



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2018년03월21일
(11) 등록번호 10-1840739
(24) 등록일자 2018년03월15일

(51) 국제특허분류(Int. Cl.)
G06F 9/50 (2018.01) G06F 1/32 (2006.01)
G06F 17/50 (2006.01)
(52) CPC특허분류
G06F 9/5094 (2013.01)
G06F 1/3203 (2013.01)
(21) 출원번호 10-2016-7019080
(22) 출원일자(국제) 2014년12월15일
심사청구일자 2017년08월14일
(85) 번역문제출일자 2016년07월14일
(65) 공개번호 10-2016-0102460
(43) 공개일자 2016년08월30일
(86) 국제출원번호 PCT/US2014/070279
(87) 국제공개번호 WO 2015/095016
국제공개일자 2015년06월25일
(30) 우선권주장
14/137,914 2013년12월20일 미국(US)
(56) 선행기술조사문헌
US20100153954 A1
US20110265092 A1
US20130047166 A1
Jin Cui et al. 'A Fast High-Level
Event-Driven Thermal Estimator for Dynamic
Thermal Aware Scheduling'. IEEE Trans on
Computer-Aided Design of Integrated Circuits
and Systems, Vol.31, No.6, pp.904-917.

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스
드라이브 5775
(72) 발명자
사하, 주희
미국 92121 캘리포니아주 샌 디에고 모어하우스
드라이브 5775
박, 희준
미국 92121 캘리포니아주 샌 디에고 모어하우스
드라이브 5775
(뒷면에 계속)
(74) 대리인
특허법인 남앤드남

전체 청구항 수 : 총 13 항

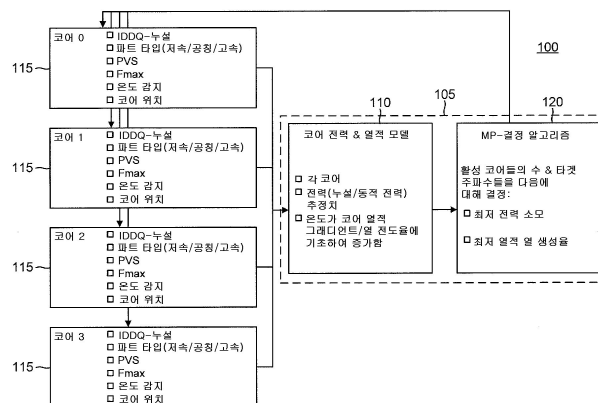
심사관 : 유진태

(54) 발명의 명칭 멀티-코어 동적 작업부하 관리

(57) 요약

코어들(210, 225, 305)에 대한 전류 동작 특징들에 기초하여 복수의 코어들에 대한 작업들(200)을 스케줄링하는 동적 스케줄러가 제공된다. 전류 동작 특징들은 분석 모델에 기초하여 각각의 코어에 대해 예상되는 누설 전류를 포함한다.

대표도



(52) CPC특허분류

G06F 1/329 (2013.01)

G06F 1/3293 (2013.01)

G06F 17/5009 (2013.01)

Y02D 10/122 (2018.01)

Y02D 10/22 (2018.01)

Y02D 10/24 (2018.01)

(72) 발명자

투, 알렉스 광-수안

미국 92121 캘리포니아주 샌 디에고 모어하우스 드
라이브 5775

모리슨, 토마스 앤드류

미국 92121 캘리포니아주 샌 디에고 모어하우스 드
라이브 5775

수에, 타오

미국 92121 캘리포니아주 샌 디에고 모어하우스 드
라이브 5775

알톤, 로날드 프랑크

미국 92121 캘리포니아주 샌 디에고 모어하우스 드
라이브 5775

명세서

청구범위

청구항 1

셀룰러 폰을 위한 시스템으로서,

복수의 코어들(cores) — 각각의 코어는 개별적인 온도 그래디언트(gradient) 및 개별적인 열 전도율(thermal conductivity)을 가짐 —;

상기 코어들의 고유 누설 전류들(native leakage currents) 및 현재 온도들에 기초하여 각각의 코어에 대한 누설 전력을 분석적으로 모델링하도록 그리고 각각의 코어에 대한 현재 전압 및 클럭 주파수에 기초하여 각각의 코어에 대한 동적 전력을 모델링하도록 그리고 코어의 누설 전력 및 코어의 동적 전력의 합에 기초하여 각각의 코어에 대한 총 전력 소모를 결정하도록 구성되는 분석적 코어 전력 및 열적 모델러(analytic core power and thermal modeler); 및

각각의 작업(task)이 유희-사용 케이스인지 또는 저-전력 사용 케이스인지 여부에 기초하여 상기 코어들에 대한 작업들을 스케줄링하도록 추가로 구성되는 동적 스케줄러를 포함하고,

사용자는 유희-사용 케이스들 동안 상기 셀룰러 폰과 상호작용하지 않고, 상기 셀룰러 폰은 상기 유희-사용 케이스들 동안 전화 또는 문자를 수신하지 않고, 상기 동적 스케줄러는 상기 유희-사용 케이스에 대해 가장 낮은 누설 전력을 갖는 코어를 위해 선택하도록 그리고 상기 저-전력 사용 케이스에 대해 가장 낮은 총 전력 소모를 갖는 코어를 위해 선택하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 2

제 1 항에 있어서,

상기 동적 스케줄러는 추가로, 코어의 개별적인 온도 그래디언트 및 코어의 개별적인 열 전도율로부터 선택되는 각각의 코어에 대한 열 파라미터에 기초하여 상기 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 3

제 2 항에 있어서,

각각의 코어에 대한 상기 열 파라미터는 상기 코어의 개별적인 열 전도율인, 셀룰러 폰을 위한 시스템.

청구항 4

제 2 항에 있어서,

상기 동적 스케줄러는 추가로, 다이 상의 각각의 코어에 대한 위치에 기초하여 상기 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 5

제 4 항에 있어서,

상기 동적 스케줄러는 추가로, 상기 다이 상의 다른 열원들에 대한 코어 위치들의 근접도에 기초하여 상기 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 6

제 4 항에 있어서,

상기 동적 스케줄러는 추가로, 최근에 스케줄링된 코어 위치들에 대한 상기 코어 위치들의 근접도에 기초하여 상기 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 7

제 2 항에 있어서,

상기 동적 스케줄러는 추가로, 각각의 코어의 전력 조절기에 대한 효율에 기초하여 상기 코어들에 대한 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 8

제 7 항에 있어서,

상기 동적 스케줄러는 추가로, 각각의 작업에 대해 예상되는 작업 시간 지속기간에 기초하여 상기 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 9

제 7 항에 있어서,

상기 동적 스케줄러는 추가로, 각각의 코어에 대한 총 전력 소모와 각각의 코어의 전력 조절기에 대한 효율에 의존하는 시스템 전력 소모를 결정하도록 구성되고, 그리고 상기 동적 스케줄러는 추가로, 상기 시스템 전력 소모를 최소화하기 위해서 상기 작업들을 스케줄링하도록 구성되는, 셀룰러 폰을 위한 시스템.

