(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2015년05월11일
(11) 등록번호 10-1518440
(24) 등록일자 2015년04월30일

(51) 국제특허분류(Int. Cl.)
G06F 1/32 (2006.01)
(21) 출원번호 10-2013-7007018
(22) 출원일자(국제) 2011년08월25일
심사청구일자 2014년09월19일
(85) 번역문제출일자 2013년03월19일
(65) 공개번호 10-2013-0137613
(43) 공개일자 2013년12월17일
(86) 국제출원번호 PCT/US2011/049171
(87) 국제공개번호 WO 2012/027584
국제공개일자 2012년03월01일
(30) 우선권주장
12/868,996 2010년08월26일 미국(US)
(56) 선행기술조사문헌
US20090235108 A1
US7254721 B1
WO2007103051 A2

(73) 특허권자
어드밴스드 마이크로 디바이시즈, 인코포레이티드
미국 캘리포니아 94088-3453 서니베일 피.오.박스 3453 원 에이엠디 플레이스
(72) 발명자
브래노버 알렉산더
미국 02467 매사추세츠 체스트넛 힐 뉴턴 스트리트 783
스테인먼 마우리스
미국 01752 매사추세츠 말버러 데논코트 스트리트 78
버처 윌리엄 엘.
미국 78751 텍사스 오스틴 에비뉴 에프 3913
(74) 대리인
박장원

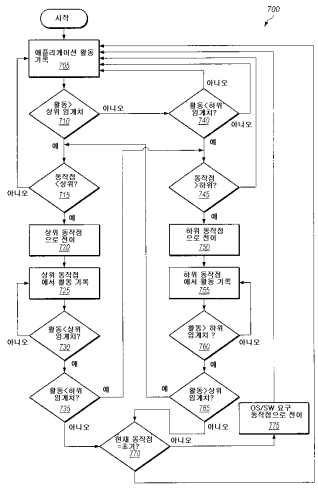
전체 청구항 수 : 총 30 항 심사관 : 김곤희

(54) 발명의 명칭 **프로세싱 노드의 동적 성능 제어**

(57) 요 약

프로세싱 노드의 성능 제어를 위한 장치 및 방법이 개시된다. 한 구체 예에서, 시스템은 프로세싱 노드 및 전력 관리 장치를 포함하며, 전력 관리 장치는 복수의 시간 간격 각각에 대하여, 프로세싱 노드의 활동 수준을 모니터링하고, 주어진 간격에서 활동 수준이 상위 활동 임계치보다 더 큰 경우 프로세싱 노드로 하여금 하나의 연속적 시간 간격 동안 상위 동작점에서 작동하도록 하고, 활동 수준이 하위 활동 임계치 미만인 경우 적어도 하나의 연속적인 시간 간격에서 하위 동작점에서 작동하도록 하거나, 또는 활동 수준이 상위 활동 임계치 미만이고 하위 활동 임계치보다 큰 경우 운영 체제 소프트웨어가 프로세싱 노드로 하여금 복수의 동작점의 하나 이상의 사전정의된 중간 동작점 중 하나에서 작동하도록 구성된다.

대 표 도 - 도8



명 세 서

청구범위

청구항 1

시스템에 있어서,

프로세싱 노드; 및

복수의 제1 시간 간격 각각에 대하여, 상기 프로세싱 노드의 활동 수준을 모니터링하도록 구성된 전력 관리 회로를 포함하여 구성되며,

상기 전력 관리 회로는:

주어진 제1 시간 간격에서의 상기 활동 수준이 상위 활동 임계치보다 큰 경우 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 적어도 하나의 연속적 제1 시간 간격 동안 복수의 동작점들 중 사전정의된 상위 동작점에서 동작하도록 하고;

상기 주어진 제1 시간 간격에서의 상기 활동 수준이 하위 활동 임계치보다 작은 경우 상기 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 상기 적어도 하나의 연속적 제1 시간 간격 동안 상기 복수의 동작점들 중 사전정의된 하위 동작점에서 동작하도록 하고; 그리고

상기 활동 수준이 상기 하위 활동 임계치와 상기 상위 활동 임계치 사이인 경우, 운영 체제 소프트웨어가 상기 프로세싱 노드로 하여금 적어도 하나의 연속적 제1 시간 간격 동안 상기 복수의 동작점들 중 하나 이상의 사전정의된 중간 동작점 중 하나에서 동작하게 하도록 구성된 것을 특징으로 하는, 시스템.

청구항 2

제 1 항에 있어서,

상기 제1 시간 간격 각각의 제1 기간(duration)은 프로세싱 노드의 활동 수준을 모니터링하기 위하여 상기 운영 체제 소프트웨어에 의해 사용됨과 아울러 상기 모니터링된 활동 수준이 상기 상위 활동 임계치보다 작고 상기 하위 활동 임계치보다 클 때 상기 프로세싱 노드의 동작점을 변화시키기 위해 상기 운영 체제 소프트웨어에 의해 또한 사용되는 복수의 제2 시간 간격 각각의 제2 기간보다 짧은 것을 특징으로 하는, 시스템.

청구항 3

제 1 항에 있어서,

상기 사전정의된 상위 동작점은 상기 복수의 동작점들 중 최상위 동작점이며, 상기 사전정의된 하위 동작점은 상기 복수의 동작점들 중 최하위 비-유휴(non-idle) 동작점이며, 상기 하나 이상의 중간 동작점 각각은 상기 사전정의된 상위 동작점보다 하위이고 사전정의된 하위 동작점보다 상위인 것을 특징으로 하는, 시스템.

청구항 4

제 1 항에 있어서,

상기 전력 관리 회로는 각각의 활동 수준을 둘 이상의 프로세싱 노드 각각에 대한 상기 상위 활동 임계치 및 하위 활동 임계치와 비교하도록 더 구성되며, 상기 둘 이상의 프로세싱 노드들은 각각 서로 독립적으로 각각의 동작점에서 동작하도록 구성된 것을 특징으로 하는, 시스템.

청구항 5

제 1 항에 있어서,

상기 전력 관리 회로는 상기 적어도 하나의 프로세싱 노드의 동작점을 변화시킬 때 클럭 신호의 클럭 주파수를 변화시키도록 구성된 것을 특징으로 하는, 시스템.

청구항 6

제 1 항에 있어서,

상기 전력 관리 회로는:

상기 복수의 제1 시간 간격 중 주어진 시간 간격에서 수신된 퇴거 명령의 각각의 표시에 응답하여 카운터를 증가시키고;

상기 복수의 제1 시간 간격 중 상기 주어진 시간 간격에 대한 명령 퇴거의 수가 명령 퇴거의 예상된 수와 같거나 작다는 결정에 응답하여 카운터를 감소시키고;

상기 복수의 제1 시간 간격 중 상기 주어진 시간 간격의 종료시에 상기 카운터에 의해 제공되는 카운트 값을 상기 상위 및 하위 활동 임계치와 비교하도록 구성된 것을 특징으로 하는, 시스템.

청구항 7

제 1 항에 있어서,

상기 전력 관리 회로는:

주어진 제1 시간 간격 동안 상기 적어도 하나의 프로세싱 노드의 활동의 이동 평균을 결정하고; 그리고

상기 활동의 이동 평균을 상기 상위 활동 임계치 및 하위 활동 임계치와 비교하도록 구성된 것을 특징으로 하는, 시스템.

청구항 8

제 1 항에 있어서,

상기 전력 관리 회로는 상기 적어도 하나의 프로세싱 노드에 의해 실행되는 사이클 당 명령의 수를 결정하도록 구성되고, 상기 사이클 당 명령의 수를 상기 상위 활동 임계치 및 하위 활동 임계치와 비교하도록 더 구성된 것을 특징으로 하는, 시스템.

청구항 9

제 1 항에 있어서,

상기 상위 활동 임계치는 제1 임계값 및 제1 히스테리시스 값에 기초하며, 상기 하위 활동 임계치는 제2 임계값 및 제2 히스테리시스 값에 기초하는 것을 특징으로 하는, 시스템.

청구항 10

제 2 항에 있어서,

상기 복수의 제1 간격 각각은 100 마이크로초 이하의 기간을 가지며, 상기 복수의 제2 시간 간격 각각은 30 밀리초 이상의 기간을 갖는 것을 특징으로 하는, 시스템.

청구항 11

제 4 항에 있어서,

최상위 동작점에서 동작할 때의 클럭 주파수는 상기 복수의 동작점들 중 나머지 동작점들 중의 임의의 하나에서 동작할 때의 클럭 주파수보다 크며,

최하위 비-유휴 동작점에서 동작할 때의 클럭 주파수는 상기 복수의 동작점들 중 나머지 동작점들 중의 임의의 하나에서 동작할 때의 클럭 주파수보다 작으며,

상기 최하위 비-유휴 동작점에서의 클럭 주파수는 0보다 큰 것을 특징으로 하는 시스템.

청구항 12

제 4 항에 있어서,

상기 복수의 동작점들 각각은 동작 전압을 더 포함하며,

상기 전력 관리 회로는 상기 프로세싱 노드의 상기 동작점을 변화시킬 때 상기 프로세싱 노드로 제공되는 동작 전압을 변화시키도록 구성된 것을 특징으로 하는 시스템.

청구항 13

제 6 항에 있어서,

상기 전력 관리 회로는, 만일 상기 명령 퇴거의 수가 상기 주어진 제1 시간 간격에서 사전결정된 값을 초과하는 경우, 상기 주어진 제1 시간 간격에서 상기 카운터를 더 증가시키는 것을 억제하도록 구성된 하위 패스 필터(low pass filter)를 포함하는 것을 특징으로 하는, 시스템.

청구항 14

제 6 항에 있어서,

상기 전력 관리 회로는 상기 카운터를 증가시킬 때 제1 가중치 인자를 적용하고 상기 카운터를 감소시킬 때 제2 가중치 인자를 적용하도록 더 구성된 것을 특징으로 하는 시스템.

청구항 15

제 13 항에 있어서,

상기 전력 관리 회로는,

실행된 명령;

퇴거된 명령;

발행된 명령;

파이프라인 스톨;

메모리 액세스 요청;

캐쉬 미스; 및

브랜치 오예측 중 하나 이상을 모니터링하는 것에 기초하여 상기 활동의 이동 평균을 결정하도록 구성된 것을 특징으로 하는, 시스템.

청구항 16

방법에 있어서,

복수의 제1 시간 간격 각각에 대하여, 프로세싱 노드의 활동 수준을 결정하는 단계;

상기 활동 수준이 상위 활동 임계치보다 큰 경우 전력 관리 회로가 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 적어도 하나의 연속적 제1 시간 간격에서 복수의 동작점들 중 사전정의된 상위 동작점에서 동작하도록 하는 단계; 및

상기 활동 수준이 하위 활동 임계치보다 작은 경우 상기 전력 관리 회로가 상기 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 적어도 하나의 연속적 제1 시간 간격에서 복수의 동작점들 중 사전정의된 하위 동작점에서 동작하도록 하는 단계; 및

상기 활동 수준이 상기 하위 활동 임계치와 상위 활동 임계치 사이인 경우 운영 체제 소프트웨어가 상기 프로세싱 노드로 하여금 상기 복수의 동작점들 중 하나 이상의 중간 동작점들 중 하나에서 동작하도록 하는 단계를 포함하는, 방법.

청구항 17

제 16 항에 있어서,

상기 운영 체제 소프트웨어가 상기 복수의 제2 시간 간격 동안 상기 활동 수준을 모니터링하는 단계를 더 포함

하며, 상기 제1 시간 간격 각각의 제1 기간은 상기 제2 시간 간격 각각의 제2 기간보다 짧은 것을 특징으로 하는, 방법.

청구항 18

제 16 항에 있어서,

상기 사전정의된 상위 동작점은 상기 복수의 동작점들 중 최상위 동작점이고, 상기 사전정의된 하위 동작점은 상기 복수의 동작점들 중 최하위 비-유휴 동작점이며, 상기 하나 이상의 중간 동작점들 각각은 상기 사전정의된 상위 동작점보다 하위이고 상기 사전정의된 하위 동작점보다 상위인 것을 특징으로 하는, 방법.

청구항 19

제 16 항에 있어서,

상기 복수의 제1 시간 간격 각각에 대하여, 둘 이상의 프로세싱 노드에 대한 각각의 활동 수준을 상기 상위 활동 임계치 및 하위 활동 임계치와 비교하는 단계;

상기 각각의 활동 수준을 비교하는 단계에 기초하여 상기 둘 이상의 프로세싱 노드들 중 제1 프로세싱 노드의 동작점을 설정하는 단계; 및

상기 각각의 활동 수준을 비교하는 단계에 기초하여 상기 둘 이상의 프로세싱 노드 중 제2 프로세싱 노드의 동작점을 설정하는 단계를 더 포함하며,

상기 둘 이상의 프로세싱 노드 중 상기 제2 프로세싱 노드의 동작점이, 상기 둘 이상의 프로세싱 노드 중 상기 제1 프로세싱 노드의 동작점을 설정하는 것과는 독립적으로 설정되는 것을 특징으로 하는, 방법.

청구항 20

제 16 항에 있어서,

상기 프로세싱 노드의 동작점을 변화시키는 것은 상기 프로세싱 노드에 제공된 클럭 신호의 주파수를 변화시키는 것을 포함하는 것을 특징으로 하는, 방법.

청구항 21

제 16 항에 있어서,

주어진 제1 시간 간격에서 수신된 퇴거 명령의 각각의 표시에 응답하여 카운트를 증가시키는 단계;

상기 주어진 제1 시간 간격에서 일어나는 명령 퇴거의 수가 예상된 명령 퇴거의 수와 같거나 작다는 결정에 응답하여 카운터를 감소시키는 단계; 및

상기 주어진 제1 시간 간격의 종료시에 상기 카운터에 의해 제공되는 카운트 값을 상기 상위 및 하위 활동 임계치와 비교하는 단계를 더 포함하는 것을 특징으로 하는, 방법.

청구항 22

제 16 항에 있어서,

적어도 하나의 제1 시간 간격 동안 상기 적어도 하나의 프로세싱 노드의 활동의 이동 평균을 결정하는 단계; 및

상기 활동의 이동 평균을 상기 상위 활동 임계치 및 하위 활동 임계치와 비교하는 단계를 더 포함하는 것을 특징으로 하는, 방법.

청구항 23

제 16 항에 있어서,

상기 적어도 하나의 프로세싱 노드에 의해 실행되는 사이클당 명령의 수를 결정하는 단계, 및 상기 사이클당 명령의 수를 상기 상위 활동 임계치 및 하위 활동 임계치와 비교하는 단계를 더 포함하는 것을 특징으로 하는, 방법.

청구항 24

제 20 항에 있어서,

상기 프로세싱 노드의 동작점을 변화시키는 것은 상기 프로세싱 노드에 공급되는 동작 전압을 변화시키는 것을 포함하는 것을 특징으로 하는, 방법.

청구항 25

제 22 항에 있어서,

실행된 명령;

퇴거된 명령;

발행된 명령;

파이프라인 스톨;

메모리 액세스 요청;

캐쉬 미스; 및

브랜치 오예측 중 하나 이상에 기초하여 이동 평균을 결정하는 단계를 더 포함하는 것을 특징으로 하는, 방법.

청구항 26

컴퓨터 시스템에서 실행가능한 프로그램에 의해 동작되는 데이터 구조가 저장된 컴퓨터 판독가능 매체에 있어서,

상기 프로그램은 상기 데이터 구조에 의해 기술되는 회로(circuitry)를 포함하는 집적 회로(integrated circuit)를 제조하기 위한 공정의 일부를 수행하기 위하여 상기 데이터 구조상에서 동작하며, 상기 데이터 구조 내에 기술된 상기 회로는 복수의 제1 시간 간격 각각에 대하여, 프로세싱 노드의 활동 수준을 모니터링하도록 구성되는 전력 관리 회로를 포함하며,

상기 전력 관리 회로는:

주어진 제1 시간 간격에서 상기 활동 수준이 상위 활동 임계치보다 더 큰 경우 상기 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 적어도 하나의 연속적 제1 시간 간격 동안 복수의 동작점들 중 사전정의된 상위 동작점에서 동작하도록 하고;

상기 주어진 제1 시간 간격에서 상기 활동 수준이 하위 활동 임계치보다 작은 경우 상기 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 상기 적어도 하나의 연속적 제1 시간 간격에 대한 상기 복수의 동작점들 중 사전정의된 하위 동작점에서 동작하도록 하고; 그리고

상기 활동 수준이 상기 하위 활동 임계치와 상위 활동 임계치 사이인 경우, 상기 운영 체제 소프트웨어가 프로세싱 노드로 하여금 상기 적어도 하나의 연속적 제1 시간 간격 동안 상기 복수의 동작점들 중 하나 이상의 사전정의된 중간 동작점 중 하나에서 동작하게 하도록 구성된 것을 특징으로 하는, 컴퓨터 판독가능 매체.

청구항 27

제 26 항에 있어서,

상기 데이터 구조는

HDL (high-level design language, 상위-수준 설계 언어) 데이터;

RTL (register transfer level, 레지스터 전송 수준) 데이터; 및

그래픽 데이터 시스템(Graphic Data System, GDS) II 데이터 중 하나 이상의 데이터를 포함하는 것을 특징으로 하는, 컴퓨터 판독가능 매체.

