



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년11월13일
 (11) 등록번호 10-1461459
 (24) 등록일자 2014년11월07일

(51) 국제특허분류(Int. Cl.)
 G06F 1/32 (2006.01)
 (21) 출원번호 10-2013-7009528
 (22) 출원일자(국제) 2013년07월08일
 심사청구일자 2013년04월15일
 (85) 번역문제출일자 2013년04월15일
 (65) 공개번호 10-2013-0044374
 (43) 공개일자 2013년05월02일
 (86) 국제출원번호 PCT/US2011/043285
 (87) 국제공개번호 WO 2012/036777
 국제공개일자 2012년03월22일
 (30) 우선권주장
 12/883,426 2010년09월16일 미국(US)
 (56) 선행기술조사문헌
 US20040139302 A1*
 *는 심사관에 의하여 인용된 문헌

(73) 특허권자
켈컴 인코포레이티드
 미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
 (72) 발명자
살스베리, 브라이언, 제이.
 미국 92121 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
가르가쉬, 노르만, 에스.
 미국 92121 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
 (74) 대리인
특허법인 남앤드남

전체 청구항 수 : 총 21 항

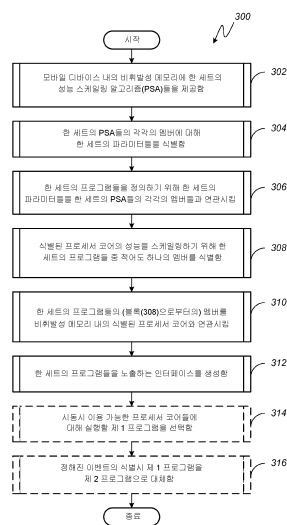
심사관 : 김근희

(54) 발명의 명칭 **한 세트의 성능 스케일링 알고리즘들을 노출 및 관리하기 위한 모바일 디바이스 및 방법**

(57) 요약

모바일 디바이스, 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법, 및 컴퓨터 프로그램 물건이 개시된다. 모바일 디바이스는 비휘발성 메모리에 통신 가능하게 연결된 다중 코어 프로세서를 포함한다. 상기 비휘발성 메모리는 성능 스케일링 알고리즘과 한 세트의 파라미터들의 각각의 조합으로 정의된 한 세트의 프로그램들, 상기 다중 코어 프로세서에 의한 실행시, 상기 한 세트의 프로그램들 중에서, 상기 모바일 디바이스의 동작을 모니터링하고 식별된 프로세서 코어의 성능을 스케일링하기에 적합한 적어도 하나의 멤버를 식별하는 시동 프로그램, 및 상기 한 세트의 프로그램들을 노출하는 애플리케이션 프로그래밍 인터페이스를 포함한다.

대표도 - 도3



특허청구의 범위

청구항 1

다중 코어 프로세서를 포함하는 모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS: dynamic voltage and frequency scaling) 동작들을 수행하는 방법으로서,

상기 다중 코어 프로세서 내의 각각의 프로세서 코어의 동작 주파수를 스케일링하기 위해 디폴트 DVFS 프로그램을 상기 다중 코어 프로세서 내의 각각의 프로세서 코어에 적용하는 단계;

상기 모바일 디바이스에서 애플리케이션 프로그래밍 인터페이스를 통해 복수의 DVFS 프로그램들을 수신하는 단계;

상기 다중 코어 프로세서 내의 각각의 프로세서 코어에 대해, 상기 수신된 복수의 DVFS 프로그램들 중 각각의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위한 적어도 하나의 DVFS 프로그램을 식별하는 단계;

각각의 개별적인 프로세서 코어와 관련하여(in association with) 그리고 각각의 개별적인 프로세서 코어에 대한 계층적(hierarchical) 순서로 비휘발성 메모리에 상기 식별된 DVFS 프로그램들의 각각을 저장하는 단계;

적어도 하나의 프로세서 코어에 대해 상기 계층적 순서를 가로지르는(traverse) 단계;

상기 계층적 순서를 가로지를 때 맞닥뜨리는(encountered) 제 1 DVFS 프로그램을 선택하는 단계; 및

상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 모바일 디바이스의 시동 루틴(startup routine)의 부분으로서 상기 선택된 제 1 DVFS 프로그램을 개별적인 프로세서 코어에 적용하는 단계를 포함하는,

모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS) 동작들을 수행하는 방법.

청구항 2

제 1 항에 있어서,

상기 모바일 디바이스에 대한 사용자-인지가능한(user-perceivable) 성능 영향을 갖지 않고 상기 프로세서 코어의 전력 소비를 감소시키도록 구성된 전력 절약 알고리즘을 적용함으로써, 상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 저장된 DVFS 프로그램들 중 적어도 하나의 DVFS 프로그램을 개별적인 프로세서 코어에 적용하는 단계를 더 포함하는,

모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS) 동작들을 수행하는 방법.

청구항 3

삭제

청구항 4

제 1 항에 있어서,

상기 선택된 제 1 DVFS 프로그램이 대체(replacement) DVFS 프로그램으로 대체되어야 함을 표시하는 조건을 검출하기 위해 상기 모바일 디바이스를 모니터링하는 단계; 및

상기 선택된 제 1 DVFS 프로그램을, 상기 조건을 검출하는 것에 응답하여 상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 2 DVFS 프로그램으로 대체하는 단계를 더 포함하는,

모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS) 동작들을 수행하는 방법.

청구항 5

제 1 항에 있어서,

상기 모바일 디바이스에서 상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수

신하는 단계는,

셀룰러 전화, 호출기, 개인용 디지털 보조기기, 리더(reader), 위성 전화, 네비게이션 디바이스, 및 무선 접속성을 가진 컴퓨터 중 하나에서 상기 복수의 DVFS 프로그램들을 수신하는 단계를 포함하는,

모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS) 동작들을 수행하는 방법.

청구항 6

제 1 항에 있어서,

상기 모바일 디바이스에서 상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하는 단계는,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 제조업체로부터 상기 복수의 DVFS 프로그램들을 수신하는 단계를 포함하는,

모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS) 동작들을 수행하는 방법.

청구항 7

제 1 항에 있어서,

상기 모바일 디바이스에서 상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하는 단계는,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 운영자로부터 상기 복수의 DVFS 프로그램들을 수신하는 단계를 포함하는,

모바일 디바이스 상에서 동적 전압 및 주파수 스케일링(DVFS) 동작들을 수행하는 방법.

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

모바일 디바이스로서,

메모리; 및

상기 메모리에 연결된 다중 코어 프로세서를 포함하며,

상기 다중 코어 프로세서의 적어도 하나의 프로세서 코어는,

디폴트 동적 전압 및 주파수 스케일링(DVFS) 프로그램이 적용되는 상기 적어도 하나의 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 디폴트 DVFS 프로그램을 상기 다중 코어 프로세서 내의 상기 적어도 하나의 프로세서 코어에 적용하는 동작;

애플리케이션 프로그래밍 인터페이스를 통해 복수의 DVFS 프로그램들을 수신하는 동작;

각각의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위한 것으로서 상기 수신된 복수의 DVFS 프로그램들 중 적어도 하나의 DVFS 프로그램을 식별하는 동작;

상기 모바일 디바이스의 개별적인 프로세서 코어와 관련하여 그리고 각각의 개별적인 프로세서 코어에 대한 계층적 순서로 상기 메모리에 상기 식별된 DVFS 프로그램들의 각각을 저장하는 동작;

상기 적어도 하나의 프로세서 코어에 대해 상기 계층적 순서를 가로지르는 동작;

상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 1 DVFS 프로그램을 선택하는 동작; 및

상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 모바일 디바이스의 시동 루틴의 부분으로서 상기 선택된 제 1 DVFS 프로그램을 개별적인 프로세서 코어에 적용하는 동작을 포함하는 동작들을 수행하기 위해 프로세서-실행가능한 명령들로 구성되는, 모바일 디바이스.

청구항 30

제 29 항에 있어서,

상기 다중 코어 프로세서의 적어도 하나의 프로세서 코어는,

상기 모바일 디바이스에 대한 사용자-인지가능한 성능 영향을 갖지 않고 상기 모바일 디바이스의 전력 소비를 감소시키도록 구성된 전력 절약 알고리즘을 적용함으로써, 상기 저장된 DVFS 프로그램들 중 적어도 하나의 DVFS 프로그램을 상기 모바일 디바이스의 개별적인 프로세서 코어에 적용하는 동작

을 추가로 포함하는 동작들을 수행하기 위해 프로세서-실행가능한 명령들로 구성되는,

모바일 디바이스.

청구항 31

제 29 항에 있어서,

상기 다중 코어 프로세서의 적어도 하나의 프로세서 코어는,

상기 선택된 제 1 DVFS 프로그램이 대체 DVFS 프로그램으로 대체되어야 함을 표시하는 모바일 디바이스 조건을 검출하는 동작; 및

상기 선택된 제 1 DVFS 프로그램을, 상기 모바일 디바이스 조건을 검출하는 것에 응답하여 상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 2 DVFS 프로그램으로 대체하는 동작

을 추가로 포함하는 동작들을 수행하기 위해 프로세서-실행가능한 명령들로 구성되는,

모바일 디바이스.

청구항 32

제 29 항에 있어서,

상기 다중 코어 프로세서의 적어도 하나의 프로세서 코어는,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하는 동작이 상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 제조업체로부터 상기 복수의 DVFS 프로그램들을 수신하는 동작을 포함하도록 동작들을 수행하기 위해 프로세서-실행가능한 명령들로 구성되는,

모바일 디바이스.

청구항 33

제 29 항에 있어서,

상기 다중 코어 프로세서의 적어도 하나의 프로세서 코어는,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하는 동작이 상기 애플리

케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 운영자로부터 상기 복수의 DVFS 프로그램들을 수신하는 동작을 포함하도록 동작들을 수행하기 위해 프로세서-실행가능한 명령들로 구성되는,

모바일 디바이스.

