

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2006-39937

(P2006-39937A)

(43) 公開日 平成18年2月9日(2006.2.9)

(51) Int. Cl.	F I	テーマコード (参考)
<b>G06F 12/08 (2006.01)</b>	G06F 12/08 519E	5B005
	G06F 12/08 501B	
	G06F 12/08 503F	
	G06F 12/08 507Z	

審査請求 未請求 請求項の数 10 O L (全 13 頁)

(21) 出願番号	特願2004-218801 (P2004-218801)	(71) 出願人	000005223 富士通株式会社 神奈川県川崎市中原区上小田中4丁目1番1号
(22) 出願日	平成16年7月27日 (2004.7.27)	(74) 代理人	100101856 弁理士 赤澤 日出夫
		(74) 代理人	100097250 弁理士 石戸 久子
		(72) 発明者	清水野 光憲 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	小島 広行 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)	5B005 JJ01 JJ12 KK12 MM05 MM22 NN12 NN45 NN72 PP03 UU32

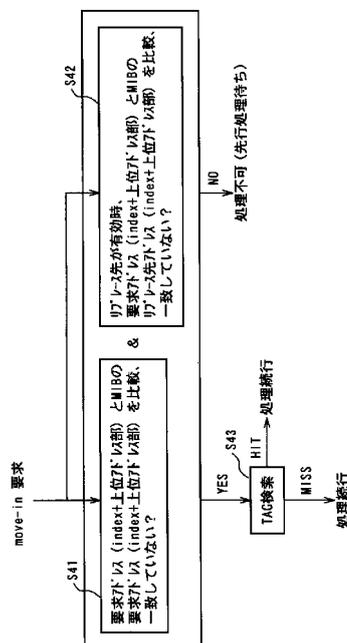
(54) 【発明の名称】 キャッシュメモリ制御回路、キャッシュメモリ制御方法

(57) 【要約】

【課題】 リプレース対象がクリーンな場合、無効化しない方式のCPUのキャッシュメモリにおいて、処理が可能な後続ムーブイン要求が先行ムーブイン要求により待機させられるケースを削減させ処理効率を向上させる。

【解決手段】 MIB内に、ムーブイン要求に対するリプレースアドレスの上位アドレス部情報を持たせると共に、リプレース先の有効性の有無についての情報を持たせ、前記要求のアドレスとMIB内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第1判断部(ステップS41)と、ムーブイン要求のアドレスとMIB内のリプレースアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第3判断部(ステップS42)とを備え、これらの判断結果が共に肯定的であり、且つリプレース先が有効である場合に、ムーブイン要求に対する処理を続行するタグ検索部(ステップS43)とを備える。

【選択図】 図4



## 【特許請求の範囲】

## 【請求項 1】

キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御回路であって、

ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第 1 判断部と、

前記第 1 判断部による判断が肯定的であった場合に、前記ムーブイン要求に対する処理を続行する第 1 処理部と

を備えるキャッシュメモリ制御回路。

10

## 【請求項 2】

請求項 1 に記載のキャッシュメモリ制御回路において、

前記第 1 判断部による判断が否定的であった場合に、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とするキャッシュメモリ制御回路。

## 【請求項 3】

請求項 1 又は請求項 2 に記載のキャッシュメモリ制御回路において、

前記第 1 処理部は、

前記ムーブイン要求に対するタグ検索を行うタグ検索部と、

前記タグ検索部により、キャッシュヒットが判断された場合に、前記ムーブイン要求のアドレスにおけるインデックス及びキャッシュヒットしたキャッシュにおけるウェイと、ムーブインバッファ内に登録されている先行ムーブイン要求のアドレスにおけるインデックス及びキャッシュのリプレースウェイとが共に一致していない場合を判断する第 2 判断部とを備え、

20

前記第 2 判断部による判断が肯定的であった場合に前記ムーブイン要求に対する処理を続行することを特徴とするキャッシュメモリ制御回路。

## 【請求項 4】

請求項 3 に記載のキャッシュメモリ制御回路において、

前記第 2 判断部による判断が否定的であった場合は、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とするキャッシュメモリ制御回路。

30

## 【請求項 5】

キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御回路であって、

MIB内に、ムーブイン要求に対するリプレースアドレスの上位アドレス部の情報を持たせると共に、リプレース先の有効性の有無についての情報を持たせ、

前記ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び前記上位アドレス部が共に一致していない場合を判断する第 1 判断部と、

前記ムーブイン要求のアドレスとMIB内に登録されたりプレースアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第 3 判断部とを備え、

40

前記第 1 判断部と前記第 3 判断部による、判断結果が共に肯定的であり、且つ前記リプレース先が有効である場合に、前記ムーブイン要求に対する処理を続行する第 2 処理部とを備えるキャッシュメモリ制御回路。

