



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0095103
(43) 공개일자 2020년08월10일

(51) 국제특허분류(Int. Cl.)
G06F 12/02 (2018.01) G06F 3/06 (2006.01)
(52) CPC특허분류
G06F 12/0246 (2013.01)
G06F 3/0616 (2013.01)
(21) 출원번호 10-2019-0012769
(22) 출원일자 2019년01월31일
심사청구일자 없음

(71) 출원인
에스케이하이닉스 주식회사
경기도 이천시 부발읍 경충대로 2091
(72) 발명자
구덕희
경기도 수원시 영통구 센트럴파크로 100 광고센터
텔타운오드카운티 6404동 903호
백승걸
경기도 성남시 분당구 수내로 181 셋별마을우방아
파트 302동 1402호
(뒤편에 계속)
(74) 대리인
김성남

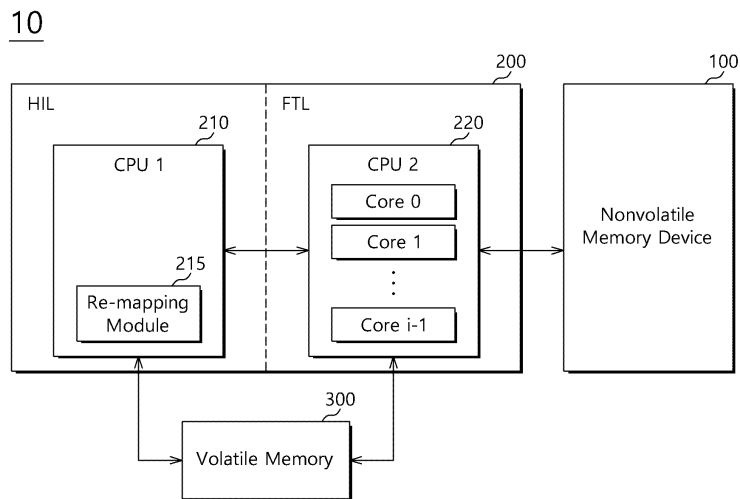
전체 청구항 수 : 총 14 항

(54) 발명의 명칭 데이터 저장 장치 및 그 동작 방법

(57) 요약

본 발명의 실시 예에 따른 데이터 저장 장치는 불휘발성 메모리 장치; 및 상기 불휘발성 메모리 장치의 동작을 제어하는 컨트롤러를 포함하고, 상기 컨트롤러는, 제1 CPU 및 복수의 코어들을 포함하는 제2 CPU를 포함한다. 상기 제1 CPU는 기 설정된 주기마다 상기 제2 CPU의 상기 복수의 코어들 각각에 대한 프로그램/소거(program/erase, P/E) 평균 카운트를 비교하고, 상기 복수의 코어들 중 상기 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하면, 호스트로부터 수신되는 논리 블록 어드레스들에 맵핑되는 코어를 변경하는 리-맵핑 동작을 수행한다.

대표도 - 도1



(52) CPC특허분류

G06F 3/064 (2013.01)

G06F 3/0658 (2013.01)

G06F 2212/7211 (2013.01)

(72) 발명자

김영호

경기도 성남시 분당구 불정로71번길 4

정석호

경기도 성남시 분당구 판교역로 98 백현마을7단지
아파트 704동 1101호

명세서

청구범위

청구항 1

불휘발성 메모리 장치; 및 상기 불휘발성 메모리 장치의 동작을 제어하는 컨트롤러를 포함하고,

상기 컨트롤러는, 제1 CPU 및 복수의 코어들을 포함하는 제2 CPU를 포함하고,

상기 제1 CPU는 기 설정된 주기마다 상기 제2 CPU의 상기 복수의 코어들 각각에 대한 프로그램/소거 (program/erase, P/E) 평균 카운트를 비교하고, 상기 복수의 코어들 중 상기 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하면, 호스트로부터 수신되는 논리 블록 어드레스들에 맵핑 되는 코어를 변경하는 리-맵핑 동작을 수행하는 데이터 저장 장치.

청구항 2

제1항에 있어서,

상기 데이터 저장 장치는, 휘발성 메모리를 더 포함하고,

상기 휘발성 메모리는,

상기 제2 CPU의 상기 복수의 코어들 각각에 대한 상기 P/E 평균 카운트로 구성된 프로그램/소거 카운트 테이블;

복수의 논리 블록 어드레스 그룹들 각각에 대한 코어 별 액세스 카운트로 구성된 코어 액세스 카운트 테이블; 및

상기 복수의 논리 블록 어드레스 그룹들 각각에 대한 코어 맵핑 오프셋으로 구성된 코어 맵핑 오프셋 테이블을 포함하는 데이터 저장 장치.

청구항 3

제2항에 있어서,

상기 제1 CPU는 상기 리-맵핑 동작을 수행하도록 구성된 리-맵핑 모듈을 포함하는 데이터 저장 장치.

청구항 4

제3항에 있어서,

상기 제1 CPU는 상기 제2 CPU의 상기 복수의 코어들 중 상기 제1 코어와 상기 제2 코어가 존재하면, 상기 리-맵핑 모듈을 구동시켜 상기 리-맵핑 동작을 수행하는 데이터 저장 장치.

청구항 5

제3항에 있어서,

상기 제1 코어의 P/E 평균 카운트가 상기 제2 코어의 P/E 평균 카운트보다 큰 경우,

상기 리-맵핑 모듈은 상기 코어 액세스 카운트 테이블을 참조하여 상기 복수의 논리 블록 어드레스 그룹들 중 상기 제1 코어의 액세스 카운트가 상기 제2 코어의 액세스 카운트보다 크고, 상기 제1 코어의 상기 액세스 카운트와 상기 제2 코어의 상기 액세스 카운트의 차이가 가장 큰 논리 블록 어드레스 그룹을 리-맵핑 대상 논리 블록 어드레스 그룹으로 선택하는 데이터 저장 장치.

청구항 6

제5항에 있어서,

상기 리-맵핑 모듈은 상기 코어 맵핑 오프셋 테이블에서 상기 리-맵핑 대상 논리 블록 어드레스 그룹에 대응하는 상기 코어 맵핑 오프셋을 재설정하는 데이터 저장 장치.

청구항 7

제6항에 있어서,

상기 리-맵핑 모듈은 리-맵핑 대상 논리 블록 어드레스 그룹에 대응하는 상기 코어 맵핑 오프셋을 상기 제1 코어부터 상기 제2 코어까지의 오프셋으로 변경하여 상기 코어 맵핑 오프셋을 재설정하는 데이터 저장 장치.

청구항 8

제6항에 있어서,

상기 리-맵핑 모듈은 상기 코어 맵핑 오프셋을 재설정하기 전에, 상기 리-맵핑 대상 논리 블록 어드레스 그룹에 포함된 논리 블록 어드레스들 각각에 대응하는 데이터를 리-맵핑될 코어의 전용 저장 영역으로 이동시키기 위한 지시를 상기 제2 CPU의 상기 복수의 코어들 각각에 제공하는 데이터 저장 장치.

청구항 9

제8항에 있어서,

상기 제2 CPU의 상기 복수의 코어들은 각각 상기 리-맵핑 모듈로부터 제공된 상기 지시에 따라 상기 리-맵핑 대상 논리 블록 어드레스 그룹에 포함된 상기 논리 블록 어드레스들 중 대응하는 적어도 하나의 논리 블록 어드레스에 대응하는 데이터를 상기 리-맵핑될 코어의 상기 전용 저장 영역으로 이동시키는 데이터 저장 장치.

청구항 10

불휘발성 메모리 장치; 및 상기 불휘발성 메모리 장치의 동작을 제어하되, 제1 CPU 및 복수의 코어들을 포함하는 제2 CPU를 갖는 컨트롤러를 포함하는 데이터 저장 장치의 동작 방법으로서,

기 설정된 주기마다 상기 제2 CPU의 상기 복수의 코어들 각각에 대한 프로그램/소거(program/erase, P/E) 평균 카운트를 비교하는 단계;

상기 제2 CPU의 상기 복수의 코어들 중 상기 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하는지 여부를 확인하는 단계; 및

상기 제1 코어와 상기 제2 코어가 존재하면, 호스트로부터 수신되는 논리 블록 어드레스들에 맵핑되는 코어를 변경하는 리-맵핑 동작을 수행하는 단계

를 포함하는 데이터 저장 장치의 동작 방법.

청구항 11

제10항에 있어서,

상기 제1 코어의 P/E 평균 카운트가 상기 제2 코어의 P/E 평균 카운트보다 큰 경우,

상기 리-맵핑 동작을 수행하는 단계는,

복수의 논리 블록 어드레스 그룹들 중 상기 제1 코어의 액세스 카운트가 상기 제2 코어의 액세스 카운트보다 크고, 상기 제1 코어의 상기 액세스 카운트와 상기 제2 코어의 상기 액세스 카운트의 차이가 가장 큰 논리 블록 어드레스 그룹을 리-맵핑 대상 논리 블록 어드레스 그룹으로 선택하는 단계; 및

상기 리-맵핑 대상 논리 블록 어드레스 그룹에 대응하는 코어 맵핑 오프셋을 재설정하는 단계

를 포함하는 데이터 저장 장치의 동작 방법.

청구항 12

제11항에 있어서,

상기 코어 맵핑 오프셋은 각 논리 블록 어드레스 그룹에 포함된 복수의 논리 블록 어드레스들 각각에 맵핑되는 코어를 결정하는 오프셋인 데이터 저장 장치의 동작 방법.

청구항 13

제12항에 있어서,

상기 코어 맵핑 오프셋을 재설정하는 단계는,

상기 리-맵핑 대상 논리 블록 어드레스 그룹에 대응하는 상기 코어 맵핑 오프셋을 상기 제1 코어로부터 상기 제2 코어까지의 오프셋으로 변경하는 단계

를 포함하는 데이터 저장 장치의 동작 방법.

청구항 14

제11항에 있어서,

상기 코어 맵핑 오프셋을 재설정하는 단계 이전에,

상기 리-맵핑 대상 논리 블록 어드레스 그룹에 포함된 논리 블록 어드레스들 각각에 대응하는 데이터를 리-맵핑 될 코어의 전용 저장 영역으로 이동시키는 단계

를 더 포함하는 데이터 저장 장치의 동작 방법.

