Variability of NVM Response Time and its Effect on the Performance of Consumer SSDs

Maria Varsamou and Theodore Antonakopoulos
University of Patras
Department of Electrical and Computer Engineering
Patras 26500, Greece
e-mails: *mtvars@upatras.gr* and *antonako@upatras.gr*

***Abstract*—Solid-State Drives (SSDs) use non-volatile memories (NVM) for storing and retrieving information in the form of sectors and/or pages. For achieving high capacity, consumer SSDs use high density multi-level cells (MLC) memories that experience high read and write times. The maximum achieved I/O performance and the minimum response time depends on the used NVM technology, which determines the read and write times, and other system parameters, like the number of simultaneously accessed NVM channels, the SSD controller architecture, its functionality, the supported commands and the applied workload. Most of these parameters remain unchanged during the lifetime of an SSD, except for the read and write times which vary as the lifetime of the device progresses and higher variability is observed. By defining the basic equations of the maximum SSD performance and using experimental results, we determine how the increase of the NVM response time affects the performance of a consumer SSD and under what conditions this is observed by the SSD's user.**

I. Introduction

Non-volatile memories (NVM), like NAND flash and phase-change media (PCM), are used increasingly in today's systems, either for storing or caching purposes [1]. A Solid-State Drive (SSD) is the most typical digital system where these memories are used. SSDs are connected to a central processor/host either as an external storage device (using interfaces like SCSI, SATA/SAS etc.) or directly to its internal I/O architecture (i.e. using a PCIe slot attached to the motherboard's root complex). The NV memories are used in sets of chips, either by sharing a common bus or/and in parallel configurations, in order to achieve high I/O data rates [2]. Various types of interfaces are used in these memories chips. The most well known interfaces are ONFI, Toggle DDR and LPDDR2-NVM [3]. In all configurations, there is a storage controller where these chips are directly attached using a number of independent NVM channels. This controller includes also the proper I/O interface for exchanging information with the central processor, usually at Gbps rates. Advances in NV memory technologies, in terms of storage density, internal architecture, write and overwrite mechanisms, signal sensing, power dissipation etc., affect the overall system performance ([4], [5] and [6]).

In order to lower the SSDs cost (in terms of $/GB), the geometries of NV memories shrink, but this affects their reliability and the overall NVM endurance drops dramatically. For example, the endurance of previous generations SLC (Single Level Cell) NAND flash memories was at least 100K P/E cycles, while the current 2x/3x nm MLC (Multi Level Cell) NAND flash memories have less than 5K P/E cycles. Likewise, the response time has also been affected. The page program/write time has been increased from 250 usecs to 1300 usecs. At the same time, the need for faster and more reliable SSDs becomes mandatory in today's competing market and intelligent block management, garbage collection, wear leveling and other algorithms are employed ([7], [8] and [9]).

Section II discusses the basic NVM technologies used in current and near-future SSDs and how their response time is affected by aging. Section III analyzes how the maximum possible performance of a NVM channel and the whole SSD depends on the NVM characteristics. Using experimental results and mean value analysis, Section IV discusses their conformity and possible factors that explain any observed differences. Finally, Section V presents the basic findings of this work.

II. Non-Volatile Memories in SSDs

In order to estimate the performance of an SSD, we have to analyze the behavior of its basic components, i.e. the used NVM technology, the NVM channel, the internal architecture of the storage controller along with its I/O interfaces and the limitations imposed by the upper layer software, i.e. device driver. As NVM channel we consider the set of NVM chips that share the same bus for communicating with the storage controller. In this section we discuss the basic NVM technologies and how their response time is affected by aging.

Today the most well known NVM technology is NAND flash, used in almost all commercial SSDs, while PCM is a new technology that demonstrates DRAM-like read performance, comparable to the NAND flash write performance and much higher endurance, but still much lower storage density. PCM is a promising technology that will be shortly introduced in the commercial SSD market and we estimate that it will become a peer competitor to NAND flash. Although the various NV technologies have different characteristics in terms of minimum and maximum data block size, rewritability, need for erasing before write, endurance, aging and raw bit error rate, they share the same functionality at their interface with the SSD's storage controller. For the

978-1-4799-1412-8/13/$31.00 ©2013 IEEE

two basic operations that exist in all memory types, read and write/program, all NV memories follow a set of three discrete phases. The common NV bus (NVM channel) is not used in all phases, and that allows the system designer to exploit the pipelining concept. In this analysis we do not consider technology-dependent functions (i.e. erasing of NAND flash) and their effect on system performance.

