



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0127694
(43) 공개일자 2017년11월22일

(51) 국제특허분류(Int. Cl.)
G06F 12/08 (2016.01) G06F 12/02 (2006.01)
G06F 9/30 (2017.01)
(52) CPC특허분류
G06F 12/0815 (2013.01)
G06F 12/0238 (2013.01)
(21) 출원번호 10-2016-0058087
(22) 출원일자 2016년05월12일
심사청구일자 2016년05월12일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암
동5가)
(72) 발명자
김선욱
경기도 남양주시 도농로 34, 214동 2003호 (도농
동, 부영그린타운)
김호권
경기도 성남시 분당구 동판교로 92, 321동 106호
(백현동, 백현마을3단지아파트)
(뒷면에 계속)
(74) 대리인
전중학

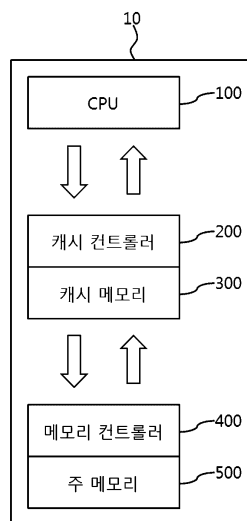
전체 청구항 수 : 총 13 항

(54) 발명의 명칭 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치 및 그 방법

(57) 요약

본 발명은 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치 및 그 방법을 개시한다. 즉, 본 발명은 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리하는 영구 메모리의 영구적 저장 순서를 제공함으로써 전원을 켜면 이전의 작업 내용으로 바로 복구가 가능한 즉시 부팅을 실현할 수 있으며, 전원 차단 또는 시스템 충돌에 의한 복구가 필요한 상황에서도 주 메모리로 영구 메모리를 사용함에 있어 주 메모리 차원에서 복구가 가능할 수 있다.

대 표 도 - 도1



(52) CPC특허분류

G06F 12/0804 (2013.01)

G06F 12/0868 (2013.01)

G06F 9/30043 (2013.01)

(72) 발명자

전재영

서울특별시 강동구 풍성로73길 28 (문촌동)

한영선

서울특별시 성북구 안암로 145, 고려대학교 공학관
237호 (안암동5가)

이 발명을 지원한 국가연구개발사업

과제고유번호 10052716

부처명 산업통상자원부

연구관리전문기관 한국산업기술평가원

연구사업명 산업기술혁신사업

연구과제명 스마트 센서 SoC용 초저전압 회로 및 IP 설계 기술 개발

기 여 율 1/1

주관기관 연세대학교

연구기간 2015.06.01 ~ 2016.05.31

명세서

청구범위

청구항 1

영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 방법에 있어서,
캐시 컨트롤러를 통해, 캐시 메모리 또는 별도의 저장 공간에 쓰기 명령 순서를 저장하는 단계; 및
상기 캐시 컨트롤러를 통해, 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계를 포함하는 메모리 관리 방법.

청구항 2

제 1 항에 있어서,
상기 캐시 메모리에 쓰기 명령 순서를 저장하는 단계는,
상기 캐시 컨트롤러를 통해, CPU로부터 전달되는 메모리 요청을 수신하는 과정;
상기 캐시 컨트롤러를 통해, 상기 수신된 메모리 요청이 쓰기 명령인지 여부를 확인하는 과정;
상기 캐시 컨트롤러를 통해, 상기 수신된 메모리 요청이 쓰기 명령일 때, 상기 수신된 메모리 요청에 대해서 모든 쓰기 명령을 요청 순서대로 시간 의존성(time dependency)을 추가하는 과정; 및
상기 캐시 컨트롤러를 통해, 상기 캐시 메모리 또는 별도의 저장 공간에 쓰기 명령의 순서를 저장하는 과정을 포함하는 것을 특징으로 하는 메모리 관리 방법.

청구항 3

제 2 항에 있어서,
상기 캐시 메모리 또는 별도의 저장 공간에 쓰기 명령의 순서를 저장하는 과정은,
상기 캐시 메모리 또는 별도의 저장 공간에 할당된 쓰기 명령의 순서 저장소에 상기 쓰기 명령의 순서를 영구적 저장 순서로 저장하는 것을 특징으로 하는 메모리 관리 방법.

청구항 4

제 1 항에 있어서,
상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계는,
상기 캐시 컨트롤러를 통해, CPU로부터 전달되는 메모리 요청이 쓰기 명령이 아닌 경우 및 상기 쓰기 명령 순서의 저장에 완료된 경우 중 어느 하나의 경우에 해당하는 상태에서, 캐시 적중이고 쓰기 및 더티 적중인 경우, 캐시 적중이 아니고 더티 에비క్ష인 경우 및 더 이상 쓰기 순서 저장을 수행할 수 없는 경우 중 어느 하나의 경우일 때, 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행하는 것을 특징으로 하는 메모리 관리 방법.

청구항 5

제 1 항에 있어서,
상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계는,
상기 캐시 컨트롤러를 통해, 캐시 적중(cache hit)의 발생 여부를 판단하는 과정;
상기 캐시 컨트롤러를 통해, 상기 캐시 적중이 발생할 때, 상기 캐시 적중이 쓰기 및 더티 적중(write and dirty hit)인지 여부를 확인하는 과정; 및

상기 캐시 컨트롤러를 통해, 상기 캐시 적중이 쓰기 및 더티 적중일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 적중(hit)이 된 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 과정을 포함하는 것을 특징으로 하는 메모리 관리 방법.

청구항 6

제 1 항에 있어서,

상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계는,

상기 캐시 컨트롤러를 통해, 캐시 적중(cache hit)의 발생 여부를 판단하는 과정;

상기 캐시 컨트롤러를 통해, 상기 캐시 적중이 발생하지 않은 상태일 때, 더티 에빅션(dirty eviction)의 발생 여부를 확인하는 과정; 및

상기 캐시 컨트롤러를 통해, 상기 더티 에빅션이 발생한 상태일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅탐(victim) 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 과정을 포함하는 것을 특징으로 하는 메모리 관리 방법.

청구항 7

영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치에 있어서,

쓰기 명령을 포함하는 메모리 요청을 전달하는 CPU(Central Processing Unit);

쓰기 명령 순서를 저장하기 위한 저장 공간이 할당된 캐시 메모리;

영구 메모리로 구성된 주 메모리; 및

상기 캐시 메모리의 저장 공간에 쓰기 명령 순서를 저장하고, 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 캐시 컨트롤러를 포함하는 메모리 관리 장치.

청구항 8

제 7 항에 있어서,

상기 캐시 메모리는,

복수의 레벨 또는 복수의 메모리 계층으로 구성되며, 상기 복수의 레벨의 캐시 메모리에 상기 쓰기 명령 순서를 각각 저장하는 것을 특징으로 하는 메모리 관리 장치.

청구항 9

제 7 항에 있어서,

상기 캐시 컨트롤러는,

상기 CPU로부터 전달되는 메모리 요청이 쓰기 명령일 때, 상기 수신된 메모리 요청에 대해서 모든 쓰기 명령을 요청 순서대로 시간 의존성(time dependency)을 추가하고, 상기 캐시 메모리의 저장 공간에 쓰기 명령의 순서를 저장하는 것을 특징으로 하는 메모리 관리 장치.

청구항 10

제 7 항에 있어서,

상기 캐시 컨트롤러는,

캐시 적중이 발생할 때, 상기 캐시 적중이 쓰기 및 더티 적중인지 여부를 확인하고, 상기 캐시 적중이 쓰기 및 더티 적중일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 적중(hit)이 된

주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 것을 특징으로 하는 메모리 관리 장치.

청구항 11

제 10 항에 있어서,

상기 하위 메모리는,

복수의 메모리 계층으로 구성된 상기 캐시 메모리에 대해서, 순서대로 복수의 쓰기 명령이 저장된 특정 메모리 계층에서 상기 캐시 메모리 내에서의 복수의 쓰기 명령이 저장된 해당 특정 메모리 계층보다 아래 단의 하나 이상의 메모리 계층과 상기 주 메모리를 포함하는 것을 특징으로 하는 메모리 관리 장치.

청구항 12

제 7 항에 있어서,

상기 캐시 컨트롤러는,

캐시 적중이 발생하지 않은 상태일 때 더티 에빅션(dirty eviction)의 발생 여부를 확인하고, 상기 더티 에빅션이 발생한 상태일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티م(victim) 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 것을 특징으로 하는 메모리 관리 장치.

