



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2019년09월26일
(11) 등록번호 10-2025180
(24) 등록일자 2019년09월19일

(51) 국제특허분류(Int. Cl.)
G06F 13/16 (2006.01) G06F 12/02 (2018.01)
(21) 출원번호 10-2013-0094291
(22) 출원일자 2013년08월08일
심사청구일자 2018년08월06일
(65) 공개번호 10-2015-0018682
(43) 공개일자 2015년02월24일
(56) 선행기술조사문헌
US20040111557 A1
US20050216653 A1
US20080201391 A1
US20120317342 A1

(73) 특허권자
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
김상목
서울 강남구 봉은사로44길 75, 304호 (역삼동)
김경호
서울 강남구 영동대로118길 7, 201호 (삼성동)
(74) 대리인
특허법인 고려

전체 청구항 수 : 총 10 항

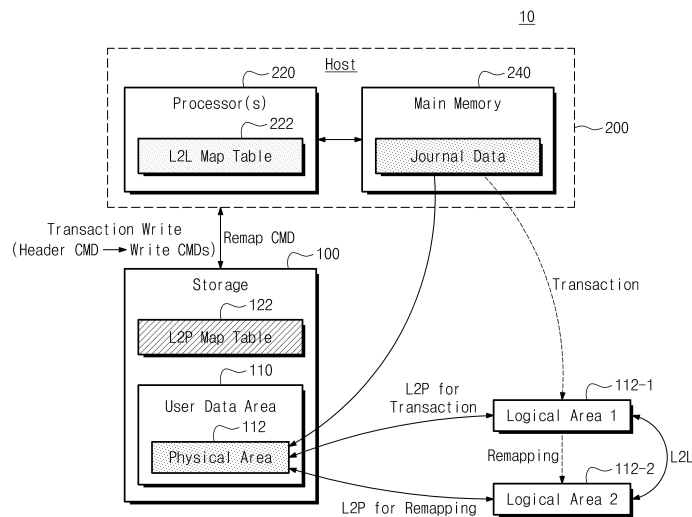
심사관 : 유주영

(54) 발명의 명칭 스토리지 시스템 및 그것의 쓰기 방법

(57) 요약

본 발명에 따른 스토리지 시스템은, 제 1 논리 영역을 제 2 논리 영역으로 매핑하는 제 1 맵 테이블을 포함하는 호스트; 및 상기 호스트에 연결되고, 적어도 하나의 비휘발성 메모리 장치 및 쓰기 동작시 상기 적어도 하나의 비휘발성 메모리 장치의 물리 영역을 상기 제 1 논리 영역으로 매핑하거나, 리맵 명령의 의거하여 상기 물리 영역을 상기 제 2 논리 영역으로 리매핑하는 제 2 맵 테이블을 포함하는 스토리지를 포함하고, 상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 호스트는 상기 리맵 명령을 상기 스토리지에 전송한다.

대표도



(72) 발명자

우영재

경기 수원시 영통구 동수원로 498, 505호 (매탄동,
유당빌딩)

신승욱

서울특별시 강북구 4.19로1길 27 402호 (수유동,
태영하이츠빌)

서성용

경기 수원시 장안구 영화로25번길 22, 2동 805호
(영화동, 태영아파트)

명세서

청구범위

청구항 1

호스트 및 상기 호스트에 연결된 스토리지를 포함하는 스토리지 시스템의 쓰기 방법에 있어서:

트랜잭션 발생시 저널 데이터를 입력 받는 단계;

상기 저널 데이터가 쓰여질 상기 스토리지의 제 1 논리 영역을 지시하는 제 1 논리 어드레스 및 상기 스토리지의 제 2 논리 영역을 지시하는 제 2 논리 어드레스 쌍으로 구성된 복수의 엔트리들을 제 1 맵 테이블에 삽입하는 단계;

상기 제 1 논리 어드레스에 대응하는 상기 스토리지의 물리 영역에 상기 저널 데이터를 쓰는 단계; 및

상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 복수의 엔트리들을 이용하여 상기 물리 영역에 대응하는 상기 제 1 논리 어드레스를 상기 제 2 논리 어드레스로 리매핑하는 단계를 포함하고,

상기 스토리지의 물리 영역에 상기 저널 데이터를 쓰는 단계는 사용자 페이지 데이터, 상기 제1,2 논리 어드레스, 및 지시 데이터를 저장하는 것을 포함하고, 상기 지시 데이터는 복수의 쓰기 명령들에 대응되는 트랜잭션이 복수의 페이지들 내에서 종료되었는지의 여부를 나타내는 데이터인, 쓰기 방법.

청구항 2

제 1 항에 있어서,

상기 제 1 맵 테이블은 상기 호스트에서 발생하는 쓰기 방법.

청구항 3

제 1 항에 있어서,

상기 제 1 맵 테이블은 상기 스토리지를 인식하는 단계에서 상기 스토리지로부터 상기 호스트로 전송되는 쓰기 방법.

청구항 4

제 1 항에 있어서,

상기 복수의 엔트리들 각각은 상기 제 1 논리 영역의 제 1 시작 어드레스, 상기 제 2 논리 영역의 제 2 시작 어드레스, 및 상기 제 1 및 제 2 시작 어드레스들의 크기를 지시하는 길이를 포함하는 쓰기 방법.

청구항 5

제 1 항에 있어서,

상기 저널 데이터를 입력 받은 후에, 트랜잭션 쓰기 동작을 지시하는 헤더 명령 및 상기 저널 데이터를 상기 물리영역에 쓰기 위한 복수의 쓰기 명령들을 발행하는 단계를 더 포함하는 쓰기 방법.

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

제 1 항에 있어서,

상기 리매핑하는 단계 이전에 파워 오프 될 때, 상기 복수의 페이지들 각각에 저장된 지시 데이터를 이용하여 파워 오프 리커버리 동작이 수행되는 쓰기 방법.

청구항 10

제 1 항에 있어서,

상기 물리 영역을 상기 제 1 논리 영역으로 매핑하거나 상기 물리 영역을 상기 제 2 논리 영역으로 리매핑하는 제 2 맵 테이블은 사전에 결정된 개수의 페이지들이 저장될 때 업데이트 되고,

상기 리매핑하는 단계 이전에 파워 오프 될 때, 상기 제 2 맵 테이블을 근거로 하여 마지막에 저장된 페이지부터 상기 트랜잭션 종료를 지시하는 지시 데이터를 갖는 페이지까지 읽음으로써 파워 오프 리커버리 동작이 수행되는 쓰기 방법.

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

제 1 항에 있어서,

파일 시스템에 전송된 데이터가 랜덤 데이터인 지를 판별하는 단계;

상기 판별된 랜덤 데이터를 모으는 단계;

상기 모아진 랜덤 데이터에 대응하는 어드레스들을 연속적인 제 1 논리 어드레스들로 변환하는 단계;

상기 변환된 제 1 논리 어드레스들에 따라 순차적으로 랜덤 데이터를 상기 스토리지의 물리 영역들에 쓰는 단계; 및

상기 제 1 논리 영역의 가용 공간의 크기가 상기 사전에 결정된 값 이하일 때, 상기 제 1 논리 어드레스들을 상기 어드레스들로 리매핑하는 단계를 더 포함하는 쓰기 방법.

청구항 15

삭제

청구항 16

적어도 하나의 프로세서, 상기 적어도 하나의 프로세서의 처리 동작시 필요한 데이터를 저장하는 메인 메모리, 및 제 1 논리 영역을 제 2 논리 영역으로 매핑하는 제 1 맵 테이블을 포함하는 호스트; 및

상기 호스트에 연결되고, 적어도 하나의 비휘발성 메모리 장치 및 복수의 쓰기 명령들에 의해 수행되는 쓰기 동작시 상기 적어도 하나의 비휘발성 메모리 장치의 물리 영역을 상기 제 1 논리 영역으로 매핑하거나, 리맵 명령의 의거하여 상기 물리 영역을 상기 제 2 논리 영역으로 리매핑하는 제 2 맵 테이블을 포함하는 스토리지를 포함하고,

상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 호스트는 상기 리맵 명령을 상기 스토리지에 전송하는 스토리지 시스템.

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

제 1 논리 어드레스와 제 2 논리 어드레스를 매핑하는 제 1 맵 테이블을 갖는 호스트와 상기 호스트에 연결된 스토리지를 포함하는 스토리지 시스템의 쓰기 방법에 있어서:

상기 스토리지의 물리 영역에 데이터를 쓰는 단계; 상기 스토리지의 물리 영역에 데이터를 쓰는 단계는 사용자 페이지 데이터, 상기 제1,2 논리 어드레스, 및 지시 데이터를 저장하는 것을 포함하고, 상기 지시 데이터는 복수의 쓰기 명령들에 대응되는 트랜잭션이 복수의 페이지들 내에서 종료되었는지의 여부를 나타내는 데이터이며,

상기 물리 영역에 대응하는 물리 어드레스와 상기 제 1 논리 어드레스와 매핑하도록 제 2 맵 테이블을 업데이트 하는 단계; 및

리맵 명령에 의거하여 상기 물리 어드레스와 상기 제 2 논리 어드레스의 매핑하도록 상기 제 2 맵 테이블을 업데이트는 하는 단계를 포함하고,

상기 호스트는 상기 제 1 맵 테이블을 이용하여 상기 리맵 명령을 발행하는 쓰기 방법.

청구항 25

삭제

청구항 26

삭제

발명의 설명

기술 분야

[0001] 본 발명의 스토리지 시스템 및 그것의 쓰기 방법에 관한 것이다.

배경 기술

[0002] 스토리지 시스템(storage system)은 호스트(host)와 저장 장치(storage device)로 구성되며, UFS(universal

flash storage), SATA(serial ATA), SCSI(small computer small interface), SAS(serial attached SCSI), eMMC(embedded MMC) 등과 같은 다양한 인터페이스 표준을 통해 연결된다.

[0003] 데이터 저장과 연관된 공통적인 문제는 저장 매체에서 파일 시스템의 일부분의 손상(corruption)이다. 데이터 손상은 흔히 저장 장치에서 데이터 기록 동작(data writing operations)의 중단이라는 결과를 가져오며, 이것은 예를 들어 전력 손실(power loss)의 결과로서 일어날 수 있다. 데이터 손상을 피하는 한가지 접근법은 저장 매체, 예를 들면, 하드 디스크 드라이브에 대한 변경사항(change)을 실행에 옮기기 전에 그 변경사항을 로그할 수 있는 저널링 시스템(journaling system)을 이용하는 것을 포함한다.

[0004] 그러므로, 전력 손실이 있을 때, 저널된 변경사항(journalled changes)이 "재생(replay)" 또는 실행되어 실제 데이터 구조를 저널된 변경사항과 같아지게 할 수 있다. 만일 저널된 액션들(journalled actions)이 전력 손실로 인해 불완전하고 이들의 설정이 저장 매체 상의 데이터 구조를 손상할 경우, 이러한 액션들은 쉽게 재생되지는 않는다. 따라서, 저널 시스템은 전력 손실 동안 발생할 수 있는 불완전 기록(incomplete write)을 보정함으로써 저장 매체 상에서 데이터 손상을 방지할 수 있다.

발명의 내용

해결하려는 과제

[0005] 본 발명의 목적은 저널링을 제공하는 시스템의 성능을 향상시키는 스토리지 시스템 및 그것의 쓰기 방법을 제공하는데 있다.

