(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5827242号
(P5827242)

(45)発行日　平成27年12月2日(2015.12.2)　(24)登録日　平成27年10月23日(2015.10.23)

(51)Int.Cl.			F I		
G06F	1/32	(2006.01)	G06F	1/32	Z
G06F	1/28	(2006.01)	G06F	1/28	Z

請求項の数 48　(全 26 頁)

(21)出願番号	特願2012-551212 (P2012-551212)
(86)(22)出願日	平成23年1月24日 (2011.1.24)
(65)公表番号	特表2013-518346 (P2013-518346A)
(43)公表日	平成25年5月20日 (2013.5.20)
(86)国際出願番号	PCT/US2011/022217
(87)国際公開番号	WO2011/094148
(87)国際公開日	平成23年8月4日 (2011.8.4)
審査請求日	平成26年1月7日 (2014.1.7)
(31)優先権主張番号	12/695,648
(32)優先日	平成22年1月28日 (2010.1.28)
(33)優先権主張国	米国(US)

(73)特許権者　508002601
カビウム・インコーポレーテッド
アメリカ合衆国，カリフォルニア州　９５１３１，サンノゼ，ノース　ファースト　ストリート　２３１５
(74)代理人　100087941
弁理士　杉本　修司
(74)代理人　100086793
弁理士　野田　雅士
(74)代理人　100112829
弁理士　堤　健郎
(74)代理人　100144082
弁理士　林田　久美子

最終頁に続く

(54)【発明の名称】電力制御方法および電力制御装置

(57)【特許請求の範囲】

【請求項１】

コンピュータチップの電力を制御する方法であって、

電力制御サイクル中、命令を発行する間に消費された電力に関する電力情報を累積する過程と、

前記累積した電力情報を所定の閾値と比較する過程と、

前記累積した電力情報が前記所定の閾値よりも大きい場合、前記コンピュータチップの電力消費を減少させるように、さらなる命令の発行を阻止する過程と、

さらなる命令の発行が阻止された後に消費された電力を求める過程と、

さらなる命令の発行が阻止された後に消費された前記電力を、後続の電力制御サイクルの累積される電力情報に加算して、総電力値を取得する過程と、

前記総電力値が前記所定の閾値よりも大きい場合、前記後続の電力制御サイクルにおけるさらなる命令の発行を阻止する過程とを備えた、コンピュータチップ電力制御方法。

【請求項２】

請求項１に記載のコンピュータチップ電力制御方法において、さらに、

命令を発行する間に消費された電力に関する前記電力情報を、所定の時間中に累積する過程を備えた、コンピュータチップ電力制御方法。

【請求項３】

請求項１に記載のコンピュータチップ電力制御方法において、さらに、

命令を発行する間に消費された電力に関する前記電力情報を、発行される各命令に伴う

電力を推定して累積する過程を備えた、コンピュータチップ電力制御方法。
【請求項４】
請求項３に記載のコンピュータチップ電力制御方法において、さらに、
前記発行される各命令に伴う電力を、既知の命令についての電力値についてのデータベースを用いて推定する過程を備えた、コンピュータチップ電力制御方法。
【請求項５】
請求項３に記載のコンピュータチップ電力制御方法において、さらに、
前記発行される各命令に伴う電力を、前記各命令の処理の間に経過するクロックサイクル数をそれぞれ求めて推定する過程を備えた、コンピュータチップ電力制御方法。
【請求項６】
請求項１に記載のコンピュータチップ電力制御方法において、さらに、
命令を発行する間に消費された電力に関する前記電力情報を、１つ以上の命令サイクル中に累積する過程を備えた、コンピュータチップ電力制御方法。
【請求項７】
請求項６に記載のコンピュータチップ電力制御方法において、さらに、
１つ以上の命令サイクルが経過するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
【請求項８】
請求項６に記載のコンピュータチップ電力制御方法において、さらに、
現在の、電力消費が減少させられている期間が完了するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
【請求項９】
請求項６に記載のコンピュータチップ電力制御方法において、さらに、
所定の数の命令サイクルが経過するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
【請求項１０】
請求項１に記載のコンピュータチップ電力制御方法において、さらに、
所定の時間が経過するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
【請求項１１】
請求項１に記載のコンピュータチップ電力制御方法において、さらに、
前記所定の閾値を、前記コンピュータチップの最大許容電力との相関関係において決定する過程を備えた、コンピュータチップ電力制御方法。
【請求項１２】
請求項１に記載のコンピュータチップ電力制御方法において、さらに、
前記所定の閾値を、１つ以上のチップヒューズとの相関関係において決定する過程を備えた、コンピュータチップ電力制御方法。
【請求項１３】
電力制御サイクル中、命令を発行する間に消費された電力に関する電力情報を累積する電力累積手段と、
電力制御サイクル中、前記累積した電力情報を所定の閾値と比較する比較部と、
電力制御サイクル中、前記累積した電力情報が前記所定の閾値よりも大きい場合、電力消費を減少させるように、さらなる命令の発行を阻止する制御部と、
さらなる命令の発行が阻止された後に消費された電力を推定するオーバーシュート電力推定手段であって、当該消費された電力が、後続の電力制御サイクルの累積される電力情報に加算されて、総電力値を取得する、電力推定手段とを備え、
前記制御ユニットは、さらに、前記後続の電力制御サイクル中に、前記総電力値が前記所定の閾値よりも大きい場合、前記後続の電力制御サイクルにおけるさらなる命令の発行を阻止する、プロセッサ。
【請求項１４】

請求項１３に記載のプロセッサにおいて、前記電力累積手段が、命令を発行する間に消費された電力に関する前記電力情報を、所定の時間中に累積する、プロセッサ。
【請求項１５】
請求項１３に記載のプロセッサにおいて、前記電力累積手段が、命令を発行する間に消費された電力に関する前記消費電力情報を、発行される各命令に伴う電力を推定して累積する、プロセッサ。
【請求項１６】
請求項１５に記載のプロセッサにおいて、さらに、
前記発行される各命令に伴う電力を、既知の命令に伴う電力値についてのデータベースを用いて推定する電力推定手段を備えた、プロセッサ。
【請求項１７】
請求項１３に記載のプロセッサにおいて、さらに、
前記発行される各命令に伴う電力を、前記各命令の処理の間に経過するクロックサイクル数をそれぞれ求めて推定する電力推定手段を備えた、プロセッサ。
【請求項１８】
請求項１３に記載のプロセッサにおいて、前記電力累積手段が、命令を発行する間に消費された電力に関する前記電力情報を、１つ以上の命令サイクル中に累積する、プロセッサ。
【請求項１９】
請求項１８に記載のプロセッサにおいて、前記制御部が、１つ以上の命令サイクルが経過するまで、さらなる命令の発行を阻止する、プロセッサ。
【請求項２０】
請求項１８に記載のプロセッサにおいて、前記制御部が、現在の、電力消費が減少させられている期間が完了するまで、さらなる命令の発行を阻止する、プロセッサ。
【請求項２１】
請求項１８に記載のプロセッサにおいて、前記制御部が、所定の数の命令サイクルが経過するまで、さらなる命令の発行を阻止する、プロセッサ。
【請求項２２】
請求項１３に記載のプロセッサにおいて、前記制御部が、所定の時間間隔が経過するまで、さらなる命令の発行を阻止する、プロセッサ。
【請求項２３】
請求項１３に記載のプロセッサにおいて、前記比較部が、前記所定の閾値を、最大許容電力との相関関係において決定する、プロセッサ。
【請求項２４】
請求項１３に記載のプロセッサにおいて、前記比較部が、前記所定の閾値を、１つ以上のチップヒューズとの相関関係において決定する、プロセッサ。
【請求項２５】
プロセッサの電力を制御する方法であって、
電力制御サイクル中、１つ以上のイベントを処理する間に前記プロセッサによって消費された電力の第１電力消費推定値を推定する過程と、
電力消費を減少させるために、前記第１電力消費推定値との相関関係において、前記プロセッサのさらなる処理を阻止する過程と、
さらなる処理が阻止された後に消費された電力を推定して、オーバーシュート電力値を決定する過程と、
後続の電力制御サイクル中、１つ以上のイベントを処理する間に前記プロセッサによって消費された電力の第２電力消費推定値を推定し、かつ、当該第２電力消費および前記オーバーシュート電力値との相関関係において、前記プロセッサでの１つ以上のイベントのさらなる処理を阻止する過程とを備えた、プロセッサ電力制御方法。
【請求項２６】
請求項２５に記載のプロセッサ電力制御方法において、さらに、

前記プロセッサによって消費された電力を、前記プロセッサに流れ込む電流との相関関係において推定する過程を備えた、プロセッサ電力制御方法。

【請求項２７】

請求項２５に記載のプロセッサ電力制御方法において、さらに、

前記プロセッサによって消費された電力を、前記１つ以上のイベントにおける各イベントによって前記プロセッサで構成される切替キャパシタンスとの相関関係において推定する過程を備えた、プロセッサ電力制御方法。

【請求項２８】

請求項２５に記載のプロセッサ電力制御方法において、さらに、

前記推定した電力を、所定時間および所定のイベントサイクル数のうちの少なくとも一方の間に累積する過程を備えた、プロセッサ電力制御方法。

【請求項２９】

請求項２８に記載のプロセッサ電力制御方法において、さらに、

前記累積した電力が所定の閾値よりも大きい場合、前記プロセッサのさらなる処理を阻止する過程を備えた、プロセッサ電力制御方法。

【請求項３０】

請求項２８に記載のプロセッサ電力制御方法において、前記１つ以上のイベントが、イベントサイクル中に前記プロセッサによって発行される命令、イベントサイクル中に有効にされる条件付クロック、およびイベントサイクル中にアクセスされるキャッシュのうちの少なくとも１つを備えた、プロセッサ電力制御方法。

【請求項３１】

請求項２５に記載のプロセッサ電力制御方法において、さらに、

前記プロセッサのキューまたは発行のパイプラインにスロットル信号を発行して、さらなる処理を阻止する過程を備えた、プロセッサ電力制御方法。

【請求項３２】

プロセッサであって、

電力制御サイクル中における１つ以上のイベントを処理する間に当該プロセッサによって消費された電力の第１電力消費推定値、および後続の電力制御サイクル中における１つ以上のイベントを処理する間に当該プロセッサによって消費された電力の第２電力消費推定値を推定する電力推定手段と、

