



**(19) 대한민국특허청(KR)**  
**(12) 등록특허공보(B1)**

(45) 공고일자 2017년03월07일  
 (11) 등록번호 10-1713051  
 (24) 등록일자 2017년02월28일

(51) 국제특허분류(Int. Cl.)  
 G06F 12/02 (2006.01) G06F 12/10 (2016.01)  
 G06F 13/16 (2006.01)  
 (21) 출원번호 10-2010-0120087  
 (22) 출원일자 2010년11월29일  
 심사청구일자 2015년11월25일  
 (65) 공개번호 10-2012-0058352  
 (43) 공개일자 2012년06월07일  
 (56) 선행기술조사문헌  
 US20100077136 A1\*  
 US20090106480 A1\*  
 US20070156964 A1\*  
 US20090307430 A1\*  
 \*는 심사관에 의하여 인용된 문헌

(73) 특허권자  
**삼성전자주식회사**  
 경기도 수원시 영통구 삼성로 129 (매탄동)  
 (72) 발명자  
**이동양**  
 경기도 용인시 수지구 진산로66번길 10, 진산마을 526-308 (풍덕천동)  
**최재영**  
 경기도 화성시 병점2로 102, 신창비바훼밀리2차 201동 206호 (병점동)  
**황주영**  
 경기도 수원시 영통구 청명로 132, 청명마을3단지 삼익아파트 324동 1201호 (영통동)  
 (74) 대리인  
**한지희, 윤재석, 권영규**

전체 청구항 수 : 총 8 항

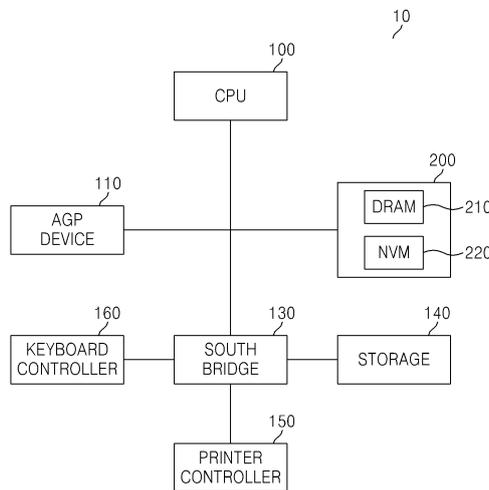
심사관 : 김경완

(54) 발명의 명칭 **하이브리드 메모리 시스템, 및 그 관리 방법**

**(57) 요약**

하이브리드 메모리 시스템, 및 그 관리 방법이 개시된다. 본 발명의 하이브리드 메모리 시스템은 중앙처리장치, 사용자 데이터 및 코드 데이터를 저장하는 스토리지 장치; 및 휘발성 메모리; 및 비휘발성 메모리를 포함하며, 상기 스토리지 장치로부터 상기 중앙처리장치의 동작을 수행하는데 필요한 데이터를 적재하여 저장하는 메인 메모리를 구비하며, 상기 휘발성 메모리의 일부는 상기 비휘발성 메모리에 저장된 데이터에 대한 캐쉬로 할당된다.

**대표도** - 도1



## 명세서

### 청구범위

#### 청구항 1

중앙처리장치;

사용자 데이터 및 코드 데이터를 저장하는 스토리지 장치; 및

휘발성 메모리; 및 비휘발성 메모리를 포함하며, 상기 스토리지 장치로부터 상기 중앙처리장치의 동작을 수행하는데 필요한 데이터를 적재하여 저장하는 메인 메모리를 구비하며,

상기 휘발성 메모리의 일부는 상기 비휘발성 메모리에 저장된 데이터에 대한 캐쉬로 할당되며,

상기 중앙처리 장치에 의해 구동되는 OS는

페이지 테이블의 정보에 기초하여, 리드 온리 속성을 가진 데이터는 상기 비휘발성 메모리에 저장하고, 리드 및 라이트 속성을 가진 데이터는 상기 휘발성 메모리에 저장하는 것을 특징으로 하는 하이브리드 메모리 시스템.

#### 청구항 2

삭제

#### 청구항 3

제1항에 있어서, 상기 페이지 테이블은

각각이 가상 주소 필드, 상기 가상 주소 필드에 대응하는 물리적 주소 필드, 프레젠탈 필드 및 속성 필드를 갖는 복수의 엔트리들을 포함하며,

상기 프레젠탈 필드는 상기 가상 주소에 상응하는 데이터가 상기 물리적 주소에 상응하는 메모리에 적재되어 있는지를 나타내고,

상기 속성 필드는 상기 데이터의 속성을 나타내는 하이브리드 메모리 시스템.

#### 청구항 4

제3항에 있어서, 상기 OS는

상기 프레젠탈 필드가 무효인 가상 주소에 해당하는 물리적 주소 생성이 요청되면, 상기 스토리지 장치에 있는 데이터를 상기 데이터의 타입이나 속성에 따라 상기 휘발성 메모리의 프리 프레임 또는 상기 비휘발성 메모리의 프리 프레임에 선택적으로 적재하고, 해당 프레임 정보를 상기 페이지 테이블에 업데이트하는 하이브리드 메모리 시스템.

#### 청구항 5

제1항에 있어서,

상기 휘발성 메모리의 캐쉬에 저장된 데이터 중 LRU(Least Recently Used) 데이터는 상기 비휘발성 메모리로 이송되는 것을 특징으로 하는 하이브리드 메모리 시스템.

#### 청구항 6

제1항에 있어서, 상기 캐쉬는

리드 온리 파일 데이터를 저장하기 위한 파일 캐쉬 및

응용 데이터를 저장하기 위한 응용 데이터 캐쉬를 포함하는 하이브리드 메모리 시스템.

**청구항 7**

휘발성 메모리 및 비휘발성 메모리를 포함하는 하이브리드 메모리 시스템에서의 메모리 관리자에 의한 하이브리드 메모리의 관리 방법에 있어서,

상기 휘발성 메모리의 일부를 상기 비휘발성 메모리의 캐쉬 영역으로 할당하는 단계;

스토리지 장치로부터 상기 하이브리드 메모리에 데이터의 적재가 필요한 경우, 상기 적재될 데이터의 속성 또는 타입에 따라 상기 데이터를 상기 휘발성 메모리 또는 상기 비휘발성 메모리에 선택적으로 적재(loading)하는 단계;

상기 비휘발성 메모리에 적재하는 데이터를 상기 휘발성 메모리의 캐쉬 영역에도 적재하는 단계; 및

상기 캐쉬 영역이 모두 차면, 상기 캐쉬 영역의 일부 데이터를 상기 비휘발성 메모리로 이송하는 단계를 포함하는 하이브리드 메모리의 관리 방법.

**청구항 8**

삭제

**청구항 9**

제7항에 있어서, 상기 일부 데이터는

상기 캐쉬 영역에 저장된 데이터 중 LRU(Least Recently Used) 데이터인 더 포함하는 하이브리드 메모리의 관리 방법.