## 【請求項 6】

キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御方法であって、

ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要

50

求のアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第1判断ステップと、

前記第1判断ステップによる判断が肯定的であった場合に、前記ムーブイン要求に対する処理を続行する第1処理ステップと  
を備えるキャッシュメモリ制御方法。

【請求項7】

請求項6に記載のキャッシュメモリ制御方法において、

前記第1判断ステップによる判断が否定的であった場合に、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とするキャッシュメモリ制御方法。

【請求項8】

請求項6又は請求項7に記載のキャッシュメモリ制御方法において、

前記第1処理ステップは、

前記ムーブイン要求に対するタグ検索を行うタグ検索ステップと、

前記タグ検索ステップにより、キャッシュヒットが判断された場合に、前記ムーブイン要求のアドレスにおけるインデックス及びキャッシュヒットしたキャッシュにおけるウェイと、ムーブインバッファ内に登録されている先行ムーブイン要求のアドレスにおけるインデックス及びキャッシュのリプレースウェイとが共に一致していない場合を判断する第2判断ステップとを備え、

前記第2判断ステップによる判断が肯定的であった場合に前記ムーブイン要求に対する処理を続行することを特徴とするキャッシュメモリ制御方法。

【請求項9】

請求項8に記載のキャッシュメモリ制御回路において、

前記第2判断ステップによる判断が否定的であった場合は、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とするキャッシュメモリ制御方法。

【請求項10】

キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御方法であって、

MIB内に、ムーブイン要求に対するリプレースアドレスの上位アドレス部の情報を持たせると共に、リプレース先の有効性の有無についての情報を持たせ、

前記ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び前記上位アドレス部が共に一致していない場合を判断すると共に、前記ムーブイン要求のアドレスとMIB内に登録されたリプレースアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第3判断ステップとを備え、

前記第3判断ステップによる、判断結果が肯定的であり、且つ前記リプレース先が有効である場合に、前記ムーブイン要求に対する処理を続行する第2処理ステップと

を備えるキャッシュメモリ制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、CPUのキャッシュメモリを制御するキャッシュメモリ制御回路及びキャッシュメモリ制御方法に関する。

【背景技術】

【0002】

図8はCPU全体の概略構成を示すブロック回路図である。図8ではCPU1とCPU2のそれぞれが、L1キャッシュ(ファーストキャッシュ)とL2キャッシュ(セカンドキャッシュ)の2階層のキャッシュを内蔵してSC(システムコントローラ)3へ接続され、SC3がメモリ(MEMORY)4に接続されている構成を示している。この構成におけるキャッシュメモリ制御動作は次のように行われる。

10

20

30

40

50

## 【0003】

L1キャッシュからのムーブイン要求がL2キャッシュのパイプラインを流れ、タグ検索部（タグ部）にて要求データのタグが検索された結果、L2キャッシュのキャッシュミス（L2キャッシュミス）であれば（S1）、MIB（ムーブインバッファ）を獲得しSC3へムーブイン要求が発行される（S2）。SC3は主記憶または他系CPUからデータを応答し、L2キャッシュへの登録とMIBの開放を行うとともに（S3）、L1キャッシュへのデータ応答を行う（S4）。

## 【0004】

図9は上述の動作を行うキャッシュ制御部を示すブロック図である。図9に示すMIポート（MI-PORT）11はL1キャッシュからの新規リクエストを格納し、MOポート（MO-PORT）12はL1キャッシュからL2キャッシュへの応答やL1キャッシュからのムーブアウトデータを格納し、SCポート（SC-PORT）13はSC3からのオーダーを格納する内部資源である。

## 【0005】

プライオリティ部（PRIORITY）14では、各資源の優位性を考慮してコマンドをパイプライン（PIPE LINE）15へと投入する。パイプライン15では投入されたコマンドのアドレスでタグ検索部（タグ部）16によりタグ（TAG）の検索を行う。L2キャッシュにヒットすればL1キャッシュに要求されたデータが返され、キャッシュミスの時はMIB17が獲得されてSC3にムーブイン要求が発行される。この時、同時にMIB3に登録された先行するムーブイン要求との情報比較がなされ処理が続行可能かどうかの判断が行われる。

## 【0006】

一般的にムーブイン要求（後続）と先行する先行ムーブイン要求のインデックス（index）が一致した場合、リプレースブロックの保護、キャッシュコヒーレンシの保証の為に後続のムーブイン要求は、先行ムーブイン要求に対する処理が完了し、MIB17が開放されるまで待たされる。この時の処理のフローを図10に示す。図10によれば、後続のムーブイン要求のインデックスとMIBに登録されている先行ムーブイン要求とのインデックスが一致していない場合が判断され（ステップS11）、判断結果が肯定的でない場合（ステップS11、NO）は後続の処理は待機させられる。

## 【0007】

一方、結果が肯定的であった場合（ステップS11、YES）は、処理は続行され、タグ検索が行われて（ステップS12）その検索結果に応じて（ステップS12、ヒット、ミス）それぞれの処理が行われる。

