



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0017178
(43) 공개일자 2018년02월20일

(51) 국제특허분류(Int. Cl.)
G06F 13/24 (2006.01)
(52) CPC특허분류
G06F 13/24 (2013.01)
(21) 출원번호 10-2018-7001251
(22) 출원일자(국제) 2016년05월06일
심사청구일자 **없음**
(85) 번역문제출일자 2018년01월15일
(86) 국제출원번호 PCT/GB2016/051310
(87) 국제공개번호 WO 2016/203193
국제공개일자 2016년12월22일
(30) 우선권주장
1510597.6 2015년06월16일 영국(GB)

(71) 출원인
노르딕 세미컨덕터 에이에스에이
노르웨이 엔-7004 트론드헤임 오토 니엘센스 베그
12
(72) 발명자
루스텐 조아르 올라이
노르웨이 7004 트론드헤임 오토 니엘센스 베그 12
노르딕 세미컨덕터
(74) 대리인
제일특허법인

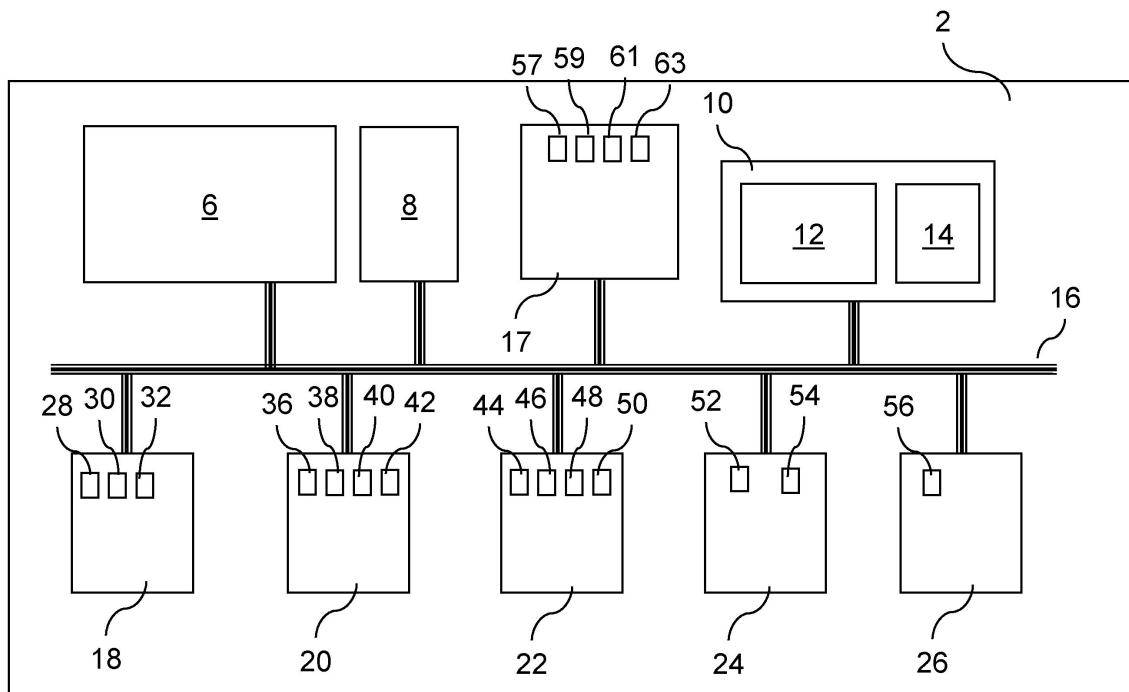
전체 청구항 수 : 총 23 항

(54) 발명의 명칭 **이벤트 생성 유닛**

(57) 요 약

마이크로컨트롤러(2)는 프로세서(6), 주변기기들(18, 20, 22, 24, 26), 프로그램 가능한 주변기기 상호연결부(PP1)(10), 이벤트-생성 유닛(EGU)(17), 및 메모리(8)를 갖는다. 주변기기들은 PP1로부터의 태스크 신호들에 응답한다. EGU는 PP1에 이벤트를 신호함으로써 이벤트-생성 레지스터(57, 59)의 컨텐츠들에 미리 결정된 변화에 응(뒷면에 계속)

대 표 도



답한다. 저장된 PPI 맵핑들은 EGU 이벤트를 주변기기들 중 하나의 태스크에 맵핑할 수 있다. 하나의 EGU 이벤트로부터 두개 이상의 주변기기 태스크들로의 맵핑들은 최대 시간 한계치내에 개별 태스크 신호들을 발송함으로써 PPI가 EGU로부터의 이벤트 신호에 응답하게 한다. 메모리에 소프트웨어는 이런 맵핑들을 맵핑 메모리에 저장하고, 미리 결정된 변화를 이벤트-생성 레지스터의 컨텐츠들에 수행하는 지시들을 포함한다. 다른 측면에서, 인터럽트-생성 유닛(17)은 PPI(10)로부터 태스크 신호를 수신한 것에 응답하여 프로세서(6)에 인터럽트를 발송하도록 배열된다.

명세서

청구범위

청구항 1

마이크로컨트롤러로서,

프로세서;

복수의 주변기기들(peripherals);

프로그램 가능한 주변기기 상호연결부(programmable peripheral interconnect);

이벤트-생성 유닛; 및

소프트웨어를 저장하는 메모리를 포함하되,

상기 이벤트-생성 유닛은 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는(addressable) 이벤트-생성 레지스터를 포함하고;

상기 이벤트-생성 유닛은 상기 프로그램 가능한 주변기기 상호연결부에 연결되고;

상기 이벤트-생성 유닛은 상기 이벤트-생성 레지스터의 컨텐츠들에 대한 미리 결정된 변화를 감지하고 이런 미리 결정된 변화를 감지한 것에 응답하여, 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록 (signal) 배열되고;

상기 주변기기들의 각각은 상기 프로그램 가능한 주변기기 상호연결부에 연결되고;

상기 주변기기들의 각각은 개별 태스크를 수행함으로써 상기 프로그램 가능한 주변기기 상호연결부로부터의 태스크 신호에 응답하도록 구성되고;

상기 프로그램 가능한 주변기기 상호연결부는 복수의 맵핑들이 저장될 수 있는 맵핑 메모리를 액세스하도록 구성되고, 각각은 상기 이벤트-생성 유닛의 이벤트를 상기 주변기기들 중 하나의 태스크에 맵핑시키고;

상기 이벤트-생성 유닛으로부터의 이벤트로부터 상기 주변기기들 중 하나의 태스크로의 맵핑이 상기 맵핑 메모리에 저장될 때, 상기 프로그램 가능한 주변기기 상호연결부가 태스크 신호를 상기 주변기기로 발송함으로써 상기 이벤트-생성 유닛으로부터의 상기 이벤트의 신호에 응답하도록 상기 프로그램 가능한 주변기기 상호연결부가 구성되고;

상기 맵핑 메모리가 상기 이벤트-생성 유닛의 하나의 이벤트로부터 두개 이상의 상이한 태스크들로의 맵핑들을 저장할 때, 상기 프로그램 가능한 주변기기 상호연결부가 상기 이벤트의 신호를 수신한 것으로부터 미리 결정된 최대 시간내에 상기 두개 이상의 개별 태스크 신호들을 발송하도록 상기 프로그램 가능한 주변기기 상호연결부가 구성되고;

상기 소프트웨어는 (i) 상기 맵핑 메모리에, 상기 이벤트-생성 유닛의 하나의 이벤트로부터 적어도 두개의 상이한 주변기기 태스크들로의 맵핑들을 저장하고, (ii) 상기 이벤트-생성 레지스터의 컨텐츠들에 상기 미리 결정된 변화를 수행하도록 상기 프로세서에 의해 실행 가능한 지시들을 포함하는

마이크로컨트롤러.

청구항 2

제 1 항에 있어서,

상기 미리 결정된 최대 시간은 10 마이크로초 또는 그 미만인

마이크로컨트롤러.

청구항 3

제 1 항 또는 제 2 항에 있어서,

상기 프로그램 가능한 주변기기 상호연결부는 상기 두개 이상의 개별 태스크 신호들을 동시에 발송하도록 배열된

마이크로컨트롤러.

청구항 4

제 3 항에 있어서,

모든 가능한 맵핑들에 대하여 이벤트의 신호를 수신하고 상기 이벤트에 맵핑된 두개 이상의 태스크 신호들을 발송하는 상기 프로그램 가능한 주변기기 인터페이스간에 일정한 시간 지연이 있는

마이크로컨트롤러.

청구항 5

제 1 항 내지 제 4 항 중 어느 한 항에 있어서,

이벤트를 생성하고, 이벤트를 상기 프로그램 가능한 주변기기 상호연결부에 신호하도록 구성된 하나 이상의 주변기기들을 포함하고, 제 1 주변기기의 이벤트와 제 2 주변기기의 태스크간의 맵핑이 상기 맵핑 메모리에 저장될 때, 상기 프로그램 가능한 주변기기 상호연결부가 상기 제 1 주변기기로부터 상기 이벤트의 신호에 응답하여 상기 제 2 주변기기로 태스크 신호를 발송함으로써 채널을 제공하도록 상기 프로그램 가능한 주변기기 상호연결부가 배열되는

마이크로컨트롤러.

청구항 6

제 1 항 내지 제 5 항 중 어느 한 항에 있어서,

상기 이벤트-생성 유닛은 복수의 상이한 이벤트들을 신호하도록 배열된

마이크로컨트롤러.

청구항 7

제 6 항에 있어서,

상기 이벤트-생성 유닛은 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 복수의 이벤트-생성 레지스터들을 갖고, 각각의 이벤트-생성 레지스터는 상이한 개별 이벤트와 관련되는

마이크로컨트롤러.

청구항 8

제 1 항 내지 제 7 항 중 어느 한 항에 있어서,

상기 이벤트-생성 유닛은 상기 프로그램 가능한 주변기기 상호연결부로부터 태스크 신호를 수신할 수 있는

마이크로컨트롤러.

