

**(19) 대한민국특허청(KR)**  
**(12) 특허공보(B1)**

(51) Int. Cl.<sup>5</sup>  
G06F 13/24

(45) 공고일자 1995년01월24일  
(11) 공고번호 특 1995-0000552

---

(21) 출원번호	특 1991-0017866	(65) 공개번호	특 1992-0008609
(22) 출원일자	1991년 10월 11일	(43) 공개일자	1992년 05월 28일

---

(30) 우선권주장	90-270477	1990년 10월 11일	일본(JP)
(71) 출원인	가부시키가이샤	도시바	아오이 죠이치

일본국 가나가와현 가와사키시 사이와이구 호리가와정 72번지

(72) 발명자	기시가미 히데치카
(74) 대리인	일본국 가나가와현 가와사키시 사이와이구 호리가와정 580번 1호 가부시키가이샤 도시바 반도체시스템 기술센터내
	이범일, 김윤배

**심사관 : 허상무 (책자공보 제3852호)**

---

**(54) 마이크로프로세서**

---

**요약**

내용 없음.

**대표도**

**도1**

**명세서**

[발명의 명칭]

마이크로프로세서

[도면의 간단한 설명]

제 1 도는 본 발명에 따른 마이크로프로세서의 기구 블럭도,  
제 2 도는 본 발명에 따른 마이크로프로세서를 사용한 컴퓨터의 시스템 구성도,  
제 3 도 및 제 4 도는 종래의 마이크로프로세서를 사용한 컴퓨터의 시스템 구성도이다.

\* 도면의 주요부분에 대한 부호의 설명

1 : 제 1 기억수단	3 : 제 2 기억수단
5 : 제 3 기억수단	7 : 제 4 기억수단
9 : 제 5 기억수단	21 : 마이크로프로세서
23 : 내부 마이크로프로세서	25 : PSW
27 : PC	29 : CPU코어
31, 33 : 내부배선	41 : 메모리
43 : 인터럽트스택	45 : 인터럽트벡터테이블

[발명의 상세한 설명]

[산업상의 이용분야]

본 발명은 마이크로프로세서에 관한 것으로, 특히 실시간 제어등의 고속인터럽트대응이 요구되는 제어용 프로세서에 사용되는 인터럽트처리 기술에 관한 것이다.

[종래의 기술 및 그 문제점]

제 3 도는 종래의 마이크로프로세서의 인터럽트처리를 설명하기 위한 컴퓨터의 구성도로서, 마이크로프로세서(23)와 메모리(41)가 외부버스(51)에 의해 접속되어 있고, 상기 마이크로프로세서(23)는 CPC코어(29)와, PSW(25 ; Program Status Word) 및 PC(27 ; Program Counter)등을 갖추고 있으며, 또한 메모리(41)는 인터럽트스택(43) 및 인터럽트벡터테이블(45)등을 갖추고 있다.

이와 같은 컴퓨터의 구성에서의 인터럽트처리[여기서는 인터럽트신호가 마이크로프로세서(23)에 입

력되면서부터 인터럽트핸들러에 분기되기까지의 처리]는 통상, 다음의 수순으로 수행된다. 먼저, 마이크로프로세서(23)가 인터럽트신호를 접수하면, 우선 제 1 스텝에서 마이크로프로세서(23)의 현재의 상태[현재의 PC(27), PSW(25)등의 내용]를 인터럽트스택(43)으로 퇴피시킨다. 다음에 제 2 스텝에서 인터럽트벡터(인터럽트핸들러에서의 PC, PSW등)를 인터럽트벡터테이블(45)로부터 읽어들이고, 제 3 스텝에서 인터럽트핸들러로 분기한다.