청구항 28

제 26 항에 있어서,

상기 제1 시간 간격 각각의 제1 기간은 프로세싱 노드의 활동 수준을 모니터링하기 위하여 상기 운영 체제 소프트웨어에 의해 사용됨과 아울러 상기 모니터링된 활동 수준이 상기 상위 활동 임계치보다 작고 상기 하위 활동 임계치보다 클 때 상기 프로세싱 노드의 동작점을 변화시키기 위해 상기 운영 체제 소프트웨어에 의해 또한 사용되는 복수의 제2 시간 간격 각각의 제2 기간보다 짧으며,

상기 전력 관리 회로는

실행된 명령;

퇴거된 명령;

발행된 명령;

파이프라인 스톨;

메모리 액세스 요청;

캐쉬 미스; 및

브랜치 오예측 중 하나 이상에 기초하여 상기 활동 수준을 결정하도록 구성된 것을 특징으로 하는, 컴퓨터 판독가능 매체.

청구항 29

제 26 항에 있어서,

상기 데이터 구조에 기술된 상기 전력 관리 회로는

상기 적어도 하나의 프로세싱 노드에 의해 수신된 클럭 신호의 주파수; 및

상기 적어도 하나의 프로세싱 노드에 공급된 동작 전압 중 하나 이상의 변화를 야기시킴으로써 상기 프로세싱 노드의 동작점의 변화를 야기하도록 구성된 것을 특징으로 하는, 컴퓨터 판독가능 매체.

청구항 30

시스템에 있어서,

프로세싱 노드; 및

복수의 제1 시간 간격 각각에 대하여, 상기 프로세싱 노드의 활동 수준을 모니터링하는 전력 관리 회로를 포함하며,

상기 전력 관리 회로는:

주어진 제1 시간 간격에서 상기 활동 수준이 상위 활동 임계치보다 더 큰 경우 상기 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 적어도 하나의 연속적 제1 시간 간격 동안 복수의 동작점들 중 사전정의된 상위 동작점에서 동작하도록 하고;

상기 주어진 제1 시간 간격에서 상기 활동 수준이 하위 활동 임계치보다 작은 경우 상기 프로세싱 노드로 하여금 운영 체제 소프트웨어와는 독립적으로 상기 적어도 하나의 연속적 제1 시간 간격에 대한 상기 복수의 동작점들 중 하위 동작점에서 동작하도록 하며,

상기 제1 시간 간격 각각의 제1 기간은 상기 프로세싱 노드의 활동 수준을 모니터링하기 위하여 상기 운영 체제 소프트웨어에 의해 사용됨과 아울러 상기 모니터링된 활동 수준에 기초하여 상기 프로세싱 노드의 동작점을 변화시키기 위해 운영 체제 소프트웨어에 의해 또한 사용되는 복수의 제2 시간 간격 각각의 제2 기간보다 짧은 것을 특징으로 하는, 시스템.

발명의 설명

기 술 분 야

[0001] 1. 발명의 분야

[0002] 본 발명은 프로세서에 관한 것이며, 더욱 상세하게는, 프로세서에서의 평형 성능 및 전력 소모에 관한 것이다.

배 경 기 술

[0003] 2. 관련 기술의 설명

[0004] 컴퓨터 또는 또 다른 프로세서-기반 시스템의 설계 동안, 많은 설계 요인이 고려되어야 한다. 성공적인 설계는 전력 소모, 성능, 열 출력, 등의 사이의 일부 균형(tradeoff)을 요구할 수 있다. 예를 들어, 고성능을 강조하는 컴퓨터 시스템의 설계는 더 큰 전력 소모 및 열 출력을 가능하게 할 수 있다. 이와 반대로, 가끔 배터리에 의해 전력공급되는 휴대용 컴퓨터 시스템의 설계는 일부 성능을 희생하면서 전력 소모를 감소시키는 것을 강조할 수 있다.

[0005] 일부 컴퓨터 시스템에서, 동작점(operating point)(예컨대, 클럭 주파수 및 동작 전압)이 워크로드에 따라 조절될 수 있다. 이러한 시스템에서, 컴퓨팅-집중적(compute-bounded) 워크로드(즉, 프로세서 집약적(processor intensive)인 것)는 프로세싱 노드가 상위 동작점에서 작동하도록 할 수 있으며, 반면 메모리-집중적(memory-bounded) 워크로드(즉, 큰 대기시간(latency)을 포함할 수 있는 다수의 메모리 액세스를 포함하는 것)는 프로세싱 노드가 하위 동작점에서 작동하도록 할 수 있다. 예를 들어, 일부 컴퓨터 시스템에서, 활성 상태의 시간의 양 대 동작 시간(operational time)의 전체 양(즉, 활성 상태 및 및 유휴 상태)의 비율이 주어진 간격에 대하여 운영 체제 소프트웨어에 의해 계산될 수 있다. 상기 비율이 특정 임계치를 초과하는 경우, 프로세싱 노드는 상위 동작점에서 작동될 수 있다. 그렇지 않은 경우, 작동은 하위 동작점에서 일어날 수 있다. 상기 계산을 수행하기 위한 전형적인 간격은 30-100 밀리초 범위일 수 있으며, 일부 고급 시스템은 상기 계산을 매 10 밀리초마다 만큼 빈번하게 수행한다.

발명의 내용

[0006] **본 발명의 구체 예의 개요**

[0007] 프로세싱 노드의 동적 성능 제어를 위한 장치 및 방법이 개시된다. 한 구체 예에서, 시스템은 프로세싱 노드 및 전력 관리 장치를 포함한다. 전력 관리 장치는 복수의 제1 시간 간격 각각에 대하여, 상기 프로세싱 노드의 활동 수준을 모니터링하도록 구성될 수 있다. 전력 관리는, 주어진 제1 시간 간격에서 활동 수준이 상위 활동 임계치보다 큰 경우 프로세싱 노드가 적어도 하나의 연속적 제1 시간 간격 동안 복수의 동작점 중 사전정의된 상위 동작점에서 작동하도록 구성될 수 있으며, 그리고 상기 주어진 제1 시간 간격에서 상기 활동 수준이 하위 활동 임계치보다 작은 경우 프로세싱 노드가 상기 적어도 하나의 연속적 제1 시간 간격에 대한 상기 복수의 동작점 중 사전정의된 하위 동작점에서 작동하도록 구성될 수 있다. 전력 관리 장치는, 적어도 하나의 연속적 제1 시간 간격에 대한 활동 수준이 상기 주어진 제1 시간 간격에서 상위 활동 임계치보다 작고 하위 활동 임계치보다 큰 경우, 운영 체제 소프트웨어가 프로세싱 노드를 복수의 동작점의 하나 이상의 사전정의된 중간 동작점 중 하나에서 작동하게 하도록 구성될 수 있다.

[0008] 한 구체 예에서, 방법은, 복수의 제1 시간 간격 각각에 대하여, 프로세싱 노드의 활동 수준을 결정하는 단계를 포함한다. 본 방법은 활동 수준이 상위 활동 임계치를 초과하는 경우 전력 관리 장치가 프로세싱 노드로 하여금 적어도 하나의 연속적 제1 시간 간격에서 복수의 동작점 중 사전정의된 상위 동작점에서 작동하도록 하는 단계, 및 활동 수준이 하위 활동 임계치 미만인 경우 전력 관리 장치가 프로세싱 노드로 하여금 적어도 하나의 연속적 제1 시간 간격에서 복수의 동작점 중 사전정의된 하위 동작점에서 작동하도록 하는 단계를 더욱 포함할 수 있다. 본 방법은 활동 수준이 상위 활동 임계치 미만이고 하위 활동 임계치보다 큰 경우 운영 체제 소프트웨어가 프로세싱 노드로 하여금 복수의 동작점의 하나 이상의 중간 동작점 중 하나에서 작동하도록 하는 단계를 더욱 포함할 수 있다.

도면의 간단한 설명

[0009] 본 발명의 또 다른 양상이 이하의 상세한 설명을 읽고 첨부된 도면를 참조할 때 자명해 질 것이며, 여기서:

도 1은 칩 상의 집적 회로(IC) 시스템(SOC)의 하나의 구체 예의 블록 다이어그램이며;

도 2는 프로세싱 노드의 하나의 구체 예의 블록 다이어그램이며;

도 3은 전력 관리 장치의 하나의 구체 예의 작동을 나타내는 다이어그램이며;

도 4는 전력 관리 장치의 하나의 구체 예의 블록 다이어그램이며;

도 5는 활동 모니터의 하나의 구체 예를 나타내는 블록 다이어그램이며;

도 6은 CIPS(committed instructions per second, 초당 수행되는 명령) 장치의 하나의 구체 예를 나타내는 블록 다이어그램이며;

도 7은 IPC (instructions per cycle, 사이클 당 명령) 장치의 하나의 구체 예의 블록 다이어그램이며;

도 8은 전력 관리 장치 작동 방법의 하나의 구체 예의 흐름도이며;

도 9는 다수의 시간 간격 동안 방법의 하나의 구체 예의 작동을 나타내는 다이어그램이며; 그리고

도 10은 전력 관리 장치를 포함하는 회로의 구체 예를 묘사하는 데이터 구조를 포함하는 컴퓨터 판독가능 매체의 하나의 구체 예의 블록 다이어그램이다.

본 발명이 여러 변형 및 대안적인 형태이기 쉬운 한편, 이들의 특정 구체 예가 도면에서 예로서 제시되며 본 명세서에서 상세하게 설명될 것이다. 그렇지만, 도면 및 이에 대한 설명은 기재된 특정한 형태에 본 발명을 제한하고자 하는 의도가 아니며, 이와 반대로, 본 발명은 첨부된 청구항에 의해 정의되는 본 발명의 사상 및 범위에 속하는 모든 변형, 균등물, 및 대안을 포함하는 것으로 의도됨이 이해되어야 한다.

발명을 실시하기 위한 구체적인 내용

[0010] 본 발명은 프로세서-기반 시스템의 와트 당 성능(performance-per-watt)을 개선하기 위한 방법 및 장치에 관한 것이다. 더욱 상세하게는, 본 명세서에 기재된 방법 및 장치는 애플리케이션 활성(예컨대, 프로세싱 워크로드)에 기초하여 하나 이상의 프로세싱 노드의 동작점(예컨대, 단일 또는 다중-코어 마이크로프로세서의 프로세서 코어, 개별 자립형(stand-alone) 마이크로프로세서, 등)을 조절하기 위하여 사용될 수 있다. 여러 구체 예에서, 프로세싱 노드의 동작점은 자신에게 제공되는 클록 신호의 주파수를 포함할 수 있다. 동작점은 프로세싱 노드에게 제공되는 동작 전압(예컨대, 공급 전압)을 더욱 포함할 수 있다. 따라서 최상위 동작점은 프로세싱 노드에 활용가능한 최상위 클럭 주파수를 갖는 동작점으로서 정의될 수 있으며, 또한 프로세싱 노드에 활용가능한 최상위 동작 전압을 갖는 동작점으로서 정의될 수 있다. 이와 반대로, 최하위 동작점은 프로세싱 노드에 활용가능한 최하위 동작(예컨대, 비-제로(non-zero)) 클럭 주파수를 갖는 동작점으로서 정의될 수 있으며, 또한 활용가능한 최하위 비-제로 동작 전압을 갖는 동작점으로서 정의될 수 있다. 중간 동작점은 클럭 주파수 및 동작 전압 중 적어도 하나가 다른 방식으로 최상위 및 최하위 동작점을 정의하기 위하여 사용될 수 있는 값들 사이의 각각의 값으로 설정되는 동작점으로 정의될 수 있다.

[0011] 하나 이상의 프로세싱 노드 각각에 대한 동작점은 하나 이상의 프로세싱 노드 중 적어도 하나에서 실행되는 운영 체제(OS) 소프트웨어에 의하거나 또는 전력 관리 장치에 의하여 설정될 수 있다. 각각의 프로세싱 노드에 대하여, 전력 관리 장치는 다수의 제1시간 간격에 걸쳐 그 활동 수준을 모니터링할 수 있으며, 모니터링된 활동 수준을 상위 및 하위 활동 임계치와 비교할 수 있다. 전력 관리 장치는, 활동 수준이 상위 활동 임계치를 초과하는 경우, 프로세싱 노드가 복수의 동작점 중 상위 동작점(예컨대, 최상위)에서 작동하도록 할 수 있다. 전력 관리 장치는, 활동 수준이 하위 활동 임계치 미만인 경우, 프로세싱 노드가 복수의 동작점 중 하위 동작점에서 작동하도록 할 수 있다. 활동 수준이 하위 임계치보다 크고 상위 임계치 미만인 경우, 전력 관리 장치는 운영 체제 소프트웨어가 프로세싱 노드에 대한 하나 이상의 중간 동작점 중 하나를 선택하도록 할 수 있다. 운영 체제 소프트웨어는 제2 시간 간격 기간에 걸쳐 활동 수준을 모니터링 할 수 있으며, 이러한 기간은 제1 시간 간격 각각보다 더 크다. 한 구체 예에서, 제1 시간 간격 각각의 기간은 100 마이크로초 이하인 반면, 제2 시간 간격 각각의 기간은 30 밀리초 이상이다. 따라서, 전력 관리 장치는 운영 체제 소프트웨어보다 더 빨리 동작점에 대한 변화에 영향을 미칠 수 있으며, 이에 따라 전력 소모 및 이에 따른 프로세서의 와트 당 성능의 더욱 정교한 제어(fine-grained control)를 가능하게 한다. 전력 관리 장치는, 모니터링된 활동 수준이 상위 활동 임계치를 만족하거나 초과하는 경우 또는 하위 활동 임계치 이하인 경우, 운영 체제에 의해 형성된 동작점 선택을 언제든지 무시할 수 있음에 주목하라.

[0012] 본 명세서에 기재된 전력 관리 장치는 프로세서를 갖는 하드웨어 장치로서 구현될 수 있으며, 이러한 구체 예는 검출된 활동 수준에 기초하여 동작점을 선택하고 설정하는데 있어서 OS 소프트웨어와 독립적으로 작동할 수 있다. 일부 구체 예에서, 전력 관리 장치의 기능은 OS와 독립하여 작동하는 펌웨어 또는 또 다른 소프트웨어로서 구현될 수 있다. 구체적인 구현과 무관하게, 전력 관리 장치는 OS 단독에 의해 제어되는 구체 예를 동반할 수 있는 오버헤드(예컨대, 인터럽트(interrupt), 등) 없이 본 명세서에 기재된 기능을 수행할 수 있다. 활동이 전

력 관리 장치의 다양한 구체 예에 대하여 모니터링 되는 시간 간격은, OS가 동작점을 설정하기 위하여 사용되는 활동 모니터링을 수행할 수 있는 시간 간격보다 상당히 더 작다(예컨대, 10의 1승 이상).

[0013] 한 구체 예에서, 프로세싱 노드의 동작점은 그 안의 애플리케이션 활성이 컴퓨터-집중적인지, 메모리-집중적인지, 또는 이러한 두 포인트 사이의 어느 곳인지 여부에 의존하여 조절될 수 있다. 컴퓨팅-집중적 워크로드는 주메모리에 대한 드문(존재하는 경우) 액세스를 갖는, 전산 집중적(computationally intensive)인 프로세싱 워크로드로서 정의될 수 있다. 가능한 한 짧은 시간 내 컴퓨팅-집중적 워크로드의 완료는 워크로드를 실행시키는 프로세싱 노드가 최상위 활용가능 클럭 주파수에서 작동하는 한편 사이클 당 실행되는 명령의 수를 최대화시키는 것을 요구할 수 있다. 따라서, 본 명세서에 기재된 방법 및 장치는 언제 컴퓨팅-집중적 워크로드가 실행되는지를 결정하도록, 그리고 또한 동작점(클럭 주파수 및/또는 동작 전압)을 이에 대응하는 고성능 상태까지 증가시키도록 구현될 수 있다. 한 구체 예에서, 본 방법 및 장치는, 프로세싱 노드가 컴퓨팅-집중적 워크로드의 검출에 응답하여 해당 특정 노드에 대하여 활용가능한 최고성능 상태에 대응하는 동작점에서 작동하도록 할 수 있다.

[0014] 프로세싱 워크로드가 메모리-집중적인 경우, 프로세싱 노드는 주메모리의 빈번한 액세스(frequent access)를 수행할 수 있다. 주메모리 액세스와 관련된 대기시간이 프로세서 사이클 시간보다 수십 배 더 클 수 있기 때문에, 메모리-집중적 워크로드는 실행될 때 프로세싱 노드의 작동 주파수(예컨대, 클럭 주파수)에 대해 훨씬 덜 민감할 수 있다. 더욱 특히, 메모리 액세스는 프로세서가 스톨(stall)하도록 할 수 있으며, 이러한 스콜의 기간은 메모리 액세스 대기시간의 함수이다. 메모리 액세스와 관련된 대기시간은 메모리 버스 클럭 주파수의 함수이며, 이는 전형적으로 코어 클럭 주파수보다 훨씬 더 작다. 따라서, 코어 클럭 주파수의 증가는 전형적으로 메모리-집중적 워크로드의 프로세싱에서 대응하는 성능 증가를 야기하지 않는다. 더욱이, 메모리-집중적 워크로드를 프로세싱할 때 코어 클럭 주파수를 감소시키는 것은 전형적으로 대응하는 성능 손실을 야기하지 않는데, 왜냐하면 메모리 액세스 대기시간은 통상 이러한 워크로드가 실행될 수 있는 속도를 결정하는 제한 요인(limiting factor)이기 때문이다. 따라서, 본 명세서에 기재된 방법 및 장치는 언제 메모리-집중적 워크로드가 실행되는지를 결정하도록, 그리고 또한 동작점(예컨대, 클럭 주파수 및/또는 동작 전압)을 이에 대응하는 저성능 상태까지 감소시키도록 구현될 수 있다. 메모리-집중적 워크로드를 실행할 때 동작점을 저성능 상태로 감소시키는 것은 성능에 불리한 영향을 주지 않으면서 전력 절약을 유발할 수 있다. 한 구체 예에서, 본 방법 및 장치는, 프로세싱 노드가 메모리-집중적 워크로드의 검출에 응답하여 최하위 비-유휴 성능 상태에 대응하는 동작점에서 작동하도록 할 수 있다. 최하위 비-유휴 동작점은 본 명세서에서 비-제로(non-zero) 주파수에서 클럭 신호 및 전력을 수신하는 동작점으로 정의될 수 있다.

