

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6070321号  
(P6070321)

(45) 発行日 平成29年2月1日(2017.2.1)

(24) 登録日 平成29年1月13日(2017.1.13)

(51) Int.Cl.		F I			
<b>G06F</b>	<b>1/04</b>	<b>(2006.01)</b>	G06F	1/04	575
<b>G06F</b>	<b>1/32</b>	<b>(2006.01)</b>	G06F	1/32	Z
<b>H04M</b>	<b>1/73</b>	<b>(2006.01)</b>	H04M	1/73	

請求項の数 7 (全 16 頁)

(21) 出願番号	特願2013-58359 (P2013-58359)	(73) 特許権者	000005223
(22) 出願日	平成25年3月21日 (2013.3.21)		富士通株式会社
(65) 公開番号	特開2014-182736 (P2014-182736A)		神奈川県川崎市中原区上小田中4丁目1番 1号
(43) 公開日	平成26年9月29日 (2014.9.29)	(74) 代理人	100105142
審査請求日	平成27年11月6日 (2015.11.6)		弁理士 下田 憲次
		(72) 発明者	野呂 正明
			神奈川県川崎市中原区上小田中4丁目1番 1号 富士通株式会社内
		審査官	塩澤 如正

最終頁に続く

(54) 【発明の名称】 携帯情報端末、制御方法、制御プログラム

(57) 【特許請求の範囲】

【請求項1】

CPUの動作周波数を制御する携帯情報端末に於いて、  
前記携帯情報端末の状態を示す状態情報を取得する状態情報取得部と、  
前記状態情報に基づき前記状態の変更を検知し、前記状態の変更が検知されてから所定の時間の経過前に発生するイベントを処理する際、前記CPUの動作周波数の下限値を上昇させる制御部と、  
を備える携帯情報端末。

【請求項2】

請求項1に記載の携帯情報端末に於いて、  
前記制御部は、前記所定の時間の経過後に発生するイベントを処理する際、前記CPUの動作周波数の下限値を、前記状態に対応する値とする、携帯情報端末。

【請求項3】

請求項1又は2に記載の携帯情報端末に於いて、  
前記状態の変更は、前記携帯情報端末のスリープモードからの復帰もしくは前記携帯情報端末の充電に付随するものである、携帯情報端末。

【請求項4】

CPUの動作周波数を制御する携帯情報端末の制御方法に於いて、  
前記携帯情報端末の状態を示す状態情報を取得し、  
前記状態情報に基づき前記状態の変更を検知し、

10

20

前記状態の変更が検知されてから所定の時間の経過前に発生するイベントを処理する際、前記CPUの動作周波数の下限値を上昇させる、制御方法。

【請求項5】

請求項4に記載の制御方法に於いて、

前記CPUは、前記携帯情報端末に搭載され、前記イベントを処理するCPUである、制御方法。

【請求項6】

CPUの動作周波数を制御する携帯情報端末の制御プログラムに於いて、

前記制御プログラムは、前記携帯情報端末に、

前記携帯情報端末の状態を示す状態情報を取得させ、

前記状態情報に基づき前記状態の変更を検知させ、

前記状態の変更が検知されてから所定の時間の経過前に発生するイベントを処理する際、前記CPUの動作周波数の下限値を上昇させる、制御プログラム。

【請求項7】

請求項1乃至3何れかに記載の携帯情報端末に於いて、

前記動作周波数の制御に関する複数の値を有するパラメータを格納するパラメータ格納部を更に有し、

前記制御部は、前記パラメータ格納部に格納された前記複数の値を有するパラメータのうち、何れかのパラメータを選択し、選択された前記パラメータに基づいて前記CPUの前記動作周波数を制御することにより、前記CPUの前記動作周波数の前記下限値を上昇させる

ことを特徴とする携帯情報端末。

【発明の詳細な説明】

【技術分野】

【0001】

本技術は、携帯情報端末、制御方法、制御プログラムに関する。

【背景技術】

【0002】

例えばスマートフォンなどの携帯情報端末は、通常のデスクトップPC(Personal Computer)用OSと同様、ユーザが多種多様なアプリ(アプリケーション)を自由にダウンロード及びインストールすることができる為、それぞれのアプリが要求するCPUの動作周波数を事前に把握することは困難である。同様に、アプリ開発者も、多種多様な携帯情報端末で、どの程度のCPU性能が必要であるかを把握することは困難である為、アプリ毎に必要なCPUの動作周波数に基づき、携帯情報端末のCPUの動作周波数を決定することはできない。

【0003】

デスクトップ用マルチタスクOSでのCPU動作周波数制御を説明する。

【0004】

先ず、ユーザもしくはミドルウェアに於いて、CPU動作周波数ポリシ(CPUの動作周波数を変更するアルゴリズム、動作周波数の上限値及び下限値、アルゴリズムや上限値及び下限値の継続時間)を指定して、OSカーネルに動作周波数の制御を指示する。OSカーネルは、CPU利用率と動作周波数ポリシに基づき、CPUの動作周波数の目標値を決定して、CPUの制御回路ドライバに動作周波数の制御を実施させる。

【0005】

ところが、従来技術にかかるCPU動作周波数制御は、CPUの利用率の観測後に実施される為、なんらかの負荷が発生してからCPUの動作周波数が上昇するまでに時間がかかる。このため、瞬間的に高い処理能力が必要となった場合、ユーザの体感性能が劣化することがある。

【0006】

10

20

30

40

50

さらに、CPU動作周波数制御モジュールに於いて、CPUの動作周波数の上昇が遅れると、以下のような問題も生じる。例えば携帯情報端末のスリープモードを解除する場合、先ず、OSカーネルが所定時間スリープを禁止するタイマを設定して、続いて該タイマが切れるまでに、ミドルウェアやアプリがスリープ禁止の設定を実行する。ところが、スリープモードの解除では、大量の処理が発生する。もし、CPU動作周波数制御モジュールによる、CPUの動作周波数の上昇制御が遅れると、発生した大量の処理をCPUが処理しきれずに、スリープモードの解除前にOSカーネルのスリープ禁止タイマが切れることがある。即ち、携帯情報端末をスリープモードから復帰させる為に電源ボタンを押下したにも関わらず、スリープモードの解除前に再度スリープモードに遷移することがある。スリープモードに遷移しないまでも、電源ボタンを押下してから実際に携帯情報端末が操作可能になるまでに膨大な時間がかかり、ユーザの操作性を損なうこともある。

