(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0137839
(43) 공개일자 2014년12월03일

(51) 국제특허분류(Int. Cl.)
G06F 12/08 (2006.01) *G06F 11/08* (2006.01)
(21) 출원번호 10-2013-0058947
(22) 출원일자 2013년05월24일
심사청구일자 2013년05월24일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
김선욱
경기 남양주시 도농로 34, 214동 2003호 (도농동, 부영그린타운)
정동하
서울 성북구 인촌로24길 37, 303호 (안암동5가)
(74) 대리인
특허법인엠에이피에스

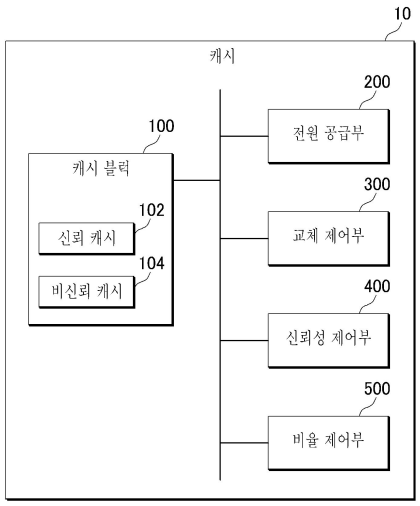
전체 청구항 수 : 총 17 항

(54) 발명의 명칭 **캐시 및 그 운영 방법**

(57) 요 약

본 발명은 캐시에 있어, 더티(dirty) 데이터를 저장하는 신뢰 캐시; 및 클린(clean) 데이터를 저장하는 비신뢰 캐시;를 포함하되, 상기 캐시는 상기 신뢰 캐시와 상기 비신뢰 캐시에 기지정된 비율로 캐시 영역을 할당하여 동작을 시작하며, 상기 비신뢰 캐시는 상기 신뢰 캐시보다 낮은 전압이 공급되는 캐시를 제공한다.

대 표 도 - 도2



특허청구의 범위

청구항 1

캐시에 있어서,

더티(dirty) 데이터를 저장하는 신뢰 캐시; 및

클린(clean) 데이터를 저장하는 비신뢰 캐시;를 포함하되,

상기 캐시는 상기 신뢰 캐시와 상기 비신뢰 캐시에 기지정된 비율로 캐시 영역을 할당하여 동작을 시작하며,

상기 비신뢰 캐시는 상기 신뢰 캐시보다 낮은 전압이 공급되는 캐시.

청구항 2

제 1 항에 있어서,

상기 신뢰 캐시에는 캐시 데이터의 신뢰성을 보장하는 최저 인가 전압이 공급되고,

상기 비신뢰 캐시에는 상기 최저 인가 전압보다 낮은 전압이 공급되는 캐시.

청구항 3

제 1 항에 있어서,

상기 신뢰 캐시에 저장된 데이터 또는 상기 더티 데이터에 대해서는 오류 정정 코드(ECC: error correcting code)를 사용하여 오류 발생 검출 및 오류 정정을 수행하고,

상기 비신뢰 캐시에 저장된 데이터 또는 상기 클린 데이터에 대해서는 오류 검출 코드(EDC: error detection code)를 사용하여 오류 발생을 검출하고, 오류 발생이 검출되면 다른 메모리에 중복 저장되어 있는 데이터를 대신 사용하는 캐시.

청구항 4

제 3 항에 있어서,

상기 캐시는 상기 오류 정정 코드와 상기 오류 검출 코드를 하나의 저장소를 공유하여 기록하는 캐시.

청구항 5

제 1 항에 있어서,

상기 캐시는 상기 비신뢰 캐시에 저장되어 있는 데이터에 오류가 발생하면 캐시 미스(cache miss)가 발생한 것으로 처리하는 캐시.

청구항 6

제 1 항에 있어서,

상기 캐시는 상기 신뢰 캐시와 상기 비신뢰 캐시에 대해 단일 교체 정책(replacement policy)을 사용하여 교체 희생자를 찾는 캐시.

청구항 7

제 1 항에 있어서,

상기 캐시는 상기 신뢰 캐시와 상기 비신뢰 캐시에 대해 하나의 듀얼 PLRU 트리(Dual PLRU tree)를 사용하여 교체 희생자를 찾는 캐시.

청구항 8

제 1 항에 있어서,

상기 캐시 영역은 웨이(way)별로 할당되는 캐시.

청구항 9

제 1 항에 있어서,

상기 캐시는 상기 신뢰 캐시와 상기 비신뢰 캐시의 비율을 동적으로 조정하는 캐시.

청구항 10

제 1 항에 있어서,

상기 비신뢰 캐시에 상기 신뢰 캐시보다 많은 캐시 영역을 할당하여 동작을 시작하는 캐시.

청구항 11

제 1 항에 있어서,

상기 캐시는 교체 희생자가 발생할 때마다 신뢰 캐시 교체 희생자 수 또는 비신뢰 캐시 교체 희생자 수를 증가시키며,

상기 캐시 교체 희생자 수 또는 상기 비신뢰 캐시 교체 희생자 수의 증가율이 임계값 이상이 되는 시점마다, 상기 신뢰 캐시 교체 희생자 수 또는 상기 비신뢰 캐시 교체 희생자 수에 기초하여, 상기 신뢰 캐시 및 상기 비신뢰 캐시의 비율을 조정하는 캐시.

청구항 12

캐시 운영 방법에 있어서,

기지정된 비율로 신뢰 캐시와 비신뢰 캐시에 캐시 영역을 할당하는 단계; 및

상기 신뢰 캐시에는 더티(dirty) 데이터를 저장하고, 상기 비신뢰 캐시에는 클린(clean) 데이터를 저장하는 단계;를 포함하되,

상기 비신뢰 캐시는 상기 신뢰 캐시보다 낮은 전압이 공급되는 캐시 운영 방법.

청구항 13

제 12 항에 있어서,

상기 신뢰 캐시에는 캐시 데이터의 신뢰성을 보장하는 최저 인가 전압이 공급되고, 상기 비신뢰 캐시에는 상기 최저 인가 전압보다 낮은 전압이 공급되는 캐시 운영 방법.

청구항 14

제 12 항에 있어서,

상기 방법은

상기 신뢰 캐시에 저장된 데이터 또는 상기 더티 데이터에 대해 오류 정정 코드(ECC: error correcting code)를 사용하여 오류 발생 검출 및 오류 정정을 수행하는 단계; 및

상기 비신뢰 캐시에 저장된 데이터 또는 상기 클린 데이터에 대해 오류 검출 코드(EDC: error detection code)를 사용하여 오류 발생을 검출하고, 오류 발생이 검출되면 다른 메모리에 중복 저장되어 있는 데이터를 대신 사용하는 단계;를 포함하되,

상기 오류 정정 코드 및 상기 오류 검출 코드는 하나의 저장소를 공유하여 기록되는 캐시 운영 방법.

청구항 15

제 12 항에 있어서,

상기 방법은

상기 신뢰 캐시와 상기 비신뢰 캐시에 대해 하나의 듀얼 PLRU 트리(Dual PLRU tree)를 사용하여 교체 희생자를

찾는 단계; 및

교체 희생자가 발생할 때마다 신뢰 캐시 교체 희생자 수 또는 비신뢰 캐시 교체 희생자 수를 증가시키는 단계;를 포함하는 캐시 운영 방법.

청구항 16

제 12 항에 있어서,

상기 할당 단계는

상기 비신뢰 캐시에 상기 신뢰 캐시보다 많은 캐시 영역을 할당하는 캐시 운영 방법.

청구항 17

제 15 항에 있어서,

상기 방법은

상기 신뢰 캐시 및 상기 비신뢰 캐시의 비율을 조정하는 단계;를 더 포함하되,

상기 비율 조정 단계는

상기 캐시 교체 희생자 수 또는 상기 비신뢰 캐시 교체 희생자 수의 증가율이 임계값 이상이 되는 시점마다 수행되는 캐시 운영 방법.

명 세 서

기 술 분 야

[0001] 본 발명은 캐시 및 그 운영 방법에 관한 것이다.

배 경 기 술

[0002] 최근의 고성능 프로세싱 장치는 도 1에 도시되어 있는 바와 같이, 여러 개의 코어(core)를 채용할 뿐 아니라, 여러 계층의 대용량 캐시(cache)를 구비하고 일부를 각 코어가 공유하는 추세로 발전하고 있다.

[0003] 도 1은 본 발명의 일실시예에 따른 캐시를 포함하는 프로세싱 장치의 실시예를 도시하고 있다.

[0004] 첫번째 실시예는 오라클의 울트라스팍T3(Oracle Ultasparc-T3)이고, 두번째 실시예는 인텔의 아이비 브릿지(Intel Ivy Bridge)이다. 이들 실시예들은 각 코어 당 할당된 4개의 L2 캐시를 명시적 또는 암시적으로 도시하고 있으며, 1개의 공유된 L3 캐시를 명시적으로 도시하고 있다.

[0005] 도면에서 볼 수 있는 바와 같이, 이러한 대용량 다계층 캐시는 프로세싱 장치 면적의 30~35%이상을 차지하는 것이 일반적이다. 면적이 큰 만큼 소모하는 전력의 비중도 높다.

[0006] 전압 조정으로 캐시가 소모하는 전력을 낮출 수 있으나, 공급하는 전압을 지나치게 낮추면 데이터 신뢰도 문제가 발생한다. 예를 들어, 최근 캐시에 많이 사용되는 SRAM은 저장된 데이터가 공정 차이(process variation), 입자 충돌(particle strike) 등의 이유로 데이터를 쓰거나 유지하거나 읽는 과정에서 값이 변이될 수 있다는 데이터 신뢰도 문제를 갖는다. 특히 공정 차이는 전압에 따라 큰 신뢰성 차이를 보인다. 하나의 SRAM 비트(bit)가 값을 잃을 확률은 입력 전압의 지수 함수에 비례하여 증가하며, 어느 정도 이상의 값이 변하게 되면 해당 캐시의 값을 신뢰하여 사용할 수 없다.

[0007] 따라서, 캐시의 데이터 신뢰성을 보장하기 위해서는 캐시의 신뢰성을 보장할 수 있는 최저 인가 전압(Vcc_MIN) 이상의 전압을 캐시에 공급하거나, 오류 발생시 정정할 수 있는 방법이 필요하다. 그러나 종래의 오류 정정 방법은 공간 및 시간 자원, 그리고 오류 정정에 드는 추가적인 전력 자원의 오버헤드로 인해 비효율적이라는 문제가 있다.

