



(19) 대한민국특허청(KR)  
 (12) 등록특허공보(B1)

(45) 공고일자 2016년09월23일  
 (11) 등록번호 10-1659369  
 (24) 등록일자 2016년09월19일

- (51) 국제특허분류(Int. Cl.)  
*G06F 11/16* (2006.01) *G06F 12/08* (2016.01)  
 (52) CPC특허분류  
*G06F 11/1612* (2013.01)  
*G06F 12/0815* (2013.01)  
 (21) 출원번호 10-2015-7004898  
 (22) 출원일자(국제) 2012년07월31일  
 심사청구일자 2015년02월25일  
 (85) 번역문제출일자 2015년02월25일  
 (65) 공개번호 10-2015-0038344  
 (43) 공개일자 2015년04월08일  
 (86) 국제출원번호 PCT/US2012/048997  
 (87) 국제공개번호 WO 2014/021853  
 국제공개일자 2014년02월06일

## (56) 선행기술조사문헌

US20020124143 A1\*

US20070038798 A1\*

US20100180084 A1\*

'PERFECTORY: A Fault-Tolerant Directory Memory Architecture', IEEE TRANSACTIONS ON COMPUTERS VOL. 59 NO. 5(2010)\*

\*는 심사관에 의하여 인용된 문헌

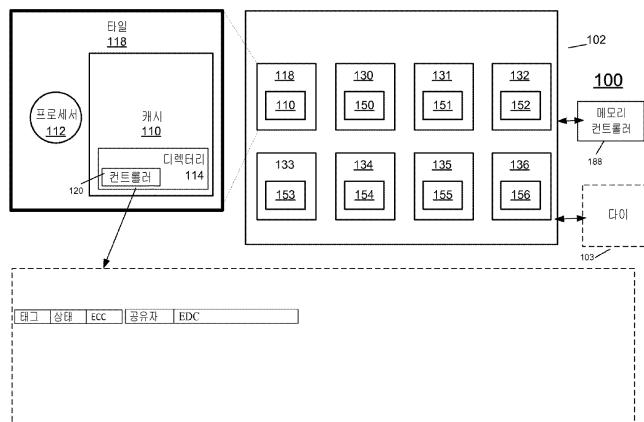
전체 청구항 수 : 총 31 항

심사관 : 이동하

## (54) 발명의 명칭 멀티 코어 프로세서 아키텍처 내 디렉터리 오류 정정

**(57) 요약**

캐시 일관성 정보를 처리하는 것과 데이터 블록에 대한 요청을 처리하는 것에 관한 기술이 일반적으로 설명된다. 일부 예시에서, 캐시 일관성 정보를 위한 방법이 기술되며, 이러한 방법은 데이터 블록을 식별하는 데에 유효한 태그 식별자를 디렉터리에 저장하는 단계를 포함할 수 있다. 방법은 태그 식별자와 연관하여 상태 식별자를 저 (뒷면에 계속)

**대 표 도**

장하는 단계를 더 포함할 수 있다. 상태 식별자는 데이터 블록의 일관성 상태를 식별하는 데에 유효할 수 있다. 방법은 태그 식별자와 연관하여 공유자 정보를 저장하는 단계를 더 포함할 수 있다. 공유자 정보는 데 이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효할 수 있다. 방법은, 컨트롤러에 의해, 디렉터리에서, 공유자 정보와 연관하여 복제 정보를 저장하는 단계를 포함할 수 있다. 복제 정보는 디렉터리에서 공유자 정보의 복제의 유형을 나타내는 데에 유효할 수 있고 복제된 세그먼트를 나타내는 데에 유효할 수 있다.

(52) CPC특허분류

*G06F 12/0817* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

멀티 코어 프로세서의 디렉터리의 캐시 일관성 정보(cache coherence information)를 저장하기 위한 방법으로서,

컨트롤러에 의해, 상기 멀티 코어 프로세서에서의 메모리 요청을 모니터링하는 단계;

상기 컨트롤러에 의해, 상기 디렉터리에서, 상기 메모리 요청에 기초하여,

데이터 블록을 식별하는 데에 유효한 태그 식별자;

상기 태그 식별자와 연관하여, 상기 데이터 블록의 일관성 상태를 식별하는 데에 유효한, 상태 식별자;

상기 태그 식별자와 연관된 공유자 정보 - 상기 공유자 정보는 상기 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효하고, 상기 공유자 정보는 상기 디렉터리에서 적어도 제1 세그먼트에 저장됨 -; 및

상기 공유자 정보와 연관된 복제 정보 - 상기 복제 정보는 상기 공유자 정보의 복제의 유형을 나타내는 데에 유효하고, 상기 복제 정보는 상기 디렉터리에서 적어도 제2 세그먼트를 식별하는데 더 유효하고, 상기 제2 세그먼트는 상기 공유자 정보의 복제를 포함함 - 를 저장하는 단계

를 포함하는 방법.

#### 청구항 2

제1항에 있어서,

상기 복제 정보는 상기 공유자 정보를 저장하는 상기 제1 세그먼트를 나타내는 데에 더 유효한 것인, 방법.

#### 청구항 3

제2항에 있어서,

상기 태그 식별자는 제1 태그 식별자를 포함하고, 상기 복제 정보는 제2 태그 식별자를 나타내는 데에 더 유효한 것인, 방법.

#### 청구항 4

제2항에 있어서,

상기 디렉터리의 적어도 둘 이상의 세그먼트는 동일한 크기인 것인, 방법.

#### 청구항 5

제1항에 있어서,

상기 디렉터리에서 상기 컨트롤러에 의해, 상기 태그 식별자 및 상기 상태 식별자와 연관하여 오류 정정 코드를 저장하는 단계를 더 포함하는 방법.

#### 청구항 6

제1항에 있어서,

상기 공유자 정보를 저장하는 단계는 공유자 표에 상기 공유자 정보를 저장하는 단계를 포함하고; 그리고

상기 공유자 표에 상기 공유자 정보를 저장하는 단계는 적어도 두 개의 세그먼트를 포함하는 공유자 표에 상기 공유자 정보를 저장하는 단계를 포함하는 것인, 방법.

**청구항 7**

제6항에 있어서,

상기 컨트롤러에 의해, 하나 이상의 오류 정정 코드를 저장하는 단계를 더 포함하고, 각각의 오류 정정 코드는 상기 공유자 표의 하나 이상의 세그먼트와 연관되는 것인, 방법.

**청구항 8**

제6항에 있어서,

상기 복제 정보를 저장하는 단계는, 상기 공유자 정보가 상기 디렉터리의 가로 단(row)의 모든 세그먼트에서 복제된다는 것을 나타내는 데에 유효한 복제 정보를 저장하는 단계를 포함하는 것인, 방법.

**청구항 9**

제1항에 있어서,

상기 데이터 블록은 제1 데이터 블록을 포함하고;

상기 공유자 정보를 저장하는 단계는 공유자 표에 상기 공유자 정보를 저장하는 단계를 포함하고, 상기 공유자 표는 상기 제1 세그먼트 및 상기 제2 세그먼트를 포함하고; 그리고

상기 방법은,

상기 공유자 정보가, 상기 제1 세그먼트에서, 상기 제1 데이터 블록을 저장하는 제1 캐시를 나타내는 데에 유효하고;

상기 공유자 정보가, 상기 제2 세그먼트에서, 상기 제1 데이터 블록을 저장하는 상기 제1 캐시를 나타내는 데에 유효하고; 그리고

상기 복제 정보가, 상기 제1 세그먼트의 상기 공유자 정보가 상기 제2 세그먼트에서 복제된다는 것을 나타내는 데에 더 유효하도록, 상기 컨트롤러에 의해 상기 디렉터리를 업데이트하는 단계를 더 포함하는 것인, 방법.

**청구항 10**

제9항에 있어서,

상기 공유자 표는 제3 세그먼트를 포함하고; 그리고

상기 방법은, 상기 제3 세그먼트에 저장되는 상기 공유자 정보가 상기 제1 데이터 블록을 저장하는 제2 캐시를 나타내는 데에 유효하도록 상기 컨트롤러에 의해 상기 디렉터리를 업데이트하는 단계를 더 포함하는 것인, 방법.

**청구항 11**

제1항에 있어서,

상기 태그 식별자는 제1 태그 식별자를 포함하고, 상기 상태 식별자는 제1 상태 식별자를 포함하고, 상기 데이터 블록은 제1 데이터 블록을 포함하고,

상기 공유자 정보를 저장하는 단계는, 제2 태그 식별자와 연관하여, 상기 공유자 정보를 저장하는 단계를 포함하고, 상기 제2 태그 식별자는 제2 데이터 블록을 식별하는 데에 유효하고, 상기 제2 태그 식별자와 연관된 제2 상태 식별자는 상기 제2 데이터 블록의 일관성 상태를 캐시되지 않음(uncached) 또는 유효하지 않음(invalid)으로 식별하는 데에 유효하고; 그리고

상기 복제 정보를 저장하는 단계는 상기 제1 태그 식별자와 연관하여, 복제 정보를 저장하는 단계를 포함하고,

상기 복제 정보는, 상기 제1 태그 식별자와 연관된 상기 공유자 정보가 상기 제2 태그 식별자와 연관된 상기 공유자 정보에서 복제된다는 것을 나타내는 데에 유효한 것인, 방법.

## 청구항 12

제6항에 있어서,

상기 공유자 정보를 상기 공유자 표에 저장하는 단계는 상기 제1 세그먼트 및 상기 제2 세그먼트를 포함하는 공유자 표에 상기 공유자 정보를 저장하는 단계를 포함하고;

상기 복제 정보를 저장하는 단계는 상기 제1 세그먼트에 대응하는 제1 슬롯 및 상기 제2 세그먼트에 대응하는 제2 슬롯을 포함하는 복제 정보를 저장하는 단계를 포함하고; 그리고

상기 복제 정보를 저장하는 단계는 상기 제2 세그먼트로 포인팅하는 상기 제1 슬롯의 포인터를 포함하는 복제 정보를 저장하는 단계를 포함하는 것인, 방법.

## 청구항 13

제1항에서,

상기 태그 식별자는 상기 디렉터리의 제1 가로 단에 저장되고; 그리고

상기 복제 정보는 상기 디렉터리의 제2 가로 단을 포인팅하는 포인터를 포함하는 것인, 방법.

## 청구항 14

제1항에 있어서,

상기 데이터 블록은 제1 데이터 블록을 포함하고;

상기 태그 식별자를 저장하는 단계는 상기 디렉터리의 제1 가로 단에 제1 태그 식별자를 저장하는 단계를 포함하고;

상기 공유자 정보를 저장하는 단계는 공유자 표에 상기 공유자 정보를 저장하는 단계를 포함하고;

상기 공유자 정보를 상기 공유자 표에 저장하는 단계는 적어도 두 개의 세그먼트를 포함하는 공유자 표에 상기 공유자 정보를 저장하는 단계를 포함하고,

상기 방법은,

상기 컨트롤러에 의해, 상기 제1 데이터 블록을 저장하는 상기 캐시에 대응하는 캐시 식별자의 비트의 수가 상기 적어도 두 개의 세그먼트의 비트의 전체 수의 절반보다 더 크다고 결정하는 단계; 및

상기 컨트롤러에 의해, 상기 디렉터리의 제2 가로 단에 저장된 제2 태그 식별자를 식별하는 단계 - 상기 제2 태그 식별자는 제2 데이터 블록을 식별하고 상기 제2 태그 식별자와 연관된 상태 식별자는 캐시되지 않은 또는 유효하지 않은 상태를 나타냄 - 를 더 포함하고,

상기 복제 정보를 저장하는 단계는, 상기 컨트롤러에 의해, 상기 제2 가로 단으로 포인팅하는 포인터와 함께 상기 제1 가로 단의 상기 제1 데이터 블록에 대한 복제 정보를 저장하는 단계를 포함하는 방법.

## 청구항 15

멀티 코어 프로세서에서 데이터 블록을 위한 요청을 처리하기 위한 방법으로서,

컨트롤러에 의해, 상기 멀티 코어 프로세서의 캐시 일관성 디렉터리에서, 상기 요청을 수신하는 단계;

상기 컨트롤러에 의해, 상기 데이터 블록과 연관된 태그 식별자를 식별하는 단계;

상기 컨트롤러에 의해,

상기 태그 식별자와 연관된, 상기 데이터 블록의 캐시 일관성 상태를 나타내는 데에 유효한, 상태 식별자;

상기 태그 식별자와 연관된 공유자 정보 - 상기 공유자 정보는 상기 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효하고, 상기 공유자 정보는 적어도 제1 세그먼트를 포함하는 공유자 표에서 저장됨 -; 및

상기 공유자 정보와 연관된 복제 정보 - 상기 복제 정보는 상기 공유자 정보의 복제의 유형을 나타내는

데에 유효하고, 상기 복제 정보는 상기 공유자 표에서 적어도 제2 세그먼트를 식별하는 데에 더 유효하고, 상기 제2 세그먼트는 상기 공유자 정보의 복제를 포함함 - 를 분석하는 단계;

상기 컨트롤러에 의해, 상기 공유자 표에서 상기 제1 세그먼트에서 오류를 검출하는 단계;

상기 공유자 표의 상기 제1 세그먼트에서 상기 오류를 검출하는 것에 응답하여, 상기 컨트롤러에 의해, 상기 복제 정보를 사용하여, 상기 공유자 정보의 상기 복제를 포함하는 적어도 상기 제2 세그먼트를 식별하는 단계; 및

상기 컨트롤러에 의해, 상기 식별된 제2 세그먼트로부터 상기 공유자 정보를 분석함으로써 상기 요청을 처리하는 단계

를 포함하는 방법.

