LETTER

Performance and Energy Efficiency Tradeoffs of Storage Class Memory*

Heekwon PARK[†], Member, Seungjae BAEK^{†a)}, and Jongmoo CHOI[†], Nonmembers

SUMMARY The traditional mobile consumer electronics such as media players and smart phones use two distinct memories, SDRAM and Flash memory. SDRAM is used as main memory since it has characteristic of byte-unit random accessibility while Flash memory as secondary storage due to its characteristic of non-volatility. However, the advent of Storage Class Memory (SCM) that supports both SDRAM and Flash memory characteristics gives an opportunity to design a new system configuration. In this paper, we explore four feasible system configurations, namely RAM-Flash, RAM-SCM, SCM-Flash and SCM-Only. Then, using a real embedded system equipped with FeRAM, a type of SCM, we analyze the tradeoffs between performance and energy efficiency of each configuration. Experimental results have shown that SCM has great potential to reduce energy consumption for all configurations while performance is highly application dependent and might be degraded on the SCM-Flash and SCM-Only configuration.

key words: storage class memory, new memory hierarchy, performance, energy consumption

1. Introduction

Recently, new types of revolutionary memories such as FeRAM (Ferroelectric RAM), MRAM (Magnetoresistive RAM) and PCRAM (Phase-Change RAM) are being actively developed [1], [2], and some of them have already been shipped into market [3], [4]. These memories share two common characteristics, one is byte-unit random accessibility and the other is non-volatility. Thus, they are collectively called as Storage Class Memory (SCM).

The advent of SCM gives an opportunity to design a new system configuration for mobile consumer electronics. Traditionally, mobile consumer electronics such as portable media player and smart phones use SDRAM as main memory and Flash memory as secondary storage. However, the characteristic of byte-unit random accessibility makes it possible to apply SCM for main memory instead of SDRAM. Furthermore, the non-volatile characteristic enables to apply SCM for secondary storage as a substitute of Flash memory. When we go one step further, we can use SCM as both main memory and secondary storage simultaneously and exclude SDRAM and Flash memory, leading to simplify system organizations.

Another advantage of employing SCM on mobile consumer electronics is energy saving since SCM consumes

less power than SDRAM and Flash memory [5]. However, such employment needs to judiciously consider other aspects of SCM in terms of performance, capacity, and cost. The main goal of this study is investigating how the different combinations of SDRAM, Flash memory, and SCM result in various performance and energy saving tradeoffs.

In this paper, we explore four feasible system configurations, namely RAM-Flash, RAM-SCM, SCM-Flash and SCM-Only configuration. The RAM-Flash configuration corresponds to the traditional mobile consumer electronics utilizing SDRAM as main memory and Flash memory as secondary storage. The RAM-SCM and SCM-Flash configuration employ SCM as main memory and secondary storage, respectively. The final SCM-Only configuration uses SCM as both main memory and secondary storage simultaneously. We actually have built an experimental platform consisting of a 400 MHz XScale CPU, 64 MB SDRAM, 64 MB NAND Flash memory, 0.5 MB NOR flash memory, and 64 MB FeRAM and have materialized the four configurations.

Then, by executing various applications, we have evaluated quantitatively how the configurations affect performance and energy consumption. Empirical observations have shown that the introduction of SCM can enhance power efficiency by a factor of 2 to 8 and the SCM-only organization saves energy the most. However, performance is highly application dependent. Specifically, when we utilize SCM as main memory, the response time is degraded by a factor of 2.2, especially for a memory intensive application, due to the differences of access latencies between FeRAM and SDRAM. On the contrary, a CPU intensive application does not show any marginal performance differences and a file intensive application can obtain performance gains since FeRAM has relatively faster latency than Flash memory. We also have observed that even the degraded memory intensive application can guarantee the Quality of Service (QoS) of MPEG decoding required in the embedded system considered in this paper.

The remainder of the paper is organized as follows. In Sect. 2, we describe the characteristics of SCM and related works. Then, we present four system configurations and our experimental platform in Sect. 3. In Sect. 4, we discuss the results mainly in terms of performance and energy efficiency. Finally, we conclude our work in Sect. 5.

Manuscript received May 25, 2010.

[†]The authors are with the Dankook University, Korea.

