Using Magnetic RAM to Build Low-Power and Soft Error-Resilient L1 Cache

Hongbin Sun, Chuanyin Liu, Wei Xu, Jizhong Zhao, *Associate Member, IEEE*, Nanning Zheng, *Fellow, IEEE*, and Tong Zhang, *Senior Member, IEEE*

***Abstract*—Due to its great scalability, fast read access, low leakage power, and nonvolatility, magnetic random access memory (MRAM) appears to be a promising memory technology for on-chip cache memory in microprocessors. However, the write-to-MRAM process is relatively slow and results in high dynamic power consumption. Such inherent disadvantages of MRAM make researchers easily conclude that MRAM can only be used for low-level caches (e.g., L2 or L3 cache), where cache memories are less frequently accessed and slow write to MRAM can be more easily compensated using simple architectural techniques. By developing a hybrid cache architecture, this paper attempts to show that, with appropriate architecture design, MRAM can also be used in L1 cache to improve both the energy efficiency and soft error immunity. The basic idea is to supplement the MRAM L1 cache with several small SRAM buffers, which can substantially mitigate the performance degradation and dynamic energy overhead induced by MRAM write operations. Moreover, the proposed hybrid cache architecture is also an efficient solution to protect cache memory from radiation-induced soft errors, as MRAM is inherently invulnerable to emissive particles. Simulation results show that, with only less than 2% performance degradation, the proposed design approach can reduce the power consumption by up to 76.1% on average compared with the traditional SRAM L1 cache. In addition, the architectural vulnerability factor of L1 data cache is reduced from 28.3% to as low as 0.5%.**

***Index Terms*—Cache, low power, magnetic RAM (MRAM), reliability, soft error.**

I. Introduction

As MAINSTREAM memory technologies including SRAM, DRAM, and flash memories are all facing serious scaling problems, there have been a resurgence of interest in searching for highly scalable universal memory [1]. MRAM is one of the most promising candidates. The basic building block of MRAM is magnetic tunneling junction (MTJ), and the data storage is realized by configuring the resistance of MTJs into one of two possible states (i.e., high-resistance state and low-resistance state). Different from the first-generation MRAM that uses explicitly generated magnetic fields to switch the state of MTJs, a new technique called spin-torque transfer (STT) uses through-MTJ current of spin-aligned electrons to switch the state of MTJ, which has a much greater scalability potential. Hence, this work is only interested in STT MRAM, which is simply called MRAM throughout this paper.

Besides its great scalability, MRAM has several other attractive features, such as short read latency, high density, low leakage power, and nonvolatility. Hence, MRAM has received growing attentions from the computer architecture community. It is promising to design cache with MRAM technology to address the design challenges faced by SRAM cache, such as poor scalability and ever-increasing leakage power. However, one major drawback of MRAM is its high write overhead in terms of latency and power consumption, especially compared with its SRAM counterpart. Hence, a direct use of MRAM to replace SRAM will inevitably incur prohibitive performance degradation and dynamic power increase. The dynamic power increase due to MRAM write may even offset the leakage power saving. Many recent research efforts studied how to design MRAM-based low-level on-chip caches, where hybrid architecture and other techniques are investigated to efficiently compensate the slow and energy-consuming MRAM write operations [2]–[5].

Since L1 cache is accessed much more frequently compared with low-level caches, many researchers tend to easily preclude the possibility of using MRAM in L1 cache memory. In this paper, we explore the feasibility of using MRAM technology to realize low-power L1 cache. In particular, we propose a SRAM-MRAM hybrid L1 cache architecture, where the MRAM cache core is associated with two small SRAM buffers that store recently accessed data blocks. Due to the cache memory access locality, these small SRAM buffers can respond to the majority of L1 cache accesses. As a result, the performance degradation can be substantially mitigated and the overall power consumption, including both dynamic and leakage power consumption, is much lower compared with that of traditional SRAM L1 cache.

Another advantage of MRAM deserving more attention is that MRAM is inherently invulnerable to radiation-induced soft errors [6], because MRAM data storage does not involve electrical charge and hence cannot be corrupted by emissive particles. The soft error invulnerability of MRAM is very valuable as soft error becomes an increasingly crucial issue in high-performance microprocessors. As CMOS process technology continues to scale down, soft error rate is projected to grow rapidly and even multibit soft errors may become inevitable in cache

Manuscript received March 16, 2010; revised July 16, 2010; accepted October 21, 2010. Date of publication December 06, 2010; date of current version December 14, 2011. This work was supported in part by the National Natural Science Foundation of China under Grant 60772096 and by the National Science Foundation for Post-doctoral Scientists of China under Grant 20090461299.

H. Sun, C. Liu, J. Zhao, and N. Zheng are with the Xi'an Jiaotong University, Xi'an, 710049 Shaanxi, China (e-mail: sunsir@mail.xjtu.edu.cn; cyinliu@gmail.com; {zjz@mail.xjtu.edu.cn; nnzheng@mail.xjtu.edu.cn).

W. Xu is with Marvell Technology, Santa Clara, CA 95054 USA (e-mail: weixu@marvell.com).

T. Zhang is with the Electrical, Computer and Systems Engineering Department, Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: tzhang@ecse.rpi.edu).

Digital Object Identifier 10.1109/TVLSI.2010.2090914

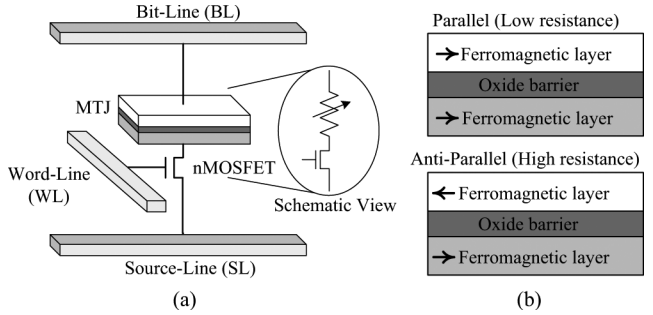


Fig. 1. (a) Structure of a 1T1MTJ MRAM cell. (b) Resistance state of MTJ.

memories at future technology nodes. Traditional cache protection techniques, i.e., error correction codes (ECCs), will be deficient to effectively tolerate multibit soft errors. On the contrary, MRAM cache provides an efficient solution to protect cache memory from soft errors. In this paper, we investigate the overall soft error immunity improvement of the proposed SRAM–MRAM hybrid L1 cache by using the architectural vulnerability factor analysis.

The effectiveness of the proposed SRAM–MRAM hybrid L1 cache architecture and its soft error immunity improvement has been demonstrated through system-level simulations. The SimpleScalar and Cacti tools are used for processor simulation and cache memory modeling at 65-nm technology node. Results show that the proposed hybrid architecture can reduce the overall power consumption by up to 76.1% on average with less than 2% performance degradation. In addition, compared with the SRAM cache, the proposed hybrid cache can reduce the architecture vulnerability factor from 28.3% to as low as 0.5%.

The remainder of this paper is organized as follows. Section II briefly reviews the background of MRAM and soft errors. Section III presents the proposed hybrid L1 cache architecture and discusses its soft error immunity. Section IV describes our experimental methodology, including simulator, benchmarks, and memory modeling tool, and Section V presents the simulation results. Finally, conclusions are drawn in Section VI.

