

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3836322号

(P3836322)

(45) 発行日 平成18年10月25日(2006.10.25)

(24) 登録日 平成18年8月4日(2006.8.4)

(51) Int. Cl.		F I			
G06F	9/38	(2006.01)	G06F	9/38	310A
G06F	12/08	(2006.01)	G06F	12/08	509Z

請求項の数 25 (全 21 頁)

(21) 出願番号	特願2000-556301 (P2000-556301)	(73) 特許権者	591016172
(86) (22) 出願日	平成11年1月25日 (1999.1.25)		アドバンスト・マイクロ・デバイス・
(65) 公表番号	特表2002-519755 (P2002-519755A)		インコーポレイテッド
(43) 公表日	平成14年7月2日 (2002.7.2)		ADVANCED MICRO DEVI
(86) 国際出願番号	PCT/US1999/001466		CES INCORPORATED
(87) 国際公開番号	W01999/067705		アメリカ合衆国、94088-3453
(87) 国際公開日	平成11年12月29日 (1999.12.29)		カリフォルニア州、サニベイ、ピ・
審査請求日	平成18年1月10日 (2006.1.10)		オウ・ボックス・3453、ワン・エイ・
(31) 優先権主張番号	09/103,956		エム・ディ・プレイス、メイル・ストップ
(32) 優先日	平成10年6月24日 (1998.6.24)		・68 (番地なし)
(33) 優先権主張国	米国 (US)	(74) 代理人	100064746
			弁理士 深見 久郎
		(74) 代理人	100085132
			弁理士 森田 俊雄

最終頁に続く

(54) 【発明の名称】 プリデコード情報を記憶するためのECC/パリティビットの使用

(57) 【特許請求の範囲】

【請求項1】

レベル2 キャッシュから命令バイトを受取り対応するプリデコード情報を生成するよう構成されるプリデコードユニットと、

前記プリデコードユニットに結合される命令キャッシュとを含み、前記命令キャッシュは、前記プリデコードユニットから前記命令バイトおよび前記プリデコード情報を受取り記憶するよう構成され、前記命令キャッシュは、前記命令バイトおよび前記プリデコード情報が置換されるのに応じて前記プリデコード情報の少なくとも一部を出力してレベル2 キャッシュに記憶させるよう構成され、前記レベル2 キャッシュは、パリティ情報を記憶するよう指定される記憶場所にプリデコード情報を記憶するよう構成される、マイクロプロセッサ。

【請求項2】

前記プリデコードユニットは、レベル2 キャッシュから前記命令バイトに対応するプリデコード情報を受取るようさらに構成される、請求項1に記載のマイクロプロセッサ。

【請求項3】

前記プリデコード情報は、開始ビット、終了ビットおよびoppコードビットのうち1つ以上を含む、請求項1に記載のマイクロプロセッサ。

【請求項4】

前記プリデコード情報は命令長情報を含む、請求項1に記載のマイクロプロセッサ。

【請求項5】

10

20

前記プリデコードユニットは、前記プリデコードユニットが予め定められた無効定数に等しいレベル2 キャッシュからのプリデコード情報を受取るとき前記命令バイトについてのプリデコード情報を生成するよう構成される、請求項1に記載のマイクロプロセッサ。

【請求項6】

前記レベル2 キャッシュに結合され、かつ主メモリから受取った命令バイトを検出すると、レベル2 キャッシュに前記無効定数を記憶するよう構成される制御ユニットをさらに含む、請求項5に記載のマイクロプロセッサ。

【請求項7】

レベル2 キャッシュから命令バイトを受取り対応するプリデコード情報を生成するよう構成されるプリデコードユニットと、

10

前記プリデコードユニットに結合される命令キャッシュとを含み、前記命令キャッシュは、前記プリデコードユニットから前記命令バイトおよび前記プリデコード情報を受取り記憶するよう構成され、前記命令キャッシュは、前記命令バイトおよび前記プリデコード情報が置換されるのに応じて前記プリデコード情報の少なくとも一部を出力してレベル2 キャッシュに記憶させるよう構成され、前記レベル2 キャッシュは、エラーチェックおよび訂正(ECC)情報を記憶するよう指定される記憶場所にプリデコード情報を記憶するよう構成される、マイクロプロセッサ。

【請求項8】

前記プリデコードユニットは、レベル2 キャッシュから前記命令バイトに対応するプリデコード情報を受取るようさらに構成される、請求項7に記載のマイクロプロセッサ。

20

【請求項9】

前記プリデコード情報は、開始ビット、終了ビットおよびオペコードビットのうち1つ以上を含む、請求項7に記載のマイクロプロセッサ。

【請求項10】

前記プリデコード情報は命令長情報を含む、請求項7に記載のマイクロプロセッサ。

【請求項11】

前記プリデコードユニットは、前記プリデコードユニットが予め定められた無効定数に等しいレベル2 キャッシュからのプリデコード情報を受取るとき前記命令バイトについてのプリデコード情報を生成するよう構成される、請求項7に記載のマイクロプロセッサ。

【請求項12】

30

前記レベル2 キャッシュに結合され、かつ主メモリから受取った命令バイトを検出すると、レベル2 キャッシュに前記無効定数を記憶するよう構成される制御ユニットをさらに含む、請求項11に記載のマイクロプロセッサ。

【請求項13】

主システムメモリから命令バイトを讀出すステップと、
命令バイトに対応するプリデコード情報を生成するステップと、
命令バイトおよびプリデコード情報を命令キャッシュに記憶するステップと、
プリデコード情報に対応する命令バイトが命令キャッシュ中で上書きされるときプリデコード情報の少なくとも一部をレベル2 キャッシュに出力して記憶させるステップとを含み、

40

前記出力ステップは、プリデコード情報をレベル2 キャッシュに出力して、パリティ情報を記憶するよう構成される記憶場所に記憶させるステップを含む、方法。

【請求項14】

前記生成ステップは、命令バイトのための以下のプリデコードビット、すなわち、開始ビット、終了ビットおよびオペコードビットのうち1つ以上を生成するステップを含む、請求項13に記載の方法。

【請求項15】

無効定数をレベル2 キャッシュに記憶することによってレベル2 キャッシュに記憶されるプリデコード情報を初期化するステップをさらに含む、請求項13に記載の方法。

【請求項16】

50

対応する命令バイトが変更された場合にプリデコード情報を再計算するステップをさらに含む、請求項 13 に記載の方法。

【請求項 17】

主システムメモリから命令バイトを読み出すステップと、
命令バイトに対応するプリデコード情報を生成するステップと、
命令バイトおよびプリデコード情報を命令キャッシュに記憶するステップと、
プリデコード情報に対応する命令バイトが命令キャッシュ中で上書きされるときプリデコード情報の少なくとも一部をレベル 2 キャッシュに出力して記憶させるステップとを含む、

前記出力ステップは、プリデコード情報をレベル 2 キャッシュに出力して、エラーチェックおよび訂正 (ECC) 情報を記憶するよう構成される記憶場所に記憶させるステップを含む、方法。

10

【請求項 18】

前記生成ステップは、命令バイトのための以下のプリデコードビット、すなわち、開始ビット、終了ビットおよび op コードビットのうち 1 つ以上を生成するステップを含む、請求項 17 に記載の方法。

【請求項 19】

無効定数をレベル 2 キャッシュに記憶することによってレベル 2 キャッシュに記憶されるプリデコード情報を初期化するステップをさらに含む、請求項 17 に記載の方法。

【請求項 20】

20

対応する命令バイトが変更された場合にプリデコード情報を再計算するステップをさらに含む、請求項 17 に記載の方法。

【請求項 21】

複数のキャッシュメモリを有するコンピュータシステムを動作させるための方法であって、

第 1 のキャッシュメモリに命令プリデコード情報を記憶するステップと、

前記第 1 のキャッシュメモリ中の古いキャッシュエントリを新たなキャッシュエントリで上書きするステップと、

前記古いキャッシュエントリに対応する命令プリデコード情報を第 1 のキャッシュから第 2 のキャッシュメモリのエラーチェックおよび訂正部に退避させるステップとを含む、方法。

