

# 저전력 FTL Modeling을 위한 기존의 FTL 소모전력 분석

조석희<sup>o</sup>, 안기주, 윤준식, 원유집  
한양대학교

misost@ece.hanyang.ac.kr, akjlove99@nate.com, younjunsik@nate.com, yjwon@ece.hanyang.ac.kr

## Power consumption analysis of existing FTL for low power FTL modeling

Seokhei Cho<sup>o</sup>, Kiju An, Junsik Yoon, Youjip Won  
Department of Electronics and Computer Engineering Hanyang University

### 요 약

플래시 메모리는 비휘발성으로 빠른 접근 속도와 저 전력 소비 그리고 간편한 휴대성 등의 장점을 가진다. 따라서 최근에 많은 임베디드 시스템에서 많이 사용되고 있다. 또 플래시 메모리는 하드웨어 특성상 플래시 변환 계층(FTL : Flash Translation Layer)이라는 시스템 소프트웨어를 필요로 한다. 이 FTL의 주요 기능은 파일 시스템으로부터 내려오는 논리주소를 플래시 메모리의 물리주소로 변환하는 일이다. 특히 오늘날 컴퓨터 시스템에서 중요하게 생각하는 분야 중 하나가 저 전력 소비인 만큼 성능을 최대한 보장하면서 사용 전력을 절감하기 위해 많은 기술들을 연구하고 개발하고 있다.

본 논문에서는 DiskSim 시뮬레이터를 기반으로 SSD(Solid State Disk) 모듈이 패치 된 기존 시뮬레이터에 SSD의 성능 및 에너지 사용량 집계 기능을 추가한 시뮬레이터를 사용하여 FTL의 소모 전력을 측정하여 비교해 보인다.

## 1. 개요

### 1.1 연구동기

과거에는 모바일 장비에 사용되는 저장 매체로 하드 디스크 드라이브가 주로 사용되었으나, 최근에는 NAND 플래시 메모리 혹은 고성능, 고용량을 위해서 Solid State Disk(SSD)를 채택하는 제품이 늘어났다. SSD는 다수의 NAND 플래시 메모리 칩을 병렬적으로 구성하여 오늘날의 컴퓨터 기억장치로 주목 받고 있다. 기계적인 장치가 없어 하드디스크에 비해 전력을 적게 사용하며 외부 충격에 강할 뿐만 아니라 처리 속도 면에서도 우수한 성능을 나타낸다. 그러나 NAND 플래시 메모리의 특성상 In-Place-Update가 불가능하고, 하드 디스크와 달리 같은 섹터에 데이터를 쓰고 지울 수 있는 횟수가 제한되어 있어, 이를 충분히 고려하여 개발하지 않으면 오히려 하드디스크보다 성능이 떨어지는 장비가 될 수 있다. 또한, 앞으로 개발될 새로운 SSD들은 고성능과 고용량을 위해 지금보다 더 복잡한 FTL 알고리즘을 사용하고, 병렬화 기법을 극대화해서 사용하게 될 것이다. 그로 인해 미래의 SSD들은 매우 높은 순간 소모 전력 값을 가지게 될 것이다. 예상되는 문제점을 보완하기 위해서는 기존의 FTL 알고리즘의 소모 전력을 측정, 평가, 분석하여 이를 통해 SSD의 구성요소를 고려한 소모전력 모델을 세워, 저 전력 FTL을 설계해야한다.

SSD 장치를 개발하는 회사의 입장에서는 제품의 성능과 안정성, 소모전력 등을 측정하기 위해서 개발된 제품을 이용하여 실제 측정을 하게 되면 개발비용이 많이 든다. 그러므로 실제 생산된 제품이 존재하지 않은 상태에서 FTL 알고리즘의 시뮬레이션을 통해 소모전력 측

정을 할 수 있다면 개발 기간의 단축뿐만 아니라 개발 비용까지 절약할 수 있을 것이다. 본 논문은 시뮬레이션을 통하여 기존의 FTL의 소모 전력을 측정 할 것이다.

### 1.2 관련연구

이 장에서는 디스크 기반의 성능과 전력 사용량을 모델링 하는 연구와 NAND 플래시 메모리, SSD 기반의 성능과 전력 사용량을 모델링 하는 연구에 대해서 논의한다.

