

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4137641号
(P4137641)

(45) 発行日 平成20年8月20日(2008.8.20)

(24) 登録日 平成20年6月13日(2008.6.13)

(51) Int.Cl. F I
G 0 6 F 12/08 (2006.01)
 G 0 6 F 12/08 5 0 7 J
 G 0 6 F 12/08 5 1 1 E
 G 0 6 F 12/08 5 7 9

請求項の数 8 (全 10 頁)

(21) 出願番号	特願2002-572007 (P2002-572007)	(73) 特許権者	590000248
(86) (22) 出願日	平成14年2月28日(2002.2.28)		コーニンクレッカ フィリップス エレク
(65) 公表番号	特表2004-519776 (P2004-519776A)		トロニクス エヌ ヴィ
(43) 公表日	平成16年7月2日(2004.7.2)		オランダ国 5621 ベーアー アイン
(86) 国際出願番号	PCT/IB2002/000609		ドーフエン フルーネヴァウツウェッハ
(87) 国際公開番号	W02002/073415		1
(87) 国際公開日	平成14年9月19日(2002.9.19)	(74) 代理人	100075812
審査請求日	平成17年2月28日(2005.2.28)		弁理士 吉武 賢次
(31) 優先権主張番号	09/805,384	(74) 代理人	100088889
(32) 優先日	平成13年3月13日(2001.3.13)		弁理士 橋谷 英俊
(33) 優先権主張国	米国 (US)	(74) 代理人	100082991
			弁理士 佐藤 泰和
		(74) 代理人	100096921
			弁理士 吉元 弘

最終頁に続く

(54) 【発明の名称】 命令ベースレジスタに基づくキャッシュウェイ予測

(57) 【特許請求の範囲】

【請求項 1】

それぞれに複数の格納データ列(125)が格納される複数のセクション(Way-a, b, . . . n)を含むキャッシュメモリ(120)を備えるキャッシュシステム(200)であって、

アドレス指定されたデータ項目が前記複数のセクションのうち予測されたセクション(p)にある前記複数の格納データ列(125)の対応するデータ列に置かれているかどうかを判定するものであり、

前記予測されたセクション(p)は、間接アドレス指定命令(160)のサブセットである、前記アドレス指定されたデータ項目のアドレス(110)のベースとなるベースアドレスを含む、ベースレジスタの識別子(162)のみを索引として、所定のテーブルを調べることにより定められるものであることを特徴とする、キャッシュシステム(200)。

【請求項 2】

前記キャッシュメモリ(120)はNウェイアソシアティブキャッシュに相当し、前記Nウェイアソシアティブキャッシュの各ウェイは、前記複数のセクションの一つを形成し、前記予測されたセクション(p)は予測されたウェイであり、

各ウェイは、前記複数の格納データ列(125)に対応する複数の格納タグ(121)をさらに備え、

前記複数の格納データ列(125)と前記複数の格納タグ(121)のそれぞれは、前

記アドレス指定されたデータ項目の前記アドレス(110)のインデックスフィールド(112)を介して、一義的にアドレス可能であり、

前記Nウェイアソシアティブキャッシュは、前記アドレス(110)のタグフィールド(111)と、前記予測されたウェイ内の、前記アドレス(110)の前記インデックスフィールド(112)に相当する、前記複数の格納タグ(121)の一つとを比較して、前記アドレス指定されたデータ項目が前記予測されたウェイに置かれているかどうかを判定することを特徴とする、請求項1に記載のキャッシュシステム(200)。

【請求項3】

前記Nウェイアソシアティブキャッシュは、前記アドレス指定されたデータ項目が前記予測されたウェイの対応する格納データ列(125)に置かれていない場合、前記アドレス(110)の前記タグフィールド(111)と、前記複数のセクションの他のウェイにある、格納タグ(121)とを比較し、前記アドレス指定されたデータ項目が前記他のウェイの対応する格納データ列(125)に置かれているかどうかを判定するように構成されていることを特徴とする、請求項2に記載のキャッシュシステム(200)。

【請求項4】

前記Nウェイアソシアティブキャッシュは、前記アドレス指定されたデータ項目が、前記予測されたウェイ及び前記他のウェイの対応する格納データ列(125)のいずれにも置かれていない場合、前記アドレス指定されたデータ項目をメモリから、前記複数のセクションの中の選択ウェイにロードすることを特徴とする、請求項3に記載のキャッシュシステム(200)。

