DOVA PRO: A Dynamic Overwriting Voltage Adjustment Technique for STT-MRAM L1 Cache Considering Dielectric Breakdown Effect

Jinbo Chen, Chengcheng Lu, Jiacheng Ni, Xiaochen Guo, *Senior Member, IEEE*, Patrick Girard, *Fellow, IEEE*, and Yuanqing Cheng, *Senior Member, IEEE*

***Abstract*— As device integration density increases exponentially as predicted by Moore's law, power consumption becomes a bottleneck for system scaling where leakage power of on-chip cache occupies a large fraction of the total power budget. Spin transfer torque magnetic random access memory (STT-MRAM) is a promising candidate to replace static random access memory (SRAM) as an on-chip last level cache (LLC) due to its ultralow leakage power, high integration density, and nonvolatility. Moreover, with the prevalence of edge computing and Internet-of-Things (IoT) applications, it can be beneficial to build a total nonvolatile cache hierarchy, including the L1 cache. However, building an L1 cache with STT-MRAM still faces severe challenges particularly because reducing its relatively high write latency by increasing write voltage can accelerate oxide breakdown of the MTJ device and threaten the L1 cache lifetime significantly due to intensive accesses. In our previous work, we proposed a dynamic overwriting voltage adjustment (DOVA) technique to deal with this challenge. In this article, we improve this technique by a DOVA promotion (DOVA PRO) technique for the STT-MRAM L1 cache, considering the cache write endurance and performance simultaneously. A high write voltage is used for performance-critical cache lines, while a low write voltage is used for other cache lines to approach an optimal tradeoff between reliability and performance. Experimental results show that the proposed technique DOVA PRO can improve cache performance by 23.5%, on average, compared to the DOVA technique. In the meantime, the average degradation of cache lifetime remains almost unchanged compared with the DOVA technique on average. Furthermore, DOVA PRO can support flexible configurations to achieve various optimization targets, such as higher performance or a longer lifetime.**

Manuscript received October 12, 2020; revised March 11, 2021; accepted April 5, 2021. This work was supported in part by the Beijing Natural Science Foundation under Grant 4192035 and in part by the Science, Technology and Innovation Commission of Shenzhen Municipality under Grant JCYJ20180307123657364. *(Corresponding author: Yuanqing Cheng.)*

Jinbo Chen was with the School of Electrical and Information Engineering, Beihang University, Beijing 100191, China. He is now with the School of Engineering, Westlake University, Hangzhou 310024, China (e-mail: jinbo.chen@foxmail.com).

Chengcheng Lu, Jiacheng Ni, and Yuanqing Cheng are with the MIIT Key Laboratory of Spintronics, School of Integrated Circuit Science and Engineering, Beihang University, Beijing 100191, China (e-mail: yuanqing@ieee.org).

Xiaochen Guo is with the Department of Electrical and Computer Engineering, Lehigh University, Bethlehem, PA 18015 USA.

Patrick Girard is with the Laboratory of Computer Science, Robotics and Microelectronics of Montpellier (LIRMM), University of Montpellier/CNRS, 34095 Montpellier, France.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TVLSI.2021.3073415.

Digital Object Identifier 10.1109/TVLSI.2021.3073415

***Index Terms*— Performance, reliability, spin transfer torque magnetic random access memory (STT-MRAM), time-dependent dielectric breakdown (TDDB).**

I. Introduction

THE shrinking of transistor feature size enables fast switching speed and high integration density. However, the off-chip memory bandwidth cannot improve at the same pace. In order to fill the performance gap between the processor and the main memory, on-chip cache size increases quickly and induces large area and power overhead [2].

Spin transfer torque magnetic random access memory (STT-MRAM) is a promising candidate to replace static random access memory (SRAM) for on-chip caches owing to its small cell size and ultralow leakage power [3]. STT-MRAM is a nonvolatile memory and is especially suitable to replace SRAM as the last level cache (LLC) to save energy and keep data persistent with ultralow power consumption [4]. Recently, several ultralow-power processors with totally nonvolatile memory hierarchy have been proposed for energy harvesting [5], resistive computation [6], computing in memory design [7], and so on. These applications require the cache to be totally made by nonvolatile memory, such as STT-MRAM.

Since the write performance of STT-MRAM depends on the write voltage magnitude significantly, we can improve the write performance by increasing the write voltage. However, as the L1 cache is more write-intensive compared to lower level caches, high write voltage plus intensive write accesses can accelerate oxide breakdown of the MTJ device and threaten the lifetime of the L1 cache dramatically [8]. Moreover, high write voltage incurs more write energy, which may swallow the energy savings brought by STT-MRAM.

In this work, we manage to accelerate the write speed of the STT-MRAM L1 cache while not degrading the lifetime of STT-MRAM significantly. To achieve this goal, we propose a dynamic overwriting voltage adjustment promotion (DOVA PRO) technique, which classifies write operations in the L1 cache as critical writes and noncritical writes. The critical writes adopt a high write voltage to reduce write latency, while noncritical ones use a low voltage to save write energy and prolong the lifetime of STT-MRAM. To the best of our knowledge, this is the first work to explore STT-MRAM based

totally nonvolatile cache hierarchy considering performance, energy, and reliability together. The main contributions are listed as follows.

1) Noting that the write speed of STT-MRAM depends on the write voltage, we propose a stochastic reliability model to characterize the dependence of STT-MRAM lifetime on write voltage quantitatively.
2) To choose write voltage appropriately, L1 writes are classified into critical writes and noncritical writes. A critical write prediction technique is proposed such that the proper write voltage can be selected accordingly.
3) Experimental results on SPEC2006 benchmarks running on a quad-core CPU show that DOVA PRO can improve cache performance by 23.5%, on average, compared to our previously proposed DOVA technique [1]. In the meantime, the average degradation of cache lifetime remains almost unchanged compared with the DOVA technique with negligible storage overhead.
4) DOVA PRO can be used to achieve various optimization goals to obtain either higher performance or a longer lifetime, which provides cache system designers a flexible design space.

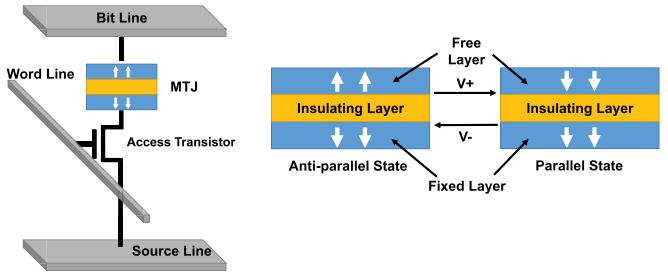
The rest of this article is organized as follows. Section II introduces STT-MRAM technology and the dielectric breakdown mechanism of STT-MRAM. Section III summarizes the related work. Section IV first establishes an STT-MRAM reliability model by investigating the relationship between dielectric breakdown and write voltage quantitatively. Then, a statistical analysis on write operations in the L1 cache is presented to motivate our work. Section V describes the implementation details of DOVA PRO. Section VI presents experimental results in terms of write energy, performance, lifetime, and sensitivity analyses. Section VII concludes this article.

