

(19)대한민국특허청(KR)  
(12) 공개특허공보(A)

(51) Int. Cl.<sup>8</sup>  
*G11C 7/10* (2006.01)  
*G11C 11/408* (2006.01)

(11) 공개번호 10-2006-0010849  
(43) 공개일자 2006년02월02일

(21) 출원번호 10-2005-7025031(분할)  
(22) 출원일자 2005년12월27일  
(62) 원출원 특허10-2004-7000533  
원출원일자 : 2004년01월13일 심사청구일자 2004년04월28일  
번역문 제출일자 2005년12월27일  
(86) 국제출원번호 PCT/US2002/022458 (87) 국제공개번호 WO 2003/007303  
국제출원일자 2002년07월10일 국제공개일자 2003년01월23일

(30) 우선권주장 09/905,004 2001년07월13일 미국(US)

(71) 출원인 미크론 테크놀로지,인코포레이티드  
미국, 아이다호 83716, 보이스, 사우스 페드럴웨이 8000

(72) 발명자 잔젠, 제프리 더블유.  
미국, 아이다호 83642, 메리디안, 노스 실버리프 웨이 2727

(74) 대리인 강명구

심사청구 : 있음

(54) 메모리 소자에서의 워드 순서지정 방법

요약

본 발명은 두 종류의 액세스(읽기 및 쓰기)를 구현하기 위한 어드레싱 기법 및 관련 하드웨어에 관한 것이다. 발명에 따른 메모리 소자는 다수의 메모리 셀 어레이를 포함한다. 다수의 메모리 셀로부터 정보를 읽어들이고 다수의 메모리 셀에 정보를 기록하기 위해 아래와 같은 주변 장치들이 제공된다.

- 어드레스 비트에 따라, 다수의 어레이로부터 수신한 비트를 순서지정하기 위한 리오더 회로(reorder circuit), 그리고
- 읽기 동작 중 상기 리오더 회로에 상기 어드레스 비트 일부를 전달하기 위한 어드레스 시퀀서(address sequencer).

본 발명의 방법은 메모리 소자로부터 한개 이상의 n-비트 워드를 출력하기 전에 어떤 어드레스 비트의 정보에 따라 메모리 어레이로부터 출력되는 n-비트 워드들의 블록을 순서지정(reordering)하는 과정을 포함한다.

대표도

도 3

명세서

## 도면의 간단한 설명

도 1은 컴퓨터 시스템 구조의 기능적 블록도표.

도 2는 메모리 회로의 뱅크에 대한 블록도표.

도 3은 본 발명의 버스트 읽기 순서지정을 구현하기 위한 구조의 단순화된 블록도표.

도 4A, 4B, 4C는 유효 워드에 대한 랩 시작 위치(wrap start location)를 식별하기 위해 각각 512 메가비트 x4 파트, x8 파트, x16 파트의 어드레싱을 도시하는 도면.

도 5는 본 발명이 사용될 수 있는 컴퓨터 시스템의 단순화된 블록도표.

## 발명의 상세한 설명

### 발명의 목적

#### 발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 메모리 소자에 관한 것으로서, 특히, 메모리 소자로부터 정보를 읽어들이고 메모리 소자에 정보를 기록하기 위한 방법 및 회로에 관한 것이다.

컴퓨터 설계자들은 고속 컴퓨터 설계를 가능하게 하는 고속 메모리 소자를 찾기 위한 노력을 경주하고 있다. 컴퓨터 동작 속도에 관한 제약 요소 중 한가지는 프로세서와 메모리 회로간 데이터 전송(가령, 읽기 또는 쓰기 데이터 전송)에 소요되는 시간이다. DRAM, SDRAM, 플래시 메모리, 등과 같은 메모리 소자들은 행과 열로 각각 구성된 한개 이상의 어레이로 배열되는 다수의 메모리 셀을 포함하는 것이 일반적이다. 각각의 메모리 셀은 프로세서가 메모리 비트나 엠비트(mbit)라 불리는, 1 비트 데이터를 저장 및 불러올 수 있는 위치를 제공한다. 프로세서가 메모리 셀 내의 데이터에 더 빨리 액세스할 수 있다면, 이 데이터를 이용한 연산이나 프로그램을 더 빨리 실행시킬 수 있다.