【請求項5】

前記予測されたセクション(p)は、最小使用頻度判定に基づいていることを特徴とする、請求項1に記載のキャッシュシステム(200)。

【請求項6】

メモリ内で参照されるデータ項目を間接参照する少なくとも一つの間接アドレス指定命令を含むプログラム命令を受け取るように構成される命令レジスタ(160)と、

前記命令レジスタ(160)に含まれるビットのサブセットに基づいて、参照されるデータ項目を含むと思われる予測されたセクション(p)を、キャッシュ(200)内の複数のセクション(Way-a, b, . . . n)から特定するよう構成される、予測装置(250)と、を備え、

前記予測されたセクション(p)は、前記命令レジスタ(160)に含まれる前記ビットのサブセットである、前記メモリ内の前記参照されるデータ項目のアドレス(110)を判定するのに用いられるベースアドレスを含む、ベースレジスタの識別子(162)のみを索引として、所定のテーブルを調べることにより定められるものであることを特徴とする、プロセッサ。

【請求項7】

前記キャッシュ(200)の前記複数のセクションは、Nウェイアソシアティブキャッシュの複数のウェイに相当し、

前記予測されたセクション(p)は、前記複数のウェイの中の予測されたウェイに相当することを特徴とする、請求項6に記載のプロセッサ。

【請求項8】

参照されたデータ項目を含むと思われるキャッシュ(200)内のセクションの予測方法であって、

前記参照されたデータ項目を間接的にアドレス指定する間接アドレス指定命令(160)から、前記参照されたデータ項目のアドレス(110)を判定するのに用いられるベースレジスタの識別子(162)を抽出する工程と、

前記ベースレジスタの識別子(162)のみを索引として、所定のテーブルを調べることにより、前記参照されたデータ項目を含むと思われる前記キャッシュ(200)内の前記セクションを判定(250)する工程と、を有することを特徴とする、予測方法。

【発明の詳細な説明】

10

20

30

40

50

【 0 0 0 1 】

発明の背景

1. 発明の技術分野

本発明はコンピュータシステムの分野に関し、特にアドレス指定されたデータ項目を含むと思われるキャッシュ内の特定のウェイの予測装置として、命令ベースレジスタを用いる、予測型Nウェイアソシアティブキャッシュに関する。

【 0 0 0 2 】

2. 関連技術の記載

一般的にキャッシュシステムは、比較的低速のメモリ装置へのアクセスに関連しておこる有効遅延を低減するために用いられる。そのような低速のメモリ内にある、特定のデータ項目へのアクセスをプロセッサが要求する場合、キャッシュシステムは要求されたデータ項目をより高速のメモリにロードする。その後、同じデータ項目に対するアクセスは高速のメモリ経路で行われ、これにより低速のメモリに関連する遅延を回避する。一般的に、データ項目の要求があった場合は、要求されたデータ項目を含むデータ項目の「列」が低速のメモリから高速のメモリへとロードされ、高速メモリは、ロードされた列の全てのデータ項目を順次提供することができるようになる。

10

【 0 0 0 3 】

キャッシュメモリアクセスシステムの有効性は、将来的になされるデータアクセスは、過去になされたデータアクセスに関連するという可能性に基づいている。一般的に、要求されたデータ項目が、キャッシュにおいて過去に要求されたデータ項目と同じ列に含まれている可能性は、実質的にゼロ以上であり、よって、高速のキャッシュメモリからの要求を満たす可能性はこれに対応して実質的にゼロより大きい。

20

【 0 0 0 4 】

高速メモリのコストは低速メモリのコストより高く、よって一般的に、利用可能なキャッシュメモリの量は限られる。高速メモリに新しいデータの列をロードする必要がある時、どのデータ項目を高速メモリから削除するのかを決定するために、キャッシュ管理スキームが用いられる。高速メモリ内のデータ項目を維持するために一般的に用いられる優先順位スキームには、「最小使用頻度」(Least Recently Used = LRU)条件があり、これは最近の使用頻度が最も低い(つまり、「古めの」)メモリアクセスの列を新しい列に置き換えることにより、最近使われ、アクセスされているデータ項目を保持するものである。他の条件、例えば「最大使用頻度」を、通常はLRU優先順位スキームと併用して、使うことも可能である。

