# Low-Current Probabilistic Writes for Power-Efficient STT-RAM Caches

Nikolaos Strikos*, Vasileios Kontorinis*, Xiangyu Dong†, Houman Homayoun‡, Dean Tullsen*
*University of California, San Diego †Qualcomm ‡George Mason University
e-mail: *{nstrikos, vkontori, tullsen}@cs.ucsd.edu †xydong@cse.psu.edu ‡hhomayou@gmu.edu

***Abstract*—MRAM has emerged as one of the most attractive non-volatile solutions due to fast read access, low leakage power, high bit density, and long endurance. However, the high power consumption of write operations remains a barrier to the commercial adoption of MRAM technology. This paper addresses this problem by introducing low-current probabilistic writes (LCPW), a technique that reduces write access energy by lowering the amplitude of the write current pulse. Although low current pulses no longer guarantee successful bit write operations, we propose and evaluate a simple technique to ensure correctness and achieve significant power reduction over a typical MRAM implementation.**

## I. Introduction

Cache memories, while relatively energy efficient on a per-transistor level, have always been significant consumers of power, due primarily to their large size and transistor count. A previous study [7] shows that leakage energy can account for 30% of L1 cache energy and as much as 80% of L2 cache energy. As static leakage power becomes a more critical component of dissipated power, the relative power devoted to caches will continue to increase, particularly in large last-level caches.

New advancements in non-volatile memory technology have made them competitive not just with DRAM, but also with SRAM [9], [2]. This is particularly true for large last-level caches where static leakage is high – replacing SRAM with non-volatile memory is attractive because it virtually eliminates the leakage energy consumed by memory cells. Recent advances have reduced the read latency to the point where it is reasonable for this level of cache [3]. Also, large write buffers can hide much of the write latency, especially for last-level caches which are almost always off the write critical path, and in this work we exploit this tolerance for long write delays. However, the high write energy remains one of the most significant barriers to use MRAM as a cache.

In MRAM, high write energy is due in large part to the use of current pulses that cannot nevertheless guarantee successful write of each bit. This paper introduces the concept of low-current probabilistic writes for MRAM caches. By using a current pulse that is allowed to fail with higher probability, we can retry the failed writes and still save significant overall energy. With minor design modifications, we can achieve a 10% reduction in cache write power, and as much as a 9.2% reduction in overall cache power for a write-heavy workload.

## II. Background

Magnetic RAM (MRAM) is one of the more mature technologies among the emerging non-volatile memory options. The basic storage element in MRAM is a magnetic tunnel junction (MTJ), which is composed of two ferromagnetic layers; one has a fixed magnetization direction and the other has a free one, whose orientation relative to the fixed one is used to represent a digital "0" or "1". MRAM cell size is much smaller than SRAM cell size [12], while it can also scale down to sizes smaller than DRAM cells in 10nm processes and below [6]. Finally, high endurance of the order of $10^{15}$ can be expected in the near future [15]. These advances render MRAM a good candidate technology for future high-capacity caches.

One of the key design parameters for MRAM cells is the *critical switching current* ($I_{C0}$), which is the minimum amount of current required to flip the MRAM cell state. $I_{C0}$ is much higher than the write current required by SRAM and MRAM, resulting in larger energy consumption and limited cell scaling due to the need for large NMOS transistors. Worse, even a large write current density cannot fully guarantee the success of a write operation. The relationship between the MRAM switching probability and the switching current has been extensively studied [5] and for the so-called thermal activation regime ($t_{sw} > 10\text{ns}$) can be presented as follows:

$$p_{sw} = 1 - \exp\left\{-\frac{t_{sw}}{\tau_0}\exp\left[-\Delta\left(1 - \frac{I}{I_{C0}}\right)\right]\right\} \qquad (1)$$

where $p_{sw}$ is the switching probability, $I$ is the switching current and $t_{sw}$ is the switching pulse duration. Other parameters in Equation 1 are the relaxation time $\tau_0$, the thermal stability factor $\Delta$ and the critical switching current $I_{C0}$ that causes a spin flip in the absence of any external magnetic fields. Equation 1 implies that:

- Even with a sufficiently high write current, it is impossible to achieve 100% bit write success rate;
- Using lower write current, the bit write success rate degrades gradually as shown in Figure 1.

Later, we show how we can leverage the second property to implement an energy-efficient write scheme for MRAM caches.