예컨대 도시바제작 TRON 사양 32비트 MPU「IX1」이 이와 같은 수순으로 인터럽트처리를 수행한다면, 스텝(1)에서 12~28바이트의 메모리기록과, 스텝(2)에서 8바이트의 메모리독출이 필요하다. 상기 메모리(41)가 DRAM으로 구성되어 있는 경우에는 1회(4바이트)의 메모리엑세스에 5클럭정도 필요하기 때문에 스텝(1)에서 60~140클럭, 스텝(2)에서 40클럭을 요구하는 것으로 된다. 따라서, 본 예와 같은 컴퓨터에서는 인터럽트처리에 최저로 100~180클럭이 소요되어, 응답이 지연됨으로써 실시간 제어에 사용할 수 없다는 문제가 있었다.

또한, 인터럽트처리를 고속으로 실행하기 위하여 메모리(41)의 일부를 억세스시간이 고속인 SRAM으로 구성하면, 2클럭으로 메모리를 억세스할 수 있게 되고, 인터럽트처리가 40~72클럭정도에서 실행되어 다소 인터럽트응답이 개량되지만, 메모리시스템이 고가로 되는 문제가 남는다.

더욱이, 제 4 도에 나타낸 바와 같이 마이크로프로세서 내부에 온칩의 RAM[61 ; 인터럽트스택(63), 인터럽트벡터테이블(65)]을 설치하는 수법이 널리 이용되는데, 이 경우, 1클럭으로 억세스할 수 있게 됨으로써 인터럽트처리가 20~36클럭정도에서 실행되어 인터럽트응답이 개선되지만, 온칩에 메모리영역이 필요로되어 칩비용이 상승하는 문제가 있다.

이상 설명한 바와 같이, 종래의 마이크로프로세서의 인터럽트처리기술에서는 메모리를 DRAM으로 구성하는 경우, 인터럽트응답이 지연되어 실시간 제어에 사용될 수 없었다. 또한 메모리를 SRAM으로 구성하거나 마이크로프로세서 내부에 온칩의 RAM을 구성하는 경우에는 응답시간이 몇분 개선되지만 비용이 상승한다는 결점이 있었다.

#### [발명의 목적]

본 발명은 상기한 점을 감안하여 발명된 것으로, 간단한 하드웨어의 추가로 고속 인터럽트처리를 실현할 수 있는 마이크로프로세서를 제공함에 그 목적이 있다.

#### [발명의 구성]

상기 목적을 달성하기 위한 본 발명은, 프로세서의 상태를 보유하는 제 1 기억수단(1)과, 실행중 명령의 선두어드레스를 보유하는 제 2 기억수단(3)을 갖춘 마이크로프로세서에 있어서, 특정의 인터럽트발생시에 상기 제 1 및 제 2 기억수단(1, 3)의 내용을 일시적으로 퇴피하는 1개 이상의 제 3 기억수단(5)과, 이 제 3 기억수단(5)의 내용이 유효인지 아닌지의 여부를 나타내는 1비트이상 제 4 기억수단(7) 및, 1개이상의 인터럽트벡터를 보유하여 상기 특정의 인터럽트발생시에 대응하는 인터럽트벡터를 상기 제 1 및 제 2 기억수단(1, 3)에 격납하는 제 5 기억수단(9)으로 구성된 것을 특징으로 하는 마이크로프로세서로 구성된다.

#### [작용]

상기와 같이 구성된 본 발명은, 특정 인터럽트가 발생되면, 제 1 기억수단(1) 및 제 2 기억수단(3)의 내용을 제 3 기억수단(5)으로 퇴피시키고, 제 4 기억수단(7)의 대응하는 1비트의 내용을 1로 세트함과 동시에 제 5 기억수단(9)으로부터 대응되는 인터럽트벡터를 제 1 기억수단(1) 및 제 2 기억수단(3)에 격납하여, 인터럽트핸들러로 분기한다. 또한, 인터럽트핸들러로부터 복귀하는 경우에는 전용명령에 의해 제 4 기억수단(7)의 대응하는 1비트의 내용이 0에 리세트되어, 상기 전용명령에 의해 제 3 기억수단(5)으로 퇴피된 내용을 제 1 기억수단(1) 및 제 2 기억수단(3)으로 격납한다.

