



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2011년08월11일  
(11) 등록번호 10-1056460  
(24) 등록일자 2011년08월05일

(51) Int. Cl.

G06F 12/06 (2006.01) G06F 5/00 (2006.01)

G06F 11/30 (2006.01)

(21) 출원번호 10-2009-0049731

(22) 출원일자 2009년06월05일

심사청구일자 2009년06월05일

(65) 공개번호 10-2010-0131054

(43) 공개일자 2010년12월15일

(56) 선행기술조사문헌

KR1020090034629 A

JP2005293300 A

KR1020000054925 A

전체 청구항 수 : 총 10 항

(73) 특허권자

세종대학교산학협력단

서울 광진구 군자동 98 세종대학교

(72) 발명자

박기호

서울특별시 성동구 행당2동 한진아파트 116동 1404호

(74) 대리인

특허법인 엠에이피에스

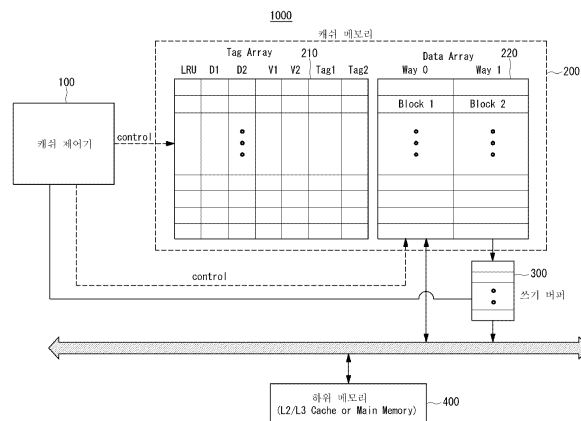
심사관 : 권오성

(54) 캐쉬 제어기 및 캐쉬 블록 교체 방법

(57) 요약

동일한 셋(set)에 둘 이상의 블록을 가지는 집합 연관 캐쉬 메모리를 제어하는 캐쉬 제어기 및 캐쉬 블록 교체 방법이 제공된다. 동일한 셋에 둘 이상의 블록을 가지는 집합 연관 캐쉬 메모리를 제어하는 캐쉬 제어기는, 상기 캐쉬 메모리의 동일한 셋의 둘 이상의 블록 중 일부가 내용이 변경되었는지 감시하는 내용 변경 상태 감시부 및 상기 캐쉬 메모리의 동일한 셋의 블록 중 일부가 내용이 변경된 경우, 내용이 변경되지 않은 블록을 교체하는 캐쉬 블록 교체부를 포함한다.

대표도



## 특허청구의 범위

### 청구항 1

동일한 셋(set)에 둘 이상의 블록이 존재하는 집합 연관 캐쉬 메모리를 제어하는 캐쉬 제어기에 있어서,  
 쓰기 버퍼에 저장된 캐쉬 블록의 수가 미리 정해진 임계치 이상인지를 감시하는 쓰기 버퍼 감시부,  
 상기 캐쉬 메모리의 동일한 셋(set)에 존재하는 블록 중 일부가 내용이 변경되었는지 감시하는 내용 변경 상태 감시부 및  
 상기 쓰기 버퍼에 저장된 캐쉬 블록의 수가 임계치 이상이며, 상기 동일한 셋에 블록 중 일부가 내용이 변경되지 않은 경우, 캐쉬 참조 실패로 인해 인출된 하위 메모리 블록과 상기 내용이 변경되지 않은 블록을 교체하는 캐쉬 블록 교체부  
 를 포함하는 캐쉬 제어기.

### 청구항 2

제 1 항에 있어서,  
 미리 정해진 제 1 캐쉬 교체 알고리즘을 수행하는 제 1 캐쉬 교체 수행부를 더 포함하고,  
 상기 캐쉬 블록 교체부는, 상기 쓰기 버퍼에 저장된 캐쉬 블록의 수가 상기 임계치 이하인 경우에는 상기 제 1 캐쉬 교체 알고리즘을 수행하는 것인 캐쉬 제어기.

### 청구항 3

제 2 항에 있어서,  
 상기 캐쉬 블록 교체부는, 상기 캐쉬 메모리의 동일한 셋에 존재하는 블록 전부가 내용이 변경된 경우에는 상기 제 1 캐쉬 교체 알고리즘을 수행하는 것인 캐쉬 제어기.

