



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0041233
(43) 공개일자 2021년04월15일

(51) 국제특허분류(Int. Cl.)
G06F 13/16 (2006.01) G06F 3/06 (2006.01)
(52) CPC특허분류
G06F 13/1657 (2013.01)
G06F 3/0604 (2013.01)
(21) 출원번호 10-2019-0123643
(22) 출원일자 2019년10월07일
심사청구일자 없음

(71) 출원인
에스케이하이닉스 주식회사
경기도 이천시 부발읍 경충대로 2091
(72) 발명자
장대훈
경기도 수원시 영통구 효원로 363 매탄 위브 하늘
채 131-2104
임동함
경기도 용인시 기흥구 보정로 87 죽현마을 아이과
크 210동 1001호
(74) 대리인
특허법인(유한)유일하이스트

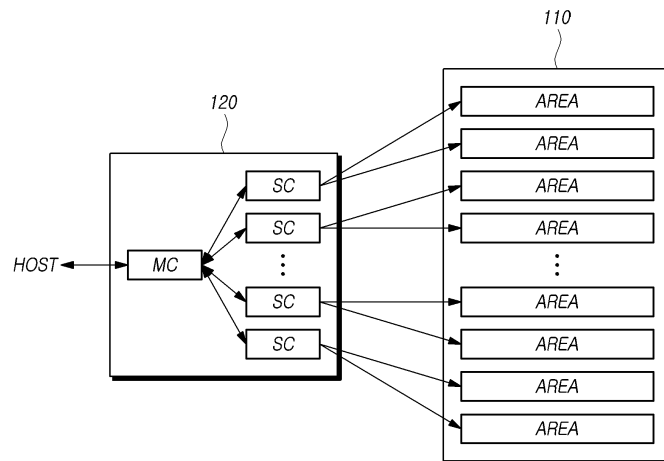
전체 청구항 수 : 총 19 항

(54) 발명의 명칭 메모리 시스템, 메모리 컨트롤러 및 동작 방법

(57) 요약

본 발명의 실시예들은 메모리 시스템, 메모리 컨트롤러 및 동작 방법에 관한 것으로서, 호스트로부터 메모리 장치에 타겟 데이터를 라이트하는 동작을 지시하는 타겟 커맨드를 수신하면, 타겟 데이터를 하나 이상의 데이터 유닛으로 분할하고 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해 제1 데이터 유닛 전체가 메모리 장치에 라이트되거나 또는 제1 데이터 유닛 전체가 메모리 장치에 라이트되지 않도록 제어함으로써, 멀티 코어를 사용하는 메모리 시스템에서 데이터를 특정한 단위로 라이트할 수 있다.

대표도 - 도5



(52) CPC특허분류

G06F 3/0631 (2013.01)

G06F 3/0658 (2013.01)

G06F 3/0659 (2013.01)

G06F 3/0683 (2013.01)

(72) 발명자

차영훈

경기도 수원시 장안구 하롤로30번길 51-12 우성하이힐 401호

최영근

경기도 성남시 분당구 내정로165번길 38 양지마을청구금호아파트 602동 1704호

박정선

서울특별시 강남구 개포로 516, 609동 206호(개포동, 주공아파트)

정천욱

경기도 용인시 수지구달맞이로 19 조이빌B동 502호

명세서

청구범위

청구항 1

복수의 메모리 영역을 포함하는 메모리 장치; 및

상기 메모리 장치를 제어하는 메모리 컨트롤러를 포함하고,

상기 메모리 컨트롤러는 하나의 메인 코어 및 상기 메인 코어와 통신하고 상기 복수의 메모리 영역 중 하나 이상의 메모리 영역에 대한 데이터 라이트 동작을 제어하는 복수의 서브 코어를 포함하고,

상기 메인 코어는 호스트로부터 상기 메모리 장치에 타겟 데이터를 라이트하는 동작을 지시하는 타겟 커맨드를 수신하면, 상기 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 상기 복수의 서브 코어에 할당하고,

상기 메모리 컨트롤러는 상기 복수의 서브 코어가 상기 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해, 상기 제1 데이터 유닛 전체가 상기 메모리 장치에 라이트되거나 또는 상기 제1 데이터 유닛 전체가 상기 메모리 장치에 라이트되지 않도록 제어하는 메모리 시스템.

청구항 2

제1항에 있어서,

상기 복수의 서브 코어 각각은,

서로 다른 메모리 영역에 대한 라이트 동작을 독립적으로 제어하는 메모리 시스템.

청구항 3

제1항에 있어서,

상기 메모리 컨트롤러는,

상기 제1 데이터 유닛을 상기 복수의 메모리 영역에 복수의 서브 데이터 유닛으로 분산하여 라이트하는 메모리 시스템.

청구항 4

제3항에 있어서,

상기 복수의 서브 코어 중 제1 서브 코어는,

상기 복수의 서브 데이터 유닛 중 제1 서브 데이터 유닛을 상기 제1 서브 코어가 제어하는 타겟 메모리 영역에 라이트하는 메모리 시스템.

청구항 5

제4항에 있어서,

상기 제1 서브 코어는,

상기 제1 서브 데이터 유닛을 상기 타겟 메모리 영역에 라이트할 때, 상기 타겟 커맨드의 아이디 및 상기 제1

서브 데이터 유닛의 사이즈 값을 함께 라이트하는 메모리 시스템.

청구항 6

제5항에 있어서,

상기 타겟 커맨드의 아이디 및 상기 제1 서브 데이터 유닛의 사이즈 값을 상기 제1 서브 데이터 유닛이 라이트 되는 복수의 페이지 별로 라이트되는 메모리 시스템.

청구항 7

제6항에 있어서,

상기 타겟 커맨드의 아이디 및 상기 제1 서브 데이터 유닛의 사이즈 값을 상기 복수의 페이지 각각에 대한 메타 정보가 저장되는 스페어 영역에 라이트되는 메모리 시스템.

청구항 8

제4항에 있어서,

상기 제1 서브 코어는,

SPO(Sudden Power Off)에 대한 리커버리 동작 중 상기 제1 서브 데이터 유닛의 일부에 UECC(Uncorrectable ECC)가 검출되면, 상기 타겟 커맨드의 아이디를 상기 메인 코어로 리포트하는 메모리 시스템.

청구항 9

제8항에 있어서,

상기 메인 코어는,

상기 타겟 커맨드의 아이디를 기초로 하여,

상기 복수의 서브 코어 중 상기 제1 서브 코어와 다른 제2 서브 코어가, 상기 복수의 서브 데이터 유닛 중 상기 제2 서브 코어가 제어하는 메모리 영역에 라이트된 제2 서브 데이터 유닛을 언맵(un-map)하도록 지시하는 메모리 시스템.

청구항 10

복수의 메모리 영역을 포함하는 메모리 장치와 통신하기 위한 메모리 인터페이스; 및

상기 메모리 장치를 제어하기 위한 제어 회로를 포함하고,

상기 제어 회로는 하나의 메인 코어 및 상기 메인 코어와 통신하고 상기 복수의 메모리 영역 중 하나 이상의 메모리 영역에 대한 데이터 라이트 동작을 제어하는 복수의 서브 코어를 포함하고,

상기 메인 코어는 호스트로부터 상기 메모리 장치에 타겟 데이터를 라이트하는 동작에 대한 타겟 커맨드를 수신 하면, 상기 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 상기 복수의 서브 코어에 할당하고,

상기 제어 회로는 상기 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해, 상기 제1 데이터 유닛 전체가 상기 메모리 장치에 라이트되거나 또는 상기 제1 데이터 유닛 전체가 상기 메모리 장치에 라이트되지 않도록 제어 하는 메모리 컨트롤러.

청구항 11

제10항에 있어서,
상기 복수의 서브 코어 각각은,
서로 다른 메모리 영역에 대한 라이트 동작을 독립적으로 제어하는 메모리 컨트롤러.

청구항 12

제10항에 있어서,
상기 제어 회로는,
상기 제1 데이터 유닛을 상기 복수의 메모리 영역에 복수의 서브 데이터 유닛으로 분산하여 라이트하는 메모리 컨트롤러.

청구항 13

제12항에 있어서,
상기 복수의 서브 코어 중 제1 서브 코어는,
상기 복수의 서브 데이터 유닛 중 제1 서브 데이터 유닛을 상기 제1 서브 코어가 제어하는 타겟 메모리 영역에 라이트하는 메모리 컨트롤러.

청구항 14

제13항에 있어서,
상기 제1 서브 코어는,
상기 제1 서브 데이터 유닛을 상기 타겟 메모리 영역에 라이트할 때, 상기 타겟 커맨드의 아이디 및 상기 제1 서브 데이터 유닛의 사이즈 값을 함께 라이트하는 메모리 컨트롤러.

청구항 15

제14항에 있어서,
상기 타겟 커맨드의 아이디 및 상기 제1 서브 데이터 유닛의 사이즈 값은 상기 제1 서브 데이터 유닛이 라이트되는 복수의 페이지 별로 라이트되는 메모리 컨트롤러.

청구항 16

제15항에 있어서,
상기 타겟 커맨드의 아이디 및 상기 제1 서브 데이터 유닛의 사이즈 값은 상기 복수의 페이지 각각에 대한 메타 정보가 저장되는 스페어 영역에 라이트되는 메모리 컨트롤러.

청구항 17

제13항에 있어서,
상기 제1 서브 코어는,

SPO(Sudden Power Off)에 대한 리커버리 동작 중 상기 제1 서브 데이터 유닛의 일부에 UECC가 검출되면, 상기 타겟 커맨드의 아이디를 상기 메인 코어로 리포트하는 메모리 컨트롤러.

청구항 18

제17항에 있어서,

상기 메인 코어는,

상기 타겟 커맨드의 아이디를 기초로 하여,

상기 복수의 서브 코어 중 상기 제1 서브 코어와 다른 제2 서브 코어가, 상기 복수의 서브 데이터 유닛 중 상기 제2 서브 코어가 제어하는 메모리 영역에 라이트된 제2 서브 데이터 유닛을 언맵(un-map)하도록 지시하는 메모리 컨트롤러.

청구항 19

하나의 메인 코어 및 상기 메인 코어와 통신하고 복수의 메모리 영역 중 하나 이상의 메모리 영역에 대한 데이터 라이트 동작을 제어하는 복수의 서브 코어를 포함하는 메모리 컨트롤러의 동작 방법에 있어서,

상기 메인 코어가 메모리 장치에 타겟 데이터를 라이트하는 동작에 대한 타겟 커맨드를 호스트로부터 수신하는 단계;

상기 메인 코어가 상기 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 상기 복수의 서브 코어에 할당하는 단계; 및

상기 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해, 상기 제1 데이터 유닛 전체가 상기 메모리 장치에 라이트되거나 또는 상기 제1 데이터 유닛 전체가 상기 메모리 장치에 라이트되지 않도록 제어하는 단계를 포함하는 메모리 컨트롤러의 동작 방법.

발명의 설명

기술 분야

[0001] 본 발명의 실시예들은 메모리 시스템, 메모리 컨트롤러 및 동작 방법에 관한 것이다.

배경 기술

[0002] 저장 장치에 해당하는 메모리 시스템은 컴퓨터와, 스마트 폰, 태블릿 등의 모바일 단말, 또는 각종 전자 기기와 같은 호스트(host)의 요청을 기초로 데이터를 저장하는 장치이다. 메모리 시스템은 하드 디스크 드라이브(HDD: Hard Disk Drive)와 같이 자기 디스크에 데이터를 저장하는 장치뿐 아니라, 솔리드 스테이트 드라이브(SSD: Solid State Drive), UFS(Universal Flash Storage) 장치, eMMC(embedded MMC) 장치 등과 같이 비휘발성 메모리에 데이터를 저장하는 장치 등을 포함할 수 있다.

[0003] 메모리 시스템은 메모리 장치를 제어하기 위한 메모리 컨트롤러를 더 포함할 수 있으며, 이러한 메모리 컨트롤러는 호스트로부터 커맨드(Command)를 입력 받아, 입력 받은 커맨드에 기초하여 메모리 시스템에 포함된 휘발성 메모리 또는 비휘발성 메모리에 데이터를 리드(Read), 라이트(Write), 또는 소거(Erase) 하기 위한 동작들을 실행하거나 제어할 수 있다. 그리고 메모리 컨트롤러는 이러한 동작들을 실행하거나 제어하기 위한 논리 연산을 수행하기 위한 펌웨어를 구동할 수 있다.

[0004] 한편, 메모리 컨트롤러는 진술한 논리 연산을 수행하기 위한 펌웨어를 내부의 프로세서를 통해 구동한다. 이때, 프로세서는 병렬적으로 동작하는 복수의 코어를 포함하는 멀티 코어 프로세서일 수 있다.

발명의 내용

해결하려는 과제

[0005] 본 발명의 실시예들은, 멀티 코어를 사용하는 메모리 시스템에서 데이터를 특정한 단위로 라이트하는 메모리 시스템, 메모리 컨트롤러 및 동작 방법을 제공할 수 있다.

[0006] 또한, 본 발명의 실시예들은, 멀티 코어를 사용하는 메모리 시스템에서 SPO가 발생한 경우에 데이터가 원자적으로 라이트되는 것을 보장할 수 있는 메모리 시스템, 메모리 컨트롤러 및 동작 방법을 제공할 수 있다.

과제의 해결 수단

[0007] 일 측면에서, 본 발명의 실시예들은, 복수의 메모리 영역을 포함하는 메모리 장치 및 메모리 장치를 제어하는 메모리 컨트롤러를 포함하는 메모리 시스템을 제공할 수 있다.

[0008] 메모리 컨트롤러는 하나의 메인 코어를 포함할 수 있다.

[0009] 메모리 컨트롤러는 메인 코어와 통신하고 복수의 메모리 영역 중 하나 이상의 메모리 영역에 대한 데이터 라이트 동작을 제어하는 복수의 서브 코어를 포함할 수 있다.

[0010] 메인 코어는 호스트로부터 메모리 장치에 타겟 데이터를 라이트하는 동작에 대한 타겟 커맨드를 수신할 수 있다.

[0011] 메인 코어는 타겟 커맨드를 수신하면, 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 복수의 서브 코어에 할당할 수 있다.

[0012] 그리고 메모리 컨트롤러는 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해, 제1 데이터 유닛 전체가 메모리 장치에 라이트되거나 또는 제1 데이터 유닛 전체가 메모리 장치에 라이트되지 않도록 제어할 수 있다.

[0013] 메모리 컨트롤러는 제1 데이터 유닛을 복수의 메모리 영역에 복수의 서브 데이터 유닛으로 분산하여 라이트할 수 있다.

[0014] 복수의 서브 코어 중 제1 서브 코어는 복수의 서브 데이터 유닛 중 제1 서브 데이터 유닛을 제1 서브 코어가 제어하는 타겟 메모리 영역에 라이트할 수 있다.

[0015] 제1 서브 코어는 제1 서브 데이터 유닛을 타겟 메모리 영역에 라이트할 때, 타겟 커맨드의 아이디 및 제1 서브 데이터 유닛의 사이즈 값을 함께 라이트할 수 있다.

[0016] 제1 서브 코어는 SPO에 대한 리커버리 동작 중 제1 서브 데이터 유닛의 일부에 UECC가 검출되면, 타겟 커맨드의 아이디를 메인 코어로 리포트할 수 있다.

[0017] 메인 코어는 제1 서브 코어로부터 리포트된 타겟 커맨드의 아이디를 기초로 하여, 복수의 서브 코어 중 제1 서브 코어와 다른 제2 서브 코어가, 복수의 서브 데이터 유닛 중 제2 서브 코어가 제어하는 메모리 영역에 라이트된 제2 서브 데이터 유닛을 언맵(un-map)하도록 지시할 수 있다.

[0018] 다른 측면에서, 본 발명의 실시예들은, 복수의 메모리 영역을 포함하는 메모리 장치와 통신하기 위한 메모리 인터페이스 및 메모리 장치를 제어하기 위한 제어 회로를 포함하는 메모리 컨트롤러를 제공할 수 있다.

[0019] 제어 회로는 하나의 메인 코어를 포함할 수 있다.

[0020] 제어 회로는 메인 코어와 통신하고 복수의 메모리 영역 중 하나 이상의 메모리 영역에 대한 데이터 라이트 동작을 제어하는 복수의 서브 코어를 포함할 수 있다.

