Guest Editor Introduction: Recent Advances in DRAM and Flash Memory Architectures

Onur Mutlu[1,2] Saugata Ghose[2] Rachata Ausavarungnirun[2]

[1]*ETH Zürich* [2]*Carnegie Mellon University*

Memory and storage systems are a fundamental system performance, energy, and reliability bottleneck in modern systems [1, 2, 3, 34, 35, 36]. This bottleneck is becoming increasingly severe due to (1) the very limited latency reductions in memory and storage devices over the last several years; (2) aggressive manufacturing process technology scaling and other techniques to improve memory density, such as multi-level cell technology, which increase the storage capacity of these devices, but introduce more raw bit errors and increase manufacturing process variation; (3) limited pin counts in chip packages, which prevent system designers from adding more and/or wider buses to increase bandwidth; (4) overwhelmingly data-intensive applications, which require high-bandwidth access to very large amounts of data; and (5) the increasing fraction of overall system energy consumed by memory systems and data movement. To make matters worse, it is becoming increasingly difficult to continue scaling these devices to smaller process technology nodes, and even though alternative emerging memory and storage technologies can potentially alleviate some of the shortcomings of existing memory and storage technologies, they also introduce *new* shortcomings that were previously absent. Therefore, there is a pressing need to comprehensively understand and mitigate these bottlenecks in both existing and emerging memory and storage systems and technologies.

This issue features extended summaries and retrospectives of some of the recent research done by our group, SAFARI [41, 43], on (1) understanding, characterizing, and modeling various critical properties of modern DRAM and NAND flash memory, the dominant memory and storage technologies, respectively; and (2) several new mechanisms we have proposed based on our observations from these analyses, characterization, and modeling, to tackle various key challenges in memory and storage scaling. In order to understand the sources of various bottlenecks of the dominant memory and storage technologies, these works perform rigorous studies of device-level and application-level behavior, using a combination of detailed simulation and experimental characterization of *real* memory and storage devices.

The works that perform real device characterization make use of custom FPGA-based platforms that we build to provide us with fine-grained control over the devices. We devise specific tests that perform a controlled measurement of each phenomenon that we aim to explore. Our experimental characterizations have often discovered many unexpected types of behavior in real state-of-the-art devices, and have inspired the research community to pursue further investigations (e.g., on the RowHammer phenomenon [21, 34], DRAM retention behavior [20, 30, 39], NAND flash memory error patterns [1, 2, 3, 5, 6, 7, 8, 9, 11]). In order to aid future research, we have released much of our experimental characterization data online [43, 45], and have open-sourced our DRAM characterization platform, SoftMC [19, 44].

The works that perform application and architectural analyses rely on real system characterizations and simulation to develop a rigorous understanding of the bottlenecks and to provide solutions. Our analyses have shown key scaling bottlenecks, proposed new solutions, and have inspired the research community to develop further investigations (e.g., on DRAM refresh [12, 30, 31], DRAM latency reduction [28, 29], the RowHammer phenomenon [21, 34], and in-memory data movement and computation [16, 47, 49, 50]). In order to aid future research, we have released our flexible and extensible memory system simulator, Ramulator, as open-source software [22, 42].

In each work that is featured in this issue, based on our observations and analyses from our experimental studies of real systems and applications as well as future trends and problems, we propose novel solutions that overcome many of the scaling bottlenecks that memory and storage systems face. For each of the works presented in this special issue, its corresponding article examines the work's significance in the context of modern computer systems, and discusses several new research questions and directions that each work motivates.

We start with five of our works that explore new opportunities in DRAM systems to reduce latency and/or energy consumption. As we mentioned earlier, the latency and energy consumption of DRAM have not reduced significantly in the last several years. We find that by introducing heterogeneity into DRAM architectures, or by taking advantage of the existing variation within and across DRAM modules, we can develop new mechanisms that improve DRAM access latency and/or energy efficiency.

