

	(19) 대한민국특허청(KR) (12) 공개특허공보(A)	(11) 공개번호 10-2014-0086577 (43) 공개일자 2014년07월08일
(51) 국제특허분류(Int. Cl.) G06F 1/32 (2006.01) G06F 9/46 (2006.01) G06F 9/50 (2006.01)	(71) 출원인 경상대학교산학협력단 경상남도 진주시 진주대로 501 (가좌동)	(72) 발명자 김경훈 경남 진주시 돛골로 110, 202호 (상대동) 이완연 강원 춘천시 칠전서길 15-2, 205동 602호 (칠전동, 칠전대우2차아파트)
(21) 출원번호 10-2012-0157252 (22) 출원일자 2012년12월28일 심사청구일자 2012년12월28일	(74) 대리인 특허법인명인	

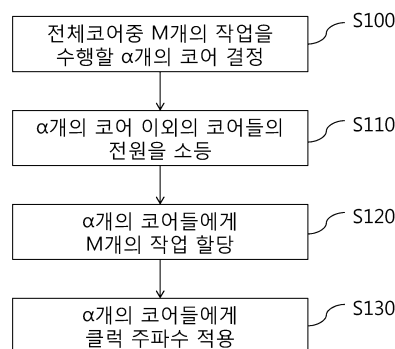
전체 청구항 수 : 총 5 항

(54) 발명의 명칭 **변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 위한 멀티 코어 프로세서의 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법**

(57) 요약

본 발명은 변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 멀티 코어 프로세서에서 수행할 때 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법에 관한 것으로서, 변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 위한 멀티 코어 프로세서의 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법에 있어서, 멀티 코어 프로세서의 전체 코어들 중에서 M개의 주기적 실시간 작업을 수행하는데 사용될 α 개의 코어들을 결정하는 제1 단계; 상기 결정된 α 개의 코어들을 제외한 나머지 코어들의 전원을 소등하는 제2 단계; 상기 결정된 α 개의 코어들에게 M개의 주기적 실시간 작업들을 할당하는 제3 단계; 및 상기 α 개의 각 코어에서 상기 M개의 주기적 실시간 작업들을 각각 수행할 때 전력 소모량 확률적 기대값을 최소화하는 실행 속도를 결정하는 제4 단계를 포함하는 스케줄링 방법을 제공한다. 본 발명에 의하면, 멀티 코어 프로세서상에서 변동하는 불확실한 계산량을 갖는 다수의 실시간 작업들의 수행할 때 누수 전력 소모를 고려하여 확률적 모델에 기초하여 전력 소모를 최소화할 수 있는 스케줄링 방법을 제공할 수 있는 효과가 있다.

대표도 - 도1



특허청구의 범위

청구항 1

변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 위한 멀티 코어 프로세서의 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법에 있어서,

멀티 코어 프로세서의 전체 코어들 중에서 M개의 주기적 실시간 작업을 수행하는데 사용될 α 개의 코어들을 결정하는 제1 단계;

상기 결정된 α 개의 코어들을 제외한 나머지 코어들의 전원을 소등하는 제2 단계;

상기 결정된 α 개의 코어들에게 M개의 주기적 실시간 작업들을 할당하는 제3 단계; 및

상기 α 개의 각 코어에서 상기 M개의 주기적 실시간 작업들을 각각 수행할 때 전력 소모량 확률적 기대값을 최소화하는 실행 속도를 결정하는 제4 단계

를 포함하는 스케줄링 방법.

청구항 2

제1항에 있어서,

상기 제1 단계는, 하나의 코어에 할당된 i 개 작업들의 확률적 전력 소모량 최소 기대값을 이용하여 멀티 코어 프로세서의 전체 코어들 중에서 M개의 주기적 실시간 작업을 수행하는데 사용될 α 개의 코어들을 결정하는 것을 특징으로 하는 스케줄링 방법.

청구항 3

제1항에 있어서,

상기 제3 단계는,

주어진 M개의 모든 작업들에 대한 Q값에 기초하여 상기 결정된 α 개의 코어들에게 M개의 주기적 실시간 작업들을 할당하는 것을 특징으로 하는 스케줄링 방법.

청구항 4

제3항에 있어서,

주어진 모든 M개의 작업들의 Q값을 계산하여 내림차순으로 정렬하는 제3-1 단계; 및

정렬된 순서에 따라 각 작업을 각각의 코어에 할당하는 제3-2 단계

를 포함하는 것을 특징으로 하는 스케줄링 방법.

청구항 5

제4항에 있어서,

상기 제3-2 단계는,

각각의 코어들에게 기존에 할당된 작업들의 $\sum e$ 값이 최소인 코어를 선택하여 할당하는 것을 특징으로 하는 스케줄링 방법.

