

(19)대한민국특허청(KR)  
(12) 등록특허공보(B1)

(51) 。 Int. Cl. G06F 12/02 (2006.01)	(45) 공고일자 (11) 등록번호 (24) 등록일자	2006년08월10일 10-0611305 2006년08월03일
---	-------------------------------------	--

(21) 출원번호 (22) 출원일자	10-2004-0097433 2004년11월25일	(65) 공개번호 (43) 공개일자	10-2006-0058401 2006년05월30일
------------------------	--------------------------------	------------------------	--------------------------------

(73) 특허권자            삼성전자주식회사  
                              경기도 수원시 영통구 매탄동 416

(72) 발명자                이상형  
                              서울 강남구 논현2동 동현아파트 2동 705호

(74) 대리인                정홍식

심사관 : 김기천

(54) 주소 맵핑 방법 및 장치

요약

주소 맵핑 방법 및 장치가 개시된다. 본 주소 맵핑 방법은, 먼저 시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역을 각각 설정한다. 그리고, 제1 주소 영역 중 소정 영역에 대응하는 제1 물리 주소 영역과, 제2 주소 영역 중 소정 영역에 대응하는 제2 물리 주소의 영역이, 동일한 주소 영역을 액세스 하도록 맵핑한다. 이에 의해, 프로그래밍 과정에서 물리 주소의 재할당 등에 대한 고려가 필요없게 되고, 주소 맵핑을 위한 디코딩 과정이 단순하게 된다.

대표도

도 2

색인어

MIPS, 물리 주소, 가상 주소, 주소 맵핑

명세서

도면의 간단한 설명

도 1은 일반적인 MIPS 기반의 시스템에서 사용되는 어드레스 맵의 일 예,

도 2는 본 발명의 일실시예에 따른 주소 맵핑 방법이 적용되는 시스템 환경을 설명하기 위한 도면, 그리고

도 3 및 도 4는 본 발명의 일실시예에 따른 주소 맵핑 방법의 설명에 참조되는 어드레스 맵이다.

\* 도면의 주요 부분에 대한 부호의 설명 \*

10 : 가상 주소 20 : TLB

30 : 물리 주소 150 : 디코더 로직부

**발명의 상세한 설명**

**발명의 목적**

**발명이 속하는 기술 및 그 분야의 종래기술**

본 발명은 주소 맵핑 방법 및 장치에 관한 것으로, 더욱 상세하게는 MIPS 프로세서(processor) 기반의 시스템에서 프로세서에 의해 액세스되는 주소를 맵핑하는 주소 맵핑 방법 및 장치에 관한 것이다.

고성능 컬러 레이저 프린터에 적용될 컨트롤러 등의 설계에 있어서, 성능(performance) 요구 조건을 만족시키기 위해, MIPS 프로세서(processor)를 사용하는 경우가 증가하고 있다. 그런데, MIPS 프로세서의 경우, 부트(boot)시에는 TLB (Translation Look-aside Buffer) 설정 없이 액세스 가능한 0.5GB 정도 크기의 주소 영역을 사용하고, 프로그램이 수행되면서 TLB 설정을 통해 나머지 주소 영역을 확장하여 사용하게 된다.

도 1은 일반적인 MIPS 기반의 시스템에서 사용되는 어드레스 맵의 일 예를 도시한 것이다. 도 1을 참조하면, DDR Controller, GDMA Controller, IO Controller, PCI Controller, Special Function Registers, ROM Controller 등에 할당된 각 영역에 해당하는 주소를 디코딩하는데 있어서, 기준 주소 레지스터(base address register) 및 크기 레지스터(size register)에 저장된 정보를 참조하여 디코딩 동작을 수행하게 된다. 즉, 디코딩을 수행해야 할 주소가 디코딩 로직부에 입력되면, 레지스터에 저장된 (base address ~ base address + size) 내에 해당하는 주소인지 판단하여, 해당 영역에 액세스 가능하도록 하는 CS(chip select) 신호를 발생하게 된다. 그리고, 프로그램 수행중에 사용자가 해당 기준 주소 레지스터 및 크기 레지스터의 값을 변경시켜 해당 영역을 재할당 시키게 된다.

