(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl.
G06F 12/00 (2006.01)
G06F 12/02 (2006.01)

(11) 공개번호 10-2006-0133071
(43) 공개일자 2006년12월22일

(21) 출원번호 10-2006-7022716
(22) 출원일자 2006년10월30일
심사청구일자 2006년10월30일
번역문 제출일자 2006년10월30일
(86) 국제출원번호 PCT/US2005/009524
국제출원일자 2005년03월23일
(87) 국제공개번호 WO 2005/098629
국제공개일자 2005년10월20일

(30) 우선권주장 10/812,950 2004년03월29일 미국(US)

(71) 출원인 마이크론 테크놀로지 인코포레이티드
미국, 아이다호 83706-9632, 보이즈, 사우스 패더럴 웨이 8000

(72) 발명자 제델로, 조세프, 엠.
미국 미네소타 55126, 쇼뷰, 레일랜드 레인 4302

(74) 대리인 이범래

전체 청구항 수 : 총 42 항

(54) 메모리 시퀀싱 힌트들을 제공하기 위한 메모리 허브 및방법

(57) 요약

메모리 모듈은 몇 개의 메모리 디바이스들에 결합된 메모리 허브를 포함한다. 메모리 허브는 메모리 디바이스들의 후속 동작을 나타내는 메모리 힌트를 포함한 메모리 패킷을 시스템 제어기로부터 수신하도록 결합된다. 메모리 모듈은 다수의 페이지들이 개방되어 있게 하는 등의 메모리 모듈의 동작성을 조정하기 위해 힌트를 사용하거나 가져올 캐시 라인들을 사용한다.

대표도

도 5

특허청구의 범위

청구항 1.

메모리 모듈에 있어서,

복수의 메모리 디바이스들; 및

상기 메모리 디바이스들에 결합된 메모리 허브를 포함하고, 상기 메모리 허브는,

상기 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위한 메모리 요청들을 수신하는 링크 인터페이스로서, 상기 메모리 요청들 중 적어도 일부는 상기 메모리 디바이스들의 후속 동작에 관한 정보를 제공하는 각각의 메모리 힌트들을 포함하는, 상기 링크 인터페이스;

상기 메모리 디바이스들 및 상기 링크 인터페이스에 결합된 메모리 디바이스 인터페이스로서, 상기 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위해 메모리 디바이스들에 메모리 요청들을 결합하고 상기 메모리 요청들 중 적어도 일부에 응답하여 판독 데이터를 수신하도록 동작가능한, 상기메모리 디바이스 인터페이스; 및

상기 링크 인터페이스 및 상기 메모리 디바이스 인터페이스에 결합된 메모리 시퀀서로서, 상기 링크 인터페이스로부터 수신된 메모리 요청들에 응답하여 메모리 요청들을 상기 메모리 디바이스 인터페이스에 결합하도록 동작가능하며, 또한 상기 메모리 힌트들에 응답하여 상기 메모리 디바이스들의 동작성을 동적으로 조정하도록 동작가능한, 상기 메모리 시퀀서를 포함하는, 메모리 모듈.

청구항 2.

제 1 항에 있어서, 상기 링크 인터페이스는 광 입력/출력 포트를 포함하는, 메모리 모듈.

청구항 3.

제 1 항에 있어서, 상기 힌트들 중 적어도 하나는 상기 메모리 디바이스들을 페이지 모드에 두는 신호들을 포함하는, 메모리 모듈.

청구항 4.

제 3 항에 있어서, 상기 힌트들 중 적어도 하나는 개방되어 있는 다수의 페이지들을 포함하는, 메모리 모듈.

청구항 5.

제 1 항에 있어서, 상기 힌트들 중 적어도 하나는 프리페칭 힌트를 포함하는, 메모리 모듈.

청구항 6.

제 1 항에 있어서, 상기 힌트들 중 적어도 하나는 보내질 다수의 캐시 라인들을 포함하는, 메모리 모듈.

청구항 7.

제 1 항에 있어서, 상기 힌트들 중 적어도 하나는 데이터를 가져올 일련의 주소들을 표시하는 스트라이드를 포함하는, 메모리 모듈.

청구항 8.

제 1 항에 있어서, 상기 힌트들 중 적어도 하나는 스킵할 다수의 캐시 라인들을 포함하는, 메모리 모듈.

청구항 9.

제 1 항에 있어서, 상기 메모리 디바이스들은 동적 랜덤 액세스 메모리 디바이스들을 포함하는, 메모리 모듈.

청구항 10.

제 1 항에 있어서, 상기 링크 인터페이스 및 상기 메모리 시퀀서에 결합된 요청 디코더를 더 포함하고, 상기 요청 디코더는 상기 메모리 요청들 내 상기 힌트를 디코딩하도록 동작가능한, 메모리 모듈.

청구항 11.

메모리 허브에 있어서,

상기 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위한 메모리 요청들을 수신하는 링크 인터페이스로서, 상기 메모리 요청들 중 적어도 일부는 상기 메모리 디바이스들의 후속 동작에 관한 정보를 제공하는 각각의 메모리 힌트들을 포함하는, 상기 링크 인터페이스;

상기 메모리 디바이스들에 결합된 메모리 디바이스 인터페이스로서, 상기 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위해 메모리 디바이스들에 메모리 요청들을 결합하고 상기 메모리 요청들 중 적어도 일부에 응답하여 판독 데이터를 수신하도록 동작가능한, 상기 메모리 디바이스 인터페이스; 및

상기 링크 인터페이스 및 상기 메모리 디바이스 인터페이스에 결합된 메모리 시퀀서로서, 상기 링크 인터페이스로부터 수신된 메모리 요청들에 응답하여 메모리 요청들을 상기 메모리 디바이스 인터페이스에 결합하도록 동작가능하며, 또한 상기 메모리 힌트들에 응답하여 상기 메모리 디바이스들의 동작성을 동적으로 조정하도록 동작가능한, 상기 메모리 시퀀서를 포함하는, 메모리 허브.

청구항 12.

제 11 항에 있어서, 상기 링크 인터페이스는 광 입력/출력 포트를 포함하는, 메모리 허브.

청구항 13.

제 11 항에 있어서, 상기 힌트들 중 적어도 하나는 상기 메모리 디바이스들을 페이지 모드에 두는 신호들을 포함하는, 메모리 허브.

청구항 14.

제 13 항에 있어서, 상기 힌트들 중 적어도 하나는 개방되어 있는 다수의 페이지들을 포함하는, 메모리 허브.

청구항 15.

제 11 항에 있어서, 상기 힌트들 중 적어도 하나는 프리페칭 힌트를 포함하는, 메모리 허브.

청구항 16.

제 11 항에 있어서, 상기 힌트들 중 적어도 하나는 보내질 다수의 캐시 라인들을 포함하는, 메모리 허브.

청구항 17.

제 11 항에 있어서, 상기 힌트들 중 적어도 하나는 데이터를 가져올 일련의 주소들을 표시하는 스트라이드를 포함하는, 메모리 허브.

청구항 18.

제 11 항에 있어서, 상기 힌트들 중 적어도 하나는 스킵할 다수의 캐시 라인들을 포함하는, 메모리 허브.