The first paper in the issue describes Tiered-Latency DRAM (TL-DRAM), which originally appeared in HPCA 2013 [29]. This work (1) proposes a new DRAM architecture that can provide us with the performance benefits of costly reduced-latency DRAM products in a cost-effective manner, by isolating a small portion of a DRAM array so that it can behave as a low-latency DRAM buffer; and (2) exploits the low-latency in-DRAM buffer using various hardware or software mechanisms to improve overall system performance.

The second paper in the issue describes Adaptive-Latency DRAM (AL-DRAM), which originally appeared in HPCA 2015 [28]. This work experimentally characterizes (1) the large latency variation across DRAM modules and (2) the large timing margins designed to account for worst-case variation and operating conditions. Based on the findings from the characterization, the work proposes a new mechanism that can identify and safely reduce the timing margin to speed up DRAM accesses, and thus improve overall system performance and energy consumption.

The third paper in the issue describes Flexible-Latency DRAM (FLY-DRAM), which originally appeared in SIGMETRICS 2016 [15]. This work experimentally characterizes the latency variation that exists *within* each DRAM module, showing that there are regions of fast cells and regions of slow cells that exist in real DRAM modules. Based on these findings, the work proposes a new mechanism that identifies regions of fast cells and reduces the latency of DRAM operations to these regions.

The fourth paper in the issue describes Voltron, which originally appeared in SIGMETRICS 2017 [13]. This work experimentally characterizes the relationship between DRAM latency, reliability, and supply voltage, showing that these three can be traded off intelligently for various purposes. The work proposes a new mechanism that uses this relationship to dynamically reduce DRAM energy consumption within a bounded performance loss target.

The fifth paper in the issue describes SoftMC, which originally appeared in HPCA 2017 [19]. This work describes our open-source DRAM characterization infrastructure, and demonstrates its versatility for use in a wide range of DRAM research topics. SoftMC is a result of 6+ years of effort, which led to at least 11 works at top conferences, and we hope it will enable other researchers to explore the detailed behavior of existing and emerging memory architectures and develop new mechanisms and memory architectures.

Next, we look at a couple of our works that reduce data movement between the CPU and DRAM, as this movement consumes (1) a large fraction of DRAM energy and (2) much of the limited available DRAM bandwidth. We find that a large portion of DRAM bandwidth is consumed by the movement of data between DRAM and the CPU to perform simple operations such as data copy and initialization. We can instead take advantage of the underlying DRAM architecture to efficiently perform these simple operations directly within DRAM, eliminating the need to move the data to/from the CPU.

The sixth paper in the issue describes RowClone, which originally appeared in MICRO 2013 [50]. Many applications perform data copy and initialization operations, requiring only simple computation, but these operations require expensive data movement between the CPU and DRAM. This work proposes a new DRAM architecture that can internally perform bulk data copy and initialization operations at very low hardware cost, avoiding the costly data movement, and shows that doing so provides 1–2 orders of magnitude speedup and energy reduction for such operations.

The seventh paper in the issue describes low-cost interlinked subarrays (LISA), which originally appeared in HPCA 2016 [16]. This work (1) builds a general substrate that facilitates the bulk movement of data between two different rows in memory by improving the interconnectivity of DRAM arrays, and (2) demonstrates that LISA can be used to efficiently implement a number of mechanisms, such as bulk data copy/initialization, latency reduction, and fast in-DRAM caching. Each of these mechanisms provides significant performance and energy improvements.

Finally, we investigate the reliability of NAND flash memory. As NAND flash memory based solid-state drives (SSDs) are now widely-used in a large variety of modern systems (e.g., data centers [33,38,46], smartphones), there is continued demand to increase the density of SSDs while lowering the cost per bit. While manufacturers have employed several methods (e.g., aggressive manufacturing process technology scaling and multi-level cell technology), these methods have exacerbated a number of sources of raw bit errors. Due to limitations to the number of errors that can be corrected by error-correcting codes (ECC), SSDs have a limited lifetime, after which manufacturers cannot reliably retain data for a minimum guaranteed time without data loss [1, 2, 3]. Over the last decade, as a result of aggressive density scaling, the typical lifetime of an SSD has dropped by 1–2 orders of magnitude, and the various sources of raw bit errors now pose a key scaling challenge for storage [1,2,3]. As a sampling of our 7+ years of research into NAND flash memory reliability, we feature three papers that design mechanisms to significantly mitigate reliability issues and extend the limited lifetime of NAND flash memory based devices.

