



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2012-0034976  
(43) 공개일자 2012년04월13일

(51) 국제특허분류(Int. Cl.)  
G06F 12/02 (2006.01) G06F 3/06 (2006.01)  
(21) 출원번호 10-2010-0096402  
(22) 출원일자 2010년10월04일  
심사청구일자 2010년10월04일

(71) 출원인  
한국과학기술원  
대전 유성구 구성동 373-1  
(72) 발명자  
맹승렬  
대전광역시 유성구 대학로 291, 한국과학기술원 4403호 (구성동)  
허재혁  
대전광역시 유성구 대학로 291, 한국과학기술원 전산학과 (구성동)  
진성욱  
대전광역시 유성구 대학로 291, 한국과학기술원 전산학과 4414호 (구성동)  
(74) 대리인  
특허법인충현

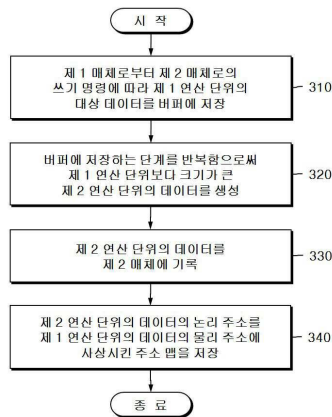
전체 청구항 수 : 총 15 항

(54) 발명의 명칭 **낸드 플래시 메모리에서 데이터의 주소를 사상시키는 장치 및 방법**

**(57) 요약**

본 발명은 낸드 플래시 메모리에서 섹터 단위의 데이터의 주소를 페이지 단위의 데이터의 주소로 사상시키는 장치, 방법 및 이를 기록한 기록매체에 관한 것으로, 본 발명에 따른 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법은 제 1 매체로부터 제 2 매체로의 쓰기 명령에 따라 제 1 연산 단위의 대상 데이터를 버퍼에 저장, 버퍼에 저장하는 단계를 반복함으로써 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성하고, 생성된 제 2 연산 단위의 데이터를 제 2 매체에 기록하며, 제 2 연산 단위의 데이터의 논리 주소를 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소 맵을 저장한다.

**대표도** - 도3



이 발명을 지원한 국가연구개발사업

과제고유번호 K1002090

부처명 지식경제부

연구사업명 지식경제기술혁신사업

연구과제명 신뢰성 컴퓨팅(Trustworthy Computing)기반 기술 개발

주관기관 한국과학기술원

연구기간 2010.03.01 ~ 2011.02.28

---

## 특허청구의 범위

### 청구항 1

서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상(mapping)시키는 방법에 있어서,

제 1 매체로부터 제 2 매체로의 쓰기(write) 명령에 따라 제 1 연산 단위의 대상 데이터를 버퍼에 저장하는 단계;

상기 버퍼에 저장하는 단계를 반복함으로써 상기 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성하는 단계;

상기 생성된 제 2 연산 단위의 데이터를 상기 제 2 매체에 기록하는 단계; 및

상기 제 2 연산 단위의 데이터의 논리 주소를 상기 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소 맵(map)을 저장하는 단계를 포함하는 방법.

### 청구항 2

제 1 항에 있어서,

상기 제 1 연산 단위는 섹터(sector)이고, 상기 제 2 연산 단위는 페이지(page)이며, 상기 제 2 매체는 낸드 플래시 메모리(NAND flash memory)인 것을 특징으로 하는 방법.

### 청구항 3

제 1 항에 있어서,

상기 주소 맵을 저장하는 단계는 상기 제 2 연산 단위의 데이터의 논리 주소를 키(key) 값으로 하는 해시(hash) 함수를 사용하여 주소를 사상시키는 것을 특징으로 하는 방법.

### 청구항 4

제 3 항에 있어서,

상기 주소 맵의 엔트리(entry)에는 상기 제 2 연산 단위의 데이터에 포함되는 제 1 연산 단위의 데이터의 물리 주소가 사상되어 저장되는 것을 특징으로 하는 방법.

### 청구항 5

제 1 항에 있어서,

제 1 매체로부터 제 2 매체로의 쓰기 명령에 따라 수신된 대상 데이터의 크기가 상기 제 2 연산 단위의 크기와 일치하는지 여부를 검사하는 단계를 더 포함하고,

상기 검사 결과 상기 수신된 대상 데이터의 크기가 상기 제 2 연산 단위의 크기와 일치하는 경우 상기 버퍼에 저장하는 단계 및 상기 제 2 연산 단위의 데이터를 생성하는 단계를 생략하는 것을 특징으로 하는 방법.

### 청구항 6

제 1 항에 있어서,

상기 제 2 매체에 기록하는 단계는,

상기 생성된 제 2 연산 단위의 데이터를 상기 제 2 매체 내의 소정 공간에 기록하는 단계;

소정 변환 계층에 따라 상기 제 1 매체의 논리 주소를 상기 제 2 매체의 물리 주소로 변환하는 단계; 및

상기 소정 공간을 상기 제 2 매체의 기록 공간으로 전환하는 단계를 포함하는 방법.

### 청구항 7

제 6 항에 있어서,

상기 소정 변환 계층은 플래시 변환 계층(Flash Translation Layer, FTL)인 것을 특징으로 하는 방법.

**청구항 8**

제 6 항에 있어서,

상기 소정 공간과 상기 제 2 매체는 저장 공간을 공유함으로써, 상기 소정 공간의 물리적인 위치 및 크기를 가변적으로 변화시키는 것을 특징으로 하는 방법.

**청구항 9**

서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법에 있어서,

제 1 연산 단위의 크기를 갖는 데이터에 대한 읽기(read) 명령을 수신하는 단계;

주소 맵을 이용하여 상기 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 크기를 갖는 데이터의 물리 주소를 획득하는 단계; 및

상기 획득된 물리 주소에 저장된 데이터로부터 상기 읽기 명령의 대상 데이터를 독출하는 단계를 포함하고,

상기 주소 맵은 상기 제 2 연산 단위의 데이터의 논리 주소를 상기 제 1 연산 단위의 데이터의 물리 주소에 미리 사상시켜 저장하는 것을 특징으로 하는 방법.

**청구항 10**

제 9 항에 있어서,

상기 제 1 연산 단위는 섹터이고, 상기 제 2 연산 단위는 페이지이며, 상기 읽기 명령의 대상 데이터는 낸드 플래시 메모리에 저장되어 있는 것을 특징으로 하는 방법.

**청구항 11**

제 9 항에 있어서,

상기 주소 맵은 상기 제 2 연산 단위의 데이터의 논리 주소를 키 값으로 하는 해시 함수를 사용하여 주소를 사상시킨 것을 특징으로 하는 방법.

**청구항 12**

제 1 항 내지 제 11 항 중에 어느 한 항의 방법을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체.

**청구항 13**

서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 장치에 있어서,

제 1 매체로부터 제 2 매체로의 쓰기 명령에 따라 제 1 연산 단위의 대상 데이터를 저장하고, 상기 명령에 따라 저장된 복수 개의 대상 데이터로부터 상기 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성하는 버퍼;

상기 제 2 연산 단위의 데이터의 논리 주소를 상기 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소 맵을 저장하는 주소 맵 관리부; 및

상기 생성된 제 2 연산 단위의 데이터를 상기 제 2 매체에 기록하는 기록부를 포함하는 장치.

**청구항 14**

제 13 항에 있어서,

상기 제 1 연산 단위는 섹터이고, 상기 제 2 연산 단위는 페이지이며, 상기 제 2 매체는 낸드 플래시 메모리인 것을 특징으로 하는 장치.

**청구항 15**

제 13 항에 있어서,

상기 주소 맵은 상기 제 2 연산 단위의 데이터의 논리 주소를 키 값으로 하는 해시 함수를 사용하여 주소를 사상시키고,

상기 주소 맵의 엔트리에는 상기 제 2 연산 단위의 데이터에 포함되는 제 1 연산 단위의 데이터의 물리 주소가 사상되어 저장되는 것을 특징으로 하는 장치.

**명세서**

**기술분야**

[0001] 본 발명은 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 장치 및 방법에 관한 것으로, 특히 낸드 플래시 메모리에서 섹터 단위의 데이터의 주소를 페이지 단위의 데이터의 주소로 사상시키는 장치, 방법 및 이를 기록한 기록매체에 관한 것이다.

