

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2005-285123

(P2005-285123A)

(43) 公開日 平成17年10月13日(2005. 10. 13)

(51) Int. Cl.⁷

G06F 9/50
G06F 1/04

F I

G06F 9/46 465D
G06F 1/04 301C

テーマコード(参考)

5B079

審査請求 有 請求項の数 52 O L (全 27 頁)

(21) 出願番号 特願2005-88403 (P2005-88403)
(22) 出願日 平成17年3月25日(2005. 3. 25)
(31) 優先権主張番号 10/812, 177
(32) 優先日 平成16年3月29日(2004. 3. 29)
(33) 優先権主張国 米国 (US)

(71) 出願人 395015319
株式会社ソニー・コンピュータエンタテインメント
東京都港区南青山二丁目6番21号
(74) 代理人 100105924
弁理士 森下 賢樹
(72) 発明者 井上 敬介
東京都港区南青山2丁目6番21号 株式会社ソニー・コンピュータエンタテインメント内
Fターム(参考) 5B079 BA01 BC01 BC07

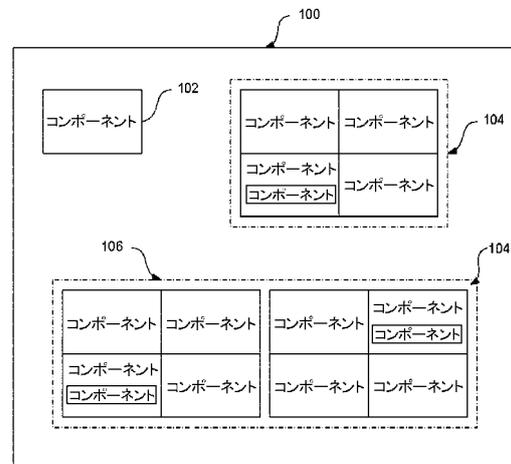
(54) 【発明の名称】 タスクスケジューリング作成の処理を使用するタスク温度管理を達成する方法および装置

(57) 【要約】

【課題】 コンピュータを使用する環境において温度管理を実行するための装置および方法を提供する。

【解決手段】 熱特性は、演算および/または処理コンポーネント102に関連し、演算は温度閾値を越えないようにコンポーネント102による処理がスケジューリングされる。ホットキューまたはクールキューは、選択された演算に提供され、処理コンポーネント102は、熱閾値を越えないように、適当なキューから演算を選択することができる。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

コンポーネントによって実行される複数の演算を提供し、
前記演算を、前記演算の実行の間前記コンポーネントによって発生すると予想される熱量に関連する値を表す熱特性と関連付け、

熱閾値を越えないように、前記熱特性に基づく実行順序に応じて前記演算をスケジューリングすることを特徴とする演算スケジューリング方法。

【請求項 2】

更に温度検知手段により前記熱特性を測定することを特徴とする請求項 1 に記載の方法。

10

【請求項 3】

更に前記コンポーネントの電力消費に基づく前記熱特性を推定することを特徴とする請求項 1 または 2 に記載の方法。

【請求項 4】

前記熱特性を推定することは、前記コンポーネントの回路シミュレーションを実行することを更に含むことを特徴とする請求項 3 に記載の方法。

【請求項 5】

前記熱特性を推定することは、前記コンポーネントの電力密度を決定することを更に含むことを特徴とする請求項 3 に記載の方法。

【請求項 6】

前記実行順序に応じて演算を実行する前記コンポーネントを更に備えることを特徴とする請求項 1 から 5 のいずれかに記載の方法。

20

【請求項 7】

前記熱特性は前記演算を実行する処理デバイスの選択された一つの集合の熱特性であって、前記コンポーネントは、複数の前記処理デバイスを含むことを特徴とする請求項 6 に記載の方法。

【請求項 8】

処理デバイスの各々は個別熱閾値を有し、前記熱特性は複数の個別熱特性を含み、個別熱特性の各々は前記処理デバイスの一つと関連するものであって、前記コンポーネントは、複数の前記処理デバイスを含むことを特徴とする請求項 1 から 7 のいずれかに記載の方法。

30

【請求項 9】

更に前記処理デバイスの少なくともいくつかを前記演算を実行するために選択し、
前記選択された処理デバイスをモニタし、
前記個別熱閾値を越えないように、前記選択された処理デバイスの中に前記演算を発送することを特徴とする請求項 8 に記載の方法。

【請求項 10】

前記熱特性は複数の処理デバイスの中に割り当てられ、前記コンポーネントは前記複数の処理デバイスを含むことを特徴とする請求項 1 から 9 のいずれかに記載の方法。

【請求項 11】

更に (i) 前記コンポーネントの電力消費を決定し、
(i i) 前記コンポーネントのスペースを決定し、
(i i i) 領域毎の電力消費を得るために、前記コンポーネントの前記スペースによって前記コンポーネントの前記電力消費を割り、
(i v) 熱評価定数を前記領域毎の電力消費に掛け、前記熱特性を決定することを特徴とする請求項 1 から 10 のいずれかに記載の方法。

40

【請求項 12】

更に冷却特性を決定し、
前記熱特性に加えて前記冷却特性を組み込む演算をスケジューリングすることを特徴とする請求項 1 から 11 のいずれかに記載の方法。

50

- 【請求項 13】
一連の演算を含むプログラムコードを獲得し、
前記演算の一つ以上と関連した熱特性を決定し、
コンポーネントの熱閾値を決定し、
前記熱閾値を越えないように、前記熱特性にしたがって、前記コンポーネントによる実行のための前記演算をスケジューリングすることを特徴とする温度スケジューリング方法。
- 【請求項 14】
前記熱特性は、前記コンポーネントが選択された演算を実行するにしたがって発生することが予想される熱量を見積もることを特徴とする請求項 13 に記載の方法。 10
- 【請求項 15】
前記熱特性は、ある期間にわたって発生する熱量を見積もることを特徴とする請求項 13 に記載の方法。
- 【請求項 16】
前記熱特性は、前記コンポーネントの電力消費および前記コンポーネントの電力密度の少なくとも一つを見積もることを特徴とする請求項 13 に記載の方法。
- 【請求項 17】
更に前記コンポーネントは前記演算を実行し、
実行の間、前記コンポーネントの温度をモニタし、
前記熱閾値を越える場合、前記演算を再スケジューリングすることを特徴とする請求項 13 から 16 のいずれかに記載の方法。 20
- 【請求項 18】
更に前記演算の各々において実行されるタスクの数を計数することによって、前記熱特性を推定することを特徴とする請求項 13 から 17 のいずれかに記載の方法。
- 【請求項 19】
前記コンポーネントは、複数の処理デバイスを含むものであって、
更に前記処理デバイスの選択されたものによって前記演算の選択されたものの動的な実行をモニタし、
前記選択された処理デバイスの演算周波数を決定し、
コンパイラに前記選択された処理デバイスの前記演算周波数を伝えることを特徴とする請求項 13 から 18 のいずれかに記載の方法。 30
- 【請求項 20】
更に前記コンポーネントを含む計算デバイスの冷却特性を決定し、
前記演算のスケジューリングは、前記冷却特性および前記熱特性にしたがって実行されることを特徴とする請求項 13 から 19 のいずれかに記載の方法。
- 【請求項 21】
前記冷却特性は、前記計算デバイスのパッケージのタイプに基づくことを特徴とする請求項 20 に記載の方法。
- 【請求項 22】
前記冷却特性は、さらに前記計算デバイスの冷却手段に基づくことを特徴とする請求項 21 に記載の方法。 40
- 【請求項 23】
前記冷却手段が単一の状態を有する場合、前記冷却特性は固定され、
前記冷却手段が複数の状態を有する場合、前記冷却特性は動的であることを特徴とする請求項 22 に記載の方法。
- 【請求項 24】
コンポーネントを含む計算デバイスと、
前記コンポーネントによって実行される複数の演算と、
前記演算の選択された一つの実行の後の前記コンポーネントの温度の変化を表す熱特性である、前記コンポーネントおよび前記演算の選択された一つに関連する前記熱特性の少 50

なくとも一つと、を備えることを特徴とする演算処理システム。

【請求項 25】

前記熱特性によって前記演算のうちの少なくとも一つを前記コンポーネントに割り当てるように操作可能なスケジューラを更に備えることを特徴とする請求項 24 に記載の演算処理システム。

【請求項 26】

前記スケジューラは、前記熱特性によって格納場所から前記演算のうち選ばれた一つを取り出すように操作可能であることを特徴とする請求項 25 に記載の演算処理システム。

【請求項 27】

前記熱特性は前記コンポーネントと関連し複数のサブコンポーネントと関連しない全体熱特性であって、前記コンポーネントは、前記複数のサブコンポーネントを含むことを特徴とする請求項 25 に記載の演算処理システム。 10

【請求項 28】

前記熱特性はさらに少なくともいくつかのサブコンポーネントに関連するものであって、前記コンポーネントは、複数の前記サブコンポーネントを含むことを特徴とする請求項 25 に記載の演算処理システム。

【請求項 29】

前記コンポーネントは処理デバイスであり、そして、スケジューラは前記処理デバイスと統合されることを特徴とする請求項 25 に記載の演算処理システム。

【請求項 30】

前記熱特性はタスク熱特性であって、前記選択された演算は、前記タスクを有することを特徴とする請求項 24 から 29 のいずれかに記載の演算処理システム。 20

【請求項 31】

前記タスク熱特性は、前記コンポーネントの演算周波数、前記コンポーネントの熱特性および冷却特性のうちの少なくとも一つに基づくことを特徴とする請求項 30 に記載の演算処理システム。

【請求項 32】

前記演算の少なくともいくつかは、優先度を含むものであって、
複数の優先度キューを更に備え、
各々の優先度キューは、前記演算の第 1 セットを格納する第 1 キューと、前記演算の第 2 セットを格納する第 2 キューである、第 1 キューおよび第 2 キューを含むことを特徴とする請求項 24 から 31 のいずれかに記載の演算処理システム。 30

【請求項 33】

前記演算の優先度と前記熱特性に基づいて選択された優先度キューの一つにおいて、前記第 1 または前記第 2 キューいずれかに少なくともいくつかの前記演算を割り当てるように操作可能なスケジューラを更に備えることを特徴とする請求項 32 に記載の演算処理システム。

【請求項 34】

前記スケジューラは、更に、前記熱特性および前記選択された演算の優先度により選択された優先度キューの第 1 キューまたは第 2 キューから前記演算の選択された一つを取り出すように操作可能であることを特徴とする請求項 33 に記載の演算処理システム。 40

【請求項 35】

演算上の閾値を越える第 1 熱特性を有する第 1 演算と、
前記演算上の閾値を越えない第 2 熱特性を有する第 2 演算と、
前記熱特性に基づいて前記第 1 および前記第 2 演算を備える複数の演算を管理するスケジューラと、
複数のプロセッサの各々が熱閾値を有し、前記複数の演算を実行する複数のプロセッサと、を含むことを特徴とする演算処理システム。

