

전력측정을 통한 FTL 동작예측

유백은^o, 원유집

한양대학교

starhunter@ece.hanyang.ac.kr, yjwon@ece.hanyang.ac.kr

Empirical Analysis of FTL Through Power Measurement

Balgeun Yoo^o, Youjip Won

Department of Electronics and Computer Engineering
Hanyang University

요약

차세대 저장매체로 기대되는 Solid State Drive(SSD)는 저전력, 고성능이 그 장점으로 알려져 있다. 하지만 일반적인 인식과는 달리, 현재 판매되고 있는 SSD의 소모전력은 일반적인 Hard Disk Drive(HDD)와 비교해 볼 때 에너지 절약의 관점에서 장점이 크지 않다. 에너지 친화적인 SSD를 만들기 위해 새로운 FTL을 고안하려 해도 기존의 상용 SSD들은 내부의 정보가 공개되지 않고 실험을 통해서도 알 수 없었다. 이러한 FTL의 동작을 예측하기 위해 본 논문은 전력분석 기법을 사용한다. 기존의 전력분석 기법들은 샘플링 간격이 매우 커서 실제 SSD가 소모하는 최대 소모전류 값을 수집하지 못하고 있다. 본 논문은 이러한 최대값을 매우 짧은 샘플링 간격의 크기를 줄여 모두 검출하였고 이것을 통하여 H/W 의존적 정보인 채널과 way의 구성을 예측하였다. 정확한 데이터 수집을 바탕으로 본 논문이 제시하는 채널과 way 분류 방법으로 SSD의 소모전류를 분석하면 입력되는 I/O에 대하여 SSD의 FTL이 어떻게 동작하는지를 예측할 수 있다. 실험은 Intel의 X-25M SSD에 대해 진행하였으며 8KB당 1개의 채널이 동작하고 채널당 16개의 way를 가지고 있다는 분석 결과를 도출하였다.

1. 서론

SSD는 구동속도의 한계에 부딪힌 HDD를 대체하는 새로운 저장매체로 등장하였다. 또한 21세기의 그린IT 이슈와 맞물려 저전력 저장매체로서도 기대되고 있다. 일반적으로 SSD는 기계적 구동계를 가지고 있지 않기 때문에 HDD에 비해 소모전력이 적은 것으로 알려져 있다. 하지만 실제 측정결과는 그러하지 못하다. 그림 1은 3가지의 SSD와 1가지의 HDD에 대하여 컴퓨터의 전원을 켜고 SSD가 초기화 될 때 소모하는 전류를 측정하는 것이다. 실험에 따르면 SSD가 소모하는 전력은 제조사에 따라 매우 다르다. 이렇게 회사별로 SSD의 소모전력에 차이가 발생하는 것은 내부 구성의 차이일 수도 있지만, 사용하는 알고리즘, 즉 FTL이 소모전력을 고려하고 있지 못하기 때문이다. 앞으로 개발되고 출시될 새로운 SSD들은 더 큰 용량과 더 빠른 속도를 위해 지금보다 더 복잡한 내부 알고리즘을 사용하고, 병렬화 기법을 극대화해서 사용하게 된다. 지금까지의 FTL 알고리즘을 그대로 적용하게 된다면 미래의 SSD들은 매우 높은 순간 소모전력 값을 가지게 될 것이 분명하다. 그렇다면, 현재 SSD의 소모전력에서 큰 비율을 차지하는 부분의 동작 알고리즘이나 내부 구성을 저전력에 걸맞도록 바꾸게 된다면 적은 전력을

