



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년10월27일

(11) 등록번호 10-1563576

(24) 등록일자 2015년10월21일

(51) 국제특허분류(Int. Cl.)

G06F 13/24 (2006.01)

(21) 출원번호 10-2014-7012326

(22) 출원일자(국제) 2012년10월04일

심사청구일자 2014년05월07일

(85) 번역문제출일자 2014년05월07일

(65) 공개번호 10-2014-0082787

(43) 공개일자 2014년07월02일

(86) 국제출원번호 PCT/US2012/058780

(87) 국제공개번호 WO 2013/052684

국제공개일자 2013년04월11일

(30) 우선권주장

13/252,670 2011년10월04일 미국(US)

(56) 선행기술조사문헌

US20010052043 A1

(73) 특허권자

퀄컴 인코포레이티드

미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775

(72) 발명자

벤큐마한티, 수례쉬 케이.

미국 92121 캘리포니아주 샌 디에고 모어하우스 드라이브 5775

코드레스쿠, 루시안

미국 92121 캘리포니아주 샌 디에고 모어하우스 드라이브 5775

(뒷면에 계속)

(74) 대리인

특허법인 남앤드남

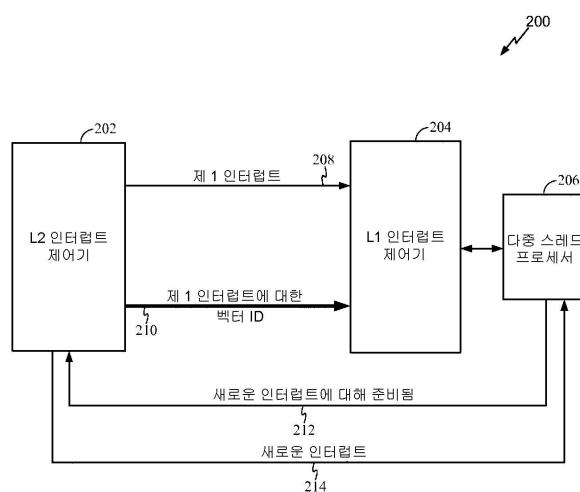
전체 청구항 수 : 총 29 항

심사관 : 고재용

(54) 발명의 명칭 다중 스레드 프로세서에 대한 저 레이턴시 2 레벨 인터럽트 제어기 인터페이스

**(57) 요약**

다중 스레드 프로세서에서 인터럽트 레이턴시 시간을 감소시키기 위한 시스템들 및 방법이 제공된다. 제 1 인터럽트 제어기는 다중 스레드 프로세서에 연결된다. 제 2 인터럽트 제어기는 제 1 인터럽트 및 제 1 벡터식별자를 제 1 인터럽트 제어기에 전달하도록 구성되며, 여기서 제 1 인터럽트 제어기는 제 1 인터럽트 및 제 1 벡터식별자를 처리하고 처리된 인터럽트를 다중 스레드 프로세서의 스레드에 전송하도록 구성된다. 다중 스레드 프로세서가 언제 제 2 인터럽트를 수신할 준비가 되는지를 결정하도록 로직이 구성된다. 다중 스레드 프로세서가 제 2 인터럽트를 수신할 준비가 되었다는 표시를 제 2 인터럽트 제어기에 전달하기 위해 전용 라인이 사용된다.

**대 표 도 - 도2**

(72) 발명자

**프론드케, 에리히 제임스**

미국 92121 캘리포니아주 샌 디에고 모어하우스 드  
라이브 5775

**챈, 수영**

미국 92121 캘리포니아주 샌 디에고 모어하우스 드  
라이브 5775

---

**종, 페이신**

미국 92121 캘리포니아주 샌 디에고 모어하우스 드  
라이브 5775

## 명세서

### 청구범위

#### 청구항 1

인터럽트 레이턴시 시간을 감소시키기 위한 방법으로서,  
제 1 인터럽트 및 제 1 벡터 식별자를 제 2 인터럽트 제어기로부터 제 1 인터럽트 제어기로 전달하는 단계;  
상기 제 1 인터럽트 제어기에서 상기 제 1 인터럽트 및 상기 제 1 벡터 식별자를 처리하는 단계;  
처리된 인터럽트를 상기 제 1 인터럽트 제어기로부터 코어의 스레드로 전송하는 단계;  
상기 코어가 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하는 단계; 및  
상기 제 1 인터럽트가 상기 코어에 의한 처리를 위해 받아들여졌음을 상기 제 1 인터럽트 제어기에 나타내는 단일 명령을 상기 코어로부터 상기 제 1 인터럽트 제어기에 전송하고, 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 상기 단일 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하는 단계를 포함하는,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 2

제 1 항에 있어서,  
상기 단일 명령은 상기 코어로부터 상기 제 2 인터럽트 제어기로의 전용 라인을 통해 전송되는,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 3

제 1 항에 있어서,  
상기 제 2 인터럽트 제어기는 레벨 2 인터럽트 제어기이고,  
상기 제 1 인터럽트 제어기는 레벨 1 인터럽트 제어기인,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 4

제 1 항에 있어서,  
상기 단일 명령은 상기 제 1 인터럽트가 상기 제 1 인터럽트 제어기에서 보류(pending)중임을 나타내는 상태를 클리어(clear)함으로써 동작하는 CIAD(Clear Interrupt Auto Disable) 명령인,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 5

제 4 항에 있어서,  
상기 CIAD(Clear Interrupt Auto Disable) 명령은 상기 제 2 인터럽트 제어기, 상기 제 1 인터럽트 제어기 및 상기 코어 사이의 핸드쉐이크(handshake) 메커니즘인,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 6

제 1 항에 있어서,

상기 코어는 다중 스레드 프로세서인,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 7

제 6 항에 있어서,  
상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하는 단계는, 상기 제 2 인터럽트를 서비스 할 유휴 스레드를 식별하는 단계를 포함하는,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 8

제 6 항에 있어서,  
상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하는 단계는, 상기 제 2 인터럽트를 서비스 할, 최저 우선순위를 갖는 스레드를 식별하는 단계를 포함하는,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 9

제 1 항에 있어서,  
상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 상기 단일 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하는 단계는,  
상기 코어가 상기 제 2 인터럽트를 즉시 수신할 준비가 되었음을 나타내는 상기 단일 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하는 단계를 포함하는,  
인터럽트 레이턴시 시간을 감소시키기 위한 방법.

