

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6696987号

(P6696987)

(45) 発行日 令和2年5月20日 (2020.5.20)

(24) 登録日 令和2年4月27日 (2020.4.27)

(51) Int. Cl. F I  
**GO 6 F 12/0802 (2016.01)** GO 6 F 12/0802  
**GO 6 F 12/1027 (2016.01)** GO 6 F 12/1027

請求項の数 16 (全 20 頁)

(21) 出願番号	特願2017-534307 (P2017-534307)	(73) 特許権者	390023641
(86) (22) 出願日	平成27年12月10日 (2015.12.10)		ウイスコンシン アラムナイ リサーチ
(65) 公表番号	特表2018-504694 (P2018-504694A)		ファウンデーション
(43) 公表日	平成30年2月15日 (2018.2.15)		WISCONSIN ALUMNI RE
(86) 国際出願番号	PCT/US2015/064955		SEARCH FOUNDATION
(87) 国際公開番号	W02016/105961		アメリカ合衆国 ウイスコンシン 537
(87) 国際公開日	平成28年6月30日 (2016.6.30)		26、マディソン、ウオルナット ストリ
審査請求日	平成30年9月27日 (2018.9.27)		ート 614
(31) 優先権主張番号	62/097, 342	(74) 代理人	100107364
(32) 優先日	平成26年12月29日 (2014.12.29)		弁理士 齊藤 達也
(33) 優先権主張国・地域又は機関	米国 (US)	(72) 発明者	ソヒ, グリンダール, エス.
(31) 優先権主張番号	62/096, 962		アメリカ合衆国 53705 ウィスコン
(32) 優先日	平成26年12月26日 (2014.12.26)		シン州 マディソン, グレググ ロード
(33) 優先権主張国・地域又は機関	米国 (US)		4617

最終頁に続く

(54) 【発明の名称】 仮想アドレスを使用してアクセスされるキャッシュ

(57) 【特許請求の範囲】

【請求項 1】

物理アドレスにて格納するメモリに関して使用するための電子プロセッサコンピュータシステムであって、

プロセッサと、

メモリキャッシュと、

仮想アドレスによって物理アドレスのメモリデータをキャッシュするキャッシュ制御回路と、

前記プロセッサからの仮想アドレスと物理アドレスとの間で変換するための変換回路と

、

与えられた仮想アドレスを、前記メモリキャッシュにアクセスするために前記プロセッサから受け取るシノニム追跡回路であって、

(1) 前記与えられた仮想アドレスが、前記メモリキャッシュにおけるデータの同一の物理アドレスにマッピングしている他の仮想アドレスに関するシノニムであるかを判定し、

(2) 前記与えられた仮想アドレスがシノニムである場合、前記メモリキャッシュに関するアクセスするアドレスとして、前記他の仮想アドレスを用いて前記メモリキャッシュにアクセスする、シノニム追跡回路と、を備え、

前記シノニム追跡回路は、与えられた仮想アドレスがシノニムであるかを判定するためにシノニムとしてリンクされた仮想アドレスの第1のテーブルを備え、

10

20

前記シノニム追跡回路は、前記与えられた仮想アドレスに関連する物理アドレスを判定することによって、物理アドレス及び仮想アドレスをリンクしている第2のテーブルに前記判定した物理アドレスを適用することによって、及び、判定された物理アドレスが前記第2のテーブルにおける仮想アドレスにリンクする場合に、前記メモリキャッシュのためのアクセスするアドレスとして前記第2のテーブルの前記仮想アドレスを用いることによって、前記メモリキャッシュのキャッシュミスに対応するように構成されている、

電子プロセッサコンピュータシステム。

【請求項2】

前記メモリキャッシュは、仮想アドレスによってのみアクセス可能な物理アドレスのデータを保持する、

請求項1に記載の電子プロセッサコンピュータシステム。

【請求項3】

他の仮想アドレス各々は、多数の仮想アドレスのシノニムであるが他の鍵仮想アドレスのシノニムでない、鍵仮想アドレスである、

請求項1に記載の電子プロセッサコンピュータシステム。

【請求項4】

前記与えられた仮想アドレスが、他の仮想アドレスに関するシノニムでない場合、前記シノニム追跡回路は、前記メモリキャッシュに関するアクセスするアドレスとして、前記与えられた仮想アドレスを用いて前記メモリキャッシュにアクセスする、

請求項1に記載の電子プロセッサコンピュータシステム。

【請求項5】

前記第1のテーブルは、リンクされた仮想アドレスの組に関連するメモリアクセス権限の組を更に含み、前記他の仮想アドレスが前記アクセスするアドレスとして用いられる場合に、前記メモリキャッシュへのアクセスは、前記第1のテーブルの前記メモリアクセス権限によって制御される、

請求項4に記載の電子プロセッサコンピュータシステム。

【請求項6】

前記シノニム追跡回路は、前記第1のテーブルにおける少なくともいくつかの前記仮想アドレスの圧縮されたシグネチャを含み、前記圧縮されたシグネチャは、前記与えられた仮想アドレスが前記第1のテーブルにありそうであることを示し、前記シノニム追跡回路は、まず、前記圧縮されたシグネチャを検査し、前記与えられた仮想アドレスが前記第1のテーブルにありそうであることを圧縮されたシグネチャが示している場合にのみ、前記第1のテーブルを検査する、

請求項4に記載の電子プロセッサコンピュータシステム。

【請求項7】

前記第2のテーブルは、更に、リンクされた物理及び仮想アドレスを用いる場合に、メモリへのアクセスを制御するメモリアクセス権限と、リンクされた物理アドレス及び仮想アドレスを関連付ける、

請求項1に記載の電子プロセッサコンピュータシステム。

【請求項8】

前記メモリキャッシュの行は、対応している仮想アドレスにリンクされた前記第2のテーブルの対応しているエントリーへのインデックスを保持しており、前記シノニム追跡回路は、前記メモリキャッシュの行が追い出され又は無効化される場合に、更新のために前記第2のテーブルにおけるエントリーを識別するために前記インデックスを受け取る、

請求項1に記載の電子プロセッサコンピュータシステム。

【請求項9】

前記第2のテーブルにおける単一の鍵仮想アドレスのみが、与えられた物理アドレスにリンクする、

請求項1に記載の電子プロセッサコンピュータシステム。

【請求項10】

10

20

30

40

50

前記プロセッサは、ロードキュー及びストアキューを使用しているアウトオブオーダー・プロセッサであり、前記ロードキュー及び前記ストアキューにおけるデータは、鍵仮想アドレスにリンクされる、

請求項 9 に記載の電子プロセッサコンピュータシステム。

【請求項 11】

前記シノニム追跡回路は、与えられた仮想アドレスを物理アドレスに変換するために変換索引バッファ及びページテーブルを用いる、

請求項 1 に記載の電子プロセッサコンピュータシステム。

【請求項 12】

第 2 のテーブルは、更に、前記メモリキャッシュにおける前記物理アドレスのデータの  
前記第 2 のテーブルのサブセットの物理アドレス各々を識別してよく、前記キャッシュ制御回路は、前記メモリキャッシュに登録された前記第 2 のテーブルの物理アドレスの一部を識別するために前記第 2 のテーブルを用いてよい、

請求項 1 に記載の電子プロセッサコンピュータシステム。

【請求項 13】

前記第 2 のテーブルは、前記メモリキャッシュにおいて登録された前記物理アドレスの多数のサブセットを示している物理アドレスに関連するカウンタを実現する、

請求項 1 に記載の電子プロセッサコンピュータシステム。

【請求項 14】

メモリキャッシュ及び仮想アドレスによって物理アドレスのメモリデータをキャッシュするキャッシュ制御回路と組んでいる少なくとも 1 つのプロセッサであって、更に前記プロセッサからの仮想アドレスと物理アドレスとの間の変換のための変換回路を備えている前記プロセッサと、与えられた仮想アドレスが、前記メモリキャッシュにおけるデータの同一の物理アドレスにマッピングしている他の仮想アドレスに関するシノニムであるかを判定するために、前記与えられた仮想アドレスを前記プロセッサから受け取るシノニム追跡回路であって、前記与えられた仮想アドレスがシノニムである場合、前記メモリキャッシュに関するアクセスするアドレスとして、前記他の仮想アドレスを用いてメモリキャッシュにアクセスする前記シノニム追跡回路であり、与えられた仮想アドレスがシノニムであるかを判定するためにシノニムとしてリンクされた仮想アドレスの第 1 のテーブルを備える前記シノニム追跡回路を備えている種類の電子プロセッサコンピュータシステムにおいてデータをキャッシュする方法であって、当該方法は、

