-state. The switching between the low resistance state (LRS) or ON-state and high resistance state (HRS) or OFF-state is caused by the formation and rupture of the conductive filaments (CFs) in the oxides (or other insulating material) between the two electrodes [1]. We refer to the switching from OFF-state to ON-state as SET and the switching from ON-state to OFF-state as RESET.

TABLE 1.1 [1]
DEVICE CHARACTERISTICS OF MAINSTREAM AND EMERGING MEMORY TECHONOLOGIES

| | **Mainstream Memories** | | | | **Emerging Memories** | | |
|---|---|---|---|---|---|---|---|
| | **SRAM** | **DRAM** | **FLASH** | | **STT-MRAM** | **PCRAM** | **ReRAM** |
| | | | **NOR** | **NAND** | | | |
| **Cell Area** | > $100F^2$ | $6F^2$ | $10F^2$ | < $4F^2$ (3D) | 6 ~ $20F^2$ | 4 ~ $20F^2$ | < $4F^2$ (3D) |
| **Multi-bit** | 1 | 1 | 2 | 3 | 1 | 2 | 2 |
| **Voltage** | < 1V | < 1V | > 10V | > 10V | < 2V | < 3V | < 3V |
| **Read Time** | ~ 1ns | ~ 10ns | ~ 50ns | ~ 10μs | < 10ns | < 10ns | < 10ns |
| **Write Time** | ~ 1ns | ~ 10ns | 10μs ~ 1ms | 100μs ~ 1ms | < 5ns | ~ 50ns | < 10ns |
| **Retention** | N/A | ~ 64ms | > 10y | > 10y | > 10y | > 10y | > 10y |
| **Endurance** | > $10^6$ | > $10^{16}$ | > $10^5$ | > $10^4$ | > $10^{15}$ | > $10^9$ | $10^6$~ $10^{12}$ |
| **Write Energy/bit** | ~ fJ | ~10fJ | 100pJ | ~10fJ | ~0.1pJ | ~10pJ | ~0.1pJ |
| F: Feature size of the lithography, and the energy estimation is on the cell-level (not the array-level) | | | | | | | |

ReRAM has several attractive features, including excellent scalability (<10 nm), low programming voltage (<3 V), fast switching speed (<10 ns), large resistance OFF/ON ratio (>10×), long retention (10 years at 85 °C), good endurance (up to $10^{12}$ cycles), and great

compatibility with silicon CMOS technology [1]. However, ReRAM suffers from reliability degradation due to process variations, structural limits and material property shift. In order for it to be a main-stream memory technology, the minimum number of write operations (endurance capability) and the ability of keeping unaltered the stored information for years and years (retention capability) must be guaranteed.

In the ReRAM cross-point architecture, the bit-line (BL) and the word-line (WL) are perpendicular to each other and memory cells are sandwiched in between. Such a structure has $4F^2$ cell area, where F is the lithography technology node. Unfortunately, the cross-point array suffers from sneak path and IR drop, resulting in lower reliability [4]. To reduce the effect of sneak paths during memory cell operation, a highly nonlinear, bidirectional selector device (1S) or a transistor (1T) is serially connected with each bipolar resistor (1R) and the corresponding is referred to as 1S1R or 1T1R cell configuration [1]. Of the two types of ReRAM array architectures, the cross-point 1S1R array architecture has higher integration density compared with the 1T1R architecture [1]. On the other hand, 1T1R eliminates the sneak path current problem of cross-point array, resulting in higher reliability. 1S1R has almost the same area as the cross-point (= $4F^2$) structure since the selector device is vertically stacked with the ReRAM cell. 1T1R ReRAM cell has the same density as 1T1C DRAM cell, featuring $6F^2$ cell by using the contact borderless layout. Both structures suffer from device variations, endurance and retention issues at the device level; and IR drop at the array level. Compared to DRAM, ReRAM has larger write latency, energy and lower write endurance. In this thesis, we propose cross-layer techniques which span device level, circuit level and system level to improve energy-efficiency, latency-efficiency and reliability of 2D and 3D ReRAM-based storage systems. We also propose the multi-tile ReRAM-based accelerator framework for

supporting multiple CNN topologies that maximizes on-chip data reuse and reduces on-chip bandwidth to minimize energy consumption due to data movement.

## 1.1 1T1R 2D Array Architecture

ReRAM can be organized into the 1-transistor-1-resistor (1T1R) array architecture, similar to the 1-transistor-1-capacitor (1T1C) in array in DRAM. However, several challenges need to be overcome before incorporating 1T1R ReRAM into main memory. These include the following: (1) Large write latency: Write in 1T1R ReRAM is typically slower than that in DRAM, thereby compromising the system performance; (2) Low write endurance: ReRAM-based main memory is only able to sustain $10^8$ to $10^{12}$ programming cycles as opposed to > $10^{16}$ for DRAM-based main memory; and (3) High write energy: Programming energy of ReRAM is higher than that of DRAM due to the relatively larger current (~10 μA) required to switch the state.

There have been a few studies on latency/energy/reliability of ReRAM. At the circuit level, the write schemes in [5-6] exploit the differences in write latencies due to cell to cell variations to cut off the voltage pulse [5] or the SET current [6] earlier than the worst case time. By implementing the cut off at different times for individual cells, the energy consumption is reduced. For reliability, a scheme to retain the endurance capability of the 1T1R cell after NPC of $10^{10}$ cycles is proposed in [7]. At the system level, multi-level design of ReRAM spanning array, bank and chip levels is proposed in [8] and a design space exploration scheme for determining the array size, bank size for improving the latency and energy of a 1-diode-1-resistor (1D1R) ReRAM system is provided in [9].

***Contributions:*** We present cross-layer techniques to improve reliability of 1T1R array with minimum latency and energy cost. At the circuit level, we show how voltage settings (pulse amplitude and pulse width) of word-line (WL), bit-line (BL), and source-line (SL) can be used to lower latency, lower energy consumption and improve reliability. We also show how appropriate choice of voltage settings can help reduce retention and endurance errors while minimizing energy. At the architecture level, we propose a new bit-flipping scheme that helps reduce the Bit Error Rate (BER) even further. We show how application of circuit-level and architecture-level techniques makes it possible to achieve a lifetime of 10 years with a simple BCH code. Finally, we evaluate the system-level performances of a 1GB ReRAM and 1GB DRAM memory system using CACTI and GEM5. Simulation results using SPEC CPU INT 2006 and DaCapo-9.12 benchmarks show that the proposed ReRAM based main memory can improve Instruction Per Cycle (IPC) by 5.2% and energy by up to 72% compared to a DRAM memory system. This work appeared in [10] – [12].

## 1.2 1S1R 2D Array Architecture

The 1S1R structure enables design of a largescale cross-point array by cutting off the sneak path current of the half-selected and unselected cells. Compared with 1T1R, 1S1R has smaller area with cross-point (= $4F^2$) since the selector device is vertically stacked with the ReRAM cell. However, the 1S1R array still suffers from IR drop along the interconnect wires. The IR drop problem becomes significant when the WL and BL wire width scales to sub-50-nm regime where the interconnect resistivity drastically increases due to the electron surface scattering [1]. During write operation, the farthest cell from the driver has insufficient voltage drop, resulting in unsuccessful write.

Most of the prior work on ReRAM cross-point array focused on device and circuit issues [13]–[19]. These include selector and ReRAM cell level designs that improve read/write margins [13]–[18]. There has also been work on cross-point array organization as well as array size evaluation with respect to energy consumption and reliability. However, most of the previous work was based on "single bit per read/write" per subarray [13]–[17], a scheme which incurred large power consumption since multiple subarrays have to be activated at a time to meet the I/O bandwidth.

***Contributions:*** In order to improve the performance of ReRAM system, we focus on an access scheme where a data line is parallelly accessed from multiple subarrays with multi-bits accessed per subarray. A direct implementation of such a scheme has high energy efficiency but lower reliability compared with a single bit per subarray baseline scheme. So we propose a low cost multilayer approach to improve energy-efficiency of multi-bits per access scheme without compromising reliability. At the cell level, we show how proper choices of bit-line and source-line voltage and SET recovery help reduce error rate by ten times. At the system level, we propose a new rotated multi-array access scheme where the average error rate of every accessed data line is one order of magnitude lower than the worst case, making it possible to achieve block failure rate of $10^{-10}$ with a simple BCH t = 4 code. We show that for a 1 GB 1S1R ReRAM, the proposed approach can reduce energy by 41% with 2% extra area while maintaining latency and reliability compared with the baseline system. This work appeared in [20].

## 1.3 1S1R 3D Array Architecture

The key challenge in competing with NAND flash for storage class memory is ReRAM's lower integration density and thus higher cost per bit. To reduce cost per bit, 3D cross-point

ReRAM architecture has been widely studied. By simply stacking the cross-point ReRAM cells layer by layer [21-24], the integration density of ReRAM can be increased. In the stacked layer approach referred to as 3-D horizontal ReRAM (3D-HRAM) [25], [26], the adjacent layers share the word lines (WLs) and bitlines (BLs). An alternative to 3D-HRAM is the 3-D vertical ReRAM (3D-VRAM), which has higher cost efficiency but suffers from several fabrication-related issues, e.g., high aspect-ratio pillar etching for multiple metal/dielectric stacks, selector integration on the sidewall, etc. Since 3D-HRAM is a more mature technology with two-layer chip-scale demonstrations [21-24], we focus on this 3-D structure in our investigation.

***Contributions:*** We present access schemes which activate multiple subarrays with multiple layers in a subarray to achieve high energy efficiency through activating fewer subarray and good reliability through innovative data organization. We propose two low-cost access schemes [namely, multilayer access scheme (MAS)-I and MAS-II] which enable multilayer programming but differ in the number of activated layers (NL) and hence differ in energy efficiency. To improve reliability, we propose to distribute data across subarrays as well as along the layers of a subarray such that the error characteristics of all accessed data lines are the same. At the system level, we proposed to use error correcting codes such as Bose, Chaudhuri, and Hocquenghem (BCH) codes with different strengths so that all competing systems have the same reliability. We show that for a 1-GB 3-D horizontal 1S1R ReRAM system with an I/O width of 64 bits, the NB = 16, NL = 4 system based on MAS-I that utilizes BCH t = 6 code consumes the lowest energy with 33% lower energy consumption compared to the baseline system where only one layer is activated at a time in [27].

## 1.4 CNN Accelerator using 1T1R 2D Array

Deep convolutional neural networks (CNN) are increasingly being used in a wide range of domains from computer vision to natural language processing to robotics to gaming [28-31]. These networks achieve very high accuracy but at the price of large computational complexity [32]. A recent study demonstrates that the bottleneck for these networks is the number of memory accesses. Nearly 60G DRAM accesses are required by VGG-16 [33] to classify one image, resulting in several orders of magnitude higher memory energy compared to computation energy [34].

Processing-in-memory (PIM) is an efficient technique to reduce the number of memory accesses through integration of the computations and storage. Emerging non-volatile memory (eNVM) technologies, such as resistive random access memory (ReRAM) [35-37] and phase change memory [38], are more promising PIM candidates due to their compatibility with the CMOS back-end-of-line process. The PIM accelerator designs in [35, 36] use the conventional crossbar architecture where writing the weights into the eNVM cells is a non-trivial task due to the sneak paths.

***Contributions:*** we propose MAX$^2$, a multi-tile ReRAM accelerator framework for supporting multiple CNN topologies including VGG-19 [33], AlexNet [29] and ResNet [81]. MAX2 maximizes on-chip data reuse and reduces on-chip bandwidth to minimize energy consumption due to data movement. Building upon the fact that a large filter can be built with a stack of smaller (3×3) filters, we design every tile with 9 processing elements (PE). Each PE consists of multiple 1T1R ReRAM subarrays to compute the dot product. The PEs operate in a systolic fashion, thereby maximizing input feature map reuse and minimizing interconnection cost. MAX$^2$ chooses the data size granularity in the systolic array in

conjunction with weight duplication to achieve very high area utilization without requiring additional peripheral circuits. We provide an in-depth system-level evaluation of $MAX^2$ for VGGNet, ResNet and AlexNet-based benchmarks based on NeuroSim [88]. Simulation results show that for VGG-19, $MAX^2$ implemented with 1-bit weight and 1-bit activation can improve computation efficiency (TOPs/s/$mm^2$) by 2.5×, energy efficiency (TOPs/s/W) by 5.2× compared to a state-of-the-art ReRAM-based accelerator [35].

## 1.5 Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 describes our work on improving the latency, energy and reliability of ReRAM 1T1R system through cross-layer techniques, which span circuit, architecture and system levels. Chapter 3 is on improving the reliability of ReRAM 1S1R system. It first analyzes the effect of spatial variations and temporal variations on resistance distributions followed by a multi-layer approach and finally presents a system-level evaluation. Chapter 4 describes our approach on improving the reliability of 3D ReRAM 1S1R system, which suffers from worse reliability degradation compared to 2D systems. Chapter 5 describes our work on an ReRAM-based CNN accelerator to improve upon intra-layer processing by maximizing input feature map (IFM) data reuse, minimizing interconnection cost and reducing intra-layer bandwidth. Chapter 6 summarizes this thesis.

# CHAPTER 2

# IMPROVING RELIABILITY OF 1T1R RERAM SYSTEM

## 2.1 Introduction

The ReRAM device is a two-terminal variable resistor where the memory states are represented by a high resistance state (off state) and low resistance state (on-state). As shown in Fig. 2.1, the ReRAM memory cell has a capacitor-like structure composed of semiconducting or insulating material sandwiched between two metal electrodes (MIM structure) [4]. Due to the resistive switching phenomenon, the resistance of the cell can be set to a desired value by adjusting the characteristics of the voltage pulse. The physical mechanism of ReRAM relies on the formation (on-state) and the rupture (off-state) of conductive filaments composed of oxygen vacancies in the oxides between two electrodes. The switching from off-state to on-state is called SET, while the switching from on-state to off-state is called RESET.

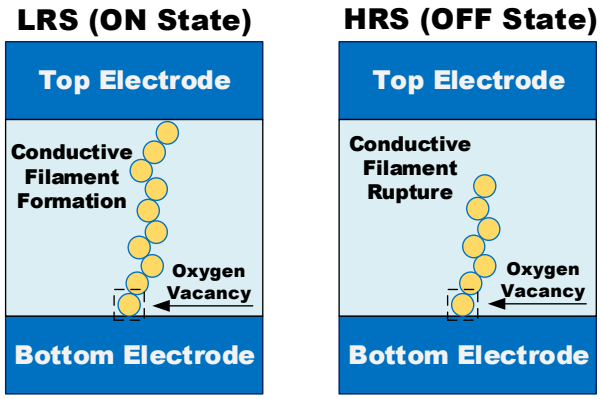



Fig. 2.1. Schematics of LRS and HRS (adapted from [1]).

ReRAM has poor reliability due to device variations, retention and endurance issues. In order that ReRAM be adopted as a main-stream memory technology, error control approaches have to be employed to address the reliability issues. Figure 2.2 gives an overview of proposed

scheme. At the circuit level, we choose WL, BL and SL voltage settings that enable the system to have high reliability and energy-efficiency. At the architecture level, we employ a bit-flipping technique to further reduce raw BER so that a (t = 2) BCH based scheme is sufficient to achieve 10-year lifetime.

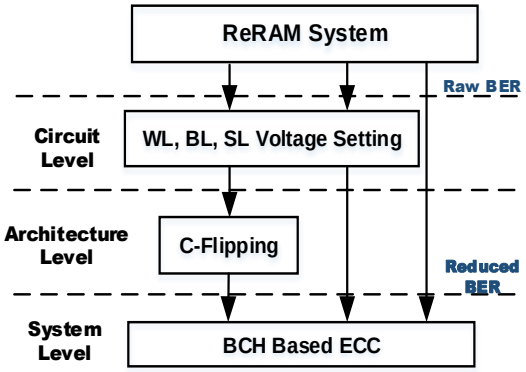



Fig. 2.2. Cross-layer techniques for improving reliability of ReRAM systems.

The rest of this chapter is organized as follows. In Section 2.2, we review 1T1R array programming schemes and reliability characteristics. Existing work has been summarized in Section 2.3. In Section 2.4, we show how WL, BL, and SL voltages can be chosen for high performance and low power. In Section 2.5, we show how to improve reliability with negligible latency and energy penalty by choosing proper voltage settings. This is followed by Section 2.6 where we present the new bit flipping technique to further reduce the bit error rate. In Section 2.7, we evaluate the proposed ReRAM systems with respect to IPC performance, energy and lifetime and compare their performance to a DRAM system. We conclude the chapter in Section 2.8.

## 2.2 Background

In general, there are two types of ReRAM array architectures. The first one is the cross-point architecture, where the bit-line (BL) and word-line (WL) are perpendicular to each other

and the memory cells are sandwiched in between. The cross-point architecture has $4F^2$ cell area and thus can achieve high integration density. However, the cross-point architecture suffers from interference among cells and the commonly known sneak path problem that limits the array size, increases the power consumption, and degrades the reliability [1]. The second architecture is the 1T1R array (Fig. 2.3), where each memory cell is in series with a cell selection transistor. The addition of a selection transistor helps isolate the selected cell from other unselected cells. Although it increases the minimum cell area to $6F^2$ (using similar DRAM-like design rules), 1T1R eliminates the sneak path current, thereby reducing the power consumption. Furthermore, it prevents READ disturbance from the other half-selected cells, thereby improving the reliability.

**Programming:** The conventional 1T1R design uses different WL voltages for SET and RESET. For example, for SET, a small WL voltage is applied to turn on the selection transistor, and BL voltage is applied to set the state; for RESET, a large WL voltage is applied to turn on the selection transistor so that the voltage drop on ReRAM cell can be compensated, and source line (SL) voltage is applied to reverse the current. Thus, in conventional WRITE, some cells are SET using one WL voltage and then the remaining cells are RESET using another WL voltage [7]. This two-step process results in high programming latency.

**Reliability:** The reliability of an ReRAM cell can be characterized by its retention and endurance characteristics. The ReRAM resistance may spontaneously drift even without voltage bias, thereby resulting in retention errors [41]. On the other hand, endurance is a function of the OFF/ON resistance ratio which is defined as the ratio of high resistance over low resistance. This ratio is a function of WL, BL and SL voltages. Use of strong WL, SL(BL) voltage pulses for RESET(SET) helps in boosting the OFF/ON ratio. OFF/ON ratio reduces

with the number of programming cycles, resulting in endurance errors. The best reported ReRAM endurance is up to $10^{12}$ [42]. Boosting the OFF/ON resistance ratio improves both the endurance and the retention capability of the cell.

**ReRAM Cell Settings:** All SPICE results presented in this chapter are based on an ReRAM device model [43] calibrated by $HfO_2$ ReRAM (1R) [7] and PTM [44] transistor model in the 45nm technology node. In the ReRAM model, we use activation energy $E_a$ of 0.8eV to enable ReRAM cells to operate under WL voltage ranging from 1V to 1.5V, which matches the supply voltage for low power main memory in 45nm [45].

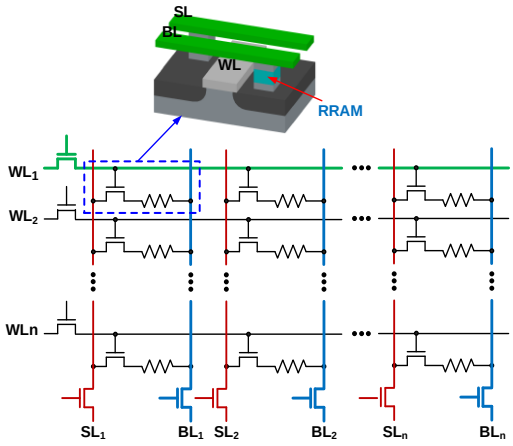



Fig. 2.3. 1T1R ReRAM memory array.

**Baseline ReRAM System:** The voltage settings of the baseline system are chosen such that it achieves a latency of 10ns for good performance and OFF/ON ratio of 30 for good reliability. Furthermore, the WL voltages for SET and RESET are different as in [7]. We use the notation $V_x$ to represent amplitude of $x$ and $\tau_x$ to represent pulse width of $x$. The baseline voltage settings are given by $V_{WL}$ = 0.9V, $V_{BL}$ = 1.3V and $\tau_{BL}$ = 5ns for SET and $V_{WL}$ = 1.5V, $V_{SL}$ = 1.85V and $\tau_{SL}$ = 5ns for RESET; $\tau_{WL}$ is the sum of $\tau_{BL}$ and $\tau_{SL}$ and equals 10ns.

## 2.3 Related Work

In recent years, there have been a few studies on latency/energy/reliability of ReRAM at the cell level [5], [6], [7]. Traditionally, SET and RESET use different WL voltages and the time to program an ReRAM block depends on the sum of worst case SET and RESET latencies. The write schemes in [5], [6] exploit the differences in write latencies due to cell to cell variations to cut off the SET current [6] or the SET and RESET voltage pulses [5] earlier than the worst case time. By implementing the cut off at different times for individual cells, the energy consumption is reduced. For reliability, a scheme to retain the endurance capability of the 1T1R cell after NPC of cycles is proposed in [7]. However, the proposed scheme results in very limited improvement of endurance ($\sim 10^7$ cycles) at the price of additional latency and energy consumption.

At the system level, there are several prior studies on incorporating ReRAM into main memory [8], [9], [46], [47]. Multi-level design of ReRAM spanning array, bank and chip levels is proposed in [8]. The reliability study in [8] is based on read noise margin of sense amplifier and does not take into account errors in the ReRAM cell. A design space exploration scheme for determining the array size, bank size for improving the latency and energy of a 1-diode-1-resistor (1D1R) ReRAM system is provided in [9]. Compared with the proposed 1D1R system, for an array size of 1024×1024, our proposed ReRAM system has 77% better write performance and 5× lower energy consumption. For cross-point ReRAM array, a procedure to detect and correct hard errors is presented in [46]. Since this procedure is implemented during decoding, it is likely to adversely affect the timing performance. Another ECC scheme that operates on several smaller sub-blocks simultaneously to improve the write (read) latency at the expense of large storage overhead is proposed in [47]. This method does not consider

retention errors which are also quite important when the data storage time is long or when the number of programming cycles is large.

## 2.4 Voltage Settings for Improving Latency and Energy — Circuit-level Strategy

As mentioned earlier, the two-step WRITE process based on different WL voltages for SET and RESET results in high programming latency. The high latency can be reduced by BL/SL boosting [47]. However, BL/SL boosting results in higher energy consumption, and also increases the risk of reversed p-n junction breakdown for the unselected cells. Moreover, BL/SL voltage boosting for 1T1R ReRAM shifts the SET/RESET pulse combination further away from the balance point, resulting in earlier write failure [7]. We propose using the same WL voltage for both SET and RESET to reduce the programming latency and to avoid the disadvantages of boosting BL (SL) voltages.

In our scheme, a common WL voltage is applied to the selection transistor of a block. BL voltage is applied to some cells to implement the SET operation and the SL voltage is applied to the others to implement the RESET operation. Since the same WL voltage is used in both SET and RESET operations, the WRITE latency is now determined by the larger of SET and RESET latencies. At the circuit level, we show how WL, BL and SL voltage pulses can be chosen properly to optimize one or more of the following metrics --- latency, energy and reliability. We assume that the programming latency is less than 30ns.

The voltage settings are chosen under the following four constraints:

(i) $1V \leq V_{WL} \leq 1.5V$: $V_{WL}$ is constrained to be larger than 1V to ensure that the ReRAM cell achieves both SET and RESET. However, $V_{WL}$ is set to be less than 1.5V to avoid high

gate leakage of the transistor. Note that boosting $V_{WL}$ (> 1.5V) can reduce latency significantly at the expense of degrading transistor reliability and hence is not considered here.

(ii) 1.3V ≤ $V_{SL}$ ≤ 2V: $V_{SL}$ is set larger than 1.3V to guarantee successful RESET within 30ns when $V_{WL}$ is 1.5V; $V_{SL}$ is set less than 2V to guarantee that the unselected cells do not undergo p-n junction breakdown.

(iii) 0.8V ≤ $V_{BL}$ ≤ 1.2V: $V_{BL}$ is set larger than 0.8V to ensure successful SET within 30ns when $V_{WL}$ is 1.5V; $V_{BL}$ is set less than 1.2V to ensure ReRAM cells operation in the low current region (10 – 40μA);

(iv) 10 ≤ OFF/ON ratio ≤ 100: OFF/ON ratio is set larger than 10 to handle noise margin and process variation of ReRAM. Also OFF/ON ratio < 10 requires a more sophisticated sense amplifier to determine the resistance state of ReRAM. OFF/ON ratio is set less than 100, which corresponds to HRS of 1MΩ and LRS of 10kΩ.

In the rest of this section, we show how write latency and energy can be reduced by appropriate choice of WL, BL and SL settings. The goal is to choose a voltage setting that is competitive with DRAM which has WRITE latency of 2ns and programming energy of 0.15pJ based on 1T1C SPICE simulation.

Fig. 2.4 shows the SET ($\tau_{BL}$) and RESET ($\tau_{SL}$) pulse widths as a function of $V_{WL}$, $V_{BL}$ and $V_{SL}$. We find that $\tau_{SL}$ decreases with either increasing $V_{WL}$ or increasing $V_{SL}$ while $\tau_{BL}$ is not sensitive to increasing $V_{WL}$. $\tau_{SL}$ is always larger than $\tau_{BL}$ because when the same WL voltage is used for both SET and RESET, $V_{GS}$ of the transistor during SET is always larger than the $V_{GS}$ of the transistor during RESET. This is because of the additional drop across the ReRAM device during RESET. The WRITE latency defined as $T_{READY} + \max\{\tau_{BL}, \tau_{SL}\}$ is determined by $T_{READY} + \tau_{SL}$, where $T_{READY}$ is the time to turn WL on before turning BL/SL on. $T_{READY}$ is

chosen to be 1ns, which is the smallest time unit in our system. The minimum WRITE latency is achieved at the largest possible $V_{WL}$ and $V_{SL}$.

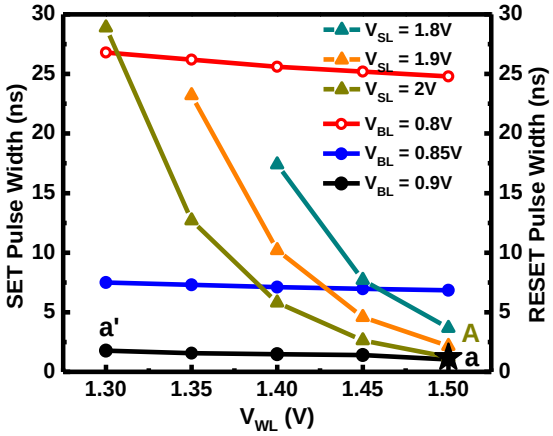



Fig. 2.4. Pulse widths of SET/RESET for different $V_{WL}$, $V_{BL}$ and $V_{SL}$. All points achieve OFF/ON ratio of 10. Large values of $V_{BL}$ and $V_{WL}$ reduce SET/RESET pulse widths.

Figure 2.5 shows that SET energy consumption increases mildly with increasing $V_{WL}$ while RESET energy reduces significantly when $V_{WL}$ or $V_{SL}$ increases. The RESET energy is significantly larger than the SET energy and the average energy consumption defined as (SET energy + RESET energy)/2 is dominated by RESET energy. The minimum average energy consumption is also achieved at the largest permissible $V_{WL}$ and $V_{SL}$ and therefore, the most latency-efficient configuration is also the most energy-efficient. This configuration corresponds to $V_{WL}$ = 1.5V and $V_{BL}$ = 0.9V (for SET) and $V_{SL}$ = 2V (for RESET). We denote this configuration as Config. (A, a) (Config. A is for RESET configuration and Config. a is for SET configuration). Its latency is 2.2ns and the corresponding average energy is 0.08pJ.

From Fig. 2.6 and 2.5, we can see that use of different WL voltages could result in slightly lower energy but at the price of much higher latency compared to use of same WL voltage. Config. (A, a') which uses different WL voltages has average energy of 0.07pJ but latency is

$T_{READY} + \tau_{BL} + T_{READY} + \tau_{SL}$ = 1ns + 2ns + 1ns + 1.2ns = 5.2ns, which is 2.4× larger than that of Config. (A, a).

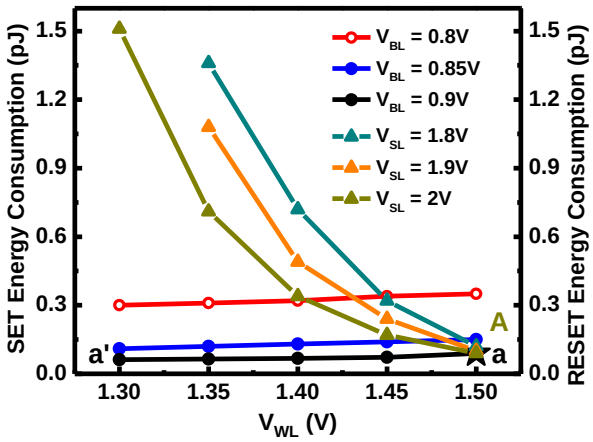



Fig. 2.5. Energy consumption of SET/RESET for different $V_{WL}$, $V_{BL}$ and $V_{SL}$. All points achieve OFF/ON ratio of 10. Large values of $V_{BL}$ and $V_{WL}$ reduce SET/RESET energy consumption.

## 2.5 Voltage Settings for Improving Reliability — Circuit-level Strategy

One major drawback of 1T1R ReRAM is that it suffers from reliability degradation due to process variations, structural limits and material property shift. Recent work in [7], [41], [48], [49] showed that errors in 1T1R ReRAM can be classified into retention errors and endurance errors. For instance, trapped oxygen vacancies ($V_O$) in conductive filament (CF) of ReRAM leak over time and cause resistance increase in both LRS and HRS resulting in data retention errors. Repeated programming of ReRAM results in shrinking of OFF/ON window causing endurance errors.

In the rest of this section, we describe procedures to find appropriate WRITE voltage settings to make ReRAM cell have better retention and endurance capabilities with small energy overhead in Section 2.5.1 and Section 2.5.2, respectively. Section 2.5.3 presents the trade-offs between improving retention and endurance capabilities, and Section 2.5.4 describes

an algorithm to find the proper WL, BL and SL pulse settings to minimize the total (retention and endurance) BER.

### 2.5.1 Voltage Setting for High Retention

We define data retention time (DRT) as the longest time that the data can be stored reliably, and data storage time (DST) as the time that the data is stored in memory between two consecutive WRITEs. Thus DST has to be less than DRT to avoid retention failures. We introduce three parameters that affect the DRT of 1T1R ReRAM at the circuit-level:

(1) OFF/ON ratio: As DST increases, OFF/ON ratio reduces and could result in retention failure. Thus, ReRAM cell with larger OFF/ON ratio can store data reliably for a longer time before the cell gets stuck at '0'. As stated earlier, a stronger voltage pulse for WL, BL (SET) and SL (RESET) can help achieve larger OFF/ON ratio.

(2) Current Compliance (abbreviated as CC): This is defined as the operation current constraint for SET [49]. In the low-current region, $10\mu A \leq CC \leq 40\mu A$, to keep the energy consumption low. Lowering the operation current during SET causes reduced amount of $V_O$ in the CF and results in data retention degradation. The operation current of SET is independent of $\tau_{WL}$ or $\tau_{BL}$, and is determined only by their amplitudes. Higher $V_{WL}$ and $V_{BL}$ improve data retention for ReRAM cell. Here we fix $V_{WL}$ at its largest possible value of 1.5V and find the value of $V_{BL}$. For instance, $V_{BL}$ of 1.2V is required to reach CC of 40μA.

(3) Number of Programming Cycles (NPC): With higher NPC, the loss rate of $V_O$ in the CF is accelerated and $R_{LRS}$ increases resulting in SET failure.

In order to estimate the retention time of the different configurations, we derived a model to fit the retention curves of IMEC $HfO_2$ ReRAM device [41]. The model is expressed in terms of $R_{LRS}$ which is a function of $gap_R$, the gap between CF and the top metal electrode (see

Fig. 1). The gap increases by $\Delta gap_R$ every time interval $\Delta t$ (1 h in our model) from the initial value of 0.35 nm. This increase is described by an exponential function [see (1)] [20]. $E_a$ (=0.8eV) is the activation energy of oxygen vacancies diffusion in $HfO_2$, which determines the slope of the Arrhenius plot. $A$ is the scaling factor of the exponential function which is determined by CC and NPC. Thus $R_{LRS}$ is a function of CC, NPC and DST. $R_{LRS}$ can also be expressed by (2), where the parameter values ($V_{READ}$ = 0.5V, current density $I_0$ = 61.4μA, $g_0$= 0.275nm and $V_0$ = 0.43V) were chosen to match the I-V curves in [7].

$$\Delta gap = A * exp\left(\frac{E_a}{kT}\right) * \Delta t \quad (1)$$

$$R_{LRS} = \frac{V_{READ}}{I_{READ}} = \frac{V_{READ}}{I_0 * e^{\left(-\frac{gap_R}{g_0}\right)} * \sinh\left(\frac{V_{READ}}{V_0}\right)} \quad (2)$$

As DST increases, the CF shrinks causing an increase in $R_{LRS}$ and could result in SET failure. We define DRT corresponds to the time when $R_{LRS} = R_{th,}$ where $R_{th}$ is given by (3). SET failure is defined by $R_{LRS} > R_{th}$.

$$R_{th} = (1 - \mu) \cdot R_{LRS\_Initial} \cdot \sqrt{\frac{OFF}{ON}\ ratio} \quad (3)$$

where $R_{LRS\_Initial}$ is the initial low resistance, which only depends on CC and temperature, and μ is the margin that is set to 10% in this chapter. Temperature here is 85°C because of industrial test requirement for retention capability.

Figure 2.6 shows a cartoon figure describing how retention and energy are affected by OFF/ON ratio and CC. All configurations on a curve have the same DRT. For fixed DRT, energy consumption increases with increasing OFF/ON ratio and decreasing CC. Both energy and DRT increase with increasing OFF/ON ratio and CC. Therefore, for a fixed DRT, the most energy-efficient configuration marked as black stars occurs at the largest possible CC

which corresponds to the smallest allowable $\tau_{BL}$. Thus, there are two approaches to improve retention of ReRAM: boosting OFF/ON ratio and increasing CC. With fixed $V_{WL}$, OFF/ON ratio can be boosted by increasing $V_{BL}$ or $\tau_{BL}$ (SET) and $V_{SL}$ or $\tau_{SL}$ (RESET). However, increasing $\tau_{BL}$ ($\tau_{SL}$) to boost OFF/ON ratio incurs high energy consumption compared to increasing $V_{BL}$ ($V_{SL}$). Since increasing $V_{BL}$ also increases CC, we choose to increase $V_{BL}$ to improve retention.