### 청구항 4

제 2 항에 있어서,  
 상기 캐쉬 블록 교체부는, 상기 캐쉬 메모리의 동일한 셋에 존재하는 블록 중 변경되지 않은 블록이 복수개인 경우에는 상기 복수의 변경되지 않은 블록에 대해서 상기 제 1 캐쉬 교체 알고리즘을 수행하는 것인 캐쉬 제어기.

### 청구항 5

제 2 항 내지 제 4 항 중 어느 하나의 항에 있어서,  
 상기 제 1 캐쉬 교체 알고리즘은, 가장 예전에 참조된 블록을 교체할 블록으로서 선정하는 LRU(Least Recently Used) 알고리즘인 것인 캐쉬 제어기.

### 청구항 6

동일한 셋(set)에 둘 이상의 블록이 존재하는 집합 연관 캐쉬 메모리의 캐쉬 블록 교체 방법에 있어서,  
 캐쉬 참조 실패 시 쓰기 버퍼에 저장된 캐쉬 블록을 감시하는 단계,  
 상기 저장된 캐쉬 블록의 수가 임계치 이상인지를 판단하는 단계,  
 태그 어레이의 더티 비트에 기초하여 데이터 어레이에 더티 블록이 있는지 판단하는 단계 및

쓰기 버퍼에 저장된 캐쉬 블록의 수가 임계치 이상이며 상기 동일한 셋에 존재하는 둘 이상의 블록 중 일부에 더티 블록이 있는 경우, 상기 캐쉬 참조 실패로 인해 인출된 하위 메모리 블록과 클린 블록을 교체하는 단계를 포함하는 캐쉬 블록 교체 방법.

#### 청구항 7

삭제

#### 청구항 8

제 6 항에 있어서,

상기 저장된 캐쉬 블록의 수가 임계치 이하인 경우, 상기 제 1 캐쉬 교체 알고리즘을 수행하는 것인 캐쉬 블록 교체 방법.

#### 청구항 9

제 8 항에 있어서,

상기 저장된 캐쉬 블록의 수가 임계치 이상인 경우에도, 상기 동일한 셋에 존재하는 둘 이상의 블록이 모두 더티 블록이면 상기 제 1 캐쉬 교체 알고리즘을 수행하는 것인 캐쉬 블록 교체 방법.

#### 청구항 10

제 8 항에 있어서,

상기 동일한 셋에 존재하는 클린 블록이 복수개인 경우에는 상기 복수개의 클린 블록에 대하여 상기 제 1 캐쉬 교체 알고리즘을 수행하는 단계를 더 포함하는 것인 캐쉬 블록 교체 방법.

#### 청구항 11

제 8 항 내지 제 10 항 중 어느 하나의 항에 있어서,

상기 제 1 캐쉬 교체 알고리즘은, 가장 예전에 참조된 블록을 교체할 블록으로서 선정하는 LRU(Least Recently Used) 알고리즘인 것인 캐쉬 블록 교체 방법.

### 명세서

#### 발명의 상세한 설명

##### 기술 분야

[0001] 동일한 셋에 둘 이상의 블록을 가지는 집합 연관 캐쉬 메모리를 제어하는 캐쉬 제어기 및 캐쉬 블록 교체 방법에 관한 것이다.

##### 배경 기술

[0002] 일반적으로 중앙 처리 장치(CPU)의 처리 속도 증가에 비해 메모리의 처리 속도 증가는 매우 느리다. 이에 따라, 중앙 처리 장치와 하위 메모리 간에는 속도 차이가 발생하는데 이를 해결하기 위하여 대부분의 시스템에서는 캐쉬 메모리(Cache Memory) 시스템을 채택하고 있다.

[0003] 캐쉬 메모리를 이용한 구조에서, 데이터 접근 시 중앙 처리 장치가 필요로 하는 내용이 캐쉬 메모리에 존재할 경우에는 아무런 문제없이 정상 동작하지만, 중앙 처리 장치가 필요로 하는 내용이 캐쉬 메모리에 존재하지 않을 경우에는 접근 실패가 발생하게 된다.

[0004] 이와 같이, 접근 실패가 발생하면 해당 내용을 캐쉬 메모리에 존재하도록 하기 위하여 하위 메모리에서 블록을 인출하여 캐쉬 메모리에 저장하는 캐쉬 블록 교체 작업을 수행하여야 한다.