예를 들어 MIPS 프로세서를 사용하는 DDR Controller 에서 2GB 범위의 DDR(Double Data rate)램 영역을 지원한다고 하면, 2GB는 이미 MIPS 프로세서가 부트(boot)시 액세스할 수 있는 0.5GB 정도의 범위를 초과한다. 따라서, 부트시 이 중 일부 범위만을 맵핑시켜 놓고, 부트 이후 프로그램을 통해서 이 영역을 재할당(relocation)시켜 2GB 범위의 영역을 사용하게 된다.

도 1의 경우에는, 부트시 DDR 영역을 [0x0000\_0000 ~ 0x17FF\_FFFF]의 384MB 크기의 영역을 맵핑시켜 놓는다. 프로그램이 수행되면서 재할당을 통해 DDR영역을 [0x4000\_0000 ~ 0xBFFF\_FFFF] 2GB 크기의 주소 영역으로 맵핑하게 된다. 각 영역별로 기준 주소 레지스터와 크기 레지스터들을 내장한 디코더 로직부는 해당 영역을 재할당시킬 수 있기 때문에, 사용자가 원하는 물리 주소를 생성할 수 있다.

그러나, 이와 같은 방식의 주소 맵핑 방법은 다음과 같은 문제점이 있다.

첫째, 물리 주소가 가변적이기 때문에 이에 대한 대처가 필수적이다. 즉, 물리 주소가 변동되면 가상 주소를 물리 주소로 변환하는데 쓰이는 TLB 내에서 해당 주소에 맵핑되어 있는 모든 엔트리(entry)들도 같이 수정되어야 한다. 또한 해당 주소에 참조된 포인터(pointer)가 있었다면 이 참조값들도 모두 변경시켜주어야 한다. 따라서, 프로그래밍시 많은 고려가 필요하게 된다.

둘째, 프로그래머가 임의로 주소 영역을 매핑할 수 있기 때문에 오류가 발생할 가능성이 있다. 예컨대, 여러 명의 개발자가 동시에 프로젝트를 수행할 때 각각의 영역에 대한 기준 주소(base address)를 각자 별도로 설정한다고 가정하면, 이 때 서로 설정한 영역의 경계가 다른 영역을 침범할 수도 있다. 이 경우 맵핑된 영역에 중복 구간이 발생하게 되어, 예상 못한 오동작을 유발할 수 있다.

셋째, 각 영역에 해당하는 주소를 디코딩하기 위한 디코더 로직(decoder logic)부의 구성이 복잡해져 성능을 저하시킬 수 있다. 즉, 각 영역의 기준 주소(base address)가 변동되기 때문에 모든 영역별로 기준 주소 레지스터(base address

register)와 크기 레지스터(size register)를 비교하는 조합로직(combination logic)이 필요하게 된다. 이것은 디코딩 과정의 성능 저하 요인이 될 수 있다. 더욱이 지원하는 영역의 개수에 비례하여 디코더 로직부내의 조합로직이 추가되어야 하므로, 전체 디코더 로직부의 복잡성이 증가하게 된다.

**발명이 이루고자 하는 기술적 과제**

따라서, 본 발명의 목적은, 물리 주소의 재할당을 용이하게 하여, 디코더 로직부의 구성을 단순화하고 성능 향상을 꾀할 수 있도록 하는 주소 맵핑 방법 및 장치를 제공함에 있다.

**발명의 구성 및 작용**

상기 목적을 달성하기 위한 본 발명에 따른 주소 맵핑 방법은, 시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역을 각각 설정하는 단계, 및 상기 제1 주소 영역 중 소정 영역에 대응하는 제1 물리 주소 영역과, 상기 제2 주소 영역 중 소정 영역에 대응하는 제2 물리 주소 영역이, 동일한 주소 영역을 액세스 하도록 맵핑하는 단계를 포함한다.

상기 프로세서는, MIPS 기반의 프로세서인 것이 바람직하다. 또한, 상기 동일한 주소 영역은, 동일한 주기억장치의 영역인 것이 가능하다.