과제의 해결 수단

[0006] 본 발명의 실시 예에 따른 호스트 및 상기 호스트에 연결된 스토리지를 포함하는 스토리지 시스템의 쓰기 방법은: 트랜잭션 발생시 저널 데이터를 입력 받는 단계; 상기 저널 데이터가 쓰여질 상기 스토리지의 제 1 논리 영역을 지시하는 제 1 논리 어드레스 및 상기 스토리지의 제 2 논리 영역을 지시하는 제 2 논리 어드레스 쌍으로 구성된 복수의 엔트리들을 제 1 맵 테이블에 삽입하는 단계; 상기 제 1 논리 어드레스에 대응하는 상기 스토리지의 물리 영역에 상기 저널 데이터를 쓰는 단계; 및 상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 복수의 엔트리들을 이용하여 상기 물리 영역에 대응하는 상기 제 1 논리 어드레스를 상기 제 2 논리 어드레스로 리매핑하는 단계를 포함한다.

[0007] 실시 예에 있어서, 상기 제 1 맵 테이블은 상기 호스트에서 발생된다.

[0008] 실시 예에 있어서, 상기 제 1 맵 테이블은 상기 스토리지를 인식하는 단계에서 상기 스토리지로부터 상기 호스트로 전송된다.

[0009] 실시 예에 있어서, 상기 복수의 엔트리들 각각은 상기 제 1 논리 영역의 제 1 시작 어드레스, 상기 제 2 논리 영역의 제 2 시작 어드레스, 및 상기 제 1 및 제 2 시작 어드레스들의 크기를 지시하는 길이를 포함한다.

[0010] 실시 예에 있어서, 상기 저널 데이터를 입력 받은 후에, 트랜잭션 쓰기 동작을 지시하는 헤더 명령 및 상기 저널 데이터를 상기 물리영역에 쓰기 위한 복수의 쓰기 명령들을 발행하는 단계를 더 포함한다.

[0011] 실시 예에 있어서, 상기 헤더 명령은 상기 복수의 엔트리들을 포함한다.

[0012] 실시 예에 있어서, 상기 쓰기 명령들 각각은 트랜잭션 쓰기 동작의 마지막 데이터인 지 여부를 지시하는 콘텍스트 식별번호를 포함한다.

[0013] 실시 예에 있어서, 상기 물리 영역에 상기 저널 데이터를 쓰는 단계는, 상기 복수의 쓰기 명령들에 응답하여 사용자 페이지 데이터, 상기 제 1 논리 어드레스, 상기 제 2 논리 어드레스, 및 트랜잭션 종료 여부를 지시하는 지시 데이터를 복수의 페이지들에 저장하는 단계를 더 포함한다.

[0014] 실시 예에 있어서, 상기 리매핑하는 단계 이전에 파워 오프 될 때, 상기 복수의 페이지들 각각에 저장된 지시 데이터를 이용하여 파워 오프 리커버리 동작이 수행된다.

[0015] 실시 예에 있어서, 상기 물리 영역을 상기 제 1 논리 영역으로 매핑하거나 상기 물리 영역을 상기 제 2 논리 영역으로 리매핑하는 제 2 맵 테이블은 사전에 결정된 개수의 페이지들이 저장될 때 업데이트 되고, 상기 리매핑하는 단계 이전에 파워 오프 될 때, 상기 제 2 맵 테이블을 근거로 하여 마지막에 저장된 페이지부터 상기 트랜잭션 종료를 지시하는 지시 데이터를 갖는 페이지까지 읽음으로써 파워 오프 리커버리 동작이 수행된다.

- [0016] 실시 예에 있어서, 상기 제 1 논리 영역의 가용 공간이 없을 때, 상기 호스트는 리맵 명령을 발행하는 단계를 더 포함한다.
- [0017] 실시 예에 있어서, 상기 호스트는 상기 제 1 맵 테이블을 이용하여 상기 가용 공간의 크기가 상기 사전에 결정된 값 이하인 지를 판별한다.
- [0018] 실시 예에 있어서, 상기 스토리지는 상기 가용 공간의 크기가 상기 사전에 결정된 값 이하인 지를 판별하고, 상기 판별된 결과를 상기 호스트로 전송한다.
- [0019] 실시 예에 있어서, 파일 시스템에 전송된 데이터가 랜덤 데이터인 지를 판별하는 단계; 상기 판별된 랜덤 데이터를 모으는 단계; 상기 모아진 랜덤 데이터에 대응하는 어드레스들을 연속적인 제 1 논리 어드레스들로 변환하는 단계; 상기 변환된 제 1 논리 어드레스들에 따라 순차적으로 랜덤 데이터를 상기 스토리지의 물리 영역들에 쓰는 단계; 및 상기 제 1 논리 영역의 가용 공간의 크기가 상기 사전에 결정된 값 이하일 때, 상기 제 1 논리 어드레스들을 상기 어드레스들로 리매핑하는 단계를 더 포함한다.
- [0020] 실시 예에 있어서, 상기 스토리지의 물리 영역들에 쓰는 단계 이전에, 상기 랜덤 데이터를 랜덤 쓰기 버퍼에 버퍼링하는 단계를 더 포함한다.
- [0021] 본 발명의 실시 예에 따른 스토리지 시스템은, 적어도 하나의 프로세서, 상기 적어도 하나의 프로세서의 처리 동작시 필요한 데이터를 저장하는 메인 메모리, 및 제 1 논리 영역을 제 2 논리 영역으로 매핑하는 제 1 맵 테이블을 포함하는 호스트; 및 상기 호스트에 연결되고, 적어도 하나의 비휘발성 메모리 장치 및 쓰기 동작시 상기 적어도 하나의 비휘발성 메모리 장치의 물리 영역을 상기 제 1 논리 영역으로 매핑하거나, 리맵 명령의 의거하여 상기 물리 영역을 상기 제 2 논리 영역으로 리매핑하는 제 2 맵 테이블을 포함하는 스토리지를 포함하고, 상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 호스트는 상기 리맵 명령을 상기 스토리지에 전송한다.
- [0022] 실시 예에 있어서, 상기 쓰기 동작은 저널링 시스템에 의한 트랜잭션 쓰기 동작이다.
- [0023] 실시 예에 있어서, 상기 쓰기 동작은 파일 시스템에 의한 랜덤 쓰기 동작이다.
- [0024] 실시 예에 있어서, 상기 스토리지는 상기 랜덤 쓰기 동작시 랜덤 데이터를 버퍼링하는 랜덤 쓰기 버퍼를 더 포함하고, 상기 랜덤 쓰기 버퍼에 버퍼링된 랜덤 데이터가 상기 제 1 논리 영역에 대응하는 상기 물리 영역에 저장된다.
- [0025] 실시 예에 있어서, 상기 쓰기 동작은 복수의 쓰기 명령들에 의해 수행되고, 상기 복수의 쓰기 명령들에 응답하여 복수의 페이지들에게 사용자 페이지 데이터가 쓰여질 때, 상기 복수의 페이지들 각각에는 상기 제 1 논리 영역의 제 1 논리 어드레스, 상기 제 2 논리 영역의 제 2 논리 어드레스, 및 상기 쓰기 동작의 마지막 데이터인지 여부를 지시하는 지시 데이터가 함께 쓰여진다.
- [0026] 실시 예에 있어서, 파워 오프시 상기 지시 데이터를 이용하여 상기 제 2 맵 테이블이 복구된다.
- [0027] 실시 예에 있어서, 파일 시스템에서 상기 제 2 논리 영역에 대한 읽기 요청을 발행할 때, 상기 호스트의 디스크 드라이버는 상기 제 1 맵 테이블로부터 상기 제 2 맵 테이블의 상기 리매핑 이전인지 여부를 판별하고, 상기 제 2 맵 테이블이 상기 리매핑 이전일 때 상기 제 1 논리 어드레스에 따른 읽기 요청을 상기 스토리지에 전송한다.
- [0028] 실시 예에 있어서, 상기 호스트는 어플리케이션 프로세서이고, 상기 스토리지는 eMMC이다.
- [0029] 본 발명의 다른 실시 예에 따른 제 1 논리 어드레스와 제 2 논리 어드레스를 매핑하는 제 1 맵 테이블을 갖는 호스트와 상기 호스트에 연결된 스토리지를 포함하는 스토리지 시스템의 쓰기 방법은: 상기 스토리지의 물리 영역에 데이터를 쓰는 단계; 상기 물리 영역에 대응하는 물리 어드레스와 상기 제 1 논리 어드레스와 매핑하도록 제 2 맵 테이블을 업데이트하는 단계; 및 리맵 명령에 의거하여 상기 물리 어드레스와 상기 제 2 논리 어드레스의 매핑하도록 상기 제 2 맵 테이블을 업데이트는 하는 단계를 포함하고, 상기 호스트는 상기 제 1 맵 테이블을 이용하여 상기 리맵 명령을 발행한다.
- [0030] 본 발명의 또 다른 실시 예에 따른 호스트 및 상기 호스트에 연결된 스토리지를 포함하는 스토리지 시스템의 쓰기 방법은: 트랜잭션 발생시 저널 데이터를 제 1 논리 영역에 대응하는 물리 영역에 쓰는 단계; 및 상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 물리 영역에 쓰여진 저널 데이터의 논리 영역을 상기 제 1 논리 영역에서 제 2 논리 영역으로 리매핑하는 단계를 포함한다.

[0031] 본 발명의 실시 예에 따른 호스트에서 운용되는 프로그램의 응용 프로그래밍 인터페이스(application programming interface; API)를 이용한 저널링 방법은: 저널링 쓰기 요청시 상기 스토리지의 제 1 논리 영역을 지시하는 제 1 논리 어드레스 및 상기 스토리지의 제 2 논리 영역을 지시하는 제 2 논리 어드레스 쌍으로 구성된 복수의 엔트리들을 발생하는 단계; 저널 데이터 및 상기 복수의 엔트리들을 제 1 논리 영역에 대응하는 물리 영역에 쓰여지도록 상기 스토리지로 전송하는 단계; 및 상기 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 상기 물리 영역에 쓰여진 저널 데이터의 논리 영역을 상기 제 1 논리 영역에서 상기 제 2 논리 영역으로 변경하기 위한 리맵 명령을 발행하는 단계를 포함한다.

발명의 효과

[0032] 상술한 바와 같이 본 발명에 따른 스토리지 시스템 및 그것의 쓰기 방법은, 스토리지의 조건에 따라 리매핑을 수행하는 저널링을 수행함으로써, 중복 쓰기를 줄이면서 맵 테이블의 업데이트 횟수를 크게 줄일 수 있다.

