(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) Int. Cl.[6]
G06F 9/38

(45) 공고일자 2002년11월30일
(11) 등록번호 10-0335785
(24) 등록일자 2002년04월24일

(21) 출원번호	10-1996-0701507	(65) 공개번호	특1996-0705271
(22) 출원일자	1996년03월22일	(43) 공개일자	1996년10월09일
번역문제출일자	1996년03월22일		
(86) 국제출원번호	PCT/GB1994/01793	(87) 국제공개번호	WO 1995/08801
(86) 국제출원일자	1994년08월16일	(87) 국제공개일자	1995년03월30일
(81) 지정국	국내특허 : 중국 일본 대한민국 미국 EA 유라시아특허 : 러시아 EP 유럽특허 : 오스트리아 벨기에 스위스 리히텐슈타인 사이프러스 독일 덴마크 스페인 핀랜드 프랑스 영국 그리스 아일랜드 이탈리아 룩셈부르크 모나코 네덜란드 포르투칼 스웨덴		

(30) 우선권주장 9319662.4 1993년09월23일 영국(GB)

(73) 특허권자 에이알엠 리미티드
영국 캠브리지 씨비1 9엔제이 체리힌톤 풀번로드 110
(72) 발명자 자가르 데이비드 비비안
영국 캠브리지 CB1 4TN 체리힌톤 맨드릴 클로스 48
(74) 대리인 이화익

심사관 : 정준모

(54) 데이타처리명령의실행

명세서

<1> **발명의 배경**

<2> **발명의 분야**

<3> 본 발명은 데이터 처리 명령의 실행에 관한 것이다.

<4> **종래기술 설명**

<5> 일부 데이터 프로세서는 최근 실행된 데이터 처리 명령의 제어 하에, 중간 메모리 관리부를 경유해서 랜덤 액서스 메모리(RAM)에 기억된 데이터에 액세스할 수 있는 중앙처리장치(CPU)를 구비한다. 이전에 제안된 그러한 데이터 프로세서의 예로서는 1993년 Advanced Risc Machines Limited에 의해 발표된 "ARM6 Data Sheet"에 기술된 ARM 6 프로세서가 있다.

<6> 메모리 액서스시, 메모리 관리부는 현재의 메모리 액서스가 완료될 수 없다는 것을 나타내는 중단신호를 발생할 수도 있다. 많은 이유로 중단 신호가 발생될 수 있다. 일례로, 중단된 메모리 액서스는 가상 메모리를 사용하는 데이터 처리 시스템에서 발생할 수 있다. RAM과 보다 느린 디스크 기억장치 사이의 데이터 교환은 가상 메모리에서 일어난다. 따라서, 어드레스 가능한 메모리 공간이 실재 RAM의 양보다 크다는 착각을 일으키게 한다. 이러한 시스템에 있어서, 필수의 가상 어드레스에 대응하는 데이터가 RAM보다는 디스크 기억장치에 보유되고 있으면, 이들 데이터가 쉽게 액서스하기 전에 지연이 있을 것이고, 그 지연 동안 데이터는 디스크 기억장치에서 RAM으로 전송되어야 할 것이다. 이 경우에, 현재의 메모리 액서스가 중단되고, 나중에 이들 데이터에 액서스하는 것이 시도된다.

<7> 메모리 관리부로부터 공급된 중단 신호는 너무 늦게 발생하기 때문에, 이미 메모리 액서스를 시작하였으나 액서스에 실패한 명령의 실행을 중단시킬 수는 없지만, 그 대신 다음의 데이터 처리 명령의 실행, 즉 이미 메모리 액서스를 시작하였으나 액서스에 실패한 그 명령 이후의 데이터 처리 명령의 실행을 취소하는 데에는 사용될 수 있다. 이것은 이후의 명령이 이전에 메모리 액서스가 성공하였느냐에 따라 달라지기 때문에 유용하다.

<8> 중단 신호를 이용하여, 이미 메모리 액서스를 시작하였으나 액서스에 실패한 명령 이후에 즉시 이어지는 다음의 명령의 실행을 취소하려면 중단 신호의 정확한 타이밍을 요구한다. 그렇지 않으면, 그 명령의 실행을 완료한 후에, 즉시 이어지는 다음 명령의 실행 결과를 "원상태"로 돌리기 위해 복잡한 메커니즘이 필요하게 된다.

<9> 첨부된 도면 중 제1 도는 상술한 이전에 제안된 데이터 프로세서에 의한 (데이터가 RAM에 기록되는) 데이터 기록동작 중의 중단 신호의 타이밍 요구를 나타내는 개략적인 타이밍도이다.

<10> 제1 도를 참조하면, 클럭신호(10)는 데이터 프로세서에 의해 데이터 처리 명령의 실행을 제어한다. 데이터 기록 동작이 시작되면, 메모리 어드레스(20)는 데이터 프로세서에 의해 메모리 관리부에 공급되고, 1/2 사이클의 클럭신호 후에, 그 어드레스에 기록되어야 할 데이터(30)가 데이터 프로세서에 의해 출력된다.