청구항 34

다중 코어 프로세서를 포함하는 모바일 디바이스로서,

상기 다중 코어 프로세서 내의 각각의 프로세서 코어의 동작 주파수를 스케일링하기 위해 디폴트 동적 전압 및 주파수 스케일링(DVFS) 프로그램을 각각의 프로세서 코어에 적용하기 위한 수단;

애플리케이션 프로그래밍 인터페이스를 통해 복수의 DVFS 프로그램들을 수신하기 위한 수단;

각각의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위한 것으로서 상기 수신된 복수의 DVFS 프로그램들 중 적어도 하나의 DVFS 프로그램을 식별하기 위한 수단;

상기 모바일 디바이스의 개별적인 프로세서 코어와 연관하여 그리고 각각의 개별적인 프로세서 코어에 대한 계층적 순서로 상기 식별된 DVFS 프로그램들의 각각을 저장하기 위한 수단;

적어도 하나의 프로세서 코어에 대해 상기 계층적 순서를 가로지르기 위한 수단;

상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 1 DVFS 프로그램을 선택하기 위한 수단; 및

상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 모바일 디바이스의 시동 루틴의 부분으로서 상기 선택된 제 1 DVFS 프로그램을 개별적인 프로세서 코어에 적용하기 위한 수단을 포함하는,

모바일 디바이스.

청구항 35

제 34 항에 있어서,

상기 모바일 디바이스에 대한 사용자-인지가능한 성능 영향을 갖지 않고 상기 프로세서 코어의 전력 소비를 감소시키도록 구성된 전력 절약 알고리즘을 적용함으로써, 상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 저장된 DVFS 프로그램들 중 적어도 하나의 DVFS 프로그램을 개별적인 프로세서 코어에 적용하기 위한 수단을 더 포함하는,

모바일 디바이스.

청구항 36

제 34 항에 있어서,

상기 선택된 제 1 DVFS 프로그램이 대체 DVFS 프로그램으로 대체되어야 함을 표시하는 조건을 검출하기 위한 수단; 및

상기 선택된 제 1 DVFS 프로그램을, 상기 조건을 검출하는 것에 응답하여 상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 2 DVFS 프로그램으로 대체하기 위한 수단을 더 포함하는,

모바일 디바이스.

청구항 37

제 34 항에 있어서,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하기 위한 수단은,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 제조업체로부터 상기 복수의 DVFS 프로그램들을 수신하기 위한 수단을 포함하는,

모바일 디바이스.

청구항 38

제 34 항에 있어서,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하기 위한 수단은,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 운영자로부터 상기 복수의 DVFS 프로그램들을 수신하기 위한 수단을 포함하는,

모바일 디바이스.

청구항 39

프로세서-실행가능한 소프트웨어 명령들을 저장한 컴퓨터-판독가능 저장 매체로서,

상기 저장된 프로세서-실행가능한 소프트웨어 명령들은 모바일 디바이스의 다중 코어 프로세서의 프로세서 코어로 하여금,

디폴트 동작 전압 및 주파수 스케일링(DVFS) 프로그램이 적용되는 적어도 하나의 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 디폴트 DVFS 프로그램을 상기 다중 코어 프로세서 내의 상기 적어도 하나의 프로세서 코어에 적용하는 동작;

애플리케이션 프로그래밍 인터페이스를 통해 복수의 DVFS 프로그램들을 수신하는 동작;

상기 수신된 복수의 DVFS 프로그램들 중 적어도 하나의 프로세서 코어의 동작 주파수를 스케일링하기 위한 적어도 하나의 DVFS 프로그램을 식별하는 동작;

상기 식별된 DVFS 프로그램들이 식별되는 프로세서 코어들과 관련하여 그리고 상기 식별된 DVFS 프로그램들이 식별되는 프로세서 코어들의 각각에 대한 계층적 순서로 상기 식별된 DVFS 프로그램들의 각각을 저장하는 동작;

상기 계층적 순서를 가로지르는 동작;

상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 1 DVFS 프로그램을 선택하는 동작; 및

상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 시동 루틴의 부분으로서 상기 선택된 제 1 DVFS 프로그램을 개별적인 프로세서 코어에 적용하는 동작

을 포함하는 동작들을 수행하게 하기 위해 구성되는,

컴퓨터-판독가능 저장 매체.

청구항 40

제 39 항에 있어서,

상기 저장된 프로세서-실행가능한 소프트웨어 명령들은 상기 프로세서 코어로 하여금,

상기 모바일 디바이스에 대한 사용자-인지가능한 성능 영향을 갖지 않고 상기 모바일 디바이스 상의 전력 소비를 감소시키도록 구성된 전력 절약 알고리즘을 적용함으로써, 상기 모바일 디바이스의 개별적인 프로세서 코어의 동작 주파수를 스케일링하기 위해 상기 저장된 DVFS 프로그램들 중 적어도 하나의 DVFS 프로그램을 개별적인 프로세서 코어에 적용하는 동작

을 추가로 포함하는 동작들을 수행하게 하기 위해 구성되는,

컴퓨터-판독가능 저장 매체.

청구항 41

제 39 항에 있어서,

상기 저장된 프로세서-실행가능한 소프트웨어 명령들은 상기 프로세서 코어로 하여금,

상기 선택된 제 1 DVFS 프로그램이 대체 DVFS 프로그램으로 대체되어야 함을 표시하는 조건을 검출하는 동작;

및

상기 선택된 제 1 DVFS 프로그램을, 상기 조건을 검출하는 것에 응답하여 상기 계층적 순서를 가로지를 때 맞닥뜨리는 제 2 DVFS 프로그램으로 대체하는 동작

을 추가로 포함하는 동작들을 수행하게 하기 위해 구성되는,

컴퓨터-판독가능 저장 매체.

청구항 42

제 39 항에 있어서,

상기 저장된 프로세서-실행가능한 소프트웨어 명령들은 상기 프로세서 코어로 하여금,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하는 동작이 상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 제조업체로부터 상기 복수의 DVFS 프로그램들을 수신하는 동작을 포함하도록 동작들을 수행하게 하기 위해 구성되는,

컴퓨터-판독가능 저장 매체.

청구항 43

제 39 항에 있어서,

상기 저장된 프로세서-실행가능한 소프트웨어 명령들은 상기 프로세서 코어로 하여금,

상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 복수의 DVFS 프로그램들을 수신하는 동작이 상기 애플리케이션 프로그래밍 인터페이스를 통해 상기 모바일 디바이스의 운영자로부터 상기 복수의 DVFS 프로그램들을 수신하는 동작을 포함하도록 동작들을 수행하게 하기 위해 구성되는,

컴퓨터-판독가능 저장 매체.

명세서

배경 기술

- [0001] 무선 통신 프로토콜들, 에너지 저장 밀도, 프로세서 성능 및 효율, 그리고 셀룰러 송신기들의 밀도 증가에 있어 기술적 진보들은 1990년대에 무선 디지털 네트워크들의 도입을 가능하게 했다. 2세대 모바일 디바이스들의 사용이 더 확산됨에 따라, 인터넷에 대한 액세스를 포함하는 데이터 서비스들에 대한 요구가 증가하고 있음이 명백해졌다. 3세대 무선 통신 시스템들을 이용한 데이터 송신을 위해 회선 교환보다는 패킷 교환의 도입이 스트리밍 미디어를 모바일 디바이스에 전달하는데 요구되는 데이터 송신 레이트들의 증가들을 가능하게 했다.
- [0002] 오늘날의 데이터 네트워크 가능 모바일 디바이스들은 최종 사용자에게 다양한 애플리케이션들을 제공하기 위한 고성능 프로세서들을 포함한다. 그러나 성능에 대한 요구는 그에 상응하는 전력 소비의 증가를 초래한다. 최신 프로세서들의 전력 소실은 주어진 구현에 요구되는 트랜지스터들의 수와 클럭 주파수 모두가 증가함에 따라 빠르게 증가하고 있다. 유감스럽게도, 배터리 기술은 이러한 데이터 네트워크 가능 무선 디바이스들의 에너지 요건들과 보조를 맞추지 못해왔고, 이는 애플리케이션 성능과 배터리 수명 간의 설계 트레이드 오프로 이어진다. 최대 성능을 위해 배터리 수명이 희생될 수 있다고 결정되는 경우에도, 흔히 손 크기 하우징들 내의 배터리와 전자 회로들 모두에 대해 열 관리가 문제가 될 수 있다.
- [0003] 고성능 및 저 전력 소비를 제공하는 프로세서들에 대한 요구는 프로세서 설계들에 동적 전압 및 주파수 스케일링(DVFS: dynamic voltage and frequency scaling)의 사용을 이끌어 왔다. DVFS는 전력 소비와 성능 간의 트레이드 오프들을 가능하게 한다. DVFS를 이용하도록 설계된 프로세서들은 프로세서의 클럭 주파수가 전압의 상응하는 조정과 함께 조정되게 한다. 클럭 주파수만을 감소시키는 것은 유용하지 않은데, 이는 어떠한 전력 절약이든 실행 시간 증가만큼 상쇄되어, 소비되는 총 에너지의 순 감소를 야기하지 않기 때문이다. 그러나 동작 전압의 감소는 소비되는 전력의 비례 절약을 야기한다.
- [0004] DVFS 가능 프로세서들에 대한 주요한 문제는 성능과 전력 절약 간의 균형을 어떻게 제어하는가 하는 것이다. 종래의 DVFS 제어 알고리즘은 예상 작업량이 기한까지 시간을 늘리도록 허용하는 수정된 성능 레벨을 식별하기

위해 유휴 시간의 측정을 사용한다. 즉, 더 낮은 전압 및 더 느린 클록 주파수로 프로세서를 작동시킴으로써 유휴 시간이 감소된다. 이러한 접근 방식은 연속적이고 알려져 있는 작업량에 대해서는 적절할 수 있지만, 많은 모바일 디바이스 애플리케이션들은 상호적이고 그리고/또는 프로세서에 불규칙적이거나 예측할 수 없는 요구들을 한다. 따라서 어떠한 알고리즘도 모든 모바일 디바이스 애플리케이션들, 애플리케이션 조합들 및 사용들에 대해 최선이 아님이 명백하다.