도면의 간단한 설명

- [0033] 도 1은 본 발명의 개념을 설명하기 위한 스토리지 시스템을 보여주는 도면이다.
- 도 2는 도 1에 도시된 스토리지 시스템에 대한 소프트웨어 계층을 예시적으로 설명한 도면이다.
- 도 3은 도 1에 도시된 L2L 맵 테이블의 엔트리에 대한 실시 예를 예시적으로 보여주는 도면이다.
- 도 4는 본 발명의 실시 예에 따른 스토리지 시스템에서 저널 데이터 쓰기 동작을 간단하게 설명하기 위한 도면이다.
- 도 5는 본 발명의 실시 예에 따른 저널 데이터의 쓰기 동작에 대한 실시 예를 보여주는 도면이다.
- 도 6은 본 발명의 실시 예에 따른 스토리지 시스템의 읽기 동작을 간단하게 설명하기 위한 도면이다.
- 도 7은 본 발명의 실시 예에 따른 스토리지 시스템의 읽기 동작에 대한 실시 예를 보여주는 도면이다.
- 도 8은 본 발명의 실시 예에 따른 스토리지 시스템의 리매핑을 간단하게 설명하기 위한 도면이다.
- 도 9는 본 발명의 실시 예에 따른 스토리지 시스템의 리매핑에 대한 실시 예를 보여주는 도면이다.
- 도 10은 본 발명의 실시 예에 따른 스토리지 시스템의 랜덤 쓰기를 설명하기 위한 소프트웨어 계층을 예시적으로 보여주는 도면이다.
- 도 11은 본 발명의 실시 예에 따른 스토리지 시스템의 파워 오프 리커버리 동작을 설명하기 위한 도면이다.
- 도 12는 본 발명의 실시 예에 따른 스토리지 시스템의 저널 데이터 쓰기 방법을 예시적으로 보여주는 흐름도이다.
- 도 13은 본 발명의 실시 예에 따른 파워 오프 복구 방법에 대한 제 1 실시 예를 보여주는 흐름도이다.
- 도 14는 본 발명의 실시 예에 따른 파워 오프 복구 방법에 대한 제 2 실시 예를 보여주는 흐름도이다.
- 도 15 내지 도 18은 본 발명의 응용 예들을 예시적으로 보여주는 도면들이다.

발명을 실시하기 위한 구체적인 내용

[0034] 아래에서는 도면들을 이용하여 본 발명의 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있을 정도로 본 발명의 내용을 명확하고 상세하게 기재할 것이다.

[0035] 본 발명의 실시 예에 따른 비휘발성 메모리 장치로 구현된 스토리지 시스템 및 그것의 쓰기 방법은, 스토리지의 조건에 따라 발생하는 리맵 명령(remap command)에 의거하여 어느 하나의 물리 영역 대응하는 제 1 논리 영역을 제 2 논리 영역으로 변환시키는 리매핑(remapping) 수행함으로써, 스토리지의 성능을 향상시킬 수 있다. 여기서 스토리지 조건은 제 1 논리 영역/제 2 논리 영역의 가용 공간에 대한 정보일 수 있다.

[0036] 아래에서 비휘발성 메모리 장치는 낸드 플래시 메모리(NAND Flash Memory), 수직형 낸드 플래시 메모리(Vertical NAND; VNAND), 노아 플래시 메모리(NOR Flash Memory), 저항성 램(Resistive Random Access Memory: RRAM), 상변화 메모리(Phase-Change Memory: PRAM), 자기저항 메모리(Magnetoresistive Random Access Memory: MRAM), 강유전체 메모리(Ferroelectric Random Access Memory: FRAM), 스핀주입 자화반전 메모리(Spin

Transfer Torque Random Access Memory: STT-RAM) 등이 될 수 있다. 또한, 비휘발성 메모리 장치는 3차원 어레이 구조(three-dimensional array structure)로 구현될 수 있다. 본 발명은 전하 저장층이 전도성 부유 게이트로 구성된 플래시 메모리 장치는 물론, 전하 저장층이 절연막으로 구성된 차지 트랩형 플래시(charge trap flash; CTF)에도 모두 적용 가능하다. 아래에서는 설명의 편의를 위하여 비휘발성 메모리 장치가 낸드 플래시 메모리 장치라고 하겠다.

- [0037] 도 1은 본 발명의 개념을 설명하기 위한 스토리지 시스템을 보여주는 도면이다. 도 1을 참조하면, 스토리지 시스템(10)은 스토리지(100, 혹은 "디스크") 및 호스트(20)를 포함한다.
- [0038] 스토리지(100)는 사용자 데이터 영역(110) 및 L2P 맵 테이블(122)을 포함한다.
- [0039] 사용자 데이터 영역(110)은 비휘발성 메모리 장치에 의해 구현될 수 있다. 도 1에 도시된 바와 같이, 어느 하나의 물리 영역(112)은 제 1 논리 영역(112-1, 혹은 "journal data area") 및 제 2 논리 영역(112-2, 혹은 "disk data area") 중 어느 하나에 매핑 될 수 있다.
- [0040] L2P 맵 테이블(122, "제 2 맵 테이블")은 논리 영역의 어드레스(이하, "논리 어드레스")과 논리 어드레스에 대응하는 물리 영역의 어드레스(이하, "물리 어드레스")에 대한 정보를 저장할 수 있다. L2P 맵 테이블(122)은 소프트웨어 계층 측면에서 FTL(flash translation layer)에서 관리될 수 있다. 스토리지(100)의 FTL은 호스트(200)로부터 전송된 논리 어드레스를 L2P 맵 테이블(122)을 이용하여 물리 어드레스로 변환시킬 수 있다.
- [0041] 정리하면, 스토리지(100)는 물리 영역(110)을 제 1 논리 영역(112-1)으로 매핑하거나, 물리 영역(110)을 제 2 논리 영역(112-2)으로 리매핑하는 L2P 맵 테이블(122)을 포함할 수 있다.
- [0042] 호스트(200)는 스토리지(100)를 관리할 수 있다. 호스트(200)는 적어도 하나의 프로세서(220) 및 메인 메모리(240)를 포함할 수 있다.
- [0043] 적어도 하나의 프로세서(220)는 호스트(200)의 전반적인 동작을 제어할 수 있다. 프로세서(220)는 특정 계산들 혹은 테스크들을 수행하는 특정 소프트웨어를 실행하는 것과 같은 다양한 컴퓨팅 기능들을 수행할 수 있다. 프로세서(220)는 운영 체제(operating system ; OS) 및/혹은 어플리케이션과 같은 다양한 프로그램을 수행할 수 있다. OS 및/혹은 어플리케이션은 메인 메모리(240)에 저장되거나 혹은 도시되지 않은 호스트(200) 내부의 메모리 장치(들)에 저장될 수 있다.
- [0044] 호스트(200)는 L2L 맵 테이블(222, 혹은 "제 1 맵 테이블")을 포함할 수 있다. 호스트(200)는 L2L 맵 테이블(222)을 발생 및 관리할 수 있다. L2L 맵 테이블(222)은 스토리지(100)의 제 1 논리 영역(112-1)의 어드레스(이하, "제 1 논리 어드레스")에 대응하는 제 2 논리 영역(112-2)의 어드레스(이하, "제 2 논리 어드레스") 쌍에 관련된 엔트리들로 구성될 수 있다.
- [0045] 메인 메모리(240)는 프로세서(220)의 구동에 필요한 데이터를 저장할 수 있다. 실시 예에 있어서, 메인 메모리(240)는 휘발성 메모리 장치(예를 들어, DRAM, SRAM, ...) 혹은 비휘발성 메모리 장치(PRAM,...)로 구현될 수 있다.
- [0046] 한편, 아래에서는 파일의 업데이트 혹은 변경 등과 같은 트랜잭션(transaction) 발생시 저널 데이터(journal data) 처리 과정을 설명하도록 하겠다. 이때, 현상적으로 메인 메모리(240)의 저널 데이터는 스토리지(100)의 물리 영역(112)에 쓰여진다. 그런데, 호스트(200)는 물리 영역(112)의 쓰여진 저널 데이터를 두 가지 측면에서 해석할 수 있다.
- [0047] 첫째로, 저널 데이터를 제 1 논리 영역(112-1, "저널 데이터 영역")에 저장했다는 트랜잭션 쓰기 동작으로 해석할 수 있다. 트랜잭션 쓰기 동작은 트랜잭션 쓰기 동작을 지시하는 헤더 명령 및 저널 데이터를 물리 영역(112)에 쓰기 위한 복수의 쓰기 명령들로 구성될 수 있다.
- [0048] 헤더 명령은 복수의 쓰기 명령들에 대응하는 L2L 맵 테이블(222)의 일부, 즉 엔트리들을 포함할 수 있다. 즉, 트랜잭션 쓰기 동작시 호스트(200)는 헤더 명령을 이용하여 L2L 맵 테이블(222)의 일부를 스토리지(100)로 전송할 수 있다.
- [0049] 둘째로, 저널 데이터를 제 2 논리 영역(112-2)으로 옮겼다는 리매핑(remapping)으로 해석할 수 있다. 여기서 리매핑은 트랜잭션 쓰기에서 물리 영역(112)에 대응하는 제 1 논리 영역(112-1)의 제 1 논리 어드레스를 제 2 논리 영역(112-2)의 제 2 논리 어드레스로 변환시키는 것을 의미한다.
- [0050] 실시 예에 있어서, 리매핑은 호스트(200)의 리맵 명령(Remap CMD)에 의거하여 수행될 수 있다. 즉, 리매핑에 의

거하여 L2P 맵 테이블(122, 이하 "제 2 맵 테이블")은 변경될 수 있다.

