(19) 대한민국특허청(KR)
(12) 특허공보(B1)

(51) Int. Cl.[6]
G06F 12/00

(45) 공고일자 1996년12월11일
(11) 공고번호 특1996-0016397
(24) 등록일자 1996년12월11일

(21) 출원번호 특1994-0004411
(22) 출원일자 1994년03월08일
(65) 공개번호 특1994-0022272
(43) 공개일자 1994년10월20일
(30) 우선권주장 93-051041 1993년03월11일 일본(JP)

(73) 특허권자 가부시끼가이샤 히다찌세이사꾸쇼 가나이 쯔또무
일본국 도꾜도 지요다구 간다 스루가다이 4-6
(72) 발명자 가따야마 구니히로
일본국 요꼬하마시 도쯔까구 요시다쵸 1545
가끼 겐이찌
일본국 자마시 이리야 3-5816-2
기따하라 쥰
일본국 요꼬하마시 미나미구 나가따다이 9-3
도비따 쯔네히로
일본국 요꼬하마시 도쯔까구 마이오까쵸 850
후루사와 가즈노리
일본국 고다이라시 오가와히가시쵸 1-39-4-502
(74) 대리인 백남기

심사관 : 오홍수 (책자공보 제4748호)

(54) 화일기억장치 및 그것을 사용한 정보처리장치

요약

요약 없음

대표도

도1

명세서

[발명의 명칭]

화일기억장치 및 그것을 사용한 정보처리장치

[도면의 간단한 설명]

제 1 도는 본 발명을 실현하는 정보처리장치의 시스템구성을 도시한 블럭도.

제 2 도는 화일기억장치의 실시예의 구성을 도시한 블럭도.

제 3 도는 실시예의 동작을 설명하기 위한 주요부분의 블럭도.

제 4 도는 실시예에 있어서의 라이트가 느린 메모리칩을 사용한 경우 데이타 제어를 도시한 설명도.

제 5 도A, 제 5 도B는 실시예에 있어서의 메모리로의 화일저장의 실시예의 설명도.

제 6 도는 본 발명의 다른 실시예에 있어서의 연쇄정보저장의 실시예의 설명도.

제 7 도는 신호단자 입력지정에 의한 데이타 분쇄기능을 메모리소자내에 구비한 메모리 구성예의 설명도.

제 8 도는 커맨드 설정 입력지정에 의한 데이타 분배기능을 메모리소자내에 구비한 메모리 구성예의 설명도.

제 9 도는 실시예에 있어서의 메모리군의 어드레스접속의 설명도.

제10도는 실시예에 있어서의 4바이트 동시 라이트시의 동작의 설명도.

제11도는 실시예에 있어서의 4바이트 동시 라이트시의 동작의 설명도.

제12도는 실시예에 있어서의 메모리 1칩 동시 액세스의 데이타 분배의 설명도.

제13도는 실시예에 있어서의 메모리 2칩 동시 액세스의 데이타 분배의 설명도.

제14도는 실시예에 있어서의 메모리 4칩 동시 액세스의 데이타 분배의 설명도.

제15도는 시스템버스와 메모리의 데이타 버스의 여러가지 구성에 있어서의 하드웨어 구성의 설명도.

[발명의 상세한 설명]

본 발명은 화일기억장치를 탑재한 정보처리장치에 관한 것으로, 특히 고속의 화일 액세스에 적합한 화일 기억장치 및 그것을 사용한 정보처리장치에 관한 것이다.

퍼스널 컴퓨터를 비롯한 현재의 범용의 정보처리장치에 있어서 화일기억장치는 거의 필수라고 할 수 있는 주변기기이다. 게다가, 화일기억장치는 본체의 기기에 내장되어 있는 것이 일반적이고, 사용자는 항상 대용량의 화일을 취급할 수 있게 되어 있다.

최근에는 노트사이즈나 팜톱형의 퍼스널 컴퓨터가 사용되는 경우가 많아져 이동해서 사용하는 용도가 중요시되기 시작하였다. 이 때문에 진동에 약하고 소비전력이 큰 자기디스크 기억장치 대신에 반도체를 기억 매체로 한 화일기억장치가 주목받고 있다. 예를 들어, 일본국 특허공개공보 평성 2-292798호에는 기억매체로서 플래쉬메모리를 사용한 반도체 화일기억장치의 기술에 개시되어 있다.

플래쉬메모리라는 것은 전기적으로 리라이트 가능한 불휘발성 메모리로서, 대용량 저가격화가 가능하므로, 반도체 화일기억장치의 기억메체로서 가장 유효한 메모리의 하나이다. 상기한 일본국 특허공개공보에 기재된 기술은 이 메모리를 사용해서 화일기억장치를 구축할때 문제로 되는 점을 해결하고, 사용상 편리함을 향상시키는 연구를 하고 있다. 예를 들어 리라이트의 빈발에 의해 소자가 저하하는 플래쉬메모리의 결점을 구제하는 방법 및 플래쉬메모리의 리라이트에 필요한 소거라는 포즈에 있어서의 고속화의 방법등을 제안하고 있다. 또, 호스트로 되는 정보처리장치와의 인터페이스로서는 자기디스크장치와 동일한 것을 제안하고 있고, 자기디스크장치로 대체할 수 있는 시스템의 구축을 목적으로 하고 있다.

상기 종래기술의 반도체 화일기억장치는 그것을 사용하는 정보처리장치의 기존의 인터페이스버스를 사용하고, 자기디스크장치와의 호환성을 중시하고 있다. 이것에 의해 반도체 화일기억장치를 사용자가 저항없이 받아들이도록 하고 있지만, 자기디스크장치와의 호환성을 중시하는 반면, 자기디스크에 대한 반도체 기억소자의 우위성의 이용에 고려가 이루어져 있지 않다.

예를 들면, 반도체 기억소자는 회전하는 디스크에서 데이타를 수집하거나, 또는 라이트를 실행하는 자기디스크장치와 달리 정적인 기억매체이므로 매우 고속인 데이타 액세스를 가능하게 하지만, 자기기억장치와 동일한 인터페이스에서는 이 고속 액세스의 우위성을 살릴수 없다는 문제가 있다.

또, 현상의 퍼스널 컴퓨터등의 정보처리장치에 사용되고 있는 자기디스크장치에서는 데이타 액세스가 퍼스널 컴퓨터의 메모리액세스에 비해서 늦어 정보처리장치내의 CPU와 동기해서 동작할 필요는 없다. 그 때문에 자기디스크장치의 데이타의 수수는 비동기의 버스상에서 실행되고 있다. 반도체를 기억매체로 한 경우에는 CPU의 동작에 추수할 수 있으므로, 동기동작을 하는 것에 의의가 있다.

그러나 이때, CPU 처리의 버스폭과 플래쉬메모리 1칩의 데이타 액세스의 버스폭의 차이를 메모리칩을 병렬해서 사용하는 것에 의해 해결하려고 하면 문제가 발생한다. 플래쉬메모리에서는 소거단위가 정해져 있고, 예를 들어 512바이트이다. 따라서 여러개의 칩을 병렬로 이용하면 한번에 소자를 실행하는 소거단위의 용량(512×병렬칩수)이 바이트로 되어 버린다.

한편, 현재의 퍼스널 컴퓨터의 대부분은 512바이트를 1단락(1섹터)로 해서 화일관리의 기억용량단위(이하, 화일관리단위)로 하고 있으므로, 화일관리단위인 1섹터의 액세스에 있어서 예를 들어 플래쉬메모리 4칩을 병렬에 이용하려고 하면, 결과적으로 최소 소거단위가 커지고, 이 섹터의 리라이트시에 상술한 바와 같이 1섹터의 4배의 용량을 동시에 소거하게 된다. 이것으로는 소거대상 용량이 너무커서 소거대상 이외의 데이타까지 소거해 버리게 된다.

본 발명의 목적은 반도체 화일기억장치를 구비한 정보처리장치에 있어서 고속 액세스성능과 경제성을 양립시킨 화일기억장치 및 화일저장방법을 제공하는 것이다.

본 발명의 다른 목적은 최저 소거단위가 어느정도 큰 플래쉬메모리를 사용한 화일장치에 있어서 다른 화일데이타에 영향을 미치지 않는 화일기억장치 및 화일저장방법을 제공하는 것이다.

또, 본 발명의 또다른 목적은 화일기억장치의 소형화를 도모하는 것이다.

또, 본 발명의 또다른 목적은 화일저장방법을 동적으로 변경하고, 여러가지 사이즈로 이루어지는 여러개의 화일을 저장하는 과정에서 발생하는 간극적인 빈영역을 유효하게 이용하는 것이다.

본 발명에 의한 화일장치는 최소 소거단위가 화일기억장치의 데이타 버스폭보다 크고, 데이타 액세스폭이 화일기억장치의 데이타 버스폭보다 작은 기억소자군을 병렬로 이용해서 데이타 버스를 거쳐서 화일데이타를 저장하는 화일기억장치로서, 1개 이상의 최소 기억단위로 이루어지는 화일데이타를 임의의 최소 기억단위의 조합으로 이루어지는 조합 요소로 분할하는 화일분할수단, 데이타 버스상의 데이타를 데이타 액세스폭을 단위로 해서 임의로 조합하고, 데이타 액세스폭을 단위로 한 동일한 단위수의 기억소자군의 임의의 조합에 대응시키는 데이타 분배수단, 상기 조합요소의 각각을 화일기억장치에 저장할때 기억소자군의 상기 임의의 조합중의 어느것인가 1조에 대응시킬 수 있도록 데이타 분배수단을 제어하는 제어수단을 구비한다.

이 결과, 본 발명에 의하면, 플래쉬메모리와 같이 최소 소거단위가 큰 기억매체를 사용한 화일기억장치를 탑재한 정보처리장치에 있어서, 자기디스크장치와 비교해서 우수한 고속 액세스기능을 최대한으로 발휘할 수 있다. 정보처리장치의 CPU에 입력하는 레디신호에 의해 적절히 CPU를 대기상태로 해서 타이밍을 맞출 수 있으므로, 시스템의 데이타 버스폭과 플래쉬메모리 1칩의 액세스 데이타 비트수가 다른 경우라도 고속으로 액세스하는 것이 가능하게 된다.

특히 액세스시간이 사용자가 인식하기 쉬운 대용량이 화일의 리드 라이트 즉 연속 섹터 액세스에 있어서 효과가 크다. 또, 시스템의 데이타 버스폭과 플래쉬메모리 1칩의 액세스 데이타 비트수는 용도, 성능, 시

대에 따라 다르지만, 여러가지의 것에 대응할 수 있는 유연성이 높은 구성을 가능하게 한다. 또 메모리를 인터리브 엑세스해서 고속화하는 방법에도 응용할 수 있다.

상기 구성을 사용한 화일저장방법에 의하면, 시스템으로부터의 액세스 순서를 간략화하여 고속 액세스를 원조한다. 그리고 화일관리 자체를 간략화할 수 있어 컨트롤회로나 제어프로그램등의 간소화가 도모된다.

또, 메모리소자중에 데이타 분배기능을 구비하는 것에 의해, 주변회로의 삭감, 데이타처리의 고속화가 도모된다.

또, 이상 기술한 바와 같은 본 발명의 개념에 의하면, 플래쉬메모리장치에 있어서, 화일 리라이트시에 다른 화일 데이타를 소거하는 일 없이, 또 메모리의 기억영역을 유효하게 이용하면서 여러개의 바이트 병렬 동시 라이트에 의해 고속의 화일 데이타 저장을 달성하는 것이 가능하게 된다.

이하, 본 발명의 개념을 도면을 사용해서 설명한다.

이하의 설명에서는 정보처리장치로서 퍼스널 컴퓨터, 반도체 기억매체로서 플래쉬메모리를 전제로 한다.

종래예의 문제로서 기술한 바와 같은 소거대상외의 화일데이타까지 소거해버리는 사태를 피하기 위해, 예를 들어 CPU가 32비트버스, 플래쉬메모리가 8비트버스이면, 1칩을 4회 순차로 액세스해서 CPU의 32비트버스를 들어 CPU가 32비트버스에 대응할 필요가 있다.

그러나, 4섹터 이상의 연속 액세스시에는 각 4섹터의 데이타에 대해서 4개의 칩에 병렬로 동시 액세스하고, 1회의 액세스로 CPU의 데이타 버스로부터의 32비트를 동시 액세스할 수 있다.

화일관리상 한번 연속 섹터로 처리한 것을 나중에 분할하는 것을 통상 있을 수 없다. 즉, 화일시스템은 화일단위로 밖에 액세스되지 않는다. 따라서, 여러개의 섹터의 연속 액세스의 경우는 라이트의 시점에서 여러개의 칩을 동시 액세스해서 여러개의 섹터를 병렬 사용하도록 하면, 리드시에 동일한 병렬액세스를 하면 라이트한 상태 그대로의 화일 데이타를 얻을 수 있다.

