(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) Int. Cl.[7]
G06F 9/46

(45) 공고일자 2000년10월16일
(11) 등록번호 10-0269582
(24) 등록일자 2000년07월21일

(21) 출원번호	10-1997-0700074	(65) 공개번호	특1997-0705081
(22) 출원일자	1997년01월08일	(43) 공개일자	1997년09월06일
번역문제출일자	1997년01월08일		
(86) 국제출원번호	PCT/US 95/15824	(87) 국제공개번호	WO 96/18150
(86) 국제출원일자	1995년12월06일	(87) 국제공개일자	1996년06월13일
(81) 지정국	AP ARIPO특허 : 케냐 레소토 말라위 수단 스와질랜드 우간다 EP 유럽특허 : 오스트리아 벨기에 스위스 리히텐슈타인 독일 덴마크 스페인 프랑스 영국 그리스 이탈리아 룩셈부르크 모나코 네덜란드 포르투칼 스웨덴 국내특허 : 아일랜드		
(30) 우선권주장	08/351,637 1994년12월07일 미국(US)		

(73) 특허권자 인텔 코오퍼레이션 피터 엔. 데트킨
미합중국 캘리포니아 산타클라라 미션 칼리지 블러바드 2200
(72) 발명자 카다크 제임즈
미합중국 캘리포니아 95128 산호세 라스웰 애비뉴 541
조성수
미합중국 캘리포니아 94087 서니베일 재스퍼 드라이브 1521
피터슨 니콜라스 비.
미합중국 캘리포니아 95134 산호세 어플라우즈 플레이스 704
레인 토마스 알.
미합중국 캘리포니아 95126 산호세 채프만 스트리트616
죠쉬 제이이쉬 엠.
미합중국 캘리포니아 95054 산타클라라 맨션 코트#104, 510
손거 닐
미합중국 캘리포니아 95129 산호세 노웍 드라이브 #웨스트 106, 4281
(74) 대리인 장용식

심사관 : 오홍수

(54) 직렬인터럽트 버스프로토콜

요약

컴퓨터시스템내에서 어떤 수의 주변장치라도 각각의 가능한 인터럽트를 위한 전용핀을 요구하지 않고 시스템인터럽트 제어기(22)에게 어떤 소정의 인터럽트신호라도 신호할 수 있도록 구현되는 직렬인터럽트 버스프로토콜. 그 직렬인터럽트 프로토콜 버스상에서 구현된 각각의 주변장치는 가능한 인터럽트 상태를 통하여 사이클링하기 위한 상태기를 채용한다. 주변장치는 데이지형 체인으로 접속되어, 시스템 주변장치와 같은 상태기 논리를 따라가는 직렬인터럽트 제어기(25)와 함께 개시하고 종료한다. 직렬인터럽트 제어기(25)가 활성신호를 수신하는 경우에 그것은 어떤 인터럽트신호가 인터럽트 제어기 상태기 논리의 인터럽트 상태에 기초하여 시스템의 인터럽트 제어기(22)에게 제공되는지를 판정한다.이차 직렬인터럽트 버스상의 주변장치(41, 42)는 시스템 주변장치와 같은 상태다이어그램을 따르는 상태기 논리를 포함하는 시스템 인터럽트 브릿지(40)를 통하여 일차 시스템 인터럽트 버스상에서 주변장치(30, 31, 32)와 함께 데이지 체인형으로 접속될 수도 있다.

명세서

기술분야

본발명은 컴퓨터시스템의 주변장치접속에 관련한다. 특히, 본발명은 직렬주변장치 인터럽트버스를 이용하여 컴퓨터시스템내에서 복수의 가능한 인터럽트를 처리하는 것에 관련된다.

배경기술

컴퓨터 산업에 있어서, 널리 채택되고 있는 개인용 컴퓨터용 시스템 아키텍추어는 AT시스템설계이었다. 이러한 아키텍추어를 채용하는 종래의 시스템은 ISA, 더나아가 EISA 버스프로토콜을 수행하는 시스템버스를 포함한다. 이러한 프로토콜은 시스템내의 다양한 구성장치가 이용하는 15개의 구별되는 시스템 인터럽트를 규정한다. 이러한 인터럽트는 시스템의 프로그램가능 인터럽트제어기(PIC) 메카니즘내에서 전용되

는 IRQ2등과 같이 IRQ 신호들중의 하나와 함께 IRQ0내지 IRQ15로서 지정된다.

ISA 또는 EISA 버스프로토콜 인터럽트 메카니즘을 수행하는 상기한 컴퓨터 시스템은 일반적으로 다양한 IRQ신호를 수신하고 그것에 응답하여 미처리 인터럽트가 존재한다는 것을 지시하는 신호를 시스템의 중앙처리장치(CPU)에게 송신하는 인터럽트 제어기를 채용한다. 활성인터럽트신호에 응답하여, CPU는 적절한 인터럽트 서비스루틴(ISR)을 실행하기 위하여 CPU에게 코드벡터를 제공하도록 인터럽트 제어기에게 인터럽트신호를 승인한다. 이러한 인터럽트 제어기로서 현재 널리 알려진 Intel8259 주변장치 인터럽트 제어기(PIC)가 있다. 일 실시예에서 그 인터럽트제어기는 8개의 구분되는 IRQ 신호를 수신하는 능력을 가지고 있다. IRQ신호[0:15] 전체 범위를 지원하기 위하여 제1PIC는 IRQ8내지 IRQ15를 수신하고 그것에 응답하여 인터럽트 신호출력을 발생시킨다. 제1PIC의 출력은 다른 구성장치로부터 오는 다른 7개의 입력과 함께 IRQ2 입력으로서 제2PIC에게 제공된다. 제2PIC는 그의 출력을 CPU에게 제공한다. 이러한 방식으로 2개의 8-IRQ입력 PIC는 함께 시스템내에서 15개의 가능한 IRQ신호를 제공하기 위하여 체인과 같이 접속되어 있다.

상기 인터럽트 신호 프로토콜은 약간의 바라지 않은 결점을 가지고 있다. 컴퓨터시스템의 어떤 구성장치가 특정된 IRQ 출력을 시스템 PIC 에게 제공하도록 지정된 경우에, 그외의 구성장치는 복수의 가능한 IRQ신호를 위하여 구성되도록 구속되지 않는다. 종래의 주변장치설계 방법하에서, 이것은 그 장치가 이용할 수도 있는 각각의 인터럽트를 위하여 별도의 핀(개념적으로는 15개의 가능한 ISA IRQ인터럽트신호를 지시하는 15개까지의 핀)을 요구한다. 이들 각각의 핀은 필요한 인터럽트가 서비스되려는 경우에 CPU에게 신호하기 위하여 시스템 PIC 에게 별도의 신호선을 구동한다. 이것은 주변장치를 구현하는 비용을 증가시키고 또한 주변장치 상호접속을 복잡하게 하고 그 크기를 증가시킨다. 최근에는 컴퓨터시스템의 크기를 감소시키는데에 많은 관심이 집중되고 있으며, 특히 휴대용 및 노트북 컴퓨터시스템에서 더욱 그러하다. 본발명의 목적은 핀의 요구사항을 줄이고 복수의 인터럽트를 위하여 구성될 수도 있는 주변장치를 구현하는 복잡성을 감소시키는 것이다.

