

# Influence of Number of Channels and FTL on SSD Power Consumption

Seokhei Cho, Youjip Won, Sooyong Kang, Jaehyuk Cha, Jongmoo Choi, Sungroh Yoon

**Abstract**— Power consumption is increasing as the number of NAND flash memories is increasing for improvement of SSD performance. SSD power consumption also receives influence of firmware. This paper analyzes the influence of channel among SSD hardware factors and FTL among firmware on SSD power consumption. We have found that FTL has more influence on SSD power consumption among the two factors.

**Keywords**—SSD, FTL, Power Consumption, Channel, NAND Flash

## I. INTRODUCTION

Number of NAND flash connected to the inside must be increased to improve SSD performance and capacity. However when number of NAND flash increases, power consumption also increases. For Z-DRIVE R4 that was currently released in OCZ has 64 to 128 NAND flashes in which a maximum of 128 NAND flashes can be operated at the same time [1]. This product uses PCIe interface and has increased parallel operation by connecting 8 SSD controllers by RAID. Bandwidth of R4 could be improved to 2.8GB/s through this, but power consumption had increased to 23 ~ 26W [1]. SATA3-use SSD, Vertex4 has 560MB/s bandwidth and 2.5W power consumption. R4 shows about 5 times higher Bandwidth compared to Vertex4 but has about 10 times higher power consumption.

SSD power consumption should be analyzed by synthesizing FTL and hardware composition. SSD hardware composition is developing in direction of increased power consumption. Number of NAND flash, number of channel/way, page size, and calculation speed is increasing. However, power consumption of SSD varies even though it has same hardware depending on FTL efficiency. For Intel X25-M, power consumption consistently increases depending on IO size. However for MXP, power consumption increased by increase of very 32KB of IO size. Also, waiting power consumption of 2 SSD showed about 2 times difference [3]. Like this, SSD power consumption can be variously shown depending on FTL of SSD that analysis should be conducted by synthesizing FTL

Seokhei Cho, Youjip Won, Sooyong Kang and Jaehyuk Cha are with Hanyang University, Seoul, Korea. (e-mail: {misost|yjwon|sykang|chajh}@hanyang.ac.kr).

Jongmoo Choi is with Dankook University, Yongin-si, Korea. (e-mail: choijm@dankook.ac.kr)

Sungroh Yoon is with Seoul National University, Seoul, Korea. (e-mail: sryoon@snu.ac.kr)

and hardware.

## II. RELATED STUDIES

Studie on SSD power consumption were mainly based on practical measurements. Shin et al.[1] measured power consumption by applying several workloads (random, orderly, etc) on SSD. Bjorling et al.[2] measured SSD power consumption with 100MHz sampling. SSD power consumption characteristics were recognized through uFlip benchmark. Yoo et al.[3] used an oscilloscope with higher sampling rate than NAND calculation speed to measure SSD power consumption and inferred inner writing operation of SSD. These studies analyzed SSD power consumption followed by workload. However, influence of firmware on SSD power consumption could not be accurately known because inner control algorithm of SSD could not be known. Also, change of power consumption could not be measured when changing SSD inner factors..

SSD power consumption study using simulators was also conducted. Park et al.[4] developed a simulator that can predict power consumption, but only analyzed the correlation between energy consumption and performance when waiting power was simply reduced. However, selected number of channels which is the hardware factor and FTL which is the software factor influencing SSD performance to analyze the influence they have on SSD.

## III. COMPOSITION OF SSD

### A. NAND Device

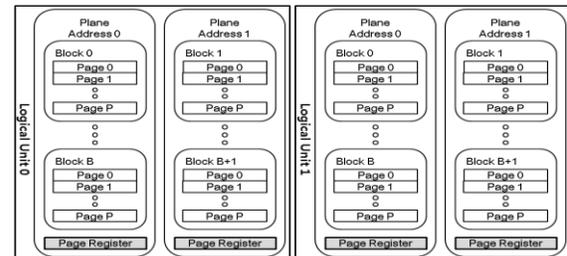


Fig. 1 NAND Flash Memory Map[7]

NAND chip is composed of several logical units (LUN). (Refer to Fig. 1) LUN is the minimum unit to perform orders which is composed of at least one page resistor and flash array. There are several blocks in the flash array. Block is the minimum unit to delete data inside flash array of LUN. Page is the minimum unit of reading and writing. Page is categorized

into data area and spare area. User data enters data area and ECC code or metadata can be entered in spare area. Page register is temporary saving space for data to be read or have been read at page to be saved.