**배경기술**

[0002] 플래시 메모리는 소비전력이 작고, 전원이 꺼지더라도 저장된 정보가 사라지지 않은 채 유지되는 비휘발성 저장 매체로서, 디램(DRAM)과는 달리 전원이 끊기더라도 저장된 정보를 그대로 보존할 수 있을 뿐만 아니라 정보의 입출력도 자유로워 디지털 텔레비전, 디지털 캠코더, 휴대전화, 디지털 카메라, 개인휴대단말기(PDA), 게임기, MP3 플레이어 등에 널리 활용되고 있다.

[0003] 플래시 메모리의 종류에는 크게 저장 용량이 큰 낸드(NAND, Not AND) 플래시 메모리와 처리 속도가 빠른 노어(NOR, Not OR) 플래시 메모리의 2 가지로 분류된다. 비트선과 접지선 사이에 셀(cell)이 병렬로 배치된 노어 플래시 메모리와 달리 낸드 플래시 메모리는 셀이 직렬로 배치된 데이터 저장형 플래시 메모리이다. 특히, 낸드 플래시 메모리는 기존의 저장 매체인 하드디스크와 같은 자기식 저장장치들과는 달리 내구성이 우수하고, 기계적인 메카니즘이 존재하지 않음으로 인하여 매우 빠른 액세스 속도를 가진다. 나아가, 고집적과 대용량화가 가능하며, 값이 저렴하다는 장점이 있다. 따라서 휴대용 정보통신기기 사용이 증가하면서 낸드 플래시 메모리 시장도 점점 성장하고 있는 추세이다.

**발명의 내용**

**해결하려는 과제**

[0004] 본 발명이 해결하고자 하는 기술적 과제는 기록하려는 대상 데이터와 기록 매체가 서로 다른 연산 단위를 가짐으로 인해 매체에 불필요한 쓰기 명령이 발생하는 한계를 극복하고, 이로 인해 매체의 공간이 낭비되며 쓰기 성능이 저하되는 문제점을 해결하는데 있다. 나아가 본 발명은 쓰기 명령에 취약한 매체의 특성으로 인해 장치의 수명이 짧아지는 문제점을 해결하고자 한다.

**과제의 해결 수단**

[0005] 상기 기술적 과제를 해결하기 위하여, 본 발명에 따른 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법은 제 1 매체로부터 제 2 매체로의 쓰기(write) 명령에 따라 제 1 연산 단위의 대상 데이터를 버퍼에 저장하는 단계; 상기 버퍼에 저장하는 단계를 반복함으로써 상기 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성하는 단계; 상기 생성된 제 2 연산 단위의 데이터를 상기 제 2 매체에 기록하는 단계; 및 상기 제 2 연산 단위의 데이터의 논리 주소를 상기 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소 맵(map)을 저장하는 단계를 포함한다.

[0006] 상기 또 다른 기술적 과제를 해결하기 위하여, 본 발명에 따른 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법은 제 1 연산 단위의 크기를 갖는 데이터에 대한 읽기(read) 명령을 수신하는 단계; 주소 맵을 이용하여 상기 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 크기를 갖는 데이터의 물리 주소를 획득하는 단계; 및 상기 획득된 물리 주소에 저장된 데이터로부터 상기 읽기 명령의 대상 데이터를 독출하는 단계를 포함하고, 상기 주소 맵은 상기 제 2 연산 단위의 데이터의 논리 주소를 상기 제 1 연산 단위의 데이터의 물리 주소에 미리 사상시켜 저장한다.

[0007] 상기 기재된 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법들에서 상기 제 1 연산 단위는 섹터(sector)이고, 상기 제 2 연산 단위는 페이지(page)이며, 상기 제 2 매체는 낸드 플래시 메모리(NAND flash memory)인 것이 바람직하다.

[0008] 또한, 이하에서는 상기 기재된 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법들을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공한다.

[0009] 상기 기술적 과제를 해결하기 위하여, 본 발명에 따른 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 장치는 제 1 매체로부터 제 2 매체로의 쓰기 명령에 따라 제 1 연산 단위의 대상 데이터를 저장하고, 상기 명령에 따라 저장된 복수 개의 대상 데이터로부터 상기 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성하는 버퍼; 상기 제 2 연산 단위의 데이터의 논리 주소를 상기 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소 맵을 저장하는 주소 맵 관리부; 및 상기 생성된 제 2 연산 단위의 데이터를 상기 제 2 매체에 기록하는 기록부를 포함한다.

[0010] 상기 기재된 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 장치에서 상기 제 1 연산 단위는 섹터이고, 상기 제 2 연산 단위는 페이지이며, 상기 제 2 매체는 낸드 플래시 메모리인 것이 바람직하다.

**발명의 효과**

[0011] 본 발명은 기록하려는 대상 데이터와 기록 매체가 서로 다른 연산 단위를 갖는 상황에서 대상 데이터를 매체의 연산 단위로 가공함으로써 과도한 쓰기 명령을 수행을 방지하고, 그 결과 매체의 기록 공간을 절약할 수 있으며, 쓰기 성능을 향상시킬 수 있다. 또한, 본 발명은 쓰기 명령에 취약한 매체의 특성을 고려하여 쓰기 명령을 감소시킴으로써 장기적으로 장치의 수명을 증가시킬 수 있다.

**도면의 간단한 설명**

[0012] 도 1은 서로 다른 연산 단위를 갖는 매체들에서 데이터를 기록할 때 발생하는 기록 현상과 이를 개선하기 위한 본 발명의 기본 아이디어를 설명하기 위한 도면이다.

도 2는 본 발명의 일 실시예가 구현되는 환경에서 서로 다른 연산 단위의 데이터를 기록하기 위한 처리 과정을 도시한 흐름도이다.

도 3은 본 발명의 일 실시예에 따라 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법을 도시한 흐름도이다.

도 4a 및 도 4b는 본 발명의 일 실시예에 따라 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 장치가 구현되는 환경 및 그 장치를 도시한 블록도이다.

도 5는 본 발명의 일 실시예에 따른 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법을 데이터의 흐름에 따라 설명하기 위한 도면이다.

도 6은 도 5의 실시예에서 데이터의 쓰기 연산을 보다 구체적으로 설명하기 위한 도면이다.

도 7은 도 5의 실시예에서 데이터의 기록 공간을 확보하는 방법을 보다 구체적으로 설명하기 위한 도면이다.

도 8은 도 5의 실시예에서 데이터의 읽기 연산을 보다 구체적으로 설명하기 위한 도면이다.

도 9는 본 발명의 일 실시예에 따라 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시킴에 있어서 해시 함수를 사용하는 방법을 설명하기 위한 도면이다.

**발명을 실시하기 위한 구체적인 내용**

[0013] 본 발명의 실시예들을 설명하기에 앞서, 본 발명의 실시예들이 활용 및 구현될 수 있는 낸드 플래시 메모리의 특성과 이를 디스크로서 활용하기 위한 기술들에 대해 개괄적으로 소개하고자 한다. 이하에서는 다양한 실시예들을 설명함에 있어서 낸드 플래시 메모리를 예시하고 있으나, 이는 연산의 기본 단위가 서로 다른 경우에 있어서의 기록 특성을 설명하기 위한 것일 뿐이며, 단지 낸드 플래시 메모리에 한정되지 않는다. 또한, 본 발명의 실시예들은 그 동작 원리와 아이디어의 동일성이 유지되는 한 서로 다른 연산 단위를 갖는 다양한 매체들의 데이터 주소 사상(mapping)에서 유연하게 활용될 수 있다.

[0014] 낸드 플래시 메모리는 기본 단위인 페이지(page)와, 여러 개의 페이지로 구성된 블록(block)으로 구성된다. 낸

드 플래시 메모리의 페이지는 보통 2KB나 4KB의 크기를 가지며 이러한 페이지는 낸드 플래시 메모리의 읽기 및 쓰기의 기본 단위이다. 블록은 보통 64개나 128개의 페이지로 구성되며 낸드 플래시 메모리의 삭제 연산의 기본 단위이다. 낸드 플래시 메모리는 일반적으로 쓰기 연산이 읽기 연산보다 8-10배 정도 느리며 기록할 수 있는 회수에 제한이 있으므로 기록 방식의 개선에 따라 빠른 속도를 보장할 수 있으며, 낸드 플래시 메모리의 수명 또한 증가시킬 수 있다.

