

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

(51) Int. Cl.<sup>6</sup>  
G06F 13/14

(45) 공고일자 1999년05월15일  
(11) 등록번호 10-0175980  
(24) 등록일자 1998년11월12일

---

(21) 출원번호	10-1995-0006997	(65) 공개번호	특 1995-0029955
(22) 출원일자	1995년03월30일	(43) 공개일자	1995년11월24일

---

(30) 우선권주장 8/222, 147 1994년04월01일 미국(US)

(73) 특허권자 인터내셔널 비지네스 머신즈 코포레이션 월리암 티. 엘리스  
미합중국 뉴욕 10504 아몬크

(72) 발명자 허만 리 블랙몬  
미합중국 미네소타 55906 로체스터 에이피티 #331 31 스트리트 엔이 402  
로버트 알렌 드레멜  
미합중국 미네소타 55027 구듀 배처 트래일 34420  
라일 에드윈 그로스바크  
미합중국 미네소타 55906 로체스터 태튼 코트 엔이 1644  
켄트 해롤드 하셀호스트  
미합중국 미네소타 55920 바이런 9 애비뉴 엔더블유 406  
데이비드 존 크로락  
미합중국 미네소타 55927 도지 센터 박스 57 알알 2  
제임스 안토니 마셀라  
미합중국 55901 로체스터 33 스트리트 엔더블유 820  
피더 제임스 파울슨  
미합중국 미네소타 55901 로체스터 3 스트리트 엔더블유 3731  
(74) 대리인 김창세, 김영, 장성구, 김원준

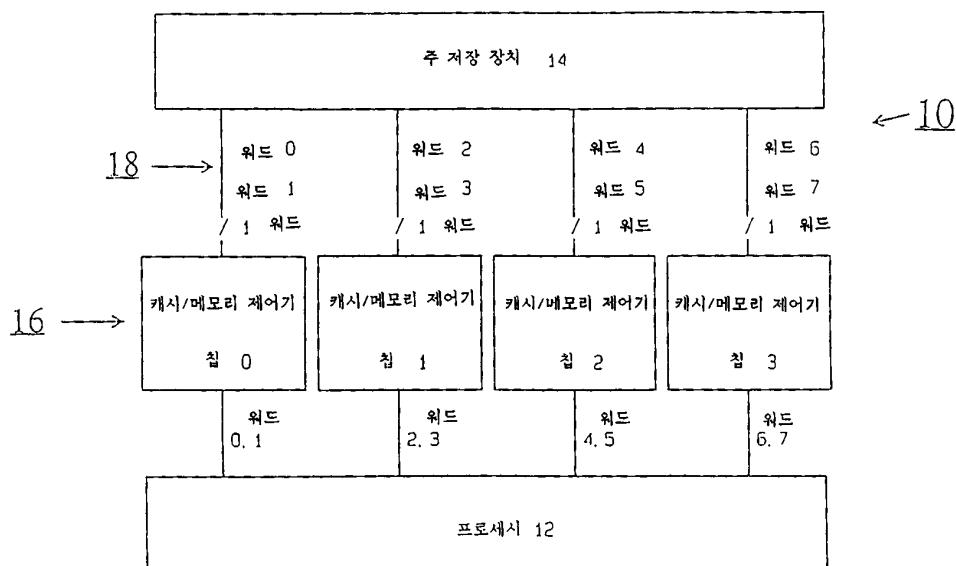
**심사관 : 오를수**

**(54) 공유버스의 비순차적인 데이터 순서화 방법 및 장치**

**요약**

본 발명은 공유 버스의 비순차적인 데이터 순서화 방법 및 장치에 관한 것이다. 최대 버스폭 값 및 최소 전송값이 식별된다. 식별된 최대 버스폭 값 및 최소 전송값에 응답하여 최소수의 서브전송이 식별된다. 데이터를 수신 및/또는 전신하기 위해 최대 개수의 칩을 구비한 버스 유닛은 다수의 서브전송 동안 사전 정의된 순서로 데이터를 수신한다. 각각의 데이터 서브전송 동안 대응하는 사전정의된 워드는 버스 유닛의 각각의 칩으로 전송된다.

## 대표도



## 명세서

### [발명의 명칭]

공유 버스의 비순차적인 데이터 순서화 방법 및 장치

### [도면의 간단한 설명]

제1도는 본 발명을 구현하는 컴퓨터 또는 데이터 처리 시스템을 도시한 블럭도.

제2도는 본 발명의 데이터 명명 규정을 예시한 블럭도.