10

【0007】

このような問題を解決する為、ブースト(以下、boostとする)という仕組みが存在する。boostは、CPU動作周波数制御モジュールの制御パラメータである動作周波数の下限値をCPU動作周波数制御モジュールの外部から変更して、CPUの動作周波数を瞬時に新たな下限値以上の動作周波数に変更する仕組みである。boostを使用すれば、瞬時に高い処理能力が必要なイベントが発生した際に、CPUの動作周波数の下限値が一時的に急上昇するので、イベントを確実に処理することができる。

【先行技術文献】

【特許文献】

20

【0008】

【特許文献1】特開2010-39791号公報

【発明の概要】

【発明が解決しようとする課題】

【0009】

ところで、従来技術にかかるboostは、ユーザが携帯情報端末の画面をOFFからONに切り替えるときなど、もっともCPUの処理能力が必要となる場面に用いられてきた。このため、CPU動作周波数制御モジュールに与える動作周波数の下限値やboostの持続時間は、最大の性能要求を満足できるように決められていた。それゆえ、高い処理能力を必要としないイベントが発生したときには、boostを動作させることで、無駄な消費電力が発生していた。

30

【0010】

開示の技術によれば、CPUの能力不足を避けつつ、消費電力を削減できる携帯情報端末、制御方法、制御プログラムを提供する。

【課題を解決するための手段】

【0011】

開示の技術の一観点によれば、CPUの動作周波数を制御する携帯情報端末に於いて、前記携帯情報端末の状態を取得する状態取得部と、前記状態の変更後かつ前記状態の取得に関する時間の経過前に発生するイベントの検知を契機に、前記CPUの動作周波数の下限値を上昇させる制御部と、を備える携帯情報端末、が提供される。

40

【発明の効果】

【0012】

本開示の技術の一観点によれば、CPUの能力不足を避けつつ、消費電力を削減することができる。

【図面の簡単な説明】

【0013】

【図1】第1の実施形態にかかる携帯情報端末のハードウェア構成の概略図である。

【図2】第1の実施形態にかかる携帯情報端末の機能ブロックの概略図である。

【図3】第1の実施形態にかかるboostモジュールの詳細な機能ブロックの概略図である。

50

【図4】第1の実施形態にかかる端末状態パラメータテーブルの概略図である。

【図5】第1の実施形態にかかるboostのフローチャートである。

【図6】第1の実施形態にかかるboostの具体例の説明図である。

【図7】第2の実施形態にかかる携帯情報端末の機能ブロックの概略図である。

【図8】第2の実施形態にかかるboostのフローチャートである。

【図9】第2の実施形態にかかるboostの具体例の説明図である。

【発明を実施するための形態】

【0014】

[第1の実施形態]

以下、図1～図6を参照して、第1の実施形態を説明する。

10

【0015】

本実施形態では、携帯情報端末100に発生するイベントの種別に基づき、boostの制御パラメータを変更することで、CPUの能力不足を回避しつつ消費電力の削減をはかる。

【0016】

例えば、画面OFFから画面ONへの切り替えや緊急地震速報のプッシュ通知などのイベントは、CPUの能力不足による処理遅延や処理中断が許されない為、このようなイベントが発生した場合には、事前に用意された、あらゆるイベントを確実に処理できる制御パラメータを使用して、boost(最優先クラスboost)を実行する。

20

【0017】

一方、例えばメールチェックなどの、CPUの能力不足が許されるイベントが発生した場合には、消費電力を削減する為に、携帯情報端末の端末状態に紐付けられた制御パラメータを使用して、boost(標準クラス用boost)を実行する。

【0018】

しかし、携帯情報端末に発生した特定のイベントに起因して、端末状態が変更した場合、携帯情報端末による端末状態の把握が遅延することがある。このため、端末状態の変更直後にCPUの能力不足が許される別イベントが発生した場合、変更後の端末状態に紐付けられた制御パラメータを使用して、boost(標準クラス用boost)を実行できないことがある。従って、特定のイベントに起因する端末状態の変更後の所定時間は、事前に決められた、通常のイベントを確実に処理できる制御パラメータを使用して、boost(中優先クラスboost)を実行する。

30

【0019】

[携帯情報端末100のハードウェア]

図1は、第1の実施形態にかかる携帯情報端末100のハードウェア構成の概略図である。本実施形態では、携帯情報端末100として、例えばスマートフォンやタブレットPCなどの携帯型の情報処理装置を想定する。

【0020】

図1に示すように、携帯情報端末100は、CPU(Central Processing Unit)101、メインメモリ102、補助メモリ103、クロック供給回路104、電圧供給回路105、電源回路106、バッテリー107、外部給電部108、ディスプレイ109、タッチスクリーン110、ネットワークI/F(Interface)111、センサ112、をハードウェアモジュールとして備える。これらのハードウェアモジュールは、バス113により相互接続されている。

40

【0021】

CPU101は、ベースバンドLSI(Large Scale Integrated)ではなく、アプリケーションプログラムを実行するプロセッサを想定している。CPU101は、クロック供給回路104から供給されるクロック信号と、電圧供給回路105から供給される電圧と、により動作して、携帯情報端末100の各種ハードウェアモジュールを制御する。さらに、CPU101は、補助メモリ103に格納された各種プログラムをメインメモリ102に読み出すとともに、該メインメモリ102に読み出した各種