30

【請求項 22】

前記第 2 のキャッシュメモリのエラーチェックおよび訂正部は、第 2 のキャッシュメモリのパリティ部である、請求項 21 に記載の方法。

【請求項 23】

前記命令プリデコード情報は、開始ビット、終了ビットおよび op コードビットのうち 1 つ以上を含む、請求項 21 に記載の方法。

【請求項 24】

コンピュータシステムであって、

レベル 2 キャッシュと、

40

前記レベル 2 キャッシュに結合されるマイクロプロセッサとを含み、前記マイクロプロセッサは、

前記レベル 2 キャッシュから命令バイトを受取り対応するプリデコード情報を生成するよう構成されるプリデコードユニットと、

前記プリデコードユニットから前記命令バイトおよび前記プリデコード情報を受取り記憶するよう構成される命令キャッシュとを含み、前記命令キャッシュは、前記命令バイトが置換されるときに前記対応するプリデコード情報の少なくとも一部を出力して前記レベル 2 キャッシュに記憶させるよう構成され、前記レベル 2 キャッシュは、複数の命令記憶場所と、対応する複数のエラーチェックおよび訂正 (ECC) 記憶場所とを含み、前記レベル 2 キャッシュは、前記 ECC 記憶場所に前記プリデコード情報を記憶するよう構成さ

50

れる、コンピュータシステム。

【請求項 25】

前記 ECC 記憶場所はパリティ記憶場所である、請求項 24 に記載のコンピュータシステム。

【発明の詳細な説明】

【0001】

【発明の背景】

1. 技術分野

この発明は、マイクロプロセッサに関し、より特定的には、マイクロプロセッサ内の可変長命令をデコードすることに関する。

10

【0002】

2. 背景技術

x86 命令セットのために書かれたソフトウェアアプリケーションの数は、かなり多い。その結果、より新しくより進歩した命令セットの導入にもかかわらず、マイクロプロセッサの設計者は、x86 命令セットを実行することのできるマイクロプロセッサを設計し続けてきた。

【0003】

x86 命令セットは、比較的複雑であり、複数個の可変長命令を特徴とする。x86 命令セットを示す一般的なフォーマットを図 1 に示す。図に例示されるように、x86 命令は、1 から 5 のオプションのプレフィックスバイト 102 と、その後続く操作コード (op コード) フィールド 104 と、オプションのアドレッシングモード (Mod R/M) バイト 106 と、オプションのスケール - インデックス - ベース (SIB) バイト 108 と、オプションの変位フィールド 110 と、オプションの即値データフィールド 112 とからなる。

20

【0004】

操作コードフィールド 104 は、特定の命令のための基本操作を定義する。特定の操作コードのデフォルト操作は、1 つ以上のプレフィックスバイト 102 によって変更可能である。たとえば、プレフィックスバイト 102 の 1 つを用いて、命令についてのアドレスまたはオペランドサイズを変更し、メモリアドレッシングに用いられるデフォルトセグメントを無効にし、または複数回ストリング操作を繰返すようプロセッサに命令することが可能である。操作コードフィールド 104 は、もし存在するならばプレフィックスバイト 102 の後に続き、1 または 2 バイトの長さであってもよい。アドレッシングモード (Mod R/M) バイト 106 は、使用されるレジスタおよびメモリアドレッシングモードを特定する。スケール - インデックス - ベース (SIB) バイト 108 は、スケールファクタおよびインデックスファクタを用いる 32 ビットベース相対アドレッシングにおいてのみ用いられる。SIB バイト 108 内のベースフィールドは、どのレジスタがアドレス計算のためのベース値を含むかを特定し、SIB バイト 108 内のインデックスフィールドは、どのレジスタがインデックス値を含むかを特定する。SIB バイト 108 内のスケールフィールドは、任意の変位とともにベース値に加算されるより前に、インデックス値に乘算される 2 のべきを特定する。次の命令フィールドは、変位フィールド 110 であり、これはオプションであり、1 から 4 バイトの長さであってもよい。変位フィールド 110 は、アドレス計算に用いられる定数を含む。オプションの即値フィールド 112 は、これもまた 1 から 4 バイトの長さであってもよいが、命令オペランドとして用いられる定数を含む。最短の x86 命令は、1 バイトのみの長さであり、単一の操作コードバイトを含む。80286 は、命令の最大長を 10 バイトに設定し、80386 および 80486 はどちらも、最大 15 バイトの命令長を可能にする。

30

40

【0005】

x86 命令セットの複雑性は、高性能の x86 互換のマイクロプロセッサを実現する際に多くの困難を提起する。特に、x86 命令の可変長は、命令のデコードを困難にする。命令のデコードは典型的には、命令の境界を決定し、次に命令内の各フィールド、たとえば

50

操作コードフィールドおよびオペランドフィールドを特定することを伴なう。デコードは典型的には、命令が実行されるより前に命令キャッシュからフェッチされるときに起こる。

【0006】

命令の境界を決定するためのある方法は、主メモリから読出され命令キャッシュに記憶される際に各命令バイトについて1つ以上のプリデコードビットを生成しかつ記憶することを伴なう。プリデコードビットは、それが関連付けられる命令バイトについての情報を与える。たとえば、アサートされたプリデコード開始ビットは、関連付けられた命令バイトが命令の第1のバイトであることを示す。特定の命令バイトについての開始ビットが一旦計算されると、それは命令バイトとともに命令キャッシュ内に記憶される。「フェッチ」が実行されるとき、多数の命令バイトが命令キャッシュから読出され、実行のための準備としてデコードされる。関連付けられた開始ビットはいずれも、フェッチとともに個々の命令のための有効マスクを生成するためにスキャンされる。有効マスクは、各ビットが特定の命令バイトに対応している一連のビットである。命令の第1のバイト、命令の最終バイト、および命令の第1のバイトと最終バイトとの間にあるすべてのバイトに関連付けられる有効マスクビットがアサートされる。すべての他の有効マスクビットは、アサートされない。有効マスクが計算されると、それを用いて他の命令からのバイトをマスク消去(mask-off)可能である。

10

【0007】

図2を次に参照すると、例示の有効マスクが示される。図は、フェッチ120の一部と、その関連付けられた開始ビット122とを示す。命令B128のための有効マスク126が生成されるべきであると想定すると、開始ビット122Aと、開始ビット122Bとの間のすべてのビットがアサートされてマスク126を生成する。一旦生成されると次に、有効マスク126を用いて、命令B128の一部ではないフェッチ120内のすべてのバイトをマスク消去可能である。

20

【0008】

上の記載が示すように、プリデコード情報は、デコード回数を低減するために特に有用であるだろう。命令キャッシュ内に命令バイトとともにプリデコード情報を記憶することによって、対応する命令が複数回(たとえばループで)実行されたとしても、プリデコード情報は1回計算されるだけで済む。しかし、残念ながら、命令が置換されるかまたは命令キャッシュから廃棄されるとき、関連付けられたプリデコード情報はいずれも失われる。次に命令が命令キャッシュに読出されるときに、プリデコード情報はまたもや生成されなければならない。プリデコード情報が計算されるのを待つことによって生じる時間遅延は、分岐予測誤りまたはキャッシュミスの結果として命令が命令キャッシュに読出されるとき特に性能を損なう可能性がある。必要とされるより前に投機的にプリフェッチされる命令とは対照的に、分岐予測誤りまたはキャッシュミスによって生じるフェッチは、要求された命令を受取るのを待つ間にマイクロプロセッサのデコーダおよび機能ユニットをストールさせてしまう可能性がある。この場合には、プリデコード情報を発生するために必要とされる時間は、マイクロプロセッサの性能にかなりの影響を与え兼ねない。

30

【0009】

これらおよび他の理由のために、プリデコード回数を低減するための方法および装置が所望される。特に、命令キャッシュから先に廃棄されてしまった命令についてのプリデコード情報を生成するために必要とされる時間を減ずるための方法および装置が所望される。

40

【0010】

【発明の開示】

以上に概略した課題は、レベル2キャッシュにプリデコード情報を記憶することのできるマイクロプロセッサによって大部分解決される。ある実施例では、マイクロプロセッサは、プリデコード情報または対応する命令バイトのいずれかが命令キャッシュ内で置換されるときレベル2キャッシュ内にプリデコード情報を記憶するよう構成可能である。

【0011】

50

ある実施例では、マイクロプロセッサは、プリデコードユニットおよび命令キャッシュを含む。プリデコードユニットは、レベル2キャッシュから命令バイトを受取り対応するプリデコード情報を生成するよう構成される。命令キャッシュは、プリデコードユニットに結合され、2つの複数個の記憶場所：命令バイトを記憶するためのものと、命令バイトに対応するプリデコード情報を記憶するための第2のものを含む。命令キャッシュは、命令バイトを受取り記憶しかつプリデコードユニットからの情報をプリデコードするよう構成可能である。命令キャッシュはまた、命令バイトおよび対応するプリデコード情報が命令キャッシュ内で置換されるとき対応するプリデコード情報の少なくとも一部を出力するよう構成可能である。プリデコード情報は出力されレベル2キャッシュ内に記憶され得る。命令キャッシュ内で置換された後およびレベル2キャッシュ内で置換されるより前に同じ命令バイトが必要とされる場合、命令バイトおよび命令に付随するプリデコード情報は、レベル2キャッシュから取り出し可能である。有利なことには、命令バイトをプリデコードすることの遅延は、いくつかの場合においては回避可能である。