여기에서 2섹터 및 4섹터의 분산 저장 상태에 대해서 제10도 및 제11도에 의해 설명한다. 제10도는 4바이트 동시 라이트의 예를 나타내고, 제11도는 2바이트 동시 라이트의 예를 나타낸다. 여기에서는 호스트의 시스템버스를 32비트, 1개의 메모리군의 데이타 액세스폭을 8비트로 한다.

제10도는 4개의 메모리칩(도면에 따라서 메모리군이라 한다) a~d로의 4섹터 동시 라이트의 예를 나타낸다. 도면중 (212)은 호스트의 시스템버스로부터의 데이타를 나타내고, 32비트(4바이트)를 단위로 해서 전송되어 온다. (222)~(225)는 8비트(1바이트)단위의 데이타의 일련번호를 나타내고, (222)는 1바이트째의 데이타, (223)은 2바이트째의 데이타, (224)는 3바이트째, (225)는 4바이트째의 데이타이다. 이하, 연속해서 4섹터분으로서 2048바이트째까지가 전송되어 온다. 실제로는 32비트버스이기 때문에 1바이트째에서 4바이트째까지가 동시에 수령되고, 이하도 마찬가지이다. (51)~(54)는 데이터 일시기억을 위한 데이타래치이다. 4바이트씩의 동시 라이트이기 때문에 32비트 데이타, 즉 4바이트를 동시에 데이타래치 (51)~(54)에 저장한 후, 각 바이트가 메모리군 a~d로 분산해서 1바이트씩 라이트된다. 다음의 32비트 데이타에 대해서 각 바이트가 메모리군 a~d의 후속 부분에 라이트된다. 이와 같이해서 1섹터분의 512바이트째까지가 128회로 나뉘어 라이트된다. 계속해서, 제 2 의 섹터에 대해서 마찬가지로 4바이트씩 메모리군 a~d의 후속부분에 라이트된다. 마찬가지로, 제 3 섹터의 512바이트를, 또 제 4 섹터의 512바이트를 라이트한다. 이 시점에서 4칩의 최소 소거단위의 용량이 2048바이트째까지의 데이타로 충분하게 된다. 따라서, 이 화일의 리라이트시에는 이 4섹터분의 용량을 일괄하여 소거해서 문제없다.

이상에서 설명한 바와 같이, 실제로는 각 섹터의 데이타는 특정한 메모리군에만 저장되는 것이 아니라, 메모리군 a~d로 분산해서 라이트된다. 단, 4섹터 전체의 데이타로서는 메모리군 a~d 4섹터분의 영역에 저장되게 된다.

이와 같이해서 화일을 4섹터씩 동시에 4칩 병렬 액세스를 반복한다. 최후의 4섹터에 못미치는 데이타가 남은 경우, 또는 처음부터 4섹터에 못미치는 화일인 경우에는 다음과 같이 실행된다.

3섹터인 경우에는 CPU의 32비트를 3으로 나눌수 없으므로, 2섹터와 1섹터로 나누어 대처한다.

2섹터인 경우에 대해서 제11도를 사용해서 설명한다. 제11도는 메모리군 a와 메모리군 b로의 2바이트 동시 라이트의 예를 도시한다.

2바이트씩의 동시 라이트이므로, 32비트 데이타, 즉 4바이트를 동시에 데이타래치 (51)~(54)에 저장한 후, 이것이 2바이트씩 이분되어 전반의 2바이트가 우선 메모리군 a와 메모리군 b에 1바이트씩 라이트되고, 다음에 후반의 2바이트가 메모리군 a와 메모리군 b에 1바이씩 라이트된다. 이와 같이해서 1섹터분의 512바이트째까지가 256회로 나뉘어 라이트된다. 계속해서 제 2 의 섹터의 1바이트째부터 마찬가지로 해서 메모리군 a,b의 후속부분에 256호로 나뉘어 라이트된다.

1섹터의 경우는 도시하지 않지만, CPU로부터의 32비트의 데이타를 4분할하고, 한번에 8비트씩 1칩에 액세스한다. 1섹터의 512바이트는 512회로 나뉘어 어느 칩의 1섹터 영역에 저장된다.

이상과 같이, 연속 액세스하는 섹터수에 의해 플래쉬메모리로의 액세스방법을 바꾸는 것으로 액세스의 고속화가 가능하게 된다. 또, 데이타가 저장방법을 바꾸는 것에 의해 플랫쉬메모리에 따른소거의 문제를 해결할 수 있다. 즉, 어떠한 저장방법에 있어서도 1칩의 최소 소거영역내에는 동일한 화일의 데이타만이 저장되는 것이 보증되므로, 플래쉬메모리의 소거에 따른 상술한 문제가 해소된다.

또, 본 발명에서는 플래쉬메모리를 CPU와 동기동작시키지만, 액세스형식에 의해 지연이 발생한다. 이 문제에 대해서는 CPU에 레디신호 입력에 있는 것을 사용하는 것에 의해, 임의로 CPU에 대기상태를 요구하는 것에 대응한다. 레디신호 입력이 있는 CPU는 현재에는 일반적이다. 예를 들면 현재의 범용의 퍼스널 컴퓨터에 탑재되어 있는 CPU로서 가장 일반적인 INTEL사의 16비트처리 이상의 CPU에는 모두 구비되어 있다. 이 레디신호를 네게이트하는 것에 의해 CPU는 처리 사이클을 정지하므로, 액세스 데이타의 리드, 라이트가 종료해 있지 않으면 이 신호를 네게이트하는 것만으로 좋고, 액세스가 종료한 시점에서 어서트하면

CPU는 처리의 실행을 개재한다. 이들은 화일기억장치가 CPU와 동기동작을 해서 비로서 가능하게 되는 제어이므로, 이때 동기한 클럭을 쌍방으로 입력해서 동기동작시키는 것은 필수이다. 이들 제어에 의해 1회의 처리에 있어서의 데이타폭이 CPU와 플래쉬메모리에서 다른 경우에는 데이타폭이 일치할 때까지 CPU를 대기시키는 것에 의해 용이하게 대응가능하다.

다음에, 본 발명에 있어서의 상세한 실시예를 도면을 사용해서 설명한다.

우선, 제 1 도에 표준적인 퍼스널 컴퓨터의 구성에 본 발명의 플래쉬메모리 화일기억장치를 설치한 것을 도시한다. 도면중, (1)은 데이타나 프로그램의 처리를 담당하는 CPU이고, 그 데이타 버스폭은 32비트인 것으로 한다. (2)는 시스템 전체의 동기클럭을 발생하는 클럭발생기이다. (3)은 시스템 내부의 공유버스로 데이타 버스, 어드레스버스, 메모리커맨드, IO 커맨드등을 포함하고, 총칭해서 시스템버스라 한다. (4)는 본 발명의 플래쉬메모리 화일기억장치의 화일관리나 메모리제어를 실현하는 화일제어회로이다. (5)는 플래쉬메모리 화일기억장치의 기억매체로 되는 플래쉬메모리어레이이고, 메모리 1칩의 액세스 데이타 비트수는 8비트인 것으로 한다. 또, 화일제어회로(4), 플레쉬메모리어레이(5)로 플래쉬메모리 화일기억장치를 구성한다. (6)은 시스템의 주기억을 관리제어하는 주기억 제어회로이다. (7)은 주기억으로 DRAM 등을 사용한다. (8)은 주변 IO 버스를 제어하는 회로이고, 주변 IO 장치의 하나로서 표시제어회로(9)와 통신장치(11)이나 대용량 외부기억장치(12)등이 접속되어 있다. 표시제어회로(9)에는 표시장치(10)이 접속되어 있다.

통상, 주변 IO 제어회로(8)에는 다른 클럭발생기가 내장되어 있고, 이들 주변 IO 장치는 그 클럭의 주기에 따라서 동작한다. 그러나 고속화를 위해 내부 시스템버스(3)에 직접 접속해서 CPU(1)과 동기동작하는 것도 고려된다. (13)은 내부 시스템버스(3)에 접속된 회로를 동기시키기 위해 CPU(1)을 포함하는 각 회로에 공급되는 클럭신호이다. 단, CPU(1)에 부여되는 클럭과 똑같을 필요는 없고, 회로에 따라서는 분주한 것이라도 동기해 있으면 좋다.

(14)는 CPU(1)에 입력되는 레디제어신호이고, 각 회로에서 출력된 상태제시신호를 레디제어신호(15)에서 총괄해서 CPU(1)에 입력한다. (16)은 본 시스템의 사용자가 바라는 처리를 지시하기 위한 입력장치의 제어회로이고, (17)은 입력장치이다. 도면에서는 입력장치(17)을 키보드로 하고, 그 제어회로(16)은 키보드 컨트롤러(KBDC)로 하고 있다.

다음에, 제 1 도의 시스템의 동작을 설명한다. 통상의 동작시는 사용자로부터 입력장치(17)에 의해 지시된 처리를 실행하도록 주기억(7)에 저장되어 있는 프로그램이나 데이타를 CPU(1)이 연산처리하고, 그 결과를 표시장치(10)에 표시한다. 또, 필요하다면 통신장치(11)을 가동하거나, 대용량 외부기억장치(12)로 대용량 데이타의 저장을 실행한다. 그리고, 화일을 인출하거나 저장하는 동작시에는 화일제어회로(4)를 동작시킨다. 시스템의 가동시도 이곳에서부터 시스템 프로그램을 로드하게 된다. 이들 동작시에는 클럭발생기(2) 가 발생하는 클럭신호(13)에 의해 동기동작하고, 어떤 회로가 CPU(1)에 대기를 요구할 필요가 발생하면, 그 회로가 레디제어신호(15)에 CPU 대기를 요구하고, 레디제어신호(14)의 네게이트에 의해 CPU(1)에 그것이 전달된다. CPU(1)은 레디제어신호(14)가 재차 어서트될때까지 대기를 계속하게 된다. 이때, 화일제어회로(4)는 CPU(1)에 요구되는 화일수에 의해 CPU 대기시간을 증감하는 제어를 실행할 수 있게 되어 있다. 이 구성을 제 2 도에 의해 설명한다.

제 2 도는 플래쉬메모리 화일기억장치의 내부구성을 설명하는 도면이다. 도면중, (3),(4),(5),(13)은 제 1 도와 동일한 것이고, 이하는 화일제어회로(4)의 내부의 구성요소이다.

(21)은 시스템버스(3)과 인터페이스를 실행하는 레지스터군(I/F 레지스터군)이고, (22)는 화일제어회로(4)의 상태를 CPU(1)에 보고하기 위한 스테이터스 레지스터, (23)은 액세스하는 최초의 섹터번호를 설정하는 스타트 섹터 레지스터, (24)는 액세스하는 최후의 섹터번호를 설정하는 엔드 섹터 레지스터, (25)는 CPU(1)이 요구하는 처리를 커맨드코드로서 지시하는 커맨트 레지스터, (26)은 시스템버스와의 데이타의 수수를 실행하는 데이타 레지스터이다. (27)은 화일제어회로(4)의 내부의 제어를 총괄하는 컨트롤러이고, 1칩 마이컴등의 프로그램 가능한 인텔리전트한 LSI가 이상적이다. (28)은 기억매체인 플레쉬메모리어레이(5)의 제어를 실행하는 메모리제어회로, (29)는 플레쉬메모리어레이(5)에 라이트 또는 리드하는 데이타의 제어를 실행하는 데이타 제어회로이다. (30)은 메모리의 액세스를 고속으로 실행하기 위한 DMA 제어회로이고, 시스템의 클럭신호(13)은 본 회로에 입력된다. (31)은 레디제어신호(15)로 출력하는 상태제시신호, (32)는 이 플래쉬메모리 화일기억장치내의 로컬버스, (33)은 플레쉬메모리를 액세스하기 위한 제어신호 및 어드레스이다.

다음에 제 2 도의 플래쉬메모리 화일기억장치의 동작을 설명한다.

CPU(1)은 플래쉬메모리 화일기억장치로의 액세스의 필요성이 발생하면 시스템버스(3)을 통해서 액세스를 실행한다. 먼저 스테이터스 레지스터(22)의 내용을 리드해서 액세스 가능한 상황인지 확인한다. 그리고 다음에 액세스하는 섹터를 스타트 섹터 레지스터(23)과 엔드 섹터 레지스터(24)에 설정한다. 그리고 요구하는 액세스의 커맨드코드(리드 또는 라이트)를 커맨트 레지스터(25)에 라이트한다. 그리고 재차 스테이터스 래지스터(22)를 리드하여 액세스 가능하면 데이타 레지스터(26)으로의 데이타의 라이트 또는 리드를 실행한다.

이때, 컨트롤러(27)은 이들 인터페이스 레지스터군(21)의 관리를 실행하여 CPU(1)로부터의 요구에 따르게 한다. 즉 스타트 섹터 레지스터(23)이나 엔드 섹터 레지스터(24), 커맨트 레지스터(25)를 리드해서 플레쉬메모리어레이(5)로의 액세스내용을 파악하고, 현재의 상태를 나타내는 코드를 스테이터스 레지스터(22)에 라이트해서 CPU(1)에 보고한다.