발명의 설명

기술 분야

[0001] 본 발명은 전자 장치에 관한 것으로, 보다 구체적으로는 데이터 저장 장치 및 그 동작 방법에 관한 것이다.

배경 기술

[0002] 최근 컴퓨터 환경에 대한 패러다임(paradigm)이 언제, 어디서나 컴퓨터 시스템을 사용할 수 있도록 하는 유비쿼터스 컴퓨팅(ubiquitous computing)으로 전환되고 있다. 이로 인해 휴대폰, 디지털 카메라, 노트북 컴퓨터 등과 같은 휴대용 전자 장치의 사용이 급증하고 있다. 이와 같은 휴대용 전자 장치는 일반적으로 메모리 장치를 이용하는 데이터 저장 장치를 사용한다. 데이터 저장 장치는 휴대용 전자 장치에서 사용되는 데이터를 저장하기 위해서 사용된다.

[0003] 메모리 장치를 이용한 데이터 저장 장치는 기계적인 구동부가 없어서 안정성 및 내구성이 뛰어나며 정보의 액세스 속도가 매우 빠르고 전력 소모가 적다는 장점이 있다. 이러한 장점을 갖는 데이터 저장 장치는 USB(Universal Serial Bus) 메모리 장치, 다양한 인터페이스를 갖는 메모리 카드, UFS(Universal Flash Storage) 장치, 솔리드 스테이트 드라이브(Solid State Drive)를 포함한다.

발명의 내용

해결하려는 과제

[0004] 본 발명의 실시 예는 코어 별로 할당된 저장 영역들 간의 마모도를 평균화할 수 있는 데이터 저장 장치 및 그 동작 방법을 제공한다.

과제의 해결 수단

[0005] 본 발명의 실시 예에 따른 데이터 저장 장치는 불휘발성 메모리 장치; 및 상기 불휘발성 메모리 장치의 동작을 제어하는 컨트롤러를 포함하고, 상기 컨트롤러는, 제1 CPU 및 복수의 코어들을 포함하는 제2 CPU를 포함한다. 상기 제1 CPU는 기 설정된 주기마다 상기 제2 CPU의 상기 복수의 코어들 각각에 대한 프로그램/소거(program/erase, P/E) 평균 카운트를 비교하고, 상기 복수의 코어들 중 상기 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하면, 호스트로부터 수신되는 논리 블록 어드레스들에 맵핑 되는 코어를 변경하는 리-맵핑 동작을 수행한다.

[0006] 본 발명의 실시 예에 따른 데이터 저장 장치의 동작 방법은 기 설정된 주기마다 상기 제2 CPU의 상기 복수의 코어들 각각에 대한 프로그램/소거(program/erase, P/E) 평균 카운트를 비교하는 단계; 상기 제2 CPU의 상기 복수의 코어들 중 상기 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하는지 여부를 확인하는 단계; 및 상기 제1 코어와 상기 제2 코어가 존재하면, 호스트로부터 수신되는 논리 블록 어

드레스들에 맵핑되는 코어를 변경하는 리-맵핑 동작을 수행하는 단계를 포함한다.

발명의 효과

[0007] 본 실시 예에 따르면, 코어 별로 맵핑된 논리 블록 어드레스들을 리-맵핑하여 코어 별 저장 영역 간의 웨어-레벨링(wear-leveling)을 수행할 수 있다. 이에 따라, 코어 별 저장 영역 간의 마모도를 평균화할 수 있다.

도면의 간단한 설명

- [0008] 도 1은 본 발명의 실시 예에 따른 데이터 저장 장치를 나타낸 도면이다.
- 도 2는 HIL(host interface layer)의 제1 CPU에서 FTL(flash translation layer)의 제2 CPU의 각 코어로 논리 블록 어드레스들을 맵핑하는 것을 나타낸 도면이다.
- 도 3은 도 1의 휘발성 메모리를 나타낸 도면이다.
- 도 4는 FTL의 제2 CPU에 포함된 코어들 각각에 대한 P/E 평균 카운트들을 포함하는 P/E 평균 카운트 테이블을 나타낸 도면이다.
- 도 5는 논리 블록 어드레스 그룹들 각각의 코어 별 액세스 카운트를 포함하는 코어 액세스 카운트 테이블을 나타낸 도면이다.
- 도 6은 논리 블록 어드레스 그룹 별 코어 맵핑 오프셋을 포함하는 코어 맵핑 오프셋 테이블을 나타낸 도면이다.
- 도 7은 본 발명의 실시 예에 따른 데이터 저장 장치의 동작 방법을 나타낸 순서도이다.
- 도 8은 본 발명의 실시 예에 따른 솔리드 스테이트 드라이브(SSD)를 포함하는 데이터 처리 시스템을 예시적으로 나타낸 도면이다.
- 도 9는 도 8의 컨트롤러의 구성을 예시적으로 나타낸 도면이다.
- 도 10은 본 발명의 실시 예에 따른 데이터 저장 장치를 포함하는 데이터 처리 시스템을 예시적으로 나타낸 도면이다.
- 도 11은 본 발명의 실시 예에 따른 데이터 저장 장치를 포함하는 데이터 처리 시스템을 예시적으로 나타낸 도면이다.
- 도 12는 본 발명의 실시 예에 따른 데이터 저장 장치를 포함하는 네트워크 시스템을 예시적으로 나타낸 도면이다.
- 도 13은 본 발명의 실시 예에 따른 데이터 저장 장치에 포함된 불휘발성 메모리 장치를 예시적으로 나타낸 블록도이다.

발명을 실시하기 위한 구체적인 내용

- [0009] 이하, 첨부한 도면에 의거하여 본 발명의 바람직한 실시 예를 설명하도록 한다.
- [0010] 도 1은 본 발명의 실시 예에 따른 데이터 저장 장치(10)의 구성을 예시적으로 나타낸 도면이다.
- [0011] 도 1을 참조하면, 본 실시 예에 따른 데이터 저장 장치(10)는 휴대폰, MP3 플레이어, 랩탑 컴퓨터, 데스크탑 컴퓨터, 게임기, TV, 차량용 인포테인먼트(in-vehicle infotainment) 시스템 등과 같은 호스트(도시되지 않음)에 의해서 액세스되는 데이터를 저장할 수 있다. 데이터 저장 장치(10)는 메모리 시스템으로 불릴 수 있다.
- [0012] 데이터 저장 장치(10)는 호스트와 연결되는 인터페이스 프로토콜에 따라서 다양한 종류의 저장 장치들 중 어느 하나로 제조될 수 있다. 예를 들어, 데이터 저장 장치(10)는 솔리드 스테이트 드라이브(solid state drive, SSD), MMC, eMMC, RS-MMC, micro-MMC 형태의 멀티미디어 카드(multimedia card), SD, mini-SD, micro-SD 형태의 시큐어 디지털(secure digital) 카드, USB(universal storage bus) 저장 장치, UFS(universal flash storage) 장치, PCMCIA(personal computer memory card international association) 카드 형태의 저장 장치, PCI(peripheral component interconnection) 카드 형태의 저장 장치, PCI-E(PCI-express) 카드 형태의 저장 장치, CF(compact flash) 카드, 스마트 미디어(smart media) 카드, 메모리 스틱(memory stick) 등과 같은 다양한 종류의 저장 장치들 중 어느 하나로 구성될 수 있다.