[0015] 낸드 플래시 메모리는 덮어쓰기(overwrite)가 지원되지 않는 특징을 가지므로 미리 써진 데이터에 덮어쓰기가 발생한 경우에는 여분의 공간(미리 지워진 페이지)에 갱신된 데이터를 기록하고 사상 장치를 통해서 새롭게 갱신되었다는 사실을 알려준다. 또한, 낸드 플래시 메모리는 읽기, 쓰기 단위는 페이지인데 반해서, 삭제 단위는 페이지보다 큰 블록 단위라는 특징을 가진다. 이로 인해 낸드 플래시 메모리는 가비지 컬렉션(garbage collection)을 통해 기존에 사용된 블록을 재활용하게 된다. 이 과정에서 기존의 블록을 삭제하기 전에 해당 블록에 유효한 페이지가 존재하는 경우에는 해당 페이지를 빈 공간에 복사한 후, 해당 블록을 삭제하여 추가적인 여유 공간을 확보하게 된다. 이러한 가비지 컬렉션은 복사하는 작업을 통해서 추가적인 쓰기 요청을 유발하게 되고 낸드 플래시 메모리의 수명을 단축시키는 원인이 되기도 한다.

[0016] 기존의 파일 시스템이나 블록 인터페이스 레이어(block interface layer)는 기본적으로 저장 매체로서 하드디스크를 활용한다는 가정하게 구현되어 있다. 그러나, 낸드 플래시 메모리는 쓰기 전에 지우기(erase before write)와 각 연산의 단위가 다른 특성으로 인해서 종래의 하드디스크와는 구조적으로 동일하게 동작할 수 없다는 약점을 가진다. 따라서, 낸드 플래시 메모리를 하드디스크와 동일하게 동작시킬 수 있도록 해주는 인터페이스 역할의 소프트웨어가 필요한데 이를 플래시 변환 계층(flash translation layer, FTL)라고 한다. FTL은 파일 시스템이나 블록 인터페이스 레이어에서 사용하는 논리 블록 주소(logical block address)를 낸드 플래시 메모리에 해당하는 물리 블록 주소(physical block address)로 변환해주는 사상(mapping) 장치이다. 이런 FTL은 사상 단위에 따라서 페이지 단위 주소 사상 (page-level address mapping)과 블록 단위 주소 사상(block-level address mapping)으로 분류될 수 있다.

[0017] 페이지 단위 주소 사상은 보다 상세한 단위의 사상이 가능하므로 페이지의 단위의 데이터 갱신이 발생할 경우에 좋은 성능을 보여줄 수 있지만 사상 단위가 작음으로 인해서 많은 양의 사상 정보를 고가의 SRAM이나 DRAM에 저장해야 하는 단점을 가진다. 이에 반해 블록 단위 주소 사상은 사상 단위가 페이지에 비해서 64배 또는 128배 크므로 보다 적은 양의 사상 정보를 관리해야 하는 장점을 가지지만 상대적으로 성능이 떨어진다. 그 이유는 보다 큰 사상 단위를 가지게 되므로, 사상 단위 내에서 일부분에 대한 데이터 갱신 요청이 발생하더라도 해당 사상 단위의 데이터 모두를 복사해야 하기 때문이다.

[0018] 도 1은 서로 다른 연산 단위를 갖는 매체들에서 데이터를 기록할 때 발생하는 기록 현상과 이를 개선하기 위한 본 발명의 기본 아이디어를 설명하기 위한 도면으로서, 앞서 설명한 낸드 플래시 메모리의 특성으로부터 그 원인을 찾을 수 있다.

[0019] 통상적으로 페이지의 크기는 2KB나 4KB의 크기를 가지므로 하드디스크에서 사용하는 섹터 단위보다 훨씬 큰 단위를 가지게 된다. 낸드 플래시 메모리를 사용하는 파일 시스템이나 블록 인터페이스 레이어는 종래의 하드디스크를 저장매체로 예상하여 제작되었기 때문에 섹터 단위로 I/O 요청을 하게 되고, 이로 인해서 FTL은 페이지보다 상대적으로 더 작은 단위의 요청을 받게 된다. 그러나 페이지는 섹터보다 훨씬 큰 단위이며, 낸드 플래시 메모리의 기본 쓰기 단위는 페이지이므로, 페이지보다 훨씬 작은 단위의 섹터를 기록하기 위해서는 페이지 전체를 써야 하는 과도한 쓰기 요청이 발생할 수밖에 없다. 또한 기존의 페이지에 섹터 단위의 갱신이 발생하게 되는 경우에는 기존의 페이지의 데이터를 읽은 후, 새로 쓸 섹터 크기의 쓰기 데이터와 병합한 다음에 비로소 하나의 페이지를 쓰게 된다. 즉, 섹터 단위의 크기를 쓰기 위해 하나의 페이지 쓰기가 발생하게 되고 추가적으로 하나의 페이지 크기의 읽기 요청 또한 발생하게 된다.

[0020] 도 1에서 [case 1]은 상기된 상황을 시각적으로 표현한 것으로서, 먼저 호스트로부터 8개의 쓰기 요청이 순서대로 수신된다. 이 중, 페이지 3에 포함된 일련의 섹터(4개의 연속된 섹터를 의미한다.)로 구성된 하나의 쓰기만이 전체 페이지 쓰기 요청이라고 가정하자(즉, 1개의 페이지는 4개의 섹터를 저장할 수 있다.). 나머지는 모두 부분 페이지 쓰기 요청으로서, 하나의 섹터 크기를 가지는 쓰기 요청이다. 이런 쓰기 요청 각각을 그대로 순차적으로 낸드 플래시 메모리에 쓰게 되면, 8개의 전체 페이지 쓰기 요청으로서 매체에 기록된다. 왜냐하면 낸드 플래시 메모리의 기본 쓰기 단위가 페이지이기 때문에 그보다 작은 섹터 크기의 쓰기 요청이 오더라도 페이지 전체를 다 쓸 수밖에 없기 때문이다. 도 1에서 하나의 페이지(110)에 비록 하나의 섹터(120)밖에 기록하지 못하였음에도 다음 페이지(140)에 쓰기 명령을 수행한 것을 볼 수 있다.