데이타의 라이트 또는 리드시, CPU(1)로부터의 액세스 요구에 따르기 때문에 컨트롤러가 직접 플래쉬메모리를 액세스하면 동작속도가 늦어지는 것이 고려되므로, DMA 제어회로(30)에 의해 고속으로 플래쉬메모리 액세스를 실행하여 시스템버스(3)과의 데이타의 수수를 실행하는 구성으로 하고 있다. 그것을 위한 DMA 제어회로(30)이나 메모리제어회로(28)로의 액세스 내용의 설정이나 DMA의 기동등을 컨트롤러(27)이 실행한다.

DMA 제어회로(30)은 DMA를 실행하기 위한 어드레스발생 및 타이밍생성을 실행하고, 메모리제어회로(28)은 그 타이밍에 따라서 액세스신호를 발생한다. 플레쉬메모리어레이(5)는 이들 입력신호에 의해 데이타제어회로(29)와 데이타의 수수를 실행한다. 데이타제어회로(29)는 액세스의 섹터수에 따라서 데이타의 생성을 실행한다.

예를 들어, 1섹터의 라이트 액세스이면 시스템버스(3)으로부터 1회의 액세스에 의해 전송되는 데이타를 플레쉬메모리어레이(5)의 1칩의 라이트 비트수에 일치시킨다. 본 실시예에서는 시스템버스(3)의 데이타 버스폭을 32비트로 하고 있으므로 1회의 액세스에 의한 전송으로 32비트 데이타가 얻어진다. 그리고 플래쉬 메모리 1칩의 데이타폭을 8비트로 하고 있으므로 전송된 데이타를 4회로 나누어 플레쉬메모리어레이(5)에 라이트하게 된다. 그 때문에 32비트 데이타를 4개의 8비트 데이타로 분할하는 처리를 데이타제어회로(29)가 래치회로를 사용해서 실행한다. 또 반대로 1섹터의 리드 액세스이면, 플레쉬메모리어레이(5)를 4회 리드 액세스해서 32비트 데이타를 마련해서 1회의 버스전송으로 한다. 이때 발생하는 시스템버스(3)의 대기시간은 데이타제어회로(29)가 발생하는 상태제어신호(31)에 의한 CPU(1)로의 대기요구로 발생한다.

한편, 여러개의 섹터의 액세스인 경우는 데이타제어회로(29)가 래치하는 데이타를 조절해서 액세스를 고속화하여 정상적으로 실행한다. 예를 들어 4섹터의 연속 리드 액세스이면, 8비트 액세스의 플래쉬메모리를 4칩 동시에 리드하는 것에 의해 32비트의 데이타를 고속으로 액세스한다. 또, 상술한 바와 같이 플레쉬메모리어레이(5)로의 라이트의 시점에서 4칩 동시 라이트 액세스를 실행해 둘 필요가 있다. 그렇게 하지 않으면 데이타의 순번이 달라져 정상적인 화일데이타로 되지 않는다. 단, 화일기억장치를 처리하는 시스템에서는 화일단위로 관리를 실행하는 것이 일반적으므로, 라이트했을 때의 섹터수의 같은 섹터수의 리드액세스하는 것이 보통이고, 섹터수에 의핸 액세스의 형식을 라이트와 리드에서 똑같게 해두면 상술한 정보를 특별히 기록할 필요는 없다.

즉, 예를 들어 5섹터의 연속 라이트시에는 최초의 4섹터는 4칩에 병렬로 동시에 라이트하고, 남은 1섹터는 1칩에 저장한다. 액세스형식을 채택하고, 리드시에도 마찬가지로 리드하면, 항상 정상적인 화일데이타를 액세스할 수 있다. 또, 만약을 위해 플레쉬메모리어레이(5)에 저장되어 있는 섹터단위의 데이타에 데이타 저장 형식의 정보를 기록해 두어도 좋다. 그 기록장소로서는 플래쉬메모리내에 데이타 저장영역 이외의 용장데이타 저장영역이 존재하면 그곳에 저장하는 것이 적당하고, 용장영역이 없는 경우는 다른 기억영역을 설치해서 기록한다.

또, 6섹터의 연속 액세스시에는 4섹터의 병렬 액세스와 2섹터의 병렬 액세스에 의해 처리를 실행한다.

다음에, 이들 액세스신호 및 데이타 제어방법을 제 3 도에 따라서 더욱 상세하게 설명한다.

제 3 도에서는 시스템버스를 32비트, 플래쉬메모리를 8비트의 비트폭으로 했을 때의 구성을 도시하고 있다. 도면중, 기출 번호는 지금까지 설명한 것과 동일한 것이다. (41)은 DMA 제어회로(30)의 어드레스 발생을 위한 카운터이고, 제 2 도에는 도시하지 않았지만, 클럭신호(13)이나 시스템버스(3)의 IO 액세스(커맨드)신호 또는 메모리 액세스(커맨드)신호를 입력하고, 이것에 동기해서 카운터 업한다. (42)는 컨트롤러(27)의 로컬버스(32)가 접속되는 DMA 제어의 기동레지스터이고, 이 레지스터에 코드를 라이트하는 것에 의해 바라는 DMA 전송을 개시할 수 있다. (43)도 역시 로컬버스(32)에 접속되는 섹터 레지스터이고, 액세스하는 섹터번호를 라이트하는 것에 의해 임의의 섹터번호의 DMA 전송을 실행할 수 있다. 실동작상에서는 이 섹터번호의 라이트값을 메모리제어회로(28)에 입력해서 플래쉬메모리의 상위 어드레스나 칩 셀렉트신호의 생성에 사용한다.

(44)는 DMA 전송시에 각 제어회로에서 동기를 취하기 위한 타이밍신호를 발생하는 타이밍제어회로이다. (45)는 카운터(41)과 섹터 레지스터(43)의 값에 따라서 메모리제어회로(28)의 발생하는 메모리어드레스이다. (46)은 메모리어드레스(45)의 발생에 맞춰 생성하는 메모리제어신호이다.

(51),(52),(53),(54)는 각각 1바이트(8비트)의 데이타래치이고, 데이타제어회로(29)내에 있어서의 32비트 데이타와 8비트 데이타 사이의 데이타폭 변환을 위한 4바이트(32비트)의 데이타래치를 구성하고 있다. 시스템버스(3)의 데이타를 D0~D31로 하면, 래치(51)은 D0~D7, 래치(52)는 D8~D15, 래치(53)은 D16~D23, 래치(54)는 D24~D31을 각각 래치한다. (55)는 이들 데이타래치의 래치신호 발생회로이다. (56)은 로컬버스(32)와 접속되어 데이타폭이나 데이타의 나열법을 설정하는 데이타폭 제어설정 레지스터이다. 본 실시예에서는 이 데이타폭 제어설정 레지스터(56)에 리드 액세스시 ? ? ?의 연속 액세스 섹터수로서 '1', '2', '4' 중 어느것인가를 설정하는 것에 의해 래치신호의 발생과 그 타이밍을 래치신호 발생회로(55)에 지시한다. 그리고 데이타래치(51),(52),(53),(54)의 각각에 입력하는 래치신호가 (57),(58),(59),(60)이다. 예를 들어 '1'를 설정하면, 래치신호(57),(58),(59),(60)을 순차로 하나씩 출력하고, 1칩의 플래쉬메모리로부터의 데이타를 4회 액세스해서 32비트의 데이타를 생성하여 시스템버스(3)으로 출력한다. 또 '2'를 설정하면, 래치신호(57),(58)을 동시에 출력하고, 다음에 래치신호(59),(60)으로 동시에 출력하는 것을 교대로 실행하고. 2칩의 플래쉬메모리로부터의 데이타를 2회씩 액세스해서 32비트의 데이타를 생성하여 시스템버스(3)으로 출력한다. 또 '4'를 설정하면 래치신호(57),(58),(59),(60)으로 동시에 출력하고, 4칩의 플래쉬메모로부터의 데이타를 1회만 액세스해서 32비트의 데이타를 생성하여 시스템버스(3)으로 출력한다.

한편, 라이트 액세스시에는 항상 데이타래치(51),(52),(53),(54)에 동시에 래치신호를 보내어 시스템으로부터의 32비트 데이타를 한번에 래치하게 된다. (61)은 데이타래치(51),(52),(53),(54)에 저장된 데이타 또는 메모리(5)로부터의 데이타를 분배하는 데이타 분배회로이다. (62)는 플레쉬메모리어레이(5)와 데이타 분배회로(61)을 연결하는 32비트의 데이타 버스이다. (63)은 데이타 분배회로(61)의 데이타의 방향을 결정하기 위한 리드 라이트신호이고, 인터페이스 레지스터(21)의 커맨트 레지스터(25)등에서 공급을 받는 것이 좋다.

데이타 분배회로(61)은 쌍방향의 버퍼로 되어 있고, 한쪽은 데이타래치(51),(52),(53),(54)에 접속되고, 다른 한쪽은 플레쉬메모리어레이(5)에 접속되어 있다. 플레쉬메모리어레이(5)측에는 32비트분의 입력으로 되어 있고, 플레쉬메모리어레이(5)의 전체 칩을 4조의 메모리군으로 나누어 1조씩 다른 비트군(비트

0~7)(비트 8~15)(비트 16~23)(비트 24~31)을 할당하고, 계32비트로 해서 입력하고 있다. 그리고 리드나 라이트에 의해 방향을 정하고, 각각의 데이타의 분배를 데이타폭 제어설정 레지스터(56)의 설정내용에 따라서 결정한다.

이 데이타의 분배에 대해서 제12도, 제13도, 제14도에 의해 구체적으로 설명한다. 이들 도면은 데이타 분배회로(61)의 데이타 분배열을 데이타폭 제어설정 레지스터(56)의 설정값별로 도시한 것이다. 각각, 제12도는 연속 액세스 섹터수가 1, 제13도는 연속 액세스 섹터수가 2, 제14도는 연속 액세스 섹터수가 4인 경우에 있어서의 데이타 분배를 나타내고 있고, 각 경우에 4개의 메모리 군중 어느것에 액세스하는가에 따라 4종류의 데이타 분배가 있다. 또 리드 라이트도 구별해서 나타내고 있다. 또 연속 액세스 섹터수가 1인 경우에는 4개의 시스템 사이클에서 1액세스로 되고, 연속 액세스 섹터수가 2인 경우에는 2개의 시스템 사이클에서 1 액세스, 연속 액세스 섹터수가 4인 경우에는 1개의 시스템 사이클에서 1액세스로 된다.

연속 액세스 섹터수가 1인 경우는 4개로 분할된 메모리의 어느곳에 액세스하는가에 따라 4종류로 분리되고, 연속 액세스수가 2 및 4인 경우도 어느 메모리군을 기점으로 하는가에 따라 역시 4종류로 분리된다. 즉 메모리의 사용법에 의해 기점을 적절하게 결정할 수 있고, 이와 같이 하는 것에 의해 사용되는 메모리군이 한쪽으로 치우치는 것을 방지할 수 있다. 예를 들어 반드시 메모리군 1을 기점으로 하면 데이타의 분배법은 간소화할 수 있지만, 메모리군 1이 사용되는 비율로 높아져 사용량, 사용빈도 모두 치우침이 발생하는 것이 예상된다. 치우침이 발생하면 최종적으로는 1개의 메모리군을 사용할 수 없게 되어 연속 섹터수가 4인 고속 라이트 액세스를 할 수 없게 되어 버린다. 따라서 기점은 모든 메모리군에 설정가능하게 한다.

또, 연속 섹터수가 1 또는 2인 리드에 있어서는 도면에서 분배회로의 데이타결선을 각각의 사이클에 있어서 따로따로 하고 있지만, 래치신호를 목적의 래치 이외로는 출력하지 않으므로, 각 사이클에서 분할하는 일없이 결선을 함께 해도 좋다. 즉, 연속 섹터수가 1(제12도)인 A의 리드에 있어서는 분배를 순차 사이클에서 래치 1-메모리군 1, 래치 2-메모리군 1, 래치 3-메모리군 1, 래치 4-메모리군 1로 하고 있지만, 모든 사이클에서 모든 래치로의 결선을 메모리군의 1에 접속해도 상관없다. 이것은 데이타의 결선을 하고 있어도 래치신호를 출력하지 않으면 영향이 없기 때문이다.

또 구체적인 설명을 덧붙이면, 예를 들어 1섹터의 라이트 액세스에서는 제12도에 도시한 바와 같이, 32비트의 데이타 버스(62)중 액세스 대상의 플래쉬메모리군의 1개에 접속된 비트군에 대해서 1회째에서는 데이타래치(51)에서, 2회째에서는 데이타래치(52)에서, 3회째에서는 데이타래치(53)에서, 4회째에서는 데이타래치(54)에서, 이하 (51)부터 차례대로 반복해서 데이타를 받아 라이트를 실행한다. 리드에 있어서는 래치신호에 의해 1사이클씩 각 래치로 데이타를 분배하여 32비트 데이타를 정렬한다.

