



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2009-0129161
(43) 공개일자 2009년12월16일

(51) Int. Cl.

G11C 16/08 (2006.01) G11C 16/34 (2006.01)

(21) 출원번호 10-2008-0055289

(22) 출원일자 2008년06월12일

심사청구일자 2008년06월12일

(71) 출원인

한국전자통신연구원

대전 유성구 가정동 161번지

인하대학교 산학협력단

인천 남구 용현동 253 인하대학교

(72) 발명자

김덕환

서울특별시 양천구 목동 940 목동아이파크아파트
103-1004

박광희

서울특별시 강서구 화곡1동 424-156

(74) 대리인

이은철, 유완식

전체 청구항 수 : 총 10 항

(54) 플래시 변환 계층 구조와 이를 이용한 선 반입 방법 및 플래시 변환 구조를 기반으로 한 비동기 쓰기 방법

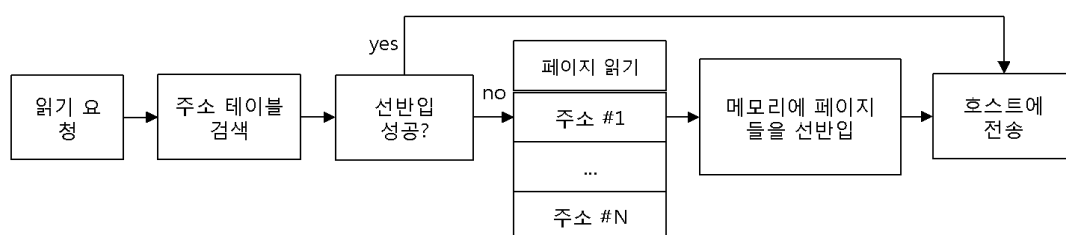
(57) 요약

본 발명에서 플래시 메모리의 효율적 관리가 가능 한 플래시 변환 계층 구조와 이를 이용한 선 반입 방법 및 플래시 변환 구조를 기반으로 한 비동기 쓰기 방법을 개시한다.

본 발명에 따른 플래시 변환 구조를 이용한 선 반입 방법은, a) 군집형 해시 테이블로 동일한 버킷 안에 연속된 LBA 및 이에 대응하는 PBA들을 저장하는 단계; b) 상기 버킷의 LBA에 대한 읽기 요청을 접수하는 단계; c) Coarse-Grained 군집형 해시 테이블로 탑재된 상기 연속성 플래그를 토대로 연속적인 LBA의 주소변환을 수행하는 단계; d) 상기 LBA가 소속된 버킷의 전체 페이지를 선 반입하는 단계;로 이루어진다.

따라서, 본 발명은 NAND 플래시 메모리 관련 소프트웨어에 관한 기술 연구개발로 세계적인 메모리 반도체 분야에 서 독보적인 위치를 유지할 수 있으며, 나아가 NVRAM(Non Volatile Random Access Memory)과 같은 향후 차세대 비휘발성 메모리에도 충분히 적용할 수 있어, 매출 증가와 독점 기술확보로 세계적인 경쟁력을 갖출 수 있는 효과가 있다.

대표도



이 발명을 지원한 국가연구개발사업

과제고유번호 2008-전공실습-1000186

부처명 기타기관

연구사업명 IT SoC 핵심 설계 인력 양성사업

연구과제명 차세대 퓨전 플래시 메모리에 적합한 임베디드 소프트웨어플랫폼에 관한 연구

주관기관 한국전자통신연구원

연구기간 2008년 03월 01일 ~ 2008년 11월 30일

특허청구의 범위

청구항 1

1단계 Fine-Grained 군집형 해시 테이블, 2단계 Fine-grained 군집형 해시 테이블 및 Coarse-grained 군집형 해시 테이블로 구성된 플래시 변환 계층(FTL) 구조에 있어서,

상기 Coarse-grained 군집형 해시 테이블로 연속적인 물리 주소(PBA:Physical Block Address)에 관한 정보를 저장하고, 상기 물리 주소에 관한 정보를 기반으로 연속적인 LBA(Local Block Address)의 주소변환 요청에 따른 반복적인 주소변환 메카니즘을 최소화하기 위한 연속성 플래그가 포함되는 것을 특징으로 하는 플래시 변환 계층 구조.

청구항 2

제 1 항에 있어서,

상기 연속성 플래그는 VBA(Virtual Block Address)와 PBA(Physical Block Address)가 연속적인 경우 연속적인 블록의 갯 수가 기입되는 것을 특징으로 하는 플래시 변환 계층 구조.

청구항 3

제 1 항 또는 제 2 항에 있어서,

상기 연속성 플래그는 상기 Coarse-grained 군집형 해시 테이블의 각 버킷의 서브 블록마다 설정되는 것을 특징으로 하는 플래시 변환 계층 구조.

청구항 4

제 1 항 또는 제 2 항에 있어서,

상기 플래시 변환은 NAND 플래시 메모리 기반인 것을 특징으로 하는 플래시 변환 계층 구조.

청구항 5

제 1 항에 따른 플래시 변환 계층 구조를 이용한 페이지 선 반입 방법에 있어서,

- a) 상기 군집형 해시 테이블로 동일한 버킷 안에 연속된 LBA 및 이에 대응하는 PBA들을 저장하는 단계;
- b) 상기 버킷의 LBA에 대한 읽기 요청을 접수하는 단계;
- c) 상기 Coarse-Grained 군집형 해시 테이블로 탑재된 상기 연속성 플래그를 토대로 연속적인 LBA의 주소변환을 수행하는 단계;
- d) 상기 LBA가 소속된 버킷의 전체 페이지를 선 반입하는 단계;로 이루어진 것을 특징으로 하는 플래시 변환 계층 구조를 이용한 선 반입 방법.

청구항 6

제 5 항에 있어서 상기 c) 단계는,

- c-1) 상기 1단계 Fine-grained 군집형 해시 테이블을 검색하여 매칭되는 PBA가 존재할 경우 주소 변환을 종료하는 단계;
- c-2) 상기 1단계 Fine-grained 군집형 해시 테이블에서 검색에 실패했을 경우, 상기 2단계 Fine-grained 군집형 해시 테이블을 검색하여 매칭되는 PBA가 존재할 경우 주소 변환을 종료하는 단계; 및
- c-3) 상기 2단계 Fine-grained 군집형 해시 테이블에서도 검색에 실패 한 경우, 상기 Coarse-grained 군집형 해시 테이블에서 매칭되는 주소 변환 정보를 찾는 단계;로 이루어진 것을 특징으로 하는 플래시 변환 계층 구조를 이용한 선 반입 방법.