한편, 본 발명에 따른 주소 맵핑 장치는, 시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역의 설정이 가능한 시스템에서, 상기 제2 주소 영역의 가상 주소를 물리 주소로 주소 변환하는 주소변환부, 및 상기 제1 주소 영역 중 소정 영역 속하는 제1 물리 주소와, 상기 물리 주소 중 소정 영역에 속하는 제2 물리 주소가, 동일한 주소를 액세스 하도록 디코딩하는 디코더 로직부를 포함한다.

상기 프로세서는 MIPS 기반의 프로세서인 것이 바람직하며, 상기 동일한 주소 영역은 동일한 주기억장치의 영역인 것이 가능하다. 그리고, 상기 시스템은, 컬러 레이저 프린터 시스템인 것이 바람직하다.

본 발명에 따른 컴퓨터 판독 가능한 매체는, 주소 맵핑을 위한 동작을 컴퓨터에 실행시키기 위한 명령어를 제공하며, 시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역을 각각 설정하는 단계; 및 상기 제1 주소 영역 중 소정 영역에 대응하는 제1 물리 주소 영역과, 상기 제2 주소 영역 중 소정 영역에 대응하는 제2 물리 주소 영역이, 동일한 주소 영역을 액세스 하도록 맵핑하는 단계를 포함한다.

이하에서는 도면을 참조하여 본 발명을 보다 상세하게 설명한다.

본 발명에 따른 메모리 관리방법은 기본적으로 MIPS 프로세서를 기반으로 하는 시스템에 적용되지만, 반드시 이에 국한되는 것은 아니며 본 발명에 따른 조건을 만족하는 다른 프로세서를 사용하는 시스템에 적용이 가능하다.

도 2는 본 발명의 일실시에 따른 메모리 관리방법이 적용되는 시스템 환경을 설명하기 위한 도면이다. 도 2를 참조하면, 프로그램에서 참조되는 가상 주소  $A_V$ 는 기준 주소  $B_V$ 와 변위  $D$ 가 결합하여 만들어진 주소이다. 가상 주소  $A_V$ 에는 프로그램에서 정의된 주소지정 모드에 따라 계산되는 실질 주소가 포함되며, 시스템에 의해 정해제는 세그먼트 주소 등의, 제어 정보도 포함되어 있다. 기준 주소  $B_V$ 의 실제 주소  $B_R = f(B_V)$ 는 페이지 테이블에 저장되어 있으며, 주소 변환을 빨리 처리하기 위해서, 페이지 테이블의 일부 또는 전부가 프로세서 내의 고속 메모리인 TLB(20)에 저장된다.

TLB(20)의 입력값은 가상 주소  $A_V$ 에서 기준 주소 부분인  $B_V$ 이며, TLB(20)의 출력은 실제 기준 주소인  $B_R$ 이다. 이 주소가  $A_V$ 의  $D$  부분과 합쳐져서 물리 주소  $A_R$ 이 만들어진다. 기준 주소  $B_V$ 가 TLB(20)에 있지 않다면, 페이지 테이블 중  $B_V$ 를 포함하는 부분이 우선적으로 TLB(20)에 전송된다. 따라서, TLB(20)는 페이지 테이블을 위한 캐시 역할을 하며, 이런 이유로 TLB(20)를 주소 캐시(address cache)라고도 한다.

디코더 로직부(50)는 물리 주소  $A_R$ 을 입력받아, 이에 대응하는 각 주소 영역에 액세스 가능하도록 하는 CS(Chip Select) 신호, 즉  $CS_0 \sim CS_n$ 를 생성한다. 디코더 로직부(50)에서, CS 신호는 물리 주소  $A_R$ 의 상위 주소에서 해당 주소 비트를 디코딩하여 생성한다. 물리 주소  $A_R$ 의 하위 주소는 CS 신호가 생성된 해당 주소 영역에서 변위로 사용된다.

이와 같은, 시스템 환경에서, 본 발명에 따른 주소 맵핑 방법은, 디코더 로직부(50)에서 특정 주소 영역에 대응하는 물리 주소를 이중으로 맵핑하도록 구성 된다. 여기서, 이중으로 맵핑한다는 것은, 서로 다른 물리 주소가 동일한 주소 영역을 액세스하도록 맵핑된다는 것이며, 이에 대해서는 어드레스 맵을 예로서 보다 상세하게 설명하기로 한다.