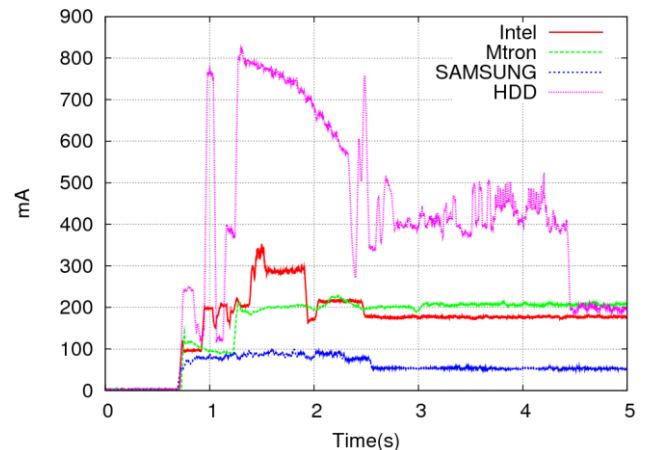


그림 1. SSD와 HDD의 Power On 소모전류

소비하는 SSD를 만들 수 있다. SSD의 플래시메모리 매핑정보는 중간계층인 FTL로 인해 블랙박스로 알려져 있다. 실제로 IOzone이나 LMbench와 같은 벤치마크를 사용해도 SSD의 읽기/쓰기 특징과 특성은 파악할 수 있지만, 내부 매핑정보나 플래시메모리의 동작은 알 수가 없다. 본 연구에서는 플래시 메모리의 동작을 예측하기 위해 전력분석 기법을 사용한다. SSD에 대하여 특정 I/O를 입력하고 그때 발생하는 소모전류를 측정하였다. 측정결과를 윈도우 사이즈 20~30의

이동평균을 취하여 다양한 그래프를 얻었고, 이 결과를 분석하여 실험에 사용한 SSD의 FTL이 입력된 I/O를 어떻게 나누어 플래시 메모리에 저장 하는지를 예측하였다. 결과에 따르면 현재의 SSD는 입력된 I/O를 최대한 빨리 처리하기 위해 채널과 way를 최대한 많이 사용한다. 이러한 정책은 매우 높은 순간 소모전류를 발생시킨다. 더 많은 채널과 way를 가진 SSD를 만들기 위해서는 지금처럼 채널과 way에 비례하여 소모전류가 증가해서는 안 된다. 이를 위해서는 현재의 SSD의 동작을 분석하고 보완점을 찾아야 한다. 본 연구에서는 이러한 과정의 초석으로서 전력분석 기법으로 플래시 메모리의 동작을 예측한다. 2장에서는 관련연구를 소개하고, 3장에서는 어떤 부분의 전력을 측정하는 것인지에 대한 전력측정 요소를 설명하고, 4장에서는 실험환경을 설명하고, 5장에서는 실험의 결과를 분석하고 결과를 설명하고, 6장에서는 결론을 내린다.

2. 관련연구

엔터프라이즈 시장에서는 디스크가 소모하는 전력이 문제가 되고 있다 [1]. 문제의 해결을 위해서 기존 제품들의 소모전력을 측정하고 문제점을 파악하여야 한다. 디바이스의 소모전력 측정에는 직접적인 방법과 간접적인 방법이 있다. 직접적인 방법은 실제 디바이스를 하드웨어 측정 장비로 측정하는 방법이다. 간접적인 방법은 시뮬레이션 툴을 사용하는 방법과 소프트웨어적인 방법을 통해 결과 값을 추측하는 방법이 있다. 간접적인 방법의 사례로 HDD의 소모전력 측정을 위해 사용된 시뮬레이션 툴인 Dempsey 가 있다 [2]. Dempsey는 제조사에서 제공하는 HDD의 자세한 스펙 없이 자동으로 타겟 HDD로부터 정확한 성능과 소모전력에 대한 파라미터를 추출할 수 있다.

직접적인 방법은 크게 3가지가 있다. 먼저 Shin et al.의 논문에서 SSD의 전력 소모량을 좌우하는 요소로 플래시셀의 종류, 디램 버퍼의 크기, 플래시 칩의 병렬성, 펌웨어를 제시하였다 [3]. 소모전력의 측정은 SSD와 입력전원선 사이를 단선시켜, 그 사이에 0.5Ω 의 저항을 연결하였다. 이 저항에 흐르는 전류를 Labview로 수집하고 관찰하였다. 실험에서 공장도 초기화 상태의 깨끗한 SSD를 Clean, 사용량이 누적된 SSD를 Dirty로 설정하고 Clean과 Dirty에 동일한 워크로드를 실행하고 소모전력을 비교하였다. 이 비교를 통해 Dirty에서 발생하는 추가적인 소모전력을 사용자가 아닌 SSD 컨트롤러에서 실행하는 백그라운드연산으로 가정하였다. 또한 입력 워크로드를 4kbyte 부터 64kbyte까지 다양하게 입력하여 각 단계별 소모전력을 수집하였다. 하지만 이 방법은 입력 워크로드의 전체적인 동작 패턴만 파악이 가능하고 SSD의 내부 구성요소 별로 소모하는 정확한 소모전력의 분석이 불가능하다.