DiskSim [1]은 Michigan University, Carnegie Mellon University 에서 개발된 스토리지 시스템 분야에서 연구를 지원할 수 있도록 만든 효율적이고 정확하며 다양한 설계 환경을 적용할 수 있는 디스크 시스템 시뮬레이터이다. 이 시뮬레이터의 소스코드는 인터넷에 공개되어 있고, 다양한 목적으로 추가 개발하기 쉽도록 구조화되어 있다. 실제 하드디스크 드라이브 디바이스 이외에도 RAM 디스크 메모리 디바이스에 대한 시뮬레이션이 가능하도록 모델링이 추가 구현하여 배포되어 있다.

Agrawal et al. [2]은 SSD에 대해서 성능을 측정할 수 있도록 DiskSim을 기반으로 SSD 모델을 추가 하였다. 기존의 DiskSim으로는 다룰 수 없는 Address Mapping 알고리즘, Garbage Collection, Parallelism 알고리즘이 추가되어 있고, SSD만의 설계 속성을 파라미터화하여 SSD의 성능 평가를 위한 시뮬레이터를 개발하였다.

John Zedlewski et al. [3]에 의해 만들어진 Dempsey (Disk Energy Modeling and Performance Simulation Environment)는 DiskSim을 기반으로 하드 디스크의 성능뿐만 아니라 전력 사용량에 대해서도 모델링 할 수 있도록 시뮬레이터를 개발하였다. Dempsey는 전력 소모에 대한 특성 파라미터 값을 모델링 하고자 하는 실제

디스크를 단위 전력 측정 장치를 통해 실측한 값으로 사용한 것이 특징이다. 이렇게 함으로써 제조업체에서 제공하는 데이터에 의존하지 않고 정확한 값의 파라미터를 사용함으로써 시뮬레이터의 전력 측정 모델링 결과를 좀 더 정확하게 해주는 효과를 가질 수 있다.

Cagdas Dirik et al. [4]의 시뮬레이터를 통한 SSD에 대한 성능 모델링 연구는 Agrawal et al.[2]에서 연구한 방식과 유사하다. 그러나 이 논문에서는 NAND 플래시 메모리의 특성 중심인 디바이스 레벨보다는 좀 더 시스템 레벨에서의 성능을 모델링 할 수 있는 방향으로 접근하였다. Bus의 수, 각각의 속도와 Bandwidth, 동시 접근 속성을 다양화함으로써 시스템 레벨의 성능을 측정할 수 있는 모델링을 연구하였다.

Lee et al. [5]의 CPS-SIM (Clock Precision SSD Simulator)은 SSD의 성능을 모델링 하는 시뮬레이터를 제공하며, 시뮬레이션 내부의 동작이 글로벌 Clock pulse를 기준으로 모든 칩과 버스의 상태 및 성능 데이터가 수집된다는 것이 특징이다.

본 논문의 구성은 다음과 같다. 2장에서는 SSD의 구조와 작동에 대해 개략적으로 살펴본다. 3장에서는 FTL의 구성 및 작동을 살펴보고 4장은 SSD의 소모전력 modeling에 대한 방법론을 설명한다. 5장은 시뮬레이터를 통한 실험결과를 제공하고 6장에서 결론을 맺는다.

## 2. SSD 구조 및 작동

### 2.1 NAND FLASH의 연산 : read/write/erase

플래시 메모리에 데이터를 저장하고 읽는 방식은 기존의 RAM이나 하드디스크와는 다른 특성을 가지고 있다. 기존 저장 장치의 기본 연산인 읽기/쓰기와 다르게 NAND 플래시 메모리는 읽기/쓰기/소거 연산을 가진다. NAND 플래시 메모리에서 읽기와 쓰기는 페이지 단위로 동작하고 소거는 블록 단위로 동작한다. 소거연산은 쓰기연산에 비해 비용이 크기 때문에 FTL은 소거연산이 필요한 덮어쓰기 연산을 빈 공간에 대한 쓰기연산으로 변환하기 위한 알고리즘을 사용한다. 알고리즘에 따라 같은 작업에 대한 읽기/쓰기/소거 연산 횟수도 다르게 나타나기 때문에, 효율적인 알고리즘을 사용한 FTL을 개발하면 플래시 메모리의 더 좋은 성능과 긴 수명을 보장할 수 있다.