청구항 7

제 6 항에 있어서 상기 c-3) 단계는,

가) LBA를 상기 Coarse-grained 군집형 해시 테이블 내의 VBA와 오프셋으로 변환하는 단계;

나) 상기 VBA를 해싱하여 PPBA와 RPBA의 주소가 저장되어 있는 버킷을 찾는 단계; 및

다) 매칭된 상기 PPBA와 오프셋에 있는 페이지 데이터가 유효하지 않을 경우 상기 RPBA를 순차적으로 검색하여 해당 물리 주소를 찾는 단계;로 이루어진 것을 특징으로 하는 플래시 변환 계층 구조를 이용한 선 반입 방법.

청구항 8

플래시 변환 구조를 기반으로 한 플래시 메모리 쓰기 방법에 있어서,

a) 임의의 정보가 저장되는 예측 테이블을 생성하는 단계;

b) 반복적으로 저장되는 LBA가 포함된 VBA를 상기 예측 테이블에 추가하는 단계;

c) 동일한 VBA로 소속된 LBA에 쓰기 요청이 들어올 때마다, 쓰기 요청 회수에 비례하여 상기 예측 테이블로 저장되는 예측 비트를 갱신하는 단계;

d) 상기 예측 비트가 기 설정된 값으로 갱신될 경우, 상기 VBA를 예약 버퍼에 저장하는 단계; 및

e) 상기 예약 버퍼가 포화될 경우, 상기 예약 버퍼로 탑재된 최신 데이터만 플래시 메모리로 적재하는 단계;로 이루어진 것을 특징으로 하는 플래시 변환 구조를 기반으로 한 비동기 쓰기 방법.

청구항 9

제 8 항에 있어서,

상기 예측 비트는 적어도 4가지의 상태 표시를 위한 2비트 구조인 것을 특징으로 하는 플래시 변환 구조를 기반으로 한 비동기 쓰기 방법.

청구항 10

제 8 항에 있어서,

상기 예약 버퍼로 탑재되는 최신 데이터는 상기 VBA를 일괄 합병된 데이터인 것을 특징으로 하는 플래시 변환 구조를 기반으로 한 비동기 쓰기 방법.

명세서

발명의 상세한 설명

기술분야

<1> 본 발명은 플래시 메모리의 주소 변환 성능과 수명 관리에 관한 것으로, 더욱 상세하게는 연속성 플래그와 선반입 기법을 통해 연속적인 주소변환 성능을 제공하고, 2비트 쓰기 예측기법과 비동기 쓰기 기법을 통해 플래시 메모리의 효율적 관리가 가능 한 플래시 변환 계층 구조와 이를 이용한 선 반입 방법 및 플래시 변환 구조를 기반으로 한 비동기 쓰기 방법에 관한 것이다.

배경기술

<2> 최근 대용량 낸드 플래시 메모리가 매년 출시되면서 향후 자기 디스크를 대체할 새로운 저장장치로 각광받고 있다. 낸드 플래시 메모리의 장점으로 일반 하드 디스크에 비해 랜덤 액세스가 가능해 입출력 속도가 빠르고 충격에 강해 임베디드 시스템이나 산업용 시스템에 유용하다. 그러나 이와 같은 장점 외에 자기디스크와 다른 물리적 특성을 아래와 같이 포함하고 있다.

<3> 첫째 낸드 플래시 메모리는 페이지(512B, 2KB)와 블록(16KB, 128KB)으로 구성되어 있다. 페이지는 낸드 플래시 메모리를 구성하는 최소 단위로 일반적으로 페이지 32개를 하나의 블록을 구성한다. 둘째 각 블록들은 갱신되는 횟수에 한계를 가지고 있으며 이를 마모도라고 한다. SLC(Single Level Cell)에서는 약 10만 번 MLC(Multi Level Cell)에서는 약 1만 번 정도의 수명을 가지고 있다. 셋째, 낸드 플래시 메모리는 읽기/쓰기/삭제 3개의 동작으로 구분되며 읽기/쓰기 동작은 페이지(512B, 2KB) 단위이지만 삭제 동작은 블록(16KB, 128KB) 단위이다.

넷째 블록 단위의 삭제는 페이지 32개를 지우는 속도이므로 읽기/쓰기 속도보다 많이 느리다.