제3도는 본 발명의 순서와 방법 및 장치에 따른 순차적인 단계를 예시하는 흐름도.

제3a도는 본 발명의 데이터 순서화 방법에 따른 순차적인 단계를 예시하는 흐름도.

제4도는 16바이트 주 저장 버스에 대한 바이트 넘버링 및 순서화를 예시한 도면.

제5도는 32바이트 주 저장 버스에 대한 바이트 넘버링 및 순서화를 예시한 도면.

제6도는 순차적인 데이터 순서화 버스 배열을 도시한 블럭도.

### \* 도면의 주요 부분에 대한 부호의 설명

10 : 컴퓨터 또는 데이터 처리 시스템

12 : 프로세서

14 : 주 저장장치

16 : 주 저장장치의 제어 유닛

18 : 데이터 버스

### [발명의 상세한 설명]

본 발명은 전반적으로 데이터 처리 시스템(data processing system)에 관한 것으로, 특히 데이터 처리 시스템의 공유 버스의 비순차적인 데이터를 순서화하는 방법 및 장치(shared bus non-sequential data ordering method and apparatus)에 관한 것이다.

전형적으로, 데이터 처리 혹은 컴퓨터 시스템은 버스로서 알려진 공유 데이터 경로(shared data bus)를 구비한다. 이 버스는 프로세서, 메모리 및 저장장치와 같은 버스 유닛(bus unit)을 접속시킨다. 데이터는 버스를 경유하여 버스유닛들 사이로 전송된다. 버스상의 데이터 전송 속도는 컴퓨터 시스템의 성능에 있어서 중요한 요소이다.

프로세서가 실행할 수 있는 인스트럭션의 수와 프로세서가 실행할 수 있는 속도가 증가함에 따라 버스상에서의 데이터 전송 속도를 증가시킬 필요성이 더욱 커졌다. 칩상에 트랜지스터를 백만 개 이상으로 집적 할 수 있는 집적 회로 기술이 출현함에 따라, 슈퍼스칼라(superscalar) 및 VLIW(very long instruction word) 프로세서를 제조하는 것이 가능하게 되었다. 이들 프로세서는 종종 한 사이클당 하나 이상의 인스트럭션(instruction)을 실행하므로, 메모리 혹은 저장장치로부터 다양한 데이터 전송이 이루어진다. 프로세서의 집적도 증가와 더불어 프로세서의 처리 속도는 메모리의 액세스 시간보다 훨씬 빠르게 증가하고 있

으며, 심지어 버스상의 신호의 이동 시간보다도 빠르게 증가하고 있다. 이러한 속도의 부정합(mismatch)으로 인해, 다량의 데이터 전송이 더욱 요구된다.

이와 같은 다량의 데이터 전송 요구를 충족시키기 위해서는 버스의 속도를 증가시켜야 한다. 공유 버스의 데이터 전송 속도는 버스의 물리적인 길이, 버스상의 부하(load) 및 최소 전송량(minimum transfer amount)(버스 폭(bus width) 또는 버스 크기(bus size)라고도 칭함)에 의하여 부분적으로 결정된다. 버스의 물리적인 길이는 단순히 버스 유닛들간의 접속 길이이다. 버스의 물리적인 길이가 길수록 버스의 속도는 느려질 것이다. 부하는 드라이버(driver)가 바라본 버스의 단일 와이어의 임피던스(impedance)이다. 부하가 클수록 버스의 속도는 느려질 것이다. 버스상의 부하는 버스상의 버스 유닛의 갯수와 각각의 버스 유닛에 의해 제공된 부하에 의해 결정된다. 최소 전송량은 버스를 통해 한번에 전송되는 데이터의 양이다. 최소 전송량이 클수록 데이터의 전송속도는 더욱 빨라질 것이다. 현대의 프로세서의 요구 조건들을 충족시키기 위해서는, 프로세서 및 프로세서의 캐쉬가 필요로 하는 버스상의 최소 데이터 전송의 크기를 4 또는 8 바이트에서 16 또는 32바이트 이상으로 바람직하게 증가시켜야 한다.

