

(19) 대한민국특허청(KR)  
(12) 특허공보(B1)

(51) Int. Cl.<sup>5</sup>  
G06F 15/16

(45) 공고일자 1994년03월 10일  
(11) 공고번호 특1994-0001877

(21) 출원번호	특 1986-0001849	(65) 공개번호	특 1986-0007597
(22) 출원일자	1986년03월 14일	(43) 공개일자	1986년 10월 15일
(30) 우선권 주장	85-51995 1985년03월 15일 일본(JP)		
(71) 출원인	소니 가부시끼가이샤 오오가 노리오 일본국 도오교도 시나가와구 기다시나가와 6쵸메 7반 35고		
(72) 발명자	호소노 요시마사 일본국 도오교도 시나가와구 기다시나가와 6쵸메 7반 35고 소니 가부시 끼가이샤 나이 우노 도시오 일본국 도오교도 시나가와구 기다시나가와 6쵸메 7반 35고 소니 가부시 끼가이샤 나이		
(74) 대리인	이병호, 최달용		

심사관 : 김연호 (책자공보 제3559호)

(54) 멀티 프로세서 시스템

요약

내용 없음.

대표도

도 1

명세서

[발명의 명칭]

멀티 프로세서 시스템

[도면의 간단한 설명]

제1도는 본 발명의 일실시예를 도시하는 블록도.

제2도는 16비트 CPU의 어드레스 공간상의 메모리 맵의 일예를 도시하는 도면.

제3도는 8비트 CPU의 메모리 소자 관리의 일예를 설명하기 위한 도면.

\* 도면의 주요부분에 대한 부호의 설명

- |                  |                       |
|------------------|-----------------------|
| 1 : 16비트 CPU     | 2 : 8비트 CPU           |
| 3 : 제어신호 변환회로    | 6 : RAM               |
| 10 : 어드레스 데이터 버스 | 11 : 16비트용 콘트롤 버스     |
| 12 : 8비트용 콘트롤 버스 | 100 : 8비트 CPU용 RAM 영역 |
| 101 : 창          |                       |

[발명의 상세한 설명]

본 발명은, 2 이상의 프로세서를 구비하여 형성되는 멀티 프로세서 시스템에 관한 것이다.

본 발명은, 어드레스 공간의 크기가 서로 다른 2 이상의 프로세서를 사용해서, 어드레스 공간의 큰 제1의 프로세서에 의해 액세스 가능한 기억수단의 일부를 제2의 프로세서에 의해서도 액세스 가능하도록 하므로서, 메모리 사용 효율의 향상을 도모함과 동시에, 제1의 프로세서의 어드레스 공간내에 제2의 프로세서의 메모리 공간의 창을 설치해, 이 창을 거쳐서, 제2의 프로세서에 의해 관리되는 메모리 디바이스를 제1의 프로세서에 의해서도 관리할 수 있도록 하므로서, 메모리 디바이스의 이용

효율을 높이는 것이다.

복수의 프로세서(말하자면 CPU)를 사용한 멀티 프로세서 시스템에는, 여러 종류의 구성의 것이 알려져 있으나, 이와 같은 멀티 프로세서 시스템의 많은 것은, 성능 혹은 기능이 동등한 복수의 프로세서를 사용한 것이다.

그런데, 예컨대 8비트 CPU의 컴퓨터 시스템에 있어서 개발된 많은 소프트웨어를, 보다 기능적으로 향상한 16비트 CPU 시스템에 있어서도 이용할 수 있도록 하고 싶은 경우등에는 호환성을 높이기 위해서, 각 기능, 특히 어드레스 공간의 크기가 대폭으로 다른 8비트 CPU와 16비트 CPU를 구비한 멀티 프로세서 시스템을 구성하는 것이 필요해진다.