청구항 13

제 7 항에 있어서,

상기 캐시 컨트롤러는,

멀티 프로세서 시스템에서, 프로세서 간의 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행하기 위해서 캐시 일관성 프로토콜에서 수정(M) 상태에서 무효(I) 상태로 전이가 일어날 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 순차적으로 저장된 쓰기 명령을 하위 메모리로 수행하는 것을 특징으로 하는 메모리 관리 장치.

발명의 설명

기술 분야

[0001] 본 발명은 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치 및 그 방법에 관한 것으로, 특히 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리하는 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치 및 그 방법에 관한 것이다.

배경 기술

[0002] 기존의 컴퓨터 시스템은 주 메모리로 DRAM을 사용하고 있어, 컴퓨터의 전원이 꺼졌을 때, DRAM에 있던 데이터가 지워지기 때문에 켜진 이후 보조 기억 장치로부터 다시 시작하는 것이 일반화되어 있다. 따라서, 명시적인 파일 저장 명령을 통해서 메모리의 내용이 보조 기억 장치에 기록되어야 한다.

[0003] 또한, 휘발성인 DRAM과는 달리 차세대 주 메모리로 개발되고 있는 PRAM, MRAM 등은 비휘발성 특성을 가지므로 영구적 저장을 주 메모리 차원에서 제공할 수 있다. 프로세서는 데이터의 읽기, 쓰기를 빠르게 하기 위해 데이터를 주 메모리에 저장하지 않고 프로세서와 주 메모리 사이에 있는 캐시 메모리에 우선적으로 저장하기 때문에, 프로세서가 실행될 때 캐시 메모리와 주 메모리에 저장된 데이터의 내용이 상이할 수 있다. 캐시 메모리와 주 메모리 간의 데이터의 일관성을 제공하기 위한 방법에는 크게 라이트-쓰루(write-through)와 라이트-백(write-back) 방법이 있다. 상기 라이트-쓰루(write-through)는 캐시 메모리와 주 메모리 간의 내용을 항상 일치시키기 위해 프로세서에서 메모리 쓰기 명령이 발생할 때마다, 캐시 메모리와 주 메모리를 모두 업데이트하는 정책이다. 이러한 라이트-쓰루 정책은 메모리 쓰기 명령마다 캐시 메모리와 주 메모리를 모두 사용하므로, 메모리 트래픽(traffic)을 증가시켜 시스템 성능을 저하시킬 수 있다. 따라서, 이러한 단점을 극복하기 위하여 일반

적으로 캐시 메모리에만 데이터를 쓰게 하고 주 메모리에는 특이 사항(예를 들어 캐시 에빅션, 캐시 플러시 등 포함)이 발생했을 때에만 데이터를 업데이트하는 라이트-백(write-back) 정책을 사용하고 있다. 현재의 캐시 메모리에서는 대부분 라이트-백 정책을 사용하기 때문에 주 메모리에 데이터가 기록되는 순서가 프로세서에서 데이터를 기록하는 순서와 일치하지 않아, 시스템의 충돌이나 전원이 차단되는 상황에서 주 메모리를 통해 시스템을 복구하는 것이 불가능하다.

선행기술문헌

특허문헌

- [0004] (특허문헌 0001) 한국공개특허 제10-2014-0066392호 [명칭: 캐시 메모리 및 캐시 메모리를 포함하는 어플리케이션 프로세서의 데이터 관리 방법]

발명의 내용

해결하려는 과제

- [0005] 본 발명의 목적은 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템(또는 메모리 관리 장치)에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리하는 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치 및 그 방법을 제공하는 데 있다.
- [0006] 본 발명의 다른 목적은 프로세서와 주 메모리 사이에 위치하고 있는 캐시 메모리에서 영구적 저장 순서를 제공하는 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치 및 그 방법을 제공하는 데 있다.

과제의 해결 수단

- [0007] 본 발명의 실시예에 따른 메모리 관리 방법은 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 방법에 있어서, 캐시 컨트롤러를 통해, 캐시 메모리 또는 별도의 저장 공간에 쓰기 명령 순서를 저장하는 단계; 및 상기 캐시 컨트롤러를 통해, 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계를 포함할 수 있다.
- [0008] 본 발명과 관련된 일 예로서 상기 캐시 메모리에 쓰기 명령 순서를 저장하는 단계는, 상기 캐시 컨트롤러를 통해, CPU로부터 전달되는 메모리 요청을 수신하는 과정; 상기 캐시 컨트롤러를 통해, 상기 수신된 메모리 요청이 쓰기 명령인지 여부를 확인하는 과정; 상기 캐시 컨트롤러를 통해, 상기 수신된 메모리 요청이 쓰기 명령일 때, 상기 수신된 메모리 요청에 대해서 모든 쓰기 명령을 요청 순서대로 시간 의존성(time dependency)을 추가하는 과정; 및 상기 캐시 컨트롤러를 통해, 상기 캐시 메모리 또는 별도의 저장 공간에 쓰기 명령의 순서를 저장하는 과정을 포함할 수 있다.
- [0009] 본 발명과 관련된 일 예로서 상기 캐시 메모리 또는 별도의 저장 공간에 쓰기 명령의 순서를 저장하는 과정은, 상기 캐시 메모리 또는 별도의 저장 공간에 할당된 쓰기 명령의 순서 저장소에 상기 쓰기 명령의 순서를 영구적 저장 순서로 저장할 수 있다.
- [0010] 본 발명과 관련된 일 예로서 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계는, 상기 캐시 컨트롤러를 통해, CPU로부터 전달되는 메모리 요청이 쓰기 명령이 아닌 경우 및 상기 쓰기 명령 순서의 저장이 완료된 경우 중 어느 하나의 경우에 해당하는 상태에서, 캐시 적중이고 쓰기 및 더티 적중인 경우, 캐시 적중이 아니고 더티 에빅션인 경우 및 더 이상 쓰기 순서 저장을 수행할 수 없는 경우 중 어느 하나의 경우일 때, 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행할 수 있다.
- [0011] 본 발명과 관련된 일 예로서 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계는, 상기 캐시 컨트롤러를 통해, 캐시 적중(cache hit)의 발생 여부를 판단하는 과정; 상기 캐시 컨트롤러를 통해, 상기 캐시 적중이 발생할 때, 상기 캐시 적중이 쓰기 및 더티 적중(write and dirty hit)인지 여부를 확인하는 과정; 및 상기 캐시 컨트롤러를 통해, 상기 캐시 적중이 쓰기 및 더티 적중일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 적중

(hit)이 된 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 과정을 포함할 수 있다.

[0012] 본 발명과 관련된 일 예로서 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 단계는, 상기 캐시 컨트롤러를 통해, 캐시 적중(cache hit)의 발생 여부를 판단하는 과정; 상기 캐시 컨트롤러를 통해, 상기 캐시 적중이 발생하지 않은 상태일 때, 더티 에빅션(dirty eviction)의 발생 여부를 확인하는 과정; 및 상기 캐시 컨트롤러를 통해, 상기 더티 에빅션이 발생한 상태일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티м(victim) 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 과정을 포함할 수 있다.

[0013] 본 발명의 실시예에 따른 메모리 관리 장치는, 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치에 있어서, 쓰기 명령을 포함하는 메모리 요청을 전달하는 CPU(Central Processing Unit); 쓰기 명령 순서를 저장하기 위한 저장 공간이 할당된 캐시 메모리; 영구 메모리로 구성된 주 메모리; 및 상기 캐시 메모리의 저장 공간에 쓰기 명령 순서를 저장하고, 상기 쓰기 명령 순서가 저장된 캐시 메모리에서 상기 영구 메모리로 구성된 주 메모리까지 메모리 쓰기 순서를 순서대로 수행하는 캐시 컨트롤러를 포함할 수 있다.

[0014] 본 발명과 관련된 일 예로서 상기 캐시 메모리는, 복수의 레벨 또는 복수의 메모리 계층으로 구성되며, 상기 복수의 레벨의 캐시 메모리에 상기 쓰기 명령 순서를 각각 저장할 수 있다.