The eighth paper in the issue describes a new data retention study in NAND flash memory, which originally appeared in HPCA 2015 [7]. This work experimentally characterizes the susceptibility of state-of-the-art NAND flash memory to data retention errors using our FPGA-based flash memory testing infrastructure [1, 2, 3, 5], and proposes (1) a new mechanism that mitigates the impact of retention errors at runtime, which increases the lifetime of the SSD; and (2) a new mechanism that exploits retention behavior to recover data in the event of data loss, thereby improving SSD robustness.

The ninth paper in the issue describes a new read disturb study in NAND flash memory, which originally appeared in DSN 2015 [6]. This work experimentally characterizes read disturb errors in NAND flash memory, where a read operation introduces errors in unread parts of the memory. Based on the characterization, the work proposes (1) a new mechanism that mitigates read disturb errors, thereby improving the SSD lifetime; and (2) a new mechanism that exploits read disturb behavior to recover data in the event of data loss, thereby improving SSD robustness.

The last paper in the issue describes a new study on two-step programming in NAND flash memory, which originally appeared in HPCA 2017 [4]. This work demonstrates that the programming algorithm used in many state-of-the-art NAND flash memory devices can introduce previously-unknown data vulnerabilities, which can be exploited by malicious applications to perform security attacks. The work proposes three mechanisms to eliminate or mitigate these vulnerabilities, thereby improving both reliability and security.

Throughout all of these works, we find that by understanding and taking advantage of the behavior and architecture of memory and storage devices and appropriately modifying them at low cost and low overhead, we can successfully mitigate many of the scalability challenges in memory and storage devices. Even though the works presented are described in the context of DRAM and NAND flash memory, the two dominant memory and storage technologies of today, we believe many of the basic ideas and concepts can be applied or adapted to emerging memory technologies [32], e.g., phase-change memory [24, 25, 26, 40, 53, 54, 55], STT-MRAM [18, 23, 37], and memristors/RRAM [17, 51, 52]. We hope that the works featured in this special issue inspire readers to explore the presented challenges, and to develop new solutions that can enable high-performance, low-energy, low-latency, high-reliability memory and storage systems, and thus the computing systems, of the future.

Acknowledgments

The works featured in this issue, along with our related works that we reference in each featured work, are a result of the research done together with many students and collaborators over the course of the past 10+ years, whose contributions we acknowledge. In particular, we acknowledge and appreciate the dedicated effort of current and former students and postdocs in our research group, SAFARI [41, 43], who contributed to the ten featured works, including Yu Cai, Kevin Chang, Chris Fallin, Hasan Hassan, Kevin Hsieh, Ben Jaiyen, Abhijith Kashyap, Samira Khan, Yoongu Kim, Tianshi Li, Jamie Liu, Donghyuk Lee, Yixin Luo, Justin Meza, Gennady Pekhimenko, Vivek Seshadri, Lavanya Subramanian, Nandita Vijaykumar, and Abdullah Giray Yağlıkçı.

Aside from our featured works and other referenced papers from our group, where a wealth of information on modern memory and storage systems can be found, at least four Ph.D. dissertations have shaped the works that we feature in this special issue:

- Yu Cai's thesis entitled "NAND Flash Memory: Characterization, Analysis, Modeling and Mechanisms" [10],
- Donghyuk Lee's thesis entitled "Reducing DRAM Latency at Low Cost by Exploiting Heterogeneity" [27],
- Vivek Seshadri's thesis entitled "Simple DRAM and Virtual Memory Abstractions to Enable Highly Efficient Memory Subsystems" [48], and
- Kevin Chang's thesis entitled "Understanding and Improving the Latency of DRAM-Based Memory Systems" [14].

