



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2012년05월23일
(11) 등록번호 10-1143397
(24) 등록일자 2012년04월30일

(51) 국제특허분류(Int. Cl.)
G11C 8/12 (2006.01) G11C 16/06 (2006.01)
(21) 출원번호 10-2009-0069607
(22) 출원일자 2009년07월29일
심사청구일자 2009년07월29일
(65) 공개번호 10-2011-0012062
(43) 공개일자 2011년02월09일
(56) 선행기술조사문헌
KR100389867 B1

(73) 특허권자
에스케이하이닉스 주식회사
경기도 이천시 부발읍 경충대로 2091
(72) 발명자
양운모
경기도 여주군 여주읍 점봉리
김경로
경기도 수원시 영통구 영통로200번길 156, 1003
동 601호 (망포동, 방죽마을영통뜨란채)
곽정순
경기도 하남시 덕산로 80, 한솔아파트 303동
1303호 (덕풍동)
(74) 대리인
김성남

전체 청구항 수 : 총 18 항

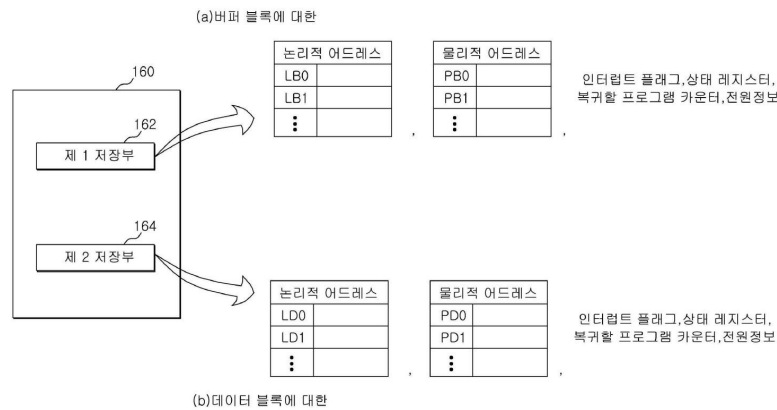
심사관 : 김중기

(54) 발명의 명칭 페이지 복사 발생 빈도를 줄이는 반도체 스토리지 시스템 및 그 제어 방법

(57) 요약

개시된 반도체 스토리지 시스템은, 메모리 영역의 메모리 블록을 데이터 블록 및 이에 대응되는 버퍼 블록으로 구분하고 병합 시점이 되면, 상기 데이터 블록 및 상기 버퍼 블록의 유효 페이지 수 및 프리 페이지(free page) 수를 각각 비교하여, 병합이 되는 타겟 블록을 상기 데이터 블록 및 버퍼 블록으로 선택할 수 있는 메모리 컨트롤러를 포함한다.

대표도



특허청구의 범위

청구항 1

메모리 영역의 메모리 블록을 데이터 블록 및 이에 대응되는 버퍼 블록으로 구분하고, 어떤 정보를 병합이 되는 타겟 블록에 라이트하는 병합 오퍼레이션(operation)을 수행하는 반도체 스토리지 시스템에 있어서,

병합 시, 상기 데이터 블록 및 상기 버퍼 블록의 유효 페이지 수 및 프리 페이지(free page) 수를 각각 비교하여, 그 비교결과에 따라 상기 데이터 블록 및 상기 버퍼 블록 중 어느 하나를 병합이 되는 상기 타겟 블록으로 선택할 수 있는 메모리 컨트롤러를 포함하는 반도체 스토리지 시스템.

청구항 2

제 1항에 있어서,

상기 메모리 컨트롤러는,

상기 데이터 블록에 존재하는 유효 페이지 수와 해당 버퍼 블록에 존재하는 유효 페이지 수를 카운트하고, 상기 데이터 블록에 존재하는 프리 페이지 수와 해당 버퍼 블록에 존재하는 프리 페이지 수를 카운트하여 카운트한 결과를 비교하는 블록 관리부; 및

상기 데이터 블록 및 상기 버퍼 블록의 각각의 전원 불량 상태를 체크한 결과를 상기 블록 관리부에 제공하는 전원 관리부를 포함하는 반도체 스토리지 시스템.

청구항 3

제 2항에 있어서,

상기 블록 관리부는,

상기 데이터 블록의 유효 페이지 수가 상기 버퍼 블록의 상기 프리 페이지 수 이상인지를 비교하고, 상기 버퍼 블록의 유효 페이지 수가 상기 데이터 블록의 상기 프리 페이지 수 이상인지를 비교하여, 병합의 소스 블록(source block)과 병합의 타겟 블록(target block)을 결정하는 반도체 스토리지 시스템.

청구항 4

제 3항에 있어서,

상기 블록 관리부는,

상기 전원 관리부의 체크 결과를 반영하여 상기 데이터 블록 및 상기 버퍼 블록의 프리 페이지를 카운트하는 반도체 스토리지 시스템.

청구항 5

제 3항에 있어서,

상기 블록 관리부는,

상기 각각의 비교 결과에 따라 상기 데이터 블록 및 상기 버퍼 블록외에 병합을 위한 새로운 버퍼 블록을 할당하는 것을 더 포함하는 반도체 스토리지 시스템.

청구항 6

어떤 정보를 병합이 되는 타겟 블록에 라이트하는 병합 오퍼레이션(operation)을 수행하는 반도체 스토리지 시스템에 있어서,

데이터 블록 및 이에 대응되는 버퍼 블록을 포함하는 메모리 영역;

상기 데이터 블록 및 상기 버퍼 블록의 맵핑 정보, 복귀할 프로그램 카운터 및 프로그램 레지스터 관련 정보를 임시 저장하는 버퍼부; 및

병합 시, 상기 버퍼부의 저장 결과를 이용하여 상기 데이터 블록의 프리 페이지 수를 카운트함으로써, 상기 데이터 블록의 상기 프리 페이지 수가 상기 버퍼 블록의 유효 페이지 수 이상이 되면 상기 데이터 블록을 병합의 상기 타겟 블록으로 설정하는 메모리 컨트롤러를 포함하는 메모리 영역을 포함하는 반도체 스토리지 시스템.

청구항 7

제 6항에 있어서,

상기 메모리 컨트롤러는,

상기 데이터 블록에 존재하는 유효 페이지 수와 해당 버퍼 블록에 존재하는 유효 페이지 수를 카운트하고, 상기 해당 버퍼 블록에 존재하는 프리 페이지 수가 상기 데이터 블록의 상기 유효 페이지 수를 만족하는지 여부를 판단하여 병합시 유효 페이지의 카피 빈도수를 조절하는 블록 관리부; 및

상기 데이터 블록 및 상기 버퍼 블록의 각각의 전원 불량 상태를 체크한 결과를 상기 블록 관리부에 제공하는 전원 관리부를 포함하는 반도체 스토리지 시스템.

청구항 8

제 7항에 있어서,

상기 블록 관리부는,

상기 데이터 블록의 유효 페이지 수가 상기 버퍼 블록의 상기 프리 페이지 수 이상인지를 비교하고, 상기 버퍼 블록의 유효 페이지 수가 상기 데이터 블록의 상기 프리 페이지 수 이상인지를 비교하여, 병합의 소스 블록과 병합의 타겟 블록을 결정하는 반도체 스토리지 시스템.