- <4> 따라서, 기존에 사용하던 하드 디스크 전용 파일시스템은 이와 같은 낸드 플래시 메모리의 특성을 고려하지 않고 설계되었기 때문에 기존의 파일 시스템과 낸드 플래시 메모리와의 호환을 위해서는 FTL(Flash Translation Layer)이라는 미들웨어를 거쳐서 수행해야 한다. FTL 이 수행하는 역할을 살펴 보면 다음과 같다.
- <5> 첫째, 제자리 갱신(In-place update)을 하기 위해서는 쓰기 전 지우기를 수행해야 하므로 이에 따른 삭제 비용이 매우 높다. 그러므로 낸드 플래시 메모리는 제자리 갱신 대신 외부 갱신(Out-place update)을 한다. 외부 갱신 할 때마다 파일 시스템의 논리 주소(LBA: Logical Block Address)와 사상되는 플래시 메모리의 물리 주소(PBA: Physical Block Address)가 갱신되므로 주소 변환 테이블 기능을 포함한다. 둘째, 낸드 플래시 메모리를 오래 쓰기 위해서 모든 블록의 마모도를 평준화 시키는 기능을 가지고 있다. 셋째 삭제 속도가 읽기, 쓰기보다 느리므로 플래시 메모리의 유효하지 않은(Invalid) 블록을 일괄적으로 삭제하는 블록 수거 기법(Garbage Collection) 기능을 포함한다.
- <6> 한편, 전술된 바와 같이 페이지 단위 주소 변환 기법의 가장 큰 단점인 주소 변환 테이블의 크기를 줄이기 위해, 근래에는 블록 단위의 주소 변환 방식을 적용하고 있으며, 일 예로 NFTL 방식이 존재한다. 여기서, NFTL은 M-Systems사에서 설계한 낸드 플래시 메모리 전용 FTL이다. 즉, 도 1에 도시된 바와 같이, LBA (Logical Block Address)의 값을 블록당 페이지 개수로 나누어 몫은 VBA(Virtual Block Address)로 사용하고 나머지 값은 오프셋을 인덱스로 사용한다. 그리고 주소 변환 테이블로 접근 시 VBA의 값을 통해 주 블록(Primary block)과 교체 블록(Replacement block)의 주소를 가지고 있는 엔트리에 접근한다.
- <7> 먼저, 주 블록 PBA가 존재할 경우 나머지 값인 블록 오프셋으로 페이지 단위 접근을 시도한다. 하지만 주 블록에 대응하는 PBA의 데이터가 유효하지 않은 경우 교체 블록의 주소로 다시 접근하여 각 페이지를 순차적으로 검색하는 기법을 사용한다. 여기서 이를 NFTL의 블록 단위 주소 변환 기법이라고 한다. 블록 단위 주소 변환 기법은 주소 변환 테이블의 크기가 작은 대신 주 블록에서 주소 검색을 실패할 경우 교체 블록을 순차적으로 다시 검색해야 하는 오버헤드가 존재한다.
- <8> 한편, 주소 변환 방식의 또 다른 예로, AFTL 방식이 존재하는데, 상기 AFTL은 대만 국립 대학교에서 제안한 FTL로 도 2와 같이 Coarse-grained 해시 테이블, Fine-grained 해시 테이블 두 개 단위의 주소 변환 테이블로 이루어져 있다.
- <9> 기본적으로 Coarse-grained 해시 테이블에서는 NFTL과 같이 블록 단위의 주소 변환 기법을 사용하고, Fine-grained 해시 테이블에서는 FTL과 같은 페이지 단위의 주소 변환 기법을 사용한다. 페이지 단위의 주소 변환 기법을 사용하는 경우는 교체 블록의 데이터가 유효 상태일 경우 이 데이터들이 앞으로 자주 쓰일 수 있다고 가정하여 Fine-grained 해시 테이블에 저장하여 페이지 단위로 주소를 변환한다.
- <10> 주 블록일 경우는 오프셋을 이용하여 바로 찾기 때문에 Coarse-grained 해시 테이블을 사용한다. Fine-grained 해시 테이블의 크기의 제한이 있으므로 Fine-to-Coarse 간의 교체 기법이 필요한데 이때 사용하는 기법이 LRU를 적용하여 일정 기간 동안 사용하지 않는 슬롯을 Coarse-grained 해시 테이블로 이동하는 기법을 사용한다.
- <11> 그러나, 전술된 NFTL 방식 및 AFTL 방식에서 적용되는 해시 테이블은, 주소 변환 속도가 항상 고정되어 있어서 유연성이 부족하다는 단점이 있었으며, 태그(Tag) 및 넥스트(Next) 포인터가 노드 별로 추가적으로 요구되어 메모리 사용량에 있어서 200 % 정도의 오버헤드가 발생하는 문제점이 있다.

발명의 내용

해결 하고자하는 과제

- <12> 본 발명은 이와 같은 문제점을 해결하기 위한 것으로, 본 발명의 목적은 연속적인 물리주소에 관한 정보를 저장하는 연속성 플래그를 두어, 반복적인 주소변환 횟수를 감소시켜 Coarse-grained 군집형 해시 테이블의 주소변환 성능을 향상시킬 수 있는 플래시 변환 계층을 위한 선반입 방법을 제공함에 있다.
- <13> 본 발명의 다른 목적은, 갱신이 자주 발생하는 주소 데이터를 예측함으로써, 플래시 메모리의 수명을 늘리기 위한 불필요한 반복적인 쓰기 동작을 최소화시킬 수 있는 플래시 변환 계층을 위한 선반입 방법을 기반으로 한 비동기 쓰기 방법을 제공함에 있다.

과제 해결수단

- <14> 상기 목적을 달성하기 위한 본 발명의 관점에 따른 플래시 변환 계층 구조는, 1단계 Fine-Grained 군집형 해시 테이블, 2단계 Fine-grained 군집형 해시 테이블 및 Coarse-grained 군집형 해시 테이블로 구성된 플래시 변환 계층(FTL) 구조에 있어서, 상기 Coarse-grained 군집형 해시 테이블로 연속적인 물리 주소(PBA:Physical Block Address)에 관한 정보를 저장하고, 상기 물리 주소에 관한 정보를 기반으로 연속적인 LBA(Local Block Address)의 주소변환 요청에 따른 반복적인 주소변환 메카니즘을 최소화하기 위한 연속성 플래그가 포함되는 것을 특징으로 한다.
- <15> 또한, 상기 목적을 달성하기 위한 본 발명의 관점에 따른 플래시 변환 계층을 위한 선반입 방법은, a) 군집형 해시 테이블로 동일한 버킷 안에 연속된 LBA 및 이에 대응하는 PBA들을 저장하는 단계; b) 상기 버킷의 LBA에 대한 읽기 요청을 접수하는 단계; c) Coarse-Grained 군집형 해시 테이블로 탑재된 상기 연속성 플래그를 토대로 연속적인 LBA의 주소변환을 수행하는 단계; d) 상기 LBA가 소속된 버킷의 전체 페이지를 선 반입하는 단계; 로 이루어진 것을 특징으로 한다.
- <16> 구체적으로, 상기 c) 단계는, c-1) 상기 1단계 Fine-grained 군집형 해시 테이블을 검색하여 매칭되는 PBA가 존재할 경우 주소 변환을 종료하는 단계; c-2) 상기 1단계 Fine-grained 군집형 해시 테이블에서 검색에 실패했을 경우, 상기 2단계 Fine-grained 군집형 해시 테이블을 검색하여 매칭되는 PBA가 존재할 경우 주소 변환을 종료하는 단계; 및 c-3) 상기 2단계 Fine-grained 군집형 해시 테이블에서도 검색에 실패한 경우, 상기 Coarse-grained 군집형 해시 테이블에서 매칭되는 주소 변환 정보를 찾는 단계;로 이루어진 것을 특징으로 한다.
- <17> 한편, 상기 목적을 달성하기 위한 본 발명의 관점에 따른 플래시 변환 계층을 위한 선반입 방법을 기반으로 한 비동기 쓰기 방법은, a) 임의의 정보가 저장되는 예측 테이블을 생성하는 단계; b) 반복적으로 저장되는 LBA가 포함된 VBA를 상기 예측 테이블에 추가하는 단계; c) 동일한 VBA로 소속된 LBA에 쓰기 요청이 들어올 때마다, 쓰기 요청 회수에 비례하여 상기 예측 테이블로 저장되는 예측 비트를 갱신하는 단계; d) 상기 예측 비트가 기 설정된 값으로 갱신될 경우, 상기 VBA를 예약 버퍼에 저장하는 단계; 및 e) 상기 예약 버퍼가 포화될 경우, 상기 예약 버퍼로 탑재된 최신 데이터만 플래시 메모리로 적재하는 단계;로 이루어진 것을 특징으로 한다.