발명의 상세한 설명

<발명의 요약>

컴퓨터시스템의 아키텍추어가 복수핀 인터럽트 프로토콜을 준수하면서 주변장치핀에 대한 요구사항을 줄이고 또한 주변장치를 구현하는 비용을 줄이는 방식으로 향상되고 있다는 것은 인식할 수 있다. 따라서, 본발명의 목적은 각각의 가능한 인터럽트신호에 대한 전용핀을 요구하지 않고도 어떠한 소정의 인터럽트신호라도 발생시킬 수 있게 시스템 주변장치 또는 구성장치가 구성되는 것을 허용하는 직렬 버스프로토콜을 구현하기 위한 방법 및 장치를 제공하는 데에 있다. 특히, 본발명의 목적은 본발명의 프로토콜에 따라서 신호를 전파시키는 2개의 핀을 사용함으로써 어떠한 IRQ 신호라도 발생시키도록 직렬인터럽트 버스상에서 구현되는 주변장치를 위한 프로토콜을 제공하는 데에 있다.

본발명의 일 실시예에서, 컴퓨터시스템은 일차 및 이차 직렬인터럽트 버스 모두를 가지도록 구현된다. 이러한 직렬인터럽트버스는 그 버스상에 구현된 주변장치의 체인을 따라서 전파되는 인터럽트신호 뿐만아니라 그 버스에 접속된 주변장치에 클록신호도 제공한다. 일차 및 이차 직렬인터럽트버스는 때때로 도킹스테이션에 도킹되도록 노트북 컴퓨터에서 사용되기 위한 본발명의 안정성을 강조하도록 일실시예에서 도시되어 있다. 이 실시예에서 일차 직렬인터럽트 버스는 휴대용 컴퓨터에 채용되는 주변장치에 대하여 이용되고, 한편 이차 직렬인터럽트와 대응하는 주변장치는 도킹스테이션에서 채용된다. 이것은 또한 후술하는 직렬인터럽트주변장치체인에서 이차 인터럽트 버스의 주변장치를 포함하기 위한 직렬인터럽트 브릿지 구성장치의 기능을 나타내도록 제공된다.

상기한 실시예의 컴퓨터 시스템의 아키텍추어는 미처리 인터럽트가 서비스될 필요가 있는 경우에 시스템 처리장치에게 신호를 하기 위한 주변장치 인터럽트제어기(PIC)메카니즘을 포함한다. 주변장치 인터럽트 제어기 메카니즘은 입력으로서 ISA 또는 EISA 버스 인터럽트 프로토콜에 따라서 16개의 가능한 IRQ 신호를 수신한다. 그 PIC 메카니즘은 직렬인터럽트 버스를 구동하는 직렬인터럽트 제어기(SIC)로부터 이들 IRQ신호를 수신한다.

본발명의 실시예에 따르면, 컴퓨터시스템내의 각각의 주변장치를 인터럽트 신호동작을 위하여 2개의 핀을 전용할 필요가 있다. 그 각각은 직렬출력핀($S_{출력}$)과 직렬입력핀($S_{입력}$)을 가진다. 직렬인터럽트 제어기는 또한 직렬버스내에 상주하는 제1주변장치의 $S_{입력}$ 핀에 접속된 $S_{출력}$ 핀을 가진다(또는 만약 잠정적으로 도킹목적을 위하여 구현된다면, 직렬인터럽트 브릿지(SIB)에 접속되는 핀을 가진다). 제1 주변장치(또는 SIB)의 $S_{출력}$ 핀은 직렬 주변장치체인내에서 다음 주변장치의 $S_{입력}$ 핀에 접속되어 있다. 다음에 그것의 $S_{출력}$ 핀은 직렬버스상에 구현된 주변장치라인을 따라서 계속적으로 다음 $S_{입력}$ 핀과 접속되어 있다. 이 체인내의 마지막 주변장치의 $S_{출력}$ 핀은 직렬인터럽트 제어기의 $S_{입력}$ 핀과 접속된다.

브릿지를 채용하는 컴퓨터시스템의 일실시예에서는 SIB가 4개의 핀을 사용한다. 그것의 제1 $S_{입력}$ 핀은 SIC로부터의 $S_{출력}$ 핀의 출력신호를 수신한다. SIB는 또한 2개의 $S_{출력}$ 핀을 가진다.

그 브릿지의 제1 $S_{출력}$ 핀은 이차 직렬 인터럽트 버스상에서 채용되는 제1주변장치의 $S_{입력}$ 핀에게 그것의 신호를 구동한다. 이차 버스상의 주변장치들은 상기한 일차 직렬인터럽트 버스의 주변장치들과 동일한 방식으로 그들의 $S_{출력}$ 핀과 $S_{입력}$ 핀을 통하여 체인형으로 접속된다. 마지막 주변장치 및 이차 버스의 $S_{출력}$ 핀은 SIB에 제2 $S_{입력}$ 입력을 구동한다. 다음, SIB는 그것의 제2 $S_{출력}$ 신호를 일차버스의 제1주변장치의 $S_{입력}$ 핀으로 구동시키고, 그래서 효과적으로 모든 이차 버스 주변장치를 시스템내의 데이지형 체인의 일차 버스 주변장치에 추가한다.

이차 버스가 도킹 스테이션 주변장치버스를 나타내는 본발명의 일 실시예에서 그 이차 버스는 시스템이 도킹된 경우에만 활성화하고, 그렇지 않으면 일차버스만 이용된다. 후자의 경우에 브릿지는 직렬인터럽트

신호를 일차 버스의 제1주변장치로 전진하게 한다.

직렬인터럽트 버스상에서 사용되기 위하여 채용되는 각각의 주변장치는 그것내에 직렬인터럽트 주변장치(SIP)논리를 채용한다. SIP 논리와 SIC 각각은 동일한 클록에서 클록된다. 그 각각은 가능한 IRQ 인터럽트 상태 각각을 통하여 1클록 사이클동안 사이클링하는 상태 다이어그램을 따라가는 논리를 가지고 있다. 직렬 체인내의 각각의 주변장치는 그것을 뒤따르는 주변장치에 하나 앞선 상태를 유지한다. 주변장치가 특별한 IRQ신호를 시스템 PIC에게 송신할 필요가 있는 경우에 그것은 SIP논리 상태기의 대응하는 IRQ상태에 있는 경우에만 그것의 $S_{입력}$ 핀을 통하여 활성신호를 송신할 것이다. 그 체인내의 각각의 주변장치는 그것은 출력 $S_{출력}$을 통하여 $S_{입력}$ 상에서 수신되는 어떠한 신호라도 전진시킬 것이다. 일 실시예에서 시스템 I/O (입력/출력)제어기내에 상주하는 SIC는 SIP논리와 같은 상태기를 실행시키기 위한 상태기 논리를 포함한다. 따라서 임의의 활성신호가 직렬체인내의 마지막 주변장치를부터 SIC로 전진되었을 경우에 SIC는 어떤 IRQ신호가 시스템 PIC 에게 제공될 필요가 있는지를 인식할 것이다. 이것은 임의의 컴퓨터 시스템내에서 어떠한 수의 주변장치에 대해서도 각각의 신호에 대한 전용핀을 필요로 하지 않고도 어떠한 가능한 인터럽트 신호라도 발생시킬 수 있는 메카니즘을 제공한다. 더욱이 이것은 SIC가 시스템 PIC에게 적절한 IRQ 신호를 전진시킬 것을 보장하면서 복수의 주변장치는 동일한 IRQ 상태로서 어떠한 활성신호라도 송신할 수 있기 때문에, IRQ 들의 공유를 허락한다.

또한 본발명의 일실시예에서는 SIC가 시스템 PIC를 바이패스하고 시스템 CPU에게 시스템 관리 인터럽트(SMI)를 직접 제공하도록 하는 설비가 제공된다.각각의 주변장치의 SIP 논리 및 SIC 논리 각각은 시스템 관리 인터럽트(SMI)를 발생시키기 위한 상태를 포함한다.

