Emerging and Nonvolatile Memory

14

Chun Jason Xue

Abstract

In recent years, Non-Volatile Memory (NVM) technologies have emerged as candidates for future computer memory. Nonvolatility, the ability of storing information even after powered off, essentially differentiates them from traditional CMOS-based memory technologies. In addition to the nonvolatility, NVMs are also favored because of their low leakage power, high density, and comparable read speed compared with volatile memories. However, there are challenges to efficiently utilize NVMs due to the high write cost and potential endurance issues. In this chapter, we first introduce representative NVM technologies including their physical construction for data storage, as well as characteristics, and then summarize recent work aiming to exploring NVMs' characteristic to optimize their behaviors.

Acronyms

CMOS	Complementary Metal-Oxide-Semiconductor
DRAM	Dynamic Random-Access Memory
DWM	Domain Wall Memory
FeRAM	Ferro-electric Random-Access Memory
MTJ	Magnetic Tunnel Junction
NMOS	Negative-type Metal-Oxide-Semiconductor
NVM	Non-Volatile Memory
PCM	Phase Change Memory
RRAM	Resistive Random-Access Memory
SRAM	Static Random-Access Memory
STT-RAM	Spin-Transfer Torque Random-Access Memory
WL	Word Line

C.J. Xue (✉)
City University of Hong Kong, Hong Kong, Hong Kong
e-mail: jasonxue@cityu.edu.hk

S. Ha, J. Teich (eds.), *Handbook of Hardware/Software Codesign*,
DOI 10.1007/978-94-017-7267-9_15

Contents

14.1 Introduction

With the continuously increasing scalability, traditional CMOS-based memories are facing great challenges. Taking Dynamic Random-Access Memory (DRAM) as an example, the limited scalability and large leakage power make it undesirable for next-generation main memory. Emerging Non-Volatile Memories (NVMs) are proposed to take this challenge in future computing systems due to several promising advantages. First, NVMs have high scalability. For example, Phase Change Memory (PCM), a representative NVM, has been demonstrated in a 20 nm device prototype and is projected to scale to 9 nm, while manufacturable solutions for scaling DRAM beyond 40 nm are unknown [1, 22, 44]. Second, NVMs have a much larger storage density. In addition to scalability, the feasibility of storing multiple bits in one NVM cell further enlarges the density. Third, NVMs are nonvolatile, indicating that data will not be lost even the memory is out of power supply.

However, NVMs are commonly associated with large programming cost and possible endurance issues. As a result, new management and optimization policies should be proposed to efficiently utilize NVMs in computer and embedded systems. These policies should fully exploit the physical characteristics of NVMs, which are significantly different from volatile memories, and also tune their behaviors to fit into the memory hierarchy.

14.2 Classification of Emerging Nonvolatile Memories

Emerging nonvolatile memory includes Spin-Transfer Torque Random-Access Memory (STT-RAM), Resistive Random-Access Memory (RRAM), Domain Wall Memory (DWM), Ferro-electric Random-Access Memory (FeRAM), PCM, and so on. Various technologies differ in cell size, endurance, access speed, leakage, and dynamic power, making them fit different levels in memory hierarchy. The detailed characteristics are summarized in Table 14.1. In the following, the specific physical rationality of each representative NVM is introduced.

14.2.1 Spin-Transfer Torque Random-Access Memory

The information carrier of STT-RAM is a Magnetic Tunnel Junction (MTJ) [74]. Each MTJ contains two ferromagnetic layers and one tunnel barrier layer. One of the ferromagnetic layers (reference layer) has fixed magnetic direction, while the other one (free layer) can change its magnetic direction by an external electromagnetic field or a spin-transfer torque. If the two ferromagnetic layers have different directions, the MTJ resistance is high, indicating a "1" state; if the two layers have the same direction, the MTJ resistance is low, indicating a "0" state. The STT-RAM cell structure is composed of one Negative-type Metal-Oxide-Semiconductor (NMOS) transistor as the access device and one MTJ as the storage element, as shown in Fig. 14.1. The MTJ is connected in series with the NMOS

Table 14.1 Characteristics of different memory technologies [13, 18, 30, 35, 58, 63, 78]

	STT-RAM	RRAM	DWM	FeRAM	PCM
Cell size (F^2)	6–50	4–10	≥2	∼10	4–12
Write endurance	4×10^{12}	10^{11}	10^{16}	10^{14}	10^8–10^9
Speed (R/W)	Fast/slow	Fast/slow	Fast/slow	Fast/slow	Fast/very slow
Leakage power	Low	Low	Low	Low	Low
Dynamic energy (R/W)	Low/high	Low/high	Low/high	Low/high	Medium/high

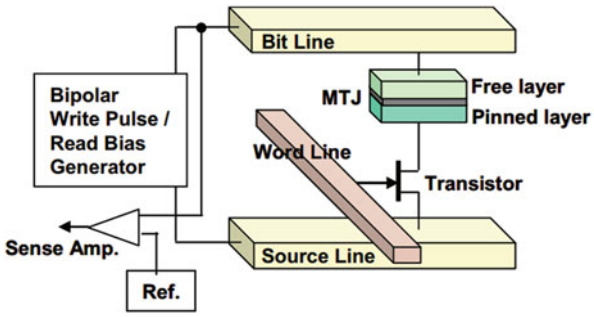

Fig. 14.1 An illustration of a "1T1J" STT-RAM cell [65]

transistor. The NMOS transistor is controlled by the wordline (WL) signal. When a write operation happens, a large positive voltage difference is established for writing "0"s or a large negative one for writing "1"s. The crystalline amplitude required to ensure a successful status reversal is called the threshold current. The current is related to the material of the tunnel barrier layer, the writing pulse duration, and the MTJ geometry [12]. STT-RAM has been widely used for designing caches due to its comparatively faster access and higher endurance than other types of NVM.