30

【 0 0 0 5 】

要求された項目のアドレスのサブセットに基づいてデータ項目の列を格納するために、一般的にアソシアティブキャッシュが用いられる。図1は、アソシアティブキャッシュ100に対する従来のアドレス指定スキームを示す。一般的にはプロセッサから出力されるアドレス110(以下でさらに記載する)は、タグフィールド111、インデックスフィールド112、およびワードフィールド113に論理的に区分けされる。インデックスフィールド112は、キャッシュ120内の関連するキャッシュ列のセットに索引を与える。このセットのそれぞれのキャッシュ列を「ウェイ」と呼び、キャッシュ100はNウェイアソシアティブキャッシュに相当する。ワードフィールド113のサイズjは、データ列のサイズ2^jに対応する。よって、もしデータ列あたり16ワードあれば、ワードフィールド113のサイズは4ビットである。データ列あたり64ワードあれば、ワードフィールド113は6ビット幅である。この、ワードフィールド113とデータ列のサイズとの間の、2の累乗の関係を用いて、タグフィールドとインデックスフィールドはメモリ中のデータ列を一義的に特定する。

40

【 0 0 0 6 】

低速メモリ(図示せず)からキャッシュ120にアドレス指定されたデータがロードされると、そのデータ項目を含むデータ列が選択ウェイに置かれ、インデックスフィールドは選択ウェイの中でデータ列を置く場所を定める。ウェイの選択は、上述したLRU優先順

50

位スキームのような、一般的に用いることのできる種々のアルゴリズムの一つを用いて行われる。アドレス指定されたデータ項目がキャッシュ120内の特定の列領域D L i n e - a、D L i n e - b等に格納されるとき、図1にT a g - a、T a g - b等のフィールドで表されるように、タグフィールド111もまた格納される。格納されたタグフィールドは、データ列のインデックスフィールドに対応する、ウェイ内でのデータ列の場所との組み合わせにより、一義的にキャッシュ120内に格納されているデータ列を特定する。

【0007】

アドレス指定されたデータ項目がキャッシュ120にロードされる前に、キャッシュ120はそのデータ項目がすでに自己の中に置かれているかどうかをチェックし、低速メモリからデータ項目をロードする可能性を回避する。アドレス指定されたデータは、過去にアクセスされたことによって、あるいはそのデータ項目と同じデータ列(D L i n e - a、D L i n e - b等)にあるデータ項目が過去にアクセスされたことによって、キャッシュ内に置かれている可能性がある。インデックスフィールド112は、このアドレスに関連するキャッシュ内のN列のセットを特定する。関連するセット内に格納されている125a、125b等の各列に対応する、121a、121b等の格納タグはそれぞれ、コンパレータ130a、130b等を介して、アドレス指定されたデータ項目のタグフィールド111と比較される。比較が行われている間、インデックスフィールド112に対応する格納データ列125a、125b等は、高速バッファ140にロードされ、データ項目が現在キャッシュにロードされている場合に利用できるようになる。

【0008】

アドレス指定されたデータ項目が現在キャッシュ内にロードされている場合、対応するコンパレータ130a、130b等は、キャッシュヒット信号をアサートにし、これによりデータ列を含む特定のウェイH i t - a、H i t - bなどを割り出す。ヒットがアサートされると、適当なワードが対応するバッファ140から取り出される。適当なワード141a、141b等をバッファ140に収容されているデータ列から選択するために、ワードフィールド113が用いられる。取り出されたワードはアドレス110を与えたプロセッサに渡される。従来例のキャッシュシステム100においては、タグフィールド111と格納タグフィールド120a、120b等とを比較し、キャッシュヒットがあったとき、続いて適当なワード141a、141b等を選択するのに必要な時間は、低速メモリに関する遅延時間より実質的に少ない。このように、データ項目がキャッシュ120内にあるとき、そのデータ項目にアクセスする実効アクセス時間は、実質的に低減されている。

【0009】

キャッシュヒットがなかった場合、主としてデータ列を使用頻度が最小の(L R U)ウェイや、上述した他の優先順位スキームにロードすることにより、アドレス指定されたデータ項目をメモリからキャッシュ120の選択ウェイW a y - a、W a y - b等にロードする上述の作業が行われる。