【請求項 36】

前記複数のプロセッサのうち選択された一つの前記熱閾値を越えない場合、前記選択さ 50

れたプロセッサは、前記第 1 演算および前記第 2 演算の少なくとも一つを獲得し実行するように操作可能であることを特徴とする請求項 3 5 に記載の演算処理システム。

【請求項 3 7】

前記選択されたプロセッサの前記熱閾値を越えない場合、前記選択されたプロセッサは前記第 1 演算を獲得することを特徴とする請求項 3 6 に記載の演算処理システム。

【請求項 3 8】

前記複数のプロセッサのうち選択された一つの前記熱閾値を越える場合、前記選択されたプロセッサは前記第 2 演算を獲得し実行するように操作可能であることを特徴とする請求項 3 5 から 3 7 のいずれかに記載の演算処理システム。

【請求項 3 9】

前記プロセッサは、前記プロセッサの温度をモニタするかまたは推定する温度検知手段を含むことを特徴とする請求項 3 5 から 3 8 のいずれかに記載の演算処理システム。

【請求項 4 0】

各々のプロセッサは、前記温度検知手段から温度値を受け取って、デジタル温度値を提供するように操作可能であるアナログデジタル変換器を更を含むことを特徴とする請求項 3 9 の演算処理システム。

【請求項 4 1】

前記プロセッサの選択された一つは、複数のサブプロセッサを含むことを特徴とする請求項 3 5 から 4 0 のいずれかに記載の演算処理システム。

【請求項 4 2】

前記プロセッサは、前記プロセッサと関連し前記サブプロセッサと関連しない全体熱特性を有することを特徴とする請求項 4 1 に記載の演算処理システム。

【請求項 4 3】

各々のサブプロセッサは、他のサブプロセッサのコンポーネント熱特性とは別のコンポーネント熱特性を有することを特徴とする請求項 4 1 に記載の演算処理システム。

【請求項 4 4】

第 1 演算の熱特性に基づいて第 1 演算を格納し、
第 2 演算の熱特性に基づいて第 2 演算を格納し、
プロセッサの熱閾値によって、前記第 1 および前記第 2 の演算の少なくとも一つを取り出すことを特徴とする演算実行方法。

【請求項 4 5】

前記プロセッサの熱閾値を越えない場合、前記第 1 演算および前記第 2 演算のうちの少なくとも一つが取り出されることを特徴とする請求項 4 4 に記載の方法。

【請求項 4 6】

前記第 1 演算だけが取り出されることを特徴とする請求項 4 5 に記載の方法。

【請求項 4 7】

前記プロセッサの熱閾値を越える場合、前記第 2 演算が取り出されることを特徴とする請求項 4 4 から 4 6 のいずれかに記載の方法。

【請求項 4 8】

更に前記第 1 演算の優先度を決定し、
前記第 2 演算の優先度を決定し、
各々の優先度キューは第 1 キューおよび第 2 キューを含む、複数の優先度キューを提供し、

前記第 1 演算は前記第 1 演算の優先度に基づいて前記第 1 キューの一つに格納され、前記第 1 演算は、前記第 2 演算の優先度に基づいて前記第 2 キューの一つに格納されることを特徴とする請求項 4 4 に記載の方法。

【請求項 4 9】

更にコンポーネント熱特性を前記プロセッサと関連付け、
少なくとも一つの演算を取り出すことは、前記コンポーネント熱特性の現在の状態を評価すること、および、前記コンポーネント熱特性に基づいて前記少なくとも一つの演算を

10

20

30

40

50

選択することを更に含むことを特徴とする請求項 4 4 に記載の方法

【請求項 5 0】

プロセッサの温度が熱閾値を越えるかを決定し、

(i) 前記熱閾値を越えない場合、実行中の前記プロセッサの温度を維持または増加させそうな第 1 演算であって、前記第 1 演算が利用可能かを決定し、前記第 1 演算が利用できる場合、前記第 1 演算を実行し、

(i i) 前記熱閾値を越える場合、実行中の前記プロセッサの温度を減少させそうな第 2 演算であって、前記第 2 演算が利用可能かを決定し、前記第 2 演算が利用できる場合、前記第 2 演算を実行することを特徴とする演算実行方法。

【請求項 5 1】

更に前記第 2 演算が利用できない場合、演算を実行しないことを特徴とする請求項 5 0 に記載の方法。

【請求項 5 2】

更に優先度レベルを決定し、

前記優先度レベルの前記優先度キューから前記第 1 演算が利用可能かどうかを決定し、

前記第 1 演算が前記優先度キューから利用できない場合、前記第 2 演算が前記優先度キューから利用可能かどうかについて決定することを特徴とする請求項 5 0 に記載の方法。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本発明は、処理環境において熱管理を実行し、および特に命令およびタスクを効果的に割り当てることにより、熱ホットスポットを減少させる方法と装置に関する。

【背景技術】

【0 0 0 2】

コンピュータシステムはよりますます複雑になっている。そして、同時に、コンポーネントのサイズを縮小して、高密度にコンピュータチップ上のデバイスを実装しており、より高処理速度を達成している。これらの長所は、多くのアプリケーション、例えばリアルタイムでマルチメディアのゲームおよび他の計算集約型のアプリケーションの成功にきわめて重大である。コンピュータシステムは、処理効率を上げるために、しばしば並行に（または協調して）作動する多数のプロセッサを組み込む。

【0 0 0 3】

コンポーネントおよび素子が演算、例えば命令およびタスクを実行する時に、熱がよく発生する。過剰な熱は、逆に電子構成部品、例えばコンピュータチップの処理能力に影響を与え得る。例えば、チップの一つの領域が計算において集中的なタスクを実行している場合、その領域はかなり加熱する可能性があり、また他のチップと関連してホットスポットを形成する可能性がある。ホットスポットがある熱閾値を越える場合、コンポーネントの処理能力またはチップのその領域の素子は性能が低下する可能性があり、または、チップが損害を受けるかまたは破壊される可能性すらある。

【0 0 0 4】

過去において、種々の解決手段が加熱問題を解決するために採用されている。機械的解決手段の一つは、ヒートシンクをコンピュータチップに取り付けることである。しかしながら、ヒートシンクは大きく、単にチップから、そして、チップを囲んでいるスペース領域に熱を放出するのに役立つにすぎない。チップがエンクロージャ、例えばパソコンキャビネットに格納されるときに、この熱はファンの使用などによって除去されなければならないが、それ自身スペースをとり、不要なノイズを発生させる。

【0 0 0 5】

ほかに、より複雑な温度管理手段も存在する。例えば、ある解決手段では、温度センサが重要な回路要素、例えばプロセッサに配置されることができ、そして、ファンが関連するシステムエンクロージャに載置されることができる。温度センサが特定の温度に達したことを示すときに、ファンがオンになる。そして、プロセッサを冷却するためのシステ

10

20

30

40

50

ムエンクロージャで、エアフローを増加させる。あるいは、温度センサが所定の温度を越えることを示すときに、処理環境にシャットダウンをするよう警報が発せられることもできる。このセンサは、しばしばホットスポットから少し離れて配置される。残念なことに、このフィードバック方法は、オーバーヒートを防止するにはあまりに遅く、または信頼性なく機能する可能性がある。

【0006】

温度管理を行うための更なる試みでは、ソフトウェアの使用を採用する。例えば、ある技術では、演算間で冷却するより多くの時間をもつように、コンポーネントのクロックを減速する。ある従来のシステムは、命令キャッシュから抑制機構を使用する命令バッファまで、命令取得率を制御する。取得率を減らすことは、熱の発生を低減する。更により抜本的な方法は、プロセッサをシャットダウンして、それを冷やすことである。残念なことに、これらの技術の全てコンポーネントの演算速度に直接影響し、リアルタイム処理要求に不利益となる可能性がある。

【発明の開示】

【発明が解決しようとする課題】

【0007】

したがって、追加的なハードウェアまたは非効率的なソフトウェアルーチンを避けると共に、温度管理を達成するための新規な方法と装置の必要性が本技術においてある。

【課題を解決するための手段】

【0008】

本発明の態様にしたがって、温度スケジューリング方法は提供される。好ましい実施の形態において、コンポーネントは熱閾値を有し、そして、本方法はコンポーネントによって実行される複数の演算を提供することを含む。演算と熱特性とを関連させる温度情報が提供される。熱特性は、演算の実行の間、コンポーネントによって発生することまたは受けることが予想される熱量に関連する値を表す。演算を実行する命令は、熱閾値を越えないように、熱特性に基づいてスケジューリングされる。

【0009】

一つの形態において、本方法は、温度検知手段で熱特性を測定することを更に備える。他の形態において、本方法は、コンポーネントの電力消費に基づいて熱特性を推定することを更に備える。熱特性は、異なる方法で推定されることができ、一つのケースにおいて、推定は、コンポーネントの回路シミュレーションを実行することを含む。他のケースにおいて、推定は、コンポーネントの電力密度を決定することを含む。

【0010】

更に別の一つの形態において、本方法は、コンポーネントに処理能力の順に演算を実行させることを更に備える。より好ましくは、熱特性は演算を実行する選択された処理デバイスの集積された熱特性であって、コンポーネントは、複数の処理デバイスを含む。

【0011】

更なる形態において、処理デバイスの各々は個別熱閾値を有し、そして、熱特性は、各々が処理デバイスの一つと関連する複数の個別熱特性を含むものであって、コンポーネントは、複数の処理デバイスを含む。この場合、本方法は、好ましくは、演算を実行するために、処理デバイスの少なくともいくつかを選択することを含む。選択されたデバイスはモニタされ、そして、個別熱閾値を越えないように、演算は、選択されたデバイスの中に発送（ルーティング）される。選択し得る形態において、コンポーネントは、熱特性が複数の処理デバイスに割り当てられる、複数の処理デバイスを含む。

【0012】

更にもう一つの実施の形態において、熱特性を決定するステップは、(i) コンポーネントの電力消費を決定し、(ii) コンポーネントのスペースを決定し、(iii) 領域毎の電力消費を得るために、スペースによって電力消費を割り、(iv) 一定の温度評価定数を掛けること、を含むことができる。他の形態において、本方法は、冷却特性が熱特性と一緒に演算のスケジューリング作成に組み込まれるように、冷却特性を決定することを

10

20

30

40

50

更に含むことができる。

【0013】

他の好ましい実施の形態において、本方法は、最初に一連の演算を有するプログラムコードを得て、一つ以上の演算と関連する熱特性を決定する。コンポーネントの熱閾値もまた、決定される。演算は、熱閾値を越えないように、熱特性にしたがって、コンポーネントによる実行のためスケジューリングされる。

【0014】

一つの形態において、コンポーネントが選択された演算を実行するにしたがって、熱特性は発生すると予想される熱量を見積もる。他の形態において、熱特性は、ある期間にわたって発生する熱量を見積もる。更なる形態において、熱特性は、コンポーネントの少なくとも一つの(a)電力消費および、(b)コンポーネントの電力密度を見積もる。

10

【0015】

更に別の形態において、本方法は、コンポーネントに演算を実行させることを更に備える。コンポーネントの温度は、実行の間モニタされる。熱閾値を越える場合、演算は再スケジューリングされる。

