

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3798404号

(P3798404)

(45) 発行日 平成18年7月19日(2006.7.19)

(24) 登録日 平成18年4月28日(2006.4.28)

(51) Int. Cl.

G06F 9/38 (2006.01)

F I

G06F 9/38 330B

請求項の数 17 (全 37 頁)

(21) 出願番号	特願2003-521935 (P2003-521935)	(73) 特許権者	591016172
(86) (22) 出願日	平成14年6月27日(2002.6.27)		アドバンスト・マイクロ・デバイス・
(65) 公表番号	特表2005-500616 (P2005-500616A)		インコーポレイテッド
(43) 公表日	平成17年1月6日(2005.1.6)		ADVANCED MICRO DEVI
(86) 国際出願番号	PCT/US2002/020481		CES INCORPORATED
(87) 国際公開番号	W02003/017091		アメリカ合衆国、94088-3453
(87) 国際公開日	平成15年2月27日(2003.2.27)		カリフォルニア州、サニibel、ピィ・
審査請求日	平成17年5月24日(2005.5.24)		オウ・ボックス・3453、ワン・エイ・
(31) 優先権主張番号	09/912,011		エム・ディ・プレイス、メイル・ストップ
(32) 優先日	平成13年7月24日(2001.7.24)		・68 (番地なし)
(33) 優先権主張国	米国 (US)	(74) 代理人	100099324
			弁理士 鈴木 正剛
		(74) 代理人	100111615
			弁理士 佐野 良太

最終頁に続く

(54) 【発明の名称】 2レベルの分岐予測キャッシュによる分岐予測

(57) 【特許請求の範囲】

【請求項1】

1次レベルキャッシュ(206)が第1のアドレスに対応する第1分岐予測情報(2100)を有さないことを検出するステップと、

2次レベルキャッシュ(260, 1828)が、前記第1のアドレスに対応し、前記第1分岐予測情報の一部を含む第2分岐予測情報(2120, 2122)を有するかどうかを判定するステップと、

前記2次レベルキャッシュが前記第2分岐予測情報を有することが検出されると、前記第1の分岐予測情報を再生成するステップと、

分岐命令のタイプを示す第3分岐予測情報を生成するステップと、

前記第2分岐予測情報と、前記第3分岐予測情報とを組み合わせるステップと、

前記第1のアドレスに対応する、前記1次レベルキャッシュの第1のエントリに前記第1の分岐予測情報として、前記組み合わせられた第2および第3分岐予測情報をストアするステップとを含む方法。

【請求項2】

前記1次レベルキャッシュ(206)の前記第1のエントリ(2100)が使用可能であるかどうかを判定するステップと、

前記第1のエントリが使用不可能であることが検出されると、前記第1のエントリ(2100)の内容を排除するステップと、

前記排除を受けて、前記内容の一部(2120, 2122)を前記2次レベルキャッシ

10

20

ユ (2 6 0 , 1 8 2 8) にストアするステップと、をさらに含む請求項 1 記載の方法。

【請求項 3】

前記生成するステップは、前記第 1 アドレスに対応するプログラム命令を受信し、デコードするステップを含む請求項 1 または 2 記載の方法。

【請求項 4】

分岐予測情報 (2 1 0 0) をストアするように構成された 1 次レベルキャッシュ (2 0 6) と、

前記分岐予測情報の一部 (2 1 2 0 , 2 1 2 2) をストアするように構成された 2 次レベルキャッシュ (2 6 0) と、

前記 1 次レベルキャッシュ (2 0 6) および前記 2 次レベルキャッシュ (2 6 0 , 1 8 2 8) に結合された回路系 (4 0 0) とを備え、前記回路系は、

前記 1 次レベルキャッシュが第 1 のアドレスに対応する 第 1 分岐予測情報を有さないことを検出し、

前記 2 次レベルキャッシュが前記第 1 のアドレスに対応し、前記第 1 分岐予測情報の一部を含む第 2 分岐予測情報を有するかどうかを判定し、

前記 2 次レベルキャッシュが前記第 2 分岐予測情報を有することが検出されると、前記第 1 分岐予測情報を再生成するように構成され、前記回路系は、前記第 1 分岐予測情報を再生成するために、さらに、

分岐命令のタイプを示す第 3 分岐予測情報を生成し、

前記第 2 分岐予測情報と、前記第 3 分岐予測情報とを組み合わせ、

前記第 1 のアドレスに対応する、前記 1 次レベルキャッシュの第 1 のエントリに前記第 1 の分岐予測情報として、前記組み合わせられた第 2 および第 3 分岐予測情報をストアするように構成されている、分岐予測機構。

【請求項 5】

前記回路系 (4 0 0) は、

前記 1 次レベルキャッシュの前記第 1 のエントリが使用可能であるかどうかを判定し、

前記第 1 のエントリが使用不可能であることが検出されると、前記第 1 のエントリの内容を排除し、

前記排除を受けて、前記内容の一部を前記 2 次レベルキャッシュにストアするようにさらに構成されている請求項 4 記載の分岐予測機構。

【請求項 6】

前記第 3 分岐予測情報を生成するために、前記回路系は、前記第 1 アドレスに対応するプログラム命令を受信し、デコードするように構成されている、請求項 4 または 5 記載の分岐予測機構。

【請求項 7】

前記プログラム命令は、前記第 2 次レベルキャッシュ (2 6 0 , 1 8 2 8) からフェッチされる、請求項 6 記載の分岐予測機構。

【請求項 8】

相互接続 (2 6 2 4) と、

前記相互接続に結合されたメモリ (2 6 0 4) と、

分岐予測情報をストアするように構成された 2 次レベルキャッシュ (2 6 0 , 1 8 2 8) と、

1 次レベルキャッシュ (2 0 6) を備えたプロセッサ (1 0) とを備え、前記プロセッサは、

前記 1 次レベルキャッシュ (2 0 6) が第 1 のアドレスに対応する 第 1 分岐予測情報を有さないことを検出し、

前記 2 次レベルキャッシュ (2 6 0 , 1 8 2 8) が前記第 1 のアドレスに対応し、前記第 1 分岐予測情報の一部を含む第 2 分岐予測情報を有するかどうかを判定し、

前記 2 次レベルキャッシュが前記第 2 分岐予測情報を有することが検出されると、前記第 1 の分岐予測を再生成するように構成され、前記プロセッサは、前記第 1 分岐予測情報

10

20

30

40

50

を再生成するために、さらに、

分岐命令のタイプを示す第3分岐予測情報を生成し、

前記第2分岐予測情報と、前記第3分岐予測情報とを組み合わせ、

前記第1のアドレスに対応する、前記1次レベルキャッシュの第1のエントリ(2100)に前記第1の分岐予測として、前記組み合わせられた第2および第3分岐予測情報をストアするように構成されている、コンピュータシステム。

【請求項9】

前記プロセッサ(10)は、前記1次レベルキャッシュ(206)の前記第1のエントリ(2100)が使用可能であるかどうかを判定し、前記第1のエントリが使用不可能であることが検出されると、前記第1のエントリ(2100)の内容を排除し、前記排除を受けて、前記内容の一部を前記2次レベルキャッシュ(260, 1828)にストアするようにさらに構成されている請求項8記載のコンピュータシステム。

10

【請求項10】

前記2次レベルキャッシュ(260, 1828)および前記1次レベルキャッシュ(206)は重複する情報をストアしない、請求項1、2または3記載の方法。

【請求項11】

前記2次レベルキャッシュ(260, 1828)は、前記1次レベルキャッシュ(206)から排除された情報のみをストアするように構成されている、請求項10記載の方法

【請求項12】

前記第3分岐予測情報は、前記プログラム命令のなかの成立予測分岐の位置をさらに示す、請求項3記載の方法。

20

【請求項13】

前記2次レベルキャッシュ(260, 1828)および前記1次レベルキャッシュ(206)は重複する情報をストアしない、請求項4、5、6または7記載の分岐予測機構。

【請求項14】

前記2次レベルキャッシュ(260, 1828)は、前記1次レベルキャッシュ(206)から排除された情報のみをストアするように構成されている、請求項13記載の分岐予測機構。

【請求項15】

前記第3分岐予測情報は、前記プログラム命令のなかの成立予測分岐の位置をさらに示す、請求項6または7記載の分岐予測機構。

30

【請求項16】

前記2次レベルキャッシュ(260, 1828)および前記1次レベルキャッシュ(206)は重複する情報をストアしない、請求項8または9記載のコンピュータシステム。

【請求項17】

前記2次レベルキャッシュ(260, 1828)は、前記1次レベルキャッシュ(206)から排除された情報のみをストアするように構成されている、請求項16記載のコンピュータシステム。

【発明の詳細な説明】

40

【技術分野】

【0001】

本発明は、スーパースカラーマイクロプロセッサの分野に関するものであり、より詳細には、分岐予測のための方法およびメカニズム(機構)に関するものである。

【背景技術】

【0002】

IEEE Transactions on Computers, IEEE Inc, New York, US, vol. 42, no. 4, 1 April 1993, 396から412ページ, Perleberg CH et al, 題名"Branch Target Buffer Design and Optimization"(分岐ターゲットバッファの設計および最適化)には、マルチレベル分岐ターゲットバッファ(BTB)が説明されている。マルチレベルBTBの各レベ

50

ルには、エントリ毎に異なった量/タイプの情報が含まれており、エントリの数的およびエントリ毎の情報の量のよりよいバランスを達成することによって、性能の最大化を達成する。

スーパースカラーマイクロプロセッサでは、1クロックサイクルに複数の命令を実行し、その設計に合わせた最も短いクロックサイクルを選択することによって高性能が実現されている。本明細書で使用されるように、「クロックサイクル」との用語は、マイクロプロセッサ内の命令処理パイプラインの各種ステージに与えられる時間間隔を指す。記憶装置（例えばレジスタやアレイ）は、クロックサイクルに従って値を取得する。例えば、記憶装置が、クロックサイクルを定義しているクロック信号の立ち上がりまたは立ち下がりによって値を取得し得る。次に、記憶装置は、クロック信号の立ち上がりによって値を取得した場合は次の立ち上がりまで、足り下がりによって値を取得した場合は立ち下がりまでこの値をストアし得る。本明細書に使用する「命令処理パイプライン」との用語は、パイプライン方式での命令の処理に使用される論理回路を指す。パイプラインを、任意の数のステージに分割して、これらステージで命令処理の一部を実行することが可能であるが、一般に命令処理では、命令のフェッチ、命令のデコード、命令の実行、およびデスティネーションへの実行結果のストアが行われる。

【発明の開示】

【発明の概要】

【0003】

スーパースカラーマイクロプロセッサ（およびスーパーパイプライン方式のマイクロプロセッサ）の重要な機能に、分岐予測メカニズムがある。分岐予測メカニズムは、分岐命令の予測された行き先（分岐の成立 [taken] または不成立 [not-taken] ）を示し、これにより、分岐予測によって示される予測された命令の流れに含まれる後続の命令のフェッチを続行できるようになる。分岐命令とは、分岐命令の直後の命令から始まる命令の流れを示す連続するアドレスか、メモリ内の任意の場所から始まる命令の流れを示すターゲットアドレスか、の2つ以上のアドレスの1つから後続の命令をフェッチさせる命令のことである。無条件分岐命令では、常にターゲットアドレスに分岐するのに対し、条件付き分岐命令では、先行する命令の結果によって、連続アドレスかターゲットアドレスかのいずれかが選択され得る。予測された命令の流れに含まれる命令は、分岐命令の実行前に投機的に実行され得、いずれの場合であっても分岐命令の実行前に命令処理パイプラインに投入される。予測された命令の流れが正しければ、1クロックサイクルに実行される命令数が増え有利となる。しかし、予測された命令の流れが外れた（つまり、1つ以上の分岐命令が誤って予測された）場合には、その予測された命令の流れに含まれる命令が命令処理パイプラインから破棄され、1クロックサイクルに実行される命令数が低下する。

【0004】

分岐予測メカニズムが効果を発揮するには、予測された命令の流れができるだけ的中するように、分岐予測メカニズムの精度が極めて高くなければならない。過去の分岐の実行履歴を使用して、特定の分岐の挙動の精度を向上させることがよく行われる。通常、このような分岐予測履歴では、分岐命令に対応するデータを記憶域に保持しなければならない。分岐予測のデータが記憶域から排除されているかデータが失われている場合は、その分岐命令の実行履歴を後から再生成する必要がある。上記の問題に対する解決策の1つとして、分岐予測用の記憶域のサイズを増やすことが考えられる。しかし、分岐予測用の記憶域のサイズを増やすには、ゲート面積の大幅な拡張と、分岐予測メカニズムのサイズの拡張とが必要となることがある。このため、分岐の挙動に関する貴重なデータが失われるおそれがあり、再生成が必要となる。したがって、ゲート数および分岐予測メカニズムのサイズの大幅な拡張を必要としない、分岐予測の性能を向上させるためのメカニズムが求められている。

【0005】

上記の問題は、ここに記載するマイクロプロセッサおよび方法によって大部分が解決される。一実施形態においては、命令の組に対応する分岐予測情報をストアするように構成

10

20

30

40

50

された1次レベル分岐予測キャッシュが、プロセッサに設けられる。さらに、1次レベルキャッシュから排除された分岐予測情報をストアするために2次レベル分岐予測キャッシュが使用される。2次レベル分岐予測キャッシュは、1次レベルキャッシュから排除された情報の一部(subset)のみをストアするように構成されている。1次レベルキャッシュから排除され、2次レベルキャッシュにストアされない分岐予測情報は破棄される。1次レベルキャッシュでミスが発生すると、2次レベルキャッシュが、ミスに対応する分岐予測情報を格納しているかどうか判定される。対応する分岐予測情報が2次レベルキャッシュで検出された場合、2次レベルキャッシュから検出された分岐予測情報がフェッチされて、完全な分岐予測情報の生成に使用される。この予測情報は、のちに分岐予測の作成に使用され得る。一実施形態においては、2次レベルキャッシュからフェッチした命令のデコードを実行するデコード回路系が設けられ得る。2次レベルキャッシュからフェッチした命令のデコードが、完全な分岐予測情報の再生成の際に使用され得る。サイズの小さいキャッシュを使用して、1次レベル分岐予測キャッシュから排除された分岐予測情報をストアすることができ有利である。さらに、1次レベルキャッシュでミスが発生した場合に、2次レベルキャッシュにストアされているデータから、完全な分岐予測を迅速に再生成することができる。

10

【発明を実施するための最良の形態】

【0006】

本発明の他の目的ならびに利点は、添付の図面を参照しつつ下記に記載する本発明の詳細な説明を参照すれば明白となる。

20

【0007】

本発明は、様々に変形されたり代替形態を取り得るが、その特定の実施形態が、例として図中に図示され、かつ本明細書に詳細に記載される。しかし、図面および詳細な説明は、開示の実施形態に本発明を限定することを意図するものではなく、本発明が添付の特許請求の範囲によって規定される本発明の趣旨ならびに範囲に含まれる全ての変形例、均等物および代替例を含むことが意図にあることが理解されるべきである。

【0008】

プロセッサの概略

図1を参照すると、プロセッサ10の一実施形態のブロック図が示される。別の実施形態も可能であり、考慮される。プロセッサ10は、プリフェッチユニット12、分岐予測ユニット14、命令キャッシュ16、命令整列ユニット18、複数個のデコードユニット20A~20C、複数個のリザベーションステーション22A~22C、複数個の機能ユニット24A~24C、ロード/ストアユニット26、データキャッシュ28、レジスタファイル30、リオーダーバッファ32、MROMユニット34およびバスインターフェイスユニット37を備える。本明細書において、後に文字が付された参照符号によって参照される複数の要素は、参照符号のみによって集合的に参照される。例えば、デコードユニット20A~20Cは、デコードユニット20として集合的に参照される。