명세서

기술분야

본 발명은 멀티 코어 프로세서에서의 스케줄링 방법에 관한 것으로서, 보다 상세하게는 변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 멀티 코어 프로세서에서 수행할 때 누수 전력 소모를 고려한 확률적 최소 전력

[0001]

스케줄링 방법에 관한 것이다.

배경 기술

- [0002] 최근, 관련 기술의 발전에 따라 고성능 전자 기기가 널리 보급되고 있다. 이러한 전자 기기는 사용자의 고성능 및 다기능 요구에 부응하기 위하여 멀티코어 프로세서를 구비할 수 있다.
- [0003] 멀티코어 프로세서(Multi-Core Processer)는 2개 이상의 코어를 갖는 프로세서로서, 멀티코어 프로세서를 이용하면, 단일 프로세서의 발열과 전력 문제를 어느 정도 해결할 수 있고, 복수의 작업을 각 코어에서 독립적으로 또는 병렬적으로 처리할 수 있다는 장점이 있으나 다수의 코어를 관리해야 하므로 적절한 스케줄링 방법이 필요하다.
- [0004] 이러한 멀티코어 프로세서의 스케줄링 방법으로서 동적 전압 및 주파수 스케일링(Dynamic voltage and frequency scaling, DVFS) 방법이 알려져 있다. DVFS 기법에서는 프로세서의 코어에 적용되는 전압값을 증가시키면 클럭 주파수가 정비례하여 증가하고 또한 프로세서의 코어의 실행 속도 또한 증가한다. 프로세서 코어에 적용되는 전압(클럭주파수 값 또는 코어의 실행속도)를 증가시키면 에너지 소모량도 이에 비례하여 증가하고 일반적으로 에너지 소모량 증가율이 전압값 증가율보다 크다.
- [0005] 이러한 관계를 이용하여 실시간 작업(real-time tasks)의 데드라인(deadline)에서 최대 계산량(worst-case computation amount)을 종료하도록 클럭 주파수를 감소시키면서 에너지 소모량을 최소화하기 위한 다양한 방법이 제안되어 있다.
- [0006] 한편, 고정된 최대 계산량이 아닌 변동하는 계산량을 고려하는 방안이 최근 제안되어 있다. 실제 계산량은 대부분의 경우 최대 계산량보다 작으며 또한 종료시까지 불확실한 경우가 대부분이다. 이러한 방안들은 변동하는 계산량을 확률적 계산량으로 변환하여 확률적 계산량의 평균 에너지 소모량을 최소화한다. 낮은 에너지 소모량을 갖는 낮은 클럭 주파수가 높은 확률을 갖는 연산부에 할당되고 높은 클럭 주파수는 낮은 확률을 갖는 연산부에 할당된다. 실제 계산량이 최대 계산량보다 작은 경우 이러한 확률적 계산 기법은 데드라인에 정확히 최대 계산량을 종료하는 고정 주파수를 할당하는 종래의 방식에 비해 보다 적은 에너지를 소모한다. 이러한 기법들은 클럭 주파수와 에너지 소모량 사이의 수식에 의해 무한 연속 주파수를 갖는 실시간 작업들의 에너지 소모를 최소화하는 문제를 해결하고 있다. 그러나, 실제의 DVFS 기법의 시스템 환경에서는 유한한 클럭 이산 주파수(discrete frequency)가 제공되며 적용가능한 클럭 이산 주파수와 에너지 소모량과의 관계 또한 불규칙하다. 또한, 종래의 기술은 단일 실시간 작업(single real-time task)의 경우만을 고려하고 있다.
- [0007] 본 출원의 발명자가 제안했던 대한민국 공개특허공보 제10-2010-0026989호(2010.03.10. 공개)는 "에너지 절약형 멀티 코어 프로세서"에 대하여 개시하고 있으나 이는 고정된 계산량(worst-case 최대 계산량)을 기준으로 데드라인 이내에 수행을 완료하는 에너지 소모량을 최소화하는 방법을 개시하고 있다는 점에서 한계점을 가지고 있다.
- [0008] 또한, 본 출원의 발명자가 또 다시 제안했던 대한민국 등록특허공보 제10-1065436호(2011.09.19. 공고)는 "불확실한 계산량을 가진 실시간 병렬 작업을 위한 멀티코어 프로세서의 확률적 스케줄링 방법"에 관한 것으로서, 이는 멀티코어 프로세서상에서 병렬로 수행가능한 단일 실시간 작업의 변동하는 불확실한 계산량을 확률적 계산량 모델로 표현하여 에너지 소모량의 확률적 기대값을 최소화하는 방법에 대해 개시하고 있으나, 단일코어 프로세서상에서 변동하는 불확실한 계산량을 갖는 다수의 실시간 작업들에 대해서는 적용할 수 없다는 한계점을 갖는다.
- [0009] 또한, 본 출원의 발명자가 또 다시 제안했던 대한민국 특허출원 제10-2011-0135518호(2011.12.15. 출원, 미공개)는 "변동하는 계산량을 갖는 실시간 작업을 위한 단일 코어 프로세서의 에너지 효율적 스케줄링 방법"에 관한 것으로서, 변동하는 계산량을 갖는 실시간 작업을 통계적 모델에 기초하여 스케줄링하는 방법을 개시하고 있으나 단일코어 프로세서를 기초로 한 스케줄링 방법이므로 멀티코어 프로세서에는 적용할 수 없다는 한계점이 있다.