【0010】

ワードを有効にプロセッサからメモリに格納するのに必要な時間も、キャッシュ120を経由することにより同様に短縮される。キャッシュ120内にアドレス指定されたデータ項目が存在することが、上述の比較プロセスを用いて判断される。データ項目が現在キャッシュ内にある場合、プロセッサからのデータ項目の新しい値がバッファ140の選択ワードに取って代わり、バッファ140はアドレス指定されたデータ項目を含むデータ列125a、125b等にロードされる。「修正された」フィールド129は、キャッシュされた列の内容に変更があったことを表すために用いられる。データ列が新しいデータ列に上書きされる前に、修正されたフィールド129はチェックされ、もしデータ列が修正されていた場合、修正されたデータ列は、メモリ内に列を格納する場所を特定するのに格納タグフィールド121a、121b等を用いて、メモリに格納されなおされる。

【0011】

Nウェイアソシアティブキャッシュは実効メモリアクセススピードを増すための有効な手段であるが、アドレス指定されたデータ項目のタグと格納されている全てのタグとを比較

10

20

30

40

50

する同時ウェイ比較スキームは、1ウェイアソシアティブキャッシュのN倍もの割合でエネルギーを消費する。Nウェイアソシアティブキャッシュが、集積回路やプリント基板の他の部分より熱くなるのは珍しいことではない。

【0012】

従来のNウェイアソシアティブキャッシュの電力消費を低減するために、予測テクニックを利用して、与えられたアドレスに対応する可能性のあるウェイを選択することがある。従来例のウェイ予測スキームでは、アドレス指定されたデータ項目に対し、可能性のあるウェイがまずチェックされ、このウェイがアドレス指定されたデータ項目を含んでいなかった場合に限り、残りのウェイがチェックされる。Koji Inoue他による、「推測的ウェイ選択による高性能及び低電力キャッシュアーキテクチャ」(IEICE Trans. Electron, 第E83-C巻、第2号、2000年2月、186-194ページ)(引用により、本明細書中に含まれているものとする)には、ウェイ予測スキームと、ウェイ予測スキームによるエネルギー消費と非予測スキームによるエネルギー消費の比較が示されている。予測成功率が高ければ、エネルギーの節約はかなりのものとなる。ウェイ予測が成功する度に、N倍分のエネルギー削減が達成できるからである。

【0013】

図1に、アドレス指定されたデータ項目に関連する特定のウェイを予測するために用いられるウェイ予測テーブル150の例を示す。データアドレス110のサブセットが、ウェイ予測テーブル150に索引をつけるために用いられる。アドレス110のサブセットを特定するため、およびウェイ予測テーブル150の内容を得るために用いられるアルゴリズムを特定するために、様々なスキームを用いることができる。簡単に表された実施形態では、テーブル150に索引をつけるために用いられるサブセットとして、インデックスフィールド112を用い、テーブル150の内容は、各索引に対し、最小使用頻度(LRU)ウェイ、Way-a、Way-b等に対応する。あるいは、インデックスフィールド112のサブセット、またはタグフィールド111およびインデックスフィールド112の両方から得られたサブセットを用いてウェイ予測テーブル150の索引を得てもよい。ウェイ予測テーブルに索引をつけるためのアドレス110のサブセットと、ウェイの数Nの選択により、必要なウェイ予測テーブルのサイズが決定される。8ウェイアソシアティブキャッシュでは、ウェイ予測テーブルの各ウェイを一義的に特定するために3ビットが必要とされ、テーブル150における3ビットエントリの数は、テーブル150に索引をつけるためのアドレスのサブセットの一義的な組み合わせの数によって決定される。例えば、テーブル150に索引をつけるのに10ビットが使われるとすると、テーブル150では1,024(2¹⁰)の3ビットエントリができなければならない。