[0005] 이러한 캐쉬 블록 교체 방식으로는 LRU(Least Recently Used), Random, Pseudo LRU 방식 등이 있으며, 이들 방

식은 가능한 다시 사용될 가능성이 적은 캐쉬 블록을 교체함으로써 성능의 향상을 꾀하고자 설계되었다.

[0006] 특히, LRU 방식은 동일한 세트(set)에 존재하는 복수 개의 교체 가능 블록들 중 가장 오랜 시간 전에 참조된 블록을 우선적으로 교체하는 캐쉬 블록 교체 방식으로서, 이는 가장 오래 전에 참조된, 즉 최근에 참조되지 않은 캐쉬 블록이 이후에도 사용될 가능성이 적다는 점에 착안하여 고안되었다.

[0007] 이와 같은 종래의 LRU 방식은 중앙 처리 장치에 의하여 캐쉬 블록이 참조되는 경우에, 해당 블록은 해당 세트에서 가장 최근에 참조된 블록, 즉 MRU(Most Recently Used) 블록이 되므로, 캐쉬 메모리 구성 방식 중 2 way 집합 연관 캐쉬(2 way set-associative cache)의 경우에는 나머지 블록이 가장 예전에 참조된 블록, 즉 LRU(Least Recently Used) 블록이 된다. 따라서, LRU 블록의 정보를 태그 어레이 내의 LRU 비트에 저장하는데, 예를 들면, LRU 블록이 way 0에 저장된 블록이면 0을, way 1에 저장된 블록이면 1을 LRU 비트에 저장한다. 이후, 캐쉬 접근 실패가 발생하여 하위 메모리로부터 블록을 인출하여 캐쉬에 저장해야 하는 경우에 캐쉬 메모리 내의 LRU 비트를 참조하여 새로이 인출되는 블록을 저장하기 위하여, 캐쉬 메모리에 저장된 블록 중 어떤 블록을 교체할 것인가를 결정한다. 즉, LRU 비트에 0이 저장되어 있으면 way 0에 저장된 캐쉬 블록을, LRU 비트에 1이 저장되어 있으면 way 1에 저장된 캐쉬 블록 교체할 블록으로서 선정한다.

[0008] 그러나, 종래의 캐쉬 블록 교체 방식은 교체될 블록이 캐쉬 메모리에 저장된 이후에 갱신이 되어, 해당 블록이 교체되면 쓰기 버퍼를 통한 하위 메모리로의 저장 동작이 필요한지의 여부에 대한 고려 없이, 어느 블록에 보다 최근에 접근되었는가의 정보만으로 교체될 블록을 결정한다는 단점이 있다.

[0009] 또한, 종래의 캐쉬 블록 교체 방식은 새로운 캐쉬 참조에 대한 실패가 발생하여 캐쉬 블록을 교체해야 하는 경우에, 교체할 블록이 더티 블록(Dirty Block)이어서 쓰기 버퍼에 저장되어야 함에도 불구하고 쓰기 버퍼가 이미 모두 사용되어, 쓰기 버퍼에 해당 캐쉬 블록이 저장될 수 있는 공간이 발생할 때까지 기다려야 하는 지연이 발생한다는 문제점이 있다.

[0010] 또한, 종래의 캐쉬 블록 교체 방식의 문제점은 CPU 동작 속도의 증가, CPU와 메모리 간의 상대적인 동작 속도 차이의 증가, 버스트(Burst) 특성을 가진 데이터의 처리 시 캐쉬 블록에 대한 참조 실패가 짧은 시간에 집중되어 발생하는 점 및 최근과 같이 복수 개의 CPU를 사용하는 시스템에서는 보다 빈번하게 발생할 수 있다.

## 발명의 내용

### 해결 하고자하는 과제

[0011] 본 발명의 일 실시예는 쓰기 버퍼(Write Buffer)에 저장되어 있는 블록이 일정 수준 이상이고 캐쉬 메모리에 저장된 블록 중 일부가 내용이 변경된 경우에 내용이 변경되지 않은 클린 블록을 교체 블록으로서 선정하는 캐쉬 제어기 및 캐쉬 블록 교체 방법을 제공한다.