선행기술문헌

특허문헌

- [0010] (특허문헌 0001) 대한민국 공개특허공보 제10-2010-0026989호(2010.03.10. 공개)

(특허문헌 0002) 대한민국 등록특허공보 제10-1065436호(2011.09.19. 공고)

발명의 내용

해결하려는 과제

- [0011] 본 발명은 상기한 바와 같은 한계점을 감안하여 안출된 것으로서, 멀티 코어 프로세서상에서 변동하는 불확실한 계산량을 갖는 다수의 실시간 작업들의 수행할 때 누수 전력 소모를 고려하여 확률적 모델에 기초하여 전력 소모를 최소화할 수 있는 스케줄링 방법을 제공하는 것을 목적으로 한다.

과제의 해결 수단

- [0012] 상기한 바와 같은 과제를 해결하기 위하여 본 발명은, 변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 위한 멀티 코어 프로세서의 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법에 있어서, 멀티 코어 프로세서의 전체 코어들 중에서 M개의 주기적 실시간 작업을 수행하는데 사용될 α 개의 코어들을 결정하는 제1 단계; 상기 결정된 α 개의 코어들을 제외한 나머지 코어들의 전원을 소등하는 제2 단계; 상기 결정된 α 개의 코어들에게 M개의 주기적 실시간 작업들을 할당하는 제3 단계; 및 상기 α 개의 각 코어에서 상기 M개의 주기적 실시간 작업들을 각각 수행할 때 전력 소모량 확률적 기대값을 최소화하는 실행 속도를 결정하는 제4 단계를 포함하는 스케줄링 방법을 제공한다.
- [0013] 여기에서, 상기 제1 단계는, 하나의 코어에 할당된 i 개 작업들의 확률적 전력 소모량 최소 기대값을 이용하여 멀티 코어 프로세서의 전체 코어들 중에서 M개의 주기적 실시간 작업을 수행하는데 사용될 α 개의 코어들을 결정할 수 있다.
- [0014] 또한, 상기 제3 단계는, 주어진 M개의 모든 작업들에 대한 Q값에 기초하여 상기 결정된 α 개의 코어들에게 M개의 주기적 실시간 작업들을 할당하도록 할 수도 있다.
- [0015] 또한, 상기 할당하는 단계는, 주어진 모든 M개의 작업들의 Q값을 계산하여 내림차순으로 정렬하는 제3-1 단계; 및 정렬된 순서에 따라 각 작업을 각각의 코어에 할당하는 제3-2 단계를 포함할 수 있다.
- [0016] 또한, 상기 제3-2 단계는, 각각의 코어들에게 기존에 할당된 작업들의 $\sum Q$ 값이 최소인 코어를 선택하여 할당하도록 구성할 수도 있다.

발명의 효과

- [0017] 본 발명에 의하면, 멀티 코어 프로세서상에서 변동하는 불확실한 계산량을 갖는 다수의 실시간 작업들의 수행할 때 누수 전력 소모를 고려하여 확률적 모델에 기초하여 전력 소모를 최소화할 수 있는 스케줄링 방법을 제공할 수 있는 효과가 있다.

도면의 간단한 설명

- [0018] 도 1은 본 발명에 의한 변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 위한 멀티 코어 프로세서의 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법의 전체적인 과정을 나타낸 흐름도이다.

도 2는 과거 계산량 확률 분포도이다.

도 3은 도 2의 확률 분포의 계산량 누적 확률 분포도이다.

도 4는 미래 계산량 확률 분포도를 나타낸 것이다.

도 5 및 도 6은 확률적 계산량 모델 활용 방법을 설명하기 위한 도면으로서, 도 5는 기존의 방법으로서 최대 계산량(worst case)을 테드라인 이내에 수행을 완료하도록 프로세싱 코어의 실행 속도(클럭 주파수)를 결정하는

경우를 나타낸 것이고, 도 6은 미래 계산량 확률(P^t)에 근거하여 수행될 확률이 높은 전반부는 낮은 실행 속도로 수행하고 확률이 낮은 후반부는 높은 실행 속도로 수행하는 경우를 나타낸 것이다.