30

【0009】

プリフェッチユニット12は、バスインターフェイスユニット37から命令を受け取るために同ユニットに結合され、さらに命令キャッシュ16および分岐予測ユニット14にも結合されている。同様に、分岐予測ユニット14は命令キャッシュ16に結合されている。さらに、分岐予測ユニット14は、デコードユニット20および機能ユニット24に結合されている。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に結合されている。さらに、命令整列ユニット18は、デコードユニット20に結合されている。各デコードユニット20A~20Cは、ロード/ストアユニット26と、それぞれ対応するリザベーションステーション22A~22Cとに結合されている。リザベーションステーション22A~22Cは、それぞれ対応する機能ユニット24A~24Cにさらに結合されている。さらに、デコードユニット20とリザベーションステーション22とは、レジスタファイル30およびリオーダーバッファ32に結合されている。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

40

50

ーダーバッファ32とも結合されている。データキャッシュ28は、ロード/ストアユニット26とバスインターフェイスユニット37とに結合されている。バスインターフェイスユニット37は、L2キャッシュへのL2インターフェイスとバスとにさらに結合されている。最後に、MROMユニット34はデコードユニット20に結合されている。

【0010】

命令キャッシュ16は、命令をストアするために設けられた高速キャッシュメモリである。命令キャッシュ16から命令がフェッチされて、デコードユニット20にディスパッチされる。一実施形態においては、命令キャッシュ16は、64バイト(1バイトは8つのバイナリビットを含む)のラインを有する2ウェイアソシエーティブ構造に、最大64キロバイトの命令をストアできるように構成されている。別法として、他の任意の構成およびサイズを使用してもよい。例えば、命令キャッシュ16は、フルアソシエーティブ構成、セットアソシエーティブ構成、ダイレクトマップド構成のいずれでも実装可能であるという点が注目される。

10

【0011】

プリフェッチユニット12によって、命令が命令キャッシュ16にストアされる。プリフェッチ方式では、命令は、その命令が要求される前に命令キャッシュ16からプリフェッチされ得る。種々のプリフェッチ方式が、プリフェッチユニット12によって採用され得る。命令キャッシュからフェッチされた命令は、スキャナ/整列器(scanner/aligner)に渡される。命令は、初めてフェッチされる際には、プリデコードタグが付けられていない。この場合、スキャナ/整列器は、1クロックにつき4バイトをデコードユニット20に渡す。デコードユニット20がプリデコードされていない命令をコアにディスパッチすると、デコードユニットは、この命令に対応するプリデコードデータを生成し得る。このデータは命令の境界を示すものである。

20

【0012】

次に、可変バイト長の命令セットを使用するプロセッサ10の一実施形態のプリデコードタグのエンコード方式(encoding)を以下に記載する。可変バイト長の命令セットとは、命令が占めるバイト数が命令によって変わり得る命令セットのことである。プロセッサ10の一実施形態が使用する可変バイト長の命令セットの例に、x86命令セットがある。

【0013】

本例に示すエンコード方式においては、あるバイトが命令の最終バイトである場合、このバイトのエンドビットがセットされる。デコードユニット20が直接デコード可能な命令は「ファストパス」命令と呼ばれる。一実施形態によると、それ以外のx86命令はMROM命令と呼ばれる。例えば、Mod R/Mバイトおよび即値バイトの2つのプリフィックスバイトを含むファストパス命令は、下記のようなエンドビットを有し得る。

30

エンドビット 00001

【0014】

MROM命令とは、複雑過ぎるためにデコードユニット20がデコードすることができないと判断された命令のことである。MROM命令は、MROMユニット34を呼び出すことにより実行される。より詳細には、MROMユニット34は、MROM命令が出現すると、この命令をパース(parse)し、定義済みのファストパス命令のサブセットを発行して、所望の操作が実行されるようにする。MROMユニット34は、ファストパス命令のサブセットをデコードユニット20にディスパッチする。

40

【0015】

プロセッサ10は、条件付き分岐命令に後続する命令を投機的にフェッチするために分岐予測を使用する。分岐予測処理を実行するために分岐予測ユニット14が設けられる。一実施形態においては、分岐予測ユニット14は、命令キャッシュ16のキャッシュラインの16バイト部分につき、最大3個の分岐ターゲットアドレスと、分岐の成立/不成立の予測とをキャッシュ可能な分岐ターゲットバッファを使用する。例えば、分岐ターゲットバッファには、2048個のエントリが含まれていても、これ以外の適切な数のエント

50

りが含まれていてもよい。プリフェッチユニット12は、特定のラインがプリデコードされたときに、最初の分岐ターゲットを決定する。その後、キャッシュライン内の命令の実行によって、このキャッシュラインに対応する分岐ターゲットが更新され得る。命令キャッシュ16は、フェッチされようとしている命令アドレスを示す情報(indication)を提供し、これによって分岐予測ユニット14は、選択すべき分岐ターゲットアドレスを決定して、分岐予測を作成し得る。デコードユニット20と機能ユニット24とは、分岐予測ユニット14に更新情報を提供する。デコードユニット20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。機能ユニット24は、分岐命令を実行して、予測した分岐方向が誤っているかどうかを判定する。この分岐方向を「迎る」可能性があり、この場合、後続の命令が、分岐命令のターゲットアドレスからフェッチされる。逆に、この分岐方向を「迎らない」可能性もあり、この場合、後続の命令が、分岐命令の後のメモリ場所からフェッチされる。予測誤りの分岐命令が検出されると、この誤って予測された分岐に後続する命令が、プロセッサ10の種々のユニットから破棄される。別法による構成においては、分岐予測ユニット14が、デコードユニット20および機能ユニット24の代わりにリオーダーバッファ32と結合されており、分岐予測ミスの情報をリオーダーバッファ32から受け取ってもよい。分岐予測ユニット14によって、種々の適切な分岐予測アルゴリズムが使用され得る。

10

【0016】

命令キャッシュ16からフェッチされた命令は、命令整列ユニット18に送られる。命令キャッシュ16から命令がフェッチされる際に、これに対応するプリデコードデータがスキャンされ、フェッチされようとしている命令に関する情報が命令整列ユニット18に(およびMROMユニット34に)提供される。命令整列ユニット18はプリデコードデータをスキャンして、各デコードユニット20に送る命令を並び替える。一実施形態においては、命令整列ユニット18は、16個の命令バイトの組2つに含まれる命令を並び替えてデコードユニット20に送る。デコードユニット20Aは、(プログラムの順序において)デコードユニット20B, 20Cが同時に受け取る命令に先行する命令を受け取る。同様に、デコードユニット20Bは、プログラムの順序においてデコードユニット20Cが同時に受け取る命令に先行する命令を受け取る。

20

【0017】

デコードユニット20は、命令整列ユニット18から受け取った命令をデコードするように構成されている。レジスタオペランド情報が検出されて、レジスタファイル30およびリオーダーバッファ32に転送される。さらに、命令が1つ以上のメモリ操作の実行を要求する場合、デコードユニット20は、この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。各命令がデコードされて、機能ユニット24用の制御値の組が生成される。この制御値は、オペランドアドレス情報、ならびに変位データまたは即値データ(命令に含まれていることがある)と共に、リザベーションステーション22にディスパッチされる。特定の実施形態においては、各命令がデコードされて最大2つの操作が生成され、これらの操作は機能ユニット24A~24Cによって別個に実行され得る。

30

【0018】

プロセッサ10はアウトオブオーダー実行に対応しており、このため、レジスタの読み出し操作および書き込み操作のために、プログラムの元のシーケンスを追跡するためにリオーダーバッファ32を使用して、レジスタリネーミングを実行し、投機的命令実行と分岐予測ミスから回復できるようにすると共に、正確な例外を可能にする。レジスタの更新を伴う命令がデコードされると、リオーダーバッファ32内に一時的な記憶場所が確保され、これにより投機的なレジスタの状態がストアされる。分岐予測が正しくない場合、誤って予測された経路に沿って投機的に実行された命令による結果が、レジスタファイル30に書き込まれる前に、これをバッファ内で無効化することができる。同様に、特定の命令が例外を発生させる場合、この特定の命令に後続する命令を破棄することができる。このように、例外は「厳密」である(すなわち、例外を発生させるこの特定の命令に後続する命令が、例外が発生する前に完了することはない)。特定の命令が、プログラムの順序

40

50

においてこの命令に先行する命令より先に投機的に実行されるという点が注目される。先行する命令は、分岐命令のこともあれば、例外を発生させる命令のこともあり、この場合、投機的な実行の結果がリオーダーバッファ32によって破棄され得る。

【0019】

デコードユニット20の出力において提供される命令制御値と、即値データまたは変位データとは、対応するリザベーションステーション22に直接転送される。一実施形態においては、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機している保留中の命令の命令情報(すなわち、命令制御値のほか、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、最大5つの命令分だけ保持することができる。図1に示す実施形態については、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に対応しているという点が注目される。したがって、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、3つの専用の「発行位置(issue position)」が形成される。つまり、発行位置0が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。整列されて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が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が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。

10

【0020】

特定の命令がデコードされたときに、要求されたオペランドがレジスタ位置である場合は、レジスタのアドレス情報が、リオーダーバッファ32とレジスタファイル30とに同時に転送される。一実施形態においては、リオーダーバッファ32は、デコードユニットからのオペランド要求を受け取るフューチャファイル(future file)も備える。x86レジスタファイルは、32ビットの実レジスタを8つ(すなわち、通常はEAX、EBX、ECX、EDX、EBP、ESI、EDIおよびESP)有するというを当業者は理解するであろう。x86プロセッサアーキテクチャを採用しているプロセッサ10の実施形態においては、レジスタファイル30は、32ビットの実レジスタの各々に対する記憶場所を有する。レジスタファイル30に、MROMユニット34が使用する付加的な記憶場所が設けられてもよい。リオーダーバッファ32は、これらのレジスタの内容を変更し、これによってアウトオブオーダー実行を可能にする結果用に、一時的な記憶場所を有する。リオーダーバッファ32内の一時的な記憶場所が、デコード時に、実レジスタのうち1つの内容を変更するように判定される各命令のために確保される。このため、特定のプログラムの実行中のさまざまな点において、リオーダーバッファ32は、所定のレジスタの投機的に実行された内容を格納している場所を1つ以上有し得る。所定の命令のデコードの後に、リオーダーバッファ32が、所定の命令において、オペランドとして使用されているレジスタに割り当てられた以前の場所を1つ以上有することが判定された場合、リオーダーバッファ32は、1)最後に割り当てられた場所にある値か、または2)そのうち前の命令を実行する機能ユニットによって値が未だ生成されていない場合は最後に割り当てられた場所に対するタグ、のいずれかを対応するリザベーションステーションに転送する。リオーダーバッファ32が、所定のレジスタ用に確保されている場所を有する場合、オペランド値(またはリオーダーバッファタグ)は、レジスタファイル30ではなくリオーダーバッファ32から提供される。要求されるレジスタのために確保された場所がリオーダーバッファ32に存在しない場合、レジスタファイル30から値が直接取得される。このオペランドがメモリ場所に対応している場合、ロード/ストアユニット26を介して、このオペランドの値がリザベーションステーションに提供される。

20

30

40

【0021】

特定の実施形態においては、リオーダーバッファ32は、並列的にデコードされた複数の命令を1つの単位としてストアおよび操作するように構成される。ここでは、この構成を「ライン指向型(line-oriented)」と呼ぶ。複数の命令をまとめて操作することによって、リオーダーバッファ32内に採用されているハードウェアを簡略化することができる。例えば、本実施形態に使用されているライン指向型のリオーダーバッファは、1つ以

50

上の命令が命令整列ユニット20によって発行(issue)されたときは必ず、3つの命令(各デコードユニット20から1つずつ)に関する命令情報に十分足る記憶領域を割り当てる。これに対して、従来のリオーダーバッファでは、実際にディスパッチされる命令の数に応じて、割り当てられる記憶領域の大きさが変わる。可変の記憶領域を割り当てるために、比較的多くの論理ゲートが必要となることがある。並列的にデコードされた複数の命令のうちの各々が実行されると、これらの命令の結果が、レジスタファイル30に同時にストアされる。この記憶領域は解放されて、並列的にデコードされる複数の命令の別の組に割り当て可能になる。さらに、制御論理が、同時にデコードされたいくつかの命令に分散されるため、命令1つにつき使用される制御論理回路系の数が減る。特定の命令を識別するリオーダーバッファタグを、ラインタグとオフセットタグとの2つのフィールドに分けてもよい。ラインタグは、この特定の命令を含む、並列的にデコードされた複数の命令のセットを識別し、オフセットタグは、このセットの中のどの命令が特定の命令に対応しているかを識別する。命令の結果をレジスタファイル30にストアして、対応する記憶領域を解放する操作は、この命令の「リタイヤ」と呼ばれているという点が注目される。フューチャファイルを使用して、レジスタファイル30の投機的状態を保存するなど、プロセッサ10の様々な実施形態に、任意のリオーダーバッファ構成を使用し得るという点がさらに注目される。

10

【0022】

上記したように、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24によって命令が実行されるまで、その命令をストアする。(i)命令のオペランドが提供され、かつ(ii)同じリザベーションステーション22A~22C内にあり、プログラムの順序においてこの命令に先行する命令にオペランドが提供されていない場合、命令が実行のために選択される。命令が機能ユニット24のうちの1つによって実行された場合、この命令の結果が、この結果を待機しているリザベーションステーション22のいずれかに直接転送されると同時に、この結果がリオーダーバッファ32を更新するために送られる(この方法は一般に「結果転送」と呼ばれる)という点が注目される。関連する結果が転送されるクロックサイクルの間に、命令が実行のために選択されて、機能ユニット24A~24Cに送られ得る。この場合、リザベーションステーション22は、転送された結果を機能ユニット24に送信する。命令が、複数の操作にデコードされて、機能ユニット24によって実行される実施形態においては、これらの操作が互いに独立してスケジューリングされ得る。

20

30

【0023】

一実施形態においては、各機能ユニット24は、加算および減算の整数演算操作のほか、シフト、ローテート、論理演算および分岐処理も実行するように構成されている。特定の命令について、デコードユニット20によってデコードされた制御値を受けて、操作が実行される。浮動小数点演算を処理するために、浮動小数点演算ユニット(図示せず)が使用され得るという点が注目される。この浮動小数点演算ユニットは、MROMユニット34またはリオーダーバッファ32から命令を受け取って、次にリオーダーバッファ32と通信を行って命令を遂行するコプロセッサとしても動作し得る。さらに、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に、アドレス生成を行うように構成され得る。特定の実施形態においては、各機能ユニット24は、アドレスを生成するアドレス生成ユニットと、それ以外の機能を実行する実行ユニットとを備え得る。この2つのユニットは、あるクロックサイクルの間に別の命令または操作を独立して処理することができる。

40

【0024】

また、各機能ユニット24は、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に提供する。分岐予測が正しくない場合、分岐予測ユニット14は、命令処理パイプラインに既に入っている、予測ミスとなった分岐に後続する命令を破棄して、必要な命令を命令キャッシュ16またはメインメモリからフェッチさせる。このような場合、投機的に実行され、ロード/ストアユニット26とリオーダーバッファ32とに一時的にス

