

(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) Int. Cl. ⁶ G06F 9/46		(45) 공고일자 2001년11월30일	
		(11) 등록번호 10-0306636	
		(24) 등록일자 2001년08월11일	
(21) 출원번호	10-1996-0707280	(65) 공개번호	특1997-0704181
(22) 출원일자	1996년12월19일	(43) 공개일자	1997년08월09일
번역문제출일자	1996년12월19일		
(86) 국제출원번호	PCT/US1995/07392	(87) 국제공개번호	WO 1996/00940
(86) 국제출원일자	1995년06월07일	(87) 국제공개일자	1996년01월11일
(81) 지정국	국내특허 : 오스트레일리아 바베이도스 불가리아 브라질 캐나다 중국 체코 에스토니아 그루지야 헝가리 아이슬란드 일본 북한 대한민국 스리랑카 라이베리아 리투아니아 라트비아 마다가스카르 몽고 멕시코 노르웨이 뉴질랜드 폴란드 루마니아 싱가포르 슬로베니아 슬로바키아 AP ARIPO특허 : 말라위 수단 EA 유라시아특허 : 아르메니아 벨라루스 카자흐스탄 몰도바 러시아 EP 유럽특허 : 오스트리아 스위스 독일 덴마크 스페인 핀란드 영국 룩셈부르크 포르투갈 스웨덴 OA OAPI특허 : 부르키나파소 베냉 중앙아프리카 콩고 코트디부와르 카 메룬 가봉 기네 말리 모리타니 니제르 세네갈 차드 토고		
(30) 우선권주장	08/267,827	1994년06월28일	미국(US)
(73) 특허권자	인텔 코오퍼레이션 피터 엔. 데트킨		
(72) 발명자	미합중국 캘리포니아 산타클라라 미션 칼리지 블러바드 2200 알버트 알. 빌슨 미합중국 오리건 97123 힐스보로 사우스이스트 몬테고 4918 아니루드하 쿤두 미합중국 오리건 97124 힐스보로 노스이스트 스테이블 드라이브 5214		
(74) 대리인	박종혁, 장용식		

심사관 : 임인권

(54) PCI-ISA인터럽트프로토콜컨버터및선택메카니즘

명세서

[발명의 명칭]

PCI-ISA 인터럽트 프로토콜 컨버터 및 선택메카니즘

[도면의 간단한 설명]

제1도는 본 발명의 일 실시예에서 채용된 버스계층구조를 이용하는 컴퓨터 시스템 아키텍처를 나타내는 블록도.

제2도는 본 발명의 일 실시예에서 구현되는 프로그램가능 논리회로의 입력 및 출력을 나타내는 논리회로도.

제3도는 본 발명의 일 실시예에서 구현되는 프로그램가능 논리회로에 의해서 PCI인터럽트를 ISA 준수 인터럽트로 변화시키기 위한 프로그램가능 논리회로에 의해 구현되는 상태도.

제4도는 본 발명의 일 실시예에서 구현되는 프로그램가능 논리회로에 의하여 PCI 인터럽트를 ISA 준수 인터럽트로 변환시키기 위한 프로그램가능 논리회로에 의해 구현될 수 있는 상태도.

[발명의 상세한 설명]

[발명의 배경]

1. 발명의 분야

본 발명은 컴퓨터시스템의 주변장치접속에 관련된다. 특히 본 발명은 복수의 주변버스를 가지는 컴퓨터시스템을 위한 인터럽트처리에 관련된다.

2. 저작권의 보유

본 출원서류에 개시된 부분에는 저작권 보호를 받는 사항이 포함되어 있다. 저작권자는 특허상표 국 특허관련서류 내지 기록에 나타난 대로, 특허서류 내지 공개서류에 대한 누구에 의한 팩시밀리등을 이용한 복제에 대해서도 이의 제기하지 않는다. 그럼에도 불구하고 저작권은 저작권자가 보유한다.

3. 기술배경

컴퓨터 산업분야에서, 독립형 시스템을 위하여 광범위하게 채용되는 시스템 아키텍처는 AT시스템 설계이었다. 이 시스템 타입을 지원하는 컴퓨터 시스템 마더보드는 다양한 시스템 구성장치 사이에서 시스템 내부통신을 위한 표준 입력/출력(I/O)버스를 포함한다.

예를 들면, 시스템의 중앙처리장치(CPU)는 시스템의 하드디스크 드라이브 내지 다른 저장장치에서 데이터를 검색하거나 기록하기 위하여 이러한 버스를 이용할 것이다.

이러한 시스템 버스의 규격에는 잘 알려진 ISA규격 및 EISA 규격이 있고, 예로서 이 규격에 따라서 하드드라이브가 통합 드라이브 전자장치(Integrated Drive Electronics: IDE)인터페이스를 통하여 접속될 수도 있다.

컴퓨터 프로세서에 대한 기술이 발전함에 따라, 종래의 ISA규격, EISA 규격 및 다른 입력/출력 버스 규격은 진보된 프로세서를 갖춘 컴퓨터 시스템에 제공되는 것이 부적절하다고 판명되고 있다. 최근에, 일차 입력/출력버스로서 이용되기 위하여 기존의 버스 규격 보다 훨씬 넓은 대역폭을 제공하는 새로운 입력/출력 버스 규격이 도입되고 있다. 주변장치 상호접속(Peripheral Component Interconnect; PCI)버스는 컴퓨터 산업분야에서 신속한 채용되고 있다. PCI 버스 규격은 새로운 프로세서 기술 및 프로세서 속도의 증가에 독립적인 유연성과 고 대역폭을 제공한다.

현재, 컴퓨터 시스템 설계자는 우선적으로 PCI 버스와 함께 이용될 수 있는 그래픽 엑셀러레이터 및 SCSI디스크 드라이브 제어기와 같은 속도민감성(speed sensitive) 주변장치를 설계한다.

오늘날 PCI버스능력을 채용하도록 설계되는 컴퓨터시스템은 또한 기존의 다양한 주변장치와의 호환성을 위하여 상기 언급한 것과 같은 저속의 이차 입력/출력(I/O)버스를 포함한다. PCI버스 규격과 기존 입력/출력 버스규격 사이에는 많은 차이점이 있고, 특히 다양한 인터럽트 프로토콜면에서 더욱 그러하다.

예를들면, EISA/ISA버스 아키텍처는 각각이 다른 이용을 위하여 설계되는 16개의 인터럽트신호(IRQ[0:15])를 제공한다. 한편, PCI버스는 각각이 PCI버스상의 모든 슬롯에 의하여 공유되는 4개의 공유 인터럽트(INTR[A:D])를 제공한다.

이 두 버스규격은 그 지정된 인터럽트의 수나 명칭면에서 보다도 인터럽트 프로토콜면에서 더욱 상이하다. PCI 버스와 ISA 버스 모두를 갖춘 컴퓨터 시스템 아키텍처내에서 인터럽트를 조정하는데 있어서 더욱 어려운 점은, PCI 인터럽트는 활성 로우 레벨민감성 인터럽트가 되도록 정의되는데 반하여, ISA버스규격은 로우-하이 에지민감성 인터럽트가 되도록 정의한다는 것이다. PC AT인터럽트 구조는 주/종(master/slave)인터럽트 제어기쌍을 통하여 16개의 활성 에지민감성 인터럽트를 지원한다.

이러한 16개의 인터럽트중에서, AT 시스템은 최소한의 구성을 위하여 5개를 전용하고, 6개는 플로피 드라이브, IDE 드라이브, 2개의 직렬포트, 병렬포트 및 PS/2마우스 지원을 가지는 통상적인 시스템 환경을 위하여 이용되고, 나머지 5개는 애드-인 카드로 이용될 수 있도록 남겨둔다.