청구항 19.

제 11 항에 있어서, 상기 메모리 디바이스들은 동적 랜덤 액세스 메모리 디바이스들을 포함하는, 메모리 허브.

청구항 20.

제 12 항에 있어서, 상기 링크 인터페이스 및 상기 메모리 시퀀서에 결합된 요청 디코더를 더 포함하고, 상기 요청 디코더는 상기 힌트를 디코딩하도록 동작가능한, 메모리 허브.

청구항 21.

컴퓨터 시스템에 있어서,

중앙 처리 유닛("CPU");

상기 CPU에 결합된 시스템 제어기로서, 입력 포트 및 출력 포트를 구비하는, 상기 시스템 제어기;

상기 시스템 제어기를 통해 상기 CPU에 결합된 입력 디바이스;

상기 시스템 제어기를 통해 상기 CPU에 결합된 출력 디바이스;

상기 시스템 제어기를 통해 상기 CPU에 결합된 저장 디바이스;

복수의 메모리 모듈들을 포함하고, 상기 메모리 모듈들 각각은,

복수의 메모리 디바이스들; 및

상기 시스템 제어기 및 상기 메모리 디바이스들에 결합된 메모리 허브를 포함하고, 상기 메모리 허브는,

상기 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위한 메모리 요청들을 상기 시스템 제어기로부터 수신하는 링크 인터페이스로서, 상기 메모리 요청들 중 적어도 일부는 상기 메모리 디바이스들의 후속 동작에 관한 정보를 제공하는 각각의 메모리 힌트들을 포함하는, 상기 링크 인터페이스;

상기 메모리 디바이스들 및 상기 링크 인터페이스에 결합된 것으로, 상기 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위해 메모리 디바이스들에 메모리 요청들을 결합하고 상기 메모리 요청들 중 적어도 일부에 응답하여 판독 데이터를 수신하도록 동작가능한, 상기 메모리 디바이스 인터페이스; 및

상기 링크 인터페이스 및 상기 메모리 디바이스 인터페이스에 결합된 메모리 시퀀서로서, 상기 링크 인터페이스로부터 수신된 메모리 요청들에 응답하여 메모리 요청들을 상기 메모리 디바이스 인터페이스에 결합하도록 동작가능하며, 또한 상기 메모리 힌트에 응답하여 상기 메모리 디바이스들의 동작성을 동적으로 조정하도록 동작가능한, 상기 메모리 시퀀서를 포함하는, 컴퓨터 시스템.

청구항 22.

제 21 항에 있어서, 상기 링크 인터페이스는 광 입력/출력 포트를 포함하는, 메모리 시스템.

청구항 23.

제 21 항에 있어서, 상기 시스템 제어기에 의해 생성된 상기 힌트들 중 적어도 하나는 상기 메모리 디바이스들을 페이지 모드에 두는 신호들을 포함하는, 메모리 시스템.

청구항 24.

제 21 항에 있어서, 상기 힌트들 중 적어도 하나는 개방되어 있는 다수의 페이지들을 포함하는, 메모리 시스템.

청구항 25.

제 21 항에 있어서, 상기 시스템 제어기에 의해 생성된 상기 힌트들 중 적어도 하나는 프리페칭 힌트를 포함하는, 메모리 시스템.

청구항 26.

제 21 항에 있어서, 상기 힌트들 중 적어도 하나는 상기 시스템 제어기로부터 보내질 다수의 캐시 라인들을 포함하는, 메모리 시스템.

청구항 27.

제 21 항에 있어서, 상기 시스템 제어기에 의해 생성된 상기 힌트들 중 적어도 하나는 데이터를 가져올 일련의 주소들을 표시하는 스트라이드를 포함하는, 메모리 시스템.

청구항 28.

제 21 항에 있어서, 상기 힌트들 중 적어도 하나는 스킵할 다수의 캐시 라인들을 포함하는, 메모리 시스템.

청구항 29.

제 21 항에 있어서, 상기 메모리 디바이스들은 동적 랜덤 액세스 메모리 디바이스들을 포함하는, 메모리 시스템.

청구항 30.

제 21 항에 있어서, 상기 링크 인터페이스 및 상기 메모리 시퀀스에 결합된 요청 디코더를 더 포함하고, 상기 요청 디코더는 상기 힌트를 디코딩하도록 동작가능한, 메모리 시스템.

청구항 31.

복수의 메모리 디바이스들을 포함한 메모리 모듈에서 메모리 시퀀싱을 조정하는 방법에 있어서,

상기 메모리 모듈에 장착된 메모리 디바이스에의 액세스를 위한 메모리 요청들을 생성하는 단계로서, 상기 메모리 요청들 중 적어도 일부는 메모리 힌트를 포함하는, 상기 생성 단계;

상기 메모리 디바이스들의 후속 동작에 관한 정보를 제공하는 단계;

상기 메모리 모듈에서 상기 메모리 요청들을 수신하는 단계;

상기 메모리 힌트에 기초하여 메모리 모듈의 동작성을 조정하는 단계; 및

상기 수신된 메모리 요청에 응답하여 상기 메모리 디바이스에 상기 메모리 요청을 결합하는 단계를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 32.

제 31 항에 있어서, 상기 메모리 힌트에 기초하여 메모리 모듈의 동작성을 조정하는 단계는 상기 메모리 힌트에 기초하여 상기 메모리 시퀀서의 동작성을 조정하는 단계를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 33.

제 31 항에 있어서, 상기 메모리 힌트에 기초하여 메모리 모듈의 동작성을 조정하는 단계는 상기 메모리 힌트에 기초하여 상기 메모리 디바이스들의 동작성을 조정하는 단계를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 34.

제 31 항에 있어서, 상기 메모리 디바이스에 상기 메모리 요청을 결합하는 단계는 광 입력/출력 포트를 포함하는 링크 인터페이스를 통해 상기 메모리 요청을 결합하는 단계를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 35.

제 31 항에 있어서, 상기 메모리 모듈의 동작성을 조정하는 단계는 상기 메모리 디바이스들을 페이지 모드에 두는 단계를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 36.

제 31 항에 있어서, 상기 메모리 모듈의 동작성을 조정하는 단계는 다수의 페이지들을 개방하여 두는 단계를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 37.

제 31 항에 있어서, 상기 시스템 제어기에 의해 생성된 상기 힌트는 프리페칭 힌트를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 38.

제 31 항에 있어서, 상기 시스템 제어기에 의해 생성된 상기 힌트는 보내질 다수의 캐시 라인들을 포함하는, 메모리 시퀀싱 조정 방법.

청구항 39.

제 31 항에 있어서, 상기 시스템 제어기에 의해 생성된 상기 힌트는 데이터를 가져올 일련의 주소들의 시퀀스를 표시하는 스트라이드를 포함하는, 메모리 시퀀싱 조정 방법.

청구항 40.

제 31 항에 있어서, 상기 힌트는 스킵할 다수의 캐시 라인들을 포함하는, 메모리 시퀀싱 조정 방법.

청구항 41.

제 31 항에 있어서, 상기 메모리 디바이스들은 동적 랜덤 액세스 메모리 디바이스들을 포함하는, 메모리 시퀀싱 조정 방법.

