A Novel L1 Cache Based on Volatile STT-RAM

Zhang Hongguang[1(✉)] and Zhang Minxuan[1,2]

[1] College of Computer, National University of Defense Technology, Changsha 410073, People's Republic of China
{zhanghongguang14,mxzhang}@nudt.edu.cn
[2] National Key Laboratory of Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, People's Republic of China

Abstract. Spin-transfer torque random access memory (STT-RAM) is one of the most promising substitutes for universal main memory and cache due to its excellent scalability, high density and low leakage power. Nevertheless, the current non-volatile STT-RAM cache architecture also has some drawbacks, such as long write latency and high write energy, which limit the application of STT-RAM in the top level cache design. To solve these problems, we relax the retention time of STT-RAM to explore its different write performance, and propose a novel STT-RAM L1 cache architecture implemented with volatile STT-RAM as well as its related refresh scheme. The performance of proposed design is the same as SRAM L1 cache while its overall power consumption is only 63.8% of the latter one.

Keywords: STT RAM · L1 cache · Volatile · Refresh scheme

1 Introduction

SRAM has been the mainstream technology of caches for years due to its high access speed, low dynamic power and other good features. However, with more and more cores are embedded on chip, caches need larger size. However, increasing capacity of SRAM cache leads to high leakage power, which will bring in a serious on-chip heat sink problem. So researchers are focusing on alternative substitutes for SRAM.

STT-RAM is regarded as the most promising replacement for SRAM because it owns almost all desired features of an universal memory and cache, including high storage density, fast read speed and non-volatility. However, there are two drawbacks, namely, long write latency and high write energy, which limit the application of STT-RAM in L1 cache design. In [2–4, 6, 10], there are some efficient schemes proposed to overcome the two drawbacks when applying STT-RAM in cache design, such as relaxing the non-volatility and hybrid cache design.

To overcome the two problems, we propose to relax the non-volatility of STT-RAM to gain a significant optimization in performance and power consumption. In addition, we design the related refresh scheme to improve the cache's reliability. We simulate the proposed L1 cache architecture on GEM5 simulator, and collect the simulation results to analysis its overall performance.

W. Xu et al. (Eds.): NCCET 2016, CCIS 666, pp. 32–39, 2016.
DOI: 10.1007/978-981-10-3159-5_4

2 STT-RAM Features

The basic storage cell of STT-RAM is magnetic tunnel junction (MTJ) shown in Fig. 1. There are two magnetic layers in a MTJ, namely, free layer and reference layer. They are isolated by an oxide layer. The magnetic direction of reference layer is fixed, however, that of free layer can be switched by current. If the directions of the two layers are parallel, the MTJ is in low-resistance state; if they are anti-parallel, the MTJ is in high-resistance state.

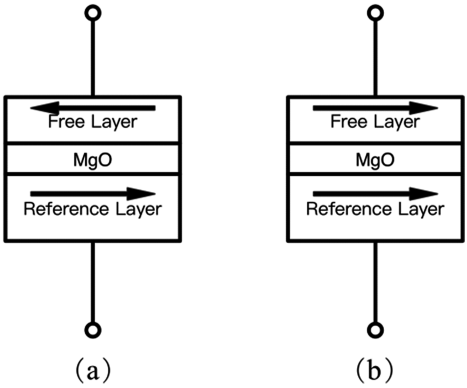

Fig. 1. The MTJ design (1T1J). (a) High-resistance state. (b) Low-resistance state.

The MTJ's non-volatility can be analyzed quantitatively with the retention time of MTJ. We use τ to represent its retention time. τ is related to the thermal stability factor Δ and can be calculated with Eq. (1) [1].

$$\tau \approx \tau_0 exp(\Delta) \tag{1}$$

τ_0: The attempt time and set as 1 ns.