*An earlier version of this paper appeared in the Proceedings of the International Conference on Consumer Electronics (ICCE), 2009.

a) E-mail: baeksj@dankook.ac.kr

DOI: 10.1587/transinf.E93.D.3112

2. Characteristics of SCM and Related Works

Table 1 represents the characteristics of SDRAM, Flash memory and SCM including FeRAM, MRAM and PCRAM considered in this paper [3]–[5]. Like SDRAM, SCM has byte-unit random accessibility and there are several noticeable studies using SCM as an extension of main memory. Zhou et al. employed PCM as main memory for a 3D stacked chip and investigated techniques that remove redundant writes from memory to improve durability and energy efficiency [5]. Lee et al. exploited the scalability of PCM as a DRAM alternative and devised novel buffer reorganizations [6] and Qureshi et al. designed DRAM-PCM hybrid main memory [7].

Also, SCM has non-volatility just like Flash memory. Studies using SCM as an extension of secondary storage have been presented. Wang et al. proposed a file system, Conquest, that provides a transition from disk to SCM as the primary storage medium [8]. The HeRMES file system presented by Miller et al. makes use of MRAM to store meta-data, while storing file data in disks [9]. Condit et al. presented a file system and a hardware architecture to provide atomic, fine-grained updates for persistent, byte addressable memory [10].

Most of these studies focused on PC and server computing environments while ours focuses on mobile consumer electronics, evaluating performance and energy consumption on a real embedded board that can be used for portable media players and smart phones. Also, our study is different from previous ones in that we exploit a new configuration that uses SCM both for main memory and secondary storage simultaneously.

Table 1 also shows that SCM has a potential to reduce energy consumptions. In terms of performance, SCM has relatively faster latency than Flash memory, but slower than SDRAM. In general, they say that SCM is 26 times faster than Flash memory but 4~8 times slower than DRAM [7]. These gaps imply that by replacing Flash memory with SCM, we have a chance to improve performance, but replacing SDRAM with SCM may cause performance degradation.

More interesting case is exploiting SCM not only for main memory but also secondary storage, where the mixed effects affect the overall performance, which in turn affects the total energy consumption. To the best of our knowledge, this is the first study that evaluates the tradeoff of per-

formance and energy efficiency among various SCM usages quantitatively in a real embedded system.

3. New System Configurations

Figure 1 shows a hardware platform that we have built for our study. The main board shown at the bottom of the figure consists of 400 MHz XScale CPU, 64 MB SDRAM (32 MB / chip * 2 chips), 64 MB Flash NAND memory (32 MB / chip * 2 chips), 0.5 MB NOR Flash memory and embedded peripherals such as UART, LCD, and JTAG. SDRAM is used as main memory for dynamic program execution space, while Flash memory is used as storage for file data. On this main board, we have also constructed a daughter board with 32 MB FeRAM (0.5 MB / chip * 64 chips), a type of SCM, as shown at the top of the figure.

In this platform, various combinations of RAM, Flash memory, and SCM may be possible as a system configuration. We investigate four feasible combinations, and refer to them as RAM-Flash, RAM-SCM, SCM-Flash, and SCM-Only configuration, respectively, as shown in Fig. 2.

The RAM-Flash configuration utilizes SDRAM as main memory and Flash memory as secondary storage, which is the typically architecture of contemporary mobile consumer electronics. The RAM-SCM configuration replaces Flash memory with SCM and utilizes it as secondary storage, while the SCM-Flash configuration substitutes SDRAM with SCM as main memory. Finally, the SCM-Only configuration utilizes SCM for both main memory and secondary storage. In actuality, other combinations such as hybrid main memory with SDRAM and SCM, examined in Qureshis study [7], and hybrid storage with Flash

Fig. 1 Experimental platform with FeRAM.

Fig. 2 New feasible system configurations using SCM.

Table 1 Comparisons of memory technologies.

	Non Volatility	Random Accessibility	Access Time	Operating Current
SDRAM	No	Yes	~7.5 ns	105~155 mA
NAND Flash	Yes	No	20~200 us	25~35 mA
FeRAM	Yes	Yes	~55 ns	~18 mA
PCM	Yes	Yes	48~150 ns	200~600 uA
MRAM	Yes	Yes	~35 ns	~20 mA

memory and SCM, inspected by Wang's work [8], are feasible, but we mainly focus the four combinations in this paper.