이와 같은 예컨대 8비트 CPU와 16비트 CPU와 같이 성능이나 어드레스 공간이 대폭으로 다른 복수의 프로세서를 사용해서, 멀티 프로세서 시스템을 구성하는 경우에 있어서, 각 프로세서가 공통의 주기억수단(메인 메모리)을 액세스 하도록 구성하면, 이 메인 메모리의 용량이 어드레스 공간의 적은 8비트 CPU에 의해 제한을 받기 때문에 16비트 CPU의 우수한 기능을 충분히 발휘할 수 없는 결점이 있다. 또한, 각 CPU에 각각 독립의 메인 메모리를 설치한 경우에 있어서는, 메모리 사용 효율이 나쁘고 불경제적이다.

또다시, 상기 8비트 CPU 시스템이, 예컨대 말하자면 메모리 뱅크의 전환(뱅킹)방법등에 의해, 어드레스 공간에 대응하는 메모리 용량 이상의 메모리 디바이스를 관리하고 있는 경우에 있어서, 이들의 메모리 디바이스를 16비트 CPU에 의해서도 관리하려면, 각 메모리 디바이스로의 인터페이스등이 필요해져서 구성이 복잡할 뿐만 아니라, 각 메모리 디바이스마다 어드레스 공간의 배당이 필요해져서, 16비트 CPU의 어드레스 공간이 침식되어서 좁아진다는 결점이 있다.

본 발명은, 이와 같은 실정을 감안하여, 어드레스 공간의 서로 다른 복수의 프로세서를 사용하는 시스템에 있어서, 상위 기능의 프로세서의 어드레스 공간에 제한을 가하는 일 없이 기억수단(메모리)의 공용화가 행해질 뿐만 아니라, 하위 기능의 프로세서가 관리하는 메모리 디바이스를 상위 기능의 프로세서에 의해서도 간단한 구성으로 관리할 수 있도록 함과 동시에 어드레스 공간의 협력화도 방지하고, 하위 기능의 프로세서 시스템에서 개발되고 시장에서 공급되고 있는 다종의 소프트웨어를, 높은 호환성 아래에 이용가능하게 함과 동시에, 상위 기능의 프로세서 시스템의 기능을 저하시키는 일이 없는 멀티 프로세서 시스템의 제공을 목적으로 한다.

본 발명의 멀티 프로세서 시스템의 특징은, 제1의 어드레스 공간을 갖는 제1의 프로세서와, 상기 제1의 어드레스 공간보다도 좁은 제2의 어드레스 공간을 갖는 제2의 프로세서와, 상기 제1의 프로세서에 의해 액세스할 수 있는 기억수단과, 이 기억수단을 상기 제1의 프로세서에 의해 액세스하는 수단과, 상기 기억수단의 일부를 상기 제2의 프로세서용으로 맞추어, 상기 제2의 프로세서에 의해 상기 기억수단의 일부를 액세스하는 수단을 최소한 구비하는 것이다.

또한, 본 발명의 멀티 프로세서 시스템의 다른 특징은, 상기한 특징을 갖는 멀티 프로세서 시스템에 있어서, 상기한 제1의 어드레스 공간내에, 상기한 제2의 프로세서가 관리하는 메모리 디바이스용의 어드레스 공간에 대응하는 창을 설치하고, 상기 제1의 프로세서에 의해 상기 창을 거쳐서, 상기 메모리 디바이스를 관리하는 일이다.

제2의 프로세서는, 보다 큰 어드레스를 공간을 갖는 제1의 프로세서에 의해 액세스되는 기억 수단의 일부가 제2의 프로세서용의 기억 수단으로서 배당되고, 이것을 액세스한다. 또한, 제1의 프로세서는, 어드레스 공간내에 설치된 창을 거쳐서, 제2의 프로세서에 의해 관리되는 메모리 디바이스를 관리한다.