청구항 9

제 8항에 있어서,

상기 블록 관리부는,

상기 비교 결과에 따라, 상기 해당 버퍼 블록으로부터 상기 데이터 블록으로 카피할 유효 페이지 수가 상기 데이터 블록에서 상기 해당 버퍼 블록으로 카피할 유효 페이지 수보다 적으면 상기 버퍼 블록을 병합의 소스 블록으로 설정하여 상기 버퍼 블록의 상기 유효 페이지들을 상기 데이터 블록의 상기 프리 페이지에 카피하도록 제어하는 반도체 스토리지 시스템.

청구항 10

제 8항에 있어서,

상기 블록 관리부는,

상기 맵핑 정보에 따라 상기 데이터 블록의 최종 라이트 명령이 수행된 페이지를 감지하고, 상기 해당 페이지를 시작으로 잔여 페이지들을 스캐닝하여 상기 데이터 블록의 상기 프리 페이지를 카운트하는 반도체 스토리지 시스템.

청구항 11

제 10항에 있어서,

상기 블록 관리부는,

상기 전원 관리부의 체크 결과를 반영하여 상기 데이터 블록의 프리 페이지를 카운트할 때, 전원 불량이 발생한 페이지는 상기 프리 페이지에서 제외시키는 반도체 스토리지 시스템.

청구항 12

제 8항에 있어서,

상기 블록 관리부는,

상기 각각의 비교 결과에 따라 상기 데이터 블록 및 상기 버퍼 블록외에 병합을 위한 새로운 버퍼 블록을 할당하는 것을 더 포함하는 반도체 스토리지 시스템.

청구항 13

제 6항에 있어서,

상기 메모리 영역에는 병합이 완료된 상기 병합의 타겟 블록에 대한 갱신된 맵핑 정보가 저장되는 반도체 스토리지 시스템.

청구항 14

어떤 정보를 병합이 되는 타겟 블록에 라이트하는 병합 오퍼레이션(operation)을 제어하기 위한 반도체 스토리지 시스템의 제어 방법에 있어서,

병합 시, 데이터 블록의 유효 페이지 수, 버퍼 블록의 유효 페이지 수 및 버퍼 블록의 프리 페이지 수를 교차 비교하는 단계;

상기 비교 결과에 따라, 상기 버퍼 블록을 병합의 타겟 블록으로 설정하거나 상기 데이터 블록을 병합의 상기 타겟 블록으로 설정하거나 상기 데이터 블록 및 상기 버퍼 블록을 제외한 새로운 버퍼 블록을 병합의 타겟 블록으로 설정하여 병합을 수행하는 단계를 포함하는 반도체 스토리지 시스템의 제어 방법.

청구항 15

제 14항에 있어서,

상기 교차 비교하는 단계는,

상기 데이터 블록의 유효 페이지 수가 상기 버퍼 블록의 유효 페이지 수보다 많고, 상기 데이터 블록의 프리 페이지 수가 상기 버퍼 블록의 유효 페이지 수 이상인지 비교하는 단계를 포함하는 반도체 스토리지 시스템의 제어 방법.

청구항 16

제 15항에 있어서,

상기 데이터 블록을 병합의 타겟 블록으로 설정하는 것은,

상기 비교 결과에 따라, 상기 데이터 블록의 유효 페이지 수가 상기 버퍼 블록의 유효 페이지 수보다 많고, 상기 데이터 블록의 프리 페이지 수가 상기 버퍼 블록의 유효 페이지 수 이상이면 상기 버퍼 블록의 상기 유효 페이지들을 상기 데이터 블록으로 카피하도록 제어하는 반도체 스토리지 시스템의 제어 방법.

청구항 17

제 15항에 있어서,

상기 데이터 블록을 병합의 타겟 블록으로 설정하는 것은,

상기 비교 결과에 따라, 상기 데이터 블록의 유효 페이지 수가 상기 버퍼 블록의 유효 페이지 수보다 적지만, 상기 버퍼 블록의 상기 프리 페이지 수는 상기 데이터 블록의 유효 페이지 수 미만이고 상기 데이터 블록의 프리 페이지 수가 상기 버퍼 블록의 유효 페이지 수 이상이면 상기 버퍼 블록의 상기 유효 페이지들을 상기 데이터 블록으로 카피하도록 제어하는 반도체 스토리지 시스템의 제어 방법.

청구항 18

제 15항에 있어서,

상기 새로운 버퍼 블록을 할당하는 것은,

상기 비교 결과에 따라, 상기 데이터 블록의 상기 프리 페이지 수도 상기 버퍼 블록의 유효 페이지 수 미만이고 상기 버퍼 블록의 상기 프리 페이지 수도 상기 데이터 블록의 유효 페이지 수 미만일 때 수행하는 반도체 스토리지 시스템의 제어 방법.

명세서

발명의 상세한 설명

기술 분야

[0001] 본 발명은 반도체 스토리지 시스템 및 그 제어 방법에 관한 것으로서, 보다 구체적으로는 블록의 병합을 제어하는 반도체 스토리지 시스템 및 그 제어 방법에 관한 것이다.

배경 기술

[0002] 최근 멀티 미디어 기술의 발달로 데이터가 점차 대용량화되고 있다. 또한, 컴퓨터가 아닌 디지털 카메라, 디지털 캠코더, 디지털 녹음기 등 멀티 미디어 데이터의 저장을 필요로 하는 기기가 다양화되면서 멀티 미디어 데이터를 저장하는 비휘발성 저장 장치에 대한 관심도 증가하고 있다. 이러한 비휘발성 저장 장치 중 대표적인 것으로 NAND 플래시 메모리가 있다. 잘 알려진 바와 같이, 최근에는 PC(Personal Computer)에서 HDD(Hard Disk Drive)를 대신하여 낸드(NAND) 플래시 메모리를 사용한 SSD(Solid State Drive)가 출시되고 있고, 급속도로 HDD 시장을 잠식할 것으로 전망되고 있다.

[0003] 통상적으로 이러한 SSD와 같은 반도체 스토리지 시스템의 메모리 영역의 어드레스 �핑 방법으로는 블록 단위 �핑 방법, 페이지 단위 �핑 방법, 이들 블록 단위 �핑과 페이지 단위 �핑을 함께 사용하는 하이브리드(hybrid) �핑 방법이 있다. 블록 단위 �핑은 물리적인 블록 어드레스와 논리적인 블록 어드레스를 �핑한다. 페이지 �핑은 섹터 �핑이라고 하며, 페이지 단위로 �핑 테이블을 만들어 물리적인 페이지 어드레스와 논리적인 섹터 어드레스를 �핑하는 방식이다. 페이지 �핑을 수행하면 �핑 테이블 검색에 의해 낸드 플래시 메모리의 실질적인 페이지 어드레스를 알 수 있어 고속으로 동작할 수 있는 장점이 있다. 하지만, 대용량화에 따른 낸드 플래시 내 이러한 �핑 저장 용량의 증가는 피할 수 없다. 블록 �핑은 낸드 플래시 메모리의 블록 수에 대응되어 테이블을 낸드 플래시내에 저장하여 사용하기 때문에 페이지 �핑 방식에 비해 상대적으로 �핑 저장 용량을 줄일 수 있는 것이 특징이다. 그러나, 블록 �핑은 덮어쓰기(overwrite) 과정에서 비효율적이다. 하이브리드 �핑은 기본적으로 블록 �핑을 사용하여 �핑 저장 용량을 줄이고, 버퍼 블록(또는 로그 블록이라고도 함)에 대해서는 페이지 �핑을 사용하는 방식이다. 여기서 버퍼 블록을 사용할 수 있는 �핑 방식은 주로 페이지 �핑 방식과 하이브리드 �핑 방식이다. 설명의 편의상 하이브리드 �핑 방식을 기준으로 설명한다.

