

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4459628号
(P4459628)

(45) 発行日 平成22年4月28日(2010.4.28)

(24) 登録日 平成22年2月19日(2010.2.19)

(51) Int.Cl.

F 1

G06F 12/08 (2006.01)
G06F 12/10 (2006.01)G06F 12/08 507Z
G06F 12/08 565
G06F 12/10 551Z

請求項の数 2 (全 11 頁)

(21) 出願番号 特願2003-581019 (P2003-581019)
 (86) (22) 出願日 平成15年3月13日 (2003.3.13)
 (65) 公表番号 特表2006-505834 (P2006-505834A)
 (43) 公表日 平成18年2月16日 (2006.2.16)
 (86) 國際出願番号 PCT/US2003/007710
 (87) 國際公開番号 WO2003/083663
 (87) 國際公開日 平成15年10月9日 (2003.10.9)
 審査請求日 平成18年3月13日 (2006.3.13)
 (31) 優先権主張番号 10/104,815
 (32) 優先日 平成14年3月22日 (2002.3.22)
 (33) 優先権主張国 米国(US)

(73) 特許権者 593096712
 インテル コーポレイション
 アメリカ合衆国 95052 カリフォルニア州 サンタクララ ミッション カレッジ ブールバード 2200
 (74) 代理人 100070150
 弁理士 伊東 忠彦
 (74) 代理人 100091214
 弁理士 大貫 進介
 (74) 代理人 100107766
 弁理士 伊東 忠重
 (72) 発明者 ハマーランド, パー¹⁰
 アメリカ合衆国 97124 オレゴン州
 ヒルズボロ ノースイースト セカンド ドライブ 2601

最終頁に続く

(54) 【発明の名称】キャッシュメモリにおけるコンテキスト識別子の使用

(57) 【特許請求の範囲】

【請求項 1】

第1のスレッドで使用されるページディレクトリのベースのアドレスと、第2のスレッドで使用されるページディレクトリのベースのアドレスとを比較し、前記第1のスレッドで使用されるページディレクトリのベースのアドレスが、前記第2のスレッドで使用されるページディレクトリのベースのアドレスと一致するかどうかを決定し、

前記第1のスレッドで使用されるページディレクトリのベースのアドレスが、前記第2のスレッドで使用されるページディレクトリのベースのアドレスと一致する場合に、前記第1のスレッド及び前記第2のスレッドの双方に同一のコンテキスト識別子を割り当てること。

前記第1のスレッドで使用されるページディレクトリのベースのアドレスが、前記第2のスレッドで使用されるページディレクトリのベースのアドレスと一致しない場合に、前記第1のスレッド及び前記第2のスレッドに異なるコンテキスト識別子を割り当てることを有する方法。

【請求項 2】

プロセッサにより発行された命令に応じて、アドレスを生成するアドレス生成ユニット(A G U)と、

複数のキャッシュラインを保存し、前記複数のキャッシュラインのそれぞれは少なくともコンテキスト識別子と、タグと、データブロックとを有するキャッシュと、

前記 A G U に結合され、前記キャッシュに結合され、前記 A G U によって生成されたアドレス部分と、前記複数のキャッシュラインのそれぞれの各タグとを比較する少なくとも

10

20

1つの比較手段と、

実行中のコンピュータの命令の第1のスレッドに関連付けられたページディレクトリのベースのアドレスを保存する第1のレジスタと、

実行中のコンピュータの命令の第2のスレッドに関連付けられたページディレクトリのベースのアドレスを保存する第2のレジスタと、

前記第1のレジスタと前記第2のレジスタとに結合され、前記第1のレジスタに保存された値と、前記第2のレジスタに保存された値とを比較する他の比較手段と、

少なくとも2つのコンテクスト識別子を生成するコンテクスト識別子生成手段と、

前記コンテクスト識別子生成手段と前記他の比較手段とに結合され、前記少なくとも2つのコンテクスト識別子のうち1つを選択するマルチプレクサと、

前記マルチプレクサに結合され、前記キャッシュに結合され、前記マルチプレクサによって選択されたコンテクスト識別子と、前記複数のキャッシュラインのそれぞれの各コンテクスト識別子とを比較する少なくとも1つの比較手段と、

を有する装置。

【発明の詳細な説明】

【発明の詳細な説明】

【0001】

[技術分野]

本発明はコンピュータ技術に関する。特に、本発明は、コンピュータ装置のキャッシュメモリの性能を改善することに関する。

[背景技術]

コンピュータメモリの階層内で1つ以上のキャッシュメモリシステムを使用することは、コンピュータの性能を増大させる周知の技術である（例えば、1998年Academic PressによるHandy, JimのThe Cache Memory Bookを参照）。図1は典型的なキャッシュメモリアレイ100を示したものである。キャッシュメモリアレイ100は、キャッシュライン110を含む。各キャッシュラインは、タグ120とデータブロック130とを含む。例示のキャッシュライン140は、タグ150とデータブロック160とを含む。参考番号170は、例示のタグ150がメインメモリのアドレス170の一部であることを示す。メインメモリのアドレス170は、データブロック160に対応するメインメモリのアドレスである。