### 청구항 16

제15항에 있어서,

상기 제1 세그먼트의 상기 오류를 검출하는 단계는 하나 이상의 오류 검출 코드 - 각각은 상기 공유자 표의 하나 이상의 세그먼트와 연관됨 - 를 사용하여 상기 오류를 검출하는 것인, 방법.

### 청구항 17

제15항에 있어서,

상기 컨트롤러에 의해, 상기 식별된 제2 세그먼트로부터 상기 공유자 정보를 분석함으로써 상기 요청을 처리하는 단계는,

상기 식별된 제2 세그먼트에 기초하여, 상기 컨트롤러에 의해, 상기 데이터 블록을 저장하는 상기 멀티 코어 프로세서에서 하나 이상의 캐시에 대한 주소 정보를 식별하는 단계; 및

상기 컨트롤러에 의해, 상기 요청에 응답하여, 검색된 주소 정보를 발송하는 단계를 더 포함하는 것인, 방법.

### 청구항 18

제15항에 있어서,

상기 태그 식별자를 식별하는 단계는 제1 태그 식별자를 식별하는 단계를 포함하고, 상기 제2 세그먼트를 식별하는 단계는 제2 태그 식별자와 연관되는 하나 이상의 다른 세그먼트를 식별하는 단계를 포함하는 것인, 방법.

### 청구항 19

제15항에 있어서,

상기 식별된 제2 세그먼트에서 오류를 검출하는 단계;

상기 식별된 제2 세그먼트에서 상기 오류를 검출하는 것에 응답하여,

상기 멀티 코어 프로세서의 하나 이상의 캐시로 메시지를 브로드캐스팅하는 단계;

상기 멀티 코어 프로세서의 상기 하나 이상의 캐시로부터 하나 이상의 응답 메시지를 수신하는 단계; 및

상기 수신된 응답 메시지에 기초하여, 상기 공유자 표에서 적어도 상기 제2 세그먼트를 업데이트하는 단계

를 더 포함하는 방법.

### 청구항 20

데이터 블록에 대한 요청을 처리하는 데에 유효한 멀티 코어 아키텍처로서,

제1 프로세서 및 제1 캐시를 포함하는 제1 타일(tile);

제2 프로세서 및 제2 캐시를 포함하는 제2 타일; 및

상기 제1 타일 및 상기 제2 타일과 통신하도록 구성되는 컨트롤러  
를 포함하고,

상기 컨트롤러는,

상기 제1 타일 및 상기 제2 타일 중에서 메모리 요청을 모니터링하고; 그리고 상기 메모리 요청에 기초하여,

디렉터리에서, 데이터 블록을 식별하도록 태그 식별자를 저장하고;

상기 디렉터리에서, 상기 태그 식별자와 연관하여, 상기 데이터 블록의 일관성 상태를 식별하는 데에 유효한, 상태 식별자를 저장하고;

상기 디렉터리에서, 상기 태그 식별자와 연관하여, 공유자 정보를 저장하고 - 상기 공유자 정보는 상기 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효하고, 상기 하나 이상의 캐시는 상기 제1 캐시 및 상기 제2 캐시를 포함하고, 상기 공유자 정보는 상기 디렉터리의 적어도 제1 세그먼트에 저장됨 - ; 그리고

상기 디렉터리에서, 상기 태그 식별자와 연관하여, 복제 정보를 저장하는데 유효하고, 상기 복제 정보는 상기 공유자 정보의 복제의 유형을 나타내는 데에 유효하고, 상기 복제 정보는 상기 디렉터리에서 적어도 제2 세그먼트를 식별하는 데에 더 유효하고, 상기 제2 세그먼트는 상기 공유자 정보의 복제를 포함하는 것인, 멀티 코어 아키텍처.

## 청구항 21

제20항에 있어서,

상기 복제 정보는 상기 공유자 정보를 저장하는 상기 제1 세그먼트를 나타내는 데에 더 유효한 것인, 멀티 코어 아키텍처.

## 청구항 22

제21항에 있어서,

상기 태그 식별자는 제1 태그 식별자를 포함하고, 상기 복제 정보는 제2 태그 식별자를 나타내는 데에 더 유효한 것인, 멀티 코어 아키텍처.

## 청구항 23

삭제

## 청구항 24

제20항에 있어서,

상기 컨트롤러는 공유자 표에 상기 공유자 정보를 저장하도록 더 구성되고, 상기 공유자 표는 적어도 두 개의 세그먼트를 포함하는 것인, 멀티 코어 아키텍처.

## 청구항 25

제20항에 있어서,

상기 복제 정보는, 상기 공유자 정보가 상기 디렉터리의 가로 단에서 모든 세그먼트에서 복제된다는 것을 나타내는 데에 더 유효한 것인, 멀티 코어 아키텍처.

## 청구항 26

제20항에 있어서,

상기 컨트롤러는 공유자 표에서 상기 공유자 정보를 저장하도록 구성되는 것이고,

상기 데이터 블록은 제1 데이터 블록을 포함하고;

상기 공유자 표는 상기 제1 세그먼트 및 상기 제2 세그먼트를 포함하는 것이고; 그리고  
상기 컨트롤러는,

상기 공유자 정보가, 상기 제1 세그먼트에서, 상기 제1 데이터 블록을 저장하는 상기 하나 이상의 캐시 중 제1 캐시를 나타내고;

상기 공유자 정보가, 상기 제2 세그먼트에서, 상기 제1 데이터 블록을 저장하는 상기 하나 이상의 캐시 중 상기 제1 캐시를 나타내고; 그리고

상기 복제 정보가, 상기 제1 세그먼트의 상기 공유자 정보가 상기 제2 세그먼트에서 복제된다는 것을 나타내도록, 상기 디렉터리를 업데이트하는 데에 유효한 것인, 멀티 코어 아키텍처.

### 청구항 27

제26항에 있어서,

상기 공유자 표는 제3 세그먼트를 포함하고, 상기 제3 세그먼트에 저장된 상기 공유자 정보는 상기 제1 데이터 블록을 저장하는 상기 하나 이상의 캐시 중 제2 캐시를 나타내는 데에 유효한 것인, 멀티 코어 아키텍처.

### 청구항 28

제20항에 있어서,

상기 컨트롤러는, 공유자 표에서 상기 공유자 정보를 저장하도록 구성되고,

상기 컨트롤러는,

상기 공유자 표가 상기 제1 세그먼트 및 상기 제2 세그먼트를 포함하고;

상기 복제 정보가 상기 제1 세그먼트에 대응하는 제1 슬롯 및 상기 제2 세그먼트에 대응하는 제2 슬롯을 포함하고; 그리고

상기 복제 정보가 상기 제2 세그먼트로 포인팅하는 데에 유효한 상기 제1 슬롯에 포인터를 포함하도록, 상기 디렉터리를 업데이트하는 데에 유효한 것인, 멀티 코어 아키텍처.

### 청구항 29

제20항에 있어서,

상기 컨트롤러는 상기 디렉터리에서 제1 가로 단에 상기 태그 식별자를 저장하도록 구성되고,

상기 컨트롤러는, 상기 제1 가로 단의 상기 복제 정보가 상기 디렉터리의 제2 가로 단으로 포인팅하는 데에 유효한 포인터를 포함하도록 상기 디렉터리를 업데이트하는 데에 유효한 것인, 멀티 코어 아키텍처.

### 청구항 30

제20항에 있어서,

상기 컨트롤러는 공유자 표에서 상기 공유자 정보를 저장하도록 구성되고, 상기 공유자 표는 적어도 상기 제1 세그먼트 및 상기 제2 세그먼트를 포함하고; 그리고

상기 컨트롤러는,

상기 데이터 블록에 대한 요청을 수신하고;

적어도 두 번은 상기 공유자 정보에 저장된 캐시 식별자를 식별하도록 상기 공유자 정보를 분석하고; 그리고

상기 데이터를 블록에 대한 상기 요청에 응답하여 상기 캐시 식별자를 처리하도록 더 구성되는 것인, 멀티 코어 아키텍처.

### 청구항 31

제20항에 있어서,

상기 컨트롤러는,

상기 디렉터리의 제1 가로 단에서, 제1 데이터 블록에 대응하는 제1 태그 식별자를 저장하고; 그리고

적어도 두 개의 세그먼트를 포함하는 공유자 표에 상기 공유자 정보를 저장하도록 구성되고,

그리고 상기 컨트롤러는,

상기 제1 데이터 블록을 저장하는 데에 유효한 캐시의 수가 상기 적어도 두 개의 세그먼트의 수의 절반보다 더 많다고 결정하고;

제2 가로 단의 제2 태그 식별자를 식별하고 - 상기 제2 태그 식별자는 제2 데이터 블록을 식별하고, 상기 제2 태그 식별자와 연관된 상태는 캐시되지 않은 또는 유효하지 않은 상태임 -; 그리고

상기 제2 가로 단으로 포인팅하는 데에 유효한 포인터와 함께 상기 제1 가로 단에서 상기 제1 데이터 블록에 대한 복제 정보를 저장하는 데에 더 유효한 것인, 멀티 코어 아키텍처.

### 청구항 32

제24항에 있어서,

상기 공유자 표에서 상기 적어도 두 개의 세그먼트의 둘 이상의 세그먼트는 동일한 크기인 것인, 멀티 코어 아키텍처.

### 발명의 설명

#### 배경 기술

[0001] 여기에서 달리 지적되지 않는다면, 본 섹션에서 설명되는 내용은 본 출원에서 청구범위에 대한 종래 기술이 아니며, 본 섹션에 포함함으로써 선행 기술로 인정되지 않는다.

[0002] 멀티 코어 프로세서 아키텍처에서, 복수의 프로세서 코어는 단일 칩 패키지에서 배열되는 복수의 직접회로 다이 위에서 또는 단일 직접회로 다이에서 포함될 수 있다. 캐시는 프로세서 코어 중 하나 이상에 의한 액세스를 위해 데이터를 저장하도록 사용될 수 있다. 데이터는 다이의 외부에 통상적으로 위치되는 더 큰 메모리에서 저장되는 데이터의 서브세트일 수 있다. 대응하는 프로세서에 대한 데이터를 저장하도록 사용되는 캐시가 각각의 프로세서 코어에게 제공될 수 있다. 데이터의 단일 조각이 복수의 캐시에 저장될 수 있음에 따라, 캐시 일관성 프로토콜(cache coherence protocol)은 복수의 캐시에서 저장되는 데이터를 추적하도록 이용될 수 있다. 캐시 내 데이터 블록에 대한 상태 정보는 디렉터리에 저장될 수 있고, 캐시 일관성 프로토콜은 적절한 데이터가 데이터 블록에 대한 요청에 응답하여 식별되고 복귀되는 것을 보장하도록 구현될 수 있다.

#### 발명의 내용

[0003] 일부 예시에서, 멀티 코어 프로세서의 디렉터리에 캐시 일관성 정보를 저장하기 위한 방법이 일반적으로 기술된다. 일부 방법은, 컨트롤러에 의해, 디렉터리에서, 멀티 코어 프로세서에서 메모리 요청을 모니터링하는 단계를 포함할 수 있다. 모니터링에 응답하여, 일부 방법은, 컨트롤러에 의해, 디렉터리에서, 데이터 블록을 식별하는 데 유효한 태그 식별자를 저장하는 단계를 포함할 수 있다. 모니터링에 응답하여, 일부 방법은, 컨트롤러에 의해, 디렉터리에서, 태그 식별자와 연관하여 상태 식별자를 저장하는 단계를 포함할 수 있다. 상태 식별자는 데이터 블록의 일관성 상태를 식별하는 데에 유효할 수 있다. 모니터링에 응답하여, 일부 방법은, 컨트롤러에 의해, 디렉터리에서, 태그 식별자와 연관하여 공유자 정보를 저장하는 단계를 포함할 수 있다. 공유자 정보는 데이터 블록을 저장하는 하나 이상의 캐시를 나타나는 데에 유효할 수 있다. 모니터링에 응답하여, 일부 방법은, 컨트롤러에 의해, 디렉터리에서, 공유자 정보와 연관하여 복제 정보를 저장하는 단계를 포함할 수 있다. 복제 정보는 디렉터리에서 공유자 정보의 복제의 유형을 나타내는 데에 유효할 수 있다.