#### 청구항 10

다중 스레드 프로세서로서,  
코어;  
레벨 1 인터럽트 제어기;  
레벨 2 인터럽트 제어기; 및  
상기 코어를 상기 레벨 2 인터럽트 제어기에 연결하는 라인을 포함하며,  
상기 코어는, 상기 코어가 레벨 2 인터럽트를 수신할 준비가 되었음을 상기 라인을 통해 상기 레벨 2 인터럽트 제어기에 나타내도록 구성되고,  
상기 코어는 레벨 1 인터럽트가 상기 코어에 의한 처리를 위해 받아들여졌음을 나타내는 단일 명령을 상기 레벨 1 인터럽트 제어기에 전송하고, 상기 코어가 상기 레벨 2 인터럽트를 수신할 준비가 되었음을 나타내는 상기 단일 명령을 상기 레벨 2 인터럽트 제어기에 전송하도록 구성되는,  
다중 스레드 프로세서.

#### 청구항 11

제 10 항에 있어서,  
상기 레벨 2 인터럽트 제어기로부터 상기 레벨 2 인터럽트를 수신하도록 구성된 전용 하드웨어 포트를 더 포함하는,  
다중 스레드 프로세서.

#### 청구항 12

제 10 항에 있어서,

상기 레벨 2 인터럽트 제어기는, 상기 레벨 1 인터럽트 및 제 1 벡터 식별자를 상기 레벨 1 인터럽트 제어기에 전달하도록 구성되는,

다중 스레드 프로세서.

### 청구항 13

제 12 항에 있어서,

상기 레벨 1 인터럽트 제어기는, 상기 레벨 1 인터럽트 및 제 1 벡터 식별자를 처리하여 상기 레벨 1 인터럽트를 상기 코어에 전달하도록 구성되는,

다중 스레드 프로세서.

### 청구항 14

제 13 항에 있어서,

상기 코어는, 상기 코어에 전달되는 상기 레벨 1 인터럽트를 기초로, 상기 코어가 레벨 2 인터럽트를 수신할 준비가 되었음을 결정하도록 구성되는,

다중 스레드 프로세서.

### 청구항 15

제 12 항에 있어서,

상기 코어는, CIAD(Clear Interrupt Auto Disable) 명령에 의해 상기 코어가 상기 레벨 2 인터럽트를 수신할 준비가 되었음을 상기 라인을 통해 나타내는,

다중 스레드 프로세서.

### 청구항 16

제 10 항에 있어서,

상기 코어는, 상기 레벨 2 인터럽트를 서비스할 상기 코어의 스레드를 식별하도록 구성된 로직을 통해 상기 코어가 상기 레벨 2 인터럽트를 수신할 준비가 되었음을 결정하도록 구성되는,

다중 스레드 프로세서.

### 청구항 17

제 16 항에 있어서,

상기 로직은 레벨 2 인터럽트를 서비스할, 최저 우선순위를 갖는 스레드를 식별하도록 구성되는,

다중 스레드 프로세서.

### 청구항 18

제 16 항에 있어서,

상기 로직은 레벨 2 인터럽트를 서비스할 유휴 스레드를 식별하도록 구성되는,

다중 스레드 프로세서.

### 청구항 19

제 10 항에 있어서,

적어도 하나의 반도체 다이에 집적되는,

다중 스레드 프로세서.

### 청구항 20

제 10 항에 있어서,

셋톱박스, 뮤직 플레이어, 비디오 플레이어, 엔터테인먼트 유닛, 네비게이션 디바이스, 통신 디바이스, 개인용 디지털 보조기기(PDA: personal digital assistant), 고정 위치 데이터 유닛 및 컴퓨터로 구성된 그룹으로부터 선택된 디바이스에 접속되는,

다중 스레드 프로세서.

### 청구항 21

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템으로서,

코어에 연결된 제 1 인터럽트 제어기;

제 1 인터럽트 및 제 1 벡터 식별자를 제 2 인터럽트 제어기로부터 상기 제 1 인터럽트 제어기로 전달하기 위한 수단;

상기 제 1 인터럽트 제어기에서 상기 제 1 인터럽트 및 상기 제 1 벡터 식별자를 처리하기 위한 수단;

처리된 인터럽트를 상기 코어의 스레드로 전송하기 위한 수단;

상기 코어가 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하기 위한 수단; 및

상기 제 1 인터럽트가 상기 코어에 의한 처리를 위해 받아들여졌음을 상기 제 1 인터럽트 제어기에 나타내는 단일 명령을 상기 코어로부터 상기 제 1 인터럽트 제어기에 전송하고, 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 상기 단일 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하기 위한 수단을 포함하는,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

### 청구항 22

제 21 항에 있어서,

상기 단일 명령을 전송하기 위한 수단은 상기 코어로부터 상기 제 2 인터럽트 제어기로의 전용 라인을 포함하는,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

### 청구항 23

제 21 항에 있어서,

상기 단일 명령은 CIAD(Clear Interrupt Auto Disable) 명령인,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

### 청구항 24

제 21 항에 있어서,

상기 코어는 다중 스레드 프로세서인,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

### 청구항 25

제 24 항에 있어서,

상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하기 위한 수단은, 상기 제 2 인터럽트를 서비스할 유휴 스레드를 식별하기 위한 수단을 포함하는,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

**청구항 26**

제 24 항에 있어서,

상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하기 위한 수단은, 상기 제 2 인터럽트를 서비스할, 최저 우선순위를 갖는 스레드를 식별하기 위한 수단을 포함하는,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

**청구항 27**

제 21 항에 있어서,

적어도 하나의 반도체 다이에 집적되는,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

**청구항 28**

제 21 항에 있어서,

셋톱박스, 뮤직 플레이어, 비디오 플레이어, 엔터테인먼트 유닛, 네비게이션 디바이스, 통신 디바이스, 개인용 디지털 보조기기(PDA), 고정 위치 데이터 유닛 및 컴퓨터로 구성된 그룹으로부터 선택된 디바이스에 집적되는,

감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템.

**청구항 29**

프로세서에 의해 실행될 때, 상기 프로세서로 하여금 인터럽트 레이턴시 시간을 감소시키기 위한 동작들을 수행하게 하는 코드를 포함하는 비-일시적 컴퓨터 판독 가능 저장 매체로서,

제 1 인터럽트 및 제 1 벡터 식별자를 제 2 인터럽트 제어기로부터 제 1 인터럽트 제어기로 전달하기 위한 코드;

상기 제 1 인터럽트 제어기에서 상기 제 1 인터럽트 및 상기 제 1 벡터 식별자를 처리하기 위한 코드;

처리된 인터럽트를 상기 제 1 인터럽트 제어기로부터 코어의 스레드로 전송하기 위한 코드;

상기 코어가 제 2 인터럽트를 수신할 준비가 되는 시점을 결정하기 위한 코드; 및

상기 제 1 인터럽트가 상기 코어에 의한 처리를 위해 받아들여졌음을 상기 제 1 인터럽트 제어기에 나타내는 단일 명령을 상기 코어로부터 상기 제 1 인터럽트 제어기에 전송하고, 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 상기 단일 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하기 위한 코드를 포함하는,

비-일시적 컴퓨터 판독 가능 저장 매체.