도면의 간단한 설명

도 1은 본발명의 일실시예에 따르는 직렬인터럽트버스 프로토콜을 채용하는 컴퓨터시스템의 아키텍추어를 도시하는 블록다이어그램,

도 2는 본발명의 일 실시예에 따르는 직렬인터럽트버스상에 채용되는 각각의 주변장치에 대응하는 직렬인터럽트논리에 의하여 수행되는 상태다이어그램,

도 3은 직렬인터럽트 체인을 통하여 상태를 신호하는 인터럽트핀을 도시하는 타이밍도,

도 4는 인터럽트의 송신을 위한 버스클록을 시작하는 주변장치 또는 브릿지를 위한 타이밍도,

도 5는 본발명에 따르는 직렬인터럽트 주변장치를 위한 일반화된 상태도.

<발명의상세한설명>

본발명은 직렬인터럽트버스를 채용하는 컴퓨터시스템내에서 인터럽트 신호 프로토콜을 위하여 제공되는 방법 및 장치이다. 본 상세한 설명란에서 컴퓨터시스템의 일실시예는 휴대용 컴퓨터 및 도킹스테이션 모두에 관계되어 도킹가능 휴대용 컴퓨터 및 주변구성장치와 일치하는 일차 및 이차 직렬인터럽트 버스 모두를 포함하는 것으로서 설명된다. 본발명은 본발명의 이점을 성취하기 위하여 직렬인터럽트 버스를 채용하는 다른 컴퓨터시스템에서도 실시가능하기 때문에 이것은 예시를 위한 목적이지 제한적으로 제공되는 것이 아니라는 것으로 이해되어야 한다.

본 상세한 설명란 전체를 통하여 특별한 신호이름, 버스프로토콜 및 컴퓨터시스템 아키텍추어와 같은 많은 특정된 세부사항들은 본발명의 완전한 이해를 제공하기 위한 것이다. 본발명의 기술분야의 평균적 전문가라면 이러한 특정의 세부사항이 없더라도 본발명이 실시될 수 있다는 것을 인식할 것이다. 한편, 주지의 구성장치, 구조 및 기술은 본발명의 요지를 불명확하게 하지 않기 위하여 상세히 설명하지는 않는다. 특히 본발명의 많은 기능들은 다양한 타입의 논리회로에 의하여 구현될 수 있도록 상태다이어그램 논리 및 타이밍 다이어그램으로써 설명될 것이다. 본발명의 기능이 응용특정집적회로(ASIC), PAL 및 PLD와 같은 프로그램가능 논리장치 또는 전용논리회로를 이용하는(이것에만 한정하지 않음) 다양한 기술에 의하여 구현될 수 있다는 것은 본발명의 기술분야의 평균적 전문가에 의해 인식될 것이다. 또한, 본 상세한 설명란에서 명명한 신호이름은 어떤 경우에는 일 상태에서 활성이고 다른 상태에서 비활성이라는 것으로 설명된다. 본 발명의 개시는 물론 그 역의 경우에도 적용된다.

도 1를 참조하면, 본발명에 채용될 수도 있는 컴퓨터시스템의 아키텍추어가 도시되어 있다. 도 1의 컴퓨터시스템 아키텍추어는 그 아키텍추어의 계층적 인터럽트 조직을 강조하도록 도시되어 있다. 본발명에 영향을 받지 않거나 연관되지 않은 컴퓨터시스템의 많은 구성장치는 도시되지 않았다. 예로써, 컴퓨터시스템은 중앙처리장치(CPU; 10)를 포함하고 그것은 컴퓨터시스템의 도시되지 않은 메모리 시스템과 통신하기 위하여 메모리버스(15)와 접속된다.

예시된 CPU(10)는 또한 시스템 I/O 제어기(25)로부터 2개의 제어신호를 수신하는 것으로서 도시되어 있다. 이들 신호는 CPU 및 시스템 관리 인터럽트(SMI)에게 INTR 인터럽트 신호로서 식별된다.

다음 더욱 자세히 설명되겠지만, CPU 대한 INTR 신호는 시스템 구성장치가 인터럽트 서비스루틴(ISR)를 실행하도록 CPU에게 요구할 때 CPU(10)에게 알리는 시스템의 주변장치 인터럽트 제어기(PIC) 메카니즘(22)으로부터의 종래의 일반적인 신호이다.

본발명의 일 실시예에서 CPU(10)는 인텔사의 마이크로프로세서 아키텍추어에 대응하도록 설계된 마이크로프로세서이다. 이 아키텍추어는 시스템 관리모드(SMM)로 불리우는 동작모드를 지원한다. 시스템 관리모드를 지원하기 위하여 CPU는 일정한 환경에서 시스템 관리 인터럽트(SMI)로 불리우는 시스템의 다양한 주변장치로부터의 어떤 특별한 인터럽트를 수행하여야 한다. 따라서, 이 예시된 컴퓨터시스템의 CPU(10)는 종래의 INTR 인터럽트신호 뿐만아니라 시스템 관리 인터럽트 처리를 위한 SMI 인터럽트를 수신하는 것으로 도시되어 있다.

시스템 I/O제어기(20)는 컴퓨터시스템을 위한 주변장치 인터럽트 제어기(PIC) 메카니즘(22)을 포함한다. 상기한 것처럼, 이것은 두 개의 Intel 8259 PIC로써 구성되어, 다양한 시스템 구성장치로부터의 15개의

가능한 IRQ신호를 수신하고 그에 대응하여 CPU(10)에게 INTR신호를 제공하도록 구성된다. 대체의 실시예에서는 다른 PIC메카니즘이 동일한 기능을 수행하도록 채용될 수도 있다. 시스템 I/O 제어기(20)는 또는 본발명의 직렬인터럽트 제어기(SIC;25)를 포함한다. 다음에 더욱 상세히 설명하겠지만, SIC(25)는 본발명의 직렬인터럽트 프로토콜을 채용하여, 시스템 주변장치로부터의 특별한 IRQ요구를 검출하고 시스템 PIC(22)에게 적절한 IRQ 입력신호를 제공한다. 더욱이 SIC(25)는 본발명의 프로토콜을 채용하는 시스템 주변장치중의 하나가 시스템 관리 인터럽트를 필요하다고 지시하는 경우에 CPU(10)에게 SMI 인터럽트신호를 제공한다. 또한 PIC(22)는 시스템 마더보드상에구현된 것과 같은 다른 시스템 구성장치로부터 또는 직렬인터럽트 버스와 독립적인 다른 메카니즘을 통하여 IRQ 신호를 수신하도록 결합되어 있다.

본발명의 예시된 실시예에 따르면, 직렬인터럽트버스는 컴퓨터시스템의 주변장치(30, 31, 32)를 결합시키기 위한 일차 직렬인터럽트버스로서 도시되어 있다. 또한, 주변장치(41, 42)를 결합시키기 위한 이차 직렬인터럽트버스가 있다. 이것은 본발명이 도크(45)로서 나타낸 도킹스테이션에 "도킹"될 수도 있는 휴대용 컴퓨터시스템에 이용되기에 적합하다는 것을 보여준다. 본발명의 직렬인터럽트 프로토콜은 주변장치(41, 42)와 같은 도킹 스테이션과 관련된 주변장치 뿐만아니라 주변장치(30, 31, 32)와 같은 휴대용 컴퓨터시스템의 온-보드 방식의 주변장치 모두에 의해서 발생되는 인터럽트 신호를 처리하는 데에 제공된다. 도 1에 도시된 일차 및 이차 직렬인터럽트버스 각각에 있어서, 그들은 직렬인터럽트 브릿지(40)를 통하여 결합되어 있다는 것을 알 수 있고, 그것에 관하여는 후술한다.