## III. Low-Current Probabilistic Writes

### A. Overview

In this work, we propose a probabilistic write scheme for MRAM caches in which the MRAM write operation contains multiple write pulses with lower write current (i.e. less than $I_{C0}$) instead of one write pulse with a much larger write current (i.e. sufficiently greater than $I_{C0}$ to guarantee a near-100% write success rate). We call this new scheme LCPW (Low-Current Probabilistic Writes).

The rationale of LCPW can be seen in Figure 1, which illustrates the relationship between *write current* and *bit write success rate* for an MRAM cell [5]. It is clear that lowering the write current below $I_{C0}$ eliminates the guarantee that all write attempts will successfully flip the MRAM cell and introduces considerable randomness in the cell write process. However, this also implies a quadratic decrease in cell write energy, which has the potential to significantly reduce overall cache write power.

To address the randomness introduced by low success rates, we propose a technique that aims to ensure correctness for any point of operation on the energy-BER curve. We employ an iterative scheme that for every block write: (a) performs a low-current write operation; (b) reads back the contents of the written MRAM cells; and (c) compares the written and the original values to determine whether the write succeeded[1]. If any cell failed to store the written value, the process starts over.

To reduce overhead due to redundant writes, we assume a write mechanism that can identify and bypass correctly written bits with bit granularity. Initially, the stored bits are read from the array of MRAM cells and compared with the head of the write buffer to form the initial write mask. We refer to set bits in the initial bitmask as the *non-identical* bits. If the mask is non-zero, the first write operation attempts a low-current write for non-identical bits. On completion, the written bits are read back and compared again with the write buffer, updating the write mask, which now indicates which bits should be written in the second iteration of low-current writes. The process repeats until all non-identical bits have been successfully written to the device cells. For simplicity, our analysis and simulation methodology assume that every subsequent bit write operation occurs with the same latency and power overhead as the first operation.

To reduce the impact of long write latencies on performance, we use a store buffer that pools pending writes until they complete, as has been proposed elsewhere ([10]). In this work we model non-preemptive reads and writes, but, to ensure consistency, we assume that the write buffer is accessed in parallel with the tag array and can serve cache read and write accesses to pending blocks. We omit write buffer operations from power calculations since, by use of CACTI, we have determined that any reasonably sized write buffer only consumes a negligible fraction of overall cache power.

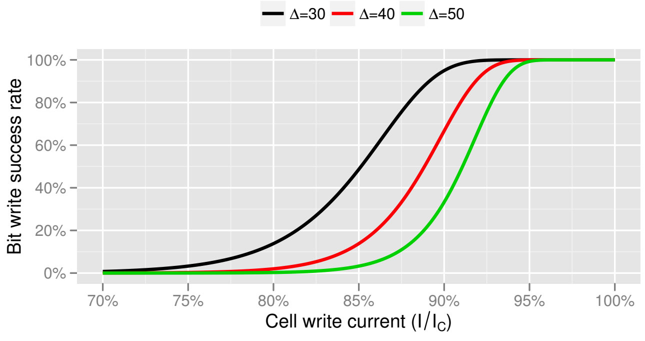

Fig. 1. Bit write success rate vs. write current for $\Delta = 30, 40, 50$.

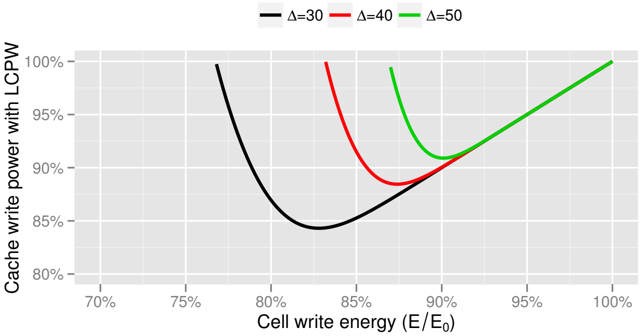

Fig. 2. Estimated reduction in cache write power vs. cell write energy scaling for an MRAM cell with $\Delta = 30, 40, 50$ and $t_{sw} = 60$ns. For $\Delta = 30$, maximum write power reduction is expected for $E/E_0 \approx 83\%$, while scaling below 77% results in power loss due to an excessive number of write attempts.

## B. Analysis

In this part we derive simple and accurate predictions of cache power reduction due to LCPW. For the rest of this section, we assume that the write current is scaled by $I/I_{C0}$, which results in cell switching probability $p_{sw}$ given by Equation 1 and relative cell write energy $e = E_{LC}/E_0$, where $E_{LC}$ and $E_0$ is the low-current and critical current ($I_{C0}$) cell write energy, respectively.