### 2.2 SSD 구조

SSD는 NAND 플래시 메모리를 이용한 새로운 저장장치로서 외부 충격에 강하고 소비 전력이 작으며 데이터 접근 속도가 빠른 장점이 있다. SSD의 빠른 데이터 접근 속도는 보조 기억 장치의 병목 현상을 해소할 수 있다. SSD의 기본 구성은 ARM 프로세서, SDRAM 캐시, SDRAM 버퍼, 그리고 다수의 플래시 메모리 칩과 버스로 이루어진다. ARM 프로세서는 SSD를 제어하는 FTL 코드를 수행하고, SDRAM 버퍼는 읽기/쓰기 요청된 데이터를 보관한다. 또한 요청들을 SDRAM에 보관하고 이들을 다수의 버스를 통해 병렬적으로 처리하여 전체적인 성능을 높인다. 그림 2에서 보듯이 하나의 버스에 다수의 칩들이 연결된다. 하나의 버스에 연결되는 칩들의 수가 증가하면 SSD의 전체 용량은 증가하고 또한 병

렬로 동작할 수 있는 칩이 증가하여 성능도 증가할 수 있다. 그러나 버스 하나의 데이터 전송량은 한정되어 있으므로 버스가 병목이 될 수 있다. 이를 해결하는 한 가지 방법으로 버스 수를 증가시키면 병렬로 전송하는 데이터가 증가하여 SSD의 처리 속도가 증가하게 된다.

## 3.FTL 구성 및 작동

FTL (Flash Translation Layer)의 주요 기능은 파일 시스템으로부터 내려오는 논리 주소를 플래시 메모리의 물리 주소로 변환하는 작업을 합니다. FTL은 주소 사상 테이블을 구성하는 정보의 크기에 따라 블록 사상 FTL 기법과 페이지 사상 FTL 기법으로 크게 구분합니다.

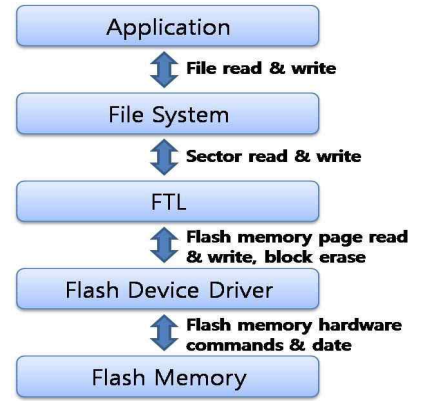


그림1 .FTL의 구조

### 3.1 페이지 사상 기법

페이지 사상 기법은 주소 사상 테이블을 논리 섹터 크기와 같은 페이지 단위로 구성 하는 것으로 논리 섹터 번호와 플래시 저장 장치의 전체 물리 페이지 번호가 일대일로 사상된다. 모든 논리 섹터가 플래시 블록 내의 페이지 위치와 상관없이 쓰기 요청이 들어온 순서대로 기록한다. 페이지 사상 기법은 쓰기를 위해 사용하고 있는 블록의 페이지를 모두 소비하기 전까지 새로운 블록을 할당 하지 않기 때문에 플래시 메모리의 이용률이 상당히 높다. 또한 블록 내 페이지 위치가 논리 주소의 수서와 상관없이 관리되기 때문에 임의 접근에 용이하다. 하지만 페이지 사상 테이블의 정보는 플래시 메모리 전체 페이지를 관리해야 하기 때문에 메모리 사용량이 많을 수밖에 없다.

### 3.2 블록 사상 기법

블록 사상 기법은 주소 사상 테이블의 단위를 플래시 메모리의 블록 크기로 구성한다. 블록 사상 기법은 하드웨어 캐시 정책 중 하나인 직접 사상 캐시와 같은 방식으로 하나의 논리 섹터 번호는 하나의 물리 블록 내에 정해진 페이지 위치에 대해서만 참조가 가능하다. 주소 사상 테이블의 크기는 플래시 메모리의 전체 물리 블록의 개수에 비례하기 때문에 상대적으로 상당히 적은 메모리를 사용하여 주소 사상 테이블을 구성할 수 있는 장점이 있다. 하지만 논리 주소에 대한 물리 블록 내의 사용되지 않는 페이지가 남아 있을 수 있어 플래시 메모리의 이용률이 낮은 단점을 가지고 있다.

