Modeling and Leveraging Emerging Non-Volatile Memories for Future Computer Designs

Student: Xiangyu Dong, Supervisor: Yuan Xie
Department of Computer Science and Engineering, Pennsylvania State University
111N Information Science and Technology Building, University Park, PA 16802, USA
{xydong,yuanxie}@cse.psu.edu

Abstract

Many new types of non-volatile memory technologies are now evolving. For example, emerging non-volatile memories such as STTRAM, PCRAM, and ReRAM show their attractive properties of high access performance and low access energy. In this work, we aim to facilitate these emerging non-volatile memory technologies in designing future high-performance and low-power computer systems. We start this work by building circuit-level performance, energy, and area models for STTRAM, PCRAM, and ReRAM. And then, we provide architecture-level techniques that mitigate the drawbacks in non-volatile memory write operations. Finally, we study the application-level cases of adopting these emerging technologies.

I. Program & Motivation

In order to build the next-generation exascale computing systems, disruptive technologies are required to provide solutions of high-performance as well as low-power computing capability. Consider the facts that memory access latency and disk access latency are several orders of magnitudes slower than the processing cores and system memory power (DRAM power) and disk power contribute as much as 40% to the overall power consumption in a data center [1], [2], it is highly necessary to improve the performance and the power profiles of the traditional memory hiearacy.

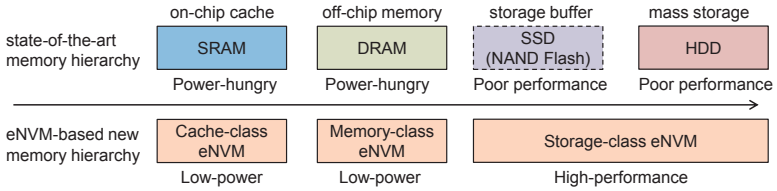

Fig. 1. While the state-of-the-art memory hierarchy includes SRAM, DRAM, NAND flash, and HDD, we propose an eNVM-based new hierarchy that only uses non-volatile memory technology and provides both high performance and low-power operations.

Therefore, the state-of-the-art memory hierarchy, as shown in Fig. 1, is facing several challenges. On the performance side, the HDD speed becomes a severe bottleneck and its mechanical structure limits the upper bound of the access speed. Although the recent adoption of SSD storage raises the performance level, the slow programming speed of NAND flash devices and the write endurance of 10^5 still hamper the SSD storage to be the mainstream storage in the future. On the energy consumption side, the existing DRAM main memory already contributes to a large portion of the total system energy consumption, and the increasing leakage power makes SRAM on-chip caches and DRAM off-chip main memories impractical to scale down to the next fabrication process level. Therefore, disruptive technologies are strongly required to improve the memory hierarchy performance without incurring proportional energy consumption increment.

TABLE I
Characteristics of emerging non-volatile memory (eNVM)

Device type	SRAM	DRAM	NAND flash	HDD	STTRAM	PCRAM	ReRAM
Maturity	Product	Product	Product	Product	Prototype	Prototype	Prototype
Cell size	$150F^2$	$6F^2$	$4F^2$	$(2/3)F^2$	$6F^2$	$4F^2$	$4F^2$
MLC capability	No	No	4 bits/cell	No	2 bits/cell	4 bits/cell	4 bits/cell
Write energy	$1pJ$	$2pJ$	$10nJ$	N/A	$0.02pJ$	$100pJ$	$0.1pJ$
Write latency	$5ns$	$10ns$	$200\mu s$	$10ms$	$20ns$	$100ns$	$20ns$
Write endurance	10^{16}	10^{16}	10^5	N/A	10^{16}	10^8	10^8

In recent years, significant efforts and resources have been put on the researches and developments of emerging memory technologies. Several promising candidates, such as Spin-Torque Transfer RAM (STTRAM) [3], [4], Phase-Change RAM (PCRAM) [5], [6], [8] and Resistive RAM (ReRAM) [9]–[11], have gained substantial attentions and are being actively pursued by both academic and industrial research. In this work, we call them eNVM (emerging non-volatile memory). eNVMs have attractive features such as scalability, fast read access, zero standby power, and non-volatility. In addition, by using different types of peripheral circuitry design, eNVM can cover a wide design space from highly performance-optimized cache to highly area-optimized mass storage. However, as shown in Table I, the drawbacks of eNVMs are their relatively long write latency and high write energy (for PCRAM and ReRAM, the limited write endurance is another issue). However, it is still promising to redesign the memory hierarchy by adopting various styles of eNVMs in different hierarchy levels as demonstrated in Fig. 1. Compared to the state-of-the-art designs,

- The non-volatility nature of eNVMs lowers the memory subsystem energy consumption that were originally contributed to SRAM cache and DRAM main memory;
- The nanosecond-level access speed of eNVMs easily outperforms today's SSD and HDD in the storage level of the hierarchy.