10

【0012】

別の実施例では、プリデコード情報は、パリティおよびまたはエラーチェックおよび訂正情報を記憶するよう構成されるレベル2キャッシュ内の記憶場所に記憶され得る。パリティおよびまたはエラーチェックおよび訂正記憶場所を用いることによって、レベル2キャッシュ構成への変更を低減したり排除したりすることが可能である。

【0013】

さらに別の実施例では、無効定数が、有効プリデコード情報をまだ有していない命令バイトのためにレベル2キャッシュ内に記憶され得る。プリデコードユニットは、レベル2キャッシュから命令バイトを読み出すときに無効定数を検出し、これに回答して新しいプリデコード情報を計算するよう構成され得る。

20

【0014】

可変長命令をプリデコードするための方法もまた企図される。ある実施例では、この方法は、主システムメモリから命令バイトを読み出し、命令バイトに対応するプリデコード情報を生成するステップを含む。命令バイトおよびプリデコード情報は、命令キャッシュ内に記憶される。プリデコード情報に対応する命令バイトが命令キャッシュ内で上書きされるとき、プリデコード情報の少なくとも一部がレベル2キャッシュに出力され記憶される。プリデコード情報はやはり、パリティおよびまたはエラーチェックおよび訂正データを記憶するよう構成されるレベル2キャッシュ内の記憶場所内に記憶され得る。

30

【0015】

プリデコードデータを記憶するよう構成されるコンピュータシステムもまた企図される。ある実施例では、コンピュータシステムは、レベル2キャッシュおよびマイクロプロセッサを含んでもよい。マイクロプロセッサは、レベル2キャッシュに結合されてもよく、上述したようにプリデコードユニットおよび命令キャッシュで構成されてもよい。プリデコードユニットは、レベル2キャッシュから命令バイトを受取り対応するプリデコード情報を生成するよう構成されてもよい。命令キャッシュは、プリデコードユニットに結合されてもよく、前記命令バイトが置換されたときその記憶されたプリデコード情報の少なくとも一部を出力してレベル2キャッシュ中に記憶するよう構成されてもよい。レベル2キャッシュは、パリティおよびまたはエラーチェックおよび制御データを記憶するよう構成される記憶場所内に命令キャッシュからのプリデコード情報を記憶するよう構成されてもよい。

40

【0016】

この発明の他の目的および利点は、以下の詳細な記載を読みかつ添付の図面を参照すると明らかとなるであろう。

【0017】

この発明は、さまざまな変形および代替の態様を受入れるが、その具体的な実施例は、図面に例として示され、ここに詳細に記載される。しかしながら、図面およびその詳細な記載は、開示される特定の態様にこの発明を限定するものではなく、反対に、その意図は、

50

前掲の特許請求の範囲によって定義されるようなこの発明の範囲内にあるすべての変形、等価および代替を含むものであることが理解されるべきである。

【0018】

【発明を実施するモード】

ここで図3を参照すると、マイクロプロセッサ10のある実施例のブロック図が示される。マイクロプロセッサ10は、プリフェッチ/プリデコードユニット12と、分岐予測ユニット14と、命令キャッシュ16と、命令整列ユニット18と、複数のデコードユニット20A~20Cと、複数のリザベーションステーション22A~22Cと、複数の機能ユニット24A~24Cと、ロード/ストアユニット26と、データキャッシュ28と、レジスタファイル30と、リオーダバッファ32と、MROMユニット34とを含む。特定の参照番号が文字の後に付されているここに参照される要素は、参照番号のみによって集合的に参照されることがある。たとえば、リザベーションステーション22A~22Cは、リザベーションステーション22として集合的に参照されることがある。

10

【0019】

プリフェッチ/プリデコードユニット12は、主メモリサブシステム(図示せず)からの命令を受取るよう結合され、さらに命令キャッシュ16および分岐予測ユニット14に結合される。同様に、分岐予測ユニット14は、命令キャッシュ16に結合される。分岐予測ユニット14はまた、命令整列ユニット18および機能ユニット24A~Cに結合される。命令キャッシュ16はさらに、MROMユニット34および命令整列ユニット18に結合される。命令整列ユニット18は、ロード/ストアユニット26におよびそれぞれのデコードユニット20A~Cに結合される。それぞれのデコードユニット20A~Cは、リザベーションステーション22A~Cに結合され、これはさらにそれぞれの機能ユニット24A~Cに結合される。加えて、命令整列ユニット18およびリザベーションステーション22は、レジスタファイル30およびリオーダバッファ32に結合される。機能ユニット24は、ロード/ストアユニット26、レジスタファイル30およびリオーダバッファ30にも結合される。データキャッシュ28は、ロード/ストアユニット26におよび主メモリサブシステムに結合される。最後に、MROMユニット34は、命令整列ユニット18に結合される。

20

【0020】

命令は、プリフェッチ/プリデコードユニット12によって主メモリからプリフェッチされる。プリフェッチ/プリデコードユニット12は、可変長命令を固定長命令にプリデコードし、これは次に命令キャッシュ16内に記憶される。命令は、それらが実際にプリフェッチ方式を用いることによって要求されるより前にプリフェッチおよびプリデコード可能である。さまざまなプリフェッチ方式が、プリフェッチ/プリデコードユニット12によって採用可能である。プリデコードユニット12および命令キャッシュ16のさらに詳細な記載に進む前に、図に示される例示のマイクロプロセッサ10の実施例に関する一般的な局面を記載する。

30

【0021】

マイクロプロセッサ10は、条件付き分岐命令の後の命令を投機的にフェッチするために分岐予測を採用し得る。分岐予測ユニット14は、分岐予測操作を実行するために含まれる。ある実施例では、最大2つの分岐ターゲットアドレスが、命令キャッシュ16中に各キャッシュラインの各16バイト部分ごとに記憶される。プリフェッチ/プリデコードユニット12は、特定のラインがプリデコードされると最初の分岐ターゲットを決定する。キャッシュラインに対応する分岐ターゲットに対するその後の更新は、キャッシュライン内の命令の実行によって起こり得る。命令キャッシュ16は、分岐予測ユニット14にフェッチされるべき命令アドレスの表示を与える。これは、分岐予測ユニット14が分岐予測を形成するときどの分岐ターゲットアドレスを選択すべきかを決定することを可能にする。命令整列ユニット18および機能ユニット24は、分岐予測ユニット14に更新情報を与える。分岐予測ユニット14は、キャッシュラインの16バイト部分当り2個のターゲットを記憶するので、ライン内のいくつかの分岐命令についての予測は、分岐予測ユ

40

50

ニット14中に記憶されないかもしれない。命令整列ユニット18は、分岐予測ユニット14によって予測されなかった分岐命令を検出するよう構成され得る。機能ユニット24は、分岐命令を実行し、予測された分岐方向が予測誤りでなかったかどうかを決定する。分岐方向は、その後の命令が分岐命令のターゲットアドレスからフェッチされる場合には「テイクン」であり得る。反対に、分岐方向は、その後の命令が分岐命令に連続するメモリ場所からフェッチされる場合には「ノットテイクン」であり得る。予測誤りされた分岐命令が検出されると、予測誤りされた分岐の後に続く命令は、マイクロプロセッサ10のさまざまなユニットから廃棄される。さまざまな好適な分岐予測アルゴリズムが、分岐予測ユニット14によって採用可能である。

【0022】

命令キャッシュ16は、プリフェッチ/プリデコードユニット12から受取られた命令を記憶するために設けられる高速キャッシュメモリである。記憶された命令は次に、命令キャッシュ16からフェッチされ、命令整列ユニット18に転送される。ある実施例では、命令キャッシュ16は、セットアソシアティブ構造として構成可能である。命令キャッシュ16は加えて、アクセス時間を早めるためにウェイ予測方式を採用してよい。たとえば、命令の各ラインを識別するタグにアクセスしタグをフェッチアドレスと比較してウェイを選択する代わりに、命令キャッシュ16は、アクセスされるウェイを予測し得る。この態様では、ウェイは、アレイにアクセスするより前に投機的に選択される。ウェイ予測を用いて、命令キャッシュ16のアクセス時間は、ダイレクトマッピングされたキャッシュと同様になり得る。命令バイトが読出された後、ベリフィケーションのためにタグ比較が

