

하이브리드 하드디스크: 데이터 저장 장치 특성 향상을 위한 플래시 메모리의 응용

삼성전자 | 이시윤

1. 서론

비휘발성 메모리(NVM; non-volatile memory)를 이용한 데이터 저장 장치는 크게 나누어 멀티미디어 데이터 등의 저장 및 이동에 이용되는 카드 형태의 것들과, 컴퓨팅 시스템에서 전형적으로 하드 디스크 드라이브가 이용되던 데이터 저장 장치의 대체를 위한 것들이 있다. 이 중 시스템 디스크로서 HDD(hard disk drive)를 대체/보완하려는 응용을 또다시 크게 나누면, 아래와 같은 세 가지의 데이터 저장 장치를 생각해 볼 수 있다.

- 1) SSD(solid state disk) - 반도체 메모리로만 이루어진 디스크 대용 데이터 저장 장치를 일컬으며, 수십~백수십 GB 정도의 용량을 제공한다.
- 2) H-HDD(hybrid hard disk drive) - HDD 내에 비휘발성 메모리를 추가로 장착한 것으로서, 수백 MB~수 GB 정도의 비휘발성 메모리를 디스크의 캐쉬로 이용한다.
- 3) Platform NVC(non-volatile cache) - PC의 마더보드에 비휘발성 메모리를 모듈 형태로 장착하는 방식으로서, 그 응용은 H-HDD 경우와 유사하다.

그림 1은 연도별 비휘발성 캐쉬(NV Cache) 탑재 PC의 전망을 나타낸 것으로서, 올해인 2007년에는 약 6.2%에 불과하나 2010년에는 약 45%의 PC가 어떠한 형태로든 NVM을 캐쉬의 형태로 채용할 것이 전망된다. 이는 특히 NAND 플래시 메모리의 가격 하락에 따라, 기존의 플래시 메모리의 응용처에 비교하여 다소 높은 용량 요구 사항을 가지는 PC의 저장 장치 캐쉬에 플래시 메모리가 이용되기 시작하는 추세로 파악할 수 있다.

† 본 연구는 정보통신부 및 정보통신연구진흥원의 IT신성장동력 핵심기술개발사업의 일환으로 수행하였음.
[2006-S-040-01, Flash Memory 기반 임베디드 멀티미디어 소프트웨어 기술 개발]

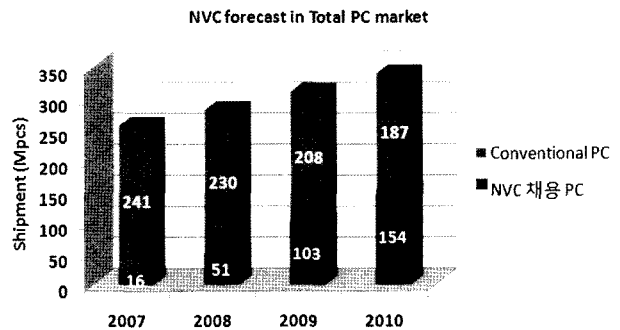


그림 1 NV Cache 탑재 PC 채용 전망
(Source: Gartner, 2006년 12월)

이 글에서는, H-HDD 응용을 중심으로 설계 개념과 특징점, 그리고 기술적 이슈들을 소개하기로 한다. 제 2절에서는 H-HDD의 설계 목표와 기존 HDD에 비하여 가지는 장점을 나열하고, 제 3절에서는 이를 가능하게 하기 위한 호스트 컴퓨팅 시스템과의 인터페이스의 설계를 보인다. 제 4절에서는 H-HDD의 구현에서 해결하여야 할 기술적 이슈들을 소개하고, 제 5절에서 향후 전망에 대한 언급과 함께 결론을 맺는다.

2. H-HDD의 설계 원리

그림 2는 H-HDD의 구성을 개념적으로 표시한 것이다. 여기에서 비휘발성 캐쉬로는 어떠한 형태의 NVM도 이용될 수가 있으나, 가격과 성능 특성 등에 기인하여 NAND 플래시 메모리가 이용되는 것이 일반적이다. 여기에서는 NAND 플래시가 비휘발성 캐쉬로 이용되는 H-HDD에 그 논의를 국한하기로 한다.