**청구항 10**

제7항에 있어서,

상기 데이터의 속성은

RO(리드 온리) 및 RW(read/write)중 하나이고,

상기 데이터의 타입은

코드, 스택(Stack), 힙(Heap), 라이브러리를 위한 버퍼 및 일시적 파일을 위한 버퍼 중 하나인 하이브리드 메모리의 관리 방법.

**발명의 설명**

**기술 분야**

[0001] 본 발명은 메모리 장치에 관한 것으로, 보다 상세하게는 비휘발성 메모리와 휘발성 메모리를 가진 하이브리드 메모리 시스템 및 이를 관리하는 방법에 관한 것이다.

**배경 기술**

[0002] 컴퓨터 핵심 구성 요소 중의 하나인 메모리로 사용되고 있는 디램(DRAM)과 관련하여, 용량의 지속적인 증가에 따른 전력 소모(특히 대기 전력) 증가, 휘발성인 DRAM에 저장된 정보의 손실을 방지하기 위한 스토리지로의 백업의 중요성 증가, 장기적으로 DRAM의 비트 성장 지속 가능성에 대한 의문 등 메모리 및 메모리 시스템 관련 다양한 논의가 진행되고 있다.

[0003] 그 일환으로, DRAM의 대체재로서 비휘발성 메모리를 활용하는 방안에 대한 연구와 DRAM과 비휘발성 메모리를 함께 사용하는 하이브리드 메모리에 대한 연구가 진행되고 있다.

[0004] PRAM은 랜덤 액세스(Random Access)가 가능하고, 비휘발성이면서, DRAM과NAND 플래시 메모리의 중간에 위치한 성능을 가진 장점이 있지만, 여전히 리/라이트 레이턴시(Read/Write Latency)는 DRAM 보다 길면서 에너지 효율은 낮은 문제를 가지고 있다. 그리고 쓰기 인듀어런스 싸이클(Write Endurance Cycle)도 제한적인 상황이다. 따라서 이와 같은 특성을 가진 PRAM으로 DRAM을 그대로 대체하는 것은 불가능 할 것이다.

[0005] 따라서, 비휘발성 메모리(예컨대, PRAM)으로 DRAM을 대체 하고자 한다면, 두 가지 핵심 문제를 해결해야 한다. 첫째로 DRAM 대비 훨씬 낮은 수준의 인듀어런스(Endurance) 특성 해결이며, 다음으로는 낮은 쓰기 성능을 극복하는 것이다. 이러한 문제점을 극복하기는 쉽지 않으므로, 실제로 PRAM이 DRAM을 대체하기는 쉽지 않을 것으로 예상된다.

[0006] 다음으로, DRAM과 PRAM을 같이 사용하는 하이브리드 메모리에 대한 연구를 살펴본다. DRAM과 PRA을 같이 사용하는 경우 DRAM과 PRAM이 구분 관리되지 못하면 PRAM의 인듀어런스 문제를 해결하기 어렵고, 또한 PRAM의 비휘발성의 활용도 어렵게 된다.

[0007] 이를 해결하기 위한 종래의 방법으로, PRAM과 DRAM의 구분 활용에 적합한 페이지 스와퍼를 개발하고, 이를 이용하여 쓰기의 발생 빈도에 따라 PRAM이나 DRAM에 페이지를 할당하는 방법이 있다. 이 종래 기술에 의하면, PRAM으로 구성된 메모리 영역을 페이지 단위로 관리하고, 모든 페이지에 대하여 쓰기된 숫자를 카운트하고, 이 숫자를 PRAM에 저장하여 시스템에 폐기 될 때까지 유지하며, 이 정보를 활용하여 쓰기가 많은 페이지의 물리적 주소를 DRAM 영역으로 매핑 하도록 하였다. 이를 통하여 PRAM의 인듀어런스에 따른 문제 가능성을 방지하고, DRAM을 적절히 활용하도록 하여 PRAM 단점을 보완 하였다. 하지만, 이 방법에 의하더라도 성능 향상과 전력 소모 절감에서의 효과가 미미할 예상된다.

### 발명의 내용

#### 해결하려는 과제

[0008] 따라서, 본 발명이 이루고자 하는 기술적인 과제는 비휘발성 메모리를 휘발성 메모리의 보완재로 사용하여 메모리 용량을 증가 시키고, 또한 메모리 용량 증가에 따른 전력 소모의 증가를 줄일 수 있는 하이브리드 메모리 시스템 및 그 관리 방법을 제공하는 것이다.

#### 과제의 해결 수단

[0009] 상기 기술적 과제를 달성하기 위한 본 발명의 일 실시예에 따르면, 하이브리드 메모리 시스템, 및 그 관리 방법이 제공된다. 본 발명의 실시예에 따른 하이브리드 메모리 시스템은 중앙처리장치, 사용자 데이터 및 코드 데이터를 저장하는 스토리지 장치; 및 휘발성 메모리; 및 비휘발성 메모리를 포함하며, 상기 스토리지 장치로부터 상기 중앙처리장치의 동작을 수행하는데 필요한 데이터를 적재하여 저장하는 메인 메모리를 구비하며, 상기 휘발성 메모리의 일부는 상기 비휘발성 메모리에 저장된 데이터에 대한 캐쉬로 할당된다.

#### 발명의 효과

[0010] 본 발명의 일 실시예의 하이브리드 메모리 시스템에 의하면, 프로그램의 수행에 따른 워킹 셋으로 인해 휘발성 메모리 공간이 부족해 지면, 프로세싱이 종료된 응용 프로그램 데이터를 휘발성 메모리에서 비휘발성 메모리로 페이지 이송(Page Migration)하여 프로그램 재 실행 시 스토리지 액세스 이벤트를 줄일 수 있다. 또한 빠른 동작을 위해 휘발성 메모리 영역에 응용 프로그램 데이터를 위한 캐쉬를 할당하여, 비휘발성 메모리로의 페이지 이송(Page Migration)시 발생하는 성능 저하를 줄일 수 있다.

#### 도면의 간단한 설명

- [0011] 도 1은 본 발명의 일 실시예에 따른 하이브리드 메모리 시스템의 개략적인 구성 블록도이다.  
 도 2는 본 발명의 일 실시예에 따른 OS와 프로세스들 간의 관계를 도면이다.  
 도 3은 본 발명의 일 실시예에 따른 가상 메모리와 물리적 메모리 간의 관계를 나타내는 도면이다.  
 도 4는 본 발명의 실시예에 따른 페이지 테이블의 일 예를 나타내는 도면이다.  
 도 5는 본 발명의 실시예에 따른 페이지 폴트시의 동작을 설명하기 위한 도면이다.  
 도 6a 및 도 6b는 각각 본 발명의 일 실시예에 따른 메모리에서의 프로그램 및 데이터 배치를 설명하기 위한 도면이다.  
 도 7은 본 발명의 일 실시예에 따른 페이지 타입별 페이지 할당 영역을 나타내는 표이다.  
 도 8은 통상의 PC에서 메모리 쓰기 특성을 측정한 결과를 나타내는 그래프이다.