발명의 내용

- [0005] 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법은, 상기 모바일 디바이스 내의 비휘발성 메모리에 한 세트의 성능 스케일링 알고리즘들을 제공하는 단계, 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버들 각각에 대해 적절한 한 세트의 파라미터들을 식별하는 단계, 한 세트의 프로그램들을 정의하기 위해 상기 한 세트의 파라미터들을 비휘발성 메모리 내의 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버와 연관시키는 단계, 식별된 프로세서 코어의 성능을 스케일링하기에 적합한, 상기 한 세트의 프로그램들 중의 적어도 하나의 멤버를 식별하는 단계, 상기 비휘발성 메모리에서 상기 한 세트의 프로그램들 중 상기 적어도 하나의 멤버를 상기 식별된 프로세서 코어와 연관시키는 단계, 및 상기 한 세트의 프로그램들을 노출하는 인터페이스를 생성하는 단계를 포함한다.
- [0006] 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법의 대안적인 실시예는, 상기 모바일 디바이스 내의 비휘발성 메모리에 한 세트의 성능 스케일링 알고리즘들을 제공하는 단계 - 상기 한 세트의 성능 스케일링 알고리즘들을 제공하는 단계는 상기 모바일 디바이스에 대한 성능 영향을 최소화하면서 전력 절약 알고리즘들을 제공하는 단계를 포함함 -, 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버들 각각에 대해 적절한 한 세트의 파라미터들을 식별하는 단계, 한 세트의 프로그램들을 정의하기 위해 상기 한 세트의 파라미터들을 비휘발성 메모리 내의 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버와 연관시키는 단계, 식별된 프로세서 코어의 성능을 스케일링하기에 적합한, 상기 한 세트의 프로그램들 중의 적어도 하나의 멤버를 식별하는 단계, 상기 비휘발성 메모리에서 상기 한 세트의 프로그램들 중 상기 적어도 하나의 멤버를 상기 식별된 프로세서 코어와 연관시키는 단계, 및 상기 한 세트의 프로그램들을 노출하는 인터페이스를 생성하는 단계를 포함한다.
- [0007] 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법의 또 다른 대안적인 실시예는, 상기 모바일 디바이스 내의 비휘발성 메모리에 한 세트의 성능 스케일링 알고리즘들을 제공하는 단계, 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버들 각각에 대해 적절한 한 세트의 파라미터들을 식별하는 단계, 한 세트의 프로그램들을 정의하기 위해 상기 한 세트의 파라미터들을 비휘발성 메모리 내의 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버와 연관시키는 단계, 식별된 프로세서 코어의 성능을 스케일링하기에 적합한, 상기 한 세트의 프로그램들 중의 적어도 하나의 멤버를 식별하는 단계, 상기 비휘발성 메모리에서 상기 한 세트의 프로그램들 중 상기 적어도 하나의 멤버를 상기 식별된 프로세서 코어와 연관시키는 단계, 상기 한 세트의 프로그램들을 노출하는 인터페이스를 생성하는 단계, 및 상기 한 세트의 프로그램들 중에서 시동시 실행할 제 1 프로그램을 선택하는 단계를 포함한다.
- [0008] 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법의 또 다른 대안적인 실시예는, 상기 모바일 디바이스 내의 비휘발성 메모리에 한 세트의 성능 스케일링 알고리즘들을 제공하는 단계, 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버들 각각에 대해 적절한 한 세트의 파라미터들을 식별하는 단계, 한 세트의 프로그램들을 정의하기 위해 상기 한 세트의 파라미터들을 비휘발성 메모리 내의 상기 한 세트의 성능 스케일링 알고리즘들의 개개의 멤버와 연관시키는 단계, 식별된 프로세서 코어의 성능을 스케일링하기에 적합한, 상기 한 세트의 프로그램들 중의 적어도 하나의 멤버를 식별하는 단계, 상기 비휘발성 메모리에서 상기 한 세트의 프로그램들 중 상기 적어도 하나의 멤버를 상기 식별된 프로세서 코어와 연관시키는 단계, 상기 한 세트의 프로그램들을 노출하는 인터페이스를 생성하는 단계, 상기 한 세트의 프로그램들 중에서 시동시 실행할 제 1 프로그램을 선택하는 단계, 및 정해진 이벤트의 식별시 상기 제 1 프로그램을 제 2 프로그램으로 대체하는 단계를 포함한다.
- [0009] 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법의 다른 양상에 따르면, 상기 방법은 상기 한 세트의 프로그램들을 계층 순서로 배열하는 단계를 더 포함한다. 일부 실시예들에서, 상기 계층 순서는 가장 상세한 멤버들에서부터 가장 덜 상세한 멤버들까지로 구조화된다.
- [0010] 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 다양한 방법들의 한 양상에 따르면, 인터페이스는 상기 모바일 디바이스의 제조업체와 운영자 중 하나 또는 그보다 많은 것에 의해 액

세스 가능하다.

- [0011] 다른 양상에 따르면, 모바일 디바이스가 개시된다. 상기 모바일 디바이스는 비휘발성 메모리에 통신 가능하게 연결된 다중 코어 프로세서를 포함한다. 상기 비휘발성 메모리에는 성능 스케일링 알고리즘과 한 세트의 파라미터들의 각각의 조합으로 정의된 한 세트의 프로그램들, 상기 다중 코어 프로세서에 의한 실행시, 상기 한 세트의 프로그램들 중에서, 상기 모바일 디바이스의 동작을 모니터링하고 식별된 프로세서 코어의 성능을 스케일링하기에 적합한 적어도 하나의 멤버를 식별하는 시동 프로그램, 및 상기 한 세트의 프로그램들을 노출하는 애플리케이션 프로그래밍 인터페이스가 저장된다.
- [0012] 일부 예시적인 실시예들에서, 상기 모바일 디바이스는 셀룰러 전화, 호출기, 개인용 디지털 보조기기, 리더(reader), 위성 전화, 네비게이션 디바이스, 또는 무선 접속성을 가진 컴퓨팅 디바이스 중 하나이다.
- [0013] 예시적인 실시예에서, 상기 모바일 디바이스는 다수의 프로세서 코어들에 걸쳐 로드를 밸런싱하는 성능 스케일링 알고리즘을 포함한다.
- [0014] 대안적인 실시예에서, 상기 모바일 디바이스는 프로세서에 의한 실행시, 이벤트의 발생을 검출하고, 상기 이벤트에 응답하여 상기 한 세트의 프로그램들 중의 상기 적어도 하나의 멤버를 대체할, 상기 한 세트의 프로그램들 중의 제 2 멤버를 식별하고, 상기 식별된 프로세서 코어에서의 실행을 위해 상기 제 2 멤버를 로딩하고, 그리고 상기 한 세트의 프로그램들 중의 상기 제 2 멤버를 실행하도록 상기 식별된 프로세서 코어에 지시하는 관리 애플리케이션을 더 포함한다.
- [0015] 상기 모바일 디바이스의 다른 양상에 따르면, 상기 애플리케이션 프로그래밍 인터페이스는 상기 모바일 디바이스의 제조업체와 운영자 중 하나 또는 그보다 많은 것에 의해 액세스 가능하다.
- [0016] 상기 모바일 디바이스의 또 다른 양상에 따르면, 상기 비휘발성 메모리는 상기 한 세트의 프로그램들을 계층 순서로 저장한다. 일부 실시예들에서, 상기 계층 순서는 가장 상세한 멤버들에서부터 가장 덜 상세한 멤버들까지로 구조화된다.
- [0017] 더욱이, 모바일 디바이스 상에서 동작하는 한 세트의 성능 스케일링 알고리즘들을 관리하는 모바일 디바이스는 한 세트의 프로그램들을 정의하기 위해 한 세트의 성능 스케일링 알고리즘들을 파라미터들의 각각의 세트에 연관시키고 저장하기 위한 메커니즘들 - 상기 한 세트의 프로그램들 중 적어도 하나의 멤버는 식별된 프로세서 코어의 성능을 스케일링하기에 적합하고, 상기 연관시키고 저장하기 위한 메커니즘들은 상기 한 세트의 프로그램들 중에서 상기 모바일 디바이스 상에서 이용 가능한 프로세서 코어들 상에서의 실행에 적합한 멤버들을 식별하기 위한 각각의 표시자를 포함함 -, 및 상기 한 세트의 프로그램들을 노출하기 위한 메커니즘들을 포함한다.
- [0018] 한 세트의 성능 스케일링 알고리즘들을 관리하는 모바일 디바이스의 다른 양상에 따르면, 상기 성능 스케일링 알고리즘들 중 하나 또는 그보다 많은 성능 스케일링 알고리즘은 로드 밸런싱 알고리즘을 포함한다.
- [0019] 한 세트의 성능 스케일링 알고리즘들을 관리하는 모바일 디바이스의 또 다른 양상에 따르면, 상기 모바일 디바이스는 이벤트의 발생을 검출하기 위한 하나 또는 그보다 많은 메커니즘들, 상기 이벤트에 응답하여 상기 한 세트의 프로그램들 중의 상기 적어도 하나의 멤버를 대체할, 상기 한 세트의 프로그램들 중의 제 2 멤버를 식별하기 위한 하나 또는 그보다 많은 메커니즘들, 및 상기 제 2 멤버를 로딩하여 실행하기 위한 하나 또는 그보다 많은 메커니즘들을 더 포함한다.
- [0020] 한 세트의 성능 스케일링 알고리즘들을 관리하는 모바일 디바이스의 또 다른 양상에 따르면, 노출하기 위한 하나 또는 그보다 많은 메커니즘들은 상기 한 세트의 프로그램들을 상기 모바일 디바이스의 제조업체와 운영자 중 하나 또는 그보다 많은 것에 노출한다.
- [0021] 한 세트의 성능 스케일링 알고리즘들을 관리하는 모바일 디바이스의 또 다른 양상에 따르면, 연관시키고 저장하기 위한 하나 또는 그보다 많은 메커니즘들은 상기 한 세트의 프로그램들을 계층 순서로 배열한다. 일부 실시예들에서, 상기 계층 순서는 가장 상세한 멤버들에서부터 가장 덜 상세한 멤버들까지로 구조화된다.
- [0022] 또 다른 양상에서, 컴퓨터 판독 가능 프로그램 코드가 구제화된 컴퓨터 사용 가능 매체를 포함하는 컴퓨터 프로그램 물건이 개시된다. 상기 컴퓨터 판독 가능 프로그램 코드는 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리하기 위한 방법을 실행 및 구현하도록 적응된다. 상기 방법은 선택 프로그램을 정의하기 위해 상기 모바일 디바이스 상에서 동작 가능한 프로세서 코어의 성능을 스케일링하기에 적합한 멤버를 한 세트의 프로그램들 중에서 선택하는 단계 - 상기 한 세트의 프로그램들은 성능 스케일링 알고리즘과 파라미터들의 각각의 세트의 각각의 조합으로 정의됨 -, 시동 시퀀스에 응답하여 상기 모바일 디바이스 상에서 동작 가

능한 프로세서에 상기 선택 프로그램을 로딩하는 단계, 상기 한 세트의 프로그램들 중의 대체 프로그램이 상기 선택 프로그램을 대체해야 함을 나타내는 조건을 식별하기 위해 상기 모바일 디바이스의 사용을 모니터링하는 단계, 및 런타임 동안 상기 선택 프로그램을 상기 대체 프로그램으로 대체하는 단계를 포함한다.