[0021] 메인 코어는 호스트로부터 메모리 장치에 타겟 데이터를 라이트하는 동작에 대한 타겟 커맨드를 수신할 수 있다.

[0022] 메인 코어는 타겟 커맨드를 수신하면, 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 복수의 서브 코어에 할당할 수 있다.

[0023] 그리고 제어 회로는 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해, 제1 데이터 유닛 전체가 메모리 장치에 라이트되거나 또는 제1 데이터 유닛 전체가 메모리 장치에 라이트되지 않도록 제어할 수 있다.

[0024] 제어 회로는 제1 데이터 유닛을 복수의 메모리 영역에 복수의 서브 데이터 유닛으로 분산하여 라이트할 수 있다.

[0025] 복수의 서브 코어 중 제1 서브 코어는 복수의 서브 데이터 유닛 중 제1 서브 데이터 유닛을 제1 서브 코어가 제

어하는 타겟 메모리 영역에 라이트할 수 있다.

- [0026] 제1 서브 코어는 제1 서브 데이터 유닛을 타겟 메모리 영역에 라이트할 때, 타겟 커맨드의 아이디 및 제1 서브 데이터 유닛의 사이즈 값을 함께 라이트할 수 있다.
- [0027] 제1 서브 코어는 SPO에 대한 리커버리 동작 중 제1 서브 데이터 유닛의 일부에 UECC가 검출되면, 타겟 커맨드의 아이디를 메인 코어로 리포트할 수 있다.
- [0028] 메인 코어는 제1 서브 코어로부터 리포트된 타겟 커맨드의 아이디를 기초로 하여, 복수의 서브 코어 중 제1 서브 코어와 다른 제2 서브 코어가, 복수의 서브 데이터 유닛 중 제2 서브 코어가 제어하는 메모리 영역에 라이트된 제2 서브 데이터 유닛을 언맵(un-map)하도록 지시할 수 있다.
- [0029] 또 다른 측면에서, 본 발명의 실시예들은, 하나의 메인 코어 및 복수의 서브 코어를 포함하는 메모리 컨트롤러의 동작 방법을 제공할 수 있다.
- [0030] 이때, 복수의 서브 코어는 메인 코어와 통신하고 복수의 메모리 영역 중 일부에 대한 데이터 라이트 동작을 제어할 수 있다.
- [0031] 메모리 컨트롤러의 동작 방법은 메인 코어가 타겟 커맨드를 호스트로부터 수신하는 단계를 포함할 수 있다. 타겟 커맨드는 메모리 장치에 타겟 데이터를 라이트하는 동작에 대한 커맨드이다.
- [0032] 메모리 컨트롤러의 동작 방법은 메인 코어가 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 복수의 서브 코어에 할당하는 단계를 포함할 수 있다.
- [0033] 메모리 컨트롤러의 동작 방법은 하나 이상의 데이터 유닛 중 제1 데이터 유닛에 대해, 제1 데이터 유닛 전체가 메모리 장치에 라이트되거나 제1 데이터 유닛 전체가 메모리 장치에 라이트되지 않도록 제어하는 단계를 포함할 수 있다.

발명의 효과

- [0034] 본 발명의 실시예들에 의하면, 멀티 코어를 사용하는 메모리 시스템에서 데이터를 특정한 단위로 라이트할 수 있다.
- [0035] 또한 본 발명의 실시예들에 의하면, 멀티 코어를 사용하는 메모리 시스템에서 SPO가 발생한 경우에 데이터가 원자적으로 라이트되는 것을 보장할 수 있다.

도면의 간단한 설명

- [0036] 도 1은 본 발명의 실시예들에 따른 메모리 시스템의 개략적인 구성도이다.
- 도 2는 본 발명의 실시예들에 따른 메모리 장치를 개략적으로 나타낸 블록도다.
- 도 3은 본 발명의 실시예들에 따른 메모리 장치의 각 메모리 블록을 개략적으로 나타낸 도면이다.
- 도 4는 본 발명의 실시예들에 따른 메모리 장치의 워드 라인 및 비트 라인의 구조를 나타낸 도면이다.
- 도 5는 본 발명의 실시예들에 따른 메모리 시스템에서 메인 코어 및 서브 코어의 구조를 나타낸 구성도이다.
- 도 6은 본 발명의 실시예들에 따른 메모리 시스템에서 데이터 유닛을 라이트하는 동작을 나타낸 도면이다.
- 도 7은 본 발명의 실시예들에 따른 메모리 시스템에서 데이터 유닛을 메모리 장치에 서브 데이터 유닛으로 분산하여 라이트하는 동작을 나타낸 도면이다.
- 도 8은 본 발명의 실시예들에 따른 메모리 시스템에서 서브 데이터 유닛을 메모리 장치에 라이트하는 동작의 일 예를 나타낸 도면이다.
- 도 9는 본 발명의 실시예들에 따른 메모리 시스템에서 서브 데이터 유닛을 메모리 장치 내부의 메모리 영역에 라이트하는 동작의 일 예를 나타낸 도면이다.
- 도 10은 본 발명의 실시예들에 따른 메모리 시스템에서 서브 데이터 유닛을 메모리 장치 내부의 메모리 영역에 라이트하는 동작의 다른 예를 나타낸 도면이다.
- 도 11은 본 발명의 실시예들에 따른 메모리 시스템에서 SPOR 동작을 나타낸 도면이다.

도 12 내지 도 14는 본 발명의 실시예들에 따른 메모리 시스템에서 SPOR 동작 시 UECC가 검출되는 경우의 동작을 나타낸 도면이다.

도 15는 본 발명의 실시예들에 따른 메모리 시스템에서 SPOR 동작을 나타낸 흐름도이다.

도 16은 본 발명의 실시예들에 따른 메모리 컨트롤러의 동작 방법을 나타낸 흐름도이다.

도 17은 본 발명의 실시예들에 따른 컴퓨팅 시스템의 구성도이다.

발명을 실시하기 위한 구체적인 내용

- [0037] 이하, 본 발명의 일부 실시예들을 예시적인 도면을 참조하여 상세하게 설명한다. 각 도면의 구성 요소들에 참조 부호를 부가함에 있어서, 동일한 구성 요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 부호를 가질 수 있다. 또한, 본 발명을 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략할 수 있다. 본 명세서 상에서 언급된 "포함한다", "갖는다", "이루어진다" 등이 사용되는 경우 "~만"이 사용되지 않는 이상 다른 부분이 추가될 수 있다. 구성 요소를 단수로 표현한 경우에 특별한 명시적인 기재 사항이 없는 한 복수를 포함하는 경우를 포함할 수 있다.
- [0038] 또한, 본 발명의 구성 요소를 설명하는 데 있어서, 제1, 제2, A, B, (a), (b) 등의 용어를 사용할 수 있다. 이러한 용어는 그 구성 요소를 다른 구성 요소와 구별하기 위한 것일 뿐, 그 용어에 의해 해당 구성 요소의 본질, 차례, 순서 또는 개수 등이 한정되지 않는다.
- [0039] 구성 요소들의 위치 관계에 대한 설명에 있어서, 둘 이상의 구성 요소가 "연결", "결합" 또는 "접속" 등이 된다고 기재된 경우, 둘 이상의 구성 요소가 직접적으로 "연결", "결합" 또는 "접속" 될 수 있지만, 둘 이상의 구성 요소와 다른 구성 요소가 더 "개재"되어 "연결", "결합" 또는 "접속"될 수도 있다고 이해되어야 할 것이다. 여기서, 다른 구성 요소는 서로 "연결", "결합" 또는 "접속" 되는 둘 이상의 구성 요소 중 하나 이상에 포함될 수도 있다.
- [0040] 구성 요소들이나, 동작 방법이나 제작 방법 등과 관련한 시간적 흐름 관계에 대한 설명에 있어서, 예를 들어, "~후에", "~에 이어서", "~다음에", "~전에" 등으로 시간적 선후 관계 또는 흐름적 선후 관계가 설명되는 경우, "바로" 또는 "직접"이 사용되지 않는 이상 연속적이지 않은 경우도 포함할 수 있다.
- [0041] 한편, 구성 요소에 대한 수치 또는 그 대응 정보(예: 레벨 등)가 언급된 경우, 별도의 명시적 기재가 없더라도, 수치 또는 그 대응 정보는 각종 요인(예: 공정상의 요인, 내부 또는 외부 충격, 노이즈 등)에 의해 발생할 수 있는 오차 범위를 포함하는 것으로 해석될 수 있다.
- [0042] 이하에서는, 본 발명의 실시예들을 첨부된 도면을 참조하여 상세히 설명한다.
- [0043] 도 1은 본 발명의 실시예들에 따른 메모리 시스템(100)의 개략적인 구성도이다.
- [0044] 도 1을 참조하면, 본 발명의 실시예들에 따른 메모리 시스템(100)은 데이터를 저장하는 메모리 장치(110)와, 메모리 장치(110)를 제어하는 메모리 컨트롤러(120) 등을 포함할 수 있다.
- [0045] 메모리 장치(110)는 다수의 메모리 블록(Memory Block)을 포함하며, 메모리 컨트롤러(120)의 제어에 응답하여 동작한다. 여기서, 메모리 장치(110)의 동작은 일 예로, 읽기 동작(Read Operation), 프로그램 동작(Program Operation; "Write Operation" 이라고도 함) 및 소거 동작(Erasure Operation) 등을 포함할 수 있다.
- [0046] 메모리 장치(110)는 데이터를 저장하는 복수의 메모리 셀(Memory Cell; 간단히 줄여서 "셀" 이라고도 함)을 포함하는 메모리 셀 어레이(Memory Cell Array)를 포함할 수 있다. 이러한 메모리 셀 어레이는 메모리 블록 내에 존재할 수 있다.
- [0047] 예를 들어, 메모리 장치(110)는 DDR SDRAM(Double Data Rate Synchronous Dynamic Random Access Memory), LPDDR4(Low Power Double Data Rate4) SDRAM, GDDR(Graphics Double Data Rate) SDRAM, LPDDR(Low Power DDR), RDRAM(Rambus Dynamic Random Access Memory), 낸드 플래시 메모리(NAND Flash Memory), 3차원 낸드 플래시 메모리(3D NAND Flash Memory), 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), 또는 스핀주입 자화반전 메모리(Spin Transfer Torque Random Access Memory: STT-RAM) 등으로 다양한 타입으로 구현될 수 있다.

- [0048] 한편, 메모리 장치(110)는 3차원 어레이 구조(three-Dimensional Array structure)로 구현될 수 있다. 본 발명의 실시예들은 전하 저장층이 전도성 부유 게이트(Floating Gate)로 구성된 플래시 메모리 장치는 물론, 전하 저장층이 절연막으로 구성된 차지 트랩형 플래시(Charge Trap Flash; CTF)에도 적용될 수 있다.
- [0049] 메모리 장치(110)는 메모리 컨트롤러(120)로부터 커맨드 및 어드레스 등을 수신하고, 메모리 셀 어레이 중 어드레스에 의해 선택된 영역을 액세스하도록 구성된다. 즉, 메모리 장치(110)는 어드레스에 의해 선택된 영역에 대해 커맨드에 해당하는 동작을 수행할 수 있다.
- [0050] 예를 들면, 메모리 장치(110)는 프로그램 동작, 읽기 동작 및 소거 동작 등을 수행할 수 있다. 이와 관련하여, 프로그램 동작 시, 메모리 장치(110)는 어드레스에 의해 선택된 영역에 데이터를 프로그램 할 것이다. 읽기 동작 시, 메모리 장치(110)는 어드레스에 의해 선택된 영역으로부터 데이터를 읽을 것이다. 소거 동작 시, 메모리 장치(110)는 어드레스에 의해 선택된 영역에 저장된 데이터를 소거할 것이다.
- [0051] 메모리 컨트롤러(120)는 메모리 장치(110)에 대한 쓰기(프로그램), 읽기, 소거 및 백그라운드(background) 동작을 제어할 수 있다. 여기서, 백그라운드 동작은 일 예로 가비지 컬렉션(GC, Garbage Collection), 웨어 레벨링(WL, Wear Leveling), 또는 배드 블록 관리(BBM, Bad Block Management) 동작 등 중 하나 이상을 포함할 수 있다.
- [0052] 메모리 컨트롤러(120)는 호스트(HOST)의 요청에 따라 메모리 장치(110)의 동작을 제어할 수 있다. 이와 다르게, 메모리 컨트롤러(120)는 호스트(HOST)의 요청과 무관하게 메모리 장치(110)의 동작을 제어할 수도 있다.
- [0053] 한편, 메모리 컨트롤러(120)와 호스트(HOST)는 서로 분리된 장치일 수도 있다. 경우에 따라서, 메모리 컨트롤러(120)와 호스트(HOST)는 하나의 장치로 통합되어 구현될 수도 있다. 아래에서는, 설명의 편의를 위하여, 메모리 컨트롤러(120)와 호스트(HOST)가 서로 분리된 장치인 것을 예로 들어 설명한다.
- [0054] 도 1을 참조하면, 메모리 컨트롤러(120)는 메모리 인터페이스(122) 및 제어 회로(123) 등을 포함할 수 있으며, 호스트 인터페이스(121) 등을 더 포함할 수 있다.
- [0055] 호스트 인터페이스(121)는 호스트(HOST)와의 통신을 위한 인터페이스를 제공한다.
- [0056] 제어 회로(123)는 호스트(HOST)로부터 커맨드를 수신할 때, 호스트 인터페이스(121)를 통해서 커맨드를 수신하여, 수신된 커맨드를 처리하는 동작을 수행할 수 있다.
- [0057] 메모리 인터페이스(122)는, 메모리 장치(110)와 연결되어 메모리 장치(110)와의 통신을 위한 인터페이스를 제공한다. 즉, 메모리 인터페이스(122)는 제어 회로(123)의 제어에 응답하여 메모리 장치(110)와 메모리 컨트롤러(120)를 인터페이스를 제공하도록 구성될 수 있다.
- [0058] 제어 회로(123)는 메모리 컨트롤러(120)의 전반적인 제어 동작을 수행하여 메모리 장치(110)의 동작을 제어한다. 이를 위해, 일 예로, 제어 회로(123)는 프로세서(124), 워킹 메모리(125) 등 중 하나 이상을 포함할 수 있으며, 경우에 따라서, 에러 검출 및 정정 회로(ECC Circuit, 126) 등을 더 포함할 수 있다.
- [0059] 프로세서(124)는 메모리 컨트롤러(120)의 제반 동작을 제어하고, 논리 연산을 수행할 수 있다. 프로세서(124)는 호스트 인터페이스(121)를 통해 호스트(HOST)와 통신하고, 메모리 인터페이스(122)를 통해 메모리 장치(110)와 통신할 수 있다.
- [0060] 프로세서(124)는 플래시 변환 레이어(FTL: Flash Translation Layer)의 기능을 수행할 수 있다. 프로세서(124)는 플래시 변환 계층(FTL)을 통해 호스트가 제공한 논리 블록 어드레스(LBA, logical block address)를 물리 블록 어드레스(PBA, physical block address)로 변환할 수 있다. 플래시 변환 계층(FTL)은 맵핑 테이블을 이용하여 논리 블록 어드레스(LBA)를 입력 받아, 물리 블록 어드레스(PBA)로 변환시킬 수 있다.
- [0061] 플래시 변환 계층의 주소 맵핑 방법에는 맵핑 단위에 따라 여러 가지가 있다. 대표적인 어드레스 맵핑 방법에는 페이지 맵핑 방법(Page mapping method), 블록 맵핑 방법(Block mapping method), 그리고 혼합 맵핑 방법(Hybrid mapping method)이 있다.
- [0062] 프로세서(124)는 호스트(HOST)로부터 수신된 데이터를 랜더마이징하도록 구성된다. 예를 들면, 프로세서(124)는 랜더마이징 시드(seed)를 이용하여 호스트(HOST)로부터 수신된 데이터를 랜더마이징할 것이다. 랜더마이징된 데이터는 저장될 데이터로서 메모리 장치(110)에 제공되어 메모리 셀 어레이에 프로그램 된다.
- [0063] 프로세서(124)는 읽기 동작 시 메모리 장치(110)로부터 수신된 데이터를 디랜더마이징하도록 구성된다. 예를 들

면, 프로세서(124)는 디랜더마이징 시드를 이용하여 메모리 장치(110)로부터 수신된 데이터를 디랜더마이징할 것이다. 디랜더마이징된 데이터는 호스트(HOST)로 출력될 것이다.