도 9는 본 발명의 일 실시예에 따른 데이터 타입에 따른 초기 할당 및 이송 정책을 나타내는 표이다.

도 10a는 통상의 메인 메모리의 구성의 일 예를 나타내는 도면이다.

도 10b 및 도 10c는 각각 본 발명의 실시예에 따른 하이브리드 메모리의 구성의 일 예를 나타내는 도면이다.

도 11은 본 발명의 다른 일 실시예에 따른 하이브리드 메모리 시스템의 구성 블록도이다.

**발명을 실시하기 위한 구체적인 내용**

- [0012] 본 명세서 또는 출원에 개시되어 있는 본 발명의 실시 예들에 대해서 특정한 구조적 내지 기능적 설명들은 단지 본 발명에 따른 실시 예를 설명하기 위한 목적으로 예시된 것으로, 본 발명에 따른 실시 예들은 다양한 형태로 실시될 수 있으며 본 명세서 또는 출원에 설명된 실시 예들에 한정되는 것으로 해석되어서는 아니 된다.
- [0013] 본 발명에 따른 실시 예는 다양한 변경을 가할 수 있고 여러 가지 형태를 가질 수 있으므로 특정 실시 예들을 도면에 예시하고 본 명세서 또는 출원에 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시 예를 특정한 개시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.
- [0014] 제1 및/또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 이탈되지 않은 채, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.
- [0015] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.
- [0016] 본 명세서에서 사용한 용어는 단지 특정한 실시 예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0017] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미이다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미인 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0018] 이하, 첨부한 도면을 참조하여 본 발명의 바람직한 실시 예를 설명함으로써, 본 발명을 상세히 설명한다. 각 도면에 제시된 동일한 참조부호는 동일한 부재를 나타낸다.
- [0019] 도 1은 본 발명의 일 실시예에 따른 하이브리드 메모리 시스템의 개략적인 구성 블록도이다. 도 1에 도시된 실시예는 하이브리드 메모리가 적용된 컴퓨팅 시스템의 일 예이다.
- [0020] 이를 참조하면, 본 발명의 일 실시예에 따른 컴퓨팅 시스템(10)은 중앙 처리 장치(Central Processing Unit, 100), AGP 장치(Accelerated Graphics Port, 110), 메인 메모리(200), 스토리지 장치(예컨대, SSD, HDD 등, 140), 사우스 브리지(130), 키보드 컨트롤러(160), 및 프린터 컨트롤러(150) 등을 포함한다. 컴퓨팅 시스템(10)은 개인용 컴퓨터 또는 노트북 컴퓨터의 블록도일 수 있다. 그러나 본 발명의 범위가 이에 한정되는 것은 아니다.
- [0021] 컴퓨팅 시스템(10)에서 AGP 장치(110), 및 메인 메모리(200) 등은 CPU(100)에 직접 접속되거나, 노스 브리지(미도시)를 통해 CPU(100)에 접속될 수도 있다. 즉, 본 실시예에서는, 노스 브리지(미도시)가 CPU(100)에 포함된 형태일 수 있으나, 본 발명이 이에 한정되는 것은 아니다.

- [0022] 스토리지 장치(140), 키보드 컨트롤러(160), 프린터 컨트롤러(150), 및 각종 주변 장치들(미도시) 등은 사우스 브리지(130)에 연결된다.
- [0023] AGP(110)는 3차원 그래픽 표현을 빠르게 구현할 수 있게 해주는 버스 규격이다. AGP 장치(110)에는 모니터 이미지를 재생하는 비디오 카드 등이 포함될 수 있다.
- [0024] 중앙처리장치(100)는 컴퓨팅 시스템(10)의 구동에 필요한 각종 연산을 수행하고, 또한 OS 및 응용 프로그램을 실행한다. 스토리지 장치(140)는 사용자 데이터 및 코드 데이터를 저장하는 대용량 데이터 저장 장치로서, HDD, SSD 등일 수 있으나, 이에 한정되는 것은 아니다.
- [0025] 메인 메모리(200)는 상기 스토리지 장치(140)로부터 상기 중앙처리장치(100)의 동작을 수행하는데 필요한 데이터를 적재하여 저장한다. 메인 메모리(200)는 휘발성 메모리(210)와 비휘발성 메모리(220)를 포함하는 하이브리드 타입의 메모리이다. 휘발성 메모리(210)는 DRAM(Random Access Memory)일 수 있고, 비휘발성 메모리(220)는 PRAM(Phase-change andom Access Memory), FRAM(Ferroelectric andom Access Memory), MRAM(Magnetic Random Access Memory) 등 일 수 있으나 본 발명의 범위가 이에 한정되는 것은 아니다.
- [0026] 또한, 본 발명의 일 실시예에 따른 컴퓨팅 시스템(10)에서는 스토리지 장치(140)가 사우스 브리지(130)에 연결되는 구조이나, 이에 한정되는 것은 아니며, 스토리지 장치(140)가 노스 브리지(미도시)에 연결되거나, CPU(100)에 직접 연결되는 구조일 수도 있다.
- [0027] 도 2는 본 발명의 일 실시예에 따른 OS와 프로세스들 간의 관계를 도면이다.
- [0028] 도 2를 참조하면, 컴퓨팅 시스템의 OS(310)는 컴퓨팅 시스템의 하드웨어와 응용프로그램(이하 프로세스(301, 302, 303)라고도 함)간의 인터페이스 역할을 하면서 CPU(100), 메인 메모리(200) 등을 관리한다. OS(310)는 커널(311)를 포함하며, 커널(311)은 메모리 관리자(312)를 포함한다.
- [0029] 메모리 관리자(312)는 하드브리드 타입의 메인 메모리(200)를 관리한다.
- [0030] 도 3은 본 발명의 일 실시예에 따른 가상 메모리와 물리적 메모리 간의 관계를 나타내는 도면이다.
- [0031] 이를 참조하면, 컴퓨터의 메모리 관리는 가상 메모리(Virtual Memory)와 물리적 메모리(Physical Memory)로 나누어 수행되고 있다. 가상 메모리 시스템은 실행되는 여러 개의 프로세스들이 하나의 물리적인 메모리 공간을 공유하도록 하여, 주어진 시스템이 가진 물리적인 메모리 양보다 많은 메모리를 가진 것처럼 하는 효과가 있다. 즉, 주어진 물리적인 메모리 보다 큰 프로그램도 실행이 가능하도록 해 주며, 또한 동시에 여러 개의 프로그램 또는 프로세스가 효율적으로 실행될 수 있도록 해 주는 중요한 수단이 되고 있다. 예를 들어 주어진 시스템의 메모리 용량이 1GB (Giga Byte) 이더라도, 가상 메모리 공간은 32bit OS의 경우  $2^{32}$ 에 해당하는 4GB, 64bit OS 라면  $2^{64}$ 에 해당하는 16EB(Exa Byte)가 되어, 이 용량을 넘지 않는 한 어떤 프로그램도 1GB의 메모리에서 실행할 수 있게 된다. 시스템에서 실행되는 각 프로세스는 개별적으로 독립된 가상 주소(Virtual Address) 공간을 갖고 있는데, 이들 가상 주소 영역은 서로 분리되어 있기 때문에 실행되는 프로세스는 다른 프로세스에 영향을 주지도 받지도 않게 된다. 하지만 프로세스가 실행되기 위해서는 궁극적으로는 모든 프로세스가 임의의 필요한 시간에 물리적 메모리에 임시로 저장되게 된다. 따라서 메모리 관리자는 가상 주소를 사용하는 프로세스를 적절하게 물리적 주소(Physical Address)에 배치 할 수 있어야 하며, 물리적 주소 변환도 알 수 있는 방법이 필요하다.
- [0032] 이와 같은 메모리 관리는 OS(310)의 메모리 관리자(312)가 수행하는데, OS(310)는 가상 메모리의 페이지(page)를 물리적 메모리에 위치 시킨다.
- [0033] 물리적 메모리는 도 1에 도시된 메인 메모리(200)에 상응한다. 본 실시예에 따르면, OS(310)는 메인 메모리(200)에 포함된 휘발성 메모리(210)와 비휘발성 메모리(220)를 하나의 물리적 메모리로 인식한다. 따라서, 본 발명의 일 실시예에 따른 물리적 메모리는 휘발성 메모리(210)와 비휘발성 메모리(220)를 포함하며, 물리적 메모리에 적재될 데이터의 타입이나 속성에 따라, 휘발성 메모리(210)나 비휘발성 메모리(220)로 선택적으로 적재될 수 있다.
- [0034] 페이지는 가상 메모리를 분할 관리하는 단위이며, 프레임(Frame)은 물리적 메모리를 분할 관리하는 단위인데, 페이지와 프레임의 크기는 동일하다. 멀티-프로세스 환경에서는 여러 개의 프로세스가 동시에 메모리에 상주해야 하는데, 멀티-프로세스 조건이기 때문에 프로세스의 페이지 수는 프레임 수보다 훨씬 많게 된다. 따라서 동시에 모든 페이지를 물리적 메모리 상주시키는 것은 불가능하다. 이를 해결하기 위하여 가상 메모리 페이지 가