(1) 仮想アドレスを用いてメモリキャッシュにおけるデータを格納するステップであって、前記メモリキャッシュは、前記仮想アドレスの少なくとも一部を保持しているステップと、

(2) 与えられた仮想アドレスが前記メモリキャッシュにアクセスするための指示されたシノニム仮想アドレスを有するかをまず判定し、有する場合、前記メモリキャッシュにアクセスするために前記指示されたシノニム仮想アドレスを用いることによって、前記与えられた仮想アドレスを用いて前記メモリキャッシュからのデータにアクセスするステップと、

(3) 前記与えられた仮想アドレスに関連する物理アドレスを判定することによって、物理アドレス及び仮想アドレスをリンクしている第 2 のテーブルに前記判定した物理アドレスを適用することによって、及び、判定された物理アドレスが前記第 2 のテーブルにおける仮想アドレスにリンクする場合に、前記メモリキャッシュのためのアクセスするアドレスとして前記第 2 のテーブルの前記仮想アドレスを用いることによって、前記メモリキャッシュのキャッシュミスに対応するステップと、

を含む方法。

【請求項 15】

仮想アドレスを用いて前記メモリキャッシュにおけるデータを格納するステップは、仮想アドレスのみを用いて前記格納されたデータへのアクセスを許可する、

請求項 14 に記載の方法。

## 【請求項 16】

前記指示された仮想アドレスは、多数の仮想アドレスのシノニムであるが他の鍵仮想アドレスのシノニムでない、

請求項 14 に記載の方法。

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

連邦政府の支援による研究又は開発に関する宣言

## 【0002】

関連出願の相互参照

10

本願は、全体の参照によって本明細書に組み込まれる、2014年12月26日付け出願の米国仮出願第62/096,962号、及び2014年12月29日付け出願の米国仮出願第62/097,342号の利益を主張する。

## 【背景技術】

## 【0003】

発明の背景技術

本願発明は、コンピュータ構成に関連し、特に、物理アドレスというよりも仮想アドレスによって、キャッシュコンテンツのアクセスを許容するメモリキャッシュを備えている改良されたコンピュータ構成に関連する。

## 【0004】

20

キャッシュメモリは、比較的コンパクトである高速アクセスメモリ構造をプロセッサの近傍に備えることによって、メモリデータにアクセスするためにプロセッサが要する時間を最少化するために用いられる。時間的に近接するメモリアクセスは、ロード部分（参照局所性）において密集する傾向があり、従って、一度ロードされると再ロードを必要とする前に、キャッシュメモリがプロセッサによる多数のメモリアクセスを提供することを許容する期待を持って、メインメモリの一部がキャッシュメモリにロードされる。多くの場合、キャッシュの多数のレベル（例えば、L1、L2、L3）は、キャッシュ構造における高速アクセス及び固有の制限された格納量の間のトレードオフを最適化するために用いられ得る。

## 【0005】

30

キャッシュメモリのデータにアクセスするために、一般的にインデックスビットと称する、データのアドレスビットの第1の組みからのビットの組みが、キャッシュにおいてインデックスし、インデックスされたエントリーにおける行を選択するために用いられる。一般的にタグビットと称する、アドレスビットの第2の組みからのビットの組みは、この後、選択されたエントリーに対応しているタグビットの組みと比較され、アドレスビットの第2の組みからのビットの組みがタグビットの選択された組に合致する場合に、ヒットが宣言される。

## 【0006】

現代のプロセッサにおいて動作しているプログラムは、通常、メモリにおけるデータの物理アドレスとは異なる仮想アドレスを使用して、メモリにアクセス（読み込み又は書き込み）する。仮想アドレスの使用は、技術分野において知られているように、他のプロセスへの割り当てによって影響を与えられない連続メモリ領域を多数のプログラムに見せることを可能にすることによって、物理メモリがアクティブ仮想アドレスにのみ割り当てられることを可能にすることによって、及び他のプロセスの動作による1つのプロセスのメモリ破損を防止することによって、多数のプログラムの実行を大幅に容易にする。

40

## 【0007】

仮想アドレス領域の使用は、この利点に関わらず、プログラムによって用いられる仮想アドレス領域からコンピュータメモリによって要求される物理アドレス領域への変換を要することにより、メモリアクセスにおける遅延を増大させる。通常、この変換は、各仮想アドレス領域から各物理アドレス領域への相互参照を行うエントリーを備えているページ

50

テーブルによって行われる。通常、ページテーブルエントリーは、また、対応している仮想ページにおいて、データのためのアクセス権限を備え得る。ページテーブルは、アドレスを変換する及び必要なアクセス権限を検査するプロセスを高速化するために、エントリーにページテーブルからの最近アクセスされたエントリーにキャッシュを割り当てる変換索引バッファ (TLB) によって増加され得る。TLBは、高速、高関連構造を採用することによって、低いアクセスレイテンシ及び低い誤り率のために最適化されてよい。

#### 【0008】

仮想メモリを備えるシステムにおいて、キャッシュメモリは、通常、物理メモリアドレスによってアクセスされるにも関わらず、キャッシュにおけるデータにアクセスするために用いられるアドレスのビットの第1の及び第2の組み (例えば、インデックス及びタグ) は、同一の物理アドレスの両方の部分である。TLB及びページテーブルを用いるアドレス変換のレイテンシは、メインメモリのために寛容性があるが、アドレス変換によって相当低速化され得る頻繁で迅速なアクセスを行うことを意図されるキャッシュメモリで用いられるときに、より負担になる。また、高関連TLBでの変換は、エネルギー不足であらう。

#### 【0009】

理想的には、キャッシュは、プログラムから仮想アドレスを用いて直接的にアクセスされることができ、キャッシュにおけるデータにアクセスするために用いられるビットの第1及び第2の組み (例えば、インデックス及びタグ) は、同一の仮想アドレスの両方の部分である。キャッシュされたデータにアクセスするための仮想アドレスの使用は、TLB探索からの結果であるレイテンシ及びエネルギーオーバーヘッドを除去することができるが、しかしながら、同一の物理アドレスにマップされる別の仮想アドレスのグループになるシノニムの可能性によって複雑になる。このエイリアシング (又は、オーバーラッピング) は、(物理又はメイン) メモリにおけるデータである、例えば、別の仮想アドレス領域での異なるプロセスにわたっている共有情報を効率的に管理することを可能にして好適である。このような場合において、キャッシュのための仮想アドレスの使用は、多数の異なるキャッシュエントリーを同一の物理アドレス (シノニム仮想アドレス) にマッピングすることを許容することができ、これは、別の仮想アドレスにおける同一のデータを保持する。重複を許容することは、キャッシュ容量を減少させる。これは、また、重複の間の一貫性の問題を表示する。例えば、1つのプロセスが第1のキャッシュ位置を用いている共通物理アドレスに関するデータを更新する場合に、第2のプロセスは、第2のキャッシュ位置を用いている共通の物理アドレスのための最新の値を読み込むことができない。

#### 【0010】

キャッシュアクセスのために仮想アドレスを用いるキャッシュにおいて固有であるこの問題への1つの解決手段は、物理アドレス領域において仮想アドレスエイリアシング (又は、オーバーラッピング) を禁止することである。例えば、解決手段は、単一の物理アドレスからのデータがキャッシュにおける異なる仮想アドレスにてキャッシュされることを禁止することができる。この解決手段は、データへの参照の位置関係を利用するためのキャッシュの能力を著しく減少させる。同様に、解決手段は、特に、データが書き込まれる場合に、同一の物理ページからのデータがキャッシュにおける異なる仮想アドレスにてキャッシュされることを防止することができる。代替的に、解決手段は、シノニム自体の発生を除去する単一のグローバル仮想アドレス領域を採用することができる。これらの解決手段の各々は、ソフトウェアに相当な要求を設け、この実用的な効用を著しく制限する。