【0002】

プロセッサは、メモリ転送の間、プロセッサの実行コア部とコンピュータメモリの階層との間で、命令とオペランドを往復して転送する。メモリ転送の例は、メモリの階層からプロセッサに命令／オペランドをロードし、プロセッサからメモリの階層に命令／オペランドを保存することである。メモリ転送の間、プロセッサはメインメモリのアドレスを生成する。メインメモリのアドレスの一部が、キャッシュ検索の間にタグ120のエントリと比較され、キャッシュメモリ100が、メモリ転送に対応するエントリを有するかどうか決定する。タグ150とメインメモリのアドレス170との間の関係によって示される通り、キャッシュ検索の処理は、プロセッサが各メインメモリのアドレスの一部のみとタグの各エントリとを比較することによって加速される。一般的に、キャッシュメモリはプロセッサにより生成された各リニアアドレスの一部を用いて、キャッシュアレイ100に保存されるデータに索引を付ける。

【0003】

スレッドは、コンピュータプログラムの他の部分と関係なく実行し得るコンピュータプログラムの一部である。複数のスレッドがプロセッサで同時に実行されると、プロセッサの性能は高められ得る。複数のスレッドの多様な命令の間の依存関係が検出され、適切に管理されると、複数のスレッドの同時の実行が可能になる。

【0004】

図2は、多くのIntel（商標）プロセッサが、仮想メモリ環境を使用して、多くのリニアアドレス空間が小さい量の物理メモリ（例えば、ランダムアクセスメモリ）によっ

10

20

30

40

50

て対応可能にする方法を示したものである。メモリ転送の間、プロセッサはリニアアドレス 210 を生成する。リニアアドレス 210 は、ディレクトリ領域 220 と、テーブル領域 225 と、オフセット領域 230 とを有する。ページディレクトリ 235 のベースは、制御レジスタ CR3240 に含まれる。リニアアドレス 210 のディレクトリエントリ 220 は、制御レジスタ CR3240 に含まれる値へのオフセットを与える。ページディレクトリは、ページ領域のベースポインタ 245 を含む。テーブル領域 225 は、ページテーブルのベースポインタ 245 に結びつけられたオフセットを備え、物理アドレス 255 を含むページのベースを識別する。オフセット領域 230 は、ページテーブルのエントリと結びつけられ、物理アドレス 255 を識別する。

【0005】

10

図 3 は、一部のメインメモリのアドレスのみのキャッシュ検索を実行することに関する欠点を示したものである。プロセッサ(図示なし)は、ロード命令 LD0 と LD1 に応じて、リニアアドレス 304 と 306 とを生成する。305 と 307 は、キャッシュ検索を実行するために用いられる各アドレスの一部を示したものである。305 と 307 が同一に見えるが、それぞれ 304 と 306 の一部にしか過ぎない。304 と 306 は、各アドレスがそれぞれのディレクトリ領域(320 と 325)とオフセット領域(330 と 335)に異なるエントリを有するため、2つの異なる物理アドレスに位置する。図 3 の通り、プロセッサが複数のスレッドの同時実行に対応する場合は、追加の複雑な状態が持ち込まれる。スレッド 0 とスレッド 1 は、それぞれのページディレクトリ(340 と 345)のベースに異なる値を有し得る。従って、304 と 306 が同じであっても、それらは 2 つの異なる物理アドレスに位置する。

[詳細な説明]

20

本発明は例示の目的で示され、限定の目的で示されるのではない。添付の図面の図において、類似の参照番号は類似の要素を参照する。

【0006】

30

本願でキャッシュメモリのコンテクスト識別子の使用について説明される。以下の説明において、本発明の実施例の十分な理解を提供するため、多数の特定の詳細が提供される。しかし、関連技術に熟練した人は、1つ以上の特定の詳細を用いずに、又は他の方法や、構成要素や、材料等を用いて、本発明が実行され得ることがわかるだろう。他の場合において、本発明の形態を不明確にすることを回避するために、周知の構造や、材料や、動作が、詳細に示されない、又は説明されない。

【0007】

本明細書を通じて、“一実施例”又は“実施例”への言及は、本発明に関連して説明される特定の機能又は構造又は特徴が、本発明の少なくとも1つの実施例に含まれることを意味する。従って、本明細書を通じて多様な場所における“一実施例において”又は“実施例において”という語句が現れるることは、必ずしも全てが同じ実施例を参照するのではない。さらに、特定の機能又は構造又は特徴は、1つ以上の実施例において、何らかの適当な方法と組み合わせられ得る。