電力消費を減少させるために、前記第１電力消費推定値との相関関係において、当該プロセッサのさらなる処理を阻止する電力制御部と、

さらなる処理が阻止された後に消費された電力を推定して、オーバーシュート電力値を決定するオーバーシュート電力推定手段とを備え、

前記電力制御部が、前記第２電力消費推定値および前記オーバーシュート電力値との相関関係において、前記後続の電力制御サイクル中、当該プロセッサでの１つ以上のイベントのさらなる処理を阻止する、プロセッサ。

【請求項３３】

請求項３２に記載のプロセッサにおいて、前記電力推定手段が、当該プロセッサによって消費された前記電力を、当該プロセッサに流れ込む電流との相関関係において推定する、プロセッサ。

【請求項３４】

請求項３２に記載のプロセッサにおいて、前記電力推定手段が、当該プロセッサによって消費された前記電力を、前記１つ以上のイベントにおける各イベントによって当該プロセッサで構成される切替キャパシタンスとの相関関係において推定する、プロセッサ。

【請求項３５】

請求項３２に記載のプロセッサにおいて、さらに、

前記推定した電力を、所定の時間および所定のイベントサイクル数のうちの少なくとも一方の間に累積する電力累積手段を備えた、プロセッサ。

【請求項３６】

請求項３５に記載のプロセッサにおいて、前記電力制御部が、前記累積した電力が所定の閾値よりも大きい場合、当該プロセッサのさらなる処理を阻止する、プロセッサ。
【請求項３７】
請求項３５に記載のプロセッサにおいて、前記１つ以上のイベントが、イベントサイクル中に当該プロセッサによって発行される命令、イベントサイクル中に有効にされる条件付クロック、およびイベントサイクル中にアクセスされるキャッシュのうちの少なくとも１つを含む、プロセッサ。
【請求項３８】
請求項３２に記載のプロセッサにおいて、前記電力制御部が、当該プロセッサのキューまたは発行のパイプラインにスロットル信号を発行して、さらなる処理を阻止する、プロセッサ。
【請求項３９】
電力消費を制御する方法であって、
電力制御サイクル中、１つ以上のタスクを実行する間に、あるユニットで消費された電力に関する第１電力情報を累積し、後続の電力制御サイクル中、１つ以上のタスクを実行する間に、あるユニットで消費された電力に関する第２電力情報を累積する過程と、
前記累積した第１電力情報が所定の閾値よりも大きい場合、前記電力制御サイクル中に実行する前記１つ以上のタスクのさらなる実行を阻止することによって、前記ユニットの電力消費を制御する過程と、
前記１つ以上のタスクのさらなる実行が阻止された後に消費された電力に関するオーバーシュート電力情報を推定する過程と、
前記オーバーシュート電力情報と前記第２電力情報の合計が前記所定の閾値よりも大きい場合、前記後続の電力制御サイクル中の１つ以上のタスクのさらなる実行を阻止する過程とを備えた、電力消費制御方法。
【請求項４０】
請求項３９に記載の電力消費制御方法において、さらに、
各タスクを実行する間に消費された電力を推定して、前記電力情報を累積する過程を備えた、電力消費制御方法。
【請求項４１】
請求項４０に記載の電力消費制御方法において、さらに、
各タスクに伴う既知の電力値についてのデータベースに基づいて、前記消費された電力を推定する過程を備えた、電力消費制御方法。
【請求項４２】
請求項３９に記載の電力消費制御方法において、さらに、
所定時間および所定のタスク数のうちの少なくとも一方の間、前記ユニットの前記電力情報を累積する過程を備えた、電力消費制御方法。
【請求項４３】
請求項３９に記載の電力消費制御方法において、さらに、
前記ユニットの実行パイプラインにスロットル信号を発行して、前記１つ以上のタスクのさらなる実行を阻止する過程を備えた、電力消費制御方法。
【請求項４４】
電力消費を制御する装置であって、
電力制御サイクル中、１つ以上のタスクを実行する間に、あるユニットで消費された電力に関する第１電力情報、および後続の電力制御サイクル中、１つ以上のタスクを実行する間に、あるユニットで消費された電力に関する第２電力情報を累積する電力累積手段と、
前記累積した第１電力情報が所定の閾値よりも大きい場合、前記電力制御サイクル中に実行する前記１つ以上のタスクのさらなる実行を阻止することによって、前記ユニットの電力消費を制御する電力制御手段とを備え、
前記電力制御手段は、

前記１つ以上のタスクのさらなる実行が阻止された後に消費された電力に関するオーバーシュート電力情報を推定し、

前記後続の電力制御サイクル中に、前記オーバーシュート電力情報と前記第２電力情報の合計が前記所定の閾値よりも大きい場合、１つ以上のタスクのさらなる実行を阻止する、電力消費制御装置。

【請求項４５】

請求項４４に記載の電力消費制御装置において、前記電力累積手段が、各タスクを実行する間に消費された電力を推定して、前記電力情報を累積する、電力消費制御装置。

【請求項４６】

請求項４５に記載の電力消費制御装置において、前記電力累積手段が、各タスクに伴う既知の電力値についてのデータベースに基づいて、前記消費された電力を推定する電力推定手段を有する、電力消費制御装置。

【請求項４７】

請求項４４に記載の電力消費制御装置において、前記電力累積手段が、所定時間および所定のタスク数のうちの少なくとも一方の間、前記ユニットの前記電力情報を累積する、電力消費制御装置。

【請求項４８】

請求項４４に記載の電力消費制御装置において、前記電力制御手段が、前記ユニットの実行パイプラインにスロットル信号を発行して、前記１つ以上のタスクのさらなる実行を阻止する、電力消費制御装置。

【発明の詳細な説明】

【関連出願】

【０００１】

本願は、2010年1月28日付出願の米国特許出願第12/695,648号の利益を主張する。この米国特許出願の全教示内容は、参照をもって本明細書に取り入れたものとする。

【技術分野】

【０００２】

コンピュータチップの総消費電力は、ハードウェア設計だけでなく当該コンピュータチップで実行されるソフトウェアに依存する。また、一部のソフトウェアアプリケーションの消費電力は、他のソフトウェアアプリケーションの消費電力よりも大幅に（例えば、最大で５倍）大きい。電力損失とシステムの維持に伴うコストとは直接関係しているので、最悪の場合の電力損失は、システム設計において考慮すべき重要なファクタ（因子）である。最悪の場合の電力損失は、主に電源および冷却システムに対して悪影響を及ぼす。

【発明の概要】

【０００３】

本発明の例示的な一実施形態は、コンピュータチップの電力を制御する方法およびこれに対応する装置に関する。この例示的な実施形態は、命令を発行することによって生じる電力を累積し、累積した電力を所定の閾値と比較し、前記累積した電力が前記所定の閾値を超えると、前記コンピュータチップの電力消費を減少させるために、さらなる命令の発行を阻止する。本明細書において「電力累積」とは、システムにおいて消費される電力（消費電力）または損失される電力（損失電力）についての推定値（電力に関する情報）を合計することを指す。

【０００４】

本発明の他の実施形態は、１つ以上のイベントを処理する間にプロセッサによって消費される電力を推定し、電力消費を減少させるために、推定された電力に応じて、前記プロセッサのさらなる処理を阻止する、プロセッサの電力を制御する方法およびこれに対応する装置に関する。

【０００５】

本発明のさらに他の実施形態は、１つ以上のタスクを実行する間あるユニットで消費される電力を累積することによって電力消費を制御する方法およびこれに対応する装置に関

する。この例示的な実施形態は、累積した電力が所定の閾値を超えると、前記１つ以上のタスクのさらなる実行を阻止することにより、前記ユニットの電力消費を制御する。

【０００６】

本発明のさらに他の実施形態は、コンピュータチップの温度を制御する方法およびこれに対応する装置に関する。この例示的な実施形態は、前記チップの目下（現在）の温度を監視し、前記チップの目下の温度が所定の閾値を超えると、当該チップの温度を減少させるために、さらなるイベントの処理を阻止する。

【０００７】

前述の内容を踏まえて、以下の説明は、電力を制御するシステムに組み込むことが可能な例示的な実施形態および構成についての説明である。なお、「システム」とは、本発明の実施形態に関しては、システム、サブシステム、装置、方法またはこれらの任意の組合せのことを指す。

【０００８】

前記システムは、発行される各命令に伴う電力を、既知の命令に伴う電力値についてのデータベースを用いて推定してもよい。前記システムは、当該命令の処理の間に経過するクロックサイクル数に対して消費される電力を求め、これから発行される各命令に伴う電力を推定してもよい。前記システムは、各タスクに伴う既知の電力値についてのデータベースに基づいて、消費電力を推定してもよい。

【０００９】

前記システムは、当該プロセッサに流れ込む電流、および／または１つ以上のイベントのうちの各イベントによって当該プロセッサで構成される切替キャパシタンスに応じて、プロセッサによって消費される電力を推定してもよい。

【００１０】

前記システムは、命令を発行することによって生じる電力を、所定の時間中、および／または１つ以上の命令サイクル中に累積してもよい。前記システムは、発行される各命令に伴う電力を推定し、これらに応じて、命令を発行することによって生じる電力を累積してもよい。さらなる命令の発行は、１つ以上の命令サイクルが経過するまで、目下（現在）のクロックサイクルが完了するまで、所定の数の命令サイクルが経過するまで、および／または所定の時間枠が経過するまで阻止されてもよい。

【００１１】

前記システムは、電力の推定値を、所定の時間枠および所定のイベントサイクル数のうちの少なくとも一方の間に累積し、累積した電力が所定の閾値を超えると、プロセッサのさらなる処理を阻止してもよい。前記１つ以上のイベントには、イベントサイクルの間にプロセッサによって発行される命令、イベントサイクルの間に有効にされる条件付クロック、およびイベントサイクルの間にアクセスされるキャッシュのうちの少なくとも１つが含まれてもよい。本明細書において「イベントサイクル」とは、イベントが処理または実行される期間のことを指す。

【００１２】

前記システムは、各タスクを実行する間に消費電力を推定して電力を累積してもよく、および／または所定の時間枠および所定のタスク数のうちの少なくとも一方の間に消費電力を推定して電力を累積してもよい。