The outcome of a write operation can be modeled by a Bernoulli trial with probability of success $p_{sw}$. For non-identical bits, a sequence of Bernoulli trials will determine the number of write attempts required before the bit is successfully written. In this Bernoulli sequence, the number of failures that precede the first success follows the so-called geometric distribution, whose mean value is $1/p_{sw}$. For example, a bit-error rate of 20% will require on average 1.25 low-current iterations to successfully write one bit. For multiple write attempts, each with energy $E_{LC}$, the average energy required by a successful bit write is $\frac{1}{p}E_{LC}$. Compared to the baseline bit write energy $E_0$, our technique will result in more power-efficient bit writes if $\frac{1}{p_{sw}}E_{LC} < E_0$. In terms of the relative write power $e$ and the bit-write success rate $p_{sw}$, the above condition can be written as:

$$\frac{e}{p_{sw}} < 1 \tag{2}$$

which implies that the expected reduction in write power is simply:

$$\text{LCPW}_{\text{eff}} = 1 - \frac{e}{p_{sw}} \tag{3}$$

Although the above discussion refers to single-bit write operations, it is easy to see that the same conclusions hold

[1] Note that such a write-read-verify scheme is common in MRAM designs since a theoretical 100% write success rate can never be reached.

for an arbitrary block of $n$ bits. Assuming only $m <= n$ non-identical bits, the write operation can be modeled by $m$ independent Bernoulli sequences, all with mean $1/p_{sw}$. The average write energy for the case of an $n$-bit block with $m$ non-identical bits will be $mE_0$ for the baseline case and $m\frac{1}{p_{sw}}E_{LC}$ for our technique, which implies the same energy reduction as Equation 2. Note that this is independent of $n$ or $m/n$, therefore the *percent* reduction in write power is independent of cache block size or ratio of non-identical bits. The result is summarized in Figure 2, which displays the expected reduction in write power for various values of $e$.

This research focuses on MRAM technology, but these principles may apply to other NV memory technologies where write success varies with current over some reasonable range. However, MRAM has one key advantage – all writes fail or succeed, according to a probability, never leaving the cell in a transitional state that cannot be read reliably [5].

### *C. MRAM cell design*

Figure 2 implies that our technique can yield increasing benefits as the value of the thermal stability factor $\Delta$ is decreased. $\Delta$ depends on the physical properties of the MTJ cell and the temperature; for this analysis we consider a fixed temperature $T = 80°\mathrm{C}$. [2] However, lower $\Delta$ means that spontaneous MTJ free layer flips due to thermal energy are more probable. This behavior is quantified by the *retention time* $t_{ret} = e^{\Delta}$, which is the time after which the MTJ will have flipped with probability $1 - 1/e \approx 63\%$. Indeed, setting $I = 0$ in Equation 1 gives the thermal flip CDF:

$$p_{sw}^{thermal} = 1 - \exp\left(-\frac{t_{sw}}{\tau_0}e^{-\Delta}\right) = 1 - \exp\left(-\frac{t_{sw}}{t_{ret}}\right) \qquad (4)$$

Reasonable reliability assumptions would require very high values of $\Delta$. For example, an 8MB cache with rate of failure 1000 FIT [3] at 80°C would require $\Delta > 70$. Fortunately, use of ECC protection and refresh mechanisms can provide the same level of reliability at much lower $\Delta$. Following Smullen [8], the minimum $\Delta$ required by an $N \times m$-bit STT-RAM memory with $k$-bit ECC protection, refresh interval $t_{\text{ref}}$ and failure rate $F$ at room temperature is given by:

$$\Delta \geq \frac{1}{1+k}\left[\sum_{i=0}^{k} \log\frac{m-i}{1+i} + \log\frac{N}{\tau_0 F} + k\log\frac{t_{\text{ref}}}{\tau_0}\right] \qquad (5)$$

For simplicity, in this work we assume a refresh mechanism with very large refresh interval of $t_{\text{ref}} = 10$ sec and hence negligible refresh power. Using Equation 5 we determine a lower value of $\Delta = 46$. The cell and memory design parameters we used are listed in Table II.