운데 현재 수행에 필요한 페이지만 물리적 메모리에 적재하는 방식을 사용한다.

- [0035] 도 3은 두 개의 프로세스(Process A, B)가 하나의 물리적 메모리 공간을 공유하면서 현재 실행에 필요한 가상 메모리 페이지를 물리적 메모리 프레임에 상주시키는 예이다. 도 3에서 보는 바와 같이 모든 페이지가 한꺼번에 적재되지 않으며, 임의의 한 페이지는 물리적 메모리의 어떤 프레임이라도 적재가 가능하다. 가상 메모리 주소는 연속적인 공간으로 보이지만, 실제 물리적 주소 상에는 페이지 할당자(Allocator)에 의해서 임의의 어떤 프레임에도 적재가 될 수 있기 때문에, 어느 페이지가 물리적 메모리의 어느 프레임에 적재되었는지를 알 수 있는 방법이 필요하게 된다. 이 역할을 하는 것을 페이지 테이블이라고 하며, 이 페이지 테이블은 가상 주소 영역의 페이지와 CPU의 MMU(Memory management Unit)를 통해서 변환된 물리적 주소 영역의 프레임 위치를 기록하고, 이들에 관한 정보를 저장한다.
- [0036] 도 4는 본 발명의 실시예에 따른 페이지 테이블의 일 예를 나타내는 도면이다. 이를 참조하면, 페이지 테이블(230)은 다수(예컨대, 페이지 수만큼)의 엔트리를 포함하며, 각 엔트리는 가상 주소 필드(231), 이에 대응하는 물리적 주소 필드(232), 프레젠탈 필드(233) 및 속성 필드(234)를 포함할 수 있다.
- [0037] 구체적으로, 도 4는, 하나의 페이지 테이블(table)을 이용하여 가상 주소(Virtual Address)를 가지는 프로세스의 페이지가 물리적 메모리 영역의 프레임으로 매핑되는 예이다. 페이지 테이블(230)의 엔트리(entry)는 페이지마다 하나씩 대응 된다. 도 4에서 보는 바와 같이 가상 주소영역에 속하는 페이지 0, 2, 5 가 물리적 메모리의 4, 6, 9 프레임에 각각 적재되어 있으며, 페이지 테이블 엔트리에는 이를 보여주는 정보가 담겨 있다.
- [0038] 도 4에서 유효(Valid)-무효(Invalid)를 표시하는 필드(233)를 프레젠탈 필드(Present Field, 233)라고 하는데, 어떤 임의의 가상 페이지가 프레임에 적재되어 있는지 여부를 나타내고 있으며, 적재되어 있으면 1 (Valid)로, 적재 되어 있지 않으면 0 (Invalid)로 기록한다. 프로그램이 실행되면 CPU의 MMU는 가상 메모리 주소를 이용하여 물리적 메모리 주소를 만들어 내는데, 이때 "0"(Invalid) 페이지에 해당하는 가상 메모리 주소 영역에 해당하는 물리적 메모리 주소 생성이 요청되면, 이 프레젠탈 비트를 참조하여 무효 상태(Invalid Status)로 인식하게 되는데 이를 페이지 폴트(page Fault)라고 한다.
- [0039] 페이지 폴트가 발생하면 CPU(100)는 익셉션 인터럽트(Exception Interrupt)을 발생 시킨 후, 프로그램 카운터에 해당하는 인스트럭션(Instruction)은 보류(Hold)되고, 해당 인스트럭션과 연관성이 없는 다음 인스트럭션이 수행되거나, 이에 해당하는 인스트럭션이 없으면 필요한 페이지가 메모리에 적재되어 사용할 수 있을 때까지 대기하게 된다.
- [0040] 도 5는 본 발명의 실시예에 따른 페이지 폴트시의 동작을 설명하기 위한 도면이다.
- [0041] 이를 참조하면, 페이지 폴트가 발생하면(S11), OS(310)의 익셉션 핸들러(Exception Handler)가 호출되어(S12), 스토리지 장치(140)에 있는 페이지를 물리적 메모리의 프리 프레임에 적재하고(S13, S14), 해당 프레임 정보를 페이지 테이블 엔트리에 업데이트한다(S15). 스토리지 장치(140)에 있는 페이지를 물리적 메모리의 프리 프레임에 적재할 때 페이지 타입(예컨대, 데이터 타입이나 속성)에 따라 물리적 메모리의 비휘발성 메모리 영역의 프리 프레임이 할당되거나, 휘발성 메모리 영역의 프리 프레임이 할당될 수 있다. 페이지 타입(예컨대, 데이터 타입이나 속성)은 속성 필드(234)를 참조하여 결정될 수 있다. 예컨대, 속성 필드(234)는 데이터 속성이 RO(리드 온리) 인지 RW(read/write)인지를 나타낼 수 있다. 이 경우 데이터 속성이 RO(리드 온리)인 경우 해당 페이지는 비휘발성 메모리 영역에 적재되고, 데이터 속성이 RW(read/write)인 경우 해당 페이지는 휘발성 메모리 영역에 적재될 수 있으나, 본 발명이 이에 한정되는 것은 아니며, 다양한 실시예에 대하여 후술된다.
- [0042] 페이지 테이블 엔트리 업데이트 후, 프레젠탈 비트 필드를 "1"로 수정하여 유효 상태를 만들고 익셉션 핸들러를 종료 한다. 이후 CPU에서 진행되지 못하고 보류되어 있던 인스트럭션은 요청한 주소의 정보를 받아서 계속하여 프로그램을 수행하게 된다(S16).
- [0043] 도 6a 및 도 6b는 각각 본 발명의 일 실시예에 따른 메인 메모리에서의 프로그램 및 데이터 배치를 설명하기 위한 도면이다. 도 7은 본 발명의 일 실시예에 따른 페이지 타입별 페이지 할당 영역을 나타내는 표이다.
- [0044] 통상의 DRAM으로만 구성된 메모리 대비, 서로 다른 특징을 가지는 휘발성메모리(예컨대, DRAM)과 비휘발성 메모리(예컨대, PRAM)으로 구성된 본 발명의 실시예에 따른 하이브리드 메모리를 포함하는 시스템에서, 휘발성메모리(예컨대, DRAM)과 비휘발성 메모리(예컨대, PRAM)의 장점을 활용하기 위해서는 각각에 상주(적재)할 내용을 효과적으로 관리하는 것이 필요하다.
- [0045] 가상 주소는 메모리 물리 주소로 변환될 때, 상술한 페이지 테이블을 참조하게 된다. 이 때 통상의 OS는 메모