제1도는, 본 발명의 일 실시예를 도시하는 블록도이고, 예컨대 16M 바이트의 메모리 어드레스 공간을 갖는 16비트 CPU1과, 64K 바이트의 메모리 어드레스 공간을 갖는 8비트 CPU2를 사용한 멀티 프로세서 시스템의 예를 표시하고 있다. 이 경우의 8비트 CPU2는, 기존의 말하자면 8비트 퍼스널 컴퓨터 시스템으로서 널리 보급하고 있는 시스템 사양 혹은 규격을 만족하는 8비트 CPU 시스템을 구성하기 위한 것이며, 이들의 16비트 CPU1과 8비트 CPU2와는 전환으로 사용된다.

이 제1도에 도시하는 멀티 프로세서 시스템에 있어서, 어드레스, 데이터 버스 10는 각 CPU1, 2에 의해 공용되고 있으며, 예컨대, 16비트 CPU1의 24개의 어드레스 라인중의 16개가 8비트 CPU2의 어드레스 라인으로서도 사용되고, 16비트 CPU1의 16개의 데이터 라인중의 8개가 8비트 CPU2의 데이터 라인으로서도 사용하도록 되어 있다. 단, 각 버스라인이 어드레스의 상위, 하위 부분이나 데이터의 상위, 하위 부분등에서 시분할적으로 사용되는 경우등에는, 각 버스 라인의 개수나 공용 관계가 달라지는 것은 물론이다.

이에 대해서, 제어 신호버스, 말하자면 콘트롤 버스는, 각 CPU1, 2마다 각각 설치되어 있고, 16비트 CPU1에는 16비트용 콘트롤 버스 11가, 또는, 8비트 CPU2에는 8비트용 콘트롤 버스(12)가 각각 접속되어 있다. 그래서, 16비트용 콘트롤 버스(11)( 및 후기하는 바와 같이 어드레스, 데이터 버스 10의 일부)는, 제어신호 변환회로(3)를 거쳐서, 8비트용 콘트롤 버스(12)에 접속되어 있다.

여기에서, 각 CPU1, 2의 어드레스 공간은, 각각 예컨대 16K 바이트, 64K 바이트로 되어 있고, 16비트 CPU1의 어드레스 공간에 있어서 메모리 맵은, 예컨대 제2도와 같이 되어 있다.

이 제2도에 있어서, 기억수단으로서의 RAM(랜덤 액세스 메모리)은, 16진 표시된 어드레스의 "000000"번지에서 예컨대 256K 바이트 단위로 설치되고, 도면중에서는 4개의 256K 바이트 RAM이 "000000"번지에서 "0FFFF"번지까지의 1M 바이트 영역에 배설되어 있다. 이들의 RAM 영역중의, 예컨대 "020000"번지에서 "03FFFF"번지까지의 가운데 우수번지에 대응하는 64K 바이트의 영역(도면중 사선부의 영역) 100이, 상기 8비트 CPU2용의 RAM로서 배당되고, 이 CPU2에 의해 액세스할 수 있도록 되어 있다. 이것은, 16비트 CPU1의 1워드는 16비트이나, 메모리는 1바이트(8비트) 단위로 어드레스가 부여되어 액세스되도록 되어 있고, 1워드 2바이트중의 하위 바이트를 우수번지에 상위 바이트를 기수번지에 각각 분담하고 있음을 고려하여, 16비트 워드의 하위 바이트가 8비트 워드로 되도록, 8

비트 CPU에 의해 액세스되는 메모리 영역 100을 결정한 것이나, 이 외에, 예컨대 "030000"번지에서 "03FFFF"번지까지의 우, 기 양번지의 64K 바이트를 8비트 CPU2용으로 배당하여도 좋은 것은 물론이다.

또한, 제2도의 메모리 맵에 있어서, 상기 8비트 CPU용 RAM영역 100과는 별도로, 8비트 CPU2의 어드레스 공간에 해당하는 64K 바이트의 창(101)을, 예컨대 "F40000"번지에서 "F5FFFF"번지까지중의 우수번지에 대응하는 영역에 설치되어 있다. 그래서, 16비트 CPU1은, 이 창(101)을 거쳐서, 후기하는 바와 같이 8비트 CPU2가 관리하는 메모리 디바이스 등을 관리할 수가 있다. 이외에, 예컨대, "F60000"번지에서 "F7FFFF"번지까지의 우수번지의 64바이트의 영역에, 8비트 CPU용 어드레스 공간의 예비용으로 창(102)을 설치하고, 또한, "F80000"번지에서 "F9FFFF"번지까지의 사이에 우수번지의 64K 바이트의 영역에, 8비트 CPU용의 I/O 공간의 창(103)을 설치하고 있다.