PCI버스 규격은 지원된 각각의 슬롯을 위한 4개의 활성 로우 레벨 민감성 인터럽트 핀을 요구하고, 하드웨어 공유가 가능하도록 그들의 이용을 정의한다. 이것의 의미는, 복수의 PCI장치가 동일한 인터럽트 선을 구동할 수 있거나 복수의 PCI 인터럽트 선이 다른 장치에 의하여 구동될 수 있지만, 공유 인터럽트 드라이버에 의하여 제공되도록 시스템 인터럽트제어기에게 발생하는 단일 인터럽트로 될 수도 있다.

PCI 기반 애드-인 카드에 부가하여 결합된 PC AT 시스템 내의 가능한 인터럽트수가 작음으로 인하여, 인터럽트 공유기술은 분명히 필요하다는 것으로 파악되고 있다.

이것은 이점을 갖고 있고, 그래서 본 발명의 목적은 향상된 컴퓨터 시스템의 운영을 위하여 비유사 인터럽트 프로토콜을 하나의 컴퓨터 시스템에 통합시키기 위한 메카니즘을 제공하는 것이다.

[발명의 요약]

컴퓨터 시스템의 성능이 다양한 인터럽트 프로토콜에 따라서 발생하는 인터럽트의 처리를 위한 메카니즘을 제공하므로써 향상된다는 것을 이해될 수 있다. 따라서, 본 발명의 목적은 비유사한 인터럽트 프로토콜에 대응하여 발생한 인터럽트를 통합하기 위한 방법 및 장치를 제공하는 것이다. 특히, 본 발명의 목적은 이차 입력/출력 버스를 가지는 시스템내에서 PCI 버스 규격을 준수하는 인터럽트 및 이차 입력/출력버스상에서 에이전트에 의하여 발생하는 인터럽트를 처리하기 위하여 기존에 존재하는 인터럽트를 통합하기 위한 레벨-에지 변환 메카니즘을 제공하는 것이다. 본 발명의 이들 목적 및 다른 목적은 PCI 버스상에서 에이전트에 의해 발생한 인터럽트를 처리하기 위한 인터럽트 변환회로를 도입하므로써 성취된다.

본 발명의 일실시예에서, 컴퓨터 시스템은 일차 입력/출력버스로서 PCI 버스 및 이차 입력/출력 버스로서 ISA 버스를 가지고서 구현된다. 프로그램가능 논리회로 장치는 PCI 버스상에서 에이전트에 의해 발생하는 인터럽트를 수신하도록 결합된다. 프로그램가능 논리회로 장치는 PCI 규격을 준수하는 활성로우 레벨 민감성 인터럽트를 ISA 버스 규격에 의하여 정의되는 로우-하이 에지 민감성 인터럽트로 변환시키는 데에 이용된다. PCI 에이전트 인터럽트는 적절한 인터럽트 프로토콜로 변환된 후, 미처리 인터럽트임을 CPU에게 알리기 위하여 시스템 상주 인터럽트 제어기에게 제공된다.

본 발명의 다른 실시예에서는 또한 시스템의 인터럽트 제어기에게 제공되도록 특별한 ISA IRQ 인터럽트를 선택하기 위한 소프트웨어(통상 BIOS) 기록가능 레지스터를 포함한다. 이 제어 레지스터는 PCI 버스로부터 인터럽트 제어기로 인터럽트를 경로배정하는데 사용되는 소망의 IRQ선을 선택하기 위하여 선택제어선을 인터럽트 변환 로직에게 제공한다.

이것은 프로그램가능 논리회로 장치로부터 인터럽트 제어기로의 단일 인터럽트 신호의 이용을 용이하게 한다.

[발명의 상세한 설명]

본 발명은 다양한 인터럽트 프로토콜을 따르는 복수의 입력/출력 버스를 가진 컴퓨터 시스템내에서의 인터럽트 처리를 위하여 제공되는 방법 및 장치이다. 이 상세한 설명란에서 PCI 버스 및 ISA 버스 모두를 포함하는 컴퓨터 시스템의 일 실시예를 설명한다.

하지만, 본 발명은 다양한 인터럽트 프로토콜을 따르는 복수의 버스를 이용하는 것이 바람직한 다른 컴퓨터 시스템내에서도 실행될 수 있기 때문에, 이것은 예시적 목적으로 설명한 것이고 제한적으로 제공된 것이 아니라는 것이 이해되어야 한다. 이 상세한 설명란 전체를 통하여, 많은 세부사항은 본 발명의 충분한 이해를 제공하기 위하여 특별한 신호명칭, 버스프로토콜 및 논리회로 장치 유형등으로 기재된다. 당해 기술분야의 평균적 전문가라면, 본 발명이 이러한 세부사항이 없이도 실시될 수 있다는 것은 인식할 것이다.

한편, 주지의 구성장치, 구조 및 기술은 본 발명의 요지를 불확실하게 하는 것을 방지하기 위하여 상세히 설명하지 않는다. 또한, 이 상세한 설명란에서 신호명칭은 주어진 신호의 활성상태(즉 활성로우 또는 활성하이)만을 의미하도록 의도된 것이 아니고 단지 논의의 편의를 위하여 인터페이스 신호를 명명하는 데에 이용된다.

제1도를 참조하면, 본 발명을 구현하는 하나의 컴퓨터 시스템의 아키텍처가 도시되어 있다. 제1도의 컴퓨터 시스템의 아키텍처는 그 아키텍처의 계층적 버스조직을 강조하도록 도시되어 있다. 컴퓨터 시스템의 중앙처리장치(CPU; 10)는 호스트 버스(20)를 통하여 그것의 SRAM 캐쉬(15)와 통신하고 있음을 알 수 있다. 일 실시예에서, 호스트 버스는(20) 제어선(21), 주소선(22) 및 데이터선(23)을 포함한다. 다른 실시예에서, 데이터선 및 주소선은 공통신호경로를 공유하도록 멀티플렉싱될 수도 있다. 호스트 버스는 통상, CPU(10)를 위하여 구현되는 프로세서의 타입이 어떤 것이든지, 그에 대응하여 고도의 연산 능력의 요구를 만족시키도록 설계된다.

어떤 컴퓨터 시스템의 아키텍처에서, 컴퓨터 시스템의 주메모리는 프로세서 호스트버스에 직접 결합될 것이다. 예시된 구현에서, DRAM 주메모리(18)는 호스트/PCI 버스 브릿지(25)를 통하여 호스트버스에 결합된다. 호스트/PCI 버스 브릿지(25)는 호스트 버스(20)와 시스템의 PCI버스(30)사이에서 브릿지로서 이용된다.

상기 PCI버스는 컴퓨터 시스템을 위한 일차 입력/출력 버스로서 역할을 하고, 컴퓨터 프로세서 버스의 속도 및 크기에 직접 의존하지 않는 표준 로컬버스를 위한 증가하는 업계의 요구에 부응하도록 설계되었다. 제1도에 도시된 것처럼 PCI버스(30)에는 제어선(31) 및 주소/데이터 선(32)이 분리되어 있다. 호스트/PCI 버스 브릿지(25)는 호스트 버스(20) 및 PCI버스(31,32)를 감시하고, 트랜잭션이 컴퓨터 시스템의 주메모리에 대하여 지정되었는지의 여부 내지 다음 단계가 PCI버스(30)로 전파되는 CPU의 요구가 버스계층구조를 따라서 하향전파되어 나가야 하는지의 여부를 판단한다.

제1도에서는 PCI버스(30)에 결합되고 PCI버스 규격을 준수하도록 설계된 주변장치를 수용하기 위한 3개의 PCI슬롯(36,37,38)이 도시되어 있다. 물론 컴퓨터시스템은 어떠한 개수의 PCI슬롯도 채용할 수 있으며, 본 발명에서도 그러하다. 이러한 애드-인 카드에는 그래픽 가속기, 디스크 드라이브 제어기, 및 다른 속도 민감성 주변장치가 포함될 수도 있고, 그것들은 PCI로컬 버스(30)의 기능으로부터 이점을 얻을 수 있다. 또한 PCI버스의 주소/데이터 선(32)과 호스트 버스 사이에는 두 버스(20,30)사이에서의 데이터 전송을 버퍼하기 위하여 이용되는 LBX버퍼(34)로 명명된 버퍼집단이 도시되어 있다. LBX버퍼(34)는 두 버스가 동작되는 속도의 가변성을 보상하기 위하여 구현되며, 각각의 버스처리능력의 증가와, 대기 시간의 감소를 제공하는, 호스트버스(20)와 PCI버스(30) 사이에서의 동시 처리기능을 허락함으로써 시스템의 성능을 향상시킨다. 대안 실시예에서, 호스트버스, 주메모리 및 일차 입력/출력 버퍼 사이에서의 데이터 전송기능이 유지되는 한, LBX 버퍼는 다른 버퍼 메카니즘으로 대체될 수도 있다.