2섹터의 라이트 액세스에서는 즉 데이타폭 2바이트의 라이트이면, 제13도에 도시한 바와 같이 32비트의 데이타 버스(62)중 해당하는 2개의 메모리군에 접속되어 있는 2개의 비트군에 대해서, 1회째의 액세스에서는 데이타래치(51),(52)에서, 2회째의 액세스에서는 (53),(54)에서 데이타를 받고, 이하 교대로 분배한다. 한편, 리드 액세스에서는 방향이 반대로 되어 해당하는 2개의 메모리군에서 1회째의 액세스에서는 데이타래치(51),(52)로, 2회째의 액세스에서는 (53),(54)로 데이타를 분배하고, 이것을 반복한다.

4섹터의 라이트 액세스, 즉 데이타폭 4바이트의 라이트면, 32비트의 데이타 버스(62)는 선두에 해당하는 메모리군부터 순서대로 데이타래치(51),(52),(53),(54)에 접속되고, 1사이클의 액세스로 32비트의 액세스가 종료한다. 리드에서는 방향이 반대로 된다.

이상의 동작을 실행하기 위해 컨트롤러(27)은 DMA 제어회로(30)을 기동하기 전에 지금까지 설명한 레지스터류에 적절한 값을 설정해 둔다.

고속화를 위해 상술한 바와 같이 여러개의 메모리군에 동시에 액세스를 실행하는 경우, 각 메모리군에 부여하는 어드레스에 대해서, 이하 제 5 도A, 제 5 도B를 참조해서 검토한다.

제 5 도A, 제 5 도B에 있어서, (81)~(84)는 각각 메모리군 1~메모리군 4를 나타내고 있다. 제 5 도A는 어느 정도의 화일의 라이트가 실행된 상태를 나타내고, 제 5 도B는 그후 어느 화일에 대해서 갱신되어 그 화일 용량이 증대한 상태를 나타내고 있다. 도시한 데이타기호(m-n)의 m은 화일번호, n은 각 화일을 저장하고 있는 섹터번호이다. 예를 들면, (3-2)는 화일번호 3번의 제 2 섹터를 나타낸다. 단, 이 도면은 각 화일의 1이상의 섹터에 대해서 어느 메모리군의 영역(1섹터분의 용량을 갖는다)이 어떻게 이용되는지를 총체적으로 도시한 것이고, 실제적으로는 2 이상의 섹터의 연속 액세스시에는 각 메모리군의 1섹터 용량 영역내에 단독의 섹터내용만이 저장되는 것은 아니고, 여러개의 섹터의 내용이 분산해서 저장된다. 이점에 관해서는 화일관리의 설명에 있어서 다음에 상세하게 기술한다.

제 5 도A에서 화일번호 1(1-n으로 붙여져 있는 모든 섹터)의 액세스에 있어서 1-1~1-4까지의 동시 액세스에 있어서는 이들의 같은 열에 나열해 있기 때문에 각 메모리군에 동일한 어드레스를 부여할 수 있다. 그러나, 화일번호 2에 있어서는 2-1~2-4까지를 동시 액세스 사용으로 하면, 메모리군 4에는 다른 메모리군과는 다른 어드레스를 부여할 필요가 있다. 이것은 어드레스버스를 메모리의 수만큼 마련할 필요가 있다는 것을 의미한다. 또는 메모리군에 다른 어드레스를 부여하는 것등의 방법이 필요로 된다. 따라서, 이것을 피하기 위해서는 화일번호 2의 액세스에 있어서 2-1은 단독으로 액세스하고, 2-2, 2-3을 동시 액세스하고, 또 2-4는 재차 단독으로 액세스하는 액세스형식을 취할 필요가 있다. 그러나 이것에서는 고속화에 있어서 낭비가 발생한다. 그래서 다른 어드레스를 부여하는 구성예를 제 9 도에 의해 설명한다.

제 9 도는 메모리군수를 4로 했을 때의 메모리제어회로(28)내의 어드레스 생성회로의 구성예이다. 도면중, (201)~(204)는 플레쉬메모리어레이(5)를 구성하는 4개의 메모리군이고, (201)은 제 1 의 메모리군 a, (202)는 제 2 의 메모리군 b, (203)은 제 3 의 메모리군 c, (204)는 제 4 의 메모리군 d이다. (205)는 메모리군 b에 부여하는 상위 어드레스의 래치회로 b, (206)은 메모리군 c에 부여하는 상위 어드레스의 래치회로 c, (207)은 메모리군 d에 부여하는 상위 어드레스의 래치회로 d이다. (208)은 메모리제어회로(28)의 어드레스버스(제 3 도의 (45)에 대응), (209)는 어드레스버스(208)의 하위 어드레스이다. 그 어드레스 비트수는 플레쉬메모리칩의 데이타 소거 최소단위 또는 화일관리의 최소단위의 기억용량을 액세스하는데 상당하는 어드레스 비트수로 한다. 예를 들어, 데이터 소거 최소단위가 512바이트이면, 9비

트로 된다. 플래쉬메모리칩의 데이타 소거 최소단위가 화일관리의 최소단위 이하인 경우는 화일관리의 최소단위에 상당하는 어드레스 비트수로 하는 것이 유효한다. (210)은 어드레스버스(208)의 하위 어드레스(209)를 제외한 상위 어드레스이다. 단, 메모리군을 액세스하는데 불필요한 상위의 어드레스는 제외하는 것으로 한다. (211)은 메모리군 b를 액세스하기 위해 어드레스 래치 b에 기억된 상위 어드레스, (212)는 메모리군 c를 액세스하기 위해 어드레스 래치 c에 기억된 상위 어드레스, (213)은 메모리군 d를 액세스하기 위해 어드레스 래치 d에 기억된 상위 어드레스이다. (214)는 메모리군 a(201)의 메모리제어신호, (215)는 메모리군 b(202)의 메모리제어신호, (216)은 메모리군 c(203)의 메모리제어신호, (217)은 메모리군 d(204)의 메모리제어신호이다.

제 9 도는 구성에 있어서, 메모리군 a에서 메모리군 d까지를 동시에 액세스하는 경우에는 미리 어드레스래치(205)~(207)에 각각의 메모리군을 액세스하는 상위 어드레스를 라이트해 둔다. 그리고, 액세스를 실행할 때에는 하위 어드레스(209)는 모든 메모리군에 공통이므로, 어드레스버스는 메모리군 a만을 액세스하기 위한 어드레스를 공급하고, 메모리군 a 이외의 메모리군은 대응하는 어드레스래치회로에서 상위 어드레스에 의해 액세스를 실행한다. 또, 액세스의 제어는 메모리제어신호(214)~(217)에 맡겨지고, 예를 들어 메모리군 a와 메모리군 d만의 액세스이면, 메모리제어신호(214),(215)만이 액티브로 되게 한다. 또, 예를 들어 메모리군 d만의 액세스에 있어서, 메모리군 a에는 메모리제어신호(214)를 액티브로 하지 않으면 어떠한 어드레스가 부여되어도 문제없다. 이상에 의해 각 메모리군에 다른 어드레스를 부여하는 것이 가능하게 되고, 메모리군마저 4개로 분산되어 있으면, 물리적으로 어드레스가 다른곳에 저장되어 있는 동일 화일의 데이타를 동시에 액세스할 수 있어 고속화의 공헌도를 높일 수 있다. 제 9 도의 구성에서는 메모리군 a의 어드레스래치를 간략화하기 위해 생략하고 있지만, 물론 메모리군 a에도 어드레스래치를 마련하는 것이 유효하면, 그렇게 해도 좋다.

이상의 실시예에 의하면, CPU(1)은 비교적 적은 레지스터에 간단한 지정을 실행하는 것에 의해 바라는 섹터의 리드 라이트 액세스를 고속으로 액세스할 수 있다. 또, 화일 컨트롤러를 1칩 마이컴으로 해서 섬세한 제어를 소프트웨어에 의해 지시할 수 있고, DMA 전송 제어회로를 탑재하는 것에 의해 1칩 마이컴이 시스템보다 늦은 동작밖에 할 수 없어도 데이타전송을 고속으로 실행할 수 있다. 단, 만일 1칩 마이컴의 동작속도가 시스템에 대해서 충분히 추수할 수 있는 동작이 가능하다면, DMA 제어회로는 필요없이 모든 데이타전송을 1칩 마이컴이 실행하는 구성도 고려된다.

또, 본 실시예에서는 데이타 버스를 시스템 32비트, 플래쉬메모리 8비트로 해서 설명을 하였지만, 16비트 동작의 CPU(1)이나 64비트동작의 CPU, 또 16비트 입출력의 플래쉬메모리등에 대해서도 데이타래치의 수나 데이타폭 제어설정 레지스터등의 구성이나 컨트롤러의 제어프로그램등을 바꾸는 것에 의해 용이하게 다른 데이타폭에 대응할 수 있다. 그 구체적인 예를 제15도에 도시하였다. 제15도는 각 비트폭의 조합에 있어서의 하드웨어 구성의 구체적인 수치를 나타내고 있다. 표의 가로방향은 플레쉬메모리의 데이타의 비트수이고, 4비트, 8비트, 16비트, 32비트를 예를 들고 있다. 세로방향은 시스템의 데이타폭을 나타내고 있고, 간단한 정보처리장치의 8비트에서부터, 현재의 퍼스날 컴퓨터의 주류인 16비트, 32비트, 그리고 장래적인 퍼스널 컴류터나 고성능 컴퓨터의 64비트, 128비트의 예를 나타내고 있다. 또 하드웨어 구성으로서 들고 있는 데이타 분배회로는(1개의 분배회로가 분배하는 데이타 비트수)×(분배회로의 구성수)를 나타내고 있고, 실시예의 데이타 분배회로(61)로 설명하면, 도면중의 원 1개가 1개의 분배회로가 분배하는 데이타를 나타내고, 원의 수가 분배회로의 구성수를 나타내고 있다. 각 분배회로의 접속은 제12도~제14도와 동일한 개념으로 나타낼 수 있다. 래치수는 실시예에 있어서의 래치 (51)~(54)의 수의 증감이지만, 시스템 데이타폭보다 메모리데이타 비트수가 큰 경우에는 래치의 위치는 데이타 분배회로의 메모리측에 구성되게 된다. 메모리군수는 실시예에서는 4개로 분할되어 있는 메모리의 분할수를 나타내고 있다.

단, 본 실시예를 포함하여 데이타 분배회로의 분배수나 래치수, 메모리군수는 최저 한도의 수이고, 회로규모, 단자수등에 여유가 있으면, 증가시키는 것에 의해 더욱 효과적인 시스템 구성으로 된다.

이러한 것은 메모리군수를 증가시키는 것에 의해 데이타의 저장메모리의 선택에 여유가 생기므로, 상술한 바와 같은 메모리의 사용량, 사용빈도의 치우침을 더욱 방지할 수 있기 때문이다. 또, 병렬 액세스하는 메모리수를 증가시킬 수 있으므로, 고속화를 더욱 촉진시킬 수 있다. 특히 시스템 데이타폭과 메모리 데이타 비트수가 동일하거나 또는 메모리의 데이타 비트수가 많은 경우에는 실시예에서 기술한 바와 같은 데이타 버스폭을 맞추는 효과는 없지만, 여러개의 메모리칩을 동시에 액세스해서 소위 인터리브방식에 의한 고속화를 실현할 수 있다. 이때에는 메모리군수를 증가시키는 것에 의해 동시에 데이타 분배회로에 있어서의 메모리의 데이타의 분배수나 래치수를 동시에 증가시킬 필요가 있다.

또, 본 실시예에서는 플래쉬메모리가 시스템으로부터의 데이타전송에 추수할 수 있는 것으로서 설명하고 있지만, 그러기 위해서는 플래쉬메모리내에 라이트버퍼를 탑재하는 것이 바람직하다. 금후, 고속으로 라이트를 실행할 수 있는 플래쉬메모리가 실현되면, 이와 같은 라이트버퍼는 필요없게 된다. 플래쉬메모리에 라이트버퍼가 내장되어 있지 않은 경우에는 라이트버퍼를 플래쉬메모리와 데이타제어회로 사이에 설치할 수 있다. 그리고 라이트 액세스시에는 직접 플래쉬메모리에 라이트하지 않고 라이트버퍼에 라이트를 실행해서 시스템으로부터의 데이타전송 종료후에 라이트버퍼에서 플래쉬메모리로의 라이트를 실행하게 된다. 이 경우의 실시예의 구성도를 제 4 도에 도시하였다.