- [0064] 프로세서(124)는 펌웨어(FirmWare)를 실행하여 메모리 컨트롤러(120)의 동작을 제어할 수 있다. 다시 말해, 프로세서(124)는, 메모리 컨트롤러(120)의 제반 동작을 제어하고, 논리 연산을 수행하기 위하여, 부팅 시 워킹 메모리(125)에 로딩 된 펌웨어를 실행(구동)할 수 있다.
- [0065] 펌웨어(FirmWare)는 메모리 시스템(100) 내에서 실행되는 프로그램으로서, 다양한 기능적 레이어들을 포함할 수 있다.
- [0066] 예를 들어, 펌웨어는, 호스트(HOST)에서 메모리 시스템(100)에 요구하는 논리 주소(Logical Address)와 메모리 장치(110)의 물리주소(Physical Address) 간의 변환 기능을 하는 플래시 변환 레이어(FTL: Flash Translation Layer)와, 호스트(HOST)에서 저장 장치인 메모리 시스템(100)에 요구하는 커맨드를 해석하여 플래시 변환 레이어(FTL)에 전달하는 역할을 하는 호스트 인터페이스 레이어(HIL: Host Interface Layer)와, 플래시 변환 레이어(FTL)에서 지시하는 커맨드를 메모리 장치(110)로 전달하는 플래시 인터페이스 레이어(FIL: Flash Interface Layer) 등 중 하나 이상을 포함할 수 있다.
- [0067] 이러한 펌웨어는, 일 예로, 메모리 장치(110)에 저장되어 있다가 워킹 메모리(125)에 로딩 될 수 있다.
- [0068] 워킹 메모리(125)는 메모리 컨트롤러(120)를 구동하기 위해 필요한 펌웨어, 프로그램 코드, 커맨드 또는 데이터들을 저장할 수 있다. 이러한 워킹 메모리(125)는, 일 예로, 휘발성 메모리로서, SRAM (Static RAM), DRAM (Dynamic RAM) 및 SDRAM(Synchronous DRAM) 등 중 하나 이상을 포함할 수 있다.
- [0069] 에러 검출 및 정정 회로(126)는 에러 정정 코드(Error Correction Code)를 이용하여 확인 대상 데이터의 에러 비트를 검출하고, 검출된 에러 비트를 정정하도록 구성될 수 있다. 여기서, 확인 대상 데이터는, 일 예로, 워킹 메모리(125)에 저장된 데이터이거나, 메모리 장치(110)로부터 읽어온 데이터 등일 수 있다.
- [0070] 에러 검출 및 정정 회로(126)는 에러 정정 코드로 데이터를 디코딩하도록 구현될 수 있다. 에러 검출 및 정정 회로(126)는 다양한 코드 디코더로 구현될 수 있다. 예를 들어, 비체계적 코드 디코딩을 수행하는 디코더 또는 체계적 코드 디코딩을 수행하는 디코더가 이용될 수 있다.
- [0071] 예를 들면, 에러 검출 및 정정 회로(126)는 읽기 데이터들 각각에 대해 섹터(Sector) 단위로 에러 비트를 검출할 수 있다. 즉, 각각의 읽기 데이터는 복수의 섹터(Sector)로 구성될 수 있다. 섹터(Sector)는 플래시 메모리의 읽기 단위인 페이지(Page)보다 더 작은 데이터 단위를 의미할 수 있다. 각각의 읽기 데이터를 구성하는 섹터들은 어드레스를 매개로 서로 대응될 수 있다.
- [0072] 에러 검출 및 정정 회로(126)는 비트 에러율(Bit Error Rate, BER)을 산출하고, 섹터 단위로 정정 가능 여부를 판단할 수 있다. 에러 검출 및 정정 회로(126)는 예를 들어, 비트 에러율(BER)이 기준값(reference value)보다 높은 경우 해당 섹터를 정정 불가능(Uncorrectable or Fail)으로 판단할 것이다. 반면에, 비트 에러율(BER)이 기준값보다 낮은 경우 해당 섹터를 정정 가능(Correctable or Pass)으로 판단할 것이다.
- [0073] 에러 검출 및 정정 회로(126)는 모든 읽기 데이터들에 대해 순차적으로 에러 검출 및 정정 동작을 수행할 수 있다. 에러 검출 및 정정 회로(126)는 읽기 데이터에 포함된 섹터가 정정 가능한 경우 다음 읽기 데이터에 대해서는 해당 섹터에 대한 에러 검출 및 정정 동작을 생략할 수 있다. 이렇게 모든 읽기 데이터들에 대한 에러 검출 및 정정 동작이 종료되면, 에러 검출 및 정정 회로(126)는 마지막까지 정정 불가능으로 판단된 섹터를 검출할 수 있다. 정정 불가능한 것으로 판단된 섹터는 하나 또는 그 이상일 수 있다. 에러 검출 및 정정 회로(126)는 정정 불가능으로 판단된 섹터에 대한 정보(ex. 어드레스 정보)를 프로세서(124)로 전달할 수 있다.
- [0074] 버스(127)는 메모리 컨트롤러(120)의 구성 요소들(121, 122, 124, 125, 126) 사이의 채널(Channel)을 제공하도록 구성될 수 있다. 이러한 버스(127)는, 일 예로, 각종 제어 신호, 커맨드 등을 전달하기 위한 제어 버스와, 각종 데이터를 전달하기 위한 데이터 버스 등을 포함할 수 있다.
- [0075] 메모리 컨트롤러(120)의 전술한 구성 요소들(121, 122, 124, 125, 126)은 예시일 뿐이다. 메모리 컨트롤러(120)의 전술한 구성 요소들(121, 122, 124, 125, 126) 중 일부의 구성 요소는 삭제되거나, 메모리 컨트롤러(120)의 전술한 구성 요소들 (121, 122, 124, 125, 126) 중 몇몇 구성 요소들이 하나로 통합될 수 있다. 경우에 따라, 메모리 컨트롤러(120)의 전술한 구성 요소들 이외에 하나 이상의 다른 구성 요소가 추가될 수도 있다.
- [0076] 아래에서는, 도 2를 참조하여 메모리 장치(110)에 대하여 더욱 상세하게 설명한다.

- [0077] 도 2는 본 발명의 실시예들에 따른 메모리 장치(110)를 개략적으로 나타낸 블록도다.
- [0078] 도 2를 참조하면, 본 발명의 실시예들에 따른 메모리 장치(110)는, 메모리 셀 어레이(Memory Cell Array, 210), 어드레스 디코더(Address Decoder, 220), 읽기 및 쓰기 회로(Read and Write Circuit, 230), 제어 로직(Control Logic, 240) 및 전압 생성 회로(Voltage Generation Circuit, 250) 등을 포함할 수 있다.
- [0079] 메모리 셀 어레이(210)는 다수의 메모리 블록(BLK1~BLKz, z는 2 이상의 자연수)을 포함할 수 있다.
- [0080] 다수의 메모리 블록(BLK1~BLKz)에는, 다수의 워드 라인(WL)과 다수의 비트 라인(BL)이 배치되며, 다수의 메모리 셀(MC)이 배열될 수 있다.
- [0081] 다수의 메모리 블록(BLK1~BLKz)은 다수의 워드 라인(WL)을 통해 어드레스 디코더(220)와 연결될 수 있다. 다수의 메모리 블록(BLK1~BLKz)은 다수의 비트 라인(BL)을 통해 읽기 및 쓰기 회로(230)와 연결될 수 있다.
- [0082] 다수의 메모리 블록(BLK1~BLKz) 각각은 다수의 메모리 셀을 포함할 수 있다. 예를 들어, 다수의 메모리 셀은 불휘발성 메모리 셀들이며, 수직 채널 구조를 갖는 불휘발성 메모리 셀들로 구성될 수 있다.
- [0083] 메모리 셀 어레이(210)는 2차원 구조의 메모리 셀 어레이로 구성될 수 있으며, 경우에 따라서는, 3차원 구조의 메모리 셀 어레이로 구성될 수도 있다.
- [0084] 한편, 메모리 셀 어레이(210)에 포함되는 복수의 메모리 셀 각각은 적어도 1비트의 데이터를 저장할 수 있다. 일 예로, 메모리 셀 어레이(210)에 포함되는 복수의 메모리 셀 각각은 1비트의 데이터를 저장하는 싱글-레벨 셀(SLC: Single-Level Cell)일 수 있다. 다른 예로, 메모리 셀 어레이(210)에 포함되는 복수의 메모리 셀 각각은 2비트의 데이터를 저장하는 멀티-레벨 셀(MLC: Multi-Level Cell)일 수 있다. 또 다른 예로, 메모리 셀 어레이(210)에 포함되는 복수의 메모리 셀 각각은 3비트의 데이터를 저장하는 트리플-레벨 셀(TLC: Triple-Level Cell)일 수 있다. 또 다른 예로, 메모리 셀 어레이(210)에 포함되는 복수의 메모리 셀 각각은 4비트의 데이터를 저장하는 쿼드-레벨 셀(QLC: Quad-Level Cell)일 수 있다. 또 다른 예로, 메모리 셀 어레이(210)는 5비트 이상의 데이터를 각각 저장하는 복수의 메모리 셀을 포함할 수도 있다.
- [0085] 도 2를 참조하면, 어드레스 디코더(220), 읽기 및 쓰기 회로(230), 제어 로직(240) 및 전압 생성 회로(250) 등은 메모리 셀 어레이(210)를 구동하는 주변 회로로서 동작할 수 있다.
- [0086] 어드레스 디코더(220)는 다수의 워드 라인(WL)을 통해 메모리 셀 어레이(210)에 연결될 수 있다.
- [0087] 어드레스 디코더(220)는 제어 로직(240)의 제어에 응답하여 동작하도록 구성될 수 있다.
- [0088] 어드레스 디코더(220)는 메모리 장치(110) 내부의 입출력 버퍼를 통해 어드레스(Address)를 수신할 수 있다. 어드레스 디코더(220)는 수신된 어드레스 중 블록 어드레스(Block Address)를 디코딩 하도록 구성될 수 있다. 어드레스 디코더(220)는 디코딩된 블록 어드레스에 따라 적어도 하나의 메모리 블록을 선택할 수 있다.
- [0089] 어드레스 디코더(220)는 전압 생성 회로(250)로부터 읽기 전압(Vread) 및 패스 전압(Vpass)을 입력 받을 수 있다.
- [0090] 어드레스 디코더(220)는 읽기 동작 중 읽기 전압 인가 동작 시, 선택된 메모리 블록 내 선택된 워드 라인(WL)으로 읽기 전압(Vread)을 인가하고, 나머지 비 선택된 워드 라인들(WL)에는 패스 전압(Vpass)을 인가할 수 있다.
- [0091] 어드레스 디코더(220)는 프로그램 검증 동작 시, 선택된 메모리 블록 내 선택된 워드 라인(WL)에 전압 생성 회로(250)에서 발생된 검증 전압을 인가하고, 나머지 비 선택된 워드 라인들(WL)에 패스 전압(Vpass)을 인가할 수 있다.
- [0092] 어드레스 디코더(220)는 수신된 어드레스 중 열 어드레스를 디코딩 하도록 구성될 수 있다. 어드레스 디코더(220)는 디코딩 된 열 어드레스를 읽기 및 쓰기 회로(230)에 전송할 수 있다.
- [0093] 메모리 장치(110)의 읽기 동작 및 프로그램 동작은 페이지 단위로 수행될 수 있다. 읽기 동작 및 프로그램 동작 요청 시에 수신되는 어드레스는 블록 어드레스, 행 어드레스 및 열 어드레스 중 하나 이상을 포함할 수 있다.
- [0094] 어드레스 디코더(220)는 블록 어드레스 및 행 어드레스에 따라 하나의 메모리 블록 및 하나의 워드 라인을 선택할 수 있다. 열 어드레스는 어드레스 디코더(220)에 의해 디코딩 되어 읽기 및 쓰기 회로(230)에 제공될 수 있다.
- [0095] 어드레스 디코더(220)는 블록 디코더, 행 디코더, 열 디코더 및 어드레스 버퍼 등 중 하나 이상을 포함할 수 있

다.

- [0096] 읽기 및 쓰기 회로(230)는 다수의 페이지 버퍼(PB)를 포함할 수 있다. 읽기 및 쓰기 회로(230)는 메모리 셀 어레이(210)의 읽기 동작(Read Operation) 시에는 "읽기 회로(Read Circuit)"로 동작하고, 쓰기 동작(Write Operation) 시에는 "쓰기 회로(Write Circuit)"로 동작할 수 있다.
- [0097] 전술한 읽기 및 쓰기 회로(230)는 다수의 페이지 버퍼(PB)를 포함하는 페이지 버퍼 회로(Page Buffer Circuit) 또는 데이터 레지스터 회로(Data Register Circuit)라고도 한다. 여기서, 읽기 및 쓰기 회로(230)는 데이터 처리 기능을 담당하는 데이터 버퍼(Data Buffer)를 포함할 수 있고, 경우에 따라서, 캐싱 기능을 담당하는 캐쉬 버퍼(Cache Buffer)를 추가로 더 포함할 수 있다.
- [0098] 다수의 페이지 버퍼(PB)는 다수의 비트 라인(BL)을 통해 메모리 셀 어레이(210)에 연결될 수 있다. 다수의 페이지 버퍼(PB)는 읽기 동작 및 프로그램 검증 동작 시, 메모리 셀들의 문턱전압(Vth)을 센싱하기 위하여, 메모리 셀들과 연결된 비트 라인들(BL)에 센싱 전류를 계속적으로 공급하면서, 대응하는 메모리 셀의 프로그램 상태에 따라 흐르는 전류량이 변화되는 것을 센싱 노드를 통해 감지하여 센싱 데이터로 래치할 수 있다.
- [0099] 읽기 및 쓰기 회로(230)는 제어 로직(240)에서 출력되는 페이지 버퍼 제어 신호들에 응답하여 동작할 수 있다.
- [0100] 읽기 및 쓰기 회로(230)는 읽기 동작 시, 메모리 셀의 데이터를 센싱하여 독출 데이터를 임시 저장한 후, 메모리 장치(110)의 입출력 버퍼로 데이터(DATA)를 출력한다. 예시적인 실시 예로서, 읽기 및 쓰기 회로(230)는 페이지 버퍼들(PB) 또는 페이지 레지스터들 이외에도, 열 선택 회로 등을 포함할 수 있다.
- [0101] 제어 로직(240)은 어드레스 디코더(220), 읽기 및 쓰기 회로(230), 및 전압 생성 회로(250) 등과 연결될 수 있다. 제어 로직(240)은 메모리 장치(110)의 입출력 버퍼를 통해 커맨드(CMD) 및 제어 신호(CTRL)를 수신할 수 있다.
- [0102] 제어 로직(240)은 제어 신호(CTRL)에 응답하여 메모리 장치(110)의 제반 동작을 제어하도록 구성될 수 있다. 제어 로직(240)은 다수의 페이지 버퍼(PB)의 센싱 노드의 프리 차지 전위 레벨을 조절하기 위한 제어 신호를 출력할 수 있다.
- [0103] 제어 로직(240)은 메모리 셀 어레이(210)의 읽기 동작을 수행하도록 읽기 및 쓰기 회로(230)를 제어할 수 있다. 전압 생성 회로(250)는, 제어 로직(240)에서 출력되는 전압 생성 회로 제어 신호에 응답하여, 읽기 동작 시, 이용되는 읽기 전압(Vread) 및 패스 전압(Vpass)을 생성할 수 있다.
- [0104] 도 3은 본 발명의 실시예들에 따른 메모리 장치(110)의 각 메모리 블록(BLK) 를 개략적으로 나타낸 도면이다.
- [0105] 도 3을 참조하면, 메모리 장치(110)에 포함된 메모리 블록(BLK)은, 일 예로, 다수의 페이지(PG)와 다수의 스트링(STR)이 교차하는 방향으로 배치되어 구성될 수 있다.
- [0106] 다수의 페이지(PG)는 다수의 워드 라인(WL)과 대응되고, 다수의 스트링(STR)은 다수의 비트 라인(BL)과 대응된다.
- [0107] 메모리 블록(BLK)에는 다수의 워드 라인(WL)과 다수의 비트 라인(BL)이 교차하면서 배치될 수 있다. 예를 들어, 다수의 워드 라인(WL) 각각은 행 방향으로 배치되고, 다수의 비트 라인(BL) 각각은 열 방향으로 배치될 수 있다. 다른 예를 들어, 다수의 워드 라인(WL) 각각은 열 방향으로 배치되고, 다수의 비트 라인(BL) 각각은 행 방향으로 배치될 수 있다.
- [0108] 다수의 워드 라인(WL)과 다수의 비트 라인(BL)이 서로 교차하여, 다수의 메모리 셀(MC)이 정의될 수 있다. 각 메모리 셀(MC)에는 트랜지스터(TR)가 배치될 수 있다.
- [0109] 예를 들어, 각 메모리 셀(MC)에 배치된 트랜지스터(TR)는 드레인, 소스 및 게이트 등을 포함할 수 있다. 트랜지스터(TR)의 드레인(또는 소스)은 해당 비트 라인(BL)과 직접 또는 다른 트랜지스터(TR)를 경유하여 연결될 수 있다. 트랜지스터(TR)의 소스(또는 드레인)는 소스 라인(그라운드일 수 있음)과 직접 또는 다른 트랜지스터(TR)를 경유하여 연결될 수 있다. 트랜지스터(TR)의 게이트는 절연체에 둘러싸인 플로팅 게이트(FG: Floating Gate)와 워드 라인(WL)으로부터 게이트 전압이 인가되는 컨트롤 게이트(CG: Control Gate)를 포함할 수 있다.
- [0110] 다수의 메모리 블록(BLK1~BLKz) 각각에는, 2개의 최외곽 워드 라인 중 읽기 및 쓰기 회로(230)와 더 인접한 제1 최외곽 워드 라인의 바깥쪽에는 제1 선택 라인(소스 선택 라인 또는 드레인 선택 라인이라고도 함)이 더 배치될 수 있으며, 다른 제2 최외곽 워드 라인의 바깥쪽에는 제2 선택 라인(드레인 선택 라인 또는 소스 선택 라인이라

고도 함)이 더 배치될 수 있다.