청구항 9

제 1 항 내지 제 8 항 중 어느 한 항에 있어서,

상기 프로그램 가능한 주변기기 인터페이스는 각각의 이벤트 및 태스크에 대한 개별 라인들에 의해 상기 주변기기들에 그리고 상기 이벤트-생성 유닛에 연결되는

마이크로컨트롤러.

청구항 10

제 1 항 내지 제 9 항 중 어느 한 항에 있어서,

상기 프로그램 가능한 주변기기 인터페이스는 상기 이벤트-생성 유닛의 이벤트와 관련되고 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 적어도 하나의 이벤트 레지스터를 포함하는

마이크로컨트롤러.

청구항 11

제 10 항에 있어서,

상기 이벤트 레지스터는 상기 이벤트-생성 레지스터로부터 분리되고 상기 이벤트-생성 유닛은 상기 이벤트-생성 레지스터의 컨텐츠들에 대한 미리 결정된 변화가 상기 이벤트 레지스터의 컨텐츠들에 미리 결정된 변화를 유발하도록 구성되는

마이크로컨트롤러.

청구항 12

제 1 항 내지 제 11 항 중 어느 한 항에 있어서,

상기 이벤트-생성 유닛은 복수의 이벤트-생성 레지스터들 및 복수의 이벤트 레지스터들을 포함하고, 각각의 이벤트-생성 레지스터는 상기 이벤트 레지스터들의 개개의 이벤트 레지스터와 관련되어, 상기 이벤트-생성 레지스터들 중 하나에 대한 변화가 상기 이벤트 레지스터들 중 하나와 관련된 변화를 야기하는

마이크로컨트롤러.

청구항 13

제 10 항에 있어서,

상기 이벤트-생성 레지스터가 상기 이벤트 레지스터인

마이크로컨트롤러.

청구항 14

제 1 항 내지 제 13 항 중 어느 한 항에 있어서,

상기 이벤트-생성 유닛은 PPI로 하나 이상의 이벤트 라인들; 상기 프로세서로 제로 또는 그 이상의 인터럽트 라

인들; 및 제로 또는 그 이상의 레지스터들 외에 어떠한 출력들도 갖지 않는 마이크로컨트롤러.

청구항 15

제 1 항 내지 제 14 항 중 어느 한 항에 있어서,
상기 이벤트-생성 유닛은 클럭 입력, 하나 이상의 레지스터들, 및 제로 또는 그 이상의 태스크 라인들 외에 어떠한 입력들도 갖지 않는 마이크로컨트롤러.

청구항 16

제 1 항 내지 제 15 항 중 어느 한 항에 있어서,
이벤트를 신호할 때 상기 프로세서로 인터럽트를 발송하기 위한 회로부를 포함하는 마이크로컨트롤러.

청구항 17

제 1 항 내지 제 16 항 중 어느 한 항에 있어서,
복수의 이벤트-생성 유닛들을 포함하고, 각각은 일련의 인터럽트 라인들에 의해 상기 프로세서에 연결되고, 인터럽트 라인들의 각각의 세트는 상이한 개별 인터럽트 우선(priority level)를 갖는 마이크로컨트롤러.

청구항 18

마이크로컨트롤러에 있어서,
프로세서;
복수의 주변기기들;
프로그램 가능한 주변기기 상호연결부; 및
인터럽트-생성 유닛을 포함하되,
상기 주변기기들 중 이벤트-생성 주변기기는 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록 구성되고;
상기 인터럽트-생성 유닛은 상기 프로그램 가능한 주변기기 상호연결부로부터 태스크 신호를 수신한 것에 응답하여 상기 프로세서에 인터럽트를 발송하도록 배열되고;
상기 프로그램 가능한 주변기기 상호연결부는 (i) 제 1 주변기기의 이벤트와 (ii) 상기 인터럽트-생성 유닛의 또는 제 2 주변기기의 태스크 사이의 맵핑이 저장될 수 있는 메모리를 액세스를 하도록 구성되고;
제 1 주변 기기의 이벤트와 상기 인터럽트-생성 유닛의 태스크 사이의 맵핑이 상기 메모리에 저장될 때, 상기 프로그램 가능한 주변기기 상호연결부가 상기 인터럽트-생성 유닛에 상기 태스크를 위하여 태스크 신호를 발송함으로써 상기 제 1 주변기기로부터의 이벤트의 신호에 응답하도록 상기 프로그램 가능한 주변기기 상호연결부가 구성되는

마이크로컨트롤러.

청구항 19

제 18 항에 있어서,

상기 인터럽트-생성 유닛은 또한 상기 이벤트-생성 레지스터의 컨텐츠들에 대한 미리 결정된 변화를 감지하고 이런 미리 결정된 변화를 감지한 것에 응답하여, 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록 배열된, 이벤트-생성 유닛인

마이크로컨트롤러.

청구항 20

제 18 항 또는 제 19 항에 있어서,

상기 인터럽트-생성 유닛은

상기 태스크 신호를 수신하는 것으로부터 미리 결정된 최대값내에 상기 인터럽트를 발송하도록 배열된
마이크로컨트롤러.

청구항 21

제 18 항 내지 제 20 항 중 어느 한 항에 있어서,

상기 인터럽트-생성 유닛은 상기 태스크 신호를 수신한 후에 일정한 시간 지연에서 상기 인터럽트를 발송하도록 배열된

마이크로컨트롤러.

청구항 22

제 18 항 내지 제 21 항 중 어느 한 항에 있어서,

복수의 인터럽트-생성 유닛들을 포함하고, 각각은 일련의 인터럽트 라인들에 의해 상기 프로세서에 연결되고, 인터럽트 라인들의 각각의 세트는 상이한 개별 인터럽트 우선 순위를 갖는

마이크로컨트롤러.

청구항 23

제 18 항 내지 제 22 항 중 어느 한 항에 있어서,

소프트웨어를 저장하는 메모리를 포함하고, 상기 소프트웨어는 상기 맵핑 메모리에, 상기 이벤트-생성 주변기기의 이벤트로부터 상기 인터럽트-생성 유닛의 태스크로의 맵핑을 저장하도록 상기 프로세서에 의해 실행 가능한 지시들을 포함하는

마이크로컨트롤러.

발명의 설명

기술 분야

본 발명은 마이크로컨트롤러에 관한 것이다.

배경기술

[0002] 마이크로컨트롤러들은 타이밍이 중요한 애플리케이션들에, 예컨대 라디오 디바이스들에 종종 사용된다. 하나 이상의 주변기기들을 수반하는 두개의 태스크(task)들이 필연적으로 동시에 마이크로컨트롤러 상의 프로세서에 의해 개시되는 것이 중요할 수 있는 상황들이 있다. 디바이스는 예를 들어, 시리얼 인터페이스(serial interface)를 통하여 통신을 개시하고 정확하게 동일한 시간에 타이머를 시작할 필요가 있을 수 있다. 이것은 보통은 태스크들 중 제 1 태스크를 개시하도록 프로세서에 지시하고 그런 다음 즉각적으로 다른 태스크를 개시하도록 프로세서에 지시함으로써 수행된다. 그러나, 설사 이들이 근접한 소프트웨어 지시들을 이용하여 수행한다 할지라도, 프로세서는 제 1 태스크를 트리거링 한 후에, 제 2 태스크를 트리거링 하기 전에 인터럽트(interrupt)를 받을 수 있다는 것이 여전히 가능하다. 이것은 바람직하지 않은 행동으로 이어질 수 있다.

[0003] 이 문제는 중요한 시간들에 어떤 인터럽트들을 감춤(mask)함으로써, 또는 상이한 인터럽트들 상에 상이한 우선순위(priority)를 설정함으로써 어느 정도까지 완화될 수 있다. 그러나, 이들 접근법들 중 어느 것도 완벽하지 않고, 그것들은 또한 소프트웨어에 추가의 복잡도를 초래한다.

발명의 내용

과제의 해결 수단

[0004] 본 발명은 대안 접근법을 제공한다.

[0005] 제 1 측면에서, 본 발명은 마이크로컨트롤러를 제공하고, 상기 마이크로컨트롤러는:

[0006] 프로세서;

[0007] 복수의 주변기기들;

[0008] 프로그램 가능한 주변기기 상호연결부(programmable peripheral interconnect);

[0009] 이벤트 생성 유닛; 및

[0010] 소프트웨어를 저장하는 메모리를 포함하되,

[0011] 상기 이벤트-생성 유닛은 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는(addressable) 이벤트-생성 레지스터를 포함하고;

[0012] 상기 이벤트-생성 유닛은 상기 프로그램 가능한 주변기기 상호연결부에 연결되고;

[0013] 상기 이벤트-생성 유닛은 상기 이벤트-생성 레지스터의 상기 컨텐츠들에 대한 미리 결정된 변화를 감지하고 이런 미리 결정된 변화를 감지한 것에 응답하여, 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록(signal) 배열되고;

[0014] 상기 주변기기들의 각각은 상기 프로그램 가능한 주변기기 상호연결부에 연결되고;

[0015] 상기 주변기기들의 각각은 개별 태스크를 수행함으로써 상기 프로그램 가능한 주변기기 상호연결부로부터의 태스크 신호에 응답하도록 구성되고;

[0016] 상기 프로그램 가능한 주변기기 상호연결부는 복수의 맵핑들이 저장될 수 있는 맵핑 메모리를 액세스하도록 구성되고, 각각은 상기 이벤트-생성 유닛의 이벤트를 상기 주변기기들 중 하나의 태스크에 맵핑시키고;

[0017] 상기 맵핑 메모리에 상기 이벤트-생성 유닛으로부터의 이벤트로부터 상기 주변기기들 중 하나의 태스크로의 맵핑이 저장될 때, 상기 프로그램 가능한 주변기기 상호연결부가 태스크 신호를 상기 주변기기로 발송함으로써 상기 이벤트-생성 유닛으로부터의 상기 이벤트의 신호에 응답하도록 상기 프로그램 가능한 주변기기 상호연결부는 구성되고;

[0018] 상기 맵핑 메모리가 상기 이벤트-생성 유닛의 하나의 이벤트로부터 두개 이상의 상이한 태스크들로 맵핑들을 저장할 때, 상기 프로그램 가능한 주변기기 상호연결부가 상기 이벤트의 신호를 수신한 것으로부터 미리 결정된 최대 시간내에 상기 두개 이상의 개별 태스크 신호들을 발송하도록 상기 프로그램 가능한 주변기기 상호연결부는 구성되고; 및

[0019] 상기 소프트웨어는 (i) 상기 맵핑 메모리에, 상기 이벤트-생성 유닛의 하나의 이벤트로부터 적어도 두개의 상이

한 주변기기 태스크들로의 맵핑들을 저장하고, 및 (ii) 상기 이벤트-생성 레지스터의 컨텐츠들에 상기 미리 결정된 변화를 수행하도록 상기 프로세서에 의해 실행 가능한 지시들을 포함한다.