【0008】

40

キャッシュメモリシステムは、タグと呼ばれるリニアアドレスの一部でキャッシュラインに索引を付ける。プロセッサは、メモリ転送動作の間にリニアアドレスを生成する。キャッシュ検索の間に、前記リニアアドレスの一部がタグと比較される。多くの Intel(商標) プロセッサは、一部のアドレスとタグとの間の見かけ上の一致が真の一致であることを確保する仕組みを提供する。前記仕組みは、ヒット/ミス決定手段と呼ばれる。ヒット/ミス決定手段は、キャッシュメモリのキャッシュ検索処理と同時に動作する。前記ヒット/ミス決定手段は、発行された命令に関連付けられたリニアアドレスを、対応する物理アドレスに変換し、前記一部のリニアアドレスと一致するタグに関連付けられた完全なアドレスと、物理アドレスとを比較する。前記ヒット/ミス決定手段は、リニアアドレスを物理アドレスに変換しなければならないため、及び、完全なアドレスの比較を実行するため、前記ヒット/ミス決定手段はキャッシュメモリよりかなり遅い。キャッシュメモリ

50

リが誤った一致を引き起こすことを前記ヒット／ミス決定手段が検出すると、前記ヒット／ミス決定手段はキャッシュメモリによって行われた誤りを訂正する。キャッシュメモリで誤った一致によって持ち込まれた誤りを検出し、それに対応する処理は、プロセッサの性能を低下し得る。

【0009】

図4Aは、コンテクスト識別子を組み込んだキャッシュメモリアレイの一実施例を示したものである。キャッシュメモリアレイ400は、複数のキャッシュライン410を含む。各キャッシュライン410は、コンテクスト識別子405と、タグ420と、データブロック430とを含む。従って、キャッシュライン440Aは、コンテクスト識別子415Aとタグ450Aとを含む。同様に、キャッシュライン440Bは、コンテクスト識別子415Bとタグ450Bとを含む。2つの異なるスレッドがコンテクスト識別子415Aと415Bとを生成する。
10

【0010】

リニアアドレス407Aと470Bがタグ450Aと450Bとに対応する。図4Aと4Bにおいて、リニアアドレスはコンテクスト識別子を含み、類似のリニアアドレスの間を識別するためのコンテクスト識別子の用い方について示す。コンテクスト識別子は、リニアアドレスの一部ではないが、前記リニアアドレスと別に生成される。

【0011】

先行技術に従って実現されたキャッシュメモリシステムは、プロセッサ（図示なし）によって生成された一部のリニアアドレスと、キャッシュアレイ400の各タグ420とを比較することによってキャッシュ検索を行う。プロセッサが、123という一部のリニアアドレスを生成した場合は、前記アドレスは、タグ450Aとタグ450Bとの双方に一致する。前述の通り、ヒット／ミス決定手段は、どちらのキャッシュラインが前記プロセッサにより生成されたメインメモリのアドレスに一致するかどうかを最終的に決定する。先行技術のシステムは、前記ヒット／ミス決定手段が一部のリニアアドレスとタグ420との間の誤った一致を管理して訂正する場合に、性能の不利を受ける。
20

【0012】

図4Bは、前記プロセッサにより生成された一部のリニアアドレスとタグ420との間の誤った一致の頻度を減少させるコンテクスト識別子の使用を示したものである。リニアアドレス470Aと470Bは、図4Aのキャッシュライン440Aと440Bに関連付けられる。発行された命令に応じて、アドレス475がプロセッサによって生成される。アドレス475は、コンテクスト識別子480と一部のリニアアドレス485とを含む。一部のリニアアドレス485がタグ450Bと一致するため、先行技術のキャッシュは475と470Bとの間の誤った一致を識別する場合がある。一実施例において、コンテクスト識別子480がコンテクスト識別子415Bと比較される。また、一実施例において、一部のリニアアドレス485がタグ450Bと比較される。従って、コンテクスト識別子480がコンテクスト識別子415Bと一致しないため、キャッシュアレイ400は475と470Bとの間の誤った一致を見つける。
30

【0013】

図5は、ページミスハンドラー（PMH）の一部の一実施例を示したものである。一実施例において、PMH500は、コンテクスト識別子の割り当てを生成し、制御する。PMH500は、コンテクスト識別子生成手段510と、比較手段520と、コンテクスト識別子マルチプレクサ530と、インバーター540とを含む。一実施例において、コンテクスト識別子生成手段510は、現在実行するスレッドに対応する1ビットのバイナリ番号を生成する。本発明の他の実施例において、コンテクスト識別子生成手段510は、1ビット以上のバイナリ番号を生成する。比較手段520は、プロセッサで実行する2つのスレッドにより用いられるページディレクトリのベースのアドレスを比較する。異なる実施例において、コンテクスト識別子は2つ以上のビットを有する。
40

【0014】

一実施例によると、PMH500は、同じコンテクスト識別子が2つの異なるスレッド
50

の命令に割り当てられているかどうかを決定する。図5に示される実施例では、PMH500は、前記プロセッサで実行する第1のスレッドに第1のコンテクスト識別子を割り当てる。例えば、スレッド0が前記プロセッサで実行する前記第1のスレッドである場合は、前記プロセッサは、スレッド0のコンテクスト識別子として0を割り当て得る。コンテクスト識別子は、スレッド番号と一致する必要はない。従って、スレッド0は1又は0のコンテクスト識別子を有し得る。本発明の他の実施例において、プロセッサは2以上のスレッドの命令を実行し、1ビット以上のコンテクスト識別子が使用される。