직렬인터럽트 제어기 및 직렬인터럽트 브릿지 뿐만아니라 직렬인터럽트 버스 프로토콜에 대응하는 각각의 주변장치는 클로킹신호인 PCCLK# 및 CLKRUN#을 수신한다. 본발명의 일실시예에 따르는 실제적인 신호전파타이밍은 도 3 및 도 4의 타이밍 다이어그램과 관련하여 다음에 상세히 설명될 것이다.

본발명의 직렬인터럽트 버스 프로토콜을 채용하는 시스템 주변장치는 그것의 동작을 위하여 단지 2개의 핀을 전용할 필요가 있다. 이것은 또한 직렬인터럽트제어기(25)를 위해서는 그러하다. 본발명의 주변장치는 서로 체인형으로 결합되어, SIC(25)로부터의 $S_{출력}$ 신호가 직렬인터럽트 체인의 제1주변장치(또는 브릿지)에게 $S_{입력}$ 신호로서 제공된다. 체인(32)의 마지막 주변장치는 그것의 $S_{입력}$신호를 그 체인내의 마지막에서 바로전의 주변장치(31)의 $S_{출력}$ 핀으로부터 수신한다. 그 체인내의 마지막 주변장치는 그것의 $S_{출력}$ 핀을 SIC(25)의 $S_{입력}$ 핀에게 구동한다.

직렬인터럽트 브릿지(SIC;40)가 채용되는 경우에 그것은 2개의 $S_{입력}$ 핀 및 2개의 $S_{출력}$ 핀을 포함한다. $S_{입력1}$ 핀은 SIC(25)의 $S_{출력}$으로부터 신호를 수신한다. 컴퓨터시스템이 도킹 스테이션(45)을 통하여 도킹된 경우에 직렬인터럽트 브릿지는 이차 직렬인터럽트버스의 제1 주변장치(41)의 $S_{입력}$입력에게 $S_{출력1}$신호를 제공한다. 이차 인터럽트 버스상의 주변장치는 일차 직렬인터럽트버스상의 주변장치와 동일한 방식으로 서로 체인형으로 결합되어 있다. 이차 직렬인터럽트 버스의 마지막 주변장치(42)는 그것의 $S_{출력}$신호를 SIB(40)의 $S_{입력2}$ 핀에게 구동한다. 그러면 SIB(40)는 일차 직렬인터럽트버스상의 제1 주변장치(30)의 $S_{입력}$ 입력에게 신호를 구동시키도록 결합된 $S_{출력2}$ 핀을 가진다. 시스템내에 브릿지가 존재하지만 시스템이 도킹되지 않은 경우에 SIB(40)는 도킹되었다면 이차 직렬인터럽트버스에 결합되었을 수도 있는 핀들은 무시하고 그것은 $S_{출력2}$ 핀을 통하여 $S_{입력1}$ 핀상에 제공된 신호들을 전진시킨다.

도 1에서 다양한 주변장치(30, 31, 32, 41, 42)는 주변장치의 타입에 관련하여 식별되어 있지는 않다. 어떠한 주변구성장치라도 직렬인터럽트 버스상에서 구현되도록 본발명을 채용할 수 있다. 각각의 주변장치내에서 채용되는 직렬인터럽트 주변장치(SIP)논리만이 도면에 도시되어 있다. 시스템내의 각각의 주변장치가 인터럽트 서비스를 요구하는 경우에 상이한 IRQ 신호를 지정할 수도 있다는 것은 이해되어야 한다. 더욱이, 어떤 주변장치는 시스템 PIC(22)에게 제공될 상이한 IRQ 입력을 요구하도록 구성될 수도 있다는 것으로 이해되어야 한다. 각각의 주변장치의 SIP논리, SIB(40) 및 SIC(25)는 도 2에 도시된 상태 다이어그램을 따르기 위한 상태기 논리를 채용할 것이다. 본발명의 프로토콜에 따라 각각의 주변장치 SIP논리는 그것의 $S_{출력}$ 핀을 통하여 이전의 클록사이클에서 그것의 $S_{입력}$ 핀 상에서 수신된 신호값을 전진시킨다. 본발명의 일실시예를 따르면, $S_{입력}$ 신호는 로우 논리상태에서 활성인 것으로 가정될 것이다. 물론 대체 실시예에서는 그것은 반전될 수도 있다.

직렬인터럽트버스상의 어떠한 주변장치도 서비스된 인터럽트를 요구하지 않는 경우에, 각각의 $S_{출력}$ 핀은 하이상태로 올라간다. 임의의 주어진 주변장치를 위한 상태기논리는 초기에 정지상태(220)에 있는다. 주변장치(32)와 같은 주변장치의 SIP논리가 그것의 입력핀 $S_{입력}$($S_{입력}$은 로우상태로 당겨진다)상의 활성값을 인식하는 경우에 SIP논리는 시작 메시지상태(200)로 천이한다. 그러면 그 주변장치는 그것의 $S_{출력}$핀을 통하여 활성신호를 그 체인내의 다음 주변장치의 $S_{입력}$핀에게로 전진시킨다. 주변장치(2)의 경우에 있어서, 그것은 활성 $S_{입력}$신호를 SIC(25)에게 전파시킨다. 그것은 SIC(25)가 그것의 상태기논리내의 시작메시지 상태(200)로 천이하도록 만든다. 이때, 직렬체인내에서 그 다음의 주변장치는 그것의 상태기내에서 1상태를 전진하여 SMI상태(201)로 간다.

만약 직렬인터럽트체인내의 마지막 주변장치인 주변장치(32)가 그것이 SMI상태(201)에 있는 경우에 그것의 입력핀($S_{입력}$)상에서 활성신호를 수신한다면, 그것은 그 활성신호를 SIC(25)의 $S_{입력}$핀에게 다음 클록 사이클의 상승부에서 전진시킨다. 만약 SIC(25)가 그것의 상태기 논리가 SMI상태(201)에 있는 경우에 그것의 $S_{입력}$ 핀상에서 활성신호를 수신한다면, 그것은 CPU(10)의 시스템 관리 인터럽트(SMI) 입력에게 인터럽트신호를 전파시킨다. 이것은 CPU(10)가 시스템 관리 인터럽트를 즉시 처리하도록 하게 한다.

상기 언급한 것처럼, 직렬인터럽트버스상의 주변장치 체인내의 각각의 주변장치는 도 2에 도시된 상태다이어그램을 따르는 상태기논리를 포함한다. 각각의 주변장치에 대한 상태기는 정지상태(220)에 있지 않은 경우에 1클록 사이클동안 상태 다이어그램의 각각의 상태에 남아있게 된다. 정지상태(220)로부터 시작메시지상태(200)로의 천이는 주변장치의 $S_{입력}$핀상에서 활성신호의 제1 수신과 동시에 일어난다. 각각의 주

변장치는 이전의 클록사이클에서 그것의 $S_{입력}$핀상에서 수신하였던 신호를 그것의 $S_{출력}$핀을 통하여 계속하여 전진시킨다. 단 그 주변장치가 그것이 활성인터럽트를 구동하기를 원하는 상태에 있는 경우에는 제외된다. 따라서 각각의 주변장치의 SIP논리는 직렬인터럽트체인내에 그것을 바로 선행하여 갈 주변장치보다 1상태 앞에 있다는 것으로 이해될 수 있다.