[0020] 따라서 본 발명에 따른, 상기 소프트웨어가 실행될 때, 두개 이상의 주변기기 태스크들이 극미적으로 (atomically), 그리고 미리 결정된 최대 시간 프레임내에 트리거되도록 하는 것이 당해 기술분야의 통상의 기술자들에 의해 인식될 것이다. 단지 단일 프로세서만이 기록하기 때문에 지시는 상기 주변기기 태스크들을 개시하기 위해 사용되고, 상기 두개의 태스크들의 트리거링이 프로세서 인터럽트에 의해 중단되어지는 위험은 없다.

[0021] 상기 미리 결정된 최대 시간은 임의의 값을 가질 수 있지만, 그러나 바람직하게는 10 마이크로초 또는 그 미만; 보다 바람직하게는 1 마이크로초 또는 그 미만; 및 가장 바람직하게는 0.1 마이크로초 또는 그 미만이다. 그것은 바람직하게는 충분히 작아서 두개 이상의 태스크들은 실제적인 용도를 위해, 동시에 트리거되는 것으로 간주될 수 있다. 상기 미리 결정된 최대 시간은 100 프로세서 클럭 사이클들보다 작을 수 있고; 보다 바람직하게는, 그것은 10 또는 5 클럭 사이클들보다 작을 수 있다. 일부 실시예들에서, 그것은 하나의 프로세서 클럭 사이클보다 작을 수 있다. 상기 미리 결정된 최대 시간은 바람직하게는 모든 맵핑들에 대하여 일정하다. 바람직하게는, 상기 프로그램 가능한 주변기기 상호연결부는 상기 두개 이상의 개별 태스크 신호들을 동시에 발송하도록 배열된다(예를 들어, 상기 프로그램 가능한 주변기기 상호연결부에 제공되거나 또는 상기 프로세서에 제공되는 클럭 신호의 한 사이클내에). 이 경우에, 바람직하게는 모든 가능한 맵핑들에 대하여 이벤트의 신호를 수신하고 상기 이벤트에 맵핑된 두개 이상의 태스크 신호들을 발송하는 상기 프로그램 가능한 주변기기 인터페이스간에 일정한 시간 지연이 있다.

[0022] 프로그램 가능한 주변기기 상호연결부들은 - 예를 들어, 본 출원인에 의한 WO 2013/088121에 설명된 대로 이미 알려져 있다. 일부 선호되는 실시예들에서, 상기 프로그램 가능한 주변기기 상호연결부들은 WO 2013/088121에 설명된 프로그램 가능한 주변기기 상호연결부와 실질적으로 동일하고, 비록 이것이 필수가 아닐지라도 이의 전체 내용들은 참조로서 본 출원에 통합된다.

[0023] 상기 프로그램 가능한 주변기기 상호연결부는 바람직하게는 주변기기-대-주변기기 채널들의 프로그래밍을 또한 지원한다. 일부 실시예들에서, 상기 마이크로컨트롤러는 또한 이벤트를 생성하고, 상기 프로그램 가능한 주변기기 상호연결부(PPI)에 이벤트를 신호하도록 구성된 하나 이상의 주변기기들을 포함한다. 상기 바람직하게는 제 1 주변기기의 이벤트와 제 2 주변기기의 태스크간의 맵핑이 저장될 수 있도록, 제 1 주변기기의 이벤트와 제 2 주변기기의 태스크간의 맵핑이 메모리에 저장될 때, 상기 상호연결부가 상기 제 1 주변기기로부터 상기 이벤트의 신호에 응답하여 상기 제 2 주변기기로 태스크 신호를 발송함으로써 채널을 제공하도록 상기 프로그램 가능한 주변기기 상호연결부가 배열된다. 이런 식으로, 상기 프로그램 가능한 주변기기 상호연결부는 그것이 이벤트-생성 주변기기와 인터페이스하는 동일한 방식으로 상기 이벤트-생성 유닛과 안성맞춤으로 인터페이스할 수 있다. 일부 실시예들에서, 상기 이벤트-생성 유닛은 이벤트-생성 주변기기의 일 유형으로 간주될 수 있다.

[0024] 상기 주변기기들은 임의의 특정 유형에 제한되지 않고, 타이머, UART, 전압 비교기, 암호화 엔진, 아날로그-디지털 컨버터(ADC), 디지털-아날로그 컨버터(DAC), 라디오 송신기, 라디오 수신기, 등등과 같은 주변기기들을 포함할 수 있다. 이벤트는 임의의 입력, 상태의 변화, 기준을 충족시키는 것, 등., 당해 기술의 통상의 기술자에 익숙한 것-예를 들어, 타이머가 타겟 값에 도달한 때에 응답하여 주변기기로부터 신호될 수 있다. 상기 태스크들은 인터페이스를 통하여 데이터를 송신하는 것과 같은 주변기기에 의해 수행될 수 있는 임의의 기능 또는 동작일 수 있다.

[0025] 상기 이벤트-생성 유닛은 단지 단일 유형의 이벤트를 신호하도록 배열될 수 있지만, 그러나 그것은 바람직하게는 복수의 상이한 이벤트들을 신호할 수 있다. 이들은 상이한 개별 이벤트 라인들을 통하여, 또는 상이한 개별 이벤트 레지스터들을 이용하여 신호될 수 있다. 상기 이벤트-생성 유닛은 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 복수의 이벤트-생성 레지스터들을 가질 수 있다. 각각의 이벤트-생성 레지스터는 상이한 개별 이벤트와 관련될 수 있다. 상기 상이한 이벤트들은 바람직하게는 저장된 맵핑에서 - 예를 들어, 식별자로 상기 이벤트-생성 레지스터의 어드레스를 이용함으로써, 또는 일부 다른 미리 결정된 셋의 식별자들을 이용함으로써 고유하게 식별될 수 있다.

[0026] 이벤트들이 상기 주변기기에 상태의 일부 변화를(예를 들어, 일부 동작의 완료) 의미하는 통상의 주변기기와 달리, 상기 이벤트-생성 유닛으로부터 이벤트는 고유한 의미를 가지지 않고, 대신 상기 이벤트-생성 레지스터에 기록함으로써 그것을 야기하는 상기 프로세서(또는 다른 컴포넌트)에 의해 의미가 주어진다고 인식될 것이다.

[0027] 상기 PPI는 각각의 이벤트 및 태스크에 대하여 이벤트 및 태스크 신호들이 발송될 수 있는 개별 라인들; 즉 각

각의 이벤트 또는 태스크에 대하여 하나의 라인에 의해 상기 주변기기들 및/또는 상기 이벤트-생성 유닛에 연결될 수 있다. 상기 신호들은 펄스들 또는 인코딩된 값들일 수 있다. 착신 라인들은 상기 PPI내의 하나 이상의 다중화기들에; 예를 들어 각각의 채널에 대하여 하나의 다중화기와 연결될 수 있다. 아웃고잉(outgoing) 라인들은 상기 PPI내의 하나 이상의 역다중화기들로부터; 예를 들어 각각의 채널에 대하여 하나의 역다중화기로부터 떠날 수 있다. 상기 PPI는 메모리에 저장된 맵핑에 따라 채널의 다중화기 및 역다중화기를 설정 또는 제어하도록 구성될 수 있다.

[0028] 일부 실시예들에서, 상기 PPI는 버스를 통하여, 예를 들어, 메모리-맵핑 입력/출력(MMIO)을 이용하여 어드레스 버스를 통하여 주변기기들 및/또는 상기 이벤트-생성 유닛의 상기 태스크 레지스터들 또는 이벤트 레지스터들, 또는 둘모두를 액세스할 수 있다. 상기 PPI는 따라서 상기 관련 이벤트 레지스터의 컨텐츠들에 변화를 통하여 이벤트를 신호를 수신할 수 있고 및/또는 상기 관련 태스크 레지스터의 컨테츠들을 변화시킴으로써 태스크 신호를 발송할 수 있다. 상기 PPI는 상기 프로세서가 또한 연결되는 버스(예를 들어, 어드레스 버스, 데이터 버스 및 제어 버스를 포함할 수 있는 시스템 버스)에 연결될 수 있다. 상기 동일한 버스는 상기 주변기기들 및/또는 상기 이벤트-생성 유닛을 상기 프로세서에 연결할 수 있다. 상기 PPI는 간격들로; 예를 들어, 규칙적인 폴링(polling)로 상기 이벤트 레지스터를 판독함으로써; 또는 상기 이벤트-생성 주변기기 또는 상기 이벤트-생성 유닛으로부터 인터럽트를 수신함으로써 이벤트 레지스터의 컨텐츠들에 변화를 감지할 수 있다. 인터럽트 수신시에, 상기 PPI는 그런 다음 대응하는 이벤트 레지스터의 컨텐츠들을 판독할 수 있다.

[0029] 이벤트 레지스터 또는 태스크 레지스터는 단일 비트를 함유할 수 있거나(즉, 플래그 또는 바이너리 신호를 보내기 위해), 또는 그것은 복수의 비트들, 예를 들어, 8, 16 또는 32 비트들을 포함할 수 있다.