도 1은 전형적인 컴퓨터 시스템 구조를 도시한다. CPU나 프로세서(10)가 프로세서 버스(12)에 연결되고, 다시 시스템/메모리 컨트롤러(14)에 연결된다. 메모리 컨트롤러(14)는 확장 버스(16)에 연결될 수 있다. 메모리 컨트롤러(14)는 프로세서(10)와 메모리 소자(18)간 인터페이스 회로로 기능한다. 프로세서(10)는 메모리 컨트롤러(14)에 의해 수신되어 변환되는 명령과 어드레스를 제공한다. 메모리 컨트롤러(14)는 다수의 명령 라인(20) 상에서 변환된 명령 신호를, 다수의 어드레스 라인(22) 상에서 변환된 어드레스를 메모리 소자(18)에 공급한다. 이 명령 신호들은 당 분야에 잘 알려진 것으로서, DRAM의 경우에, RAS(row address strobe), CAS(column address strobe), WE(write enable), 그리고 OE(output enable)를 포함한다. 클럭 신호가 CLK 라인(24) 상에도 제공된다. 프로세서-제공 명령 및 어드레스에 해당하는 데이터가 컨트롤러(14)와 메모리(18) 사이에서 데이터경로 라인(26)을 통해 전달된다.

메모리(18)는 다수의 메모리 순위(memory ranks)(27)를 통상적으로 포함하며, 그 중 하나가 도 2에 도시된다. 본 예에서, 메모리 순위(27)는 8개의 8비트 메모리 회로(28(0)-28(7))를 가진 64비트 시스템에 대해 구성된다. 명령 신호 RAS, CAS, WE가 순위(27) 내 모든 메모리 회로(28(0)-28(7))에 공급된다. 추가 순위를 가진 메모리(18)(도 1)에서, 별도의 CS 명령 신호가 각각의 순위에 대해 제공될 것이다. 따라서, 명령 신호 CS는 순위-전용 명령 신호라 불린다. 어드레스 버스(22)가 순위(27) 내 모든 메모리 회로(28(0)-28(7))에 연결되고, 메모리(18)의 모든 다른 순위(도시되지 않음) 내 모든 다른 메모리 회로(도시되지 않음)에 연결된다. 따라서, 어드레스 버스(22)가 "전역-연결(globally connected)"된다고 일컬어진다.

동기식 DRAM(SDRAM)은 내부 동작을 이용하여, 고속에서 어떤 범위의 어드레스들에 순차적으로 액세스할 수 있는 메모리 소자이다. 전형적인 SDRAM에서, 100MB/s 또는 그 이상의 읽기/쓰기 속도가 가능하다. 이러한 속도 구현을 위해, SDRAM의 읽기/쓰기가 버스트 모드(burst mode)로 실행된다. 버스트 모드는 동일한 행 어드레스를 지닌 데이터가 2, 4, 또는 8 비트 워드의 블록으로 연속적으로 읽혀지거나 쓰여지는 어드레스 액세스 모드이다. 추가적으로, 블록 단위의 이러한 워드에 대한 액세스는 블록의 시작 어드레스를 간단히 제공함으로써 구현된다. 그후, 나머지 어드레스들이 그 동작 모드(sequential 또는 interleave)에 따라 SDRAM에서 자동적으로 발생된다. 이 동작 모드는 CPU로부터 어드레스 시퀀스에 의해 결정된다. 각각의 버스트 어드레스 시퀀스 방법에 대한 어드레스가 내부 카운터의 출력과 버스트 시작 어드레스의

가산(addition)으로, 시퀀셜 모드에서, 발생된다. 인터리브 모드에서는, 어드레스들이 내부 카운터의 출력과 버스트 시작 어드레스의 배타적 OR에 의해 발생된다. 동일한 랩 모드(wrap mode)가 읽기 및 쓰기 동작에 대해 사용되며, 모든 열 어드레스 비트들이 읽기 및 쓰기 동작에 대해 사용된다.