A. Basic NVM Commands

Writing data is performed in three steps: initially the write command is issued and determines the base address at the memory's linear space where the data have to be stored, then the data are transferred to an internal data buffer in the NV memory chip and, at the final phase, the data are stored to the NVM cells. As long as the last step is in progress, new data cannot be applied to the NVM cells, although some NV chips use a second data buffer for applying pipelining at the chip level. The chip status can be sensed by reading an internal status register or by monitoring a separate ready/busy signal.

Similar functionality is experienced when a read command is executed. Initially the command is applied and the base address, where the data are stored, is specified. The memory chip retrieves the data from the NVM cells and the data are transferred to an internal data register. During that phase the NVM channel is not utilized and it can be used for executing a command into another NVM chip that is attached on the same channel. At the third phase of the read command, the data are transferred to the system's memory controller. For NAND flash memories, the read time is comparable with the page transfer time, while the program and erase times are a few hundreds usecs (much larger than the page transfer time), and in MLC NAND flash the block erase time may last for a few msecs. In most advanced NV memories like PCM, the second phase during read is negligible, just a few clock cycles (a few nsecs), while the write time is more than a hundred usecs and still much higher than the data transfer time. Table I shows some values for the above mentioned parameters for current and near-future technologies. Storing 4K pages in a PCM chip requires multiple accesses to the same chip, or distribution of the data of a page to multiple NVM dies or channels.

The duration of the data transfer phase is determined by the interface characteristics (i.e. clock frequency, bus width, double data rate operation) and the data transfer size. On the contrary, the duration of the NVM cells accessing phase for reading or writing data is determined only by the NVM technology (single or multiple levels cells, programming/erasing mechanism, number of read/write heads, etc.). Although in the first generation of NVM interfaces the data transfer time was equal or higher to the read time, and a few times smaller than the data write time, in the latest NVM interfaces where data rates of a few hundreds of MBps are supported, the data transfer time is always smaller than the read time and at least an order of magnitude smaller than the write time. These time differences are exploited in various SSDs in order to improve their I/O performance.

TABLE I
BASIC CHARACTERISTICS OF NVM TECHNOLOGIES

Parameter	SLC Flash	cMLC Flash	PCM #1	PCM #2
Page Read [usecs]	25	50	0	0
Page Write [usecs]	250	1300	120	120
Data Size [bytes]	4K	8K	64	1024
Data Rate [MBps]	40	200	33	200
Transfer Time [usecs]	100	40	2	5
Endurance [P/E cycles]	100k	5k	1M	1M

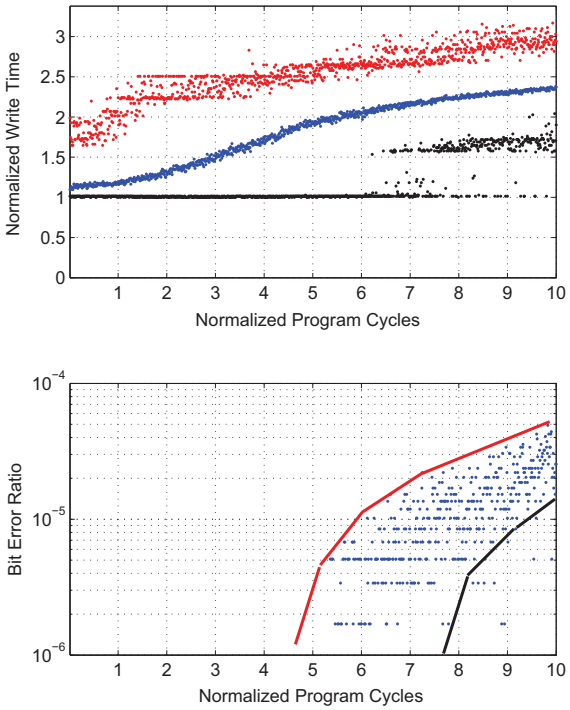

Fig. 1. Variability of NVM write time and bit error ratio (BER) as a function of the normalized write cycles.