#### 【0011】

他の一般的に用いられる解決手段は、従来技術における仮想インデックス物理タグ (VIPT) キャッシュとして参照されるものである。ここで、エントリーを選択するべくキャッシュにインデックスするために用いられるビットの第1の組みは、仮想アドレスの一部であり、選択されたエントリーのタグビットとの比較のために用いられるビットの第2の組みは、仮想アドレスに対応している物理アドレスの一部である。解決手段は、仮想アドレスのいくつかの下位のビット (ページオフセットビット) がアドレス変換の結果として

変化しないという考えを利用する。従って、（対応している物理アドレスのものと同一である）仮想アドレスのこれらの下位のビットは、キャッシュへのインデックスのために用いられることができ、及び、対応しているエントリーに存在しているデータ及びタグビットへのアクセスを開始することができる。並列に、TLBは、物理アドレスを取得するためにアクセスされる。両方の動作が完了したときに、TLBアクセスから取得される物理アドレスビットは、キャッシュタグに格納されている物理アドレスと比較され、これらが合致する場合に、ヒットが宣言される。この手法は、TLBが並列にアクセスされるので、キャッシュアクセスのレイテンシを、従来のキャッシュアクセスよりも減少させるかもしれない。しかしながら、TLBアクセスエネルギーは、未だ消費され、従って、仮想アドレスにてアクセスされるキャッシュのエネルギーの利点は得られない。また、キャッシュをインデックスするために用いられる多数のビット（これらのビットは、アドレス変換の結果として変化すべきではない）に設けられる制限は、エネルギーの効率化を図ることを所望され得るというよりは、高い度合いの関連性を潜在的に要求している、キャッシュの構成を制限する。

10

【発明の概要】

【発明が解決しようとする課題】

【0012】

望まれることは、第1のアドレスがキャッシュにおいてインデックスするために、及び、エントリーを選択するために用いられ、及び第2のアドレスが選択されたタグビットに対して比較されるために、及び、ヒットが同一の仮想アドレスの両方の部分であることを宣言するために用いられるように、仮想アドレスにてアクセスされ得るキャッシュを備えることである。従って、キャッシュにおいてヒットするデータアクセスは、TLBにアクセスすること又は仮想から物理アドレスへ変換することを必要とせずに、仮想アドレスを用いて単独で行われ得る。このようなキャッシュは、キャッシュアクセスを行うための物理アドレスを採用した構成に対して、著しいアクセスレイテンシ及び/又はエネルギー消費の利点を有する。更に、このようなキャッシュの動作がソフトウェアに即応し、正しい動作を確実にするためのソフトウェアにおける要求が存在しないことが、また、望まれる。

20

【課題を解決するための手段】

【0013】

本願発明者は、多数の重要な計算タスクにおいて、シノニムがプログラムの実行において存在している間に、データがキャッシュに残っている時間に匹敵する短い時間内に、（1）同一の物理アドレスにマップされ相当少ないシノニムが存在し、（2）キャッシュへの相当少ないアクセスがシノニムに関するデータへのものであるということについて認識してきた。本願発明は、与えられた仮想アドレスに関して直接的にキャッシュにアクセスすることにより、及びこの後にシノニム仮想アドレスを検出することにより、及びキャッシュアクセスの前に選択された鍵仮想アドレスにシノニム仮想アドレスを効果的に変換することによりこれらの考えを利用する。キャッシュされたデータの寿命の間のシノニムの相対的な不足により、控えめなサイズのデータ構造にてシノニムを追跡することが実用的である。

30

40

【0014】

具体的には、発明は、物理アドレスにて格納するメモリに関して使用するための電子プロセッサ構成を提供する。構成は、プロセッサと、メモリキャッシュと、仮想アドレスによってメモリキャッシュにおける物理アドレスのメモリデータをキャッシュするキャッシュ制御回路と、プロセッサからの仮想アドレスと物理アドレスとの間で変換するための変換回路とを備える。加えて、構成は、与えられた仮想アドレスを、キャッシュにアクセスするためにプロセッサから受け取るシノニム追跡回路であって、1）与えられた仮想アドレスが、キャッシュにおけるデータの同一の与えられた物理メモリアドレスにマッピングしている他の仮想アドレスに関するシノニムであるかを判定し、（2）与えられた仮想メモリアドレスがシノニムである場合、メモリキャッシュに関するアクセスするアドレスと

50

して、他の仮想アドレスを用いてメモリキャッシュにアクセスする。

【0015】

例えば、発明の少なくとも1つの実施形態の特徴は、シノニムを追跡することにより、及び単一の指示された（鍵）仮想アドレスを用いて全てのキャッシュ要求を生成することにより、仮想アドレスを用いてアクセスされたキャッシュにおける仮想アドレスの可能なエリアジング又はオーバーラッピングを調節することである。

【0016】

与えられた仮想アドレスが、他の仮想アドレスに関するシノニムでない場合、シノニム追跡は、メモリキャッシュのためのアクセスするアドレスとして、与えられた仮想アドレスを用いてメモリキャッシュにアクセスしてよい。

10

【0017】

例えば、発明の少なくとも1つの実施形態の特徴は、シノニムが存在しない支配的な場合において仮想メモリアドレスを用いてキャッシュにより高速にアクセスすることを可能にすることである。

【0018】

キャッシュコントローラは、与えられた仮想アドレスが他の仮想アドレスに関するシノニムであるかを判定するシノニム追跡回路と並列に、与えられた仮想アドレスを用いてメモリキャッシュへのアクセスを準備してよい。

【0019】

例えば、発明の少なくとも1つの実施形態の特徴は、シノニムが存在しない場合に、シノニム追跡への遅延アテンダントを減少させることである。

20

【0020】

シノニム追跡回路は、与えられた仮想アドレスが、データの同一の物理メモリアドレスにマップされた他の仮想アドレスに関するシノニムであるかを判定するためのシノニムである仮想アドレスの第1のテーブル（ARTと称する）を備えてよい。

【0021】

例えば、発明の少なくとも1つの実施形態の特徴は、短い時間枠において予期される少数のアクティブシノニムと一致するシノニムを監視する単純でコンパクトな方法を提供することである。

【0022】

30

シノニム追跡回路は、与えられた仮想アドレスが第1のテーブルにありそうであることを示している第1のテーブルの圧縮されたシグネチャを備えてよく、シノニム追跡回路は、まず、圧縮されたシグネチャを検査してよく、与えられた仮想アドレスが第1のテーブルにありそうであることを圧縮されたシグネチャが示している場合にのみ、第1のテーブルを検査してよい。

【0023】

例えば、発明の少なくとも1つの実施形態の特徴は、シノニムが存在しない共通の場合において第1のテーブルの検討をショートカットして終了することである。

【0024】

シノニム追跡回路は、与えられた仮想アドレスに関連する物理アドレスを判定することによって、物理アドレス及び指示された（鍵）仮想アドレスをリンクしている第2のテーブルに判定した物理アドレスを適用することによって、及び、判定された物理アドレスが第2のテーブルにおける（鍵）仮想アドレスにリンクする場合に、メモリキャッシュのためのアクセスするアドレスとして第2のテーブルの仮想アドレスを用いることによって、メモリキャッシュのキャッシュミスに応じてよい。

40

【0025】

例えば、発明の少なくとも1つの実施形態の特徴は、また、シノニムが第1のテーブルにおける仮想アドレスによってリンクされていない場合に、共通の物理アドレスによって「隠れた」シノニムの識別を可能にすることである。

【0026】

50

シノニム追跡回路は、アクセスするアドレスを物理アドレスに変換するために変換索引バッファ及びページテーブルを用いてよい。

【0027】

例えば、発明の少なくとも1つの実施形態の特徴は、シノニム判定の目的のために現存する変換回路（例えば、TLB）を活用することである。

【0028】

第1のテーブル、第2のテーブル、及びキャッシュ行は、仮想アドレスにリンクされたメモリアクセス権限を備えてよい。

【0029】

例えば、発明の少なくとも1つの実施形態の特徴は、本発明によってより可能にされるように、ページテーブル情報を参照する必要性なしで、仮想アドレスによってアクセスされるキャッシュのためのメモリアクセス権限を取り扱うことである。