50

トアされた命令を含め、元のプログラムシーケンスにおいて、予測ミスとなった分岐命令に後続する命令の結果が破棄されるという点が注目される。機能ユニット24からリオーダーバッファ32に分岐の実行結果が提供されてもよく、この結果が、分岐予測ミスを機能ユニット24に通知し得るとい点がさらに注目される。

【0025】

レジスタ値が更新されようとしている場合、機能ユニット24によって生成された結果は、リオーダーバッファ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。結果がレジスタにストアされる必要がある場合、リオーダーバッファ32は、命令がデコードされたときにレジスタの値のために確保された場所に、結果をストアする。機能ユニット24およびロード/ストアユニット26から結果を転送するために、複数の結果バス38が設けられる。結果バス38は、生成された結果のほかに、実行されようとしている命令を識別するリオーダーバッファタグを伝達する。

10

【0026】

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との間のインターフェイスを提供する。一実施形態においては、ロード/ストアユニット26は、2つのロード/ストアバッファを備えて構成される。第1のロード/ストアバッファは、データキャッシュ28に未だアクセスしていない実行待ちのロードおよびストアに対応するデータおよびアドレス情報用の記憶場所を備える。第2のロード/ストアバッファは、データキャッシュ28に既にアクセスしている実行待ちのロードおよびストアに対応するデータおよびアドレス情報用の記憶場所を備える。例えば、第1のバッファは12個の場所を備え得、第2のバッファは32個の場所を備え得る。デコードユニット20は、ロード/ストアユニット26へのアクセスを調停(arbitrate)する。第1のバッファが満杯の場合は、デコードユニットは、待機中のロード要求またはストア要求に関する情報を格納するための空きが、ロード/ストアユニット26に生じるまで待機する。また、ロード/ストアユニット26は、データの一貫性が確実に維持されるように、待機中のストアメモリ操作に対するロードメモリ操作の依存関係を検査する。メモリ操作とは、プロセッサ10とメインメモリサブシステムとの間で行われるデータ転送である。メモリ操作は、メモリにストアされているオペランドを利用する命令の結果生じることあるれば、データ転送のみを生じさせるが他の操作を生じさせないロード/ストア命令の結果生じることある。さらに、ロード/ストアユニット26は、セグメントレジスタやx86プロセッサアーキテクチャによって規定されるアドレス変換機構に関わるその他のレジスタなどの特殊レジスタのための、特別なレジスタ記憶領域を備えていてもよい。

20

30

【0027】

データキャッシュ28は、ロード/ストアユニット26とメイン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ストアするために設けられる高速キャッシュメモリである。一実施形態においては、データキャッシュ28は、2ウェイセットアソシエーティブ構造において、最大64キロバイトのデータをストアできるだけの容量を有する。データキャッシュ28は、セットアソシエーティブ構成、フルアソシエーティブ構成、ダイレクトマップド構成、または任意の構成かつ任意のサイズのその他の構成など、様々なメモリ構成に実装可能であるということが理解される。

40

【0028】

x86プロセッサアーキテクチャを使用するプロセッサ10の特定の実施形態においては、命令キャッシュ16およびデータキャッシュ28は、リニアアドレス指定されて物理タグが付される。リニアアドレスは、命令によって指定されるオフセットと、x86のアドレス変換メカニズムのセグメント部分によって指定されるベースアドレスとから生成される。リニアアドレスは、メインメモリにアクセスするために、随意的に物理アドレスに変換されることがある。リニアアドレスから物理アドレスへの変換は、x86のアドレス変換メカニズムのページング部分によって指定される。物理アドレスと物理タグとが比較されて、ヒット/ミス状態が判定される。

【0029】

50

バスインターフェイスユニット 37 は、バスを介して、プロセッサ 10 とコンピュータシステム内の他の部品とを通信させるように構成されている。例えば、バスは、デジタルイクイップメント コーポレーション (Digital Equipment Corporation) によって開発された EV-6 バスと互換性を有するバスであり得る。あるいは、パケットベースのリンク、一方向リンクまたは双方向リンクなど、任意の適した相互接続構造を使用することができる。また、オプションの L2 キャッシュインターフェイスを使用して、レベル 2 キャッシュとインターフェイスを取ってもよい。

【0030】

以下の説明では、x86 マイクロプロセッサアーキテクチャを例に採り上げる。しかし、ここに記載する分岐予測技法 (technique) は、どのようなマイクロプロセッサアーキテクチャ内でも使用することができ、そのような実施形態も考察される。x86 マイクロプロセッサアーキテクチャでは、シングルバイトのオペコードからなるサブルーチンリターン命令 (RET 命令など) が定義されているという点が注目される。サブルーチンリターン命令は、その分岐ターゲットアドレスを、ESP レジスタによって示されるスタックの最上部から取得するように指示する。場合によっては、このシングルバイトの RET 命令の処理が特別な問題を引き起こし得る。このような場合に対処するメカニズムを、下記に詳細に示す。

【0031】

分岐予測の概略

図 2 に、分岐予測ユニット 14 の一実施形態の一部を示す。図 2 に示した部分のほか、分岐予測ユニット 14 の他の実施形態も可能であり、考慮される。図 2 に示すように、分岐予測ユニット 14 は、グローバル予測記憶域 205、ローカル予測記憶域 206、分岐ターゲット記憶域 208、更新論理 200, 202、グローバル履歴シフトレジスタ (GHSR) 204、ラインバッファ 210、戻りアドレススタック (RAS) 230、連続アドレス生成器 (SEQ) 232、予測論理 220、犠牲キャッシュ 260、分岐アドレス計算器 270、命令キャッシュ 16、L2 キャッシュ 1828、およびデコード回路 400 を備える。犠牲キャッシュ 260 は、ローカル予測記憶域 206 から排除された分岐予測情報をストアするように構成されたレベル 2 (L2) キャッシュである。一実施形態においては、犠牲キャッシュ 260 は、以前はローカル予測記憶域 206 に保持されていたが、別のデータを記憶する領域を確保するために排除されたデータのみをキャッシュするように構成されている。ローカル予測記憶域 206 と犠牲キャッシュ 260 とがデータを重複して記憶することがないため、より多くの分岐予測情報を保持することができ有利である。

【0032】

グローバル予測記憶域 205、ローカル予測記憶域 206、分岐ターゲット記憶域 208、命令キャッシュ 16、予測論理 220、分岐アドレス計算器 270、およびラインバッファ 210 は、フェッチアドレスマルチプレクサ 222 から出力されるフェッチアドレスバス 236 に結合されている。グローバル履歴シフトレジスタ 204 は、バス 234 を介してグローバル予測記憶域 205 およびラインバッファ 210 に結合されている。更新論理 200 は、グローバル予測記憶域 205、ローカル予測記憶域 206 および分岐ターゲット記憶域 208 に結合されている。ラインバッファ 210 は、バス 248 を介して更新論理 200, 202 に結合されている。さらに、更新論理 202 は、バス 246 を介してグローバル履歴シフトレジスタ 204 に結合されている。リオーダーバッファ 32 は、選択制御を行い、マルチプレクサ 222 にアドレスを転送する。また、リオーダーバッファ 32 は、更新論理 200 および更新論理 202 に、分岐の予測された挙動と実際の挙動とに関する情報を提供する。グローバル予測記憶域 205 とローカル予測記憶域 206 とは、それぞれバス 238, 242 を介して予測論理 220 に結合されている。予測論理 220 は、バス 250 を介して分岐アドレス計算器 270 に、選択信号 240 を介してマルチプレクサ 212 に結合されている。命令キャッシュ 16 は、バス 241 を介して分岐アドレス計算器 270 に結合されている。マルチプレクサ 212 の出力は、バス 243 を介して分岐アドレス計算器 270 およびマルチプレクサ 221 に結合されている。また、分

岐アドレス計算器 270 は、バス 245 および選択信号 223 を介してマルチプレクサ 221 に結合されている。最後に、マルチプレクサ 221 からの出力はマルチプレクサ 222 に結合されている。

【0033】

一般に、分岐予測ユニット 14 の図 2 に示す部分の基本的な操作は下記のとおりである。フェッチアドレス 236 が、ラインバッファ 210、ローカル予測記憶域 206、ターゲットアレイ記憶域 208 および分岐アドレス計算器 270 に送られる。さらに、フェッチアドレス 236 の一部がグローバル履歴 234 と結合されて、インデックスが生成され、グローバル予測記憶域 205 に送られる。さらに、フェッチアドレス 236 の部分 225 が予測論理 220 に送られる。グローバル予測記憶域 205 はグローバル予測 238 を送り、ローカル予測記憶域 206 はローカル予測 242 を送り、ターゲットアレイ 208 は受け取ったフェッチアドレスに対応するターゲットアドレスを送る。ローカル予測記憶域 206 から送られるローカル予測 242 は、予測論理 220 が分岐予測を作成するために使用する情報を提供する。同様に、グローバル予測記憶域 205 は、分岐予測の作成に使用されるグローバル予測 238 を予測論理 220 に送る。一実施形態においては、動的な挙動を示す分岐について、グローバル予測 238 によって、ローカル予測記憶域 206 から提供されるローカル予測 242 が無効にされ得る。動的な挙動を示す分岐の分類については後述する。最後に、予測論理 220 は、新しい命令のフェッチに使用される次のフェッチアドレス 243 を選択する信号をマルチプレクサ 212 に送る。場合によっては、マルチプレクサ 212 によって送信されるフェッチアドレス 243 が、現在の分岐予測の唯一のフェッチアドレスのこともある。しかし、分岐アドレス計算器 270 が、マルチプレクサ 212 によって送信されるフェッチアドレス 243 が誤っていたことを検出すると、現在の分岐予測に対応する第 2 のフェッチアドレス 245 を送信することもある。そのような場合、分岐アドレス計算器 270 は、フェッチアドレス 245 を選択して、マルチプレクサ 221 から出力させるための信号 223 を送信し得る。このように、予測ミスが初期の段階で検出されて訂正される。

【0034】

動的な挙動およびグローバル分類

上記したように、一実施形態においては、分岐予測ユニット 14 にグローバル予測メカニズムが含まれ得る。グローバル予測メカニズムの詳細は、同時係属中であって譲受人が共通する特許出願「グローバル履歴分岐予測における条件付き分岐の動的分類 (Dynamic Classification of Conditional Branches in Global History Branch Prediction)」、米国特許出願第 09 / 441, 630 号、1999 年 11 月 16 日出願、ズラスキー ジュニアら (Zuraski, Jr. et al.) に記載されている。同開示は、参照として本願に包含される。上記したように、プリフェッチユニット 12 は、分岐命令を検出して、分岐命令に対応する分岐情報を分岐予測ユニット 14 に送るように構成され得る。条件付き分岐が検出されると、更新論理 200 は、この分岐予測に対応するエントリをローカル予測記憶域 206 内に作成し、新たに作成したこの分岐予測エントリを「分岐不成立」に初期化し得る。一実施形態においては、ローカル予測記憶域 206 は、分岐マーカを含む分岐予測情報をストアし得る。この情報は、分岐予測を作成して、分岐ターゲット記憶域 208、連続アドレス 232、または戻りスタックアドレス 230 にストアされている複数の分岐ターゲットアドレスのうちから行き先を選択するために使用される。ローカル予測記憶域 206 内に分岐のエントリを作成する際に、予測分岐方向が「分岐不成立」に初期化され、これと対応する分岐マーカが、連続アドレス 232 を示すように初期化される。このほかに、条件付き分岐に対応するエントリがラインバッファ 210 内に作成される。ラインバッファのエントリには、グローバル履歴、フェッチアドレス、グローバル予測およびグローバルビットが含まれ得る。

【0035】

ローカル予測記憶域 206 にある分岐予測が「分岐不成立」の場合、マルチプレクサ 212 から連続アドレス 232 が送られる。最終的な予測 250 が、更新論理 202 に送ら

10

20

30

40

50

れ、更新論理 202 は、動的であると分類された予測分岐方向を、グローバル履歴シフトレジスタ 204 に移す。ローカル分岐予測のエントリが「分岐不成立」の場合、最終的な予測 250 は「分岐不成立」を示し、信号 240 は、マルチプレクサ 212 からの連続アドレス 232 を次のフェッチアドレスとして選択する。この分岐予測のエントリが分岐予測ユニット 14 から削除される前に、この分岐が次に実行されるとき、予測分岐方向は「分岐不成立」であり、次のフェッチアドレスとして連続アドレス 232 が送られる。リタイヤ時に、ラインバッファ 210 内の対応するエントリが、更新論理 200 および更新論理 202 に送られ、ラインバッファ 210 から削除される。ラインバッファのエントリが、分岐が非動的であると分類されていることを示し、リオーダーバッファ 32 が、分岐予測が的中したことを示す場合、更新論理 200 または 202 によって更新が実行されることはな
10

【0036】

成立した非動的な分岐

リオーダーバッファ 32 は、リタイヤ時または予測ミスの発生時に、分岐の挙動に関する情報を更新論理 200 に送る。また、ラインバッファ 210 も、更新論理 200、202 にラインバッファのエントリを送る。ラインバッファの分岐のエントリが、分類が非動的であると分類されており、「分岐不成立」と予測されていることを示し、かつリオーダーバッファ 32 が、対応する分岐の予測が外れたことを示す場合、更新論理 200 は、予測ミスとなつた分岐に対応する分岐予測のエントリを更新する。更新論理 200 は、ローカル予測記憶域 206 内の分岐予測を「分岐不成立」から「分岐成立」に更新し、分岐ターゲットアドレスを分岐ターゲット記憶域 208 にストアする。ストアした分岐ターゲットアドレスに対応する「動的」（すなわち「グローバル」）ビットが、分岐が静的すなわち非動的であることを示すように初期化される。この値は、2進数の 0 によって表され得る。この分岐予測のエントリが分岐予測ユニット 14 から削除される前に、この分岐が次に実行されたとき、分岐予測のエントリは「分岐成立」を示し、分岐は非動的として分類されている。分岐が「分岐成立」と予測され、非動的であると分類されている場合、予測論理 220 は、マルチプレクサ 212 からターゲットアドレスを選択する。上記と同様に、分岐予測が正しい場合は、更新論理 200 または 202 によって分岐予測を更新する必要はない。これに対して、「分岐成立」と予測された非動的な分岐が成立しなかった場合、分岐予測のエントリとグローバル履歴シフトレジスタ 204 とが更新される。
20

【0037】

動的な分岐

非動的であると分類され、かつ「分岐成立」と予測された分岐の予測が外れた場合、更新論理 200 は、ローカル予測記憶域 206 内の、予測ミスとなつた分岐に対応する動的ビットを更新して、分岐を動的すなわちグローバルであると分類する。さらに、更新論理 200 は、グローバル予測記憶域 204 内の、予測ミスとなつた分岐に対応するグローバル予測エントリを、「分岐不成立」に更新する。また、更新論理 202 は、分岐が不成立だったことを示すようにグローバル履歴シフトレジスタ 204 を更新する。一実施形態においては、グローバル履歴シフトレジスタ 204 は、最新の 8 個の動的な分岐の挙動を記録する。
40

【0038】

動的な分岐がフェッチされると、ローカル予測記憶域 206、ターゲットアレイ 208 およびラインバッファ 210 にフェッチアドレス 236 が送られる。さらに、このフェッチアドレスは、グローバル履歴シフトレジスタ 204 の内容と結合されて、インデックス 203 が生成され、これがグローバル予測記憶域 205 に送られる。また、グローバル履歴シフトレジスタ 204 の内容は、バス 234 を介してラインバッファ 210 にも送られる。一実施形態においては、フェッチアドレス 236 のビット 9 ~ 4 と、グローバル履歴シフトレジスタ 204 の内容を連結することによってインデックス 203 が生成される。
50