## 【0008】

この場合、後続のムーブイン要求がキャッシュヒットして実際には処理が可能な要求であった場合（処理を行っても不具合が生じない場合）でも、MIB17が開放されるまでその処理は待たされることとなる。なお、後続のムーブイン要求と先行のムーブイン要求のインデックスが一致しない場合、タグ検索を行った結果に基づいて、処理が行われる。

## 【0009】

タグ検索部（タグ部）16におけるタグの検索方法を図11に示す。タグの検索はまず要求アドレス100内のインデックス（index）110で一致をとり、いずれのブロックに該当するかを確定し、その後、上位アドレス部120の一致をとることで、いずれの64バイト（BYTE）かの特定を行う。MIB17の構成を図12に示す。MIBは要求アドレス100、登録ウェイ及びリプレースウェイ200、そしてその他のフラグ300で構成されている。

## 【0010】

MIポート11からのムーブイン要求に対するTAG検索部16の検索結果がキャッシュミスであった場合、MIB17を獲得してムーブイン要求に係るデータを格納し、SC3へとムーブイン要求を発行する。この時、すれ違いによるスヌープ処理に対して正しく応答する為に、リプレース対象を全てリプレースバッファ（MODQ）へと退避させる方式もあるが、リプレース対象がclean victim（上書き、変更されておらず、ライトバックの必要がない）である場合は、MODQ資源を無駄に獲得することとなる。そこで、リプレース対象がclea

10

20

30

40

50

n victimであるならリプレースバッファ (MODQ) へと退避させず、キャッシュを無効化しないことで、すれ違いのスヌープ処理に応答する方式を採用している。

【0011】

前述の先行するムーブイン要求であって、MIB17に格納されたムーブイン要求に係るインデックスと後続のムーブイン要求に係るインデックスのインデックスマッチを行う場合、キャッシュヒットしていても待機させられる後続が多数存在してしまう。そこで、処理続行可能なものは処理できるようにする制御回路や方法が要望される。

【0012】

このような場合において、先行ムーブイン要求がキャッシュミスを起こしMIB17を獲得時、MIBポート11から更なる後続のムーブイン要求が発行されると、先行のリプレース対象がclean victimである場合ヒットしてしまう。これを解決する為に従来技術では以下の様にしていた。

【0013】

まず、リプレース対象のキャッシュ状態を細かく分けてみる。キャッシュミスが発生時、リプレース対象のブロックがdirty victim (上書き、変更され、ライトバックの義務があるブロック) である場合、タグを無効化しMODQへとデータの書き込みを行う。リプレースデータはシステムからの要求によりMODQから読み込まれメモリへとライトバックされる。

【0014】

dirty victimはタグが無効化されてもすれ違いによるスヌープ処理に対して正しく応答する必要があるため、MIB17獲得時にWBA (リプレースアドレス) も保持し、キャッシュヒットしなくてもMIB17のWBAとスヌープ要求が一致する場合はリプレースデータのあるMODQからデータが応答される。

【0015】

一方、clean victim (ライトバックの義務がなく、かつ、無効でないブロック) はメモリへのライトバックの必要がないため、新規のキャッシュの登録によって上書きされる。この場合、システム側でムーブイン要求を受け取って、キャッシュタグコピーに新規登録が行われた時点 (clean victimが消えた時点) で、以後の処理ではclean victimがないものとして扱われる。ところがキャッシュ側ではそのタイミングを知ることができないので、キャッシュミスをした時点ではキャッシュ内のclean victimの使用は不可能である。

【0016】

また、ストア要求が共有状態のキャッシュにヒットすると排他権を求めてシステムにブロックタイプチェンジ (ブロックタイプの変更許可) を発行する。ストアの際には他で同一ブロックを持ってはいけないので、この時先行ストア要求と同一ブロックの共有キャッシュに後続要求がヒットしてもキャッシュコヒーレンシを保証する為にストア実行まで後続要求を待機させなければならない。

【0017】

このようなシステムにおいて、MIB17とインデックスマッチする際でも、処理を継続して何ら不具合が生じない場合は、その処理を継続できるように、キャッシュヒットした際のキャッシュ状態によってムーブインが可能であるかを判断する。このフローを図13に示す。キャッシュミス検出時点でdirty victimの場合は無効化されているはずなので、dirtyなブロックにヒットした場合 (ステップS21) はMIBの先行リクエストと全く異なるウェイを対象としている。よって処理が可能であるので、この場合 (ステップS21、YES) のみMIBのインデックスが一致した後続の処理続行を可能としている。cleanなブロックの場合は上記の理由でタグの状態からは判断できないので処理ができない。