II. Background

A. Introduction to STT-MRAM

The commonly used STT-MRAM cell is made of an access transistor and a magnetic tunnel junction (MTJ), as shown in Fig. 1(a). The access transistor controls read and write operations on STT-MRAM. An MTJ is made up of an insulating (oxide) layer that is typically MgO and two ferromagnetic layers. The magnetization of one ferromagnetic layer is fixed, called the fixed layer, whereas the magnetization of the other layer, called the free layer, can be changed by injecting a specific spin-polarized current in MTJ. The magnetizations of the two ferromagnetic layers represent the bit stored in a cell. Specifically, the antiparallel magnetization has high resistance representing a logic "1," and the parallel magnetization represents a logic "0," as shown in Fig. 1(b).

The write operation in an STT-MRAM cell is performed by setting up the voltage difference between a source line and a bitline. Writing "0" to an STT-MRAM cell (reset operation) is completed by applying a large positive voltage between the source line and the bitline. Writing "1" (set operation) is achieved by applying a negative voltage between the two. The read operation is performed by applying a small sensing voltage between the source line and the bitline to generate a sensing current, which is then compared with a reference current to detect "0" or "1" stored in an STT-MRAM cell.



(a) A 1T-1MTJ STT-MRAM cell (b) The resistive switching of an MTJ

Fig. 1. Illustration of the structure of a 1T-1MTJ cell and an MTJ. (a) 1T-1MTJ STT-MRAM cell. (b) Resistive switching of an MTJ.

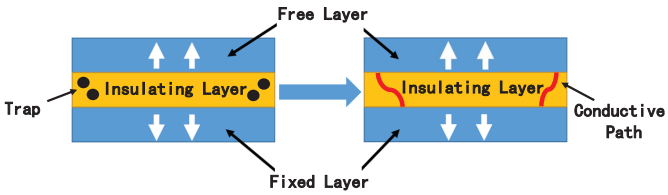


Fig. 2. Illustration of the TDDB effect of an MTJ.

B. Time-Dependent Dielectric Breakdown (TDDB) of STT-MRAM

Experimental measurements confirm that the breakdown of insulating layer within an MTJ highly depends on the write voltage, and the constant high-voltage stress is one of the main sources of MTJ failure [9], as shown in Fig. 2. A certain duration of high-voltage stress can cause oxide trapping in the insulating (oxide) layer. In the oxide breakdown process, factors such as oxide thickness and impurities in the oxide layer can also accelerate the formation of oxide trapping. These traps can happen at any time under the influence of voltage stress and are irreversible. These traps eventually overlap with each other, forming a conductive path between the fixed layer and the free layer, hence reducing the effective resistance of the MTJ and causing the malfunction of the STT-MRAM cache.

III. Related Work

A. Nonvolatile Cache Hierarchy Design

Most of the state-of-the-art STT-MRAM-based L1 caches are implemented with a buffer to cope with the high write latency issue, such as the SRAM buffer in [10] and the very wide buffer in [11]. Although adding a buffer is helpful, it induces a hybrid CMOS/MTJ cache hierarchy, which contradicts the pure nonvolatile cache hierarchy investigated in this article, and may not be suitable for the ultralow-power requirement of edge computing devices.

In terms of the pure STT-MRAM cache hierarchy, Guo *et al*. [6] presented a resistive computation design, which aimed at eliminating the power wall by migrating most of

the components of a modern microprocessor from CMOS to STT-MRAM. Natsui *et al*. [7] presented an MTJ/MOS-hybrid video coding hardware that uses a cycle-based power-gating technique for an MTJ-based nonvolatile logic-in-memory chip with the established semiautomated MTJ-oriented design flow. However, these works did not consider the TDDB reliability issues of the STT-MRAM cache.

B. Nonvolatile Cache Lifetime Extension Techniques

Conventional nonvolatile cache lifetime extension techniques can be divided into two categories. The first category focused on balancing cache intraset and interset access variations in order to avoid fast wear-out of frequently accessed cache lines [12]–[14]. Further improvements based on the same concept have also been proposed in [15] and [16].

The second category has resorted to error correction codes. Conventional ECC has been implemented in some STT-MRAM cache architecture, such as in [17]. New ECC strategy, for example, interleaved single-error correction-double error detection (SEC-DED), has been proposed in [18] to improve the lifetime of L2 and L3 caches. Different from these works, we deal with the TDDB reliability problem by adjusting write voltage without sacrificing the precious storage space for the ECC code.

C. Performance–Lifetime Tradeoffs of NVMs

Most of the existing works in this field focused on phase-change memory (PCM) and resistive random access memory (ReRAM). Qureshi *et al*. [19] proposed to cancel or pause the ongoing write operations when critical read requests arrived, thus improving the PCM lifetime. The authors also investigated the latency differences between fast RESET and slow SET operations and came up with a scheme to improve the write performance by proactively performing the SET operations. Zhang *et al*. [20] proposed a type of wear leveling technique for ReRAM, called Mellow Writes, which reduces the wear-out induced by individual write rather than reducing the number of writes. It presented three microarchitectural optimizations (Bank-Aware Mellow Writes, Eager Mellow Writes, and Wear Quota) that selectively performed slow writes to increase memory lifetime while minimizing the performance impact. Zhang *et al*. [21] proposed a region retention monitor (RRM) for PCRAM, a structure that recorded and predicted the write time of PCM memory regions, based on the observation that only a small part of memory (i.e., hot memory regions) was frequently accessed in a given period of time. For every incoming memory write operation, RRM selected a proper write latency, which helped the system improve the balance between system performance and memory lifetime. The most relevant work is our previously proposed "DOVA" technique [1]. With an STT-MRAM TDDB reliability model, all the read-intensive lines except for the write-intensive lines were considered as critical lines that were written by the high write voltage. The rest of the lines were written with the low voltage, and we set DOVA as the baseline in this work.

TABLE I
$t_{63\%}$ WITH DIFFERENT WRITE VOLTAGES

Write Voltage/V	$t_{63\%}$/s
1.81	9.802
1.75	49.457
1.69	264.023

IV. MODELING OF STT-MRAM LIFETIME DEPENDENCE ON THE WRITE VOLTAGE

A. Stochastic Modeling of the Write Endurance of a Single MTJ Due to the TDDB Effect

The MTJ breakdown procedure mentioned above is a time-dependent probabilistic event. The Weibull distribution is widely used for reliability analysis and lifetime prediction of semiconductor devices, which can be used to describe the statistical characteristics of the MTJ TDDB breakdown [22]. The MTJ breakdown mechanism can be described by the Weibull distribution as follows:

$$F(t_s) = 1 - e^{\left(-\frac{t_s}{t_{63\%}}\right)^{\beta}} \tag{1}$$

where t_s denotes the voltage stress duration, namely, the accumulated switching time of an MTJ, $t_{63\%}$ is the specific accumulated switching time when the breakdown probability approaches 63%, and β represents the shape parameter of the Weibull distribution. For a specific MTJ, β is related to oxide thickness and process variation and is independent of the stress voltage [22].