We also acknowledge various funding agencies (the National Science Foundation, the Semiconductor Research Corporation, the Intel Science and Technology Center on Cloud Computing, CyLab, the CMU Data Storage Systems Center, and the NIH) and industrial partners (AMD, Facebook, Google, HP Labs, Huawei, IBM, Intel, Microsoft, NVIDIA, Oracle, Qualcomm, Rambus, Samsung, Seagate, VMware), and ETH Zürich, who have supported the featured works in this issue and other related work in our research group generously over the years.

References

[1] Y. Cai, S. Ghose, E. F. Haratsch, Y. Luo, and O. Mutlu, "Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives," *Proc. IEEE*, 2017.

[2] Y. Cai, S. Ghose, E. F. Haratsch, Y. Luo, and O. Mutlu, "Error Characterization, Mitigation, and Recovery in Flash Memory Based Solid-State Drives," arXiv:1706.08642 [cs.AR], 2017.

[3] Y. Cai, S. Ghose, E. F. Haratsch, Y. Luo, and O. Mutlu, "Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery," arXiv:1711.11427 [cs.AR], 2017.

[4] Y. Cai, S. Ghose, Y. Luo, K. Mai, O. Mutlu, and E. F. Haratsch, "Vulnerabilities in MLC NAND Flash Memory Programming: Experimental Analysis, Exploits, and Mitigation Techniques," in *HPCA*, 2017.

[5] Y. Cai, E. F. Haratsch, O. Mutlu, and K. Mai, "Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis," in *DATE*, 2012.

[6] Y. Cai, Y. Luo, S. Ghose, E. F. Haratsch, K. Mai, and O. Mutlu, "Read Disturb Errors in MLC NAND Flash Memory: Characterization, Mitigation, and Recovery," in *DSN*, 2015.

[7] Y. Cai, Y. Luo, E. F. Haratsch, K. Mai, and O. Mutlu, "Data Retention in MLC NAND Flash Memory: Characterization, Optimization, and Recovery," in *HPCA*, 2015.

[8] Y. Cai, O. Mutlu, E. F. Haratsch, and K. Mai, "Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation," in *ICCD*, 2013.

[9] Y. Cai, G. Yalcin, O. Mutlu, E. F. Haratsch, A. Cristal, O. Unsal, and K. Mai, "Flash Correct and Refresh: Retention Aware Management for Increased Lifetime," in *ICCD*, 2012.

[10] Y. Cai, "NAND Flash Memory: Characterization, Analysis, Modelling, and Mechanisms," Ph.D. dissertation, Carnegie Mellon Univ., 2012.

[11] Y. Cai, E. F. Haratsch, O. Mutlu, and K. Mai, "Threshold Voltage Distribution in MLC NAND Flash Memory: Characterization, Analysis, and Modeling," in *DATE*, 2013.

[12] K. K. Chang, D. Lee, Z. Chishti, A. R. Alameldeen, C. Wilkerson, Y. Kim, and O. Mutlu, "Improving DRAM Performance by Parallelizing Refreshes with Accesses," in *HPCA*, 2014.

[13] K. K. Chang, A. G. Yağlıkçı, S. Ghose, A. Agrawal, N. Chatterjee, A. Kashyap, D. Lee, M. O'Connor, H. Hassan, and O. Mutlu, "Understanding Reduced-Voltage Operation in Modern DRAM Devices: Experimental Characterization, Analysis, and Mechanisms," in *SIGMETRICS*, 2017.

[14] K. K. Chang, "Understanding and Improving the Latency of DRAM-Based Memory Systems," Ph.D. dissertation, Carnegie Mellon Univ., 2017.

