

(19) 日本国特許庁(JP)

再公表特許(A1)

(11) 国際公開番号

W02012/144011

発行日 平成26年7月28日 (2014. 7. 28)

(43) 国際公開日 平成24年10月26日 (2012. 10. 26)

(51) Int.Cl.

G06F 15/167 (2006.01)

F I

G06F 15/167 610B

テーマコード (参考)

5B045

審査請求 有 予備審査請求 未請求 (全 45 頁)

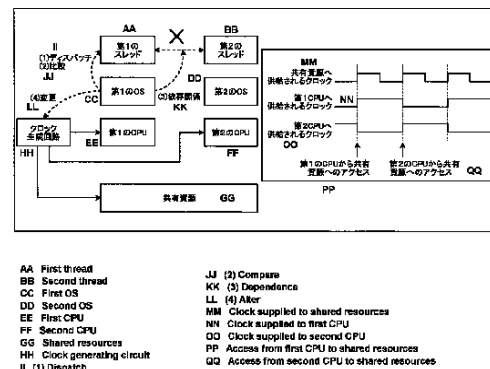
出願番号	特願2013-510762 (P2013-510762)	(71) 出願人	000005223
(21) 国際出願番号	PCT/JP2011/059570		富士通株式会社
(22) 国際出願日	平成23年4月18日 (2011. 4. 18)		神奈川県川崎市中原区上小田中4丁目1番1号
(81) 指定国	AP (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), EA (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OA (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW	(74) 代理人	100104190 弁理士 酒井 昭徳
		(72) 発明者	山内 宏真 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	山下 浩一郎 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	鈴木 貴久 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

最終頁に続く

(54) 【発明の名称】 スレッド処理方法、およびスレッド処理システム

(57) 【要約】

第1のOSが、(1) 第1のスレッドを第1のCPUへディスパッチする。第1のOSが、(2) 第1のスレッドの共有資源へのアクセス頻度を第1閾値と比較する。第1のOSが、(3) 第1のスレッドと第2のCPUで実行中の第2のスレッドとの依存関係を判断する。第1のOSが、(4) アクセス頻度が第1閾値よりも大きいと判断し、かつ依存関係がない(×印)と判断した場合、第1のCPUへ供給されるクロックの位相と第2のCPUへ供給されるクロックの位相とが逆相となるようにクロック生成回路の設定を変更する。第1のOSが、(4) 共有資源へ供給されるクロックの周波数を第1のCPUへ供給されるクロックの周波数の2倍になるようにクロック生成回路の設定を変更する。



【特許請求の範囲】**【請求項 1】**

第 1 スレッドの実行命令を第 1 プロセッサに供給し、
前記第 1 スレッドと第 2 プロセッサで実行される第 2 スレッドとの依存関係を判断し、
前記第 1 スレッドの共有メモリまたは共有キャッシュメモリへのアクセス頻度を第 1 閾値と比較し、

前記アクセス頻度が第 1 閾値よりも大きいとき、かつ前記依存関係がないと判断される
ときは、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、
を特徴とするスレッド処理方法。

【請求項 2】

前記第 1 動作クロックの位相を前記第 2 プロセッサの第 2 動作クロックと逆相になるように変更すること、
を特徴とする請求項 1 に記載のスレッド処理方法。

【請求項 3】

前記第 1 動作クロックの位相を変更するとき、前記共有メモリまたは共有キャッシュメモリに供給される動作クロックの周波数を変更すること、
を特徴とする請求項 1 または請求項 2 に記載のスレッド処理方法。

【請求項 4】

前記動作クロックの周波数を 2 倍に変更すること、
を特徴とする請求項 3 に記載のスレッド処理方法。

【請求項 5】

前記第 1 スレッドを実行するときの第 1 負荷を第 2 閾値と比較し、
前記第 1 負荷が前記第 2 閾値よりも大きく、前記アクセス頻度が第 1 閾値よりも大きく、かつ前記依存関係がないと判断されるときは、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、
を特徴とする請求項 1 乃至請求項 4 の何れかーに記載のスレッド処理方法。

【請求項 6】

前記依存関係があるときは、前記第 1 動作クロックの位相を変更しないこと、
を特徴とする請求項 1 乃至請求項 5 の何れかーに記載のスレッド処理方法。

【請求項 7】

前記依存関係がないとき、前記第 1 動作クロックの第 1 位相から第 2 位相へ変更されていた場合には、前記第 1 動作クロックの位相を前記第 2 位相から前記第 1 位相に変更すること、
を特徴とする請求項 1 乃至請求項 6 の何れかーに記載のスレッド処理方法。

【請求項 8】

前記依存関係は、前記共有メモリまたは前記第 1 プロセッサのメモリに登録されること、
を特徴とする請求項 1 乃至請求項 7 の何れかーに記載のスレッド処理方法。

【請求項 9】

依存関係があるスレッド間を関連付けるテーブルへアクセス可能な複数のコアのうちの
一のコアが、

前記複数のコアの各コアで実行中のスレッド間に直接的または間接的に依存関係がある
か否かを、前記テーブルを参照することで判断し、

判断結果に基づいて、前記複数のコアを直接的または間接的に依存関係があるスレッド
を実行するコア群ごとにグループ化し、

前記グループ化したコア群が複数ある場合、前記コア群ごとに前記コア群へ供給するク
ロックの位相を前記コア群間で非同期となる位相に設定し、

前記複数のコアで共有する共有資源へ供給するクロックの周波数をいずれのコア群とも
同期する周波数に設定すること、

を特徴とするスレッド処理方法。

10

20

30

40

50

【請求項 10】

第 1 プロセッサと、
前記第 1 プロセッサと第 2 プロセッサとがアクセス可能な共有メモリまたは共有キャッシュメモリと、
前記第 1 プロセッサまたは前記共有メモリに供給される動作クロックを制御するクロック制御ユニットと、
を含み、
前記クロック制御ユニットは、
前記第 1 プロセッサで実行される第 1 スレッドの前記共有メモリまたは前記共有キャッシュメモリへのアクセス頻度と第 1 閾値との比較結果と前記第 1 スレッドと前記第 2 プロセッサが実行する第 2 スレッドとの依存関係とに基づいて、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、
を特徴とするスレッド処理システム。

10

【請求項 11】

前記クロック制御ユニットは、前記比較結果が前記アクセス頻度が前記第 1 閾値よりも大きいことを示すとともに前記依存関係があるときに、前記第 1 動作クロックの位相を変更すること、
を特徴とする請求項 10 に記載のスレッド処理システム。

【請求項 12】

前記クロック制御ユニットは、前記第 1 プロセッサが前記第 1 スレッドを実行するときの第 1 負荷と第 2 閾値との比較結果に基づいて、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、
を特徴とする請求項 10 または請求項 11 に記載のスレッド処理システム。

20

【請求項 13】

前記クロック制御ユニットは、前記第 1 動作クロックの位相を前記第 2 プロセッサの第 2 動作クロックと逆相になるように変更すること、
を特徴とする請求項 10 乃至請求項 12 の何れか一に記載のスレッド処理システム。

【請求項 14】

前記クロック制御ユニットは、前記第 1 動作クロックの位相を変更するとき、前記共有メモリまたは共有キャッシュメモリに供給される動作クロックの周波数を変更すること、
を特徴とする請求項 10 乃至請求項 12 の何れか一に記載のスレッド処理システム。

30

【請求項 15】

前記クロック制御ユニットは、前記動作クロックの周波数を 2 倍に変更すること、
を特徴とする請求項 14 に記載のスレッド処理システム。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、スレッドを処理するスレッド処理方法、およびスレッド処理システムに関する。

【背景技術】

40

【0002】

マルチコアプロセッサシステムでは、複数の CPU が複数のスレッドを動作させる。メモリやバスなどの複数の CPU で共有する共有資源では、複数の CPU から同時にアクセスを受け付ける場合がある。共有資源では、たとえば、同時に複数のアクセスに関する処理を行うことができないため、各アクセスの優先度を判断し、アクセスに関する処理を順に行っている。

【0003】

また、フォールトトレラントシステムにおいて、故障モードになると、すべての CPU の動作を同一にする技術が知られている。すなわち、複数の CPU から共有資源へのアクセスが競合するが、共有資源の同一のデータにアクセスする。そこで、マスタ CPU には

50

共有資源へアクセスさせ、スレーブCPUには共有資源へアクセスさせずに、マスタCPUのアクセス結果をスレーブCPUに返すために、マスタCPUに供給されるクロックの位相と逆位相のクロックをスレーブCPUに供給させる。そして、スレーブCPUから共有資源へのアクセスのタイミングをマスタCPUから共有資源へのアクセスのタイミングよりも遅らせる技術が知られている（たとえば、特許文献1参照。）。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特表2008-518311号公報

【発明の概要】

10

【発明が解決しようとする課題】

【0005】

しかしながら、従来技術では、故障モードでない場合に、共有資源へ同時に2つのアクセスが発生した場合、いずれか一方のアクセスに関する処理は待機させられることになるという問題点がある。

【0006】

本発明は、上述した従来技術による問題点を解消するため、複数のCPUから共有資源へのアクセスが衝突することを減少させることができるスレッド処理方法、およびスレッド処理システムを提供することを目的とする。

【課題を解決するための手段】

20

【0007】

上述した課題を解決し、目的を達成するため、本発明の一側面によれば、第1スレッドの実行命令を第1プロセッサに供給し、前記第1スレッドと第2プロセッサで実行される第2スレッドとの依存関係を判断し、前記第1スレッドの共有メモリまたは共有キャッシュメモリへのアクセス頻度を第1閾値と比較し、前記アクセス頻度が第1閾値よりも大きいとき、かつ前記依存関係がないと判断されるときは、前記第1プロセッサの第1動作クロックの位相を変更するスレッド処理方法、およびスレッド処理システムが提案される。

【発明の効果】

【0008】

本発明の一側面によれば、共有資源へのアクセスが衝突することを減少させることができるという効果を奏する。

30

【図面の簡単な説明】

【0009】

【図1】図1は、本発明の一実施例を示す説明図である。

【図2】図2は、マルチコアプロセッサシステムの一例を示すブロック図である。

【図3】図3は、クロック生成回路204の一例を示している。

【図4】図4は、アプリケーションテーブルの一例を示す説明図である。

【図5】図5は、割当テーブルの一例を示す説明図である。

【図6】図6は、OS231-iの機能例を示すブロック図である。

【図7】図7は、例1を示す説明図である。

40

【図8】図8は、例2を示す説明図である。

【図9】図9は、例3を示す説明図である。

【図10】図10は、例4を示す説明図である。

【図11】図11は、OS231-iによるスレッド処理手順の一例を示すフローチャート（その1）である。

【図12】図12は、OS231-iによるスレッド処理手順の一例を示すフローチャート（その2）である。

【図13】図13は、OS231-1によるスレッド処理手順の一例を示すフローチャート（その1）である。

【図14】図14は、OS231-1によるスレッド処理手順の一例を示すフローチャート

50

ト（その２）である。

【図１５】図１５は、クロック生成回路２０４による変更処理手順の一例を示すフローチャートである。

【発明を実施するための形態】

【００１０】

以下に添付図面を参照して、本発明にかかるスレッド処理方法、およびスレッド処理システムの実施の形態を詳細に説明する。本実施の形態では、スレッド処理システムの一例として、マルチコアプロセッサシステムを挙げる。ここで、マルチコアプロセッサシステムにおいて、マルチコアプロセッサとは、コアが複数搭載されたプロセッサである。コアが複数搭載されていれば、複数のコアが搭載された単一のプロセッサでもよく、シングル

10

コアのプロセッサが並列されているプロセッサ群でもよい。なお、本実施の形態では、説明を単純化するため、シングルコアのプロセッサが並列されているプロセッサ群を例に挙げて説明する。

【００１１】

図１は、本発明の一実施例を示す説明図である。第１のＯＳが、（１）第１のスレッドを第１のＣＰＵへディスパッチする。第１のＯＳが、（２）第１のスレッドの共有資源へのアクセス頻度を第１閾値と比較する。ここで、共有資源には、たとえば、共有メモリ、共有キャッシュメモリやバスなどが挙げられる。第１のＯＳが、（３）第１のスレッドと第２のＣＰＵで実行中の第２のスレッドとの依存関係を判断する。ここで、スレッドは、周知のようにアプリケーションプログラム内での処理の実行単位である。依存関係がある

20

とは、たとえば、第１のスレッドと第２のスレッドとでデータを共有している場合が挙げられる。たとえば、ＣＰＵ間でキャッシュコヒーレントを取るか否かをアプリケーションの設計時に設計者があらかじめ解析しておくこととする。

【００１２】

第１のＯＳが、（４）アクセス頻度が第１閾値よりも大きいと判断し、かつ依存関係がない（×印）と判断した場合、第１のＣＰＵへ供給されるクロックの位相と第２のＣＰＵへ供給されるクロックの位相とが逆相となるようにクロック生成回路の設定を変更する。第１のＯＳが、（４）共有資源へ供給されるクロックの周波数を第１のＣＰＵへ供給されるクロックの周波数の２倍になるようにクロック生成回路の設定を変更する。

【００１３】

30

たとえば、第１のＣＰＵと第２のＣＰＵと共有資源とがクロックの立ち上がりエッジで処理が行われるとすると、第１のＣＰＵと第２のＣＰＵとの共有資源へのアクセスが衝突しない。

【００１４】

（マルチコアプロセッサシステム）

図２は、マルチコアプロセッサシステムの一例を示すブロック図である。マルチコアプロセッサシステム２００は、ＣＰＵ２０１－１～２０１－Ｎ（ここでは、 $N=4$ を例に挙げている。）と、スヌープ回路２０２と、２次キャッシュ２０３と、クロック生成回路２０４と、を有している。さらに、マルチコアプロセッサシステム２００は、ディスプレイ２０５と、キーボード２０６と、Ｉ／Ｆ２１２（Interface）と、メモリコントローラ２０７と、共有メモリ２０８と、を有している。

40

【００１５】

２次キャッシュ２０３と、クロック生成回路２０４と、ディスプレイ２０５と、キーボード２０６と、Ｉ／Ｆ２１２と、メモリコントローラ２０７とは、バス２１３を介して接続されている。ＣＰＵ２０１－１～２０１－４は、２次キャッシュ２０３を介して各部に接続されている。共有メモリ２０８は、メモリコントローラ２０７を介して各部に接続されている。

【００１６】

ここで、ＣＰＵ２０１－１～２０１－４は、それぞれレジスタとコアと１次キャッシュ２２１－１～２２１－４を有している。コアは、演算機能を有している。各ＣＰＵ内のレ

50

ジスタは、P C (P r o g r a m C o u n t e r) やリセットレジスタを有している。

【0017】

C P U 2 0 1 - 1 はマスタC P Uであり、マルチコアプロセッサシステム200の全体の制御を司り、O S 2 3 1 - 1を実行する。O S 2 3 1 - 1はマスタO Sであり、C P U 2 0 1 - 1に割り当てられたスレッドを実行する。O S 2 3 1 - 1はスケジューラを有し、スケジューラは起動指示を受け付けたアプリケーションをマルチコアプロセッサのうちのいずれのC P U 2 0 1に割り当てるかを制御する機能を有している。スケジューラはC P U 2 0 1 - 1に割り当てられたアプリケーションの実行順序を制御する機能を有する。

【0018】

C P U 2 0 1 - 2 ~ 2 0 1 - 4 はスレーブC P Uであり、それぞれO S 2 3 1 - 2 ~ 2 3 1 - 4を実行する。O S 2 3 1 - 2 ~ 2 3 1 - 4はスレーブO Sであり、それぞれC P U 2 0 1 - 2 ~ 2 0 1 - 4に割り当てられたスレッドを実行する。O S 2 3 1 - 2 ~ 2 3 1 - 4はそれぞれスケジューラを有し、各スケジューラはそれぞれC P U 2 0 1 - 2 ~ 2 0 1 - 4に割り当てられたアプリケーションの実行順序を制御する機能を有する。

【0019】

ここで、O S 2 3 1 - 1 ~ 2 3 1 - 4はそれぞれランキュー241-1~241-4を有し、ランキュー241-1~241-4にはC P U 2 0 1 - 1 ~ 2 0 1 - 4のそれぞれに割り当てられたアプリケーションのコンテキスト情報のポイントが積まれる。コンテキスト情報とは、たとえば、ロードされたアプリケーションの実行状態や該アプリケーション内の変数などが含まれる情報である。O S 2 3 1 - 1 ~ 2 3 1 - 4はそれぞれランキュー241-1~241-4内のコンテキスト情報のポイントを取得し、アプリケーションのコンテキスト情報にアクセスすることで、アプリケーションを直ぐに実行することができる。

【0020】

1次キャッシュ221-1~221-4は、それぞれキャッシュメモリとキャッシュコントローラとを有している。1次キャッシュ221-1~221-4はそれぞれO S 2 3 1 - 1 ~ 2 3 1 - 4が実行するスレッドから共有メモリ208への書込処理を一時的に記憶する。また、1次キャッシュ221-1~221-4は、共有メモリ208から読み出されたデータを一時的に記憶する。

【0021】

スヌープ回路202は、1次キャッシュ221-1~221-4で共有するデータがいずれかの1次キャッシュ221で更新された場合、該更新を検出し、他の1次キャッシュ221も更新する。