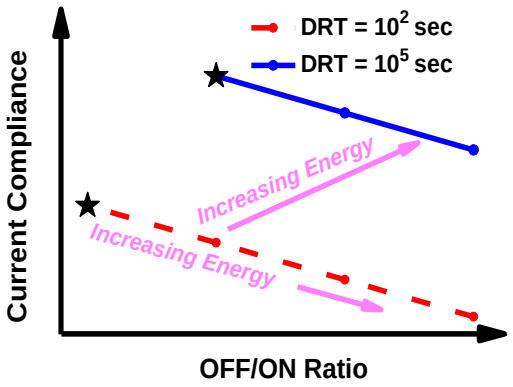



Fig. 2.6. Data retention and energy as a function of current compliance (CC) and OFF/ON ratio. All configurations on a curve have the same data retention time but different energy.

We pick three representative configurations b, c and d which have OFF/ON ratio of 10, 30 and 50, respectively. We choose the lowest possible value of $\tau_{BL}$ of 1ns. For this choice, Config. b, c and d have $V_{BL}$ of 0.90V, 1.05V and 1.20V, respectively, and also the largest possible CC of 28, 34 and 40μA, respectively. Thus, Config. b, c and d all have long retention times with small energy overhead.

Figure 2.7 shows DRT degradation for different configurations as a function of NPC. SET voltage settings ($V_{WL}$, $\tau_{WL}$, $V_{BL}$ and $\tau_{BL}$) determine CC and OFF/ON ratio. Given NPC, CC and OFF/ON ratio, we can obtain $R_{LRS}$ as a function of DST by using the retention fitting model [See (1) and (2)]. Then, DRT corresponds to the time when $R_{LRS} = R_{th}$. [See (3)]. Of these three configurations, Config. d has the highest DRT because of its highest CC and

OFF/ON ratio. For DRT constraint of $10^4$s, the corresponding NPC for Config. c is $10^{11}$, and for Config. b is $10^8$.

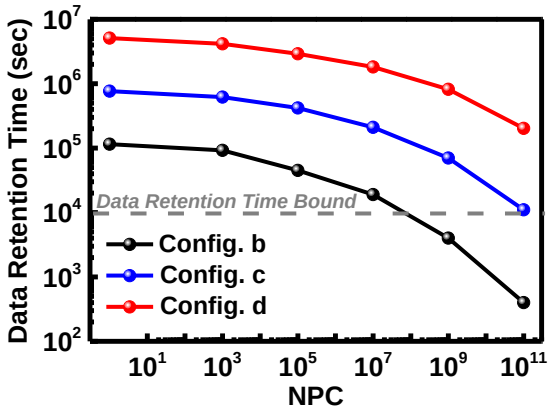



Fig. 2.7. Data retention time degradation due to NPC.

Based on the above analysis, we present a procedure to find the SET voltage settings for high retention with small energy overhead:

1. Set $V_{WL}$ to be the largest value in the permissible range.

2. Choose $\tau_{BL}$ to be the lowest value in the permissible range.

3. Find the largest possible $V_{BL}$ corresponding to the largest permissible value of CC and the largest possible $V_{WL}$.

### 2.5.2 Voltage Setting for High Endurance

There are two types of endurance errors: (1) The SET failure is due to extra recombination between $V_O$ and oxygen ion ($O^{2-}$) which causes the widening of electron tunneling gap and the reduction in the CF size. (2) The RESET failure originates from extra $V_O$ generation during SET process and causes an increase in the size of the CF, accompanied by reduction in resistances in HRS and LRS [7]. For CC > 40μA, the RESET failure is dominant while for CC

$\leq 40\mu A$ (which is considered in this chapter), the SET failure is dominant [49]. There are two parameters that affect the endurance of 1T1R ReRAM:

(1) OFF/ON ratio: As NPC increases, the OFF/ON window becomes narrower due to excess $V_O$ generation/recombination and could result in endurance failure. As stated earlier, use of strong WL, SL (BL) voltage pulses for RESET (SET) helps in boosting OFF/ON ratio. We fix $V_{WL}$ and $\tau_{SL}$ ($\tau_{BL}$) to find the value of $V_{SL}$ ($V_{BL}$) that helps achieve a certain OFF/ON ratio. For example, for RESET, when $V_{WL}$ is fixed at 1.5V and $\tau_{SL}$ is 10ns, $V_{SL}$ of 1.52V is required to reach the OFF/ON ratio of 10.

(2) Pulse Amplitude Ratio (abbreviated as P.A.R.): It is proportional to the strengths of the SET and RESET pulses and directly affects failure type and endurance of the ReRAM cell. It is defined by:

$$Pulse\ Amplitude\ Ratio = \frac{V_{SL}}{V_{wL}} \qquad (4)$$

A large $V_{WL}$ results in earlier RESET failure, while a large $V_{SL}$ results in earlier SET failure [7]. To improve endurance to SET failure, we can reduce P.A.R.; however, reducing P.A.R. by too much will lead to earlier RESET failure.

As NPC increases, the CF shrinks causing a significant increase in LRS. Increasing OFF/ON ratio does not change the CF shrink rate with increasing NPC but only delays the time of SET failure. A strong SET pulse that results in smaller P.A.R. helps ReRAM device to slow down the filament shrink rate during SET and improves endurance. Thus, lowering P.A.R. is a better approach to increasing the endurance compared to increasing OFF/ON ratio.

In order to estimate the endurance of different configurations, we also derived a model to fit the endurance curves of IMEC $HfO_2$ ReRAM device [7]. Note that the gap between CF and metal electrode, $gap_E$, also increases with increasing NPC. It increases by $\Delta gap_E$ every $\Delta$cycle (1000 cycles in our model). $\Delta gap_E$ is also described by an exponential function [see (5)] where $B$ is the scaling factor of the exponential function which is determined by P.A.R. and NPC. Thus $R_{LRS}$, which is related to $gap_E$, is a function of P.A.R. and NPC. $R_{LRS}$ can also be expressed by (6), where the parameter values of $V_{READ}$, $I_0$, $g_0$ and $V_0$ are the same as in the retention model.

$$\Delta gap_E = B * exp\left(\frac{E_a}{kT}\right) * \Delta cycle \tag{5}$$

$$R_{LRS} = \frac{V_{READ}}{I_{READ}} = \frac{V_{READ}}{I_0 * e^{\left(-\frac{gap_E}{g_0}\right)} * \sinh\left(\frac{V_{READ}}{V_0}\right)} \tag{6}$$

Figure 2.8 shows a cartoon figure showing how endurance and energy are affected by OFF/ON ratio and P.A.R.. All configurations on a curve have the same endurance. For fixed OFF/ON ratio, both energy and endurance reduces with increasing P.A.R.. For fixed endurance (in terms of NPC), energy consumption increases with increasing OFF/ON ratio and P.A.R.. Therefore, for fixed endurance, the most energy-efficient configuration, marked as black stars, can be achieved at the lowest permissible OFF/ON ratio.

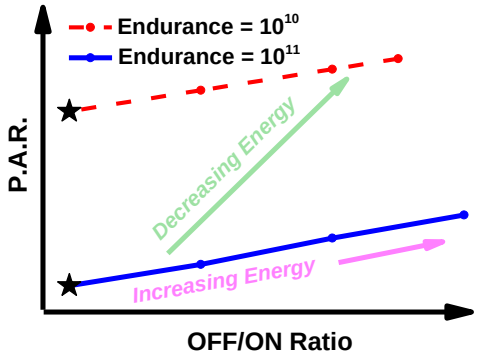



Fig. 2.8. Endurance and energy as a function of OFF/ON ratio and P.A.R.. All configurations on a curve have the same endurance but different energy.

We describe a procedure to find the voltage settings for high endurance with small energy overhead:

1. Choose OFF/ON ratio to be *N*, the lowest value in the acceptable range.

2. Set both $\tau_{SL}$ and $V_{WL}$ to the largest values in the permissible range.

3. Find the $V_{SL}$ corresponding to OFF/ON ratio of *N*.

For example, when OFF/ON ratio $\geq 10$, if the upper bound of $\tau_{SL}$ is 10ns and the upper bound of $V_{WL}$ is 1.5V, then $V_{SL}$ is 1.52V. When OFF/ON ratio lower bound increases to 30, and the other bounds are kept the same, the corresponding best configuration has $V_{SL}$ of 1.68V and an endurance of $10^{11}$.

### 2.5.3 Voltage Setting for High Retention and Endurance

In order to generate the BER curves for retention and endurance, we use the retention and endurance fitting models based on the IMEC $HfO_2$ ReRAM device [7], [41], [48] and use these models to estimate $R_{LRS}$ as a function of NPC. We run $10^8$ Monte-Carlo simulations in MATLAB [50] by varying the parameters according to Table 2.1 and calculating the number of retention and endurance errors for each NPC. Note that the variation parameters are chosen to guarantee that the I-V curves of 1T1R ReRAM with variations changes within a reasonable range (one order of magnitude).

TABLE 2.1 PARAMETER VALUES USED IN SPICE AND MATLAB SIMULATIONS FOR BER GENERATION

| | **Parameter** | **Value ($\mu \pm \sigma$)** |
|---|---|---|
| **ReRAM** | $g_0$ | 0.275nm ± 5% |
| | $V_0$ | 0.43V ± 5% |
| | $I_0$ | 61.4μA ± 5% |
| **CMOS** | $V_{th}$ | 469mV ± 47mV |
| | W/L | 1 |

Figure 2.9 shows the retention BER and endurance BER for SET and RESET configurations. Since SET and RESET should have the same OFF/ON ratio, we consider pairs of configurations, one for RESET and one for SET. For instance, Config. (B, b) is for OFF/ON ratio of 10, Config. (C, c) is for OFF/ON ratio of 30 and Config. (D, d) is for OFF/ON ratio of 50. From Fig. 2.10, we can see that (1) retention BERs are much more sensitive to NPC. (2) Config. (B, b) has the lowest endurance BER but the highest retention BER due to its lowest OFF/ON ratio. Similarly, Config. (D, d) with highest OFF/ON ratio of 50 has the lowest retention BER and the highest endurance BER. (3) Config. (C, c) has comparable retention BER and endurance BER.

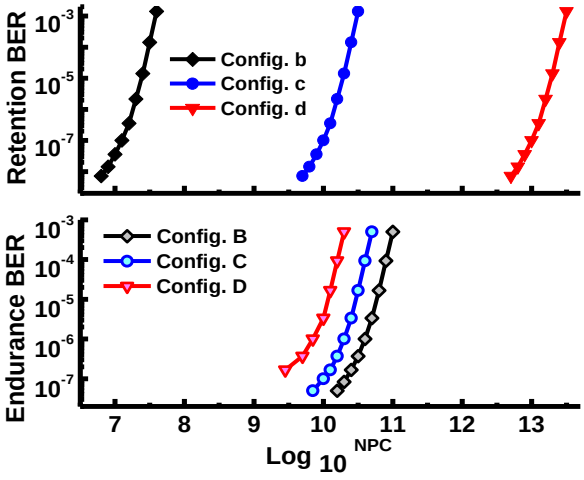



Fig. 2.9. Retention BER and endurance BER for different configurations.

In order to compute the BERs due to a combination of retention and endurance errors, we do not sum the endurance BER and retention BER. This is because some ReRAM cells contribute to both retention and endurance errors and should not be counted twice. So we build another MATLAB based simulation engine to accurately calculate the total BER. Figure 2.10 describes total BER for three candidate configurations as a function of NPC at DST of $10^4$s. The total BER for Config. (B, b) with OFF/ON ratio of 10 is dominated by retention

errors (see Fig. 2.9) and hence is much larger compared to the other two candidates. Similarly, the total BER for Config. (D, d) is dominated by endurance errors. Config. (C, c) has comparable retention and endurance errors and has the lowest total BER. Thus the configuration with comparable retention and endurance errors achieve the best reliability.

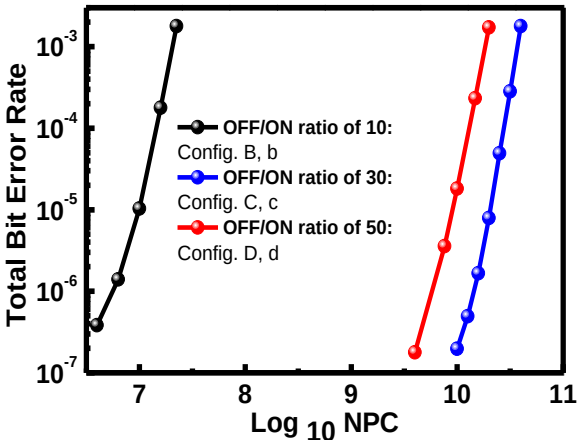



Fig. 2.10. Total BERs for the different configurations at DST of $10^4$s.

TABLE 2.2. VOLTAGE SETTINGS CANDIDATE CONFIGURATIONS

| Candidates | SET Operation | | | RESET Operation | | |
|---|---|---|---|---|---|---|
| | $V_{WL}$ (V) | $V_{BL}$ (V) | PW (ns) | $V_{WL}$ (V) | $V_{SL}$ (V) | PW (ns) |
| **Baseline** | 0.9 | 1.30 | 4.0 | 1.5 | 1.85 | 4.0 |
| **Config. (A, a)** | 1.5 | 0.90 | 1.0 | 1.5 | 2.00 | 1.2 |
| **Config. (B, b)** | 1.5 | 0.90 | 1.0 | 1.5 | 1.57 | 9.0 |
| **Config. (C, c)** | 1.5 | 1.05 | 1.0 | 1.5 | 1.73 | 9.0 |
| **Config. (D, d)** | 1.5 | 1.20 | 1.0 | 1.5 | 1.88 | 9.0 |

We list the voltage settings of Config. (A, a), Config. (B, b), Config. (C, c) and Config. (D, d) in Table 2.2. The DRT, endurance and energy consumption for the four candidate configurations are listed in Table 2.3. We calculate the endurance (in terms of NPC) first by using the endurance fitting model [see (5) and (6)]. Then, for a given endurance (NPC), DRT is calculated by using retention fitting model [see (1) and (2)]. For example, Config. (A, a) has

NPC of $10^{10.3}$ which corresponds to $10^3$s. From Table 2.3, we can see that (1) Config. (A, a) consumes the smallest energy but also has the worst endurance; (2) Config. (B, b) has the highest endurance (10× larger than Config. (A, a)) but the poorest DRT due to the lowest CC and OFF/ON ratio and consumes 2.3× higher energy; (3) Config. (D, d) has the largest DRT due to the highest CC but also consumes highest energy owing to the highest OFF/ON ratio. (4) Compared to Config. (D, d), Config. (C, c) has comparable endurance but lower energy consumption by 41% and much lower DRT by 100×.

TABLE 2.3. DRT, ENDURANCE AND ENERGY FOR CANDIDATE CONFIGURATIONS

| Candidates | OFF/ON ratio | P.A.R. | CC (μA) | Endurance (NPC) | DRT (sec) | Energy (pJ) |
|---|---|---|---|---|---|---|
| **Baseline** | 30 | 1.28 | 30 | $10^{10.5}$ | $10^{3.8}$ | 0.43 |
| **(A, a)** | 10 | 1.33 | 28 | $10^{10.3}$ | $10^{3}$ | 0.08 |
| **(B, b)** | 10 | 1.05 | 28 | $10^{11.3}$ | $10^{2}$ | 0.18 |
| **(C, c)** | 30 | 1.15 | 34 | $10^{11}$ | $10^{4}$ | 0.34 |
| **(D, d)** | 50 | 1.26 | 40 | $10^{10.9}$ | $10^{6}$ | 0.48 |

### 2.5.4 Voltage Setting for High Retention and Endurance

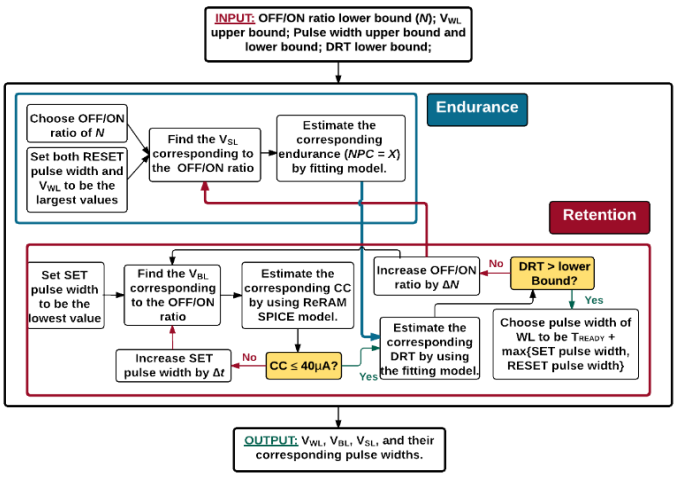



Fig. 2.11. Flowchart to find the WRITE setting, which enables the ReRAM cell to minimize the total BER with small energy overhead.

In order to derive voltage settings that minimize total BER, we need to find an optimal OFF/ON ratio which enables ReRAM cell to have comparable retention and endurance BER (like Config. (C, c)). This process is shown in Fig 2.11. The steps in the blue box are used to find appropriate RESET voltage settings to achieve high endurance. Similarly, the steps in the red box are used to obtain proper SET voltage settings for high retention. OFF/ON ratio is the most important parameter that links these two procedures. Also, if DRT calculated by the retention model is larger than the DRT lower bound, we choose $\tau_{WL}$ to be the $\max\{\tau_{BL}, \tau_{SL}\}$. Otherwise, OFF/ON ratio has to be increased to improve retention at the price of endurance capability. Also all steps have to be repeated until the DRT bound is satisfied.

Note that DRT lower bound is affected by wear-leveling. For example, if DRT $\geq 10^4$s, the upper bound of $\tau_{SL}$ is 10ns and the upper bound of $V_{WL}$ is 1.5V, then the optimal OFF/ON ratio is 30 (Config. (C, c)). When DRT lower bound decreases to $10^2$s due to wear-leveling, and the other bounds are kept the same, the corresponding best configuration has OFF/ON ratio of 10.

## 2.6 Bit-flipping – Architectural-level Strategy

In this section, we propose an architecture-level approach based on bit flipping to further reduce BER so that a low cost ECC scheme can be used to achieve high reliability.

Endurance errors of 1T1R ReRAM can be classified into 'visible' (*V*) endurance errors and 'invisible' (*I*) endurance errors. *V* error only occurs during WRITE '1' while an *I* error occurs during WRITE '0'. Blind flipping (B-Flipping) [51] is a technique that flips the information block after read-and-verify process in the WRITE operation. Note that while 'visible' endurance errors are stuck at the opposite value of what was written and can be detected by

READ-and-VERIFY process, the invisible endurance errors cannot be found by this process. By using B-Flipping, data is flipped only if data that is written (*d0*) and data that is read (*d1*) are different. In this way, all *V* errors are eliminated since they are flipped to *I* errors.

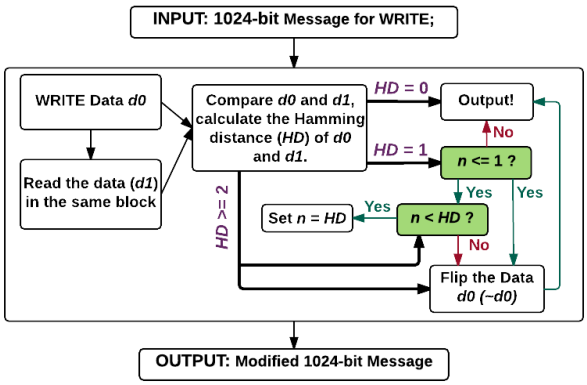



Fig. 2.12. Encoding Procedure of C-Flipping ($m$ = 2).

One downside of B-flipping is that it also flips *I* errors to *V* errors and could even increase the number of *V* errors. In order to overcome this side-effect, we propose an approach using an $m$-bit counter to record the total number of endurance errors and decide if flipping will reduce the number of *V* errors. This approach is named C-flipping. For example, if $m$ is 2, the counter records up to 3 endurance errors in total. If we observe one *V* error through read-and-verify, we do not flip the bits. This is because there are 3 - 1 = 2*I* errors and flipping will cause two new *V* errors while eliminating only one *V* error. The encoding procedure of C-Flipping with $m$ = 2 is shown in Figure 2.12. Hamming Distance (abbreviated as *HD*) between original data and the data stored in memory after WRITE is calculated. Thus, *HD* indicates the number of *V* errors. Let $n$ be the number of total endurance errors observed in the past WRITEs and it is recorded in the *2*-bit counter. If $n$ is less than *HD*, the value of $n$ is updated. We flip the data only if V errors are more than I errors, that is, if 1) *HD* = 1 and $n$ = 0 or 1, or 2) *HD* = 2 or 3. Therefore, C-Flipping with $m$ = 2 can help avoid the erroneous flipping

that happens when $HD = 1$ and $n = 3$. Note that larger $m$ gives more reduction in endurance errors at the expense of higher circuit overhead.

We find that in the proposed flipping scheme, using a simple 2-bit counter helps drop the endurance BER for NPC of $10^{10}$ (which corresponds to 10 years) from $10^{-7}$ to $3\times10^{-12}$. Thus the total BER drops by 2× resulting in BFR reduction of 10×. Such a reduction enables us to use a simple BCH ($t = 2$) code instead of a BCH ($t = 3$) code. Using a larger counter ($m > 2$) results in larger hardware overhead but with no overall benefit. This is because the total BER is dominated by the retention BER which is still $10^{-7}$ and thus BCH ($t = 2$) would still have to be used.

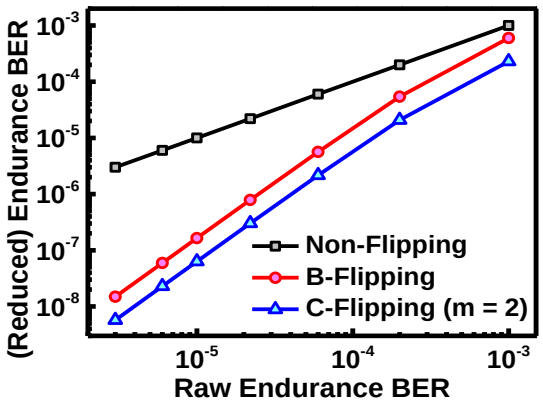



Fig. 2.13. Endurance BER reduction due to different flipping schemes.

Fig. 2.13 shows how flipping helps reduce the endurance BER. Without use of flipping or Non-Flipping, (reduced) endurance BER is equal to the raw endurance BER. B-Flipping and C-Flipping ($m = 2$) schemes provide two decade reduction in endurance BER; C-Flipping scheme has 2× lower endurance BER compared to B-Flipping. Thus with flipping, a simple ECC scheme is sufficient to handle the remaining errors as will be shown in the next section.

## 2.7 System-level Evaluation

In this section, we evaluate the system-level performance of the different ReRAM configuration for memory size of 1GB using CACTI [52] and GEM5 [53]. In order to study the potential use of ReRAM as main memory, we compare it with a DRAM system.

### 2.7.1 Voltage Setting for High Retention and Endurance

1) CACTI Setup

TABLE 2.4. CACTI RESULTS FOR 1T1R RERAM AND DRAM OF 1GB

| Candidate Configurations | Avg. Write (Read) Energy (nJ) | Write (Read) Latency (ns) | Leakage Power (mW) |
|---|---|---|---|
| **Baseline** | 4.92 (1.21) | 15.3 (4.6) | 9.53 |
| **Config. A (a)** | 1.87 (1.21) | 5.5 (4.6) | |
| **Config. B (b)** | 2.69 (1.21) | 12.3 (4.6) | |
| **Config. C (c)** | 3.98 (1.21) | 12.3 (4.6) | |
| **Config. D (d)** | 5.15 (1.21) | 12.3 (4.6) | |
| **DRAM** | 2.44 (2.3) | 5 (10) | 70.8 |

We obtain the ReRAM parameters, such as write (read) current, resistance, and access latency of a single cell using SPICE results (energy and latency per cell) in Section 2.4 and 2.5 and embed them into CACTI [52]. The results from CACTI for a 1GB memory are shown in Table 2.4. Since ReRAM is a resistive memory, the equations for bit-line energy and latency have to be modified accordingly. Note the read energy for ReRAM arrays are the same since read energy for a single ReRAM cell is quite small ($\sim 10^{-5}$ pJ) and the read energy for memory array is dominated by decoder energy and routing energy. The parameters for peripheral circuits are kept the same as the default parameters used in DRAM memory simulator with ITRS Low Operation Power (LOP) setting [54].

2) BFR Generation

We derive the Block Failure Rate (BFR) from BER using the following equation:

$$BFR = P(error > t) = \sum_{i=t+1}^{K} \binom{K}{i} BER^i (1 - BER)^{K-i} \quad (7)$$

where BER is the input to the ECC, $t$ is the correction strength of the ECC, and $K$ is the block size. We pick $K$ = 1024 for this chapter. In the rest of the chapter, we assume BFR is $10^{-13}$. This is quite typical and corresponds to failure of at most 1 block in one day when main memory access frequency is $5\times10^7$/s [55].

3) BCH Based ECC Schemes

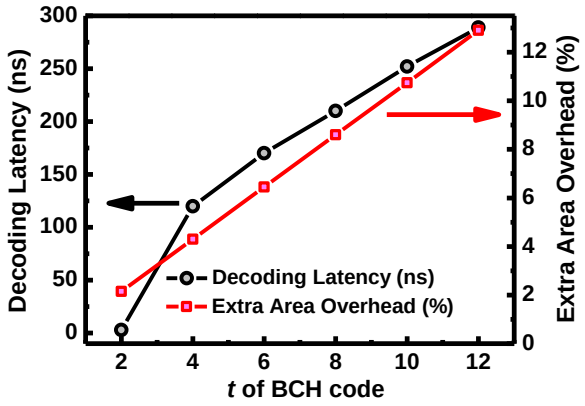



Fig. 2.14. Latency and area cost of BCH based ECC.

All ECC schemes are based on BCH [51]. While the iterative scheme is applicable for all $t$, for small t such as when $t$ = 1 or 2, an alternative way is to implement it using the method in [56]. For the case when $t$ = 2, the error locator equation is a quadratic equation, and its roots can be computed easily. When $t$ is large, the 2$t$-folded SiBM architecture [57] is used to minimize the circuit overhead of Key-equation solver at the expense of increase in latency. The syndromes are calculated in parallel and a parallel factor of 8 is used for calculations in the Chien search blocks. The BCH encoders and decoders are synthesized in 45 nm

technology using Nangate cell library [58] and Synopsys Design Compiler [59]. The cost of BCH code in terms of decoding latency and extra area overhead are shown in Figure 2.14. We see that the decoding latency and extra area overhead increase significantly with *t*.

4) Gem5 Setup

We use an out-of-order single core setting in GEM5 [53] to simulate the performance of a system with ReRAM based main memory of size 1GB. Our workload includes the benchmarks of SPEC CPU INT 2006 [60] and DaCapo-9.12 [61]. The ReRAM and DRAM write (read) latencies and energies obtained by CACTI are embedded in GEM5. The ECC latency of the BCH based schemes is expressed in number of cycles corresponding to the processor frequency of 2GHz. Read latency from main memory includes 95 cycles of wire routing delay, memory read operation latency and ECC decoder latency.

5) Wear-Leveling Scheme

In this chapter, we employ a popular wear leveling mechanism called Start-Gap [62] to make the writes uniform in each block of the ReRAM system. Thus, DST of a cell can be calculated based on the time interval when no write takes place in the cell during a period when there are φ writes to ReRAM.

$$DST = NB \times [\varphi \times (t_A + t_{WRITE} + t_{INTERVAL}) + (t_A + t_{READ} + t_{WRITE})] \quad (8)$$

where *NB* is the number of blocks in main memory and is equal to 8M if the block size is 1Kb and main memory size is 1GB. $t_A$ is the time for CPU transferring the logic address to physical address and is 95 cycles. φ is the parameter that determines the wear leveling frequency. We choose φ = 100 here. Therefore, the average DST is $10^4$s based on the benchmarks of SPEC CPU INT 2006 [60] and DaCapo-9.12 [61].

### 2.7.2 IPC, Lifetime and Energy Evaluation

**IPC**: We find that all the candidate ReRAM systems have comparable IPC in spite of having different write latencies. This is due to write latency of main memory being hidden by use of the multi-level caches. We find that IPC decreases mildly until write pulse width becomes 10× larger. If the normalized IPC loss is constrained to 2%, the corresponding write pulse width is less than 10ns. Therefore, we set the latency of write scheme to be within 10ns.

**Lifetime:** The lifetime is obtained from the Block Failure Rate (BFR) vs NPC curves. Assuming lifetime $Y$ in terms of years, we can derive the Endurance Requirement ($W_{max}$) using the following equations [34]:

$$W_{max} = \frac{f_{WRITE} \cdot Y}{NE \cdot NB} \cdot 2^{25} \quad (9)$$

where $NB$ is the number of blocks and $f_{WRITE}$ is the write frequency of main memory ($f_{WRITE}$ is $5\times10^7$/s based on the worst case GEM5 benchmarks). $NE$ is the Normalized Endurance determined by the wear-leveling approach used; for Start-Gap, $NE$ is 20% [62]. Thus, the main memory must sustain for $f_{WRITE} \cdot Y \cdot 2^{25}$ processor cycles, given that there are approximately $2^{25}$ seconds in a year. Therefore, $W_{max}$ is $10^{10}$ programming cycles for 10 years.

TABLE 2.5. REQUIRED BCH CODE FOR DIFFERENT CONFIGURATIONS FOR THE SAME LIFETIME OF $10^{10}$

| Candidate Systems | ECC | Flipping | Lifetime (NPC) |
|---|---|---|---|
| **DRAM** | No | No | $10^{16}$ |
| **Baseline** | $t = 12$ | No | $10^{10}$ |
| **Config. (C, c)** | $t = 5$ | No | $10^{10}$ |
| **Config. (C, c)** | $t = 3$ | B-Flipping | $10^{10}$ |
| **Config. (C, c)** | $t = 2$ | C-Flipping | $10^{10}$ |

**Energy:** Total energy includes ReRAM write (read) energy along with energy consumed by parity storage, ECC encoding/decoding energy and leakage energy of peripheral circuit. Note

that the ECC encoding/decoding energy is trivial compared to the write (read) energy of the system. Also parity storage is a function of the error correction capability. For instance, for BCH ($t$ = 5), the extra overhead is 5.4%, while for BCH ($t$ = 2), it is only 2.1% (See Fig. 2.14). Table 2.5 describes the lifetime in terms of NPC when different ECC schemes are employed. In order to achieve lifetime of 10 years (NPC = $10^{10}$), different candidates require ECC with different strengths except for DRAM. For instance, Baseline ReRAM system requires BCH $t$ =12, Config. (C, c) with Non-Flipping needs BCH $t$ = 5, Config. (B, b) with C-Flipping ($m$ = 2) needs BCH $t$ = 2. Note that DRAM does not require any ECC due to its superior endurance.

Consider ReRAM systems that have a lifetime of 10 years. Figure 2.15 compares the IPC of ReRAM systems normalized to that of a DRAM system for SPEC CPU INT 2006 and DaCapo-9.12 benchmarks. From the figure, we can see that our circuit-level scheme (ReRAM + Ckt) can improve IPC on average by 21% compared to the baseline system. However, its IPC is still 23% lower than that of the DRAM system. The proposed ReRAM system with cross-layer technique (ReRAM + Ckt + Arch) has 5.2% higher IPC compared to the DRAM system and is a clear winner.

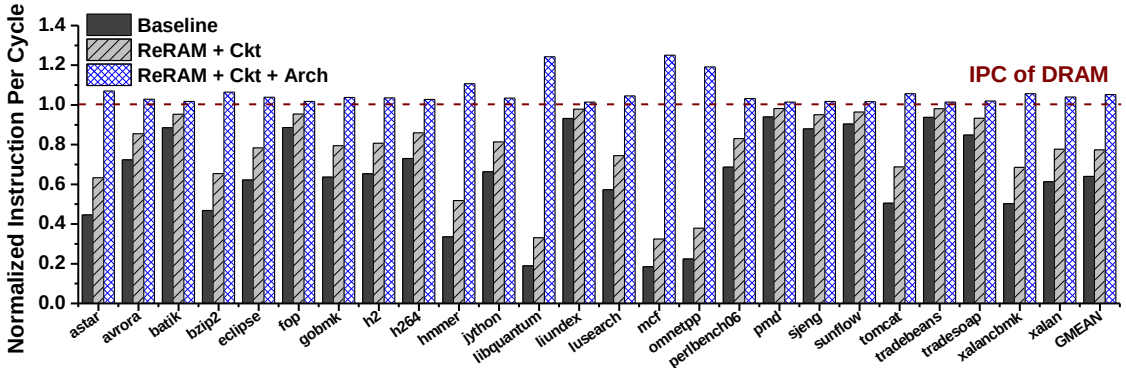


Fig. 2.15. IPC of SPEC CPU INT 2006 and DaCapo-9.12 benchmarks normalized to that of the DRAM system.

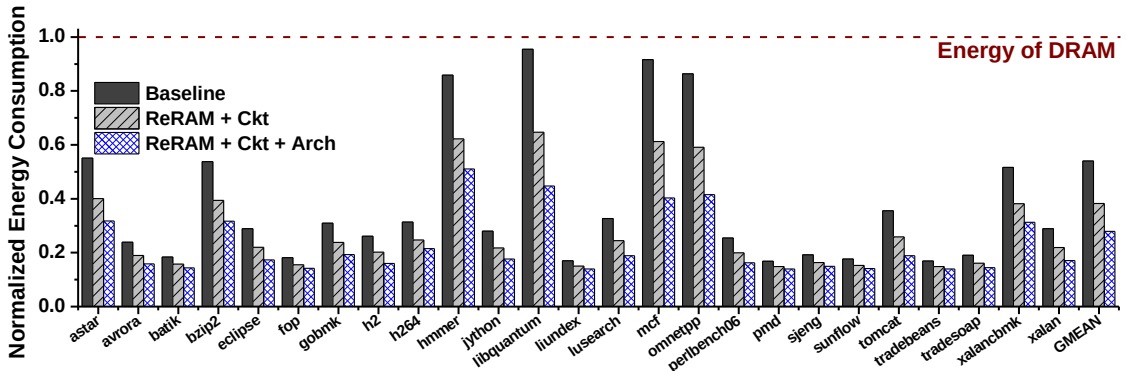


Fig. 2.16. Energy of SPEC CPU INT 2006 and DaCapo-9.12 benchmarks normalized to that of the DRAM system.