【0016】

他の形態において、本方法は、演算の各々において実行されるタスクの数を計数することにより、熱特性を推定することをさらに備える。

【0017】

更なる実施の形態において、コンポーネントは複数の処理デバイスを含む。この場合、本方法は、好ましくは、選択された処理デバイスによって選択された演算の動的な実行をモニタすることを含む。選択された装置の演算周波数は、決定される。演算周波数は、コンパイラに伝えられる。

20

【0018】

更に別の形態において、本方法は、計算デバイスの冷却特性を決定することを更に備える。計算デバイスは、コンポーネントを含む。演算をスケジューリングするステップは、冷却特性および熱特性にしたがって実行される。冷却特性は、計算デバイスのパッケージのタイプに基づいてもよい。それは、また、計算デバイスの冷却手段に基づいてもよい。この場合、冷却手段が一つの状態を有する場合、冷却特性は確定する。冷却手段が多数の状態を有する場合、冷却特性は動的である。

30

【0019】

他の本発明の態様にしたがって、処理システム及び方法は、コンピュータの使用環境において、演算を扱うために提供される。処理システムの一実施の形態において、演算は計算デバイスの一部であるコンポーネントによって実行される。少なくとも一つの熱特性は提供される。特性は、コンポーネントおよび選択された演算と関連する。それは、演算を実行した後のコンポーネントの温度の変化を表す。

【0020】

一つの形態において、演算処理システムはスケジューラを含む。スケジューラは、熱特性によって、コンポーネントに少なくとも一つの演算を割り当てることができる。一つのケースにおいて、スケジューラは、熱特性によって、選ばれた演算を記憶場所から取り出すように操作可能である。他のケースにおいては、コンポーネントはサブコンポーネントを含み、スケジューラは簡易なスケジューラであり、そして熱特性はコンポーネントに関連しサブコンポーネントに関連しない全体熱特性である。更に別のケースにおいては、コンポーネントはサブコンポーネントを含み、スケジューラは高度なスケジューラであり、そして、熱特性は更にサブコンポーネントの少なくともいくつかと関連する。

40

【0021】

他の形態において、コンポーネントは処理デバイスであり、そして、スケジューラは処理デバイスと統合される。更なる実施の形態において、熱特性はタスク熱特性であり、選択された演算はタスクを備える。このケースにおいて、タスク熱特性は、好ましくは、(a)コンポーネントの演算周波数、(b)コンポーネントの熱特性、(c)冷却特性

50

、の少なくとも一つに基づく。

【0022】

更に別の形態において、演算の少なくともいくつかは、優先度を含む。この場合、システムは複数の優先度キューから更に成り、そして、各々の優先度キューは、第1セットの演算を格納するための第1（例えば、ホット）キュー、および第2セットの演算を格納するための第2（例えば、クール）キューを含む。好ましくは、システムはスケジューラを更に含む。スケジューラは、演算の優先度および熱特性に基づいて選択された優先度キューにおいて、少なくともいくつかの演算を、ホットまたはクールキューのいずれかに割り当てるように操作可能である。より好ましくは、スケジューラは、更に、熱特性および選択された演算の優先度によって選択された優先度キューのホットキューまたはクールキューから選ばれた一つの演算を取り出すように操作可能である。

10

【0023】

本発明の態様による、演算処理システムの他の実施の形態において、スケジューラ、第1および第2演算、および複数のプロセッサが提供される。スケジューラは、第1および第2演算を備える演算を管理する。第1演算は、演算上の閾値を越える熱特性を有する。第2演算は、演算上の閾値を越えない熱特性を有する。プロセッサは、演算を実行することができる。各々のプロセッサは、熱閾値を有する。

【0024】

一つの実施の形態において、選択されたプロセッサの熱閾値を越えない場合、そのプロセッサは第1第2演算のうちの一つを獲得し実行することができる。他の形態において、選択されたプロセッサの熱閾値を越えない場合、プロセッサは第1演算を獲得する。更なる実施の形態において、選択されたプロセッサの熱閾値を越える場合、プロセッサは第2演算を獲得し実行するように操作可能である。

20

【0025】

更に別の形態において、プロセッサは、プロセッサの温度をモニタし、または推定する、温度検知手段を含む。好ましくは、プロセッサは、温度検知手段から温度値を受け取って、デジタル温度値を提供するように操作可能であるアナログデジタル変換器を更に含む。

【0026】

他の形態において、選択されたプロセッサは、複数のサブプロセッサを含む。望ましくは、選択されたプロセッサは、選択されたプロセッサと関連し、サブプロセッサと関連しない、全体熱特性を有する。あるいは、各々のサブプロセッサは、望ましくは、他のサブプロセッサのコンポーネント熱特性と異なるコンポーネント熱特性を有する。

30

【0027】

本発明の態様にしたがって、コンピュータを使用する環境において演算を実行する好ましい方法が提供される。本方法は、第1演算を格納すること、および第2演算を格納することを含む。演算は、演算の熱特性に基づいて格納される。本方法は、プロセッサの熱閾値によって、少なくとも一つの演算を取り出すことを更に含む。

【0028】

好ましくは、プロセッサの熱閾値を越えない場合、演算のうちの一つは取り出されることができる。より好ましくは、第1演算だけが取り出される。プロセッサの熱閾値を越える場合、好ましくは、第2演算が取り出される。

40

【0029】

本方法は、好ましくは、第1および第2演算の優先度を決定し、複数の優先度キューを提供する追加ステップを含む。この場合、優先度キューは、第1および第2キューをその中で有して提供される。第1演算は、第1演算の優先度に基づいて、第1キューの一つに格納され、そして、第2演算は、第2演算の優先度に基づいて、第2キューの一つに格納される。

【0030】

他の形態においては、本方法はコンポーネント熱特性をプロセッサと関連付けることを

50

更に備える。この場合、演算を取り出すことは、コンポーネント熱特性の現在の状態を評価すること、およびコンポーネント熱特性に基づいて演算を選択することを含む。

【0031】

本発明の態様によれば、演算を実行する好ましい他の方法は、プロセッサの温度が熱閾値を越えるかどうかについて決定することを含む。それを越えない場合、本方法は、第1演算が利用可能かどうか決定する。第1演算は、実行中のプロセッサの温度を維持し、または増加させやすい。第1演算が利用できる場合、それは実行される。熱閾値を越える場合、本方法は第2演算が利用可能かどうか決定する。第2演算は、実行中のプロセッサの温度を減少させやすい。第2演算が利用できる場合、それは実行される。

【0032】

一つの形態において、クールキュー演算が利用できない場合、本方法は、好ましくは、「非演算」、または「nop」を実行する。他の形態においては、本方法は、優先度レベルを決定し、第1演算が優先度レベルの優先度キューから利用可能かどうか決定し、第1演算が優先度キューから利用可能でない場合、第2演算が優先度キューから利用可能かどうか決定することを更に備える。

【0033】

本発明の更なる態様による別の実施の形態においては、処理装置が提供される。処理装置は、熱特性と関連する演算を処理することができる。処理装置は、メモリおよび複数の処理デバイスを備える。メモリは、第1および第2演算を格納する。第1演算は、演算上の閾値を越える熱特性を有する。第2演算は、演算上の閾値を越えない熱特性を有する。処理デバイスは、演算を実行することができる。処理デバイスの少なくともいくつかは、熱閾値を有し、およびメモリへアクセスする。少なくとも、選択された処理デバイスは、処理要素、処理ユニットまたはサブ処理ユニットを含む。選択された処理デバイスの熱閾値を越えない場合、選択された処理デバイスは、処理するメモリから第1演算を獲得することができる。選択された処理デバイスの熱閾値を越える場合、選択された処理デバイスは、処理するメモリから第2演算を獲得することができる。

【0034】

一つの形態において、少なくともいくつかの処理デバイスは、処理要素である。好ましくは、少なくともいくつかの処理要素は、少なくとも一つのサブ処理ユニットを更に含む。この場合、サブ処理ユニットは、浮動小数点ユニット、整数ユニット、および浮動小数点および整数ユニットと関連したレジスタを含むことができる。より好ましくは、サブ処理ユニットは、ローカル記憶部を更に含む。

【0035】

他の形態において、少なくともいくつかの処理要素は、処理ユニット、および処理ユニットと関連した複数のサブ処理ユニットを更に備える。この場合、サブ処理ユニットの各々は、好ましくはローカル記憶部を含む。

【0036】

更なる実施の形態において、第1処理デバイスは、第1処理デバイスの熱閾値によって、演算を第2処理デバイスと交換するように操作可能である。

【0037】

更なる実施の形態において、メモリはサブ処理ユニットのローカル記憶部を含み、選択された処理デバイスは、サブ処理ユニットを含む。この場合、ローカル記憶部は、第1演算を管理するための第1キューおよび第2演算を管理するための第2キューを含むことができる。第1および第2演算は、時分割方式の装置のメモリにおいて維持されることができる。

【0038】

本発明の態様による他の変形例において、一对のメモリがあってもよい。一つの形態において、第1メモリは、演算上の閾値を越える演算特性を有し第1セットを管理する第1キュー、および演算上の閾値を越えない演算特性を有し第2セットを管理する第2キューを有する第2メモリを含む。選択された処理デバイスの熱閾値を越えない場合、その処理

10

20

30

40

50

デバイスは、処理を行う少なくとも一つの第1セット演算を獲得することができる。熱閾値を越える場合、処理デバイスは処理のための演算の第2セットのうちの少なくとも一つを獲得することができる。他の形態において、第1および第2演算を格納するメモリであって、第1演算は演算上の閾値を越える熱特性を有し、および第2演算は演算上の閾値を越えない熱特性を有する。選択された処理デバイスの熱閾値を越えない場合、その処理デバイスは、処理を行う第1または第2メモリのいずれか一方から第1演算を獲得することができる。熱閾値を越える場合、処理デバイスは、処理を行う第1または第2メモリのいずれか一方から第2演算を獲得することができる。

【0039】

本発明の態様にしたがって、処理タスクの方法が提供される。本方法は、特性に基づいて、コンポーネントによる実行のための複数のタスクのうち選択された一つを含む。各々のタスクの特性は、関連するタスクの実行の後のコンポーネントの温度に関連する。それから、選択されたタスクが実行される。

10

【0040】

好ましくは、特性は、関連するタスクの実行の後の、コンポーネントの予想される温度の増加または減少に関連する。予想される増加または減少は、コンポーネントの電力密度に基づく。タスクは、メモリの少なくとも一つのキューに格納されることができる。あるいは、タスクはメモリの少なくとも2つのキューに格納されることができる。一つのキューは、条件を満たす属性のタスクを格納する。他のキューは、条件を満たさない属性のタスクを格納する。条件は、特性が閾値を越える、ということであってもよい。タスクはまた、コンポーネントの現在の温度に基づいて選択されることができる。タスクは、実行の前に、メモリの異なるアドレスに格納されることができる。タスクはまた、実行の前に、メモリの同じアドレスに異なる時間に格納されることができる。好ましくは、コンポーネントはプロセッサである。