### 과제 해결수단

[0012] 상술한 기술적 과제를 달성하기 위한 기술적 수단으로서, 본 발명의 제 1 측면은 동일한 셋(set)에 둘 이상의 블록이 존재하는 집합 연관 캐쉬 메모리를 제어하는 캐쉬 제어기에 있어서, 상기 캐쉬 메모리의 동일한 셋(set)에 존재하는 블록 중 일부가 내용이 변경되었는지 감시하는 내용 변경 상태 감시부 및 상기 동일한 셋에 존재하는 블록 중 일부가 내용이 변경된 경우, 내용이 변경되지 않은 블록을 교체하는 캐쉬 블록 교체부를 포함하는 캐쉬 제어기를 제공한다.

[0013] 상술한 기술적 과제를 달성하기 위한 기술적 수단으로서, 본 발명의 제 2 측면은 동일한 셋(set)에 둘 이상의 블록이 존재하는 집합 연관 캐쉬 메모리의 캐쉬 블록 교체 방법에 있어서, a) 태그 어레이의 더티 비트에 기초하여 데이터 어레이에 더티 블록이 있는지 판단하는 단계 및 b) 상기 동일한 셋에 존재하는 둘 이상의 블록 중 일부에 더티 블록이 있는 경우, 클린 블록을 교체하는 단계를 포함하는 캐쉬 블록 교체 방법을 제공한다.

## 효 과

[0014] 전술한 본 발명의 과제 해결 수단 중 하나에 의하면, 쓰기 버퍼의 현재 상태, 즉 쓰기 버퍼가 일정 수준 이상인 경우에 교체 대상 블록 중 더티 비트가 아닌 클린 블록을 우선적으로 교체 블록으로서 선정함으로써 쓰기 버퍼

에 저장하기 위해 기다리는 지연 없이 새로운 블록을 하위 메모리로부터 인출하여 캐쉬 메모리에 저장할 수 있다.

### 발명의 실시를 위한 구체적인 내용

- [0015] 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.
- [0016] 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.
- [0017] 이하에는 도 1 내지 도 3을 참조하여, 본 발명의 일 실시예에 따른 캐쉬 제어기 및 캐쉬 블록 교체 방법을 설명한다.
- [0018] 도 1은 본 발명의 일 실시예에 따른 캐쉬 메모리 시스템을 도시한 도면이다. 도 1은 2-way 집합 연관 캐쉬로 LRU(Least Recently Used) 캐쉬 교체 정책을 이용하는 캐쉬 시스템을 제시하고 있다.
- [0019] 도 1을 참조하면, 본 발명의 일 실시예에 따른 캐쉬 메모리 시스템(1000)은 캐쉬 제어기(100), 캐쉬 메모리(200), 쓰기 버퍼(300) 및 하위 메모리(400)를 포함하며, 캐쉬 메모리(200)는 태그 어레이(210) 및 데이터 어레이(220)를 포함하여 이루어진다.
- [0020] 캐쉬 제어기(100)는 캐쉬 접근 실패가 발생하여 하위 메모리(400)로부터 블록을 인출하여 캐쉬 메모리(200)에 저장해야 하는 경우에 쓰기 버퍼(300)의 상태에 따라 교체할 캐쉬 블록을 결정한다.
- [0021] 구체적으로, 캐쉬 제어기(100)는 쓰기 버퍼(300)가 일정 수준 이상인 경우에 캐쉬 메모리(200)의 태그 어레이(210)에 저장되어 있는 해당 블록에 대한 더티 비트(D1, D2)를 참조하여, 해당 블록이 인출되어 저장된 이후에 갱신된 상태이면 해당 블록을 쓰기 버퍼(300)에 저장하여, 하위 메모리(400)에 저장되도록 한다. 반면에, 해당 블록이 인출되어 저장된 이후에 갱신 등이 되지 않은 클린 블록(Clean Block)인 경우에는 쓰기 버퍼(300)에의 저장 등의 수행 없이 해당 블록을 새로 인출된 블록으로 대체함으로써 캐쉬 접근 실패에 따른 새로운 블록의 인출 작업을 수행한다.
- [0022] 캐쉬 메모리(200)는 중앙 처리 장치(미도시)와 하위 메모리(400) 사이에 위치하며, 자주 액세스되는 하위 메모리의 일부 영역만을 저장하여 중앙 처리 장치의 메모리 액세스 속도를 빠르게 하는 고속 메모리이다. 일반적으로, 캐쉬 메모리(200)는 복수 개의 블록(block)들로 구성되며 블록 단위로 읽기와 쓰기 동작을 수행한다.
- [0023] 캐쉬 메모리(200)는 그 내부에 저장된 블록의 주소와 저장된 데이터의 상태를 나타내는 태그 어레이(210) 및 데이터를 저장하는 데이터 어레이(220)를 포함하여 이루어진다.
- [0024] 태그 어레이(210)에는 캐쉬 메모리(200)에 저장된 각 블록의 최근의 참조 정보를 저장하는 LRU(Least Recently Used) 비트, 캐쉬 메모리(200)에 저장된 각 블록의 어드레스의 일부에 해당하는 태그(Tag1, Tag2), 해당 블록이 유효한(Valid) 상태를 가지고 있는가를 나타내는 유효 비트(V1, V2) 및 해당 블록이 인출되어 캐쉬 메모리(200)에 저장된 이후에 값이 변경되었는지를 나타내는 더티 비트(D1, D2)를 포함하여 이루어진다.
- [0025] 쓰기 버퍼(300)는 캐쉬 메모리(200)와 하위 메모리(400) 사이에서 캐쉬 메모리(200) 내의 데이터가 업데이트 되었을 경우, 해당 데이터가 캐쉬 메모리(200)에서 제거될 때 이를 하위 메모리(400)로 전달하는 역할을 한다.
- [0026] 하위 메모리(400)는 제어 연산 등의 기능을 수행하기 위한 중앙 처리 장치(CPU)와 각종 동작을 위한 데이터를 저장하는 기능을 수행하는 메인 메모리 또는 캐쉬 메모리(200)보다 하위 계층 캐쉬 메모리(L2, L3 등) 등으로 구현될 수 있다.
- [0027] 도 2는 본 발명의 일 실시예에 따른 캐쉬 제어기의 세부 구성도이다.
- [0028] 도 2를 참조하면, 본 발명의 일 실시예에 따른 캐쉬 제어기(100)는 LRU 수행부(110), 내용 변경 상태 감시부