Figure 2.16 shows the energy of ReRAM systems with lifetime of 10 years normalized to that of a DRAM system for SPEC CPU INT 2006 and DaCapo-9.12 benchmarks. In the figure, the proposed ReRAM system with cross-layer technique (ReRAM + Ckt + Arch) has lowest energy consumption, which is, on average, only 28% of the DRAM system.

TABLE 2.6. IPC, ENERGY AND LIFETIME FOR DIFFERENT CONFIGURATIONS

| Candidate Systems | IPC | Energy (mJ) | Lifetime (Yrs) |
|---|---|---|---|
| **DRAM** | 0.3249 | 17.2 | > 10 years |
| **Baseline + BCH ($t$ = 12)** | 0.2080 | 9.29 | 10 years |
| **Config. (C, c) + BCH ($t$ = 5)** | 0.2513 | 6.97 | 10 years |
| **Config. (C, c) + BCH ($t$ = 3) + B-Flipping** | 0.2696 | 6.22 | 10 years |
| **Config. (C, c) + BCH ($t$ = 2) + C-Flipping ($m$ = 2)** | 0.3418 | 4.80 | 10 years |

Table 2.6 compares the average IPC, average energy and lifetime of different configurations. While all the ReRAM systems have the same lifetime of 10 years, the DRAM system has higher lifetime due to its outstanding endurance. Among all ReRAM systems, baseline has the poorest IPC and highest energy consumption owing to use of a strong BCH

code ($t$ = 12) with a large decoding latency (see Fig. 2.14). Circuit level optimizations resulted in Config. (C, c) which has better IPC and lower energy compared to the baseline. However, its IPC still much lower than that of a DRAM system.

Architecture-level schemes, which reduce BER, results in use of low-$t$ BCH codes for the same lifetime. For example, Config. (C, c) with B-Flipping requires BCH ($t$ = 3) instead of $t$ = 5 code. While this reduces the energy due to lower parity storage, its IPC is comparable. This is because the decoding latency of BCH ($t$ = 5) and BCH ($t$ = 3) are not significantly different. With C-Flipping, it is sufficient to use BCH ($t$ = 2) scheme, resulting in significant enhancement in IPC due to its very small decoding latency. Config. (C, c) with C-Flipping also outperforms DRAM system with respect to IPC by 5.2% and has an energy saving of 72%. Therefore, a combination of BL, WL and SL voltage settings at the circuit-level, selective bit-flipping at the architecture level and BCH-based ECC at the system level can help the ReRAM system be competitive with the DRAM system.

## 2.8 Conclusion

In this chapter, we propose cross-layer techniques to improve reliability of ReRAM systems with minimum latency and energy overhead. At the circuit level, we first propose to use the same WL voltage for SET and RESET to reduce latency. We show how WL, BL and SL voltage settings can improve write latency, energy and reliability of 1T1R ReRAM. We show that the most latency-efficient configuration is the same as the most energy-efficient configuration. Next, we show how appropriate choice of voltage settings can help improve ReRAM cell retention or endurance. However, the voltage settings used for minimizing retention errors cannot be used to minimize endurance errors and so we present a procedure to derive the optimal voltage settings that minimize the total number of errors (retention and

endurance). Next, we show how a bit flipping technique can be used to further relax the requirement of ECC. Finally, we evaluate the system-level performance for a 1GB ReRAM and DRAM main memory. We show that if the proposed circuit-level and architecture-level schemes are used, the ReRAM system can reach lifetime of 10 years by using the simplest BCH code ($t$ = 2). Simulation results using SPEC CPU INT 2006 and DaCapo-9.12 benchmarks show that proposed schemes for ReRAM outperform DRAM main memory with respect to IPC performance (5.2% higher) and energy (72% lower).

# CHAPTER 3

# IMPROVING RELIABILITY OF 1S1R RERAM SYSTEM

## 3.1 Introduction

As mentioned in previous chapters, ReRAM can be organized into the 1-transistor-1-resistor (1T1R) or 1-selector-1-resistor (1S1R) array architecture. Of the two types of ReRAM array architectures, the cross-point 1S1R array architecture has higher integration density compared to the 1T1R architecture [1] and is hence considered in this work. In the cross-point architecture, the bit-line (BL) and word-line (WL) are perpendicular to each other and memory cells are sandwiched in between. Such a structure has $4F^2$ cell area, where F is the lithography technology node. Unfortunately, the cross- point array suffers from sneak path and IR drop, resulting in lower reliability [1]. To reduce the effect of sneak paths during memory cell operation, a highly nonlinear, bidirectional selector device (1S) is serially connected with each bipolar resistor (1R) in a 1-selector-1-resistor (1S1R) cell configuration [63]. 1S1R has almost the same area as the cross-point (= $4F^2$) structure since the selector device is vertically stacked with the ReRAM cell.

Most of the prior work on ReRAM cross-point array focused on device and circuit issues [13]–[19]. These include selector and ReRAM cell level designs that improve read/write margins [13]–[18]. There has also been work on cross-point array organization as well as array size evaluation with respect to energy consumption and reliability. However, most of the previous work was based on "single bit per read/write" per subarray [13]–[17], a scheme which incurred large power consumption since multiple subarrays have to be activated at a time to meet the I/O bandwidth.

In this chapter, we propose a 1S1R cross-point array system with "multi-bit per access" per subarray that achieves high energy-efficiency and good reliability. To the best of our knowledge, this is the first work that considers energy, latency and reliability of such an architecture. It analyzes the effect of cell-level as well as array-level variations sources on error rates and proposes a low cost scheme to maintain reliability and latency with low energy consumption.

The rest of this chapter is organized as follows. In Section 3.2, we review the ReRAM basics including reliability characteristics. Section 3.3 summarizes related work. In Section 3.4, we analyze the effect of device-to-device (D2D) and cycle-to-cycle (C2C) variations on the resistance values at the cell level and show how appropriate choice of BL and SL voltages can help improve reliability. In Section 3.5, we show how different variation sources, namely D2D, C2C as well as IR drop, affect the resistance distributions in an array. In Section 3.6, we describe how the proposed Rotated Multi-array Access scheme can be used to relax the ECC requirement. This is followed by system-level evaluation of the proposed ReRAM system with respect to area, performance, energy and reliability. We conclude the chapter in Section 3.7.

## 3.2 Background

### 3.2.1 Cross-point ReRAM Array Architecture

There are two types of ReRAM array architectures: the 1-transistor-1-resistor (1T1R) structure and the cross-point structure. In 1T1R array, each memory cell is in series with a cell selection transistor [1]. As the size of the transistor is typically much larger than the size of ReRAM cell, the total area of memory array is primarily dominated by transistors rather than the ReRAM cells. In contrast, the cross-point architecture has $4F^2$ cell area and hence is more area-efficient than the 1T1R structure [1]. However, the cross-point architecture suffers from

interference among cells and the commonly known as sneak path problem that limits the array size, increases the power consumption, and degrades the reliability [8]. A two-terminal selector device is typically added in series with the ReRAM cell at each cross-point. The resulting 1-selector-1-resistor (1S1R) structure enables design of a large-scale cross-point array by cutting off the sneak path current of the half-selected and unselected cells [1]. 1S1R has the same area with cross-point (= $4F^2$) since the selector device is vertically stacked with the ReRAM cell.

**Reliability Issues:** The cross-point array suffers from two well-known problems: (1) IR drop along the interconnect wires. The IR drop problem becomes significant when the WL and BL wire width scales to sub-50 nm regime where the interconnect resistivity drastically increases due to the electron surface scattering [1]. During write operation, the farthest cell from the driver has insufficient voltage drop, resulting in unsuccessful write. (2) Sneak path problem through the half-selected cells and unselected cells. The half-selected cells along the selected WL and BL lines conduct leakage current and form sneak paths during the read/write operation. The sneak paths contribute current to the IR drop and further degrade the read/write margin.

### 3.2.2 Cross-point ReRAM System Organization

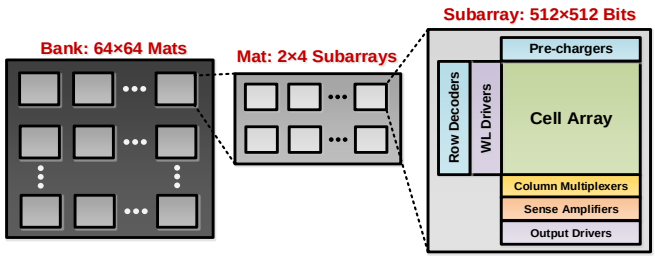



Fig. 3.1. A hierarchical memory organization with one bank, 64×64 mats per bank, and 8 subarrays per mat (adapted from [64]).

The cross-point ReRAM system organization that is supported in NVSim is shown in Fig. 3.1 [64]. A 1GB bank consists of 64×64 mats, where each mat consists of 2×4 subarrays and each subarray consists of a cell array with 512×512 1S1R cells (512 rows with 512 bits per row) as well as peripheral circuitry with row decoders, column multiplexers, sense amplifiers and output drivers. A subset of mats and a subset of subarrays within each mat can be activated simultaneously. Activating multiple mats and multiple subarrays per mat improve the timing performance at the expense of higher energy. While similar time performance can be achieved by activating multiple (say *K*) subarrays in one mat versus *K* mats with one subarray per mat, the energy consumption of activating multiple mats is higher, as will be illustrated in next Section 3.5.

**Baseline Cross-point ReRAM System:** The conventional cross-point ReRAM system accesses single bit for read/write per subarray and so we choose this as the baseline system. If the I/O width is 64 bits, for better performance, 64 subarrays (8 mats with 8 subarrays per mat) are activated every time. Such a scheme has high energy overhead due to 64 subarrays being activated per access. In the next section, we propose a scheme that accesses multi-bit per read/write to reduce the number of subarrays that are required to be activated per access, resulting in higher energy-efficiency.

### 3.2.3 Cross-point ReRAM System Organization

All SPICE results presented in this chapter are based on a ReRAM device compact model [65] calibrated by IMEC's $HfO_2$ ReRAM (1R) [43] and the field-assisted super-linear threshold (FAST) [66] selector model in the 22nm technology node. The conductive filament of $HfO_2$ ReRAM (which is our case) is composed of oxygen vacancies as in [43, 67]. Here both the ON and OFF states are assumed to have the same nonlinearity of 10×, defined as the ratio of the

current at $V_{WRITE}$ to that at $V_{WRITE}/2$ [1]. The threshold voltage ($V_{TH}$) of FAST is set at 1.2V. $\Delta V_{TH}$, the tolerance for $V_{TH}$ variation in selectors, is set at 0.1V. During the read operation for a single cell, $V_{READ}$ (= 1.35V) is set to be larger than $V_{TH\_MAX} = V_{TH} + \Delta V_{TH}$ (= 1.3V) to ensure that there is enough readout current to sense the status of the selected cells. In order to guarantee that all the half-selected and unselected cells remain OFF during write operation, $0.5 \times V_{WRITE}$ (= 0.975V) is set to be less than $V_{TH\_MIN} = V_{TH} - \Delta V_{TH}$ (= 1.1V). The FAST selector increase the 1S1R's nonlinearity to $10^6$ [66]. The sense amplifier is based on current mode and has a sensing speed of 10ns [68].

TABLE 3.1. PARAMETER SETTINGS FOR 1S1R CROSS-POINT ARRAY

| | **Parameters** | **Notes** |
|---|---|---|
| **ReRAM** | Nonlinearity: 10× | I @ $V_{WRITE}$ / I @ $0.5V_{WRITE}$ |
| | $V_{SET}$ = 1.95V; $\tau_{SET}$ = 5ns. | Mean OFF/ON Ratio = ~15; Tail-to-tail OFF/ON Ratio = ~3. |
| | $V_{RESET}$ = -1.95V; $\tau_{RESET}$ = 5ns. | |
| **FAST Selector** | Type: Threshold Selector | $0.5V_{SET} < V_{TH} < V_{READ} < V_{SET}$ |
| | $V_{TH} \pm \Delta V_{TH}$: 1.2V ± 0.1V | $0.5V_{SET} < V_{TH} - \Delta V_{TH}$ |
| | OFF Leakage: ~fA. | When $V < V_{TH} - \Delta V_{TH}$ |
| | $V_{READ}$ : 1.35V | $V_{READ} > V_{TH} + \Delta V_{TH}$ |
| **1S1R Array** | Array Size: 512×512 | Bit-cell Area = $4F^2$ = 1936nm$^2$ |
| | The Number of Bits per read/write (*NB*): 1, 4, 8, 16 and 32 | Group Size = 1, 4, 8, 16 and 32 bits |
| | $V_{WRITE}$ ($V_{READ}$) : 3V(2V) | Boosted due to IR Drop |
| | Wire Resistance per Length: 1Ω | Copper, L = 2F, S = $1.6F^2$ |
| | W/L of the Driver: 10 Technology Node: 22nm | W/L of NMOS = W/L of PMOS |
| | Driver Transistor: 22nm_LP PTM | 22nm_LP PTM; Its leakage < 22nm_HP PTM |
| | Sense Amplifier: Current-mode | Sense Speed = ~10ns |

Parameter settings of the ReRAM cell, the selector, and array configurations are summarized in Table 3.1. To guarantee a successful write operation in the cross-point array, the read and write voltages have to be boosted above the actual voltage drop on the ReRAM cell to compensate for the IR drop [1]. For array size of 512×512, $V_{DD}$ is boosted from 1.35V to 2V for read and from ±1.95V to ±3V for write operation so that the farthest cell from the driver can be accessed successfully.

## 3.3 Related Work

Existing work on 1S1R cross-point memory focuses mostly on the selector design to achieve significant reduction in the half-write current [13-18], [63] or increase the nonlinearity of the RRAM cell to minimize the IR drop and effect of sneak paths [13-16, 66]. At the array level, strategies to partition large arrays into multiple smaller subarrays to increase the overall read/write performance have been proposed in [47, 8]. Multi-level design of ReRAM spanning array, bank and chip levels is proposed in [47]. The reliability study in [8] is based on read noise margin of sense amplifier and does not take into account errors in the ReRAM cell. Also, work in [47, 8] evaluates the reliability based on the worst case scenario which is dictated by the cell located farthest away from the driver. However, in their evaluation, the variability sources such as those due to D2D only, C2C and IR drop have not been considered, resulting in inaccurate estimation of reliability.

Also, most existing 1S1R array systems are based on single bit per read/write per subarray [13-17]. In order to reduce latency, multiple subarrays have to be activated, resulting in high energy consumption. A multi-bit per access scheme has been suggested to improve the energy-efficiency in [18, 19]. It has been shown that the driving current requirement and corresponding area overhead for each word line in multi-bit per access scheme is much larger

than that of single-bit per access scheme. However, the focus has mostly been on the design of the peripheral circuits such as drivers and sense amplifiers to support multi-bit per access; reliability issues due to a multi-bit per access scheme have not been considered. In contrast, this work is a comprehensive study of energy, latency and reliability of an 1S1R cross-point array architecture with multi-bit per access.

Another competitive ReRAM technology is based on 1T1R. The 1T1R ReRAM cell has the same density as 1T1C DRAM cell, featuring $6F^2$ cell area (where F is the lithography technology node) and does not have the sneak path current problem of cross-point array. At cell level, prior work for 1T1R focus on fabrication procedure as well as retention and endurance [69-71]. At the circuit level, related work [48, 49, 72] show the effect of different programming conditions on endurance. At the system level, our previous work shows that how voltage settings (pulse amplitude and pulse width) of word-line, source-line and bit-line voltage can be used to lower latency, lower power and improve reliability [10-12].

## 3.4 Effect of Variations on ReRAM Cell Resistance

In this section, we show the effect of spatial variations or device to device variations (described in Section 3.4.1) and temporal variations or cycle to cycle variations (described in Section 3.4.2) on the resistance distribution of an ReRAM cell.

### 3.4.1 Effect of D2D Variation on Resistance Distribution at Cell Level

We present LRS and HRS resistance distributions due to device-to-device or D2D variations for $HfO_2$ ReRAM device [43], shown in Fig. 3.2. We run $10^6$ Monte-Carlo simulations in MATLAB by varying the parameters of the compact device model [65]

according to Table 3.2. The variation parameters are chosen to match the experimental resistance distribution data in [15].

TABLE 3.2. PARAMETER VALUES USED IN MATLAB SIMULATIONS

| | Parameter | Value ($\mu \pm \sigma$) |
|---|---|---|
| **D2D Variations** | $g_0$ | 0.275nm ± 3~5% |
| | $V_0$ | 0.43V ± 3~5% |
| | $I_0$ | 61.4μA ± 3~5% |
| | $v_0$ | 150m/s ± 3~5% |
| | $g_{MIN}$ | 0.54nm ± 3% |
| | $g_{Max}$ | 1.37nm ± 3% |
| **C2C Variations** | $g_{VAR}$ | $\sim 2.5\times10^{-7}\times\tanh(g-g_{MIN})\times \tanh(g_{MAX}-g)$ |

When the number of programming cycles (NPC) increases, the OFF/ON ratio (defined as $R_{HRS}/R_{LRS}$) shrinks, resulting in reliability degradation. We represent the OFF/ON ratio in terms of mean OFF/ON ratio, which is the ratio of mean $R_{HRS}$ to mean $R_{LRS}$, and tail-to-tail OFF/ON ratio, which is the ratio of the lowest $R_{HRS}$ to the largest $R_{LRS}$. We target NPC of $10^6$, which is the lifetime of ReRAM that most previous papers have reported [1, 4]. For NPC of $10^6$, the tail-to-tail OFF/ON ratio is chosen to be 3 based on the experimental data presented in [15]. Mean OFF/ON ratio depends on the SET and RESET pulse strengths and varies from 10 to 30 according to previous work [48, 49, 72]. Therefore, we set mean OFF/ON ratio to be 15 ($\approx \sqrt{10 \times 30}$), which is the average in log scale.

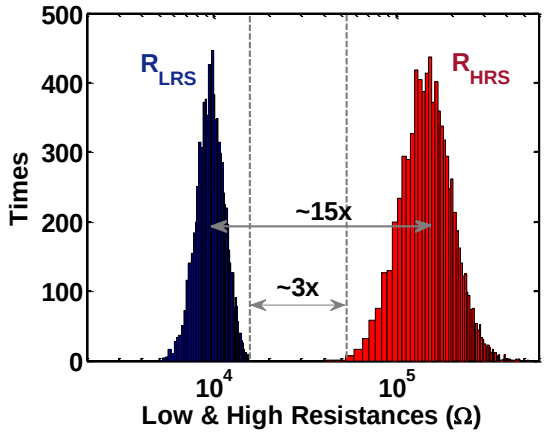



Fig. 3.2. Write resistance distributions due to D2D variations @ NPC = $10^6$. The mean OFF/ON ratio is 15, and the tail-to-tail OFF/ON ratio is 3.

### 3.4.2 Effect of C2C Variation on Resistance Distribution at Cell Level

The cycle-to-cycle or C2C variation is attributed to the stochastic nature of the oxygen vacancies/ions. Due to the randomness of the oxygen vacancy generation and ion migration at the nanoscale, the shape of the conductive filament varies from C2C even under the same programming condition [1].

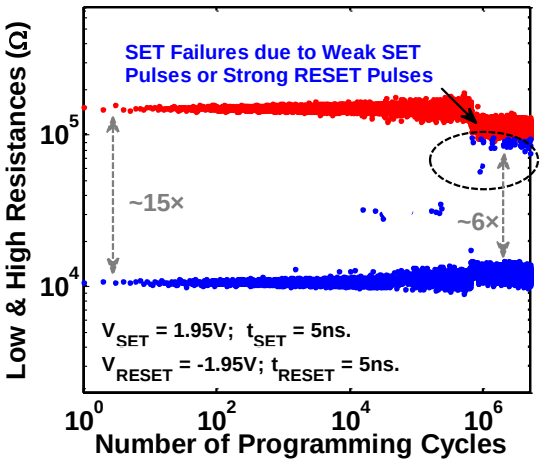



Fig. 3.3. Write resistance distributions due to C2C variations. SET Failures due to weak SET pulse and strong RESET pulse. Blue and red dots correspond to low resistance and high resistance values of the ReRAM device.

In order to evaluate the effect of C2C variation on ReRAM resistance distribution, we vary the parameters in Table II and simulate for $10^6$ consecutive cycles, where each cycle consists

of a SET followed by a RESET. We found that the errors due to C2C variations are dominated by SET failures and these failures increase with NPC, as shown in Fig. 3.3. SET failures can be caused by weak SET pulse or strong RESET pulse in the previous cycle (marked by the dashed black circle). SET failures due to weak SET pulse can be recovered by a second SET operation. The remaining SET failures, after a second SET operation, are due to a strong RESET pulse.

We run Monte-Carlo simulations and evaluate the Bit Error Rate (BER) due to continuous cycling of the ReRAM cell under different SET and RESET programming conditions. From Fig. 3.4, we see that a stronger SET voltage can be used to significantly reduce the SET failures. However, the reduction in BER comes at the expense of increase in the energy consumption because of increasing SET voltage. We pick SET voltage of 1.95V in this chapter since SET voltage larger than 1.95V does not significantly reduce BER and yet incurs large energy consumption. In the rest of the chapter, we use the following settings: $V_{SET}$ = 1.95V, $\tau_{SET}$ = 5ns for SET and $V_{RESET}$ = -1.95V, $\tau_{RESET}$ = 5ns for RESET. Here $V_x$ represents amplitude of $x$ and $\tau_x$ represents pulse width of $x$.

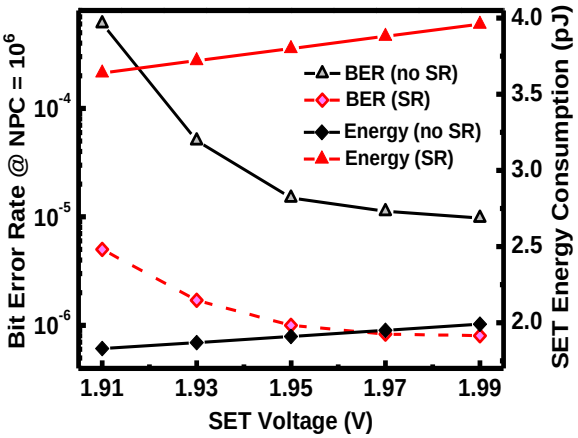



Fig. 3.4. BER and SET energy consumption as a function of SET voltage. SET Recovery is abbreviated as 'SR' in the figure.

## 3.5 Access Scheme with Multi-bit per Read/Write

Accessing multi-bit is possible by using the V/2 bias scheme [1]. Consider the $N \times N$ array shown in Fig. 3.5, where $N$ is both the number of WLs and the number of BLs. We choose the V/2 bias scheme [1] because of its lower read/write energy consumption over V/3 bias [1] and full scheme [1]. In the V/2 bias scheme, for SET operation, all the selected WLs and BLs are set to '$V_{WRITE}$' and '0', respectively. For the RESET operation, the bias conditions on WL and BL are reversed to be '0' and '$V_{WRITE}$' to enable bipolar switching. In both SET and RESET operations, all the unselected WLs and BLs are set to '$V_{WRITE}/2$'. In this way, the access voltage on the selected cell is '$V_{WRITE}$', the half-selected cells have voltage drop of '$V_{WRITE}/2$' and unselected cells ideally have no voltage drop. Bias condition for read operation is similar to that for SET operation with $V_{READ}$ instead of $V_{WRITE}$.

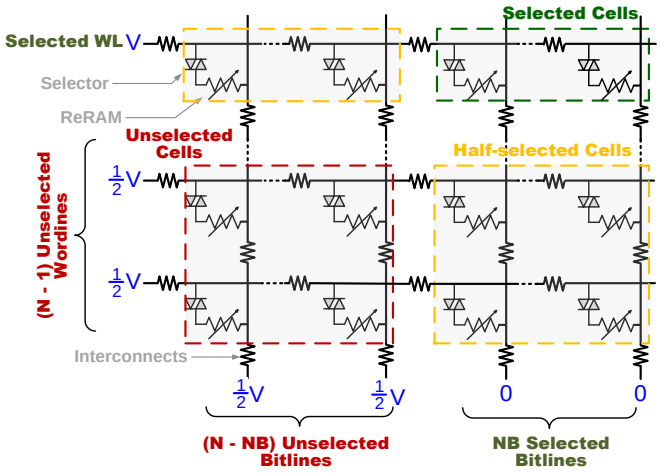



Fig. 3.5. V/2 bias scheme used in the 1S1R cross-point array architecture of size $N \times N$. $NB$ is the number of selected BLs.

Define a 'group' as $NB$ consecutive bits in a subarray, as shown in Fig. 3.5. An $NB$-bit group can be read simultaneously by using the V/2 bias scheme [1]. However, an $NB$-bit write takes

two steps: all the '1's are simultaneously written into a subset of cells first, and then the all the '0's are simultaneously written into the remaining cells in a group.

In this section, we evaluate the ReRAM memory system using multi-bit per read/write scheme with respect to timing, energy-efficiency and area overhead in Section 3.5.1. We analyze the effect of IR drop in Section 3.5.2. We evaluate the reliability and Bit Error Rate (BER) in Section 3.5.3 and Section 3.5.4, respectively.

### 3.5.1 Latency and Energy Evaluation

We evaluate a memory system with I/O width of 64 bits in terms of area, energy consumption and latency. We consider *NB* values of 1, 4, 8, 16 and 32. *NB* = 1 corresponds to the baseline system where 8 mats with 8 subarrays per mat are activated to match the I/O width.

TABLE 3.3. COMPARISON OF AREA, ENERGY AND LATENCY FOR 1GB MEMORY WITH DIFFERENT NUMBER OF BITS PER READ/WRITE

| ***NB*** | **Activated Mats; Subarrays** | **Area ($mm^2$)** | **R (W) Energy Consumption (pJ)** | **R (W) Latency (ns)** |
|---|---|---|---|---|
| **1** | **8; 8** | 18.20 | 52.50 (54.49) | 18.01 (18.09) |
| **4** | **2; 8** | 18.27 | 44.31 (45.65) | 18.16 (18.29) |
| **8** | **1; 8** | 18.33 | 32.97 (37.58) | 18.22 (18.43) |
| | 2; 4 | | 36.94 (41.92) | |
| | 4; 2 | | 44.51 (51.48) | |
| | 8; 1 | | 51.43 (60.13) | |
| **16** | **1; 4** | 18.65 | 22.32 (27.25) | 18.40 (18.68) |
| | 2; 2 | | 25.21 (30.48) | |
| | 4; 1 | | 30.35 (37.31) | |
| **32** | **1; 2** | 19.07 | 18.10 (23.80) | 18.87 (18.96) |

Table 3.3 describes the area, read/write energy and read/write latency for different values of *NB*. The number of active mats and number of subarrays per mat are chosen such that the

read/write latencies are comparable. In order to support multi-bit per read/write, the driver has to be larger than the baseline case. Also more sense amplifiers are required [19]. The driver size is obtained by setting current constraint to be 15μA during SET for the cell that is farthest from the driver. The driver, based on 22nm PTM [44] transistor model, is a two staged buffer [64]. The first stage has W/L = 1 for NMOS and PMOS. The W/L of the second stage for *NB* = 1, 4, 8, 16 and 32 bits is set to 2, 3, 4, 10 and 24, respectively.

From Table 3.3, we see that energy saving is obtained by activating fewer mats and fewer subarrays per mat. First, for a given *NB*, the system with smaller number of active mats consumes lower energy; these are marked in bold in Table 3.3. To better understand the reason behind this choice, consider the case when *NB* = 16. Since the maximum number of subarrays per mat is 8 [19], we can choose between 1 mat with 4 subarrays or 2 mats with 2 subarrays per mat or 4 mats with 1 subarray per mat. The system with one active mat has 26.5% lower read energy consumption compared to the system with four active mats. Similarly, for *NB* = 8, the system with one active mat has 35.9% lower energy compared the system with eight active mats. Therefore, we always choose the memory configuration with the smallest number of active mats. The number of active mats is 8, 2, 1, 1, 1 for *NB* = 1, 4, 8, 16 and 32, respectively.

Second, a system with smaller *NB* has to activate more subarrays at a time (to match the I/O width), resulting in higher energy. For example, the system with *NB* = 8 has 37%/31% lower read/write energy and the system with *NB* = 16 has 57%/50% lower read/write energy compared to the baseline system. This is expected since the system with smaller *NB* activates more subarrays at a time, resulting in higher energy. Table 3.3 also shows that the area increases slightly with increasing *NB*. While the driver size is larger and more sense amplifiers are used,

the cell array area is significantly larger compared to driver area and so the increase is not significant. Finally, all systems have comparable read/write latency (within 2% difference) as per design requirements. The access latency increases slightly with increasing *NB* due to slight increase in H-tree routing delay.

From this study, we conclude that while all systems have comparable timing performance, systems with smaller *NB* consume more energy. The system with *NB* = 32 has the lowest energy but unfortunately the largest area. In the next sub-section, we will also show that the system with *NB* = 32 also suffers from severe reliability issues, making it an impractical choice for memory design.

### 3.5.2 IR Drop Analysis

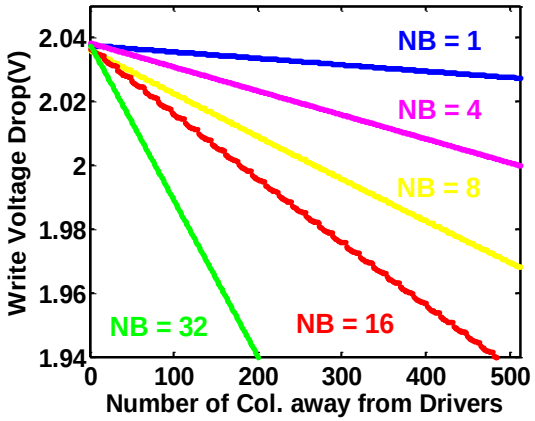



Fig. 3.6. Write voltage drop as a function of the location for different values of *NB* in a 512×512 subarray.

During read/write operations, access voltage across the selected cell decreases with increasing distance from the driver. Fig. 3.6 shows the write voltage drop on every cell (in HRS) along the row. For array size of 512×512, with *NB* = 1, the write voltage drop on the farthest cell from the driver is 99.5% of voltage drop on the nearest cell from the driver; only 0.5% voltage drop occurs in the interconnection wires. For the case when there are more bits

per write, the voltage loss in the interconnection wires is larger. For instance, for *NB* = 32, the voltage loss in wires is 12%, incurring poor reliability for the cells far away from the driver. The voltage loss for *NB* = 4, 8 and 16 is less than 5%, which is acceptable. So in the rest of the chapter, we focus on the lowest energy configurations for *NB* = 1, 4, 8 and 16.

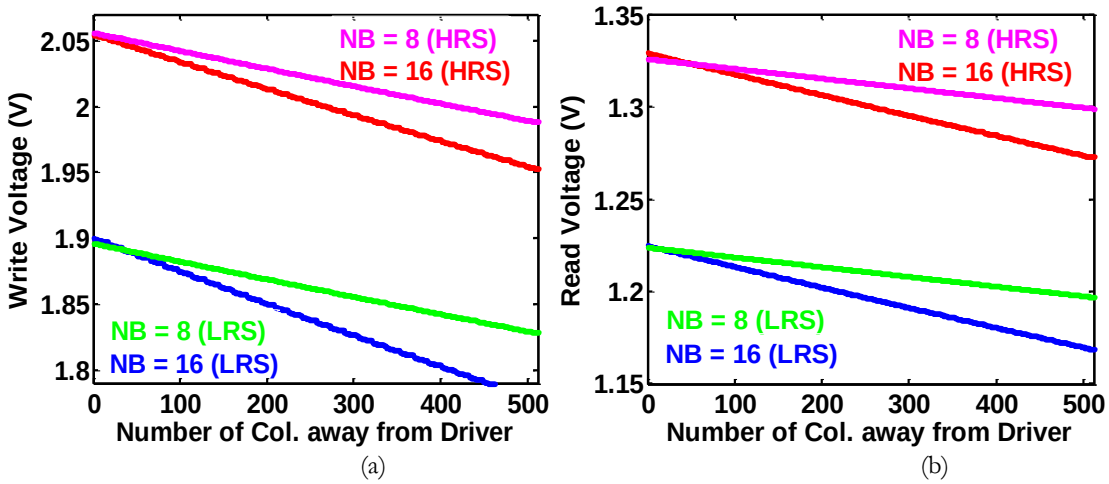



Fig. 3.7. (a) Write voltage drop and (b) read voltage drop as a function of the location for different values of *NB* in a 512×512 subarray.

Next, we show the voltage drop as a function of location of the selected cell for write and read operations. Fig. 3.7 shows how the access voltage drop on the selected cells for *NB* = 8 and 16 decreases with increasing distance from the driver. For simplicity, we show the voltage drops of HRS and LRS for *NB* = 8 and 16; the trend is the same for other values of *NB*. From Fig. 3.7, we can see that (1) larger *NB* results in larger voltage loss for both read and write in HRS as well as LRS cells. (2) For a given *NB*, voltage loss after write is larger than that after read. This is because the selected cells suffer from larger IR drop after write (compared to after read) since write voltage is larger and hence the voltage loss in interconnection is higher.

3.5.3 Reliability Analysis

In order to evaluate the reliability of the memory system, we first derive the resistance distributions by considering the effect of the different variation sources, namely D2D, C2C and IR drop. To analyze the effect of D2D variation, C2C variation and IR drop, we run $10^6$ Monte-Carlo simulations in MATLAB and SPICE. To obtain the resistance distributions due to D2D and C2C variations, we use the variation parameters in Table 3.1 and run the simulations. We assume that all groups have the same D2D and C2C variations since both these variations do not depend on the location of the device. To calculate the effect of only IR drop, we consider the mean value of resistance. To derive the combined effect of D2D, C2C and IR drop, the resistance values are picked from the resistance distributions obtained using D2D and C2C variations, and the voltage drops at every location along the row of a 512×512 1S1R array are calculated using SPICE. The voltage drops are used to calculate the net resistance values and these values are then used to derive the resistance distributions of each group.