[0015] 주어진 프로세싱 노드에 대한 컴퓨팅-집중적 및 메모리-집중적 워크로드의 검출은 해당 노드의 활동 수준을 상위 임계치 및 하위 임계치와 비교하는 것을 포함할 수 있다. 활동 수준이 상위 임계치를 초과하는 경우, 프로세싱 노드는 고성능 상태에 대응하는 동작점에서 작동될 수 있다. 활동 수준이 하위 임계치 미만인 경우, 프로세싱 노드는 저성능 상태에 대응하는 동작점에서 작동될 수 있다. 활동 수준이 상위 임계치 미만 그러나 하위 임계치 초과인 경우, 프로세싱 노드는 하나 이상의 중간 동작점 중 하나에서 작동될 수 있다. 한 구체 예에서, 중간 동작점은 소프트웨어(예컨대, 운영 체제 소프트웨어)에 의해 선택될 수 있다.

[0016] 각각의 프로세싱 노드에 대한 활동 수준의 검출 및 비교는 다수의 연속적 제1 시간 간격 각각에 대하여 전력 관리 장치에 의하여 수행될 수 있다. 전술한 바와 같이, 간격 기간은 대략 마이크로초일 수 있다. 한 구체 예에서, 간격의 최대 기간은, 10 마이크로초만큼의 작은 간격을 고려했을 때, 100 마이크로초일 수 있다. 제1 간격의 기간이 10 마이크로초 미만인 구체 예가 또한 고려된다. 따라서, 본 명세서에 기재된 방법 및 장치는 운영 체제 소프트웨어에 의해 제공되는 것에 비하여 정교한(fine-grained) 동작점 제어를 가능하게 할 수 있으며, 여기서 동작점을 모니터링하고, 비교하고, 그리고 설정하기 위한 시간 간격은 30 내지 100 밀리초 범위일 수 있다. 더욱이, 활동 수준 및 동작점의 결정은 운영 체제 소프트웨어에 의해 요구될 수 있는 인터럽트 또는 또 다른 오버헤드를 요구하지 않고 수행될 수 있다. 따라서, 소비되는 전력의 와트 당 성능이 최적화될 수 있다.

[0017] **<u>전력 관리 장치를 구비한 프로세서:</u>**

[0018] 도 1은 메모리에 연결된 집적 회로(IC)의 하나의 구체 예의 블록 다이어그램이다. IC(2) 및 메모리(6)는, 디스플레이(3) 및 디스플레이 메모리(300)와 함께, 본 실시예에서 컴퓨터 시스템(10)의 적어도 일부분을 형성한다. 제시된 구체 예에서, IC(2)는 다수의 프로세싱 노드(11)를 갖는 프로세서이다. 프로세싱 노드(11)는 본 특정 실시예에서 프로세서 코어이며, 따라서 또한 코어 #1, 코어 #2, 등으로 표시된다. 본 명세서에 기재될 방법론은 예컨대 별도의 고유 IC 다이 상에서 다중 프로세서(이는 단일-코어 또는 멀티-코어 프로세서일 수 있음)를 실행

하는 멀티-프로세서 컴퓨터 시스템과 같은, 또 다른 배치에 적용될 수 있다는 점에 주목하라. 더욱이, 단지 단일 프로세싱 노드(11)만을 갖는 배치가 또한 가능하며 고려된다.

[0019] 각각의 프로세싱 노드(11)는 제시된 구체 예에서 노스 브리지(north bridge)(12)에 연결된다. 노스 브리지(12)는 메모리에 대한 인터페이스 및 다양한 주변기기에 대한 인터페이스를 비롯하여, 프로세싱 노드(11) 각각에 대한 광범위한 인터페이스 기능을 제공할 수 있다. 더욱이, 노스 브리지(12)는 전력 관리 장치(20)를 포함하는데 이는 각각의 활동 수준 또는 워크로드에 기초하여 프로세싱 노드(11) 각각의 전력 소모를 관리하도록 구성되며, 이하에서 더욱 상세하게 기재할 것이다. 더욱이, 멀티-코어(또는 멀티-프로세서) 구체 예에서, 전력 관리 장치(20)는 서로 독립적으로 개별 프로세싱 노드(11)의 동작점을 설정할 수 있다. 따라서, 제1 프로세싱 노드(11)가 제1 동작점에서 작동할 수 있는 한편, 제2 프로세싱 노드(11)는 제1 동작점과 다른 제2 동작점에서 작동할 수 있다.

[0020] 여러 구체 예에서, 프로세싱 노드(11)의 수는 1만큼 적을 수 있거나, 또는 IC 다이 상에서의 실행을 위하여 가능한 한 많을 수 있다. 멀티-코어 구체 예에서, 프로세싱 노드(11)는 서로 동일할 수 있거나(즉, 동종 멀티-코어), 또는 하나 이상의 프로세싱 노드(11)는 서로 다를 수 있다(즉, 이종 멀티-코어). 프로세싱 노드(11)는 각각 하나 이상의 실행 유닛, 캐쉬 메모리, 스케줄러, 분기 예측 회로(branch prediction circuit), 등을 포함할 수 있다(예시적인 프로세싱 노드가 도 2를 참조하여 이하에서 논의될 것이다). 더욱이, 각각의 프로세싱 노드(11)는 메모리(6)로의 액세스에 대한 요청을 어서트(assert)하도록 구성될 수 있으며, 이는 컴퓨터 시스템(10)에 대한 주메모리로서 기능을 할 수 있다. 이러한 요청은 판독 요청 및/또는 기록 요청을 포함할 수 있으며, 초기에 노스 브리지(12)에 의해 각각의 프로세싱 노드(11) 로부터 수신될 수 있다. 메모리(6)로의 액세스에 대한 요청은 제시된 구체 예에서 메모리 제어기(18)를 통하여 라우팅(route)될 수 있다.

[0021] I/O 인터페이스(13)가 또한 제시된 구체 예에서 노스 브리지(12)에 연결된다. I/O 인터페이스(13)는 컴퓨터 시스템(10)에서 사우스 브리지 디바이스로서 기능을 할 수 있다. 다수의 서로 다른 종류의 주변 버스가 I/O 인터페이스(13)에 연결될 수 있다. 이러한 특정 실시예에서, 버스 종류는 주변 장치 연결(peripheral component interconnect, PCI) 버스, PCI-확장(PCI-X), PCIE (PCI 고속) 버스, 기가비트 이더넷(gigabit Ethernet, GBE) 버스, 및 범용 직렬 버스 (USB)를 포함한다. 그렇지만, 이러한 버스 종류는 예시적인 것이며, 많은 다른 버스 종류가 또한 I/O 인터페이스(13)에 연결될 수 있다. 주변 장치는 주변 버스 중 일부 또는 모두에 연결될 수 있다. 이러한 주변 장치는 (비-제한적으로) 키보드, 마우스, 프린터, 스캐너, 조이스틱 또는 또 다른 종류의 게임 제어기, 매체 기록 장치, 외부 저장 장치, 네트워크 인터페이스 카드, 등을 포함한다. 대응하는 주변 버스를 통하여 I/O 장치(13)에 연결될 수 있는 주변 장치 중 적어도 일부는 직접 메모리 액세스(DMA)를 사용하여 메모리 액세스 요청을 어서트할 수 있다. 이러한 요청(이는 판독 및 기록 요청을 포함할 수 있음)은 I/O 인터페이스(13)를 통하여 노스 브리지(12)로 전송될 수 있으며, 메모리 제어기(18)로 라우팅될 수 있다.

[0022] 제시된 구체 예에서, IC(2)는 컴퓨터 시스템(10)의 디스플레이(3)에 연결된 디스플레이/비디오 엔진(14)을 포함한다. 디스플레이(3)는 평판(flat-panel) LCD(liquid crystal display, 액정 디스플레이), 플라즈마 디스플레이, CRT(cathode ray tube, 음극선관), 또는 또 다른 임의 적절한 디스플레이 종류일 수 있다. 디스플레이/비디오 엔진(14)은 다양한 비디오 프로세싱 기능을 수행할 수 있으며 프로세싱된 정보를 시각 정보로서 출력을 위하여 디스플레이(3)에 제공할 수 있다. 일부 비디오 프로세싱 기능, 예컨대 3-D 프로세싱, 비디오 게임용 프로세싱, 및 더욱 복잡한 유형의 그래픽 프로세싱이 그래픽 엔진(15)에 의해 수행될 수 있으며, 프로세싱된 정보는 노스 브리지(12)를 통하여 디스플레이/비디오 엔진(14)으로 릴레이된다.

[0023] 이러한 특정 실시예에서, 컴퓨터 시스템(10)은 비-통합 메모리 아키텍처(non-unified memory architecture, NUMA) 실행을 구현하며, 여기서 비디오 메모리와 RAM은 서로 분리된다. 제시된 구체 예에서, 컴퓨터 시스템(10)은 디스플레이/비디오 엔진(14)에 연결된 디스플레이 메모리(300)를 포함한다. 따라서, 메모리(6)로부터 비디오 데이터를 수신하는 대신, 비디오 데이터는 디스플레이 메모리(300)로부터 디스플레이/비디오 엔진(14)에 의해 액세스될 수 있다. 이는 차례로 주변 버스 중 하나를 통하여 I/O 인터페이스(13)에 연결된 임의 주변 장치 및 각각의 코어(11)에 대한 더 큰 메모리 액세스 대역폭을 가능하게 한다.

[0024] 제시된 구체 예에서, IC(2)는 시스템 클럭 신호를 수신하기 위하여 연결된 위상-잠금 루프(PLL)(4)를 포함한다. PLL(4)은 대응하는 클럭 신호를 각각의 프로세싱 노드(11)에 분배시킬 수 있다. 본 구체 예에서, 각각의 프로세싱 노드(11)에 의해 수신된 클럭 신호는 서로에 대하여 독립적이다. 더욱이, PLL(4)은 본 구체 예에서 서로 독립적으로 프로세싱 노드(11) 각각에 제공된 클럭 신호 각각의 주파수를 개별적으로 제어하고 변화시키도록 구성된다. 이하에서 더욱 상세하게 논의되듯이, 프로세싱 노드(11)의 임의 주어진 하나에 의해 수신되는 클럭 신호

의 주파수는 이들에 부여된 성능 수요에 따라 증가하거나 감소할 수 있다. 클럭 신호가 PLL(4)로부터 출력되는 여러 주파수는 프로세싱 노드(11) 중 하나에 대한 서로 다른 동작점에 대응할 수 있다. 따라서, 프로세싱 노드(11) 중 특정한 하나에 대한 동작점의 변화는 그 각각의 수신된 클럭 신호의 주파수의 변화에 의해 효과를 나타낼 수 있다.

[0025] 하나 이상의 프로세싱 노드(11)의 각각의 동작점을 변화시키는 것이 하나 이상의 대응하는 클럭 주파수를 변화시키는 것을 포함하는 경우에 있어서, 전력 관리 장치(20)가 PLL(4)에 제공된 디지털 신호 SetF[M:0]의 상태를 변화시킬 수 있다. 이러한 신호 변화에 응답하여, PLL(4)은 영향받은 프로세싱 노드(들)의 클럭 주파수를 변화시킬 수 있다.

[0026] 제시된 구체 예에서, IC(2)는 또한 전압 조절기(5)를 포함한다. 또 다른 구체 예에서, 전압 조절기(5)는 IC(2)로부터 분리되어 실행될 수 있다. 전압 조절기(5)는 공급 전압을 각각의 프로세싱 노드(11)에 제공할 수 있다. 일부 구체 예에서, 전압 조절기(5)는 특정 동작점에 따라 가변적인 공급 전압을 제공할 수 있다(예컨대, 고성능을 위하여 증가되고, 높은 전력 절약을 위해 감소됨). 일부 구체 예에서, 각각의 프로세싱 노드(11)는 전압 플레인을 공유할 수 있다. 따라서, 각각의 구체 예에서 각각의 프로세싱 노드(11)는 프로세싱 노드(11) 중 다른 것과 동일한 전압에서 작동한다. 또 다른 구체 예에서, 전압 플레인은 공유되지 않으며, 따라서 각각의 프로세싱 노드(11)에 의해 수신된 공급 전압은 프로세싱 노드(11) 중 다른 것에 의해 수신된 각각의 공급 전압에 독립하여 설정되고 조절될 수 있다. 따라서, 공급 전압의 조절을 포함하는 동작점 조절은 비-공유된 전압 플레인을 갖는 구체 예의 또 다른 것과 독립적으로 각각의 프로세싱 노드(11)에 선택적으로 적요될 수 있다. 동작점을 변화시키는 것이 하나 이상의 프로세싱 노드(11)에 대한 동작 전압을 변화시키는 것을 포함하는 경우에 있어서, 전력 관리 장치(20)는 전압 조절기(5)에 제공된 디지털 신호 SetV[M:0]의 상태를 변화시킬 수 있다. 신호 SetV[M:0]의 변화에 응답하여, 전압 조절기(5)은 프로세싱 노드(11) 중 영향받은 하나에 제공된 공급 전압을 조절할 수 있다.

[0027] 동작점은 각각의 활동 수준(예컨대, 프로세싱 워크로드)에 따라, 서로 독립적으로, 각각의 프로세싱 노드(11)에 대하여 설정될 수 있다. 제시된 구체 예에서, 전력 관리 장치(20)는 각각의 프로세싱 노드(11)에 대한 활동 수준을 지시하는 정보를 수신할 수 있다. 각각의 프로세싱 노드에 대한 지시된 활동 수준은 임계 값과 비교될 수 있으며, 대응하는 프로세싱 노드의 동작점이 이에 따라 비교 결과에 기초하여 조절될 수 있다. 임계치는 상위 활동 임계치 및 하위 활동 임계치를 포함할 수 있다. 주어진 프로세싱 노드(11)에 대한 활동 수준이 상위 활동 임계치를 초과하는 경우, 그 동작점은 사전정의된 상위 동작점(예컨대, 최상위 클럭 주파수 및 동작 전압)으로 조절될 수 있다. 주어진 프로세싱 노드(11)에 대한 활동 수준이 하위 활동 임계치 미만인 경우, 동작점은 사전정의된 하위 비-유휴 동작점(예컨대, 최하위 비-제로 클럭 주파수 및 전압)으로 조절될 수 있다.

[0028] 비교 동작이 주어진 프로세싱 노드(11)의 활동 수준이 상위 활동 임계치 미만이지만 하위 활동 임계치 초과인 것을 나타내는 경우, 전력 관리 장치(20)는 운영 체제 소프트웨어(또는 기타 소프트웨어)가 특정 프로세싱 노드(11)로 하여금 하나 이상의 중간 동작점 중 하나에서 작동하게 하도록 할 수 있다. 일부 구체 예에서, 단일 중간 동작점이 구현될 수 있다. 또 다른 구체 예에서, 다중 중간 동작점이 사용될 수 있다.

[0029] 각각의 프로세싱 노드(11)에 대한 활동 수준을 전술한 임계 값과 비교하는 것은 다수의 연속적 시간 간격에서 수행될 수 있다. 주어진 시간 간격 동안의 비교가 프로세싱 노드(11)의 동작점을 조절할 필요성을 나타내는 경우,

[0030] 동작점은 그 다음 이어지는 시간 간격 동안 조절될 수 있다. 비교 동작이 각각의 간격 동안 1회 수행될 수 있으며, 비교 결과에 기초하여 이루어지는 모든 조절이 그 다음 이어지는 간격에 적용될 수 있다. 비교 결과가 변화가 필요없다는 것을 나타내는 경우, 프로세싱 노드(11)는, 비교 결과가 서로 다른 동작점에 대응하는 활동 수준을 나타낼 때까지, 하나 이상의 잇따르는 시간 간격 동안 자신의 현재 동작점에서 지속할 수 있다.

[0031] 한 구체 예에서, 운영 체제(OS) 소프트웨어와 같은 집적 회로(2)에서 실행되는 소프트웨어는,

[0032] 활동 수준이 상위 임계치 미만이고 하위 임계치보다 더 큰 경우, 각각의 프로세싱 노드(11)에 대한 동작점을 선택할 수 있다. 그렇지만, 비교 동작이 각각의 시간 간격 동안 전력 관리 장치에 의해 수행되도록 지속될 수 있다. 비교 동작이 상위 임계치를 초과하거나 또는 하위 임계치 미만인 활동 수준을 검출하는 경우, 영향받은 프로세싱 노드(11)에 대한 동작점이 이에 따라 변화될 수 있으며, 그렇지 않은 경우 중간 동작점을 선택하기 위하여 사용되는 OS 또는 기타 메커니즘에 의해 특정될 수 있는 것을 무시한다. OS는 각각의 시간 간격이 전력 관리 장치(20)에 의해 사용되는 시간 간격의 기간보다 더 큰 기간을 갖는 시간 간격의 또 다른 세트에 걸쳐 활동을

모니터링할 수 있다. OS가 활동 수준을 모니터링하는 시간 간격의 기간은 전력 관리 장치(20)의 기간보다 최소 10배 더 클 수 있다. 예를 들어, 한 구체 예에서, 전력 관리 장치(20)는 100 마이크로초 이하의 시간 간격 동안 활동 수준을 모니터링 할 수 있다(그리고 이에 따라 동작점 변화를 구현할 수 있다). 이와 반대로, 본 구체 예에서 운영 체제 소프트웨어는 30 밀리초 이상의 시간 간격 동안 활동 수준을 모니터링 할 수 있다(그리고 이에 따라 동작점 변화를 구현할 수 있다).