### 3.3 BAST

페이지단위 사상과 블록단위 사상을 혼합한 방식인 BAST(Block Associative Sector Translation) [6]는 하나의

가상 블록을 각각 하나의 물리 블록과 로그 블록으로 연결된다. 데이터는 먼저 데이터 블록에 먼저 기록한다. 데이터가 수정되면 로그 블록의 첫 번째 페이지부터 순서대로 기록한다. BAST 기법은 시스템의 전체 로그 블록의 수를 제한하고, 만약 어떤 데이터가 수정되었을 때 사용할 수 있는 로그 블록이 없다면 기존의 로그 블록과 데이터 블록을 합병하여 지운 후 새로운 로그 블록을 할당한다.

#### 4. SSD의 소모전력 모델링

소모전력 모델링의 목적은 실질적으로 발생한 소비전력에 대한 값을 얻기 위한 것이다. 시스템에서 소비된 전력의 양을 알기 위해서는 작동된 시간동안에 각각 다른 단계에서의 활성화된 값을 알아야 한다. 플래시 메모리에서 전력을 사용하는 주요 동작은 읽기, 쓰기, 소거, 데이터 전송 그리고 동작을 하지 않는 Idle 상태가 있다.

##### 4.1 SSD 전력 소모 분석

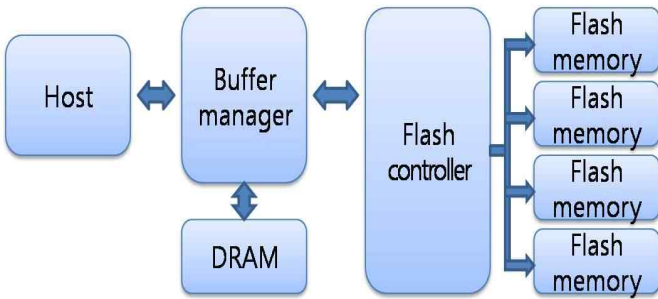


그림 2. SSD Module Block Diagram

ARM 코어는 FTL 소프트웨어를 처리하기 위한 중앙 처리장치이다. ARM 코어는 Active 모드와 NOP을 처리하는 Idle 모드로 크게 나누어 동작한다. DRAM Buffer는 SSD의 read/write buffer management를 사용하기 위한 내부 모듈로써 NAND에 비해서 속도는 빠르나 비교적 많은 전력을 소모하고 있다. Flash memory 칩 하나에서 사용되는 소모 전력량은 굉장히 적으나, 대용량, 고성능 SSD에서는 많은 칩을 사용하기 때문에 전체 사용 전력량에 큰 부분을 차지하게 된다.

##### 4.2 시뮬레이터 설계 및 구현

본 논문에서는 DiskSim 기반의 SSD 모듈이 패치된 기존 시뮬레이터를 기반으로 하여, SSD의 성능 및 에너지 사용을 예측할 수 있도록 개발하였다.

본 논문에서는 시뮬레이터에서 CPU의 Power 모드와 Idle 모드일 때의 작업 시간을 분리하기가 어렵기 때문에 SSD가 각각의 I/O 요청에 대해서 처리하는 시간 동안을 Power 모드의 시간으로, 그 외에는 Idle 모드 시간으로 가정하였다.

각 SSD 내부 장치 별 전력을 측정하는 방식은 다음과 같다.

$$E_f = (C_{op} * V_f) * T_{op}$$

$E_f$ 는 Flash에서 사용되는 총 에너지양이고,  $C_{op}$ 는 Flash

의 각 operation을 수행하면서 소요되는 단위 전류량이다.  $V_f$ 는 Flash 메모리에 인가되는 전압이며  $T_{op}$ 는 Flash 메모리의 operation을 수행하는 동안의 소요 시간을 의미한다.

$$E_{pc} = (C_{pc} * V_c) * (T_{total\ op} - T_{idle\ op})$$

$$E_{ic} = (C_{ic} * V_c) * T_{idle\ op}$$

$$E_c = E_{pc} + E_{ic}$$

$E_{pc}$ 는 Power 모드에서 CPU가 사용한 에너지이고,  $C_{pc}$ 는 Power 모드의 전류,  $V_c$ 는 CPU에 인가된 전압이다.  $E_{ic}$ 는 Idle 모드에서 사용한 CPU의 에너지이고,  $C_{ic}$ 는 Idle 모드에서의 전류이다.

$$E_r = (C_{op} * V_r) * T_{op}$$

$E_r$ 은 DRAM에서 사용된 총 에너지양이고,  $C_{op}$ 는 DRAM 각 operation시에 요구되는 전류 크기이다.  $V_r$ 은 DRAM에 인가된 전압이며,  $T_{op}$ 는 DRAM 각 operation을 수행하는데 걸린 시간을 의미한다.