Since eNVM technologies are still premature and they have drawbacks in handling write operations, the challenges of eNVM research includes: (1) on circuit level, estimation tools are required for high-level designs to understand and facilitate the underlying eNVM devices; (2) on architecture level, non-standard techniques are necessary to mitigate the long write latency, high write energy, and limited write endurance of eNVMs; (3) on application level, eNVMs need to demonstrate their effectiveness in easing the future computing system designs.

II. Background & Related Work

STTRAM uses Magnetic Tunnel Junction (MTJ) as the memory storage and leverages the difference in magnetic directions to represent the memory bit. As shown in Figure 2, MTJ contains two ferromagnetic layers. One ferromagnetic layer is has fixed magnetization direction and it is called the reference layer, while the other layer has a free magnetization direction that can be changed by passing a write current and it is called the free layer. The relative magnetization direction of two ferromagnetic layers determines the resistance of MTJ and thus stores the data.

PCRAM uses chalcogenide material (e.g. GST) to store information. The chalcogenide materials can be switched between a crystalline phase an amorphous phase with the application of heat. The crystalline phase shows low resistivity while the amorphous phase is characterized by high resistivity. Figure 3 shows an example of a MOS-accessed PCRAM cell. The SET operation crystallizes GST by heating it above its crystallization temperature, and the RESET operation melt-quenches GST to make the material amorphous. The temperature is controlled by passing a specific electrical current profile and generating the required Joule heat.

ReRAM is another eNVM technology based on electro- and thermochemical effects in the resistance change of a metal/oxide/metal system. Figure 4 illustrates the general concept for the ReRAM working mechanism. An ReRAM cell consists of a metal oxide layer (e.g Ti [9] and Hf [10]) sandwiched by two metal (e.g. Pt [9]) electrodes. The electronic behavior of metal/oxide interfaces depends on the oxygen vacancy concentration of the metal oxide layer. Typically, the metal/oxide interface shows Ohmic in the case of very high doping and rectifying in the case of low doping [9]. The oxygen vacancy in metal oxide is n-type dopant, whose draft under the electric field can cause the change of doping profiles. Thus, applying electronic current can modulate the I-V curve of the ReRAM cell and further switch the cell from one state to the other state.

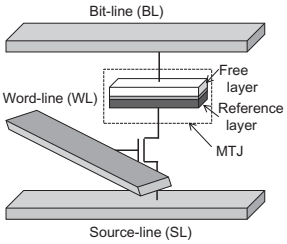

Fig. 2. The conceptual view of an STTRAM cell.

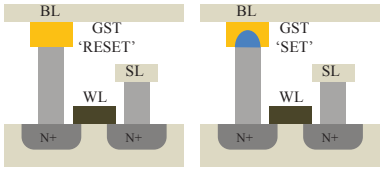

Fig. 3. The conceptual view of a PCRAM cell.

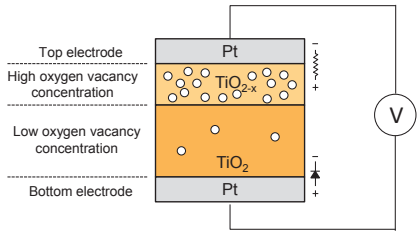

Fig. 4. The working mechanism of an ReRAM cell.

These eNVM technologies share some common properties. For example, the advantages of STTRAM, PCRAM, and ReRAM include fast memory access speed (in several nanoseconds) and byte-accessible capability while legacy non-volatile technologies are extremely slow (e.g. several microseconds for NAND flash and several milliseconds for HDD) and can only be accessed in the unit of pages or blocks. Some previous research leverages these advantages to design high-performance file systems [12] and storage device [13]. However, the major disadvantages of these emerging memory technologies are the asymmetric read/write latency and energy consumption (for all of these technologies), as well as the limited write endurance (except STTRAM). Early write termination [14] and write pause [15] are proposed to mitigate the performance penalty caused by long write latency. In addition, several architecture-level techniques, such as ECP [16], dynamically replicated memory [17], SAFER [18], and FREE-p [19], are designed to improve the memory module lifetime under the limited write endurance condition.