리 주소 공간의 위치에 관계없이 비어 있는 페이지를 활용하여 새로운 프로그램 또는 데이터를 메모리에 상주 시키게 된다. 그러나, 본 발명의 실시예에 따른 OS의 메모리 관리자는 메인 메모리에 저장할 내용(즉, 데이터)의 타입이나 속성에 따라 휘발성 메모리(예컨대, DRAM)과 비휘발성 메모리(예컨대, PRAM) 영역으로 분리하여 저장할 수 있다.

- [0046] 도 6a에 도시된 바와 같이, 본 발명의 일 실시예에 따른 메인 메모리(200)는 휘발성 메모리(210, 예컨대, DRAM)과 비휘발성 메모리(220, 예컨대, PRAM) 영역으로 분리되고, 휘발성 메모리(210, 예컨대, DRAM)의 일부는 비휘발성 메모리(220)의 캐쉬 영역(211)으로 할당될 수 있다.
- [0047] PRAM의 쓰기 속도가 느린 특징을 감안하여 쓰기 이벤트(Write Event)가 발생하는 스택(Stack), 힙(Heap), 리드/라이트(Read/Write) 파일 등 OS(Operating System) 관련 데이터(OS data, 213)는 도 6b에 도시된 바와 같이 DRAM(210)으로 할당하고, PRAM(220)은 OS 및 응용 프로그램(Application)의 코드(Code, 222) 및 리드 온리 파일(RO file, 221), 데이터 베이스 등(223), 파일 시스템의 메타 데이터(FS meta data, 224), 응용 프로그램 데이터(App data, 225) 등을 위한 용도로 할당될 수 있다. 도 6b의 실시예에서도, DRAM(210)의 일부는 캐쉬(211, 212)로 할당될 수 있다.
- [0048] 프로그램의 수행을 위해 메모리 할당에 대한 요청이 들어올 경우, OS의 메모리 관리자(312)는 도 7과 같은 각 메모리 페이지의 타입 및 사용 용도에 따라 초기 메모리 할당 영역을 선택적으로 결정할 수 있다.
- [0049] 도 7에 도시된 표를 참조하면, 페이지 타입은 코드, 스택(Stack), 힙(Heap), 라이브러리를 위한 버퍼 및 일시적 파일을 위한 버퍼 등으로 구분될 수 있다.
- [0050] 예를 들어 프로그램 코드 페이지의 경우 페이지의 접근 빈도수는 낮고 대부분이 리드 액세스이기 때문에 PRAM에 할당되어 있을 경우에도 큰 문제가 없으나 스택(Stack) 페이지의 경우는 순간적인 접근 빈도수도 높고 쓰기 이벤트도 많이 발생하기 때문에 PRAM에 할당 할 경우는 많은 프로세스 또는 블로킹을 유발하고 전체 메모리 성능을 저하시킬 수 있다.
- [0051] 따라서, 본 발명의 실시예에서는 메모리 페이지를 할당 할 때 페이지 폴트 익셉션이 발생하면, 페이지 폴트가 발생한 메모리 페이지에 대한 가상 주소 정보를 사용하여 메모리 페이지 타입을 구별 한 후 실제 메모리 페이지의 물리적인 할당 영역을 비휘발성 메모리(220, 예컨대, PRAM)과 휘발성 메모리(210, 예컨대 DRAM) 사이에서 선택적으로 결정할 수 있다.
- [0052] 가상 메모리를 사용하는 컴퓨터에서 프로세스의 MMU (Memory Management Unit)는 물리적 주소를 만드는 역할을 한다. 프로세스가 가상 주소를 MMU에 넘겨 주면 MMU는 그 주소를 받아 물리적 주소로 전환하게 되는데, 이 때 가상 메모리 주소와 물리적 메모리 주소 사이의 변환을 위해 MMU는 TLB (Translation Look-aside Buffer)를 참조하고, TLB 미스(Miss)가 발생하면 페이지 테이블을 이용하여 변환 정보를 얻어 오게 된다.
- [0053] 프로그램 실행에 따른 워킹 셋(워킹 셋)의 변화로 인해 메모리 공간이 부족해 질 때, 통상의 시스템은 DRAM과 스토리지 장치 사이의 스와핑(Swapping)이 발생하게 되어 메모리, 즉 DRAM 용량이 충분하지 않을 경우 성능 저하를 유발하게 된다. 그러나, 본 발명의 실시예에 따른 하이브리드 메모리 시스템에서는 DRAM 과 PRAM이 모두 메모리 공간으로 인식되어 총 메모리 용량 증대 효과가 발생한다.
- [0054] 실제로는 메모리 영역이 지속적으로 사용되는 것이 아니라 지역성(Locality)를 보이는 성향이 있다. 따라서, 본 발명의 실시예에서는, 메모리에 상주시킬 내용 중, 상대적으로 자주 사용되는 부분은 DRAM 영역에 두고, 나머지는 PRAM 영역에 둬으로써 성능 저하 없이, 용량 증대 효과를 확보 할 수 있다.
- [0055] 이와 같은 일반적인 지역성에 추가하여, PRAM은 쓰기 인듀어런스 싸이클(Write Endurance Cycle)이 DRAM 대비 낮음에 따라, 인듀어런스 문제 없이 시스템에서 사용하기 위해서는 프로그램의 메모리 쓰기 특성이 중요한 요인이 된다.
- [0056] 도 8은 통상의 PC에서 메모리 쓰기 특성을 측정한 결과를 나타내는 그래프이다. 도 8에서 보는 바와 같이 메모리 쓰기가 특정 주소에 집중되는 현상을 가진다는 것을 알 수 있다. 여러 번 동일한 실험을 반복한 결과, 전원을 끄지 않고 프로그램만 재 실행 시킬 경우는 집중 액세스되는 주소가 변하지 않았으나, 전원을 껐다가 다시 켜서 재 실행할 때는 집중 쓰기되는 주소가 변하는 것을 알 수 있다.
- [0057] 통상의 컴퓨터(PC)에서 쓰기가 특정 주소에 집중된다는 현상을 기반으로, 본 발명의 일 실시예에서는 상술한 바와 같이 DRAM 중 일부를 PRAM에 저장된 응용 프로그램 데이터의 캐쉬로 할당하여 사용한다. 캐쉬의 페이지 교체(Replace) 정책은 LRU(Least Recently Used) 일 수 있다. 즉, DRAM 중 캐쉬로 할당된 영역은 PRAM에 저장된 응