**발명의 설명****기술 분야**

[0001] 개시된 실시예들은 프로세서들에서의 인터럽트들을 처리하기 위한 기술들에 관한 것이다. 보다 구체적으로, 예시적인 실시예들은 다중 스레드 프로세서들에 대해 구성된 2 레벨 인터럽트 제어기들에서 인터럽트 레이턴시를 감소시키기 위한 시스템들 및 방법들에 관한 것이다.

**배경기술**

[0002] 처리 시스템들은 대개 인터럽트 메커니즘들을 지원하는데, 여기서 인터럽트는 인터럽트가 서비스될 수 있도록 프로세서의 현재 실행 스레드 또는 명령 스트림을 비동기식으로 중단 또는 보류할 수 있다. 인터럽트는 온-칩(on-chip) 또는 오프-칩(off-chip) 외부 디바이스들을 비롯하여 다양한 소스들로부터 발생될 수 있다. 인터럽트들은 또한, 예컨대 다중 스레드 프로세서에서 하나 또는 그보다 많은 스레드들로부터 프로세서 또는 CPU 내에서 내부적으로 발생될 수도 있다.

[0003]

인터럽트를 서비스하기 위해, 인터럽트를 수신하는 프로세서에 의해 인터럽트 서비스 루틴(ISR: interrupt service routine)이 실행될 수 있다. 각각의 인터럽트는 인터럽트와 연관된 특정 ISR을 포함할 수 있다. 인터럽트들은 다양한 소스들로부터 수신될 수 있기 때문에, 인터럽트 제어기는 대개, 인터럽트들을 수신하고, 여러 개의 아직 처리되지 않은(outstanding) 인터럽트들 사이에 우선순위를 정하고, 새로운 인터럽트들을 처리하기 위한 프로세서의 이용 가능성이 확인될 수 있도록 보류(pending) 인터럽트들의 상태를 추적하는 작업들을 처리하는데 사용된다. 인터럽트들 그리고 연관된 소스들 및 ISR들을 파악하기 위해, 각각의 인터럽트와 연관된 벡터 어드레스들을 추적하기 위한 벡터 인터럽트 제어기(VIC: vectored interrupt controller)가 해당 기술분야에 공지되어 있는데, VIC는 연관된 ISR을 가진 인터럽트를 서비스하는 프로세서를 제공하도록 인에이블될 수 있다.

[0004]

2개 또는 그보다 많은 스레드들을 병렬로 실행하도록 구성된 다중 스레드 프로세서들의 경우, 인터럽트를 서비스하기 위해 어느 스레드가 인터럽트되어야 하는지를 인터럽트 제어기가 결정하도록, 스레드들에 우선순위 레벨들이 동적으로 또는 정적으로 할당될 수 있다. 제 1 레벨 또는 L1 인터럽트 제어기가 예를 들어, 다중 스레드 프로세서와 같은 프로세서 코어와 관련된 인터럽트들을 처리하도록 구성될 수 있다. 제 2 레벨 또는 L2 인터럽트 제어기가 예를 들어, 외부 디바이스들로부터의 인터럽트들 또는 전역 범위(global scale)의 인터럽트들을 처리하도록 구성될 수 있다. L2 인터럽트 제어기는 AHB/AXI와 같은 시스템 버스들을 통해 L1 인터럽트 제어기와 통신하여 그에 따라 L2 인터럽트 제어기에서 L1 인터럽트 제어기로 인터럽트들을 전달할 수 있다. 해당 기술분야에서 통상의 지식을 가진 자에 의해 인식되는 바와 같이, L1 인터럽트 제어기와 L2 인터럽트 제어기와 같은 2 레벨 인터럽트 제어기들은 처리 시스템들에서 여러 다른 애플리케이션들을 발견할 수 있다.

[0005]

도 1을 참조하면, 2 레벨 인터럽트 제어기의 종래의 구현이 제공된다. L2 인터럽트 제어기(102)는 버스(108)를 통해 L1 인터럽트 제어기(104)에 인터럽트를 전달할 수 있으며, L1 인터럽트 제어기(104)는 코어(106)에 부착될 수 있다. 도시된 바와 같이, 코어(106)는 L1 인터럽트 제어기(104)와만 직접 통신하고, L2 인터럽트 제어기(102)와는 직접 통신하지 않는다. 처음에, L1 인터럽트 제어기(104)는 L2 인터럽트 제어기(102)로부터 제 1 인터럽트를 수신할 수 있다. 그 후에, 이후의 인터럽트들의 처리는 예를 들어, 프로세서 자원들을 기초로 두 가지 방법들 중 하나로 처리될 수 있다.

[0006]

첫 번째 시나리오에서, 제 1 인터럽트의 수신 즉시, 코어(106)는 코어(106)가 새로운 인터럽트에 대해 준비되었다는 표시를 L1 인터럽트 제어기(104)를 통해 L2 인터럽트 제어기(102)에 제공할 수 있다. 그 다음, L2 인터럽트 제어기(102)에서 제 2 인터럽트가 보류중이라면, L2 인터럽트 제어기(102)는 프로세서 코어에 제 2 인터럽트를 전송할 수 있다. 예컨대, 코어(106)가 다중 스레드 프로세서로서 구성된다면, 제 1 인터럽트는 다중 스레드 프로세서의 제 1 스레드에 의해 서비스될 수 있으며, 제 2 스레드는 대기(WAIT) 상태이며 제 2 인터럽트를 처리하는데 이용 가능할 수 있다. 이 경우, 프로세서 코어가 제 1 인터럽트를 수신한 직후, 다중 스레드 프로세서는 L2 인터럽트 제어기(102)가 제 2 인터럽트를 전송할 수 있다는 표시를, 예를 들어 L1 인터럽트 제어기(104)를 통해 L2 인터럽트 제어기(102)에 제공할 수 있다.

[0007]

대안으로, 두 번째 시나리오에서, 코어(106)는 이후의 시점까지 또는 추가 통보까지 임의의 새로운 요청들의 전송을 미룬다는 표시를 L2 인터럽트 제어기(102)에 제공할 수 있다. 다시 한번, 코어(106)가 다중 스레드 프로세서로서 구성된다면, 모든 스레드들이 사용중일 수도 있고, 프로세서 코어와 연관된 실시간 동작 시스템(RTOS: real-time operating system)은 어느 스레드를 인터럽트할지를 결정하기 위해 시간 지연을 필요로 할 수도 있다. 예를 들어, RTOS는 L1 인터럽트 제어기(104)가 L2 인터럽트 제어기(102)로부터의 제 2 인터럽트를 최저 우선순위 소프트웨어 스레드로 전달할 수 있도록, 어느 하드웨어 스레드가 가장 낮은 우선순위를 갖는 소프트웨어 스레드를 실행하고 있는지를 결정하여 그 스레드를 최저 우선순위 소프트웨어 스레드로서 지정할 수 있다. 최저 우선순위 소프트웨어 스레드의 결정은 상당한 시간 지연을 초래할 수 있으며, 이에 대응하여 인터럽트들이 처리될 수 있는 레이트가 저하를 겪게 된다.