[0033] 전술한 바와 같이, 프로세싱 노드(11)의 동작점은 적어도 클럭 주파수에 의해 정의될 수 있으며, 또한 동작 전압에 의해 정의될 수 있다. 일반적으로 말하면, "상위" 동작점으로의 전이는 영향받은 프로세싱 노드(11)에 대한 클럭 주파수를 증가시키는 것에 의해 정의될 수 있다. 상위 동작점으로의 전이는 또한 자신의 동작 (예컨개, 공급) 전압을 증가시키는 것을 포함할 수 있다. 유사하게, "하위" 동작점으로의 전이는 영향받은 프로세싱 노드(11)에 대한 클럭 주파수를 감소시키는 것에 의해 정의될 수 있다. 영향받은 프로세싱 노드(11)에 제공된 동작/공급 전압의 감소가 또한 하위 동작점으로의 전이의 정의에 포함될 수 있다.

[0034] 한 구체 예에서, 동작점은 고급 구성 및 전력 인터페이스(Advanced Configuration and Power Interface, ACPI) 규범(specification)의 성능 상태(performance states)(이하 'P-상태'라 함)에 대응할 수 있다. 아래 표 1은 ACPI 표준을 사용하여 실행되는 한 구체 예에 대한 P-상태를 나열한다.

표 1

[0035]

P-상태 색인	주파수	전압
P0	2 GHz	1.1 V
P1	1.8 GHz	1.0 V
P2	1.5 GHz	0.9 V
P3	1 GHz	0.85 V
P4	800 MHz	0.8 V

[0036] 위 표 1에 나열된 P-상태는 ACPI-순응 프로세서가 C0으로 알려진 비-유휴 상태에서 작동할 때 적용될 수 있다. 위 표 1에 대응하는 구체 예에 대하여, P-상태 P0은 2 GHz의 클럭 주파수와 1.1 볼트의 동작 전압을 갖는 최상위 동작점이다. 전력 관리 장치(20)는 한 구체 예에서 프로세싱 노드(11)가 상위 활동 임계치를 초과하는 대응하는 활동 수준에 응답하여 P-상태 P0에서 작동하도록 할 수 있다. P-상태 P0에서의 작동은 컴퓨팅-집중적인 프로세싱 워크로드에 대하여 사용될 수 있다. 컴퓨팅-집중적 워크로드는 소량(존재하는 경우)의 메모리 액세스를 요구하는 시간 민감성 및 전산 집중적(computationally intensive)일 수 있다. 최대 성능을 유지하는 한편 또한 낮은 전력 소모에 상응하는 P-상태로 신속하게 되돌리도록 워크로드를 가능한 한 짧은 시간에서 실행하는 것이 바람직할 수 있다. 따라서, 상위 활동 수준을 갖는 컴퓨팅-집중적 워크로드는 P-상태 P0에서 실행될 수 있으며, 이는 더욱 신속한 완료를 가능하게 한다.

[0037] P-상태 P4는 본 특정 구체 예에서 800 MHz의 클럭 주파수 및 0.8V의 동작 전압을 갖는 최하위 비-유휴 동작점이다. 전력 관리 장치(20)는 프로세싱 노드(11)가 하위 활동 임계치 미만인 대응하는 활동 수준에 응답하여 P-상태 P4에서 작동하도록 할 수 있다. P-상태 P4는 메모리-집중적 워크로드뿐만 아니라 시간-민감성(또는 주파수-민감성)이지 않은 또 다른 업무에 대해 사용될 수 있다. 메모리-집중적 워크로드는 시스템 메모리에 대한 빈번한 액세스를 포함하는 것들이다. 메모리 액세스가 대규모 대기시간을 포함하기 때문에(메모리에 액세스하지 않는 명령의 실행 시간과 비교하여), 메모리-집중적 워크로드에 대한 클럭 주파수를 감소시키는 것은 시스템의 와트 당 성능(performance-per-watt) 용량을 개선할 수 있는 전력 절약과 함께 최소한의 성능 영향을 가질 수 있다.

[0038] 검출된 활동 수준이 하위 활동 임계치보다 크고 상위 활동 임계치 미만인 경우, 대응하는 프로세싱 노드(11)의 동작은 구체적인 실시예에 따라, OS, 기타 소프트웨어, 펌웨어, 및/또는 다른 하드웨어의 지시에 의해, P-상태 P1, P2, P3 중 어느 하나로 설정될 수 있다.

[0039] 표 1에 나열된 P-상태는 동작점의 예시적인 세트임에 유의하라. 다른 클럭 주파수 및 동작 전압을 갖는 동작점을 사용하는 구체 예가 가능하며 고려된다. 또한, 전술한 바와 같이, 일부 구체 예는 프로세싱 노드(11)에 대한 공유된 전압 플레인을 사용할 수 있으며, 이에 따라 이들 각각의 동작점은 클럭 주파수에 기초하여 정의될 수 있다. 일부 구체 예에서, 각각의 프로세싱 노드에 대한 동작 전압은 고정되어 잔류할 수 있으며, 한편 또 다른 구체 예에서, 동작 전압은 동시에 모든 프로세싱 노드(11)에 대하여 조절될 수 있다.

[0040] 전술한 여러 장치가 별도의 IC에서 실행되는 구체 예가 가능하며 고려된다는 것에 주목하여야 한다. 예를 들어, 코어(11)가 제1 IC에서 실행되고, 노스 브리지(12) 및 메모리 제어기(18)가 또 다른 IC에서 실행되며, 한편 나머지 기능 유닛이 또 다른 IC에서 실행되는 한 가지 구체 예가 고려된다. 일반적으로, 전술한 기능 유닛은 단일 IC를 비롯하여, 가능한 많은 또는 적은 서로 다른 IC에서 실행될 수 있다.

[0041] 위 표1에서 P-상태로서 나열된 동작점은 또한 비-ACPI 구체 예에서 사용될 수 있음에 주목하라.

[0042] **프로세싱 노드:**

[0043] 도 2는 프로세싱 노드(11)의 하나의 구체 예의 블록 다이어그램이다. 프로세싱 노드(11)는 시스템 메모리(200)에 저장될 수 있는 명령을 실행하도록 구성된다. 많은 이러한 명령들은 또한 시스템 메모리(200)에 저장되는 데이터에서 작동한다. 시스템 메모리(200)가 컴퓨터 시스템 전반에 물리적으로 분포될 수 있으며 및/또는 하나 이상의 프로세서(100)에 의해 액세스될 수 있음에 주목하라.

[0044] 제시된 구체 예에서, 프로세싱 노드(11)는 레벨 1(L1) 명령 캐쉬(106) 및 L1 데이터 캐쉬(128)를 포함할 수 있다. 프로세싱 노드(11)는 명령 캐쉬(106)에 연결된 프리페치 유닛(108)을 포함할 수 있다. 디스패치 유닛(dispatch unit)(104)은 명령 캐쉬(106)로부터 명령을 수신하고 스케줄러(118)로 동작을 디스패치하도록 구성될 수 있다. 하나 이상의 스케줄러(118)가 디스패치 유닛(104)으로부터 디스패치된 동작을 수신하고 동작을 하나 이상의 실행 유닛(124)으로 발송(issue)하기 위하여 연결될 수 있다. 실행 유닛(124)은 하나 이상의 정수 유닛(integer unit), 하나 이상의 부동 소수점 유닛(floating point unit), 및 하나 이상의 로드/저장 유닛을 포함할 수 있다. 실행 유닛(124)에 의해 발생된 결과는 하나 이상의 결과 버스(130)로 출력될 수 있다(하나의 결과 버스가 명확성을 위해 본 명세서에 제시되지만, 복수의 결과 버스가 가능하며 고려된다). 이러한 결과는 후속하는 발송된 명령에 대한 오퍼랜드 값(operand value)으로서 사용되거나 및/또는 레지스터 파일(116)에 저장될 수 있다. 퇴거 큐(retire queue)(102)가 스케줄러(118) 및 디스패치 유닛(104)에 연결될 수 있다. 퇴거 큐(102)는 언제 각각의 발송된 동작이 퇴거되는가를 결정하도록 구성될 수 있다.

[0045] 한 구체 예에서, 프로세싱 노드(11)는 x86 아키텍처(또한 Intel Architecture-32, 즉 IA-32로 알려져 있음)과 호환성이도록 설계될 수 있다. 또 다른 구체 예에서, 프로세싱 노드(11)는 64-비트 아키텍처와 호환성일 수 있다. 기타 아키텍처와 호환성인 프로세싱 노드(11)의 구체 예가 또한 고려된다.

[0046] 각각의 프로세싱 노드(11)는 또한 많은 다른 성분을 포함할 수 있음에 주목하라. 예를 들어, 프로세싱 노드(11)는 명령 쓰레드(instruction thread)를 실행할 때 브랜치를 예측하도록 구성된 브랜치 예측 유닛(branch prediction unit)(도시되지 않음)을 포함할 수 있다.

[0047] 명령 캐쉬(106)가 디스패치 유닛(104)에 의한 페치(fetch)용 명령을 저장할 수 있다. 명령 코드가 프리페치 유닛(108)을 통하여 시스템 메모리(200)로부터 프리페칭 코드에 의한 저장을 위하여 명령 캐쉬(106)에 제공될 수 있다. 명령 캐쉬(106)는 다양한 구성에서 구현될 수 있다(예컨대, 세트-연관형(set-associative), 완전-연관형(fully-associative), 또는 직접 매핑형(direct-mapped)).

[0048] 프로세싱 노드(11)는 또한 레벨 2(L2) 캐쉬(140)를 포함할 수 있다. 명령 캐쉬(106)가 명령을 저장하기 위하여 사용될 수 있고 데이터 캐쉬(128)가 데이터(예컨대, 오퍼랜드)를 저장하기 위하여 사용될 수 있는 한편, L2 캐쉬(140)는 명령 및 데이터를 저장하기 위해 사용되는 통합형일 수 있다. 비록 본 명세서에서 명확하게 제시되지 않지만, 일부 구체 예는 또한 레벨 3(L3) 캐쉬를 포함할 수 있다. 일반적으로, 캐쉬 레벨의 수는 구체 예마다 변할 수 있다.

[0049] 프리페치 유닛(108)은 명령 코드를 명령 캐쉬(106) 내에 저장하기 위해 시스템 메모리(200)로부터 프리페치(prefetch)할 수 있다. 프리페치 유닛(108)은 기법 및 알고리즘을 프리페치하는 다양한 특정 코드를 사용할 수 있다.

[0050] 디스패치 유닛(104)은 실행 유닛(124)에 의해 실행가능한 동작뿐만 아니라 오퍼랜드 어드레스 정보, 즉시 데이터 및/또는 변위 데이터(displacement data)를 출력할 수 있다. 일부 구체 예에서, 디스패치 유닛(104)은 일부 명령을 실행 유닛(124) 내에서 실행가능한 동작으로 디코딩하기 위한 디코딩 회로 (제시되지 않음)를 포함할 수 있다. 단순한 명령들이 단일 동작에 대응할 수 있다. 일부 구체 예에서, 더욱 복잡한 명령들이 복수의 동작에 대응할 수 있다. 레지스터의 업데이트를 포함하는 동작의 디코딩시에, 레지스터 파일(116) 내 레지스터의 위치는 추측 레지스터 상태를 저장하기 위하여 보존될 수 있다(대안적인 구체 예에서, 재배열 버퍼가 각각의 레지스터에 대한 하나 이상의 추측 레지스터 상태를 저장하기 위하여 사용될 수 있으며 레지스터 파일(116)이 각각의

레지스터에 대한 투입된 레지스터 상태를 저장할 수 있다). 레지스터 맵(register map)(134)은 레지스터 재명명을 촉진하기 위하여, 소스 및 대상 오퍼랜드의 로직컬 레지스터 명칭을 물리 레지스터 번호로 번역할 수 있다. 레지스터 맵(134)은 레지스터 파일(116) 내 어느 레지스터가 현재 할당되고 할당되지 않은지를 기록할 수 있다.

[0051] 도 2의 프로세싱 노드(11)는 비순차적 실행을 지원할 수 있다. 퇴거 큐(102)는 레지스터 판독 및 기록 동작에 대한 초기 프로그램 시퀀스(original program sequence)를 기록할 수 있으며, 이는 추측 명령 실행 및 브랜치 오추측 회복을 허용하며, 정밀한 예외를 촉진한다. 일부 구체 예에서, 퇴거 큐(102)는 또한 추측 레지스터 상태(예컨대 재배열 버퍼와 유사)에 대한 데이터 값 저장을 제공함으로써 레지스터 재명명을 지원할 수 있다. 또 다른 구체 예에서, 퇴거 큐(102)는 재배열 버퍼와 유사한 기능을 하지만 어떠한 데이터 값 저장도 제공하지 않는다. 동작이 퇴거되면서, 퇴거 큐(102)가 레지스터 파일(116)에서 레지스터를 할당해제할 수 있으며, 더 이상 추측 레지스터 상태를 저장하고 어느 레지스터가 현재 미사용(free)인지를 나타내는 레지스터 맵(134)에게 신호를 제공할 필요가 없다. 이러한 상태를 발생시켰던 동작이 승인될 때까지 레지스터 파일(116) 내에서(또는, 다른 구체 예에서, 재배열 버퍼 내에서) 추측 레지스터 상태를 유지함으로써, 브랜치 예측이 옳지 않은 경우 오추측 경로를 따른 추측성-실행 동작(speculatively-executed operation)의 결과가 무효가 될 수 있다.

[0052] 한 구체 예에서, 레지스터 파일(116) 중 소정의 레지스터는 실행된 명령의 데이터 결과를 저장하도록 구성될 수 있으며 또한 실행된 명령에 의해 업데이트될 수 있는 하나 이상의 슬래그 비트를 저장할 수 있다. 슬래그 비트는 후속 명령을 실행하는데 있어서 중요할 수 있는 여러 유형의 정보를 전송할 수 있다(예컨대 캐리(carry) 또는 오버플로우(overflow) 상황이 합 또는 곱 연산의 결과로서 존재한다는 것을 나타냄). 아키텍처적으로, 플래그 레지스터는 플래그를 저장하는 것으로 정의될 수 있다. 따라서, 소정의 레지스터에 대한 기록은 로직컬 레지스터 및 플래그 레지스터 둘 모두를 업데이트 할 수 있다. 모든 명령이 하나 이상의 플래그를 업데이트하는 것은 아니라는 점에 유의하라.

[0053] 레지스터 맵(134)은 물리 레지스터를 동작을 위한 대상 오퍼랜드로서 특정된 특정한 로직컬 레지스터(예컨대, 아키텍처 레지스터 또는 마이크로아키텍처적으로 특정된 레지스터)에 할당할 수 있다. 디스패치 유닛(104)은, 레지스터 파일(116)이 주어진 동작에서 소스 오퍼랜드로서 특정된 로직컬 레지스터에 할당된 사전 할당된 물리 레지스터를 갖는지를 결정할 수 있다. 레지스터 맵(134)은 해당 로직컬 레지스터에 가장 최근 할당된 물리 레지스터에 태그를 제공할 수 있다. 이러한 태그는 레지스터 파일(116) 내 오퍼랜드의 데이터 값에 액세스하기 위하거나 또는 결과 버스(130) 상에서의 결과 전달을 통하여 데이터 값을 수신하기 위하여 사용될 수 있다. 오퍼랜드가 메모리 장소(memory location)에 대응하는 경우, 오퍼랜드 값은 로드/저장 유닛(제시되지 않음)을 통하여(결과 전달 및/또는 레지스터 파일(116) 내 저장을 위하여) 결과 버스에 제공될 수 있다. 오퍼랜드 데이터 값은 오퍼랜드가 스케줄러(118) 중 하나에 의해 발송될 때 실행 유닛(124)에 제공될 수 있다. 대안적인 구체 예에서, (동작이 발송될 때 대응하는 실행 유닛(124)에 제공되기 더욱 오히려) 오퍼랜드 값은 동작이 디스패치될 때, 대응하는 스케줄러(118)에 제공될 수 있음에 주목하라.

[0054] 본 명세서에서 사용되듯이, 스케줄러는 동작이 언제 실행할 준비가 되고 준비 동작을 하나 이상의 실행 유닛에 발송하는가를 탐지하는 장치이다. 예를 들어, 예약 스테이션이 스케줄러의 한 종류일 수 있다. 실행 유닛마다 독립적 예약 스테이션이 제공될 수 있거나, 또는 이로부터 동작이 발송되는 중앙 예약 스테이션이 제공될 수 있다. 또 다른 구체 예에서, 퇴거까지 동작을 유지하는 중앙 스케줄러가 사용될 수 있다. 각각의 스케줄러(118)는 실행 유닛(124)으로의 발송을 대기하는 몇몇 펜딩 동작(pending operation)에 대한 동작 정보(예컨대, 동작뿐만 아니라 오퍼랜드 값, 오퍼랜드 태그, 및/또는 즉시 데이터)를 홀딩할 수 있다. 일부 구체 예에서, 각각의 스케줄러(118)는 오퍼랜드 값 저장을 제공하지 않을 수 있다. 그 대신에, 각각의 스케줄러는 언제 오퍼랜드 값이(레지스터 파일(116) 또는 결과 버스(130)로부터) 실행 유닛(124)에 의해 판독가능할지를 결정하기 위하여 레지스터 파일(116)에서 활용가능한 발송된 동작 및 결과를 모니터링할 수 있다.