[0023] 컴퓨터 프로그램 물건의 다른 양상에 따르면, 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리하고 구체적으로는 멤버를 선택하기 위한 방법은 계층적으로 배열된 파일 시스템을 탐색하는 단계를 포함한다.

[0024] 컴퓨터 프로그램 물건의 또 다른 양상에 따르면, 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리하고 구체적으로 상기 모바일 디바이스의 사용을 모니터링하기 위한 방법은, 활동중인 프로세스들의 리스트를 유지하는 단계를 포함한다.

[0025] 컴퓨터 프로그램 물건의 또 다른 양상에 따르면, 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리하고 구체적으로는 상기 모바일 디바이스의 사용을 모니터링하기 위한 방법은 활동중인 자원들의 리스트를 유지하는 단계를 포함한다.

[0026] 컴퓨터 프로그램 물건의 또 다른 양상에 따르면, 모바일 디바이스 상에서 한 세트의 성능 스케일링 알고리즘들을 관리하고 구체적으로 상기 모바일 디바이스의 사용을 모니터링하기 위한 방법은, 운영자 인터페이스로부터 입력을 수신하는 단계를 포함한다.

[0027] 모바일 디바이스 상에서 동작 가능한 한 세트의 성능 스케일링 알고리즘들을 노출 및 관리하기 위한 모바일 디바이스 및 방법들이 다음 도면들을 참조로 더 잘 이해될 수 있다. 도면들 내의 컴포넌트들이 반드시 비율이 맞아야 하는 것은 아니며, 대신에 알고리즘들 및 각각의 런타임 파라미터들을 포함하는 한 세트의 프로그램들을 관리하는 원리들을 명확히 설명하는데 강조를 둔다. 더욱이, 도면들에서, 동일한 참조 부호들이 서로 다른 시점들 전체에 걸쳐 해당 부분들을 명시한다.

도면의 간단한 설명

- [0028] 도 1은 모바일 디바이스의 실시예를 나타내는 개략도이다.
- 도 2는 도 1의 모바일 디바이스의 특징들을 나타내는 개략도이다.
- 도 3은 도 1의 모바일 디바이스 상에서 동작하는 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법의 실시예를 나타내는 흐름도이다.
- 도 4는 도 1의 모바일 디바이스 상에서 동작 가능한 컴퓨터 프로그램 물건의 실시예를 나타내는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0029] "예시적인"이라는 단어는 본 명세서에서 "예시, 실례 또는 예증으로서의 역할"을 의미하는데 사용된다. 본 명세서에서 "예시적인" 것으로서 설명되는 어떠한 양상도 다른 양상들에 비해 반드시 선호되거나 유리한 것으로 해석되는 것은 아니다.

[0030] 이 설명에서, "애플리케이션"이라는 용어는 또한 객체 코드, 스크립트들, 바이트 코드, 마크업 언어 파일들 및 패치들과 같은 실행 가능 콘텐츠를 갖는 파일들을 포함할 수도 있다. 추가로, 본 명세서에서 언급되는 "애플리케이션"은 또한 공개될 필요가 있을 수도 있는 문서들이나 액세스될 필요가 있는 다른 데이터 파일들과 같이, 본래 실행 가능하지 않은 파일들을 포함할 수도 있다.

[0031] 이 설명에서, "모바일 디바이스"라는 용어는 배터리와 같은 한정된 용량의 파워 서플라이로 작동하는 임의의 디바이스를 설명하는데 사용된다. 수십 년간 배터리 작동 모바일 디바이스들이 사용되어 왔지만, 3세대("3G") 무선 기술의 출현과 결부된 충전 가능한 배터리들의 기술적 진보들은 다양한 능력들을 가진 수많은 모바일 디바이스들을 가능하게 하였다. 따라서 모바일 디바이스는 무엇보다도, 셀룰러 전화, 위성 전화, 호출기, PDA, 스마트폰, 네비게이션 디바이스, 스마트북 또는 리더, 미디어 플레이어, 앞서 언급한 디바이스들의 조합, 무선 접속성을 가진 랩톱 컴퓨터일 수 있다.

[0032] 모바일 디바이스 상에 다양한 기능들을 구현하기 위한 다중 코어 프로세서들 및 전용 하드웨어 자원들의 도입 및 배터리 수명의 개선에 대한 지속적인 요구는 만족스러운 최종 사용자 체험을 제공하면서 전력 소비를 줄이도록 모바일 디바이스를 동적으로 조정하는 작업을 더 복잡하게 만들어왔다. 이에 대응하여, 다수의 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 개선된 모바일 디바이스 및 방법들이 개시된다. 이용 가능한 서로 다

른 모바일 디바이스들의 수, 디바이스들과 통신하는데 사용되는 다양한 통신 프로토콜들이나 표준들, 이용 가능한 서로 다른 타입들의 애플리케이션들, 디바이스가 사용되는 서로 다른 방식들 등은 모든 가능한 동작 조합들에 대해 어떠한 단일 접근 방식이나 솔루션도 조정 또는 최적화될 수 없다는 점을 명확히 한다.

[0033] 이에 대응하여, 모바일 디바이스에 대해 한 세트의 성능 스케일링 알고리즘들을 제공 및 구현하는 것이 제안된다. 한 세트의 성능 스케일링 알고리즘들의 이용 가능성은 선택 성능 스케일링 알고리즘들이 정확히 적용될 때 성능 영향을 최소화하면서 전력 절약 기회들을 제공한다. 한 세트의 성능 스케일링 알고리즘들을 다양한 모바일 디바이스들의 제조업체들에 노출하는 것이 추가로 제안된다. 제조업체들은 다양한 사용 조건들에 대해 원하는 성능을 기반으로 모바일 디바이스들에 대한 선택 또는 최적화된 성능 스케일링 솔루션들을 제공하는데 관심이 있을 것으로 여겨진다. 이러한 제조업체 선택 또는 최적화된 솔루션들은 시동시(at startup) 적용될 수 있다. 어떤 상황들에서, 제조업체들은 자신들의 고객들에게 각각의 모바일 디바이스들을 통해 성능 스케일링 대안들을 노출할 의향이 있을 수도 있다. 이러한 상황들에서, 제조업체는 최종 사용자가 시동시 적용될 수 있는 자신들의 원하는 성능 대 배터리 수명 트레이드 오프를 선택할 수 있게 할 애플리케이션 프로그래밍 인터페이스를 제공 또는 수정할 수 있거나 성능 스케일링 솔루션의 런타임 선택 및 대체를 실시간으로 제공할 수 있다. 이러한 대안적인 런타임 솔루션들은 모바일 디바이스 상에서의 모든 성능 스케일링 솔루션들의 완벽한 불능화를 포함할 수 있다.

[0034] 모바일 디바이스 상에서 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법들은 일반적으로 소프트웨어 또는 펌웨어와 하드웨어의 조합을 통해 구현된다. 소프트웨어와 펌웨어는 모바일 디바이스 상에서 동작 가능한 하나 또는 그보다 많은 프로세서들에 연결된 비휘발성 메모리 엘리먼트에 저장될 수 있다. 비휘발성 메모리 엘리먼트에 저장된 소프트웨어는 적당한 명령 실행 시스템(마이크로프로세서)에 전달되어 그 시스템에 의해 실행된다. 모바일 디바이스에 대한 하드웨어 구현은 다음의 기술들: 이산 전자 컴포넌트들, 데이터 신호들에 대한 로직 기능들을 구현하기 위한 로직 게이트들을 갖는 이산 로직 회로(들), 적절한 로직 게이트들을 갖는 주문형 집적 회로, 프로그래밍 가능 게이트 어레이(들)(PGA: programmable gate array), 필드 프로그래밍 가능 게이트 어레이(FPGA: field programmable gate array) 등 중에서 임의의 기술 또는 이들의 조합을 포함할 수 있으며, 이러한 기술들은 모두 해당 기술분야에 잘 알려져 있다.

[0035] 도 1을 참조하면, 이 도면은 무선 전화 형태인 모바일 디바이스(100)의 예시적인 비-한정 양상의 도면이다. 도시된 바와 같이, 모바일 디바이스(100)는 함께 연결된 디지털 신호 프로세서(110) 및 아날로그 신호 프로세서(126)를 포함하는 온-칩(on-chip) 시스템(102)을 포함한다. 도 1에 예시된 바와 같이, 디스플레이 제어기(128) 및 터치스크린 제어기(130)가 디지털 신호 프로세서(110)에 연결된다. 온-칩 시스템(102) 외부의 터치스크린 디스플레이(132)가 디스플레이 제어기(128) 및 터치스크린 제어기(130)에 연결된다.