【0014】

アドレス110が要求されると、予測されたウェイのみを、選択的にアクセスするために、ウェイ予測テーブル150から予測されたウェイのみが使用される。便宜上、予測されたウェイを表すために、今後添え字pを用いることにする。選択されたウェイpのインデックスフィールド112に対応する格納タグ121pは、選択されたウェイのコンパレータ130pに提供され、対応するデータ列125pが選択されたウェイpのバッファ140pに提供される。予測されたウェイがデータ列を含んでいた場合、Hit-p信号がアサートされ、アドレス指定されたワードがバッファ140pから、要求を出しているプロセッサに提供される。予測されたウェイpがアドレス指定されたデータ項目を含んでいなかった場合は、アドレス指定されたデータ項目を探してNウェイアソシアティブキャッシュをチェックする、上述のテクニックを用いて、予測されなかった各ウェイがチェックされ、アドレス指定されたデータ項目の存在を探す。

【0015】

もしいずれのウェイもアドレス指定されたデータ項目を含んでいなければ、指定されたデータ項目を含んだデータ列がメモリからキャッシュ120の、通常は使用頻度が最小のウェイの索引位置にロードされる。ウェイ予測テーブル150は使用頻度が最大のウェイを格納するように構成されていると仮定して、データ列を格納するために使用されたウェイ

10

20

30

40

50

の識別 (i d e n t i f i c a t i o n) (識別子) がウェイ予測テーブル 1 5 0 に格納される。このようにして、現在アドレス指定されているデータ項目と同じデータ列にあるデータ項目が引き続き要求されると、ウェイが正しく予測され、電力消費を節約することができる。

【 0 0 1 6 】

この節電スキームの他の種類のものもまた用いられる。例えば、タグ 1 2 1 a、1 2 1 b、等のすべてが対応するコンパレータ 1 3 0 a、1 3 0 b、等にロードされるが、予測されたウェイ p のデータ列 1 2 5 p のみがバッファ 1 4 0 p にロードされてもよい。これにより、予測されたウェイが指定されたデータ項目を含んでいなかった場合に、予測されなかったウェイの全てのデータ列 1 2 5 a、1 2 5 b、等をバッファ 1 4 0 a、1 4 0 b 等にロードするのを回避し、またすべてのタグフィールド 1 2 1 a、1 2 1 b 等を再チェックするのに要する時間を省けるため、節電をすることができる。もし他のタグコンパレータ 1 3 0 の一つがヒット信号をアサートにすると、対応するウェイのデータ列が対応するバッファ 1 4 0 にロードされ、適当なワードがプロセッサに提供される。他のタグコンパレータ 1 3 0 がいずれもヒット信号をアサートにしなければ、上述の通り、アドレス指定されたデータ列がメモリからキャッシュ 1 2 0 にロードされる。

10

【 0 0 1 7 】

図 1 の従来のウェイ予測スキームでは、命令の間接アドレスに対応するメモリアドレスを判定した後でウェイ予測を行っている。パイプラインアーキテクチャでは、メモリアドレスの判定はしばしばクリティカルパス上で行われ、ウェイ予測のために遅延が生じると、

20

全体的なプロセッサ性能に直接的な影響を与えることになりうる。

【 0 0 1 8 】

発明の概要

本発明の目的は、プロセッサのクリティカルパスの外でのウェイ予測を可能にする方法およびシステムを提供することである。さらに、本発明の目的は、Nウェイアソシアティブキャッシュのウェイ予測の成功率を潜在的に向上させる方法およびシステムを提供することである。また、本発明の目的は、従来のアドレスベースのウェイ予測スキームより効率のよい、ウェイ予測スキームを提供することである。

【 0 0 1 9 】

これらおよび他の目的は、メモリ内のデータ項目にアクセスするのに間接アドレス指定を用いる命令の内容に基づいたNウェイアソシアティブキャッシュのウェイ予測スキームを提供することによって達成される。パイプライン処理のために間接アドレス命令の内容はいつでも参照でき、本発明のウェイ予測スキームはメモリアドレスの計算を必要としない。間接アドレス指定命令の内容により、メモリアクセスの抽象化をより高いレベルで行うことができ、よってデータ項目の絶対アドレスよりも、データ項目間の関係をよりよく示すものと思われる。好適な実施形態では、間接アドレス命令に含まれるベースレジスタがウェイ予測テーブルの索引を与える。同じベースレジスタを用いて間接アドレス指定されるデータ項目は関係があるものと思われ、よって間接アドレス指定命令のベースレジスタに基づいてNウェイアソシアティブメモリの特定のウェイを予測することは、キャッシュヒットにつながり、キャッシュ内の全てのウェイへのアクセスに伴うエネルギー消費を削減することができるものと思われる。