[0030] 상기 이벤트-생성 유닛은 상기 이벤트 신호 라인을 통하여 상기 프로그램 가능한 주변기기 상호연결부(PPI)에 이벤트를 신호할 수 있다. 대안적으로, 그것은 레지스터에 대한 변화를 통하여 상기 이벤트를 신호할 수 있고, 이 경우에 상기 PPI는 상기 변화를 감지하기 위해 간격들에서 상기 레지스터를 폴링하도록 구성될 수 있다. 이 레지스터는 상기 이벤트-생성 레지스터 그 자체일 수 있거나, 또는 상기 이벤트-생성 유닛을 위한 별개의 이벤트 레지스터일 수 있다.

[0031] 일부 실시예들에서, 맵핑내 태스크의 식별은 WO 2013/088121에 설명된 상기 태스크와 관련된 태스크 레지스터의 어드레스이다. 그러나, 이는 필수가 아니고, 태스크-수신 주변기기의 태스크의 식별은 예를 들어, 임의의 관련 태스크 레지스터의 (상기 주변기기상에 존재하거나 또는 존재하지 않을 수 있는) 상기 어드레스에 관련되지 않은 (i) 주변기기-식별자 및 (ii) 태스크-식별자 또는 신호-라인-식별자를 포함할 수 있다.

[0032] 태스크-수신 주변기기는 태스크와 관련된 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 태스크 레지스터를 포함할 수 있고, 상기 태스크 레지스터의 컨텐츠들의 변화에 응답하여 해당 태스크를 수행하도록 구성될 수 있다. 그런 다음 제 1 주변기기의 이벤트와 제 2 주변기기의 태스크 사이의 맵핑은 (i) 상기 제 1 주변기기의 상기 이벤트의 식별, 및 (ii) 상기 태스크와 관련된 태스크 레지스터의 어드레스를 포함할 수 있다. 제 1 주변기기의 이벤트와 제 2 주변기기의 태스크간의 맵핑이 메모리에 저장될 때, 상기 상호연결부가 상기 제 1 주변기기로부터 상기 이벤트의 신호에 응답하여 상기 제 2 주변기기로 태스크 신호를 발송함으로써 채널을 제공하도록 PPI가 구성될 수 있다.

[0033] 이벤트-생성 주변기기는 이벤트와 관련된 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 이벤트 레지스터를 포함할 수 있다. 상기 이벤트 레지스터는 집적 회로상에 상기 주변 기기와 관련된 로직에 바로 인접하여 위치될 수 있지만, 이는 필수가 아니고 그것은 상기 주변기기의 다른 엘리먼트로부터 일부 거리만큼 떨어져 위치될 수 있고; 상기 태스크 레지스터에 대하여 동일하게 적용된다. 상기 레지스터들은 임의의 적절한 메모리 구조를 포함할 수 있다. 상기 주변기기는 이벤트를 신호할 때 상기 이벤트 레지스터의 컨텐츠들을; 예를 들어, 그것이 이벤트를 신호할 때마다 바이너리 '1'을 상기 이벤트 레지스터에 기록함으로써 변화시킬 수 있다.

[0034] 일부 실시예들에서, 맵핑내 제 1 주변기기의 이벤트의 식별은 상기 이벤트와 관련된 이벤트 레지스터의 어드레스이다. 이런 식으로, 채널은 메모리내 두개의 레지스터 어드레스들: 이벤트-레지스터 어드레스 및 태스크-레지스터 어드레스를 저장함으로써 간단하게 구성될 수 있다. 상기 프로세서상에서 운영하는 소프트웨어는 기능 호출을 통하여 이것을 수행할 수 있거나 또는 상기 맵핑이 저장되는 상기 메모리에 직접 상기 어드레스들을 기록할 수 있다. 그러나, 이는 필수가 아니고, 상기 제 1 주변기기의 이벤트의 식별은 존재하거나 또는 존재하지 않을 수 있는 임의의 관련 태스크 레지스터의 어드레스에 관련되지 않은 주변기기 식별자 및 이벤트 또는 신호 라인 식별자를 포함할 수 있다.

[0035] 상기 이벤트-생성 유닛은 바람직하게는 상기 이벤트-생성 유닛의 이벤트와 관련된 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 적어도 하나의 이벤트 레지스터를 포함한다. 일부 실시예들에서 상기 이벤트-생성 레지스터는 상기 이벤트 레지스터이다. 다른 실시예들에서, 상기 이벤트 레지스터는 상기 이벤트-생성 레지스터와 분리된다(즉, 상이한 어드레스를 갖는다). 상기 유닛은 상기 이벤트-생성 레지스터에 대한 미리 결정된 변화가 상기 이벤트 레지스터에 미리 결정된 변화를 유발할 수 있도록 구성될 수 있다. 상기 이벤트-생성 레지스터는 일부 실시예들에서, 상기 이벤트-생성 유닛에 대하여 태스크 레지스터일 수 있다. 상기 이벤트-생성 유닛은 복수의 이벤트-생성 레지스터들 및 복수의 이벤트 레지스터들을 포함할 수 있다. 각각의 이벤트-생성 레지스터는 상기 이벤트 레지스터들의 개개의 이벤트 레지스터와 관련될 수 있어서, 상기 이벤트-생성 레지스터들 중 하나에 대한 변화가 상기 이벤트 레지스터들 중 하나와 관련된 변화를 유발한다.

[0036] 상기 이벤트-생성 유닛은 - 예를 들어, 특정한 이벤트를 신호하기 전에 어쩌면 미리 결정된 순서로 두개의 상이한 이벤트-생성 레지스터들에 미리 결정된 변화들을 요구하여 하나의 이벤트로 두개 이상의 이벤트-생성 레지스터들을 연관시키는 로직을 포함할 수 있다. 다시 말해서, 상기 이벤트-생성 유닛은 두개 이상의 이벤트-생성 레지스터들에 관련한 기준이 충족된 때 또는 그 때에만 이벤트 신호를 발송하기 위한 로직을 포함할 수 있다. 이런 로직은 고정될 수 있거나(즉, 하드웨어에 내장된), 또는 그것은 프로그램 가능할 수 있다.

[0037] 일부 실시예들에서, 상기 이벤트-생성 유닛은 상기 프로그램 가능한 주변기기 상호연결부로부터 (예를 들어, 태스크 라인을 통하여, 또는 상기 이벤트-생성 레지스터의 컨텐츠들에 대한 변화의 방식으로) 태스크 신호를 수신할 수 있고, 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호함에 의해 응답할 수 있다. 이것은 만약 상기 이벤트-생성 유닛이 또한 이하에 보다 상세하게 설명되는 것처럼 이벤트를 신호할 때 상기 프로세서에 인터럽트들을 발송하도록 구성되면 유용할 수 있다.

[0038] 상기 이벤트-생성 유닛의 이벤트는 이벤트-생성 주변기기의 이벤트가 식별되는 것과 동일한 방식으로-예를 들어, 관련 이벤트 레지스터의 어드레스에 의해, 또는 상기 이벤트에 고유한 일부 다른 식별자에 의해 맵핑내에서 식별될 수 있다.

[0039] 상기 PPI는 그것이 하나 이상의 맵핑들을 이용하여 주변기기들간에 연결들을 정의하도록 프로그래밍될 수 있거나 또는 구성될 수 있다는 면에서 프로그램 가능하다. 상기 PPI는 그것 자체의 프로세싱 유닛을 포함할 수 있지만 소프트웨어 지시들을 실행시키기 위한 그것 자체의 프로세싱 유닛을 반드시 포함할 필요는 없다.

[0040] 맵핑 메모리내 맵핑은 임의의 적절한 형태를 취할 수 있다. 일부 실시예들에서, 이벤트 레지스터 및 태스크 레지스터의 어드레스, 또는 이를 레지스터들에 대한 포인터(pointer)들은 어레이, 테이블 또는 데이터베이스에 관련 엔트리들로서 저장될 수 있다. 예를 들어, 상기 PPI는 많은 로우들, 각각이 상이한 채널에 대응하고, 두개의 컬럼들, 첫번째는 이벤트 레지스터를 식별하기 위한 것이고 두번째는 태스크 레지스터를 식별하기 위한 것을 갖는 테이블을 유지할 수 있다. 이런 어레이 또는 테이블은 로직 구조물일 수 있고 메모리내 데이터의 임의의 특정 물리적 위치들에 반드시 제한될 필요는 없다는 것이 인식될 것이다. 상기 PPI는 태스크-레지스터 어드레스들을 저장하기 위한 제 1 셋의 레지스터들 및 이벤트-레지스터 어드레스들을 저장하기 위한 제 2 셋의 레지스터들을 포함할 수 있다. 각각의 세트내에 같은 수의 레지스터들이 있을 수 있다. 상기 제 1 세트내 레지스터는 상기 제 2 세트내 대응하는 레지스터에 의해 제어되는 역다중화기에 연결된 다중화기를 제어할 수 있고, 그렇게 함으로써 채널을 정의한다.

[0041] 상기 메모리는 예를 들어, 실리콘의 별개의 영역에 또는 상이한 칩상에 상기 PPI로부터 분리될 수 있지만, 그러나 바람직하게는 상기 PPI의 통합 컴포넌트이고, 이는 액세스 시간들을 줄일 수 있다. 상기 메모리는 휘발성(예를 들어, RAM) 또는 비-휘발성(예를 들어, EEPROM 또는 플래시)일 수 있다. 상기 맵핑들은 바람직하게는 상기 프로세서에 의해 자체 어드레스로 호출할 수 있는 하나 이상의 레지스터들에 저장된다. 상기 PPI에 의해 제공된 각각의 채널은 하나의 관련 이벤트 엔드-포인트 및 하나의 관련 태스크 엔드-포인트를 가질 수 있고, 이는 이벤트 레지스터의 어드레스 및 태스크 레지스터의 어드레스를 개별적으로 저장하는데 적절할 수 있다. 상기 채널들의 일부는 다른 유형들의 이벤트 및/또는 태스크 식별자들을 저장하는데 이를 레지스터들을 사용할 수 있다.