【0015】

第2のスレッドがプロセッサで実行し始めると(例えば、スレッド1)、PMH500は、前記第2のスレッドの命令に、スレッド0の命令に割り当てられたコンテクスト識別子と異なるコンテクスト識別子を割り当てるかどうかを決定する。スレッド0とスレッド1がメインメモリの同じページを使用することが可能である。その場合は、リニアアドレスがタグと同じページに位置することが知られている場合に、タグとリニアアドレスの間の誤った一致があまり生じないため、同じコンテクスト識別子を2つのスレッドの命令に割り当てることが望ましい。比較手段520は、2つのスレッドにより用いられるページディレクトリのベースを比較することにより、スレッド0とスレッド1がメインメモリで同じページを使用しているかどうかを決定する。

【0016】

スレッド0とスレッド1が、それぞれのページディレクトリのベースとして、同じアドレスを使用している場合、マルチプレクサ530は、スレッド0に選択されたものと同じコンテクスト識別子をスレッド1にも選択する。逆に言えば、スレッド0とスレッド1がページディレクトリのベースに同じアドレスを使用していないことを比較手段520が示すと、マルチプレクサ530は、スレッド1のコンテクスト識別子として、インバーター540の出力を選択する。2つのスレッドが同じコンテクスト識別子を共有するかどうかを決定することは、異なる方法で実現され得る。例えば、本発明の実施例においては、2つのスレッドが同じコンテクスト識別子を共有しているかどうかをオペレーティングシステムが決定し、プロセッサに正しいコンテクスト識別子を明示的に知らせ、スレッドに関連づける。本発明の更に他の実施例においては、スレッドのメモリアクセスの傾向を学習する予測手段が、コンテクスト識別子を予測する。一実施例によると、2つの異なるスレッドの命令に同じ値を常に選択するように、又は2つの異なるスレッドの命令に異なるコンテクスト識別子の値を常に選択するようにユーザがPMH500に指示し得る。また、ある実施例においては、ベースディレクトリのアドレスの他に追加の基準を使用し、2つのスレッドが同じコンテクスト識別子を共有するかどうかを決定する。

【0017】

本発明の他の実施例において、前記プロセッサに2つ以上のスレッドが存在する。前記プロセッサにn個のスレッドが存在する実施例においては、前記コンテクスト識別子は、n個の異なるコンテクスト識別子が特定されることを可能にするエンコードの仕組みまで拡張される。nビットのコンテクスト識別子が使用される場合は、単一のビットがキャッシュに位置されるのと同じ方法でnビットがキャッシュに位置されることが、その技術に通常熟練した人はわかるだろう。また、2つの単一のビットのコンテクスト識別子を比較するのとほぼ同じ方法で、nビットのコンテクスト識別子をお互いに比較することが行われ得ることが、その技術に通常熟練した人はわかるだろう。

【0018】

複数のスレッドを備える本発明の実施例は、未使用的コンテクスト識別子のプールを使用する。スレッドがページテーブルのベース(CR3)を変更すると、他のアクティブなスレッドで使用されるページテーブルのベースの値、又は前記プールのコンテクスト識別子の何らかの有効なページテーブルのベースの値と新しい値が比較される。一致している場合には、前記一致を引き起こしたスレッドのコンテクスト識別子の値が取られる。他のアクティブなスレッドと一致しない場合には、前記プールから未使用的コンテクスト識別子が取られる。スレッドがページのベースの値を変更する場合、又は所定のコンテクスト

10

20

30

40

50

識別子を備えたキャッシュの最後のラインが割り当てを外される場合に、コンテクスト識別子は未使用のプールに戻される。後者の構成は、以前のスレッドに持ち込まれたデータでスレッドがヒットすることを可能にする。スレッドのコンテクスト識別子の値を予測する多くの仕組みが使用され得ることが、その技術に通常熟練した人はわかるだろう。

【0019】

図6は、コンテクスト識別子を有するキャッシュメモリシステムからの読み取りを示したものである。キャッシュアレイ610は、キャッシュライン605を含む。各キャッシュライン605は、コンテクスト識別子610Aと、タグ610Bと、データブロック610Cとを含む。キャッシュアレイ610はキャッシュコントローラ620に結合される。キャッシュコントローラ620は、キャッシュアレイ610に含まれるデータへのアクセスを制御する。10

【0020】

キャッシュコントローラ620は、ヒット/ミス決定手段670及び比較手段675と680に結合される。ヒット/ミス決定手段670及び比較手段675と680の動作は、以下に説明する。一実施例において、比較手段675はAGU630とANDゲート685とに結合される。AGU630は、メモリ転送を必要とする発行された命令（例えば、ロードと保存の命令）のためのリニアアドレスを生成する。一実施例において、比較手段680は、マルチブレクサ650を通して、ANDゲート685とPMH640とに結合する。PMH640は、どのコンテクスト識別子がどの発行された命令に割り当てられるかを決定するために用いられる制御ロジックを含む。20