[0055] 비록 본 명세서에서 명확하게 제시되지 않지만, 다수의 다른 통신 경로가 프로세싱 노드(11)의 여러 유닛(명확하게 제시되지 않은 유닛 포함)과 도 1에 제시된 전력 관리 장치(20) 사이에 제공될 수 있다. 더욱 구체적으로, 프로세싱 노드(11)는 활동 수준을 전력 관리 장치(20)에 나타내는 정보를 제공하기 위하여 이러한 통신 경로를 사용할 수 있다. 예를 들어, 퇴거 큐(102)는 명령 퇴거에 관한 정보를 전력 관리 장치(20)에 제공할 수 있다. 또 다른 실시예에서, 실행 유닛(124)은 실행된 명령에 관한 정보를 제공할 수 있고, 디스패치 유닛(104)은 디스패치된 명령에 관한 정보를 제공할 수 있고, 스케줄러(118)는 스케줄된 명령에 관한 정보를 제고할 수 있으며, 그리고 여러 캐쉬 중 어느 하나(또는 모두)는 캐쇠 히트(hit) 또는 미스(miss)에 관한 정보를 제공할 수 있다. 존재할 수 있으나 본 명세서에 명확하게 제시되지 않은 유닛 중에서, 브랜치 예측 유닛은, 한 예로서, 브랜치 오예측에 관한 정보를 제공할 수 있다. 여기서 언급되지 않은 기타 유닛이 또한 또 다른 종류의 정보를 전력 관

리 장치(20)에게 제공할 수 있다. 프로세싱 노드(11)의 여러 유닛으로부터 수신된 정보는 그 활동 수준을 결정하기 위하여 사용될 수 있다. 활동 수준을 결정하는 여러 방법이 적용될 수 있으며, 다른 종류 정보보다 특정 종류의 정보에 가중치를 둘 수 있다. 또한, 일부 종류의 정보는 일정 시간에서 함께 무시될 수 있다. 제공된 정보로부터 결정된 활동 수준은 순서대로 프로세싱 노드(11)에 대한 동작점을 선택하기 위해 사용될 수 있다.

[0056] **개념 뷰(Conceptual View):**

[0057] 도 3은 애플리케이션 활성에 관한 전력 관리 장치의 하나의 구체 예의 작동을 나타내는 다이어그램이다. 제시된 실시예에서, 그래프는 3가지 주요 영역으로 분할되는데, 즉 '유휴(Idle)'와 하위 임계치 사이의 영역, 하위 임계치와 상위 임계치 사이 영역, 그리고 상위 임계치 상부 영역이다.

[0058] 제1 영역(즉, 하위 임계치 미만)에서의 활동 수준은 메모리-집중식이거나 또는 그렇지 않은 경우 저주파수 민감도(대응하는 프로세싱 노드(11)의 클럭 주파수에 비하여)인 프로세싱 워크로드를 나타낸다. 이러한 활동 수준이 검출되는 프로세싱 노드(11)는 전체 프로세싱 노드 성능에 거의(존재하는 경우) 부적정 영향 없이 가능한 한 최하위 비-유휴 동작점에 위치할 수 있다. 최하위 비-유휴 동작점은 최하위 클럭 주파수를 갖는 것으로 정의될 수 있다. 가능한 최하위 동작점은 가능한 최하위 동작 전압을 갖는 것으로 정의될 수 있다. 위 표 1의 예를 사용하여, 프로세싱 노드(11)의 활동 수준이 소정의 시간 간격 동안 이러한 영역에 존재하는 것으로 결정될 때, 이는 적어도 그 다음 시간 간격 동안 P-상태 P4에 위치할 수 있다.

[0059] 제3 영역(즉, 상위 임계치 초과)에서의 활동 수준은 컴퓨팅-집중식이거나 또는 그렇지 않은 경우 고주파수 민감도(클럭 신호에 비하여)인 프로세싱 워크로드를 나타낸다. 이러한 활동 수준이 시간 간격 동안 소정의 프로세싱 노드(11)에 대하여 검출될 때, 최상위 동작점이 적어도 그 다음 시간 간격 동안 해당 노드에 대하여 선택될 수 있다. 위 표 1의 예를 사용하여, 상위 활동 수준을 검출하는 것은 전력 관리 장치(20)에 의해 대응하는 프로세싱 노드가 P0 상태에 놓이게 되는 결과를 야기한다. 이는 워크로드가 최대 성능을 갖는 한편 업무를 더욱 신속하게 완료하도록 실행되는 것을 가능하게 한다.

[0060] 주어진 시간 간격 동안 검출된 활동 수준이 제2 영역(하위 임계치를 초과하나 상위 임계치 미만)에서 검출될 때, 대응하는 프로세싱 노드는 운영 체제 (OS), 기타 소프트웨어, 펌웨어, 또는 또 다른 하드웨어에 의해 요청되는 동작점에 위치될 수 있다. 표 1의 P-상태 P1, P2, 및 P3는 제2 영역에서 사용될 수 있는 중간 동작점의 예이다.

[0061] OS 또는 또 다른 소프트웨어가 제2 영역 내 동작점을 선택하는 구체 예에서, 프로세싱 노드는 다수의 시간 간격에 대한 해당 특정 지점에서의 작동을 유지할 수 있으며, 이는 검출된 애플리케이션 활동 수준이 제2 영역(즉, 하위 임계치 초과 그리고 상위 임계치 미만)에 상응하도록 유지되는 것을 가정한다. 하나의 중간 동작점으로부터 또 다른 동작점으로의 변화는 최상위 및 최하위 동작점에서의 동작을 결정하기 위하여 사용되는 시간 간격보다 훨씬 더 긴 시간 간격에 따라 일어날 수 있다. 하위 및 상위 임계치에 대한 애플리케이션 활동 수준의 비교가 대략 매 10-100 마이크로초마다의 간격 동안 지속적으로 일어날 수 있는 한편, 어느 중간 동작점이 가장 적합한지를 결정하기 위하여 사용되는 비교는 대략 매 30-100 밀리초마다의 간격 동안 일어날 수 있다. 따라서, 전력 관리 장치(20)는 하위 및 상위 임계치에 대한 애플리케이션 활동 수준의 비교를 지속적으로 수행할 수 있으며, 대응하는 활동 수준이 검출되는 임의 시간에서 상기 영향받은 프로세싱 노드(11)를 최상위 또는 최하위 동작점으로 이동시킴으로써 중간 동작점 선택을 무시할 수 있다.

[0062] 도 3에 제시된 상위 및 하위 임계치에 추가하여, 히스테리시스 임계 레벨(hysteresis threshold level)이 또한 프로세싱 노드의 동작점을 변화시키거나 그렇지 않은지를 결정할 때 고려될 수 있다. 상위 히스테리시스 임계치가 또한 최상위 동작점으로 또는 이로부터 밖으로 전이하는지를 결정할 때 고려될 수 있는 한편, 하위 히스테리시스 임계치가 최하위 동작점으로 또는 이로부터 밖으로 전이하는지를 결정할 때 고려될 수 있다. 이러한 히스테리시스 임계치를 사용하는 것은 변칙으로 인한 비-최적 동작점으로의 전이를 방지할 수 있다. 예를 들어, 컴퓨팅-집중적 워크로드가 P-상태 P0에서 실행되는 상황을 고려하자. 이러한 상황에서 브랜치 오예측이 파이프라인 스톨(pipeline stall)을 야기할 수 있으며, 이에 따라 전체 활동 수준의 순간 감소를 초래할 수 있다. 상위 히스테리시스 임계 레벨은 이러한 상황에서 인자화(factored)될 수 있으며, 이에 따라 대응하는 프로세싱 노드(11)가 P-상태 P0에서의 작동을 유지하도록 할 수 있다.

[0063] 앞서 기재된 동작은 소모되는 전력의 와트 당 성능을 개선함으로써 프로세서의 효율을 개선할 수 있다. 최소 주파수 민감성 및/또는 메모리-집중식 애플리케이션에 대한 가능한 한 최하위 동작 값으로 클럭 주파수 및 동작 전압을 감소시키는 것은 순서대로 이러한 애플리케이션이, 다른 방식의 경우 필요하지 않은 전력을 소모하지 않

으면서 적절한 방식으로 여전히 실행하는 것을 가능하게 한다. 최대 주파수 민감성 및/또는 컴퓨팅-집중식 애플리케이션인 이러한 애플리케이션에 대한 가능한 한 최상위 동작 값으로 클럭 주파수 및 동작 전압을 증가시키는 것은 이러한 애플리케이션이 바람직한 성능 레벨에서 더욱 신속하게 실행하는 것을 허용하며 이에 따라 완료시에 낮은 전력 소모로써 동작점으로 신속하게 복귀하도록 한다.

[0064] **전력 관리 장치 및 그 성분.**

[0065] 도 4는 전력 관리 장치의 하나의 구체 예를 나타내는 블록 다이어그램이다. 제시된 구체 예에서, 전력 관리 장치(20)는 활동 모니터(202), 초 당 수행된 명령(committed instructions per second, CIPS) 유닛(204), 및 사이클-당-명령(instructions-per-cycle, IPC) 유닛(206)을 사용하여 하나 이상의 프로세싱 노드 각각의 활동 수준을 모니터링하도록 구성되며, 이들 각각은 이하에서 더욱 상세하게 논의될 것이다. 이러한 여러 유닛들은 정보를 결정 유닛(208)에게 제공할 수 있다. 이러한 정보는 각각의 프로세싱 노드(11)에 대한 동작점을 설정하기 위하여 결정 유닛(208)에 의해 사용될 수 있다. 결정 유닛(208)은 이러한 유닛들 중 선택된 유닛 또는 이들 각각으로부터 제공된 정보를 각각의 프로세싱 노드(11)에 대한 동작점을 결정하기 위하여 고려할 수 있다. 결정 유닛(208)은 또한 각각의 프로세싱 노드(11)의 현재 동작점을 나타내는 정보를 수신하도록 연결될 수 있다.

[0066] 주어진 프로세싱 노드(11)의 동작점에 대한 변화는 여기에 제공된 클럭 신호의 주파수를 변화시킴으로써 수행될 수 있으며, 또한 각각의 공급 전압을 변화시키는 것을 포함할 수 있다. 결정 유닛(208)에 의한 각각의 프로세싱 노드(11)에 대한 클럭 주파수 조절은 신호 SetF[N:0]의 상태에 대한 변화를 통하여 수행될 수 있다. 각각의 프로세싱 노드(11)에 대한 동작 전압 조절은 신호 SetV[N:0]의 상태에 대한 변화를 통하여 수행될 수 있다.

[0067] 도 4에 도시된 전력 관리 장치(20)의 구체 예는 예시적인 것이며, 다른 구체 예가 가능하며 고려된다는 것에 유의하라. 활동 모니터(202), CIPS 유닛(204), 또는 IPC 유닛(206) 중 단지 하나를 포함하는 구체 예가 또한 가능하며 고려되고, 이러한 구체 예는 결정 유닛(208)의 사용 없이 구현될 수 있다. 예를 들어, 프로세싱 노드(11)의 활동 수준이 CIPS 유닛(204)에 제공된 정보에 기초하여 결정되는 구체 예가 고려된다. 이러한 구체 예에서, CIPS 유닛(204)은 결정 유닛(208)에 의해 다른 방식으로 제공되는 기능부 중 적어도 일부를 포함할 수 있다.

[0068] 일반적으로, 전력 관리 장치(20)는 하나 이상의 프로세싱 노드의 활동 수준/프로세싱 워크로드가 모니터링되고 이에 따라 동작점의 변화를 수행할 수 있는 어떠한 구성으로도 구현될 수 있다. 더욱 상세하게, 전력 관리 장치(20)는 하나 이상의 프로세싱 노드 각각의 활동 수준이 복수의 시간 간격 각각에 대한 상위 및 하위 활동 임계치와 비교되고 그 다음 이어지는 시간 간격에 대한 각각의 동작점이 이러한 비교의 결과에 따라 선택될 수 있는 어떠한 구성으로도 구현될 수 있다. 더욱 상세하게, 전력 관리 장치(20)는 이후 이어지는 시간 간격에서의 동작이, 모니터링된 활동 수준이 상위 활동 임계치를 초과하는 경우에는 최상위 동작점(예컨대, 상기 표 1의 P-상태 P0)으로, 활동 수준이 하위 활동 임계치 미만인 경우에는 최하위 동작점(예컨대, P-상태 P4)으로, 또는 활동 수준이 상위 임계치 미만이고 하위 임계치 초과인 경우 하나 이상의 중간 동작점 중 어느 하나(예컨대, P-상태 P1, P2, 또는 P3)로 설정되는 어떠한 구성으로도 구현될 수 있다.

[0069] 이제 도 5를 참조하면, 활동 모니터의 한 가지 구체 예를 도시하는 블록 다이어그램이 제시된다. 제시된 구체 예에서, 활동 모니터(202)는 평균 활동 계산기(average activity calculator, AAC) (250), 임계치 비교기(252), 및 간격 타이머(254)를 포함한다. 활동 모니터(202)는 제시된 구체 예에서 하나 이상의 프로세싱 노드(11)에 대한 활동의 이동 평균을 계산하고 기록하고, 평균 활동 수준을 상위 및 하위 활동 임계치와 비교하도록 구성된다. 비교 결과는 프로세싱 노드에 대한 적절한 동작점을 결정하기 위하여 사용될 수 있다. 제시된 구체 예에서, 비교는 각각의 간격에서 한 번 수행되며, 이는 간격 타이머(254)에 의한 시간이다.

[0070] AAC(250)는 하나 이상의 계측에 기초하여 여기에 연결된 각각의 프로세싱 노드(11)에 대한 활동 수준을 결정할 수 있다. 이러한 특정 구체 예에서, AAC(250)는 파이프라인 스톨, 명령 실행, 캐쉬 히트 및 미스, 브랜치 오예측, 및 발송된 명령에 관한 정보를 수신하도록 연결된다. 본 명세서에 명확하게 제시된 계측 이외에 다른 계측을 사용하거나 본 명세서에 기재된 계측에 부가하여 다른 계측을 사용하는 구체 예가 또한 가능하며 고려된다. 이러한 계측 중 일부가 일부 구체 예에서 다른 것들보다 더욱 가중치를 부여받을 수 있음에 주목하라.

[0071] AAC(250)에 의해 계산되고 기록된 평균 활동 수준은 이러한 계측 중 어느 하나, 이러한 계측 중 둘 이상의 합, 또는 이러한 계측 모두의 결합에 기초하여 결정될 수 있다. 예를 들어, AAC(250)는 프로세서 워크로드가 캐쉬 미스 및 파이프라인 스톨 둘 모두에 기초하는 메모리-집중식인 것을 결정할 수 있으며, 이들 둘 모두는 다수의 메모리 액세스를 요구하는 애플리케이션에서 빈번하게 일어날 수 있다. 또 다른 실시예에서, 적은 캐쉬 액세스(캐쉬 히트 및 미스의 전체 수에 의해 표시됨)를 갖는 다수의 명령 실행이 컴퓨팅-집중적 워크로드를 나타낼 수

있다.

[0072] 현재 시간 간격 동안 수신된 정보뿐만 아니라 연혁적 정보 둘 모두에 기초하여, AAC(250)는 이에 대하여 활성이 보고되었던 각각의 프로세싱 노드(11)에 대한 평균 활성을 결정할 수 있다. 제시된 구체 예에서, 평균은 신호 T를 어서트(assert)하는 간격 타이머(254)에 응답하여 결정될 수 있다. 간격 타이머(254)는 제시된 구체 예에서 간격 클럭 신호를 수신하도록 연결되며, 이러한 클럭 신호의 일정 횟수 사이클이 수신된 이후 신호 T를 어서트할 수 있다

[0073] 각각의 프로세싱 노드(11)에 대한 연산된 활동 평균은 AAC(250)로부터 임계치 비교기(252)로 제공될 수 있다. 임계치 비교기(252)는 신호 T를 어서트하는 간격 타이머(254)에 응답하여 상기 수신된 활동 수준들의 비교를 수행할 수 있으며, 상기 신호 T는 그 비교 입력에서 수신될 수 있다. AAC(250)가 결과를 결정하고 제공하도록 지연 시간(delay time)이 허용되며, 임계치 비교기(252)는 상기 지연 시간이 경과한 이후 비교를 수행한다. 임계치 비교기(252)는 각각의 프로세싱 노드에 대한 계산된 활성 평균을 앞서 논의한 하위 및 상위 활동 임계치와 비교할 수 있다. 이러한 비교 결과는 그 후 임계치 비교기(252)로부터 결정 유닛(208)으로 제공될 수 있으며, 이는 그 후 전술한 바와 같이 전력 관리 장치(20)의 작동에 따라 이러한 프로세싱 노드(11)에 대한 동작점의 조절을 초래할 수 있다. 활동 모니터(202)에서의 이러한 동작은 간격 타이머(254)에 의해 시간측정되는 각각의 시간 간격 동안 반복될 수 있다.