14.2.2 Resistive Random-Access Memory

Figure 14.2 shows the cell structure of RRAM. An RRAM with unipolar switching uses an insulating dielectric [26]. When a sufficiently high voltage is applied, a filament or conducting path is formed in the insulating dielectric. After this, by applying suitable voltages, the filament may be set (which leads to a low resistance) or reset (which leads to a high resistance) [35]. Compared to Static Random-Access Memory (SRAM), an RRAM cache has high density, comparable read latency, and much smaller leakage energy. However, the limitation of RRAM is its low write endurance of 10^{11} [21] and high write latency and write energy. For example, a typical 4 MB RRAM cache has a read latency of 6–8 ns and a write latency of 20–30 ns [14].

14.2.3 Domain Wall Memory

DWM works by controlling domain wall (DW) motion in ferromagnetic nanowires [58]. The ferromagnetic wire can have multiple domains which are separated by domain walls. These domains can be individually programmed to store a single bit (in the form of a magnetization direction), and thus, DWM can store multiple bits per memory cell. Logically, a DWM macro-cell appears as a tape, which stores multiple bits and can be shifted in either direction, as shown in Fig. 14.3. The challenge in using DWM is that the time consumed in accessing a bit depends on its location relative to the access port, which leads to nonuniform access latency and

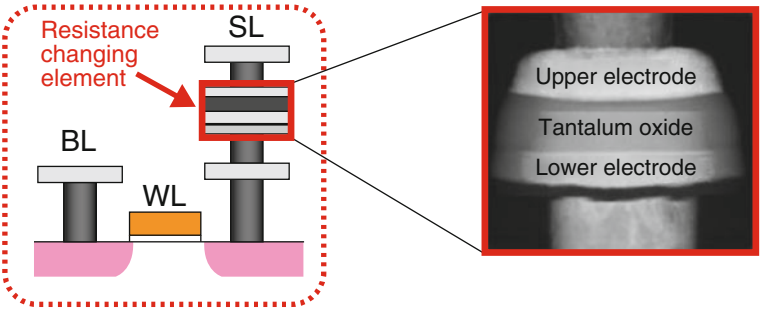

Fig. 14.2 The cell structure of RRAM [2]

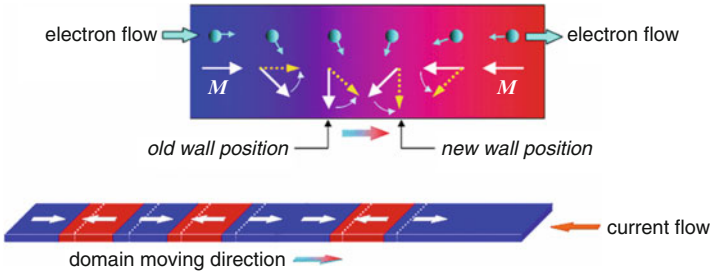

Fig. 14.3 The cell structure of DWM [77]

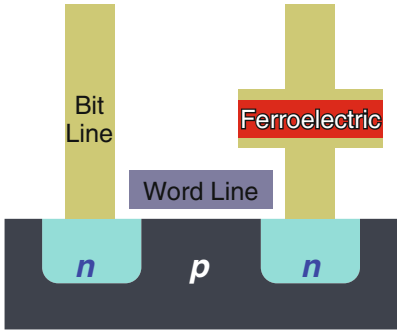

Fig. 14.4 The cell structure of FRAM with 1T-1C design [3]

makes the performance dependent on the number of shift operations required per access. Compared to other NVMs, DWM is less mature and is still in research and prototype phase.

14.2.4 Ferro-electric Random-Access Memory

A FeRAM cell contains materials with two stable polarization states, which can switch from one state to another if an external strong electric field is applied [32]. Figure 14.4 shows a popular design of a ferroelectric memory cell. This 1T-1C design uses one ferroelectric capacitor to store the information [10]. When we need to read a specific ferroelectric capacitor in a 1T-1C cell, a poled reference cell is used. By measuring the voltage/current between the capacitor and the reference cell, the stored value can be determined. The superiority of FeRAM includes nearly unlimited operation cycles, ultrashort access time and easy integration to Complementary Metal-Oxide-Semiconductor (CMOS) technology.

14.2.5 Phase Change Memory

Figure 14.5 shows the schematic of PCM memory array and cells. PCM exploits the large resistance contrast between amorphous and crystalline states in the chalcogenide alloy (GST) material. The amorphous phase tends to have high electrical resistivity, while the crystalline phase exhibits a low resistivity, sometimes

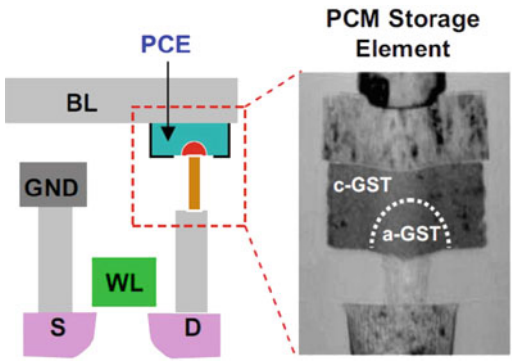

Fig. 14.5 The cell structure of PCM [37]

three or four orders of magnitude lower [44]. The amorphous and crystalline states can be transferred to each other by programming the cell into high/low resistance levels, achieved by heating the PCM cell to be with various amorphous-GST and crystalline-GST portions.

14.3 On-Chip Memory and Optimizations

In this section, the utilization of NVM as on-chip memory is summarized. First, the optimizations of on-chip caches using NVM are discussed, and then the on-chip memory for embedded systems are explored in the context of NVM.

SRAM, which has been typically used as on-chip cache, faces the problems of large leakage power and limited scalability. It is reported that, with ever-increasing required cache size, cache will occupy 90% of the chip area with tremendous fraction of chip power in future CMOS generations if consisted of SRAM [46]. Thus NVMs such as STT-RAM, RRAM, and DWM are proposed to serve as on-chip caches [35]. The write speed of NVM should be optimized when it is applied as L1 cache; the density and access energy are more important when it is used for L2 or lower-level caches [17]. Motivated by these requirements, researches have been conducted to build appropriate cache architecture, develop NVM-oriented optimizations, and revisit cache management policies to achieve high-performance low-power and dense on-chip caches.