TABLE 3.4. EFFECT OF VARIATIONS ON RERAM RESISTANCE DISTRIBUTION @ NPC = $10^6$ FOR AN $NB$ = 16 SYSTEM

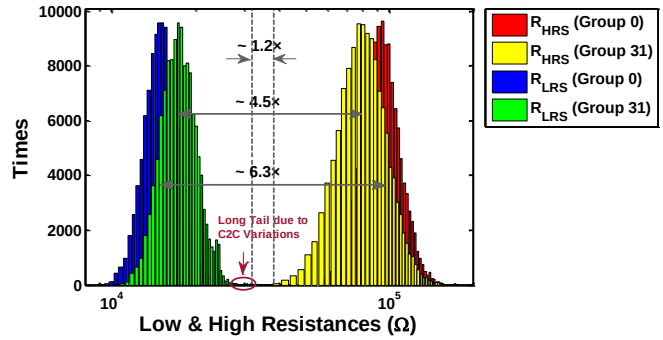
| Variation Sources | Mean OFF/ON Ratio | | Tail-to-tail OFF/ON Ratio | |
|---|---|---|---|---|
| | Group 0 | Group 31 | Group 0 | Group 31 |
| **D2D** | 15 | 15 | 3 | 3 |
| **C2C** | 6 | 6 | 1.5 | 1.5 |
| **IR Drop for Write** | 15 | 10 | NA | |
| **IR Drop for Read** | 15 | 12 | NA | |
| **Combined** | 6 | 3 | 1.5 | < 1 |

Table 3.4 first lists the effect of different variations, namely D2D, C2C, IR drop after write and IR drop after read, one by one. All groups have the same mean OFF/ON ratio of 15 and

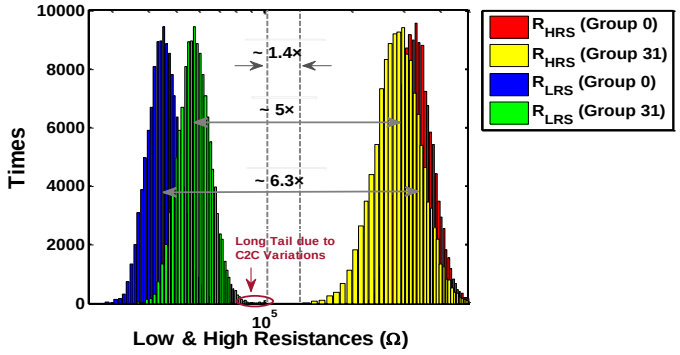
tail-to-tail OFF/ON ratio of 3 due to D2D variations. The mean OFF/ON ratio and tail-to-tail OFF/ON ratio reduce to 6 and 1.5, respectively, due to consecutive cycling. IR drop causes the group farthest away from the driver to suffer from significant reduction in mean OFF/ON ratio. Note that we list only the mean OFF/ON ratio since we only consider the mean value of $R_{LRS}$ and $R_{HRS}$ for each group. The last entry in Table 3.4 evaluates the combined effect due to all variations (including IR drop after write and read) on the mean OFF/ON ratio and tail-to-tail OFF/ON ratio of the resistance distributions.

1) Resistance Distributions After Write

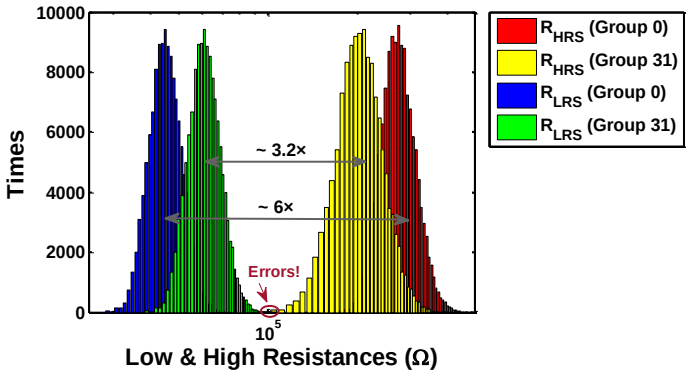
Fig. 3.8 (a) shows resistance distributions of HRS and LRS caused by D2D, C2C and IR drop after write operation in an $NB = 16$ system. The group which is closest to the driver, ie., Group 0 (is marked in blue for LRS and red for HRS) and the group which is farthest from the driver, ie., Group 31 (is marked in green for LRS and yellow for HRS). From this figure, we can find that (1) the mean OFF/ON ratio of Group 31 shrinks from 6.3 to 4.5, and in the tail-to-tail OFF/ON ratio shrinks from 1.5 to 1.2. This is because the voltage drop in the cells in Group 31 is small and so these cells cannot switch to the correct resistance value like cells in Group 0. (2) Compared to $R_{HRS}$ distribution, $R_{LRS}$ has a long tail; this is caused by C2C variation. Note that the probability of the long tail crossing into the neighboring state results in an error. (3) Group 31 for both $R_{LRS}$ and $R_{HRS}$ has wider resistance distributions compared with Group 0. The intra group voltage loss of Group 31 is larger resulting in larger BER due to C2C variations.




(a)




(b)




(c)

Fig. 3.8. Resistance distributions of HRS and LRS for Group 0 and Group 31 in an *NB* = 16 system (a) after write, (b) after read and, (c) after write and read.

2) Resistance Distributions After Read

Fig. 3.8 (b) shows resistance distributions of HRS and LRS of Groups 0 and 31 caused by D2D, C2C and IR drop for an *NB* = 16 system after read. We find that (1) mean $R_{LRS}$ increases by 29% while mean $R_{HRS}$ decreases by 12%. This is because during read operation, there is less voltage drop on $R_{LRS}$ than that on $R_{HRS}$, resulting in larger shift on the LRS distribution due to

the non-linearity of the ReRAM. (2) The mean OFF/ON ratio of Group 31 shrinks from 6.3 to 5, and the tail-to-tail OFF/ON ratio shrinks from 1.5 to 1.4. However, the tail-to-tail OFF/ON ratio of Group 31 in Fig. 3.8 (b) is larger than that in Fig. 3.8 (a). This is because the cells in Group 31 suffer from larger IR drop after write (compared to after read) since write voltage is larger and hence there is higher voltage loss in interconnection after write than after read.

3) Resistance Distributions After Write and Read

Fig. 3.8 (c) shows resistance distributions of HRS and LRS of Groups 0 and 31 caused by D2D, C2C and IR drop after write and read for an $NB = 16$ system. This corresponds to the last entry in Table IV. We find that compared to the distributions of Group 0, the mean OFF/ON ratio of Group 31 shrinks from 6 to 3 and tail-to-tail OFF/ON ratio of Group 31 is less than 1, resulting in errors. Therefore, Group 31 is highly prone to errors.

### 3.5.4 Bit Error Rate Evaluation

We used MATLAB to build a simulation environment for calculating the BER of different read groups. The BER can be calculated by the ratio of the number of failures over the total number of Monte-Carlo simulations. There are two types of failures – SET failure and RESET failure. In our case, SET failures dominate since LRS distributions shift more than HRS distributions (as shown in Fig. 3.8 in Section 3.5). Let SET failure be defined by $R_{LRS} > R_{th}$, where $R_{th}$ is $10^5\Omega$.

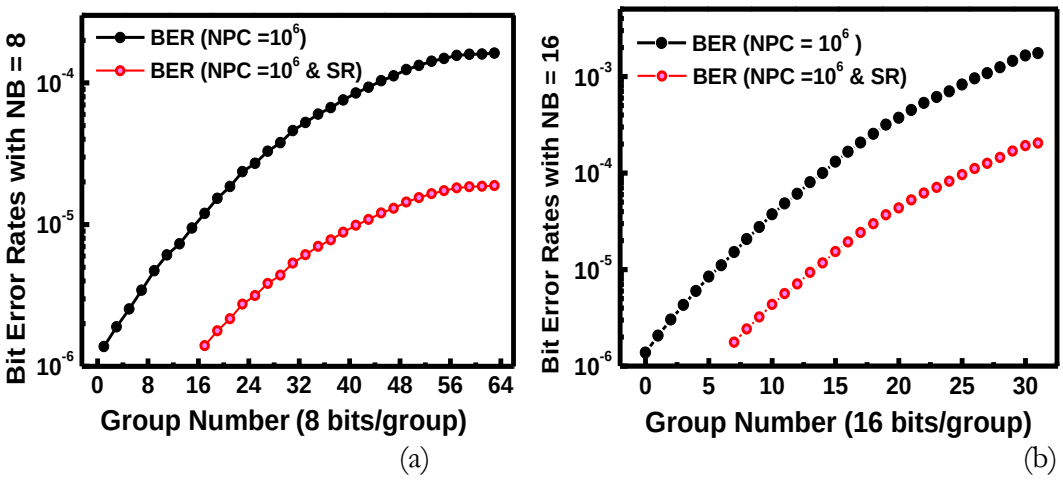



Fig. 3.9. BER for different readout groups with (a) *NB* = 8 and (b) *NB* = 16.

A group consists of *NB* bits and $n^{th}$ read/write group consists of bits from $NB \cdot n$ to $NB \cdot (n+1)-1$, where $n$ varies from 0 to $512/NB-1$. We present the error performance in terms of group BER, defined as the highest BER of *NB* consecutive bits that form a group. For example, for *NB* = 8, for Group 63, the group BER is $1.5\times10^{-6}$, which is also the BER of the farthest cell from the driver.

The BERs of 64 groups with *NB* = 8 are shown in Fig. 3.9 (a) and BERs of 32 groups with *NB* = 16 are shown in Fig. 3.9 (b). We see that BER increases as the group number increases, as expected. For *NB* = 8, the BER of Group 63 is the highest and is 100× higher than that of Group 0. For larger *NB*, the variation in BER across the groups is larger. This is because a system with larger *NB* suffers from higher IR drop than the system with smaller *NB*. For instance, for *NB* = 16, the BER of Group 31 is 2000× higher than that of Group 0. Thus, an ECC scheme that is designed to handle errors in Group 31 is an overkill for groups that are closer to the driver, such as Group 0. Also note that with SET recovery, the BER is one order of magnitude lower than the naïve multi-bit access scheme for both *NB* = 8 and 16, thereby lowering the requirement of ECC.

### 3.5.5 Write Disturbance & Read Disturbance

In this chapter, we do not consider write disturbance. The voltage drops on half-selected and unselected cells are ideally V/2 and 0, which are smaller than the threshold of FAST selector. The OFF leakage (~fA) of FAST selector [66] is so small that voltage drop on ReRAM can be ignored, resulting in immunity to write disturbance.

As for read disturbance, the cell with the highest read disturbance is the one that is closest to the driver. We find that these cells would suffer from read disturbance (BER = $10^{-5}$) only after $10^5$ consecutive read operations. Thus, read disturbance is unlikely to happen since the read/write ratio in memory applications is often around 10, and so new data is written into a cell long before any read disturbance can occur. So in the rest of the chapter, we do not take write disturbance and read disturbance into consideration.

## 3.6 Rotated Multi-array Access – A System-level Approach

From Section 3.5, we see that multi-bit groups that are farther away from the driver have higher loss in voltage, resulting in incomplete read/write operation and hence poor reliability. Thus if the data is striped across multiple subarrays, then the worst case scenario occurs when, in each subarray, the group that is farthest away from the driver is read. While the errors can be corrected by a strong BCH scheme, the area overhead due to larger parity storage is significant. To reduce the cost of ECC, we propose a new Rotated Multi-array Access (RMA) scheme where the multi-bit groups are located in different positions in each subarray.

### 3.6.1 ECC schemes

In order to make the cross-point ReRAM system reliable, ECC will always designed for the worst case (such as Group 63 for $NB = 8$ or Group 31 for $NB = 16$), resulting in over-design for the rest of groups. Here we use Block Failure Rate (BFR) as the reliability metric and set a

constraint of BFR = $10^{-10}$, which corresponds to a lifetime of 10 years [12]. We derive the BFR from BER by using the following equation [51]:

$$BFR = P(error > t) = \sum_{i=t+1}^{n} \binom{n}{i} BER^{i}(1 - BER)^{n-i} \qquad (1)$$

where BER is the input to the ECC, $t$ is the correction strength of the BCH, and $n$ is the block size, which includes the 512-bit information and 10$t$-bit parity. For instance, if the number of information bits is 512 and $t$ = 7. $n$ = 512 + 7×10 = 582 bits.

We employ BCH code in this chapter since BCH has lower code rate ( = parity bits/ codeword bits) compared to Reed Solomon (RS) code for the same BFR. For example, if BER is $3.1\times10^{-4}$, to obtain BFR of $10^{-10}$, BCH $t$ = 7 code with rate of 70/582 = 12% is required compared to RS $t$ = 6 code with rate of 96/608 = 16%.

### 3.6.2 Rotated Multi-array Access Scheme

In a memory system where the I/O width is 64 bits, a data line of size 512 bits is read in 512/64 = 8 beats. Each beat here is defined as one clock tick as in commodity DRAM systems. So in each beat, 64/*NB* groups from 64/*NB* subarrays are accessed (1 group per subarray) and in each subarray, 8 groups are accessed in 8 beats. In a conventional scheme, groups at the same location in different subarrays are read. The worst case scenario corresponds to the case when the same set of 8 groups that are farthest away from the driver are read from all subarrays over 8 beats. For example, for *NB* = 16, the worst case is when groups 24 through 31 are read from all subarrays. For such a case, the BER = $2.21\times10^{-3}$ and a strong ECC (BCH with $t$ = 14) is required to guarantee BFR of $10^{-10}$. The best case scenario corresponds to the case when Groups 0 through 7 are read from all subarrays. Since the BER is only $6.4\times10^{-6}$ for this case, BCH with $t$ = 3 would have been sufficient.

Since the data line size is 512 bits and I/O width is 64 bits, total *NG* groups where *NG* = 512/*NB* are accessed in 512/64 = 8 beats to obtain 512-bit data. For every beat, *M* groups are read out from *M* subarrays to obtain 64-bit data, where *M* = 64/*NB*. Note that these *M* subarrays could be activated in one mat (when *NB* ≤ 4) or multi-mat (when *NB* ≥ 8). To avoid the larger BER difference between the best case and worst case scenarios, we propose to access the *NG* groups located in *NG* different positions across the *M* subarrays. We refer to this scheme as Rotated Multi-array Access (RMA) scheme. An important feature of this access scheme is that all data accessed from multiple subarrays have the same error characteristics. Moreover, the resulting BER is lower than the conventional multi-bit access scheme. Thus, a lower cost BCH code can be used to achieve the same level of reliability resulting in lower area and energy overhead.

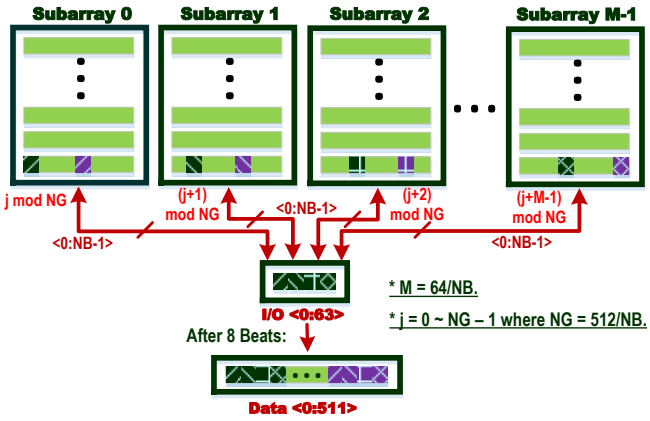



Fig. 3.10. Rotated Multi-array Access (RMA) scheme.

A high level diagram of RMA scheme is shown in Fig. 3.10. In the $k^{th}$ beat, one group from each subarray is read out, namely, Group $j$ *mod NG* from subarray 0, Group $(j+1)$ *mod NG* from subarray 1, Group $(j+2)$ *mod NG* from subarray 2 and Group $(j+M-1)$ *mod NG* from subarray $M-1$, where $0 \leq j \leq NG - 1$ and $k$ is the beat number that goes from 0 to 7. Thus, after 8 beats, *NG* groups (Group 0 to Group *NG*-1) are read out, from different physical

locations in the $M$ subarrays. The BER for the 512 bits that were read out in this way is $3.1\times10^{-4}$, which is almost one order of magnitude lower than that of the naïve scheme.

An alternate scheme that also reads from different groups residing in different physical locations across $M$ subarrays accesses Groups $j \bmod NG$ through $(j+7) \bmod NG$ from subarray 0, Groups $(j+8) \bmod NG$ through $(j+15) \bmod NG$ from subarray 1, Groups $(j+16) \bmod NG$ through $(j+23) \bmod NG$ from subarray 2 and Groups $(j+8M-8) \bmod NG$ through $(j+8M-1) \bmod NG$ from subarray $M-1$. Both schemes have the same BER characteristics and comparable routing overhead. Finally, for the case when consecutive bit-lines share a sense amplifier, bit-interleaving can be employed on top of RMA, resulting in lower routing complexity.

### 3.6.3 Evaluation

Table 3.5 compares the area, read/write energy and latency for the different configurations. It also lists the BER and the BCH code that is required to guarantee BFR of $10^{-10}$. The BER for different groups is obtained by Monte-Carlo simulations in MATLAB and presented in Fig. 3.9. Conventional system with *NB* bits per access does not implement SET recovery or RMA scheme. The baseline system is the conventional system with $NB = 1$. The BER for the baseline system is the BER of the rightmost bit. The BER for conventional systems with $NB > 1$ is the average BER among the 8 rightmost Groups $NG - 8$ to $NG - 1$. The system with SET Recovery (SR) has one order of magnitude lower BER than conventional system (see Fig. 3.9). The BER for the proposed system with RMA scheme is calculated by taking the average BER among all groups and is thus an order of magnitude lower.

Table 3.5 also lists the required BCH code for each system calculated by (1) and the corresponding area overhead and decoding latency of the ECC unit obtained from [57].

Implementation of BCH code with different values of $t$ consumes different area and delay. For instance, BCH $t$ = 4, 7 and 14 has decoding circuit area of 0.06, 0.08 and 0.1 $mm^2$ and delay of 2.3, 3.4 and 7.7ns, respectively. Thus, decoding circuit area is quite small (< 0.5% of total area) and can be ignored. Use of a BCH code with small $t$ results in low parity storage. For instance, the baseline system requires BCH $t$ = 4 code and has parity storage of 7.2%. In contrast, the conventional $NB$ = 16 system requires BCH $t$ = 14 and has parity storage of 21.5%.

TABLE 3.5. COMPARISONS OF AREA, ENERGY CONSUMPTION AND LATENCY OF DIFFERENT ARRAY LEVEL ACCESS SCHEMES

| Read/Write Methods | BER | Required BCH [27] | Total Area ($mm^2$) | R (W) Energy (pJ) | R (W) Delay (ns) |
|---|---|---|---|---|---|
| **$NB$ = 1 (Baseline)** | $1.6\times10^{-5}$ | $t$ = 4 | 19.30 | 318.4 (335.5) | 63.5 (78.1) |
| **$NB$ = 4** | $5.8\times10^{-5}$ | $t$ = 5 | 19.51 | 276.5 (284.9) | 63.7 (78.2) |
| **$NB$ = 4 + SR+ RMA** | $1.5\times10^{-6}$ | $t$ = 2 | 18.95 | 260.7 (362.8) | 63.7 (106.2) |
| **$NB$ = 8** | $1.4\times10^{-4}$ | $t$ = 6 | 19.71 | 208.8 (244.4) | 63.9 (78.4) |
| **$NB$ = 8 + SR+ RMA** | $5.6\times10^{-6}$ | $t$ = 3 | 19.16 | 197.8 (300.7) | 63.9 (106.5) |
| **$NB$ = 16** | $2.1\times10^{-3}$ | $t$ = 14 | 24.18 | 169.9 (175.2) | 64.4 (78.6) |
| **$NB$ = 16 + SR** | $2.4\times10^{-4}$ | $t$ = 7 | 20.33 | 144.0 (236.5) | 64.4 (107.8) |
| **$NB$ = 16 + RMA** | $3.1\times10^{-4}$ | $t$ = 7 | 20.33 | 144.0 (175.2) | 64.4 (78.6) |
| **$NB$ = 16 + SR+ RMA** | $2.6\times10^{-5}$ | $t$ = 4 | 19.68 | 124.8 (202.5) | 64.4 (107.8) |

The total memory area includes the area of cell array, peripheral circuits, parity storage and ECC unit. For the proposed system with $NB$ = 16, the breakdown is cell array area of 17.2$mm^2$, peripheral circuits area of 1.05$mm^2$, parity storage area of 1.35$mm^2$ and ECC area of 0.08$mm^2$. Energy consumption and latency are estimated by NVSim. These correspond to read/write of 512-bit data. The read latency here includes the latency of the syndrome calculation (0.5ns),

which is very small compared to the data read latency. The write latency does not include the encoding latency since it can be always hidden in the pipeline.

All systems have comparable timing performance, which depends on read latency. Note that write latency has little effect on timing performance since it can be hidden by use of the multi-level caches [12]. We evaluate all systems by weighing two metrics – area overhead and energy consumption. To achieve the same lifetime (BFR of $10^{-10}$) of different systems, different strengths of ECC are employed. Conventional systems with larger *NB* suffers from reliability issues and hence require stronger ECC, thereby incurring larger parity storage and higher memory area. Compared to the baseline, the conventional scheme with *NB* = 8 improves energy-efficiency for read (write) by about 34% (27%) at the price of 2% area overhead. In contrast, the system with *NB* = 16 has lower read (write) energy by 46.6% (47.8%) compared to the baseline scheme, it has 25.3% extra area overhead which is unacceptable.

For *NB* = 16, circuit-level optimization (SET Recovery) or system-level RMA scheme relaxes the ECC requirement from BCH $t = 14$ to BCH $t = 7$. The system with SET Recovery has higher energy and lower performance than the system with RMA scheme so that a system with SR alone would not be taken into consideration. The candidate system with SET Recovery at circuit level and RMA scheme at system level requires BCH $t = 4$ code instead of BCH $t = 14$ code. Use of a smaller code helps reduce the area and read/write energy due to lower parity storage compared to the conventional *NB* = 16 system.

Figure 3.11 illustrates the memory area and energy of different systems based on read/write ratio of 10. As shown in Fig. 3.11 (a), the memory area increases with increasing *NB*. This is because the system with larger *NB* has lower reliability and hence requires stronger ECC to maintain BFR of $10^{-10}$. The area differs from system to system due to additional parity

storage and peripheral circuits. For example, compared to the baseline system, the conventional system with *NB* = 16 has 25.3% higher area consumption due to use of BCH *t* = 14 ECC. Circuit-level optimization (SR) and system-level RMA scheme help system with multi-bit per access (*NB* > 1) to maintain same reliability with little additional area. For example, compared to the baseline system, the proposed systems with *NB* = 16 only has 2% area penalty. Fig. 3.11 (b) compares the energy consumption of the different systems. We see that the energy decreases with increasing *NB*. We find that with the multi-layer techniques, while the energy consumption reduces slightly for systems with *NB* = 4 and 8, for *NB* = 16, the energy consumption reduces by 59%. After weighing two metrics – area and energy-efficiency, the proposed ReRAM system (*NB* = 16) with multi-layer technique is the best option. It has the lowest energy consumption, which is, only 41% of the baseline system, with only 2% area penalty.

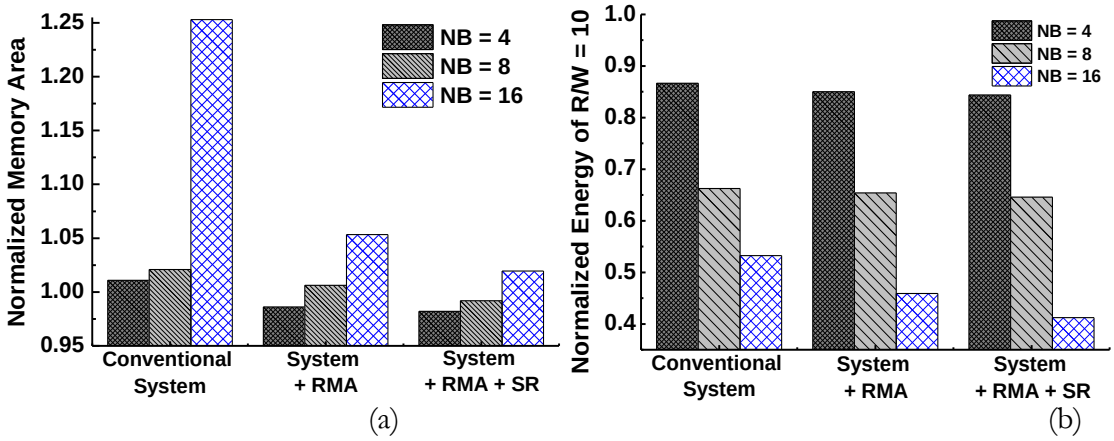



Fig. 3.11. (a) Memory area and (b) energy of different systems for read/write = 10 normalized to that of the baseline system (*NB* = 1).

## 3.7 Conclusion

In this chapter we propose a multi-layer technique to improve energy-efficiency and reliability of ReRAM cross-point systems with minimum area and latency overhead. At the cell level, we find that the errors due to temporal variations are dominated by SET failures, which can be significantly reduced by SET recovery. In contrast to existing systems which are based on single bit per read/write, we propose to use multi-bit per read/write. At the array level, we show that the system with multi-bit per read/write has very high energy-efficiency but lower reliability due to voltage loss in interconnect wires. We study the resistance distributions due to different variation sources and evaluate the corresponding Bit Error Rate (BER). Since the BER for a group with multi-bit which is far away from the driver is much higher than a group near the driver, the ECC has to be designed for the worst case scenario when the data access only includes groups that are far away from the driver. So we propose RMA scheme, a new data access scheme where the data is striped across multi-array such that the constituent multi-bit groups are located in different positions in each subarray. We show that if the group size is 16 bits, then the RMA scheme based system can reach BFR of $10^{-10}$ by using BCH $t = 4$ code instead of BCH $t = 14$ code that is needed for the naïve multi-bit access scheme. Simulation results using NVSim show that the proposed scheme for ReRAM system with multi-bit per read/write outperform a system with single bit per read/write in terms of energy while maintaining latency and reliability with only a small area overhead.

# CHAPTER 4

# IMPROVING RELIABILITY OF 1S1R RERAM 3D SYSTEM

## 4.1 Introduction

The key challenge in competing with NAND flash for storage class memory is ReRAM's lower integration density and thus higher cost per bit. To reduce cost per bit, 3D cross-point ReRAM architecture has been widely studied. By simply stacking the cross-point ReRAM cells layer by layer [21-24], the integration density of ReRAM can be increased. In the corresponding approach referred to as 3-D horizontal ReRAM (3D-HRAM) [25], [26], the adjacent layers share the word lines (WLs) and bitlines (BLs). An alternative to 3D-HRAM is the 3-D vertical ReRAM (3D-VRAM), which has higher cost efficiency but suffers from several fabrication-related issues, e.g., high aspect-ratio pillar etching for multiple metal/dielectric stacks, selector integration on the sidewall, etc. Since 3D-HRAM is a more mature technology with two-layer chip-scale demonstrations [21-24], we focus on this 3-D structure in this chapter.

In this chapter, we present a full stack approach (from cell to array to system) to analyze latency, energy and reliability of a 3D-HRAM system. Our evaluations are based on accurate SPICE models of ReRAM cell and 3D array. We focus on 3D-HRAM cross-point array system where each subarray is a multi-layered structure (16 layers). We propose to access multiple subarrays with multiple layers in a subarray to achieve high energy-efficiency and good reliability. We extend the RMA scheme for 2D cross-point array developed in [20] to improve the reliability of multi-layered 3D cross-point array. We also propose two low cost read/write schemes that utilize multi-layer programming to achieve high energy-efficiency. To guarantee system-level reliability represented by Block Failure Rate (BFR) of $10^{-10}$, we make use of BCH codes. We provide a thorough evaluation of competing 3D-HRAM systems in terms of energy,

latency and reliability. We also evaluate the scalability of the 3D-HRAM system with respect to I/O width and subarray size. To the best of our knowledge, this is the first comprehensive work on design and analysis of 3D-HRAM systems.

The rest of this chapter is organized as follows. In Section 4.2, we review the ReRAM basics, including cell basics, array architecture, 3D system organization and reliability characteristics. We summarize existing work in Section 4.3. In Section 4.4, we describe how the proposed MAS-I and MAS-II schemes can be used to implement multi-layer access, thereby improving energy-efficiency. In Section 4.5, we show how *NB* and *NL* affect reliability. This is followed by system-level evaluation of the proposed 3D-HRAM system with respect to area, performance, energy, and reliability in Section 4.6. We conclude the chapter in Section 4.7.

## 4.2 Background

### 4.2.1 Cross-point ReRAM Array Architecture

#### 1) Planar Structure

There are two types of ReRAM array architectures: the 1- transistor-1-resistor (1T1R) structure and the cross-point structure. We choose the cross-point structure because it is more area-efficient than 1T1R [1]. In the proposed cross-point structure, a two-terminal selector device is added in series with the ReRAM cell at each cross-point so that the sneak path current of the unselected cells can be cut off [1]. 1S1R has the same area as cross-point (= $4F^2$) since the selector device is vertically stacked with the ReRAM cell.

**Reliability Issues:** In this paper, we consider endurance issues due to shift in the resistance distribution as well as system-level issues due to IR drop and sneak path. IR drop along the interconnection wires becomes significant when the WL and BL wire width scales

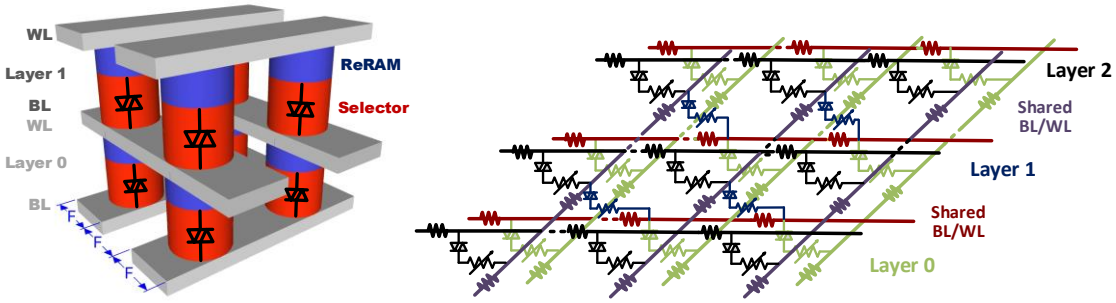
in sub-50 nm regime [1]. Also, sneak path through the half-selected cells and unselected cells causes an extra voltage drop that can lead to an insufficient voltage at the selected cell required for a successful read/write [4]. We focus on errors due to SET failures since these failures result in a shift in the LRS distribution which is significantly larger than the shift in the HRS distribution.

We have not considered retention degradation and read/write disturbance in our analysis. Retention is not an issue for continuous read/write operations considered here. There is no write disturbance since the OFF leakage of threshold-type selector is very small and consequently the voltage drop on ReRAM cells is negligible. Read disturbance starts affecting only after $10^5$ consecutive read operations which is an improbable scenario, and hence has not been considered.

We have not considered errors due to thermal crosstalk between neighboring cells as well. We built a lumped RC model for 3D-HRAM system in SPICE and found that the errors for unselected cells due to thermal cross talk are quite small. The thermal crosstalk is defined by the temperature difference before and after disturbance ($\Delta T = 200K$). Since the time interval between two continuous WRITE operations is ~ 2ms based on SPEC2006 benchmarks, there is sufficient time for the cells to cool down resulting in no thermal-related errors.

2) Three-dimensional (3D) Structure

In a 3D-HRAM, the planar cross-point structures are stacked layer by layer, as shown in Fig. 4.1 (a). It increases bit density to $0.25L$ b/$F^2$ where $L$ is the number of layers in 3D-HRAM. There are now 3D-VRAM designs with 4 to 16 layers [73, 74]. 3D-HRAM is a mature technology and so we anticipate that it will be able to support more layers in the near future. So in this chapter, we focus on the 16-layer 3D-HRAM cross-point array architecture.




(a) (b)

Fig. 4.1. (a) Schematic of 3D 1S1R array (adapted from [20]); (b) SPICE schematic of 3-layer RRAM 1S1R array.

The 3D-HRAM array schematic is shown in Fig. 4.1. For simplicity, only two memory layers are shown in Fig. 4.1 (a) [75]. Each layer is essentially a cross-point array; where two adjacent layers share WL or BL. For instance, the BLs of the top layer serve as the WLs of the bottom layer. In general, WL of Layer *i* also serves as the BL of Layer *i*+1. We develop a circuit model of the 3D-HRAM array in SPICE, shown in Fig. 4.1 (b).

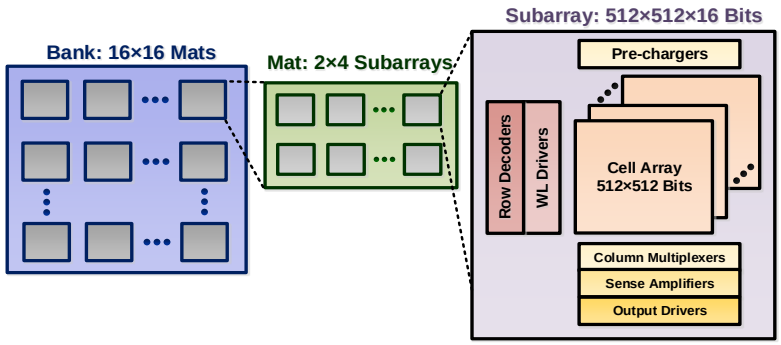



Fig. 4.2. A hierarchical memory organization with one bank, 16×16 mats per bank, and 8 subarrays per mat (adapted from [64]).

### 4.2.2 Cross-point ReRAM System Organization

Figure 4.2 shows the cross-point ReRAM system organization supported by NVSim [64]. A 1GB bank consists of 16×16 mats, where each mat consists of 2×4 subarrays, and each subarray is of size 512×512×16 (16 layers where each layer has 512×512 bits). Each subarray

has its own set of peripheral circuitry, specifically, row decoders, column multiplexers, sense amplifiers and output drivers. These components had to be redesigned to enable multi-layer programming in 3D-HRAM system and will be discussed in Section 4.4. As NVSim does not support 3D-HRAM subarray, we obtain energy consumption of the 16-layer 3D-HRAM structure using SPICE. Then, we input the corresponding results in NVSim to analyze the performance and energy consumption of the 1GB system.