용 프로그램 데이터의 캐쉬로 사용되며, 이 캐쉬 영역이 다 찬 경우, 캐쉬 영역에서 LRU(Least Recently Used) 데이터는 PRAM으로 이송(migration)될 수 있다.

- [0058] 도 9는 본 발명의 실시예에 따른 데이터(혹은 페이지) 타입에 따른 초기 할당 및 이송 정책을 나타내는 표이다.
- [0059] 응용 프로그램 데이터를 PRAM 영역에 할당하고, 성능 저하 방지와 인듀어런스 문제 해결을 위해 DRAM 캐쉬를 할당하므로, 할당된 DRAM 캐쉬 공간이 부족한 상황이 되면 페이지 이송(Migration)이 필요하게 된다.
- [0060] 상술한 바와 같이 페이지 파일을 생성할 때, 파일 속성이 페이지 테이블 엔트리에 기록되게 되는데, 본 발명의 실시예에서는 페이지 테이블 엔트리(PTE)의 속성 정보를 참조하여, 도 9와 같이 초기 메모리 영역을 할당하고, LRU 에 의하여 페이지 이송(Migration)을 수행한다. 그러나, 본 발명이 이에 한정되는 것은 아니며, 다양한 변형이 가능하다. 예컨대, 스토리지 장치(140)로부터 데이터를 메인 메모리(200)로 처음 적재할 때, 페이지 테이블 엔트리(PTE)의 속성 정보가 아직 생성(혹은 파악)되지 않은 경우에는, 스토리지 장치(140)로부터 추출된 데이터는 미리 정해진 메모리 영역(예컨대, DRAM)에 저장될 수 있다. 추후, 페이지 테이블 엔트리(PTE)의 속성 정보가 생성(혹은 파악)된 후에는, 속성 정보를 이용하여 해당 데이터의 저장 영역이 선택될 수 있다.
- [0061] 위와 같은 기준 하에서, DRAM 캐쉬에 상주하는 페이지의 교체는 아래와 같은 순서로 실행될 수 있다. 먼저, 기존에 진행되던 프로세스의 워킹 셋으로 인해 DRAM 캐쉬 영역이 부족한 상황이 되면 메모리 할당자에서 "아웃 오브 캐쉬"를 발생한다. 두 번째로, DRAM 캐쉬 내부의 희생(Victim) 페이지를 선택한다. 여기서 희생 페이지란, 스왑-아웃(Swap-out) 대상이 되는 페이지로서, LRU에 의해 선정될 수 있다.
- [0062] 세 번째로, 선택된 페이지의 PTE 참조하여 연결된 리스트(Linked List)를 해제한다. 다음으로, 메모리 할당자에 의해 호출된 PRAM 할당자는 PRAM 영역에서 쓰기할 페이지를 선택한다. 다음으로, DRAM 캐쉬 영역에서 선택된 희생 페이지를 선택된 PRAM 영역으로 복사한 후, PRAM 영역으로 복사된 페이지를 페이지 테이블에 다시 설정하여 페이지 매핑을 변경한다.
- [0063] 본 발명의 실시예에 따른 하이브리드 메모리 시스템에서 DRAM 과 PRAM의 용량의 구성비는 달라질 수 있다. 예컨대, DRAM 과 PRAM 반반(50대 50)으로 나누어 구성 할 수도 있고, PRAM을 상대적으로 많이 사용할 수도 있으며, 다양하게 변형 가능하다.
- [0064] 도 10a는 통상의 메인 메모리의 구성의 일 예를 나타내는 도면이고, 도 10b및 도 10c는 각각 본 발명의 실시예에 따른 하이브리드 메모리의 구성의 일 예를 나타내는 도면이다.
- [0065] 도 10b에 도시된 바와 같이 DRAM과 PRAM을 합친 총 용량을 도 10a에 도시된 바와 같이 DRAM으로만 구성한 경우와 같도록 할 경우, 즉 PRAM을 활용하지만 총 용량은 동일한 경우에는 총 메모리 용량 부족으로 스토리지 장치로의 스와핑이 자주 발생할 수 있다.
- [0066] 이러한 스와핑을 줄이기 위해서는 도 10c에 도시된 바와 같이 PRAM용량을 상대적으로 크게 하여, 응용 프로그램 데이터(Application Data)를 저장할 공간을 최대한 확보하여 스토리지 장치로의 스와핑이 최소화 되도록 하고, 동시에 DRAM 캐쉬 용량도 충분히 확보함으로써, 성능 저하 없이 전력 소모 감소와 PRAM의 인듀어런스 문제를 해결 할 수 있다.
- [0067] 도 11은 본 발명의 다른 일 실시예에 따른 하이브리드 메모리 시스템의 구성 블록도이다.
- [0068] 이를 참조하면, 하이브리드 메모리 시스템(500)은 CPU(510), 메인 메모리(520), 및 스토리지 장치(530)를 포함할 수 있다. 메인 메모리(520)는 도 1에 도시된 바와 같이, 휘발성 메모리(521, 예컨대, DRAM)와 비휘발성 메모리(522, 예컨대, PRAM)를 포함하는 하이브리드 메모리일 수 있다.
- [0069] 스토리지 장치(530)는 데이터를 비휘발적으로 저장하기 위한 저장 장소로서, OS(Operating System), 각종 프로그램들, 및 각종 데이터를 저장할 수 있으며, 하드 디스크 드라이브(HDD), 솔리드 스테이트 드라이브(Solid State Drive, 이하 'SSD'라 함) 등일 수 있으나, 이에 한정되지 않는다.
- [0070] 하이브리드 메모리 시스템(500)은 도 11에 도시된 구성 요소 이외에도 사용자와 인터페이스 하기 위한 I/O 모듈, 전원부 등을 더 구비할 수 있다.
- [0071] 본 발명의 실시예에 따른 하이브리드 메모리 시스템은 개인용 컴퓨터 또는 노트북뿐만 아니라, PDA(personal digital assistance), 휴대 전화기(cellular telephone), MP3 플레이어, PMP(portable multimedia player), 차량자동항법장치(automotive navigation system), MID(Mobile Internet Device) 등일 수 있으나, 이에 한정되는 것은 아니다.