- [0111] 경우에 따라서, 제1 최외곽 워드 라인과 제1 선택 라인 사이에는 하나 이상의 더미 워드 라인이 더 배치될 수 있다. 또한, 제2 최외곽 워드 라인과 제2 선택 라인 사이에도 하나 이상의 더미 워드 라인이 더 배치될 수 있다.
- [0112] 도 3과 같은 메모리 블록 구조를 가질 때, 읽기 동작 및 프로그램 동작(쓰기 동작)은 페이지 단위로 수행될 수 있으며, 소거(Erase) 동작은 메모리 블록 단위로 수행될 수 있다.
- [0113] 도 4는 본 발명의 실시예들에 따른 메모리 장치(110)의 워드 라인(WL) 및 비트 라인(BL)의 구조를 나타낸 도면이다.
- [0114] 도 4를 참조하면, 메모리 장치(110)에는, 메모리 셀들(MC)이 모여 있는 핵심 영역과 이 핵심 영역의 나머지 영역에 해당하며 메모리 셀 어레이(210)의 동작을 위해 서포트(Support) 해주는 보조 영역이 존재한다.
- [0115] 핵심 영역은 페이지들(PG)과 스트링들(STR)으로 구성될 수 있다. 이러한 핵심 영역에는, 다수의 워드 라인(WL1 ~ WL9)과 다수의 비트 라인(BL)이 교차하면서 배치된다.
- [0116] 다수의 워드 라인(WL1 ~ WL9)은 행 디코더(410)와 연결되고, 다수의 비트 라인(BL)은 열 디코더(420)와 연결될 수 있다. 다수의 비트 라인(BL)과 열 디코더(420) 사이에는 읽기 및 쓰기 회로(230)에 해당하는 데이터 레지스터(430)가 존재할 수 있다.
- [0117] 다수의 워드 라인(WL1 ~ WL9)은 다수의 페이지(PG)와 대응된다.
- [0118] 예를 들어, 도 4와 같이 다수의 워드 라인(WL1 ~ WL9) 각각은 하나의 페이지(PG)와 대응될 수 있다. 이와 다르게, 다수의 워드 라인(WL1 ~ WL9) 각각이 사이즈가 큰 경우, 다수의 워드 라인(WL1 ~ WL9) 각각은 둘 이상(예: 2개 또는 4개)의 페이지(PG)와 대응될 수도 있다. 페이지(PG)는 프로그램 동작과 읽기 동작을 진행하는데 있어서 최소 단위가 되며, 프로그램 동작 및 읽기 동작 시, 동일 페이지(PG) 내에서의 모든 메모리 셀(MC)은 동시 동작을 수행할 수 있다.
- [0119] 다수의 비트 라인(BL)은 홀수 번째 비트 라인(BL)과 짝수 번째 비트 라인(BL)을 구분되면서 열 디코더(420)와 연결될 수 있다.
- [0120] 메모리 셀(MC)에 액세스 하기 위해서는, 주소가 먼저 입출력 단을 거쳐 행 디코더(410)와 열 디코더(420)를 통하여 핵심 영역으로 들어와서, 타깃 메모리 셀을 지정할 수 있다. 타깃 메모리 셀을 지정한다는 것은 행 디코더(410)와 연결된 워드 라인들(WL1 ~ WL9)과 열 디코더(420)와 연결된 비트 라인들(BL)의 교차되는 사이트에 있는 메모리 셀(MC)에 데이터를 프로그램 하거나 프로그램 된 데이터를 읽어 내기 위하여 액세스 한다는 것을 의미한다.
- [0121] 제1 방향(예: X축 방향)의 페이지(PG)는 워드 라인(WL)이란 공통으로 사용하는 라인으로 묶여 있으며, 제2 방향(예: Y축 방향)의 스트링(STR)도 비트 라인(BL)이란 공통 라인으로 묶여(연결되어) 있다. 공통으로 묶여 있다는 것은 구조적으로 동일한 물질로 연결되어 있고, 전압 인가 시에도 모두 동일한 전압이 동시에 인가된다는 것을 의미한다. 물론, 직렬로 연결된 중간 위치나 마지막 위치의 메모리 셀(MC)은 앞의 메모리 셀(MC)의 전압 강하에 의하여, 처음에 위치하는 메모리 셀(MC)과 맨 마지막에 위치하는 메모리 셀(MC)에 인가되는 전압은 약간 다를 수 있다.
- [0122] 메모리 장치(110)의 데이터 처리 모두는, 데이터 레지스터(430)를 경유하여 프로그램 및 읽기가 되므로, 데이터 레지스터(430)는 중추적 역할을 한다. 데이터 레지스터(430)의 데이터 처리가 늦어지면 다른 모든 영역에서는 데이터 레지스터(430)가 데이터 처리를 완료할 때까지 기다려야 한다. 또한, 데이터 레지스터(430)의 성능이 저하되면, 메모리 장치(110)의 전체 성능을 저하시킬 수 있다.
- [0123] 도 4의 예시를 참조하면, 1개의 스트링(STR)에는, 다수의 워드 라인(WL1 ~ WL9)과 연결되는 다수의 트랜지스터(TR1 ~ TR9)가 존재할 수 있다. 다수의 트랜지스터(TR1 ~ TR9)가 존재하는 영역들이 메모리 셀들(MC)에 해당한다. 여기서, 다수의 트랜지스터(TR1 ~ TR9)는 전술한 바와 같이, 제어 게이트(CG)와 플로팅 게이트(FG)를 포함하는 트랜지스터들이다.
- [0124] 다수의 워드 라인(WL1 ~ WL9)은 2개의 최외곽 워드 라인(WL1, WL9)을 포함한다. 2개의 최외곽 워드 라인(WL1, WL9) 중 신호 경로적 측면에서 데이터 레지스터(430)와 더 인접한 제1 최외곽 워드 라인(WL1)의 바깥쪽에는 제1 선택 라인(DSL)이 더 배치되고, 다른 제2 최외곽 워드 라인(WL9)의 바깥쪽에는 제2 선택 라인(SSL)이 더 배치될

수 있다.

- [0125] 제1 선택 라인(DSL)에 의해 온-오프가 제어되는 제1 선택 트랜지스터(D-TR)는 제1 선택 라인(DSL)과 연결된 게이트 전극을 가지고 있을 뿐, 플로팅 게이트(FG)를 포함하지 않는 트랜지스터이다. 제2 선택 라인(SSL)에 의해 온-오프가 제어되는 제2 선택 트랜지스터(S-TR)는 제2 선택 라인(SSL)과 연결된 게이트 전극을 가지고 있을 뿐, 플로팅 게이트(FG)를 포함하지 않는 트랜지스터이다.
- [0126] 제1 선택 트랜지스터(D-TR)는 해당 스트링(STR)과 데이터 레지스터(430) 간의 연결을 온 또는 오프 시키는 스위치 역할을 한다. 제2 선택 트랜지스터(S-TR)는 해당 스트링(STR)과 소스 라인(SL) 간의 연결을 온 또는 오프 시켜주는 스위치 역할을 한다. 즉, 제1 선택 트랜지스터(D-TR) 및 제2 선택 트랜지스터(S-TR)는 해당 스트링(STR)의 양쪽 끝에 있으면서, 신호를 이어주고 끊어내는 문지기 역할을 한다.
- [0127] 메모리 시스템(100)은, 프로그램 동작 시, 프로그램 할 비트 라인(BL)의 타겟 메모리 셀(MC)에 전자를 채워야 하기 때문에, 제1 선택 트랜지스터(D-TR)의 게이트 전극에 소정의 턴-온 전압(Vcc)를 인가하여 제1 선택 트랜지스터(D-TR)를 턴-온 시키고, 제2 선택 트랜지스터(S-TR)의 게이트 전극에는 소정의 턴-오프 전압(예: 0V)을 인가하여 제2 선택 트랜지스터(S-TR)를 턴-오프 시킨다.
- [0128] 메모리 시스템(100)은, 읽기 동작 또는 검증(Verification) 동작 시, 제1 선택 트랜지스터(D-TR) 및 제2 선택 트랜지스터(S-TR)를 모두 턴-온 시켜준다. 이에 따라, 전류가 해당 스트링(STR)을 관통하여 그라운드에 해당하는 소스 라인(SL)으로 빠질 수 있어서, 비트 라인(BL)의 전압 레벨이 측정될 수 있다. 다만, 읽기 동작 시, 제1 선택 트랜지스터(D-TR) 및 제2 선택 트랜지스터(S-TR)의 온-오프 타이밍의 시간 차이가 있을 수 있다.
- [0129] 메모리 시스템(100)은, 소거(Erase) 동작 시, 소스 라인(SL)을 통하여 기판(Substrate)에 소정 전압(예: +20V)을 공급하기도 한다. 메모리 시스템(100)은, 소거(Erase) 동작 시, 제1 선택 트랜지스터(D-TR) 및 제2 선택 트랜지스터(S-TR)를 모두 플로팅(Floating) 시켜서 무한대의 저항을 만들어 준다. 이에 따라, 제1 선택 트랜지스터(D-TR) 및 제2 선택 트랜지스터(S-TR)의 역할이 없도록 해주고, 플로팅 게이트(FG)와 기판(Substrate) 사이에서만 전위 차이에 의한 전자(electron)가 동작할 수 있도록 구조화 되어 있다.
- [0130] 도 5는 본 발명의 실시예들에 따른 메모리 시스템(100)에서 메인 코어(MC) 및 서브 코어(SC)의 구조를 나타낸 구성도이다.
- [0131] 도 5를 참조하면, 메모리 시스템(100)의 메모리 컨트롤러(120)는 하나의 메인 코어(MC) 및 복수의 서브 코어(SC)를 포함할 수 있다.
- [0132] 메인 코어(MC)는 호스트(HOST)로부터 커맨드를 수신하고, 해당 커맨드의 처리 결과를 호스트(HOST)로 전송할 수 있다. 그리고 메인 코어(MC)는 복수의 서브 코어(SC)와 통신할 수 있으며 복수의 서브 코어(SC) 각각의 동작을 제어할 수 있다.
- [0133] 메인 코어(MC)는 호스트(HOST)로부터 타겟 커맨드를 수신할 수 있다. 타겟 커맨드는 메모리 장치(110)에 타겟 데이터를 라이트하는 동작을 지시하는 커맨드이다.
- [0134] 메모리 장치(110)는 복수의 메모리 영역(AREA)를 포함할 수 있다. 각 메모리 영역(AREA)은 다양한 방법으로 결정될 수 있다.
- [0135] 일 예로 메모리 장치(110)는 복수의 메모리 다이를 포함하고, 각 메모리 다이가 하나의 메모리 영역(AREA)에 대응할 수 있다.
- [0136] 이때, 복수의 메모리 다이 각각은 복수의 메모리 블록을 포함할 수 있다. 복수의 메모리 블록 각각은 복수의 페이지를 포함할 수 있다.
- [0137] 다른 예로 각 메모리 영역(AREA)은 메모리 장치(110)에 포함된 복수의 메모리 블록 중 하나 이상을 포함할 수 있다.
- [0138] 복수의 서브 코어(SC) 각각은 메인 코어(MC)와 통신하고, 메모리 장치(110)에 포함된 복수의 메모리 영역(AREA) 중 하나 이상의 메모리 영역에 대한 데이터 라이트(저장) 동작을 제어할 수 있다.
- [0139] 이때, 메모리 영역(AREA)에 데이터가 라이트된다는 것은 해당 메모리 영역(AREA)에 포함된 메모리 다이 또는 메모리 블록에 데이터가 라이트된다는 것을 의미한다.
- [0140] 이때, 메모리 장치(110)에 포함된 복수의 메모리 영역(AREA) 각각은 복수의 서브 코어(SC) 중 하나의 서브 코어

에 의해서만 제어된다. 즉, 두 개 이상의 서브 코어(SC)가 하나의 메모리 영역(AREA)을 함께 제어하지 않는다.