50

プログラムを実行することで、各種機能を実現する。各種機能の詳細は、後述することとする。

【0022】

メインメモリ102は、CPU101により実行される各種プログラムを格納する。さらに、メインメモリ102は、CPU101のワークエリアとして使用され、CPU101による処理に必要な各種データを記憶する。メインメモリ102としては、例えばRAM(Random Access Memory)などを用いても良い。

【0023】

補助メモリ103は、携帯情報端末100を動作させる各種プログラムを格納している。各種プログラムとしては、例えば、携帯情報端末100により実行されるアプリケーションプログラムのほか、アプリケーションプログラムの実行環境であるOS(Operating System)などがある。本実施形態にかかる制御プログラムも補助メモリ103に格納されている。補助メモリ103としては、例えば、ハードディスクやフラッシュメモリ等の不揮発メモリを用いても良い。

10

【0024】

クロック供給回路104は、CPU101に供給する為のクロック信号を生成する。電圧供給回路105は、電源回路106から供給される電力に基づき、CPU101に供給する為の可変電圧を生成する。

【0025】

電源回路106は、バッテリー107から供給される電力を、電源線(図示しない)を介して、携帯情報端末100の各種ハードウェアモジュールに供給する。但し、外部給電部108に外部電源(図示しない)が接続されている場合、電源回路106は、外部給電部108から供給される電力を、携帯情報端末100の各種ハードウェアモジュールに供給しても良い。バッテリー107は、電源回路106に電力を供給する。バッテリー107としては、例えばリチウムイオン電池等の電池を使用しても良い。

20

【0026】

ディスプレイ109は、CPU101により制御され、画像情報を表示する。タッチスクリーン110は、ディスプレイ109に貼り付けられ、ユーザの指先やペン先などで接触された位置情報を入力する。

【0027】

ネットワークI/F111は、CPU101により制御され、例えばメールやツイッターの受信などのインターフェースとして機能する。

30

【0028】

センサ112は、携帯情報端末100の状態情報(端末状態)を取得する。センサ112としては、例えば加速度センサ、ジャイロセンサ、照度センサ、地磁気センサ、傾きセンサ、加圧センサ、接近センサ、温度センサ、などが想定される。

【0029】

[携帯情報端末100の機能ブロック]

図2は、第1の実施形態にかかる携帯情報端末100の機能ブロックの概略図である。

【0030】

図2に示すように、携帯情報端末100は、CPU動作周波数制御モジュール201、boostモジュール202、スケジューラ203、ミドルウェア・ドライバ204、を備える。

40

【0031】

CPU動作周波数制御モジュール201、boostモジュール202、スケジューラ203、ミドルウェア・ドライバ204は、何れもCPU101がメインメモリ102に制御プログラムを読み込むとともに、該メインメモリ102に読み込んだ制御プログラムを実行することで実現される。

【0032】

CPU動作周波数制御モジュール201は、CPU101の負荷と、制御パラメータと

50

、に基づき、CPU 101の動作周波数を制御する。制御パラメータは、例えばCPU 101の動作周波数を変更する制御アルゴリズム、動作周波数の上限値及び下限値、制御アルゴリズムや動作周波数の上限値及び下限値の継続時間、などを含む。例えば、動作周波数の下限値を100MHz、継続時間を2秒とすると、CPU動作周波数制御モジュール201は、制御パラメータの設定後2秒間、100MHz以上の動作周波数でCPU 101を動作させることとなる。

#### 【0033】

boostモジュール202は、ミドルウェア・ドライバ204からのboost依頼に基づき、CPU動作周波数制御モジュール201の制御パラメータを書き換える。さらに、boostモジュール202は、携帯情報端末100の端末状態に基づき、boostモジュール202が管理する標準クラス用パラメータを書き換える。boostモジュール202は、標準クラス用パラメータの他、中優先クラス用パラメータ及び最優先クラス用パラメータを管理する。中優先クラス用パラメータ及び最優先クラス用パラメータは、何れも事前に決められた固定値である。boostモジュール202の詳細は、後述することとする。

10

#### 【0034】

スケジューラ203は、OSカーネルのシステム状態情報からCPU 101の負荷を取得する。CPU 101の負荷としては、例えばCPU利用率を使用しても良い。

#### 【0035】

ミドルウェア・ドライバ204は、ミドルウェアもしくはデバイスドライバとして機能する。例えば、ミドルウェア・ドライバ204は、イベントの発生を契機に、boostモジュール202にboost依頼を通知する。イベントとしては、例えば画面OFFから画面ONへの切り替え（スリープモードからの復帰）、緊急地震速報のプッシュ通知、画面操作、メールのサーバチェック（以下、メールチェックとする）、などを想定する。なお、本実施形態では、画面OFFから画面ONへの切り替えや緊急地震速報のプッシュ通知を、CPU 101の能力不足（処理能力の不足や低下など）が許されないイベントとし、画面操作やメールチェックを、CPU 101の能力不足が許されるイベントとする。boost依頼は、イベントの発生に起因するboostの開始指示である。boost依頼は、少なくともイベント内容を含む。

20

#### 【0036】

（boostモジュール202の詳細）

図3は、第1の実施形態にかかるboostモジュール202の詳細な機能ブロックの概略図である。

30

#### 【0037】

図3に示すように、boostモジュール202は、boost制御部202a、パラメータ判定部202b、boost受付I/F（Interface）202c、タイマ202d、標準クラス用パラメータ格納部202e、中優先クラス用パラメータ格納部202f、最優先クラス用パラメータ格納部202g、を含む。

#### 【0038】

boost制御部202aは、boost受付I/F 202cからの最優先boost依頼もしくは非最優先boost依頼に基づき、標準クラス用パラメータ格納部202e、中優先クラス用パラメータ格納部202f、最優先クラス用パラメータ格納部202gから標準クラス用パラメータ、中優先クラス用パラメータ、最優先クラス用パラメータの何れかを取得して、CPU動作周波数制御モジュール201の制御パラメータに設定する。