【0021】

一般的に、読み取り周期は、発行されたロード命令(LD)660によって開始される。LD660は、PMH640とAGU630とに送信される。特に、PMH640はどのスレッドがLD660の出所であるかを決定する。PMH640は、一つには、どのスレッドがLD660の出所であるかに基づいて、LD660にコンテクスト識別子を割り当てる。PMH640は、LD660に関連付けられたコンテクスト識別子を、マルチブレクサ650を通して、比較手段680に発行する。同様に、AGU630は、一部のリニアアドレスを比較手段675に発行する。

【0022】

比較手段675と680は前記一部のリニアアドレスとLD660のコンテクスト識別子とを用い、LD660が必要とする情報をどのキャッシュアレイ610が含むかを決定する。キャッシュアレイ610が必要な情報を含む場合、キャッシュヒットが結果として生じる。キャッシュアレイ610が必要な情報を含まない場合、キャッシュミスが結果として生じる。30

【0023】

特定のキャッシュラインが、LD660のコンテクスト識別子に一致するコンテクスト識別子と、LD660に対応する一部のリニアアドレスに一致するタグとを含むことを、キャッシュヒットは要求する。比較手段675と680は、各キャッシュラインのタグ領域及びコンテクスト識別子領域と、LD660のコンテクスト識別子及び一部のリニアアドレスとを比較する。一実施例によると、比較が単一の平行動作で行われ得るように、キャッシュラインと同じだけの比較手段が存在する。単一のキャッシュラインが、LD660のコンテクスト識別子及び一部のリニアアドレスに一致するコンテクスト識別子及びタグを含む場合、ANDゲート685がキャッシュヒットの信号を出す。しかし、比較手段675がLD660のアドレスの一部のみと、キャッシュアレイ610Aの各タグとを比較すると、キャッシュヒットが誤った一致に基づくという可能性が存在する。40

【0024】

ヒット/ミス決定手段670は、キャッシュヒットが誤った一致に基づくかどうかを検出する。ヒット/ミス決定手段は、LD660が命令/オペランドを要求するメインメモリの位置の完全なリニアアドレスを受信する。一実施例によると、ヒット/ミス決定手段670は、AGUによってそれに伝達されるリニアアドレスを物理アドレスに変換する。50

ヒットノミス決定手段 670 は、LD660 の一部のリニアアドレスに一致するタグに対応する物理アドレスを決定する。ヒットノミス決定手段 670 は、完全な物理アドレスを比較し、キャッシュヒットが誤った一致に基づくかどうか決定する。

【0025】

図 7 は、コンテクスト識別子を用いたキャッシュメモリへの書き込みを示したものである。キャッシュアレイ 710 は複数のキャッシュライン 705 を含む。各キャッシュライン 705 はコンテクスト識別子 710A と、タグ 710B と、データブロック 710C とを含む。キャッシュアレイ 710 は、キャッシュコントローラに結合される。キャッシュコントローラ 720 は、どのキャッシュラインにデータのブロックが書き込まれるかを決定することを含む複数の機能を実行する。キャッシュコントローラ 720 は、マルチブレクサ 750 を通して AGU730 と PMH740 とに結合される。10

【0026】

発行された命令 760 がプロセッサ（図示なし）から PMH740 と AGU730 とに送信される。一実施例において、発行された命令 760 がメモリ転送を要求する場合に、AGU730 はリニアアドレスを生成する。AGU730 は、概してプロセッサの実行コアの一部である。他の実施例において、AGU730 は物理アドレスを生成し得る。一実施例において、PMH740 は、マルチブレクサ 750 を制御する少なくとも 2 つのコンテクスト識別子と制御信号とを生成する。他の実施例において、PMH740 は何からの数のコンテクスト識別子を生成し得る。20

【0027】

一般的に、書き込み周期は、発行された保存命令（ST）（例えば、ST760）によって開始される。ST760 は、PMH740 と AGU730 とに送信される。PMH740 は、どのスレッドが ST760 の出所であるか決定する。PMH740 は、どのコンテクスト識別子が ST760 に割り当てられるかを更に決定する。一実施例によると、PMH740 は、3 つの動作モード：OFF と NON-SHARED と ADAPTIVE を有する。PMH740 が OFF モードで動作している場合、PMH740 は、各コンテクスト識別子に同じ値を書き込み得る。スレッドがキャッシュラインに保存されたコンテクスト識別子と同じコンテクスト識別子を有する場合、2 つ以上のスレッドの命令によってキャッシュラインが共有される。30

【0028】

NON-SHARED モードで動作する場合、PMH740 は発行された命令を含むスレッドに特有のコンテクスト識別子を割り当て得る。一実施例において、ADAPTIVE モードにより、異なるスレッドから生じる同じ又は異なるコンテクスト識別子を割り当てるかどうかを PMH740 が動的に判断することが可能になる。一実施例において、スレッドがページディレクトリのベースに同じアドレスを共有する場合は、PMH740 は、スレッドに含まれる命令に同じコンテクスト識別子を動的に割り当てる。ある実施例においてそうでない場合は、PMH740 は、スレッドに含まれる命令に異なるコンテクスト識別子を動的に割り当てる。40