In-place update is impossible in NAND flash. To update data, block corresponding to it page must be first eliminated for writing to be performed on particular page. There is limitation in number of eliminations for each block which is about 10,000 times for MLC(Multi-Level Cell) flash memory and 100,000 times for SLC(Single-Level Cell) flash memory chip.

### B. Channel/way Structure

Maximum reading speed of one NAND flash memory chip is 66MB/s and maximum writing speed is 6MB/s which is slower than the maximum speed of HDD that is 150MB/s. Therefore, high SSD performance can be gained by parallel performing reading/writing by connecting several NAND flash memories in several channel/ways. Speed and bandwidth of channels connecting each flash memory chip when parallel arranging NAND flash memory chips, should also be considered to bring improvement of overall performance.

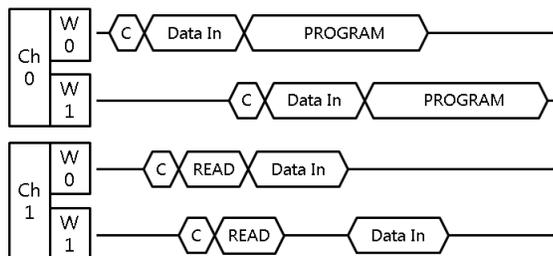


Fig. 2 Calculation Algorithm

Channel is the I/O bus independently operating by the SSD controller. I/O request can be simultaneously delivered to several channels. However, several  $\mu$ s of channel delay occur when changing channel [5].

Several NAND flash can be connected to each channel. Way is the data path connecting each channel and NAND flash. I/O must be orderly delivered to the connected NAND flash because way share channel. Way delay is delay occurring from orderly I/O deliver to the NAND flash connected to channel. Way delay is the time needed to deliver NAND orders and data which can vary depending on NAND interface. Fig. 2 shows the calculation flow chart.

### C. Flash Translation Layer(FTL)

FTL is storage of original file system and NAND flash which is located between the device. It is middle-ware to hide elimination calculation of the NAND flash. Most FTL hide writing before elimination characteristic of NAND flash through mapping information management between logical address and physical address. FTL can be categorized into FTL and Hybrid FTL[8][9][10] depending on mapping unit.

Update cost of original data on SSD varies depending on mapping unit of FTL. PAGE FTL allocated a new page when updating data and updated mapping information. Also, the previous page condition is invalidated. That is why there is no

additional calculation cost before performing garbage collection. Garbage collection cost of PAGE FTL varies depending on the garbage collection algorithm. It varies depending on over-provision factor in greedy algorithm[11]. When data update occurs in BLOCK FTL, block is replaced. LPN from block mapping method can only be used in fixed page offset. If there is written data on page offset of the corresponding LPN, other page of the same block cannot be written on even though it is an empty page. That is why a new empty block is allocated to copy valid pages of original block and eliminate original blocks. If empty pages are left within log block when updating original data in hybrid FTL, cost same as page FTL occurs. However, cost of data block and log block merge occurs when using all log block. If there are n pages in log black, n page data updates occur and cost of maximum n page copying and cost of 2 block elimination occur.

Other than mapping function, FTL has functions of garbage collection (GC) and wear-leveling. GC is the function of deleting invalidated pages to secure empty block. Wear-leveling prevents fast bad blocking of only a few blocks by evenly organizing number of block eliminations.

## IV. EXPERIMENT AND EVALUATION

In this study, influence of channel number and FTL among several design factors of SSD was looked into.

### A. Simulator

We developed a SDD simulator. This simulator was developed based on DiskSim[12]. It is a simulator that can predict performance and electric consumption of SSD. 3 types of FTL(PAGE, BLOCK, BAST FTL) were implemented to look into the influence of inner software.