도 7은 함수 MP를 이용하여 사용할 코어를 결정하는 방법을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0019] 이하 첨부 도면을 참조하여 본 발명에 의한 실시예를 상세하게 설명하기로 한다.
- [0020] 도 1은 본 발명에 의한 변동하는 계산량을 갖는 복수개의 주기적 실시간 작업을 위한 멀티 코어 프로세서의 누수 전력 소모를 고려한 확률적 최소 전력 스케줄링 방법의 전체적인 과정을 나타낸 흐름도이다.
- [0021] 도 1을 참조하여 본 발명에 의한 방법을 개략적으로 설명하면 다음과 같다. 즉, 우선 멀티 코어 프로세서의 전체 코어들 중에서 M개의 주기적 실시간 작업을 수행하는데 사용될 a개의 코어들을 결정한다(S100). 그리고, M개의 주기적 실시간 작업들을 수행하는데 사용될 a개의 코어들이 결정되면 결정된 a개의 코어들을 제외한 나머지 코어들의 전원을 소등한다(S110).
- [0022] 본 발명에서 멀티 코어 프로세서는 복수개 즉, 적어도 2 이상의 코어(core)를 갖는 것으로 하며 수행해야 할 주기적 실시간 작업(real-time tasks)들은 M개(여기서 M은 2이상의 자연수)인 것으로 가정한다. 본 발명은 이러한 M개의 주기적 실시간 작업들을 복수개의 코어들 중에서 a개(여기서, a는 1 이상의 자연수)의 코어를 선택하고 선택된 코어들에서 M개의 주기적 실시간 작업을 수행할 수 있도록 한다.
- [0023] 복수개의 코어들 중에서 M개의 주기적 실시간 작업들을 수행할 a개의 코어를 선택하는 것은 다음과 같은 원리에 기초하여 이루어진다.
- [0024] 알려져 있는 바와 같이, DVFS(Dynamic Voltage and Frequency Scaling) 기법에서는 프로세싱 코어에 적용되는 전압(Voltage)을 증가시키면 클럭 주파수(Clock Frequency)가 정비례하여 증가하고 또한 프로세싱 코어의 실행 속도도 정비례하여 증가한다. 프로세싱 코어에 적용되는 전압값(클럭 주파수 또는 코어의 실행 속도)을 증가시키면 에너지 소모량도 비례하여 증가하고 일반적으로 에너지 소모량 증가율이 전압값 증가율보다 크다. 다만, 실제 DVFS 기법 시스템 환경에서는 코어에 적용가능한 클럭 주파수가 한정되어 있고 각각의 에너지 소모량도 불규칙하게 증가한다.
- [0025] 이러한 DVFS 기법을 적용한 프로세서에서는 코어의 동작 속도(클럭 주파수)가 높아지면 단위 시간당 전력 소모량($P(f)$)은 클럭 주파수(f)의 세제곱(f^3)에 비례한다. ($P(f) \propto f^3$) 이 때, 클럭 주파수(f)가 0($f=0$)인 경우 즉, CPU가 아이들(idle) 상태인 경우에도 상수의 누수 전력 소모가 발생하며 이를 b라 표기한다.
- [0026] 이렇게 하면 단위 시간당 전력 소모량 $P(f)$ 는 다음과 같이 나타낼 수 있다.
- [0027]
$$P(f) = a \cdot f^3 + b$$
- [0028] (여기서 a, b는 상수)
- [0029] 이 때 전원을 소등하지 않으면 $P(0) = b$ 이고, 전원을 소등하면 $P(0) = 0$ 이다.
- [0030] $P(f) = a \cdot f^3$ 인 경우에는 모든 코어들을 사용하는 것이 전력 소모량을 최소화하지만, $P(f) = a \cdot f^3 + b$ 인 경우에는 ($b > 0$) 일부 코어들만을 사용하는 것이 전력 소모량을 최소화할 수 있다.
- [0031] 이를 설명하기 위하여 M개의 실시간 작업들의 단일 코어상에서의 확률적 전력 소모량 최소 기대값에 대해 설명한다.
- [0032] 주어진 M개의 실시간 작업 중에서 m번째의 작업을 T^m 이라 하고, m번째 작업의 최대(worst case) 계산량은 $w(m)$ 이라고 한다. 또한 m번째 작업의 도착 주기(데드라인, deadline)를 D^m 이라 한다.
- [0033] 각 작업(T^m)에 대해 종료시까지 소요되는 계산량은 확률 분포에 의해 추정하는데, m번째 작업의 불확실한

계산량의 확률 분포를 Φ^m 이라 하고 m번째 작업의 c번째 계산의 실행 확률을 Φ_c^m 라 한다.