14.3.1 STT-RAM as On-Chip Cache

STT-RAM is the mostly recommended alternative of traditional SRAM. The challenges to apply STT-RAM lie in its high programming cost and endurance. In this section, optimizations for access efficiency, endurance, as well as density are summarized.

14.3.1.1 Optimizations for Access Efficiency

Many researches focus on the latency and energy reduction of write operations in STT-RAM.

The nonvolatility of STT-RAM cells can be relaxed by cutting the planar area [52] or reducing the number of programming cycles [19, 28], by which the writes can be faster and energy efficient, however, with a shorter data retention time. The retention time should be guaranteed to be smaller than inter-write time; otherwise, additional refreshes are necessary [55]. STT-RAM cell sizing is studied for the impact on overall computing system performance, showing that different computing workloads may have conflicting expectations on memory cell sizing [64]. Leveraging MTJ device switching characteristics, the authors further propose an STT-RAM architecture design method that can make STT-RAM cache with relatively small memory cell size perform well over a wide spectrum of computing benchmarks [64]. Early write termination is proposed to save programming energy by terminating write operations once it confirmed that the data to write are the same with the old values [76]. Instead of read-before-write, this strategy senses the old data during the write process and thus does not induce latency overhead. Based on the observation that a large fraction of data written to L2 cache are "all-zero-data," flags are employed to label these words to avoid being written and read [20]. Accurate data can be constructed by unlabeled words and zero flags.

In hybrid caches consisting of both SRAM and STT-RAM, cache partitioning is explored in [49] to determine the amount of SRAM and STT-RAM ways by exploiting the performance advantage of SRAM and high density of STT-RAM. A dynamic reconfiguration scheme which determines the portion of SRAM and STT-RAM is proposed in [7] for access energy saving. Data migrations aiming to minimize the number of writes to STT-RAM can reduce the cache energy [40].

14.3.1.2 Optimizations for Endurance

Wear leveling in STT-RAM aims to balance the number of writes across different physical regions so that the cache endurance can be prolonged. The basic idea is to swap write-intensive regions with those rarely written ones. Wear leveling can be conducted with various granularites. A concept of cache color containing a number of sets is developed as the granularity for data swapping in [50]. The mapping between physical regions and cache colors is periodically remapped, with the objective of mapping the mostly written region to cache colors with the least number of writes. In a set-level wear leveling [6], cache sets are reorganized through XOR operation between changing remap register and set indexes in order to balance writes across cache sets. As an intra-set weal leveling, WriteSmoothing [36] logically divides the cache sets into multiple modules. For each module, it collectively records number of writes in each way for any of the sets. WriteSmoothing then periodically makes most frequently written ways in a module unavailable to shift the write pressure to other ways in the sets of the module. A coding scheme for STT-RAM last-level cache is proposed based on the concept of value locality in [67]. The

switching probability in cache can be reduced by swapping common patterns with limited weight codes to make writes less often as well as more uniform. This belongs to bit-level wear leveling.

In hybrid caches consisting of both SRAM and STT-RAM, similar wear leveling strategies can be modified and applied to balance writes among the whole cache space, such as data swapping between SRAM and STT-RAM [27,30]. Prementioned early write termination [76], write mode selection [19, 28], and write reductions [5,20,45] also benefit the cache endurance.

14.3.1.3 Optimizations for Density

Three-dimensional die stacking increases transistor density by vertically integrating multiple dies with a high-bandwidth, high-speed, and low-power interface [35]. Using 3D die stacking, dies of even different types can be stacked. Several researchers discuss 3D stacking of NVM caches [30, 49, 53, 54, 63]. For example, an STT-RAM cache and CMP logic can be designed as two separate dies and stacked together in a vertical manner [53]. One benefit of this is that the magnetic-related fabrication process of STT-RAM may not affect the normal CMOS logic fabrication. Three-dimensional stacking also enables shorter global interconnect, lower interconnect power consumption, and smaller footprint [35]. Cell sizing can also potentially improve the cache density by shrinking STT-RAM cells.

14.3.2 Other NVMs as On-Chip Memory

RRAM and DWM can also be applied as on-chip caches. Compared with STT-RAM, the endurance of RRAM is more serious based on the report that RRAM can withstand 10^{11} writes while STT-RAM can withstand 4×10^{12}. Thus there are researches focusing on the endurance enhancement of RRAM at the levels of both inter-set and intra-set [34,61]. Since DMW exploits the shift of access port to access data, the optimizations of DMW include hiding the shifting time [60] and reduce the shift cost [56,59].

NVMs can also be used in embedded systems as on-chip memory, such as nonvolatile flip-flops [62], Flash, and FeRAM [4]. A novel usage of NVM in embedded systems is to back up volatile program execution states upon power failures, so that it can be resumed after being recharged. In this scenario, the backup efficiency directly affects system performance and energy consumption. Nonvolatile flip-flops are connected to each volatile cell for efficient backup in [62], which is suitable for registers. Contents to back up can be reduced to achieve high performance and energy efficiency [29, 73]. Hybrid on-chip memory with Flash and FeRAM leads to a promising tradeoff between system performance and price [4].

14.4 Hybrid Main Memory and Optimizations

In this section, the utilization of hybrid PCM/DRAM as main memory is summarized. Firstly, the architecture of pure-PCM working as the main memory is developed by a few works. Then the hybrid DRAM/PCM memory architecture overview is presented. Next, we summarize the research works on two different hybrid architectures.