청구항 10

복수의 코어들을 갖는 멀티-코어 프로세서를 포함하는 셀룰러 폰을 위한 방법으로서, 상기 방법은 동적 스케줄러에 의해 수행되는 단계들을 포함하고, 상기 단계들은:

상기 복수의 코어들의 각각의 코어에 대한 예상 누설 전력을 결정하는 단계 — 상기 결정은 각각의 코어에 대한 고유 누설 전류, 각각의 코어의 온도, 및 각각의 코어에 대한 동작 전압에 의존하는 분석적 모델에 응답하고, 각각의 코어는 개별적인 온도 그래디언트 및 개별적인 열 전도율을 가짐 —;

각각의 코어에 대한 동작 전압에 기초하여 각각의 코어에 대한 동적 전력 소모를 결정하는 단계;

코어의 예상 누설 전력 및 코어의 동적 전력 소모에 기초하여 각각의 코어에 대한 총 전력 소모를 결정하는 단계;

어느 코어가 가장 낮은 예상 누설 전력을 갖는지에 기초하여 유휴-사용-케이스 작업을 위해 상기 코어들 중 하나를 스케줄링 하는 단계 — 사용자는 상기 유휴-사용-케이스 작업 동안 상기 셀룰러 폰과 상호작용하지 않고, 상기 셀룰러 폰은 상기 유휴-사용-케이스 작업 동안 전화들 또는 문자들을 수신하지 않음 —;

어느 코어가 가장 낮은 총 전력 소모를 갖는지에 기초하여 저-전력 사용-케이스 작업을 위해 상기 코어들 중 적어도 하나를 스케줄링하는 단계; 및

코어의 개별적인 온도 그래디언트 및 코어의 개별적인 열 전도율로부터 선택되는 각각의 코어에 대한 열 파라미터에 기초하여 그리고 어느 코어가 가장 낮은 온도를 갖는지에 기초하여 고-성능 사용-케이스 작업을 위해 상기 코어들 중 적어도 하나를 스케줄링하는 단계를 포함하는, 복수의 코어들을 갖는 멀티-코어 프로세서를 포함하는 셀룰러 폰을 위한 방법.

청구항 11

제 10 항에 있어서,

각각의 코어에 대한 상기 열 파라미터는 코어의 개별적인 열적 그래디언트이고, 상기 단계들은 어느 코어가 가장 낮은 개별적인 열적 그래디언트를 갖는지를 결정하는 단계를 더 포함하고, 상기 고-성능 사용-케이스 작업을 위해 상기 코어들 중 적어도 하나를 스케줄링하는 단계는 추가로 어느 코어가 가장 낮은 개별적인 열적 그래디언트를 갖는지에 기초하는, 복수의 코어들을 갖는 멀티-코어 프로세서를 포함하는 셀룰러 폰을 위한 방법.

청구항 12

제 10 항에 있어서,

각각의 코어에 대한 상기 열 파라미터는 상기 코어의 개별적인 열 전도율이고, 상기 단계들은 어느 코어가 가장

높은 개별적인 열 전도율을 갖는지를 결정하는 단계를 더 포함하고, 상기 고-성능 사용-케이스 작업에 대해 상기 코어들 중 적어도 하나를 스케줄링하는 단계는 추가로 어느 코어가 가장 높은 개별적인 열 전도율을 갖는지에 기초하는, 복수의 코어들을 갖는 멀티-코어 프로세서를 포함하는 셀룰러 폰을 위한 방법.

청구항 13

제 10 항에 있어서,

상기 저-전력 사용-케이스 작업을 위해 상기 코어들 중 적어도 하나를 스케줄링하는 단계는,

각각의 코어에 대한 전력 조절기 효율을 결정하는 단계; 및

각각의 코어에 대한 상기 전력 조절기 효율에 의존하는 시스템-레벨 효율에 기초하여 저-전력 사용-케이스 작업에 대한 코어를 선택하는 단계를 포함하는, 복수의 코어들을 갖는 멀티-코어 프로세서를 포함하는 셀룰러 폰을 위한 방법.

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

발명의 설명

기술 분야

[0001] 관련 출원의 교차 참조

[0002] [0001]본 출원은, 2013년 12월 20일에 출원된 미국 정규 출원 제14/137,914호를 우선권으로 주장하며, 상기 출원은 그 전체가 인용에 의해 본원에 포함된다.

[0003] [0002]본 출원은 멀티-코어 프로세서를 위한 작업들의 동적 스케줄링에 관한 것이다.

배경 기술

[0004] 회로 밀도가 증가함에 따라, 컴퓨터 시스템 또는 다른 전자 시스템 중 전부 또는 대부분의 컴포넌트들이 시스템 온 칩(SOC)을 형성하기 위해서 하나의 칩에 통합될 수 있다. 회로 밀도에 있어의 진보는, SOC로 하여금 회로 모듈 또는 하드웨어 모듈의 다수의 인스턴스화(instantiation)들을 포함할 수 있게 한다. 예를 들어, 스마트 폰을 위한 SOC는 쿼드 코어 CPU(central processing unit), 4개의 하드웨어 스레드들을 갖는 DSP(digital

sensing processor), 4개의 ALU(algorithm logic units)들을 포함하는 GPU(graphic processing unit), 복수의 WLAN(wireless local area network) 인터페이스들, 3개의 VPE(vector processing engine)들을 갖는 모듈, 및 다수의 L2 캐시 온-칩 메모리들을 포함할 수 있다. 본원에서 사용되는 바와 같이, 용어 "코어" 또는 "모듈"은 동일한 회로 구조의 이러한 다수의 인스턴스화들에 대하여 상호교환가능하게 사용된다.

[0005] [0004]SOC는 통상적으로, 특정 작업에 대해 동일한 코어들의 그룹으로부터 특정 코어를 선택하는 운영 체제 또는 일부 다른 타입의 호스트 소프트웨어 애플리케이션으로 구성될 것이다. 이러한 선택을 가능하게 하기 위해서, 동일하거나 또는 실질적으로 동일한 코어들의 그룹으로부터의 각각의 코어는 통상적으로, 하드웨어 모듈의 식별(ID) 또는 주소로서 기능을 하는 시리얼 넘버 또는 문자와 연관될 것이다. 주소는 다이 상의 하드웨어 코어의 지리적 위치 또는 그의 제조 순서(또는 제조 파라미터들)에 기초할 수 있다. 어드레스를 이용하여, 운영 체제는 특정 작업을 위한 이러한 하드웨어 모듈들의 그룹으로부터 특정 하드웨어 모듈을 선택한다. 이상적으로, 동일한 하드웨어 모듈들은 동일한 특징과 성능을 나타내야 한다. 그러나 회로 밀도의 진보에 따라, CPU 코어들과 같은 동일한 모듈들이 그들의 성능 및 특성들에 있어서 상당한 변화를 보여주는 것이 일반적이다. 예를 들어, 코어들이 다이 상에 서로 바로 인접해 있더라도, 다이 그 자체는 동일한 회로들로 하여금 상이하게 거동하게 하는 반도체 프로세스 변화들을 갖는다. 또한, 다이는, 하드웨어 모듈들에 대한 상이한 위치들에서 온도 또는 전압의 변화들을 가질 수 있으며, 이는 그들의 성능에 또한 영향을 미칠 것이다. 그러나, 현재의 하드웨어 코어 선택 방식들은 이러한 개별 특징들을 고려하지 않는다.