[0042] 상기 맵핑 또는 맵핑들은 상기 PPI에 의해 또는 상기 프로세서에 의해, 또는 둘 모두에 의해 상기 메모리에 기록될 수 있다. 상기 프로세서는 상기 적절한 맵핑을 기록하도록 상기 PPI에 지시함으로써 주변기기들간에 채널을 간접적으로 수립할 수 있다.

[0043] 상기 PPI는 바람직하게는 해당 레지스터 어드레스에 대응하는 특정 이벤트 라인 또는 태스크 라인(또는 이벤트 포트 또는 태스크 포트)을 결정하기 위해 맵핑으로부터 레지스터 어드레스를 룩 업하는데 사용할 수 있는 하나

이상의 루업 테이블들을 포함하거나 또는 액세스할 수 있다. 그런 다음 그것은 채널에 대한 입력 또는 출력으로서 상기 라인 또는 포트를 선택할 수 있다(예를 들어, 다중화기 또는 역다중화기를 적절하게 제어함으로써). 상기 루업 테이블은 임의의 적절한 형태를 취할 수 있고, 반드시 메모리내 물리적 테이블로서 구현될 필요는 없다.

[0044] 상기 PPI는 임의 개수의 채널들; 예를 들어, 1, 8, 16, 32 또는 그 이상을 지원할 수 있다. 채널은 상기 PPI로의 이벤트 입력 포트 또는 라인과 상기 PPI로부터의 태스크 출력 포트 또는 라인 사이에 임의의 물리적(예를 들어, 전기적) 또는 로직연결일 수 있다. 채널은 링크 하나 이상의 입력들을 하나 이상의 출력들에 링크할 수 있다(예를 들어, 상기 채널 경로를 따라서 분기(fork)들 또는 브랜치(branch)들을 정하는 로직 게이트들을 이용하여).

[0045] 상기 PPI는 바람직하게는 채널이 인에이블(enable)되고 및/또는 디스에이블(disable)되는 것을 허용하기 위한 메커니즘을 포함한다. 상기 PPI는 각각의 채널과 관련된 하나의 비트를 갖는 비트-필드 레지스터(bit-field register)를 포함할 수 있고, 만약 상기 비트-필드 레지스터내 그것의 관련 비트가 미리 결정된 값(예를 들어, 바이너리 1)로 설정되면 채널이 인에이블되도록 구성될 수 있다.

[0046] 이벤트-생성 주변기기는 하나 초과의 이벤트 레지스터를 가질 수 있다. 유사하게, 태스크-수신 주변기기는 하나 초과의 태스크 레지스터를 가질 수 있다.

[0047] 이미 설명된 대로, 상기 PPI는 잠재적으로 상이한 태스크-수신 주변기기들에 두개 이상의 태스크 신호들을 발송함으로써 하나의 이벤트 신호에 응답하도록 구성 가능하다. 일부 실시예들에서 이것은 상기 맵핑 메모리에 복수의 맵핑 관계들을 저장함으로써 성취될 수 있고, 각각은 상기 이벤트(예를 들어, 이벤트 레지스터 어드레스에 의해 식별된)를 상이한 개별 태스크(예를 들어, 개별 태스크 레지스터 어드레스들에 의해 식별된)에 맵핑시킨다. 다른 실시예들에서, 그것은 하나의 이벤트를 두개 이상의 태스크들에 직접 맵핑시키는 단일 맵핑 관계를 저장함으로써; 예를 들어, 세개 또는 그 이상 컬럼들, 상기 이벤트에 대하여 하나, 및 개별 태스크들에 대하여 두개 이상의 추가 컬럼들을 포함하여 상기 맵핑 메모리에 저장된 테이블을 가짐으로써 성취될 수 있다.

[0048] 상기 이벤트-생성 유닛은 바람직하게는 상기 PPI로 하나 이상의 이벤트 라인들, 상기 프로세서로 제로 또는 그 이상의 인터럽트 라인들, 및 제로 또는 그 이상의 레지스터들외에 어떠한 출력들도 갖지 않는다. 그것은 바람직하게는 클럭 입력, 하나 이상의 레지스터들, 및 제로 또는 그 이상의 태스크 라인들외에 어떠한 입력들도 갖지 않는다. 상기 이벤트-생성 유닛은 전형적으로 통상의 주변기기보다 훨씬 더 간단하다. 그것은 바람직하게는 그것의 컴포넌트들의 수 또는 그것의 게이트들의 수의 면에서 상기 마이크로컨트롤러상에 임의의 다른 주변기기보다 적다. 그것은 바람직하게는 타이밍 메커니즘을 갖지 않는다(타이머 주변기기와 달리). 그것은 바람직하게는 제로 또는 그 이상 이벤트 레지스터들, 태스크 레지스터들 및 구성 레지스터들외에 어떠한 데이터 입력 메커니즘도 갖지 않는다(예를 들어, 임의의 데이터를 다른 컴포넌트들 또는 디바이스들로 송신 및 수신하기 위한 연결들을 갖는 시리얼 인터페이스, 또는 디지털-아날로그 컨버터와 달리).

[0049] 상기 이벤트-생성 유닛은 이벤트를 신호할 때 상기 프로세서로 인터럽트를 발송하기 위한 회로부를 포함할 수 있다. 상기 인터럽트는 감춰질 수 있는(maskable) 인터럽트 또는 감춰질 수 없는(non-maskable) 인터럽트일 수 있다. 그것은 특정한 이벤트를 신호할 때 상기 프로세서로 인터럽트의 발송을 인에이블 및/또는 디스에이블하기 위한 입력을 가질 수 있다. 상기 이벤트-생성 유닛은 그것이 상기 이벤트-생성 레지스터의 컨텐츠들에 대한 변화를 감지할 때 상기 프로세서로 인터럽트를 발송하도록 상기 이벤트-생성 유닛을 구성하기 위한 하나 이상의 구성 레지스터들을 포함할 수 있다. 상기 유닛은 감춰질 수 있는 및 감춰질 수 없는 인터럽트들을 개별적으로 구성하기 위해 별개의 구성 레지스터들을 포함할 수 있다.

[0050] 상기 프로세서로 인터럽트를 발송하는 능력은 상기 이벤트-생성 유닛이 상기 프로그램 가능한 주변기기 인터페이스로부터 태스크 신호를 수신하기 위한 입력을 포함할 때 특별히 유용하다. 이것은 그런 다음 주변기기상의 이벤트가 상기 이벤트-생성 유닛으로 하여금 상기 프로세서로 인터럽트를 발송하게 할 수 있도록 상기 프로그램 가능한 주변기기 인터페이스가 프로그래밍되는 것을 허용한다. 만약 상기 주변기기가 인터럽트 그 자체를 발송하는 것이 가능하지 않거나, 또는 만약 상기 주변기기로부터의 모든 인터럽트들은 제한된 수의 인터럽트 서비스 루틴들(예를 들어, 하나의 인터럽트 서비스 루틴)로 프로세스되어야한다는 것을 의미하는 상기 주변기기가 단지 제한된 수의 인터럽트들(예를 들어, 하나의 인터럽트)이 할당되었다면 이것은 유용할 수 있다. 대신에 인터럽트를 상기 프로세서로 발송하기 위해서 상기 PPI 및 상기 이벤트-생성 유닛을 이용함으로써, 상기 주변기기는 그것이 상기 프로세서로 발송되도록 야기할 수 있는 상이한 인터럽트들의 수를 (예를 들어, 상이한 인터럽트 수들로) 효율적으로 증가시킬 수 있다. 이것은 상기 인터럽트 서비스 루틴의 프로그래밍을 단순화할 수 있다. 또한,

프로세서들이 상이한 우선 순위들에서 인터럽트들을 지원하는, 주변기기가 복수의 상이한 인터럽트 레벨들에서 상기 프로세서를 인터럽트하는 것을 가능하게 할 수 있고, 이는 상기 이벤트-생성 유닛을 이용하지 않고서 상기 주변기기에 액세스 가능한 인터럽트 레벨들의 수보다 더 클 수 있다.

[0051] 예를 들어, 주변기기는 매 100us에 일어나는 하나의 이벤트 소위 "NEWVALUE", 무언가 잘못 진행될 때에만 그것이 신호하는 다른 이벤트 소위 "ERROR"를 가질 수 있다. 상기 주변기기는 그러나, 상기 프로세서로 하나의 인터럽트 라인만을 가질 수 있고, 따라서 상기 "ERROR" 이벤트에 대하여 더 높은 우선순위에서 인터럽트하는 것을 할 수 없다. 그러나, 상기 주변기기로부터의 상기 "ERROR" 이벤트의 신호가 상기 PPI로 하여금 상기 이벤트-생성 유닛상에 태스크 라인을 활성시키도록 상기 주변기기와 상기 이벤트-생성 유닛간에 PPI채널을 구성함으로써, 그리고 그것이 해당 태스크 라인을 통하여 신호를 수신할 때 상기 프로세서로 높은-우선순위 인터럽트를 발송하도록 상기 이벤트-생성 유닛을 설정함으로써, 상기 주변기기는 높은 우선순위에서 상기 프로세서로 "ERROR" 인터럽트를 발송할 용량(capacity)이 제공될 것이다. 이것은 상기 "ERROR" 이벤트에 상기 프로세서가 빠르게 응답하는 것을 가능하게 할 수 있다.

[0052] 일부 실시예들에서, 상기 마이크로컨트롤러는 복수의 이벤트-생성 유닛들을 포함하고, 각각은 일련의 인터럽트 라인들에 의해 상기 프로세서에 연결되고, 각각의 세트는 상이한 개별 인터럽트 우선 순위(priority level)를 갖는다. 대안적으로, 하나의 이벤트-생성 유닛은 제 1 우선 순위에서 인터럽트를 발송하기 위해 사용될 수 있는 하나 이상의 태스크들(또는 이벤트들), 및 제 2 우선 순위에서 인터럽트를 발송하기 위해 사용될 수 있는 하나 이상의 다른 태스크들(또는 이벤트들)을 가질 수 있다. 상기 태스크들(또는 이벤트들)과 상기 인터럽트 우선 순위들 사이의 연관(association)은 고정될 수 있거나 또는 구성 가능할 수 있다(예를 들어, 상기 이벤트-생성 유닛상의 하나 이상의 레지스터 값들에 따라).