論理和または排他的論理和を求めるなど、インデックスを生成する別の方法も同様に考察される。このインデックスを基にグローバル予測記憶域 205 のエントリが選択されて、このエントリがラインバッファ 210、更新論理 202 およびマルチプレクサ 220 に送られる。グローバル予測記憶域 204 から送られた予測分岐方向が、更新論理 202 によってグローバル履歴シフトレジスタ 204 に移される。例えば、2進数の 1 は「分岐成立」を、2進数の 0 は「分岐不成立」を表し得る。対応する動的ビットが、分岐がグローバルであると分類されていることを示し、かつグローバル予測が「分岐成立」を示す場合、マルチプレクサ 212 から送られるターゲットアドレスが次のフェッチアドレスとして選択される。グローバル予測が「分岐不成立」を示す場合、マルチプレクサ 212 から、連続アドレス 232 が次のフェッチアドレスとして選択される。

10

【0039】

リタイヤ時に、リオーダーバッファ 32 は更新論理 200 および更新論理 202 に分岐情報を送る。さらに、ラインバッファ 210 は、対応する分岐情報を更新論理 202 に送る。リオーダーバッファ 32 が、動的分岐の予測が正しいことを示す場合、更新論理 200 は、分岐のこの挙動を示すためにグローバル予測のエントリ 205 を変更する。一実施形態においては、グローバル分岐予測のエントリには飽和カウンタが含まれる。このカウンタは 2 ビット長で、分岐が成立するとインクリメントされ、分岐が成立しないとデクリメントされ得る。この指標 (indicator) を使用して、「分岐成立の可能性大 (strongly taken)」、「分岐成立の可能性小 (weakly taken)」、「分岐不成立の可能性大 (strongly not taken)」、「分岐不成立の可能性小 (weakly not taken)」を表すことができる。動的分岐の予測が外れた場合、更新論理 200 は、分岐のこの挙動を示すためにグローバル予測のエントリ 205 を更新する。さらに、更新論理 202 は、予測ミスの発生時に、動的分岐の予測による挙動ではなく実際の挙動を反映するようにグローバル履歴シフトレジスタ 204 を修正する。

20

【0040】

犠牲分岐予測用のキャッシュ

上記の分岐予測ユニット 14 の動作の概略は、対応するフェッチアドレスがローカル予測記憶域 206 に存在する場合の分岐予測を想定している。しかし、受け取ったフェッチアドレスに対応する分岐予測がローカル予測記憶域 206 にない場合もある。次に、このような状況に対処する方法を概説する。

30

【0041】

図 3 は、1 次レベル (L1) 分岐予測用の記憶域 206 内に分岐予測のエントリを作成する方法の一実施形態を示すフローチャートであり、存在する分岐予測のエントリを排除することと、排除された分岐予測情報の一部 (サブセット) を 2 次レベル (L2) 記憶域にストアすることとが行われる。2 次レベルの分岐予測記憶域 260 を利用することによって、L1 予測記憶域 206 から排除された分岐予測情報を保持できるようになり有利である。特定の分岐に対して正確な予測を作成するには、数多くの分岐の実行を記録した分岐の実行履歴の作成が必要となり得ることを当業者は認めるであろう。このため、排除された分岐予測に対応する分岐予測情報をキャッシュに保持することにより、分岐予測の精度を向上させることができる。L1 記憶域 206 へのデータの再ロードが要求されていることが検出される (判断ブロック 301) と、分岐予測用の対応するエントリを作成するための空きが L1 分岐予測記憶域 206 に存在するかどうか判定される (判断ブロック 303)。再ロードするデータに対応する、L1 分岐予測記憶域 206 のエントリが既に占有されている場合、現在のエントリが排除され得 (ブロック 305)、排除された分岐予測情報の一部のみが L2 記憶域 260 にストアされる (ブロック 307)。再ロードするデータを L2 記憶域 260 からフェッチする場合 (判断ブロック 309)、L2 記憶域 260 からフェッチされるデータは分岐予測のエントリの一部を有しており、これが対応する分岐予測の完全なエントリ (ブロック 311) の再生成に使用され得る。次に、再生成されたエントリが L1 記憶域 206 にストアされ得る (ブロック 315)。一方、再ロードするデータを L2 記憶域 260 からフェッチしない場合 (判断ブロック 309)、新

40

50

しい分岐予測のエントリが初期化され（ブロック 313）、L1 記憶域にストアされる（ブロック 315）。

【0042】

図 4 は、対応する分岐予測が L1 分岐予測記憶域に存在しない場合の方法の別の実施形態である。初めに、フェッチアドレスが L1 分岐予測記憶域 206 に提示される（ブロック 400）。フェッチアドレスが提示されると、提示されたフェッチアドレスに対応する分岐予測が L1 予測記憶域に存在するかどうか判定される（判断ブロック 402）。対応する分岐予測が L1 記憶域に存在する場合、予測が送られる（ブロック 418）。しかし、対応する分岐予測が L1 記憶域 206 にない場合、フェッチアドレスに対応する L1 予測記憶域 206 のエントリが、別のフェッチアドレスに対応する分岐予測のエントリによって占有されているかどうか判定される（判断ブロック 404）。L1 予測記憶域 206 に別のエントリが存在する場合、現在のエントリが L1 予測記憶域 206 から排除される（ブロック 406）。L1 予測記憶域の対応するエントリが有効なエントリによって占有されていない場合、L2 予測記憶域 260 に対して、このフェッチアドレスに対応する分岐予測のエントリを要求する（ブロック 408）。一実施形態においては、L1 予測記憶域 206 と L2 予測記憶域 260 とに対して同時に要求を行うことができる。対応するエントリが L2 予測記憶域 260 に存在しない場合（ブロック 410）、提示されたフェッチアドレス用の、新しい分岐予測のエントリが L1 予測記憶域 206 に作成され得る（ブロック 412）。一方、L2 分岐予測記憶域 260 内にこのフェッチアドレスに対応するエントリが存在する場合、L2 のエントリのデータが使用されて、フェッチアドレスに対応する完全な分岐予測が作成される（ブロック 414）。次に、再作成された分岐予測が L1 分岐予測記憶域にストアされ（ブロック 416）、送られて分岐予測の作成に使用される（ブロック 418）。分岐命令に対応する分岐予測情報を、L2 分岐予測記憶域から復旧することができ、分岐実行の履歴から再生成する必要はなく有利である。さらに、分岐予測に対応する情報の一部のみを L2 分岐予測用の記憶域 260 にストアするだけで済む。このようにして、L2 分岐予測用の記憶域 260 のサイズを小さくすることができる。

【0043】

ローカル分岐予測記憶域

一実施形態においては、ローカル予測記憶域 206 は、命令キャッシュ 16 と同様な方法によって編成され得る。ローカル予測記憶域 206 にストアされているデータは、ウェイに編成された記憶域のラインから構成され得る。一実施形態においては、ローカル予測記憶域 206 は、命令キャッシュ 16 内の全エントリを格納できるだけの十分なサイズを有する。別の実施形態においては、ローカル予測記憶域 206 のサイズは、命令キャッシュ 16 のサイズよりも小さくてもよい。例えば、ローカル予測記憶域 206 は、命令キャッシュ 16 のサイズの 4 分の 1 であり得る。このような実施形態においては、ローカル予測記憶域 206 内の、同じ記憶場所を指している複数のフェッチアドレスのエントリ同士を区別するために、ローカル予測記憶域 206 にローカル予測と共に追加ビットがストアされ得る。一実施形態においては、グローバル予測記憶域 205 が、複数のフェッチアドレスの予測を 1 つの記憶場所に格納できるように構成され得る。当業者は、ローカル予測記憶域 206 およびグローバル予測記憶域 205 を様々に構成可能であることを容易に理解するであろう。

【0044】

図 5 を参照すると、連続する命令バイト 50 と、これに対応するローカル予測記憶域 206 のエントリとの組の例を表す図が示される。図 5 では、命令内の各バイトが、短い垂直線（参照符号 54 など）によって表される。さらに、組 50 にある命令同士を区切っている線（参照符号 56 など）もバイトの境界を示す。図 5 に示す命令は可変長であり、このため、図 5 に示す命令を含む命令セットは可変バイト長の命令セットである。換言すれば、この可変バイト長の命令セットの先頭の命令のバイト数は、この命令セットの 2 番目の命令のバイト数とは異なる。命令セット内の各命令のバイト数が全て等しくなるような

10

20

30

40

50

固定長の命令セットもあり得る。

【 0 0 4 5 】

図 5 に示すように、組 5 0 には、非分岐命令 I N 0 ~ I N 4 が含まれる。命令 I N 0 , I N 3 , I N 4 は 2 バイト命令である。命令 I N 1 は 1 バイト命令であり、命令 I N 2 は 3 バイト命令である。このほか、「分岐成立」と予測された 3 つの分岐命令 P B 0 , P B 1 , P B 2 も示されており、これらは全て 2 バイトを占める。非分岐命令および分岐命令は様々なバイト数を取りうるという点が注目される。

【 0 0 4 6 】

本明細書で使用されるように、アドレスのオフセットは、アドレスの所定数の最下位ビット (least significant bit) を含む。この数は、オフセットが関連しているバイトの組に含まれる各バイトに対して、異なったビットのエンコーディング (コード化) を付与するのに十分な数である。例えば、組 5 0 は、バイト 0 ~ 1 5 の 1 6 バイトを含む。このため、このグループにあるアドレスの最下位の 4 ビットが、そのアドレスのオフセットを形成している。このアドレスの残りのビットは、メインメモリ内の隣接する他の命令バイトの組から、組 5 0 を識別する。さらに、残りのビットのうちの所定数の最下位ビットが、命令キャッシュ 1 6 によって使用され、組 5 0 をストア可能な記憶場所の行を選択するためのインデックスを形成している。

【 0 0 4 7 】

セット 5 2 は 9 個の分岐マーカービット 0 ~ 8 からなるセットの例で、組 5 0 に対応している。セット 5 2 内の 9 個の分岐マーカービットの各々は、組 5 0 の特定のバイト群に対応しており、「分岐成立」と予測された分岐命令が、対応するバイト群で終端しているかどうかを示す。本図の実施形態においては、分岐マーカービット 0 は命令バイト 0 に、分岐マーカービット 1 は命令バイト 1 ~ 2 に、分岐マーカービット 2 は命令バイト 3 ~ 4 に、分岐マーカービット 3 は命令バイト 5 ~ 6 に、分岐マーカービット 4 は命令バイト 7 ~ 8 に、分岐マーカービット 5 は命令バイト 9 ~ 1 0 に、分岐マーカービット 6 は命令バイト 1 1 ~ 1 2 に、分岐マーカービット 7 は命令バイト 1 3 ~ 1 4 に、分岐マーカービット 8 は命令バイト 1 5 に対応している。分岐マーカービット 5 2 のほかに、分岐マーカービット 5 2 に対応する分岐予測情報 4 5 0 も含まれる。分岐予測情報 4 5 0 は、「分岐成立」と予測された分岐に対応する情報を含んでおり、以下に詳細に記載するように、分岐予測の作成に使用される。

【 0 0 4 8 】

図 5 の実施形態においては、成立予測分岐命令 (predicted taken branch production) が、対応するバイト範囲で終端している場合は、分岐マーカービットが「1」にセットされる。成立予測分岐命令が、分岐マーカービットに対応するバイト範囲で終端していない場合は、この分岐マーカービットが「0」にセットされる。したがって、分岐マーカービット 2 は、成立予測分岐 P B 0 が命令バイト範囲 3 ~ 4 内で終端していることを示す。分岐マーカービット 6 は、成立予測分岐 P B 1 が命令バイト範囲 1 1 ~ 1 2 内で終端していることを示す。最後に、分岐マーカービット 7 は、成立予測分岐 P B 2 が命令バイト範囲 1 3 ~ 1 4 内で終端していることを示す。一実施形態においては、分岐マーカービット 5 2 によって、最大 3 つの成立予測分岐が示され得る。このような実施形態においては、分岐予測情報 4 5 0 は、3 つの別個の分岐命令に関する情報を格納することができる。この情報 4 5 0 は、分岐命令の種類を含むほか、対応する分岐の過去の挙動に関する情報を示し得る。分岐予測情報 4 5 0 については、図 1 5 を説明する際に詳述する。

【 0 0 4 9 】

フェッチアドレスがローカル予測記憶域 2 0 6 に提示されると、分岐マーカービット 5 2 と分岐情報 4 5 0 とからなる組が予測論理 2 2 0 に送られ、分岐予測の作成に使用される。一実施形態においては、分岐予測の作成の際に考慮すべき分岐マーカービット 5 2 の識別に、フェッチアドレスのオフセットが使用される。図 6 を参照すると、分岐マーカービット 5 2 とフェッチアドレスのオフセットとの関係の一実施形態が示される。図 6 において、列 6 0 2 ~ 6 1 8 を有するテーブルは、オフセットアドレスと、これと対応する分

10

20

30

40

50

岐マーカービットとを有する。一般に、対応するオフセットが、テーブル内にある条件を満足するかどうかによって、分岐予測の作成の際に分岐マーカービットが考慮されるかが決まる。例えば、列602は、アドレスのオフセットが0以下の場合、分岐予測の作成の際に全ての分岐マーカービットが考慮されることを示す。このように、このテーブルは、所定のオフセットについて、考慮される分岐マーカービットの番号の最小値を示している。したがって、オフセットが7の場合、分岐マーカービット4～8が考慮される。オフセットが15の場合、分岐マーカービット8のみが考慮される。

【0050】

図7を参照すると、上記した命令バイトと分岐マーカービットとの関係を表す図が示される。図7には、命令バイト50と、これと対応する分岐マーカービットとが記載されている。また、命令バイト#702、分岐マーカービット#704、各分岐マーカービットに対応する命令バイト706、および各分岐マーカービットに対応するフェッチアドレスのオフセット708も記載されている。分岐情報450は記載されていない。図5に関して上記したように、「分岐成立」と予測された3つの分岐命令PB0～PB2に対応する分岐マーカービット2, 6および7がセットされている。例として、オフセットが3のフェッチアドレスがローカル予測記憶域206に提示された場合を考える。この例では、分岐マーカービット2～8は、全て各ビットの条件（つまり、分岐マーカービット2ではオフセット3、分岐マーカービット3ではオフセット5など）を満足している。したがって、分岐予測の作成時に、分岐マーカービット2～8が使用される一方、分岐マーカービット0～1は、命令シーケンスにおいて現在フェッチされようとしている命令に先行する命令に対応しているため、効率的に無視され得る。このようにして、分岐予測の作成時に考慮すべき分岐マーカービットを、容易に判定できるようになる。

【0051】

図8は、予測の作成に分岐マーカービット52を使用する方法の一実施形態を示すものである。図8には、受け取ったフェッチアドレスのオフセット（本図で「a」として示される）と、図7に示すオフセット値708に相当する値0, 1, 3, 5, 7, 9, 11, 13および15とを比較する回路系802が記載されている。受け取ったオフセットが、比較対象の値以下である度に、このことを示す信号808がANDゲート804に送られる。さらに、各分岐マーカービット52もANDゲート804に送られる。論理ANDゲート808は、論理ORゲート806に結合されており、その出力は、分岐予測が作成されようとしているかどうかを示す。図8の例でも上記と同様に、オフセットが値3に指定されたフェッチアドレスを受け取った場合を考える。図7に示すように、オフセットが3の命令は、成立予測分岐PB0に対応している。次に、対応する分岐マーカービット52が、ローカル予測記憶域206から図8に示す論理に送られる。回路系802は、値が3に指定されたオフセットを受け取って、信号808C～808Iをアサートし、ゲート804C～804Iが各々これらを受け取る。さらに、分岐マーカービット2, 6および7は、2進数値の「1」をゲート804C, 804Gおよび804Hのそれぞれに送る。残りの分岐マーカービットは、2進数値の「0」を送る。したがって、ゲート804C, 804Gおよび804Hの各々は、ゲート806に2進数値の「1」を送り、この結果、ゲート806は、分岐予測が作成されようとしていることを示す2進数値の「1」を送る。