[0008] 프로세싱 장치에 포함되는 캐시는 전체 시스템 성능에 큰 영향을 미치기 때문에, 캐시의 신뢰도 유지는 매우 중요하다. 그러므로, 캐시 메모리에 저장되는 데이터의 신뢰성을 유지하면서도 캐시가 소모하는 전력을 낮출 수 있는 방법이 필요하다.

[0009] 이와 관련하여 한국등록특허 제10-02540800000호("마이크로프로세서의 파워 에스티메이터")에는 복수의 메모리를 구비한 마이크로프로세서에 있어서의 명령 레벨에서 소비 전력의 평가 정밀도를 향상시키는 구성이 개시되어 있다.

[0010] 또한, 한국등록특허 제10-07949730000호("캐시 메모리 시스템")는 통상 상태와 저소비 전력 상태와의 전환이 가능한 태그 메모리 RAM부 및 캐시 메모리 RAM부가 개시되어 있다.

발명의 내용

해결하려는 과제

[0011] 본 발명은 전술한 문제를 해결하기 위한 것으로서, 그 목적은 데이터 신뢰성을 보장하면서도 소모 전력이 낮은 캐시 및 그 운영 방법을 제공하는 것이다.

과제의 해결 수단

[0012] 상기와 같은 목적을 달성하기 위한 본 발명의 제 1 측면에 따른 캐시는 더티(dirty) 데이터를 저장하는 신뢰 캐시; 및 클린(clean) 데이터를 저장하는 비신뢰 캐시;를 포함하되, 상기 캐시는 상기 신뢰 캐시와 상기 비신뢰 캐시에 기지정된 비율로 캐시 영역을 할당하여 동작을 시작하며, 상기 비신뢰 캐시는 상기 신뢰 캐시보다 낮은 전압이 공급되는 것을 특징으로 한다.

[0013] 상기와 같은 목적을 달성하기 위한 본 발명의 제 2 측면에 따른 캐시 운영 방법은 기지정된 비율로 신뢰 캐시와 비신뢰 캐시에 캐시 영역을 할당하는 단계; 및 상기 신뢰 캐시에는 더티(dirty) 데이터를 저장하고, 상기 비신뢰 캐시에는 클린(clean) 데이터를 저장하는 단계;를 포함하되, 상기 비신뢰 캐시는 상기 신뢰 캐시보다 낮은 전압이 공급되는 것을 특징으로 한다.

발명의 효과

[0014] 본 발명은 캐시 및 방법에 있어, 데이터 신뢰성을 보장하면서도 소모 전력을 낮추는 효과를 얻는다.

[0015] 본 발명의 일실시예에 따른 캐시는 수직적 메모리 계층의 데이터 중복 및 멀티 코어 상의 수평적 데이터 중복을 활용하여, 캐시 데이터의 신뢰성을 보장하면서도 캐시의 소모 전력을 줄인다는 발명의 목적을 달성한다. 즉, 클린 데이터의 경우, 오류 발생시 해당 데이터를 중복 저장하고 있는 다른 메모리에서 해당 데이터를 읽어와 복구할 수 있으므로, 클린 데이터가 저장되는 캐시 영역의 전압을 낮추어도 데이터의 신뢰성을 보장할 수 있다는 특징을 활용한다.

[0016] 본 발명의 일실시예에 따른 캐시는 이렇게 캐시 메모리를 두 그룹으로 나누어 상태에 따라 구분하여 데이터를 저장하고, 클린 데이터를 담은 캐시의 전압을 기존의 최저 공급 전압 이하로 낮춤으로써, 시스템에서의 전력 소모 비중이 큰 캐시의 동적/정적 공급 전압을 크게 낮출 수 있다.

도면의 간단한 설명

[0017] 도 1은 본 발명의 일실시예에 따른 캐시를 포함하는 프로세싱 장치의 실시예를 도시함.

도 2는 본 발명의 일실시예에 따른 캐시의 구조를 도시함.

도 3은 본 발명의 일실시예에 따른 캐시의 상세한 구조를 도시함.

도 4는 본 발명의 일실시예에 따른 전원 공급부를 도시함.

도 5는 종래의 PLRU 트리의 실시예를 도시함.

도 6은 본 발명의 일실시예에 따른 듀얼 PLRU 트리의 실시예를 도시함.

도 7은 본 발명의 일실시예에 따른 신뢰성 제어부를 도시함.

도 8 내지 도 10은 본 발명의 일실시예에 따른 캐시 비율 조정 중, 신뢰 캐시를 비신뢰 캐시로 변경하는 과정에서 더티 블록을 처리하는 방법을 도시함.

도 11은 본 발명의 일실시예에 따른 캐시 운영 방법의 흐름을 도시함.

도 12는 본 발명의 일실시예에 따른 캐시 운영 방법의 성능을 도시함.

발명을 실시하기 위한 구체적인 내용

[0018] 아래에서는 첨부한 도면을 참조하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예를 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0019] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.

[0020] 앞에서 살펴본 바와 같이, 도 1은 본 발명의 일실시예에 따른 캐시를 포함하는 프로세싱 장치의 실시예를 도시하고 있다.

[0021] 본 발명의 일실시예에 따른 캐시(10)는 다른 캐시(10) 또는 하위 메모리와 데이터를 중복 저장할 수 있다. 도 1에 도시되어 있는 실시예를 참조하면, 본 발명의 일실시예에 따른 캐시(10)를 사용하는 프로세싱 장치는 계층적인 메모리 체계 및 멀티 코어 구조를 사용한다. 또한 이러한 프로세싱 장치는 여러 프로세싱 장치가 사용되는 멀티 프로세서 시스템으로 구성될 수 있다. 이러한 구조로 인해, 본 발명의 일실시예에 따른 캐시(10)가 사용되는 시스템에서는 데이터 중복이 존재하게 된다.

[0022] 예를 들어, 프로세싱 장치에 의해 값이 변경된 데이터를 더티(dirty) 데이터, 값이 변경되지 않은 데이터를 클린(clean) 데이터라고 할 때, 클린 데이터의 경우에는 메모리 계층에서 하위 계층은 상위 계층이 가진 데이터를 항상 가지고 있다. 예를 들어, 메인 메모리는 캐시에 위치한 모든 클린 데이터를 중복해서 가지고 있다.

[0023] 또한, 멀티 코어 또는 멀티 프로세서 시스템으로 구성된 실시예에서는 같은 데이터를 여러 코어 또는 여러 프로세서가 공유할 수 있으며, 공유된 데이터는 각 코어 또는 각 프로세싱 장치의 캐시에 중복 존재한다.

[0024] 본 발명의 일실시예에 따른 캐시(10)는 이러한 수직적 메모리 계층의 데이터 중복 및 멀티 코어 상의 수평적 데이터 중복을 활용하여, 캐시 데이터의 신뢰성을 보장하면서도 캐시의 소모 전력을 줄인다는 발명의 목적을 달성한다. 즉, 클린 데이터의 경우, 오류 발생시 다른 메모리(예: 하위 계층의 메모리 또는 다른 코어의 캐시)에서 해당 데이터를 읽어 들여 복구할 수 있으므로, 클린 데이터가 저장되는 캐시 영역의 전압을 낮추어도, 데이터의 신뢰성을 보장할 수 있다.

[0025] 종래의 기술은 캐시 데이터의 신뢰성을 보장하기 위해, 캐시에 오류 정정 코드를 추가하거나 중복 데이터를 포함하는 등 캐시 자체의 신뢰성을 향상시키는데 초점을 맞추어왔다. 이에 반해, 본발명은 캐시 자체의 신뢰성을 향상시키는 것보다는 낮은 신뢰성으로 인해 문제가 발생하더라도 효율적으로 이를 해결하는 것에 집중한다. 즉, 본발명은 공급 전압을 낮추어 발생하는 데이터 신뢰성 저하를 데이터 중복을 활용해 해결함으로써, 캐시의 소모 전력을 감소시킨다.

[0026] 즉, 본 발명의 일실시예에 따른 캐시(10)는 클린 데이터에 대해서는 오류 정정 기술보다 오버헤드가 낮은 오류 검출 기술을 사용함으로써, 캐시 신뢰성 제공을 위한 종래 기술이 갖고 있던 문제를 해결하였다. 예를 들어, 종래의 캐시가 채용하는 오류 정정 기술은 오류가 빈번하게 발생하지 않음에도 다중 오류 보호에 필요한 오버헤드로 인해 최저 인가 전압이 높은 수준으로 유지되어야 하는 경우가 많았다. 본 발명의 일실시예에 따른 캐시(10)는 오류 정정 기술만을 사용하였던 종래 캐시의 높은 보호 수준에 따른 추가 부담이 급격하게 증가하고, 전압을 일정 수준 이하로 낮추는 것이 불가능했던 이러한 문제를 오류 정정 기술보다 오버헤드가 낮은 오류 검출 기술을 함께 사용하여 해결하였다.

[0027] 도 2 및 도 3은 본 발명의 일실시예에 따른 캐시(10)의 구조를 도시하고 있다.

[0028] 본 발명의 일실시예에 따른 캐시(10)는 더티 데이터를 저장하는 캐시 영역(이하 신뢰 캐시: 102)과 클린 데이터를 저장하는 캐시 영역(이하 비신뢰 캐시: 104)으로 구분되어 운영되는 캐시 블럭(100)을 포함한다. 또한 캐시(10)는 캐시 블럭(100)의 제어를 위해, 전원 공급부(200), 교체 제어부(300), 신뢰성 제어부(400), 및 비율 제어부(500)를 포함할 수 있다.

[0029] 본명세서에서 캐시 블럭(100)은 캐시 메모리, 즉, 캐시 데이터를 저장하는 저장소라는 의미로 쓰였다. 일실시예

에서 캐시 블럭(100)은 SRAM을 사용하여 구현될 수 있으나, 캐시 블럭(100)을 구현하는 메모리 종류에는 제한이 없다.

[0030] 또한 본명세서에서 캐시 블럭(100) 이외의 '블럭'은 캐시 메모리의 데이터 단위 또는 해당 블럭에 저장되어 있거나 저장될 데이터를 지칭한다. 캐시 데이터의 클린 및 더티 여부는 캐시 데이터에 존재하는 더티 비트로 판별할 수 있다.