또한 본 발명은, 특정 인터럽트가 발생되면, 제 1 기억수단(1) 및 제 2 기억수단(3)의 내용을 제 3 기억수단(5)으로 퇴피시키고, 제 4 기억수단(7)의 대응하는 부분의 내용을 +1 또는  $\times 2$ 로 하며, 동시에 제 5 기억수단(9)으로부터 대응하는 인터럽트벡터를 제 1 기억수단(1) 및 제 2 기억수단(3)에 격납하여 인터럽트핸들러에 분기한다. 또한, 상기 인터럽트 처리중에 다른 인터럽트가 발생될 때마다 제 4 기억수단(7)의 대응하는 부분의 내용을 +1 또는  $\times 2$ 로 하고, 인터럽트핸들러로부터 복귀하는 경우에는 전용명령에 의해 제 4 기억수단(7)의 인터럽트처리중에 있는 부분의 내용을 -1 또는  $\div 2$ 로, 더욱이 제 4 기억수단(7)의 내용이 0으로 된 경우에는 상기 전용명령의 실행에 의해 제 3 기억수단(5)으로 퇴피한 내용을 제 1 기억수단(1) 및 제 2 기억수단(3)에 격납시킨다.

#### [실시예]

이하, 예시도면을 참조하여 본 발명에 따른 1 실시예를 상세히 설명한다.

제 2 도는 본 발명에 따른 고속인터럽트처리가 가능한 마이크로프로세서의 실시예를 나타낸 것으로, 마이크로프로세서(21)와 메모리(41)는 외부버스(51)에 의해 접속되어 있고, 상기 마이크로프로세서(21)는 내부마이크로프로세서(23)를 갖추고 있으며, 상기 내부마이크로프로세서(23)는 프로세서상태를 보유하는 PSW(25)와, 현재 실행중인 명령의 선두어드레스를 보유하는 PC(27) 및, CPU코어(29)를 갖추고 있다. 더욱이, 상기 CPU코어(29)는 통상의 마이크로프로세서가 갖는 명령디코더, 연산기, 레지스터화일등을 포함한다.

그리고, 본 실시예에 따른 마이크로프로세서(21)는 특히 인터럽트벡터번호인 #218 및 #129의 2종류의 인터럽트를 고속으로 처리하기 위한 구성요소로서, 인터럽트용 스택레지스터(SR0, SR1)와, 인터럽트벡터레지스터(IRQ, IR1) 및 유효플랙(V0, V1)을 갖추고 있고, 인터럽트벡터번호 #128에 대해서 SR0, IRQ, V0가, #129에 대해 SR1, IR1, V1이 준비되어 있다.

각 인터럽트용 스택레지스터(SR0, SR1)는 인터럽트처리에 의해서 스택으로 퇴피되는 PSW, EITINF(인터럽트번호등), PC의 각 필드로 구성되고, 또한 인터럽트벡터레지스터(IRQ, IR1)는 통상 메모리(4

1)에 배치되는 인터럽트벡터테이블(45)의 일부로서, 인터럽트핸들러에서의 PSW, PC를 보유하는 각 필드로 구성되어 있다. 또한 유효플랙(V0, V1)은 인터럽트용 스택레지스터가 유효인지 아닌지의 여부를 나타내는 플랙으로, 예컨대 인터럽트용 스택레지스터(SR0)가 사용되고 있는 경우에는 V0의 내용이 1로 된다. 결국, 인터럽트처리의 마이크로프로그램이 기동되면 세트되고, 인터럽트핸들러의 복귀명령인 REIT명령을 처리하는 마이크로프로그램에 의해 리세트된다.