- [0051] 실시 예에 있어서, 리맵 명령은 제 1 논리 영역(112-1)의 가용 공간의 크기가 사전에 결정된 값 이하일 때 호스트(200)에서 발행될 수 있다. 예를 들어, 제 1 논리 영역(112-1)의 가용 공간이 없을 때, 즉, 제 1 논리 영역(112-1)이 전부 사용되었을 때, 리맵 명령이 발행될 수 있다.
- [0052] 실시 예에 있어서, 제 1 논리 영역(112-1)의 크기 및 제 2 논리 영역(112-2)의 크기는 호스트(200)에 의해 설정/가변할 수 있다.
- [0053] 다른 실시 예에 있어서, 제 1 논리 영역(112-1)의 크기 및 제 2 논리 영역(112-2)의 크기는 스토리지(100)에 의해 설정/가변될 수 있다.
- [0054] 실시 예에 있어서, 호스트(200)는 제 1 논리 영역(112-1)의 가용 공간에 대한 정보를 이용하여 리맵 명령을 발행할 수 있다. 실시 예에 있어서, 호스트(200)는 스토리지(100)의 인식 단계에서 제 1 논리 영역(112-1)의 가용 공간에 대한 정보를 스토리지(100)로부터 전송받을 수 있다. 다른 실시 예에 있어서, 호스트(200)는 L2L 맵 테이블(222, "제 1 맵 테이블")을 이용하여 제 1 논리 영역(112-1)의 가용 공간에 대한 정보를 얻을 수 있다.
- [0055] 실시 예에 있어서, 호스트(200)는 L2L 맵 테이블(222)로부터 리매핑이 완료된 엔트리들을 제거/삭제할 수 있다.
- [0056] 일반적인 스토리지 시스템은 트랜잭션 발생시 저널 데이터를 저널 데이터 영역에 우선적으로 썼다가, 체크 포인트(check point) 발생시 혹은 메인 메모리의 더티 페이지(dirty page)가 플러시(flush) 될 때 오리지널 데이터 영역으로 저널 데이터를 다시 쓴다. 즉, 저널 데이터의 중복 쓰기가 필연적으로 유발된다.
- [0057] 반면에, 본 발명의 실시 예에 따른 스토리지 시스템(10)은 트랜잭션 발생시 물리 영역(112)에 한번만 쓰고 리매핑을 이용하여 저널링을 수행할 수 있다. 이로써, 본 발명의 스토리지 시스템(10)은 종래의 그것과 비교하여 저널 데이터의 중복 쓰기를 피할 수 있다.
- [0058] 게다가, 본 발명의 스토리지 시스템(10)은 스토리지(100)의 조건, 즉, 예를 들어 제 1 논리 영역(112-1)의 가용 공간의 정보를 근거로 하여 리매핑함으로써, L2P 맵 테이블(122, "제 2 맵 테이블")의 업데이트 회수를 줄일 수 있다. L2P 맵 테이블(122)의 업데이트 회수가 줄어들면서, 본 발명은 시스템적인 성능 향상을 꾀할 수 있다.
- [0059] 도 2는 도 1에 도시된 스토리지 시스템(10)에 대한 소프트웨어 계층을 예시적으로 설명한 도면이다. 도 2를 참조하면, 스토리지(100)을 관리하는 호스트(200)는 소프트웨어적으로 파일 시스템, 저널링 시스템, 디바이스 드라이버로 구성될 수 있다. 저널링 시스템은 스토리지(100)를 관리하는 파일 시스템의 일부가 손상을 방지하기 위해 사용된다. 디바이스 드라이버는 파일 시스템/저널 시스템의 요청에 따라 스토리지(100)를 구동한다.
- [0060] 본 발명의 디바이스 드라이버는 L2L 맵 테이블(222, "제 1 맵 테이블" 도 1 참조)을 포함할 수 있다. 저널링 시스템으로부터 저널 데이터의 쓰기/읽기 요청시 디바이스 드라이버는 L2L 맵 테이블(222)을 이용하여 쓰기/읽기 요청에 따른 어드레스가 있는지 혹은 어드레스를 변경할 지를 판별하고, 판별 결과에 따라 스토리지(100)에 쓰기/읽기 요청을 전송한다. 스토리지(100)의 소프트웨어 계층(예를 들어, FTL)은 디바이스 드라이버로부터 전송된 쓰기/읽기 요청에 따라 스토리지(100)에 대한 쓰기/읽기 동작을 수행한다.
- [0061] 도 3는 도 1에 도시된 L2L 맵 테이블(222)의 엔트리에 대한 실시 예를 예시적으로 보여주는 도면이다. 도 3을 참조하면, L2L 맵 테이블(222)의 엔트리는 제 1 논리 어드레스, 제 2 논리 어드레스, 및 길이로 구성될 수 있다.
- [0062] 실시 예에 있어서, 제 1 논리 어드레스는 저널 어드레스이고, 제 2 논리 어드레스는 디스크 어드레스일 수 있다.
- [0063] 실시 예에 있어서, 길이는 트랜잭션 쓰기 동작에서 복수의 쓰기 명령들의 개수에 대응하는 값일 수 있다.
- [0064] 실시 예에 있어서, 디바이스 드라이버는 저널링 시스템의 트랜잭션 쓰기 요청시 L2L 맵 테이블(222)의 엔트리를 발생할 수 있다.
- [0065] 도 4는 본 발명의 실시 예에 따른 스토리지 시스템(10)에서 저널 데이터 쓰기 동작을 간단하게 설명하기 위한 도면이다. 도 4를 참조하면, 트랜잭션 발생시 저널링 시스템으로부터 쓰기 요청이 입력될 때, 디바이스 드라이버는 L2L 맵 테이블(222)에 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003, "저널 어드레스들")에 관련된 엔트리들이 존재하는 지를 판별한다. 도 4에서는 설명의 편의를 위하여 쓰기 요청된 어드레스들의 개수는 4라고 하였다. 하지만, 본 발명은 여기에 제한되지 않을 것이다.

- [0066] 만일, 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 관련된 엔트리들이 적어도 하나가 존재하지 않는다면, 디바이스 드라이버는 L2L 맵 테이블(222)에 관련된 적어도 하나의 엔트리를 발생 및 삽입할 수 있다.
- [0067] 반면에, 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 관련된 엔트리들이 적어도 하나 존재한다면, 디바이스 드라이버는 L2L 맵 테이블(222)에 관련된 적어도 하나의 엔트리를 업데이트할 수 있다.
- [0068] 이후, 디바이스 드라이버는 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 관련된 엔트리들을 포함하는 헤더 명령을 발생하고, 발생된 헤더 명령을 스토리지(100)로 전송한다.
- [0069] 이후에, 디바이스 드라이버는 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 대응하는 저널 데이터를 스토리지(100)의 물리 영역(112, 도 1 참조)에 쓰기 위한 복수의 쓰기 명령들을 발행하고, 발행된 쓰기 명령들 및 대응하는 저널 데이터들을 스토리지(100)로 전송한다. 이때 쓰기 명령들 각각에는 저널 데이터가 트랜잭션 쓰기의 마지막 데이터인지 여부를 지시하는 컨텍스트 식별번호(context identification)를 포함할 수 있다. 예를 들어, 컨텍스트 식별번호가 "0"이면 저널 데이터가 트랜잭션 쓰기의 마지막 데이터가 아니고, 컨텍스트 식별번호가 "1"이면 저널 데이터가 트랜잭션 쓰기의 마지막 데이터이다.
- [0070] 스토리지(100)는 헤더 명령, 쓰기 명령들, 저널 데이터를 입력 받고, 저널 데이터를 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 대응하는 물리 영역들에 쓸 것이다. 이때 물리 영역들에 저널 데이터를 저장할 때, 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 관련된 엔트리들도 함께 저장될 수 있다. 예를 들어, 저널 어드레스(1000)에 대응하는 물리 영역에 저널 데이터를 저장할 때, 저널 어드레스(1000)에 관련된 제 1 논리 어드레스(저널 어드레스, 1000) 및 제 2 논리 어드레스(디스크 어드레스, 8)가 함께 물리 영역에 저장될 수 있다.
- [0071] 실시 예에 있어서, 저널 데이터를 쓰기 요청된 어드레스들(1000, 1001, 1002, 1003)에 대응하는 물리 영역들에 저장할 때, 쓰여질 데이터가 저널 데이터의 마지막 데이터인 지 여부를 지시하는 지시 데이터도 함께 저장될 수 있다. 여기서 지시 데이터는 쓰기 명령들에 포함된 컨텍스트 식별번호에 대응하는 값들일 수 있다.
- [0072] 도 5는 본 발명의 실시 예에 따른 저널 데이터의 쓰기 동작에 대한 실시 예를 보여주는 도면이다. 도 5를 참조하면, 트랜잭션 발생시 메인 메모리(240)의 메모리 유닛들(M1, M2, M3) 각각에 저장된 저널 데이터는 트랜잭션 쓰기 동작으로 제 1 논리 영역의 어드레스들(101, 102, 103)에 대응하는 사용자 데이터 영역(110)의 물리 영역들(26, 27, 28)로 쓰여질 것이다.
- [0073] 도 6은 본 발명의 실시 예에 따른 스토리지 시스템의 읽기 동작을 간단하게 설명하기 위한 도면이다. 도 6을 참조하면, 설명의 편의를 위하여 파일 시스템은 어드레스들(10, 18)에 대한 읽기 요청을 발행한다고 가정하겠다. 디바이스 드라이버는 이러한 읽기 요청을 입력 받고, L2L 맵 테이블(222)에 읽기 요청된 어드레스들(10, 18)에 관련된 엔트리들이 존재하는 지를 검색한다. 만일, 읽기 요청된 어드레스들(10, 18)에 관련된 엔트리들이 존재한다면, 디바이스 드라이버는 읽기 요청된 어드레스들(10, 18)을 대응하는 저널 어드레스들(1001, 1002)로 변경하고, 변경된 저널 어드레스들(1001, 1002)을 이용하여 읽기 요청을 스토리지(100)에 전송한다. 스토리지(100)는 저널 어드레스들(1001, 1002)에 대응하는 데이터를 읽고, 읽혀진 데이터를 호스트(200)로 전송할 수 있다.
- [0074] 만일, 읽기 요청된 어드레스들(10, 18)에 관련된 엔트리들이 존재하지 않는다면, 디바이스 드라이버는 읽기 요청된 어드레스들(10, 18)에 대응하는 읽기 요청을 스토리지(100)에 전송한다. 스토리지(100)는 오리지널 어드레스들(10, 18)에 대응하는 데이터를 읽고, 읽혀진 데이터를 호스트(200)로 전송할 수 있다.
- [0075] 도 7은 본 발명의 실시 예에 따른 스토리지 시스템(10)의 읽기 동작에 대한 실시 예를 보여주는 도면이다. 도 7을 참조하면, 호스트(200)로부터 전송된 읽기 요청에 따라 스토리지(100)는 제 2 논리 영역의 어드레스들(10, 12, 15)에 대응하는 사용자 데이터 영역(110)의 어드레스들(21, 23, 25)의 물리 영역들로부터 데이터를 읽고, 읽혀진 데이터를 메인 메모리(240)의 메모리 유닛들(M1, M2, M3)로 전송할 수 있다.
- [0076] 도 8은 본 발명의 실시 예에 따른 스토리지 시스템(10)의 리매핑을 간단하게 설명하기 위한 도면이다. 도 8을 참조하면, 호스트(200)은 리맵 명령을 발행할 수 있다. 예를 들어, 디바이스 드라이버는 L2L 맵 테이블(222)를 이용하여 제 1 논리 영역(122-1)의 가용 공간이 사전에 결정된 값 이하인 지를 판별하고, 만일 제 1 논리 영역(122-1)의 가용 공간이 사전에 결정된 값 이하일 때 리맵 명령을 발행하고, 발행된 리맵 명령을 스토리지(100)에 전송할 수 있다. 여기서 리맵 명령은 리매핑과 관련된 L2L 맵 테이블(222, "제 1 맵 테이블")의 엔트리들을 포함할 수 있다. 한편, 리맵 명령이 전송된 엔트리들은 L2L 맵 테이블(222)에서 삭제될 수 있다.
- [0077] 스토리지(100)는 리맵 명령에 응답하여 제 1 논리 어드레스를 제 2 논리 어드레스로 변경하도록 L2P 맵 테이블(122, "제 2 맵 테이블")을 업데이트 한다.

