Chapter 7
STT-RAM Cache Hierarchy Design and Exploration with Emerging Magnetic Devices

Hai (Helen) Li, Zhenyu Sun, Xiuyuan Bi, Weng-Fai Wong, Xiaochun Zhu and Wenqing Wu

Abstract Spin-transfer torque random access memory (STT-RAM) is a promising new nonvolatile technology that has good scalability, zero standby power, and radiation hardness. The use of STT-RAM in last level on-chip caches has been proposed as it significantly reduced cache leakage power as technology scales down. Having a cell area only 1/9 to 1/3 that of SRAM, this will allow for a much larger cache with the same die footprint. This will significantly improve overall system performance, especially in this multicore era where locality is crucial. However, deploying STT-RAM technology in L1 caches is challenging because write operations on STT-RAM are slow and power-consuming. In this chapter, we propose a range of cache hierarchy designs implemented entirely using STT-RAM that delivers optimal power saving and performance. In particular, our designs use STT-RAM cells with various data retention times and write performances, made possible by novel magnetic tunneling junction (MTJ) designs. For L1 caches where speed is of the utmost importance, we propose a scheme that uses fast STT-RAM cells with reduced data retention time coupled with a dynamic refresh scheme. We will show that such a cache can achieve 9.2 % in performance improvement and saves up to 30 % of the total energy when compared to one that uses traditional SRAM. For lower-level caches with relatively

H. H. Li (✉) · Z. Sun · X. Bi
Department of Electrical and Computer Engineering, University of Pittsburgh,
3700 O'Hara Street, Pittsburgh, PA 15238, USA
e-mail: hai.helen.li@gmail.com

W. F. Wong
National University of Singapore,
13, Computing Drive, Singapore 117417, Republic of Singapore

X. Zhu
Qualcomm Inc.,
10185 Mckellar Ct., San Diego, CA 92109, USA

W. Wu
Qualcomm Inc.,
5665 Morehouse Dr., San Diego, CA 92121, USA

Y. Xie (ed.), *Emerging Memory Technologies*,
DOI: 10.1007/978-1-4419-9551-3_7,
© Springer Science+Business Media New York 2014

larger cache capacities, we propose a design that has partitions of different retention characteristics and a data migration scheme that moves data between these partitions. The experiments show that on the average, our proposed multiretention-level STT-RAM cache reduces total energy by as much as 30–70 % compared to previous single retention-level STT-RAM cache, while improving IPC performance for both 2-level and 3-level cache hierarchies.

7.1 Introduction

Increasing capacity and cell leakage have caused the standby power of SRAM on-chip caches to dominate the overall power consumption of the latest microprocessors. Many circuit design and architectural solutions, such as V_{DD} scaling [14], power gating [17], and body biasing [13], have been proposed to reduce the standby power of caches. However, these techniques are becoming less effective as technology scaling has caused the transistor's leakage current to increase exponentially. Researchers have been prompted to look into the alternatives of SRAM technology. One possibility is the embedded DRAM (eDRAM) which is denser than SRAM. Unfortunately, it suffers from serious process variation issues [1]. Another alternative technology is the embedded phase change memory (PCM) [5], a new nonvolatile memory that can achieve very high density. However, its slow access speed makes PCM unsuitable as a replacement for SRAM.

Another substitute for SRAM, the spin-transfer torque RAM (STT-RAM), is receiving significant attention because it offers almost all the desirable features of a universal memory: the fast (read) access speed of SRAM, the high integration density of DRAM, and the nonvolatility of Flash memory. Also, the compatibility with the CMOS fabrication process and similarities in the peripheral circuitries makes the STT-RAM an easy replacement for SRAM.

However, there are two major obstacles to use STT-RAM for on-chip caches, namely its longer write latency and higher write energy. During an STT-RAM write operation in the sub-10 ns region, the magnetic tunnel junction (MTJ) resistance switching mechanism is dominated by *spin precession*. The required switching current rises exponentially as the MTJ switching time is reduced. As a consequence, the driving transistor's size must increase accordingly, leading to a larger memory cell area. The lifetime of memory cell also degrades exponentially as the voltage across the oxide barrier of the MTJ increases. As a result, a 10 ns programming time is widely accepted as the performance limit of STT-RAM designs and is adopted in mainstream STT-RAM research and development [6, 8, 12, 25, 30].

Several proposals have been made to address the write speed and energy limitations of the STT-RAM. For example, the early write termination scheme [32] mitigates the performance degradation and energy overhead by eliminating unnecessary writes to STT-RAM cells. The dual write speed scheme [30] improves the average access time of an STT-RAM cache by having a fast and a slow cache partition. A classic SRAM/STT-RAM hybrid cache hierarchy with 3D stacking structure

was proposed in Ref. [25] to compensate the performance degradation caused by STT-RAM by migrate write-intensive data block into SRAM-based cache way.

In memory design, the *data retention time* indicates how long data can be retained in a nonvolatile memory cell after it has been written. In other words, it is the unit for measuring nonvolatility of a memory cell. Relaxing this nonvolatility can make the memory cells easier to be programmed and leads to a lower write current or faster switching speed. In Ref. [22], the volume (cell area) of the MTJ device is reduced to achieve better writability by sacrificing the retention time of the STT-RAM cache cells. A simple DRAM-style refresh scheme was also proposed to maintain the correctness of the data.

The key insight informing this paper is that the access patterns of L1 and lower-level caches in a multicore microprocessor are different. Based on this realization, we propose the use of STT-RAM designs with different nonvolatilities and write characteristics in different parts of the cache hierarchy so as to maximize power and performance benefits. A low-power dynamic refresh scheme is proposed to maintain the validity of the data. Compared to the existing works on STT-RAM cache designs, our work makes the following contributions:

- We present a detailed discussion on the trade-off between the MTJ's write performance and its nonvolatility. Using our macromagnetic model, we qualitatively analyze and optimize the device
- We propose a multiretention-level cache hierarchy implemented entirely with STT-RAM that delivers the optimal power saving and performance improvement based on the write access patterns at each level. Our design is easier to fabricate and has a lower die cost
- We present a novel refresh scheme that achieves the theoretically minimum refresh power consumption. A counter is used to track the life span of L1 cache data. The counter can be composed of SRAM cells or the spintronic memristor. As an embedded on-chip timer, spintronic memristor can save significant energy and on-chip area when compared to SRAM. Moreover, the cell size of the memristor is close to that of STT-RAM, making the layout easier
- We propose the use of a hybrid lower-level STT-RAM design for cache with large capacity that simultaneously offers fast average write latency and low standby power. It has two cache partitions with different write characteristics and nonvolatility. A data migration scheme to enhance the cache response time to write accesses is also described. The proposed hybrid cache structure has been evaluated in lower-level cache of both 2-level and 3-level cache hierarchies.

The rest of our chapter is organized as follows. Section 7.2 introduces the technical backgrounds of STT-RAM and spintronic memristor. Section 7.3 describes the trade-offs involved in MTJ nonvolatility relaxation and the memristor-based counter design. Section 7.4 proposes our multiretention STT-RAM L1 and L2 cache structures. Section 7.5 discusses our experimental results. Related works are summarized in Sect. 7.6, followed by our conclusion in Sect. 7.7.

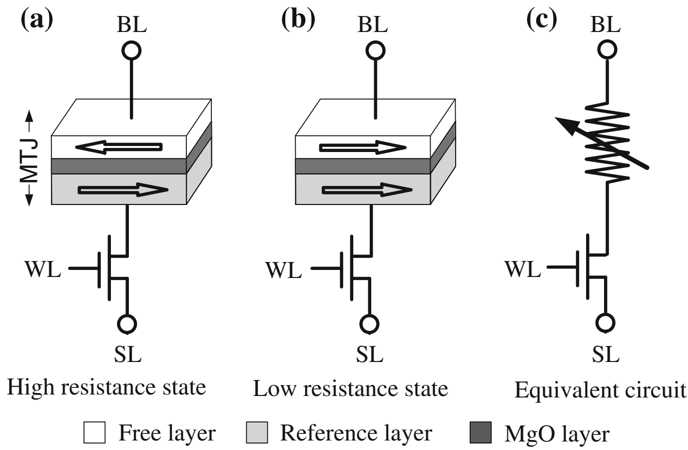

Fig. 7.1 1T1J STT-RAM design. **a** MTJ is in anti-parallel state; **b** MTJ is in parallel state; **c** the equivalent circuit

7.2 Background

7.2.1 STT-RAM

The data storage device in an STT-RAM cell is the magnetic tunnel junction (MTJ), as shown in Fig. 7.1a, b. A MTJ is composed of two ferromagnetic layers that are separated by an oxide barrier layer (e.g., MgO). The magnetization direction of one ferromagnetic layer (the *reference layer*) is fixed, while that of the other ferromagnetic layer (the *free layer*) can be changed by passing a current that is polarized by the magnetization of the reference layer. When the magnetization directions of the free layer and the reference layer are parallel (anti-parallel), the MTJ is in its low (high)-resistance state.

The most popular STT-RAM cell design is one-transistor-one-MTJ (or 1T1J) structure, where the MTJ is selected by turning on the word line (WL) that is connected to the gate of the NMOS transistor. The MTJ is usually modeled as a current-dependent resistor in the circuit schematic, as shown in Fig. 7.1c. When writing “1” (high-resistance state) into the STT-RAM cell, a positive voltage is applied between the source line (SL) and the bit line (BL). Conversely, when writing a “0” (low resistance state) into the STT-RAM cell, a negative voltage is applied between the SL and the BL. During a read operation, a sense current is injected to generate the corresponding BL voltage V_{BL}. The resistance state of the MTJ can be read out by comparing the V_{BL} to a reference voltage.

Table 7.1 Memristor design parameters

Length (L)	Width (w)	Thickness (h)	R_H (Ω)	R_L (Ω)	Γ_v ($\mathrm{nm}^3 \times \mathrm{C}^{-1}$)	J_{cr} ($\mathrm{A} \times \mathrm{nm}^{-2}$)
90 nm	45 nm	14 nm	7,500	2,500	2.01×10^{-14}	2×10^{-8}

7.2.2 Spintronic Memristor

As the fourth passive circuit element, the memristor has the natural property to record the historical profile of its electrical excitations [4]. In 2008, HP Lab reported the discovery of the memristor device, which was realized in a TiO_2 thin-film device [23]. In this work, we use the magnetic version of memristors, the tunneling magnetoresistance (TMR)-based spintronic memristor, as its device structure is similar to the MTJ, having a compatible manufacturing process.

Figure 7.2a illustrates the structure of a spintronic memristor [20, 27]. Like a MTJ, the free layer in a spintronic memristor is divided into two domains whose magnetization directions are, respectively, parallel or anti-parallel to the one of the reference layer. The domain wall can move along the length of the free layer when a polarized current is applied vertically.