【0030】

電子プロセッサは、ロードキュー及びストアキューを用いている[アウトオブオーダー・プロセッサ](#)であってよく、ロードキュー及びストアキューは、鍵仮想アドレスにリンクされたデータを格納してよい。

【0031】

例えば、発明の少なくとも1つの実施形態の特徴は、プログラム実行における潜在的な誤りのリスクなしでロードキュー及びストアキューにおいて用いられることを仮想アドレスに可能にさせることである。キャッシュメモリの行は、対応している仮想アドレスにリンクされた第2のテーブルの対応しているエントリーへのインデックスを保持してよく、シノニム追跡回路は、キャッシュメモリの行が追い出された場合に、第2のテーブルにおけるエントリーを更新するためにこのインデックスを受け取ってよい。

【0032】

例えば、発明の少なくとも1つの実施形態の特徴は、第2のテーブルにおけるエントリーの更新を調整するために、仮想アドレスの物理アドレスへの変換のための必要性を除去することである。

【0033】

これらの特定のもの及び利点は、請求の範囲内ないであるが発明の範囲を画定しないいくつかの実施形態においてのみ適用されてよい。

【図面の簡単な説明】

【0034】

【図1】本発明毎の仮想アドレスに応じているキャッシュを採用しているコンピュータ構成のブロック図であり、キャッシュコントローラ、変換索引バッファ、及びシノニム追跡回路を採用しているメモリ管理ユニットを示している。

【図2】メモリアクセス要求に応じており仮想アドレスを有しているメモリ管理ユニットの動作のフローチャートである。

【図3】シノニム追跡回路において用いられたシノニムシグネチャの図である。

【図4】シノニム追跡回路において用いられるアドレス再マッピングテーブルの図である。

【図5】シノニム追跡回路において用いられるアクティブシノニム検出の図である。

【図6】キャッシュにおける単一のキャッシュ行エントリーの図である。

【図7a】シノニム追跡回路へのアクセス及びアドレス再マッピングテーブルがパイプラインプロセッサに分配される多数の実施形態の図である。

【図7b】シノニム追跡回路へのアクセス及びアドレス再マッピングテーブルがパイプラインプロセッサに分配される多数の実施形態の図である。

【図7c】シノニム追跡回路へのアクセス及びアドレス再マッピングテーブルがパイプラインプロセッサに分配される多数の実施形態の図である。

【図7d】シノニム追跡回路へのアクセス及びアドレス再マッピングテーブルがパイプラインプロセッサに分配される多数の実施形態の図である。



## 【発明を実施するための形態】

## 【0035】

ここで、図1を参照して、コンピュータ処理システム10は、後述の様々な要素を備えるメモリシステム16と通信する1つ以上のプロセッサシステム12、及びネットワーク、ターミナル等のような他の装置15を備え得る。

## 【0036】

各プロセッサシステム12は、メモリシステム16に関して、及び具体的にはL1キャッシュ18a及びL2キャッシュ18bのような1つ以上のキャッシュ18に関して、メモリ管理ユニット(MMU)14を介して通信するプロセッサ13を備え得る。プロセッサシステム12は、ランダムアクセスメモリディスクドライブ等を含んでいる物理メモリ20と順次通信する低レベルキャッシュ(例えば、L3キャッシュ18c)を備え得る。プロセッサ13は、条件分岐等のフロー制御動作と共に、演算及び論理動作、メモリアクセス動作を含んでいる、標準コンピュータ命令を実行する能力を提供し得る。プロセッサ13は、下記のように技術分野において一般的に理解される機能を提供する、ロードキュー17及びストアキュー19を備え得る。

## 【0037】

MMU14は、技術分野において一般的に理解される技術によるキャッシュ18のエビクト及びロードラインを備えている物理メモリ20からのキャッシュ18の更新を行うキャッシュコントローラ22を備え得る。以下にてより詳細に述べられるように、MMU14は、物理アドレスというよりも仮想アドレスによってアクセスされ得るように、キャッシュ18aにおけるデータを格納する。データは、例えば、仮想アドレスによって又は物理アドレスによってアクセスされ得るように、他のキャッシュ18b及び18cに格納され得る。これに関して、キャッシュコントローラ22は、物理アドレスPbを提供するために、仮想アドレスVbの仮想アドレス領域から、物理メモリ20の物理アドレス領域へのマッピングを判定することができる変換索引バッファ(TLB)26と通信し得る。この物理アドレスPbは、キャッシュ18においてキャッシュミスが存在する場合に、最終的に、ライン30を介して物理メモリ20にアクセスするために用いられる。後述の記載において、仮想アドレス(例えば、Vb)は、格納されたデータを独自に識別するために必要な仮想アドレス全体、及びキャッシュ18のような構造をインデックスするために用いられる仮想アドレスの部分集合の両方を参照するために用いられる。特有の使用法が、明細書から明白になる。

## 【0038】

技術分野において理解されるように、TLB26は、仮想アドレス領域及び物理アドレス領域の間の現在の変換に関連するキャッシュを提供する。特定の仮想アドレスVbのためのマッピングがTLB26において見つからない場合、キャッシュコントローラ22は、1つ以上のページテーブル28を参照して、「ページウォーク」することによって、必要な変換を取得し得る。TLB26又はページテーブル28へのアクセスは、時間及びエネルギー消費であり、望ましくは避けられる。ページテーブル28は、破線箱の20、18にラベルされることにより一般的に示された、キャッシュ18に部分的に、及び、物理メモリ20に部分的に格納されてよい。

## 【0039】

MMU14は、また、1つの実施形態において、以下にて詳細に記述されるように、アクティブシノニムシグネチャ(SS)32、アドレス再マッピングテーブル(ART)34を備える第1のテーブルと、アクティブシノニム検出テーブル(ASDT)36を備える第2のテーブルとを備え得る、シノニム回路35を備える。

## 【0040】

仮想アドレスを使用するキャッシュアクセスの概要

MMU14は、アドレスライン24を介してプロセッサ13からメモリアクセス要求を受信し、当該要求は、アクセスされるデータの仮想アドレス(Vb)を含んでいる。これらの要求は、実際に、Vbがキャッシュ18における同一のデータにアクセスするために

用いられている、他のアクティブ仮想アドレス（鍵仮想アドレス）のアクティブシノニムであるかを判定するシノニム回路 35 の S S 3 2 及び A R T 3 4 に送信され得る。用語「アクティブ」は、キャッシュ 18 における現在のデータにマップする仮想アドレスを称する。

【 0 0 4 1 】

S S 3 2 が、アクティブシノニムが存在しないことを示す場合、仮想アドレス V b は、仮想アドレス V b が鍵仮想アドレスであるという期待を持ってキャッシュ 18 に直接的にアクセスするために用いられる。

【 0 0 4 2 】

一方、S S 3 2 が、V b がアクティブシノニムであるかもしれないことを示す場合、仮想アドレス V b は、この関係を確実にして V b がシノニムであるための鍵仮想アドレス（例えば、V a）を識別するアドレス再マッピングテーブル（A R T）34 に送られる。

【 0 0 4 3 】

アクティブ鍵仮想アドレスを識別すると、V a は、メモリキャッシュ 18 にアクセスするために代わりに用いられる。

【 0 0 4 4 】

上述の手法は、V b が他のアクティブアドレス（鍵仮想アドレス）である場合、又は、V b が鍵仮想アドレスである場合を含んでいる、ほとんどの場合において、T L B 26 による処理なしで、又は、ページテーブル 28 を介してウォークする必要なしで、仮想アドレスを用いてキャッシュ 18 にプロセッサ 13 によって直接的にアクセスすることを可能にする。