- [0078] 도 9는 본 발명의 실시 예에 따른 스토리지 시스템(10)의 리매핑에 대한 실시 예를 보여주는 도면이다. 도 9를 참조하면, 스토리지(100)는 리맵 명령에 응답하여 제 1 논리 영역의 어드레스들(101, 102, 103)를 각각 제 2 논리 영역의 어드레스들(10, 12, 14)로 변경하도록 L2P 맵 테이블(122)를 업데이트 한다. 따라서, 리매핑 이후에 L2P 맵 테이블(122)에 따르면, 논리 어드레스(10)는 물리 어드레스(26)에 대응되고, 논리 어드레스(12)는 물리 어드레스(27)에 대응되고, 논리 어드레스(14)는 물리 어드레스(28)에 대응될 것이다. 한편, 논리 어드레스들(10, 11, 14) 각각은 메모리 유닛들(M1, M2, M3)에 대응될 수 있다.
- [0079] 본 발명의 스토리지 시스템(10)은 메타 데이터의 저널링뿐만 아니라 데이터 저널링을 지원할 수도 있다. 즉, 본 발명은 랜덤 쓰기(random write)에 적용할 수 있다. 여기서 랜덤 쓰기는 쓰기 요청에 따른 어드레스의 연속성을 갖지 않는 쓰기를 의미한다. 본 발명은 랜덤 데이터를 모으고, 랜덤 데이터에 대응하는 어드레스들을 연속성을 갖는 제 1 논리 어드레스들(예를 들어, "저널 어드레스들")로 변환하고, 변환된 제 1 논리 어드레스들에 대응하는 순서에 따라 랜덤 데이터를 쓰고, 이후에 리매핑을 수행할 수 있다.
- [0080] 도 10은 본 발명의 실시 예에 따른 스토리지 시스템(10)의 랜덤 쓰기를 설명하기 위한 소프트웨어 계층을 예시적으로 보여주는 도면이다. 도 10을 참조하면, 디바이스 드라이버는 파일 시스템/저널링 시스템에서 요청된 쓰기 요청이 랜덤 쓰기 요청인 지를 판별하고, 랜덤 쓰기 요청에 따른 랜덤 데이터를 모으는 데이터 로거 모듈(data logger module)을 포함한다. 디바이스 드라이버는 모아진 랜덤 데이터의 어드레스들을 연속성을 갖는 제 1 논리 어드레스들(저널 어드레스들)로 변환시킨다. 스토리지(100)는 변환된 제 1 논리 어드레스들의 순서에 따라 랜덤 데이터를 저장하고, 이후 리맵 명령에 응답하여 리매핑을 수행할 수 있다.
- [0081] 실시 예에 있어서, 스토리지(100)는 제 1 논리 어드레스들에 따라 순차적으로 랜덤 데이터를 버퍼링하는 랜덤 쓰기 버퍼(RWB)를 포함할 수 있다.
- [0082] 실시 예에 있어서, 랜덤 쓰기 버퍼(RWB)에 저장된 랜덤 데이터는 제 1 논리 어드레스들의 순서에 따라 사용자 데이터 영역에서 제 1 논리 영역에 대응하는 물리 영역으로 저장될 수 있다.
- [0083] 실시 예에 있어서, 스토리지(100)는 도 10에 도시된 바와 같이 저널 영역을 포함하는 사용자 파티션과 랜덤 쓰기 버퍼(RWB)를 포함하는 RWB 파티션으로 구성될 수 있다. 여기서 저널 영역은 저널링에 의한 저널 데이터가 쓰여질 영역이다.
- [0084] 도 11은 본 발명의 실시 예에 따른 스토리지 시스템(10)의 파워 오프 리커버리 동작을 설명하기 위한 도면이다. 도 11을 참조하면, 설명의 편의를 위하여 제 1 트랜잭션 쓰기 동작은 성공적으로 완료되었고, 제 1 트랜잭션 쓰기 동작은 도중에 파워-오프 되었다고 가정하겠다. 트랜잭션 쓰기 동작은 각 페이지에 제 1 논리 어드레스(LAB1), 제 2 논리 어드레스(LAB2), 및 트랜잭션 완료 여부를 지시하는 지시 데이터(TX/TC)를 함께 저장한다.
- [0085] 파워 오프 리커버리 동작시, 우선적으로 트랜잭션 로그 블록의 마지막 페이지부터 "TC"를 갖는 페이지를 검출하기까지 제 1 읽기 동작(POR Read 1)이 수행될 것이다. "TC"를 갖는 페이지가 검출되면, 트랜잭션 로그 블록의 첫 페이지부터 "TC"를 갖는 페이지까지 제 2 읽기 동작(POR Read 2)이 수행된다. 여기서 제 2 읽기 동작(POR Read 2)에 따라 읽혀진 페이지 데이터를 근거로 하여 L2P 맵 테이블(도 1참조, 122, "제 2 맵 테이블" 혹은 "논리-물리 어드레스 변환 테이블")이 복구될 수 있다.
- [0086] 도 12는 본 발명의 실시 예에 따른 스토리지 시스템의 저널 데이터 쓰기 방법을 예시적으로 보여주는 흐름도이다. 도 12를 참조하면, 저널링 시스템으로부터 저널 데이터가 디바이스 드라이버로 입력 된다(S110). 저널 데이터에 대응하는 페이지 캐쉬는 메인 메모리에 업데이트 된다. 디바이스 드라이버는 저널 데이터에 대응하는 제 1 논리 어드레스 및 제 2 어드레스 쌍을 발생하고, 발생된 어드레스 쌍에 관련된 엔트리를 L2L 맵 테이블(도 1 참조, 222 "제 1 맵 테이블" "논리-논리 어드레스 변환 테이블")에 삽입한다(S120). 헤더 명령 및 복수의 쓰기 명령들이 발행되고, 발행된 헤더 명령 및 쓰기 명령들에 따라 스토리지의 제 1 논리 어드레스에 대응하는 물리 영역에 저널 데이터가 쓰여지고, L2P 맵 테이블(도 1 참조, 122, "제 2 맵 테이블" "논리-물리 어드레스 변환 테이블")이 업데이트 된다(S130). 쓰기 명령에 따른 저널 데이터 쓰기 동작시, 저널 데이터의 마지막 데이터인 지 여부를 지시하는 트랜잭션 완료 여부 데이터(TX/TC)가 함께 저장될 수 있다.
- [0087] 이후, L2L 맵 테이블(222)에 의해 제 1 논리 영역의 가용 공간의 크기가 사전에 결정된 값 이하일 때, 리맵 명령이 발행될 수 있다. 스토리지(100)는 리맵 명령에 응답하여 리매핑을 수행한다. 여기서 리매핑은 물리 어드레스에 대응하는 제 1 논리 어드레스(저널 어드레스)를 제 2 논리 어드레스(오리지널 어드레스)로 변환시킨 L2P 맵 테이블(122)을 업데이트할 것이다(S140).

- [0088] 도 13은 본 발명의 실시 예에 따른 파워 오프 복구 방법에 대한 제 1 실시 예를 보여주는 흐름도이다. 도 11 및 도 13을 참조하면, 저널 데이터 복구 방법은 다음과 같다. 트랜잭션 로그 블록에서 "TC"을 갖는 페이지가 검색된다(S210). 트랜잭션 로그 블록에서 첫 페이지부터 검색된 페이지까지 페이지 데이터가 읽혀진다(S220). 이후, 읽혀진 페이지 데이터를 이용하여 L2P 맵 테이블(122)이 업데이트 될 것이다(S230).
- [0089] 본 발명에서 트랜잭션 로그 블록은 주기적으로 페이지 맵 테이블이 업데이트될 수 있다. 예를 들어, 10개의 페이지가 쓰여지면 맵 테이블이 업데이트 될 수 있다. 이 경우, 파워 오프 복구 방법은 다음과 같다.
- [0090] 도 14는 본 발명의 실시 예에 따른 파워 오프 복구 방법에 대한 제 2 실시 예를 보여주는 흐름도이다. 도 11 및 도 14를 참조하면, 저널 데이터 복구 방법은 다음과 같다. 마지막으로 저장된 트랜잭션 로그 블록의 맵 테이블이 읽혀진다(S310). 마지막 저장된 트랜잭션 로그 블록에서 "TC"을 갖는 페이지가 검색된다(S320). 읽혀진 맵 테이블을 저장했던 페이지부터 검색된 페이지까지 페이지 데이터가 읽혀진다(S330). 읽혀진 페이지 데이터를 이용하여 L2P 맵 테이블(122)이 업데이트된다(S340).
- [0091] 본 발명은 SSD(solid state drive)에 적용 가능하다.
- [0092] 도 15는 본 발명의 실시 예에 따른 SSD를 예시적으로 보여주는 블록도이다. 도 15를 참조하면, SSD(1000)는 복수의 비휘발성 메모리 장치들(1100) 및 SSD 제어기(1200)를 포함한다. SSD(1000)은 도 1에 도시된 스토리지(100)를 포함하도록 구현될 수 있다.
- [0093] 비휘발성 메모리 장치들(1100)은 옵션적으로 외부 고전압(Vpp)을 제공받도록 구현될 수 있다. SSD 제어기(1200)는 복수의 채널들(CH1 ~ CHi, i는 2이상의 정수)을 통하여 비휘발성 메모리 장치들(1100)에 연결된다. SSD 제어기(1200)는 적어도 하나의 프로세서(1210), 버퍼 메모리(1220), 에러 정정 회로(1230), 호스트 인터페이스(1250) 및 비휘발성 메모리 인터페이스(1260)를 포함한다.
- [0094] 버퍼 메모리(1220)는 메모리 제어기(1200)의 구동에 필요한 데이터를 임시로 저장할 것이다. 실시 예에 있어서, 버퍼 메모리(1220)는 동작 조건에 따른 에러율 테이블(ERT)을 저장할 수 있다. 버퍼 메모리(1220)는 데이터 혹은 명령을 저장하는 복수의 메모리 라인들을 포함할 수 있다. 여기서 복수의 메모리 라인들은 캐시 라인들(1212)에 다양한 방법으로 매핑 될 수 있다. 또한, 버퍼 메모리(1220)는 쓰기 요청시 파인 프로그램 동작에 이용될 데이터를 버퍼링해 놓을 수 있다. 도 15에서 버퍼 메모리(1220)는 SSD 제어기(1200) 내부에 존재하지만, 반드시 여기에 제한되지 않을 것이다. 버퍼 메모리는 SSD 제어기(1200)의 외부에 별도로 존재할 수도 있다. 에러 정정 회로(1230)는 쓰기 동작에서 프로그램될 데이터의 에러 정정 코드 값을 계산하고, 읽기 동작에서 읽혀진 데이터를 에러 정정 코드 값에 근거로 하여 에러 정정하고, 데이터 복구 동작에서 비휘발성 메모리 장치(1100)로부터 복구된 데이터의 에러를 정정할 수 있다. 도시되지 않았지만, 메모리 제어기(1200)를 구동하는 데 필요한 코드 데이터를 저장하는 코드 메모리가 더 포함될 수 있다. 코드 메모리는 비휘발성 메모리 장치로 구현될 수 있다. 호스트 인터페이스(1250)는 외부의 장치와 인터페이스 기능을 제공할 수 있다. 비휘발성 메모리 인터페이스(1260)는 비휘발성 메모리 장치(1100)와 인터페이스 기능을 제공할 수 있다.
- [0095] 본 발명의 실시 예에 따른 SSD(1000)는 외부의 리맵 명령에 근거로 하여 논리-물리 어드레스 변환 테이블을 리매핑함으로써, 시스템적 성능 향상을 꾀할 수 있다.
- [0096] 본 발명은 eMMC(embedded multimedia card, moviNAND, iNAND)에도 적용 가능하다.
- [0097] 도 16은 본 발명의 실시 예에 따른 eMMC를 예시적으로 보여주는 블록도이다. 도 16을 참조하면, eMMC(2000)는 적어도 하나의 낸드 플래시 메모리 장치(2100) 및 제어기(2200)를 포함할 수 있다. 낸드 플래시 메모리 장치(2100)는 원-스텝/멀티-스텝 프로그램 동작들을 수행하도록 구현될 수 있다. eMMC(2000)는 도 1에 도시된 스토리지(100)로 구현될 수 있다.
- [0098] 메모리 제어기(2200)는 복수의 채널들을 통하여 낸드 플래시 메모리 장치(2100)에 연결된다. 메모리 제어기(2200)는 적어도 하나의 제어기 코어(2210), 호스트 인터페이스(2250) 및 낸드 인터페이스(2260)를 포함한다. 적어도 하나의 제어기 코어(2210)는 eMMC(2000)의 전반적인 동작을 제어한다. 호스트 인터페이스(2250)는 제어기(2210)와 호스트의 인터페이싱을 수행한다. 낸드 인터페이스(2260)는 낸드 플래시 메모리 장치(2100)와 제어기(2200)의 인터페이싱을 수행한다. 실시 예에 있어서, 호스트 인터페이스(2250)는 병렬 인터페이스(예를 들어, MMC 인터페이스)일 수 있다. 다른 실시 예에 있어서, eMMC(2000)의 호스트 인터페이스(2250)는 직렬 인터페이스(예를 들어, UHS-II, UFS 인터페이스)일 수 있다. 또한, eMMC(2000)는 호스트로부터 전원 전압들(Vcc, Vccq)을 제공받는다. 여기서, 제 1 전원 전압(Vcc, 예를 들어 3.3V)은 낸드 플래시 메모리 장치(2100) 및 낸드 인터페이스

스(2230)에 제공되고, 제 2 전원 전압(Vccq, 예를 들어 1.8V/3.3V)은 제어기(2200)에 제공된다. 실시 예에 있어서, eMMC(1000)는 외부 고전압(Vpp)을 옵션적으로 제공받을 수 있다.