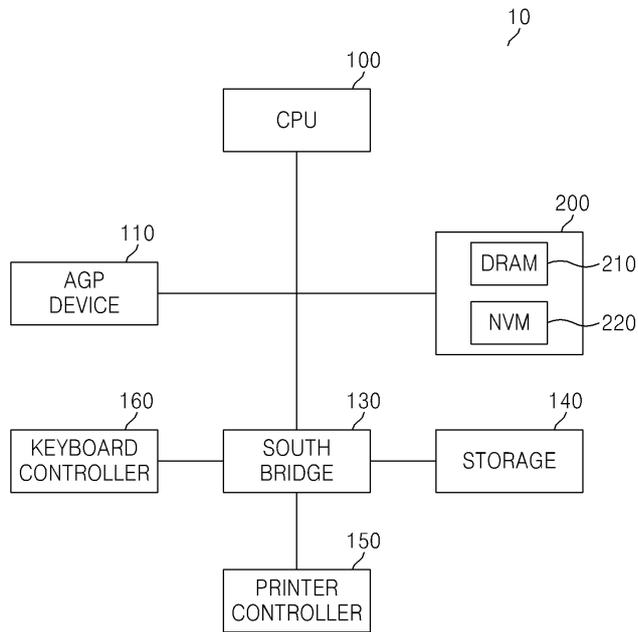
- [0072] 본 발명은 또한 컴퓨터로 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록장치를 포함한다.
- [0073] 컴퓨터가 읽을 수 있는 기록매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피 디스크, 광 데이터 저장장치 등이 있다.
- [0074] 또한 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다. 그리고 본 발명을 구현하기 위한 기능적인(functional) 프로그램, 코드 및 코드 세그먼트들은 본 발명이 속하는 기술분야의 프로그래머들에 의해 용이하게 추론될 수 있다.
- [0075] 상기 본 발명의 내용은 도면에 도시된 실시 예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시 예가 가능하다는 점을 이해할 것이다. 따라서 본 발명의 진정한 기술적 보호 범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

### 부호의 설명

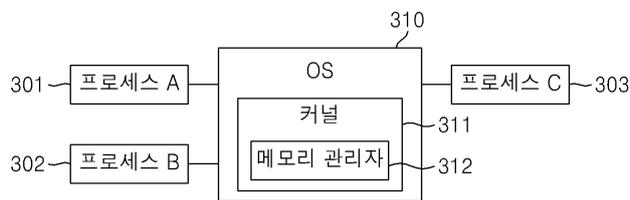
- [0076] 컴퓨팅 시스템(10),  
 중앙 처리 장치(Central Processing Unit, 100, 510),  
 AGP 장치(Accelerated Graphics Port, 110),  
 메인 메모리(200, 520),  
 휘발성 메모리(210, 521),  
 비휘발성 메모리(220, 522),  
 스토리지 장치(140, 530),  
 사우스 브리지(130),  
 키보드 컨트롤러(160),  
 프린터 컨트롤러(150),  
 하이브리드 메모리 시스템(500)