H-HDD의 설계는 기존의 HDD에 비하여 아래와 같은 장점을 가지는 것을 목표로 한다.

- 저전력 소모, 저소음, 저진동
- 빠른 부트 및 시스템 재개
- 향상된 일관적 응답성
- 데이터 저장 장치의 내구성 향상

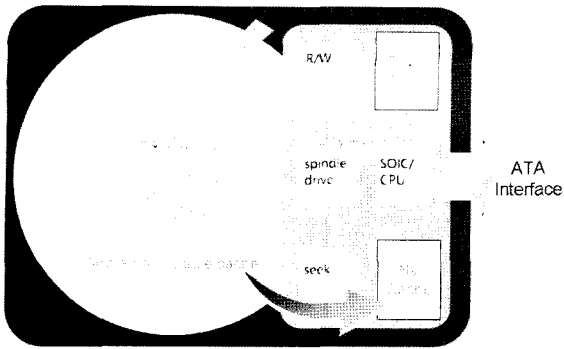


그림 2 H-HDD의 개념적 구성도
(Source: Microsoft WinHEC 2005)

2.1 전력 소모의 감소

통상적인 2.5-인치 HDD에서는 약 80~90% 정도의 전력 소모가 스핀들 모터에서 발생한다. H-HDD는 이점에 착안하여 디스크 스핀들 모터를 세워 두고, (따라서 디스크에는 데이터 읽기/쓰기가 불가능하다) 내장된 NVM을 이용하여 호스트로부터의 쓰기 데이터를 버퍼링함으로써 디스크 드라이브의 전력 소모를 크게 줄일 수 있다. 쓰기 버퍼로서의 NVC에 빈 공간이 부족해지면, 이때까지 버퍼링된 호스트로부터의 쓰기 데이터를 디스크에 기록함으로써 NVC에 새로 빈 공간을 확보할 수 있으며, 이 동작을 플러시(flush)라고 부른다.

전력 소모 감소 효과의 극대화를 위해서는 플러시 시간 간격을 가급적 크게 할 필요가 있다. 또한, 디스크 스핀들이 멈추어 있는 동안에는 디스크로부터의 데이터 읽기가 불가능하므로, 호스트로부터의 읽기 요청이 NVC에서 서비스되도록 함으로써 디스크 스핀-업(spin-up) 동작이 최소화되도록 해야 한다. 즉, 읽기 캐쉬의 접근 실패율(miss ratio)을 최소화하여 전력 소모 효과의 증대를 도모할 수 있다.

또한, H-HDD는 디스크 스핀들 모터를 세워 둠에 의하여 발열, 진동, 소음 발생을 줄일 수 있는 이점을 가지고 있다.

2.2 빠른 부트 및 시스템 재개

H-HDD는 두 가지 측면에서 시스템의 부트 성능을 향상시킬 수 있다. 먼저, 디스크 드라이브의 전원이 공급된 직후로부터 드라이브로부터 데이터 서비스가 개시될 때까지의 지연 시간을 단축할 수 있다. 구체적으로는, 기존의 하드디스크는 전원이 인가된 때로부터 디스크 스핀들이 정해진 회전수에 도달할 때까지 수 초의 지연 시간이 있다. 반면, H-HDD에서는 부트 초기에 읽어야 할 데이터들을 NVC에 미리 캐싱되도록 함으로써 디스크 스핀들이 정해진 회전수에 도달함을 기다리지 않고 데이터 서비스를 개시할 수 있음으

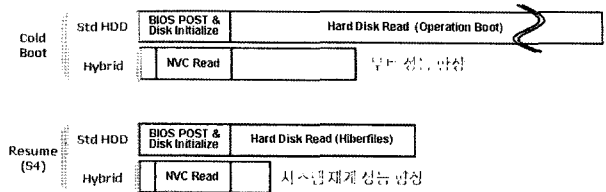


그림 3 H-HDD를 이용한 부트 및 시스템 재개 성능 향상

로 하여 부트 시간의 단축을 꾀할 수 있다.