【0022】

2次キャッシュ203は、キャッシュメモリとキャッシュコントローラとを有している。2次キャッシュ203では、1次キャッシュ221-1~221-4から追い出されたデータを記憶する。2次キャッシュ203では、O S 2 3 1 - 1 ~ O S 2 3 1 - 4で共有するデータを記憶する。2次キャッシュ203は、1次キャッシュ221-1~221-4よりも、記憶容量が大きくかつC P U 2 0 1 - 1 ~ 2 0 1 - 4からのアクセス速度が遅い。2次キャッシュ203は共有メモリ208より、記憶容量が小さくかつC P U 2 0 1 - 1 ~ 2 0 1 - 4からのアクセス速度が速い。

【0023】

ディスプレイ205は、カーソル、アイコンあるいはツールボックスをはじめ、文書、画像、機能情報などのデータを表示する。ディスプレイ205は、たとえば、T F T液晶ディスプレイなどを採用することができる。キーボード206は、数字、各種指示などの入力のためのキーを有し、データの入力を行う。また、キーボード206は、タッチパネル式の入力パッドやテンキーなどであってもよい。

【0024】

I / F 2 1 2は、通信回線を通じてL A N (L o c a l A r e a N e t w o r k)、W A N (W i d e A r e a N e t w o r k)、インターネットなどのネットワーク

10

20

30

40

50

に接続され、ネットワークを介して他の装置に接続される。そして、I/F 212は、ネットワークと内部のインターフェースを司り、外部装置からのデータの入出力を制御する。I/F 212には、たとえばモデムやLANアダプタなどを採用することができる。

【0025】

メモリコントローラ207は、共有メモリ208へのアクセスを制御する。共有メモリ208は、たとえば、RAM 209 (Random Access Memory) と、ROM 210 (Read Only Memory) と、フラッシュROM 211を有している。ROM 210は、ブートプログラムなどのプログラムを記憶している。RAM 209は、各CPUのワークエリアとして使用される。フラッシュROM 211は、OS 231-1~231-4などのシステムソフトウェアやアプリケーションのプログラムを記憶している。RAM 209はフラッシュROM 211よりも各CPUからのアクセス速度が速い。各OSがアプリケーションのプログラムをフラッシュROM 211からRAM 209へロードすることにより、該アプリケーションのコンテキスト情報がRAM 209内に展開される。

10

【0026】

クロック生成回路204は、各部にクロックを供給する。クロックCLKが発振回路や外部から入力される基準クロックである。クロックCLK1~CLK4がそれぞれCPU 201-1~201-4へ供給されるクロックである。クロックCLK-2CMが2次キャッシュ203へ供給されるクロックである。また、メモリコントローラ207、共有メモリ208、バス213、I/F 212、ディスプレイ205やキーボード206などにもクロック生成回路204からクロックが共有されているが、図では配線を省略している。

20

【0027】

たとえば、制御信号CPUCTRL1と、制御信号CPUCTRL2と、制御信号CPUCTRL3と、制御信号CPUCTRL4とに応じて、各CPUへ供給されるクロックの位相と2次キャッシュ203へ供給されるクロックの周波数が決定される。

【0028】

図3は、クロック生成回路204の一例を示している。クロック生成回路204は、制御決定回路301と、選択回路302と、インバータ303と、120°遅延回路304と、90°遅延回路305と、選択回路306と、インバータ307と、120°遅延回路308と、を有している。クロック生成回路204は、90°遅延回路309と、選択回路310と、インバータ311と、90°遅延回路312と、通倍回路313と、を有している。

30

【0029】

インバータ303は、クロックCLKを反転して出力することにより、クロックCLKと逆相のクロックを出力する。インバータ307は、クロックCLKを反転して出力することにより、クロックCLKと逆相のクロックを出力する。インバータ311は、クロックCLKを反転して出力することにより、クロックCLKと逆相のクロックを出力する。

【0030】

120°遅延回路304は、クロックCLKの位相を120°ずらす。120°遅延回路308は、120°遅延回路304から出力されるクロックの位相を120°ずらす。90°遅延回路305は、クロックCLKの位相を90°ずらす。90°遅延回路309は、90°遅延回路305から出力されるクロックの位相を90°ずらす。90°遅延回路312は、90°遅延回路309から出力されるクロックの位相を90°ずらす。

40

【0031】

選択回路302は、制御信号CTRL2の値に応じてCPU 201-2に供給するクロックを選択する。選択回路302では、制御信号CTRL2が「1」の場合、クロックCLKが選択される。選択回路302では、制御信号CTRL2が「2」の場合、クロックCLKの位相と逆位相のクロックが選択される。選択回路302では、制御信号CTRL2が「3」の場合、クロックCLKの位相が120°ずらされたクロックが選択される。

50

選択回路 302 では、制御信号 CTRL 2 が「4」の場合、クロック CLK の位相が 90° ずらされたクロックが選択される。

【0032】

選択回路 306 は、制御信号 CTRL 3 の値に応じて CPU 201-3 に供給するクロックを選択する。選択回路 306 では、制御信号 CTRL 3 が「1」の場合、クロック CLK が選択される。選択回路 306 では、制御信号 CTRL 3 が「2」の場合、クロック CLK の位相と逆位相のクロックが選択される。選択回路 306 では、制御信号 CTRL 3 が「3」の場合、クロック CLK の位相が 120° ずらされたクロックが選択される。選択回路 306 では、制御信号 CTRL 3 が「4」の場合、クロック CLK の位相が 240° ずらされたクロックが選択される。選択回路 306 では、制御信号 CTRL 3 が「5」の場合、クロック CLK の位相が 180° ずらされたクロックが選択される。

10

【0033】

選択回路 310 は、制御信号 CTRL 4 の値に応じて CPU 201-4 に供給するクロックを選択する。選択回路 310 では、制御信号 CTRL 4 が「1」の場合、クロック CLK が選択される。選択回路 310 では、制御信号 CTRL 4 が「2」の場合、クロック CLK の位相と逆位相のクロックが選択される。選択回路 310 では、制御信号 CTRL 4 が「3」の場合、クロック CLK の位相が 120° ずらされたクロックが選択される。選択回路 310 では、制御信号 CTRL 4 が「4」の場合、クロック CLK の位相が 240° ずらされたクロックが選択される。選択回路 310 では、制御信号 CTRL 3 が「5」の場合、クロック CLK の位相が 270° ずらされたクロックが選択される。

20

【0034】

通倍回路 313 は、RATE に応じてクロック CLK を通倍にする。通倍回路 313 では、制御信号 RATE の値が「1」の場合、クロック CLK がそのまま出力される。通倍回路 313 では、制御信号 RATE の値が「2」の場合、クロック CLK の周波数の 2 倍の周波数となるクロックが出力される。通倍回路 313 では、制御信号 RATE の値が「3」の場合、クロック CLK の周波数の 3 倍の周波数となるクロックが出力される。通倍回路 313 では、制御信号 RATE の値が「4」の場合、クロック CLK の周波数の 4 倍の周波数となるクロックが出力される。制御信号 RATE の最大値は「4」である。

【0035】

制御決定回路 301 は、制御信号 CPUCTRL 1 から制御信号 CPUCTRL 4 に応じて、各 CPU へ供給されるクロックの位相と 2 次キャッシュ 203 へ供給されるクロックの周波数を決定する。たとえば、制御信号 CPUCTRL 1 から制御信号 CPUCTRL 4 のそれぞれのビット幅が 12 ビット (3 [ビット] × 4 [制御信号]) である。制御信号 CPUCTRL 1 から制御信号 CPUCTRL 4 はそれぞれ上位 3 ビットごとに順に制御信号 CTRL 2 の値、制御信号 CTRL 3 の値、制御信号 CTRL 4 の値、制御信号 RATE の値を示していることとする。

30

【0036】

たとえば、OS 231-1 が、通常時には制御信号 CPUCTRL 1 を「0000 (10 進表記)」に設定し、各 CPU へ供給するクロックの位相や 2 次キャッシュ 203 へ供給するクロックの周波数を変更する際に変化させる。たとえば、制御信号 CPUCTRL 1 の値が「1234 (10 進表記)」に変化させられると、制御決定回路 301 により、制御信号 CTRL 1 の値が「1」となり、制御信号 CTRL 2 の値が「2」となることとする。さらに、制御決定回路 301 により、制御信号 CTRL 3 の値が「3」となり、制御信号 RATE の値が「4」となることとする。また、制御決定回路 301 は、制御信号 CPUCTRL 1 ~ CPUCTRL 4 の「0000」から変更値への変化は検出するが、変更値から「0000」への変化は検出しないこととする。

40

【0037】

(アプリケーションテーブル)

図 4 は、アプリケーションテーブルの一例を示す説明図である。アプリケーションテーブル 400 は、アプリ ID、スレッド ID、依存スレッド ID、アクセス頻度、負荷量の

50

フィールドを有している。

【0038】

アプリIDのフィールドには、各アプリケーションを識別するための識別情報が登録される。スレッドIDのフィールドには、各スレッドを識別するための識別情報が登録される。依存スレッドIDのフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドと依存関係のあるスレッドの識別情報が登録される。

【0039】

アクセス頻度のフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドから2次キャッシュ203への単位時間あたりのアクセス回数が登録される。負荷量のフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドの単位時間当たりの実行時間の比率が登録される。アクセス頻度や負荷量は、各スレッドのコンパイル時の解析およびシミュレーション結果に基づいて登録される。

10

【0040】

フィールドに情報が設定されることで、アプリ情報（たとえば、アプリ情報401-1～401-4）がレコードとして記憶されている。アプリケーションテーブル400は、RAM209、ROM210、フラッシュROM211などの記憶装置に記憶されている。さらに、2次キャッシュ203や1次キャッシュ221などに複製されたアプリケーションテーブル400が記憶されていてもよい。

20

【0041】

図5は、割当テーブルの一例を示す説明図である。割当テーブル500は、アプリID、スレッドID、割当先CPUIDのフィールドを有している。アプリIDのフィールドには、各アプリケーションを識別するための識別情報が登録される。スレッドIDのフィールドには、各スレッドを識別するための識別情報が登録される。割当先CPUIDのフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドの割当先CPUの識別情報が登録される。

【0042】

各フィールドに情報が設定されることで、割当情報（たとえば、割当情報501-1～501-5）がレコードとして記憶されている。割当テーブル500は、RAM209、ROM210、フラッシュROM211などの記憶装置に記憶されている。さらに、2次キャッシュ203や1次キャッシュ221などに複製されたアプリケーションテーブル400が記憶されていてもよい。

30

【0043】

（OS231-i（i=1～4）の機能例）

図6は、OS231-iの機能例を示すブロック図である。OS231-iは、受付部601-iと、判断部602-iと、比較部603-iと、グループ化設定部604-iと、変更部605-iと、実行部606-iと、を有している。受付部601-iから変更部605-iを有するOS231-iがフラッシュROM211などのストレージに記憶されている。CPU201-iが該記憶装置にアクセスして該OS231-iを読み出し、該OS231-i内にコーディングされている処理を実行することにより、該受付部601-iから実行部606-iの処理が実行される。ここでは、i=1を例に挙げて説明する。

40

【0044】

受付部601-1は、イベントを受け付ける。マスタOSが受け付けるイベントとしては、スレッドの生成、スレッドのディスパッチ、スレッドの切り替えやスレッドの実行終了が挙げられる。スレーブOSが受け付けるイベントとしては、スレッドのディスパッチ、スレッドの切り替えやスレッドの実行終了が挙げられる。ここでは、イベントがスレッドの切り替えの場合を例に挙げて説明する。

【0045】

受付部601-1は、スレッドの切り替えを受け付け、判断部602-1は、受付部6

50

01-1により切り替えが受け付けられたスレッド（第1のスレッド）と他のCPUで実行中のそれぞれのスレッドとに依存関係があるか否かを判断する。

【0046】

たとえば、判断部602-1は、アプリケーションテーブル400から、第1のスレッドのIDをスレッドIDのフィールドに有するアプリ情報を検出する。たとえば、判断部602-1は、検出したアプリ情報の依存スレッドIDのフィールドに登録されているスレッドのIDを特定する。これにより、第1のスレッドと依存関係のあるスレッドのIDが特定される。

【0047】

たとえば、判断部602-1は、プロセッサ間通信機能により他のOS231から、実行中のスレッドのIDを取得する。たとえば、判断部602-1は、特定したスレッドのIDと取得したスレッドのIDとを比較する。これにより、第1のスレッドと他のCPUで実行中のそれぞれのスレッドとに依存関係があるか否かが判断される。なお、判断結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0048】

比較部603-1は、第1のスレッドがいずれのCPU201で実行中のスレッドとも依存関係がないと判断部602-1により判断された場合、第1のスレッドの2次キャッシュ203へのアクセス頻度を第1閾値と比較する。

【0049】

たとえば、比較部603-1は、アプリケーションテーブル400から、第1のスレッドのIDをスレッドIDのフィールドに有するアプリ情報を検出する。たとえば、比較部603-1は、検出したアプリ情報のアクセス頻度のフィールドに登録されたアクセス頻度を特定する。たとえば、比較部603-1は、特定したアクセス頻度が第1閾値以上であるか否かを判断する。これにより、第1のスレッドの2次キャッシュ203へのアクセス頻度が第1閾値以上であるか否かが判断される。

【0050】

比較部603-1は、第1のスレッドの2次キャッシュ203へのアクセス頻度が第1閾値以上であると判断した場合、第1のスレッドの負荷量を第2閾値と比較する。たとえば、比較部603-1は、検出したアプリ情報の負荷量のフィールドに登録された第1のスレッドの負荷量が、第2閾値以上であるか否かを判断する。

【0051】

比較部603-1は、第1のスレッドの負荷量が第2閾値以上であると判断した場合、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大であるか否かを判断する。たとえば、比較部603-1は、制御信号RATEの値をクロック生成回路204から取得し、制御信号RATEの値が最大値であるか否かを判断する。上述最大値は4である。

【0052】

判断部602-1は、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大でないと比較部603-1により判断された場合、CPU201-1を除く残余のCPUで実行中のスレッド間に依存関係があるか否かを判断する。たとえば、判断部602-1は、CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドとに依存関係があるか否かを判断する。たとえば、判断部602-1は、CPU201-2で実行中のスレッドとCPU201-4で実行中のスレッドとに依存関係があるか否かを判断する。たとえば、判断部602-1は、CPU201-3で実行中のスレッドとCPU201-4で実行中のスレッドとに依存関係があるか否かを判断する。具体的な判断方法については、CPU201-1で実行中のスレッドと、CPU201-2～201-4のそれぞれで実行中のスレッドとの依存関係の有無を判断した処理と同一処理であるため、詳細な説明は省略する。なお、判断結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0053】

10

20

30

40

50

(例 1)

図 7 は、例 1 を示す説明図である。図 7 では、丸印の中に符号が付されているものが、各符号に即した CPU 201 で実行中のスレッドを示している。CPU 201-1 で切り替えが発生した第 1 のスレッドと CPU 201-2 ~ 201-4 のそれぞれで実行中のスレッドには依存関係がない。そのため、図 7 では、CPU 201-1 で切り替えが発生した第 1 のスレッドと CPU 201-2 ~ 201-4 のそれぞれで実行中のスレッドとを結ぶ矢印間に×印が付されている。

【0054】

図 7 では、CPU 201-2 で実行中のスレッドと CPU 201-3 で実行中のスレッドは依存関係がなく、CPU 201-2 で実行中のスレッドと CPU 201-4 で実行中のスレッドは依存関係がない。CPU 201-3 で実行中のスレッドと CPU 201-4 で実行中のスレッドは依存関係がない。

【0055】

グループ化設定部 604-1 は、依存関係に応じて CPU をグループ化する。ここで、1 グループは、1 CPU 以上であり、直接的または間接的に依存関係があるスレッドを実行中の CPU を 1 グループとする。たとえば、グループ化設定部 604-1 が、各 CPU を異なるグループに設定する。CPU 201-1 はグループ G1 に属し、CPU 201-2 はグループ G2 に属し、CPU 201-3 はグループ G3 に属し、CPU 201-4 はグループ G4 に属する。CPU グループ数 GC は 4 である。なお、グループ化の設定結果は、RAM 209、フラッシュ ROM 211 などの記憶装置に記憶される。

【0056】

変更部 605-1 は、グループ化の設定結果に応じて各 CPU へ供給するクロックの位相を変更し、CPU グループ数 GC に応じて 2 次キャッシュ 203 へ供給するクロックの周波数を変更する。たとえば、変更部 605-1 は、制御信号 CPUCTRL1 の値を 0 から「4554」に変更する。クロック生成回路 204 は、制御信号 CPUCTRL1 の変化を検出する。クロック生成回路 204 は、制御信号 CPUCTRL1 の値に基づいて、制御信号 CTRL2 の値を「4」にし、制御信号 CTRL3 の値を「5」にし、制御信号 CTRL4 の値を「5」にし、制御信号 RATE の値を「4」にする。

【0057】

これにより、CPU 201-1 へ供給されるクロックの位相よりも 90° ($= 360^\circ / G$) ずらされた位相のクロックが CPU 201-2 へ供給される。CPU 201-2 へ供給されるクロックの位相よりも 90° ずらされた位相のクロックが CPU 201-3 へ供給される。CPU 201-3 へ供給されるクロックの位相よりも 90° ずらされた位相のクロックが CPU 201-4 へ供給される。クロック CLK-2CM の周波数がクロック CLK の周波数の 4 倍となる。そして、実行部 606-1 は、変更部 605-1 による変更が終了後、第 1 のスレッドを実行する。すなわち、グループ間で供給されるクロックが非同期となり、クロック CLK-2CM の周波数がいずれのグループとも同期する周波数となる。

【0058】

(例 2)

