

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4338842号
(P4338842)

(45) 発行日 平成21年10月7日 (2009. 10. 7)

(24) 登録日 平成21年7月10日 (2009. 7. 10)

(51) Int. Cl.		F I			
G 0 6 F	1/26	(2006. 01)	G 0 6 F	1/00	3 3 4 H
G 0 6 F	1/32	(2006. 01)	G 0 6 F	1/00	3 3 2 Z

請求項の数 13 (全 34 頁)

(21) 出願番号	特願平11-264359	(73) 特許権者	000005821
(22) 出願日	平成11年9月17日 (1999. 9. 17)		パナソニック株式会社
(65) 公開番号	特開2000-155626 (P2000-155626A)		大阪府門真市大字門真1006番地
(43) 公開日	平成12年6月6日 (2000. 6. 6)	(74) 代理人	100098291
審査請求日	平成18年5月17日 (2006. 5. 17)		弁理士 小笠原 史朗
(31) 優先権主張番号	特願平10-264627	(72) 発明者	中嶋 直紀
(32) 優先日	平成10年9月18日 (1998. 9. 18)		大阪府門真市大字門真1006番地 松下
(33) 優先権主張国	日本国 (JP)		電器産業株式会社内
		審査官	緑川 隆
		(56) 参考文献	特開平05-265602 (JP, A)
			特開平07-219688 (JP, A)
			特開平08-234872 (JP, A)
			特開平09-069017 (JP, A)
			最終頁に続く

(54) 【発明の名称】 電力制御装置、電力制御方法、及び記録媒体

(57) 【特許請求の範囲】

【請求項 1】

アプリケーションプログラムを実行する電気機器において、アプリケーションプログラムを実行するためのリソースの消費電力を制御する電力制御装置であって、

前記アプリケーションプログラムの動作状態を検出する動作状態検出手段と、

前記検出されたアプリケーションプログラムの動作状態において前記リソースが必要とする電力供給率を算出する電力供給率算出手段と、

前記リソースが前記電力供給率算出手段によって算出された電力供給率で動作するように、前記リソースに電力供給率を割り当てる電力供給率割当手段と、

前記アプリケーションプログラムの動作状態に応じて予め設定された前記リソースの電力供給率を記憶する電力供給率記憶手段と、

を備え、

前記電力供給率算出手段は、

前記動作状態検出手段によって検出された動作状態に対応するリソースの電力供給率を前記電力供給率記憶手段から読み出す電力供給率取得手段を含み、

前記電力供給率取得手段によって同一のリソースに対して複数の電力供給率が取得される場合に当該複数の電力供給率を統合する電力供給率統合手段をさらに含み、

同時に複数のアプリケーションプログラムの動作状態を有するリソースも、前記アプリケーションプログラムの個々の動作の実行に対して十分な電力で駆動されることを特徴とする電力制御装置。

10

20

【請求項 2】

前記電力供給率統合手段は、前記同一のリソースに対して取得された複数の電力供給率の合計値が 100% 以下の場合には、当該電力供給率の合計値に統合し、

前記複数の電力供給率の合計値が 100% 以上の場合には、100% に統合し、

リソースの最大能力以上に電力供給されることを防止することを特徴とする請求項 1 に記載の電力制御装置。

【請求項 3】

前記動作状態検出手段は、

前記電気機器にロードされているオペレーションシステムの API 層で検出される API 通信情報とカーネル層で検出される動作状態監視情報に基づいて、前記アプリケーションプログラムの動作状態を検出することを特徴とする請求項 1 に記載の電力制御装置。

10

【請求項 4】

前記電力供給率統合手段は、統合された電力供給率を、前記電気機器にロードされているオペレーションシステムのドライバインターフェース層に供給することを特徴とする請求項 1 に記載の電力制御装置。

【請求項 5】

前記ドライバインターフェース層は、統合された電力供給率を対応するリソースのドライバに供給することを特徴とする請求項 4 に記載の電力制御装置。

【請求項 6】

アプリケーションプログラムを実行する電気機器において、アプリケーションプログラムを実行するためのリソースの消費電力を制御する電力制御方法であって、

20

前記アプリケーションプログラムの動作状態を検出する動作状態検出ステップと、

前記検出されたアプリケーションプログラムの動作状態において前記リソースが必要とする電力供給率を算出する電力供給率算出ステップと、

前記リソースが前記電力供給率算出ステップによって算出された電力供給率で動作するように、前記リソースに電力供給率を割り当てる電力供給率割当ステップと、

前記アプリケーションプログラムの動作状態に応じて予め設定された前記リソースの電力供給率を記憶する電力供給率記憶ステップと、

を備え、

前記電力供給率算出ステップは、

30

前記電力供給率記憶ステップで記憶された電力供給率の中から、前記動作状態検出ステップにおいて検出された動作状態に対応するリソースの電力供給率を読み出す電力供給率取得ステップを含み、

前記電力供給率取得ステップによって同一のリソースに対して複数の電力供給率が取得される場合に当該複数の電力供給率を統合する電力供給率統合ステップを含み、

同時に複数のアプリケーションプログラムの動作状態を有するリソースも、前記アプリケーションプログラムの個々の動作の実行に対して十分な電力で駆動されることを特徴とする電力制御方法。

【請求項 7】

前記電力供給率統合ステップは、前記同一のリソースに対して取得された複数の電力供給率の合計値が 100% 以下の場合には、当該電力供給率の合計値に統合し、

40

前記複数の電力供給率の合計値が 100% 以上の場合には、100% に統合し、

リソースの最大能力以上に電力供給されることを防止することを特徴とする請求項 6 に記載の電力制御方法。

【請求項 8】

前記動作状態検出ステップは、

前記電気機器にロードされているオペレーションシステムの API 層で検出される API 通信情報とカーネル層で検出される動作状態監視情報に基づいて、前記アプリケーションプログラムの動作状態を検出することを特徴とする請求項 6 に記載の電力制御方法。

【請求項 9】

50

前記電力供給率統合ステップで統合された電力供給率は、前記電気機器にロードされているオペレーションシステムのドライバインターフェース層に供給されることを特徴とする請求項6に記載の電力制御方法。

【請求項10】

前記ドライバインターフェース層は、統合された電力供給率に対応するリソースのドライバに供給することを特徴とする請求項9に記載の電力制御方法。

【請求項11】

コンピュータに請求項6から10までの何れかに記載の電力制御方法を実行させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【請求項12】

前記電力供給率算出手段は、さらに、異なるリソース毎に必要とする電力供給率を算出することによって、

前記リソースは個々に、前記アプリケーションプログラムの動作状態に応じて前記電力供給率が変動することにより、当該アプリケーションプログラムを実行するのに適した電力で駆動されることを特徴とする請求項1に記載の電力制御装置。

【請求項13】

前記電力供給率算出ステップは、さらに、異なるリソース毎に必要とする電力供給率を算出することによって、

前記リソースは個々に、前記アプリケーションプログラムの動作状態に応じて前記電力供給率が変動することにより、当該アプリケーションプログラムを実行するのに適した電力で駆動されることを特徴とする請求項6に記載の電力制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、パーソナルコンピュータ等の情報処理装置において実行中のアプリケーションプログラムの動作状態の変化に応じて、同情報処理装置における消費電力の制御方法に関し、さらに詳述すれば、電池駆動可能な携帯型のパーソナルコンピュータに用いられる電力制御方法およびその装置に関する。

【0002】

【従来の技術】

従来、電池駆動可能なパーソナルコンピュータにおいて、アプリケーションの動作状態に応じて電力を制御する方法としては、大別して以下に示す4種類の方法が知られている。

【0003】

第1の方法は、特開平4-125718号公報に記載の「省電力制御方式」である。同方法においては、アプリケーションプログラムが入力待ち状態かつ入力装置からの入力がない状態が、一定時間継続した時に、CPUのクロックの供給或いは電力供給を停止する。その結果、消費電力を低減して、電池の稼働時間の延長或いは電池容量の削減を図る。つまり、CPUへ入力される負荷が少ない或いはない場合には、CPUを一律に休眠状態にすることによって省電力を図る。

【0004】

第2の方法は、特開平7-302138号公報に記載の「電子機器および該機器における電源制御方法」である。同方法においては、先ず、アプリケーションプログラム毎に各デバイスへのアクセス履歴を記録しておく。次に、アクセス履歴の記録に基づいて、当該アプリケーションプログラムがそのデバイスにアクセスする時刻を予想する。そして、予測時刻に基づいて、当該デバイスを省電力モードに移行させる。つまり、デバイスを現時点から予測時刻まで省電力モードに移行させることによって期待できる節約電力量と、予測時刻に通常モードに復帰させるために要する電力量を比較して、消費電力量がより小さくなるようにデバイスの動作モードを制御する。

【0005】

第3の方法は、特表平8-503566号公報（対応US Pat No. 5, 339, 4

10

20

30

40

50

45)に記載の「コンピュータ装置の電力消費を自動的に減少させる方法」である。同方法においては、アプリケーションプログラムの動作中に、装置の電力消費機器の電力消費を監視して、装置資源の特性を保持するコンピュータ装置のテーブルに記録する。そして、アプリケーションプログラムが再び作動される時に、必要とされる資源の特性をテーブルから呼び出して、装置の機器に供給する電力を自動的に調節することによって、コンピュータ装置の電力消費を低減する。

【0006】

また、第4の方法としては、家庭や事務所に供給されているAC電源駆動および電池駆動の両方が可能な情報処理装置に対して、一般的に用いられる電力制御方法がある。同方法においては、情報処理装置が、AC電源か電池のどちらで駆動されているかを判断する。そして、電池で駆動されている場合には、ユーザが予め設定しておいた電力制御方法で動作させる。この方法では、電池駆動の時にCPUの動作速度やLCDの輝度をユーザが指定した固定値で下げて電力消費量を低減して電池駆動時間を延長する。

【0007】

しかしながら、近年のCPUの処理速度やハードディスク等の周辺機器の性能の飛躍的な向上に伴うCPUや周辺機器の性能を最大限に利用するアプリケーションプログラムの急速な増加に対しては、上述の従来の電力制御方法では対応できない。そのようなアプリケーションプログラムの代表的な例としては、動画再生、或いは動画録画等の動画像処理を行うアプリケーションソフトウェアや、モデム機能を実現するアプリケーションソフトウェアなどがある。

【0008】

【発明が解決しようとする課題】

上述の状況下では、上述の第1および第4の電力制御方法のように、CPUの動作速度や周辺機器の消費電力を固定値になるように一律に下げるという方法で電力制御を行うと、CPUや周辺機器の性能を最大限に利用する動画像処理アプリケーションプログラム等が動作できないか、或いは十分にその性能を発揮できない。

【0009】

また、上述の第2および第3の電力制御方法は、アプリケーションプログラムの使用するデバイス毎に電力制御を行うという点で優れている。しかし、アプリケーションプログラムの実行に関して、プログラムの起動時や動画の再生時等のように、CPUや周辺装置の性能を特に最大限活用する要求がある場合に、その要求の変動に応じてCPU速度や電力量を変化させるものではない。従って、アプリケーションプログラムの動作内容によって、ハードウェア毎に、そのリソースに要求される負荷が大きく変動する場合には、そのような要求の変動に適切に対応した電力制御は不可能である。よって、第1および第4の電力制御方法と同様に、アプリケーションプログラムが動作できないか、或いは十分にその性能を発揮できない場合がある。

【0010】

なお、従来の電力制御方法の何れにおいても、アプリケーションプログラムを適正に実行させながら電力制御を行うには、ユーザが電力制御方法を、予め詳細に設定しておくか、常にCPU速度を十分に大きな値に設定しておく必要がある。電力制御方法の詳細設定を、アプリケーションプログラムを実行中に、逐次、手動で行うことは非常に面倒である。あえて、そのような面倒な詳細設定を行ったとしても、アプリケーションプログラムの実行中に、正しいタイミングで行うことは非常に難しい。また、CPU速度を十分に大きな値で固定的に設定しておく、不必要に電力消費量が増加し、電池駆動時間が減少してしまい、電力制御の本来の目的を達成できない。

【0011】

なお、上述の従来の電力制御方法においては、電池駆動の情報機器の駆動時間の延長或いは、同一の駆動時間を提供するのに必要な電池のサイズを小さくすることを目的としている。そのため、アプリケーションプログラムの実行時に、ハードウェアの各リソースに供給する電力を如何に低減するかという観点でしか電力制御を考えていない。それゆえ、上

10

20

30

40

50

述のように、消費電力は低減できるがアプリケーションプログラムを正常に実行できないという問題を有している。

【0012】

また、現在においては、地球温暖化やエネルギー資源の枯渇が現実には大きな問題になっており、消費電力の低減は、単に電池駆動の情報機器に限定されるものではなく、AC電源で駆動される電気機器全般にも要求されている技術である。