- [0021] 이제 본 발명의 실시예들이 제안하고자 하는 기본 아이디어를 [case 2]를 통해 살펴보자. [case 2]에서는 버퍼 (140)를 이용하여 같은 논리 페이지에 속하는 섹터들을 모아 하나의 페이지 쓰기기로 변환하는 방법을 제시하고 있다. 즉, [case 2]에서는 쓰기 명령의 대상 데이터들(섹터 단위의 데이터를 의미한다.)을 각각의 논리 페이지 별로 취합하여 하나의 페이지를 구성하고, 이렇게 구성된 데이터를 매체에 페이지 단위로 기록한다. 따라서, 낸드 플래시 메모리에는 3개의 전체 페이지 쓰기 요청만으로 쓰기 명령의 대상 데이터를 처리할 수 있다. 이러한 방법을 통해 [case 1]에 비해 [case 2]의 방법은 5개의 전체 페이지 쓰기 요청을 줄일 수 있다.
- [0022] 이상에서 검토한 기록 현상은 낸드 플래시 메모리에서 섹터의 크기보다 페이지의 크기가 더 크다는 데서 원인을 찾을 수 있다. 만약 페이지와 섹터의 크기가 동일하다면 상기된 문제는 발생하지 않는다. 그러나, 최근 낸드 플래시 메모리의 공정이 발달함에 따라 페이지의 크기는 점점 더 커지는 경향을 보이고 있으므로, 그에 따라 작은 섹터를 기록하기 위해서 낭비되는 공간 또한 점점 커질 것으로 예상된다. 특히 SSD(solid state drive)와 같은 저장 기기에서 활용되는 낸드 플래시 메모리는 병렬성을 극대화하기 위해서 다수의 칩들을 묶어서 활용하게 되는데, 이 경우 페이지와 섹터의 크기 차이는 점점 더 벌어지게 된다.
- [0023] 따라서, 상기 제시된 [case 2]와 같은 섹터의 압축을 통해 페이지를 구성하는 방법을 제안하는 바이다. 이하에서 설명할 본 발명의 실시예들에서는 섹터 쓰기 요청을 최대한 축적하여 하나의 페이지를 구성한 다음, FTL에 전달하여 최대한 쓰기 연산을 줄이고자 한다. 이하에서, 도면을 참조하여 본 발명의 다양한 실시예들을 상세하게 설명한다.
- [0024] 도 2는 본 발명의 일 실시예가 구현되는 환경에서 서로 다른 연산 단위의 데이터를 기록하기 위한 처리 과정을 도시한 흐름도이다.
- [0025] 210 단계에서 호스트로부터 쓰기 요청을 수신한다.
- [0026] 이어서, 220 단계를 통해 DRAM 버퍼를 통해 해당 쓰기 요청을 버퍼링한 후, 해당 공간이 가득 찼는지 검사한다. 이러한 DRAM 버퍼는 최근에 쓰거나 읽은 데이터를 저장함으로써 빠른 속도로 호스트에게 해당 데이터를 재전송 하는데 활용된다. 본 발명의 실시예에서 이러한 DRAM 버퍼는 필요에 따라 선택적으로 활용될 수 있을 것이다. 만약 DRAM 버퍼가 가득 찼다면 버퍼링된 데이터 중에 하나를 선택하여 내보내게 된다.
- [0027] 230 단계에서는 쓰기 요청에 따라 대상 데이터의 크기가 페이지의 크기와 일치하는지 여부를 검사한다. 만약 대상 데이터의 크기가 페이지의 크기와 일치한다면(즉, 페이지 단위의 쓰기 명령에 비어있는 섹터 단위의 공간이 없음을 의미한다.) 240 단계를 통해 바로 FTL로 전달되어 쓰기 처리될 것이다.
- [0028] 따라서, 본 발명의 실시예들은 제 1 매체(종래의 하드디스크와 같은 구조의 파일 시스템을 가지는 매체를 의미한다.)로부터 제 2 매체(낸드 플래시 메모리를 의미한다.)로의 쓰기 명령에 따라 수신된 대상 데이터의 크기가 제 2 연산 단위(페이지가 될 수 있다.)의 크기와 일치하는지 여부를 검사하는 단계를 더 포함할 수 있으며, 검사 결과 수신된 대상 데이터의 크기가 제 2 연산 단위의 크기와 일치하는 경우 곧바로 플래시 변환 계층(Flash Translation Layer, FTL)로 전달된다.
- [0029] 반면, 대상 데이터의 크기가 페이지의 크기보다 일치하지 않는다면(대상 데이터의 크기가 페이지의 크기보다 작음을 의미하며, 비어있는 섹터 단위의 공간이 존재한다.), 250 단계로 진행하여 본 발명의 실시예들이 제안하는 처리 과정 A에 따르게 된다. 처리 과정 A는 도 3을 통해 구체적으로 설명한다.
- [0030] 도 3은 본 발명의 일 실시예에 따라 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법을 도시한 흐름도이다. 이후의 실시예들에서는 편의상 제 1 매체 및 제 2 매체를 각각 종래의 파일 시스템을 갖는 매체 및 낸드 플래시 메모리를 의미하는 용어로서 사용하도록 하겠다. 또한, 제 1 연산 단위 및 제 2 연산 단위는 각각 섹터 단위 및 페이지 단위를 의미하는 용어로서 사용하도록 하겠다.
- [0031] 310 단계에서 제 1 매체로부터 제 2 매체로의 쓰기 명령에 따라 제 1 연산 단위의 대상 데이터를 버퍼에 저장한다. 이러한 버퍼는 통상적인 임시 저장 공간을 활용하여 구현될 수 있다.
- [0032] 320 단계에서는 310 단계의 버퍼에 저장하는 단계를 반복함으로써 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성한다. 즉, 페이지보다 작은 단위의 섹터 쓰기들을 하나의 페이지 단위로 모은다.
- [0033] 330 단계에서는 320 단계를 통해 생성된 제 2 연산 단위의 데이터를 제 2 매체에 기록한다. 즉, 하나의 페이지 단위의 데이터를 낸드 플래시 메모리 영역에 쓴다.
- [0034] 340 단계에서는 제 2 연산 단위의 데이터의 논리 주소를 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소



맵(address map)을 저장한다. 앞서 330 단계의 기록 과정에서 하나의 페이지 내에 모여진 섹터들은 각기 다른 논리 페이지 번호(logical page number)에 속하므로, 보다 상세한 섹터 단위의 사상이 필요하다. 본 실시예에서는 섹터 단위의 사상을 제공하여 각기 다른 논리 페이지에 속하는 섹터들을 하나의 물리 페이지에 기록하고, 공간이 부족해지는 경우 같은 논리 페이지 속하는 섹터들을 모아서 FTL에 전달한다. 이러한 과정을 통해 섹터 쓰기들이 모여서 하나의 페이지 단위의 쓰기로 변환되게 된다.

[0035] 상기된 과정들은 이러한 일련의 연산을 처리할 수 있는 처리기(processor) 및 이러한 연산에 필요한 기억공간(memory)을 통해 구현될 수 있으며, 필요에 따라서는 처리기와 기억공간 간의 데이터 처리를 적절하게 제어할 수 있는 제어기(controller)가 활용될 수도 있을 것이다. 이러한 처리기, 기억공간 및 제어기는 본 발명이 속하는 기술분야의 활용 환경이나 동작 환경을 고려하여 통상의 기술자에 의해 적절하게 선택될 수 있을 것이다. 나아가, 이러한 제어 과정에는 이상에서 예시된 하드웨어들을 제어하기 위한 추가적인 소프트웨어 코드(code)도 활용될 수 있을 것이다.

[0036] 상기된 본 발명의 실시예에 따르면 낸드 플래시 메모리의 과도한 쓰기 명령 수행을 감소시킬 수 있고, 이에 따라 기록 공간을 절약할 수 있다. 종래의 방식에 따라 섹터 쓰기 명령들을 그대로 FTL에 전달하게 되면, FTL은 섹터 쓰기를 위해 섹터보다 큰 페이지 단위의 쓰기를 수행하게 된다. 앞서 설명한 바와 같이 낸드 플래시 메모리의 기본 쓰기 단위가 페이지 단위이기 때문이다. 이로 인해 작은 크기의 쓰기 요청이 이보다 훨씬 큰 단위의 쓰기 요청으로 변경이 되게 되고, 이는 쓰기에 취약한 낸드 플래시 메모리의 성능을 저해하는 요소가 된다. 따라서, 본 발명의 실시예에 따르면 섹터 단위의 쓰기 명령들을 모아 하나의 페이지 단위의 쓰기 명령을 생성하여 FTL로 전달하게 되므로 쓰기 연산의 횟수를 줄일 수 있게 되고, 낸드 플래시 메모리의 쓰기 성능을 향상시킬 수 있을 뿐만 아니라 장기적으로 장치의 수명을 증가시킬 수 있다.

[0037] 도 4a 및 도 4b는 각각 본 발명의 일 실시예에 따라 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 장치가 구현되는 환경 및 그 장치를 도시한 블록도이다.

[0038] 도 4a를 참조하면, 서로 다른 연산 단위의 데이터를 사상시키기 위해 호스트(10)로부터 쓰기 명령이 수신된다. 임시 데이터 처리 장치(20)는 FTL에 쓰기 명령의 대상 데이터를 전달하기 전에 임시로 데이터를 처리하는 장치를 의미한다. 앞서 도 2를 통해 설명한 바와 같이 DRAM 버퍼가 활용될 수 있다. DRAM 버퍼에서는 쓰기 명령의 대상 데이터의 크기가 페이지의 크기와 일치하는지 여부를 판단한 후, 그 결과에 따라 FTL(30)에 직접 데이터를 전달하거나, 본 발명의 실시예들이 제안하고 있는 서로 다른 연산 단위의 주소를 사상시키는 장치(25)에 전달한다.