아쉽게도, 최소 전송의 크기는 물리적인 제한(physical constraints)에 의해 제한되기 때문에, 단순히 최소 전송 크기만을 증가시켜 버스 전송 속도를 증가시키는 것은 항상 가능한 것은 아니다. 이와 같은 물리적인 제한은 칩, 칩 모듈 및 카드 접속기상의 I/O 핀의 갯수와, 칩, 카드 및 각종 칩 모듈의 배선 제한(wiring constraints)과, 그리고 광폭 버스(wide bus)의 비용을 포함한다. 접적 회로는 그 접적도가 커지고 보다 많은 데이터 처리 능력을 제공함에 따라, 이들 접적회로는 모듈과 카드의 데이터 처리 능력을 능가하고 있다. 비록 칩의 I/O 능력을 이용하는 기술이 존재한다고 할지라도, 이것은 비용이 너무 많이 소요되기 때문에 기존의 절약형 패키지 기술을 사용하는 것이 훨씬 경제적이다.

따라서, 물리적인 제한 및 비용상의 제한 때문에, 복잡한 컴퓨터 시스템은 다양한 크기를 갖는 다수의 버스를 가질 수 있다. 예를 들면, 캐쉬 데이터 버스는 주 저장 데이터 버스보다 폭이 넓을 수 있다. 상이한 크기의 두개의 버스가 접속될 때, 한 버스의 전송 크기는 다른 버스의 전송 크기로 변환되어야 한다. 이를 성취하기 위해, 큰 전송 크기에서 작은 전송 크기로 변환할 때, 둘 혹은 그 이상의 서브 전송(sub-transfer)을 행하여 최소 전송 크기를 구성해야 한다. 기존의 변환 방법은 각각의 바이트를 데이터 버스 상에서 이웃하는 바이트와 순차적인 순서로 존재하도록 하는 순차적인 데이터 순서화(sequential data ordering)방안을 이용하고 있다. 예를 들어, 큰 전송 크기에서 작은 전송 크기로 변환시킬 때, 전체 폭의 제1부분을 전송하고, 이어서 제2부분을 전송하고, 이어서 제3부분을 전송하는 등과 같은 과정으로 전체 폭의 전송을 변환이 완료될 때까지 수행한다. 미국 특허 제5,243,701 호에는 순차적인 데이터 순서화 방안을 이용하여 8 비트 또는 16 비트 버스상에서 동작하는 능력을 가진 메모리 서브시스템의 예가 개시되어 있다.

이와 같은 기존의 변환 방법은 다수의 칩이 버스와 인터페이스할 때 부하가 증가되어 성능이 저하되는 문제점을 가지고 있다. 버스의 속도를 증가시키기 위하여 폭넓은 최소 전송이 필요하게 됨에 따라, 버스에는 종종 다수의 칩이 인터페이스 된다. 이들 각각의 칩은 전송된 데이터의 일부를 수신할 수 있으며, 버스상에는 하나 이상의 칩 부하가 존재할 수 있다. 데이터 버스에 접속된 다수의 칩은 예를 들어, 네 개의 DRAM 제어기 칩을 들 수 있는데, 이와 같은 각각의 칩은 DRAM의 네 개의 바이트를 제어하므로써 모두 16 바이트의 최소 전송 크기를 제공한다. 다른 예로는 4개의 데이터 바이트에 대한 캐쉬를 포함하는 4개의 캐쉬 제어기 칩을 들 수 있는데, 이 칩도 16바이트의 최소 전송 크기를 또한 제공할 수 있다.

접적회로 및 프로세서 설계의 발전은 공유 버스의 설계자에게 새로운 문제점을 던져주고 있다. 최신 프로세서 및 이 프로세서에 의해 감소된 사이클 시간에서는 매우 넓은 최소 데이터 전송이 요구되므로, 공유 버스 설계자는 예전보다 적은 시간동안 더 많은 데이터를 전송할 수 있도록 설계해야 한다. 공유 버스 폭이 요구된 최소 전송보다 작으면, 요구된 데이터는 두개 이상의 서브 전송으로 전송되어야 한다. 프로세서의 사이클 시간 동안 버스의 동작이 유지되도록, 설계자는 또한 버스상의 부하를 줄여야 한다. 넓은 최소 전송이 필요해짐에 따라, 버스에는 종종 다수의 칩이 인터페이스될 것이다. 만약 설계자가 기존의 기술을 사용하여 필요로 하는 최소 전송의 크기를 실제의 버스 폭으로 변환하면, 버스에 가외의 부하(extra load)가 발생되므로, 속도 저하가 야기될 것이다. 본 발명에 따른 공유 버스의 비순차적인 데이터 순서화 방법 및 장치는 시스템의 요건 및 제한을 적절히 조화시켜 종래 기술보다 월등한 성능을 제공한다.

본 발명의 주 목적은 버스의 부하를 감소시키고 버스의 성능을 개선시켜 종래 기술의 장치가 갖는 많은 문제점을 극복한 공유 버스의 비순차적인 순서화 방법 및 장치를 제공하는 데 있다.