40

#### 【0039】

詳細には、boost受付I/F 202cから最優先boost依頼が通知された場合、boost制御部202aは、最優先クラス用パラメータ格納部202gから最優先クラス用パラメータを取得して、CPU動作周波数制御モジュール201の制御パラメータに設定する。boost受付I/F 202cから非最優先boost依頼が通知された場

50

合、boost制御部202aは、タイマ202dからの後述する経過時間に基づき、標準クラス用パラメータ格納部202eもしくは中優先クラス用パラメータ格納部202fから、標準クラスパラメータもしくは中優先クラス用パラメータを取得して、CPU動作周波数制御モジュール201の制御パラメータに設定する。

【0040】

パラメータ判定部202bは、ミドルウェア・ドライバ204からの端末情報に基づき、標準クラス用パラメータ格納部202eに格納されている標準クラス用パラメータを更新する。具体的には、パラメータ判定部202bは、定期的にミドルウェア・ドライバ204から端末情報を取得して、端末情報が変化するたびに、変化後の端末情報に紐付けられた制御パラメータを端末状態パラメータテーブルTから読み出して、標準クラス用パラメータ格納部202dの制御パラメータに書き込む。端末状態パラメータテーブルTの詳細は、後述することとする。

10

【0041】

boost受付I/F202cは、ミドルウェア・ドライバ204からのboost依頼を受け付け、該boost依頼にかかるイベントの種類に基づき、最優先boost依頼もしくは非最優先boost依頼をboost制御部202aに通知する。最優先boost依頼を通知する場合、boost受付I/F202cは、さらに、最優先boostの発生をタイマ202dに通知する。

【0042】

タイマ202dは、boost受付I/F202cからの最優先boost発生通知に基づき、経過時間の計測を実行する。即ち、タイマ202dは、最優先boost依頼にかかるイベント発生後の経過時間を計測する。

20

【0043】

標準クラス用パラメータ格納部202eは、標準クラス用パラメータを格納する。標準クラス用パラメータは、携帯情報端末100の端末情報に基づき、動的に書き換えられる。標準クラス用パラメータを使用するboostを、標準クラスboostとする。

【0044】

中優先クラス用パラメータ格納部202fは、中優先クラス用パラメータを格納する。中優先クラス用パラメータは、事前に決められた固定パラメータである。中優先クラス用パラメータを使用するboostを、中優先クラスboostとする。

30

【0045】

最優先クラス用パラメータ格納部202gは、最優先クラス用パラメータを格納する。最優先クラス用パラメータは、事前に決められた固定パラメータである。最優先クラス用パラメータを使用するboostを、最優先クラスboostとする。中優先クラス用パラメータ及び最優先クラス用パラメータは、最優先クラスboostによるCPU101の処理能力が、中優先クラスboostによるCPU101の処理能力よりも大きくなるように設定される。

【0046】

(端末状態パラメータテーブルT)

図4は、第1の実施形態にかかる端末状態パラメータテーブルTの概略図である。

40

【0047】

図4に示すように、端末状態パラメータテーブルTは、携帯情報端末100の端末状態毎に制御パラメータを記憶する。本実施形態にかかる端末状態パラメータテーブルTでは、端末状態を画面のON/OFF及び動作中のアプリ種別の組み合わせとし、制御パラメータをCPU101の動作周波数の下限値とする。

【0048】

例えば、「画面：ON、動作中アプリ：メール」である端末状態「1」には、制御パラメータとして、動作周波数の下限値300MHzが紐付けられている。「画面：OFF、動作中アプリ：メール」である端末状態「2」には、制御パラメータとして、動作周波数の下限値100MHzが紐付けられている。但し、本実施形態にかかる端末状態は、これ

50

に限定されるものではなく、画面のON/OFFや動作中のアプリ種別の他に、携帯情報端末100への充電の有無などを組み合わせても良い。さらに、本実施形態にかかる制御パラメータは、動作周波数の下限値の他に、boostの継続時間やboostのアルゴリズムなどを組み合わせても良い。

【0049】

[boostの処理フロー]

図5は、第1の実施形態にかかるboostのフローチャートである。

【0050】

図5に示すように、boost受付I/F202cは、ミドルウェア・ドライバ204からのboost依頼に基づき、boostの対象となるイベントが発生したかどうかを判断する(ステップS001)。

10

【0051】

ここで、boostの対象となるイベントが発生したと判断された場合(ステップS001のYes)、boost受付I/F202cは、イベントが最優先クラスであるかどうかを判断する(ステップS002)。なお、最優先クラスのイベントとは、例えば画面OFFから画面ONへの切り替えや緊急地震速報のプッシュ通知などの、CPU101の能力不足が許されないイベントである。

【0052】

ここで、イベントが最優先クラスであると判断された場合(ステップS002のYes)、即ちCPU101の能力不足が許されない場合、boost受付I/F202cは、boost制御部202aに最優先boost依頼を通知する。boost制御部202aは、boost受付I/F202cからの最優先boost依頼に基づき、最優先クラスboostを実行する(ステップS003)。具体的には、boost制御部202aは、最優先クラス用パラメータ格納部202gから最優先クラス用パラメータを取得して、CPU動作周波数制御モジュール201に書き込む。こうして、CPU動作周波数制御モジュール201は、最優先クラス用パラメータを使用して、CPU101の動作周波数を制御する。これにより、CPU101は、例えば画面OFFから画面ONへの切り替えや緊急地震速報のプッシュ通知などの、あらゆるイベントを確実に処理できる処理能力を発揮することとなる。

20

【0053】

次に、boost受付I/F202cは、タイマ202dに優先boostの発生を通知して、タイマ202dに最優先クラスboostからの経過時間の計測を開始させる(ステップS004)。

30

【0054】

次に、boost受付I/F202cは、再度、ミドルウェア・ドライバ204からのboost依頼に基づき、boostの対象となるイベントが発生したかどうかを判断する(ステップS001)。