[0039] 즉, 본 발명의 실시예가 제안하고자 하는 주소 사상 장치(25)는 DRAM 버퍼(20)와 FTL(30)의 사이에 위치하여 섹터 단위의 쓰기 명령을 페이지 단위의 쓰기 명령으로 처리하기 위한 장치로 구현 가능하다. 이러한 구조를 통해서 주소 사상 장치(25)는 어떠한 FTL에서도 동작할 수 있는 장점을 가지며, 기존의 FTL를 이용하는 저장장치를 큰 구조 변경 없이 그대로 사용할 수 있다는 유연함을 갖는다.

[0040] 이하에서는 도 4a의 주소 사상 장치(25)를 도 4b를 통해 보다 구체적으로 설명한다.

[0041] 버퍼(410)는 제 1 매체로부터 제 2 매체로의 쓰기 명령에 따라 제 1 연산 단위의 대상 데이터를 저장하고, 명령에 따라 저장된 복수 개의 대상 데이터로부터 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 데이터를 생성한다. 즉, 버퍼(410)는 섹터 단위의 쓰기를 하나의 페이지 크기로 모으는 역할을 한다. 버퍼(410)가 필요한 이유는 낸드 플래시 메모리는 페이지 단위로 쓰기를 하므로, 동일한 논리 페이지 주소에 속하지 않는 섹터 쓰기 요청이 수신되더라도 일단 하나의 페이지 단위로 모아서 낸드 플래시 메모리에 써야 하기 때문이다. 이러한 버퍼(410)는 통상적인 기록 매체로 구현될 수 있으며, SRAM이나 DRAM으로 구성되는 것이 바람직하다.

[0042] 주소 맵 관리부(420)는 제 2 연산 단위의 데이터의 논리 주소를 제 1 연산 단위의 데이터의 물리 주소에 사상시킨 주소 맵을 저장한다. 주소 맵 관리부(420)는 논리 섹터 주소를 실제 물리 섹터 주소로 변환해주는 역할을 한다. 구현의 측면에서 주소 맵은 제 2 연산 단위의 데이터의 논리 주소를 키 값으로 하는 해시 함수를 사용하여 주소를 사상시킬 수 있으나, 이러한 해시 함수 외에 다양한 사상 방법이 활용될 수 있을 것이다. 해시 함수를 사용하여 주소 맵의 사상을 구현할 경우, 주소 맵의 엔트리(entry)에는 제 2 연산 단위의 데이터에 포함되는 제 1 연산 단위의 데이터의 물리 주소가 사상되어 저장될 수 있다. 해시 함수를 이용하여 데이터를 사상시키는 보다 구체적인 방법은 이후에 도 9를 통해 다시 설명하겠다.

[0043] 기록부(430)는 버퍼(410)를 통해 생성된 제 2 연산 단위의 데이터를 제 2 매체에 기록한다. 기록부(430)는 버퍼(410)에 축적된 하나의 페이지 크기를 갖는 섹터 쓰기 명령들을 실제 낸드 플래시 메모리에 쓰는 역할을 수행한

다.

- [0044] 상기된 본 발명의 실시예에 따르면 낸드 플래시 메모리의 과도한 쓰기 명령 수행을 감소시킬 수 있고, 이에 따라 기록 공간을 절약할 수 있다. 그 결과, 낸드 플래시 메모리의 쓰기 성능을 향상시킬 수 있을 뿐만 아니라 장기적으로 장치의 수명을 증가시킬 수 있다.
- [0045] 도 5는 본 발명의 일 실시예에 따른 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시키는 방법을 데이터의 흐름에 따라 설명하기 위한 도면이다. 우선 논리 주소에 따른 쓰기 명령들(510)이 수신된다. 이렇게 수신된 명령들(510)은 DRAM으로 구현된 버퍼(520)에 저장된다. 버퍼(520)에 저장된 데이터들 중 인접하지 않은 섹터들(non-contiguous sectors)은 본 발명의 실시예들이 제안하고 있는 데이터 사상 장치(530)를 통해 각각의 섹터 데이터를 하나의 페이지로 축적한다. 반면, 버퍼(520)에 저장된 데이터들 중 하나의 페이지를 구성하고 있는 인접한 섹터들(contiguous sectors)은 곧바로 FTL로 전달된다.
- [0046] 즉, 섹터 데이터의 저장 공간(버퍼를 의미한다.)이 가득 차서 더 이상 기록할 공간이 없는 경우에 버퍼로부터 내보낼 데이터를 선정해서 FTL로 최대한 페이지 단위의 쓰기로 모아서 전송하게 된다. 이 때, 섹터 단위의 쓰기 요청이 발생하게 되는 경우 본 발명의 실시예에 따른 데이터 사상 장치(530)에 의해 다수의 섹터 쓰기 요청들을 축적하여 하나의 페이지 크기의 쓰기로 변환해 줌으로서 쓰기 연산의 회수를 감소시킬 수 있다.
- [0047] 이 과정에서 사용되는 보다 구체적인 연산들은 이하의 도 6 내지 도 8을 통해 설명한다. 이하의 도면들에 도시된 실시예들에서는 4개의 섹터가 1개의 페이지를 구성한다고 가정한다.
- [0048] 도 6은 도 5의 실시예에서 데이터의 쓰기 연산을 보다 구체적으로 설명하기 위한 도면이다. 도 6에서 쓰기 명령의 대상 데이터(610)는 논리 섹터 번호(logical sector number)가 0이고 데이터의 크기는 1인 섹터 쓰기와 논리 섹터 번호가 5이고 데이터의 크기가 4인 섹터 쓰기가 존재한다.
- [0049] 우선, 이러한 두 개의 쓰기 요청이 버퍼(620)에 저장되게 되고, 버퍼(620)가 가득 되면, 기록부(630) 내의 빈 공간을 찾아 데이터를 기록하게 된다. 도 6에서는 기록부(630)의 블록 0번, 페이지 0번의 위치에 기록하였다. 이제 주소 맵 관리부(640)는 이러한 기록 위치를 파악하여 해당 논리 섹터 번호에 대응하는 물리 섹터 번호를 기록하게 된다.
- [0050] 본 발명의 실시예에 따른 주소 사상 장치를 사용하지 않은 경우, 최초에 수신된 2개의 쓰기 명령에 의해 2번의 페이지 쓰기가 발생하였을 것이나, 도 6에 도시된 주소 사상 장치를 활용함으로써 단 1회의 페이지 쓰기를 통해 2개의 쓰기 명령을 모두 수행할 수 있었다.
- [0051] 도 7은 도 5의 실시예에서 데이터의 기록 공간을 확보하는 방법을 보다 구체적으로 설명하기 위한 도면이다. 우선 데이터를 제 2 매체(기록 공간인 낸드 플래시 메모리를 의미한다.)에 기록하는 방법을 개괄적으로 살펴보면 다음과 같다.
- [0052] 버퍼를 통해 생성된 제 2 연산 단위의 데이터를 제 2 매체 내의 특정 공간에 기록한다. 이어서, FTL에 따라 제 1 매체의 논리 주소를 제 2 매체의 물리 주소로 변환한 후, 특정 공간을 제 2 매체의 기록 공간으로 전환한다. 즉, 본 발명의 실시예에서 언급하고 있는 기록 공간은 낸드 플래시 메모리에서 관리되는 통상의 저장 공간으로 전환된다.
- [0053] 일단 기록부(710)에 여분의 공간이 없는 경우에 기존의 저장된 데이터들을 내보내야 한다. 기록부(710)의 저장 공간에 저장된 데이터들은 섹터 쓰기 명령의 대상 데이터들을 저장한 것으로서, 최대한 한 페이지 단위로 묶어서 내보내는 것이 바람직하다. 먼저 기록부(710)가 데이터를 내보내기 위해 물리 블록 0번을 선택한 경우, 해당 블록에 유효한 데이터들만 내보내야 한다. 해당 블록에서 유효한 데이터를 포함하고 있는 페이지는 0번과 1번이다. 기록부(710)는 주소 맵 관리부(720)를 참조하여 논리 페이지 번호 0번에 해당되는 섹터들을 하나의 페이지로 모으게 되고, 또한 논리 페이지 번호 1번에 해당되는 섹터들을 하나의 페이지로 모으게 된다. 결국 기록부(710)는 기존의 섹터 쓰기 명령의 데이터들을 모아서 2개의 페이지로 만들어 FTL(730)에 전달하게 되고, FTL(730)은 기존의 섹터 쓰기가 아닌 페이지 쓰기로 요청을 받게 된다. 마지막으로 본 발명에 따른 실시예는 기록부(710)로부터 내보내기를 위해 선택된 블록 0을 지우게 된다.
- [0054] 이 경우 본 발명이 속하는 기술 분야에서 잘 알려진 '소거 횡수 평준화 기법(wear-leveling)'을 채용하는 것이 아니라, 지운 블록을 FTL과 교환하는 방식을 통해서 소거 횡수를 최대한 평준화시키는 것이 바람직하다. 즉, 기록부(710)에서 사용하는 낸드 플래시 메모리 영역은 그 위치나 크기가 고정되어 있는 것이 아니라 FTL(730)(보다 정확하게는 제 2 매체를 의미한다.)와 물리적인 공간을 공유함으로써 교환하는 방식을 취하게 된다. 따라서,