[0008]

더욱이, 종래의 데이터 처리 시스템들에서는, 인터럽트의 ISR과 연관된 벡터 어드레스들과 같은 인터럽트들에 관한 정보가 어드밴스드 마이크로컨트롤러 버스 아키텍처 고성능 버스(AHB: Advanced Microcontroller Bus Architecture High Performance Bus)를 통해 L2 인터럽트 제어기(102)와 L1 인터럽트 제어기(104) 사이에 전달된다. 상기 정보를 리트리브하기 위한 AHB의 판독과 연관된 프로세스는 인터럽트 레이턴시에 상당한 지연을 부가하여, 인터럽트들이 처리되는 레이트에 추가 영향을 줄 수 있다.

[0009]

종래의 인터럽트 처리와 연관된 앞서 언급한 문제들을 완화하기 위해, 저 레이턴시 2 레벨 인터럽트 제어기들을 포함하는 해결책들이 해당 기술분야에 필요하다.

### 발명의 내용

[0010] 본 발명의 예시적인 실시예들은 다중 스레드 프로세서들에 대해 구성된 2 레벨 인터럽트 제어기들에서 인터럽트 레이턴시를 감소시키기 위한 시스템들 및 방법에 관한 것이다.

[0011] 예를 들어, 예시적인 실시예는 인터럽트 레이턴시 시간을 감소시키기 위한 방법에 관한 것으로, 이 방법은 제 1 인터럽트 및 제 1 벡터 식별자를 제 2 인터럽트 제어기로부터 제 1 인터럽트 제어기로 전달하는 단계; 상기 제 1 인터럽트 제어기에서 상기 제 1 인터럽트 및 상기 제 1 벡터 식별자를 처리하는 단계; 처리된 인터럽트를 상기 제 1 인터럽트 제어기로부터 코어의 스레드로 전송하는 단계; 상기 코어가 언제 제 2 인터럽트를 수신할 준비가 되는지를 결정하는 단계; 및 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하는 단계를 포함한다.

[0012] 다른 예시적인 실시예는 다중 스레드 프로세서에 관한 것으로, 이 다중 스레드 프로세서는 코어, 레벨 2 인터럽트 제어기, 및 상기 코어를 상기 레벨 2 인터럽트 제어기에 연결하는 라인을 포함하고, 여기서 상기 코어는, 상기 코어가 레벨 2 인터럽트를 수신할 준비가 되었음을 상기 라인을 통해 상기 레벨 2 인터럽트 제어기에 나타내도록 구성된다.

[0013] 또 다른 예시적인 실시예는 감소된 인터럽트 레이턴시를 위해 구성된 처리 시스템에 관한 것으로, 이 처리 시스템은 코어에 연결된 제 1 인터럽트 제어기; 제 1 인터럽트 및 제 1 벡터 식별자를 제 2 인터럽트 제어기로부터 상기 제 1 인터럽트 제어기로 전달하기 위한 수단; 상기 제 1 인터럽트 제어기에서 상기 제 1 인터럽트 및 상기 제 1 벡터 식별자를 처리하기 위한 수단; 처리된 인터럽트를 상기 코어의 스레드로 전송하기 위한 수단; 상기 코어가 언제 제 2 인터럽트를 수신할 준비가 되는지를 결정하기 위한 수단; 및 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하기 위한 수단을 포함한다.

[0014] 다른 예시적인 실시예는 프로세서에 의해 실행될 때, 상기 프로세서로 하여금 인터럽트 레이턴시 시간을 감소시키기 위한 동작들을 수행하게 하는 코드를 포함하는 비-일시적 컴퓨터 판독 가능 저장 매체에 관한 것으로, 비-일시적 컴퓨터 판독 가능 저장 매체는, 제 1 인터럽트 및 제 1 벡터 식별자를 제 2 인터럽트 제어기로부터 제 1 인터럽트 제어기로 전달하기 위한 코드; 상기 제 1 인터럽트 제어기에서 상기 제 1 인터럽트 및 상기 제 1 벡터 식별자를 처리하기 위한 코드; 처리된 인터럽트를 상기 제 1 인터럽트 제어기로부터 코어의 스레드로 전송하기 위한 코드; 상기 코어가 언제 제 2 인터럽트를 수신할 준비가 되는지를 결정하기 위한 코드; 및 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하기 위한 코드를 포함한다.

### 도면의 간단한 설명

[0015] 첨부 도면들은 본 발명의 실시예들의 설명에 도움이 되도록 제시되며 실시예들의 한정이 아닌 단지 실시예들의 설명을 위해서만 제공된다.

도 1은 코어에 대한 종래의 2 레벨 인터럽트 제어기 인터페이스를 나타낸다.

도 2는 예시적인 실시예들에 따라 구성된 다중 스레드 프로세서에 대한 2 레벨 인터럽트 제어기 인터페이스를 나타낸다.

도 3은 예시적인 실시예들에 따라 다중 스레드 프로세서에 대한 2 레벨 인터럽트 제어기 인터페이스를 구성하기 위한 방법을 상술하는 흐름도를 나타낸다.

도 4는 본 개시의 실시예가 유리하게 이용될 수 있는 예시적인 무선 통신 시스템(400)을 나타낸다.

### 발명을 실시하기 위한 구체적인 내용

[0016] 본 발명의 양상들이 본 발명의 특정 실시예들에 관련한 다음의 설명 및 관련 도면들에서 개시된다. 본 발명의 범위를 벗어나지 않으면서 대안적인 실시예들이 안출될 수 있다. 추가로, 본 발명의 잘 알려진 엘리먼트들은 본 발명의 관련 세부사항들을 모호하게 하지 않도록 상세히 설명되지 않거나 생략될 것이다.

[0017] "예시적인"이라는 단어는 본 명세서에서 "예시, 실례 또는 예증으로서의 역할"을 의미하는데 사용된다. 본 명세서에서 "예시적인" 것으로서 설명되는 어떠한 실시예도 다른 실시예들보다 반드시 선호되거나 유리한 것으로 해석되는 것은 아니다. 마찬가지로, "본 발명의 실시예들"이라는 용어는 본 발명의 모든 실시예들이 논의되는

특징, 이점 또는 동작 모드를 포함할 것을 요구하는 것은 아니다.