- [0013] 데이터 저장 장치(10)는 다양한 종류의 패키지(package) 형태들 중 어느 하나로 제조될 수 있다. 예를 들어, 데이터 저장 장치(10)는 POP(package on package), SIP(system in package), SOC(system on chip), MCP(multi-chip package), COB(chip on board), WFP(wafer-level fabricated package), WSP(wafer-level stack package) 등과 같은 다양한 종류의 패키지 형태들 중 어느 하나로 제조될 수 있다.
- [0014] 데이터 저장 장치(10)는 불휘발성 메모리 장치(100) 및 컨트롤러(200)를 포함할 수 있다.
- [0015] 불휘발성 메모리 장치(100)는 데이터 저장 장치(10)의 저장 매체로서 동작할 수 있다. 불휘발성 메모리 장치(100)는 메모리 셀에 따라서 낸드(NAND) 플래시 메모리 장치, 노어(NOR) 플래시 메모리 장치, 강유전체 커패시터를 이용한 강유전체 램(ferroelectric random access memory, FRAM), 터널알(tunneling magneto-resistive, TMR) 막을 이용한 마그네틱 램(magnetic random access memory, MRAM), 칼코겐 화합물(chalcogenide alloys)을 이용한 상 변화 램(phase change random access memory, PRAM), 전이 금속 화합물(transition metal oxide)을 이용한 저항성 램(resistive random access memory, ReRAM) 등과 같은 다양한 형태의 불휘발성 메모리 장치들 중 어느 하나로 구성될 수 있다.
- [0016] 도면의 간략화를 위해, 도 1에서는 불휘발성 메모리 장치(100)를 하나의 블록으로 도시하였으나, 불휘발성 메모리 장치(100)는 복수의 메모리 칩들을 포함할 수 있다. 본 실시 예는 복수의 메모리 칩들로 구성된 불휘발성 메모리 장치(100)를 포함하는 데이터 저장 장치(10)에 대해서도 동일하게 적용될 수 있다.
- [0017] 불휘발성 메모리 장치(100)는 복수의 비트라인들(도시되지 않음) 및 복수의 워드라인들(도시되지 않음)이 교차하는 영역들에 각각 배치되는 복수의 메모리 셀들을 갖는 메모리 셀 어레이(도시되지 않음)를 포함할 수 있다. 메모리 셀 어레이는 복수의 메모리 블록들을 포함할 수 있고, 복수의 메모리 블록들은 각각 복수의 페이지들을 포함할 수 있다.
- [0018] 예를 들어, 메모리 셀 어레이의 각 메모리 셀은 하나의 비트를 저장하는 싱글 레벨 셀(single level cell, SLC), 2 비트의 데이터를 저장할 수 있는 멀티 레벨 셀(multi-level cell, MLC), 3 비트의 데이터를 저장할 수 있는 트리플 레벨 셀(triple level cell, TLC) 또는 4 비트의 데이터를 저장할 수 있는 쿼드 레벨 셀(quad level cell, QLC)일 수 있다. 메모리 셀 어레이(110)는 싱글 레벨 셀, 멀티 레벨 셀, 트리플 레벨 셀, 및 쿼드 레벨 셀 중 적어도 하나 이상을 포함할 수 있다. 예를 들어, 메모리 셀 어레이(110)는 2차원 수평 구조의 메모리 셀들을 포함할 수도 있고, 또는 3차원 수직 구조의 메모리 셀들을 포함할 수도 있다.
- [0019] 컨트롤러(200)는 데이터 저장 장치(10)의 제반 동작을 제어할 수 있다. 컨트롤러(200)는 호스트로부터 수신된 요청을 처리할 수 있다. 컨트롤러(200)는 호스트로부터 수신된 요청들에 근거하여 불휘발성 메모리 장치(100)의 동작을 제어하기 위한 제어 신호들을 생성하고, 생성된 제어 신호들을 불휘발성 메모리 장치(100)로 제공할 수 있다. 컨트롤러(200)는 호스트 인터페이스 계층(host interface layer, HIL)(이하, 'HIL' 이라 함) 및 플래시 변환 계층(flash translation layer, FTL) (이하, 'FTL' 이라 함)을 포함할 수 있다.
- [0020] 컨트롤러(200)의 HIL은 중앙 처리 장치(central processing unit, CPU)를 포함할 수 있다. 설명의 편의를 위해, HIL의 CPU를 '제1 CPU(CPU 1)' 라 한다. 또한, 컨트롤러(200)의 FTL은 중앙 처리 장치(central processing unit, CPU)를 포함할 수 있다. 설명의 편의를 위해, FTL의 CPU를 '제2 CPU(CPU 2)' 라 한다.
- [0021] 제1 CPU(210)는 호스트의 프로토콜에 대응하여 호스트와 데이터 저장 장치(10) 사이를 인터페이싱하도록 구성될 수 있다. 이에 따라, 제1 CPU(210)는 프로토콜 CPU(protocol CPU)로도 불릴 수 있다. 예를 들어, 제1 CPU(210)는 USB(universal serial bus), UFS(universal flash storage), MMC(multimedia card), PATA(parallel advanced technology attachment), SATA(serial advanced technology attachment), SCSI(small computer system interface), SAS(serial attached SCSI), PCI(peripheral component interconnection), PCI-E(PCI express) 프로토콜 중 어느 하나를 통해 호스트와 통신할 수 있다.
- [0022] 제1 CPU(210)는 호스트로부터 전송되는 커맨드들을 수신하고, 수신된 커맨드들을 스케줄링하여 제2 CPU(220)로 제공하도록 구성될 수 있다. 제1 CPU(210)는 호스트로부터 커맨드와 함께 수신된 논리 블록 어드레스(logical block address, LBA)를 제2 CPU(220)로 제공할 수 있다. 이때, 제2 CPU(220)가 복수의 코어(core)를 포함하는 경우, 제1 CPU(210)는 호스트로부터 수신된 논리 블록 어드레스들을 각각 대응하는 코어로 제공할 수 있다. 이에 대해서는 이후 도 2를 참조하여 상세히 설명한다.
- [0023] 제1 CPU(210)는 리-맵핑 모듈(215)을 포함할 수 있다. 리-맵핑 모듈(215)은 특정 조건에서 제1 CPU(210)에 의해 구동될 수 있다. 여기에서, 특정 조건은 리-맵핑을 수행해야 할 조건을 의미한다. 일 실시 예에서, 특정 조건은

제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1) 중에서 프로그램/소거 평균 카운트(P/E average count)가 기 설정된 임계 값 이상 차이가 나는 2개의 코어가 존재하는 경우일 수 있다.

- [0024] 특정 조건에 도달하는 경우 즉, 복수의 코어들(Core 0 ~ Core i-1) 중 프로그램/소거 평균 카운트가 기 설정된 임계 값 이상 차이가 나는 2 개의 코어가 존재하는 경우, 제1 CPU(210)는 리-맵핑 모듈(215)을 구동시킬 수 있다.
- [0025] 리-맵핑 모듈(215)은 복수의 논리 블록 어드레스 그룹들 중 리-맵핑 대상 논리 블록 어드레스 그룹을 선택하고, 리-맵핑 대상 논리 블록 어드레스 그룹에 대한 코어 맵핑 오프셋을 변경할 수 있다. 여기에서, 코어 맵핑 오프셋은 논리 블록 어드레스 그룹 내에 포함된 복수의 논리 블록 어드레스들 각각에 맵핑될 코어를 결정하기 위한 오프셋을 의미할 수 있다.
- [0026] 제1 CPU(210)는 호스트로부터 수신된 데이터(예컨대, 라이트 데이터)를 휘발성 메모리(300)에 임시 저장할 수 있다. 또한, 제1 CPU(210)는 휘발성 메모리(300)에 임시 저장된 데이터(예컨대, 리드 데이터)를 호스트로 전송할 수 있다. 이를 위해, 컨트롤러(200)는 제1 CPU(210)의 제어에 따라 호스트로부터 수신된 라이트 데이터를 휘발성 메모리(300)로 전송하거나 또는 휘발성 메모리(300)에 저장된 리드 데이터를 호스트로 전송하도록 구성된 별도의 데이터 전송 블록(도시하지 않음)을 더 포함할 수도 있다.
- [0027] 제2 CPU(220)는 복수의 코어들(Core 0 ~ Core i-1)을 포함할 수 있다. 복수의 코어들(Core 0 ~ Core i-1)에는 각각 불휘발성 메모리 장치(100)의 일부 저장 영역이 할당될 수 있다. 예컨대, 불휘발성 메모리 장치(100)는 복수의 저장 영역들을 포함할 수 있고, 복수의 저장 영역들 중 적어도 하나의 저장 영역이 각 코어(Core 0 ~ Core i-1)의 전용 저장 영역으로서 할당될 수 있다. 여기에서, 저장 영역은 다이(die) 또는 메모리 칩(memory chip)을 의미할 수 있다.
- [0028] 복수의 코어들(Core 0 ~ Core i-1)은 각각 할당된 전용 저장 영역에 대한 동작 제어 및 관리를 독립적으로 수행할 수 있다. 예를 들어, 복수의 코어들(Core 0 ~ Core i-1)은 각각 전용 저장 영역에서 쓰기 동작, 읽기 동작, 및 소거 동작 등이 수행되도록 전용 저장 영역을 제어할 수 있다. 또한, 복수의 코어들(Core 0 ~ Core i-1)은 각각 전용 저장 영역에 대한 맵핑 정보를 독립적으로 관리하고, 전용 저장 영역에 대한 가비지 컬렉션(garbage collection), 웨어-레벨링(wear-leveling)을 독립적으로 수행할 수 있다.
- [0029] 복수의 코어들(Core 0 ~ Core i-1)은 각각 휘발성 메모리(300)(또는 도시하지 않은 컨트롤러(200)의 내부 메모리)에 로딩된 펌웨어 또는 소프트웨어의 구동을 통해서 데이터 저장 장치(10)의 제반 동작을 제어할 수 있다. 복수의 코어들(Core 0 ~ Core i-1)은 각각 펌웨어 또는 소프트웨어와 같은 코드 형태의 명령(instruction) 또는 알고리즘을 해독하고 구동할 수 있다. 이에 따라, 복수의 코어들(Core 0 ~ Core i-1)은 각각 FTL(flash translation layer) 코어로도 불릴 수 있다.
- [0030] 제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1)은 각각 제1 CPU(210)를 통해 제공된 커맨드에 근거하여 불휘발성 메모리 장치(100)의 동작을 제어하기 위한 제어 신호들을 생성하고, 생성된 제어 신호들을 불휘발성 메모리 장치(100)로 제공할 수 있다. 제어 신호들은 불휘발성 메모리 장치(100)를 제어하기 위한 커맨드, 어드레스, 동작 제어 신호 등을 포함할 수 있다.
- [0031] 복수의 코어들(Core 0 ~ Core i-1)은 각각 휘발성 메모리(300)에 저장된 라이트 데이터를 불휘발성 메모리 장치(100)의 대응하는 저장 영역으로 제공하거나, 또는 대응하는 저장 영역으로부터 독출된 리드 데이터를 휘발성 메모리(300)에 저장할 수 있다. 이를 위해, 컨트롤러(200)는 복수의 코어들(Core 0 ~ Core i-1) 각각의 제어에 따라 휘발성 메모리(300)에 저장된 라이트 데이터를 불휘발성 메모리 장치(100)로 전송하거나 또는 불휘발성 메모리 장치(100)로부터 독출된 리드 데이터를 휘발성 메모리(300)로 전송하도록 구성된 별도의 데이터 전송 블록(도시하지 않음)을 더 포함할 수도 있다.
- [0032] 휘발성 메모리(300)는 동적 랜덤 액세스 메모리(DRAM)와 같은 랜덤 액세스 메모리로 구성될 수 있으나, 특별히 이에 한정되는 것은 아니다. 도 1에서는 컨트롤러(200) 외부에 구비된 휘발성 메모리(300)만을 도시하였으나, 특별히 이에 한정되는 것은 아니며, 컨트롤러(200)의 내부에도 메모리가 구비될 수 있다. 컨트롤러(200)의 내부 메모리는 정적 랜덤 액세스 메모리(SRAM)일 수 있으나, 특별히 이에 한정되는 것은 아니다.
- [0033] 휘발성 메모리(300)는 호스트로부터 불휘발성 메모리 장치(100)로 전송될 라이트 데이터 또는 불휘발성 메모리 장치(100)로부터 호스트로 전송될 리드 데이터를 임시 저장하도록 구성될 수 있다. 또한, 휘발성 메모리(300)는 불휘발성 메모리 장치(100)에 대한 맵 데이터(또는 맵핑 정보)를 저장하도록 구성될 수 있다. 본 실시 예를 설명함에 있어서 맵 데이터와 맵핑 정보는 상호 교환적으로 사용될 수 있다. 휘발성 메모리(300)의 내부 구성에

대해서는 이후 도 3을 참조하여 상세히 설명한다.