The Weibull distribution can be converted into the following linear form:

$$\ln\{-\ln[1 - F(t_s)]\} = \beta(\ln t_s - \ln t_{63\%}) \tag{2}$$

where β can be obtained by interpolation of experimental measurements.

Our reliability model is built on the Weibull distribution, considering MTJ write endurance and MTJ physical structure from [22] (refer to Section V-A for more details).

Assume that the thickness of the MgO oxide layer is 1.25 nm [22]. Let $\ln\{-\ln[1 - F(t_s)]\} = 0$ to obtain the corresponding voltage stress time $t_{63\%}$, and the result is shown in Table I.

Moreover, considering the voltage acceleration of barrier breakdown in MgO-based MTJ, three sets of parameters shown in Table I can be used to calculate $t_{63\%}$ with different write voltages based on the voltage power law [23]

$$t_{63\%} = aV^{-M} \tag{3}$$

where a is a process-dependent parameter and M denotes the voltage acceleration factor (AF).

According to the three pairs of $(t_{63\%}, V)$ obtained from Fig. 3, we can derive the parameters as follows:

$$a = 2.3 \times 10^{13} \quad M = 48.01. \tag{4}$$

Thus, we can obtain

$$t_{63\%} = 2.3 \times 10^{13} \times V^{-48.01}. \tag{5}$$

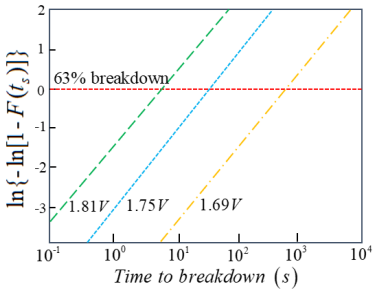


Fig. 3. TDDB effect of STT-MRAM characterized by the Weibull distribution (reproduced from [22]).

TABLE II
MTJ WRITE ENDURANCE

MTJ write voltage /V	MTJ write endurance /cycles
1.18	2.75×10^{18}
1.41	6.85×10^{14}
1.60	1.97×10^{12}
1.81	5.72×10^{9}

Then, with $t_{63\%}$ and MTJ write latency, we can get its write endurance with the following formula [24]:

$$E = \frac{t_{63\%}}{t_{\text{MTJ write latency}}} \tag{6}$$

where E is the MTJ endurance.

Based on the formula and different pairs of $t_{63\%}$ and MTJ write latency, we can get its write endurance estimation, as shown in Table II. We can observe that, with the increase in the write voltage, write endurance decreases dramatically.

Intuitively, it is possible to calculate cache lifetime from MTJ write endurance and write times when running applications. In this article, however, an MTJ cell is applied with different write voltages depending on the cache access behavior. Therefore, we need a method that can figure out the MTJ effective write times under different write voltages. To deal with this problem, the formula of MTJ write time AF is given as follows [25]:

$$\text{AF} = \left(\frac{t_{v_1}}{t_{v_2}}\right)^N \tag{7}$$

where t_{v_1} is the MTJ write latency corresponding to voltage 1, and t_{v_2} is the MTJ write latency corresponding to voltage 2. N relates to MTJ activation energy of failure, practically ranging from 1 to 3. In general, higher MTJ write voltage leads to shorter MTJ write latency and shorter lifetime. Based on (7), we derive the formula of calculating the MTJ effective write times with different write voltages

$$n_{\text{eff}} = n_{v_1} + \text{AF} \times n_{v_2} \tag{8}$$

where n_{v_1} and n_{v_2} are, respectively, MTJ write times of write voltage 1 and write voltage 2.

From (7) and (8), we can observe that the MTJ write time AF is exponentially related to the parameter N. In other words, when the MTJ write latency is determined, the MTJ effective write time increases exponentially with the value of N. Therefore, parameter N is critical for cache lifetime, and detailed experimental evaluations will be discussed in Section VI.

B. Reliability Modeling of L1 STT-MRAM Cache Considering the TDDB Effect

Based on the cache raw lifetime metric proposed in [12], the MTJ write endurance, and the effective write time model built above, we propose a reliability model for an L1 STT-MRAM cache.

The read voltage of an MTJ is relatively low and, hence, has little or no effect on the device failure. Therefore, in this article, we only consider the impact of write voltage. From the Weibull distribution, it is clear that the higher the write voltage is, the greater the MTJ failure probability is, under a given voltage stress duration.

Assuming that the cache line size is 64 bytes, and each bit is implemented with the 1T-1MTJ structure, the cache raw lifetime is defined as the time of the first MTJ failure occurring in a cache line. When running a program, we assume that each MTJ in a cache line is written if the cache line write operation happens, which means that the effective write times of a cache line are equal to the MTJ effective write times within the cache line. Adding up the effective write times of every cache line, we can get effective write times of the whole cache, namely, W

$$w_i = w_{v_1} + \text{AF} \times w_{v_2} \tag{9}$$

$$W = \sum_{i=1}^{p} w_i \tag{10}$$

where w_{v_1} and w_{v_2} are, respectively, cache write times of cache line i with write voltage 1 and write voltage 2. p is the number of cache lines in L1 cache.

Then, we are able to figure out the average effective write times of the MTJ, i.e., n_{ave}, by dividing W with the number of cache lines

$$n_{\text{ave}} = \frac{W}{\text{Number of cache lines}}. \tag{11}$$

Taking MTJ write endurance E with 1.18-V write voltage as the reference, we finally get L1 cache lifetime as follows:

$$E_{1.18V} = \frac{t_{63\%}}{t_{1.18\,\text{V}}} \tag{12}$$

$$\text{Lifetime} = \frac{E}{n_{\text{ave}}} \tag{13}$$

where $t_{1.18\,\text{V}}$ is the MTJ write latency with 1.18-V write voltage. In the following, we will use this model to evaluate the lifetime of L1 STT-MRAM cache.

V. DYNAMIC OVERWRITING VOLTAGE ADJUSTMENT PRO (DOVA PRO) TECHNIQUE

A. Motivation and the Main Idea

In order to improve the performance of the STT-MRAM L1 cache, an intuitive method is to increase the write voltage since write latency decreases with write voltage. We adopted

TABLE III
MTJ MODEL PARAMETERS [26]

Symbol	Value
L	40 nm
W	40 nm
K_u	$5000A/m$
α	0.007
M_s at $25°C$	$800A/m$
t_{ox}	1.25 nm
P	0.5
γ	$1.76 \times 10^7 rad/(s \cdot T)$

the MTJ model proposed in [26] to evaluate the dependence of write latency on voltage. The MTJ model parameters are listed in Table III. We performed circuit simulations on a 1T-1MTJ cell to obtain the write latency with different write voltages. Then, we fed simulation results to NVSim [27] to get the cache line write latency with different voltages. NVSim results show that, when MTJ write voltage increases from 1.18 to 1.81 V, the cache write latency can be reduced from 3.463 to 2.083 ns, i.e., reduced by 39.85%.