[0036] 도 1은 또한, 비디오 인코더(134), 예를 들어 "PAL(phase-alternating line)" 인코더, "SECAM(sequential couleur avec memoire)" 인코더, "NTSC(national television system(s) committee)" 인코더 또는 임의의 다른 비디오 인코더가 디지털 신호 프로세서(110)에 연결됨을 나타낸다. 또한, 비디오 증폭기(136)가 비디오 인코더(134) 및 터치스크린 디스플레이(132)에 연결된다. 비디오 포트(138)가 비디오 증폭기(136)에 연결된다. 도 1에 도시된 바와 같이, 범용 직렬 버스("USB(universal serial bus)") 제어기(140)가 디지털 신호 프로세서(110)에 연결된다. 또한, USB 포트(142)가 USB 제어기(140)에 연결된다. 메모리(112) 및 가입자 식별 모듈(SIM: subscriber identity module) 카드(146)가 또한 디지털 신호 프로세서(110)에 연결될 수 있다. 또한, 도 1에 도시된 바와 같이, 디지털 카메라(148)가 디지털 신호 프로세서(110)에 연결될 수 있다. 예시적인 양상에서, 디지털 카메라(148)는 전하 결합 소자("CCD(charge-coupled device)") 카메라 또는 상보성 금속 산화물 반도체("CMOS(complementary metal-oxide semiconductor)") 카메라이다.

[0037] 도 1에 더 예시된 바와 같이, 스테레오 오디오 코덱(150)이 아날로그 신호 프로세서(126)에 연결될 수 있다. 더욱이, 오디오 증폭기(152)가 스테레오 오디오 코덱(150)에 연결될 수 있다. 예시적인 양상에서, 제 1 스테레오 스피커(154) 및 제 2 스테레오 스피커(156)가 오디오 증폭기(152)에 연결된다. 도 1은 마이크로폰 증폭기(158)가 또한 스테레오 오디오 코덱(150)에 연결될 수 있음을 보여준다. 추가로, 마이크로폰(160)이 마이크로폰 증폭기(158)에 연결될 수 있다. 특정 양상에서, 주파수 변조("FM(frequency modulation)") 라디오 튜너(162)가 스테레오 오디오 코덱(150)에 연결될 수 있다. 또한, FM 안테나(164)가 FM 라디오 튜너(162)에 연결된다. 또한, 스테레오 헤드폰들(166)이 스테레오 오디오 코덱(150)에 연결될 수 있다.

[0038] 도 1은 또한, 무선 주파수("RF(radio frequency)") 트랜시버(168)가 아날로그 신호 프로세서(126)에 연결될 수 있음을 나타낸다. RF 스위치(170)가 RF 트랜시버(168) 및 RF 안테나(172)에 연결될 수 있다. 도 1에 도시된

바와 같이, 키패드(174)가 아날로그 신호 프로세서(126)에 연결될 수 있다. 또한, 마이크로폰을 구비한 모노 헤드셋(176)이 아날로그 신호 프로세서(126)에 연결될 수 있다. 또한, 진동기 디바이스(178)가 아날로그 신호 프로세서(126)에 연결될 수 있다. 도 1은 또한 파워 서플라이(180), 예를 들어 배터리가 온-칩 시스템(102)에 연결됨을 보여준다. 특정 양상에서, 파워 서플라이는 교류("AC(alternating current)") 전원에 접속된 AC-DC 변환기로부터 유도되는 재충전 가능한 DC 배터리 또는 DC 파워 서플라이를 포함한다.

[0039] 도 1은 또한, 모바일 디바이스(100)가 모니터 모듈(114)을 포함할 수 있음을 보여준다. 모니터 모듈(114)은 온-칩 시스템(102) 전역에 분산된 다수의 가동 센서들과 그리고 모바일 디바이스(100)의 디지털 신호 프로세서(110)와 통신한다. 도 1에 도시된 바와 같이, 터치스크린 디스플레이(132), 비디오 포트(138), USB 포트(142), 카메라(148), 제 1 스테레오 스피커(154), 제 2 스테레오 스피커(156), 마이크로폰(160), FM 안테나(164), 스테레오 헤드폰들(166), RF 스위치(170), RF 안테나(172), 키패드(174), 모노 헤드셋(176), 진동기(178) 및 파워 서플라이(180)는 온-칩 시스템(102) 외부에 있다. 그러나 모니터 모듈(114)은 또한 모바일 디바이스(100) 상에서 동작 가능한 자원들의 실시간 관리에 도움이 되도록 아날로그 신호 프로세서(126) 및 디지털 신호 프로세서(110)에 의해 이러한 외부 디바이스들 중 하나 또는 그보다 많은 디바이스로부터 하나 또는 그보다 많은 표시들 또는 신호들을 수신할 수 있다고 이해되어야 한다.

[0040] 특정 양상에서, 본 명세서에서 설명되는 방법 단계들 중 하나 또는 그보다 많은 단계는 메모리(112)에 저장된 실행 가능 명령들 및 파라미터들에 의해 구현될 수 있다. 이러한 명령들은 본 명세서에서 설명되는 방법들을 수행하도록 디지털 신호 프로세서(110), 아날로그 신호 프로세서(126) 또는 다른 프로세서에 의해 실행될 수 있다. 또한, 프로세서들(110, 126), 메모리(112), 그 내부에 저장된 명령들, 또는 이들의 조합은 본 명세서에서 설명되는 방법 단계들 중 하나 또는 그보다 많은 단계를 수행하기 위한 수단의 역할을 할 수 있다.

[0041] 도 2는 도 1의 모바일 디바이스(100)의 특징들을 나타내는 개략도이다. 도 2에 지시된 바와 같이, 디지털 신호 프로세서(110)는 버스(211)를 통해 메모리(112)에 연결된다. 디지털 신호 프로세서(110)는 N개의 코어 프로세서들을 갖는 다중 코어 프로세서이다. 즉, 디지털 신호 프로세서(110)는 제 1 코어(222), 제 2 코어(224) 및 제 N 코어(230)를 포함한다. 알려진 바와 같이, 제 1 코어(222), 제 2 코어(224) 및 제 N 코어(230) 각각은 전용 애플리케이션 또는 프로그램을 지원하기 위해 이용 가능하다. 대안으로, 이용 가능한 코어들 중 2개 또는 그보다 많은 코어에 걸친 처리를 위해 하나 또는 그보다 많은 애플리케이션들 또는 프로그램들이 분산될 수 있다.

[0042] 제 1 코어(222), 제 2 코어(224) 내지 제 N 코어(230)는 단일 집적 회로 다이 상에 집적될 수 있거나, 아니면 이들은 다중 회로 패키지로 개별 다이들 상에 집적 또는 연결될 수도 있다. 설계자들은 하나 또는 그보다 많은 공유 캐시들을 통해 제 1 코어(222), 제 2 코어(224) 내지 제 N 코어(230)를 연결할 수 있으며 그들은 버스, 링, 그물망(mesh) 및 크로스바 토폴로지들과 같은 네트워크 토폴로지들을 통해 메시지 또는 명령 전달을 구현할 수 있다.

[0043] 예시된 실시예에서, RF 트랜시버(268)는 디지털 회로 엘리먼트들을 통해 구현되며 ("코어"로 표기된) 코어 프로세서(210)와 같은 적어도 하나의 프로세서를 포함한다. 이러한 디지털 구현에서, RF 트랜시버(268)는 버스(213)를 통해 메모리(112)에 연결된다.

[0044] 버스(211)와 버스(213) 각각은 해당 기술분야에 공지된 바와 같이, 하나 또는 그보다 많은 유선 또는 무선 접속들을 통하는 다수의 통신 경로들을 포함할 수 있다. 버스(211) 및 버스(213)는 통신들을 가능하게 하기 위한, 제어기들, 버퍼들(캐시들), 드라이버들, 리피터들 및 수신기들과 같은, 단순화를 위해 생략된 추가 엘리먼트들을 가질 수 있다. 또한, 버스(211) 및 버스(213)는 앞서 언급한 컴포넌트들 사이의 적절한 통신들을 가능하게 하기 위한 어드레스, 제어 및/또는 데이터 접속들을 포함할 수 있다.

[0045] 모바일 디바이스(100)에 의해 사용되는 로직이 도 2에 도시된 것과 같이 소프트웨어로 구현될 경우, 시동 로직(250), 관리 로직(260), DVFS 인터페이스 로직(270), 애플리케이션 저장소(280) 내의 애플리케이션들 및 파일 시스템(290)의 부분들 중 하나 또는 그보다 많은 것이 임의의 컴퓨터 관련 시스템 또는 방법과 관련하여 또는 이들에 의한 사용을 위해 임의의 컴퓨터 판독 가능 매체 상에 저장될 수 있다는 점에 주목해야 한다.

[0046] 본 문서의 맥락에서, 컴퓨터 판독 가능 매체는 컴퓨터 관련 시스템 또는 방법과 관련하여 또는 이들에 의한 사용을 위해 컴퓨터 프로그램 및 데이터를 포함하거나 저장할 수 있는 전자, 자기, 광 또는 다른 물리적 디바이스 또는 수단이다. 다양한 로직 엘리먼트들과 데이터 저장소들이 컴퓨터 기반 시스템, 프로세서 포함 시스템, 또는 명령 실행 시스템, 장치 또는 디바이스로부터 명령들을 페치(fetch)하여 명령들을 실행할 수 있는 다른 시스

템과 같은 명령 실행 시스템, 장치 또는 디바이스와 관련하여 또는 이들에 의한 사용을 위해 임의의 컴퓨터 판독 가능 매체에 구현될 수 있다. 본 문서의 맥락에서, "컴퓨터 판독 가능 매체"는 명령 실행 시스템, 장치 또는 디바이스와 관련하여 또는 이들에 의한 사용을 위해 프로그램을 저장, 전달, 전파 또는 전송할 수 있는 임의의 수단일 수 있다.