또다시, 16비트 CPU1에 의해 액세스되는 16M 바이트의 어드레스 공간에는, 상기 하는 것 외에도, 예컨대 16비트 CPU용 I/O 영역(105)이나, 한자 ROM 영역(106)이나, 16비트 CPU 시스템의 OS(오퍼레이팅 시스템)용 ROM영역(107)이나, 비디오 RAM 이미지 영역(108)등이 설치되어져 있다.

이들의 한자 ROM 영역(105)이나 OS용 ROM 영역(107)에 대응하여, 제1도의 ROM(4)가 설치되어 있고, 이 ROM(4)는 어드레스, 데이터 버스(10) 및 16비트용 콘트롤 버스를(11)에 접속되어 있다. 또한, 각 CPU1, 2에 의해 공용되는 것으로서, 비디오 처리회로(5) 및 RAM(랜덤 액세스 메모리)(6)가, 각각 각 버스(10), (11), (12)에 접속되어 있다. 비디오 처리회로(5)에는, 상기한 영역(108)에 대응하는 비디오 RAM(7)가 접속되어 있다. 또한, 어드레스, 데이터 버스(10) 및 16비트용 콘트롤 버스를(11)에 다른 여러 종류의 기기를 접속하기 위한 16비트용 확장 스토트(8)가 설치되어 있다.

또한, 16비트 CPU1의 내부에는, 일반적으로, 상기 16M 바이트 어드레스 공간내에 배설되는 RAM6 등의 메모리 디바이스를 액세스하기 위한 MMU(메모리 관리 유닛)등이 설치되어 있다.

다음에, 8비트 CPU2에 대해서는, 예컨대 기존의 8비트 CPU 시스템의 사양이나 규격을 만족하는 하드웨어 구성 및 시스템 소프트웨어가 부가되어서, 기존의 8비트 CPU 시스템과 호환 가능한 시스템이 구성되어 있다. 즉, 8비트 CPU 시스템의 일부가 되는 어드레스, 데이터 버스(10) 및 8비트용 콘트롤 버스(12)에는, 8비트 시스템용 ROM(21)이나 8비트용 확장 스토트(22)가 설치되고, 키보드 콘트롤(23)을 거쳐서 키보드 입력장치(24)가 접속되어 있고, 또한, 프린터 인터페이스 회로(25)를 거쳐서 프린터 장치(26)가 접속되어 있다. 이외에, 8비트 CPU2에는, 어드레스, 데이터 버스(10) 및 콘트롤 버스(12)를 거쳐서, 프로그램 가능한 사운드 제네레이터 회로(27), 카렌다 타이머 회로(28), 데이터 전송용의 말하자면 RS232C 접속단자(29) 및 플로피 디스크 콘트롤 회로(30)등이 설치되어 있다. 상기한 확장 스토트(22)에는, 예컨대, 확장 RAM 카트리지, 껌이나 간이소프트 등의 ROM 카트리지, I/O(입출력)장치 코넥타 카드리지등이 삽입되어 접속된다.

또한, 비트 CPU2가 RAM(6)등을 액세스할때에는 제2도의 16M 바이트 어드레스 공간(24비트 어드레스)내의 64K 바이트의 메모리 영역(100)을 액세스 하기 위해서, 8비트 CPU2의 16비트의 어드레스를 원리적으로는 24비트 어드레스로 변환하는 것이 필요로 되어, 이 때문의 어드레스 디코더가 필요하다.