[0006] [0005]따라서, 성능을 증가시키고 전력 소모를 낮추도록 하드웨어 코어들을 지능적으로 선택하는 프로세서 아키텍처에 대한 본 기술의 요구가 존재한다.

발명의 내용

[0007] [0006]복수의 코어들에 대해 작업들을 스케줄링하는 동적 스케줄러가 제공된다. 동적 스케줄러 코어 전력 및 열 모델은 코어들의 고유 누설 전류들과 현재 온도들에 기초하여 각각의 코어에 대한 누설 전류 전력을 분석적으로 모델링하도록 구성되고 또한 현재 전압 및 각각의 코어에 대한 클럭 주파수에 기초하여 각각의 코어에 대해 동적 전력을 모델링하도록 구성된다. 동적 스케줄러는, 작업들을 스케줄링하기 위해 누설 전류 전력들 및 동적 전력들과 같이 코어들에 대한 동작 파라미터들을 이용하는 스케줄링 알고리즘을 구현한다.

[0008] [0007]동적 스케줄러에 의해 구현되는 스케줄링 알고리즘은 또한, 각각의 코어에 대한 열적 그래디언트(또는 등가적으로는, 코어의 열 전도율)와 같은 매우 다양한 다른 동작 파라미터들을 사용할 수 있다. 다른 동작 파라미터들은 다이 상의 코어의 위치, 각각의 코어에 대한 최대 클럭킹 주파수, 및 각각의 코어에 대한 관련 전압 조절기 효율을 포함한다. 코어들에 대한 작업들의 동적 스케줄링과 관련하여 코어에 대한 이러한 동작 파라미터들 및 다른 동작 파라미터들의 유익한 사용이 아래에 추가로 설명될 것이다.

도면의 간단한 설명

[0009] [0008]도 1은 본 개시물의 일 실시예에 따른 동적 작업부하 스케줄러를 갖는 예시적인 멀티-코어 프로세서 시스템의 블록도이다.

[0009]도 2는 유휴 사용-케이스뿐만 아니라 저-전력 사용-케이스에 대한 작업부하 요청의 동적 스케줄링에 대한 흐름도이다.

[0010]도 3은 고-전력 사용-케이스에 대한 작업부하 요청의 동적 스케줄링을 위한 흐름도이다.

[0011]도 4는 전압 조절기 효율들을 이들의 출력 전류의 함수로서 도시하는 그래프이다.

[0012]도 5a는 프로세서 전력 소모를 복수의 코어들에 대한 프로세서 클럭킹 주파수의 함수로서 도시하는 그래프이다.

[0013]도 5b는 프로세서 효율을 프로세서 동작 주파수의 함수로서 도시하는 그래프이다.

[0014]도 6은 본 개시물의 실시예에 따른 일 예시적인 동적 스케줄링 알고리즘에 대한 흐름도이다.

[0015]본 발명의 실시예들 및 이들의 이점들은, 다음의 상세한 설명을 참고함으로써 가장 잘 이해된다. 동일한 도면 부호들은 하나 이상의 도면들에 도시된 동일한 엘리먼트들을 식별하는 데 사용된다는 것을 인식해야 한다.