【0018】

なお、従来技術として、下記特許文献が知られている (特許文献1参照)。

【特許文献1】特開2002-229852号公報

【発明の開示】

【発明が解決しようとする課題】

10

20

30

40

50

## 【0019】

cleanなブロックにキャッシュヒットした場合、それが先行のclean victim、あるいはブロックタイプチェンジのケースであれば処理が不可能であるが、それ以外のcleanなブロックなら処理を行っても不具合は何ら生じない。ところが従来の技術では、タグから判断するものであるので、そのようなケースを判断することができず、後続の処理は一括してMIB開放まで待たされざるを得ない。

## 【0020】

また、タグの検索結果に基づいて判断するため、その読み出しに時間を要し、一時的に別の処理の起動や資源の確保を行うこととなつて、他のリクエストの処理を妨げてしまうという問題点もある。

10

## 【0021】

本発明は、上述した問題点を解決するためになされたものであり、MIBとインデックスマッチする場合において、後続のムーブイン要求がcleanなブロックにヒットした場合に、後続の処理の続行を行っても差し支えないケースを判断可能とすることにより、処理可能な後続処理が一律に待機させられるケースを軽減し、もって処理の効率化を図ることができるとともに、その後続要求に対する処理においてキャッシュヒットの有無の判断時間を短縮することができ、もって、後続処理が継続できないケースにおいても、その判断を迅速に行えることができるキャッシュメモリ制御回路及びキャッシュメモリ制御方法を提供することを目的としている。

## 【課題を解決するための手段】

20

## 【0022】

上述した課題を解決するため、本発明は、キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御回路であつて、ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第1判断部と、前記第1判断部による判断が肯定的であつた場合に、前記ムーブイン要求に対する処理を続行する第1処理部とを備えるものである。

## 【0023】

また、本発明は、上記キャッシュメモリ制御回路において、前記第1判断部による判断が否定的であつた場合に、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とする。

30

## 【0024】

また、本発明は、上記キャッシュメモリ制御回路において、前記第1処理部は、前記ムーブイン要求に対するタグ検索を行うタグ検索部と、前記タグ検索部により、キャッシュヒットが判断された場合に、前記ムーブイン要求のアドレスにおけるインデックス及びキャッシュヒットしたキャッシュにおけるウェイト、ムーブインバッファ内に登録されている先行ムーブイン要求のアドレスにおけるインデックス及びキャッシュのリプレースウェイトとが共に一致していない場合を判断する第2判断部とを備え、前記第2判断部による判断が肯定的であつた場合に前記ムーブイン要求に対する処理を続行することを特徴とする。

40

## 【0025】

また、本発明は、上記キャッシュメモリ制御回路において、前記第2判断部による判断が否定的であつた場合は、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とする。

## 【0026】

また、本発明は、キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御回路であつて、MIB内に、ムーブイン要求に対するリプレースアドレスの上位アドレス部の情報を持たせると共に、リプレース先の有

50

効性の有無についての情報を持たせ、前記ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び前記上位アドレス部が共に一致していない場合を判断する第1判断部と、前記ムーブイン要求のアドレスとMIB内に登録されたリプレースアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第3判断部とを備え、前記第1判断部と前記第3判断部による、判断結果が共に肯定的であり、且つ前記リプレース先が有効である場合に、前記ムーブイン要求に対する処理を続行する第2処理部とを備えるものである。

**【0027】**

また、本発明は、キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御方法であって、ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第1判断ステップと、前記第1判断ステップによる判断が肯定的であった場合に、前記ムーブイン要求に対する処理を続行する第1処理ステップとを備えるものである。

10

**【0028】**

また、本発明は、上記キャッシュメモリ制御方法において、前記第1判断ステップによる判断が否定的であった場合に、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とする。

**【0029】**

また、本発明は、上記キャッシュメモリ制御方法において、前記第1処理ステップは、前記ムーブイン要求に対するタグ検索を行うタグ検索ステップと、前記タグ検索ステップにより、キャッシュヒットが判断された場合に、前記ムーブイン要求のアドレスにおけるインデックス及びキャッシュヒットしたキャッシュにおけるウェイと、ムーブインバッファ内に登録されている先行ムーブイン要求のアドレスにおけるインデックス及びキャッシュのリプレースウェイとが共に一致していない場合を判断する第2判断ステップとを備え、前記第2判断ステップによる判断が肯定的であった場合に前記ムーブイン要求に対する処理を続行することを特徴とする。

20

**【0030】**

また、本発明は、上記キャッシュメモリ制御方法において、前記第2判断ステップによる判断が否定的であった場合は、前記ムーブイン要求に対する処理を待ち状態にすることを特徴とする。