여기에서, 8비트 CPU 시스템에 대해서는, 말하자면 메모리 뱅크의 전환, 혹은 스토트 선택등의 뱅킹수법에 의해, 64K 바이트를 초과하는 메모리 디바이스를 관리하고 있다.

예컨대, 스토트에 의한 메모리 디바이스의 관리의 한계에 대해서 간단히 설명하면, 먼저, 8비트 CPU2의 64바이트의 어드레스 공간을 16K 바이트씩 잘라서 각각 페이지 0, 페이지 1, 페이지 2, 페이지 3으로 하고, 사용되는 메모리 디바이스 별로 전환할 수 있도록 함과 동시에, 전환용의 메모리 디바이스는 스토트를 선택함으로써, 지정되도록 하고 있다. 이 스토트로서는, 각각 64K 바이트의 기본 스토트를 예컨대 4개 설치하고, 이들의 각 기본 스토트는 또다시 각각 예컨대 4개의 장스토트에 확장되고, 결과로서, 각각 64K 바이트의 스토트를 최대한 16까지 확장한 경우에는, 메모리의 바이스를 1M 바이트까지 사용이 가능해진다.

이제 4개의 기본 스토트로서, 제3도에 도시하는 것과 같이 스토트 0, 스토트 1, 스토트 2, 스토트 3을 상정하는 경우, 어느 스토트의 어느 페이지를 선택하느냐 하는 것은, 예컨대 8비트 CPU 2의 입출력으로서의 PPI(프로그램머블 페리페럴 인터페이스)의 출력보트 A의 데이터에 의해 결정되고, 제3도의 예에서는, 스토트 0의 페이지 0, 1과 스토트 1의 페이지 3과, 스토트 3의 페이지 2가 선택되고 있다. 이들 가운데, 예컨대 스토트 0의 페이지 0, 1의 32K 바이트에는, 8비트 CPU 시스템용의 상기 ROM21이 배당되고, 스토트 1의 페이지 3의 16K 바이트에는 시스템용의 RAM로서 상기 RAM6내의 제2도 사선부 영역내의 16K 바이트분이 배당되는 등과 같은 사용형태로, 각종 메모리 디바이스를 페이지 전환하여 스토트 선택하여 사용하는 것이다.

16비트 CPU(1)는, 상기한 제2도의 64K 바이트의 창(101)을 거쳐서 이들의 메모리 디바이스를 액세스할 수 있고, 이때의 스토트 선택이나 페이지 전환을 위한 제어신호를 얻기 위해서, 제어신호 변환회로 3가 사용되는 것이다.

이 제어신호 변환회로(3)는, 16비트 CPU(1)의 콘트롤 버스(11)위의 신호뿐만 아니라, 도면에서는 생략하고 있으나, 각 CPU1, 2의 클럭 신호나, 어드레스 데이터 버스(10)상의 어드레스 신호의 일부등에 의하여, 8비트 CPU2측의 콘트롤 신호나 어드레스 신호의 일부등을 출력함과 동시에, 8비트 CPU 시스템측에서 16비트 CPU 시스템측으로의 각 메모리 디바이스에서의 응답신호의 변환등도 행하므로써, 각 메모리 디바이스에 대해서는, 8비트 CPU2가 관리하고 있을때와 같은 각종 제어신호의 송수를 16비트 CPU1 사이에서 행하는 것이다.

따라서, 8비트 CPU 시스템으로서의 기존의 8비트 퍼스널 컴퓨터 사양이나 규격에 대한 완전 호환성을 구비한 구성이 잡힘과 동시에, 16비트 CPU 시스템의 보다 우수한 기능을 하등의 저해하는 일 없는 멀티 프로세서 시스템을 실현할 수 있다. 이 경우, 8비트 CPU 시스템측에서는, 16비트 CPU 시