본 발명의 목적 및 장점들은 공유 버스의 비순차적인 데이터 순서화 방법 및 장치에 의해 달성된다. 첫째로, 최대 버스폭 값(maximum bus width value) 및 최소 전송값을 식별한다. 둘째, 식별된 최대 버스폭 값 및 최소 전송간에 응답하여 최소수의 서브 전송(sub-transfer)을 식별한다. 세째, 데이터를 수신 및/또는 전송하는 최대수의 칩을 가지는 버스 유닛을 식별한다. 마지막으로, 각각의 데이터 서브전송 동안, 대응하는 사전정의된 워드를 버스상에서 최대수의 칩을 갖는 것으로 식별된 버스 유닛의 각 칩으로/로부터 전송한다.

이하, 본 발명의 공유 버스의 비순차적인 데이터 순서화 방법 및 장치를 도면을 참조하여 설명한다.

제1도를 참조하면, 본 발명을 구현하는 컴퓨터 또는 데이터 처리 시스템이 참조번호(10)로 도시되어 있다. 컴퓨터 시스템(10)은 프로세서(12)와, 메모리 혹은 주 저장장치(14)와, 주 저장장치의 제어 유닛(main storage controller unit: MSCU)(16)을 포함한다. 제어 유닛(16)은 참조번호(18)로 표시된 비순차적인 데이터 순서화 방식으로 동작하는 주 저장장치의 양방향 공유 버스(shared main store bi-directional bus)를 구비한 다수의 캐시/메모리 제어기 칩(칩0, 칩1, 칩2, 칩3)을 포함한다. 제1도에 도시된 바와 같이, 공유된 주 저장장치의 양방향 버스(18)는 주저장장치(14)와 MSCU(16)사이에 접속된다. 주 저장 장치의 양방향 버스(18)는 MSCU(16)의 캐쉬/메모리 제어기 칩(칩0, 칩1, 칩2, 칩3)에 의해 공유된다. 공유된 주 저장장치의 양방향 버스(18)는 각각의 캐쉬/메모리 제어기 칩(칩0, 칩1, 칩2, 칩3)에 대해 1 워드 폭의 인터페이스를 갖는다.

본 발명의 특징에 따르는 공유 버스의 비순차적인 데이터 순서화 방법 및 장치를 이용하면, 제6도에 도시된 것과 같은 순차적으로 순서화된 버스 배열에 비해 다수의 칩의 소스 및 목적지(multiple chip source

and destination)의 버스 부하가 감소되고, 버스의 성능이 향상되고, I/O 카운트가 감소된다. I/O 카운트의 감소는 각각의 칩에 대한 단일 워드-폭 인터페이스로 인해 야기된다. 서브전송 동안 데이터를 각각의 칩에 순차적으로 전송하도록, 데이터를 수신 혹은 전송하기 위해 최대수의 칩을 가진 버스 유닛은 비순차적인 방법으로 순서화된다.

제1도에 예시된 바와 같이, 본 발명의 순서화 방안을 이용하면, 데이터 버스(18)의 각 부분은 버스 유닛마다 하나의 부하만을 갖는다. 제1도에서 버스 유닛은 주 저장장치(14)와 MSCU(16)이다. 제1데이터의 서브전송은 0, 2, 4, 6 워드를 포함한다. 제1데이터 서브전송 동안, 각각의 칩(칩 0, 칩 1, 칩 2, 칩 3)은 하나의 워드, 즉, 짹수 워드 0, 2, 4, 6을 얻는다. 제2 데이터 서브전송은 홀수 워드 1, 3, 5, 7을 포함한다. 제2데이터의 서브전송 동안, 각각의 칩(칩 0, 칩 1, 칩 2, 칩 3)은 하나의 워드, 즉, 홀수 워드 1, 3, 5, 7을 얻는다. 이 방법을 이용하면, 각각의 칩은 단일 워드-폭 인터페이스만을 필요로 한다. 이와 같이 단일 워드-폭 인터페이스를 이용하면 비용이 적게 소요되고, 패키지의 소형화가 가능하고, 칩상에 다른 기능을 위해 더욱 많은 공간을 확보할 수 있다.