청구항 42.

제 31 항에 있어서, 요청 디코더로 상기 힌트를 디코딩하는 단계를 더 포함하는, 메모리 시퀀싱 조정 방법.

명세서

기술분야

이 발명은 컴퓨터 시스템들에 관한 것으로, 특히 몇 개의 메모리 디바이스들을 한 프로세서 혹은 다른 메모리 액세스 디바이스에 결합하는 메모리 허브를 구비하는 컴퓨터 시스템에 관한 것이다.

배경기술

컴퓨터 시스템들은 프로세서에 의해 액세스되는 데이터를 저장하기 위해서, 이를테면 동적 랜덤 액세스 메모리("DRAM") 디바이스들과 같은 메모리 디바이스들을 사용한다. 이들 메모리 디바이스들은 통상적으로 컴퓨터 시스템 내 시스템 메모리로서 사용된다. 통상적인 컴퓨터 시스템에서, 프로세서는 프로세서 버스 및 메모리 제어기를 통해 시스템 메모리와 통신한다. 프로세서는 판독 코맨드(read command)와 같은 메모리 코맨드, 및 데이터 혹은 명령들을 읽어낼 위치를 지정하는 주소를 포함하는 메모리 요청을 발생한다. 메모리 제어기는 시스템 메모리에 인가되는, 행 주소 및 열 주소들뿐만 아니라 적절한 코맨드 신호들을 생성하도록 코맨드 및 주소를 사용한다. 코맨드들 및 주소들에 응답하여, 시스템 메모리와 프로세서 간에 데이터가 전송된다. 메모리 제어기는 시스템 제어기의 일부이고, 이 시스템 제어기는 프로세서 버스를 PCI 버스와 같은 확장버스에 결합하기 위한 버스 브릿지 회로를 또한 포함한다.

메모리 디바이스들의 동작속도가 계속적으로 증가하였을지라도, 동작 속도의 이러한 증가는 프로세서들의 동작속도의 증가와 보조를 맞추지 못하였다. 프로세서들을 메모리 디바이스들에 결합하는 메모리 제어기들의 동작 속도 증가는 훨씬 더 느렸다. 메모리 제어기들 및 메모리 디바이스들의 상대적으로 느린 속도는 프로세서와 메모리 디바이스들간에 데이터 대역폭을 제한시킨다.

프로세서들과 메모리 디바이스들간의 제한된 대역폭 외에도, 컴퓨터 시스템들의 수행은 시스템 메모리 디바이스들로부터 데이터를 읽는데 필요한 시간을 증가시키는 레이턴시 문제들에 의해서도 제한된다. 특히, 메모리 디바이스 판독 코맨드가 동기형 DRAM("SDRAM") 디바이스와 같은 시스템 메모리 디바이스에 결합될 때, 읽혀진 데이터는 몇 클럭 기간들의 지연 후에만 SDRAM 디바이스로부터 출력된다. 그러므로, SDRAM 디바이스들이 동기에 맞추어 높은 데이터 레이트로 버스트 데이터를 출력할 수 있을지라도, 데이터를 초기에 제공함에 있어 지연은 이러한 SDRAM 디바이스들을 사용하는 컴퓨터 시스템의 동작속도를 현저하게 느리게 할 수 있다.

메모리 레이턴시 문제를 완화시키는 한 방법은 메모리 허브를 통해 프로세서에 결합된 복수의 메모리 디바이스들을 사용하는 것이다. 메모리 허브 구조에서, 시스템 제어기 혹은 메모리 제어기는 몇 개의 메모리 모듈들에 결합되고, 그 각각은 몇 개의 메모리 디바이스들에 결합된 메모리 허브를 포함한다. 메모리 허브는 메모리 요청들과 응답들을 제어기와 메모리 디바이스들간에 효율적으로 보낸다. 이러한 구조를 채용하는 컴퓨터 시스템들을 다른 메모리 디바이스가 이전의 메모리 액세스에 응답하고 있을 동안에 프로세서가 한 메모리 디바이스를 액세스할 수 있기 때문에 큰 대역폭을 가질 수 있다. 예를 들면, 프로세서는 시스템 내 다른 메모리 디바이스가 프로세서에 판독 데이터를 제공할 준비를 하고 있는 동안에 시스템 내 메모리 디바이스들 중 하나에 기입 데이터를 출력할 수 있다.

메모리 허브들을 사용하는 컴퓨터 시스템들이 우수한 성능을 제공할 수 있을지라도, 그럼에도 이들은 몇가지 이유로 종종 최적 속도로 동작하지 못한다. 예를 들면, 메모리 허브들이 컴퓨터 시스템들에게 보다 큰 메모리 대역폭을 제공할 수 있을지라도, 이들은 위에 기술한 유형의 레이턴시 문제들을 여전히 갖는다. 구체적으로, 다른 메모리 디바이스가 데이터를 전송할 준비를 하고 있는 동안에 프로세서가 한 메모리 디바이스와 통신할 수 있을지라도, 그 다른 메모리 디바이스로부터 데이터가 사용될 수 있기 전에 한 메모리로부터 데이터를 수신하는 것이 종종 필요하다. 다른 메모리 디바이스로부터 수신된 데이터가 사용될 수 있기 전에 한 메모리 디바이스로부터 데이터가 수신되어야 하는 경우에, 레이턴시 문제는 이러한 컴퓨터 시스템들의 동작 속도를 계속적으로 느리게 한다.

메모리 디바이스들에서 레이턴시를 감소시키는데 사용된 한 기술은 데이터를 프리페치(prefetch)하는 것, 즉, 실행 중의 어떤 프로그램이 데이터를 요청하기 전에 시스템 메모리로부터 데이터를 판독하는 것이다. 일반적으로, 프리페치될 데이터는 전에 페치된 데이터의 패턴에 근거하여 선택된다. 패턴은 실행중의 프로그램에 의해 데이터가 필요하기 전에 시퀀스에서 후속 주소들로부터 데이터가 페치될 수 있도록 데이터를 페치해 올 일련의 주소들만큼이나 간단할 수 있다. "스트라이드"("stride")라고도 알려진 패턴은 물론 더욱 복잡할 수도 있다.

또한, 메모리 허브들이 컴퓨터 시스템들에 보다 큰 메모리 대역폭을 제공할 수 있을지라도, 이들은 스루풋 문제들(throughput problems)을 여전히 갖는다. 예를 들면, 데이터가 한 특정의 행의 메모리 셀들로부터 판독될 수 있기 전에, 통상적으로 어레이 내 디지트 라인들은 어레이 내 디지트 라인들을 등화시킴으로써 프리차지된다. 이 특정의 행은 그 행 내의 메모리 셀들을 각각의 열들 내 한 디지트 라인에 결합함으로써 개방된다. 각 열 내 디지트 라인들 간에 결합된 각각의 센스 증폭기는 각각의 메모리 셀에 저장된 데이터에 대응하는 전압 변화에 응답한다. 일단 행이 개방되었으면, 디지트 라인들을 데이터 판독 경로에 결합함으로써 그 개방된 행의 각 열로부터 데이터가 결합될 수 있다. 페이지라고도 하는 한 행을 개방하는 것은 유한한 시간량을 소비하여 메모리 스루풋에 한계를 둔다.