【0013】

よって、アプリケーションプログラムの実行に際して、アプリケーションプログラムの実行状態の変化に応じて、ハードウェアの各リソースを適切に駆動させるのに必要な最低電力を供給することが必要とされている。そのことによって、アプリケーションプログラムの実行において、ハードウェアにアプリケーションプログラム実行に必要な以上に電力を過剰供給して、その過剰な電力が無駄に消費されることを防止する。その結果、アプリケーションプログラムの実行に対して、電力が無駄なく効率的に利用されると共に、電力消費を低減できる。

【0014】

本発明は以上の問題を解決するものであり、情報機器に代表される電気機器において、アプリケーションプログラムの実行状態に応じて、各リソースに電力を適正かつ効果的に配分し、電力の過剰供給による無駄な消費を防止し、その結果消費電力を低減する電力制御方法およびその装置を提供することを目的とする。

【0015】

【課題を解決するための手段および発明の効果】

第1の発明は、アプリケーションプログラムを実行する電気機器において、アプリケーションプログラムを実行するためのリソースの消費電力を制御する電力制御装置であって、

アプリケーションプログラムの動作状態を検出する動作状態検出器と、

検出されたアプリケーションプログラムの動作状態においてリソースが必要とする必要とする電力供給率を算出する電力供給率算出器と、

リソースが電力供給率算出器によって算出された電力供給率で動作するように、リソースに電力供給率を割り当てる電力供給率割り当器と、

アプリケーションプログラムの動作状態に応じて予め設定されたリソースの電力供給率を記憶する電力供給率記憶器と、を備え、

電力供給率算出器は、動作状態検出器によって検出された動作状態に対応するリソースの電力供給率を電力供給率記憶器から読み出す電力供給率取得器を含み、

電力供給率取得器によって同一のリソースに対して複数の電力供給率が取得される場合に当該複数の電力供給率を統合する電力供給率統合器をさらに含み、

同時に複数のアプリケーションプログラムの動作状態を有するリソースも、アプリケーションプログラムの個々の動作の実行に対して十分な電力で駆動されることを特徴とする電力制御装置。

【0016】

上述のように、第1の発明においては、複数のアプリケーションプログラムの同時実行の場合に、アプリケーションプログラムの個々の動作の実行に対して、同一のリソースに同時に必要とされる電力供給率を、システム全体に対する電力供給率に応じて正しく決定できる。

【0021】

第2の発明は、第1の発明において、電力供給率統合器は、同一のリソースに対して取得された複数の電力供給率の合計値が100%以下の場合には、電力供給率の合計値に統合し、複数の電力供給率の合計値が100%以上の場合には、100%に統合し、リソースの最大能力以上に電力供給されることを防止することを特徴とする。

【0022】

上述のように、第2の発明においては、リソースの実際の能力に応じてアプリケーショ

10

20

30

40

50

ンプログラムからの電力供給要求に対して最大限応えることができる。

【0023】

第3の発明は、第1の発明において、動作状態検出器は、電気機器にロードされているオペレーションシステムのAPI層で検出されるAPI通信情報とカーネル層で検出される動作状態監視情報に基づいて、アプリケーションプログラムの動作状態を検出することを特徴とする。

【0024】

上述のように、第3の発明においては、オペレーションシステムを有するコンピュータシステムに本発明を適用できる。

【0025】

第4の発明は、第1の発明において、電力供給率統合器は、統合された電力供給率を、電気機器にロードされているオペレーションシステムのドライバインターフェース層に供給することを特徴とする。

【0026】

上述のように、第4の発明においては、オペレーションシステムを有するコンピュータシステムに本発明を適用できる。

【0027】

第5の発明は、第4の発明において、ドライバインターフェース層は、統合された電力供給率に対応するリソースのドライバに供給することを特徴とする。

【0028】

上述のように、第5の発明においては、コンピュータシステムに接続されている任意の周辺機器の電力消費量を制御できる。

【0030】

第6の発明は、アプリケーションプログラムを実行する電気機器において、アプリケーションプログラムを実行するためのリソースの消費電力を制御する電力制御方法であって、

アプリケーションプログラムの動作状態を検出する動作状態検出ステップと、

検出されたアプリケーションプログラムの動作状態においてリソースが必要とする電力供給率を算出する電力供給率算出ステップと、

電力供給率算出ステップによって算出された電力供給率で動作するように、リソースに電力供給率を割り当てる電力供給率割当ステップと、

アプリケーションプログラムの動作状態に応じて予め設定されたリソースの電力供給率を記憶する電力供給率記憶ステップと、を備え、

電力供給率算出ステップは、電力供給率記憶ステップで記憶された電力供給率の中から、動作状態検出ステップにおいて検出された動作状態に対応するリソースの電力供給率を読み出す電力供給率取得ステップを含み、

電力供給率取得ステップによって同一のリソースに対して複数の電力供給率が取得される場合に当該複数の電力供給率を統合する電力供給率統合ステップを含み、

同時に複数のアプリケーションプログラムの動作状態を有するリソースも、アプリケーションプログラムの個々の動作の実行に対して十分な電力で駆動されることを特徴とする電力制御方法。

【0031】

上述のように、第6の発明においては、複数のアプリケーションプログラムの同時実行の場合に、アプリケーションプログラムの個々の動作の実行に対して、同一のリソースに同時に必要とされる電力供給率を、システム全体に対する電力供給率に応じて正しく決定できる。

【0036】

第7の発明は、第6の発明において、電力供給率統合ステップは、同一のリソースに対して取得された複数の電力供給率の合計値が100%以下の場合には、電力供給率の合計値に統合し、

10

20

30

40

50

複数の電力供給率の合計値が100%以上の場合には、100%に統合し、リソースの最大能力以上に電力供給されることを防止することを特徴とする。

【0037】

上述のように、第7の発明においては、リソースの実際の能力に応じてアプリケーションプログラムからの電力供給要求に対して最大限応えることができる。

【0038】

第8の発明は、第6の発明において、動作状態検出ステップは、電気機器にロードされているオペレーションシステムのAPI層で検出されるAPI通信情報とカーネル層で検出される動作状態監視情報に基づいて、アプリケーションプログラムの動作状態を検出することを特徴とする。

10

【0039】

上述のように、第8の発明においては、オペレーションシステムを有するコンピュータシステムに本発明を適用できる。

【0040】

第9の発明は、第6の発明において、電力供給率統合ステップは、統合された電力供給率は、電気機器にロードされているオペレーションシステムのドライバインターフェース層に供給されることを特徴とする。

【0041】

上述のように、第9の発明においては、オペレーションシステムを有するコンピュータシステムに本発明を適用できる。

20

【0042】

第10の発明は、第9の発明において、ドライバインターフェース層は、統合された電力供給率に対応するリソースのドライバに供給することを特徴とする。

【0043】

上述のように、第10の発明においては、コンピュータシステムに接続されている任意の周辺機器の電力消費量を制御できる。

【0047】

第11の発明は、コンピュータに請求項6から10までの何れかに記載の電力制御方法を実行させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【0049】

30

第12の発明は、第1の発明において、電力供給率決定器は、さらに、異なるリソース毎に必要な電力供給率を決定することによって、リソースは個々に、アプリケーションプログラムの動作状態に応じて電力供給率が変動することにより、アプリケーションプログラムを実行するのに適した電力で駆動されることを特徴とする。

【0050】

第13の発明は、第6の発明において、電力供給率算出ステップは、さらに、異なるリソース毎に必要な電力供給率を算出することによって、

リソースは個々に、アプリケーションプログラムの動作状態に応じて電力供給率が変動することにより、アプリケーションプログラムを実行するのに適した電力で駆動されることを特徴とする。

40

【0051】

【発明の実施の形態】

先ず、添付の図1～図11を参照して、本発明の第1の実施形態にかかる電力制御装置について説明する。その後、図12～図32を参照して、本発明の第2の実施形態にかかる電力制御装置について説明する。

【0052】

(第1の実施形態)

図1に、本発明の第1の実施形態にかかる電力制御装置の構造を示す。電力制御装置PCA1は、動作状態検出器101、電力供給率算出器102、電力供給率記憶器103、電力供給率割当器104、CPU105、およびN個(Nは正の整数)の周辺機器106__

50

1～106__Nを含む。

【0053】

動作状態検出器101は、電力制御装置PCA1が組み込まれている携帯情報機器を始めとする電気機器において実行されるアプリケーションプログラムの動作状態を検出する。つまり、動作状態検出器101は、アプリケーションプログラムが起動されたり、終了されたり、或いはアプリケーションプログラムの動作状態が変化したことを検知する。さらに、動作状態検出器101は、検出した動作状態を示す動作状態信号S1を生成する。

【0054】

電力供給率算出器102は、動作状態検出器101から入力される動作状態信号S1に基づいて、アプリケーションプログラムの動作状態の変化に応じて、電力制御装置PCA1が組み込まれている電気機器のハードウェアの各リソースに対する電力供給率を算出する。電力供給率とは、各リソースが持てる能力を完全に発揮して稼働するために必要とされる電力供給量を100%とした場合に、リソースに実際に提供される電力供給量の割合を%で表示したものを言う。本明細書中においては、リソースとしてハードウェアによって構成されるデバイスが例示されている、ソフトウェアによって構成されるデバイスも含まれることは言うまでもない。

【0055】

これは、本発明においては、アプリケーションプログラムからの要求を満たすために、リソースを常に100%稼働させる必要はないという点に鑑みての処理である。つまり、リソースに実際に要求される能力は、アプリケーションプログラムの動作状態により異なる。それゆえ、動作状態信号S1に基づいて、アプリケーションプログラムの動作状態に対応して、各リソースが可能な限り必要十分な能力を発揮できるように、各リソースにする電力供給率を適切に算出するものである。

【0056】

電力供給率算出器102は、動作状態信号S1で規定される動作状態に対する、各リソースに対する電力供給率が予め一義的に定められている場合は、その規定の供給率を示す電力供給率信号S4を生成して電力供給率割当器104に出力する。一方、動作状態信号S1で規定される動作状態に対する各リソースの電力供給率が一義的に定められていない場合には、動作状態信号S1に基づいて、電力供給率問い合わせ信号S2を生成して、電力供給率記憶器103に出力する。

【0057】

電力供給率記憶器103は、電力制御装置PCA1が組み込まれる電気機器のリソース毎に適正に予め定められた、アプリケーションプログラムの各動作状態における電力供給率の値を格納している。なお、電力供給率記憶器103に格納されている電力供給率の値は必要に応じて変更される。電力供給率記憶器103は、自身に格納されている電力供給率の中から電力供給率問い合わせ信号S2によって表されるアプリケーションプログラムの動作状態に対応する値を選出する。そして、電力供給率記憶器103は、選出した値を示す電力供給率信号S3を生成して、電力供給率算出器102に出力する。

【0058】

電力供給率算出器102は、電力供給率記憶器103から読み出したアプリケーションプログラムの動作状態毎に、各リソースに設定された複数の電力パラメータを総合して最終の電力パラメータを決定する。さらに、電力供給率算出器102はその電力パラメータを電力供給率割当器104に設定する。つまり、電力供給率算出器102は、個々のリソースに対して予め定められた電力供給率を表す電力供給率信号S3に基づいて、現在アプリケーションプログラムが稼働するように要求しているリソース全体に関して、個々のリソースに対する電力供給率を再計算する。そして、電力供給率算出器102は、再計算値を前述の電力供給率信号S4として、電力供給率割当器104に出力する。

【0059】

電力供給率割当器104は、CPU105およびN個の周辺機器106__1～106__Nに電力負荷に関する設定を行う。つまり、電力供給率割当器104は、電力供給率信号S

10

20

30

40

50

4に基づいて、稼働が要求されているリソースの個々に対して実際の電力供給率を割り当てる。さらに、電力供給率割当器104は、リソースであるCPU105に割り当てられた電力供給率を示す電力供給割当信号S5を生成して、CPU105に出力する。同様に、電力供給率割当器104は、N個の周辺機器106__1~106__Nのそれぞれに割り当てられた電力供給率を示す電力供給割当信号S6__1~S6__Nを生成して出力する。なお、電力供給率とは、リソースが具備する能力を100%発揮するのに必要な供給電力量に対する実際の供給電力量の割合を言う。

【0060】

なお上述のように、本明細書において、アプリケーションプログラムの動作状態に応じて、必要十分な性能を発揮するようにリソースを稼働させるために必要十分な電力供給率を求める本発明の基本概念について述べた。しかしながら、実際にリソース毎に異なる電力供給率で電力を分配供給するには、現在では巨大な電力分配供給装置が必要である。さらに、電力分配供給装置自身の電力消費量も無視できない。そのため、本実施形態においては、電力供給率の代わりに、各リソースの稼働率を算出する。