[0031] 캐시 블럭(100)은 여러 개의 웨이(way)를 포함하도록 구성될 수 있다. 일실시예에서, 신뢰 캐시(102)와 비신뢰 캐시(104)는 웨이별로 할당될 수 있다. 예를 들어, 8-웨이 캐시의 경우 4개의 웨이는 신뢰 캐시(102)로, 4개의 웨이는 비신뢰 캐시(104)로 구분되어 운영될 수 있다. 단, 이는 예시를 위한 것으로, 캐시 블럭(100)의 구성은 여기에 한정되지 않는다.

[0032] 바람직한 실시예에서, 비신뢰 캐시(104)는 신뢰 캐시(102)보다 낮은 전압을 공급받을 수 있다. 예를 들어, 전원 공급부(200)는 신뢰 캐시(102)에는 캐시 데이터의 신뢰성을 보장할 수 있는 최저 인가 전압(Vcc_MIN)을 공급하고, 비신뢰 캐시(104)에는 최저 인가 전압보다 낮은 전압을 공급할 수 있다.

[0033] 또다른 실시예에서, 비신뢰 캐시(104)는 신뢰 캐시(102)보다 작은 메모리 셀로 구성될 수 있다. SRAM은 셀의 크기에 따라 신뢰성을 조정할 수 있다. 작은 셀의 경우 상대적으로 적은 면적이 필요하지만 신뢰성이 떨어지고 큰 셀의 경우 큰 면적이 필요하지만 더 안정적이다. 이와 같은 이종 SRAM 셀들과 선택적 전압 인가를 통하여 차등적 신뢰성을 가지는 캐시(10)를 구현할 수 있으며, 이는 전력뿐만 아니라 면적까지 줄일 수 있는 효과가 있다.

[0034] 본 발명의 일실시예에 따른 캐시(10)를 운영할 때, 더티 데이터는 프로세싱 장치에서 값이 변경된 데이터이기 때문에 중복된 데이터가 다른 메모리에 존재하지 않으므로, 캐시 데이터의 신뢰성을 보장할 수 있는 최저 인가 전압을 공급받는 신뢰 캐시(102)에 저장되어야 한다. 그러나 클린 데이터는 전술한 바와 같이 하위 메모리 또는 다른 코어의 캐시 등 중복 데이터를 저장하고 있는 다른 메모리가 있으므로, 데이터의 값이 변경되는 등의 오류가 발생하여도 쉽게 복구가 가능하므로, 캐시 데이터의 신뢰성을 보장할 수 있는 최저 인가 전압보다 낮은 전압을 공급받는 비신뢰 캐시(104)에 저장하여도 다른 메모리로부터 데이터를 읽어와 복원함으로써 캐시 데이터 신뢰성을 용이하게 제공할 수 있다.

[0035] 예를 들어, 본 발명의 일실시예에 따른 캐시(10)는 클린 데이터를 저장한 캐시 웨이들, 즉, 비신뢰 캐시(104)에 기존의 최저 공급 전압보다 더 낮은 전압을 공급함에 따라 발생하는 데이터 손상을 단순히 캐시 미스(cache miss)가 발생한 것으로 간주할 수 있다.

[0036] 이렇게 비신뢰 캐시(104)에 저장되어 있는 데이터에 오류 발생시 캐시 미스가 발생한 것으로 처리하면, 일반 캐시 미스와 동일하게 하위 메모리(예: 메인 메모리 또는 하위 계층의 캐시)에서 데이터를 가져옴으로써 해당 데이터를 복구할 수 있으므로, 본 발명의 일실시예에 따른 캐시(10)는 데이터 복원을 위해 별도의 동작을 필요로 하지 않아, 캐시 데이터 신뢰성을 효율적으로 제공할 수 있다.

[0037] 따라서, 바람직한 실시예에서, 신뢰 캐시(102)에 저장되어 있는 데이터는 오류 정정 코드(ECC: error correcting code)를 사용하여 오류 발생이 검출되고 오류가 정정되며, 비신뢰 캐시(104)에 저장되어 있는 데이터는 오류 발생시 오류 검출 코드(EDC: error detection code)를 사용하여 오류 발생이 검출되고, 오류 발생이 검출되면 다른 메모리에서 가져온 올바른 데이터가 대신 사용된다. 예를 들어, 하위 계층의 메모리 또는 다른 코어의 캐시에 중복 저장되어 있는 데이터가 대신 사용될 수 있다. 이때 하위 계층의 메모리에 저장되어 있는 중복 데이터를 대신 사용하는 것은 전술한 바와 같이 캐시 미스가 발생한 것으로 처리하여 간단히 수행될 수 있다.

[0038] 이때, 본 발명의 일실시예에 따른 캐시(10)는 오류 정정 코드 및 오류 검출 코드를 하나의 저장소를 공유하여 기록하고, 캐시(10)에 저장되어 있는 데이터의 상태에 따라 오류 정정 코드를 사용할 것인지 오류 검출 코드를 사용할 것인지 결정할 수 있다.

[0039] 이렇게 하나의 저장소를 공유하여 기록하므로, 본 발명의 일실시예에 따른 캐시(10)는 종래 캐시의 오류 정정 코드를 사용하기 위한 회로를 크게 변경하지 않고도 오류 검출 코드를 함께 사용할 수 있으며, 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율이 달라질 때도 하나의 공간이 오류 정정 코드를 기록하기도 하고 오류 검출 코드를 기록하기도 하므로 문제없이 운영할 수 있어 유연성이 좋다.

[0040] 한편, 본 발명의 일실시예에 따른 캐시(10)는 신뢰 캐시(102)와 비신뢰 캐시(104)에 대해 통합 교체 정책을 사용하여, 하나의 듀얼 PLRU 트리(Dual PLRU tree)를 사용하여 교체 희생자를 찾을 수 있다. 자세한 내용은 후술

한다.

[0041] 또한, 본 발명의 일실시예에 따른 캐시(10)는 교체 희생자가 발생할 때마다 신뢰 캐시 교체 희생자 수 또는 비신뢰 캐시 교체 희생자 수를 증가시키고, 캐시 교체 희생자 수 및 비신뢰 캐시 교체 희생자 수의 증가율이 임계값 이상이 되는 시점마다, 신뢰 캐시 교체 희생자 수 또는 비신뢰 캐시 교체 희생자 수에 기초하여, 신뢰 캐시(102) 및 비신뢰 캐시(104)의 비율을 동적으로 조정할 수 있다.

[0042] 예를 들어, 본 발명의 일실시예에 따른 캐시(10)는 비신뢰 캐시(104)에 신뢰 캐시(102)보다 많은 캐시 영역을 할당하여 동작을 시작한 후, 신뢰 캐시 교체 희생자 수가 소정의 임계값을 넘어설 때마다 신뢰 캐시(102)를 증가시킬 수 있다. 자세한 내용은 후술한다.

[0043] 도 3은 본 발명의 일실시예에 따른 캐시의 상세한 구조를 도시하고 있다.

[0044] 입력 블럭은 상태(M, O, S, E, I)에 따라, 캐시 블럭(100)의 신뢰 캐시(102)와 비신뢰 캐시(104)로 구분되어 저장된다. 전술한 바와 같이, 캐시 블럭(100)은 SRAM으로 구현될 수 있으며, 신뢰 캐시(102)와 비신뢰 캐시(104)는 웨이 별로 영역이 할당될 수 있다.

[0045] 입력 블럭의 상태에는 예를 들어, 다음과 같이 것들이 있을 수 있다.

[0046] * M(modified state): 캐시 블럭이 더티(dirty)이면서 유일(sole)하여, 손실되어서는 안 되는 상태. 이 상태의 데이터는 신뢰 캐시(102)에 저장되어야 한다.

[0047] * O(owned state): 중복 데이터가 존재하지만, 라이트백(write-back)이 필요한 상태. 이 상태의 데이터는 신뢰 캐시(102)에 저장되어야 한다.

[0048] * E(exclusive state): 메인 메모리에 중복 데이터가 존재하므로, 비신뢰 캐시(104)에 저장될 수 있다. 그러나, 메인 메모리에 접근해야 한다는 비용이 있다.

[0049] * S(shared state): 다른 캐시 또는 메인 메모리에 중복 데이터가 존재하므로, 비신뢰 캐시(104)에 저장될 수 있다.

[0050] * I(invalid state): 캐시 블럭이 더티(dirty)였을 수도 있고 클린(clean)이었을 수도 있으나, 현재는 유효하지 않으므로, 버려도 된다.

[0051] * F(faulty state): 데이터를 읽을 때 판정된다. S, E, I가 오류가 있음을 나타내는 F로 판정될 수 있으며, 오류가 해결된 후 S 또는 E로 돌아갈 수 있다.

[0052] 즉, 유효한 데이터 중 M, O는 더티 데이터를, E, S는 클린 데이터를 나타내며, I, F는 유효하지 않거나 오류가 있는 데이터를 나타낸다.

[0053] 이러한 상태에 따라 구분되어 신뢰 캐시(102)와 비신뢰 캐시(104)에 입력되거나 출력되는 데이터 블럭은 신뢰성 제어부(미도시)에 의해 오류 정정 코드 또는 오류 검출 코드를 사용하여 오류가 정정되거나 검출된다(도 7 참조).

[0054] 전술한 바와 같이, 신뢰 캐시(102)와 비신뢰 캐시(104)에는 서로 상이한 제 1 공급 전압과 제 2 공급 전압이 인가된다. 예를 들어, 비신뢰 캐시(104)로 할당된 웨이(way)에는 신뢰 캐시(102)로 할당된 웨이보다 낮은 전압이 인가될 수 있다.

[0055] 이러한 웨이별 공급 전압 정보는 비율 제어부(500)에 의해 전원 공급부(200)에 제공될 수 있다.

[0056] 또한 비율 제어부(500)는 교체 제어부(300)에 의해 신뢰 캐시(102) 또는 비신뢰 캐시(104)에서 쫓겨나는 희생 블럭 정보를 기초로 비신뢰 캐시 교체 희생자 수 및 신뢰 캐시 교체 희생자 수를 각각 저장하는 클린 카운터(502)와 더티 카운터(504)를 포함할 수 있으며, 비신뢰 캐시 교체 희생자 수 또는 신뢰 캐시 교체 희생자 수가 소정의 임계값을 넘어설 때마다, 신뢰 캐시(102)와 비신뢰 캐시(104)로 할당되는 웨이의 비율을 조절할 수 있다.

[0057] 도 4는 본 발명의 일실시예에 따른 전원 공급부를 도시하고 있다.