클럭 속도가 200 MHz 이상으로 증가하면(즉, RDRAM, 또는 SDRAM), DRAM의 핵심 동작은 동일한 속도로 증가하지 않았다. 따라서, DRAM은 내부적으로 4 또는 8 워드로 읽기 및 쓰기를 완료하였고 이 워드를 외부 버스에 대해 순차적으로 출력하였다. 전체 데이터 워드 그룹이 전송됨에 따라, 최소 유효 열 어드레스가 DRAM에 더 이상 전달되지 않았다.

이 방법은 컨트롤러로부터 DRAM으로의 쓰기 데이터에 대해 잘 적용된다. 왜냐하면 이 데이터가 캐시 필(cashe fill)에 정렬될 수 있기 때문이다. 그러나, 데이터 워드들의 완전한 블록이 읽기와 동시에 전송되기 때문에, 최대 유효 워드가 컨트롤러에 의해 항상 먼저 수신되지 않는으며, 이는 시스템에 대기시간(latency)을 일으킬 수 있다. 최대 유효 워드를 먼저 컨트롤러에 전달하면서 데이터 워드의 블록 전송을 지원하는 고속 DRAM 메모리가 필요하다. 이러한 새로운 특징을 지원하기 위해 DRAM과 메모리 컨트롤러간 통신 프로토콜이 또한 필요하다.

### 발명이 이루고자 하는 기술적 과제

본 발명은 두 종류의 액세스(읽기 및 쓰기)를 구현하기 위한 어드레싱 기법 및 관련 하드웨어에 관한 것이다. 발명에 따른 메모리 소자는 다수의 메모리 셀 어레이를 포함한다. 다수의 메모리 셀로부터 정보를 읽어들이고 다수의 메모리 셀에 정보를 기록하기 위해 아래와 같은 주변 장치들이 제공된다.

- 어드레스 비트에 따라, 다수의 어레이로부터 수신한 비트를 순서지정하기 위한 리오더 회로(reorder circuit), 그리고
- 읽기 동작 중 상기 리오더 회로에 상기 어드레스 비트 일부를 전달하기 위한 어드레스 시퀀서(address sequencer).

\*본 발명의 방법은 메모리 소자로부터 한개 이상의 n-비트 워드를 출력하기 전에 어떤 어드레스 비트의 정보에 따라 메모리 어레이로부터 출력되는 n-비트 워드들의 블록을 순서지정(reordering)하는 과정을 포함한다. 일례의 실시예에서, 이 방법은 DRAM에 액세스하기 위한 것으로서,

- 뱅크 어드레스 입력 상의 값들을 이용하여 어레이 뱅크를 선택하고,
- 입력 A3-Ai 상에 제공되는 열 어드레스를 이용하며, 이때, I는 최대 유효 열 어드레스이며,
- 입력 A0-A2 상에 제공되는 열 어드레스를 이용하며, 읽기 액세스에 대한 버스트 순서를 식별하고, 그리고
- 쓰기 액세스 중 입력 A0-A2 상에 제공되는 열 어드레스를 무시하는,

이상의 단계를 포함한다. 따라서, 읽기 동작에 대하여 특정 8비트 버스트가 최대 유효 열 어드레스 비트로 식별되고, 이때, 최소 유효 비트 CA0-CA2는 최대 유효 워드와, 최대 유효 워드 다음의 읽기 랩 시퀀스(read wrap sequence)를 식별한다. 읽기의 경우, 버스트가 최대 유효 열 어드레스에 의해 식별되며, 이때, CA0-CA2는 "don't care" 비트(000)이다. 다른 구현 기법도 가능하다.