마지막으로, 데이터를 프리페치할지(그리고 어느 데이터를 프리페치할지) 여부와, 행을 프리차지 혹은 개방할지 여부와, 액세스된 데이터를 캐시할지 여부의 최적의 결정은 시간에 따라 변할 수 있고 메모리 허브에 결합된 프로세서에 의해 실행되는 애플리케이션의 함수로서 달라질 수도 있다.

그러므로 메모리 허브 구조의 이점들을 제공하며 이러한 시스템들에서 공통적인 레이턴시 및/또는 스루풋 문제들도 최소화하고, 그럼으로써 메모리 디바이스들에 큰 대역폭과, 큰 스루풋과, 낮은 레이턴시를 제공하는 컴퓨터 구조에 대한 요구가 있다. 이러한 시스템은 메모리 허브의 동작이 시간에 따라 변경될 수 있게 하는 것이 바람직할 것이다.

발명의 상세한 설명

본 발명의 한 양상에 따라, 복수의 메모리 디바이스들 및 메모리 허브를 포함하는 메모리 모듈 및 방법이 제공된다. 메모리 허브는 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위한 메모리 요청들을 수신하는 광학 입력/출력포트와 같은 링크 인터페이스를 내장한다. 또한, 메모리 허브는 메모리 디바이스들에 결합된 메모리 디바이스 인터페이스를 내장하고, 메모리 디바이스 인터페이스는 메모리 디바이스들 중 적어도 한 메모리 디바이스 내 메모리 셀들에의 액세스를 위해 메모리 디바이스들에 메모리 요청들을 결합하고 메모리 요청들 중 적어도 일부를 응답하여 판독 데이터를 수신하게 동작한다. 또한 메모리 허브는 시스템 제어기에 결합되고, 시스템 제어기는 메모리 힌트를 생성하게 동작한다. 메모리 허브는 링크 인터페이스 및 메모리 디바이스 인터페이스에 결합된 메모리 시퀀서를 내장한다. 메모리 시퀀서는 링크 인터페이스로부터 수신된 메모리 요청들에 응답하여 메모리 디바이스 인터페이스에 메모리 요청들을 결합하게 동작한다. 메모리 시퀀서는 메모리 힌트에 응답하여 동작성을 동적으로 조정하게 또한 동작한다.

실시예

본 발명의 일례에 따른 컴퓨터 시스템(100)이 도 1에 도시되었다. 컴퓨터 시스템(100)은 특정 계산들 혹은 태스크들을 수행하기 위한 특정 소프트웨어를 실행하는 등의 각종의 계산 기능들을 수행하기 위한 프로세서(104)를 포함한다. 프로세서(104)는 통상적으로 주소 버스, 제어 버스, 및 데이터 버스를 포함하는 프로세서 버스(106)를 포함한다. 통상적으로, 프로세서 버스(106)는 앞에서 언급한 바와 같이 정적 랜덤 액세스 메모리("SRAM")인 캐시 메모리(108)에 결합된다. 마지막으로, 프로세서 버스(106)는 "노스 브리지" 혹은 "메모리 제어기"라고도 하는 시스템 제어기(110)에 결합된다.

시스템 제어기(110)는 다양한 상이한 구성 성분들에 대한 프로세서(104)에의 통신로로서 기능한다. 구체적으로, 시스템 제어기(110)는 통상적으로 그래픽스 제어기(112)에 결합되는 그래픽스 포트를 포함하고, 그래픽스 제어기는 비디오 단말(114)에 결합된다. 시스템 제어기(110)는 조작자가 컴퓨터 시스템(100)과 인터페이스할 수 있게, 키보드 혹은 마우스와 같은 하나 이상의 입력 디바이스들(118)에 결합된다. 통상적으로, 컴퓨터 시스템(100)은 시스템 제어기(110)를 통해 프로세서(104)에 결합된, 프린터와 같은 하나 이상의 출력 디바이스들(120)을 또한 포함한다. 프로세서(104)가 내부 혹은 외부 저장 매체(도시 생략)에 데이터를 저장 혹은 데이터를 불어들일 수 있게 시스템 제어기(110)를 통해 프로세서(104)에 하나 이상의 데이터 저장 디바이스들(124)이 또한 결합된다. 전형적인 저장 디바이스들(124)의 예들은 하드 및 플로피 디스크들, 테이프 카세트들, 및 콤팩트 디스크 판독전용 메모리들(CD-ROMS)을 포함한다.

시스템 제어기(110)는 컴퓨터 시스템(100)용 시스템 메모리로서 작용하는 몇 개의 메모리 모듈들(130a,b...n)에 결합된다. 메모리 모듈들(130)은 광학 혹은 전기적 통신로 혹은 그 외 어떤 상이한 유형의 통신로일 수 있는 고속 링크(134)를 통해 시스템 제어기(110)에 결합되는 것이 바람직하다. 고속링크(134)는 양방향 링크일 수도 있고, 두 개의 개별적인 양방향 링크들 중 하나는 시스템 제어기(110)에서 메모리 모듈들(130)로 신호들을 결합하고 다른 하나는 메모리 모듈들(130)에서 시스템 모듈들(110)로 신호들을 결합하는 이들 두 개별적 양방향 링크들을 포함할 수도 있다. 고속링크(134)가 광통신로로서 구현되는 경우에, 광통신로는 예를 들면 하나 이상의 광섬유들 형태일 수 있다. 이러한 경우에, 시스템 제어기(110) 및 메모리 모듈들은 광 입력/출력 포트 혹은 광통신로에 결합된 별도의 입력 및 출력 포트들을 포함할 것이다. 메모리 모듈들(130) 각각을 서로간에 혹은 시스템 제어기(110)에 결합하기 위해 별도의 고속 링크(134)의 한 세그먼트가 사용되는 점대점 결합 구성으로 메모리 모듈들(130)이 시스템 제어기(110)에 결합된 것이 도시되었다. 그러나, 메모리 모듈들(130) 모두에 하나의 고속 링크(도시 생략)이 결합되는 멀티-드롭 장치와 같은 그 외 상이한 토폴로지들도 사용될 수 있음을 알 것이다. 스위치(도시 생략)을 통해 메모리 모듈들(130) 각각에 선택적으로 시스템 제어기(110)가 결합되는 스위칭 토폴로지가 사용될 수도 있다. 그 외 사용될 수 있는 토폴로지들은 당업자에게 명백할 것이다.

고속 링크(134)는 시스템 제어기(110)와 메모리 모듈들간에 코맨드, 주소 및 데이터 신호들을 통신하기 위한 경로로서 작용한다. 코맨드, 주소 및 데이터 신호들은 다양한 포맷들을 취할 수 있다. 그러나, 도 1에 도시한 실시예에서, 주소 및 기입 데이터 신호들은 모두가, 시스템 제어기(100)에서 메모리 모듈들(130)로 전송되는 메모리 패킷들 내 내장된다. 판독 데이터 신호들을 내포한 메모리 패킷들은 메모리 모듈들(130)에서 시스템 제어기(110)로 결합된 패킷들에 내장된다.