[0018] 본 명세서에서 사용되는 용어는 단지 특정 실시예들을 설명할 목적이 뿐이며 본 발명의 실시예들의 한정인 것으로 의도되지는 않는다. 본 명세서에서 사용되는 바와 같이, "하나의"라는 단수 형태들은 문맥상 명백하게 달리 표시하지 않는 한, 복수 형태들도 역시 포함하는 것으로 의도된다. 또한, "포함하는"("comprises", "comprising, ", "includes" 및/또는 "including")이라는 용어들은 본 명세서에서 사용될 때, 언급되는 특징들, 정수들, 단계들, 동작들, 엘리먼트들 및/또는 컴포넌트들의 존재를 명시하지만, 하나 또는 그보다 많은 다른 특징들, 정수들, 단계들, 동작들, 엘리먼트들, 컴포넌트들, 및/또는 이들의 그룹들의 존재 또는 부가를 배제하는 것은 아니라고 이해될 것이다.

[0019] 또한, 많은 실시예들은 예를 들어, 컴퓨팅 디바이스의 엘리먼트들에 의해 수행될 동작들의 시퀀스들에 관하여 설명된다. 본 명세서에서 설명되는 다양한 동작들은 특정 회로들(예를 들어, 주문형 집적 회로(ASIC: application specific integrated circuit)들)에 의해, 하나 또는 그보다 많은 프로세서들에 의해 실행되는 프로그램 명령들에 의해, 또는 이 둘의 결합에 의해 수행될 수 있다고 인식될 것이다. 추가로, 본 명세서에서 설명되는 이러한 동작들의 시퀀스는, 실행시 연관된 프로세서로 하여금 본 명세서에서 설명된 기능을 수행하게 할 대응하는 세트의 컴퓨터 명령들을 저장하고 있는 임의의 형태의 컴퓨터 관독 가능 저장 매체 내에 전부 구현되는 것으로 여겨질 수 있다. 따라서 본 발명의 다양한 양상들은, 전부 본 발명의 청구대상의 범위 내에 있다고 여겨진 다수의 다양한 형태들로 구현될 수 있다. 또한, 본 명세서에서 설명되는 실시예들 각각에 대해, 임의의 이러한 실시예들의 대응하는 형태는 본 명세서에서, 예를 들어 설명되는 동작을 수행"하도록 구성된 로직"으로서 설명될 수 있다.

[0020] 도 1을 참조로 앞서 설명한 바와 같이, 종래의 2 레벨 인터럽트 제어기들은 결점들을 겪는다. 코어(106)가 새로운 인터럽트를 받을 준비가 되면, 그러한 표시가 AHB 버스와 같은 긴 레이턴시의 네트워크(110)를 거쳐 L1 인터럽트 제어기(104)를 통해 L2 인터럽트 제어기(102)에 제공된다. 인터럽트의 이러한 종래의 처리는 심각한 지연들을 초래하며 낮은 레이트의 인터럽트 처리로 이어진다.

[0021] 상기 종래의 기술들에 반해, 예시적인 실시예들은 높은 레이트의 인터럽트 처리를 위해 구성된 저 레이턴시 인터럽트 제어기들에 관한 것이다. 보다 구체적으로, 실시예들은 다중 스레드 프로세서 코어들에 대해 인터페이스될 수 있는 2 레벨 저 레이턴시 인터럽트 제어기들을 포함할 수 있다.

[0022] 이제 도 2를 참조하면, 다중 스레드 프로세서(206)와 직접 인터페이스하도록 구성될 수 있는 L1 인터럽트 제어기(204)를 포함하는 시스템(200)이 예시된다. L1 인터럽트 제어기(204)는 L2 인터럽트 제어기(202)를 포함하는 하나 또는 그보다 많은 디바이스들로부터 다중 스레드 프로세서로 전달되는 인터럽트들을 처리할 수 있다.

[0023] 한 실시예에서, L1 인터럽트 제어기(204)와 L2 인터럽트 제어기(202) 모두, 앞서 설명한 것과 같은 벡터 인터럽트 제어기일 수 있다. 따라서 L2 인터럽트 제어기(202)는 인터럽트들의 ISR들과 연관된 벡터 어드레스들과 함께 인터럽트들을 전송하도록 구성될 수 있다. 한정적이지 않은 예시에 따르면, L2 인터럽트 제어기(202)는 저 레이턴시 인터럽트들을 1024개까지 지원할 수 있다. 1024개의 저 레이턴시 인터럽트들은 L2 인터럽트 제어기(202)에 의해 우선순위가 부여될 수 있다. L1 인터럽트 제어기(204)는, L1 인터럽트 제어기(204)의 레지스터 엔트리 [31]이 L2 인터럽트 제어기(202)에 대응할 수 있도록, 32개의 레지스터 엔트리들 [31:0]을 갖는 VIC일 수 있다. L1 인터럽트 제어기(204)의 나머지 31개의 레지스터 엔트리들은 스레드 간 시그널링을 위해, 또는 레거시 애플리케이션들을 위해 다중 스레드 프로세서(206)의 2개 또는 그보다 많은 스레드들로부터 내부적으로 발생된 인터럽트들을 비롯하여, 다른 소스들로부터의 인터럽트들을 위해 확보될 수 있다. L1 인터럽트 제어기(204)와 L2 인터럽트 제어기(202)의 예시된 구성에서, (도시되지 않은) 외부 디바이스들로부터의 모든 인터럽트들은 우선 L2 인터럽트 제어기(202)에서 수신될 수 있고, 그 후에 L1 인터럽트 제어기(204)에 전달될 수 있다.

[0024] 예를 들어, 계속해서 도 2를 참조하면, (도시되지 않은) 외부 디바이스로부터 제 1 인터럽트가 L2 인터럽트 제어기(202)에 수신될 수 있다. 제 1 인터럽트는 버스(210) 상의 대응하는 벡터 ID와 함께, 버스(208)를 통해 L1 인터럽트 제어기(204)에 전달될 수 있다. (도시되지 않은) 전역 벡터 ID 레지스터가 또한 벡터 ID로 업데이트 될 수 있으며, 여기서 전역 벡터 ID 레지스터는 제어 레지스터(CR: control register) 전송 명령을 통해 액세스 가능할 수 있다. 전역 벡터 ID 레지스터는 인터럽트들의 추적을 보조할 수 있다. 예를 들어, 전역 VID 레지스터는 L1 인터럽트 제어기(204)에 어느 특정 L2 인터럽트가 전송되었는지를 추적하도록 구성될 수 있다.