On the other hand, however, the increased write voltage may aggravate the TDDB effect and lead to shortened STT-MRAM lifetime according to our reliability model. From Table II, we can see that, when increasing the MTJ write voltage from 1.18 to 1.81 V, its write endurance drops by approximately nine orders of magnitude. Therefore, it is of great importance to find an acceptable tradeoff between write performance and reliability.

There are some existing works focusing on optimizing the tradeoff in nonvolatile memories with adaptive write voltages. However, most of the existing works chose to utilize customized circuits to track the critical path writes in the instruction queue to apply adaptive voltages. These designs resulted in large hardware complexity and area overhead. On the contrary, this article proposes a novel method by examining the cache access behavior and structural hazards in the CPU pipeline and, therefore, can approach the optimal tradeoff considering performance, reliability, and energy consumption together.

In modern CPU architectures, the L1 cache is commonly divided into an instruction cache and a data cache. Normally, one instruction enters the pipeline for execution in each clock cycle of CPU.[1] However, in some cases, the coming instruction cannot be executed immediately due to data dependence or structural hazard [28].

If the two consecutive accesses to the L1 cache in the pipeline are write and read operations of different cache lines, respectively, the cache read operation can only be executed after the cache write operation is completed.[2] In STT-MRAM,

[1]In this work, we take the single issue pipeline as an example. However, for the multi-issue architecture, our proposed technique is still effective.

[2]For ease of understanding, we assume the cache has only one port to reduce the power and area overheads [29]. However, for the multiport cache design, the read operation can also be blocked by the write operation due to long write latency of STT-MRAM.

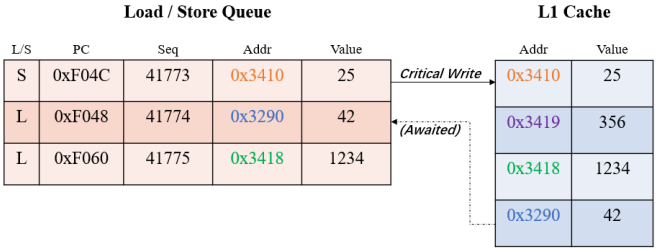


Fig. 4. Illustration of the "write blocking read" problem.

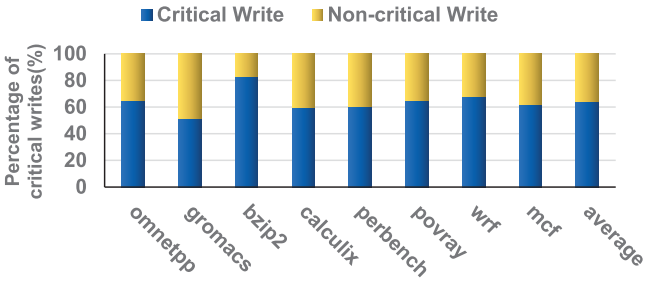


Fig. 5. Percentages of critical write operations of some SPEC2006 benchmarks.

the write latency of L1 cache is usually several times longer than the read latency, and the write operation can block the following read, thus degrading CPU performance. For example, as shown in Fig. 4, when performing a write operation to cache line 1, the next read operation to cache line 4 is stalled due to the relative long write latency.

To solve the problem, we first give the definition of *Critical Write* in cache: when the L1 cache is performing a write operation, if the next instruction issued to it is a read operation, the current write operation is considered as a critical write. In Fig. 4, the write operation to cache line 1 would be considered as a critical write. Fig. 5 shows the percentages of critical writes for several SPEC2006 benchmarks. We can observe that approximately 64% cache writes are critical writes, which should be optimized to improve the system performance.

Based on the notion of the critical write, we further define and use the *critical write ratio* (CWR) to determine performance-critical cache lines. In the program profiling stage, the number of accesses to each cache set is recorded. For a cache set, the total number of writes in the profiling stage is denoted as V, and the number of critical writes is C; then, the CWR is defined as follows:

$$\text{CWR} = \frac{C}{V}. \tag{14}$$

Then, each L1 cache set has a specific CWR value. The CWR value of a cache line equals that of the cache set where it is located. The larger the CWR value of the cache line is, the larger the probability of the "write-blocking-read" operations may occur, and it is more critical to the system performance. According to the CWR value, we can identify the *performance-critical lines* by setting a CWR threshold. If the CWR value of one cache line is larger than the threshold, it is a performance-critical line; otherwise, it is

TABLE IV
NVSim CONFIGURATIONS

Write voltage	1.18V	1.41V	1.60V	1.81V
Read voltage /V	0.6	0.6	0.6	0.6
Read energy /fJ	11.17	11.17	11.17	11.17
Write current (Reset/set) /uA	90.84 / 124.17	128.23 / 160	164.2 / 188.5	208.74 / 225.32
Write latency (Reset/set) /ns	2.96 / 3.37	2.30 / 2.65	1.85 / 2.01	1.71 / 1.99
Write energy (Reset/set) /fJ	395.30 / 452.41	479.26 / 550.75	524.16 / 555.68	664.99 / 785.18

not. For performance-critical lines, the high write voltage is used to reduce write latency to alleviate the "write blocking read" problem; for other cache lines, the low write voltage is applied to prolong the cache lifetime, thus obtaining an optimal tradeoff between performance and lifetime.

B. Implementation of DOVA PRO

Based on the previous analysis, we propose a DOVA PRO technique that aims to improve the performance of the STT-MRAM-based L1 cache while do not degrade reliability significantly.

The detailed workflow of DOVA PRO is illustrated in Fig. 6 and is described as follows.

1) Record the read and write operations of each L1 cache set during the program profiling stage.
2) Calculate CWR of each L1 cache set based on its write times and the number of critical writes. The CWR value of a cache line equals that of the cache set where it is located.
3) Compare the CWR value of each L1 cache set with the default threshold; then, generate the table of critical write (TCW) to store the comparison results: 1 denotes that the CWR value is larger than the default threshold, while 0 represents the opposite result.
4) During the runtime, before every write operation to L1 cache, the TCW is first checked to identify if the L1 cache line is performance-critical or not. If the L1 cache line is a performance-critical line, a high write voltage will be used. Otherwise, a low write voltage will be applied.

In step 4 as mentioned above, performance-critical L1 cache lines have CWR values larger than the threshold. The optimization results are closely related to the threshold value.

1) If the optimization goal focuses on an L1 cache write performance, a small threshold can be set to get more performance-critical lines written by the high voltage, in order to reduce the write latency.
2) If L1 cache lifetime is more important, a large threshold should be used to reduce the number of cache lines written by the high voltage.

As will be shown in Section VI-C, changing CWR thresholds can achieve various optimization objectives. Therefore, the cache hierarchy designers are able to use the proposed technique to get either higher performance or a longer lifetime.

DOVA PRO requires a high supply voltage to enable the fast write operation for performance-critical lines. Since STT-MRAM typically already requires more than one supply voltage (e.g., write and read operations typically need different voltages), it can be easily extended to the multiple write voltage scheme similar to the design in [30].