20

【0041】

他の本発明の態様にしたがって、処理されるタスクを格納するメモリ、およびメモリに格納されるタスクを処理するコンポーネントを含む、タスクを処理するシステムが提供される。タスクは、特性と関連する。各々のタスクの特性は、関連するタスクを処理した後のコンポーネントの温度に関連する。タスクの一つは、特性に基づいて、コンポーネントによる処理のために選択される。

30

【0042】

特性は、好ましくは、選ばれたタスクの処理の後のコンポーネントの温度の予想される増加または減少に関連する。予想される増加または減少は、コンポーネントの電力密度に基づいてもよい。タスクは、メモリの少なくとも一つのキューに格納されることができる。タスクは、また、メモリの少なくとも2つのキューに格納されることができる。この場合、一つのキューは、条件を満たす属性のタスクを格納し、別のキューは、条件を満たさない属性のタスクを格納する。条件は、特性が閾値を越える、ということであってもよい。

【0043】

メモリは、好ましくは、2つの別々のメモリ群を備える。一つのメモリ群は、条件を満たす属性のタスクを格納し、他のメモリ群は、条件を満たさない属性のタスクを格納する。2つの別々のメモリ群は、異なる時間で同じメモリアドレスに格納されることができる。2つの別々のメモリ群が、異なるメモリアドレスであってもよい。この場合、2つの別々のメモリ群は、同じ半導体デバイスであることができ、または異なる半導体デバイスであることができる。

40

【0044】

好ましくは、コンポーネントは温度センサを含む。この場合、タスクは、温度センサの出力に基づいて選択されることができる。システムは、また、タスクを処理することができる第2コンポーネントを含むことができる。1以上のタスクは、タスクを処理した後の各々のコンポーネントの予想温度に基づいて、各々のコンポーネントのために選択される

50

ことができる。システムは、また、1以上のコンポーネントのタスクを選択するスケジューラも含むことができる。

【発明を実施するための最良の形態】

【0045】

図面において示される本発明の好ましい実施例を記載することにおいて、特定の用語が明確性のため使われる。しかしながら、本発明は選択される詳細事項に限定されることを意図されない。そして、各々の詳細事項は、類似の目的を達成するための類似の方法において実行する、すべての技術的な同義語を含むことは理解される。

【0046】

現在図3Aに参照がなされ、図3Aは、本発明の態様にしたがって採用され得る、基本的な処理モジュールまたは処理要素300のブロック図である。この図に示すように、処理要素(PE)300は、好ましくは、I/Oインタフェース302、処理ユニット(PU)304、ダイレクトメモリアクセスコントローラ(DMAC)306、複数のサブ処理ユニット(SPUs)308、すなわち、複数のサブ処理ユニット308a-308dを含む。4つのサブ処理ユニット308a-dが示されているが、処理要素300はいかなる数のこのようなデバイスを含むことができる。ローカル(または内部)処理要素バス320は、処理ユニット304、複数のサブ処理ユニット308、I/Oインタフェース302、ダイレクトメモリアクセスコントローラ306およびメモリアクセス310の中のデータおよびアプリケーションを送信する。ローカル処理要素バス320は、例えば、従来のアーキテクチャを有することができ、またはパケットスイッチネットワークとして提供される。パケットスイッチネットワークとしての機器は、より多くのハードウェアを必要とする一方、利用できるバンド幅を増やす。

【0047】

デジタル論理の機器を提供するさまざまな方法を使って処理要素300が作成されることができる。しかしながら、処理要素300は、好ましくはシリコン基板上のCMOSを使用している単一の集積回路として作成される。処理要素300は、高バンド幅メモリ接続322を通じてメモリ330と密接に関連する。メモリ330は、望ましくは処理要素300のためのメインメモリとして機能する。メモリ330は好ましくはダイナミックランダムアクセスメモリ(DRAM)であるが、メモリ330は、例えばスタティックRAM(SRAM)、他の手段、例えば、磁気ランダムアクセスメモリ(MRAM)、光メモリ、ホログラフィックメモリ、その他などを使って提供されてもよい。ダイレクトメモリアクセスコントローラ306およびメモリアクセス310は、メモリ330および複数のサブ処理ユニット308および処理要素300の処理ユニット304間のデータ転送を容易にする。

【0048】

処理ユニット304は、例えば、データおよびアプリケーションの単独処理ができる標準のプロセッサでもよい。演算中において、処理ユニット304は、複数のサブ処理ユニット308によってデータおよびアプリケーションの処理をスケジューリングし、編成する。複数のサブ処理ユニット308は、好ましくは単命令複データ(SIMD)プロセッサである。処理ユニット304の制御のもと、複数のサブ処理ユニット308は、並行かつ独立した方法においてデータおよびアプリケーションの処理を実行することができる。ダイレクトメモリアクセスコントローラ306は、共有メモリ330に格納されるデータおよびアプリケーションへのアクセスを、処理ユニット304および複数のサブ処理ユニット308によって制御する。好ましくは、処理要素300のような多数の処理要素は、高い処理能力を提供するために、ともに接続され、パッケージされてもよく、また、相互に論理的に関連付けられてもよい。

【0049】

図3Bは、本発明の態様による、演算されることができる多数の処理要素350(処理要素1、処理要素2、処理要素3および処理要素4)を備える処理アーキテクチャを示す。好ましくは、処理要素350は単一のチップ上にある。複数の処理要素350は、図3

10

20

30

40

50

Aの処理要素300に関して上記にて論じた処理ユニット、および/または複数のサブ処理ユニットのようにサブシステムを含んでいてもよく、または含んでいなくてもよい。必要な処理のタイプによって、複数の処理要素350は同じタイプでもよく、または異なるタイプでもよい。例えば、処理要素350は、一般的なマイクロプロセッサ、デジタル信号プロセッサ、グラフィックプロセッサ、その他であってもよい。

【0050】

複数の処理要素350は、好ましくは共有バス352に結合される。メモリコントローラまたはダイレクトメモリアクセスコントローラ356は、メモリバス354を通じて共有バス352に接続されてもよい。ダイレクトメモリアクセスコントローラ356は、メモリ358に接続し、メモリ358は、メモリ330に関して上記にて論じたタイプの一つでもよいI/Oコントローラ362もまた、I/Oバス360を通じて共有バス352に接続されてもよい。I/Oコントローラ362は、一つ以上の入出力装置36、例えばフレームバッファ、ディスク駆動装置、その他に接続してもよい。上記の処理モジュールおよびアーキテクチャは単に典型的なだけであると理解されるべきであり、そして、さまざまな本発明の態様は、2001年3月22日に出願され、「ブロードバンドネットワークのコンピュータアーキテクチャのためのメモリ保護システムおよび方法」という名称の米国特許番号6,526,491、および2003年2月25日に出願され、「ブロードバンドネットワークのためのコンピュータアーキテクチャおよびソフトウェアセル」という名称の米出願番号第09/816004号において開示されるタイプのマルチプロセッサシステムを含むが、これに限られない。これにより、この中での参照により明確に具体化される。

10

20

【0051】

図4は、本発明の態様による、採用され得るサブ処理ユニット400の構造および機能を示す。サブ処理ユニット400は、好ましくはローカル記憶部402、レジスタ404、一つ以上の浮動小数点ユニット406および一つ以上の整数ユニット408を含む。サブ処理ユニット400のコンポーネントは、同様に後述するようなサブコンポーネントを備える。必要とされる処理能力によって、より多くのまたはより少ない数の浮動小数点ユニット(FPUs)406および整数ユニット(IUs)408が採用されてもよい。好ましい実施の形態においては、ローカル記憶部402は少なくとも128キロバイトの記憶部を含み、そして、レジスタ404の性能は128×128ビットである。浮動小数点ユニット406は、好ましくは1秒につき少なくとも320億の浮動小数点操作(32ギガフロップス)の速度で操作する、そして、整数ユニット408は、好ましくは1秒につき少なくとも320億の操作(32ギガオプス)の速度で操作する。

30

【0052】

ローカル記憶部402は、好ましくはキャッシュメモリでない。サブ処理ユニット400のためのキャッシュ整合性サポートは、不必要である。その代わりに、ローカル記憶部402が好ましくはSRAMとして作成される。処理ユニット204は、処理ユニット204によって開始されるダイレクトメモリアクセスのキャッシュ整合性サポートを要求することができる。しかしながら、キャッシュ整合性サポートは、サブ処理ユニット400によって開始されるダイレクトメモリアクセスのために、または外部素子へまたは外部素子からの接続のために要求されない。

40

【0053】

サブ処理ユニット400はバスインターフェース(Bus I/F)412を通じてサブ処理ユニット400へおよびサブ処理ユニット400からアプリケーションおよびデータを送信するバス410を更に含む。好ましい実施の形態においては、バス410は1,024ビット長である。サブ処理ユニット400は、内部バス414、416および418を更に含む。好ましい実施の形態においては、バス414は256ビットの幅を有し、ローカル記憶部402およびレジスタ404の間の通信を提供する。バス416および418は、レジスタ404、浮動小数点ユニット406、レジスタ404、および整数ユニット408の間に、それぞれ通信を提供する。好ましい実施の形態において、レジスタ4

50

04から浮動小数点または整数ユニットへのバス416および418の幅は384ビットであり、そして、浮動小数点または整数ユニットからレジスタ404へのバス416および418の幅は128ビットである。レジスタ404から浮動小数点ユニット406および整数ユニット408へのバスの大きな幅は、処理の間、レジスタ404から大きなデータフローに適応する。一例では、最高3つのワードが、各々の計算に必要とされる。しかしながら、各々の計算の結果は、通常一つのワードだけである。

【0054】

現在図1に参照がなされ、この図1は基板100中または基板100上に形成されるコンポーネント102を示す。基板100およびコンポーネント102は、コンピュータチップの一部または全体を含むことができる。コンポーネント102は、論理デバイスまたは回路の他のコンポーネントであってもよい。基板100の領域の一つ以上のコンポーネント102は、ユニット104として共に関連付けられることができる。ユニット104およびユニット104のグループ106もまた、例えば処理要素300、処理ユニット304、複数のサブ処理ユニット308、処理要素350または複コンポーネントそれ自体を形成するために相互に関連づけられることができる。例えば、ユニットのグループ106は、グループ106内のサブ処理ユニット400およびユニット104は、ローカル記憶部402、レジスタ404、浮動小数点ユニット406、整数ユニット408およびバスインターフェース412を含むことができる。各々のユニット104は、同様に、例えばDRAMメモセル、論理ゲート、バッファ等ような、他のユニット104およびコンポーネント102も含むことができる。コンポーネント102、ユニット104、およびグループ106は様々なレベルの複雑さを例証するのに使用されているが、より一般に、「コンポーネント」という用語は、処理要素300または処理要素350までの最も基本的な要素（例えばトランジスタおよびコンデンサ）および全体のコンピュータチップ自体から、すべてのレベルでデバイスについて言及するのにもまた使用される。典型的には、コンポーネントは基板100上の相補型金属酸化物半導体（CMOS）を採用している集積回路として作成される。基板100は、好ましくはシリコン基板である。基板100のために選択可能な材料はヒ化ガリウム、ヒ化ガリウムアルミニウム物、およびその他、多種多様なドーパントを採用したいわゆるIII-B合成物を含むが、これに限定されるものではない。コンポーネント102は、また、例えば高速単一磁束量子（RSFQ）論理などの超電導材料を使って提供されることができる。