[0025] 일단 L1 인터럽트 제어기(204)에 의해 제 1 인터럽트가 수신되면, 실시예들은 다음 요청들의 처리를 더 신속히 처리하기 위해 여러 가지 방법들로 종래의 기술들에서 벗어날 수 있다. 도 1에 예시된 것과 같은 종래의 기술

들은 새로운 요청의 수락에 대한 코어(106)의 이용 가능성에 관해 네트워크(110)를 통해 L2 인터럽트 제어기(102)에 통보하기 위해 하드웨어 해결책들에 의존하였지만, 실시예들은 다중 스레드 프로세서(206)의 준비 상태를 모니터링하도록 구성된 소프트웨어 루틴들을 포함할 수 있다. 예를 들어, 소프트웨어 루틴은 다중 스레드 프로세서(206)의 2개 또는 그보다 많은 스레드들에 대해 실행하는 프로세스들의 상태를 효과적으로 추적할 수 있다. 소프트웨어 루틴은, 하나 또는 그보다 많은 스레드들이 새로운 인터럽트를 즉시 수락하도록 대기 상태일 수 있는지 여부, 또는 새로운 인터럽트를 즉시 서비스하기 위해 스레드에 대해 실행하는 저 우선순위 프로세스가 인터럽트될 수 있는지 여부를 결정할 수 있다.

[0026] 일단 인터럽트를 수락할 다중 스레드 프로세서(206)의 준비 상태에 관한 결정이 이루어지면, 실시예들은 처리를 위해 제 1 인터럽트가 받아들여졌음을 L1 인터럽트 제어기(204)에 알리는 것 그리고 또한 다중 스레드 프로세서(206)가 이제 새로운 인터럽트를 받을 준비가 되었음을 L2 인터럽트 제어기(202)에 알리는 것 모두를 달성하기 위한 단일 명령을 포함할 수 있다. 이러한 실시예들은 도 1에 예시된 것과 같은 종래의 2 레벨 인터럽트 프레임워크에 의해 저지되지 않는다고 인식될 것이며, 여기서 코어(106)는 새로운 인터럽트에 대한 자신의 준비 상태를 단지 네트워크(110)를 통해서만 L2 인터럽트 제어기(102)에 전달할 수 있다. 다른 한편으로, 실시예들은 도 2의 라인(212)과 같은 전용 하드웨어 라인을 통해 L2 인터럽트 제어기(202)에 직접 다중 스레드 프로세서(206)의 준비 상태를 전달할 수 있다.

[0027] 더욱이, 일부 실시예들은 L2 인터럽트 제어기(202)가 라인(214)과 같은 전용 하드웨어 라인을 통해 다중 스레드 프로세서(206)에 인터럽트들(예를 들어, 도 2의 "새로운 인터럽트")을 직접 전달할 수 있도록, 다중 스레드 프로세서(206)에 전용 하드웨어 포트를 또한 포함할 수도 있다. 이런 식으로, 다중 스레드 프로세서(206)로부터 L2 인터럽트 제어기(202)로의 준비 상태 전달뿐 아니라, L2 인터럽트 제어기(202)로부터 다중 스레드 프로세서(206)로의 인터럽트의 다음의 전달 또한 L1 인터럽트 제어기(204) 및 연관된 지연들을 완전히 피할 수 있다.

[0028] 예시적인 실시예들은 CIAD(Clear Interrupt Auto Disable) 레지스터를 포함할 수 있다. CIAD 명령은, 동일한 인터럽트가 한번만이 아니라 여러 번 받아들여지지 않게 하는데 사용될 수 있다. 예를 들어, 다중 스레드 프로세서(206)는 제 1 인터럽트가 받아들여지자마자 자동으로 CIAD 레지스터를 설정할 수 있다. 일단, 예를 들어 앞서 논의한 바와 같은 소프트웨어 루틴에 의해 다중 스레드 프로세서(206)가 새로운 인터럽트를 받을 준비가 되어 있다고 결정되면, 소프트웨어 루틴은 발생될 CIAD 명령을 트리거할 수 있다. 그 다음, CIAD 명령은 동일한 라인 상에서 새로운 인터럽트가 받아들여질 수 있도록, CIAD 레지스터를 클리어하여 인터럽트 라인을 활성화 할 수 있다.

[0029] 한 구현에서는, L1 인터럽트 제어기(204)에서 제 1 인터럽트가 보류중임을 나타내는 상태를 클리어하고 그리고 또한 다중 스레드 프로세서(206)가 버스들(208, 210)을 통해 다른 인터럽트 및 이에 수반되는 벡터 ID를 각각 수신할 준비가 되어있음을 L2 인터럽트 제어기(202)에 알리도록 다중 스레드 프로세서(206)에 의해 CIAD 명령이 발생될 수 있다. CIAD 명령은 L1 인터럽트 제어기(204)의 레지스터 엔트리 [31]과 연관될 수 있으며, 이는 앞서 설명한 바와 같이, L2 인터럽트 제어기(202)로부터의 인터럽트들에 전용될 수 있다. 따라서 CIAD 명령은 L2 인터럽트 제어기(202), L1 인터럽트 제어기(204) 그리고 다중 스레드 프로세서(206) 간에 효과적인 핸드쉐이크 메커니즘을 제공할 수 있다. 일단 L1 인터럽트 제어기(204)에서 제 1 인터럽트가 수신되면, 소프트웨어 루틴과 같은 프로세스가 시작되어, 다음 인터럽트들의 처리를 더 신속히 처리하기 위해 CIAD 명령을 생성할 수 있다. 실시예들은 또한 앞서 언급한 예에서 논의한 것과 같은 소프트웨어 루틴으로 한정되지 않으면서, 전용 하드웨어 또는 하드웨어와 소프트웨어의 결합으로 상기 프로세스들을 구현할 수도 있다고 이해될 것이다.

[0030] 더욱이, 실시예들은 L1 인터럽트 제어기(204)의 레지스터 엔트리 [31]가 클록의 상승 에지에 대한 정보를 포착하도록 프로그램될 수 있는 구현들을 포함할 수 있다. L2 인터럽트 제어기(202)는 에지 트리거되거나 레벨에 민감할 수 있으며, 인터럽트들 및 대응하는 벡터 ID들을 버스들(208, 210)을 통해 비동기식으로 L1 인터럽트 제어기(204)에 전송할 수 있다. 에지 트리거되는 것으로 L1 인터럽트 제어기(204)를 구성함으로써, 인터럽트들은 다중 스레드 프로세서(206)에 대응하는 클록으로 동기화될 수 있다. 이러한 에지 트리거 구성들은 L1 인터럽트 제어기(204)와 L2 인터럽트 제어기(202) 간의 개선된 통신 프로토콜들을 인에이블할 수 있다.

[0031] 또한, 전용 버스(212)를 통해 전송되는 CIAD 명령과 같은 다중 스레드 프로세서(206)의 준비 상태에 관해 L2 인터럽트 제어기(202)에 알리기 위한 메커니즘들은 제 1 인터럽트의 완료 전에 트리거될 수 있다고 인식될 것이다. 즉, 다중 스레드 프로세서(206)가 새로운 인터럽트를 처리할 준비 단계에 도달하기 위해, 제 1 인터럽트가 다중 스레드 프로세서(206)에 의해 완전히 처리될 필요는 없다. 일부 실시예들에서, L1 인터럽트 제어기에서 제 1 인터럽트가 수신된 직후, 다중 스레드 프로세서(206)는 예를 들어, L1 인터럽트 제어기(204)의 레지

스터 엔트리 [31]을 클리어함으로써 새로운 인터럽트를 받을 준비 상태로의 전이를 시작할 수 있다.