10

【0061】

稼働率とは、リソースが持てる能力を完全に発揮して稼働している状態を100%とした場合に、リソースが実際に稼働している割合を%で表示したものを言う。リソースの電力消費量と稼働率と比例関係にある。それゆえ、リソースの稼働率を変化させることによって、そのリソースの消費電力を変化させることができる。ゆえに、リソースの稼働率を調整することで、リソースへの電力供給率を調整するのと同じ効果が得られる。この意味において、稼働率はリソースの電力負荷を決定するパラメータである。

20

【0062】

つまり、本実施形態においては、より好ましくは、電力供給率算出器102は、動作状態検出器101から入力される動作状態信号S1に基づいて、アプリケーションプログラムの動作状態の変化に対応して、電力制御装置PCA1が組み込まれている電気機器のハードウェアの各リソースから必要十分な能力を引き出すのに適正な稼働率を算出する。

【0063】

電力供給率算出器102は、動作状態信号S1で規定される動作状態における各リソースに対する稼働率が予め一義的に規定されている場合は、その規定の稼働率を示す稼働率信号S4を生成して電力供給率割当器104に出力する。一方、動作状態信号S1で規定される動作状態に対する各リソースの稼働率が一義的に定められていない場合には、動作状態信号S1に基づいて、稼働率問い合わせ信号S2を生成して、電力供給率記憶器103に出力する。

30

【0064】

電力供給率記憶器103は、電力制御装置PCA1が組み込まれる電気機器のリソース毎に適正に予め定められたアプリケーションプログラムの各動作状態における電力供給率に対応する稼働率を格納している。なお、電力供給率記憶器103に格納されている稼働率の値は、必要に応じて変更される。電力供給率記憶器103は、自身に格納されている稼働率の中から、稼働率問い合わせ信号S2によって表されるアプリケーションプログラムの動作状態に対応する値を選出する。そして、電力供給率記憶器103は、選出した値を示す選出稼働率信号S3を生成して、電力供給率算出器102に出力する。

40

【0065】

電力供給率算出器102は、電力供給率記憶器103から読み出したアプリケーションプログラムの動作状態毎に、各リソースに設定された複数の電力パラメータを総合して最終の電力パラメータを決定する。さらに、電力供給率算出器102はその電力パラメータを電力供給率割当器104に設定する。つまり、電力供給率算出器102は、個々のリソースに対して予め定められた稼働率を表す稼働率信号S3に基づいて、現在アプリケーションプログラムが稼働するように要求しているリソースの全体に関して、個々のリソースの稼働率を再計算する。そして、電力供給率算出器102は、再計算値を前述の稼働率信号S4として、電力供給率割当器104に出力する。

50

【0066】

電力供給率割当器104は、CPU105およびN個の周辺機器106__1～106__Nに電力負荷のパラメータに関する設定を行う。つまり、電力供給率割当器104は、稼働率信号S4に基づいて、稼働が要求されているリソースの個々に対して実際の稼働率を割り当てる。さらに、電力供給率割当器104は、リソースであるCPU105に割り当てられた稼働率を示す稼働率割当信号S5を生成して、CPU105に出力する。同様に、電力供給率割当器104は、N個の周辺機器106__1～106__Nのそれぞれに割り当てられた稼働率を示す稼働率割当信号S6__1～S6__Nを生成して出力する。

【0067】

電力供給率割当器104から入力される稼働率割当信号S5は、CPU105の動作設定部に入力されて、CPU105の稼働率が稼働率割当信号S5に指定された値に設定される。結果、CPU105は、指定の稼働率で稼働することによって、その電力消費が低減された状態で、アプリケーションプログラムの要求に対して必要十分な能力を提供する。同様に、周辺機器106__1～106__Nの稼働率も、稼働率割当信号S6__1～S6__Nのそれぞれによって割り当てられた稼働率に設定される。そして、周辺機器106__1～106__Nは割り当てられた稼働率で稼働することにより、その電力消費が低減された状態で、アプリケーションプログラムの要求に対して必要十分な能力を提供する。

【0068】

図2を参照して、動作状態信号S1の具体例について説明する。同図において、「動作」と表示されている左端の列は、電力制御装置PCA1が組み込まれている電気機器のシステム、オペレーションシステム(OS)、およびアプリケーションプログラムの動作を区別して表している。「動作」としては、「システム」「アプリケーションプログラム1」「アプリケーションプログラム2」および「アプリケーションプログラム3」に分類されている。

【0069】

「状態」と表示されている中央の列は、前述の各「動作」における具体的な動作状態を細かく分類表示している。さらに「S1」と表示されている右端の列は、図1を参照して説明した動作状態信号S1をさらに細分化した信号を示している。

【0070】

つまり、システムが稼働している時(図2において、「動作」が「システム」の場合)は、その動作状態は大別して、「システム起動処理開始」、「システム起動処理完了」、「アイドル状態開始」、「アイドル状態終了」、「サスペンドモードに移行」、「サスペンドモードから復帰」、「システム終了処理開始」、および「システム終了処理完了」の8つの状態に分類されている。これら8つの状態をそれぞれ識別するために、動作状態信号S1のS1の後に__S1～__S8の接尾辞を付与している。

【0071】

つまり、システム起動処理開始状態は、動作状態信号S1__S1によって表される。システム起動処理完了状態は動作状態信号S1__S2によって表される。アイドル状態開始状態は、動作状態信号S1__S3によって表される。アイドル状態終了状態は、動作状態信号S1__S4によって表される。サスペンドモードに移行状態は、動作状態信号S1__S5によって表される。サスペンドモードから復帰状態は、動作状態信号S1__S6によって表される。システム終了処理開始状態は、動作状態信号S1__S7によって表される。そして、システム終了処理完了状態、は動作状態信号S1__S8によって表される。

【0072】

同様に、ビデオの表示録画を実現するアプリケーションプログラム1が稼働している時(図2において、「動作」が「アプリケーションプログラム1」の場合)は、その動作状態は大別して、起動処理開始状態S1__A1__1、起動処理完了状態S1__A1__2、画像表示開始状態S1__A1__3、画像表示終了状態S1__A1__4、録画処理開始状態S1__A1__5、録画処理終了状態S1__A1__6、終了処理開始状態S1__A1__7、および終了処理完了状態S1__A1__8の8つの状態に分類されている。

【0073】

さらに、同様に、ワードプロセッサを実現するアプリケーションプログラム2が稼働している時（図2において、「動作」が「アプリケーションプログラム2」の場合）は、その動作状態は大別して、起動処理開始状態S1__A2__1、起動処理完了状態S1__A2__2、表示開始状態S1__A2__3、表示終了状態S1__A2__4、入力処理開始（キー入力検知）状態S1__A2__5、入力処理終了（最後のキー入力より所定時間経過）S1__A2__6、文書保存処理開始S1__A2__7、文書保存処理完了S1__A2__8、終了処理開始状態S1__A2__9、および終了処理完了S1__A2__10の10の状態を表される。

【0074】

また、アプリケーションプログラム3が稼働している時（図2において、「動作」が「アプリケーションプログラム3」の場合）は、その動作状態は大別して、起動処理開始状態S1__A3__1、起動処理完了状態S1__A3__2、表示開始状態S1__A3__3、表示終了状態S1__A3__4、終了処理開始状態S1__A3__5、および終了処理完了S1__A3__6の6つの状態を表される。

【0075】

図3を参照して、電力制御装置PCA1を組み込んだ電気機器の動作中に、動作状態検出器101によって検出される状態と出力される動作状態信号S1の内容について具体的に説明する。

【0076】

まず、図3に「動作」が「システム」と表示されているシステムの動作状態に関して述べる。システムが起動中とは、動作状態検出器101が図2に示した動作状態信号S1__S1を出力した以降、次に動作状態信号S1__S2を出力するまでの間である。システムがアイドル中とは、動作状態信号S1__S3が出力されてから、動作状態信号S1__S4が出力されるまでの間である。サスペンド中とは、動作状態信号S1__S5が出力されてから、動作状態信号S1__S6が出力されるまでの間である。終了処理中とは、動作状態信号S1__S7が出力されてから、動作状態信号S1__S8が出力されるまでの間である。そして、システムの終了とは、動作状態信号S1__S8が出力された後である。

【0077】

次に、図3に「動作」が「アプリケーションプログラム1」と表示されているアプリケーションプログラム1（ビデオの表示および録画）の動作状態に関して述べる。起動中とは、動作状態検出器101が図2に示した動作状態信号S1__A1__1を出力してから、次に動作状態信号S1__A1__2を出力するまでの間である。画像表示中とは、動作状態信号S1__A1__3が出力されてから、動作状態信号S1__A1__4が出力されるまでの間である。録画動作中とは、動作状態信号S1__A1__5が出力されてから、動作状態信号S1__A1__6が出力されるまでの間である。終了処理中とは、動作状態信号S1__A1__7が出力されてから、動作状態信号S1__A1__8が出力されるまでの間である。

【0078】

さらに、図3に「アプリケーションプログラム2」と表示されているアプリケーションプログラム2（ワードプロセッサ）の動作状態に関して述べる。起動中とは、動作状態検出器101が図2に示した動作状態信号S1__A2__1を出力してから、次に動作状態信号S1__A2__2が出力されるまでの間である。表示中とは、動作状態信号S1__A2__3が出力されてから、動作状態信号S1__A2__4が出力されるまでの間である。入力中とは、動作状態信号S1__A2__5が出力されてから、動作状態信号S1__A2__6が出力されるまでの間である。保存中とは、動作状態信号S1__A2__7が出力されてから、動作状態信号S1__A2__8が出力されるまでの間である。終了処理中とは、動作状態信号S1__A2__9が出力されてから、動作状態信号S1__A2__10が出力されるまでの間である。

【0079】

そして、図3に「動作」が「アプリケーションプログラム3」と表示されているアプリケ

10

20

30

40

50

ーションプログラム 3（時計表示）の動作状態に関して述べる。起動中とは、動作状態検出器 101 が図 2 に示した動作状態信号 S 1 __ A 3 __ 1 を出力してから、次に動作状態信号 S 1 __ A 3 __ 2 を出力するまでの間である。表示中とは、動作状態信号 S 1 __ A 3 __ 3 が出力されてから、動作状態信号 S 1 __ A 3 __ 4 が出力されるまでの間である。終了処理中とは、動作状態信号 S 1 __ A 3 __ 5 が出力されてから、動作状態信号 S 1 __ A 3 __ 6 が出力されるまでの間である。

【0080】

なお、上述の図 2 および図 3 を参照して、図 1 に示した電力制御装置 P C A 1 を組み込んだ電気機器の動作状態と、電力制御装置 P C A 1 の動作状態検出について以下に簡単に述べる。動作状態検出器 101 は、アプリケーションプログラム 1 の起動処理の開始を検知すると、先ず動作状態信号 S 1 __ A 1 __ 1 を電力供給率算出器 102 に出力する。そして、アプリケーションプログラム 1 の起動処理が完了した時点で、動作状態検出器 101 は動作状態信号 S 1 __ A 1 __ 2 を電力供給率算出器 102 に出力する。図 3 に示した動作状態の定義に基づいて、電力供給率算出器 102 は、動作状態信号 S 1 __ A 1 __ 1 を受信してから、動作状態信号 S 1 __ A 1 __ 2 までの期間は、アプリケーションプログラム 1 が「起動中」であると認識する。同様に、アプリケーションプログラム 1 が「画像表示中」および「録画動作中」および「終了処理中」である状態を認識する。

【0081】

同様に動作状態信号に基づいて、電力供給率算出器 102 は、アプリケーションプログラム 2 が「起動中」または「表示中」または「入力中」または「保存中」または「終了処理中」のどの状態かを正確に認識する。アプリケーションプログラム 3 に対しても同様に、電力供給率算出器 102 は「起動中」または「表示中」または「終了処理中」のどの状態かを認識できる。

【0082】

次に図 4 に示すフローチャートを参照して、電力制御装置 P C A 1 の動作について説明する。電力制御装置 P C A 1 が組み込まれた電気機器に電源が投入されると、電力制御装置 P C A 1 はその電力制御動作を開始する。

先ず、ステップ # 100 において、動作状態検出器 101 により、電気機器の動作状態を検出する。そして、図 2 に例示した状態毎に検出した動作状態を示す動作状態信号 S 1 を生成して、電力供給率算出器 102 に出力する。そして、処理は、検出した動作状態に応じた電力パラメータを取得するサブルーチンであるステップ # 200 に進む。