TABLE 4.1.
PARAMETER SETTINGS FOR 1S1R 3D-HRAM SYSTEM

| | **Parameters** | **Notes** |
|---|---|---|
| **ReRAM** | $V_{SET}$ = 1.7V; $\tau_{SET}$ = 10ns | Mean OFF/ON Ratio = ~15; Tail-to-tail OFF/ON Ratio = ~3 |
| | $V_{RESET}$ = -1.7V; $\tau_{RESET}$ = 10ns | |
| **Selector** | Type: Threshold Selector | $0.5V_{SET} < V_{TH} < V_{READ} < V_{SET}$ |
| | $V_{TH} \pm \Delta V_{TH}$: 1.0V ± 0.1V | $0.5V_{SET} < V_{TH} - \Delta V_{TH}$ |
| | Non-linearity: 500 | When $V < V_{TH} - \Delta V_{TH}$ |
| | $V_{READ}$: 1.2V | $V_{READ} > V_{TH} + \Delta V_{TH}$ |
| **1S1R 3D Array** | Subarray Size: 512×512×16 | Bit-cell Area = $4F^2$ = 1936nm$^2$ |
| | The Number of Bits per read/write (*NB*): 1, 8, 16 and 32 | Group Size = 1, 8, 16, 32 bits Interleaving Access Scheme |
| | The Number of Layers per read/write (*NL*): 4/4 or 8 | Multi-layer Access Scheme Extended RMA |
| | $V_{WRITE}$ ($V_{READ}$): 3.5V (2.5V) | Boosted due to IR Drop |
| | Wire Resistance per Length: 1 Ω | Copper, L = 2F, S = $1.6F^2$ |
| | Wire Capacitance: 0.278 fF/μm | Wires: (Bit-line and Word-line) |
| | W/L of the Driver: 10 Technology Node: 22nm | W/L of NMOS = W/L of PMOS |
| | 22nm_LP PTM | Driver Transistor |
| | Sense Amplifier: Current-mode | Sense Speed = ~10ns |

In the ReRAM organization [64], a subset of mats and a subset of subarrays within each mat can be activated at the same time. Activating multiple mats and multiple subarrays per mat improves the timing performance at the expense of higher energy. While similar time performance can be achieved by activating multiple (say K) subarrays in one mat versus K

mats with one subarray per mat, the energy of activating multiple mats is significantly higher and not encouraged; corresponding results will be in Section 4.5.

### 4.2.3 Simulation Settings for ReRAM Cell and Array

All SPICE results presented in this paper are based on an ReRAM device compact model [65] calibrated by IMEC's $HfO_2$ ReRAM [43] with predictive 22-nm technology node [44]. The threshold voltage ($V_{TH}$) of FAST is set at 1.0 V and the tolerance for $V_{TH}$ variation in selectors, $\Delta V_{TH}$ is set at 0.1 V. $V_{READ}$ (= 1.2 V) is set to be larger than $V_{THMAX} = V_{TH} + \Delta V_{TH}$ (= 1.1 V) during read operation to ensure that there is enough readout current to sense the status of the selected cells. In order to guarantee that all half-selected and unselected cells remain OFF state during write operation, $0.5 \times V_{WRITE}$ (= 0.85 V) should be less than $V_{THMIN} = V_{TH} - \Delta V_{TH}$ (= 0.9 V). The current-mode sense amplifier has a sensing speed of ~10ns [68].

Table 4.1 shows the summary of parameter settings of ReRAM cell, selector, and array configurations. To guarantee a successful read/write in the cross-point array, $V_{READ}$ and $V_{WRITE}$ have to be boosted to compensate for the IR drop [1]. For array size of $512 \times 512 \times 16$, $V_{DD}$ is boosted from 1.2 to 2.5 V for read and from $\pm 1.7$ to $\pm$ 3.5 V for write to ensure that the farthest cell from the driver can be accessed successfully.

## 4.3 Related Work

Most of the earlier work on 3D ReRAM cross-point array focused on device, circuit and array level issues for 3D-VRAM [73, 76-78]. At the device level, the work included design and analysis of interconnection/contact geometry and ReRAM cell geometry to improve integration density [76-78]. At the circuit level, an analysis of read/write margin and power consumption found that reducing the voltage applied on unselected WL improves the read margin but at the expense of higher leakage current and hence higher total power [4]. The new

write bias scheme in [78] that resulted in reduced voltage drops on un-selected and half-selected cells was shown to achieve energy-efficiency as high as that of 1/2 voltage bias scheme and write margin as large as that of 1/3 voltage bias scheme. At the array level, there has been work on selecting array geometry (the total number of layers and array size) as well as designing a multi-bit write strategy to lower energy consumption while achieving higher bandwidth [73, 76]. The design analysis in [77] showed how array geometry impacts 3D V-RAM reliability in terms of IR drop; however, there was no system-level reliability analysis of the array level design choices.

## 4.4 Multi-bit/Multi-layer Access Schemes

TABLE 4.2.
PARAMETER SUMMARY

| Parameters | Definitions | Range |
|---|---|---|
| ***IO*** | I/O width | 64, 128 |
| ***N*** | The number of WLs, BLs | 512 |
| ***NB*** | The number of bits accessed in each group | 1, 8, 16, 32 |
| ***NG*** | The number of accessed groups in each beat | $NG \cdot NB = IO$ |
| ***NL*** | The number of accessed layers in each subarray | 1, 2, 4, 8 |
| ***NS*** | The number of accessed subarrays in each mat | 1, 2, 4 |
| ***NM*** | The number of accessed mats | 1, 2 |
| **Eqn. 1** | $NB \cdot NL \cdot NS \cdot NM = IO$ | |

We summarize some of the important parameters in Table 4.2. If the data line size is 512 bits, the I/O width is 64 bits and the number of bits in a group is *NB*, then a total of 512/*NB* groups are accessed in 8 beats to obtain 512-bit data. In every beat *NG* groups are read out from *NM* mats with *NS* subarrays per mat, where each subarray spans *NL* layers. If $NB = 8$ and $NM = 1$, we can choose between 1 subarray with 8 accessed layers ($NS = 1$, $NL = 8$) or 2 subarrays with 4 accessed layers per subarray ($NS = 2$, $NL = 4$) or 4 subarrays with 2 accessed

layers per subarray ($NS$ = 4, $NL$ = 2). Each of these configurations have different latency, energy and reliability, as will be demonstrated in Sections 4.4 and 4.5.

**Baseline 3D-HRAM System:** The 3D-HRAM system accesses multiple subarrays ($NS$) with only one layer per subarray being activated at a time ($NL$ = 1). Furthermore, $NB$ consecutive bits are accessed from a layer in the subarray and the location of these bits are the same across all the subarrays.

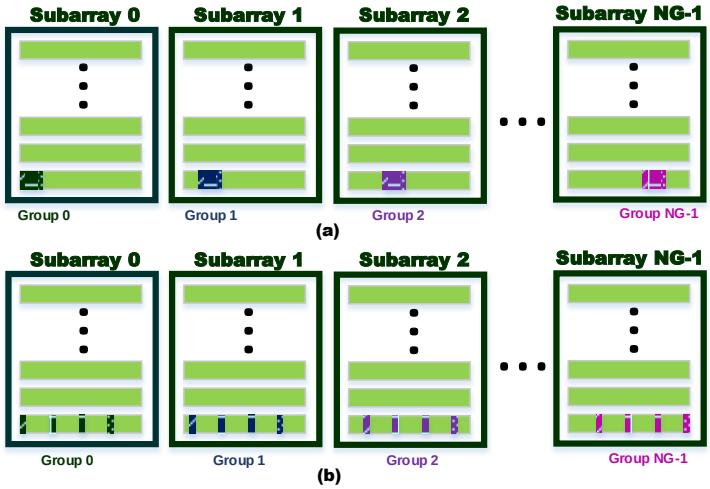



Fig. 4.3. (a) "Multi-bit group" access using RMA [20] and (b) "multi-bit interleaved group" access using proposed scheme.

### 4.4.1 Multi-bit Access Scheme

To improve energy-efficiency in 2D ReRAM system, multi-bit per access schemes have been suggested in [18, 19, 20]. While the focus had been on the design of peripheral circuits to support multi-bit per access in [18, 19], in our previous work [20], we considered reliability issues due to IR drop. Specifically, we proposed Rotated Multi-array Access (RMA) scheme, where the multi-bit groups in a data line are retrieved from different locations in each subarray, as shown in Fig. 4.3 (a). In a system where the data line is 512 bits and I/O width is 64 bits, after 8 beats, a total of 8·$NG$ groups (Group 0 to Group 8·$NG$−1) are read out, from different physical locations across $NG$−1 subarrays. Such an access pattern guarantees that the error

characteristics of all data lines are the same and the BER is one order of magnitude lower than the naïve multi-bit access scheme [20]. However, one drawback of the method in [20] is that since each group consists of *NB* consecutive bits, each sense amplifier has to be shared by every $NB^{th}$ bit-line (BL), resulting in high routing complexity.

So in this paper, the *NB* bits in a group are no longer consecutive; instead, they are spaced 512/*NB* bits apart, as shown in Fig. 4.3 (b). The $n^{th}$ bit-interleaved group consists of bits $n + NG \cdot nb$, where *nb* varies from 0 to *NB*-1 and *n* varies from 0 to 512/*NB*-1. So for *NB* = 8, group 63 consists of bits 63, 127, 191, 255, 319, 383, 447 and 511.

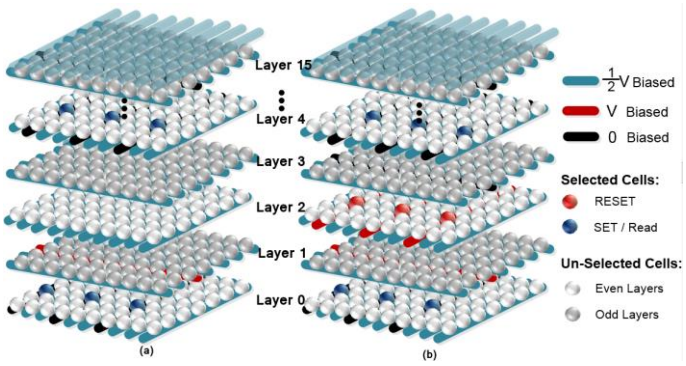



Fig. 4.4. (a) Multi-layer Access Scheme I (MAS-I) which accesses every 4th layer simultaneously; (b) Multi-layer Access Scheme II (MAS-II) which accesses all odd/even number of layers simultaneously at the price of higher read latency.

In the proposed 3D-HRAM system, each subarray is a 512 × 512 × 16 memory array (16 layers with 512×512 per layer). In each beat, every accessed subarray provides data from *NL* groups (one group per accessed layer) with *NB* bits per group. Here RMA [20] is applied not only across different subarrays but also across different layers within a subarray; we refer to this as extended RMA. For instance, if I/O width is 64 bits, the system with *NB* = 8 and *NL* = 4, accesses 2 subarrays (*NS* = 2) if *NM* = 1. In the first beat, 32 bits come from subarray 0 and another 32 bits come from subarray 1. Specifically, in subarray 0, Group 0 is accessed in Layer 0, Group 1 in Layer 4, Group 2 in Layer 8, and Group 3 in Layer 12. Similarly, in

subarray 1, Group 4 is accessed in Layer 0, Group 5 in Layer 4, Group 6 in Layer 8, and Group 7 in Layer 12. In the second beat, in subarray 0, Group 8 in Layer 0, Group 9 in Layer 4, Group 10 in Layer 8, and Group 11 in Layer 12 are accessed, and so on. By using a combination of proposed bit-interleaving in a group, and rotated access across subarrays and across layers in a subarray, the error characteristics of all data lines are the same. Such a method guarantees that a low cost ECC scheme will be sufficient to guarantee high system-level reliability.

### 4.4.2 Multi-layer Read/Write Scheme

For the subarray shown in Fig. 4.4, we choose the V/2 bias scheme for R/W access because of its lower energy consumption over V/3 bias and full bias schemes [1]. Here the WRITE operation is done in two steps with the '1's being written using SET operation followed by '0's being written using RESET operation. For SET operation, all the selected WLs and BLs are set to '$V_{WRITE}$' and '0', shown in red and black lines in Fig. 4.4, respectively. For the RESET operation, the bias conditions on WL and BL are '0' and '$V_{WRITE}$' to enable bipolar switching. In both SET and RESET operations, all the unselected WLs and BLs are set to '$V_{WRITE}/2$', shown in blue lines. Bias condition for read operation is similar to that for SET operation with all the selected WLs being set to $V_{READ}$ instead of $V_{WRITE}$.

Next, we describe two competing R/W access schemes with the same *NB* and I/O width but different number of active layers(*NL*). In the first scheme, for WRITE operation, the '1's can be written using SET operation in every 3rd layer (Layers 0, 3 …) and then the '0's can be written using RESET operation in the same set of layers (Layers 0, 3 …). For READ operation, the selected groups located in every 3rd layer can be accessed in one step. For ease of addressing, we choose to activate every 4th layer (instead of every 3rd layer) so that the '1's

and '0's are written in Layers 0, 4, 8, 12 or Layers 1, 5, 9, 13, and so on. This scheme, where every 4th layer is accessed simultaneously, is referred to as MAS-I.

To improve energy-efficiency, we propose Multi-layer Access Scheme II (MAS-II), which enables accessing larger number of layers at the expense of read performance degradation. In MAS-II, for WRITE operation, the '1's are written using SET operation in every 4th layer (for instance, Layers 0, 4, 8 and 12). At the same time, the '0's are written using RESET operation in Layers 2, 6, 10 and 14. For a given fixed I/O width of 64 bits, both MAS-I and MAS-II have the same write throughput of 64 bits. However, since MAS-I activates 2 subarrays while MAS-II activates only 1 subarray, MAS-II has higher energy-efficiency.

1) Multi-layer Access Scheme I (MAS-I)

Fig. 4.4 (a) describes the MAS-I version, where every 4th layer is accessed simultaneously. Here one WL in Layer 0 is set to 'V' to perform SET operation for the group shown in (blue bubbles). Similarly, another group (blue bubbles) in Layer 4 is selected for SET operation by setting corresponding BLs to '0' and WLs to 'V'. In order to guarantee that there is less than 'V/2' voltage drop on un-selected cells, Layers 1, 2 and 3 cannot be accessed once Layers 0 and 4 are activated. This is because of the following reasons. First, since WL/BL lines are shared across adjacent layers, accessing a group in Layer 0 sets one WL to 'V' which implies that one BL of Layer 1 is also biased at 'V'. Thus, only one bit in Layer 1 can be accessed instead of NB bits, which is not acceptable! Then, in order to guarantee less than 'V/2' voltage drop on un-selected cells in Layer 1, all WLs of Layer 1 ought to be set to a voltage ≥ 'V/2'. So in MAS-I, we set all WLs of Layer 1 to be 'V/2'. This means that the voltage bias of all BLs in Layer 2 is 'V/2', which means that data in Layer 2 can no longer be accessed. Layer 3 acts

as a dummy layer and so all BLs of Layer 3 are set to '0'. Recall that we chose to activate every 4th layer (instead of every 3rd layer) for ease of addressing.

2) Multi-layer Access Scheme II (MAS-II)

MAS-II enables every 2nd layer to be accessed simultaneously during WRITE. For instance, as shown in Fig. 4.4 (b), one WL and a few BLs (based on the data) in Layer 0 are set to 'V' and others are set to '0' to perform SET operation. The selected group in Layer 2 performs RESET by setting the selected WL and corresponding BLs of Layer 2 to '0' and 'V', which implies that corresponding WLs of Layer 1 are also biased at 'V' due to WL/BL sharing. In this way, the voltage drop for cells in Layer 1 is less than 'V/2' to avoid write disturbance. Thus, MAS-II enables write in every 2nd layer --- through SET for every 4th layer (e.g. Layers 0, 4, 8 and 12) and RESET every 4th layer (e.g. Layers 2, 6, 10 and 14).

In MAS-II, the READ operation cannot be performed in every 2nd layer. Since the bias condition for READ operation is similar to that for SET operation with $V_{READ}$, there has to be at least two unselected layers between two accessed layers. Thus, READ has to be done in two steps, where in each step, READ operates on every 4th layer. For instance, the groups from Layers 0, 4, 8 and 12 are READ first and then the groups from Layers 2, 6, 10 and 14 are READ.

Compared to MAS-I, MAS-II enables more number of layers (*NL*) to be accessed at the same time, resulting in higher energy-efficiency. This is because for a system with fixed *NB*, increasing the number of active layers (*NL*) results in fewer number of active subarrays (*NS*) or fewer number of active mats (*NM*), thereby reducing energy consumption. However, READ latency increases since READ operation has to be done in two steps, resulting in performance degradation for the system.

4.4.3 Peripheral Circuitry

As described earlier in Section 4.2, every subarray has its own peripheral circuitry (row decoders, WL drivers, column multiplexers, sense amplifiers and output drivers). The drivers and sense amplifiers that are used in the 2D design in [79] can be used here, and so this sub-section, we focus on row decoder and column multiplexer design for the 3D-HRAM system.

TABLE 4.3. PERIPHERAL CIRCUITS DESIGN PER SUBARRAY FOR 3D SUBARRAY IN 22NM TECHNOLOGY NODE

| **Peripheral Circuits** | **Row Decoder** | | | **Column Mux.** |
|---|---|---|---|---|
| | ***NL* (R; W)** | **Decoder** | **Area** | **Area** |
| **3D Subarray (512×512×16)** | 1; 1 (*NL* = 1) | 13:8192 | 512.7 $\mu m^2$ | 20.9 $\mu m^2$ |
| | 4; 4 (MAS-I) | 11:2048 | 426.4 $\mu m^2$ | 83.6 $\mu m^2$ |
| | 4; 8 (MAS-II) | | | 167.2 $\mu m^2$ |

1) Row Decoder Design

The row decoders are responsible for decoding the address bits and generating decoded signals. The WL drivers are connected with the corresponding WLs and responsible for driving the WL load. For the 3D system with *NL* = 1, (which means only one layer is selected at a time), 13:8192 row decoder is used to choose one of 8192 WLs in the active subarray. Note that there are total 512×16 = 8192 WLs – 16 layers with 512 WLs for each layer. The area of this decoder is 512.7$\mu m^2$ in 22nm technology node (according to NVSim [64]), and shown in Table III. In 3D systems with *NL* > 1, since each subarray is of size 512×512×16 and 4 layers are accessed simultaneously, the decoder is of size 11:2048. For instance, since Layers 0, 4, 8 and 12 are activated at the same time, they share the same WL drivers (shown in bold black in Fig. 4.5). Thus, the first two bits (00, 01, 10, 11) of 11-bit address are used to

select which group of 4 layers are accessed and the last nine bits of the address are used to select one WL among 512 WLs from these selected layers.

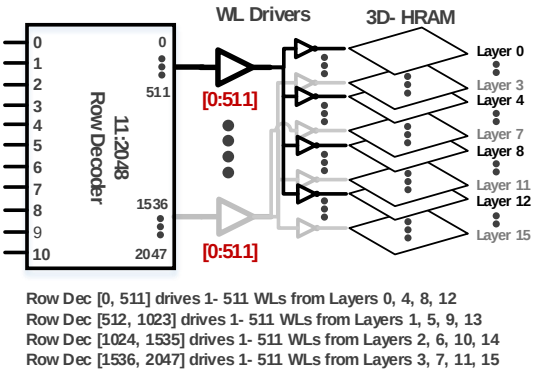



Fig. 4.5. Row decoder schematic of MAS-I for the 512×512×16 subarray.

2) Column Multiplexer Design

In each 3D-HRAM active subarray, a total of $NL \cdot NB$ bits are accessed at a time during WRITE. Since each column multiplexer along with one sense amplifier can access 1 bit per access, there are $NB$ column multiplexers in each layer and a total of $NL \cdot NB$ column multiplexers per subarray. For example, for a $NB$ = 8 and $NL$ = 4 system, 32 column multiplexers are required to access 32 bits; the corresponding area is 83.6μm$^2$ per subarray [64]. Compared to the system with MAS-I, the system with MAS-II requires 2× more column multiplexers since 8 layers are activated during WRITE instead of 4.

### 4.4.4 Latency and Energy Evaluation

We evaluate a 1GB memory system with I/O width of 64 bits in terms of area, energy consumption, and latency. We consider $NB$ values of 8, 16, and 32. We evaluated two 3D systems with $NL$ = 1, namely, one with $NM$ = 1, $NB$ = 8, $NS$ = 8, and one with $NM$ = 1, $NB$ = 16, $NS$ = 4. We choose the system with $NL$ = 1, $NM$ = 1, $NS$ = 4 and $NB$ =16 as the baseline system since it has higher energy-efficiency. Table IV shows the area, read/write

energy, and read/write latency for different values of *NB*, *NM, NL* and *NS* for MAS-I. In order to support multi-bit/multi-layer per read/write, the driver has to be larger than the baseline case and more sense amplifiers are required [64], as mentioned in Section 4.4.3. The driver size is obtained by setting current constraint for the cell that is farthest from the driver to be 15μA during SET.

Our proposed systems have lower read and write cell latencies compared to [22] but higher than those in [80]. The read and write latencies in [80] are smaller due to use of smaller subarray size and hence smaller routing delay, and better sense amplifier that have higher read-out current.

TABLE 4.4.
COMPARISON OF AREA, ENERGY AND LATENCY FOR 1GB MEMORY WITH DIFFERENT VALUES OF NB, NM, NS AND NL

| | ***NB*** | **NM; NL; NS** | **Area Footprint ($mm^2$)** | **R (W) Energy Consumption (pJ)** | **R (W) Latency (ns)** |
|---|---|---|---|---|---|
| **Array (*NL* = 1)** | **8** | **1; 1; 8** | 2.39 | 34.49 (36.04) | 18.22 (24.43) |
| | **16** | **1; 1; 4** | **2.52** | **26.45 (28.80)** | **18.40 (24.68)** |
| **3D Array using MAS-I (NL > 1)** | **8** | **1; 4; 2 (A)** | 2.15 | 26.65 (28.95) | 22.22 (28.43) |
| | | 2; 4; 1 | | 30.72 (32.67) | |
| | | 1; 2; 4 | | 32.24 (34.94) | |
| | | 2; 2; 2 | | 36.61 (38.71) | |
| | | 4; 2; 1 | | 43.18 (46.90) | |
| | **16** | **1; 2; 2 (B)** | 2.32 | 22.98 (25.49) | 22.40 (29.68) |
| | | **1; 4; 1 (C)** | | 19.05 (21.93) | |
| | | 2; 2; 1 | | 27.68 (29.82) | |
| | **32** | **1; 2; 1 (D)** | 2.51 | 15.30 (16.82) | 22.87 (30.69) |
| **MAS-II** | **8** | **1; 8; 1 (E)** | 2.33 | 23.98 (24.87) | 32.12 (29.25) |

From Table 4.4, we see that for a given *NB* and *NL*, the system with smaller number of active mats consumes lower energy. This trend is the same as in 2D ReRAM system [20]. So in the rest of chapter, we set *NM* = 1. All systems with *NL* > 1 using MAS-I have comparable read/write latency (within 2% difference) as per design requirements. The access latency

increases slightly with increasing *NB* due to slight increase in H-tree routing delay. The system with *NB* = 32 has the lowest energy but unfortunately the largest area. When MAS-II is used, for I/O width of 64 bits, there is only one possible configuration (E) with *NL* = 8, *NB* = 8 and *NS* = 1.

Next, we analyze five systems for better understanding of the impact of *NB, NL* and *NS* on energy saving.

System A: *NB* = 8, *NL* = 4 and *NS* = 2;

System B: *NB* = 16, *NL* = 2 and *NS* = 2;

System C: *NB* = 16, *NL* = 4 and *NS* = 1.

System D: *NB* = 32, *NL* = 2 and *NS* = 1;

System E: *NB* = 8, *NL* = 8 and *NS* = 1.

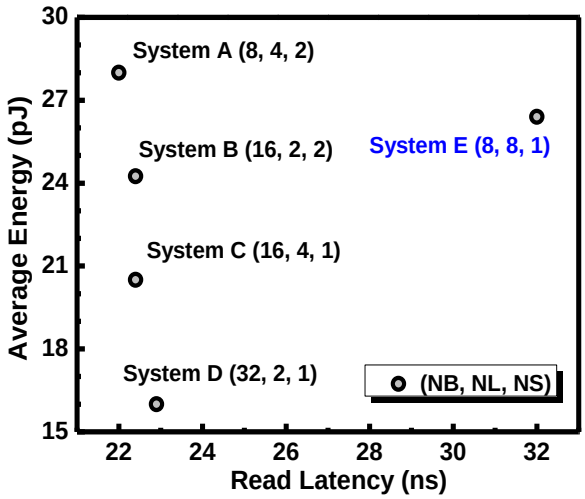



Fig. 4.6. Average energy vs read latency for different systems.

Fig. 4.6 describes energy and read performance of different systems with different values of *NB, NL* and *NS*. Note that write latency has little effect on timing performance since it can be hidden by use of the multilevel caches [20] and so, we do not consider it in Fig. 4.6. All systems with *NB* = 16 consume less energy compared with the systems with *NB* = 8. Of the

two systems that have the same *NB* = 16 (Systems B and C), System C with lower *NS* but higher *NL* has 17%/14% lower read/write energy. Of the two systems that have the same *NS* = 2 (Systems A and B), System B has smaller *NL* but higher *NB*, resulting in 14%/12% lower read/write energy. Of the two systems that have the same *NL* = 4 (Systems A and C), System C has smaller *NS* but higher *NB*, resulting in 29%/24% lower read/write energy. System D with *NB* = 32 has the lowest energy but suffers from reliability issues. All the other systems (A, B and C) exhibit tradeoffs between energy and reliability, as will be shown in Section 4.5.

From this study, we conclude that 3D-HRAM systems improve energy-efficiency by choosing larger *NB* (most effective) or larger *NL* (next most effective); larger *NM* should be avoided, followed by larger *NS*.

## 4.5 Reliability Analysis

### 4.5.1 Resistance Distributions

We evaluate the reliability of the memory system by deriving the shift in the resistance distributions caused by D2D, C2C variations at the cell level and IR drop at the system level. We run $10^6$ Monte Carlo simulations in MATLAB and SPICE to analyze the effect of the variations. First, we assume that all groups from different locations have the same D2D and C2C variations but different IR drop. This is because D2D and C2C variations do not depend on the location of the device [79]. In order to evaluate the combined effect of D2D, C2C, and IR drop, the resistance values are selected based on the resistance distributions, which are obtained using D2D and C2C variations, and the voltage drops at each location along the row of the array in each programming layer are calculated using SPICE. The voltage drops are used to calculate the net resistance values and these values are then used to derive the resistance distributions of each group.

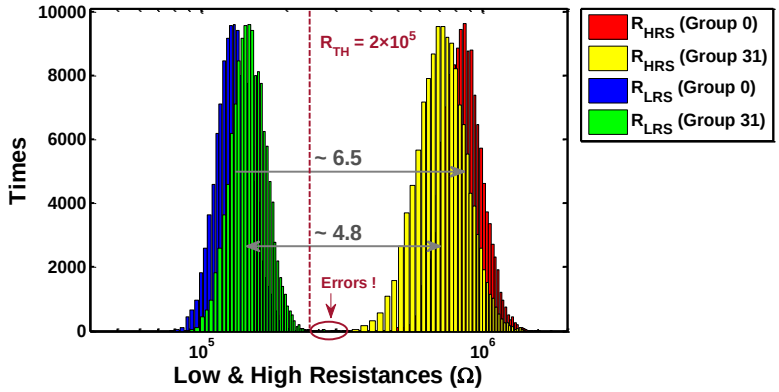
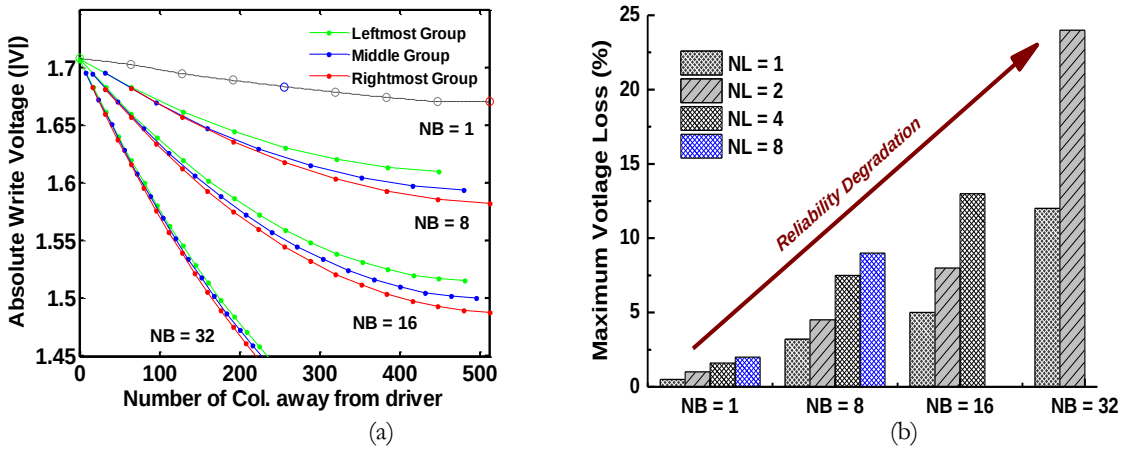



Fig. 4.7. Resistance distributions of HRS and LRS for Group 0 and Group 31 in an NB = 16 and NL = 4 system for read after write.

Fig. 4.7 shows resistance distributions of HRS and LRS caused by D2D, C2C, and IR drop for read after write operation for a *NB* = 16 and *NL* = 4 system. Group 0 starts at bit 0 and ends at bit 480 (marked in blue for LRS and red for HRS), and Group 31 starts at bit 31 and ends at bit 511 (marked in green for LRS and yellow for HRS). From Fig. 4.7, we see that the mean OFF/ON ratio of Group 31 has reduced from 6.5 to 4.8. This is because the voltage drop in the cells in Group 31 is smaller and these cells cannot switch to the correct resistance value like the cells in Group 0.

We use the overlap in the resistance distributions to calculate the bit error rates (BER) using Monte Carlo simulations. Of the two types of failures, SET failures dominate since LRS distributions shift more than HRS distributions (as shown in Fig. 4.7). Let SET failure be defined by $R_{LRS} > R_{TH}$, where $R_{TH}$ is $2\times10^5\ \Omega$. Then, the BER for Group 0 is $1.4\times10^{-4}$ and for Group 31, it is $2.8\times10^{-4}$.




(a) (b)

Fig. 4.8. (a) Write voltage drop as a function of the location for different values of *NB* in a 512×512×16 subarray with *NL* = 4; (b) Maximum voltage loss as a function of *NB* and *NL*.

4.5.2 IR Drop Analysis

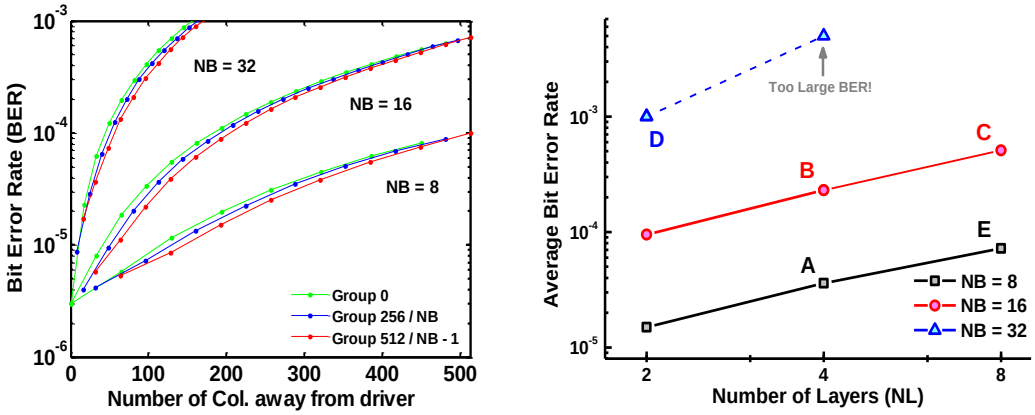
During read/write operations, access voltage across the selected cells reduces with increasing distance from the driver. Fig. 4.8 (a) shows the absolute write voltage drop on every cell (in LRS) along the row for a subarray size of 512 × 512 × 16, with *NL* = 4. When *NB* = 1, the write voltage drop on the farthest cell from the driver is 97.5% of the voltage drop on the nearest cell from the driver, thus there is only a 2.5% voltage drop due to interconnection wires. For the case when there are more bits per write, the voltage loss in the interconnection wires is larger due to more ON selectors and hence more sneak paths [20]. For *NB* = 32, the voltage loss in wires is 34%, incurring poor reliability for the cells far away from the driver. The voltage loss for *NB* = 8 and 16 is less than 7% and 12.5%, respectively.

Next, we show the maximum voltage loss, which corresponds to the voltage loss of the farthest cell from the driver, as a function of *NB* and *NL*. As shown in Fig. 4.8 (b), for fixed *NL*, the maximum voltage loss increases with increasing *NB*. For instance, for *NL* = 2, the voltage loss increases from 1% to 24% when *NB* increases from 1 to 32. In Fig. 4.8 (b), we can also see that the maximum voltage loss increases with increasing *NL* when NB is fixed.

This is expected, since as *NL* increases, more bits are accessed per WL driver, resulting in more voltage loss in the interconnection. However, a system with larger *NL* has higher energy efficiency, as demonstrated in subsection 4.4.4.

### 4.5.3 Bit Error Rates

The BER for different systems, characterized by different values of *NB*, *NL* and *NS*, is obtained by Monte Carlo simulations in MATLAB and shown in Fig. 9. The baseline system with *NB* bits per access activates only one layer at a time (*NL* = 1) and thus does not implement MAS-I or MAS-II. In the baseline system, the worst case scenario corresponds to the case when the groups that are farthest away from the driver is read from all subarrays. Thus, for the baseline system with *NB* = 16, the worst case is when groups 24 through 31 are read from all subarrays, incurring high BER of $4.7 \times 10^{-4}$.




(a) (b)

Fig. 4.9. (a) BERs for a system with NL = 4 and different values of NB as a function of the distance from the driver; and (b) average BER for systems as a function of NB and NL.

Fig. 4.9 shows how BER is affected by *NB*, *NL* and distance from the driver. From Fig. 4.9 (a), we see that BER increases as distance from the driver increases, as expected. We also find that for larger *NB*, the variation in BERs (defined as the BER difference between the

leftmost cell and the rightmost cell) across the 512 bits data line is larger. This is because a system with larger *NB* suffers from higher IR drop than the system with smaller *NB*.

Next, we present the error performance of different systems in terms of average BER (obtained by taking the average BER across all groups) in Fig. 4.9 (b). The proposed systems with lower *NB* (Systems A, E) have overall lower BER, as expected. Also, for fixed *NB*, the system with lower *NL* has lower IR drop resulting in better reliability. The systems with *NB* = 32 and *NL* ≥ 2 suffer from severe reliability issues (BER ≥ $10^{-3}$), making it an impractical choice for memory design. So, in the rest of this chapter, we do not consider System D with *NB* = 32 and *NL* ≥ 2.