<11> 메모리 어드레스(20)가 무효하다는 것을 메모리 관리부가 검출하면(예컨대, 그 어드레스에 대응하는 데이터가 가상 메모리 시스템 내의 디스크 기억장치에 보유되어 있기 때문에), 중단신호(40)는 메모리 관리부에 의해 발생되어 데이터 프로세서에 공급된다.

<12> 기록되어야 할 데이터가 데이터 버스 상에 배치된 후에는 메모리 관리부로부터의 응답을 기다릴 필요가 없기 때문에, 데이터 기록 명령 후의 다음 명령이 즉시 실행된다. 따라서, 중지 신호가 바로 다음의 명령의 실행을 취소하려는 시간에 도달하기 위해서는, 상술한 이전에 제안된 데이터 프로세서가 유효한 중단신호를 요구하고, 1/2 사이클의 클럭신호 전에 기록되어야 할 데이터는 데이터 프로세서에 의해 출력된다.

<13> 실질적으로, 이 타이밍 제약조건 때문에 완수하기 어렵고, 특히 이 타이밍 계약조건은 특히 (메모리 관리부에 의해 높은 전력소모를 가지는) 메모리 관리부의 고속 동작을 필요로 한다.

<14> 상술한 이전에 제안된 데이터 프로세서도 그것의 전체 명령 세트의 조건부 실행을 제공한다. 이것은 4개의 처리 플래그까지의 현재의 상태와 각 명령에 포함된 조건 코드에 의해 정의된 각각의 상태를 비교함으로써 달성된다. 이 비교는 명령의 실행과 동시에 발생한다. 그 후 처리 플래그의 상태가 조건 코드에 의해 지정된 것과 부합하지 않으면, 전부의 명령의 실행이 방지된다.

<15> 이 형태의 데이터 프로세서를 설명하는 다른 문헌으로서는 "ARM 6: A High Performance Low Power Consumption Macrocell", M.Muller, COMPCON Spring 93. pages 80~87이 있다. 이것은 상기 언급된 ARM 6 프로세서의 메모리 중단 및 조건부 명령 실행 특징에 대하여 기술한다.

<16> **발명의 요약**

<17> 본 발명은 연속하는 데이터 처리 명령이 파이프라인 방식으로 실행되는 데이터 처리장치를 제공하는 데, 상기 장치는 1개 또는 그 이상의 명령에 응답하여 데이터 메모리에 액서스하고, 각 메모리 액서스가 무효한지의 여부를 검출하는 수단을 구비하는 메모리 액서스 수단과; 이전에 실행된 명령에 의해 발생되며 각 명령의 실행 중에 동작 가능한 데이터 처리장치의 처리 상태에 응답하여, 그 명령이 실행되어야 하는지의 여부를 검출하는 조건 테스트 수단과; 메모리 액서스 수단 및 조건 테스트 수단에 응답하여, 메모리 액서스 수단이 바로 앞의 명령에 의해 시작된 메모리 액서스가 무효하다는 것을 검출하면 또는 조건 테스트 수단이 현재의 명령이 실행되어서는 안 된다는 것을 검출하면 현재의 명령의 전부의 실행을 방지하는 조건부 제어 수단을 구비한 것을 특징으로 한다.

<18> 본 발명에 따른 데이터 처리장치에 있어서, 완전히 조건 명령 세트가 사용되고, 조건부로 각 명령의 전부의 실행을 방지하는 메카니즘도 메모리 중단을 처리하기 위해 사용된다. 이와 같이, 바로 전의 명령에 의해 시작된 메모리 액서스에 관한 메모리 중단 신호들은, 각 명령의 실험 중에 그 명령이 실행되어야 하는지의 여부를 조건 테스트 수단으로 검출하기 때문에, 동시에 수신될 수 있다. 이것에 의해 메모리 중단이 상술한 이전에 제안된 데이터 프로세서에 대해서 허용한 것보다 더 늦은 시간에 각 명령에 대하여 처리될 수 있다.

<19> 바람직한 실시예에 있어서, 상기 장치는 데이터 처리장치의 현재의 처리상태를 나타내는 데이터를 기억하는 1개 또는 그 이상의 처리 플래그를 구비하고; 각 명령은 그 명령이 실행되는 것을 필요로 했던 처리 플래그의 상태를 정의하는 조건 코드를 포함하며; 상기 조건 테스트 수단은 각 명령의 조건 코드에 의해 정의된 처리 플래그의 요구 상태와 처리 플래그의 현재의 상태를 비교하도록 동작할 수 있다. 상기 조건코드는 특정한 처리 플래그가 특정한 논리 상태로 설정되어야 한다는 것을 지정하거나, 또는 현재의 명령이 실행되어야 하는지의 여부에 그 처리 플래그의 상태가 영향을 끼치지 않는다는 것을 지정한다. 극단적인 경우에는, 특정한 명령이 어떠한 처리 플래그의 상태에도 관계없이 실행되어야 한다는 것을 가능 조건코드가 지정할 수도 있다.