30

**【0031】**

また、本発明は、キャッシュメモリに対するムーブイン要求に対して先行する先行ムーブイン要求がキャッシュミスを起こしムーブインバッファを獲得している場合にキャッシュメモリの制御を行うキャッシュメモリ制御方法であって、MIB内に、ムーブイン要求に対するリプレースアドレスの上位アドレス部の情報を持たせると共に、リプレース先の有効性の有無についての情報を持たせ、前記ムーブイン要求のアドレスとムーブインバッファ内に登録されている先行ムーブイン要求のアドレスとにおけるインデックス及び前記上位アドレス部が共に一致していない場合を判断すると共に、前記ムーブイン要求のアドレスとMIB内に登録されたリプレースアドレスとにおけるインデックス及び上位アドレス部が共に一致していない場合を判断する第3判断ステップとを備え、前記第3判断ステップによる、判断結果が肯定的であり、且つ前記リプレース先が有効である場合に、前記ムーブイン要求に対する処理を続行する第2処理ステップとを備えるものである。

40

**【発明の効果】****【0032】**

本発明によれば、従来技術では先行処理が終わるのを待って処理されていた、実際には処理が可能なcleanブロックへキャッシュヒットした後続要求処理を、何ら物量を増やすことなく、継続とすることができ、もって、後続の要求に対する処理の効率化を図ることができる。また、ムーブイン要求時の最初の段階で異なるウェイが対象であるか否かを判

50

断できるので、従来技術のようにキャッシュヒットまで待たずとも処理が不可能なケースが判明する。それによりムーブイン要求発生時から処理継続不可能な場合が判断されるまでの間に、無駄な処理の起動、資源の獲得が行われることを軽減させることができる。

【発明を実施するための最良の形態】

【0033】

以下、本発明の実施の形態について図面を参照しつつ説明する。

【0034】

上述したように、ムーブイン要求（後続ムーブイン要求）に係る処理対象となるウェイが先行ムーブインのそれと異なるものであれば、後続ムーブイン要求に係る処理を続行しても何等不具合が生じることはなく、その処理は続行可能である。従来ではこれをキャッシュヒット時にキャッシュ状態を確認することで行っていた。本発明では、この処理が可能かどうかの判断を先行の要求アドレスのキャッシュ状況でなくウェイの状況を見ることで行うようにしたものである。以下説明する。

10

【0035】

図1は本発明の実施の形態におけるキャッシュメモリ制御回路を示すブロック図である。この回路においては、L2キャッシュのタグ検索部16による検索結果、キャッシュヒットした場合は、L2キャッシュのデータ部18からデータがL1キャッシュに応答される。一方、キャッシュミスした場合は、L2キャッシュのMIB17が獲得され、SC3にムーブイン要求が出される。ここで、本制御回路のマッチ回路21がL1キャッシュとL2キャッシュのMIB17との間に設けらる。マッチ回路21では、それらからの後述する所定の情報を比較し、マッチングする所定の場合にアポートをパイプラインに出力する。アポートが出力されない場合は後続のムーブイン要求に係る処理が続行される。

20

実施の形態1

以下、上記構成における、本発明の実施の形態1について説明する。図2は実施の形態1の動作を示すフローチャートであり、その具体的回路構成を図3に示す。

【0036】

まず、L1キャッシュからのムーブイン要求に係る要求アドレスとMIBに登録されている先行ムーブイン要求に係る要求アドレスとにおけるインデックスと上位アドレス部が夫々一致していない場合を判断し（ステップS31）、判断結果が否定的である（一致していない）場合（ステップS31、NO）は、ムーブイン要求に係るウェイが一致することとなるので、処理を待機させるべくアポート情報を出力し、先行ムーブイン要求において獲得されたMIBが開放されるまで処理待ち状態となる。

30

【0037】

一方、判断結果が肯定的である（一致していない）場合（ステップS31、YES）は、処理を続行するべくタグ検索を行い（ステップS32）、タグ検索の結果、キャッシュミスの場合はムーブイン要求をSCに出力する。

【0038】

タグ検索の結果、キャッシュヒットした場合は、そのヒットウェイを獲得し、ムーブイン要求に係る要求アドレスのインデックスとヒットウェイとがMIBに登録されているインデックスとキャッシュへのムーブインウェイとの夫々にマッチ回路において比較され（ステップS33：一致しない場合が判断され）、一致する場合（結果が否定的である場合）はやはり先行ムーブイン要求において獲得されたMIBが開放されるまで処理待ち状態となる。

40

【0039】

上記比較において、一致しないと判断された場合（結果が肯定的である場合）は、処理が続行され、L2キャッシュのデータがL1キャッシュに応答される。

【0040】

図3における回路図において、第1一致回路41は、その第1入力端子にムーブイン要求に係る要求アドレスのインデックス及びその上位アドレス部に関する情報を入力し、その第2入力端子にMIBに登録されている先行ムーブイン要求に係る要求アドレスにおける

50

インデックスとその上位アドレス部に関する情報を入力する。そして、第1一致回路41は、ムーブイン要求に係る要求アドレスとMIBに登録されている先行ムーブイン要求に係る要求アドレスとにおけるインデックスと上位アドレス部が夫々一致しているか否かを判断し（一致していない場合を判断し）、一致している場合（結果が否定的である場合）に「1」を出力する。