제 4 도중, 기출 번호는 지금까지 설명한 것과 동일한 것이다. (71)은 액세스가 리드인지 라이트인지에 따라 데이타 분배회로(61)과의 접속을 전환하는 데이타 셀렉타, (72)는 라이트데이타를 일시적으로 저장하는 CPU 버스폭과 같은 데이타 버스폭의 라이트버퍼, (73)은 데이타 셀렉터(71)을 전환하기 위한 리드라이트 신호이고, 커맨트 레지스터(25)에 라이트된 커맨드 코드에서 생성하면 좋다. 만일 액세스가 라이트이면 데이타 분배회로(61)을 라이트버퍼(72)에 접속하고, 라이트데이타를 라이트버퍼(72)에 저장해서 시스템으로부터의 데이타전송후, 컨트롤러가 플레쉬메모리어레이(5)로의 라이트를 실행한다. 리드이면 데이타 분배회로(61)이 플레쉬메모리어레이(5)에 직접 접속되어 상술한 제 3 도의 리드액세스와 동일한 동작이 실행된다.

이상의 구성으로 하는 것에 의해, 라이트버퍼를 내장하고 있지 않고 라이트가 느린 플래쉬메모리를 사용해도 시스템에서 보면 고속인 액세스가 가능한게 된다.

다음에 화일관리의 실시예로서 제 5 도A, 제 5 도B를 예로 해서 설명한다. 지금까지는 하드웨어 구성의 실시예를 설명하였지만, 실제로 플래쉬메모리에 어떻게 저장하는가에 대해서 다음에 기술한다. 기본적으로 이 조작은 시스템의 CPU 및 시스템 프로그램과 플래쉬메모리 화일의 컨트롤러가 실행하게 되고, 소프트웨어에 의한 조작이 중심으로 된다. 제 5 도A, 제 5 도B에 의한 설명에서는 그 소프트웨어 자체의 내용이 아니고, 그 조작의 결과로서 플래쉬메모리의 각 메모리군의 단위용량 영역(1섹터분)이 총체적으로 어떻게 화일의 섹터에 할당되는지를 나타낸다. 화일의 각 섹터에 대해서 순차로 메모리군에 단위용량 영역이 확보된다. 4 이상의 연속 섹터에 대해서는 4섹터(예를 들면, 도면의 1-1~1-4)에 대해서 순차로 메모리군 1~4의 1단위용량 영역이 할당된다. 단, 실제로는 이들 각 섹터의 데이타는 4개의 단위용량 영역내로 분산해서 저장된다. 2섹터(예를 들면 도면의 1-5~1-6)에 대해서는 2개의 메모리군의 단위용량 영역이 할당된다. 이 경우도 각 섹터의 데이타는 2개의 단위용량 영역내로 분산해서 저장된다. 화일 1의 최후의 섹터(17)에는 단독의 메모리군의 단위용량 영역이 할당된다.

여기에서, 2섹터 및 4섹터의 분산저장 상태에 대해서 제10도 및 제11도에 의해 더욱 상세하게 설명한다. 제10도는 4바이트 동시 라이트의 예를 도시한 도면이고, 제11도는 2바이트 동시 라이트의 예를 도시한 도면이다. 상술한 바와 같이, 호스트의 시스템버스가 32비트, 1개의 메모리군의 데이타 액세스폭이 8비트로 된다.

제10도에 있어서, 메모리군 a~d로의 4섹터 동시 라이트의 예를 나타낸다. 도면중, (221)은 호스트의 시스템버스로부터의 데이타를 나타내고, 32비트(4바이트)를 단위로 해서 전송되어 온다. (222)~(225)는 8비트(1바이트)단위의 데이타의 일련번호를 나타내고 있고, (222)는 1바이트째의 데이타, (223)은 2바이터째의 데이타, (224)는 3바이트째, (225)는 4바이트째의 데이타이다. 이하, 연속해서 4섹터분으로서 2048바이트째까지가 전송되어 온다. 실제로는 32비트버스이므로, 1바이트째부터 4바이트째까지가 동시에 수령되고, 이하도 마찬가지이다. (51)~(54)는 제 3 도에 도시한 데이타 일시기억을 위한 데이타래치이다.

4바이트씩의 동시 라이트이므로, 32비트 데이타 즉 4바이트를 동시에 데이타래치(51)~(54)에 저장한 후, 각 바이트가 메모리군 a~d로 분산해서 1바이트씩 라이트된다. 다음의 32비트 데이타에 대해서 각 바이트가 메모리군 a~d의 후속부분에 라이트된다. 이와 같이해서 1섹터분의 512바이트째까지가 라이트되면, 계속해서 제 2 의 섹터에 대해서 마찬가지로 4바이트씩 메모리군 a~d의 후속부분에 라이트된다. 제 5 도에서는 편의상 각 섹터를 특정한 메모리군으로 할당하도록 도시하였지만, 이상 설명한 바와 같이 실제로는 각 섹터의 데이타는 특정한 메모리군에만 저장되는 것은 아니고, 메모리군 a~d로 분산해서 라이트된다. 단, 4섹터 전체의 데이타로서는 메모리군 a~d의 4섹터분의 영역에 저장되게 된다.

제11도에 있어서는 메모리군 a와 메모리군 b로의 2바이트 동시 라이트의 예를 나타낸다.

2바이트씩의 동시 라이트이므로, 32비트 데이타 즉 4바이트를 동시에 데이타래치(51)~(54)에 저장한 후, 이것이 2바이트씩으로 이분되어 전반의 2바이트가 우선 메모리군 a와 메모리군 b에 1바이트씩
라이트되고, 다음에 후반의 2바이트가 메모리군 a와 메모리군 b에 1바이트씩 라이트된다. 이와 같이해서 1섹터분의 512바이트째까지가 라이트되면 계속해서 제 2 의 섹터의 1바이트째부터 마찬가지로해서, 메모리군 a,b의후속부분에 라이트된다. 이 경우도 제 1 섹터가 특정한 메모리군에만 저장되는 것은 아니고, 메모리군 a,b로 분해해서 라이트되고, 마찬가지로 제 2 섹터도 제 1 섹터에 이어서 메모리군 a,b로 분산해서 라이트된다. 단일 섹터의 라이트에 대해서는 도시하지 않지만, 이 경우에는 동시에 데이타래치(51)~(54)의 각 바이트 데이타가 순차로 어느 1개의 메모리군에 저장된다. 이 경우에만 1섹터의 데이타는 분산되지 않고 특정한 메모리군내에 저장된다.

이상에서 알 수 있는 바와 같이, 중요한 것은 이상의 동작의 결과, 플래쉬메모리 소자의 최소 소거단위(여기에서는 1섹터 용량에 상당하는 512바이트)의 영역내에는 다른 화일 데이타가 혼재하지 않는 것이다. 이것이 보증되는 것에 의해 화일 리라이트시에 다른 화일의 데이타를 소거하는 일 없이, 또한 메모리의 기억 영역을 유효하게 이용하면서 여러개의 바이트 병렬 동시 라이트에 의해 고속의 화일 데이타 저장을 달성하는 것이 가능해진다.

상술한 바와 같이, 제 5 도A는 어느 정도의 화일의 라이트가 실행된 상태, 제 5 도B는 그후 어느 화일에 대해서는 갱신되고 또한 화일용량이 증가한 상태를 나타내고 있다. 제 5 도A에서는 저장되는 화일번호순, 섹터번호순으로 다른 메모리가 할당되도록 저장된다. 관리상 낭비가 없도록 원칙적으로서 채워 빈공간이 없도록 한다.

제 5 도B에 도시한 바와 같이, 메모리군에 빈공간이 생기지 않도록 화일의 섹터를 순차, 연속적으로 채워 저장하는 것으로 하면, 화일갱신에 의해 화일 용량이 증대한 경우에는 제 5 도A에 도시한 바와 같이 물리적인 저장위치로서는 연속적으로 저장할 수 없는 경우가 발생한다. 이와 같은 경우라도 메모리군의 순번으로서는 연속적으로 되도록 저장한다. 즉, 화일번호 4번의 화일을 예로 하면, 제 5 도A에서는 4-1~4-5까지의 5섹터가 저장되어 있지만, 제 5 도B에서 2섹터 추가되면, 4-1이 메모리군 3에서 개시되고 있으므로, 4-6, 4-7의 증가분은 메모리군 2,3의 영역이 확보되게 된다. 섹터수가 증가한 화일의 라이트시에는 4-5와 4-6은 2섹터 연속섹터로서 취급되고, 4-7은 단독 섹터로서 취급되게 된다. 즉, 추가 섹터에 대해서도 이미 저장된 섹터와 함께 연속 섹터로서의 취급을 받게 된다. 이들 화일관리의 저장과 리드를 원활하게 실행하기 위해 하드웨어 구성으로서 시스템으로부터의 액세스 섹터의 지정은 개시 섹터와 섹터수로 하고, 물리적인 저장위치의 파악은 화일 시스템 내부에서 실행해도 좋다. 이때는 제 2 도에 도시한 엔드 섹터 레지스터(24)는 불필요하다. 그리고 화일의 연쇄를 나타내는 정보를 메모리의 데이타 이외의 정보를 저장하는 용장영역이 있으면 그곳에, 없으면 다른 기억수단에 저장하는 것에 의해 화일의 개시번호가 지정되면 그것에 이어지는 섹터번호의 물리적인 위치가 모두 연쇄적으로 명확하게 되므로, 연속 액세스가 가능하게 된다.

제 6 도에는 연쇄정보의 예를 도시하고 있다. 도면중, (85)는 저장되어 있는 화일저장 데이타이고, 화일번호 4, 섹터번호 5의 데이타이다. (86)은 그것에 이어지는 화일번호 4, 섹터번호 6의 데이타이다. 그리고, (87)은 또 그것에 이어지는 화일번호 4, 섹터번호 7의 데이타이다. 단, 이들 데이타는 섹터번호를 순번으로 하고 있지만, 실제의 저장상태로서는 혼합되어 있는 경우가 있다. 즉, 연속 섹터 액세스에 의해 여러개의 칩을 동시에 액세스하는 것에 의해 데이타를 스크램블해서 저장하고 있는 경우가 있다. 단, 이

때에도 바이트마다 데이타를 저장하는 순번을 붙이는 것이 필요하므로, 섹터번호는 매우 중요하다. (88)은 화일저장테이타(85)가 저장되어 있는 메모리상의 물리적인위치를 나타내는 물리 어드레스이고, 좌측의 수자 '3'은 메모리군번호, 우측의 수자 '5'는 그 메모리군중의 어드레스이다. (89),(90)은 마찬가지로 화일저장데이타(86), 화일저장데이타(87)의 물리 어드레스이다. (91)은 화일저장데이타(85)의 다음의 섹터가 저장되어 있는 물리 어드레스를 나타내는 연쇄정보이고, 화일저장데이타(85)가 화일번호 4, 섹터번호 5의 데이타이므로, 연쇄정보(91)은 화일번호 4, 섹터번호 6의 데이타가 저장되어 있는 물리 어드레스를 나타내게 된다. 좌측의 수자가 메모리군번호, 우측이 메모리군중의 어드레스이다. 마찬가지로 해서 연쇄정보(92)는 화일번호 4, 섹터번호 7의 데이타가 저장되어 있는 물리 어드레스를 나타내고 있고, 연쇄정보(93)은 다음의 섹터가 존재하지 않고 이 경우를 나타내고 있다. 즉, 화일번호 4의 화일은 7섹터에서 종료하는 것을 알 수 있다.

이 연쇄정보가 화일데이타에 첨부되어 있으면, 시스템의 CPU는 화일저장의 물리적인 위치를 파악할 필요는 없고 단지 화일 자체를 액세스하는 형식을 취할 수 있다. 그리고 화일제어의 컨트롤러가 연쇄정보를 참조하면서 메모리의 물리적인 액세스를 실행하는 것에 의해 1개의 화일이면서 물리적인 저장위치가 연속해 있지 않고 분산해 있는 화일이라도 연속적인 액세스가 가능해진다. 그때, 다른 메모리군의 동시 액세스를 위해 다른 어드레스를 입력하는 구성이 필요하다. 단, 제 9 도에서 상술한 바와 같이, 1섹터 액세스분의 하위 어드레스는 공용할 수 있고, 그것 이상의 상위 어드레스에 관한것만으로 충분하다. 이상, 연쇄정보를 사용하는 실시예에 의하면 시스템으로부터의 액세스 지정을 간략화할 수 있고, 적은 정보량으로 실시 가능하다.

다음에 메모리소자 자체에 각종 기능을 조립하는 실시예에 대해서 설명한다.