【0083】

ステップ # 200 において、稼働率信号 S 3 において得られた動作状態信号 S 1 に基づいて検出された状態における各リソースに対する稼働率が予め一義的に規定されている場合は、規定の稼働率を示す稼働率信号 S 4 が生成されて電力供給率割当器 104 に出力する。一方、動作状態信号 S 1 で規定される動作状態に対する各リソースの稼働率が一義的に定められていない場合には、稼働率問い合わせ信号 S 2 を電力供給率記憶器 103 に出力する。電力供給率記憶器 103 は、稼働率問い合わせ信号 S 2 によって表されるアプリケーションプログラムの動作状態に対応する値を示す選出稼働率信号 S 3 を生成して、電力供給率算出器 102 に出力する。そして、処理は、所得された電力パラメータ毎に、各リソースに設定された複数の電力パラメータを総合して、最終の電力パラメータを決定する電力供給率算出サブルーチンであるステップ # 300 に進む。

【0084】

ステップ # 300 において、電力供給率算出器 102 は、個々のリソースに対して予め定められた稼働率を表す選出稼働率信号 S 3 に基づいて、現在アプリケーションプログラムが稼働するように要求しているリソースの全体に関して、個々のリソースに割り当てる稼働率を再計算する。そして、電力供給率算出器 102 は、再計算値を前述の稼働率信号 S 4 として電力供給率割当器 104 に出力する。そして、処理は、電力供給率を割り当てるために、各リソースに再計算された稼働率を割り当てるサブルーチンであるステップ # 400 に進む。

【0085】

ステップ#400において、#300で再計算された稼働率を示す稼働率割当信号S5、およびS6__1～S6__NをCPU105およびN個の周辺機器106__1～106__Nのそれぞれに対して発送する。CPU105および周辺機器106__1～106__Nは、それぞれ電力供給率割当器104から発送された稼働率割当信号S5、およびS6__1～S6__Nに基づいて、自身の稼働率を設定し直すことにより、それぞれの消費電力量が制御される。そして、処理は、次のステップ#500に進む。

【0086】

ステップ#500においては、電力制御装置PCA1が組み込まれている電気機器のシステムが終了しているか否かが、前述の動作状態信号S1に基づいて判断される。Noの場合は、処理はステップ#100に戻り、前述のステップ#200、#400、および#500の処理を繰り返し、ステップ#500でYesと判断された時点で処理を終了する。なお動作状態検出器101は、上述のように、ステップ#100で電気機器の動作状態を逐次検出して動作状態信号S1を出力する代わりに、動作状態に変化がある場合にのみ、動作状態信号S1を出力するようにしても良い。

10

【0087】

次に、図5、図6、図7、および図8を参照して、電力供給率記憶器103に格納されるシステムおよびアプリケーションプログラム1からアプリケーションプログラム3までのそれぞれの動作状態において、各リソースに対して設定されている電力パラメータである稼働率の一例について説明する。なお、これらの図面中において、稼働率はシステムの状態やアプリケーションプログラムの状態に応じて、リソースであるCPU（中央演算装置）、HDD（ハードディスクドライブ）、LCD（液晶教示装置）、メモリ、およびVRC（録画回路）の各デバイスが、それらに固有に設定されている稼働率（電力パラメータ値）と共に表示されている。なお、図1に示す構成で言えば、CPUがCPU105に、HDD、LCD、メモリ、およびVRCが周辺機器106__1～106__4に対応する。各数値は、%（パーセント）で表示され0から100までの数値が設定されている。0はそのデバイスの動作を停止させることを意味し、100はそのデバイスを100%動作させることを意味する。

20

【0088】

図5に、図1に示した電力供給率記憶器103に格納されているシステム全体の稼働率（電力パラメータ）の設定値の一例を示す。システム起動中や、アイドル中、サスペンド中、システムの終了処理中、および終了状態の時の稼働率（電力パラメータ）を設定した場合の例が示されている。この例では、起動中には、CPU、HDDおよびメモリの稼働率（電力パラメータ）を100%とすることにより、システムの高速起動を図っている。一方、LCDは起動中は高輝度である必要はないので、稼働率を25%に設定している。

30

【0089】

アプリケーションプログラムが何も動作していないアイドル中は、CPUの稼働率を10%とし、HDDの稼働率を0%、すなわち停止状態に設定する。LCDは、起動時と同様に高輝度である必要はないので、稼働率を25%に設定する。メモリの稼働率は100%に設定される。

40

【0090】

サスペンド中は、CPU、HDD、およびLCDの稼働率を0%に設定して、それらを停止状態にする。一方、メモリの稼働率は、10%に設定して、サスペンド前の情報を保持する。

【0091】

終了処理中は、CPUおよびHDDの稼働率を50%に設定すると共に、LCDの稼働率を0%に設定する。一方、メモリの稼働率を100%に設定する。

【0092】

終了時は、CPU、HDD、LCD、メモリ、およびVRC)のすべての稼働率を0%に設定する。なお、VRCは起動中、アイドル中、サスペンド中、および終了処理中も、稼

50

働率は0%に設定される。

【0093】

次に、図6に、アプリケーションプログラム1の各動作状態に対応して設定された稼働率（電力パラメータ）の割当の一例を示す。本例においては、アプリケーションプログラム1は、例として、ビデオ表示や録画を行う。

【0094】

CPUに関しては、起動中および終了処理中の動作には、それほど演算能力を必要とされないため、その稼働率を50%に設定している。一方、録画中には、かなりの演算能力が必要とされるために、CPUの稼働率を80%に設定している。また、単に画像を表示している状態の時には、CPUの演算能力が必要とされないため、CPUの稼働率を10%に設定している。

10

【0095】

HDDに関しては、起動中と終了処理中には、かなりの量のデータを入出力するので、その稼働率を50%に設定している。一方、録画動作中には、HDDは大量のデータを高速度に入出力するので、その稼働率を100%に設定している。表示中には、HDDの稼働率を0%に設定して、HDDを停止状態にする。

【0096】

LCDに関しては、起動中および終了処理中は、中程度の輝度で十分であるためその稼働率を50%に設定している。そして、画像表示中と録画動作中は輝度を上げて見えやすくするために、LCDの稼働率を100%に設定している。

20

【0097】

メモリに関しては、すべての動作状態において、稼働率を100%に設定して、データの高速な入出力に備えている。

【0098】

VRCに関しては、録画動作中には100%の稼働率が、終了動作中に50%の稼働率が必要と設定している。しかし、その他の状態の時には、稼働率を0%に設定している。

【0099】

図7に、アプリケーションプログラム2の各動作状態に対応して設定された稼働率（電力パラメータ）の一例を示す。本例においては、アプリケーションプログラム2は、ワードプロセッサである。プログラムの起動中および終了処理中は、図6を参照して説明したのと同様の理由により、CPUおよびHDDの稼働率は共に50%に設定されている。さらに、表示中、入力中、および保存中の状態においては、CPUの稼働率は、それぞれ、10%、50%、および50%に設定されている。一方、HDDの稼働率は、0%、20%、および100%に設定されている。

30

【0100】

これらの稼働率は、以下の点を考慮して決められている。つまり、表示中には、CPUおよびHDDは、図6で示したアイドル時と同様に、それほど能力が必要とされない。また表示中は、ユーザが思考中と見なし、単に画面に文書を表示するためにはCPUパワーをそれほど必要としない。ユーザがキーボード等から入力中は、仮名漢字変換や辞書の読み込みなどのために、CPUおよびHDDに対して若干のサービスが必要とされる。保存動作に関しては、HDDにデータを高速に保存する必要がある。

40

【0101】

なお、LCD、メモリ、およびVRCの稼働率は、表示中、入力中、保存中、および終了処理中のすべての状態において、それぞれ、50%、100%、および0%に固定的に設定されている。つまり、LCDに関しては、ある程度の輝度があればワードプロセッサとして使用できるという考えである。

【0102】

図8に、アプリケーションプログラム3の各動作状態に対応して設定された稼働率（電力パラメータ）の一例を示す。本例においては、アプリケーションプログラム3は、時計表示として動作する。起動中時には、CPUとHDDのサービスがある程度必要であるので

50

、それぞれの稼働率を10%と20%に設定している。表示中はHDDのサービスを必要とせず、CPUも最低限のサービスで良いので、それぞれ10%と0%が稼働率として設定されている。また、終了処理中は、CPUおよびHDDとも、かなりのサービスが必要とされるので、それぞれの稼働率は共に50%に設定されている。

【0103】

一方、LCD、メモリ、およびVRCの稼働率は、起動中、表示中、および終了処理中の何れに対しても、それぞれ25%、100%、および0%に固定的に設定されている。

【0104】

図9を参照して、複数のアプリケーションプログラムが動作している時に図6、図7、および図8に例示したアプリケーションプログラムの動作状態に対応した個々の電力パラメータ（稼働率）からシステム全体としての各デバイス（リソース）の電力パラメータ（稼働率）の値を計算する方法の一例について説明する。CPU、HDD、メモリ、およびVRCは、動作中のアプリケーションプログラムの動作状態に対応した各電力パラメータ（稼働率）を合計し、その合計値が100%を超える場合は、システム全体として、各リソース毎の電力パラメータ（稼働率）を100%に決定する。

【0105】

LCDに関しては、動作中のアプリケーションプログラムの状態に対応した各電力パラメータ（稼働率）のうちの最大値を、システム全体として、LCDの電力パラメータ（稼働率）とする。

【0106】

図10を参照して、アプリケーションプログラム1が画像表示中で、そして、アプリケーションプログラム2が入力中の場合の電力パラメータ（稼働率）の計算について説明する。なお本例において、まず、アプリケーションプログラム1が画像表示中の状態であることより、図6の画像表示中の行を選択する。結果、CPU、HDD、LCD、メモリ、およびVRCの稼働率は、それぞれ、10%、0%、100%、100%、および0%に設定されていることが分かる。

【0107】

次に、アプリケーションプログラム2が、入力中の状態であることより、図7の入力中の行を選択する。結果、CPU、HDD、LCD、メモリ、およびVRCの稼働率は、それぞれ、50%、20%、50%、100%、および0%に設定されていることが分かる。このようにして、複数のリソースのそれぞれに対しても、アプリケーションプログラムの動作状態に応じて電力パラメータである稼働率を設定することによって、電気機器のリソースに対して二次元的に電力制御を行うことができる。

【0108】

さらに、図9を参照して説明した計算方法に基づいて、各デバイス（リソース）に対するシステム全体としての稼働率（電力パラメータ）を算出する。CPUの稼働率は10%に50%を加えて60%となる。また、HDDの稼働率は、0%に20%を加え20%となる。LCDの稼働率は、100%と50%のうちの最大値を選択し、100%となる。メモリの稼働率は、100%に100%を加え200%となり、100%を超えるため100%とする。VRCの稼働率は、0%に0%を加えて、0%となる。結果、システム全体としてのCPU、HDD、LCD、メモリ、およびVRCの稼働率は、図10の最下行に示すように、それぞれ、60%、20%、100%、100%、および0%に設定される。

【0109】

図11を参照して、アプリケーションプログラム1が録画動作中、アプリケーションプログラム2が表示中、かつアプリケーションプログラム3が表示中の場合のシステム全体としての各デバイス（リソース）の稼働率（電力パラメータ）の計算について説明する。まず、アプリケーションプログラム1が録画動作中であることより、図6の録画動作中の行を選択する。結果、CPU、HDD、LCD、メモリ、およびVRCの稼働率は、それぞれ、80%、100%、100%、100%、および100%に設定されていることが分

かる。

【0110】

次に、アプリケーションプログラム2が表示中であることより、図7の表示中の行を選択する。結果、CPU、HDD、LCD、メモリ、およびVRCの稼働率は、それぞれ、10%、0%、50%、100%、および0%に設定されていることが分かる。

【0111】

さらに、アプリケーションプログラム3が表示中であることより、図8の表示中の行を選択する。結果、CPU、HDD、LCD、メモリ、およびVRCの稼働率は、それぞれ、10%、0%、25%、100%、および0%に設定されていることが分かる。

【0112】

図6、図7、および図8の上述の各行において、デバイス（リソース）毎に取得した稼働率（電力パラメータ）に対して、図9を参照して説明した計算方法を適用し、システム全体に対する各デバイスの稼働率を算出する。つまり、CPUの電力パラメータである稼働率は、80%に10%と10%を加え100%となる。また、HDDの稼働率は、100%に0%と0%を加え100%となる。LCDの稼働率は、100%と50%と25%のうちの最大値をとり100%となる。メモリの稼働率は、100%に100%と100%を足すことにより、100%以上となるので、100%に設定される。VRCの稼働率は、100%に0%と0%を足すことにより100%となる。結果、システム全体としてのCPU、HDD、LCD、メモリ、およびVRCの稼働率は、図11の最下行に示すように、すべて100%に設定される。

【0113】

なお、以上の説明では、電力供給率記憶器103に記憶する電力パラメータ（稼働率）を各アプリケーションプログラム毎に、図4、図5、および図6に例示したように設定している。しかしながら、同一のアプリケーションプログラムに、電力パラメータ（稼働率）の複数のパターンを用意することにより、同じアプリケーションプログラムでも、用途に応じて異なる電力パラメータ（稼働率）を設定することができる。