(120), 쓰기 버퍼 감시부(130) 및 캐쉬 블록 교체부(140)를 포함한다.

- [0029] LRU 수행부(110)는 캐쉬 메모리(200)에 저장된 교체 블록의 선정에 있어서, 가장 예전에 참조된 블록을 교체할 블록으로서 선정하는 LRU(Least Recently Used) 알고리즘을 수행한다. 즉, LRU 수행부(110)는 캐쉬 접근 실패가 발생하여 하위 메모리로부터 블록을 인출하여 캐쉬에 저장해야 하는 경우에 캐쉬 메모리(200) 내의 LRU 비트를 참조하여 새로이 인출되는 블록을 저장하기 위하여, LRU 비트에 0이 저장되어 있으면 way 0에 저장된 캐쉬 블록을, LRU 비트에 1이 저장되어 있으면 way 1에 저장된 캐쉬 블록 교체할 블록으로서 선정한다.
- [0030] 내용 변경 상태 감시부(120)는 캐쉬 메모리(200)의 블록 중 일부의 내용이 변경되었는지를 감시한다. 즉, 내용 변경 상태 감시부(120)는 캐쉬 메모리(200)에 저장된 블록 중 중앙 처리 장치에 의하여 캐쉬 블록이 참조된 경우에 내용이 변경된 블록이 존재하는지를 감시한다.
- [0031] 구체적으로, 내용 변경 상태 감시부(120)는 캐쉬 메모리(200) 내의 태그 어레이(210)의 더티 비트에 기초하여 데이터 어레이에 더티 블록이 있는지를 감시하는 역할을 수행한다.
- [0032] 쓰기 버퍼 감시부(130)는 쓰기 버퍼(300)에 저장된 캐쉬 블록의 수가 미리 정해진 임계치 이상인지를 감시한다. 즉, 쓰기 버퍼 감시부(130)는 쓰기 버퍼(300)에 저장된 캐쉬 블록의 수가 일정 수준 이상 예를 들어, 2/3, 1/2 이상이거나 또는 모두 차 있는지의 여부를 감시한다.
- [0033] 캐쉬 블록 교체부(140)는 캐쉬 메모리(200)에 저장된 블록 중 일부의 내용이 변경된 경우 내용이 변경되지 않은 클린 블록(Clean Block)을 교체한다. 즉, 캐쉬 블록 교체부(140)는 쓰기 버퍼 감시부(130)의 감시 결과에 따라, 쓰기 버퍼(300)에 저장된 캐쉬 블록의 개수가 미리 정해진 임계치(k) 이하인 경우에는 LRU 알고리즘을 수행하여 블록을 교체하고, 쓰기 버퍼(300)에 저장된 캐쉬 블록의 개수가 임계치(k) 이상인 경우에는 내용이 변경되지 않은 클린 블록을 교체한다. 예를 들어, 캐쉬 블록 교체부(140)는 LRU 비트에 1이 저장되어 기존의 LRU 교체 정책에 의해서는 way 1에 저장되어 있는 블록을 교체 블록으로 선정하였을 것이나, 본 발명에서는 이 경우에도 way 1의 캐쉬 블록이 더티 블록이고, way 0의 캐쉬 블록이 클린 블록인 경우에는 way 0의 블록을 교체 블록으로서 선정하고, 새로 인출할 블록을 way 0에 저장한다.
- [0034] 또한, 캐쉬 블록 교체부(140)는 캐쉬 메모리(200)의 블록들 전부가 내용이 변경된 경우에는 LRU 알고리즘을 수행하여 블록을 교체한다. 예를 들어, way 1과 way 2 모두가 더티 블록인 경우에는 기존의 LRU 알고리즘을 이용하여 LRU 블록을 교체할 블록으로서 선정한다.