또한, 부트에 필요한 데이터들을 미리 NVC에 적재해 둠으로써 추가의 부팅 시간 단축이 가능하다. 디스크에서의 읽기 서비스 시간에 트랙 탐색 시간이 포함되는데 비하여 NVC에서의 읽기에는 트랙 탐색 시간이 수반되지 않으므로, NVC로부터의 읽기는 디스크로부터의 읽기에 비하여 비순차 접근 지연시간이 매우 짧다는 이점을 가지고 있다. 이에 근거하여 시스템 부팅 때에 H-HDD는 기존 HDD에 비하여 높은 데이터 처리율을 제공하고, 이는 부팅 시간의 단축 효과로 나타난다.

위의 두 가지 개선점에 의한 부팅 시간 단축 효과는, 시스템의 콜드 부트시에만 적용되는 것이 아니며, S4(hibernation)[1] 상태로부터의 시스템 재개(resume)에도 유사하게 적용될 수 있다. 부팅 시간 및 시스템 재개 시간의 단축 효과를 도식화한 것이 그림 3에 나타나 있다.

2.3 시스템의 응답성 향상

자주 이용되는 데이터를 NVC에 둠으로써 이러한 데이터에 대해서는 일관된 읽기 응답성을 보장할 수 있다. 예를 들어, PC의 운영체제가 응용 프로그램 A가 자주 이용됨을 알 때, 시스템은 응용 프로그램 A의 코드 페이지들을 NVC에 적재할 것을 지시할 수 있으며, 이렇게 미리 캐싱된 응용 프로그램 A의 코드 페이지들은 사용자가 응용 프로그램 A를 실행할 때 디스크 접근 없이 NVC로부터 서비스됨으로써 그 응답성을 향상시킬 수 있다.

또한, 응답성의 향상 못지않게 중요한 성능 특성은 일관된 응답성이다. H-HDD는 시스템에 의하여 지정된 코드/데이터를 NVC에 둠으로써 디스크 접근에 의하여 발생하는 예측하기 어려운 지연 시간을 제거함으로써 보다 일관된 응답성을 제공한다.

2.4 내구성의 향상

하드디스크의 결함은 주로 취급 부주의나 충격 등에 의하여 발생한다. 특히 모바일 PC 환경에서는 이동성을 제공한다는 응용의 특성에 기인하여 의도되지 않은 충격 등에 의하여 디스크 시스템의 파손 또는 데

이터의 손실이 시스템 전체의 결함 발생의 높은 비중을 차지하기도 한다.

H-HDD는 두 가지 측면에서의 시스템 내구성을 향상시킨다. 첫째, 시스템의 정상 동작 중에도 디스크 스핀들을 멈추어 뒹으로써 기계적 동작에 의존하는 스핀들 모터 및 디스크 암(arm)의 동작 시간을 최소화하여 결함 발생률을 줄인다. 둘째, H-HDD가 호스트로부터의 데이터 서비스 요청을 처리하고 있는 중에도 디스크 헤드는 파크되어 있는 상태를 유지함으로써 갑작스런 진동이나 충격에 의하여 헤드와 디스크 사이의 접촉에 의한 결함 발생을 최소화할 수 있다.

3. H-HDD의 제어를 위한 인터페이스

앞 절에서 설명한 H-HDD의 특징점들을 이용하기 위해서는 호스트의 운영체제와 H-HDD 장치 사이의 밀접한 협응이 필요하다. 이를 위하여 NVC의 동작을 제어하기 위한 명령어들이 새로이 정의되었으며, 이는 기존에 하드디스크 등의 데이터 저장 장치의 인터페이스 표준을 규정한 ATA(advanced technology attachment) 규약에 추가되었다[2,3].

3.1 NVC 전원 모드

기존의 장치별 전원 관리 정책에 부가하여 NVC 전원 모드가 정의된다. NVC 전원 모드로의 진입 및 그로부터의 탈출을 위해서는 아래의 두 명령이 지원된다.

- Set NV Cache Power Mode
- Return from NV Cache Power Mode

위의 두 명령에 의하여 장치(H-HDD)는 NVC 전원 모드를 이용할 수 있다. 호스트는 Set NV Cache Power Mode 명령의 인자로서 스핀들의 회전에 최소 셋업 시간을 지정하며, 장치는 이 최소 스핀-업 셋업 시간을 만족하는 범위 내에서는 자신의 전원 관리 정책에 따라 스핀들 모터를 중지시키도록 한다.