14.4.1 PCM as Main Memory Architecture

As emerging nonvolatile memory technologies, several researches have tried to replace DRAM with nonvolatile memory like PCM, as a candidate of the main memory. The work [23, 75] firstly developed the architecture-level studies on using PCM to implement main memory. [23] proposed to use narrow rows and multiple buffers to improve write coalescing and perform partial writes. [75] took advantage of redundant bit writes to eliminate unnecessary writes to PCM and perform dynamic memory mapping at memory controller to achieve wear-out leveling.

Based on the proposed architecture, some works have been proposed to address different issues of PCM when working as the memory, including improving endurance [23, 41, 42, 75], improving write latency by reducing the writes of bits to PCM [8, 15, 23, 66, 70, 72, 75], and preventing malicious wear-outs [48].

However, pure PCM-based main memory has many challenging issues such as slow latency and limited endurance to be used as the main memory. To overcome these limitations, the hybrid main memory is proposed instead, which is constructed using both PCM and DRAM. The following sections introduce the hybrid memory architecture and summarize the techniques used in the hybrid memory.

14.4.2 PCM/DRAM Hybrid Memory Overview

In the hybrid PCM and DRAM memory, the DRAM can help improve the slow write performance of PCM and also increases the lifetime of PCM by "absorbing" the write-intensive operations to PCM. As shown in Fig. 14.6, two different architectures have been proposed for the hybrid memory system. For the first architecture, as depicted in Fig. 14.6a, DRAM is regarded as an upper-level cache to the main memory, which only consists of PCM, to combine the short latency of DRAM with large capacity of PCM. When a memory access happens, the DRAM cache is checked firstly, and the PCM will be accessed only under the cache miss. In this architecture, the DRAM cache is invisible to the operating system. All the operations to the DRAM cache are managed by the memory controller. Qureshi et al. [42] showed that a small DRAM cache, which is 3% size of the PCM storage, can make up the latency gap between DRAM and PCM.

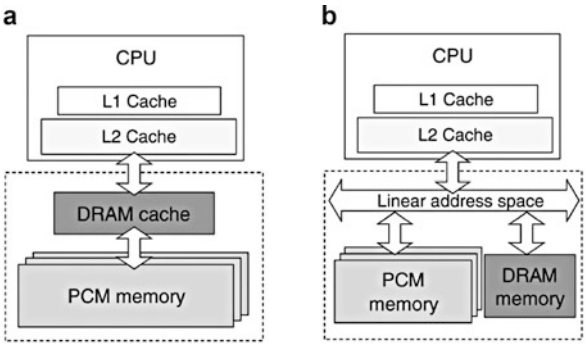

Fig. 14.6 Hybrid memory architectures consisting of both PCM and DRAM [25]. (**a**) DRAM-as-cache hybrid architecture. (**b**) Parallel hybrid memory architecture

In the second memory architecture, as shown in Fig. 14.6b, both of PCM and DRAM memory are directly connected to memory controller in parallel with each other and thus share the single memory physical space. The operating system can determine the page allocation and memory management strategy to improve the performance, energy saving, and endurance. Dhiman et al. [11] proposed the write-frequency-guided page placement policy to perform uniform wear leveling across all PCM pages, while migrating write-intensive pages to DRAM to improve performance.

In the following sections, we summarize the research works about the hybrid DRAM and PCM memory system.

14.4.3 DRAM-as-Cache Architecture

In the DRAM-as-Cache architecture, DRAM works as a cache to serve the memory requests at low latency with low energy cost and high endurance. On the other hand, PCM works as a large background main memory to exploit low standby power as well as the low costs, but the write operations should be reduced because of the limited endurance and high write latency. The key challenge in the design of such a hybrid memory system lies in the following aspects: what is the caching/paging scheme to the DRAM-PCM architecture, in terms of different objectives, including power, performance, capacity, endurance, etc., and what data should be cached in DRAM to best exploit the advantages and overcome the disadvantages of hybrid system.

14.4.3.1 Caching/Paging Schemes to the Hybrid Architecture

A series of works have proposed different caching/paging schemes to the hybrid architecture to minimize the write operations in PCM for the benefits of both PCM endurance and performance and reduce the refreshes of DRAM to explore further energy saving.

Qureshi et al. [42] firstly proposed to increase the size of main memory by using PCM as main memory and using DRAM as a cache to PCM. They developed the

Lazy-Write scheme based on the proposed memory model to minimize the write operations to PCM by only writing the pages fetched from hard disk to DRAM cache. Intra-page wear leveling is also applied in PCM to swap cache lines to achieve uniform writing of lines within pages to further improve the endurance. Experimental results show that the hybrid memory can improve 3× latency and reduce 5× page faults because of the increase in main memory size, compared to the pure-DRAM memory and achieve 3× endurance improvement compared to the pure-PCM memory.

Ferreira et al. [15] proposed write-back minimization scheme with new cache replacement policies and PCM-aware swap algorithm for wear leveling, while avoiding unnecessary writes. Zhang and Li [71] employed the PCM to implement the 3D-stacked memory systems to exploit the low standby power feature of PCM and proposed an OS-level paging scheme that takes into account the memory reference characteristics and migrated the hot-modified pages from PCM to DRAM. In this way, the lifetime degradation of PCM is alleviated, and the energy efficiency of the memory system is also improved.

To reduce the energy consumption of the hybrid system, Park et al. [38] proposed to decay the contents in DRAM. In this way, the clean, old data in DRAM are evicted, while the dirty data are written back to PCM and then evicted. Thus the evicted rows do not need refresh operations, and the energy consumption of the hybrid memory can be reduced compared to the DRAM-only memory. A long-term dirty data write-back scheme is further developed to minimize the PCM writes.

14.4.3.2 What Data Should Be Cached in DRAM

The work in [68, 69] mainly proposed an answer to the question "what data should be placed in DRAM cache." Yoon et al. observed that PCM and DRAM have the same latency to access their row buffers, while the write latency is much higher in PCM than in DRAM when accessing the content which is not stored in the row buffer. For the sake of performance and PCM endurance, the access to PCM when the row buffer miss happens should be avoided. With this observation, Yoon et al. proposed to put those rows of data that are likely to miss in the row buffer and also likely to be reused in DRAM cache. Further, their technique uses a caching policy such that the pages that are written frequently are more likely to stay in DRAM.