[0058] 캐시(10)의 각 웨이를 동적으로 신뢰 캐시(102) 또는 비신뢰 캐시(104)로 사용하기 위해서는 각 웨이에 다른 전압을 인가할 수 있어야 한다. 본 발명의 일실시예에 따른 캐시(10)는 이를 위해 여러 가지 방법을 사용할 수 있다. 예를 들어, DC/DC 컨버터를 추가하거나 SRAM 내부에서 트랜지스터를 추가하여 인가 전압을 설정하는 등의

방법을 사용할 수 있다.

[0059] 전압 스케일링(voltage scaling)은 회로의 전압을 동적으로 조절하는 기술이다. 캐시 메모리는 주로 SRAM으로 구현되며, SRAM의 동적 전력(dynamic power)는 입력 전압(supply voltage)의 제곱함수에 비례하여 줄어들고, 정적 전력(static power)는 전압의 지수 함수에 비례하여 줄어듦을 고려할 때, 전압 스케일링은 SRAM 캐시의 전력을 낮추는데 매우 효과적으로 적용될 수 있다.

[0060] 도면은 종래의 전압 스케일링 회로(DVS: dynamic voltage scaling circuit)를 확장하여 신뢰 캐시(102)와 비신뢰 캐시(104)에 서로 다른 전압을 공급하는 실시예를 도시하고 있다. 이러한 방법에는 예를 들어, DC/DC 컨버터 또는 전원 조절(supply regulation) 회로 등 다양한 방법이 사용될 수 있다.

[0061] 상이한 전압을 공급하기 위해, PWM 로직(210)이 피드백 라인과 함께 신뢰 캐시(102)에 할당된 각 웨이에 연결된 최저 인가 전압(Vcc_MIN) 라인을 제어하고 있다. 전술한 바와 같이, PWM 로직(210)은 비율 제어부(500)에 의해 웨이별 공급 전압 정보를 받을 수 있다.

[0062] 이러한 칩 내부(on chip) 구성 요소들에 도시한 바와 같은 칩 외부(off chip) 구성 요소들을 추가하여, 비신뢰 캐시(104)에 최저 인가 전압보다 더 낮은 전압을 인가할 수 있다. 이들 구성 요소를 칩 외부에 추가하는 것은 해당 구성 요소들이 DC/DC 컨버터 및 Vcc 선택기 등의 부피가 큰 수동(passive) 소자이기 때문으로, 이들 요소를 반드시 칩 외부에 위치시켜야 하는 것은 아니다.

[0063] 도면은 종래 DVS회로에 추가한 요소들을 굵은 선으로 표시하고 있다.

[0064] 도시된 실시예에서 칩 외부에 추가된 또하나의 DC/DC 컨버터는 신뢰 캐시(102)에 할당된 각 웨이에 공급되는 최저 인가 전압(Vcc_MIN)보다 더 낮은 전압(Vcc_MIN - a)을 만들어낸다.

[0065] 웨이 #0에 연결되어 있는 것과 같은 구조를 갖는 Vcc 선택기(218)가 각 웨이마다 연결되어 있으며, 가상 Vcc 제어부(220)가 각 Vcc 선택기(218)의 두 트랜지스터 중 하나만을 활성화시켜, 두개의 Vcc 라인 중 하나만을 선택적으로 각 웨이에 공급해준다.

[0066] 이러한 구조를 통해 본 발명의 일실시예에 따른 전원 공급부(200)는 신뢰 캐시(102)로 할당된 웨이에 최저 인가 전압(Vcc_MIN)을, 비신뢰 캐시(104)로 할당된 웨이에 그보다 더 낮은 전압(Vcc_MIN -a)를 공급할 수 있다.

[0067] 도 5는 종래의 PLRU 트리의 실시예를 도시하고 있으며, 도 6은 본 발명의 일실시예에 따른 듀얼 PLRU 트리의 실시예를 도시하고 있다.

[0068] 전술한 바와 같이, 본 발명의 일실시예에 따른 캐시(10)는 신뢰 캐시(102)와 비신뢰 캐시(104)를 별도의 교체 정책(replacement policy)이 아닌, 통합 교체 정책 및 하나의 데이터 구조를 사용하여 관리할 수 있다.

[0069] 예를 들어, 교체 제어부(300)는 종래의 PLRU 트리(Pseudo-LRU tree)를 변형한 듀얼 PLRU 트리를 교체될 희생자 블럭을 결정하는 데 사용할 수 있다. 도 5에 도시되어 있는 바와 같이, 종래의 PLRU 트리는 각 세트(set) 별로 이진 트리를 가지고 있고, 트리의 각 노드는 접근될 때마다 반대편을 가리키도록 해, 캐시 웨이(way)들이 고르게 사용되도록 하고 있다. 본 발명의 일실시예에 따른 캐시(10)가 사용하는 변형된 PLRU 트리는 비율이 바뀜에 따라 정해진 패턴대로 변하여 웨이(way) 사용이 고르게 분포되도록 하고 있다.

[0070] 이에 따라, 본 발명의 일실시예에 따른 캐시(10)는 캐시 영역 구분에 따라 각 구역에 대해 별도의 저장소 및 회로를 추가 구현할 필요없이, 종래의 PLRU를 약간 변형한 데이터 구조를 사용하는 기존의 저장소를 사용하여 신뢰 캐시(102)와 비신뢰 캐시(104)를 통합 관리할 수 있다.

[0071] 예를 들어, 본 발명의 일실시예에 따른 듀얼 PLRU 회로는 종래의 PLRU 트리를 위한 회로에 9비트 카운터 2개와 1개의 비교기를 추가하여 구현될 수 있다. 앞서 도 3에 도시되어 있는 바와 같이, 신뢰 캐시는 태그와 상태 정보를 저장한다.

[0072] 도 3을 도 5 및 도 6과 함께 참조하여, 본 발명의 일실시예에 따른 캐시(10)의 교체 제어부(300)의 구조 및 동작을 살펴본다. 설명의 편의를 위해, 도면의 실시예는 8웨이 L2 캐시라고 가정한다.

[0073] 상위 캐시인 L1으로부터 쫓겨난(evicted) 블럭이 도착하면, L2 캐시(10)는 해당 데이터의 상태를 먼저 검사한다. 클린 상태(E, S, 또는 I)의 데이터는 신뢰 캐시(102)에 저장되고, 더티 상태(M 도는 O)의 데이터는 비신뢰 캐시(104)에 저장되며, 이때 해당 블럭과 교체되어 신뢰 캐시(102) 또는 비신뢰 캐시(104)에서 쫓겨나는 교체 희생자(victim) 블럭이 있을 수 있다.

[0074] 이러한 교체 희생자를 찾기 위해 사용하는 듀얼 PLRU의 동작을 설명하기 위해 먼저 도 5를 참조하여 트리 기반의 PLRU를 설명하면, PLRU는 트리를 사용하여 LRU 블럭을 찾아낸다. 8웨이 캐시가 실시예로 사용되었기 때문에 트리의 말단 노드는 도시되어 있는 바와 같이 8개이다.

[0075] 트리의 각 노드의 값이 다음 노드를 결정하며, 그렇게 도달한 말단 노드가 쫓겨날 희생자이다. 예를 들어, 도 5의 실시예에서, 트리는 4번째 웨이가 희생자 블럭임을 나타내고 있다. 루트 노드의 값이 1이므로, 왼쪽에 있는 자식 노드로 내려가고, 해당 노드의 값이 0이므로, 오른쪽에 있는 자식 노드로 가는 식이다.

[0076] 신뢰 캐시(102)와 비신뢰 캐시(104)의 교체 희생자 찾기를 동시에 수행하기 위해 도 5의 트리를 도 6과 같이 수정할 수 있다. 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율은 전술한 바와 같이 동적으로 조정될 수 있는데, 이를 위해 도 6의 하단에 있는 표와 같이, 신뢰 캐시(102)와 비신뢰 캐시(104)가 되도록 대칭적으로 증가/감소하도록 하는 패턴을 구성할 수 있다.

[0077] 예를 들어, 신뢰 캐시(102)로 사용하는 웨이(R, 색칠됨)와 비신뢰 캐시(104)로 사용되는 웨이(U, 색칠되지 않음)를 첫번째 웨이에서부터 순서대로 RRUUUUUU(2:6), RRRUUUUU(3:5), RRRRUUUU(4:4)의 패턴으로 할당하면, 지나치게 희생자 수가 많아질 것이지만, 도면의 표와 같은 패턴을 사용하면 신뢰 캐시(102)와 비신뢰 캐시(104)가 대칭적으로 증가/감소하므로, 교체 희생자 수를 줄일 수 있다.

[0078] 도면의 표는 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율이 변하면 표의 좌측에 있는 비율에 맞게 우측에 있는 패턴을 사용하여 각 웨이를 할당할 수 있음을 보이고 있다. 예를 들어, 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율이 1:7임을 나타내는 첫번째 행은 RUUUUUUU의 패턴에 따라 첫번째 캐시만 신뢰 캐시로 할당함을 나타낸다.

[0079] 또한 도면의 상단에 도시되어 있는 트리는 색칠된 웨이가 3개이므로 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율이 3:5일 때를 도시하고 있음을 알 수 있는데, 하단의 표에서 3:5일 때의 패턴은 RURURUUU이므로, 트리의 말단이 나타내고 있는 웨이도 RURURUUU의 패턴으로 신뢰 캐시 또는 비신뢰 캐시로 할당되었다.

[0080] 교체 희생자를 찾는 과정은 도 5의 PLRU 트리에서의 과정과 같다(단, D: 더티, C: 클린). 비율이 3:5일 때 RRRUUUUU 패턴을 사용하고 있는 도 5의 실시예와 비교하여 보면, 도 5에서는 5번째부터 8번째 웨이까지의 접근은 반대 방향으로 향하게 하기 위해 루트 값을 1로 만들기 때문에, 4번째 웨이에서의 희생자 수를 지나치게 증가시킨다. 4번째 웨이가 루트의 왼쪽에 있는 4개의 웨이 중 유일한 클린 데이터 저장 웨이이기 때문이다. 그러나 도 6의 실시예에서는 신뢰 캐시(102)로 사용하는 웨이와 비신뢰 캐시(104)로 사용하는 웨이가 골고루 섞여 있으므로 이러한 문제가 완화된다.

[0081] 이와 같이, 본 발명의 일실시예에 따른 캐시(10)는 균형잡힌 패턴을 사용하여 신뢰 캐시(102)와 비신뢰 캐시(104)를 구분하고, 교체 희생자를 효과적으로 찾아낼 수 있는 듀얼 PLRU 트리를 사용함으로써, 특정 웨이에 교체 희생자가 몰리는 문제를 해결할 수 있다.