【 0 0 4 5 】

メモリキャッシュ 18 にキャッシュミスが存在する場合、キャッシュコントローラ 22 は、S S 3 2 又は A R T 3 4 によって現在識別されていないが、同一の物理アドレス P x にマップされた既知のアクティブ鍵仮想アドレスに対して、与えられた仮想アドレス（V b）を比較することによって検出可能な、V b への既知でないシノニムが存在するかもしれないかを判定するために A S D T 3 6 を参照する。このような隠れたシノニムが検出される場合、必要なキャッシュ行 71 が各 A S D T 3 6 に利用可能になり、メモリキャッシュ 18 は、鍵仮想アドレス V a を用いて再びアクセスされる。他の状態では、アクセスはキャッシュ又は物理メモリ 20 の他のレベルに行われ、このアクセスが完了するときに、取得されたデータが、仮想アドレス V a にインデックスされているキャッシュ 18 に格納される。何れの場合においても、S S 3 2、A R T 3 4、及び A S D T 3 6 は、適宜更新される。

【 0 0 4 6 】

仮想キャッシュ動作の詳細の記載

図 2 を参照して、M M U 1 4 は、プロセスブロック 40 によって示されているように、メモリアクセスのための与えられた仮想アドレス V b を受信し得る。この与えられた仮想アドレス V b は、好ましい実施形態において、判定ブロック 52 毎に S S 3 2 に対して検査される。

【 0 0 4 7 】

また、図 3 を参照して、与えられた仮想アドレス V b を受信すると、S S 3 2 は、例えば、S S 3 2 におけるビットベクトル 48 の特定のビット 46 へのポインターを生成するためにハッシュコーダー 44 を用いて与えられた仮想アドレス V b をハッシュすることによって、与えられた仮想アドレス V b を処理する。V b がアクティブシノニム仮想アドレスのようである場合、ビット 46 に指し示されたものが設定される。これは、A R T 3 4 における与えられた仮想アドレス V b のもののためのエントリーであるかもしれないことを示す。インデックスしているハッシュは、極めて速いが、A R T 3 4 に関する成功の可能性を提供するのみであり、下記のように、A R T 3 4 の検討を要求するシノニムを示さない。

【 0 0 4 8 】

ＳＳ３２において、ビットベクトル４８の各ビット４６は、ＡＲＴ３４における多数の関連するエントリー（又は、ＡＲＴ３４におけるこのビット４６にマッピングしている仮想アドレス）を示しているビットカウンター５０に関連付けられてよい。カウンター５０のこのカウンター値は、もはやＡＲＴ３４における有効なシノニムデータのハッシュでないビット４６をリセットするために、ビットベクトル４８が更新されることを可能にする。カウンター５０は、また、全てのシノニムがＡＲＴ３４から追い出される前に、ハッシュコーダー４４が同一のビット４６に多数のシノニムをマップする場合に、ビット４６のリセットを防止する。

【００４９】

判定ブロック５２にて、ＳＳ３２における関連ビット４６が、Ｖｂがアクティブシノニム仮想アドレスでないことを示している場合に、キャッシュ１８は、Ｖｂを用いて矢印４１によって示されるようにプロセスブロック４２に直接的にアクセスされる。

【００５０】

判定ブロック５２にて、ＳＳ３２における関連ビット４６が、Ｖｂがシノニム仮想アドレスである可能性を示す場合、ＭＭＵ１４は、プロセスブロック５４に進め、ＡＲＴ３４は、シノニム仮想アドレスＶｂが関連する鍵仮想アドレスを識別するために用いられる。

【００５１】

ここで、図４を参照して、ＡＲＴ３４は、アドレスライン２４（図１）を介して受信した仮想アドレスが比較されるシノニム仮想アドレス５６に対応している各列として示される多数のエントリーを提供してよい。いくつかの「既知の」アクティブシノニム仮想アドレス５６（例えば、Ｖｂ）は、ＡＲＴ３４においてリストされ、ＡＲＴ３４における単一鍵仮想アドレス５８（例えば、Ｖａ）にマップする。ＡＲＴ３４は、また、特定のメモリアドレス（例えば、Ｖｂ）が読み込まれ得るか書き込まれ得るかを制御するために動作し、技術分野において知られたメモリ管理の標準であるメモリアクセス「権限」５９を保持してよい。

【００５２】

仮想アドレスＶｂをＡＲＴ３４に適用すると、ＡＲＴ３４の権限５９は、メモリアクセスが適切であることを確認するために評価される。適切でない場合、エラーが生成されて、技術分野で理解されるようにメモリアクセス違反例外として要求を取り扱う結果になる。

【００５３】

権限５９が所望されたアクセスを許容する場合、判定ブロック６１にて、ＡＲＴ３４におけるシノニム仮想アドレス５６の１つに係る、シノニムＶａへの仮想アドレスＶｂの合致が存在するか判定される。このような合致が存在する場合、対応する鍵仮想アドレス５８（例えば、Ｖａ）は、矢印５５毎にキャッシュ１８に直接的に適用される。

【００５４】

判定ブロック６１にて仮想アドレスＶｂのためにＡＲＴ３４において合致するシノニムが存在しない場合、仮想アドレスＶｂは、矢印５７毎にキャッシュ１８に直接的に提供される。

【００５５】

ここで、図６を参照して、キャッシュ１８における各キャッシュ行７１は、このキャッシュ行７１のデータ６５が関連する仮想アドレス６３からのタグビットの組みを備えてよい。これは、キャッシュ１８が、仮想的にインデックスされているが物理的にタグ付けされており、正しく機能するためにＴＬＢアクセスを要するキャッシュ構造と対称的に、仮想的にインデックスされ及び仮想的にタグ付けされている、ことである。キャッシュ行７１は、また、後述されるようなＡＳＤＴ３６のための権限データ６２及びエントリー識別インデックス値６９を備えてよい。

【００５６】

プロセスブロック４２のキャッシュアクセスプロセスを完了して、判定ブロック６０にて検出されるときに、成功したキャッシュヒットが存在し得る（キャッシュミスがない）

10

20

30

40

50

。ART34においてシノニムが発見された場合、ART34における権限59の方を選んで、キャッシュ行71における権限データ62は無視される。ART34においてシノニムが発見されなかった場合、権限データ62が、アクセスが許容されるべきか否かを判定するために採用される。アクセス権限が、この場合において所望されたアクセスを許可しない場合、エラーが生成されて、技術分野で理解されるようにメモリアクセス違反例外として要求を取り扱う結果になる。他の状態では、データ65がアクセスされて、アクセスプロセスは、プロセスブロック64によって示されているように完了する。

#### 【0057】

判定ブロック60にてキャッシュミスが検出される場合、Vb（又は、Va）にインデックスされるキャッシュ18におけるエントリーが存在しなかったことを意味しており、キャッシュコントローラ22は、プロセスブロック67に進める。プロセスブロック67にて、与えられた仮想アドレスVbは、この与えられた仮想アドレスVbに関連付けられている対応している物理アドレス（例えば、Pb）を取得するために、TLB26に適用される（及び、ページテーブル28に適用され得る）。この物理アドレスPbは、この後、「隠れた」シノニムを見つけるためにASDT36に適用される。

#### 【0058】

ここで、また、図5を参照して、ASDT36は、キャッシュ18において既知のアクティブ物理アドレス68（Px）に対応している列として示される多数のエントリーを有する論理テーブルを備えてよい。また、ASDT36の各列において、これによって物理アドレスPxにリンクされ、この物理アドレス68のための単一鍵仮想アドレス58になる。ASDT36は、また、鍵仮想アドレス58と関連するデータのためのアクセス権限を提供しているこれらの上述のものと同様な権限73を備えてよい。

#### 【0059】

ASDT36の各列は、また、各物理アドレスPxに関連するキャッシュ18における物理アドレスの特定の行を示しているビットベクトル72を備えてよい。ビットベクトル72は、従って、キャッシュ18におけるデータの識別における行レベル分解能を備える。例えば、物理アドレス68がページ（例えば、4キロバイト）であるときに、ビットベクトル72の各ビット74は、ページの対応している行（64バイト）がキャッシュ18において登録されるときに、設定される。