<20> 이 처리 플래그는 상기 장치의 처리 상태의 다양한 특징을 지정할 수 있다. 바람직한 실시예에 있어서, 상기 장치는 4개의 플래그, 즉

<21> (i) 상기 장치의 이전의 데이터 처리동작이 네가티브 결과를 발생했는지의 여부를 나타내는 플래그;

<22> (ii) 상기 장치의 이전의 데이터 처리동작이 제로 결과를 발생했는지의 여부를 나타내는 플래그;

<23> (iii) 올림 비트가 상기 장치의 이전의 데이터 처리동작에 의해 설정되었는지의 여부를 나타내는 플래그;

<24> (iv) 연산 오버플로우가 상기 장치의 이전의 데이터 처리 동작 중에 발생했는지의 여부를 플래그를 구비한다.

<25> 바람직하게는 상기 메모리 액서스 수단은 메모리 어드레스를 데이터 메모리에 전송하는 수단과, 데이터를 데이터 메모리에 순차 전송하거나 또는 데이터 메모리로부터 데이터를 수신하는 수단을 구비한다.

<26> 상기 조건부 제어 수단이 메모리 액서스 수단 및 조건 테스트 수단 양쪽에 적절하게 응답할 수 있기 위해, 상기 메모리 액서스 수단은 중단 제어 신호를 발생하여 메모리 액서스가 무효하다는 것을 나타내도록 동작할 수 있고; 상기 조건 테스트 수단은 조건 고장 제어 신호를 발생하여 현재의 명령이 실행되지 않아도 된다는 것을 나타내도록 동작할 수 있으며; 상기 장치는 중단 제어신호와 조건 고장 제어신호를 결합하여, 조건 제어 수단에 공급하기 위한 결합된 제어 신호를 발생하는 수단을 구비하는 것이 바람직하다.

<27> 바람직한 실시예에 있어서, 결합 수단은 논리 OR게이트를 구비한다.

<28> 간편한 실시예에 있어서, 상기 장치의 데이터 처리 동작은 클럭신호에 의해 제어된다.

<29> 제2 관점에서 보면, 본 발명은 상기 정의한 것과 같은 장치를 구비하는 집적 회로를 제공한다.

<30> 제3 관점에서 보면, 본 발명은 연속하는 데이터 처리 명령이 파이프라인 방식으로 실행되는 데이터 처리 방법을 제공하는 데, 상기 방법은 1개 또는 그이상의 명령에 응답하여 데이터 메모리에 액서스하는 단계와; 각 메모리 액서스가 무효한지의 여부를 검출하는 단계와; 각 명령의 실행 중에, 이전에 실행된 명령에 의해 발생된 상기 장치의 처리 상태에 의존하여, 그 명령이 실행되어야 하는지의 여부를 검출하는 단계와; 이전의 명령에 의해 시작된 메모리 액서스가 무효하다는 것 또는 현재의 명령이 실행되지 않아도 된다는 것을 검출한 경우에 현재의 명령의 전부의 실행을 방지하는 단계를 구비한 것을 특징으로 한다.

도면의 간단한 설명

<31> 첨부도면을 참조하여 본 발명에 대해서 설명하고, 도면에서 동일한 부분에는 동일한 부호를 부착한다.

<32> 제1 도는 이전에 제안된 데이터 프로세서에 의한 데이터 기록 동작 중에 중단신호의 타이밍 요구를 나타내는 개략 타이밍도이고,

<33> 제2 도는 본 발명의 실시예에 따른 데이터 처리장치의 개략적인 블록도이며,

<34> 제3 도는 중앙처리장치의 일부의 개략적인 블록도이고,

<35> 제4 도는 제2 도의 데이터 처리장치에 의한 데이터 기록 동작 중에, 중단신호의 타이밍 요구를 나타내는 개략적인 타이밍도이다.

<36> **바람직한 실시예의 설명**

<37> 제2 도는 본 발명의 실시예에 따른 데이터 처리장치의 개략적인 블록도를 나타낸다. 상기 데이터 처리장치는 중앙처리장치(CPU)(100), 메모리 관리부(MMU)(110) 및 랜덤 액서스 메모리(RAM)(120)를 구비한다. 상기 CPU(100)와 MMU(110)는 메모리 어드레스의 교환을 위한 어드레스 버스(130) 및 데이터 교환을 위한 데이터 버스(140)에 의해 연결된다. 중단 제어 라인(150)은 또한 고장 또는 무효의 메모리 액서스를 나타내는 중단신호를 운반하기 위한 MMU(110)에서 CPU(100)로 제공된다.

<38> 클럭신호를 CPU(100)및 MMU(110)에 제공하여 이들 CPU 및 MMU의 각각의 동작을 제어한다. 각 데이터 처리 명령은 (그 명령의 특징에 의존하는) 클럭신호의 특정한 수의 주기로 CPU(100)에 의해 실행되고, MMU(110)에 의해 메모리 액서스가 클럭신호와 동시에 수행된다.