【００１３】

前記システムは、前記所定の閾値を、前記コンピュータチップの最大許容電力に応じて決定してもよい。

【００１４】

前記システムは、プロセッサのキューパイプラインまたは発行パイプラインにスロットル信号をアサートすることにより、さらなる処理を阻止してもよい。

【００１５】

前記システムは、さらに、さらなる命令の発行が阻止された後に消費される電力（消費電力）を求め、当該消費電力を、後続の命令サイクルの累積される電力に加算してもよい

。前記システムは、プロセッサにおけるイベントのさらなる処理を、当該イベントの推定される電力と先のイベントにおいてさらなる処理が阻止された後に消費された電力とに応じて阻止してもよい。

【００１６】

前記システムは、前記チップの接合温度を監視して、これにより当該チップの目下（現在）の温度を監視してもよい。

【００１７】

前述の内容は、添付の図面に示された本発明にかかるシステムについての以下の詳細な説明から明らかになる。なお、異なる図をとおして、同一の符号は同一の構成又は構成要素を指すものとする。図面は必ずしも縮尺どおりではなく、本発明の実施形態を示すことに重点を置いている。

【図面の簡単な説明】

【００１８】

【図１】電力を制御する本発明の例示的な一実施形態の概観図である。

【図２Ａ】イベント監視手段を用いる例示的な一実施形態における各種処理のハイレベル流れ図である。

【図２Ｂ】電力推定手段を用いる例示的な一実施形態における各種処理のハイレベル流れ図である。

【図２Ｃ】オーバーシュート電力推定手段を用いる例示的な一実施形態における各種処理のハイレベル流れ図である。

【図３Ａ】スロットル信号を印加することによってイベントのさらなる処理を阻止する例示的な一実施形態のハイレベル流れ図である。

【図３Ｂ】スロットル信号を印加することによってイベントのさらなる処理を阻止する例示的な一実施形態のハイレベル流れ図である。

【図３Ｃ】スロットル信号を印加することによってイベントのさらなる処理を阻止する例示的な一実施形態のハイレベル流れ図である。

【図４】電力推定手段のハイレベル流れ図である。

【図５】オーバーシュート電力推定手段を利用する電力累積手段のハイレベル流れ図である。

【図６】コンピュータチップの温度を監視する例示的な一実施形態のハイレベル流れ図である。

【発明を実施するための形態】

【００１９】

図１は、電力を制御する例示的な一実施形態の概観図である。電力スロットリング（電力抑制）論理部１１０は、プロセッサで発生する各種イベント１３０（イベントＡ、イベントＢなど）を監視し、消費される電力（消費電力）の量を推定し、必要に応じてスロットル信号１４０をアサートする（、つまり信号１４０を送信する）。前記イベントには、命令キャッシュ１６０、デコード／スロット部１６２、データキャッシュ／ローカルメモリ部１６４、コヒーレントメモリインターコネクト１６６、モジュール型乗算器１６８などの各種プロセッサモジュールにおける命令の発行または実行等のイベントが含まれる。

【００２０】

電力スロットリング論理部１１０は、消費電力を推定するにあたって、サイクル毎に消費電力を推定してもよい。サイクル毎の電力は、プロセッサに流れ込む電流を測定することによって取得してもよい。他方で、一部の例示的な実施形態では、ＣＭＯＳにおいて消費電力を下記の式とみなすことにより、電力を推定してもよい：

$$P = i_L v + v^2 c f$$

ここで、Ｐは消費電力を表し、i_Lはリーク電流（漏れ電流）であり、ｖは電圧であり、ｃはキャパシタンス（静電容量）であり、ｆはスイッチング周波数である。

【００２１】

リーク電流i_Lはほぼ一定の数値であるため、一般性を失わずに無視することができる

。項v^2cfは、電力スロットリング論理部１１０によって推定される動的電力（変動する電力）を表す。電圧ｖも一定の数値であるとみなすことにより、電力スロットリング論理部１１０は、動的電力の代わりに切替キャパシタンス（スイッチドキャパシタンス）ｃｆを推定（算出）してもよい。切替キャパシタンスｃｆは、所与の時間中にキャパシタンスが切り替わる回数（すなわち、キャパシタンスが０から１に遷移する回数）で、プロセッサにおける配線のキャパシタンスを乗算した数値を表す。

【００２２】

電力スロットリング論理部１１０は、損失電力／消費電力を推定するにあたって、プロセッサにおける各種重要なイベント１３０を特定する。一部の実施形態では、損失電力／消費電力の推定値を、各イベントが生じさせる切替キャパシタンスの量を測定することによって得てもよい。この測定値は、スイッチレベルトランジスタレベルシミュレータ（switch level transistor level simulator）を用いて得てもよい。プロセッサにおける各配線のキャパシタンス、ならびに配線が切り替わるタイミングおよび頻度に関する情報は、電力スロットリング論理部１１０が把握しているため、少数のイベントを監視するだけで、総切替キャパシタンス（その結果、消費電力）を効率的に推定することができる。

【００２３】

例示的な実施形態１００では、プロセッサで発生した各種イベント１３０について推定された電力を累積する、推定電力累積モジュール１２０が用いられる。「電力累積」とは、推定された電力値の累積（すなわち、電力情報の累積）を指す。累積される消費電力量を、プログラム可能な期間（例えば、時間枠、サイクル数など）中に累積される消費電力量とし、これを所定の閾値以下に制限してもよい。例示的な実施形態では、前記プログラム可能な期間中、変動する電力消費の推定値を累積して、この累積した電力を前記所定の閾値と繰り返して比較する。前記累積した電力が前記所定の閾値を超えると、電力制御論理部１１０がスロットル信号１４０をアサートし、さらなる処理を阻止する。電力制御論理部１１０は、１つ以上の命令サイクル中、および／または１つ以上の命令サイクルが経過するまで、および／または目下（現在）の命令サイクルが完了するまで、および／または所定時間中、さらなる処理を阻止してもよい。本明細書において「命令サイクル」とは、命令が処理または実行される期間のことを指す。

【００２４】

電力制御論理部１１０は、プロセッサのキュー／発行パイプラインステージ１５０にスロットル信号１４０をアサートして、さらなる命令の実行または発行を停止してもよい。一部の例示的な実施形態では、スロットル信号１４０がアサートされることによって動的電力の消費／損失が確実に完全停止するように、キュー／発行パイプラインステージ１５０以外のユニットに当該スロットル信号１４０をアサートしてもよい。例えば、図１に示すレジスタファイル１９１、パイプ０（シフト動作、ロード動作、ストア動作、および算術論理装置の動作に使用される）１９３、パイプ１（シフト動作、乗算動作、コピー動作、ブランチ動作、ジャンプ動作、および算術論理装置の動作に使用される）１９２、ジャンプ予測部１９４、ブランチ予測部１９５、トランスレーションルックアサイドバッファ（ＴＬＢ）１９６、ライトバッファ１９７、ミスファイルエントリ部（miss file entry unit）１９８などのユニットがスロットル信号を受け取ることにより、当該ユニットによるイベントのさらなる処理を阻止してもよい。

【００２５】

コアプロセッサパイプラインの様々な箇所で起こるイベントを監視してもよい。例えば、例示的な一実施形態では、パイプラインの様々なステージで起こる最大３０個の異なるイベント（例えば、目下（現在）のプロセッササイクル中に発行される命令、あるサイクル中に有効にされる条件付クロック、またはあるサイクル中にアクセスされるキャッシュメモリなど）を監視することにより、プロセッサコアによって消費される電力を正確に推定してもよい。電力スロットリング論理部１１０は、プロセッサコアがチップの動的エネルギの大半を消費するので、プロセッサコアを監視してもよい。例示的な実施形態では、コアの電力を制御または制限することができる。一部の実施形態では、電力制御に関係す

るパラメータが、電力スロットルレジスタ（図示せず）を用いてプログラムされる。
【００２６】
例示的な実施形態は、（必ずしもコンピュータチップまたはプロセッサでなくとも）電力を消費または損失するあらゆるユニットに対して用いることができ、電力が所定の閾値に達すると、さらなる電力の消費／損失を阻止する。例えば、例示的な実施形態を、通信、ネットワーキング、科学計算、グラフィックレンダリング等に使用される相補型金属酸化膜半導体（ＣＭＯＳ）に用いることにより、装置（デバイス）の性能の調節や、チップで消費される電力の制御を実行することができる。
【００２７】
一部の実施形態において、電力スロットリング論理部１１０は、閉ループまたは開ループで動作してもよい。開ループでの動作の場合、電力スロットリング論理部１１０は、時間間隔（時間枠）と、命令の発行が可能な時間率（％）とを指定する。これにより、電力スロットリング論理部１１０は、前記時間間隔において経過時間が前記指定された命令の発行が可能な時間率（％）を超えるまで、イベント／命令の実行／発行を許可する。
【００２８】
閉ループでの動作の場合、電力スロットリング論理部１１０に、最大許容電力が入力される。電力スロットリング論理部１１０は、ある時間間隔（時間枠）において、累積される電力が前記最大許容電力を超えるまで、命令の発行を許可する。前記最大許容電力に達すると、電力スロットリング論理部１１０は、スロットル信号１４０をアサートして、前記時間間隔が経過するまでさらなる命令の発行／実行を阻止する。
【００２９】
電力スロットリング論理部１１０は、さらなる命令の発行／実行が阻止された後に消費電力（例えば、オーバーシュート電力など）を求めてもよい。このさらなる電力値は記憶され、後続の時間期間に累積される電力に加算されてもよい。
【００３０】
電力スロットリング論理部１１０のソフトウェアインターフェースは、下記の数値を用いて動作し得る：
●　電力スロットリング最大値（ＭＡＸＴＨＲ）：これは、許容される最大のスロットリング値を示す。この数値は、所定の数値の形式でコアまたは他のモジュールに供給されてもよく、あるいは、コアまたは他のモジュールによって決定されてもよい。
●　電力スロットリング最小値（ＭＩＮＴＨＲ）：これは、許容される最小のスロットリング値を示す。
●　現在の、コアによる電力消費についての推定値（ＰＯＷＥＲ）：この返却される電力値は、平均消費電力を示す。この平均消費電力は、指数関数的に重み付けされてもよい。
●　先の期間で使用されたスロットリング値（ＴＨＲＯＴＴ）：スロットリングにより、ＣＰＵコアは一定時間中、強制的にアイドル状態になり、電力消費を減少させる。スロットリング値がゼロに設定されると、コアは大半の時間中アイドル状態になり、最低限の電力のみを消費する。前記スロットリング値は、コアが電力を減少させるためのアイドル状態に全くならないことを示す数値に設定されてもよい。前記スロットリング値を増加させると、強制的なアイドル時間が線形的に減少する。アイドル状態はサイクル期間の終わりの方で強制的に生じ、当該期間の始めの方では生じない。一般的に、ハードウェアは、性能を最大限に引き出すにあたって、一定の制約はあるものの、出来る限り大きいスロットル値を使用する。
●　コアの電力リミッタ（電力制限）値（ＴＡＲＧＥＴ）が使用されてもよい。この数値は、コアの電力消費の概算的なリミッタを表す。このリミッタ電力（上限電力）は概算値であり、閉ループフィードバックシステムにおいて電力の概算に利用される。
【００３１】
以下に、前記電力スロットリング論理部によって使用され得るアルゴリズム擬似コードの一例を示す：