- [0141] 복수의 서브 코어(SC) 각각은 서로 다른 메모리 영역(AREA)에 대한 라이트 동작을 독립적으로 제어할 수 있다. 복수의 서브 코어(SC) 각각은 자신이 제어하는 메모리 영역(AREA)에 대한 데이터 라이트 동작을 제어할 때, 다른 서브 코어의 영향을 받지 않는다.
- [0142] 그리고 복수의 서브 코어(SC) 각각은 메인 코어(MC)와 통신할 수 있지만 각 서브 코어(SC)끼리는 서로 통신하지 않는다.
- [0143] 한편, 메인 코어(MC)와 복수의 서브 코어(SC)는 메모리 컨트롤러(120) 내의 프로세서(124)에 포함된 복수의 프로세서 코어로 구현될 수 있다.
- [0144] 도 6은 본 발명의 실시예들에 따른 메모리 시스템(100)에서 데이터 유닛(DU)을 라이트하는 동작을 나타낸 도면이다.
- [0145] 도 6을 참조하면, 메모리 시스템(100)의 메모리 컨트롤러(120)에 포함된 메인 코어(MC)는 호스트(HOST)로부터 메모리 장치(110)에 타겟 데이터(TGT_DATA)를 라이트하는 동작을 지시하는 타겟 커맨드(TGT_CMD)를 수신하면, 먼저 타겟 데이터(TGT_DATA)를 임계 사이즈(THR) 이하인 하나 이상의 데이터 유닛(DU)으로 분할하여 복수의 서브 코어(SC)에 할당할 수 있다.
- [0146] 예를 들어, 타겟 데이터(TGT_DATA)가 100KB이고 임계 사이즈(THR)가 30KB이면, 메모리 컨트롤러(120)의 메인 코어(MC)는 타겟 데이터(TGT_DATA)를 30KB 크기의 데이터 유닛 3개와 10KB 크기의 데이터 유닛 1개로 분할할 수 있다.
- [0147] 만약 타겟 데이터(TGT_DATA)의 크기가 임계 사이즈(THR) 이하이면 메모리 컨트롤러(120)의 메인 코어(MC)는 타겟 데이터(TGT_DATA)를 하나의 데이터 유닛으로 분할할 수 있다.
- [0148] 그리고 메모리 컨트롤러(120)는 복수의 서브 코어(SC)가 하나 이상의 데이터 유닛(DU) 중 제1 데이터 유닛(DU_1)을 메모리 장치(110)에 원자적(atomic)으로 라이트하도록 제어할 수 있다.
- [0149] 여기서 메모리 컨트롤러(120)가 제1 데이터 유닛(DU_1)을 메모리 장치(110)에 원자적(atomic)으로 라이트하도록 제어한다는 것은, 메모리 컨트롤러(120)가 제1 데이터 유닛(DU_1) 전체가 메모리 장치(110)에 라이트되거나 또는 제1 데이터 유닛(DU_1) 전체가 메모리 장치(110)에 라이트되지 않도록 보장한다는 의미이다. 즉, 메모리 컨트롤러(120)가 제1 데이터 유닛(DU_1) 중 일부만이 메모리 장치(110)에 라이트되는 상태가 유지되지 않도록 제어한다는 의미이다.
- [0150] 일 예로, 제1 데이터 유닛(DU_1)이 메모리 장치(110)에 라이트되는 도중 SPO(Sudden Power Off)가 발생한 경우를 가정한다. 이때, 메모리 컨트롤러(120)는 SPO에 대한 리커버리 동작 시에 메모리 장치(110)에 라이트된 제1 데이터 유닛(DU_1)의 일부분을 롤백(rollback)하여 제1 데이터 유닛(DU_1) 전체가 메모리 장치(110)에 라이트되지 않도록 할 수 있다.
- [0151] 메모리 장치(110)에 라이트된 제1 데이터 유닛(DU_1)의 일부분을 롤백(rollback)하기 위해서, 메모리 컨트롤러(120)는 제1 데이터 유닛(DU_1)의 일부분에 대한 맵 정보(즉, 논리 주소와 물리 주소 간의 매핑 정보)를 제1 데이터 유닛의 일부분이 메모리 장치(110)에 라이트되기 이전으로 복원할 수 있다.
- [0152] 다른 예로 메모리 컨트롤러(120)는 SPO에 대한 리커버리 동작 시에 메모리 장치(110)에 라이트되지 않은 제1 데이터 유닛(DU_1)의 나머지 부분을 메모리 장치(110)에 라이트하여 제1 데이터 유닛(DU_1) 전체가 메모리 장치(110)에 라이트되게 할 수도 있다.
- [0153] 이하, 메모리 컨트롤러(120)가 제1 데이터 유닛(DU_1)을 메모리 장치(110)에 원자적으로 라이트하는 구체적인 방법에 대해 설명한다.
- [0154] 도 7은 본 발명의 실시예들에 따른 메모리 시스템(100)에서 데이터 유닛을 메모리 장치(110)에 서브 데이터 유닛(SUB_DU)으로 분산하여 라이트하는 동작을 나타낸 도면이다.
- [0155] 도 7을 참조하면, 전술한 바와 같이 메모리 시스템(100)의 메모리 컨트롤러(120)는 메인 코어(MC)가 호스트(HOST)로부터 메모리 장치(110)에 타겟 데이터(TGT_DATA)를 라이트하는 동작을 지시하는 타겟 커맨드(TGT_CMD)를 수신하면, 먼저 타겟 데이터(TGT_DATA)를 임계 사이즈(THR) 이하인 복수의 데이터 유닛(DU)으로 분할할 수 있다.

- [0156] 메모리 컨트롤러(120)는 하나 이상의 데이터 유닛(DU) 중 하나인 제1 데이터 유닛(DU_1)을 복수의 서브 데이터 유닛(SUB_DU)으로 분산하여 메모리 장치(110)에 포함된 복수의 메모리 영역(AREA)에 라이트할 수 있다.
- [0157] 서브 데이터 유닛(SUB_DU) 각각은 제1 데이터 유닛의 일부(DU_1)를 포함할 수 있으며 서브 데이터 유닛(SUB_DU) 각각의 사이즈는 서로 다를 수 있다. 그리고 서브 데이터 유닛(SUB_DU) 각각은 서로 다른 메모리 영역(AREA)에 라이트될 수 있다.
- [0158] 이때, 메모리 컨트롤러(120)에 포함된 복수의 서브 코어(SC) 각각은 복수의 서브 데이터 유닛(SUB_DU) 중 일부를 자신이 제어하는 메모리 영역(AREA)에 라이트하도록 제어할 수 있다.
- [0159] 도 8은 본 발명의 실시예들에 따른 메모리 시스템(100)에서 서브 데이터 유닛(SUB_DU)을 메모리 장치(110)에 라이트하는 동작의 일 예를 나타낸 도면이다.
- [0160] 메모리 시스템(100)의 메모리 컨트롤러(120)는 도 7에서 전술한 바와 같이 제1 데이터 유닛(DU_1)을 복수의 서브 데이터 유닛(SUB_DU)으로 분할할 수 있다.
- [0161] 그리고 메모리 컨트롤러(120)에 포함된 복수의 서브 코어(SC) 중 제1 서브 코어(SC_1)는 복수의 서브 데이터 유닛(SUB_DU) 중 제1 서브 데이터 유닛(SUB_DU_1)을 타겟 메모리 영역에 라이트할 수 있다. 이때, 타겟 메모리 영역은 메모리 장치(110)에 포함된 복수의 메모리 영역(AREA) 중에서 제1 서브 코어가 데이터 라이트를 제어하는 메모리 영역중 하나일 수 있다.
- [0162] 이때, 제1 서브 코어(SC_1)는 제1 서브 데이터 유닛(SUB_DU_1)을 타겟 메모리 영역에 라이트할 때, 타겟 커맨드의 아이디(CID)와 제1 서브 데이터 유닛(SUB_DU_1)의 사이즈(SIZE) 값을 함께 라이트할 수 있다.
- [0163] 이때, 타겟 커맨드의 아이디(CID)는 메인 코어에 의해 생성될 수 있다. 예를 들어 메인 코어는 시간이 흐를수록 증가하는 별도의 카운트 값을 이용하거나 또는 현재 시간 등의 인자를 시드(seed)로 생성한 랜덤한 숫자 값을 이용하여 타겟 커맨드의 아이디(CID)를 생성할 수 있다.
- [0164] 이처럼 제1 서브 코어(SC_1)가 타겟 커맨드의 아이디(CID)와 제1 서브 데이터 유닛(SUB_DU_1)의 사이즈(SIZE) 값을 제1 서브 데이터 유닛(SUB_DU_1)과 함께 라이트하는 이유는, 메모리 컨트롤러(120)가 제1 데이터 유닛(DU_1)을 메모리 장치(110)에 원자적(atomic)으로 라이트하도록 제어하기 위해서이다.
- [0165] SPO 발생 후 리커버리 동작 시에, 메모리 컨트롤러(120)는 제1 데이터 유닛을 메모리 장치(110)에 원자적(atomic)으로 라이트하도록 제어하기 위해 메모리 장치(110)에 라이트된 제1 데이터 유닛(DU_1)의 일부분을 롤백하여 제1 데이터 유닛(DU_1)의 일부분이 라이트되기 이전 상태로 복귀한다.
- [0166] 이를 위해 메모리 컨트롤러(120)는 메모리 장치(110)에 라이트된 데이터 중에서 어느 부분이 제1 데이터 유닛(DU_1)에 대한 서브 데이터 유닛(SUB_DU)인지 탐색해야 한다.
- [0167] 메모리 컨트롤러(120)는 메모리 장치(110)에 라이트된 데이터 중 어느 부분이 제1 데이터 유닛(DU_1)에 대한 서브 데이터 유닛(SUB_DU)인지를 함께 라이트된 타겟 커맨드의 아이디(CID)로 판단할 수 있다. 만약 메모리 장치(110)에 라이트된 데이터와 함께 라이트된 커맨드의 아이디가 타겟 커맨드의 아이디(CID)와 다르다면, 메모리 컨트롤러(120)는 해당 데이터는 제1 데이터 유닛(DU_1)에 대한 서브 데이터 유닛(SUB_DU)이 아니라고 판단할 수 있다.
- [0168] 그리고 메모리 컨트롤러(120)는 메모리 장치(110)에 라이트된 서브 데이터 유닛(SUB_DU)의 크기를 함께 라이트된 사이즈(SIZE) 값으로 판단할 수 있다.
- [0169] 도 9는 본 발명의 실시예들에 따른 메모리 시스템(100)에서 서브 데이터 유닛(SUB_DU)을 메모리 장치(110) 내부의 메모리 영역(AREA)에 라이트하는 동작의 일 예를 나타낸 도면이다.
- [0170] 도 9를 참조하면, 서브 데이터 유닛(SUB_DU)은 메모리 장치(110) 내부의 하나 이상의 메모리 영역(AREA)에 포함된 페이지에 분산되어 라이트(저장)될 수 있다. 서브 데이터 유닛(SUB_DU)이 분산되어 저장되는 페이지 각각은 하나 이상의 메모리 영역(AREA) 중 어느 하나에 포함된 메모리 블록에 포함될 수 있다.
- [0171] 이때, 서브 데이터 유닛(SUB_DU)에 대응하는 타겟 커맨드의 아이디(CID)와 서브 데이터 유닛(SUB_DU)의 사이즈 값(SIZE)은 서브 데이터 유닛(SUB_DU)이 분산되어 저장되는 복수의 페이지 별로 라이트될 수 있다.
- [0172] 예를 들어, 서브 데이터 유닛(SUB_DU)에 대응하는 타겟 커맨드의 아이디(CID)가 300이고 서브 데이터 유닛(SUB_DU)의 사이즈 값(SIZE)이 12KB라고 가정한다. 그리고 서브 데이터 유닛(SUB_DU)이 3개의 페이지(Page1,

Page2, Page3)에 분산되어 저장된다고 가정한다.

- [0173] 이때, 3개의 페이지(Page1, Page2, Page3) 각각에 서브 데이터 유닛(SUB_DU)에 대응하는 타겟 커맨드의 아이디(CID)인 300과 서브 데이터 유닛(SUB_DU)의 사이즈 값(SIZE)이 12KB가 라이트될 수 있다.
- [0174] 도 10은 본 발명의 실시예들에 따른 메모리 시스템(100)에서 서브 데이터 유닛(SUB_DU)을 메모리 장치(110) 내부의 메모리 영역(AREA)에 라이트하는 동작의 다른 예를 나타낸 도면이다.
- [0175] 도 10을 참조하면, 도 9와 마찬가지로 서브 데이터 유닛(SUB_DU)은 메모리 장치(110) 내부의 하나 이상의 메모리 영역(AREA)에 포함된 페이지에 분산되어 라이트(저장)될 수 있다. 그리고 서브 데이터 유닛(SUB_DU)에 대응하는 타겟 커맨드의 아이디(CID)와 서브 데이터 유닛(SUB_DU)의 사이즈 값(SIZE)은 서브 데이터 유닛(SUB_DU)이 분산되어 저장되는 복수의 페이지 별로 라이트될 수 있다.
- [0176] 이때, 서브 데이터 유닛(SUB_DU)에 대응하는 타겟 커맨드의 아이디(CID)와 서브 데이터 유닛(SUB_DU)의 사이즈 값(SIZE)은 스페어 영역(SPARE_AREA)에 라이트될 수 있다.
- [0177] 스페어 영역(SPARE_AREA)은 페이지에 대한 메타 정보(e.g. 페이지의 물리 주소에 대응하는 논리 주소)가 저장되는 영역이다. 스페어 영역은 메모리 블록에 포함된 특정 페이지일 수도 있고, 도 10과 같이 서브 데이터 유닛(SUB_DU)이 분산되어 저장되는 복수의 페이지 각각에 위치할 수도 있다.
- [0178] 이상에서는 메모리 시스템(100)에서 타겟 데이터를 메모리 장치(110)에 라이트하는 동작에 대해 설명하였다.
- [0179] 이하에서는, SPO가 발생한 이후 리커버리 동작 시에 타겟 데이터에 대한 데이터 유닛이 메모리 장치(110)에 원자적으로 라이트되도록 하는 메모리 시스템(100)의 동작을 설명한다.
- [0180] 도 11은 본 발명의 실시예들에 따른 메모리 시스템(100)에서 SPOR(Sudden Power Off Recovery) 동작을 나타낸 도면이다.
- [0181] 메모리 시스템(100)의 메모리 컨트롤러(120)가 메모리 장치(110)에 데이터를 라이트하는 도중에 SPO(Sudden Power Off)가 발생하면, 메모리 컨트롤러(120)는 SPO 이후 재부팅 시에 SPOR(Sudden Power Off Recovery) 동작을 수행하여 메모리 장치(110)를 복원하고자 한다.
- [0182] SPOR 동작 시에, 메모리 컨트롤러(120)는 SPO 발생시 메모리 장치(110)에 라이트 중인 데이터에 대한 맵 정보를 업데이트해야 한다. 이때, 데이터에 대한 맵 정보를 업데이트한다는 것은 해당 데이터에 대한 맵 정보를 메모리 장치(110)에 라이트하고 메모리 컨트롤러(120) 내부의 맵 캐시에 캐싱한다는 의미이다.
- [0183] SPO가 발생하면 메모리 장치(110)에 라이트되는 중인 데이터에 대한 맵 정보가 메모리 장치(110)에 라이트되지 않고 손실될 수 있기 때문이다.
- [0184] 따라서, 메모리 컨트롤러(120)는 메모리 장치(110)를 스캔(scan)하면서, SPO 발생시 메모리 장치(110)에 라이트 중인 데이터인 제1 데이터 유닛에 대한 서브 데이터 유닛(SUB_DU) 중 메모리 장치(110)에 라이트된 서브 데이터 유닛(SUB_DU)에 대한 맵 정보를 업데이트한다.
- [0185] 메모리 컨트롤러(120)는 맵 정보를 업데이트할 때, 도 10에서 설명한 스페어 영역(SPARE_AREA)에 페이지의 물리적 주소에 대응하는 논리 주소를 라이트할 수 있다.
- [0186] 그러나 SPO로 인해 복수의 메모리 영역(AREA)에 라이트된 서브 데이터 유닛(SUB_DU) 중 일부에 UECC(Uncorrectable ECC)가 발생한 경우 해당 서브 데이터 유닛에 대한 맵 정보는 업데이트되지 않아야 한다. 서브 데이터 유닛(SUB_DU) 중 일부에 UECC가 발생했다는 것은 해당 서브 데이터 유닛(SUB_DU)에 대한 라이트가 실패했다는 것을 의미하기 때문이다.
- [0187] 이 경우 메모리 컨트롤러(120)는 제1 데이터 유닛이 원자적(atomic)으로 라이트되는 것을 보장하기 위해서, 제1 데이터 유닛에 대한 모든 서브 데이터 유닛(SUB_DU)을 언맵(un-map)해야 한다. 이때, 데이터를 언맵한다는 것은 맵 정보를 해당 데이터가 라이트되기 이전 상태로 롤백(roll-back)하는 것을 의미한다. 따라서, 데이터가 언맵되면 해당 데이터는 더 이상 리드되지 않는다.
- [0188] 즉, 메모리 컨트롤러(120)는 제1 데이터 유닛에 대한 모든 서브 데이터 유닛(SUB_DU)에 대한 맵 정보를 롤백하여 제1 데이터 유닛 전체가 메모리 장치(110)에 라이트되지 않은 상태가 되도록 제어해야 한다.
- [0189] 이하, 도 12 내지 도 14에서 메모리 컨트롤러(120)가 제1 데이터 유닛에 대한 모든 서브 데이터 유닛을 언맵하

는 구체적인 동작을 설명한다.