메모리 모듈들(130) 각각은 도 1에 예시한 예에서 동기식 동적 랜덤 액세스 메모리("SDRAM") 디바이스들인 32 메모리 디바이스들(148)에의 액세스를 제어하기 위한 메모리 허브(140)를 포함한다. 그러나, 소수 혹은 다수의 메모리 디바이스들(148)이 사용될 수도 있고, SDRAM 이외의 메모리 디바이스들도 물론 사용될 수도 있다. 도 1에 예시한 예에서, 메모리 허브들(140)은 고속 링크(134)를 통해 4개의 독립적인 메모리 채널들(149)로 통신한다. 이 예에서, 도 1에는 도시되지 않았지만, 4개의 메모리 허브 제어기들(128)은 각각이 하나의 메모리 채널(149)로부터 데이터를 받도록 제공된다. 그러나, 소수 혹은 다수의 메모리 채널들(149)이 사용될 수 있다. 메모리 허브(140)는 통상적으로 제어 버스, 주소 버스 및 데이터 버스를 포함하는 버스 시스템(150)을 통해 시스템 메모리 디바이스들(148) 각각에 결합된다.

본 발명의 실시예에 따른 메모리 허브(200)가 도 2에 도시되었다. 메모리 허브(200)는 도 1의 메모리 허브(140)를 대신할 수 있다. 메모리 허브(200)는 본 예에서 종래의 SDRAM 디바이스들인 4개의 메모리 디바이스들(240 a 내지 d)에 결합된 것으로서 도 2에 도시되었다. 대안적인 실시예에서, 메모리 허브(200)는 각 뱅크는 전형적으로 복수의 메모리 디바이스들을 구비한 것으로, 단지 4개의 서로 다른 메모리 디바이스들(240 a 내지 d)이 아닌 4개의 서로 상이한 뱅크들의 메모리 디바이스들에 결합된다. 그러나, 예를 제공할 목적으로, 본 설명은 4개의 메모리 디바이스들(240 a 내지 d)에 결합된 메모리 허브(200)를 참조할 것이다. 복수의 메모리 뱅크들을 수용하기 위해 메모리 허브(200)에 대한 필요한 변형들은 당업자들의 지식 내에 있음을 알 것이다.

메모리 허브(200)에는 또한 메모리 허브(200)가 위치한 제1 고속 데이터 링크(220) 및 제2 고속 데이터 링크(222) 각각에 메모리 모듈을 결합하기 위한 링크 인터페이스들(210 a 내지 d, 212 a 내지 d)가 포함된다. 도 1을 참조하여 앞에서 논한 바와 같이, 고속 데이터 링크들(220, 222)은 광학 혹은 전기적 통신로 혹은 이외 어떤 상이한 유형의 통신로를 사용하여 구현될 수 있다. 링크 인터페이스들(210 a 내지 d, 212 a 내지 d)은 통상적인 것이고, 데이터, 코맨드, 주소 정보를 고속 데이터 링크들(220, 220)에 및 이들로부터 전송하는데 사용되는 회로를 포함한다. 공지된 바와 같이, 이러한 회로는 본 기술에 공지된 전송기 및 수신기 로직을 포함한다. 당업자들은 특정 유형들의 통신로들에 사용하기 위해 링크 인터페이스들(210 a 내지 d, 212 a 내지 d)을 수정하기 위한 충분한 이해를 갖고 있으며 링크 인터페이스들(210 a 내지 d, 212 a 내지 d)에 대한 이러한 변형들은 본 발명의 범위 내에서 행해질 수 있다는 것을 알 것이다. 예를 들면, 고속 데이터 링크(220, 222)가 광통신로를 사용하여 구현되는 경우에, 링크 인터페이스들(210 a 내지 d, 212 a 내지 d)은 광통신로들을 통해 결합된 광학 신호들을 전기 신호들로 변환할 수 있는 광 입력/출력 포트를 포함할 것이다.

링크 인터페이스들(210 a 내지 d, 212 a 내지 d)는 버스들(214)로 나타낸 복수의 버스 및 신호 라인들을 통해 스위치(260)에 결합된다. 대안적으로 단일 양방향 데이터 버스가 링크 인터페이스들(210 a 내지 d, 212 a 내지 d)을 통해 양 방향들로 데이터를 결합하기 위해 제공될 수도 있지만, 버스들(214)은 통상적인 것이고 기입 데이터 버스 및 판독 데이터 버스를 포함한, 버스들(214)이 예로 제공된 것이고, 버스들(214)은 이를테면 캐시 코히어런시를 유지하는데 사용될 수 있는 요청 라인 및 스눕 라인을 더 포함하는 등 소수의 혹은 다수의 신호 라인들을 포함할 수도 있음을 당업자들은 알 것이다.

링크 인터페이스들(210 a 내지 d, 212 a 내지 d)은 다양한 구성으로 시스템 메모리에 메모리 허브(200)가 접속될 수 있게 하는 회로를 포함한다. 예를 들면, 도 1에 도시한 바와 같이, 각 메모리 모듈(130)을 다른 메모리 모듈(130)에 혹은 메모리 허브 제어기(128)에 링크 인터페이스들(210 a 내지 d 혹은 212 a 내지 d)을 통해 결합함으로써 점대점 장치가 구현될 수 있다. 이러한 유형의 상호접속은 비교적 낮은 커패시턴스, 신호들을 반사시키는 비교적 적은 라인 불연속성들 및 비교적 짧은 신호 경로들을 포함하여, 몇가지 이유로 프로세서(104)와 메모리 허브(200)간에 더 나은 신호 결합을 제공한다. 대안적으로, 메모리 모듈들을 직렬로 결합함으로써 멀티-드롭 혹은 데이지 체인 구성이 구현될 수도 있다. 예를 들면, 링크 인터페이스들(210 a 내지 d)은 제 1 메모리 모듈을 결합하는데 사용될 수 있고 링크 인터페이스들(212 a 내지 d)는 제 2 메모리 모듈을 결합하는데 사용될 수 있다. 프로세서 혹은 시스템 제어기에 결합된 메모리 모듈은 한 세트의 링크 인터페이스들을 통해 그에 결합될 것이고 또한 상이한 한 세트의 링크 인터페이스들을 통해 다른 메모리 모듈에 결합될 것이다. 본 발명의 일 실시예에서, 메모리 모듈의 메모리 허브(200)는 멀티-드롭 구성으로 프로세서에 결합된다.

스위치(260)는 4개의 메모리 인터페이스들(270 a 내지 d)에 또한 결합되고, 이들 인터페이스들은 각각 시스템 메모리 디바이스들(240 a 내지 d)에 결합된다. 각 시스템 메모리 디바이스(240 a 내지 d)마다 별도의 독립적인 메모리 인터페이스(270 a 내지 d)를 제공함으로써, 메모리 허브(200)은 통상적으로 단 채널 메모리 구조들에서 발생하는 것인 버스 혹은 메모리 뱅크 충돌들을 피한다. 스위치(260)는 버스들(274)에 의해서 각각 복수의 버스 및 신호 라인들을 통해 각각의 메모리 인터페이스에 결합된다. 버스들(274)는 기입 데이터 버스, 판독 데이터 버스, 및 요청라인을 포함한다. 그러나, 단일 양방향 데이터 버스가 대안적으로 개별 기입 데이터 버스 및 판독 데이터 버스 대신으로 사용될 수도 있음을 알 것이다. 또한, 버스들(274)는 이전에 기술된 것들보다 적은 혹은 많은 신호 라인들을 포함할 수 있다.