제2도는 데이터 명명 규칙(data-naming convention)을 도시한 도면이고, 제3도는 본 발명의 공유 버스의 비순차적인 데이터 순서화 방법 및 장치에 따른 데이터 순서화의 예를 도시한 도면이다. 최소 데이터 전송폭 X은 캐쉬와 프로세서간의 인터페이스에 의해 결정될 수 있다. X/2개의 I/O핀이 이용가능하면, 보다 작은 버스 Y는 예를 들어, X/2 바이트로 표현된다. 제1도에는 네개의 캐쉬/메모리 제어기 칩(칩 0, 칩 1, 칩 2, 칩 3)이 도시되었으며, 이를 각각의 칩은 X/4 바이트의 데이터를 얻는다. 데이터 버스 Y는 원하는 최소 전송의 1/20이기 때문에, 전송을 완료하는데 두개의 서브전송(sub-transfers: 57)이 필요하다. X 바이트가 n개의 워드로 구성된다고 정의하자(여기서, 워드는 각 칩에 대한 서브전송의 크기이다). 예시된 예에 있어서,  $n = X/8$ 이다. 각각의 서브전송 동안, 각각의 캐쉬/메모리 제어기 칩(칩 0, 칩 1, 칩 2, 칩 3)은 전체 전송의 1/8을 공급한다. 왜냐하면, 각각의 칩은 X/4 바이트를 필요로 하지만 각각의 서브전송으로부터 절반만을 얻기 때문이다.

예를 들어, 제1도에 도시된 바와 같이, 논리적 수신 MSCU(logical receiving MSCU)(16)내의 각각의 수신 유닛(칩 0, 칩 1, 칩 2, 칩 3)은 비순차적인 들어, 원하는 데이터 순서화 방안의 다수의 전송을 통해 데이터를 수신한다. 예를 들어, 원하는 데이터 전송폭이 0, 1, 2, 3, 4, 5, 6, 7로 넘버링된 8개의 워드로 구성된 X 바이트라고 가정하면, 두개의 서브전송시 X 바이트의 워드 0, 2, 4, 6가 버스를 통해 전송되고, 이어서 워드 1, 3, 5, 7가 전송된다.

그 결과, 각각의 수신 유닛(칩 0, 칩 1, 칩 2, 칩 3)은 표준 순차적 순서(standard sequential order)로 데이터를 수신한다. 수신 유닛(칩 0)은 워드 0, 1를 수신하고, 수신 유닛(칩 1)은 워드 2, 3을 수신하고, 수신 유닛(칩 2)은 워드 4, 5를 수신하고, 수신 유닛(칩 3)은 워드 6, 7를 수신한다.

제3a도를 참조하면, 본 발명의 데이터 순서화 방법에 따라 서브전송의 크기 및 데이터 순서화를 결정하는 단계들이 순차적인 흐름도로 도시되어 있다. 수학식으로 표현된 비순차적인 순서화 방법은 다음과 같다. 먼저 블럭(300)에서, X는 최소 데이터 전송시의 바이트 수이고, i는 버스 유닛상의 최대 칩의 갯수이고, Y는 I/O 제한으로 인한 데이터 버스의 최대 크기인 것으로 가정한다.

이어서, 블럭(302)에서 필요한 서브전송의 수를 식별한다. j가 최소 전송에 필요한 서브전송의 수인 것으로 가정하면, j는 다음과 같이 정의된다.

$j = X/Y$ 이다(여기서, j는 정수). 만약 Y가 2의 몇수가 아니면 j가 정수가 되도록 반올림한다.

그 후, 블럭(304)에서 워드당 바이트 수인 정수값 n이 식별되며, 정수값 n은 다음과 같이 정의된다.

$n = X/(i * j)$ 이다(여기서, n은 정수이고, 2의 몇수인 것으로 가정한다).

다음 블럭(306)에서, 데이터 순서화 알고리즘은 다음과 수행된다.

```

do ST = 0 to j-1           /* 각각의 서브 전송에 대해 */
    do CN = 0 to i-1        /* 각각의 칩에 대해 */
        chip(CN) <= word {(CN * j) + ST}; /* 이 워드를 그 칩으로 */
    end;
end;

```

블럭(306)에서 do루프는 서브 전송당 특정 워드를 특정 칩에 지정하지만, 다른 데이터 순서화 방안도 가능하고 또한 요구될 수도 있음을 이해하여야 한다. 버스 유닛들간의 요구되는 I/O 핀 및 부하는 칩의 입/출력(I/O)카운트와 버스 사이클 시간을 최소화하기 위해 감소된다.

다음의 표 1은 제1도에 도시된 예의 데이터 순서화를 배열한 것이다(여기서,  $j = 20$ 이고,  $i = 40$ 이다).