[0004] 버퍼 블록 �핑 방법은 메모리 영역의 메모리 블록을 데이터 블록과 버퍼 블록으로 구분하고, 논리적 어드레스에 따라 지정된 데이터 블록보다 우선적으로, �핑을 통해 버퍼 블록을 할당하여 데이터를 저장하는 것이다.

[0005] 따라서, 호스트의 쓰기 요청에 응답하여, 해당 데이터 블록과 연관되어 �핑된 버퍼 블록의 여유 공간이 부족할 경우, 해당 데이터 블록에 여유 공간이 충분하다 하더라도, FTL(Flash Translation Layer) 알고리즘에 의해 새로운 버퍼 블록을 할당하여 쓰기 연산을 하게 된다. 이와 같이, 항상 데이터 블록에 대한 명령을, 상기 데이터 블록과 연관되어 �핑된 버퍼 블록에서 수행되도록 제어함으로써 블록 병합 또는 불필요 정보 수집(Garbage Collection) 성능을 저하시킬 수 있다. 그리하여, 최근에는 보다 효율적인 블록 관리가 절실히 요구되고 있다.

발명의 내용

해결하고자하는 과제

[0006] 본 발명의 기술적 과제는 블록 병합을 제어하는 반도체 스토리지 시스템을 제공하는 것이다.

[0007] 본 발명의 기술적 과제는 블록 병합을 제어하는 반도체 스토리지 시스템의 제어 방법을 제공하는 것이다.

과제 해결수단

[0008] 본 발명의 기술적 과제를 달성하기 위하여, 본 발명의 일 실시예에 따른 반도체 스토리지 시스템은, 메모리 영역의 메모리 블록을 데이터 블록 및 이에 대응되는 버퍼 블록으로 구분하고 병합 시점이 되면, 상기 데이터 블록 및 상기 버퍼 블록의 유효 페이지 수 및 프리 페이지(free page) 수를 각각 비교하여, 병합이 되는 타겟

블록을 상기 데이터 블록 및 버퍼 블록으로 선택할 수 있는 메모리 컨트롤러를 포함한다.

[0009] 본 발명의 기술적 과제를 달성하기 위하여, 본 발명의 다른 실시예에 따른 반도체 스토리지 시스템은, 데이터 블록 및 이에 대응되는 버퍼 블록을 포함하는 메모리 영역, 상기 데이터 블록 및 상기 버퍼 블록의 맵핑 정보, 복귀할 프로그램 카운터 및 프로그램 레지스터 관련 정보를 임시 저장하는 버퍼부 및 병합 시점이 되면, 상기 버퍼부의 저장 결과를 이용하여 상기 데이터 블록의 프리 페이지 수를 카운트함으로써, 상기 데이터 블록의 상기 프리 페이지 수가 상기 버퍼 블록의 유효 페이지 수 이상이 되면 상기 데이터 블록을 병합의 타겟 블록으로 설정하는 메모리 컨트롤러를 포함하는 메모리 영역을 포함한다.

[0010] 본 발명의 다른 기술적 과제를 달성하기 위하여, 본 발명의 일 실시예에 따른 반도체 스토리지 시스템의 제어 방법은, 병합 시점이 되면, 데이터 블록의 유효 페이지 수, 버퍼 블록의 유효 페이지 수 및 버퍼 블록의 프리 페이지 수를 교차 비교하는 단계 및 상기 비교 결과에 따라, 상기 버퍼 블록을 병합의 타겟 블록으로 설정하거나 상기 데이터 블록을 병합의 타겟 블록으로 설정하거나 상기 데이터 블록 및 상기 버퍼 블록을 제외한 새로운 버퍼 블록을 병합의 타겟 블록으로 설정하여 병합을 수행하는 단계를 포함한다.

효 과

[0011] 본 발명의 일 실시예에 따르면, 데이터 블록과 맵핑된 버퍼 블록간의 여유 공간을 비교하여 여유 공간이 더 많은 블록으로 데이터가 이동되도록 제어할 수 있다. 이로써, 여유 공간의 자원 효율을 높이는 한편, 병합시 카피의 빈도수를 줄이므로 병합 효율이 높아지도록 제어할 수 있다.

발명의 실시를 위한 구체적인 내용

[0012] 이하, 본 발명의 실시예에 의한 본 발명의 일 실시예에 따른 반도체 스토리지 시스템 및 제어 방법을 설명하기 위한 블록도 또는 플로우 차트에 대한 도면들을 참고하여 본 발명에 대해 설명하도록 한다.

[0013] 또한, 각 블록도는 특정된 논리적 기능(들)을 실행하기 위한 하나 이상의 실행 가능한 인스트럭션들을 포함하는 모듈, 세그먼트 또는 코드의 일부를 나타낼 수 있다. 또, 몇 가지 대체 실행예들에서는 블록들에서 언급된 기능들이 순서를 벗어나서 발생하는 것도 가능함을 주목해야 한다. 예컨대, 잇달아 도시되어 있는 두 개의 블록들은 사실 실질적으로 동시에 수행되는 것도 가능하고 또는 그 블록들이 때때로 해당하는 기능에 따라 역순으로 수행되는 것도 가능하다.

[0014] 이 때, 본 실시예에서 사용되는 '~부'라는 용어는 소프트웨어 또는 FPGA또는 ASIC과 같은 하드웨어 구성요소를 의미하며, '~부'는 어떤 역할들을 수행한다. 그렇지만 '~부'는 소프트웨어 또는 하드웨어에 한정되는 의미는 아니다. '~부'는 어드레싱할 수 있는 저장 매체에 있도록 구성될 수도 있고 하나 또는 그 이상의 프로세서들을 재생시키도록 구성될 수도 있다.

[0015] 따라서, 일 예로서 '~부'는 소프트웨어 구성요소들, 객체지향 소프트웨어 구성요소들, 클래스 구성요소들 및 태스크 구성 요소들과 같은 구성요소들과, 프로세스들, 함수들, 속성들, 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 드라이버들, 펌웨어, 마이크로코드, 회로, 데이터, 데이터베이스, 데이터 구조들, 테이블들, 어레이들, 및 변수들을 포함한다. 구성 요소들과 '~부'들 안에서 제공되는 기능은 더 작은 수의 구성요소들 및 '~부'들로 결합되거나 추가적인 구성요소들과 '~부'들로 더 분리될 수 있다. 뿐만 아니라, 구성요소들 및 '~부'들은 디바이스 또는 보안 멀티미디어카드 내의 하나 또는 그 이상의 CPU들을 재생시키도록 구현될 수도 있다.

[0016] 먼저, 도 1을 참조하여, 본 발명의 일 실시예에 따른 반도체 스토리지 시스템에 대해서 설명하기로 한다.

[0017] 도 1은 본 발명의 일 실시예에 따른 반도체 스토리지 시스템(100)의 블록도이고 도 2는 도 1에 따른 정보 저장 영역(160)의 개념적 블록도이다. 여기서, 반도체 스토리지 시스템(100)은 낸드 플래시 메모리를 이용한 시스템으로 예시한다.

[0018] 도 1 및 도 2를 참조하면, 반도체 스토리지 시스템(100)은 호스트 인터페이스(110), 버퍼부(120), MCU(130), 메모리 컨트롤러(140) 및 메모리 영역(150)을 포함한다.