[0053] 프로세서 인터럽트들을 생성하기 위해서 상기 프로그램 가능한 주변기기 상호연결부에 연결된 유닛을 이용하는 이 아이디어는 그 자체의 권리로 새롭고 창의적이어서, 추가 측면으로부터, 상기 발명은 마이크로컨트롤러를 제공하고 상기 마이크로컨트롤러는:

프로세서;

복수의 주변기기들;

프로그램 가능한 주변기기 상호연결부; 및

인터럽트-생성 유닛을 포함하고,

상기 주변기기들 중 이벤트-생성 주변기기는 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록 구성되고;

상기 인터럽트-생성 유닛은 상기 프로그램 가능한 주변기기 상호연결부로부터 태스크 신호를 수신하는 것에 응답하여 상기 프로세서로 인터럽트를 발송하도록 배열되고;

상기 프로그램 가능한 주변기기 상호연결부는 (i) 제 1 주변기기의 이벤트와 (ii) 상기 인터럽트-생성 유닛의 또는 제 2 주변기기의 태스크 사이의 맵핑이 저장될 수 있는 메모리를 액세스하도록 구성되고;

제 1 주변기기의 이벤트와 상기 인터럽트-생성 유닛의 태스크 사이의 맵핑이 메모리에 저장될 때, 상기 프로그램 가능한 주변기기 상호연결부가 상기 인터럽트-생성 유닛에 상기 태스크를 위하여 태스크 신호를 발송함으로써 상기 제 1 주변기기로부터의 이벤트의 신호에 응답하도록 상기 프로그램 가능한 주변기기 상호연결부가 구성된다.

임의의 상기 선행하는 측면 및 실시예들의 임의의 특징은 또한 이 측면의 특징일 수 있다. 특별히, 상기 인터럽트-생성 유닛은 이벤트-생성 유닛일 수 있다. 따라서 그것은 상기 이벤트-생성 레지스터의 상기 컨텐츠들에 대한 미리 결정된 변화를 감지하고 이런 미리 결정된 변화를 감지한 것에 응답하여, 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록 배열될 수 있다. 그것은 앞에서 설명된 이벤트-생성 유닛일 수 있다. 상기 인터럽트-생성 유닛은 본 출원에서 설명된 상기 이벤트-생성 유닛들의 임의의 특징들을 가질 수 있다. 상기 프로그램 가능한 주변기기 상호연결부(PPI)는 앞에서 설명된 마이크로컨트롤러 PPI들의 임의의 마이크로컨트롤러 특징들을 가질 수 있다.

상기 인터럽트-생성 유닛은 태스크 라인에 의해 상기 프로그램 가능한 주변기기 상호연결부에 연결될 수 있고, 상기 태스크 라인을 통하여 상기 태스크 신호를 발송할 수 있다.

[0064] 대안으로, 그것은 상기 프로그램 가능한 주변기기 상호연결부에 의해 모니터링되는 태스크 레지스터에 기록할 수 있다.

[0065] 상기 이벤트-생성 주변기기는 이벤트 라인을 통하여 상기 프로그램 가능한 주변기기 상호연결부에 이벤트를 신호하도록 구성될 수 있다. 대안으로, 그것은 상기 프로그램 가능한 주변기기 상호연결부에 의해 모니터링되는 이벤트 레지스터에 기록할 수 있다.

[0066] 상기 인터럽트-생성 유닛은 바람직하게는 태스크 신호를 수신한 것에 바로 응답하여- 바람직하게는 상기 태스크 신호를 수신한 것으로부터 미리 결정된 최대 시간내에, 또는 태스크 신호를 수신한 후에 일정한 시간 지연에서 상기 인터럽트를 발송하도록 배열된다.

[0067] 상기 마이크로컨트롤러는 소프트웨어를 저장하는 메모리를 포함할 수 있고, 상기 소프트웨어는 상기 맵핑 메모리에, 상기 이벤트-생성 주변기기의 이벤트로부터 상기 인터럽트-생성 유닛의 태스크로의 맵핑을 저장하도록 상기 프로세서에 의해 실행 가능한 지시들을 포함한다.

[0068] 상기 측면들 중 임의의 측면에서, 비록 상기 이벤트-생성 레지스터 또는 레지스터들이 상기 이벤트-생성 유닛의 일부인 것으로 말해졌지만, 이것은 상기 마이크로컨트롤러상의 그것들의 물리적 위치를 제한하지 않는 것으로 인식될 것이다. 상기 이벤트-생성 레지스터들은 바람직하게는 예를 들어, 상기 버스를 통하여 상기 프로세서에 의해 자체 어드레스로 호출할 수 있다. 그것들은 바람직하게는 상기 프로세서에 의해 기록되고 및/또는 판독될 수 있다. 각각의 이벤트-생성 레지스터는 메모리의 연접 영역(contiguous region)을 점유할 수 있거나, 또는 그것은 복수의 위치들에 걸쳐서 분리될 수 있다. 본 출원에서 언급된 레지스터는 단지 단일 비트 길이일 수 있거나(어쩌면 더 큰 비트 필드내에), 또는 그것은 복수의 비트들(예를 들어, 32-비트 워드)를 포함할 수 있다.

[0069] 상기 마이크로컨트롤러는 바람직하게는 휘발성 및/또는 비 휘발성 메모리, 예컨대 RAM 및/또는 플래시 메모리를 포함한다. 상기 메모리의 부분은 프로그램 코드를 저장할 수 있다.

[0070] 상기 프로세서는 임의의 적절한 프로세서일 수 있다. 일부 실시예들에서 그것은 ARMTM로부터의 프로세서, 예컨대 ARMTM의 CortexTM 레인지로부터의 프로세서이다. 상기 마이크로컨트롤러는 복수의 버스들, 예컨대 프로세서 버스 및 주변기기 버스를 포함할 수 있다. 상기 이벤트-생성 유닛은 바람직하게는 하나 이상의 버스들에 연결된다.

[0071] 상기 마이크로컨트롤러는 비록 그것은 물론, 동작시키기 위해서 일부 오프-칩 컴포넌트들, 예컨대 크리스탈, 커패시터들, 등을 필요로 할 수 있지만 바람직하게는 통합 디바이스이다. 이들 컴포넌트들은 상기 마이크로컨트롤러의 일부인 것으로 간주될 수 있거나, 또는 그것들은 상기 마이크로컨트롤러와는 별개인 것으로 간주될 수 있다.

[0072] 상기 인터럽트- 또는 이벤트-생성 유닛은 바람직하게는 상기 프로세서와는 별개인 상기 마이크로컨트롤러내 별개의 하드웨어 모듈이다. 상기 인터럽트- 또는 이벤트-생성 유닛은 본 출원에서 설명된 기능들을 제공하도록 배열된 트랜지스터들, 저항기들, 커패시터들, 등을 포함하는 아날로그 및/또는 디지털 컴포넌트들을 포함할 수 있다. 상기 마이크로컨트롤러는 물론, 태스크 라인에 의해 상기 프로그램 가능한 주변기기 상호연결부에 연결되지 않거나 또는 이벤트 라인에 의해 연결되지 않은 다른 주변기기들을 가질 수 있다.

[0073] 일부 실시예들에서, 상기 마이크로컨트롤러는 라디오 트랜시버를 포함한다. 그것은 라디오-온-칩 디바이스(radio-on-a-chip)일 수 있다.

도면의 간단한 설명

[0074] 본 발명의 임의의 선호 실시예들은 첨부 도면들을 참조하여 단지 예제의 방식으로 이제 설명될 것이다:

도 1은 본 발명을 구체화한 제 1 시스템의 컴포넌트들을 보여주는 개략도이다.

도 2는 주변기기 레지스터들 사이에서 맵핑들을 저장하는 메모리 구조를 나타내는 테이블이다;

도 3은 본 발명을 구체화한 제 2 시스템내 PPI에 연결된 몇몇의 주변기기들의 개략도이다;

도 4는 PPI내 특정 채널에 관련된 엘리먼트들의 개략도이다; 및

도 5는 이벤트-생성 유닛을 보여주는 개략도이다.

발명을 실시하기 위한 구체적인 내용

[0075] 도 1은 중앙 프로세싱 유닛(CPU)(6), 메인 메모리(8), 및 로직 영역(12) 및 내부 메모리(14)를 갖는 PPI(10)를 포함하는 마이크로컨트롤러(MCU)(2)(예를 들어, 집적 회로 또는 멀티-칩 모듈)를 도시한다. CPU(6) 및 PPI(10)는 둘 모두 버스(16)에 연결된다.

[0076] 이벤트-생성 유닛(17) 및 다섯개의 대표적인 주변기기들: 타이머(timer)(18), 디지털-아날로그 컨버터(DAC)(20), UART(22), 하드웨어 암호화 엔진(24), 전압 비교기(26)도 또한 버스(16)에 연결된다.

[0077] 타이머(18)는 출력 이벤트 레지스터(28), 오버플로우(overflow) 이벤트 레지스터(30) 및 입력 태스크 레지스터들(32)를 갖는다. DAC(20)는 두개의 변환-마감된(conversion-finished) 이벤트 레지스터들(36,38) 및 두개의 트리거 태스크 레지스터들(40,42)을 갖는다. UART(22)는 RX 수신 이벤트 레지스터(44), TX 완료 이벤트 레지스터(46), RX 트리거 태스크 레지스터(48) 및 TX 트리거 태스크 레지스터(50)를 갖는다. 하드웨어 암호화 엔진(24)은 암호화-마감된 이벤트 레지스터(52) 및 트리거 태스크 레지스터(54)를 갖지만, 전압 비교기(26)는 단일 비교기 출력 이벤트 레지스터(56)를 갖는다. 레지스터들 중 일부 레지스터는 단일-비트 레지스터들이지만, 그러나 다른 것들은 멀티-비트 값을 보유할 수 있다. 물론, 다른 실시예들은 상이한 주변기기들을 가질 수 있고, 주변기기들은 임의 개수의 이벤트 또는 태스크 레지스터들을 가질 수 있다.