III. Approach & Uniqueness

We study the feasibility of leveraging eNVM in future computer designs from three aspects. We start the research from the circuit-level modeling so that an in-house eNVM performance, energy, and area estimation tool is available throughout the rest parts of the research. Then, we focus on architecture-level techniques to mitigate the eNVM drawbacks on long latency, high energy, and limited endurance of write operations. Finally, we conduct an application-level case study on exascale fault-resilience problem to demonstrate how eNVM is critical in the future computer system designs.

A. Circuit-Level eNVM Modeling

As the ultimate goal of this eNVM research is to devise a universal memory hierarchy as shown in Fig. 1, each of these eNVM technologies has to supply a wide design space that covers a spectrum from highly latency-optimized microprocessor caches to highly density-optimized secondary storage. Therefore, specialized peripheral circuitry is required for each optimization target. However, since few of these eNVM technologies are mature so far, only a limited number of prototype chips have been demonstrated and just cover a small portion of the entire design space. Therefore, circuit-level estimation models are necessary to facilitate the eNVM research by predicting the eNVM performance, energy consumption, and chip area without heavy efforts in building prototypes.

Hence, as the first step of this eNVM research, we develop *NVSim*, a circuit-level model for eNVM performance, energy, and area estimations. As an extension of CACTI [20], NVSim supports not only SRAM and DRAM, but also STTRAM, PCRAM, and ReRAM.

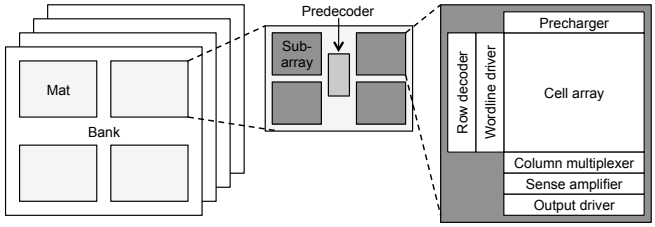

Fig. 5. The memory array organization modeled in NVSim: a hierarchical memory organization includes banks, mats, and subarrays with decoders, multiplexers, sense amplifiers, and output drivers.

Fig. 5 shows the array organization. There are 3 hierarchy levels in such organization, which are *bank*, *mat*, and *subarray*.

Compared to CACTI, the new features of NVSim includes support of external sensing and current sensing, flexible array organizations and data activation modes, multiple design options of buffers, capability of modeling cross-point memory cells, and extensible user interface.

NVSim is implemented in C++ from scratch, and it contains more than 20,000 lines of source code. We use it for our later architecture- and application-level studies. The NVSim binary code can be downloaded at http://www.cse.psu.edu/~xydong/files/nvsim_binary_v2.tar.bz2.

B. Architecture-Level Techniques for Alleviating eNVM Write Overhead

Compared to volatile memories such as SRAM and DRAM, eNVMs have more stable data keeping mechanism. Accordingly, it needs to take a longer time and consume more energy to overwrite the existing data. Hence, the major drawbacks of eNVM are the relatively long write latency and high write energy. In addition, the switching mechanisms of PCRAM and ReRAM cause cell wear-out and lead to limited write endurance. Since these drawbacks are the intrinsic device characteristics of eNVMs, techniques on higher levels are needed to alleviate these issues. In this work, we propose three techniques for alleviating eNVM write overhead. For these studies, NVSim is used to get the eNVM cache parameters such as access latency, access energy, and occupied chip area. Upon circuit-level eNVM data, full-system Simics simulations are run to predict the architecture-level effectiveness of each proposed technique.

Read-Preemptive Write Buffer: Write buffer is a common technique to hide long write latency. However, the normal write buffer is not sufficiently effective to hide the extra long latency of eNVM write operations because multiple read operations can be blocked by an ongoing write operation and thus drastically harm the overall performance. To solve this problem, we design a priority policy to grant read operations the rights of canceling an ongoing write operation in some conditions. The scheduling rules of such buffer are as follows,

- *Rule 1:* The read operation always has the higher priority in a competition for the execution right;
- *Rule 2:* When a read request is blocked by a write retirement and the write buffer is not full, the read request can trap and stall the write retirement if the progress of that write retirement is less than 50% as shown in Fig. 6. Then, the read operation is granted the execution right. The cancelled write retirement will be retried later.