As shown in Fig. 7.2b, the overall resistance of such a spintronic memristor can be modeled as two resistors connected in parallel with resistances R_L/α and $R_H/(1-\alpha)$, respectively [27]. Here, $0 \leq \alpha \leq 1$ represents the relative position of the domain wall which is the ratio of the domain wall position (x) over the total length of the free layer (L). The overall memristance can be expressed as

$$M(\alpha) = \frac{R_H \times R_L}{R_H \times \alpha + R_L \times (1 - \alpha)}. \tag{7.1}$$

The domain wall moves only when the applied current density (J) is above the critical current density (J_{cr}) [15]. The domain wall velocity, $v(t)$, is determined by the spin-polarized current density J as [15]

$$v(t) = \frac{\mathrm{d}\alpha(t)}{\mathrm{d}t} = \frac{\Gamma_v}{L} \times J_{\mathrm{eff}}(t), \quad J_{\mathrm{eff}} = \begin{cases} J, J \geq J_{\mathrm{cr}} \\ 0, J \geq J_{\mathrm{cr}}. \end{cases} \tag{7.2}$$

where Γ_v is the domain wall velocity coefficient. It is determined by the device's structure and material. It is particularly noteworthy that the domain wall motion process has been successfully demonstrated in fabrication very recently [18].

Figure 7.3 shows the simulated programming property of a spintronic memristor when successive square waves are applied. The dashed gray line represents the pushing pulses applied across the spintronic memristor; the solid pink line is the corresponding current amplitude through the memristor. At the beginning of the simulated period, the pushing pulses raise up memristance, and therefore, the current amplitude decreases. Once the domain wall hits the device boundary, subsequent application of

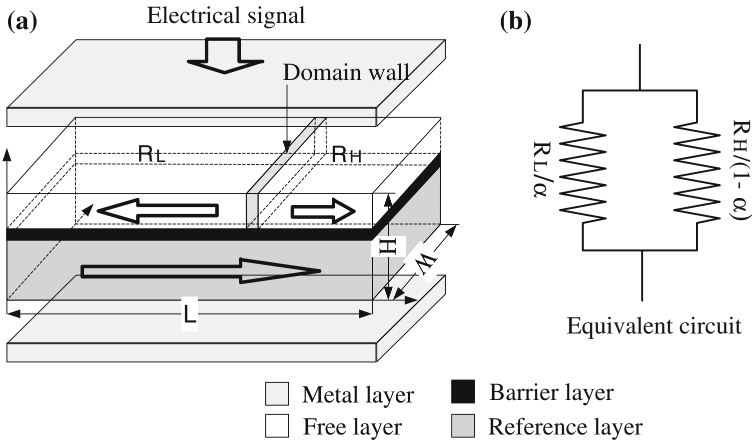

Fig. 7.2 TMR-based memristor. **a** the device view; **b** the equivalent circuit

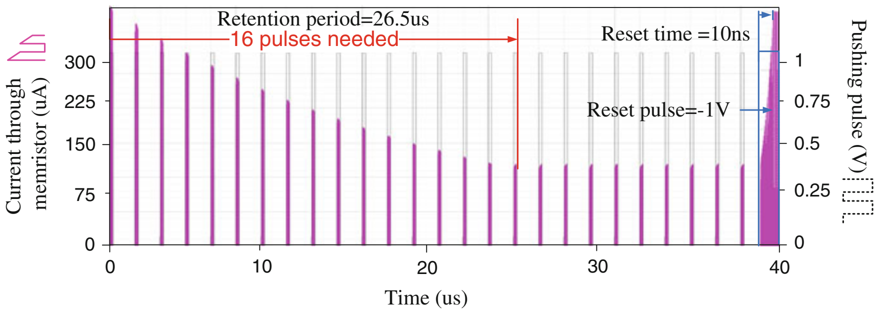

Fig. 7.3 The simulated programming property of a spintronic memristor

electrical excitations will not change the memristance. The larger the level number of memristance is, the more accurate the decision granularity will be. However, that also indicates a smaller sense margin. As we shall show in Sect. 7.4 that for our purpose, we partitioned the memristance of the spintronic memristor into a moderate number of 16 levels. The parameters of the spintronic memristor are summarized in Table 7.1.

7.3 Design

7.3.1 MTJ Write Performance Versus Nonvolatility

The *data retention time*, T_{store}, of a MTJ is determined by the *magnetization stability energy height*, Δ:

$$T_{\text{store}} = \frac{1}{f_0} e^{\Delta}. \tag{7.3}$$

f_0 is the thermal attempt frequency, which is of the order of 1 GHz for storage purposes [7]. Δ can be calculated by

$$\Delta = \left(\frac{K_u V}{k_{\text{B}} T}\right) = \left(\frac{M_s H_k V \cos^2(\theta)}{k_B T}\right), \tag{7.4}$$

where M_{s} is the saturation magnetization. H_k is the effective anisotropy field including magnetocrystalline anisotropy and shape anisotropy. θ is the initial angle between the magnetization vector and the easy axis. T is the working temperature. k_B is Boltzmann constant. V is the effective activation volume for the spin-transfer torque writing current. As Eqs. (7.3) and (7.4) show, the data retention time of an MTJ decreases exponentially when its working temperature, T, rises.

The required *switching current density*, J_{C}, of an MTJ operating in different working regions can be approximated as [21, 24]

$$J_C^{\text{THERM}}(T_{\text{sw}}) = J_{C0}\left(1 - \frac{1}{\Delta}\ln\left(\frac{T_{\text{sw}}}{\tau_0}\right)\right) \quad (T_{\text{sw}} > 10\,\text{ns}) \tag{7.5}$$

$$J_C^{\text{DYN}}(T_{\text{sw}}) = \frac{J_C^{\text{THERM}}(T_{\text{sw}}) + J_C^{\text{PREC}}(T_{\text{sw}})e^{(-A(T_{\text{sw}}-T_{\text{PIV}}))}}{1 + e^{(-A(T_{\text{sw}}-T_{\text{PIV}}))}} \quad (10\,\text{ns} > T_{\text{sw}} > 3\,\text{ns}) \tag{7.6}$$

$$J_C^{\text{PREC}}(T_{\text{sw}}) = J_{C0} + \frac{C\ln(\frac{\pi}{2\theta})}{T_{\text{sw}}} \quad (T_{\text{sw}} < 3\,\text{ns}). \tag{7.7}$$

Here, A, C, and T_{PIV} are the fitting parameters. T_{sw} is the switching time of MTJ resistance. $J_C = J_C^{\text{THERM}}(T_{\text{sw}})$, $J_C^{\text{DYN}}(T_{\text{sw}})$, or $J_C^{\text{PREC}}(T_{\text{sw}})$ are the required switching currents at T_{sw} in different working regions, respectively. The switching threshold current density J_{C0}, which causes a spin flip in the absence of any external magnetic field at 0 K, is given by

$$J_{C0} = (\frac{2e}{\hbar})\left(\frac{\alpha}{\eta}\right)(t_F M_s)\,(H_k \pm H_{\text{ext}} + 2\pi M_s). \tag{7.8}$$

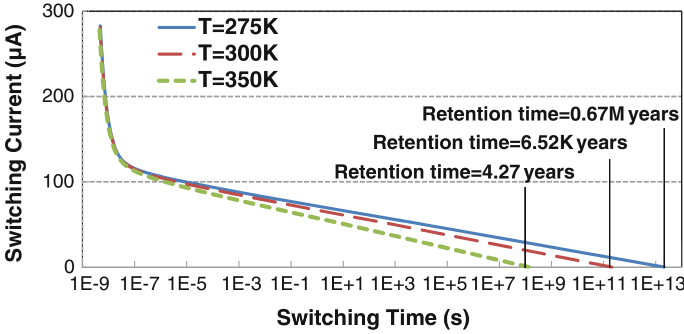

Fig. 7.4 The relationship between the switching current and the switching time of "Base" MTJ design

Here, e is the electron charge, α is the damping constant, τ_0 is the relaxation time, t_F is the free layer thickness, $\hbar$ is the reduced Planck's constant, H_{ext} is the external field, and η is the spin-transfer efficiency.

As proposed by [22], shrinking the cell surface area of the MTJ can reduce Δ and consequently decreases the required switching density J_C, as shown in Eq. (7.5). However, such a design becomes less efficient in the fast-switching region ($T_{\text{SW}} < 3$ ns) because the coupling between Δ and J_C is less in this region, as shown in Eq. (7.7). Based on the MTJ switching behavior, we propose to change M_s, H_k, or t_F to reduce J_c. Such a technique can lower not only Δ but also J_{c0}, offering efficient performance improvement over the entire MTJ working range.

We simulated the switching current versus the switching time of a baseline 45×90 nm elliptical MTJ over the entire working range, as shown in Fig. 7.4. The simulation is conducted by solving the stochastic magnetization dynamics equation describing spin torque-induced magnetization motion at finite temperature [28]. The MTJ parameters are taken from [28], which are close to the measurement results recently reported in [31]. The MTJ data retention time is measured as the MTJ switching time when the switching current is zero. When the working temperature rises from 275 to 350 K, the MTJ's data retention time decreased from 6.7×10^6 to

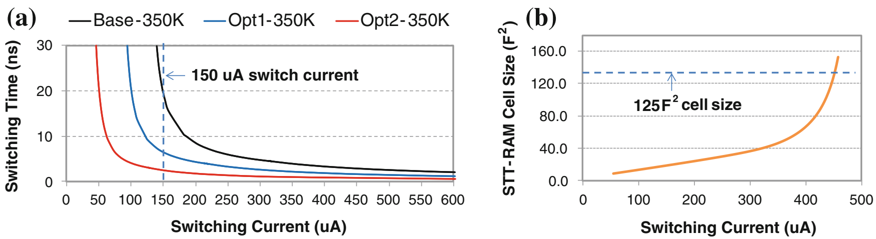

Fig. 7.5 **a** MTJ switching performances for different MTJ designs at 350 K. **b** The minimal required STT-RAM cell size at given switching current

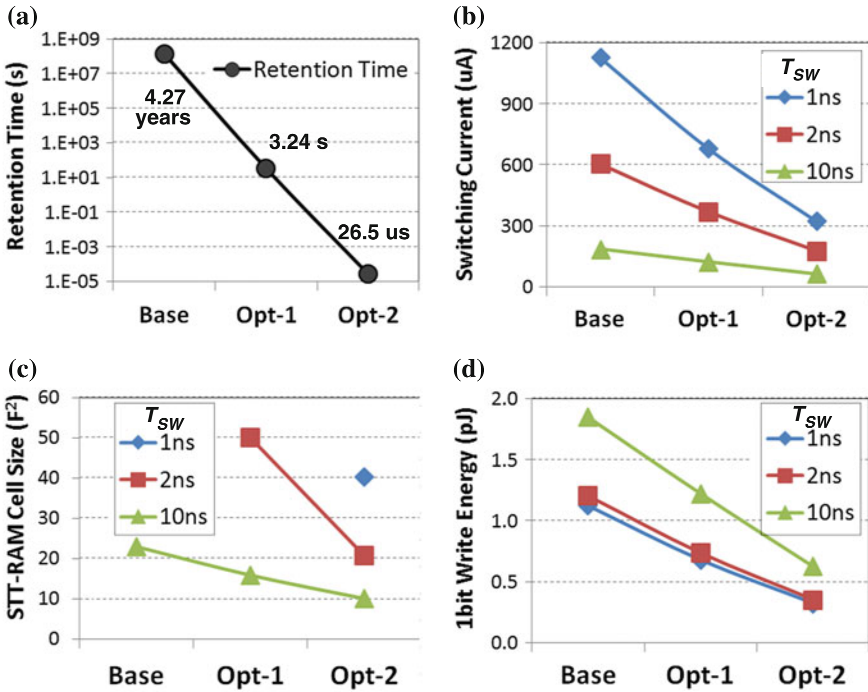

Fig. 7.6 Comparison of different MTJ designs at 350K: **a** the retention time. **b** the switching current. **c** STT-RAM cell size, and **d** the bit write energy

4.27 years. In the experiments reported in this work, we shall assume that the chip is working at a high temperature of 350 K.