효 과

- <18> 본 발명에 따른 플래시 변환 계층을 위한 선반입 및 비동기 쓰기 방법은, 플래시 메모리에 적합한 FTL에 대한 실질적인 알고리즘 구현 및 정책을 제시함으로써, 유관분야의 연구를 선도하고 후속 연구의 활성화에 기여할 수 있는 효과가 있다.
- <19> 또한 본 발명은, NAND 플래시 메모리 관련 소프트웨어에 관한 기술 연구개발로 세계적인 메모리 반도체 분야에서 독보적인 위치를 유지할 수 있으며, 나아가 NVRAM(Non Volatile Random Access Memory)과 같은 향후 차세대 비휘발성 메모리에도 충분히 적용할 수 있어, 매출 증가와 독점 기술확보로 세계적인 경쟁력을 갖출 수 있는 효과가 있다.

발명의 실시를 위한 구체적인 내용

- <20> 이하, 본 발명을 첨부된 예시도면에 의거 상세히 설명하면 다음과 같다.
- <21> 먼저, 발명에서 개시되는 주소변환 테이블 기법은 일반 해시 테이블 보다 메모리 사용량이 적고 대용량 주소 변환에 적합한 Clustered Hash Table을 사용하는 것으로, 이하 CFTL(Clustered Flash Translation Layer) 방식이라 지칭한다. 본 발명에 따른 CFTL 방식은 블록, 페이지 단위의 주소 변환 테이블 기법을 혼용하면서 Fine-grained Clustered Hash Table을 2단계로 설계하여 미스 페널티를 감소시켜, 주소 변환 속도를 향상시킨다. 또한 Continuity Flag와 Prefetch를 통해 연속적인 주소 변환 성능을 향상시키고 2-bit access prediction 기법을 이용한 쓰기 지연을 통해 플래시 메모리의 수명 및 성능을 향상시킨다.
- <22> 그러면, 전술된 플래시 변환 계층을 위한 선반입 절차를 설명한다. 도 3은 플래시 변환 계층을 위한 선반입 절차로 적용되는 CFTL 방식의 주소변환 테이블 기법을 나타낸 도면이고, 도 4는 상기 CFTL 방식을 이용한 플래시 변환 계층을 위한 선반입 절차를 도시한 흐름도이다.
- <23> 도시된 바와 같이, 군집형 해시 테이블은 선형 테이블 기법과 해시 테이블을 혼용하여, 유사한 주소들을 동일한 버킷(Bucket)에 그룹화하는 1단계 Fine-Grained 군집형 해시 테이블과, TLB(Translation Lookaside Buffer)와 같은 캐시를 차용하여, 미스 페널티를 최소화하고 상대적으로 주소변환 오버헤드가 큰 Coarse-grained 군집형 해시 테이블의 접근 빈도를 감소시키는 2단계 Fine-grained 군집형 해시 테이블 및, 연속성 플래그를 사용하여

연속적인 LBA(Logical Block Address)의 주소변환 요청 시 주소 변환 부담을 줄이는 Coarse-grained 군집형 해시 테이블로 구성된다.

- <24> 이와 같은 구성을 토대로 주소 변환 요청 시, 상기 Fine-grained 군집형 해시 테이블에 요청된 LBA가 존재할 경우 그 LBA가 소속된 버킷의 전체 페이지를 미리 선반입하여 주소 변환을 수행함으로써, 주소변환 성능을 향상시킨다.
- <25> 여기서, 상기 1단계 Fine-Grained 군집형 해시 테이블은 군집형 해시 테이블은 선형 테이블 기법과 해시 테이블을 혼용하는 기법으로 메모리 사용량이 기존 해시 테이블보다 적고, 유사한 주소들을 동일한 버킷(Bucket)에 그룹화하므로 지역성 확보가 가능할 것이다.
- <26> 따라서, 특정 LBA의 주소변환 요청이 호스트로부터 들어오면 가장 먼저 1단계 Fine-grained 군집형 해시 테이블을 검색하고 매칭되는 PBA를 찾으면 주소 변환은 종료되며, 상기 1단계 Fine-grained 군집형 해시 테이블에서 검색에 실패했을 경우, 2단계 Fine-grained 군집형 해시 테이블을 검색하여 매칭되는 PBA를 찾으면 주소 변환은 종료된다.
- <27> 반면, 상기 2단계 Fine-grained 군집형 해시 테이블에서도 검색에 실패 한 경우 최종적으로 Coarse-grained 군집형 해시 테이블에서 매칭되는 주소 변환 정보를 찾는다. Coarse-grained 군집형 해시 테이블은 NFTL과 동일하게 우선 LBA를 VBA와 오프셋으로 변환한다. 이렇게 구해진 VBA를 해싱하여 PPBA(Primary Physical Block Address)와 RPBA(Replacement Physical Block Address)의 주소가 저장 되어 있는 버킷을 찾는다. 그리고 매칭된 PPBA와 오프셋에 있는 페이지 데이터가 유효하지 않을 경우 RPBA를 순차적으로 검색하여 해당 물리 주소를 찾게 된다.
- <28> 또한, 전술된 2단계 fine-grained 군집형 해시 테이블은 Coarse-to-Fine 또는 Fine-to-Coarse간의 이동이 필요하며, 이를 위한 정책으로, 승급정책 및 강등정책이 적용된다. 즉, Coarse-to-fine으로 이동하는 경우, 특정 LBA에 찾은 접근으로 인해 접근 빈도가 일정횟수만큼 증가하면 처음에는 2단계 fine-grained 군집형 해시 테이블로 승급되는 승급정책이 적용되거나, NUR 기법을 통해 일정 시간 동안 접근이 되지 않을 시 Fine-grained 군집형 해시 테이블에서 Coarse-grained 군집형 해시 테이블로 강등되는 강등 정책이 적용될 수 있다. 한편, 상기 승급정책이 실행된 후, 접근 빈도는 초기화되며, 상기한 1단계 fine-grained 군집형 해시 테이블에서도 이러한 정책은 동일하게 적용된다.
- <29> 따라서, 2단계 fine-grained 군집형 해시 테이블을 이용해 미스 페널티를 감소시키며 Coarse-grained 군집형 해시 테이블의 접근 빈도를 낮출 수 있으며, 주소 변환 대부분이 Fine-grained 군집형 해시 테이블에서 이뤄지므로 주소 변환 시간을 줄일 수 있다.
- <30> 상기한 NUR 기법은 수정 비트라는 비트 벡터를 사용하여 시스템에서 정해진 주기마다 비트에 따라 최근에 사용하지 않는 슬롯들을 분별할 수 있는 방법이다. 따라서, 두 개의 비트를 통해 참조된 상황과 수정이 되었는지를 판단하여 표 1과 같이 Fine-to-Coarse 교체 기법에 적용된다.

