STT-RAM Memory Hierarchy Designs Aimed to Performance, Reliability and Energy Consumption

Carlos Escuin*[1], Teresa Monreal†,
José M. Llabería†, Víctor Viñals*,
Pablo Ibáñez*

* *Universidad de Zaragoza*
† *Universitat Politècnica de Catalunya, BarcelonaTech*

ABSTRACT

Current applications demand larger on-chip memory capacity since off-chip memory accesses become a bottleneck. However, if we want to achieve this by scaling down the transistor size of SRAM-based Last-Level Caches (LLCs) it may become prohibitive in terms of cost, area and energy. Therefore, other technologies such as STT-RAM are becoming real alternatives to build the LLC in multicore systems.

Although STT-RAM bitcells feature high density and low static power, they suffer from other trade-offs. On the one hand, STT-RAM writes are more expensive than STT-RAM reads and SRAM writes. In order to address this asymmetry, we will propose microarchitectural techniques to minimize the number of write operations on STT-RAM cells.

On the other hand, reliability also plays an important role. STT-RAM cells suffer from three types of errors: write, read disturbance, and retention errors. Regarding this, we will suggest techniques to manage redundant information allowing error detection and information recovery.

KEYWORDS: Memory Hierarchy, Last-Level Caches, Non-Volatile Memories, STT-RAM

1 Introduction

The off-chip memory accesses are the main problem in the execution of applications in a multicore system. Besides, the number of cores in these systems keeps on increasing. To feed the cores in an efficient way, it is needed the memory subsystem to offer a reduced latency and a great bandwidth. One way to achieve this is to have more on-chip capacity enlarging the Last Level Cache (LLC).

However, the bitcells that conform the LLC are traditionally built with SRAM technology which has two main limitations: low density and high static power. Therefore, keeping on enlarging the LLC may become prohibitive in terms of cost, area and energy. The emerging non-volatile technologies (NVM) begin to be an alternative for building LLCs. These

[1]E-mail: escuin@unizar.es

memories are characterized by their high density and low static power.

The main NVMs drawback is that writes spoil the materials, limiting their lifetime. However, among all the NVM technologies, the STT-RAM is the one that get harmed the less by writes, becoming a good alternative to build LLCs [NIS+14].

2 Background

The STT-RAM bitcell use a *Magnetic Tunnel Junction* (MTJ) to enable information storage. A MTJ is formed by an insulating layer (MgO) sandwiched between two layers of ferromagnetic material (free and reference layers). The reference layer has a magnetic field fixed in a given direction. The magnetic field of the free layer can change depending on the logical value stored ('0' or '1').

Despite the energy savings the STT-RAM technologies offer, their potential gets reduced due to new trade-offs. For example, the asymmetry between the read and write operations. Writes are more expensive in terms of latency and energy consumption than reads. This is because in write operations the voltage has to be kept constant for a period of time in order to update the cell magnetic field. Moreover, STT-RAM writes are more expensive than SRAM write. Therefore, this asymmetry creates new challenges in the cache hierarchy design.

Due to this asymmetry, many techniques are proposed to minimize the number of write operations in the STT-RAM cells. With the goal of reducing the number of written blocks in the LLC, Ahn et al. / Rodríguez-Rodríguez et al. propose a (dead-block / reuse) predictor along with a bypass policy to avoid the write operation of (dead / non-reused) predicted blocks [AYC14, RRDC+17].

Choi et al. proposes a hybrid LLC that combines ways made up of SRAM cells with STT-RAM ways [CP17]. Moreover, way selection is done by assigning ways to cores with two main goals: 1) reducing cache misses, and 2) reducing write operations on STT-RAM cells.

Korgaonkar et al. suggest to monitor the write operations occupation in the LLC request queue [KBL+18]. In order to control the LLC write congestion, first, they suggest a bypass policy based in the detection of potentially-alive write operations. They also propose relaxing the LLC exclusivity to avoid redundant write-backs. Besides, some of the LLC capacity gets reserved for the write intensive blocks.

Another important cache design challenge with STT-RAM technologies is related to reliability. STT-RAM bitcells are exposed to three kind of errors, namely:

1. **Write errors**. During the write operation the voltage has to remain constant during a period of time that is, statistically, large enough to change the cell value. A write error happens when this time has been too short to success in the update. That is, the cell ends up in a state whose read operation returns the wrong value. Besides, the probability that the write operation fails in the '0' $\rightarrow$ '1' transition is two orders of magnitude greater than the '1' $\rightarrow$ '0' one [BSLW12]. Error correction codes (ECC) can deal with this kind of errors. However, usage of ECC codes implies an overhead in terms of area and energy. With the goal of minimizing it, Wang et al. [WME+16] propose a LLC with some ways having more protection than others. Each block is assigned to a different way depending on the number of 1s it contains; being the number of 1s an upper bound of the number of '0' $\rightarrow$ '1' transitions. The more the number of 1s of the incoming block, the greater the protection of the way where it is allocated.

2. **Retention errors**. With process technology scaling, the non-volatility of the STT-RAM cells gets compromised. That is, the cells start to fail in retaining information when their size is reduced meaningfully [GBGI18]. Idle cells randomly experience loss of information in a non foreseeable way.