7.3.2 STT-RAM Cell Design Optimization

To quantitatively study the trade-offs between the write performance and nonvolatility of an MTJ, we simulated the required switching current of three different MTJ designs with the same cell surface shapes. Besides the "Base" MTJ design shown in Fig. 7.4, two other designs ("Opt1" and "Opt2") that are optimized for better switching performance with degraded nonvolatility were studied. The corresponding MTJ switching performances of these three designs at 350 K are shown in Fig. 7.5a. The detailed comparisons of data retention times, the switching currents, the bit write energies, and the corresponding STT-RAM cell sizes of three MTJ designs at the given switching speed of 1, 2, and 10 ns are given in Fig. 7.6.

Significant write power saving is achieved if the MTJ's nonvolatility can be relaxed. For example, when the MTJ data retention time is scaled from 4.27 years

("Base") to 26.5 μs ("Opt2"), the required MTJ switching current decreases from 185.2 to 62.5 μA for a 10 ns switching time at 350 K. Or, at an MTJ switching current of 150 μA, the corresponding switching times of all three MTJ designs varied from 20 to 2.5 ns. A switching performance improvement of 8× can be obtained, as shown in Fig. 7.5a.

Since the switching current of an MTJ is proportional to its area, the MTJ is normally fabricated with the smallest possible dimension. The STT-RAM cell's area is mainly constrained by the NMOS transistor which needs to provide sufficient driving current to the MTJ. Figure 7.5b shows the minimal required NMOS transistor size when varying the switching current and the corresponding STT-RAM cell area at 45 nm technology node. The PTM model was used in the simulation [3], and the power supply V_{DD} is set to 1.0 V. Memory cell area is measured in F^2, where F is the feature size at a certain technology node.

According to the popular cache and memory modeling software CACTI [2], the typical cell area of SRAM is about 125 F^2. For an STT-RAM cell with the same area, the maximum current that can be supplied to the MTJ is 448.9 μA. A MTJ switching time of less than 1ns can be obtained with the "Opt2" design under such as a switching current, while the corresponding switching time for the baseline design is longer than 4.5 ns. In this chapter, we will not consider designs that are larger than 125 F^2.

Since "Opt1" and "Opt2" require less switching current than the baseline design for the same write performance, they also consume less write energy. For instance, the write energies of "Base" and "Opt2" designs are 1.85 and 0.62 pJ, respectively, for a switching time of 10 ns. If the switching time is reduced to 1 ns, the write energy of "Opt2" design can be further reduced down to 0.32 pJ. The detailed comparisons on the write energies of different designs can be found in Fig. 7.6d.

7.4 Multiretention-level STT-RAM Cache Hierarchy

In this section, we will describe our multiretention-level STT-RAM-based cache hierarchy. Our multiretention-level STT-RAM cache hierarchy takes into account the difference in access patterns in L1 and the lower-level cache (LLC).

For L1, the overriding concern is access latency. Therefore, we propose the use of our "Opt2" nonvolatility-relaxed STT-RAM cell design as the basis of the L1 cache. In order to prevent data loss introduced by relaxing its nonvolatility, we propose a dynamic refresh scheme to monitor the life span of the data and refresh cells when needed. LLC caches are much larger than L1 cache. As such, a design built with only "Opt2" STT-RAM cells will consume too much refresh energy. Use of the longer retention "Base" or "Opt1" design is more practical. However, to recover the lost performance, we propose a hybrid LLC that has a regular and a nonvolatility-relaxed STT-RAM portions. Data will be migrated from one to other accordingly. The details of our proposed cache hierarchy will be given in the following subsections.

7.4.1 The Nonvolatility-Relaxed STT-RAM L1 Cache Design

As established earlier, using the "Opt2" STT-RAM cell design for L1 caches can significantly improve the write performance and energy. However, its data retention time of 26.5 μs may not be sufficient to retain the longest living data in L1. Therefore, a refresh scheme is needed. In Ref. [22], a simple DRAM-style refreshing scheme was used. This scheme refreshes all cache blocks in sequence, regardless of its data content. Read and write accesses to memory cells that are being refreshed must be stalled. As we shall show in Sect. 7.5.2, this simple scheme introduces many unnecessary refreshing operations whose elimination will significantly improve performance and save energy.

7.4.1.1 Dynamic Refresh Scheme

To eliminate unnecessary refresh, we propose the use counters to track the life span of cache data blocks. Refresh is performed only on cache blocks that have reached their full life span. In our refresh scheme, we assign one counter to each data block in the L1 cache to monitor its data retention status. Figure 7.7 illustrates our dynamic refresh scheme. The operation of the counter can be summarized as follows:

- *Reset*: On any write access to a data block, its corresponding counter is reset to '0'
- *Pushing*: We divide the STT-RAM cell's retention time into N_{mem} periods, each of which is T_{period} long. A global clock is used to maintain the countdown to T_{period}. At the end of every T_{period}, the level of every counter in the cache is increased by one
- *Checking*: The data block corresponding to a counter would have reached the maximum retention time when the counter reaches its highest level and hence needs

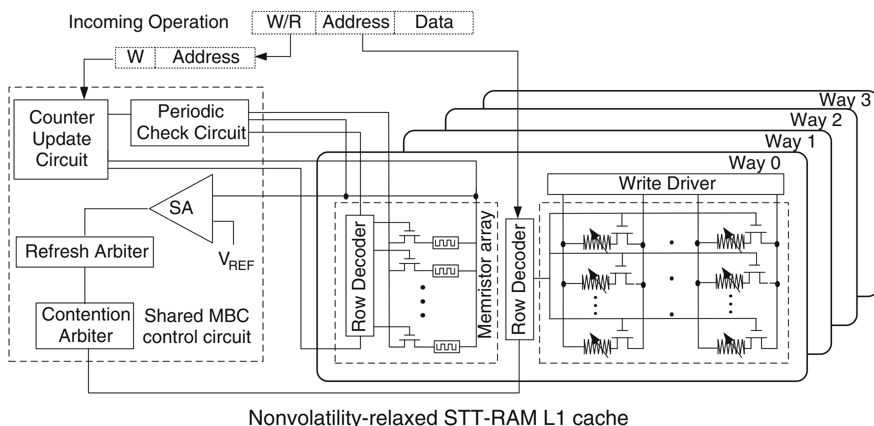

Fig. 7.7 Memristor counter-based refreshing scheme

to be refreshed. The overhead of such counter pushing scheme is very moderate. Take, for example, a 32 KB L1 cache built using the "Opt2" STT-RAM design and a counter can represent 16 values from 0 to 15. A pushing operation happens once every 3.23 ns, $= (26.5\,\mu s/512/16)$ in the entire L1 cache. This is more than 6 cycles at a 2 GHz clock frequency. A larger cache may mean a higher pushing overhead.

The following is some design details of the proposed dynamic refresh scheme:

- *Cache access during refresh*: During a refresh operation, the block's data are read out into a buffer and then saved back to the same cache block. If a read request to the same cache block comes before the refresh finishes, the data are returned from this buffer directly. There is therefore no impact on the read response time of the cache. Should a write request comes, the refresh operation is terminated immediately, and the write request is executed. Again, no penalty is introduced
- *Reset threshold* N_{th}: However, we observe that during the life span of a cache block, updates happen more frequently within a short period of time after it has been written. Many resets of the cache block data occur far from their data retention time limits, giving us an optimization opportunity. We altered the reset scheme to eliminate counter resets that happen within a short time period after data have been written. We define a threshold level, N_{th}, that is much smaller than N_{mem}. The counter is reset only when its resistance is higher than N_{th}. The larger N_{th} is, the more resets are eliminated. On the other hand, the refresh interval of the data next written into the same cache block is shortened. However, our experiments in Sect. 7.5.2 shall show that such cases happen very rarely and the lifetimes of most data blocks in the L1 cache are much shorter than 26.5 μs.

7.4.1.2 Counter Design

In the proposed scheme, the counters are used in two ways: (1) to monitor the time duration for which the data have been written into the memory cells and (2) to monitor the read and write intensity of the memory cells. These counters can be implemented either by the traditional SRAM or by the recently discovered memristor device. The design detail of memristor as an on-chip analog counter will be introduced here. A Verilog-A model for spintronic memristor [27] was used in circuit simulations.

As demonstrated in Eqs. (7.1) and (7.2), when the magnitude of programming pulse is fixed, the memristance (resistance) of a spintronic memristor is determined by the accumulated programming pulse width. We utilize this characteristic to implement a high-density, low-power, and high-performance counter used in our cache refresh scheme: The memristance of a memristor can be partitioned into multiple levels, corresponding to the values the counter can record.

The maximum number of memristance levels is constrained by the minimal sense margin of the sense amplifier/comparator and the resolution of the programming pulse, i.e., the minimal pulse width. The difference between R_H and R_L of the spintronic memristor used in this work is 5,000 Ω (see Table 7.1), which is sufficiently

Table 7.2 Comparison between SRAM and memristor counter

A 4-bit counter	SRAM	Memristor
Area of a memory cell	$100 \sim 150\,F^2$	$33\,F^2$
Number of memory cells	4	1
Pushing and checking energy	0.7 pJ	0.45 pJ
Reset energy	0.46 pJ	7.2 pJ
Sense margin	$50 \sim 100$ mV at 45 nm tech.	46.875 mV

large to be partitioned into 16 levels. Moreover, we use the pushing current of 150 μA as the read current, further enlarging the sensing margin. The sense margin of the memristor-controlled counter $\Delta V = 46.875\,\text{mV}$ ($150\,\mu A \times 5{,}000\,\Omega/16$ levels) is at the same level as the sense margin in nowadays SRAM design.

The area of a memristor is only $2\,F^2$ (refer Table 7.1). The total size of a memristor counter including a memristor and a control transistor is below $33\,F^2$. For comparison, the area of a 6T SRAM cell is about $100 \sim 150\,F^2$ [33]. More importantly, the memristor counter has the same layout structure as STT-RAM and therefore can be easily integrated into STT-RAM array.

The memristance variation induced by process variations [10] is the major issue when utilizing memristors as data storage device. The counter design faces the same issue, but the impact is not that critical: As a timer, the memristance variation can be overcome by giving enough design margin to guarantee the on-time refresh.