Δ is derived from Eq. (2).

$$\Delta = \frac{E_F}{k_B T} = \frac{M_s V H_K}{2 k_B T} \tag{2}$$

M_s: The saturation magnetization.
H_k: The effective anisotropy field.
T: The working temperature.
k_B: The Boltzmann constant.
V: The volume for the STT-RAM write current.

From Eqs. (1) and (2), we can know that the data retention time of a MTJ decreases exponentially when its working temperature T increases.

According to the different T_w, MTJ has three regions, namely, the thermal activation, dynamic reverse and processional switching. We relax the MTJ's retention time to 30 μs ($\Delta = 10.3$) and adjust the T_w to get three different design options. In this paper they are called LRS1, LRS2 and LRS3 respectively [2].

3 STT-RAM LLC Design

3.1 Performance Parameters

Based on the analysis in Chapter 2, we simulate the performance of three LRS STT-RAM designs on NVSim [5]. The results are shown in Table 1.

Table 1. The parameters for multi-retention STT-RAM cells.

Parameters	SRAM	LRS1	LRS2	LRS3
Area/F^2	125	21	22	40
Switching time/*ns*	/	2.0	1.5	1.0
Retention time	/	30 μs	30 μs	30 μs
Read latency/*ns*	1.125	0.780	0.856	0.981
Read energy/*nJ*	0.075	0.083	0.087	0.099
Write latency/*ns*	1.091	2.363	1.889	1.497
Write energy/*nJ*	0.059	0.177	0.180	0.197
Leakage power/*mW*	24.8	1.74	1.99	1.81

From Table 1, we find that if we relax the retention time of MTJ to μs level, its access speed is almost the same with SRAM. The gap between their performance is not very large, however, the area of LRS3 is much larger than the other two designs.

Based on the above analysis, the LRS2 STT-RAM has a better overall performance and it is used in our cache design.

3.2 L1 Cache Architecture

In 2 GHz processor system, LRS2's read latency is 2 cycles, and its write latency is 3 cycles while both the two parameters of SRAM are 3 cycles. However, the retention time of LRS2 is only 30 us, which is shorter than the write interval of many blocks in L1 cache. It means that many data will be invalid if we do not take any measures.

So we propose to add two counters, namely, the refresh-counter and the access-counter, for every LRS2 block in L1 cache. The refresh-counter is used to monitor the duration time that the data have been stored in volatile blocks. We divide the retention time of STT-RAM to N_R, which is the maximum value of refresh-counter. Generally, we set $N_R = 15$ for 32 KB L1 cache. The maximum value of the access-counter is 8. So the refresh-counter is 4 bits, and the access-counter is 3 bits. The architecture is shown as Fig. 2.