제 7 도는 메모리 어레이의 여러개의 메모리칩을 1개의 패키지에 수납하고, 그것에 부가회로를 마련한 메모리소자의 구성도이다. 도면중, (101)은 메모리소자, (102)~(105)는 동일한 기능을 갖고 있는 메모리칩, (106)~(109)는 메모리패키지(101)의 입출력 데아타단자로서, 각각 하나에 1개의 메모리칩의 데이타 입출력이 가능하다. 예를 들어 메모리칩(102)가 8비트의 데이타 입출력이면(106)~(109)까지 전부 8비트×4=32비트의 데이타 입출력단자를 갖게 된다. (110)은 이들의 메모리칩과 입출력단자를 접속하는 경로를 지정하는 신호입력단자, (111)은 지정신호, (112)는 집속경로 설정회로이다. 본 메모리소자(101)의 사용자는 데이타 입출력단자(106)~(109)의 설정을 신호입력단자(110)에 의해 실행하고 나서 액세스를 실행한다. 이 설정에는 상술한 제12도, 제13도, 제14도와 같은 접속경로에서 선택하는 것이 1예로서 고려된다. 메모리소자(101)에서는 설정된 데이타접속을 지정신호(111)로서 접속경로 설정회로(112)로 보내고, 접속경로 설정회로(112)에서는 설정에 따라서 메모리칩(102)~(105)를 데이타 입출력단자(106)~(109)에 접속하고, 바라는 데아타 버스에 메모리칩의 데이타를 접속하는 것을 실현한다.

제 8 도는 사용자로부터의 데이타접속의 설정을 커맨드 입력에 실행하는 메모리소자의 구성예이다. 도면중, (113)은 커맨드 설정값의 레지스터와 그 설정값에서 지정신호(111)을 만드는 커맨드 제어회로이다. (114)는 데이타 입출력단자의 일부를 사용해서 사용자가 커맨드 설정값을 설정하는 데이타선이다. 그 이외에는 제 7 도의 동일번호와 동일한 것이다.

사용자는 데이타의 접속경로를 제12도, 제13도, 제14도를 1예로서 선택하고, 커맨드코드로서 커맨드 제어회로(113)에 데이타 버스의 일부(114)에서 설정하고, 커맨드 제어회로(113)에서는 설정된 커맨드코드에서 접속경로 설정회로(112)에 해당하는 지정신호(111)을 보낸다. 이하는 제 7 도의 설명과 동일한다 제 7 도, 제 8 도에 의한 본 실시예에 의하면, 메모리소자로서 지금까지 설명한 다른 실시예를 실현한 수 있고, 주변회로의 삭감을 도모할 수 있는 효과가 있다. 또, 본 실시예에서는 메모리 패키지내에 여러개의 칩과 제어회로를 탑재하는 것으로 하고 있지만, 이들을 1개의 칩으로 모으는 것에 의해 소형화와 고속화를 도모할 수 있다.

(57) 청구의 범위

청구항 1

최소 소거단위가 화일기억장치의 데이타 버스폭보다 크고, 데이타 액세스폭이 화일기억장치의 데이타 버스폭보다 작은 기억소자군을 병렬로 사용한 화일기억장치로서, 1개 이상의 최소 기억단위로 이루어지는 화일데이타를 임의의 최소 기억단위의 조합으로 이루어지는 조합요소로 분할하는 화일분할수단, 데이타 버스상의 데이타를 데이타 액세스폭을 단위로 해서 임의로 조합하고, 데이타 액세스폭을 단위로 한 같은 단위의 기억소자군의 임의의 조합에 대응시키는 데이타 분배수단, 상기 조합요소의 각각을 화일기억장치에 저장할때 기억소자군의 상기 임의의 조합중의 어느것인가 1조에 대응시크도록 데이타 분배수단을 제어하는 제어수단을 구비한 화일기억장치.

청구항 2

최소 소거단위가 화일기억장치의 데이타 버스폭보다 크고, 데이타 액세스폭이 화일기억장치의 데이타 버스폭보다 작은 기억소자군을 병렬로 사용해서 데이타 버스를 거쳐서 화일데이타를 저장하는 화일기억장치로서, 1개 이상의 최소 기억단위로 이루어지는 화일데이타를 임의의 최소 기억단위의 조합으로 이루어지는 조합요소로 분할하는 스텝, 상기 조합요소로 분할하는 스텝, 상기 조합요소의 각각을 화일기억장치에 저장할때, 데이타 버스상의 데이타를 데이타 액세스을 단위로 해서 임의로 조합하고, 데이타 액세스폭을 단위로 한 같은 단위수의 기억소자군의 임의의 조합에 대응시키는 스텝 및 상기 조합요소의 각각을 기억소자군의 상기 임의의 조합중의 어느것인가 1조에 대응시키는 스텝을 갖는 화일저장방법.

청구항 3

최소 소거단위가 화일기억장치의 데이타 버스폭보다 크고, 데이타 액세스폭이 화일기억장치의 데이타 버스폭보다 작은 기억소자군을 병렬로 사용해서 데이타 버스를 거쳐서 화일데이타를 리드 라이트하는 화일기억장치로서, 1개 이상의 최소 기억단위로 이루어지는 화일데이타를 임의의 최소 기억단위의 조합으로 이루어지는 조합요소로 분할하는 화일분할수단, 데이타 버스상의 데이타를 데이타 액세스폭을 단위로 하

는 데이타의 임의의 조합으로 분할하는 데이타 분할수단, 기억소자군을 데이타 분할수단의 분할수에 맞춰 분할하는 기억소자 분할수단, 분할된 화일데이타의 각 조합요소를 화일기억장치에 저장할때, 데이타분할수단이 분할한 분할데이타의 각각을 기억소자 분할수단이 분할한 기억소자군중 어느것인가 1조에 대응시키고, 상기 조합요소의 각각의 내용을 기억소자군의 임의의 조합중의 어느것인가 1조에 저장하도록 제어하는 제어수단을 구비한 화일기억장치.

청구항 4

제 3 항에 있어서, 기억소자군중의 어느것인가를 특정하는 기억소자군정보와 각 기억소자군내에 특정한 최소 소거단위를 지정하는 어드레스정보로 이루어지는 물리 어드레스를 화일을 특정하는 화일특정정보와 화일내의 최소 기억단위를 특정하는 최소 기억단위 특정정보로 이루어지는 화일저장정보 및 동일 화일내의 다른 최소 기억단위의 물리 어드레스인 연쇄정보에 대응시킨 화일제어정보를 기억소자군 또는 다른 기억장치 내에 기억하는 화일기억장치.

청구항 5

제 3 항에 있어서, 화일분할수단의 분할내용, 데이타 분할수단의 분할내용, 기억소자군의 분할내용 및 제어수단의 제어내용을 화일마다의 화일제어정보로 해서 기억소자군내 또는 다른 기억장치수단내에 기억하는 화일기억장치.

청구항 6

최소 소거단위가 화일기억장치의 데이타 버스폭보다 크고, 데이타 액세스폭이 화일기억장치의 데이타 버스폭보다 작은 기억소자군을 병렬로 사용해서 데이타 버스를 거쳐서 화일데이타를 리드 라이트하는 화일기억장치로서, 1개 이상의 최소 기억단위로 이루어지는 화일데이타를 임의의 최소 기억단위의 조합으로 이루어지는 조합요소로 분할하는 화일분할수단, 데이타 버스상의 데이타를 데이타 액세스폭을 단위로 하는 데이타의 임의의 조합으로 분할하는 데이타 분할수단, 기억소자군을 데이타 분할수단에 맞춰 분할하는 기억소자 분할수단, 분할된 화일데이타의 각 조합요소를 화일기억장치에 저장할때, 데이타 분할수단이 분할한 분할데이타의 각각을 기억소자 분할수단이 분할한 기억소자군중 어느것인가 1조에 대응시키고, 상기 조합요소의 각각의 내용을 기억소자군의 임의의 조합중의 어느것인가 1조에 저정하도록 제어하는 제어수단, 화일분할수단을 분할내용, 데이타 분할수단의 분할내용, 기억소자 분할수단의 분할내용 및 제어수단의 제어내용을 화일마다의 화일제어정보로 해서 기억소자군내 또는 다른 기억장치수단내에 기억하는 기억제어수단 및 기억된 화일저장정보에 따라서 화일정보를 리드하는 기드수단을 구비한 화일기억장치.

청구항 7

최소 소거단위가 X비트보다 크고, 데이타 액세스폭이 Y비트(Y=X/P : P는 2 이상의 정수)인 기억소자군을 병렬로 사용해서 X비트의 데이타 버스를 거쳐서 화일데이타를 리드 라이트하는 화일기억장치로서, 1개 이상의 최소 기억단위로 이루어지는 화일을 1에서 P까지의 임의의 최소 기억단위의 조합으로 분할하는 화일분할수단, 화일분할수단에 의해 분할된 화일의 각 조합요소를 화일기억장치에 저장할때, 데이타 버스상의 X비트의 데이타를 Y비트를 단위로 하는 Q개(Q=X/Y, Q=1일 때는 분할하지 않는다)로 분할하는 데이타 분할수단, 기억소자군을 Q개로 분할하는 기억소자 분할수단, Q개로 분할된 분할데이타의 각각을 기억소자 분할수단이 분할한 어느것인가 1조에 대응시키고, 최소 기억단위의 임의의 조합요소의 내용이 같은 기억소자군에 대응되도록 제어해서 기억소자에 저장하는 제어수단, 화일분할수단의 분할내용, 데이타 분할수단의 분할내용, 기억소자군의 분할내용 및 제어수단의 제어내용을 화일마다 기억하는 기억수단 및 기억수단에 저장된 저장정보에 따라서 화일정보를 리드하는 리드수단을 구비하는 화일기억장치.

청구항 8

화일의 데이타를 X비트의 데이타 버스를 거쳐서 저장하고, 최소 소거단위가 X비트보다 크고 데이타 액세스폭이 Y비트(Y=X/P : P는 2 이상의 정수)인 플래쉬메모리 소자를 사용하고, 각각 동시에 액세스 가능한 P조는 플래쉬메모리 소자군으로 구성한 플래쉬메모리장치, X비트의 데이타 버스상의 데이타를 적어도 P분할하는 분할수단, 적어도 상기 분할에 의해 얻어진 P개의 Y비트 데이타를 플래쉬메모리 소자군의 1조에 대응시키는 제 1 의 기능과 P개의 Y비트 데이타를 각각 플래쉬메모리 소자군의 별개의 조에 대응시키는 제 2 의 기능을 갖는 데이타 분배수단, 액세스 대상 화일의 화일관리의 기억용량 단위의 수에 따라서 제 1 및 제 2 의 기능을 전환하도록 데이타 분배수단을 제어하는 제어수단을 구비한 플래쉬메모리 화일기억장치.

청구항 9

제 8 항에 있어서, 상기 제어수단은 상기 플래쉬메모리 소자의 최소 소거단위의 메모리 영역내에 다른 화일의 데이타가 혼재하지 않도록 상기 데이타 분배수단을 제어하는 플래쉬메모리 화일기억장치.

청구항 10

제 8 항에 있어서, 상기 기억용량단위와 상기 플래쉬메모리 소자의 최소 소거단위와 같은 플래쉬메모리 화일기억장치.

청구항 11

제 8 항에 있어서, 상기 기억용량단위와 상기 플래쉬메모리 소자의 소거단위가 같은 플래쉬메모리 화일기억장치.

청구항 12

프로그램 및 데이타를 처리하는 중앙연산 처리수단, 상기 중앙연산 처리수단을 구동하는 클럭발진수단, 플래쉬메모리를 기억매체로 한 화일기억수단, 상기 화일기억수단의 플레쉬메모리의 액세스를 제어하는 화

일기억수단, 상기 클럭발진수단이 발생하는 클럭신호와 동일한 신호 또는 동기하는 신호를 상기 화일기억 제어수단에 입력하고, 상기 중앙연산처리수단과 상기 화일기억 제어수단이 동기동작을 실행하여 화일 데이타를 수수하는 플래쉬메모리 화일기억장치를 탑재하는 정보처리장치.

청구항 13

제12항에 있어서, 상기 화일기억 제어수단은 상기 중앙연산 처리수단과 상기 화일기억수단의 1회의 데이타 액세스폭이 다른 경우, 화일 데이타의 처리중인지 처리종료인지를 나타내는 상태제시신호를 출력하는 수단을 구비하고, 상기 중앙연산 처리수단은 상기 상태제시신호를 받고, 이것이 처리중인 것을 나타내고 있으면, 처리종료까지의 기간, 처리의 진행을 정지하는 수단을 갖고, 상기 화일기억 제어수단은 상기 화일기억수단이 취급하는 데이타 비트수를 상기 중앙연산 처리장치의 처리데이타 비트폭에 맞추는 데이타 비트폭 제어수단을 갖고, 상기 데이타 비트폭 제어수단은 데이타 비트폭을 일치시키기 위해 필요로 하는 기간, 상기 상태제시신호에 의해 처리중인 것을 상기 중앙연산 처리수단에 표시해서 대기시키고, 쌍방의 처리데이타폭을 일치시켜 데이타의수수를 실행하는 정보처리장치.