Every *pushing and checking* operation of a SRAM counter should include two actions: increase the counter value by one and read it out. In the proposed memristor counter design, the injected current can obtain the two purposes simultaneously —pushing the domain wall to enable counter value increment and meanwhile serving as read current for data detection. The comparison between the two types of counter designs is summarized in Table 7.2. Note that the memristor counter has a larger energy consumption during a reset operation in which its domain wall moves from one end to the other.

7.4.2 Lower-level Cache with Mixed High- and Low-Retention STT-RAM Cells

The data retention time requirement in the mainstream STT-RAM development of $4 \sim 10$ years was inherited from Flash memory designs. Although such a long data retention time can save significant standby power of on-chip caches, it also entails a long write latency (~ 10 ns) and large write energy [25]. Relaxing the nonvolatility of the STT-RAM cells in the lower-level cache will improve write performance as well as save more energy. However, if further reducing retention time to μs scale, e.g., 26.5 μs of our "Opt2" cell design, the refresh energy dominates, and hence, any refresh scheme becomes impractical for the large lower-level cache.

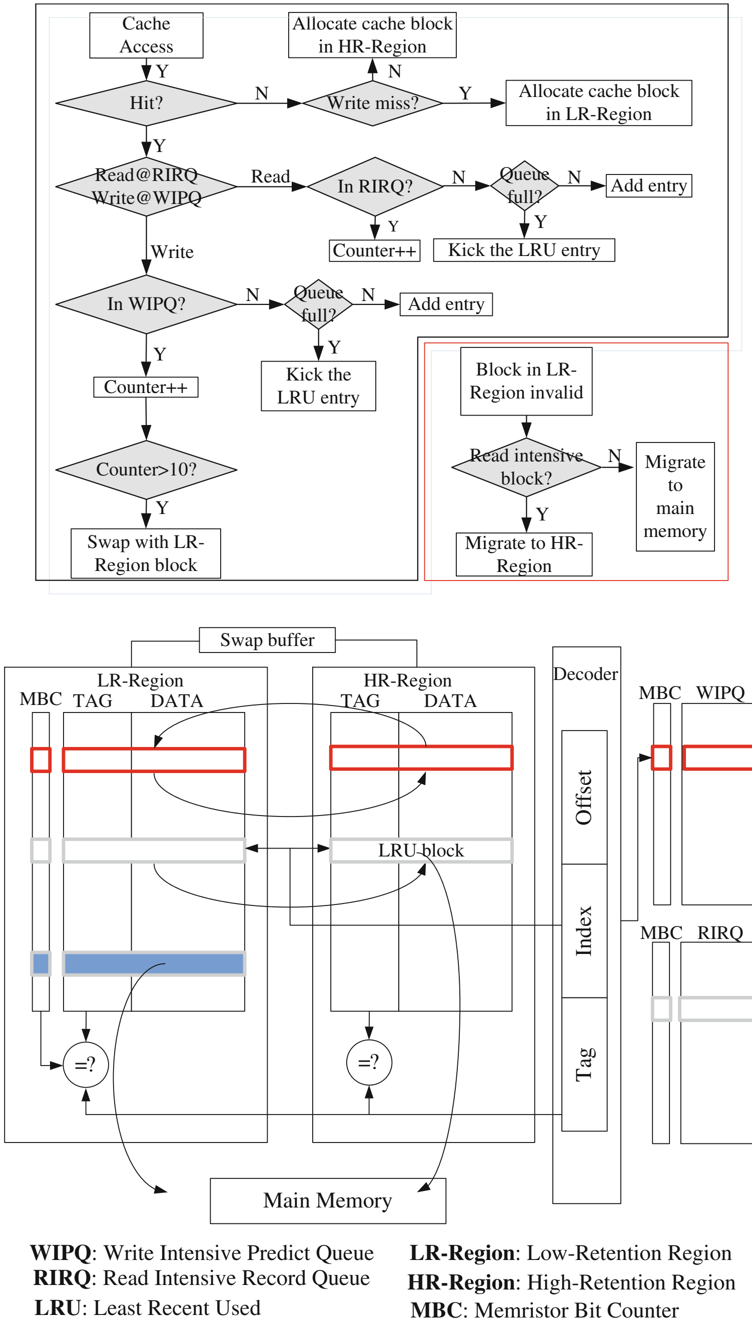

Fig. 7.8 Hybrid lower-level cache migration policy: Flow graph (*top*). Diagram (*bottom*)

The second technique we proposed is a hybrid memory system that has both high- and low-retention STT-RAM portions to satisfy both the power and performance targets simultaneously. We take a L2 cache with 16 ways as a case study as shown in Fig. 7.8, way 0 of the 16-way cache is implemented with a low-retention STT-RAM design ("Opt2"), while ways 1–15 are implemented with the high-retention STT-RAM ("Base" or "Opt1"). Write-intensive blocks are primarily allocated from way 0 for a faster write response, while read-intensive blocks are maintained in the other ways.

Like our proposed L1 cache, counters are used in way 0 to monitor the blocks' data retention status. However, unlike in L1 where we perform a refresh when a memristor counter expires, here we move the data to the high-retention STT-RAM ways.

Figure 7.8 demonstrates the data migration scheme to move the data between the low and the high-retention cache ways based on their write access patterns. A write intensity prediction queue (WIPQ) of 16 entries is added to record the write access history of the cache. Every entry has two parts, namely the data address and an access counter.

During a read miss, the new cache block is loaded to the high-retention (HR) region (ways 1–15), following the regular LRU policy. On a write miss, the new cache block is allocated from the low-retention (LR) region (way 0), and its corresponding memristor counter is reset to '0'. On a write hit, we search the WIPQ first. If the address of the write hit is already in WIPQ, the corresponding access counter is incremented by one. Note that the block corresponding to this address may be in the HR- or the LR-region of the cache. Otherwise, the hit address will be added in to the queue if any empty entry available. If the queue is full, the LRU entry will be evicted and replaced by the current hit address. The access counters in the WIPQ are decremented periodically, for example, every 2,000 clock cycles, so that the entries that are in the queue for too long will be evicted. Once an access counter in a WIPQ entry reaches a preset value, $N_{\mathrm{HR}\rightarrow\mathrm{LR}}$, the data stored in the corresponding address will be swapped with a cache block in the LR-region. If the corresponding address is already in the LR-region, no further action is required. A read hit does not cause any changes to the WIPQ.

Likewise, a read intensity record queue (RIRQ) with the same structure and number of entries is used to record the read hit history of the LR-region. Whenever there is a read hit to the LR-region, a new entry is added into the RIRQ. Or if a corresponding entry already exists in the RIRQ, the value of the access counter is increased by one. When the memristor counter of a cache block B_i in the LR-region indicates the data are about to become unstable, we check to see whether this cache address is read intensive by searching the RIRQ. If B_i is read intensive, it will be moved to HR-region. The cache block being replaced by B_i in the HR-region will be selected using the LRU policy. The evicted cache block will be send to main memory. If B_i is not read intensive, it will be written back to main memory.

In a summary, our proposed scheme uses the WIRQ and RIRQ to dynamically classify cache blocks into three types:

1. *Write intensive*: The addresses of such cache blocks are kept in the WIRQ. They will be moved to the LR-region once their access counters in WIRQ reach $N_{\mathrm{HR}\rightarrow\mathrm{LR}}$;
2. *Read intensive but not write intensive*: The addresses of such cache blocks are found in the RIRQ but not in the WIRQ. As they approach to their data retention time limit, they will be moved to the HR-region
3. *Neither write nor read intensive*: Neither WIRQ nor RIRQ has their addresses. They are kept in HR-region or evicted from LR-region to main memory directly.

Identifying a *write intensive* cache blocks also appeared in some previous works. In Ref. [25], they check whether two successive write accesses go to the same cache block. It is highly possible that a cache block may be accessed several times within very short time and then becomes inactive. Our scheme is more accurate and effective as it monitors the read and write access histories of a cache block throughout its entire life span. The RIRQ ensures that *read intensive* cache blocks migrate from the LR-region to HR-region in a timely manner that, at the same time, also improves energy efficiency and performance.

7.5 Simulation Results and Discussion

7.5.1 Experimental Setup

We modeled a 2 GHz microprocessor with 4 out-of-order cores using MARSSx86 [16]. Assume a two-level or a three-level cache configuration and a fixed 200-cycle main memory latency. The MESI cache coherency protocol is utilized in the private L1 caches to ensure consistency, and the shared lower-level cache uses a write-back policy. The parameters of our simulator and cache hierarchy can be found in Tables 7.3 and 7.4.

Table 7.5 shows the performance and energy consumptions of various designs obtained by a modified NVSim simulator [19]. All the "*-hi*", "*-md*", and "*-lo*" configurations use the "Base", "Opt1", and "Opt2" MTJ designs, respectively. Note that as shown in Fig. 7.5, they scale differently. We simulated a subset of multithreaded workloads from the PARSEC 2.1, and the SPEC 2006 benchmark

Table 7.3 Simulation platform

Max issue width	4 insts	Fetch width	4 insts
Dispatch width	4 insts	Write-back width	4 insts
Commit width	4 insts	Fetch queue size	32 insts
Reorder buffer	64 entries	Max branch in pipeline	24
Load store queue size	32 entries	Functional units	2 ALU 2 FPU
Clock cycle period	0.5 ns	Main memory	200 cycle latency

suites so as to cover a wider spectrum of read/write and cache miss characteristics. We simulated 500 million instructions of each benchmark after their initialization.

SPICE simulations were conducted to characterize the performance and energy overheads of the memristor counter and its control circuit. The reset energy of a memristor counter is 7.2 pJ, and every pushing–checking operation consumes 0.45 pJ.

We compared the performance (in terms of instruction per cycle, IPC) and the energy consumption of different configurations for both 2-level and 3-level hybrid cache hierarchies. The conventional all-SRAM cache design is used as the baseline. The optimal STT-RAM cache configuration based on our simulations is summarized as follows. The detailed experimental results will be shown and discussed in Sects. 7.5.2, 7.5.3, and 7.5.4.

- An optimal 2-level STT-RAM cache hierarchy is the combination of (a) a L1 cache of the "L1-lo2" design and (b) a hybrid L2 cache of using the "L2-lo" in the LR-region and "L2-md2" in the HR-region;
- An optimal 3-level STT-RAM cache hierarchy is composed of (a) a L1 cache of the "L1-lo2" design, (b) a hybrid L2 cache of using the "L2-lo" in the LR-region and "L2-md1" in the HR-region and (c) a hybrid L3 cache of the "L3-lo" design in the LR-region and "L3-md2" in the HR-region.