- [0034] 도 2는 제1 CPU(210)에서 제2 CPU(220)의 각 코어(Core 0 ~ Core i-1)에 맵핑되는 논리 블록 어드레스를 판단하고, 각 코어(Core 0 ~ Core i-1)로 맵핑된 논리 블록 어드레스를 제공하는 것을 나타낸 도면이다.
- [0035] 도 2를 참조하면, 불휘발성 메모리 장치(100)는 복수의 다이(D)들(또는, 복수의 저장 영역들)을 포함할 수 있다. 제2 CPU(220)의 각 코어(Core 0 ~ Core i-1)에는 j개의 다이들(D1-Dj)이 할당 및 연결될 수 있다. 여기에서, j는 2 이상의 자연수일 수 있다. 도 2에서는 제2 CPU(220)의 각 코어(Core 0 ~ Core i-1)에 동일한 수의 다이들 즉, j개의 다이들(D1-Dj)이 할당 및 연결된 것으로 도시하였으나, 각 코어(Core 0 ~ Core i-1)에 할당 및 연결되는 다이들의 수는 서로 다를 수 있다. 제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1)은 각각 할당된 다이들(D1-Dj)에 대한 동작 제어 및 관리를 독립적으로 수행할 수 있다.
- [0036] 한편, 제1 CPU(210)는 호스트(host)로부터 수신된 논리 블록 어드레스들(LBAs)을 각각 대응하는 제2 CPU(220)의 코어로 제공할 수 있다. 일 실시 예에서, 제1 CPU(210)는 모듈러 연산(modular arithmetic)을 이용하여 호스트로부터 수신된 논리 블록 어드레스에 맵핑되는 코어를 판단하고, 해당 코어에 수신된 논리 블록 어드레스를 제공할 수 있다.
- [0037] 예를 들어, 호스트로부터 수신된 논리 블록 어드레스가 'LBA0' 이면, 제1 CPU(210)는 논리 블록 어드레스 번호 '0' 을 제2 CPU(220)의 코어 수(예컨대, 'i')로 나눈 나머지 값 즉, '0' 을 논리 블록 어드레스 'LBA0' 에 맵핑되는 코어 번호로 판단하고, 논리 블록 어드레스 'LBA0' 을 제2 CPU(220)의 코어 0(Core 0)으로 제공할 수 있다. 이와 같은 방식으로, 제1 CPU(210)는 호스트로부터 수신된 논리 블록 어드레스들 각각에 맵핑되는 코어를 판단하고, 각 코어에 대응하는 논리 블록 어드레스를 제공할 수 있다. 도 2에서, 'LBA#0', 'LBA#1', 'LBA#i-1' 은 각각 제2 CPU(220)의 코어 수로 나눈 나머지 값이 '0', '1', 'i-1' 인 논리 블록 어드레스들을 나타낸다.
- [0038] 이와 같이, 제2 CPU(220)의 각 코어(Core 0 ~ Core i-1) 별로 제공되는 논리 블록 어드레스들이 연산에 의해 결정되고 고정적임에 따라, 호스트로부터 특정 논리 블록 어드레스에 대한 프로그램 커맨드가 반복적으로 수신되면, 특정 논리 블록 어드레스에 맵핑된 코어의 전용 다이들 중 특정 다이에 대한 프로그램/소거 카운트(program/erase count)가 급격히 증가할 수 있다. 그 결과, 특정 논리 블록 어드레스에 맵핑된 코어의 특정 다이는 다른 코어들에 할당된 다이들보다 빨리 마모될 수 있다. 즉, 제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1)의 다이들 간의 불균일한 마모가 초래되어 성능 하락에 영향을 미칠 수 있다.
- [0039] 따라서, 본 실시 예에서는 특정 조건에서 코어들 간의 논리 블록 어드레스 리-맵핑을 수행한다. 예를 들어, 제1 CPU(210)는 기 설정된 주기마다 제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1) 각각의 P/E 평균 카운트를 비교하고, 복수의 코어들(Core 0 ~ Core i-1) 중 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하는지 여부를 확인한다. 제1 코어와 제2 코어가 존재하면, 제1 CPU(210)는 특정 조건 즉, 코어들 간의 논리 블록 어드레스 리-맵핑을 수행해야 할 조건에 도달한 것으로 판단하고 리-맵핑 모듈(215)을 구동시킬 수 있다. 이후부터는 설명의 편의를 위해, 제1 코어의 P/E 평균 카운트가 제2 코어의 P/E 평균 카운트보다 큰 것으로 가정한다.
- [0040] 리-맵핑 모듈(215)은 복수의 논리 블록 어드레스 그룹들 중에서 제1 코어의 액세스 카운트가 제2 코어의 액세스 카운트보다 큰 제1 조건과 제1 코어의 액세스 카운트와 제2 코어의 액세스 카운트 간의 차이가 가장 큰 제2 조건을 만족하는 논리 블록 어드레스 그룹을 리-맵핑 대상 논리 블록 어드레스 그룹으로 선택할 수 있다. 또한, 리-맵핑 모듈(215)은 리-맵핑 대상 논리 블록 어드레스 그룹의 논리 블록 어드레스들 각각에 맵핑되는 코어를 변경하기 위한 코어 맵핑 오프셋을 재설정할 수 있다. 또한, 코어 맵핑 오프셋을 재설정하기 전에, 리-맵핑 모듈(215)은 리-맵핑 대상 논리 블록 어드레스 그룹의 논리 블록 어드레스들 각각에 대응하는 물리 블록 어드레스들에 저장된 데이터를 이동시키기 위한 지시를 각 코어(Core 0 ~ Core i-1)로 제공할 수 있다.
- [0041] 도 3은 도 1의 휘발성 메모리(300)의 구성을 나타낸 도면이다.
- [0042] 도 3을 참조하면, 휘발성 메모리(300)는 호스트로부터 불휘발성 메모리 장치(100)로 전송될 라이트 데이터 및 불휘발성 메모리 장치(100)로부터 독출되어 호스트로 전송될 리드 데이터 등을 임시로 저장하는 사용자 데이터 영역(310), 펌웨어 등과 같은 소프트웨어 및 소프트웨어를 구동시키는데 필요한 메타 데이터 등을 저장하는 메타 데이터 영역(320) 및 맵 데이터를 저장하는 맵 데이터 영역(330)을 포함할 수 있다.
- [0043] 메타 데이터 영역(320)에는 제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1) 각각의 P/E 평균 카운트를 저장하도록 구성된 P/E 평균 카운트 테이블(P/E ACT), 논리 블록 어드레스 그룹 별로 각 코어(Core 0 ~ Core i-1)에

대한 액세스 카운트를 저장하도록 구성된 코어 액세스 카운트 테이블(CACT), 및 논리 블록 어드레스 그룹 별 코어 맵핑 오프셋을 저장하도록 구성된 코어 맵핑 오프셋 테이블(CMOT)이 저장될 수 있다. P/E 평균 카운트 테이블(P/E ACT), 코어 액세스 카운트 테이블(CACT) 및 코어 맵핑 오프셋 테이블(CMOT)은 HIL의 제1 CPU(210)에 의해 생성되고, 저장되고, 갱신될 수 있으나, 특별히 이에 한정되는 것은 아니다.