The storage overhead of DOVA PRO is mainly attributed to counters and the TCW table. In order to implement DOVA PRO, we need a two-byte counter for each cache set to record the cache accesses during the profiling stage. Taking 32KB L1 cache with four-way set-associativity as an example, the total storage overhead of counters is 1 KB. Based on the configuration of the L1 cache mentioned above, the TCW table has 128 rows and one column, corresponding to 128 sets in the cache. Each element in the TCW table stores 1-bit information, indicating if the cache lines within the corresponding set are critical or not. Therefore, the storage overhead of the TCW table is 128 bits, namely, 16 bytes. In general, the storage overhead is approximately 1 KB, which is negligible compared to several MB on-chip multilevel cache capacity.

Runtime overhead is mainly due to the four implementation steps. The first three steps of the DOVA PRO technique, namely, recording cache access behavior, calculating CWR value, and generating TCW table, are all off-line operations, which have no impact on the program running performance, so they do not affect the system performance. Considering step 4, running programs need to check the TCW table before every write operation. Since the table is very small (only 128 bits), the table access time is negligible compared to the STT-MRAM cache access time.

VI. EXPERIMENTAL RESULTS

A. Experimental Setup

First, we present the experimental methodology to evaluate the proposed DOVA PRO technique. We used NVSim [27] to get parameters, including cache access latency, access energy, and leakage power at the 45-nm technology node. The NVSim simulation configurations are shown in Table IV. Afterward, our proposed strategy was implemented in a system-level simulator, i.e., gem5 [31]. The detailed gem5 simulation configurations are shown in Table V. In addition, nonblocking gem5 was modified to accommodate the asymmetry of read latency and write latency. Eight benchmarks from SPEC2006 [32] were used for performance evaluations. In addition, the energy consumption was obtained by L1 cache access statistics, and the expected lifetime was derived based on cache line access statistics produced by gem5 and the reliability model presented in Section IV. TCW was constructed by one-million-instruction profiling for every

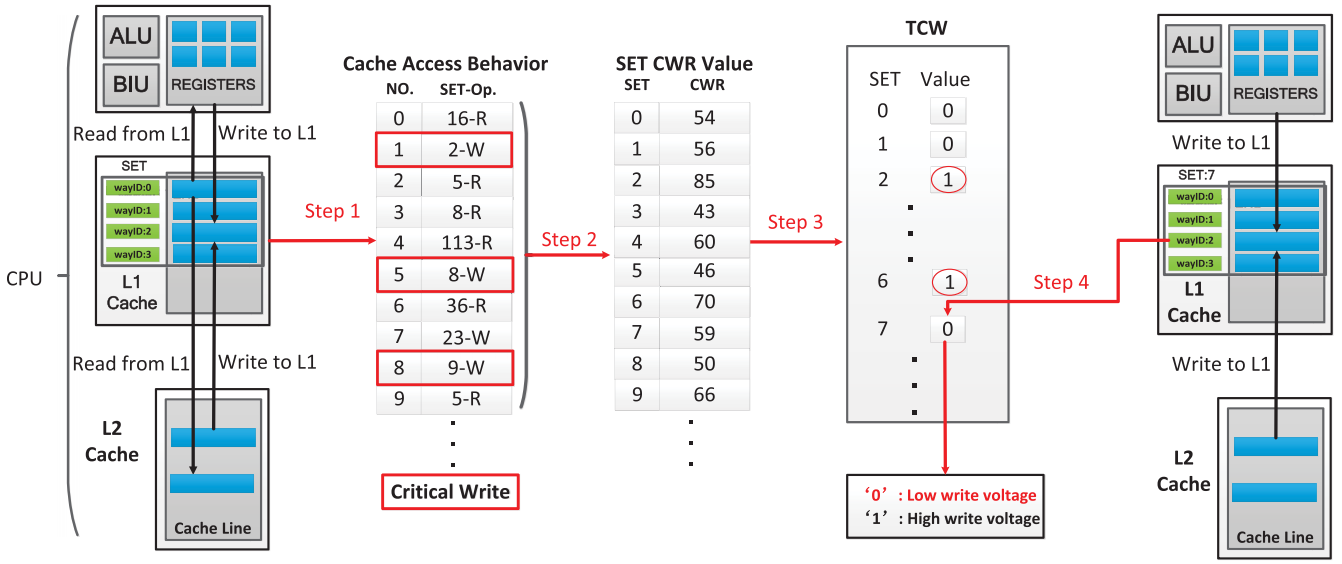


Fig. 6. Work flow of DOVA PRO technique. Step 1: record read/write access behavior (profiling). Step 2: calculate CWR. Step 3: generate the TCW. Step 4: check TCW, identify performance-critical lines, and apply write voltage accordingly.

TABLE V
gem5 SIMULATION CONFIGURATIONS

Module	Setup
CPU	Quad-core, 2.9GHz, X86, OoO
L1 Cache STT-MRAM	Private, Split I/D caches, 32KB 64Bytes block size, Write-back policy 4-way set-associativity, LRU Write latency low 8 cycles & high 11 cycles Read latency 4 cycles
L2 Cache STT-MRAM	Shared, 2MB, 64Bytes block size 8-way set-associativity Read latency 5 cycles, Write latency 12 cycles LRU, write-back policy
Main memory	8GB, DDR3
Protocol	MI_example

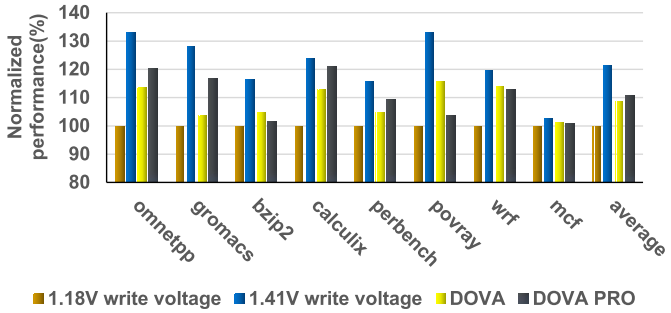


Fig. 7. Normalized performance improvements when running SPEC2K6 benchmarks on a quad-core processor.

benchmark. In the simulations, each benchmark was executed for one time with one billion instructions after 100 million warming-up instructions.

Considering the dynamic write voltage selection, we chose 1.18 V as the low write voltage, which was identical to that in the most recent work [33]. The parameter N in (7) is set to 1 as the default value. The high write voltage and CWR threshold are set to 1.41 V and 60, respectively, after considering the tradeoff between performance, energy consumption, and cache lifetime.

B. Experimental Results and Analyses

We evaluated the proposed strategy against the other three settings: only using the low write voltage 1.18 V (LWV) and only using the high write voltage 1.41 V (HWV) and our previous work [1].

1) Performance Evaluations: Fig. 7 shows the normalized performance results of DOVA PRO, HWV, and DOVA compared to the baseline, i.e., LWV. The figure indicates that the performance improvement of DOVA PRO can be up to 21.19% and 11% on average compared to the LWV baseline, which is 23.5% faster than DOVA. The above experimental data shows that DOVA PRO improves the accuracy of critical write prediction, thus further improving the system performance.