**[표 1]**

CN (칩의 개수)	ST (서브 전송)	
	0	1
칩 0	워드 0	1
칩 1	워드 2	3
칩 2	워드 4	5
칩 3	워드 6	7

제4도 및 제5도는 128 바이트 캐시 라인 전송의 16바이트 주 저장 버스 및 32 바이트 주 저장 버스에 대해 32 바이트의 최소 데이터 전송값을 취하는 바이트 넘버링을 예시한 도면이다. 16 바이트 버스에 대해서는  $Y = X/2$ 이고, 32 바이트 버스에 대해서는  $Y = X$ 가 된다.

제6도를 참조하면, 순차적으로 순서화된 데이터 버스의 예가 도시되어 있다. 데이터 버스의 각각의 부분은 캐쉬/메모리 제어기 버스 유닛당 두개의 부하를 가짐에 유의하여야 한다. 제1 서브전송은 칩 0과 칩 1에 대한 워드 0-3을 포함한다. 칩 0과 칩 1은 제각기 두개의 워드, 즉, 워드 0, 1과 워드 2, 3을 얻는다. 제2 서브전송은 칩 2과 칩 3에 대한 워드 4-7를 포함하며, 칩 2과 칩 3은 제각기 워드 4, 5와 워드 6, 7를 얻는다. 데이터 버스는 네개의 칩을 갖는 각각의 버스 유닛에 대해서 두개의 부하를 가지며, 이는 버스의 성능을 저하시킨다. 또한, 각각의 칩은 두개의 워드-폭 인터페이스를 필요로 한다. 만약 시스템이 다중 프로세서 및 메모리 카드와 같이 네 개의 칩 인터페이스를 구비한 다수의 유닛을 갖는다면, 이러한 가외의 부하는 더욱 많아진다.

이상에서, 본 발명은 바람직한 실시예를 참조하여 상세히 설명하였지만, 청구범위에서 정의된 본 발명의 영역은 이러한 실시예로 한정되는 것은 아니다.

**(57) 청구의 범위****청구항 1**

데이터 처리 시스템(a data processing system)의 공유 버스의 비순차적인 데이터 순서화 방법(a shared bus non-sequential data ordering method)에 있어서, ① 상기 데이터 처리 시스템에 대한 최대 버스폭값(a maximum bus width value)을 식별하는 단계와, ② 상기 데이터 처리 시스템에 대해 사전정의된 바이트 수인 최소 데이터 전송값(a minimum data transfer value)을 식별하는 단계와, ③ 상기 식별된 최소 전송값 및 상기 최대 버스폭 값을 이용하여, 서브전송값(a sub-transfer value)을 계산하는 단계 - 상기 계산된 서브전송 값은 j로 표시되며, 상기 최소 데이터 전송값을 전송하는데 필요한 서브전송의 수이다 - 와, ④ 다수의 칩을 갖는 버스 유닛(a bus unit)를 식별하는 단계 -상기 다수의 칩 각각은 상기 데이터 처리 시스템에 대한 데이터를 수신하거나 혹은 전송하고, 워드 서브전송 크기(a word sub-transfer size)을 갖는다 - 와, ⑤ 데이터 서브전송 동안, 상기 최소 데이터 전송값의 비순차적인 워드를 순서화하고 - 각각의 데이터 서브전송 동안 상기 비순차적인 워드는 짹수 워드 또는 홀수 워드중 어느 하나를 포함함 -, 각각의 데이터 서브전송 동안 상기 순서화된 비순차적인 워드들을 제각기 하나의 워드씩 이에 대응하는 상기 버스 유닛의 상기 다수의 칩에 각각 전송하는 단계를 포함하는 공유 버스의 비순차적인 데이터 순서화 방법.

**청구항 2**

제1항에 있어서, 상기 최소 데이터 전송값을 식별하는 단계와, 상기 최대의 버스폭 값을 식별하는 단계와, 상기 식별된 최소 전송값 및 상기 최대 버스폭 값을 이용하여 서브전송값을 계산하는 단계는, X 바이트로 표현된 상기 최소 전송값을 식별하는 단계(여기서, X 바이트는 n 워드를 포함함)와, Y로 표현된 상기 최대 버스폭 값을 식별하는 단계(여기서, Y는 X 바이트 미만임)와,  $j=X/Y$ 로 표현된 상기 서브전송값을 계산하는 단계(여기서, X, Y, j는 정수)를 포함하는 공유 버스의 비순차적인 데이터 순서화 방법.