[0004] 일부 예시에서, 멀티코어 프로세서 내 데이터 블록에 대한 요청을 프로세싱하기 위한 방법이 일반적으로 기술된다. 일부 방법은, 멀티 코어 프로세서의 캐시 일관성 디렉터리 내 컨트롤러에 의해, 요청을 수신하는 단계를 포함할 수 있다. 일부 방법은, 디렉터리 내 컨트롤러에 의해, 데이터 블록과 연관된 태그 식별자를 식별하는 단계를 포함할 수 있다. 일부 방법은, 디렉터리 내 컨트롤러에 의해, 태그 식별자와 연관된 상태 식별자를 분석하는 단계를 포함할 수 있다. 상태 식별자는 데이터 블록의 캐시 일관성 상태를 나타내는 데에 유효할 수 있다. 일부 방법은, 디렉터리 내 컨트롤러에 의해, 태그 식별자와 연관된 공유자 정보를 분석하는 단계를 포함할

수 있다. 공유자 정보는 데이터 블록을 저장하는 하나 이상의 캐시를 식별하는 데에 유효할 수 있다. 공유자 정보는 적어도 두 세그먼트를 포함하는 공유자 표에서 저장될 수 있다. 일부 방법은, 디렉터리 내 컨트롤러에 의해, 공유자 정보와 연관된 복제 정보를 분석하는 단계를 포함할 수 있다. 복제 정보는 디렉터리 내 공유자 정보의 복제의 유형을 나타내는 데에 유효할 수 있다. 일부 방법은, 디렉터리 내 컨트롤러에 의해, 공유자 표에서 하나 이상의 세그먼트의 오류를 검출하는 단계를 포함할 수 있다. 일부 방법은, 공유자 표에서 제1 세그먼트에서 오류를 검출하는 것에 응답하여, 컨트롤러에 의해, 복제 정보를 사용하여, 제1 세그먼트 내 공유자 정보를 복제하는 하나 이상의 다른 세그먼트를 식별하는 단계를 포함할 수 있다. 오류를 검출하는 것에 응답하여, 일부 방법은, 컨트롤러에 의해, 식별된 하나 이상의 다른 세그먼트로부터 공유자 정보를 분석함으로써 요청을 프로세싱하는 단계를 포함할 수 있다.

[0005] 일부 예시에서, 멀티 코어 프로세서의 디렉터리 내 캐시 일관성 정보를 저장하는 데에 유효한 멀티 코어 아키텍처가 일반적으로 기술된다. 아키텍처는 제1 타일, 제2 타일, 컨트롤러 및 메모리를 포함할 수 있다. 일부 아키텍처에서, 제1 타일은 제1 프로세서 및 제1 캐시를 포함할 수 있다. 일부 아키텍처에서, 제2 타일은 제2 프로세서 및 제2 캐시를 포함할 수 있다. 일부 아키텍처에서, 컨트롤러는 제1 타일 및 제2 타일과 통신에서 구성될 수 있다. 일부 아키텍처에서, 메모리는 컨트롤러와 통신에서 구성될 수 있다. 일부 아키텍처에서, 컨트롤러는 멀티 코어 프로세서에서 메모리 요청을 모니터링하는 데에 유효할 수 있다. 메모리 요청에 응답하여, 컨트롤러는, 디렉터리에서, 데이터 블록을 식별하는 데에 유효한 태그 식별자를 저장하는 데에 유효할 수 있다. 메모리 요청에 응답하여, 컨트롤러는, 디렉터리에서, 태그 식별자와 연관하여 상태 식별자를 저장하는 데에 유효할 수 있다. 상태 식별자는 데이터 블록의 일관성 상태를 식별하는 데에 유효할 수 있다. 메모리 요청에 응답하여, 컨트롤러는, 디렉터리에서, 태그 식별자와 연관하여 공유자 정보를 저장하는 데에 유효할 수 있다. 공유자 정보는 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효할 수 있다. 메모리 요청에 응답하여, 컨트롤러는 디렉터리에서, 태그 식별자와 연관하여 복제 정보를 저장하는 데에 유효할 수 있다. 복제 정보는 디렉터리에서 공유자 정보의 복제의 유형을 나타내는 데에 유효할 수 있다.

[0006] 이상의 요약은 단순히 예시적인 것으로서 어떠한 방식으로든 제한적으로 의도된 것이 아니다. 이하의 상세한 설명과 도면을 참조함으로써, 상기 설명된 예시적인 양태, 실시예, 그리고 특징에 더하여, 추가적인 양태, 실시예, 그리고 특징 또한 명확해질 것이다.

### 도면의 간단한 설명

[0007] 본 개시의 전술한 특징 및 다른 특징은 첨부 도면과 결합하여, 다음의 설명 및 첨부된 청구범위로부터 더욱 충분히 명백해질 것이다. 이들 도면은 본 개시에 따른 단지 몇 개의 예시를 묘사할 뿐이고, 따라서, 본 개시의 범위를 제한하는 것으로 고려되어서는 안 될 것임을 이해하면서, 본 개시는 첨부 도면의 사용을 통해 더 구체적이고 상세하게 설명될 것이다.

도 1은 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시하고,

도 2는 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시하고,

도 3은 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시하고,

도 4는 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시하고,

도 5는 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시하고,

도 6은 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하기 위한 예시적인 프로세스를 위한 흐름도를 도시하고,

도 7은 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 컴퓨터 프로그램 제품을 도시하고,

도 8은 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 배열되는 예시적인 컴퓨팅 장치를

도시하는 블록도이고, 모두 여기에서 기술된 적어도 일부 실시예에 따라 배열된다.

### 발명을 실시하기 위한 구체적인 내용

[0008] 이하의 상세한 설명에서 본 개시의 일부를 이루는 첨부된 도면이 참조된다. 문맥에서 달리 지시하고 있지 않은 한, 통상적으로, 도면에서 유사한 부호는 유사한 컴포넌트를 나타낸다. 상세한 설명, 도면, 그리고 청구범위에 설명되는 예시적인 실시예는 제한적으로 여겨지지 않는다. 본 개시에서 제시되는 대상의 범위 또는 사상에서 벗어나지 않으면서도 다른 실시예가 이용되거나, 다른 변경이 이루어질 수 있다. 여기에서 일반적으로 설명되고, 도면에 도시되는 본 개시의 양태는 다양한 다른 구성으로 배열, 대체, 조합 및 설계될 수 있음과 이 모두가 여기에서 명시적으로 고려됨이 기꺼이 이해될 것이다.

[0009] 본 개시는, 그 중에서도, 멀티 프로세서 아키텍처에서 디렉터리 오류 정정에 관련된 방법, 장치, 시스템, 기기 및 컴퓨터 프로그램 제품에 관련된다.

[0010] 간략하게 기술된 기술은 캐시 일관성 정보를 처리하고 데이터 블록에 대한 요청을 처리하는 것에 관련되는 것으로 일반적으로 설명된다. 일부 예시에서, 데이터 블록을 정의하는 데 유효한 태그 식별자를 디렉터리 내에 저장하는 단계를 포함할 수 있는 캐시 일관성 정보를 처리하기 위한 방법이 기술된다. 방법은 태그 식별자와 연관하여 상태 식별자를 저장하는 단계를 더 포함할 수 있다. 상태 식별자는 데이터 블록의 일관성 상태를 식별하는 데에 유효할 수 있다. 방법은 태그 식별자와 연관하여 공유자 정보를 저장하는 단계를 더 포함할 수 있다. 공유자 정보는 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효할 수 있다. 방법은, 디렉터리의 컨트롤러에 의해, 공유자 정보와 연관하여 복제 정보를 저장하는 단계를 포함할 수 있다. 복제 정보는 디렉터리 내의 공유자 정보의 복제의 유형을 나타내는 데에 유효할 수 있고, 복제된 세그먼트를 나타내는 데에 유효할 수 있다.

[0011] 도 1은 여기에서 기술된 적어도 일부 실시예에 따라 배열되는, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시한다. 예시적인 시스템(100)은 복수의 타일을 포함하는 다이(102)를 포함할 수 있다. 예시를 위한 타일(118)에 집중하면, 각각의 타일(118)은 캐시(110), 하나 이상의 프로세서 또는 프로세서 코어(이후 "프로세서"로 지칭됨)(112) 및/또는 디렉터리(114)를 포함할 수 있다. 프로세서(112)는 코드(이후 모두 데이터 및/또는 코드는 "데이터 블록"으로 지칭될 수 있음)를 포함하는 데이터를 처리하도록 적응될 수 있다. 캐시(110)는 프로세서(112)에 국지적인 데이터 블록을 저장하도록 구성될 수 있다. 디렉터리(114)는 디렉터리 컨트롤러(120)를 포함할 수 있다. 디렉터리 컨트롤러(120)는 여기에서 설명된 디렉터리의 동작을 제어하도록 사용될 수 있다.

[0012] 아래에서 더 기술된 바와 같이, 디렉터리 컨트롤러(120)는 디렉터리(114) 내에서 발생할 수 있는 오류를 다루도록 디렉터리(114)의 오류 정정 코드 및/또는 오류 검출 코드를 저장하도록 구성될 수 있다. 오류 정정 코드는 태그 및 정보 필드 내의 데이터와 연관하여 사용될 수 있고 오류 검출 코드는 공유자 필드 내 데이터와 연관하여 사용될 수 있다. 시간이 흘러, 다이(102)가 사용됨에 따라, 오류는 디렉터리(114)에서 발생할 수 있다. 예컨대, 하드웨어 내 물리적인 컴포넌트가 시간이 감에 따라, 제조상 결점으로 인해 절연 파괴되는 경우 하드웨어 오류가 발생할 수 있다. 소프트 오류는(예컨대, 로직 게이트가 공기 중에 방출된 입자에 의해 부딪히는 상황에서), 로직 0 값이 로직 1 값으로 잘못되게 벗어나거나 로직 1 값이 로직 0 값으로 잘못되게 벗어나는 경우와 같이 발생할 수 있다. 여기에서 논의된 오류 정정 및 검출 코드는 이러한 기술된 오류 및 다른 것을 검출 및/또는 정정하도록 사용될 수 있다.

[0013] 다이(102)는 각각의 캐시(110, 150-156)를 포함하는 타일(118, 130-136)의 매트릭스(예컨대, 어레이)를 포함할 수 있다. 각각의 타일은 각각의 프로세서(112) 및/또는 디렉터리(114) 중 하나 이상을 또한 포함할 수 있다. 다이(102) 내 각각의 타일은 동종의 배열과 같이 실질적으로 동일할 수 있거나 일부 타일은 이종의 배열과 같이 상이할 수 있다. 다이(102)는 데이터 블록이 복수의 다이 중에서 공유될 수 있도록 다른 다이(103)와 통신하여 배열될 수 있다.

[0014] 디렉터리(114)는 다이(102)의 타일에서 저장되는 각각의 데이터 블록과 연관된 위치(예컨대, 인덱스)를 식별하는 데이터 구조일 수 있다. 디렉터리(114)는 많은 또는 모든 타일 중에서 분배되거나, 다른 타일로부터 분리되어 구현되거나 또는 다이와 다이가 아닌 구조 사이에서 배치된 다이(102) 상의 단일 타일에서 위치될 수 있다. 디렉터리(114)가 예컨대, 분배되어 있으면, (0x0000-0x1000과 같은)주소의 제1 범위가 제1 타일에 저장될 수 있고, (0x1001-0x2000와 같은)주소의 제2 범위가 제2 타일에 저장될 수 있다. 도면의 디렉터리(114)는 그러므로 전체 다이 디렉터리의 제1 부분을 도시할 수 있다. 제1 부분이 타일(118)에 저장될 수 있고, 추가적인 부분이

타일(130, 131, 132 등)과 같은 다른 타일에 저장될 수 있다.

[0015] 디렉터리(114)는 데이터 블록을 공유할 수 있는 캐시를 나타내는 데에 유효한 공유자 정보 및 데이터 블록의 상태에 따라, 다이(102)에 저장된 데이터 블록을 인식하는 태그를 포함할 수 있다. 예컨대, 디렉터리(114)는 MESI 캐시 일관성 프로토콜 또는 MOESI, MSI, MOSI 등과 같은 자신의 변형을 사용할 수 있다. MESI 프로토콜에서, 디렉터리(114)는,

[0016] 블록이 메인 메모리로부터 검색됨으로써, 데이터 블록이 수정되는 경우 또는 데이터 블록이 메인 메모리에 매치되지만 하나의 캐시에 있는 경우와 같은, 배타적으로 수정됨(exclusive modified)(“EM”);

[0017] 데이터 블록이 하나 이상의 캐시에 저장될 수 있고 데이터 블록이 메인 메모리에 매치되는 경우와 같은, 공유됨(shared)(“S”);

[0018] 데이터 블록이 유효하지 않은 경우와 같은, 유효하지 않음(invalid)(“I”);

[0019] 데이터 블록이 캐시되지 않은 경우와 같은, 캐시되지 않음(uncached)(“U”); 및/또는

[0020] 나타난 데이터 블록이 데이터 블록의 가상 최근의 복사본을 포함하는 경우와 같은, 소유됨(owned)(“0”)를 포함하는, 데이터 블록에 대한 공유 상태를 추적할 수 있다.

[0021] 도 2는, 여기에서 기술된 적어도 일부 실시예에 따라 배열되는, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시한다. 도 2는 추가적인 세부 사항과 함께, 시스템(100)에 실질적으로 유사하다. 도 1의 컴포넌트에 동일하게 표시된 도 2의 그러한 컴포넌트는 명확성의 목적을 얻기 위해 다시 설명되지 않을 것이다.