도 3 및 도 4는 본 발명의 일 실시예에 따른 주소 맵핑 방법의 설명에 참조되는 어드레스 맵이다. 먼저 도 3에서, "Boot Physical Address"로 표기한 것이 부트시 프로세서가 액세스할 수 있는 제1 주소 영역을 나타내고, "Extened Physical Address"로 표기한 것은 TLB 설정 등을 통해 가상 주소에 의해 확장된 제2 주소 영역을 나타낸다. 그리고, "Total Physical address"는 제1 주소 영역과 제2 주소 영역을 모두 나타낸 것이다.

이와 같은 구성의 어드레스 맵 상태에서, DDR, GDMA 등의 영역에 대응하는 물리 주소를 맵핑함에 있어서 이중으로 맵핑한다. 즉, MIPS 프로세서가 부트시 액세스 가능한 대략 0.5GB 크기의 제1 주소 영역과, 부트 완료 후 TLB 설정을 통하여 가상 주소(virtual address)에 맵핑되는 대략 3.5GB(32bit address일 경우) 크기의 제2 주소 영역에서, 일부 영역을 이중으로 맵핑한다.

예를 들어 MIPS 프로세서를 사용하는 시스템에서, 2GB 범위의 DDR램 영역을 지원하는 경우, 이 중 일부 범위는 MIPS 프로세서가 부트시 접근할 수 있는 제1 주소 영역과, 확장된 영역인 제2 주소 영역에 이중으로 맵핑시켜 놓고, 나머지 범위는 확장된 영역인 제2 주소 영역에만 맵핑시켜 놓는다. 즉, DDR램 영역을 [0x0000\_0000 ~ 0x17FF\_FFFF] 384MB의 영역과 [0x4000\_0000 ~ 0xBFFF\_FFFF] 2GB영역으로 맵핑하게 된다. 이 중 [0x4000\_0000 ~ 0x57FF\_FFFF] 384MB 영역은 [0x0000\_0000 ~ 0x17FF\_FFFF] 384MB 영역과 동일한 주소 영역을 액세스하도록 디코더 로직부(50)는 디코딩한다. 따라서, [0x4000\_0000 ~ 0x57FF\_FFFF] 영역과 [0x0000\_0000 ~ 0x17FF\_FFFF] 영역은 동일한 DDR램 영역을 액세스하게 된다.

다음으로, 도 4는 가상 주소와 이에 대응하는 물리 주소의 어드레스 맵의 일 예를 도시한 것이다. 도 4를 참조하면, "Virtual Address"로 표시된 가상 주소의 어드레스 맵에서, kuseg는 2 GB 크기의 세그먼트이며, 사용자의 코드와 데이터가 저장된다. 이 영역의 주소들은 캐시를 최대한 활용하며, TLB를 통해 물리 주소로 변환된다. 이 세그먼트 영역은 [0x4000\_0000 ~ 0xBFFF\_FFFF]의 물리 주소 영역으로 맵핑된다.

kseg0는 0.5 GB 크기의 시스템 세그먼트이며, 캐시를 사용하나 주소 변환은 필요하지 않은 영역이다. 즉, 이 세그먼트의 가상 주소는 물리 주소 공간의 앞부분 0.5 GB에 해당한다. 따라서, 이 세그먼트의 영역은 [0x0000\_0000 ~ 0x1FFF\_FFFF]의 물리 주소 영역에 맵핑된다. 이 세그먼트에는 일반적으로 운영체제 중 많이 쓰이는 부분이 저장된다.

kseg1도 0.5 GB 크기의 세그먼트이며, 캐시를 사용하지 않고, 주소변환도 필요치 않은 영역이다. 이 세그먼트에는 캐시에 넣을 수 없는 부팅 코드를 저장하거나, 고속 I/O 데이터 같이 캐시의 속도를 심각하게 떨어뜨릴 데이터를 저장하는데 쓰인다. 이 세그먼트 영역도 [0x0000\_0000 ~ 0x1FFF\_FFFF]의 물리 주소 영역에 맵핑된다.

kseg2는 0.5 GB 크기의 세그먼트로서, ksuseg 처럼 캐시를 사용하며, 주소 변환이 필요하다. 이 세그먼트 영역은 [0xE000\_0000 ~ 0xFFFF\_FFFF]의 물리 주소 영역으로 맵핑된다.

