ELectronics EXpress

LETTER

IEICE Electronics Express, Vol.13, No.13, 1–7

Accelerating master-slave databases launched on LDPC-induced solid state drives with improved efficiency and reliability

Jinhua Cui[1a)], **Weiguo Wu**[1], **Nianjun Zou**[1], **and Yinfeng Wang**[2]

[1] *Department of Electronic and Information Engineering, Xi'an Jiaotong University, Shaanxi, China*

[2] *Department of Software Engineering, ShenZhen Institute of Information Technology, Guangdong, China*

a) *cjhnicole@gmail.com*

Abstract: An optimized latency model and a write speed adjustment scheme are proposed to accelerate master-slave databases launched on LDPC-induced solid state drives. To the best of our knowledge, no prior work has been published to systematically study the utilization of read/write latency model on LDPC-induced master-slave databases and the impact of fast write operations on databases reliability. Experimental results show that the proposed MFW-SFR approach outperforms traditional ones with a 57.11% reduction of average read latency time on slave sites and a 33.23% reduction of average write latency time on master sites.

Keywords: solid state drives, master slave database, latency model, reliability, LDPC, RBER

Classification: Circuits and modules for storage

References

[1] C. Lee, *et al.*: "F2FS: A new file system for flash storage," FAST Dig. Tech. Papers (2015) 273.

[2] N. Agrawal, *et al.*: "Design tradeoffs for SSD performance," USENIX ATC (2008) 57.

[3] R. S. Liu, *et al.*: "Optimizing NAND flash-based SSDs via retention relaxation," FAST Dig. Tech. Papers (2012) 125.

[4] Y. Pan, *et al.*: "Exploiting memory device wear-out dynamics to improve NAND flash memory system performance," FAST Dig. Tech. Papers (2011) 245.

[5] G. Dong, *et al.*: "Enabling nand flash memory use soft-decision error correction codes at minimal read latency overhead," IEEE Trans. Circuits Syst. I, Reg. Papers **60** (2013) 2412 (DOI: 10.1109/TCSI.2013.2244361).

[6] K. C. Ho, *et al.*: "A 45 nm 6b/cell charge-trapping flash memory using LDPC-based ECC and drift-immune soft-sensing engine," ISSCC Dig. Tech. Papers (2013) 222 (DOI: 10.1109/ISSCC.2013.6487709).

[7] K. Zhao, *et al.*: "LDPC-in-SSD: making advanced error correction codes work

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016

effectively in solid state drives," FAST Dig. Tech. Papers (2013) 243.
[8] Q. Li, *et al.*: "Maximizing IO performance via conflict reduction for flash memory storage systems," DATE (2015) 904.
[9] UMass Trace Repository (2009) http://traces.cs.umass.edu/index.php/Storage.
[10] D. Narayanan, *et al.*: "Write off-loading: Practical power management for enterprise storage," TOS **4** (2008) 10 (DOI: 10.1145/1416944.1416949).

1 Introduction

An increasing number of storage systems are equipped with state-of-the-art solid state drives (SSDs) as the secondary storage device owing to advanced technologies in non-volatile memory. As the capacity continues to scale with Moore's law, low-density-parity-check (LDPC) induced SSDs have attracted tremendous research attention since they offer a stronger error correction code (ECC) than conventional BCH code with improved reliability of the flash-based systems [1, 2]. Moreover, a big trend of SSD in the next generation is the open-framework SSD where hosts permitted to directly manage the raw flash memory can organize data to maximally exploit the performance characteristics of flash memory SSD.

However, exploiting the potential merits of LDPC-induced SSDs which can efficiently collaborate with master-slave databases is not well studied. With different soft-decision sensing levels in LDPC-induced SSDs, the appropriateness of latency model has to be reconsidered depending on the access characteristics of master-slave databases. Furthermore, we observed the descending of the reliability in the master site because the raw bit error rate (RBER) will gradually approach the error correction capability of soft-decision LDPC. This observation motivates us to investigate possible solutions to enhance reliability, and thus we propose the write speed adjustment scheme. While all existing database optimizations, such as database sharding, are completely orthogonal to our work, they can be used concurrently with our proposed approach to optimize flash-based master-slave database system performance.