図 8 は、例 2 を示す説明図である。CPU 201-1 で切り替えが発生した第 1 のスレッドと CPU 201-2 ~ 201-4 のそれぞれで実行中のスレッドには依存関係がない。CPU 201-2 で実行中のスレッドと CPU 201-3 で実行中のスレッドは依存関係がない。CPU 201-2 で実行中のスレッドと CPU 201-4 で実行中のスレッドは依存関係がない。CPU 201-3 で実行中のスレッドと CPU 201-4 で実行中のスレッドは直接的に依存関係がある。×印の矢印間は、依存関係がないことを示し、○印の矢印間は、依存関係があることを示している。

【0059】

グループ化設定部 604-1 は、CPU 201-1 をグループ G1 にし、CPU 201-2 をグループ G2 にし、CPU 201-3 と CPU 201-4 をグループ G3 にする。

CPUグループ数GCは3である。なお、グループ化の設定結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0060】

変更部605-1は、CPUグループ数GCに応じて各CPUへ供給するクロックの位相を変更し、グループ数に応じて2次キャッシュ203へ供給するクロックの周波数を変更する。たとえば、変更部605-1は、制御信号CPUCTRL1の値を0から「3443」に変更する。クロック生成回路204は、制御信号CPUCTRL1の変化を検出する。クロック生成回路204は、制御信号CPUCTRL1の値に基づいて、制御信号CTRL2の値を「3」にし、制御信号CTRL3の値を「4」にし、制御信号CTRL4の値を「4」にし、制御信号RATEの値を「3」にする。

10

【0061】

これにより、CPU201-1へ供給されるクロックの位相よりも120°（=360°/CPUグループ数GC=360°/3）ずらされた位相のクロックがCPU201-2へ供給される。CPU201-2へ供給されるクロックの位相よりも120°ずらされた位相のクロックがCPU201-3へ供給される。CPU201-2へ供給されるクロックの位相よりも120°ずらされた位相のクロックがCPU201-4へ供給される。クロックCLK-2CMの周波数がクロックCLKの周波数の3倍となる。そして、実行部606-1は、変更部605-1による変更が終了後、第1のスレッドを実行する。

【0062】

（例3）

20

図9は、例3を示す説明図である。CPU201-1で切り替えが発生した第1のスレッドとCPU201-2～201-4のそれぞれで実行中のスレッドには依存関係がない。図9では、CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドには依存関係がある。CPU201-3で実行中のスレッドとCPU201-4で実行中のスレッドには依存関係がある。CPU201-2で実行中のスレッドとCPU201-4で実行中のスレッドには直接的に依存関係がないが、CPU201-3を介して間接的に依存関係がある。

【0063】

グループ化設定部604-1は、CPU201-1をグループG1にし、CPU201-2とCPU201-3とCPU201-4をグループG2にする。CPUグループ数GCは2である。

30

【0064】

変更部605-1は、CPUグループ数GCに応じて各CPUへ供給するクロックの位相を変更し、グループ数に応じて2次キャッシュ203へ供給するクロックの周波数を変更する。たとえば、変更部605-1は、制御信号CPUCTRL1の値を0から「2222」に変更する。クロック生成回路204は、制御信号CPUCTRL1の変化を検出する。クロック生成回路204は、制御信号CPUCTRL1の値に基づいて、制御信号CTRL2の値を「2」にし、制御信号CTRL3の値を「2」にし、制御信号CTRL4の値を「2」にし、制御信号RATEの値を「2」にする。

【0065】

40

これにより、CPU201-1へ供給されるクロックの位相よりも180°（=360°/G）ずらされた位相（逆相）のクロックがCPU201-2へ供給される。CPU201-2へ供給されるクロックの位相と逆相のクロックがCPU201-3へ供給される。CPU201-2へ供給されるクロックの位相と逆相のクロックがCPU201-4へ供給される。クロックCLK-2CMの周波数がクロックCLKの周波数の2倍となる。そして、実行部606-1は、変更部605-1による変更が終了後、第1のスレッドを実行する。

【0066】

（例4）

また、判断部602-1は、受付部601-1により切り替えが受け付けられた第1の

50

スレッドと他のCPUで実行中のそれぞれのスレッドとに依存関係があると判断した場合、すべてのCPUで実行中のスレッド間に依存関係があるか否かを判断する。

【0067】

図10は、例4を示す説明図である。CPU201-1で切り替えが発生した第1のスレッドとCPU201-2で実行中のスレッドは直接的に依存関係がある。CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドは直接的に依存関係がある。CPU201-3で実行中のスレッドとCPU201-4で実行中のスレッドは直接的に依存関係がある。

【0068】

CPU201-1で切り替えが発生した第1のスレッドとCPU201-3で実行中のスレッドは、直接的に依存関係がないが、CPU201-2で実行中のスレッドを介して間接的に依存関係がある。CPU201-1で切り替えが発生した第1のスレッドとCPU201-4で実行中のスレッドは、直接的に依存関係がないが、CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドを介して間接的に依存関係がある。

10

【0069】

たとえば、グループ化設定部604-1は、CPU201-1~201-4をグループG1にする。CPUグループ数GCは1である。なお、グループ化の設定結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0070】

変更部605-1は、CPUグループ数GCに応じて各CPUへ供給するクロックの位相を変更し、CPUグループ数GCに応じて2次キャッシュ203へ供給するクロックの周波数を変更する。たとえば、変更部605-1は、制御信号CPUCTRL1の値を0から「1111」に変更する。クロック生成回路204は、制御信号CPUCTRL1の変化を検出する。クロック生成回路204は、制御信号CPUCTRL1の値に基づいて、制御信号CTRL2の値を「1」にし、制御信号CTRL3の値を「1」にし、制御信号CTRL4の値を「1」にし、制御信号RATEの値を「1」にする。

20

【0071】

これにより、クロックCLKと同相のクロックが各CPUへ供給される。クロックCLK-2CMの周波数がクロックCLKの周波数と同一となる。そして、実行部606-1は、変更部605-1による変更が終了後、第1のスレッドを実行する。

30

【0072】

(OS231-iによるスレッド処理手順)

図11および図12は、OS231-iによるスレッド処理手順の一例を示すフローチャートである。OS231-iは、受付部601-iにより、イベントの発生を受け付けたか否かを判断する(ステップS1101)。OS231-iは、イベントの発生を受け付けていないと判断した場合(ステップS1101:No)、ステップS1101へ戻る。OS231-iは、ディスパッチを受け付けると(ステップS1101:ディスパッチ)、ステップS1123へ移行する。

【0073】

OS231-iは、スレッドの実行終了を受け付けたと判断した場合(ステップS1101:実行終了)、実行待ちのスレッドがあるか否かを判断する(ステップS1102)。実行待ちのスレッドがあるか否かについては、ランキュー241-iにコンテキスト情報のポインタが登録されているか否かによって判断される。OS231-iは、実行待ちのスレッドがないと判断した場合(ステップS1102:No)、ステップS1116へ移行する。OS231-iは、実行待ちのスレッドがあると判断した場合(ステップS1102:Yes)、ステップS1103へ移行する。

40

【0074】

ステップS1101において、OS231-iは、スレッドのスイッチを受け付けたと判断した場合(ステップS1101:スイッチ)、ステップS1103へ移行する。ステ

50

ップS 1 1 0 1のスイッチ、またはステップS 1 1 0 2のY e sのつぎに、OS 2 3 1 - iが、判断部6 0 2 - iにより、つぎに実行するスレッド（第1のスレッド）と他のCPUで実行中のスレッドとに依存関係があるか否かを判断する（ステップS 1 1 0 3）。つぎに実行するスレッドとは、スレッドのスイッチの場合にはスイッチ後のスレッドであり、スレッドの実行終了の場合にはランキュー2 4 1 - iの先頭にコンテキスト情報のポインタが登録されているスレッドである。

【0 0 7 5】

OS 2 3 1 - iは、第1のスレッドと他のCPUで実行中のスレッドとに依存関係があると判断した場合（ステップS 1 1 0 3：Y e s）、他のCPUで実行中のスレッド間の依存関係を確認する（ステップS 1 1 0 4）。OS 2 3 1 - iは、すべてのCPUで直接的または間接的に依存関係があるか否かを判断する（ステップS 1 1 0 5）。OS 2 3 1 - iは、すべてのCPUで依存関係があると判断した場合（ステップS 1 1 0 5：Y e s）、すべてのCPUに供給するクロックの位相を同一位相に変更する（ステップS 1 1 0 6）。OS 2 3 1 - iは、2次キャッシュ2 0 3に供給するクロックの周波数をデフォルトに変更し（ステップS 1 1 0 7）、ステップS 1 1 2 3へ移行する。

【0 0 7 6】

ステップS 1 1 0 5において、OS 2 3 1 - iは、すべてのCPUで依存関係があると判断しなかった場合（ステップS 1 1 0 5：N o）、ステップS 1 1 1 2へ移行する。ステップS 1 1 0 3において、OS 2 3 1 - iは、第1のスレッドと他のCPUで実行中のスレッドとに依存関係がないと判断した場合（ステップS 1 1 0 3：N o）、第1のスレッドのメモリアクセス頻度>所定頻度であるか否かを判断する（ステップS 1 1 0 8）。所定頻度は第1閾値である。OS 2 3 1 - iは、第1のスレッドのメモリアクセス頻度>所定頻度でないと判断した場合（ステップS 1 1 0 8：N o）、ステップS 1 1 1 6へ移行する。OS 2 3 1 - iは、第1のスレッドのメモリアクセス頻度>所定頻度であると判断した場合（ステップS 1 1 0 8：Y e s）、第1のスレッドの負荷量>所定負荷量であるか否かを判断する（ステップS 1 1 0 9）。所定負荷量は第2閾値である。

【0 0 7 7】

OS 2 3 1 - iは、第1のスレッドの負荷量>所定負荷量でないと判断した場合（ステップS 1 1 0 9：N o）、ステップS 1 1 1 6へ移行する。OS 2 3 1 - iは、第1のスレッドの負荷量>所定負荷量であると判断した場合（ステップS 1 1 0 9：Y e s）、2次キャッシュ2 0 3へ供給するクロックの周波数の通倍率が最大であるか否かを判断する（ステップS 1 1 1 0）。OS 2 3 1 - iは、2次キャッシュ2 0 3へ供給するクロックの周波数の通倍率が最大であると判断した場合（ステップS 1 1 1 0：Y e s）、ステップS 1 1 2 3へ移行する。OS 2 3 1 - iは、2次キャッシュ2 0 3へ供給するクロックの周波数の通倍率が最大でないと判断した場合（ステップS 1 1 1 0：N o）、判断部6 0 2 - iにより、他のCPUで実行中のスレッド間の依存関係を確認する（ステップS 1 1 1 1）。

【0 0 7 8】

OS 2 3 1 - iは、直接的または間接的に依存関係のあるCPUごとにグループ化し（ステップS 1 1 1 2）、CPUグループ数G Cを決定する（ステップS 1 1 1 3）。OS 2 3 1 - iは、グループ化したCPUごとに該CPUへ供給するクロックの位相を変更し（ステップS 1 1 1 4）、CPUグループ数G Cに基づいて、2次キャッシュ2 0 3に供給するクロックの周波数を変更し（ステップS 1 1 1 5）、ステップS 1 1 2 3へ移行する。

【0 0 7 9】

ステップS 1 1 0 2のN o、ステップS 1 1 0 8のN o、またはステップS 1 1 0 9のN oのつぎに、OS 2 3 1 - iは、比較部6 0 3 - iにより、スイッチ前に実行していたスレッドまたは実行終了したスレッドのメモリアクセス頻度>所定頻度であるかを判断する（ステップS 1 1 1 6）。OS 2 3 1 - iは、スイッチ前に実行していたスレッドまたは実行終了したスレッドのメモリアクセス頻度>所定頻度でないと判断した場合（ステッ

プ S 1 1 1 6 : N o) 、ステップ S 1 1 2 3 へ移行する。OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドのメモリアクセス頻度 > 所定頻度であると判断した場合 (ステップ S 1 1 1 6 : Y e s) 、ステップ S 1 1 1 7 へ移行する。

【 0 0 8 0 】

OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドの負荷量 > 所定負荷量であるか否かを判断する (ステップ S 1 1 1 7) 。OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドの負荷量 > 所定負荷量でないと判断した場合 (ステップ S 1 1 1 7 : N o) 、ステップ S 1 1 2 3 へ移行する。

【 0 0 8 1 】

OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドの負荷量 > 所定負荷量であると判断した場合 (ステップ S 1 1 1 7 : Y e s) 、CPU グループ数 G C をデクリメントする (ステップ S 1 1 1 8) 。OS 2 3 1 - i は、他の CPU で実行中のスレッド間の依存関係を確認し (ステップ S 1 1 1 9) 、減らした CPU グループ数 G C と依存関係に沿って CPU 2 0 1 をグループ化する (ステップ S 1 1 2 0) 。たとえば、依存関係に沿って CPU 数が均等になるように CPU グループ数 G C 分のグループを作成することとしてもよい。

【 0 0 8 2 】

OS 2 3 1 - i は、グループ化した CPU ごとに該 CPU へ供給するクロックの位相を変更する (ステップ S 1 1 2 1) 。OS 2 3 1 - i は、CPU グループ数 G C に基づいて、2 次キャッシュ 2 0 3 に供給するクロックの周波数を変更し (ステップ S 1 1 2 2) 、ステップ S 1 1 2 3 へ移行する。OS 2 3 1 - i は、実行部 6 0 6 - i により、第 1 のスレッドを実行し (ステップ S 1 1 2 3) 、ステップ S 1 1 0 1 へ戻る。ステップ S 1 1 2 3 は、ステップ S 1 1 0 1 のディスパッチ、ステップ S 1 1 0 7 、ステップ S 1 1 1 0 の Y e s 、ステップ S 1 1 1 5 、ステップ S 1 1 1 6 の N o 、ステップ S 1 1 1 7 の N o 、またはステップ S 1 1 2 2 のつぎに実行される。

【 0 0 8 3 】

(OS 2 3 1 - 1 によるスレッド処理手順)

図 1 3 および図 1 4 は、OS 2 3 1 - 1 によるスレッド処理手順の一例を示すフローチャートである。OS 2 3 1 - 1 が、受付部 6 0 1 - i により、スレッドの生成を受け付けたか否かを判断する (ステップ S 1 3 0 1) 。OS 2 3 1 - 1 が、スレッドの生成を受け付けていないと判断した場合 (ステップ S 1 3 0 1 : N o) 、ステップ S 1 3 0 1 へ戻る。

【 0 0 8 4 】

OS 2 3 1 - 1 が、スレッドの生成を受け付けたと判断した場合 (ステップ S 1 3 0 1 : Y e s) 、生成されたスレッドが他の CPU で実行中のスレッドと依存関係があるか否かを判断する (ステップ S 1 3 0 2) 。OS 2 3 1 - 1 が、生成されたスレッドが他の CPU で実行中のスレッドと依存関係があると判断した場合 (ステップ S 1 3 0 2 : Y e s) 、他の CPU で実行中のスレッド間の依存関係を確認する (ステップ S 1 3 0 3) 。OS 2 3 1 - 1 が、すべての CPU で直接的または間接的に依存関係があるか否かを判断する (ステップ S 1 3 0 4) 。OS 2 3 1 - 1 が、すべての CPU で直接的または間接的に依存関係がないと判断した場合 (ステップ S 1 3 0 4 : N o) 、ステップ S 1 3 1 3 へ移行する。

【 0 0 8 5 】

OS 2 3 1 - 1 が、すべての CPU で直接的または間接的に依存関係があると判断した場合 (ステップ S 1 3 0 4 : Y e s) 、すべての CPU に供給するクロックの位相を同一位相に変更する (ステップ S 1 3 0 5) 。OS 2 3 1 - 1 が、2 次キャッシュ 2 0 3 に供給するクロックの周波数をデフォルトに変更し (ステップ S 1 3 0 6) 、負荷が最小の CPU に生成されたスレッドを割り当て (ステップ S 1 3 0 7) 、ステップ S 1 3 0 1 へ戻る。負荷が最小の CPU については、まず、OS 2 3 1 - 1 が、割当テーブル 5 0 0 を用いて各 CPU に割り当てられているスレッドを特定する。OS 2 3 1 - 1 が、アプリケー

ションテーブル400を参照し、特定したスレッドの負荷量を検索する。OS231-1が、CPUごとに割当済のスレッドの検索した負荷量の合計値を算出し、合計値が最小のCPUを負荷が最小のCPUとして特定する。

【0086】

OS231-1が、生成されたスレッドが他のCPUで実行中のスレッドと依存関係がないと判断した場合（ステップS1302：No）、生成されたスレッドのメモリアクセス頻度>所定頻度であるか否かを判断する（ステップS1308）。OS231-1が、生成されたスレッドのメモリアクセス頻度>所定頻度でないとして判断した場合（ステップS1308：No）、ステップS1311へ移行する。

【0087】

OS231-1が、生成されたスレッドのメモリアクセス頻度>所定頻度であると判断した場合（ステップS1308：Yes）、生成されたスレッドの負荷量>所定負荷量であるか否かを判断する（ステップS1309）。OS231-1が、生成されたスレッドの負荷量>所定負荷量でないとして判断した場合（ステップS1309：No）、ステップS1311へ移行する。

【0088】

OS231-1が、生成されたスレッドの負荷量>所定負荷量であると判断した場合（ステップS1309：Yes）、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大であるか否かを判断する（ステップS1310）。OS231-1が、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大であると判断した場合（ステップS1310：Yes）、ステップS1311へ移行する。ステップS1308のNo、ステップS1309のNo、またはステップS1310のYesのつぎに、OS231-1が、負荷が最小のCPUに生成されたスレッドを割り当て（ステップS1311）、ステップS1301へ戻る。