- [0035] 4 way 이상의 집합 연관 캐쉬나 완전 연관 캐쉬에서 복수개의 클린 블록 존재하면, 이 경우에는 복수개의 클린 블록에 대해 LRU 알고리즘을 이용하여 클린 블록들 중에 교체할 블록을 선정한다.
- [0036] 지금까지는 본 발명의 일 실시예에 따른 캐쉬 메모리(200)가 2 way 집합 연관 캐쉬(2 way set-associative cache) 메모리인 경우를 예로 들어 설명하였지만, 본 발명이 반드시 이에 한정되는 것은 아니며, 4 way 집합 연관 캐쉬(4 way set-associative cache) 또는 완전 연관 캐쉬(Fully associative cache)인 경우에도 적용할 수 있다.
- [0037] 상술한 바와 같이, 본 발명의 일 실시예에 따른 캐쉬 제어기(100)는 쓰기 버퍼(300)의 현재 상태, 즉 쓰기 버퍼(300)가 일정 수준 이상인 경우에 교체 대상 블록 중 더티 비트가 아닌 클린 블록을 교체 블록으로서 선정함으로써 쓰기 버퍼(300)에 저장하기 위해 기다리는 지연 없이 새로운 블록을 하위 메모리(400)로부터 인출하여 캐쉬 메모리(200)에 저장할 수 있다.
- [0038] 지금까지는 본 발명의 일 실시예에 따른 LRU 수행부(110)가 LRU(Least Recently Used) 알고리즘을 수행하는 것을 예를 들어 설명하였지만, 본 발명이 반드시 LRU에 한정되는 것은 아니며 다른 캐쉬 교체 알고리즘이 채택될 수 있음은 당업자라면 쉽게 이해할 수 있을 것이다.
- [0039] 도 3은 본 발명의 일 실시예에 따른 캐쉬 블록의 교체 방법을 설명하기 위한 흐름도이다.
- [0040] 먼저, 단계(S110)에서 캐쉬 제어기(100)는 중앙 처리 장치가 필요로 하는 내용이 캐쉬 메모리에 존재하지 않을 경우 즉, 캐쉬 실패일 경우 단계(S120)으로 진행하여, 쓰기 버퍼(300)를 감시한다.
- [0041] 단계(S130)에서 캐쉬 제어기(100)는 쓰기 버퍼(300)에 저장된 캐쉬 블록이 일정 수준(k) 이상 예를 들어, 2/3 이상, 1/2 이상 또는 모두 차 있는 경우인지를 판단한다.
- [0042] 단계(S130)에서의 판단 결과, 쓰기 버퍼(300)에 저장된 캐쉬 블록이 일정 수준(k) 이하인 경우에는 단계(S140)으로 진행하여, 가장 예전에 참조된 블록을 교체할 블록으로서 선정하는 기존의 LRU(Least Recently Used Bit)



알고리즘을 수행하여 캐쉬 블록을 교체하는 반면에, 쓰기 버퍼(300)에 저장된 캐쉬 블록이 일정 수준(k) 이상인 경우에는 단계(S200)으로 진행하여, 캐쉬 메모리(200) 내의 태그 어레이(210)의 더티 비트(Dirty Bit)를 검색한다.