<39> 상기 MMU(100)는 RAM(120)에 기억된 데이터에 액서스하기 위해 클럭신호 및 CPU(110)의 제어 하에 동작한다. 따라서, 상기 MMU는 복수의 어드레스 및 데이터 라인(160)에 의해 RAM에 연결된다.

<40> 제3 도는 중앙처리 장치(100)의 일부의 개략적인 블록도이다. 상기 CPU(100)는 처리 및 메모리 동작이 사실상 연속적으로 수행될 수 있도록 명령 파이프라인을 사용한다. 일반적으로, 1개의 데이터 처리 명령이 실행되고 있는 동안, 그것의 계승자가 디코드되고, 제3 명령이 메모리로부터 인출된다. 이 배치를 3단계 실행 파이프라인이라고 부른다.

<41> 제3 도에 나타낸 CPU(100)의 일부분에 있어서는, 메모리로부터 인출된 후에 디코드된 데이터 처리 명령이 실행을 위해 전달된다. 상기 데이터 처리 명령은 32비트 데이터 워드이고, 그 중 비트 28~31 이 4비트 조건 필드를 형성한다. 나머지 비트(비트 0~27)는 그 명령에 응답하여 실행될 동작을 정의하고, 어떤 경우에는 상기 동작이 실행되기로 되어 있는 오퍼랜드를 정의한다.

<42> 조건 필드(비트 28~31)는 조건필드의 비트와 미리 정의된 16개의 조건코드를 비교하는 조건 시험장치(200)에 전달된다. 16개의 조건 코드는 N, Z, C 및 V 플래그라고 칭하는 4개의 프로세서 플래그(210)의 1개 또는 그 이상의 상태를 정의한다. 이들 플래그는 이전에 실행된 명령에 의해 발생된 CPU(100)의 처리 상태를 나타낸다.

<43> N 플래그는 CPU(100)의 이전의 연산 동작으로부터의 네가티브 결과를 나타내고, Z 플래그는 이전의 CPU동작으로부터의 제로(동일한)결과를 나타내며; C 플래그는 올림 비트가 이전의 CPU동작 중에 설정되었는지의 여부를 나타내고, V 플래그는 이전의 CPU 동작 중에 발생하는 연산 오버플로우를 나타낸다.

<44> 조건필드의 4개의 비트와 N, Z, C 및 V플래그에 적용된 조건과의 관계는 다음의 리스트에 나타나 있다.

<45> 0000 = EQ - Z 세트 (동일)

<46> 0001 = NE - Z 클리어 (동일하지 않음)

<47> 0010 = CS - C 세트 (사인이 없는 더 높거나 같은)

<48> 0011 = CC - C 클리어 (사인이 없는 더 낮은)

<49> 0100 = MI - N 세트 (네가티브)

<50> 0101 = PL - N 클리어 (포지티브 또는 제로)

<51> 0110 = VS - V 세트 (오버플로우)

<52> 0111 = VC - V 클리어 (오버플로우 없음)

<53> 1000 = HI - C 세트 및 Z클리어 (사인이 없는 더 높은)

<54> 1001 = LS - C 클리어 또는 Z세트(사인이 없는 더 낮거나 같은)

<55> 1010 = GE - N 세트 및 V세트, 또는 N클리어 및 V클리어(보다 크거나 같은)

<56> 1011 = LT - N 세트 및 V클리어, 또는 N클러어 및 V세트(보다 작은)

<57> 1100 = GT - Z 클리어, 또는 N세트 및 V세트, 또는 N클리어 및 V클리어(보다 큰)

<58> 1101 = LE - Z 세트, 또는 N세트 및 V클리어, 또는 N클리어 및 V세트(보다 작거나 같은)

<59> 1110 = AL - 올웨이즈(always)

<60> 1111 = NV - 네버(never)

<61> 조건 시험장치(200)는 비트가 현재의 명령의 조건필드에서 설정되는 것에 의존하여, 상기 리스트에 기록된 플래그의 상태를 테스트한다. 이 비교는 현재의 명령의 실행 중에 발생한다. 현재의 명령은 적절한 플래그가 조건 필드에 의해 지정된 상태로 설정된 경우에만 그 실행을 완료하도록 허용된다.

<62> 올웨이즈(AL) 조건이 지정되면, 상기 명령은 플래그에 상관없이 실행될 것이다. 네버(NV) 조건코드가 상기 플래그(210)의 상태에 상관없이 명령의 실행을 방지한다.

<63> 조건 시험장치(200)는 현재의 명령이 완전히 실행되어야 하는지의 여부를 나타내는 출력신호(220)를 발생한다. 출력신호(220)는 OR-게이트(230)를 이용하여, MMU(110)에서 CPU(100)로의 중단 제어라인(150)에 공급된 중단신호와 결합된다. 따라서, 현재의 명령이 완전히 실행되어서는 안 된다는 것을, 또는 중단신호가 MMU(110)에 의해 표명된다는 것을 조건 시험장치(200)가 나타내면, OR-게이트(230)의 출력이 설정된다.