【0055】

一方、イベントが最優先クラスであると判断されない場合(ステップS002のNo)、即ちCPU101の能力不足が許される場合、boost受付I/F202cは、boost制御部202aに非最優先boost依頼を通知する。boost制御部202aは、boost受付I/F202cからの非最優先boost依頼に基づき、最優先クラスboostからの経過時間がしきい値以上であるかどうかを判断する(ステップS005)。

40

【0056】

ここで、最優先クラスboostからの経過時間がしきい値以上であると判断された場合(ステップS005のYes)、boost制御部202aは、最優先クラスboost後の端末状態に紐付けられた標準クラス用パラメータが標準クラス用パラメータ格納部202eに記憶されたと判断して、標準クラスboostを実行する(ステップS006)。具体的には、boost制御部202aは、標準クラス用パラメータ格納部202e

50



から標準クラス用パラメータを読み出して、CPU動作周波数制御モジュール201に書き込む。こうして、CPU動作周波数制御モジュール201は、標準クラス用パラメータを使用して、CPU101の動作周波数を制御する。これにより、CPU101は、それぞれのイベントを処理できる最適な処理能力を発揮することとなる。

【0057】

次に、boost受付I/F202cは、再度、ミドルウェア・ドライバ204からのboost依頼に基づき、boostの対象となるイベントが発生したかどうかを判断する(ステップS001)。

【0058】

一方、最優先クラスboostからの経過時間がしきい値以上であると判断されない場合(ステップS005のNo)、boost制御部202aは、最優先クラスboost後の端末状態に紐付けられた標準クラス用パラメータが標準クラス用パラメータ格納部202eに記憶されていないと判断して、中優先クラスboostを実行する(ステップS007)。具体的には、boost制御部202aは、中優先クラス用パラメータ格納部202fから中優先クラス用パラメータを読み出して、CPU動作周波数制御モジュール201に書き込む。こうして、CPU動作周波数制御モジュール201は、中優先クラス用パラメータを使用して、CPU101の動作周波数を制御する。これにより、CPU101は、あらゆる通常のイベント、即ち最優先クラスのイベントではないイベントを確実に処理できる処理能力を発揮することとなる。

【0059】

次に、boost受付I/F202cは、再度、ミドルウェア・ドライバ204からのboost依頼に基づき、boostの対象となるイベントが発生したかどうかを判断する(ステップS001)。

【0060】

一方、boostの対象となるイベントが発生したと判断されない場合(ステップS001のNo)、パラメータ判定部202bは、ミドルウェア・ドライバ204からの各種情報に基づき、携帯情報端末100の端末状態を特定する(ステップS008)。具体的には、パラメータ判定部202bは、ミドルウェア・ドライバ204からの各種情報に基づき、画面のON/OFFや動作中アプリの種別などを取得して、携帯情報端末100の端末状態を特定する。

【0061】

次に、パラメータ判定部202bは、端末状態パラメータテーブルTを参照して、携帯情報端末100の端末状態に紐付けられた制御パラメータを取得する(ステップS009)。例えば、端末状態が「画面：ON、動作中アプリ：メール」であれば、パラメータ判定部202bは、端末状態パラメータテーブルTから端末状態「1」に紐付けられた制御パラメータ「下限値：300MHz」を取得することとなる。

【0062】

次に、パラメータ判定部202bは、標準クラス用パラメータ格納部202eに格納された標準クラス用パラメータを、携帯情報端末100の端末状態に紐付けられた制御パラメータに書き換える(ステップS010)。例えば、端末状態「1」であれば、パラメータ判定部202bは、標準クラス用パラメータを「下限値：300MHz」に更新することとなる。

【0063】

[boostの具体例]

図6は、第1の実施形態にかかるboostの具体例の説明図である。

【0064】

図中のパート(a)は、boostを実行したときのCPU101の処理能力及び処理負荷(実際の処理量)のグラフである。グラフでは、横軸を時間(sec)、縦軸を処理能力(MHz)としている。実線は、CPU101の処理能力、点線(破線)は、CPU101の処理負荷である。図中のパート(b)は、携帯情報端末100の端末状態、端末

10

20

30

40

50

状態の判定状況、boostに使用するパラメータの関係である。

【0065】

端末状態の判定状況が「判定済」かつ端末状態が「画面：OFF、動作中アプリ：メール」である区間R1にメールチェックのイベントが発生すると、boostモジュール202は、端末状態に紐付けられた標準クラス用パラメータを使用して、標準クラスboostを実施する。即ち、端末状態が「画面：OFF、動作中アプリ：メール」であれば、携帯情報端末100が意識的に操作されていないと推測できる為（ポケットに入っているなど）、boostモジュール202は、端末状態に紐付けられた標準クラス用パラメータを使用することで、CPU101の処理能力を低調に維持したまま、メールチェックのイベントを処理する。これにより、イベント処理の為に消費電力を削減する。

10

【0066】

一方、画面OFFから画面ONへの切り替えのイベントが発生すると、boostモジュール202は、携帯情報端末100の端末状態に関係なく、最優先クラス用パラメータを使用して、最優先クラスboostを実行する。即ち、画面OFFから画面ONへの切り替えのイベントは、CPU101の能力不足が許されないイベントである為、boostモジュール202は、最優先クラス用パラメータを使用して、CPU101の処理能力を大幅に上昇させることで、イベントを確実に処理することを優先する。

【0067】

他方、画面OFFから画面ONへの切り替えのイベント後の経過時間がしきい値未満である区間R2、即ちCPU101の能力不足が許されないイベント後の経過時間がしきい値未満である区間R2に画面操作やメールチェックなどのイベントが発生すると、boostモジュール202は、イベントの種別に関係なく、中優先クラス用パラメータを使用して、中優先クラスboostを実行する。即ち、CPU101の能力不足が許されないイベントの終了後は、例えば画面操作などの比較的高負荷のイベントが発生して、端末状態の判定終了が遅延することが予想される為、boostモジュール202は、中優先クラス用パラメータを使用することで、イベントを確実に処理することを優先する。