【００３２】
【数１】

```
while(true){
        accumulated_power = 0;
        Issue_stall = 0;
        interval = 1024>>period;
        for (cycle=0; cycle<interval; cycle++)
        {
            accumulated_power += estimated power;
            if ((cycle/interval > MINTHR/256) && …
            (cycle/interval > …
            MAXTHR/256 ||
            accumulated_power+overshoot>=TARGET*interval))
            {
                if (!Issue_stall) THROTT=cycle;
                Issue_stall = 1;
            }
        }
        overshoot = overshoot + accumulated_power/interval -target;
        if (overshoot<0) overshoot=0;
        if (!Issue_stall) THROTT=255;
        POWER = POWER*255/256 + accumulated_power;
    }
```

【００３３】
上記の擬似コードにおいて、「accumulated_power」は累積される電力情報の量を示し、「interval」は監視期間（例えば、時間間隔（時間枠）など）を示し、「estimated power」は目下の期間において推定される電力情報を示し、「overshoot」は推定されるオーバーシュート電力の量を示し、「THROTT」はスロットリング値を示す。
【００３４】
一部の例示的な実施形態は、コンピュータチップの接合温度を（外部回路に支援されながら）監視するのに用いられるサーマルダイオードと、チップ上の個々のユニット／構成の電力使用を動的に制御する構成とを備える。これにより、この例示的な実施形態では、サーマルダイオードを用いてチップの温度を監視し、必要に応じてチップの電力消費を減少させて熱的過負荷を防ぐことができるため、厳しく熱出力を制限しながら最大の性能を達成することができる。
【００３５】
前記ダイオードは、チップの接合温度を監視するように当該チップに設けられてもよい。一部の実施形態（例えば、ＣａｖｉｕｍＯＣＴＥＯＮ ６３ｘｘ；Ｃａｖｉｕｍ Ｎｅｔｗｏｒｋｓ（米国カリフォルニア州マウンテンビュー市）社製）では、ダイオードが、ＴＨＥＲＭＡＬ＿Ｄ＿Ｎ／ＴＨＥＲＭＡＬ＿Ｄ＿Ｌチップピンに取り付けられることができる。ダイオードを制御および感知するための外部装置が使用されてもよい。この外部装置は、ＴＨＥＲＭＡＬ＊ピンおよびＯＣＴＥＯＮＴＷＳＩインターフェースに接続される。ＴＷＳＩインターフェースに接続されることにより、ソフトウェアがアナログ温度センサ

チップであるＴＭＰ４２１を設定して計測温度を監視するための経路が構成される。ＴＭＰ４２１は、１秒間あたり８回の頻度で温度をサンプリングできる。

【００３６】

チップの電力消費は、電源電圧、周波数、温度（例えば、周囲温度、外気温度など）、チップのワークロード（処理量）等の複数の変数による関数である。電力を、ワークロードを変化させることによって減少させてもよい。これは、システム及びチップの複雑性の軽減につながる。ワークロードを変化させることで性能が低下する場合もあるが、例示的な実施形態では、性能を抑制することにより、過剰な電力消費／電力損失によって起こり得るチップの損傷を防止する。

【００３７】

前記Ｃａｖｉｕｍ ＯＣＴＥＯＮ ６３ｘｘの６個の中央処理装置（ＣＰＵ）コアによって使用される動的電力は、当該ＯＣＴＥＯＮ ６３ｘｘチップの総電力の約３０ないし約５０％である。この総電力には、静的漏れ及び動的コア（コアの数が変化すること）を含めた電源によってチップで消費される電力、プロセッサコアの電力、ならびに入力／出力（Ｉ／Ｏ）の電力が含まれる。この総電力および電力減少量は、動作状態によって異なり、また、チップ間の製造の違いによってチップ毎で異なる。

【００３８】

一部の実施形態では、ＣＰＵコアの動的電力を推定および制御するにあたって、条件付クロック動作を用いてもよい。具体的に述べると、前記ＯＣＴＥＯＮのＣＰＵコアは、積極的に、論理回路に対して条件付クロックを供給してもよい。ＣＰＵコアがアイドル状態であれば、ハードウェアがコアのクロックを停止してもよい。例えば、キャッシュミスのためにメモリにアクセスすることでコアが待機しており、他のタスクが実行されていない場合、コアのクロックを停止させてもよい。一般的に、ＣＰＵの処理量とコアの電力消費とは正比例する。

【００３９】

ＣＰＵコアの動的電力を減少させる最も簡易な方法は、ＣＰＵコアを無効にする（使用不可にする）こと、つまりコアの使用を停止することである。ＣａｖｉｕｍＯＣＴＥＯＮファミリーのチップにおけるＣＰＵコアは、標準的なＭＩＰＳ ＷＡＩＴ命令を実行中のアイドル状態であるときに、電力消費が最小限となる。ＷＡＩＴ命令により、ソフトウェアは、割込みが生じるまでＣＰＵコアをストール（自動待機）することができる。前記ＯＣＴＥＯＮのハードウェアは、割込みを待っている間、ＣＰＵコアのほぼ全ての条件付クロックを停止させる。割込みが生じると、ＣＰＵコアは、待機後の命令（または割込みハンドラ）の実行を続ける。前記ＯＣＴＥＯＮのＣＰＵコアは、ＷＡＩＴ実行中のアイドル状態よりもリセット状態である方が、電力消費が若干大きい。

【００４０】

多くの場合、ＣＰＵコアの無効（使用不可）はソフトウェアにとって大部分が不可視化された動作（例えば、Ｌｉｎｕｘ（登録商標）におけるＷＡＩＴ命令のアイドルループ使用）であるが、このような動作は、一般的にソフトウェアによって編成される必要がある。無効化されたコアを動作から強制的に外して、当該コアで実行されていたタスクを、別のコアで実行する必要が生じ得る。

【００４１】

ＣＰＵコアを無効にする電力的な利点は、そのコアがもし無効にされていなかった場合に何をしていたかに依存する。コアが全力で冪剰余計算を実行していたなら、最大の節約が得られるであろう。ＣＰＵコアが、処理すべき命令を待つ間にストールされるように通常設定されている場合、当該コアの電力は既に小さいので、得られる節約も小さくなる。

【００４２】

前記ＯＣＴＥＯＮ ６３ｘｘのＣＰＵコア内部における電力消費の大きいユニットは、ロード／ストアユニットおよびサイズの大きい乗算ユニットの２つである。前記冪剰余計算のアプリケーションは、これら両方のユニットを大いに利用する傾向にあるため、電力消費も大きくなる。ＣＰＵコア内部の他のユニットは、電力消費が小さい。電力消費の大

きいユニットを全く使用しないか、またはそのようなユニットの使用頻度を少なくすることにより、コアのピーク電力消費を減少させることができる。これは、前記ＯＣＴＥＯＮ６３ｘｘも含め、全てのＯＣＴＥＯＮのコアに当てはまる。

【００４３】

ＯＣＴＥＯＮのＣＰＵコア内部の機能ユニットは、ソフトウェアまたはハードウェアによって無効にする（使用不可にする）ことができる。このようなアプリケーションのためのソフトウェアは、問題のあるユニットを使用する命令の発行を避けるために、ハードウェアによる当該ユニットへのクロックの供給を、条件付クロックによって防いでもよい。

【００４４】

各ＣＰＵの設定条件レジスタ（ＣＳＲ）フィールドを用いて、コアが消費してもよい最大の動的電力を指定してもよい。ＣＳＲは、制約されていない場合に、コアの電力消費が限度未満であれば、コアの挙動に影響を与えない。しかし、コアで実行されているソフトウェアによる消費電力がＣＳＲの限度を超えると、ＣＰＵコアのハードウェアは、当該コアの電力消費が指定の限度未満に減少するまでストール命令を挿入することで対応する。一部の実施形態では、ＣＳＲは最大で２５６個の異なるスロットル値を提供し得る。これらの数値は、コアの動的電力の極めて小さい範囲から最大の範囲までを線形的にカバーするように設定されてもよい。

【００４５】

コアにおける所与のＣＳＲ設定による的確な論理的効果および性能的効果は、当該コアで実行されるソフトウェアアプリケーションに依存する。コアが多大な電力を消費している（例えば、冪剰余計算等の電力消費の大きいアプリケーションが実行されている）際、ＣＳＲ限度によっては、ＣＰＵコアの性能が大いに低下する可能性がある。ハードウェアによって、性能の低下を抑えながら、コアによって消費される最大電力を制限することができる。

【００４６】

ソフトウェアにより、コア毎にＣＳＲ設定を変更することが可能である。遠隔装置により、これらのＣＳＲ設定を直接変更してもよい。このようなＣＳＲ変更は、任意の時点で行うことができる。前記ＯＣＴＥＯＮ６３ｘｘチップの電力消費を制御するために、システムが、各コアのＣＳＲ設定を、任意でコア毎に個別に、動的に調節してもよい。

【００４７】

一部の実施形態では、ソフトウェアにより、システムの温度を監視し、必要に応じて電力を減少させてもよい。例えば、前記ＯＣＴＥＯＮ６３ｘｘのサーマルダイオードを、外部回路と共に使用してもよい。この外部回路は、前記ＯＣＴＥＯＮ ６３ｘｘの温度（ＴＷＳＩ）インターフェースに直接取り付けられてもよい。この場合、ソフトウェアが、ＴＷＳＩ読出しを定期的に取得することにより、温度を定期的にサンプリングできる。