【0023】

MROMユニット34は、「高速パス命令」のシーケンスを記憶するよう構成されるリードオンリメモリである。高速パス命令は、デコーダ20A~Cおよび機能ユニット24A~Cによってデコードおよび実行され得る命令である。対照的に、「MROM命令」は、デコーダ20A~Cおよび機能ユニット24A~Cが直接デコードまたは実行するには複雑すぎる命令である。命令キャッシュ16がMROM命令を出力すると、MROMユニット34は、高速パス命令のシーケンスを出力することによって応答する。より具体的には、MROMユニット34は、MROM命令を構文解析しかつ定義された高速パス命令のサブセットに変換して所望の操作を実行する。MROMユニット34は、高速パス命令のサブセットをデコードユニット20A~Cにディスパッチする。

【0024】

一旦命令バイトが命令キャッシュ16からフェッチされると、それらは命令整列ユニット18に送られる。命令整列ユニット18は、デコードユニット20A~Cの1つに命令を送る。レジスタオペランド情報もまた検出されレジスタファイル30およびリオーダバッファ32に送られる。加えて、もし命令が1つ以上のメモリ操作が実行されることを要求するならば、命令整列ユニット18は、メモリ操作をロード/ストアユニット26にディスパッチする。デコードされた命令の各々は、命令に含まれ得る変位または即値データおよびオペランドアドレス情報とともにリザベーションステーション22にディスパッチされる。

【0025】

マイクロプロセッサ10は、飛越し実行をサポートし、したがってリオーダバッファ32を採用してレジスタ読出および書込操作のための元々のプログラムシーケンスを追跡し、投機的命令実行および分岐予測誤り復旧を可能にするためにレジスタリネーミングを実現し、正確な例外を促進する。レジスタの更新を伴う命令がデコードされると、リオーダバッファ32内の一時記憶場所が確保される。一時記憶場所は、命令の投機的実行から生じる投機的レジスタ状態を記憶する。分岐予測が正しくなければ、予測誤りされた経路と

10

20

30

40

50

ともに投機的に実行された命令からの結果は、それらがレジスタファイル30に書込まれる前にリオーダバッファ32中で無効化され得る。同様に、特定の命令が例外を引起こした場合、例外を引起こした命令の後に続く命令が廃棄され得る。この態様では、例外は、「正確である」(すなわち、例外を引起す命令の後に続く命令は、例外より前に完了されない)。なお、特定の命令は、それがプログラム順序で特定の命令に先行する命令より前に実行されるならば、投機的に実行される。先行する命令は、分岐命令または例外を引起す命令であり得、この場合には、投機的結果は、リオーダバッファ32によって廃棄され得る。

【0026】

命令整列ユニット18の出力に与えられるデコードされた命令および即値データまたは変位データは、それぞれのリザベーションステーション22に直接送られる。ある実施例では、各リザベーションステーション22は、対応する機能ユニットへの発行を待っている最大3個のペンディングの命令についての命令情報(すなわち、デコードされた命令ならびにオペランド値、オペランドタグおよび/または即値データ)を保持することができる。なお、図に示す実施例では、各リザベーションステーション22は、専用の機能ユニット24に関連付けられる。したがって、3つの専用の「発行位置」が、リザベーションステーション22および機能ユニット24によって形成される。言い換えれば、発行位置0は、リザベーションステーション22Aおよび機能ユニット24Aによって形成される。整列させられリザベーションステーション22Aにディスパッチされた命令は、機能ユニット24Aによって実行される。同様に、発行位置1は、リザベーションステーション22Bおよび機能ユニット24Bによって形成され、発行位置2は、リザベーションステーション22Cおよび機能ユニット24Cによって形成される。

【0027】

特定の命令がデコードされると、要求されるオペランドがレジスタ場所であれば、レジスタアドレス情報は、リオーダバッファ32およびレジスタファイル30に同時に送られる。当業者は、x86レジスタファイルが8個の32ビット実レジスタ(すなわち、EAX、EBX、ECX、EDX、EBP、ESI、EDIおよびESPと典型的に呼ばれる)を含むことを認めるであろう。x86マイクロプロセッサアーキテクチャを採用するマイクロプロセッサ10の実施例では、レジスタファイル30は、32ビット実レジスタの各々について記憶場所を含む。さらなる記憶場所が、レジスタファイル30内に含まれMR
OMユニット34によって使用されてもよい。リオーダバッファ32は、これらのレジスタの内容を変更することによって飛越し実行を可能にする結果のための一時記憶場所を含む。リオーダバッファ32の一時記憶場所は、デコードされると実レジスタの1つの内容を変更するよう決定される各命令ごとに確保される。したがって、特定のプログラムの実行中のさまざまな点で、リオーダバッファ32は、所与のレジスタの投機的に実行された内容を含む1つ以上の場所を有し得る。

【0028】

所与の命令のデコードの後に、リオーダバッファ32が、先行の場所または所与の命令中のオペランドとして用いられるレジスタに割当てられる場所を有していると決定されたならば、リオーダバッファ32は、対応するリザベーションステーションに、1)最も最近に割当てられた場所の値、または2)先の命令をやがて実行するであろう機能ユニットによって値がまだ発生されていない場合、最も最近に割当てられた場所についてのタグのいずれかを転送する。リオーダバッファ32が所与のレジスタについて確保された場所を有していれば、オペランド値(またはリオーダバッファタグ)は、レジスタファイル30からではなくリオーダバッファ32から与えられる。要求されるレジスタについて確保された場所がリオーダバッファ32中になければ、値はレジスタファイル30から直接取られる。オペランドがメモリ場所に対応していれば、オペランド値は、ロード/ストアユニット26を介してリザベーションステーションに与えられる。

【0029】

ある特定の実施例では、リオーダバッファ32は、ユニットとして同時にデコードされた

10

20

30

40

50

命令を記憶しかつ操作するよう構成される。この構成は、ここでは「ライン指向」と呼ばれる。いくつかの命令をとともに操作することによって、リオーダバッファ32内に採用されるハードウェアが簡素化され得る。たとえば、この実施例に含まれるライン指向のリオーダバッファは、1つ以上の命令が命令整列ユニット18によってディスパッチされるたびに、3個の命令に付属する命令情報のために十分な記憶装置を割当てる。対照的に、実際にディスパッチされる命令の数に依存して、可変量の記憶装置が従来のリオーダバッファに割当てられる。比較的より数の多い論理ゲートが、可変量の記憶装置を割当てるために必要とされるであろう。同時にデコードされた命令の各々が実行されると、命令結果は、レジスタファイル30中に同時に記憶される。次に記憶装置は、同時にデコードされた命令の別の組に自由に割当てられる。加えて、制御論理はいくつかの同時にデコードされた命令にわたって償却されるので、命令当り採用される制御論理回路の量は低減される。特定の命令を識別するリオーダバッファタグは、2つのフィールド：ラインタグおよびオフセットタグに分割され得る。ラインタグは、特定の命令を含む同時にデコードされた命令の組を識別し、オフセットタグは、組内のどの命令が特定の命令に対応するかを識別する。なお、命令結果をレジスタファイル30に記憶し対応する記憶装置を自由にすることを、命令を「リタイアする」と言う。さらになお、いかなるリオーダバッファ構成が、マイクロプロセッサ10のさまざまな実施例において採用されてもよい。

【0030】

先に記したように、リザベーションステーション22は、命令が対応する機能ユニット24によって実行されるまで命令を記憶する。命令は、もし(i)その命令のオペランドが与えられてしまっており、(ii)同じリザベーションステーション22A~22C内にあり、かつプログラム順序でその命令より前にある命令についてのオペランドがまだ与えられていないならば、選択され実行される。なお、命令が機能ユニット24の1つによって実行されると、その命令の結果は、その結果を待っているいずれかのリザベーションステーション22に直接送られ、同時に結果はリオーダバッファ32を更新するために送られる(この技術は、普通「結果転送」と呼ばれている)。命令は、関連付けられる結果が転送されるクロックサイクル中に、実行のために選択され機能ユニット24A~24Cに送られ得る。リザベーションステーション22は、この場合、転送された結果を機能ユニット24に送る。

【0031】

ある実施例では、各機能ユニット24A~Cは、加算および減算の整数算術演算、ならびにシフト、循環、論理演算および分岐操作を実行するよう構成される。なお、浮動小数点ユニット(図示せず)もまた、浮動小数点演算を可能にするために採用されてもよい。浮動小数点ユニットは、コプロセッサとして動作してよく、MROMユニット34から命令を受取りその後リオーダバッファ32と通信して命令を完了する。加えて、機能ユニット24は、ロード/ストアユニット26によって実行されるロードメモリ操作およびストアメモリ操作のためのアドレス生成を行なうよう構成され得る。