[0032] 이에 따라, 상기 섹션들에서 논의한 바와 같은 하드웨어 및 소프트웨어 구성들의 결합에 의해, 실시예들은 인터럽트들의 처리 레이트를 상당히 개선할 수 있으며, 또한 인터럽트 처리의 레이턴시를 감소시킬 수 있다.

[0033] 또한, 실시예들은 본 명세서에 개시된 프로세스들, 기능들 및/또는 알고리즘들을 수행하기 위한 다양한 방법들을 포함한다고 인식될 것이다. 예를 들어, 도 3에 예시된 바와 같이, 실시예는 인터럽트 레이턴시 시간을 감소시키기 위한 방법을 포함할 수 있으며, 이 방법은 L1 인터럽트 제어기(204)와 같은 제 1 인터럽트 제어기(204)를 다중 스레드 프로세서(206)와 같은 코어에 연결하는 단계(블록(302)); 제 1 인터럽트 및 제 1 벡터 식별자를, 예를 들어 각각 버스들(208, 210)을 통해 L2 인터럽트 제어기(202)와 같은 제 2 인터럽트 제어기로부터 제 1 인터럽트 제어기로 전달하는 단계(블록(304)); 제 1 인터럽트 제어기에서 제 1 인터럽트 및 제 1 벡터 식별자를 처리하는 단계(블록(306)); 처리된 인터럽트를 코어의 스레드에 전송하는 단계(블록(308)); 코어가 언제 제 2 인터럽트를 수신할 준비가 되는지를 결정하는 단계(블록(310)); 및 상기 코어가 상기 제 2 인터럽트를 수신할 준비가 되었음을 나타내는 명령을 상기 코어로부터 상기 제 2 인터럽트 제어기에 전송하는 단계(블록(312))를 포함한다.

[0034] 해당 기술분야에서 통상의 지식을 가진 자들은 정보 및 신호들이 다양한 다른 기술들 및 기법들 중 임의의 것을 이용하여 표현될 수 있음을 이해할 것이다. 예를 들어, 상기 설명 전반에 걸쳐 참조될 수 있는 데이터, 명령들, 명령어들, 정보, 신호들, 비트들, 심벌들 및 칩들은 전압들, 전류들, 전자파들, 자기 필드들 또는 자기 입자들, 광 필드들 또는 광 입자들, 또는 이들의 임의의 결합으로 표현될 수 있다.

[0035] 또한, 해당 기술분야에서 통상의 지식을 가진 자들은, 본 명세서에 개시된 실시예들과 관련하여 설명된 다양한 예시적인 로직 블록들, 모듈들, 회로들 및 알고리즘 단계들이 전자 하드웨어, 컴퓨터 소프트웨어, 또는 이 둘의 결합들로 구현될 수 있다고 인식할 것이다. 하드웨어와 소프트웨어의 이러한 상호 호환성을 명확히 설명하기 위해, 각종 예시적인 컴포넌트들, 블록들, 모듈들, 회로들 및 단계들은 일반적으로 이들의 기능과 관련하여 위에서 설명되었다. 이러한 기능이 하드웨어로 구현되는지 아니면 소프트웨어로 구현되는지는 전체 시스템에 부과된 설계 제약들 및 특정 애플리케이션에 좌우된다. 해당 기술분야에서 통상의 지식을 가진 자들은 설명된 기능을 특정 애플리케이션마다 다양한 방식들로 구현할 수도 있지만, 이러한 구현 결정들이 본 발명의 범위를 벗어나게 하는 것으로 해석되지는 않아야 한다.

[0036] 본 명세서에 개시된 실시예들과 관련하여 설명된 방법들, 시퀀스들 및/또는 알고리즘들은 직접 하드웨어로, 프로세서에 의해 실행되는 소프트웨어 모듈로, 또는 이 둘의 결합으로 구현될 수 있다. 소프트웨어 모듈은 RAM 메모리, 플래시 메모리, ROM 메모리, EEPROM 메모리, EEPROM 메모리, 레지스터들, 하드디스크, 착탈식 디스크, CD-ROM, 또는 해당 기술분야에 공지된 임의의 다른 형태의 저장 매체에 상주할 수 있다. 예시적인 저장 매체는 프로세서가 저장 매체로부터 정보를 읽고 저장 매체에 정보를 기록할 수 있도록 프로세서에 연결된다. 대안으로, 저장 매체는 프로세서에 통합될 수도 있다.

[0037] 이에 따라, 본 발명의 실시예는 다중 스레드 프로세서에 대한 2 레벨 인터럽트 제어기 인터페이스에서 인터럽트 레이턴시를 감소시키기 위한 방법을 구현하는 컴퓨터 판독 가능 매체를 포함할 수 있다. 따라서 본 발명은 설명된 예들로 한정되는 것이 아니며, 본 명세서에서 설명된 기능을 수행하기 위한 임의의 수단이 본 발명의 실시예들에 포함된다.

[0038] 도 4를 참조하면, 예시적인 실시예들에 따라 구성된 다중 코어 프로세서를 포함하는 무선 디바이스의 특정한 예시적인 실시예의 블록도가 도시되며 일반적으로 400으로 명시된다. 디바이스(400)는 도 2의 시스템(200)을 포함할 수 있는 디지털 신호 프로세서(DSP: digital signal processor)(464)를 포함한다. 도 4는 또한 DSP(464)에 그리고 디스플레이(428)에 연결되는 디스플레이 제어기(426)를 보여준다. 코더/디코더(코덱)(CODEC)(434)(예를 들어, 오디오 및/또는 음성 CODEC)가 DSP(464)에 연결될 수 있다. (모뎀을 포함할 수 있는) 무선 제어기(440)와 같은 다른 컴포넌트들이 또한 예시된다. 스피커(436)와 마이크로폰(438)이 코덱(434)에 연결될 수 있다. 도 4는 또한, 무선 제어기(440)가 무선 안테나(442)에 연결될 수 있음을 나타낸다. 특정 실시예에서, DSP(464), 디스플레이 제어기(426), 메모리(432), 코덱(434) 및 무선 제어기(440)는 시스템-인-패키지(system-in-package) 또는 시스템-온-칩(system-on-chip) 디바이스(422)에 포함된다.