전형적인 제어 예는, 노트북 PC가 AC 전원에 의하여 전원 공급되는 경우에는 NVC 전원 모드를 이용하지 않고, 배터리에 의하여 전원이 공급되는 경우에는 NVC 전원 모드를 이용하는 것이다. 그 외에도 NVC 전원 모드를 제공함으로써 해서 호스트 운영 체제에서는 다양한 전력 소모 최적화 기법이 가능해진다.

3.2 핀 집합과 언핀 집합

H-HDD의 NVC에 캐싱된 데이터는 크게 핀 집합(pin-ned set)과 언핀 집합(unpinned set)으로 나누어진다. 핀 집합은 호스트 운영 체제에 의하여 명시적으로 캐싱되어 있도록 지정된 데이터의 집합을 일컬으며, 그

외에 NVC에 저장되어 있는 모든 데이터는 언핀 집합으로 정의된다. 핀 집합에 대한 제어를 위하여 아래의 명령들이 제공된다.

- Add LBAs to NV Cache Pinned Set
- Remove LBAs from NV Cache Pinned Set

Add LBAs to NV Cache Pinned Set 명령은, 시작 주소(LBA; logical block address)와 섹터 개수로 이루어지는 인자를 전달받으며, 이로서 정의되는 데이터의 집합은 이 명령을 받은 이후에는 항상 NVC에서 서비스되도록 한다. 반면, Remove LBAs from NV Cache Pinned Set 명령은 지정된 범위의 주소에 해당하는 섹터들을 NVC 핀 집합에서 제거하는 역할을 한다.

Add LBAs to NV Cache Pinned Set 명령은 그 인자 중 하나로서 PI(pin immediate) 비트 플래그를 가진다. 이 플래그가 세트되어 있으면 해당 주소의 섹터들을 NVC에 사상함과 동시에 디스크로부터 데이터를 읽어 NVC에 저장한다. 부트에 필요한 데이터 또는 자주 이용되는 응용 프로그램을 NVC에 적재하는 것이 이 명령의 이용의 대표적인 예가 될 것이다. 반면, PI 비트가 세트되어 있지 않으면 NVC에 해당 주소를 사상하기는 하되, 즉시 데이터를 NVC에 적재할 필요는 없으며, 전형적으로 그 이후에 호스트로부터 데이터가 곧 쓰여질 것이라는 힌트로서의 역할을 한다. 즉, 호스트는 해당 섹터들에 데이터 쓰기를 곧 행할 것이므로, 굳이 디스크로부터 데이터를 읽어 NVC에 적재하지 않아도 됨을 표시하는 것이다. 대표적인 예는 하이버네이션(hibernation) 진입 직전에 메모리 이미지를 저장 장치에 저장하기 위한 하이버파일(hiberfile)을 NVC에 위치시키는 경우가 될 것이다. 즉, 시스템 재개를 빠르게 하기 위하여 하이버파일은 NVC에 두되, 이는 해당 주소를 디스크에서 읽어 NVC를 채우지 아니하고 호스트로부터 쓰여지도록 한다는 것이다.

위에 소개된 두 개의 명령 이외에도 아래와 같은 명령어들이 NVC의 핀 집합/언핀 집합에 대한 제어를 위하여 제공된다.

- Query NV Cache Pinned Set
- Query NV Cache Misses
- Flush NV Cache

Query NV Cache Pinned Set 명령은 현재 NVC 핀 집합에 어떠한 섹터들이 적재되어 있는지를 알기 위하여 제공된다. Query NV Cache Misses 명령은 현재까지 호스트로부터의 읽기 요청 중 NVC에서 히트되지 않은 주소 범위를 알아내기 위하여 제공된다. 마지막으로, Flush NV Cache 명령은 NVC 내에 가용 공간을

확보하기 위하여 NVC 언핀 집합의 데이터를 디스크에 플러시(flush)하기를 개시하라는 의미를 가지는 명령이다.