II. Background

A. MRAM Basics

The new-generation MRAM technology has a much greater scalability and, hence, has received growing interest. Fig. 1(a) shows the typical 1T1MTJ MRAM cell structure. Each cell contains one MTJ as the storage element and one nMOS transistor as the access control device. As the basic storage element in MRAM, each MTJ has two ferromagnetic layers separated by one oxide barrier layer. The resistance of each MTJ depends on the relative magnetization directions of the two ferromagnetic layers, i.e., when the magnetization is parallel (or antiparallel), MTJ is in a low- (or high-) resistance state, as illustrated in Fig. 1(b). In MRAM, parallel and anti-parallel magnetization are realized by steering a write current directly through MTJs along opposite directions.

During write or read operations, the bit-line (BL) and source-line (SL) establish appropriate voltage drop across the cell, and the word-line (WL) turns on/off the nMOS transistor to realize memory cell access control. One important MTJ device parameter in MRAM is its write current threshold: To successfully switch the MTJ resistance state, the through-MTJ write current not only has to be larger than the write current threshold but also should sustain for a certain amount of time, called MTJ switching time. Therefore, the time to write to MRAM cells is much longer than for SRAM and consumes much more energy, which consequently limits the use of MRAM as the embedded memory in microprocessors to some extent.

Nevertheless, MRAM has several attractive features compared with SRAM, which may make a significant impact on the future processor design. MRAM has much higher storage density and greater scalability potential, which means that more memory can be integrated on-chip. MRAM consumes much less standby leakage power due to its nonvolatile nature, which is very attractive, especially when considering that leakage power will get increasingly larger as the CMOS process technology continues to scale down. Attracted by the above two advantages, several recent research efforts have explored the potential of using MRAM to design on-chip low-level cache memory [2]–[5], where several architectural approaches have been proposed to effectively compensate the slow and energy-consuming write operation of MRAM.

Another important advantage is that MRAM is inherently invulnerable to radiation-induced soft errors as data stored in MRAM is realized by MTJ instead of electrical charges [6]. Emissive particles such as alpha particles and high-energy neutrons are unable to change the magnetization directions in MTJ, and hence do not threaten the data integrity. Therefore, it is possible to leverage MRAM to achieve a substantially improved soft error immunity in cache memory.

B. Soft Error in Cache Memory

Over the past decades, industry has encountered with many soft-error-induced system crashes. Studies [7] have found that soft error in semiconductor devices is caused by two primary radiation sources: alpha particles and high-energy neutrons from cosmic radiation. When energetic particles, i.e., alpha and neutron particles, pass through a semiconductor device, they generate electron–hole pairs, which can be collected by transistor gate and diffusion nodes. A sufficient amount of accumulated charge may invert the state of a logic device thereby introducing a logical fault into the circuit's operation. At sea level, alpha particle is the major cause of the total transient failures [8]. Fortunately, not all of the transient faults will show up as architecture errors. The possibility that a raw fault will result in a user-visible error can be estimated by computing the architectural vulnerability factor (AVF) [9], [10].

As semiconductor process technology continues to scale down, the soft-error rate in future microprocessors is expected to grow rapidly. Cache memories constitute a significant portion of the transistor budget in current microprocessors, thereby playing a key role in processor reliability. Previous studies have concluded that unprotected memory elements are the most vulnerable soft errors in current systems. Conventionally, error-detection codes and ECCs are the most prevalent soft-error protection techniques widely used in modern

processors [11]. Regarding cache memory, single-error-correction/double-error-detection (SEC-DED) code appears to be the best choice due to its acceptable decoding latency and appropriate error-correction capability [12], [13]. Nevertheless, researches have shown that implementing ECC protection circuits in caches already increases the cache access time by up to 95% [14] and power consumption by up to 22% [15].

Recent works have shown that the multibit soft errors may become inevitable at future technology nodes, which clearly makes the soft error problem much worse [16], [17]. Multibit soft errors can be both temporal and spatial. Temporal multibit soft errors result from multiple independent upsets over the time and can be mitigated by periodically scrubbing the cache memory at the cost of extra bandwidth occupation. While the spatial multibit soft errors occur when a high-energy particle strikes the circuit causing several transistors flip at the same time. Although bit-interleaving with ECC protection seems to be a natural solution for spatial multibit soft errors, it can be very expensive in terms of design cost and energy dissipation. To address the multibit soft-error issue, several other approaches have been proposed to improve the reliability at the penalty of performance degradation and/or extra overhead [18]–[20]. As a result, how to efficiently tolerate multibit soft errors in cache memory still remains as an open question.

III. Proposed Hybrid L1 Cache Architecture

Here, we present the proposed cache architecture that enables the efficient use of MRAM in L1 cache memory to improve the energy efficiency and soft error immunity. Fig. 2 illustrates this proposed SRAM-MRAM hybrid L1 cache architecture, which has the following key features.

1) The proposed L1 cache basically follows the current cache design practice (e.g., see [12]), while the underlying difference is that the main component (L1 cache core) in this work is designed with MRAM technology.
2) Two small SRAM buffers, i.e., the *filter buffer* and *victim buffer*, are located in front of the MRAM cache core and hold the recently accessed data blocks. These SRAM buffers are used to reduce the accesses to MRAM cache core, especially for the slow and energy-consuming write operations.
3) The soft-error immunity of the overall L1 cache architecture can be significantly improved, as the great majority of data blocks are stored in the MRAM cache core and MRAM is inherently invulnerable to radiation-induced soft errors.

A. SRAM-MRAM Hybrid L1 Cache Design

The MRAM cache core follows the same design practice as SRAM cache, which consists of subarray, H-tree routing, WL, BL, and sense amplifier, etc., except that each memory cell is fabricated as MRAM cell instead of SRAM cell. Hence, MRAM cache can be operated similarly to its SRAM counterpart. The main drawback of MRAM cache is its long latency and high energy consumption during the write operations. As a result, data update due to cache writes and misses will occupy the data bus for a relatively long period and, hence, block other accesses to L1 cache. Moreover, frequent data update in MRAM cache will also dramatically increase dynamic energy dissipation. Therefore, a direct use of MRAM as L1 cache memory will inevitably result in significant performance degradation and energy consumption increase.

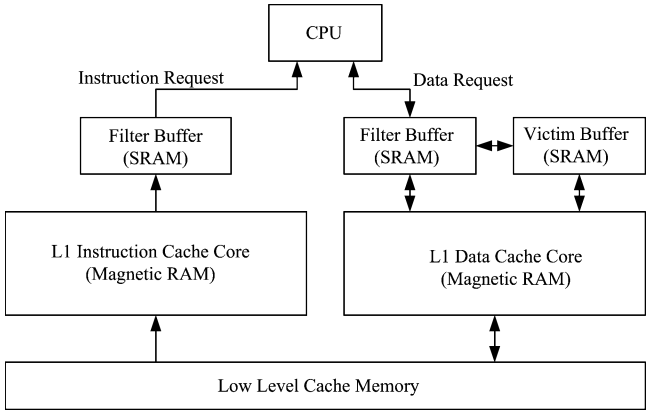


Fig. 2. Illustration of the proposed SRAM-MRAM hybrid L1 cache architecture.