[0019] 우선, 호스트 인터페이스(110)는 버퍼부(120)와 연결되며, 외부 호스트 (미도시)와 호스트 인터페이스(110)간에 제어 명령, 어드레스 신호 및 데이터 신호를 송수신한다. 호스트 인터페이스(110)와 외부 호스트 (미도시)간의 인터페이스 방식은 직렬 ATA(Serial Advanced Technology Attachment;SATA), 병렬 ATA(Parallel Advanced Technology attachment;PATA), SCSI, Express Card 및 PCI-Express 방식 중 어느 하나일 수 있으며

제한되지 않는다.

- [0020] 본 발명의 일 실시예에 따른 버퍼부(120)는 정보 저장 영역(160)을 포함한다. 그리하여, 버퍼부(120)는 호스트 인터페이스(110)로부터의 출력 신호들을 버퍼링하거나, 메모리 영역(150)으로부터의 데이터들을 버퍼링한다. 또한, 버퍼부(120)는 MCU(130)로부터의 출력 신호들을 버퍼링함으로써, 호스트 인터페이스(110) 및 메모리 컨트롤러(140)에 제공할 수 있다. 이러한 버퍼부(120)는 SRAM(Static Random Access Memory)을 이용한 버퍼로 예시한다.
- [0021] 한편, 도 2에 도시된 바와 같이, 정보 저장 영역(160)은 제 1 저장부(162) 및 제 2 저장부(164)를 포함한다.
- [0022] 제 1 저장부(162)는 버퍼 블록에 대한 논리적 어드레스 및 물리적 어드레스 맵핑 테이블, 인터럽트 플래그(interrupt flag), 상태 레지스터(status register), 복귀할 프로그램 카운터(counter)등을 임시 저장한다. 본 발명의 일 실시예에 따른 제 2 저장부(164)는 데이터 블록에 대한 논리적 어드레스 및 물리적 어드레스 맵핑 테이블, 인터럽트 플래그(interrupt flag), 상태 레지스터(status register), 복귀할 프로그램 카운터등을 저장한다.
- [0023] 통상의 반도체 스토리지 시스템에서는 버퍼 블록에 대한 맵핑 정보 및 제어 관련 정보들은 버퍼부(120)에 임시 저장하였으나, 데이터 블록에 대한 맵핑 정보는 최종 병합이 된 후 오직 메모리에 저장하였다. 즉, 종래의 반도체 스토리지 시스템에서는 버퍼 블록의 병합(merge) 과정에 의해 최종적으로 완성된 버퍼 블록을 새로운 데이터 블록으로 설정하여, 해당 버퍼 블록의 데이터들을 메모리에 저장하였다. 다시 말하면, 병합 타겟 블록이 된 버퍼 블록을 데이터 블록으로서 처리하고, 이에 대한 갱신된 맵핑 정보를 임시 저장 영역(휘발성 메모리)이 아닌 메인 메모리 셀에 저장하였던 것이다.
- [0024] 하지만, 본 발명의 일 실시예에 따르면, 데이터 블록의 맵핑 정보 및 제어 관련 정보등도 버퍼 블록과 같이 제 2 저장부(164)에 임시 저장하도록 제어한다. 그리하여, 데이터 블록도 소스 블록이 아닌 병합 타겟 블록이 될 수 있도록 제어할 수 있다. 이에 대한 자세한 설명은 이후에 후술하기로 한다.
- [0025] MCU(Micro Control Unit; 130)는 호스트 인터페이스(110)간에 제어 명령, 어드레스 신호 및 데이터 신호등을 송수신하거나, 이러한 신호들에 의해 메모리 컨트롤러(140)를 제어하기도 한다. 이러한 MCU(130)는 메모리 영역(150)의 어드레스 맵핑을 제어할 수 있다.
- [0026] 메모리 컨트롤러(140)는 메모리 영역(150)의 복수개의 낸드 플래시 메모리 소자 중 소정 낸드 플래시 메모리 소자(미도시)를 선택하고, 라이트, 삭제 또는 리드 명령을 제공한다. 이러한 메모리 컨트롤러(140)는 블록 관리부(170) 및 전원 관리부(180)를 포함한다.
- [0027] 구체적으로, 블록 관리부(170)는 데이터 블록에 존재하는 유효 페이지 수와 해당 버퍼 블록에 존재하는 유효 페이지 수를 비교하거나, 각각의 데이터 블록 및 버퍼 블록에 존재하는 프리(free) 페이지 수를 비교하여 카피 빈도가 적게 발생하는 형태로 데이터가 쓰여지도록 블록의 병합을 관리할 수 있다. 즉, 본 발명의 일 실시예에 따른 블록 관리부(170)는 데이터 블록으로부터 버퍼 블록으로만의 데이터 이동을 가능하게 했던 종래의 단방향 제어와는 달리, 데이터 블록과 해당 버퍼 블록간의 쌍방향(interactive) 데이터의 이동을 가능하도록 제어할 수 있다. 이에 따라, 블록 관리부(170)는 데이터 블록과 버퍼 블록간에 쌍방향으로의 데이터 이동이 가능하도록, 데이터 블록도 버퍼 블록처럼 관리한다. 즉, 블록 관리부(170)는 데이터 블록에 대해 전술한 데이터 블록의 맵핑 테이블, 프로그램 카운터등을 체크하여 제 2 저장부(164)에 저장하도록 제어한다.
- [0028] 전원 관리부(180)는 반도체 스토리지 시스템의 전원 상태의 불량 유무를 체크한다. 여기서, 본 발명의 일 실시예에 따른 전원 관리부(180)는 버퍼 블록뿐 아니라 데이터 블록에 대해서도 전원의 불량 유무를 체크한 정보를 저장하도록 한다. 그리하여, 전원 관리부(180)는 버퍼 블록 및 데이터 블록의 병합 시점에 해당 블록의 전원 불량 체크 정보를 블록 관리부(170)에 제공하여, 프리 페이지 개수 판단을 정확히 하도록 한다.
- [0029] 통상, 플래시 메모리는 한번 쓰여진(write) 페이지에 다른 데이터를 덮어쓸 수 없으며, 반드시 블록 단위의 삭제(erase)를 한 후에 새로운 데이터를 기록할 수 있다.
- [0030] 즉, 플래시 메모리의 물리적인 특성으로 인하여 반드시 쓰기 전 삭제(erase-before-write) 연산을 실시한다. 그러므로 섹터에 덮어쓰기(overwrite) 연산을 수행할 수 있는 하드 디스크에 비하여 동일한 I/O에 대해 더 많은 소요 시간이 걸릴 수 있다. 또한 플래시 메모리의 블록은 10만 번 정도의 삭제 연산을 수행하면 더 이상 삭제 연산을 수행할 수 없다. 따라서 삭제 연산은 플래시 메모리의 수명을 단축시키는 요소(factor)이다. 이러한 특성으로 인해 플래시 메모리의 특정 섹터에 대한 쓰기 연산을 수행하면 플래시에서는 비어있거나 삭제 연산의 횟수가 적은 블록을 할당하여 쓰기 연산을 수행한다. 전술한 바와 같이, 이러한 알고리즘(algorithm)을

FTL(Flash Translation Layer)이라 부른다.