Seo et al.의 논문에서는 SATA 전원 선에 대하여

Signametrics SM2040 Digital PCI Multimeter를 사용하였고, DIO Tool이라 불리는 도구를 사용하여 순차쓰기, 임의쓰기, 순차쓰기, 임의쓰기의 워크로드를 입력하고, SSD의 소모전류를 측정하였다 [4]. 실험에 사용한 멀티미터는 데이터 샘플링의 최소 단위가 1ms이다. 이 단위는 SSD 내부 구성요소의 동작시간을 요소 별로 구분 지어 분석하기에는 너무 길다. 이 방법 또한 SSD의 전체적인 동작 패턴만 파악이 가능하다. 내부 구성요소의 소모전력 파악을 위해서는 본 논문에서 제시하는 미시적 측정방법이 필요하다.

Park et al.의 논문에서는 SSD의 소모전력 모델링 요소를 channel과 way로 보고 하나의 way에 대한 읽기/쓰기 연산을 SSD 동작의 최소단위로 보았다 [5]. 실측을 바탕으로 구한 하나의 way에 대한 write를 기준으로 입력 워크로드의 용량에 따라 순차적으로 소모전류가 증가하는 모델링을 제안하였다. 실험에 사용한 SSD의 물리적 페이지 크기가 4kbyte 이지만 4kbyte의 순차쓰기 요청이 2개의 채널에 동시에 2kb씩 쓰이고 입력 워크로드의 크기가 클 때 SSD의 최대 가용자원(channel과 way) 모두 동작한다는 모델을 사용하고 있다. 또한 하나의 way에 대한 write를 최소단위로 설정하고 예상 시나리오에 따라 최소단위의 소모전력이 단계적으로 더해진다고 가정한다. 본 논문에서는 물리적인 페이지 크기가 4kbyte인 SSD를 타겟으로 사용하고 있다. 본 논문의 Samsung SSD 실험결과에서 4kbyte의 소모전력과 8kbyte의 소모전력이 동일한 것을 보여주고 있다. 이는 SSD의 내부 동작이 물리적인 페이지 크기 별로 순차적으로 증가하지는 않는다는 것을 보여준다.

3. 전력측정 요소

SSD는 외부의 다양한 물리적인 연결을 위해 Host Interface Logic을 가지고 있다. 컨트롤러는 시스템 관점에서 SSD를 HDD와 동일하게 보이도록 변환하는 역할을 수행한다. 이를 Flash Translation Layer(FTL) 이라고 한다. SSD 컨트롤러에는 내부 캐싱을 위한 DRAM과 디스크의 플래터에 해당하는 플래시칩이 있다. 플래시칩은 컨트롤러의 Demux/Mux에 연결되어 데이터를 동시에 병렬적으로 저장하고 로드 할 수 있다. 이를 SSD의 channel이라고 부른다. 하나의 channel에

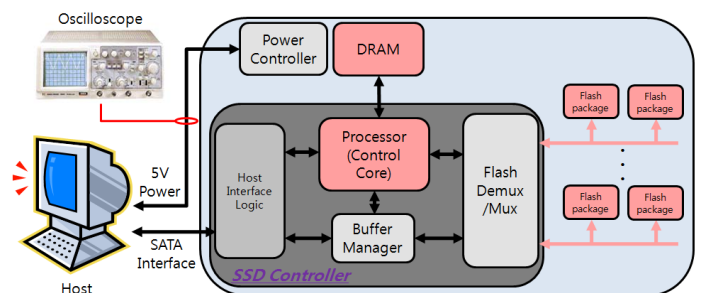


그림 2. 전력측정 요소

연결된 다수의 플래시칩 구성을 way라 하고, 순차적으로 인터리빙 접근이 가능하다. SSD에 입력되는 전원은 호스트 시스템의 파워서플라이에서 5V의 전원을 공급받는다. 본 논문에서는 이 5V 전원선에 대해 전류Probe를 사용하여 회로의 단선 없이 오실로스코프로 소모전류를 측정한다.