[0022] 디렉터리(114)는 다이 내 메모리 요청을 모니터링하고 캐시에 저장된 데이터 블록에 관한 일관성 정보를 저장하도록 구성될 수 있다. 예컨대, 디렉터리 컨트롤러(120)는 디렉터리(114) 내에 태그/상태 표(170), 세그먼트 또는 세트 인덱스 표(172), 및/또는 다이(102) 내 캐시에 저장된 데이터 블록에 관한 공유자 표(186)을 저장하도록 구성될 수 있다. 태그/상태 표(170), 세그먼트 또는 세트 인덱스 표(172) 및 공유자 표(186)는, 조합하여, 디렉터리(114)를 형성할 수 있다. 디렉터리(114) 내 화살표는 표(170, 172, 186) 중 하나 이상에 속하는 둘 이상의 세로 단으로 분할될 수 있다. 태그/상태 표(170)는 태그(174), 상태(176) 및/또는 오류 정정 코드(error correction code)(“ECC”)와 같은 필드 내 정보를 포함할 수 있다. 오류 정정 코드(178)는 가로 열 태그 및 상태에 일대일 대응할 수 있거나 태그 및 상태에 일대다 대응할 수 있다. 태그 필드(174), 상태 필드(176) 및 오류 정정 코드 필드(178)는 서로와 연관될 수 있다. 태그 필드(174)는 다이(102)에 저장된 데이터의 블록의 표시를 포함할 수 있다. 상태 필드(176)는 전술된 프로토콜 중 하나를 사용하여 이러한 데이터 블록의 일관성 상태를 나타낼 수 있다. 오류 정정 코드(178)는 태그 필드(174) 및/또는 상태 필드(176)의 엔트리 내 오류를 정정하도록 사용될 수 있는 코드를 포함할 수 있다. 오류 정정 코드(178)는 태그 필드(174) 및/또는 상태 필드(176) 내 비트를 복제하도록 사용될 수 있다. 오류가 발생하는 경우, 코드(178)는, 오류가 발생됐다고 검출하고, 오류와 함께 비트를 식별하고, 오류를 정정하기 위해 해당되는 비트를 플립하도록 사용될 수 있도록 충분한 비트가 복제될 수 있다. 오류 정정 코드(178)는 비트 상호배치(bit interleaving)을 사용할 수 있다.

[0023] 세그먼트 또는 세트 인덱스 표(172)는 유형 필드(180), 세그먼트 또는 세트 인덱스 값 필드(182) 및 오류 정정 코드/오류 검출 코드 필드(183)와 같은 정보를 포함할 수 있다. 세그먼트 또는 세트 인덱스 표(172)의 가로 단의 데이터 블록은 태그/상태 표(170) 내 대응하는 가로 단의 데이터 블록과 연관하여 디렉터리 컨트롤러(120)에 의해 저장될 수 있다. 세그먼트 또는 세트 인덱스 표(172)는 세그먼트 또는 세트 인덱스 표(172) 내 오류를 검출 및/또는 정정하도록 이용될 수 있는 오류 정정 코드 또는 오류 검출 필드(183)를 또한 포함할 수 있다. 여기에서 논의된 바와 같이, 유형 필드(180)는 공유자 표(186)에 저장된 데이터 블록에 대한 디렉터리 컨트롤러(120)에 의해 사용될 복제의 유형을 식별하는 복제 정보를 포함할 수 있다. 예시에서, 네 개의 다른 복제 유형은 공유자 정보가 공유자 표(186)의 하나 이상의 세그먼트에 저장되는 경우 사용될 수 있다. 공유자 정보는 공유자 표(186)의 가로 단 및/또는 공유자 표(186)의 하나 이상의 세그먼트에 관련될 수 있다.

[0024] “UR”은 공유자 정보가 복제되지 않은 경우, 복제되지 않음(Un-Replicated)에 대응할 수 있고,

[0025] “SR”은 둘 이상의 세그먼트에 저장된 공유자 정보가 동일한 디렉터리 가로 단의 다른 세그먼트에 복제되는 경우, 세그먼트 복제됨(Segment-Replicated)에 대응할 수 있고,

[0026] “ER”은 공유자 정보의 전체 가로 단이 다른 가로 단에서 복제되는 경우, 엔트리 복제됨(Entry-Replicated)에 대

옹할 수 있고,

- [0027] "FR"은 하나의 세그먼트에 저장된 공유자 정보가 동일한 디렉터리 가로 단의 모든 다른 세그먼트에서 복제되는 경우, 전체 세그먼트 복제됨(Full Segment-Replicated)에 대응할 수 있다.
- [0028] 공유자 정보는 공유자 정보와 연관된 태그 식별자에 대응하는 데이터 블록을 저장할 수 있는 다이의 캐시를 식별하는 데에 유용할 수 있다. 공유자 정보는, 이에 제한되지는 않으나, 다음의 형식을 포함하는, 하나 이상의 형식에서 공유자 표(186)의 하나 이상의 세그먼트에서 저장될 수 있다.
- [0029] 전체 비트 벡터(Full bit vector) - 전체 비트 벡터 형식에서, 공유자 세그먼트의 각각의 비트는 하나의 캐시(110)에 대응할 수 있다.
- [0030] 포인터(Pointer) - 포인터 형식에서, 공유자 세그먼트의 둘 이상의 비트는 하나의 캐시를 식별할 수 있고, 각각의 캐시는 공유자 세그먼트의 둘 이상의 비트의 값의 유일한 조합에 의해 식별될 수 있다. 예컨대, "00"은 캐시(110)에 대응할 수 있고, "01"은 캐시(150)에 대응할 수 있고, "10"은 캐시(151)에 대응할 수 있고, "11"은 캐시(152)에 대응할 수 있다.
- [0031] 코스 비트 벡터(Coarse bit vector) - 코스 비트 벡터 형식에서, 공유자 세그먼트의 각각의 비트는 둘 이상의 캐시의 그룹을 식별할 수 있다. 예컨대, 제1 비트는 캐시(110 및 150)를 포함하는 그룹에 대응할 수 있고, 제2 비트는 캐시(151 및 152)을 포함하는 그룹에 대응할 수 있다.
- [0032] 공유자 표(186)는 하나 이상의 세그먼트 필드(190, 192, 194, 196)(세그먼트 0, 1, 2 및 3은 도면에 도시됨)와 둘 이상의 패러티 필드(200, 202, 204, 206)로 분할될 수 있다. 세그먼트의 크기 및 수는 위의 형식 중 하나 이상에 따라 캐시의 식별자를 인코딩하도록 사용되는 비트의 수에 기초하여 정의될 수 있다. 패러티 필드(200, 202, 204, 206)는 대응하는 세그먼트의 잘못된 또는 오류가 없는 데이터 블록을 나타내도록 사용될 수 있다.
- [0033] 데이터 블록은 공유자 데이터 블록을 복제하는 데에 유효한 방식에서 디렉터리 컨트롤러(120)에 의해 공유자 표(186)에서 저장될 수 있다. 복제는 공유자 정보의 오류를 검출하고 정정하도록 사용될 수 있다. 복제를 통해, 공유자 표(186)는 패러티 필드(200, 202, 204, 206)의 형식에서 오류 정정 코드를 사용하여 오류 검출 및 정정을 고려할 수 있다.
- [0034] 예컨대, 관계가 있는 공유자 정보가 세그먼트 0에 저장되는 경우, 그 동일한 공유자 정보는 세그먼트 1, 2 및 3에서 복제될 수 있다. 공유자 정보가 하나의 캐시가 태그 식별자와 연관된 데이터의 블록을 공유하는 중이라고 나타내는 경우, 그 하나의 캐시에 대한 식별자는 하나의 세그먼트(예컨대, 세그먼트 0)에 저장될 수 있다. 디렉터리 컨트롤러(120)는 세그먼트(1, 2 또는 3)와 같은, 다른 세그먼트의 캐시 식별자를 복제하도록 구성될 수 있다. 블록이 하나 또는 두 개의 캐시에만 공유되는 예시에서, 디렉터리 컨트롤러(120)는 세그먼트의 제1 세트에서 그러한 캐시에 대한 식별자를 저장하고, 세그먼트의 제2 세트에서 그러한 식별자를 복제하도록 구성될 수 있다. 디렉터리 컨트롤러(120)는 공유자 정보가 복제될 수 있는, 공유자 표(186)의 다른 위치의 식별을 표(172)에 저장하는 데에 더 유효할 수 있다. 공유된 표(186)의 이러한 다른 위치는, 요청된 블록을 저장하는 캐시의 식별이 오류가 공유자 표의 하나 이상의 세그먼트에서 검출되는 경우 발견될 수 있도록 오류의 이벤트에서, 디렉터리 컨트롤러(120)에 의해 분석될 수 있다.
- [0035] 도 3은, 여기에서 기술된 적어도 일부 실시예에 따라 배열되는, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시한다. 도 3은 추가적인 세부 사항과 함께 시스템(100)에 실질적으로 유사하다. 도 1 및 2의 컴포넌트에 동일하게 표시된 도 3의 그러한 컴포넌트는 명확성의 목적을 위해 다시 설명되지 않을 것이다.
- [0036] 도시된 예시에서, 블록 "Z"는 배타적으로 수정된 상태("EM")에서 저장되는 것을 나타내므로, 블록 Z는 오직 하나의 캐시에 의해 저장된다. 디렉터리 컨트롤러(120)는 이러한 EM 상태를 판독하고 "FR"이 사용될 수 있다고 결정하는 데에 유효하게 구성될 수 있다. 오직 하나의 캐시에 대응하는 식별자가 공유자 표(186)에 저장되어야 하고 그러한 식별자의 비트의 수의 크기가 공유자 표의 세그먼트의 크기보다 작을 수 있으므로, FR이 사용될 수 있다. 디렉터리 컨트롤러(120)는 공유자 표(186)의 블록 Z("111110")을 저장하는 캐시에 대응하는 식별자를 저장하는 데에 유효하도록 구성될 수 있다. 디렉터리 컨트롤러(120)는 블록 Z에 대한 태그로서 동일한 가로 단에서 공유자 표(186)의 모든 세그먼트에서 "111110"을 복제하도록 구성될 수 있다. 디렉터리 컨트롤러(120)는 블록 Z와 연관된 동일한 가로 단에서 유형 필드(180)에 "FR"을 저장하는 데에 유효할 수 있다.
- [0037] 도시된 예시와 계속하여, 블록 A가 하나의 캐시보다 더 많은 캐시에 의해 공유되는 것으로("S") 나타난다. 예

컨대, 그러한 캐시에 대한 식별자를 위한 인코딩이 이용 가능한 세그먼트의 수의 절반의 크기에 동일하거나 그보다 작은 키기를 갖도록 복수의 코드에 의해 저장될 수 있다. 식별자의 크기가 이용 가능한 세그먼트의 수의 절반보다 더 적은 수에 대응하므로, 세그먼트 복제 SR이 사용될 수 있다. 디렉터리 컨트롤러(120)는 블록 A를 저장하는 두 개의 캐시에 대한 식별자를 저장하도록 구성될 수 있다. 오직 두 개의 세그먼트(예컨대, 세그먼트 0 및 1)이 공유자 표시를 저장하도록 사용되고, 두 개의 다른 세그먼트(세그먼트 2 및 3)가 공유자 표시의 복제를 위해 이용 가능하다.

[0038] 디렉터리 컨트롤러(120)는 공유자 표(186)의 블록 A를 저장하는 것으로 표시되는 캐시에 대응하는 식별자를 저장하도록 적응될 수 있다. 디렉터리 컨트롤러(120)는 공유자 데이터 블록이 복제될 수 있는 공유자 표(186)의 다른 위치를 포인팅하는 세그먼트/세트 인덱스 표(172)에 포인터를 저장하도록 구성될 수 있다. 예시에서, 세그먼트 또는 세트 인덱스 표(172)는 공유자 표(186)의 세그먼트 0이 세그먼트 2에 대한 복제된 정보를 포함하는 것을 나타내는 포인터를 포함할 수 있다. "세그 2"는 제1 또는 세그먼트 0 슬롯의 세그먼트/세트 인덱스 표(172)에 도시된다. 유사하게, 세그먼트 또는 세트 인덱스 표(172)는 공유자 표(186)의 세그먼트 1이 세그먼트 3에 대한 복제된 정보를 포함하는 것을 나타내는 포인터를 포함할 수 있다. "세그 3"이 제2 또는 세그먼트 1 슬롯의 세그먼트/세트 인덱스 표(172)에 도시된다. 세그먼트 또는 세트 인덱스 표(172)는 공유자 표(186)의 세그먼트 2가 세그먼트 0에 대한 복제된 정보를 포함하는 것을 나타내는 포인터를 포함할 수 있다. "세그 0"이 제3 또는 세그먼트 2 슬롯의 세그먼트/세트 인덱스 표(172)에 도시된다. 세그먼트/세트 인덱스 표(172)는 공유자 표(186)의 세그먼트 3이 세그먼트 1에 대한 복제된 정보를 포함하는 것을 나타내는 포인터를 포함한다. "세그 1"은 제4 또는 세그먼트 3 슬롯의 세그먼트/세트 인덱스 표(172)에 도시된다. 기술된 예시에서, 블록 A는 두 개의 캐시에 의해 저장될 수 있다. 제1 캐시에 대응하는 식별자는 세그먼트 0 및 2에 대한 슬롯의 공유자 표(186)에 저장될 수 있다. 제2 캐시에 대한 식별자는 세그먼트 1 및 3에 대한 슬롯의 공유자 표(186)에 저장될 수 있다.

[0039] 도 4는, 여기에서 기술된 적어도 일부 실시예에 따라 배열된, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시한다. 도 4는 추가적인 세부 사항과 함께, 시스템(100)과 실질적으로 유사하다. 도 1, 2 및 3의 컴포넌트에 동일하게 표시된 도 4의 그러한 컴포넌트는 명확성의 목적을 위해 다시 설명되지 않을 것이다.