더욱이, 인터럽트용 스택레지스터(SR0, SR1) 및 인터럽트벡터레지스터(IR0, IR1)는 각각 내부배선(31, 33)에 의해 내부 마이크로프로세서(23)와 접속되어 있고 동시에 억세스가 가능하다. 더욱이, 메모리(41)는 인터럽트벡터번호 #128 및 #129이외의 인터럽트를 처리하기 위해 인터럽트스택(43) 및 인터럽트벡터테이블(45)을 갖추고 있다.

이하, 본 실시예에 의한 인터럽트처리의 동작을 설명한다.

먼저, 인터럽트가 발생되어, 인터럽트처리의 마이크로프로그램의 실행이 개시되면, 그 인터럽트벡터번호가 #128 또는 #129인가를 판단하게 된다. #128 또는 #129가 아닐 때에는 종래예에 나타낸 통상의 처리[스텝(1)에서 스텝(3)]로 되어, 외부의 메모리(41)에 대해 억세스가 수행된다. 한편, #129 또는 #129인 경우에는 인터럽트용 스택레지스터(SR0 또는 SR1)에 내부마이크로프로세서(23)의 상태가 퇴피됨과 동시에 대응하는 인터럽트벡터레지스터(IR0 또는 IR1)로부터 인터럽트핸들러의 PSW, PC가 마이크로프로세서(23)내부의 PSW(25), PC(27)에 로드되어 인터럽트핸들러에 분기된다. 또한 이 때, 유효플랙(V0 또는 V1)이 세트된다. 이들의 처리는, 예컨대 마이크로프로세서(21)내부에서 인터럽트용 스택레지스터(SR0, SR1)와 인터럽트벡터레지스터(IR0, IR1)에 동시에 억세스가능한 구성으로 하거나 또는 하드웨어에 의해 PSW(25), PC(27)와, 인터럽트용 스택레지스터(SR0 또는 SR1)의 PSW, PC필드를 교체하는 것들을 수행함으로써 수eler정도로 실행할 수 있게 된다.

한편, REIT명령에 의해 인터럽트핸들러로부터 원래의 프로그램으로 복귀하는 경우에는 REIT명령을 처리하는 마이크로프로그램에서 유효플랙(V0, V1)이 점검되고, 이것이 세트되지 않으면, 통상의 처리[메모리(41)의 인터럽트스택(43)으로부터 상태를 마이크로프로세서(23)의 PSW(25), PC(27)에 로드하는 처리]가 수행되지만, 유효플랙(V0, V1)의 어느 한쪽이 세트되게 되면 메모리(41)의 인터럽트스택(43)대신 유효플랙의 세트되어 있는 인터럽트용 스택레지스터(SR0 또는 SR1)로부터 상태를 마이크로프로세서(23)의 PSW(25), PC(27)에 로드하고, 유효플랙(V0 또는 V1)을 리세트한다.

또한, #128의 인터럽트핸들러실행중에 #129의 인터럽트가 발생한 경우에는 미리 인터럽트의 우선순위를 결정함으로써 대처한다. 즉, #129의 인터럽트쪽이 우선순위가 높은 경우에는 #129의 인터럽트핸들러에 분기한다. 이 인터럽트핸들러로부터 복귀시(REIT 명령실행시)에는 유효플랙(V0, V1)이 모두 1로 세트되어 있지만, 이 경우 우선순위가 높은 인터럽트에 대응하는 유효플랙(V1)을 리세트하도록 하여, REIT명령으로 #128의 인터럽트핸들러로 복귀하고, 더욱이 #128의 인터럽트핸들러로부터의 복귀시에 #128에 대응하는 인터럽트용 스택레지스터(SR0)의 유효플랙(V0)이 리세트된다. 반대로 #129의 인터럽트쪽이 우선순위가 낮은 경우에는 #128의 인터럽트핸들러실행중 #129의 인터럽트처리는 마스크되고, #128의 인터럽트핸들러로부터의 복귀후에 #129의 인터럽트처리가 수행된다.