4. H-HDD 구현의 기술적 이슈들

H-HDD를 성공적으로 구현하기 위해서는 아래와 같은 기술적 이슈들에 대한 답을 제공할 수 있어야 한다.

- 주소 사상 및 이의 빠른 참조
- 플래시 메모리의 관리
- NVC 내의 가용 공간 관리

이 글에서는 위와 같은 문제들에 대한 해법을 제시하기보다는, 문제를 구체적으로 정의함으로써 차후 연구 개발의 가능성을 열고자 한다.

4.1 주소 사상

H-HDD의 NVC가 캐싱 대상으로 삼고 있는 주소 공간은 하드 디스크 전체이기 때문에 수십 GB로부터 수백 GB에 이르기까지의 큰 주소 공간을 갖는다. 반면, 실제 데이터가 캐싱되는 NVC의 용량은 수백 MB로부터 수 GB 정도의 공간이므로, 효과적인 주소 사상 방법이 필요해진다. 그림 4는 이 관계를 도식화한 것이다.

이 주소 사상 기법은 빠른 참조와 빠른 업데이트가 가능할 것을 요구한다. NVC로부터의 읽기 연산의 속도는 주소 참조와 플래시 메모리의 물리적인 읽기 속도에 의존적이며, 따라서 빠른 읽기를 위하여 효과적인 참조가 가능한 데이터 구조를 채택한 주소 사상이 필수적이다. 유의할 점은, NVC에서 히트되는 데이터의 읽기 성능뿐만 아니라 미스되는 데이터의 읽기, 즉 디스크의 읽기 성능에도 이 주소 사상의 참조 시간은 큰 영향을 미친다는 점이다. 즉, 매 읽기 요청에 대하여 먼저 NVC에 데이터가 있는지를 검사하기 위하여 이 주소 사상 정보를 참조하여야 하므로, 빠르면서도 NVC 용량의 증가에 따라 지나치게 검색 시간이 증가하지 않는 데이터 구조의 채택이 필요하다.

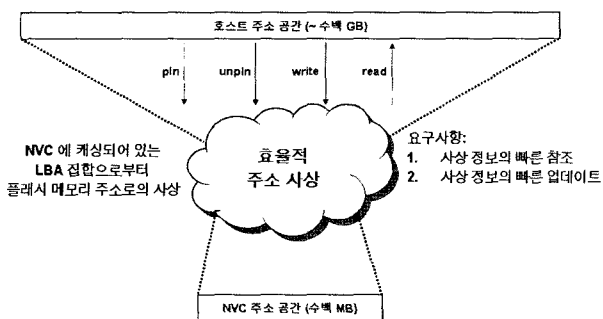


그림 4 호스트 주소 공간으로부터 플래시 메모리 주소 공간으로의 사상

NVC에 데이터의 쓰기 동작이 발생할 때에는 주소 사상 데이터의 업데이트가 수반된다. 즉, 새로이 캐싱된 데이터의 주소 사상 정보를 데이터 구조 내에 표현하여야 한다. 이는 NVC의 쓰기 성능에 영향을 미칠 것이므로 주소 사상 데이터 구조는 새로운 엔트리의 삽입과 같은 업데이트 연산 또한 빠를 필요가 있다. 단, 상대적으로 읽기에 비하여 느린 플래시 메모리의 쓰기 연산과 병렬적으로 행할 수 있으므로 업데이트에 대한 성능 요구사항은 참조 연산에 비하여 높지 않다.

4.2 플래시 메모리 관리

전형적으로, 플래시 메모리의 관리는 FTL(flash translation layer)라고 불리는 소프트웨어에 의하여 이루어진다[4]. H-HDD의 구현에 있어서도 플래시 메모리의 관리는 매우 중요한 요소이다. 즉, NAND 플래시의 아래와 같은 특징에 근거하여 부가의 고려사항이 등장한다.