To quantitatively evaluate the energy consumption and performance of the four configurations, we have ported the Linux kernel version 2.6.24, where main memory (either SDRAM or SCM) is managed by the conventional buddy system and secondary storage (either NAND Flash memory or SCM) is managed by the FAT file system. For the SCM-Only organization, a half is used for main memory and the rest for secondary storage.

We have executed three applications, namely Postmark, MPEG coding, and DES encryption and decryption, and have measured the response time and consumed energy of each application. We chose these applications to represent file intensive, memory intensive, and computation intensive applications, respectively. In this experiment, we execute Postmark as the default setting and MPEG decoding and DES applications with a 4.4 MB file. To measure the energy consumption of applications, we used the Micron system power calculator [11] and multiplied the measured power value with the application elapsed time.

4. Results and Analysis

In the first set of experiments, we measured the response time of three applications on the four combinations, as shown in Fig. 3. From the results, we can observe the followings:

- Postmark performs six times faster in the RAM-SCM configuration than in the traditional RAM-Flash configuration. This is because SCM is faster than Flash memory and SCM storage is the best solution to boost file intensive applications among the four configurations.
- SCM as main memory (SCM-Flash and SCM-Only configuration) shows worse performance than RAM-Flash configuration because SCM is slower than SDRAM. It is more evident in MPEG decoding results, which is a memory intensive application, where two configurations with SDRAM main memory are about two times faster than the other two configurations with SCM main memory.
- DES, which is a computation intensive application, does not show any marginal performance differences

Fig. 3 Performance evaluation results.

among the four combinations. We conjecture that the effects of the CPU cache can hide the memory latency differences between SDRAM and SCM.

- In the SCM-Only configuration, the gains of storage aspect and the degradations of memory aspect of SCM affect results as the same time. As the results, applications in SCM-Only shows better performance than in SCM-Flash while they performs worse in RAM-Flash.

When we compare the results of the SCM-Only configuration with them of the baseline RAM-Flash configuration, we can find out that performance is highly application dependent. By employing SCM, the file intensive application obtains performance gains while the memory intensive application suffers from performance degradation. On the other hand, the computation intensive application shows similar performance.

Note that the SCM-Only configuration provides a new possibility to enhance performance at the operating system level. For instance, metadata update can be carried out in place without data copying between main memory area and storage area. Another practicable scheme is program execution in place where a thread maps its address space directly into the ELF (Executable Linking Format) file. These issues are beyond the scope of this paper and left as future works.

One question that arises in the experiment is that the degraded memory intensive application can still satisfy the QoS requirements of mobile consumer electronics. To investigate the question, we measured the number of frames per second decoded by the MPEG application as represented in Fig. 4.

In the figure, about 386~401 frames are decoded per second when SDRAM is used for main memory. However, it drops to 186~189 frames per second when SCM is used for main memory. In general, MP3 player requires a minimum of 38 frames per second, which implies that even the degraded memory intensive application can guarantee the QoS of MPEG decoding required in the system considered in the paper.

In terms of energy efficiency, the four configurations show quite a difference in power consumption results regardless of the application properties, as seen in Fig. 5. Specifically, if SCM is used for main memory, the power consumption of the applications reduce to roughly a third

Fig. 4 Frames per second decoded by MPEG application.

Fig. 5 Energy consumption evaluation results.

and, if SCM is used for both main memory and storage, the power consumption of applications plunge to a eighth of the SDRAM-Flash configuration. Note that even the MPEG decoding application that shows longer response times gives energy efficiency.

From the results, we can find out that SCM has great potential to reduce energy consumption. Also, though SCM might degrade performance especially the memory intensive application, judicious designs can guarantee the QoS required by the considered consumer electronics. Furthermore, the SCM-Only combination can simplify embedded system design by merging main memory and the storage device, which could lead to cost-effective system organizations. We also think that, though SCM for main memory might deteriorate performance, arranging frequently referenced code and data in small amount of SDRAM and others in SCM may keep performance comparable to pure SDRAM memory while saving energy.