## IV. Evaluation

This section demonstrates the benefits of LCPW. First, we describe our experimental methodology. Then, we develop intuition by applying LCPW to single-threaded workloads.

| | |
|---|---|
| Issue, Commit width | 4 |
| INT instruction queue | 16 entries |
| FP instruction queue | 16 entries |
| Reorder Buffer entries | 64 entries |
| INT registers | 64 |
| FP registers | 64 |
| Functional units | 4 int/ldst 2 fp |
| L1 cache | 32KB, 4-way assoc, 1 cyc access |
| L2 cache (priv) | 512KB, 8-way assoc, 3 cyc access |
| L3 cache (shared) | (See Table II) |
| Main memory | 250 cyc access |
| Frequency | 1GHz |

TABLE I. Baseline configuration

MTJ cell

| | |
|---|---|
| Write current ($I$) | 100$\mu$A |
| Write pulse duration ($t_{sw}$) | 60ns |
| Cell size | 90x40nm$^2$ |
| Critical write current ($J_0$) | 2.78MA/cm$^2$ |
| Failure rate ($F$) | 1000 FIT |
| Temperature ($T$) | 80°C |
| Refresh interval ($t_{\text{ref}}$) | 10sec |
| Thermal stability ($\Delta$) | 46 |
| $J/J_0$ | 94.38% |
| Bit-write success rate ($p_{sw}$) | 98.89% |
| Relative write energy ($e$) | 89.08% |

Cache array

| | |
|---|---|
| Size | 8MB |
| Banks | 8 |
| Associativity | 16 |
| Read latency | 8 cycles |
| Write latency | 63 cycles |
| Read energy | 1.078nJ |
| Write energy | 46.983nJ |
| Leakage/bank | 0.710mW |
| Expected LCPW write power | 90.00% |

TABLE II. MRAM design parameters

In order to evaluate LCPW, we modify the SMTSIM simulator [13]. We model a single-core system with a write-back, no-write-allocate 3-level hybrid cache hierarchy consisting of private SRAM L2 caches and an 8MB shared MRAM L3 cache. Our baseline configuration is shown in Table I. We evaluate our technique using the SPEC2K and SPEC2006 benchmark suites. For each application, we fast-forward for 5 billion instructions to skip the initialization phase, warm up the system for 50 million instructions and then simulate for 1 billion instructions.

We derive the MRAM cache parameters from the NVSIM [4] simulator. Our power model uses a methodology similar to [1]. To model cache power, we assume constant energy per read or write access and constant leakage power. For bit-level write operations we also assume that the write energy is equally distributed to all bits of the cache block. The read energy is derived by scaling the read energy per access with the total number of reads, while the total write energy is the write energy per bit written scaled by the total number of bits written. The bit-write success rate depends on the write current according to Figure 1 and a cell flip is modeled as the outcome of a Bernoulli trial, evaluated using the built-in random number generator. Table II shows the design parameters and configuration details of our MRAM model.

Figure 3 shows the results of our simulations for single-core benchmarks. As we expected, the write power is reduced by 10%. The result is consistent across all benchmarks, because

[2] The benefits also increase as $t_{sw}$ is increased, but we consider the switching pulse duration a given parameter of the system.

[3] 1 FIT = 1 device failure in $10^9$ hours of continuous operation

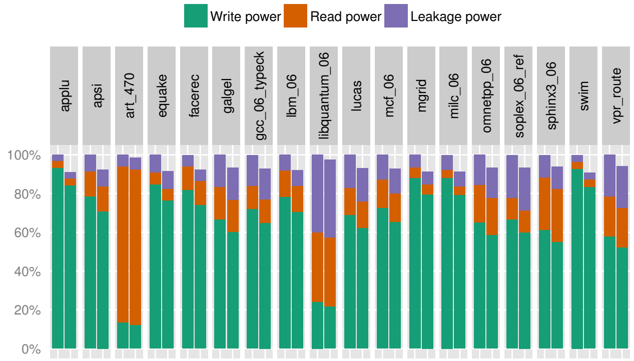

Fig. 3. Breakdown of MRAM L3 cache power, for baseline (left bar in each group) and LCPW (right). Benchmarks that exhibit significant number of writes in L3 benefit most from our approach.

it depends only on the write failure rate. The reduction in total L3 cache power (which depends on the ratio of writes to reads) ranges from almost 0% for benchmarks that exhibit very low write activity in L3, like *art* and *libquantum*, to about 9% for benchmarks like *applu* and *swim* that experience a large number of L3 misses and generate a significant amount of write operations, including cache fills. The average improvement in total L3 power for the single-core benchmarks is 6.9%.

Finally, we note that additional experiments show that a 4-entry buffer is sufficient to hide all delays associated with iterative writes. This is much smaller than, for example, the 20-entry buffer suggested in other work for an MRAM L2 cache [10].