[0040] 세그먼트 복제("SR") 또는 전체 복제("FR")가 사용되는 예시에서, 디렉터리 컨트롤러(120)는 복제된 데이터 블록을 사용하여 오류 검출을 또한 수행할 수 있다. 오류 검출은, 오류의 하나 이상의 비트가 검출될 수 있도록 저장된 식별자에 대한 값을 비교하여 디렉터리 컨트롤러(120)에 의해 수행될 수 있다. 디렉터리 컨트롤러(120)는 공유자 표(186)의 적어도 두 개의 다른 위치에 저장되는 캐시 식별자의 값을 사용하도록 적응될 수 있다. 값이 적어도 두 개의 다른 위치에 저장되는 경우, 그의 값이 오류가 아닐 가능성이 크다. 예시에서, 블록 Z에 대해, 전체 복제 "FR"은 세그먼트 또는 세트 인덱스 표(172)의 유형 필드(180)에서 표시된다. 디렉터리 컨트롤러(120)는 공유자 표(186)를 분석하여 필드(190, 192 및 194)가 "111110"의 캐시 식별자를 모두 가진다고 결정할 수 있다. 캐시 식별자 "111110"이 적어도 두 개의 다른 위치에서 저장됨에 따라, 값이 맞을 확률이 높다. 게다가, 필드(196)가 다른 값 "000000"을 나타냄에 따라, 필드(196)의 값은 틀릴 확률이 높다. 디렉터리 컨트롤러(120)은 필드(196)의 값이 "1111110"을 나타내도록 변경하도록 구성될 수 있다.

[0041] 도 3을 다시 참조하여, 블록 P가 하나의 캐시보다 더 많은 캐시에 의해 저장되는 것으로 표시된다. 예시에서, 블록 P는, 블록 P를 저장하는 캐시에 대한 식별자가 세그먼트 절반의 조합된 크기보다 더 큰 크기일 수 있으므로 동일한 가로 단의 세그먼트에 의해 수행되지 않을 수 있도록 다수의 캐시에 의해 공유될 수 있다. 예컨대, 블록 P는 세그먼트의 수의 절반보다 많은 수에 대응하는 복수의 캐시에 저장될 수 있다. 기술된 예시에서, 디렉터리 컨트롤러(120)는 블록 C가 캐시되지 않음("U")으로 나타난다고 결정할 수 있다. 블록 C는 캐시되지 않음에 따라, 블록 C와 연관된 가로 단의 공유자 표(186)의 저장 공간이 사용되는 것이 이용 가능하다. 디렉터리 컨트롤러(120)는 블록 P와 같은, 다른 블록에 대한 공유자 정보를 저장하기 위해 공유자 표(186)의 블록 C와 연관된 가로 단의 저장 공간을 사용하도록 구성될 수 있다. 디렉터리 컨트롤러(120)는 블록 P에 대한 공유자 정보가 공유자 표(186)의 제2 및 제3 가로 단에서 복제되는 것을 나타내는 표(172)의 표시를 저장하도록 구성될 수 있다. 공유자 표(186)의 제2 및 제3 가로 단은 블록 P 및 블록 C에 대한 가로 단에 대응한다. 디렉터리 컨트롤러(120)는 블록 P와 연관된 복제의 유형이 ER - 엔트리 복제라는 것을 표(172)에서 나타낼 수 있다. 디렉터리 컨트롤러(120)는 제2 가로 단에 대한 데이터 블록이 제3 가로 단에서 복제되는 것을 나타내는 포인터를 표(172)의 제2 가로 단에서 저장할 수 있다. 유사하게, 디렉터리 컨트롤러(120)는 가로 단 3에 대한 데이터 블록이 가로 단 2에서 복제되는 것을 나타내는 포인터를 표(172)의 제3 가로 단에서 저장하도록 구성될 수 있다.

- [0042] 도시된 예시에서, 블록 B가 세그먼트 복제를 사용하기에는 너무 큰 캐시 식별자에 대응하여, 너무 많은 캐시에 의해 공유되는 것으로 나타난다. 예컨대, 블록 B는 세그먼트 수의 절반의 조합된 크기보다 더 큰 캐시 식별자에 대응하는 캐시의 수에서 공유되는 것으로 도시된다. 나아가, 상태 식별자가 "U"인 이용 가능한 가로 단이 없으므로, 공유자 표에서 다른 가로단이 이용 가능하지 않은 것으로 보인다. 디렉터리 컨트롤러(120)는 블록 B로 복제되지 않거나 복제 없음 "UR"을 할당하도록 구성될 수 있다. 블록 B에 대한 데이터를 공유하는 것에 관한 디렉터리 컨트롤러에 의해 오류가 검출되는 경우, 캐시 일관성 오류 복구 프로토콜이 사용될 수 있다.
- [0043] 도 5는, 여기에서 기술된 적어도 일부 실시예에 따라 배열되는, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 예시적인 시스템을 도시한다. 도 5는 추가적인 세부 사항과 함께, 시스템(100)에 실질적으로 유사하다. 도 1, 2, 3 및 4의 컴포넌트에 동일하게 표시된 도 5의 그러한 컴포넌트는 명확성의 목적을 위해 다시 기술되지 않을 것이다.
- [0044] 예시에서, 블록 B는 복제되지 않은 것으로 나타난다. 블록 B에 대한 공유자 데이터에 관한 디렉터리 컨트롤러(120)에 의해 오류는 검출되는 경우, 캐시 일관성 오류 복구 프로토콜이 사용될 수 있다. 그러한 캐시 일관성 오류 복구 프로토콜에서, 디렉터리 컨트롤러(120)는 다이(102)의 모든 캐시로 브로드캐스트 메시지(210)를 발송하도록 구성될 수 있다. 메시지는 특정 블록의 위치를 요청하는 요청을 포함할 수 있다. 다이(102) 내 타일은 응답 메시지(212)와 함께 브로드캐스트 메시지(210)에 응답하도록 구성될 수 있다. 응답 메시지(212)는 각각의 캐시가 블록을 저장하고 있는지 나타낼 수 있다. 디렉터리 컨트롤러(120)는 응답 메시지(212)를 수신하고 따라서 디렉터리(114)의 공유자 표(186)을 업데이트하도록 구성될 수 있다.
- [0045] 도 3을 다시 참조하여, 예시에서, 디렉터리 컨트롤러(120)는 타일로부터 요청을 수신하도록 구성될 수 있다. 요청은 다이(102) 내의 캐시 중 하나에 저장될 수 있는 특정 블록에 대한 것일 수 있다. 디렉터리 컨트롤러(120)는 디렉터리(114)의 특정 가로 단을 식별하기 위해 태그/상태 표(170)를 분석하도록 구성될 수 있다. 특정 가로 단은 특정 블록과 연관된 태그를 저장하는 중일 수 있다. 특정 가로 단이 식별되면, 디렉터리 컨트롤러(120)는 특정 블록의 캐시 일관성 상태를 식별하도록 가로 단의 상태 필드를 분석하도록 구성될 수 있다. 디렉터리 컨트롤러(120)는 특정 블록에 대한 공유자 정보에 대해 사용될 수 있는 복제의 유형을 결정하도록 특정 가로 단에 대해 유형 필드(180)를 분석하도록 구성될 수 있다. 디렉터리 컨트롤러(120)는 특정 블록을 저장 중일 수 있는 캐시 또는 캐시들의 식별자를 결정하도록 공유자 필드(186)를 분석할 수 있다. 대응하는 패러티 필드(200, 202, 204, 206)가 공유자 정보에 오류가 없다고 나타내면, 디렉터리 컨트롤러(120)는 특정 블록을 복구하도록 검출된 캐시 식별자를 사용할 수 있다.
- [0046] 대응하는 패러티 필드가 공유자 정보가 유효하지 않다고 나타내면, 디렉터리 컨트롤러(120)는 공유자 정보에 대한 공유자 표(186)의 다른 위치를 분석할 수 있다. 다른 위치는 세그먼트 또는 세트 인덱스 표(172)에서 식별된 복제의 유형에 기초할 수 있다. 세그먼트 복제 "SR"가 유형 필드(180)에서 식별되면, 디렉터리 컨트롤러(120)는 표(172)의 포인터에 의해 식별된 공유자 표(186)의 특정 가로 단의 다른 세그먼트를 분석할 수 있다. 전체 세그먼트 복제 "FR"가 유형 필드(180)에서 식별되는 경우, 디렉터리 컨트롤러(120)는 특정 가로 단의 다른 세그먼트를 또한 복제할 수 있다. 엔트리 복제 "ER"이 유형 필드(180)에서 식별되는 경우, 디렉터리 컨트롤러(120)는 표(172)의 포인터에 의해 식별되는 공유자 표(186)의 다른 가로 단을 분석할 수 있다. 공유자 표(186)의 데이터 블록의 모두가 유효하지 않은 패러티 상태를 나타내거나 복제 유형이 복제되지 않음 "UR"을 나타내는 예시에서, 디렉터리 컨트롤러(120)는 특정 블록의 위치를 식별하도록 전술된 캐시 일관성 오류 복구 프로토콜을 사용할 수 있다. 복구 컨트롤러(120)는 특정 블록을 요청하는 다이에서 모든 타일에 대한 요청을 발송하도록 구성될 수 있다. 특정 블록의 위치가 결정되면, 디렉터리 컨트롤러(120)는 정정 공유자 데이터 블록과 함께 공유자 표(186)의 대응하는 세그먼트 또는 세그먼트들을 겹쳐 쓰도록 구성될 수 있다.
- [0047] 패러티 필드(200, 202, 204, 206)는 일관성 이벤트에 기초하여 디렉터리 컨트롤러(120)에 의해 수정될 수 있다. 일관성 이벤트의 예시는 디렉터리가 블록이 요청 캐시에 의해 공유되는 블록을 관리하는 것을 캐시가 요청하는 것일 수 있다. 일관성 이벤트의 다른 예시는 캐시가 블록으로 기록하라고 요청하고 디렉터리가 블록이 기록하기 위한 캐시 요청에 의해 저장되지 않은 것을 나타내는 것이다.
- [0048] 다른 가능한 이점 중에서, 본 개시에 따른 시스템은 스토리지 오버헤드의 많은 양을 반드시 사용할 필요 없이, 오류 검출과 함께 캐시 일관성 프로토콜에 기초하여 디렉터리의 사용을 고려할 수 있다. 복수의 비트 오류 정정 및 검출은 멀티 비트 오류가, 블록을 저장하는 캐시를 틀리게 식별하여, 검출되고 정정될 수 있도록 구현될 수 있다. 오류 정정 코드는 디렉터리(114)의 모든 데이터 블록에 대해 사용될 필요가 없고 이에 의해 디렉터리(114)에 의해 사용되는 공간의 양을 줄인다. 예컨대, 패러티 필드의 형식에서, 오류 검출 코드는 복제가 이용

가능할 수 있기 때문에 오류를 정정하기에 충분할 수 있다. 오류 정정 코드는 태그 및 상태 필드에 대해 사용될 수 있다. 오류 검출 코드는 공유자 필드에 대해 사용될 수 있다.

[0049] 도 6은, 여기에서 기술되는 적어도 일부 실시예에 따라 배열되는, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하기 위한 예시적인 프로세스를 위한 흐름도를 도시한다. 일부 예시에서, 도 6의 프로세스는 전술된 시스템(100)을 사용하여 구현될 수 있고, 멀티 코어 프로세서의 디렉터리의 캐시 일관성 정보를 저장하기 위해 사용될 수 있다.

[0050] 예시적인 프로세스는 블록(S2, S4, S6, S8 및/또는 S10) 중 하나 이상에 의해 도시되는 하나 이상의 동작, 작동 또는 기능을 포함할 수 있다. 분리된 블록으로 도시되었으나, 요구되는 구현예에 따라, 다양한 블록이 추가적인 블록으로 분할될 수 있고, 더 적은 블록으로 조합될 수 있거나 제거될 수 있다.

[0051] 프로세싱은 블록(S2)에서, "컨트롤러에 의해, 디렉터리에서, 멀티 코어 프로세서의 메모리 요청을 모니터링함"에서 시작할 수 있다. 블록(S2)에서, 컨트롤러는 메모리 요청을 모니터링할 수 있다.

[0052] 프로세싱은 블록(S2)에서 블록(S4), "컨트롤러에 의해, 디렉터리에서, 데이터 블록을 식별하는 데에 유효한 태그 식별자를 저장함"으로 계속할 수 있다. 블록(S4)에서, 컨트롤러는 데이터 블록을 식별하는 데에 유효한 태그 식별자를 저장하도록 (예컨대, 실행가능 명령어를 통해) 구성될 수 있다.

[0053] 프로세싱은 블록(S4)에서 블록(S6), "컨트롤러에 의해, 디렉터리에서, 태그 식별자와 연관하여, 데이터 블록의 일관성 상태를 식별하는 데에 유효한 상태 식별자를 저장함"으로 계속할 수 있다. 블록(S4)에서, 디렉터리 컨트롤러는 태그 식별자와 연관하여 상태 식별자를 저장하는 데에 유효하도록 구성될 수 있다. 상태 식별자는 태그에 의해 식별된 블록의 일관성 상태를 식별할 수 있다.

[0054] 프로세싱은 블록(S6)에서 블록(S8), "컨트롤러에 의해, 디렉터리에서, 태그 식별자와 연관하여, 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유효한 공유자 정보를 저장함"으로 계속할 수 있다. 블록(S8)에서, 디렉터리 컨트롤러는 데이터 블록을 저장 중일 수 있는 캐시를 나타내는 공유자 정보를 저장하도록 구성될 수 있다.