기록부(710)의 기록 공간은 필요에 따라 물리적인 위치 및 크기를 가변적으로 변화시킴으로써 기록 상황에 유연하게 대응하는 것이 가능하다.

- [0055] 도 8은 도 5의 실시예에서 데이터의 읽기 연산을 보다 구체적으로 설명하기 위한 도면으로서, 그 과정을 개괄적으로 설명하면 다음과 같다.
- [0056] 우선, 제 1 연산 단위의 크기를 갖는 데이터에 대한 읽기 명령(810)을 수신한다. 도 8에서는 논리 섹터 번호 0번에 해당되는 4의 크기를 가지는 읽기 요청이 발생하였다고 가정하자.
- [0057] 본 실시예에서는 버퍼(820) 내에 해당되는 데이터가 존재하는지 살펴본다. 만약 해당되는 데이터가 존재하지 않는 경우에는 주소 맵(830)을 통해서 해당 위치를 찾게 된다. 이러한 버퍼(820)를 검사하는 과정은 선택적으로 구현될 수 있을 것이다.
- [0058] 이제, 주소 맵(830)을 이용하여 제 1 연산 단위보다 크기가 큰 제 2 연산 단위의 크기를 갖는 데이터의 물리 주소를 획득한다. 도 8에서는 주소 맵(830)을 통해 읽기 요청에 해당하는 위치가 물리 블록 0번의 페이지 0번과 1번에 존재하므로 2번의 페이지 읽기를 통해서 해당되는 요청을 처리할 수 있다. 따라서, 획득된 물리 주소에 저장된 데이터로부터 읽기 명령의 대상 데이터를 독출한다.
- [0059] 한편, 주소 맵(830)은 제 2 연산 단위의 데이터의 논리 주소를 제 1 연산 단위의 데이터의 물리 주소에 미리 사상시켜 저장하는 것이 바람직하다. 또한, 주소 맵(830)은 제 2 연산 단위의 데이터의 논리 주소를 키 값으로 하는 해시 함수를 사용하여 주소를 사상시킬 수 있으나, 이러한 사상 방법은 본 발명이 구현되는 상황 및 실시예에 따라 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 유연하게 변경 가능하다.
- [0060] 도 9은 본 발명의 일 실시예에 따라 서로 다른 연산 단위를 갖는 매체들에서 데이터의 주소를 사상시킴에 있어서 해시 함수를 사용하는 방법을 설명하기 위한 도면이다. 앞서 간단히 언급한 바와 같이 주소 맵은 제 2 연산 단위의 데이터의 논리 주소를 키(key) 값으로 하는 해시(hash)(920) 함수를 사용하여 주소를 사상시킬 수 있다. 또한, 주소 맵의 엔트리(entry)에는 제 2 연산 단위의 데이터에 포함되는 제 1 연산 단위의 데이터의 물리 주소가 사상되어 저장될 수 있다.
- [0061] 본 발명의 실시예들에서 제안하고 주소 사상 방법에 따르면, 섹터 단위의 사상 방식은 기존의 페이지 단위의 사상 방식보다 훨씬 정교하므로 많은 양의 메모리를 사상 정보 저장에 사용할 수밖에 없다. 이를 효과적으로 처리하기 위해 본 실시예에서는 논리 페이지 주소(910)를 해시하여 저장하게 된다. 이 때, 버킷(bucket)의 수는 섹터 맵(930)의 공간 크기에 따라서 n개를 가질 수 있으며, 각 버킷은 m개의 하부 엔트리(940)를 가진다. 여기서 m개는 한 페이지에 포함되는 섹터의 수를 나타낸다. m개의 엔트리는 해당 데이터의 논리 페이지에 속하는 섹터들의 실제 물리주소를 나타낸다.
- [0062] 특히, 본 실시예에서는 페이지 단위의 데이터의 논리 주소를 섹터 단위의 데이터의 물리 주소에 사상시키는 방법을 사용하고 있다. 만약 논리 섹터 주소와 물리 섹터 주소를 직접 사상할 경우, 하나의 논리 페이지 주소에 해당되는 섹터들을 모두 찾아야 하는 경우가 빈번하게 발생하게 되므로 다수의 해시를 통해 목적 데이터를 검색해야 하는 불편함이 발생하게 될 우려가 있다. 따라서, 상기된 본 발명의 실시예에서는 논리 페이지 주소를 물리 섹터 주소에 사상시켜, 단 한 번의 논리 페이지 주소 검색을 통해서 하부의 섹터들을 검색함으로써, 검색 속도를 크게 향상시킬 수 있다.
- [0063] 한편, 본 발명은 컴퓨터로 읽을 수 있는 기록 매체에 컴퓨터가 읽을 수 있는 코드로 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다.
- [0064] 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피디스크, 광 데이터 저장장치 등이 있으며, 또한 캐리어 웨이브(예를 들어 인터넷을 통한 전송)의 형태로 구현하는 것을 포함한다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다. 그리고 본 발명을 구현하기 위한 기능적인(functional) 프로그램, 코드 및 코드 세그먼트들은 본 발명이 속하는 기술 분야의 프로그래머들에 의하여 용이하게 추론될 수 있다.
- [0065] 이상에서 본 발명에 대하여 그 다양한 실시예들을 중심으로 살펴보았다. 본 발명에 속하는 기술 분야에서 통상의 지식을 가진 자는 본 발명이 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 변형된 형태로 구현될 수 있음을 이해할 수 있을 것이다. 그러므로 개시된 실시예들은 한정적인 관점이 아니라 설명적인 관점에서 고려되어야 한다. 본 발명의 범위는 전술한 설명이 아니라 특허청구범위에 나타나 있으며, 그와 동등한 범위 내에 있