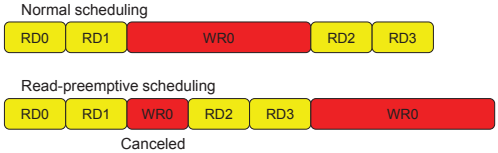

Fig. 6. In a read-preemptive write buffer, the read operations can be granted over the ongoing write operation if the progress of that write operation is less than 50%.

Hybrid SRAM-eNVM Cache: The aforementioned read-preemptive write buffer hides the eNVM long write latency, but the total number of write operations remains the same. In order to reduce the number of write operations to eNVM cache lines, we propose hybrid SRAM-eNVM cache as well. The proposed hybrid cache implementation is that, instead of building

a pure eNVM cache, we compose the ways in each cache set with a majority of eNVM cache lines and a minority of SRAM ones. The main purpose is to keep as many write intensive data in the SRAM part as possible and hence reduce the number of write operations to the eNVM part. The management policy of the hybrid SRAM-eNVM cache can be described as follows,

- The cache controller is aware of the locations of SRAM cache ways and eNVM cache ways. When there is a write miss, the cache controller first try to place the data in the SRAM cache ways.
- Data in eNVM lines will be migrated to SRAM lines when they are accessed by two successive write operations.

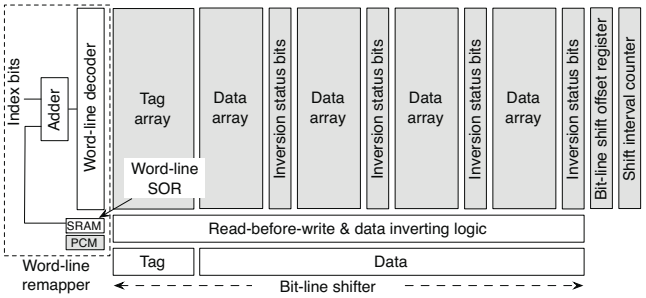

Fig. 7. Block diagram of the suggested data inverting cache architecture for eNVM lifetime improvement.

Data Inverting: Besides long write latency and high write energy consumption, the finite write endurance is another major challenge for PCRAM and ReRAM implementation. To improve their lifetime, we design a data inverting architecture, where a read-before-write logic, a data inverting logic, a bitline shifter, and a word-line remapper are integrated with the eNVM tag and data arrays as illustrated in Fig. 7. The amount of writing bits are reduced by two factors,

- *Read-before-write:* Directly writing data into eNVM cells wastes energy and the number of writes allowed for an eNVM cell because it mostly performs redundant writes, i.e., the new value written to an eNVM cell is the same to its previous value.
- *Data inverting:* When we write a new data value to a cache line, we first read its current data value, and compute the Hamming distance between the two values. If the calculated distance is larger than the half of the cache line size, we invert the new data value before storing it.

Since this technique reduces the number of writes, it reduces the write energy consumption as well.

With the help of these proposed techniques, we later show that the write overhead of eNVM is alleviated.

C. Application-Level eNVM Case Study on Exascale Fault-Resilience

One of the system-level applications that we have considered for eNVMs is to enhance the HDD-based checkpointing/rollback scheme, which is one of the most common approaches to ensure the fault-resilience of a computing system. In the current petascale computing systems, the HDD-based checkpointing already incurs a large performance overhead and is not a scalable solution for the future exascale computing. To solve this problem, we investigate how to use eNVM technologies to provide a low performance penalty fault-resilience scheme.

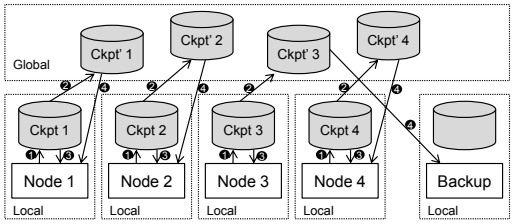

Fig. 8. The local/global hybrid checkpoint model.