마지막으로 제1도에는 대체 실시예에서 ISA 규격을 준수하는 주변장치 입력/출력 버스인 이차 입력/출력 버스(40)가 도시되어 있다. 본 발명의 개시내용은 물론 EISA버스, 내지 PCI버스와는 상이한 인터럽트 프로토콜을 갖는 다른 버스와 같은 다른 이차 입력/출력버스를 가지는 컴퓨터 시스템에도 확장된다.

이 ISA버스(40)에 결합된 4개의 입력/출력슬롯(41,42,43,44)은 다양한 주변장치를 수용하기 위하여 사용될 수 있다. 플로피 디스크 드라이브, IDE규격 상호접속을 갖춘 하드 디스크 드라이브 및 컴퓨터 시스템과 통상적으로 접속되는 다른 주변장치와 같은 보드상의 주변장치(45)는 종종 시스템의 마더보드 위에 직접 설계되기도 하고, 다른 애드-인 카드의 업그레이드를 위하여 제공되는 입력/출력슬롯을 필요로 하지 않는다. 보드상의 주변장치(45)는 전기적으로 이차 입력/출력버스(40)상에 존재한다. 일반적으로, 이차 입력/출력버스(40)는 PCI버스(30)보다 저속이지만, 종래의 주변장치와의 호환성을 유지하기 위하여 새로운 컴퓨터 시스템에서도 여전히 포함된다.

ISA버스와 PCI버스 모두를 가지므로써 일차 입력/출력버스인 PCI버스를 따라 설치된 개인용 컴퓨터 플랫폼은 광범위한 ISA제품 기준을 달성할 수 있게 된다. ISA버스 24비트 주소지정 및 16비트 데이터 경로를 제공한다. EISA 입력/출력버스는 일차 입력/출력버스로서 PCI버스를 따라 설치된 개인용 플랫폼이 광범위한 EISA/ISA제품 기준을 달성할 수 있게 한다.

16비트 및 8비트의 ISA하드웨어 및 소프트웨어에 대한 호환성을 가지는 EISA버스는 32비트 주소 지정 및 32비트 데이터 경로를 위하여 제공된다. 물론, 다른 이차 입력/출력버스도 마찬가지로의 특성을 제공할 수도 있다.

제1도의 컴퓨터 아키텍처의 PCI버스(30)와 이차 ISA버스(40)사이에서 결합되는 시스템 입력/출력 버스(SIO)구성장치(35)가 도시되어 있다. SIO구성장치(35)는 PCI버스와 ISA 버스 사이에 브릿지를 제공하고, 어떠한 필요적 DMA제어기 및 인터럽트 제어논리에 대한 지원이라도 채용할 뿐만아니라, PCI버스(30)와 ISA 버스(40)를 인터페이스시키기 위한 논리를 채용할수도 있다. 예를들면, 만약 IDE상호접속을 가지는 종래의 하드디스크 드라이브가 입력/출력슬롯(41)에 존재하고 IDE 드라이브가 CPU와 함께 트랙잭션을 요구한다면, IDE 드라이브는 시스템에 의한 일차 하드디스크 드라이브 접근으로서 인식될 인터럽트(IRQ14)를 신호할 것이다. 컴퓨터 시스템이 IRQ14인터럽트를 인식할 때에, 그것은 하드디스크 드라이브를

접근하기 위해 필요한 코드로 백터링할 것이다.

제1도에 따라서 구현된 컴퓨터 아키텍처에서, SIO 구성장치(35)는 ISA버스 인터럽트를 준수하는 인터럽트에 응답하도록 구성된 인텔 82C59인터럽트제어기와 같은 인터럽트 제어기를 포함한다. SIO 구성장치(35)는 프로그래밍 가능한 중재기법을 통하여 다양한 소스로부터의 인터럽트를 해결할 것이고, 적절한 인터럽트 서비스 루틴을 실행하는 동작을 하도록 CPU에게 단일 인터럽트를 제시할 것이다.

설명한 것처럼, PCI버스상에서 PCI슬롯에 상주하는 에이전트는 4개의 PCI지정 레벨 민감성 인터럽트를 공유한다. 제1도에 나타난 컴퓨터 아키텍처의 구성에서, 각각의 PCI슬롯은 PCI에이전트가 인터럽트 PAL(50)로 식별된 프로그램가능 논리장치에 그들의 인터럽트를 제공하도록 결합된다. 인터럽트 PAL(50)의 구성 및 프로그래밍은 여기에서 상세히 설명될 것이다. 본질적으로 인터럽트 PAL(50)의 기능은 활성로우 레벨 민감성 인터럽트를 ISA버스 규격에 의하여 요구되는 로우-하이 에지 민감성 인터럽트로 변환시키는 것이다. 인터럽트 PAL(50)는 모든 또는 임의의 PCI에이전트에 의하여 요구된 PCI인터럽트에 응답하여 SIO 구성장치(35)에 에지 민감성 인터럽트를 제공한다.

IRQ신호가 SIO 구성장치(35)에게 ISA 인터럽트를 신호하는데 이용될 수도 있다는 것은 앞에서 언급하였다. 그래서 SIO 구성장치(35)에 제공되도록 단일 IRQ 신호를 신호하는 인터럽트 PAL(50)에 대한 방향지시를 위한 선택선을 제어하는 소프트웨어(통상 BIOS)기록 가능 제어 레지스터(48)가 제공된다. 일 실시예에서, 선택선은 10주소(73h)의 하위 2비트에 의하여 제어된다. 다른 레지스터 또는 래치는 다른 10 또는 메모리위치에 상주할 수도 있고 저장장치 구성요소(48)를 위하여 이용될 수도 있다. 제어 레지스터(48) 또는 인터럽트 PAL(50)의 내부동작은 여기에서 더욱 상세히 설명될 것이다.

제2도를 참조하면, 본 발명의 일 실시예에서 이용되는 인터럽트 PAL(50)의 더욱 상세한 구성이 도시되어 있다. 본 발명의 일 실시예에서, 프로그램 가능 논리는 인텔사에 의하여 제조된 85C220 EPLD를 포함할 수도 있다. 물론, 이하 설명될 기능을 수행할 수 있는 다른 프로그램가능 논리도 다른 구현을 위하여 적합하다는 것이 판명되었다. PAL(50)은 PCI버스(30)에 결합된 PCI에이전트로부터 모든 공유 인터럽트를 수신한다. 조합된 공유 인터럽트는 함께 게이트 논리(55)에 의하여 OR연산되고, 그 결과는 INT로 명명된 신호로서 상태기(60)에 제공된다. 그래서, 어떠한 인터럽트를 신호하는 어떠한 PCI에이전트라도 PCICLK신호에 의하여 클럭되는 상태기(60)에 대하여 INT신호를 활성으로 되도록 하게 할 것이다.

프로그램가능 논리블록(50)의 상태기 논리(60)는 PCI 에이전트 인터럽트들에 대한 논리 OR 및 PCICLK타이밍 신호를 수신할 뿐만 아니라, 트랜잭션의 상태기에서 INTA로 식별된 SIO구성장치(35)부터의 INTR신호 및 시스템 리세트신호를 또한 수신한다. 일 실시예에서, EPLD는 ISA 버스 전체를 리세트시키는 동일한 신호에 의하여 핀(8)상에서 리세트된다. CPU에게 SIO구성장치(35)로부터 제공되는 INTR신호는 CPU가 인터럽트 서비스 루틴(ISR)을 처리하도록 하게 하는 SIO에 의하여 발생하는 인터럽트 요구이다.