표 1

참조 비트	수정 비트	정책
0	0	Coarse로 강등
0	1	상태 유지
1	0	상태 유지
1	1	상태 유지

- <31> 즉, 표 1과 같이, 특정 주소의 NUR의 참조, 수정 비트가 모두 0인 경우는 그 주소는 사용되지 않는 것으로 보고, 상기 Fine-grained 군집형 해시 테이블에서 Coarse-grained 군집형 해시 테이블로 이동 시키고 Fine-grained 군집형 해시 테이블의 슬롯을 삭제한다.
- <32> 그러면, 도 4에 도시된 군집형 해시 테이블의 특정 버킷의 서브블록들이 가리키고 있는 PBA의 페이지들을 선 반입하는 방식을 설명하면 다음과 같다.
- <33> 먼저 일반적으로 사용되는 방식 즉, FTL에 의해 LBA 요청할 때마다 주소를 변환하고 데이터를 전송하는 것과는 달리, 본 발명에 따른 연속적인 페이지를 위한 선반입 기법은 Fine-grained 군집형 해시 테이블에서 하나의 버

켓 내에 연속된 LBA를 갖는 페이지를 선반입하는 것이다.

- <35> 즉, 주소 변환 기법에서 사용되는 자료구조인 군집형 해시 테이블에서는 동일한 버킷 안에 연속된 LBA들과 이에 대응하는 PBA들을 저장하는 것으로, 특정 버킷의 LBA에 대한 읽기 요청 시 인접한 서브블록들의 물리 페이지들을 선 반입 할 수 있는 것이다. 따라서, 반복적인 메모리 적재 및 플래시 메모리에 명령어를 매 시간마다 내리는 것을 방지할 뿐만 아니라, 다수의 페이지를 선반입하므로 다음 읽기 수행 시에 읽기 동작의 속도를 단축하게 된다.
- <36> 상기한 군집형 해시 테이블은 CFTL에서 사용하는 주소 변환 자료구조이며 썬마이크로시스템즈 사의 64비트 운영 체제인 솔라리스의 가상 메모리 페이지 기법에서 처음 사용하던 자료구조로서, 서브블록의 집합인 버킷으로 구성된다. 여기서, 하나의 버킷의 서브블록의 개수를 서브블록 팩터(Subblock factor)라 하며, 일반적인 해시 테이블과 달리 하나의 버킷에 여러 개의 서브블록이 존재하므로 주소당 포인터의 메모리 사용량을 1/(서브블록 팩터) 만큼 줄일 수 있는 특징을 갖고 있다.
- <37> 따라서, 연속적이면서 동시에 사용하는 데이터들이 한 버킷의 서브 블록들에 연속적으로 저장되므로 주소 변환 테이블 내에서 지역성을 유지하는데 매우 유용하며, 본 발명에서의 주소변환 자료구조로 적용된다.
- <38> 한편, 본 발명에서 전술된 Coarse-grained 군집형 해시 테이블의 각 버킷의 서브블록마다 연속성 플래그를 추가하고 있는데, VBA(Virtual Block Address)와 PBA(Physical Block Address)가 함께 연속적인 경우 연속적인 블록의 개수가 상기 연속성 플래그의 값에 기입된다. 이를 통해 주소변환 메커니즘을 다시 반복하지 않고도 PBA를 획득할 뿐만 아니라, 주소 변환을 가능하게 한다. 결국, 상기 Fine-grained 군집형 해시 테이블은 접근 빈도가 높은 LBA를 독립적인 테이블로 구성하므로 주소 검색 시간을 감소시켜 주소 변환 성능을 향상시키며, 해당 버킷의 서브블록들이 가리키는 PBA의 데이터를 미리 주 메모리에 선반입하여 주소 변환 성능을 향상시킨다.
- <39> 여기서, 전술된 연속성 플래그 즉, 연속적인 LBA의 주소변환 요청 시 주소변환 방법으로 적용되는 연속성 플래그는 도 3의 Coarse-grained 군집형 해시 테이블 내에 도시된 서브블록들에 저장되는 'C' 값이다. 연속적인 VBA와 함께 물리블록 또한 연속적인 경우 연속성 플래그인 C에 연속적인 블록의 개수를 기입하는데 사용되며, 이를 통해 주소변환 메커니즘을 다시 반복하지 않고도 주소 변환이 가능하다. 예컨대, VBA가 128, PPBA가 386, 연속성 플래그가 8일 경우 VBA의 증가 값이 8 이하인 경우 주소 변환 없이 PPBA의 값을 얻을 수 있는 것이다.
- <40> 도 7은 본 발명에 따른 CFTL의 성능을 비교한 그래프이다. CFTL의 환경은 리눅스 2.6.17의 MTD(Memory Technology Device)를 사용하여 실험하였으며, 성능 평가를 위해 1단계 Fine-grained 군집형 해시 테이블, 2단계 fine-grained 군집형 해시 테이블의 슬롯의 비율은 1:4로 할당하였다. 예를 들어 MSFS(Maximum Short Fine-grained Slot)가 1000개일 경우, MLFS(Maximum Long Fine-grained Slot)의 개수는 4000개를 갖는 것을 의미한다. 실험을 위해 군집형 해시 테이블의 서브블록 팩터, 즉 서브블록의 개수를 8로 설정하고 MSFS는 2500, MLFS는 10,000으로 설정했다. 2비트 쓰기 예측 기법 예측테이블의 슬롯 수는 1024, 2048, 4096개로 설정하였다.
- <41> 여기서, 주소 변환 시간 및 FTL의 전체적인 성능을 측정하기 위해 저장장치 성능평가로 방식으로 잘 알려진 앤드류 벤치마크를 사용했다. 앤드류 벤치마크는 1) 디렉터리 생성, 2) 데이터 복사, 3) 재귀적인 파일 검색, 4) 파일 접근, 5) 컴파일과 같이 5단계로 분류된다. 이러한 테스트는 저장장치에서 읽기/쓰기의 성능을 확인하는데 효과적이다. 도시된 바와 같이 10회의 값을 평균을 낸 결과 CFTL의 주소 변환 속도가 NFTL보다 약 13% 빠르고 AFTL보다 약 8% 빠른 것으로 나타났다. 또한 추가적으로 연속성 플래그와 선반입 기법을 적용한 경우 연속적인 주소에 관한 예측으로 인해 NFTL보다 17%이상의 성능이 향상되고 AFTL보다 11% 이상 성능 향상되었다.
- <42> 더구나, CFTL의 주소변환 테이블의 크기를 비교할 경우, 주소변환 테이블 메모리의 사용량이 감소됨을 알 수 있다. 이는 표 2에 도시한 바와 같이 NFTL과 AFTL 그리고 서브블록 팩터가 8인 CFTL의 주소 변환 테이블의 크기를 비교할 경우 CFTL이 AFTL보다 주소 변환 테이블 메모리 사용량을 최대 43% 이상 감소되고 있음을 인지할 수 있다. 또한 CFTL의 서브블록 팩터를 32로 설정할 경우 메모리 사용량이 최대 65%까지 절약할 수 있음을 나타낸다.