쓰기 액세스와는 다른 읽기 액세스를 가지는 경우의 중요한 특징은 인터리브 버스트 모드가 지원되도록 유효 워드가 메모리 컨트롤러에 가용해지는 방식으로 읽기가 실행되는 것이다. 다른 한편 쓰기는 쓰기 데이터가 캐시에 보유된 데이터로부터 발생될 수 있기 때문에 시작 시퀀셜 버스트를 바탕으로 단순화될 수 있다. 본 발명은 임계 워드를 메모리 컨트롤러에 먼저 제공함으로써 시스템의 대기시간(latency)을 개선시킬 수 있다. 또한, 본 발명의 시스템은 읽기 및 쓰기 명령간 열 어드레스 비트를 다시 순서지정(reorder)할 필요가 없다.

### 발명의 구성 및 작용

도 3은 본 발명의 버스트 읽기 순서지정을 구현할 수 있는 DRAM에 대한 구조의 단순화된 블록도표이다. DRAM 메모리 소자(29)는 명령 버스나 명령 라인, 그리고 어드레스 버스나 어드레스 라인에 따라 반응하는 명령/어드레스 입력 버퍼(30)로 구성된다. 명령 디코더 및 시퀀서(32)와 어드레스 시퀀스(34)가 명령/어드레스 입력 버퍼(30)에 연결된다.

뱅크 어드레스 디코더(36)는 어드레스 시퀀서(34)에 연결되며, 뱅크 제어 로직(38)은 뱅크 어드레스 디코더(36)에 연결된다. 행 래치/디코더/드라이버 시리즈(40)가 뱅크 제어 로직(38)과 어드레스 시퀀서(34)에 연결된다. 각각의 메모리 블록(42)에 대해 한개의 행 래치/디코더/드라이버(40)가 제공된다. 뱅크 0에서 뱅크 7까지로 표시되는 8개의 메모리 어레이가 도 3에 도시된다. 따라서, 뱅크 0에서 뱅크 7까지 중 하나에 각각 연결되는 8개의 행 래치/디코더/드라이버 회로(40)가 존재한다.

열 래치/디코더 회로(44)는 어드레스 시퀀서(34)에 연결된다. i/o 게이팅 회로(46)는 메모리 어레이(44) 각각 내에 센스 증폭기를 제어하기 위해 열 래치/디코더 회로(44)에 연결된다. 열/어드레스 입력 버퍼(30), 명령 디코더 및 시퀀서(32), 어드레스 시퀀서(34), 뱅크 어드레스 디코더(36), 뱅크 제어 로직(38), 행 래치/디코더/드라이버(40), 열 래치 디코더 회로(44), 그리고 i/o 게이팅 회로(46)가 명령 버스 및 어드레스 버스에 연결되는 다수의 제 1 주변 장치라고 간주된다. 다수의 제 1 주변 장치로 앞서 언급한 소자들의 설명은 현재 선호되는 실시예의 설명을 제공하고자 의도한 것으로서, 발명의 범위를 제약하고자 하는 의도가 아니다.

DRAM(29)은 쓰기 동작이나 읽기 동작을 위해 다수의 데이터 패드(48)를 통해 액세스될 수 있다. 쓰기 동작의 경우, 데이터 패드(48) 상의 데이터가 수신기(50)에 의해 수신되어 입력 레지스터(52)에 전달된다. 쓰기 버퍼(54)는 i/o 게이팅 버퍼(46)를 통해 메모리 어레이(42)로의 입력을 위해 쓰기 래치 및 드라이버 회로(56)에 입력되는 수신 데이터를 버퍼링한다.

메모리 어레이(42)로부터 읽어들이는 데이터는 i/o 게이팅 회로(46)를 통해 읽기 래치(58)에 출력된다. 읽기 래치(58)로부터, 드라이버(62)를 통해 데이터 패드(48)에 데이터를 출력하는 멀티플렉서/리오더 회로(60)에 정보가 입력된다. 수신기(50), 입력 레지스터(52), 쓰기 버퍼(54), 쓰기 래치 및 드라이버 회로(56), i/o 게이팅 회로(46), 읽기 래치(58), 맥스/리오더 회로(60), 그리고 드라이버(62)는 데이터에 연결되는 다수의 제 2 주변 장치를 포함한다. 다수의 제 2 주변 장치로 앞서 언급한 소자들의 설명은 현재 선호되는 실시예의 설명을 제공하고자 의도한 것으로서, 발명의 범위를 제약하고자 하는 의도가 아니다.