템의 기억수단인 RAM6의 일부를 사용할 수 있고, 8비트 CPU 시스템 전용의 RAM이 불필요해져서 경제적이다. 또한, 8비트 CPU 시스템에 의해 관리되는 메모리 디바이스는, 16비트 CPU 시스템의 어드레스 공간(제2도 참조)내의 창 101을 거쳐서 16비트 CPU1에 의해 관리할 수 있고, 이들의 메모리 디바이스는 양 CPU1, 2에 의해 공용되고, 이용 효율이 높아진다. 또다시, 8비트 CPU2측에서 뱅크 전환이 나 스로트 선택등의 뱅킹에 의해 대용량의 메모리 디바이스 관리를 행하고 있는 경우에, 16비트 CPU1는 각 메모리 디바이스마다 어드레스 공간내의 영역을 확보할 필요가 없고, 64K 바이트의 창 101을 거쳐서 모든 메모리 디바이스를 관리할 수가 있다.

또한 본 발명은, 상기한 실시예만에 한정되는 적은 아니고, 예컨대 사용되는 프로세서는, 16비트 CPU와 8비트 CPU에 한정되지 않고, 동일한 CPU일지라도 기능이나 어드레스 공간의 대폭적으로 다른 프로세서를 사용해도 되고, 또한, 3개 이상의 프로세서를 사용해도 좋다. 따라서, 상위의 프로세서의 어드레스 공간의 분배는, 제2도의 예에 한정되지 아니함은 물론이다.

본 발명에 따른 멀티 프로세서 시스템에 의하면, 서로 어드레스 공간이 다르고 기능도 대폭으로 다른 복수의 프로세서를 사용하므로써, 한쪽의 프로세서의 기능을 저하시키는 일없이, 다른쪽의 프로세서만이 사용한 시스템으로 개발된 기존의 소프트웨어의 이용이 가능해져서, 시스템의 완전 호환성을 확보할 수 있음과 동시에, 한쪽의 프로세서에 의해 액세스되는 기억 수단의 일부를 다른쪽의 프로세서에 의해 액세스가 가능하도록 하므로써, 메모리 사용효율이 높아진다. 또한, 한쪽의 프로세서는, 창을 거치므로써, 다른쪽의 프로세서가 관리하는 경우와 동일한 경로를 거쳐서 각 메모리 디바이스를 관리할 수 있고, 한쪽의 프로세서의 어드레스 공간을 좁히는 일이 없이 대메모리 용량을 관리할 수 있음과 동시에, 각 메모리 디바이스를 다른쪽의 프로세서와 공용이 가능하며, 경제적이다.

**(57) 청구의 범위**

**청구항 1**

제1어드레스 공간을 가진 제1프로세서 수단과, 상기 제1어드레스 공간보다 좁은 제2어드레스 공간을 가진 제2프로세서 수단과, 상기 제1프로세서 수단에 의해 액세스되기에 충분히 큰 메모리 용량을 갖고 있으며, 상기 제2프로세서 수단에 의해 액세스되도록 할당된 부분을 갖고 있는 메모리(기억)수단과, 그 라인의 일부가 제1 및 제2프로세서와 메모리 수단에 접속되어 있는 다중 라인 어드레스/데이터 버스와, 제1프로세서 수단과 메모리 수단에 접속된 제1제어버스와 어드레스/데이터 버스를 포함하며 상기 제1프로세서 수단에 의해 상기 메모리 수단에 대한 액세스를 제공하기 위한 제1액세스 수단과, 제2프로세서 수단과 메모리 수단에 접속된 제2제어버스와 어드레스/데이터 버스를 포함하며, 상기 제2프로세서 수단에 의해 상기 제2프로세서 수단에 할당된 메모리 수단 부분에 대한 액세스를 제공하기 위한 제2액세스 수단을 포함해서 이루어진 멀티-프로세서 시스템.

**청구항 2**

제1항에 있어서, 상기 제2액세스 수단이 상기 제2프로세서 수단에 의해 액세스 가능한 다수의 메모리 스로트를 관리하기 위한 스로트 선택 수단을 포함하고 있으며, 제2프로세서 수단에 액세스 가능한 메모리 수단 부분이 상기 다수의 메모리 스로트중 선택된 스로트가 되는 멀티-프로세서 시스템.