발명을 실시하기 위한 구체적인 내용

- [0010] [0016]복수의 하드웨어 코어들에 대한 작업들을 스케줄링하기 위한 동적 스케줄러를 포함하는 멀티-프로세서 시스템이 제공된다. 종래 기술의 스케줄러들과는 달리, 개시된 동적 스케줄러는, 하드웨어 코어들의 고유 동작 파라미터들 이외에 하드웨어 코어들에 대한 현재 동작 파라미터들을 고려하는 스케줄링 알고리즘을 구현한다. 본원에 사용된 바와 같이, "고유" 동작 파라미터들은 제조에 있어서 특징으로 하는 성능 파라미터들을 지칭한다. 이러한 파라미터들은 정적이다 - 다시 말해서, 이들은 하드웨어 코어들에 대한 현재 동작 파라미터들에 대해 의존하지 않는다. 예를 들어, 이러한 고유 파라미터들은, 온도 및 최대 동작 주파수(Fmax)의 함수로서 누설 전류를 포함한다. 이는, 고유 파라미터들을 사용하는 것으로 알려져 있다. 예를 들어, 고유 누설 전류가 각각의 하드웨어 코어에 대해 결정되는 경우, 종래의 스케줄러는 단순히, 저 전력 작업을 최저의 고유 누설 전류를 갖는 코어에 할당할 수 있다. 하지만 고유 동작 파라미터들에만 기초하는 스케줄링은 최적이 아니다. 예를 들어, 최저 고유 누설을 갖는 코어를 가정하면, 저 전력 작업이 종래의 스케줄러에 의해서 그에 맞춰 코어에 할당된다. 그러나, 코어가 단지 활성 상태이어서, 그의 온도가 상승하는 경우(또는 이것이, 활성 상태이거나 막 활성 상태가 되었고 따라서 열 확산을 통해 코어가 가열되었던 카메라 또는 그래픽스 프로세서와 같은 회로 모듈의 다이 상에 근방에 있는 경우), 이러한 상대적으로 고온의 코어에 대한 누설 전류가, -이러한 코어들이 표면적으로 더 높은 고유 누설 전류들을 갖더라도- 나머지 코어들에 대한 누설보다 실제로 더 높을 수 있다. 본원에 개시된 동적 스케줄러에 대한 스케줄링 알고리즘은, 현재 동작 조건들뿐만 아니라 고유 동작 파라미터들에 대해 고려하므로, 동작 성능을 유리하게 증가시키고 전력 소모를 감소시킨다.
- [0011] [0017]예시적인 멀티-프로세서 시스템(100)이 도 1에 도시된다. 시스템(100)은 SoC(system-on-a-chip)를 포함할 수 있거나 또는 다수의 다이들을 포함할 수 있다. 동적 스케줄러(105)는, 코어 0, 코어 1, 코어 2 및 코어 3을 포함한 복수의 CPU 코어들(115)에 대한 작업들을 스케줄링한다. 그러나, 하드웨어 코어들의 다른 타입들 및 수량이 이러한 방식으로 스케줄링될 수 있다는 것을 인식할 것이다. 다음 설명은, 동적 스케줄러(105)가 또한, 전용 상태 머신 또는 다른 적절한 하드웨어로 구현될 수 있음을 인식하더라도, 동적 스케줄러(105)가 소프트웨어로 구현된다는 것을 가정할 것이다. 소프트웨어-기반 실시예에서, 코어들(115) 중 하나가 동적 스케줄러(105)를 구현하도록 구성될 수 있다. 대안으로, 동적 스케줄러(105)가 코어들(115)에 걸쳐 분산될 수 있어서, 현재 활성 상태인 코어(또는 코어들)가 무엇이든 간에 동적 스케줄러(105)를 구현할 것이다.
- [0012] [0018]동적 스케줄러(105)는, 동적 스케줄러(105)로 하여금, 코어의 고유 및 현재 동작 파라미터들에 기초하여 각각의 코어에 대한 현재 전력 소모 및 누설 전류를 계산할 수 있게 하는 각각의 코어(115)에 대한 분석적 전력 및 열적 모델러(110)를 포함한다. 예를 들어, 각각의 코어(115)는 동적 스케줄러(105)에 코어의 온도를 송신하도록 구성되는 온도 센서(미도시)와 연관된다. 모델러(110)는 이후, 현재 온도의 함수로서의 코어의 고유 누설 전류를 분석적 모델을 이용하여 코어의 현재 누설 전류($I_{leakage}$)로 맵핑할 수 있다. 이 점에 있어서, 고유 누설 전류가 기준 온도에서 측정된다. 분석 모델은, 현재 온도와 기준 온도 사이의 변화에 부분적으로 기초하여 현재 또는 예상 누설 전류를 예측한다. 모델링된 현재 누설 전류들을 이용하여, 모델러(110)는, 각각의 코어에 대한 누설 전류 전력($P_{leakage}$)을 결정하기 위해 각각의 코어에 대한 전류 동작 전압(V)을 사용할 수 있다. 이외에도, 동적 스케줄러(105)는 코어 당 동적 전력 소모($P_{dynamic}$)를 결정할 수 있으므로, 총 전력 소모(P_{total})가 계산되며, 이는 현재 누설 전류($P_{leakage}$)뿐만 아니라 동적 전력($P_{dynamic}$)의 합과 동일하다. 동적 전력($P_{dynamic}$)은 $f \cdot C \cdot V^2$ 와 동일하며, f는 코어의 현재 동작 주파수 또는 클럭킹 레이트이며, C는 코어의 동적 캐패시턴스이며, V는 코어의 현재 동작 전압이다. 따라서, 코어에 대한 총 전력 소모(P_{total})는 $P_{leakage}$ 와 $P_{dynamic}$ 의 합과 동일하다. 일 실시예에서, 동적 스케줄러(105)는, 코어들에 대해 예측되는 누설 전류의 분석 모델에 기초하여 코어들에 대한 작업들을 스케줄링하기 위한 수단을 포함하는 것으로 간주될 수 있다.
- [0013] [0019]분석적 코어 전력 및 모델러(110)에 의해 사용되는 고유 누설 전류(IDDQ)는 기준 동작 전압(V_{ref}) 및 기준 온도(T_{ref})와 관련하여 결정될 수 있다. 즉, 고유 누설 전류(IDDQ)는 기준 동작 전압(V_{ref}) 및 기준 온도(T_{ref})에서 각각의 코어에 대한 제조 시에 측정(또는 추정)된다. 다양한 분석 모델들이 분석 코어 전력 & 열적 모델러(110)에서 사용될 수 있다. 예를 들어, 일 예시적인 해석 모델은 다음 식을 사용하여 현재 누설 전류($I_{leakage}$)를 계산한다.
- [0014]
$$I_{leakage} = IDDQ * \exp(m \cdot (V - V_{ref}) + N \cdot (T - T_{ref}))$$