도면

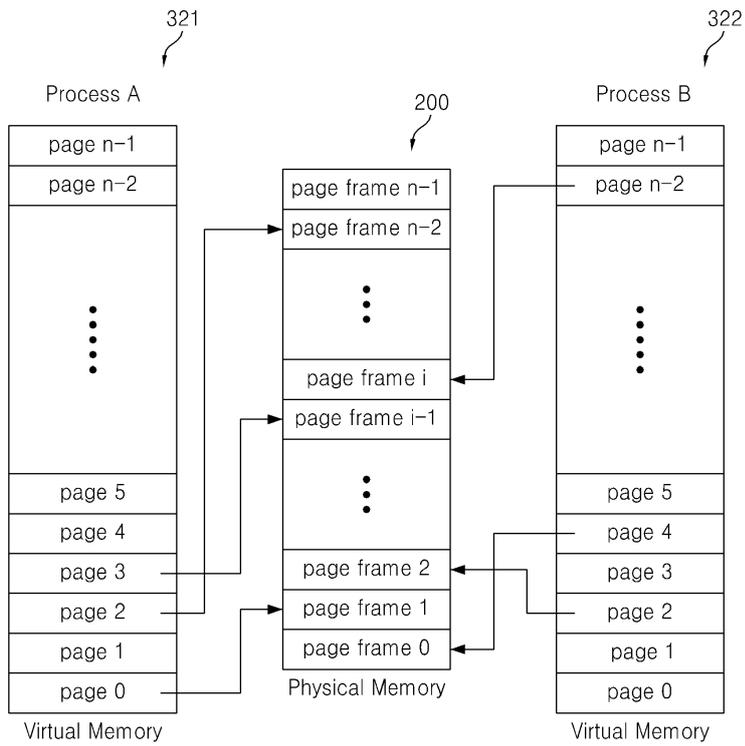
도면1



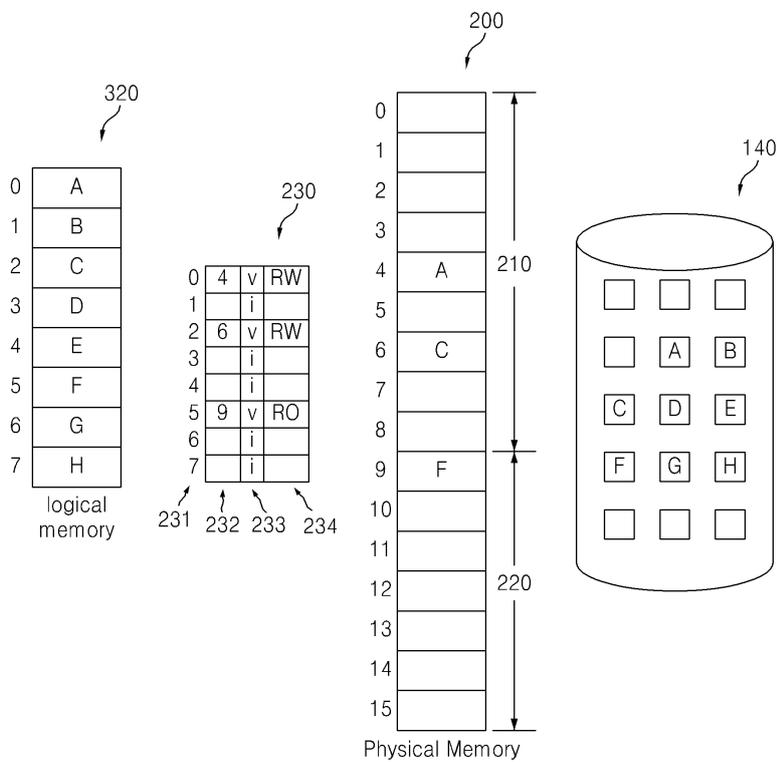
도면2



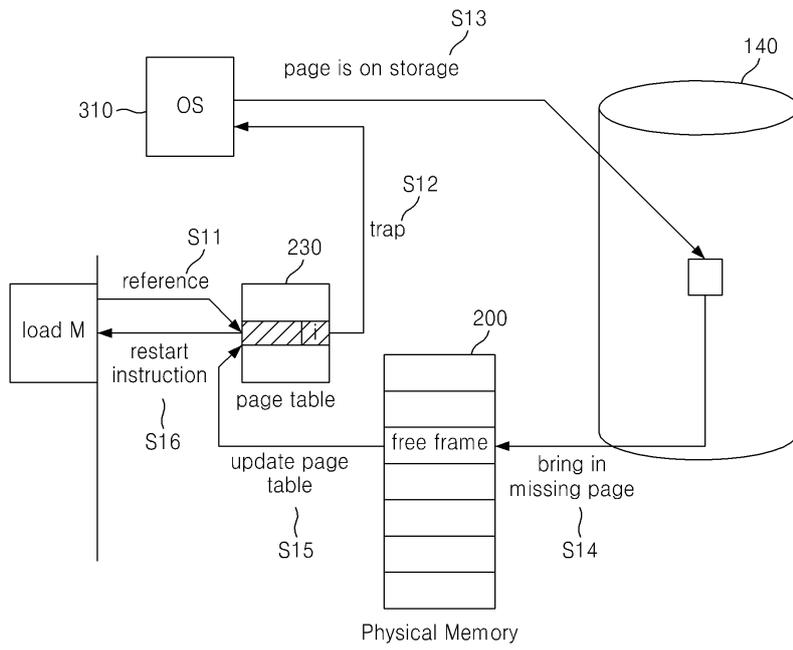
도면3



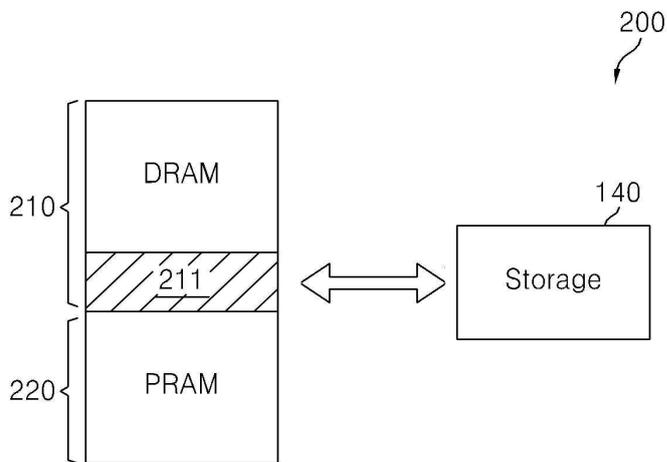
도면4



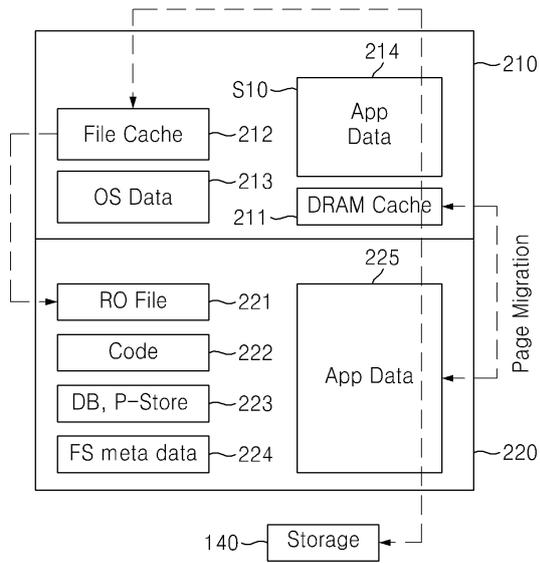
도면5



도면6a



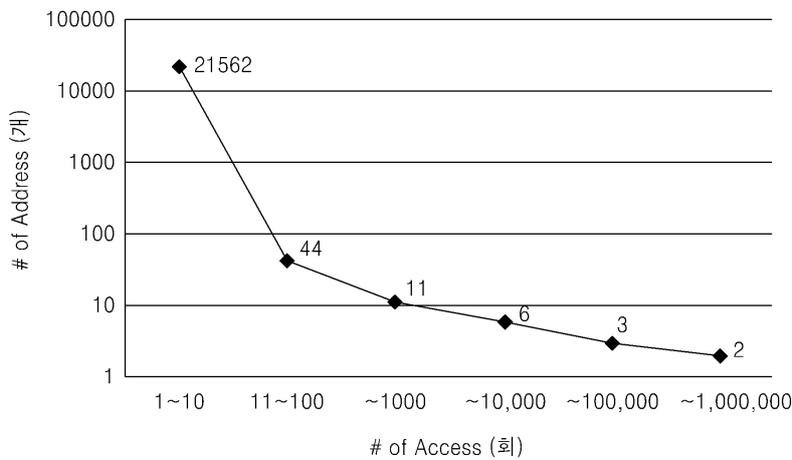
도면6b



도면7

Page Type	특징	Page 할당 영역
Code	Read Only	PRAM
Stack	Read/Write 빈번히 발생	DRAM
Heap	Read/Write 발생	DRAM 또는 PRAM
Buffer for Library File	Read Only	PRAM
Buffer for Temporary File	Read/Write 발생	DRAM 또는 PRAM

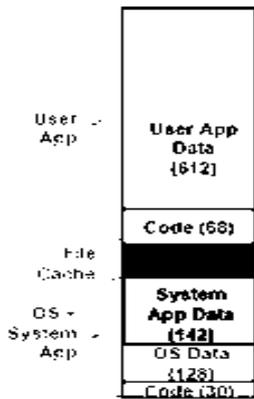
도면8



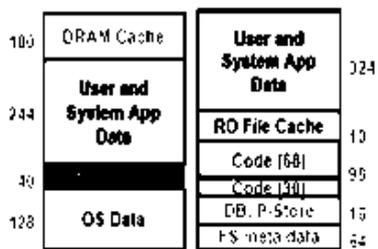
도면9

Contents		Initial Allocation	Migration Policy
Kernel	Code	PRAM	No migration
	Data	DRAM	No migration
User	Code	PRAM	No migration
	Stack	DRAM	No migration
	RO data	PRAM	No migration
	RW data	DRAM	Migrated to PRAM when required
	Heap	DRAM	Migrated to PRAM when required

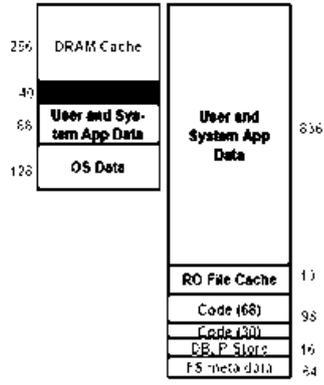
도면10a



도면10b



도면10c



도면11