**청구항 3**

제2항에 있어서, 각각의 상기 메모리 스로트가 다수의 페이지 영역으로 분할되고, 각 메모리 스로트 내의 페이지 영역에 소정의 셋트가 상기 제2프로세서 수단으로부터의 요청에 따라 상기 스로트 선택 수단에 의해 선택되도록 되어 있는 멀티-프로세서 시스템.

**청구항 4**

제3항에 있어서, 상기 제2프로세서 수단이 상기 페이지 영역의 선택을 결정하기 위해 상기 요청을 출력시키기 위한 프로그램 가능 주변 인터페이스를 더 포함하고 있는 멀티-프로세서 시스템.

**청구항 5**

제1항에 있어서, 상기 메모리 수단이 제1제어 버스를 통해 상기 제1프로세서 수단으로부터의 제1제어 신호에 의해 제어되고, 제2프로세서 수단에 의해 액세스 가능한 메모리 수단의 부분은 제2제어 버스를 통해 상기 제2프로세서 수단으로부터의 제2제어 신호에 의해 제어되며, 상기 제1액세스 수단은 상기 제1프로세서 수단에 의한 메모리 수단의 상기 부분의 액세스를 허용하도록 상기 제1제어 신호를 상기 제2제어 신호로 변환하기 위해 상기 제1제어 bus와 제2제어 bus 사이에 접속된 제어 신호 변환 회로 수단을 포함하고 있는 멀티-프로세서 시스템.

**청구항 6**

제5항에 있어서, 상기 메모리 수단이 제1액세스 수단으로 하여금 상기 제2프로세서 수단에 의해 관리되는 다수의 메모리 스로트를 액세스할 수 있도록 제2프로세서 수단의 어드레스 공간에 대응하는 윈도우(창) 영역을 포함하고 있는 멀티-프로세서 시스템.

**청구항 7**

제6항에 있어서, 상기 윈도우 영역이 상기 제1프로세서의 수단의 상기 제1어드레스 공간에서 상기 제2어드레스 공간에 대한 대응하는 정보를 한정하도록 되어 있는 멀티-프로세서 시스템.

**청구항 8**

제1어드레스 공간 및 운용 시스템을 갖고 있는 제1프로세서 수단과, 상기 제1프로세서 수단과 운용 시스템을 제어하기 위해 상기 제1프로세서 수단과 접속된 제1리드 온리 메모리 수단과, 상기 제2프

로세서 수단의 운용 시스템을 제어하기 위해 상기 제2프로세서 수단과 접속되어 있는 제2리드 온리 메모리 수단과, 상기 제1프로세서 수단에 의해 액세스되기에 충분히 큰 메모리 용량을 갖고 있고 또한 상기 제2프로세서 수단에 의해 액세스되도록 할당된 부분을 갖고 있는 데이터 저장을 위한 랜덤 액세스 메모리 수단과, 제1 및 제2프로세서 랜덤 액세스 메모리 수단에 접속되어 있는 라인의 일부를 갖고 있는 다중 라인 어드레스/데이터 버스와, 제1프로세서 수단 및 랜덤 액세스 메모리 수단에 접속된 제1제어 버스 및 어드레스/데이터 버스를 포함하며 상기 제1프로세서 수단에 의해 상기 랜덤 액세스 메모리 수단을 액세스 하기 위한 제1액세스 수단과, 제2프로세서 수단과 랜덤 액세스 메모리 수단에 접속된 제2제어 버스 및 어드레스/데이터 버스를 포함하며 상기 제2프로세서 수단에 의해 상기 제2프로세서 수단에 할당된 상기 랜덤 액세스 메모리 수단의 일부분을 액세스 하기 위한 제2액세스 수단을 포함해서 이루어진 멀티 프로세서 시스템.