30

40

【 0 0 2 0 】

添付の図面を参照しながら本発明をさらに詳細に、例をあげながら説明する。

【 0 0 2 1 】

図全体に渡り、同じ参照番号は同様のあるいは対応する特徴又は機能を表す。添え字のある項目は、一般的な特徴又は機能の特定の例を表している。明細書中で添え字なしで参照された項目は、図示された項目のうち、同じ参照番号をもつ一つまたはすべてのものを表す。

【 0 0 2 2 】

発明の詳細な説明

50

図1は、間接アドレス指定命令160を介してメモリ内のデータ項目を間接アドレス指定する一般的な方法を示す。プロセッサに与えられる命令160は、「ロード」「格納」など、実行される動作を特定するオペレーションコードフィールド161を含む。ベースレジスタフィールド162およびオフセットフィールド164は、後で詳細に述べるようにメモリ内のアドレスを間接参照するために用いられる。レジスタフィールド163は(「ロード」命令において)データ項目を受け取ったり、(「格納」命令において)提供したりするプロセッサの内部レジスタを特定する。

【0023】

アドレス指定スペースが大きいと、明示アドレスまたは直接アドレスを収容するのに必要な命令のサイズも大きくなる。64Kワードという比較的小さなアドレススペースをアドレス指定するために、たとえばアドレスを含む命令に16ビットを割り当てなくてはならない。アドレススペースがより大きくなると、アドレスを収容するためにより大きな命令幅が必要となる。各データ転送命令において、明示アドレスを提供する代わりに、アドレスがレジスタ170に格納され、命令160は単にこのレジスタの識別子162を含む。命令160においてレジスタ170を参照することは、レジスタ170に含まれるアドレスを間接的に参照することである。一般的には8、16、32、または64レジスタが提供され、それぞれ命令160のアドレスの識別子162を格納するのに3、4、5、または6ビットを必要とするのみである。それぞれのレジスタ170のサイズは、メモリのアドレススペースに対応するアドレスを収容するのに十分なサイズである。新しいアドレスをそのたび毎にレジスタ170にリロードする必要性を回避するために、間接アドレス指定命令160にはオフセットフィールド164が与えられている。ベースレジスタ170への参照162と命令160におけるオフセット164とに対応する絶対アドレス110は、加算器180を介してベースレジスタ170とオフセット164の内容を加算することによって決定される。特定のプロセッサによっては、オフセットは符号付きの、あるいは符号無しの整数であり、符号付き整数の構成にすると、ベースレジスタ170の内容から順方向および逆方向のオフセットが可能となる。

【0024】

本発明は、間接アドレス指定命令は、プログラムにおけるデータの関係をより高いレベルで抽象化できるという考えを前提条件にしている。例えば、一般的にコンパイラはあるサブルーチン内の全ての一時変数を近接するメモリ領域に置く。これらの一時変数はそれぞれ一つのベースアドレスのオフセット範囲にあると仮定すると、コンパイラは一時変数のそれぞれを参照するために特別なベースレジスタを特定することになる。一時変数が参照されるたび、命令160には同じベースレジスタが現れる。同様に、サブルーチンへバスされる特別な引数や、これらの引数へのポインタは、メモリ内の近接した、またはほぼ近接した領域に一時的に置かれることが多く、共通のベースレジスタから参照されることになる。共通のベースレジスタがメモリ内で空間的に関連したデータ項目を参照するのに用いられる他の例としては、アレイの項目の参照、文字列内の文字の参照、などがある。他の見方をすると、同じベースレジスタを含む間接アドレス指定命令は、互いに空間的に関連した項目を参照しがちであり、よって、同じベースレジスタを含む間接アドレス指定命令の発生に基づいてデータの関係を予想することは、成功し、高いキャッシュヒットを得られるものと思われる。

【0025】

本発明は、Nウェイアソシアティブキャッシュに用いるのに特に適しているが、参照されるデータ項目を含むと思われるキャッシュのセクションを予想するのにアドレスのフィールドを従来通りに用いる他のキャッシュスキームに適用することもできる。予測に用いられる従来のアドレスフィールドを増したり、参照されるベースレジスタの識別子と置き換えることにより、より高いキャッシュヒット率が期待できる。ベースレジスタの使用とデータ項目間の空間的關係の存在との間には、間接的な関係があるからである。本発明が区分されたキャッシュを用いる他のキャッシュスキームに適用できることは当業者にとっては明らかであろうが、便宜上本明細書では、本発明はNウェイアソシアティブキャッシュ