<64> 명령실행부(240)는 실행될 동작을 정의하고, 어떤 경우에는 동작이 실행되기로 되어 있는 오퍼랜드를 정의하는 각 명령의 비트(0~27)를 수신한다. 현재의 명령의 실행 중에(즉, 조건 시험장치(200) 및 OR-게이트(230)의 전파 및 처리 지연 후에), 명령 실행부(240)는 현재의 명령이 완전히 실행되어야 하는지의 여부를 나타내는 OR-게이트(230)의 출력을 수신한다. OR-게이트(230)의 출력이 현재의 명령이 완전히 실행되어서는 안 된다는 것을 나타내면, 어떠한 레지스터의 상태도 변경하지 않고, 또는 상기 장치와 관련된 메모리 위치의 상태를 변경하지 않고 상기 명령이 취소된다.

<65> 조건 시험장치(200) 및 명령실행부(240)는 상술한 이전에 고안된 ARM 6 프로세서의 대응 구성소자와 같은 형상이어도 된다.

<66> 제3 도의 장치를 이용한 결과는 바로 앞의 명령에 의해 시작된 메모리 액서스에 관한 중단신호는 늦은 단계에서 각 명령의 실행 중에 설정될 수 있도록, 중단신호의 엄중한 타이밍이 상당히 늦춰지게 된다는 것이다.

<67> 중단신호가 고장 명령 인출 동작의 결과로서 수신되면, 그 명령은 상기 언급된 3단계의 명령 파이프라인으로 더 늦은 단계에서 폐기된다.

<68> 메모리로부터 판독된 데이터가 적절한 프로세서 레지스터 내에 로드되는 시간을 준비하도록, 비메모리 액서스 (내부)프로세서 사이클은 데이터 판독동작을 따른다. 따라서, 고장 데이터 판독 동작의 결과로서 중단신호가 수신되면, 판독 동작을 따르는 내부 사이클은 (데이터가 상당히 잘못된 것일 수도 있는) MMU(110)에 의해 메모리로부터 공급된 어떤 데이터가 무시되어 예정된 위치(즉, 프로세서 레지스터) 내에 저장되지 않는 시간을 고려한다.

<69> CPU(100)에 의해 명령실행을 제어하는 클럭신호(300)가 어드레스 버스(130) 및 데이터 버스(140)의 상태와 함께 예시되어 있는 제4 도에는 명령실행의 타이밍 관계 및 데이터 기록 동작에 대한 중단신호의 수신에 대해서 예시되어 있다.

<70> 데이터 기록 동작을 시작하기 위해, 어드레스(310)는 CPU(100)에 의해 어드레스 버스 상에 배치된다. 1/2 사이클의 클럭신호(300) 후에, 데이터(320)는 CPU(100)에 의해 데이터 버스(140) 상에 배치된다. 일단 이것이 행해지면, CPU(100)는 클럭신호(300)의 다음의 사이클 동안에 다음 명령을 실행할 수 있다.

<71> 동시에 데이터(320)가 데이터 버스 상에 배치되기 때문에, (제4 도의 330으로 표시된) 중단신호가 수신되면, 또는 조건 시험장치(200)가 (1/2 사이클 후에 데이터가 데이터 버스 상에 배치되어 있는 동안 조건 시험장치(200)에 의해 결정되며, 340으로 표시된)다음 명령이 실행되어서는 안된다는 것을 결정하면, 다음 명령의 실행이 포기된다.

<72> 본 발명의 예시적인 실시예를 참조도면을 참조하여 상세히 설명하였지만, 본 발명은 이들 실시예에 제한되지 않으며, 다양한 변형 및 변경은 첨부된 청구항에 의해 정의한 바와 같이 본 발명의 범위로부터 벗어나지 않고 본 발명이 속하는 기술분야의 당업자에 의해 실행될 수 있다는 것을 알 수 있다.

(57) 청구의 범위

청구항 1

연속하는 데이터 처리명령이 파이프라인 방식으로 실행되는 데이터 처리장치(100, 110, 120)에 있어서,

1개 또는 그 이상의 상기 명령에 응답하여 데이터 메모리를 액서스하고, 각 메모리 액서스가 무효한지의 여부를 검출하는 수단을 구비하는 메모리 액서스 수단(110)과,

이전에 실행된 명령에 의해 발생되고 각 명령의 실행시 사용 가능한 상기 장치의 처리 상태에 응답하여, 그 명령이 실행되어야 하는지의 여부를 검출하는 조건 테스트 수단(200)과,

상기 메모리 액서스 수단(110) 및 상기 조건 테스트 수단(220)에 응답하여, 상기 메모리 액서스

수단(110)이 바로 앞의 명령에 의해 시작된 메모리 액서스가 무효한지의 여부를 검출하면, 또는 상기 조건 테스트 수단(200)이 상기 현재의 명령이 실행되어서는 안 된다는 것을 검출하면, 현재의 명령의 전부의 실행을 방지하는 조건 제어수단(230, 240)을 구비하는 것을 특징으로 하는 데이터 처리장치.