Until now, we mainly discuss about performance and energy issues. Another consideration of SCM deployment includes capacity, cost, and endurance issues. With today's SCM capacity and price, deploying SCM in consumer electronics looks infeasible. However, many experts agree that the scalability of SCM is better than that of DRAM [2], [7]. In the market, Samsung provides a prototype of PCM with 512 Mb in 2009 and announces its capacity will be increased twice a year [10]. The projections of SCM cost is that the cost/MB for PCM will be close to that of enterprise disks by the year 2012 [2]. Also, several schemes to enhance the endurance of SCM have been proposed [5], [6]. Considering these trends, various combinations of SDRAM, Flash memory and SCM are really feasible in consumer electronics with an affordable cost that can save energy substantially while satisfying performance requirement.

5. Conclusion

In this paper, we have explored a new memory hierarchy when employing SCM and examined tradeoffs between performance and energy efficiency. Experimental results based on various applications have shown that SCM can save energy greatly but may sacrifice performance. We have also presented the new SCM-Only configuration that can save energy the most and give a new chance to simplify the whole

system organizations and to apply new performance improving techniques at the operating system level.

We will extend our study in two directions. One direction is in developing efficient algorithms appropriate for SCM. In current experiments, we just adopt the conventional FAT and buddy algorithms, but we conjecture the SCM optimized algorithms and data structures are different from those for conventional SDRAM or disks. Especially, we are interested in the operating system supports that can utilize both characteristics of SCM, simultaneously. The second direction is designing new integrated configurations that use SDRAM and SCM as hybrid main memory and/or Flash memory and SCM as hybrid secondary storage and providing a spectrum of system organizations that balance performance, energy consumption and cost factors.

Acknowledgments

This work was supported by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korea government (MEST) (No. 2009-0085883).

References

- [1] G.W. Burr, B.N. Kurdi, J.C. Scott, C.H. Lam, K. Gopalakrishnan, and R.S. Shenoy, "Overview of candidate device technologies for storage-class memory," *IBM J. Res. Dev.*, vol.52, no.4, pp.449–464, 2008.
- [2] R.F. Freitas and W.W. Wilcke, "Storage-class memory: The next storage system technology," *IBM J. Res. Dev.*, vol.52, no.4, pp.439–447, 2008.
- [3] Everspin inc., MRAM specification, "<http://everspin.com>."
- [4] Ramtron inc., FeRAM specification, "<http://ramtron.com>."
- [5] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "A durable and energy efficient main memory using phase change memory technology," *ISCA '09: Proc. 36th Annual International Symposium on Computer Architecture*, pp.14–23, ACM, New York, NY, USA, 2009.
- [6] B.C. Lee, E. Ipek, O. Mutlu, and D. Burger, "Architecting phase change memory as a scalable dram alternative," *ISCA '09: Proc. 36th Annual International Symposium on Computer Architecture*, pp.2–13, ACM, New York, NY, USA, 2009.
- [7] M.K. Qureshi, V. Srinivasan, and J.A. Rivers, "Scalable high performance main memory system using phase-change memory technology," *ISCA '09: Proc. 36th Annual International Symposium on Computer Architecture*, pp.24–33, ACM, New York, NY, USA, 2009.
- [8] A.I. Wang, P.L. Reiher, G.J. Popek, and G.H. Kuenning, "Conquest: Better performance through a disk/persistent-ram hybrid file system," *ATEC '02: Proc. General Track of the Annual Conference on USENIX Annual Technical Conference*, pp.15–28, USENIX Association, Berkeley, CA, USA, 2002.
- [9] E. Miller, S. Brandt, and D. Long, "Hermes: High-performance reliable mram-enabled storage," *Hot Topics in Operating Systems*, 2001, Proc. Eighth Workshop, pp.95–99, May 2001.
- [10] J. Condit, E.B. Nightingale, C. Frost, E. Ipek, B. Lee, D. Burger, and D. Coetsee, "Better i/o through byte-addressable, persistent memory," *SOSP '09: Proc. ACM SIGOPS 22nd Symposium on Operating Systems Principles*, pp.133–146, ACM, New York, NY, USA, 2009.
- [11] "Micron, "micron system power calculator," <http://www.micron.com>."