In this paper, we demonstrate that appropriately using an optimized cross-layer latency approach can decrease LDPC-induced system response time. We propose the latency trade-off approach called MFW-SFR (Master-Fast-Write and Slave-Fast-Read) to speed up insert/update/delete operations and maximize the effective bandwidth in outstanding select requests. Furthermore, we design the write speed adjustment scheme in master sites to extend retention time. With our proposed approach, the flash-based SSDs can be effectively utilized in master-slave databases with high efficiency and reliability, achieving a huge gain in performance.

2 The proposed MFW-SFR approach

LDPC-induced flash memory SSDs exhibit a fundamental trade-off between flash memory program speed and data retention time. In LDPC-induced systems, program operations are always in compliance with the incremental step pulse programming (ISPP) which gradually accumulates voltage to transcend the threshold voltage (V_{th}) with the increment voltage size (ΔV_p) [3]. The relationship

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016

between program speed (V_{prog}) and increment voltage size (ΔV_p) can be represented as follows:

$$V_{prog} = \kappa * \frac{\Delta V_p}{V_{th} - V_{start} + \delta} \tag{1}$$

where κ is a constant factor, and δ is the permissible fluctuant voltage. The dependence of δ on different starting voltage (V_{start}) can be represented as follows:

$$\delta = V_{start} + \left\lceil \frac{V_{th} - V_{start}}{\Delta V_p} \right\rceil * \Delta V_p - V_{th} \tag{2}$$

In that way, the program speed V_{prog} is directly proportional to the increment voltage size ΔV_p. Therefore, with a larger ΔV_p in each step of ISPP, fewer hop counts are required and program speed gets correspondingly faster. But in the meantime, faster program results in a predictable decline in the sensing precision and an increase in the RBER of LDPC code decoding [4, 5, 6]. To harmonize well with the increased RBER, more extra sensing levels should be applied to the LDPC code to enhance the strength of error correction and consequently the sensing latency linearly enlarges with increased numbers of sensing quantization levels [7]. Besides, more fine-grained sensing levels lead to more data streams transferring and thereby increasing the data transfer latency, which corresponds to a slower read speed. Hence, data with faster writing speed are read slower and vice-versa.

2.1 Cross-layer latency trade-off approach

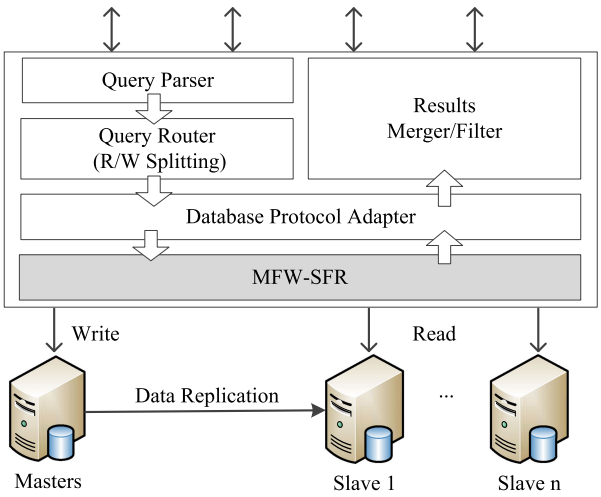

Fig. 1. Structure of SSD-based master-slave database.

Based on above observations, an appropriate cross-layer latency trade-off approach called MFW-SFR is proposed to improve the I/O efficiency of LDPC-induced master-slave databases, as illustrated in Fig. 1. Since data are processed with the read/write splitting in typical master-slave databases, masters receive write operations whereas slaves provide read-only service. In our MFW-SFR approach, masters always boost the process of writing data without taking into account the read latency. To achieve fast read in slave sites, a straightforward approach is synchronizing modified records with slower write speed when data synchronization

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016

occurs, then reading these data in the near future will be faster. Nevertheless, this slow synchronization may cause that more requests get stuck in the I/O queue once too many I/O requests inflow during periods of high workload intensity. Thus, during a high-intensity period, MFW-SFR synchronizes data for slaves with a faster program speed and flags them to re-write with slower speed in the future idle time, otherwise slaves only need to program slowly for the fast read. The combination of these policies improves the performance of flash-based master-slave database system significantly.