[0034] 도 2 내지 도 4는 변동하는 불확실한 계산량의 확률 분포 모델을 설명하기 위한 도면이다. 도 2는 과거 계산량 확률 분포도이고, 도 3은 도 2의 확률 분포의 계산량 누적 확률 분포도, 도 4는 미래 계산량 확률 분포도를 나타낸 것이다.

[0035] 도 2 내지 도 4를 참조하면, c번째 주기(c^{th} cycle)에서의 확률은 P_c 로 나타낼 때, c번째 주기에서의 누적

확률(cumulative probability)은 $\int_0^c p_x dx$ 이며 또한 c번째 주기에서의 테일 누적 확률(tail probability)

은 $\Phi_c = 1 - \int_0^c p_x dx$ 이다. 따라서, 작업(T^m)의 c번째 주기에서의 테일 누적 확률을

$\Phi_c^m = (1 - \int_0^c p_x^m dx)$ 으로 나타낼 수 있다. 이는 바꾸어 말하면 c번째 진행된 주기에서 작업(T^m)이 아직 수행되고 있을 확률을 의미한다.

[0036] 이상과 같은 방법에 의하여 M개의 실시간 작업(T^m) 각각에 대하여 각 작업의 테드라인(D^m) 이전에 수행해야 할 계산량을 예측하기 위한 확률적 계산량 모델(Φ_c^m)을 결정할 수 있다.

[0037] 도 5 및 도 6은 확률적 계산량 모델 활용 방법을 설명하기 위한 도면이다.

[0038] 도 5는 기존의 방법을 나타낸 것으로서 최대 계산량(worst case)을 테드라인 이내에 수행을 완료하도록 프로세서 코어의 실행 속도(클럭 주파수)를 결정하는 경우를 나타낸 것이다. 대부분의 경우 실제 계산량이 최대 계산량보다 적기 때문에 낮은 실행 속도를 적용해서 전력 소모율을 줄일 수 있는 기회를 활용하지 못한다.

[0039] 도 6은 미래 계산량 확률(P_t)에 근거하여 수행될 확률이 높은 전반부는 낮은 실행 속도로 수행하고 확률이 낮은 후반부는 높은 실행 속도로 수행하는 경우이다. 계산량의 확률(Φ_c)에 따라 전체 전력 소모량 확률적 기대값을 최소화하도록 계산량의 확률에 따라 실행 속도(클럭 주파수)를 적용한 경우이다. 최대 계산량 종료 시간은 도 5 및 도 6 모두 테드라인 시간에 종료한다.

[0040] 이러한 상태에서 다수의 실시간 작업들을 단일 코어상에서 수행하기 위한 확률적 최소 전력 스케줄링 방법은 실행 확률이 높은 계산 진행 전반부는 낮은 실행 속도(클럭 주파수)를 적용하고 실행 확률이 낮은 계산 진행 후반부는 갈수록 높은 실행 속도를 적용하고, 또한 테드라인이 짧은 순서대로 실행한다는 점을 기본 동작 원리로 한다.

[0041] 앞서 설명한 바와 같이, 단위 시간당 전력 소모량 $P(f)$ 는 $P(f) = a \cdot f^3 + b$ (여기서 a, b는 상수)이고, m번째 작업의 c번째 동작 명령(클럭 사이클)에 적용된 클럭 주파수 값은 f_c^m 으로 나타낼 수 있다.

[0042] 이 때, 확률적 최소 전력 스케줄링 문제는 수행 완료 테드라인을 만족하면서 확률적 전력 소모량 기대값을 최소화하는 볼 수 있다. 이는 다음과 같은 수식으로 표현할 수 있다.

[0043] 수행완료 데드라인의 만족 :

$$\sum_{m=1}^M \left(\sum_{c=1}^{w(m)} (1/f_c^m)/D^m \right) \leq 1$$

[0044]

[0045] 확률적 전력 소모량 기대값 :

$$a \cdot \sum_{m=1}^M \sum_{c=1}^{w(m)} \Phi_c^m \cdot (f_c^m)^3 \cdot (1/f_c^m) \cdot (1/D^m) + b$$

[0046]

[0047] 즉,

$$\sum_{m=1}^M \sum_{c=1}^{w(m)} \Phi_c^m \cdot (f_c^m)^3$$

[0048] minimize

$$\sum_{m=1}^M \left(\sum_{c=1}^{w(m)} (1/f_c^m)/D^m \right) \leq 1$$

[0049] subject to

[0050] 로 나타낼 수 있다.

