Exploiting Data Longevity for Enhancing the Lifetime of Flash-based Storage Class Memory

Wonil Choi †

Mohammad Arjomand †

Myoungsoo Jung ‡

Mahmut T. Kandemir †

† Pennsylvania State University

‡ Yonsei University

ABSTRACT

This paper proposes to exploit the capability of retention time relaxation in flash memories for improving the lifetime of an SLC-based SSD. The main idea is that as a majority of I/O data in a typical workload do not need a retention time larger than a few days, we can have multiple partial program states in a cell and use every two states to store one-bit data at each time. Thus, we can store multiple bits in a cell (one bit at each time) without erasing it after each write – that would directly translates into lifetime enhancement. The proposed scheme is called Dense-SLC (D-SLC) flash design which improves SSD lifetime by $5.1\times$ – $8.6\times$.

ACM Reference format:

Wonil Choi †, Mohammad Arjomand †, Myoungsoo Jung ‡, Mahmut T. Kandemir †. 2017. Exploiting Data Longevity for Enhancing the Lifetime of Flash-based Storage Class Memory. In *Proceedings of SIGMETRICS '17, Urbana-Champaign, IL, USA, June 05-09, 2017*, 1 pages. <https://doi.org/http://dx.doi.org/10.1145/3078505.3078527>

1 DENSE-SLC NAND FLASH-BASED SSD

A typical flash-based Storage Class Memory (SCM) has a hierarchal internal structure: there is a fast Single-Level Cell (SLC) Solid State Drive (SSD) with tens of gigabytes capacity at the upper level and a slow Multi-Level Cell (MLC) SSD with terabyte capacity at the lower level. In this setup, the SLC SSD services a great portion of the incoming traffic which poses high write pressure on it – this makes write endurance a significant challenge for the SLC SSD part (each flash cell can tolerate 10^4 – 10^5 program/erase cycles). This paper targets the lifetime problem of SLC SSD and discusses the opportunity for improving it by relaxing its *retention time*.

Motivation – The flash devices are traditionally expected to retain data for one or more years. However, the stored data in an SSD SCM does *not* require this long-term non-volatility. Within a hierarchical SCM, we expect the SLC SSD to handle the I/O requests with short-term longevity, while the I/O requests with long-term longevity are normally handled by the underlying MLC SSD. As a motivation example, Figure 1 (left) shows the CDF of data longevity for I/O data in mds_0. We can see that more than 90% of written data in mds_0 have a longevity of up to 10 hours. Similar behaviors were observed for a wide range of enterprise workloads.

Proposal – The main idea is that, by relaxing the retention time of an SLC device, we can have more than two states in a cell. At each given time, similar to the conventional SLC, we use every

Figure 1: Data longevity and PWE analysis for mds_0.

two states to write one bit information. In this way, a device stores multiple bits (one bit at each time) before it needs an erase, thereby increasing the number writes to cell during one erase cycle, or increasing the maximum number of *logical pages* stored in *one physical page* during *one P/E cycle* (we call this “page writes per erase cycle” or PWE, i.e., always one in the conventional SLC flash). Clearly, increasing PWE leads to device lifetime improvement.

We propose Dense-SLC (D-SLC) flash where each flash block has one of the three modes: all pages/cells in it are in 2-state mode (same as the conventional SLC), 4-state mode (it has two intermediate states; its PWE is three), or 8-state mode (it has 6 intermediate states; its PWE is 7). An 8-state mode block has the shortest retention time and suitable for storing data with short longevity (e.g., less than an hour), whereas a 2-state mode block has the longest retention time and suitable for data with long longevity (e.g., greater than 3 days). The 4-state mode has a moderate retention time and is used for data with “10 hours to 3 days longevity”.

Results – We used DiskSim with the SSD extensions by Microsoft to model an SLC SSD for our evaluation. Figure 1 (right) shows the percentages of blocks with 2, 4, and 8 states for mds_0 during the entire device lifetime – this shows that our proposed flash can store more I/O data by storing them into 4 and 8-state mode blocks (compared to the conventional SLC with 2-state mode blocks only). For the 15 evaluated workloads from MSR Cambridge suite, our D-SLC design improves the device lifetime by $5.1\times$ – $8.6\times$.

D-SLC in detail – D-SLC implementation requires hardware/software supports, including calibrating the parameters of the flash programming algorithm and modifying block selection and garbage collection algorithms at flash software (FTL). Detailed description and analysis of the D-SLC design are available at [1].

2 ACKNOWLEDGMENT

This work is supported in part by NSF grants 1302557, 1213052, 1439021, 1302225, 1629129, 1526750, and 1629915, a grant from Intel, NRF 2016R1C1B2015312 and 2015M3C4A7065645, and MSIP IITP-2015-R0346-15-1008.

REFERENCES

- [1] Wonil Choi, Mohammad Arjomand, Myoungsoo Jung, and Mahmut T. Kandemir. 2017. Exploiting Data Longevity for Enhancing the Lifetime of Flash-based Storage Class Memory. In *SIGMETRICS*. <https://arxiv.org/abs/1704.05138>.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGMETRICS '17, June 05-09, 2017, Urbana-Champaign, IL, USA

© 2017 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5032-7/17/06.

<https://doi.org/http://dx.doi.org/10.1145/3078505.3078527>