- [0015] n 및 m 은 동작 파라미터들이고, \exp 는 지수 함수이다. 따라서, 모델링되는 누설 전류($I_{leakage}$)는 대응하는 코어(115)의 현재 동작 파라미터들에 기초하여 예측되는 누설 전류이다. 즉, 주어진 코어(115)에 대한 $I_{leakage}$ 는, 현재 작업에 대해 코어(115)가 선택될 경우 코어(115)가 경험할 것을 모델러(110)가 예측하는 누설 전류이다.
- [0016] [0020]동적 스케줄러(105)는, 주어진 작업에 최저 전력 소모를 제공하는 코어(115)를 선택하기 위해 모델러(110)로부터 코어들(115)에 대해 모델링된 누설 및 동적 전력들을 사용할 수 있는 멀티-프로세서(MP) 작업-할당 또는 스케줄링 알고리즘(120)을 구현한다. 그러나, 전력 소모가, MP 스케줄링 결정에 관한 유일한 팩터는 아니다. 다른 팩터는, 요구되는 작업 또는 고전력 작업이 실행되고 있는 경우에 도달되는 코어들(115)에 대한 열적 한계치이다. 이와 관련하여, 코어 열적 한계치 문제들을 수용하기 위해 코어-호핑 및 다른 스케줄링 기술들을 구현하는 것이 알려져 있다. 그러나 MP 스케줄링 알고리즘(120)은, 열적 한계치들을 회피하는 것과 관련하여 코어(115)를 더욱 지능적으로 선택하기 위해 각각의 코어(115)에 대한 열적 그래디언트를 사용할 수 있다. 고유 누설 전류(IDDQ)와 마찬가지로, 각각의 코어(115)에 대한 열적 그래디언트가 제조 시에 결정될 수 있다. 열적 그래디언트는, 각각의 코어(115)가 주어진 작업에서 동작함에 따라 그 코어가 경험하게 될 시간의 함수로서의 온도 변화이다. 예를 들어, 프로세서들이 이들의 최대 클러킹 주파수(f_{max})에서 동작될 경우, 각각의 코어(115)에 대한 열적 그래디언트들이 특징화될 수 있다. 열적 그래디언트에 대한 등가의 파라미터는, 주어진 코어(115)가 동작함에 따라 이것이 열을 발산하는 능력을 지칭하는 각각의 코어(115)에 대한 열 전도율이다. 다양한 코어들(115)에 대한 온도 그래디언트들과 열 전도율들이 서로 역으로 관련된다는 것을 인식할 수 있다. 즉, 상대적으로 높은 열적 그래디언트를 갖는 코어(115)는 비교적 낮은 열 전도율을 갖는다. 역으로, 상대적으로 낮은 열적 그래디언트를 갖는 코어(115)는 비교적 높은 열 전도율을 가질 것이다. 열적 그래디언트(또는 열적 전도율)에 기초하여, 모델러(110)는, 주어진 코어(115)가 현재 작업을 할당받게 됨에 따라 그 코어가 경험할 온도 증가를 예측할 수 있다. 따라서, MP 스케줄링 알고리즘(110)은 주어진 작업에 대한 최저 온도 상승이 있을 코어(115)에 대해 선택할 수 있다.
- [0017] [0021]동적 스케줄러(105)는, 이들의 고유 누설(IDDQ) 및 전류 온도 이외에, 코어들(115)로부터의 매우 다양한 동작 파라미터들을 사용할 수 있다. 예를 들어, 동적 스케줄러(105)는, 코어들에 대한 프로세스 코너 (코어들이 저속/공칭(nominal)/또는 고속 프로세스에 해당하는지 여부)에 의존할 수 있다. 그 점에서, 동적 스케줄러(105)는, 프로세스 코너를 포함하는 다양한 동작 파라미터들에 따라 주어진 작업에 대해 하나의 코어(115) 또는 복수의 코어들(115)을 선택할 수 있다. 예를 들어, PVS(processor voltage scaling)는 각각의 코어(115)에 대해 상이할 수 있다. 예를 들어, 주어진 클럭 주파수에서, 하나의 코어(115)는, 동일한 클럭 레이트에서 동작하는 다른 코어(115)와 비교하여 더 높은 공급 전압이 필요할 수 있다. 이와 같이, 동적 스케줄러(105)는, 총 전력 소모를 감소시키기 위해서 더 낮은 전압에서 동작할 수 있는 코어(115)를 선택할 수 있다. 클럭 주파수는, DCVS(dynamic clock and voltage scaling) 구현들에서 공지된 바와 같이 또한 스케일링될 수 있다. 그 점에 있어서, 최대 동작 주파수 또는 클러킹 레이트(F_{max})는 다양한 코어들(115)에 대해 상이할 수 있다. 주어진 동작 전압에서, 이와 같이 하나의 코어(115)가 더 높은 F_{max} 를 달성할 수 있다. 이와 같이, 동적 스케줄러(105)는, 최저의 전력 소모로 가장 빠른 성능을 달성하기 위해서 요구되는 작업들 또는 고전력 작업들에 대해 이러한 코어(115)를 선택할 수 있다. 그러나, PVS/DCVS는 코어(115)에 대해 독립적이지 않을 수 있다는 것을 인식할 것이다. 이러한 구현들에서, 모든 코어들(115)이, 그로부터 선택될 수 있는 동일한 세트의 전압 및 클럭 레이트들을 가질 것이다. 이러한 균일한 DCVS 실시예에서 코어들(115) 사이에는 이들의 F_{max} 및 동작 전압들과 관련하여 차이가 없을 수 있다.
- [0018] [0022]동적 스케줄러(105)가 코어들(115)에 대해 의존할 수 있는 다른 동작 파라미터는, 다이 (또는 다이들) 상에서의 그들의 위치 및 그 위치 대 다른 열원들의 관계이다. 예를 들어, 코어(115)는 그의 열적 한계치로 이제 막 푸시되었던 다른 코어(115)로부터 가장 멀리 떨어져 있는 위치를 가질 수 있다. 따라서, 이러한 원격 코어(115)에 대해서는 비디오 프로세싱과 같은 고전력 작업의 스케줄링이 유익하다. 대안으로, 코어(115)가, 모뎀 프로세서 또는 비디오 프로세서와 같은 다른 능동 회로에 대해 다이 상에 비교적 가깝게 위치될 수 있다. 이러한 코어(115)는, 코어가 또한 이 근방의 능동 회로로부터의 열을 수용해야 한다는 점에서 코어가 더욱 신속하게 그의 열적 한계치에 도달할 것이기 때문에 고전력 작업을 위해 양호한 선택이 되지 않을 것이다. 따라서, 이 능동 회로로부터 더 멀리 떨어져 있는 코어(115)의 선택이 유익하다.
- [0019] [0023]동적 스케줄러(105)의 유익한 특징들을 더 잘 예시하기 위해서 일부 예시적인 사용 케이스를 이제 더 설명할 것이다. 예를 들어, 시스템(100)이 유휴 상태인 경우, 동적 스케줄러(105)가 최저 누설을 갖는 코어에 대해 유휴 작업을 스케줄링함으로써 전력 소모를 최소화할 수 있다. 결과적인 스케줄링 프로세스가 도 2에 도시

되며, 여기서, 인입하는 작업부하 요청(200)은, 이것이 단계(205)에서의 유휴 사용-케이스인지 여부를 결정하기 위해서 테스트된다. 예를 들어, 셀룰러 폰 애플리케이션에서는, 사용자가 전화로 상호작용하지 않거나 어떤 전화나 문자도 수신하지 않는 기간들이 종종 있다. 따라서, 코어들(115)은 이러한 유휴-사용 케이스 동안에 대부분 유휴 상태로 있게 된다. 작업 요청이 유휴-사용 케이스에 대응한다는 결정 시, 룩업 테이블(LUT)(240)과 같은 메모리는, 코어들(115)에 대한 현재 시스템 정보 및 그들의 동작 파라미터들 및 특징들을 저장할 수 있다. 예를 들어, LUT(240)는 모델러(110)에 의해 결정되는 바와 같이 최저 누설 전류로 코어(115)의 아이덴티티를 저장할 수 있다. 단계(205)에서의 결정이 긍정적인 경우, 이후, 동적 스케줄러(105)는 LUT(240)에 의해 식별되는 바와 같이 단계(210)에서 최저 누설 전류를 갖는 이용가능한 코어(115)를 선택한다. 동적 스케줄러(105)는 이후, 단계(215)에서 스케줄링 결정에 따라 LUT(240)를 업데이트할 수 있다. 마지막으로, 동적 스케줄러(105)는 요청된 셋팅들을 이용하여 선택한 코어(115)를 턴온시킨다(단계 (220)).

[0020] [0024]단계(205)에서의 결정이 부정적인 경우, 인입하는 작업부하 요청(200)이 단계(221)에서 결정되는 바와 같이 저전력 사용-케이스를 포함할 수 있다. 저전력 사용-케이스들의 예들은 비디오 재생 또는 웹-브라우징과 같은 작업들을 포함한다. 단계(221)에서의 결정이 긍정적인 경우, 동적 스케줄러(105)는 이후, 모델러(110)에 의해 결정된 바와 같이 LUT(240)에서 식별된 바와 같이 최저 총 전력 소모 특징을 이용하여 코어(또는 코어들)를 스케줄링함으로써 총 전력 소모를 최소화할 수 있다(단계 225). 후속 단계(230)에서, 동적 스케줄러(105)는 스케줄링 결정에 따라 LUT(240)를 업데이트할 수 있다. 마지막으로, 동적 스케줄러(105)는 요청된 셋팅들을 이용하여 선택한 코어(115)를 턴온시킨다(단계 (235)).