【0055】

コンポーネントが処理命令またはタスク（例えば一連の命令）のような演算を実行するにしたがって、それらはしばしば熱を発生させる。ここで使用しているように、「演算」または「タスク」という表現は、実行される活動に関連し、命令、タスクおよび一回または多数のステップのプログラムを含むが、これに限定されるものではない。

【0056】

本発明の一態様において、コンポーネントによって実行される演算は、熱特性の値がその演算を実行するときに構造部分によって発生することが予想される熱量に関連するような熱特性と関連付けられることができる。好ましくは、熱特性はまた、適切なときに基礎が形成される。例えば、特性の値は、一定の期間にわたって発生された熱量を表してもよい。

【0057】

熱特性は、測定されることができるか推定されることができる。例えば、温度計または他の温度検出装置が、特定の演算をしているときのユニットの温度を計測するために実際に使用されることができる。

【0058】

熱特性は、好ましくはコンポーネントの電力消費に基づいて推定される。例えば、いくつかのコンポーネントは、演算するための電力をより多く必要としてもよく、そして高い熱特性を持ってよい。他のコンポーネントは同じ電力消費を有することができるが、共により高密度にパッケージされ、それらはかなり離れて間隔を置かれるコンポーネントよ

10

20

30

40

50

り多くの熱を発生させる傾向がある。この点に関しては、熱特性は、両方の要因に基づいて推定されることができ、その場合、熱属性は、コンポーネントまたはコンポーネントのグループの電力密度に基づく。このように、いくつかのケースでは、熱特性は、コンポーネントが演算を実行するにしたがって発生すると予想される熱、一定の期間にわたって発生された大量の熱、全体の消費電力、コンポーネントの電力密度、コンポーネントの関連グループの電力密度、を反映することができる。チップの効果的な温度管理を達成するために、各々のコンポーネントに電力消費をスケジューリングすることが望まれてもよい。コンポーネントの電力消費は、チップの開発中に推定されることができ、例えば、チップ、サブシステムおよび/または個別コンポーネントの回路シミュレーションが実行されてもよい。

10

【0059】

好ましくは、熱特性は更に特定のコンポーネントと関連している。例えば、整数加算演算のような演算が整数ユニット408だけを含んでいる場合、熱特性は整数ユニット408と特に関連付けられることができる。同様に、浮動小数点演算の熱特性は、浮動小数点ユニット406と特に関連付けられていてもよい。他の演算は、ローカル記憶部402からレジスタ404へデータを移動するような、1セットのコンポーネントを含むことができる。さらに他の演算は、すべてのコンポーネントを含んでも良く、いずれか特定のコンポーネントのセットに原因があるとすることが困難でもよい。例えば、三次元図を描写することは、サブ処理ユニット400の全てのコンポーネントに従事させることができ、その場合、熱特性はサブ処理ユニット400の全てのコンポーネントに適用される。あるいは、演算を実行するとき、どれくらいの熱が個別コンポーネントによって発生するかを予測するのが困難でもよく、その場合、演算の熱特性は、コンポーネントのグループに全体的に割り当てられても良い。以下のテーブルは、演算、コンポーネントおよび熱特性のサンプルセットを示す。

20

【0060】

【表1】

演算	熱特性	コンポーネント
三次元図描写	12	サブ処理ユニット400
整数加算	3	整数ユニット408
浮動小数点加算	7	浮動小数点ユニット406
メモリ移動	2	記憶部402 レジスタ404

30

【0061】

好ましい実施の形態において、与えられたコンポーネント（またはコンポーネントのセット）の熱特性は、次のように算出されることができる。

【0062】

$$T A = k * (P / S)$$

T A（熱特性）は、コンポーネントの電力密度または電力消費（P）が、サイズまたはスペース（S）で割られ、温度の推定に使われる因数または定数（k）がかけられたものと等しい。

40

【0063】

本発明の一態様による、プログラムコンパイラは、コンポーネントがオーバーヒートすることを防止するのに役立てるために熱特性を使用する。コンパイラは、ソフトウェア、ファームウェア、ハードウェアまたは上記の組合せにおいて提供されることができる。それは、処理要素（例えば処理要素300または処理要素350）かそのサブコンポーネントと関連して（例えば、組み込まれて）もよい。図11は、本発明の態様によるコン

50

パイラ機能を示す。本技術で公知であるように、コンパイラはソースコードを受信し、コンピュータシステム上で動作することができるオブジェクトコードを生成する。本発明の態様によれば、コンパイラは、演算および/またはコンポーネントに関する熱特性と同様にソースコードを受信する。コンパイラは、好ましくは熱特性に基づいてオブジェクトコードを生成する。コンパイラが、命令の数を計数することによってコンパイルを管理するときに、コンパイラによって編集されたオブジェクトコードの熱特性は不変的に推定される。強化された熱特性判定は、好ましくは、命令の動的な実行を計数することができ、各々のコンポーネントの演算周波数を伝えることができるパフォーマンスモニタである、「プロファイラ」を使用してなされる。プロファイラは、コンパイラにより正確な温度評価を提供することができ、それは同様に、熱的に最適化されたオブジェクトコード生成につ

10

【0064】

図2A-Bは、コンパイラまたは他の命令スケジューラが、どのように処理の低下またはコンポーネントへの損害を防止するように演算を管理することができるかについて示す。図の目的のため、熱閾値(T_{max})は、越えることが望ましくない温度を表すものとされる。三角形部分A、BおよびCは、コンポーネントによって実行される命令を表す。例えば、セグメントAおよびBは、例えば、計算上、非常に大量の熱を発生する集中的な命令またはタスクを表し、一方、セグメントCは、例えば、計算上、集中的でなく、またAまたはBのいずれかと同じ熱を発生しない。より詳細には、タスクA、BおよびCが全体的な算出の一部であると仮定すると、 $(2 * 3) + (4 * 5) + (6 + 7)$ において、タスクAは $(2 * 3)$ を表し、タスクBは $(4 * 5)$ を表し、そしてタスクCは $(6 + 7)$ を表す。図2Aに示すように、タスクがA、BおよびCの命令において実行されるときに、温度は T_{max} を越える可能性がある。この際、AおよびBが連続的に実行されるので、熱閾値 T_{max} は突破される。

20

【0065】

コンパイラが確実な命令を命じる方法に関して、しばしばコンパイラが指示を有していることは、公知技術である。本発明の好ましい実施の形態にしたがって、コンパイラは演算の熱特性に基づいて、選択的にスケジュールを再び命令することができる。好ましくは、コンパイラはまず最初に、演算A、BまたはCのいずれかに、それと関連する熱特性があるかどうかを決定する。その場合は、コンパイラは選択的に、その演算を、 T_{max} を越えることを防止する命令を使用するオブジェクトコードに編集することができる。上記の実施の形態において、コンパイラは、方程式が最後の結果を変えずに、計算される命令を変えることができる。例えば、それはA、CおよびBの順序に、演算をスケジュールリングすることができる。したがって、図2Bによって見られるように、命令の順序が変わるときに、温度は T_{max} を越えない。

30

【0066】

熱閾値 T_{max} が必ずしもフェール温度であるというわけではない点に注意する。その代わりに、 T_{max} は、例えば、評価される実施パラメータに基づいて選択された設計基準であってもよい。

【0067】

さらに、演算を再び命令するときに、コンパイラは、好ましくは計算をするコンポーネントの経過を追う。例えば、サブ処理ユニット他の一部(例えば浮動小数点ユニット406)が冷めたままである一方、一連の演算はサブ処理ユニット400の一部(例えば浮動小数点ユニット406)をオーバーヒートさせることができる。それらがサブ処理ユニットのさまざまなコンポーネントに均一に分配されるように、コンパイラは、好ましくは演算をスケジュールリングすることを試みることによってこの課題に対処する。コンパイラがこれを行い得る方法は、これらが熱特性を使ってプログラムの演算を実行するときに、コンポーネントの温度の経過を追い、シミュレーションすることである。例えば、コンポーネントXは、クロックサイクル毎に2熱特性の割合においては、冷えたとみなされることができ、それがオーバーヒートする前に8熱特性の閾値を有する。そのコンポーネントと

40

50

関連する演算が、サイクル毎に5ポイントの熱特性を有する場合、その演算が続けて3回実行された場合は、コンポーネントはオーバーヒートしたとみなされる(第1サイクルの後の5-2ポイントは、3の現在の熱インデックスの結果となる)。第2サイクルの後の5-2ポイントは、6の現在の総熱指数のためさらに3ポイントを加える。第2サイクルの後の5-2ポイントは、9の現在の総熱指数のためさらに3ポイントを加える。コンポーネントXがこのようなスケジューリングでオーバーヒートするかもしれないと検知された場合、コンパイラは、コンポーネントXが非稼働で冷たいままの間、別のコンポーネントにより実行される演算をスケジューリングするよう試みる。

【0068】

あるいは、コンパイラは、異なる演算であって、熱特性がコンポーネントが冷えると予想される速度よりも低い演算を選択することを試みることができる。例えば、コンパイラが、コンポーネントXが演算の現在のスケジュールでオーバーヒートする可能性があるとして決定する場合、それは、5の熱特性を持つ演算の間において1(それで、1サイクルあたり2熱属性の割合で冷えるなら、コンポーネントは冷えるであろう)の熱特性を持つ演算を点在させるよう試みることができる。

【0069】

その程度までコンポーネントは他のコンポーネントによって包含され、コンパイラは更により大きなコンポーネントの熱特性を、そのサブコンポーネントに、または、サブコンポーネントから親コンポーネントに割り当ててもよい。例えば、図10Aに示すように、個別コンポーネントが同時に2、3、2および7の熱特性を有する演算を実行している場合、それらの演算の全てのためのサブ処理ユニットの熱特性は14であるとみなされることができ。他方、全てのサブ処理ユニット400に起因している熱特性は、個別コンポーネントに割り当てられることができる。図10Bに示すように、三次元図の熱特性が12で全てのサブ処理ユニットに起因している場合、その値はサブ処理ユニット400の範囲内で均一にコンポーネントに割り当てられることができる。容器関連性、論理的機能および物理的近接によって関連するコンポーネント間の配分を含む、配分における他のバリエーションは可能である。

【0070】

さまざまなコンポーネントの熱値が、個別コンポーネントの即時の演算を反映するだけでなく、時間とともに累積的なものでもよく、コンポーネントのセットのために集計されることができることが分かる。これらの要因を考慮し、コンパイラは効果的に、熱閾値 T_{max} を避けるよう演算をスケジューリングすることができる。

【0071】

好ましくは、冷却特性は、さまざまなコンポーネントを含むコンピュータチップと関連する。冷却特性は、コンピュータチップの冷却装置の特定の特徴に依存する。例えば、冷却特性は、好ましくは、もしあればチップパッケージおよび冷却器(例えばヒートシンクまたはファン)に依存する。冷却システム冷却器(例えば、常に、設定された回転速度でファンを作動して)のための一つの状態を有している場合、冷却特性が確定される。例えばファンの回転速度を変えることによって、冷却システムの状態が変えられることができる場合、冷却特性は好ましくは動的であり、冷却システムが冷却器の演算状態を変えるときに、決定され、または更新されてもよい。実施の形態において、コンパイラは、冷却器の典型的な演算状態に基づいて算出される確定された冷却特性を使用する。特定のコンポーネントに属する演算の密度を算出するときに、コンパイラは冷却特性を使用する。より好ましくは、コンパイラもまたチップパッケージの熱放出能力の要素として入れる。他の実施の形態において、コンパイラまたはプリフィーラは、コンパイラがオブジェクトコード生成を実行することに役立てるため、動的な冷却特性を採用する。下のテーブルは、熱および冷却特性に基づいて、与えられた整数ユニット(IU)408により、および与えられたローカル記憶部(LS)402により、処理される整数演算の典型的なスケジュールを示す。

【0072】

10

20

30

40

50

【表 2】

命令#	命令処理主体	整数ユニット	ローカル記憶部
1	ローカル記憶部	0	2
2	整数ユニット	3	1
3	ローカル記憶部	2	3
4	整数ユニット	5	2
5	非演算	4	1
6	整数ユニット	7	0
7	整数ユニット	10	0
8	その他	9	0
9	非演算	8	0
10	非演算	7	0
...