이와 같이 프로세서가 부트시 액세스할 수 있는 제1 주소 영역과 나머지 TLB 설정을 통하여 가상 주소에 매핑되는 제2 주소 영역에서 일부 영역을 이중으로 맵핑하여, 가상주소와 이에 대응하는 물리 주소가 고정되면, 물리 주소에 대한 재할당에 대한 고려 없이, 각 프로그램에서는 가상 주소의 어드레스 맵만 고려하면 된다. 또한, 물리 주소를 고정함으로써, 디코더 로직부(50)는 물리 주소에서 설정된 주소에 대응하는 주소 비트만을 디코딩하여 CS 신호를 발생시켜, 해당 주소에 대응하는 영역에 액세스 가능하도록 할 수 있다. 따라서, 기존 주소 레지스터와 크기 레지스터를 비교하는 조합로직 등이 불필요하게 되므로, 디코더 로직부의 구성이 간단하게 된다.

한편, 상기한 실시예에서는, 디코더 로직부에서 디코딩 과정을 수행하는 것을 설명하였으나, 본 발명에 따른 주소 맵핑 방법은 이와 같이 하드웨어 장치로 구현할 수도 있으나, 프로그램화 되어 처리되는 것도 가능하다.

### 발명의 효과

이상 설명한 바와 같이, 본 발명에 따르면, 주소 확장이 필요한 영역에 대해 MIPS 프로세서가 부트시 액세스할 수 있는 제1 주소 영역과 나머지 TLB 설정을 통하여 가상 주소에 매핑되는 제2 주소 영역에 대하여 이중적으로 물리 주소를 매핑시킬 수 있다. 이에 따라 다음과 같은 효과를 기대할 수 있다.

첫째, 물리 주소가 고정되어 있기 때문에, 물리 주소의 재할당에 대한 고려없이 각 프로그램내에서 가상 주소의 맵만 고려하면 되므로, 프로그래밍 과정이 단순하게 된다.

둘째, 고정된 물리 주소를 프로그래머가 임의로 주소 영역으로 매핑할 수 없기 때문에, 잘못된 주소 영역의 맵핑에 따른 오동작할 가능성을 배제할 수 있다.

셋째, 물리 주소가 고정되어 있기 때문에 해당 주소 비트만으로 각 영역에 대한 디코딩이 가능해지므로, 디코더 로직부의 구성이 단순해져 성능이 향상을 꾀할 수 있게 된다. 이에 따라, 주소 영역을 아무리 세분화하더라도 이를 위한 레지스터들이 추가적으로 필요 없게 된다.

또한, 이상에서는 본 발명의 바람직한 실시예에 대하여 도시하고 설명하였지만, 본 발명은 상술한 특정의 실시예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진자에 의해 다양한 변형실시가 가능한 것은 물론이고, 이러한 변형실시들은 본 발명의 기술적 사상이나 전망으로부터 개별적으로 이해되어져서는 안될 것이다.

### (57) 청구의 범위

#### 청구항 1.

시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역을 각각 설정하는 단계; 및

상기 제1 주소 영역 중 소정 영역에 대응하는 제1 물리 주소 영역과, 상기 제2 주소 영역 중 소정 영역에 대응하는 제2 물리 주소 영역이, 동일한 주소 영역을 액세스 하도록 맵핑하는 단계;를 포함하는 것을 특징으로 하는 주소 맵핑 방법.

#### 청구항 2.

제1항에 있어서, 상기 프로세서는, MIPS 기반의 프로세서인 것을 특징으로 하는 주소 맵핑 방법.

#### 청구항 3.

제2항에 있어서, 상기 동일한 주소 영역은, 동일한 주기억장치의 영역인 것을 특징으로 하는 주소 맵핑 방법.