표 2

MFS	NFTL	AFTL	CFTL
500 + 2,000	64.0KB	201.8KB	78.1KB
1,000 + 4,000	64.0KB	211.5KB	84.2KB
1,500 + 6,000	64.0KB	221.3KB	90.3KB
2,000 + 8,000	64.0KB	231.1KB	96.4KB

2,500 + 10,000	64.0KB	240.8KB	102.5KB
3,000 + 12,000	64.0KB	250.6KB	108.6KB

- <44> 한편, 본 발명은 PBA의 데이터를 미리 주 메모리에 선반입하여 주소 변환을 수행함에 있어, 자주 갱신되는 데이터를 플래시 메모리에 바로 저장하지 않고 쓰기 버퍼를 이용하여 먼저 저장한다. 이는 플래시 메모리의 수명을 늘리기 위한 것으로, 반복적인 쓰기 동작을 최소화하기 위한 수단이다. 본 발명에서는 이러한 쓰기 방법을 '2비트 쓰기 예측'이라 칭할 것이다.
- <45> 상기 2비트 쓰기 예측은 예측을 위한 상태를 4가지로 분류한다. 도 5는 2비트 쓰기 예측 기법을 설명하기 위한 도면이며, 도 6은 도 5를 이용한 비동기 쓰기방법을 설명하기 위한 도면이다. 이를 참조하여 설명하면 다음과 같다.
- <46> 먼저, 반복적으로 저장되는 LBA를 포함하는 VBA를 예측 테이블에 추가하고 그 후 동일한 VBA에 소속된 LBA에 쓰기 요청이 들어올 때마다 예측 비트가 갱신된다. 예측 비트가 10, 11인 경우 자주 갱신되는 VBA로 판단하고 RAM의 예약 버퍼에 갱신된 페이지를 적재한다.
- <47> 차후 예측 비트가 00, 01로 변하는 경우 예약 버퍼에 있는 데이터의 최신 데이터만을 플래시 메모리에 저장한다. 2비트 쓰기 예측 기법을 적용했을 경우, 쓰기 요청이 들어오는 순서에 따라 예측 테이블을 갱신한다. 2개의 비트로 4가지 상태를 표현하고 비트가 10, 11인 Predict written 상태로 판단되는 VBA(Virtual Block Address)를 예약 버퍼에 저장하고 있다가 예약 버퍼가 가득 차거나 예측 비트가 00, 01인 Predict not written 상태인 경우 예약 버퍼의 최신 데이터만 합병하여 이를 플래시 메모리에 적재한다. 이러한 쓰기 지연 기법은 플래시 메모리에 쓰기 횟수를 감소시켜 플래시 메모리의 수명을 향상시킬 수 있으며 데이터의 지역성을 유지하는 데도 많은 효과를 갖는다.
- <48> 도 8 내지, 도 10은 전술된 2비트 쓰기 예측 기법에 의한 플래시 메모리의 수명을 나타낸 그래프이다. 본 발명에 따른 실험에서는 2비트 쓰기 예측 기법과 쓰기 버퍼를 이용한 플래시 메모리 쓰기 횟수 감소에 주안점을 두었다. 쓰기 횟수 감소는 두 가지 장점을 가지고 온다.
- <49> 첫째 플래시 메모리의 수명의 척도인 마모도 감소를 가지고 오며, 둘째 쓰기 횟수 감소로 인해 쓰기와 함께 진행되는 블록 수거 기법의 실행 횟수를 같이 감소시킨다. 그러므로 2비트 쓰기 예측 기법은 매우 효율적으로 메모리 수명 및 성능을 향상시킬 수 있다. 도 8은 기존 파일들을 삭제하고 덮어쓰기에 중점을 둔 실험 결과를 나타낸다. 마지막으로, 도 9는 존재하는 파일에 데이터를 반복적으로 추가하는 실험 결과를 나타낸다.
- <50> 상기한 앤드류 벤치마크의 경우 다양한 파일 접근 패턴을 가지므로 2비트 쓰기 예측 기법이 적용되는 쓰기 영향을 많이 받지 않는 것을 알 수 있다. 하지만 덮어쓰기(Overwrite)와 추가 갱신(Append)에서는 2비트 쓰기 예측 기법 표의 크기와 비례하여 쓰기 횟수가 감소하며 예측 테이블의 슬롯이 4096개인 CFTL이 NFTL과 AFTL보다 쓰기 횟수를 최대 60%이상 감소시켰다.
- <51> 도 10은 기존 NFTL, AFTL과 2비트 쓰기 예측 기법을 사용한 CFTL의 쓰기 시간을 각각 앤드류 벤치마크, 갱신, 테스트에 따라 측정한 것이다. 그 결과 CFTL이 NFTL, AFTL 보다 쓰기 시간이 앤드류 벤치마크에서 약 10% 이상, 덮어쓰기, 덧붙이기 동작 벤치마크에서 약 30% 이상 감소했다. 또한 단축된 결과는 블록 수거 기법의 오버헤드를 감소시키는데 사용할 수 있음을 증명하고 있다.
- <52> 여기서, 본 발명에 따른 2비트 쓰기 예측 기법에서 사용하는 메모리의 사용량을 살펴 보면, 표 3과 같이 예측 테이블의 슬롯의 개수가 각각 1024, 2048, 4096개인 경우 표의 사이즈는 8KB, 16KB, 32KB까지 증가하고, 쓰기 버퍼가 사용하는 메모리 사용량은 각각 512KB, 1MB, 2MB으로 증가하고 있다.

표 3

<53>

슬롯 수	메모리 사용량	쓰기 버퍼 크기
1024	8KB	512KB
2048	16KB	1MB
4096	32KB	2MB

<54> 결과적으로, 본 발명에 따른 2비트 쓰기 예측 기법을 이용한 비동기 쓰기 기법을 통해, 선반입과 비동기 쓰기

기법이 주소 변환 성능에서 기존 NFTL보다 17% 빠르고, 쓰기 성능에서도 약 30%이상 향상됨을 증명하고 있다.