주변장치가 특별한 IRQ신호를 신호하기를 원하는 경우에 그것은 도 2의 상태다이어그램의 대응하는 IRQ상태에 있을 때에 그것의 $S_{출력}$핀을 통하여 활성신호를 제공함으로써 상기 신호를 신호할 것이다. 주변장치에 의한 그것의 $S_{출력}$핀상의 모든다른 출력은 그것의 $S_{입력}$핀상에서 수신된 값이 될 것이다. 예를들면, 만약 도 1에 도시된 것처럼 주변장치(30)가 IRQ12가 CPU(10)에 의해서 서비스되도록 신호할 필요가 있다면, 그것의 상태기논리가 IRQ12상태 (213)에 있는 경우에 그것은 그것의 $S_{출력}$핀을 통하여 활성신호를 전파할 것이다. 이때, 주변장치(31)는 IRQ11상태 (212)에 있을 것이고, 한편 주변장치(32)는 IRQ10 상태(211)에 있을 것이다. 마찬가지로 이때에 SIC(25)는 IRQ9 상태(210)에 있게될 것이다. SIP논리(30)가 그것의 $S_{출력}$핀을 통하여 활성신호를 전파한 후 1클록사이클에서, SIP논리(31)는 IRQ12상태(213)에 있을 것이고 그것의 $S_{입력}$핀을 통하여 활성신호를 수신할 것이다. 다시, 주변장치(32)는 여전히 주변장치(31) 다음의 1상태에 있을 것이고, IRQ11상태(212)에 있을 것이며, 한편 SIC(25)는 IRQ10상태(211)에 있는 것이다. 이 신호는 계속하여 이러한 방식으로 전파될 것이다. 그래서 그것이 SIC(25)에 의하여 수신되는 경우에 SIC(25)는 IRQ12상태(213)에 있을 것이다. 따라서, SIC(25)는 어떠한 IRQ신호(IRQ12)가 3클록사이클 이전에 SIP논리(30)의 $S_{출력}$핀을 통하여 출력된 활성신호에 기초하여 시스템 PIC(22)에 제공되어야 하는지를 검출할 수 있을 것이다.이러한 방식으로 컴퓨터 시스템내에서 어떠한 수의 가능한 IRQ 인터럽트라도 전파시시킬 수 있는 직렬인터럽트버스상의 각각의 주변장치를 위하여 단지 2개의 핀만이 요구된다.

상기의 인터럽트신호 원리는 도킹장치(45)를 통하여 도킹되는 휴대용 컴퓨터 시스템을 위하여 매우 효과적이다. 이러한 경우에 직렬인터럽트 브릿지(SIB; 40)는 동작상태로 된다. SIB(40)는 상태기 논리를 포함하고, 이것은 또한 도 2의 상태 다이어그램에 의해 표시되는 동일한 상태기를 따른다. SIB(40)이 $S_{입력}$을 통하여 그것의 제1활성신호를 수신한 경우에, 그것은 정지상태(220)로부터 시작메시지상태(200)로 천이한다. 다음, 그 브릿지가 이차 직렬인터럽트버스에 활성적으로 도킹되었다고 가정하면, 그것은 다음 클록 사이클중에 그것의 $S_{출력1}$핀을 통하여 활성신호를 이차 직렬인터럽트 버스 체인상의 제1주변장치에게 전파시킬 것이다. 이 경우에 그것은 SIP논리(41)를 가진다. 다음, SIB(40)는 SMI 상태(201)로 천이할 것이다. SIB(40)가 SMI상태(201)에 있는 경우에, 제1주변장치는 상태 메시지상태(200)로 천이한다.

시작메시지상태(200)로부터 주변장치(41)는 시작메시지 활성신호를 이차 직렬인터럽트버스상의 다음 주변장치의 SIP논리(42)에게 전파시킨다. 상기한 바와같이 이차 직렬인터럽트버스의 마지막 주변장치는 그것의 $S_{출력}$출력을 SIB(40)의 제2 $S_{입력2}$입력을 통하여 전전시키고, 이것은 다음에 일차 직렬인터럽트버스상의 제1주변장치(30)의 제1 $S_{입력}$입력핀에게 그 SIB(40)의 $S_{출력2}$출력핀을 통하여 전진된다. 이러한 방식으로 본 발명의 직렬인터럽트 버스 프로토콜은 체인형 직렬인터럽트 버스의 복수 레벨로 크기 조절가능하다. 그 각각은 일련의 직렬인터럽트 브릿지를 통하여 결합된 복수의 직류 인터럽트 주변장치를 가질 수도 있다.

직렬인터럽트 주변장치는 적절한 상태에서 전진되는 인터럽트가 시스템 PIC(22)의 통지를 위하여 대응하는 상태에 있는 SIC(25)에 의하여 검출될 수 있도록 하나의 논리적 체인을 형성할 것이다. 이것은 시스템에 채용되는 주변장치 및 브릿지의 수와 관계없이 그러하다. 더 나아가, 본발명의 프로토콜은 복수의 주변장치가 동일한 IRQ신호를 발생시키도록 구성되는 것을 허용한다. 그래서, 만약 어떤 주변장치가 활성 IRQ13을 신호하기를 원한다면 그리고 그 체인내의 앞선 다른 주변장치가 이미 그것을 수행하였다면, 효과적으로 그 동일한 동작이 제2주변장치에 의하여 취해진 것이다. 즉, 그것은 IRQ13상태(214)에 있는 경우에 그것의 $S_{출력}$핀을 통하여 활성신호를 전진시킨다.

도 3은 직렬인터럽트 브릿지를 갖지 않은 시스템에서 상기한 소망하는 프로토콜을 나타내는 타이밍 다이어그램을 도시한다. 제1시간격에서 정지상태로부터 시작상태로의 제1주변장치는 대응하는 수의 클록사이클후에 직렬인터럽트제어기(25)에 도달할 때까지 천이하고, 그것은 체인내에서 나머지 주변장치를 따라가면서 캐스케이드 형으로 연결된다. 본발명의 다른 이점은 SIC가 주변장치에 의하여 발생된 인터럽트가 에지 트리거형 인터럽트인지 레벨 민감(sensitive)형 인터럽트인지를 인식할 필요없이, 주변장치 SIP논리에 의하여 인터럽트 신호의 에지-레벨 전환이 허용된다는 것이다.

본발명의 특별한 실시예에서는 주변장치가 PCICLK 신호에 대하여 클록되는 실행을 하도록 자동적으로 설계되고, 활성 CLKRUN#신호가 송신되는 경우에 만약 그것이 정지되어 있다면 PCICLK을 재시작하는 것으로 이미 응답한다. 그래서, 단지 2개의 추가적인 핀만이 본발명의 직렬인터럽트 버스 프로토콜을 지원하기 위하여 요구된다. 주목할 것은 예시된 실시예의 상태다이어그램 및 SIP 논리는 IRQ1과 IRQ3상태 사이에서 IRQ2상태를 생략한다. 그 이유는 종래의 IRQ 구현에 있어서, IRQ2는 앞에서 설명된 2개의 8-입력 인터럽트 제어기를 서로 결합하기 위하여 캐스케이드형 채널을 위하여 전용된다. 물론 이것은 변경될 수도 있고, IRQ2상태가 본발명의 대체 실시예에서 포함될 수도 있다.