10

20

30

40

50

の例として表すことにする。

【0026】

図2は、本発明による、ウェイ予測を行うNウェイアソシアティブキャッシュ200の例を示すブロック図である。図2には、参照されるデータ項目に対応するキャッシュ120のウェイを予測する基準として、プロセッサに与えられる間接アドレス指定命令160の内容を用いるウェイ予測テーブルが示されている。図の例では、ベースレジスタ162は、予測されたウェイを得るためのウェイ予測テーブル250に索引を与える。同じベースレジスタ162を含む命令はレジスタ170の内容のオフセット164の範囲内にある関連するデータ項目を参照することが多いため、もし参照されるデータ項目が実際にキャッシュ120内にある場合、予測されたウェイはおそらくそのデータ項目を含むウェイである。

10

【0027】

予測されたウェイが参照されたデータ項目を含んでいる可能性は、各キャッシュ列に含まれるデータ項目の数による。各列のデータ項目が多いほど、その列内に参照されたデータ項目がある可能性が高くなる。同様に、予測されたウェイが参照されたデータ項目を含む可能性は、関連するキャッシュ列を同じウェイに格納するキャッシュ割り当てスキームを用いることにより、増大する。つまり、例えば新しいキャッシュ列をロードするのにどのウェイを用いるかを決定する、上述のLRU優先順位スキームは、新しいキャッシュ列を、同じベースレジスタ162を有し、過去にロードされたキャッシュ列と同じウェイにロードされるように修正されてもよい。ここで開示されているような、間接アドレスベース

20

【0028】

メモリアドレスの計算を必要とせず、予測成功率を潜在的に向上させる予測スキームの提供に加えて、本発明のウェイ予測テーブル250によるリソースの消費は、従来のウェイ予測テーブル150より、はるかに少ないと期待できる。上述したとおり、通常レジスタ170の数は64までに制限されており、一般的には32である。本発明によるウェイ予測テーブル250の総メモリ要求量は、レジスタ170の数をキャッシュ120の各ウェイを一義的に特定するのに必要なビット数をかけたものに等しい。一般的に、レジスタ170の数はアドレス110のインデックスフィールド112に与えられる索引の数よりかなり小さい。故に、ウェイ予測テーブルのサイズを削減することができるのである。

30

【0029】

好適な実施形態では、命令を実行するときにデータアクセスを開始してもかまわないにもかかわらず、命令160は実際に実行される前にデータアクセスを開始するように処理されている。ベースレジスタ162の内容はウェイ予測テーブル250で予測されたウェイに索引を付けるために使われる。この予測されたウェイは、キャッシュ120の予測されたウェイpをイネーブルにし、コンパレータ130p経由で予測されたウェイ内にアドレス指定されたデータ項目が含まれているかどうかをチェックできるようにする。このチェック作業と平行して、もし予測されたウェイpがアドレス指定されたデータ項目を含んでいると、データ列125pが直ちに使用可能なようにバッファ140pにロードされる。予測されたウェイがアドレス指定されたデータ項目を含んでいると、ワードフィールド113がバッファ140pのデータ列からアドレス指定されたデータ項目を選択し、命令160が実行されるときにプロセッサにそれを渡す。予測されたウェイpがアドレス指定されたデータ項目を含んでいなかった場合は、図1に関連して上に詳述したように、他のウェイがチェックされる。

40

【0030】

上述のことは、単に本発明の原理を述べただけにすぎない。当業者であれば、本明細書では明白に記載はしていなくても、本発明の範囲内で本発明の原理を具体化した様々な構成例を考案することは可能である。例えば、予測のレベルをさらに正確なものにするために、ベースレジスタ162とオフセット164のサブセットの組み合わせを用いることがで