【0089】

ステップS1310において、OS231-1が、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大でないとして判断した場合（ステップS1310：No）、他のCPUで実行中のスレッド間の依存関係を確認する（ステップS1312）。OS231-1が、直接的または間接的に依存関係のあるCPUごとにグループ化し（ステップS1313）、CPUグループ数GCとグループ化されたCPUに基づいて、各CPUへ供給するクロックの位相を変更する（ステップS1314）。OS231-1が、CPUグループ数GCに基づいて、2次キャッシュ203に供給するクロックの周波数を変更し（ステップS1315）、ステップS1301へ戻る。

【0090】

（クロック生成回路204による変更処理手順）

図15は、クロック生成回路204による変更処理手順の一例を示すフローチャートである。まず、クロック生成回路204が、各CPUからの制御命令を受け付けたか否かを判断する（ステップS1501）。制御命令とは、たとえば、制御信号CPUCTRL1からCPUCTRL4のいずれかの「0000」から変更値への変化である。

【0091】

クロック生成回路204が、各CPUからの制御命令を受け付けたと判断した場合（ステップS1501：Yes）、制御信号を変更し（ステップS1502）、ステップS1501へ戻る。ここで、制御信号とは、たとえば、制御信号CTRL1からCTRL4や制御信号RATEである。クロック生成回路204が、各CPUからの制御命令を受け付けていないとして判断した場合（ステップS1501：No）、ステップS1501へ戻る。

【0092】

以上説明したように、スレッド処理方法、およびスレッド処理システムによれば、第1のCPUで実行される第1のスレッドと第2のCPUで実行される第2のスレッドに依存関係がないかを判断する。第1のスレッドと第2のスレッドとに依存関係がなく、かつ第1のスレッドの共有資源へのアクセス頻度が所定頻度以上であれば、第1のCPUへ供給

10

20

30

40

50

するクロックの位相を第2のCPUへ供給するクロックの位相と異なる位相に変更する。

【0093】

第1のスレッドのアクセス頻度が多いほど、第1のスレッドから共有資源へのアクセスが第2のスレッドから共有資源へのアクセスと衝突する可能性が高い。そのため、各CPUへ供給されるクロックの位相が異なることで、複数のCPUから共有資源へのアクセスが衝突することを減少させることができる。また、CPUへ供給されるクロックの位相が異なることで、ピーク電流を減少させることができる。

【0094】

また、第1のCPUへ供給されるクロックの位相と第2のCPUに供給されるクロックの位相とが逆相となるように変更することにより、複数のCPUから共有資源へのアクセスが衝突することを減少させることができる。

10

【0095】

また、第1のCPUへ供給されるクロックの位相を変更するとともに、複数のCPUへ供給するクロックの周波数を上げるのではなく、共有資源へのクロックの周波数を上げる。これにより、消費電力を抑制しつつ、複数のCPUから共有資源へのアクセスが衝突することを減少させることができ、マルチコアプロセッサシステムの性能を向上させることができる。

【0096】

また、共有資源に供給されるクロックの周波数を第1のCPUへ供給されるクロックの周波数の2倍に変更する。複数のCPUから共有資源へのアクセスが衝突することを減少させることができ、複数のCPUから共有資源へのアクセスの集中により発生する待ち時間を削減することができる。

20

【0097】

また、第1のスレッドを実行するときの負荷量を第2閾値と比較し、負荷量が第2閾値よりも大きいときに、依存関係を判断する。第1のスレッドの負荷量が多いほど、第1のスレッドから共有資源へのアクセスが第2のスレッドから共有資源へのアクセスと衝突する可能性が高い。そのため、各CPUへ供給されるクロックの位相が異なることで、複数のCPUから共有資源へのアクセスが衝突することを減少させることができ、最小限に消費電力を抑制することができる。

【0098】

また、第1のスレッドと第2のスレッドとに依存関係があるときは、第1のCPUへ供給されるクロックの位相を変更しない。これにより、第1のスレッドと第2のスレッドとで共有しているデータや排他制御に影響が生じるのを防止することができる。

30

【0099】

また、依存関係がないとき、第1動作クロックの第1位相から第2位相への変更がされている場合、第1動作クロックの位相を第2位相から第1位相に変更する。これにより、複数のCPUから共有資源へのアクセスが衝突することを減少させることができる。

【0100】

また、依存関係は、共有メモリまたは第1プロセッサのメモリに登録される。これにより、第1のスレッドと第2のスレッドとの依存関係を容易に判断することができ、処理時間を短縮化することができる。

40

【0101】

また、依存関係があるスレッドを実行中のCPU群ごとにグループ化する。グループ化したCPU群が複数ある場合、CPU群ごとに該CPU群へ供給されるクロックの位相を異なるCPU群間で非同期となる位相に設定する。共有資源へ供給するクロックの周波数をいずれのCPU群とも同期する周波数に設定する。これにより、マルチコアプロセッサのCPU数が増加しても、アクセスの衝突を減少させることができ、マルチコアプロセッサの性能を向上させることができる。

【0102】

なお、本実施の形態で説明したスレッド処理方法は、あらかじめ用意されたプログラム

50

をパーソナル・コンピュータやワークステーション等のコンピュータで実行することにより実現することができる。本スレッド処理プログラムは、ハードディスク、フレキシブルディスク、CD-ROM、MO、DVD等のコンピュータで読み取り可能な記録媒体に記録され、コンピュータによって記録媒体から読み出されることによって実行される。また本スレッド処理プログラムは、インターネット等のネットワークを介して配布してもよい。

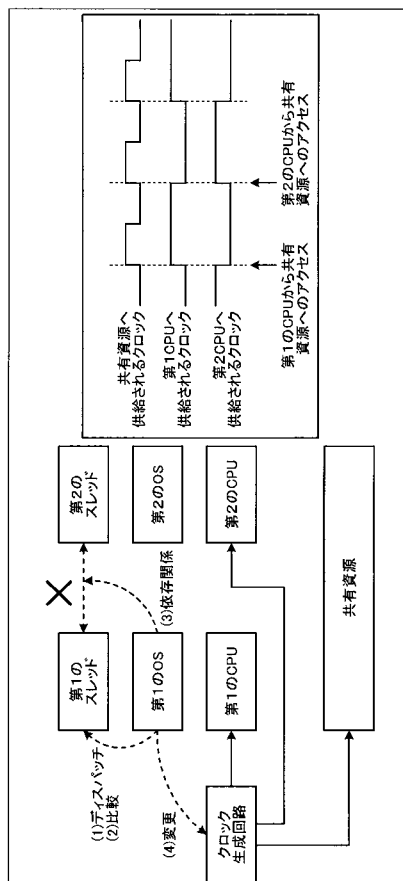
【符号の説明】

【0103】

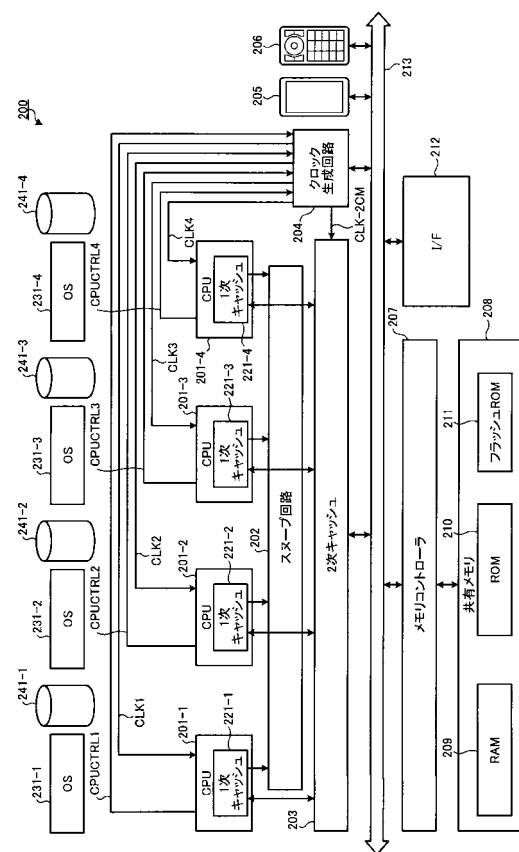
- 200 マルチコアプロセッサシステム
- 201-1~201-4 CPU
- 203 2次キャッシュ
- 204 クロック生成回路
- 208 共有メモリ
- 400 アプリケーションテーブル

10

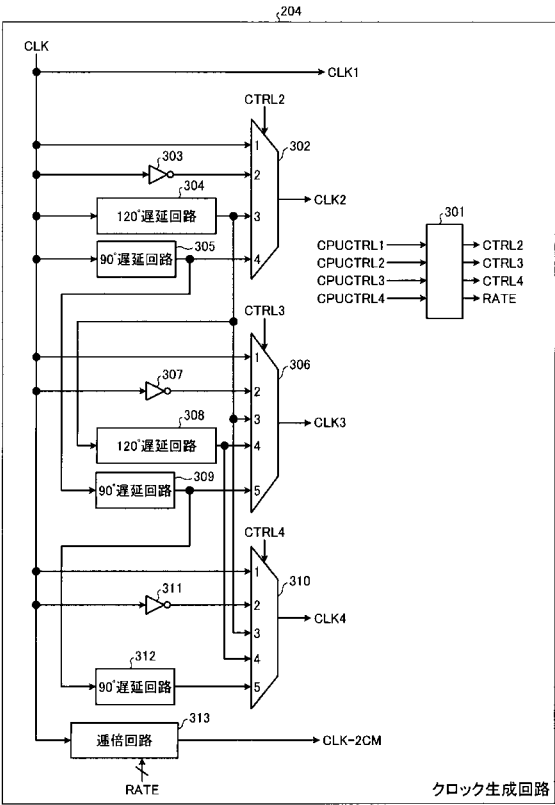
【図1】



【図2】



【図 3】



【図 4】

アプリケーションテーブル 400

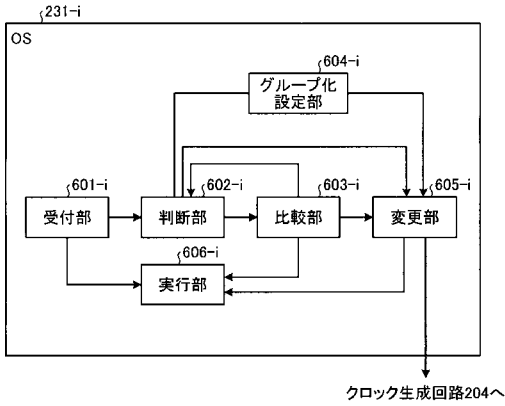
アプリ	スレッド	依存スレッド	アクセス頻度	負荷量
アプリ#1	スレッド#1	スレッド#2	100000[回/s]	50[%]
	スレッド#3	スレッド#3		
アプリ#1	スレッド#2	スレッド#3	90000[回/s]	80[%]
アプリ#1	スレッド#3	スレッド#1	80000[回/s]	40[%]
アプリ#2	スレッド#1	-	17000[回/s]	40[%]
⋮	⋮	⋮	⋮	⋮

【図 5】

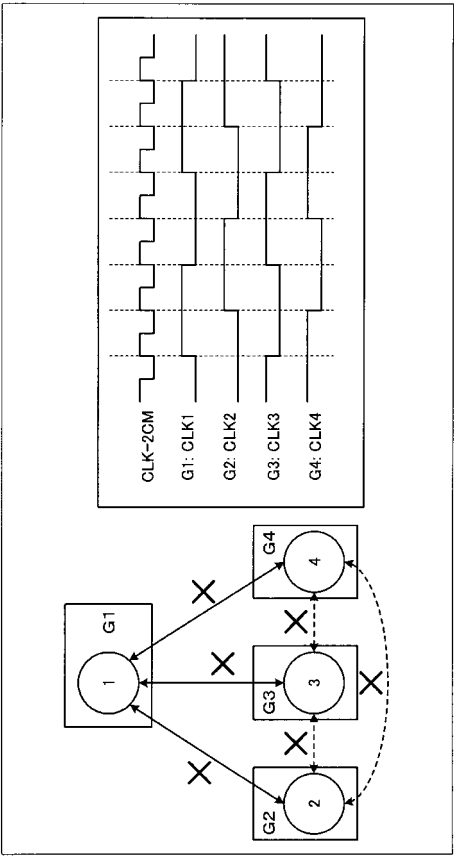
割当テーブル 500

アプリ	スレッド	割当先CPU
アプリ#1	スレッド#1	201-1
アプリ#1	スレッド#2	201-1
...	...	201-2
...	...	201-3
...	...	201-4