Although the improvement can be higher if HWV is applied, the expected lifetime and energy consumption are seriously deteriorated, which will be discussed next.

2) Cache Lifetime: Fig. 8 presents the cache lifetime of DOVA PRO compared with the other three schemes. From the result, we can observe that DOVA PRO leads to 10.6% average degradation of lifetime compared to the LWV case but remains almost unchanged compared with the DOVA technique and is 15.06% better than the HWV case. This is because DOVA PRO could distinguish between critical writes and uncritical writes, hence achieving a better tradeoff between performance and lifetime.

3) Write Energy Consumptions: Fig. 9 shows normalized write energy for all four schemes. DOVA PRO only incurs

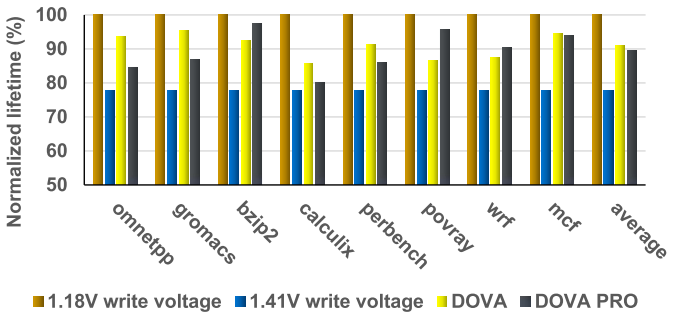


Fig. 8. Normalized lifetime comparisons of different schemes.

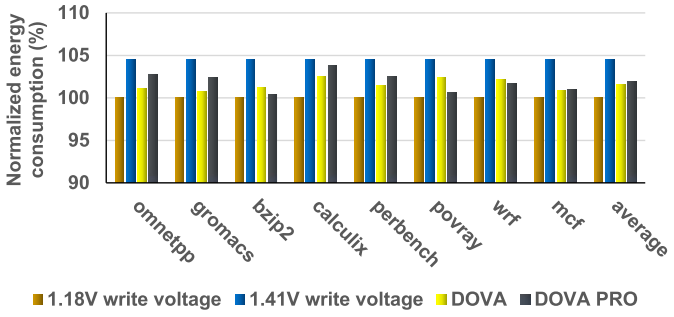


Fig. 9. Normalized energy consumptions of different schemes.

1.91% energy overhead on average (0.39% minimum). Conversely, the HWV case incurs 4.47% energy overhead on average because it uses the high voltage for all write accesses, which is unnecessary since only cache critical writes are crucial to system performance, as mentioned in Section V.

C. Sensitivity Analysis of the CWR Threshold Value

In step 4 of the DOVA PRO scheme, when setting different values of the CWR threshold, the number of performance-critical lines changes accordingly, thus causing different optimization results. In Section VI-B, the CWR threshold is set to 60 by default; that is, if the CWR value of the cache line is higher than 60, it is predicted as a critical cache line; otherwise, it is not. In the sensitivity analysis, we set the CWR value as 60, 65, and 70, respectively, to see how the system performance is affected.

The experimental results in Fig. 10 indicate that, setting the high write voltage of DOVA PRO to 1.41 V, compared with the only low write voltage case, when the CWR threshold reduces from 70 to 60, the program performance improvement increases from 5.36% to 10.85%. However, this also leads to the decrease of cache lifetime from 93.76% to 89.4% compared to the baseline case. The increase in energy consumption under different CWR thresholds remains almost the same, as shown in Fig.12(a).

In summary: 1) if the optimization goal is to improve the system running speed, the CWR threshold can be set to 60 or less so that more cache lines are written by the high voltage; 2) if the goal is to prolong the cache lifetime, the CWR threshold can be set to 70 or more, so as to guarantee the lifetime without degrading the performance significantly; and 3) if the optimization goal is achieving the balance of running speed and cache lifetime, the CWR threshold can be set to 65 or so. Therefore, by changing the threshold of CWR, different optimization objectives can be achieved flexibly.

TABLE VI
RELATIONSHIP OF MTJ WRITE VOLTAGE AND CACHE WRITE LATENCY AND CACHE WRITE ENERGY

MTJ Write Voltage /V	Cache Write Latency /ns	Cache Write Energy /nJ
1.41	2.743	0.421
1.60	2.103	0.422
1.81	2.083	0.463

D. Sensitivity Analysis of the High Write Voltage Selection

The high write voltage of DOVA PRO is of great importance to the system performance. Table VI shows that, when the MTJ write voltage increases, the cache write latency decreases significantly. However, the write energy consumption increases as well. The MTJ write endurance also decreases by several orders of magnitude according to Table II.

As shown in Fig. 11, when the CWR threshold is set to 60, compared with the only low write voltage 1.18-V case, when the high write voltage increases from 1.41 to 1.81 V, the running speed improvement increases from 10.85% to 18.62% compared to the baseline case, while the cache lifetime deteriorates from 89.4% to 77.6% of that of the baseline case. The energy consumption also increases from 1.91% to 6.38% compared to the baseline in Fig. 12(b).

In summary, when the high write voltage of DOVA PRO varies, the system performance will change remarkably. In practice, the amplitude of high write voltage should be carefully selected to achieve a balanced optimization result. The default value of high write voltage in Section VI-B is 1.41 V. In this case, the system can get an appropriate tradeoff on the program running speed, cache lifetime, and energy consumption. Thus, it is used as the default setting in this work.

E. Sensitivity Analysis of the Reliability Model

In Section IV, we showed that cache lifetime was closely related with the parameter N in (7). With the previous analysis, $N = 1$ was reasonable for our exploration. However, for various MTJ manufacturing techniques, their N values might be different. We explored how our system performed as N changed. In addition to the default value 1, we compared the lifetime of our schemes with the case when $N = 2$. We wanted to figure out how much the cache lifetime depended on the exponential factor N.

We set the CWR threshold to 60 and the high write voltage to 1.41 V. According to the experimental results in Fig. 13, when the parameter N increases from 1 to 2, the cache lifetime decreases from 89.4% to 79.31% compared to the baseline. Therefore, in order to achieve a long cache lifetime, it is expected to optimize relevant MTJ manufacturing techniques so that the value of N can be reduced as much as possible.

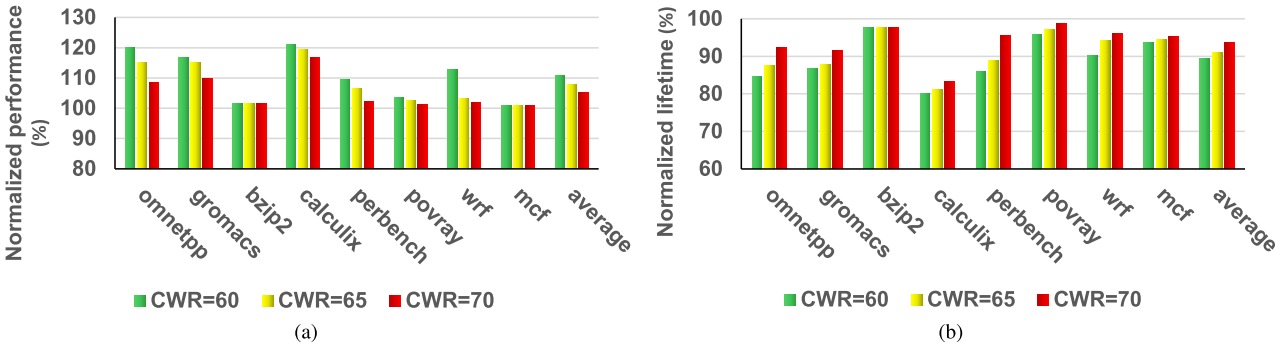


Fig. 10. Sensitivity to CWR threshold settings. (a) Normalized performance improvement. (b) Normalized cache lifetime.