10

20

【0073】

上記命令セットのため、整数ユニット408の熱特性が3であると仮定すると、チップの冷却特性は1であり、そして、整数ユニット408の熱閾値は10である。最も左の欄は命令番号を表し、第2欄はどのコンポーネントがその命令を扱うかを表し、そして、テーブルの右側上の2つの欄は、命令を扱った後に発生する熱またはコンポーネントの温度を示す。例えば、命令1はローカル記憶部によって処理されるかまたは実施され、2の熱値に結果となり、その一方で、整数ユニットはゼロのままとなる。命令2は整数ユニットによって作動され、そしてその後、整数ユニットは3の熱値を有し、そして、ローカル記憶部は1の熱値まで冷めている。この処理は命令5まで続き、そして、それは「非演算」(nop)である。これによって、整数ユニットおよびローカル記憶部がある程度まで冷めることができる。整数ユニットは命令6および7を処理する。そして、閾値までその熱値を増加させる。閾値を越えることを妨げるために、異なるコンポーネント(「その他」)は、好ましくは次の命令を処理する。例えば、プロフィーラは、整数ユニットおよびローカル記憶部による命令の実行をモニタすることができ、情報をコンパイラに伝えることができる。コンパイラは、他の整数ユニットによって処理される命令8を持つために、熱特性および冷却特性とともにこの情報を使用することができる。

30

【0074】

現在図5に参照がなされ、図5は、本発明の態様による、手順500をスケジューリングしているマルチキューを示す。図5に示すように、スケジューラ502は、好ましくは2つのキューと関連する。便宜のために、第1キューは本願明細書において「ホットキュー」504と称され、そして、第2キューは本願明細書において「クールキュー」506と称される。キュー504、506は、例えば、データ構造、またはメモリにおける連続的または不連続の集合など、多くの異なる方法で実行されることができる。複数のサブ処理ユニット400を採用する一つの実施の形態においては、キュー504、506は複数のサブ処理ユニット400の外部で実行される。キュー504、506はまた、例えばメモリ330(またはメモリ358)と関連する、処理ユニット304または処理要素300(または処理要素350)の外部で実行されても良い。他の例では、キュー504、506は、複数のサブ処理ユニット400の内部で実行される。望ましくは、キュー504、506はローカル記憶部402またはレジスタ404に関連して実行される。例えば、

40

50

ホットキュー 504 は第 1 サブ処理ユニット 400 のローカル記憶部 402 と連絡して実行され、そして、クールキュー 506 は第 2 サブ処理ユニット 400 のローカル記憶部 402 と連絡して実行されることができる。サブ処理ユニット 400 が多数のローカル記憶部 402 を含む場合には、ホットキュー 504 は、ローカル記憶部 402 の一つである第 1 ローカル記憶部 402 に格納されることができ、一方、クールキュー 506 は、同じサブ処理ユニット 400 において、ローカル記憶部 402 の一つである第 2 ローカル記憶部 402 に格納されることができる。あるいは、ホットキュー 504 およびクールキュー 506 は、同じローカル記憶部 402 で、またはサブ処理ユニット 400 の外部または処理要素 300 の外部の同じメモリで、実行されることができる。キュー 504、506 がレジスタ 404 を経由して実行される場合、さまざまな変形例が可能である。一つのケースでは、ホットキュー 504 は、第 1 サブ処理ユニット 400 のレジスタ 404 を経由して実行されることができ、そして、クールキュー 506 は第 2 サブ処理ユニット 400 のレジスタ 404 を経由して実行されることができる。キュー 504、506 はまた、例えば、キュー 504、506 のうちの 하나가、第 1 期間にメモリに格納され、キュー 504、506 のうちの他の 하나가、第 2 期間にメモリに格納されていた時分割方式の装置において行うことができる。

10

【0075】

スケジューラ 502 は、熱特性に応じて、命令、タスクまたは他の演算により、ホットキュー 504 およびクールキュー 506 を占めることができる。好ましくは、スケジューラ 502 は、熱特性を含むルックアップテーブルにアクセスする。スケジューラ 502 は、実行時間演算の前および/または間、演算することができる。スケジューラ 502 は、コンポーネントの現在の（または予測された）温度によって、タスクをホットキュー 504 またはクールキュー 506 から選択することができる。好ましい実施の形態においては、装置の現在の温度が演算上の閾値を越えない限り、スケジューラ 502 は、ホットまたはクールキュー 504、506 からタスクを選択することができる。好ましい他の実施の形態においては、演算上の閾値を越えない場合、およびホットおよびクールタスクの両方が利用可能である場合、スケジューラ 502 は、クールキュー 506 からタスクを選択する前に、ホットキュー 504 からタスクを選択する。例えば、多数の演算を必要とする浮動小数点命令またはタスクは、比較的高いまたは正の熱特性値と関連していてもよい。例えば、タスク H1 ... HN によって見られるように、これらの演算はホットキュー 504 に配置される。例えば、整数命令および一つの演算タスクなどの他の演算は、比較的低いまたは負の熱特性と関係していてもよい。タスク C1 ... CN によって見られるように、例えば、この種の演算はクールキュー 506 において配置される。タスクの熱特性は、コンパイラおよび/またはプロファイラから情報を使用して、好ましくは決定される。そして、そのいずれかは、タスクを実行している各々のコンポーネントの演算周波数を伝えることができる。より好ましくは、タスクの熱特性は、コンポーネントの演算周波数（例えば使用の頻度）、コンポーネントの熱特性、および冷却特性を組み込む。一つの実施の形態にしたがって、簡易なスケジューラは、例えば、サブ処理ユニット 400 などのサブコンポーネントを有するコンポーネントの全体熱特性を使用するだけである。他の実施の形態にしたがって、高度なスケジューラは、例えば、ローカル記憶部 402、浮動小数点ユニット 406 および整数ユニット 408 などのサブ処理ユニットのサブコンポーネントの熱特性を管理する。下のテーブルは三次元タスクおよび M P E G - 2 タスクのための与えられたサブ処理ユニットにおける、整数ユニット、浮動小数点ユニット、およびローカル記憶部のための熱特性を示す。

20

30

40

【0076】

【表 3】

タスク	整数ユニット	浮動小数点ユニット	ローカル記憶部	合計(サブ処理ユニット)
三次元	3	7	2	12
MPEG-2	2	0	0	2

【0077】

サブ処理ユニットの全体熱特性を見るだけの簡易なスケジューラは、それが12の値を有することを認識し、そして、それはサブ処理ユニットの熱閾値を越えることができる。このように、簡易なスケジューラは、サブ処理ユニットによって実行するMPEG-2タスクを選択することだけができてもよい。対照的に、高度なスケジューラは、好ましくはサブ処理ユニットのサブコンポーネントをモニタする。この場合、高度なスケジューラは、サブコンポーネントがその熱閾値を越えないことを認識することができ、そのため三次元タスクが選択されることができる。変形例において、スケジューラは、MPEG-2タスクが特定の段階で実行され、浮動小数点ユニットを冷やす時間を与えるように、タスクの範囲内でタスクまたは演算を再命令することができる。この適応性は、サブコンポーネント、コンポーネントおよび/または全ての多重処理システムがオーバーヒートせずに演算することを可能にする強力なツールである。

【0078】

当業者にとって明らかであるように、スケジューラ502はハードウェア、ファームウェアまたはソフトウェアで実行されることができる。好ましくは、スケジューラ502はハードウェアに基礎が置かれ、処理ユニット204において実行される。好ましい他の変形例においては、スケジューラ502は、全体的な計算デバイスの演算システムの一部として、ソフトウェアに基礎がおかれる。ホットキュー504およびクールキュー506は、好ましくは、バス508によるプログラム実行の間、一つ以上の処理要素(処理要素1...処理要素N)、処理ユニット(処理ユニット1...処理ユニットN)および/または複数のサブ処理ユニット(サブ処理ユニット1...サブ処理ユニットN)にアクセスすることができる。一つの実施の形態にしたがって、各々の処理要素、処理ユニットおよび/またはサブ処理ユニットは、好ましくは、それらの温度をモニタする、または、選択的に、現在の温度を推定する温度センサ(温度検知手段)を含む。他の実施の形態にしたがって、各々の処理要素は、好ましくは、温度のデジタル評価を提供するための温度センサ、およびアナログ-デジタルA/Dコンバータを含む。処理要素上の各々のカーネルは、好ましくは、それ自身のデジタル化された温度をいつでも読み込むことができる。処理要素、処理ユニットおよび複数のサブ処理ユニットは、望ましくは各々が熱閾値 T_{max} を有し、そして、それはコンポーネントごとに異なり得る。熱センサが利用できない場合、現在の温度は、タスクおよび現在の冷却特性の熱特性によって算出されることができる。

【0079】

スケジューラ502はまた、キューを使用することなく演算を管理することができる。演算はメモリに格納されることができる、そして、スケジューラ502は、熱特性によって、いくつかの演算をプロセッサに割り当てることができる。例えば、2つの演算がある場合、スケジューラは、熱特性に基づいて、2つの演算を2つの別々の処理要素300(または他の処理デバイス)に割り当てることができる。演算は、別々のメモリ(または単一のメモリの別々の部分)に格納されることができる。第1演算は第1メモリ(または単一のメモリの第1部分)に格納されることができ、そして、第2演算は第2メモリ(または単一のメモリの第2部分)に格納されることができる。それは、2つの演算を同時に格納することは必要でない。むしろ、それらは異なる期間の間、同一のまたは異なるメモリに格納されることができる(そして、不変のまたは可変的な、連続的または不連続な期間の間に交互に生じることができる)。さらに、2つのメモリ(または単一のメモリの2つ