【0052】

図9は、図8に示す回路系の、動的論理を使用した一実施形態900を示すものである。当業者は動的論理の動作を容易に理解するであろう。図9には、結合トランジスタ902A～902Iが含まれており、これらは論理ANDゲート804A～804Iにそれぞれ対応する。また、プリチャージされた値を動的論理に供給し得るトランジスタ904およびインバータ906も記載されている。まず、回路の評価段階の間に、インバータ906の出力908は、分岐予測が作成されようとしていないことを示す2進数の状態「0」を送る。しかし、少なくとも1つのゲート902の入力の両方がアサートされると、インバータ906の出力908が、分岐予測が作成されようとしていることを示す2進数値の「1」に遷移する。当業者は、同じ動作を達成するために、図9, 10の論理を実装し得

10

20

30

40

50

る多くの方法を容易に認めるであろう。

【 0 0 5 3 】

分岐マーカービットの更新

一般に分岐命令は、少なくとも1つのオペコード（マイクロプロセッサ10が使用する命令セット内で分岐命令を識別する）と、分岐ターゲットアドレスを生成するために分岐命令のアドレスに加算される変位（あるいは分岐命令の直後の命令のアドレス）とを有する。このため、分岐命令は、通常は少なくとも2バイトを占める。このことを利用すれば、全命令バイトに対する分岐マーカービットをストアする必要がなく、連続する命令バイトの組に対してストアされる分岐マーカービットのビット数を低減させることができる。しかし、分岐が「分岐成立」と予測されていることを示すように分岐マーカーがセットされていても、分岐予測が見落とされ得る、シングルバイトのRET命令が関与する留意すべき特定の場合が存在する。この特定の場合に対処するための実施形態を以下に記載する。

10

【 0 0 5 4 】

図11～15には、分岐マーカービットを更新する例が示される。図10には、連続する命令バイト50、分岐マーカービット52および分岐情報450からなる組が示されている。図10の例においては、「分岐成立」と予測された分岐はなく、分岐マーカービット52もセットされていない。図11を参照すると、成立予測分岐が検出された後の、命令バイト50、マーカービット52および分岐情報450と、これに対応する分岐マーカービット52の更新の様子とが示される。図11では、命令バイト50内の命令バイト#8および#9で、成立予測分岐PB0 1201が検出されている。この成立予測分岐は、命令バイト#9で終了しているため、分岐マーカービット#5 1203を2進数値の「1」に設定して、分岐マーカービット52が更新される。さらに、一実施形態においては、分岐情報のエントリ500A～500Cは、命令バイト50に存在する成立予測分岐の1番目、2番目および3番目にそれぞれ対応し得る。このような実施形態においては、成立予測分岐PB0 1201は、命令バイトの組50の、1番目の成立予測分岐であると検出され得る。したがって、分岐命令PB0 1201に対応する情報が分岐情報のエントリ500Aにストアされ得る。

20

【 0 0 5 5 】

図12に、2番目の成立予測分岐が検出されたときの、分岐マーカービット52の更新の様子を示される。成立予測分岐PB1 1301が組50で検出される。分岐PB1 1301が検出され、かつ分岐PB1 1301が命令バイト#3で終了しているため、分岐マーカービット#2 1303が2進数値の「1」にセットされる。さらに、本図の実施形態においては、新たに検出された分岐PB1 1301が、組50の、1番目の成立予測分岐であることが判定される。このため、分岐PB1 1301に対応する分岐情報は、分岐情報のエントリ500Aとなる。しかし、図11の説明で上記したように、分岐情報のエントリ500Aは、今は分岐PB0 1201に対応する分岐情報を格納している。したがって、エントリ500A内のデータがエントリ500Bに移され、分岐PB1 1301に対応する新しい分岐情報がエントリ500Aにストアされる。

30

【 0 0 5 6 】

図13には、成立予測分岐のエントリの更新の様子を示される。図13の例では、分岐PB1 1301に対して、予測が「分岐不成立」であることがのちに判定される。「分岐成立」となっているPB1 1301の予測を更新するために、対応する動的ビット500Aが2進数値の「1」にセットされる。このため、分岐PB0およびPB1に対応する分岐予測が存在することになる。しかし、分岐PB1は動的500Aであると分類される一方で、分岐PB0はローカル500Bであると分類されたままである。

40

【 0 0 5 7 】

上記の議論においては、分岐情報のエントリ500A～500Cのそれぞれが、組50の特定の成立予測分岐に対応しているものとして記載した。例えば、エントリ500Aは組50の1番目の成立予測分岐に対応し、エントリ500Bは2番目の分岐に対応し、エ

50

ントリ500Cは3番目の分岐に対応している。したがって、更新後もこの対応関係を維持するには、エントリ500A~500C内のデータをシフトさせる必要がある。一般に、新しいエントリが作成される場合、このエントリに対応するデータエントリと、このエントリの右にある全エントリとが右にシフトされて、新しいエントリが入る空きが作られる。例えば、エントリ500Aに対応する新たに成立予測分岐が検出され、エントリ500Aに既に有効なエントリがストアされている場合、エントリ500Aの内容がエントリ500Bにシフトされ、エントリ500Bの内容がエントリ500Cにシフトされ、エントリ500Cの内容が分岐情報450から排除されて失われる。一実施形態においては、ターゲットアレイ208にあるターゲットアドレスも同様にシフトされて、組50の成立予測分岐、エントリ500の分岐情報、およびターゲットアレイ208のターゲットの間
10
の対応関係が保たれ得る。上記の実施形態では、エントリ間でデータをシフトすることによって対応関係を保っているが、別の実施形態も可能であり、考察される。例えば、エントリ間でデータを移動させるのではなく、エントリ500にストアされているビットとターゲットアレイ208のエントリとが特定の分岐のエントリに対応していてもよい。

【0058】

上記したように、シングルバイトの分岐命令では特殊な場合が存在する。図14は、シングルバイトの分岐命令の予測が見落とされる例を示す。図14に、バイト位置8にシングルバイトのRET命令1501を含む連続する命令バイトの組が示される。分岐マーカービット#4 1503がセットされており、成立予測分岐であるRET命令1501が
20
、命令バイト範囲7~8で終端していることを示す。これ以外に、成立予測分岐はない。ここで、オフセット値に8が指定されたRET命令1501のフェッチアドレスが受信された場合を考える。分岐マーカービット#4は、この命令の分岐予測を作成する必要があることを示している。しかし、予測を作成するためにこの分岐マーカービットが送られる場合に、対応する分岐マーカービット#4は、考慮対象となるマーカービット位置の条件を満たさず(つまり、値が8のオフセットは7とまらない)、分岐マーカービット#4が効果的に無視される。図8の論理に示されるように、分岐マーカービット#4の b_4 が設定されて、ゲート804Eに送られる。しかし、「オフセット7」の条件が真とならないため、ゲート808Eに結合されている信号808Eがセットされない。したがって、図8の論理は、分岐予測を示さない。一般に、この予測の見落としの例は、シングルバイトのRET命令が命令バイト位置2, 4, 6, 8, 10, 12, 14のいずれにある場
30
合にも発生する可能性がある。この特別な場合に対処するための様々な実施形態およびメカニズムを以下に記載する。

【0059】

分岐命令の情報

図15を参照すると、分岐予測情報450の一実施形態が示される。上記のように、連続する命令バイト50の各組は、ローカル予測記憶域206にストアされている、対応する分岐マーカービット52と分岐予測情報450とのセットを有する。本図の実施形態においては、分岐予測情報450は、500A~500Cの3つの分岐命令情報を有する。500A~500Cの各々は6ビットであり、成立予測分岐に対応する情報を提供している。図15に示すように、分岐予測情報には、動的ビット502、エンド調整ビット50
40
4、および分岐命令の種類およびサイズの分類に使用する4つのビット506が含まれる。動的ビット502は、ローカル予測記憶域206にあるローカル分岐予測ではなく、グローバル予測記憶域205にあるグローバル分岐予測を使用すべきことを示し得る。エンド調整ビット504は、対応する分岐命令が偶数バイトで終端しているか奇数バイトで終端しているかを示し得る。最後に、分岐情報506は、分岐命令の種類およびサイズの分類に使用され得る。

【0060】

図15には、分岐情報506のエンコーディング(encodings、コード化)の例が4つ記載されている。本図に示すように、各分岐命令には、エンドビット504(本例ではビット位置4)が含まれる。第1のコード508では、第2ビットおよび第3ビットがそれ
50

ぞれ「0」、「0」に設定されることで、条件命令または無条件分岐命令を表し得る。第2のコード510では、第3ビットおよび第2ビットがそれぞれ「0」、「1」に設定されることで、call命令を表し得る。第3のコード512では、ビット3~0が「1000」に設定されることで、return型の命令を表し得る。最後に、第4のコード(514)では、ビット3を「1」に、ビット2~0を「000」以外の値に設定することで、偶数バイトで終端する1バイトのreturnオペコードを表し得る。最初の2つのコード508~510には、命令の任意の即値データまたは変位データのサイズを示すフィールド(ビット1および0)もそれぞれ含まれる。一実施形態においては、1, 2, 4バイトのデータは、コード化ビット1, 0を、それぞれ「00」、「01」、「10」に設定することによって表され得る。最後に、図15に示す第4のコード514には、シングルバイトのreturn命令が現れる、組50内の0以外の偶数命令バイトを示すフィールド(ビット2~0)が含まれている。上記の実施形態においては、第4のコード514の位置フィールドは、「001」~「111」の7つの値を取り得、これらは、バイト位置2, 4, 6, 8, 10, 12, 14にそれぞれ対応している。上記した分岐情報の使用方法を以下に記載する。

10

【0061】

予測論理

図16を参照すると、予測論理の一実施形態220が示される。図16には、比較回路系802、ローカル予測回路900、ターゲット選択回路1100、および制御回路1102が記載されている。比較回路802はフェッチアドレスのオフセットを受信するように結合され、かつ予測回路900およびターゲット選択回路1100に信号1704を送るようさらに結合されている。ターゲット選択回路1100および予測回路900は、ローカル予測記憶域206から分岐予測のエントリに関する情報を受信するように結合されている。制御回路1102は、予測回路900、ターゲット選択回路1100、ローカル予測記憶域206、およびグローバル予測記憶域205に結合されている。一般に、比較回路802は、上記のようにフェッチアドレスを受け取って回路900, 1100に信号1704を送る。予測回路900は、ローカル予測記憶域206から受け取った分岐マーカービットおよび回路802から受け取った信号に基づいて分岐予測が作成されるかどうかを判定するように構成されており、得られた予測1710を制御回路1102に送る。また、ターゲット選択回路1100は、ローカル予測記憶域206から受け取った分岐マーカービットを使用して、分岐成立の際にどの分岐ターゲットを使用すべきかを示す情報(indication)1706を制御回路1102に送る。また、制御回路は、分岐ターゲット情報1714を受け取って、フェッチアドレスを選択する信号240をマルチプレクサ202に送る。さらに、一実施形態においては、回路1102は、分岐アドレス計算器270に分岐予測250を送る。また、本実施形態には特定の分岐について別の予測を提供し得るグローバル予測メカニズムが記載されているが、本メカニズムは、グローバル予測メカニズムがなくても使用することができる。

20

30

【0062】

図16に示すターゲット選択回路1100は一般に、複数のターゲットのうちから、分岐予測に使用するターゲットを1つ選択するように構成され得る。一実施形態においては、ターゲット選択回路1100は、受け取ったフェッチアドレスのオフセットに先行するオフセットに対応する命令バイトに対して設定された分岐マーカービットの個数を決定し得る。この情報に基づいて、1番目~3番目の分岐ターゲットが選択され得る(3つの分岐ターゲットが保持される一実施形態の場合)。例えば、可能性のある3つの分岐ターゲットを保持でき、値が6のオフセットを受け取る場合を考える。回路1100は、分岐マーカービット6に先行する分岐マーカービットの範囲(つまり、分岐マーカービット0~5)内で、設定されている分岐マーカービットの個数を決定し得る。この範囲に、設定された分岐マーカービットがない場合は、この分岐が、1番目の成立予測分岐であると識別され、1番目のターゲットが選択される。分岐マーカービットがこの範囲で1つだけ設定されている場合、この分岐は2番目の成立予測分岐命令であると識別され、ターゲット2

40

50

が選択される。2つの分岐マーカビットが設定されている場合、ターゲット3が選択される。

【0063】

ターゲット選択回路1100の実装方法として、数多くの方法が考えられるが、図17に動的論理を使用した実施形態の一例1800を示す。図17の実施形態においては、トランジスタの行1202~1216および列1220~1234が、受け取った分岐マーカビットとオフセットアドレスとに基づいてターゲットが選択され得るように構成されている。回路1800では、T1 1212、T2 1214およびT3 1218の3通りのターゲット選択が可能である。一般に、回路系1800は、受け取った分岐マーカビットと、比較回路802から受け取った信号とを使用して、特定のターゲットが検出されると、これを考慮対象から除外する。例えば、フェッチアドレスのオフセットに先行して1つの分岐マーカビットが出現することが検出されると、ターゲット1が考慮対象から除外されてターゲット2が選択される。フェッチアドレスのオフセットに先行して2つの分岐マーカビットの組が出現することが検出されると、ターゲット3が選択される。当業者は回路1800の動作を容易に理解するであろうが、以下に概略を記載する。

10

【0064】

図17の第1の行1202は、トランジスタ対1220~1234からなり、これらは受信したオフセットに先行して、セットされている分岐マーカビットがあるかどうかを判定する。標準では、ターゲットT1 1212の値は、真すなわち2進数の「1」である。T1 1212の値は、論理ANDゲートの反転された入力にもなり、T2 1214の値が提供される。したがって、T1 1212の値が2進数の「1」の場合、T2は2進数の「0」となる。行1202の各トランジスタ対は、このフェッチアドレスオフセットに対応する分岐マーカビットと比較回路802からの信号とを受け取る。例えば、列1220、行1202のトランジスタ対は、分岐マーカビット0 (b_0) に対応する第1の入力と、受け取ったオフセットアドレスが0より大きいかどうか(つまり、受け取ったオフセットアドレスを「a」で表すと $a > 0$) を示す第2の入力とを受け取る。2つの入力が共に2進数の「1」である場合、ターゲットT1 1212の値は2進数の「0」となり、受け取ったフェッチアドレスに先行して成立予測分岐が存在することを示し、ターゲットT1 1212が予測されたターゲットとして考慮対象から除外される。残りの行1204~1216は、現在のオフセットに先行して2つの分岐マーカビットがセットされているかどうかを判定するように構成されており、2つの分岐マーカビットがセットされている場合、ターゲットT2 1214が考慮対象から除外される。行1204は、分岐マーカビット b_0 と、分岐マーカビット $b_1 \sim b_7$ とを比較する。行1206は、分岐マーカビット b_1 と分岐マーカビット $b_2 \sim b_7$ とを比較し、以下、同様の比較を行う。現在のオフセットに先行して2つの分岐マーカビットがセットされていることが検出された場合、ターゲットT3 1218の値は「1」となり、ターゲットT2 1214の値は「0」となる。しかし、先行する2つの分岐マーカビットがセットされていることが行1204~1216によって検出されなければ、ターゲットT2 1214の値は2進数の「1」となる。既に説明したように、回路系1800は例示的なものに過ぎない。対応するターゲットの選択方法として数多くの方法が可能であり、考