- 1) NAND 플래시 메모리는 데이터 쓰기 이전에 소거 연산이 수반된다.
- 2) 데이터 쓰기의 단위(페이지)와 소거의 단위(블록)이 다르다.
- 3) NAND 플래시의 블록들 중에는 올바르게 이용할 수 없는 블록(배드 블록; bad block)들이 있다. 배드 블록은 제조시에 생겨나는 것들도 있으며, 플래시 메모리의 이용 도중에 발생하는 것들도 있다.
- 4) 한 블록에 데이터 소거/쓰기 연산을 행할 수 있는 회수가 한정적이다. 이 회수를 넘기면 해당 블록은 배드 블록이 될 가능성이 높아진다.

따라서, NVC로 NAND 플래시가 이용되는 경우 배드 블록의 관리가 필요해진다. 배드 블록은 블록 할당에 있어 다른 블록으로 사상되는 것이 일반적이다. 즉, 플래시 메모리 관리를 담당하는 소프트웨어 계층(예: FTL의 일부)에서 배드 블록의 재사상을 담당함으로써, 이보다 위의 계층에서는 배드 블록이 없는 논리적인 블록들의 집합으로 플래시 메모리를 다룰 수 있도록 하는 기능을 제공할 필요가 있다. 이렇게 함으로써 호스트의 입장에서 가용한 플래시 메모리 공간은 배드 블록들 및 이의 대체를 위하여 예비하여 둔 블록들을 제외한 만큼의 공간이 된다. 통상적으로는 배드 블록들의 개수가 증가하여 예비된 블록들을 모두 소진하고 난 뒤에는 제품의 수명이 다한 것으로 본다.

이러한 이유로, 가급적 배드 블록의 발생을 최소화할 필요가 있다. 그런데, 단순히 블록들을 이용하면 특정 블록들만 소거/쓰기가 집중적으로 발생하여 배드 블록의 발생을 야기할 가능성이 있다. 이렇게 하

여 소수의 블록들에 소거/쓰기가 집중되면 배드 블록을 위한 블록 재사상에 의하여 예비된 블록들을 일찍 소진함으로써 제품의 수명을 단축시킬 가능성이 있으므로, 플래시 메모리의 관리에 있어서는 모든 블록들의 소거/쓰기 회수가 균등하도록 블록들을 관리할 필요가 있다. 이러한 작업을 마모 균등화(wear leveling)라고 부르며, 여타의 플래시 메모리 응용에서와 마찬가지로 H-HDD의 NVC 응용에서도 플래시의 마모 균등화는 기술적 해결책이 필요한 중요한 문제의 하나이다.

4.3 가용 공간 확보

NVC 내의 가용 공간이 부족해지는 경우, 두 가지 중 하나의 연산을 행하여 가용 공간을 확보해야 한다. 먼저, NVC 연편 집합에 대한 플러시(flush) 연산은 NVC에 기록되어 있던 데이터를 읽어 디스크에 저장하고 이 데이터가 차지하고 있던 공간을 재활용 가능하도록 만들어 가용 공간을 확보할 수 있게 한다. 플러시 연산을 행함에 있어서는 지나치게 잦은 플러시가 발생하지 않도록 해야 한다. 플러시 연산은 디스크 쓰기를 수반함에 의하여 디스크 스핀들이 회전하여야 하며, 만약 전력 소모 감소를 위하여 스핀들을 멈추어 둔 상태에서 플러시가 발생하면 부가의 전력 소모가 야기되기 때문이다. 또한, NVC 내의 가용 공간이 부족한 경우에 강제적 플러시(forced flush)를 행해야 하지만, NVC에서의 읽기 미스 등에 의하여 디스크 스핀들이 회전하게 된 경우에 NVC 내의 가용 공간이 부족하지 않더라도 기회적 플러시(opportunistic flush)를 적용할 수 있다.

NVC 내의 가용 공간을 확보하는 다른 하나의 방법은 쓰레기 수집(garbage collection)에 의해서이다. 쓰레기 수집은 NVC 내의 무효한 데이터에 의하여 차지된 공간을 수집하여 새로운 빈 블록들을 만들어 냄에 의하여 가용 공간을 확보한다.