In this work, we propose to mitigate the penalty by leveraging the access locality of L1 cache memory. We first supplement the MRAM cache core with a small direct-mapped SRAM buffer to store the recently used data blocks. Due to the cache access locality, this SRAM buffer can respond to most of the cache accesses, while accesses to MRAM cache core only occur when this small buffer misses. As a result, writes to MRAM cache core can be substantially reduced, and the performance degradation and energy consumption increase will be accordingly mitigated as well. This idea is similar to those techniques proposed for reducing cache energy consumption [21]. In this paper, we refer this small SRAM buffer as *filter buffer (FB)*.

An FB may be sufficient for L1 instruction cache: as the processor core does not write to the instruction cache, accesses only suspend when new data blocks are being loaded due to instruction cache misses, while for data cache, a single filter buffer may be inadequate, because MRAM writes also occur when processor core writes data cache besides the data cache misses. Conflicted dirty data blocks tend to be frequently moved between the FB and MRAM cache core, due to the direct-mapped nature of the FB. To improve the performance of the direct-mapped filter buffer, we propose to use another small fully associative SRAM buffer only for data cache, which is referred to as *victim buffer (VB)* [22]. Data blocks in the victim buffer employ the least recently used (LRU) policy for replacement when it is full. By leveraging this small fully associative victim buffer, data block conflicts in direct-mapped filter buffer can be largely mitigated.

By supplementing the MRAM cache core with the small SRAM filter and victim buffers, we can largely mitigate the impact of slow and energy-consuming MRAM write on the performance of overall L1 cache hierarchy. Fig. 3 shows the operation flow of the proposed hybrid L1 data cache described above. The operation flow of the instruction cache is less complicated than data cache, as instruction cache does not employ victim buffer. We note that the size of the filter and victim

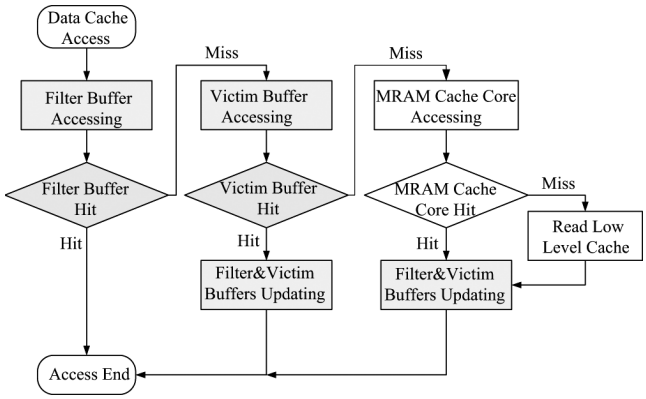


Fig. 3. Operation flow chart of hybrid L1 data cache.

buffers is quite small and hence they incur relatively small overhead in terms of area cost and leakage energy consumption. Furthermore, since they can respond to a great majority of L1 cache accesses, the overall SRAM–MRAM hybrid cache will achieve a comparable performance to SRAM cache. Moreover, as the hybrid cache consumes much less leakage energy, the overall L1 cache design is substantially low power, which will be further demonstrated by experimental results.

B. Soft-Error Immunity Improvement

As pointed out earlier, not all of the particle-induced data corruption in cache memory will show up as user-visible architecture errors. The possibility that a raw fault will result in an user-visible error can be estimated by computing the AVF [9], [10]. A structure's AVF is estimated by computing the ACE (required for architecturally correct execution) and un-ACE (unnecessary for ACE) intervals for each single bit. A bit is un-ACE for any interval where its value can be changed without affecting the program's final outcome. Any interval that cannot be proven un-ACE is assumed to be ACE. Fig. 4 illustrates an example of a bit's lifetime in a write-back data cache. The AVF for a single-bit storage cell is simply the fraction of time that it holds ACE state. Assuming that all cells have equal raw fault rates, the AVF of a structure can be computed by averaging the individual AVFs of all of its storage cells. The detailed analysis of AVF can be found in [10].

As explained earlier, MRAM is invulnerable to radiation-induced soft errors. Therefore, the proposed SRAM-MRAM hybrid hierarchy can naturally reduce the architectural vulnerability factor thereby improving the reliability against soft errors. The vulnerability of a cache block tends to dynamically change according to its location. When a cache block locates within SRAM buffers, it is vulnerable to soft errors; while it becomes invulnerable when being moved back into the MRAM cache core. For those cache blocks located within MRAM cache core, they do not contribute to the AVF anymore even if they are in ACE state. Taking Fig. 5, for example, the data bit is filled into the FB and turned into ACE status after being written, and it is vulnerable due to its SRAM buffer location. Nevertheless, the data bit becomes invulnerable when the latter data access evicts this data bit into the MRAM cache core. In this case, AVF estimation of the proposed L1 cache hierarchy should take into

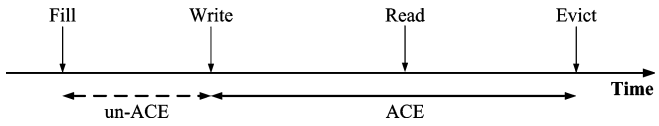


Fig. 4. Example activities in SRAM write-back data cache during a bit's lifetime.

account this vulnerable–invulnerable transition. Therefore, the AVF of this SRAM–MRAM hybrid cache should be computed by statistically recording the fraction of time that data bit holds ACE state and meanwhile stays in vulnerable memory location, i.e., SRAM buffers.

In the proposed L1 MRAM cache memory, the size of SRAM buffers is quite small and only stores the recently active data blocks. Mostly, data blocks that are less recently used will be evicted from SRAM buffers to MRAM cache core and stay inactive or even dead for a relatively long time before being evicted from the L1 cache. As a result, the proposed hybrid cache can achieve a substantially reduced AVF and hence an improved soft error immunity. This is extremely attractive when multibit soft error rate is expected to grow rapidly, and traditional error-correction techniques will be deficient to effectively address this issue in the future.

IV. Evaluation Methodology

To evaluate the performance of the above presented hybrid L1 cache architecture, we carry out simulations using the popular *SimpleScalar 3.0* simulator.[1] We assume that the entire design is implemented at the 65-nm technology node and the processor works at 2.6-GHz frequency. Table I lists the simulator configuration parameters. We use cache configurations that are similar to Intel Core 2 architecture: a 32-kB, 8-way, 64-byte block L1 instruction and data cache, and a 2-MB, 16-way, 64-byte block L2 unified cache in our study. The L1 data cache and L2 cache are set as write-back mode. The basic cache hierarchy is assumed to be implemented with SRAM technology, and the characteristics of its cache memory modules are estimated by using Cacti 6.0, the latest version of a widely used cache modeling tool Cacti [23]. The modeling results are listed in Table II. We note that the access latency of L2 cache is assumed to be constant no matter whether L1 cache is SRAM or MRAM, as this work mainly focuses on the L1 cache while L2 cache is applied with the same configuration to demonstrate the performance of different L1 cache architecture more clearly.