- [0044] 제1 CPU(210)는 기 설정된 주기마다 휘발성 메모리(300)에 저장된 P/E 평균 카운트 테이블(P/E ACT)을 참조하여 복수의 코어들(Core 0 ~ Core i-1) 각각의 P/E 평균 카운트를 비교하고, 복수의 코어들(Core 0 ~ Core i-1) 중 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하는지 여부를 확인할 수 있다. 제1 코어와 제2 코어가 존재하면, 제1 CPU(210)는 리-맵핑 모듈(215)을 구동시킬 수 있다.
- [0045] P/E 평균 카운트 테이블(P/E ACT)은 코어 별로 각 코어(Core 0 ~ Core i-1)의 전용 다이들(또는 저장 영역들) 각각의 P/E 카운트의 평균 값을 포함할 수 있다.
- [0046] 리-맵핑 모듈(215)은 휘발성 메모리(300)에 저장된 코어 액세스 카운트 테이블(CACT)을 참조하여, 제1 코어의 액세스 카운트가 제2 코어의 액세스 카운트보다 큰 제1 조건과 제1 코어의 액세스 카운트와 제2 코어의 액세스 카운트 간의 차이가 가장 큰 제2 조건을 만족하는 논리 블록 어드레스 그룹을 리-맵핑 대상 논리 블록 어드레스 그룹으로 선택할 수 있다.
- [0047] 코어 액세스 카운트 테이블(CACT)은 복수의 논리 블록 어드레스 그룹들(LBAG0 ~ LBAGn) 각각에 대한 코어 별 액세스 카운트들(Core 0 access count ~ Core i-1 access count)을 포함할 수 있다. 하나의 논리 블록 어드레스 그룹은 복수의 논리 블록 어드레스들을 포함하고, 복수의 코어들(Core 0 ~ Core i-1) 각각에는 복수의 논리 블록 어드레스들 중 적어도 하나의 논리 블록 어드레스가 맵핑될 수 있다. 예를 들어, 도 5를 참조하면, 논리 블록 어드레스 그룹(LBAG0)에 포함된 복수의 논리 블록 어드레스들 중 코어 수 'i' 로 나눈 나머지 값이 '0' 인 적어도 하나의 논리 블록 어드레스 'LBA#0' 은 'Core 0' 에 맵핑될 수 있다. 마찬가지로, 논리 블록 어드레스 그룹(LBAG0)에 포함된 복수의 논리 블록 어드레스들 중 코어 수 'i' 로 나눈 나머지 값이 '1' 내지 'i-1' 인 논리 블록 어드레스들은 각각 'Core 1' 내지 'Core i-1' 에 맵핑될 수 있다.
- [0048] 따라서, 도 5에 도시된 코어 별 액세스 카운트들(Core 0 access count ~ Core i-1 access count)은 각 코어 별로 맵핑된 논리 블록 어드레스들에 대한 액세스 카운트의 합을 의미할 수 있다. 예컨대, 논리 블록 어드레스 그룹 0(LBAG0)에서 'Core 0' 의 액세스 카운트는 LBAG0에 포함된 논리 블록 어드레스들 중 'Core 0' 에 맵핑된 논리 블록 어드레스들 각각에 대한 액세스 카운트의 합을 의미한다.
- [0049] 이와 같이, 본 실시 예에서, P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어(제1 코어의 P/E 카운트 > 제2 코어의 P/E 카운트)가 존재할 때, 제1 코어의 액세스 카운트가 제2 코어의 액세스 카운트보다 크고, 제1 코어와 제2 코어의 액세스 카운트의 차이 값이 가장 큰 논리 블록 어드레스 그룹을 리-맵핑 대상 논리 블록 어드레스 그룹으로 선택하는 것은 해당 논리 블록 어드레스 그룹 내에서 제1 코어에 맵핑된 논리 블록 어드레스들 지속적으로 증가할 가능성이 높기 때문이다.
- [0050] 또한, 리-맵핑 모듈(215)은 리-맵핑 대상 논리 블록 어드레스 그룹의 논리 블록 어드레스들 각각에 대응하는 물리 블록 어드레스들에 저장된 데이터를 리-맵핑될 코어의 전용 다이로 이동시키기 위한 지시를 각 코어(Core 0 ~ Core i-1)로 제공할 수 있다. 제2 CPU(220)의 복수의 코어들(Core 0 ~ Core i-1)은 각각 리-맵핑 모듈(215)의 지시에 따라 리-맵핑될 코어로 데이터를 이동시킬 수 있다. 이에 따라, 리-맵핑 전에 각 코어에 저장된 데이터를 무효 데이터로 만들 수 있다.
- [0051] 또한, 리-맵핑 모듈(215)은 데이터의 이동이 완료되면, 코어 맵핑 오프셋 테이블(CMOT)에 저장된 리-맵핑 대상 논리 블록 어드레스 그룹에 대응하는 코어 맵핑 오프셋을 재설정할 수 있다. 예를 들어, 전술한 바와 같이, 제1 코어의 P/E 평균 카운트가 제2 코어의 P/E 평균 카운트보다 크면, 제1 코어가 소스 코어가 되고, 제2 코어가 타겟 코어가 될 수 있다. 즉, 리-맵핑 모듈(215)은 제1 코어에 맵핑된 논리 블록 어드레스가 제2 코어에 맵핑되도록 리-맵핑을 수행할 수 있다. 이를 위해, 리-맵핑 모듈(215)은 제1 코어로부터 제2 코어까지의 오프셋을 리-맵핑 대상 논리 블록 어드레스 그룹의 코어 맵핑 오프셋으로 재설정할 수 있다. 일 실시 예에서, 제1 코어가 'Core 0' 이고, 제2 코어가 'Core 3' 인 경우, 오프셋은 '3' 이므로, 도 6의 코어 맵핑 오프셋 테이블(CMOT)에서 리-맵핑 대상 논리 어드레스 그룹(예컨대, LBAG0)에 대응하는 코어 맵핑 오프셋은 '3' 으로 재설정될 수 있다.
- [0052] 도 6을 참조하면, 초기 단계에서 생성된 코어 맵핑 오프셋 테이블(CMOT)에서 각 논리 블록 어드레스 그룹(LBAG0~LBAGn)의 코어 맵핑 오프셋의 디폴트 값은 '0' 으로 설정된다. 이에 따라, 리-맵핑 이전에 호스트로부터

터 수신되는 논리 블록 어드레스는 코어 수로 나눈 나머지 값에 해당하는 코어에 맵핑될 수 있다. 한편, 리-맵핑 이후에 호스트로부터 수신되는 논리 블록 어드레스는 코어 수로 나눈 나머지 값에 재설정된 코어 맵핑 오프셋(예컨대, 3)을 더한 값을 다시 코어 수로 나눈 나머지 값에 해당하는 코어에 맵핑될 수 있다.

[0053] 예를 들어, 제2 CPU(220)의 코어 수가 4개(Core 0 ~ Core 3)이고, 제1 코어는 'Core 0', 제2 코어는 'Core 3' 이고, 제1 코어의 P/E 평균 카운트가 제2 코어의 P/E 평균 카운트보다 큰 것으로 가정하면, 코어 리맵핑 오프셋은 '3' 으로 설정될 수 있다. 이에 따라, 당초 'Core 0' 에 맵핑된 'LBA0, LBA4, LBA8, ...' 은 'Core 3' 으로 리-맵핑되고, 당초 'Core 1' 에 맵핑된 'LBA1, LBA5, LBA9, ...' 는 'Core 0' 으로 리-맵핑되고, 당초 'Core 2' 에 맵핑된 'LBA2, LBA6, LBA10, ...' 은 'Core 1' 로 리-맵핑되고, 당초 'Core 3' 에 맵핑된 'LBA3, LBA7, LBA11, ...' 은 'Core 2' 로 리-맵핑될 수 있다.

[0054] 한편, 제2 코어의 P/E 평균 카운트가 제1 코어의 P/E 평균 카운트보다 큰 것으로 가정하면, 코어 리맵핑 오프셋은 '1' 로 설정될 수 있다. 이에 따라, 당초 'Core 0' 에 맵핑된 'LBA0, LBA4, LBA8, ...' 은 'Core 1' 로 리-맵핑되고, 당초 'Core 1' 에 맵핑된 'LBA1, LBA5, LBA9, ...' 는 'Core 2' 로 리-맵핑되고, 당초 'Core 2' 에 맵핑된 'LBA2, LBA6, LBA10, ...' 은 'Core 3' 으로 리-맵핑되고, 당초 'Core 3' 에 맵핑된 'LBA3, LBA7, LBA11, ...' 은 'Core 0' 으로 리-맵핑될 수 있다.

[0055] 도 7은 본 실시 예에 따른 데이터 저장 장치의 동작 방법을 나타낸 순서도이다. 도 7을 참조하여 본 실시 예에 따른 데이터 저장 장치의 동작 방법을 설명함에 있어서, 도 1 내지 도 6 중 적어도 하나의 도면이 참조될 수 있다.

[0056] S710 단계에서, 컨트롤러(200)의 제1 CPU(210)는 기 설정된 주기마다 휘발성 메모리(300)에 저장된 P/E 평균 카운트 테이블(P/E ACT)을 참조하여 복수의 코어들(Core 0 ~ Core i-1) 각각의 P/E 평균 카운트를 비교할 수 있다.

[0057] S720 단계에서, 제1 CPU(210)는 복수의 코어들(Core 0 ~ Core i-1) 중 P/E 평균 카운트의 차이 값이 기 설정된 임계 값 이상인 제1 코어와 제2 코어가 존재하는지 여부를 판단할 수 있다. 설명의 편의를 위해, 본 실시 예에서는, 제1 코어의 P/E 평균 카운트가 제2 코어의 P/E 평균 카운트보다 큰 것으로 가정한다. 제1 코어와 제2 코어가 존재하지 않으면, 해당 프로세스는 종료될 수 있다. 제1 코어와 제2 코어가 존재하면, 프로세스는 S730 단계로 진행될 수 있다. 본 단계에서, 제1 코어와 제2 코어가 존재하면, 제1 CPU(210)는 리-맵핑 모듈(215)을 구동시켜 S730 내지 S750 단계들을 수행할 수 있다.

[0058] S730 단계에서, 제1 CPU(210)에 의해 구동된 리-맵핑 모듈(215)은 휘발성 메모리(300)에 저장된 코어 액세스 카운트 테이블(CACT)을 참조하여, 제1 코어의 액세스 카운트가 제2 코어의 액세스 카운트보다 큰 제1 조건 및 제1 코어의 액세스 카운트와 제2 코어의 액세스 카운트 간의 차이 값이 가장 큰 제2 조건을 만족하는 논리 블록 어드레스 그룹을 리-맵핑 대상 논리 블록 어드레스 그룹(LBAG)으로 선택할 수 있다.

[0059] S740 단계에서, 리-맵핑 모듈(215)은 리-맵핑 대상 논리 블록 어드레스 그룹에 포함된 논리 블록 어드레스들 각각에 대응하는 물리 블록 어드레스들에 저장된 데이터를 리-맵핑될 코어의 전용 다이(die)들(또는 저장 영역들)로 이동시키기 위한 지시를 각 코어(Core 0 ~ Core i-1)에 제공할 수 있다. 각 코어(Core 0 ~ Core i-1)는 리-맵핑 모듈(215)의 지시에 따라 리-맵핑될 코어의 전용 다이(die)들로 데이터를 이동시킬 수 있다. 데이터의 이동이 완료되면, 프로세스는 S750으로 진행될 수 있다.

[0060] S750 단계에서, 리-맵핑 모듈(215)은 코어 맵핑 오프셋 테이블(CMOT)에서 리-맵핑 대상 논리 블록 어드레스 그룹에 대응하는 코어 맵핑 오프셋을 재설정할 수 있다. 예를 들어, 리-맵핑 모듈(215)은 소스 코어(즉, 제1 코어)로부터 타겟 코어(즉, 제2 코어)까지의 오프셋을 리-맵핑 대상 논리 블록 어드레스 그룹의 코어 맵핑 오프셋으로 재설정할 수 있다.

[0061] 도 8은 본 발명의 실시 예에 따른 솔리드 스테이트 드라이브(SSD)를 포함하는 데이터 처리 시스템을 예시적으로 나타낸 도면이다. 도 8을 참조하면, 데이터 처리 시스템(2000)은 호스트 장치(2100)와 솔리드 스테이트 드라이브(solid state drive)(2200)(이하, SSD라 칭함)를 포함할 수 있다.

[0062] SSD(2200)는 컨트롤러(2210), 버퍼 메모리 장치(2220), 불휘발성 메모리 장치들(2231~223n), 전원 공급기(2240), 신호 커넥터(2250) 및 전원 커넥터(2260)를 포함할 수 있다.

[0063] 컨트롤러(2210)는 SSD(2200)의 제반 동작을 제어할 수 있다.