【0114】

例えば、図7に示したアプリケーションプログラム2によって実現されるワードプロセッサの場合に、文書入力用の電力パラメータ（稼働率）と文書校正用の電力パラメータ（稼働率）の2種類のパターンを用意しておく。そして、アプリケーションプログラムの起動時に、ユーザが2種類のパターンのうちの何れかを選択することにより、さらに木目細やかな電力制御を行うことができる。なお、上述の稼働率は、一例であって、各デバイス毎に調整可能な分解能に応じて、任意の値に設定してよい。

【0115】

また、電力パラメータ（稼働率）は、予め電力供給率記憶器103に記憶しておく代わりに、ユーザがアプリケーションプログラム毎に、任意に設定しても良いことは言うまでもない。

【0116】

（第2の実施形態）

図12に、本発明の第2の実施形態にかかる電力制御装置の構造を示す。本例における電力制御装置PCA2は、図1～図11を参照して説明した第1の実施形態にかかる電力制御装置PCA1を、パーソナルコンピュータ等に代表されるオペレーションシステムを有する電気機器（図示せず）に適応したものである。電力制御装置PCA2は、電力制御装置PCA1を構成する動作状態検出器101、電力供給率算出器102、電力供給率記憶器103、および電力供給率割当器104を含む。

【0117】

なお、電力制御装置PCA2において、M個（Mは正の整数）のアプリケーションプログラム1000_1からアプリケーションプログラム1000_M）はOS上で動作しているアプリケーションプログラムであり、OSのサービスであるAPIを利用して動作している。なお、同図において、M個のアプリケーションプログラム1000_1～1000

__Mはそれぞれ、A P G 1～A P G Mとして表現されている。

【0118】

本例においては、O SはA P I層1005、カーネル層1006、およびドライバインターフェース層1007の3階層からなる。A P I層1005は、アプリケーションプログラムとシステムとの間を取り持つインターフェースである。カーネル層1006は、スケジューリングなどのO Sの基本制御を行う。ドライバインターフェース層1007は、各種ドライバとのインターフェースである。

【0119】

ドライバインターフェース層1007の下位には、N個のデバイスドライバがある。これらN個のドライバの一例としては、キーボードドライバ1100__1、ビデオ表示・録画ドライバ1100__2、サウンドドライバ1100__3、3D機能ドライバ1100__4、モデムドライバ1100__5、I r D Aドライバ1100__6、I E E E 1394インターフェースドライバ1100__7、U S Bドライバ1100__8、およびC P U動作制御ドライバ1100__Nがある。

10

【0120】

上述のN個のデバイスドライバには、対応するN個のデバイスが接続されている。これらN個のデバイスには、キーボード1200__1、ビデオ表示・録画器（図中では「V．D．R」と略称）1200__2、サウンド処理器（図中では「サウンド」と略称）1200__3、3D機能処理器（図中では「3D-F」と略称）1200__4、モデム1200__5、I r D A 1200__6、I E E E 1394インターフェース（図中では「I E E E 1394」と略称）1200__7、U S B 1200__8、およびC P U動作制御器1200__Nがある。上述のドライバ1100__1～1100__Nのそれぞれは、対応するデバイス1200__1～1200__Nの制御を、制御信号I F__1～I F__Nにより行う。

20

【0121】

ドライバインターフェース層1007と、ドライバ1100__1～1100__Nのそれぞれの間では、稼働率設定信号S V__1からS V__Nおよびデバイス出力信号E D__1～E D__Nが交換される。稼働率設定信号S V__1～S V__Nは、デバイス（リソース）毎の稼働率（電力パラメータ）設定を始めとする各種設定および、データを含む信号である。デバイス出力信号E D__1～E D__Nは、デバイス1200__1～1200__Nのそれぞれの動作状態および、それらデバイスから出力されるデータを含む信号である。

30

【0122】

カーネル層1006は、システム起動時、アイドル時、サスペンド時、およびシステム終了時等の動作状態を示す動作状態監視信号M S 1を生成する。なお、動作状態監視信号M S 1の示す状態については、後ほど図30を参照して説明する。

【0123】

A P I層1005は、アプリケーションプログラム1000__1～1000__Mのそれぞれに対して、キーボードイベントなどの情報や通信データなどを通知する通知信号E A__1～E A__Mを出力する。一方、アプリケーションプログラム1000__1～1000__Mのそれぞれは、A P I層1005に対して、システムに接続されているリソースを活用するために出力するアプリケーション信号A P__1～A P__Mを生成する。

40

【0124】

さらに、A P I層1005は、上述の通知信号E A__1～E A__Mとアプリケーション信号A P__1～A P__Mを監視し、アプリケーションプログラム毎の動作状態を検知して、検知した動作状態を通知するA P I通信情報M S 2を生成する。なお、A P I層1005が検知する動作状態については、後ほど図31を参照して説明する。

【0125】

動作状態検出器101は、A P I層1005から入力されるA P I通信情報M S 2と、カーネル層1006から入力される動作状態監視信号M S 1とに基づいて動作状態信号S 1信号を生成する。なお、電力供給率割当器104は、第1の実施形態にかかる電力制御装置P C A 1における稼働率割当信号S 5およびS 6が示す情報を同時に含む稼働率割当信

50

号 SVA1 を生成する。

【0126】

ドライバインターフェース層 1007 は、電力供給率割当器 104 から入力される稼働率割当信号 SVA1 を、各デバイス（リソース）に対する稼働率を示す上述の稼働率設定信号 SV__1 ～ SV__N を生成して、対応するドライバ 1100__1 ～ 1100__N に出力する。そして、ドライバ 1100__1 ～ 1100__N は、稼働率設定信号 SV__1 ～ SV__N に基づいて、制御信号 IF__1 ～ IF__N を生成して、対応するデバイス 1200__1 ～ 1200__N の稼働率を変更する。

【0127】

図 30 を参照して、カーネル層 1006 が出力する動作状態監視信号 MS1 の内容について説明する。カーネル層 1006 が、システム起動処理開始を検知した時には、信号 MSS1 が動作状態監視信号 MS1 として動作状態検出器 101 に送信される。カーネル層 1006 が、システム起動処理完了を検知した時には、信号 MSS2 が動作状態監視信号 MS1 として動作状態検出器 101 に送信される。同様に、カーネル層 1006 が、アイドル状態開始、アイドル状態終了、サスペンド状態への移行、サスペンド状態からの復帰、システム終了処理開始、およびシステム終了処理完了のそれぞれの状態を検知した時には、それぞれ信号 MSS4、MSS5、MSS6、MSS7、および MSS8 が動作状態監視信号 MS1 として動作状態検出器 101 に出力される。

10

【0128】

なお、電力制御装置 PCA2 において、動作状態検出器 101 は、カーネル層 1006 から受信した動作状態監視信号 MS1 と API 層 1005 から受信した API 通信情報 MS2 に基づいて、所定の定義に従って、動作状態信号 S1 を生成する。なお、この動作状態信号 S1 の生成の定義については、後ほど図 32 を参照して説明する。

20

【0129】

なお、電力供給率割当器 104 の出力である稼働率割当信号 SVA1 は、電力供給率割当器 104 に設定された各デバイスの電力パラメータ（稼働率）を、ドライバインターフェース層 1007 に伝達するものである。伝達された信号は、ドライバインターフェース層 1007 により稼働率設定信号 SV__1 ～ SV__N として各デバイス 1200__1 ～ 1200__N 用のドライバ 1100__1 から 1100__N のそれぞれに割り振られる。各ドライバ 1100__1 から 1100__N は、信号 IF__1 から信号 IF__N を用いて、デバイス 1200__1 ～ 1200__N の各々に電力パラメータ（稼働率）の設定を行う。

30

【0130】

電力制御装置 PCA2 の CPU 動作設定部 1100__N は、CPU 動作制御ドライバ 1100__N からの信号 IF__N により、電力パラメータ（稼働率）を受け取る点が、電力制御装置 PCA1 における CPU 105 と異なる。

【0131】

次に、図 31 を参照して、API 層 1005 が出力する API 通信情報 MS2 の内容について説明する。API 層 1005 がアプリケーションプログラム 1000__1 ～ 1000__M と OS との間に交わされる通信（アプリケーション信号 AP__1 ～ AP__M および通知信号 EA__1 から EA__M）を監視することによって、検知したアプリケーションプログラム毎の動作状態に対応して、一例として 34 種類の信号 MSA1 から MSA34 が生成される。そして API 層 1005 は、これらの信号 MSA1 から MSA34 のそれぞれを、API 通信情報 MS2 として動作状態検出器 101 に出力する。

40

【0132】

つまり、図 31 に例示されているように、起動処理開始 API の検知に対しては、信号 MSA1 が API 通信情報 MS2 として出力される。

【0133】

起動処理完了 API の検知に対しては、信号 MSA2 が API 通信情報 MS2 として出力される。

【0134】

50

終了処理開始 A P I の検知に対しては、信号 M S A 3 が A P I 通信情報 M S 2 として出力される。

【0135】

終了処理完了 A P I の検知に対しては、信号 M S A 4 が A P I 通信情報 M S 2 として出力される。

【0136】

表示開始 A P I の検知に対しては、信号 M S A 5 が A P I 通信情報 M S 2 として出力される。

【0137】

表示終了 A P I の検知に対しては、信号 M S A 6 が A P I 通信情報 M S 2 として出力される。 10

【0138】

録画処理開始 A P I の検知に対しては、信号 M S A 7 が A P I 通信情報 M S 2 として出力される。

【0139】

録画処理終了 A P I の検知に対しては、信号 M S A 8 が A P I 通信情報 M S 2 として出力される。

【0140】

キー入力 A P I の検知に対しては、信号 M S A 9 が A P I 通信情報 M S 2 として出力される。 20

【0141】

キー入力 A P I 検知後所定時間経過の検知に対しては、信号 M S A 10 が A P I 通信情報 M S 2 として出力される。

【0142】

保存処理開始 A P I の検知に対しては、信号 M S A 11 が A P I 通信情報 M S 2 として出力される。

【0143】

保存処理完了 A P I の検知に対しては、信号 M S A 12 が A P I 通信情報 M S 2 として出力される。

【0144】

モデムオープン A P I の検知に対しては、信号 M S A 13 が A P I 通信情報 M S 2 として出力される。 30

【0145】

モデムクローズ A P I の検知に対しては、信号 M S A 14 が A P I 通信情報 M S 2 として出力される。

【0146】

モデム通信開始 A P I の検知に対しては、信号 M S A 15 が A P I 通信情報 M S 2 として出力される。

【0147】

モデム通信終了 A P I の検知に対しては、信号 M S A 16 が A P I 通信情報 M S 2 として出力される。 40

【0148】

I r D A オープン A P I の検知に対しては、信号 M S A 17 が A P I 通信情報 M S 2 として出力される。

【0149】

I r D A クローズ A P I の検知に対しては、信号 M S A 18 が A P I 通信情報 M S 2 として出力される。

【0150】

I r D A 通信開始 A P I の検知に対しては、信号 M S A 19 が A P I 通信情報 M S 2 として出力される。 50

【0151】

I r D A通信終了A P Iの検知に対しては、信号M S A 2 0がA P I通信情報M S 2として出力される。

【0152】

I E E E 1 3 9 4 インターフェースオープンA P Iの検知に対しては、信号M S A 2 1がA P I通信情報M S 2として出力される。

【0153】

I E E E 1 3 9 4 インターフェースクローズA P Iの検知に対しては、信号M S A 2 2がA P I通信情報M S 2として出力される。

【0154】

I E E E 1 3 9 4 インターフェース通信開始A P Iの検知に対しては、信号M S A 2 3がA P I通信情報M S 2として出力される。

【0155】

I E E E 1 3 9 4 インターフェース通信終了A P Iの検知に対しては、信号M S A 2 4がA P I通信情報M S 2として出力される。

【0156】

U S BオープンA P Iの検知に対しては、信号M S A 2 5がA P I通信情報M S 2として出力される。

【0157】

U S BクローズA P Iの検知に対しては信号M S A 2 6がA P I通信情報M S 2として出力される。

【0158】

U S B通信開始A P Iの検知に対しては、信号M S A 2 7がA P I通信情報M S 2として出力される。

【0159】

U S B通信終了A P Iの検知に対しては、信号M S A 2 8がA P I通信情報M S 2として出力される。

【0160】

3 D機能開始A P Iの検知に対しては、信号M S A 2 9がA P I通信情報M S 2として出力される。

【0161】

3 D機能終了A P Iの検知に対しては、信号M S A 3 0がA P I通信情報M S 2として出力される。

【0162】

音声再生開始A P Iの検知に対しては、信号M S A 3 1がA P I通信情報M S 2として出力される。

【0163】

音声再生終了A P Iの検知に対しては、信号M S A 3 2がA P I通信情報M S 2として出力される。

【0164】

ビデオ再生開始A P Iの検知に対しては、信号M S A 3 3がA P I通信情報M S 2として出力される。

【0165】

そして、ビデオ再生終了A P Iの検知に対しては、信号M S A 3 4がA P I通信情報M S 2として出力される。

【0166】

以下に、図13のフローチャートを参照して、電力制御装置P C A 2の動作について説明する。電力制御装置P C A 2が組み込まれている電気機器に電源が投入されて運転が開始される。