### 4.5.4 Trade-offs between Energy-efficiency and Bit Error Rate

As discussed in previous sections, the systems with larger *NB* or *NL* have higher energy-efficiency but at the price of lower reliability due to larger IR drop. Fig. 4.10 shows energy and reliability of different systems with different number of *NB, NL* and *NS;* energy is normalized to that of System D with *NB* = 32. We can clearly see that the proposed systems with lower BER have higher energy consumption.

Among all systems with *NB* = 8 and 16, Systems A and E with *NB* = 8 reduce BER more significantly but at the price of much higher energy consumption than Systems B and C with *NB* = 16. This is because the system with smaller *NB* has better reliability due to smaller voltage loss in interconnection. However, it incurs higher energy consumption due to more subarrays being activated. We can also see that the system with smaller *NB* but fixed *NL* can have significantly better reliability but at the expense of higher energy consumption. For example, System A with *NB* = 8 lowers BER by 6.4× but at the price of 26% more energy consumption compared with System C with *NB* = 16. The systems with smaller *NL* and fixed

*NB* can improve reliability to some degree at the expense of higher energy. For example, System B lowers BER by 2.4× but at the expense of 15% more energy consumption compared to System C.

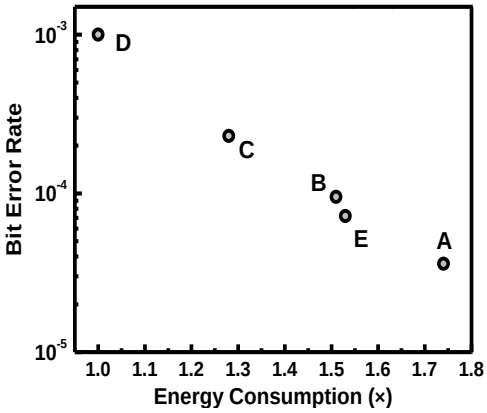



Fig. 4.10. Average energy vs reliability for different systems; the energy is normalized to that of System D.

System B based on MAS-I with larger *NB* and smaller *NL* and System E with smaller *NB* and larger *NL* based on MAS-II have comparable BER and energy consumption. This is partly because *NB* and *NL* have opposing effects on energy-efficiency and reliability.

## 4.6 System-level Analysis

In order to guarantee the same system-level reliability for all competing systems described in the earlier sections, we use BCH codes with different error correction capabilities (Section 4.6.1). Then, we analyze the area, read/write energy, and latency for the different systems that now have the same reliability in Section 4.6.2. Finally, we evaluate the performance of systems with wider I/O width or larger number of layers in Section 4.6.3.

4.6.1 ECC schemes

We use Block Failure Rate (BFR) as the reliability metric and set a constraint of BFR = $10^{-10}$, which corresponds to a lifetime of 10 years [20]. We use the relation that is used to calculate the BFR from BER to derive '*t*', the error correction strength of the BCH code [20].

$$BFR = P(error > t) = \sum_{i=t+1}^{n} \binom{n}{i} BER^i (1 - BER)^{n-i} \qquad (1)$$

Here *n* is the block size, which includes the 512-bit information and 10*t*-bit parity. For instance, if the number of information bits is 512 and $t = 4$, then $n = 512 + 4\times10 = 552$ bits. The baseline system has a high BER = $4.7\times10^{-4}$ and thus requires BCH $t = 8$ code. In comparison, System C with BER = $2.3\times10^{-4}$ requires a $t = 6$ code.

4.6.2 Evaluation

Table 4.5 compares the area, read/write energy, and latency for different systems in the 22nm technology node. Implementation of BCH code with different values of *t* consumes different decoding area and delay values. We obtain the decoding latency and corresponding area overhead of the ECC unit obtained from [57]. For instance, BCH $t = 4$, 5 and 6 has decoding circuit area of 0.06, 0.065 and 0.07 $mm^2$ and delay of 2.3, 3.2ns and 4 ns, respectively. Thus, decoding circuit area is quite small (< 0.5% of total area) and can be ignored. Use of a BCH code with smaller *t* causes in lower parity storage. For example, the baseline system requires BCH $t = 8$ code and has parity storage of 15.6%, while the System A requires BCH $t = 4$ and has parity storage of 7.8%.

The total memory (footprint) area includes the area of cell array, peripheral circuits, parity storage, and ECC unit. For System C, the breakdown is cell array area of 1.04 $mm^2$, peripheral circuitry area of 1.35 $mm^2$, parity storage area of 0.12 $mm^2$, and ECC area of 0.07 $mm^2$.

Compared with the baseline system, the proposed systems consume 17% lower area on average due to smaller peripheral circuitry and smaller parity storage area. Energy consumption and latency numbers correspond to read/write of 512-bit data and are estimated by NVSim [64]. The read latency here includes the latency of the ECC syndrome calculation (0.5ns), which is very small compared with the data read latency. The write latency does not include the encoding latency since it can always be hidden in the pipeline.

TABLE 4.5.
COMPARISONS OF AREA FOOTPRINT, ENERGY CONSUMPTION AND LATENCY OF DIFFERENT SYSTEMS OF 1GB

| 3D Array | *NB* | *NM; NL; NS* | BER | Required BCH [57] | Area Footprint ($mm^2$) | R (W) Energy (pJ) | R (W) Delay (ns) |
|---|---|---|---|---|---|---|---|
| **Baseline** | **16** | **1; 1; 4** | $4.7\times10^{-4}$ | $t = 8$ | 3.02 | 179.7 (282.9) | 84.1 (126.2) |
| **MAS-I** | **8** | **1; 4; 2 (A)** | $3.6\times10^{-5}$ | $t = 4$ | 2.31 | 156.6 (249.7) | 80.6 (122.2) |
| | **16** | **1; 2; 2 (B)** | $9.5\times10^{-5}$ | $t = 5$ | 2.55 | 133.9 (225.1) | 81.6 (122.4) |
| | | **1; 4; 1 (C)** | $2.3\times10^{-4}$ | $t = 6$ | 2.58 | 115.7 (195.9) | 82.1 (123.8) |
| **MAS-II** | **8** | **1; 8; 1 (E)** | $7.2\times10^{-5}$ | $t = 5$ | 2.54 | 140.6 (231.9) | 120.2 (176.8) |

Systems A, B and C improve both energy-efficiency and area-efficiency while maintaining latency comparable with the baseline system. For example, System C can reduce average energy by 33% with 15% smaller area for comparable read/write latency. Both Systems B and C based on MAS-I outperform System E based on MAS-II. System C achieves 16% higher energy-efficiency and 32% lower read latency compared to System E while maintaining comparable area-efficiency. System E has low energy-efficiency because it only supports *NB* = 8 for I/O width of 64 bit. However, for a wider I/O width of 128 bits, a larger *NB* can be supported by a MAS- II system, resulting in lower energy consumption, as will be discussed in the next subsection.

Based on our analysis, we propose to achieve high energy-efficiency and good reliability by choosing *NB*, *NL* and *NS* appropriately. Recall that for higher energy-efficiency, larger *NB* (most effective) or larger *NL* (next most effective) is preferable, while for better reliability, lower *NB* (most effective) or lower *NL* (next most effective) is preferable. In order to achieve both high energy-efficiency and reliability, we choose *NB* to be 8 or 16 and set *NL* to be as large as possible (according to MAS-I or MAS-II), so that *NS* would be as small as possible. System C is a perfect example with $NB = 16$, $NL = 4$ and $NS = 1$.

TABLE 4.6.
COMPARISONS OF AREA FOOTPRINT, ENERGY CONSUMPTION AND LATENCY OF DIFFERENT SYSTEMS OF 1GB WITH 3% OF SWITCHING VARIATIONS

| **Systems** | **BER** | ***t*** | **Area Footprint ($mm^2$)** | **R (W) Energy (pJ)** | **R (W) Delay (ns)** |
|---|---|---|---|---|---|
| **Baseline** | $9.7\times10^{-4}$ | 10 | 3.12 | 224.7 (353.6) | 84.1 (126.2) |
| **A** | $6.1\times10^{-5}$ | 5 | 2.35 | 180.1 (287.2) | 80.6 (122.2) |
| **B** | $1.7\times10^{-4}$ | 6 | 2.59 | 154.0 (258.8) | 81.6 (122.4) |
| **C** | $4.1\times10^{-4}$ | 7 | 2.62 | 133.1 (225.3) | 82.1 (123.8) |
| **E** | $1.3\times10^{-4}$ | 6 | 2.58 | 161.7 (266.7) | 120.2 (176.8) |

Now if there is an additional switching voltage variation in the SET/RESET threshold, the BER increases. Specifically, with 3% switching voltage variation, the BER doubles compared to the system without any variations. Table 4.6 shows comparisons of area, energy consumption and latency when the switching voltage variation is 3%. We see that a stronger ECC has to be used to guarantee the same lifetime of 10 years. For instance, System C now uses stronger BCH with $t = 7$ instead of $t = 6$ code. However, System C still has the lowest energy consumption with negligible performance loss and area overhead.

### 4.6.3 Scalability of 3DHRAM

1) Doubling I/O width (128 bits)

We investigated the performance of 1GB 3D-HRAM systems for the case when the I/O width is doubled from 64 bits to 128 bits. All systems use BCH code to achieve BFR of $10^{-10}$. Table 4.7 compares the area, read/write energy, and latency of four systems, A', C', E' and F. Compared to the systems with I/O width of 64 bits, the proposed systems have to access either more subarrays (such as System A', C' and E') or more layer (such as System F) to match the higher I/O width (128 bits). However, total energy as well as the read/write delay decreases because the total number of beats to access 512 bits reduces from 8 to 4. For example, System A' saves energy about 39% and reduces delay by about 47% in average compared with System A. System A', C' and E' have the same BER as System A, C and E, respectively since BER only depends on *NB* and *NL* and not on *NS*. Table VII also shows that System F which uses MAS-II has the lowest read/write energy consumption but at the price of read performance degradation compared to other systems.

TABLE 4.7.
COMPARISONS OF AREA FOOTPRINT, ENERGY CONSUMPTION AND LATENCY OF DIFFERENT 1GB SYSTEMS WITH I/O WIDTH OF 128 BITS

| | ***NB*** | **NM; NL; NS** | **BER** | **Required BCH [57]** | **Area Footprint ($mm^2$)** | **R (W) Energy (pJ)** | **R (W) Delay (ns)** |
|---|---|---|---|---|---|---|---|
| **MAS-I** | **8** | **1; 4; 4 (A')** | $3.6\times10^{-5}$ | $t = 4$ | 2.31 | 97.3 (151.9) | 47.9 (60.2) |
| | **16** | **1; 4; 2 (C')** | $2.3\times10^{-4}$ | $t = 6$ | 2.58 | 71.7 (123.5) | 49.3 (60.7) |
| **MAS-II** | **8** | **1; 8; 2 (E')** | $7.2\times10^{-5}$ | $t = 5$ | 2.54 | 79.8 (123.9) | 75.5 (91.7) |
| | **16** | **1; 8; 1 (F)** | $3.8\times10^{-4}$ | $t = 7$ | 2.62 | 59.7 (101.5) | 76.8 (93.5) |

2) Larger Subarray Size

Table 4.8 compares the area, read/write energy, and latency for the 1GB 3D-HRAM systems with different subarray sizes and different I/O widths when MAS-I is used. Compared

to the systems described earlier with subarray size of 512 × 512 × 16, the systems in Table VII have larger subarray size through use of more layers (Systems G, H have 32 layers per subarray) or more bits in each layer (Systems G', H' have 1024 × 1024 bits per layer). The system with larger subarray size has smaller total area footprint due to fewer number of mats. For example, System G with larger subarray size has 128 mats and hence smaller area footprint compared to System C with 256 mats.

All configurations incur more IR drop due to longer interconnection. To avoid read/write failures, we restrict $NL \leq 8$ and $NB = 8$ for the larger subarray size case. (We found that write/read failures happen when $NB = 16$ under larger subarray size, resulting in BER > 0.5.) Even then BERs are higher than the systems with smaller subarray size, resulting in higher ECC storage area overhead. Among all proposed systems in Table VIII, the system with larger subarray size (System G', H') has lower routing delay, resulting in lower R/W latency and corresponding R/W energy. For example, System G' lowers R/W latency by 5% and R/W energy by 20% compared with System G. However, System G' has larger subarray size and higher ECC parity storage area, incurring 14% higher area overhead compared with System G.

TABLE 4.8.
COMPARISON OF AREA FOOTPRINT ENERGY AND LATENCY FOR 1GB MEMORY WITH DIFFERENT ARRAY SIZE BY USING MAS-I

| **Memory Cell Subarray** | **I/O width (bits)** | ***NB*** | ***NM; NS; NL*** | **BER (BCH)** | **Area Footprint ($mm^2$)** | **R (W) Energy (pJ)** | **R (W) Latency (ns)** |
|---|---|---|---|---|---|---|---|
| **512×512 ×32** | 64 | 8 | 1; 1; 8 (G) | $3.7\times10^{-4}$ ($t = 7$) | 1.39 | 110.9 (176.5) | 86.2 (130.2) |
| | 128 | | 1; 2; 8 (H) | | | 60.4 (93.8) | 52.8 (71.7) |
| **1024×1024 ×16** | 64 | | 1; 2; 4 (G') | $7.9\times10^{-4}$ ($t = 9$) | 1.61 | 89.6 (139.1) | 81.7 (123.8) |
| | 128 | | 1; 4; 4 (H') | | | 51.5 (7 7.3) | 49.3 (62.0) |

## 4.7 Conclusion

In this chapter, we propose multi-layer access schemes with new data organization to improve energy-efficiency and reliability of 3D-HRAM systems. We present two low cost Multi-layer Access Schemes (MAS-I and MAS-II) that differ in the number of layers that are activated and thus differ in energy efficiency. In order to improve reliability, we propose to use a combination of bit-interleaving in a group along with rotated access across subarrays and across layers in a subarray. Such a scheme ensures that the error characteristics of all data lines are the same, resulting in low average BER. Our analysis shows that 3D-HRAM systems improve energy-efficiency by choosing larger *NB* (most effective) or larger *NL* (next most effective). Since different memory systems (corresponding to different values of *NB, NL, NS, NM*) have different bit error rates, we use BCH codes with appropriate strength so that all systems achieve the same reliability (BFR = $10^{-10}$). Simulation results using NVSim show that for a 1GB 3D-HRAM 1S1R ReRAM system, when I/O width is 64 bits, the *NB* = 16, *NL* = 4 MAS-I system with BCH $t = 6$ has the lowest energy consumption.

# CHAPTER 5

# AN RERAM-BASED NEURAL NETWORK ACCELERATOR THAT MAXIMIZES DATA REUSE AND AREA UTILIZATION

## 5.1 Introduction

Convolutional neural networks (CNN) are increasingly being applied in computer vision, natural language processing and robotics [28-31]. These networks achieve very high accuracy at the cost of large computational cost [34]. For example, AlexNet [29], ResNet [81], VGG-16 [33] require 724M, 3.9G and 15.5G multiply-and-accumulate operations to process one ImageNet image, respectively.

The bottleneck of these architectures is the number of memory accesses, leading the way to process-in-memory (PIM) architectures [34]. Compared with CMOS-based PIM architectures [82-84], emerging non-volatile memory (eNVM) technologies, such as phase change memory [38] and resistive random access memory (ReRAM) [35-37], are more promising candidates due to their compatibility with the CMOS back-end-of-line process. In an ReRAM based architecture, the MAC operations used in filter computations are typically obtained through analog computation in the crossbar array. The accelerator designs in [35, 36] use the conventional crossbar architecture where writing the weights into the eNVM cells is a non-trivial task due to the sneak paths. In this section, we focus on the 1-transistor-1-resistor (1T1R) structure which does not have the problem of weight loading.

We propose a multi-tile ReRAM-based CNN accelerator, MAX$^2$, for AlexNet, ResNet and VGGNet-based networks, where each tile consists of 3×3 processing elements (PE) corresponding to a receptive field (or filter) size of 3×3. By implementing larger receptive field filter with a stack of 3×3 filters, all tiles in our design have the same structure since they all are optimized for a 3×3 filter implementation. It improves upon intra-layer processing by

maximizing IFM data reuse, minimizing interconnection cost and reducing number of data transactions. MAX$^2$ employs weight duplication as in [36] but uses it with appropriate choice of data granularity so that the cost of additional peripheral circuitry is minimized.

The rest of this chapter is organized as follows. In Section 5.2, we provide a brief background of CNN and ReRAM-based computation. In Section 5.3, we present the MAX$^2$ multi-tile architecture, where each tile consists of systolic arrays built with ReRAM array based PEs. In Section 5.4, we show how multiple architectural approaches are used to improve MAX$^2$ efficiency with respect to area, performance and energy. This is followed by a system-level evaluation of MAX$^2$ and comparisons with related work in Section 5.5. Section 5.6 shows the implementation details for AlexNet and Section 5.7 shows the implementation details for ResNet. This is followed by extensions of MAX$^2$ to support multi-bit weight and multi-bit activations in Section 5.8. We summarize the related work in Section 5.9 before concluding the chapter in Section 5.10. We present details of the mapping of weights in different tiles for the three networks in the Appendix.

## 5.2 Background

### 5.2.1 Convolutional Neural Network (CNN) Basics

Recent CNN models, such as AlexNet [29], ResNet [81] and VGGNet [33], consist of multiple convolution (CONV) layers to learn the important features, followed by a small number (e.g., 1 to 3) of fully-connected (FC) layers for classification. In a CONV layer, an output feature map (OFM) is the result of multiply-and-accumulation (MAC) operations on a collection of weights (or filters) operating in a sliding window fashion over the input feature map (IFM). The convolution operation in the CONV layers is composed of high-dimensional convolutions, such as 3D convolution with C channels, as shown in Fig. 5.1. Consider the case

where the IFM of size $H \times H \times C$ is processed by M filters, each of size $R \times R \times C$. Then the OFM of size $M \times E \times E$, where $E = H - R + 1$ (given the stride of 1), is computed as follows. Here $\mathbf{I}, \mathbf{W}, \mathbf{O}$ are the IFM, weights and OFM, respectively.

$$\mathbf{O}[m][x][y] = \sum_{i=0}^{2} \sum_{j=0}^{2} \sum_{k=0}^{C-1} \mathbf{I}[k][x+i][y+j] \times \mathbf{W}[m][k][i][j]$$

$$0 \leq i, j \leq 2, \ 0 \leq m < \mathrm{M} \text{ and } 0 \leq x, y < E.$$

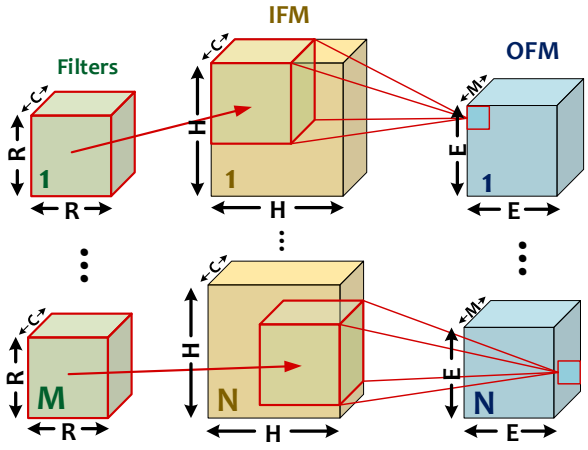



Figure 5.1: 3D convolution in CNN, adopted from [32].

In this chapter, we focus on the acceleration of the inference engine where the weights have been pre-trained offline. We consider three popular CNNs, namely, AlexNet, ResNet and VGGNet. While the number of layers in these networks are different, most of the layers have a receptive field (or filter) of size 3×3, a feature which we exploit in our architecture. We focus on efficient computation of the CONV layers since they account for more than 90% of the computation. We also focus on 1-bit weight precision and 1-bit activation precision since binary neural networks have been shown to be an efficient design point considering the

tradeoff between accuracy and hardware resources [87, 90-91]. A high precision weight can also be supported by employing multiple ReRAM cells, as will be discussed in Section 5.8.

### 5.2.2 ReRAM-based Processing-in-Memory (PIM) Basics

In the past few years, several ReRAM based CNN accelerators have been proposed [35-37]. The most compact ReRAM based synaptic array structure is the crossbar structure. Unfortunately, it suffers from write disturbance and sneak path issues. The two-terminal selector device used to mitigate these problems is still a premature technology. So in this chapter, we consider the 1T1R structure, where the bit-lines (BLs) and the source-lines (SLs) are perpendicular to form a "pseudo-crossbar" [85]. To perform dot product operation, the weight matrix is stored as conductance in the 1T1R array.

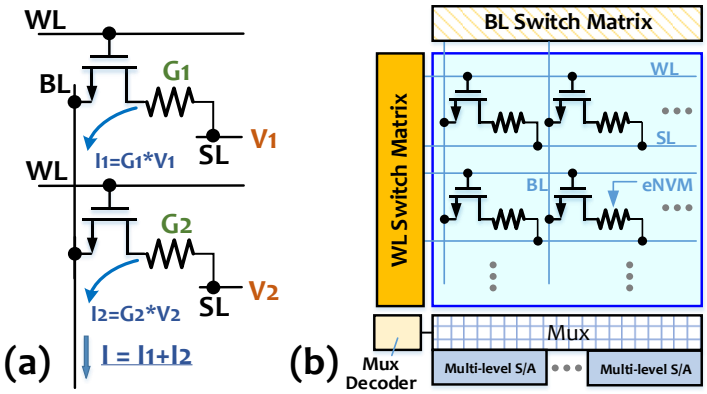



Figure 5.2: "Pseudo-crossbar" array is a modified 1T1R array [85]. (a) Illustration of multiply-and-accumulate (MAC) operation being performed along a bit line. (b) The circuit diagram of a "pseudo-crossbar" array that enables dot product computations.

For binary networks where IFM is 1 bit, the input 'vector' is translated to voltages that are applied to the WLs; small voltages (< 0.5V) are applied to BLs so that the dot product of the input vector and weight vector now corresponds to the analog summed current along the SLs (Fig. 5.2(a)). To digitize the analog current generated in the SLs, a multi-level sense amplifier (SA) is used, as shown in Fig. 5.2(b). It is impractical to implement a SA for each SL since the

SA layout area is much larger than a single 1T1R cell. So in our design, 8 SLs share 1 SA. When array size is 128×128, 16 results are generated simultaneously. Each result has a precision of 4 bits, where the precision is chosen based on both subarray size and the distribution of partial sum values as in [87].

## 5.3 MAX$^2$ Architecture

In this section, we describe our proposed CNN accelerator, MAX$^2$, that (1) reduces heavy data movement by maximizing IFM reuse between PEs, (2) minimizes interconnection cost by implementing a systolic architecture, and (3) reduces intra-PE bandwidth by using LUT sharing. While the details are given for VGG-19, the same architecture skeleton is used for AlexNet (details in Section 5.6) and ResNet (details in Section 5.7).

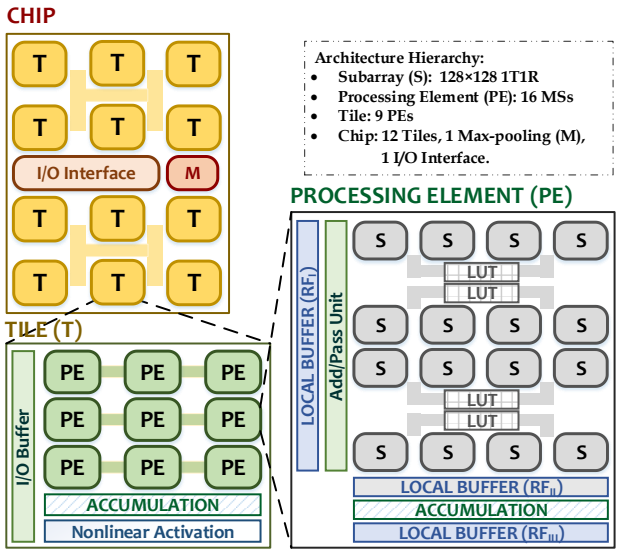



Figure 5.3: MAX$^2$ hierarchical architecture for VGG-19.

### 5.3.1 System Overview

The MAX$^2$ architecture consists of multiple tiles, where each tile consists of multiple processing elements (PE) that operate in the systolic mode. Each PE consists of multiple

ReRAM subarrays that are configured to perform the dot product operation. All weights are stored on chip and so the weights have to be loaded only once. Figure 5.3 shows the hierarchical system architecture of $MAX^2$ customized for VGG-Net. While $MAX^2$ currently supports only inference (forward propagation), it can be extended to support backward propagation and weight update with additional circuitry.

- **Processing Element (PE):** Each PE consists of 16 ReRAM based memory subarrays, where each subarray is a 1-transistor-1-resistor (1T1R) array of size 128×128. Each subarray has its own set of peripheral circuitry. A set of 4 subarrays share 1 look up table (LUT). Each PE also has three local buffers, $RF_I$, $RF_{II}$ and $RF_{III}$, to store IFM, dot product and partial sums, respectively.
- **Tile:** Each tile consists of 3×3 PEs, an I/O buffer to store input/output data, a non-linear activation function unit and an accumulation unit for partial sum addition.
- **Chip:** Each chip consists of 12 tiles, and a DFF-based I/O interface to store multiple input feature maps and one max-pooling unit.

In $MAX^2$, each tile stores weights of one or more layers, depending on the size of the filter. So if the VGG network has 16 layers, then the naïve implementation needs 16 tiles. However, $MAX^2$ has 12 tiles since some tiles store weights of two layers. Each tile has 3×3 PEs to support a 3×3 filter. We use the same tile design for all layers since the receptive field size is 3×3 for all layers in the VGG network.

The $MAX^2$ chip reads the IFM data via a DFF-based I/O interface. This is sent to the input buffer of the first tile and processed by a systolic array of PEs. In each PE, the IFM stored in local buffer $RF_I$ feeds into four subarrays located in one column. Once the first tile is done with processing, the output feature map is sent to the second tile. This process

continues until the final layer generates an output that is sent to the I/O interface. We use inter-layer pipelining to speed up the computation, as in ISAAC [35]. Basically, as soon as enough number of outputs are generated by a tile and aggregated in the I/O buffer, the next tile can start its operations.

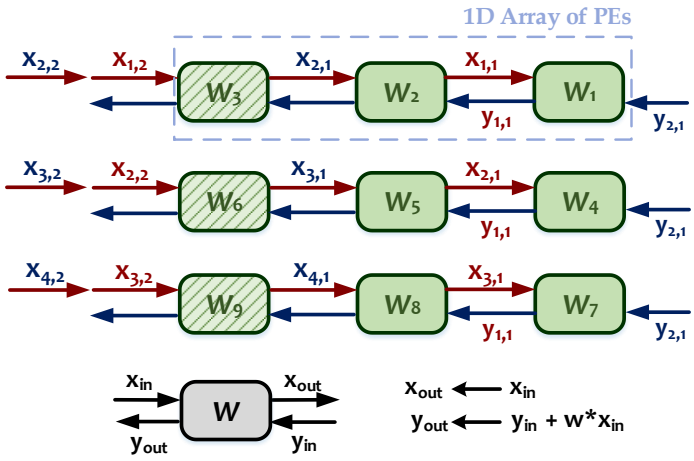



Figure 5.4. Systolic array based design.

5.3.2 Systolic Architecture Design

Each tile consists of a systolic architecture of PEs that rhythmically compute and pass data through the system [86]. Our design consists of three linear systolic arrays with bidirectional inter-connection. These arrays work in parallel and the PS outputs computed by the three arrays are added to generate the OFM outputs. Figure 5.4 shows the block diagram of the systolic array architecture. Here weights stay, while the inputs and outputs move systolically in opposite directions [86]. If each input element is fed in odd cycles (i.e. data with all-zero are fed in even cycles), and the delay between two cells is 1 unit in both the forward and reverse directions, then the outputs are generated every 2 units. Since only approximately one-half of the cells work at any given time, two independent convolution computations can be interleaved. Thus, instead of sending data with all-zeros, two sets of IFM inputs can be interleaved and sent to the systolic array every cycle and an output can be obtained every cycle.

Figure 4 shows how data from rows 1, 2, 3 are interleaved with data from rows 2, 3, 4 to generate two sets of partial sum products.

5.3.3 PE Design

**Mapping Method:** Recall that there are 9 PEs in a tile and each PE consists of multiple ReRAM-based subarrays that do all computations corresponding to position $(i, j)$ of a 3×3 filter, where $0 \leq i, j < 3$. Each OFM output is the sum of 9 dot products with each PE contributing to one dot product. The sum of the 3 dot products along a row (referred to as partial sum) is generated by the linear systolic array, and the 3 partial sums from 3 systolic arrays are added by an adder tree to generate the OFM output. Figure 5.5 illustrates the proposed scheme of mapping IFM, and weights to generate a volume of dot products in a PE. Here the IFM is of size 30×30×512, (i.e., 512 channels with 30×30 per channel).

Consider the case where the ReRAM logical array is of size 512×512. Instead of mapping all elements from one filter with size 3×3×512 into a column of ReRAM array like ISAAC [35] and PipeLayer [36], we map the elements in the same location across 512 channels from one filter into a column of the ReRAM array. Thus, 512 columns of ReRAM logical array are loaded with weights from $M = 512$ filters. These groups are marked as cuboids (light green to dark green) in Fig. 5.5. The IFM data across $C = 512$ channels are fed into 512 WLs of ReRAM array. Each column computes the dot product of the IFM data with the weights of one filter over 512 channels. Since there are $M = 512$ columns, the IFM data is used to generate 512 dot product results, marked by blue cuboids in Figure 5. The 30 IFM vectors of size 512 along a row (marked as cuboids from dark yellow to light yellow) are fed one after the other to generate 30 sets of 512 dot products along a row.

In the physical design, the 512×512 matrix is decomposed into a group of 4×4 matrices, where each matrix is mapped to a 128×128 1T1R array. We set 128×128 to be the largest possible ReRAM subarray size since subarrays larger than this have been shown to suffer from IR drop and sneak path [20]. The 4×4 memory subarrays in one PE store the weights in one location across 512 channels from 512 filters, and 3×3 PEs in a tile store all weights (nine locations across 512 channels) from 512 filters. If the matrix size is smaller (such as 256×256, 128×128 and 64×64), we store the same weights in the subarrays along a column. We also load weights of different layers along row subarrays to speed-up the inference and also improve area utilization. These features will be discussed in Section 5.4.

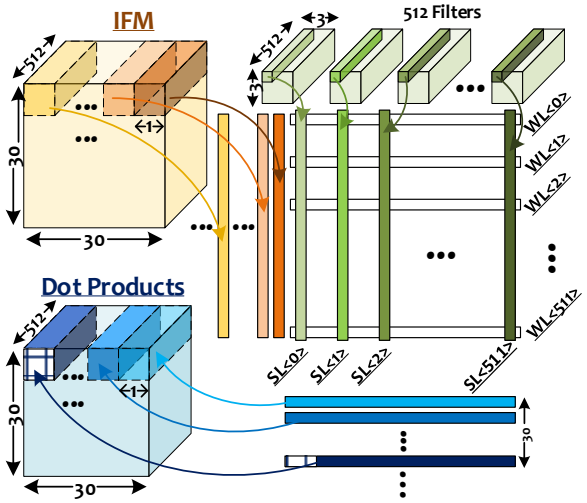


Figure 5.5: Novel mapping of IFM and weights in a PE to generate dot products.

**Dataflow:** IFM data from $RF_I$ are fed into a column of 4 subarrays. In each subarray, the dot product results of 128 sized vectors are processed by multilevel sense amplifiers. Since 8 consecutive SLs share one SA, the results from 16 SLs (= 128/8) with 4 bits/SL (total of 8B) are sent to a look-up table (LUT) for quantization. Next, results of 16 SLs × 9 bits/SL = 16B from 4 column subarrays are added by the Add/Pass Unit to generate the dot products,

where each dot product is represented by 11 bits. These results are aggregated in the $RF_{II}$. Accumulation Unit adds the partial sums stored in $RF_{II}$ with the partial sum results from the right PE and stores the updated partial sum in $RF_{III}$. Each partial sum is represented by 13 bits. This process is repeated 512/16 = 32 times to compute all the partial sums in each PE.

**LUT Sharing:** To minimize the quantization error of the partial sums, nonlinear quantization is performed. Nonlinear quantization has been shown to achieve better accuracy than linear quantization for the same number of quantization levels [87]. The LUT that is needed for nonlinear quantization of the dot product [87] has a large area overhead and has to be designed properly. Instead of using a SRAM-based LUT, ReRAM-based LUT is used due to its smaller area overhead. From a practical consideration, we set the LUT budget to be 10% of total PE area. This constraint forces four subarrays along a row direction to share one LUT. A more relaxed LUT budget would have enable more bits from different subarrays in a row to be processed at a time. While this has the potential to speed up the computation in the PE, it comes with area overhead due to higher bus bandwidth and larger size of $RF_{II}$ and $RF_{III}$. So here, we choose to share a LUT among four subarrays. The corresponding intra-PE bandwidth is 40 GB/s, which is much smaller than 320GB/s used in prior work [36].

### 5.3.4 Tile Design

At the tile level, the IFM data are processed by a systolic array of PEs. Three partial sums from three sets of arrays are sent to the Accumulation Unit to compute the total sum, which is then processed by the Activation Function Unit to generate the final output. We use a LUT to implement the activation function based on ReLU for ease of realization.

The I/O buffer in each tile can store 4×16×512 input data and 4×16×512 output data. We store 4 rows since two interleaved rows of partial sums require 4 rows of IFM data to be fed into the 3 linear systolic arrays.

## 5.4 Improving MAX$^2$ Efficiency

In this section, we propose several architectural approaches to improve energy-efficiency, performance and area-efficiency of MAX$^2$. At the PE level, we propose several data Granularity Options (GO) along with Same Weight Duplication (SWD) to speed up the computations in different layers with different matrix sizes without additional area overhead. Different Weight Loading (DWL) is used to improve area-efficiency when the matrix size is small. While in this section, we give details for VGG-19, the same approaches have been applied for AlexNet and ResNet.

### 5.4.1 Same Weight Duplication (SWD)

As mentioned in Section 5.3, each PE consists of 16 1T1Rsubarrays, each of size 128×128 to store a weight matrix of size 512×512. However, area utilization rate (defined as the ratio between the area for weight storage and the subarray area) is quite low when smaller weight matrix is stored in the PE. For example, area utilization rates for weight matrix of 128×128 (layers 3 and 4 in VGG-19) and 256×256 (layers 5 - 8 in VGG-19) are only 6.25% and 25%, respectively. To improve the area utilization rate and also speed up computation, we propose to load same weights to the subarrays along the columns, as shown in Fig. 5.6. This procedure is referred to as Same Weight Duplication or SWD. It enables 4× more outputs to be computed if the matrix size is 128×128, and 2× more outputs if the matrix size is 256×256 at the expense of additional peripheral circuitry. Note that loading the same weights for subarrays along rows is meaningless since 4 subarrays along the horizontal direction share one LUT.