[15] K. K. Chang, A. Kashyap, H. Hassan, S. Ghose, K. Hsieh, D. Lee, T. Li, G. Pekhimenko, S. Khan, and O. Mutlu, "Understanding Latency Variation in Modern DRAM Chips: Experimental Characterization, Analysis, and Optimization," in *SIGMETRICS*, 2016.

[16] K. K. Chang, P. J. Nair, D. Lee, S. Ghose, M. K. Qureshi, and O. Mutlu, "Low-Cost Inter-Linked Subarrays (LISA): Enabling Fast Inter-Subarray Data Movement in DRAM," in *HPCA*, 2016.

[17] L. Chua, "Memristor—The Missing Circuit Element," *TCT*, 1971.

[18] X. Guo, E. Ipek, and T. Soyata, "Resistive Computation: Avoiding the Power Wall with Low-Leakage, STT-MRAM Based Computing," in *ISCA*, 2010.

[19] H. Hassan, N. Vijaykumar, S. Khan, S. Ghose, K. K. Chang, G. Pekhimenko, D. Lee, O. Ergin, and O. Mutlu, "SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies," in *HPCA*, 2017.

[20] S. Khan, D. Lee, Y. Kim, A. Alameldeen, C. Wilkerson, and O. Mutlu, "The Efficacy of Error Mitigation Techniques for DRAM Retention Failures: A Comparative Experimental Study," in *SIGMETRICS*, 2014.

[21] Y. Kim, R. Daly, J. Kim, C. Fallin, J. H. Lee, D. Lee, C. Wilkerson, K. Lai, and O. Mutlu, "Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors," in *ISCA*, 2014.

[22] Y. Kim, W. Yang, and O. Mutlu, "Ramulator: A Fast and Extensible DRAM Simulator," *CAL*, 2015.

[23] E. Kültürsay, M. Kandemir, A. Sivasubramaniam, and O. Mutlu, "Evaluating STT-RAM as an Energy-Efficient Main Memory Alternative," in *ISPASS*, 2013.

[24] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger, "Architecting Phase Change Memory as a Scalable DRAM Alternative," in *ISCA*, 2009.

[25] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger, "Phase Change Memory Architecture and the Quest for Scalability," *CACM*, 2010.

[26] B. C. Lee, P. Zhou, J. Yang, Y. Zhang, B. Zhao, E. Ipek, O. Mutlu, and D. Burger, "Phase-Change Technology and the Future of Main Memory," *IEEE Micro*, 2010.

[27] D. Lee, "Reducing DRAM Latency at Low Cost by Exploiting Heterogeneity," Ph.D. dissertation, Carnegie Mellon Univ., 2016.

[28] D. Lee, Y. Kim, G. Pekhimenko, S. Khan, V. Seshadri, K. Chang, and O. Mutlu, "Adaptive-Latency DRAM: Optimizing DRAM Timing for the Common-Case," in *HPCA*, 2015.

[29] D. Lee, Y. Kim, V. Seshadri, J. Liu, L. Subramanian, and O. Mutlu, "Tiered-Latency DRAM: A Low Latency and Low Cost DRAM Architecture," in *HPCA*, 2013.

[30] J. Liu, B. Jaiyen, Y. Kim, C. Wilkerson, and O. Mutlu, "An Experimental Study of Data Retention Behavior in Modern DRAM Devices: Implications for Retention Time Profiling Mechanisms," in *ISCA*, 2013.

[31] J. Liu, B. Jaiyen, R. Veras, and O. Mutlu, "RAIDR: Retention-Aware Intelligent DRAM Refresh," in *ISCA*, 2012.

[32] J. Meza, Y. Luo, S. Khan, J. Zhao, Y. Xie, and O. Mutlu, "A Case for Efficient Hardware-Software Cooperative Management of Storage and Memory," in *WEED*, 2013.

[33] J. Meza, Q. Wu, S. Kumar, and O. Mutlu, "A Large-Scale Study of Flash Memory Errors in the Field," in *SIGMETRICS*, 2015.