【0167】

10

20

30

40

50

まず、ステップS 1において、カーネル層1 0 0 6は、システム状態を監視する。そして、A P I層1 0 0 5は、アプリケーションプログラム1 0 0 0 __1～1 0 0 0 __MとO Sとの間に交わされる通信（アプリケーション信号A P __1～A P __Mおよび通知信号E A __1～E A __M）を監視する。そして、処理は、次のステップS 3に進む。

【0 1 6 8】

ステップS 3において、動作状態監視信号M S 1によってシステムの監視情報が、カーネル層1 0 0 6から動作状態検出器1 0 1に入力される。また、各アプリケーションプログラム1 0 0 0 __1から1 0 0 0 __MとO Sとの間に交わされるA P I通信情報は、A P I通信情報M S 2により、A P I層1 0 0 5から動作状態検出器1 0 1に入力される。つまり、図3 1を参照して説明した信号M S A 1～M S A 3 4のうちで該当するものがA P I通信情報M S 2として動作状態検出器1 0 1に伝達される。そして、処理は、次のステップS 5に進む。

10

【0 1 6 9】

ステップS 5において、動作状態検出器1 0 1が受け取った動作状態監視信号M S 1がシステムに関するものであるか否かが判断される。つまり、動作状態監視信号M S 1がシステムに関するものであると判断される場合には、処理はステップS 7に進む。一方、システムに関するものではない、つまりアプリケーションプログラムに関する判断される場合には、処理はステップS 1 5に進む。

【0 1 7 0】

ステップS 7において、電力供給率記憶器1 0 3に記憶されているシステム全体の稼働率パラメータを格納したテーブルから該当するシステム状態の時に設定すべき電力パラメータ（稼働率）を読み出す。なお、システム全体の電力パラメータ（稼働率）については、後ほど図1 4を参照して説明する。そして、処理は次のステップS 9に進む。

20

【0 1 7 1】

ステップS 9において、電力供給率算出器1 0 2が電力供給率割当器1 0 4に設定すべき電力パラメータ（稼働率）を算出する。そして、処理は次のステップS 1 1に進む。

【0 1 7 2】

ステップS 1 1において、電力供給率割当器1 0 4は、ステップS 9において算出された電力パラメータ（稼働率）を稼働率割当信号S V A 1として、ドライバインターフェース層1 0 0 7に出力する。そして、ドライバインターフェース層1 0 0 7は、稼働率割当信号S V A 1に基づいて、前述の稼働率設定信号S V __1～S V __Nを生成して、それぞれ対応するデバイスドライバ1 1 0 0 __1～1 1 0 0 __Nに出力する。デバイスドライバ1 1 0 0 __1～1 1 0 0 __Nは、稼働率設定信号S V __1～S V __Nに基づいて、対応するデバイス1 2 0 0 __1～1 2 0 0 __Nの稼働率をそれぞれ設定する。そして、処理は次のステップS 1 3に進む。

30

【0 1 7 3】

ステップS 1 3において、システムが終了しているか否かが判断される。システムが終了していない場合には、N oと判断されて、処理はステップS 1に戻る。また、システムが終了している場合は、Y e sと判断されて、処理が終了する。

【0 1 7 4】

40

ステップS 1 5において、電力供給率算出器1 0 2が、電力供給率記憶器1 0 3に記憶されている稼働率テーブルの中に現在着目しているアプリケーションプログラムに関する固有の電力パラメータ（稼働率）が存在するかどうか判断される。電力供給率記憶器1 0 3にアプリケーション固有の電力パラメータ（稼働率）が存在する場合には、Y e sと判断されて、処理は次のステップS 1 9に進む。アプリケーション固有の電力パラメータが存在しない場合には、N oと判断されて、処理はステップS 1 7に進む。

【0 1 7 5】

ステップS 1 7において、電力供給率算出器1 0 2が電力供給率記憶器1 0 3に格納されているアプリケーションプログラムに固有に設定させた電力パラメータ（稼働率）テーブルを参照して、電力パラメータ（稼働率）を読み出す。なお、本ステップにおいて、参照

50

される電力パラメータ（稼働率）テーブルに関しては、図 1 5 を参照して後ほど説明する。そして、処理はステップ S 9 に進む。

【0176】

ステップ S 1 9 においては、電力供給率記憶器 1 0 3 に格納されている標準の電力パラメータテーブルを参照して、検知したアプリケーションプログラムの状態に対応した電力パラメータ（稼働率）を取り出す。なお、本ステップで参照される電力パラメータテーブルについては、図 1 6 から図 2 2 を参照して後ほど説明する。そして、処理は上述のステップ S 9 に進む。

【0177】

以下に、図 1 4 ～図 2 2 を参照して、図 1 3 に示したフローチャートにおいて参照される電力パラメータ（稼働率）のそれぞれについて説明する。

10

【0178】

先ず、図 1 4 に、ステップ S 7 において参照されるシステム全体の電力パラメータの例を示す。システム起動中、アイドル中、サスペンド中、終了処理中、および終了状態の時の電力パラメータ（稼働率）が示されている。本例においては、システムの起動中には、CPU、HDD、およびメモリの電力パラメータ（稼働率）を 1 0 0 % に設定することにより、システムの高速起動を図っている。また、アプリケーションプログラムが何も動作していないアイドル中に関して、CPU の稼働率は 1 0 % に設定し、HDD の稼働率は 0 % すなわち停止状態に設定している。システムの起動中およびアイドル中には、LCD は高輝度の必要はないため、稼働率を 2 5 % に設定している。

20

【0179】

サスペンド中には、サスペンドに移行する前の動作状態を保存するために、メモリの稼働率を 1 0 % に設定し、メモリ以外のデバイスの稼働率を 0 % に設定することによって無駄な電力消費を防止する。

【0180】

終了処理中は、CPU および HDD の稼働率を 5 0 % し、LCD の稼働率は 0 % に設定し、メモリの稼働率は 1 0 0 % に設定する。

【0181】

終了時には、当然、CPU、HDD、LCD、およびメモリのすべての稼働率を 0 % に設定する。

30

【0182】

上述のように、固有の電力パラメータが指定されているアプリケーションプログラムが動作している時は、ステップ S 1 5 において、図 1 6 に示す標準の電力パラメータではなく、図 1 5 に示す固有の電力パラメータが適用される。電力パラメータが指定されていないアプリケーションプログラムが動作している場合は、図 1 6 から図 2 2 に示す標準パラメータが適用される。

【0183】

図 1 5 にステップ S 1 7 において参照される、動作状態検出器 1 0 1 で検知したアプリケーションプログラムに対応した固有の電力パラメータの一例が示されている。本例においては、ビデオ画像を表示し録画するアプリケーションプログラム 1 の、起動中、画像表示中、録画動作中、および終了中の電力パラメータが対象である。録画回路は録画動作中の時のみ稼働率を 1 0 0 % としており、その他の場合は稼働率を 0 % に設定する。

40

【0184】

アプリケーションプログラムの起動中は、CPU、HDD、および LCD とも、稼働率を 5 0 % に設定する。画像表示中は、CPU の稼働率は 1 0 % に、HDD の稼働率は 0 % に、そして LCD の稼働率は 1 0 0 % に、それぞれ設定されている。録画動作中は、CPU の稼働率は 8 0 % に、HDD の稼働率は 1 0 0 % に、そして LCD の稼働率は 1 0 0 % に、それぞれ設定されている。なお、終了処理中は、CPU の稼働率は 5 0 % に、HDD の稼働率は 5 0 % に、そして LCD の稼働率は 5 0 % に、それぞれ設定されている。

【0185】

50

図16～図22を参照して、上述のステップS19において参照される電力パラメータテーブルについて、以下に説明する。

図16に、動作状態検出器101で検知したアプリケーションに電力パラメータが特に設定されていない時に適用される標準の電力パラメータの例を示す。電力供給率記憶器103に記憶されている、起動中と終了処理中の各電力パラメータを示している。CPU、HDD、およびLCDに関する電力パラメータ（稼働率）のみを明示しており、その他の電力パラメータ（稼働率）は0%である。なお、本例において、起動中と終了処理中のCPU、HDD、およびLCDの稼働率は50%に設定されている。

【0186】

図17に、ユーザが行ったキー入力の状態に応じて適用される標準の電力パラメータ（稼働率）の例が示されている。CPUおよびHDDに関する電力パラメータ（稼働率）のみを明示しており、その他の電力パラメータ（稼働率）は0%である。キー入力待ち状態時には、CPUの電力パラメータ（稼働率）は30%、HDDの稼働率は0%に設定される。ユーザがキー入力を開始した場合には、CPUおよびHDDの電力パラメータ（稼働率）として、50%が適用される。また、仮名漢字変換などの変換動作を行う場合には通常、HDDからの高速な読み込みが必要であるためHDDの稼働率を100%としているが、CPUの稼働率は50%に設定している。

【0187】

図18に、周辺機器がモデムである場合の電力パラメータの例を示す。モデムのオープン・クローズの動作および通信中かどうかは、API層1005で監視され、動作状態検出器101で検出される。オープンしているが待ち状態の時は、CPUの動作がほとんど必要ないために、CPUの稼働率は10%に設定している。この時モデム自身も待ち状態のため、稼働率を50%に設定している。

【0188】

通信中はCPUをある程度動作させる必要があるため、CPUの稼働率を50%に設定する。また、その場合に、モデムは高速に通信させる必要があるため、稼働率を100%に設定する。一方、クローズ状態の時には、CPUもモデムも動作させる必要がないため、稼働率を0%に設定している。

【0189】

図19に、IrDA（赤外線通信）デバイスに関する電力パラメータ（稼働率）の例を示す。本例においては、図18に示すモデムをIrDAと置き換えた状態で、CPUとIrDAの稼働率がそれぞれ設定されている。

【0190】

図20に、IEEE1394デバイスに関する電力パラメータを示す。本例においても、図18に示すモデムをIEEE1394インターフェースと置き換えた状態で、CPUとIEEE1394インターフェースの稼働率がそれぞれ設定されている。

【0191】

図21に、USBデバイスに関する電力パラメータの一例を示す。本例においても、図18に示すモデムをUSBと置き換えた状態で、CPUとUSBデバイスの稼働率がそれぞれ設定されている。

【0192】

図22に、動作状態検出器101で検知されるAPIの使用状態に対応した電力パラメータ（稼働率）の例を示す。本例においては、音声を再生するAPI、3D機能を実現するAPI、およびビデオ再生を行うAPIについて、それらのAPIを使用中に適用される電力パラメータ（稼働率）を定義している。本例においては、CPUの稼働率は常に100%に設定されている。

【0193】

音声再生のAPIを使用中は、サウンドの稼働率を100%と定義しており、3D機能およびLCDは必要がないため、稼働率を0%に設定している。また、3D機能のAPIを使用中は、3D機能の稼働率は100%であり、サウンド機能の稼働率は0%、そしてL

10

20

30

40

50

C Dの稼働率は70%に設定している。

【0194】

ビデオ再生API使用中はサウンド機能でもデータ処理を行うため、サウンド機能の稼働率を100%に設定している。3D機能は、使用しないため稼働率を0%に設定する。一方、LCDの稼働率は70%に設定する。

【0195】

図23を参照して、上述のステップS9において、電力供給率算出器102によって行われる電力パラメータの計算方法の例を示す。動作状態検出器101で検知された状態に対応する電力パラメータ(稼働率)が電力供給率記憶器103に記憶されており、電力供給率算出器102によってシステム全体に対する電力パラメータ(稼働率)に総合される。この具体的な方法は、既に図9を参照して説明した方法と同一であるので、詳しい説明を省く。なお、図23には、図9に示したリソース(デバイス)に、新たなモデム、IrDA、IEEE1394デバイス、USBデバイス、音声デバイス、および3D機能デバイスが追加されている。

10

【0196】

以下に、図24～図29を参照して、電力供給率算出器102で行われる電力パラメータの計算について説明する。

【0197】

図24に、アプリケーションプログラムが1つも動作していない時の、CPUの電力パラメータ(稼働率)の計算例を示す。システムが動作中でアプリケーションプログラムが1つも動作していない時には、図14に示したシステム全体の電力パラメータ(稼働率)が適用される。つまり、システム状態がアイドル状態であるため、図14に示した電力パラメータのアイドル状態時の値10%(稼働率)がCPUの電力パラメータとして設定される。