더욱이, 본 실시예에서는 다중으로 인터럽트가 발생한 경우에는, 본 발명의 기능을 이용할 수 있지만, 유효플랙(V0, V1)을 업/다운카운터 또는 시프트레지스터로 구성하여, 인터럽트가 발생한 때에  $+1$  또는  $\times 2$ , REIT명령에서  $-1$  또는  $\div 2$ 로 하도록 하여 유효플랙(V0 또는 V1)이 0으로 될 때에, REIT명령의 처리에서, 대응하는 인터럽트용 스택레지스터(SR0 또는 SR1)를 사용하도록 구성하면, 다중인터럽트에도 대처할 수 있게 된다.

#### [발명의 효과]

이상 설명한 바와 같이 본 발명에 의하면, 특히 고속대응이 필요한 인터럽트의 처리를 마이크로프로세서내부의 인터럽트용 스택레지스터, 인터럽트벡터레지스터를 이용해서 수행함으로써 고속인터럽트처리를 간단한 하드웨어의 추가로 실현할 수 있게 된다.

#### (57) 청구의 범위

##### 청구항 1

프로세서의 상태를 보유하는 제 1 기억수단(1)과, 실행중 명령의 선두어드레스를 보유하는 제 2 기억수단(3)을 갖춘 마이크로프로세서에 있어서, 특정의 인터럽트발생시에 상기 제 1 및 제 2 기억수단(1, 3)의 내용을 일시적으로 퇴피하는 1개이상의 제 3 기억수단(5)과, 이 제 3 기억수단(5)의 내용이 유효인지 아닌지의 여부를 나타내는 1비트이상의 제 4 기억수단(7) 및, 1개이상의 인터럽트벡터를 보유하여 상기 특정의 인터럽트 발생시에 대응하는 인터럽트벡터를 상기 제 1 및 제 2 기억수단(1, 3)에 격납하는 제 5 기억수단(9)으로 구성된 것을 특징으로 하는 마이크로프로세서.

##### 청구항 2

제 1 항에 있어서, 상기 제 4 기억수단(7)의 내용은 특정 인터럽트발생에 의해 대응하는 1비트가 1로 세트되고, 인터럽트핸들러로부터 복귀하는 전용명령에 의해 대응하는 1비트가 0으로 리세트되며, 상기 제 4 기억수단(7)의 내용의 해당 비트가 0으로 리세트된 경우에 상기 전용명령의 실행에 의해 상기 제 3 기억수단(5)의 대응하는 인터럽트벡터를 상기 제 1 및 제 2 기억수단(1, 3)에 격납하도록 된 것을 특징으로 하는 마이크로프로세서.

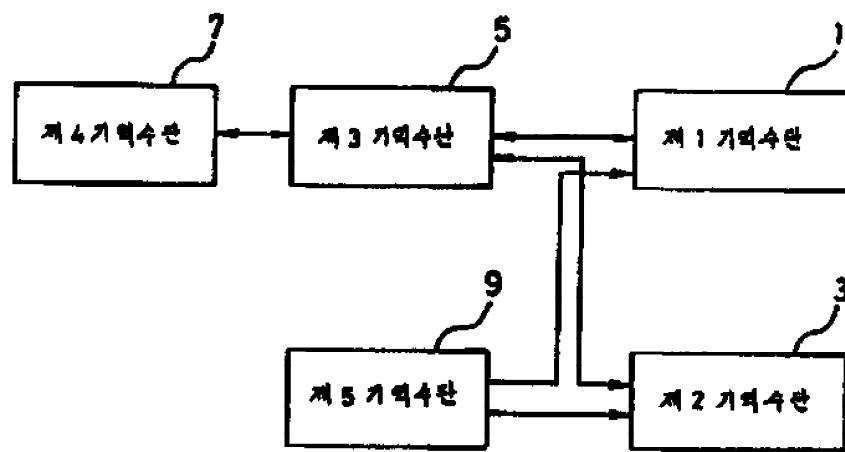
##### 청구항 3

제 2 항에 있어서, 상기 제 4 기억수단(7)은 1개이상의 업/다운카운터 또는 시프트레지스터로 구성되고, 상기 기억수단(7)의 내용은 상기 특정 인터럽트발생에 의해 대응하는 부분이  $+1$  또는  $\times 2$ 로 되고, 상기 인터럽트처리중에 다른 인터럽트가 발생할 때마다 상기 부분이  $+1$  또는  $\times 2$ 로되며, 상