【0032】

機能ユニット24の各々はまた、条件付き分岐命令の実行に関する情報を分岐予測ユニット14に与える。分岐予測が正しくなかった場合、分岐予測ユニット14は、命令処理パイプラインに入った予測誤りされた分岐の後の命令をフラッシュ(flush)し、命令キャッシュ16または主メモリから要求される命令のフェッチを起こさせる。なお、そのような状況においては、投機的に実行され、一時的にロード/ストアユニット26およびリオーダバッファ32中に記憶されたものを含む、予測誤りされた分岐命令の後に起こる元々のプログラムシーケンス中の命令の結果は廃棄される。

【0033】

機能ユニット24によって発生された結果は、レジスタ値が更新されている場合にはリオーダバッファ32に送られ、メモリ場所の内容が変更される場合にはロード/ストアユニット26に送られる。結果がレジスタに記憶されるべきであれば、リオーダバッファ32は、命令がデコードされたときレジスタの値のために確保された場所に結果を記憶する。

10

20

30

40

50

複数個の結果バス38が、機能ユニット24およびロード/ストアユニット26からの結果の転送のために含まれる。結果バス38は、生成された結果、および実行されている命令を識別するリオーダーバッファタグを送る。

【0034】

ロード/ストアユニット26は、機能ユニット24とデータキャッシュ28との間にインターフェイスを与える。ある実施例では、ロード/ストアユニット26は、ロードまたはストアをペンディングするためにデータおよびアドレス情報のための8個の記憶場所を有するロード/ストアバッファで構成される。バッファが一杯になると、命令整列ユニット18は、ロード/ストアユニット26がペンディングのロードまたはストア要求情報のための空きを有するまで待つ。ロード/ストアユニット26はまた、ペンディングのストアメモリ操作に対してロードメモリ操作の依存性チェックを行ない、確実にデータコヒーレンスを維持する。メモリ操作は、マイクロプロセッサ10と主メモリサブシステムとの間のデータの転送である。メモリ操作は、メモリに記憶されるオペランドを利用する命令の結果であってもよく、またはデータ転送をもたらすが他の操作をもたらさないロード/ストア命令の結果であってもよい。加えて、ロード/ストアユニット26は、x86マイクロプロセッサアーキテクチャによって定義されるアドレス変換メカニズムに関連するセグメントレジスタおよび他のレジスタなどの特殊レジスタについての特殊レジスタ記憶装置を含み得る。

10

【0035】

ある実施例では、ロード/ストアユニット26は、ロードメモリ操作を投機的に実行するよう構成される。ストアメモリ操作は、プログラム順序で実行可能であるが、予測されたウェイに投機的に記憶されてもよい。予測されたウェイが正しくなければ、ストアメモリ操作の前のデータは、予測されたウェイにその後リストアされ、ストアメモリ操作は、正しいウェイに実行される。別の実施例では、ストアはまた投機的に実行されてもよい。投機的に実行されたストアは、更新より前にキャッシュラインのコピーとともにストアバッファに置かれる。投機的に実行されたストアが分岐予測誤りまたは例外のために後に廃棄されるならば、キャッシュラインは、バッファ中に記憶された値にリストアされ得る。なお、ロード/ストアユニット26は、投機的実行なしを含む、いかなる量の投機的実行を行なうよう、構成されてもよい。

20

【0036】

データキャッシュ28は、ロード/ストアユニット26と主メモリサブシステムとの間で転送されるデータを一時的に記憶するために設けられる高速キャッシュメモリである。ある実施例では、データキャッシュ28は、8ウェイのセットアソシアティブ構造で最大16キロバイトのデータを記憶する容量を有する。命令キャッシュ16と同様に、データキャッシュ28は、ウェイ予測メカニズムを採用し得る。データキャッシュ28は、セットアソシアティブ構成およびダイレクトマップド構成を含む、さまざまな特殊メモリ構成で実現され得る。

30

【0037】

x86マイクロプロセッサアーキテクチャを採用するマイクロプロセッサ10のある特定の実施例においては、命令キャッシュ16およびデータキャッシュ28は、線形にアドレスされる。線形アドレスは、命令によって特定されるオフセットと、x86アドレス変換メカニズムのセグメント部分によって特定されるベースアドレスとから形成される。線形アドレスは、オプションで、主メモリにアクセスするために物理的地址に変換されてもよい。線形から物理的への変換は、x86アドレス変換メカニズムのページング部分によって特定される。なお、線形アドレスされたキャッシュは、線形アドレスタグを記憶する。1組の物理的タグ(図示せず)を採用して、線形アドレスを物理的地址にマッピングし変換アライアスを検出してもよい。加えて、物理的タグブロックは、線形から物理的への変換を実行し得る。

40

【0038】

プリフェッチ/プリデコードユニットおよび命令キャッシュ構成

50

先に記したように、たとえばインテル (Intel) の 8 2 4 9 1 / 8 2 4 9 2 キャッシュ S R A M などの、マイクロプロセッサのための外部キャッシュとして使用されるよう設計された多くのメモリデバイスは、パリティ情報を記憶するよう構成可能である。ある実施例では、1 パリティビットは、8 データビットごとに記憶され得る。偶数パリティが所望されると想定すると、 $0 1 1 0 1 0 1 1_2$ のデータバイトは、アサートされたビットの総数が偶数になるように、アサートされたパリティビットを有するであろう。パリティビットは、マイクロプロセッサによって生成され、次にデータバイトとともに外部キャッシュ中に記憶され得る。マイクロプロセッサがキャッシュからデータをリードバックすると、それはアサートされたデータおよびパリティビットを数えるであろう。もし結果として得られた値が選択されたパリティに一致していなければ、パリティエラーが発生してしまっており、マイクロプロセッサは、たとえばオペレーティングシステムにメモリエラーの発生の信号を与えるなどの、適切な行動を取り得る。他のメモリデバイス構成が、エラーチェックおよび訂正 (E C C) のために付加的ビットを割当ててもよい。

10

【 0 0 3 9 】

サーバなどのハイエンドシステムは、典型的にパリティおよび E C C をサポートするが、多くのローおよび中間システム設計者は、データエラー発生の確率が比較的低いためにこれらの特徴を起動することを選択しない。これらのシステムでは、レベル 2 キャッシュ中のパリティビットおよび E C C ビットは、プリデコード情報を記憶するために用いられるかもしれない。これは、有利には、システムレベルでの広範なハードウェア変更を必要とすることなしに性能を向上させることが可能である。

20

【 0 0 4 0 】

次に図 4 を参照すると、プリデコードユニット 1 2 および命令キャッシュ 1 6 のある実施例の詳細を示す図が示される。この実施例では、プリデコードユニット 1 2 は、バスインターフェイス論理 5 2 を介してキャッシュ 5 0 に結合される。キャッシュ 5 0 は、マイクロプロセッサ 1 0 と同じシリコン上か、またはたとえばマイクロプロセッサ 1 0 の近くのマザーボードもしくはドーターカードの上などの近くに結合される別個のシリコン上のいずれかに存在するローレイテンシの広帯域幅メモリを含む。キャッシュ 5 0 は、スタティックランダムアクセスメモリ (S R A M)、シンクロナスダイナミックアクセスメモリ (S D R A M) またはローレイテンシメモリの他のタイプを含み得る。キャッシュ 5 0 は、マイクロプロセッサ 1 0 と同じシリコン上、または別個のシリコン上に存在してもよい。あるいくつかの実施例においては、キャッシュ 5 0 は、マイクロプロセッサ 1 0 の機能ユニットに 2 番目に近いキャッシュ、すなわちレベル 1 命令キャッシュ 1 6 およびデータキャッシュ 2 8 の後にあるので、「レベル 2」キャッシュと呼ばれることがある。キャッシュ 5 0 はまた、マイクロプロセッサの外部にあるので、「外部キャッシュ」と呼ばれることもある。

30

【 0 0 4 1 】

バスインターフェイス論理 5 2 は、マルチプレクサ、バッファ、トランシーバ、ドライバまたは、マイクロプロセッサ 1 0 とキャッシュ 5 0 との間にデータ信号、アドレス信号および制御信号の伝送を可能にするかまたは向上させ得るいかなる他の種類のバスインターフェイス論理であってもよい。いくつかの実施例では、バスインターフェイス論理 5 2 は、マイクロプロセッサ 1 0 とレベル 2 キャッシュ 5 0 との間に必要とされないことがある。たとえば、マイクロプロセッサ 1 0 およびレベル 2 キャッシュ 5 0 が互いに物理的に十分に近く、かつその出力トランジスタの駆動能力が十分に高ければ、マイクロプロセッサ 1 0 およびキャッシュ 5 0 は、インターフェイス論理 5 2 なしに互いに結合可能である。