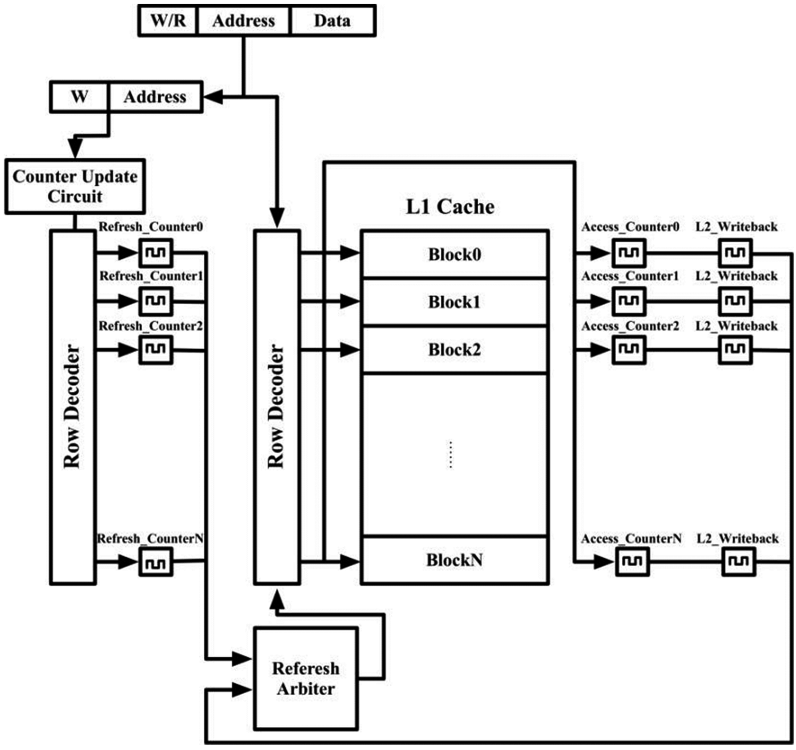

Fig. 2. The L1 cache architecture

All counters are controlled by a global clock, whose period is $\boldsymbol{T_{gc} = \tau/N_R}$. The access-counter is used to record the read access number in recent 5 $\boldsymbol{T_{gc}}$ because most data in L1 are accessed again shortly since it is written into a block. If one block is not accessed in 5 $\boldsymbol{T_{gc}}$, we think that it will no longer be accessed again in its life span. The L2_Writeback bit is used to monitor that if the last write operation is a writeback operation from L2 cache. If it is, we set its value to 1, otherwise set it to 0.

To complete the refresh operation, the data stored in a LRS block will be extracted to the buffer firstly, then is written back to the block again. If a write request comes during this process, we stop the refresh process and execute the write operation; if a read request comes, it gets data form the buffer directly and does not need to wait for the completion of refresh process. The duration of the whole process is about 5 cycles 2 GHz CPU system, which is much smaller than the retention time of LRS2 STT-RAM, so it is not necessary to consider the refresh duration in the calculation of N_R and $\boldsymbol{T_{gc}}$.

At the end of $\boldsymbol{T_{gc}}$, all counters are increased by 1. Both the refresh-counter and the access-counter of a LRS2 block will be reset to 0 if a write access is executed, however, the access-counter will also be reset to 0 for a read access. When a refresh-counter reaches N_R, we detect the value of access counter, if it is lower than 5, we continue the refresh

process; if it is higher than 5, we detect the value of L2_Writeback. If it is 0, we write its data back to L2 cache and invalidate it in L1 cache, otherwise we only invalidate it.

The hardware overhead is (4 bits × 2)/(64 bytes) = 1.56%. Based on simulation results, these counters' power consumption takes up only less than 6% of the overall dynamic power consumption, which has little influence on the overall performance.

4 Simulation

4.1 Experimental Setup

In this article, we use GEM5 [7, 8] to conduct the architectural experiment to test the overall system performance of LRS STT-RAM. The configuration is shown in Table 2. The benchmarks are selected from SPEC CPU 2006 [10].

Table 2. GEM5 configuration

Computer system	Configuration
CPU	X86, O3, 2 GHz
L1 Icache	Private, 32 KB, 2-way
L1 Dcache	Private, 32 KB, 2-way
L2 cache	Private, 256 KB, 8-way
L3 cache	Shared, 1 MB, 16-way
Main memory	1024 MB, 1-channel

4.2 Architectural Simulation

The performance is measured by instructions per cycle (IPC), and the results are shown as Fig. 3. It can be seen that LR3 owns the best performance (at 100.5%) while LRS1's performance is the lowest one (at 99.1 %). The IPC of LRS2 is the same with SRAM.

The leakage power consumption results are shown as Fig. 4. It can be seen that the leakage power of LRS3 is the highest one, at 18.6% on average, however, the LRS1 share the lowest one, at 13.9%. The LRS2's leakage power is at 16.0%. The leakage power consumption is doubled because of the introduction of the refresh-buffer. However, the total value is still much less than that of SRAM cache.

The dynamic power consumption results are shown as Fig. 6, which includes the refresh energy. We find that the dynamic power of LRS3 is also the highest, at 101.1%, which is followed by LRS2, at 87.3%. The dynamic power of LRS1 is the lowest one, at 80.4% (Fig. 5).