#### 【0060】

カウンター76は、キャッシュ18における全ての列が追い出されて、カウンターが特定の物理アドレスPxに関してゼロの値を有するときに、ASDT36の列のエントリーが無効にされ得るように、ビットベクトル72に設定されるビット74の数の合計を提供してよい。カウンター76は、従って、ASDT36の列がもはや有用でないとき、後述するASDT36の更新において、ASDT36の列を追い出すために用いられてよい有効ビットのように動作する。ASDT36は、また、後述するようなアクティブシノニム検出ビット79を備えてよい。

#### 【0061】

格納されなければならない新たな行のために利用可能なエントリーが無いとき、ASDT36の有効な行を追い出すことが必要であるかもしれない。ページ情報は変化し（例えば、物理アドレス68又は権限59にて変化し）、又は、キャッシュ18をフラッシュすることは、また、キャッシュ18における全ての対応している行が追い出されることを要求するASDT36の他の状態の有効な行の無効化をトリガーし得る。ART34及びSS32における関連するアクティブシノニムは、また、追い出されなければならない。このプロセスは、ASDT36に関して上述のアクティブシノニム検出ビット79の使用を通して支援されてよい。代替的な実施形態において、ART/SSは、この時に完全にフラッシュされてよい。

#### 【0062】

判定ブロック78にて、プロセスブロック67からの物理アドレスPbが、ASDT36に適用されるときに、キャッシュ18におけるデータに関連するアクティブ物理アドレ

10

20

30

40

50

ス 6 8 の 1 つに合致しない場合、新たなデータはキャッシュ 1 8 に付加される必要があり、キャッシュ 1 8 は、下記に更に記載されるように、A S D T 3 6 と共に、プロセスブロック 8 0 毎に更新される。

【 0 0 6 3 】

他方、判定ブロック 7 8 にて、物理アドレス P b が A S D T 3 6 にて見つけれられる場合、判定ブロック 8 4 にて、与えられた仮想アドレス V b が A S D T 3 6 における鍵仮想アドレス 5 8 と同一であり P b に関連しているかについての判定が行われる。

【 0 0 6 4 】

判定ブロック 8 4 にて、与えられた仮想アドレス V b が A S D T 3 6 にある場合、キャッシュミスがキャッシュ 1 8 内からの物理アドレス P b の適切な行の失敗によって生じることが想定されなければならない。これは、キャッシュ 1 8 に付加される新たなデータを要求し、キャッシュ 1 8 は、プロセスブロック 8 0 毎に A S D T 3 6 と共に更新される。

【 0 0 6 5 】

他方、判定ブロック 8 4 にて、与えられた仮想アドレス V b が A S D T 3 6 に無い場合、A S D T 3 6 内（例えば、V e ）にある鍵仮想アドレスへのシノニムとしての V b の役割は、プロセスブロック 8 5 にて A R T 3 4 及び S S 3 2 を適切に更新することによって記憶される。この時に、A S D T 3 6 におけるアクティブシノニム検出ビット 7 9 は、設定され得る。A S D T 3 6 は、この後、与えられた仮想アドレス V b によって要求される必要なキャッシュ行 7 1 がメモリキャッシュ 1 8 にあるかどうかを確認するために（例えば、ビットベクトル 7 2 の対応ビット 7 4 が A S D T 3 6 に設定されているか否かを検査することによって）検査される。キャッシュ行 7 1 がメモリキャッシュ 1 8 にある場合、ライン 8 9 によって示されているように、キャッシュ 1 8 へのアクセスが、A S D T 3 6 において発見された鍵仮想アドレス V e を用いてプロセスブロック 4 2 によって行われて得る。

【 0 0 6 6 】

判定ブロック 8 8 にて、必要とされる行が A S D T 3 6 に無い場合、キャッシュ 1 8 及び A S D T 3 6 は、プロセスブロック 8 0 毎に更新される。

【 0 0 6 7 】

プロセスブロック 8 0 において、キャッシュコントローラ 2 2 は、キャッシュ 1 8 を更新することを許容され、この更新が成功した場合、A S D T 3 6 は適切に更新される。A S D T 3 6 がいくつかの仮想アドレスの下での所望のデータを有さないとき（判定ブロック 7 8 での「N O」）、この更新は、第 1 の欄における物理アドレス P b 及びこの更新に関連する（ここでは、鍵仮想アドレス 5 8 ）第 2 の欄における与えられた仮想アドレス V b を A S D T 3 6 に付加し、ビットベクトル 7 2 の適切なビット 7 4 が設定され、カウンタ 7 6 がインクリメントされる。適切なビット 7 4 は、与えられた仮想アドレス V b から導き出されるオフセット値から推測され得る。A S D T 3 6 が与えられた物理アドレス（P b）のページのためのエントリーを有しているが必要なデータ行を有していないとき（判定ブロック 8 4 にて「Y E S」）、キャッシュ 1 8 にて新たに登録されるためのビットベクトル 7 2 の適切なビット 7 4 が設定され、カウンタ 7 6 がインクリメントされる。A S D T 3 6 が与えられた物理アドレス（P b）のページのためのエントリーを有するが異なる仮想アドレス（V e）にリンクされており必要なデータ行を有していないとき（判定ブロック 8 8 にて「N O」）、キャッシュ 1 8 に新たに登録されるデータのビットベクトル 7 2 の適切なビット 7 4 が設定され、カウンタ 7 6 がインクリメントされ、A R T 3 4 及び S S 3 2 はこの新たなシノニムを反映するために適切に調整される。

【 0 0 6 8 】

A S D T 3 6 の犠牲エントリーは、A S D T 3 6 の無効列（すなわち、ゼロに等価となっているカウンタ 7 6 を有している）、あるいは、非ゼロのカウンタ値である列と関連する物理アドレス 6 8 を選択するために A S D T 3 6 を見ることによって判定されてよい。前者の場合、ゼロのカウンタを有している無効列は、キャッシュ 1 8 において未だ依存する有効列が無いことを意図している。後者の場合、追い出される犠牲エントリーの

10

20

30

40

50

判定は、技術分野にて知られた多数の多種の手法のいくつかを用いて実行され得る。非ゼロのカウンター値であるエントリーがASDTから追い出されるために選択されるとき、（対応しているビットベクトル72のビット74において非ゼロの値によって示され得る）キャッシュに未だ存在しているエントリーによって追跡される列は、まずキャッシュ18から追い出されて、これによって、カウンタ値をゼロに下げて、そしてこの後にASDTエントリーが問題なしで追い出されることを可能にする。ASDTにおけるエントリーが追い出されるとき、ART及びSSの関連したエントリーは、また、適宜無効化及び/又は更新される必要がある。

#### 【0069】

概ね、キャッシュ18は、標準キャッシュと実質的に同様であるが、しかしながら、仮想アドレスによってアクセスされる。仮想アドレスは、同じ問題にアドレスするために技術分野で概ね理解されるように、アドレス領域識別子（ASID）と組み合わせられてよい。

#### 【0070】

ここで図1、2、5及び6を参照して、キャッシュ18が更新されてキャッシュ行71が追い出されなければならないときに、追い出される行に対応しているASDT36のエントリーは、対応しているビットベクトル72のビット74が更新されて対応しているカウンタ76がディクリメントされるように、識別されなければならない。ASDT36における対応している行を識別することは、ASDTが物理アドレス68を介してたいてい平易にアクセス可能であるが、追い出される行が仮想アドレス58によって識別されるという事実によって複雑にされ得る。従って、キャッシュ行71は、ASDT36における必要なエントリーの迅速な識別が更新されることを可能にするASDTインデックス値69を備えてよい。1つの実施形態において、ASDTインデックス値69は、また、キャッシュ18に設けられるためにキャッシュミスにて戻されたキャッシュ行に対応しているASDT36エントリーが更新され得るように、標準コンピュータ構成において生じるキャッシュミスを取り扱うために用いられる誤り状態取り扱いレジスタ（MSHR）に格納されてよい。