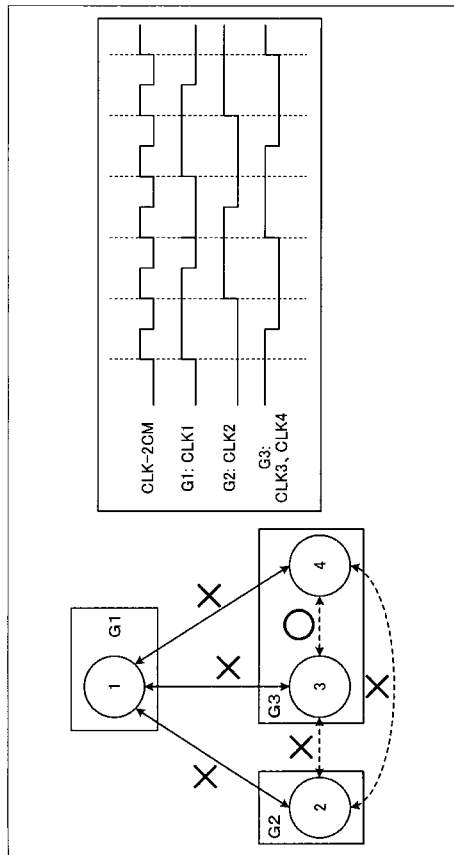
【図 6】



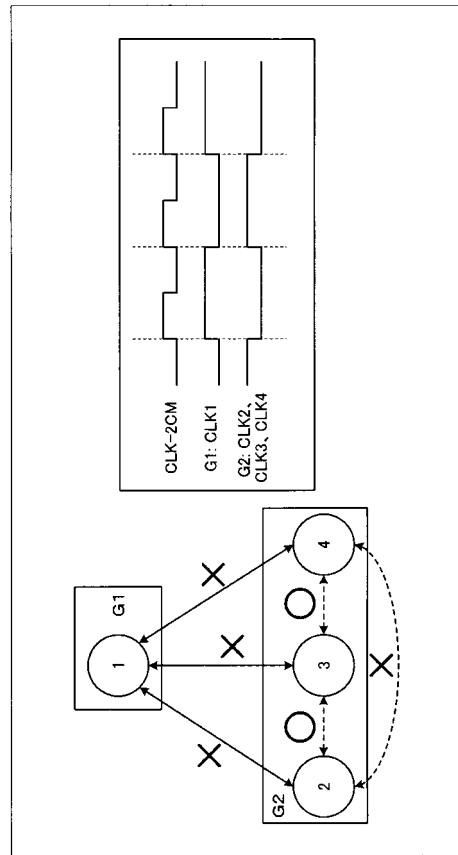
【図 7】



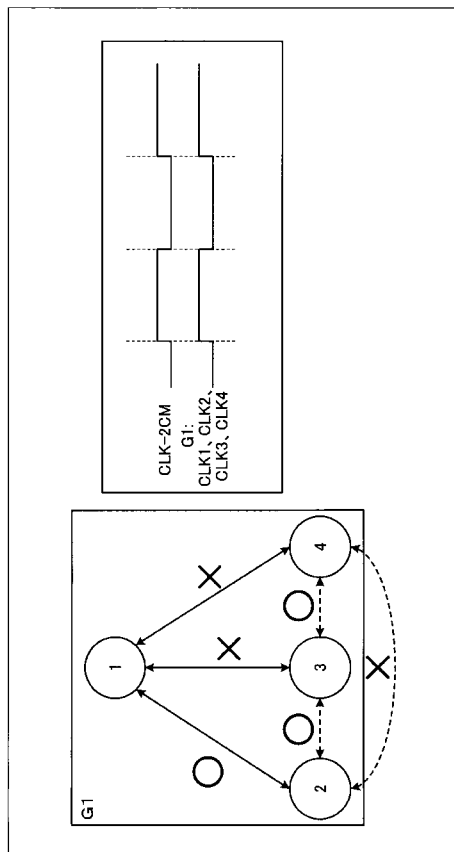
【図 8】



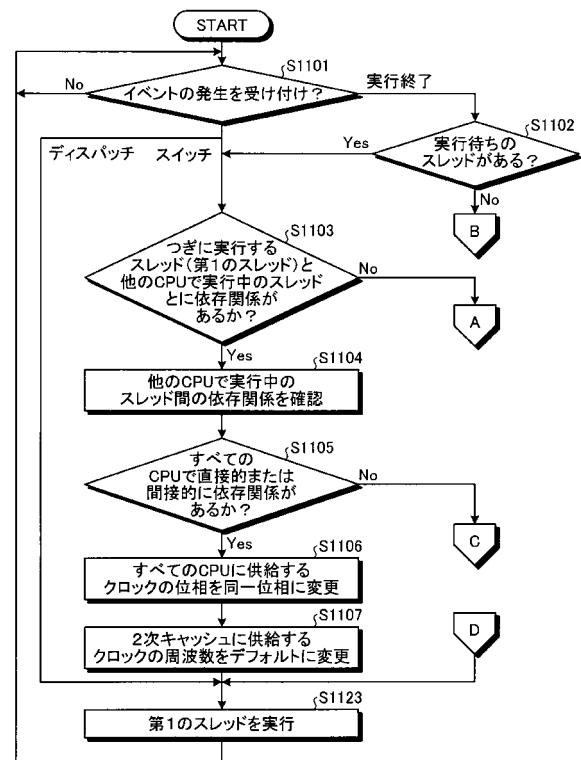
【図 9】



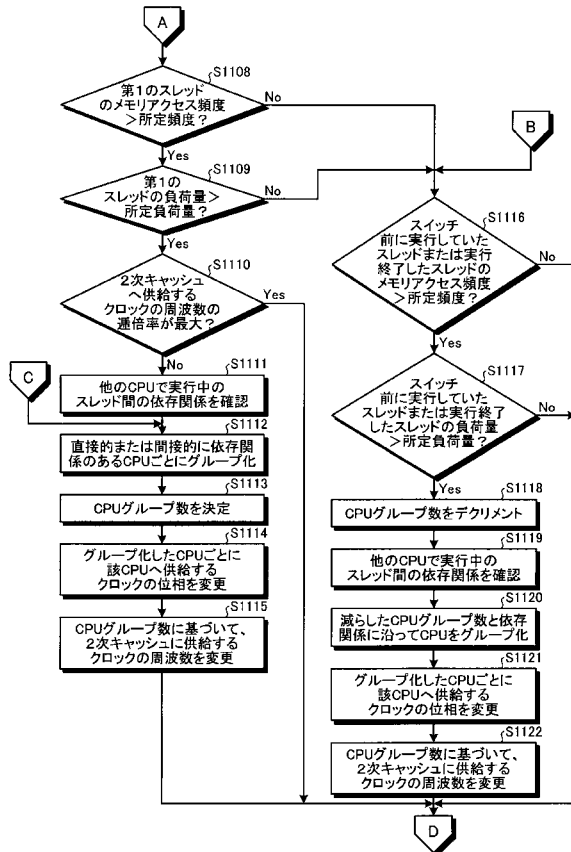
【図 10】



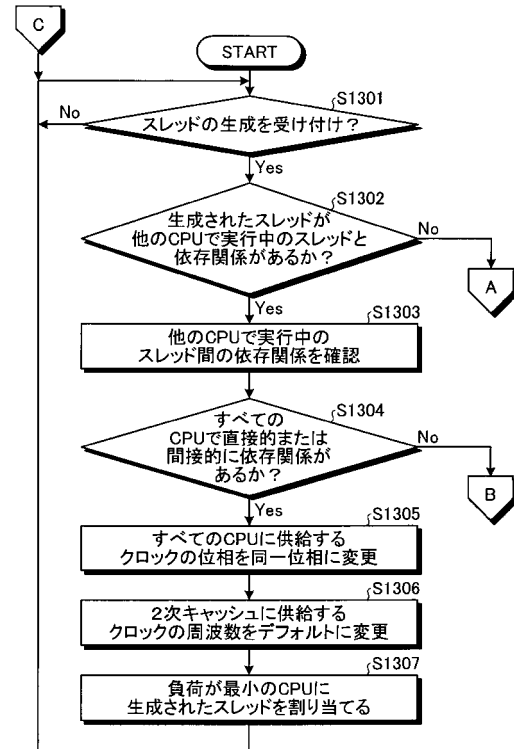
【図 11】



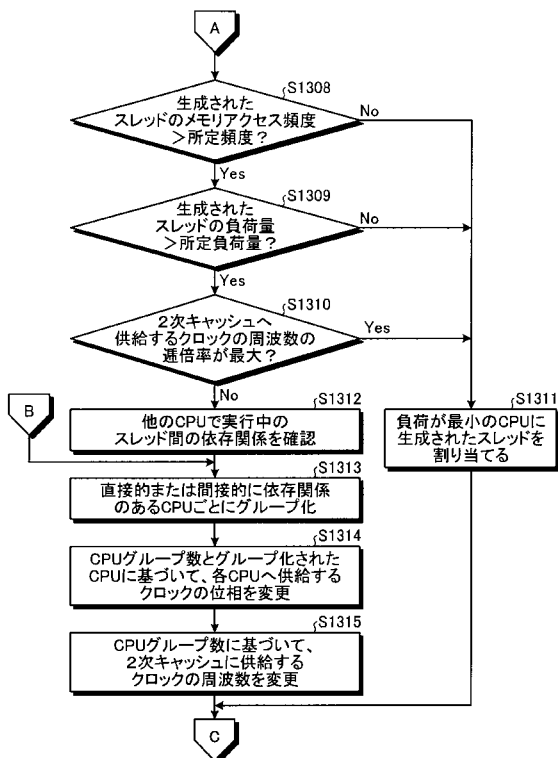
【図 1 2】



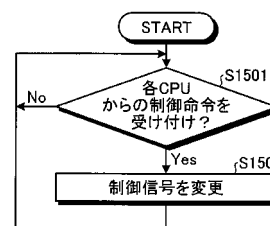
【図 1 3】



【図 1 4】



【図 1 5】



【手続補正書】

【提出日】平成25年10月23日(2013.10.23)

【手続補正 1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項 1】

第 1 スレッドの実行命令を第 1 プロセッサに供給し、
前記第 1 スレッドと第 2 プロセッサで実行される第 2 スレッドとの依存関係を判断し、
前記第 1 スレッドの共有メモリまたは共有キャッシュメモリへのアクセス頻度を第 1 閾値と比較し、
前記アクセス頻度が第 1 閾値よりも大きいとき、かつ前記依存関係がないと判断される
ときは、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、
を特徴とするスレッド処理方法。

【請求項 2】

前記第 1 動作クロックの位相を前記第 2 プロセッサの第 2 動作クロックと逆相になるよ
うに変更すること、
を特徴とする請求項 1 に記載のスレッド処理方法。

【請求項 3】

前記第 1 動作クロックの位相を変更するとき、前記共有メモリまたは共有キャッシュメ
モリに供給される動作クロックの周波数を変更すること、
を特徴とする請求項 1 または請求項 2 に記載のスレッド処理方法。

【請求項 4】

前記動作クロックの周波数を 2 倍に変更すること、
を特徴とする請求項 3 に記載のスレッド処理方法。

【請求項 5】

前記第 1 スレッドを実行するときの第 1 負荷を第 2 閾値と比較し、
前記第 1 負荷が前記第 2 閾値よりも大きく、前記アクセス頻度が第 1 閾値よりも大きく
、かつ前記依存関係がないと判断されるときは、前記第 1 プロセッサの第 1 動作クロック
の位相を変更すること、
を特徴とする請求項 1 乃至請求項 4 の何れか一に記載のスレッド処理方法。

【請求項 6】

前記依存関係があるときは、前記第 1 動作クロックの位相を変更しないこと、
を特徴とする請求項 1 乃至請求項 5 の何れか一に記載のスレッド処理方法。

【請求項 7】

前記依存関係がないとき、前記第 1 動作クロックの第 1 位相から第 2 位相へ変更されて
いた場合には、前記第 1 動作クロックの位相を前記第 2 位相から前記第 1 位相に変更する
こと、
を特徴とする請求項 1 乃至請求項 6 の何れか一に記載のスレッド処理方法。

【請求項 8】

前記依存関係は、前記共有メモリまたは前記第 1 プロセッサのメモリに登録されること
、
を特徴とする請求項 1 乃至請求項 7 の何れか一に記載のスレッド処理方法。

【請求項 9】

第 1 プロセッサと、
前記第 1 プロセッサと第 2 プロセッサとがアクセス可能な共有メモリまたは共有キャッ
シュメモリと、
前記第 1 プロセッサまたは前記共有メモリに供給される動作クロックを制御するクロッ

ク制御ユニットと、

を含み、

前記クロック制御ユニットは、

前記第 1 プロセッサで実行される第 1 スレッドの前記共有メモリまたは前記共有キャッシュメモリへのアクセス頻度と第 1 閾値との比較結果と前記第 1 スレッドと前記第 2 プロセッサが実行する第 2 スレッドとの依存関係とに基づいて、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、

を特徴とするスレッド処理システム。

【請求項 10】

前記クロック制御ユニットは、前記比較結果が前記アクセス頻度が前記第 1 閾値よりも大きいことを示すとともに前記依存関係があるときに、前記第 1 動作クロックの位相を変更すること、

を特徴とする請求項 9 に記載のスレッド処理システム。

【請求項 11】

前記クロック制御ユニットは、前記第 1 プロセッサが前記第 1 スレッドを実行するときの第 1 負荷と第 2 閾値との比較結果に基づいて、前記第 1 プロセッサの第 1 動作クロックの位相を変更すること、

を特徴とする請求項 9 または請求項 10 に記載のスレッド処理システム。

【請求項 12】

前記クロック制御ユニットは、前記第 1 動作クロックの位相を前記第 2 プロセッサの第 2 動作クロックと逆相になるように変更すること、

を特徴とする請求項 9 乃至請求項 11 の何れか一に記載のスレッド処理システム。

【請求項 13】

前記クロック制御ユニットは、前記第 1 動作クロックの位相を変更するとき、前記共有メモリまたは共有キャッシュメモリに供給される動作クロックの周波数を変更すること、

を特徴とする請求項 9 乃至請求項 11 の何れか一に記載のスレッド処理システム。

【請求項 14】

前記クロック制御ユニットは、前記動作クロックの周波数を 2 倍に変更すること、

を特徴とする請求項 13 に記載のスレッド処理システム。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、スレッドを処理するスレッド処理方法、およびスレッド処理システムに関する。

【背景技術】

【0002】

マルチコアプロセッサシステムでは、複数の CPU が複数のスレッドを動作させる。メモリやバスなどの複数の CPU で共有する共有資源では、複数の CPU から同時にアクセスを受け付ける場合がある。共有資源では、たとえば、同時に複数のアクセスに関する処理を行うことができないため、各アクセスの優先度を判断し、アクセスに関する処理を順に行っている。

【0003】

また、フォールトトレラントシステムにおいて、故障モードになると、すべての CPU の動作を同一にする技術が知られている。すなわち、複数の CPU から共有資源へのアクセスが競合するが、共有資源の同一のデータにアクセスする。そこで、マスタ CPU には

共有資源へアクセスさせ、スレーブCPUには共有資源へアクセスさせずに、マスタCPUのアクセス結果をスレーブCPUに返すために、マスタCPUに供給されるクロックの位相と逆位相のクロックをスレーブCPUに供給させる。そして、スレーブCPUから共有資源へのアクセスのタイミングをマスタCPUから共有資源へのアクセスのタイミングよりも遅らせる技術が知られている（たとえば、特許文献1参照。）。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特表2008-518311号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、従来技術では、故障モードでない場合に、共有資源へ同時に2つのアクセスが発生した場合、いずれか一方のアクセスに関する処理は待機させられることになるという問題点がある。

【0006】

本発明は、上述した従来技術による問題点を解消するため、複数のCPUから共有資源へのアクセスが衝突することを減少させることができるスレッド処理方法、およびスレッド処理システムを提供することを目的とする。

【課題を解決するための手段】

【0007】

上述した課題を解決し、目的を達成するため、本発明の一側面によれば、第1スレッドの実行命令を第1プロセッサに供給し、前記第1スレッドと第2プロセッサで実行される第2スレッドとの依存関係を判断し、前記第1スレッドの共有メモリまたは共有キャッシュメモリへのアクセス頻度を第1閾値と比較し、前記アクセス頻度が第1閾値よりも大きいとき、かつ前記依存関係がないと判断されるときは、前記第1プロセッサの第1動作クロックの位相を変更するスレッド処理方法、およびスレッド処理システムが提案される。

【発明の効果】

【0008】

本発明の一側面によれば、共有資源へのアクセスが衝突することを減少させることができるという効果を奏する。

【図面の簡単な説明】

【0009】

【図1】図1は、本発明の一実施例を示す説明図である。

【図2】図2は、マルチコアプロセッサシステムの一例を示すブロック図である。

【図3】図3は、クロック生成回路204の一例を示している。

【図4】図4は、アプリケーションテーブルの一例を示す説明図である。

【図5】図5は、割当テーブルの一例を示す説明図である。

【図6】図6は、OS231-iの機能例を示すブロック図である。

【図7】図7は、例1を示す説明図である。

【図8】図8は、例2を示す説明図である。

【図9】図9は、例3を示す説明図である。

【図10】図10は、例4を示す説明図である。

【図11】図11は、OS231-iによるスレッド処理手順の一例を示すフローチャート（その1）である。

【図12】図12は、OS231-iによるスレッド処理手順の一例を示すフローチャート（その2）である。

【図13】図13は、OS231-1によるスレッド処理手順の一例を示すフローチャート（その1）である。

【図14】図14は、OS231-1によるスレッド処理手順の一例を示すフローチャート（その2）である。

ト（その２）である。

【図１５】図１５は、クロック生成回路２０４による変更処理手順の一例を示すフローチャートである。

【発明を実施するための形態】

【００１０】

以下に添付図面を参照して、本発明にかかるスレッド処理方法、およびスレッド処理システムの実施の形態を詳細に説明する。本実施の形態では、スレッド処理システムの一例として、マルチコアプロセッサシステムを挙げる。ここで、マルチコアプロセッサシステムにおいて、マルチコアプロセッサとは、コアが複数搭載されたプロセッサである。コアが複数搭載されていれば、複数のコアが搭載された単一のプロセッサでもよく、シングルコアのプロセッサが並列されているプロセッサ群でもよい。なお、本実施の形態では、説明を単純化するため、シングルコアのプロセッサが並列されているプロセッサ群を例に挙げて説明する。

【００１１】

図１は、本発明の一実施例を示す説明図である。第１のＯＳが、（１）第１のスレッドを第１のＣＰＵへディスパッチする。第１のＯＳが、（２）第１のスレッドの共有資源へのアクセス頻度を第１閾値と比較する。ここで、共有資源には、たとえば、共有メモリ、共有キャッシュメモリやバスなどが挙げられる。第１のＯＳが、（３）第１のスレッドと第２のＣＰＵで実行中の第２のスレッドとの依存関係を判断する。ここで、スレッドは、周知のようにアプリケーションプログラム内での処理の実行単位である。依存関係があるとは、たとえば、第１のスレッドと第２のスレッドとでデータを共有している場合が挙げられる。たとえば、ＣＰＵ間でキャッシュコヒーレントを取るか否かをアプリケーションの設計時に設計者があらかじめ解析しておくこととする。

【００１２】

第１のＯＳが、（４）アクセス頻度が第１閾値よりも大きいと判断し、かつ依存関係がない（×印）と判断した場合、第１のＣＰＵへ供給されるクロックの位相と第２のＣＰＵへ供給されるクロックの位相とが逆相となるようにクロック生成回路の設定を変更する。第１のＯＳが、（４）共有資源へ供給されるクロックの周波数を第１のＣＰＵへ供給されるクロックの周波数の２倍になるようにクロック生成回路の設定を変更する。

【００１３】

たとえば、第１のＣＰＵと第２のＣＰＵと共有資源とがクロックの立ち上がりエッジで処理が行われるとすると、第１のＣＰＵと第２のＣＰＵとの共有資源へのアクセスが衝突しない。

【００１４】

（マルチコアプロセッサシステム）

図２は、マルチコアプロセッサシステムの一例を示すブロック図である。マルチコアプロセッサシステム２００は、ＣＰＵ２０１－１～２０１－Ｎ（ここでは、 $N=4$ を例に挙げている。）と、スヌープ回路２０２と、２次キャッシュ２０３と、クロック生成回路２０４と、を有している。さらに、マルチコアプロセッサシステム２００は、ディスプレイ２０５と、キーボード２０６と、Ｉ／Ｆ２１２（Interface）と、メモリコントローラ２０７と、共有メモリ２０８と、を有している。

【００１５】

２次キャッシュ２０３と、クロック生成回路２０４と、ディスプレイ２０５と、キーボード２０６と、Ｉ／Ｆ２１２と、メモリコントローラ２０７とは、バス２１３を介して接続されている。ＣＰＵ２０１－１～２０１－４は、２次キャッシュ２０３を介して各部に接続されている。共有メモリ２０８は、メモリコントローラ２０７を介して各部に接続されている。

【００１６】

ここで、ＣＰＵ２０１－１～２０１－４は、それぞれレジスタとコアと１次キャッシュ２２１－１～２２１－４を有している。コアは、演算機能を有している。各ＣＰＵ内のレ

ジスタは、P C (P r o g r a m C o u n t e r) やリセットレジスタを有している。

【 0 0 1 7 】

C P U 2 0 1 - 1 はマスタC P Uであり、マルチコアプロセッサシステム 2 0 0 の全体の制御を司り、O S 2 3 1 - 1 を実行する。O S 2 3 1 - 1 はマスタO Sであり、C P U 2 0 1 - 1 に割り当てられたスレッドを実行する。O S 2 3 1 - 1 はスケジューラを有し、スケジューラは起動指示を受け付けたアプリケーションをマルチコアプロセッサのうちのいずれのC P U 2 0 1 に割り当てるかを制御する機能を有している。スケジューラはC P U 2 0 1 - 1 に割り当てられたアプリケーションの実行順序を制御する機能を有する。