Meza et al. [33] propose a technique for efficiently storing and managing the metadata (such as tag, replacement-policy information, valid, and dirty bits) for data in a DRAM cache at a fine granularity. They observed that in DRAM cache storing metadata in the same row as their data can exploit DRAM row buffer locality, and it also reduces the access latency from two row buffer conflicts (one for the metadata and another for the datum itself). Based on this observation, they proposed to put the metadata for recently accessed rows in a small row buffer to the DRAM cache. Since metadata needed for data with temporal or spatial locality is cached on chip, it can be accessed with the similar latency as an SRAM tag store, while providing better energy efficiency than using a large SRAM tag store.

14.4.4 Parallel Hybrid Architecture

In the parallel hybrid architecture, both DRAM and PCM are used as the main memory. This architecture reduces the power budget because large portion of main memory is replaced by PCM.

In the hybrid memory, write-intensive accesses to PCM should be minimized due to its high write latency and lifetime limitation. Several works proposed to migrate write-intensive data/pages from PCM to DRAM to reduce the write accesses to PCM.

Dhiman et al. [11] proposed a hybrid main memory system that exposes both DRAM and PCM to software (operating system). If the number of writes to a particular PCM page exceeds a certain threshold, the contents of the page are copied to another page (either in DRAM or PCM), thus facilitating PCM wear leveling, while movement of hot pages to DRAM leads to saving of energy due to faster and more energy-efficient DRAM accesses.

Work in [43] suggested a management policy that monitors program access patterns and migrates hot pages to DRAM while keeping cold pages in PCM. In this way, the write-intensive operations are avoided, and thus the performance and energy saving are improved. Park et al. [39] proposed the main memory management mechanism of the operating system for the hybrid main memory. They proposed the migration scheme using access bits and power-off technique of DRAM chunk to mitigate background power consumption. Seok et al. [47] proposed a migration-based page caching technique for PCM-DRAM hybrid main memory system. Their technique aims to overcome the problem of the long latency and low endurance of PCM. For this, read-bound access pages are kept in PCM, and write-bound access pages are kept in DRAM. Their technique uses separate read and write queues for both PCM and DRAM and uses page monitoring to make migration decisions. Write-bound pages are migrated from PCM to DRAM, and read-bound pages are migrated from DRAM to PCM. The decision to migrate is taken as follows: when a write access is hit and the accessed page is in PCM write queue, it is migrated. Similarly, if a read access is hit and the accessed page is in the DRAM read queue, it is migrated.

Dynamically migrating data between DRAM and PCM can reduce the writes to PCM, but will bring energy cost and performance delay to the system. To handle the problem, reducing frequent page migration between PCM and DRAM is targeted at work [51]. Shin et al. proposed to store the page information and let it be visible to the operating system. With the information, page migration granularity can be dynamically changed based on whether the migration of pages is heavy. Heavy migration implies that pages which have similar access properties can be grouped to reduce the migration frequency.

Instead of dynamically migrating data between DRAM and PCM, a series of work [9, 16, 31] assumed that memory accessing patterns are given or can be predicted and proposed data allocation schemes in the hybrid memory to optimize different objectives. Choi et al. [9] developed the page-level allocation technique to

obtain the optimal performance, with well-designed proportion of DRAM's size to PCM's and proportion of DRAM's useful space to PCM's. Lee et al. [24] proposed a memory management algorithm which makes use of the write frequency as well as the recency of write references to accurately estimate future write references. The proposed scheme, which is applied on the parallel hybrid architecture, is compared to the scheme on DRAM-as-Cache architecture and is shown to be better in memory writes reduction. Liu et al. [31] developed variable-level partition schemes on the hybrid main memory to achieve the tradeoff between energy consumption and performance when the memory accesses and variables are given by the data-flow graph. Fu et al. [16] targeted at minimizing the energy consumption of the hybrid memory system while meeting the performance constraint and proposed a parallelism- and proximity-based variable partitioning scheme.

For task scheduling, Tian et al. [57] present a task-scheduling-based technique for addressing the challenges of hybrid DRAM/PCM main memory. They study the problem of task scheduling, assuming that a task should be entirely placed in either PCM bank or DRAM bank. Their approach works for different optimization objectives such as (1) minimizing the energy consumption of hybrid memory for a given PCM and DRAM size and given PCM endurance, (2) minimizing the number of writes to PCM for a given PCM and DRAM size and given threshold on energy consumption, and (3) minimizing PCM size for a given DRAM size, given threshold on energy consumption and PCM endurance.

14.5 Conclusion

In this chapter, emerging and nonvolatile memories and the state-of-the-art technologies in this area are introduced. A classification of different kinds of NVMs are firstly presented. Each NVM is introduced with a brief description to its features. Next, based on different objectives, several optimizations for memory architecture are introduced when NVM is working as on-chip cache and on-chip memory. Finally, we introduce that when NVM is working as the off-chip main memory, it is a widely used method to utilize both NVM and DRAM and combine them as a hybrid memory system. In the hybrid main memory, DRAM can be used either as a cache to the NVM main memory or as one of the memory partitions of the system. For both of the hybrid architecture, we present a series of schemes for performance and energy optimizations.