- [0031] FTL은 파일 시스템의 섹터 번호, 예컨대 논리적 섹터 번호를 저장 영역인 플래시 메모리의 섹터 번호, 예컨대 물리적 섹터 번호로 맵핑하는 기능을 수행한다. FTL 알고리즘은 이러한 삭제 연산을 가능하면 적게 해야 할 것이고, 가능하면 특정한 어느 하나의 블록에 집중 되지 않도록 해야 한다.
- [0032] 한편, FTL알고리즘에서 사용되는 맵핑은 섹터(sector) 맵핑, 블록 맵핑, 하이브리드(hybrid) 맵핑으로 구분할 수 있다. 섹터 맵핑은 섹터 단위로 맵핑 테이블을 만들어 물리적인 섹터와 논리적인 섹터를 맵핑하는 것이다. 또한, 블록 맵핑은 물리적인 블록과 논리적인 블록을 맵핑하는 것이다. 하이브리드 맵핑은 두가지 맵핑을 동시에 사용하는 것으로 두 가지 맵핑의 장점을 취할 수 있다. 이러한 맵핑 테이블은 컨트롤러의 RAM에 저장된다. 또한, 각 알고리즘은 내부적으로 기본적인 읽기/쓰기/삭제 연산 외에 필요에 따라 교환(switch)이나 병합(merge), 할당(allocation) 연산 등을 포함한다. 여기서 교환 연산은 블록의 내용을 변화시키지 않은 채 원래 블록을 소거하고, 블록 맵핑 테이블의 값만 바꾸어주는 연산이다. 병합 연산은 두 개의 블록에서 최신 정보만 복사하여 새로운 블록에 쓰고, 맵핑 테이블을 새로운 블록으로 바꾸어 준 다음 병합에 사용된 두 개의 블록을 소거하는 연산이다. 할당 연산은 빈 블록이 필요할 경우 빈 블록을 탐색 및 반환하는 연산이다.
- [0033] 이러한 플래시 메모리를 채용하는 반도체 스토리지 시스템에서는, 이미 페이지에 기록된 데이터의 변경이 있을 때는 새로운 데이터를 다른 페이지에 기록하고, 이전의 데이터는 유효하지 않은(invalid) 영역으로 관리한다. 이에 따라, 병합 또는 불필요 정보 수집(Garbage Collection) 과정이 요구된다. 불필요 정보 수집 과정은 특정 블록 내의 유효한(valid) 페이지만을 다른 블록에 복사(copy)해 두고, 해당 블록은 삭제(erase)함으로써 다시 사용할 수 있도록 한다. 종래에는, 데이터 블록의 맵핑 방식에 따라 할당된 버퍼 블록의 여유 공간이 부족하다고 판단되면, 새로운 버퍼 블록을 할당하도록 하였다. 간단히 말하면, 종래에는 해당 데이터 블록에 대한 병합은 반드시 버퍼 블록(새롭게 할당된 버퍼 블록 포함)에서만 수행될 수 있었다.
- [0034] 그러나, 본 발명의 일 실시예에 따르면, 전술한 바와 같이, 데이터 블록과 해당 버퍼 블록간의 여유 공간을 비교하여 여유 공간이 더 많은 블록으로 데이터가 이동되도록 제어할 수 있다. 이로써, 여유 공간의 자원 효율을 높이는 한편, 병합시 커피의 빈도수를 줄이므로 병합 효율이 높아지도록 제어할 수 있다.
- [0035] 메모리 영역(150)은 메모리 컨트롤러(140)에 제어되어 데이터의 라이트, 삭제 및 리드 동작이 수행된다.
- [0036] 이때, 메모리 영역(150)내 메모리 블록(미도시)은 데이터 블록 및 버퍼 블록을 포함하는 것으로 설정할 수 있다. 여기서, 버퍼 블록은 일종의 라이트용 버퍼로서, 라이트 명령시 데이터 블록에 저장될 데이터를 대응되는 버퍼 블록내에 저장한다.
- [0037] 이후의 도 3 내지 도 6을 참조하여, 본 발명의 일 실시예에 따른 반도체 스토리지 시스템의 동작을 설명하기로 한다.
- [0038] 도 3 내지 도 6은 본 발명의 일 실시예에 따라 다양한 데이터의 병합 과정을 도시한 개념도들이다.
- [0039] 우선, 도 3을 설명하면, 버퍼 블록의 유효 페이지 수가 데이터 블록의 페이지 수보다 적고, 데이터 블록의 프리 페이지 수는 버퍼 블록의 유효 페이지 수보다 많은 경우이다.
- [0040] <실험예 1>
- [0041] 우선, 각각의 데이터 블록과 버퍼 블록은 복수개의 페이지를 포함하는 것으로 예시한다.
- [0042] 병합 시점이 되면, 데이터 블록의 유효 페이지와 버퍼 블록의 유효 페이지를 하나의 영역에 함께 저장해야 한다.
- [0043] 종래에는 데이터 블록의 유효(valid) 페이지가 일단 버퍼 블록의 프리 페이지에 저장될 수 있는지 여부를 판단하였다. 실험예 1에서와 같이, 데이터 블록의 유효 페이지 수가 4이고 버퍼 블록의 프리 페이지 수가 2이라면, 종래 기술로는 데이터 블록의 유효 페이지는 현재 맵핑된 버퍼 블록의 프리 페이지에 저장될 수 없으므로 새로운 버퍼 블록을 할당해야 했었다. 그 뿐 아니라, 두 블록(데이터 블록 및 이와 맵핑된 버퍼 블록)으로부터 정보들을 병합해야 하므로, 각 유효 페이지 숫자만큼 전송 시간 및 라이트 시간은 증대할 수 있었다. 즉, 종래에는 병합시 많은 시간이 소요되었었다.
- [0044] 하지만, 본 발명의 일 실시예에 따르면, 블록 관리부(도 1의 170)에서 우선 데이터 블록의 유효 페이지 수와 버퍼 블록의 유효 페이지 수를 일단 비교한다. 이를 위해, 종래와 달리 본 발명의 일 실시예에서는 데이터 블록에 대해서도 스캐닝(scanning)을 실시한다. 즉, 제 2 저장 영역(도 2의 164 참조)의 정보들을 참조하여, 데

이터 블록을 스캐닝을 하면서 데이터 블록의 나머지 페이지들을 체크한다.