[0043] 단계(S210)에서 캐쉬 제어기(100)는 단계(S200)에서 검색한 더티 비트의 상태를 판단하여, 클린 블록(Clean Block)이 존재하는지를 판단한다.

[0044] 단계(S210)에서의 판단 결과, 클린 블록이 존재하지 않는 경우에는 단계(S140)으로 진행하여, LRU(Least Recently Used) 알고리즘을 수행하여 캐쉬 블록을 교체하는 반면에, 클린 블록이 존재하는 경우에는 단계(S220)으로 진행하여, 복수 개의 클린 블록이 존재하는지를 판단한다.

[0045] 단계(S220)에서의 판단 결과, 복수 개의 클린 블록이 존재하는 경우에는 단계(S240)으로 진행하여, 복수 개의 클린 블록들에 대하여 기존의 LRU(Least Recently Used) 알고리즘을 수행하여 캐쉬 블록을 교체한다.

[0046] 반면에, 단계(S220)에서의 판단 결과, 복수 개의 클린 블록이 존재하지 않는 경우에는 단계(S230)으로 진행하여, 클린 블록을 교체한다. 즉, 캐쉬 제어기(100)는 해당 블록이 더티 블록이면 해당 블록은 쓰기 버퍼(300)에 저장하고, 새로 인출된 블록을 교체된 블록의 자리에 저장함으로써 캐쉬 블록 교체를 수행한다.

[0047] 본 발명의 실시예에서는 더티 블록을 교체하는 경우에 이와 같은 추가적인 쓰기 버퍼에 저장하는 동작에 의한 수행 지연 등을 고려하여, 쓰기 버퍼에 저장된 블록의 수가 임계치를 넘는 경우에 클린 블록을 교체 블록으로 선정함으로써 이러한 지연을 감소시킬 수 있다.

[0048] 지금까지는 본 발명의 일 실시예에 따른 캐쉬 메모리(200)의 캐쉬 메모리 구성 방식이 2 way 집합 연관 캐쉬(2 way set-associative cache)인 경우를 예로 들어 설명하였지만, 본 발명이 반드시 이에 한정되는 것은 아니며, 4 way 집합 연관 캐쉬(4 way set-associative cache), 8 way 집합 연관 캐쉬(8 way set-associative cache) 또는 완전 연관 캐쉬(Fully associative cache)의 경우에도 적용할 수 있다.

[0049] 또한, 지금까지는 본 발명의 일 실시예에 따른 캐쉬 블록의 교체 방법에 있어서, 쓰기 버퍼(300)에 저장된 캐쉬 블록이 일정 수준(k) 이하이거나, 더티 비트가 존재하지 않거나, 모든 way가 더티 블록인 경우에 LRU(Least Recently Used) 알고리즘을 수행하는 것을 예로 들어 설명하였지만, 캐쉬 교체 알고리즘은 반드시 LRU에 한정되는 것은 아니며 다른 캐쉬 교체 알고리즘이 채택될 수 있음은 당업자라면 쉽게 이해할 수 있을 것이다.

[0050] 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 매커니즘을 포함하며, 임의의 정보 전달 매체를 포함한다.

[0051] 진술한 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.

[0052] 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 것으로 해석되어야 한다.