20

30

40

【0065】

図18を参照すると、制御回路1102の一実施形態が示される。図18には、エンコーダ回路系1820およびマルチプレクサ1832, 1830が記載されている。マルチプレクサ1832は、予測回路900からローカル予測1710を、グローバル予測記憶域205からグローバル予測1712を受信するように結合されている。マルチプレクサ1830は、3つの分岐命令に対応する分岐情報1714をローカル予測記憶域206から受け取るように結合されている。さらに、マルチプレクサ1830は、ターゲット選択回路1100からターゲット選択信号1706を受け取るように結合されている。選択信号1706は、マルチプレクサ1830からエンコーダ回路系1820に送る分岐情報1

50

840を選択する。さらに、本図の実施形態においては、ローカル予測1710かグローバル予測1712かのいずれを予測1806としてマルチプレクサ1832からエンコーダ回路系1820に送るかを選択するために、分岐情報1840に含まれる動的ビット1808が使用される。エンコード回路系1820は、分岐情報1840および予測1806を使用して、分岐が「分岐成立」と予測されているか「分岐不成立」と予測されているかを決定し得る。「分岐成立」と予測されている分岐がない場合、エンコーダ回路系1820は、マルチプレクサ212から連続アドレス232を送る選択信号240をエンコードするように構成され得る。別法として、エンコーダ回路系1820が、戻りアドレス230かターゲットアレイ208から受信した分岐ターゲットアドレスかを選択するために信号240をエンコードしてもよい。本図の実施形態においては、ローカル予測1710かグローバル予測1712かを選択するために、動的ビット1808が使用され得る。一実施形態においては、分岐が、比較的予測不可能な挙動を示す場合、グローバル分岐予測履歴を使用して、その分岐の予測精度を向上させることができる。この場合、ローカル分岐予測ではなくグローバル分岐予測を使用すべきことを示すように動的ビットが設定され得る。

10

【0066】

分岐アドレスの計算

上記したように、一実施形態においては、信号240によって選択されてマルチプレクサ212によって送信されるフェッチアドレス243は、所定のフェッチアドレスに関し、分岐予測ユニット16によって送信される最終的な分岐予測を提供し得る。しかし、別の実施形態においては、分岐予測ユニット14は、初期の段階で予測ミスを検出するために、分岐アドレスの計算を実行するように構成される。図19は、分岐アドレス計算器270のブロック図である。本図の実施形態においては、分岐アドレス計算器270は、マルチプレクサ222からアドレス236を、ローカル予測記憶域206から予測情報242を、命令キャッシュ16から命令バイトの組241を、マルチプレクサ212からターゲットアドレス243を受け取るように構成されている。分岐アドレス計算器270はフェッチアドレス245を送るようにも構成されている。

20

【0067】

一般に、分岐アドレス計算器270は、受け取ったフェッチアドレス236と分岐命令の即値データ241とに基づいて、相対分岐ターゲットアドレスを計算するように構成されている。次に、新たに求めたターゲットアドレスと予測されたターゲットアドレス243とが比較される。新たに求めたターゲットアドレスと予測されたターゲットアドレス243とが一致しない場合、予測ミスが検出されて、新たに求めたターゲットが次のフェッチアドレス245として送られる。分岐予測ミスが早期に検出されて、新しいフェッチアドレスを早いうちに計算できるため有利である。

30

【0068】

図19の実施形態においては、予測された分岐命令に対応する情報が、バス1850を介してローカル予測記憶域206から送られる。一実施形態においては、分岐情報1850には、命令バイトの組にある、対応する分岐命令のオフセットアドレスが含まれる。さらに、命令バイトの組241が、命令キャッシュ16からマルチプレクサ1406に送られる。受け取ったオフセットと、分岐情報1850に含まれるエンドビット504情報とが使用されて、選択信号1420がマルチプレクサ1406に送られ、対応する分岐命令がマルチプレクサ1406からマルチプレクサ1408に送られ得る。さらに、受け取った情報1850に含まれる受け取った分岐情報506が使用されて、マルチプレクサ1408に信号1422が送られ、マルチプレクサ1408は、予測された分岐命令に対応する即値データ1424を選択して送る。フェッチアドレス236が、マルチプレクサ1414と、フェッチアドレス236からベースアドレスを減算するように構成された回路1402とに送られる。一実施形態においては、ベースアドレスのベースは、86アドレス変換メカニズムの対応する部分によって指定されるアドレスに対応している。回路1402は、ベースを減算して、求めたアドレスを回路1404に送る。回路1404は、この

40

50

アドレスに、マルチプレクサ 1408 から受け取った即値データ 1424 を加算し、この結果 1413 を回路 1410 に送るように構成されている。

【0069】

上記に加え、予測されたターゲット 243 がマルチプレクサ 1414 および回路 1412 に送られる。回路 1412 は、予測されたフェッチアドレスからベースアドレスを減算してフェッチアドレスを調整し、得られたアドレス 1415 を回路 1410 に送るように構成されている。回路 1410 は、計算によって求めたアドレス 1413 と調整されたフェッチアドレス 1415 とを受け取ると、両者を比較するように構成されている。回路 1410 は、2つのアドレスが等しくない場合、フェッチアドレス 236 を選択して、マルチプレクサ 1414 から送信させる。回路 1430 は、両者が一致しない場合に、フェッチアドレス 236 を送るように構成され得る。回路 1430 は、計算によって求めたフェッチアドレスと調整済みのフェッチアドレスとが等しい場合に、新しいフェッチアドレスを送信しないように構成され得る。さらに、回路 1430 は、「分岐不成立」の予測を示す予測信号 250 を検出すると、新しいフェッチアドレスを送信しないように構成され得る。

10

【0070】

一実施形態においては、フェッチアドレス 236 が受信されると、分岐予測ユニット 14 によって2つのフェッチアドレスが決定されて送信され得る。マルチプレクサ 212 から送られる第1のフェッチアドレスは、命令キャッシュ 16 から命令をフェッチするための新しいフェッチアドレスとして直ちに使用され得る。同時に、マルチプレクサ 212 から送られるフェッチアドレスは、分岐アドレス計算器 270 にも送られ、上記の比較が実行される。分岐アドレス計算器 270 は、マルチプレクサ 212 経由のフェッチアドレスと、分岐アドレス計算器 270 が上記のように算出したフェッチアドレスとが一致しない場合、命令キャッシュ 16 から命令をフェッチするために使用する第2のフェッチアドレスを送る。

20

【0071】

分岐予測犠牲キャッシュ

上記したように、犠牲キャッシュ 260 を使用して、ローカル予測記憶域 206 から排除された分岐予測情報がストアされ得る。図 20 は、ローカル予測記憶域 206 と犠牲キャッシュ 260 との相互作用 (interaction) の一実施形態を示すブロック図である。図 20 には、ローカル予測記憶域 206 および犠牲キャッシュ 260 のほかに、I (命令) キャッシュ 16、L2 キャッシュ 1828 およびデコーダ 400 が記載されている。本図の実施形態においては、犠牲キャッシュ 260 は、ローカル予測記憶域 206 から排除された分岐予測情報をストアするように構成され、L2 キャッシュ 1828 は、プログラム命令を含め、データをストアするように構成されている。

30

【0072】

ローカル予測記憶域 206 からエントリが排除される際には、排除されるエントリに対応する分岐マーカービットと動的ビットとが、バス 2108 を介して犠牲キャッシュ 260 に送られ、犠牲キャッシュ 260 にストアされる。エンド調整ビット 504、分岐情報 506 など、排除されるエントリに対応する他の分岐予測情報は破棄される。同様に、本図の実施形態においては、I キャッシュ 16 から排除されるデータが、バス 2106 を介して送られ、L2 キャッシュ 1828 にストアされる。続いて、ローカル予測記憶域 206 および I キャッシュ 16 にフェッチアドレス 236 が提示されると、提示されたフェッチアドレスに対応する分岐予測のエントリが、ローカル予測記憶域 206 に存在するかどうか判定される。対応するエントリがローカル予測記憶域 260 にない場合、犠牲キャッシュ 260 に、対応するエントリが要求され得る。対応する有効なエントリが犠牲キャッシュ 260 で検出されると、分岐マーカービットおよび動的ビットが、バス 2102 を介してデコーダ 400 とローカル予測記憶域 206 とに送られる。ローカル予測記憶域は、受け取った分岐マーカービットおよび動的ビットを、これらに対応するエントリにストアするように構成されている。さらに、フェッチアドレス 236 に対応する命令の組が、

40

50

バス 2104 を介してデコーダ 400 と I キャッシュ 16 とに送られる。

【0073】

次にデコーダ 400 は、受け取った命令と分岐マーカービットとを使用して、ローカル予測記憶域 206 に分岐予測のエントリの残りの部分を再生成し得る。デコーダは、バス 2102 を介して受け取った分岐マーカービットを使用して、バス 2104 から受け取った命令の組内での成立予測分岐の位置を特定し得る。さらに、デコーダ 400 は、特定した成立予測分岐をデコードして、分岐の種類（無条件分岐、条件付き分岐など）、分岐の終端バイトが偶数バイトか奇数バイトかのほか、分岐命令に対応する即値データがあればそのサイズを決定し得る。デコーダ 400 は、この決定の結果を基に、ローカル予測記憶域 206 に、対応する分岐予測のエントリの残りの部分を再生成し得る。本図の実施形態 10
 においては、分岐予測のエントリの残りの部分は、分岐情報 506 およびエンド調整ビット 504 に該当する。対応する命令の組を実行することなく、ローカル予測記憶域 206 に分岐予測の完全なエントリを再生成することができ有利である。さらに、分岐予測のエントリの一部のみを犠牲キャッシュ 260 にストアするだけで済む。このため、犠牲キャッシュ 260 のサイズを小さくすることができる。

【0074】

図 21 に、ローカル予測記憶域 206 への分岐予測のエントリの再生成をさらに詳細に示す。図 21 には、犠牲キャッシュ 260、デコーダ 400、およびローカル予測記憶域 206 のエントリ 2100 が記載されている。本図に示すように、分岐マーカービット 2120 および動的ビット 2122 はデータを構成しており、このデータは、バス 2108 20
 を介して犠牲キャッシュへと排除されるほか、バス 2102 を介して犠牲キャッシュ 260 からリトリブされる。分岐情報 2130 は、分岐情報 506 およびエンド調整ビット 504 に対応するデータを含む。分岐情報 2130 は、ローカル予測記憶域 206 からエントリ 2100 が排除される際に破棄される。上記したように、デコーダ 400 は、分岐情報 2130 を再生成し、2110 を介して再生成されたデータを送るように構成されている。

【0075】

上記したように、フェッチアドレス 236 が、ローカル予測記憶域 206 および I キャッシュ 16 の双方に提示される。しかし、別の実施形態においては、フェッチアドレス 236 が、ローカル予測記憶域 206 および I キャッシュ 16 に加えて、犠牲キャッシュ 2 30
 60 および L2 キャッシュ 1828 の少なくともいずれかに提示されてもよい。さらに、L2 キャッシュ 1828 が、上記したようにローカル予測記憶域 206 から排除されたデータをストアするように構成される実施形態もあり得る。図 22 に、L2 キャッシュ 1828 が、プログラム命令と、対応する分岐予測情報とをストアするように構成されている実施形態を示す。本図の実施形態においては、ローカル予測記憶域 206 は、バス 2206 を介して排除された分岐予測情報を L2 キャッシュ 1828 に送るように構成されている。I キャッシュ 16 は、排除されたプログラム命令を、バス 220 を介して L2 キャッシュ 1828 に送信するように構成され得る。プリフェッチユニット 12 は、L2 キャッシュ 1828 からプログラム命令と分岐予測情報の両方をフェッチするように構成されている。プリフェッチユニットは、L2 キャッシュ 1828 からフェッチした分岐情報およ 40
 びプログラム命令に基づいて、分岐予測のエントリを完全に再生成するようにさらに構成されている。次に、分岐予測の完全なエントリが、バス 2202 を介してローカル予測記憶域 206 に送られると共に、対応するフェッチ済みのプログラム命令が、バス 2204 を介して I キャッシュ 16 に送られ得る。

【0076】

見落とされた予測の検出

上記したように、シングルバイトの RET 命令が関与する特別な場合がある。図 14 に示すように、特定の位置に現れる、「分岐成立」と予測されたシングルバイトの RET 命令が、予測論理 220 によって見落とされ、その後「分岐不成立」と予測される可能性がある。下記に、このような見落とされた予測を検出するように構成された分岐予測ユニッ 50

ト14の一実施形態を記載する。図23は、分岐予測ユニット14の一実施形態を示すブロック図である。図23の実施形態は図2の実施形態と類似している。しかし、図23の実施形態においては、戻りアドレススタック230が、マルチプレクサ221Aに結合されているほか、分岐アドレス計算器270と比べて分岐アドレス計算器270Aが変更されており、信号223Aは、マルチプレクサ221Aへの3つの入力の中の1つを選択するように構成されている。図23の実施形態は、全般に前の例と同じように動作する。しかし、分岐アドレス計算器270Aは、シングルバイトのRET命令に対応する見落とされた予測を検出すると、戻りスタック230から送られるフェッチアドレスを選択するように構成されている。

【0077】

図24に、分岐アドレス計算器270Aの一実施形態を示す。分岐アドレス計算器270Aは、分岐アドレス計算器270と同様に構成されており、上記したように第2のフェッチアドレスを計算し、バス245を介してこれを送信し得る。しかし、分岐アドレス計算器270のように回路1430からアドレス245および制御信号223を送るのではなく、回路1430は、可能性のある第2のフェッチアドレス2410を制御回路2400に、選択信号2412をマルチプレクサ2402に送信するように構成されている。さらに、分岐アドレス計算器270Aは、シングルバイトのRET命令に対応する、見落とされた予測を検出するように構成された見落予測回路2450を備える。見落予測回路2450は、ローカル予測記憶域206から分岐ターゲット情報450を、予測論理220から予測250を受け取るほか、フェッチアドレス236を受信するように結合されている。さらに、回路2450は、選択信号2420, 2422をマルチプレクサ2402に送るように結合されている。信号2422は、制御回路2400にさらに結合されている。

【0078】

見落予測回路2450は、見落とされた予測を検出すると、マルチプレクサ221Aから送られる戻りアドレスを選択するように構成された選択信号2420と、見落とされた予測の検出を示し、信号2420を選択してマルチプレクサ2402から出力させるように構成された選択信号2422とを送信する。一実施形態においては、見落予測回路2450は、予測250が「分岐成立」の予測を示す場合には必ず、信号2412を選択してマルチプレクサ2402から送信させるように構成され得る。このように、予測論理220が「分岐不成立」の予測を示す場合には、見落予測回路2450が、見落とされた予測の検出のみを実行してもよい。マルチプレクサ2402は、信号2460を送るよう構成されており、信号2460は、第2のフェッチアドレス245を選択してマルチプレクサ221Aから送信させるようにも、戻りアドレスを選択してマルチプレクサ221Aから送信させるようにも構成され得る。