[0047] 컴퓨터 판독 가능 매체는 예를 들어, 전자, 자기, 광, 전자기, 적외선 또는 반도체 시스템, 장치, 디바이스 또는 전파 매체일 수 있지만, 이에 한정되는 것은 아니다. 컴퓨터 판독 가능 매체의 더 특정한 예들(완전하진 않은 리스트)은 하나 또는 그보다 많은 와이어들을 갖는 전기 접속(전자), 휴대용 컴퓨터 디스켓(자기), 랜덤 액세스 메모리(RAM: random-access memory)(전자), 판독 전용 메모리(ROM: read-only memory)(전자), 소거 가능한 프로그래밍 가능 판독 전용 메모리(EPROM, EEPROM 또는 플래시 메모리)(전자), 광섬유(광) 및 휴대용 콤팩트 디스크 판독 전용 메모리(CDROM: compact disc read-only memory)(광)를 포함할 것이다. 프로그램은 예컨대 종이나 다른 매체의 광학적 스캐닝을 통해 전자적으로 캡처된 다음, 필요하다면 적당한 방식으로 컴파일되거나, 해석 또는 이와 달리 처리되고, 다음에 컴퓨터 메모리에 저장될 수 있으므로, 컴퓨터 판독 가능 매체는 심지어 프로그램이 인쇄되는 종이나 다른 적당한 매체일 수도 있다는 점에 주목한다.

[0048] 시동 로직(250), 관리 로직(260) 및 가능하다면 DVFS 인터페이스 로직(270) 중 하나 또는 그보다 많은 로직이 하드웨어로 구현되는 대안적인 실시예에서, 다양한 로직은 다음의 기술들: 데이터 신호들에 대한 로직 기능들을 구현하기 위한 로직 게이트들을 갖는 이산 로직 회로(들), 적절한 조합 로직 게이트들을 갖는 주문형 집적 회로(ASIC), 프로그래밍 가능 게이트 어레이(들)(PGA), 필드 프로그래밍 가능 게이트 어레이(FPGA) 등 중에서 임의의 기술 또는 이들의 조합으로 구현될 수 있으며, 이러한 기술들은 각각 해당 기술분야에 잘 알려져 있다.

[0049] 메모리(112)는 플래시 메모리나 교체 상태 메모리 디바이스와 같은 비휘발성 데이터 저장 디바이스이다. 단일 디바이스로서 도시된다 하더라도, 메모리(112)는 디지털 신호 프로세서 및/또는 RF 트랜시버(268) 내의 코어(210)(또는 추가 프로세서 코어들)에 연결된 개별 데이터 저장소들을 갖는 분산형 메모리 디바이스일 수도 있다.

[0050] 시동 로직(250)은 제 1 코어(222), 제 2 코어(224) 내지 제 N 코어(230)와 같은 이용 가능한 코어들 중 하나 또는 그보다 많은 코어의 성능을 관리 또는 제어하기 위한 선택 프로그램을 선택적으로 식별하여 로딩하고 실행하기 위한 하나 또는 그보다 많은 실행 가능 명령들을 포함한다. 선택 프로그램은 내장된 파일 시스템(290)의 프로그램 저장소(296)에서 발견될 수 있으며 성능 스케일링 알고리즘(297)과 한 세트의 파라미터들(298)의 특정 조합으로 정의된다. 선택 프로그램은 RF 트랜시버(268) 내의 코어(210)와 디지털 신호 프로세서 내의 코어 프로세서들 중 하나 또는 그보다 많은 코어에 의해 실행될 때, 모니터 모듈(114)에 의해 제공되는 하나 또는 그보다 많은 신호들에 따라 각각의 프로세서 코어의 성능을 스케일링하도록 동작할 수 있다. 이와 관련하여, 모니터 모듈(114)은 이벤트들, 프로세스들, 애플리케이션들, 자원 상태 조건들, 경과 시간, 온도 등의 하나 또는 그보다 많은 표시자들을 제공할 수 있다.

[0051] 관리 로직(260)은 각각의 프로세서 코어들 중 하나 또는 그보다 많은 코어에 대한 동작 성능 스케일링 프로그램을 종료할 뿐만 아니라, 이용 가능한 코어들 중 하나 또는 그보다 많은 코어의 성능을 관리 또는 제어하기 위한 더 적당한 대체 프로그램을 선택적으로 식별하여 로딩하고 실행하기 위한 하나 또는 그보다 많은 실행 가능 명령들을 포함한다. 관리 로직(260)은 런타임 시 또는 모바일 디바이스에 전원이 인가되어 디바이스의 운영자에 의해 사용되고 있는 동안 이러한 기능들을 수행하도록 배치된다. 대체 프로그램은 내장된 파일 시스템(290)의 프로그램 저장소(296)에서 발견될 수 있으며 성능 스케일링 알고리즘(297)과 한 세트의 파라미터들(298)의 특정 조합으로 정의된다. 대체 프로그램은 RF 트랜시버(268) 내의 코어(210)나 디지털 신호 프로세서 내의 코어 프로세서들 중 하나 또는 그보다 많은 코어에 의해 실행될 때, 모니터 모듈(114)에 의해 제공되는 하나 또는 그보다 많은 신호들 또는 다양한 프로세서 코어들의 각각의 제어 입력들 상에 제공되는 하나 또는 그보다 많은 신호들에 따라 각각의 프로세서 코어의 성능을 스케일링하도록 동작할 수 있다. 이와 관련하여, 모니터 모듈(114)은 이벤트들, 프로세스들, 애플리케이션들, 자원 상태 조건들, 경과 시간, 온도 등의 하나 또는 그보다 많은 표시자들을 제공할 수 있다.

[0052] DVFS 인터페이스 로직 또는 인터페이스 로직(270)은 내장된 파일 시스템(290)에 저장된 정보를 관측, 구성하거나 아니면 업데이트하도록 외부 입력들을 제시, 관리하고 이들과 상호 작용하기 위한 하나 또는 그보다 많은 실행 가능 명령들을 포함한다. 일 실시예에서, 인터페이스 로직(270)은 USB 포트(142)를 통해 수신되는 제조업체 입력들과 함께 동작할 수 있다. 이러한 입력들은 프로그램 저장소(296)로부터 삭제되거나 프로그램 저장소(296)에 추가될 하나 또는 그보다 많은 프로그램들을 포함할 수 있다. 대안으로, 입력들은 프로그램 저장소

(296) 내의 프로그램들 중 하나 또는 그보다 많은 프로그램에 대한 편집들이나 변경들을 포함할 수 있다. 더욱이, 입력들은 시동 로직(250)과 관리 로직(260) 중 하나 또는 둘 다에 대한 하나 또는 그보다 많은 변경들 또는 이들의 전체 대체들을 식별할 수 있다. 예로서, 입력들은 수신 신호 전력이 식별된 임계치 아래로 떨어질 때 RF 트랜시버(268)에서 모든 성능 스케일링을 보류하도록 모바일 디바이스(100)에 명령하는 관리 로직(260)에 대한 변경을 포함할 수 있다. 추가 예로서, 입력들은 비디오 코덱(134)이 활동중일 때 원하는 프로그램을 적용하도록 모바일 디바이스(100)에 명령하는 관리 로직(260)에 대한 변경을 포함할 수 있다.

[0053] 인터페이스 로직(270)은 제조업체가 모바일 디바이스(100)에 대해 정의된 동작 조건들 하에서 최종 사용자의 체험을 제어 가능하게 구성 및 조정하게 할 수 있다. 메모리(112)가 플래시 메모리인 경우, 시동 로직(250), 관리 로직(260), 인터페이스 로직(270), 애플리케이션 저장소(280) 내의 애플리케이션 프로그램들 또는 내장된 파일 시스템(290) 내의 정보 중 하나 또는 그보다 많은 것이 편집, 대체되거나 아니면 수정될 수 있다. 일부 실시예들에서, 인터페이스 로직(270)은 모바일 디바이스(100)의 최종 사용자 또는 운영자가 시동 로직(250), 관리 로직(260), 애플리케이션 저장소(280) 내의 애플리케이션들 및 내장된 파일 시스템(290) 내의 정보를 탐색, 위치 결정, 수정 또는 대체하는 것을 허용할 수 있다. 운영자는 결과적인 인터페이스를 사용하여 모바일 디바이스(100)의 다음 시동시 구현될 변경들을 수행할 수 있다. 대안으로, 운영자는 결과적인 인터페이스를 사용하여 런타임 동안 구현되는 변경들을 수행할 수도 있다.

[0054] 내장된 파일 시스템(290)은 계층적으로 배열된 DVFS 저장소(292)를 포함한다. 이와 관련하여, 파일 시스템(290)은 모바일 디바이스(100)에 의해 사용되는 다양한 파라미터들(298) 및 성능 스케일링 알고리즘들(297)의 구성 및 관리를 위한 정보의 저장을 위해 자신의 전체 파일 시스템 용량의 예비 섹션을 포함할 수 있다. 도 2에 도시된 바와 같이, DVFS 저장소(292)는 프로그램 저장소(296)를 포함하는 코어 저장소(294)를 포함하며, 프로그램 저장소(296)는 하나 또는 그보다 많은 DVFS 프로그램들을 포함한다. 각각의 프로그램은 각각의 성능 스케일링 알고리즘 그리고 특정 알고리즘과 연관된 한 세트의 파라미터들의 조합으로서 정의된다. DVFS 저장소(292)의 계층적 특성의 추가 예로서, 한 세트의 파일들 중 특정 멤버는 `\startup\core0\algorithm\parameterset`의 경로로 위치 결정 및 식별될 수 있다. 이 예에서, 프로그램은 파라미터 세트에 저장된 정보의 콘텐츠와 결합하여 알고리즘에 의해 식별된다. 예를 들어, "클래식(classic)"으로서 알려진 종래의 DVFS 알고리즘은, 샘플 레이트가 Mhz 단위의 `\startup\core0\classic\samplerate` 콘텐츠(100); 증가시킬 샘플들이 정수인 `\startup\core0\classic\up` 콘텐츠(2); 그리고 감소시킬 샘플들이 정수인 `\startup\core0\classic\down` 콘텐츠(1)와 같이, 샘플 레이트, ("up"으로서 기재된) 증가시킬 샘플들 및 ("down"으로서 기재된) 감소시킬 샘플들과 같은 파라미터들에 따라 코어(222)에 대한 성능 스케일링을 관리하도록 식별될 수 있다. 즉, 각각의 파일명들이 파라미터를 정의하고 파라미터의 값은 파일의 콘텐츠에 의해 식별된다. 알고리즘은 CPU 유휴 비율의 주기적 샘플링에 의해 정의되며 낮은 임계치(유휴%) 및 높은 임계치(유휴%)에 따라 동작한다. 증가시킬 샘플들 임계치 비교기가 2개의 연이은 샘플들에 대해 성능이 증가되어야 한다고 지시한다면, DVFS 알고리즘은 미리 결정된 클럭 레벨 조정치에 따라 성능을 증가시킨다. 반대로, 감소시킬 샘플들 임계치 비교기가 1개의 연이은 샘플에 대해 성능이 감소되어야 한다고 지시한다면, DVFS 알고리즘은 미리 결정된 클럭 레벨(즉, 주파수) 조정치에 따라 성능을 감소시킨다. 위에서 설명한 바와 같이, 클럭 주파수의 변경들과 함께 프로세서 또는 코어 동작 전압이 변경될 수 있다.