- [0045] 구체적으로, 해당 데이터 블록의 스캐닝은 전체적인 데이터 블록을 스캐닝이 아니라, 데이터 블록에서 라이트 명령이 가장 마지막으로 수행되었던 페이지 정보(예컨대 복귀할 프로그램 카운터)를 이용하여, 그 페이지로부터 마지막 페이지까지의 남은 페이지 수들을 스캐닝하여 총합산 하도록 한다. 한편, 남은 페이지들중에는 유효하지 않은 페이지들도 포함될 수 있다.
- [0046] 여기서 유효하지 않은 페이지들에는 전원 불량 발생 페이지를 포함한다. 전원 전원 관리부(도 1의 180 참조)에서 전원 불량 발생 정보를 저장하므로, 스캐닝시 전원 불량 발생 페이지를 알 수 있다. 따라서, 전원이 불량했던 페이지는 데이터 저장에 적합하지 않으므로 유효하지 않은 영역으로 구분하고, 해당 데이터 블록의 프리 페이지에는 참여하지 못하도록 제어한다.
- [0047] 실험예 1에서는 데이터 블록의 유효 페이지 수는 4이고 버퍼 블록의 프리 페이지 수는 2이다.
- [0048] 그러나, 실질적으로 유효 페이지 수가 더 적은 블록이 병합 소스 블록이 되고 유효 페이지 수가 더 많은 블록이 타겟 블록이 되도록 결정하는 것은, 각각의 블록의 여유 공간, 즉 프리 페이지 수이다.
- [0049] 즉, 데이터 블록의 유효 페이지, 즉 데이터로서 저장되어야 할 페이지 수는 4개이고, 데이터 블록의 프리 페이지는 3이다.
- [0050] 한편, 맵핑된 버퍼 블록의 유효 페이지의 수는 3이고, 프리 페이지의 수는 2이다.
- [0051] 이 경우, 본 발명의 일 실시예에 따르면 블록 관리부(도 1의 170 참조)는 버퍼 블록의 유효 페이지수가 일단 데이터 블록의 유효 페이지 수보다 적으므로, 데이터 블록으로의 병합을 고려할 수 있다. 이것이 가능하려면 데이터 블록의 프리 페이지의 수가 버퍼 블록의 유효 페이지 수 이상이어야 하는데, 실험예 1에서의 데이터 블록의 프리 페이지의 수가 3이므로, 이러한 조건을 만족한다.
- [0052] 그리하여, 버퍼 블록의 3개의 유효 페이지를 데이터 블록의 프리 페이지에 카피함으로써 병합을 완성할 수 있다.
- [0053] 이 때, 병합이 완료된 블록인 데이터 블록의 맵핑 정보는 블록 관리부(도 1의 170 참조)에 의해 최종적으로는 메모리 영역(도 1의 150 참조)에 저장된다.
- [0054] 다음으로, 도 4의 경우는 버퍼 블록의 유효 페이지 수가 데이터 블록의 페이지 수보다 많지만, 데이터 블록의 프리 페이지 수가 버퍼 블록의 유효 페이지 수보다 많은 경우이다.
- [0055] <실험예 2>
- [0056] 병합 시점이 되어, 데이터 블록의 유효 페이지와 버퍼 블록의 유효 페이지를 하나의 영역에 함께 저장해야 하는 상황은 실험예 1과 동일한 것으로 예시한다.
- [0057] 역시, 블록 관리부(도 1의 170 참조)에서 우선 데이터 블록의 유효 페이지 수와 버퍼 블록의 유효 페이지 수를 일단 비교한다. 전술한 바와 같이, 본 발명의 일 실시예에서는 데이터 블록에 대해서도 스캐닝(scanning)을 실시한다.
- [0058] 스캐닝 결과, 실험예 2에서는 데이터 블록의 유효 페이지 수가 3이고 버퍼 블록의 유효 페이지 수는 4이다. 하지만, 버퍼 블록의 프리 페이지 수는 2이므로, 데이터 블록의 유효 페이지 수가 버퍼 블록의 유효 페이지 수보다 적다 하더라도, 데이터 블록의 유효 페이지들이 버퍼 블록으로 병합될 수 없음은 당연하다.
- [0059] 한편, 맵핑된 버퍼 블록의 유효 페이지의 수는 4이고, 프리 페이지의 수는 2이다.
- [0060] 이 경우에는 버퍼 블록의 유효 페이지의 수가 데이터 블록의 그것보다 많았지만, 데이터 블록으로의 병합이 가능할 수 있다. 즉, 데이터 블록의 페이지 수가 버퍼 블록의 유효 페이지 수와 동일하므로, 버퍼 블록의 유효 페이지들을 데이터 블록의 프리 페이지에 카피하는 것이 가능하다.
- [0061] 종래에는, 데이터 블록의 3개의 유효 페이지들을 버퍼 블록에 카피할 수 없을 경우에는 새로운 버퍼 블록을 할당해서 병합을 시도해야 했었다. 그러나, 본 발명의 일 실시예에서는 데이터 블록이라 하더라도, 실질적으로 여유 공간이 더 많은 블록으로 판정될 경우에는 페이지의 카피 빈도수가 낮아지도록 버퍼 블록의 유효 페이지들을 데이터 블록으로 카피하여 병합을 실시한다.
- [0062] 이 때, 병합이 완료된 블록인 데이터 블록의 맵핑 정보는 블록 관리부(도 1의 170 참조)에 의해 최종적으로는

메모리 영역(도 1의 150 참조)에 저장된다.

[0063] 도 5는 버퍼 블록의 유효 페이지 수가 데이터 블록의 페이지 수보다 많고, 데이터 블록의 프리 페이지 수가 버퍼 블록의 유효 페이지 수보다 많은 경우이다.

[0064] <실험예 3>

[0065] 병합 시점이 되어, 데이터 블록의 유효 페이지와 버퍼 블록의 유효 페이지를 하나의 영역에 함께 저장해야 하는 상황은 실험예 1 및 2와 동일한 것으로 예시한다.

[0066] 역시, 블록 관리부(도 1의 170 참조)에서 우선 데이터 블록의 유효 페이지 수와 버퍼 블록의 유효 페이지 수를 일단 비교한다. 전술한 바와 같이, 본 발명의 일 실시예에서는 데이터 블록에 대해서도 스캐닝(scanning)을 실시한다.

[0067] 스캐닝 결과, 실험예 3에서는 데이터 블록의 유효 페이지 수가 3이고 버퍼 블록의 유효 페이지 수는 4이다. 또한, 데이터 블록의 프리 페이지 수는 5이고, 버퍼 블록의 프리 페이지 수는 4이다. 물론, 이 경우에 버퍼 블록의 유효 페이지 수가 4이고, 데이터 블록의 프리 페이지 수가 5이므로, 버퍼 블록의 유효 페이지들이 데이터 블록으로 카피하는 것이 가능하기는 하나, 카피의 빈도수가 보다 적게 발생하는 경우를 생각해야 한다. 이 때는 데이터 블록의 유효 페이지들을 카피하기에 버퍼 블록의 프리 페이지의 수도 충분히 여유 있음을 판단해야 한다. 또한, 데이터 블록의 유효 페이지의 수가 버퍼 블록의 유효 페이지의 수보다 작으므로, 예컨대 3번의 카피 동작으로 병합이 완료될 수 있다. 즉, 본 발명의 일 실시예에 따른 블록 관리부(도 1의 170 참조)는 데이터 블록을 병합 타겟 블록으로서 이용되도록 제어할 수도 있고, 통상처럼 버퍼 블록을 병합 타겟 블록으로서 이용되도록 제어할 수도 있는데, 이들 타겟 병합 블록을 선정하는 기준은 각 블록의 유효 페이지 수와 상대 블록의 유효 페이지들을 카피할 수 있는 여유 공간의 존재 및 개수에 대한 판단이다.

[0068] 도 6은 데이터 블록의 유효 페이지 수가 버퍼 블록의 유효 페이지 수보다는 많고, 데이터 블록의 프리 페이지 수가 버퍼 블록의 프리 페이지 수보다는 작은 경우를 나타낸다.

[0069] <실험예 4>

[0070] 병합 시점에 데이터 블록을 스캐닝한 결과, 유효 페이지의 수는 4이고 프리 페이지의 수는 1이었다. 한편, 버퍼 블록의 유효 페이지의 수는 2이고 프리 페이지의 수는 4이다.

[0071] 각 블록의 프리 페이지의 수는 맵핑으로 연관된 상대 블록의 유효 페이지들의 수와 밀접한 관련이 있다.

[0072] 즉, 데이터 블록의 유효 페이지의 수가 더 많다고 하여도, 버퍼 블록의 유효 페이지들을 데이터 블록으로 병합을 실시하는 것은 불가능한 것은, 도 6에 도시된 바와 같이, 데이터 블록의 프리 페이지가 한 개뿐이기 때문이다.

[0073] 이럴 경우, 데이터 블록의 4개의 유효 페이지들을 버퍼 블록의 4개의 프리 페이지에 각각 카피함으로써 병합이 완성될 수 있다.