[0055] 프로세싱은 블록(S8)에서 블록(S10), "컨트롤러에 의해, 디렉터리에서, 공유자 정보와 연관하여, 디렉터리 내의 공유자 정보의 복제의 유형을 나타내는 데에 유효한 복제 정보를 저장함"으로 계속할 수 있다. 블록(S10)에서, 디렉터리 컨트롤러는 공유자 정보와 연관하여 복제 정보를 저장하는 데에 유효할 수 있다. 복제 정보는 디렉터리의 공유자 정보의 복제의 유형을 나타낼 수 있다. 예컨대, 공유자 정보는 동일한 가로 단의 복수의 세그먼트에서 복제될 수 있거나, 공유자 정보는 다른 가로 단에서 복제될 수 있거나 또는 공유자 정보는 복제되지 않을 수 있다.

[0056] 특정 블록에 대한 공유자 정보가 유효하지 않은 것으로 결정되는 예시에서, 디렉터리 컨트롤러는 복제 정보를 사용할 수 있다. 복제 정보는 공유자 정보를 복제하는 공유자 표의 하나 이상의 세그먼트를 식별할 수 있다. 프로세서는 하나 이상의 세그먼트로부터 공유자 정보를 분석함으로써 요청을 프로세스할 수 있다.

[0057] 도 7은, 여기에서 기술된 적어도 일부 실시예에 따라 배열되는, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을 구현하도록 이용될 수 있는 컴퓨터 프로그램 제품(300)을 도시한다. 프로그램 제품(300)은 신호 베어링 매체(302)를 포함할 수 있다. 신호 베어링 매체(302)는, 예컨대, 프로세서에 의해 실행되는 경우, 도 1-6에 관해 전술된 기능을 제공할 수 있다. 그러므로, 예컨대, 시스템(100)을 참조하여, 타일(118, 130-144)의 프로세서(112) 중 하나 이상은 매체(302)에 의해 시스템(100)으로 전달되는 명령어(304)에 응답하여 도 7에 도시된 블록 중 하나 이상을 수행할 수 있다.

[0058] 일부 구현예에서, 신호 베어링 매체(302)는 하드 디스크 드라이브, CD(Compact Disk), DVD(Digital Video Disk), 디지털 테이프, 메모리 등과 같은 컴퓨터 판독 가능 매체(306)를 포함할 수 있으나, 이에 제한되지는 않는다. 일부 구현예에서, 신호 베어링 매체(302)는 메모리, 읽기/쓰기(R/W) CD, R/W DVD 등과 같은 기록 가능 매체(308)를 포함할 수 있으나, 이에 제한되지는 않는다. 일부 구현예에서, 신호 베어링 매체(302)는 디지털 및/또는 아날로그 통신 매체(예컨대, 광섬유 케이블, 도파관(waveguide), 유선 통신 링크, 무선 통신 링크 등)와 같은 통신 매체(310)를 포함할 수 있으나, 이에 제한되지는 않는다. 따라서, 예컨대, 프로그램 제품(300)은, 신호 베어링 매체(302)가 무선 통신 매체(310)(예컨대, IEEE 802.11 표준에 따르는 무선 통신 매체)에 의해 전달되는 RF 신호 베어링 매체(302)에 의하여 시스템(100)의 하나 이상의 모듈로 전달될 수 있다.

[0059] 도 8은, 여기에서 기술된 적어도 일부 실시예에 따라, 멀티 코어 프로세서 아키텍처에서 디렉터리 오류 정정을

구현하도록 배열되는 예시적인 컴퓨팅 장치(400)를 도시하는 블록도이고, 모두 여기에서 기술된 적어도 일부 실시예에 따라 배열된다. 매우 기본적인 구성(402)에서, 컴퓨팅 장치(400)는 하나 이상의 프로세서(404) 및 시스템 메모리(406)을 전형적으로 포함한다. 메모리 버스(408)는 프로세서(404)와 시스템 메모리(406) 사이에서 통신하기 위해 사용될 수 있다.

[0060] 요구되는 구성에 따라, 프로세서(404)는 마이크로프로세서( $\mu$ P), 마이크로컨트롤러( $\mu$ C), 디지털 신호 프로세서(DSP) 또는 그 임의의 조합을 포함하는 임의의 유형일 수 있지만, 이에 한정되는 것은 아니다. 프로세서(404)는 레벨 1 캐시(410) 및 레벨 2 캐시(412)와 같은 하나 이상의 레벨의 캐싱, 프로세서 코어(414) 및 레지스터(416)를 포함할 수 있다. 예시적인 프로세서 코어(414)는 ALU(arithmetic logic unit), FPU(floating point unit), DSP 코어(digital signal processing core), 또는 그 임의의 조합을 포함할 수 있다. 예시적인 메모리 컨트롤러(418)는 또한 프로세서(404)와 사용될 수 있거나, 또는 몇몇 구현예에서, 메모리 컨트롤러(418)는 프로세서(404)의 내부 부품일 수 있다.

[0061] 요구되는 구성에 따라, 시스템 메모리(406)는 (RAM과 같은) 휘발성 메모리, (ROM, 플래시 메모리 등과 같은) 비휘발성 메모리, 또는 그 임의의 조합을 포함할 수 있지만, 이에 한정되지 않는 임의의 유형일 수 있다. 시스템 메모리(406)는 운영 체제(420), 하나 이상의 애플리케이션(422), 및 프로그램 데이터(424)를 포함할 수 있다. 애플리케이션(422)은 도 1-3의 시스템(100)에 관해 기술되는 것을 포함하는 여기에서 기술된 기능을 수행하도록 배열되는 디렉터리 오류 정정 알고리즘(426)을 포함할 수 있다. 프로그램 데이터(424)는 여기에서 기술된 바 같이, 멀티 코어 프로세서 아키텍처의 디렉터리 오류 정정 알고리즘을 구현하는 데에 유효할 수 있는 디렉터리 오류 정정 데이터(428)를 포함할 수 있다. 일부 실시예에서, 애플리케이션(422)은 멀티 코어 프로세서 아키텍처의 디렉터리 오류 정정이 제공될 수 있도록 운영체제(420) 상의 프로그램 데이터(424)와 동작하도록 배열될 수 있다. 이러한 기술된 기본 구성(402)은 파선 내의 컴포넌트에 의해 도 8에 도시된다.

[0062] 컴퓨팅 장치(400)는 추가적인 특징 또는 기능, 및 기본 구성(402)과 임의의 요구되는 장치와 인터페이스 간 통신을 용이하게 하기 위한 추가적인 인터페이스를 가질 수 있다. 예를 들면, 버스/인터페이스 컨트롤러(430)는 저장 인터페이스 버스(434)를 통한 기본 구성(402)과 하나 이상의 데이터 저장 장치(432) 간의 통신을 용이하게 하는데 사용될 수 있다. 데이터 저장 장치(432)는 분리형 저장 장치(436), 비분리형 저장 장치(438), 또는 그들의 조합일 수 있다. 분리형 저장 장치 및 비분리형 저장 장치의 예로는, 몇 가지 말하자면, 플렉서블 디스크 드라이브 및 하드 디스크 드라이브(HDD)와 같은 자기 디스크 장치, 컴팩트 디스크(CD) 드라이브 또는 디지털 다기능 디스크(DVD) 드라이브와 같은 광 디스크 드라이브, 고체 상태 드라이브(solid state drive; SSD), 및 테이프 드라이브가 포함된다. 예시적인 컴퓨터 저장 매체는, 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 다른 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성의, 분리형 및 비분리형 매체를 포함할 수 있다.

[0063] 시스템 메모리(406), 분리형 저장 장치(436) 및 비분리형 저장 장치(438)는 모두 컴퓨터 저장 매체의 예이다. 컴퓨터 저장 매체는 RAM, ROM, EEPROM, 플래시 메모리 또는 다른 메모리 기술, CD-ROM, 디지털 다기능 디스크(DVD) 또는 다른 광학 저장 장치, 자기 카세트, 자기 테이프, 자기 디스크 저장 장치 또는 다른 자기 저장 장치, 또는 원하는 정보를 저장하는데 사용될 수 있고 컴퓨팅 장치(400)에 의해 액세스될 수 있는 임의의 다른 매체를 포함하지만, 이에 한정되는 것은 아니다. 그러한 임의의 컴퓨터 저장 매체는 장치(400)의 일부일 수 있다.

[0064] 컴퓨팅 장치(400)는 버스/인터페이스 컨트롤러(430)를 통한 다양한 인터페이스 장치(예를 들면, 출력 장치(442), 주변 인터페이스(444) 및 통신 장치(446))로부터 기본 구성(402)으로의 통신을 용이하게 하기 위한 인터페이스 버스(440)도 포함할 수 있다. 예시적인 출력 장치(442)는 그래픽 처리 유닛(448) 및 오디오 처리 유닛(450)을 포함하며, 이는 하나 이상의 A/V 포트(452)를 통해 디스플레이 또는 스피커와 같은 다양한 외부 장치로 통신하도록 구성될 수 있다. 예시적인 주변 인터페이스(444)는 직렬 인터페이스 컨트롤러(454) 또는 병렬 인터페이스 컨트롤러(456)를 포함하며, 이는 하나 이상의 I/O 포트(458)를 통해 입력 장치(예를 들면, 키보드, 마우스, 웹, 음성 입력 장치, 터치 입력 장치 등) 또는 다른 주변 장치(예를 들면, 프린터, 스캐너 등)와 같은 외부 장치와 통신하도록 구성될 수 있다. 예시적인 통신 장치(446)는 네트워크 컨트롤러(460)를 포함하며, 이는 하나 이상의 통신 포트(464)를 통해 네트워크 통신 상에서의 하나 이상의 다른 컴퓨팅 장치(462)와의 통신을 용이하게 하도록 배치될 수 있다.

[0065] 네트워크 통신 링크는 통신 매체의 일 예시일 수 있다. 통신 매체는 전형적으로 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈, 또는 반송파 또는 다른 전송 메커니즘 같은 변조된 데이터 신호 내의 다른 데이터에

의해 구현될 수 있고, 임의의 정보 전달 매체를 포함할 수 있다. "변조된 데이터 신호"는 신호 내에 정보를 인코딩하기 위한 방식으로 설정되거나 변경된 특성 중 하나 이상을 갖는 신호일 수 있다. 제한적인지 않은 예로서, 통신 매체는 유선 네트워크 또는 직접 유선 접속과 같은 유선 매체, 및 음파, 무선 주파수(RF), 마이크로웨이브, 적외선(IR) 및 다른 무선 매체와 같은 무선 매체를 포함할 수 있다. 여기서 사용되는 컴퓨터 판독가능 매체라는 용어는 저장 매체 및 통신 매체 둘 다를 포함할 수 있다.

[0066] 컴퓨팅 장치(400)는, 휴대 전화, PDA(personal data assistant), 개인용 미디어 플레이어 장치, 무선 웹-워치(web-watch) 장치, 개인용 헤드셋 장치, 특수 용도 장치, 또는 위 기능 중 임의의 것을 포함하는 하이브리드 장치 같은 소형 폼 팩터(small-form factor)의 휴대용(또는 모바일) 전자 장치의 일부로서 구현될 수 있다. 컴퓨팅 장치(400)는 또한 랙톱 컴퓨터 및 랙톱이 아닌 컴퓨터 구성을 모두 포함하는 개인용 컴퓨터로서 구현될 수 있다.

[0067] 본 개시는 다양한 태양의 예시로서 의도된 본 출원에 기술된 특정 예시들에 제한되지 않을 것이다. 당업자에게 명백할 바와 같이, 많은 수정과 변형이 그 사상과 범위를 벗어나지 않으면서 이루어질 수 있다. 여기에 열거된 것들에 더하여, 본 개시의 범위 안에서 기능적으로 균등한 방법과 장치가 위의 설명으로부터 당업자에게 명백할 것이다. 그러한 수정과 변형은 첨부된 청구항의 범위에 들어가도록 의도된 것이다. 본 개시는 첨부된 청구항의 용어에 의해서만, 그러한 청구항에 부여된 균등물의 전 범위와 함께, 제한될 것이다. 본 개시가 물론 다양할 수 있는 특정 방법, 시약, 합성 구성 또는 생물학적 시스템에 제한되지 않는 것으로 이해될 것이다. 또한, 여기에서 사용된 용어는 단지 특정 예시들을 기술하기 위한 목적으로, 제한하는 것으로 의도되지 않음이 이해될 것이다.

[0068] 여기에서 실질적으로 임의의 복수 및/또는 단수의 용어의 사용에 대하여, 당업자는 맥락 및/또는 응용에 적절하도록, 복수를 단수로 및/또는 단수를 복수로 해석할 수 있다. 다양한 단수/복수의 치환은 명확성을 위해 여기에서 명시적으로 기재될 수 있다.