청구항 14

제13항에 있어서, 상기 화일기억제어수단은, 상기 중앙연산 처리수단의 데이타 액세스폭과 같은 비트수의 데이타를 생성하시 위해 필요한 수의 플래쉬메모리군에 의해 상기 화일기억수단을 구성하고, 각각의 메모리군에 연속적인 순서를 규정하는 수단, 여러개의 기억용량 단위수에 이르는 화일에 대해서는 이 규정된 메모리군의 순서에 따라서 저장하는 수단 및 일단 저장된 화일을 갱신하는 것에 의해 기억용량 단위수가 증가하는 경우에는 상기 증가분에 대해서 전회에 저장했을때 화일의 최후의 데이타가 저장되어 있던 메모리군의 다음 순서에 규정되어 있는 메모리군에서 증가기억용량 단위분의 기억영역을 확보하는 수단을 구비한 정보처리장치.

청구항 15

중앙처리장치, 기억매체로서 여러개의 플래쉬메모리 소자를 사용한 화일기억장치를 구비한 정보처리장치로서, 상기 화일기억장치의 상기 플래쉬메모리의 데이타의 최소 소거단위를 화일관리의 기억용량 단위와 같게 하고, 또한 상기 중앙처리장치가 화일 액세스를 요구할 때의 기억용량 단위수가 여러개은 경우는 상기 여러개의 플래쉬메모리 소자를 동시에 액세스하고, 기억용량 단위수가 1인 경우에는 상기 여러개의 플래쉬메모리 소자의 1개를 액세스하는 정보처리장치.

청구항 16

플래쉬메모리를 기억매체로 하고, 기억하는 어느 1개의 화일의 기억용량이 여러개의 기억용량 단위에 이르는 경우에는 각 기억용량 단위마다 다음에 이어지는 기억용량 단위가 저장되어 있는 물리적인 위치의 정보를 기억하는 연쇄정보 기억수단을 구비한 플래쉬메모리 화일기억장치.

청구항 17

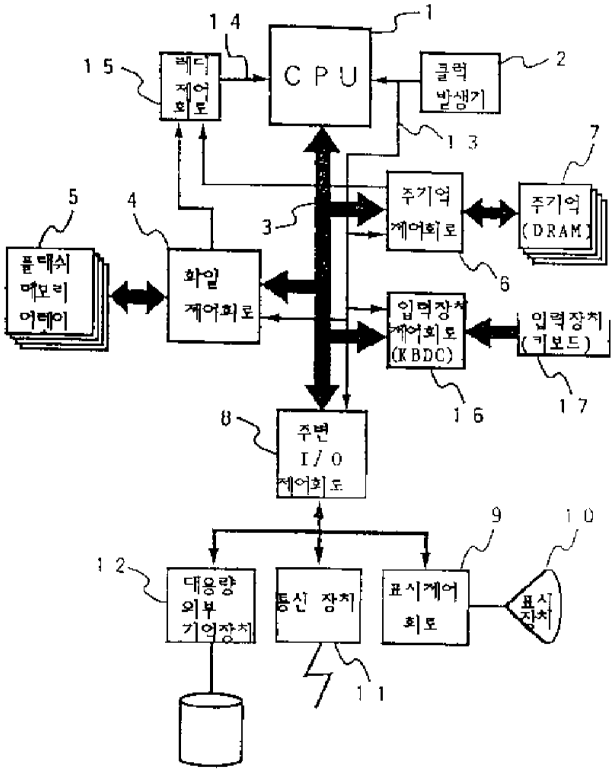
1개의 패키지에 여러개의 메모리칩이 조립되어 있고, 상기 패키지가 전체 메모리칩의 데이타 입출력수의 총계에 상당하는 수의 입출력 데이타단자, 상기 데이타 제어수단에 대한 메모리 사용자로부터의 지시를 위한 제어신호단자, 상기 제어신호단자의 지시에 의해 상기 여러개의 임의의 메모리칩의 입출력 데이타를 전환하고, 상기 입출력 데이타단자의 임의의 단자에 접속하는 데이타 제어수단을 구비한 메모리소자.

청구항 18

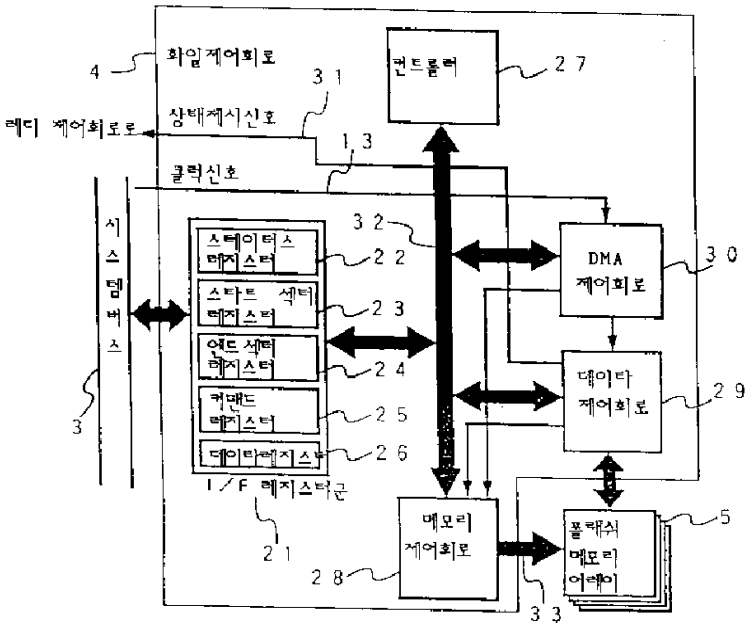
1개의 패키지에 여러개의 메모리칩이 조립되어 있고, 상기 패키지가 전체 메모리칩에 데이타 입출력수의 총계에 상당하는 수의 입출력 데이타단자, 상기 데이타 제어수단에 대한 메모리 사용자로부터의 지시를 위한 제어커맨드 설정수단, 상기 제어커맨드 설정수단의 지시에 의해 상기 여러개의 임의의 메모리칩의 입출력 데이타를 전환하고, 상기 입출력 데이타단자의 임의의 단자에 접속하는 데이타 제어수단을 구비한 메모리소자.

도면

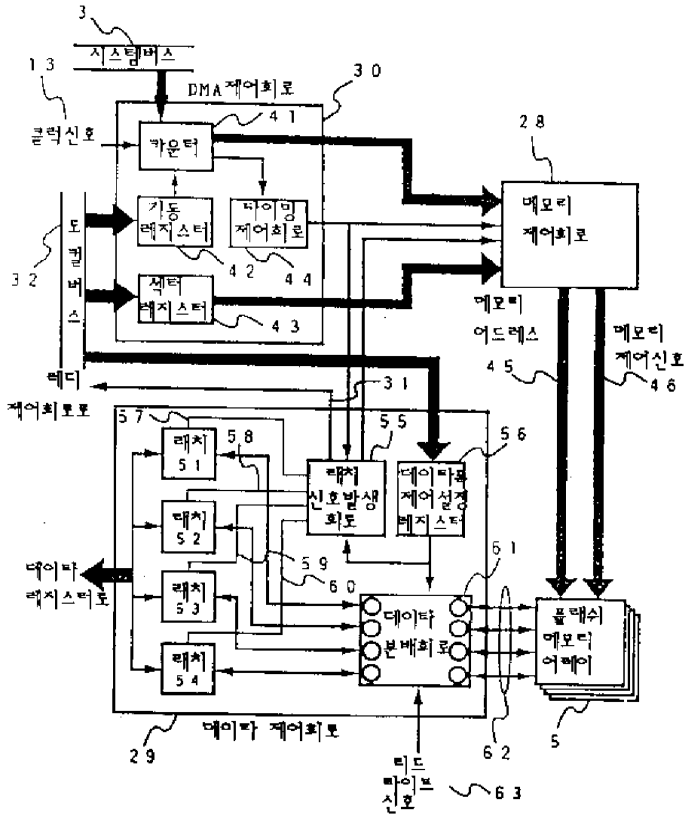
도면1


CPU
레디 제어 회로
클럭 발생기
주기억 제어회로
주기억 (DRAM)
플래쉬 메모리 어레이
화일 제어회로
입력장치 제어회로 (KBDC)
입력장치 (키보드)
주변 I/O 제어회로
대용량 외부 기억장치
통신 장치
표시제어 회로
표시 장치

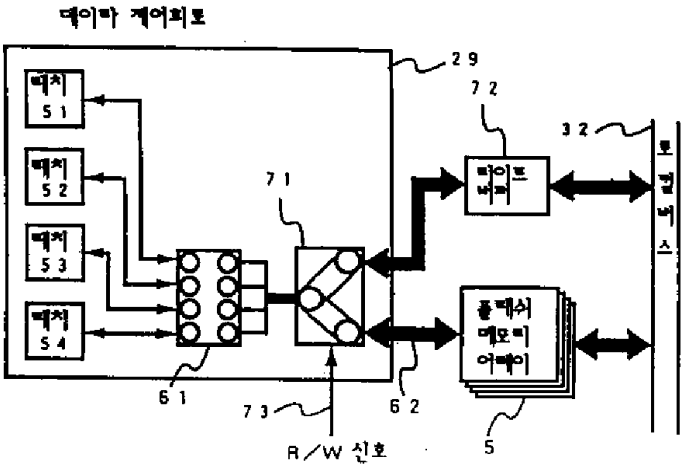
도면2


화일제어회로
레디 제어회로로
상태제시신호
클럭신호
시스템 버스
컨트롤러
스테이터스 레지스터
스타트 섹터 레지스터
엔드 섹터 레지스터
커맨드 레지스터
데이타레지스터
I/F 레지스터군
DMA 제어회로
데이타 제어회로
메모리 제어회로
플래쉬 메모리 어레이

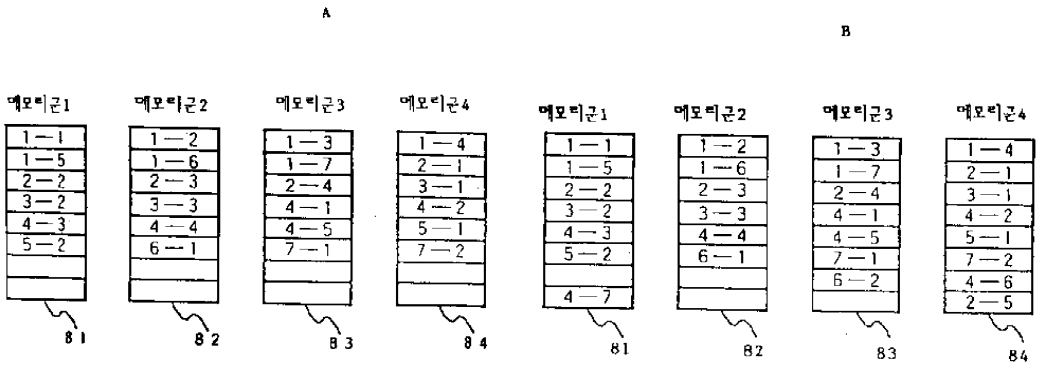
도면3


3
시스템버스
13
클럭신호
DMA제어회로
30
41
카운터
28
메모리
제어회로
기능
레지스터
42
타이밍
제어회로
44
32
로컬버스
섹터
레지스터
43
메모리
어드레스
45
메모리
제어신호
46
레디
제어회로로
31
57
래치
51
58
래치
52
래치
53
래치
54
래치
신호발생
회로
55
데이타폭
제어설정
레지스터
56
59
60
61
데이타
분배회로
플래쉬
메모리
어레이
5
데이타
레지스터
데이타 제어회로
29
62
리드
라이트
신호
63

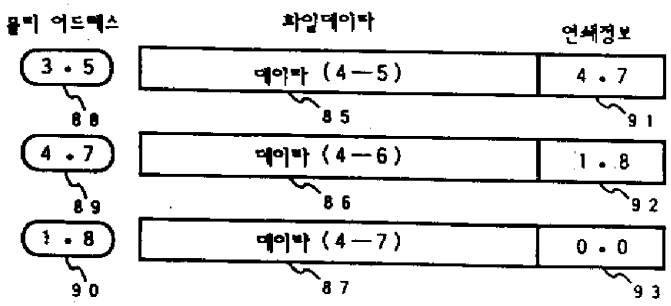
도면4


데이타 제어회로
29
래치
51
래치
52
래치
53
래치
54
72
32
71
라이트
버퍼
로컬버스
플래쉬
메모리
어레이
61
62
73
R/W 신호
5