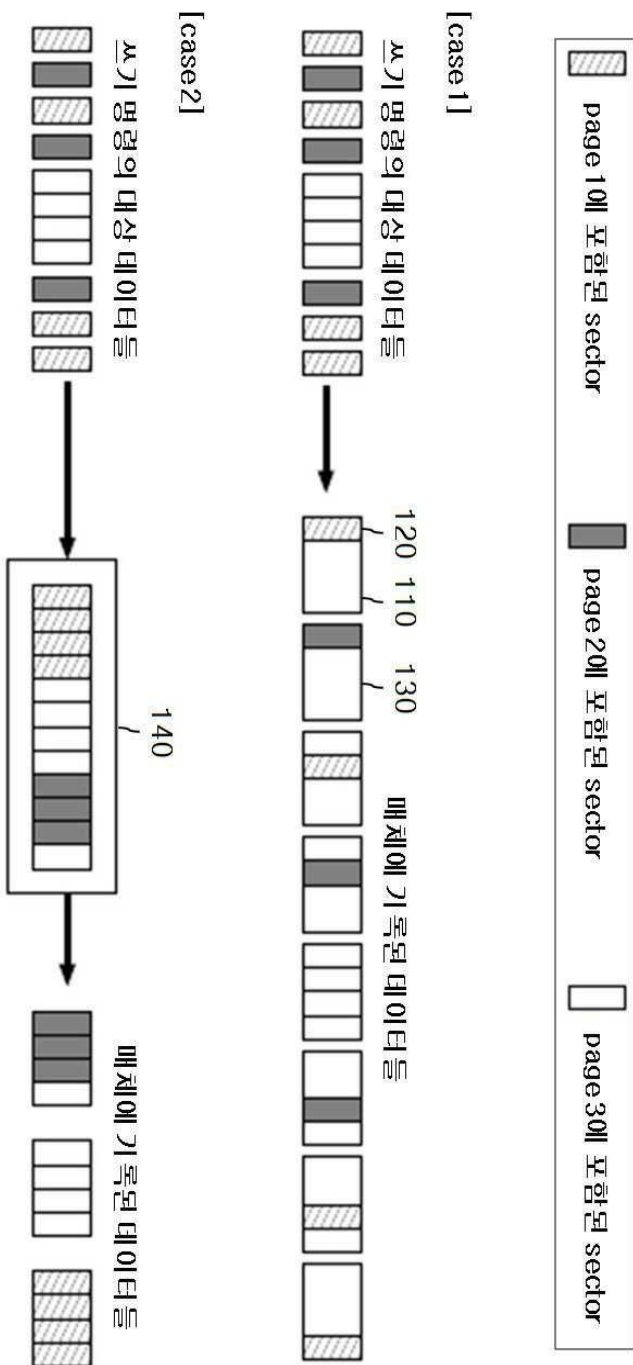
는 모든 차이점은 본 발명에 포함된 것으로 해석되어야 할 것이다.

**부호의 설명**

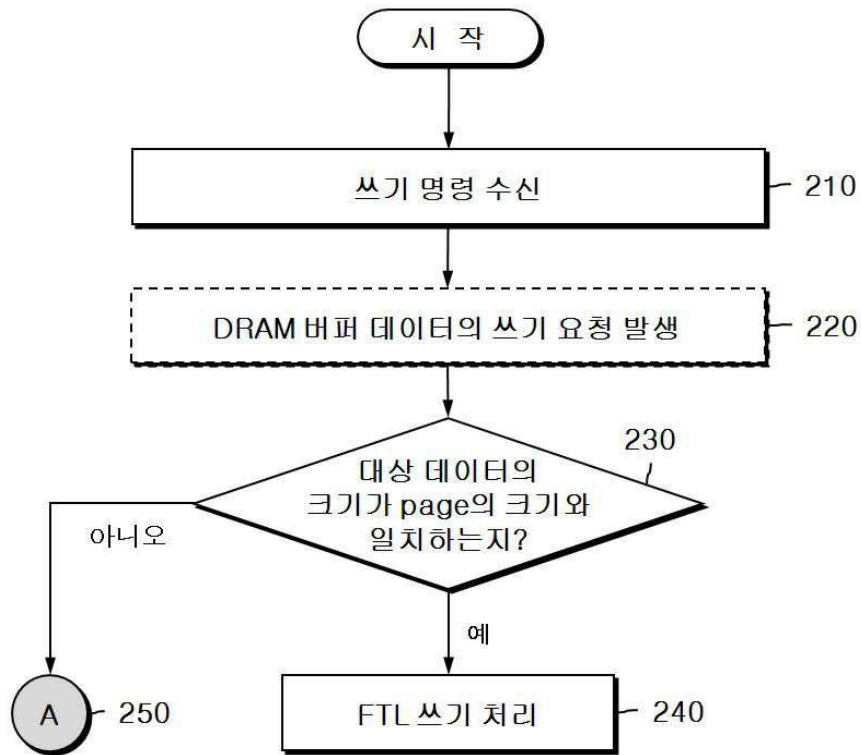
- [0066]
- 10 : 호스트
  - 20 : 임시 데이터 처리 장치
  - 25 : 서로 다른 연산 단위의 주소 사상 장치
  - 30 : 플래시 변환 계층(Flash Translation Layer, FTL)
  - 410 : 버퍼
  - 420 : 주소 맵 관리부
  - 430 : 기록부