When an SSD is under design, probably the most important factor that is taken into account is the used NVM technology, since a few system parameters depend on the behavior of the underlying technology. For example, when pipelining is used, the optimum pipeline depth mainly depends on the ratio of the write time to the data transfer time. When the write time increases (in some cases by a factor of 2 or 3) the device operates under non optimum conditions and its performance may degrade substantially.

B. NVM Aging

As the content of the NV memories is updated, aging effects are observed [10]. The high voltages applied for programming a page and the much higher voltages applied for erasing a block (in NAND flash) degrade the memory's raw storage reliability and also increase the program time. Fig. 1 shows experimental results which demonstrate the effect of program cycles on the response time and reliability of a NV memory. The program cycles have been normalized to the

manufacturer's specified endurance (in terms of Program/Eras cycles for NAND flash or Set/Reset cycles for PCM) and th write time has been normalized to the typical write time of virgin device. Fig. 1(a) shows how the write time is affecte by the aging effect. The blue line shows the mean normalize write time, while the black and the red lines are the minimu and maximum values respectively. It can be observed that a parameters are affected by the aging of the device, i.e. the increase as the memory wears out. Additionally, as Fig. shows, the distribution of the write times is also affected b the aging effect, by demonstrating a more spread distributio along with the shifting of its mean value.

Fig. 1(b) shows the effect of aging on the memory reliability. The increased error rate as the time progresse necessitates the use of error correcting codes (ECC for achieving a given user reliability level. Howeve this introduces additional latency, along with redunda information. The use of ECC extends the lifetime of an SS on the expense of increased hardware complexity and lowe read performance under aging conditions, since recovering c corrupted pages requires computational effort and probabl a number of decoding iterations, depending on the use ECC. There are also other NVM related functions that affe the performance of an SSD, like garbage collection and wear-leveling using over-provisioning. For example, when NAND flash is used in an SSD, wear-leveling has to be performed for freeing up invalid pages, and that decreases the total SSD performance. Since this operation is dependent on the NAND flash program and erase times, it is also affected by the aging effect. Analyzing these functions is out of the scope of this work.

III. The NVM Channel and the SSD Controller

For the rest of this analysis, we use the following terminology: L is the basic data structure (in bytes), R is the data transfer rate (in MBps) and T_W and T_R are the page program and read times (in usecs) respectively. In a single NVM chip, the maximum write and read rates (in pages per sec) are given by $\frac{LR}{L+RT_W}$ and $\frac{LR}{L+RT_R}$ respectively. Using the values of Table I we can easily conclude that due to the large program times, the write performance is much lower than the maximum transfer rate supported by the NVM interface. One way to increase this performance is to form a NVM channel with multiple NVM memories and to use the pipelining approach. The maximum pipeline depth is given by $maxP = \lfloor \frac{T_W R}{L} \rfloor + 1$.

In a real SSD, the actual pipeline depth depends also on other system parameters, like signal strength of the chips used and the clock frequency at the NVM channel. Due to parasitic capacitances, increasing the clock frequency results in decreased maximum pipeline depth for the same I/O technology. It has to be mentioned that usually the actual pipeline depth is much lower than $maxP$.

If N is the number of NVM chips used in the NVM channel and P is the used pipeline depth, then the maximum I/O rate that can be achieved is given by:

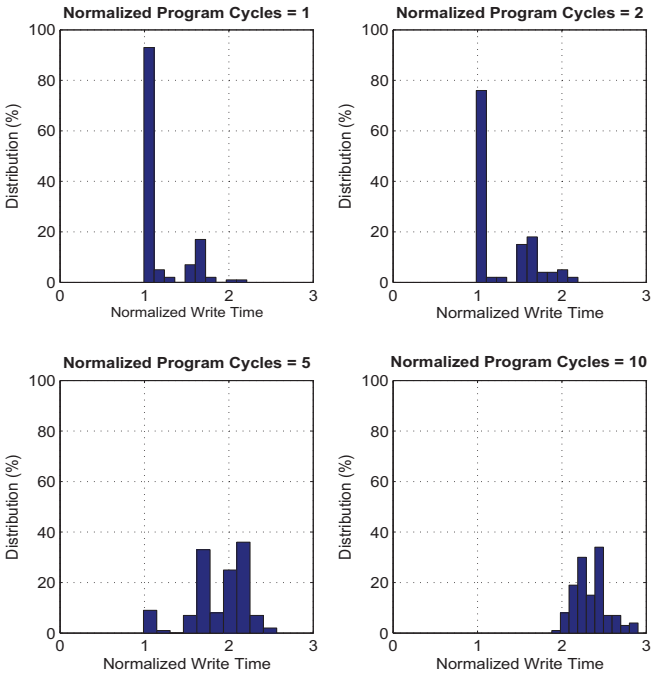

Fig. 2. Normalized write time distributions at various instances of the lifetime of a Solid-State Drive.

$$\begin{array}{lll} R_{P_R} & = min(R, \frac{PLR}{L+RT_R}) & \text{for read, and} \\ R_{P_W} & = \frac{PLR}{L+RT_W} & \text{for write} \end{array} \quad (1)$$

In the above equations, we assumed optimum loading conditions and that $maxP > P$, since $T_W >> \frac{L}{R}$ in most NVMs for commercial SSDs. From the above equations it is concluded that any increase on T_W decreases R_{P_W}, and thus affects the NVM channel's program rate. Measurements of T_R on various NV memories show no significant variations and the NVM channel's read rate remains unchanged, but at the system-level the increase on BER (as shown in Fig. 1) either drops the read performance due to the used ECC or shortens the SSD's lifetime. It has to be mentioned, that since in this analysis we are targeting the upper limit of the performance of a commercial SSD, we do not take into account various parameters that affect the final system performance, i.e. the block erase time for NAND flash and the wear-leveling and garbage collection algorithms.

Figure 3 shows the effect of the increase of program time on the performance of a NVM channel that uses pipelining. In this figure we have used realistic pipeline depths that are met in commercial SSDs and the actual read/program times of existing NVMs, as indicated in Table I. The curves of the various pipeline depths are marked with different colors and in all subfigures the achieved I/O rate has been normalized to the I/O rate achieved when no pipelining is used, and the normalization values represent different actual I/O rates. For example, for the SLC NAND flash the normalization factor

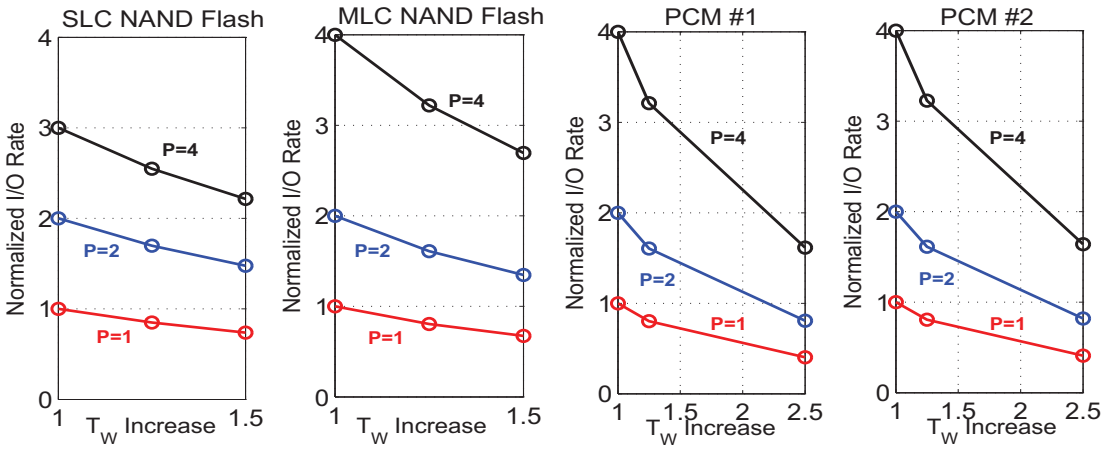

Fig. 3. The effect of program time increase on the performance of the NVM channel.

corresponds to 3 kIOPs, while for the MLC NAND flash it corresponds to 740 IOPs.