20

【0198】

図25に、固有の電力パラメータ(稼働率)が指定されていないアプリケーションプログラム1が1つだけ動作している場合のCPUの電力パラメータの計算例を示す。つまり、アプリケーションプログラムは起動中であるため、図16に示した起動中のCPUの稼働率50%が、本例におけるCPUの電力パラメータとして設定される。

【0199】

図26に、固有の電力パラメータが指定されていないアプリケーションプログラム1と、固有の電力パラメータが指定されているアプリケーションプログラム2とが同時に動作している場合のCPUの電力パラメータの計算例を示す。固有の電力パラメータが指定されていないアプリケーションプログラム1はキー入力待ち状態であるため、図17に示されたパラメータより30%の稼働率が読み出される。ビデオ表示録画アプリケーションプログラム2は固有の電力パラメータ(稼働率)が予め指定されており、図15に示す電力パラメータが適用される。

30

【0200】

アプリケーションプログラム2は録画動作中のため、80%の稼働率が図15から読み出される。これらの電力パラメータ(稼働率)を、図23に例示した計算方法に従って計算する。つまり、稼働率30%と稼働率80%を合計すると、稼働率は110%となる。稼働率が100%を超えるため、図23に例示した計算方法に従って、CPUの電力パラメータである稼働率の値は、100%に設定される。この値がCPUに対する電力パラメータの設定値である。

40

【0201】

図27に、固有の電力パラメータが指定されていないアプリケーションプログラムが2つ動作している場合の計算例を示す。アプリケーションプログラム1がキー入力待ち状態であり、アプリケーションプログラム2がキー入力中の状態である。キー入力待ち状態のCPUの電力パラメータは、図26より稼働率30%が読み出される。キー入力中のアプリケーションについては、図17に示した電力パラメータが適用される。そして、キー入力

50

中のCPUの電力パラメータとして、稼働率50%が読み出される。これらに、図23に示した計算方法を適用すると、稼働率は80%となる。この稼働率が、CPUに設定すべき電力パラメータの値となる。

【0202】

図28に、2つのアプリケーションプログラムが動作している時のモデムの電力パラメータの計算例を示す。動作状態検出器101は、アプリケーションプログラム2がモデムを使用して通信中であることを検知する。電力供給率算出器102は、電力供給率記憶器103から図18に示す通信中のモデムの電力パラメータの値として、稼働率100%を取得する。一方、アプリケーションプログラム1はモデムを使用していないため、モデムの電力パラメータとして、稼働率0%が取得される。図23に示したモデムに関する計算方法を適用すると、この場合のモデムの電力パラメータは稼働率100%となる。

10

【0203】

図29に、2つのアプリケーションが動作している時のサウンドデバイスの電力パラメータの計算例を示す。動作状態検出器101は、アプリケーションプログラム2がサウンド再生中であることを検知する。電力供給率算出器102は、対応する電力パラメータを電力供給率記憶器103より読み出す。一方、アプリケーションプログラム1は、サウンド再生を行っていないので、電力パラメータとしては稼働率0%が読み出される。これらの値に、図23に示したサウンドに関する計算方法を適用すると、サウンドの電力パラメータとして稼働率100%が得られる。

【0204】

20

図24～図29を参照して説明したように得られた各々の電力パラメータ（稼働率）は、電力供給率算出器102から電力供給率割当器104に渡され、最終的にCPU動作制御ドライバ1100_Nやデバイス1200_1から1200_Nのそれぞれが設定される（ステップS11）。また、以上の例では、電力パラメータは、予め電力供給率記憶器103に記憶されているものとして説明したが、ユーザがAPI使用状況に対応した電力パラメータを必要な時に設定できるようにしても良い。

【0205】

図32に、電力制御装置PCA2を組み込んだ電気機器の動作中に、動作状態検出器101によって検出される状態と出力される動作状態信号S1の内容を示す。まず、「動作」が「システム」と表示されているシステムの動作状態に関して述べる。システムが起動中とは、動作状態検出器101が図30に示したMSS1を出力してから、次に信号MSS2が出力されるまでの間である。システムがアイドル中とは、信号MSS3が出力されてから、信号MSS4が出力されるまでの間である。サスペンド中とは、信号MSS5が出力されてから、信号MSS6が出力されるまでの間である。終了処理中とは、信号MSS7が出力されてから、信号MSS8が出力されるまでの間である。終了状態とは、信号MSS8が出力された後である。

30

【0206】

次に、「動作」が「標準アプリケーションプログラム」と表示されているアプリケーションプログラムの動作状態に関して述べる。起動中とは、動作状態検出器101が図31に示した信号MSA1を出力してから、次に信号MSA1が出力されるまでの間である。以下同様に、画像表示中、録画動作中、入力中、待ち状態、保存処理中、モデムオープン状態、モデム通信中、IrDAオープン状態、IrDA通信中、IEEE1392インターフェースオープン状態、IEEE1392インターフェース通信中、USBデバイスオープン状態、USBデバイス通信中、3D機能使用中、音声再生中、ビデオ再生中、および終了処理中のそれぞれの期間に対応する信号MSAが具体的に記載されている。

40

【0207】

さらに、「動作」が「ビデオ表示録画アプリケーションプログラム」と表示されているアプリケーションプログラムの動作状態に関して述べる。起動中とは、動作状態検出器101が信号MSA1を出力してから、次信号MSA2が出力されるまでの間である。画像表示中とは、信号MSA5が出力されてから、信号MSA6が出力されるまでの間である。

50

録画動作中とは、信号M S A 7が出力されてから、信号M S A 8が出力されるまでの間である。終了処理中とは、信号M S A 3が出力されてから、信号M S A 4が出力されるまでの間である。

【0208】

上述のように、本発明においては、アプリケーションプログラムの実行状態に応じて、各リソースに電力を適正かつ効果的に配分し、電力の過剰供給による無駄な消費を防止すると共に、消費電力を低減できる。

さらに、本発明によれば、C P Uや周辺機器の性能が必要なアプリケーションプログラムを使用する時に発生する手動による電力制御設定の手間を省くとともに、アプリケーションプログラムが動作していない時の消費電力を最小限に設定できる。

10

【0209】

さらに、アプリケーションプログラムの実行に実際に必要な電力量にて、各リソースを動作させるので、不要な電力消費を防止できる。また、負荷率を一旦設定したリソースの設定可能負荷率の変更、或いは、リソースが別のものに交換された場合、またはアプリケーションプログラムの動作が変更になった場合にも、負荷率を適正な値に設定し直せる。

【0210】

なお、アプリケーションプログラムの個々の動作の実行に対して、同一のリソースに同時に必要とされる負荷を、システム全体に対する負荷に応じて正しく決定できる。そして、リソースの実際の能力に応じてアプリケーションプログラムからの負荷要求に対して最大限応えることができる。また、本発明は、オペレーションシステムを有するコンピュータシステムにも適用できる。コンピュータシステムに接続されている任意の周辺機器の電力消費量を制御できる。

20

【0211】

各リソースに備えられた動作制御機能を利用して消費電力を制御できる。また、アプリケーションプログラムの実行に実際に必要な電力量で、各リソースを動作させるので、不要な電力消費を防止できる。

【図面の簡単な説明】

【図1】本発明の第1の実施形態にかかる電力制御装置の構成を示すブロック図である。

【図2】図1に示した動作状態検出器が生成する動作状態信号の詳細例を示す表である。

【図3】図1に示した動作状態検出器によって検出される状態と出力される動作状態信号の詳細例を示す表である。

30

【図4】図1に示した電力制御装置の動作を表すフローチャートである。

【図5】図1に示した電力供給率記憶器に格納されているシステム全体の電力パラメータの設定値の一例を示す表である。

【図6】アプリケーションプログラム1の各動作状態に対応して設定された電力パラメータの割当の一例を示す表である。

【図7】アプリケーションプログラム2の各動作状態に対応して設定された電力パラメータの一例を表である。

【図8】アプリケーションプログラム3の各動作状態に対応して設定された電力パラメータの一例を示す表である。

40

【図9】アプリケーションプログラムの動作状態に対応した個々の電力パラメータからシステム全体としての各デバイスの電力パラメータの値を計算する方法の一例を示す表である。

【図10】2つのアプリケーションプログラムが実行中の電力パラメータの計算を示す表である。

【図11】3つのアプリケーションプログラムが実行中のシステム全体としての各デバイスの電力パラメータの計算を示す表である。

【図12】本発明の第2の実施形態にかかる電力制御装置の構成を示すブロック図であり

【図13】図12に示した電力制御装置の動作を表すフローチャートである。

【図14】図13に示したフローチャートにおいて参照される電力パラメータの例を示す

50

表である。

【図 1 5】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 1 6】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 1 7】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 1 8】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 1 9】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。 10

【図 2 0】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 2 1】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 2 2】図 1 3 に示したフローチャートにおいて参照される電力パラメータの例を示す表である。

【図 2 3】図 1 3 に示すステップ S 9 において、電力供給率算出器によって行われる電力パラメータの計算方法を示す表である。

【図 2 4】電力供給率算出器で行われる電力パラメータの計算方法を示す表である。 20

【図 2 5】電力供給率算出器で行われる電力パラメータの計算方法を示す表である。

【図 2 6】電力供給率算出器で行われる電力パラメータの計算方法を示す表である。

【図 2 7】電力供給率算出器で行われる電力パラメータの計算方法を示す表である。

【図 2 8】電力供給率算出器で行われる電力パラメータの計算方法を示す表である。

【図 2 9】電力供給率算出器で行われる電力パラメータの計算方法を示す表である。

【図 3 0】カーネル層が出力する動作状態監視信号の内容を示す表である。

【図 3 1】A P I 層が出力する A P I 通信情報の内容を示す表である。

【図 3 2】図 1 2 に示した電力制御装置を組み込んだ電気機器の動作中に、動作状態検出器によって検出される状態と出力される動作状態信号の内容を示す表である。

【符号の説明】 30

P C A 1 電力制御装置

1 0 1 動作状態検出器

1 0 2 電力供給率算出器

1 0 3 電力供給率記憶器

1 0 4 電力供給率割当器

1 0 5 C P U

1 0 6 __ 1 ~ 1 0 6 __ N 周辺機器

S 1 動作状態信号

S 2 稼働率問い合わせ信号

S 3 選出稼働率信号

S 4 稼働率信号

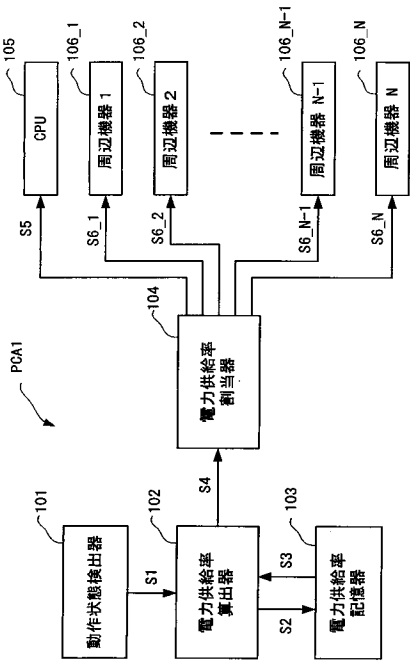
S 5 稼働率割当信号

S 6 __ 1 ~ 1 0 6 __ N 稼働率割当信号

P C A 2 電力制御装置

1 0 0 5 A P I 層

【図 1】



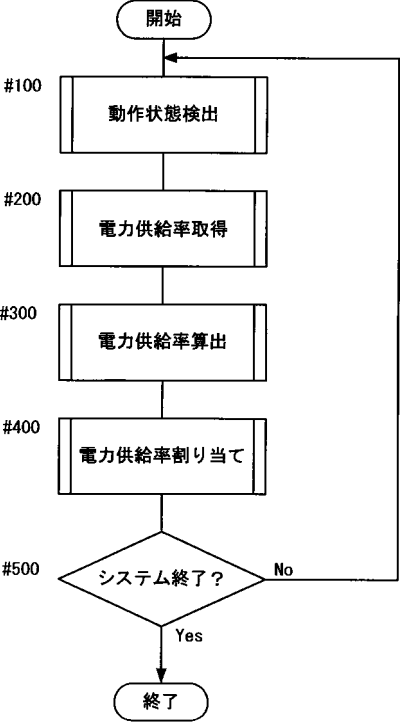
【図 2】

動作状態と状態信号		
動作	状態	S1
システム	システム起動処理開始	S1_S1
	システム起動処理完了	S1_S2
	アイドル状態開始	S1_S3
	アイドル状態終了	S1_S4
	サスペンドモードに移行	S1_S5
	サスペンドモードから復帰	S1_S6
	システム終了処理開始	S1_S7
	システム終了処理完了	S1_S8
アプリケーション プログラム1 - ビデオの 表示・録画 -	起動処理開始	S1_A1_1
	起動処理完了	S1_A1_2
	画像表示開始	S1_A1_3
	画像表示終了	S1_A1_4
	録画処理開始	S1_A1_5
	録画処理終了	S1_A1_6
	終了処理開始	S1_A1_7
	終了処理完了	S1_A1_8
アプリケーション プログラム2 - ワードプロセッサ -	起動処理開始	S1_A2_1
	起動処理完了	S1_A2_2
	表示開始	S1_A2_3
	表示終了	S1_A2_4
	入力処理開始 (キー入力検知)	S1_A2_5
	入力処理終了 (最後のキー入力より所定時間経過)	S1_A2_6
	文書保存処理開始	S1_A2_7
	文書保存処理完了	S1_A2_8
	終了処理開始	S1_A2_9
	終了処理完了	S1_A2_10
アプリケーション プログラム3 - 時計表示 -	起動処理開始	S1_A3_1
	起動処理完了	S1_A3_2
	表示開始	S1_A3_3
	表示終了	S1_A3_4
	終了処理開始	S1_A3_5
	終了処理完了	S1_A3_6