본 발명의 실시예에서, 각 메모리 인터페이스(270 a 내지 d)는 이것이 결합되는 시스템 메모리 디바이스들(240 a 내지 d)에 특히 적합하다. 구체적으로, 각 메모리 인터페이스(270 a 내지 d)는 이것이 결합되는 시스템 메모리 디바이스(240 a 내지 d)에 의해 각각 수신 및 발생되는 특정의 신호들을 제공 및 수신하도록 구성된다. 또한, 메모리 인터페이스들(270 a 내지 d)은 서로 상이한 클럭 주파수들로 동작하는 시스템 메모리 디바이스들(240 a 내지 d)과 함께 동작할 수 있다. 결국, 메모리 인터페이스들(270 a 내지 d)은 메모리 허브(230)와 메모리 허브(200)에 결합된 메모리 디바이스들(240 a 내지 d)간 인터페이스에서 발생할 수 있는 변화들로부터 프로세서(104)를 분리시키며, 메모리 디바이스들(240 a 내지 d)이 인터페이스할 수 있는 보다 제어된 환경을 제공한다.

링크 인터페이스들(210 a 내지 d, 212 a 내지 d) 및 메모리 인터페이스들(270 a 내지 d)을 결합시키는 스위치(260)는 다양한 통상적인 혹은 이후 개발될 스위치들 중 하나일 수 있다. 예를 들면, 스위치(260)는 링크 인터페이스들(210 a 내지 d, 212 a 내지 d) 및 메모리 인터페이스들(270 a 내지 d)을 서로간에 다양한 구성들로 동시에 결합시킬 수 있는 크로스-바 스위치일 수 있다. 스위치(260)는 크로스바 스위치와 동일한 수준의 접속성을 제공하지 않으나 그럼에도 링크 인터페이스들(210 a 내지 d, 212 a 내지 d) 모두를 메모리 인터페이스들(270 a 내지 d) 각각에 결합시킬 수 있는 한 세트의 멀티플렉서들일 수도 있다. 스위치(260)는 어느 메모리 액세스들이 다른 메모리 액세스들보다 우선도를 받을 것인지를 판정하는 중재 로직(도시 생략)을 포함할 수도 있다. 이러한 기능을 수행하는 버스 중재는 당업자에겐 공지된 것이다.

도 2를 참조하면, 메모리 인터페이스들(270 a 내지 d) 각각은 각각의 메모리 제어기(280), 각각의 기입 버퍼(282), 및 각각의 캐시 메모리 유닛(284)을 포함한다. 메모리 제어기(280)는 이것이 결합되는 시스템 메모리 디바이스(240 a 내지 d)에 제어, 주소 및 데이터 신호들을 제공하고 상기 메모리 제어기가 결합되는 시스템 메모리 제어기(240 a 내지 d)로부터 데이터 신호들을 수신함으로써 통상의 메모리 제어기와 동일한 기능들을 수행한다. 기입 버퍼(282) 및 캐시 메모리 유닛(284)은 공지된 바와 같이, 태그 메모리, 데이터 메모리, 비교기 등을 포함하여, 버퍼 및 캐시 메모리의 정규의 구성 성분들을 포함한다. 기입 버퍼(282) 및 캐시 메모리 유닛(284)에 사용되는 메모리 디바이스들은 DRAM 디바이스들, 정적 랜덤 액세스 메모리("SRAM") 디바이스들, 그 외 상이한 유형들의 메모리 디바이스들 혹은 이들 3개의 조합일 수 있다. 또한, 캐시 메모리 유닛(284)에서 사용되는 상이한 구성 성분들뿐만 아니라 이들 메모리 디바이스들 중 어느 하나 혹은 전부 내장형 혹은 독립형의 디바이스들일 수 있다.

각 메모리 인터페이스(270 a 내지 d) 내 기입 버퍼(282)는 판독 요청이 서비스되는 중에 기입 요청들을 저장하는데 사용된다. 이러한 시스템에서, 프로세서(104)는 기입 요청이 보내지는 메모리 디바이스가 전의 기입 혹은 판독 요청을 서비스하여 비지 상태에 있을지라도 기입 요청을 시스템 메모리 디바이스(240 a 내지 d)에 발행할 수 있다. 이러한 방식을 사용하여, 메모리 요청들은 후속 판독 요청이 서비스 되는 중에 이전의 기입 요청이 기입 버퍼(282)에 저장될 수 있기 때문에 순서없이 서비스될 수 있다. 판독 요청이 서비스될 수 있게 기입 요청들을 버퍼링하는 능력은 판독 요청들의 시간적 순서에 관계없이 이들 판독 요청들에 제 1 우선도가 주어질 수 있기 때문에 메모리 판독 레이턴시를 크게 줄일 수 있다. 예를 들면, 판독 요청들로 산재된 일련의 기입 요청들은 기입 버퍼(282)에 저장됨으로써, 파이프라인 방식으로 판독 요청들이 서비스되고 이어서 파이프라인 방식으로 그 저장된 기입 요청들을 서비스될 수 있다. 결국, 교호적 기입 및 요청들에 대해서 기입 요청을 메모리 디바이스들(270 a 내지 d)에 결합하고 이어서 판독 요청을 메모리 디바이스들(270 a 내지 d)에 기입하는 것 간에 긴 세틀링 시간들을 피할 수 있다.

각 메모리 인터페이스(270 a 내지 d)에 캐시 메모리 유닛(284)의 사용으로, 프로세서(104)는 각각의 시스템 메모리 디바이스(240 a 내지 d)에 보내진 판독 코맨드에 응한 데이터를 수신하는데, 데이터가 그 메모리 디바이스(240 a 내지 d)로부터 최근에 판독 혹은 이에 기입된 경우에 이러한 데이터를 메모리 디바이스(240 a 내지 d)가 제공할 것을 기다리지 않고도, 수신할 수 있게 된다. 이에 따라 캐시 메모리 유닛(284)은 시스템 메모리 디바이스들(240 a 내지 d)의 판독 레이턴시를 줄여 컴퓨터 시스템의 메모리 대역폭을 최대화한다. 유사하게, 프로세서(104)는 캐시 메모리 유닛(284)에 기입 데이터를 저장하고 이어서 동일 메모리 인터페이스(270 a 내지 d) 내 메모리 제어기(280)가 이것이 결합된 시스템 메모리 디바이스(240 a 내지 d)에 기입 데이터를 캐시 메모리 유닛(284)으로부터 전송하는 중에 다른 기능들을 수행할 수 있다.