청구항 2

제1 항에 있어서,

상기 장치는 상기 데이터 처리장치의 현재의 처리상태를 나타내는 데이터를 기억하는 1개 또는 그 이상의 처리 플래그(NZCV)를 구비하고,

각 명령은 그 명령이 실행될 필요가 있는 상기 처리 플래그의 상태를 정의하는 조건 코드(비트 28~31)를 포함하며,

상기 조건 테스트 수단(200)은 각 명령의 상기 조건코드에 의해 정의된 처리 플래그의 요구 상태와 상기 처리 플래그의 실제의 상태를 비교하도록 동작할 수 있는 것을 특징으로 하는 데이터 처리장치.

청구항 3

제2 항에 있어서,

(i) 상기 장치의 이전의 데이터 처리동작이 네가티브 결과를 발생했는지의 여부를 나타내는 처리 플래그(N)와,

(ii) 상기 장치의 이전의 데이터 처리 동작이 제로 결과를 발생했는지의 여부를 나타내는 처리 플래그(Z)와,

(iii) 올림 비트가 상기 장치의 이전의 데이터 처리동작에 의해 설정되었는지의 여부를 나타내는 처리 플래그(C)와,

(iv) 연산 오버플로우가 상기 장치의 이전의 데이터 처리동작시 발생했는지의 여부를 각각 나타내는 처리 플래그(V)를 구비하는 것을 특징으로 하는 데이터 처리장치.

청구항 4

제1 항 내지 제3 항 중 어느 한 항에 있어서,

상기 메모리 액서스 수단(110)은

메모리 어드레스를 상기 데이터 메모리(120)로 전송하는 수단과,

데이터를 상기 데이터 메모리로 순차 전송하거나 데이터를 상기 데이터 메모리로부터 수신하는 수단을 구비하는 것을 특징으로 하는 데이터 처리장치.

청구항 5

제4 항에 있어서,

상기 메모리 액서스 수단(100)은 중단 제어신호(150)를 발생하여 메모리 액서스가 무효하다는 것을 나타내도록 동작할 수 있고,

상기 조건 테스트 수단(200)은 조건 고장 제어 신호(220)를 발생하여 상기 현재의 명령이 실행되어서는 안 된다는 것을 나타내도록 동작할 수 있으며,

상기 장치는 상기 중단 제어신호와 상기 조건 고장 제어신호를 결합하여, 상기 조건 제어수단에 공급하기 위한 결합된 제어신호를 발신하는 수단(230)을 구비하는 것을 특징으로 하는 데이터 처리장치.

청구항 6

제5 항에 있어서,

상기 결합 수단(230)은 논리 OR 게이트를 구비하는 것을 특징으로 하는 데이터 처리장치.

청구항 7

제1 항과 제5 항과 제6 항 중 어느 한 항에 있어서,

상기 장치의 데이터 처리 동작은 클럭신호에 의해 제어되는 것을 특징으로 하는 데이터 처리장치.

청구항 8

제1 항과 제5 항과 제6 항 중 어느 한 항에 있어서,

상기 장치의 데이터 처리 동작은 클럭신호에 의해 제어되고,

상기 장치는 집적회로에 포함되는 것을 특징으로 하는 데이터 처리장치.

청구항 9

연속하는 데이터 처리명령이 파이프라인 방식으로 실행되는 데이터 처리방법에 있어서,

1개 또는 그 이상의 상기 명령에 응답하여 데이터 메모리에 액서스하는 단계(110)와,

각 메모리 액서스가 무효한지의 여부를 검출하는 단계(110)와,

각 명령의 실행시, 그 명령이 실행되어야 하는지의 여부를 이전에 실행된 명령에 의해 발생된 상기 장치의 처리 상태에 의존하여 검출하는 단계(200)와,

이전의 명령에 의해 시작된 메모리 액서스가 무효하다는 것 또는 상기 현재의 명령이 실행되어서는 안 된다는 것을 검출한 경우에 현재의 명령의 전부의 실행을 방지하는 단계(230, 240)를 구비하는 것을 특징으로 하는 데이터 처리방법.