[34] O. Mutlu, "The RowHammer Problem and Other Issues We May Face as Memory Becomes Denser," in *DATE*, 2017.

[35] O. Mutlu, "Memory Scaling: A Systems Architecture Perspective," in *IMW*, 2013.

[36] O. Mutlu and L. Subramanian, "Research Problems and Opportunities in Memory Systems," *SUPERFRI*, 2014.

[37] H. Naeimi, C. Augustine, A. Raychowdhury, S.-L. Lu, and J. Tschanz, "STT-RAM Scaling and Retention Failure," *Intel Technology Journal*, 2013.

[38] I. Narayanan, D. Wang, M. Jeon, B. Sharma, L. Caulfield, A. Sivasubramaniam, B. Cutler, J. Liu, B. Khessib, and K. Vaid, "SSD Failures in Datacenters: What? When? and Why?" in *SYSTOR*, 2016.

[39] M. Qureshi, D. H. Kim, S. Khan, P. Nair, and O. Mutlu, "AVATAR: A Variable-Retention-Time (VRT) Aware Refresh for DRAM Systems," in *DSN*, 2015.

[40] M. K. Qureshi, V. Srinivasan, and J. A. Rivers, "Scalable High Performance Main Memory System Using Phase-Change Memory Technology," in *ISCA*, 2009.

[41] SAFARI Research Group, http://www.ece.cmu.edu/~safari/.

[42] SAFARI Research Group, "Ramulator – GitHub Repository," https://github.com/CMU-SAFARI/ramulator.

[43] SAFARI Research Group, "SAFARI Software Tools – GitHub Repository," https://github.com/CMU-SAFARI/.

[44] SAFARI Research Group, "SoftMC – GitHub Repository," https://github.com/CMU-SAFARI/SoftMC.

[45] SAFARI Research Group, "Tools, Software, and Full Data Sets," http://www.ece.cmu.edu/~safari/tools.html.

[46] B. Schroeder, R. Lagisetty, and A. Merchant, "Flash Reliability in Production: The Expected and the Unexpected," in *FAST*, 2016.

[47] V. Seshadri, D. Lee, T. Mullins, H. Hassan, A. Boroumand, J. Kim, M. A. Kozuch, O. Mutlu, P. B. Gibbons, and T. C. Mowry, "Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology," in *MICRO*, 2017.

[48] V. Seshadri, "Simple DRAM and Virtual Memory Abstractions to Enable Highly Efficient Memory Systems," Ph.D. dissertation, Carnegie Mellon Univ., 2016.

[49] V. Seshadri, K. Hsieh, A. Boroumand, D. Lee, M. A. Kozuch, O. Mutlu, P. B. Gibbons, and T. C. Mowry, "Fast Bulk Bitwise AND and OR in DRAM," *CAL*, 2015.

[50] V. Seshadri, Y. Kim, C. Fallin, D. Lee, R. Ausavarungnirun, G. Pekhimenko, Y. Luo, O. Mutlu, M. A. Kozuch, P. B. Gibbons, and T. C. Mowry, "RowClone: Fast and Energy-Efficient In-DRAM Bulk Data Copy and Initialization," in *MICRO*, 2013.

[51] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The Missing Memristor Found," *Nature*, 2008.

[52] H.-S. P. Wong, H.-Y. Lee, S. Yu, Y.-S. Chen, Y. Wu, P.-S. Chen, B. Lee, F. T. Chen, and M.-J. Tsai, "Metal-Oxide RRAM," *Proc. IEEE*, 2012.

[53] H.-S. P. Wong, S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi, and K. E. Goodson, "Phase Change Memory," *Proc. IEEE*, 2010.

[54] H. Yoon, J. Meza, N. Muralimanohar, N. P. Jouppi, and O. Mutlu, "Efficient Data Mapping and Buffering Techniques for Multi-Level Cell Phase-Change Memories," *TACO*, 2014.

[55] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "A Durable and Energy Efficient Main Memory Using Phase Change Memory Technology," in *ISCA*, 2009.