[0074] 도 6은 CIPS 유닛(204)의 하나의 구체 예를 나타내는 블록 다이어그램이다. 제시된 구체 예에서, CIPS 유닛(204)은 하위 패스 필터(266), CIPS 카운터(260), 임계치 비교기(262), 및 간격 타이머(264)를 포함한다. 제시된 구체 예에서, CIPS 유닛(154)은 각각의 프로세싱 노드(11)에 대한 퇴거 큐(예컨대 도 2의 퇴거 큐(102)로부터 퇴거된 명령에 관한 정보를 수신하도록 연결된다. CIPS 유닛(204)은 사전결정된 간격에서 퇴거된 다수의 명령을 결정할 수 있으며, CIPS 값을 추정(extrapolate)하기 위하여 이러한 값을 이용할 수 있다. CIPS 값은 따라서 명령이 실행되고 실제로 퇴거된 비율을 나타낼 수 있다(예컨대 브랜치 오예측으로 인하여 추측성으로 실행되고 퇴거되지 않은 일부 명령과 대조적임).

[0075] 퇴거 표시는 프로세싱 노드(11)로부터 수신될 수 있으며 초기에 하위 패스 필터(266)에 제공될 수 있다. 하위 패스 필터(266)는 하나 이상의 카운터로서 실행될 수 있으며, 각각의 카운터는 프로세싱 노드(11) 중 하나에 대응한다. 하위 패스 필터(266) 내 카운터는 상한 M을 가질 수 있으며, 따라서 간격 T 동안 M 이하의 퇴거가 본 구체 예에서 CIPS 유닛(204)에 의해 기록된다. 이는 활동의 짧고 갑작스러운 분출이 상위 활동으로서 잘못해석되는 것을 방지하고, 보고 프로세싱 노드(11)에 대한 최상위 동작점으로의 바람직하지 않은 전이를 방지할 수 있다.

[0076] 하위 패스 필터(266)는 간격 타이머(264)에 연결된 재설정 입력부를 포함한다. 간격이 완료되면, 간격 타이머(264)가 T 신호를 어서트할 수 있다. 재설정 입력부에서의 T 신호의 수신에 응답하여, 하위 패스 필터(266)의 각각의 카운터가 영(zero)으로 재설정될 수 있다.

[0077] CIPS 카운터(260)는 제시된 구체 예에서 또한 다수의 카운터를 포함하며, 각각은 각각의 프로세싱 노드(11)에 대응한다. 하위 패스 필터 내 카운터가 증가하는 각 시간에서, CIPS 카운터(260) 내 대응하는 카운터가 또한 증가할 수 있다. CIPS 카운터(260) 내 카운터의 증가가 가중치를 부여받을 수 있다. 따라서, 하위 패스 필터(266)의 카운터와 CIPS 카운터(260)의 카운터의 증가 사이에 항상 1:1 대응이 존재하는 것은 아닐 수 있다.

[0078] CIPS 카운터(260)의 카운터는 또한 간격 타이머(264)에 의한 T 신호의 어서팅에 응답하여 조건부로 감소될 수 있다(이는 DecAll에 제공된다). 한 구체 예에서, CIPS 카운터(260)의 주어진 카운터는, 대응하는 프로세싱 노드(11)의 명령 퇴거의 수가 현재 동작점의 예상 수와 같거나 미만인 임의 간격의 종점에서 (T 신호의 어써션(assertion)에 응답하여) 감소될 수 있다. 그렇지 않으면, 대응하는 프로세싱 노드(11)에 대한 명령 퇴거의 수가 해당 주어진 간격 동안 예상되는 퇴거의 수를 초과하는 경우, 카운터는 T 신호의 어써션에 응답하여 감소되지 않을 수 있다.

[0079] 감소 또한 가중치가 부여될 수 있다. 증가 및 감소의 가중치는 본 명세서에 기재된 전력 관리 기능의 편향(biasing)이 성능 편향성 또는 전력 편향성이 되도록 할 수 있다. 전력 관리 기능이 성능 편향성인 경우, 가중치는 증가를 강조하도록 부여되며, 이에 따라 명령 퇴거에 응답하여 CIPS 카운터(260) 내 더 높은 카운트를 초래할 수 있다. 감소된 전력 소모(즉, 전력 편향성)를 강조하는 경우, 가중치는 감소를 강조하도록 부여되며, 이에 따라 명령 퇴거에 응답하여 CIPS 카운터(260) 내 더 낮은 카운트를 초래할 수 있다. 이러한 가중치는 여러 메커니즘, 예컨대 사용자에 의한 프로그래밍, 펌웨어, 프로그래밍 융합, 등에 의해 구성될 수 있다.

[0080] CIPS 카운터(260)에 의해 발생된 카운트 값은 임계치 비교기(262)로 제공될 수 있다. 제시된 구체 예에서, 임계치 비교기(262)는 도 5에 제시된 대응하는 것과 유사하게 기능할 수 있다. 각각의 카운트 값은 상위 및 하위 활동 임계치와 비교될 수 있으며, 상위 또는 하위 활동 임계치가 이전 시간 간격에 비하여 교차된 이러한 프로세싱 노드에 대하여 동작점이 조절될 수 있다. 비교는 간격 타이머(264)에 의한 T 신호의 어써션에 응답하여, 각각의 간격에서 1회 수행될 수 있으며, 이는 비교 입력부에서 수신될 수 있다. 비교 결과는 결정 유닛(208)으로 제공될 수 있으며, 따라서 프로세싱 노드(11)의 동작점을 조절할 수 있다.

[0081] 도 7은 IPC 유닛(206)의 하나의 구체 예의 블록 다이어그램이다. 제시된 구체 예에서, IPC 유닛(206)은 다수의 프로세싱 노드(11) 각각으로부터 실행된 명령의 표시를 수신하기 위하여 연결될 수 있는 명령 카운터를 포함한다. 이는 프로세싱 노드(11)가 수퍼스칼라 프로세싱 노드인 구체 예에서 둘 이상의 실행 유닛으로부터의 실행된 명령의 수신 표시를 포함할 수 있다. 명령 카운터(270)는 각각의 프로세싱 노드(11)에 대하여 실행된 명령의 카운트를 기록할 수 있다. 각각의 프로세싱 노드에 대하여 기록된 카운트는 카운트 값으로서 임계치 비교기(272)에 제공될 수 있으며, 이는 도 5 및 6을 참조하여 앞서 논의된 대응하는 것과 유사한 방식으로 상위 및 하위 활동 임계치에 대한 비교를 수행할 수 있다. 비교 결과는 결정 유닛(208)으로 제공될 수 있으며, 이는 임의 동작점 변화가 대응하는 프로세싱 노드(11)에 대하여 수행되었는가를 결정하기 위하여 카운트 값을 사용할 수 있다. 카운팅 및 비교는 간격 타이머(274)에 의해 제어되는 간격-대-간격을 기초로 하여 수행될 수 있다. 간격 타이머(274)에 의한 T 신호의 어써션에 의해 제시되듯이, 각각의 간격이 완료될 때 명령 카운터(270)의 카운터들은 재설정될 수 있다.

[0082] 활동 모니터(202), CIPS 유닛(204), 및 IPC 유닛(206)에 의해 제공되는 비교 결과는 각각 결정 유닛(208)에 제공될 수 있다. 제공된 비교 결과를 기초로 하여, 결정 유닛(208)은 각각의 클럭 주파수, 각각의 동작 전압, 또는 두 가지 모두에 대한 변화를 야기시킴으로써 하나 이상의 프로세싱 노드(11)의 동작점에 대한 변화를 야기시킬 수 있다. 일부 구체 예에서, 결정 유닛(208)은 비교 결과에게 가중치를 제공할 수 있으며, 일부에게 다른 것에 비하여 더 많은 가중치를 제공한다. 예를 들어, 결정 유닛(208)은 IPC 유닛(206)으로부터 수신된 비교 결과에 비하여 활동 모니터(202) 및 CIPS 유닛(204)으로부터 수신된 비교 결과에게 추가 가중치를 제공할 수 있다.

[0083] 또 다른 구체 예에서, 결정 유닛(208)은 다른 비교 결과가 수신되는 경우 보팅 체계(voting scheme)에 기초하여 동작점을 변화시키는가를 결정할 수 있다. 예를 들어, 활동 모니터(202) 및 CIPS 유닛(204) 둘 모두는 상위 활동 임계치를 초과하는 활동 수준을 나타내지만 IPC 유닛(206)은 이를 나타내지 않는 경우, 결정 유닛(208)은 대응하는 프로세싱 노드의 동작점을 P-상태 P0으로 변화시킬 수 있다. 또 다른 구체 예에서, 결정 유닛(208)은 프로세싱 노드의 동작점을 변화시키는지 아닌지를 결정할 때 유닛들 중 단지 하나를 고려하도록 구성될 수 있다(예컨대, 단지 CIPS 유닛(204)으로부터의 정보만을 고려).

[0084] 일반적으로, 결정 유닛(208)은 임의 적절한 방식으로 각각의 프로세싱 노드(11)에 대한 적절한 동작점을 결정하도록 구성될 수 있다. 더욱이, 유닛들 중 단지 하나(예컨대, 활동 모니터(202))를 포함하는 전력 관리 장치(20)의 구체 예가 또한 가능하고 고려되며, 이러한 구체 예에서, SetF 및 SetV 신호의 상태를 설정하는 것과 유사한 기능이 여기에 포함될 수 있다.

[0085] **동적 성능 제어 방법:**

[0086] 도 8 및 9는 앞서 논의된 전력 관리 장치(20)의 다양한 예가 작동될 수 있는 방법을 나타낸다. 본 방법 실시예는 단일 프로세싱 노드를 참조하여 본 명세서에서 설명된다. 그렇지만, 앞서 논의된 것과 같이, 전력 관리 장치(20)는 다수의 서로 다른 프로세싱 노드에 대하여 본 명세서에 기재된 방법의 여러 구체 예를 동시에 수행할 수 있음에 주목하라.

[0087] 도 8은 전력 관리 장치 작동 방법의 하나의 구체 예의 흐름도이다. 제시된 구체 예에서, 방법(700)은 프로세싱 노드에서의 애플리케이션 활성의 기록으로 시작한다(블록 705). 활동 수준은 하위 및 상위 임계치와 비교될 수 있다. 활동 수준이 상위 임계치를 초과하지만(블록 710, 예) 프로세싱 노드가 이미 상위 동작점에서 작동하는 경우(예컨대, 최상위 P-상태, P0; 블록 715, 예), 어떠한 변화도 없으며 애플리케이션 활성의 기록이 지속된다(블록 705).

[0088] 활동 수준이 상위 임계치를 초과하고(블록 710, 예) 동작점이 상위 동작점 미만(블록 715, 예)인 경우, 해당 프로세싱 노드의 동작점은 적어도 그 다음 시간 간격 동안 상위 동작으로 전이될 수 있다(블록 720). 새로운 동작점으로의 전이 이후, 활성이 다시 기록된다(블록 725). 활동 수준이 상위 임계치 상부에서 유지하는 경우(블록 730, 아니오), 프로세싱 노드의 동작이 상위 동작점에서 지속될 수 있다. 간격에 대한 활동 수준이 상위 임계치

미만(블록 730, 예)이자만 하위 임계치 상부에서 유지되고(블록 735, 아니오) 현재 동작점이 초기 동작점과 동일하지 않은 경우(블록 770, 아니오), 프로세싱 노드는 OS 또는 또 다른 소프트웨어에 의한 요구에 따라 중간 동작점으로 전이된다(블록 775).

[0089] 블록 710에서, 활동 수준이 상위 임계치 미만(블록 710, 아니오)이고 하위 임계치 초과(블록 740, 아니오)인 경우, 방법은 블록 705의 애플리케이션 활성의 기록으로 되돌아간다. 블록 740에서 활동 수준이 하위 임계치 미만(블록 740, 예)이고, 동작점이 하위 동작점에 있는 경우(블록 745, 아니오), 방법은 다시 블록 705의 애플리케이션 활성의 기록으로 되돌아간다. 그렇지만, 블록 740에서 활동 수준이 하위 임계치 미만(블록 740, 예)이고, 동작점이 하위 동작점 초과(블록 745, 예)인 경우, 프로세싱 노드는 하위 동작점으로 전이된다(블록 750). 하위 동작점으로의 전이 이후, 애플리케이션 활성이 기록되며(블록 755), 활동 수준이 하위 임계치 미만으로 유지하는 경우(블록 760, 아니오) 방법은 블록 755로 되돌아가고 프로세싱 노드 및 동작점은 변하지 않은 채로 유지한다. 그렇지만, 활동 수준이 하위 임계치를 초과하지만(블록 760, 예), 상위 임계치를 초과하지 않는 경우(블록 765, 아니오), 블록 770은 '아니오'를 초래할 것이며 프로세싱 노드는 운영 체제 또는 또 다른 소프트웨어에 의해 요구되는 중간 동작점으로 전송될 것이다.

[0090] 블록 735에서, 활동 수준이 하위 임계치 미만인 경우(블록 735, 예), 방법은 블록 745로 전이할 수 있다. 프로세싱 노드가 하위 동작점 이외의 동작점에서 작동하는 경우(블록 745, 예), 적어도 그 다음 시간 간격 동안 하위 동작점으로 전이될 수 있다(블록 750). 이러한 전이에 후속하여, 방법은 블록 755로부터 계속하여 전술한 시퀀스를 따를 수 있다. 그렇지 않은 경우, 프로세싱 노드가 이미 하위 동작점에서 작동중인 경우(블록 745, 아니오), 방법은 블록 705로 되돌아갈 수 있다.

[0091] 블록 765에서, 활동 수준이 상위 임계치 초과인 경우(블록 765, 예), 방법은 블록 715로 되돌아갈 수 있다. 프로세싱 노드가 상위 동작점 이외의 동작점에서 작동하는 경우(블록 715, 예), 적어도 그 다음 시간 간격 동안 상위 동작점으로 전이될 수 있다(블록 720). 이러한 전이에 후속하여, 방법은 블록 725로부터 계속하여 전술한 시퀀스를 따를 수 있다. 그렇지 않은 경우, 프로세싱 노드가 이미 상위 동작점에서 작동중인 경우(블록 715, 아니오), 방법은 블록 705로 되돌아갈 수 있다.

[0092] 도 9는 다수의 시간 간격 동안 방법의 하나의 구체 예의 작동을 나타내는 다이어그램이다. 설명의 간편성을 위하여, 본 방법은 단지 하나의 중간 동작점을 참조하여 설명된다. 그렇지만, 전술한 바와 같이(예컨대, 표 1 참조), 다수의 중간 동작점이 구현될 수 있는 구체 예가 가능하고 고려되는 점이 이해된다. 또한, 전술한 바와 같이, 도 9의 실시예는 단일 프로세싱 노드를 참조하여 제시되지만, 기재된 방법은 동일한 전자 시스템 내의 다수의 고유 프로세싱 노드에 대하여 동시에 수행될 수 있음이 이해된다.

[0093] 제시된 실시예에서 T0 직전의 지점에서, 프로세싱 노드에 대한 활동 수준은 하위 임계치 미만이다. 따라서, 실시예의 제1 전체 시간 간격 T0 내지 T1 동안, 프로세싱 노드의 동작점은 하위 동작점에 설정된다. 이러한 제1 간격 동안, 활동 수준은 증가하여 하위 임계치 상부의 지점에서 유지한다. 따라서, 그 다음 간격 T1 내지 T2 동안, 프로세싱 노드의 동작점은 중간 동작점에 설정된다.

[0094] 간격 T1 내지 T2 동안, 프로세싱 노드의 활동 수준은 상위 임계치를 초과하는 지점까지 증가한다. 따라서, T2 내지 T3의 간격 동안, 프로세싱 노드는 상위 동작점에서 작동된다. 활동 수준은 이러한 간격 동안 상위 임계치 상부에서 유지하며, 따라서 프로세싱 노드는 간격 폭 T3 내지 T4 동안 상위 동작점에서의 작동을 유지한다.

[0095] T3 내지 T4 사이의 간격에서, 활동 수준은 간격의 일부 동안 상위 임계치 미만으로 하강하고 그 후 다시 상위 임계치 상부의 지점으로 상승한다. 브랜치 오예측, 캐쉬 미스, 방해, 또는 또 다른 이유로 파이프라인이 플러시되거나(flushed) 또는 스톨(stall)되도록 할 수 있는 또 다른 이벤트와 같은 여러 이벤트가 프로세싱 전반에서 순간순간 하강을 나타낼 수 있으나 프로세싱 워크로드 자체의 하강을 반드시 표시하는 것은 아니다. 따라서, 본 명세서에 기재된 방법을 수행하도록 구성된 전력 관리 장치는 이러한 상황을 고려하도록 구성될 수 있다. 예를 들어, 실제 활동 수준에 기초하여 이전 간격의 종점에서 그 다음 간격에 대한 동작점을 설정하는 대신에, 전력 관리 장치는 대신 즉시 간격 및 하나 이상의 이전 간격 동안 평균 활동 수준을 결정할 수 있다(예컨대, 예를 들어 도 5를 참조하여 앞서 기재한 구체 예와 같음). 따라서, 전반적으로 프로세서의 순간 하강을 초래할 수 있으나 그 밖의 경우 어떠한 감소된 워크로드로 나타내지 않을 수 있는 이벤트가 효과적으로 필터링되어 동작점이 자신의 현재 레벨에서, 또는 이러한 특별한 경우 상위 지점에서 잔류할 수 있다. 유사하게, 현재 그리고 하나 이상의 이전 간격 동안 평균 활동 수준을 사용하는 것은, 그렇지 않은 경우의 하위(예컨대, 메모리-집중식) 워크로드에 대한 활동의 갑작스러운 분출을 효과적으로 필터링할 수 있다. 이러한 기법을 사용하는 것은 본 명세서에 기재된 바와 같이 전력 관리 장치가 동작점을 프로세서 워크로드에 더욱 효과적으로 일치시키면서 한편 변

칙 이벤트가 동작점에 영향을 미치는 것을 방지할 수 있게 한다.