【0029】

一実施例において、2 つの異なるスレッドの命令に同じコンテクスト識別子を書き込むかどうか決定する場合に、PMH740 は、コンピュータがマルチスレッドのモードにあるかどうかと、ページングが可能かどうかとを検討する。発行された命令に割り当てられるコンテクスト識別子の値を決定するために、いくつかの基準が用いられ得る。更に、コンテクスト識別子生成手段と、関連する制御ロジックとが、電子システム内の複数の場所に配置され得る。そして、それらは PMH 又は類似のユニットに含まれる必要はない。50

【0030】

PMH740 は、ST760 に適切なコンテクスト識別子を、マルチブレクサ 750 を通してキャッシュコントローラ 720 に送信する。AGU730 は ST760A がデータを保存するために探すメインメモリの場所に対応したリニアアドレスを生成する。AGU730 は、キャッシュコントローラ 720 とヒットノミス決定手段 770 に ST760 に

対応するリニアアドレスを提供する。キャッシュコントローラ 720 は、どのキャッシュラインが S T 760 のデータを保存するために使用されるかを決定するキャッシュライン交換方法を用いる。キャッシュライン交換方法は、その技術において周知である（キャッシュライン交換方法のより詳細な説明は、1998 年 Academic Press による Handy, Jim の The Cache Memory Book を参照）。

【0031】

適切なキャッシュラインを選択した後、キャッシュコントローラ 720 は前記選択されたキャッシュライン 780 のコンテクスト識別子領域 780A に、コンテクスト識別子を書き込む。キャッシュコントローラ 720 は、キャッシュライン 780 のタグ領域 780B に S T 760 のリニアアドレスの一部を書き込む。一実施例において、前記タグは、3 10 2 ビットリニアアドレスの 23 から 12 ビットを有する。前記タグに使用され得るアドレスビットの多くの組み合わせが存在する。キャッシュコントローラは、選択されたキャッシュライン 780 のデータブロック領域 780C に S T 760 に関連付けられたデータを書き込む。

【0032】

キャッシュ管理は、電子的にアクセス可能な媒体からの命令によって制御され得る。それは、本願で説明した処理を実行するコンピュータ（又は他の電子装置）をプログラムするためには用いられ得る。電子的にアクセス可能な媒体は、フロッピー（登録商標）ディスクや、光ディスクや、CD-ROM や、光磁気ディスクや、ROM や、RAM や、EPR 20 OM や、EEPROM や、磁気カード又は光カードや、フラッシュメモリや、電子命令を保存するのに適した他の形式の媒体 / 機械が読み取り可能な媒体を含むが、それに限定されない。さらに、前記命令はまた、コンピュータプログラム製品としてダウンロードされ得る。前記プログラムは、通信リンク（例えば、モデム又はネットワーク接続）を介して搬送波又は他の伝播媒体に組み込まれたデータ信号を用いて、リモートコンピュータから要求中のコンピュータに転送され得る。

【0033】

本発明の示された実施例の前述の説明は、網羅的にすることを意図するものではなく、また、開示された正確な形式に本発明を限定することを意図するものではない。本発明の特定の実施例及び例は、説明の目的で本願において開示されるが、関連技術に熟練した人がわかるように、本発明の範囲ないで多様な同等の改良が可能である。

【0034】

前記改良は、前述の詳細な説明を踏まえて行われ得る。特許請求の範囲で使用される用語は、明細書及び特許請求の範囲で開示された特定の実施例に本発明を限定するために解釈されるべきではない。むしろ、本発明の範囲は、特許請求の範囲によって完全に決定されるべきであり、特許請求の範囲の解釈の確立した原理に従って解釈されるべきである。