Our proposed eNVM checkpointing scheme is a hybrid one with both local and global checkpoints. Using such scheme, every node has a dedicated local eNVM storage that periodically backs up the state of each node in their own private storage coordinately. In addition, a global checkpoint can be made from an existing local checkpoint. Fig. 8 shows the conceptual view of the hybrid checkpoint scheme and the detailed checkpointing steps are as follows,

- *Step 1:* Each node dumps the memory image to their own local eNVM checkpoints;
- *Step 2:* After several local checkpoint interval, a global HDD checkpoint is initiated, and the new global checkpoints are made from the latest local eNVM checkpoints;
- *Step 3:* When there is a failure but all the local eNVM checkpoints are accessible, the local eNVM checkpoints are loaded to restore the computation;
- *Step 4:* When there is a failure and parts of the local eNVM checkpoints are lost (in this case, Node 3 is lost), the global HDD checkpoints are loaded, and the failure node is substituted by a backup node.

This design takes the advantage of eNVM fast access speed to boost the local checkpointing performance and the low cost HDD storage to provide the full fault coverage.

IV. Results & Contributions

A. NVSim Validation

The NVSim performance, energy, and area estimation tool is validated against several eNVM industrial prototypes. The validation results against STTRAM [4], PCRAM [5], and ReRAM [11] are listed in Table II, Table III, and Table IV, respectively. The NVSim validation results show a reasonable alignment to its modeled targets.

TABLE II
STTRAM MODEL VALIDATION WITH RESPECT TO A $65nm$ $64Mb$ STTRAM PROTOTYPE CHIP [4]

Metric	Actual	Projected
Area	$39.1mm^2$	$33.2mm^2$
Read latency	$11ns$	$9.9ns$
Write latency	$25ns$	$15.3ns$

TABLE III
PCRAM MODEL VALIDATION WITH RESPECT TO A $0.12\mu m$ $64Mb$ MOS-ACCESSED PCRAM PROTOTYPE CHIP [5]

Metric	Actual	Projected
Area	$64mm^2$	$60.9mm^2$
Read latency	$70.0ns$	$65.9ns$
Write latency	$> 180.0ns$	$180.6ns$

TABLE IV
RERAM MODEL VALIDATION WITH RESPECT TO A $0.18\mu m$ $4Mb$ MOSFET-SELECTED RERAM PROTOTYPE CHIP [11]

Metric	Actual	Projected
Area	-	$33.42mm^2$
Read latency	$7.2ns$	$7.72ns$
Write latency	$< 7.2ns$	$6.56ns$

B. Effectiveness of Architecture-Level Techniques

To evaluate the effectiveness of the proposed read-preemptive write buffer and the SRAM-eNVM hybrid cache techniques, we use 8MB STTRAM to replace a 2MB SRAM L2 cache in a 4-core chip-multiprocessor example. The high eNVM cell density makes 8MB STTRAM caches have comparable silicon area with 2MB SRAM caches, and thus have the advantage in larger cache capacity. However, without any techniques, directly replacing SRAM caches by STTRAM ones causes an overall performance loss. In contrary, the adoption of our two proposed techniques improves the overall performance by 4.9% and reduces the total energy by 73.5%.

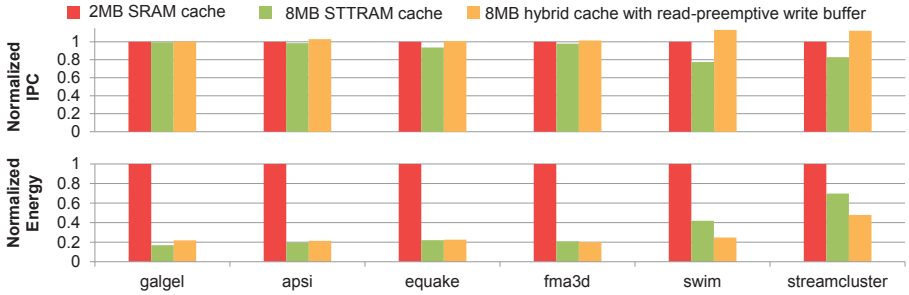

Fig. 9. The performance improvement and energy reduction after applying the read-preemptive write buffer and hybrid cache techniques.

In addition, by properly using data inverting writes in PCRAM cache designs, we could further achieve 8% of energy saving and 3.8 years of lifetime with 17% of chpi area overhead compared with the baseline designs.

C. Instant Checkpointing for Exascale Computing

To assess how the eNVM can be the disruptive technology that solves the high performance overhead problem caused by today's HDD-based checkpointing, we first derive a local/global hybrid checkpointing performance model, and using this model we estimate the checkpointing overhead from a current petaFLOPS system to a future projected exaFLOPS system. Fig. 10 shows the performance overhead increment trend for HDD-based checkpointing and eNVM-based hybrid checkpointing.