【0079】

制御回路2400は、予測ミスまたはRET命令の見落とされた予測のために、第2のフェッチアドレスが必要であるかどうかを判定し、対応する情報をバス245および信号223Aを介して送るよう構成されている。一実施形態においては、信号2410には、フェッチアドレスと、回路1410による比較の結果が不一致となったことを示す情報が含まれる。本図の実施形態においては、制御回路2400は、2つの条件のうちの1つを検出すると、第2のフェッチアドレスが必要であることを判定し得る。第2のフェッチアドレスが必要となる第1の条件は、「分岐成立」の予測250を検出し、かつ信号2410から、回路1410による比較の結果が不一致となったことを検出することである。制御回路は、第1の条件を検出すると、受け取ったアドレス2410をバス245を介して送るほか、対応する選択信号223Aを送るよう構成されている。この場合、選択信号223Aは、回路1430から送られる信号2412に対応しており、信号245を選択してマルチプレクサ221Aから送信させるよう構成されている。第2のフェッチアドレスを要求する第2の条件は、「分岐不成立」の予測250を検出し、かつ信号2422から、見落とされた予測が検出されたことを検出することである。制御回路2400

10

20

30

40

50

は、第2の条件を検出すると、選択信号223Aを送り、これは見落予測回路2450によって送られる信号2420に対応している。第2の条件の場合、送信された選択信号223Aは、戻りアドレス230を選択してマルチプレクサ221Aから送信させるように構成されている。制御回路2400によって第1の条件も第2の条件も検出されない場合、第2のフェッチアドレスが送られることはない。

【0080】

図25は、見落予測回路2450の一実施形態を示すブロック図である。見落予測回路2450は、候補回路2502、偶数バイト回路2504、照合回路2506、および制御回路2508を備える。候補回路2502は、分岐情報450を受信するように結合され、偶数バイト回路2504は、フェッチアドレス236を受信するように結合され、制御回路2508は、予測250を受信するように結合されている。さらに、照合回路2506は、候補回路2502から信号2520を、偶数バイト回路2504から信号2522をそれぞれ受信するように結合されている。制御回路2508は、照合回路2506から信号2530を受信するように結合されている。

10

【0081】

図25の実施形態においては、候補回路2502は、分岐情報450の偶数バイト位置に、RET命令に対応する有効なエントリが含まれるかどうかを判定するように構成されている。上記したように、分岐情報450は、500A~500Cの3つの分岐命令に対応する分岐情報を含み得る。図15に示すコードを使用して、RET命令が、分岐情報ビット#3が2進数の「1」である命令として特定され得る。さらに、エンド調整ビットによって、偶数バイトで終端する命令が検出され得る。次に、偶数バイトのRET命令に対応していると判定された分岐情報500A~500Cが、見落とされた予測の候補とみなされ、候補を検出したことを示す肯定的な指示(positive indication)と共に照合回路2506に送られ得る。偶数バイト回路2504は、偶数アドレスに該当するフェッチアドレスを決定するように構成されている。一実施形態においては、偶数バイト回路2504は、この判断を行うために、受け取ったフェッチアドレス236の最下位ビットを比較し得る。フェッチアドレス236の最下位ビットが2進数の「0」である場合、偶数フェッチアドレスとして判定され、対応する肯定的な指示2522が照合回路2506に送信され得る。

20

【0082】

照合回路2506は、回路2502, 2504から肯定的な指示を受け取ると、フェッチアドレス236に対応する、見落とされた予測の候補を検出するように構成されている。一実施形態においては、フェッチアドレス236のオフセットのビット3~1と、候補の分岐情報のビット2~0とが比較される。上記したように、RET命令の分岐情報506のビット2~0は、RET命令の偶数バイト位置を示している。下記の表に、偶数フェッチアドレスのオフセットと分岐情報のバイト位置との関係を示す。

30

【表 1】

偶数フェッチアドレスの オフセット (ビット 3～0)	分岐情報のバイト位置 (ビット 2～0)
0000	000
0010	001
0100	010
0110	011
1000	100
1010	101
1100	110
1110	111

表 1 に示すように、偶数フェッチアドレスのオフセットのビット 3～1 と、分岐情報 506 のビット 2～0 とを直接対応付けることができる。このため、照合回路 2506 は、RET 命令に対応する候補分岐情報とフェッチアドレスのオフセットとを照合することができる。照合回路 2506 は、この一致を検出すると、一致を検出したことを示す信号 2530 を制御回路 2508 に送るように構成されている。ここに示す例示的な実施形態では、制御回路 2508 は、「分岐不成立」の予測 250 と一致を示す情報 2530 とを検出すると、戻りアドレスを選択してマルチプレクサ 221A から送信させるように構成された信号を送信するように構成されている。制御回路 2508 は、戻りアドレス 230 を選択してマルチプレクサ 221A から送信させるために、信号 2422 および信号 2420 を送る。信号 2420 には、戻りアドレス 230 を選択してマルチプレクサ 221A から送信させるためのコードが含まれる。選択信号 2422 は、信号 2412 か信号 2420 かを選択してマルチプレクサ 2422 から出力させるように構成されている。信号 2422 は、「分岐不成立」の予測 250 と一致を示す情報 2530 とを検出すると、信号 2420 を選択してマルチプレクサ 2402 から出力させる。これらが検出されない場合、信号 2422 は、信号 2412 を選択してマルチプレクサ 2402 から出力させる。

【0083】

図 26 に、バスブリッジ 2602 を介して種々のシステム部品に結合されたプロセッサ 10 を備えたコンピュータシステム 2600 の一実施形態のブロック図が示される。別の実施形態も可能であり、考慮される。図中のシステムにおいて、メインメモリ 2604 は、メモリバス 2606 を介してバスブリッジ 2602 に結合され、グラフィックコントローラ 2608 は、AGP バス 2610 を介してバスブリッジ 2602 に結合されている。最後に、複数の PCI 機器 2612A～2612B が、PCI バス 2614 を介してバスブリッジ 2602 に結合されている。さらに、EISA/ISA バス 2620 を介して 1 つ以上の EISA 機器または ISA 機器 2618 を電気的に接続できるように、セカンダリバスブリッジ 2616 を設けてもよい。プロセッサ 10 は、CPU バス 2624 を介してバスブリッジ 2602 に結合されているほか、自由選択で設けられる L2 キャッシュ 1828 にも結合されている。

【0084】

バスブリッジ 2602 は、プロセッサ 10、メインメモリ 2604、グラフィックコントローラ 2608 および PCI バス 2614 に接続された機器の間のインターフェイスを提供している。バスブリッジ 2602 に接続された機器のうちの 1 つから操作 (operation) が受信されると、バスブリッジ 2602 は、この操作のターゲット (例えば、特定の

10

20

30

40

50

機器が、ターゲットが P C Iバス 2 6 1 4 に接続している場合は P C Iバス 2 6 1 4) を識別する。バスブリッジ 2 6 0 2 は、この操作をターゲット機器に転送する。バスブリッジ 2 6 0 2 は一般に、この操作を、ソースの機器またはバスが使用しているプロトコルから、ターゲットの機器またはバスが使用しているプロトコルに変換する。

【 0 0 8 5 】

セカンダリバスブリッジ 2 6 1 6 は、 P C Iバス 2 6 1 4 に対するインターフェイスを I S A / E I S A バスに提供すると共に、必要に応じて追加の機能も有している。また、必要に応じて、セカンダリバスブリッジ 2 6 1 6 の外側に設けたか、ブリッジ 2 6 1 6 に搭載した入出力コントローラ (図示せず) をコンピュータシステム 2 6 0 0 に設けて、キーボードおよびマウス 2 6 2 2、ならびに様々なシリアルポートおよびパラレルポートの操作をサポートしてもよい。さらに、別の実施形態においては、プロセッサ 1 0 とバスブリッジ 2 6 0 2 との間にある C P Uバス 2 6 2 4 に、外部キャッシュユニット (図示せず) を結合してもよい。別法として、外部キャッシュをバスブリッジ 2 6 0 2 に結合してもよく、この外部キャッシュのためのキャッシュ制御論理をバスブリッジ 2 6 0 2 に搭載してもよい。さらに、 L 2 キャッシュ 1 8 2 8 が、プロセッサ 1 0 のバックサイド構成 (b a c k s i d e c o n f i g u r a t i o n) で記載されている。 L 2 キャッシュ 1 8 2 8 は、プロセッサ 1 0 から独立していても、プロセッサ 1 0 にカートリッジ (例えば、スロット 1 またはスロット A) として搭載されていても、プロセッサ 1 0 と共に半導体基板に搭載されていてもよいという点が注目される。

【 0 0 8 6 】

メインメモリ 2 6 0 4 は、アプリケーションプログラムのストア先であると共に、プロセッサ 1 0 の主な実行元のメモリである。適切なメインメモリ 2 6 0 4 は、 D R A M (ダイナミックランダムアクセスメモリ) を有する。例えば、複数バンク構成の S D R A M (シンクロナス D R A M) のまたはラムバス D R A M (R D R A M) が適切であり得る。

【 0 0 8 7 】

P C I 機器 2 6 1 2 A ~ 2 6 1 2 B の例には、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、ハードディスクドライブ、フロッピーディスクドライブもしくはドライブコントローラ、 S C S I (小型コンピュータシステムインターフェイス) アダプタおよびテレフォニカードなどの様々な周辺機器がある。同様に、 I S A 機器 2 6 1 8 の例には、モデム、サウンドカード等の様々なタイプの周辺機器のほか、種々のデータ取得カード (G P I B インターフェイスカードまたはフィールドバスインターフェイスカード等) などがある。

【 0 0 8 8 】

グラフィックコントローラ 2 6 0 8 は、ディスプレイ 2 6 2 6 の文字および画像の表示を制御するために設けられる。グラフィックコントローラ 2 6 0 8 は、メインメモリ 2 6 0 4 との間で効率的に移動可能な 3 次元データ構造を表示するために、当業界において一般に公知となっているグラフィックアクセラレータを具体化し得る。このため、グラフィックコントローラ 2 6 0 8 は、バスブリッジ 2 6 0 2 内でターゲットインターフェイスに対するアクセスを要求および受け取って、これによってメインメモリ 2 6 0 4 へのアクセスを取得できるという点で、 A G P バス 2 6 1 0 のマスタであり得る。専用のグラフィックバスによって、メインメモリ 2 6 0 4 から高速にデータを取得することが可能となる。グラフィックコントローラ 2 6 0 8 は、特定の操作のために、 A G P バス 2 6 1 0 で P C I プロトコルトランザクションを生成するようにさらに構成され得る。このため、バスブリッジ 2 6 0 2 の A G P インターフェイスは、 A G P プロトコルのトランザクションと、 P C I プロトコルのターゲットトランザクション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有し得る。ディスプレイ 2 6 2 6 は、画像または文字を表現可能な任意の電子ディスプレイである。適切なディスプレイ 2 6 2 6 には、陰極線管 (「 C R T 」) ディスプレイや液晶ディスプレイ (「 L C D 」) などがある。

【 0 0 8 9 】

上記の説明において A G P バス、 P C I バス、および I S A バスまたは E I S A バスを

10

20

30

40

50

例として採り上げたが、必要に応じて任意のバスアーキテクチャを代用してもよいという点が注目される。さらに、コンピュータシステム2600は、追加のプロセッサ(コンピュータシステム2600の追加部品として示されるプロセッサ10aなど)を備えた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という点が注目される。プロセッサ10aはプロセッサ10と類似のものであってもよい。より詳細には、プロセッサ10aは、プロセッサ10と全く同一であってもよい。プロセッサ10は、独立したバスを介してバスブリッジ2602に結合されていても(図4に示される)、プロセッサ10との間でCPUバス2624を共用してもよい。さらに、プロセッサ10aは、L2キャッシュ1828と類似した追加のL2キャッシュ1828aと結合されてもよい。

【0090】

本議論は、種々の信号のアサートに言及し得るという点が注目される。本明細書で使用されるように、ある信号が特定の条件を示す値を伝達する場合、その信号は「アサートされている」という。これに対して、ある信号が特定の条件がないことを示す値を伝達する場合、その信号は「ディアサートされている」という。信号が論理0値を伝達する場合、または逆に論理1値を伝達する場合、この信号はアサートされていると定義することができる。さらに、上記の説明において種々の値が破棄されていると記載した。値は、さまざまな方法によって破棄され得るが、一般には、この値を受け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が行われる。例えば、値が1つのビットを有する場合、この値の論理状態を反転して値を破棄してもよい。値がnビット値である場合、nビットコードのうちの1つのビットが、値が無効であることを示す値を示し得る。このビットを、

【0091】

上記の開示が完全に理解されれば、数多くの変形例および変更例が当業者にとって自明となるであろう。添付の特許請求の範囲はこのような変更例および変更例を全て包含するものと解釈されることが意図される。