#### 청구항 4.

시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역의 설정이 가능한 시스템에서의 주소 맵핑 장치에 있어서,

상기 제2 주소 영역의 가상 주소를 물리 주소로 주소 변환하는 주소변환부; 및

상기 제1 주소 영역 중 소정 영역에 속하는 제1 물리 주소와, 상기 물리 주소 중 소정 영역에 속하는 제2 물리 주소가, 동일한 주소 영역을 액세스 하도록 디코딩하는 디코더 로직부;를 포함하는 것을 특징으로 하는 주소 맵핑 장치.

#### 청구항 5.

제4항에 있어서, 상기 프로세서는, MIPS 기반의 프로세서인 것을 특징으로 하는 주소 맵핑 장치.

**청구항 6.**

제4항에 있어서, 상기 동일한 주소 영역은, 동일한 주기억장치의 영역인 것을 특징으로 하는 주소 맵핑 장치.

**청구항 7.**

제4항에 있어서, 상기 시스템은, 컬러 레이저 프린터 시스템인 것을 특징으로 하는 주소 맵핑 장치.

**청구항 8.**

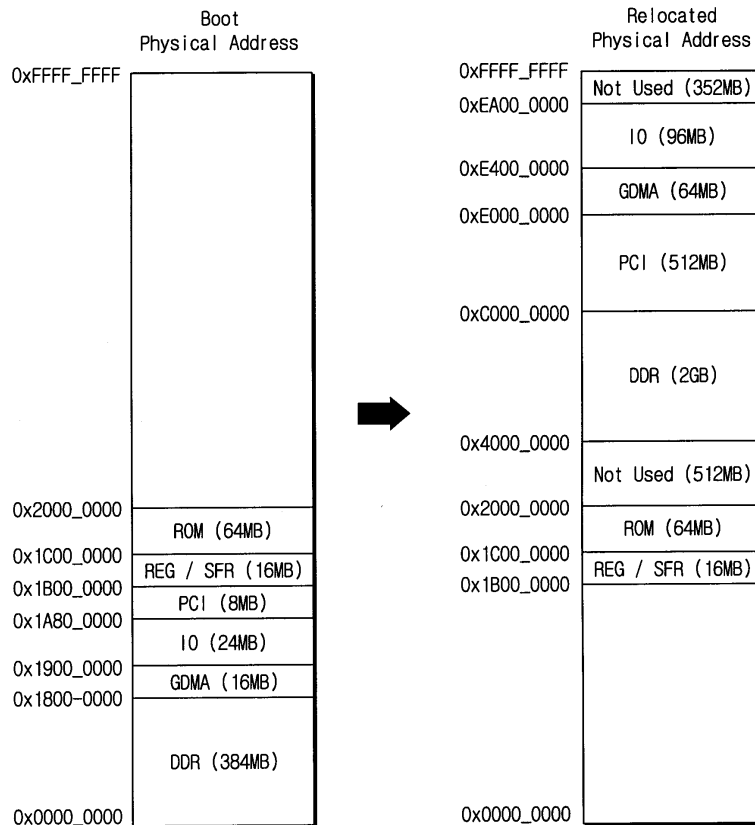
주소 맵핑을 위한 동작을 컴퓨터에 실행시키기 위한 명령어를 제공하며,

시스템 부팅시 프로세서가 액세스 가능한 제1 주소 영역과, 가상 주소에 의해 확장된 제2 주소 영역을 각각 설정하는 단계; 및

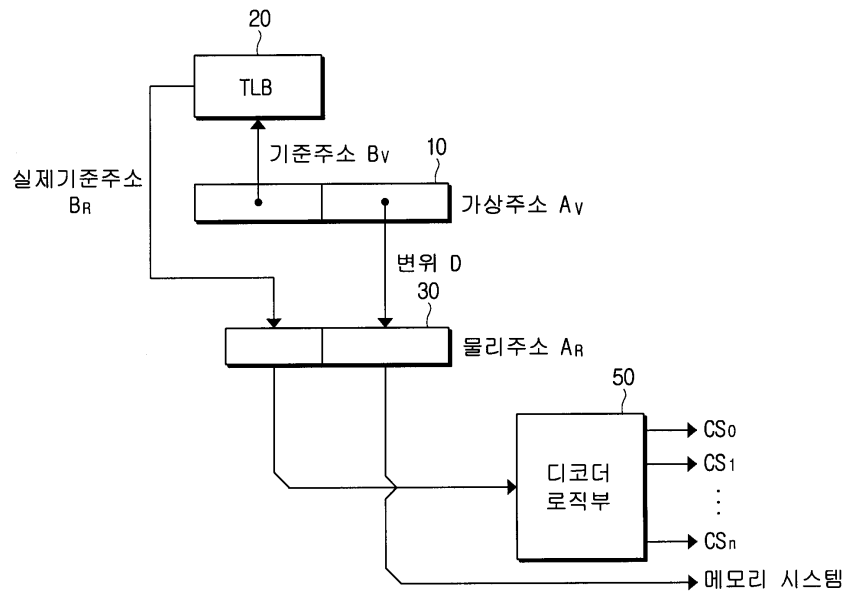
상기 제1 주소 영역 중 소정 영역에 대응하는 제1 물리 주소 영역과, 상기 제2 주소 영역 중 소정 영역에 대응하는 제2 물리 주소 영역이, 동일한 주소 영역을 액세스 하도록 맵핑하는 단계;를 포함하는 것을 특징으로 하는 컴퓨터 판독 가능한 매체.

**도면**

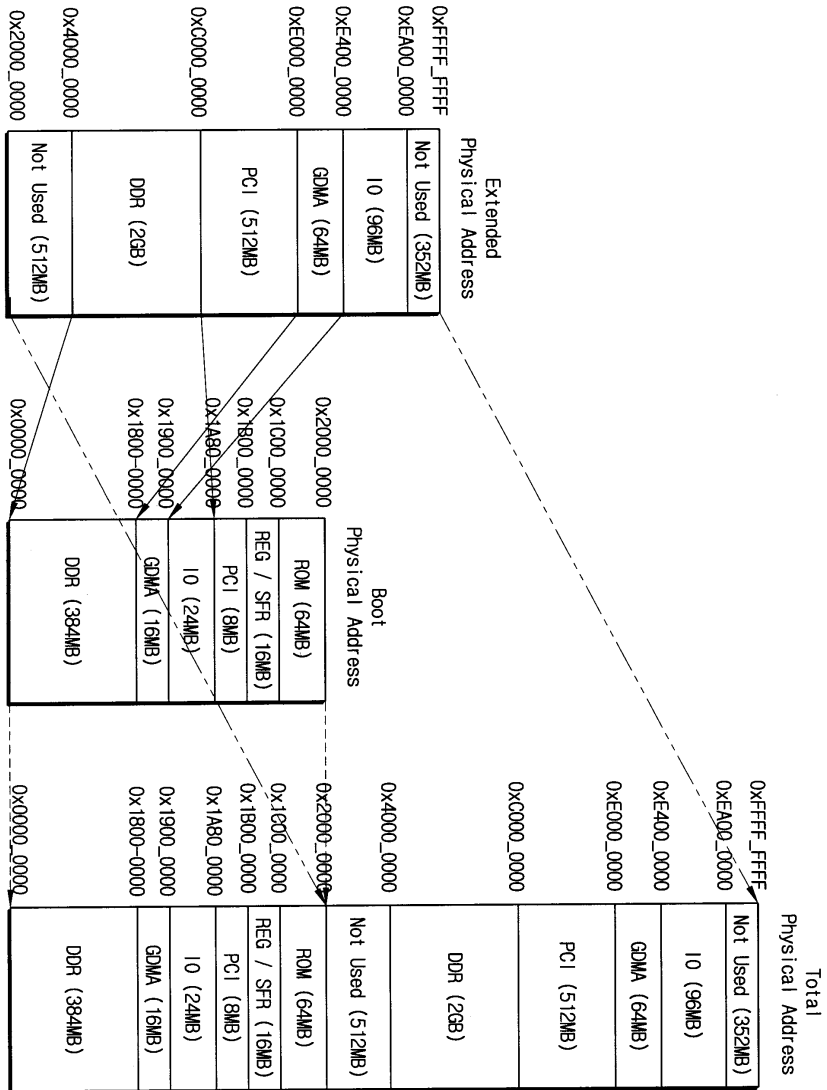
**도면1**



도면2



도면3





도면4