요약

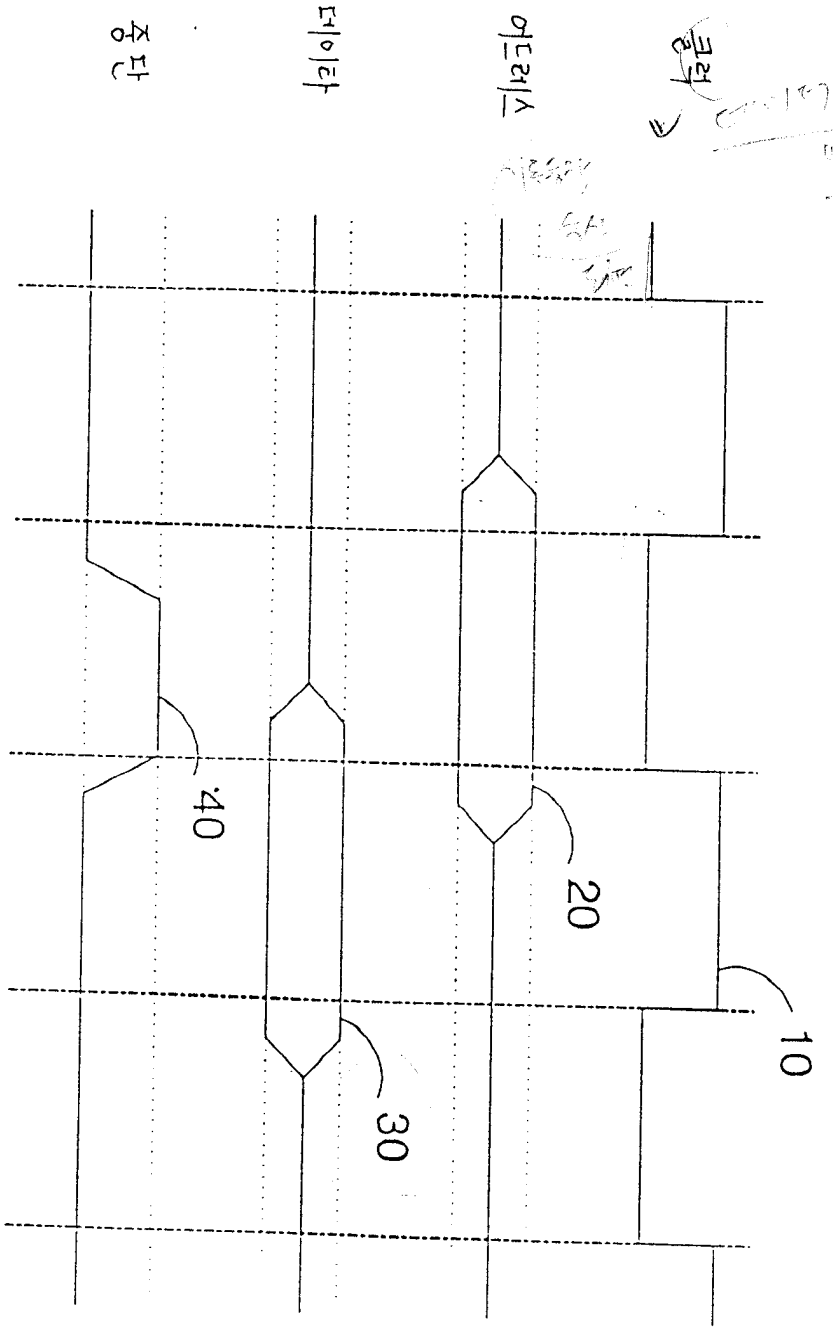
연속하는 데이타 처리 명령이 실행되는 데이타 처리 장치는 1 또는 그이상의 상기 명령에 답해서 데이타 메모리를 액세스하는 메모리 액세스 수단과, 각 메모리 액세스가 무효한지를 검출하는 수단을 구비하는 상기 메모리 액세스 수단과; 각 명령이 실행시 이전에 실행된 명령에 의해 발생되고 사용 가능한 상기 장치의 처리 상태에 응하여, 그 명령이 실행되어야 하는지를 검출하는 조건 테스트 수단과; 상기 메모리 액세스 수단과 상기 조건 테스트 수단에 응하여, 상기 메모리 액세스 수단이 선행의 명령에 의해 시작된 메모리 액세스가 무효한지를 검출하면, 또는 조건 테스트 수단이 커런트 명령을 실행하지 않아도 된다는 것을 검출하면 커런트 명령의 완전한 실행을 방지하는 조건 제어 수단을 구비한다.