[0082] 그러나 캐시(10)가 사용하는 교체 정책 및 교체 정책을 수행하기 위한 데이터 구조는 이에 제한되지 않는다.

[0083] 도 7은 본 발명의 일실시예에 따른 신뢰성 제어부를 도시하고 있다.

[0084] 본 발명에는 여러 가지 오류 검출 및 정정 기술이 적용 가능하다. 예를 들어, 본 발명의 일실시예에 따른 신뢰성 제어부(400)는 더티 데이터에는 오류 정정 코드(ECC: error correcting code)를 사용하고, 클린 데이터에는 오류 검출 코드(EDC: error detection code)를 사용할 수 있다. 오류를 검출하고 정정할 수 있는 오류 정정 코드는 오류를 검출만 할 수 있는 오류 검출 코드보다 전력 소모량 증가 등의 비용이 있으므로, 전술한 바와 같은 이유로 굳이 오류를 정정할 필요가 없는 비신뢰 캐시(104) 데이터는 오류 정정 코드 대신 오류 검출 코드를 사용하는 것이 바람직하다.

[0085] 따라서, 전술한 바와 같이, 본 발명의 일실시예에 따른 신뢰성 제어부(400)는 신뢰 캐시(102)에 저장되어 있는 데이터는 오류 정정 코드를 사용하여 오류 발생을 검출하고 오류를 정정하며, 비신뢰 캐시(104)에 저장되어 있는 데이터는 오류 검출 코드를 사용하여 오류 발생을 검출하고, 오류 발생이 검출되면 다른 메모리에서 올바른 데이터를 가져올 수 있도록, 오류 알림 신호를 발생시킨다. 또한, 전술한 바와 같이, 오류 알림 신호는 캐시 미스 신호의 형태일 수 있다.

[0086] 예를 들어, 오류 정정 코드로는 SECDED, DECTEC(2 error correct, 3 error detect), QECPED(4 error correct, 5 error detect), OECNED(8 error correct, 9 error detect) 등을 사용할 수 있으며, 오류 검출 코드로는 CRC 또는 패러티(parity) 등을 사용할 수 있다.

[0087] 이때, 도시된 바와 같이, 오류 정정 코드와 오류 검출 코드를 하나의 저장소를 공유하여 기록되고, 데이터의 상태에 따라 오류 정정 코드를 사용할 것인지 오류 검출 코드를 사용할 것인지 결정된다.

[0088] 데이터가 캐시에 쓰여지는 과정은 점선을 따라 ECC/EDD 산출부(410)를 거쳐서 이루어지며, 읽혀지는 과정은 실선을 따라 ECC/EDD 산출부(410)와 비교기(=), 그리고 필요에 따라 ECC 적용부(422)를 거쳐 이루어진다.

[0089] 예를 들어, ECC/EDD 산출부(410)는 쓰기 데이터에 대해 ECC 산출부(412)가 산출한 오류 검출 코드를 사용할지 EDC 산출부(414)가 산출한 오류 정정 코드를 사용할지는 ECC/EDD 산출부(410)에 공급된 데이터의 상태 정보(예: 데이터의 더티 비트)를 기준으로 판단할 수 있다.

[0090] 데이터를 읽을 때는, 저장되어 있는 오류 정정 코드 또는 오류 검출 코드가 새로 계산되는 오류 정정 코드 또는 오류 검출 코드와 비교된다. 신뢰 캐시(102)라면 오류 정정 코드가 계산되며, 비교 결과는 데이터를 정정할 것인지 여부를 판단하는 데 사용된다. 비신뢰 캐시(104)라면 오류 검출 코드가 계산되며, 비교 결과는 오류 알림 신호(예: 캐시 미스 신호)를 발생시킬 것인가의 여부를 판단하는 데 사용된다.

[0091] 신뢰성 제어부(400)의 이러한 구성은 기존의 캐시에 사용되는 오류 정정 코드 회로를 큰 변화 없이 효율적으로 변형하여 구현할 수 있다. 또한, 이러한 구성을 통해, 본 발명의 일실시예에 따른 신뢰성 제어부(400)는 특정 웨이가 신뢰 캐시(102) 또는 비신뢰 캐시(104)로 용도가 변경될 때, 오류 정정 코드와 오류 검출 코드를 선택적으로 사용할 수 있다.

[0092] 도 8 내지 도 10은 본 발명의 일실시예에 따른 캐시 비율 조정 중, 신뢰 캐시를 비신뢰 캐시로 변경하는 과정에서 더티 블록을 처리하는 방법을 도시하고 있다.

[0093] 먼저 도 8을 보면, 웨이 #1, 웨이 #2, 및 웨이 #3은 신뢰 캐시(102)로, 웨이 #4는 비신뢰 캐시(104)로 쓰이고 있다. 각 웨이의 상태비트(110)는 데이터의 상태를 나타내고 있다. 예를 들어, 웨이 #3의 첫번째 세트의 상태비트(110)는 더티 데이터임을 나타내는 D를, 웨이 #3의 두번째 세트의 상태비트(110)는 클린 데이터임을 나타내는 C를 표시하고 있다. 이는 도 3에서 설명한 여러 상태 중 하나를 간략하게 표시한 것으로, 예를 들어, 더티 상태는 M, O, 또는 E일 수 있다.

[0094] 웨이 #3의 두번째 세트의 상태비트(110)가 클린 데이터인 것은 웨이 #3이 이전에 비신뢰 캐시(104)에서 신뢰 캐시(102)로 변경되면서 각 세트가 골고루 사용되지 못하였기 때문일 수 있다.

[0095] 도 9 및 도 10은 도 8의 웨이 #3가 비율 조정부(500)에 의해 신뢰 캐시(102)에서 비신뢰 캐시(104)로 변경될 때의 두가지 방법을 도시하고 있다. 즉, 도 9 및 도 10은 신뢰 캐시(102)에서 비신뢰 캐시(104)로 변경될 때 신뢰 캐시(102)에 저장되어 있는 데이터를 잃지 않기 위해 비신뢰 캐시(104)로 변경하기 전에 하위 계층 메모리(예: 주 메모리, 이하 편의상 메모리로 지칭)로 라이트백(write-back)하는 방법의 두가지 실시예를 도시하고 있다.

[0096] 단, 웨이 #3의 두번째 세트처럼 전술한 바와 같은 이유로 신뢰 캐시(102)에 저장되어 있는 클린 데이터는 도 9 및 도 10의 두가지 실시예가 공통적으로 도시하고 있는 바와 같이, 웨이 #3을 다시 비신뢰 캐시(104)로 변경할 때 해당 세트에 대해서는 데이터 손실을 방지하기 위한 처리를 할 필요가 없다. 앞서 설명한 바와 같이 클린 데이터는 하위 계층 메모리 등 다른 메모리에 중복 저장되어 있기 때문이다.

[0097] 이제 더티 데이터의 처리에 대해 도 9의 실시예를 먼저 살펴보면, 도 8의 웨이 #3의 첫번째 세트는 더티 데이터를 나타내므로, 메모리 쓰기 큐(150)로 보내, 메모리로 라이트백시킨 후, 상태 비트를 I로 변경시켰음을 도시하고 있다. 즉, 도 9는 신뢰 캐시(102)에서 비신뢰 캐시(104)로 변경되는 웨이 자체의 더티 데이터를 메모리로 내보내는 실시예를 도시하고 있다.

[0098] 또다른 실시예인 도 10의 실시예는 웨이 #1, 웨이 #2, 웨이 #3의 첫번째 세트에서 LRU 블럭을 찾아, 해당 LRU 블럭인 웨이 #2의 첫번째 세트를 대신 메모리로 내보낸다. 즉, 해당 세트의 신뢰 캐시(102)에 할당된 모든 웨이들 에서 LRU를 찾은 결과, 웨이 #2의 첫번째 세트가 해당 세트의 LRU로 판정되었기 때문에, 웨이 #2의 첫번째 세트를 메모리 쓰기 큐(150)로 보내, 메모리로 라이트백시킨 후, 신뢰 캐시(102)에서 비신뢰 캐시(104)로 변경되는 웨이인 웨이 #3의 첫번째 세트에 저장되어 있던 더티 데이터를 해당 블럭으로 이동 저장하는 실시예가 도 10에 도시되어 있다.

[0099] 도 11은 본 발명의 일실시예에 따른 캐시 운영 방법의 흐름을 도시하고 있다.

[0100] 캐시 영역(예: 캐시 웨이)을 기지정된 비율에 따라 신뢰 캐시(102)와 비신뢰 캐시(104)로 구분한다(S110).

[0101] 설계시 어플리케이션을 구동시키면서 실험하여 찾은 최적의 비율에 따라 신뢰 캐시(102)와 비신뢰 캐시(104)를

구분할 수도 있다. 즉, 각 어플리케이션 또는 여러 어플리케이션을 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율을 다양하게 하여 실행시켜보고, 그 과정에서 가장 적합하다고 판단되는 비율을 찾는 것이다.

[0102] 일단 최적의 비율로 신뢰 캐시(102)와 비신뢰 캐시(104)를 구분한 후에는 해당 비율을 유지할 수 있다(하기의 S130 단계를 수행하지 않는 정적인 방법).

[0103] 그러나, 캐시 블럭(100)을 두개의 그룹으로 구분하면 캐시 미스가 증가하여 성능이 저하될 수 있으므로, 각 어플리케이션에 적합한 비율을 찾아 성능 저하를 최소화할 수 있다. 따라서, 어플리케이션들을 실행하면서 각 어플리케이션에 적합한 비율을 찾아가는 것이 바람직할 수 있다(하기의 S130 단계를 수행하는 동적인 방법).

[0104] 예를 들어, 바람직한 실시예에서, 초기 구분 단계(S110)에서는 비신뢰 캐시(104)에 신뢰 캐시(102)보다 많은 캐시 영역을 할당한 후, 운영시의 동적 비율 조정 단계(S130)에서 필요에 따라 신뢰 캐시(102)를 늘려갈 수 있다. 비신뢰 캐시(104)가 좀더 소모 전력이 낮고 오버헤드가 낮은 오류 검출 코드를 사용하기 때문이다.

[0105] 다음, 신뢰 캐시(102)와 비신뢰 캐시(104)를 구분하여 운영한다(S120). 이때, 데이터 저장, 전원 공급, 및 데이터 신뢰성 제어는 신뢰 캐시(102)와 비신뢰 캐시(104)를 구분하여 수행하되, 신뢰 캐시(102) 또는 비신뢰 캐시(104)에서 쫓겨날 데이터를 선택할 때는 통합 교체 정책(예: 듀얼 PLRU 트리 사용)을 사용한다.