### B. Workload

TABLE I  
WORKLOAD CHARACTERISTIC

Workload	Read (%)	Seq. (%)	Arrival Rate (IOP/s)	Length of Trace (sec)
Financial1[13]	18.3	1.9	91.7	5453.9
Financial2[13]	75.6	3.3	126.5	3951.0
Homes[14]	28.3	44.3	23.4	42646.9
MSNfs[15]	62.7	9.5	2021.3	247.3

TABLE I is the workload used in the experiment. Financial 1, 2 are OLTP program trace collected from financial institution made from the Storage Performance Council(SPC) [13]. MSNfs is trace collected from the MSN Storage back-end file server using ETW(Event Tracing for Windows) from Microsoft[15]. Homes is collected trace of research group activity(development, test, experiment, paper, graph) occurring from the home directory of NFS server [14].

Experiments were conducted on 3 FTLs by changing number of channels to 2, 4, 8, 16 in each workload. The page size of NAND is 4KB, reading, writing, elimination speed of each calculation is 50  $\mu$ s, 500 $\mu$ s, 2ms, respectively. Consumption current is 25mA for calculation and 5mA for waiting. 30 $\mu$ s was needed in changing channel and time delivering data to page register was 82 $\mu$ s. The number of NAND flash used in all

experiments was equivalent as 16. Each size of NAND flash was 32GB for the total SSD capacity to be 512GB.

### C. Experiment Results

Fig. 3 shows average response time. The x axis shows workload and FTL type, and y axis shows average reaction time. Performance change due to number of channel in 3 types of FTL was 1%. PAGE FTL was 48 times faster than FTL difference and 1.8 times faster than BAST FTL.

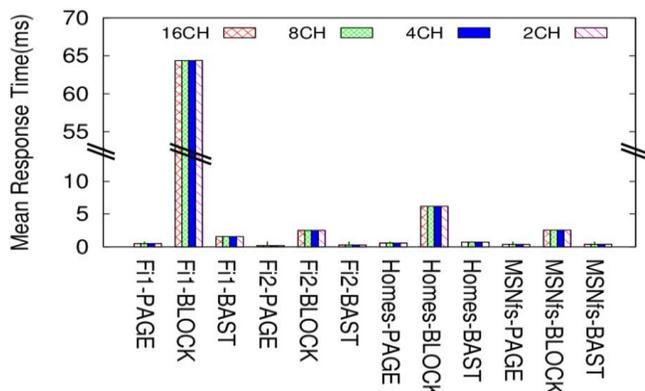


Fig. 3 Average Response Time

Fig. 4 shows average power consumption. Difference by change of channel number is 0.1%. Between FTL, BLOCK FTL consumes 1.6 times more than PAGE FTL and BAST FTL consumes 1.03 times more than PAGE FTL.

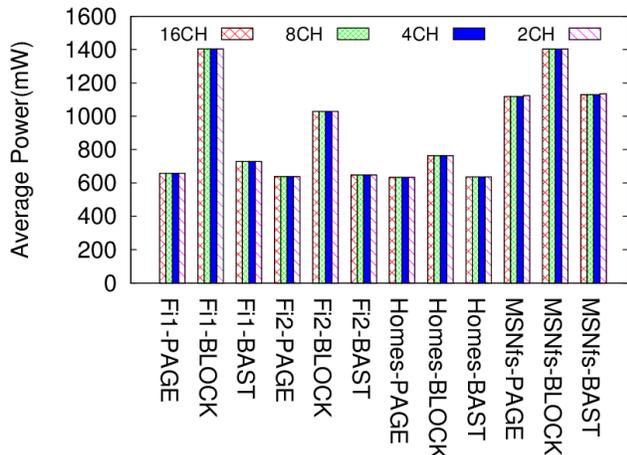


Fig. 4 Average Power Consumption

Influence of FTL on both performance and power consumption is bigger than influence of channel number. The reason of less influence is because the number of parallel units is the same. Also, time changing channel and difference of way delay is 10% of calculation time which has low influence on performance

FTL had influence on performance and power consumption with cost difference updating data. TABLE II shows number of valid page copying and block elimination occurring while conducting workload in each FTL. Page FTL filled all data area with valid page to induce GC and set GC when using 1% of empty block, but GC was not performed. For BLOCK FTL, an

average of 4.37 million valid page copying and 0.32 million block elimination occurred. In BAST FTL, average of 0.21 million valid page copying and 7000 block elimination had occurred. Data update cost of page FTL without performance of GC was 0. Data update cost of BLOCK FTL was an average of 14 page copying and 1 block elimination. Data update cost of BAST FTL was an average of 0.7 page copying and 0.02 block elimination. When data update occurs much, this cost gives substantial influence on performance and power consumption. This gives reason to why good performance has low power consumption. It was shown that the lower the data update cost of FTL was, number of additional calculation was reduced to have fast performance and low power consumption.