[0051] 상기 수식에 대한 해법은 다음과 같이 구할 수 있다.

[0052] m번째 작업의 c번째 동작 명령(클럭 사이클)에 적용된 클럭 주파수값(f_c^m)은 다음과 같이 구할 수 있다.

$$f_c^m = \sum_{m=1}^M \left(\sum_{i=1}^{w(m)} (\Phi_i^m)^{1/3}/D^m \right) / (\Phi_c^m)^{1/3}$$

[0053]

[0054] 여기에서 구해진 f_c^m 을 적용할 때 단일 코어의 확률적 전력 소모량 최소 기대값은 다음과 같다.

$$a \cdot \left(\sum_{m=1}^M \left(\sum_{i=1}^{w(m)} (\Phi_i^m)^{1/3}/D^m \right) \right)^3 + b$$

[0055]

$$\sum_{i=1}^{w(m)} (\Phi_i^m)^{1/3}/D^m = Q^m$$

[0056] 여기에서, 로 표기하면, 위 수식은

$$a \cdot \left(\sum_{m=1}^M Q^m \right)^3 + b = a \cdot (Q^1 + Q^2 + \dots + Q^M) + b$$

[0057]

[0058] 로 나타낼 수 있다.

[0059] 여기에서 앞서 설명한 바와 같이 m번째 작업의 c번째 계산의 실행 확률은 Φ_m^c 이고 m번째 작업의 도착주기(데드라인)은 D_m 이다.

[0060] M개 작업이 할당된 코어에 추가 실시간 작업 $T^{(M+1)}$ 을 실행할 때 단일 코어의 확률적 전력 소모량 최소 기대값은 다음과 같다.

$$a \cdot (Q^1 + Q^2 + \dots + Q^M + Q^{(M+1)})^3 + b = A$$

[0062] 새로운 코어에 추가 실시간 작업 $T^{(M+1)}$ 을 실행할 때 두개 코어들의 확률적 전력 소모량 최소 기대값은 다음과 같다.

$$(a \cdot (Q^1 + Q^2 + \dots + Q^M)^3 + b) + (a \cdot (Q^{(M+1)})^3 + b) = B$$

[0064] 여기에서 $A < B$ 라면, 기존에 작업이 할당된 코어에 추가 작업을 실행하고 나머지 코어의 전원을 소등하는 것이 전력 소모량을 줄이는 것임을 알 수 있다.

[0065] 따라서, 본 발명에서는 이러한 원리에 기초하여 복수개의 코어중에서 주어진 M개의 실시간 작업들을 수행할 a개의 코어들을 결정하고 결정되지 않은 나머지 코어들의 전원을 소등한다(도 1의 단계(S100), 단계(S110)).

[0066] 멀티 코어 프로세서의 전체 코어들 중에서 M개의 실시간 작업의 수행을 위해 사용할 a개의 코어를 선택하는 방법은 다음과 같다.

[0067] 우선 하기 수식과 같은 단일 코어에 할당된 i개 작업들의 확률적 전력 소모량 최소 기대값 수식 함수 MP를 활용한다.

$$MP(Q^1 + \dots + Q^i) = a(Q^1 + \dots + Q^i)^3 + b$$

[0069] 도 7은 상기 함수 MP를 이용하여 사용할 코어를 결정하는 방법을 설명하기 위한 도면으로서, 도 7에서와 같이 상기 3차 함수 MP와 정점을 지나는 1차 함수($\beta \cdot Q$)가 만나는 x지점을 찾는다.

[0070] 주어진 모든 M개의 작업들에 대해서 $(\sum_{m=1}^M Q^M)/x = \alpha$ 라면, α 개의 코어들만을 사용하고 나머지 코어들의 전원을 소등한다.

[0071] a개의 코어들에게 $\sum_{m=1}^M Q^m$ 을 균등하게 분배하면 각 코어는 MP(x)의 전력 소모량을 가지며 전체 코어들의 전력 소모량은 $a \cdot MP(x)$ 는 최소값을 갖게 된다.

[0072] 여기에서, a가 정수 소수라면 a의 올림 정수 또는 내림 정수값을 선택한다.

[0073] 이와 같은 과정을 통해 a개의 코어들을 선택할 수 있게 되며 이들 이외의 나머지 코어들의 전원을 소등할 수 있게 된다.

[0074] 다시 도 1을 참조하여 단계(S120) 내지 단계(S130)에 대하여 설명한다. 도 2 내지 도 7을 참조하여 단계(S100) 및 단계(S110)에 대해 설명한 바와 같이 a개의 코어들을 결정할 수 있으며, 결정된 코어들 즉, 상기 단계(S100)에서 M개의 주기적 실시간 작업을 수행하는데 사용하는 것으로 결정된 a개의 코어들에게 M개의 주기적 실시간 작업들을 할당한다(S120).