상태 다이어그램내에 각각의 상태는 일 클록 시퀀스를 나타내고, PCICLK 신호에 동기화되어 있다. 그래서, 전체의 상태기는 동기화된다. 하지만 일상태 즉 정지상태(220)는 비동기적 이벤트인 $S_{입력}$의 활성상태로의 제1 천이에 의존한다. 이것은 동일한 클록사이클에 모든 주변장치를 동기화는 것을 허락한다. 정지상태는 2개의 일차적인 기능을 가진다. 즉,

1) 시작비트(시작 메시지 활성신호)의 발생을 통하여 직렬인터럽트 버스 프로토콜 시퀀스의 동기화 기능

2) 어떠한 인터럽트 에이전트라도 활성인터럽트 결정에 근거하여 PCICLK(만약 정지되어 있다면)를 재시작하기 위하여 정지 브레이크 이벤트(CLKRUN#)를 동기적으로 발생시키도록 허용하는 기능

프로토콜 상태기를 통과할 때 마다 그 통과한 후, 모든 장치는 정지상태에서 종료하고 시작 메시지 상태로 천이하기 전에 그것의 $S_{입력}$ 입력에서 다음 활성지시를 기다린다. 주목할 것은 버스가 직렬이므로 각

각의 주변장치 SIP논리는 이전의 SIP논리가 시퀀스를 수신한 후 일 사이클에서 그 시작 시퀀스를 수신할 것이다. 그래서,시작 천이는 모든 장치에 대해서 동시에 일어나지는 않으며, 오히려 주변장치 체인을 따라 내려가면서 일종의 리플 형식으로 일어난다.

도 4는 PCICLK가 정지된 경우에 직렬인터럽트 요구가 어떻게 개시되는 지를 나타낸다. 이 경우에 주변장치는 PCICLK 신호가 중지되었다는 것을 샘플링한 경우에 CLKRUN# 신호를 활성으로 구동시킴으로써 그 사이클을 개시한다. 그것은 종래의 CLKRUN# 신호의 내역에 일치하는 2개의 PCICLK가 검출될 때까지, CLKRUN#신호를 로우로 유지한다. 그러한 내역에 따라, CLKRUN# 신호는 출력이 정지될 때까지 클록 제어 논리에 의하여 로우로 유지될 것이다. $S_{입력}$ 데이터핀이 활성으로 한 번 샘플링되면, 시작상태가 들어가게 된다. 주목할 것은 만약 모든 직렬인터럽트 주변장치가 정지상태(220)에 있지 않다면, PCICLK가 정지되지 않은 것을 인식하는 것은 SIC(25)의 책임이라는 것이다. 그들은 모든 장치가 정지상태에 있을 것이다 라는 것을 보장할수 있는 경우에 CLKRUN#를 4클록동안 하이상태에 머물게 하므로써 그러한 기능을 성취할 수 있다. SIC가 정지상태이고 새로운 메시지를 시작하지 않은 경우에 그것은 모든 장치가 정지상태에 있다는 것으로 보장된다.

본발명을 추가적으로 개량함으로써, 레벨인터럽트 요구가 가- 인터럽트요구를 발생시킬 가능성을 배제할 수 있다. 그것은 I/O장치에서 인터럽트 요구를 소거하고 그 다음에 시스템 PIC에게 인터럽트종료(EOI)를 발송하는 인터럽트 서비스 루틴에 의해서 가능하게 된다. 하지만 직렬버스체인의 IRQ의 직렬화 때문에, 만약 EOI가 IRQ의 갱신 이전에 일어난다면 가- 인터럽트가 발생한다. 그것은 PIC가 EOI가 일어났을 경우에 여전히 활성인 레벨 민감성 IRQ에 기초하는 프로세서에게 어떤 인터럽트를 발생시키기 때문에 일어난다. 하지만 그것은 직렬메시지가 결국 IRQ를 표명포기하기에 이르기 때문에 후에 포명포기된다. 이 경우에, PIC는 인터럽트 벡터를 가- 인터럽트를 나타낸 프로세서에게로 반환할 것이다. 한편 이것이 치명적인 것은 아니라도 그것이 초당 4,000 회 정도 발생한다면 성능에 영향을 미칠수 있다.

이러한 문제를 피하기 위하여 SIC(25)는 특별한 IRQ가 더 이상 신호전송되지 않아야 한다는 것을 SIP로부터의 갱신으로서 얻을 때까지, 인터럽트 서비스 루틴내에서 인터럽트 신호의 리세트를 지연시키도록 EOI 사이클(특정 또는 비-특정)을 인터셉트해야 한다. 이것은 다음에 특정 IRQ상태에 도달하는 경우에 SIC에 의해서 아무런 활성 신호도 수신됨이 없이 직렬체인을 통과하는 하나의 완전통과를 요구한다. EOI사이클이 발생된 경우에 SIC는 그 사이클을 종료시키고 버스 프로토콜 상태기를 추적한다. 이것은 요구하는 주변장치가 사실상 그것을 이미 표명포기한 경우에, PIC가 IRQ를 신호전송하는 것을 방지한다. 도 5는 본발명을 구현하기 위하여 일반적으로 적용될 수 있는 일반화된 상태기를 나타낸다. 그것은 도2의 상태기 다이어그램을 일반화한 것이다. 그 상태기 논리의 초기상태는 그 연관된 SIP 논리에 대한 $S_{입력}$이 표명되지 않는한 상태기 논리가 유지하는 정지상태(500)이다.

한번 SIP 논리가 활성 $S_{입력}$ 신호를 수신하면, SIP논리상태는 일 클록사이클동안 그것이 유지하는 시작 메시지 상태(501)로 천이한다. 시작메시지상태(501)로부터 상태기 논리는 INT-1 인터럽트 상태(501) 다음에서 사이클하고, 그리고 특별한 인터럽트 프로토콜과 관련된 소정의 인터럽트수 후에 INT-N 상태(510)에 도달한다. 다음, 상태기 논리는 활성 $S_{입력}$신호의 다음의 표명이 있을 때까지 그것이 유지하는 정지상태(500)로 되돌아가는 천이를 한다.

지금까지, 주어진 인터럽트 프로토콜과 관련되는 소정의 수의 가능한 인터럽트중에서 어느 것이라도 신호전송할 수 있는 주변장치가 있는 직렬인터럽트버스를 구현하기 위한 메카니즘을 설명하였다. 비록 본발명이 다양한 실시예를 통하여 설명되었지만, 많은 변형과 변경이 본발명의 사상과 범위를 벗어나지 않고도 만들어 질 수 있다는 것은 당해 기술분야의 전문가에게 인식될 것이다. 본발명은 청구의 범위에 의하여 평가된다.

(57) 청구의 범위

청구항 1

인터럽트 미처리 신호에 응답하여 인터럽트 서비스 루틴을 실행하기 위한 중앙처리장치(CPU),

복수의 잠재적 인터럽트 요구신호중의 어느 하나에 응답하여 상기 인터럽트 미처리 신호를 제공하도록 결합된 주변장치 인터럽트 제어기(PIC),

수신된 직렬인터럽트 신호에 대응하여 상기 복수의 잠재적 인터럽트 요구 신호중의 적절한 하나를 발생시키기 위하여 상기 PIC에 결합된 직렬인터럽트 제어기, 및

상호 접속되고 상기 인터럽트 제어기와 통신하는 제1복수의 직렬인터럽트 주변장치로 구성되고;

상기 제1복수의 직렬인터럽트 주변장치 각각은 상기 직렬인터럽트 제어기에 의하여 발생되는 상기 복수의 잠재적 인터럽트 요구신호중의 하나이상을 요구할 수도 있고, 상기 직렬인터럽트 제어기에게 상기 직렬인터럽트 신호를 제공하기 위한직렬인터럽트 신호프로토콜을 구현하기 위하여 프로토콜 논리를 포함하는 것을 특징으로 하는 컴퓨터시스템.