To fully evaluate the performance of the proposed design solution, MRAM cache modeling is necessary. In this work, we develop a MRAM cache modeling tool by accordingly modifying Cacti. As MRAM and SRAM cells have similar electrical interfaces from circuit designer's point of view, we assume that the MRAM cache follows the same memory organization as that of SRAM cache, where a large MRAM array consists of several small sub-arrays and each subarray is connected by H-tree routing. We apply the traditional SRAM cache structure to each subarray, while let Cacti to automatically choose its size-related parameters to achieve an optimal design. To obtain the timing and energy models of the MRAM cache, we use the similar

[1][Online]. Available: http://www.simplescalar.com.

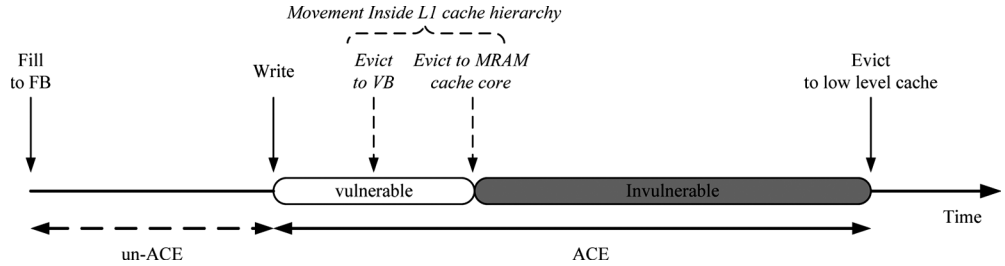


Fig. 5. Example activities in proposed hybrid write-back data cache during a bit's lifetime.

TABLE I
DEFAULT CONFIGURATION PARAMETERS USED IN SIMPLESCALAR

Configuration Parameters	Value
Processor	
Frequency	2.6 Ghz
Functional Units	4 integer ALUs 1 integer multiplier/divider 4 FP ALUs 1 FP multiplier/divider
LSQ Size	8 Instructions
RUU Size	16 Instructions
Fetch/Decode/Issue/Commit Width	4/4/4/4 Instructions/cycle
Fetch Queue Size	4 Instructions
Branch Logic	
Predictor	combined, bimodal 2KB table two-level 1KB table 8 bit history
BTB	512 entry, 4-way
Miss-prediction Penalty	3 cycles
Cache and Memory Hierarchy	
L1 Instruction Cache	32KB, 8-way, 64-byte blocks 2 cycle latency
L1 Data Cache	32KB, 8-way, 64-byte blocks 2 cycle latency
L2 Cache	2MB unified, 16-way 64-byte blocks 10 cycle latency
Main Memory	500 cycle latency

method and part of simulation results that are presented in [2]. The modified MRAM Cacti gives the modeling results as expected, which can be summarized as follows.

- MRAM cache exhibits unbalanced read and write latency, i.e., read latency is comparable to SRAM while write latency tends to be longer than 10 ns.
- Write to MRAM cache tends to consume an extraordinary large dynamic energy, which is dominated by the energy dissipation on MRAM cells.
- The leakage power of the overall MRAM cache is quite low as the leakage energy only dissipated on the peripheral circuits while the MRAM cells do not consume any leakage energy.

We use the entire integer and floating point benchmarks of the SPEC2000 suite[2] in our simulations. For each benchmark, we use *ref* inputs, fast-forward one billion instructions, and run for one billion instructions. During the fast-forward phase, we warm up both L1 and L2 caches. The performance and energy efficiency of the proposed hybrid L1 cache are fully evaluated by comparing with two baseline design, i.e., the traditional SRAM L1 cache and the pure MRAM L1 cache.

We use the AVF to evaluate the soft-error immunity improvement of the proposed hybrid L1 cache hierarchy against the traditional SRAM cache. AVF analysis determines the fraction of time that cache blocks are not affected by radiation particles. Thus, our results do not reflect failure rates, but rather show the average fraction of time when a structure is architecturally vulnerable to soft error. We adopt the methodology of Biswas *et al.* [10] to measure AVF for the proposed SRAM-MRAM hybrid L1 cache. This technique marks events in the life of each cache block (e.g., fill, read, write, evict). Between any two events, we annotate the time as either vulnerable-required for architecturally correct execution (ACE) or not vulnerable (unACE). The AVF for hybrid L1 cache is the average fraction of the cache in both the ACE state and vulnerable location over the entire execution. Our experiments for AVF use traces of memory references from the same SPEC CPU2000 benchmark suite as we evaluate the performance. All of the traces are also collected by using the modified *SimpleScalar 3.0* simulator.

[2]Standard Performance Evaluation Corporation. [online]. Available: http://www.spec.org

V. EVALUATION RESULTS

A. Performance Comparison and Overhead Expense

Table II shows a complete comparison between SRAM and MRAM caches. As expected, the write operation of the MRAM cache is almost 18 times longer and consumes 27 times more dynamic energy compared with its SRAM counterpart. Therefore, a direct use of MRAM as L1 cache will inevitably incur a significant performance degradation and unacceptable energy consumption increase. Figs. 6 and 7 illustrate the performance degradation and dynamic power increase by using an MRAM-only L1 cache for all the benchmarks in the SPEC2000 suite. For the majority of benchmarks, the instruction per cycle (IPC) performance is reduced by more than 40% of that of traditional L1 SRAM cache. And the dynamic power of MRAM data cache memory also increases dramatically, e.g., for eight benchmarks the dynamic power is as high as nine times larger than that of SRAM data cache. The dynamic power of MRAM instruction cache memory seems less serious, as illustrated in Fig. 7, however MRAM instruction cache substantially contributes to the performance degradation. Hence, it is infeasible to directly replace SRAM with MRAM in L1 cache memory.

We then investigate the performance of the proposed hybrid L1 cache memory. First, we have to choose the appropriate size for filter and victim buffers and explore how the size affects the cache performance. For the proposed hybrid L1 cache, filter and victim buffer miss may incur significant access latency. The latency should include the access time of both SRAM buffers and MRAM caches. Taking L1 data cache for example, the worst-case access latency can be as long as 1.27 and 11.29 ns for read and write, respectively. Therefore, it is critical for the proposed hybrid L1 cache to avoid the occurrence of such a worst-case

TABLE II
CACTI REPORT FOR L1 SRAM CACHE, L1 MRAM CACHE, FB, AND VICTIM BUFFER AT 65-nm NODE

Memory	Size	Associativity	Block size	Access time (ns)		Dynamic Energy (nJ)		Leakage Power (mW)	Area (mm^2)
				Read	Write	Read	Write		
L1 I/D SRAM cache	32KB	8	64-byte	0.59	0.59	0.041	0.041	85	0.468
Filter buffer	2KB	direct-map	64-byte	0.38	0.38	0.004	0.004	4.9	0.028
Victim buffer	512B	full	64-byte	0.33	0.33	0.007	0.007	1.3	0.008
L1 I/D MRAM cache	32KB	8	64-byte	0.56	10.58	0.040	1.109	1.9	0.192