[0064] 버퍼 메모리 장치(2220)는 불휘발성 메모리 장치들(2231~223n)에 저장될 데이터를 임시 저장할 수 있다. 또한,

버퍼 메모리 장치(2220)는 불휘발성 메모리 장치들(2231~223n)로부터 읽혀진 데이터를 임시 저장할 수 있다. 버퍼 메모리 장치(2220)에 임시 저장된 데이터는 컨트롤러(2210)의 제어에 따라 호스트 장치(2100) 또는 불휘발성 메모리 장치들(2231~223n)로 전송될 수 있다.

- [0065] 불휘발성 메모리 장치들(2231~223n)은 SSD(2200)의 저장 매체로 사용될 수 있다. 불휘발성 메모리 장치들(2231~223n) 각각은 복수의 채널들(CH1~CHn)을 통해 컨트롤러(2210)와 연결될 수 있다. 하나의 채널에는 하나 또는 그 이상의 불휘발성 메모리 장치가 연결될 수 있다. 하나의 채널에 연결되는 불휘발성 메모리 장치들은 동일한 신호 버스 및 데이터 버스에 연결될 수 있다.
- [0066] 전원 공급기(2240)는 전원 커넥터(2260)를 통해 입력된 전원(PWR)을 SSD(2200) 내부에 제공할 수 있다. 전원 공급기(2240)는 보조 전원 공급기(2241)를 포함할 수 있다. 보조 전원 공급기(2241)는 서든 파워 오프(sudden power off)가 발생하는 경우, SSD(2200)가 정상적으로 종료될 수 있도록 전원을 공급할 수 있다. 보조 전원 공급기(2241)는 전원(PWR)을 충전할 수 있는 대용량 캐패시터들(capacitors)을 포함할 수 있다.
- [0067] 컨트롤러(2210)는 신호 커넥터(2250)를 통해서 호스트 장치(2100)와 신호(SGL)를 주고 받을 수 있다. 여기에서, 신호(SGL)는 커맨드, 어드레스, 데이터 등을 포함할 수 있다. 신호 커넥터(2250)는 호스트 장치(2100)와 SSD(2200)의 인터페이스 방식에 따라 다양한 형태의 커넥터로 구성될 수 있다.
- [0068] 도 9는 도 8의 컨트롤러의 구성을 예시적으로 나타낸 도면이다. 도 9를 참조하면, 컨트롤러(2210)는 호스트 인터페이스 유닛(2211), 컨트롤 유닛(2212), 랜덤 액세스 메모리(2213), 에러 정정 코드(ECC) 유닛(2214) 및 메모리 인터페이스 유닛(2215)을 포함할 수 있다.
- [0069] 호스트 인터페이스 유닛(2211)은, 호스트 장치(2100)의 프로토콜에 따라서, 호스트 장치(2100)와 SSD(2200)를 인터페이스할 수 있다. 예를 들면, 호스트 인터페이스 유닛(2211)은, 시큐어 디지털(secure digital), USB(universal serial bus), MMC(multi-media card), eMMC(embedded MMC), PCMCIA(personal computer memory card international association), PATA(parallel advanced technology attachment), SATA(serial advanced technology attachment), SCSI(small computer system interface), SAS(serial attached SCSI), PCI(peripheral component interconnection), PCI-E(PCI Expresss), UFS(universal flash storage) 프로토콜들 중 어느 하나를 통해서 호스트 장치(2100)와 통신할 수 있다. 또한, 호스트 인터페이스 유닛(2211)은 호스트 장치(2100)가 SSD(2200)를 범용 데이터 저장 장치, 예를 들면, 하드 디스크 드라이브(HDD)로 인식하도록 지원하는 디스크 에뮬레이션(disk emulation) 기능을 수행할 수 있다.
- [0070] 컨트롤 유닛(2212)은 호스트 장치(2100)로부터 입력된 신호(SGL)를 분석하고 처리할 수 있다. 컨트롤 유닛(2212)은 SSD(2200)를 구동하기 위한 펌웨어 또는 소프트웨어에 따라서 내부 기능 블록들의 동작을 제어할 수 있다. 랜덤 액세스 메모리(2213)는 이러한 펌웨어 또는 소프트웨어를 구동하기 위한 동작 메모리로서 사용될 수 있다.
- [0071] 에러 정정 코드(ECC) 유닛(2214)은 불휘발성 메모리 장치들(2231~223n)로 전송될 데이터의 패리티 데이터를 생성할 수 있다. 생성된 패리티 데이터는 데이터와 함께 불휘발성 메모리 장치들(2231~223n)에 저장될 수 있다. 에러 정정 코드(ECC) 유닛(2214)은 패리티 데이터에 근거하여 불휘발성 메모리 장치들(2231~223n)로부터 독출된 데이터의 에러를 검출할 수 있다. 만약, 검출된 에러가 정정 범위 내이면, 에러 정정 코드(ECC) 유닛(2214)은 검출된 에러를 정정할 수 있다.
- [0072] 메모리 인터페이스 유닛(2215)은, 컨트롤 유닛(2212)의 제어에 따라서, 불휘발성 메모리 장치들(2231~223n)에 커맨드 및 어드레스와 같은 제어 신호를 제공할 수 있다. 그리고 메모리 인터페이스 유닛(2215)은, 컨트롤 유닛(2212)의 제어에 따라서, 불휘발성 메모리 장치들(2231~223n)과 데이터를 주고받을 수 있다. 예를 들면, 메모리 인터페이스 유닛(2215)은 버퍼 메모리 장치(2220)에 저장된 데이터를 불휘발성 메모리 장치들(2231~223n)로 제공하거나, 불휘발성 메모리 장치들(2231~223n)로부터 읽혀진 데이터를 버퍼 메모리 장치(2220)로 제공할 수 있다.
- [0073] 도 10은 본 발명의 실시 예에 따른 데이터 저장 장치를 포함하는 데이터 처리 시스템을 예시적으로 나타낸 도면이다. 도 10을 참조하면, 데이터 처리 시스템(3000)은 호스트 장치(3100)와 데이터 저장 장치(3200)를 포함할 수 있다.
- [0074] 호스트 장치(3100)는 인쇄 회로 기판(printed circuit board)과 같은 기판(board) 형태로 구성될 수 있다. 비록 도시되지 않았지만, 호스트 장치(3100)는 호스트 장치의 기능을 수행하기 위한 내부 기능 블록들을 포함할

수 있다.

- [0075] 호스트 장치(3100)는 소켓(socket), 슬롯(slot) 또는 커넥터(connector)와 같은 접속 터미널(3110)을 포함할 수 있다. 데이터 저장 장치(3200)는 접속 터미널(3110)에 마운트(mount)될 수 있다.
- [0076] 데이터 저장 장치(3200)는 인쇄 회로 기판과 같은 기판 형태로 구성될 수 있다. 데이터 저장 장치(3200)는 메모리 모듈 또는 메모리 카드로 불릴 수 있다. 데이터 저장 장치(3200)는 컨트롤러(3210), 버퍼 메모리 장치(3220), 불휘발성 메모리 장치(3231~3232), PMIC(power management integrated circuit)(3240) 및 접속 터미널(3250)을 포함할 수 있다.
- [0077] 컨트롤러(3210)는 데이터 저장 장치(3200)의 제반 동작을 제어할 수 있다. 컨트롤러(3210)는 도 9에 도시된 컨트롤러(2210)와 동일하게 구성될 수 있다.
- [0078] 버퍼 메모리 장치(3220)는 불휘발성 메모리 장치들(3231~3232)에 저장될 데이터를 임시 저장할 수 있다. 또한, 버퍼 메모리 장치(3220)는 불휘발성 메모리 장치들(3231~3232)로부터 읽혀진 데이터를 임시 저장할 수 있다. 버퍼 메모리 장치(3220)에 임시 저장된 데이터는 컨트롤러(3210)의 제어에 따라 호스트 장치(3100) 또는 불휘발성 메모리 장치들(3231~3232)로 전송될 수 있다.
- [0079] 불휘발성 메모리 장치들(3231~3232)은 데이터 저장 장치(3200)의 저장 매체로 사용될 수 있다.
- [0080] PMIC(3240)는 접속 터미널(3250)을 통해 입력된 전원을 데이터 저장 장치(3200) 내부에 제공할 수 있다. PMIC(3240)는, 컨트롤러(3210)의 제어에 따라서, 데이터 저장 장치(3200)의 전원을 관리할 수 있다.
- [0081] 접속 터미널(3250)은 호스트 장치의 접속 터미널(3110)에 연결될 수 있다. 접속 터미널(3250)을 통해서, 호스트 장치(3100)와 데이터 저장 장치(3200) 간에 커맨드, 어드레스, 데이터 등과 같은 신호와, 전원이 전달될 수 있다. 접속 터미널(3250)은 호스트 장치(3100)와 데이터 저장 장치(3200)의 인터페이스 방식에 따라 다양한 형태로 구성될 수 있다. 접속 터미널(3250)은 데이터 저장 장치(3200)의 어느 한 번에 배치될 수 있다.
- [0082] 도 11은 본 발명의 실시 예에 따른 데이터 저장 장치를 포함하는 데이터 처리 시스템을 예시적으로 나타낸 도면이다. 도 11을 참조하면, 데이터 처리 시스템(4000)은 호스트 장치(4100)와 데이터 저장 장치(4200)를 포함할 수 있다.
- [0083] 호스트 장치(4100)는 인쇄 회로 기판(printed circuit board)과 같은 기판(board) 형태로 구성될 수 있다. 비록 도시되지 않았지만, 호스트 장치(4100)는 호스트 장치의 기능을 수행하기 위한 내부 기능 블록들을 포함할 수 있다.
- [0084] 데이터 저장 장치(4200)는 표면 실장형 패키지 형태로 구성될 수 있다. 데이터 저장 장치(4200)는 솔더 볼(solder ball)(4250)을 통해서 호스트 장치(4100)에 마운트될 수 있다. 데이터 저장 장치(4200)는 컨트롤러(4210), 버퍼 메모리 장치(4220) 및 불휘발성 메모리 장치(4230)를 포함할 수 있다.
- [0085] 컨트롤러(4210)는 데이터 저장 장치(4200)의 제반 동작을 제어할 수 있다. 컨트롤러(4210)는 도 9에 도시된 컨트롤러(2210)와 동일하게 구성될 수 있다.
- [0086] 버퍼 메모리 장치(4220)는 불휘발성 메모리 장치(4230)에 저장될 데이터를 임시 저장할 수 있다. 또한, 버퍼 메모리 장치(4220)는 불휘발성 메모리 장치들(4230)로부터 읽혀진 데이터를 임시 저장할 수 있다. 버퍼 메모리 장치(4220)에 임시 저장된 데이터는 컨트롤러(4210)의 제어에 따라 호스트 장치(4100) 또는 불휘발성 메모리 장치(4230)로 전송될 수 있다.
- [0087] 불휘발성 메모리 장치(4230)는 데이터 저장 장치(4200)의 저장 매체로 사용될 수 있다.
- [0088] 도 12는 본 발명의 실시 예에 따른 데이터 저장 장치를 포함하는 네트워크 시스템(5000)을 예시적으로 나타낸 도면이다. 도 12는 참조하면, 네트워크 시스템(5000)은 네트워크(5500)를 통해서 연결된 서버 시스템(5300) 및 복수의 클라이언트 시스템들(5410~5430)을 포함할 수 있다.
- [0089] 서버 시스템(5300)은 복수의 클라이언트 시스템들(5410~5430)의 요청에 응답하여 데이터를 서비스할 수 있다. 예를 들면, 서버 시스템(5300)은 복수의 클라이언트 시스템들(5410~5430)로부터 제공된 데이터를 저장할 수 있다. 다른 예로서, 서버 시스템(5300)은 복수의 클라이언트 시스템들(5410~5430)로 데이터를 제공할 수 있다.
- [0090] 서버 시스템(5300)은 호스트 장치(5100) 및 데이터 저장 장치(5200)를 포함할 수 있다. 데이터 저장 장치(5200)는 도 1의 데이터 저장 장치(10), 도 8의 데이터 저장 장치(2200), 도 10의 데이터 저장 장치(3200) 및 도 11