[0099] 본 발명의 실시 예에 따른 eMMC(2000)는 저널링 혹은 랜덤 쓰기 동작에서 리매핑 기법을 이용함으로써, 중복 쓰기를 피하면서도 어드레스 변환 테이블의 업데이트 회수를 크게 줄일 수 있다.

[0100] 발명은 UFS(universal flash storage)에도 적용 가능하다.

[0101] 도 17은 본 발명의 실시 예에 따른 UFS 시스템을 예시적으로 보여주는 블록도이다. 도 17을 참조하면, UFS 시스템(3000)은 UFS 호스트(3100), UFS 장치들(3200, 3300), 임베디드 UFS 장치(3300), 착탈형 UFS 카드(3400)를 포함할 수 있다. UFS 호스트(3100)는 모바일 장치의 어플리케이션 프로세서일 수 있다. UFS 호스트(3100)의 도 1에 도시된 호스트(200)로 구현될 수 있다. UFS 장치들(3200, 3300), 임베디드 UFS 장치(3300), 및 착탈형 UFS 카드(3400) 각각은 UFS 프로토콜에 의하여 외부의 장치들과 통신할 수 있다. UFS 장치들(3200, 3300), 임베디드 UFS 장치(3300), 및 착탈형 UFS 카드(3400) 중 적어도 하나는 도 1에 도시된 스토리지(100)로 구현될 수 있다. 한편, 임베디드 UFS 장치(3300)와 착탈형 UFS 카드(3400)는 UFS 프로토콜이 아닌 다른 프로토콜에 의해 통신할 수 있다. UFS 호스트(3100)와 착탈형 UFS 카드(3400)는 다양한 카드 프로토콜(예를 들어, UFDs, MMC, SD(secure digital), mini SD, Micro SD 등)에 의해 통신할 수 있다.

[0102] 본 발명은 모바일 장치에도 적용 가능하다.

[0103] 도 18은 본 발명의 실시 예에 따른 모바일 장치(4000)를 예시적으로 보여주는 블록도이다. 도 18을 참조하면, 모바일 장치(4000)는 어플리케이션 프로세서(4100), 통신 모듈(4200), 디스플레이/터치 모듈(4300), 저장 장치(4400), 및 모바일 램(4500)를 포함한다.

[0104] 어플리케이션 프로세서(4100)는 모바일 장치(4000)의 전반적인 동작을 제어한다. 특히, 어플리케이션 프로세서(4100)는 도 1에 도시된 호스트(200)로 구현될 수 있다. 통신 모듈(4200)은 외부와의 유선/무선 통신을 제어하도록 구현될 것이다. 디스플레이/터치 모듈(4300)은 어플리케이션 프로세서(4100)에서 처리된 데이터를 디스플레이 하거나, 터치 패널로부터 데이터를 입력 받도록 구현될 것이다. 저장 장치(4400)는 사용자의 데이터를 저장하도록 구현될 것이다. 저장 장치(4400)는 eMMC, SSD, UFS 장치일 수 있다. 저장 장치(4400)는 도 1 내지 도 15에서 설명된 바와 같이 트랜잭션 쓰기 및 리매핑 동작을 통하여 저널링 기능을 수행할 수 있다. 모바일 램(4500)은 모바일 장치(4000)의 처리 동작 시 필요한 데이터를 임시로 저장하도록 구현될 수 있다.

[0105] 본 발명은 저널링 쓰기를 필요하는 하는 호스트에서 실행되고 있는 프로그램(운영체제, 응용프로그램)에도 적용이 가능하다. 운영체제(혹은, 커널)에서 저널링 기능을 지원하기 위한 수단(API ; application programming interface)를 제공할 수도 있다. 저널링 기능을 필요로하는 어플리케이션(데이터 베이스 프로그램, 예를 들어, Sqlite)에서 본 발명에 따른 기능을 사용하는 API를 이용함으로써, 중복 쓰기 제거의 기능을 적용할 수 있다.

[0106] 본 발명의 실시 예에 따른 모바일 장치(4000)는 저널링 혹은 랜덤 쓰기 동작시 중복 쓰기를 피하고, 저장 장치(4000)의 조건에 따라 맵 테이블을 관리함으로써 맵 테이블의 업데이트 회수를 줄일 수 있다.

[0107] 본 발명의 실시 예에 따른 메모리 시스템 혹은 저장 장치는 다양한 형태들의 패키지를 이용하여 실장될 수 있다. 실시 예에 있어서, 본 발명의 실시 예에 따른 메모리 시스템 혹은 저장 장치는 PoP(Package on Package), Ball grid arrays(BGAs), Chip scale packages(CSPs), Plastic Leaded Chip Carrier(PLCC), Plastic Dual In-Line Package(PDIP), Die in Waffle Pack, Die in Wafer Form, Chip On Board(COB), Ceramic Dual In-Line Package(CERDIP), Plastic Metric Quad Flat Pack(MQFP), Thin Quad Flatpack(TQFP), Small Outline(SOIC), Shrink Small Outline Package(SSOP), Thin Small Outline(TSOP), Thin Quad Flatpack(TQFP), System In Package(SIP), Multi Chip Package(MCP), Wafer-level Fabricated Package(WFP), Wafer-Level Processed Stack Package(WSP), 등과 같은 패키지들을 이용하여 실장될 수 있다.

[0108] 한편, 상술된 본 발명의 내용은 발명을 실시하기 위한 구체적인 실시 예들에 불과하다. 본 발명은 구체적이고 실제로 이용할 수 있는 수단 자체뿐 아니라, 장차 기술로 활용할 수 있는 추상적이고 개념적인 아이디어인 기술적 사상을 포함할 것이다.

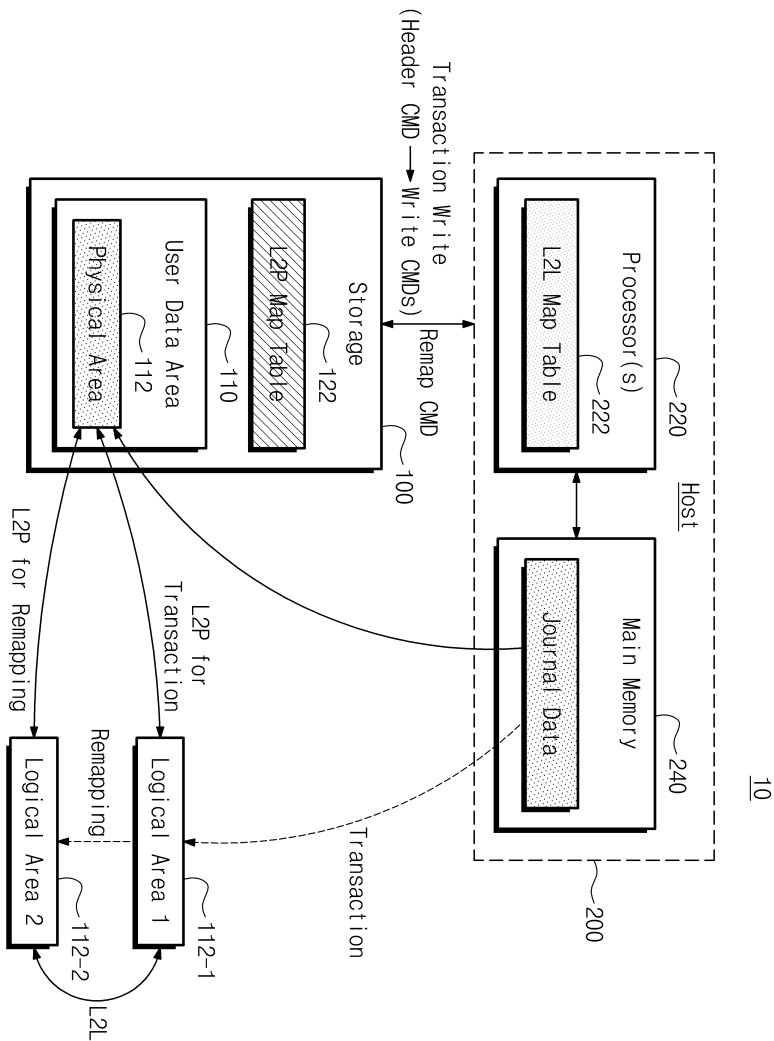
부호의 설명

- [0109] 10: 스토리지 시스템
- 100: 스토리지

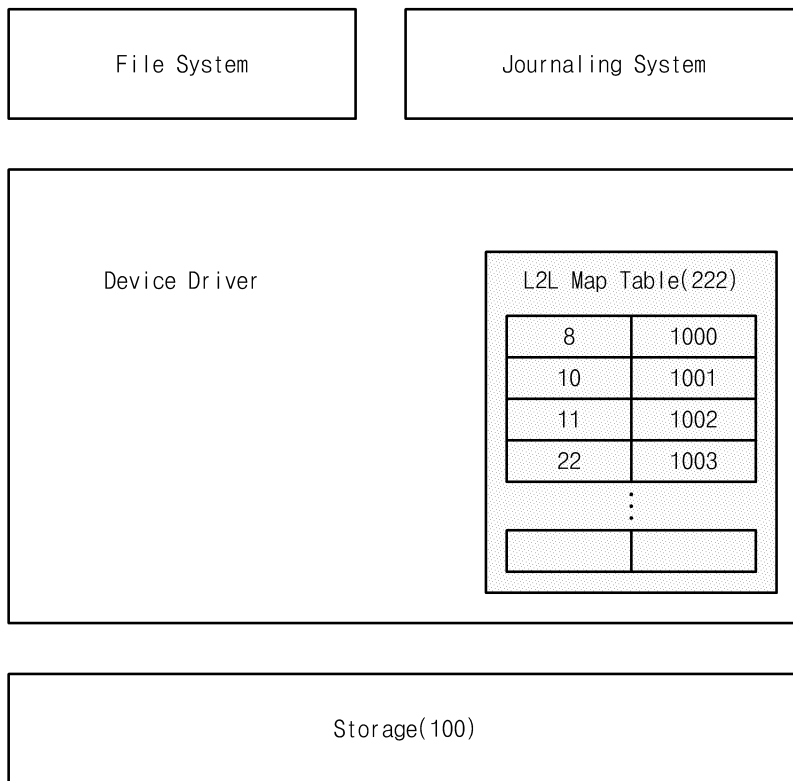
- 200: 호스트
- 220: 프로세서
- 240: 메인 메모리
- 222: L2L 맵 테이블
- 122: L2P 맵 테이블
- 110: 사용자 데이터 영역
- 112: 물리 영역
- 112-1: 제 1 논리 영역
- 112-2: 제 2 논리 영역
- Remap CMD: 리맵 명령
- Header CMD: 헤더 명령
- Write CMDs: 쓰기 명령들