40

【 0 0 4 2 】

プリデコード論理 1 2 は、分岐予測ユニット 1 4 からプリフェッチされるべき命令アドレスを受取り、これをバス 6 8、バスインターフェイス論理 5 2 およびメモリバス 5 6 を介してキャッシュ 5 0 に送るよう構成される。レベル 2 キャッシュ 5 0 が要求されたアドレスに対応する命令バイトを記憶している場合、命令バイトの予め定められた数 (たとえば、3 2 バイトの 1 キャッシュライン) が、バス 5 6、バスインターフェイス論理 5 2 およ

50

びバス68を介してプリデコードユニット12に送られる。要求された命令バイトがレベル2キャッシュ50内に記憶されていない場合、キャッシュ50は、要求された命令バイトを主メモリサブシステムから取出すよう構成される。一旦要求された命令バイトが主メモリサブシステムから送られると、それらは命令キャッシュ16に送られる。それらはキャッシュ50にも記憶され得る。

【0043】

プリデコードユニット12が要求された命令バイトを受取ると、それは各命令バイトについてのプリデコード情報を生成する。図に示される実施例では、プリデコードユニット12は、各命令バイトごとに1開始ビットを生成する。プリデコードユニット12が要求されたアドレスを出力することに応答して32バイト(たとえば、1キャッシュライン)を受取ると想定すると、プリデコードユニット12は、合計32開始ビットについて各命令バイトごとに1開始ビットを生成するよう構成され得る。生成されると、開始ビットは命令キャッシュ16に送られ、ここでそれらはその関連付けられる命令バイトとともに記憶される。命令キャッシュ16は、1プリデコード情報記憶場所が各キャッシュライン記憶場所に対応するように、複数個のキャッシュライン記憶場所64および複数個のプリデコード情報記憶場所62に局所的に編成され得る。命令バイトおよびプリデコードビットはまた、命令整列ユニット18に直接送られ得る。

10

【0044】

命令キャッシュ16内のキャッシュライン記憶場所64がすべて一杯になると、多数の異なるアルゴリズムを使用してどのキャッシュラインがキャッシュミスの際に置換されるべきかを決定可能である。たとえば、近い将来に必要とされるであろう情報を捨てる可能性を低減するために、キャッシュアクセスのオーダを記録するリスト・リーセントリ・ユーズド(LRU)置換方式を使用してもよい。上書されるべきキャッシュラインが選択されると、対応するプリデコード記憶場所62中に記憶される関連付けられたプリデコード情報は、プリデコードユニット12およびバスインターフェイス論理52経由で命令キャッシュ16からキャッシュ50に出力される。プリデコードユニット12はまた、上書される命令キャッシュ情報に対応するアドレスを送る。

20

【0045】

キャッシュ50がプリフェッチユニット12および命令キャッシュ16からプリデコードビットおよび対応するアドレスを受取ると、キャッシュ50は、対応するアドレスと関連付けられたパリティビット記憶場所60にプリデコードビットを記憶するよう構成される。いくつかの実施例では、命令キャッシュ16で上書されるべき実際の命令バイトもまたキャッシュ50に出力されプリデコードビットとともに記憶され得る。この構成は、実行変更コードをサポートするマイクロプロセッサ10のいくつかの実施例において有利である可能性がある。たとえば、ストアアドレスが命令キャッシュ16によって「詮索」(snop)され、ストア命令が命令キャッシュ16内に記憶される命令バイトを上書するかどうかを決定し得る。もしそうであれば、ストア命令が命令キャッシュ16およびデータキャッシュ28に実行され、これによって所望の命令バイトを変更することが可能である。そのような構成では、プリデコードユニット12は、変更された命令についてのプリデコード情報を再計算し、次に、再計算されたプリデコードビットを命令キャッシュ16に記憶するよう構成され得る。キャッシュ50がライトバックキャッシュとして構成されるならば、変更された命令バイトは、変更された命令バイトを記憶するキャッシュライン記憶場所が命令キャッシュ16中で上書されるとき後にキャッシュ50に書込まれるだろう。

30

40

【0046】

プリデコード情報(および、もしそのように構成されているならば命令バイト)がキャッシュ50に書込まれた後、命令キャッシュ16内のキャッシュライン記憶場所およびプリデコード情報記憶場所は、新しい命令バイトおよび新しいプリデコード情報で無事に上書され得る。プリデコードユニット12がキャッシュ50に出力された命令バイトのためのプリフェッチまたはプリフェッチ要求を受取ると、プリデコードユニット12は、要求されたアドレスをキャッシュ50に出力する。キャッシュ50がまだ、対応する命令バイト

50

およびプリデコード情報を記憶している（すなわち、それらが置換されていない）ならば、キャッシュ50は、プリデコードユニット12にそれらを送るよう構成され得る。プリデコードユニット12は、命令バイトおよび対応するプリデコード情報を命令整列ユニット18および命令キャッシュ16に転送し得る。有利には、この実施例では、要求された命令バイトについての新しいプリデコード情報を生成するプロセスは、回避可能である。先に記したように、これは、分岐予測誤りおよびまたはキャッシュミスが生じ機能ユニット24A～Cがストールする危険があるとき、マイクロプロセッサ10のいくつかの実施例において特に有利であるだろう。

【0047】

他方で、キャッシュ50が所望の命令バイトを別のアドレスに位置する他の命令バイトと置換していれば、キャッシュ50は、主メモリから要求されたバイトを抽出しそれらをプリデコードユニット12に送るよう構成され得る。キャッシュ50は、プリデコードユニット12に命令バイトを伝送するとき特定の制御信号をアサートして命令キャッシュ50が伝送される命令バイトについての有効プリデコード情報を有していないことを示すよう構成され得る。

10

【0048】

別の実施例では、制御ユニット80は、キャッシュ50をモニタし、主メモリからの新しい命令バイトで置換されている命令記憶場所58に対応するパリティ/ECC記憶場所60にプリデコードビットの特定の無効シーケンスを記憶する。これは、新しい命令バイトについてのプリデコード情報を有効に「初期化」する。この構成では、キャッシュ50は、対応する命令バイトがプリデコードユニット12によって要求されるときプリデコードビットの無効シーケンスを出力するよう構成可能である。有利には、この構成では、キャッシュ50は、パリティ/ECCをサポートする標準のキャッシュであり得、それが出力しているプリデコードビットが有効であるか無効であるかを認識するように変更される必要はない。代わりに、プリデコードユニット12が、それがキャッシュ50から受取るプリデコードビットを調べるよう構成され得る。プリデコードビットが予め定められた無効シーケンスと一致すれば、プリフェッチユニット12は、新しいプリデコード情報を計算するよう構成され得る。プリデコードビットが予め定められた無効シーケンスと一致しなければ、プリフェッチユニット12は、プリデコードビットをレベル2キャッシュ50から命令キャッシュ16（および、もし必要であれば整列ユニット16/デコードユニット20A～C）に転送するよう構成され得る。

20

30

【0049】

なお、バス56および68は、いかなる都合のよいサイズ、たとえば16、32、64または128ビットであってもよい。使用されるバスラインの数は、データバス、アドレスバスおよび制御バスを多重化することによって低減可能である。いくつかの実施例では、プリデコードユニット12およびレベル2キャッシュ50は、送られ受取られるアドレスについてのパリティビットを生成するよう構成され得る。しかしながら、これらのパリティビットは、記憶される必要がなく、各アドレスが送られた後に廃棄可能である。

【0050】

さらに別の実施例では、開始ビットに加えてプリデコード情報がプリデコードユニット12によって生成され、命令キャッシュ16に記憶され得る。たとえば、その関連付けられた命令バイトが命令の最終バイトであるかどうかを識別する終了ビットと、その関連付けられた命令バイトが操作コードバイトであるかどうかを示す操作コードバイトとがまた、プリデコードユニット12によって生成され、命令キャッシュ16に記憶されてもよい。これらのプリデコードビットのいくつかまたはすべては、キャッシュ50内で利用可能なパリティおよびまたはECCビット60の数に依存して、命令バイトがキャッシュ16中で置換されるときキャッシュ50に送られるだろう。

40

【0051】

いくつかの実施例では、デコードユニット20A～Cは、プリデコードユニット12によって与えられるかまたは命令キャッシュ16から読出されたプリデコード情報が正しくな

50

いときを検出するよう構成され得る。プリデコード情報の不正確さは、プリデコード情報を発生するために用いられる方法に依存して異なった理由のために起こり得る。不正確なプリデコード情報の場合には、デコードユニット20A~Cは、ストールするよう構成されるかもしれないが、その一方でプリデコードユニット12はプリデコード情報を再生成する。新しいプリデコード情報は次に、不正確なプリデコード情報の上から命令キャッシュ16に書込まれるであろう。別の実施例では、デコードユニット20A~Cは、不正確なプリデコード情報を単に廃棄し自らデコードを完了するよう構成され得る。