[0039] 특정 실시예에서, 입력 디바이스(430) 및 전원(444)이 시스템-온-칩 디바이스(422)에 연결된다. 더욱이, 도 4에 예시된 바와 같이, 특정 실시예에서 디스플레이(428), 입력 디바이스(430), 스피커(436), 마이크로폰(438), 무선 안테나(442) 및 전원(444)은 시스템-온-칩 디바이스(422) 외부에 있다. 그러나 디스플레이(428), 입력 디

바이스(430), 스피커(436), 마이크로폰(438), 무선 안테나(442) 및 전원(444) 각각은 인터페이스나 제어기와 같은 시스템-온-칩 디바이스(422)의 컴포넌트에 연결될 수 있다.

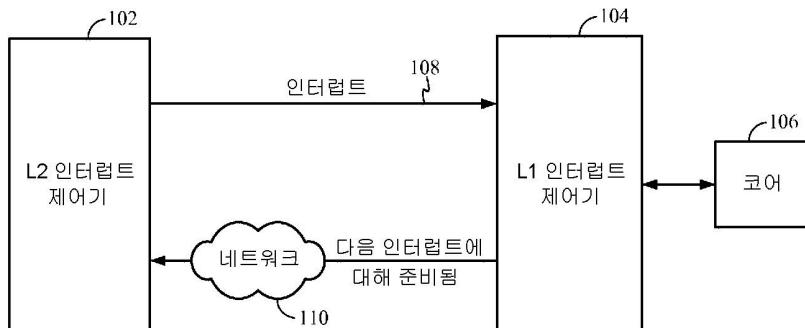
[0040] 도 4는 무선 통신 디바이스를 도시하고 있지만, DSP(464)와 메모리(432)가 또한 셋톱박스, 뮤직 플레이어, 비디오 플레이어, 엔터테인먼트 유닛, 네비게이션 디바이스, 개인용 디지털 보조기기(PDA: personal digital assistant), 고정 위치 데이터 유닛 또는 컴퓨터에 접속될 수도 있다는 점에 주의해야 한다. 프로세서(예를 들어, DSP(464))가 또한 이러한 디바이스에 접속될 수도 있다.

[0041] 위에서 개시된 디바이스들과 방법들은 일반적으로 컴퓨터 판독 가능 매체 상에 저장된 GDSII 및 GERBER 컴퓨터 파일들로 설계 및 구성된다. 차례로 이러한 파일들은 이러한 파일들을 기초로 디바이스들을 제작하는 제작 행들러들에 제공된다. 결과물들은, 이후에 반도체 다이로 컷팅되어 반도체 칩으로 패키징되는 반도체 웨이퍼들이다. 다음에 칩들은 위에서 설명된 디바이스들에 이용된다.

[0042] 상기의 개시는 본 발명의 예시적인 실시예들을 보여주지만, 침부된 청구항들에 의해 정의된 대로 본 발명의 범위를 벗어나지 않으면서 본 명세서에서 다양한 변경들 및 수정들이 이루어질 수 있다는 점에 유의해야 한다. 본 명세서에서 설명된 본 발명의 실시예들에 따른 방법 청구항들의 기능들, 단계들 및/또는 동작들은 어떠한 특정 순서로 수행될 필요는 없다. 더욱이, 본 발명의 엘리먼트들은 단수로 설명 또는 청구될 수 있지만, 단수로의 한정이 명시적으로 언급되지 않는 한 다수가 고려된다.

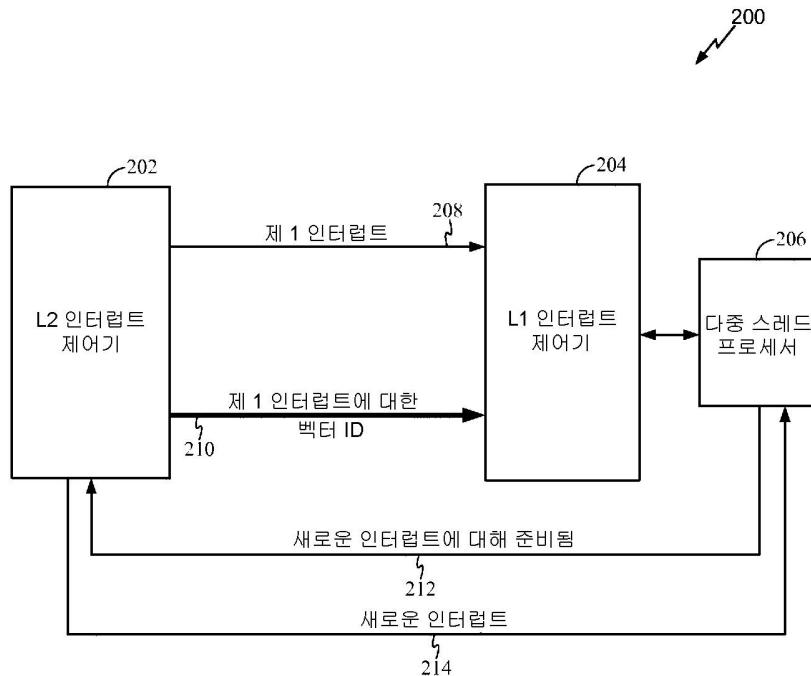
## 도면

### 도면1

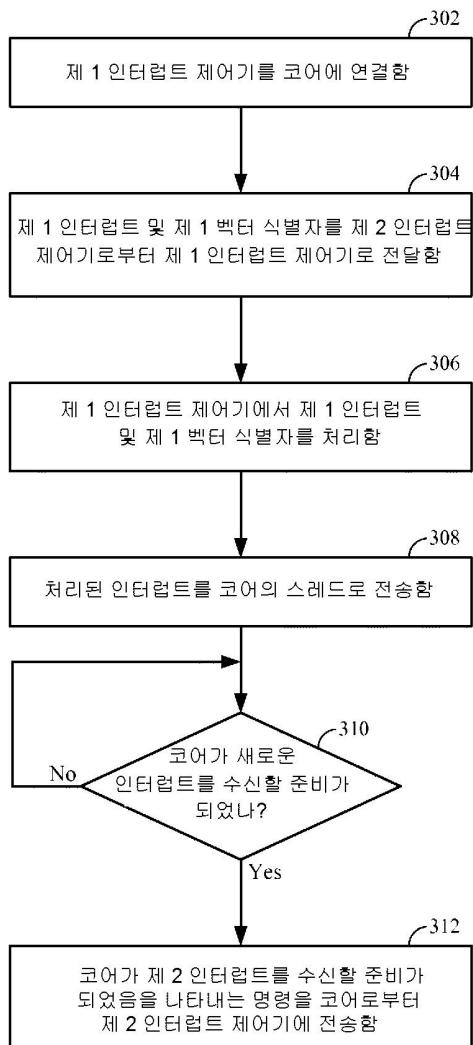


종래 기술

도면2



## 도면3



도면4