**【0041】**

第2一致回路42は、その第1入力端子にムーブイン要求に係る要求アドレスのインデックス及びそのヒットウェイに関する情報を入力し、その第2入力端子にMIBに登録されている先行ムーブイン要求に係る要求アドレスにおけるインデックスとそのリプレースウェイに関する情報を入力する。そして、第2一致回路42は、ムーブイン要求に係る要求アドレスのインデックスとヒットウェイがMIBに登録されている先行ムーブイン要求に係る要求アドレスのインデックスとリプレースアドレスに夫々一致しているか否かを判断し（一致していない場合を判断し）、一致している場合（結果が否定的である場合）に「1」を出力する。

10

**【0042】**

そして、いずれかに該当する場合にオア回路43からアポートが出力され、処理が待機される。

**【0043】**

実施の形態1においては、従来では一括して処理を待機させていたcleanなブロックについても、処理が可能なものは全て処理できる。但し、この場合、キャッシュヒット時に処理続行が可能かどうかはヒットウェイを獲得するまで分からない。この間、無駄な資源の確保や処理の起動を行ってしまう。そこで、次の実施の形態2が構成される。

20

実施の形態2

図4は実施の形態2における動作を示すフローチャートであり、図5はその場合のマッチ回路を示す図である。実施の形態2では、後続要求のアドレスと先行する要求のアドレスの比較の時点で、後続の要求アドレスと先行のリプレースアドレスとの比較を同時に行う。このため実施の形態2においては、図6に示されるように、MIBの構成において要求アドレス100、登録ウェイ及びリプレースウェイ200、その他のフラグ300に加えてリプレースアドレスの上位アドレス部領域400を加える。さらに、例えば、このリプレースアドレスが保持される際にリプレース先の状態が有効であるかのフラグ（dirty victimとclean victimのOR）を持たせるようにする。

30

**【0044】**

図4の動作について説明すると、まず要求アドレスとMIBに登録された先行する要求アドレスのインデックスと上位アドレス部が夫々一致しない場合を判断するステップ（ステップS41）と共に、リプレース先が有効である場合に要求アドレスとMIBのリプレース先アドレスのインデックスと上位アドレス部を夫々一致しない場合を判断するステップ（ステップS42）を実行し、双方の判断結果が肯定的である場合（共に一致しない場合（ステップS41及びステップS42, YES））に処理を続行し、タグ検索を行う（ステップS43）。少なくともいずれかにおいて一致があった場合（ステップS41又はステップS42, NO）はアポートを出力し、処理を待機させる。

40

**【0045】**

図5においては、第3一致回路51と第4一致回路52それぞれの第1端子にムーブイン要求に係る要求アドレスのインデックスと上位アドレス部に関する情報が入力され、第3一致回路51の第2端子にMIBの要求アドレスにおけるインデックスと上位アドレス部に関する情報が入力され、第4一致回路52の第2端子にMIBのリプレースアドレスについてのインデックスと上位アドレス部に関する情報が入力される。また、OR回路53において、リプレース先の状態が有効であるかのフラグ（dirty victimとclean victimのOR）を出力し、AND回路54において、第4一致回路52とOR回路53の論理積（AND）をとって、その出力と第3一致回路51との論理和をとり、その出力によりアポートを出力するように構成する。なお、OR回路55の二入力の同期を図るためにバッファ回

50

路 5 6 がアンド回路 5 4 に対応して設けられている。

【 0 0 4 6 】

ステップ S 4 1 , 4 2 における判断はそれぞれ、

(a)、アドレス 6 4 バイトが一致したか。

【 0 0 4 7 】

(b)、リプレースアドレス 6 4 バイトが一致し、かつリプレース先の状態が有効であるかについての判断となる。

【 0 0 4 8 】

いずれかの条件が成立する場合、後続ムーブインは先行と同じウェイを対象としていることになるのでアポートを通知する。

【 0 0 4 9 】

ステップ S 4 1、ステップ S 4 2 のいずれの条件も成立しない場合、後続ムーブインは先行と異なるウェイを対象としている。このケースは後続の処理が可能であるので処理の続行を許可する。同時にタグ検索においてキャッシュがヒットしたかの判定を行い、ヒットした場合は処理を行う。

【 0 0 5 0 】

上述したように、実施の形態 2 では、最初に要求アドレスと MIB の要求アドレスとの比較をとる時点でリプレース先のアドレスとの比較も同時に行っているため、これによりキャッシュヒット後にヒットウェイと MIB のリプレースウェイの比較を行う必要がなくなる。