20

【0068】

以上のように、本実施形態では、携帯情報端末100に発生したイベントが最優先クラスである場合、即ちCPU101の能力不足が許されない特定のイベントである場合、事前に用意された、あらゆるイベントを確実に処理できる制御パラメータ（最優先クラス用パラメータ）を使用して、最優先クラスboostを実行する。このため、CPU101の能力不足に起因するイベント処理の失敗を回避することができる。

30

【0069】

又、携帯情報端末100に発生したイベントが最優先クラスでない場合、即ちCPU101の能力不足が許される場合、携帯情報端末100の端末状態に紐付けられた制御パラメータ（標準クラス用パラメータ）を使用して、標準クラスboostを実施する。このため、携帯情報端末100の端末状態に最適なboostが実行される為、無駄な消費電力を削減することができる。

【0070】

さらに、携帯情報端末100に発生した特定のイベントに起因する端末状態の変更後の所定期間は、事前に用意された、通常のイベントを確実に処理できる制御パラメータ（中優先クラス用パラメータ）を使用して、中優先クラスboostを実行する。このため、携帯情報端末100の端末状態の変更直後でも、CPU101の能力不足に起因するイベント処理の失敗を回避することができる。

40

【0071】

[第2の実施形態]

以下、図7 - 図9を参照して、第2の実施形態を説明する。但し、第1の実施形態と同等の構成、機能、作用については、説明を省略することとする。

【0072】

本実施形態では、携帯情報端末100Aの充電直前、バッテリー107の電池残量の減少

50

に付随して、携帯情報端末 100A が省エネモード（エコモード）に遷移していると想定する。なお、省エネモードとは、CPU 101 の処理能力を抑制して、電力消費を削減するモードである。

【0073】

携帯情報端末 100A が省エネモードであるときに、携帯情報端末 100A の充電が開始されると、消費電力を抑制する必要がなくなるが、CPU 101 の処理能力が抑制されている為、充電直後、端末状態の判定終了が遅延することが予想される。このため、携帯情報端末 100A の端末状態の変更後（「非充電中」「充電中」）、標準クラス用パラメータ格納部 202e の制御パラメータが更新されるまで、消費電力を抑制する必要がないにも拘わらず、CPU 101 の処理能力が抑制され、イベント処理に膨大な時間がかかることがある。

10

【0074】

以上の問題を解決する為に、本実施形態では、例えば携帯情報端末 100A の充電開始からの経過時間がしきい値未満である場合、boost モジュール 202A は、携帯情報端末 100A の端末状態に依存しない中優先クラス用パラメータを使用して、中優先クラス boost を実行する。これにより、本実施形態では、携帯情報端末 100A の充電の開始直後に発生したイベントも確実に処理することができる。

【0075】

（boost モジュール 202A の詳細）

図 7 は、第 2 の実施形態にかかる携帯情報端末 100A の機能ブロックの概略図である。

20

【0076】

図 7 に示すように、第 2 の実施形態にかかる携帯情報端末 100A は、第 1 の実施形態にかかる boost モジュール 202 の代わりに、boost モジュール 202A を備える。

【0077】

boost モジュール 202A は、CPU 101 がメインメモリ 102 に制御プログラムを読み込むとともに、該メインメモリ 102 に読み込んだ制御プログラムを実行することで実現される。

【0078】

boost モジュール 202A は、イベント通知 I/F 202h を含む。イベント通知 I/F 202h は、ミドルウェア・ドライバ 204 からのイベント情報を受け付ける。さらに、イベント通知 I/F 202h は、携帯情報端末 100A に特定のイベントが発生した場合、タイマ 202d に特定イベントの発生を通知する。特定のイベントとしては、例えば携帯情報端末 100A への充電開始である。

30

【0079】

[boost の処理フロー]

図 8 は、第 2 の実施形態にかかる boost のフローチャートである。

【0080】

図 8 に示すように、イベント通知 I/F 202h は、ミドルウェア・ドライバ 204 からのイベント情報に基づき、特定のイベント、例えば携帯情報端末 100A への充電が発生したかどうかを判断する（ステップ S021）。

40

【0081】

ここで、boost の対象となる特定のイベントが発生したと判断された場合（ステップ S021 の Yes）、イベント受付 I/F 202h は、特定のイベントの発生をタイマ 202d に通知して、特定のイベントからの経過時間の計測を開始させる（ステップ S022）。

【0082】

次に、イベント受付部 I/F 202h は、再度、ミドルウェア・ドライバ 204 からのイベント情報に基づき、特定のイベントが発生したかどうかを判断する（ステップ S02

50

1)。

【0083】

一方、特定のイベントが発生したと判断されない場合(ステップS021のNo)、第1の実施形態と同様、boost受付I/F202cは、ミドルウェア・ドライバ204からのboost依頼に基づき、boostの対象となるイベントが発生したかどうかを判断する(ステップS001)。以降の処理は、第1の実施形態と同等である。

【0084】

[boostの具体例]

図9は、第2の実施形態にかかるboostの具体例の説明図である。

【0085】

図中のパート(a)は、boostを実行したときのCPU101の処理能力及び処理負荷(実際の処理量)のグラフである。グラフでは、横軸を時間(sec)、縦軸を処理能力(MHz)としている。実線は、CPU101の処理能力、点線(破線)は、CPU101の処理負荷である。図中のパート(b)は、携帯情報端末100Aの端末状態、端末状態の判定状況、boostに使用するパラメータの関係である。

【0086】

端末状態の判定状況が「判定済」かつ端末状態が「充電：非充電中、モード：省エネ」である区間R3に画面操作のイベントが発生すると、boostモジュール202Aは、端末状態に紐付けられた標準クラス用パラメータを使用して、標準クラスboostを実施する。即ち、端末状態が「充電：非充電中、モード：省エネ」であれば、消費電力の削減を優先すべきである為、boostモジュール202Aは、端末状態に紐付けられた標準クラス用パラメータを使用することで、CPU101の処理能力を低調に維持したまま、画面操作のイベントを処理する。これにより、イベント処理の為の消費電力の増加を回避する。