ISR의 수행 완료시, CPU는 SIO의 인터럽트 제어기 인터페이스에 "인터럽트 종료" 명령을 기록하므로써 INTR신호를 소거한다.

상태 머신 논리(60)에는 또한 선택제어신호(SELO, SEL1)가 제공된다. 도시된 실시예에서, PCI에이전트 인터럽트는 IRQ선(IRQ9, IRQ10 또는 IRQ11)중의 하나를 통하여 SIO구성장치(35)로 전송된다(제1도 참조). 어떤 IRQ신호가 SIO구성장치(35)에 제공될 것인가에 관한 선택은 시스템 구성 소프트웨어에 의하여 기록가능한 선택 레지스터(48)에 세트된 비트에 의하여 판단된다(제1도 참조). 물론, 다른 IRQ신호도 SIO구성장치(35)에 PCI인터럽트를 지시하기 위하여 사용될 수도 있다. 주목할 것은 만약 401상의 가능한 IRQ가 제공된다면, 201상의 비트가 SEL레지스터(48)내에서 세트되도록 요구될 것이고, PAL(50)에 추가 선택선이 요구될 것이다. 상태기(60)에 제공되는 이외에 선택제어비트는 또한 ISA IRQ 선택 멀티플렉서(52)에 제공된다. 상태기 논리(60)가 선택 멀티플렉서(52)에 제어신호로서 활성 IRQ신호를 송신하는 때에, 선택멀티플렉서(52)는 선택비트 및 제어신호에 응답하여 SIO구성장치(35)에게 그 선택된 IRQ신호를 제공한다.

일 실시예에서, 이들 비트의 일 상태는 SIO에 대한 아무런 IRQ출력도 일어나지 않았다는 것을 인코딩하는 데에 이용된다. 그것은 SEL REG(48)의 리세트상태이고, 그래서 RESET에서 그 비트들은 구성될 수도 있고 또는 아무런 IRQ도 PCI에이전트로부터 처리되지 않을 것이다.

제3도는 제2도에 도시된 상태기논리(60)의 일 실시예에 의하여 구성된 상태도를 나타낸다.

상태도는 리세트하는 때에 유희상태(300)에서 시작한다. 이 예에서 상태기는 동기적이고 PCICLK 상승 에지에서 상태에서-상태로 전이한다. INT신호가 비활성인 한, 상태기는 아무런 PCI인터럽트도 어떠한 PCI에이전트에 의하여 표명되고 있지 않다는 것을 지시하면서 유희상태를 유지한다.

INT신호가 상태기(60)에 대하여 활성으로 되는 때에, 그것은 다음 PCICLK입력에서 N1상태(310)로 전이한다. 일단 상태기가 N1상태(310)를 획득하면, 그것은 SIO구성장치(35)에 그 선택된 IRQ를 제공하기 위하여 ISAIRQ선택멀티플렉서(52)에게 IRQ 활성신호를 표명한다. 만약 상태기 논리가 N1상태(310)에 있을 때에 INTA신호가 이미 활성이라면, 상태기는 N1상태를 유지할 것이다. 상태기가 N1상태를 처음 획득한 때에, 활성 INTA신호는 CPU가 이전에 요구된 인터럽트 서비스 루틴을 현재에 실행하고 있음을 지시한다. 상태기는 CPU가 이전에 신호된 인터럽트의 처리를 완료하였다는 것을 지시하는, INTA신호가 비활성으로 될 때까지 N1상태에서 N2상태로 전이하지 않을 것이다.

INTA신호가 N1상태에 있는 때에 비활성으로 된다면, 상태기는 N1상태(310)에서 다시 유희상태(300)로 전이할 수도 있다. 이것은 인터럽트를 요구하는 PCI에이전트가 CPU에 의하여 처리되는 인터럽트를 더 이상 요구하지 않는다는 것을 지시한다.

이것은 기대되지는 않지만, 일어날 수는 있다. 부가적으로, 본 발명의 도시된 실시예에서, 만약 SEL1 및 SEL0 선택비트 모두가 비활성으로 되면, 상태기는 유희상태로 복귀할 것이다. 이것은 아무런 PCI 인터럽트도 승인될 수 없다라고 운영체제에 의하여 만들어지는 선택을 지시하는 것 일 수도 있다. 대체 실시예에서, 널(null)SEL1 및 SEL0의 값은 특별한 IRQ신호 및 디폴트 IRQ신호를 지시하도록 정의될 수도

있다.

일단 CPU가 INTA신호를 소거한 경우, 그것이 활성이었다면, 상태기는 N1상태(310)으로 N2상태(320)로 다음 PCICLK 입력에서 천이한다. 이 상태에서, 상태기 논리에 INT신호가 활성으로 유지되거나 선택신호가 비활성으로 되지 않는 한, IRQ신호는 프로그램가능 논리회로 장치(50)로부터 계속하여 구동된다. CPU가 어떠한 인터럽트 요구에도 응답하지 않고 있다는 것을 지시면서 INT 신호가 비활성으로 유지되는 한, 상태기는 N2상태(320)로 유지된다. INTA 신호가 다시 활성으로 되는 경우에, 그것은 비록 CPU가 다른 인터럽트에 응답하고 있을 가능성이 있더라도 PCI에이전트에 의하여 발생하는 인터럽트일수 있는 인터럽트에 CPU가 응답하고 있다는 것을 지시한다. INTA 신호가 활성으로 되는 경우에, 상태기는 INTA신호가 활성으로 남아 있는 한 그것이 유지하는 상태 즉 N3상태(330)로 천이한다. 상태기는 N3상태에 있는 때에 ISA IRQ선택 멀티플렉서(52)에게 계속하여 IRQ신호를 제공한다.

SEL1 및 SEL0 선택제어의 기동중지는 상태기를 유효상태로 복구하도록 하게 하고, 만약 그렇지 않으면 상태기는 INTA신호가 활성인 한 N3상태(330)를 유지할 것이다.

INTA신호가 다시 비활성으로 되는 때에 그것은 비록 필수적이지만 않지만 PCI에이전트가 인터럽트를 신호하므로써 요구되는 ISR로 될 수도 있었을 인터럽트 서비스 루틴(ISR)의 실행을 CPU가 완료하였다는 것을 지시한다. 다음, 상태기는 PCICLK타이밍 신호의 일 사이클에서 각각의 상태에 상주하는 R1상태(340), R2상태(350) 및 R3(360)를 통하여 천이할 것이다. 이 주기동안에, IRQ신호는 더 이상 활성이 아니고, S10구성장치(37)로 인터럽트 PAL(50)로부터 IRQ를 수신하지 않을 것이다.

R3상태(360)로부터, 상태는 유희상태(300)로 복귀한다. 만약 CPU가 PCI에이전트를 위한 ISR을 실행하였다면, 그 프로시저는 PCI에이전트의 인터럽트요구를 소거해야 만하고, 그 후 INT신호가 비활성으로 될 가능성이 있다. 하지만, 만약 상태기가 N3상태(330)에 있었을 때에 CPU가 다른 인터럽트에 응답하였다면, 그것은 INTA 신호를 비활성으로 되게 하는 다른 ISR이 되었을 것이다. 그러한 경우에, INT신호는 여전히 상태기(60)에 대한 활성입력이라는 것으로 기대되고, 여기서 상태기는 다음 PCICLK입력에서 상기 언급한 과정을 반복하도록 유희상태(300)로부터 N1상태(310)로 천이할 것이다. N3상태(330)에 후속하는 R1, R2 및 R3상태(340,350,360)를 통과하고, 다시 유희상태(300)를 통과하여, N1상태(310)로 가는 일련의 상태는 필수적인 지연을 제공한다. 이것은 인터럽트 PAL(50)에 의하여 수신되는 PCI준수 레벨민감성 인터럽트를 ISA IRQ준수 신호로서 S10구성장치(35)의 인터럽트 제어기로 구동되는 로우-하이 에지 민감성 인터럽트로 변화시키는 인터럽트 PAL(50)로 부터의 IRQ신호의 지연과 스위칭 오프 및 백온이다.