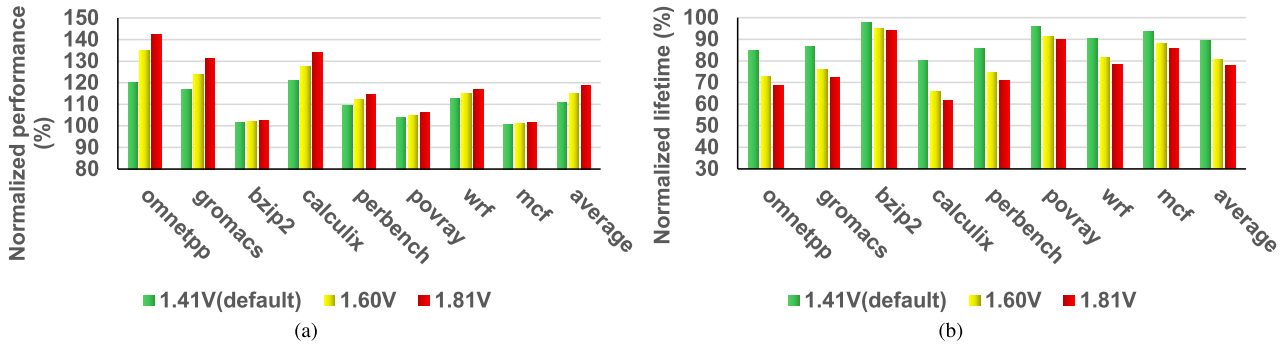


Fig. 11. Sensitivity to high write voltage settings. (a) Normalized performance improvement. (b) Normalized cache lifetime.

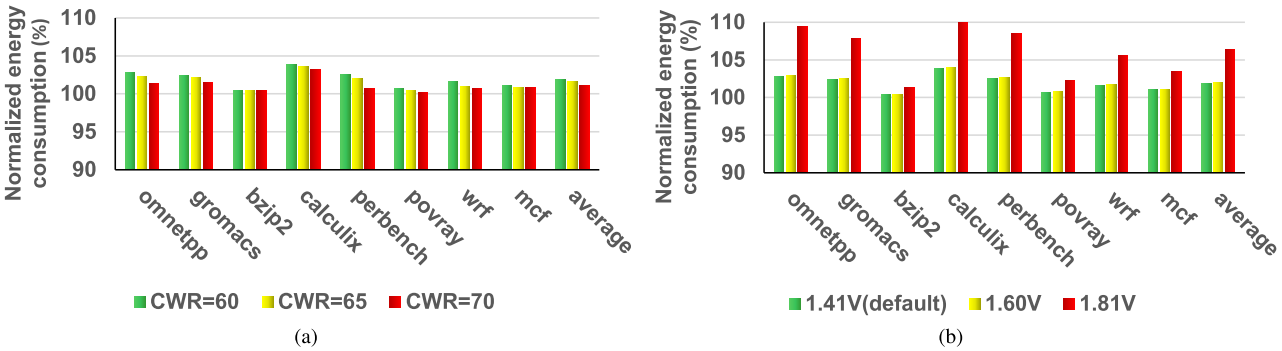


Fig. 12. Normalized energy consumptions. (a) Sensitivity to CWR threshold settings. (b) Sensitivity to high write voltage settings.

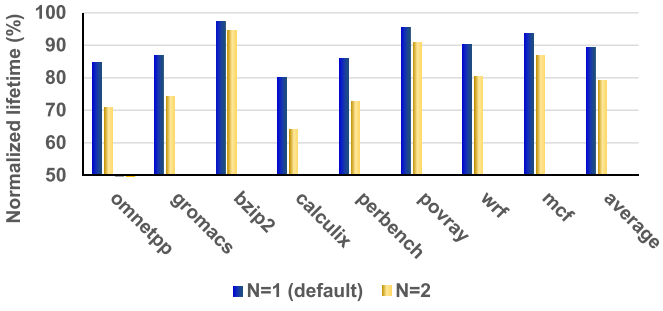


Fig. 13. Sensitivity analysis to the reliability model.

VII. Conclusion

STT-MRAM is a competitive candidate to build caches to replace SRAM because of its advantages, such as higher density, better durability, and nonvolatility. However, it suffers from the problem of high write latency, and it is necessary to accelerate write speed to enable STT-MRAM-based L1 cache. Increasing write voltage is an effective method. However, this high write voltage may reduce STT-MRAM lifetime due to the TDDB effect. Thus, it is crucial to make an optimal tradeoff between write performance and the lifetime of the STT-MRAM L1 cache.

In this article, we propose a DOVA PRO technique to write different types of cache lines with different write voltages. Experimental results show that, with default settings, DOVA PRO can improve cache speed performance by 21.19% in maximum and 11% on average compared to baseline, which is 23.5% faster than the DOVA technique. In the meantime, the average degradation of cache lifetime is only 10.6% compared to the baseline, which remains almost unchanged

compared with the DOVA technique. Moreover, the write energy consumption increase in DOVA PRO is 1.91% on average (0.39% in minimum), which is almost the same as the DOVA technique and much lower than the 4.47% write energy consumption increase in the high write voltage case. Furthermore, we can easily configure the parameters of DOVA PRO to achieve various optimization goals and get either higher performance or a longer lifetime.

REFERENCES

[1] J. Chen, K. Liu, X. Guo, P. Girard, and Y. Cheng, "DOVA: A dynamic overwriting voltage adjustment for STT-RAM L1 cache," in *Proc. 21st Int. Symp. Qual. Electron. Design (ISQED)*, Mar. 2020, pp. 1–6.

[2] Y. Meng, T. Sherwood, and R. Kastner, "Exploring the limits of leakage power reduction in caches," *ACM Trans. Archit. Code Optim.*, vol. 2, no. 3, pp. 221–246, Sep. 2005.

[3] L. Wang, Y. Zhang, Z. Wang, W. Zhao, S. Peng, and L. Chang, "Recent progresses in spin transfer torque-based magnetoresistive random access memory (STT-MRAM)," *Scientia Sinica Phys., Mechanica Astronomica*, vol. 46, no. 10, Oct. 2016, Art. no. 107306.