【0052】

次に図5を参照すると、命令キャッシュ16およびキャッシュ50のある実施例のさらなる詳細が示される。先に記載したとおり、命令キャッシュ16は、命令記憶場所62およびプリデコード情報記憶場所64で構成され得る。各プリデコード情報記憶場所は、1命令記憶場所に対応する。たとえば、記憶場所80に記憶されたプリデコード情報は、命令記憶場所82に記憶された命令バイトに対応する。図に示す実施例では、記憶場所80は、場所82に記憶された各命令バイトに対応する開始ビットを記憶する。

10

【0053】

図に示すとおり、キャッシュ50は、命令キャッシュ16と同じ命令バイトを記憶可能である。たとえば、コードセグメント72からの命令が初めて要求されたとき、それらは主メモリからレベル2キャッシュ50および命令キャッシュ16に読出され得る。しかしなお、レベル2キャッシュ50は、コードセグメント72についてプリデコードデータを有していないであろう。したがって、予め定められた無効定数(この場合には、00000000...)が場所90に記憶される。プリデコードユニット12が命令バイトを受取ると、それは無効定数を検出し、コードセグメント72についてのプリデコード情報を計算し始める。このプリデコード情報は次に、命令キャッシュ16内の記憶場所80に記憶される。

20

【0054】

同じプロセスが、コードセグメント78の命令が初めて要求される時起こるのである。しかしながら、コードセグメント78についての命令バイトが一旦命令キャッシュ16中で置換されると、プリデコード情報はレベル2キャッシュ50にライトバックされ場所94に記憶される。コードセグメント78が、それが命令キャッシュ16中で置換された後におよびそれがレベル2キャッシュ50中で置換されるより前に再び要求される場合、コードセグメント78についての命令バイトおよびプリデコード情報の両方が、場所94~96から読出され命令キャッシュ16中に記憶されるであろう。この状態が図に示される。

30

【0055】

ビットの任意の予め定められたシーケンスが無効定数として用いられ得るが、いくつかの実施例においては、プリデコードユニット12によって生成される可能性の低いプリデコードビットシーケンスを選択することが有利であろう。たとえば、16バイトの最大命令長および32バイトのキャッシュライン長さを想定すると、プリデコードユニット12が少なくとも1つの開始バイトなしに32の連続する命令バイトを有するキャッシュラインを受取る可能性は低い。したがって、32個のゼロを含む無効定数を選択することが、それが誤った無効の数を低減し得るために、有利であろう。誤った無効は、プリデコードユニット12によって計算されたプリデコードビットが予め定められた無効定数に等しいときに起こり得る。もしこれが起これば、元々のプリデコードビットが正しかったとしても、キャッシュ50から読出された元々のプリデコードビットは廃棄され新しいプリデコードビットが計算されるであろう。

40

【0056】

ある実施例では、上述した特徴は、単独にマイクロプロセッサ10内で実現可能である。たとえば制御ユニット80の機能は省かれてもよいまたはマイクロプロセッサ10内に含まれてもよい。同様に、バスインターフェイス論理52は、いくつかの実施例では、省かれてもよいまたはマイクロプロセッサ10内に含まれてもよい。記載された機能全体

50

をマイクロプロセッサ 10 内に実現することは、有利には、開示されたようなプリデコード情報を記憶するコンピュータシステムを実現するコストを削減可能である。他の実施例では、所望の機能は、マイクロプロセッサ 10 と 1 つ以上のインターフェイスまたはサポータチップとの間に分散されてもよい。

【0057】

次に図 6 を参照すると、レベル 2 キャッシュ 50 内にプリデコード情報を記憶するための方法のある実施例を示すフローチャートが示される。まず、命令バイトが命令キャッシュ 16 から要求される (ステップ 140)。要求されたバイトが命令キャッシュ 16 内に記憶されているならば、それらは整列ユニット 16 およびデコードユニット 20A ~ C に、その対応するプリデコード情報とともに送られる (ステップ 142 およびステップ 152)。他方で、要求された命令バイトが命令キャッシュ 16 内に記憶されていないならば、要求されたアドレスはレベル 2 キャッシュ 50 に転送される (ステップ 142 およびステップ 144)。要求された命令バイトがレベル 2 キャッシュ 50 内に記憶されていないならば、それらは主メモリサブシステムから読出される (ステップ 146 およびステップ 148)。要求された命令バイトが主メモリサブシステムから受取られると、それらは無効定数とともにレベル 2 キャッシュ 50 内に記憶される (ステップ 148 およびステップ 150)。先に記したように、無効定数は、レベル 2 キャッシュ 50 内のプリデコード記憶場所を有効に初期化し、これによってプリデコードユニット 12 に新しいプリデコード情報が生成されるべきであるという信号を与える。プリデコードユニット 12 が命令バイトおよびプリデコードビットを受取ると、プリデコードビットは調べられ、それらが無効定数に等しいかどうかを決定する (ステップ 154 およびステップ 156)。プリデコードビットが無効定数に等しければ、プリデコードユニット 12 は、新しいプリデコード情報を生成するよう構成される (ステップ 156 およびステップ 164)。命令バイトおよび新しいプリデコード情報は次に、命令キャッシュ 16 に記憶される (ステップ 158)。命令バイトおよび新しいプリデコード情報を記憶することが命令キャッシュ 16 内で先に記憶されたキャッシュラインを置換することを伴うならば、上書きされる先に記憶されたキャッシュラインに対応するプリデコード情報は、レベル 2 キャッシュ 50 にストアバックされる (ステップ 160 およびステップ 166)。

【0058】

なお、図のステップは、説明の目的のためにのみ直列の態様で表わされる。他の実施例が可能であり企図される。上記の多数のステップは、組合されてもよくまたは並列に行なわれてもよい。たとえば、ステップ 150 および 154 は、並列に行なわれてもよい。さらに、先に記したように、無効定数の使用を伴うステップは、オプションである。

【0059】

例示のコンピュータシステム

図 7 を参照すると、マイクロプロセッサ 10 を利用するコンピュータシステム 200 のある実施例のブロック図が示される。描写されるシステムでは、主メモリ 204 が、メモリバス 206 を介してバスブリッジ 202 に結合され、グラフィックスコントローラ 208 が、AGP バス 210 を介してバスブリッジ 202 に結合される。最後に、複数個の PCI デバイス 212A ~ 212B が、PCI バス 214 を介してバスブリッジ 202 に結合される。2 次バスブリッジ 216 をさらに設けて、EISA / ISA バス 220 を介して 1 つ以上の EISA または ISA デバイス 218 への電気的インターフェイスを可能としてもよい。マイクロプロセッサ 10 は、CPU バス 224 を介してバスブリッジ 202 に結合される。

【0060】

バスブリッジ 202 は、マイクロプロセッサ 10 と、主メモリ 204 と、グラフィックスコントローラ 208 と、PCI バス 214 につながるデバイスとの間のインタフェースを与える。操作がバスブリッジ 202 に接続されるデバイスの 1 つから受取られると、バスブリッジ 202 は、操作のターゲット (たとえば、特定のデバイスまたは、PCI バス 214 の場合には、ターゲットは PCI バス 214 上にある) を特定する。バスブリッジ

202は、ターゲットされたデバイスに操作を送る。バスブリッジ202は一般的には、ソースデバイスまたはバスによって使用されるプロトコルからターゲットデバイスまたはバスによって使用されるプロトコルに操作を変換する。

【0061】

PCIバス214のためにISA/EISAバスへのインタフェースを与えることに加えて、2次バスブリッジ216はさらに、所望のごとくさらなる機能を組込んでよい。たとえば、ある実施例では、2次バスブリッジ216は、PCIバス214の所有権を調停するためのマスタPCIアービタ(図示せず)を含む。2次バスブリッジ216の外部またはこれと統合される入力/出力コントローラ(図示せず)をもコンピュータシステム200内に含めて、所望のごとく、キーボードおよびマウス222ならびにさまざまな直列ポートおよび並列ポートのための動作上のサポートを与えてもよい。他の実施例では、外部キャッシュユニット(図示せず)が、マイクロプロセッサ10とバスブリッジ202との間のCPUバス224にさらに結合されてもよい。代替的に、外部キャッシュは、バスブリッジ202に結合されてもよく、外部キャッシュのためのキャッシュ制御論理は、バスブリッジ202に統合されてもよい。

10

【0062】

主メモリ204は、アプリケーションプログラムが記憶され、これからマイクロプロセッサ10が主に実行するメモリである。好適な主メモリ14は、DRAM(ダイナミックランダムアクセスメモリ)および、好ましくはSDRAM(シンクロナスDRAM)の複数個のバンクを含む。