제4도는 상태기 논리(60)의 일 실시예에 의하여 구현되는 더욱 일반적인 상태도이다. 제4도에 도시된 상태도는 제3도에 관하여 상기 설명된 상태(300,310,320,330)를 포함한다. 제4도에는 R1, R2 및 R3 상태가 지연상태(400)로 대체되어 있다. 제3도의 특별한 구현체는 S10구성장치(35)에 하이-로우 에지민감성 인터럽트를 적절히 위치시키는 것을 보장하기 위하여, 유희상태(300)으로 복귀하기 전에 3개 PCI 클럭 신호의 지연을 요구한다.

어떤 실시예에서는 다른 지연주기가 요구될 수도 있고, 그것은 제4도에서 채용된 지연상태(400)에 포함되어 설명될 수도 있다.

하나 이상의 구현된 입력/출력버스를 가지는 컴퓨터 시스템에서 비유사한 인터럽트 프로토콜을 조화롭게 만들기 위한 기술이 설명되었다. 비록 본 발명이 여러 가지 실시예를 통하여 설명되었지만, 본 발명은 그것의 사상과 범위를 벗어나지 않고도 다양한 변경 및 변형이 만들어질 수 있다는 것은 당해기술 분야의 전문가에게 인식될 것이다. 예를들면, PCICLK와 다른 클럭을 이용하는 프로그램가능 논리설계의 구현이 가능할 것이다. 마찬가지로 비동기식 구현도 고안할 수 있다.

(57) 청구의 범위

청구항 1

중앙처리장치(CPU); 상기 CPU로부터 및 CPU로 신호를 전달하기 위해 상기 CPU에 연결된 호스트 버스; 상기 호스트 bus와 연결되고, CPU와 주변장치사이에 통신경로를 제공하기 위해 제1버스규격을 준수하는 제1 I/O 버스로서, 제1 I/O 버스를 통하여 전파되는 인터럽트는 제1 인터럽트 유형인, 상기 제1 I/O 버스; 제2 버스규격을 준수하는 제2 I/O 버스로서, 제2 I/O 버스를 통하여 전파되는 인터럽트는 제2 인터럽트 유형인, 상기 제2 I/O 버스; CPU와 제2 I/O 버스에 연결되고, 제2 인터럽트 유형의 인터럽트에 응답하며 인터럽트를 CPU에 신호전달하는 인터럽트 제어기; 및 상기 인터럽트 제어기에 연결되고, 상기 제1 I/O 버스의 에이전트에 의해 발생된 제1 인터럽트 유형의 인터럽트에 응답하고, 제1 인터럽트 유형의 인터럽트 수신에 응답하여 제2 인터럽트 유형의 인터럽트를 인터럽트 제어기에 제공하는 인터럽트 컨버터를 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 2

제1항에 있어서, 상기 제1 인터럽트 유형의 인터럽트는 레벨 민감성 인터럽트이고 상기 제2 인터럽트 유형의 인터럽트는 에지 민감성 인터럽트인 것을 특징으로 하는 컴퓨터 시스템.

청구항 3

제2항에 있어서, 상기 인터럽트 컨버터는, 제1 I/O 버스상의 복수의 에이전트로부터 복수의 인터럽트선에 대한 논리 OR연산을 수행하기 위한 회로; 및 상기 논리 OR연산을 수행하기 위한 상기 회로에 연결되고 상기 회로에 응답하며, 에지 민감성 인터럽트 신호를 출력시키기 위한 상태머신 회로를 포함하는 것을 특징으로 하는 컴퓨터 시스템.

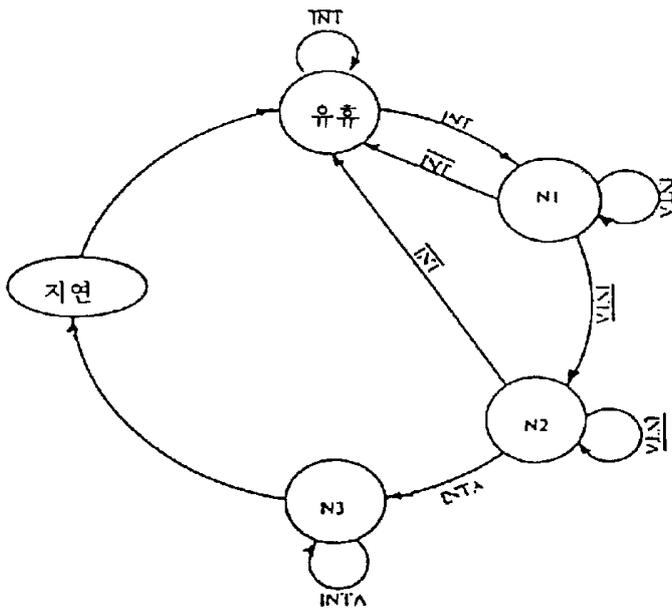
청구항 4

제3항에 있어서, 상기 상태머신은, 상태머신이 에지 민감성 인터럽트 신호를 출력하는 제1상태와 어떠한 인터럽트 신호도 출력하지 않는 제2상태를 포함하는 복수의 상태에서 동작하는 것을 특징으로 하

는 컴퓨터 시스템.

청구항 5

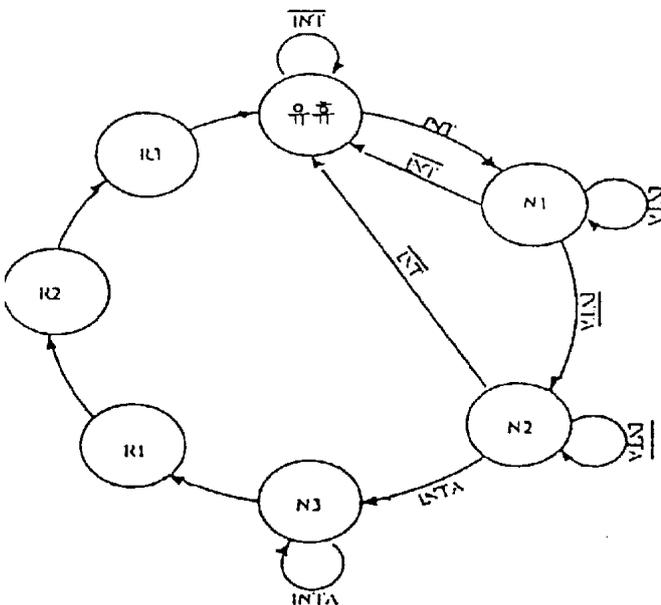
제3항에 있어서, 상기 상태머신은 다음의 상태도에 따라 동작하고,



상기 상태머신이 N1, N2 및 N3상태에 있는 때에 상기 상태머신 회로는 에지 민감성 인터럽트 신호를 출력시키고, 상기 상태머신이 지연(Delay)상태에 있을 때에 상기 상태머신회로는 아무런 인터럽트 신호도 출력시키지 않고, INT신호는 제1 I/O 버스상의 에이전트로부터 상기 인터럽트선에 대한 상기 논리 OR연산의 활성결과를 포함하고, 상기 INTA신호는 인터럽트를 처리하고 있다는 것을 지시하는 상기 CPU로부터의 신호를 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 6

제3항에 있어서, 상기 상태머신은 다음의 상태도에 따라 동작하고,



상기 상태머신이 N1, N2 및 N3 상태에 있는 때에 상기 상태머신 회로는 에지 민감성 인터럽트 신호를 출력시키고, INT신호는 제1 I/O 버스상의 에이전트로부터의 상기 인터럽트선에 대한 상기 논리 OR연산의 활성결과를 포함하고, INTA 신호는 인터럽트를 처리하고 있다는 것을 지시하는 상기 CPU로부터의 신호를 포함하고, R1, R2 및 R3상태는 상기 상태머신 회로가 아무런 인터럽트 신호도 출력하지 않는 연속적인 상태머신 상태를 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 7