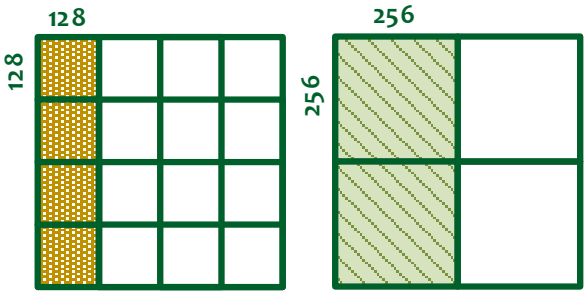


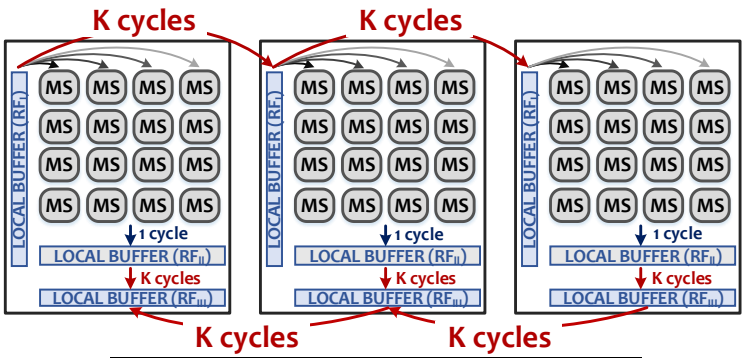

Figure 5.6: Cartoon figure for same weight duplication (SWD). Weight matrix of 128×128 is duplicated three times along the columns in left panel; and weight matrix of 256×256 is duplicated once along the columns in right panel.

Now, if the matrix is of size 128×128, SWD of a factor (S = 4) optimizes performance by 4× but at the price of 4× increase in the size of $RF_{II}$, $RF_{III}$, and 4× increase in bus bandwidth. Thus, large SWD increases the size of the peripheral circuitry significantly. Designing the peripheral circuits in each PE to speed up the computation of layers with the smallest matrix (64×64), results in overdesign for the layers with larger sized matrices, and is not desirable. In the next subsection, we present a solution that does not increase the size of peripheral circuitry.

### 5.4.2 SWD with Different Granularity K

We revisit the systolic array based design described in Section 5.3. The input data is fed into a column of subarrays, the dot product results in 16 SLs are obtained in every cycle and these results are stored in $RF_{II}$. Each cycle T = 0.67ns, corresponds to an operation frequency of 1.5GHz. In each PE, a total of 512/16 = 32 cycles are needed to process one IFM data set, which corresponds to one location in the IFM array across 512 channels. Now instead of processing one IFM data set continuously for 32 cycles, we choose to process one IFM data set for only $K$ cycles ($K \leq 32$) at a time and then start processing a new IFM data set. Thus, $32/K$ rounds are needed to process the IFM dataset and the I/O buffer has to be accessed $32/K$ times. While a smaller $K$ requires more I/O accesses, resulting in more energy

consumption, a larger $K$ requires a larger inter-PE bus bandwidth and larger register files to store the partial sums. So we choose lower $K$ ($K \leq 4$) for the systolic array.




| Granularity $K$ | I/O Access | Bandwidth (GB/s) |
|---|---|---|
| 1 | 32 | 39 |
| 2 | 16 | 78 |
| 4 | 8 | 156 |

Figure 5.7: Dataflow for different values of K. It takes K cycles to transfer data from one PE to the next, where cycle time is T = 0.67ns, corresponding to f = 1.5GHz.

We process 512×1 bit = 64B of IFM data at a time. Thus, as shown in Fig. 5.7, 64B is sent from one PE to its right neighbor via $RF_I$ every $K$ cycles and the partial sum is sent to its left neighbor every $K$ cycles. While the size of $RF_I$ and bus bandwidth in the forward direction does not change with $K$, the size of $RF_{II}$, $RF_{III}$ as well as bus bandwidth in the backward direction change with $K$. Basically, if a larger chunk of partial sum (larger $K$) is transferred every $K$ cycles, the bus bandwidth in the backward direction and local buffers ($RF_{II}$ and $RF_{III}$) have to be increased. For instance, if $K = 4$, the partial sum has to be sent out every 4 cycles. The bandwidth requirement is then 156GB/s and local buffer size is 108B. In contrast, if K = 1, the bandwidth requirement is only 39GB/s and the local buffer size is only 26B.

In order to overcome the overdesign issue due to SWD, we propose to use SWD in conjunction with granularity options (GO). The basic idea of GO + SWD is that for smaller

matrix, the additional peripheral circuit requirements incurred by SWD will be compensated by use of a smaller $K$. Specially, the buffer size and bandwidth are the same as long as the product of data granularity and duplication factor, i.e. $K \times S$, is constant.

Specially, $RF_{II}$ is of size $U \times K \times S$, where U is the number of bits required to store 16 dot products. Since each dot product is represented by 11 bits, $U = 16 \times 11b = 22B$. Similarly, $RF_{III}$ is of size $V \times K \times S$, where $V$ is the number of bits required to store 16 partial sums. Since each partial sum is represented by 13 bits, $V = 16 \times 13b = 26B$.

TABLE 5.1.
SWD+GO CHOICES FOR DIFFERENT MATRIX SIZES

| | **SWD Factor S** | **Effect on Peripheral circuits** | **GO Factor K** | **Effect on Peripheral circuits** |
|---|---|---|---|---|
| 512×512 | 1 | 1× | 4 | 1× |
| 256×256 | 2 | 2× | 2 | 0.5× |
| 128×128 | 4 | 4× | 1 | 0.25× |
| 64×64 | 4 | 4× | 1 | 0.25× |

Table 5.1 shows the SWD and GO choices for different matrix sizes. For a layer with matrix size of 512×512, we use SWD with S = 1 along with $K = 4$. The corresponding bus bandwidth is 160GB/s (> 104B × 1.5GHz) and $RF_{II}$ and $RF_{III}$ are of sizes 88B and 104B, respectively. For layers with smaller matrix of size 256×256, we use $S = 2$ and $K = 2$, while 2× more outputs are generated, the sizes of $RF_{II}$ and $RF_{III}$ are still 88B and 104B, respectively. Thus GO+SWD helps achieve speed-up in layers with smaller matrices without incurring additional area overhead. Note that a layer with matrix size of 64×64 can only employ SWD with $S = 4$ (and not 8) due to LUT budget constraint.

5.4.3. Area Saving – Different Weights Loading (DWL)

To further improve the area-efficiency, we propose to load different weights from different layers along the horizontal subarrays in each PE. For layers with small matrix size, two matrices can be loaded in one PE, as shown in Pattern I and Pattern II in Fig. 5.8. For example, in Pattern II, the matrix of 128×128 and the matrix of 256×256 are loaded in one PE, thereby increasing area utilization rate of the PE from 25% to 75%. All PEs in the same tile employ the same pattern.

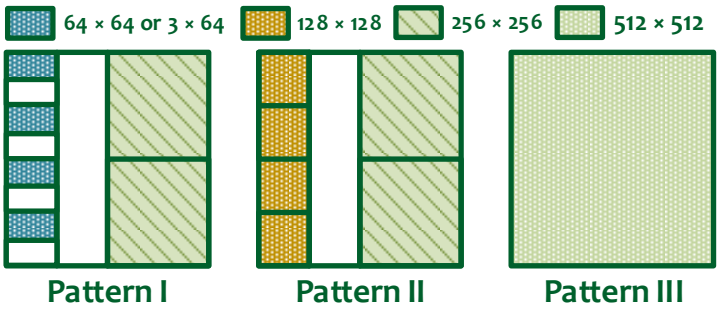



Figure 5.8: PE weight storage patterns using DWL.

Compared with the conventional architecture where one tile only stores the weights from one layer, DWL helps reduce the total number of tiles from 16 to 12 for VGG-19 and achieve high area utilization of 95.7%. The mapping of weights for VGG-19 is shown in the Appendix.

## 5.5 Evaluation on VGG-19

### 5.5.1 System Setup

We use the NeuroSim framework [88], which is an integrated framework for design space exploration of neuro- inspired architectures, to model energy, area and latency of the different components at the PE level, tile level and chip-level in the 32nm node. The 1T1R ReRAM model is based on a ReRAM device compact model [65] calibrated by IMEC's $HfO_2$ ReRAM

[43] in predictive 32nm technology node. At the PE level, the baseline NeuroSim was used to generate numbers for ReRAM subarrays along with supporting circuitry for nonlinear quantization (LUT table), adders and buffers. In order to simulate CONV implementation using systolic array, we implemented buffers in 32nm CMOS to store the IFM, dot products and partial sums in each PE. At the tile level, we added the contribution of adders and I/O buffers implemented in 32nm CMOS. At the chip level, we added the contributions of max-pooling unit built by 6K comparators and a LUT-based ReLU activation function. The area, latency and energy estimations of each of these hardware units was used to calculate the area, latency and energy numbers at the tile level and chip level. We assume a frequency of 1.5 GHz for the digital blocks.

TABLE 5.2.
PE-LEVEL COMPONENTS IN MAX$^2$ FOR VGG-19

| **PE Level at *f* = 1.5GHz** | | | | |
|---|---|---|---|---|
| **Component** | **Numbers of units** | **Size** | **Energy (pJ)** | **Area ($\mu m^2$)** |
| Subarray | 16 | 2 KB | 2.823 | $2.9\times10^3$ per subarray |
| LUT | 4 | -- | 0.056 | $9.6\times10^2$ per LUT |
| Add/Pass | 1 | -- | 0.420 | $1.6\times10^3$ |
| Accumulation Unit | 1 | -- | 2.036 | $1.8\times10^3$ |
| Activation Unit | 1 | | 0.017 | $3.4\times10^2$ |
| $RF_I$ | DFF-based | 64 B | 2.066 | $0.8\times10^3$ |
| $RF_{II}$ | | 88 B | 2.832 | $1.0\times10^3$ |
| $RF_{III}$ | | 104 B | 3.351 | $1.2\times10^3$ |
| **One PE** | | | | $5.7\times10^4$ |

**Subarray Breakdown:** For VGG-19, the energy and area breakdown for each subarray is as follows: memory array energy 0.18pJ and area 536$\mu m^2$, switch matrices energy 0.27pJ and

area 577μm$^2$, Mux decoder energy 0.13pJ and area 445μm$^2$ and multi-level SA energy 11.1pJ and area 1333μm$^2$.

**PE-level Results:** Table 5.2 shows the energy and area breakdown of each component at the PE level. Energy numbers are based on one-time access for the configuration with $K = 4$ and $S = 1$. From the table, we can see that LUT takes up small amount of area and energy, as per design. The register files ($RF_{II}$ and $RF_{III}$) account for 5% of PE area, but consume significant energy. The inter-PE bandwidth is chosen to be 160GB/s, as mentioned in Section 5.4.

**Tile-level & Chip-level Results:** The tile-level and chip-level results are summarized in Table 5.3. In addition to the 9 PEs, each tile also contains a 4 KB input buffer, which account for 10% total area. At the chip level, MAX$^2$ has 12 tiles and 12 Kb I/O interface. The Max-pooling unit has an area of 0.035 mm$^2$. To process one image in CIFAR-100, the energy consumption is 49.6 μJ and the latency is 21.7 μs.

TABLE 5.3.
TILE- AND CHIP-LEVEL COMPONENTS IN MAX2 FOR VGG-19

| **Tile Level** | | | | |
|---|---|---|---|---|
| **Component** | **Numbers of units** | **Size** | **Energy (pJ)** | **Area (mm$^2$)** |
| PE | 9 | | 131 | 0.057 per PE |
| I/O Buffer | DFF-based | 8 KB | 264.4 | 0.1 |
| Accumulation Unit | 1 | -- | 1.156 | 0.01 |
| Activation Function Unit | 1 | -- | 0.017 | $3.4\times10^{-4}$ |
| **One Tile** | | | | 0.607 |
| **Chip Level** | | | | |
| Tiles | 12 | -- | 1823 | 0.607 per tile |
| Max-pooling Unit | 1 | -- | 13.56 | 0.035 |
| I/O interface | DFF-based | 12 Kb | 12.25 | 0.02 |
| **Total Chip** | -- | -- | | 7.58 |

**Evaluation on Different VGG Networks:** Among all VGG networks, VGG-11 with 8 CONV layers is the shallowest network while VGG-19 with 16 CONV layers is the deepest network. The computing energy for VGG-11 is 28.4 μJ and latency is 15.1 μs compared to 49.6 μJ and 21.7 μs for VGG-19. The areas for all VGGs are the same since we use 12 tiles for all VGG networks.

### 5.5.2 Results

1) Effect of Different Strategies

In order to evaluate gains of using the different strategies on the performance, we compare the timing-efficiency, energy-efficiency and area-efficiency of different versions of the system:

V1: Systolic array with data interleaving; $K = 1$ for all layers.

V2: V1 along with inter-layer pipeline.

V3: V2 along with SWD + proper $K$ for different layers.

MAX$^2$: V3 along with DWL.

All results in Fig. 5.9 are normalized to the V1 system. All systems have the same off-chip memory access energy but differ in the computation energy. Specifically, the off-chip memory access energy is due to energy to access off-chip DRAM to load IFM, and energy to load weights into the chip. As NeuroSim does not support off-chip memory simulation, we obtain DRAM read and write energy consumption (7 pJ/bit) from [28]. For VGG-19, off-chip DRAM access energy is 0.02 μJ for loading one image, and energy for loading weights on chip is 51.1 μJ.

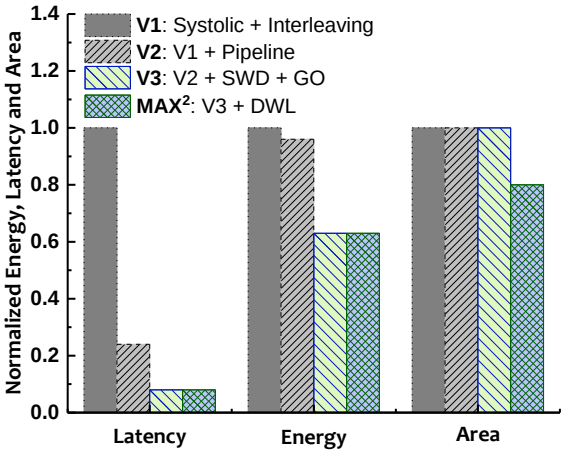


Figure 5.9. Latency, energy and area of VGG-19 normalized to that of the V1 system that employs systolic array processing.

From the figure, we see that V2 system speeds up computation by 4× compared to the V1 system by using inter-layer pipeline. Both V1 and V2 have comparable energy efficiency because even though inter-layer pipeline significantly reduces latency in V2, the power due to pipeline increases. V3 system, which also utilizes SWD along with GO, speeds up computation and reduces latency even further. MAX$^2$ improves area-efficiency by 20% compared to V1, V2 and V3 system because of use of DWL. This is because DWL enables one tile to store weights from two layers, resulting in smaller area overhead and energy overhead. Overall, MAX$^2$ has 37% lower energy consumption, 12× higher performance and 20% lower area overhead compared to V1.

2) Comparison with Related Work

**Metrics**: To compare the performance of MAX$^2$ with related work [12, 13], we use two key metrics: (1) Computational Efficiency (CE), which is represented by the number of 1-bit operations performed per second per mm$^2$ (TOPs/s/mm$^2$); and (2) Energy Efficiency (EE), which is represented by the number of 1-bit operations performed per joule (TOPs/s/W).

TABLE 5.4.
COMPARISONS OF CE, EE AND TOTAL AREA BETWEEN RELATED WORKS AND MAX$^2$ FOR VGG-19

| | **CE** *TOPs/(s × mm$^2$)* | **EE** *TOPs/s/W* | **Total Area** *(mm$^2$)* |
|---|---|---|---|
| **ISAAC** | 3.27 | 5.15 | 85.4 |
| **PipeLayer** | *5.94* | *0.57* | 82.6 |
| **MAX$^2$** | 8.08 | 26.8 | 7.58 |

* The CE and EE are scaled up by 8× in ISAAC and by 4× in Pipelayer. Original data for ISAAC is CE of 0.41 TOPs/(s × mm$^2$) and EE of 0.64 TOPs/s/W, while for PipeLayer is CE of 1.485 TOPs/(s × mm$^2$) and EE of 0.142 TOPs/s/W.

Table 5.4 compares peak CE, EE and total area for ISAAC [35], PipeLayer [36] and MAX$^2$. In order to make a fair comparison, the numbers are scaled based on use of multi-level cell (MLC) and weight precision. Both ISAAC and PipeLayer calculate TOP based on number of 16-bit operations. So far, MAX$^2$ only considers a single-level-cell (SLC) ReRAM and 1- bit operation so here TOP is based on the number of 1-bit operations. A system with 1-bit precision will automatically have 8× higher CE and EE than one with MLC of 2 (as [12]), so in the results presented in Table IV, the CE and EE in [35] have been scaled up by 8×. Similarly, since a system with 1-bit precision will automatically have 4× higher CE and EE than one with MLC of 4 (as [13]), the CE and EE in [36] have been scaled up by 4×. Table IV shows that MAX$^2$ increases CE by 2.5× (= 8.08/3.23), increases EE by 5.2× (=26.8/5.15) than ISAAC [35]. Compared to PipeLayer [36], MAX$^2$ increases CE by 1.4× and EE by 47×.

## 5.6 Evaluation on AlexNet

AlexNet [29] is a popular CNN that is widely used for image classification. Conventional AlexNet has five CONV layers and 3 FC layers and utilizes filters of sizes, 3×3, 5×5, 7×7 and 11×11 filters. In order to process the input feature map from the CIFAR-100 dataset (32×32×3), we use stride of 1 for all the layers and replace the receptive field of 11×11 with 7×7 in the first CONV layer. Furthermore, to design a multi-tiled architecture where each tile

consists of 9 PEs, we implement a version of AlexNet where a receptive field of size 5×5 is replaced by a stack of two 3×3 filters (in two CONV layers) and a receptive field of size 7×7 is obtained by stacking three 3×3 filters (in three CONV layers). This design is motivated by the work in [33] which shows that large receptive field can be realized by stacking multiple small filters.

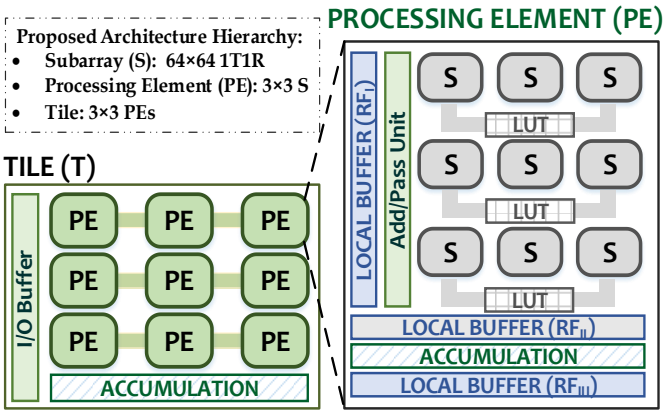



Figure 5.10: Proposed hierarchical architecture for AlexNet.

Figure 5.10 shows the proposed hierarchical system architecture for AlexNet. It has 8 CONV layers with receptive field of size 3×3, and 3 FC layers, mapped into 10-tile architecture. There are 9 PEs per tile, where each PE consists of 9 ReRAM based memory subarrays, each of size 64×64. We choose 64×64 subarray instead of larger sized subarray size so that a matrix of size 192×192 can be easily housed. A set of 3 subarrays share 1 look up table (LUT). Each PE also has three local buffers, $RF_I$, $RF_{II}$ and $RF_{III}$, to store IFM, dot product and partial sums, respectively. We use $K = 1$ for all the layers with small matrices of size 48×128 and use $K = 3$ for all layers with large matrices of size 128×192. Off-chip DRAM access energy is 0.02 μJ for loading one image, and 8.1 μJ for loading weights on chip.

**PE-, Tile- & Chip -level Results:** Table 5.5 shows the energy and area breakdown of each component at the PE level, tile level and chip level. Energy numbers correspond to the

case when $K = 3$ and $S = 1$. From the table, we can see that memory area takes up 66% of PE area while the local RF and supporting circuitry accounts for most of PE energy. The intra-PE bandwidth is chosen as to 45GB/s while the inter-PE bandwidth is 54GB/s. At the tile level, the PEs take up 85% of the area. At the system-level, there is also a Max-pooling unit built with 6K comparators and supporting logic, and a LUT based NL-A unit. To process one image in CIFAR-100, the energy consumption is 15.7 μJ and the latency is 19.8 μs.

TABLE 5.5
PE-, TILE-AND SYSTEM-LEVEL COMPONENTS FOR ALEXNET

| **PE Level at $f$ = 1.5GHz** | | | | |
|---|---|---|---|---|
| **Component** | **Numbers of units** | **Size** | **Energy (pJ)** | **Area ($\mu m^2$)** |
| Subarray | 9 | 4 Kb | 0.0784 | 1154.2 per subarray |
| LUT | 3 | -- | 0.0501 | 455.00 per LUT |
| Add/Pass | 1 | -- | 0.3276 | 687.34 |
| Accumulation Unit | 1 | -- | 0.9260 | 1814.4 |
| $RF_I$ | DFF-based | 24 B | 0.2617 | 313.72 |
| $RF_{II}$ | | 30 B | 0.4177 | 411.46 |
| $RF_{III}$ | | 36 B | 0.5012 | 457.18 |
| **One PE** | -- | -- | -- | $1.58\times10^4$ |
| **Tile Level** | | | | |
| PE | 9 | | 43.2 | $1.58\times10^4$ per PE |
| Input Buffer | DFF-based | 12 Kb | 12.251 | $1.9\times10^4$ |
| Accumulation Unit | 1 | -- | 0.5311 | 6309.0 |
| **One Tile** | -- | -- | -- | $1.68\times10^5$ |
| **Chip Level** | | | | |
| Tiles | 10 | -- | 219.4 | $1.68\times10^5$ per tile |
| Max-pooling | 1 | -- | 13.56 | $3.5\times10^4$ |
| I/O interface | | 12 Kb | 12.25 | $1.9\times10^4$ |
| NL-A | | | 0.017 | 1014.4 |
| **Total System** | -- | -- | -- | $1.79\times10^6$ |

**Comparison with related work:** We compare peak CE, EE, and total area of ISAAC [35], PipeLayer [36] and our proposed system for AlexNet, as shown in Table VI. Compared to [35], our proposed work increases CE by 2.9× (= 11.2/3.27) and EE by 8.2× (= 25.2/5.15). Compared to [36], our proposed work increases CE by 1.9× and EE by 44×.

TABLE 5.6.
COMPARISONS WITH RELATED WORK FOR ALEXNET

| | **CE** *TOPs/(s × mm²)* | **EE** *TOPs/s/W* |
|---|---|---|
| **ISAAC [12]** | 3.27 | 5.15 |
| **PipeLayer [13]** | *5.94* | *0.57* |
| **MAX²** | **11.2** | **25.2** |

* The CE and EE are scaled up by 8× in ISAAC and scaled up by 4× in Pipelayer. Original data for ISAAC is CE of 0.41 TOPs/(s × mm$^2$), EE of 0.64 TOPs/s/W and for PipeLayer is CE of 1.485 TOPs/(s × mm$^2$), EE of 0.142 TOPs/s/W.

## 5.7 Evaluation on ResNet

ResNet [81] is another popular CNN that has shown to achieve higher accuracy than VGGNet. It has 33 CONV layers and 1 FC layer and utilizes filters of sizes 3×3 for the whole network except the first layer which uses a filter size of 7×7. In order to process the input feature map from the CIFAR-100 dataset (32×32×3), we use stride of 1 and replace the receptive field of 7×7 with 5×5 in the first CONV layer. The receptive field of size 5×5 is implemented by a stack of two 3×3 filters. MAX2 design for ResNet is a 16-tile architecture, which utilizes the same PE design as VGGNet. In order to support the shortcut connections in ResNet, four additional DFF-based buffers each of size 4KB have to be used to store the outputs from previous layers. Also, additional accumulation units are needed for adding 16 groups of OFM data. Each buffer can store 4×16×512 output data, corresponding to 4 rows, 16 columns and 512 channels. PE-level results are the same as VGG-Net and shown in Table 5.2.

**Tile- & Chip -level Results:** Table 5.7 shows the energy and area breakdown of each component at the tile level and chip level. At the tile level, the PEs take up 83% of the area. At the system-level, MAX$^2$ has 16 tiles, 12 Kb I/O interface and 4 buffers. To process one image in CIFAR-100, the energy computation is 21.5 μJ and the latency is 16.3 μs.

TABLE 5.7.
TILE- AND CHIP-LEVEL COMPONENTS IN MAX$^2$ FOR RESNET

| **Tile Level** | | | | |
|---|---|---|---|---|
| **Component** | **Numbers of units** | **Size** | **Energy (pJ)** | **Area ($mm^2$)** |
| PE | 9 | | 131 | 0.057 per PE |
| I/O Buffer | DFF-based | 8 KB | 264.4 | 0.1 |
| Accumulation Unit | 2 | -- | 1.156 | 0.01 per unit |
| Activation Function Unit | 1 | -- | 0.017 | $3.4\times10^{-4}$ |
| **One Tile** | | | | 0.617 |
| **Chip Level** | | | | |
| Tiles | 16 | -- | 1823 | 0.617 per tile |
| Max-pooling Unit | 1 | -- | 13.56 | 0.035 |
| Buffer | 4 | 4 KB | 132.2 | 0.05 per buffer |
| I/O interface | DFF-based | 12 Kb | 12.25 | 0.02 |
| **Total Chip** | -- | -- | | 10.13 |

## 5.8 MAX$^2$ Extensions

### 5.8.1 Larger Matrix Size

In MAX$^2$, the largest weight matrix is 512×512. If MAX$^2$ is to be used in CNNs where the matrix size is larger, say 1024×1024, then we could introduce 4 tiles each with storage capacity of 512×512 and load the weights into 4 tiles. Such an implementation would require additional adder trees and buffers, resulting in extra area overhead. We could also fold the 1024×1024 array computation into the 512×512 array by loading a new set of 512×512 weights every time

before processing. Such a method would result in higher latency incurred by very expensive ReRAM write operations. A simpler solution is to let the PE storage capability to be equal to the largest weight matrix size. So matrix size of 1024×1024 can be mapped into the 9 PE architecture where each PE consists of 8×8 ReRAM arrays with array size of 128×128. The weights can be duplicated for the layers with smaller matrices to improve the area utilization and speed up the computation. Unfortunately, this simple method has larger area due to the larger weight storage requirement in each PE.

TABLE 5.8.
COMPARISON OF CE AND EE FOR SYSTEMS BASED ON ALEXNET AND VGG-19 WITH DIFFERENT PRECISIONS FOR WEIGHT BITS

| **Metrics** | **AlexNet** | | | **VGG-19** | | |
|---|---|---|---|---|---|---|
| | **1b** | **2b** | **4b** | **1b** | **2b** | **4b** |
| CE *TOPs/s/mm²* | 8.42 | 4.35 | 1.98 | 9.6 | 4.3 | 1.91 |
| EE *TOPs/s/W* | 4.08 | 1.90 | 0.91 | 33.5 | 16.6 | 8.30 |

TABLE 5.9.
COMPARISON OF CE AND EE FOR SYSTEMS BASED ON ALEXNET AND VGG-19 WITH DIFFERENT PRECISIONS FOR ACTIVATION BITS

| **Metrics** | **AlexNet** | | | **VGG-19** | | |
|---|---|---|---|---|---|---|
| | **1b** | **2b** | **4b** | **1b** | **2b** | **4b** |
| CE *TOPs/s/mm²* | 8.42 | 4.50 | 2.15 | 9.6 | 4.4 | 2.1 |
| EE *TOPs/s/W* | 4.08 | 2.07 | 1.04 | 33.5 | 17.6 | 9.2 |

### 5.8.2 Multi-bit Weights

MAX$^2$ can be extended to support different precision of weight bits by either using MLC ReRAM or more tiles. We prefer to use more tiles since MLC ReRAM is still a premature technology. For example, for VGG-19, to support 2-bit weight precision, we can use 12 tiles

for computation with MSB weights and 12 tiles for computation with LSB weights. The OFM results of each layer are then obtained by adding the OFM results of MSB tiles and LSB tiles. These OFM results are then separated to MSB and LSB and sent to the corresponding group of tiles for processing the next layer.

Table 5.8 represents the CE and EE results for AlexNet and VGG-19 when the weight precision is 2b and 4b and activation bit is 1 bit. For both networks, we see that CE and EE for higher precisions decrease due to significant increase in area, energy as well as slight increase in latency. SE also increases slightly with increased weight precision since storage weights increase by 2× and 4× while area increases by 1.6× and 3.1×. Compared to VGG-19, AlexNet has much lower EE and SE since AlexNet uses more tiles and more buffers to implement stacked filter computation. Both network implementations have comparable CE.

**Accuracy Analysis:** When the number of activation bit is fixed, there is only a slight accuracy improvement when the number of weight bits is increased from 1-bit to 4-bit for both AlexNet and VGG-19. For example, with 1 bit activation, for VGG-19, the accuracy increases from 61.7% to 62.6% for CIFAR-100 dataset; for AlexNet, the accuracy increases from 64.7% to 64.9%.

### 5.8.3 Multi-bit Activations

When the number of activation bits is greater than 1, IFM data is processed from LSB to MSB. For the 2-bit activation bit case, we first compute LSB OFM results by processing LSB IFM data and then store it in the IO interface. Next the MSB OFM results are computed by processing MSB IFM data, and the final OFM results are obtained by adding MSB OFM results and LSB OFM results using the Shift & Add unit. These OFM results are then separated to MSB and LSB by the non-linear activation function unit and sent to the next layer.

Increasing number of activation bits increases the inference latency since IFM data has to be fed into tiles multiple times. Also, corresponding peripheral circuitry such as Shift & Add unit and non-linear activation function unit have to be enlarged to support more activation bits. Table IX lists the CE and EE results for AlexNet and VGG-19 when the weight precision is fixed at 1b and activation precision is 1b, 2b and 4b. For a given network, we find that the SE is fixed for different activation precisions since the total area does not change. We see that CE and EE drops linear with increasing number of activation bits due to linear increase in latency and energy consumption.

**Accuracy Analysis:** For 1-bit weight precision, increasing precision of activation bits from 1- bit to 4-bit, we see that the accuracy of VGG-19 network can be improved from 61.7% to 64.4% and the accuracy of AlexNet network can be improved from 64.7% to 66.4%.

## 5.9 Related Work

**CMOS-based CNN Accelerators:** Several architectural studies of CMOS-based neural accelerators have been proposed in recent years. DaDianNao [92], an accelerator for CNNs and DNNs, uses eDRAM to store tens of megabytes of weights and activations on-chip, thereby avoiding off-chip accesses. The same neural function unit is used to process all layers to improve area-efficiency. DianNao [93] does not support local reuse but implements specialized registers to store partial sums in the PE array, thereby reducing energy consumption. ShiDianNao [94] explores the output stationary dataflow, where each PE handles the processing for each OFM value by fetching the corresponding IFM from neighboring PEs. The PuDianNao accelerator [95] supports the computation of multiple ML techniques by designing functional units for common computational primitives as well as on-chip storage. Eyeriss [96] proposes a novel dataflow which maximizes IFM reuse and hence

minimizes energy consumption. RedEye [97] moves processing of convolution layers to an image sensor's analog domain to reduce computational burden.

**ReRAM-based CNN Accelerators:** There have been several recent works that explore the use of memristors for DNNs. PRIME [37] and ISAAC [35] were the first to propose use of ReRAM to implement a CNN accelerator. PRIME shows how networks of different scales can be mapped onto the same architecture. Compared to PRIME, which is a general purpose accelerator for neural networks, ISAAC is customized for deep CNNs and achieves better performance. It implements an inter-layer pipeline along with proper weight replication (in early layers) to improve the performance as well as relax the buffering requirements between layers. In order to remove the inter-layer data dependency due to the deep pipeline, PipeLayer [36] computes outputs layer by layer but processes multiple images in a pipelined fashion. It broadcasts the input feature map (IFM) to the subarrays and also supports replication of weights to speed up the intra-layer computation. However, the mapping method used in both ISAAC and PipeLayer results in very different designs for different network structures. This is because the number of subarrays for each layer computation depends on the matrix size, which varies from layer to layer as well as benchmark to benchmark. AEPE [98] is a multi-tiled architecture, where each tile consists of $m \times n$ PEs, where $m$ is the number of channels and $n$ is the number of filters. While each PE consists of a 128×128 ReRAM subarray and associated peripheral circuitry, the number of PEs per tile is different. In contrast, in MAX$^2$, all tiles have the same structure. By utilizing the fact that a larger receptive field filter can be replaced with a stack of 3×3 filters, MAX$^2$ guarantees that every tile consists of 9 PEs that operate in a pipelined fashion. The number of ReRAM subarrays in a PE in MAX$^2$ are however different for each network (4×4 in VGG-19 vs 3×3 in AlexNet).

ISAAC, PipeLayer and PRIME duplicate weights and broadcast the IFM to speed up the computation at the price of additional resource overhead, high intra-layer bandwidth and large sized buffers. In order to improve IFM reuse, AtomLayer [99] use a chain of registers (named register ladders) along with a big buffer ladder to move the data inside the PEs. Each row of IFM is broadcast to the buffer ladders, resulting in high bandwidth requirement. Also, the data movement in each PE is quite complex. The method in [100] maps multiple filters onto a single array and computes multiple outputs at the same time. This method makes use of larger ReRAM array size (512×512). MAX$^2$ uses ReRAM arrays of size 64×64 or 128×128. Our analysis showed that use of larger array sizes resulted in unreliable design due to sneak paths. The sparsity in CNN parameters and activations are leveraged in ReCOM [101] to design an accelerator for sparse vector-matrix multiplication. Our previous work shows that how framework can be modified to support weights and activations of 1-bit, 2-bits and 4-bits [40].