[0015] 본 발명과 관련된 일 예로서 상기 캐시 컨트롤러는, 상기 CPU로부터 전달되는 메모리 요청이 쓰기 명령일 때, 상기 수신된 메모리 요청에 대해서 모든 쓰기 명령을 요청 순서대로 시간 의존성(time dependency)을 추가하고, 상기 캐시 메모리의 저장 공간에 쓰기 명령의 순서를 저장할 수 있다.

[0016] 본 발명과 관련된 일 예로서 상기 캐시 컨트롤러는, 캐시 적중이 발생할 때, 상기 캐시 적중이 쓰기 및 더티 적중인지 여부를 확인하고, 상기 캐시 적중이 쓰기 및 더티 적중일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 적중(hit)이 된 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행할 수 있다.

[0017] 본 발명과 관련된 일 예로서 상기 하위 메모리는, 복수의 메모리 계층으로 구성된 상기 캐시 메모리에 대해서, 순서대로 복수의 쓰기 명령이 저장된 특정 메모리 계층에서 상기 캐시 메모리 내에서의 복수의 쓰기 명령이 저장된 해당 특정 메모리 계층보다 아래 단의 하나 이상의 메모리 계층과 상기 주 메모리를 포함할 수 있다.

[0018] 본 발명과 관련된 일 예로서 상기 캐시 컨트롤러는, 캐시 적중이 발생하지 않은 상태일 때 더티 에빅션(dirty eviction)의 발생 여부를 확인하고, 상기 더티 에빅션이 발생한 상태일 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티м(victim) 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리를 거쳐 상기 주 메모리까지 메모리 쓰기 순서를 순서대로 수행할 수 있다.

[0019] 본 발명과 관련된 일 예로서 상기 캐시 컨트롤러는, 멀티 프로세서 시스템에서, 프로세서 간의 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행하기 위해서 캐시 일관성 프로토콜에서 수정(M) 상태에서 무효(I) 상태로 전이가 일어날 때, 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 순차적으로 저장된 쓰기 명령을 하위 메모리로 수행할 수 있다.

발명의 효과

[0020] 본 발명은 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템(또는 메모리 관리 장치)에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리함으로써, 전원을 켜면 이전의 작업 내용으로 바로 복구가 가능한 즉시 부팅을 실현할 수 있으며, 전원 차단 또는 시스템 충돌에 의한 복구가 필요한 상황에서도 주 메모리로 영구 메모리를 사용함에 있어 주 메모리 차원에서 복구가 가능한 효과가 있다.

[0021] 또한, 본 발명은 프로세서와 주 메모리 사이에 위치하고 있는 캐시 메모리에서 영구적 저장 순서를 제공함으로써, 메모리의 명령의 순서를 관리함에 따라 프로그램의 디버깅에 활용 시 오류가 난 지점 직전까지 바로 접근이 가능하며, 디버깅을 위한 프로그램 재실행 과정을 생략할 수 있는 효과가 있다.

도면의 간단한 설명

[0022] 도 1은 본 발명의 실시예에 따른 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장

치의 구성을 나타낸 블록도이다.

도 2 및 도 3은 본 발명의 실시예에 따른 멀티 프로세서에서 영구적 저장 순서로 메모리 쓰기 명령을 수행하기 위한 캐시 일관성 프로토콜(cache coherence protocol)의 프로세서와 버스에서의 상태 전이 예시도이다.

도 4 및 도 5는 본 발명의 실시예에 따른 메모리 쓰기 순서를 저장하는 공간을 구성하는 방법에 대한 예시도이다.

도 6은 본 발명의 실시예에 따른 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 방법을 나타낸 흐름도이다.

도 7은 본 발명의 실시예에 따른 캐시 메모리에 저장되는 메모리 요청 주소의 예를 나타낸 도이다.

발명을 실시하기 위한 구체적인 내용

- [0023] 본 발명에서 사용되는 기술적 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아님을 유의해야 한다. 또한, 본 발명에서 사용되는 기술적 용어는 본 발명에서 특별히 다른 의미로 정의되지 않는 한, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 의미로 해석되어야 하며, 과도하게 포괄적인 의미로 해석되거나, 과도하게 축소된 의미로 해석되지 않아야 한다. 또한, 본 발명에서 사용되는 기술적인 용어가 본 발명의 사상을 정확하게 표현하지 못하는 잘못된 기술적 용어일 때에는 당업자가 올바르게 이해할 수 있는 기술적 용어로 대체되어 이해되어야 할 것이다. 또한, 본 발명에서 사용되는 일반적인 용어는 사전에 정의되어 있는 바에 따라, 또는 전후 문맥상에 따라 해석되어야 하며, 과도하게 축소된 의미로 해석되지 않아야 한다.
- [0024] 또한, 본 발명에서 사용되는 단어의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한 복수의 표현을 포함한다. 본 발명에서 "구성된다" 또는 "포함한다" 등의 용어는 발명에 기재된 여러 구성 요소들 또는 여러 단계를 반드시 모두 포함하는 것으로 해석되지 않아야 하며, 그 중 일부 구성 요소들 또는 일부 단계들은 포함되지 않을 수도 있고, 또는 추가적인 구성 요소 또는 단계들을 더 포함할 수 있는 것으로 해석되어야 한다.
- [0025] 또한, 본 발명에서 사용되는 제 1, 제 2 등과 같이 서수를 포함하는 용어는 구성 요소들을 설명하는데 사용될 수 있지만, 구성 요소들은 용어들에 의해 한정되어서는 안 된다. 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제 1 구성 요소는 제 2 구성 요소로 명명될 수 있고, 유사하게 제 2 구성 요소도 제 1 구성 요소로 명명될 수 있다.
- [0026] 이하, 첨부된 도면을 참조하여 본 발명에 따른 바람직한 실시예를 상세히 설명하되, 도면 부호에 관계없이 동일하거나 유사한 구성 요소는 동일한 참조 번호를 부여하고 이에 대한 중복되는 설명은 생략하기로 한다.
- [0027] 또한, 본 발명을 설명함에 있어서 관련된 공지 기술에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우 그 상세한 설명을 생략한다. 또한, 첨부된 도면은 본 발명의 사상을 쉽게 이해할 수 있도록 하기 위한 것일 뿐, 첨부된 도면에 의해 본 발명의 사상이 제한되는 것으로 해석되어서는 아니 됨을 유의해야 한다.
- [0028] 도 1은 본 발명의 실시예에 따른 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 장치(10)의 구성을 나타낸 블록도이다.
- [0029] 도 1에 도시한 바와 같이, 메모리 관리 장치(10)는 CPU(100), 캐시 컨트롤러(200), 캐시 메모리(300), 메모리 컨트롤러(400) 및 주 메모리(500)로 구성된다. 도 1에 도시된 메모리 관리 장치(10)의 구성 요소 모두가 필수 구성 요소인 것은 아니며, 도 1에 도시된 구성 요소보다 많은 구성 요소에 의해 메모리 관리 장치(10)가 구현될 수도 있고, 그보다 적은 구성 요소에 의해서도 메모리 관리 장치(10)가 구현될 수도 있다.
- [0030] 여기서, 상기 메모리 관리 장치(10)는 컴퓨터 시스템일 수 있다.
- [0031] 상기 CPU(Central Processing Unit)(또는 프로세서)(100)는 싱글 프로세서 시스템, 멀티 프로세서 시스템 등을 포함한다.
- [0032] 또한, 상기 CPU(100)는 메모리 요청을 상기 캐시 컨트롤러(200)에 전달(또는 전송)한다. 여기서, 상기 메모리 요청은 쓰기 명령(또는 메모리 쓰기 명령/메모리 쓰기 요청), 읽기 명령(또는 메모리 읽기 명령/메모리 읽기 요청) 등을 포함할 수 있다.
- [0033] 상기 캐시 컨트롤러(cache controller)(200)는 상기 CPU(100)로부터 전달되는 상기 메모리 요청을 수신한다.