제3항에 있어서, 상기 인터럽트 컨버터는, 상기 에지민감성 인터럽트 신호를 상기 인터럽트 제어기의 선택된 인터럽트 입력으로 경로배정하기 위해 상기 상태머신 회로에 연결된 멀티플렉싱 로직을 더 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 8

제7항에 있어서, 상기 에지 민감성 인터럽트를 상기 선택된 인터럽트 입력으로 보내는 상기 멀티플렉싱 로직을 제어하기 위해 상기 멀티플렉싱 로직에 연결된 프로그램가능 저장 엘리먼트를 더 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 9

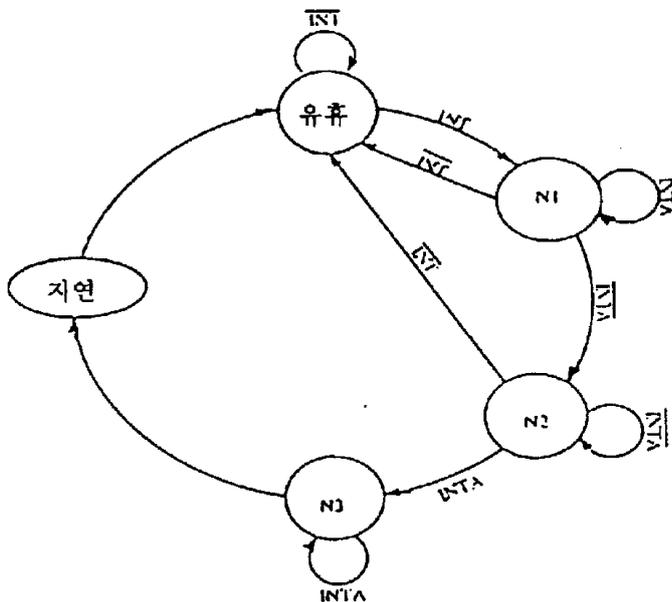
제1항에 있어서, 상기 제1버스 규격은 주변장치 상호연결(PCI)규격을 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 10

CPU, 레벨 민감성 인터럽트를 발생시키는 제1 주변장치 에이전트를 갖는 제1 I/O 버스, 에지 민감성 인터럽트를 발생시키는 제2 주변 장치 에이전트를 갖는 제2 I/O 버스를 갖고, 에지 민감성 인터럽트에 응답하는 인터럽트 제어기를 포함하는 컴퓨터 시스템에서 사용하기 위한, 레벨 민감성 인터럽트를 에지 민감성 인터럽트로 변환하는 레벨 민감성 인터럽트-에지 민감성 인터럽트 컨버터에 있어서, 제1 주변장치 에이전트에 의해 발생된 인터럽트 신호에 대한 논리 OR연산을 수행하는 회로; 및 레벨 민감성 인터럽트신호가 검출되었을 때에 에지 민감성 인터럽트 신호를 상기 인터럽트 제어기로 출력시키기 위해 논리 OR연산을 수행하기 위한 상기 회로에 연결되고 상기 회로에 응답하는 논리회로를 포함하는 것을 특징으로 하는 레벨 민감성-에지 민감성 인터럽트 컨버터.

청구항 11

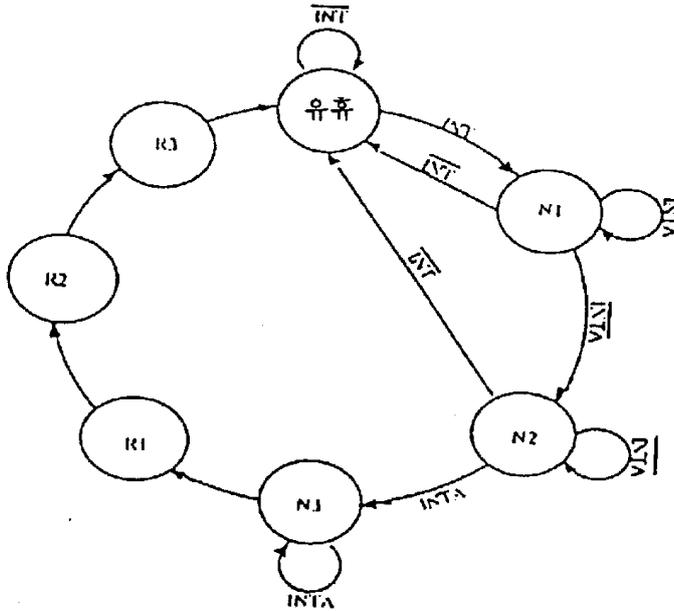
제10항에 있어서, 상기 논리회로는 다음의 상태도에 따라 동작하는 상태머신을 포함하고,



상기 상태머신이 N1, N2 및 N3 상태에 있는 때에 상기 상태머신 회로는 에지 민감성 인터럽트 신호를 출력시키고, 상기 상태머신이 지연(Delay)상태에 있을 때에 상기 상태머신 회로는 아무런 인터럽트 신호도 출력시키지 않고, INT신호는 제1 주변장치 에이전트로부터의 상기 인터럽트선에 대한 상기 논리 OR연산의 활성결과를 포함하고, INTA 신호는 인터럽트를 처리하고 있다는 것을 지시하는 상기 CPU로부터의 신호를 포함하는 것을 특징으로 하는 레벨민감성-에지민감성 인터럽트 컨버터.

청구항 12

제10항에 있어서, 상기 논리회로는 다음의 상태도에 따라 동작하는 상태머신을 포함하고,



상기 상태머신이 N1, N2 및 N3상태에 있는 때에 상기 상태머신 회로는 에지 민감성 인터럽트 신호를 출력시키고, INT신호는 제1 주변장치의 에이전트로부터의 상기 인터럽트선에 대한 상기 논리 OR연산의 활성결과를 포함하고, INTA 신호는 인터럽트를 처리하고 있다는 것을 지시하는 상기 CPU로부터의 신호를 포함하고, R1, R2 및 R3상태는 상기 상태머신 회로가 어떠한 인터럽트 신호도 출력하지 않는 연속적인 상태머신 상태를 포함하는 것을 특징으로 하는 레벨민감성-에지민감성 인터럽트 컨버터.

청구항 13

제10항에 있어서, 상기 인터럽트 컨버터는, 상기 에지민감성 인터럽트 신호를 상기 인터럽트 제어기의 선택된 인터럽트 입력으로 경로배정하기 위해 상기 상태머신 회로에 연결된 멀티플렉싱 로직 및 상태머신 회로를 더 포함하는 것을 특징으로 하는 레벨민감성-에지민감성 인터럽트 컨버터.

청구항 14

제13항에 있어서, 상기 에지 민감성 인터럽트를 상기 선택된 인터럽트 입력으로 보내도록 상기 멀티플렉싱 로직을 제어하기 위해 상기 멀티플렉싱 로직에 연결된 프로그램가능 저장 엘리먼트를 더 포함하는 것을 특징으로 하는 레벨민감성-에지민감성 인터럽트 컨버터.