7.5.2 Results for the Proposed L1 Cache Design

To evaluate the impacts of using STT-RAM in L1 cache design, we implemented the L1-cache with the different STT-RAM designs listed in the L1 cache portion of Table 7.5 while leaving the SRAM L2 cache unchanged. L1 I-cache are read operation, it is implemented by conventional STT-RAM. Compared to SRAM L1 I-cache, conventional STT-RAM has a faster read speed and large cache capacity that can reduce cache miss rate. Due to the smaller STT-RAM cell size, the overall area of L1 cache is significantly reduced. The delay components of interconnect and peripheral circuits also decrease accordingly. Even considering the relatively long sensing latency, the read latency of STT-RAM L1 cache is still similar or even slightly lower than that of a SRAM L1 cache. However, the write performance of STT-RAM L1 cache is always slower than that of the SRAM L1 cache for all the design configurations considered. The leakage power consumption of the STT-RAM

Table 7.4 Cache hierarchy configuration

Baseline 2-level cache hierarchy	Local L1 cache: 32 KB 4-way, 64 B cache block Shared L2 cache: 4 MB 16-way, 128 B cache block
3-level cache hierarchy	Local L1 cache: 32 KB 4-way, 64 B cache block Local L2 cache: 256 KB 8-way, 64 B cache block Shared L3 cache: 4 MB 16-way, 128 B cache block

Table 7.5 Cache configuration

	32 KB L1 Cache					
	SRAM	lo1	lo2	lo3	md	hi
Cell size (F^2)	125	20.7	27.3	40.3	22	23
MTJ switching time (ns)	/	2	1.5	1	5	10
Retention time	/	26.5 μs			3.24 s	4.27 year
Read latency (ns)	1.113	0.778	0.843	0.951	0.792	0.802
Read latency (cycles)	3	2	2	2	2	2
Write latency (ns)	1.082	2.359	1.912	1.500	5.370	10.378
Write latency (cycles)	3	5	4	4	11	21
Read dyn. energy (nJ)	0.075	0.031	0.035	0.043	0.032	0.083
Write dyn. energy (nJ)	0.059	0.174	0.187	0.198	0.466	0.958
Leakage power (mW)	57.7	1.73	1.98	2.41	1.78	1.82
	4 MB L2 or L3 Cache					
	SRAM	lo	md1	md2	md3	hi
Cell size (F^2)	125	20.7	22	15.9	14.4	23
MTJ switching time (ns)	/	2	5	10	20	10
Retention time	/	26.5 μs			3.24 s	4.27 year
Read latency (ns)	4.273	2.065	2.118	1.852	1.779	2.158
Read latency (cycles)	9	5	5	4	4	5
Write latency (ns)	3.603	3.373	6.415	11.203	21.144	11.447
Write latency (cycles)	8	7	13	23	43	23
Read dyn. energy (nJ)	0.197	0.081	0.083	0.070	0.067	0.085
Write dyn. energy (nJ)	0.119	0.347	0.932	1.264	2.103	1.916
Leakage power (mW)	4107	96.1	104	69.1	61.2	110

caches comes from the peripheral circuits only and is very low. The power supply to the memory cells that are not being accessed can be safely cut off without fear of data loss until the data retention limit is reached.

Figure 7.9 illustrates the ratio between read and write access numbers in L1 D-cache. Here, the read and write access numbers are normalized to the total L1 cache access number of `blackscholes`. The ratio reflects the sensitivity of the L1 cache in terms of performance, the dynamic energy toward per-read and per-write latency, and energy of the L1 cache.

Figure 7.10 shows the IPC performance of the simulated L1 cache designs normalized to the baseline all-SRAM cache. On average, implementing the L1 cache using the "Base" (used in "L1-hi") or "Opt1" (used in "L1-md") STT-RAM design incurs more than 32.5–42.5 % IPC degradation, respectively, due to the long write latency. However, the performance of the L1 caches with the low-retention STT-RAM design significantly improves compared to that of the SRAM L1 cache: the average normalized IPC's of 'L1-lo1', 'L1-lo2', and 'L1-lo3' are 0.998, 1.092, and 1.092, respectively. The performance improvement of 'L1-lo2' or 'L1-lo3' L1 cache w.r.t the baseline SRAM L1 cache comes from the shorter read latency even though

its write latency is still longer (see Table 7.5). However, L1 read accesses are far more frequent than write access in most benchmarks as shown in Fig. 7.9. In some benchmarks whose read/write ratio is pretty high, for example, `swaptions`, the 'L1-lo2' or 'L1-lo3' design achieves a better than 20 % improvement in IPC.

The energy consumptions of the different L1 cache designs normalized to the baseline all-SRAM cache are summarized in Fig. 7.11a. The reported results include the energy overhead of the refresh scheme and the counters, where applicable. Not surprisingly, all three low-retention STT-RAM L1 cache designs achieved significant energy savings compared to the SRAM baseline. The "L1-lo3" design consumes more energy because of its larger memory cell size and larger peripheral circuit having more leakage and dynamic power, as shown in Table 7.5. Figure 7.11a also shows that implementing the L1 cache with the "Base" (used in "L1-hi") or "Opt1" (used in "L1-md") STT-RAM is much less energy efficient because (1) the MTJ switching time is longer, resulting in a higher write dynamic energy and (2) a longer operation time due to the low IPC.

Figure 7.11b presents the breakdowns of the read dynamic energy, the write dynamic energy, and the leakage energy in the baseline SRAM cache. First, the leakage occupies more than 30 % of overall energy, most of which can be eliminated in STT-RAM design. Second, when comparing to Fig. 7.9, we noticed that the dynamic read/write energy ratio is close to the read/write access ratio. The high read access ratio together with the lower per-bit read energy consumption of STT-RAM results in a much lower dynamic energy of STT-RAM L1 cache design. Therefore, "L1-lo1", "L1-lo2", and "L1-lo3" STT-RAM designs save up to 30 to 40 % of overall energy compared to the baseline SRAM L1 cache.

Figure 7.12a compares the refresh energy consumptions of the 'L1-lo2' L1 cache under different refresh schemes. In each group, the three bars from left to right

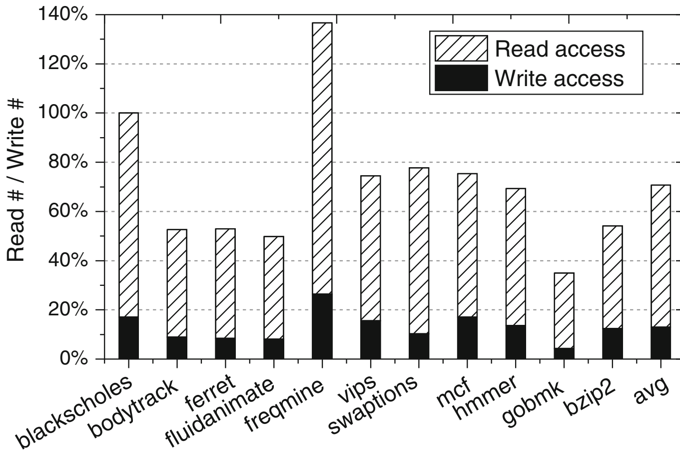

Fig. 7.9 The normalized L1 cache read and write access numbers. The access numbers are normalized to the total L1 access number of `blackscholes`

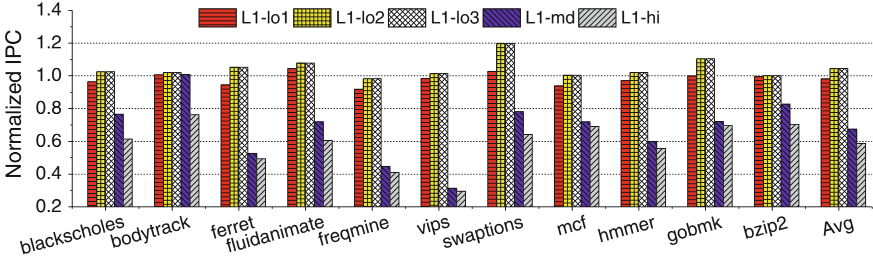

Fig. 7.10 IPC comparison of various L1 cache designs. The IPCs are normalized to all-SRAM baseline

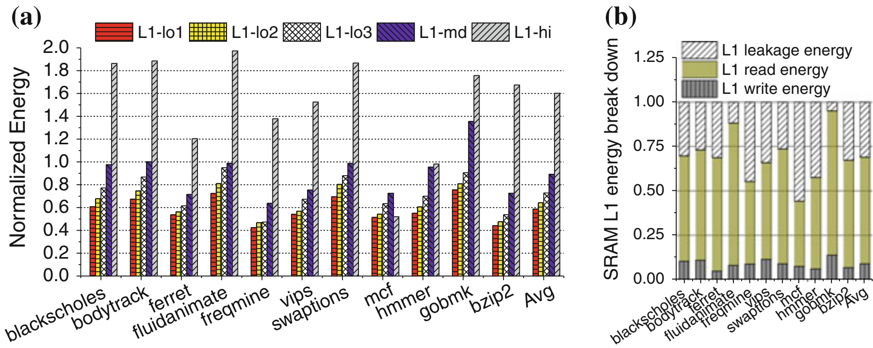

Fig. 7.11 **a** L1 cache overall energy comparison. The energy consumptions are normalized to SRAM baseline. **b** The breakdowns of energy consumption in SRAM-based L1 cache design

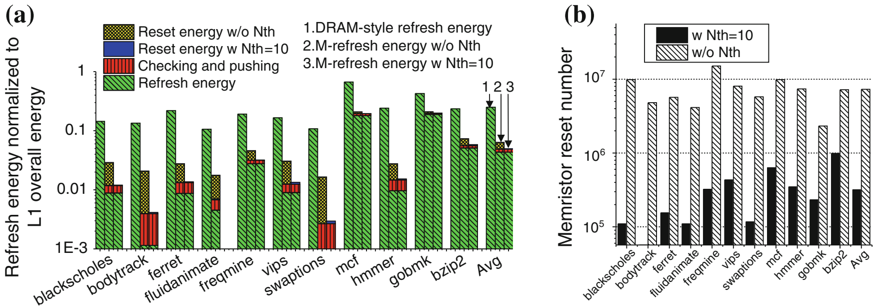

Fig. 7.12 **a** Refresh energy comparison of the different refresh schemes. **b** The number of counter reset operations in the refresh schemes without reset threshold N_{th} and with $N_{th} = 10$

represent the refresh energy consumptions of DRAM-style refresh scheme, refresh scheme without reset threshold N_{th} and with $N_{th} = 10$, respectively. The refresh energy consumptions are normalized to the overall L1 energy consumptions when

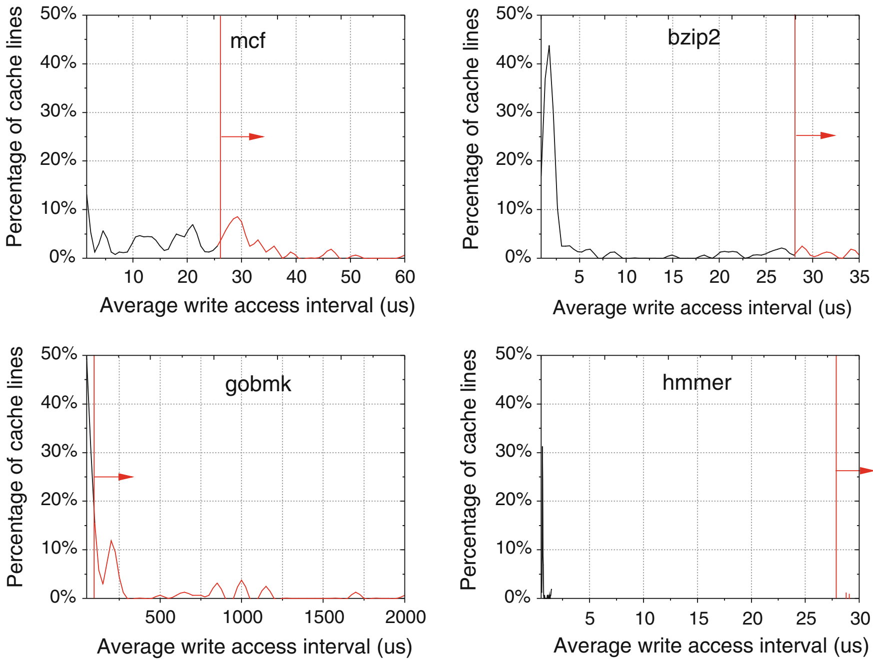

Fig. 7.13 Cache write access distributions of the selected benchmarks

implementing the refresh scheme with $N_{th} = 10$. Note that the y-axis is in logarithmic scale.