또한, 메모리 허브(200)는 진단 버스(292)를 통해 스위치(260)에 결합된 내장된 자기-테스트(BIST) 및 진단 엔진(290)을 포함한다. 진단 엔진(290)은 이를테면 시스템 관리 버스(SMBus)와 같은 관리 버스(296) 혹은 조인트 테스트 수행 그룹(JTAG) 및 IEEE 1149.1 표준들에 따른 관리 버스에 결합된다. SMBus 및 JTAG 표준들 둘 다 당업자들에 공지되어 있다. 일반적으로, 관리 버스(286)는 메모리 채널 및 링크 진단들을 수행하기 위해서 진단 엔진(290)에의 사용자 액세스를 제공한다. 예를 들면, 사용자는 진단 테스트를 행하거나 메모리 시스템 동작을 모니터하기 위해서 관리 버스(296)를 통해 별도의 PC 호스트를 결합할 수 있다. 진단 테스트 결과들을 액세스하기 위해 관리 버스(296)을 사용함으로써, 앞에서 논한 바와 같이, 테스트 프로브들의 사용에 관계된 문제들을 피할 수 있다. 본 발명의 범위 내에서 관리 버스(296)는 종래의 버스 표준들로부터 수정될 수 있음을 알 것이다. 또한, 진단 엔진(290)은 관리 버스(296)의 표준들을 수용할 것이며 이 경우 이러한 표준 관리 버스가 채용됨을 알 것이다. 예를 들면, 진단 엔진은 이러한 관리 버스가 사용되는 JTAG 버스 표준에 따르는 관리 버스 인터페이스를 가져야 한다.

또한, 메모리 허브(200)는 버스(288)를 통해 스위치(260)에 결합된 DMA 엔진(286)을 포함한다. DMA 엔진(286)은 프로세서(104)로부터의 개입없이 메모리 허브(200)가 시스템 메모리 내 한 위치에서 시스템 메모리 내 상이한 위치로 다수 데이터의 블록들을 옮길 수 있게 한다. 버스(288)는 시스템 메모리에서 데이터 전송들을 다루기 위해, 이를테면 주소, 제어, 데이터 버스들과 같은 복수의 통상의 버스 라인들 및 신호 라인들을 포함한다. DMA 엔진(286)은 당업자들에 공지된 통상의 DMA 동작들을 구현할 수 있다. DMA 엔진(286)은 프로세서 개입없이 DMA 메모리 동작들을 실행하기 위해 시스템 메

모리 내 링크 리스트를 읽을 수 있어, 프로세서(104) 및 대역폭 제한된 시스템 버스가 메모리 동작들을 실행하는 것으로부터 자유롭게 한다. DMA 엔진(286)은, 예를 들면, 시스템 메모리 디바이스들(240 a 내지 d) 각각에 대해, 복수의 채널들에 DMA 동작들을 수용시키는 회로를 포함할 수 있다. 이러한 복수의 채널 DMA 엔진들은 공지되어 있고 종래의 기술들을 사용하여 구현될 수 있다.

진단 엔진(290) 및 DMA 엔진(286)은 메모리 허브(200) 내 내장된 회로들인 것이 바람직하다. 그러나, 진단 엔진 및 메모리 허브(200)에 결합된 별도의 DMA 디바이스를 포함시키는 것은 본 발명의 범위 내이다.

위에 언급한 바와 같이, 코맨드, 주소 및 데이터 신호들은 메모리 패킷들 형태로 제어기(128)와 메모리 모듈들(130)간에 결합되는 것이 바람직하다. 본 발명의 일 실시예에 따라, 메모리 모듈들(130)의 다른 예상 수행을 표시하는 비트들인 "힌트"는 메모리 패킷들에 내장되고 메모리 모듈들(130) 내 하나 이상의 메모리 허브들(140)에 결합된다. 힌트, 혹은 힌트들은 후술하는 바와 같이 하나 이상의 메모리 허브들(140)의 거동을 수정한다. 특히, 힌트는 제어기(128)에 의해 알려진 혹은 추정된 정보에 기초하여 메모리 시퀀싱을 수정한다. 예를 들면, 제어기(128)는 메모리 요청기 혹은 주소 스트라이드와 같은 어드레싱 정보에 액세스할 수 있다.

어드레싱 힌트의 일례에서, 제어기(128)는 허브(140)를 페이지 모드에 놓이게 하고 열린 상태가 유지되게 다수의 페이지들을 확인하는 코맨드를 통신한다. 또 다른 예에서, 제어기(128)는 프리페칭에 관계된 힌트, 이를테면 후속하는 1, 2, 혹은 4 캐시 라인들을 제공한다. 또 다른 예에서, 제어기(128)는 허브(140)에 스트라이드를 통신하는데, 이를테면 다음 1, 2, 혹은 4 캐시 라인들을 스킵한다. 힌트의 또 다른 예에서, 제어기(128)는 허브 캐시 내에 한 특정의 캐시 라인을 둘지 여부를 지시할 수도 있다. 물론, 이외 상이한 힌트들도 사용될 수 있고, 혹은 기술된 힌트들에 다른 특정의 정보가 제공될 수도 있다.

도 3은 힌트(301)를 포함하는 기입 코맨드 패킷(300)을 도시한 것이다. 기입 코맨드 패킷(300)은 제어기(128)에 의해 발생되고 허브(140)에 통신된다. 패킷(300)은 힌트(301) 및 코맨드 코드(302)를 포함한다. 기입 코맨드 패킷(300)은 또한 기입 데이터(310), 기입 주소 정보(305)를 포함하며, 태그(311), 스트라이드(312), 유보(313), 길이(314) 및 에러 체크 정보(315)와 같은 다른 정보를 포함할 수도 있다.

도 4는 힌트(351)를 포함하는 판독 코맨드 패킷(350)를 도시한 것이다. 판독 코맨드 패킷(350)은 제어기(128)에 의해 발생되어 허브(140)에 통신된다. 패킷(350)은 힌트(351) 및 코맨드 코드(352)를 포함한다. 판독 코맨드 패킷(350)은 판독 주소 정보(355)를 또한 포함하며, 태그(361), 스트라이드(362), 유보(363), 길이(364), 및 에러 체크 정보(365)와 같은 다른 정보를 포함할 수 있다.

판독 및 기입 코맨드 패킷들, 이를테면 패킷들(300, 350)은 허브(140)에 보내진다. 패킷들(300 및/또는 350)을 수신하기 위한 허브(140)의 하나의 구현이 도 5에 도시되었다. 판독 혹은 기입 패킷이 400에서 링크로부터 수신된다. 요청 디코더(405)는 패킷을 수신하고 요청 큐(410)에 제공되는 요청 및 임의의 힌트 혹은 힌트들을 디코딩한다. 요청 디코더(405)는 판독 주소를 또한 디코딩하여 판독 주소를 비교기(415)에 제공한다. 기입 버퍼 큐(420)는 400에서 링크로부터 패킷들을 수신하고 기입 주소를 비교기(415)에 제공한다. 비교기(415)는 판독 및 기입 주소들을 비교하고, 임의의 기입 충돌들에 대해 요청 큐(410)에 통지한다. 요청 큐(410)가 판독 버퍼 히트를 확인하였다면, 요청 큐는 요청을 이행하기 위해서 프리페치 버퍼(425)를 액세스한다. 요청들 및 힌트들은 메모리 인터페이스(435)에 접속된 메모리 시퀀서(430)에 제공된다. 메모리 시퀀서(430)는 임의의 힌트 정보에 대해 동작하고, 메모리 인터페이스(435)를 통해 요청을 보낸다. 메모리 판독 데이터는, 적합하다면, 저장을 위해 프리페치 버퍼(425)에 결합된다.