## 5.10 Conclusion

In this work, we propose, MAX$^2$, a ReRAM based CNN accelerator design that achieves very high timing and energy performance for VGG-Net, AlexNet and ResNet. The accelerator is based on a systolic array design which minimizes interconnection cost and intra-layer bandwidth requirement. Each PE in the systolic array is built with multiple 1T1R ReRAM subarrays. For instance, for VGG-19 and ResNet, each PE consists of 4×4 1T1R subarrays where the 4 subarrays along a row share a LUT to keep the LUT overhead < 10% of the PE area. To support different matrix sizes in different layers, we choose different data size granularity in the systolic array in conjunction with weight duplication factor to achieve very high area utilization without requiring additional peripheral circuits. MAX$^2$ can be extended to support different precision of weights bit, different precision of activation bits at the expense

of additional area and energy. It can also be extended to handle other networks that utilize a larger receptive field size by stacking multiple tiles. Simulations using NeuroSim [88] on VGG-19 show that MAX$^2$ with 1-bit weight and 1-bit activation can improve computation efficiency (TOPs/s/mm$^2$) by 2.5× and energy efficiency (TOPs/s/W) by 5.2× compared to a state-of-the-art ReRAM-based accelerator [35]. Similarly, for AlexNet, MAX$^2$ improves computation efficiency by 2.9× and energy efficiency by 8.2× compared to [35]. The enhanced performance is due to higher throughput through use of systolic array with data interleaving, and lower latency through duplicating weights in shallow layers and processing multiple outputs at the same time. Finally, while this paper presents three versions of MAX$^2$ accelerator customized to the three networks, a single accelerator architecture could have also been designed to support all three networks. Such an architecture would have either low storage utilization or high latency. For instance, the 12-tile architecture optimized for VGG-19 could be used to support both AlexNet and ResNet but at the price of lower utilization for AlexNet and higher latency for ResNet.

# CHAPTER 6

# CONCLUSION

In this thesis, we first analyze reliability issues of 2D ReRAM systems, based on 1T1R and, 1S1R and 3D systems based on 1S1R. For each ReRAM system, we build an error model based on physical characteristics that include the effect of variations and provide multi-layer solutions to enhance reliability, energy-efficiency and latency-efficiency. We also propose an ReRAM-based accelerator for CNN inference operation. In this chapter, we summarize our contributions in ReRAM-based storage systems, and ReRAM-based CNN accelerator design. We also provide pointers for future work in this area.

## 6.1 1T1R 2D ReRAM System

For 1T1R 2D ReRAM system, we consider the effect of both retention and endurance errors on ReRAM reliability and propose cross-layer techniques to improve reliability with minimum latency and energy overhead. Our approach is to design circuit-level and architecture-level techniques to reduce raw Bit Error Rate (BER) significantly and then employ low cost Error Control Coding (ECC) to achieve the desired lifetime. At the circuit level, we develop efficient programming strategies to improve the latency, energy, and reliability of 1T1R ReRAM by using a single WL voltage for both the operations, thereby reducing write latency and energy. Next, we show how the retention time of ReRAM cell can be prolonged by increasing the ratio between OFF/ON ratio, while endurance can be improved by reducing OFF/ON ratio. Thus, voltage settings that improve retention do not improve endurance and so we present a procedure to choose voltage settings such that both retention errors and endurance errors are minimized. At the architecture level, we propose a bit flipping technique that reduces the number of endurance errors. The proposed flipping technique uses a 2-bit

saturating counter to record the total number of endurance errors and selectively flips the corrupted data after read-and-verify.

We benchmark the different ReRAM systems and compare them with a DRAM system in terms of Instruction Per Cycle (IPC), lifetime and energy. Simulation results using SPEC CPU INT 2006 [31] and DaCapo-9.12 [32] benchmarks show that the proposed system with cross-layer technique can use a simple BCH ($t = 2$) code at the system level to achieve lifetime of 10 years. We show that the proposed system improves the performance of a ReRAM main memory by 5.2% and energy by up to 72% compared to a 1GB DRAM main memory system.

## 6.2 1S1R 2D ReRAM System

For 1S1R ReRAM system, we propose a 1S1R cross-point array system with "multi-bit per access" per subarray that achieves high energy-efficiency and good reliability. At the cell level, we first show the effect of spatial variations and temporal variations on the resistance distribution of an ReRAM cell. We find that the errors due to temporal variation are dominated by SET failures, which can be significantly reduced by a second SET operation. At the array level, we show that multi-bit access per read/write consumes less energy (compared to the conventional single bit access) but at the price of area overhead and lower reliability. We study the resistance distributions due to different variation sources and evaluate the corresponding Bit Error Rate (BER). We find that the multi-bit group that is farthest away from the driver has the highest error rate due to IR drop.

To address the higher error rates caused by multi-bit per read/write scheme, we propose Rotated Multi-array Access scheme, where the multi-bit groups in a data line are retrieved from different locations in each subarray. This guarantees that the error characteristics of all data lines are the same and the BER is one order of magnitude lower than the naïve multi-bit

access scheme. Simulation results using NVSim show that Rotated Multi-array Access scheme with a simple BCH code with $t = 4$ helps achieve Block Failure Rate (BFR) of $10^{-10}$ that corresponds to a lifetime of 10 years. We show that the proposed system saves energy consumption by 41% with only 2% extra area overhead compared to the baseline system, which accesses single bit for read/write per subarray.

## 6.3 1S1R 3D ReRAM System

For 1S1R 3D ReRAM system, we present a full stack approach (from cell to array to system) and analyze its latency, energy and reliability. We first propose a new data organization scheme where data is stored in multiple 3D subarrays. Groups of *NB* bits are distributed across subarrays as well as along the layers of a subarray. In addition, every group of *NB* bits is bit interleaved so that multiple consecutive bits can share a sense amplifier. By using this scheme, the error characteristics of all data lines are the same, resulting in significantly lower Bit Error Rate (BER). Then, we propose two Multi-layer Access Schemes (namely, MAS-I and MAS-II) with high energy-efficiency. MAS-I enables a 16-layer system to access 4 layers simultaneously. Thus, compared to the baseline system where only one layer is activated at a time, MAS-I helps 3D-HRAM system improve its energy-efficiency and area-efficiency. MAS-II enables a 16-layer system to write 8 layers or read 4 layers at the same time, resulting in even higher energy-efficiency but at the price of read performance degradation compared with MAS-I.

We provide an in-depth evaluation of 3D-HRAM system in terms of energy consumption, read/write performance, reliability and area. To guarantee that all systems have BFR = $10^{-10}$, we use BCH codes with different error correction capabilities. Simulation results using NVSim show that for a given I/O width of 64 bits, the $NB = 16$, $NL = 4$ system based on MAS-I has the lowest energy consumption with 33% energy saving and 15% smaller area overhead

compared to the baseline system. For a wider I/O width of 128 bits, a system with more active layers namely, the $NL = 8$, $NB = 16$ system based on MAS-II, has the lowest energy consumption.

## 6.4 ReRAM-based CNN Accelerator

For ReRAM-based CNN accelerator design, we propose a multiple-tile architectural framework for supporting AlexNet, ResNet and VGGNet-based networks. Each tile consists of 3×3 processing elements (PE) corresponding to a receptive field (or filter) size of 3×3. By implementing larger receptive field filter with a stack of 3×3 filters, all tiles in our design have the same structure since they all are optimized for a 3×3 filter implementation.

We implement a systolic array of PEs with bidirectional connection which maximizes IFM reuse with minimum interconnection cost. To support ResNet and VGGNet-based network, each PE consists of 4×4 ReRAM based arrays, where 4 subarrays along a row share a Look up Table (LUT). We impose constraints on intra-PE and inter-PE bandwidth as well as LUT size to design a realistic architecture. We present several architectural approaches to improve timing-efficiency, energy-efficiency, and area-efficiency of MAX$^2$. To support different matrix sizes in different layers with same sized tiles, we choose different data granularity in systolic array processing in conjunction with weight duplication to achieve very high area utilization (95.7%) without requiring additional peripheral circuits.

Finally, we provide an in-depth system-level evaluation of MAX$^2$ for VGGNet, ResNet and AlexNet-based benchmarks based on NeuroSim [88]. Simulation results show that for VGG-19, MAX$^2$ implemented with 1-bit weight and 1-bit activation can improve computation efficiency (TOPs/s/mm$^2$) by 2.5×, energy efficiency (TOPs/s/W) by 5.2× compared to a state-of-the-art ReRAM-based accelerator [35].

## 6.5 Future Work

In the near future, we plan to improve our work in the following ways. For 2D 1T1R ReRAM system, endurance error and retention error at the cell level were well modeled but the errors due to IR drop and sneak paths at the subarray level, were not considered. Thus actual BER values are likely to be higher, resulting in need for stronger ECC. For 2D 1S1R ReRAM system, we only evaluated energy and latency for a one-time access. A more detailed system-level analysis of benchmarks should be done. For 3D ReRAM system, we only considered the 3D horizontal structure. Our current access strategies have to be modified to support 3D vertical structure, which is more practical due to higher density. Our CNN/DNN accelerator framework work, did not consider the interconnection cost. We plan to revise area, latency and energy estimates to include interconnection cost. Second, we plan to include the extensions that would be needed when the ReRAM storage is smaller than the total number of weights. This is likely to happen as the networks become larger and deeper. Third, we only considered networks which utilize a 3×3 receptive field. Recent popular networks like DenseNet and MobileNet utilize a 1×1 receptive field. Our current design has to be modified to be able to handle these networks.

Finally, sparse neural networks are becoming increasingly popular because of their lower storage and computation cost. However, these networks have lower redundancy and so less tolerant to erroneous weights. Our goal is to improve the reliability of sparse networks. The current plan is to design a regularizer, which keeps variance of the weights low at the block level, while keeping the global variance of the weights at the level high. We expect such a method to be able to improve reliability with minimal effect on the accuracy.

REFERENCE


[1] S. Yu, "Resistive Random Access Memory (RRAM): From Devices to Array Architectures," in San Rafael, CA, USA: Morgan & Claypool, 2016, pp. 1–79.

[2] S. Raghunathan, and K. Roy, "Future cache design using STT MRAMs for improved energy efficiency: devices, circuits and architecture," in ACM Design Automation Conference, 2012.

[3] M. Jung, J. Shalf, and M. Kandemir, "Design of a large-scale storage-class RRAM system," in ACM International Conference on Supercomputing, 2013.

[4] H.-S. P. Wong et al., "Metal-oxide RRAM," in Proc. IEEE, vol. 100, no. 6, pp. 1951–1970, Jun. 2012.

[5] X. Xue, et al. "A 0.13μm 8Mb Logic Based CuxSiyO Resistive Memory with Self-Adaptive Yield Enhancement and Operation Power Reduction," in VLSI Circuits, pp. 42-43, 2012.

[6] M. Chang et al., "19.4 embedded 1 Mb ReRAM in 28 nm CMOS with 0.27-to-1 V read using swing-sample-and-couple sense amplifier and self-boost-write-termination scheme," in IEEE ISSCC Tech. Dig., 2014, pp. 332–333.

[7] Y. Y. Chen et al., "Balancing SET/RESET pulse for > 1010 endurance in HfO2 1T1R bipolar RRAM," IEEE Trans. Electron Devices, vol. 59, no. 12, pp. 3243–3249, Dec. 2012.

[8] C. Xu et al., "Understanding the trade-offs in multi-level cell ReRAM memory design," in Design Automat. Conf., 2013, pp. 1–6.

[9] D. Niu et al., "Design of cross-point metal-oxide ReRAM emphasizing reliability and cost," IEEE Comput.-Aided Design, pp. 17–23, 2013.

[10] M. Mao, et. al, "Programming Strategies to Improve Energy Efficiency and Reliability of ReRAM Memory Systems," Proc. of IEEE Workshop on Signal Processing Systems (SiPS), Oct. 2015.

[11] M. Mao, et. al, "Optimizing Latency, Energy, and Reliability of 1T1R ReRAM through Appropriate Voltage Settings," Proc. of IEEE International Conference on Computer Design (ICCD), Oct. 2015

[12] M. Mao, et. al, "Optimizing Latency, Energy, and Reliability of 1T1R ReRAM through Cross-layer Techniques," IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), 2015.

[13] Y. Deng et al., "RRAM crossbar array with cell selection device: A device and circuit interaction study," in IEEE Trans. Electron Devices, vol. 60, no. 2, pp. 719–726, Feb. 2013.

[14] L. Zhang, S. Cosemans, D. J. Wouters, G. Groeseneken, M. Jurczak, and B. Govoreanu, "Selector design considerations and requirements for 1 SIR RRAM crossbar array," in Proc. IEEE 6th Int. Memory Workshop (IMW), May 2014, pp. 1–4.

[15] L. Zhang, S. Cosemans, D. J. Wouters, G. Groeseneken, M. Jurczak, and B. Govoreanu, "Cell variability impact on the one-selector one-resistor cross-point array performance," IEEE Trans. Electron Devices, vol. 62, no. 11, pp. 3490–3497, Nov. 2015.

[16] S. Lee, S. Lee, K. Moon, J. Park, B. Kim, and H. Hwang, "Comprehensive methodology for ReRAM and selector design guideline of crosspoint array," in Proc. IEEE Int. Memory Workshop (IMW), May 2015, pp. 1–4.

[17] D. Niu, C. Xu, N. Muralimanohar, N. P. Jouppi, and Y. Xie, "Design of cross-point metal-oxide ReRAM emphasizing reliability and cost," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD), Nov. 2013, pp. 17–23.

[18] D. Niu, C. Xu, N. Muralimanohar, N. P. Jouppi, and Y. Xie, "Design trade-offs for high density cross-point resistive memory," in Proc. ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED), Aug. 2012, pp. 209–214.

[19] C. Xu et al., "Overcoming the challenges of crossbar resistive memory architectures," in Proc. IEEE 21st Int. Symp. High Perform. Comput. Archit. (HPCA), Feb. 2015, pp. 476–488.

[20] M. Mao, et. al, "A Multi-layer Approach to Designing Energy-efficient and Reliable ReRAM Cross-point Array System," in IEEE Transactions on Very Large Scale Integration Systems (TVLSI), May 2017.

[21] R. Fackenthal et al., "A 16 Gb ReRAM with 200 MB/s write and 1 GB/s read in 27 nm technology," in IEEE Int. Solid- State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2014, pp. 338–339.

[22] T.-Y. Liu et al., "A 130.7 mm2 2-layer 32 Gb ReRAM memory device in 24 nm technology," in IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2013, pp. 210–211.

[23] C. J. Chevallier et al., "A 0.13 μm 64 Mb multi-layered conductive metal-oxide memory," in IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2010, pp. 260–261.

[24] S. Lee et al., "Full chip integration of 3-D cross-point ReRAM with leakage-compensating write driver and disturbance-aware sense amplifier," in Proc. IEEE Symp. VLSI Circuits (VLSI-Circuits), Jun. 2016, pp. 1–2.

[25] W. C. Chien et al., "Multi-layer sidewall WOX resistive memory suitable for 3D ReRAM," in VLSI Symp. Tech. Dig., Jun. 2012, pp. 153–154.

[26] H.-Y. Chen, S. Yu, B. Gao, P. Huang, J. F. Kang, and H.-S. P. Wong, "HfOx based vertical resistive random access memory for cost-effective 3D cross-point architecture without cell selector," in IEDM Tech. Dig., Dec. 2012, pp. 497–500.

[27] M. Mao, et. al, "Design and Analysis of Energy-efficient and Reliable 3D ReRAM Cross-point Array System," in IEEE Transactions on Very Large Scale Integration Systems (TVLSI), Oct. 2018.

[28] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," in Nature, vol. 521, no. 7553, pp. 436-444, May 2015.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in NIPS, 2012.

[30] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in ICCV, pp. 2722-2730, 2015

[31] D. Silver, et al., "Mastering the game of Go with deep neural networks and tree search," in Nature, col. 529, no. 7587, pp. 484-489, 2016.

[32] V. Sze, et al., "Efficient Processing of Deep Neural Networks: A Tutorial and Survey", in Proceedings of the IEEE, Vol: 105, pp. 2295- 2329, 2017.

[33] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," ICLR, 2015.

[34] M. Horowitz, "Computing's energy problem (and what we can do about it)," ISSCC, pp. 10-14, 2014.

[35] A. Shafiee, et al., "ISAAC: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," in ISCA, pp. 14-26, 2016

[36] L. Song, et al., "PipeLayer: A pipelined ReRAM-based accelerator for deep learning," in HPCA, pp. 541-552, 2017.

[37] P. Chi, et al., "Prime: A novel processing-in-memory architecture for neural network computation in ReRAM-based main memory," in ISCA, pp. 27-39, 2016.

[38] G.W. Burr, et al., "Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses) using phase-change memory as the synaptic weight element," in IEDM, pp. 3498-3507, 2014.

[39] M. Mao, et. al, "MAX2: An ReRAM-based Neural Network Accelerator that Maximizes Data Reuse and Area Utilization," will submitted to IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), 2018.

[40] M. Mao, et. al, "A Versatile ReRAM-based Accelerator for Convolutional Neural Networks", Proc. of IEEE Workshop on Signal Processing Systems (SiPS), Oct. 2018.

[41] Y. Y. Chen et al., "Improvement of data retention in HfO2/Hf1T1R RRAM cell under low operating current," in Proc. IEEE Int. Electron Devices Meet., pp. 252–255, 2013.

[42] M.-J. Lee, C. B. Lee, D. Lee, S. R. Lee, M. S. Change, J. H. Hur, "A fast, high-endurance and scalable non-volatile memory device made from asymmetric Ta2O5-x/TaO2-x bilayer structures," Nature Materials, Vol. 10, pp. 625-630, 2011.

[43] X. Guan, S. Yu, and H.-S. P. Wong, "A SPICE compact model of metal oxide resistive switching memory with variations," IEEE Electron Device Lett., vol. 33, no. 10, pp. 1405–1407, Oct. 2012.

[44] PTM Model [Online]. Available: http://ptm.asu.edu/

[45] MICRON DDR3 Datasheet [Online]. Available: http://www.micron.com/products/dram/ddr3-sdram

[46] C. Xu et al., "Reliability-aware cross-point resistive memory design," in GLSVLSI, 2014, pp. 145–150.

[47] D. Niu, Y. Xiao, and Y. Xie, "Low power memristor-based ReRAM design with error correcting code," in Design Automat. Conf., 2012, pp. 79–84.

[48] Y. Y. Chen et al., "Postcycling LRS retention analysis in HfO2/Hf RRAM 1T1R device," in IEEE Electron Device Lett., vol. 34, no. 5, pp. 626–628, May 2013.

[49] Y. Y. Chen et al., "Understanding of the endurance failure in scaled -based 1T1R RRAM through vacancy mobility degradation," in IEEE Electron Devices Meet., 2012, pp. 20.3.1–20.3.4.

[50] MATLAB [Online]. Available: http://www.mathworks.com

[51] H. Choi, W. Liu, and W. Sung, "VLSI implementation of BCH error correction for multilevel cell NAND flash memory," IEEE Trans. Very Large Scale (VLSI) Syst., vol. 18, no. 5, pp. 843–847, May 2010.

[52] S. Thoziyoor et al., CACTI 5.1 Tech. Rep. Palo Alto, CA, 2008, HP Labs, Tech. Rep. HPL-2008–20.

[53] GEM5 Simulator [Online]. Available: http://www.m5sim.org/ Main_Page

[54] ITRS [Online]. Available: http://www.itrs.net/home.html

[55] C. Yang et al., "A low cost multi-tiered approach to improving the reliability of multi-level cell PRAM," Signal Process. Syst., vol. 76, no. 2, pp. 133–147, Aug. 2014.

[56] S. W. Wei et al., "High-speed hardware decoder for double error correcting binary BCH codes," IEEE Proc. CSV I, vol. 136, no. 3, pp. 227–231.

[57] D. Strukov., "The area and latency tradeoffs of binary bit-parallel BCH decoders for prospective nanoelectronic memories," in Fortieth Asilomar Conference on Signals, Systems and Computers, pp. 1183-1187, 2006.

[58] 45 nm Open Cell Library. Sunnyvale, CA, Nangate, 2008 [Online]. Available: http://www.nangate.com/

[59] Synopsys Design Compiler [Online]. Available: http://www.synopsys. Com

[60] Standard Performance Evaluation Corp. [Online]. Available: http://www.spec.org/

[61] DaCapo Benchmark Suit [Online]. Available: http://www.dacapobench. org/

[62] M. K. Qureshi et al., "Enhancing lifetime and security of PCM-based main memory with start-gap wear leveling," in Proc. IEEE/ACM 42nd Annu. Int. Symp. Microarchitecture, 2009, pp. 14–23.

[63] J. Huang, et al., "One selector-one resistor (1S1R) crossbar array for high-density flexible memory applications," in Proc. IEEE Int. Electron Device Meeting (IEDM), pp. 31.7.1–31.7.4, 2011.

[64] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi, "NVSim: A circuit-level performance, energy, and area model for emerging nonvolatile memory," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 31, no. 7, pp. 994–1007, Jul. 2012.

[65] P. Chen, et al., "Compact Modeling of RRAM Devices and Its Applications in 1T1R and 1S1R Array Design," in IEEE Trans. On Electron Devices (TED), Vol. 62, No. 12, 2015.

[66] S. H. Jo, et al., "3D-stackable crossbar resistive memory based on field assisted superlinear threshold (FAST) selector," in IEDM, pp. 6.7.1–6.7.4, Dec. 2014.

[67] H. Celano, et al., "Imaging the Three-Dimensional Conductive Channel in Filamentary-Based Oxide Resistive Switching Memory," Nano Lett., 15(12), pp 7970–7975, Dec 2015.

[68] M.-F. Chang et al., "Challenges and circuit techniques for energyefficient on-chip nonvolatile memory using memristive devices," IEEE J. Emerg. Sel. Topics Circuits Syst., vol. 5, no. 2, pp. 183–193, Jun. 2015.

[69] J. Yi, et al., "Requirements of bipolar switching ReRAM for 1T1R type high density memory array," in International Symposium on VLSI Technology, Systems and Applications (VLSI-TSA), pp. 1–2, April 2011.

[70] W. G. Bennett, et al., "Single- and Multiple-Event Induced Upsets in HfO2/Hf 1T1R RRAM," in IEEE Trans. on Nuclear Science, Vol. 61 , Issue. 4, pp. 1717–1725, Aug 2014.

[71] M. Wu, et al., "Low-Power and Highly Reliable Multilevel Operation in ZrO2 1T1R RRAM," in IEEE Electron Device Letters (IEDL). Vol. 32 , Issue. 8, pp. 1026–1028, Jun 2011.

[72] Y. Y. Chen, et al., "Endurance/Retention Trade-off on HfO2/Metal Cap 1T1R Bipolar RRAM," IEEE Trans. On Electron Devices (TED), Vol. 60, pp. 1114-1121, 2013.

[73] P.-Y. Chen et al., "Design Tradeoffs of Vertical RRAM-Based 3-D Cross-Point Array," in IEEE Trans. on VLSI, April 2016, pp. 3460–3467.

[74] H. Li et al., "Device-Architecture Co-Design for Hyper-dimensional Computing with 3D Vertical Resistive Switching Random Access Memory," in IEEE Trans. on VLSI, April 2016, pp. 3460–3467.

[75] P. Sun et al., "Thermal crosstalk in 3-dimensional RRAM crossbar array," in Sci. Rep. 2015, 5, 13504.

[76] Y. Deng, et al., "Design and Optimization Methodology for 3D RRAM Arrays," IEEE International Electron Devices Meeting (IEDM), pp. 629-632, 2013.

[77] C. Xu, et al., "Modeling and Design Analysis of 3D Vertical Resistive Memory - A Low Cost Cross-Point Architecture," ASP-DAC, pp. 825- 830, 2014.

[78] L. Zhang, et al., "Analysis of vertical cross-point resistive memory (VRRAM) for 3D RRAM design," IEEE IMW, pp.155-158, 2013.

[79] S.-S. Sheu et al., "A 4Mb embedded SLC resistive-RAM macro with 7.2ns read-write random-access time and 160ns MLC-access capability," in IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2011, pp. 200–202.

[80] A. Kawahara et al., "An 8 Mb multi-layered cross-point ReRAM macro with 443 MB/s write throughput," in IEEE J. Solid-State Circuits, vol. 48, no. 1, pp. 178–185, Jan. 2012.

[81] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," CVPR, pp. 770- 778, 2016.

[82] Y. Chen, et al., "DaDianNao: A Machine-Learning Supercomputer," MICRO-47, pp. 609-622, 2014.

[83] B. Akin et al., "Data reorganization in memory using 3D-stacked DRAM," in ACM SIGARCH Computer Architecture News, vol. 43, pp. 131-143, ACM, 2015.

[84] J. Ahn et al., "A scalable processing-in-memory accelerator for parallel graph processing," ISCA, pp. 105-117, 2015.

[85] S. Yu, "Neuro-inspired computing with emerging non-volatile memory," Proc. IEEE, vol. 106, no. 2, pp. 260-285, 2018.

[86] H. T. Kung, "Why systolic architecture?", Vol. 15, Issue No. 01, pp: 37-46, 1982.

[87] X. Sun, et al., "XNOR-RRAM: A scalable and parallel synaptic architecture for binary neural networks," DATE, 2018

[88] P.-Y. Chen, et al., "NeuroSim+: An integrated device-to-algorithm framework for benchmarking synaptic devices and array architectures," IEDM, pp. 6.1.1-6.1.4, 2017.

[89] Han, S., et al., "EIE: efficient inference engine on compressed deep neural network," ISCA, pp. 243-254, 2016.

[90] L. Ni, et al., "An energy-efficient and high-throughput bitwise CNN on sneak-path-free digital ReRAM crossbar," IEEE/ACM ISLPED, pp. 1-6, 2017.

[91] T. Tang, et al., "Binary convolutional neural network on RRAM," ACM/IEEE ASP-DAC, pp. 782-787, 2017.

[92] Y. Chen, et al., "DaDianNao: A Machine-Learning Supercomputer," in Proceedings of MICRO-47, pp. 609-622, 2014.

[93] T. Chen, et al., "DianNao: A Small-Footprint High-Throughput Accelerator for Ubiquitous Machine-Learning," in Proceedings of ASPLOS, pp. 269-284, 2014.

[94] Z. Du, et al., "ShiDianNao: Shifting Vision Processing Closer to the Sensor," in Proceedings of ISCA-42, pp. 92-104, 2015.

[95] D. Liu, et al., "PuDianNao: A Polyvalent Machine Learning Accelerator," in Proceedings of ASPLOS-20, pp. 369-381, 2015.

[96] Y. H. Chen, et al., "Eyeriss: A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks," in ISCA, pp. 367-379, 2016.

[97] R. LiKamWa, et al., "RedEye: Analog ConvNet Image Sensor Architecture for Continuous Mobile Vision," in Proceedings of ISCA-43, pp. 255-266, 2016.

[98] S. Tang, et al., "AEPE: An area and power efficient RRAM crossbar-based accelerator for deep CNNs," in Proceedings of NVMSA, 2016.

[99] X. Qiao, et al., "AtomLayer: A Universal ReRAM-Based CNN Accelerator with Atomic Layer Computation", DAC, pp. 1-6, 2018.

[100] Z. Zhu, et al., "Mixed size crossbar based RRAM CNN accelerator with overlapped mapping method," in Proceedings of ICCAD, pp. 1-6. 2016.

[101] H. Ji, et al., "ReCom: An efficient resistive accelerator for compressed deep neural networks," in DATE, pp. 237-240, 2018.

# APPENDIX

## A. Mapping of Weights

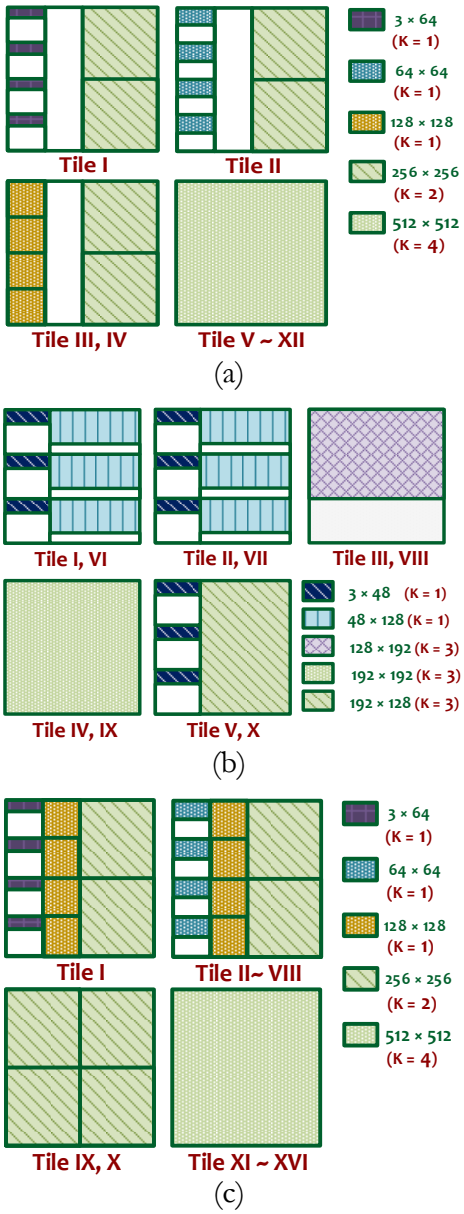



Figure 7.1. Mapping of weights from (a) VGG-19, (b) AlexNet and (c) ResNet

**VGG-19:** MAX$^2$ design for VGG-19 is a 12-tile architecture, where some tiles store weights of multiple layers and others store weights of only one layer. The weight matrix size for a layer is $m \times n$, where $m$ is the number of channels and $n$ is the number of filters. Here, tile I stores

weights of layer 1 (3×64) and layer 5 (256×256), tile II stores weights of layer 2 (64×64) and layer 6 (256×256), tile III stores weights of layer 3 (64×64) and layer 7 (256×256); tile IV stores weights of layer 4 (64×64) and layer 8 (256×256); and tiles V to XII store weights of layer 9 (512×512) to layer 16 (512×512). This weight mapping is shown in Fig. 7.1 (a).

**AlexNet:** MAX$^2$ design for AlexNet is a 10-tile architecture. As in the VGG-19 architecture, the weight matrices of multiple layers are sometime stored in the same tile to improve area efficiency. The weight mapping is shown in Fig. 7.1 (b). In order to implement the 7×7 filter in layer 1, the three 3×3 filters weights of layer 1 (3×48) are loaded into tile I, tile II and tile V. The weights of layer 2 (48×128) are loaded into tile I and tile II to implement a 5×5 filter. The filter sizes for layers 3, 4 and 5 are 3×3 and so the corresponding weights are loaded into one tile per layer. Specifically, we store weights from layer 3 (128×192) into tile IV; and store weights from layer 4 (192×192) into tile V. Tiles VI to X use the same weight mapping topology as tiles I to V.

**ResNet:** MAX$^2$ design for ResNet is a 16-tile architecture, which uses the same PE design as VGG-19. In order to implement the 5×5 filter in layer 1, the two 3×3 filters weights of layer 1 are loaded into tile I and tile II. The filter sizes for the rest of layers are 3×3 and so the corresponding weights can be loaded into one tile per layer. We store weights from layer 2 to layer 7 (64×64) into tile III ~ VIII; store weights from layer 8 to layer 15 (128×128) into tile I ~ VIII; store weights from layer 16 to layer 27 (256×256) into tile I ~ X; and store weights from layer 28 to layer 33 (512×512) into tile XI ~ XVI.

B. Mapping of Layers

Figure 7.2 shows the mapping between logical layers of AlexNet and physical tiles of the accelerator. In the first AlexNet layer, the receptive field of size 7×7 is replaced by a stack of

three 3×3 filters, which are stored in the left subarrays in tile I, tile II and tile III, respectively (marked in a red rectangle). Similarly, the receptive field of size 5×5 for the second layer is obtained by stacking two 3×3 filters, which are stored in the right subarrays of tiles I and II. Thus, layer 1 and layer 2 cannot be processed in a pipelined fashion since only 3 subarrays along the column can be accessed at a time in one PE due to LUT limitation. Layers 2 to 5 can start processing in a pipelined fashion after layer 1 finishes the computation.

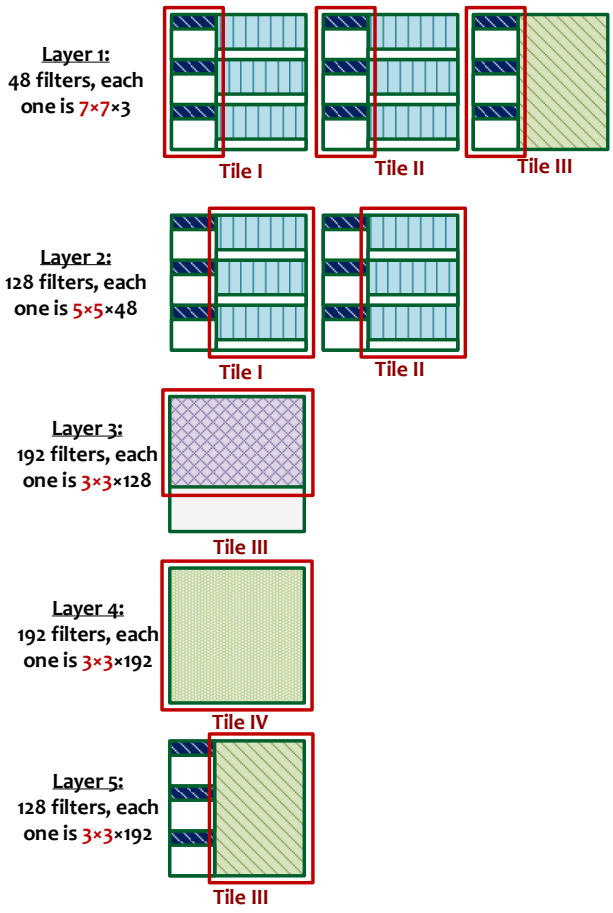



Figure 7.2. Layer topologies for AlexNet.

For VGG-19, the filter size is 3×3 for the whole network so weights of one layer in VGG-19 are mapped to one tile in pipelined fashion; Layers 5 to 8 can start processing in a pipelined

fashion after Layers 1 to 4 finish the computation. Finally, Layers 9 to 16 can start processing in a pipelined fashion after Layers 5 to 8 finish the computation.

For ResNet, the filter in the first CONV layer has receptive field of size 5×5, which can be replaced by a stack of two 3×3 filters. Thus, Layers 1 to 7 can be processed in a pipelined fashion; Layers 8 to 15 can start processing in a pipelined fashion after Layers 1 to 7 finish the computation. Layers 16 to 25 can start processing in a pipelined fashion after Layers 8 to 15 finish the computation. Finally, Layers 26 to 33 can start processing in a pipelined fashion after Layers 16 to 25 finish the computation.