[0106] 즉, 신뢰 캐시(102)에는 더티(dirty) 데이터를 저장하고, 비신뢰 캐시(104)에는 클린(clean) 데이터를 저장하고, 신뢰 캐시(102)에는 캐시 데이터의 신뢰성을 보장하는 최저 인가 전압을 공급하고, 비신뢰 캐시(104)에는 최저 인가 전압보다 낮은 전압을 공급하며, 신뢰 캐시(102) 데이터는 오류 정정 코드를 사용하여 오류를 검출 및 자체 정정하고, 비신뢰 캐시(104) 데이터는 오류 검출 코드를 사용하여 오류를 검출하고 오류 발생 검출시 다른 메모리에 저장되어 있는 중복 데이터를 사용하여 복구한다.

[0107] 단위 시간마다 신뢰 캐시(102)와 비신뢰 캐시(104)의 비율을 동적으로 조정한다(S130). 여기에서 단위 시간이란 상기 단계(S120)에서 신뢰 캐시(102) 또는 비신뢰 캐시(104)에서 데이터가 쫓겨날 때마다 카운팅되는 신뢰 캐시 교체 희생자 수 또는 비신뢰 캐시 교체 희생자 수의 증가율이 임계값 이상이 되는 시점을 뜻한다. 전술한 바와 같이 일실시예에서 비율을 동적으로 조정하는 단계(S130)는 생략될 수 있다.

[0108] 도 12는 본 발명의 일실시예에 따른 캐시 운영 방법의 성능을 도시하고 있다.

[0109] 도면은 다양한 응용 프로그램을 사용하여 측정한 결과 그래프이다. 그래프로부터 본 발명의 일실시예에 따른 캐시 운영 방법은 종래 최저 수준과 대비했을 때 동적 전력은 -15.8%, 정적 전력은 -12.0%, 에너지×지연은 -12.8% 감소시킬 수 있다는 결론을 내릴 수 있다.

[0110] 본 발명의 일 실시예는 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터에 의해 실행가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독 가능 매체는 컴퓨터 저장 매체 및 통신 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다. 통신 매체는 전형적으로 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈, 또는 반송파와 같은 변조된 데이터 신호의 기타 데이터, 또는 기타 전송 메커니즘을 포함하며, 임의의 정보 전달 매체를 포함한다.

[0111] 본 발명의 방법 및 시스템은 특정 실시예와 관련하여 설명되었지만, 그것들의 구성 요소 또는 동작의 일부 또는 전부는 범용 하드웨어 아키텍처를 갖는 컴퓨터 시스템을 사용하여 구현될 수 있다. 본 발명의 일 실시예의 하나 이상의 구성 요소 또는 동작을 실시하기 위하여 사용될 수 있는 컴퓨터 시스템 아키텍처의 일례를 설명하면, 하드웨어 시스템은 프로세서, 캐쉬, 메모리 및 상술한 기능에 관련된 하나 이상의 소프트웨어 어플리케이션 및 드라이버를 포함할 수 있다.

[0112] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

[0113] 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으

로 해석되어야 한다.