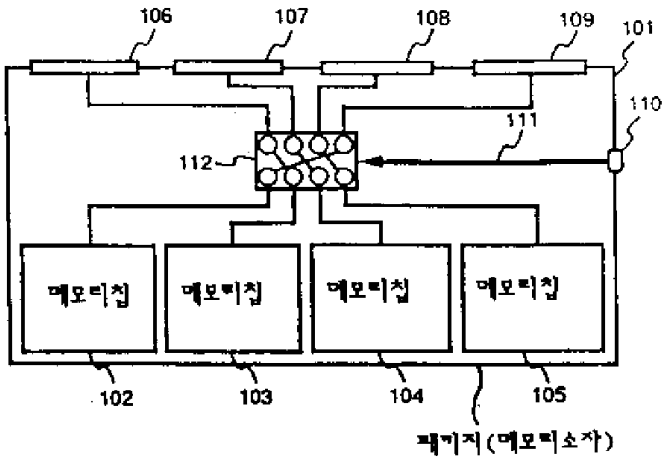
도면5


A
B
메모리군1
메모리군2
메모리군3
메모리군4
메모리군1
메모리군2
메모리군3
메모리군4
81
82
83
84
81
82
83
84

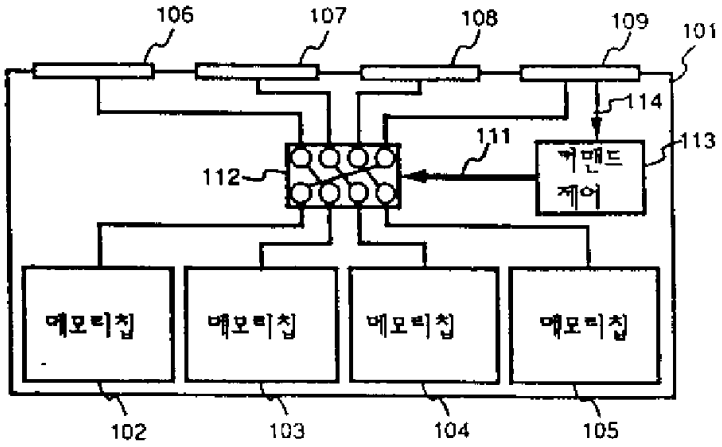
도면6


물리 어드레스
확장데이타
연쇄정보
3 • 5
데이타 (4—5)
4 • 7
88
85
91
4 • 7
데이타 (4—6)
1 • 8
89
86
92
1 • 8
데이타 (4—7)
0 • 0
90
87
93

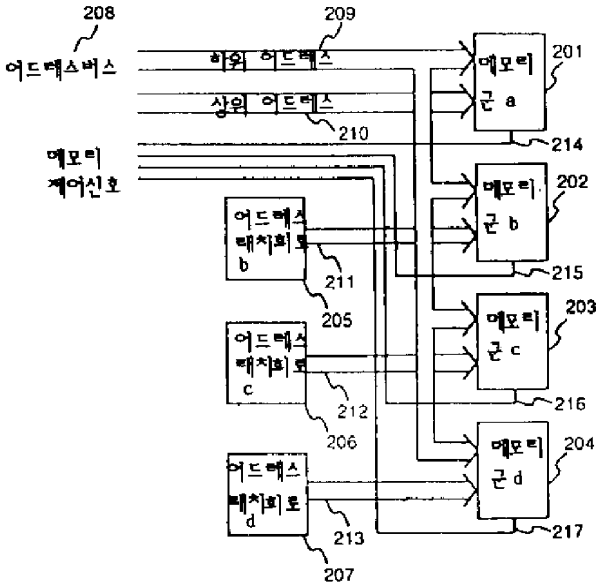
도면7


106
107
108
109
101
110
111
112
메모리칩
메모리칩
메모리칩
메모리칩
102
103
104
105
패키지(메모리소자)

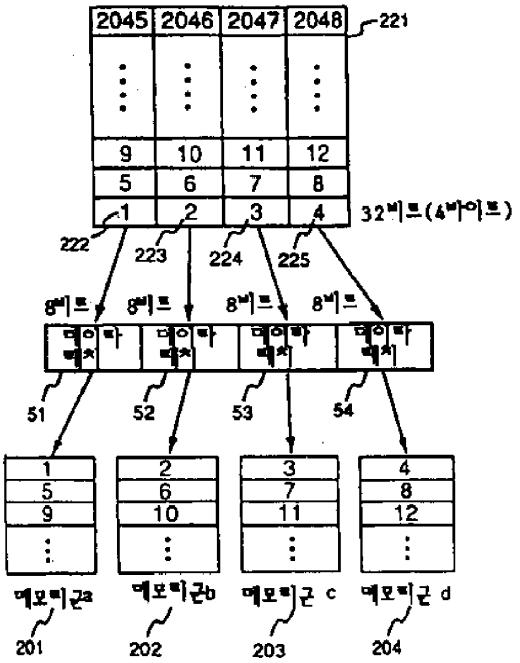
도면8


106
107
108
109
101
114
111
113
112
커맨드
제어
메모리칩
메모리칩
메모리칩
메모리칩
102
103
104
105

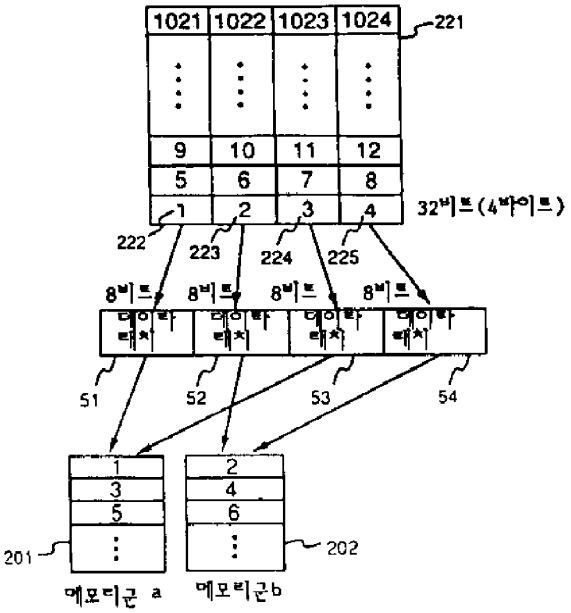
도면9


208
209
어드레스버스
하위 어드레스
상위 어드레스
210
메모리
제어신호
어드레스
패치회로
b
211
205
어드레스
패치회로
c
212
206
어드레스
패치회로
d
213
207
메모리
군 a
201
214
메모리
군 b
202
215
메모리
군 c
203
216
메모리
군 d
204
217

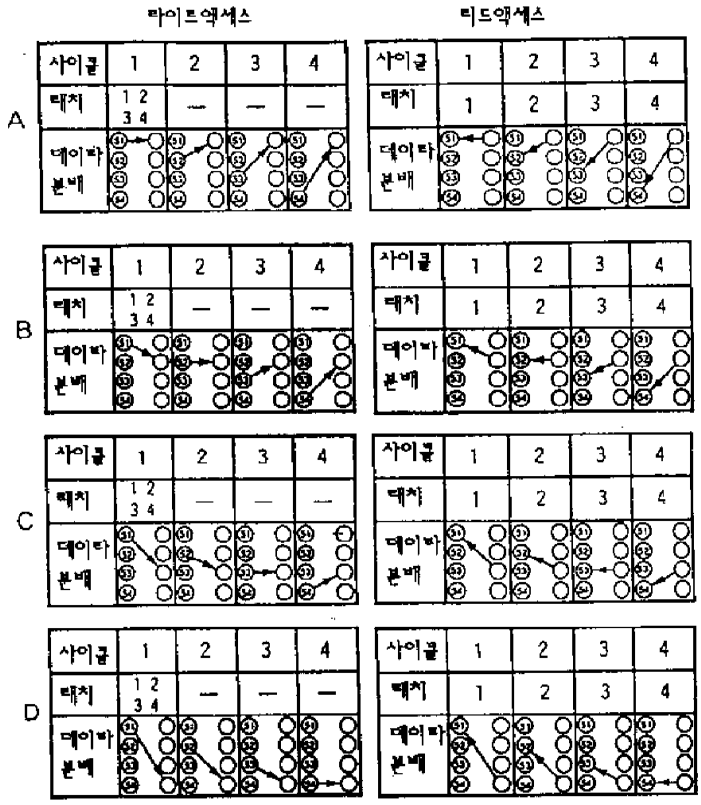
도면10


2045
2046
2047
2048
221
9
10
11
12
5
6
7
8
1
2
3
4
32비트(4바이트)
222
223
224
225
8비트
8비트
8비트
8비트
데이타 레지
데이타 레지
데이타 레지
데이타 레지
51
52
53
54
1
5
9
2
6
10
3
7
11
4
8
12
메모리군a
메모리군b
메모리군 c
메모리군 d
201
202
203
204

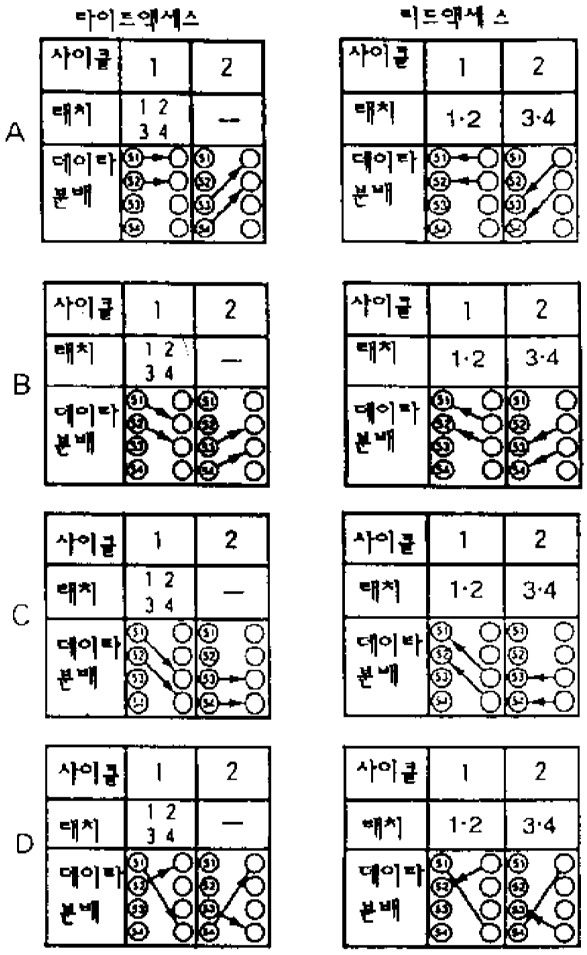
도면11


1021
1022
1023
1024
221
9
10
11
12
5
6
7
8
1
2
3
4
32비트(4바이트)
222
223
224
225
8비트
8비트
8비트
8비트
데이타 레지
데이타 레지
데이타 레지
데이타 레지
51
52
53
54
1
3
5
2
4
6
201
202
메모리군 a
메모리군b

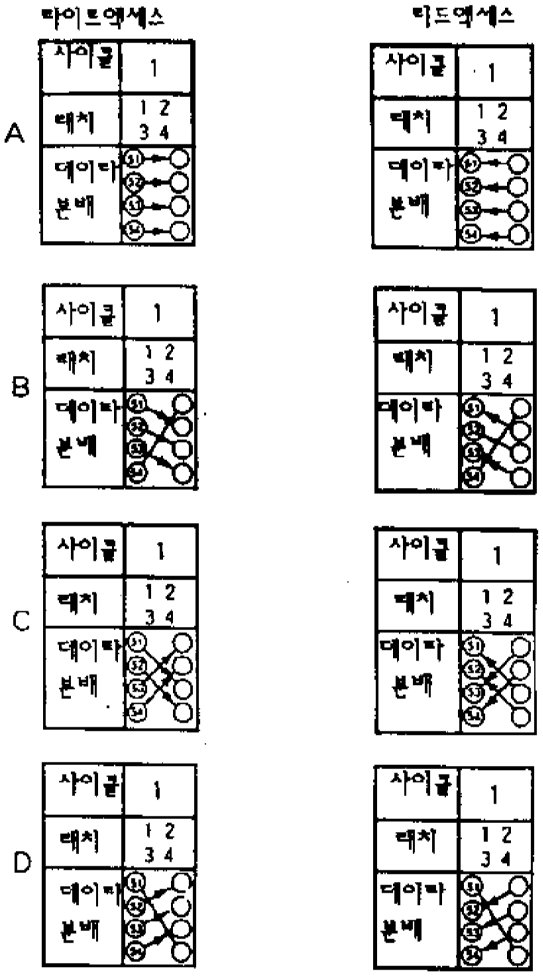
도면12



도면13



도면14



도면15

시스템 데이터 버스폭 / 메모리 데이터 버스폭 카드구성		4	8	16	32
8	데이타분배회로	4 bit× 2	8 bit× 1	8 bit× 2	8 bit× 4
	뱅크수	2	1	2	4
	메모리군수	2	1	1	1
16	데이타분배회로	4 bit× 4	8 bit× 2	16bit× 1	16bit× 2
	뱅크수	4	2	1	2
	메모리군수	4	2	1	1
32	데이타분배회로	4 bit× 8	8 bit× 4	16bit× 2	32bit× 1
	뱅크수	8	4	2	1
	메모리군수	8	4	2	1
64	데이타분배회로	4 bit×16	8 bit× 8	16bit× 4	32bit× 2
	뱅크수	16	8	4	2
	메모리군수	16	8	4	2
128	데이타분배회로	4 bit×32	8 bit×16	16bit× 8	32bit× 4
	뱅크수	32	16	8	4
	메모리군수	32	16	8	4