4. 실험환경

실험환경은 SSD에 입력되는 5V 전원 선에 대해 전류 Probe를 통해 오실로스코프로 소모전류를 관찰하였다. 특정 워크로드를 SSD에 대해 실행하고, 해당 워크로드의 소모전류 데이터를 오실로스코프에서 수집하였다. 오실로스코프는 Tektronix사의 TDS3032 모델을 사용하였다. 실험에 사용된 SSD의 모델명은 Intel X-25M이고 S-ATA2 인터페이스를 사용하며, MLC 타입의 80GB의 용량을 가지고 있으며, 내부에 4GB용량의 플래시 패키지를 20개가 10채널로 연결되어 있다. 실험에 수집된 데이터는 노이즈 제거를 위해 30개의 단위로 이동평균을 취하였다. 이는 피크값의 데이터를 누락시키지 않으면서 노이즈만을 제거할 수 있는 최소 단위를 경험적으로 선정한 값이다. 실험에는 Linux Kernel 2.6.24.4 버전에서 Ext3 파일시스템을 사용하였다. 실험의 기본 설정으로 look-ahead와 read-ahead 기능을 off 하였고, 상황에 맞게 hdparm 명령으로 타겟 디바이스의 write-caching 기능을 on/off 하였다.

5. 실험결과 및 분석

5.1 Channel의 동작 확인

채널은 SSD의 고속화를 위해 사용되는 내부 병렬화 기법이다. 입력된 I/O가 2개 이상으로 분리되어 각기 다른 채널에 할당되게 되면 해당 I/O는 동시에 처리가 가능하다. 그렇기 때문에 SSD의 FTL은 할당 정책에 따라 입력된 I/O를 최대한 다른 채널에 분리하여 할당하게 되는데, 1개의 채널이 동작하는 소모전류와 10개의 채널이 동작하는 소모전류는 다를 것이다.

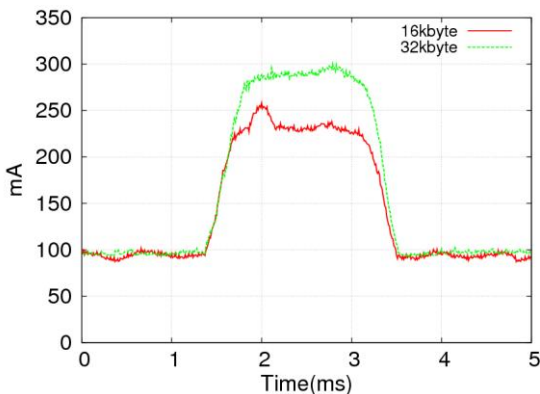


그림 3. channel의 동작 확인

입력된 I/O를 SSD의 물리적인 페이지 크기 이하로 쪼개어 다수의 채널에 할당하는 것은 SSD의 공간 효율을 극히 저하시키므로 이러한 동작은 없다고 가정하였다. 그렇다면 SSD의 물리적인 페이지크기의 I/O가 입력되었을 때 1개의 채널이 동작한다고 할 수 있다. 예를 들어 물리적인 페이지크기가 4KB인 SSD에 4KB의 쓰기연산을 하면서 소모전류를 측정한 다음 이를 해당 SSD의 1개 채널 동작 소모전류로 볼 수 있다. 이렇게 1개 채널 동작 소모전류를 측정한 다음 I/O의 크기를 늘려가며 다수의 채널이 동작하는 I/O를 찾는다. 예를 들어 16KB의 순차쓰기의 소모전류와 32KB의 순차쓰기 소모전류를 측정하여 비교해보면 그림 3과 같이 16KB의 소모전류에 비해 32KB의 경우가 소모전류의 최대값과 최대값의 지속시간이 길 것이다. 4KB부터 일정 용량의 I/O 까지 이러한 형태로 결과값을 정렬하여 그래프를 그려보면 전체적인 채널의 동작을 파악할 수 있다.

5.2 Way의 동작 확인

또 다른 고속화 기법으로 내부 인터리빙 기법인 way가 있다. 하나의 채널에는 다수의 플래시메모리 패키지가 연결되어 있고, 하나의 플래시메모리 패키지는 내부적으로 다수의 플래시메모리로 구성되어 있다. 하나의 채널에 연결되어 있는 다수의 플래시메모리들은 내부 레지스터를 통해 입력 받은 I/O를 쪼개어 Time-Interleaving 하게 처리할 수 있다. 그렇기 때문에 way가 동작하게 되면 소모전류의 최대값은 일정하고 지속시간이 늘어나게 된다. 예를 들어 256KB의 I/O가 입력되었을 때보다 512KB의 I/O가 입력될 때 시간이 대략 2배가 걸리게 된다. 그림 4는 이를 보여주고 있다. 하나의 I/O에 대하여 어떤 채널에서 몇개의 way가 동작하느냐에 I/O의 소요시간이 결정되게 된다.