부호의 설명

[0114] 10: 캐시

100: 캐시 블럭

102: 신뢰 캐시

104: 비신뢰 캐시

200: 전원 공급부

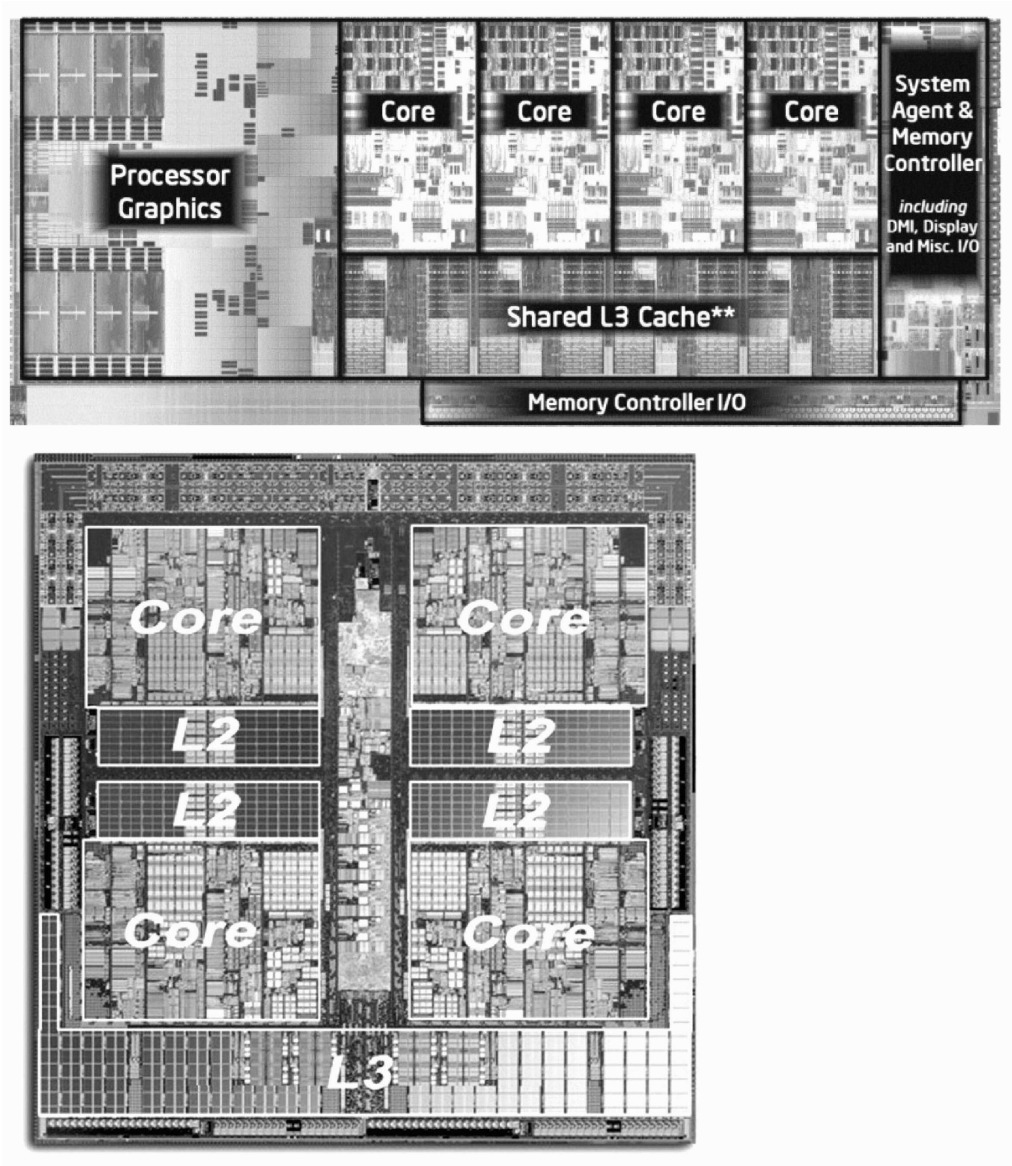
300: 교체 제어부

400: 신뢰성 제어부

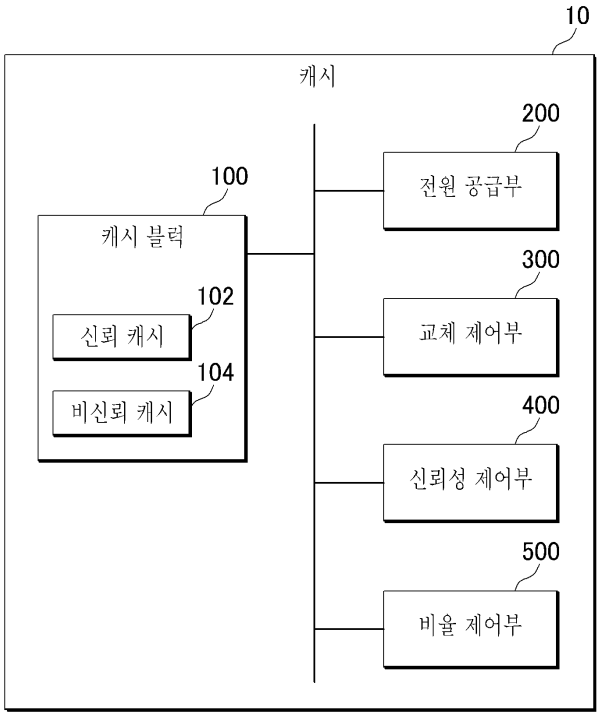
500: 비율 제어부

도면

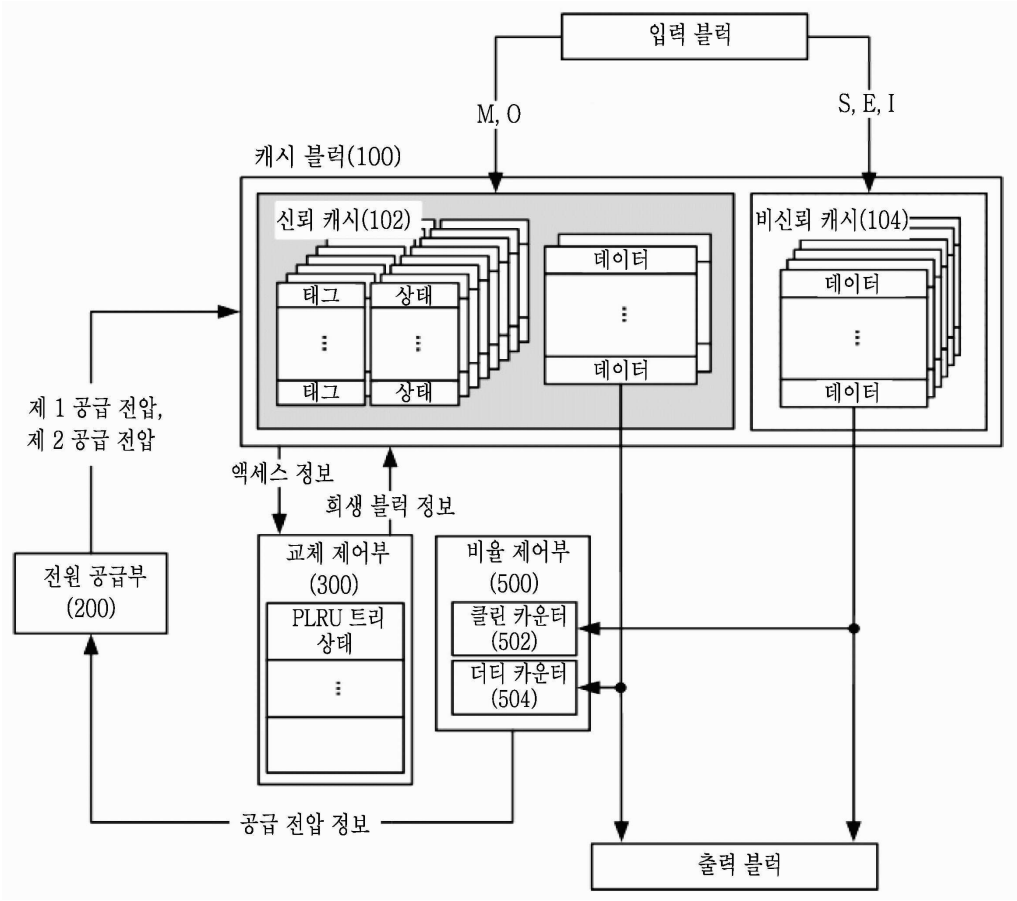
도면1



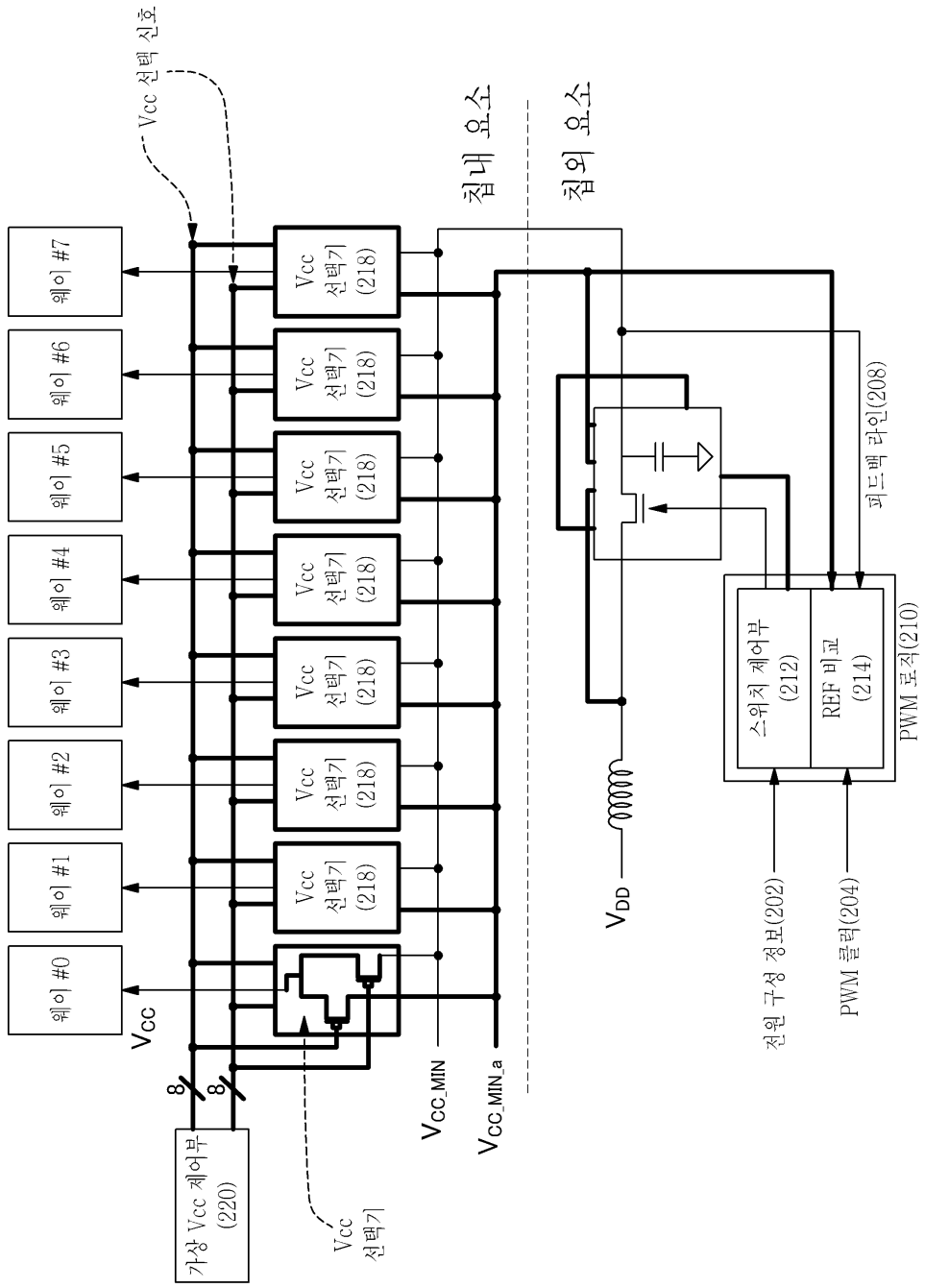
도면2


10
캐시
200
전원 공급부
100
캐시 블럭
300
교체 제어부
102
신뢰 캐시
104
비신뢰 캐시
400
신뢰성 제어부
500
비율 제어부

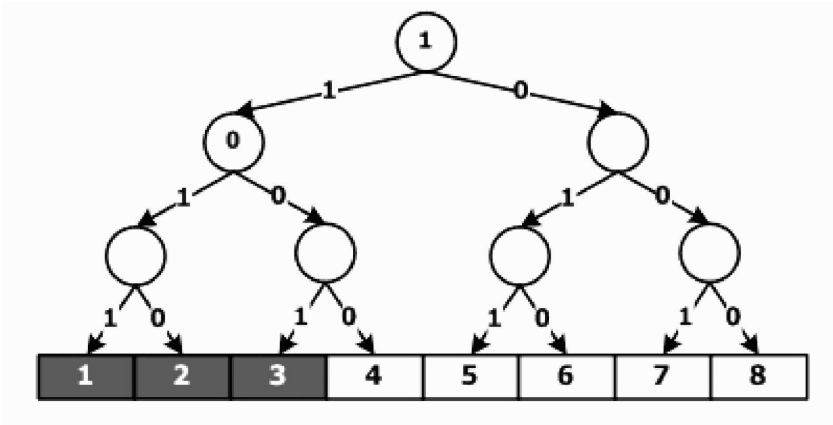
도면3


입력 블럭
M, O
S, E, I
캐시 블럭(100)
신뢰 캐시(102)
비신뢰 캐시(104)
태그
상태
데이터
제 1 공급 전압,
제 2 공급 전압
액세스 정보
희생 블럭 정보
전원 공급부
(200)
교체 제어부
(300)
PLRU 트리
상태
비율 제어부
(500)
클린 카운터
(502)
더티 카운터
(504)
공급 전압 정보
출력 블럭

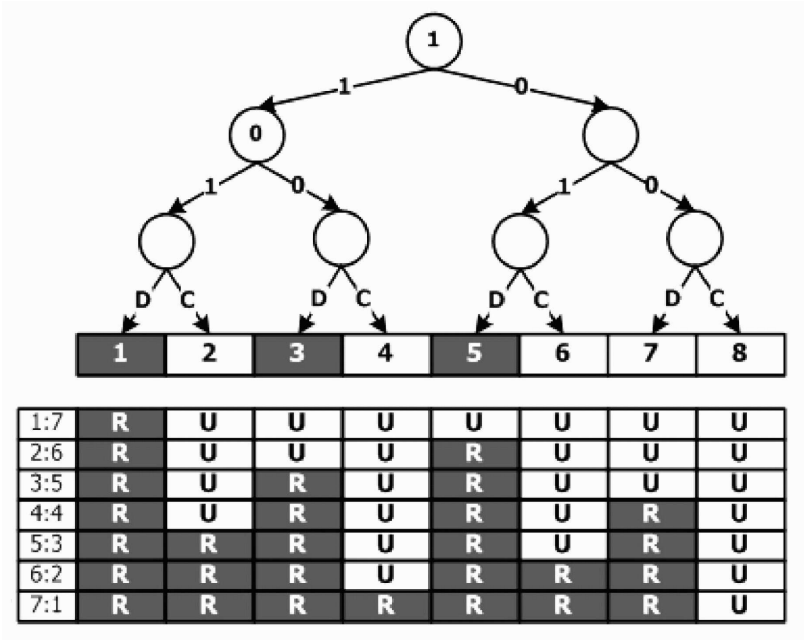
도면4


Vcc 선택 신호
칩내 요소
칩외 요소
웨이 #7
웨이 #6
웨이 #5
웨이 #4
웨이 #3
웨이 #2
웨이 #1
웨이 #0
Vcc 선택기 (218)
Vcc
8
8
가상 Vcc 제어부 (220)
Vcc 선택기
VCC_MIN
VCC_MIN_a
VDD
스위치 제어부 (212)
REF 비교 (214)
피드백 라인(208)
PWM 로직(210)
전원 구성 정보(202)
PWM 클럭(204)