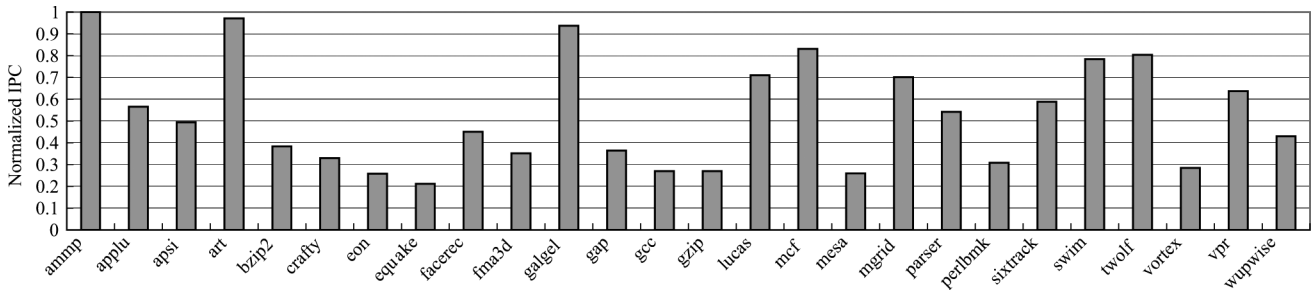


Fig. 6. Normalized IPC performance of the direct use of MRAM as L1 data and instruction cache.

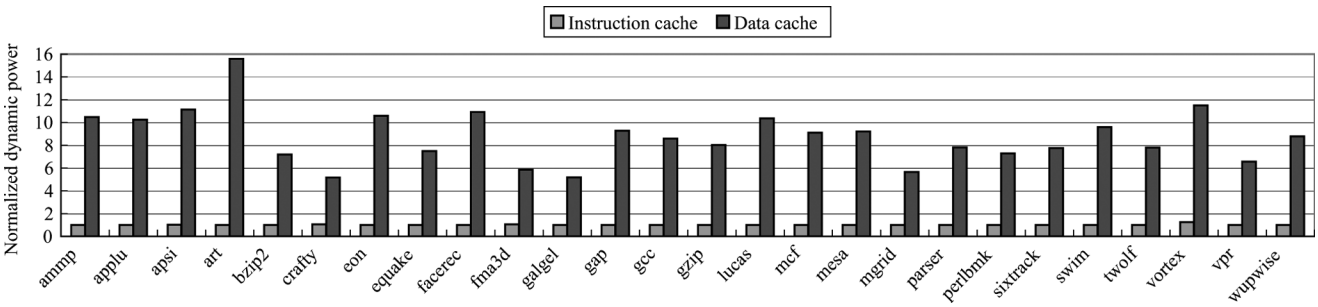


Fig. 7. Normalized dynamic power consumption of the direct use of MRAM as L1 data and instruction cache.

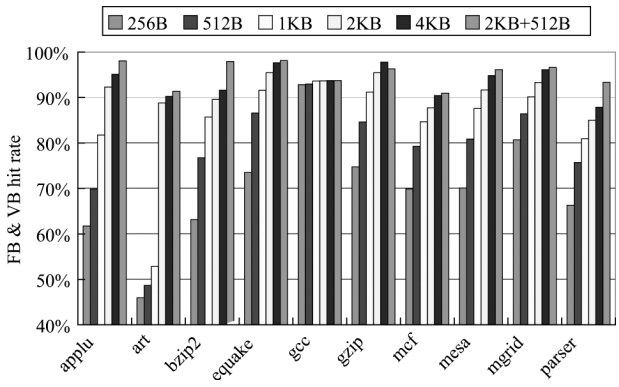


Fig. 8. FB hit rates for FBs of data cache with the size varying from 512 B to 4 kB.

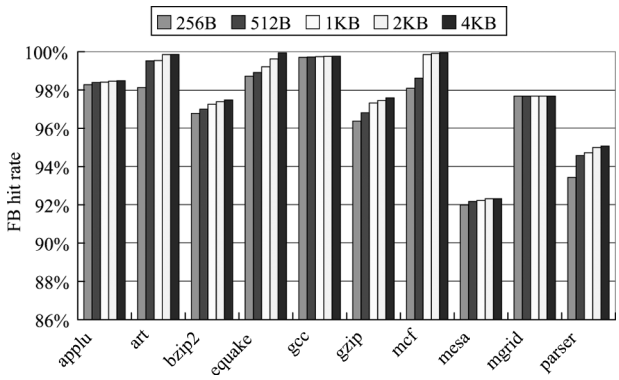


Fig. 9. FB hit rates for FBs of instruction cache with the size varying from 512 B to 4 kB.

scenario as much as possible. By setting L1 data cache core to be 32 KB, eight-way associative as shown in Table I, we vary the size of the filter buffer from 512 B up to 4 kB. Fig. 8 shows the FB hit rate for 11 representative benchmarks. Clearly, the FB hit rate increases with the increase of the FB size. With the size of 2 kB and higher, the hit rate can be larger than 91.06% on average, which means that most of explicit accesses to MRAM cache core can be avoided. Furthermore, for a 2-kB filter buffer, we find that an additional 512-B victim buffer can improve the hit rates for another 5.2%. The above results suggest that a 2-kB filter buffer associated with a 512-B victim buffer appears to be an appropriate choice for L1 MRAM data cache core in this experiment setup. For instruction cache, we also choose 2 kB as the filter cache size as illustrated in Fig. 9. In addition, we conduct experiments to examine the sensitivity of the proposed architecture to different configurations. Tables III and IV show the average FB hit rates with the size of L1 instruction and data caches varying from 16 to 64 kB and number of sets varying from two-way to eight-way, respectively. As the cache size and

TABLE III
FILTER AND VICTIM BUFFERS' SENSITIVITY TO CACHE SIZE

	Instruction cache			Data cache		
	16KB	32KB	64KB	16KB	32KB	64KB
Hit rate	98.62%	98.62%	98.63%	96.25%	96.26%	96.25%

TABLE IV
FILTER AND VICTIM BUFFERS' SENSITIVITY TO CACHE SET ASSOCIATIVITY

	Instruction cache			Data cache		
	2-way	4-way	8-way	2-way	4-way	8-way
Hit rate	98.62%	98.62%	98.63%	96.26%	96.26%	96.25%

TABLE V
FILTER AND VICTIM BUFFERS' SENSITIVITY TO ISSUE WIDTH

	Buffer for inst. cache			Buffers for data cache		
	2-issue	4-issue	8-issue	2-issue	4-issue	8-issue
Hit rate	98.59%	98.63%	98.63%	96.47%	96.25%	96.21%

set number increase, the average hit rate remains almost constant. We also carry out simulations for different issue width in Table V, as issue width may change the required capacity of cache memory. Even at the worst case (eight-issue), it can still maintain a high hit rate above 98% and 96%, respectively. The above results indicate that the proposed filter-and-victim buffers approach is robust to different L1 cache and pipeline configurations.