[0075] 결정된 a개의 코어들에게 할당하는 방법은 다음과 같다.

[0076] 우선 모든 M개의 작업들의 Q값을 계산하여 이를 내림차 순으로 정렬한다. 다음으로 정렬된 순서에 따라 각 작업을 하나의 코어에 할당하며 할당될 코어를 선택하는 방법은 각 코어들에게 기존에 할당된 작업들의 $\sum Q$

값이 최소인 코어를 선택하여 할당한다.

[0077] 모든 M개 작업들의 할당이 완료되면, 다음으로, α 개의 각 코어에서 상기 M개의 주기적 실시간 작업들을 각각 수행할 때 전력 소모량 확률적 기대값을 최소화하는 실행 속도(클럭 주파수)를 적용한다(S130). 즉, 모든 M개의 작업들의 할당이 완료되면 주어진 확률적 계산량 모델을 이용하여 앞서 설명한 바와 같은 전력 소모량 확률

적 기대값을 최소화하는 클럭 주파수(f_c^m)를 적용한다. 클럭 주파수(f_c^m)는 도 2 내지 도 6을 참조하여 설

명한 바와 같은 원리에 기초하여 구할 수 있다. 또는 종래 알려진 기술에 의해 클럭 주파수(f_c^m)를 구할 수 있다. 클럭 주파수를 구하는 방법은 본 출원인에 의해 출원된 대한민국 특허출원 제10-2011-0135518호

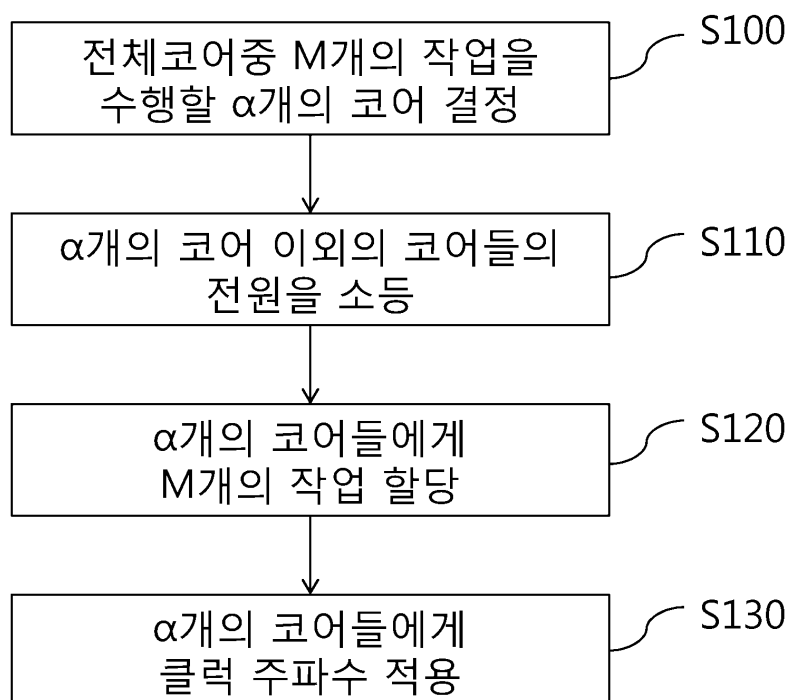
(2011.12.15. 출원, 미공개)에 개시된 방법을 사용할 수도 있다. 본 발명은 클럭 주파수(f_c^m) 자체를 구하는 것을 목적으로 하는 것은 아니며 종래 알려진 방법을 그대로 사용할 수 있으므로 이에 대한 상세 설명은 생략한다.

[0078] 이와 같은 과정을 수행하면 선택된 α 개의 코어들에서 수행할 작업들이 할당되고 할당된 작업들을 수행할 때 전력 소모량 확률적 기대값을 최소화하는 클럭 주파수를 적용하여 작업을 수행함으로써 멀티코어 프로세서에서 누수 전력 소모를 고려하여 확률적 모델에 기초한 최소 전력 스케줄링 방법을 제공할 수 있게 된다.