[4] S. P. Park, S. Gupta, N. Mojumder, A. Raghunathan, and K. Roy, "Future cache design using STT MRAMs for improved energy efficiency: Devices, circuits and architecture," in *Proc. 49th Annu. Design Autom. Conf. (DAC)*, 2012, pp. 492–497.

[5] K. Ma *et al.*, "Architecture exploration for ambient energy harvesting nonvolatile processors," in *Proc. IEEE 21st Int. Symp. High Perform. Comput. Archit. (HPCA)*, Feb. 2015, pp. 526–537.

[6] X. Guo, E. Ipek, and T. Soyata, "Resistive computation: Avoiding the power wall with low-leakage, STT-MRAM based computing," *ACM SIGARCH Comput. Archit. News*, vol. 38, no. 3, pp. 371–382, 2010.

[7] M. Natsui *et al.*, "Nonvolatile logic-in-memory array processor in 90 nm MTJ/MOS achieving 75% leakage reduction using cycle-based power gating," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2013, pp. 194–195.

[8] K. Kim, C. Choi, Y. Oh, H. Sukegawa, S. Mitani, and Y. Song, "Time-dependent dielectric breakdown of MgO magnetic tunnel junctions and novel test method," *Jpn. J. Appl. Phys.*, vol. 56, no. 4S, 2017, Art. no. 04CN02.

[9] S. Amara-Dababi, H. Bea, R. Sousa, K. Mackay, and B. Dieny, "Modelling of time-dependent dielectric barrier breakdown mechanisms in MgO-based magnetic tunnel junctions," *J. Phys. D: Appl. Phys.*, vol. 45, no. 29, Jul. 2012, Art. no. 295002.

[10] H. Sun, C. Liu, W. Xu, J. Zhao, N. Zheng, and T. Zhang, "Using magnetic RAM to build low-power and soft error-resilient L1 cache," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 1, pp. 19–28, Jan. 2012.

[11] M. P. Komalan, C. Tenllado, J. I. Gómez Pérez, F. T. Fernández, and F. Catthoor, "System level exploration of a STT-MRAM based level 1 data-cache," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2015, pp. 1311–1316.

[12] J. Wang, X. Dong, Y. Xie, and N. P. Jouppi, "I2WAP: Improving non-volatile cache lifetime by reducing inter- and intra-set write variations," in *Proc. IEEE 19th Int. Symp. High Perform. Comput. Archit. (HPCA)*, Feb. 2013, pp. 234–245.

[13] S. Mittal, "Using cache-coloring to mitigate inter-set write variation in non-volatile caches," 2013, *arXiv:1310.8494*. [Online]. Available: http://arxiv.org/abs/1310.8494

[14] S. Mittal and J. S. Vetter, "EqualWrites: Reducing intra-set write variations for enhancing lifetime of non-volatile caches," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 1, pp. 103–114, Jan. 2016.

[15] S. Agarwal and H. K. Kapoor, "Targeting inter set write variation to improve the lifetime of non-volatile cache using fellow sets," in *Proc. IFIP/IEEE Int. Conf. Very Large Scale Integr. (VLSI-SoC)*, Oct. 2017, pp. 1–6.

[16] S. Agarwal and H. K. Kapoor, "Enhancing the lifetime of non-volatile caches by exploiting module-wise write restriction," in *Proc. Great Lakes Symp. VLSI*, 2019, pp. 213–218.

[17] 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 *Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2014, pp. 1–2.

[18] H. Farbeh, H. Kim, S. G. Miremadi, and S. Kim, "Floating-ECC: Dynamic repositioning of error correcting code bits for extending the lifetime of STT-RAM caches," *IEEE Trans. Comput.*, vol. 65, no. 12, pp. 3661–3675, Dec. 2016.

[19] M. K. Qureshi, M. M. Franceschini, and L. A. Lastras-Montano, "Improving read performance of phase change memories via write cancellation and write pausing," in *Proc. 16th Int. Symp. High-Perform. Comput. Archit. (HPCA)*, Jan. 2010, pp. 1–11.

[20] L. Zhang, B. Neely, D. Franklin, D. Strukov, Y. Xie, and F. T. Chong, "Mellow writes: Extending lifetime in resistive memories through selective slow write backs," in *Proc. ACM/IEEE 43rd Annu. Int. Symp. Comput. Archit. (ISCA)*, Jun. 2016, pp. 519–531.

[21] M. Zhang, L. Zhang, L. Jiang, Z. Liu, and F. T. Chong, "Balancing performance and lifetime of MLC PCM by using a region retention monitor," in *Proc. IEEE Int. Symp. High Perform. Comput. Archit. (HPCA)*, Feb. 2017, pp. 385–396.

[22] Y. Wang *et al.*, "Compact model of dielectric breakdown in spin-transfer torque magnetic tunnel junction," *IEEE Trans. Electron Devices*, vol. 63, no. 4, pp. 1762–1767, Apr. 2016.

[23] S. Van Beek *et al.*, "Voltage acceleration and pulse dependence of barrier breakdown in MgO based magnetic tunnel junctions," in *Proc. IEEE Int. Rel. Phys. Symp. (IRPS)*, Apr. 2016, pp. MY-4-1–MY-4-4.

[24] J. J. Kan *et al.*, "Systematic validation of 2× nm diameter perpendicular MTJ arrays and MgO barrier for sub-10 nm embedded STT-MRAM with practically unlimited endurance," in *IEDM Tech. Dig.*, Dec. 2016, pp. 4–27.

[25] D. B. Strukov, "Endurance-write-speed tradeoffs in nonvolatile memories," *Appl. Phys. A, Solids Surf.*, vol. 122, no. 4, p. 302, Apr. 2016.

[26] X. Fong, S. H. Choday, P. Georgios, C. Augustine, and K. Roy, "Purdue nanoelectronics research laboratory magnetic tunnel junction model," nanoHUB, 2014, doi: 10.4231/D33R0PV04.

[27] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi, "NVSim: A circuit-level performance, energy, and area model for emerging nonvolatile memory," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 31, no. 7, pp. 994–1007, Jul. 2012.

[28] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*. Amsterdam, The Netherlands: Elsevier, 2011.

[29] D. A. Patterson and J. L. Hennessy, *Computer Organization and Design ARM Edition: The Hardware Software Interface*. Amsterdam, The Netherlands: Elsevier, 2016.

[30] D. H. Sohn, C. K. Kim, and Y. S. Lee, "Memory system having variable operating voltage and related method of operation," U.S. Patent 9 076 542, Jul. 7, 2015.

[31] N. Binkert *et al.*, "The gem5 simulator," *ACM SIGARCH Comput. Archit. News*, vol. 39, no. 2, pp. 1–7, May 2011.

[32] J. L. Henning, "SPEC CPU2006 benchmark descriptions," *ACM SIGARCH Comput. Archit. News*, vol. 34, no. 4, pp. 1–17, Sep. 2006.

[33] L. Wei *et al.*, "A 7 Mb STT-MRAM in 22 FFL FinFET technology with 4 ns read sensing time at 0.9 V using write-verify-write scheme and offset-cancellation sensing technique," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 214–216.