【0087】

一方、携帯情報端末100Aの充電開始のイベント後の経過時間がしきい値未満である区間R4に画面操作のイベントが発生すると、boostモジュール202Aは、イベントの種別に関係なく、中優先クラス用パラメータを使用して、中優先クラスboostを実行する。即ち、充電開始の直後は、携帯情報端末100Aの端末状態の判定が終了していないことが予想される為、boostモジュール202Aは、事前に用意された、中優先クラス用パラメータを使用することで、イベントを確実に処理することを優先する。

【0088】

以上のように、本実施形態にかかるboostモジュール202Aは、特定のイベントの開始後の経過時間、例えば携帯情報端末100Aの充電開始後の経過時間がしきい値未満であれば、事前に用意しておいた中優先クラス用パラメータを使用して、中優先クラスboostを実行する為、例えば携帯情報端末100Aの充電開始直後の、端末情報を把握できない区間でも、CPU101の能力不足に起因するイベント処理の失敗を回避することができる。

【符号の説明】

【0089】

100：携帯端情報末  
 100A：携帯情報端末  
 101：CPU  
 201：CPU動作周波数制御モジュール  
 202：boostモジュール  
 202a：boost制御部  
 202b：パラメータ判定部  
 202c：boost受付I/F  
 202d：タイマ  
 202e：標準クラス用パラメータ格納部

10

20

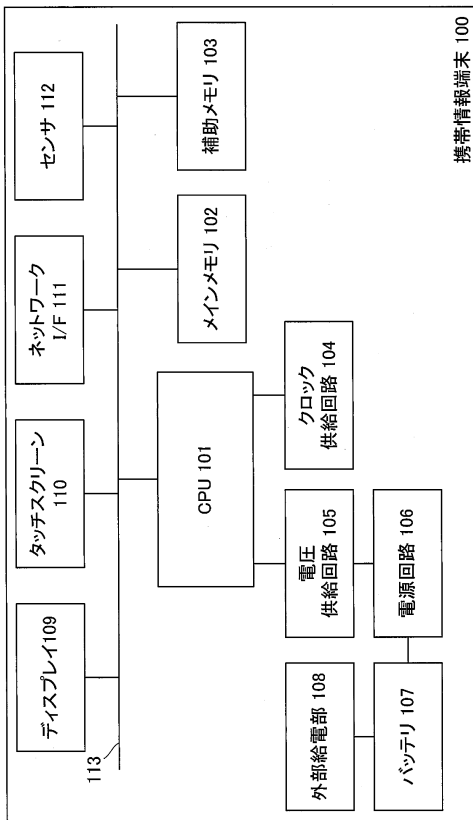
30

40

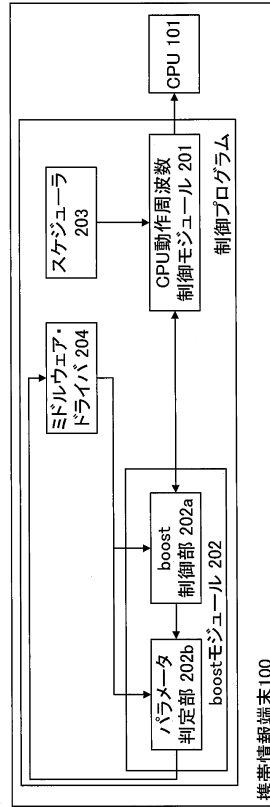
50

- 202f : 中優先クラス用パラメータ格納部
- 202g : 最優先クラス用パラメータ格納部
- 202h : イベント通知I/F
- T : 端末状態パラメータテーブル