의 데이터 저장 장치(4200)로 구성될 수 있다.

- [0091] 도 13은 본 발명의 실시 예에 따른 데이터 저장 장치에 포함된 불휘발성 메모리 장치를 예시적으로 보여주는 블록도이다. 도 13을 참조하면, 불휘발성 메모리 장치(100)는 메모리 셀 어레이(110), 행 디코더(120), 열 디코더(130), 데이터 읽기/쓰기 블럭(140), 전압 발생기(150) 및 제어 로직(160)을 포함할 수 있다.
- [0092] 메모리 셀 어레이(110)는 워드 라인들(WL1~WLm)과 비트 라인들(BL1~BLn)이 서로 교차된 영역에 배열된 메모리 셀(MC)들을 포함할 수 있다.
- [0093] 행 디코더(120)는 워드 라인들(WL1~WLm)을 통해서 메모리 셀 어레이(110)와 연결될 수 있다. 행 디코더(120)는 제어 로직(160)의 제어에 따라 동작할 수 있다. 행 디코더(120)는 외부 장치(도시되지 않음)로부터 제공된 어드레스를 디코딩할 수 있다. 행 디코더(120)는 디코딩 결과에 근거하여 워드 라인들(WL1~WLm)을 선택하고, 구동할 수 있다. 예시적으로, 행 디코더(120)는 전압 발생기(150)로부터 제공된 워드 라인 전압을 워드 라인들(WL1~WLm)에 제공할 수 있다.
- [0094] 데이터 읽기/쓰기 블럭(140)은 비트 라인들(BL1~BLn)을 통해서 메모리 셀 어레이(110)와 연결될 수 있다. 데이터 읽기/쓰기 블럭(140)은 비트 라인들(BL1~BLn) 각각에 대응하는 읽기/쓰기 회로들(RW1~RWn)을 포함할 수 있다. 데이터 읽기/쓰기 블럭(140)은 제어 로직(160)의 제어에 따라 동작할 수 있다. 데이터 읽기/쓰기 블럭(140)은 동작 모드에 따라서 쓰기 드라이버로서 또는 감지 증폭기로서 동작할 수 있다. 예를 들면, 데이터 읽기/쓰기 블럭(140)은 쓰기 동작 시 외부 장치로부터 제공된 데이터를 메모리 셀 어레이(110)에 저장하는 쓰기 드라이버로서 동작할 수 있다. 다른 예로서, 데이터 읽기/쓰기 블럭(140)은 읽기 동작 시 메모리 셀 어레이(110)로부터 데이터를 독출하는 감지 증폭기로서 동작할 수 있다.
- [0095] 열 디코더(130)는 제어 로직(160)의 제어에 따라 동작할 수 있다. 열 디코더(130)는 외부 장치로부터 제공된 어드레스를 디코딩할 수 있다. 열 디코더(130)는 디코딩 결과에 근거하여 비트 라인들(BL1~BLn) 각각에 대응하는 데이터 읽기/쓰기 블럭(140)의 읽기/쓰기 회로들(RW1~RWn)과 데이터 입출력 라인(또는 데이터 입출력 버퍼)을 연결할 수 있다.
- [0096] 전압 발생기(150)는 불휘발성 메모리 장치(100)의 내부 동작에 사용되는 전압을 생성할 수 있다. 전압 발생기(150)에 의해서 생성된 전압들은 메모리 셀 어레이(110)의 메모리 셀들에 인가될 수 있다. 예를 들면, 프로그램 동작 시 생성된 프로그램 전압은 프로그램 동작이 수행될 메모리 셀들의 워드 라인에 인가될 수 있다. 다른 예로서, 소거 동작 시 생성된 소거 전압은 소거 동작이 수행될 메모리 셀들의 웰-영역에 인가될 수 있다. 다른 예로서, 읽기 동작 시 생성된 읽기 전압은 읽기 동작이 수행될 메모리 셀들의 워드 라인에 인가될 수 있다.
- [0097] 제어 로직(160)은 외부 장치로부터 제공된 제어 신호에 근거하여 불휘발성 메모리 장치(100)의 제반 동작을 제어할 수 있다. 예를 들면, 제어 로직(160)은 불휘발성 메모리 장치(100)의 읽기, 쓰기, 소거 동작과 같은 불휘발성 메모리 장치(100)의 동작을 제어할 수 있다.
- [0098] 본 발명이 속하는 기술분야의 통상의 기술자는 본 발명이 그 기술적 사상이나 필수적 특징을 변경하지 않고서 다른 구체적인 형태로 실시될 수 있으므로, 이상에서 기술한 실시 예들은 모든 면에서 예시적인 것이며 한정적이 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.

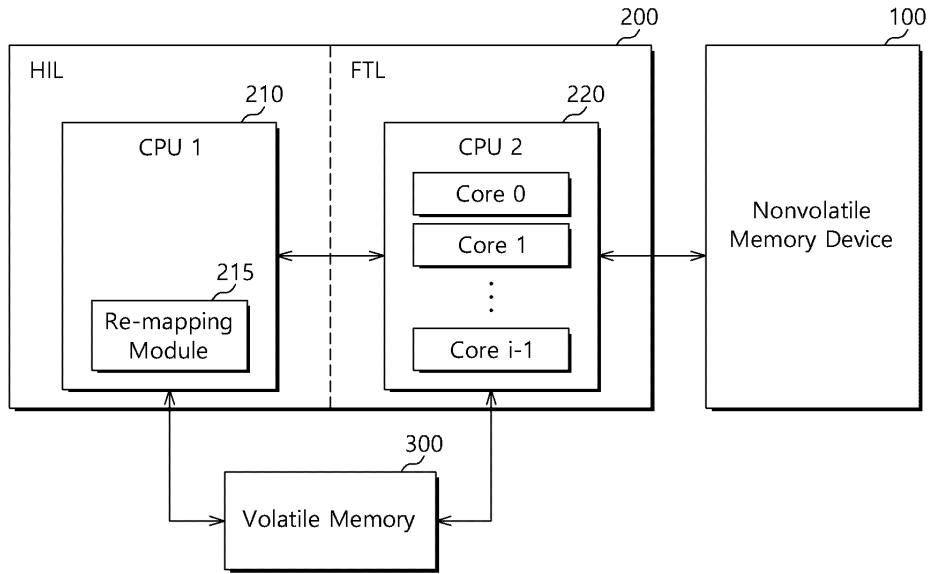
부호의 설명

- [0099] 10: 데이터 저장 장치 100: 불휘발성 메모리 장치
- 200: 컨트롤러 210: 제1 CPU
- 220: 제2 CPU 300: 휘발성 메모리

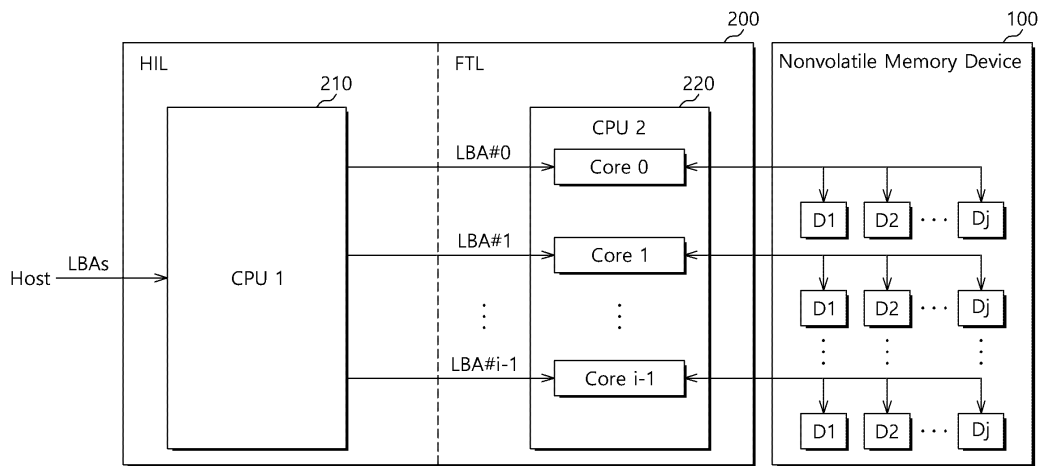
도면

도면1

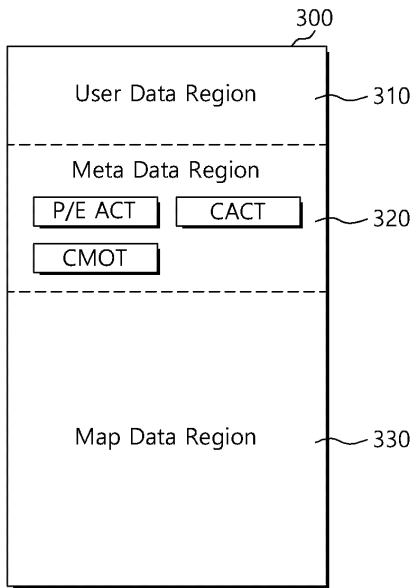
10



도면2



도면3



도면4

P/E ACT (P/E average count table)