[0074] 만약, 병합 시점이 되었으나 상기의 실험예들과 다른 상황이 발생되면, 예컨대 데이터 블록의 유효 페이지가 버퍼 블록의 프리 페이지의 수보다 많고, 동시에 버퍼 블록의 유효 페이지의 수가 데이터 블록의 프리 페이지의 수보다 많다면, 통상의 방법과 같이 새로운 버퍼 블록을 할당하여 데이터 블록과 버퍼 블록의 유효 페이지들을 모두 카피하도록 한다.

[0075] 도 7은 도 1에 따른 반도체 스토리지 시스템의 동작을 나타낸 순서도이다.

[0076] 다시 도 1 내지 도 7을 참조하면, 병합 시점이 되면 블록 관리부(170)가 데이터 블록(DB)의 유효 페이지 수와 버퍼 블록(BB)의 유효 페이지 수, 버퍼 블록(BB)의 프리 페이지 수를 비교한다(S10).

[0077] 즉, 버퍼 블록(BB)의 유효 페이지 수가 데이터 블록(DB)의 유효 페이지 수보다 더 많으면서, 버퍼 블록(BB)의 프리 페이지 수가 데이터 블록의 유효 페이지 수 이상이면(Y), 데이터 블록(DB)의 유효 페이지를 버퍼 블록(BB)에 카피한다(S20).

[0078] 만약, 비교 결과 버퍼 블록(BB)의 유효 페이지 수가 데이터 블록(DB)의 유효 페이지 수보다 더 많으면서, 버퍼 블록(BB)의 프리 페이지 수가 데이터 블록의 유효 페이지 수 이상이 아니라면, 다시 구체적으로 각 조건들을 비교한다. 즉, 맵핑 정보에 저장된 마지막 물리적 페이지 정보에 기반한 데이터 블록(DB)의 프리 페이지의 수가 버퍼 블록의 유효 페이지 수 이상인지 비교한다(S30).

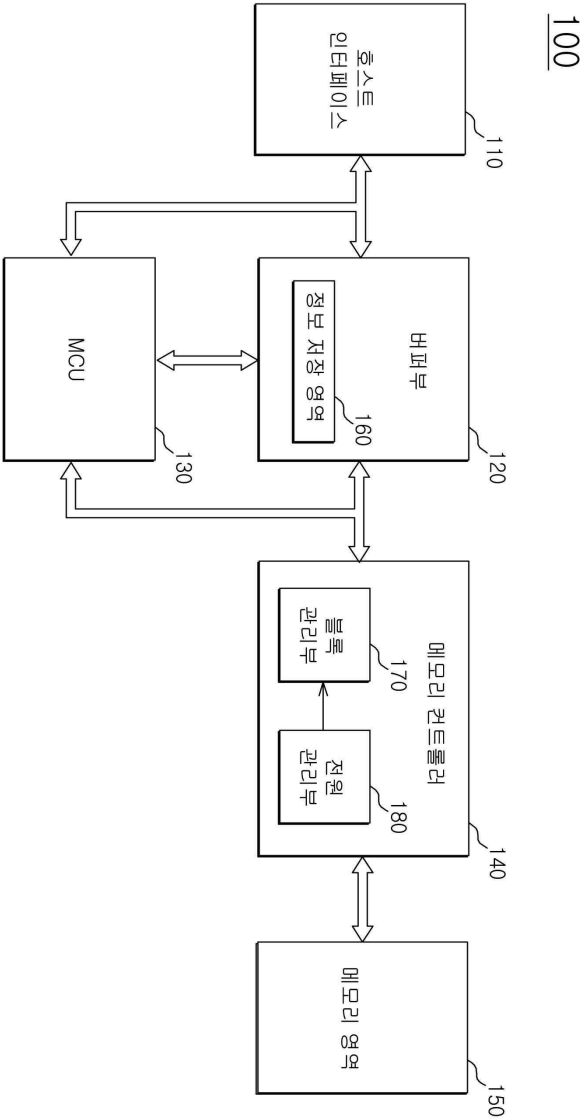
- [0079] 다시 말하면, 맵핑 정보에 저장되어 있는 데이터 블록(DB)에 대해 라이트 명령이 수행되었던 가장 마지막 물리적 페이지부터 스캐닝하여 남은 페이지의 수가 버퍼 블록(BB)의 유효 페이지 수 이상이 되는지 여부를 판단한다. 그리하여, 데이터 블록(DB)의 남은 페이지 수가 버퍼 블록(BB)의 유효 페이지 수 이상이 되면(Y), 전원 관리부(180)가 전원의 불량 발생 페이지의 여부를 체크하여 블록 관리부(170)에 그 결과를 제공하여, 블록 관리부(170)는 전원 불량 체크한 결과, 전원이 양호한 페이지, 즉 데이터 블록(DB)의 프리 페이지의 수가 버퍼 블록(BB)의 유효 페이지 수 이상인지 판단한다(S40).
- [0080] 만약, 전원 체크 후, 데이터 데이터 블록(DB)의 프리 페이지의 수가 버퍼 블록(BB)의 유효 페이지 수 이상이면(Y), 데이터 블록(BB)에 버퍼 블록(BB)의 유효 페이지를 카피하도록 제어한다(S50).
- [0081] 한편, 맵핑 정보에 저장된 마지막 물리적 페이지 정보에 기반한 데이터 블록(DB)의 프리 페이지의 수가 버퍼 블록의 유효 페이지 수 이상이 되지 않거나(S 30의 N 프로시저(procedure)) 전원 체크까지 해본 결과, 데이터 블록(DB)의 프리 페이지 수가 버퍼 블록(BB)의 유효 페이지 수보다 적었다면(S40의 N 프로시저), 이번에는 버퍼 블록(BB)의 프리 페이지의 수와 데이터 블록(DB)의 유효 페이지 수를 비교한다(S60).
- [0082] 그리하여, 버퍼 블록(BB)의 프리 페이지의 수가 데이터 블록(DB)의 유효 페이지 수 이상이 되면(Y), 버퍼 블록(BB)이 여유 공간이 많으므로, 데이터 블록(DB)의 유효 페이지를 버퍼 블록(BB)에 카피함으로써(S20) 병합을 실시한다.
- [0083] 그러나, 버퍼 블록(BB)의 프리 페이지의 수가 데이터 블록(DB)의 유효 페이지 수 미만이라면(S60의 N), 통상의 방법처럼 새로운 버퍼 블록을 할당(S70)하여 데이터 블록(DB)과 버퍼 블록(BB)의 유효 페이지들을 각각 카피하여 병합을 수행한다.
- [0084] 이 후, 병합이 완료되면 병합이 완료된 블록에 대해 맵핑 정보를 갱신하여 메모리 영역(150)에 저장한다(S80).
- [0085] 이와 같이, 본 발명의 일 실시예에 따르면 종래와 같이 병합을 위해 항상 버퍼 블록에서 연산을 수행하는 것이 아니라, 소스 데이터 블록을 병합 타겟 블록처럼 이용할 수 있도록 제어함으로써 병합시 페이지 카피의 빈도수를 줄일 수 있다. 이로써, 각 블록의 고른 사용을 유도할 수 있으며 더 나아가 병합 수행 시간을 감소시킬 수 있으므로 반도체 스토리지 시스템의 성능을 향상시킬 수 있다.
- [0086] 본 발명이 속하는 기술분야의 당업자는 본 발명이 그 기술적 사상이나 필수적 특징을 변경하지 않고서 다른 구체적인 형태로 실시될 수 있으므로, 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적인 것이 아닌 것으로서 이해해야만 한다. 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 등가개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