50

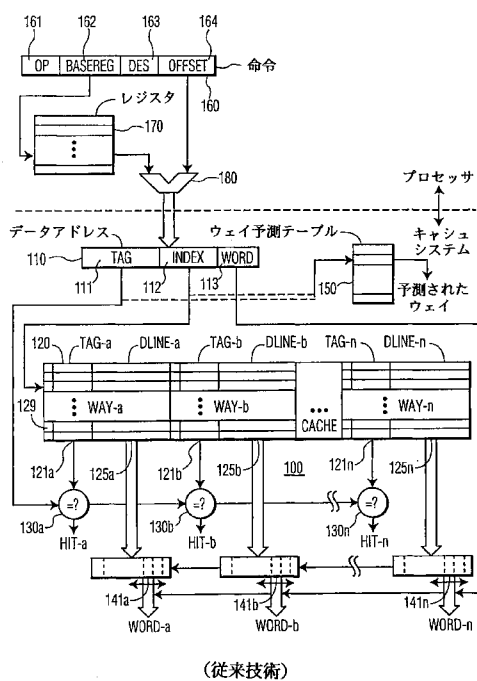
きる。例えば、もしオフセットが符号付きの整数であればベースレジスタ162とオフセット164の最上位ビットの組み合わせにより、レジスタ170に含まれるベースアドレスより小さい項目を参照するウェイ予測と、ベースアドレスより大きい項目を参照するウェイ予測を行うことができる。オフセット164の他のビットをさらに用いることにより、予測の精度をさらに上げることができる。極端な例では、ウェイ予測テーブルは、オフセット164のデータ列サイズのセグメントについてそれぞれ予測するように構成してもよい。この構成では、従来のアドレスベースのタグ予測と同じ程度、あるいはそれ以上のビット格納が必要となるかも知れないが、ウェイ予測テーブルを構成するのにベースレジスタを用いることにより、従来のアドレスベースのウェイ予測スキームよりかなり高いキャッシュヒット率を得ることが期待できる。上述の、および他のシステム構成や最適化構成は本明細書の開示内容を参照すれば当業者には明らかであり、添付の請求項の範囲に含まれるものである。

【図面の簡単な説明】

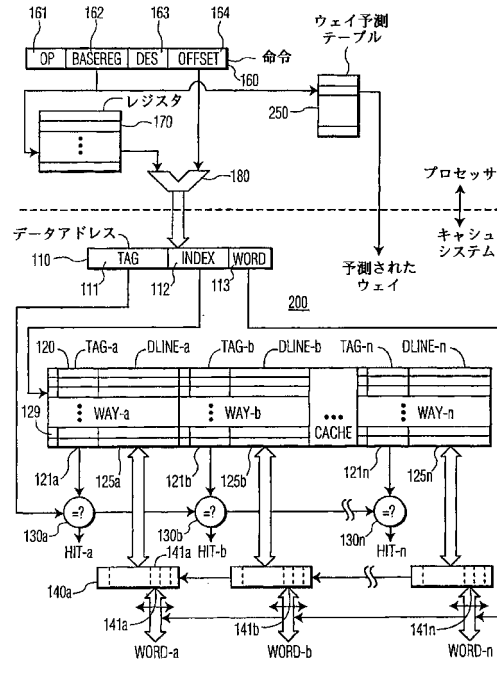
【図1】 従来の、ウェイ予測を行うNウェイアソシアティブキャッシュの例を示すブロック図。

【図2】 本発明によるウェイ予測を行うNウェイアソシアティブキャッシュの例を示すブロック図。

【図1】



【図2】



フロントページの続き

(74)代理人 100103263

弁理士 川崎 康

(72)発明者 ジャン、ダブリュー・バン、ダ、ワード

オランダ国5656、アーアー、アインドーフエン、プロフ・ホルストラーン、6

(72)発明者 ボール、ストラバース

オランダ国5656、アーアー、アインドーフエン、プロフ・ホルストラーン、6

審査官 清木 泰

(56)参考文献 特開平10-320275(JP,A)

特開平04-277846(JP,A)

特開平03-194632(JP,A)

特開平02-144640(JP,A)

特開平02-024718(JP,A)

特開昭63-145533(JP,A)

特開昭56-016981(JP,A)

国際公開第01/042927(WO,A1)

Brad Calder, Dirk Grunwald, Joel Emer, Predictive Sequential Associative Cache, Proceedings of the 2nd International Symposium on High Performance Computer Architecture, 米国, IEEE Computer Society, 1996年 2月 3日, Pages:244-253

(58)調査した分野(Int.Cl., DB名)

G06F12/08-12/12