- [0034] 또한, 상기 캐시 컨트롤러(200)는 상기 수신된 메모리 요청이 쓰기 명령(또는 메모리 쓰기 명령/메모리 쓰기 요청)인지 여부를 확인(또는 판단)한다.
- [0035] 확인 결과(또는 판단 결과), 상기 수신된 메모리 요청이 쓰기 명령인 경우, 상기 캐시 컨트롤러(200)는 상기 수신된(또는 상기 캐시 컨트롤러(200)로 들어오는) 메모리 요청에 대해서 모든 쓰기 명령을 요청 순서대로 시간 의존성(time dependency)을 추가한다.
- [0036] 또한, 상기 캐시 컨트롤러(200)는 해당 쓰기 명령의 순서를 저장한다.
- [0037] 즉, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300) 또는 별도의 저장 공간(예를 들어 큐(queue)(미도시))에 할당된 쓰기 명령의 순서 저장소에 저장된 쓰기 명령 순서 리스트에 해당 쓰기 명령의 순서를 영구적 저장 순서로 저장한다. 이때, 상기 캐시 컨트롤러(200)는 복수의 레벨(또는 복수의 메모리 계층)로 구성되는 상기 캐시 메모리(300)에 대해서, 복수의 레벨의 캐시 메모리 중 어느 하나의 레벨의 캐시 메모리(300)에 해당 쓰기 명령의 순서를 영구적 저장 순서로 저장한다.
- [0038] 이와 같이, 상기 캐시 컨트롤러(200)는 쓰기 명령의 순서를 요청 순서대로(또는 수신한 순서대로) 저장한다.
- [0039] 또한, 상기 캐시 컨트롤러(200)는 다음의 조건을 만족하는 경우에 쓰기 명령을 실행한다.
- [0040] 즉, 상기 캐시 컨트롤러(200)는 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행한다. 이때, 하위 메모리로 실행되는 단계는 두 가지 방법에 의해 수행된다. 캐시 미스(miss)가 발생하여 상기 캐시 메모리(300)의 더티 에빅션(dirty eviction)이 발생하는 경우와 쓰기 적중(write hit)이 발생하여 캐시 라인이 갱신되었을 때, 상기 캐시 컨트롤러(200)는 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티م(victim) 또는 적중(hit)이 된 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리(300)를 거쳐 상기 주 메모리(500)까지 메모리 쓰기 순서를 순서대로 수행한다.
- [0041] 또한, 상기 확인 결과(또는 판단 결과) 상기 수신된 메모리 요청이 쓰기 명령이 아닌 경우 및 상기 쓰기 순서의 기록이 완료된 경우 중 어느 하나의 경우에 해당하는 상태에서, 캐시 적중이고 쓰기 및 더티 적중인 경우, 캐시 적중이 아니고 더티 에빅션인 경우 및 더 이상 쓰기 순서 저장을 수행할 수 없는 경우 중 어느 하나의 경우, 상기 캐시 컨트롤러(200)는 상기 영구적 저장 순서로 쓰기 명령(또는 상기 순서대로 저장된 쓰기 명령)을 하위 메모리로 실행(또는 순서대로 저장된 쓰기 명령을 순차적으로 하위 메모리에 플러쉬(flush))한다. 여기서, 상기 하위 메모리는 복수의 메모리 계층으로 구성된 상기 캐시 메모리(300)에 대해서, 순서대로 복수의 쓰기 명령이 저장된 특정 메모리 계층에서 상기 캐시 메모리(300) 내에서의 복수의 쓰기 명령이 저장된 해당 메모리 계층보다 아래 단의 하나 이상의 메모리 계층과 주 메모리(500)(또는 복수의 캐시 메모리(300)가 존재할 경우, 복수의 쓰기 명령이 저장된 특정 캐시 메모리(300)로부터 상기 주 메모리(500)까지의 복수의 메모리)를 나타낸다.
- [0042] 즉, 상기 캐시 컨트롤러(200)는 캐시 적중(cache hit)의 발생 여부를 판단(또는 확인)한다. 여기서, 상기 캐시 적중은 쓰기 적중(write hit), 읽기 적중(read hit) 등을 포함한다.
- [0043] 판단 결과, 상기 캐시 적중이 발생한 경우, 상기 캐시 컨트롤러(200)는 해당 캐시 적중이 쓰기 및 더티 적중(write and dirty hit)인지 여부(또는 해당 캐시 적중이 쓰기 적중이면서 더티 적중인지 여부)를 확인한다.
- [0044] 확인 결과, 상기 캐시 적중이 쓰기 및 더티 적중이 아닌 경우, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300)에 대한 접근 기능을 수행한다.
- [0045] 또한, 확인 결과, 상기 캐시 적중이 쓰기 및 더티 적중인 경우, 상기 캐시 컨트롤러(200)는 상기 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 적중(hit)이 된 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리(300)를 거쳐 상기 주 메모리(500)까지 메모리 쓰기 순서를 순서대로 수행한다.
- [0046] 또한, 판단 결과, 상기 캐시 적중이 발생하지 않은 경우, 상기 캐시 컨트롤러(200)는 더티 에빅션(dirty eviction)의 발생 여부를 확인한다.
- [0047] 확인 결과, 상기 더티 에빅션이 발생한 경우, 상기 캐시 컨트롤러(200)는 상기 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티م(victim) 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리(300)를 거쳐 상기 주 메모리(500)까지 메모리 쓰기 순서를 순서대로 수행한다.
- [0048] 또한, 확인 결과, 상기 더티 에빅션이 발생하지 않은 경우, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300)

에 대한 접근 기능을 수행한다.

- [0049] 이와 같이, 상기 캐시 컨트롤러(200)는 싱글 프로세서에 대해서 쓰기 명령의 순서를 저장하고, 쓰기 명령을 실행할 수 있다.
- [0050] 도 2 및 도 3은 멀티 프로세서에서 영구적 저장 순서로 쓰기 명령을 수행하기 위한 캐시 일관성 프로토콜(cache coherence protocol)의 프로세서와 버스에서의 상태 전이 예시도이다.
- [0051] 또한, 상기 도 2 및 도 3은 MESI(Modified, Exclusive, Shared, Invalid) 프로토콜을 사용한 예이다.
- [0052] 상기 MESI 프로토콜은 상기 멀티 프로세서에 캐시 일관성을 위해 사용하는 라이트-백 캐시 정책을 사용한다.
- [0053] 여기서, 상기 수정(M: Modified) 상태는 데이터가 수정(또는 변경)된 상태를 나타내고, 상기 배타(E: Exclusive) 상태는 상기 주 메모리(500)와 동일한 내용을 상기 캐시 메모리(300) 내에서 유일하게 데이터를 가진 상태를 나타내고, 상기 공유(S: Shared) 상태는 데이터 변경이 없이 두 개 이상의 프로세서 캐시에 적재되어 있는 상태를 나타내고, 상기 무효(I: Invalid) 상태는 데이터가 다른 프로세서에 의해 수정되어 무효가 된 상태 또는 데이터가 적재되지 않은 상태를 나타낸다.
- [0054] 또한, 상기 MESI 프로토콜은 현재의 프로세서(또는 상기 CPU(100))와 다른 캐시 메모리에서의 요청에 의해 상태 전이를 나타낸다. 즉, 프로세서에서의 메모리 요청에 의해 상기 캐시 메모리(300)로 접근이 일어날 때, 상기 캐시 메모리(300)에 적재되어 있는 경우, 상기 캐시 컨트롤러(200)는 캐시 적중(cache hit)이 발생하고, 상기 캐시 메모리(300)에서 데이터를 읽거나 상기 캐시 메모리(300)로 데이터를 쓴다.
- [0055] 또한, 상기 메모리 요청에 대한 데이터가 상기 캐시 메모리(300)에 적재되어 있지 않은 경우, 상기 캐시 컨트롤러(200)는 캐시 미스(cache miss)가 발생하고, 데이터를 읽어 오기 위해 버스(bus) 상에서 데이터가 있는 곳으로 다른 캐시 메모리(또는 다른 캐시 컨트롤러)에 메모리 요청을 보낸다.
- [0056] 읽기 요청에 의한 미스는 읽기 미스(read miss)로 발생하고, 쓰기 요청에 의한 미스는 쓰기 미스(write miss)로 발생한다.
- [0057] 상기 MESI 프로토콜의 읽기 미스의 경우, 상기 캐시 컨트롤러(200)는 다른 프로세서 캐시 메모리에 있는지 요청을 보내고, 다른 프로세서의 캐시 메모리에 있다면, 해당 캐시 메모리에서 데이터를 읽어와 상기 캐시 메모리(300)에 적재하고, 무효(I) 상태에서 공유(S) 상태로 상태 전이를 한다. 이때, 상기 캐시 컨트롤러(200)가 다른 캐시 메모리로부터 읽기 요청을 받는 경우, 상기 캐시 컨트롤러(200)는 해당 데이터가 상기 캐시 메모리(300)에 있는지 여부를 응답하여, 있을 경우 해당 데이터를 다른 캐시에 전달하면서 배타(E) 또는 수정(M) 상태에서 공유(S) 상태로 전이한다. 또한, 다른 프로세서의 캐시 메모리에도 해당 데이터가 존재하지 않는 경우, 상기 캐시 컨트롤러(200)는 하위 메모리로 읽기 요청을 보낸다. 또한, 읽기 미스에 의해 하위 메모리로부터 상기 캐시 메모리(300)에 읽은 데이터가 적재가 되면, 상기 캐시 컨트롤러(200)는 유일한 데이터가 되므로 무효(I) 상태에서 배타(E) 상태로 상태 전이를 한다.
- [0058] 또한, 상기 MESI 프로토콜의 쓰기 미스의 경우, 최신의 값을 갖는 데이터는 유일하게 존재해야 하므로, 다른 프로세서의 캐시 메모리에 데이터를 무효화시키게 된다. 상기 캐시 컨트롤러(200)는 하위 메모리로 데이터 읽기 요청을 보내 해당 캐시 메모리(300)에 데이터를 적재한 후에, 쓰기를 수행하여 데이터를 변경하고, 캐시 메모리(300)의 상태는 무효(I) 상태에서 수정(M) 상태로 전이한다.
- [0059] 또한, MESI 프로토콜의 쓰기 적중의 경우, 수정(M) 상태에서는 데이터 변경만 발생하고, 배타(E) 상태에서는 데이터를 변경하면서 수정(M) 상태로 전이만 발생하며, 공유(S) 상태에서는 다른 캐시 메모리에 무효화 신호를 보내고 데이터를 변경하면서 수정(M) 상태로 전이가 발생한다.
- [0060] 또한, 상기 캐시 메모리(300)가 다른 데이터를 적재하기 위해 에빅션(eviction)이 되거나 무효화 신호에 의해 해당 캐시 메모리(300)에서 데이터가 쫓겨날 경우, 상기 캐시 컨트롤러(200)는 데이터 변경이 없는 클린(clean)인 배타(E) 상태와 공유(S) 상태였던 경우에는, 하위 메모리로 라이트-백이 불필요하지만, 더티 상태인 수정(M) 상태였던 경우에는 하위 메모리로 쓰기 요청을 보내 라이트-백 처리를 수행한다.
- [0061] 상기 도 2에서 클린(E 또는 S) 상태의 캐시 라인이 쓰기 명령에 의해 수정(M) 상태가 되거나 무효(I) 상태에서의 캐시 라인이 쓰기 명령에 의해 수정(M) 상태가 될 때, 상기 캐시 컨트롤러(200)는 쓰기 순서를 상기 캐시 메모리(300) 또는 상기 별도의 저장 공간에 저장한다. 또한, 도 3에서와 같이 수정(M) 상태에서 원격에서의 적중(hit)에 의한 무효(I) 신호를 받거나 로컬(local)에서 에빅션(eviction)이 발생할 때, 상기 캐시 컨트롤러(200)