- [0190] 도 12 내지 도 14는 본 발명의 실시예들에 따른 메모리 시스템(100)에서 SPOR 동작 시 UECC가 검출되는 경우의 동작을 나타낸 도면이다.
- [0191] 먼저, 메모리 장치(110)에 라이트된 제1 데이터 유닛에 대한 서브 데이터 유닛 중 제1 서브 데이터 유닛(SUB_DU_1)에서 UECC가 검출되었다고 가정한다. 그리고 제1 서브 데이터 유닛(SUB_DU_1)은 메모리 시스템(100)의 메모리 컨트롤러(120)에 포함된 복수의 서브 코어(SC) 중에서 제1 서브 코어(SC_1)에 의해 라이트되었다고 가정한다.
- [0192] 도 12에서, 제1 서브 코어(SC_1)는 제1 서브 데이터 유닛(SUB_DU_1)에서 UECC가 검출되면, 제1 서브 데이터 유닛(SUB_DU_1)과 함께 저장된 타겟 커맨드의 아이디(CID)를 메모리 영역(AREA) 중 제1 서브 데이터 유닛(SUB_DU_1)이 저장된 타겟 메모리 영역에서 리드할 수 있다.
- [0193] 도 13에서, 제1 서브 코어(SC_1)는 타겟 메모리 영역에서 리드한 타겟 커맨드의 아이디(CID)를 메인 코어(MC)로 리포트할 수 있다.
- [0194] 그리고 메인 코어(MC)는 복수의 서브 코어(SC) 중 제1 서브 코어(SC_1)와 다른 제2 서브 코어(SC_2)가, 제1 데이터 유닛에 대한 서브 데이터 유닛 중 제2 서브 코어(SC_2)가 제어하는 메모리 영역에 라이트된 제2 서브 데이터 유닛(SUB_DU_2)을 언맵(un-map)하도록 지시한다.
- [0195] 이를 위해 메인 코어(MC)는 제1 서브 코어(SC_1)로부터 리포트된 타겟 커맨드의 아이디(CID)를 제2 서브 코어(SC_2)에 리포트할 수 있다. 제2 서브 코어(SC_2)가 제어하는 메모리 영역에 라이트된 데이터 중에서 제2 서브 데이터 유닛(SUB_DU_2)이 무엇인지를 타겟 커맨드의 아이디(CID)를 기초로 판단할 수 있기 때문이다.
- [0196] 이처럼 복수의 서브 코어(SC) 중 제1 서브 코어(SC_1)가 제2 서브 코어(SC_2)에 타겟 커맨드의 아이디(CID)를 직접 리포트하는 대신에, 메인 코어(MC)를 경유하여 타겟 커맨드의 아이디(CID)를 리포트하는 이유는 전술한 바와 같이 각 서브 코어(SC)는 서로 통신하지 않고 독립적으로 동작하기 때문이다.
- [0197] 도 14에서, 제2 서브 코어(SC_2)는 메인 코어(MC)로부터 수신한 타겟 커맨드의 아이디(CID)를 기초로 하여, 제2 서브 코어(SC_2)가 제어하는 메모리 영역에 라이트된 제2 서브 데이터 유닛(SUB_DU_2)을 언맵(un-map)한다.
- [0198] 이하, 도 15에서 전술한 도 11 내지 도 14에서 설명한 내용을 흐름도를 이용하여 설명한다.
- [0199] 도 15는 본 발명의 실시예들에 따른 메모리 시스템(100)에서 SPOR 동작을 나타낸 흐름도이다.
- [0200] 이하, 메모리 시스템(100)의 메모리 컨트롤러(120)에 의해 SPOR 동작이 실행되는 것을 예시적으로 설명한다.
- [0201] 먼저, 메모리 컨트롤러(120)는 제1 데이터 유닛에 대한 복수의 서브 데이터 유닛을 메모리 장치(110)에서 스캔하면서 제1 서브 데이터 유닛을 결정한다(S1510).
- [0202] 메모리 컨트롤러(120)는 S1510 단계에서 결정된 제1 서브 데이터 유닛에 대한 에러 체크를 수행하여 제1 서브 데이터 유닛에 UECC가 발생했는지를 판단한다(S1520).
- [0203] 만약 제1 서브 데이터 유닛에 UECC가 발생한 경우(S1530-Y), 메모리 컨트롤러(120)에 포함된 복수의 서브 코어 중 제1 서브 코어는 타겟 커맨드의 아이디를 메인 코어로 리포트한다(S1540).
- [0204] 그리고 메모리 컨트롤러(120)에 포함된 메인 코어는 제1 서브 코어로부터 수신한 타겟 커맨드의 아이디를 전술한 복수의 서브 코어 중 제1 서브 코어와 다른 제2 서브 코어로 전달한다(S1550).
- [0205] 그리고 메모리 컨트롤러(120)에 포함된 제2 서브 코어는 복수의 서브 데이터 유닛 중 제1 서브 데이터 유닛과 다른 제2 서브 데이터 유닛을 언맵하고(S1560), S1510 단계로 이동한다.
- [0206] 반면 제1 서브 데이터 유닛에 UECC가 발생하지 않은 경우(S1530-N), 메모리 컨트롤러(120)는 S1510 단계로 이동한다.
- [0207] 도 16은 본 발명의 실시예들에 따른 메모리 컨트롤러(120)의 동작 방법을 나타낸 흐름도이다.
- [0208] 여기서, 메모리 컨트롤러(120)는 하나의 메인 코어 및 메인 코어와 통신하고 메모리 장치(110)에 포함된 복수의 메모리 영역 중 하나 이상의 메모리 영역에 대한 데이터 라이트 동작을 제어하는 복수의 서브 코어를 포함한다.
- [0209] 우선 메모리 컨트롤러(120)의 동작 방법은 메인 코어가 메모리 장치에 타겟 데이터를 라이트하는 동작에 대한

타겟 커맨드를 호스트로부터 수신하는 단계를 포함할 수 있다(S1610).

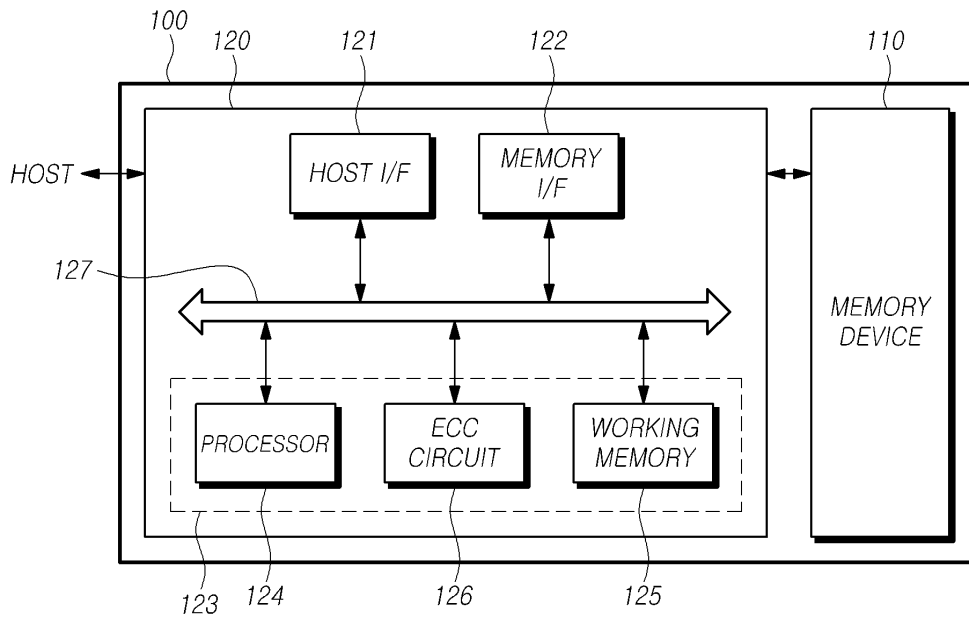
- [0210] 그리고 메모리 컨트롤러(120)의 동작 방법은 메인 코어가 타겟 데이터를 임계 사이즈 이하인 하나 이상의 데이터 유닛으로 분할하여 복수의 서브 코어에 할당하는 단계를 포함할 수 있다(S1620).
- [0211] 마지막으로 메모리 컨트롤러(120)의 동작 방법은 메모리 컨트롤러(120)에 포함된 복수의 서브 코어가 복수의 데이터 유닛 중 제1 데이터 유닛을 메모리 장치(110)에 원자적(atomic)으로 라이트하도록 제어하는 단계를 포함할 수 있다(S1630).
- [0212] 한편, 이상에서 설명한 메모리 컨트롤러(120)의 동작은 제어 회로(123)에 의해 제어될 수 있으며, 프로세서(124)가 메모리 컨트롤러(120)의 제반 동작이 프로그램된 펌웨어를 실행(구동)하는 방식으로 수행될 수 있다.
- [0213] 도 17는 본 발명의 실시예들에 따른 컴퓨팅 시스템(1700)의 구성도이다.
- [0214] 도 17을 참조하면, 본 발명의 실시예들에 따른 컴퓨팅 시스템(1700)은 시스템 버스(1760)에 전기적으로 연결되는 메모리 시스템(100), 컴퓨팅 시스템(1700)의 전반적인 동작을 제어하는 중앙처리장치(CPU, 1710), 컴퓨팅 시스템(1700)의 동작과 관련한 데이터 및 정보를 저장하는 램(RAM, 1720), 사용자에게 사용 환경을 제공하기 위한 UI/UX (User Interface/User Experience) 모듈(1730), 외부 장치와 유선 및/또는 무선 방식으로 통신하기 위한 통신 모듈(1740), 컴퓨팅 시스템(1700)이 사용하는 파워를 관리하는 파워 관리 모듈(1750) 등을 포함할 수 있다.
- [0215] 컴퓨팅 시스템(1700)은 PC(Personal Computer)이거나, 스마트 폰, 태블릿 등의 모바일 단말, 또는 각종 전자 기기 등을 포함할 수 있다.
- [0216] 컴퓨팅 시스템(1700)은, 동작 전압을 공급하기 위한 배터리를 더 포함할 수 있으며, 응용 칩셋(Application Chipset), 그래픽 관련 모듈, 카메라 이미지 프로세서(Camera Image Processor: CIS), 디램 등을 더 포함할 수도 있다. 이외에도, 이 분야의 통상적인 지식을 습득한 자들에게 자명하다.
- [0217] 한편, 메모리 시스템(100)은, 하드 디스크 드라이브(HDD: Hard Disk Drive)와 같이 자기 디스크에 데이터를 저장하는 장치뿐 아니라, 솔리드 스테이트 드라이브(SSD: Solid State Drive), UFS(Universal Flash Storage) 장치, eMMC(embedded MMC) 장치 등과 같이 비휘발성 메모리에 데이터를 저장하는 장치 등을 포함할 수 있다. 비휘발성 메모리는 ROM(Read Only Memory), PROM(Programmable ROM), EPROM (Electrically Programmable ROM), EEPROM(Electrically Erasable and Programmable ROM), 플래시 메모리, PRAM(Phase-change RAM), MRAM(Magnetic RAM), RRAM(Resistive RAM), FRAM(Ferroelectric RAM) 등을 포함할 수 있다. 이뿐만 아니라, 메모리 시스템(100)은 다양한 형태의 저장 장치로 구현되어, 다양한 전자 기기 내에 탑재될 수 있다.
- [0218] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 또한, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이므로 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리 범위에 포함되는 것으로 해석되어야 할 것이다.

부호의 설명

- [0220] 100: 메모리 시스템 110: 메모리 장치
- 120: 메모리 컨트롤러 121: 호스트 인터페이스
- 122: 메모리 인터페이스 123: 제어 회로
- 124: 프로세서 125: 워킹 메모리
- 126: 에러 검출 및 정정 회로 210: 메모리 셀 어레이
- 220: 어드레스 디코더 230: 리드 앤 라이트 회로
- 240: 제어 로직 250: 전압 생성 회로