### 도면의 간단한 설명

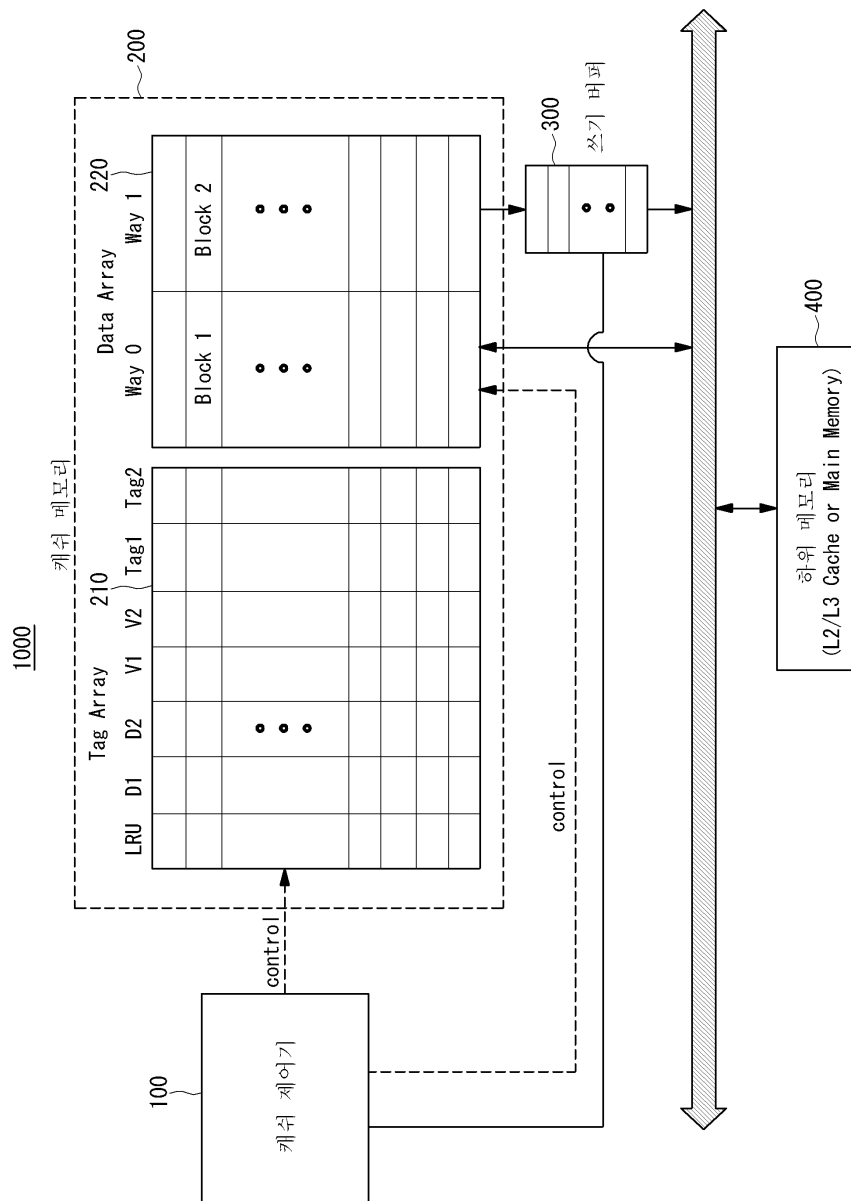
[0053] 도 1은 본 발명의 일 실시예에 따른 캐쉬 메모리 시스템을 도시한 도면이다.

[0054] 도 2는 본 발명의 일 실시예에 따른 캐쉬 제어기의 세부 구성도이다.

[0055] 도 3은 본 발명의 일 실시예에 따른 캐쉬 블록의 교체 방법을 설명하기 위한 흐름도이다.

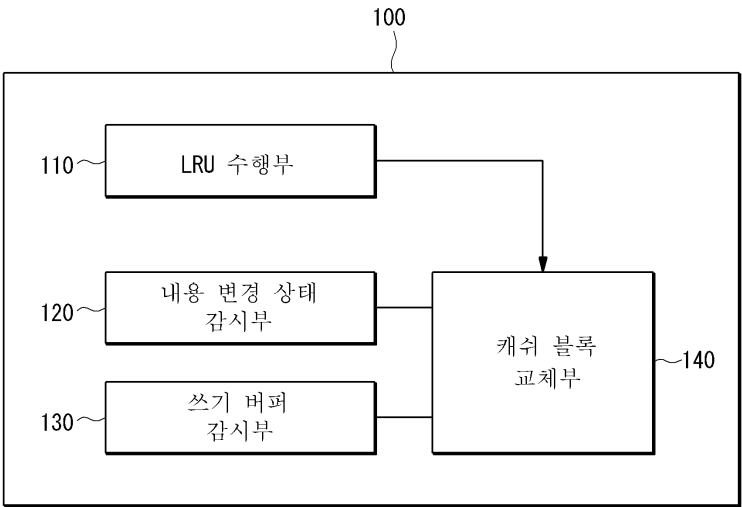
도면

도면1





도면2



도면3