청구항 2

제 1 항에 있어서, 상기 복수의 직렬인터럽트 주변장치 각각은 직렬인터럽트 입력핀 및 직렬인터럽트 출력핀을 포함하고, 상기 복수의 직렬인터럽트 주변장치는 후속하는 직렬인터럽트 주변장치의 직렬인터럽트 입력핀에게 그것의 출력을 제공하는 하나의 직렬인터럽트 핀의 직렬인터럽트 출력핀과 직렬인터럽트 주변장치 체인을 형성하도록 상호 접속되는 것을 특징으로 하는 컴퓨터시스템.

청구항 3

제 2 항에 있어서, 상기 직렬인터럽트 제어기는 직렬인터럽트 입력핀 및 직렬인터럽트 출력핀을 더욱 포

함하고,

상기 직렬인터럽트 제어기의 상기 직렬인터럽트 출력핀은 그것의 출력을 상기 직렬인터럽트 체인내의 제1 직렬인터럽트 주변장치의 상기 직렬인터럽트 입력핀에게 제공하고,

상기 직렬인터럽트 제어기의 상기 직렬인터럽트 입력핀은 그것의 입력을 상기 직렬인터럽트 체인내의 마지막 주변장치의 상기 직렬인터럽트 출력핀으로부터 수신하는 것을 특징으로 하는 컴퓨터시스템.

청구항 4

제 3 항에 있어서, 상기 직렬인터럽트 체인내에 제2복수의 직렬인터럽트 주변장치를 결합시키기 위하여 상기 직렬인터럽트 체인내에 배치된 직렬인터럽트 브릿지를 더욱 포함하는 것을 특징으로 하는 컴퓨터시스템.

청구항 5

제 4 항에 있어서, 도킹 스테이션을 더욱 포함하고, 상기 제2복수의 직렬인터럽트 주변장치는 상기 도킹 스테이션과 연계되는 주변장치를 포함하는 것을 특징으로 하는 컴퓨터시스템.

청구항 6

중앙처리장치(CPU), 복수의 잠재적 인터럽트 요구신호중의 하나에 응답하여 미처리 인터럽트를 상기 CPU에게 인식시키기 위한 주변장치 인터럽트 제어기 및 수신된 직렬인터럽트 신호에 응답하여 상기 주변장치 인터럽트 제어기에게 상기 인터럽트 요구신호를 제공하기 위한 직렬인터럽트 제어기를 포함하는 컴퓨터시스템에서 사용하기 위한 주변장치에 있어서,

직렬인터럽트 버스와 통신하기 위한 직렬인터럽트 프로토콜논리,

직렬인터럽트 입력신호를 수신하기 위하여 상기 직렬인터럽트 버스에 상기 주변장치를 결합시키기 위한 상기 직렬인터럽트 프로토콜논리에 결합되는 직렬인터럽트입력핀, 및

상기 직렬인터럽트 버스를 통하여 직렬인터럽트 신호를 전달하기 위하여 상기 직렬인터럽트 버스에 상기 주변장치를 결합시키기 위한 상기 직렬인터럽트 프로토콜논리에 결합되는 직렬인터럽트출력핀으로 구성되고;

상기 직렬인터럽트 프로토콜논리는 상기 복수의 잠재적 인터럽트 요구중의 어느것이라도 상기 직렬인터럽트 출력핀을 통하여 전달될 수 있도록 상기 직렬인터럽트 출력핀을 제어하는 것을 특징으로 하는 주변장치.

청구항 7

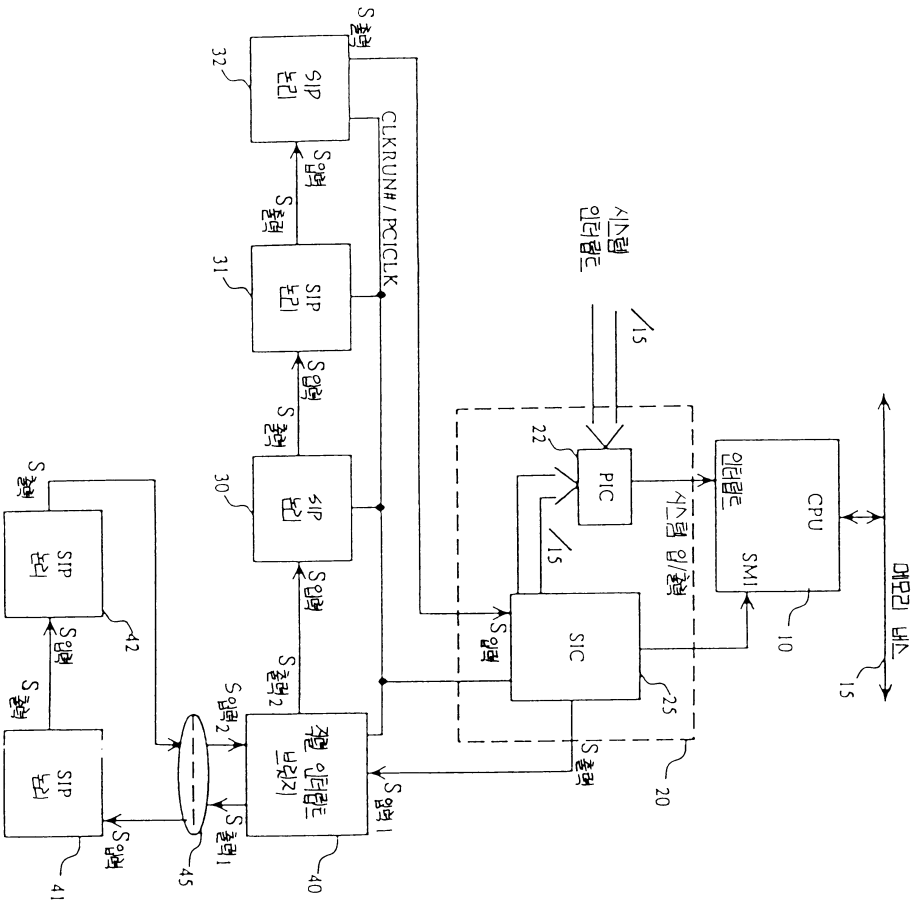
제1클록주기동안에 직렬인터럽트 입력핀상에서 직렬인터럽트신호를 수신하는 단계,