References

1. International Technology Roadmap for Semiconductors, 2007
2. https://www.semiconportal.com/en/archive/news/news-by-sin/130823-sin-panasonic-reram-production.html
3. http://loto.sourceforge.net/feram/doc/film.xhtml#(4)
4. http://www.alldatasheet.com/datasheet-pdf/pdf/465689/TI1/MSP430.html

5. Ahn J, Choi K (2012) Lower-bits cache for low power STT-RAM caches. In: International symposium on circuits and systems (ISCAS), pp 480–483
6. Chen Y, Wong WF, Li H, Koh CK, Zhang Y, Wen W (2013) On-chip caches built on multilevel spin-transfer torque RAM cells and its optimizations. J Emerg Technol Comput Syst 9(2):16:1–16:22. doi:10.1145/2463585.2463592
7. Chen YT, Cong J, Huang H, Liu B, Liu C, Potkonjak M, Reinman G (2012) Dynamically reconfigurable hybrid cache: an energy-efficient last-level cache design. In: Design, automation test in Europe conference exhibition (DATE), pp 45–50. doi:10.1109/DATE.2012.6176431
8. Cho S, Lee H (2009) Flip-n-write: a simple deterministic technique to improve PRAM write performance, energy and endurance. In: Proceedings of the 42nd annual IEEE/ACM international symposium on microarchitecture, MICRO 42. ACM, pp 347–357
9. Choi JH, Kim SM, Kim C, Park KW, Park KH (2012) Opamp: evaluation framework for optimal page allocation of hybrid main memory architecture. In: Proceedings of the 2012 IEEE 18th international conference on parallel and distributed systems, ICPADS'12. IEEE Computer Society, pp 620–627
10. Dawber M, Rabe KM, Scott JF (2005) Physics of thin-film ferroelectric oxides. Rev Mod Phys 77:1083–1130. doi:10.1103/RevModPhys.77.1083
11. Dhiman G, Ayoub R, Rosing T (2009) PDRAM: a hybrid PRAM and DRAM main memory system. In: Proceedings of the 46th annual design automation conference, DAC'09. ACM, pp 664–469
12. Diao Z, Li Z, Wang S, Ding Y, Panchula A, Chen E, Wang LC, Huai Y (2007) Spin-transfer torque switching in magnetic tunnel junctions and spin-transfer torque random access memory. J Phys 19(16):13
13. Dong X, Wu X, Sun G, Xie Y, Li H, Chen Y (2008) Circuit and microarchitecture evaluation of 3d stacking magnetic RAM (MRAM) as a universal memory replacement. In: Design automation conference (DAC), pp 554–559
14. Dong X, Xu C, Xie Y, Jouppi N (2012) Nvsim: a circuit-level performance, energy, and area model for emerging nonvolatile memory. IEEE Trans Comput-Aided Des Integr Circuits Syst (TCAD) 31(7):994–1007
15. Ferreira AP, Zhou M, Bock S, Childers B, Melhem R, Mossé D (2010) Increasing PCM main memory lifetime. In: Proceedings of the conference on design, automation and test in Europe, DATE'10. European Design and Automation Association, pp 914–919
16. Fu C, Zhao M, Xue CJ, Orailoglu A (2014) Sleep-aware variable partitioning for energy-efficient hybrid PRAM and DRAM main memory. In: Proceedings of the 2014 international symposium on low power electronics and design, ISLPED'14. ACM, pp 75–80
17. Guo X, Ipek E, Soyata T (2010) Resistive computation: avoiding the power wall with low-leakage, STT-MRAM based computing. In: International symposium on computer architecture (ISCA), pp 371–382
18. Inoue IH, Yasuda S, Akinaga H, Takagi H (2008) Nonpolar resistance switching of metal/binary-transition-metal oxides/metal sandwiches: homogeneous/inhomogeneous transition of current distribution. Phys Rev B 77:035,105. doi:10.1103/PhysRevB.77.035105
19. Jog A, Mishra AK, Xu C, Xie Y, Narayanan V, Iyer R, Das CR (2012) Cache revive: architecting volatile STT-RAM caches for enhanced performance in CMPs. In: Design automation conference (DAC), pp 243–252. doi:10.1145/2228360.2228406
20. Jung J, Nakata Y, Yoshimoto M, Kawaguchi H (2013) Energy-efficient spin-transfer torque RAM cache exploiting additional all-zero-data flags. In: International symposium on quality electronic design (ISQED), pp 216–222
21. Kim YB, Lee SR, Lee D, Lee CB, Chang M, Hur JH, Lee MJ, Park GS, Kim CJ, Chung Ui, Yoo IK, Kim K (2011) Bi-layered RRAM with unlimited endurance and extremely uniform switching. In: Symposium on VLSI technology (VLSIT), pp 52–53
22. Lee BC, Ipek E, Mutlu O, Burger D (2009) Architecting phase change memory as a scalable DRAM alternative. In: Proceedings of the 36th annual international symposium on computer architecture (ISCA), pp 2–13