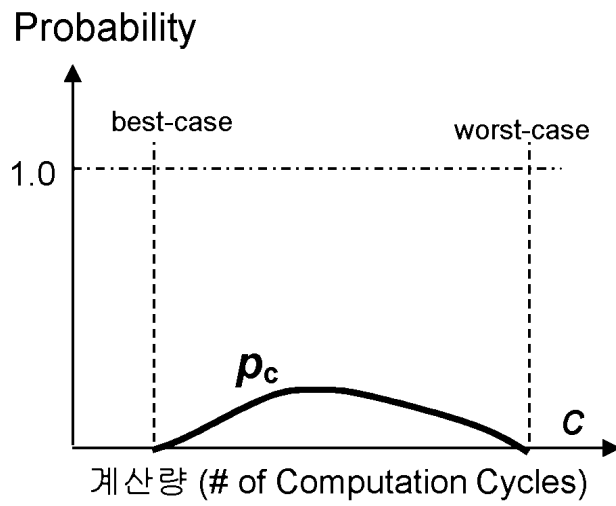
[0079] 이상에서, 바람직한 실시예들을 참조하여 본 발명을 설명하였으나, 본 발명은 상기의 실시예들에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상의 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능할 것이다. 따라서 본 발명은 첨부한 특허청구범위 및 도면 등의 전체적인 기재를 참조하여 해석되어야 할 것이며, 이의 균등 또는 등가적 변형 모두는 본 발명 사상의 범주에 속한다고 할 것이다.

도면

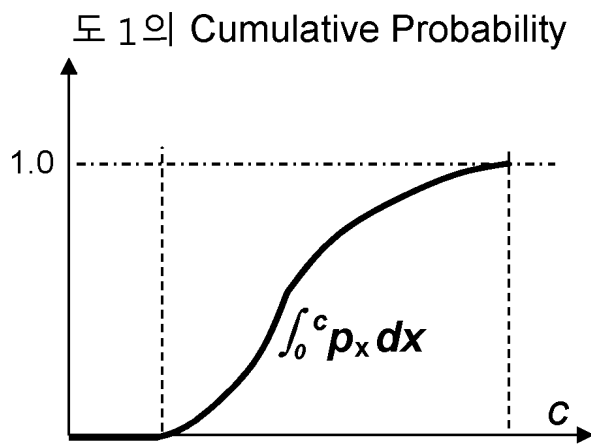
도면1



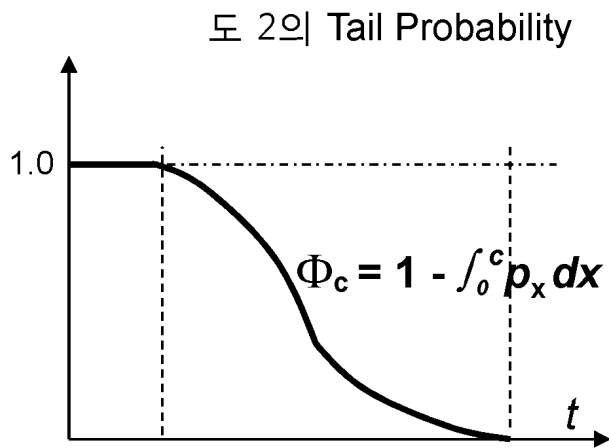
도면2



도면3

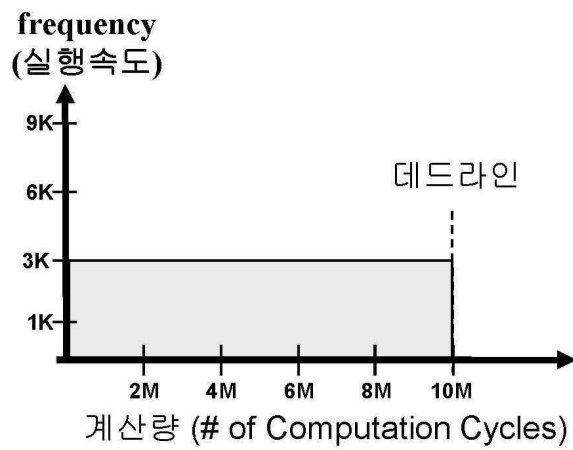


도면4

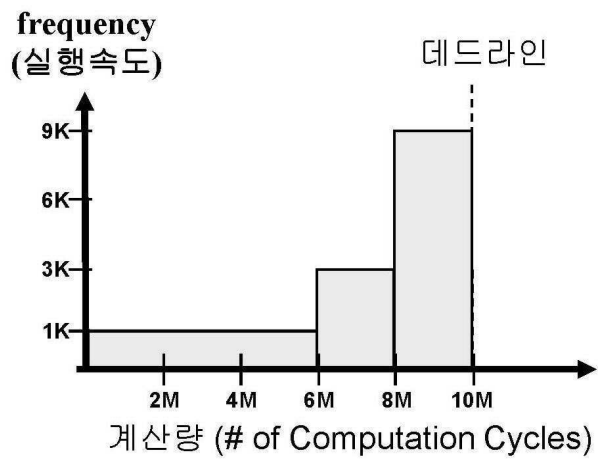


도면5

worst-case 계산량: 10M clock cycles



도면6



도면7