**청구항 3**

데이터 처리 시스템에 대해 최대 버스폭 값과 최소 전송값을 갖는 양방향의 공유 버스(a shared bi-directional bus)상에 데이터를 전송하기 위한 - 상기 최대 버스폭은 상기 최소 전송값보다 작음 - 데이터 순서화 방법에 있어서, ① 상기 데이터 처리 시스템에 대해 상기 최대 버스폭 값 및 상기 최소 전송값을 이용하여, 최소수의 서브전송을 계산하는 단계와, ② 다수의 칩을 구비한 버스 유닛을 식별하는 - 상기 다수의 칩 각각은 상기 데이터 처리 시스템에 대한 데이터를 수신 및/또는 전송하고, 워드 서브 전송 크기를 갖는다 - 단계와, ③ 각각의 데이터 서브전송 동안, 상기 계산된 최소 서브전송의 수를 이용하여 비순차적인 워드를 순서화하고, 상기 순서화된 비순차적인 워드의 사전정의된 그룹을 상기 버스 유닛의 각각의 칩에 하나의 워드씩 전송하는 단계 - 각각의 데이터 서브전송 동안 상기 순서화된 비순차적인 워드의 사전정의된 그룹은 짹수 워드 또는 홀수 워드중 어느 하나를 포함한다 - 를 포함하는 데이터 순서화

방법.

#### 청구항 4

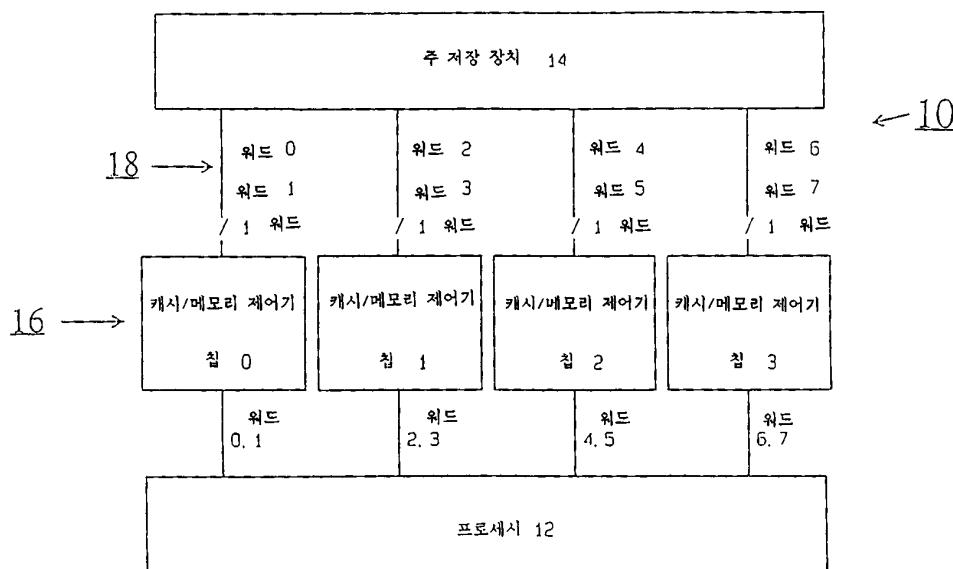
제3항에 있어서, 상기 최소 서브 전송의 수는 2이고, 상기 각각의 데이터 서브전송 동안 상기 순서화된 비순차적인 워드의 사전정의된 그룹을 상기 버스 유닛의 각각의 칩에 하나의 워드씩 전송하는 단계는, 제1데이터 서브전송 동안 짹수의 워드를 전송하는 단계와, 제2데이터 서브전송 동안 훌수의 워드를 전송하는 단계를 포함하는 데이터 순서화 방법.

#### 청구항 5

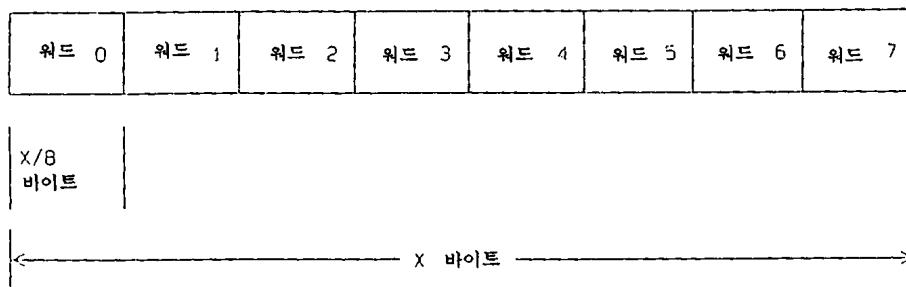
제3항에 있어서, 상기 각각의 데이터 서브전송 동안에 전송된 상기 순서화된 비순차적인 워드의 상기 사전정의된 그룹은 사전결정된 순서(a predetermined order)를 가지며, 상기 버스 유닛의 각각의 칩은 데이터 워드를 순차적인 순서로 수신하는 데이터 순서화 방법.