0)는 쓰기 명령을 실행하여, 하위 메모리로 쓰기 명령을 순서대로 실행하여, 상기 메모리 컨트롤러(400)에는 최종적으로 프로세서의 쓰기 명령이 순서대로 도착한다.

- [0062] 본 발명의 실시예에서는 상기 MESI를 예로 들어 설명하고 있으나, 이에 한정되는 것은 아니며, MESI 프로토콜을 확장한 MOESI 프로토콜의 경우에도 적용할 수 있다. 즉, 상기 MOESI 프로토콜의 경우에는 소유(O: Owned) 상태가 추가되어, 상기 MESI 프로토콜에서는 수정(M) 상태에서 다른 프로세서에 의해 읽기가 발생한 경우, 공유(S) 상태로 만들기 위해 데이터를 하위 메모리로 라이트-백이 발생하지만, 상기 MOESI 프로토콜은 상기 캐시 컨트롤러(200)에 의해 수정(M) 상태에서 소유(O) 상태로 전이만 발생시켜 라이트-백 수를 줄인다.
- [0063] 또한, 상기 MOESI 프로토콜의 경우, 소유(O) 상태에서 에빅션(eviction)이 발생할 때는 더티 에빅션이 발생할 때와 같으므로, 상기 캐시 컨트롤러(200)에 의해 쓰기 명령을 순서대로 실행하는 것은 동일하다. 다만, 수정(M) 상태에서 다른 프로세서에 의해 공유를 위하여 공유(O) 상태로 전이할 때는 모든 프로세서에 보여야 하기 때문에, 읽기 요청이 발생하지만, 상기 캐시 컨트롤러(200)에 의해 하위 메모리로 쓰기 명령이 실행되어야 한다.
- [0064] 이와 같이, 멀티 프로세서 시스템에서, 프로세서 간의 영구적 저장 순서를 관리하기 위해서 MESI 프로토콜의 임의의 상태에서 수정(M) 상태로 전이가 일어날 때, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300) 또는 별도의 저장 공간에 할당된 쓰기 명령의 순서 저장소에 저장된 쓰기 명령 순서 리스트에 해당 쓰기 명령의 순서를 영구적 저장 순서로 저장한다.
- [0065] 또한, 상기 멀티 프로세서 시스템에서, 프로세서 간의 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행하기 위해서 MESI 프로토콜에서 수정(M) 상태에서 무효(I) 상태로 전이가 일어날 때, 상기 수신된 메모리 요청이 쓰기 명령이 아닌 경우 및 상기 쓰기 순서의 기록이 완료된 경우 중 어느 하나의 경우에 해당하는 상태에서, 캐시 적중이고 쓰기 및 더티 적중인 경우, 캐시 적중이 아니고 더티 에빅션인 경우 및 더 이상 쓰기 순서 저장을 수행할 수 없는 경우 중 어느 하나의 경우, 상기 캐시 컨트롤러(200)는 상기 영구적 저장 순서로 쓰기 명령(또는 상기 순서대로 저장된 쓰기 명령)을 하위 메모리로 실행(또는 순서대로 저장된 쓰기 명령을 순차적으로 하위 메모리에 플러쉬)한다.
- [0066] 즉, 상기 멀티 프로세서 시스템에서, 프로세서 간의 영구적 저장 순서로 쓰기 명령을 하위 메모리로 실행하기 위해서 MESI 프로토콜에서 수정(M) 상태에서 무효(I) 상태로 전이가 일어날 때, 상기 캐시 컨트롤러(200)는 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 순차적으로 저장된 쓰기 명령을 하위 메모리로 수행한다.
- [0067] 상기 캐시 메모리(300)는 제 1 레벨 캐시, 제 2 레벨 캐시 등의 메모리를 포함하는 복수의 메모리 계층(또는 복수의 레벨)으로 구성한다.
- [0068] 또한, 상기 캐시 메모리(300)는 상기 캐시 컨트롤러(200)의 제어에 의해, 복수의 레벨(또는 복수의 메모리 계층) 중 일부를 쓰기 명령(또는 쓰기 명령의 주소)을 순서대로 저장하기 위한 영역으로 설정한 상태일 수 있다.
- [0069] 또한, 상기 쓰기 명령을 저장하기 위해서 도 4에 도시된 바와 같이 별도의 저장 공간을 두거나, 도 5에 도시된 바와 같이 상기 캐시 메모리(300)의 일부 공간을 사용할 수 있다.
- [0070] 즉, 상기 도 4의 경우, 하드웨어 큐(queue)를 구성하여, 쓰기 명령 주소를 저장하기 위해 별도의 공간을 두어 관리하는 방법이다.
- [0071] 또한, 상기 도 5의 경우, n개의 웨이(way)를 가진 상기 캐시 메모리(300)의 웨이 중에서 1개의 웨이를 쓰기 명령 주소를 저장하기 위한 공간으로 사용하고, 나머지 n-1개의 웨이를 상기 캐시 메모리(300)에서 사용하는 방법이다.
- [0072] 또한, 상기 도 5와 같이 상기 캐시 메모리(300) 내의 일부 공간을 상기 쓰기 명령을 저장하기 위한 용도로 사용함에 따라, 상기 캐시 메모리(300)의 동작에 맞춰 동작하게 하는 장점과 함께 별도의 저장 공간이 필요치 않아 비용을 절감할 수 있다.
- [0073] 또한, 상기 캐시 메모리(300)의 일부를 상기 쓰기 명령 주소를 저장하기 위한 공간으로 사용하는 경우, 상기 캐시 컨트롤러(200)가 쓰기 순서의 첫 주소를 가지고, 다음 주소를 계속 연결하는 방식으로 구현할 수 있다.
- [0074] 상기 메모리 컨트롤러(memory controller)(400)는 상기 주 메모리(500)를 관리한다.
- [0075] 또한, 상기 메모리 컨트롤러(400)는 상기 캐시 컨트롤러(300)로부터 전달되는 순서가 정렬된 복수의 쓰기 명령

을 수신한다.