【００４８】

前記ＯＣＴＥＯＮ ６３ｘｘは、温度が閾値を超えない限り完全動作してもよく、温度が閾値を超えると、システムのソフトウェアが、当該ＯＣＴＥＯＮの電力を減少させる必要があると判断してもよい。前記ＯＣＴＥＯＮ６３ｘｘのＣＰＵコアの電力を、電力減少の対象としてもよい。前記ソフトウェアにより、前記ＯＣＴＥＯＮ ６３ｘｘの電力を減少させることを判断し、かつ、当該ＯＣＴＥＯＮ ６３ｘｘの一部または全部のＣＰＵコアにおけるスロットリングＣＳＲを調節することで、電力の減少を達成してもよい。残りのソフトウェアについては、性能の低下を検出するが、この性能低下検出がなければ電力の減少には気付かなくてもよい。温度監視用のソフトウェアが温度の減少を認識すると、当該ソフトウェアはコアのフル電力動作を復元することができる。

【００４９】

温度監視用のソフトウェアの詳細な挙動は、システムの具体的な特徴に依存する。

【００５０】

図２Ａは、イベント監視手段２４０を用いる例示的な一実施形態における各種処理のハイレベル流れ図２００である。この例示的な実施形態は、プロセッサ２２０における電力

消費および電力損失を制御する電力制御モジュール２１０を備える。電力制御モジュール２１０は、プロセッサで生じる重要なイベント２３０－１，…，２３０－ｎを監視するイベント監視手段２４０を含む。イベント２３０－１，…，２３０－ｎは、レジスタ２２５およびキャッシュ２２７等のプロセッサモジュールで発行または実行される命令を含んでもよい。イベント監視手段２４０はこれらのイベント２３０－１，…，２３０－ｎを監視し、電力累積手段１２０が当該イベント２３０－１，…，２３０－ｎに伴う推定される電力を累積する。この推定される電力は、プロセッサに流れ込む電流を測定すること、および／または１つ以上のイベントのうちの各イベントで構成されるキャパシタンス情報を抽出したトランジスタレベルシミュレータを用いることも含め、当該技術分野で知られた各種手法を用いて得ることができる。あるいは、この推定される電力は、別のモジュールを用いて電力累積手段１２０に供給されてもよい。また、この推定される電力は、既知のプロセッサイベントの推定される電力値についてのデータベースから得られてもよい。

【００５１】

比較回路２５０が、累積した電力を所定の閾値と比較する。この所定の閾値は、ソフトウェアによって課されたもの、ハードウェアによって設定されたもの、またはユーザによって比較回路２５０に供給されたものであってもよい。この所定の閾値は、プロセッサの設計要件およびシステム要件に基づいて決定されてもよい。この所定の閾値は、チップヒューズによって設定されてもよい。比較回路２５０は、累積した電力を所定の閾値と比較し続ける。累積した電力が所定の閾値を超えると、スロットル信号１４０がプロセッサの発行／キューのパイプライン１５０にアサートされ、さらなる命令の発行または処理が阻止される。一部の実施形態では、スロットル信号１４０は、プロセッサの他の要素にアサートされてもよい。

【００５２】

図２Ｂは、電力推定手段２４２を用いる例示的な一実施形態における各種処理のハイレベル流れ図２０１である。図２Ａに示した実施形態と同様に、この例示的な実施形態は、プロセッサ２２０における電力消費および電力損失を制御する電力制御モジュール２１０を備える。電力制御モジュール２１０は、プロセッサで生じるイベント２３０－１，…，２３０－ｎの実行／発行による消費電力／損失電力を推定する電力推定手段２４２を含む。イベント２３０－１，…，２３０－ｎは、レジスタ２２５およびキャッシュ２２７等のプロセッサモジュールで発行または実行される命令を含んでもよい。電力推定手段２４２はこれらのイベント２３０－１，…，２３０－ｎを監視し、電力累積手段１２０が当該イベント２３０－１，…，２３０－ｎに伴う推定される電力を累積する。

【００５３】

電力スロットリング論理部１１０は、消費電力を推定するにあたって、サイクル毎に消費される電力を推定してもよい。サイクル毎の推定は、プロセッサに流れ込む電流を測定することによって実行されてもよい。一部の例示的な実施形態では、電力を推定するにあたって、動的電力（変動する電力）と、プロセッサにおける配線のキャパシタンスを所与の時間中にキャパシタンスが切り替わる回数（すなわち、キャパシタンスが０から１に遷移する回数）で乗算したものとを推定することにより、チップで消費される電力を考慮してもよい。また、プロセッサにおける重要なイベント１３０を特定し、各イベントが構成する切替キャパシタンスの量をスイッチレベルトランジスタレベルシミュレータを用いて測定することにより、損失電力／消費電力を推定してもよい。

【００５４】

例示的な実施形態２０１は、イベント２３０－１，…，２３０－ｎの推定される電力を累積する電力累積手段１２０を用いる。累積される消費電力は、プログラム可能な期間（例えば、時間枠、サイクル数など）中に累積される消費電力とし、これを所定の閾値以下に制限してもよい。

【００５５】

比較回路２５０が、累積した電力を所定の閾値と比較する。累積した電力が所定の閾値を超えると、制御回路２６０がスロットル信号１４０をプロセッサの発行／キューのパイ

プライン１５０（または他のモジュール）に対してアサートし、さらなる命令の発行または処理が阻止される。

【００５６】

制御回路２６０は、さらなる命令の発行または処理を阻止することにより、アプリケーションの性能を抑制し、電力の目標レベル（すなわち、所定の閾値で表されるレベル）を基本的に維持する。

【００５７】

図２Ｃは、オーバーシュート電力推定手段を用いる例示的な一実施形態２０２における各種処理のハイレベル流れ図である。

【００５８】

図２Ａおよび図２Ｂに示す実施形態と同様に、この例示的な実施形態２０２は、プロセッサ２２０における電力消費および電力損失を制御する電力制御モジュール２１０を備える。電力制御モジュール２１０は、プロセッサで生じるイベント２３０－１，…，２３０－ｎの実行／発行が阻止された後に消費電力を推定するオーバーシュート電力推定手段２４４を含む。この消費電力は、電力制御モジュール２１０に記憶され（場合によっては、記憶モジュール（図示せず）に記憶され）、次の電力累積サイクルで、累積される電力に加算される。このようにして、次の電力制御サイクルでは、累積される電力と、記憶されたオーバーシュート電力との合計値が所定の閾値を超えると、制御回路２６０が電力スロットリング信号１１０をアサートし、プロセッサで生じるイベント２３０－１，…，２３０－ｎの発行または実行を阻止する。

【００５９】

消費電力の推定は、チップに流れ込む電流を考慮するか、またはプロセッサにおける配線のキャパシタンスを所与の時間中にキャパシタンスが切り替わる回数（すなわち、キャパシタンスが０から１に遷移する回数）で乗算したものを考慮することにより行ってもよい。

【００６０】

図３Ａ～図３Ｃは、スロットル信号を印加することによってイベントのさらなる処理を阻止する例示的な実施形態のハイレベル流れ図である。図３Ａでは、１つの命令サイクル３０１の間、複数の命令を発行することによって消費電力が累積される（各命令は、２５６個の命令サイクルを含むものと仮定する）。累積した消費電力が閾値と比較される（図２Ａ～図２Ｃ）。累積した電力が閾値を超えると、スロットル信号１４０がアサートされる。図３Ａに示す例示的な実施形態３００では、スロットル信号１４０により、その命令サイクルにおける残りの命令の発行または実行が阻止される。例示的な実施形態３００では、次の命令サイクルについての命令の発行／実行を再開する。

【００６１】

図３Ｂに示す例示的な実施形態３０１では、スロットル信号１４０がアサートされると、所与の時間中３１０に命令の発行または実行が阻止される。例示的な実施形態３０１では、時間３１０が経過すると、次の命令サイクルについての命令の発行／実行を再開する。

【００６２】

図３Ｃに示す実施形態３０２では、スロットル信号１４０がアサートされると、特定の数の命令３２０の間、命令の発行または実行が阻止される（この例では、命令２００～２４０の処理が阻止される）。例示的な実施形態３０２では、選択された数の命令の処理が阻止された後、命令の発行／実行を再開する。

【００６３】

図４は、電力推定手段２４２のハイレベル流れ図４００である。電力推定手段２４２は、プロセッサ４１０に流れ込む電流の測定値を用いて、消費電力／損失電力の推定値４５０を得てもよい。電力推定手段２４２は、１つ以上のイベントのうちの各イベントによってプロセッサで構成される切替キャパシタンス４２０の測定値に基づいて、電力の推定値４５０を得てもよい。電力推定手段２４２は、命令クロックサイクルカウンタ４３０に基

づいて、電力の推定値４５０を求めてもよい。電力推定手段２４２は、既知のプロセッサイベントに伴う電力についてのデータベース４４０から電力の推定値４５０を求めてもよい。

【００６４】

図５は、オーバーシュート電力推定手段２４４を利用する電力累積手段１２０のハイレベル流れ図５００である。命令サイクル５４０の間、命令を発行することによって消費電力が累積される（各命令は、２５６個の命令を含むものと仮定する）。累積した消費電力が閾値と比較される。累積した電力が閾値を超えると、スロットル信号１４０がアサートされる。スロットル信号１４０がアサートされると、オーバーシュート電力推定手段２４４が、命令の実行／発行が阻止された後に消費電力を推定する。この消費されるオーバーシュート電力は記憶され、次の電力累積サイクルで、累積される電力に加算される。

【００６５】

スロットル信号１４０が、命令サイクルにおける残りの命令の発行または実行を阻止する。例示的な実施形態５００では、次の命令サイクルについての命令の発行／実行を再開する。電力累積手段１２０は、その命令サイクルの間消費電力を累積し、直前の命令サイクルからのオーバーシュート電力５１０をその命令サイクルの累積した電力に加算する。