#### 도면

##### 도면1



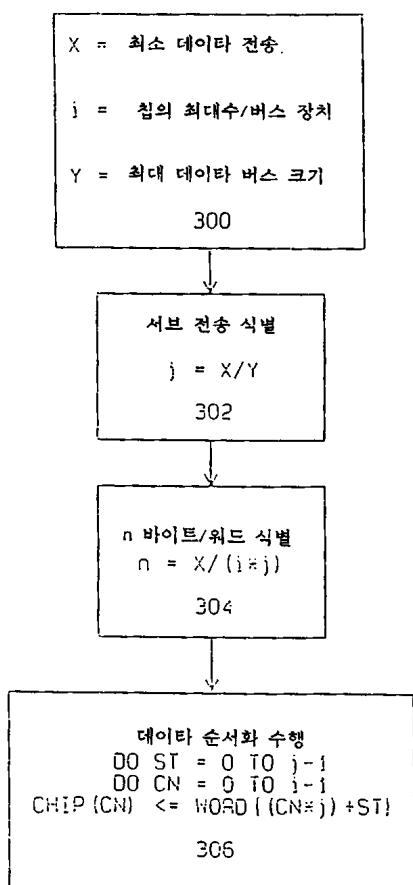
##### 도면2



## 도면3

	CN		ST	
		0	1	
칩	0	워드	0	1
칩	1	워드	2	3
칩	2	워드	4	5
칩	3	워드	6	7

## 도면3a



**도면4**

버스

전송 사이클

1	1	워드 0 (바이트 0-3)	워드 1 (바이트 0-11)	워드 4 (바이트 16-19)	워드 5 (바이트 20-27)
1	2	워드 1 (바이트 4-7)	워드 2 (바이트 12-15)	워드 5 (바이트 20-23)	워드 6 (바이트 20-31)
2	3	워드 8 (바이트 32-35)	워드 A (바이트 40-43)	워드 C (바이트 48-51)	워드 D (바이트 56-59)
2	4	워드 9 (바이트 36-39)	워드 B (바이트 44-47)	워드 D (바이트 52-55)	워드 E (바이트 60-63)
3	5	워드 10 (바이트 64-67)	워드 12 (바이트 72-75)	워드 14 (바이트 80-83)	워드 16 (바이트 98-91)
3	6	워드 11 (바이트 68-71)	워드 13 (바이트 76-79)	워드 15 (바이트 84-87)	워드 17 (바이트 92-95)
4	7	워드 18 (바이트 96-99)	워드 1A (바이트 104-107)	워드 1C (바이트 112-115)	워드 1E (바이트 120-123)
4	8	워드 19 (바이트 100-103)	워드 1B (바이트 108-111)	워드 1D (바이트 116-119)	워드 1F (바이트 124-127)

**도면5**

버스

전송 사이클

1	워드 0 (바이트 0-3)	워드 1 (바이트 4-7)	워드 2 (바이트 8-11)	워드 3 (바이트 12-15)	워드 4 (바이트 16-19)	워드 5 (바이트 20-23)	워드 6 (바이트 24-27)	워드 7 (바이트 28-31)
2	워드 8 (바이트 32-35)	워드 9 (바이트 36-39)	워드 A (바이트 40-43)	워드 B (바이트 44-47)	워드 C (바이트 48-51)	워드 D (바이트 56-59)	워드 E (바이트 60-63)	워드 F (바이트 64-67)
3	워드 10 (바이트 64-67)	워드 11 (바이트 68-71)	워드 12 (바이트 72-75)	워드 13 (바이트 76-79)	워드 14 (바이트 80-83)	워드 15 (바이트 84-87)	워드 16 (바이트 98-91)	워드 17 (바이트 92-95)
4	워드 18 (바이트 96-99)	워드 19 (바이트 100-103)	워드 1A (바이트 104-107)	워드 1B (바이트 108-111)	워드 1C (바이트 112-115)	워드 1D (바이트 116-119)	워드 1E (바이트 120-123)	워드 1F (바이트 124-127)

도면6