The energy consumption of the simple DRAM-style refresh scheme accounts for more than 20 % of the overall L1 cache energy consumption on average. In some extreme cases of low write access frequency, for example, `mcf`, this ratio is as high as 80 % because of the low dynamic cache energy consumption. The total energy consumption of our proposed refresh scheme consists of the checking and pushing, the reset, and the memory cell refresh.

As we discussed in Sect. 7.4.1, the introduction of the reset threshold N_{th} can further reduce the refresh energy consumption by reducing the number of counter resets. This is confirmed in Fig. 7.12a, b. The number of counter reset operations is reduced by more than 20× on average after setting a reset threshold N_{th} of 10, resulting in more than 95 % of the reset energy being saved. The energy consumption for the refresh scheme is very marginal, accounting for only 4.35 % of the overall L1 cache energy consumption. By accurately monitoring the life span of the cache line data, our refresh scheme significantly reduced the refresh energy in all the benchmarks.

The refresh energy saving by utilizing the dynamic refresh scheme is determined by the cache write access distribution and intensity. Figure 7.13 demonstrates the distribution of average write access intervals obtained from four selected benchmarks. In each subfigure, the STT-RAM retention time is represented by the red vertical line.

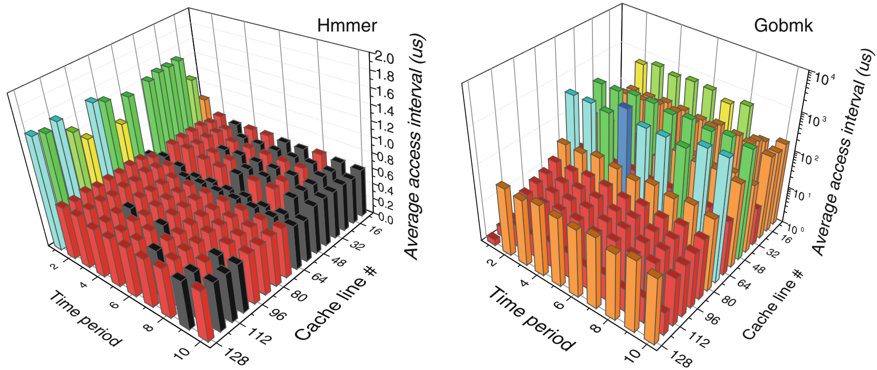

Fig. 7.14 Cache write access intensities of different cache lines

Therefore, the data stored in those cache lines on the right side of the red line need refreshment to maintain correctness.

We also collected the cache write access intensities with time. The results of two selected benchmarks are shown in Fig. 7.14. For illustration purpose, we divide the overall simulation time into ten periods and partition cache lines into eight groups. Figure 7.14 exhibits the average write access intervals for all the cache line groups in each time period. Benchmark `hammer` has a relatively uniform cache write intensity. Its average write access interval is less than 2 μs, which is much shorter than the STT-RAM data retention time 26.5 μs. Often the cache lines are updated by regular write access without refreshed. Therefore, the dynamic refresh scheme can reduce the refresh energy of `hammer` significantly—from 30 to 1 % of the total energy consumption when DRAM-style refresh is utilized. On the contrary, benchmark `gobmk` demonstrates a completely uneven write access intensities among different cache lines. Moreover, the access intervals of many cache lines are longer than the data retention time, making refresh necessary. The dynamic refresh scheme does not benefit too much in such a type of programs.

7.5.3 Evaluating the Hybrid Cache Design in 2-Level Cache Hierarchy

First, we evaluate the proposed hybrid cache design within L2 cache in 2-level cache hierarchies. In comparing the different L2 cache designs, we fixed the L1 cache to the 'L1-lo2' design. In our proposed hybrid L2 cache, way 0 assumes the 'L2-lo' design for the best read latency and the smallest leakage power among all three low-retention STT-RAM designs. Ways 1 to 15 are implemented using the 'L2-md1', 'L2-md2', or 'L2-md3' (all "Opt1" MTJ designs) because a 3.24s retention time is good enough for most applications, and they have the minimal refresh over-

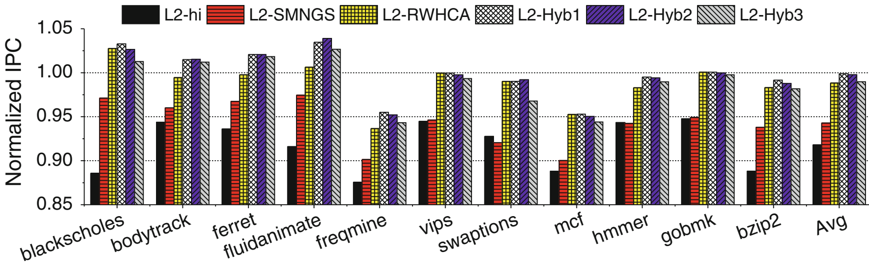

Fig. 7.15 Performance comparison of different 2-level cache designs. The IPCs are normalized to all-SRAM baseline

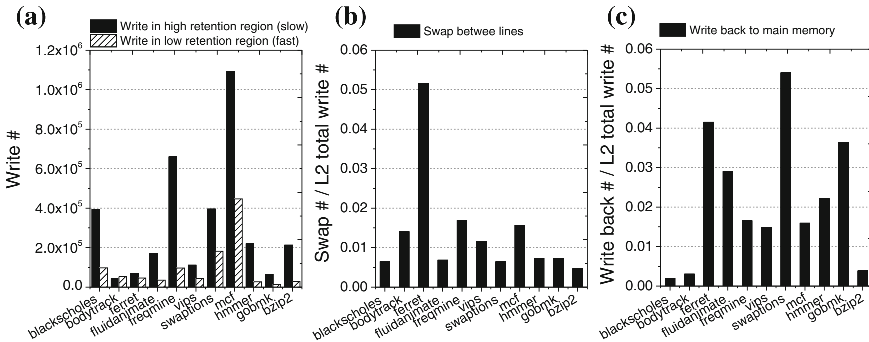

Fig. 7.16 The hybrid L2 cache statistics. **a** The write access numbers in HR- and LR-regions. **b** The ratio of data swaps between HR- and LR-regions among all the L2 accesses. **c** The ratio of data writing back to main memory among all the L2 accesses

head. The three resultant configurations are labeled as 'L2-Hyb1', 'L2-Hyb2', and 'L2-Hyb3', respectively. We compare our hybrid L2 cache with the single retention-level STT-RAM design of [22] and the *read/write aware high-performance architecture* (RWHCA) of [29] and label them as 'L2-SMNGS' and 'L2-RWHCA', respectively. For 'L2-SMNGS', we assumed that the L2 cache uses 'L2-md1' because its cell area of 22 F^2 is compatible with the 19 F^2 one reported in [22]. Instead of using 'L2-hi' in ways 1–15, 'L2-RWHCA' uses 'L2-md2' as it has an access latency that is similar to the one assumed in [29] but a much lower energy consumption. Except for hybrid, all other L2 STT-RAM schemes use the simple DRAM refresh when refresh is needed. To be consistent with the previous section, we normalize the simulation results to the all-SRAM design.

Figure 7.15 compares the normalized IPC results of the different L2 cache designs. As expected, the regular STT-RAM L2 cache with 'L2-hi' design shows the worst performance among all the configurations, especially for benchmarks with high L1 miss rates and L2 write frequencies (such as `mcf` and `swaptions`). Using relaxed retention STT-RAM design, 'L2-SMNGS' improves performance, but on the average, it still suffers 6 % degradation compared to the all-SRAM baseline due to its

longer write latency. Among the three hybrid schemes we proposed, 'L2-Hyb1' is comparable in performance (99.8 % on average) to the all-SRAM cache design. As we prolong the MTJ switching time by reducing STT-RAM cell size in 'L2-Hyb2' and 'L2-Hyb3', IPC performance suffers.

Figure 7.16a compares the write access numbers in HR- and LR-regions in hybrid L2 cache. Some benchmarks, such as `mcf` and `freqmine`, have a large amount of write accesses falling into HR-region, resulting in significant IPC performance degradation. In the contrast, other programs such as `bodytrack` and `ferret` obtain IPC improvement compared to all-SRAM baseline, which mainly benefits from the less L2 write accesses. Although `blacksholes` sends more data to HR-region than `bodytrack` and `ferret`, it has low chances to swap data between HR- and LR-regions and to write data back to main memory, as shown in Fig. 7.16b, c, respectively. So the performance of `blacksholes` also improves. In summary, all our hybrid L2 caches outperform both 'L2-SMNGS' and 'L2-RWHCA' due to their lower read latencies.

Since the savings in leakage energy by using STT-RAM designs in the L2 cache are well established, we compared the *dynamic* energy consumptions of different L2 cache designs. The energy overheads of the data refresh in LR-region and the data migration between LR- and HR-regions in our hybrid L2 caches are included in the dynamic energy. Due to the lower write energy in the LR-region, 'L2-Hyb1' has the lowest dynamic energy consumption, as shown in Fig. 7.19a. As the STT-RAM cell size is reduced, the write latency and write energy consumption increased. Thus, the corresponding dynamic energy of 'L2-Hyb2' and 'L2-Hyb3' grows rapidly. Figure 7.19b shows the leakage energy comparison. Compared to 'L2-RWHCA' which is a combination of SRAM/STT-RAM [29], all the other configurations have much lower leakage energy consumptions. 'L2-hi', 'L2-SMNGS', and 'L2-Hyb1' have similar leakage energies because their memory array sizes are quite close to each other. However, 'L2-Hyb2' and 'L2-Hyb3' benefit from their much smaller memory cell size.

The overall cache energy consumptions of all the simulated cache configurations are summarized in Fig. 7.17. On the average, 'L2-Hyb2' and 'L2-Hyb3' consume about 70 % of the energy of 'L2-SMNGS' and 26.2 % of 'L2-RWHCA'. In summary, our proposed hybrid scheme outperforms the previous techniques in [22] and [29] both in terms of performance and in terms of total energy (by an even bigger margin).