도면

도면1



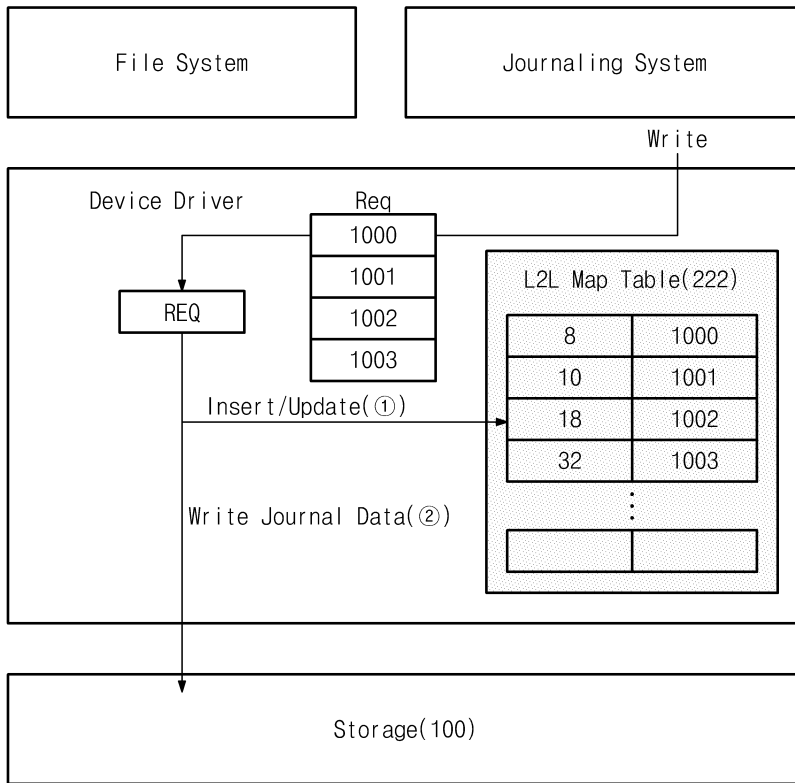
도면2



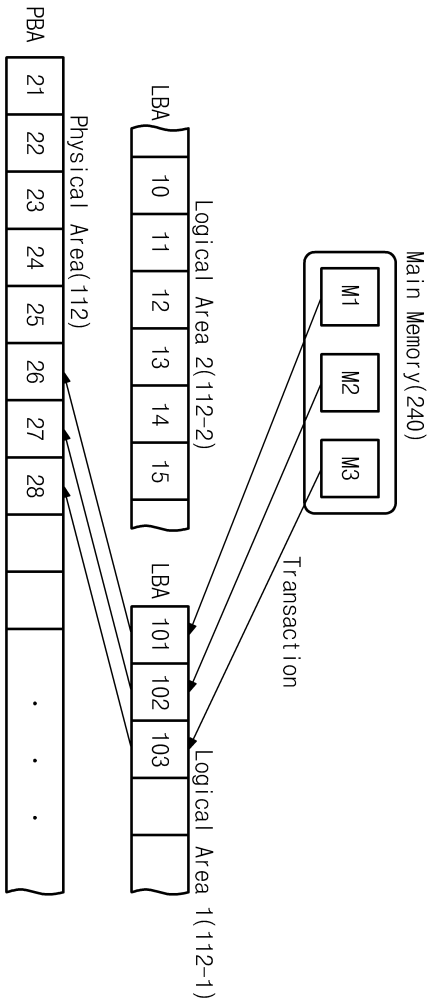
도면3

Logical Address 1	Logical Address 2	Length
-------------------	-------------------	--------