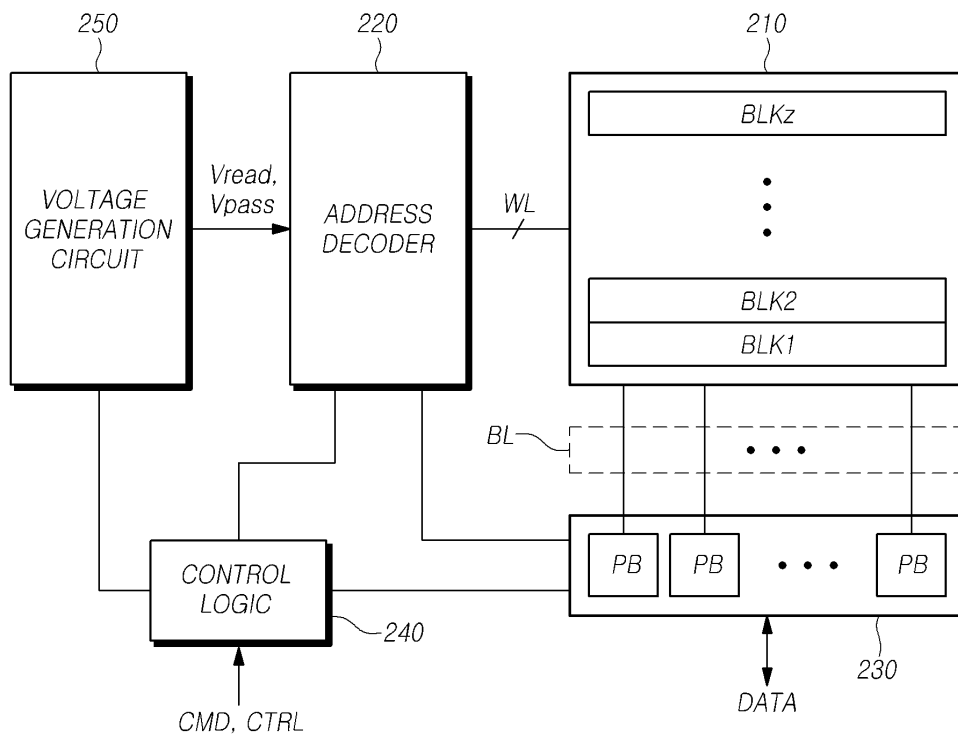
도면

도면1

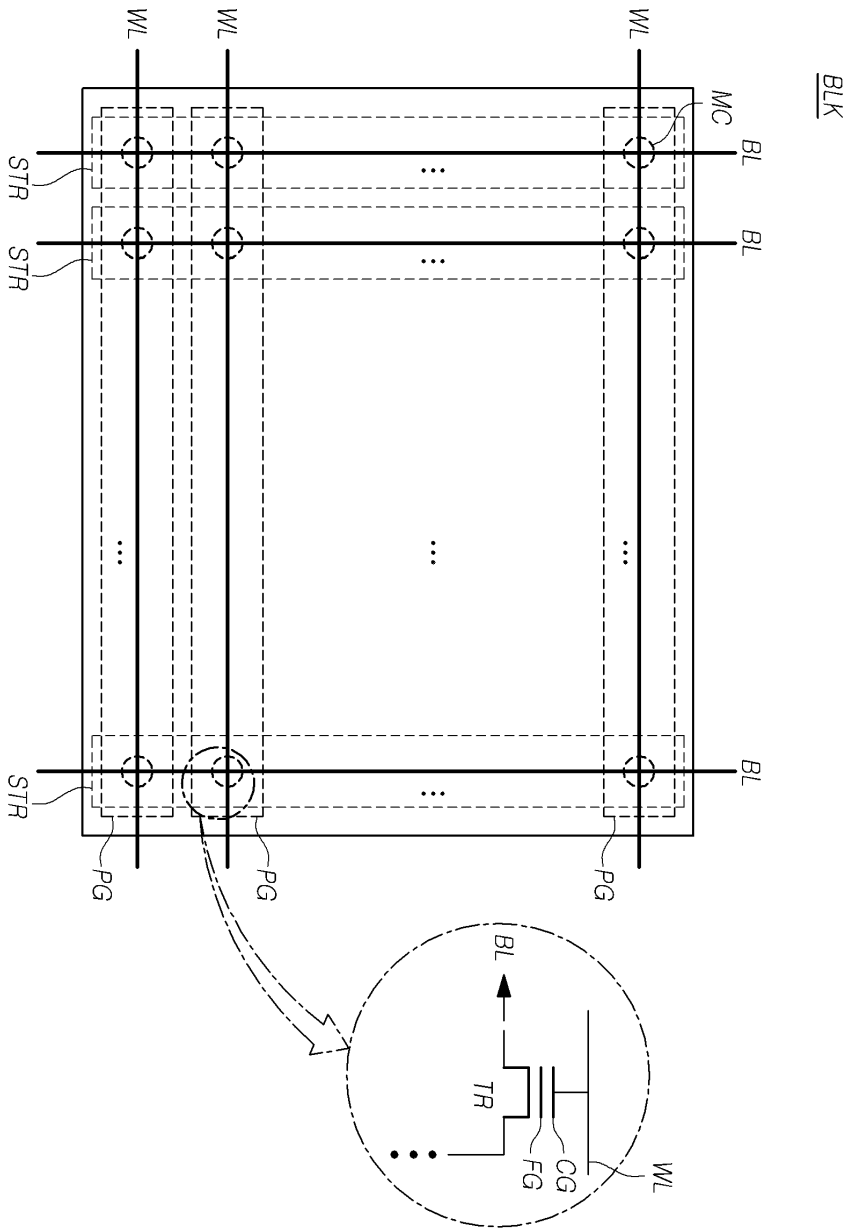


도면2

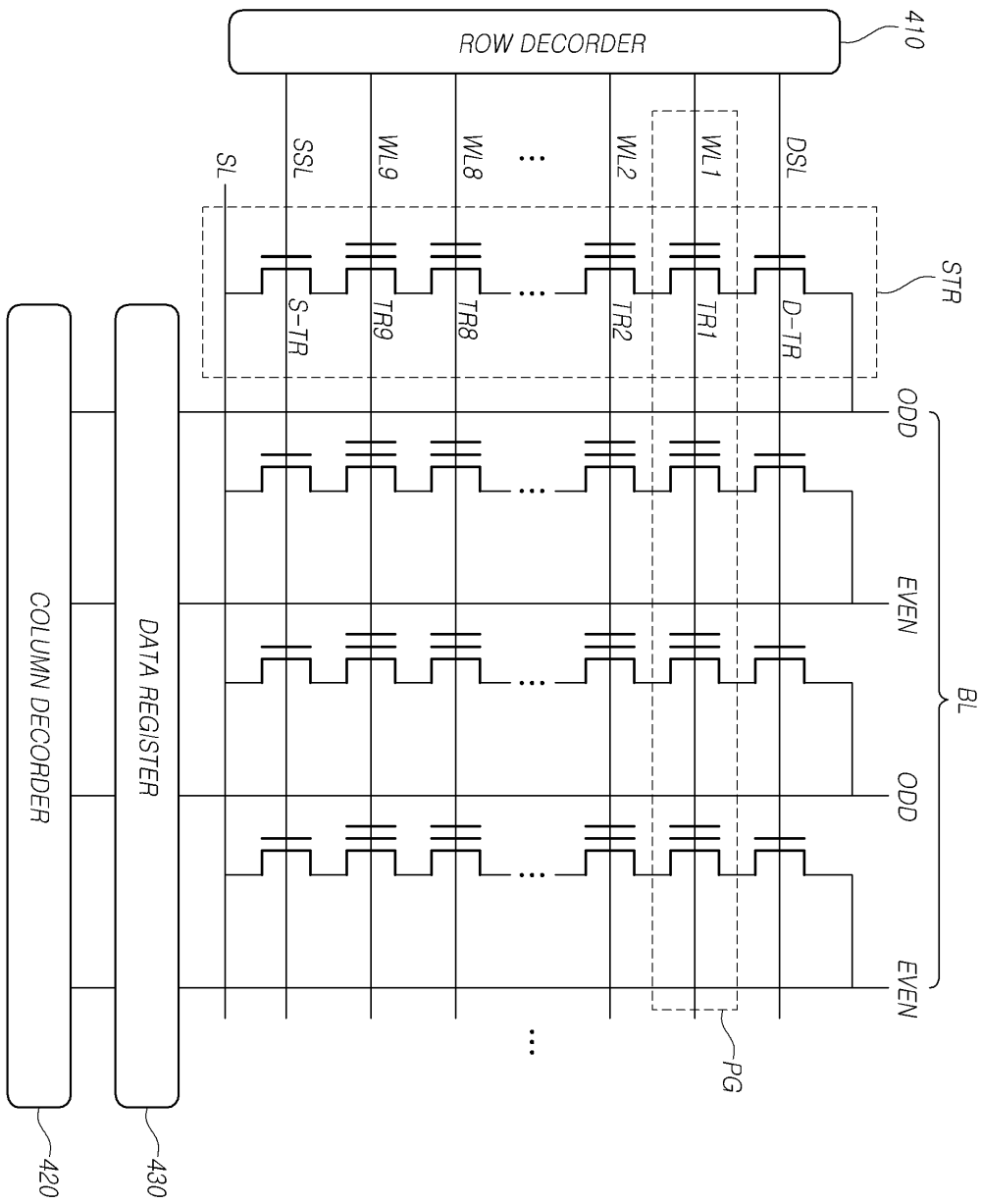
110



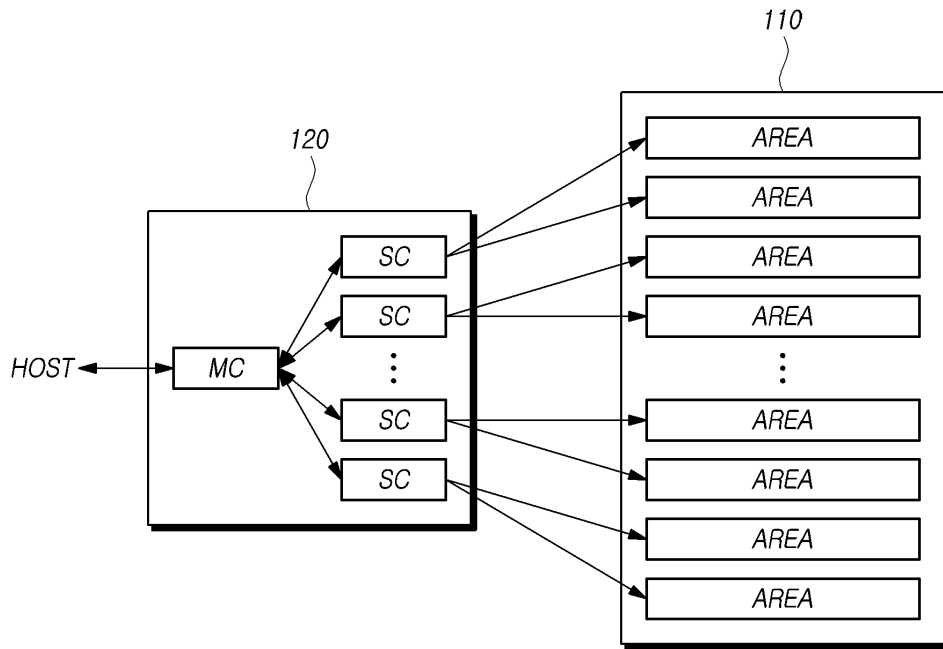
도면3



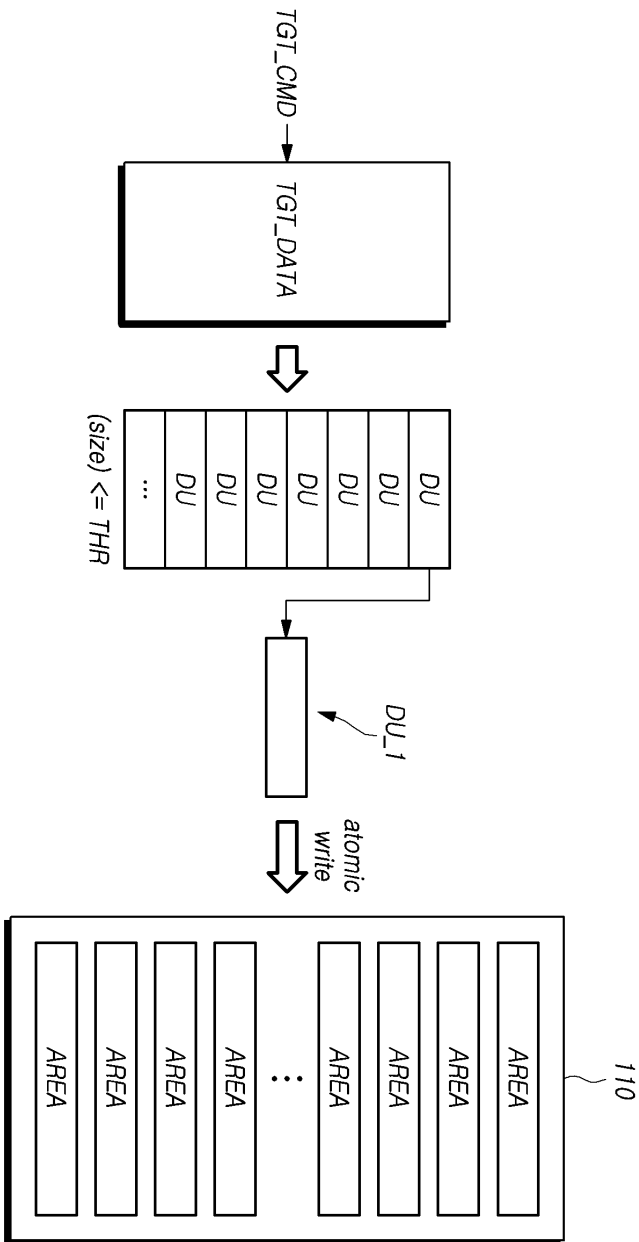
도면4



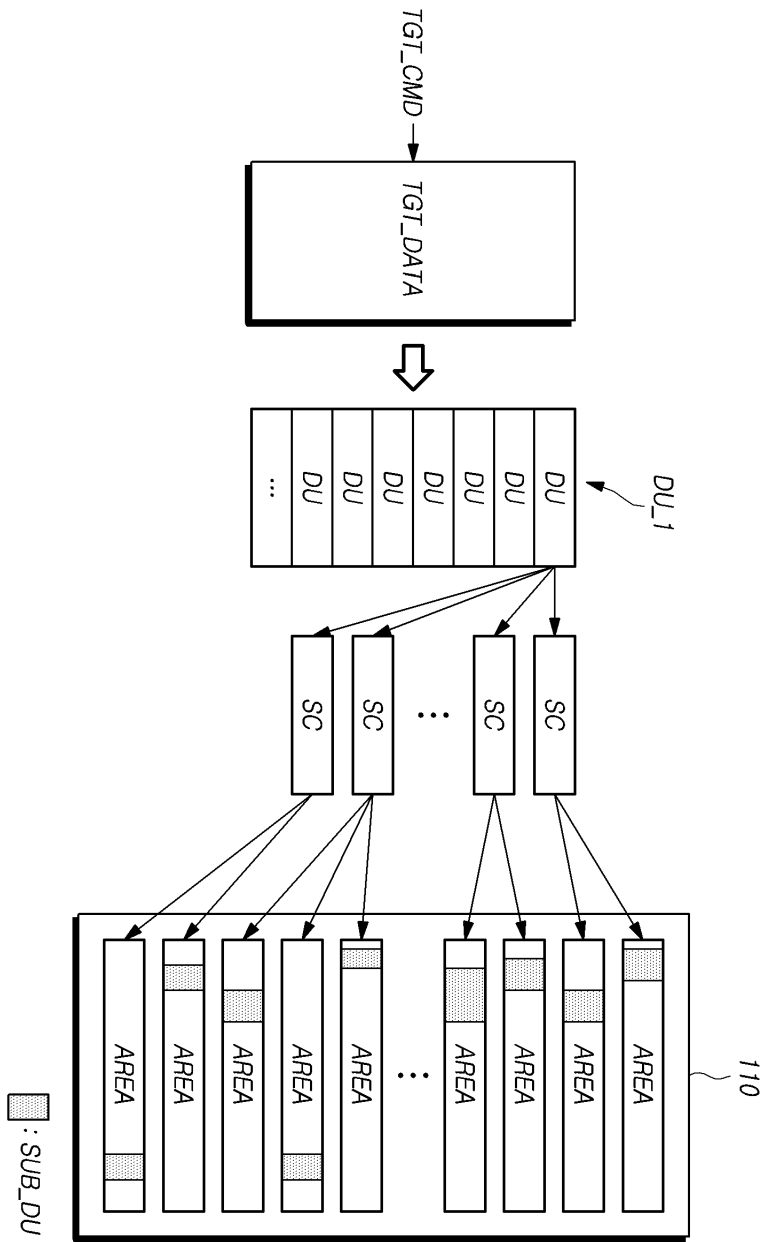
도면5



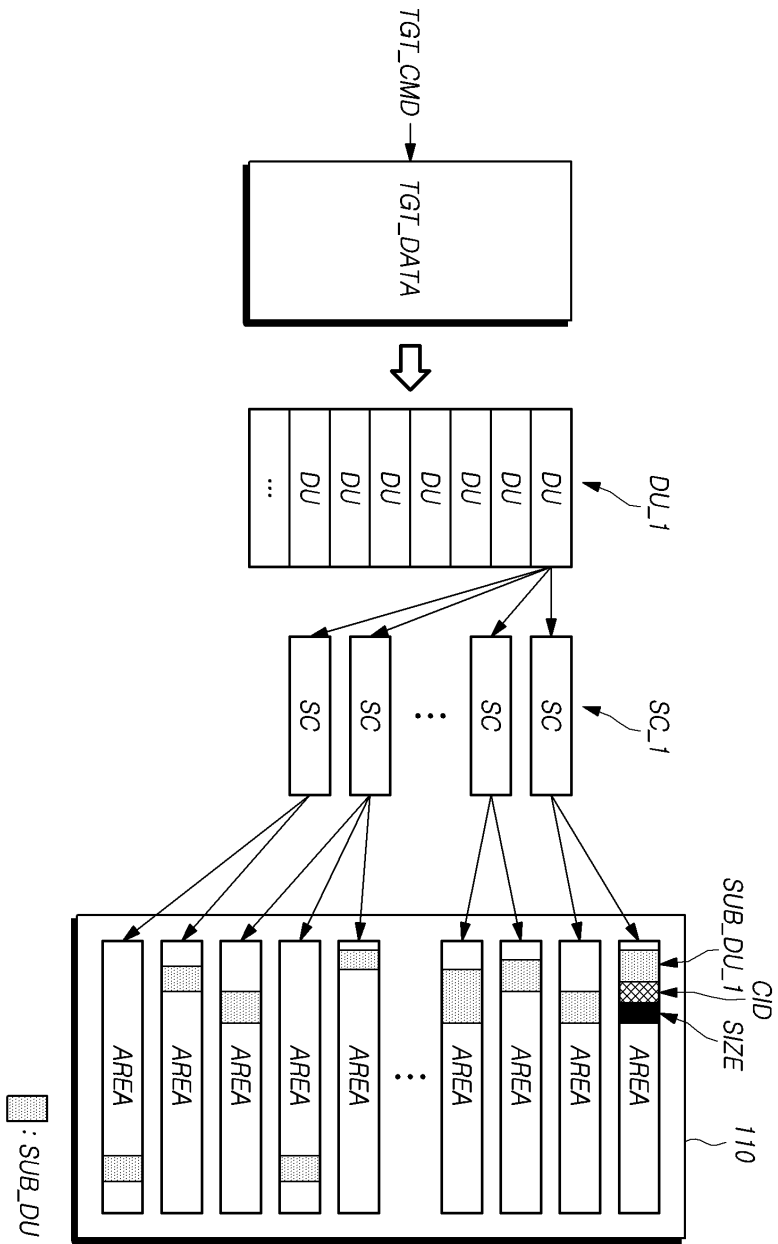
도면6



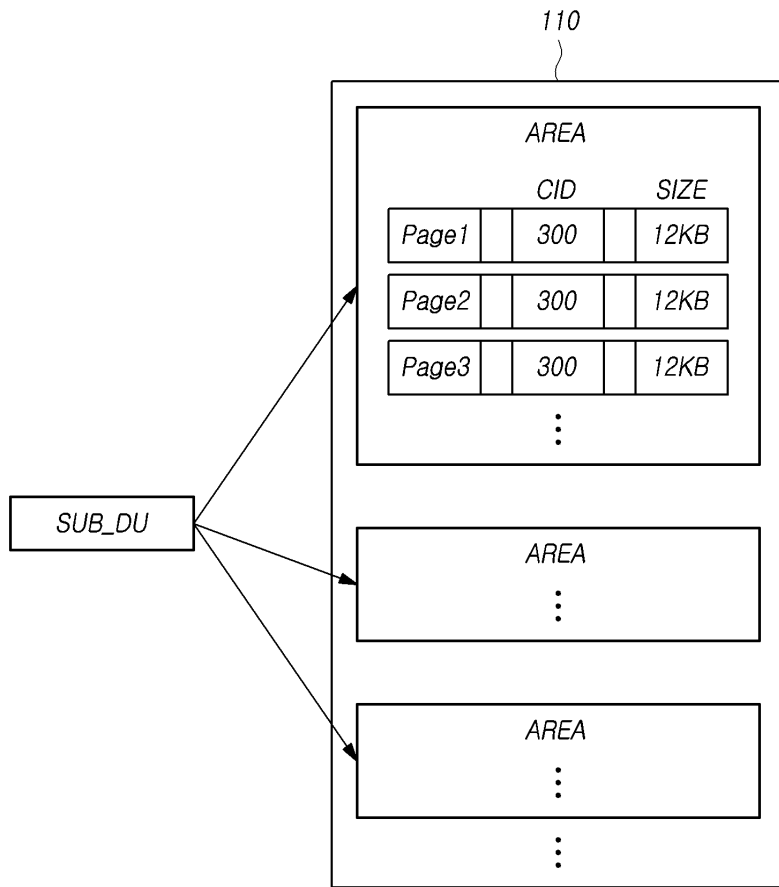