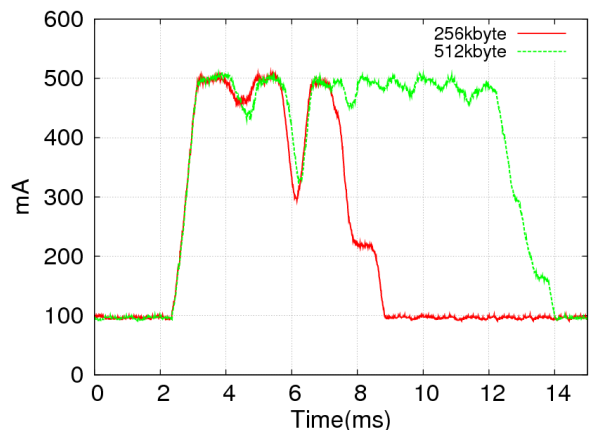


그림 4. way의 동작 확인

6.3 결과 분석

전력분석 기법을 사용하기 위해서는 타겟 SSD의 채널 수와 페이지크기, 플래시메모리 패키지 개수를 알아야 한다. 소모 전류의 수집은 하나의 I/O 단위로

이루어진다. 하나의 I/O란 한 번에 요청하는 I/O의 크기를 말한다. 타겟 SSD의 페이지크기를 I/O의 기본단위로 보고 크기를 2배씩 증가시키면서 소모전류를 수집하고, 수집된 데이터를 정렬하여 각 I/O의 소모 전류 사이에 어떤 차이가 있는지를 비교하였다. Intel X-25M의 페이지크기는 4KB, 채널은 10개, 플래시패키지의 개수는 20개이다. 그리고 상주전력은 약 100mA이다. 4KB와 8KB의 I/O는 동일한 소모전류 최고 값과 동일한 시간이 소모되었다. 따라서 1개의 채널과 1개의 way가 동작한 결과이다. 4KB의 결과를 기준으로 I/O의 크기가 증가할 때마다 소모 전류의 최고 값과, 소모 전류의 지속시간을 비교하여 매핑 정보를 예측하였다. 16KB의 I/O는 2개의 채널이 동작하고 채널 별로 2개의 way가 동작한 결과이다. 32KB의 I/O는 4개의 채널이 동작하고 채널 별로 2개의 way가 동작한 결과이다. 64KB의 I/O는 8개의 채널이 동작하고 채널 별로 2개의 way가 동작한 결과이다. 128KB의 I/O는 10개의 채널이 동작하고 일부 채널은 4개의 way, 일부 채널은 3개의 way가 동작한 결과이다. 256KB의 I/O는 10개의 채널이 동작하고 일부 채널은 8개의 way, 일부 채널은 5개의 way가 동작한 결과이다. 512KB의 I/O는 10개의 채널이 동작하고 일부 채널은 16개의 way, 일부 채널은 12개의 way가 동작한 결과이다. 512KB의 I/O는 10개의 채널과 일부의 채널은 20개의 way가, 일부의 채널은 14개의 way가 동작한 결과이다. 실험결과에 따라 Intel X-25M의 way depth는 20개로 표현하였다. 이는 512KB의 최대 소모전류 지속시간에 따른 것이다.

7. Acknowledgement

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

8. 결론

본 논문은 SSD에 대한 실제 전력측정 결과를 바탕으로 SSD의 동작을 예측하고 있다. 최근 출시되는 SSD들은 내부에 사용되는 플래시메모리의 데이터시트를 공개하지 않고 있다. 플래시메모리 패키지의 내부가 몇 개의 플래시메모리로 구성되어 있는지를 알 수 있다면 보다 정확한 예측이 가능할 것이다. 하지만 전력의 소모시간과 최대값으로 유추한 채널과 way의 정보만으로 모든 SSD의 동작을 예측하지는 못한다. 앞으로의 연구는 이러한 점을 보완할 계획이다.

Reference

- [1] S. Lee, *et al.*, "Advances in flash memory ssd technology for enterprise database applications," 2009, pp. 863-870.
- [2] J. Sumeet, *et al.*, "Modeling hard-disk power consumption," 2003.
- [3] D. Shin, "Power Analysis for Flash Memory SSD," presented at the NVRAMOS, 2010.
- [4] E. Seo, *et al.*, "Empirical Analysis on Energy Efficiency of Flash-based SSDs."
- [5] 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*, pp. 24-35, 2009.