본 시뮬레이터에서는 BUS에서 사용하는 에너지양에 대해서는 데이터를 전송하는 동안의 에너지만을 계산하며 Idle인 경우는 무시하였다.

$$E_b = (C_b * V_b) * T_{op}$$

$C_b$ 와  $V_b$ 는 Bus에 인가되는 전류와 전압을 의미하며  $T_{op}$ 는 버스를 통해 데이터를 전송하는 동안의 작업 시간을 의미한다.

#### 5. 실험

본 실험은 DiskSim기반의 SSD 모듈이 패치된 기존 시뮬레이터에 SSD의 성능 및 에너지 사용량 집계 기능을 추가한 시뮬레이터를 사용하였다. Workload는 리눅스 환경에서 blktrace를 이용하여 26개 프로그램의 실행 및 설치에 대한 trace 파일을 추출하여 사용하였다.

	Energy	Time	Read	Write	Merge
Page	92.4J	207sec	28648	54170	0
Block	116.7J	207sec	28648	53878	4822
Bast	92.5J	207sec	28648	54158	38

표 1. FTL 알고리즘에 따른 성능 및 에너지 사용량

표1은 실험한 trace 중에서 FTL간의 차이를 보인 것 중 하나를 정리한 것으로 torrent라는 P2P 프로그램을 실행했을 때의 trace이다. Energy는 전체 소모 Energy이고 Time은 전체 trace를 수행한 시간이다. 그리고 Read, Write, Merge는 플래시 칩에 연산이 일어난 횟수를 말한다. 표1을 보면 Block FTL의 에너지 사용량이 Page나 Bast FTL에 비해 16.7% 정도 더 에너지 사용량이 많다. Block FTL의 에너지 소모량이 늘어난 이유는 Merge 연산 발생에 의한 것으로 아래 그림들에서도 확인할 수 있다. Block FTL의 전력 소모량이 커진 시간과 Merge에 의해 I/O 처리가 지연된 영역의 시간이 일치하는 것을 확인할 수 있다. Bast FTL에서도 Merge가 발생했지만 횟수가 적어 거의 영향이 없었다.