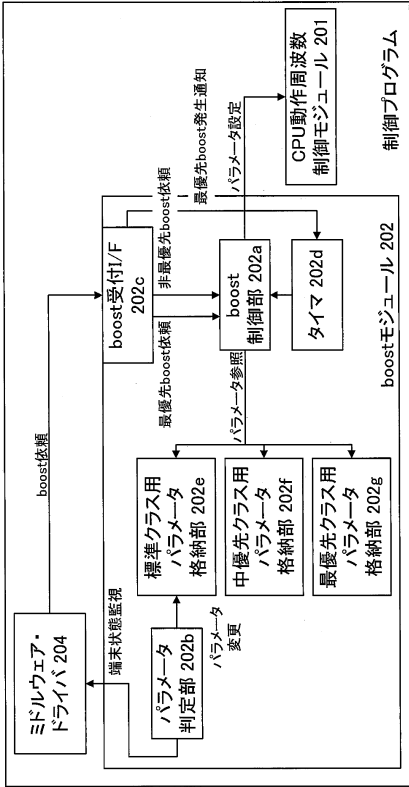
【図1】



【図2】



【図3】

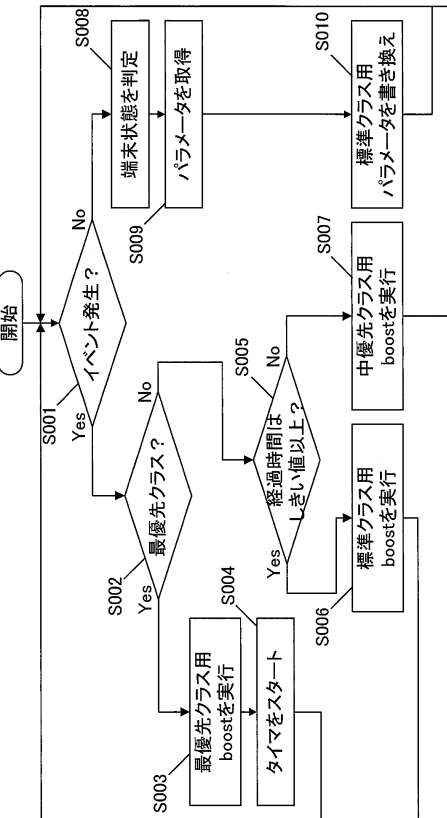


【図4】

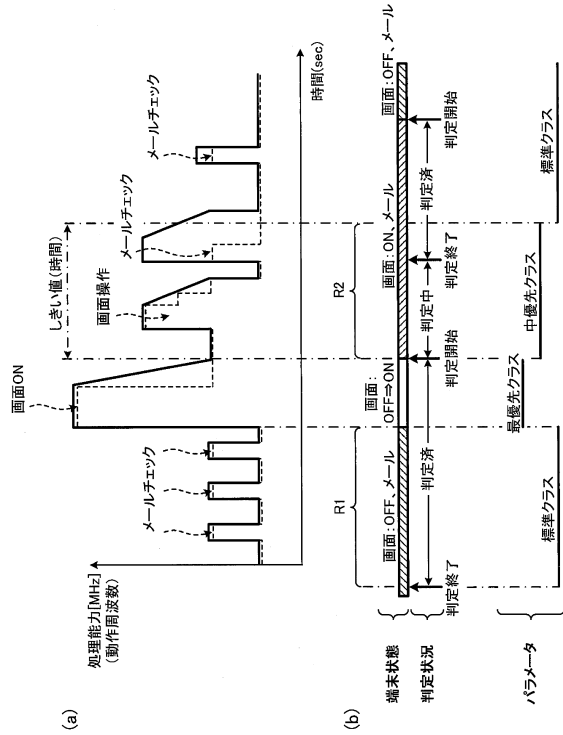
端末状態パラメータテーブル T

端末状態	画面	アプリ	...	制御パラメータ
1	on	メール	...	下限値: 300MHz
2	off	メール	...	下限値: 100MHz
...	...	...	...	...

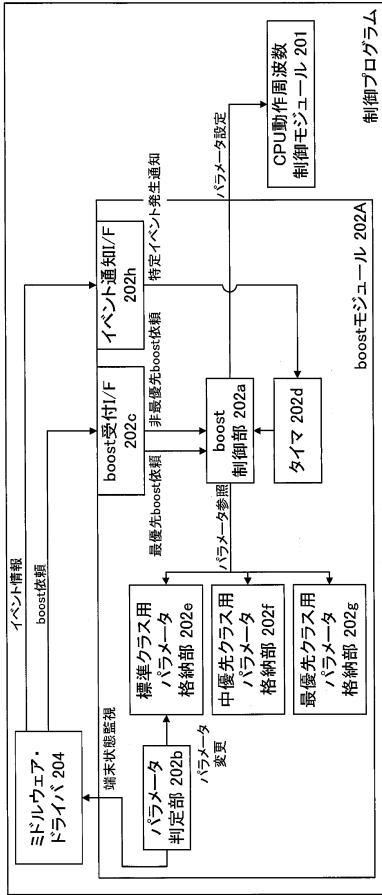
【図5】



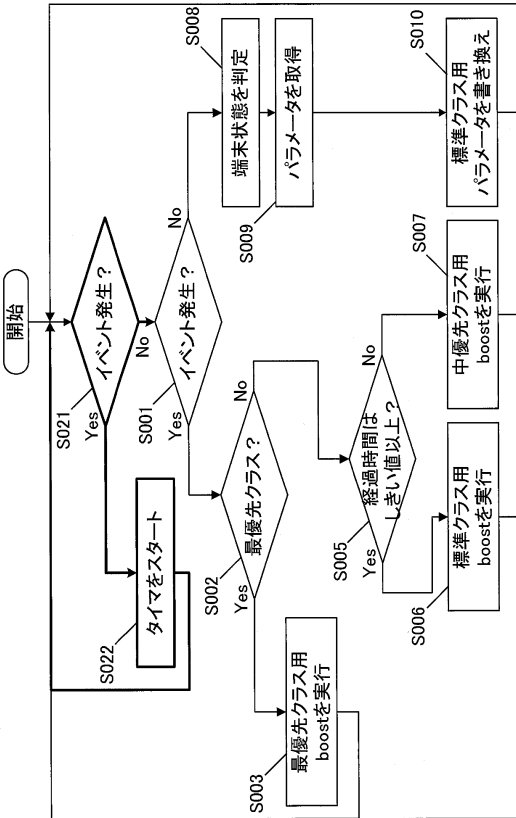
【図6】



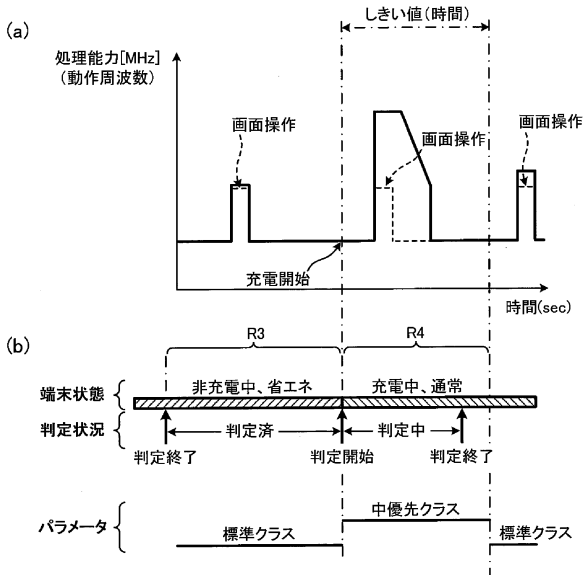
【 図 7 】



【 図 8 】



【 図 9 】



---

フロントページの続き

- (56)参考文献 特開2002-157042(JP,A)  
特開2010-146280(JP,A)  
特開2003-298499(JP,A)  
国際公開第2011/145198(WO,A1)

(58)調査した分野(Int.Cl., DB名)

G06F	1/04	-	G06F	1/14
G06F	1/26	-	G06F	1/32
H04M	1/73			