[0078] 이벤트-생성 유닛(17)은 제 1 이벤트 레지스터(57), 제 2 이벤트 레지스터(59), 제 1 태스크 레지스터(61) 및 제 2 태스크 레지스터(63)를 갖는다. 물론, 다른 실시예들은 상이한 수의 이벤트 또는 태스크 레지스터들을 가질 수 있다. 이 실시예에서, 이벤트-생성 유닛(17)은 제 1 태스크는 제 1 이벤트를 트리거하고, 제 2 태스크는 제 2 이벤트를 트리거하도록 배열된다. 다른 실시예들에서, 태스크 대 이벤트들을 맵핑하는 더 복잡한 로직이 있을 수 있다.

[0079] 주변기기들(18-26) 및 이벤트-생성 유닛(17)의 레지스터들은 메인 메모리(8) 및 PPI 메모리(14)와 메모리 어드레스지정 스페이스를 공유하여, 그것들은 PPI 로직(12)에 의해 및 옵션으로 CPU(6)에 의해 메모리-맵핑된 I/O를 이용하여 액세스될 수 있다.

[0080] 버스(16)는 도시된 바와 같이 단일 버스일 수 있거나, 또는 그것은 두개 이상의 별개의 버스들로 형성될 수 있다.

[0081] 사용시에, CPU(6)는 예를 들어, DAC(20) 변환-마감된 이벤트들 중 하나를 UART(22) 트리거 태스크에 연결하는 채널을 생성하도록 PPI(10)에 지시할 수 있어서, UART(22)는 DAC 변환이 완료된 후에 데이터를 송신한다. 이 지시 수신시에, PPI(10)는 그것의 메모리(14)에 보유된 테이블에 새로운 엔트리를 생성하고, DAC(20) 변환-마감된 레지스터(36)의 어드레스 및 UART(22) 트리거 태스크 레지스터(48)의 어드레스를 링크시킨다.

[0082] CPU(6)는 또한, 예를 들어, 이벤트-생성 유닛(17)의 제 1 이벤트 레지스터(57)를 UART(22)의 TX 트리거 태스크 레지스터(50)에 연결하는 채널을, 그리고 이벤트-생성 유닛(17)의 동일한 제 1 이벤트 레지스터(57)를 타이머(18)의 입력 태스크 레지스터(32)에 연결하는 다른 채널을 생성하도록 PPI(10)에 지시할 수 있다. 이런 식으로, CPU(6)는 UART(22) 가 이벤트-생성 유닛(17)의 제 1 태스크 레지스터(61)에 '1' 비트를 기록함으로써 송신을 시작하는 것과 동시에 타이머(18)를 시작할 수 있다. 이 프로세스는 극미(atomic)하여-즉, 그것은 CPU(6) 인터럽트(interrupt)에 의해 방해되지 않을 것이다.

[0083] 도 2 는 PPI의 메모리(14)에 저장될 수 있는 대표적인 로직 데이터 구조를 도시한다. 그것은 많은 로우들을 가지며, 이들의 각각은 맵핑 번호, 이벤트 레지스터 어드레스 및 태스크 레지스터 어드레스를 함유한다.

[0084] PPI 로직(12)은 레지스터내 값이 변화할 때를 결정하기 위해서 주기적으로 테이블에 열거된 모든 이벤트 레지스터를 폴링(po11)하도록 구성된다. 변화가 감지된 때, PPI 로직(12)은 테이블내 특정한 이벤트 레지스터에 맵핑된 모든 태스크 레지스터들에 새로운 값(또는 미리 결정된 값, 예컨대 바이너리 '1')을 기록한다.

[0085] PPI 로직(12)은 옵션으로 이벤트 레지스터내 새로운 값에 근거하여 일부 프로세싱을 수행하고 태스크 레지스터들 중 하나 이상에 프로세싱의 결과를 대신 기록하도록 구성될 수 있다. 예를 들어, 만약 이벤트 레지스터가 멀티-비트 값을 함유하면, PPI 로직(12)은 그것이 이전 값보다 더 높은 지 여부를 결정하고 메모리(14)내 맵핑에 따라 단일-비트 태스크 레지스터에 일 비트를 기록할 수 있다. PPI(10)는 CPU(6)에 의해 수행할 어떤 프로세싱(만약에 있다면)에 관하여 지시 받을 수 있다. 프로세싱의 유형은 맵핑 테이블내 추가 컬럼으로서 저장될 수 있다.

[0086] 만약 CPU(6)가 채널을 비활성화시키기를 원하면, 그것은 간단하게 PPI의 메모리(14)내 저장된 테이블내 관련 엔트리가 삭제되게 함으로써 이를 수행할 수 있다. 주변기기들에 직접 지시할 필요가 없다(비록 물론 일부 환경들에서 이것을 또한 할 수 있지만).

[0087] 도 3 내지 도 5는 각각의 개별 이벤트 및 태스크 신호에 대한 개별 라인들에 의해 PPI(10)가 주변기기들(18-26) 및 이벤트-생성 유닛(17)에 연결되는 실시예들의 세트에 관한 것이다. 이것은 상기에서 설명된 메모리-맵핑된 입력/출력을 이용하는 것에 대한 대안으로서 수행될 수 있거나, 또는 MCU 2는 양쪽 모드들의 통신을 지원할 수 있다(예를 들어, 레지스터들을 통하여 CPU(6)가 주변기기들과 상호 작용하는 것을 허용하고, 및 전용 라인(dedicated line)들을 통하여 PPI(10)가 주변기기들과 상호 작용하는 것을 허용하는 것).

[0088] 도 3 은 PPI(이는 상기의 것과 동일한 PPI(10), 또는 상이한 PPI 일 수 있다)에 연결된 m 주변기기들을 도시한다. 이벤트-생성 유닛(17)은 이들 목적을 위해 다른 주변기기로서 간주될 수 있다. PPI 는 n 채널들을 제공하고, 이들의 각각은 관련된 이벤트 엔드-포인트 레지스터(EEP) 및 태스크 엔드-포인트 레지스터(TEP)를 갖는다. 각각의 이들 레지스터들은 주변기기들 중 하나의 주변기기 상에 레지스터의 어드레스를 보유할 수 있다.

[0089] 도 4는 이들 n 채널들 중 하나의 채널과 관련된 엘리먼트들의 보다 세부사항을 제공한다. 채널들의 각각은 유사한 셋의 엘리먼트들을 갖는다. 채널의 EEP는 채널과 관련되어 PPI내 다중화기(MUX)로의 입력들 중 하나에 관련되는 내부 이벤트 라인 식별자로 일련의 주변기기 이벤트 레지스터 어드레스들을 교차 참조(cross-reference)하는 이벤트 신호 툭업 테이블에 연결된다. PPI는 EEP에 함유된 어드레스와 관련된 출력을 선택하도록 MUX를 설정하도록 구성된다.

[0090] 유사하게, 채널의 TEP는 채널과 관련되어 PPI내 역다중화기(DEMUX)의 출력들 중 하나에 관련되는 내부 태스크 라인 식별자로 일련의 주변기기 태스크 레지스터 어드레스들을 교차 참조(cross-reference)하는 태스크 신호 툭업 테이블에 연결된다. PPI는 TEP에 함유된 어드레스와 관련된 출력을 선택하도록 DEMUX를 설정하도록 구성된다.

[0091] 두개의 주변기기들 A, B가 채널의 MUX에 연결된다. 이들 중 하나는 이벤트-생성 유닛일 수 있다. 주변기기 A는 두개의 이벤트 신호들 AE0, AE1를 제공할 수 있고, 이들의 각각은 MUX로 그것 자체의 라인을 가진다. 주변기기 B는 또한 MUX로 라인을 갖는 단일 이벤트 신호 BE0를 제공할 수 있다. 유사한 라인들이 다른 n - 1 채널들(미도시)의 다중화기들에 연결될 것이다.

[0092] 채널의 DEMUX는 세개의 출력 라인들에 연결되고, 하나는 주변기기 A 상의 태스크 입력 AT0에 리딩(leading)되고 두개는 주변기기 B 상의 상이한 태스크 입력들 BT0, BT1에 리딩된다.

[0093] PPI내에서, MUX로부터의 출력은 스위치를 통하여 DEMUX에 입력으로 연결된다. 스위치가 클로우즈(close)될 때 연결이 이루어져서, MUX에 의해 선택된 주변기기 중 하나로부터 이벤트 신호(예를 들어, 펄스)가 DEMUX로 전달되고 따라서 주변기기들 중 하나의 선택된 태스크 입력으로 온 된다. 스위치의 상태는 CPU에 의해 기록가능한 레지스터에 제어된다(미도시).

[0094] 일부 대안 배열들에서, 하나의 채널은 다수의 역다중화기들을 가질 수 있고, 각각은 채널의 MUX의 출력에 연결된다. 이런 식으로, 단일 이벤트는 복수의 주변기기 태스크들을 트리거하도록 분기(fork)될 수 있다.

[0095] PPI는 확신 이벤트 채널을 수신하는 것으로부터 미리 결정된 최대 시간내에 태스크 신호들을 발행하도록 배열된다. 일부 실시예들에서, 이 시간 지연은 모든 채널들에 대하여 일정하다. 상이하거나 또는 분기된 채널들상에 신호들은 동시에 핸들링된다.

[0096] 도 5는 이벤트-생성 유닛(17)의 내부를 도식적으로 예시한다.

[0097] 특정 태스크 n에 대하여, 이벤트-생성 유닛(17)은 태스크 신호가 트리거 태스크 n에 발송될 수 있는 입력 태스크 라인(64)에 의해 PPI(10)에 연결된다. 이벤트-생성 유닛(17)은 또한 만약 CPU(6)상에 운영하는 펌웨어가 바이너리 '1'를 태스크 레지스터(66)에 기록하면 태스크를 트리거하는 신호를 발송하는 태스크 n과 관련된 태스크 레지스터(66)를 또한 갖는다. 태스크 레지스터(66)의 컨텐트들은 자동으로 PPI가 태스크 신호를 태스크 라인(64)을 통하여 발송할 때 이벤트-생성 유닛(17)에 의해 또한 자동으로 업데이트될 수 있다. PPI로부터 태스크 라인(64) 및 태스크 레지스터(66)의 상태를 나타내는 출력은 로직 또는 케이트(68)에 결합되고 이벤트-생성 유닛(17)의 코어(70)로 발송된다.