【 0 0 5 1 】

どちらかに一致した場合、先行要求と後続要求のウェイが一致することになるので、処理ができないことがこの時点で判明する。どちらにも一致しなかった場合、ウェイが異なることが明らかとなるのでタグ検索の結果を待ち、キャッシュヒットすれば即処理が可能となる。

【 0 0 5 2 】

これにより、従来技術と比較して、より多くのケースを救済する事ができ、しかも、より早い段階で処理続行の可否が判明する為、無駄な資源の確保や処理の起動を最小限に抑えることができる。

【 0 0 5 3 】

また、図 6 に示されるように、実施の形態 2 では、MIB 内にリプレースアドレス 4 0 0 と要求アドレス 1 0 0 を持つこととなるが、インデックス部は共通するので、実際にはインデックス部、要求アドレスのインデックス部以外、リプレースアドレスのインデックス部以外のアドレス指定領域で済む。更にシステムによっては、リプレース中のスヌープ処理を正しく行うために元々リプレースアドレスを保持するものもあるので、その場合更に物量のインパクトは非常に小さいものとなる。

【 0 0 5 4 】

以上の動作をタイムチャートで示すと図 7 のように示される。従来技術における動作を示す図 1 4 と比較すると、先行する L2 リクエスト (REQ-1) に対して、後続の L2 リクエスト (REQ-2) の処理が極めて迅速的に且つ効率的に行われることが明白である。

【 図面の簡単な説明 】

【 0 0 5 5 】

【 図 1 】 図 1 は本発明の実施の形態におけるキャッシュメモリ制御回路を示すブロック図である。

【 図 2 】 実施の形態 1 の動作を示すフローチャートである。

【 図 3 】 実施の形態 1 のマッチ回路の具体的構成を示す回路図である。

【 図 4 】 実施の形態 2 における動作を示すフローチャートである。

【 図 5 】 実施の形態 2 のマッチ回路の具体的構成を示す回路図である。

【 図 6 】 実施の形態 2 の MIB 構成を示す図である。

【 図 7 】 実施の形態 2 の動作を概念的に示すタイムチャートである。

10

20

30

40

50

【図8】CPU全体の概略構成を示すブロック回路図である。

【図9】一般的なキャッシュメモリ制御部を示すブロック図である。

【図10】従来のキャッシュメモリ制御方法その1を示すフローチャートである。

【図11】従来のタグ検索方法を示す概念図である。

【図12】従来のMIBの構成を示す図である。

【図13】従来のキャッシュメモリ制御方法その2を示すフローチャートである。

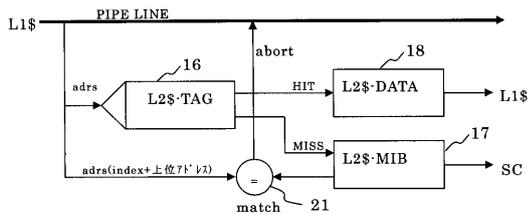
【図14】従来のキャッシュメモリ制御動作を概念的に示すタイムチャートである。

【符号の説明】

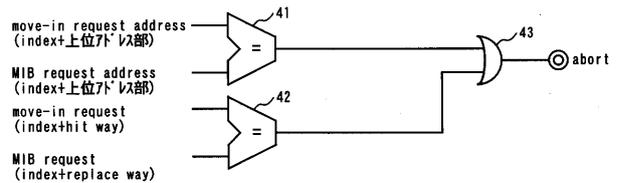
【0056】

1, 2 CPU、3 システムコントローラ(SC)、4 メモリ、16 タグ検索部(タグ部)、17 MIB(ムーブインバッファ)、18 L2キャッシュデータ部、21 マッチ回路、41 第1一致回路、42 第2一致回路、51 第3一致回路、52 第4一致回路、53 OR回路、54 アンド回路、55 OR回路、56 バッファ回路。