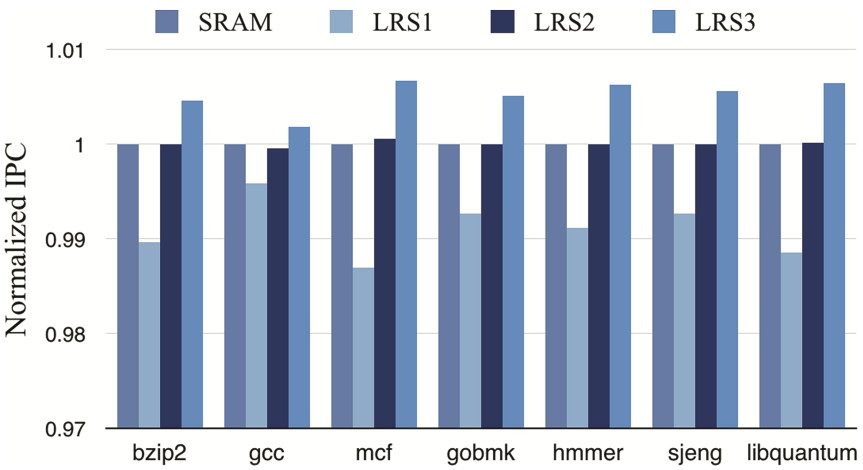

Fig. 3. The IPC test results

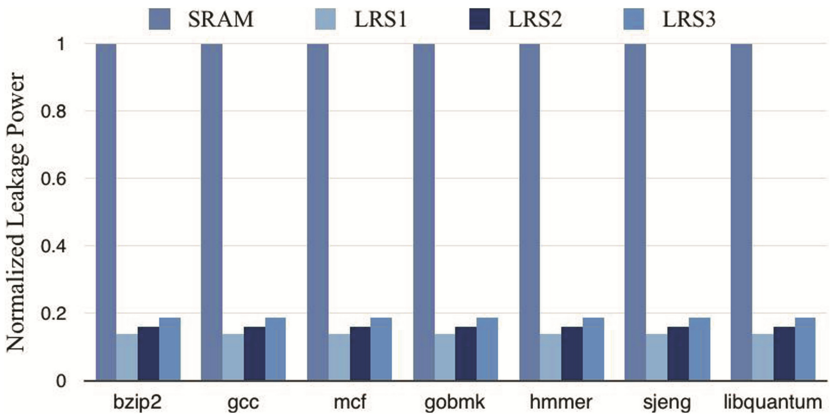

Fig. 4. The total leakage power consumption

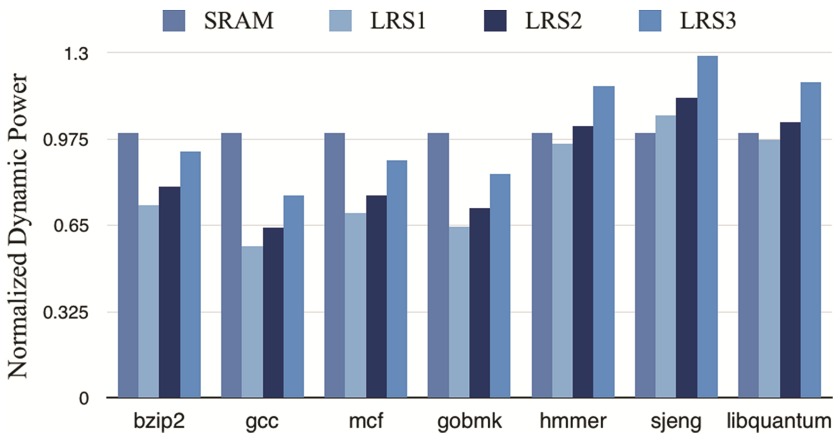

Fig. 5. The total dynamic power consumption

The overall power consumption shown in Fig. 6 is the sum of leakage and dynamic power consumption. It is clear that LRS1's power consumption is only the half of SRAM, at 58.6%. LRS2 is a bit higher than LRS1, at 63.8%. Compared with the two design, LRS3's power consumption is the highest one, at 73.9%.