전술한 바로부터, 본 발명의 특정의 실시예들이 예시 목적으로 개시되었을지라도, 본 발명의 정신 및 범위 내에서 다양한 수정들이 행해질 수 있음을 알 것이다. 따라서, 본 발명은 첨부한 청구항들에 의해 한정되는 것은 아니다.

도면의 간단한 설명

도 1은 복수의 메모리 모듈들 각각에 메모리 허브가 포함된 본 발명의 일례에 따른 컴퓨터 시스템의 블록도.

도 2는 본 발명의 예에 따라 도 1의 컴퓨터 시스템에 사용되는 메모리 허브의 블록도.

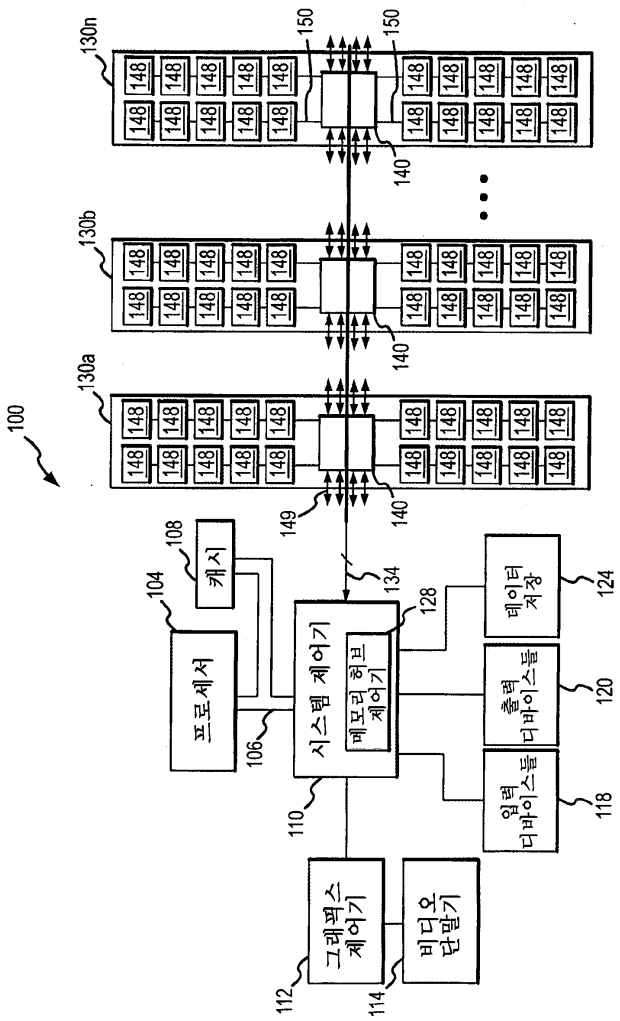
도 3은 본 발명의 일례에 따른 기입 코맨드 패킷의 개략적 개요도.

도 4는 본 발명의 일례에 따른 판독 코맨드 패킷의 개략적 개요도.

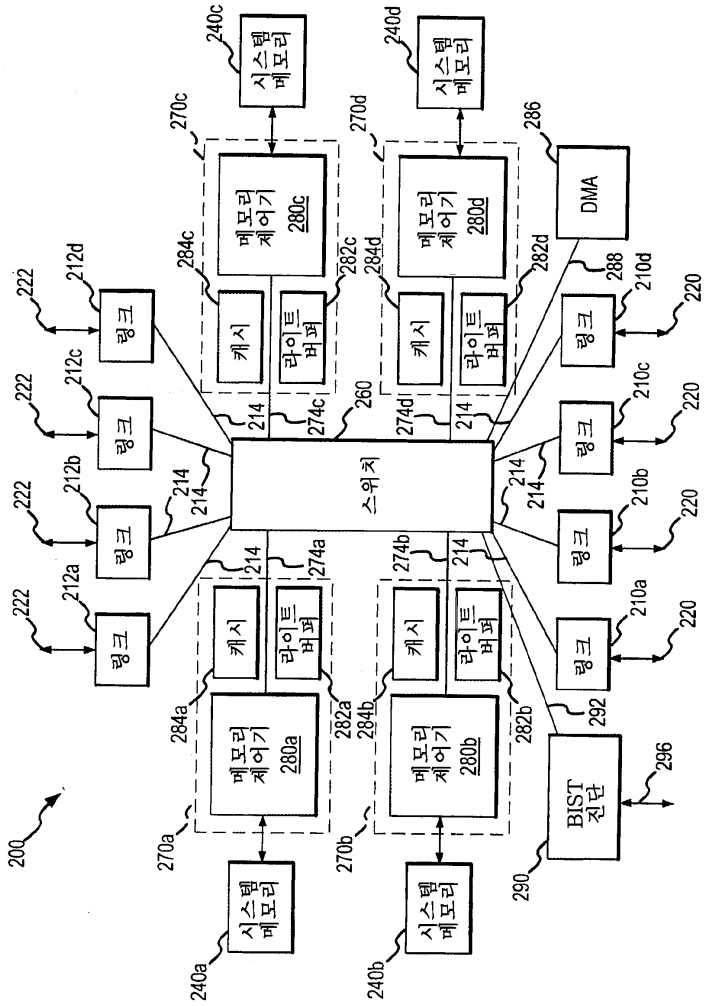
도 5는 본 발명의 일례에 따른 도 1의 컴퓨터 시스템에 사용되는 메모리 허브의 블록도.

도면

도면1



도면2



도면3

BIT-TIME	7	6	5	4	3	2	1	0
0	301 힌트			302 코맨드 코드				
1	311 TAG [7:0]							
2	312 스트라이드		313 리저브드		314 길이 [3:0]			
3	305 주소 [39:32]							
4	306 주소 [31:24]							
5	307 주소 [23:16]							
6	308 주소 [15:8]							
7	309 주소 [7:3]						Rsv	우선도
8	310 데이터 [7:0]							
9	311 데이터 [15:8]							
10	312 데이터 [23:16]							
11	313 데이터 [31:24]							
...	314 ...							
12+4N	315 CRC [7:0]							
13+4N	316 CRC [15:8]							
14+4N	317 CRC [23:16]							
15+4N	318 CRC [31:24]							

300

도면4

TIME	7	6	5	4	3	2	1	0
0	351 힌트			352 코맨드 코드				
1	361 TAG [7:0]							
2	362 스트라이드		363 리저브드		364 길이 [3:0]			
3	355 주소 [39:32]							
4	356 주소 [31:24]							
5	357 주소 [23:16]							
6	358 주소 [15:8]							
7	359 주소 [7:3]						Rsv	우선도
8	365 CRC [7:0]							
9	366 CRC [15:8]							
10	367 CRC [23:16]							
11	368 CRC [31:24]							

350

도면5