As filter and victim buffers can largely avoid the majority of accesses to L1 MRAM cache core, the performance degradation due to the write latency of MRAM will be substantially mitigated. Fig. 10 shows the normalized IPC of the proposed hybrid L1 cache against SRAM cache. For most benchmarks, the IPC performance can reach up to 95% of that of L1 SRAM cache. For those benchmarks such as ammp, applu, etc., the IPC performance can be even better. On average, the IPC performance of the proposed hybrid L1 cache is as high as 98% of that of L1 SRAM cache. This means that the filter and victim buffers are substantially effective in improving the overall performance of the proposed hybrid cache design. Moreover, by using the filter and victim buffers, the proposed approach can realize a low power cache design as well. As shown in Fig. 11, the dynamic power of hybrid data cache is dramatically reduced for all benchmarks, compared with the simulation results in Fig. 7. For most of the benchmarks, the dynamic energy consumption is comparable with that of SRAM cache. For a few benchmarks the normalized dynamic power seems not as low as we expect, which is mainly because the L1 cache in those benchmarks is accessed relatively infrequently and hence the access locality is poor. However, considering that MRAM consumes much less leakage energy than SRAM, the overall energy consumption of the proposed hybrid L1 cache will be much lower. Fig. 12 illustrates the normalized power consumption of the proposed hybrid L1 cache compared with that of SRAM cache, and the power consumption includes both the leakage and dynamic power. For most benchmarks, the overall power consumption of the proposed hybrid cache is less than 30% of that of SRAM cache.

Finally, the storage density of MRAM is higher than SRAM and fabricating MARM only incurs three more mask layers to embed MTJ devices on logic dies [24]. According to the Cacti modeling results listed in Table II, the silicon area of the MRAM cache core, filter buffer and victim buffer is less than 50% of that of SRAM cache. This clearly suggests that the proposed design is area efficient.

B. Architectural Vulnerability Factor Improvement

As data cache requires more sophisticated protection techniques compared with instruction cache, our AVF analysis only focuses on data cache. We also investigate the architectural vulnerability factor of the hybrid L1 data cache using three alternative designs: write-back with noninterleaved SECDED ECC ("A"), write-back with interleaved parity ("B"), and the proposed SRAM-MRAM hybrid write-back ("C"). We assume the number of bit flips due to soft errors may be larger than two, and are hereby not detectable by SECDED ECC. And in design "B", we assume the interleaved parity is sufficient to detect but unable to correct any multibit error. We assume that the cache can always obtain the correct value from L2 or the rest of the memory system when a raw fault is detected in clear blocks (i.e., treat the access as an L1 miss), and error only occurs in dirty blocks where it is detectable but uncorrectable. In design "C," we assume that the SRAM buffers used in the proposed design do not have any protection against soft error. We note that implementing interleaved ECC in cache memory will incur noticeable energy cost, and thereby may be not a feasible option to protect L1 cache [18]. The interleaved parity code tends to be an efficient choice, it however is unable to correct any detected multibit faults.

Three different categories of events are considered when analyzing the AVF of L1 data cache, given here.

- "$\mathrm{F/R} \Rightarrow \mathrm{R}$" corresponds to the time between a cache line fill or read and a subsequent read with no intervening events.
- "$\mathrm{W} \Rightarrow \mathrm{R}$" corresponds to the time between a cache line write and a subsequent read with no intervening events.
- "$\mathrm{X} \Rightarrow \mathrm{E}$" corresponds to any event('X') that precedes an eviction of a dirty block (i.e., write-back to L2 cache).

We note that, when analyzing the AVF of the proposed hybrid data cache, we only consider the time between two events occur in the SRAM buffers while data blocks in the MRAM cache core are assumed to be invulnerable to soft error.

Fig. 13 illustrates part of the resulting AVF under the multibit fault model. For write-back with noninterleaved SECDED ECC and proposed hybrid write-back, the reported AVF corresponds to silent data corruption (SDC) AVF. While for the write-back with interleaved parity, the reported AVF is for detectable but uncorrectable errors (DUEs). From Fig. 13, we first observe that the write-back data cache with noninterleaved SECDED ECC is extraordinary vulnerable to multibit soft errors that its AVF can be as high as 28.3% on average for all benchmarks. This largely agrees with the belief that traditional SECDED ECC is no longer sufficient for protecting cache memory in case of multibit soft error. The AVF of write-back with interleaved parity is much better than that of write-back with noninterleaved SECDED ECC. However, it is still unsatisfactory, as it leaves the dirty blocks unprotected. The AVF of the proposed hybrid L1 data cache hierarchy is as low as 0.5% on average. This is mainly because data blocks are only active and stay at SRAM buffers for a small fraction of their total lifetime in the L1 cache, and

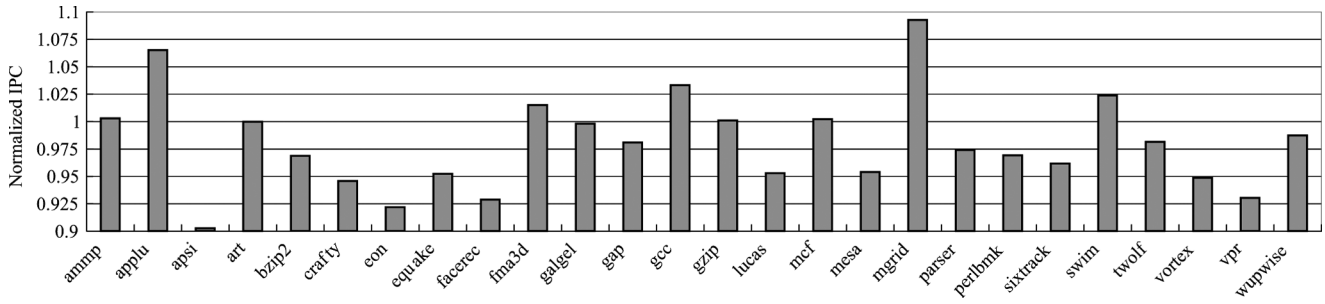


Fig. 10. Normalized IPC performance of the proposed hybrid L1 cache.

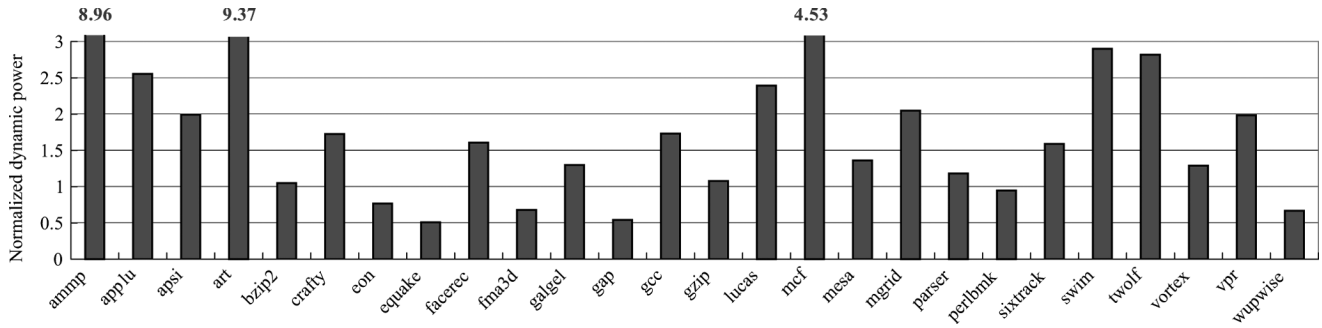


Fig. 11. Normalized dynamic power of the proposed hybrid L1 data cache.