7.5.4 Deployment in 3-level Cache Hierarchies

We also evaluate four 3-level cache designs whose parameters were given in Table 7.5. These designs are

1. The all-SRAM cache hierarchy;
2. '3L-SMNGS' that uses the "md1" STT-RAM design in all the three level of caches, just like 'L2-SMNGS' [22];

3. '3L-MultiR'—a multiretention 3-level STT-RAM cache hierarchy with "L1-lo2" , "L2-md2", and "L3-hi";
4. '3L-MultiR-Hyb'—a multiretention 3-level STT-RAM cache hierarchy with "L1-lo2", as well as the proposed hybrid cache design used in both L2 and L3 caches. Here, 'Hyb1' is used in L2 cache for the performance purpose, while 'Hyb2' is used in L3 cache to minimize the leakage energy.

In [22], the IPC performance degradations for using the single retention STT-RAM ('md1') were from 1 to 9 % when compared to an all-SRAM design. Our simulation result of '3L-SMNGS' (8 % performance degradation on average) matches this well. Comparatively, the average IPC performance degradation of '3L-MultiR' is only 1.4 % on average, as shown in Fig. 7.18. The performance gain of '3L-MultiR' over '3L-SMNGS' comes mainly from "L1-lo2". '3L-MultiR-Hyb' has the best performance which is on average 8.8 and 2.1 % better than '3L-SMNGS' and '3L-MultiR', respectively. Most of the write accesses in L2 and L3 caches of '3L-MultiR-Hyb' are allocated into the fast region, boosting up the system performance. Under the joint effort of "L1-lo2" and hybrid lower-level cache, '3L-MultiR-Hyb' can even achieve a slightly higher IPC can all-SRAM design.

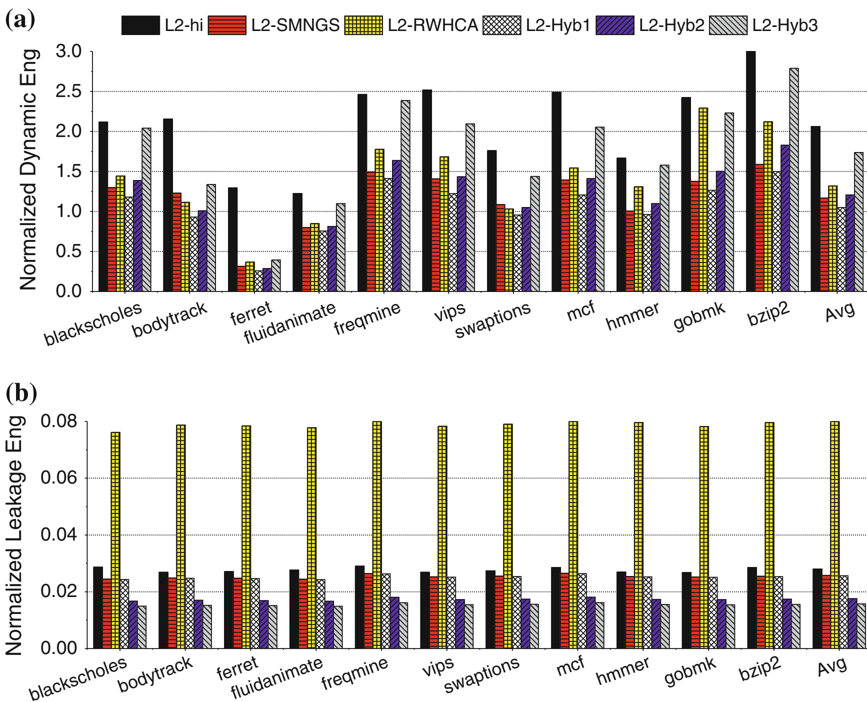

Fig. 7.17 Dynamic and leakage energy comparison of L2 cache (normalized to SRAM baseline)

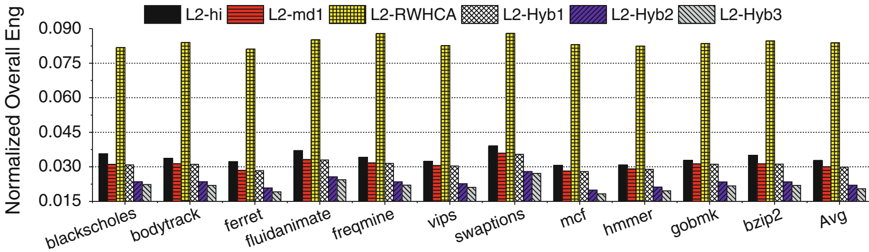

Fig. 7.18 Overall cache energy consumption comparison of 2-level cache designs (normalized to the all-SRAM design)

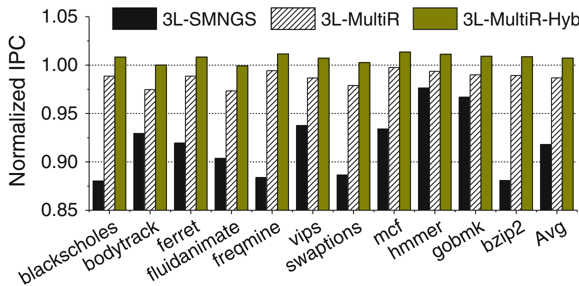

Fig. 7.19 Performance comparison of different 3-level cache designs. The IPCs are normalized to all-SRAM baseline

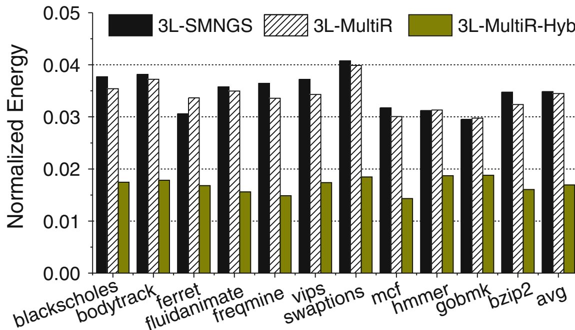

Fig. 7.20 Overall cache energy consumption comparison 3-level cache designs (Normalized to the all-SRAM design)

Normalized against an all-SRAM 3-level cache design, the overall energy comparison of 3-level cache hierarchy is shown in Fig. 7.20. All three combinations with STT-RAM save significantly more energy when compared to the all-SRAM design. ‘3L-MultiR’ saves slightly more overall energy compared to ‘3L-SMNGS’ because the ‘Lo” STT-RAM cell design has a lower per-bit access dynamic energy than the “md” design. In ‘3L-MultiR-Hyb’, shared L3 cache which embedded “md2” is much larger than local L2 cache which uses “md1”. Thereby, the leakage of L3 dominates

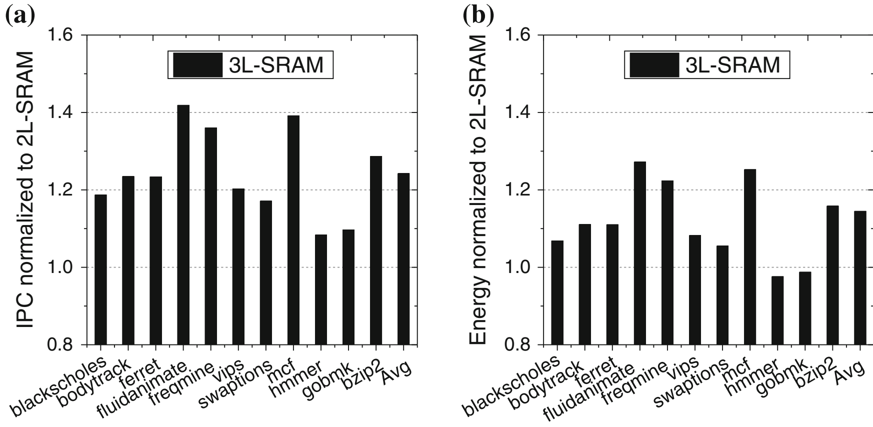

Fig. 7.21 IPC and overall cache energy comparison between 2-level and 3-level SRAM cache designs (normalized to the 2-level SRAM design)

the overall energy consumption. The leakage power ratio between "md2" and "hi" is 69.1/110 (see Table 7.5). This is why the overall energy of '3L-MultiR-Hyb' is only 60 % of '3L-MultiR' whose L3 is "hi".

7.5.5 Comparison Between 2-Level and 3-Level Cache Hierarchies

First, we directly compare 2-level and 3-level caches both implemented by SRAM. Figure 7.21a shows the IPC comparison. The 3-level SRAM cache outperforms the 2-level SRAM cache by 24.2 % in IPC performance because the 3-level cache hierarchy includes 256 KB private L2 cache within each core, enlarging the cache capacity by 32 %. Accordingly, the leakage energy increases. Figure 7.21b compares their overall energy consumptions. The total energy of 3-level SRAM cache is 14.4 % greater than that of 2-level SRAM cache.

The 2-level cache hierarchy with hybrid LR- and HR-regions ('2L-Hybrid') is compared with the 3-level multiretention STT-RAM cache hierarchy ('3L-MuliR'). With regard to IPC performance, '2L-Hybrid' is 14.36 % worse than '3L-MuliR', as shown in Fig. 7.22a. Compared to SRAM-based cache, that is to say the hybrid design actually shrinks the performance degradation between 2-level and 3-level cache hierarchies. On the one hand, since the leakage energy of STT-RAM cell is very small, the leakage energy increasing has a much smaller scalar than the growth of cache capacity. On the other hand, the access to L3 cache is filtered by L2 cache, which induces a smaller dynamic energy in '3L-MuliR' than that of '2L-Hybrid'. So the overall energy of '3L-MuliR' is not increased as 3L SRAM does. The overall energy comparison between '2L-Hybrid' and '3L-MuliR' is shown in Fig. 7.22b.

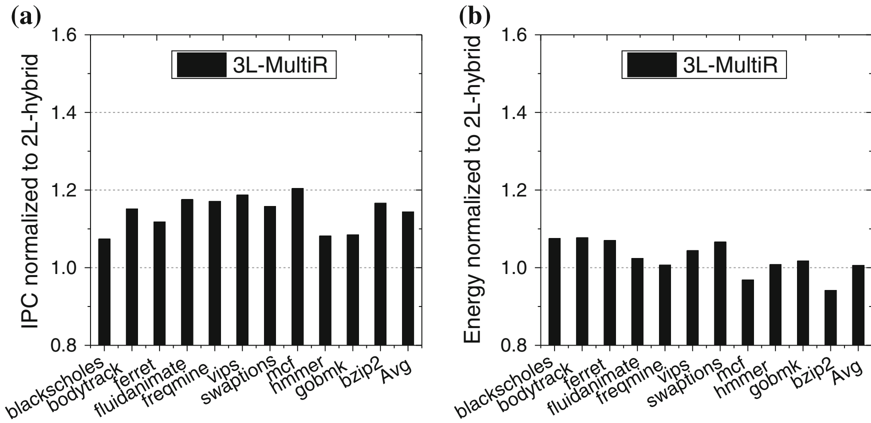

Fig. 7.22 IPC and overall cache energy comparison between 2-level and 3-level STT cache designs (normalized to the 2-level hybrid STT design)

7.6 Related Work

STT-RAM has many attractive features such as the nanosecond access time, CMOS process compatibility, and nonvolatility. The unique programming mechanism of STT-RAM—changing the MTJ resistance by passing a spin-polarized current [9]—ensures good scalability down to the 22nm technology node with a programming speed that is below 10 ns [26]. Early this year, Zhao, et al. [31] reported a subnanosecond switching at the 45 nm technology node for the in-plane MTJ devices.