## V. Related Work

In recent years, several works have tried to improve the write properties of STT-RAM. In [10], Sun et al. compare the performance and power consumption of a conventional SRAM L2 cache to a 3D stacked MRAM integration and propose SRAM-MRAM hybrid caches and a read-preemptive write buffer to overcome the serious performance degradation caused by long writes. Another hybrid SRAM-MRAM cache scheme is given in [14]. In [11], Sun et al. propose an iterative write-read-verify mechanism to tolerate high bit error rates exhibited by MTJ cells in extremely small scales. However, their research focuses on performance, while our work focuses mostly on power and we deliberately reduce the reliability of cell write operation to benefit from reduced write power.

In [15], Zhou et al. propose Early-Write Termination, a circuit technique that avoids writing identical bits to STT-RAM caches by sensing the stored bit early in the write process. Our approach is orthogonal to EWT, and in fact could employ EWT as the mechanism that detects success or failure of individual low-current bit writes.

## VI. Conclusions

This work proposes Low-Current Probabilistic Writes, a technique that reduces the write power dissipation in emerging MRAM caches. By reducing the MTJ switching current, thereby relaxing the cell's high switching probability constraints, and by deploying a simple iterative technique that repeatedly attempts a write operation until all bits have been successfully written, the LCPW approach can reduce write power consumption by 10% and total cache power by up to 9.2%. Although this comes at the cost of longer write latencies, our experiments indicate that small-sized write buffers can completely hide all delays associated with LCPW. Finally, LCPW remains effective in lowering write power across MRAM designs.

## References

[1] D. Brooks, V. Tiwari, and M. Martonosi. Wattch: A framework for architectural-level power analysis and optimizations. In *Proceedings of the International Symposium on Computer Architecture*, pages 83–94, 2000.

[2] E. Chen, D. Apalkov, Z. Diao, et al. Advances and future prospects of spin-transfer torque random access memory. *IEEE Transactions on Magnetics*, 46(6):1873–1878, 2010.

[3] X. Dong, X. Wu, Y. Xie, et al. Stacking magnetic random access memory atop microprocessors: an architecture-level evaluation. *IET Computers and Digital Techniques*, 5(3):213–220, 2011.

[4] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi. NVSim: A circuit-level performance, energy, and area model for emerging non-volatile memory. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 31(7):994–1007, 2012.

[5] M. Hosomi, H. Yamagishi, T. Yamamoto, et al. A novel nonvolatile memory with spin torque transfer magnetization switching: iSpin-RAM. In *International Electron Devices Meeting*, pages 459–462, 2005.

[6] International Technology Roadmap for Semiconductors. Process Integration, Devices, and Structures 2010 Update. http://www.itrs.net/.

[7] C. Kim, J.-J. Kim, S. Mukhopadhyay, and K. Roy. A forward body-biased low-leakage SRAM cache: device, circuit and architecture considerations. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 13(3):349–357, 2005.

[8] C. Smullen. *Designing Giga-scale Memory Systems with STT-RAM*. PhD thesis, University of Virginia, 2011.

[9] C. Smullen, V. Mohan, A. Nigam, et al. Relaxing non-volatility for fast and energy-efficient stt-ram caches. In *Proceedings of the International Symposium on High-Performance Computer Architecture*, pages 50–61, 2011.

[10] G. Sun, X. Dong, Y. Xie, et al. A novel architecture of the 3D stacked MRAM L2 cache for CMPs. In *Proceedings of the International Symposium on High-Performance Computer Architecture*, pages 239–249, 2009.

[11] H. Sun, C. Liu, N. Zheng, et al. Design techniques to improve the device write margin for MRAM-based cache memory. In *Proceedings of the Great Lakes Symposium on VLSI*, pages 97–102, 2011.

[12] S. Thoziyoor, J. H. Ahn, M. Monchiero, et al. A comprehensive memory modeling tool and its application to the design and analysis of future memory hierarchies. In *Proceedings of the International Symposium on Computer Architecture*, pages 51–62, 2008.

[13] D. Tullsen. Simulation and modeling of a simultaneous multithreading processor. In *Proceedings of the Computer Measurement Group Conference*, 1996.

[14] X. Wu, J. Li, L. Zhang, et al. Power and performance of read-write aware hybrid caches with non-volatile memories. In *Proceedings of the Design, Automation Test in Europe Conference Exhibition*, pages 737–742, 2009.

[15] 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*, pages 264–268, 2009.