대표도

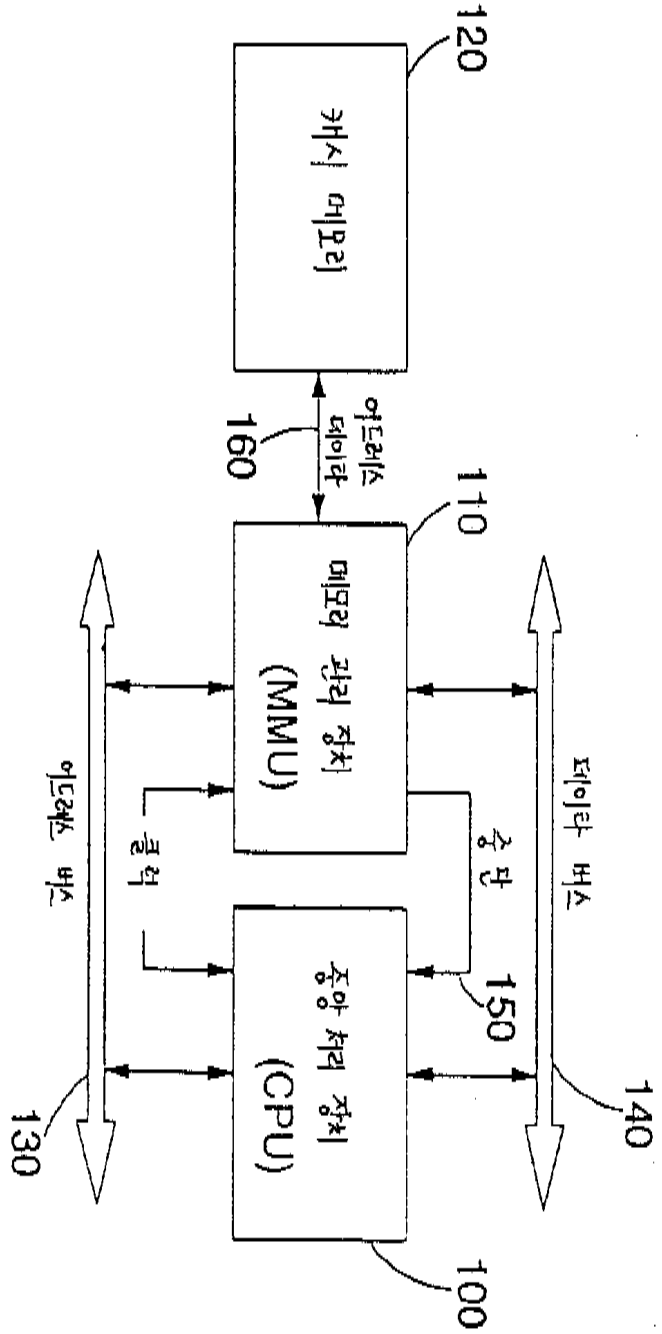
도1

도면

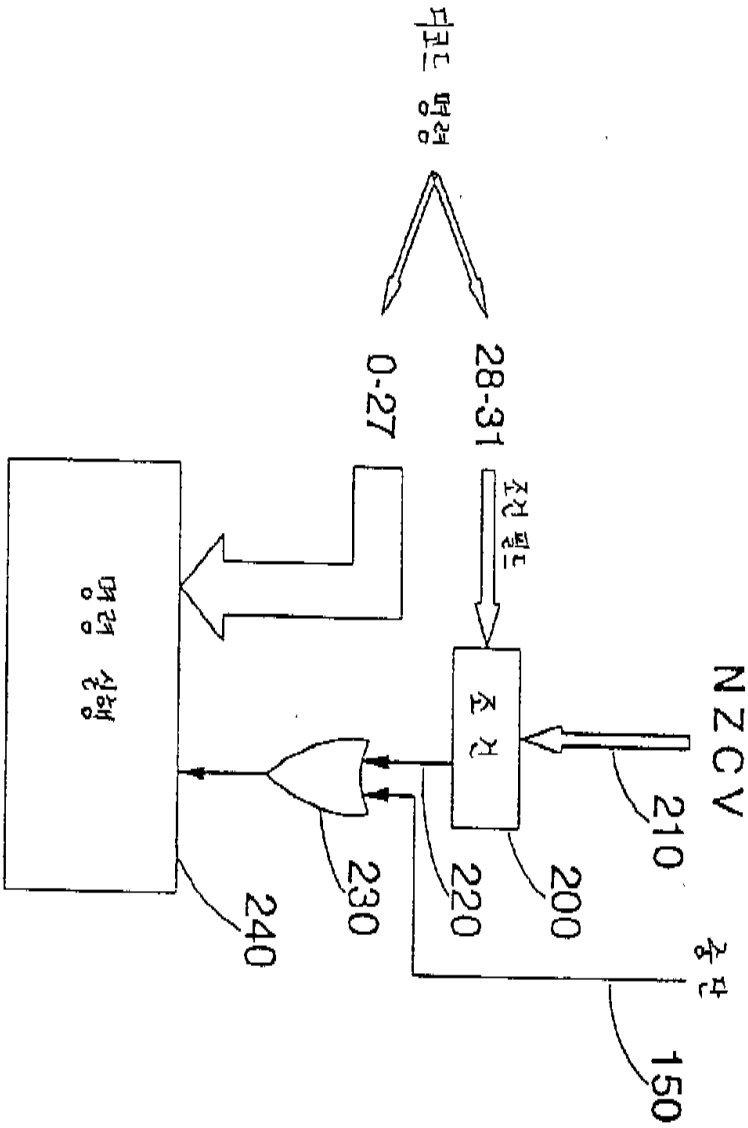
도면1



도면2


120
캐시 메모리
160
어드레스
데이타
110
메모리 관리 장치
(MMU)
어드레스 버스
클럭
중단
150
중앙 처리 장치
(CPU)
데이타 버스
130
140
100

도면3



도면4