3. **Read Disturbance Errors** (RDEs). A read operation may accidentally change the cell content. One solution to mitigate RDEs consists of performing a restore operation after every read operation. However, this implies that the every read operation has a write operation overhead. With the objective of minimizing the number of restore operations after every read, Mittal et al. suggest storing two copies of the same block using compression [MVJ17]. In this way, the first restore operation is avoided for every block.

3 Objectives

From what has been said above, it is clear that including STT-RAM technology in the memory hierarchy has room for boosting their advantages and reducing their limitations. New microarchitectural techniques targeting performance, reliability and energy consumption can contribute to foster market adoption. We will work in monocore and multicore environments and propose how to build new hybrid caches, as well as new replacement, insertion, prefetching, and bypass mechanisms that address the previously exposed trade-offs.

Specifically, we will consider three well differentiated contexts where STT-RAM technology takes importance to build cache memories. In each of these contexts, we will try to find new tradeoffs balances that make us to attain different goals.

The **first** one is a context where STT-RAM cells are relatively large. The main limitation here is the large write operation cost in terms of time and energy. So the goal is to propose novel techniques that minimize the number of write operations on STT-RAM cells. We will work in STT-RAM and SRAM hybrid cache designs, as well as in content management mechanisms. For instance, most of the previous works apply their proposals on top of a conventional replacement algorithm (LRU). We suggest analyzing the use of specific replacement algorithms for non-inclusive LLCs in order to see which synergies show up among the new write operations distribution, contents management and replacement.

The **second** context is related to small STT-RAM cells, which suffer from information retention problems. The proposed techniques will take alternative paths for managing the redundant information that allows error detection and information recovery.

Finally, after the achievement of new mechanisms for the previous contexts, we will consider their application in the *non-volatile processors* (NVP) field that have been proposed for IoT devices without battery [SML+17]. The NVPs are fed by intermittent power sources which implies frequent and unexpected power supply shutdowns. To mitigate the drawbacks of these halts, NVP designs are based on non-volatile flip-flops and memories focusing on the backup/recovery strategies. Some proposals in the NVP field use STT-RAM structures and, recently, L1 caches have been proposed using this technology [XZP+16, SZZ+19].

References

[AYC14] J. Ahn, S. Yoo, and K. Choi. Dasca: Dead write prediction assisted stt-ram cache architecture. In *International Symposium on High Performance Computer Architec-*

ture (HPCA), 2014.

[BSLW12] X. Bi, Z. Sun, H. Li, and W. Wu. Probabilistic design methodology to improve run-time stability and performance of stt-ram caches. In *International Conference on Computer-Aided Design*, 2012.

[CP17] J. Choi and G. Park. Nvm way allocation scheme to reduce nvm writes for hybrid cache architecture in chip-multiprocessors. *IEEE Transactions on Parallel and Distributed Systems*, 28(10):2896–2910, Oct 2017.

[GBGI18] X. Guo, M. N. Bojnordi, Q. Guo, and E. Ipek. Sanitizer: Mitigating the impact of expensive ecc checks on stt-mram based main memories. *IEEE Transactions on Computers*, 67(6):847–860, June 2018.

[KBL+18] K. Korgaonkar, I. Bhati, H. Liu, J. Gaur, S. Manipatruni, S. Subramoney, T. Karnik, S. Swanson, I. Young, and H. Wang. Density tradeoffs of non-volatile memory as a replacement for sram based last level cache. In *International Symposium on Computer Architecture (ISCA)*, 2018.

[MVJ17] S. Mittal, J. S. Vetter, and L. Jiang. Addressing read-disturbance issue in stt-ram by data compression and selective duplication. *IEEE Computer Architecture Letters*, 2017.

[NIS+14] H. Noguchi, K. Ikegami, N. Shimomura, T. Tetsufumi, J. Ito, and S. Fujita. Highly reliable and low-power nonvolatile cache memory with advanced perpendicular stt-mram for high-performance cpu. In *2014 Symposium on VLSI Circuits Digest of Technical Papers*, pages 1–2, June 2014.

[RRDC+17] R Rodríguez-Rodríguez, J Díaz, F Castro, P Ibáñez, D Chaver, V Viñals, J C Saez, M Prieto-Matias, L Piñuel, T Monreal, and J M Llabería. Reuse Detector: Improving the Management of STT-RAM SLLCs. *The Computer Journal*, 2017.

[SML+17] F. Su, K. Ma, X. Li, T. Wu, Y. Liu, and V. Narayanan. Nonvolatile processors: Why is it trending? In *Proceedings of the Conference on Design, Automation & Test in Europe*, 2017.

[SZZ+19] W. Song, Y. Zhou, M. Zhao, L. Ju, C. J. Xue, and Z. Jia. Emc: Energy-aware morphable cache design for non-volatile processors. *IEEE Transactions on Computers*, 2019.

[WME+16] X. Wang, M. Mao, E. Eken, W. Wen, H. Li, and Y. Chen. Sliding basket: An adaptive ecc scheme for runtime write failure suppression of stt-ram cache. In *2016 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 762–767, March 2016.

[XZP+16] M. Xie, M. Zhao, C. Pan, H. Li, Y. Liu, Y. Zhang, C. J. Xue, and J. Hu. Checkpoint aware hybrid cache architecture for nv processor in energy harvesting powered systems. In *International Conference on Hardware/Software Codesign and System Synthesis*, 2016.