【００６６】

図６に、コンピュータチップ６２０の温度を減少させる本発明にかかる一実施形態６００を示す。例示的な実施形態６００は、チップ６２０の接合温度、チップ６２０のダイオードピン６８９、およびチップ６２０の二線式シリアルインターフェースピン（ｔｗｓｉ）６８８を監視することによってコンピュータチップ６２０の温度を監視する温度センサ６９０を備えてもよい。コンピュータチップの温度は、イベントを処理する間にチップで損失される電力／消費される電力、周囲温度、およびチップの周りを流れる空気の量の関数としてもよい。例示的な実施形態６００は、センサ６９０によって感知されたコンピュータチップ６２０の温度を追跡するソフトウェア温度制御モジュール６１０を備えてもよい。チップ６２０の温度が所定の閾値（例えば、予め設定された上限温度）を超えると、温度監視モジュール６１０が情報６６０を性能抑制モジュール６４０に通信し、性能抑制モジュール６４０がコンピュータチップ６２０におけるイベントの処理を阻止することによってチップ６２０の温度を低下させる。性能抑制モジュール６４０が信号６７０をチップのイベント発行・処理モジュール６７５にアサートし、イベント発行・処理モジュール６７５がさらなるイベントの処理される速度を減少させることによって性能を抑制してもよい。一部の実施形態では、処理を阻止するための信号６７０を、チップ６２０のキュー／発行のパイプライン（図示せず）におけるさらなる命令の処理を阻止することによって性能を抑制するのに用いてもよい。

【００６７】

なお、流れ図もしくはブロック図で示した各種処理、およびそれ以外の本明細書に記載した各種処理は、ハードウェア、ファームウェアおよびソフトウェアのいずれの形態で実施されてもよい。ソフトウェアで実施される場合には、当該ソフトウェアは、本明細書の教示内容に合った任意のソフトウェア言語であれば、いずれのソフトウェア言語で記述されたものであってもよく、また、当該技術分野において既知であるかまたは将来開発され得る任意のコンピュータ読取り可能な媒体に記憶されたものであってもよい。典型的な命令の形態である当該ソフトウェアは、当該技術分野で把握されている様式でプロセッサによってコード化かつ実行され得る。

【００６８】

以上のとおり、システムに基づいて本発明を具体的に図示し詳細に説明した。しかし、当業者であれば、添付の特許請求の範囲に包含される本発明の範囲を逸脱することなく形態および細部に対して様々な変更が可能であることを理解するであろう。

なお、本発明は、実施の態様として以下の内容を含む。

〔態様１〕

コンピュータチップの電力を制御する方法であって、

命令を発行する間に消費された電力に関する電力情報を累積する過程と、
前記累積した電力情報を所定の閾値と比較する過程と、
前記累積した電力情報が前記所定の閾値よりも大きい場合、前記コンピュータチップの電力消費を減少させるように、さらなる命令の発行を阻止する過程とを備えた、コンピュータチップ電力制御方法。
〔態様２〕
態様１に記載のコンピュータチップ電力制御方法において、さらに、
命令を発行する間に消費された電力に関する前記電力情報を、所定の時間中に累積する過程を備えた、コンピュータチップ電力制御方法。
〔態様３〕
態様１に記載のコンピュータチップ電力制御方法において、さらに、
命令を発行する間に消費された電力に関する前記電力情報を、発行される各命令に伴う電力を推定して累積する過程を備えた、コンピュータチップ電力制御方法。
〔態様４〕
態様３に記載のコンピュータチップ電力制御方法において、さらに、
前記発行される各命令に伴う電力を、既知の命令についての電力値についてのデータベースを用いて推定する過程を備えた、コンピュータチップ電力制御方法。
〔態様５〕
態様３に記載のコンピュータチップ電力制御方法において、さらに、
前記発行される各命令に伴う電力を、前記各命令の処理の間に経過するクロックサイクル数をそれぞれ求めて推定する過程を備えた、コンピュータチップ電力制御方法。
〔態様６〕
態様１に記載のコンピュータチップ電力制御方法において、さらに、
命令を発行する間に消費された電力に関する前記電力情報を、１つ以上の命令サイクル中に累積する過程を備えた、コンピュータチップ電力制御方法。
〔態様７〕
態様６に記載のコンピュータチップ電力制御方法において、さらに、
１つ以上の命令サイクルが経過するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
〔態様８〕
態様６に記載のコンピュータチップ電力制御方法において、さらに、
現在のスロットル期間が完了するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
〔態様９〕
態様６に記載のコンピュータチップ電力制御方法において、さらに、
所定の数の命令サイクルが経過するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
〔態様１０〕
態様１に記載のコンピュータチップ電力制御方法において、さらに、
所定の時間が経過するまで、さらなる命令の発行を阻止する過程を備えた、コンピュータチップ電力制御方法。
〔態様１１〕
態様１に記載のコンピュータチップ電力制御方法において、さらに、
前記所定の閾値を、前記コンピュータチップの最大許容電力との相関関係において決定する過程を備えた、コンピュータチップ電力制御方法。
〔態様１２〕
態様１に記載のコンピュータチップ電力制御方法において、さらに、
前記所定の閾値を、１つ以上のチップヒューズとのとの相関関係において決定する過程を備えた、コンピュータチップ電力制御方法。
〔態様１３〕

態様１に記載のコンピュータチップ電力制御方法において、さらに、
さらなる命令の発行が阻止された後に消費された電力を求める過程と、
当該消費された電力を、後続の命令サイクルの累積される電力に加算する過程とを備えた、コンピュータチップ電力制御方法。
〔態様１４〕
命令を発行する間に消費された電力に関する電力情報を累積する電力累積手段と、
前記累積した電力情報を所定の閾値と比較する比較部と、
前記累積した電力情報が前記所定の閾値よりも大きい場合、電力消費を減少させるように、さらなる命令の発行を阻止する制御部とを備えた、プロセッサ。
〔態様１５〕
態様１４に記載のプロセッサにおいて、前記電力累積手段が、命令を発行する間に消費された電力に関する前記電力情報を、所定の時間中に累積する、プロセッサ。
〔態様１６〕
態様１４に記載のプロセッサにおいて、前記電力累積手段が、命令を発行する間に消費された電力に関する前記消費電力情報を、発行される各命令に伴う電力を推定して累積する、プロセッサ。
〔態様１７〕
態様１６に記載のプロセッサにおいて、さらに、
前記発行される各命令に伴う電力を、既知の命令に伴う電力値についてのデータベースを用いて推定する電力推定手段を備えた、プロセッサ。
〔態様１８〕
態様１４に記載のプロセッサにおいて、さらに、
前記発行される各命令に伴う電力を、前記各命令の処理の間に経過するクロックサイクル数をそれぞれ求めて推定する電力推定手段を備えた、プロセッサ。
〔態様１９〕
態様１４に記載のプロセッサにおいて、前記電力累積手段が、命令を発行する間に消費された電力に関する前記電力情報を、１つ以上の命令サイクル中に累積する、プロセッサ。
〔態様２０〕
態様１９に記載のプロセッサにおいて、前記制御部が、１つ以上の命令サイクルが経過するまで、さらなる命令の発行を阻止する、プロセッサ。
〔態様２１〕
態様１９に記載のプロセッサにおいて、前記制御部が、現在のスロットル期間が完了するまで、さらなる命令の発行を阻止する、プロセッサ。
〔態様２２〕
態様１９に記載のプロセッサにおいて、前記制御部が、所定の数の命令サイクルが経過するまで、さらなる命令の発行を阻止する、プロセッサ。
〔態様２３〕
態様１４に記載のプロセッサにおいて、前記制御部が、所定の時間間隔が経過するまで、さらなる命令の発行を阻止する、プロセッサ。
〔態様２４〕
態様１４に記載のプロセッサにおいて、前記比較部が、前記所定の閾値を、前記コンピュータチップの最大許容電力との相関関係において決定する、プロセッサ。
〔態様２５〕
態様１４に記載のプロセッサにおいて、前記比較部が、前記所定の閾値を、１つ以上のチップヒューズとの相関関係において決定する、プロセッサ。
〔態様２６〕
態様１４に記載のプロセッサにおいて、さらに、
さらなる命令の発行が阻止された後に消費された電力を推定するオーバーシュート電力推定手段を備え、当該消費された電力が、後続の命令サイクルの累積される電力に加算さ

れる、プロセッサ。
〔態様２７〕
プロセッサの電力を制御する方法であって、
１つ以上のイベントを処理する間に前記プロセッサによって消費された電力を推定する過程と、
電力消費を減少させるために、推定された電力との相関関係において、前記プロセッサのさらなる処理を阻止する過程とを備えた、プロセッサ電力制御方法。
〔態様２８〕
態様２７に記載のプロセッサ電力制御方法において、さらに、
前記プロセッサによって消費された電力を、前記プロセッサに流れ込む電流との相関関係において推定する過程を備えた、プロセッサ電力制御方法。
〔態様２９〕
態様２７に記載のプロセッサ電力制御方法において、さらに、
前記プロセッサによって消費された電力を、前記１つ以上のイベントにおける各イベントによって前記プロセッサで構成される切替キャパシタンスとの相関関係において推定する過程を備えた、プロセッサ電力制御方法。
〔態様３０〕
態様２７に記載のプロセッサ電力制御方法において、さらに、
前記推定した電力を、所定時間および所定のイベントサイクル数のうちの少なくとも一方の間に累積する過程を備えた、プロセッサ電力制御方法。
〔態様３１〕
態様３０に記載のプロセッサ電力制御方法において、さらに、
前記累積した電力が所定の閾値よりも大きい場合、前記プロセッサのさらなる処理を阻止する過程を備えた、プロセッサ電力制御方法。
〔態様３２〕
態様３０に記載のプロセッサ電力制御方法において、前記１つ以上のイベントが、イベントサイクル中に前記プロセッサによって発行される命令、イベントサイクル中に有効にされる条件付クロック、およびイベントサイクル中にアクセスされるキャッシュのうちの少なくとも１つを備えた、プロセッサ電力制御方法。
〔態様３３〕
態様２７に記載のプロセッサ電力制御方法において、さらに、
前記プロセッサのキューまたは発行のパイプラインにスロットル信号を発行して、さらなる処理を阻止する過程を備えた、プロセッサ電力制御方法。
〔態様３４〕
態様２７に記載のプロセッサ電力制御方法において、さらに、
さらなる処理が阻止された後に消費された電力を推定する過程を備えた、プロセッサ電力制御方法。
〔態様３５〕
態様３４に記載のプロセッサ電力制御方法において、さらに、
前記プロセッサにおけるイベントのさらなる処理を、当該イベントの前記推定された電力と、先のイベントにおいてさらなる処理が阻止された後に消費された電力との相関関係において阻止する過程を備えた、プロセッサ電力制御方法。
〔態様３６〕
プロセッサであって、
１つ以上のイベントを処理する間に当該プロセッサによって消費された電力を推定する電力推定手段と、
電力消費を減少させるために、推定された電力との相関関係において、当該プロセッサのさらなる処理を阻止する電力制御部とを備えた、プロセッサ。
〔態様３７〕
態様３６に記載のプロセッサにおいて、前記電力推定手段が、当該プロセッサによって