[0021] [0025]단계 (221)에서의 결정이 부정적이라면, 인입하는 작업부하 요청(200)은, 도 3의 단계(300)에서 결정된 바와 같이, 비디오 게임(또는 유사한 고 전력 모드)에 대한 집약적인 그래픽스 프로세싱과 같은 많은 (고성능) 작업부하 사용-케이스를 포함할 수 있다. 동적 스케줄러(105)는 이후, 자신의 열적 한계치를 초과하지 않고 원하는 성능을 제공할 수 있는 코어(115)(또는 코어들)를 선택할 수 있다(단계 (305)). 예를 들어, 단계(305)는, LUT(240)에서 식별된 바와 같이 양호한 열적 그래디언트 및 최저 온도를 갖는 이용가능한 코어(115)에 대해 선택할 수 있다. 후속하는 단계(310)는 스케줄링 결정에 따라 LUT(240)를 업데이트한다. 단계(315)에서, 동적 스케줄러(105)는 요청된 셋팅들을 이용하여 선택한 코어(115)를 턴온시킨다. 선택된 코어(115)가 결국 너무 과열될 수 있어, 선택된 코어(115)가 자신의 열적 한계치를 초과할 위험이 있다는 것을 주목한다. 이와 같이, 동적 스케줄러(105)는, 선택된 코어(115)가 자신의 열적 한계치 근처에 있는지 여부를 결정(단계 (325))하기 위해서, 선택된 코어 온도를 모니터링할 수 있다(단계 (320)). 단계(325)에서의 결정이 부정적이라면, 모니터링은 단계(320)에서 계속될 수 있다. 그러나, 선택된 코어(115)가 그의 열적 한계치에 도달하고 있는 경우, 동적 스케줄러(105)는 그의 스케줄링 선택을 갱신한다(단계 (305)). 단계들(310, 315, 320, 및 325)은 이후, 새롭게 선택된 코어(115)와 관련하여 반복될 수 있다. 따라서, 단계(315)는 또한 이전에 선택된 코어(115)를 턴 오프하는 것과 LUT(240)를 새로운 선택으로 업데이트하는 것을 포함할 수 있다.

[0022] [0026]동적 스케줄러(105)에 의한 총 전력 소모의 최소화가 또한 각각의 코어의 전력 조절기 효율을 고려하여 선택될 수 있다. 이와 관련하여, 멀티-프로세서 시스템(100)의 각각의 코어(115)에는, 대응하는 전력 조절기(미도시)에 의해, 전력이 전달된다. 따라서, 코어 0 전력 조절기, 코어 1 전력 조절기 등이 있을 수 있다. 대안으로, 전력 조절기는 시스템(100) 내의 코어들의 서브세트에 의해 공유될 수 있다. 일 실시예에서, 모델러(110)는 또한, 대응하는 코어(또는 코어들)에 대한 현재 총 전력 소모(P_{total})에 기초하여 각각의 전력 조절기의 효율을 모델링한다. 예를 들어, 모델러(110)는, 코어(115)에 대한 총 전력 소모를 코어의 전압 조절기로부터 코어(115)에 의해 인출되는 전류로 맵핑할 수 있다. 모델러(110)는 또한, 조절기가 공급하는 전류, 그의 동작 전압, 클럭킹 주파수, 및 다른 기능 파라미터들의 함수로서 각각의 전력 조절기의 효율의 모델을 포함할 수 있다. 이러한 조절기 효율 모델들은 통상적으로, 어떤 최적의 전류 값에서 피크이고 그런 다음, 전류가 최적값으로부터 감소되거나 또는 증가됨에 따라 강아되는 대략 종 모양의 효율 곡선을 나타낸다. 예를 들어, 도 4는 전압 조절기(400 및 405)에 대한 출력 전류의 함수로서의 효율을 도시한다. 전압 조절기(400 및 405) 둘 모두에 대한 효율은 500 내지 1000 밀리암페어의 출력 전류에서 피크이다. 따라서, 동적 스케줄러(105)는, 코어의 전압 조절기 효율이 너무 낮을 수 있는 작업들에 대해 코어(115)를 스케줄링하는 것을 방지하기 위해서 이 정보를 사용할 수 있다. 예를 들어, 도 5a는, 상대적 고온에서의 저속 프로세스 코어 코어(500), 상대적 저온에서의 저속 프로세스 코어 코어(505), 및 상대적 저온에서의 고속 프로세스 코어 코어(510)에 대한 프로세서 전력 소모를 도시한다. 도 5a에 도시된 전력 소모들 모두는, 동작 주파수(프로세서 클럭 레이트)가 증가됨에 따라 단조적으로 증가한다. 따라서, 최고 효율들은, 코어들이 가능한 한 낮게 주어지는 주파수에서 최소량의 전력을 소모하는 그러한 주파수들에서 이러한 코어들을 동작시킴으로써 생성될 것으로 기대할 수 있다. 그러나, 이러

한 가정은, 관련되는 전력 조절기들에 대해 급격히 감소되는 전력 효율을 무시하는데, 각각의 전력 조절기는 이후에, 그 최적의 전류량보다 적은 양을 그 연관된 코어로 공급하고 있을 것이기 때문이다.

[0023] [0027]일 실시예에서, 따라서, 모델러(110)는 코어 클럭킹 주파수에 기초하여 그의 효율의 함수로서 각각의 전력 조절기로부터의 전력 소모($P_{regulator}$)를 모델링한다. 따라서, 동적 스케줄러(105)가 코어들(115)에 대한 작업들을 스케줄링할 수 있으므로, 시스템-레벨 전력 소모(P_{system})가 최소화되며, 여기서, P_{system} 은 각각의 코어(또는 코어들이 그룹이 대응하는 전력 조절기를 공유하는 경우 코어들의 그룹)에 대한 P_{total} 과 각각의 코어에 대한 연관된 $P_{regulator}$ 를 합산한 것과 같다. 도 5b는, 코어들(500, 505 및 515)에 대한 동작 주파수의 함수로서 프로세서 에너지 효율을 도시하며, 에너지 효율은, 시스템-레벨 전력 소모(P_{system})를 이용하여 계산된다. 각각의 코어의 에너지 효율은 F_{max} 미만인 동작 주파수(F_{opt})에서 피크 효율을 갖는다. 따라서, 동적 스케줄러(105)는 코어들에 대해 F_{opt} 미만인 레벨(515)의 동작 주파수들을 디스플레이시킬 수 있다.