23. Lee BC, Ipek E, Mutlu O, Burger D (2009) Architecting phase change memory as a scalable DRAM alternative. SIGARCH Comput Archit News 37(3):2–13
24. Lee S, Bahn H, Noh SH (2011) Characterizing memory write references for efficient management of hybrid PCM and DRAM memory. In: Proceedings of the 2011 IEEE 19th annual international symposium on modelling, analysis, and simulation of computer and telecommunication systems, MASCOTS'11. IEEE Computer Society, pp 168–175
25. Lee S, Bahn H, Noh SH (2014) Clock-dwf: a write-history-aware page replacement algorithm for hybrid PCM and DRAM memory architectures. IEEE Trans Comput 63(9): 2187–2200
26. Li H, Chen Y (2009) An overview of non-volatile memory technology and the implication for tools and architectures. In: Design, automation test in Europe conference exhibition (DATE), pp 731–736
27. Li Q, Li J, Shi L, Xue CJ, He Y (2012) Mac: migration-aware compilation for STT-RAM based hybrid cache in embedded systems. In: International symposium on low power electronics and design (ISLPED), pp 351–356
28. Li Q, Li J, Shi L, Zhao M, Xue C, He Y (2014) Compiler-assisted STT-RAM-based hybrid cache for energy efficient embedded systems. IEEE Trans Very Large Scale Integr (VLSI) Syst 22(8):1829–1840
29. Li Q, Zhao M, Hu J, Liu Y, He Y, Xue CJ (2015) Compiler directed automatic stack trimming for efficient non-volatile processors. In: Annual design automation conference (DAC), pp 183:1–183:6
30. Li Y, Chen Y, Jones AK (2012) A software approach for combating asymmetries of non-volatile memories. In: International symposium on low power electronics and design (ISLPED), pp 191–196
31. Liu T, Zhao Y, Xue CJ, Li M (2011) Power-aware variable partitioning for dsps with hybrid PRAM and DRAM main memory. In: Proceedings of the 48th design automation conference, DAC'11. ACM, pp 405–410
32. Liu Y, Yang H, Wang Y, Wang C, Sheng X, Li S, Zhang D, Sun Y (2014) Ferroelectric nonvolatile processor design, optimization, and application. In: Xie Y (ed) Emerging memory technologies. Springer New York, pp 289–322. doi:10.1007/978-1-4419-9551-3_11
33. Meza J, Chang J, Yoon H, Mutlu O, Ranganathan P (2012) Enabling efficient and scalable hybrid memories using fine-granularity DRAM cache management. IEEE Comput Archit Lett 11(2):61–64
34. Mittal S, Vetter J, Li D (2014) Lastingnvcache: a technique for improving the lifetime of non-volatile caches. In: IEEE computer society annual symposium on VLSI (ISVLSI), pp 534–540. doi:10.1109/ISVLSI.2014.69
35. Mittal S, Vetter J, Li D (2015) A survey of architectural approaches for managing embedded DRAM and non-volatile on-chip caches. IEEE Trans Parallel Distrib Syst 26(6):1524–1537
36. Mittal S, Vetter JS, Li D (2014) Writesmoothing: improving lifetime of non-volatile caches using intra-set wear-leveling. In: Proceedings of the 24th edition of the Great Lakes symposium on VLSI (GLSVLSI), pp 139–144
37. Papandreou N, Pozidis H, Pantazi A, Sebastian A, Breitwisch M, Lam C, Eleftheriou E (2011) Programming algorithms for multilevel phase-change memory. In: IEEE international symposium on circuits and systems (ISCAS), pp 329–332
38. Park H, Yoo S, Lee S (2011) Power management of hybrid DRAM/PRAM-based main memory. In: Proceedings of the 48th design automation conference, DAC'11. ACM, pp 59–64
39. Park Y, Shin DJ, Park SK, Park KH (2011) Power-aware memory management for hybrid main memory. In: 2011 The 2nd international conference on next generation information technology (ICNIT), pp 82–85
40. Quan B, Zhang T, Chen T, Wu J (2012) Prediction table based management policy for STT-RAM and SRAM hybrid cache. In: International conference on computing and convergence technology (ICCCT), pp 1092–1097

41. Qureshi MK, Karidis J, Franceschini M, Srinivasan V, Lastras L, Abali B (2009) Enhancing lifetime and security of PCM-based main memory with start-gap wear leveling. In: Proceedings of the 42nd annual IEEE/ACM international symposium on microarchitecture, MICRO 42. ACM, pp 14–23
42. Qureshi MK, Srinivasan V, Rivers JA (2009) Scalable high performance main memory system using phase-change memory technology. In: Proceedings of the 36th annual international symposium on computer architecture, ISCA'09. ACM, pp 24–33
43. Ramos LE, Gorbatov E, Bianchini R (2011) Page placement in hybrid memory systems. In: Proceedings of the international conference on supercomputing, ICS'11. ACM, pp 85–95
44. Raoux S, Burr G, Breitwisch M, Rettner C, Chen Y, Shelby R, Salinga M, Krebs D, Chen SH, Lung H, Lam C (2008) Phase-change random access memory: a scalable technology. IBM J Res Dev 52(4.5):465–479. doi:10.1147/rd.524.0465
45. Rasquinha M, Choudhary D, Chatterjee S, Mukhopadhyay S, Yalamanchili S (2010) An energy efficient cache design using spin torque transfer (STT) RAM. In: International symposium on low power electronics and design (ISLPED), pp 389–394
46. Rogers BM, Krishna A, Bell GB, Vu K, Jiang X, Solihin Y (2009) Scaling the bandwidth wall: challenges in and avenues for CMP scaling. In: International symposium on computer architecture (ISCA), pp 371–382
47. Seok H, Park Y, Park KH (2011) Migration based page caching algorithm for a hybrid main memory of DRAM and PRAM. In: Proceedings of the 2011 ACM symposium on applied computing, SAC'11. ACM, pp 595–599
48. Seong NH, Woo DH, Lee HHS (2010) Security refresh: prevent malicious wear-out and increase durability for phase-change memory with dynamically randomized address mapping. SIGARCH Comput Archit News 38(3):383–394
49. Sharifi A, Kandemir M (2011) Automatic feedback control of shared hybrid caches in 3D chip multiprocessors. In: International conference on parallel, distributed and network-based processing (PDP), pp 393–400
50. Sharifi A, Kandemir M (2013) Using cache-coloring to mitigate inter-set write variation in non-volatile caches. In: Iowa State University, Ames, Technical report
51. Shin DJ, Park SK, Kim SM, Park KH (2012) Adaptive page grouping for energy efficiency in hybrid PRAM-DRAM main memory. In: Proceedings of the 2012 ACM research in applied computation symposium, RACS'12. ACM, pp 395–402
52. Smullen C, Mohan V, Nigam A, Gurumurthi S, Stan M (2011) Relaxing non-volatility for fast and energy-efficient STT-RAM caches. In: International symposium on high performance computer architecture (HPCA), pp 50–61
53. Sun G, Dong X, Xie Y, Li J, Chen Y (2009) A novel architecture of the 3D stacked MRAM l2 cache for CMPS. In: International symposium on high performance computer architecture (HPCA), pp 239–249
54. Sun G, Kursun E, Rivers JA, Xie Y (2013) Exploring the vulnerability of CMPS to soft errors with 3D stacked nonvolatile memory. J Emerg Technol Comput Syst 9(3):22:1–22:22. doi:10.1145/2491679
55. Sun Z, Bi X, Li HH, Wong WF, Ong ZL, Zhu X, Wu W (2011) Multi retention level STT-RAM cache designs with a dynamic refresh scheme. In: International symposium on microarchitecture (MICRO), pp 329–338
56. Sun Z, Wu W, Li H (2013) Cross-layer racetrack memory design for ultra high density and low power consumption. In: Design automation conference (DAC), pp 1–6
57. Tian W, Zhao Y, Shi L, Li Q, Li J, Xue CJ, Li M, Chen E (2013) Task allocation on nonvolatile-memory-based hybrid main memory. IEEE Trans Very Large Scale Integr Syst 21(7): 1271–1284
58. Venkatesan R, Kozhikkottu V, Augustine C, Raychowdhury A, Roy K, Raghunathan A (2012) Tapecache: a high density, energy efficient cache based on domain wall memory. In: International symposium on low power electronics and design (ISLPED), pp 185–190
59. Venkatesan R, Kozhikkottu V, Augustine C, Raychowdhury A, Roy K, Raghunathan A (2012) Tapecache: a high density, energy efficient cache based on domain wall memory. In: International symposium on low power electronics and design (ISLPED), pp 185–190