산업이용 가능성

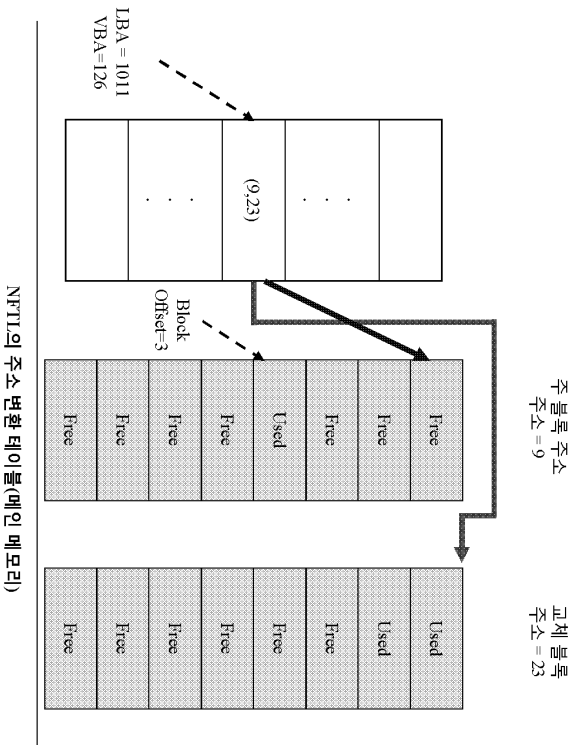
- <55> 전술된 바와 같이, 본 발명은 NAND 플래시 메모리가 필요한 모든 분야에 적용 가능하므로, 국내의 업체의 NAND 플래시의 기술을 지원하는 핵심 기술로 기업화가 가능할 것이다. 근래에는, 모바일 핸드셋, MP3 디바이스/퍼스널 미디어 플레이어, 디지털 카메라, USB 플래시 드라이브와 같은 주요 제품에서의 수요가 크게 증가하고 있기 때문에, NAND 플래시 메모리의 성장률은 2008년에 130%에서 140% 성장할 것으로 전망되고 있으며, 대부분의 공급 업체들은 제조 생산량을 늘리는 동시에 노드 공정 이전을 공격적으로 추진하고 있어 2008년 하반기에는 수요와 공급이 균형을 이룰 것으로 전망된다.
- <56> 또한 모바일 핸드셋 제조업체들의 수요는 전년에 비해 10% 가량 증가하여 12억 3,000만 달러에 이를 것으로 전망된다. 차세대 주요 분야는 MP3/PMP 디바이스가 될 것으로 20% 가량 성장한 2억만 달러에 이를 것이며, USB 플래시 드라이브가 매년 25% 가량 성장하여 1억 7,000만 달러로 그 뒤를 차지할 것으로 보인다. 디지털 카메라 시장 성장은 올해 14% 1억 3,000만 달러 매출을 달성할 것으로 기대된다.
- <57> 최근에 삼성전자, 하이닉스 등 NAND 플래시 메모리 제조 업체 및 휴대용 저장장치 제조 업체에서는 플래시 메모리 소프트웨어의 성능을 향상시켜 높은 품질의 하드웨어를 내놓기 위해 플래시 메모리 소프트웨어의 관심이 높아지고 있으므로 제조업체의 연구그룹의 규모가 커질 것으로 예상 된다. 따라서, 본 발명에 따른 NAND 메모리의 관리 방안은 산업적 이용 가치가 극히 높을 것으로 판단된다.

도면의 간단한 설명

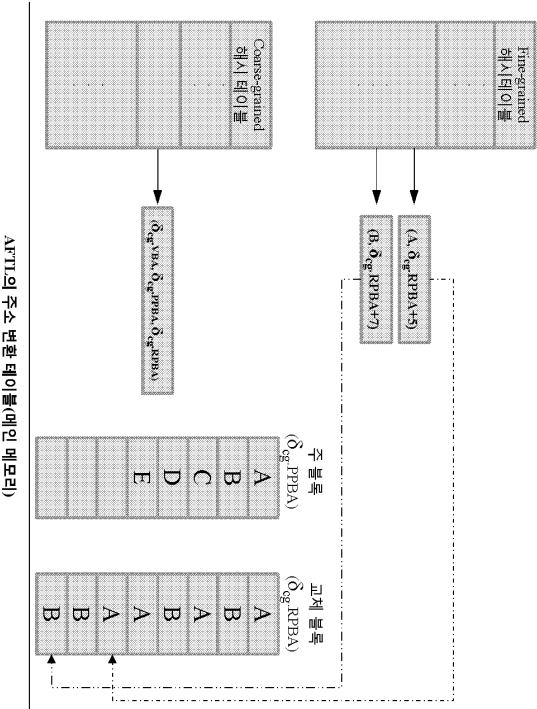
- <58> 도 1 및 도 2는 종래 주소변환 기법을 설명하기 위한 도면이다.
- <59> 도 3은 본 발명에 따른 플래시 변환 계층을 위한 선반입 절차로 적용되는 CFTL 방식의 주소변환 테이블 기법을 나타낸 도면이다.
- <60> 도 4는 도 3의 CFTL 방식을 이용한 플래시 변환 계층을 위한 선반입 절차를 도시한 흐름도이다.
- <61> 도 5는 본 발명으로 적용되는 2비트 쓰기 예측 기법을 설명하기 위한 도면이다.
- <62> 도 6은 도 5를 이용한 비동기 쓰기 방법을 설명하기 위한 도면이다.
- <63> 도 7은 본 발명에 따른 CFTL의 성능을 비교한 그래프이다.
- <64> 도 8 내지, 도 10은 전술된 2비트 쓰기 예측 기법에 의한 플래시 메모리의 수명을 나타낸 그래프이다.