10

20

30

40

50

の部分)が必ずしも、特定の演算または特定の熱特性と関連した演算に限られる専用のメモリではないと理解されなければならない。このように、第1メモリ(または単一のメモリの第1部分)は第2演算を格納することができる、そして、第2メモリ(または単一のメモリの第2部分)は第1演算を格納することができる。同様に、第1および第2キュー504、506が同じように演算することができる」と理解されなければならない。

【0080】

図6は、演算を得て処理する、好ましい処理の工程系統図600を示す。ステップ602で、処理要素、処理ユニットまたはサブ処理ユニットは、その現在の温度が、熱閾値 T_{max} より高いかどうか決定する。 T_{max} を越えない場合、処理は次にステップ604へ進み、さもなければ、処理はステップ608へ進む。ステップ604において、演算がホットキュー504から利用可能かどうか決定される。演算が利用できる場合、処理はステップ606へ進む、さもなければ、処理はステップ608へ進む。ステップ606において、処理要素、処理ユニットまたはサブ処理ユニットは、「ホット」演算を得て、それを実行する。演算終了後、処理はステップ602に戻る。ステップ608において、演算がクールキュー506から利用可能かどうか決定される。演算が利用できる場合、処理はステップ610へ進む。一方、処理はステップ612へ進む。ステップ610において、処理要素、処理ユニットまたはサブ処理ユニットは、「クール」演算を得て、それを実行する。演算終了後、処理はステップ602にもどる。タスクが処理に利用可能でない場合、処理は、ステップ602に戻る前に、アイドルするか、または一定の時間(例えば、予め定められた数のサイクル)、ステップ612で、「nop」を実行することができる。ホットおよびクールタスクが利用でき、そして、 T_{max} を越えない場合、図5に関して上記で論じたように、任意に、ホットおよびクールタスクのいずれかが選択されることができる。このように、工程系統図600によって見られるように、処理デバイスは、ホットおよびクールキュー504、506からタスクを選択することによって、ホットスポットおよびオーバーヒートを避けることができる。この処理は一つ以上の処理デバイスによって並行して実行されることができ、それによって、クロックスピードを変えること、または処理デバイスをシャットダウンすることなく、命令およびタスクの実行をすることができる。

【0081】

図7に示すように、優先度キューでホットおよびクールキューの使用を結合することが可能である。この図において、手順540をスケジューリングするマルチキューが提供される。スケジューラ542は、高優先度キュー544、中優先度キュー546、および低優先度キュー548の3つの優先度キューと関連する。但し、キューの異なる優先度レベルおよび数が採用されることができ、スケジューラ542は、スケジューラ502に関して上記の通りに演算する。優先度キュー544、546および548の各々は、好ましくは、図5に関する上記と同一の方法で作成され、演算するホットキューおよびクールキューを含む。例えば、高優先度キュー544は、タスク $H_{1H} \dots H_{NH}$ を扱うためのホットキュー、およびタスク $C_{1H} \dots C_{NH}$ を扱うためのクールキューを有する。同様に、中優先度キューは、タスク $H_{1M} \dots H_{NM}$ を扱うためのホットキュー、およびタスク $C_{1M} \dots C_{NM}$ を扱うためのクールキューを有する。低優先度キューは、タスク $H_{1L} \dots H_{NL}$ を扱うためのホットキュー、およびタスク $C_{1L} \dots C_{NL}$ を扱うためのクールキューを有する。

【0082】

図8は、優先度キューを使用するとき、演算を得て処理する好ましい処理の工程系統図800を示す。まず最初に、ステップ801で、処理要素、処理ユニットまたはサブ処理ユニットは、例えば高優先度キュー544、中優先度キュー546、低優先度キュー548など、どんな優先度キューを使用すべきかについて決定する。ステップ802で、処理要素、処理ユニットまたはサブ処理ユニットは、その現在の温度が熱閾値 T_{max} より上にあるかどうか決定する。 T_{max} を越えない場合、処理は次にステップ804へ進む、さもなければ、処理はステップ808へ進む。ステップ804において、演算が選択

された優先度キューのホットキュー504から利用可能かどうか決定される。演算が利用できる場合、処理はステップ806へ進む、さもなければ、処理はステップ808へ進む。ステップ806において、処理要素、処理ユニットまたはサブ処理ユニットは、「ホット」演算を得て、それを実行する。演算終了後、処理はステップ801に戻る。ステップ808において、演算が選択された優先度キューのクールキュー506から利用可能かどうか決定される。演算が利用できる場合、処理はステップ810へ進む、さもなければ、処理はステップ812へ進む。ステップ810において、処理要素、処理ユニットまたはサブ処理ユニットは、「クール」演算を得て、それを実行する。演算終了後、処理はステップ801にもどる。タスクが処理に利用可能でない場合、処理はステップ801に戻る前にステップ812でアイドルする。ホットおよびクールタスクが与えられた優先度レベルに利用でき、そして、 T_{max} を越えない場合、任意に、ホットおよびクールタスクのいずれかがその優先度レベルに選択されることができる。このように、工程系統図800によって見られるように、処理コンポーネントは、さまざまな優先度キュー544、546および548のホットおよびクールキューからタスクを選択することによって、実行時間の間、ホットスポットおよびオーバーヒートを避けることができる。この処理は、一つ以上の処理コンポーネントによって並行して実行されることができ、それによって、クロックスピードを変えること、または処理デバイスをシャットダウンすることなく、命令およびタスクの実行をすることができる。変形例において、処理デバイスがあまりに熱くなるか、 T_{max} に近づくかまたは越える場合、それは、より低い優先度キュー（例えば中優先度キュー546または低優先度キュー548）から演算を選択することができ、および/または演算熱特性にかかわらず、減じられたクロックサイクルで演算を実行する。このようなより低い優先度タスクは、減じられたクロックサイクルで実行されることができる。

【0083】

ある状況において、コンポーネントは、演算（例えば、タスク）を実行する前に、熱閾値 T_{max} の下にあってもよいが、それからタスク実行の間、 T_{max} を越えるかもしれない。過去においては、このような場合においては、おそらくコンポーネントをシャットダウンし、それが冷めることができることを必要とする。しかしながら、この課題に対処する技術が開発され、そして、それは特にマルチプロセッサ環境に適している。

【0084】

図9Aは、一群のタスクを走らせている多数の複数の処理要素を示す。この例においては、処理要素2がタスク1のその処理の間にオーバーヒートすると仮定する。処理要素2から他のプロセッサの一つへタスク1を移動することは可能であり、そして、それは、例えば、タスク2および3など、他のタスクを演算していることができる。他のタスクは、好ましくは、より低い優先度タスクであり、そして、処理要素2によって現在実行されているものである。

【0085】

図9Bに示すように、他のプロセッサの、例えばタスク3などのタスクは、「交換」されることができる。このように、処理要素3がタスク1を完了する間、処理要素2はタスクを実行しない。あるいは、2つのプロセッサは、図9Cに示すように、処理要素2がより低い優先度タスクを実行するように、タスクを交換することができる。図9Cに示すように、(1) まず最初に、例えば、処理要素2および処理要素3は、例えば500MHzの標準のクロックスピードで演算することができる。それから、(2) 高優先度タスク1を演算する間に処理要素2が熱くなる場合、そのタスクは、処理要素3の、より低い優先度タスク3と交換されることができる。最後に、(3) より低い優先度タスク3は、より遅い、または減じられたクロックスピード（例えば250MHz）で実行され、処理要素2が冷えることを可能とし、一方、処理要素3は500MHzの標準のクロックスピードで、タスク1の実行を継続する。より高い優先度タスクを実行するためにクロックスピード（例えば、650MHzまで）を上げることもまた可能である。標準の、増加されたおよび

減じられたクロックスピードが単に典型的なものであり、プロセッサ、サブプロセッサおよび/または多重処理システムの最大クロックレートの特定の構造によって、変化することができるという理解されなければならない。最悪のシナリオにおいては、オーバーヒートしているプロセッサは、温度が満足なレベルに達するまで、演算を停止させることができる。しかしながら、マルチプロセッサシステムの他のプロセッサは、リアルタイム演算、および他の重要な演算が迅速に実行されるように、処理を継続する。処理要素が図9A-Cで示されるが、処理ユニットおよび複数のサブ処理ユニット、またはさまざまな処理デバイスの組合せで、同じ演算を実行することは可能である。例えば、オーバーヒートしているサブ処理ユニット308は、その高優先度タスクを処理ユニット304に送信することができ、そして、それは第2サブ処理ユニット308にそのタスクを再設定することができる。同様に、処理ユニット304は、第2サブ処理ユニット308の、より低い優先度タスクを獲得することができ、そしてそれを第1サブ処理ユニット308に割り当てることができる。一旦第1サブ処理ユニット308が冷めると、それは通常のクロックスピードで、高優先度および/または「ホット」タスクを処理することを再開することができる。

10

【0086】

本発明が特定の実施の形態を参照してここに記載されているが、これらの実施の形態が単に本発明の原理および出願を例示するものであると理解されるべきである。したがって、多数の修正が、例示したものである実施の形態になされることが理解されるべきであり、そして、その他の装置は、添付の請求の範囲に記載の本発明の趣旨および

20

【図面の簡単な説明】

【0087】

【図1】本発明の態様による、さまざまな組合せにおいて体系化されたコンポーネントを示す。

【図2A】温度に対する計算デバイスのための時間を表すグラフ図である。

【図2B】温度に対する計算デバイスのための時間を表すグラフ図である。

【図3A】本発明の態様による、処理要素(PE)の典型的な構造を示す図である。

【図3B】本発明の態様による、複数の処理要素の多重処理システムの典型的な構造を示す図である。

30

【図4】本発明の態様による、サブ処理ユニットの典型的な構造を示す図である。

【図5】本発明の態様にしたがってスケジューリングするマルチキューを示す図である。

【図6】本発明の態様による、典型的な動的なスケジューリング方法を示す工程系統図である。

【図7】本発明の態様にしたがってスケジューリングするマルチキューを示す図である。

【図8】本発明の態様による典型的な動的なスケジューリング方法を示す工程系統図である。

【図9A】本発明の態様による、タスク移動(マイグレーション)を示す図である。

【図9B】本発明の態様による、タスク移動(マイグレーション)を示す図である。

【図9C】本発明の態様による、タスク移動(マイグレーション)を示す図である。

40

【図10A】本発明の態様による、コンポーネントおよびコンポーネントと関連する熱値を示す図である。

【図10B】本発明の態様による、コンポーネントおよびコンポーネントと関連する熱値を示す図である。

【図11】本発明の態様による、コンパイラ機能を示す図である。

【符号の説明】

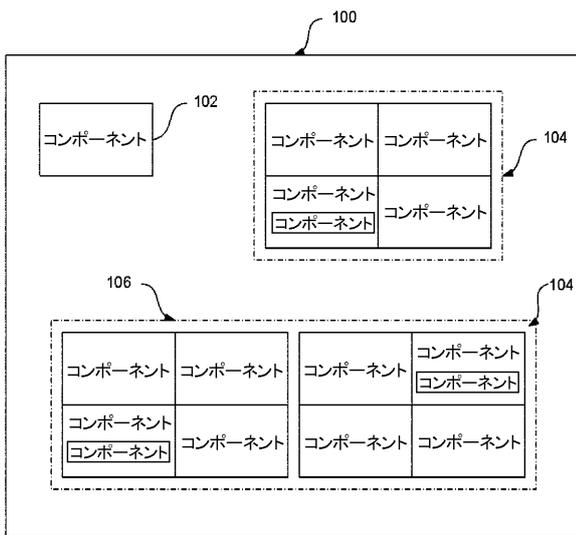
【0088】

100 基板、 102 コンポーネント、 104 ユニット、 106 グループ、
300 処理要素、 304 処理ユニット、 308 複数のサブ処理ユニット、
350 処理要素、 400 サブ処理ユニット、 402 ローカル記憶部、 40