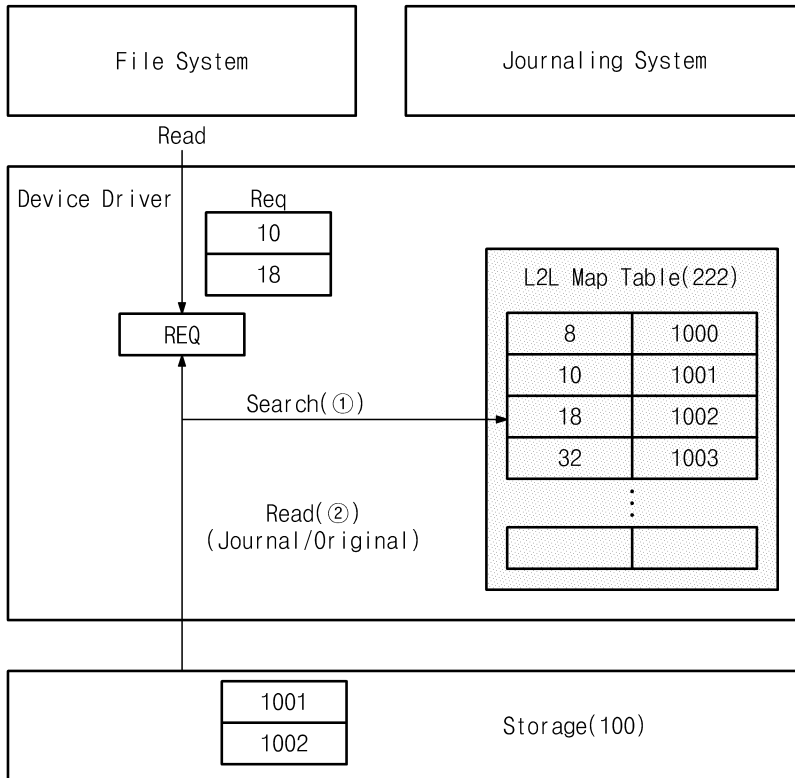
도면4



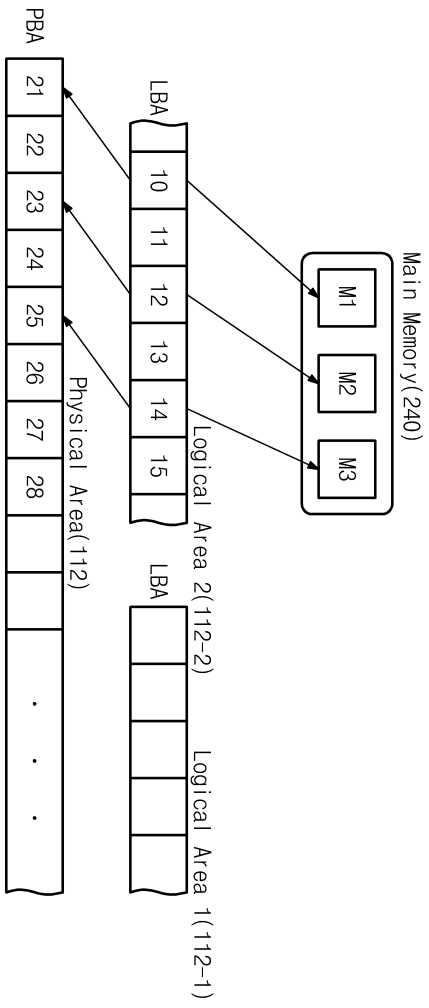
도면5



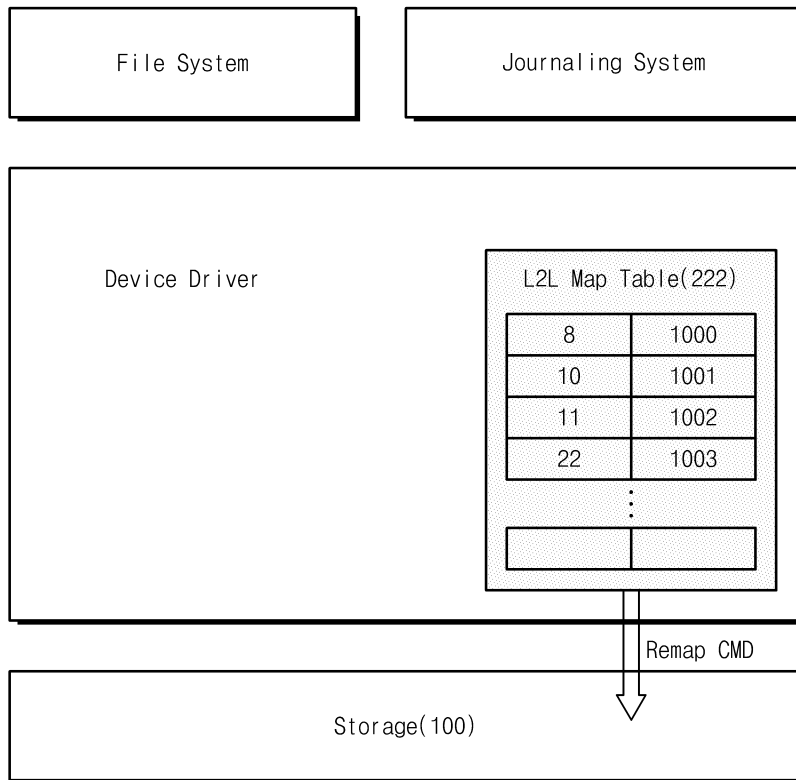
도면6



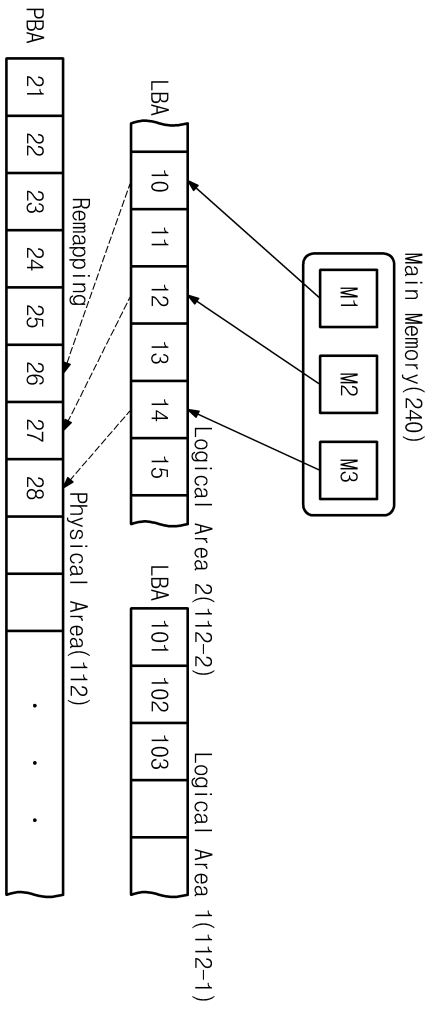
도면7



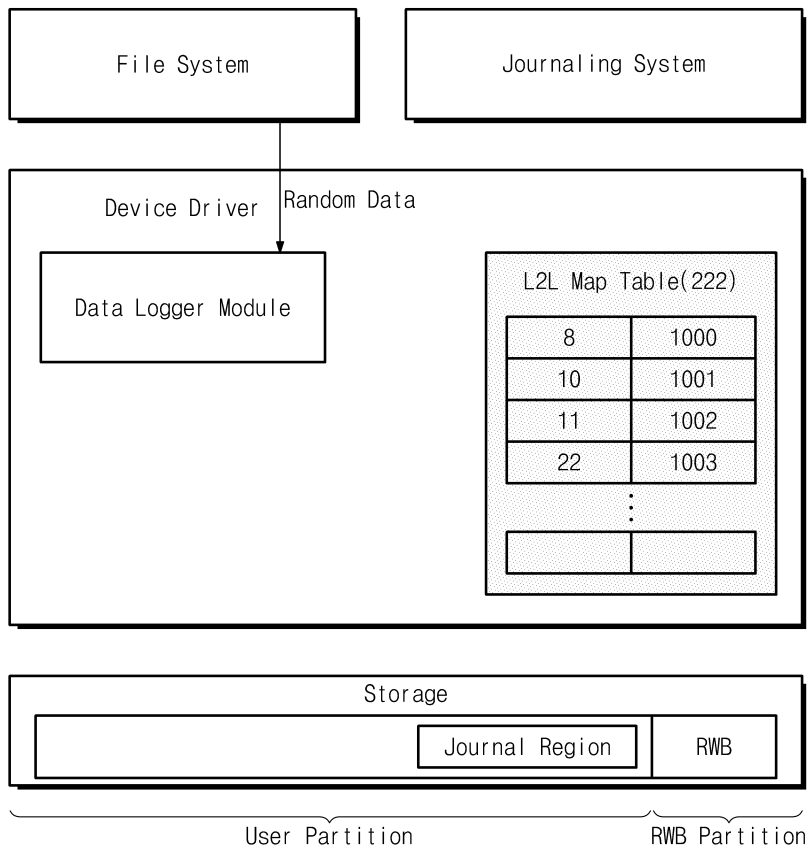
도면8



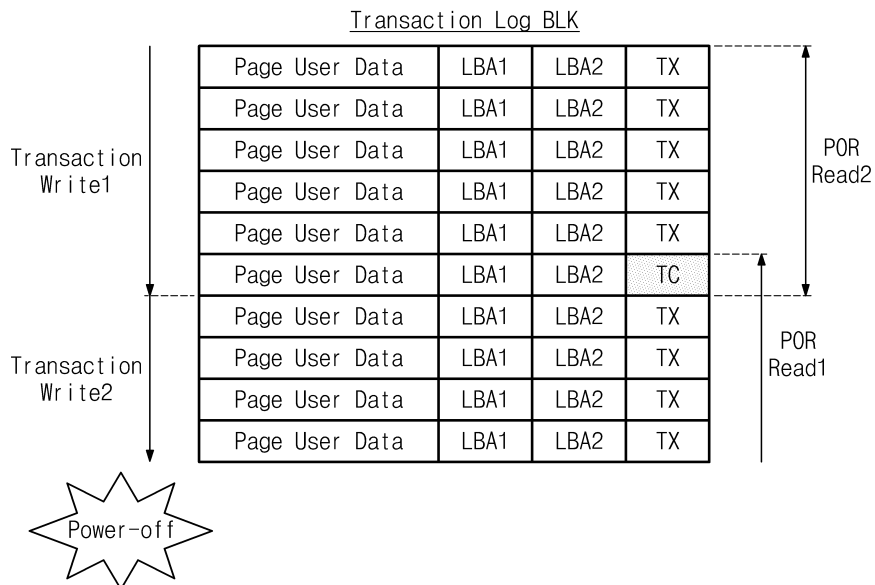
도면9



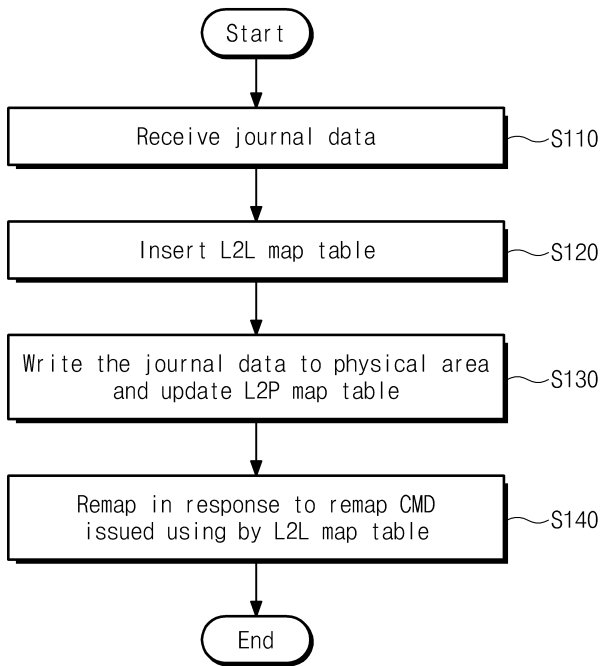
도면10



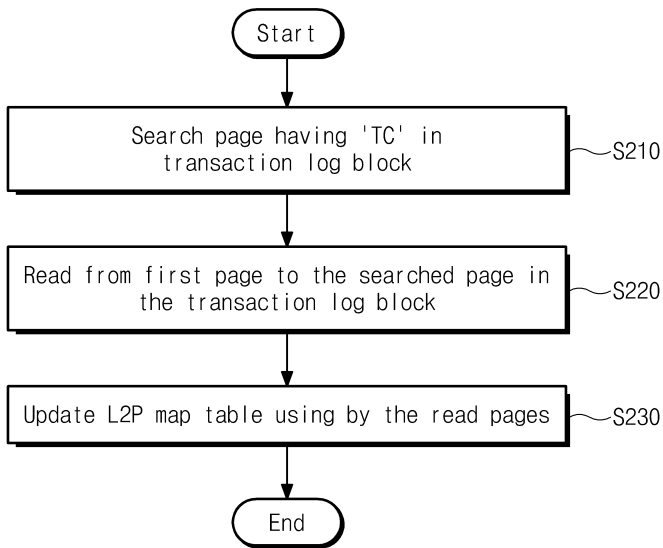
도면11



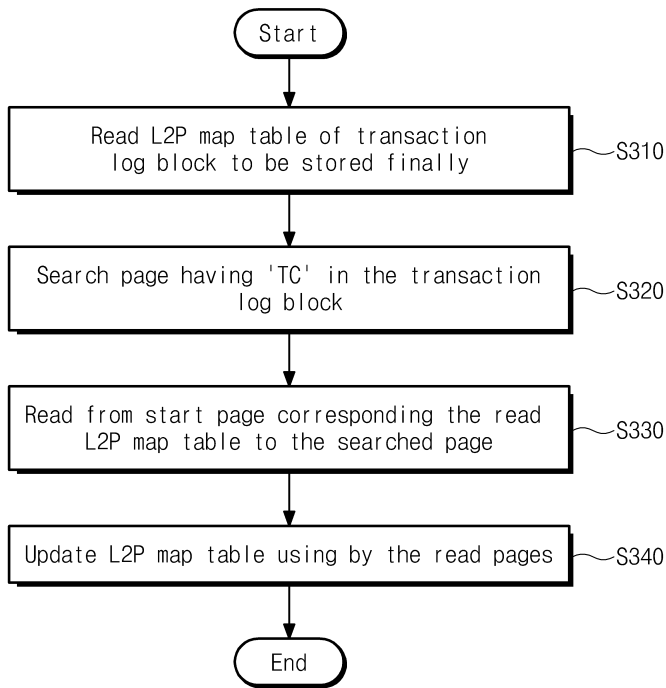
도면12



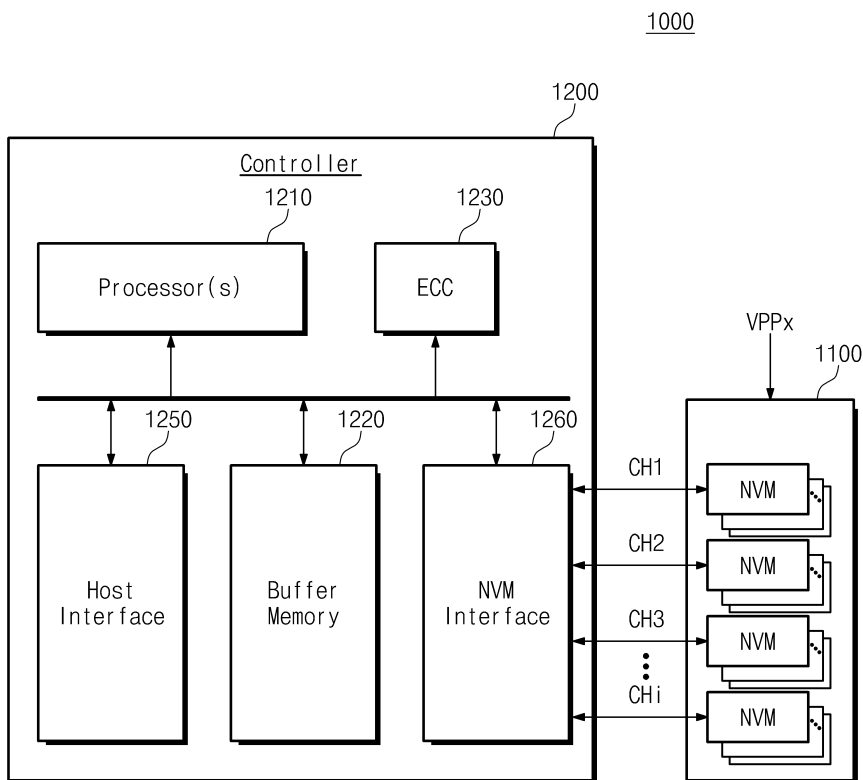
도면13



도면14

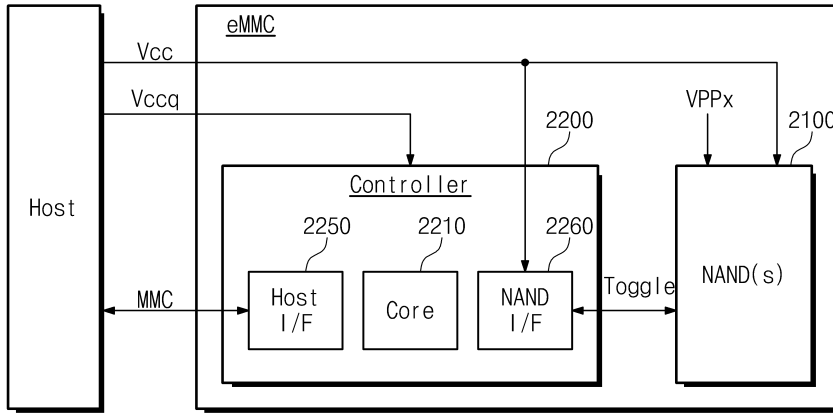


도면15



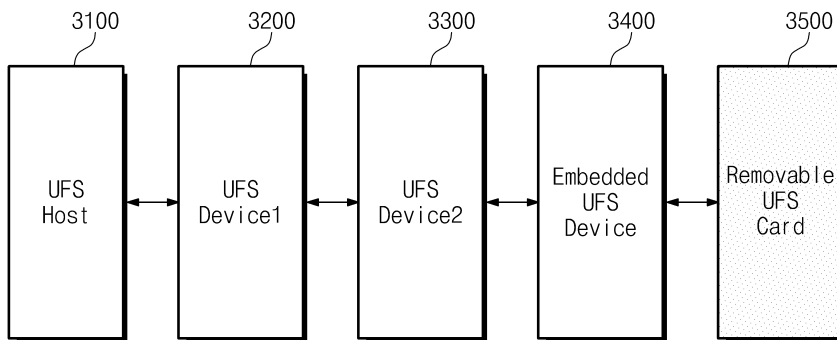
도면16

2000



도면17

3000



도면18

4000