TABLE II  
NUMBER OF VALID PAGE COPY AND NUMBER OF BLOCK ERASE

Workload	FTL	$N_{copy}$	$N_{erase}$
Financial1	BLOCK	37493301	396085
	BAST	2321168	23519
Financial2	BLOCK	3173894	97699
	BAST	43182	742
Homes	BLOCK	6428556	203905
	BAST	171772	2407
MSNs	BLOCK	6428556	203905
	BAST	87756	16848

### V. CONCLUSION

We have used a simulator to look into the number of channels and power consumption change by FTL. Change of performance and power consumption by change of channel when using same number of NAND was very low to be around 1%. However, change of performance and power consumption by FTL showed dozens of times of change.

Currently influence of channel and FTL was looked into, but factors influencing SSD performance such as page size, IO latency, garbage collection, wear-leveling, and etc. are still remained. Optimization methods to increase performance and reduce power consumption by analyzing the influence of each factors and synthesizing these results will be researched.

### ACKNOWLEDGMENT

This work is sponsored by IT R&D program MKE/KEIT. [No.10035202, Large Scale hyper-MLC SSD Technology Development].

### REFERENCES

- [1] OCZ. Ocz Z-drive r4, 2012. URL <http://www.oczenterprise.com/ssd-products/z-drive-r4-c-series.html>
- [2] OCZ. Ocz Vertex 4. URL <http://www.ocztechnology.com/ocz-vertex-4-sata-iii-2-5-sd.html#specifications>.
- [3] S. Shin and D. Shin, "Power Consumption Characterization of Flash Memory SSD," ICEIC 2010
- [4] M. Bjorling et al., "uFLIP: Understanding the Energy

Consumption of Flash Devices," IEEE Data Engineering Bulletin, 33(4):48-54, 2010

- [5] B. Yoo et al., "SSD Characterization: From Energy Consumption's Perspective," USENIX HotStorage 2011
- [6] J. Park et al., "Power Modeling of Solid State Disk for Dynamic Power Management Policy Design in Embedded Systems," Software Technologies for Embedded and Ubiquitous Systems, vol. 5860, 2009
- [7] Open NAND Flash Interface Specification (ONFI) 3.0, <http://www.micron.com/~media/Documents/Products/Other>
- [8] K. Jesung, K. Jong Min, S. H. Noh, M. Sang Lyul, and C. Yookun. "A space-efficient flash translation layer for compact flash systems." Consumer Electronics, IEEE Transactions on, 48(2):366–375, 2002.
- [9] S.-W. Lee, W.-K. Choi, and D.-J. Park. "FAST: An Efficient Flash Translation Layer for Flash Memory." volume 4097 of Lecture Notes in Computer Science, pages 879–887. Springer Berlin Heidelberg, 2006.
- [10] S. Lee, D. Shin, Y.-J. Kim, and J. Kim. "Last: locality-aware sector translation for nand flash memory-based storage systems." SIGOPS Oper. Syst. Rev., 42:36–42, October 2008. ISSN 0163-5980
- [11] P. Desnoyers. "Analytic modeling of ssd write performance." In Proceedings of the 5th Annual International Systems and Storage Conference, SYSTOR '12, pages 14:1–14:10. ACM, New York, NY, USA, 2012. ISBN 978-1-4503-1448-0.
- [12] The DiskSim, <http://www.pdl.cmu.edu/DiskSim>
- [13] Umass Trace Repository, <http://traces.cs.umass.edu/>
- [14] R. Koller et al., "I/O deduplication: Utilizing content similarity to improve i/o performance," Trans. Storage, 2010
- [15] S. Kavalanekar et al., "Characterization of storage workload traces from production windows servers," In Workload Characterization, 2008