消費された前記電力を、当該プロセッサに流れ込む電流との相関関係において推定する、プロセッサ。
〔態様３８〕
態様３６に記載のプロセッサにおいて、前記電力推定手段が、当該プロセッサによって消費された前記電力を、前記１つ以上のイベントにおける各イベントによって当該プロセッサで構成される切替キャパシタンスとの相関関係において推定する、プロセッサ。
〔態様３９〕
態様３６に記載のプロセッサにおいて、さらに、
前記推定した電力を、所定の時間および所定のイベントサイクル数のうちの少なくとも一方の間に累積する電力累積手段を備えた、プロセッサ。
〔態様４０〕
態様３９に記載のプロセッサにおいて、前記電力制御部が、前記累積した電力が所定の閾値よりも大きい場合、当該プロセッサのさらなる処理を阻止する、プロセッサ。
〔態様４１〕
態様３９に記載のプロセッサにおいて、前記１つ以上のイベントが、イベントサイクル中に当該プロセッサによって発行される命令、イベントサイクル中に有効にされる条件付クロック、およびイベントサイクル中にアクセスされるキャッシュのうちの少なくとも１つを含む、プロセッサ。
〔態様４２〕
態様３６に記載のプロセッサにおいて、前記電力制御部が、当該プロセッサのキューまたは発行のパイプラインにスロットル信号を発行して、さらなる処理を阻止する、プロセッサ。
〔態様４３〕
態様３６に記載のプロセッサにおいて、さらに、
さらなる処理が阻止された後に消費された電力を推定するオーバーシュート電力推定手段を備えた、プロセッサ。
〔態様４４〕
態様４３に記載のプロセッサにおいて、前記電力制御部が、当該プロセッサにおけるイベントのさらなる処理を、当該イベントの推定された電力と、先のイベントにおいてさらなる処理が阻止された後に消費された電力との相関関係において阻止する、プロセッサ。
〔態様４５〕
電力消費を制御する方法であって、
１つ以上のタスクを実行する間に消費された電力であって、あるユニットで消費された電力に関する電力情報を累積する過程と、
前記累積した電力情報が所定の閾値よりも大きい場合、前記１つ以上のタスクのさらなる実行を阻止することによって、前記ユニットの電力消費を制御する過程とを備えた、電力消費制御方法。
〔態様４６〕
態様４５に記載の電力消費制御方法において、さらに、
各タスクを実行する間に消費された電力を推定して、前記電力情報を累積する過程を備えた、電力消費制御方法。
〔態様４７〕
態様４６に記載の電力消費制御方法において、さらに、
各タスクに伴う既知の電力値についてのデータベースに基づいて、前記消費された電力を推定する過程を備えた、電力消費制御方法。
〔態様４８〕
態様４５に記載の電力消費制御方法において、さらに、
所定時間および所定のタスク数のうちの少なくとも一方の間、前記ユニットの前記電力情報を累積する過程を備えた、電力消費制御方法。
〔態様４９〕

態様４５に記載の電力消費制御方法において、さらに、
前記ユニットの実行パイプラインにスロットル信号を発行して、前記１つ以上のタスクのさらなる実行を阻止する過程を備えた、電力消費制御方法。
〔態様５０〕
態様４５に記載の電力消費制御方法において、さらに、
前記１つ以上のタスクのさらなる実行が阻止された後に消費された電力に関する情報を累積する過程を備えた、電力消費制御方法。
〔態様５１〕
態様５０に記載の電力消費制御方法において、さらに、
前記累積した電力情報と、消費された電力の情報との合計が前記所定の閾値よりも大きい場合、前記１つ以上のタスクのさらなる実行を阻止する過程を備えた、電力消費制御方法。
〔態様５２〕
電力消費を制御する装置であって、
１つ以上のタスクを実行する間に消費された電力であって、あるユニットで消費された電力に関する電力情報を累積する電力累積手段と、
前記累積した電力情報が所定の閾値よりも大きい場合、前記１つ以上のタスクのさらなる実行を阻止することによって、前記ユニットの電力消費を制御する電力制御手段とを備えた、電力消費制御装置。
〔態様５３〕
態様５２に記載の電力消費制御装置において、前記電力累積手段が、各タスクを実行する間に消費された電力を推定して、前記電力情報を累積する、電力消費制御装置。
〔態様５４〕
態様５３に記載の電力消費制御装置において、前記電力累積手段が、各タスクに伴う既知の電力値についてのデータベースに基づいて、前記消費された電力を推定する電力推定手段を有する、電力消費制御装置。
〔態様５５〕
態様５２に記載の電力消費制御装置において、前記電力累積手段が、所定時間および所定のタスク数のうちの少なくとも一方の間、前記ユニットの前記電力情報を累積する、電力消費制御装置。
〔態様５６〕
態様５２に記載の電力消費制御装置において、前記電力制御手段が、前記ユニットの実行パイプラインにスロットル信号を発行して、前記１つ以上のタスクのさらなる実行を阻止する、電力消費制御装置。
〔態様５７〕
態様５２に記載の電力消費制御装置において、前記電力累積手段が、さらに、前記１つ以上のタスクのさらなる実行が阻止された後に消費された電力に関する電力情報を累積する、電力消費制御装置。
〔態様５８〕
態様５２に記載の電力消費制御装置において、前記電力制御手段が、前記累積した電力情報と、消費された電力の情報との合計が前記所定の閾値よりも大きい場合、前記１つ以上のタスクのさらなる実行を阻止する、電力消費制御装置。
〔態様５９〕
コンピュータチップの温度を制御する方法であって、
前記チップの温度を監視する過程と、
前記チップの温度が所定の閾値よりも大きい場合、当該温度を減少させるために、前記チップにおけるイベントの処理を阻止する過程とを備えた、コンピュータチップ温度制御方法。
〔態様６０〕
態様５９に記載のコンピュータチップ温度制御方法において、さらに、

前記チップの接合温度を監視して、当該チップの温度を監視する過程を備えた、コンピュータチップ温度制御方法。

〔態様６１〕

コンピュータチップの温度を制御する装置であって、

前記チップの温度を監視する温度監視モジュールと、

前記チップの温度が所定の閾値よりも大きい場合、当該温度を減少させるために、前記チップにおけるイベントの処理を阻止する性能抑制モジュールとを備えた、コンピュータチップ温度制御装置。

〔態様６２〕

態様６１に記載のコンピュータチップ温度制御装置において、前記温度監視モジュールが、前記チップの接合温度を監視して、当該チップの温度を監視する、コンピュータチップ温度制御装置。

本発明は、また、実施の態様として以下の内容も含む。

〔態様６３〕

コンピュータチップの温度を制御する方法であって、

前記コンピュータチップの接合温度を監視することで、前記コンピュータチップの温度を監視する過程と、

前記監視される温度が所定の閾値よりも大きい場合、前記コンピュータチップの温度を減少させるために、前記コンピュータチップにおけるイベントの処理を阻止する過程とを備えた、コンピュータチップ温度制御方法。

〔態様６４〕

コンピュータチップの温度を制御する装置であって、

前記コンピュータチップの接合温度を監視することで、前記コンピュータチップの温度を監視する温度監視モジュールと、

前記監視される温度が所定の閾値よりも大きい場合、前記コンピュータチップの温度を減少させるために、前記コンピュータチップにおけるイベントの処理を阻止する性能抑制モジュールとを備えた、コンピュータチップ温度制御装置。

〔態様６５〕

態様６３に記載のコンピュータチップ温度制御方法において、さらに、

前記コンピュータチップのダイオードインタフェースおよびシリアル接続インタフェースによって、前記コンピュータチップを、外部温度センサチップに機能的に結合する過程と、

前記ダイオードインタフェースを介して、前記外部温度センサチップによって検知された温度を監視することにより、前記シリアルインタフェースを介して前記コンピュータチップの温度を監視する過程であって、前記ダイオードインタフェースは前記コンピュータチップのサーマルダイオードに機能的に結合されている、過程とを備えた、コンピュータチップ温度制御方法。

〔態様６６〕

態様６３に記載のコンピュータチップ温度制御方法において、さらに、

前記コンピュータチップ上にマルチプロセッサコアを含む過程と、

各マルチプロセッサコアを、そのコアの最大電力消費限界を示すフィールドに関連付ける過程と、

前記監視される温度が前記所定の閾値よりも大きい場合、前記マルチプロセッサコアの少なくとも１つに関連づけられた前記フィールドを動的に調整する過程とを備えた、コンピュータチップ温度制御方法。