2.2 Write speed adjustment scheme

As consumed flash memory P/E cycles accumulate, the tunnel oxide of floating gate transistors is gradually damaged in the form of charge trapping in the oxide and interface states, leading to the increased raw bit error rates and limited lifetime of flash-based SSD. In MFW-SFR databases, the master flash devices will become unmanageable earlier than slaves because fast programming increases the RBER [8], and should be replaced, transitioning to new ones. In this study, a write speed adjustment scheme is proposed to extend the lifetime of master flash devices by dynamically maximizing the write speed without exceeding the maximum error correction capability of LDPC.

In our write speed adjustment scheme, a relative larger write speed is used in the initial state of SSDs and the number of faulty bits within the data stored in each page is periodically read and checked. Once the RBER of some pages is approaching the error correction capability of LDPC with maximal sensing levels, the write speed is decreased by reducing ΔV_p. The overhead of write speed adjustment is related to the length of read-and-check period. By setting a relative larger step of decreasing ΔV_p, the RBER is reduced significantly when write speed adjustment happens, which extends the read-and-check period and thus reduces the overhead. Note that the write speed adjustment scheme is only designed for master flash devices for the reason that data in slave flash devices is written slowly with minimal ΔV_p for fast read, which indicates that the RBER of each page is far below the error correction capability of LDPC.

3 Performance evaluation

We performed experiments on a trace-driven simulator to verify the effectiveness of the proposed approach. The simulated 1 TB enterprise-level SSD is configured with 8 channels, and each channel is equipped with 8 chips. Read-intensive workloads are used in the performance evaluation [9, 10], and the trade-off latency speed setting in program and read of this LDPC-induced SSD is consistent with previous work [8]. The master-slave database consists of one master and three slaves, which is a representative of the master-slave model. Predicatively, SSDs that substitute for mechanical disks will tremendously improve the overall system performance in master-slave databases. Therefore, we treat SSD-based master-slave database without any further targeted optimization as the baseline case in our contrastive experiments.

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016

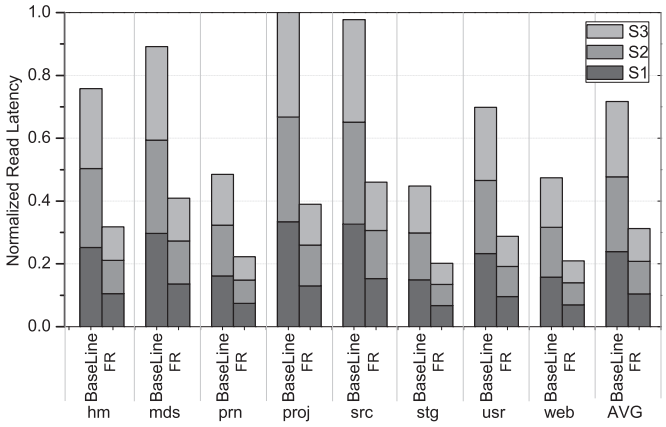

Fig. 2. Normalized read latency of slave database.

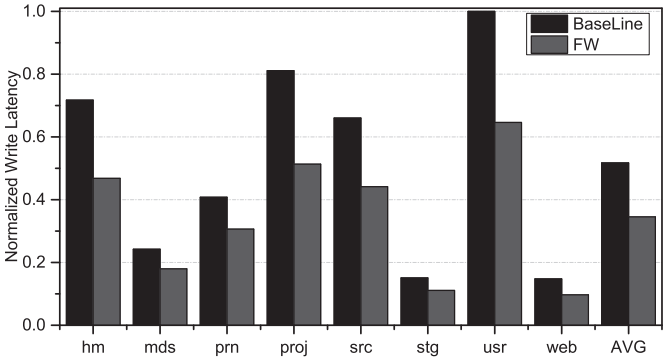

Fig. 3. Normalized write latency of master database.