- [0076] 또한, 상기 메모리 컨트롤러(400)는 상기 수신된 순서가 정렬된 복수의 쓰기 명령을 순서대로 상기 주 메모리(500)에 저장한다.
- [0077] 상기 주 메모리(500)는 영구 메모리(persistent memory)를 주 메모리로 포함한다.
- [0078] 또한, 상기 주 메모리(500)는 상기 메모리 컨트롤러(400)의 제어에 의해, 상기 캐시 컨트롤러(300)로부터 전달되는 순서가 정렬된 복수의 쓰기 명령을 순서대로 저장한다.
- [0079] 이와 같이, 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템(또는 메모리 관리 장치)에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리할 수 있다.
- [0080] 또한, 이와 같이, 프로세서와 주 메모리 사이에 위치하고 있는 캐시 메모리에서 영구적 저장 순서를 제공할 수 있다.
- [0081] 이하에서는, 본 발명에 따른 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 방법을 도 1 내지 도 7을 참조하여 상세히 설명한다.
- [0082] 도 6은 본 발명의 실시예에 따른 영구 메모리의 영구적 저장 순서를 제공하기 위한 캐시에서의 메모리 관리 방법을 나타낸 흐름도이다.
- [0083] 먼저, 캐시 컨트롤러(200)는 CPU(100)로부터 전달되는 메모리 요청을 수신한다.
- [0084] 또한, 상기 캐시 컨트롤러(200)는 상기 수신된 메모리 요청이 쓰기 명령(또는 메모리 쓰기 명령/메모리 쓰기 요청)인지 여부를 확인(또는 판단)한다.
- [0085] 일 예로, 상기 캐시 컨트롤러(200)는 상기 CPU(100)로부터 전달되는 제 11 메모리 요청이 쓰기 명령인지 여부를 확인한다(S610).
- [0086] 확인 결과(또는 판단 결과), 상기 수신된 메모리 요청이 쓰기 명령인 경우, 상기 캐시 컨트롤러(200)는 상기 수신된(또는 상기 캐시 컨트롤러(200)로 들어오는) 메모리 요청에 대해서 모든 쓰기 명령을 요청 순서대로 시간 의존성(time dependency)을 추가한다.
- [0087] 일 예로, 상기 제 11 메모리 요청이 쓰기 명령일 때, 상기 캐시 컨트롤러(200)는 상기 제 11 메모리 요청을 요청 순서대로 시간 의존성을 추가한다(S620).
- [0088] 이후, 상기 캐시 컨트롤러(200)는 해당 쓰기 명령의 순서를 저장한다.
- [0089] 즉, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300) 또는 별도의 저장 공간(미도시)에 할당된 쓰기 명령의 순서 저장소에 저장된 쓰기 명령 순서 리스트에 해당 쓰기 명령의 순서를 영구적 저장 순서로 저장한다. 이때, 상기 캐시 컨트롤러(200)는 복수의 레벨(또는 복수의 메모리 계층)로 구성되는 상기 캐시 메모리(300)에 대해서, 복수의 레벨의 캐시 메모리 중 어느 하나의 레벨의 캐시 메모리(300)에 해당 쓰기 명령의 순서를 영구적 저장 순서로 저장한다.
- [0090] 일 예로, 도 7에 도시된 바와 같이, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300)에 포함된 복수의 레벨의 캐시 중 제 1 레벨 캐시의 일부 영역(예를 들어 1개의 웨이(way))에 할당된 쓰기 명령의 순서 저장소에 상기 제 11 메모리 요청(또는 제 11 쓰기 명령)(710)을 저장한다. 이때, 상기 제 1 레벨 캐시에는 제 1 메모리 요청 내지 제 10 메모리 요청(720)이 미리 저장된 상태일 수 있다.
- [0091] 이와 같이, 상기 캐시 컨트롤러(200)는 쓰기 명령의 순서를 요청 순서대로(또는 수신한 순서대로) 저장할 수 있다(S630).
- [0092] 이후, 상기 확인 결과(또는 판단 결과) 상기 수신된 메모리 요청이 쓰기 명령이 아닌 경우 및 상기 쓰기 순서의 기록이 완료된 경우 중 어느 하나의 경우에 해당하는 상태에서, 캐시 적중이고 쓰기 및 더티 적중인 경우, 캐시 적중이 아니고 더티 에비전인 경우 및 더 이상 쓰기 순서 저장을 수행할 수 없는 경우 중 어느 하나의 경우, 상기 캐시 컨트롤러(200)는 상기 영구적 저장 순서로 쓰기 명령(또는 상기 순서대로 저장된 쓰기 명령)을 하위 메모리로 실행(또는 순서대로 저장된 쓰기 명령을 순차적으로 하위 메모리에 저장)한다. 여기서, 상기 하위 메모리는 복수의 메모리 계층으로 구성된 상기 캐시 메모리(300)에 대해서, 순서대로 복수의 쓰기 명령이 저장된 특정 메모리 계층에서 상기 캐시 메모리(300) 내에서의 복수의 쓰기 명령이 저장된 해당 메모리 계층보다 아래 단

의 하나 이상의 메모리 계층과 주 메모리(500)를 나타낸다.