【符号の説明】

【００６９】

１２０　電力累積手段

２５０　比較部

２６０　制御部

【図１】

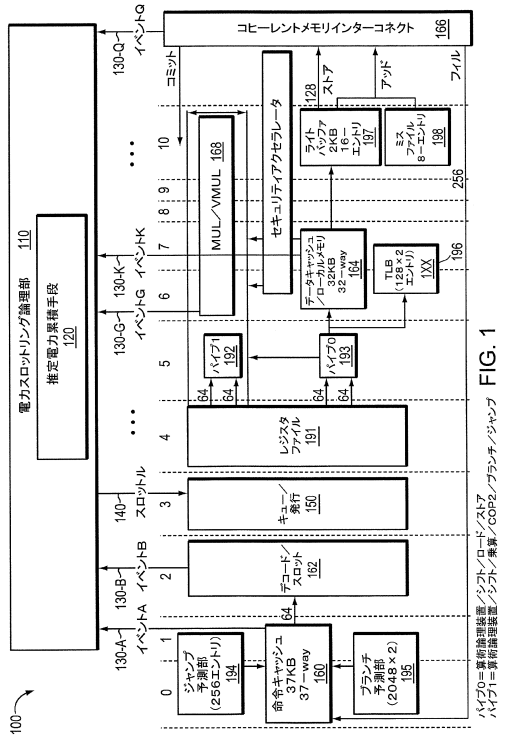

FIG. 1

【図２Ａ】

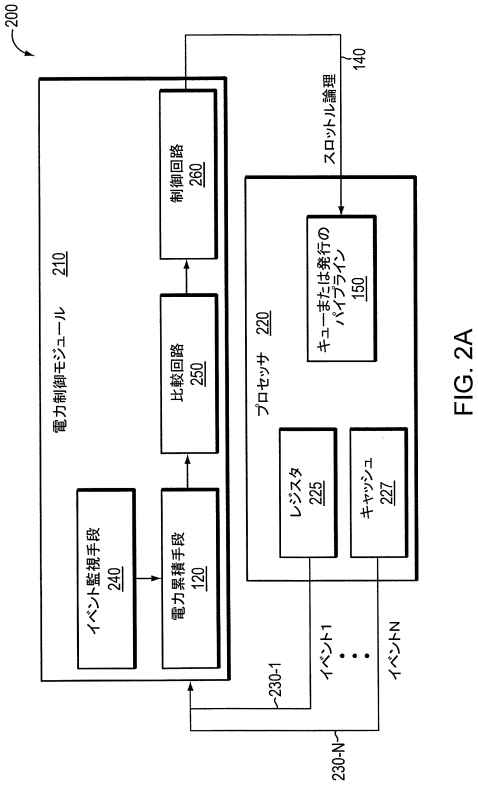

FIG. 2A

【図２Ｂ】

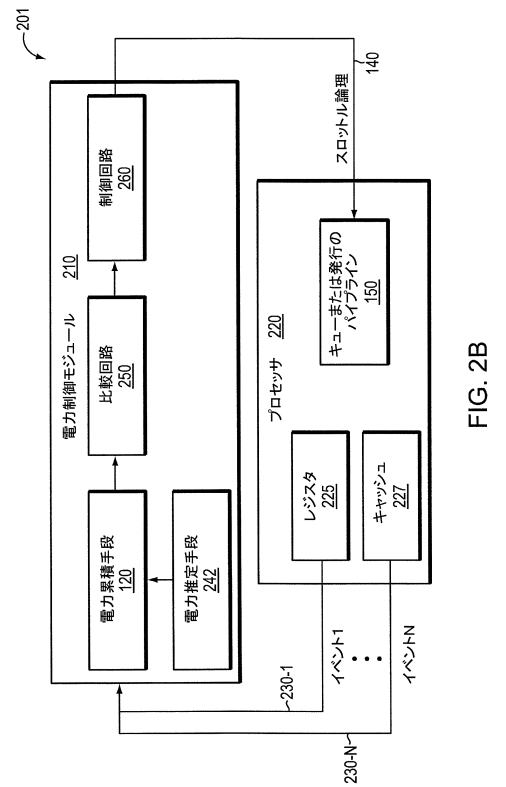

FIG. 2B

【図２Ｃ】

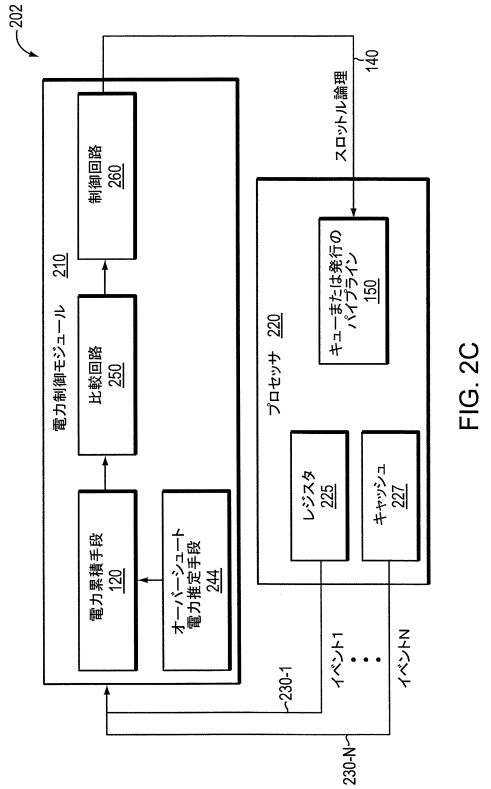

FIG. 2C

【図３Ａ】

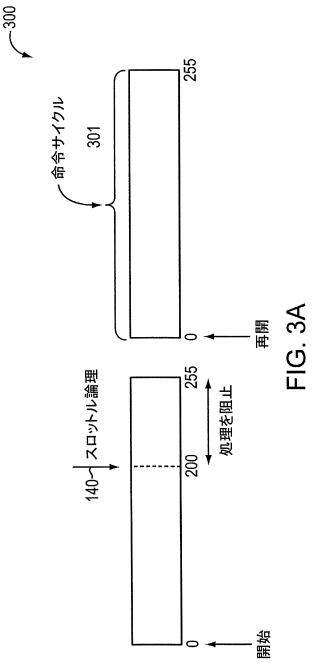

FIG. 3A

【図３Ｂ】

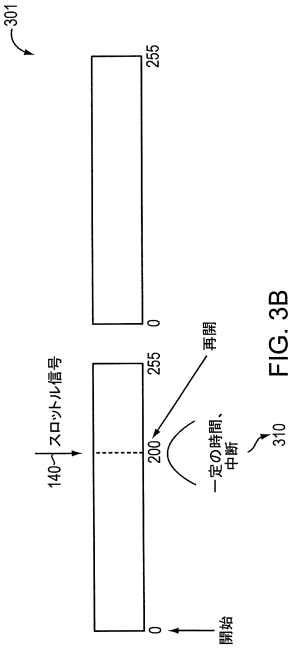

FIG. 3B

【図３Ｃ】

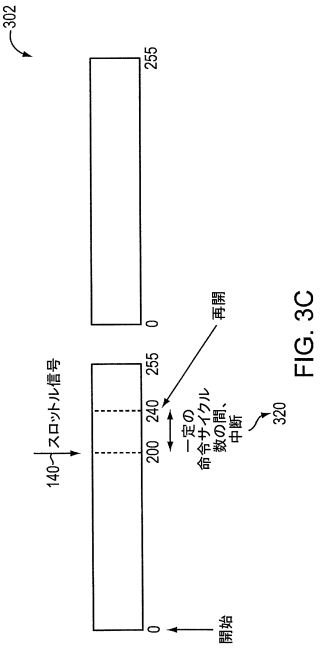

FIG. 3C

【図４】

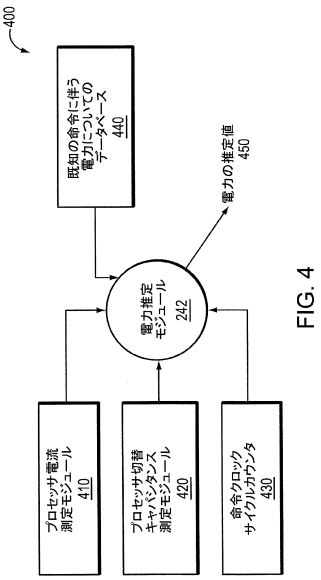

FIG. 4

【図５】

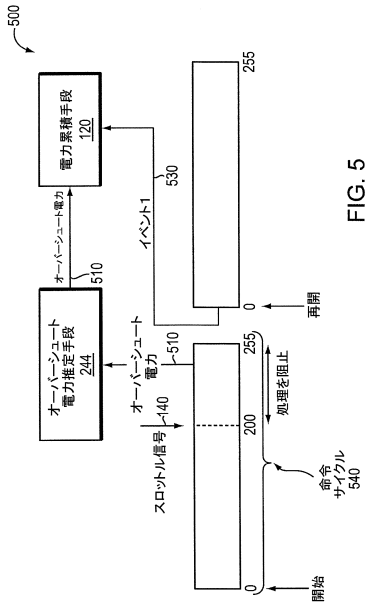

FIG. 5

【図６】

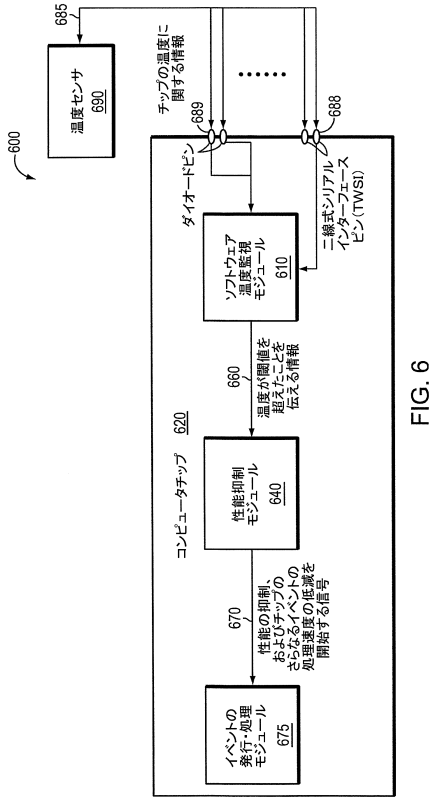

FIG. 6

フロントページの続き

(72)発明者 カールソン・ディビッド・エイ
アメリカ合衆国, テキサス州 ７６０５２, ハスレット, ネトル レーン １４０９
(72)発明者 ケスラー・リチャード・イー
アメリカ合衆国, マサチューセッツ州 ０１５４５, シュリューズバリー, ゼストランド ドライブ ３０
(72)発明者 ハイダー・アメル
アメリカ合衆国, カリフォルニア州 ９５０７０, サラトガ, パセオ セロ １２６２１

審査官 征矢 崇

(56)参考文献 国際公開第２００９／０８８４５１（WO, Ａ2）
国際公開第２００３／０３６４４９（WO, Ａ1）
国際公開第２００６／０３７１１９（WO, Ａ2）
特開２００５－２９５７６８（ＪＰ, Ａ）
国際公開第２００１／０４８５８４（WO, Ａ1）
特表２０１１－５０９４６１（ＪＰ, Ａ）
特表２００５－５２７０１０（ＪＰ, Ａ）
特表２００８－５１３９１２（ＪＰ, Ａ）
特表２００３－５２３５６３（ＪＰ, Ａ）

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ１／２８；１／３２
Ｇ０６Ｆ１／０４