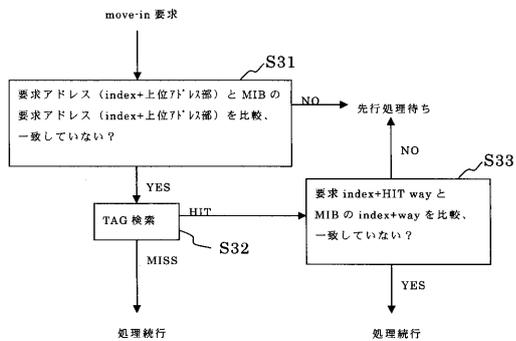
【図1】



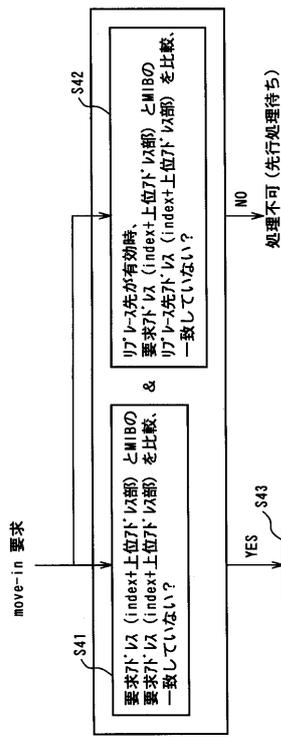
【図3】



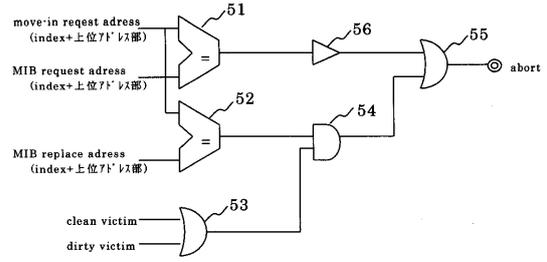
【図2】



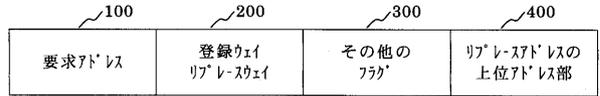
【 図 4 】



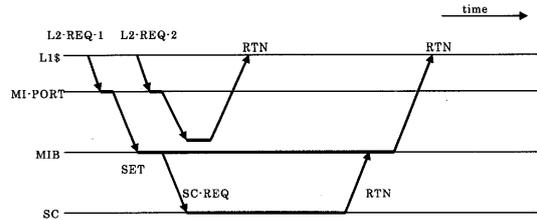
【 図 5 】



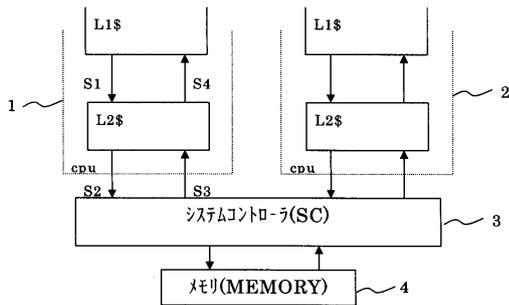
【 図 6 】



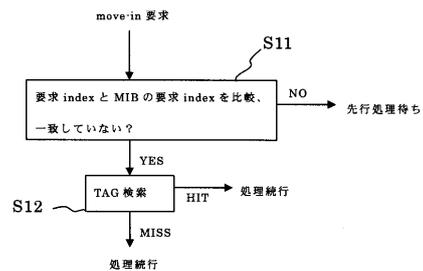
【 図 7 】



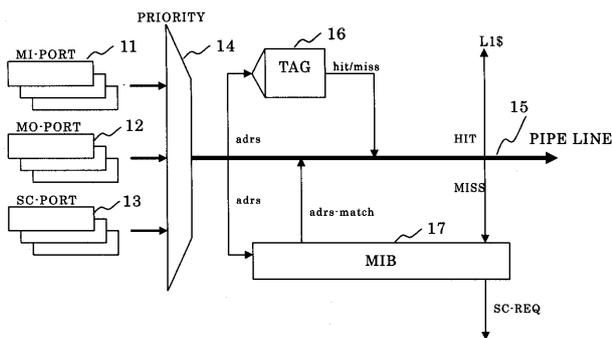
【 図 8 】



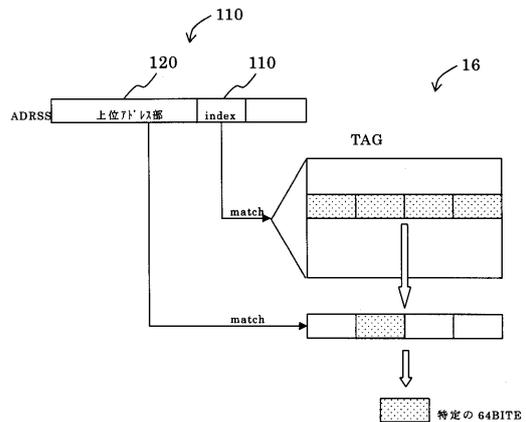
【 図 10 】



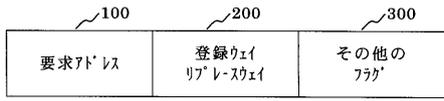
【 図 9 】



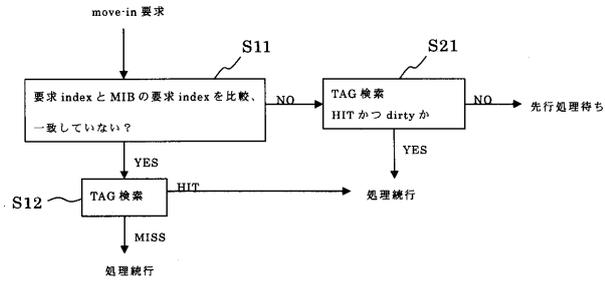
【 図 11 】



【図12】



【図13】



【図14】