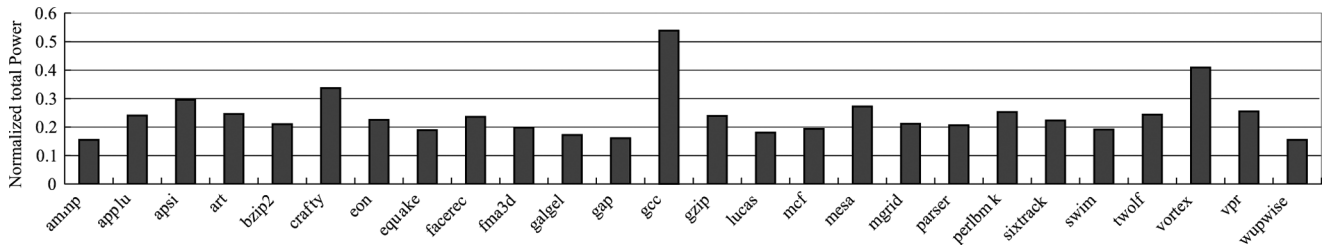


Fig. 12. Normalized power consumption of the proposed hybrid L1 cache, including both leakage and dynamic power of L1 instruction and data caches.

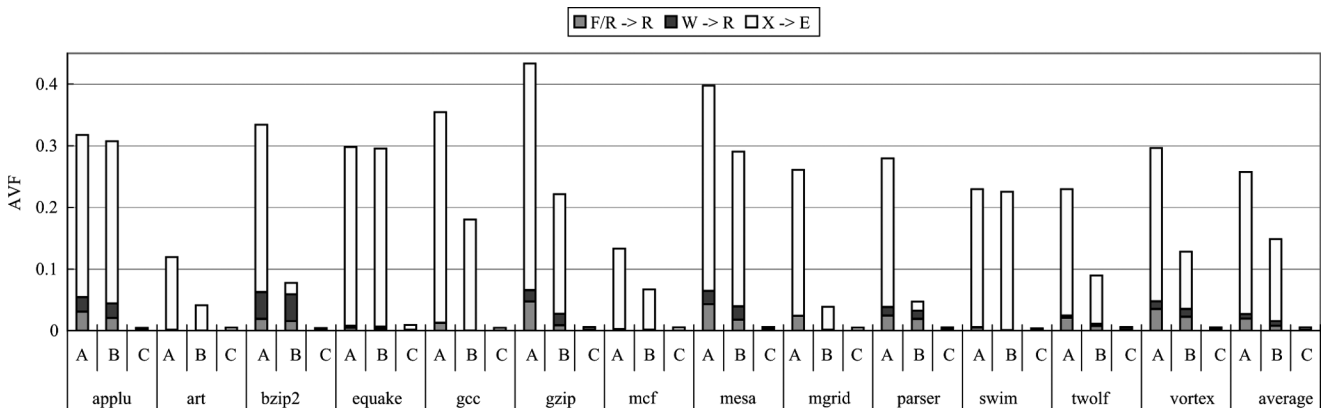


Fig. 13. Architectural vulnerability factor of three alternative cache design. A: write-back with noninterleaved SECDED ECC. B: write-back with interleaved parity. C: proposed hybrid write-back.

most of the time they stay at MRAM cache core safely. Moreover, if we protect our SRAM buffers with interleaved parity, the AVF can be even lower. The simulation results clearly suggest that the proposed hybrid cache architecture is a efficient design approach to effectively protect cache memory from radiation-induced soft errors.

VI. Conclusion

MRAM is a promising memory technology that has gained increasing attention from the computer architecture community. However, one major obstacle hindering MRAM from being used as on-chip cache memory is that the write operation to MRAM

is very slow and tends to consume significant dynamic energy, especially compared with its SRAM counterpart. Hence, a direct use of MRAM in cache memory will inevitably incur prohibitive performance degradation and dynamic power consumption increase. This paper explores the feasibility and potential of using MRAM in an L1 cache to improve the energy consumption efficiency and reliability. We propose an SRAM–MRAM hybrid cache architecture that, by supplementing MRAM cache with several small buffers, the slow and energy-consuming write operations can be well compensated. Due to the inherent soft-error invulnerability of MRAM, the proposed hybrid cache can substantially improve the soft-error immunity of cache architecture as well.

The proposed hybrid cache architecture has been extensively evaluated by using popular processor simulation and memory modeling tools. Results show that, compared with a traditional SRAM L1 cache, this proposed design solution is able to reduce the area overhead and overall power consumption by up to 50% and 76.1% on average, respectively, while only incurring less than 2% performance degradation. Moreover, its architecture vulnerability factor is only as low as 0.5%. Compared with the architecture vulnerability factor of SRAM cache, which is as high as 28.3%, it clearly suggests that the proposed design approach is an attractive option to address multibit soft errors at future technology nodes.

REFERENCES

[1] K. Kim and G. Jeong, "Memory technologies for sub-40 nm node," in *Proc. IEDM*, Dec. 2007, pp. 27–30.

[2] X. Dong, X. Wu, G. Sun, Y. Xie, H. Li, and Y. Chen, "Circuit and microarchitecture evaluation of 3-D stacking magnetic RAM (MRAM) as a universal memory replacement," in *Proc. IEEE Int. Conf. DAC*, Jun. 2008, pp. 554–559.

[3] G. Sun, X. Dong, Y. Xie, J. Li, and Y. Chen, "A novel architecture of the 3-D stacked MRAM L2 cache for CMPs," in *Proc. IEEE Int. Symp. High Performance Comput. Architecture*, Jun. 2009, pp. 239–249.

[4] W. Xu, H. Sun, X. Wang, Y. Chen, and T. Zhang, "Design of last-level on-chip cache using spin-torque transfer RAM (STT RAM)," *IEEE Trans. Very Large Scale Integration (VLSI) Syst.*, no. 99, pp. 1–11, 2009.

[5] X. Wu, J. Li, L. Zhang, E. Speight, R. Rajamony, and Y. Xie, "Hybrid cache architecture with disparat memory technologies," in *Proc. IEEE Symp. Comput. Architecture*, Jun. 2009, pp. 34–45.

[6] K. Itoh, "Trends in low-voltage embedded-RAM technology," in *Proc. 23rd Int. Conf. Microelectron.*, May 2002, pp. 497–501.

[7] R. C. Baumann, "Soft error in advanced semiconductor devices part I: The three radiation sources," *IEEE Trans. Device Mater. Reliabil.*, vol. 1, no. 1, pp. 17–22, Mar. 2001.

[8] T. Heijmen, P. Roche, G. Gasiot, and K. R. Forbes, "A comparative study on the soft-error rate of flip-flops from 90-nm production libraries," in *Proc. IEEE 44th Annu. Int. Reliabil. Phys. Symp.*, 2006, pp. 204–211.

[9] S. S. Mukherjee, C. Weaver, J. Emer, S. K. Reinhardt, and T. Austin, "A systematic methodology to compute the architectural vulnerability factors for a high-performance microprocessor," in *Proc. 36th Annu. Int. Symp. Microarchitecture (MICRO)*, 2003, pp. 29–41.