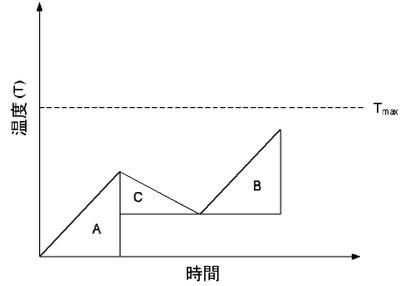
50

4 レジスタ、 502 スケジューラ、 504 ホットキュー、 506 クールキュー。

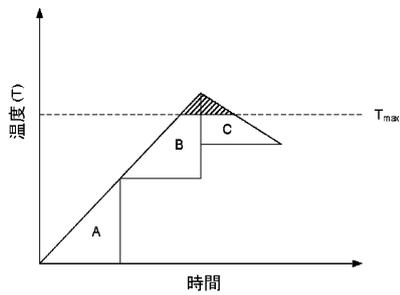
【図1】



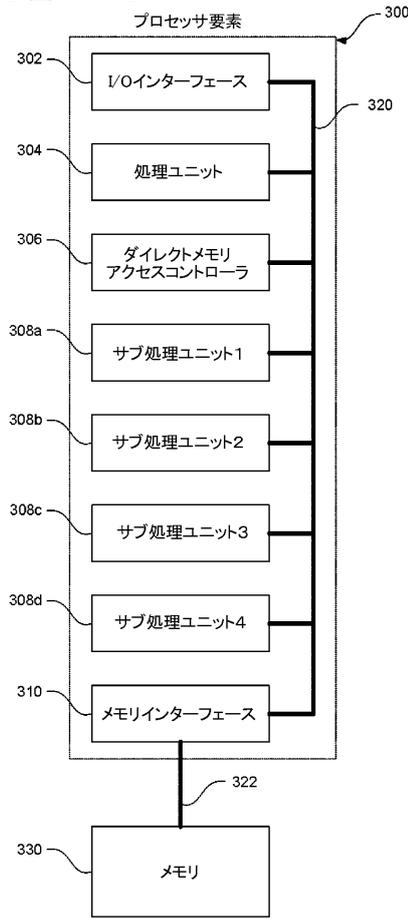
【図2B】



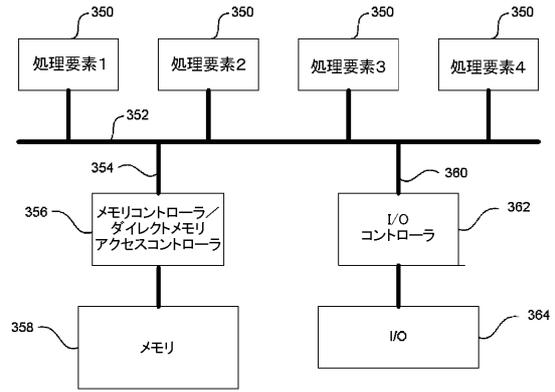
【図2A】



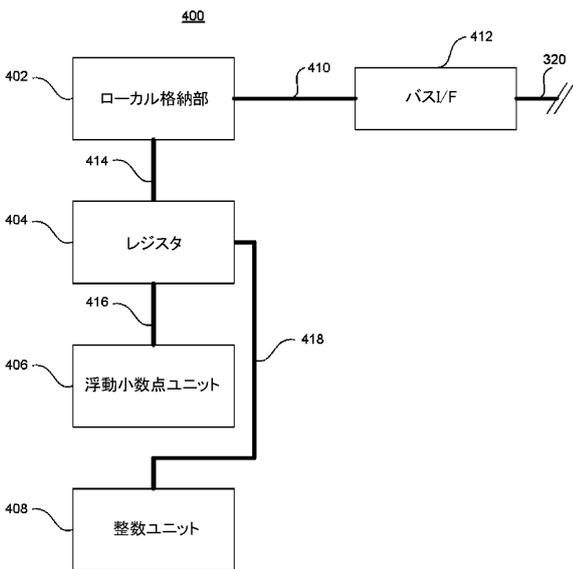
【図3A】



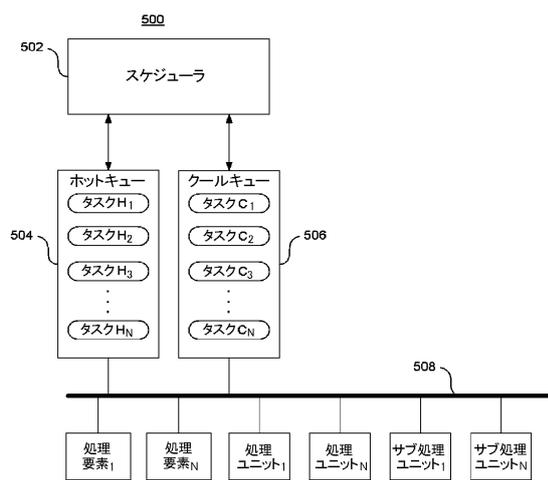
【図3B】



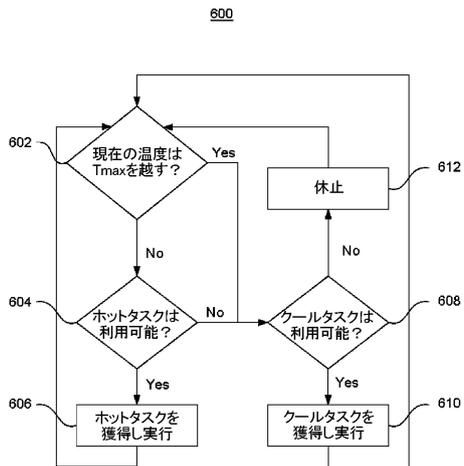
【図4】



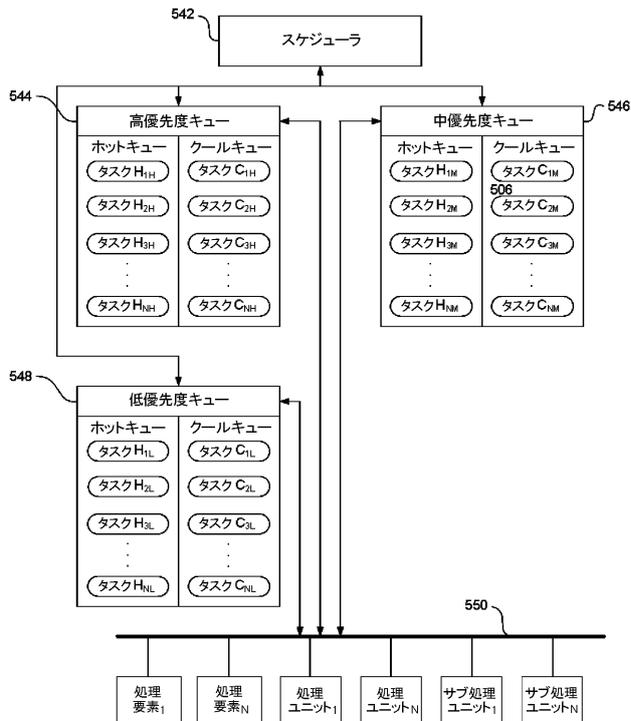
【図5】



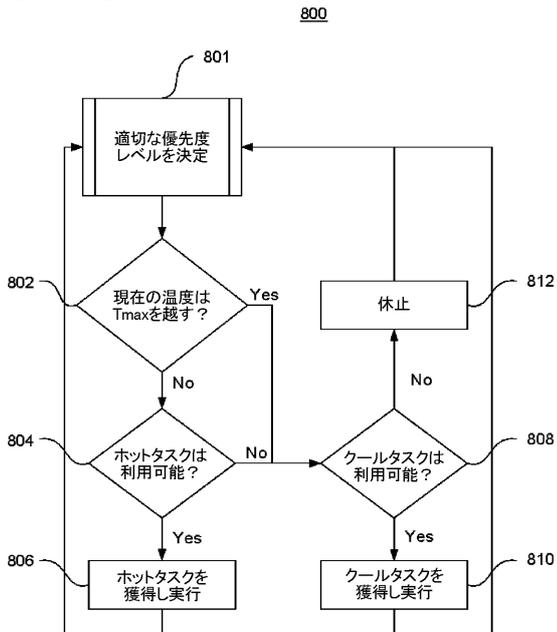
【 図 6 】



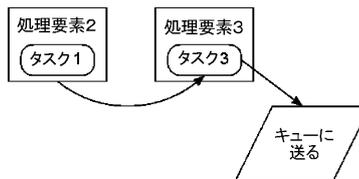
【 図 7 】



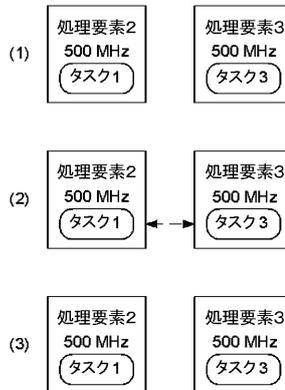
【 図 8 】



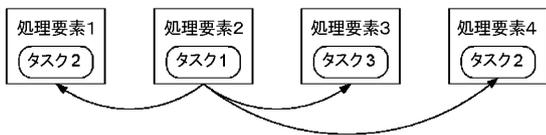
【 図 9 B 】



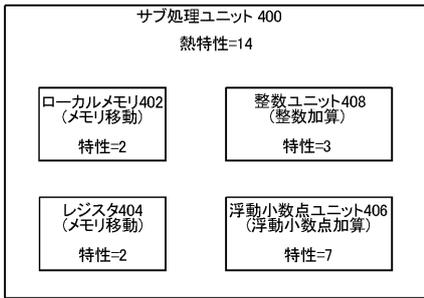
【 図 9 C 】



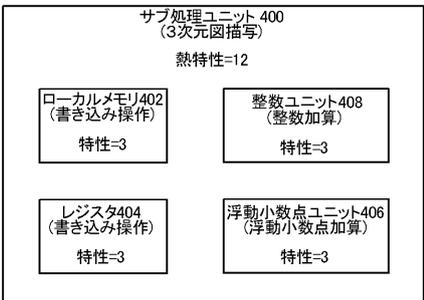
【 図 9 A 】



【 図 1 0 A 】



【 図 1 0 B 】



【 図 1 1 】