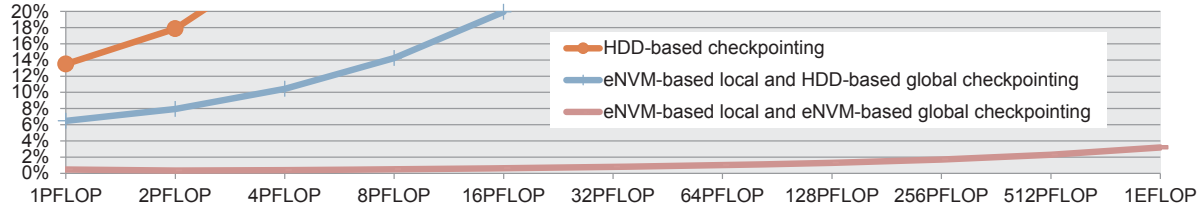

Fig. 10. The performance overhead of eNVM-based checkpointing ranging from petascale systems to exascale systems (normalized to the computation time).

The results show that the slow HDD checkpointing has trouble scaling beyond 1 petaFLOPS without taking a heavy hit in performance. The introduction of local eNVM checkpointing helps scale beyond 5 petaFLOPS, the poor scaling of HDD bandwidth hampers the benefit beyond 20 petaFLOPS. The use of eNVM for both local and global checkpoints eventually raises the bar to a exascale system.

D. Contribution

Several technical conference and journal papers have been published during the progress of this research [21]–[30]. We believe that such research will initiate a novel research direction in high-performance system design and investigate the impact of eNVM technologies on future computing systems. If success, this research will support the deployment of modern microprocessor designs that use eNVM technologies.

REFERENCES

[1] D. Roberts, T. Kgil, and T. Mudge, "Using non-volatile memory to save energy in servers," in *Proceedings of the Design, Automation & Test in Europe*, 2009, pp. 743–748.

[2] K. Lim, J. Chang, T. Mudge *et al.*, "Disaggregated memory for expansion and sharing in blade servers," in *Proceedings of the International Symposium on Computer Architecture*, 2009, pp. 267–278.

[3] T. Kawahara, R. Takemura, K. Miura *et al.*, "2Mb spin-transfer torque RAM (SPRAM) with bit-by-bit bidirectional current write and parallelizing-direction current read," in *Proceedings of the IEEE International Solid-State Circuits Conference*, 2007, pp. 480–617.

[4] K. Tsuchida, T. Inaba, K. Fujita *et al.*, "A 64Mb MRAM with clamped-reference and adequate-reference schemes," in *Proceedings of the International Solid-State Circuits Conference*, 2010, pp. 268–269.

[5] S. J. Ahn, Y. J. Song, C. W. Jeong *et al.*, "Highly manufacturable high density phase change memory of 64Mb and beyond," in *Proceedings of the International Electron Devices Meeting*, 2004, pp. 907–910.

[6] N. Matsuzaki, K. Kurotsuchi, Y. Matsui *et al.*, "Oxygen-doped GeSbTe phase-change memory cells featuring 1.5V/100μA standard 0.13μm CMOS operations," in *Proceedings of the IEEE International Electron Devices Meeting*, 2005, pp. 738–741.

[7] S. Hanzawa, N. Kitai, K. Osada *et al.*, "A 512kB embedded phase change memory with 416kB/s write throughput at 100μA cell write current," in *Proceedings of the International Solid-State Circuits Conference*, 2007, pp. 474–616.

[8] J. J. Yang, M. D. Pickett, X. Li *et al.*, "Memristive switching mechanism for metal/oxide/metal nanodevices," *Nature Nanotechnology*, vol. 3, no. 7, pp. 429–433, 2008.

[9] Y. S. Chen, H. Y. Lee, P. S. Chen *et al.*, "Highly scalable hafnium oxide memory with improvements of resistive distribution and read disturb immunity," in *Proceedings of the International Electron Devices Meeting*, 2009, pp. 105–108.

[10] S.-S. Sheu, M.-F. Chang, K.-F. Lin *et al.*, "A 4Mb embedded SLC resistive-RAM macro with 7.2ns read-write random-access time and 160ns MLC-access capability," in *Proceedings of the IEEE International Solid-State Circuits Conference*, 2011, pp. 200–201.