[0096] T4에서 시작하는 간격들에 대하여, 활동 수준은 여러 번 하강하고 다시 상승하며, 동작점도 이를 따른다. 전술한 바와 같이, 간격은 짧은 기간일 수 있다. 한 구체 예에서, 간격은 10 마이크로초 내지 100 마이크로초 범위일 수 있으나, 또 다른 기간의 간격(10 마이크로초보다 더 짧은 기간 포함)이 가능하며 고려된다. 짧은 간격을 사용하는 것은 프로세싱 노드의 동작점이 자신의 대응하는 활동 수준을 정밀하게 반영하도록 할 수 있다. 이는 프로세싱 노드가, 컴퓨팅-집중적 워크로드가 완료된 이후 상당한 시간 동안 최상위 동작점에서 잔류하거나, 또는 워크로드 요구의 증가에 후속하는 상당한 시간 동안 최하위 동작점에 잔류하는 것을 방지할 수 있다.

[0097] 도 9는 또한 제2 기간, T0-OS 내지 T1-OS의 복수의 간격 중 하나를 나타낸다. 여기서 제시된 제1 간격 T0-T10의 수 내지 제2 간격의 수는 예시적인 것이며, 제1 간격 대 제2 간격의 비율은 변할 수 있음을 주목하라. 여기에 제시된 제2 간격 중 예시적인 것은 운영 체제 소프트웨어가 이 간격 동안 프로세싱 노드의 활동 수준을 모니터링할 수 있는 간격일 수 있다. 이러한 간격은 제1 간격보다 상당히 더 긴 기간일 수 있으며, 일부 경우, 수십배 더 길 수 있다. 운영 체제는 이러한 제2 간격 동안 활동 수준을 모니터링할 수 있으며 이에 따라 동작점을 조절할 수 있다. 그렇지만, 여기서 논의된 전력 관리 장치의 여러 구체 예는 활동 수준이 상위 활동 임계치보다 크거나 같을 때, 또는 하위 활동 임계치보다 작거나 같을 때 각각의 프로세싱 노드에 대한 동작점의 제어를 유지할 수 있다. 이러한 두 가지 임계 값 사이에 포함되는 활동 수준에 대하여, 전력 관리 장치는 운영 체제가 프로세서의 각각의 프로세싱 노드(또는 프로세서 코어)에 대한 동작점을 설정하도록 할 수 있다. 그렇지만, 제1 간격 동안, 활동 수준이 상위 활동 임계치보다 크거나 같은 것으로 또는 하위 활동 임계치보다 작거나 같은 것으로 검출되는 경우, 전력 관리 장치는 제2 간격 중 하나 동안 임의 시간에서 운영 체제에 의해 이루어진 동작점 선택을 무시할 수 있다.

[0098] **컴퓨터 접속가능 저장 매체:**

[0099] 다음으로 도 10을 참조하면, 시스템(10)을 대표하는 데이터베이스(405)를 포함하는 컴퓨터 접속가능 저장 매체(400)의 블록 다이어그램이 제시된다. 일반적으로, 컴퓨터 접속가능 저장 매체(400)는 명령 및/또는 데이터를 컴퓨터에게 제공하기 위해 사용하는 동안 컴퓨터에 의해 접속가능한 임의 비-일시적 저장 매체를 포함할 수 있다. 예를 들어, 컴퓨터 접속가능 저장 매체(400)는 저장 매체 예컨대 자기 매체 또는 광학 매체, 예컨대, 디스크(고정형 또는 이동형), 테이프, CD-ROM, 또는 DVD-ROM, CD-R, CD-RW, DVD-R, DVD-RW, 또는 블루-레이(Blu-Ray)를 포함할 수 있다. 저장 매체는 범용 직렬 버스(USB) 인터페이스 등과 같은 주변 인터페이스를 통하여 접속가능한 휘발성 또는 비-휘발성 메모리 매체 예컨대 RAM (예를 들어, 동기식 동적 RAM(SDRAM), 이중 데이터 속도(DDR, DDR2, DDR3, 등) SDRAM, 저-전력 DDR(LPDDR2, 등) SDRAM, 램버스(Rambus) DRAM (RDRAM), 정적 RAM (SRAM), 등), ROM, 플래시 메모리, 비-휘발성 메모리(예컨대 플래시 메모리)를 더욱 포함할 수 있다. 저장 매체는 마이크로전자기계 시스템(microelectromechanical system, MEMS), 뿐만 아니라 네트워크 및/또는 무선 연결과 같은 통신 매체를 통하여 접속가능한 저장 매체를 포함할 수 있다.

[0100] 일반적으로, 컴퓨터 접속가능 저장 매체(400)에서 전송되는 시스템(10)의 데이터베이스(405)는 프로그램에 의해 판독될 수 있고 시스템(10)을 포함하는 하드웨어를 제작하기 위하여 직접 또는 간접적으로 사용될 수 있는 데이터베이스 또는 또 다른 데이터 구조일 수 있다. 예를 들어, 데이터베이스(405)는 고급 설계 언어(high level design language, HDL) 예컨대 베리로그(Verilog) 또는 VHDL에서 하드웨어 기능의 행위-수준 설계(behavioral-level description) 또는 레지스터-전송 수준(register-transfer level, RTL) 설계일 수 있다. 설계(description)는 합성 라이브러리로부터 게이트 목록을 포함하는 넷리스트(netlist)를 생성하기 위하여 상기 설계를 합성할 수 있는 합성 툴에 의해 판독될 수 있다. 넷리스트는 시스템(10)을 포함하는 하드웨어의 기능을 또한 나타내는 한 세트의 게이트를 포함한다. 넷리스트는 그 후 마스크에 적용될 기하학적 형상을 묘사하는 데이터 세트를 생성하기 위하여 위치되고 라이팅될 수 있다. 마스크는 그 후 다양한 반도체 제조 단계에서 사용되어 시스템(10)에 대응하는 반도체 회로 또는 회로들을 생성할 수 있다. 그 대신에, 컴퓨터 접속가능 저장 매체(400)의 데이터베이스(405)는 요구에 따라 넷리스트(합성 라이브러리가 있거나 없음) 또는 데이터 세트일 수 있다.

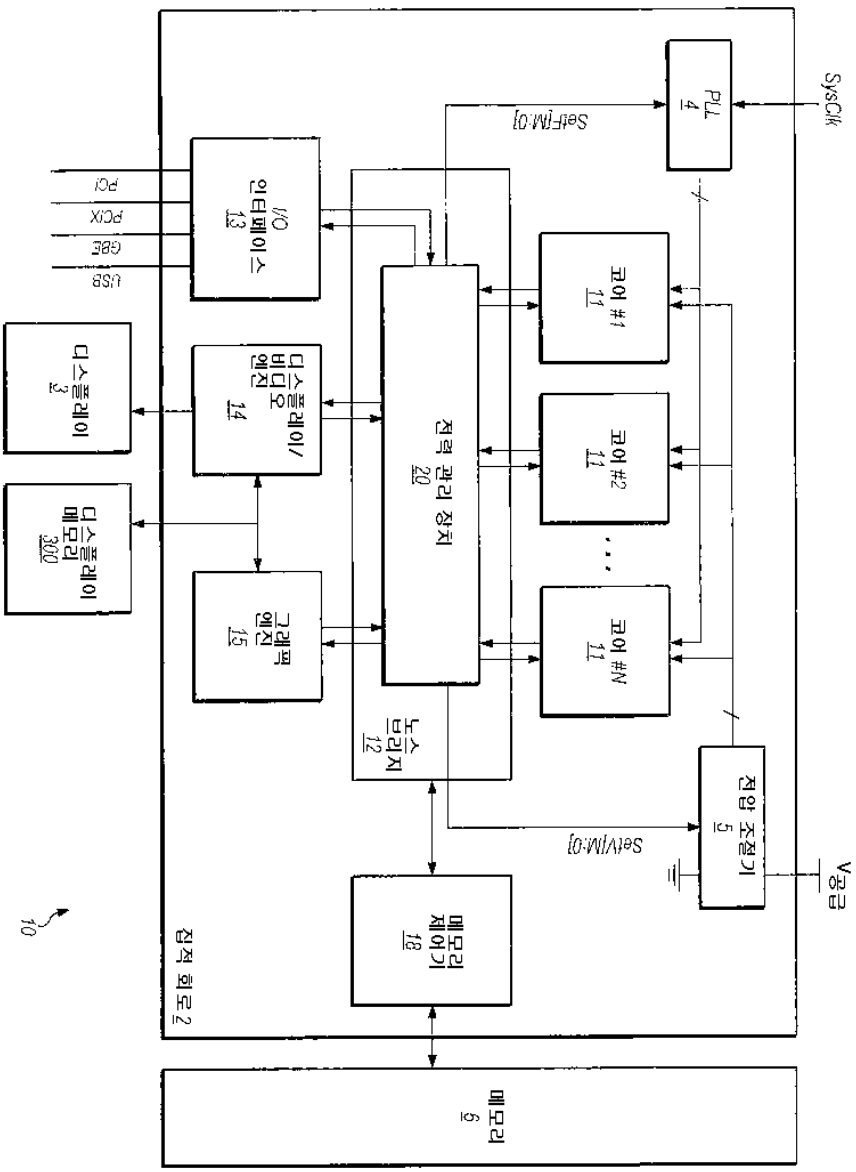
[0101] 컴퓨터 접속가능 저장 매체(400)는 시스템(10)의 표현을 전송하며, 다른 구체 예는 요구에 따라, IC(2), 임의 세트의 에이전트(예컨대, 프로세싱 노드(11), I/O 인터페이스(13), 전력 관리 장치(20), 등) 또는 에이전트의 일부(예컨대, 활동 모니터(202), CIPS 유닛(204), 등)을 포함하는, 시스템(10)의 임의 일부의 표현을 전송할 수 있다.

[0102] 본 발명이 특정 구체 예를 참조하여 설명되었으나, 구체 예는 예시적인 것이며 본 발명이 여기에 제한되지 않음

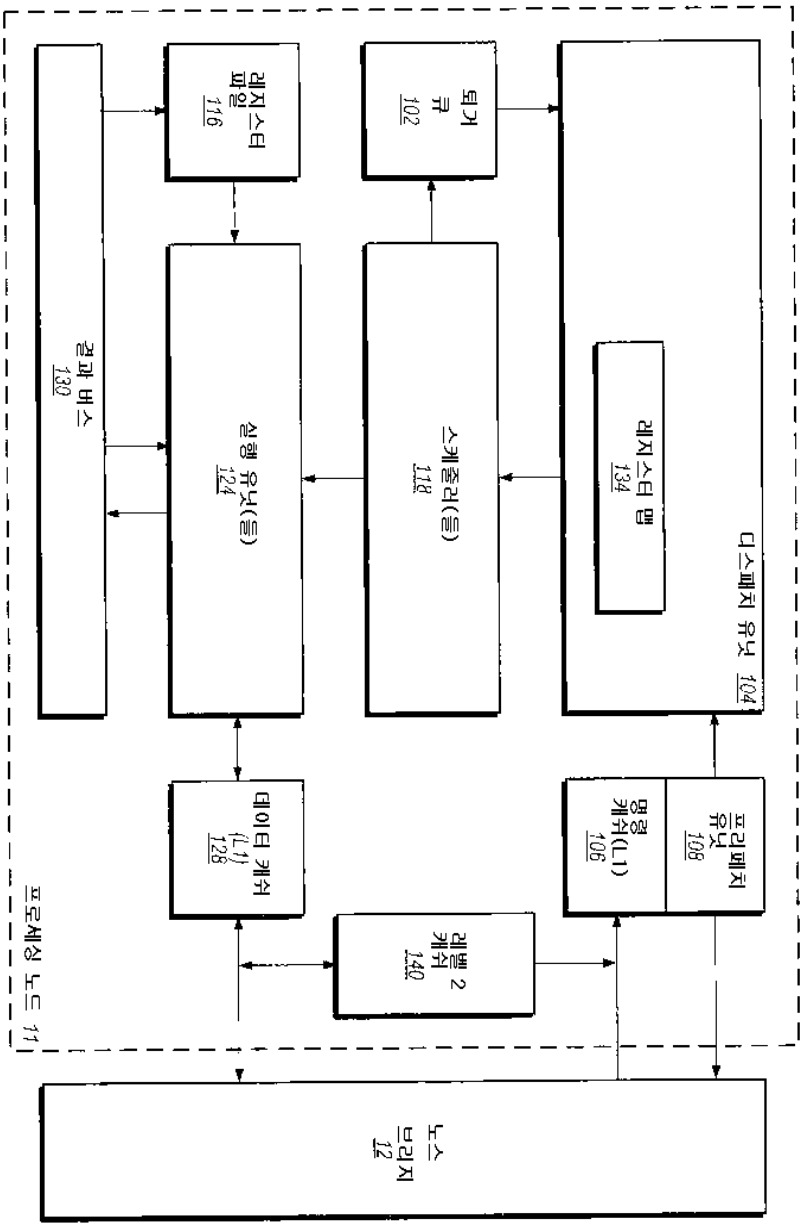
이 이해될 것이다. 기재된 구체 예에 대한 어떠한 변화, 변형, 부가사항, 및 개선사항도 가능하다. 이러한 변화, 변형, 부가, 및 개성사항은 이하의 청구항에 상술된 바와 같이 본 발명의 범위에 포함될 수 있다.

도면

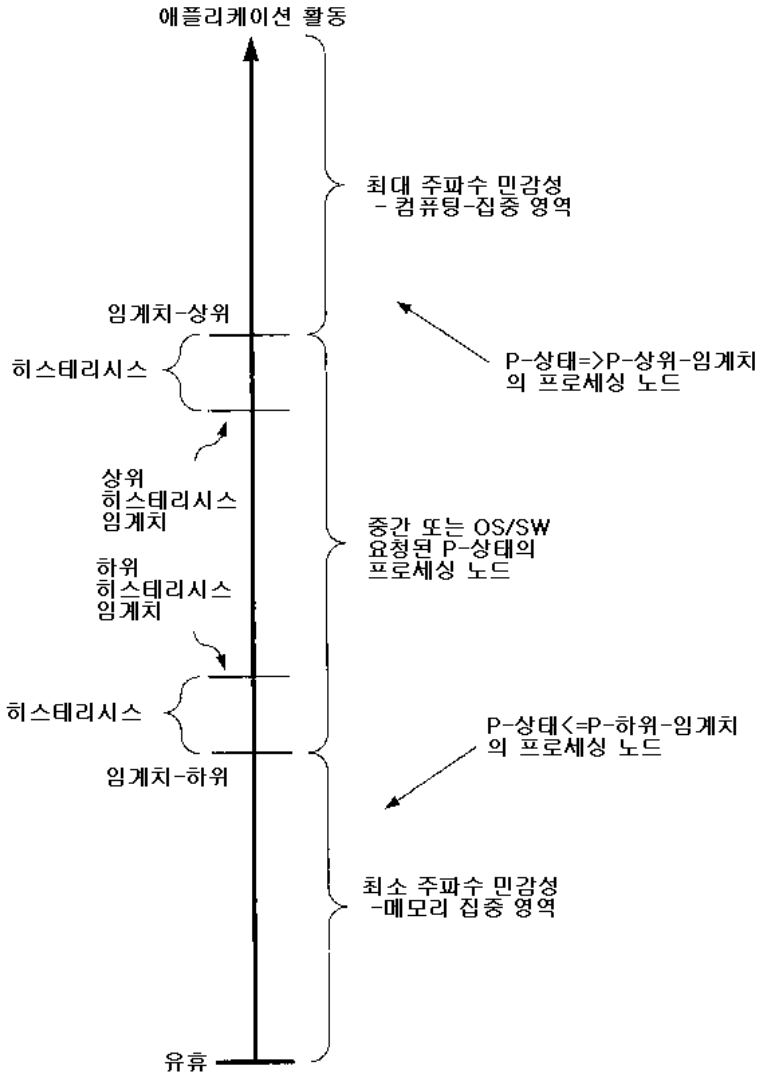
도면1



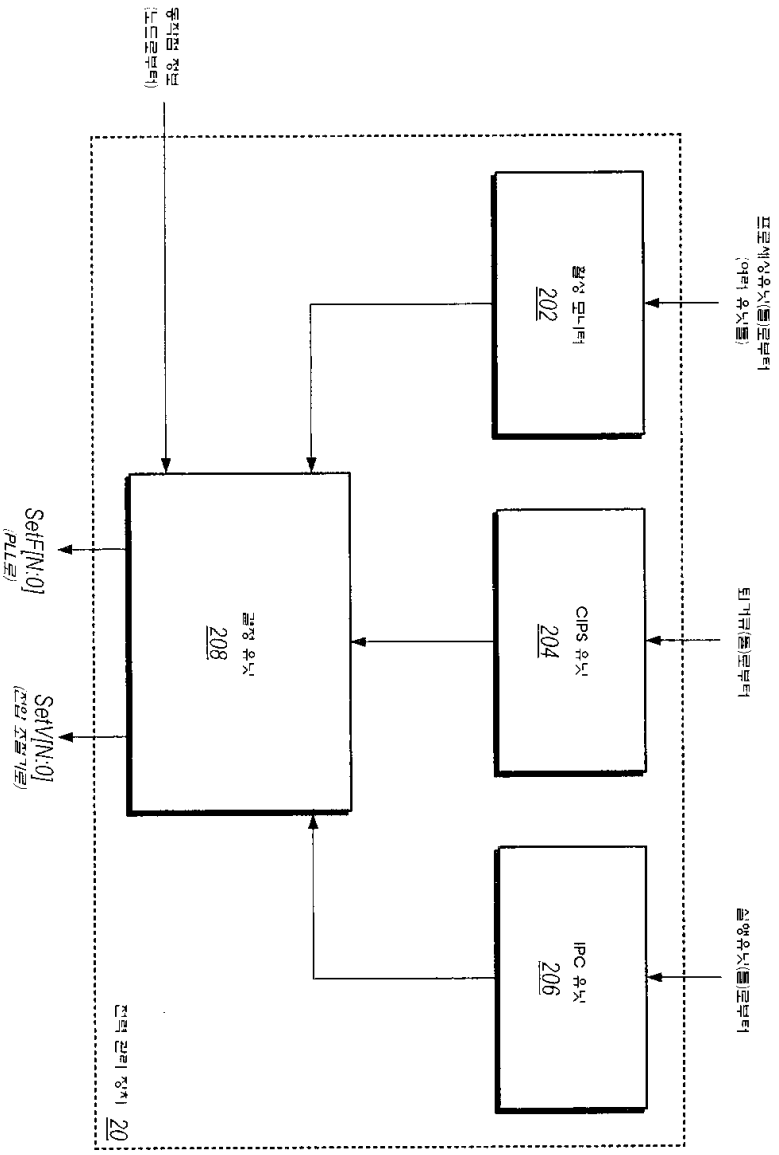
도면2


프로세싱 노드 111
디스패치 유닛 104
레지스터 맵 134
퇴거 큐 102
레지스터 파일 116
결과 버스 130
실행 유닛(들) 124
스케줄러(들) 118
프리페치 유닛 108
명령 캐쉬(L1) 106
데이터 캐쉬 (L1) 126
레벨 2 캐쉬 140
노스 브리지 12