【図面の簡単な説明】

【0035】

【図 1】先行技術のキャッシュメモリアレイの図である。

【図 2】リニアアドレスを物理アドレスに位置付ける概念図である。

【図 3】2 つの異なる物理アドレスに位置する 2 つの類似のタグの概念図である。

【図 4 A】コンテクスト識別子を用いたキャッシュメモリアレイの一実施例の図である。

【図 4 B】コンテクスト識別子の一実施例の概念図である。

【図 5】コンテクスト識別子生成手段と関連する制御回路の一実施例の簡略化したブロック図である。

【図 6】キャッシュメモリシステムの一実施例のブロック図である。

【図 7】キャッシュメモリシステムの一実施例のブロック図である。

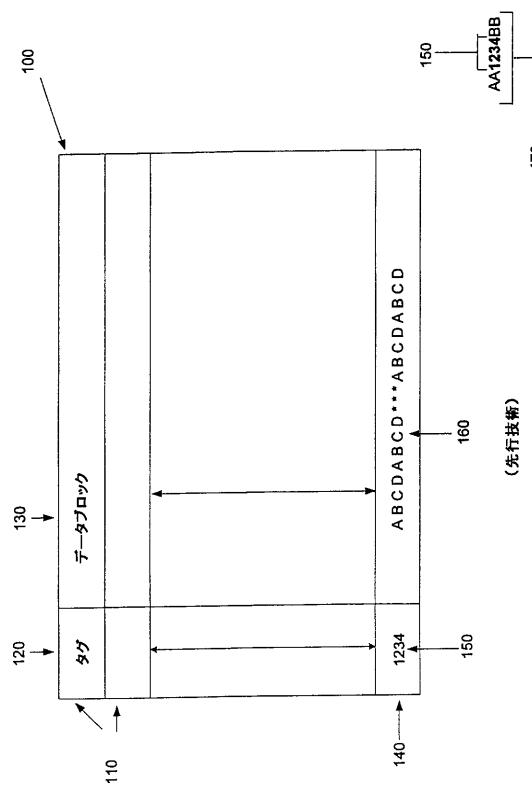
10

20

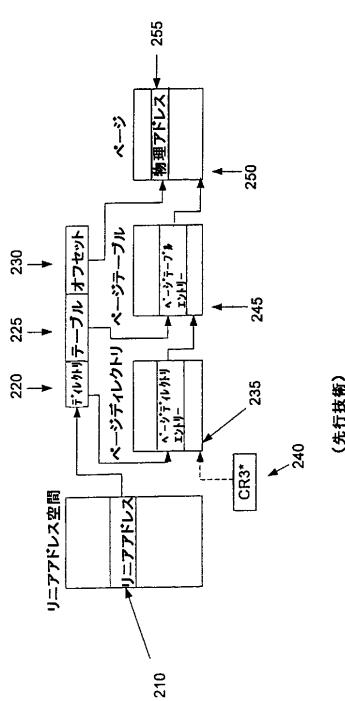
30

40

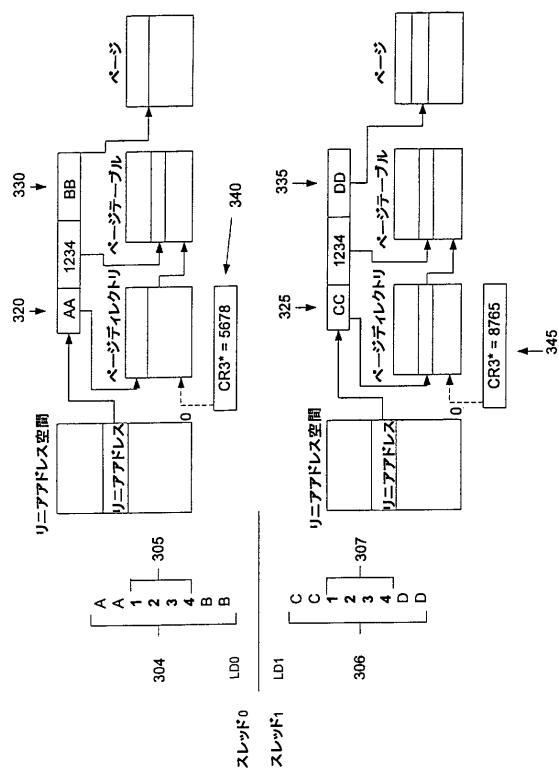
【図1】



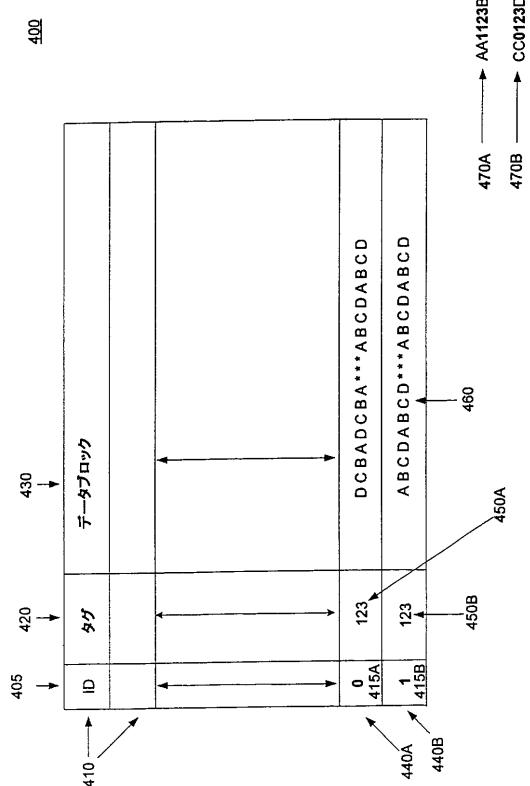
【図2】



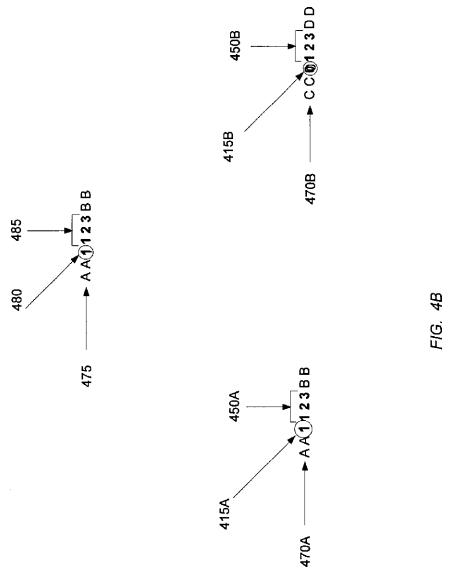