[0055] 대안으로 또는 추가로, DVFS 저장소(292)는 운영 시스템이 맞닥뜨리게 되는 첫 번째 정보가 가장 상세한 것이 되는 순서로 그 안에 저장된 정보와 맞닥뜨리도록 배열될 수 있으며, 여기서 맞닥뜨리게 되는 마지막 정보는 모바일 디바이스(100) 상의 각각의 프로세서 코어의 성능을 스케일링하는데 적합한 대로 탐색하는 동안에는 어떠한 다른 프로그램도 식별되지 않아야 하는 디폴트로서 적용될 것이다. 예를 들어, `/core0`, `/coreall` 및 `/default` 디렉터리들에서 파라미터들이 정의될 수 있다. 각각의 파라미터에 대해 그 순서로 경로들이 탐색될 것이다. 발견된 첫 번째 파라미터가 사용되는 첫 번째 파라미터가 될 것이다. `\default` 위치는 항상 유효 파라미터 파일을 가질 것이다. 개별 코어들의 아키텍처, 하나 또는 그보다 많은 공유 캐시들의 아키텍처 및 코어들 사이로 명령들을 전달하는데 사용되는 메커니즘(들)뿐만 아니라, 모바일 디바이스(100)에 대해 원하는 사용 사례들 또한 메모리(112)에 저장된 다양한 성능 스케일링 알고리즘들(297)의 특성을 지시할 것으로 예상된다.

[0056] 도 3은 도 1의 모바일 디바이스(100) 상에서 동작하는 성능 스케일링 알고리즘들을 관리 및 노출하기 위한 방법(300)의 실시예를 나타내는 흐름도이다. 모바일 디바이스(100) 내의 메모리(112)와 같은 비휘발성 메모리에 한 세트의 성능 스케일링 알고리즘들이 제공되는 블록(302)에서 방법(300)이 시작된다. 블록(304)에서, 한 세트의 성능 스케일링 알고리즘들의 각각의 멤버에 대해 한 세트의 파라미터들이 식별된다. 이후, 블록(306)에 지시된 바와 같이, 한 세트의 프로그램들을 정의하기 위해 한 세트의 파라미터들이 한 세트의 성능 스케일링 알고리즘

들의 각각의 멤버들과 연관된다. 위에 지시된 바와 같이, 이는 특정 성능 스케일링 알고리즘에 적용될 수 있는 다양한 파라미터들을 조직화하기 위한 정해진 시퀀스를 설정함으로써 파일 시스템(290)과 같은 파일 시스템에서 이루어질 수 있다.

[0057] 다음에, 블록(308)에 도시된 바와 같이, 식별된 프로세서 코어의 성능을 스케일링하기 위해 한 세트의 프로그램들 중 적어도 하나의 멤버가 식별된다. 블록(310)에서는, 한 세트의 프로그램들 중 블록(308)에서 식별된 멤버와 식별된 프로세서 코어가 서로 연관된다. 위에 지시된 바와 같이, 이는 알고리즘들과 파라미터들을 명시된 코어 식별자 하에서 계층적으로 저장하거나 아니면 배열함으로써 파일 시스템(290)과 같은 파일 시스템에서 이루어질 수 있다. 블록(312)에서, 한 세트의 프로그램들 및 코어 연관들을 노출하는 인터페이스가 생성된다. 선택적인 블록(314)에서는, 시동시 이용 가능한 프로세서 코어들에 대한 성능 스케일링을 구현하기 위한 실행을 위해 제 1 프로그램이 선택된다. 이는 각각의 프로세서 코어의 동작을 관리하기 위한 적절한 DVFS 프로그램(즉, 성능 스케일링 알고리즘과 한 세트의 파라미터들의 조합)을 식별하기 위한 시도로 DVFS 저장소(292)를 가로지르는(traverse) 검색 루틴에 의해 이루어질 수 있다. 이후, 선택적인 블록(316)에 도시된 바와 같이, 제 1 프로그램은 정해진 이벤트, 동작 조건, 특정 하드웨어 자원들에 대한 요청, 애플리케이션(들) 등 중 하나 또는 그보다 많은 것의 식별시 런타임 동안 제 2 프로그램으로 대체될 수 있다.

[0058] 도 4는 도 1의 모바일 디바이스 상에서 동작 가능한 컴퓨터 프로그램 물건의 실시예를 나타내는 흐름도이다. 일부 실시예들에서 컴퓨터 프로그램 물건은 시동 로직(250) 및 관리 로직(260)과 연관하여 위에서 설명한 기능을 포괄할 수 있다. 대안으로, 컴퓨터 프로그램 물건은 특정 모바일 디바이스(100)에 대한 DVFS 저장소(292)를 덧붙이기(populate)에 적합한 정보 및 인터페이스 로직(270)의 양상들을 더 포함할 수 있다.

[0059] 도 4에 예시된 바와 같이, 컴퓨터 프로그램 물건의 예시적인 기능은 흐름도 또는 방법(400)의 형태로 예시된다. 프로그램을 정의하기 위해 모바일 디바이스(100) 상에서 동작 가능한 프로세서 코어의 성능을 스케일링하기에 적합한 한 세트의 프로그램들 중의 한 멤버가 선택되는 블록(402)에서 방법(400)이 시작된다. 블록(402)에 추가로 예시된 바와 같이, 성능 스케일링 알고리즘과 그 알고리즘에 적용되는 파라미터들의 각각의 세트의 각각의 조합으로 한 세트의 프로그램들이 정의된다. 블록(404)에서는, 선택 프로그램이 모바일 디바이스(100) 상의 프로세서 코어에 로딩되거나 아니면 전달된다. 블록(404)에 지시된 바와 같이, 이는 시동 시퀀스에 따라 이루어진다. 해당 기술분야에서 통상의 지식을 가진 자는 시동 시퀀스가 펌웨어, 소프트웨어 또는 심지어 운영 시스템 자체에서 인에이블될 수 있다고 이해할 것이다. 일단 선택 프로그램이 각각의 프로세서 코어에 전달되면, 요구될 수 있는 대로 선택 프로그램이 실행되어 프로세서 코어 또는 모바일 디바이스(100) 상에서 동작하는 다른 프로세서 코어들의 성능을 스케일링할 수 있다.

[0060] 이후, 블록(406)에 지시된 바와 같이, 한 세트의 프로그램들 중의 대체 프로그램이 선택 프로그램을 대체해야 함을 나타내는 상태를 식별하기 위해 모바일 디바이스(100)의 사용이 모니터링된다. 블록(408)에서, 런타임 동안 선택 프로그램이 대체 프로그램으로 대체된다. 위에 지시된 바와 같이, 선택 프로그램을 대체 프로그램으로 대체하는 것은 대체 프로그램이 로딩되어 실행되기 이전의, 선택 프로그램의 세심히 관리되는 종료가 동반될 수 있다.

[0061] 본 발명이 설명된 바와 같이 기능하도록 하기 위해, 본 명세서에서 설명된 프로세스들 또는 프로세스 흐름들의 특정 단계들이 당연히 다른 것들에 우선한다. 그러나 설명된 단계들의 순서나 시퀀스가 본 발명의 기능을 변경하지 않는다면, 본 발명은 이러한 순서로 한정되지는 않는다. 즉, 일부 단계들은 본 발명의 범위 및 사상을 벗어나지 않으면서 다른 단계들 이전, 이후 또는 이들과 동시에(실질적으로 동시에) 수행될 수 있다고 인식된다. 어떤 경우들에, 본 발명을 벗어나지 않으면서 특정 단계들이 생략되거나 수행되지 않을 수도 있다. 또한, "이후," "그 다음," "다음에" 등과 같은 단어들은 단계들의 순서를 한정하는 것으로 의도되는 것은 아니다. 이러한 단어들은 단순히 예시적인 발명의 설명을 통해 독자를 지도하는데 사용된다.