20

【0063】

PCIデバイス212A~212Bは、たとえば、ネットワークインターフェイスカード、ビデオアクセラレータ、オーディオカード、ハードもしくはフロッピーディスクドライブまたはドライブコントローラ、SCSI(スモールコンピュータシステムインターフェイス)アダプタおよび電話機能カードなどの、さまざまな周辺デバイスを例示するものである。同様に、ISAデバイス218は、モデム、サウンドカードおよび、GPIBまたはフィールドバスインターフェイスカードなどのさまざまなデータ収集カードなどの、さまざまなタイプの周辺デバイスを例示するものである。

【0064】

グラフィックスコントローラ208は、ディスプレイ226上のテキストおよび画像のレンダリングを制御するために設けられる。グラフィックスコントローラ208は、先行技術に一般的に公知の典型的なグラフィックスアクセラレータを採用して、主メモリ204におよびそこから効果的にシフト可能である3次元のデータ構造をレンダリングすることが可能である。したがって、グラフィックスコントローラ208は、バスブリッジ202内のターゲットインターフェイスへのアクセスを要求しかつ受取ることによって主メモリ204へのアクセスを獲得することが可能であるという点で、AGPバス210のマスタであり得る。専用のグラフィックスバスが、主メモリ204からのデータの高速の取出しを可能とする。ある操作では、グラフィックスコントローラ208は、AGPバス210のPCIプロトコルトランザクションを生成するようさらに構成されてもよい。したがって、バスブリッジ202のAGPインターフェイスは、AGPプロトコルトランザクションと、PCIプロトコルターゲットおよびイニシエータトランザクションとの両方をサポートする機能を含み得る。ディスプレイ226は、画像またはテキストが与えられることが可能である任意の電子ディスプレイである。好適なディスプレイ226は、陰極線管(「CRT」)、液晶ディスプレイ(「LCD」)などを含む。

30

40

【0065】

なお、AGPバス、PCIバスおよびISAバスまたはEISAバスが、上記の記載では例として用いられたが、いかなるバスアーキテクチャが所望のごとく置換えられてもよい。さらになお、コンピュータシステム200は、さらなるマイクロプロセッサを含むマルチプロセッシングコンピュータシステムであってもよい。

【0066】

50

さらになお、この記載は、さまざまな信号のアサートを参照するであろう。信号は、それが特定の条件を示す値を送る場合に「アサートされる」とここでは言う。反対に、信号はそれが特定の条件の不在を示す値を送る場合には、「デアサートされる」または「アサートされない」。信号は、それが論理0値を送るとき、または反対に、それが論理1値を送るときにアサートされるよう定義され得る。加えて、さまざまな値が、上記の記載では廃棄されるものとして記載された。値は、多くの態様で廃棄可能であるが、一般的には、それが値を受取る論理回路によって無視されるように値を変更することを伴なう。たとえば、値が1ビットを含む場合、値の論理状態は、値を廃棄するよう反転されるであろう。値がnビットの値である場合、nビットの符号化の1つが、値が無効であることを示すであろう。値を無効の符号化にセットすると、値は廃棄される。加えて、nビット値は、セッ

10

【0067】

産業上の用途

この発明は、マイクロプロセッサ、コンピュータシステム、パーソナルデジタルアシスタント(PDA)、コンピュータネットワーク、単一および複数のプロセッサシステム、オンチップのシステム、埋込まれたプロセッサおよびマイクロコントローラシステム、ならびにデジタル信号プロセッサを含む多数の異なる産業分野に適用可能であるが、これに限定されるものではない。

【図面の簡単な説明】

20

【図1】 一般的なx86命令フォーマットのブロック図である。

【図2】 有効マスクのある実施例を示すブロック図である。

【図3】 マイクロプロセッサのある実施例のブロック図である。

【図4】 図3の命令キャッシュとレベル2キャッシュとの間のインターフェイスのある実施例の詳細を示す図である。

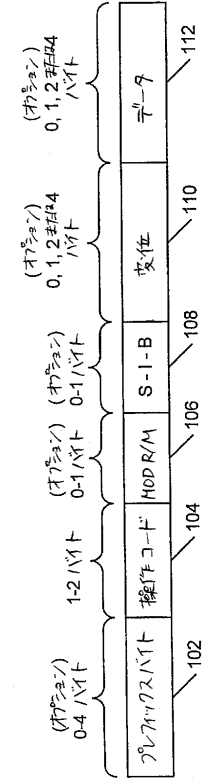
【図5】 図4に表わされる命令キャッシュのある実施例と図4からのレベル2キャッシュのある実施例との関係の詳細を示す図である。

【図6】 プリデコード情報をレベル2キャッシュ中に記憶するための方法のある実施例を表わすフローチャートである。

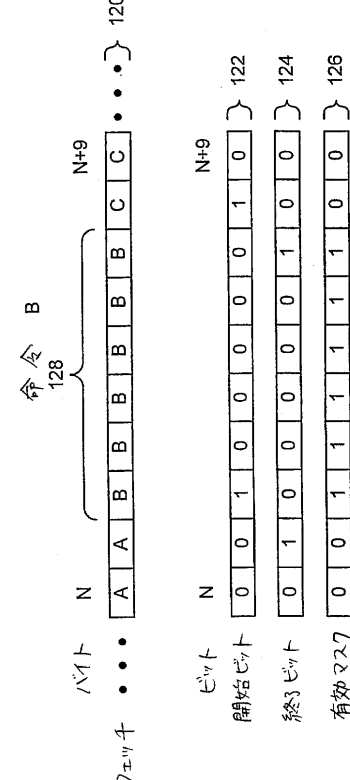
【図7】 図3のマイクロプロセッサを利用するコンピュータシステムのある実施例を示す図である。

30

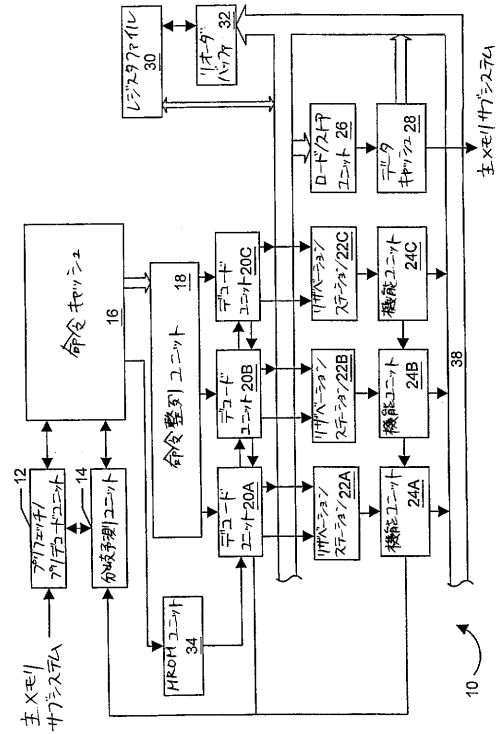
【 図 1 】



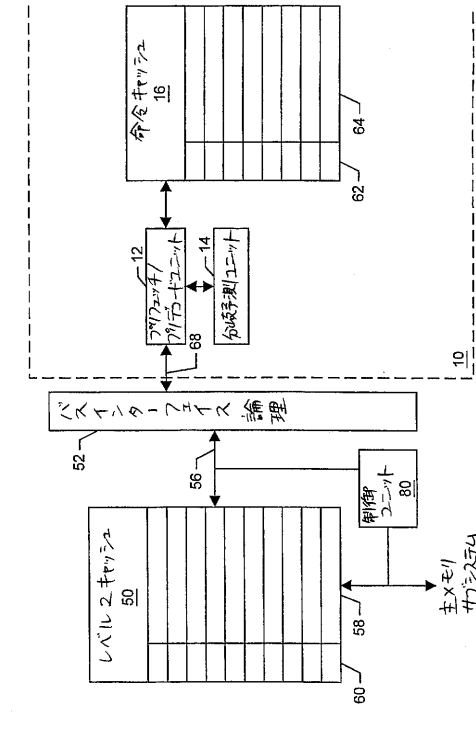
【 図 2 】



【 図 3 】



【 図 4 】



フロントページの続き

(74)代理人 100083703

弁理士 仲村 義平

(74)代理人 100091409

弁理士 伊藤 英彦

(74)代理人 100096781

弁理士 堀井 豊

(74)代理人 100096792

弁理士 森下 八郎

(72)発明者 マハリンガイア, ルパカ

アメリカ合衆国、78749 テキサス州、オースティン、ファームデイル・レーン、6503

審査官 後藤 彰

(56)参考文献 特表2000-515274(JP, A)

特開平10-55276(JP, A)

特開平5-73415(JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 9/38

G06F 12/08