[10] A. Biswas, P. Racunas, R. Cheveresan, J. Emer, S. S. Mukherjee, and R. Rangan, "Computing architectural vulnerability factors for address-based structures," in *Proc. 32nd Int. Symp. Comput. Architecture (ISCA)*, Jun. 2005, pp. 532–543.

[11] C. L. Chen and M. Y. Hsiao, "Error-correcting codes for semiconductor memory applications: A state-of-the-art review," *IBM J. Res. Devel.*, vol. 28, no. 2, pp. 124–134, Mar. 1984.

[12] J. L. Hennessy and D. A. Patterson, *Computer Architecture a Quantitative Approach*, 4th ed. San Mateo, CA: Morgan Kaufmann, 2006.

[13] N. N. Sadler and D. J. Sorin, "Choosing an error protection scheme for a microprocessor's L1 data cache," in *Proc. IEEE Int. Conf. Comput. Design (ICCD)*, Oct. 2006, pp. 499–505.

[14] J.-F. Li and Y.-J. Huang, "An error detection and correction scheme for RAMs with partial-write function," in *Proc. IEEE Int. Workshop Memory Technol., Design Testing (MTDT)*, 2005, pp. 115–120.

[15] R. Phelan, "Addressing soft errors in ARM core-based designs," Tech. Rep. ARM, 2003.

[16] J. Maiz, S. Hareland, K. Zhang, and P. Armstrong, "Characterization of multi-bit soft error events in advanced SRAMs," in *Proc. IEDM*, 2003, pp. 21.4.1–21.4.4.

[17] K. Osada, K. Yamaguchi, Y. Saitoh, and T. Kawahara, "SRAM immunity to cosmic-ray-induced multierrors based on analysis of an induced parasitic bipolar effect," *IEEE J. Solid-State Circuits*, vol. 39, no. 5, pp. 827–833, May 2004.

[18] B. Gold, M. Ferdman, B. Falsafi, and K. Mai, "Mitigating multi-bit soft errors in L1 caches using last-store prediction," in *Proc. Int. Workshop Architectural Support for Gigascale Integration*, Jun. 2007, pp. 11–18.

[19] J. Kim, N. Hardavellas, K. Mai, B. Falsafi, and J. C. Hoe, "Multi-bit error tolerant caches using two-dimensional error coding," in *Proc. 40th Annu. ACM/IEEE Int. Symp. Microarchitecture (MICRO)*, 2007, pp. 197–209.

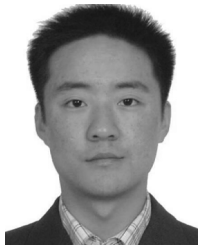
[20] P. Reviriego and J. A. Maestro, "Efficient error detection codes for multiple-bit upset correction in SRAMs with BICS," *ACM Trans. Design Autom. Electron. Syst.*, vol. 14, no. 1, pp. 18–28, Jan. 2009.

[21] J. Kin, M. Gupta, and W. H. Mangione-Smitch, "Filtering memory references to increase energy efficiency," *IEEE Trans. Computers*, vol. 49, no. , pp. 1–15, Jan. 2000.

[22] N. P. Jouppi, "Improving direct-mapped cache performance by the addition of a small fully-associative cache and prefetch buffers," in *Proc. 17th Annu. Int. Symp. Comput. Architecture*, Jun. 1990, pp. 364–373.

[23] "CACTI: An integrated cache and memory access time, cycle time, area, leakage, and dynamic power model," [Online]. Available: http://www.hpl.hp.com/research/cacti/

[24] E. J. Marinissen, B. Prince, D. Keltel-schulz, and Y. Zorian, "Challenges in embedded memory design and test," *Proc. DATE*, pp. 722–727, Mar. 2005.

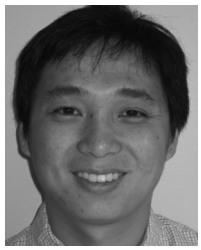


Hongbin Sun received the B.S. and Ph.D. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2003 and 2009, respectively.

Currently, he holds a post-doctoral posotion with the School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, China. His current research interests include VLSI architecture for digital video processing, computer memory system, and signal processing system design for new memory technology.

Chuanyin Liu received the B.S. degree in computer science and technology from Xi'an Jiaotong University, Xi'an, China, in 2008, where he is currently working toward the M.S. degree.

His current research interests include low-power embedded processors and fault-tolerant computer architecture.



Wei Xu received the B.S. and M.S. degrees from Fudan University, Shanghai, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, in 2009.

He joined the Marvell Technology, Santa Clara, CA, in 2009, as a Senior Design Engineer. His research interests include circuit, signal processing, and system design for memory and data storage systems, including phase change memory, spin-transfer torque magnetoresistive memory, NAND flash memory, and hard disk drives.

Jizhong Zhao (A'08) received the B.S. and M.S. degrees in mathematics and Ph.D. degree in computer science from Xi'an Jiaotong University, Xi'an, China, in 1992, 1995, and 2001, respectively.

Currently, he is a Professor with the Computer Science and Technology Department, Xi'an Jiaotong University, Xi'an, China. His research interests include computer software, distributed systems, and computer architecture.

Dr. Zhao is a member of the Association for Computing Machinery and the IEEE Computer Society.

Nanning Zheng (F'06) received the B.S. degree in electrical engineering and M.S. degree in information and control engineering from Xi'an Jiaotong University, Xi'an, China, in 1975 and 1981, respectively, and the Ph.D. degree in electrical engineering from Keio University, Yokohama, Japan, in 1985.

He jointed Xi'an Jiaotong University, Xi'an, China, in 1975, where he is currently a Professor and the Director of the Institute of Artificial Intelligence and Robotics. He was General Chair of the International Symposium on Information Theory and Its Applications and General Co-Chair of the International Symposium on Nonlinear Theory and Its Applications, both in 2002. His research interests include computer vision, pattern recognition, machine vision and image processing, neural networks, and hardware implementation of intelligent systems.

Dr. Zheng became a member of the Chinese Academy of Engineering in 1999 and has been the Chief Scientist and Director of the Information Technology Committee of the China National High Technology Research and Development Program since 2001. He is a member of the Board of Governors of the IEEE Intelligent Transportation Systems Society and the Chinese Representative on the Governing Board of the International Association for Pattern Recognition. He also serves as an executive deputy editor of the Chinese Science Bulletin.

Tong Zhang (SM'08) received the B.S. and M.S. degrees from the Xi'an Jiaotong University, Xi'an, China, in 1995 and 1998, respectively, and the Ph.D. degree from the University of Minnesota, Minneapolis, in 2002, all in electrical engineering.

Currently, he is an Associate Professor with the Electrical, Computer and Systems Engineering Department, Rensselaer Polytechnic Institute, Troy, NY. His current research interests include algorithm and architecture co-design for communication and data storage systems, variation-tolerant signal processing IC design, fault-tolerant system design for digital memory, and interconnect system design for hybrid CMOS/nanodevice electronic systems.

Prof. Zhang is currently serving as an associate editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II—EXPRESS BRIEFS and the IEEE TRANSACTIONS ON SIGNAL PROCESSING.