【 0 0 1 8 】

C P U 2 0 1 - 2 ~ 2 0 1 - 4 はスレーブC P Uであり、それぞれO S 2 3 1 - 2 ~ 2 3 1 - 4 を実行する。O S 2 3 1 - 2 ~ 2 3 1 - 4 はスレーブO Sであり、それぞれC P U 2 0 1 - 2 ~ 2 0 1 - 4 に割り当てられたスレッドを実行する。O S 2 3 1 - 2 ~ 2 3 1 - 4 はそれぞれスケジューラを有し、各スケジューラはそれぞれC P U 2 0 1 - 2 ~ 2 0 1 - 4 に割り当てられたアプリケーションの実行順序を制御する機能を有する。

【 0 0 1 9 】

ここで、O S 2 3 1 - 1 ~ 2 3 1 - 4 はそれぞれランキュー 2 4 1 - 1 ~ 2 4 1 - 4 を有し、ランキュー 2 4 1 - 1 ~ 2 4 1 - 4 にはC P U 2 0 1 - 1 ~ 2 0 1 - 4 のそれぞれに割り当てられたアプリケーションのコンテキスト情報のポインタが積まれる。コンテキスト情報とは、たとえば、ロードされたアプリケーションの実行状態や該アプリケーション内の変数などが含まれる情報である。O S 2 3 1 - 1 ~ 2 3 1 - 4 はそれぞれランキュー 2 4 1 - 1 ~ 2 4 1 - 4 内のコンテキスト情報のポインタを取得し、アプリケーションのコンテキスト情報にアクセスすることで、アプリケーションを直ぐに実行することができる。

【 0 0 2 0 】

1 次キャッシュ 2 2 1 - 1 ~ 2 2 1 - 4 は、それぞれキャッシュメモリとキャッシュコントローラとを有している。1 次キャッシュ 2 2 1 - 1 ~ 2 2 1 - 4 はそれぞれO S 2 3 1 - 1 ~ 2 3 1 - 4 が実行するスレッドから共有メモリ 2 0 8 への書込処理を一時的に記憶する。また、1 次キャッシュ 2 2 1 - 1 ~ 2 2 1 - 4 は、共有メモリ 2 0 8 から読み出されたデータを一時的に記憶する。

【 0 0 2 1 】

スヌープ回路 2 0 2 は、1 次キャッシュ 2 2 1 - 1 ~ 2 2 1 - 4 で共有するデータがいずれかの1 次キャッシュ 2 2 1 で更新された場合、該更新を検出し、他の1 次キャッシュ 2 2 1 も更新する。

【 0 0 2 2 】

2 次キャッシュ 2 0 3 は、キャッシュメモリとキャッシュコントローラとを有している。2 次キャッシュ 2 0 3 では、1 次キャッシュ 2 2 1 - 1 ~ 2 2 1 - 4 から追い出されたデータを記憶する。2 次キャッシュ 2 0 3 では、O S 2 3 1 - 1 ~ O S 2 3 1 - 4 で共有するデータを記憶する。2 次キャッシュ 2 0 3 は、1 次キャッシュ 2 2 1 - 1 ~ 2 2 1 - 4 よりも、記憶容量が大きくかつC P U 2 0 1 - 1 ~ 2 0 1 - 4 からのアクセス速度が遅い。2 次キャッシュ 2 0 3 は共有メモリ 2 0 8 より、記憶容量が小さくかつC P U 2 0 1 - 1 ~ 2 0 1 - 4 からのアクセス速度が速い。

【 0 0 2 3 】

ディスプレイ 2 0 5 は、カーソル、アイコンあるいはツールボックスをはじめ、文書、画像、機能情報などのデータを表示する。ディスプレイ 2 0 5 は、たとえば、T F T 液晶ディスプレイなどを採用することができる。キーボード 2 0 6 は、数字、各種指示などの入力のためのキーを有し、データの入力を行う。また、キーボード 2 0 6 は、タッチパネル式の入力パッドやテンキーなどであってもよい。

【 0 0 2 4 】

I / F 2 1 2 は、通信回線を通じてL A N (L o c a l A r e a N e t w o r k) 、W A N (W i d e A r e a N e t w o r k) 、インターネットなどのネットワーク

に接続され、ネットワークを介して他の装置に接続される。そして、I/F 212は、ネットワークと内部のインターフェースを司り、外部装置からのデータの入出力を制御する。I/F 212には、たとえばモデムやLANアダプタなどを採用することができる。

【0025】

メモリコントローラ207は、共有メモリ208へのアクセスを制御する。共有メモリ208は、たとえば、RAM 209 (Random Access Memory) と、ROM 210 (Read Only Memory) と、フラッシュROM 211を有している。ROM 210は、ブートプログラムなどのプログラムを記憶している。RAM 209は、各CPUのワークエリアとして使用される。フラッシュROM 211は、OS 231-1~231-4などのシステムソフトウェアやアプリケーションのプログラムを記憶している。RAM 209はフラッシュROM 211よりも各CPUからのアクセス速度が速い。各OSがアプリケーションのプログラムをフラッシュROM 211からRAM 209へロードすることにより、該アプリケーションのコンテキスト情報がRAM 209内に展開される。

【0026】

クロック生成回路204は、各部にクロックを供給する。クロックCLKが発振回路や外部から入力される基準クロックである。クロックCLK1~CLK4がそれぞれCPU 201-1~201-4へ供給されるクロックである。クロックCLK-2CMが2次キャッシュ203へ供給されるクロックである。また、メモリコントローラ207、共有メモリ208、バス213、I/F 212、ディスプレイ205やキーボード206などにもクロック生成回路204からクロックが共有されているが、図では配線を省略している。

【0027】

たとえば、制御信号CPUCTRL1と、制御信号CPUCTRL2と、制御信号CPUCTRL3と、制御信号CPUCTRL4とに応じて、各CPUへ供給されるクロックの位相と2次キャッシュ203へ供給されるクロックの周波数が決定される。

【0028】

図3は、クロック生成回路204の一例を示している。クロック生成回路204は、制御決定回路301と、選択回路302と、インバータ303と、120°遅延回路304と、90°遅延回路305と、選択回路306と、インバータ307と、120°遅延回路308と、を有している。クロック生成回路204は、90°遅延回路309と、選択回路310と、インバータ311と、90°遅延回路312と、通倍回路313と、を有している。

【0029】

インバータ303は、クロックCLKを反転して出力することにより、クロックCLKと逆相のクロックを出力する。インバータ307は、クロックCLKを反転して出力することにより、クロックCLKと逆相のクロックを出力する。インバータ311は、クロックCLKを反転して出力することにより、クロックCLKと逆相のクロックを出力する。

【0030】

120°遅延回路304は、クロックCLKの位相を120°ずらす。120°遅延回路308は、120°遅延回路304から出力されるクロックの位相を120°ずらす。90°遅延回路305は、クロックCLKの位相を90°ずらす。90°遅延回路309は、90°遅延回路305から出力されるクロックの位相を90°ずらす。90°遅延回路312は、90°遅延回路309から出力されるクロックの位相を90°ずらす。

【0031】

選択回路302は、制御信号CTRL2の値に応じてCPU 201-2に供給するクロックを選択する。選択回路302では、制御信号CTRL2が「1」の場合、クロックCLKが選択される。選択回路302では、制御信号CTRL2が「2」の場合、クロックCLKの位相と逆位相のクロックが選択される。選択回路302では、制御信号CTRL2が「3」の場合、クロックCLKの位相が120°ずらされたクロックが選択される。

選択回路302では、制御信号CTRL2が「4」の場合、クロックCLKの位相が90°ずらされたクロックが選択される。

【0032】

選択回路306は、制御信号CTRL3の値に応じてCPU201-3に供給するクロックを選択する。選択回路306では、制御信号CTRL3が「1」の場合、クロックCLKが選択される。選択回路306では、制御信号CTRL3が「2」の場合、クロックCLKの位相と逆位相のクロックが選択される。選択回路306では、制御信号CTRL3が「3」の場合、クロックCLKの位相が120°ずらされたクロックが選択される。選択回路306では、制御信号CTRL3が「4」の場合、クロックCLKの位相が240°ずらされたクロックが選択される。選択回路306では、制御信号CTRL3が「5」の場合、クロックCLKの位相が180°ずらされたクロックが選択される。

【0033】

選択回路310は、制御信号CTRL4の値に応じてCPU201-4に供給するクロックを選択する。選択回路310では、制御信号CTRL4が「1」の場合、クロックCLKが選択される。選択回路310では、制御信号CTRL4が「2」の場合、クロックCLKの位相と逆位相のクロックが選択される。選択回路310では、制御信号CTRL4が「3」の場合、クロックCLKの位相が120°ずらされたクロックが選択される。選択回路310では、制御信号CTRL4が「4」の場合、クロックCLKの位相が240°ずらされたクロックが選択される。選択回路310では、制御信号CTRL3が「5」の場合、クロックCLKの位相が270°ずらされたクロックが選択される。

【0034】

通倍回路313は、RATEに応じてクロックCLKを通倍にする。通倍回路313では、制御信号RATEの値が「1」の場合、クロックCLKがそのまま出力される。通倍回路313では、制御信号RATEの値が「2」の場合、クロックCLKの周波数の2倍の周波数となるクロックが出力される。通倍回路313では、制御信号RATEの値が「3」の場合、クロックCLKの周波数の3倍の周波数となるクロックが出力される。通倍回路313では、制御信号RATEの値が「4」の場合、クロックCLKの周波数の4倍の周波数となるクロックが出力される。制御信号RATEの最大値は「4」である。

【0035】

制御決定回路301は、制御信号CPUCTRL1から制御信号CPUCTRL4に応じて、各CPUへ供給されるクロックの位相と2次キャッシュ203へ供給されるクロックの周波数を決定する。たとえば、制御信号CPUCTRL1から制御信号CPUCTRL4のそれぞれのビット幅が12ビット（3[ビット]×4[制御信号]）である。制御信号CPUCTRL1から制御信号CPUCTRL4はそれぞれ上位3ビットごとに順に制御信号CTRL2の値、制御信号CTRL3の値、制御信号CTRL4の値、制御信号RATEの値を示していることとする。

【0036】

たとえば、OS231-1が、通常時には制御信号CPUCTRL1を「0000（10進表記）」に設定し、各CPUへ供給するクロックの位相や2次キャッシュ203へ供給するクロックの周波数を変更する際に変化させる。たとえば、制御信号CPUCTRL1の値が「1234（10進表記）」に変化させられると、制御決定回路301により、制御信号CTRL1の値が「1」となり、制御信号CTRL2の値が「2」となることとする。さらに、制御決定回路301により、制御信号CTRL3の値が「3」となり、制御信号RATEの値が「4」となることとする。また、制御決定回路301は、制御信号CPUCTRL1～CPUCTRL4の「0000」から変更値への変化は検出するが、変更値から「0000」への変化は検出しないこととする。

【0037】

（アプリケーションテーブル）

図4は、アプリケーションテーブルの一例を示す説明図である。アプリケーションテーブル400は、アプリID、スレッドID、依存スレッドID、アクセス頻度、負荷量の

フィールドを有している。

【0038】

アプリIDのフィールドには、各アプリケーションを識別するための識別情報が登録される。スレッドIDのフィールドには、各スレッドを識別するための識別情報が登録される。依存スレッドIDのフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドと依存関係のあるスレッドの識別情報が登録される。

【0039】

アクセス頻度のフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドから2次キャッシュ203への単位時間あたりのアクセス回数が登録される。負荷量のフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドの単位時間当たりの実行時間の比率が登録される。アクセス頻度や負荷量は、各スレッドのコンパイル時の解析およびシミュレーション結果に基づいて登録される。

【0040】

フィールドに情報が設定されることで、アプリ情報（たとえば、アプリ情報401-1～401-4）がレコードとして記憶されている。アプリケーションテーブル400は、RAM209、ROM210、フラッシュROM211などの記憶装置に記憶されている。さらに、2次キャッシュ203や1次キャッシュ221などに複製されたアプリケーションテーブル400が記憶されていてもよい。

【0041】

図5は、割当テーブルの一例を示す説明図である。割当テーブル500は、アプリID、スレッドID、割当先CPUIDのフィールドを有している。アプリIDのフィールドには、各アプリケーションを識別するための識別情報が登録される。スレッドIDのフィールドには、各スレッドを識別するための識別情報が登録される。割当先CPUIDのフィールドには、同一レコードのスレッドIDのフィールドに識別情報が登録されたスレッドの割当先CPUの識別情報が登録される。

【0042】

各フィールドに情報が設定されることで、割当情報（たとえば、割当情報501-1～501-5）がレコードとして記憶されている。割当テーブル500は、RAM209、ROM210、フラッシュROM211などの記憶装置に記憶されている。さらに、2次キャッシュ203や1次キャッシュ221などに複製されたアプリケーションテーブル400が記憶されていてもよい。

【0043】

(OS231-i (i=1～4)の機能例)

図6は、OS231-iの機能例を示すブロック図である。OS231-iは、受付部601-iと、判断部602-iと、比較部603-iと、グループ化設定部604-iと、変更部605-iと、実行部606-iと、を有している。受付部601-iから変更部605-iを有するOS231-iがフラッシュROM211などのストレージに記憶されている。CPU201-iが該記憶装置にアクセスして該OS231-iを読み出し、該OS231-i内にコーディングされている処理を実行することにより、該受付部601-iから実行部606-iの処理が実行される。ここでは、i=1を例に挙げて説明する。

【0044】

受付部601-1は、イベントを受け付ける。マスタOSが受け付けるイベントとしては、スレッドの生成、スレッドのディスパッチ、スレッドの切り替えやスレッドの実行終了が挙げられる。スレーブOSが受け付けるイベントとしては、スレッドのディスパッチ、スレッドの切り替えやスレッドの実行終了が挙げられる。ここでは、イベントがスレッドの切り替えの場合を例に挙げて説明する。

【0045】

受付部601-1は、スレッドの切り替えを受け付け、判断部602-1は、受付部6

01-1により切り替えが受け付けられたスレッド（第1のスレッド）と他のCPUで実行中のそれぞれのスレッドとに依存関係があるか否かを判断する。

【0046】

たとえば、判断部602-1は、アプリケーションテーブル400から、第1のスレッドのIDをスレッドIDのフィールドに有するアプリ情報を検出する。たとえば、判断部602-1は、検出したアプリ情報の依存スレッドIDのフィールドに登録されているスレッドのIDを特定する。これにより、第1のスレッドと依存関係のあるスレッドのIDが特定される。

【0047】

たとえば、判断部602-1は、プロセッサ間通信機能により他のOS231から、実行中のスレッドのIDを取得する。たとえば、判断部602-1は、特定したスレッドのIDと取得したスレッドのIDとを比較する。これにより、第1のスレッドと他のCPUで実行中のそれぞれのスレッドとに依存関係があるか否かが判断される。なお、判断結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0048】

比較部603-1は、第1のスレッドがいずれのCPU201で実行中のスレッドとも依存関係がないと判断部602-1により判断された場合、第1のスレッドの2次キャッシュ203へのアクセス頻度を第1閾値と比較する。

【0049】

たとえば、比較部603-1は、アプリケーションテーブル400から、第1のスレッドのIDをスレッドIDのフィールドに有するアプリ情報を検出する。たとえば、比較部603-1は、検出したアプリ情報のアクセス頻度のフィールドに登録されたアクセス頻度を特定する。たとえば、比較部603-1は、特定したアクセス頻度が第1閾値以上であるか否かを判断する。これにより、第1のスレッドの2次キャッシュ203へのアクセス頻度が第1閾値以上であるか否かが判断される。

【0050】

比較部603-1は、第1のスレッドの2次キャッシュ203へのアクセス頻度が第1閾値以上であると判断した場合、第1のスレッドの負荷量を第2閾値と比較する。たとえば、比較部603-1は、検出したアプリ情報の負荷量のフィールドに登録された第1のスレッドの負荷量が、第2閾値以上であるか否かを判断する。

【0051】

比較部603-1は、第1のスレッドの負荷量が第2閾値以上であると判断した場合、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大であるか否かを判断する。たとえば、比較部603-1は、制御信号RATEの値をクロック生成回路204から取得し、制御信号RATEの値が最大値であるか否かを判断する。上述最大値は4である。

【0052】

判断部602-1は、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大でないと比較部603-1により判断された場合、CPU201-1を除く残余のCPUで実行中のスレッド間に依存関係があるか否かを判断する。たとえば、判断部602-1は、CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドとに依存関係があるか否かを判断する。たとえば、判断部602-1は、CPU201-2で実行中のスレッドとCPU201-4で実行中のスレッドとに依存関係があるか否かを判断する。たとえば、判断部602-1は、CPU201-3で実行中のスレッドとCPU201-4で実行中のスレッドとに依存関係があるか否かを判断する。具体的な判断方法については、CPU201-1で実行中のスレッドと、CPU201-2～201-4のそれぞれで実行中のスレッドとの依存関係の有無を判断した処理と同一処理であるため、詳細な説明は省略する。なお、判断結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0053】

(例 1)

図 7 は、例 1 を示す説明図である。図 7 では、丸印の中に符号が付されているものが、各符号に即した CPU 201 で実行中のスレッドを示している。CPU 201-1 で切り替えが発生した第 1 のスレッドと CPU 201-2 ~ 201-4 のそれぞれで実行中のスレッドには依存関係がない。そのため、図 7 では、CPU 201-1 で切り替えが発生した第 1 のスレッドと CPU 201-2 ~ 201-4 のそれぞれで実行中のスレッドとを結ぶ矢印間に×印が付されている。