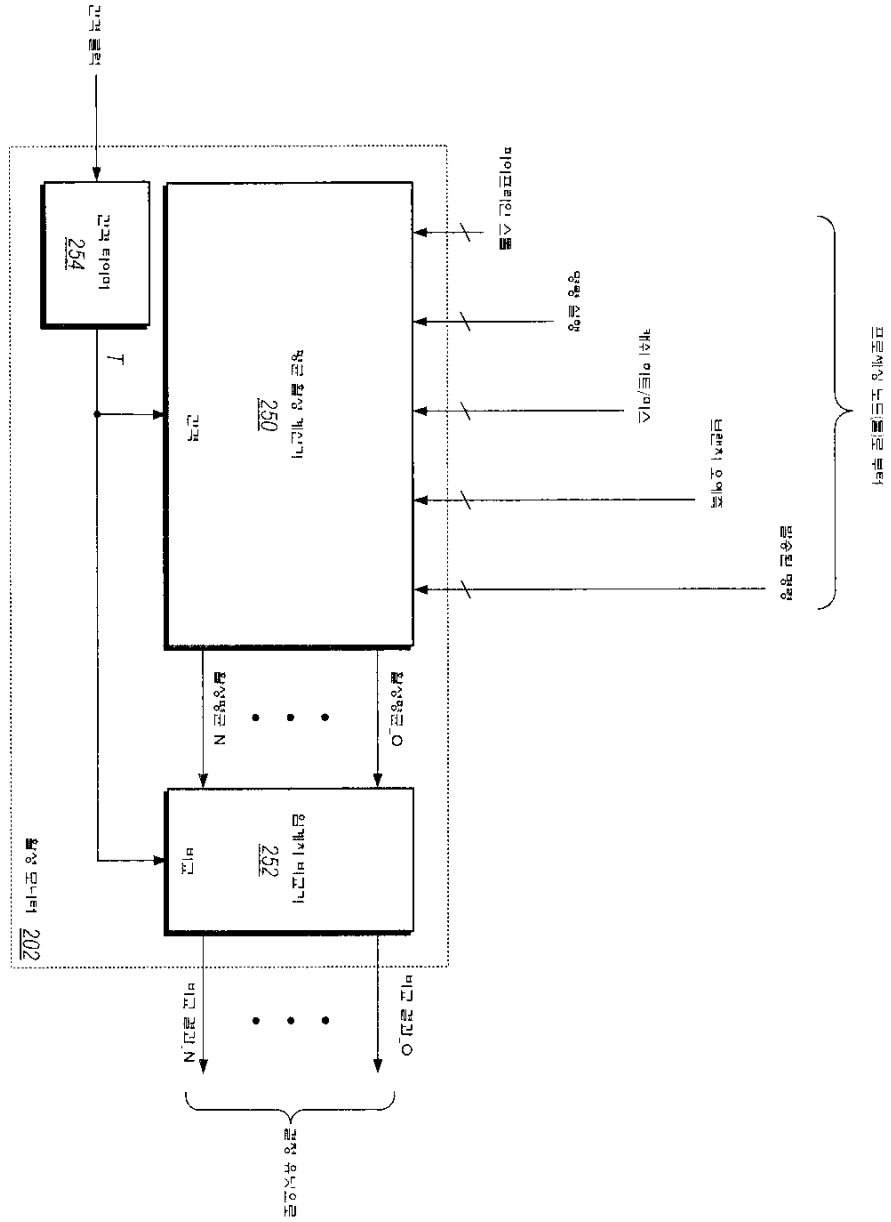
도면3


애플리케이션 활동
최대 주파수 민감성
-컴퓨팅-집중 영역
임계치-상위
히스테리시스
P-상태=>P-상위-임계치
의 프로세싱 노드
상위
히스테리시스
임계치
중간 또는 OS/SW
요청된 P-상태의
프로세싱 노드
하위
히스테리시스
임계치
히스테리시스
P-상태<=P-하위-임계치
의 프로세싱 노드
임계치-하위
최소 주파수 민감성
-메모리 집중 영역
유휴

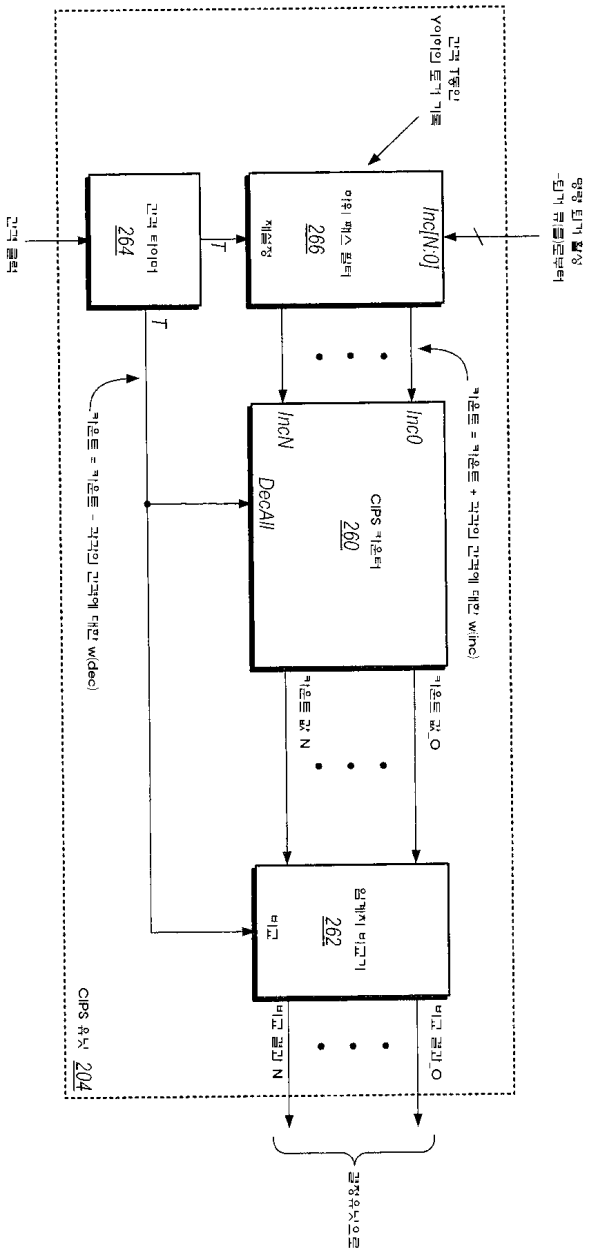
도면4


동작점 정보
(노드로부터)
프로세싱유닛(들)로부터
(여러 유닛들)
활성 모니터
202
CIPS 유닛
204
IPC 유닛
206
실행유닛(들)로부터
결정 유닛
208
SetF[N:0]
(PLL로)
SetV[N:0]
(전압 조절기로)
전력 관리 장치
20

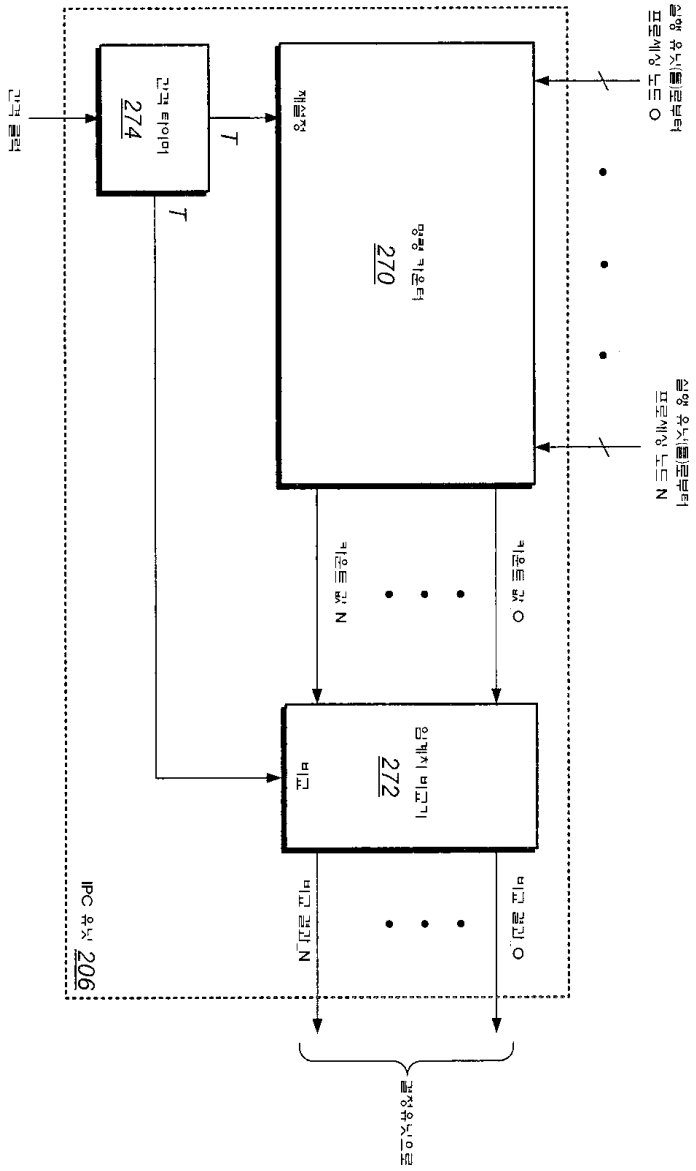
도면5



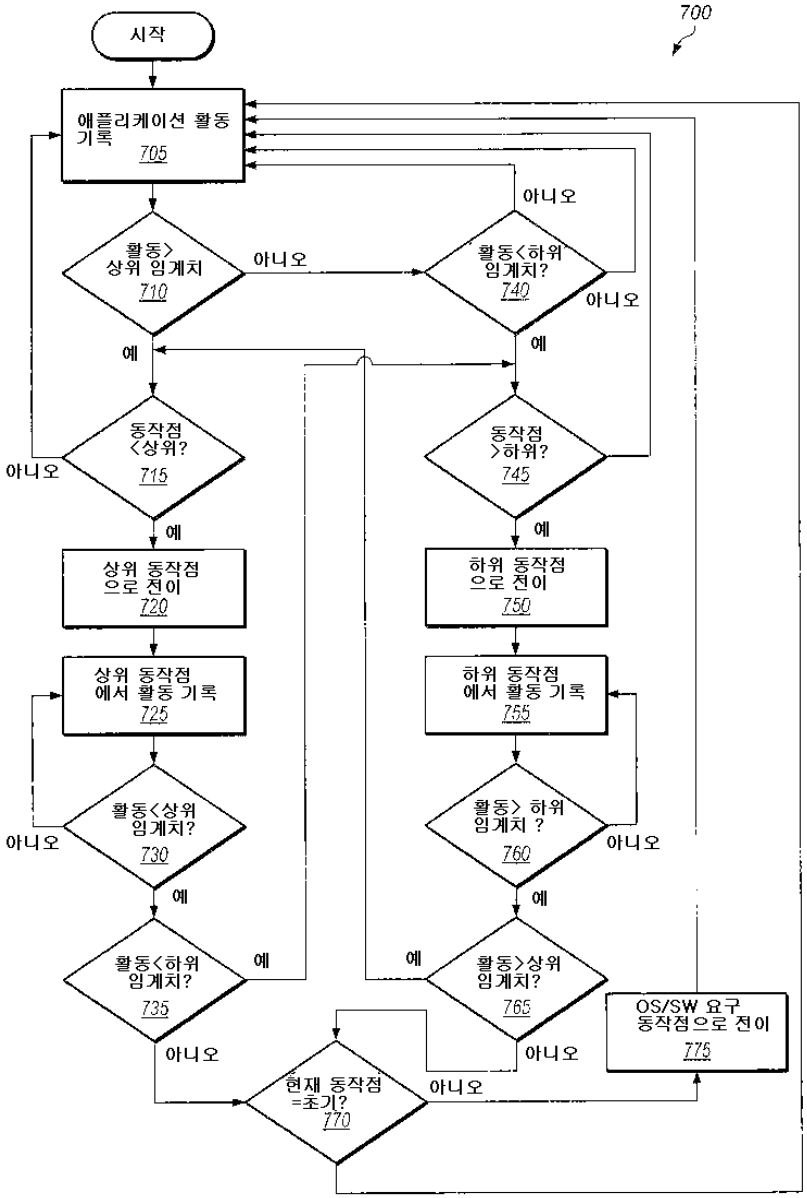
도면6



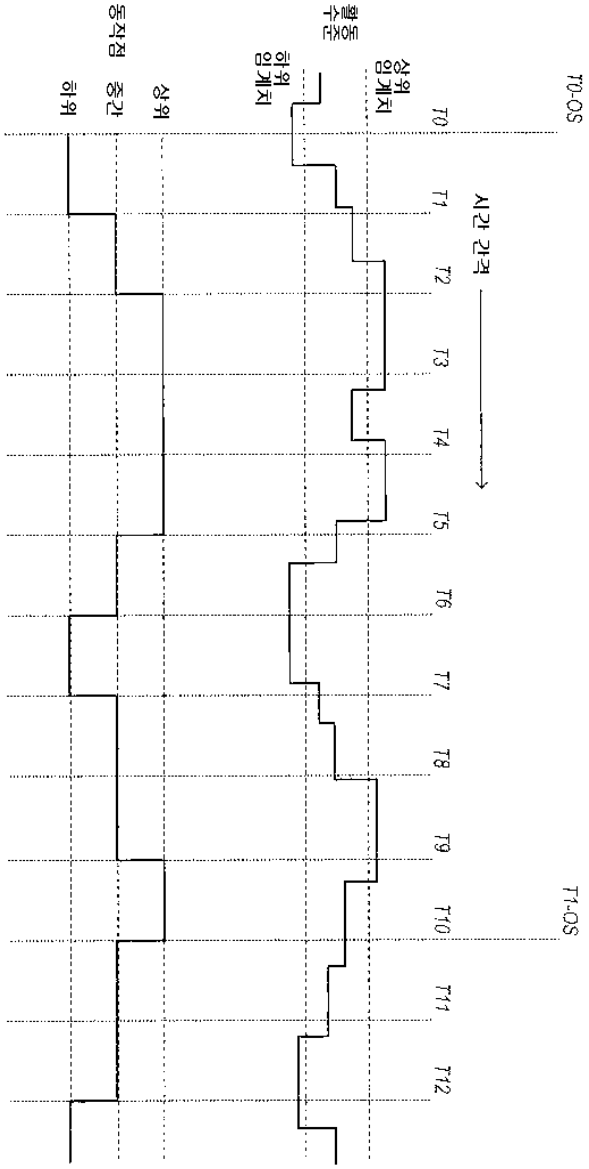
도면7


실행 유닛(들)로부터 프로세싱 노드 O
실행 유닛(들)로부터 프로세싱 노드 N
제설정
명령 카운터
270
카운트 값 N
카운트 값 O
임계치 비교기
272
비교
비교 결과 N
비교 결과 O
결정유닛으로
IPC 유닛 206
근거 클럭
근거 타이머
274
T
T

도면8


700
시작
애플리케이션 활동 기록
705
활동> 상위 임계치
710
아니오
예
활동<하위 임계치?
740
아니오
예
동작점 <상위?
715
아니오
예
동작점 >하위?
745
예
상위 동작점으로 전이
720
하위 동작점으로 전이
750
상위 동작점에서 활동 기록
725
하위 동작점에서 활동 기록
755
활동<상위 임계치?
730
아니오
예
활동> 하위 임계치 ?
760
아니오
예
활동<하위 임계치?
735
예
아니오
활동>상위 임계치?
765
예
아니오
OS/SW 요구 동작점으로 전이
775
현재 동작점 =초기?
770
아니오

도면9


T0-OS
T1-OS
시간 간격
T0
T1
T2
T3
T4
T5
T6
T7
T8
T9
T10
T11
T12
하위
중간
상위
동작점
하위 임계치
상위 임계치
홀드 구간

도면10