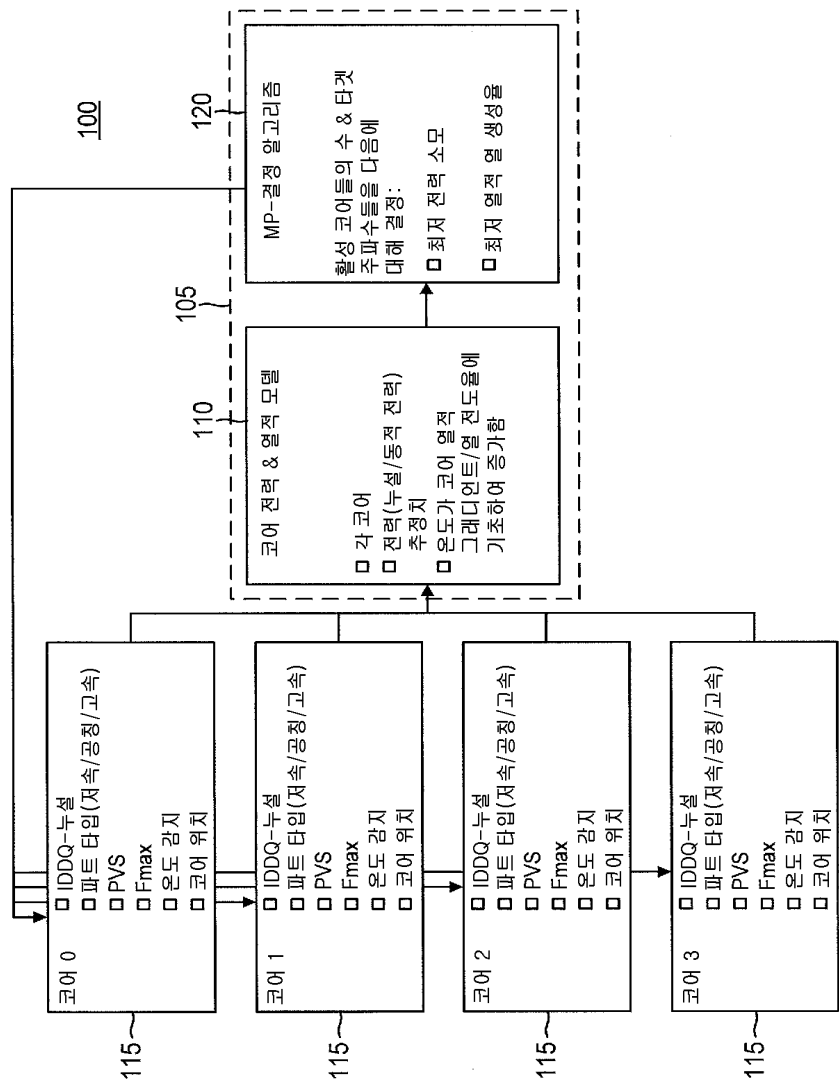
[0024] [0028]도 3을 다시 참조하면, 동적 스케줄러(105)는, 결과적으로 생성된 에너지 효율과 함께 단계(305)에서 코어(115)의 선택을 가중시키도록 구성될 수 있다. 예를 들어, 고전력 작업이 상대적으로 짧은 것으로 예상되어, 열적 한계에 도달할 위험성이 거의 없는 경우, 코어 선택은 대신에 에너지 효율의 최대화에 의해 지배될 수 있다. 작업 시간이 증가됨에 따라, 단계(305)는 에너지 효율들에 비해 열적 인자들(최저 온도 및 열적 그래디언트를 갖는 코어)을 더 높게 가중시킬 수 있다.

[0025] [0029]동적 스케줄러(105)에 대한 동작의 예시적인 방법이 이제 도 6을 참조하여 다루어질 것이다. 단계(600)는 복수의 코어들 중 각각의 코어에 대해 예상되는 누설 전류를 결정하는 단계를 포함하며, 이 결정은, 코어에 대한 고유 누설 전류 및 각각의 코어의 온도에 의존하는 분석 모델에 대해 반응한다. 단계(605)는, 최저 예상 누설 전류를 갖는 코어를 결정하기 위해서 예상되는 누설 전류들을 비교하는 단계를 포함한다. 마지막으로, 단계(610)는 작업에 대해 최저 예상 누설 전류를 갖는 코어를 스케줄링하는 단계를 포함한다.

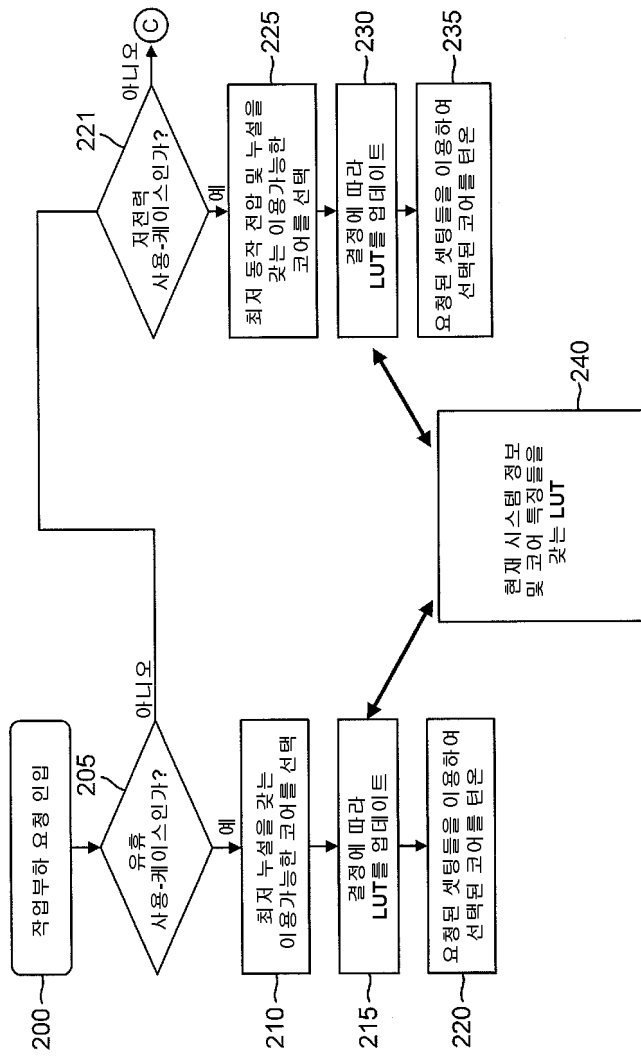
[0026] [0030]이제 당업자가 인식하는 바와 같이 그리고 준비되어 있는 특정 용도에 의존하여, 본 개시물의 디바이스들의 재료들, 장치, 구성들 및 사용 방법들에 대하여 많은 변경들, 대체들 및 변형들이, 본 발명의 정신 및 범위로부터 벗어나지 않고 이루어질 수 있다. 이러한 관점에서, 본 개시물의 범위는 본원에 예시되고 설명되는 특정 실시예들의 범위로 제한되지 않아야 하고, 이들은 단지 본 개시물의 일부 예시일 뿐이지만, 더 정확하게는, 이후에 첨부되는 청구항들 및 이들의 기능적 등가물들의 범위와 전적으로 상응되어야 한다.