Fig. 2 shows the normalized read latency of slave database, and Fig. 3 shows the normalized write latency in the LPDC-induced master database. There are three slaves namely S1, S2 and S3, respectively. Compared with the baseline, FR (Fast Read) in slave sites, as shown in Fig. 2, achieves crucial read performance improvement in all traces with a highest 61.07% reduction of read latency time, and an average of 57.11% I/O latency reduction for read requests. Intuitively, it demonstrates that FR contributes to a dramatically I/O latency reduction in slave sites. This is because only query services are offered in slaves, and users have no permission to modify data information, therefore slow write leads to fast read. Moreover, the overhead of re-write is negligible since on-demand re-programming is completed in the idle time. For write latency in master, FW (Fast Write) achieves an average of 33.23% I/O latency reduction for write requests, as shown in Fig. 3. Since there are only write commands in master, fast-write-and-slow-read markedly reduces I/O latency time. But why is there not a critical decrease in master compared to slaves? This is because in master there are garbage collection and advanced commands like copyback involving the elapsed time from target physical pages to data registers and the consume time transferring data by data bus, notwithstanding without read operations in master. Nevertheless, FW performs consistently better overall performance than baseline in all traces. With the combination of FW and FR in LDPC-induced master-slave databases, we can find that the proposed MFW-SFR is very efficient in the reduction of I/O latency.

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016

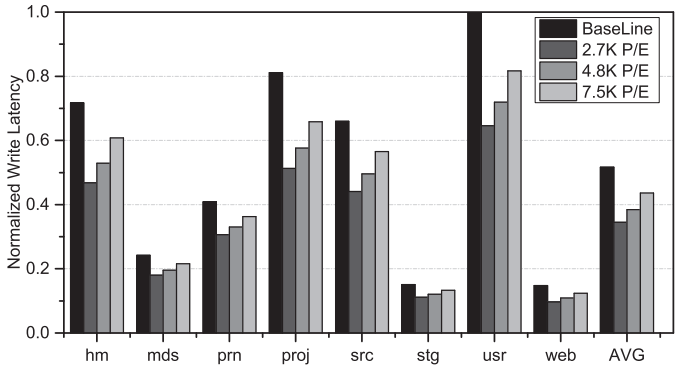

Fig. 4. The normalized write latencies under different wear-out stages of SSDs

To measure the sensitivity of the proposed MFW-SFR to the wear-out stages of SSDs, we further carried out Monte Carlo computer simulations to obtain the cell threshold voltage distribution within different stages, and three different wear-out stages corresponding to 7.5K, 4.8K and 2.7K P/E cycles are evaluated. We use 675 µs as the 2 bit/cell NAND flash memory program latency when ΔV_p is 0.3. Fig. 4 gives a comparison of the normalized average write latency compared to baseline under three different wear-out stages. It can be observed that MFW-SFR decreases write latency by an average of 33.23%, 25.67%, 15.76% under the P/E cycling of 2.7K, 4.8K and 7.5K respectively, which means MFW-SFR always outperforms baseline under different wear-out stages. Furthermore, with the increased number of P/E cycling, the write performance improvement brought by MFW-SFR gets smaller for the reason that the proposed write speed adjustment scheme reduces write speed of master flash devices for lower RBER and longer retention time.

With the combination of MFW-SFR and the write speed adjustment scheme, the performance of master-slave databases launched on SSDs can be significantly improved without reliability degradation. We believe that this approach will be useful to system designers and administrators for further optimization of the flash utilization in master-slave databases.

4 Conclusion

In this paper, we propose a cross-layer latency trade-off approach called MFW-SFR to efficiently reduce the I/O response time of LDPC-induced master-slave databases. Moreover, the write speed adjustment scheme in master site is designed to enhance the reliability of the database system. Therefore, LDPC-induced SSDs selected as the secondary storage devices in master-slave databases will tremendously improve the overall system performance. Experimental results show that the proposed approach achieves a 57.11% reduction of average slave response time, and a 33.23% reduction of average write latency in master site.

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016

Acknowledgments

The authors would like to thank anonymous reviewers for their numerous suggestions. This work was supported in part by the National Natural Science Foundation of China under grant NO. 91330117, the National High-tech R&D Program of China (863 Program) under Grant No. 2014AA01A302, the Shenzhen Scientific Plan under Grant NO. JCYJ20120615101127404, No. JCYJ2013 0401095947230 and No. JSGG20140519141854753.

DOI: 10.1587/elex.13.20160468
Received May 10, 2016
Accepted June 1, 2016
Publicized June 17, 2016
Copyedited July 10, 2016