【図3】



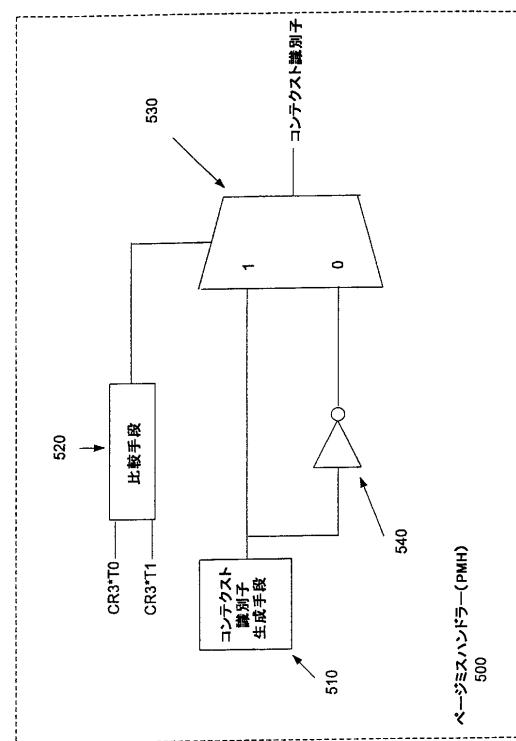
【図4 A】



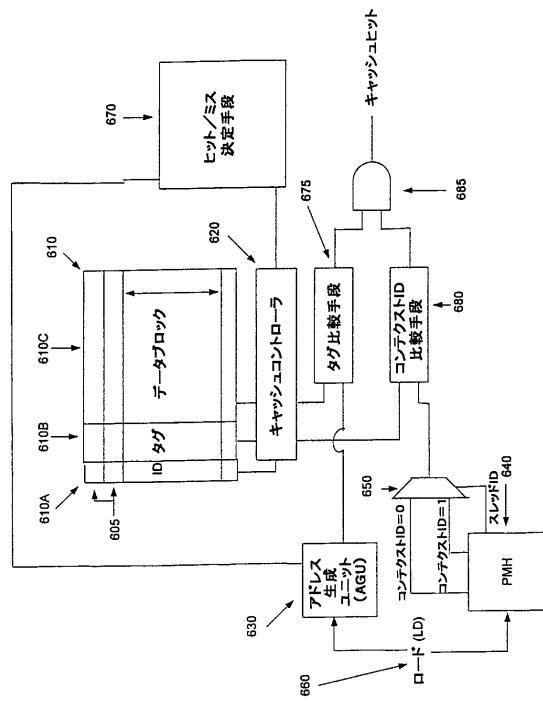
【図4B】



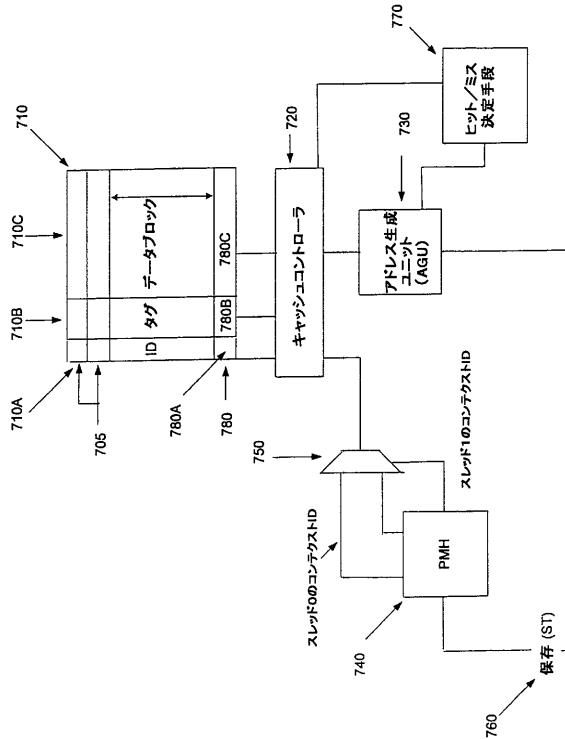
【図5】



【図6】



【図7】



フロントページの続き

(72)発明者 ヴェンカットラマン , ケイ エス
アメリカ合衆国 97123 オレゴン州 ヒルズボロ サウスイースト ラングウッド ストリ
ート 7219

(72)発明者 バクサ , アラヴィンド
アメリカ合衆国 97229 オレゴン州 ポートランド ノースウェスト ファルコンリッジ
レーン 14362

(72)発明者 アップトン , マイケル
アメリカ合衆国 97210 オレゴン州 ポートランド ノースウェスト 24ス アヴェニュー
- 1410

審査官 清木 泰

(56)参考文献 特開平10-187538 (JP, A)
特開平03-235144 (JP, A)
特開平03-235143 (JP, A)

(58)調査した分野(Int.Cl. , DB名)
G06F12/08-12/12