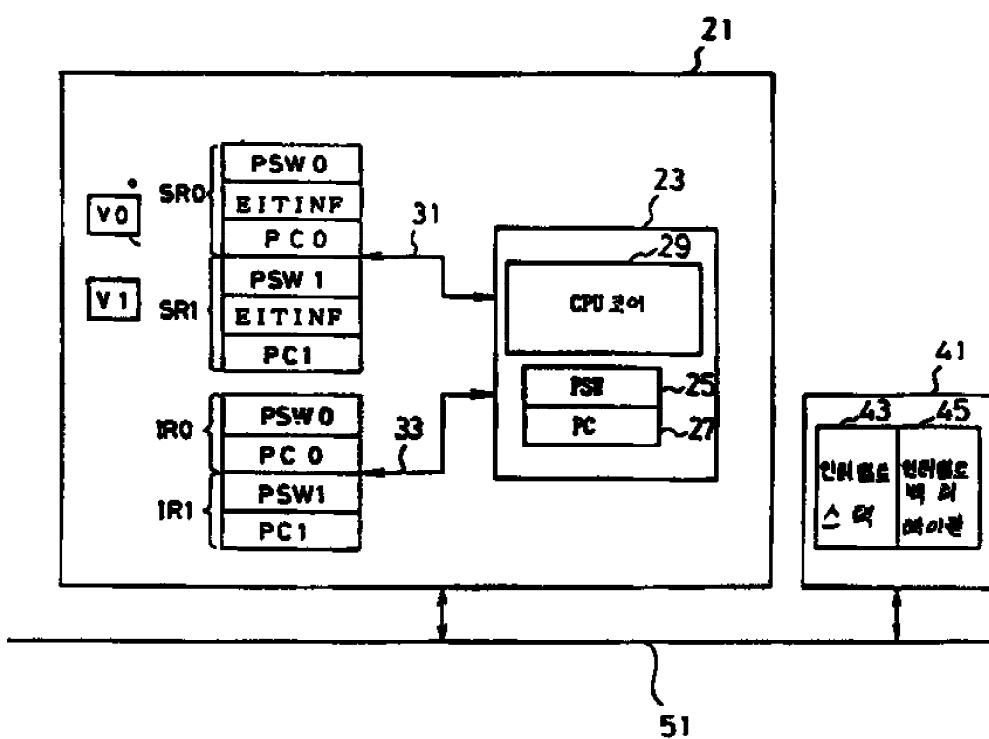
기 전용명령의 실행할때마다 인터럽트처리중의 해당 부분이 -1 또는  $\div 2$ 로 되는 것을 특징으로 하는 마이크로프로세서.

## 도면

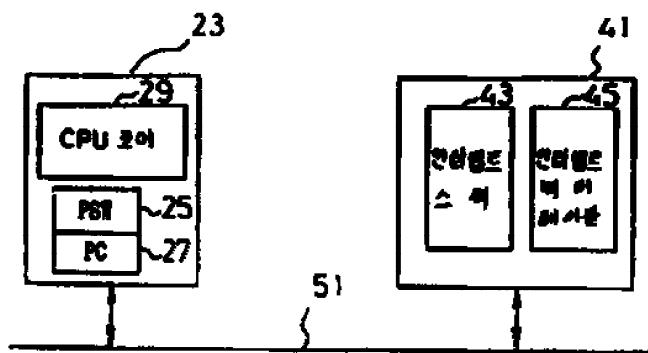
## 도면1



## 도면2



도면3



도면4