도면

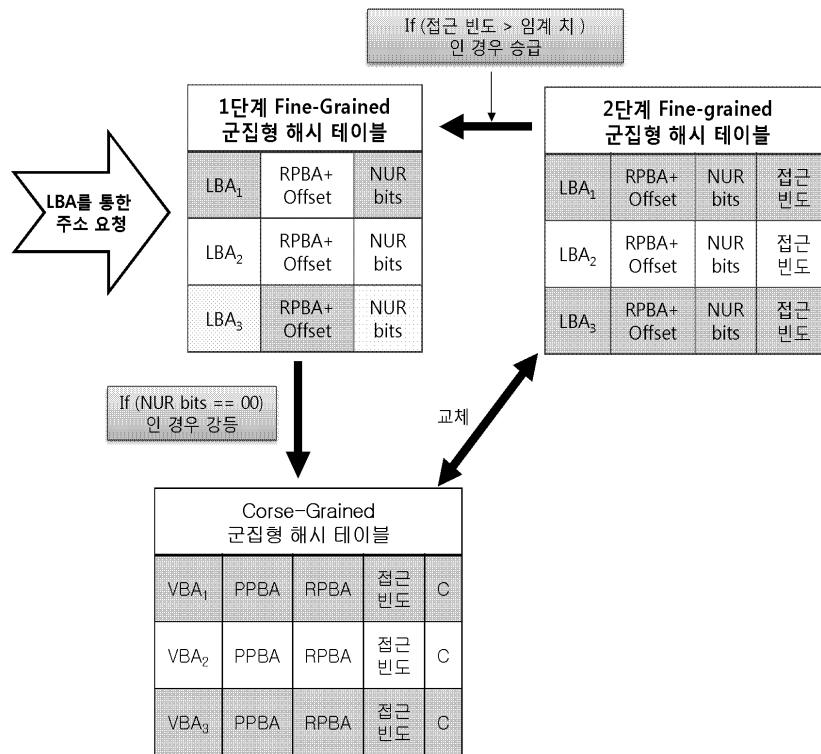
도면1



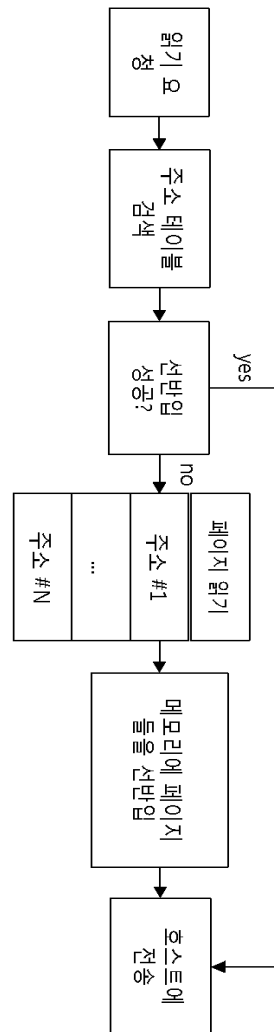
도면2



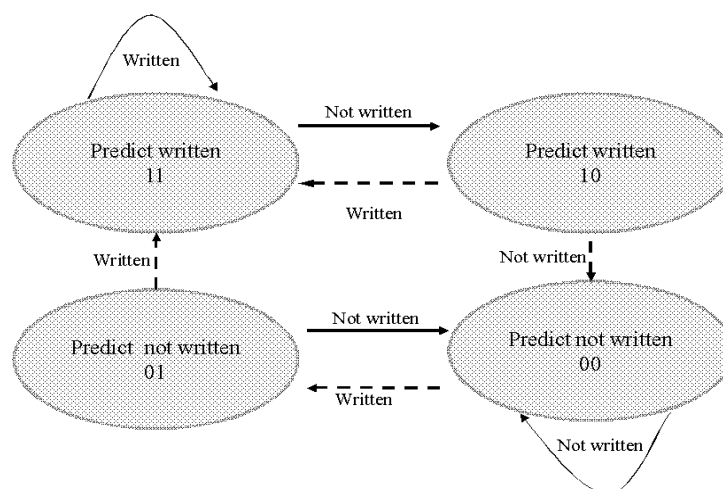
도면3



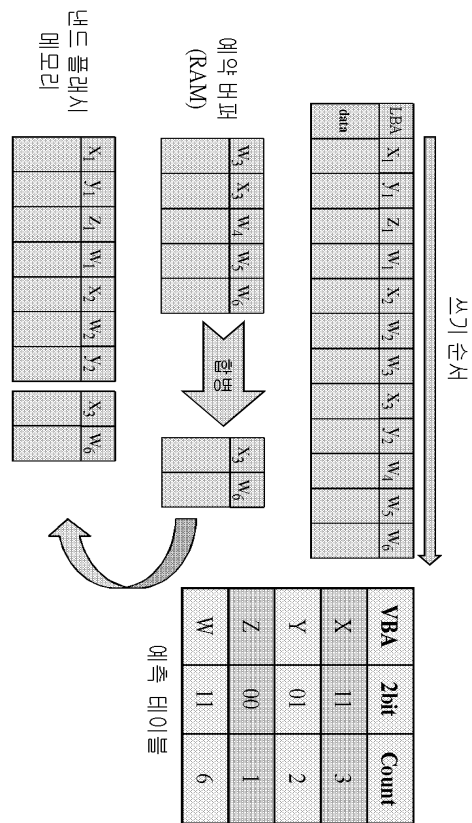
도면4



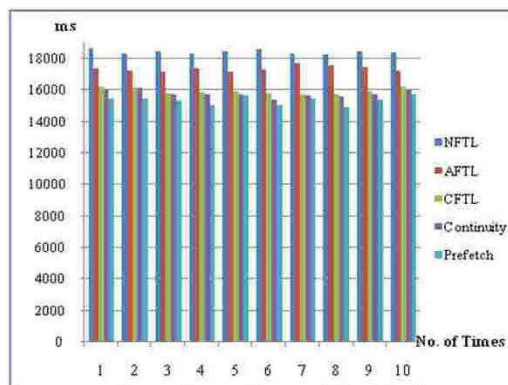
도면5



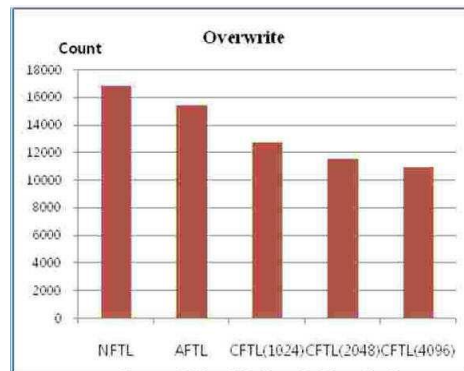
도면6



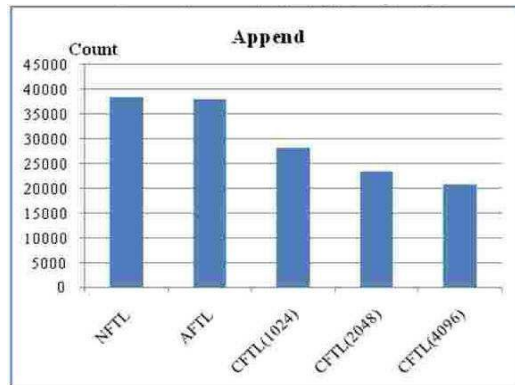
도면7



도면8



도면9



도면10