【図 3】

動作	状態	動作状態信号
システム	起動中	S1_S1 ~ S1_S2
	アイドル中	S1_S3 ~ S1_S4
	サスペンド中	S1_S5 ~ S1_S6
	終了処理中	S1_S7 ~ S1_S8
	終了	S1_S8以降
アプリケーションプログラム1 - ビデオの表示・録画 -	起動中	S1_A1_1 ~ S1_A1_2
	画像表示中	S1_A1_3 ~ S1_A1_4
	録画中	S1_A1_5 ~ S1_A1_6
	終了処理中	S1_A1_7 ~ S1_A1_8
	起動中	S1_A2_1 ~ SA_A2_2
アプリケーションプログラム2 - ワードプロ -	表示中	S1_A2_3 ~ SA_A2_4
	入力中	S1_A2_5 ~ S1_A2_6
	保存中	S1_A2_7 ~ S1_A2_8
	終了処理中	S1_A2_9 ~ S1_A2_10
	起動中	S1_A3_1 ~ S1_A3_2
アプリケーションプログラム3 - 時計表示 -	表示中	S1_A3_3 ~ S1_A3_4
	終了処理中	S1_A3_5 ~ S1_A3_6

【図 4】



【図 9】

システム全体としての割り当て稼働率算出

リソース		割り当て	
CPU	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。		
HDD	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。		
LCD	稼働率の最大値とする。		
メモリ	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。		
VRC	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。		

【図 10】

アプリケーションプログラム1(画像表示中)とアプリケーションプログラム2(入力中)の同時実行に対する稼働率(システム全体)割り当て

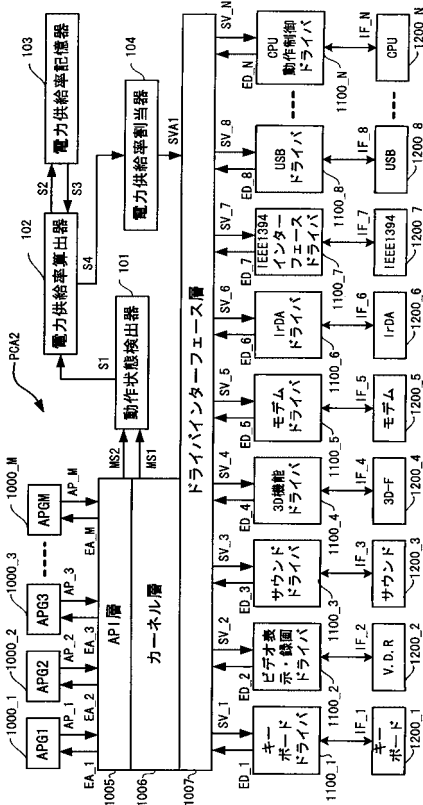
アプリケーション プログラムの動作状態	稼働率 (%)			
	CPU	HDD	LCD	Memory
アプリケーションプログラム1 画像表示中	10	0	100	100
アプリケーションプログラム2 入力中	50	20	50	100
設定稼働率(システム全体)	60	20	100	100

【図 11】

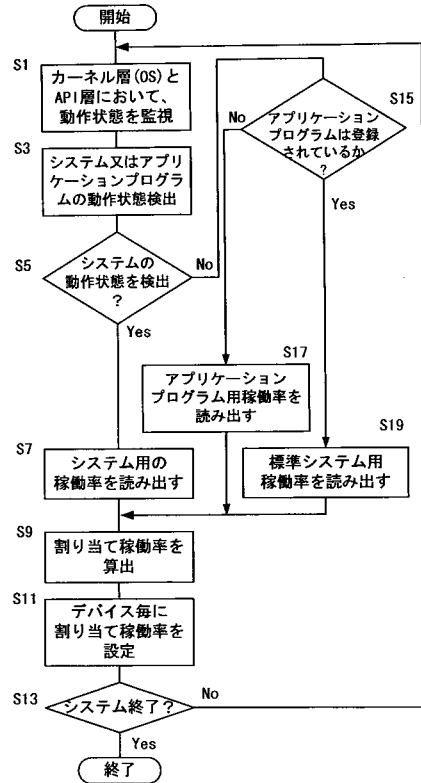
アプリケーションプログラム1(録画中)、とアプリケーションプログラム2(表示中)、および
アプリケーションプログラム3(表示中)の同時実行に対する稼働率(システム全体)割り当て

アプリケーション プログラムの動作状態		稼働率 (%)			
録画中	表示中	CPU	HDD	LCD	Memory
		80	100	100	100
アプリケーションプログラム2	表示中	10	0	50	100
アプリケーションプログラム3	表示中	10	0	25	100
設定稼働率(システム全体)		100	100	100	100

【図 12】



【図 13】



【図 14】

システム全体に対する稼働率割り当て

システムの動作状態	稼働率 (%)			メモリ
	CPU	HDD	LCD	
起動中	100	100	25	100
アイドル中	10	0	25	100
サスペンド中	0	0	0	10
終了処理中	50	50	0	100
終了	0	0	0	0

【図 15】

アプリケーションプログラム1(ビデオの表示・録画)実行に対する稼働率割り当て

アプリケーションプログラム1の動作状態	稼働率 (%)			VRC
	CPU	HDD	LCD	
起動中	50	50	50	0
画像表示中	10	0	100	0
録画中	80	100	100	100
終了処理中	50	50	50	0

【図 16】

標準稼働率:起動処理終了

アプリケーションプログラムの状態	稼働率 (%)		
	CPU	HDD	LCD
起動中	50	50	50
終了処理中	50	50	50

【図 17】

キー入力の標準稼働率

アプリケーションプログラムの状態	稼働率 (%)	
	CPU	HDD
キー入力待ち	30	0
キー入力中	50	50
文字変換	50	100

【図 18】

モデムの標準稼働率

アプリケーションプログラムの状態	稼働率 (%)	
	CPU	モデム
待機中(オープン)	10	50
通信中(オープン)	50	100
クローズ	0	0

【図 19】

IrDAの標準稼働率

アプリケーションプログラムの状態	稼働率 (%)	
	CPU	IrDA
待機中(オープン)	10	50
通信中(オープン)	50	100
クローズ	0	0

【図 2 0】

IEEE 1394 インタフェースの標準稼働率

アプリケーションプログラムの 状態	稼働率 (%)	
	CPU	IEEE1394
待機中 (オープン)	0	50
通信中 (オープン)	50	100
クローズ	0	0

【図 2 1】

USBの標準稼働率

アプリケーションプログラムの 状態	稼働率 (%)	
	CPU	USB
待機中 (オープン)	0	50
通信中 (オープン)	50	100
クローズ	0	0

【図 2 2】

音声再生API、3D機能API、およびビデオ再生APIの標準稼働率

アプリケーションプログラムの稼働率	稼働率 (%)			
	CPU	音声	3D機能	LCD
音声再生API使用中	100	100	0	0
3D機能API使用中	100	0	100	70
ビデオ再生API使用中	100	100	0	70

【図 2 3】

リソース	割り当て	
	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
CPU	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
HDD	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
LCD	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
メモリ	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
VRM	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
モザム	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
IrDA	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
IEEE1394	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
USB	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
音声	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。
3D機能	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。	各稼働率の合計とする。但し、合計が100%を超える場合は、合計は100%に設定する。

割り当て稼働率算出

【図 2 4】

例1: CPUの稼働率割り当て

システム状態	稼働率 (%)
アイドル中	10
稼働率設定値	10

【図 2 5】

例2: CPUの稼働率割り当て

アプリケーションプログラム1の状態	稼働率 (%)
起動中	50
稼働率設定値	50

【図 2 6】

例3: CPUの稼働率割り当て

状態	稼働率 (%)
アプリケーションプログラム1、キー入力待ち	30
アプリケーションプログラム2、録画中	80
稼働率設定値	100

【図 2 7】

例4: CPUの稼働率割り当て

状態	稼働率 (%)
アプリケーションプログラム1、キー入力待ち	30
アプリケーションプログラム2、キー入力中	50
稼働率設定値	80

【図 28】

例: モデムの稼働率割り当て

状態	稼働率 (%)
アプリケーションプログラム1、キー入力待ち	0
アプリケーションプログラム2、通信中	100
稼働率設定値	100

【図 29】

例: オーディオデバイスの稼働率割り当て

アプリケーションプログラム1の状態	稼働率 (%)
アプリケーションプログラム1、キー入力中	0
アプリケーションプログラム2、オーディオ再生API使用中	100
稼働率設定値	100

【図 30】

状態と信号

状態	信号
システム起動処理開始	MSS1
システム起動処理完了	MSS2
アイドル開始	MSS3
アイドル終了	MSS4
サスペンドに移行	MSS5
サスペンドから復帰	MSS6
システム終了処理開始	MSS7
システム終了処理完了	MSS8

【図 31】

状態	信号
起動処理開始 API	MSA1
起動処理完了 API	MSA2
終了処理開始 API	MSA3
終了処理完了 API	MSA4
表示開始 API	MSA5
表示終了 API	MSA6
録画処理開始 API	MSA7
録画処理終了 API	MSA8
キー入力 API	MSA9
キー入力検出後所定時間経過 API	MSA10
保存処理開始 API	MSA11
保存処理終了 API	MSA12
モデムオープン API	MSA13
モデムクローズ API	MSA14
モデム通信開始 API	MSA15
モデム通信終了 API	MSA16
IrDA オープン API	MSA17
IrDA クローズ API	MSA18
IrDA 通信開始 API	MSA19
IrDA 通信終了 API	MSA20
IEEE1394 I/O オープン API	MSA21
IEEE1394 I/O クローズ API	MSA22
IEEE1394 I/O 通信開始 API	MSA23
IEEE1394 I/O 通信終了 API	MSA24
USB オープン API	MSA25
USB クローズ API	MSA26
USB 通信開始 API	MSA27
USB 通信終了 API	MSA28
3D 機能開始 API	MSA29
3D 機能終了 API	MSA30
オーディオ再生開始 API	MSA31
オーディオ再生終了 API	MSA32
ビデオ再生開始 API	MSA33
ビデオ再生終了 API	MSA34

【図 32】

動作状態信号による動作状態定義

動作	状態	状態信号
システム	起動中	MSS1 ~ MSS2
	アイドル中	MSS3 ~ MSS4
	サスペンド中	MSS5 ~ MSS6
	終了処理中	MSS7 ~ MSS8
	終了	After MSS8
標準 アプリケーション プログラム	起動中	MSA1 ~ MSA2
	画像表示中	MSA5 ~ MSA6
	録画中	MSA7 ~ MSA8
	入力中	MSA9 ~ MSA10
	待機中	MSA10 ~ MSA9
	保存処理中	MSA11 ~ MSA12
	モデムオープン	MSA13 ~ MSA14
	モデム通信中	MSA15 ~ MSA16
	IrDA オープン	MSA17 ~ MSA18
	IrDA 通信中	MSA19 ~ MSA20
	IEEE1394 I/O オープン	MSA21 ~ MSA22
	IEEE1394 I/O 通信中	MSA23 ~ MSA24
	USB オープン	MSA25 ~ MSA26
	USB 通信中	MSA27 ~ MSA28
	3D 機能活動中	MSA29 ~ MSA30
	オーディオ再生	MSA31 ~ MSA32
	録画中	MSA33 ~ MSA34
	終了処理中	MSA3 ~ MSA4
ビデオ表示・録画用 アプリケーション プログラム	起動中	MSA1 ~ MSA2
	表示中	MSA5 ~ MSA6
	録画中	MSA7 ~ MSA8
	終了処理中	MSA3 ~ MSA4

フロントページの続き

(58)調査した分野(Int.Cl., D B名)

G06F 1/26

G06F 1/32