#### 【0071】

##### 他のプロセッサに関するキャッシュコヒーレンス

図1を参照して、他のプロセッサシステム12は、与えられたプロセッサシステム12aのキャッシュにおけるキャッシュ行71を無効化するこれらのキャッシュにおけるデータに書き込んでよい。これは、物理アドレスを用いることを無効化されるデータを概ね記述している無効化メッセージを生成する。識別のために必要なキャッシュ行71を識別する1つの方法は、鍵仮想アドレス58を取得するために、及び、必要なキャッシュ行71を識別するためにこの鍵仮想アドレス58を用いるために、無効化メッセージの物理アドレスをASDT36に適用することである。無効化メッセージに加えて他の種類のコヒーレンスイベントが存在する実施形態において、ASDT36を調べるための、及び、必要なキャッシュ行71を識別するための同様な方法が採用され得る。本願発明は、L1キャッシュ18aにおける最大の使用を概ね見つけ、L2及びL1キャッシュの間のコヒーレンスイベントはまれであり、このオーバーヘッドを最小化する。

#### 【0072】

##### ロード及びストアキュー

ここで図1を参照して、ロードキュー17及びストアキュー19は、メモリの読み書きを管理するために、アウトオブオーダー・プロセッサにおいて用いられる。ストアキュー19のために仮想アドレスを用いることは、後のロードがシノニムによって合致しているストアを識別しない場合及びシノニムによって合致しているストアを識別する場合に問題を発生し得る。このような場合、古いデータがキャッシュ18に戻され得る。同様に、ロードキューのために仮想アドレスを用いることは、コヒーレンスに基づくキャッシュ18における無効化又は追い出しがあり、ロードがこの追い出されたデータのために（推測で）行われたときに、問題が発生し得る。本発明は、鍵仮想アドレス58によって及びシノニ

10

20

30

40

50

ムによらず、ロードキュー 17 及びストアキュー 19 におけるデータを識別することによってこの問題に対応し得る。ロード又はストアが（推測で）実行されると、鍵仮想アドレス 58 は、識別されたロード又はストアキューに保持される。（鍵仮想アドレス 58 でなく）シノニムにてロードキュー 17 及びストアキュー 19 を用いてアクセスすることは、従って、キャッシュ 18 におけるキャッシュミスを引き起こし、最終的にプロセスブロック 54 の ART 34 探索を介して（又は、プロセスブロック 84 の ASDT 36 探索を介して）鍵仮想アドレス 58 での動作（ロード又はストア）の再実行を行う結果になる。

【0073】

ストアキュー 19 のストアのための TLB 誤りは、TLB 誤りが解決されるまで、分離キューにおけるより若いストアを保持することによって適切に解決され得る。代替的に、ストアキューからのストア命令の解放を効果的に遅延させるために、違反ストアからプログラムを再起動する。

【0074】

#### 大きなページサイズ

大きなページサイズのために、ASDT 36 のビットベクトル 72 は、扱い難くなり得る。この場合、ビットベクトル 72 を除去してよく、ASDT 36 から追い出される所望の行に関する与えられたページから行を探すために、キャッシュ 18 の行を介して「ウォーク」の代わりにしてよい。この高価な動作は、カウンタ 76 の非ゼロの値に関するエントリを追いつくことが必要である場合に特に、大きなページよりも小さな関連する ASDT 36 行から好適に追いつくことにより避けられ得る。概ね、大きなページの追いつきは、また、ありそうもないイベントである。

【0075】

#### SS 及び ART のアクセス

ここで、図 7a - 7d を参照して、パイプラインプロセッサにおける SS 32 及び ART 34 のアクセスは、様々な手法で行われ得る。図 6a において示されるように、SS 32 及び ART 34 へのアクセスは、アドレス生成の後であるが曖昧性解消の前に行われ得る。この手法は、仮想キャッシュを用いる電力効率の利点の大部分を達成するが、潜在的なレイテンシ利点を十分に奏するわけではない。

【0076】

図 6b において示されているように、代替的に、SS 32 及び ART 34 は、曖昧性解消ステージの間に並列的にアクセスされ得る。曖昧性解消は、まれであるが、ART 34 においてヒットする鍵仮想アドレス 58 に関して行われる必要があり得る。これは、電力消費を増大させるかもしれないが、レイテンシを減少させ得る。

【0077】

図 6c において示されているように、SS 32 へのアクセスは、ベース（又は、セグメント）レジスタに基づいてアドレス生成の前に行われることができ、ART 34 へのアクセスは、アドレス生成の後であり曖昧性解消の前に行われ得る。この手法は、命令及びデータキャッシュの両方に適用することができ、仮想キャッシュで可能なエネルギー及びレイテンシの利点の大部分を享受することができる。

【0078】

代替的に、図 6d において示されているように、アドレス生成が多数ステッププロセスである場合、SS 32 は、アドレス生成が完了する前であるが、アドレス生成の中間ステップの後にアクセスされ得る。

【0079】

#### 最適化

ART 34 へのアクセスは、連続する参照が（特に命令のために）同じページに頻繁に行われるという事実を享受することによって減少され得る。1つの実施形態において、（図 4 において図示されている）最終 LVA（LLVA）レジスタ 100 は、アクセスされた最後の鍵仮想アドレス 58 を維持する ART 34 に関連付けられ得る。関連付けされている次の仮想アドレスが LLVA 100 に含まれているものと同じであるときに、ART

10

20

30

40

50

34へのアクセスは行われる必要がなく、LLVA100におけるこの値は、直接的に用いられることができ、図2のプロセสブロック54をスキップする。

【0080】

各プロセスは、ユーザ及びカーネル領域を含む異なる仮想アドレスメモリ領域に関連付けられてよい。カーネル領域は、異なるプロセスにわたって共有され、従って、カーネル領域へのアクセスは、カーネル領域におけるアクセスされる仮想アドレスが同じであるが、異なるプロセスが異なるASIDを有するので、シノニムとして考えられ得る。これは、カーネル領域が多数のプロセスによって最新の手法においてアクセスされるときに、ART34における多数エントリを生成し得る。想定されるオーバーヘッドを避けるために、1つの実施形態は、カーネル領域へのアクセスのためのみの単一の唯一な値に、ASIDのランタイム再マッピングを用いてよい。このようなアクセスの識別は、アクセスがカーネルアクセスに関連付けられたアドレス範囲を有するという先験的な知識に基づいている。例えば、カーネル領域（又は、動作システム）がアドレス領域の上半分に設けられる場合、この再マッピングは、再マッピングプロセスをトリガーするための最上位アドレスを単純に参照することができる。

10

【0081】

上述の全ての実施形態に関して、上述した上記の構成は、キャッシュデータ転送に必要な高速性に一致した高速性能を実現するためにハードウェアにて全体的に実現されることができる。

【0082】

20

上述はL1キャッシュに焦点をあてたが、これらの原理は、メモリアクセスレイテンシを減少させる利点を確認し得るときに、他のキャッシュに広げられ得ることが理解される。

【0083】

この発明は、ART34、ASDT36及びキャッシュ行71を含んでいる各データ格納構造は、各仮想アドレス領域を独自に識別し、従って、効果的に仮想アドレスの一部になるアドレス領域識別子（ASID）と一緒に仮想アドレスを格納し得ることを期待する。

【0084】

1つの実施形態において、SS32及びART34は、シノニムVbでのキャッシュ18への直接のアクセスの方を選んで、及びシノニムVbが鍵仮想アドレスでない場合に、キャッシュミスの結果になり、適切な鍵シノニムはプロセスブロック60毎のキャッシュミスの後のASDT36へのアクセスによって解決される期待の方を選んで、除去されてよい。

30

【0085】

シノニム追跡回路35によって採用されるデータ及びシノニム追跡回路35自体を提供しているSS32、ART34、及びASDT36のデータ格納機能はコンピュータプロセッサシステム10の回路、又は、内部プロセッサ回路の中で様々な手法にて分配されてよいということについて、上述から理解される。特に、SS32、ART34、及びASDT36のデータは、キャッシュ18自身を含んでいる様々な異なる場所に自由に格納され得る。後者の場合、物理キャッシュは、常用キャッシュ構成及び分離クレーム要素の対象になり得るシノニム追跡構成を有することについて理解されたい。