제2클록주기동안에 직렬인터럽트 출력핀을 통하여 상기 직렬인터럽트 신호를 전진시키는 단계,

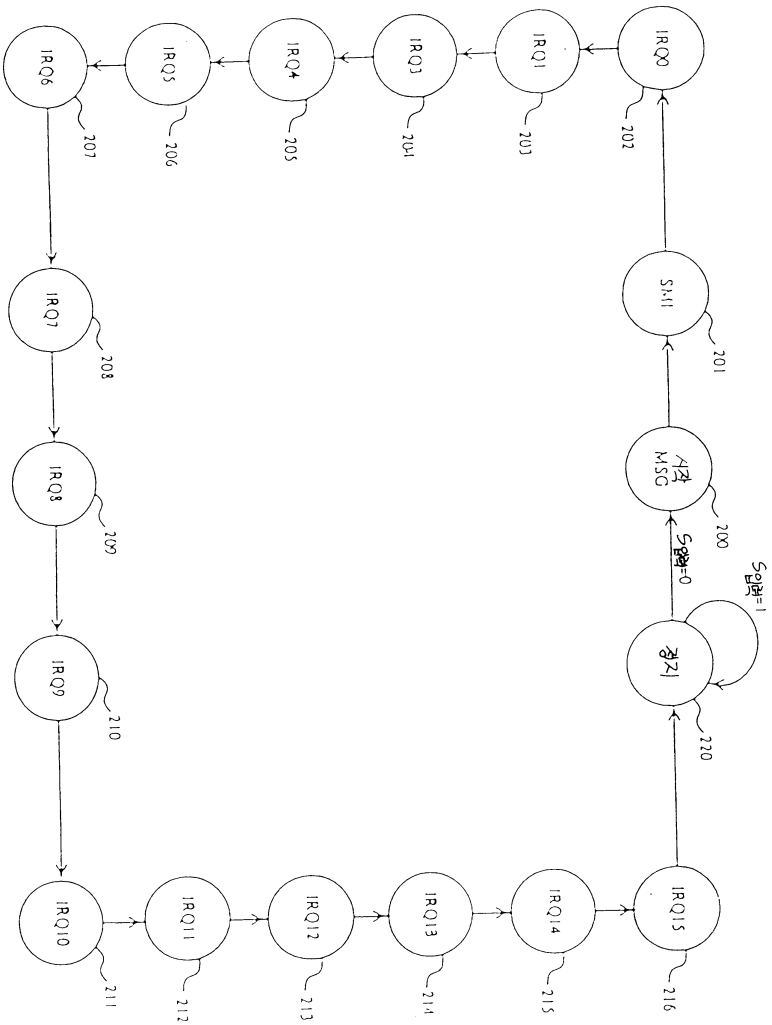
소망하는 인터럽트상태에 대응하는 클록주기동안에 상기 인터럽트 출력핀상에서 활성신호를 표명하는 단계로 구성되는 것을 특징으로 하는 직렬인터럽트 버스상에서 구현되는 주변장치에 의해 인터럽트를 처리하는 방법.

도면

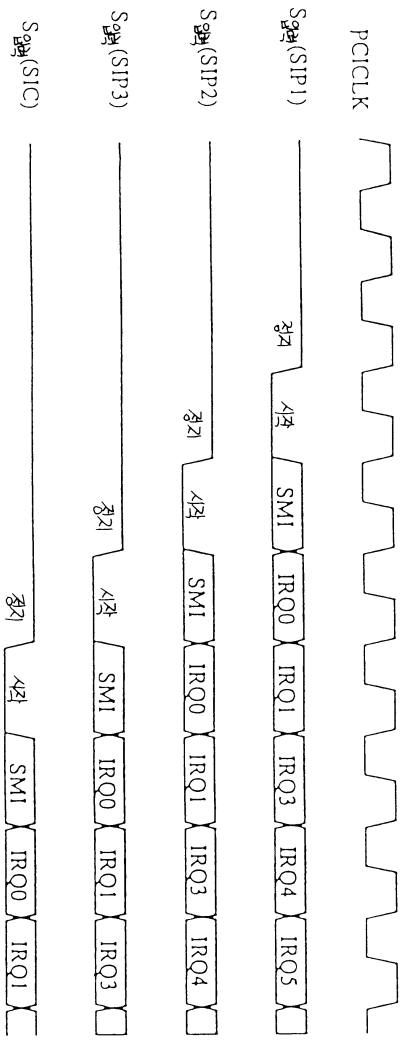
도면1


S출력
32
SIP 논리
S입력
S출력
31
SIP 논리
S입력
S출력
30
SIP 논리
S입력
S출력2
CLKRUN#/PCICLK
시스템 인터럽트
15
22
PIC
15
시스템 입/출력
CPU
인터럽트
SMI
10
메모리 버스
15
SIC
25
20
S입력
S출력
S출력
SIP 논리
S입력
42
S출력
SIP 논리
S입력
41
S입력2
S출력1
45
개별 인터럽트 브리지
40
S입력1

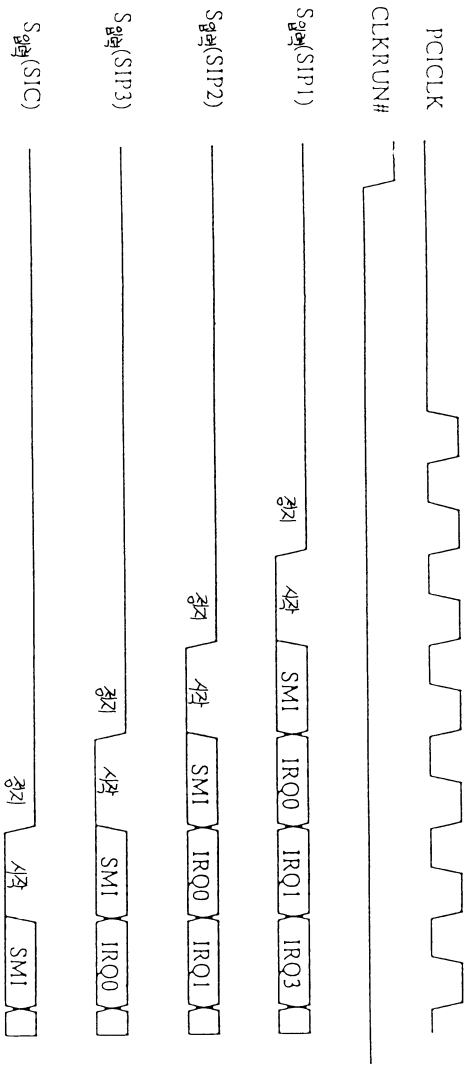
도면2


IRQ0
IRQ1
IRQ2
IRQ3
IRQ4
IRQ5
IRQ6
IRQ7
IRQ8
IRQ9
IRQ10
IRQ11
IRQ12
IRQ13
IRQ14
IRQ15
START
시작 MSG
정지
수행=0
수행=1
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
220

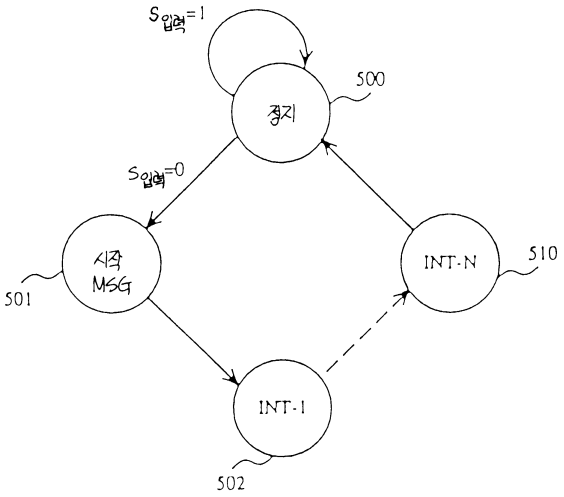
도면3


PCICLK
S입력(SIP1)
정지
시작
SMI
IRQ0
IRQ1
IRQ3
IRQ4
IRQ5
S입력(SIP2)
정지
시작
SMI
IRQ0
IRQ1
IRQ3
IRQ4
S입력(SIP3)
정지
시작
SMI
IRQ0
IRQ1
IRQ3
S입력(SIC)
정지
시작
SMI
IRQ0
IRQ1

도면4


PCICLK
CLKRUN#
S입력(SIP1)
S입력(SIP2)
S입력(SIP3)
S입력(SIC)
정지
시작
SMI
IRQ0
IRQ1
IRQ3

도면5


S입력=1
500
정지
S입력=0
시작
MSG
501
INT-N
510
INT-1
502