【0054】

図 7 では、CPU 201-2 で実行中のスレッドと CPU 201-3 で実行中のスレッドは依存関係がなく、CPU 201-2 で実行中のスレッドと CPU 201-4 で実行中のスレッドは依存関係がない。CPU 201-3 で実行中のスレッドと CPU 201-4 で実行中のスレッドは依存関係がない。

【0055】

グループ化設定部 604-1 は、依存関係に応じて CPU をグループ化する。ここで、1 グループは、1 CPU 以上であり、直接的または間接的に依存関係があるスレッドを実行中の CPU を 1 グループとする。たとえば、グループ化設定部 604-1 が、各 CPU を異なるグループに設定する。CPU 201-1 はグループ G1 に属し、CPU 201-2 はグループ G2 に属し、CPU 201-3 はグループ G3 に属し、CPU 201-4 はグループ G4 に属する。CPU グループ数 GC は 4 である。なお、グループ化の設定結果は、RAM 209、フラッシュ ROM 211 などの記憶装置に記憶される。

【0056】

変更部 605-1 は、グループ化の設定結果に応じて各 CPU へ供給するクロックの位相を変更し、CPU グループ数 GC に応じて 2 次キャッシュ 203 へ供給するクロックの周波数を変更する。たとえば、変更部 605-1 は、制御信号 CPUCTRL1 の値を 0 から「4554」に変更する。クロック生成回路 204 は、制御信号 CPUCTRL1 の変化を検出する。クロック生成回路 204 は、制御信号 CPUCTRL1 の値に基づいて、制御信号 CTRL2 の値を「4」にし、制御信号 CTRL3 の値を「5」にし、制御信号 CTRL4 の値を「5」にし、制御信号 RATE の値を「4」にする。

【0057】

これにより、CPU 201-1 へ供給されるクロックの位相よりも 90° ($=360^\circ / G$) ずらされた位相のクロックが CPU 201-2 へ供給される。CPU 201-2 へ供給されるクロックの位相よりも 90° ずらされた位相のクロックが CPU 201-3 へ供給される。CPU 201-3 へ供給されるクロックの位相よりも 90° ずらされた位相のクロックが CPU 201-4 へ供給される。クロック CLK-2CM の周波数がクロック CLK の周波数の 4 倍となる。そして、実行部 606-1 は、変更部 605-1 による変更が終了後、第 1 のスレッドを実行する。すなわち、グループ間で供給されるクロックが非同期となり、クロック CLK-2CM の周波数がいずれのグループとも同期する周波数となる。

【0058】

(例 2)

図 8 は、例 2 を示す説明図である。CPU 201-1 で切り替えが発生した第 1 のスレッドと CPU 201-2 ~ 201-4 のそれぞれで実行中のスレッドには依存関係がない。CPU 201-2 で実行中のスレッドと CPU 201-3 で実行中のスレッドは依存関係がない。CPU 201-2 で実行中のスレッドと CPU 201-4 で実行中のスレッドは依存関係がない。CPU 201-3 で実行中のスレッドと CPU 201-4 で実行中のスレッドは直接的に依存関係がある。×印の矢印間は、依存関係がないことを示し、○印の矢印間は、依存関係があることを示している。

【0059】

グループ化設定部 604-1 は、CPU 201-1 をグループ G1 にし、CPU 201-2 をグループ G2 にし、CPU 201-3 と CPU 201-4 をグループ G3 にする。

CPUグループ数GCは3である。なお、グループ化の設定結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0060】

変更部605-1は、CPUグループ数GCに応じて各CPUへ供給するクロックの位相を変更し、グループ数に応じて2次キャッシュ203へ供給するクロックの周波数を変更する。たとえば、変更部605-1は、制御信号CPUCTRL1の値を0から「3443」に変更する。クロック生成回路204は、制御信号CPUCTRL1の変化を検出する。クロック生成回路204は、制御信号CPUCTRL1の値に基づいて、制御信号CTRL2の値を「3」にし、制御信号CTRL3の値を「4」にし、制御信号CTRL4の値を「4」にし、制御信号RATEの値を「3」にする。

【0061】

これにより、CPU201-1へ供給されるクロックの位相よりも120°（=360°/CPUグループ数GC=360°/3）ずらされた位相のクロックがCPU201-2へ供給される。CPU201-2へ供給されるクロックの位相よりも120°ずらされた位相のクロックがCPU201-3へ供給される。CPU201-2へ供給されるクロックの位相よりも120°ずらされた位相のクロックがCPU201-4へ供給される。クロックCLK-2CMの周波数がクロックCLKの周波数の3倍となる。そして、実行部606-1は、変更部605-1による変更が終了後、第1のスレッドを実行する。

【0062】

（例3）

図9は、例3を示す説明図である。CPU201-1で切り替えが発生した第1のスレッドとCPU201-2～201-4のそれぞれで実行中のスレッドには依存関係がない。図9では、CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドには依存関係がある。CPU201-3で実行中のスレッドとCPU201-4で実行中のスレッドには依存関係がある。CPU201-2で実行中のスレッドとCPU201-4で実行中のスレッドには直接的に依存関係がないが、CPU201-3を介して間接的に依存関係がある。

【0063】

グループ化設定部604-1は、CPU201-1をグループG1にし、CPU201-2とCPU201-3とCPU201-4をグループG2にする。CPUグループ数GCは2である。

【0064】

変更部605-1は、CPUグループ数GCに応じて各CPUへ供給するクロックの位相を変更し、グループ数に応じて2次キャッシュ203へ供給するクロックの周波数を変更する。たとえば、変更部605-1は、制御信号CPUCTRL1の値を0から「2222」に変更する。クロック生成回路204は、制御信号CPUCTRL1の変化を検出する。クロック生成回路204は、制御信号CPUCTRL1の値に基づいて、制御信号CTRL2の値を「2」にし、制御信号CTRL3の値を「2」にし、制御信号CTRL4の値を「2」にし、制御信号RATEの値を「2」にする。

【0065】

これにより、CPU201-1へ供給されるクロックの位相よりも180°（=360°/G）ずらされた位相（逆相）のクロックがCPU201-2へ供給される。CPU201-2へ供給されるクロックの位相と逆相のクロックがCPU201-3へ供給される。CPU201-2へ供給されるクロックの位相と逆相のクロックがCPU201-4へ供給される。クロックCLK-2CMの周波数がクロックCLKの周波数の2倍となる。そして、実行部606-1は、変更部605-1による変更が終了後、第1のスレッドを実行する。

【0066】

（例4）

また、判断部602-1は、受付部601-1により切り替えが受け付けられた第1の

スレッドと他のCPUで実行中のそれぞれのスレッドとに依存関係があると判断した場合、すべてのCPUで実行中のスレッド間に依存関係があるか否かを判断する。

【0067】

図10は、例4を示す説明図である。CPU201-1で切り替えが発生した第1のスレッドとCPU201-2で実行中のスレッドは直接的に依存関係がある。CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドは直接的に依存関係がある。CPU201-3で実行中のスレッドとCPU201-4で実行中のスレッドは直接的に依存関係がある。

【0068】

CPU201-1で切り替えが発生した第1のスレッドとCPU201-3で実行中のスレッドは、直接的に依存関係がないが、CPU201-2で実行中のスレッドを介して間接的に依存関係がある。CPU201-1で切り替えが発生した第1のスレッドとCPU201-4で実行中のスレッドは、直接的に依存関係がないが、CPU201-2で実行中のスレッドとCPU201-3で実行中のスレッドを介して間接的に依存関係がある。

【0069】

たとえば、グループ化設定部604-1は、CPU201-1～201-4をグループG1にする。CPUグループ数GCは1である。なお、グループ化の設定結果は、RAM209、フラッシュROM211などの記憶装置に記憶される。

【0070】

変更部605-1は、CPUグループ数GCに応じて各CPUへ供給するクロックの位相を変更し、CPUグループ数GCに応じて2次キャッシュ203へ供給するクロックの周波数を変更する。たとえば、変更部605-1は、制御信号CPUCTRL1の値を0から「1111」に変更する。クロック生成回路204は、制御信号CPUCTRL1の変化を検出する。クロック生成回路204は、制御信号CPUCTRL1の値に基づいて、制御信号CTRL2の値を「1」にし、制御信号CTRL3の値を「1」にし、制御信号CTRL4の値を「1」にし、制御信号RATEの値を「1」にする。

【0071】

これにより、クロックCLKと同相のクロックが各CPUへ供給される。クロックCLK-2CMの周波数がクロックCLKの周波数と同一となる。そして、実行部606-1は、変更部605-1による変更が終了後、第1のスレッドを実行する。

【0072】

(OS231-iによるスレッド処理手順)

図11および図12は、OS231-iによるスレッド処理手順の一例を示すフローチャートである。OS231-iは、受付部601-iにより、イベントの発生を受け付けたか否かを判断する(ステップS1101)。OS231-iは、イベントの発生を受け付けていないと判断した場合(ステップS1101:No)、ステップS1101へ戻る。OS231-iは、ディスパッチを受け付けると(ステップS1101:ディスパッチ)、ステップS1123へ移行する。

【0073】

OS231-iは、スレッドの実行終了を受け付けたと判断した場合(ステップS1101:実行終了)、実行待ちのスレッドがあるか否かを判断する(ステップS1102)。実行待ちのスレッドがあるか否かについては、ランキュー241-iにコンテキスト情報のポインタが登録されているか否かによって判断される。OS231-iは、実行待ちのスレッドがないと判断した場合(ステップS1102:No)、ステップS1116へ移行する。OS231-iは、実行待ちのスレッドがあると判断した場合(ステップS1102:Yes)、ステップS1103へ移行する。

【0074】

ステップS1101において、OS231-iは、スレッドのスイッチを受け付けたと判断した場合(ステップS1101:スイッチ)、ステップS1103へ移行する。ステ

ップS 1 1 0 1のスイッチ、またはステップS 1 1 0 2のY e sのつぎに、O S 2 3 1－iが、判断部6 0 2－iにより、つぎに実行するスレッド（第1のスレッド）と他のC P Uで実行中のスレッドとに依存関係があるか否かを判断する（ステップS 1 1 0 3）。つぎに実行するスレッドとは、スレッドのスイッチの場合にはスイッチ後のスレッドであり、スレッドの実行終了の場合にはランキュー2 4 1－iの先頭にコンテキスト情報のポイントが登録されているスレッドである。

【0 0 7 5】

O S 2 3 1－iは、第1のスレッドと他のC P Uで実行中のスレッドとに依存関係があると判断した場合（ステップS 1 1 0 3：Y e s）、他のC P Uで実行中のスレッド間の依存関係を確認する（ステップS 1 1 0 4）。O S 2 3 1－iは、すべてのC P Uで直接的または間接的に依存関係があるか否かを判断する（ステップS 1 1 0 5）。O S 2 3 1－iは、すべてのC P Uで依存関係があると判断した場合（ステップS 1 1 0 5：Y e s）、すべてのC P Uに供給するクロックの位相を同一位相に変更する（ステップS 1 1 0 6）。O S 2 3 1－iは、2次キャッシュ2 0 3に供給するクロックの周波数をデフォルトに変更し（ステップS 1 1 0 7）、ステップS 1 1 2 3へ移行する。

【0 0 7 6】

ステップS 1 1 0 5において、O S 2 3 1－iは、すべてのC P Uで依存関係があると判断しなかった場合（ステップS 1 1 0 5：N o）、ステップS 1 1 1 2へ移行する。ステップS 1 1 0 3において、O S 2 3 1－iは、第1のスレッドと他のC P Uで実行中のスレッドとに依存関係がないと判断した場合（ステップS 1 1 0 3：N o）、第1のスレッドのメモリアクセス頻度＞所定頻度であるか否かを判断する（ステップS 1 1 0 8）。所定頻度は第1閾値である。O S 2 3 1－iは、第1のスレッドのメモリアクセス頻度＞所定頻度でないと判断した場合（ステップS 1 1 0 8：N o）、ステップS 1 1 1 6へ移行する。O S 2 3 1－iは、第1のスレッドのメモリアクセス頻度＞所定頻度であると判断した場合（ステップS 1 1 0 8：Y e s）、第1のスレッドの負荷量＞所定負荷量であるか否かを判断する（ステップS 1 1 0 9）。所定負荷量は第2閾値である。

【0 0 7 7】

O S 2 3 1－iは、第1のスレッドの負荷量＞所定負荷量でないと判断した場合（ステップS 1 1 0 9：N o）、ステップS 1 1 1 6へ移行する。O S 2 3 1－iは、第1のスレッドの負荷量＞所定負荷量であると判断した場合（ステップS 1 1 0 9：Y e s）、2次キャッシュ2 0 3へ供給するクロックの周波数の通倍率が最大であるか否かを判断する（ステップS 1 1 1 0）。O S 2 3 1－iは、2次キャッシュ2 0 3へ供給するクロックの周波数の通倍率が最大であると判断した場合（ステップS 1 1 1 0：Y e s）、ステップS 1 1 2 3へ移行する。O S 2 3 1－iは、2次キャッシュ2 0 3へ供給するクロックの周波数の通倍率が最大でないと判断した場合（ステップS 1 1 1 0：N o）、判断部6 0 2－iにより、他のC P Uで実行中のスレッド間の依存関係を確認する（ステップS 1 1 1 1）。

【0 0 7 8】

O S 2 3 1－iは、直接的または間接的に依存関係のあるC P Uごとにグループ化し（ステップS 1 1 1 2）、C P Uグループ数G Cを決定する（ステップS 1 1 1 3）。O S 2 3 1－iは、グループ化したC P Uごとに該C P Uへ供給するクロックの位相を変更し（ステップS 1 1 1 4）、C P Uグループ数G Cに基づいて、2次キャッシュ2 0 3に供給するクロックの周波数を変更し（ステップS 1 1 1 5）、ステップS 1 1 2 3へ移行する。

【0 0 7 9】

ステップS 1 1 0 2のN o、ステップS 1 1 0 8のN o、またはステップS 1 1 0 9のN oのつぎに、O S 2 3 1－iは、比較部6 0 3－iにより、スイッチ前に実行していたスレッドまたは実行終了したスレッドのメモリアクセス頻度＞所定頻度であるかを判断する（ステップS 1 1 1 6）。O S 2 3 1－iは、スイッチ前に実行していたスレッドまたは実行終了したスレッドのメモリアクセス頻度＞所定頻度でないと判断した場合（ステッ

プ S 1 1 1 6 : N o) 、ステップ S 1 1 2 3 へ移行する。OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドのメモリアクセス頻度 > 所定頻度であると判断した場合 (ステップ S 1 1 1 6 : Y e s) 、ステップ S 1 1 1 7 へ移行する。

【 0 0 8 0 】

OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドの負荷量 > 所定負荷量であるか否かを判断する (ステップ S 1 1 1 7) 。OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドの負荷量 > 所定負荷量でないと判断した場合 (ステップ S 1 1 1 7 : N o) 、ステップ S 1 1 2 3 へ移行する。

【 0 0 8 1 】

OS 2 3 1 - i は、スイッチ前に実行していたスレッドまたは実行終了したスレッドの負荷量 > 所定負荷量であると判断した場合 (ステップ S 1 1 1 7 : Y e s) 、CPU グループ数 G C をデクリメントする (ステップ S 1 1 1 8) 。OS 2 3 1 - i は、他の CPU で実行中のスレッド間の依存関係を確認し (ステップ S 1 1 1 9) 、減らした CPU グループ数 G C と依存関係に沿って CPU 2 0 1 をグループ化する (ステップ S 1 1 2 0) 。たとえば、依存関係に沿って CPU 数が均等になるように CPU グループ数 G C 分のグループを作成することとしてもよい。

【 0 0 8 2 】

OS 2 3 1 - i は、グループ化した CPU ごとに該 CPU へ供給するクロックの位相を変更する (ステップ S 1 1 2 1) 。OS 2 3 1 - i は、CPU グループ数 G C に基づいて、2 次キャッシュ 2 0 3 に供給するクロックの周波数を変更し (ステップ S 1 1 2 2) 、ステップ S 1 1 2 3 へ移行する。OS 2 3 1 - i は、実行部 6 0 6 - i により、第 1 のスレッドを実行し (ステップ S 1 1 2 3) 、ステップ S 1 1 0 1 へ戻る。ステップ S 1 1 2 3 は、ステップ S 1 1 0 1 のディスパッチ、ステップ S 1 1 0 7 、ステップ S 1 1 1 0 の Y e s 、ステップ S 1 1 1 5 、ステップ S 1 1 1 6 の N o 、ステップ S 1 1 1 7 の N o 、またはステップ S 1 1 2 2 のつぎに実行される。

【 0 0 8 3 】

(OS 2 3 1 - 1 によるスレッド処理手順)

図 1 3 および図 1 4 は、OS 2 3 1 - 1 によるスレッド処理手順の一例を示すフローチャートである。OS 2 3 1 - 1 が、受付部 6 0 1 - i により、スレッドの生成を受け付けたか否かを判断する (ステップ S 1 3 0 1) 。OS 2 3 1 - 1 が、スレッドの生成を受け付けていないと判断した場合 (ステップ S 1 3 0 1 : N o) 、ステップ S 1 3 0 1 へ戻る。

【 0 0 8 4 】

OS 2 3 1 - 1 が、スレッドの生成を受け付けたと判断した場合 (ステップ S 1 3 0 1 : Y e s) 、生成されたスレッドが他の CPU で実行中のスレッドと依存関係があるか否かを判断する (ステップ S 1 3 0 2) 。OS 2 3 1 - 1 が、生成されたスレッドが他の CPU で実行中のスレッドと依存関係があると判断した場合 (ステップ S 1 3 0 2 : Y e s) 、他の CPU で実行中のスレッド間の依存関係を確認する (ステップ S 1 3 0 3) 。OS 2 3 1 - 1 が、すべての CPU で直接的または間接的に依存関係があるか否かを判断する (ステップ S 1 3 0 4) 。OS 2 3 1 - 1 が、すべての CPU で直接的または間接的に依存関係がないと判断した場合 (ステップ S 1 3 0 4 : N o) 、ステップ S 1 3 1 3 へ移行する。