일반적인 의미로서, 리오더 회로(60)의 용도는 어떤 어드레스 비트의 정보에 따라 메모리 어레이(42)로부터 출력되는 n-비트 워드들의 블록을 순서지정하는 것이다. 도 3에 도시되는 바와 같이, 맥스/리오더 회로(60)의 입력에서 가용한 8개의 8비트 워드가 존재한다. 맥스/리오더 회로(60)는 열 어드레스(CA0-CA2)의 세 개의 최소 유효 비트를 또한 수신한다. 이 세 최소 유효 비트는 8개의 8비트 워드들의 블록에서 최대 유효 워드를 식별하여, 이 워드가 먼저 출력되게 한다. 그리고 랩이 시작되고, 즉, 읽기가 상위 유효 워드로 시작되고 상위 유효 워드가 위치 0의 워드와는 다른 어떤 워드일 경우, 읽기는 읽기 완료를 위해 위치 7로부터 위치 0으로 랩 어라운드(wrap around)를 실행한다.

특히, 발명의 선호되는 실시예에 따르면, 읽기 명령을 수신할 때, 뱅크 어드레스 입력 BA0 및 BA1의 값이 메모리 어레이(42) 중 하나를 선택한다. 각각의 어레이(42) 내 한 행이나 여러 행들을 식별하는 어드레스 정보가 이후 수신된다. 입력 A3-Ai 상에 제공되는 어드레스(i는 x16 파트의 경우 8, x8 파트의 경우 9, x4 파트의 경우 10)가 시작 열 위치를 선택한다. 도 3에 따르면, x8 파트에 대한 입력 A0-Ai 상의 값은 CA3-CA9이다. 최소 유효 비트(CA0-CA2)의 정보는 맥스/리오더 회로(60)에 입력된다. 이 값들은 입력 A0-A2에서 가용하다. 이 정보는 맥스/리오더 회로(60)에 의해 먼저 출력되는 최대 유효 워드를 식별한다. 도 4A, 4B, 4C는 512 메가비트 x4 파트, x8 파트, 그리고 x16 파트에 대한 어드레싱을 각각 도시한다.

쓰기 동작의 경우, 뱅크가 읽기 동작과 동일한 방식으로 식별된다. 마찬가지로, 열 어드레스를 시작하는 것이 동일한 방식으로 식별된다. 그러나, 쓰기 동작 중, 입력 A0-A2에서 가용한 신호들이 무시되고 낮은 값으로 가정된다.

본 발명은, 유효 워드가 컨트롤러에 가용하면서 쓰기 동작이 시작 시퀀셜 버스트로 단순화되도록, 읽기 동작이 인터리브 버스트 모드를 통합하게 하는 어드레싱 기법에 관한 것이다. 선호되는 실시예에서, DRAM에 대한 액세스가 항상 8비트 버스트 길이로 이루어진다. 모든 쓰기 버스트는 CA0=1, CA1=0, CA2=0과 같은 시작 위치로 인덱싱된다. 읽기 동작의 경우, CA0, CA1, CA2는 DRAM(29)으로부터 읽어들이는 제 1 데이터 워드를 명시한다. 나머지 7개의 데이터 워드들은 표 1에 도시되는 바와 같이 읽혀진다.

**표 1: 쓰기 및 읽기 인터리브 시퀀스**

시작액 (CA0-CA1-CA2)	데이터 워드 쓰기 시퀀스	데이터 워드 읽기 시퀀스
000	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7
001	0-1-2-3-4-5-6-7	1-0-3-2-5-4-7-6
010	0-1-2-3-4-5-6-7	2-3-0-1-6-7-4-5
011	0-1-2-3-4-5-6-7	3-2-1-0-7-6-5-4
100	0-1-2-3-4-5-6-7	4-5-6-7-0-1-2-3
101	0-1-2-3-4-5-6-7	5-4-7-6-1-0-3-2
110	0-1-2-3-4-5-6-7	6-7-4-5-2-3-0-1
111	0-1-2-3-4-5-6-7	7-6-5-4-3-2-1-0