[0098] 간단한 실시예에서, 코어(70)는 각각의 태스크 n를 일-대-일 맵핑으로 개별 이벤트 n과 간단하게 연결한다. 보다 복잡한 실시예들에서, 코어(70)는 예를 들어, "이벤트 1" 신호를 보내기 전에 "태스크 1" 및 "태스크 2" 둘

모두가 트리거되는 것을 요구하는 하나 이상의 태스크들을 하나 이상의 이벤트들에 연관시키는 로직을 포함할 수 있다.

[0099] 코어(70)는 이벤트 m 에 대한 이벤트 신호를 출력할 수 있다. 이것은 이벤트가 신호되었다는 것을(예를 들어, 바이너리 '0'으로부터 바이너리 '1'로 단일-비트 레지스터의 컨텐츠들을 변화시킴으로써) 나타내기 위해 관련된 이벤트 레지스터(72)의 컨텐츠들을 바꾸게 한다. 동일한 이벤트 신호는 또한 분열되고 이벤트 출력 라인(74)을 통하여 PPI(10)로 발송된다.

[0100] 이벤트 레지스터(72)에 대한 이 변화는 또한 CPU(6)내 인터럽트 제어기로 인터럽트 신호 라인(76)을 통하여 신호가 발송되도록 구성될 수 있다. 이 인터럽트 신호가 발송되는지 여부는 CPU(6)에 의해 기록될 수 있는 인터럽트-인에이블(interrupt-enable) 레지스터(78)의 상태에 의존한다. 만약 바이너리 '1'이 이벤트 n 에 해당하는 비트 위치에 기록되면 그러면 스위치(80)은 클로우즈되고 인터럽트 신호가 발송될 것이다.

[0101] CPU(6)는 많은 상이한 방식들로 PPI(10)와 조합하여 이벤트-생성 유닛(17)을 사용할 수 있다.

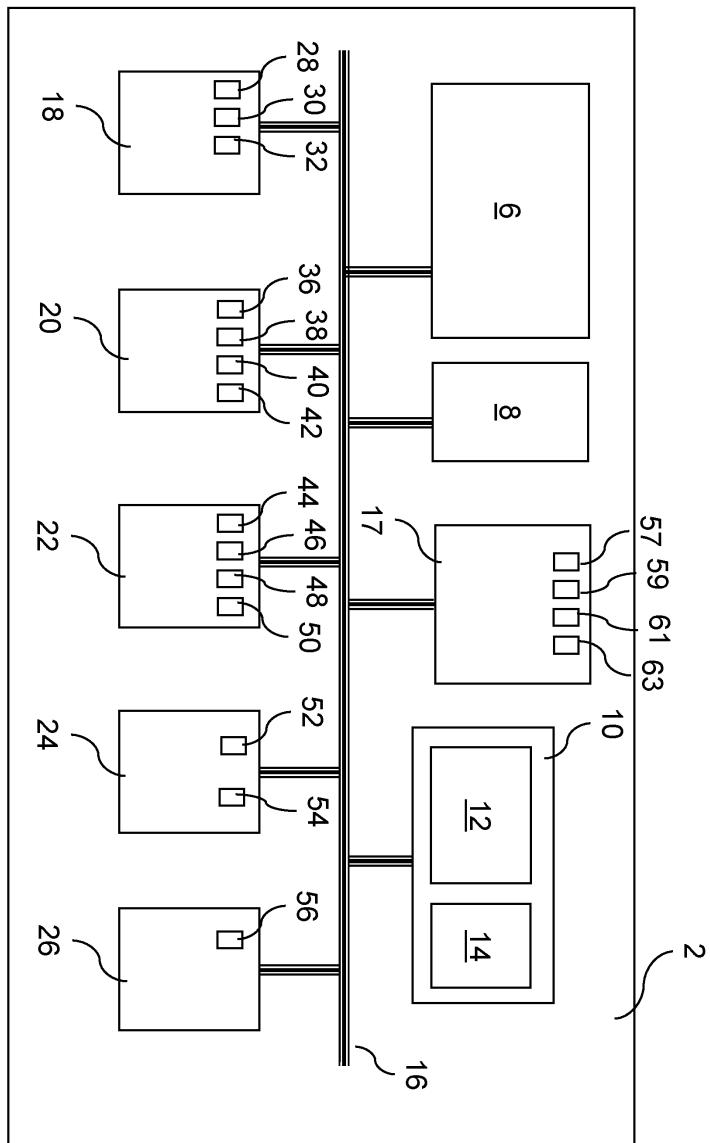
[0102] 예를 들어, CPU(6)는 이벤트-생성 유닛(17)으로부터 이벤트 n 에 대한 신호를 수신한 것에 응답하여 두개 이상의 주변기기 태스크들을 트리거하도록 PPI(10)를 구성할 수 있다. CPU(6)는 그런 다음 이벤트-생성 유닛(17)상에 태스크 레지스터(66)에 '1' 비트를 기록할 수 있고, 이는 n 번째 이벤트 신호 라인(74)을 통하여 이벤트 신호를 이벤트-생성 유닛(17)이 PPI(10)로 발송하게 할 것이다. PPI(10)는 그런 다음 동시에 그리고 극미하게 (atomically) 태스크들의 전부를 트리거할 것이다.

[0103] 다른 예에서, CPU(6)는 m 번째 인터럽트 신호 라인(76)을 활성화 할 수 있고, 주변기기의 이벤트로부터 이벤트-생성 유닛(17)에 m 번째 입력 태스크 라인으로 채널을 수립하는 PPI(10)에 대한 맵핑을 생성할 수 있다. 주변기기는 그런 다음 주변기기를 PPI(10)에 연결하는 적절한 이벤트 라인을 통하여 이벤트를 PPI(10)에 신호를 보냄으로써 인터럽트가 CPU(6)에 발송되게 할 수 있다. 이런 식으로, 주변기기는 CPU(6)에 대한 그 자체의 인터럽트 라인에 할당된 우선 순위와 다른 우선 순위에서 CPU(6)를 인터럽트할 수 있다.

[0104] 본 발명은 발명의 하나 이상의 특정 실시예들을 설명함으로써 예시되었지만, 그러나 이를 실시예들에 제한되지 않고; 첨부 청구항들의 범위내에서 많은 변형예들 및 수정예들이 가능하다는 것이 당해 기술분야의 통상의 기술자들에 의해 인식될 것이다.

도면

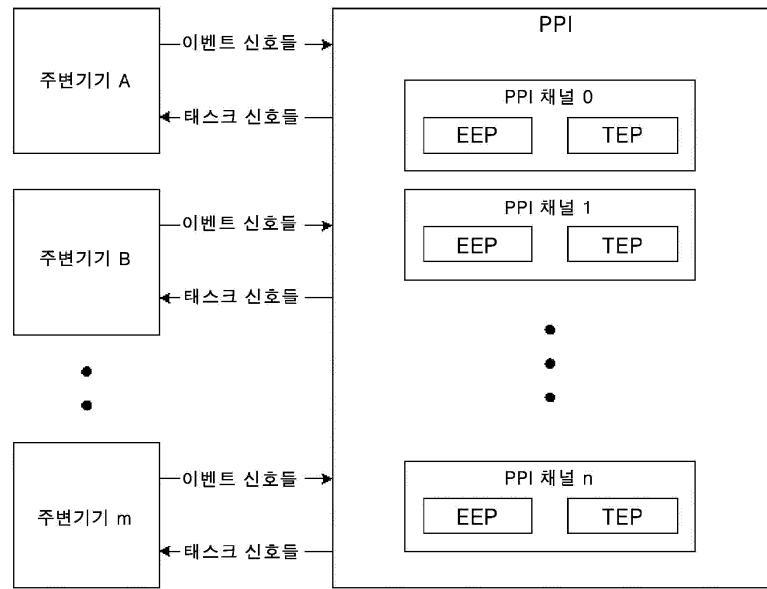
도면1



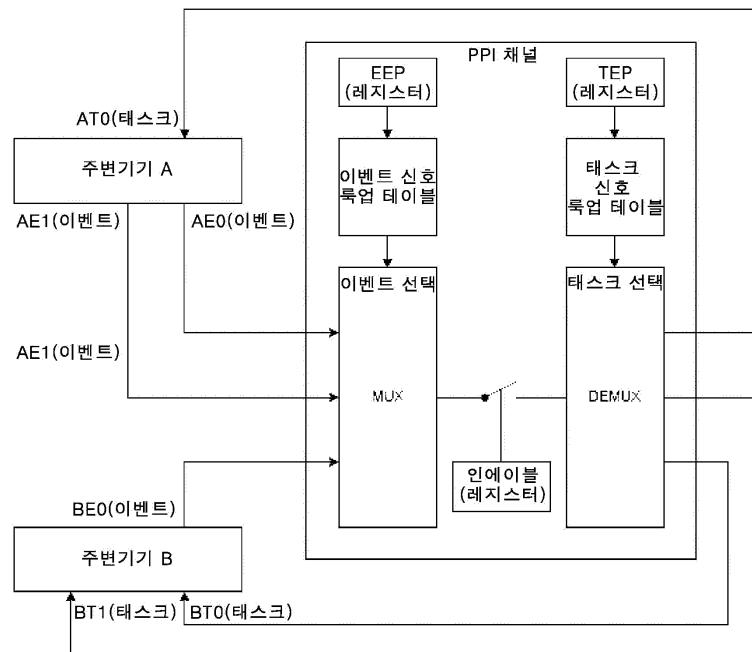
도면2

맵핑 #	이벤트 레지스터	태스크 레지스터
1	0x100	0x500
2	0x300	0x620
3
...

도면3



도면4



도면5