도면5

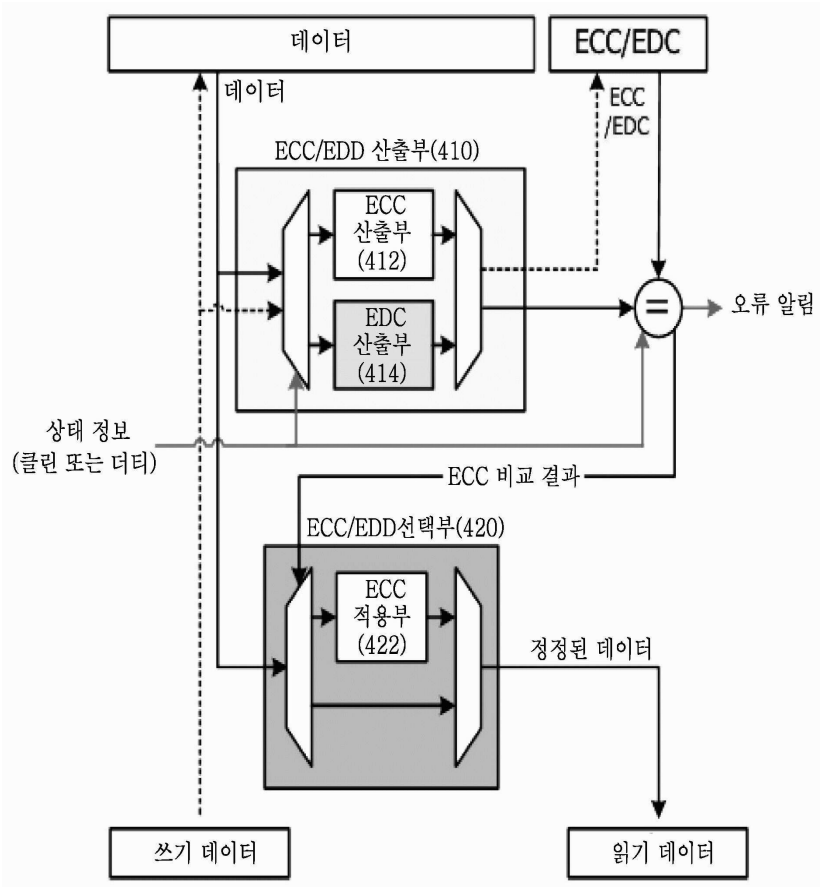


도면6

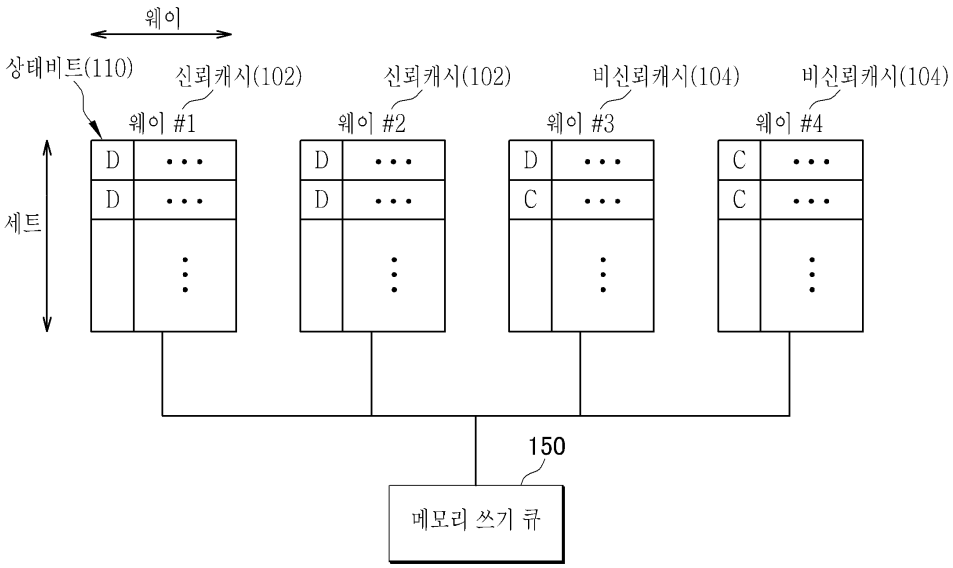


1:7	R	U	U	U	U	U	U	U
2:6	R	U	U	U	R	U	U	U
3:5	R	U	R	U	R	U	U	U
4:4	R	U	R	U	R	U	R	U
5:3	R	R	R	U	R	U	R	U
6:2	R	R	R	U	R	R	R	U
7:1	R	R	R	R	R	R	R	U

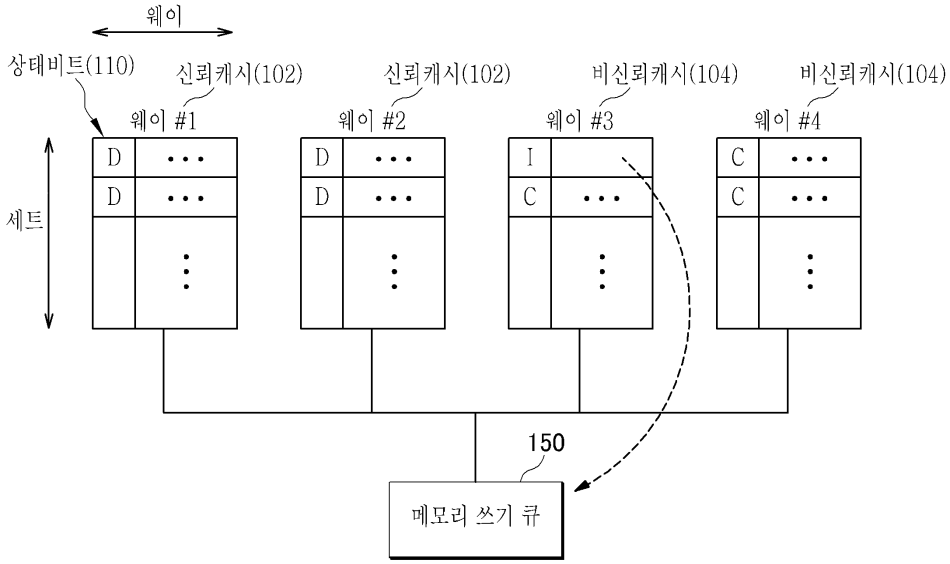
도면7


데이터
ECC/EDC
데이터
ECC
/EDC
ECC/EDD 산출부(410)
ECC
산출부
(412)
EDC
산출부
(414)
=
오류 알림
상태 정보
(클린 또는 더티)
ECC 비교 결과
ECC/EDD선택부(420)
ECC
적용부
(422)
정정된 데이터
쓰기 데이터
읽기 데이터

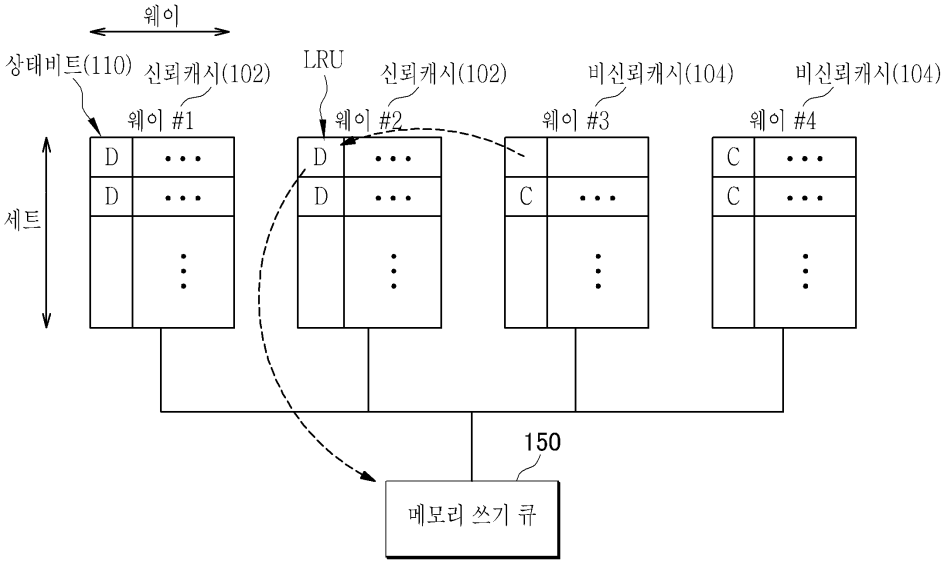
도면8


웨이
상태비트(110)
신뢰캐시(102)
신뢰캐시(102)
비신뢰캐시(104)
비신뢰캐시(104)
웨이 #1
웨이 #2
웨이 #3
웨이 #4
D
D
D
D
D
C
C
C
세트
150
메모리 쓰기 큐

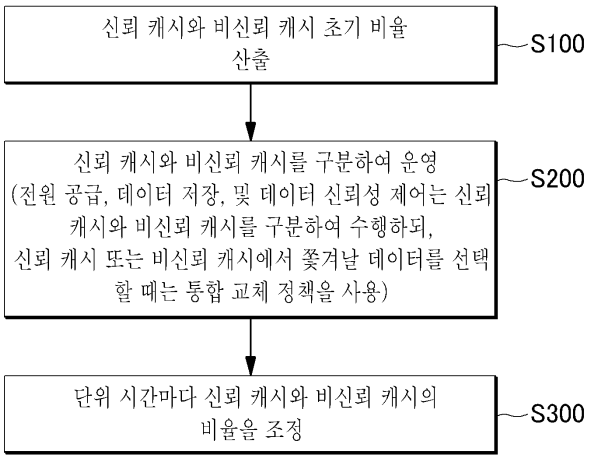
도면9


웨이
상태비트(110)
신뢰캐시(102)
신뢰캐시(102)
비신뢰캐시(104)
비신뢰캐시(104)
웨이 #1
웨이 #2
웨이 #3
웨이 #4
세트
D
D
D
D
I
C
C
C
150
메모리 쓰기 큐

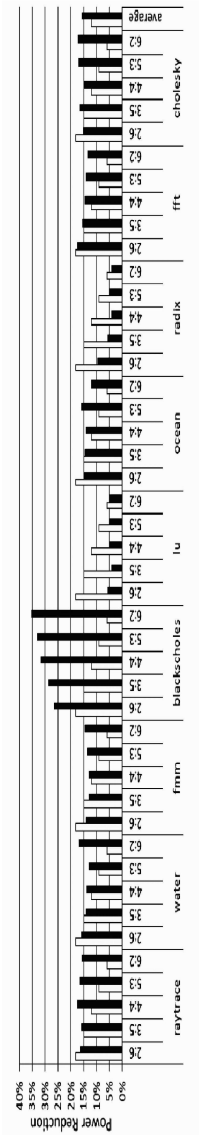
도면10


웨이
상태비트(110)
신뢰캐시(102)
LRU
신뢰캐시(102)
비신뢰캐시(104)
비신뢰캐시(104)
웨이 #1
웨이 #2
웨이 #3
웨이 #4
세트
D
D
D
D
C
C
C
150
메모리 쓰기 큐

도면11


신뢰 캐시와 비신뢰 캐시 초기 비율 산출
S100
신뢰 캐시와 비신뢰 캐시를 구분하여 운영 (전원 공급, 데이터 저장, 및 데이터 신뢰성 제어는 신뢰 캐시와 비신뢰 캐시를 구분하여 수행하되, 신뢰 캐시 또는 비신뢰 캐시에서 쫓겨날 데이터를 선택할 때는 통합 교체 정책을 사용)
S200
단위 시간마다 신뢰 캐시와 비신뢰 캐시의 비율을 조정
S300

도면12


Power Reduction
40%
35%
30%
25%
20%
15%
10%
5%
0%
raytrace
water
fmm
blackscholes
lu
ocean
radix
fft
cholesky
average
2.6
3.5
4.4
5.3
6.2