도 5는 본 발명이 구현될 수 있는 컴퓨터 시스템(110)의 한가지 예에 대한 블록도표이다. 컴퓨터 시스템(110)은 프로세서(112), 메모리 서브시스템(114), 그리고 확장 버스 컨트롤러(116)를 포함한다. 메모리 서브시스템(114)과 확장 버스 컨트롤러(116)는 로컬 버스(118)를 통해 프로세서(112)에 연결된다. 확장 버스 컨트롤러(116)는 한개 이상의 확장 버스(120)에 또한 연결되며, 다시 대량 저장 장치, 키보드, 마우스, 그래픽 어댑터, 그리고 멀티미디어 어댑터 등과 같은 여러 주변 장치(121-123)에 부착될 수 있다. 프로세서(112)와 메모리 서브시스템(114)은 단일 칩에 집적될 수 있다.

메모리 서브시스템(114)은 메모리 컨트롤러(124)를 포함하고, 메모리 컨트롤러(124)는 다수의 신호 라인(129, 130, 129a, 130a, 129b, 130b, 129c, 130c)을 통해 다수의 메모리 모듈(125, 126)에 연결된다. 다수의 데이터 신호 라인(129, 129a, 129b, 129c)이 메모리 컨트롤러(124)와 메모리 모듈(125, 126)에 의해 사용되어 데이터 DATA를 교환한다. 어드레스 ADDR은 다수의 어드레스 신호 라인(132) 상에서 신호전파되고, 클럭 신호 CLK는 클럭 라인(133) 상에 공급되며, 명령 CMD는 다수의 명령 신호 라인(134) 상에서 신호전파된다. 메모리 모듈(125, 126)은 다수의 메모리 소자(136-139, 136'-139')과 레지스터(141, 141')를 각각 포함한다. 각각의 메모리 소자(136-139, 136'-139')는 고속 동기식 메모리 소자일 수 있다. 두개의 메모리 모듈(125, 126)과 관련 신호 라인(129-129c, 130-130c)만이 도 5에 도시되지만, 어떤 숫자의 메모리 모듈도 사용될 수 있다.

메모리 모듈(125, 126)을 메모리 컨트롤러(124)에 연결하는 다수의 신호 라인(129-129c, 130-130c, 132, 133, 134)이 메모리 버스로 알려져 있다. 메모리 버스(143)는 당 분야에 잘 알려진 추가 신호 라인을 가질 수 있다(가령, 칩 선택 라인). 메모리 버스(143)에 걸쳐있는 메모리 소자들의 각각의 열은 메모리 순위(rank of memory)라고 알려져 있다. 일반적으로, 도 5에 도시되는 바와 같은 단일 측 메모리 모듈(single sided memory module)은 단일 메모리 순위를 가진다. 그러나, 두개의 메모리 순위를 지닌 이중 측 메모리 모듈(double sided memory module)이 사용될 수도 있다.

읽기 데이터는 클럭 신호 CLK에 직렬로 동기화되는 출력으로서, 다수의 클럭 신호 라인(130, 130a, 130b, 130c) 사이에서 구동된다. 읽기 데이터는 클럭 신호 CLK에 직렬로 동기화되는 입력이며, 메모리 컨트롤러(124)에 의해 다수의 클럭 신호 라인(130, 130a, 130b, 130c) 사이에서 구동된다. 명령 및 어드레스들은 메모리 모듈(125, 126)의 레지스터(141, 141') 사이에서 메모리 컨트롤러(124)에 의해 터미네이터(148)에게로 구동되는 클럭 신호 CLK를 이용하여 또한 클럭된다. 명령, 어드레스, 클럭 신호 라인(134, 132, 133)은 각각 메모리 모듈(125, 126)의 레지스터(141, 141')에 직접 연결된다. 레지스터(141, 141')는 메모리 모듈(125, 126)의 메모리 소자(136-139, 136'-139')에 분배되기 전에 이 신호들을 버퍼링한다.