- [0093] 즉, 상기 캐시 컨트롤러(200)는 캐시 적중(cache hit)의 발생 여부를 판단(또는 확인)한다(S641).
- [0094] 판단 결과, 상기 캐시 적중이 발생한 경우, 상기 캐시 컨트롤러(200)는 해당 캐시 적중이 쓰기 및 더티 적중(write and dirty hit)인지 여부(또는 해당 캐시 적중이 쓰기 적중이면서 더티 적중인지 여부)를 확인한다(S642).
- [0095] 확인 결과, 상기 캐시 적중이 쓰기 및 더티 적중이 아닌 경우, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300)에 대한 접근 기능을 수행한다(S643).
- [0096] 또한, 확인 결과, 상기 캐시 적중이 쓰기 및 더티 적중인 경우, 상기 캐시 컨트롤러(200)는 상기 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 적중(hit)이 된 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리(300)를 거쳐 상기 주 메모리(500)까지 메모리 쓰기 순서를 순서대로 수행한다(S644).
- [0097] 또한, 판단 결과, 상기 캐시 적중이 발생하지 않은 경우, 상기 캐시 컨트롤러(200)는 더티 에빅션(dirty eviction)의 발생 여부를 확인한다(S645).
- [0098] 확인 결과, 상기 더티 에빅션이 발생한 경우, 상기 캐시 컨트롤러(200)는 상기 순서대로 저장된 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티م(victim) 주소까지 순서대로 하위 메모리로 쓰기 명령을 실행하여 상기 캐시 메모리(300)를 거쳐 상기 주 메모리(500)까지 메모리 쓰기 순서를 순서대로 수행한다(S646).
- [0099] 또한, 확인 결과, 상기 더티 에빅션이 발생하지 않은 경우, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300)에 대한 접근 기능을 수행한다(S647).
- [0100] 일 예로, 상기 제 11 메모리 요청에 대한 쓰기 순서의 기록이 완료된 이후, 상기 캐시 컨트롤러(200)는 상기 캐시 적중의 발생 여부를 판단한다. 이후, 상기 판단 결과, 상기 캐시 적중이 발생할 때, 상기 캐시 컨트롤러(200)는 상기 캐시 적중이 쓰기 및 더티 적중인지 여부를 확인한다. 이후, 상기 캐시 적중이 쓰기 및 더티 적중일 때, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300) 내의 제 1 레벨 캐시에 순서대로 저장된 복수의 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소(예를 들어 제 1 메모리 요청 주소)부터 적중(hit)이 된 주소(예를 들어 제 11 메모리 요청 주소)까지 순서대로(예를 들어 제 1 메모리 요청 주소 내지 제 11 메모리 요청 주소까지 순서대로) 상기 캐시 메모리(300)에 포함된 복수의 레벨 중 상기 제 1 레벨 캐시의 하위 메모리인 제 2 레벨 캐시에 쓰기 명령을 실행한다. 이후, 상기 캐시 메모리(300)에 포함된 복수의 레벨 중 마지막 레벨의 캐시가 제 2 레벨 캐시인 상태에서 제 2 레벨 캐시가 영구적 저장 순서를 순서대로 실행하는 단계의 조건에 도달할 때, 상기 캐시 컨트롤러(200)는 주 메모리(500)에 상기 제 2 레벨 캐시의 쓰기 명령을 기록된 순서대로 수행한다.
- [0101] 다른 일 예로, 상기 제 11 메모리 요청에 대한 쓰기 순서의 기록이 완료된 이후, 상기 캐시 컨트롤러(200)는 상기 캐시 적중의 발생 여부를 판단한다. 이후, 상기 판단 결과, 상기 캐시 적중이 발생하지 않을 때, 상기 캐시 컨트롤러(200)는 더티 에빅션(dirty eviction)의 발생 여부를 확인한다. 이후, 상기 확인 결과, 상기 더티 에빅션이 발생할 때, 상기 캐시 컨트롤러(200)는 상기 캐시 메모리(300) 내의 제 1 레벨 캐시에 순서대로 저장된 복수의 쓰기 명령 중에서 가장 기록이 오래된 쓰기 명령의 주소부터 빅티م(victim) 주소까지 순서대로 상기 캐시 메모리(300)에 포함된 복수의 레벨 중 상기 제 1 레벨 캐시의 하위 메모리인 제 2 레벨 캐시에 쓰기 명령을 실행한다. 이후, 상기 캐시 메모리(300)에 포함된 복수의 레벨 중 마지막 레벨의 캐시가 제 2 레벨 캐시인 상태에서 제 2 레벨 캐시가 영구적 저장 순서를 순서대로 실행하는 단계의 조건에 도달할 때, 상기 캐시 컨트롤러(200)는 주 메모리(500)에 상기 제 2 레벨 캐시의 쓰기 명령을 기록된 순서대로 수행한다(S640).
- [0102] 본 발명의 실시예는 앞서 설명된 바와 같이, 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템(또는 메모리 관리 장치)에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리하여, 전원을 켜면 이전의 작업 내용으로 바로 복구가 가능한 즉시 부팅을 실현할 수 있으며, 전원 차단 또는 시스템 충돌에 의한 복구가 필요한 상황에서도 주 메모리로 영구 메모리를 사용함에 있어 주 메모리 차원에서 복구가 가능할 수 있다.
- [0103] 또한, 본 발명의 실시예는 앞서 설명된 바와 같이, 프로세서와 주 메모리 사이에 위치하고 있는 캐시 메모리에서 영구적 저장 순서를 제공하여, 메모리의 명령의 순서를 관리함에 따라 프로그램의 디버깅에 활용 시 오류가 난 지점 직전까지 바로 접근이 가능하며, 디버깅을 위한 프로그램 재실행 과정을 생략할 수 있다.

[0104] 전술된 내용은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 수정 및 변형이 가능할 것이다. 따라서, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 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 권리범위에 포함되는 것으로 해석되어야 할 것이다.

산업상 이용가능성

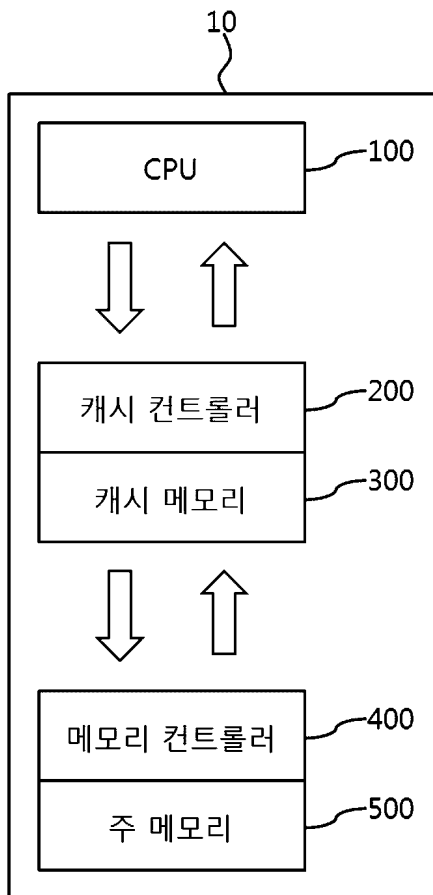
[0105] 본 발명은 주 메모리로 영구 메모리를 사용하는 컴퓨터 시스템에서 주 메모리의 메모리 쓰기 명령 순서가 프로세서에서의 메모리 쓰기 명령 순서와 일치하도록 캐시 메모리를 관리하는 영구 메모리의 영구적 저장 순서를 제 공함으로써 전원을 켜면 이전의 작업 내용으로 바로 복귀가 가능한 즉시 부팅을 실현할 수 있으며, 전원 차단 또는 시스템 충돌에 의한 복구가 필요한 상황에서도 주 메모리로 영구 메모리를 사용함에 있어 주 메모리 차원에서 복구가 가능한 것으로, 동적 메모리 장치 분야, 메모리 분야 등에서 광범위하게 이용될 수 있다.