Dong, et al. [8] gave a comparison between the SRAM cache and STT-RAM cache in a single-core microprocessor. Desikan, et al. [6] conducted an architectural evaluation of replacing on-chip DRAM with STT-RAM. Sun, et. al. [25] extended the application of STT-RAM cache to chip multiprocessor (CMP) and studied the impact of the costly write operation in STT-RAM on power and performance.

Many proposals have been made to address the slow write speed and high write energy of STT-RAM. Zhou et al. [32] proposed an early write termination scheme to eliminate the unnecessary writes to STT-RAM cells and save write energy. A dual write speed scheme was used to improve the average access time of STT-RAM cache that distinguishes between the fast and slow cache portions [30]. A SRAM/STT-RAM hybrid cache hierarchy and some enhancements, such as write buffering and data migration, were also proposed in [25, 29]. The SRAM and STT-RAM cache ways are fabricated on the different layers in the proposed 3D integration. The hardware and communication overheads are relatively high. None of these works considered using STT-RAM in L1 due to its long write latency.

In early 2011, Smullen et al. [22] proposed trading-off the nonvolatility of STT-RAM for write performance and power improvement. The corresponding DRAM-style refresh scheme to assure the data validity is not scalable for a large cache capacity. However, the single retention-level cache design is lack of optimization

space to maximize the benefits of STT-RAM writability and nonvolatility trade-offs. Also, the MTJ optimization technique they proposed, namely shrinking the cell surface area of the MTJ, is not efficient in the fast-switching region (< 10 ns), as discussed in Sect. 7.3.

The macromagnetic model used in our work was verified by a leading magnetic recording company and calibrated with the latest in-plane MTJ measurement results [31]. However, we note that our model was not able to reproduce the MTJ parameters given in [22], which are overly optimistic in the fast-switching region (< 3 ns) in terms of write energy and performance, as well as data retention time.

7.7 Conclusion

In this chapter, we proposed a multiretention-level STT-RAM cache hierarchy that trades off the STT-RAM cell's nonvolatility for energy saving and performance improvement. Taking into consideration the differences in data access behavior, we proposed a low-retention L1 cache with a counter-controlled refresh scheme and a hybrid structure for lower-level cache with both low- and high-retention portions. A memristor-controlled refresh scheme was proposed for the STT-RAM L1 cache to ensure data validity with the minimized hardware cost. For L2, a data migration scheme between the low- and the high-retention portions of the cache yielded fast average write latency and low standby power. Compared to the classic SRAM or a SRAM/STT-RAM hybrid cache hierarchy, our proposal uses only STT-RAM. This can save significant die cost and energy consumption. Moreover, compared to the previous STT-RAM-relaxed retention design that only has a single retention level, our design utilizes multiple retention levels, resulting in an architecture that is optimized for the data access patterns of the different cache levels.

Our experimental results show that our proposed multiretention-level STT-RAM hierarchy achieves on average a 73.8 % energy reduction over the SRAM/STT-RAM mixed design, while maintaining a nearly identical IPC performance. Compared with the previous single-level relaxed retention STT-RAM design, we obtained a 5.5 % performance improvement and a 30 % overall energy reduction by having multiple retention levels in 2-level hierarchy. The multiretention STT-RAM cache with proposed hybrid STT-RAM lower-level cache achieves on average of 6.2 % performance improvement and 40 % energy saving compared to the previous single-level relaxed retention STT-RAM design for a 3-level cache hierarchy. Compared to traditional SRAM L1 cache, the L1 cache with a ultralow-retention STT-RAM augmented by the proposed refresh scheme can achieve a 9.2 % performance improvement and a 30 % energy saving.

With technology scaling, and the increasing complexity of fabrication, we believe that our proposed cache hierarchy will become even more attractive because of its performance, low energy consumption, and CMOS compatibility.

References

1. Barth, J., Plass, D., Nelson, E., Hwang, C., Fredeman, G., Sperling, M., Mathews, A., Reohr, W., Nair, K., Cao, N. (2010). A 45 nm SOI embedded DRAM macro for POWER7TM 32 MB on-chip L3 cache. *IEEE international solid-state circuits conference digest of technical papers* (ISSCC), 342–343.
2. CACTI. http://www.hpl.hp.com/research/cacti/
3. Cao, Y., Sato, T., Orshansky, M., Sylvester, D., Hu, C. (2000). New paradigm of predictive MOSFET and interconnect modeling for early circuit design. *IEEE custom integrated circuit conference*, 201–204.
4. Chua, L. (2002). Memristor-the missing circuit element. *IEEE transastions on circuit theory*, *18*, 507–519.
5. De Sandre, G., Bettini, L., Pirola, A., Marmonier, L., Pasotti, M., Borghi, M., Mattavelli, P., Zuliani, P., Scotti, L., Mastracchio, G. (2010). A 90 nm 4 Mb embedded phase-change memory with .2 V 12 $ns read access time and$ MB/s write throughput. *IEEE international solid-state circuits conference digest of technical papers* (ISSCC), 268–269.
6. Desikan, R., Lefurgy, C. R., Keckler, S. W., Burger, D. (2008). On-chip MRAM as a high-bandwidth low-latency replacement for DRAM physical memories. http://www.cs.utexas.edu/ftp/pub/techreports/tr02-47.pdf
7. Diao, Z., Li, Z., Wang, S., Ding, Y., Panchula, A., Chen, E., Wang, L. C., Huai, Y. (2007) Spin-transfer torque switching in magnetic tunnel junctions and spin-transfer torque random access, memory (Vol. 19). p. 165209.
8. Dong, X., Wu, X., Sun, G., Xie, Y., Li, H., Chen, Y. (:2008). Circuit and microarchitecture evaluation of 3D stacking magnetic RAM (MRAM) as a Universal Memory Replacement. *ACM/IEEE desgin automation conference* (DAC), 554–559.
9. Hosomi, M., et al. (2005). A novel nonvolatile memory with spin torque transfer magnetization switching: Spin-RAM. *IEEE international electron devices meeting*, 459–462.
10. Hu, M., Li, H., Chen, Y., Wang, X., Pino, R. E. (2011). Geometry variations analysis of TiO 2 thin-film and spintronic memristors. *Proceedings of the 16th Asia and South Pacific design automation conference*, 25–30.
11. IntelQ8200. http://ark.intel.com/Product.aspx?id=36547
12. Kawahara, T., Takemura, R., Miura, K., Hayakawa, J., Ikeda, S., Lee, Y. M., et al. (2008). 2 Mb SPRAM (SPin-Transfer Torque RAM) With Bit-by-Bit bi-directional current write and parallelizing-direction current read. *IEEE Journal of solid-state circuit*, *43*, 109–120.
13. Kim, C. H., Kim, J. J., Mukhopadhyay, S., & Roy, K. (2005). A forward body-biased low-leakage SRAM cache: device, circuit and architecture considerations. *IEEE transations on very large scale integration* (VLSI) *system*, *13*, 349–357.
14. Kirolos, S., Massoud, Y. (2007). Adaptive SRAM design for dynamic voltage scaling VLSI systems. *IEEE international midwest symposium on circuits and systems* (MWSCAS), 1297–1300.
15. Li, Z., Zhang, S. (2004). Domain-wall dynamics driven by adiabatic spin-transfer torques. *Physics Review B*, *70*, 024417.
16. Marss86. http://www.marss86.org/
17. Nair, P., Eratne, S., John, E. (2007). A quasi-power-gated low-leakage stable SRAM cell. *IEEE international midwest symposium on circuits and systems* (MWSCAS), 761–764.
18. Nebashi, R., Sakimura, N., Tsuji, Y., Fukami, S., Honjo, H., Saito, S., Miura, S., Ishiwata, N., Kinoshita, K., Hanyu, T. (2011). A content addressable memory using mMagnetic domain wall motion cells. *Symposium on VLSI circuits*, 300–301.
19. NVSim. http://www.rioshering.com/nvsimwiki/index.php
20. Parkin, S. (2009). Racetrack memory: A storage class memory based on current controlled magnetic domain wall motion. *Device research conference*, 3–6.
21. Raychowdhury, A., Somasekhar, D., Karnik, T., De, V. (2009). Design space and scalability exploration of 1T–1STT MTJ memory arrays in the presence of variability and disturbances. *IEEE international electron devices meeting*, 1–4.

22. Smullen, C. W., Mohan, V., Nigam, A., & Gurumurthi, S., Stan, M. R. (2011). Relaxing non-volatility for fast and energy-efficient STT-RAM caches.
23. Strukov, D. B., Snider, G. S., Stewart, D. R., & Williams, R. S. (2008). The missing memristor found. *Nature*, *453*, 80–83.
24. Sun, J. Z. (2000). Spin-current interaction with a monodomain magnetic body: A model study. *Physics Review B*, 62.
25. Sun, G., Dong, X., Xie, Y., Li, J., Chen, Y. (2009). A novel architecture of the 3D stacked MRAM L2 cache for CMPs. *IEEE symposium on high-performance computer architecture* (HPCA), 239–249.
26. Wang, X., Chen, Y., Li, H., Dimitrov, D., Liu, H. (2008). Spin torque random access memory down to 22 nm technology (Vol. 44). *IEEE transactions on magnetics*, 2479–2482.
27. Wang, X., et al. (2009). Spintronic Memristor through Spin Torque Induced Magnetization Motion. *IEEE electron device letters*, *30*, 293–297.
28. Wang, X., Zhu, W., Xi, H., & Dimitrov, D. (2008). Relationship between symmetry and scaling of spin torque thermal switching barrier. *IEEE transactions on magnetics*, *44*, 2479–2482.
29. Wu, X., Li, J., Zhang, L., Speight, E., & Xie, Y. (2009). Power and performance of read-write aware hybrid caches with non-volatile memories. *Design, automation and test in Europe conference and exhibition* (pp. 737–742).
30. Xu, W., et al. (2011). Design of last-level on-chip cache using spin-torque transfer RAM (STT RAM). *IEEE transations on very large scale integration* (VLSI) *system*, *483–493*, 2011.
31. Zhao, H. and Lyle, A. and Zhang, Y. and Amiri, PK and Rowlands, G. and Zeng, Z. and Katine, J. and Jiang, H. and Galatsis, K. and Wang, KL.: Low writing energy and sub nanosecond spin torque transfer switching of in-plane magnetic tunnel junction for spin torque transfer RAM, *Journal of Applied Physics*, *109*, 07C720.
32. Zhou, P., Zhao, B., Yang, J., Zhang, Y. (2009). Energy reduction for STT-RAM using early write termination. *IEEE/ACM international conferernce on computer-aided design* (ICCAD), 264–268.
33. ITRS (2011). The International Technology Roadmap for Semiconductors. http://www.itrs.net.