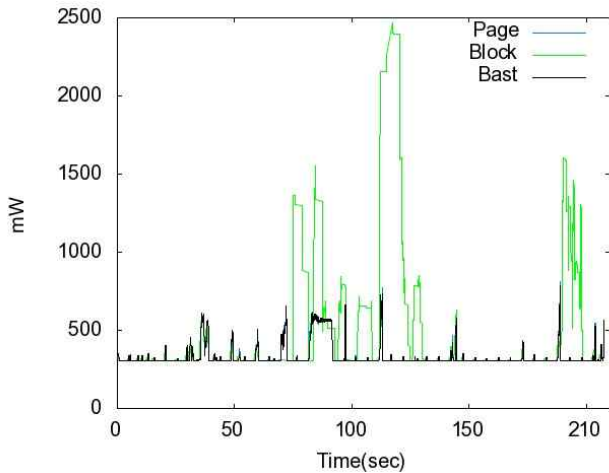


그림 3. 시간에 따른 전력 소모 변화

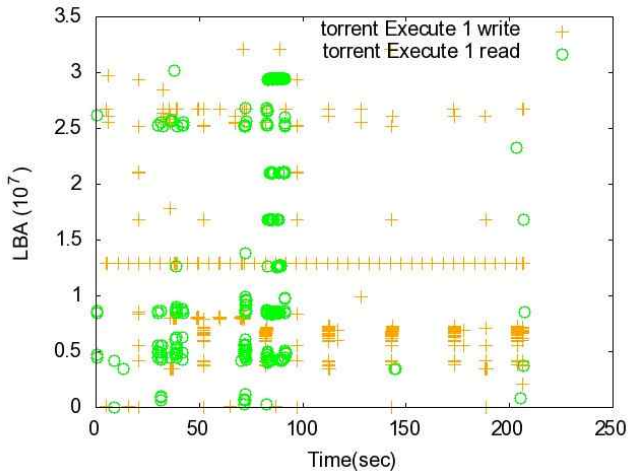


그림 4. Page & Bast FTL의 LBA-Time 그래프

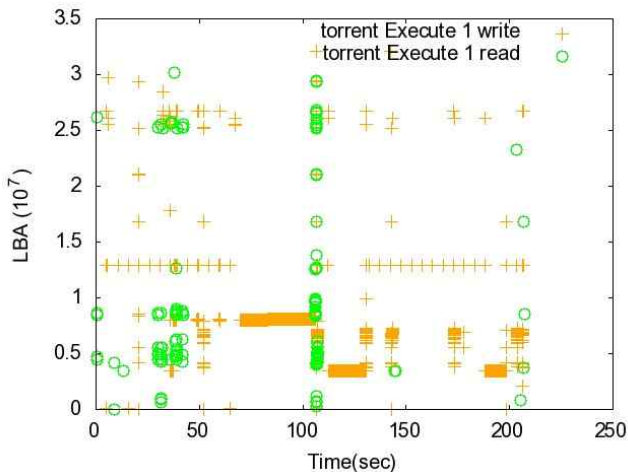


그림 5. Block FTL의 LBA-Time 그래프

## 6. 결론

본 논문에서는 FTL에 따른 소모전력 값의 차이를 보여주고 있다. 가장 기본이 되는 FTL인 페이지 & 블록 사상 기법과 BAST의 소모전력 값을 측정함으로써 FTL

별 효율을 알 수 있고, SSD장치를 직접적으로 이용하여 측정하지 않고 DiskSim 기반의 시뮬레이터를 이용함으로써 저비용으로 자료수집이 가능하다. 3장에 소개한 FTL의 소모 전력에 대한 정확한 분석을 토대로 미래를 위한 저 전력 FTL에 대한 모델링이 필요하다.

## 7. Acknowledgement

본 연구는 지식경제부 및 한국산업기술평가 관리원의 IT R&D 프로그램(Large Scale Hyper-MLC SSD Technology Development, No. 10035202)의 일환으로 수행하였음.

## 참고문헌

- [1] J. S. Bucy, G. R. Ganger, and et al. The DiskSim Simulation Environment Version 4.0 Reference Manual. <http://www.pdl.cmu.edu/DiskSim>
- [2] Agrawal, N., Prabhakaran, V., Wobber, T., Davis, J. D., M. Manasse, and Panigrahy, R. 2008. Design Tradeoffs for SSD Performance. In Proceedings of the USENIX Annual Technical Conference (Boston, MA, June 2008). USENIX 2008.
- [3] Zedlewski, J., Sobti, S., Garg, N., Zheng, F., Krishnamurthy, A., and Wang, R. 2003. Modeling Hard-Disk Power Consumption. In Proceedings of the 2nd USENIX Conference on File and Storage Technologies (San Francisco, CA, March 31-31, 2003). Conference on File and Storage Technologies. USENIX Association, Berkeley, CA, 217-230.
- [4] Dirik, C. and Jacob, B. 2009. The performance of PC solid-state disks (SSDs) as a function of bandwidth, concurrency, device architecture, and system organization. In Proceedings of the 36th Annual International Symposium on Computer Architecture (Austin, TX, USA, June 20-24, 2009). ISCA'09. ACM, New York, NY, 279-289.
- [5] Lee, J., Byun, E., Park, H., Choi, J., Lee, D., and Noh, S. H. 2009. CPS-SIM: configurable and accurate clock precision solid state drive simulator. In Proceedings of the 2009 ACM Symposium on Applied Computing (Honolulu, Hawaii). SAC'09. ACM, New York, NY, 318-325.
- [6] J. Kim, J.-M. Kim, S.-H. Noh, S.-L. Min, and Y. Cho. A space-efficient flash translation layer for compact flash system. IEEE Transaction on Consumer Electronics, 48(2):366-375. 2002.