도면7



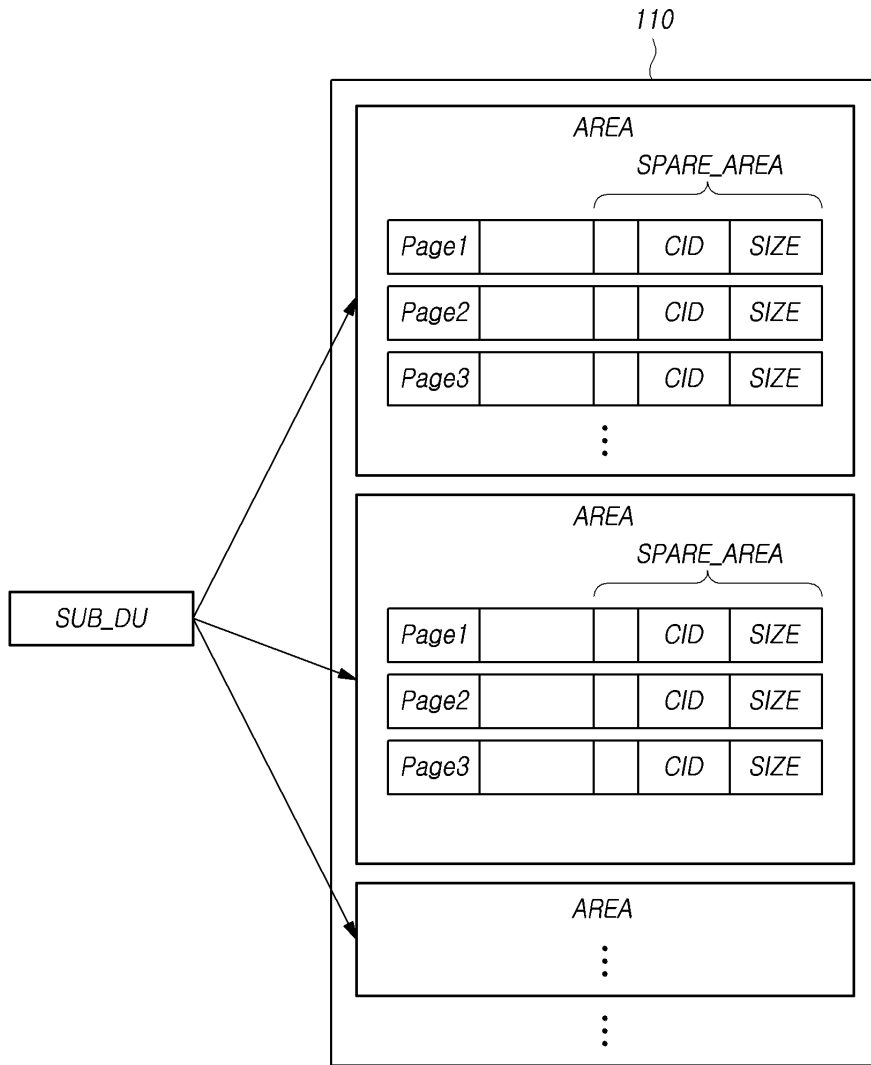
도면8



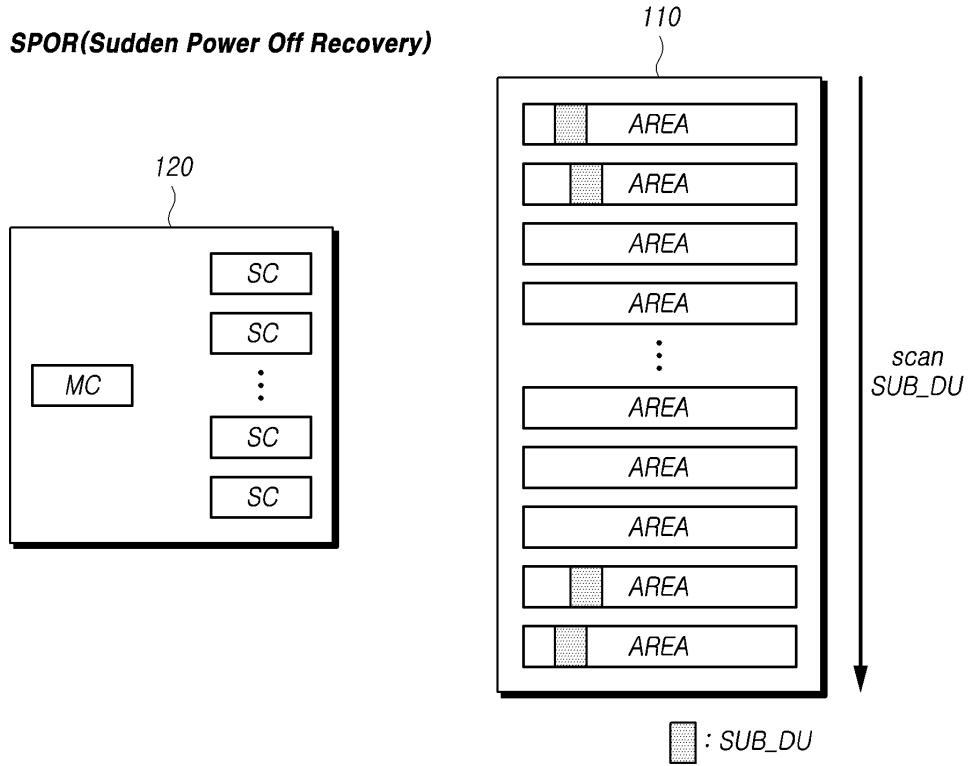
도면9



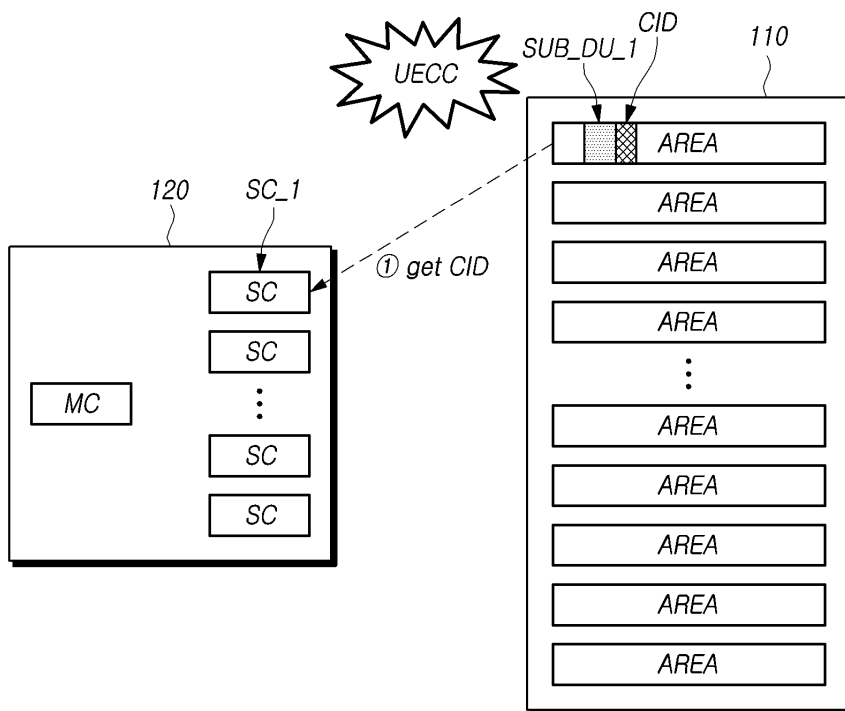
도면10



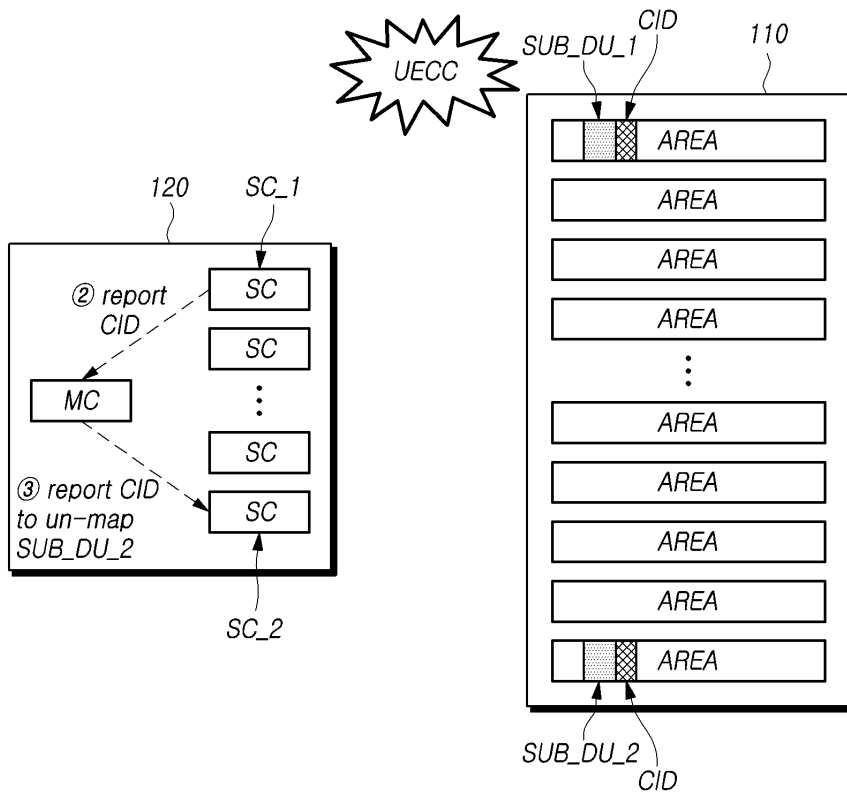
도면11



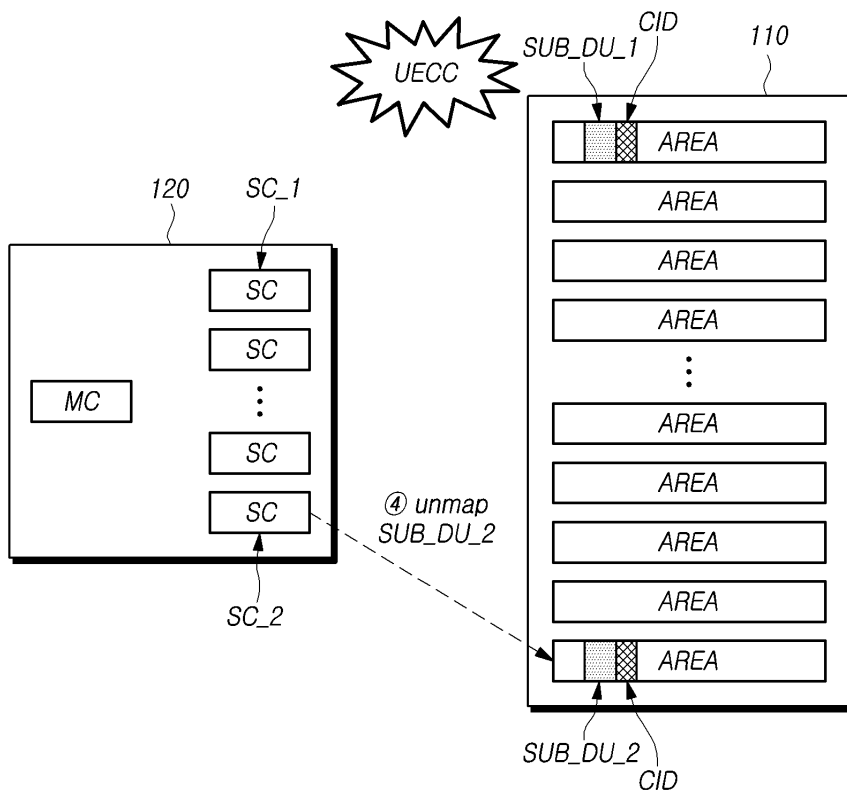
도면12



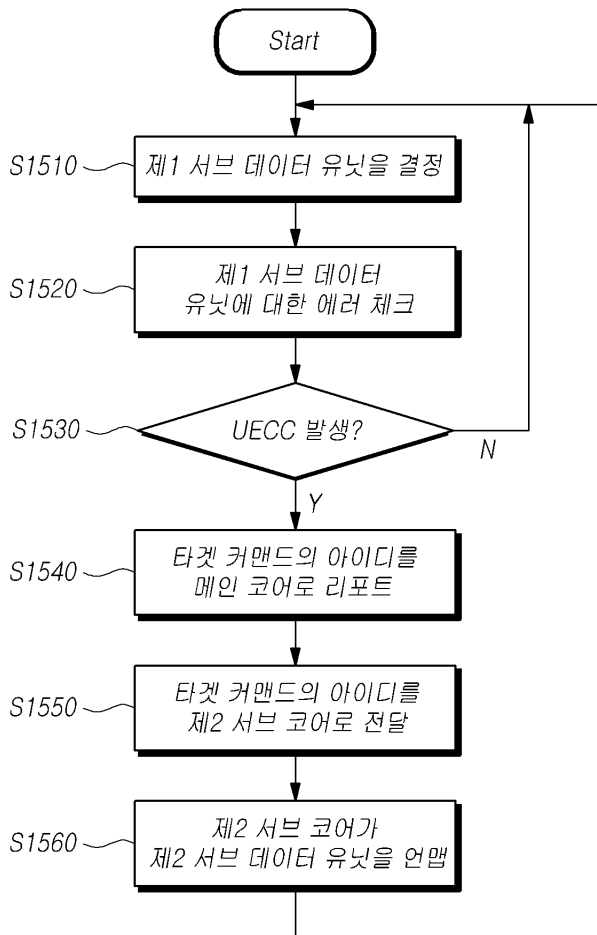
도면13



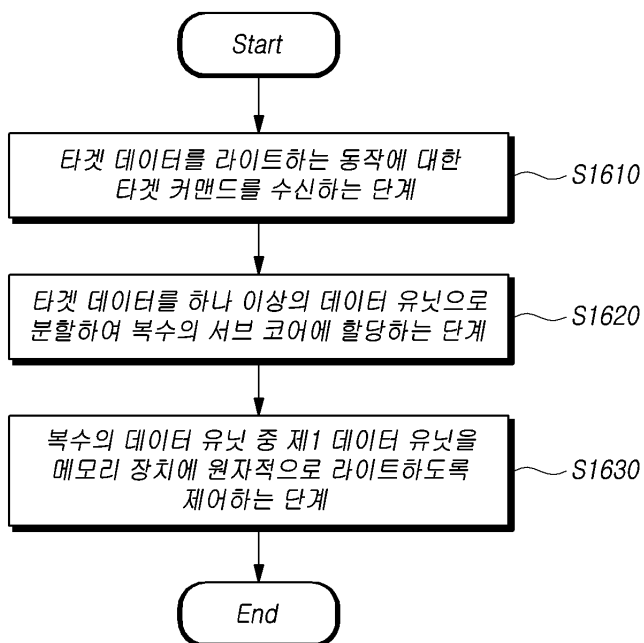
도면14



도면15



도면16



도면17