**청구항 9**

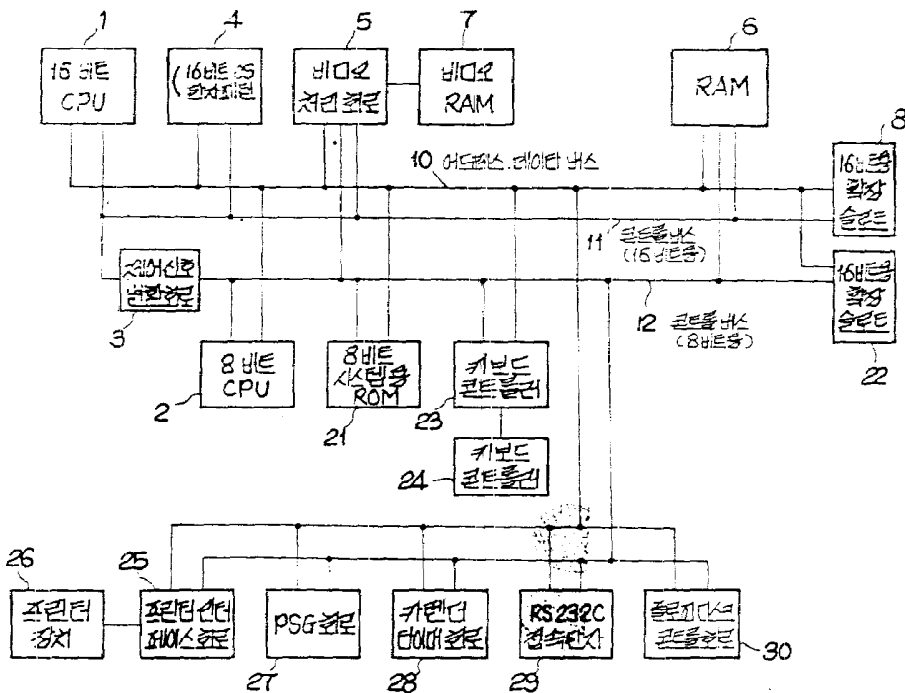
제8항에 있어서, 상기 랜덤 액세스 메모리 수단은 상기 제1프로세서 수단으로부터 제1제어 버스를 통해 전송되는 제1제어 신호에 의해 제어되고, 랜덤 액세스 메모리 수단의 상기 일부는 상기 제2프로세서 수단으로부터 제2제어 버스를 통해 전송되는 제2제어 신호에 의해 제어되며, 또한 상기 제1액세스 수단은 상기 제1프로세서 수단에 의한 랜덤 액세스 메모리 수단의 상기 부분의 액세스를 위해 상기 제1제어 신호를 상기 제2제어 신호로 변환시키기 위해 상기 제1제어버스와 제2제어 버스 사이에 접속된 제어신호 변환 회로 수단을 포함하고 있는 멀티-프로세서 시스템.

**청구항 10**

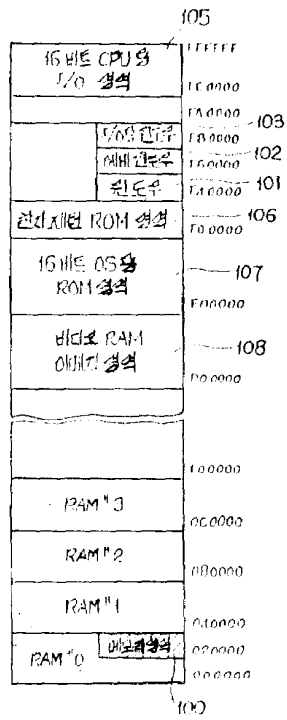
제9항에 있어서, 상기 제2리드 온리 메모리 수단과 랜덤 액세스 메모리 수단은 그 각각의 스토트가 상기 제2프로세서 수단에 의해 관리되는 다수의 메모리 스토트를 포함하고 있으며, 상기 제1액세스 수단은 상기 제2프로세서 수단에 의해 상기 다수의 메모리 스토트를 액세스 하기 위한 추가적인 수단을 포함하고 있는 멀티-프로세서 시스템.

**도면**

도면1



도면2



도면3