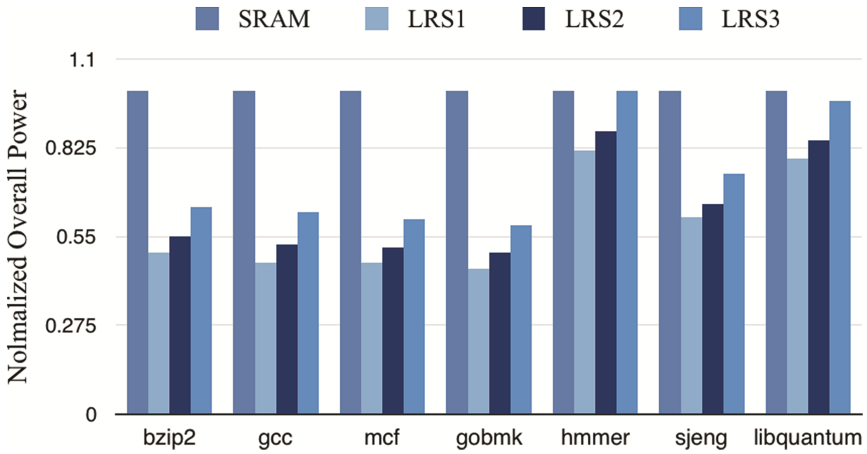

Fig. 6. The overall power consumption

5 Conclusion

In this paper, we propose a novel L1 cache architecture based on volatile STT-RAM to improve the reliability and save energy. Our simulation results show that the proposed volatile STT-RAM L1 cache has the same overall performance with SRAM, while

having only 63.8% power consumption. In addition, the total on-chip area of proposed L1 cache can be saved by 64.8% ideally.

Acknowledgements. The project is sponsored by National Science and Technology Major Project, "The Processor Design for Super Computer" (2015ZX01028) in China and the Excellent Postgraduate Student Innovation Program (4345133214) of National University of Defense Technology.

References

1. Jog, A., Mishra, A. K., et al.: Cache revive: architecting volatile STT-RAM caches for enhanced performance in CMPs. In: IEEE Design Automation Conference, pp. 243–253 (2012)
2. Sun, Z., Bi, X., et al.: STT-RAM cache hierarchy with multiretention MTJ design. IEEE Trans. Very Large Scale Integr. Syst. **22**(6), 1281–1294 (2014)
3. Smullen, C., Mohan, V., et al.: Relaxing non-volatility for fast and energy-efficient STT-RAM caches. In: IEEE Symposium on High-Performance Computer Architecture, pp. 50–61 (2011)
4. Li, J., Shi, L., et al.: Low-energy volatile STT-RAM cache design using cache-coherence-enabled adaptive refresh. ACM Trans. Des. Autom. Electron. Syst. **19**(1), 1–23 (2013)
5. NVSim. http://www.rioshering.com/nvsimwiki/index.php
6. Li, Q., Li, J., et al.: Compiler-assisted STT-RAM-based hybrid cache for energy efficient embedded systems. IEEE Trans. Very Large Scale Integr. Syst. **22**(8), 1829–1840 (2014)
7. Binkert, N., Beckmann, B., et al.: The gem5 simulator. ACM SIGARCH Comput. Architect. News **39**(2), 1–7 (2011)
8. Gem5. http://gem5.org
9. Ahn, J., Yoo, S., et al.: Write intensity prediction for energy-efficient non-volatile caches. In: IEEE International Symposium on Low Power Electronics and Design, pp. 223–228 (2013)
10. Standard Performance Evaluation Corporation. http://www.spec.org