NVC 내의 데이터가 무효화되는 현상은 크게 두 가지 경우에 의하여 나타난다. 먼저, 연편 집합에 대한 플러시 연산이 발생하면 해당 데이터가 차지하고 있는 공간은 재사용 가능한 영역이 된다. 또한, 이미 저장된 데이터와 동일한 주소를 가지는 새 데이터의 쓰기 연산이 발생하면 해당 데이터가 기존에 차지하고 있던 영역 또한 재사용 가능한 영역이 된다. 그러나 플래시 메모리의 데이터는 페이지 단위로 무효화되는데 비하여 공간의 재사용은 블록 단위의 소거 연산을 통하여 이루어지므로, 위와 같은 이유로 무효화된 데이터가 존재하는 경우에는 쓰레기 수집을 통하여 재

사용 가능 공간을 블록 단위로 확보할 필요가 있다.

여기에서 주의해야 할 점은, 기회적 플러시 연산과 쓰레기 수집 연산은 가급적 호스트의 읽기/쓰기 요청의 처리 시간에 영향을 미치지 않도록 이루어져야 한다는 것이다. 즉, 호스트의 유휴 시간(idle time)을 적절히 이용하여 플러시 및 쓰레기 수집 연산을 행함으로써 시스템의 성능에 미칠 수 있는 부정적 영향을 최소화하여야 한다.

5. 결론 및 향후 전망

이 글에서는 H-HDD라고 불리는 새로운 데이터 저장 장치에 대하여 그 설계 개념, 호스트 인터페이스, 그리고 구현상의 기술적 요소들에 대하여 살펴보았다. H-HDD는 기존의 하드디스크에 플래시 메모리로 이루어지는 비휘발성 캐쉬를 탑재하여 데이터 저장 장치의 전력 소모 감소, 빠른 부팅 및 시스템 재개, 일관된 응답성 제공, 시스템의 내구성 향상 등을 이루는 것을 목표로 한다.

비휘발성 메모리, 특히 NAND 플래시 메모리의 가격 하락이 현재와 같은 추세로 이루어질 때, 가까운 미래에 NVC의 용량은 수 GB 이상으로 증가할 전망이다. 증가된 NVC를 이용하여 시스템 성능은 아래와 같은 측면에서 향상될 수 있을 것이다.

- 1) 전력 소모의 추가 감소 - NVC의 용량 증대에 따라 NVC의 읽기 접근 실패율이 감소하고, 또한 쓰기 버퍼링에 할당할 수 있는 용량이 증가함으로써 추가의 전력 소모 감소 효과를 기대할 수 있다.
- 2) 응용 프로그램 실행 속도 증가 - 더 많은 응용 프로그램의 코드/데이터를 NVC에 저장할 수 있게 됨으로 해서(NVC 읽기 접근 성공률의 증가) 자주 실행되는 응용 프로그램의 응답성이 개선되는 효과를 기대할 수 있다.
- 3) NVC 입출력 성능의 개선 - 복수의 플래시 메모리 장치를 이용하게 되어 하드웨어/소프트웨어 측면에서 병렬성을 이용한 입출력 성능의 개선을 통하여 시스템 전체의 성능 개선을 도모할 수 있을 전망이다.

참고문헌

- [1] Compaq Computer Corporation, Intel Corporation, Micro-soft Corporation, Phoenix Technologies Ltd, and Toshiba Corporation, Advanced Configuration and Power Interface Specification, Revision 2.0, July 2000.
- [2] American National Standard, AT Attachment with Packet Interface - 7, April 2004.

- [3] N. Obr and F. Shu, Non Volatile Cache Command Proposal for ATA8-ACS, Feb 2006.
- [4] Intel Corporation, Understanding the Flash Translation Layer(FTL) Specification, <http://developer.intel.com>



이 시 윤

1996 서울대학교 컴퓨터공학과(학사)
1998 서울대학교 컴퓨터공학과(석사)
2005 서울대학교 전기컴퓨터공학부(박사)
2005~현재 삼성전자 메모리사업부 책임연구원
관심분야: 임베디드시스템 구조, 플래시 메모리
E-mail : sheayun.lee@samsung.com

SWCC 2007(하계 컴퓨터통신 워크샵)

- 일 자 : 2007년 8월 23~25일
- 장 소 : 제주 라마다프라자호텔
- 내 용 : 논문발표 등
- 주 최 : 정보통신연구회
- 상세안내 : <http://swcc.or.kr/2007>