[0069] 당업자라면, 일반적으로 본 개시에 사용되며 특히 첨부된 청구범위(예를 들어, 첨부된 청구범위)에 사용된 용어들이 일반적으로 "개방적(open)" 용어(예를 들어, 용어 "포함하는"은 "포함하지만 이에 제한되지 않는"으로, 용어 "갖는"은 "적어도 갖는"으로, 용어 "포함하다"는 "포함하지만 이에 한정되지 않는" 등으로 해석되어야 함)로 의도되었음을 이해할 것이다. 또한, 당업자라면, 도입된 청구항의 기재사항의 특정 수가 의도된 경우, 그러한 의도가 청구항에 명시적으로 기재될 것이며, 그러한 기재사항이 없는 경우, 그러한 의도가 없음을 또한 이해할 것이다. 예를 들어, 이해를 돋기 위해, 이하의 첨부 청구범위는 "적어도 하나" 및 "하나 이상" 등의 도입 구절의 사용을 포함하여 청구항 기재사항을 도입할 수 있다. 그러나, 그러한 구절의 사용이, 부정관사 "하나"( "a" 또는 "an")에 의한 청구항 기재사항의 도입이, 그러한 하나의 기재사항을 포함하는 예시들로, 그러한 도입된 청구항 기재사항을 포함하는 특정 청구항을 제한함을 암시하는 것으로 해석되어서는 안되며, 동일한 청구항이 도입 구절인 "하나 이상" 또는 "적어도 하나" 및 "하나"( "a" 또는 "an")과 같은 부정관사(예를 들어, "하나"는 "적어도 하나" 또는 "하나 이상"을 의미하는 것으로 전형적으로 해석되어야 함)를 포함하는 경우에도 마찬가지로 해석되어야 한다. 이는 청구항 기재사항을 도입하기 위해 사용된 정관사의 경우에도 적용된다. 또한, 도입된 청구항 기재사항의 특정 수가 명시적으로 기재되는 경우에도, 당업자라면 그러한 기재가 전형적으로 적어도 기재된 수(예를 들어, 다른 수식어가 없는 "두개의 기재사항"을 단순히 기재한 것은, 전형적으로 적어도 두 개의 기재사항 또는 두 개 이상의 기재사항을 의미함)를 의미하도록 해석되어야 함을 이해할 것이다. 또한, "A, B 및 C 등 중의 적어도 하나"와 유사한 규칙이 사용된 경우에는, 일반적으로 그러한 해석은 당업자가 그 규칙을 이해할 것이라는 전제가 의도된 것이다(예를 들어, "A, B 및 C 중의 적어도 하나를 갖는 시스템"은, A만을 갖거나, B만을 갖거나, C만을 갖거나, A 및 B를 함께 갖거나, A 및 C를 함께 갖거나, B 및 C를 함께 갖거나, A, B, 및 C를 함께 갖는 시스템 등을 포함하지만 이에 제한되지 않음). "A, B 또는 C 등 중의 적어도 하나"와 유사한 규칙이 사용된 경우에는, 일반적으로 그러한 해석은 당업자가 그 규칙을 이해할 것이라는 전제가 의도된 것이다(예를 들어, "A, B 또는 C 중의 적어도 하나를 갖는 시스템"은, A만을 갖거나, B만을 갖거나, C만을 갖거나, A 및 B를 함께 갖거나, A 및 C를 함께 갖거나, B 및 C를 함께 갖거나, A, B, 및 C를 함께 갖는 시스템 등을 포함하지만 이에 제한되지 않음). 또한 당업자라면, 실질적으로 임의의 이접 접속어(disjunctive word) 및/또는 두 개 이상의 대안적인 용어들을 나타내는 구절은, 그것이 상세한 설명, 청구범위 또는 도면에 있는지와 상관없이, 그 용어들 중의 하나, 그 용어들 중의 어느 하나, 또는 그 용어들 두 개 모두를 포함하는 가능성을 고려했음을 이해할 것이다. 예를 들어, "A 또는 B"라는 구절은 "A" 또는 "B" 또는 "A 및 B"의 가능성을 포함하는 것으로 이해될 것이다.

[0070] 추가적으로, 개시의 특징 또는 양태가 마쿠시(Markush) 그룹으로 기술되는 경우, 개시는 마쿠시 그룹의 임의의

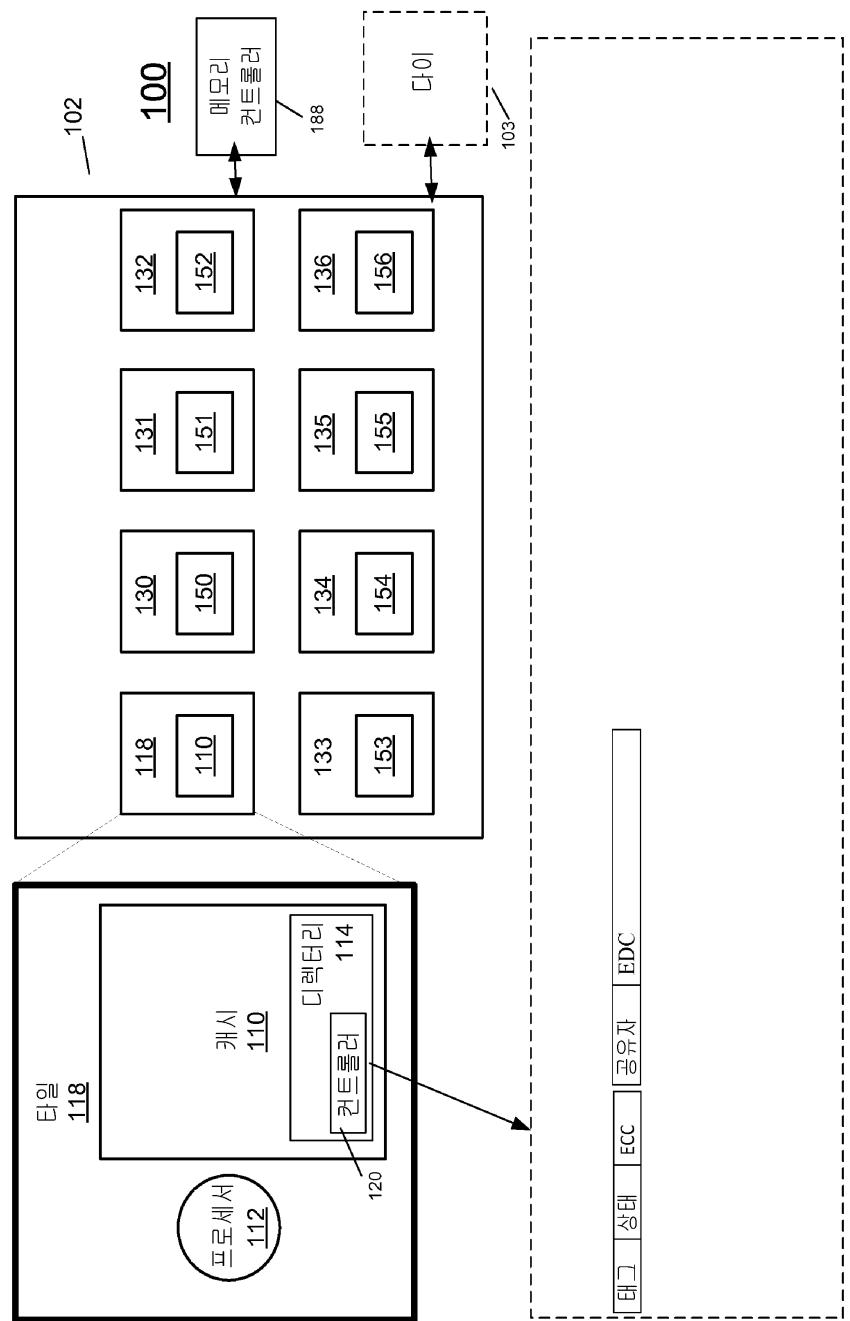
개별 요소 또는 요소들의 하위 그룹 역시 포함하고 있다는 것을 당업자는 인식할 것이다.

[0071] 당업자에게 이해될 것과 같이, 임의의 그리고 모든 목적에서든, 기술 내용을 제공하는 것 등에 있어서, 여기에 개시되어 있는 모든 범위는 임의의 그리고 모든 가능한 하위범위와 그러한 하위범위의 조합을 또한 포함한다. 임의의 열거된 범위는 적어도  $1/2$ ,  $1/3$ ,  $1/4$ ,  $1/5$ ,  $1/10$  등으로 나누어지는 동일한 범위를 충분히 설명하고 실시가능하게 하는 것으로서 쉽게 인식될 수 있다. 제한하지 않는 예시로서, 여기서 논의되는 각각의 범위는 하위  $1/3$ , 중앙  $1/3$ , 상위  $1/3$  등으로 나누어질 수 있다. 또한, "까지", "적어도", "보다 많은", "보다 적은" 등과 같은 언어는 기재된 수를 포함하며, 전술한 하위범위로 후속적으로 나누어질 수 있는 범위를 지칭함이 당업자에게 이해되어야 한다. 마지막으로, 범위는 각각의 개별 요소를 포함함이 이해되어야 한다. 따라서, 예를 들어, 1-3개의 셀을 갖는 그룹은 1, 2 또는 3개의 셀을 갖는 그룹들을 의미한다. 유사하게, 1-5개의 셀을 갖는 그룹은 1, 2, 3, 4 또는 5개의 셀을 갖는 그룹을 의미한다.

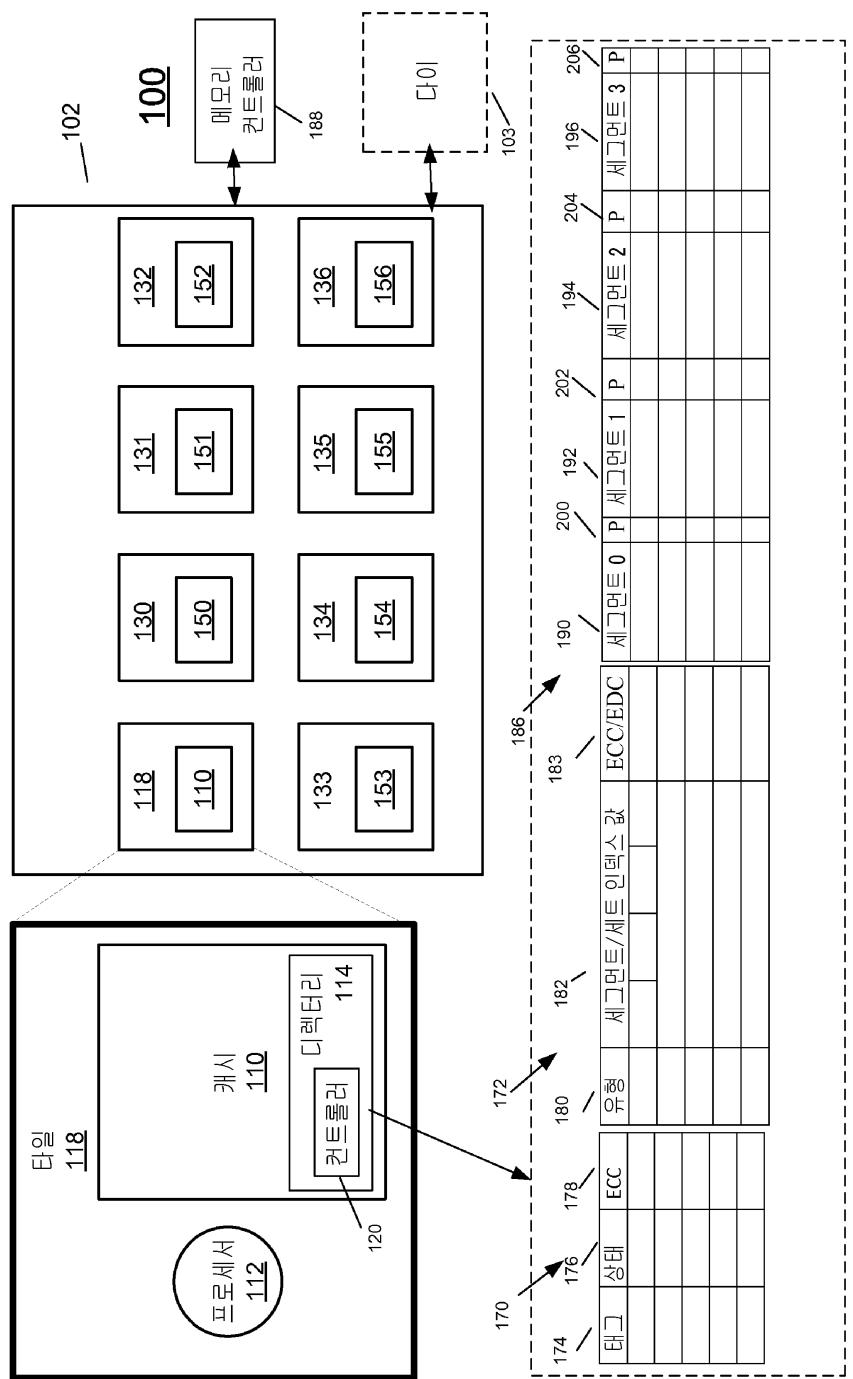
[0072] 다양한 양상 및 예시들이 여기에서 개시되었지만, 다른 양상 및 예시들이 당업자에게 명확할 것이다. 본 개시에 기재된 다양한 양상 및 예시는 예시의 목적으로 제시된 것이고, 제한하려고 의도된 것이 아니며, 진정한 범위와 사상은 이하 청구범위에 의해 나타낸다.