청구항 15

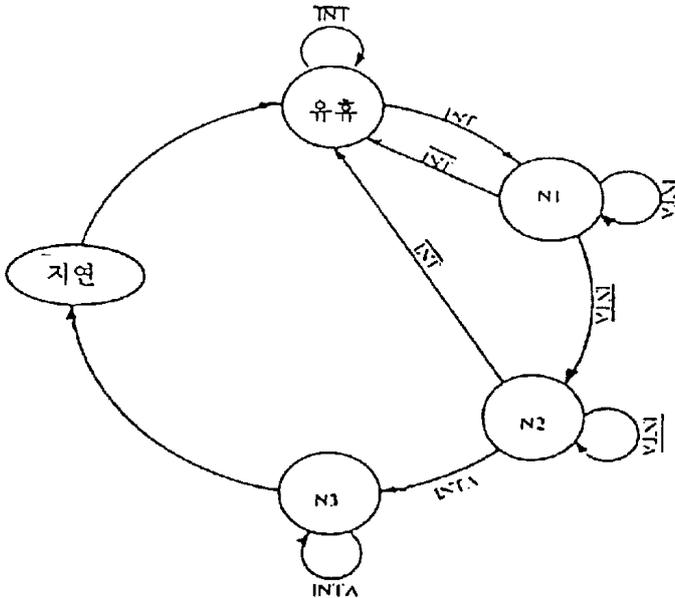
제10항에 있어서, 상기 논리회로는 에지 민감성 인터럽트 신호를 출력하는 제1상태와 어떠한 인터럽트 신호도 출력하지 않는 제2상태를 포함하는 복수의 상태에서 동작하는 상태머신을 포함하는 것을 특징으로 하는 컴퓨터 시스템.

청구항 16

CPU, 제1 인터럽트 유형의 인터럽트를 발생시키는 제1 주변장치 에이전트를 갖는 제1 I/O 버스, 제2 인터럽트 유형의 인터럽트를 발생시키는 제2 주변장치 에이전트를 갖는 제2 I/O 버스 및 제1 인터럽트 유형의 인터럽트에 응답하는 인터럽트 제어기를 갖는 컴퓨터 시스템에서 인터럽트를 처리하는 방법에 있어서, 상기 제2 인터럽트 유형의 인터럽트를 검출하는 단계; 검출된 상기 제2 인터럽트 유형의 인터럽트에 응답하여 상기 제1 인터럽트 유형을 준수하는 인터럽트 신호를 발생시키는 단계; 및 인터럽트 처리를 위하여 상기 인터럽트 제어기에 상기 제1 인터럽트 유형을 준수하는 상기 인터럽트 신호를 제공하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 17

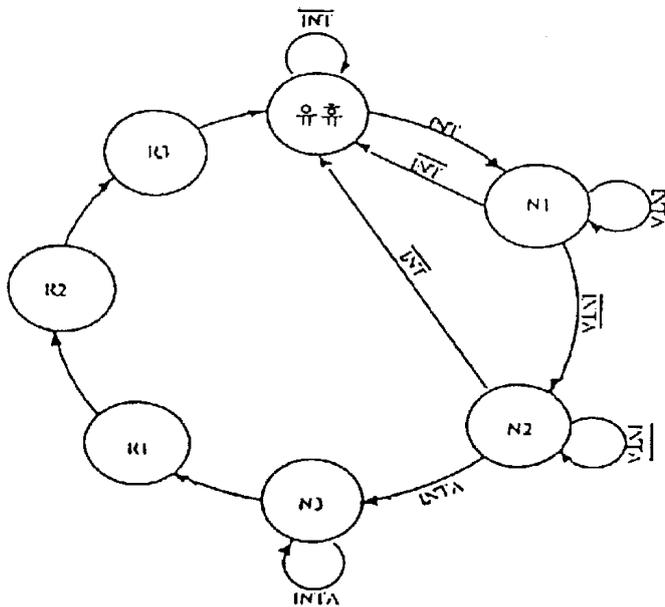
제16항에 있어서, 상기 제1 인터럽트 유형의 에지 민감성 인터럽트를 포함하고, 상기 제2 인터럽트 유형의 레벨 민감성 인터럽트를 포함하고, 상기 컴퓨터 시스템은 상태 머신을 포함하고, 상기 제1 인터럽트 유형을 준수하는 인터럽트 신호를 발생시키는 단계는 다음의 상태머신 모델을 따르는 단계를 포함하고,



상기 상태 머신이 N1, N2 및 N3상태에 있는 때에 상기 상태머신 모델은 예지 민감성 인터럽트 신호를 인터럽트 제어기에 출력시키고, 상기 상태머신이 지연(Delay)상태에 있을 때에 상기 상태머신 회로는 아무런 인터럽트 신호도 출력시키지 않고, INT신호는 상기 제2 주변장치 에이전트로부터 상기 인터럽트선에 대한 상기 논리 OR연산의 활성결과를 포함하고, INTA 신호는 인터럽트를 처리하고 있다는 것을 지시하는 상기 CPU로부터의 신호를 포함하는 것을 특징으로 하는 방법.

청구항 18

제16항에 있어서, 상기 제1 인터럽트 유형은 예지 민감성 인터럽트를 포함하고, 상기 제2 인터럽트 유형은 레벨 민감성 인터럽트를 포함하고, 상기 컴퓨터 시스템은 상태머신을 포함하고, 상기 제1 인터럽트 유형을 준수하는 인터럽트 신호를 발생시키는 단계는 다음의 상태머신 모델을 따르는 단계를 포함하고,



상기 상태머신이 N1, N2 및 N3 상태에 있는 때에 상기 상태머신 모델은 상기 인터럽트 제어기에 예지 민감성 인터럽트 신호를 출력시키고, INT신호는 상기 제2 주변장치 에이전트로부터 상기 인터럽트선에 대한 상기 논리 OR연산의 활성결과를 포함하고, INTA 신호는 인터럽트를 처리하고 있다는 것을 지시하는 상기 CPU로부터의 신호를 포함하고, R1, R2 및 R3상태는 상기 상태머신 회로가 아무런 인터럽트 신호도 출력하지 않는 연속적인 상태머신 상태를 포함하는 것을 특징으로 하는 방법.

청구항 19

제16항에 있어서, 상기 인터럽트 제어기로의 선택된 인터럽트 입력을 식별하는 제어비트를 저장하는 단계를 더 포함하는 것을 특징으로 하는 방법.

청구항 20

제19항에 있어서, 상기 인터럽트 제어기에 상기 인터럽트 신호를 제공하는 단계는 상기 제어 비

트에 응답하여 상기 선택된 인터럽트 입력에 상기 인터럽트 신호를 제공하는 단계를 더 포함하는 것을 특징으로 하는 방법.

청구항 21

중앙처리장치, 레벨 민감성 인터럽트를 발생시키는 에이전트를 갖는 제1 I/O 버스, 에지 민감성 인터럽트를 발생시키는 에이전트를 갖는 제2 I/O 버스를 갖고 에지 민감성 인터럽트에 응답하는 인터럽트 제어기를 포함하는 컴퓨터 시스템에서 사용하기 위한, 레벨 민감성 인터럽트를 에지 민감성 인터럽트로 변환하는 레벨 민감성 인터럽트-에지 민감성 인터럽트 컨버터에 있어서, 제1 I/O 버스로부터 인터럽트 신호를 수신하도록 구성된 회로; 상기 제1 I/O 버스로부터의 상기 인터럽트 신호를 에지 민감성 인터럽트 신호로 변환시키도록 구성된 변환회로; 및 상기 인터럽트 제어기에 상기 에지 민감성 인터럽트 신호를 제공하도록 구성된 회로를 포함하는 것을 특징으로 하는 레벨 민감성 인터럽트-에지 민감성 인터럽트 컨버터.

청구항 22

제21항에 있어서, 상기 레벨 민감성 인터럽트는 PCI 규격 준수 인터럽트 신호를 포함하고 상기 에지 민감성 인터럽트 ISA/EISA 규격 준수 인터럽트 신호를 포함하는 것을 특징으로 하는 레벨 민감성 인터럽트-에지 민감성 인터럽트 컨버터.