**발명의 효과**

쓰기 액세스와는 다른 읽기 액세스를 가지는 경우의 중요한 특징은 인터리브 버스트 모드가 지원되도록 유효 워드가 메모리 컨트롤러에 가용해지는 방식으로 읽기가 실행되는 것이다. 다른 한편 쓰기는 쓰기 데이터가 캐시에 보유된 데이터로부터 발생될 수 있기 때문에 시작 시퀀셜 버스트를 바탕으로 단순화될 수 있다. 본 발명은 임계 워드를 메모리 컨트롤러에 먼저 제공함으로써 시스템의 대기시간(latency)을 개선시킬 수 있다. 또한, 본 발명의 시스템은 읽기 및 쓰기 명령간 열 어드레스 비트를 다시 순서지정(reorder)할 필요가 없다.

**(57) 청구의 범위**

**청구항 1.**

- 메모리 어레이로부터  $n$  비트 워드들의 블록을 출력하기 전에, 상기  $n$ -비트 워드 중 임의의 한 워드가 먼저 출력될 수 있도록, 소정의 어드레스 비트의 정보에 따라 메모리 어레이로부터 출력되는 상기  $n$  비트 워드들의 블록을 순서지정하는 단계를,
- 상기 메모리 어레이에 한개 이상의  $n$ -비트 워드를 입력하기 전에 상기 어드레스 비트를 무시하는 단계를 포함하는 것을 특징으로 하는 방법.

## 청구항 2.

제 1 항에 있어서, 상기 방법은,

- 열 어드레스의 두개 이상의 최소 유효 비트를 검사하는 단계를 추가로 포함하며, 이때, 상기 검사하는 단계에 따라 상기 순서지정하는 단계가 구현되는 것을 특징으로 하는 방법.

## 청구항 3.

제 1 항에 있어서, 상기 방법은,

- 열 어드레스의 소정의 비트와 행 어드레스에 따라 다수의 메모리 어레이로부터  $n$  비트 워드들의 블록을 출력하는 단계를 추가로 포함하고, 이때, 상기 순서지정하는 단계는,
- 열 어드레스의 앞서와는 다른 소정의 비트에 따라,  $n$  비트 워드들의 메모리 소자로부터의 출력 순서를 제어하는 단계를 포함하는 것을 특징으로 하는 방법.

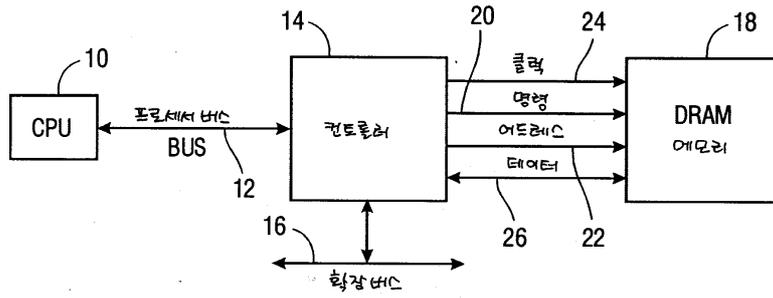
## 청구항 4.