도면

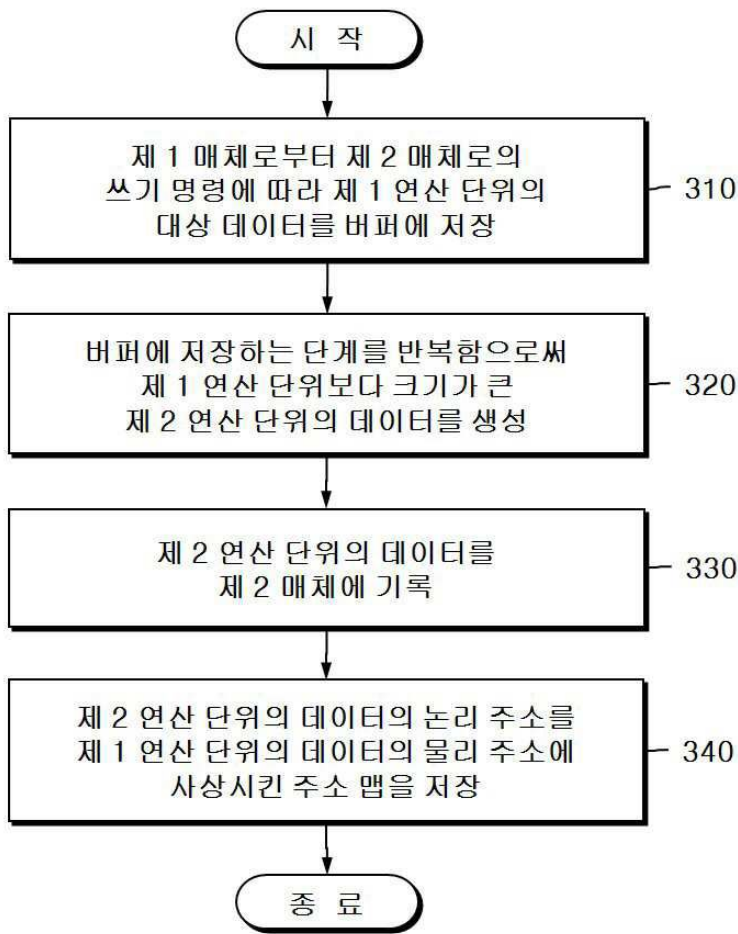
도면1



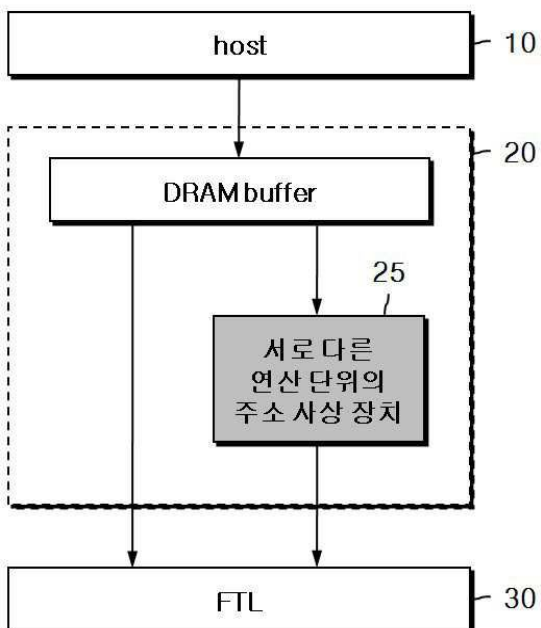
도면2



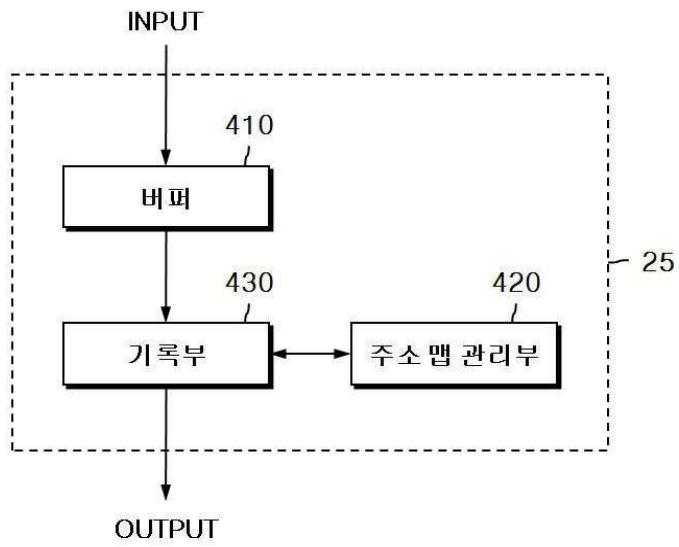
도면3



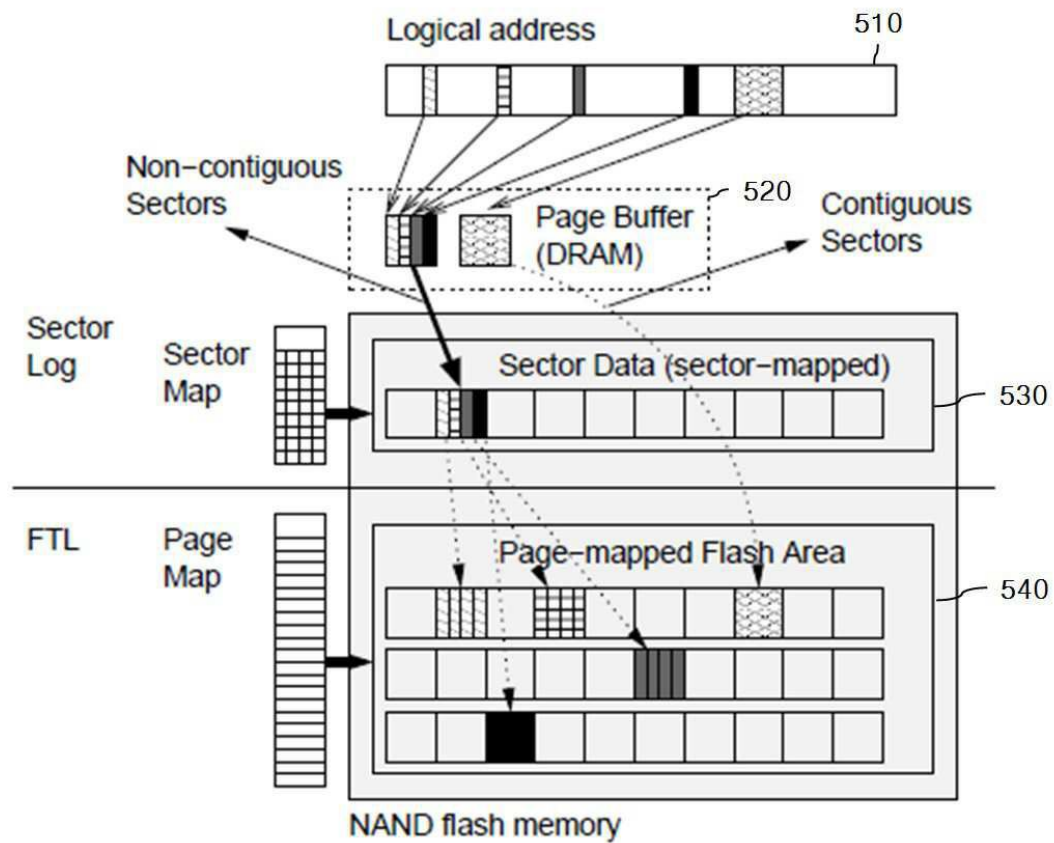
도면4a



도면4b

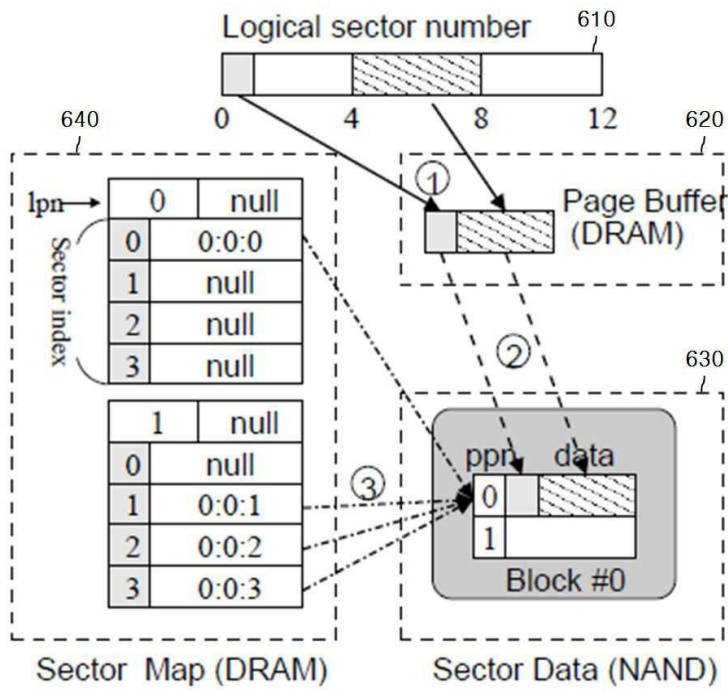


도면5

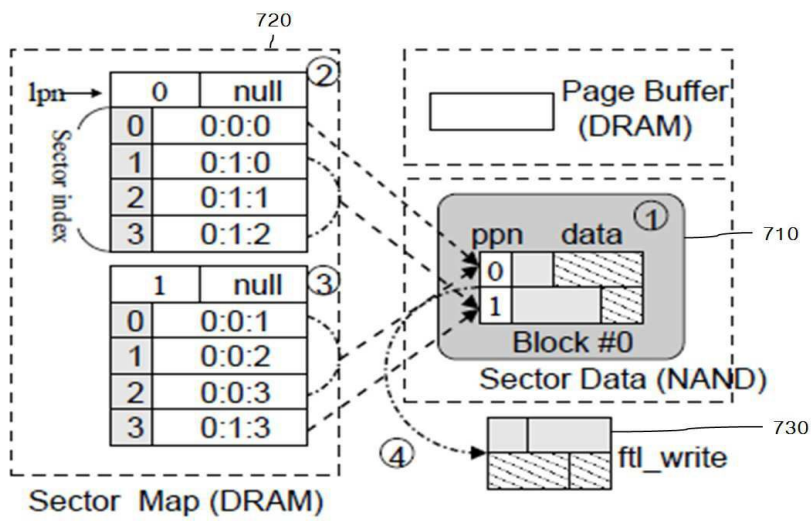




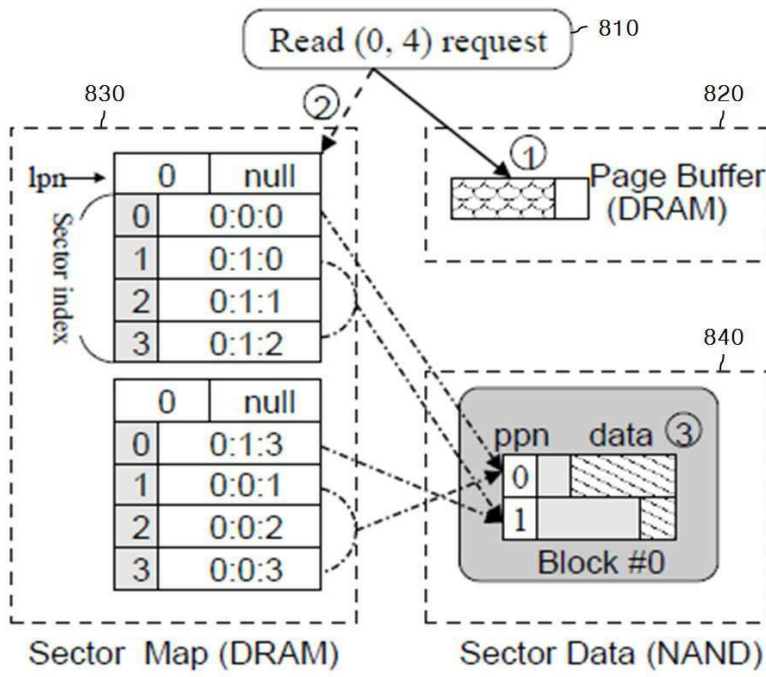
도면6



도면7



도면8



도면9