요약

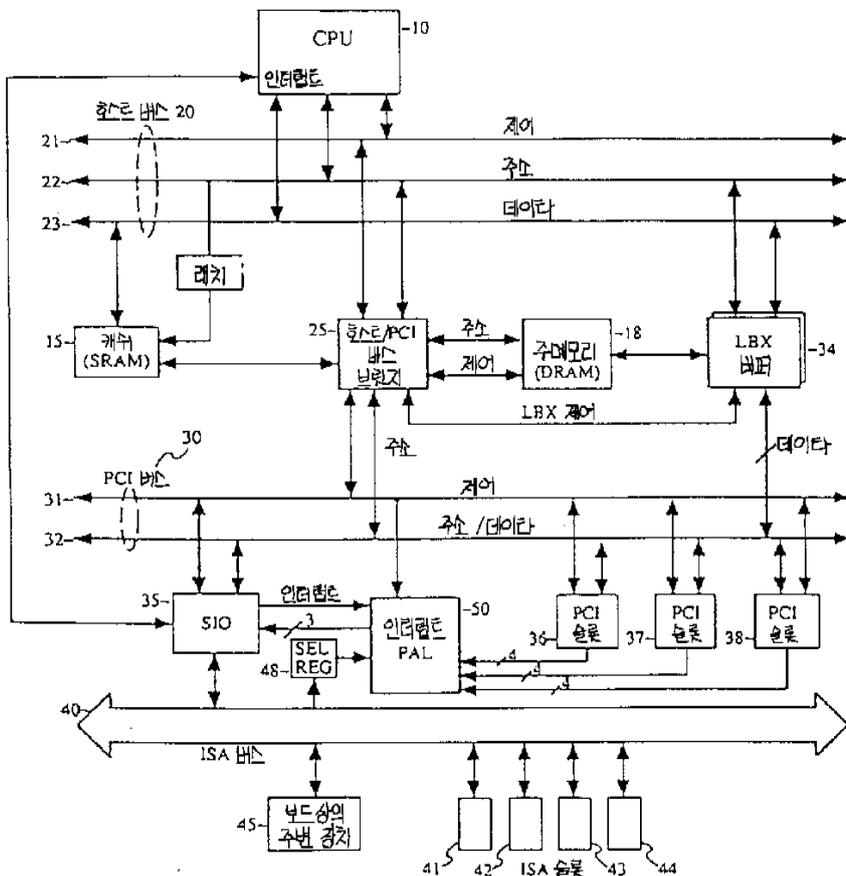
이차 버스 규격 인터럽트 프로토콜을 준수하는 PCI 에이전트 인터럽트(36 내지 38)를 변환시키기 위한 인터럽트 처리 매커니즘이 개시되었다. PCI 에이전트 인터럽트(36 내지 38)는 예로서 PCI 규격준수 인터럽트를 PCI버스(30) 및 ISA 버스(40) 모두를 구현시키는 컴퓨터 시스템에 의하여 처리하기 위해 ISA 버스 규격준수 인터럽트(40)로 변환하기 위한 프로그램가능 로직(50)에 의해 처리된다. 프로그램가능 레지스터(48)는 PCI 에이전트 인터럽트(36 내지 38)에 응답하여 프로그램가능 로직(50)에 의해 발생될 ISA 인터럽트를 선택할수 있게 된다.

대표도

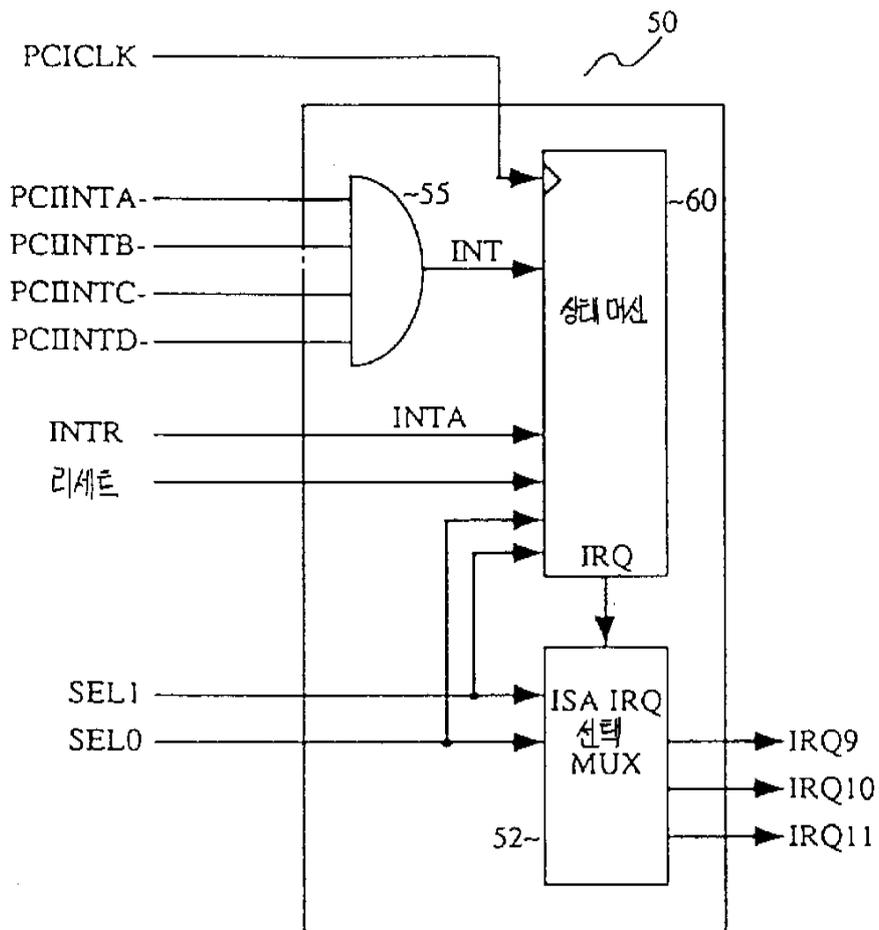
도1

도면

도면1

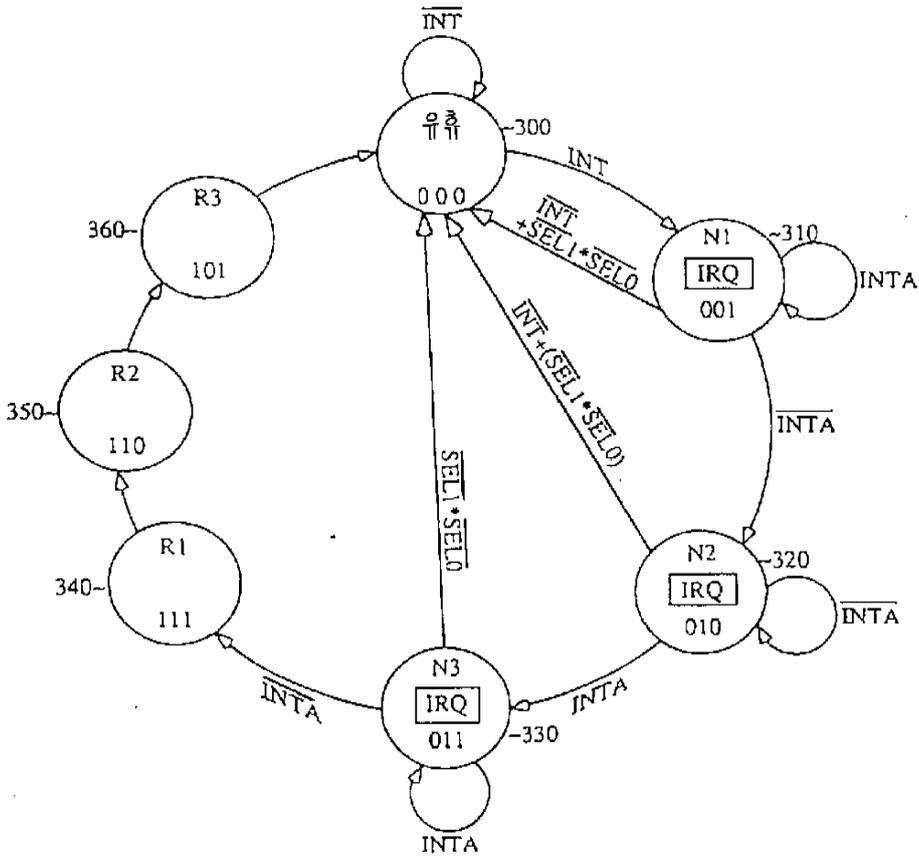


도면2



도면3

PCI INT
상태 머신



도면4