40

【0086】

特定の用語は、参照のみの目的で本明細書において使用され、したがって限定するものではない。例えば、「上部の」、「下部の」、「上方に」、「下方に」などの用語は、参照される図面中の方向を指す。「正面の」、「背部の」、「後部の」、「底部の」および「側面の」などの用語は、論じている構成要素を説明する本文および関連図面を参照することによって明らかとなる、一貫しているが任意の基準枠内の構成要素の部分の向きを説明している。そのような用語は、具体的に上で言及した単語、その派生語および同様の輸入語を含んでもよい。同様に、「第1の」、「第2の」という用語、および構造を指す他の

50



そのような数の用語は、文脈で明確に示さない限り、シーケンスまたは順序を示唆しない。

【0087】

本開示および例示的实施形態の要素または特徴を導入する場合、「1つの」および「前記」という冠詞は、そのような要素または特徴が1つまたは複数存在することを意味するものである。「備える」、「含む」および「有する」という用語は、包括的なものであり、具体的に記述するもの以外の追加の要素または特徴が存在し得ることを意味する。本明細書で説明した方法ステップ、工程、および操作は、実行順序として具体的に特定しない限り、議論または説明した特定の順序でそれらの実行を必ず要求するものとして解釈されるべきではないことをさらに理解されるべきである。また、追加または代替のステップを用いてもよいことも理解されるべきである。

10

【0088】

「プロセッサ」への参照は、単一の及び/又は分散環境において通信することができ、従って、有線又は無線通信を介して他のプロセッサと通信するように構成されることができ、1つ以上のマイクロプロセッサを含み、このような1つ以上のプロセッサは、同様な又は異なる装置であり得る他のプロセッサ制御される装置を動作させるように構成され得る、ことについて理解されたい。プロセッサは、一般目的プロセッサ、処理コア、マルチスレッドプロセッサの内容、グラフィクス処理ユニット、特別目的プロセッサ、あるいは、技術分野で知られているようにメモリにアクセスする動作を実行する他の形式のプロセッサであり得る、ことについても理解されたい。また、メモリへの参照は、他に特定されない場合、プロセッサ制御される装置の内部に設けられることができ、プロセッサ制御される装置の外部に設けられることができ、有線又は無線ネットワークを介してアクセスされることができ、1つ以上のプロセッサ読み取り可能及びアクセス可能メモリ要素及び/又は構成を含むことができる。

20

【0089】

加えて、テーブル及びシグネチャを含んでいる、示されたデータ構造は、論理的な描写を提供することを意図しており、開示された発明は、データの他の物理構成を含みまた示された機能を提供すると考えられる、ことについて理解される。

【0090】

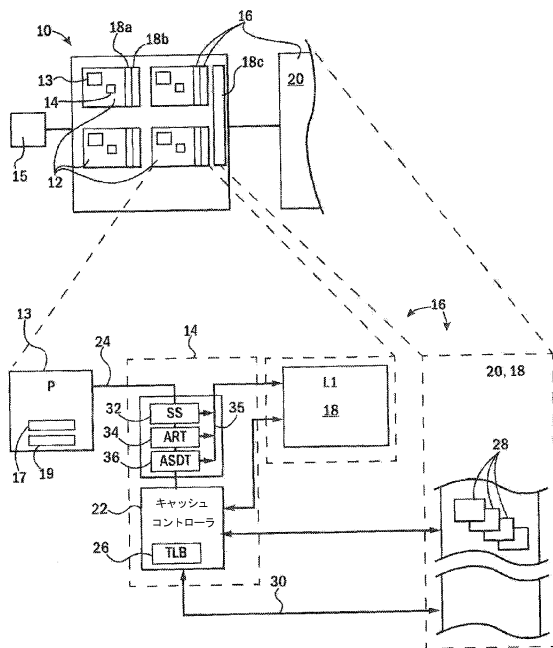
本明細書で用いられる用語「インデックス」、「インデックスされる」等は、本のインデックスのように、値に関連する情報の場所を示し及びアクセスするための値を使用するプロセスを概ね示し、キャッシュメモリの内容におけるインデックスの技術的意味に限定することを意図しているわけではないことについて、理解されたい。

30

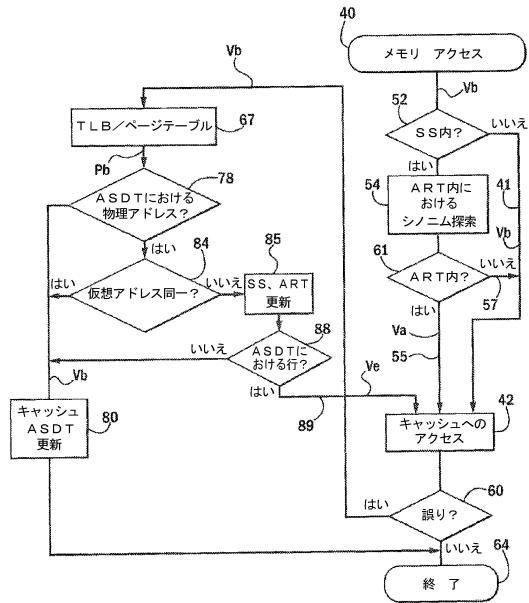
【0091】

本発明は、本明細書に含まれる実施形態および例示に限定されないことが特に意図され、特許請求の範囲は、以下の特許請求の範囲内にあるような実施形態の一部を含む修正形式の実施形態、および異なる実施形態の要素の組合せを含むことを理解されたい。特許および非特許公報を含む、本明細書で説明した刊行物のすべては、それら全内容が本明細書に援用される。

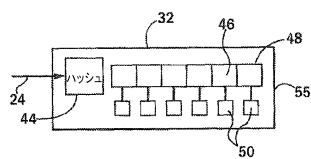
【図 1】



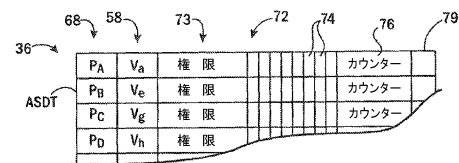
【図 2】



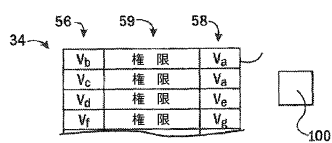
【図 3】



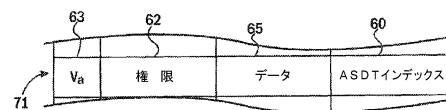
【図 5】



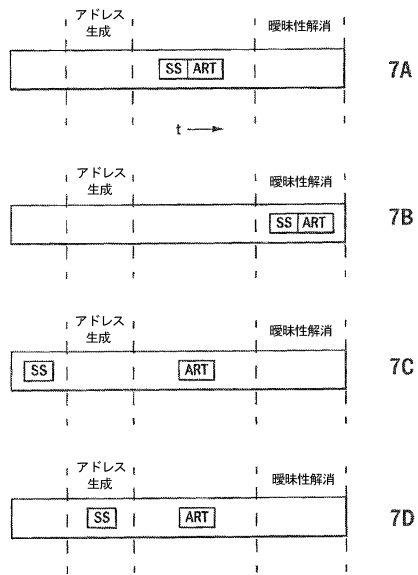
【図 4】



【図 6】



【図 7】



## フロントページの続き

(31)優先権主張番号 14/867,926

(32)優先日 平成27年9月28日(2015.9.28)

(33)優先権主張国・地域又は機関  
米国(US)

(72)発明者 ヨーン, ホンジル

アメリカ合衆国 53705 ウィスコンシン州 マディソン, シボイガン アヴェニュー 48  
59, ナンバー 315

審査官 酒井 恭信

(56)参考文献 特開2004-192615(JP, A)

米国特許出願公開第2004/0117587(US, A1)

国際公開第2013/109696(WO, A1)

米国特許出願公開第2013/0185520(US, A1)

特開平11-024992(JP, A)

米国特許出願公開第2004/0117591(US, A1)

米国特許出願公開第2005/0055528(US, A1)

特開平06-083711(JP, A)

米国特許出願公開第2007/0101044(US, A1)

(58)調査した分野(Int.Cl., DB名)

G06F 12/08 - 12/0897

G06F 12/10 - 12/109