The first remark is that the use of high speed interfaces allows the use of high pipeline depth but the bus utilization decreases due to the small value of the ratio $\frac{L}{L+RT_W}$. In the SLC case with a moderate interface, the utilization starts at 33% and goes up to 99% (13 and 39 MBps respectively), while in the MLC case with a high transfer rate interface, the channel utilization starts at 3% and goes only up to 12% (6 and 24 MBps respectively), and this is due to the high write time. Therefore it is questionable whether the current trend to increase the transfer rate as the NVM density increases (and the program time increases) is the best approach at the system level. This comment is not valid during read, where the read time is comparable to the data transfer time, and in this case full utilization can be achieved with a small pipeline depth. The second remark is that the variability of the program time affects the achieved I/O rate substantially and may reduce significantly the gain achieved by pipelining as the NVM aging progresses.

In order to increase the total I/O rate of an SSD, multiple NVM channels are used that operate in parallel. The use of wear-leveling algorithms results in uniform aging on all NVM channels and since the SSD I/O performance is a linear function of the number of NVM channels, the effect of aging discussed previously is also valid at the SSD level.

During the design of an SSD, two additional parameters have to be considered, the pin budget and the lifetime of the SSD. Both parameters are related with the maximum number of NVM channels for a given pipeline depth and the NVM aging. Pin budget is the maximum number of pins that are available for a given storage controller package. If P_B is the maximum number of available I/O pins, the following equation holds: $P_B \geq N_c(M_c + 2P)$, where M_c is the number of pins required for interfacing a single NVM chip and N_c is the number of parallel NVM channels. Increasing the pipeline depth increases also the number of pins required per NVM channel and that may result in decreased number of NVM channels. Therefore, since pipelining affects differently the read and write I/O transfers, the SSD design has to be based on the optimal combination of pipeline depth and number of parallel NVM channels for a given pin budget.

The lifetime of an SSD is defined by $\frac{S_S E}{W_t A}$, where S_S is the SSD's storage space being written, E is the NVM endurance, W_t is the user program rate and A is the write amplification factor determined by the endurance/retention related algorithms used in the SSD [11]. It should be noted that write amplification is only present in flash memories, since they must be erased before they can be rewritten, resulting in moving (or rewriting) user data and meta-data more than once. However, the same equation for the lifetime stands also for PCM memories, but in this case $A \approx 1$, since wear leveling techniques used in PCM, such as Start-Gap [12], induce just a small number of additional write operations to the user requests. In an SSD with multiple NVM channels, the SSD space (referred also as total capacity) is given by $S_S = N_c max(P, N) C_V$, where C_V is the capacity of a single chip/die. It has to be mentioned that N is always equal or greater than P. The maximum written volume per time unit (program rate) is given by $W_t = N_c R_{P_W}$ and depends on the NVM response time. As the aging progresses, the SSD becomes slower and that extends slightly its lifetime, since the user program rate decreases.

IV. Experimental Analysis of a Consumer SSD

For validating the above analysis, we analyzed a commercial PCIe-based SSD. The SSD consists of a single chip controller with 8 NAND flash parallel channels, has 256 GB capacity, uses 16 MLC NAND Flash chips, each chip has 2 planes and 8K pages are used. Therefore up to 4 commands can be executed in parallel by sharing the same data bus. The maximum data rate supported by the chips is 166 MBps and the typical read and write times are 50 usecs and 1300 usecs respectively. Since the 8 KB page transfer time is 49 usecs, almost equal to the page read time, and a few times shorter than the page program time, the maximum achievable transfer time is 40 kIOPs for read and 5.9 kIOPs for write per channel

(each IOP corresponds to 4 KB). Under optimum loading conditions the maximum achievable rate at the SSD level should be 320 kIOPs for read and 47 kIOPs for write.