도면

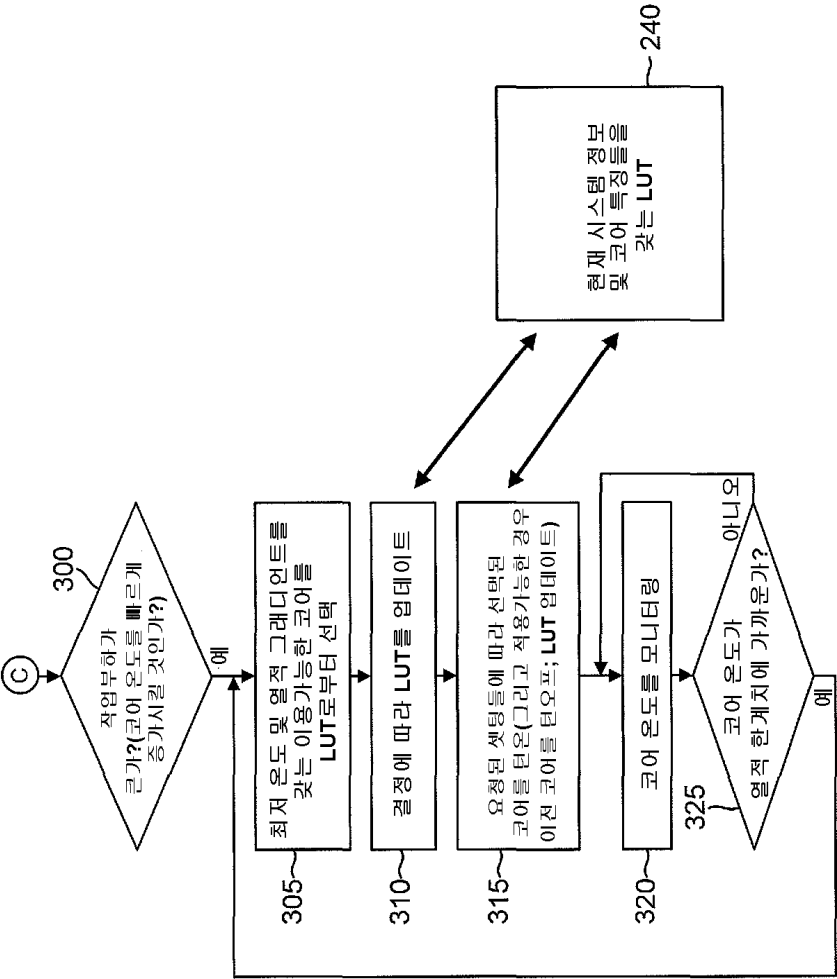
도면1



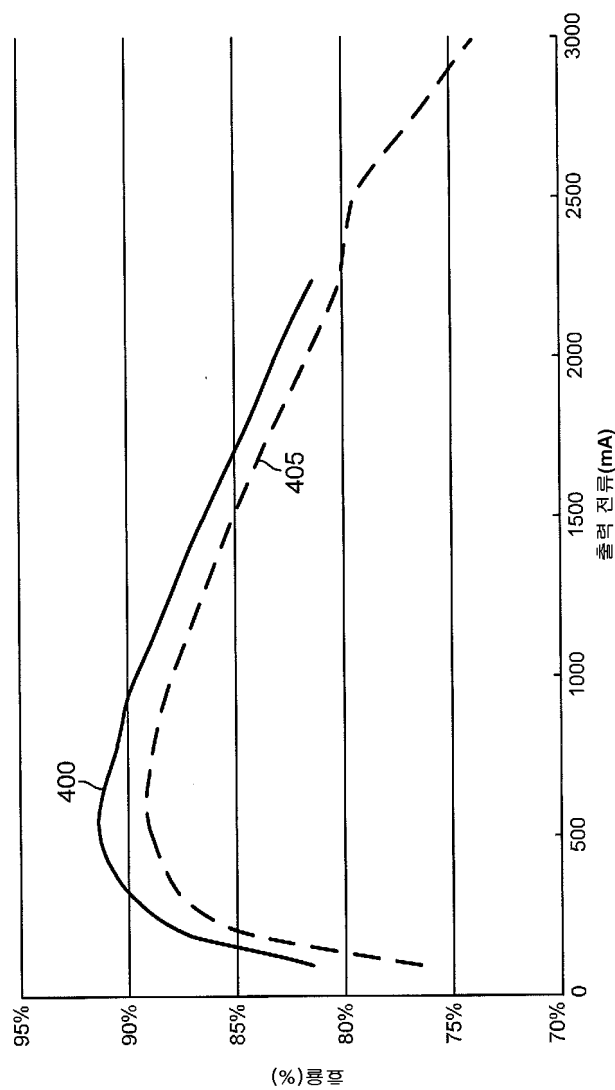
도면2



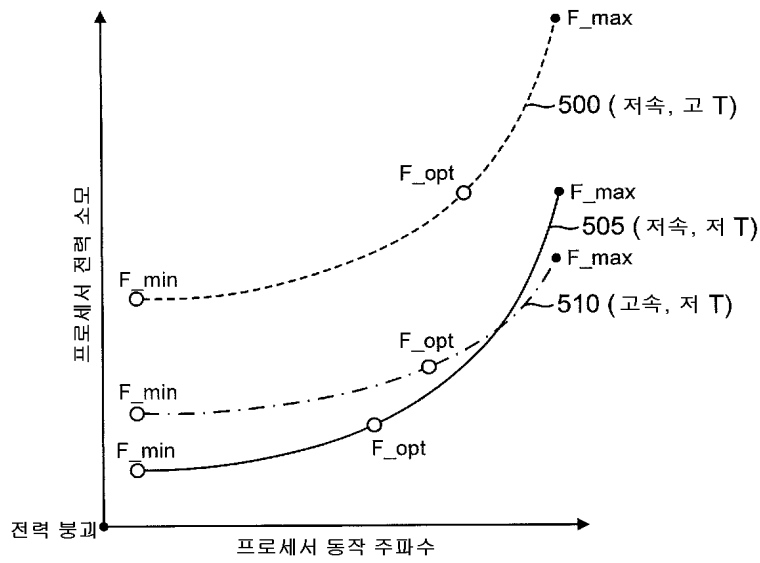
도면3



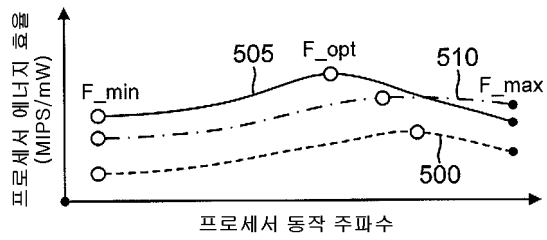
도면4



도면5a



도면5b



도면6