【 0 0 8 5 】

OS 2 3 1 - 1 が、すべての CPU で直接的または間接的に依存関係があると判断した場合 (ステップ S 1 3 0 4 : Y e s) 、すべての CPU に供給するクロックの位相を同一位相に変更する (ステップ S 1 3 0 5) 。OS 2 3 1 - 1 が、2 次キャッシュ 2 0 3 に供給するクロックの周波数をデフォルトに変更し (ステップ S 1 3 0 6) 、負荷が最小の CPU に生成されたスレッドを割り当て (ステップ S 1 3 0 7) 、ステップ S 1 3 0 1 へ戻る。負荷が最小の CPU については、まず、OS 2 3 1 - 1 が、割当テーブル 5 0 0 を用いて各 CPU に割り当てられているスレッドを特定する。OS 2 3 1 - 1 が、アプリケー

ションテーブル400を参照し、特定したスレッドの負荷量を検索する。OS231-1が、CPUごとに割当済のスレッドの検索した負荷量の合計値を算出し、合計値が最小のCPUを負荷が最小のCPUとして特定する。

【0086】

OS231-1が、生成されたスレッドが他のCPUで実行中のスレッドと依存関係がないと判断した場合（ステップS1302：No）、生成されたスレッドのメモリアクセス頻度>所定頻度であるか否かを判断する（ステップS1308）。OS231-1が、生成されたスレッドのメモリアクセス頻度>所定頻度でないとして判断した場合（ステップS1308：No）、ステップS1311へ移行する。

【0087】

OS231-1が、生成されたスレッドのメモリアクセス頻度>所定頻度であると判断した場合（ステップS1308：Yes）、生成されたスレッドの負荷量>所定負荷量であるか否かを判断する（ステップS1309）。OS231-1が、生成されたスレッドの負荷量>所定負荷量でないとして判断した場合（ステップS1309：No）、ステップS1311へ移行する。

【0088】

OS231-1が、生成されたスレッドの負荷量>所定負荷量であると判断した場合（ステップS1309：Yes）、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大であるか否かを判断する（ステップS1310）。OS231-1が、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大であると判断した場合（ステップS1310：Yes）、ステップS1311へ移行する。ステップS1308のNo、ステップS1309のNo、またはステップS1310のYesのつぎに、OS231-1が、負荷が最小のCPUに生成されたスレッドを割り当て（ステップS1311）、ステップS1301へ戻る。

【0089】

ステップS1310において、OS231-1が、2次キャッシュ203へ供給するクロックの周波数の通倍率が最大でないとして判断した場合（ステップS1310：No）、他のCPUで実行中のスレッド間の依存関係を確認する（ステップS1312）。OS231-1が、直接的または間接的に依存関係のあるCPUごとにグループ化し（ステップS1313）、CPUグループ数GCとグループ化されたCPUに基づいて、各CPUへ供給するクロックの位相を変更する（ステップS1314）。OS231-1が、CPUグループ数GCに基づいて、2次キャッシュ203に供給するクロックの周波数を変更し（ステップS1315）、ステップS1301へ戻る。

【0090】

（クロック生成回路204による変更処理手順）

図15は、クロック生成回路204による変更処理手順の一例を示すフローチャートである。まず、クロック生成回路204が、各CPUからの制御命令を受け付けたか否かを判断する（ステップS1501）。制御命令とは、たとえば、制御信号CPUCTRL1からCPUCTRL4のいずれかの「0000」から変更値への変化である。

【0091】

クロック生成回路204が、各CPUからの制御命令を受け付けたと判断した場合（ステップS1501：Yes）、制御信号を変更し（ステップS1502）、ステップS1501へ戻る。ここで、制御信号とは、たとえば、制御信号CTRL1からCTRL4や制御信号RATEである。クロック生成回路204が、各CPUからの制御命令を受け付けていないとして判断した場合（ステップS1501：No）、ステップS1501へ戻る。

【0092】

以上説明したように、スレッド処理方法、およびスレッド処理システムによれば、第1のCPUで実行される第1のスレッドと第2のCPUで実行される第2のスレッドに依存関係がないかを判断する。第1のスレッドと第2のスレッドとに依存関係がなく、かつ第1のスレッドの共有資源へのアクセス頻度が所定頻度以上であれば、第1のCPUへ供給

するクロックの位相を第2のCPUへ供給するクロックの位相と異なる位相に変更する。

【0093】

第1のスレッドのアクセス頻度が多いほど、第1のスレッドから共有資源へのアクセスが第2のスレッドから共有資源へのアクセスと衝突する可能性が高い。そのため、各CPUへ供給されるクロックの位相が異なることで、複数のCPUから共有資源へのアクセスが衝突することを減少させることができる。また、CPUへ供給されるクロックの位相が異なることで、ピーク電流を減少させることができる。

【0094】

また、第1のCPUへ供給されるクロックの位相と第2のCPUに供給されるクロックの位相とが逆相となるように変更することにより、複数のCPUから共有資源へのアクセスが衝突することを減少させることができる。

【0095】

また、第1のCPUへ供給されるクロックの位相を変更するとともに、複数のCPUへ供給するクロックの周波数を上げるのではなく、共有資源へのクロックの周波数を上げる。これにより、消費電力を抑制しつつ、複数のCPUから共有資源へのアクセスが衝突することを減少させることができ、マルチコアプロセッサシステムの性能を向上させることができる。

【0096】

また、共有資源に供給されるクロックの周波数を第1のCPUへ供給されるクロックの周波数の2倍に変更する。複数のCPUから共有資源へのアクセスが衝突することを減少させることができ、複数のCPUから共有資源へのアクセスの集中により発生する待ち時間を削減することができる。

【0097】

また、第1のスレッドを実行するときの負荷量を第2閾値と比較し、負荷量が第2閾値よりも大きいときに、依存関係を判断する。第1のスレッドの負荷量が多いほど、第1のスレッドから共有資源へのアクセスが第2のスレッドから共有資源へのアクセスと衝突する可能性が高い。そのため、各CPUへ供給されるクロックの位相が異なることで、複数のCPUから共有資源へのアクセスが衝突することを減少させることができ、最小限に消費電力を抑制することができる。

【0098】

また、第1のスレッドと第2のスレッドとに依存関係があるときは、第1のCPUへ供給されるクロックの位相を変更しない。これにより、第1のスレッドと第2のスレッドとで共有しているデータや排他制御に影響が生じるのを防止することができる。

【0099】

また、依存関係がないとき、第1動作クロックの第1位相から第2位相への変更がされている場合、第1動作クロックの位相を第2位相から第1位相に変更する。これにより、複数のCPUから共有資源へのアクセスが衝突することを減少させることができる。

【0100】

また、依存関係は、共有メモリまたは第1プロセッサのメモリに登録される。これにより、第1のスレッドと第2のスレッドとの依存関係を容易に判断することができ、処理時間を短縮化することができる。

【0101】

また、依存関係があるスレッドを実行中のCPU群ごとにグループ化する。グループ化したCPU群が複数ある場合、CPU群ごとに該CPU群へ供給されるクロックの位相を異なるCPU群間で非同期となる位相に設定する。共有資源へ供給するクロックの周波数をいずれのCPU群とも同期する周波数に設定する。これにより、マルチコアプロセッサのCPU数が増加しても、アクセスの衝突を減少させることができ、マルチコアプロセッサの性能を向上させることができる。

【0102】

なお、本実施の形態で説明したスレッド処理方法は、あらかじめ用意されたプログラム

をパーソナル・コンピュータやワークステーション等のコンピュータで実行することにより実現することができる。本スレッド処理プログラムは、ハードディスク、フレキシブルディスク、CD-ROM、MO、DVD等のコンピュータで読み取り可能な記録媒体に記録され、コンピュータによって記録媒体から読み出されることによって実行される。また本スレッド処理プログラムは、インターネット等のネットワークを介して配布してもよい。

【0103】

上述した実施の形態に関し、さらに以下の付記を開示する。

【0104】

(付記1) 第1スレッドの実行命令を第1プロセッサに供給し、

前記第1スレッドと第2プロセッサで実行される第2スレッドとの依存関係を判断し、

前記第1スレッドの共有メモリまたは共有キャッシュメモリへのアクセス頻度を第1閾値と比較し、

前記アクセス頻度が第1閾値よりも大きいとき、かつ前記依存関係がないと判断されるときは、前記第1プロセッサの第1動作クロックの位相を変更すること、

を特徴とするスレッド処理方法。

【0105】

(付記2) 前記第1動作クロックの位相を前記第2プロセッサの第2動作クロックと逆相になるように変更すること、

を特徴とする付記1に記載のスレッド処理方法。

【0106】

(付記3) 前記第1動作クロックの位相を変更するとき、前記共有メモリまたは共有キャッシュメモリに供給される動作クロックの周波数を変更すること、

を特徴とする付記1または付記2に記載のスレッド処理方法。

【0107】

(付記4) 前記動作クロックの周波数を2倍に変更すること、

を特徴とする付記3に記載のスレッド処理方法。

【0108】

(付記5) 前記第1スレッドを実行するときの第1負荷を第2閾値と比較し、

前記第1負荷が前記第2閾値よりも大きく、前記アクセス頻度が第1閾値よりも大きく、かつ前記依存関係がないと判断されるときは、前記第1プロセッサの第1動作クロックの位相を変更すること、

を特徴とする付記1乃至付記4の何れか一に記載のスレッド処理方法。

【0109】

(付記6) 前記依存関係があるときは、前記第1動作クロックの位相を変更しないこと、

を特徴とする付記1乃至付記5の何れか一に記載のスレッド処理方法。

【0110】

(付記7) 前記依存関係がないとき、前記第1動作クロックの第1位相から第2位相へ変更されていた場合には、前記第1動作クロックの位相を前記第2位相から前記第1位相に変更すること、

を特徴とする付記1乃至付記6の何れか一に記載のスレッド処理方法。

【0111】

(付記8) 前記依存関係は、前記共有メモリまたは前記第1プロセッサのメモリに登録されること、

を特徴とする付記1乃至付記7の何れか一に記載のスレッド処理方法。

【0112】

(付記9) 依存関係があるスレッド間を関連付けるテーブルへアクセス可能な複数のコアのうちの一のコアが、

前記複数のコアの各コアで実行中のスレッド間に直接的または間接的に依存関係があるか否かを、前記テーブルを参照することで判断し、

判断結果に基づいて、前記複数のコアを直接的または間接的に依存関係があるスレッドを実行するコア群ごとにグループ化し、

前記グループ化したコア群が複数ある場合、前記コア群ごとに前記コア群へ供給するクロックの位相を前記コア群間で非同期となる位相に設定し、

前記複数のコアで共有する共有資源へ供給するクロックの周波数をいずれのコア群とも同期する周波数に設定すること、

を特徴とするスレッド処理方法。

【0113】

(付記10) 第1プロセッサと、

前記第1プロセッサと第2プロセッサとがアクセス可能な共有メモリまたは共有キャッシュメモリと、

前記第1プロセッサまたは前記共有メモリに供給される動作クロックを制御するクロック制御ユニットと、

を含み、

前記クロック制御ユニットは、

前記第1プロセッサで実行される第1スレッドの前記共有メモリまたは前記共有キャッシュメモリへのアクセス頻度と第1閾値との比較結果と前記第1スレッドと前記第2プロセッサが実行する第2スレッドとの依存関係とに基づいて、前記第1プロセッサの第1動作クロックの位相を変更すること、

を特徴とするスレッド処理システム。

【0114】

(付記11) 前記クロック制御ユニットは、前記比較結果が前記アクセス頻度が前記第1閾値よりも大きいことを示すとともに前記依存関係があるときに、前記第1動作クロックの位相を変更すること、

を特徴とする付記10に記載のスレッド処理システム。

【0115】

(付記12) 前記クロック制御ユニットは、前記第1プロセッサが前記第1スレッドを実行するときの第1負荷と第2閾値との比較結果に基づいて、前記第1プロセッサの第1動作クロックの位相を変更すること、

を特徴とする付記10または付記11に記載のスレッド処理システム。

【0116】

(付記13) 前記クロック制御ユニットは、前記第1動作クロックの位相を前記第2プロセッサの第2動作クロックと逆相になるように変更すること、

を特徴とする付記10乃至付記12の何れかに記載のスレッド処理システム。

【0117】

(付記14) 前記クロック制御ユニットは、前記第1動作クロックの位相を変更するとき、前記共有メモリまたは共有キャッシュメモリに供給される動作クロックの周波数を変更すること、

を特徴とする付記10乃至付記12の何れかに記載のスレッド処理システム。

【0118】

(付記15) 前記クロック制御ユニットは、前記動作クロックの周波数を2倍に変更すること、

を特徴とする付記14に記載のスレッド処理システム。

【符号の説明】

【0119】

200 マルチコアプロセッサシステム

201-1～201-4 CPU

203 2次キャッシュ

204 クロック生成回路

208 共有メモリ

4 0 0 アプリケーションテーブル

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/059570

A. CLASSIFICATION OF SUBJECT MATTER

G06F15/167(2006.01) i, G06F12/08(2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F15/167, G06F12/08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2011
Kokai Jitsuyo Shinan Koho	1971-2011	Toroku Jitsuyo Shinan Koho	1994-2011

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y A	JP 63-307529 A (NEC Corp.), 15 December 1988 (15.12.1988), page 2, lower left column, line 12 to page 3, lower right column, line 8; fig. 1 to 2 (Family: none)	9 1-8,10-15
Y A	JP 1-142940 A (Toshiba Corp.), 05 June 1989 (05.06.1989), page 3, upper left column, line 2 to page 4, upper left column, line 1; fig. 1 to 2 (Family: none)	9 1-8,10-15
Y	JP 2007-328415 A (Waseda University), 20 December 2007 (20.12.2007), paragraphs [0034], [0082] to [0085] & US 2007/0283358 A1	9

☒ Further documents are listed in the continuation of Box C.
 ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
12 July, 2011 (12.07.11)Date of mailing of the international search report
19 July, 2011 (19.07.11)Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/059570

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2003-108389 A (Fujitsu Ltd.), 11 April 2003 (11.04.2003), paragraphs [0032] to [0038] (Family: none)	9

国際調査報告		国際出願番号 PCT/J P 2 0 1 1 / 0 5 9 5 7 0	
A. 発明の属する分野の分類 (国際特許分類 (IPC)) Int.Cl. G06F15/167(2006.01)i, G06F12/08(2006.01)i			
B. 調査を行った分野 調査を行った最小限資料 (国際特許分類 (IPC)) Int.Cl. G06F15/167, G06F12/08			
最小限資料以外の資料で調査を行った分野に含まれるもの 日本国実用新案公報 1922-1996年 日本国公開実用新案公報 1971-2011年 日本国実用新案登録公報 1996-2011年 日本国登録実用新案公報 1994-2011年			
国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)			
C. 関連すると認められる文献			
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号	
Y A	JP 63-307529 A (日本電気株式会社) 1988.12.15, 2 頁左下欄 1 2 行目～3 頁右下欄 8 行目、第 1 図～第 2 図 (ファミリーなし)	9 1-8、10-15	
Y A	JP 1-142940 A (株式会社東芝) 1989.06.05, 3 頁左上欄 2 行目～4 頁左上欄 1 行目、第 1 図～第 2 図 (ファミリーなし)	9 1-8、10-15	
☒ C 欄の続きにも文献が列挙されている。 ☐ パテントファミリーに関する別紙を参照。			
* 引用文献のカテゴリー 「A」特に関連のある文献ではなく、一般的技術水準を示すもの 「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの 「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す) 「O」口頭による開示、使用、展示等に言及する文献 「P」国際出願日前で、かつ優先権の主張の基礎となる出願日の後に公表された文献 「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの 「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの 「Y」特に関連のある文献であって、当該文献と他の 1 以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの 「&」同一パテントファミリー文献			
国際調査を完了した日 12.07.2011		国際調査報告の発送日 19.07.2011	
国際調査機関の名称及びあて先 日本国特許庁 (ISA/J P) 郵便番号 100-8915 東京都千代田区霞が関三丁目 4 番 3 号		特許庁審査官 (権限のある職員) 三坂 敏夫	5 B 4 1 7 8 電話番号 03-3581-1101 内線 3545

国際調査報告		国際出願番号 PCT/J P 2 0 1 1 / 0 5 9 5 7 0
C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
Y	JP 2007-328415 A (学校法人早稲田大学) 2007. 12. 20, 段落 3 4、8 2 ~ 8 5 & US 2007/0283358 A1	9
A	JP 2003-108389 A (富士通株式会社) 2003. 04. 11, 段落 3 2 ~ 3 8 (ファミリーなし)	9

フロントページの続き

- (72)発明者 栗原 康志
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- (72)発明者 大友 俊也
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- (72)発明者 大館 尚記
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- Fターム(参考) 5B045 DD01 DD12 EE03 EE11

(注) この公表は、国際事務局(W I P O)により国際公開された公報を基に作成したものである。なおこの公表に係る日本語特許出願(日本語実用新案登録出願)の国際公開の効果は、特許法第184条の10第1項(実用新案法第48条の13第2項)により生ずるものであり、本掲載とは関係ありません。