도면의 간단한 설명

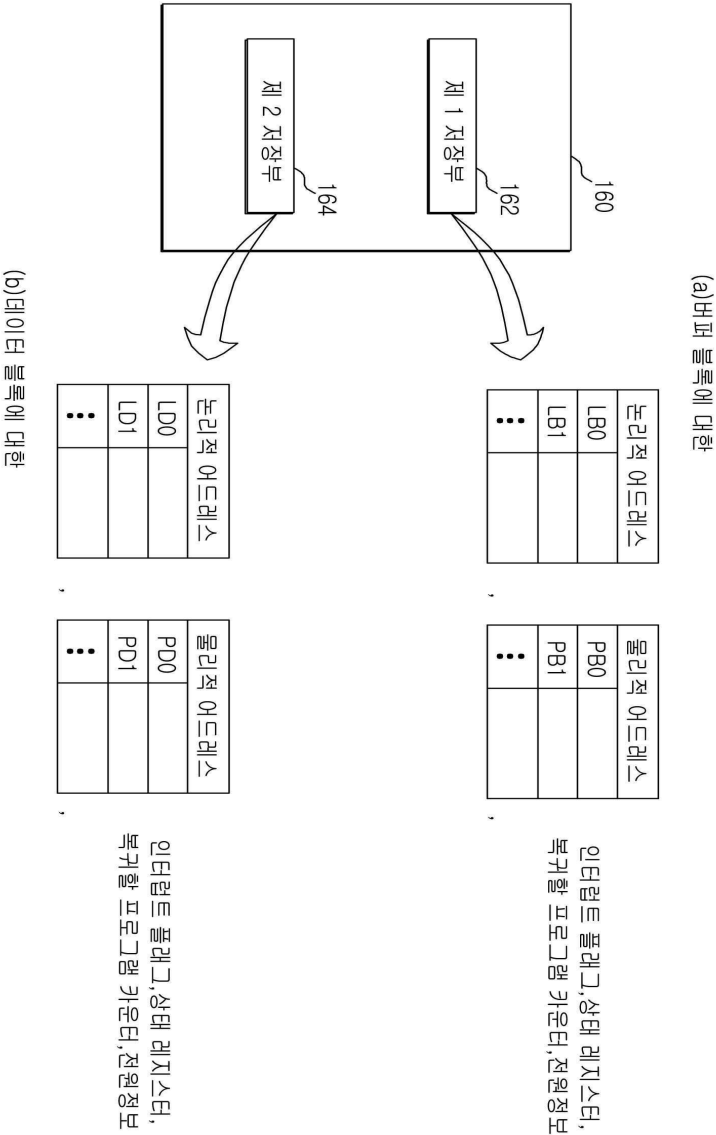
- [0087] 도 1은 본 발명의 일 실시예에 따른 반도체 스토리지 시스템의 블록도,
- [0088] 도 2는 도 1에 따른 정보 저장 영역의 개념적인 블록도,
- [0089] 도 3 내지 도 6은 도 1에 따른 병합의 다양한 실시예를 개념적으로 나타낸 블록도, 및
- [0090] 도 7은 본 발명의 일 실시예에 따른 반도체 스토리지 시스템의 제어 방법을 나타낸 플로우 차트이다.
- [0091] <도면의 주요 부분에 대한 부호의 설명>
- [0092] 110: 호스트 인터페이스 120: 버퍼부
- [0093] 130: MCU 140: 메모리 컨트롤러
- [0094] 150: 메모리 영역

도면

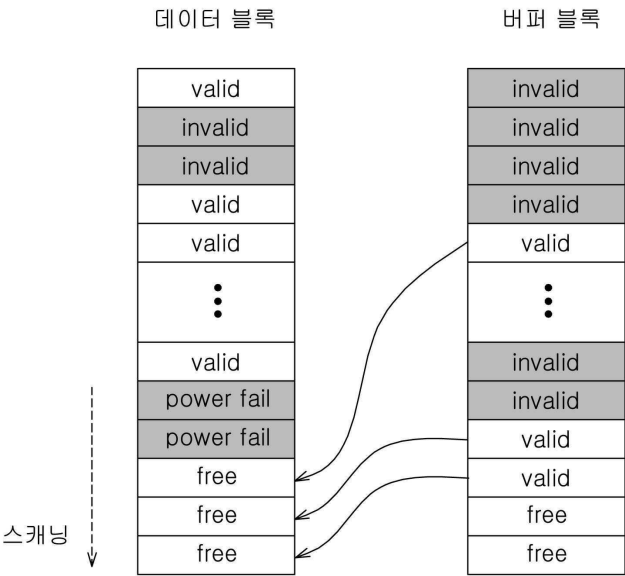
도면1



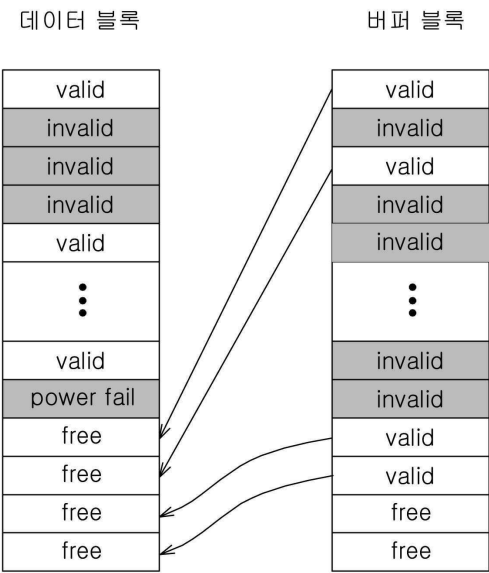
도면2



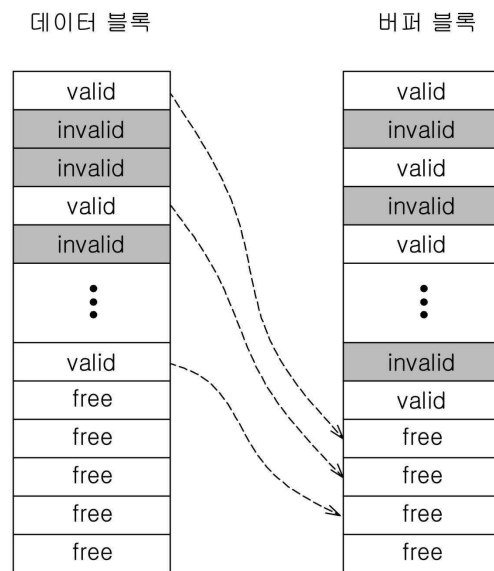
도면3



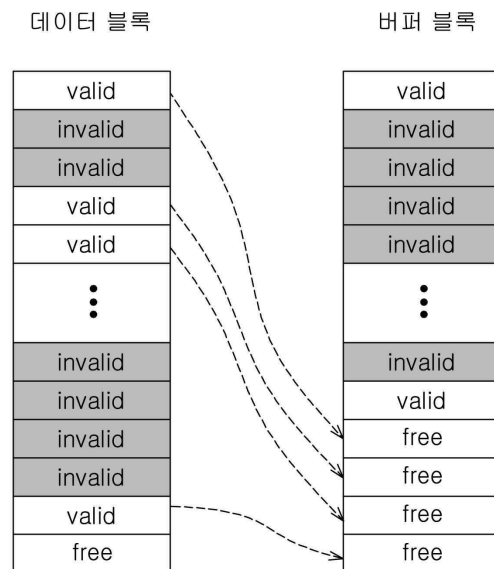
도면4



도면5



도면6



도면7