60. Venkatesan R, Sharad M, Roy K, Raghunathan A (2013) DWM-tapestri – an energy efficient all-spin cache using domain wall shift based writes. In: Design, automation & test in Europe conference & exhibition (DATE), pp 1825–1830
61. Wang J, Dong X, Xie Y, Jouppi N (2013) i2wap: improving non-volatile cache lifetime by reducing inter- and intra-set write variations. In: International symposium on high performance computer architecture (HPCA2013), pp 234–245. doi:10.1109/HPCA.2013.6522322
62. Wang Y, Liu Y, Li S, Zhang D, Zhao B, Chiang MF, Yan Y, Sai B, Yang H (2012) A 3us wake-up time nonvolatile processor based on ferroelectric flip-flops. In: Proceedings of the ESSCIRC (ESSCIRC), pp 149–152
63. Wu X, Li J, Zhang L, Speight E, Rajamony R, Xie Y (2009) Hybrid cache architecture with disparate memory technologies. In: Proceedings of the 36th annual international symposium on computer architecture (ISCA), pp 34–45
64. Xu W, Sun H, Wang X, Chen Y, Zhang T (2011) Design of last-level on-chip cache using spin-torque transfer RAM (STT RAM). IEEE Trans Very Large Scale Integr (VLSI) Syst 19(3):483–493
65. Xue CJ, Zhang Y, Chen Y, Sun G, Yang JJ, Li H (2011) Emerging non-volatile memories: opportunities and challenges. In: Proceedings of international conference on hardware/software codesign and system synthesis (CODES+ISSS), pp 325–334
66. Yang BD, Lee JE, Kim JS, Cho J, Lee SY, gon Yu B (2007) A low power phase-change random access memory using a data-comparison write scheme. In: IEEE international symposium on circuits and systems, ISCAS'07, pp 3014–3017
67. Yazdanshenas S, Pirbasti M, Fazeli M, Patooghy A (2014) Coding last level STT-RAM cache for high endurance and low power. Comput Archit Lett 13(2):73–76
68. Yoon H (2012) Row buffer locality aware caching policies for hybrid memories. In: Proceedings of the 2012 IEEE 30th international conference on computer design, ICCD'12. IEEE Computer Society, pp 337–344
69. Yoon H, Meza J, Harding R, Ausavarungnirun R, Mutlu O (2011) Dynrbla: a high-performance and energy-efficient row buffer locality-aware caching policy for hybrid memories. SAFARI Technical Report No. 2011–005
70. Yun J, Lee S, Yoo S (2012) Bloom filter-based dynamic wear leveling for phase-change RAM. In: Proceedings of the conference on design, automation and test in Europe, DATE'12. EDA Consortium, pp 1513–1518
71. Zhang W, Li T (2009) Exploring phase change memory and 3D die-stacking for power/thermal friendly, fast and durable memory architectures. In: Proceedings of the 2009 18th international conference on parallel architectures and compilation techniques, PACT'09. IEEE Computer Society, pp 101–112
72. Zhao M, Jiang L, Shi L, Zhang Y, Xue C (2015) Wear relief for high-density phase change memory through cell morphing considering process variation. IEEE Trans Comput-Aided Des Integr Circuits Syst 34(2):227–237
73. Zhao M, Li Q, Xie M, Liu Y, Hu J, Xue CJ (2015) Software assisted non-volatile register reduction for energy harvesting based cyber-physical system. In: Design, automation & test in Europe conference & exhibition (DATE), pp 567–572
74. Zhao W, Belhaire E, Mistral Q, Chappert C, Javerliac V, Dieny B, Nicolle E (2006) Macromodel of spin-transfer torque based magnetic tunnel junction device for hybrid magnetic-cmos design. In: IEEE international behavioral modeling and simulation workshop, pp 40–43
75. Zhou P, Zhao B, Yang J, Zhang Y (2009) A durable and energy efficient main memory using phase change memory technology. SIGARCH Comput Archit News 37(3):14–23
76. Zhou P, Zhao B, Yang J, Zhang Y (2009) Energy reduction for STT-RAM using early write termination. In: International conference on computer-aided design (ICCAD), pp 264–268
77. Zhu JG (2008) Magnetoresistive random access memory: the path to competitiveness and scalability. Proc IEEE 96(11):1786–1798. doi:10.1109/JPROC.2008.2004313
78. Zwerg M, Baumann A, Kuhn R, Arnold M, Nerlich R, Herzog M, Ledwa R, Sichert C, Rzehak V, Thanigai P, Eversmann BO (2011) An 82μA/MHz microcontroller with embedded feram for energy-harvesting applications. In: International solid-state circuits conference (ISSCC), pp 334–336