## 도면

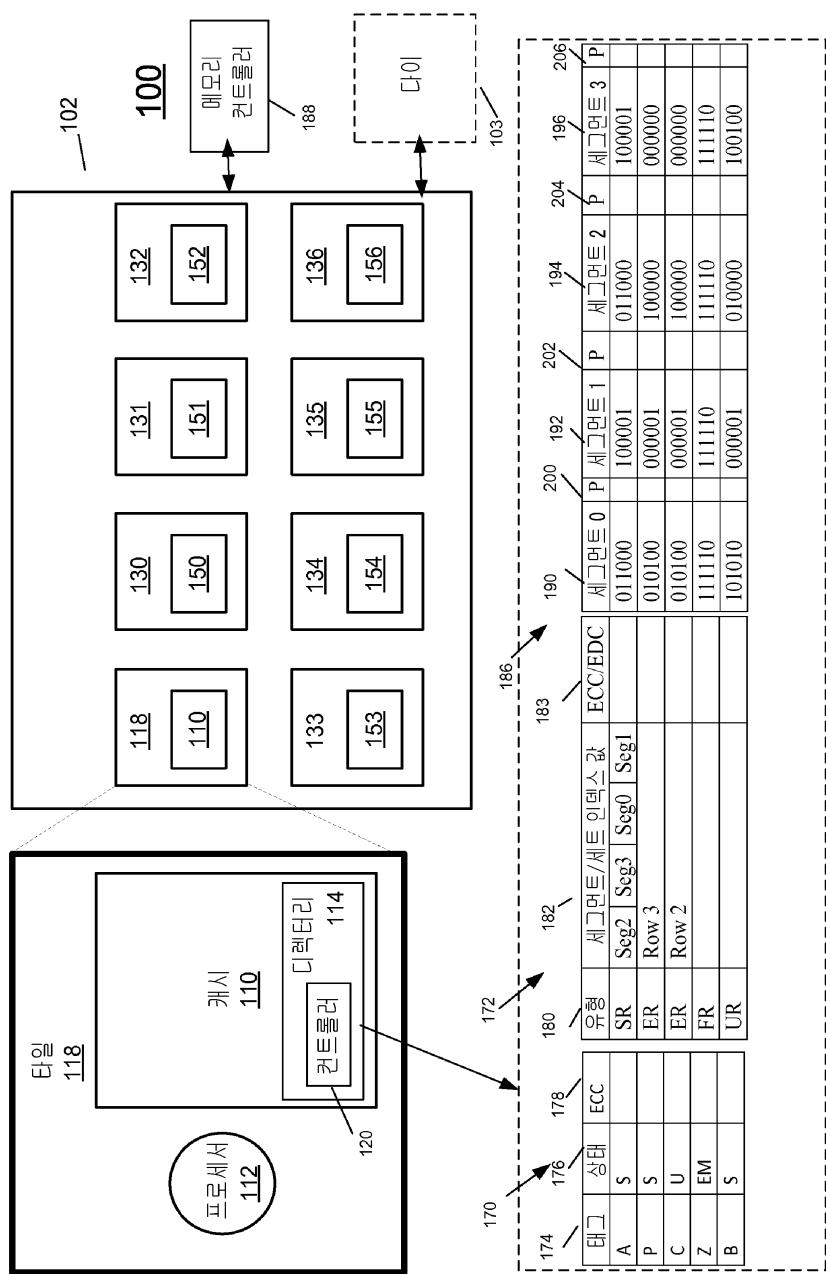
## 도면1



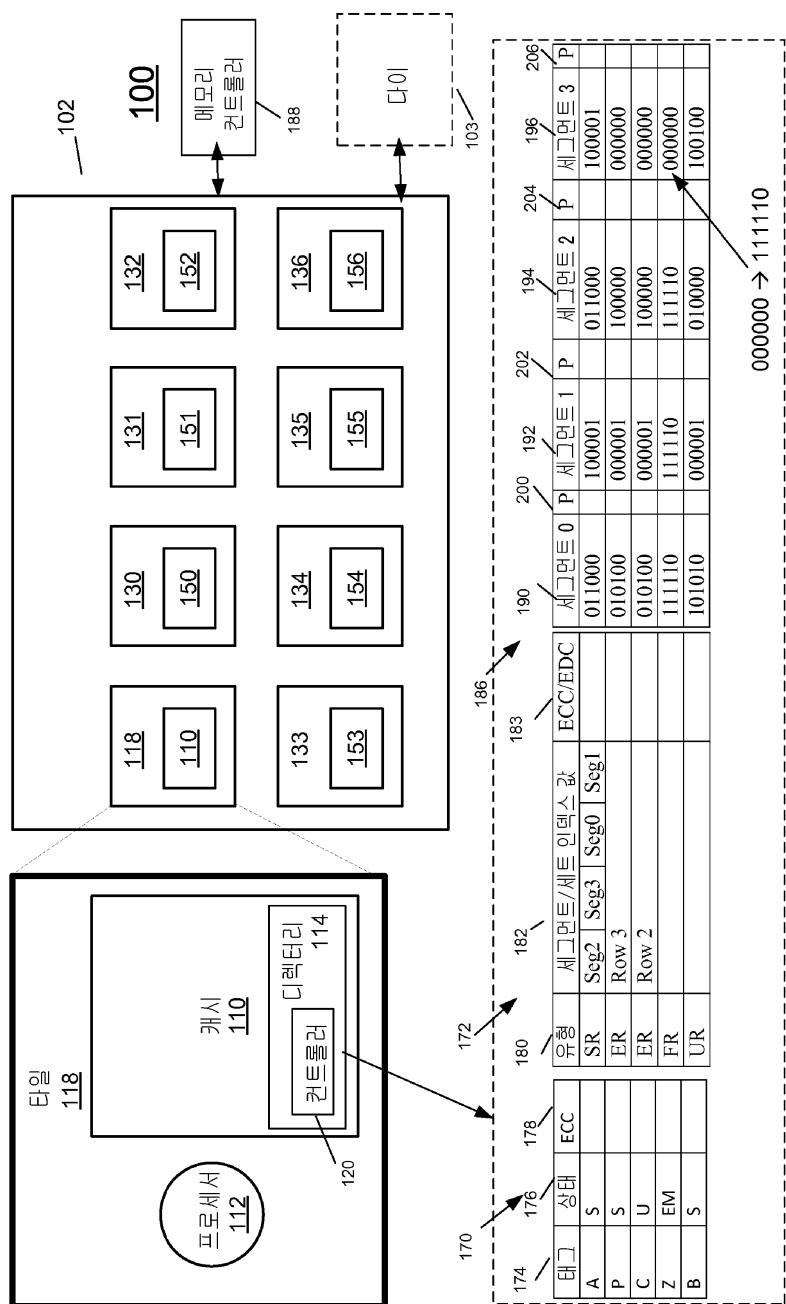
## 도면2



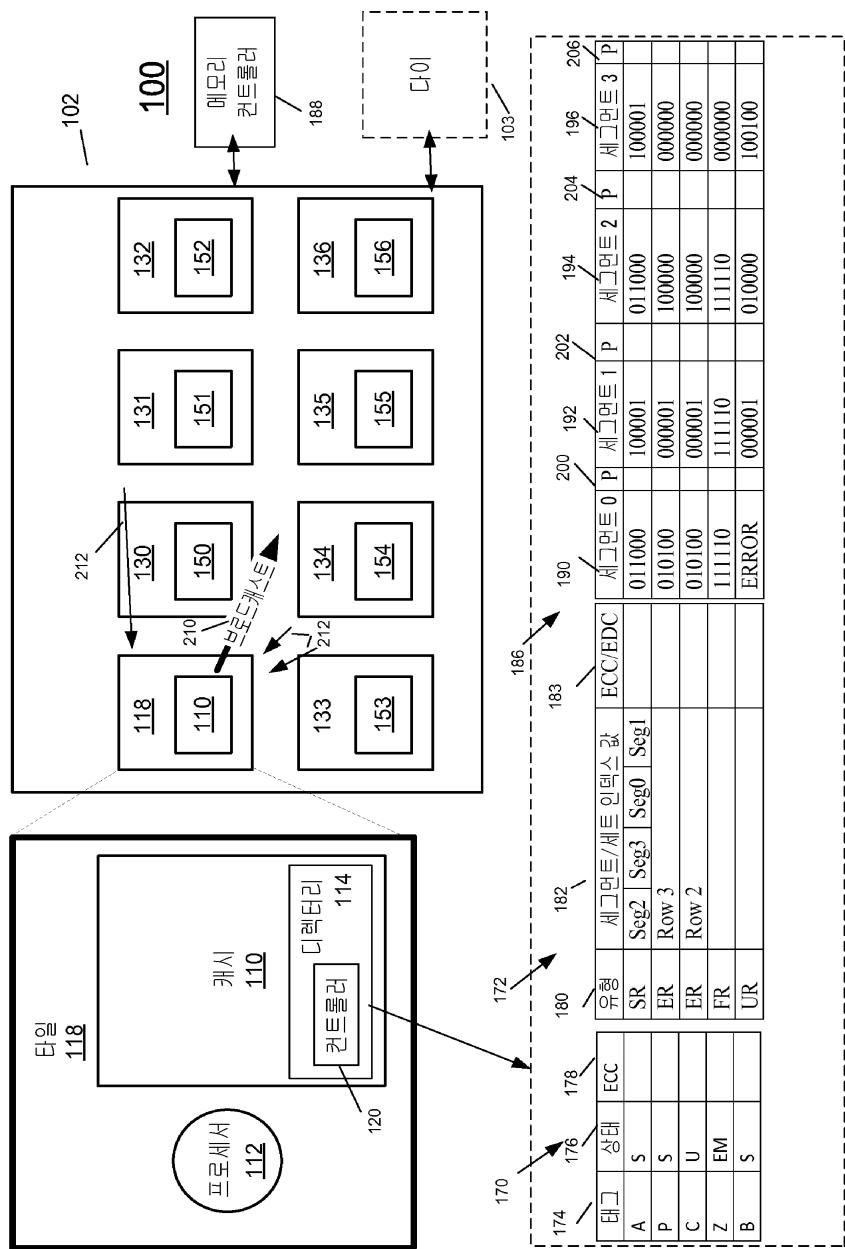
도면3

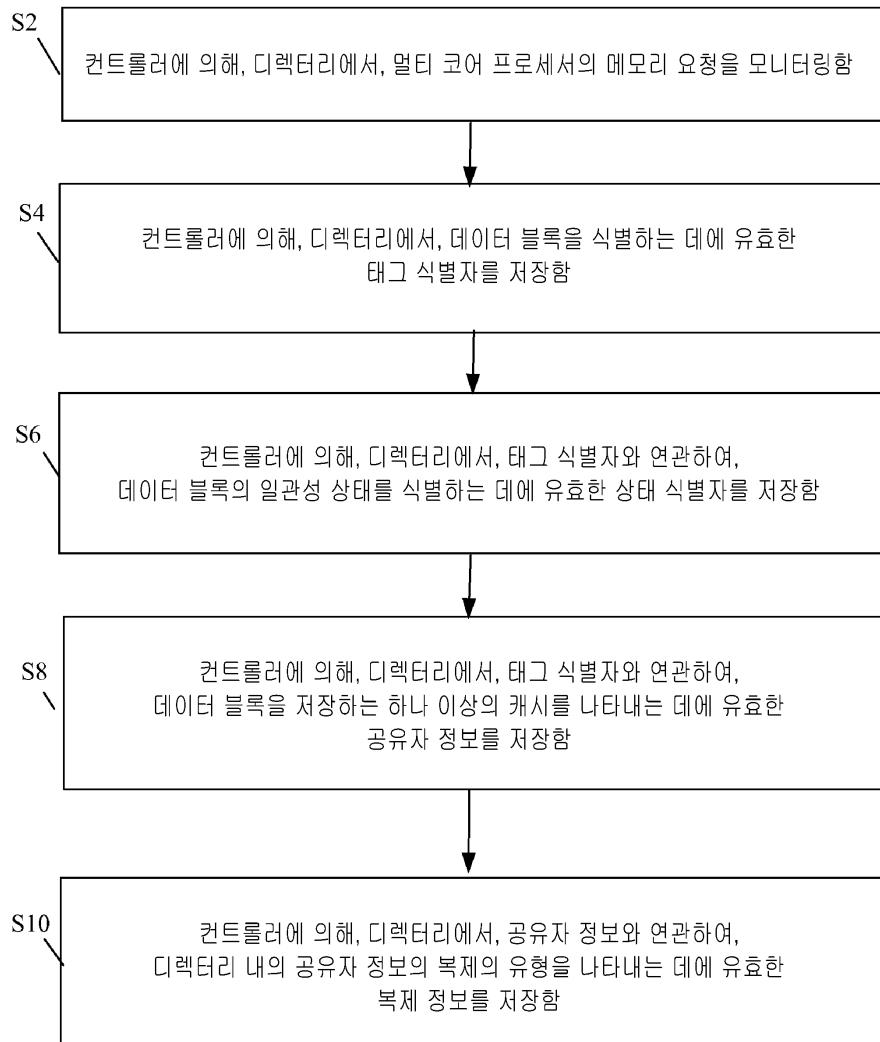


## 도면4



도면5



**도면6**

## 도면7

300 컴퓨터 프로그램 제품

302 신호 베이킹 매체

304

멀티 코어 프로세서의 디렉터리의 캐시 일관성 정보를 저장하기 위한 방법을 위한 하나 이상의 명령어;  
 또는  
 멀티 코어 프로세서의 메모리 요청을, 디렉터리에서, 컨트롤러에 의해, 모니터링하기 위한 하나 이상의  
 명령어; 또는  
 모니터링에 응답하여, 디렉터리에서, 컨트롤러에 의해,  
 데이터 블록을 식별하는 데에 유효한 태그 식별자; 또는  
 태그 식별자와 연관하여, 데이터 블록의 일관성 상태를 식별하는 데에 유효한 상태 식별  
 자; 또는  
 태그 식별자와 연관하여, 데이터 블록을 저장하는 하나 이상의 캐시를 나타내는 데에 유  
 효한 공유자 정보; 또는  
 공유자 정보와 연관하여, 디렉터리의 공유자 정보의 복제의 유형을 나타내는 데에 유효  
 한 복제 정보  
 를 저장하기 위한 하나 이상의 명령어.

<u>306</u>	컴퓨터	<u>308</u>	기록기능 매체	<u>310</u>	통신 매체
	판독기능 매체				

## 도면8