[11] J. Condit, E. B. Nightingale, C. Frost *et al.*, "Better I/O through byte-addressable, persistent memory," in *Proceedings of the Symposium on Operating Systems Principles*, 2009, pp. 133–146.

[12] A. M. Caulfield, A. De, J. Coburn *et al.*, "Moneta: A high-performance storage array architecture for next-generation, non-volatile memories," in *Proceedings of the International Symposium on Microarchitecture*, 2010, pp. 385–395.

[13] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "Energy reduction for STT-RAM using early write termination," in *Proceedings of the International Conference on Computer-Aided Design*, 2009, pp. 264–268.

[14] M. K. Qureshi, M. Franceschini, and L. Lastras, "Improving read performance of phase change memories via write cancellation and write pausing," in *Proceedings of the International Symposium on High Performance Computer Architecture*, 2010, pp. 1–11.

[15] S. Schechter, G. H. Loh, K. Straus, and D. Burger, "Use ECP, not ECC, for hard failures in resistive memories," in *Proceedings of the International Symposium on Computer Architecture*, 2010, pp. 141–152.

[16] E. Ipek, J. Condit, E. B. Nightingale *et al.*, "Dynamically replicated memory: building reliable systems from nanoscale resistive memories," in *Proceedings of the Architectural Support for Programming Languages and Operating Systems*, 2010, pp. 3–14.

[17] N. H. Seong, D. H. Woo, V. Srinivasan *et al.*, "SAFER: Stuck-at-fault error recovery for memories," in *Proceedings of the International Symposium on Microarchitecture*, 2010, pp. 115–124.

[18] D. H. Yoon, N. Muralimanohar, J. Chang *et al.*, "FREE-p: Protecting non-volatile memory against both hard and soft errors," in *Proceedings of the International Symposium on High Performance Computer Architecture*, 2011, pp. 466–477.

[19] S. Thoziyoor, N. Muralimanohar, J.-H. Ahn, and N. P. Jouppi, "CACTI 5.1 technical report," HP Labs, Tech. Rep. HPL-2008-20, 2008.

[20] X. Dong, X. Wu, G. Sun *et al.*, "Circuit and microarchitecture evaluation of 3D stacking magnetic RAM (MRAM) as a universal memory replacement," in *Proceedings of the Design Automation Conference*, 2008, pp. 554–559.

[21] G. Sun, X. Dong, Y. Xie *et al.*, "A novel 3D stacked MRAM cache architecture for CMPs," in *Proceedings of the International Symposium on High-Performance Computer Architecture*, 2009, pp. 239–249.

[22] X. Dong, N. P. Jouppi, and Y. Xie, "PCRAMsim: System-level performance, energy, and area modeling for phase-change RAM," in *Proceedings of the International Conference on Computer-Aided Design*, 2009, pp. 269–275.

[23] X. Dong, N. Muralimanohar, N. P. Jouppi *et al.*, "Leveraging 3D PCRAM technologies to reduce checkpoint overhead for future exascale systems," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2009, pp. 57.1–57.12.

[24] D. Niu, Y. Chen, X. Dong, and Y. Xie, "Energy and performance driven circuit design for emerging phase-change memory," in *Proceedings of the Asia and South Pacific Design Automation Conference*, 2010, pp. 193–198.

[25] Y. Joo, D. Niu, X. Dong *et al.*, "Energy- and endurance-aware design of phase change memory caches," in *Proceedings of the Design, Automation & Test in Europe*, 2010, pp. 136–141.

[26] X. Dong and Y. Xie, "AdaMS: Adaptive MLC/SLC phase-change memory design for file storage," in *Proceedings of the Asia and South Pacific Design Automation Conference*, 2011, pp. 31–36.

[27] C. Xu, X. Dong, N. P. Jouppi, and Y. Xie, "Design implications of memristor-based RRAM cross-point structures," in *Proceedings of the Design, Automation & Test in Europe*, 2011, pp. 1–6.

[28] X. Dong, X. Wu, Y. Xie *et al.*, "Stacking MRAM atop microprocessors: An architecture-level evaluation," *IET Computers & Digital Techniques*, vol. -, no. -, pp. –, 2011.

[29] X. Dong, Y. Xie, N. Muralimanohar, and N. P. Jouppi, "Hybrid checkpointing using emerging non-volatile memories for future exascale systems," *ACM Transactions on Architecture and Code Optimization*, vol. -, no. -, pp. –, 2011.