The experimental results show that the maximum achievable rate is much lower than the expected one in all loading conditions, and the same holds also for other consumer SSDs. Since the used device driver and the I/O testing tool do not introduce any limitations on the experimental methodology (using the same software stack rates of more than 100 kIOPs have been achieved in a custom PCIe card with a few GBs of DRAM), we conclude that the main limitation is due to the used storage controller, and any introduced aging does not result to SSD performance degradation, since the device does not operate near to its maximum performance and any NVM chips performance degradation due to aging is not observable.

V. Conclusions

Theoretically, the I/O performance of SSDs is determined by the used NVM technology. The response time of NVM chips is affected by aging and in consumer SSDs performance degradation might be expected as the lifetime of the device progresses, especially when we exceed the manufacturer specified endurance. Measuring the I/O performance of consumer SSDs under heavy loading conditions, we did not observe any measurable performance degradation. This is due to the used storage controllers which cannot support the maximum data rate determined by the NVM technology.

In SSDs that fully exploit the capabilities of the underlying NVM technology, the aging should have strong impact on the I/O performance experienced by the user. When the SSD design is suboptimum, the variability of the characteristics of the NVM technology may have slight or negligible effect, since the dominant performance factor is on the storage controller and independent of the used NVM technology.

References

[1] J. Brewer and M. Gill [Eds.], "Nonvolatile memory technologies with emphasis on flash: A comprehensive guide to understanding and using flash memory devices," *Wiley-IEEE Press*, 2008.

[2] Feng Chen, Rubao Lee and Xiaodong Zhang, "Essential roles of exploiting internal parallelism of flash memory based solid state drives in high-speed data processing," in *The 17th IEEE International Symposium on High Performance Computer Architecture (HPCA)*, San Antonio, Texas, USA, February 12-16 2011.

[3] "Open NAND Flash Interface Specification, Revision 2.0," ONFI Workgroup, February 2008.

[4] Yang Hu, Hong Jiang, Dan Feng, Lei Tian, Hao Luo and Shuping Zhang, "Performance impact and interplay of ssd parallelism through advanced commands, allocation strategy and data granularity," in *The 25th International Conference on Supercomputing - ICS11*, Tucson, Arizona, USA, May 31June 4 2011.

[5] Jaehong Kim, Sangwon Seo, Dawoon Jung, Jin-Soo Kim and Jaehyuk Huh, "Parameter-aware i/o management for solid state disks," *IEEE Transactions on Computers*, vol. 61, May 2012.

[6] H. Howie Huang, Shan Li, Alex Szalay and Andreas Terzis, "Performance modeling and analysis of flash-based storage devices," in *The 27th IEEE Symposium on Massive Storage Systems and Technologies - MSST 2011*, Denver, Colorado, USA, May 23-27 2011.

[7] Nitin Agrawal, Vijayan Prabhakaran and Ted Wobber, "Design tradeoffs for ssd performance," in *USENIX Technical Conference - USENIX08*, Boston, MA, USA, June 22-27 2008.

[8] S. Li and H. H. Huang, "Black-box performance modeling for solid-state drives," in *The 18th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems - MASCOTS 2010*, Miami Beach, Florida, USA, August 17-19 2010.

[9] Myoungsoo Jung and Mahmut Kandemir, "Revisiting widely held ssd expectations and rethinking system-level implications," in *ACM SIGMETRICS13*, Pittsburg, PA, USA, June 1721 2013.

[10] Yangyang Pan, Guiqiang Dong and Tong Zhang, "Exploiting memory device wear-out dynamics to improve nand flash memory system performance," in *The 9th USENIX Conference on File and Storage Technologies - FAST'11*, San Jose, California, USA, 2011.

[11] Xiao-yu Hu, Evangelos Eleftheriou, Robert Haas, Ilias Iliadis and Roman Pletka, "Write amplification analysis in flash-based solid state drives," in *The Israeli Experimental Systems Conference - SYSTOR 2009*, Haifa, Israel, May 4-6 2009.

[12] Moinuddin K. Qureshi, John P. Karidis, Michele Franceschini, Vijayalakshmi Srinivasan, Luis Lastras and Bulent Abali, "Enhancing lifetime and security of PCM-based main memory with start-gap wear leveling," in *42th Annual IEEE/ACM International Symposium on Microarchitecture - MICRO 2009*, New York, NY, USA, December 12-16 2009.