제 3 항에 있어서, 상기 출력하는 단계는,

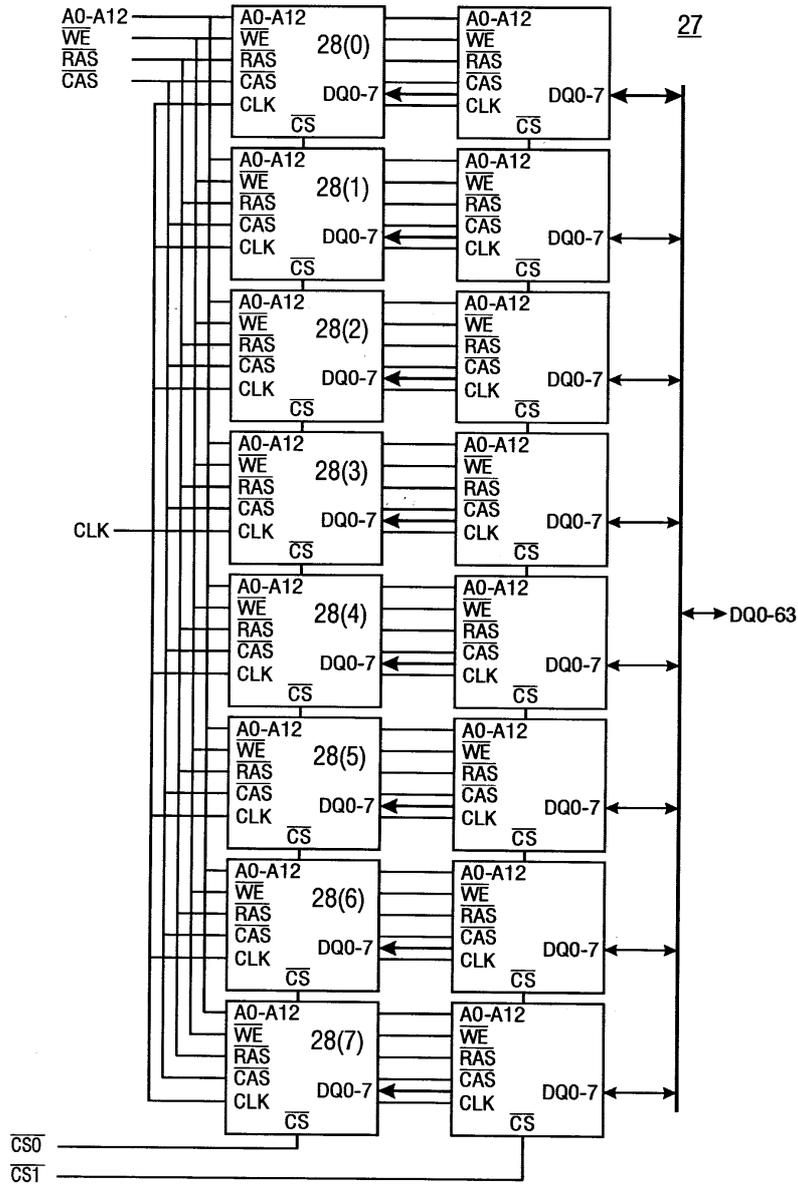
- 뱅크 어드레스 입력에 대한 값들을 이용하여 어레이 뱅크를 선택하는 단계,
- 입력  $A3-Ai$  상에 제공되는 열 어드레스를 이용하여, 읽기 및 쓰기 액세스 중 한가지에 대한 시작 열 위치를 식별하는 단계로서, 이때,  $i$ 는 최대 열 어드레스와 같은 단계,
- 입력  $A0-A2$  상에 제공되는 열 어드레스를 이용하여 읽기 액세스에 대한 버스트 순서를 식별하는 단계, 그리고
- 쓰기 액세스 중 입력  $A0-A2$  상에 제공되는 열 어드레스를 무시하는 단계를 추가로 포함하는 것을 특징으로 하는 방법.

도면

도면1



도면2





도면4b

512M(X8) 어드레싱

ADD PIN	ROW	COL
A13		
A12	ROW 12	
A11	ROW 11	
A10	ROW 10	
A9	ROW 9	COL 9
A8	ROW 8	COL 8
A7	ROW 7	COL 7
A6	ROW 6	COL 6
A5	ROW 5	COL 5
A4	ROW 4	COL 4
A3	ROW 3	COL 3
A2	ROW 2	COL 2
A1	ROW 1	COL 1
A0	ROW 0	COL 0

도면4c

512M(X16) 어드레싱

ADD PIN	ROW	COL
A13		
A12		
A11	ROW 11	
A10	ROW 10	
A9	ROW 9	COL 9
A8	ROW 8	COL 8
A7	ROW 7	COL 7
A6	ROW 6	COL 6
A5	ROW 5	COL 5
A4	ROW 4	COL 4
A3	ROW 3	COL 3
A2	ROW 2	COL 2
A1	ROW 1	COL 1
A0	ROW 0	COL 0

도면5