부호의 설명

[0106] 10: 메모리 관리 장치 100: CPU
200: 캐시 컨트롤러 300: 캐시 메모리
400: 메모리 컨트롤러 500: 주 메모리

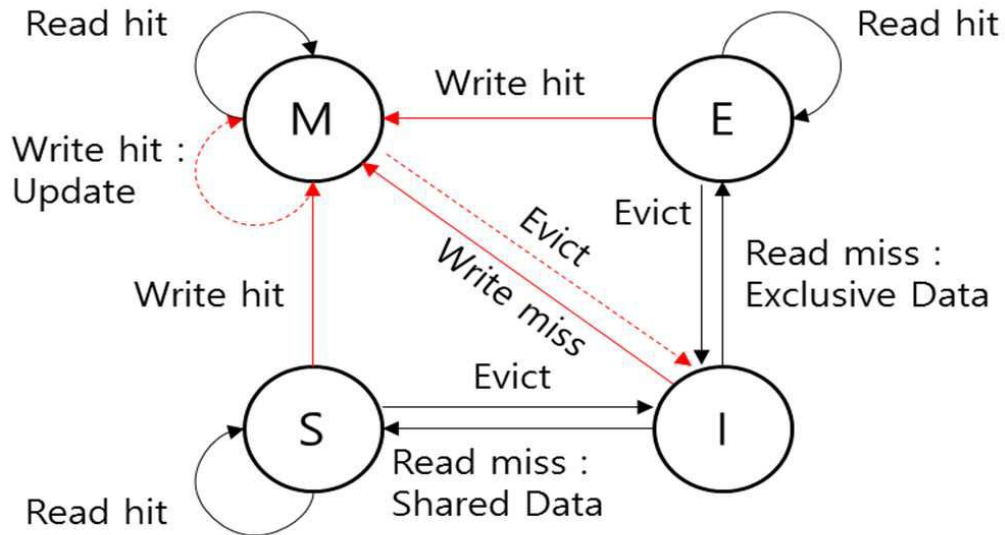
도면

도면1



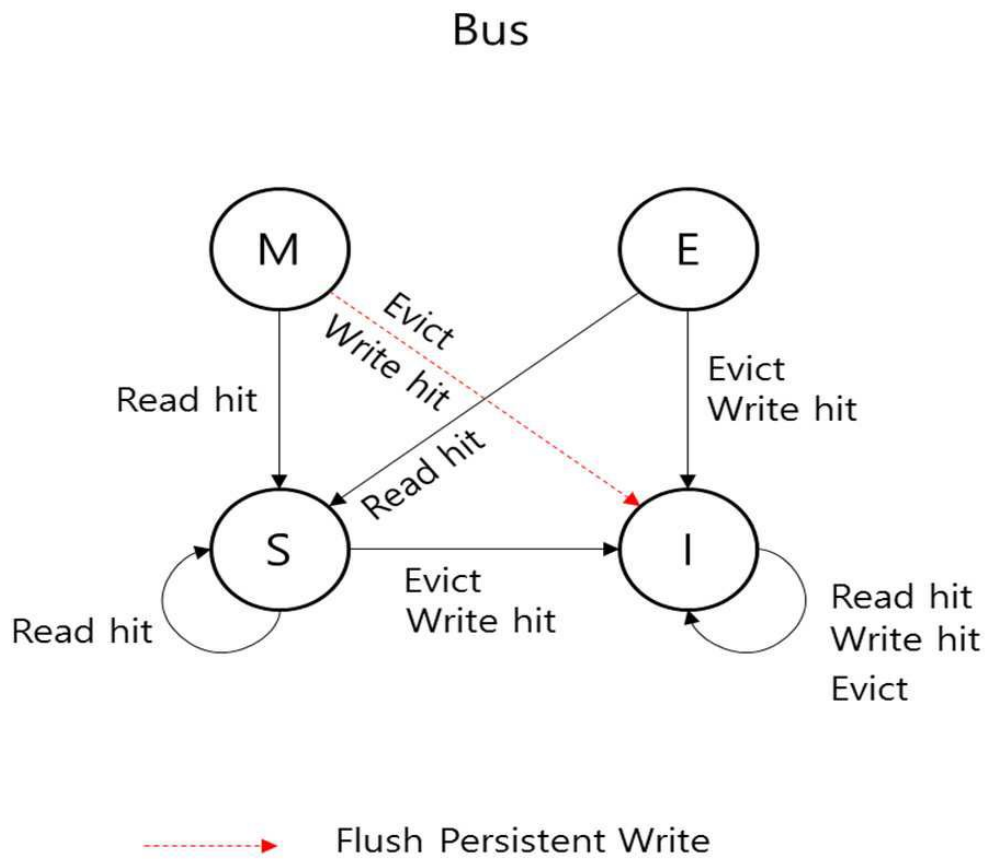
도면2

Local

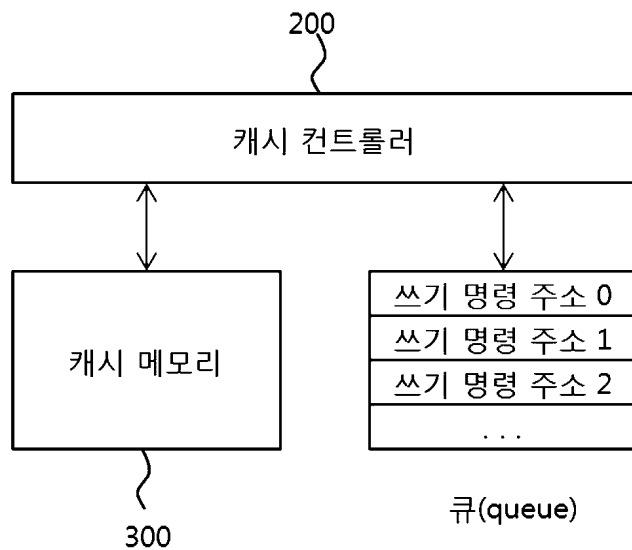


- Insert to Persistent Write List
- - -→ Flush Persistent Write

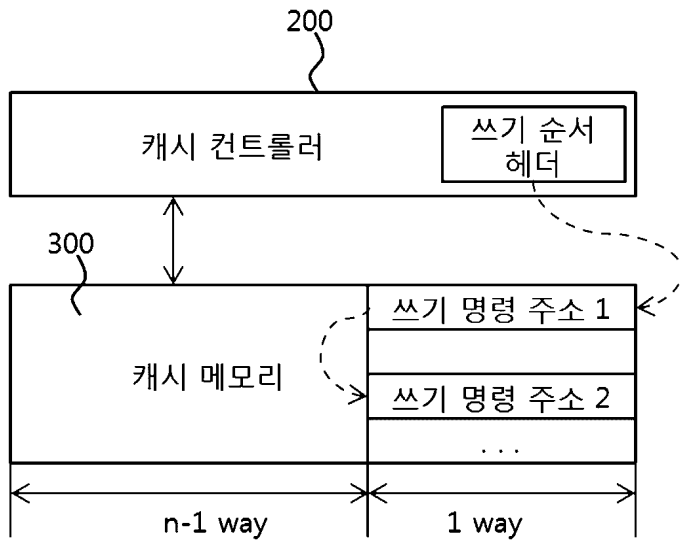
도면3



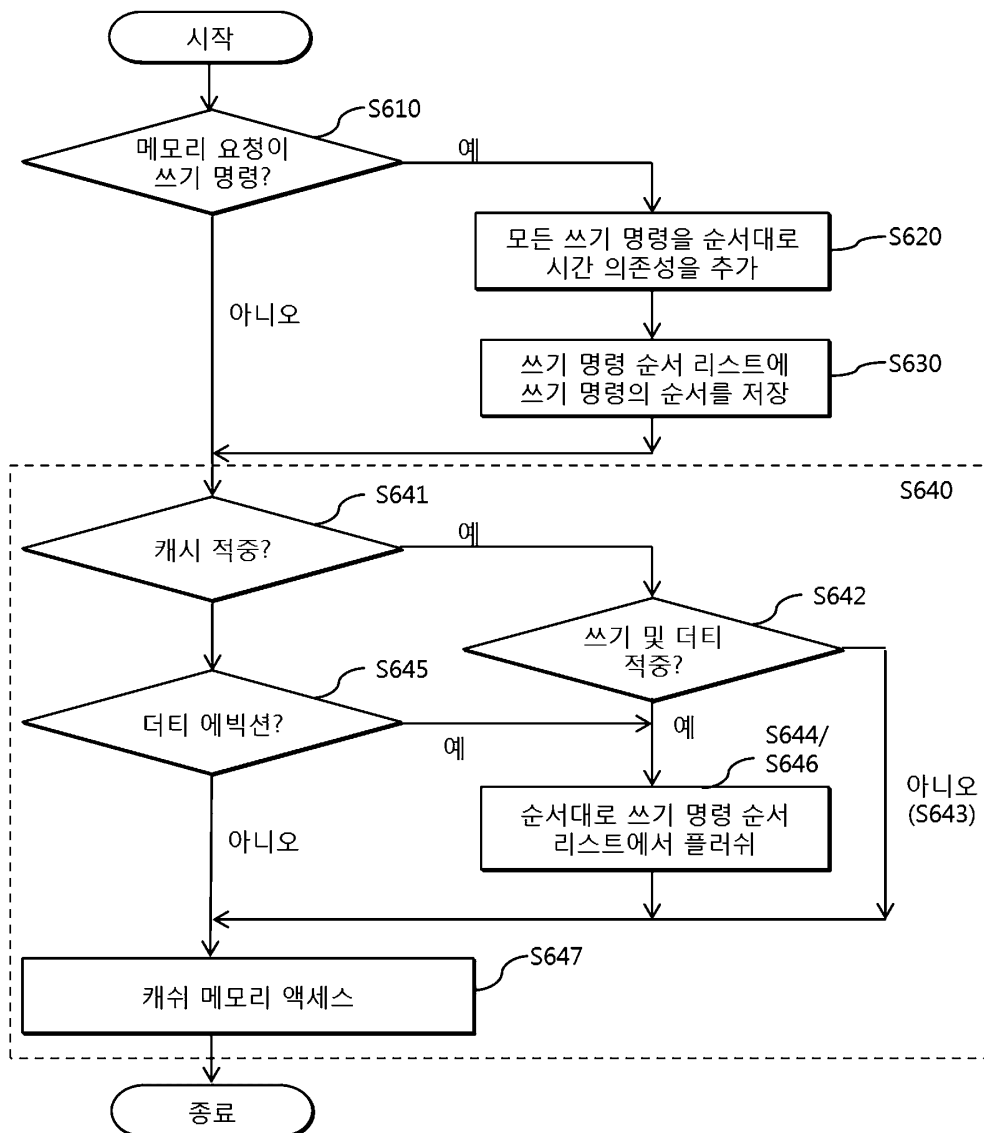
도면4



도면5



도면6



도면7