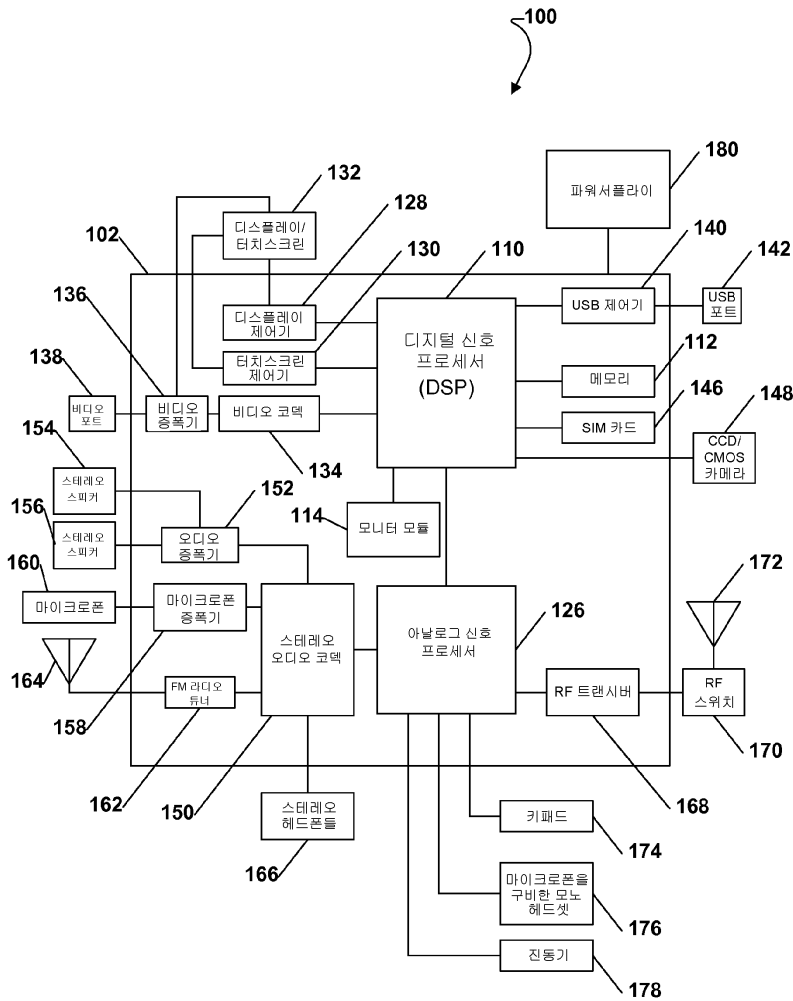
[0062] 프로그래밍 분야에서 통상의 지식을 가진 자는 개시된 프로세스들을 과도한 어려움이나 실험 없이 구현하도록 컴퓨터 코드를 작성하거나 적절한 하드웨어를 식별할 수 있다. 따라서 본 발명을 어떻게 실시 및 사용하는지에 관한 충분한 이해를 위해 특정 세트의 실행 가능 명령들 또는 상세한 하드웨어 디바이스들의 개시가 필수적으로 고려되는 것은 아니다. 청구되는 모바일 디바이스(100) 및 프로세스들에 대한 발명의 기능은 상기 설명에서 그리고 다양한 프로세스 흐름들을 나타낼 수 있는 도면들과 함께 더 상세히 설명된다.

[0063] 추가로, 프로그래밍에서 통상의 지식을 가진 자는 개시된 발명을 예를 들어, 본 명세서의 흐름도들 및 연관된 설명을 기초로 어려움 없이 구현하도록 컴퓨터 코드를 작성하거나 적절한 하드웨어 및/또는 회로들을 식별할 수 있다.

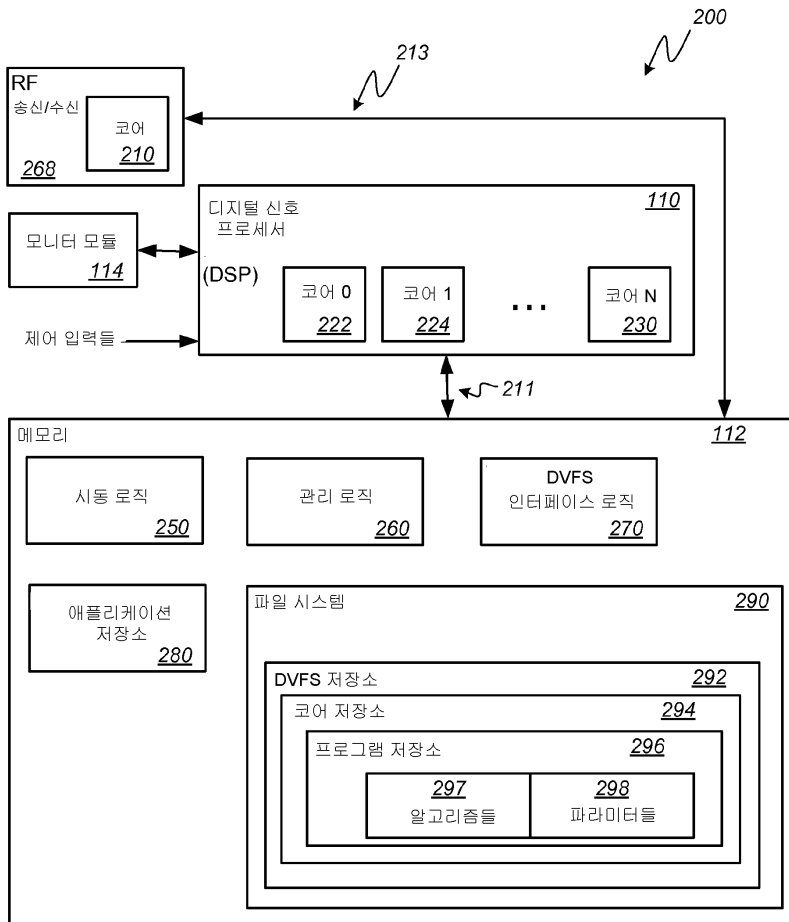
- [0064] 따라서 본 발명을 어떻게 실시 및 사용하는지에 관한 충분한 이해를 위해 특정 세트의 프로그램 코드 명령들 또는 상세한 하드웨어 디바이스들의 개시가 필수적으로 고려되는 것은 아니다. 청구되는 컴퓨터 구현 프로세스들에 대한 발명의 기능은 상기 설명에서 그리고 다양한 프로세스 흐름들을 나타낼 수 있는 도면들과 함께 더 상세히 설명된다.
- [0065] 하나 또는 그보다 많은 예시적인 양상들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수 있다. 소프트웨어로 구현된다면, 기능들은 컴퓨터 판독 가능 매체 상에 하나 또는 그보다 많은 명령들 또는 코드로서 저장되거나 이를 통해 전송될 수 있다. 컴퓨터 판독 가능 매체는 한 장소에서 다른 장소로 컴퓨터 프로그램의 전달을 용이하게 하는 임의의 매체를 포함하는 통신 매체와 컴퓨터 저장 매체를 모두 포함한다. 저장 매체는 컴퓨터에 의해 액세스 가능한 임의의 이용 가능한 매체일 수 있다. 한정이 아닌 예시로, 이러한 컴퓨터 판독 가능 매체는 RAM, ROM, EEPROM, CD-ROM이나 다른 광 디스크 저장소, 자기 디스크 저장소 또는 다른 자기 저장 디바이스들, 또는 명령들이나 데이터 구조들의 형태로 원하는 프로그램 코드를 전달하거나 저장하는데 사용될 수 있으며 컴퓨터에 의해 액세스 가능한 임의의 다른 매체를 포함할 수 있다.
- [0066] 또한, 임의의 접속이 컴퓨터 판독 가능 매체로 적절히 지칭된다. 예를 들어, 소프트웨어가 동축 케이블, 광섬유 케이블, 꼬임 쌍선, 디지털 가입자 회선("DSL(digital subscriber line)"), 또는 적외선, 라디오 및 마이크로파와 같은 무선 기술들을 이용하여 웹사이트, 서버 또는 다른 원격 소스로부터 전송된다면, 동축 케이블, 광섬유 케이블, 꼬임 쌍선, DSL, 또는 적외선, 라디오 및 마이크로파와 같은 무선 기술들이 매체의 정의에 포함된다.
- [0067] 본 명세서에서 사용된 것과 같은 디스크(disk 및 disc)는 콤팩트 디스크("CD(compact disc)"), 레이저 디스크(laser disc), 광 디스크(optical disc), 디지털 다기능 디스크("DVD(digital versatile disc)"), 플로피 디스크(floppy disk) 및 블루레이 디스크(blue-ray disc)를 포함하며, 여기서 디스크(disk)들은 통상적으로 데이터를 자기적으로 재생하는 한편, 디스크(disc)들은 데이터를 레이저들에 의해 광학적으로 재생한다. 상기의 조합들 또한 컴퓨터 판독 가능 매체의 범위 내에 포함되어야 한다.
- [0068] 선택된 양상들이 예시되고 상세히 설명되었지만, 다음의 청구항들에 의해 정의된 바와 같이, 본 발명의 범위를 벗어나지 않으면서 본 발명에 다양한 치환들 및 변형들이 이루어질 수 있다고 이해될 것이다.

도면

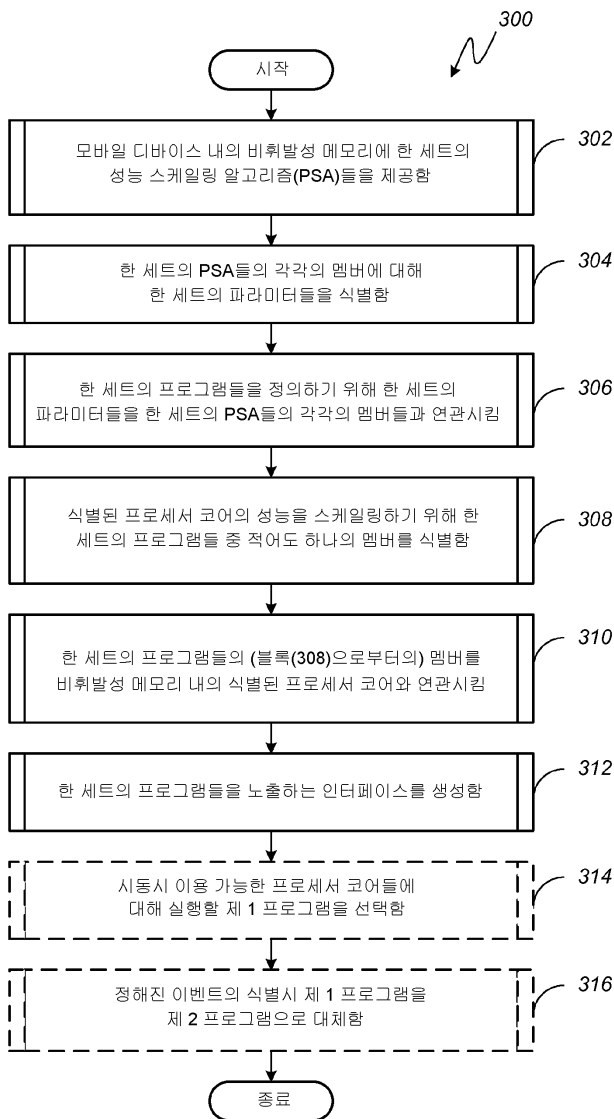
도면1



도면2



도면3



도면4