Core 0	P/E average Count 0
Core 1	P/E average Count 1
⋮	⋮
Core i-1	P/E average Count i-1

도면5

CACT (core access count table)

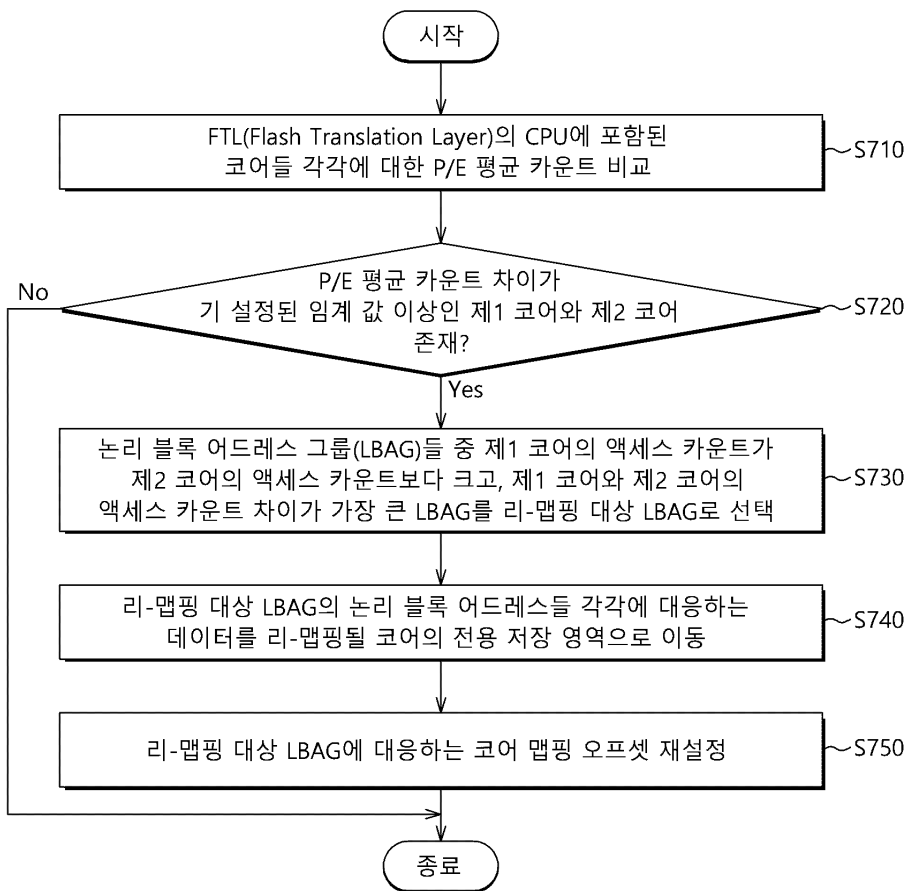
LBAG0	Core 0 access count	LBA#0 of LBAG0
	Core 1 access count	LBA#1 of LBAG0
	⋮	⋮
	Core i-1 access count	LBA#i-1 of LBAG0
⋮	⋮	⋮
LBAGn	Core 0 access count	LBA#0 of LBAGn
	Core 1 access count	LBA#1 of LBAGn
	⋮	⋮
	Core i-1 access count	LBA#i-1 of LBAGn

도면6

CMOT (core mapping offset table)

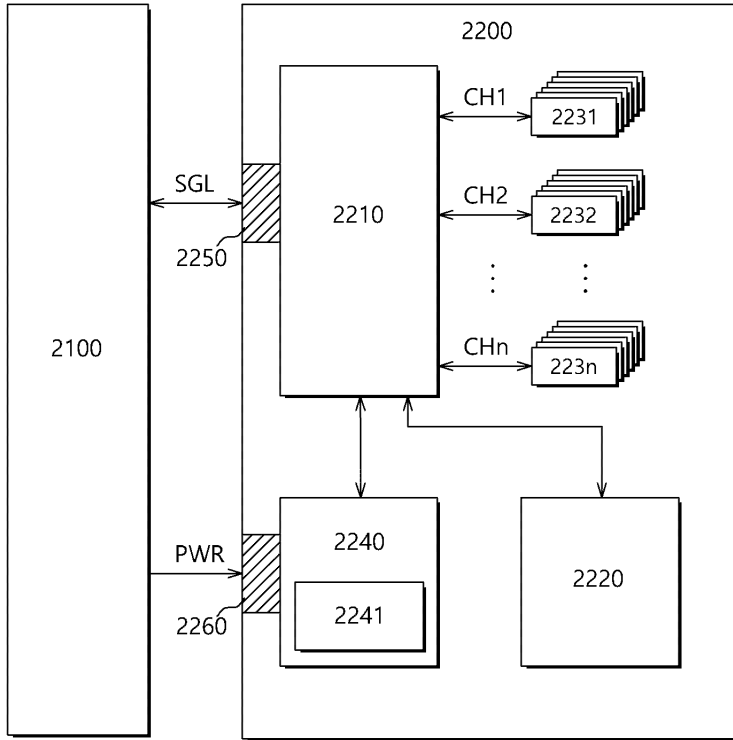
LBAG0	0
LBAG1	0
⋮	⋮
LBAGn	0

도면7



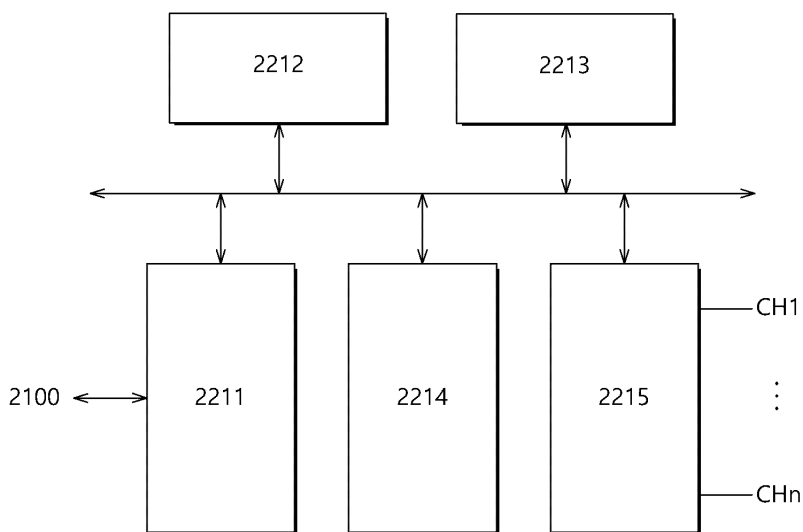
도면8

2000

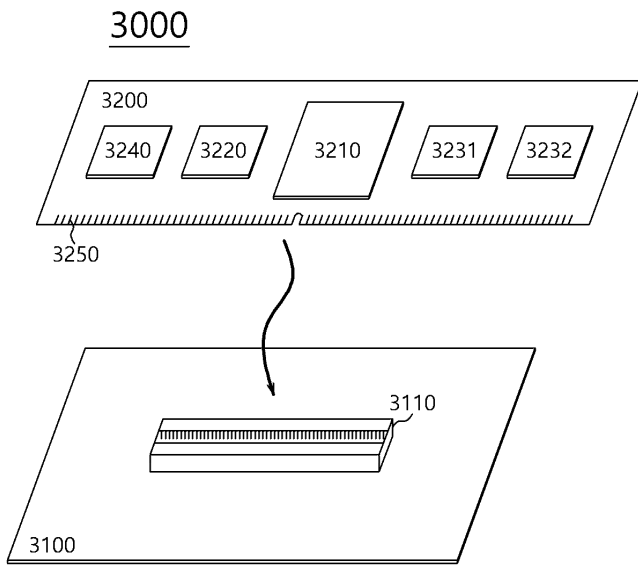


도면9

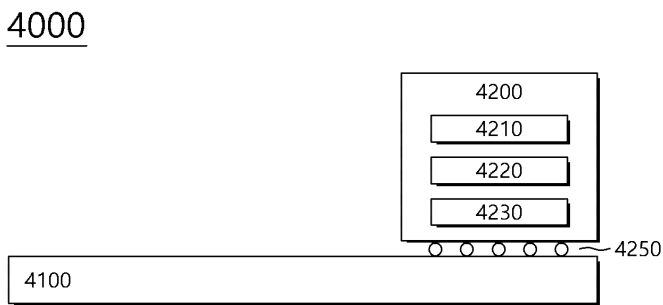
2210



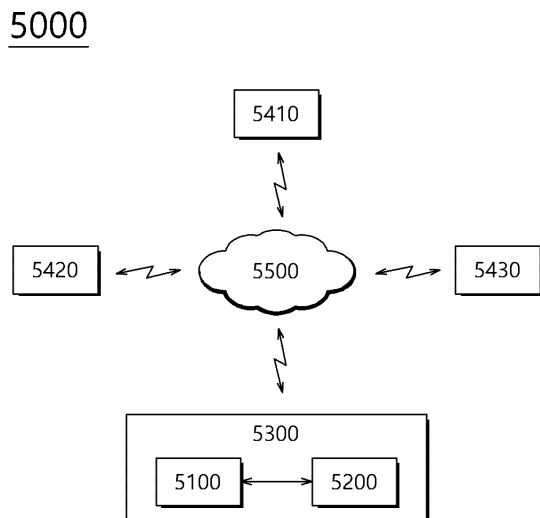
도면10



도면11



도면12



도면13

100