【産業上の利用可能性】

【0092】

本発明は一般に、マイクロプロセッサの分岐予測メカニズムに利用可能である。

【図面の簡単な説明】

【0093】

【図1】マイクロプロセッサの一実施形態を示すブロック図である。

【図2】分岐予測ユニットの一実施形態を示すブロック図である。

【図3】分岐を予測する方法を示すフローチャートである。

【図4】レベル2分岐予測キャッシュの使用法を示すフローチャートである。

【図5】プログラム命令と、対応する分岐予測のエントリとから構成される連続する組を示す図である。

【図6】分岐マーカービットとアドレスオフセットとの関係を示す図である。

【図7】プログラム命令、分岐マーカービットおよびアドレスオフセットの関係を示す図である。

【図8】予測論理の一実施形態を示す図である。

【図9】図8の予測論理から導出した動的論理を示す図である。

【図10】分岐マーカービットの使用を示す図である。

【図11】分岐マーカービットの使用を示す図である。

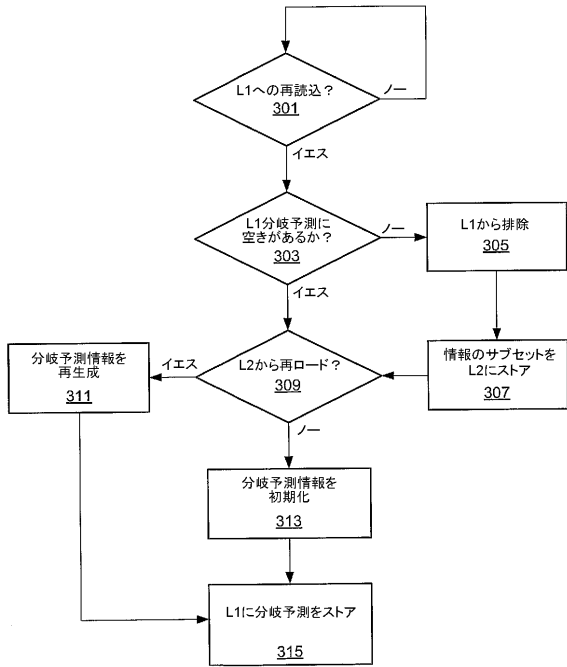
【図12】分岐マーカービットの使用を示す図である。

【図13】分岐マーカービットの使用を示す図である。

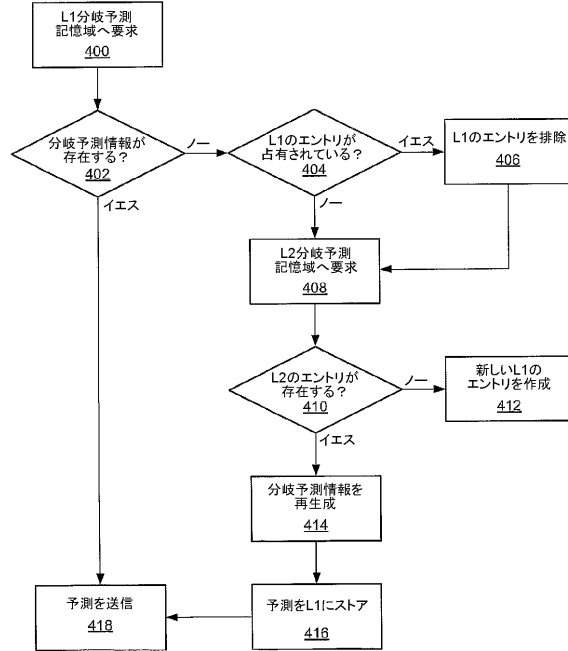
【図14】見落とされた予測を示す図である。

【図15】分岐ターゲット情報を示す図である。

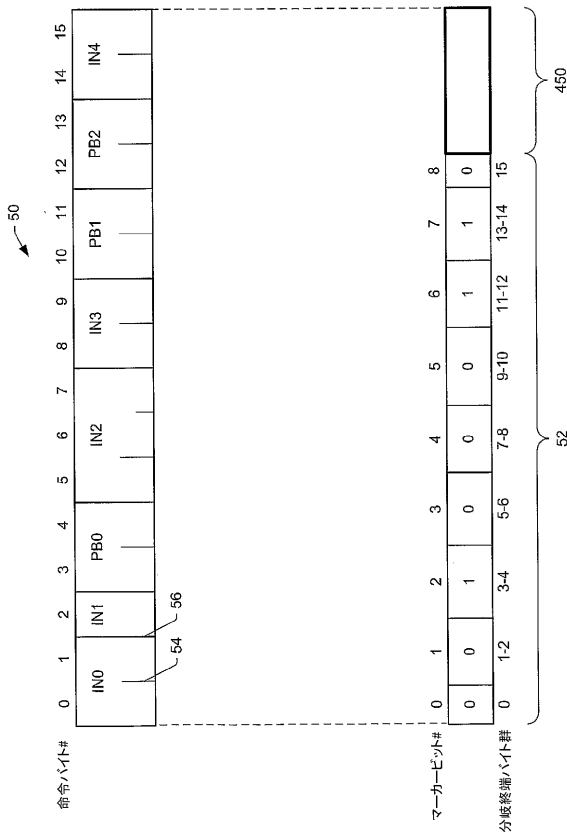
【図3】



【図4】



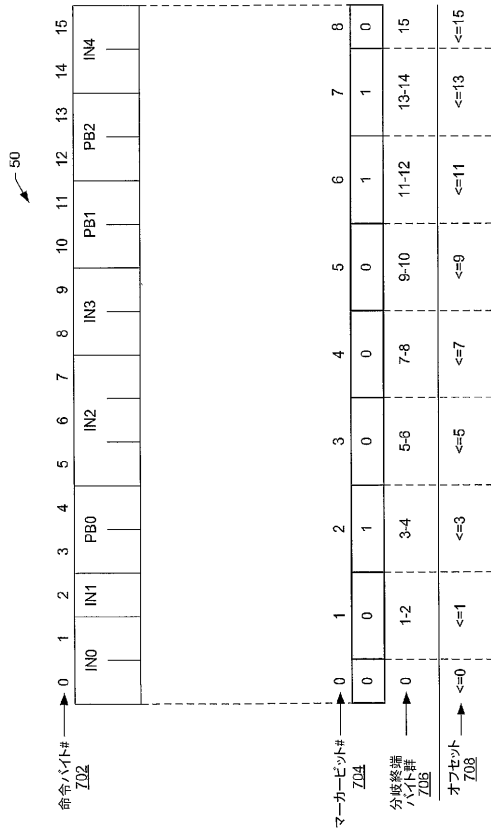
【図5】



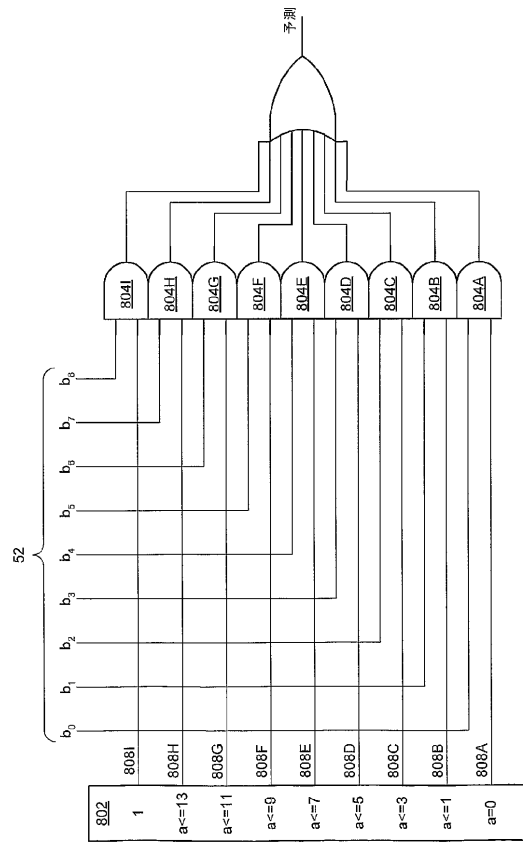
【図6】

602	604	606	608	610	612	614	616	618							
オフセット	<=0	<=1	<=3	<=5	<=7	<=9	<=11	<=13	<=15						
マーカービット#	0	1	2	3	4	5	6	7	8						

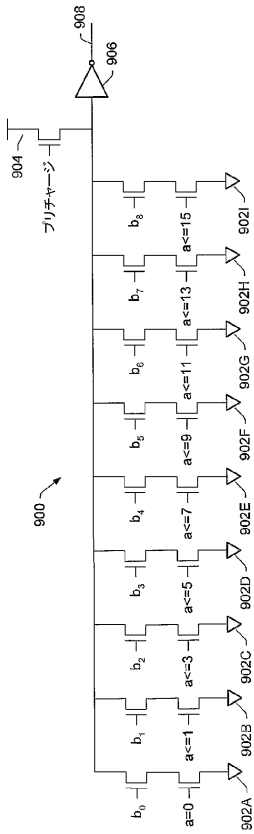
【 7 】



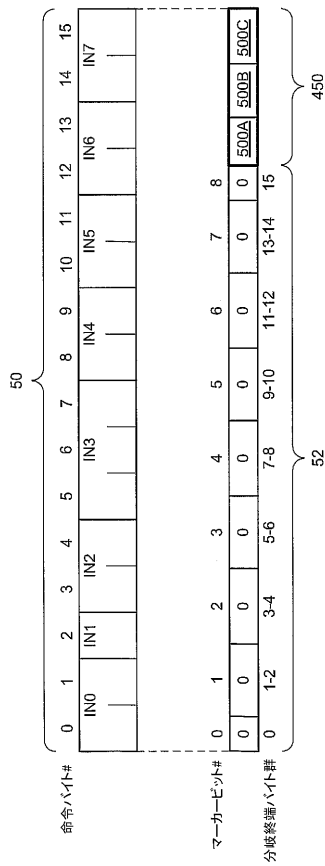
【 8 】

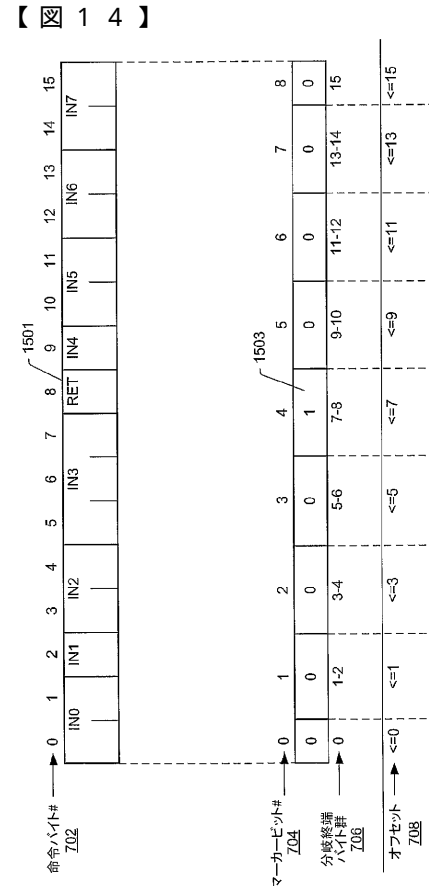
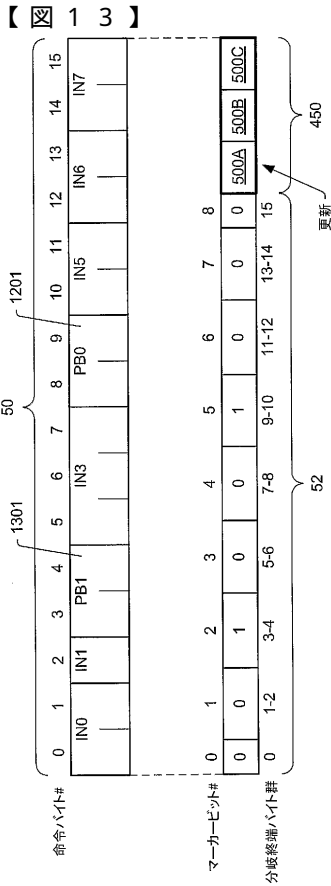
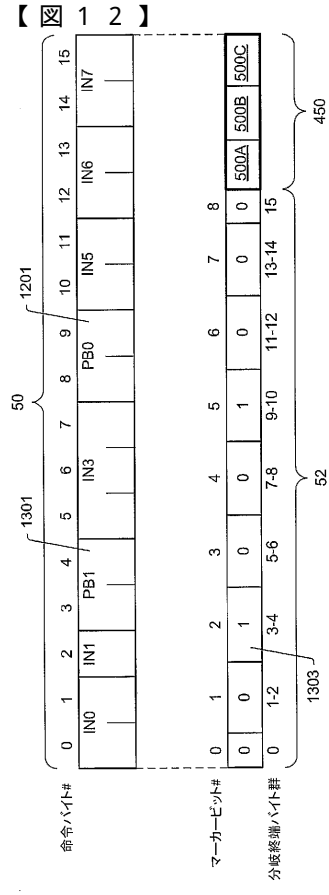
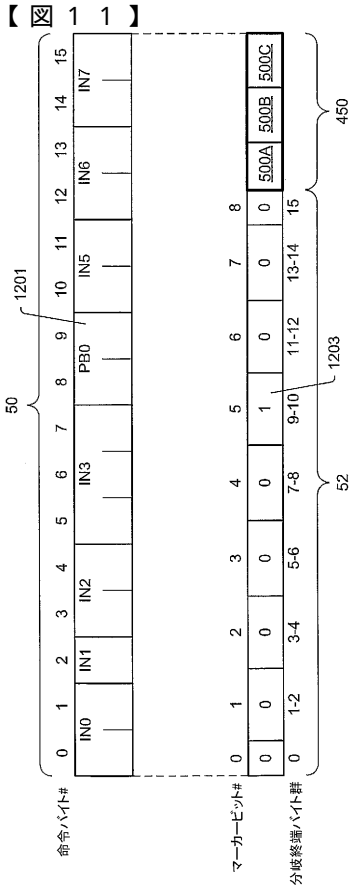


【 9 】

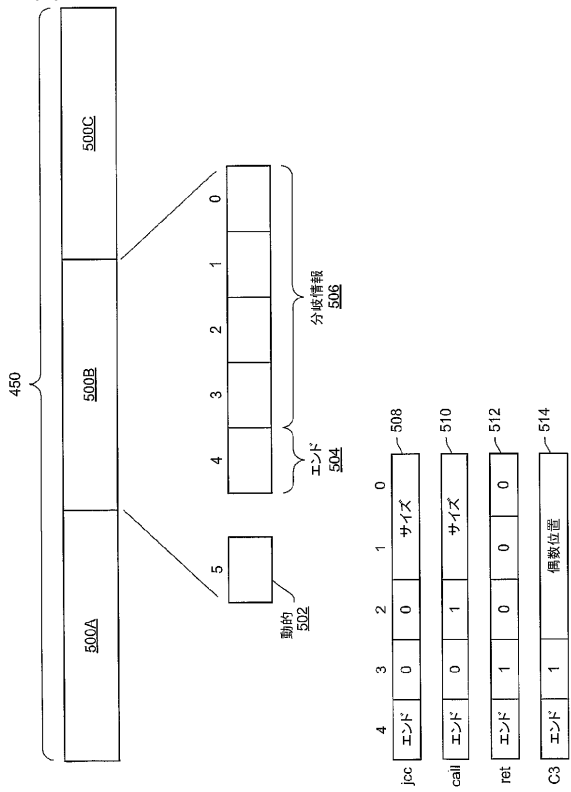


【 10 】

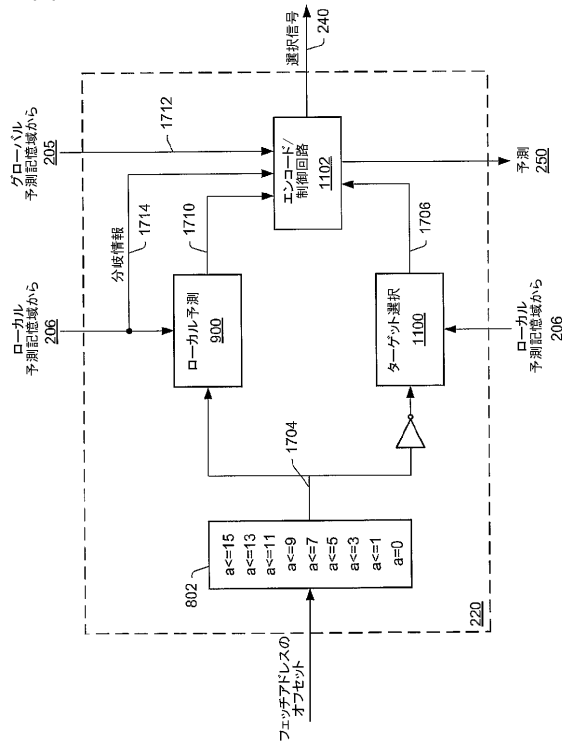




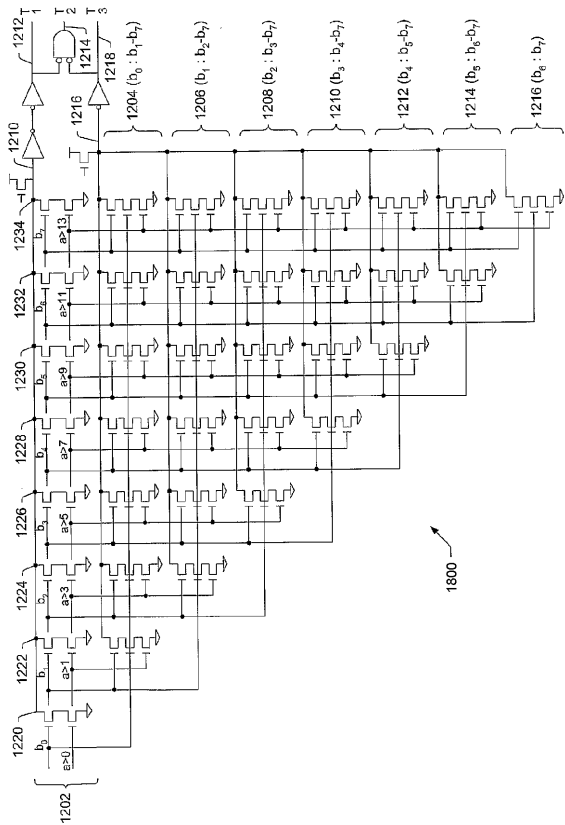
【 図 15 】



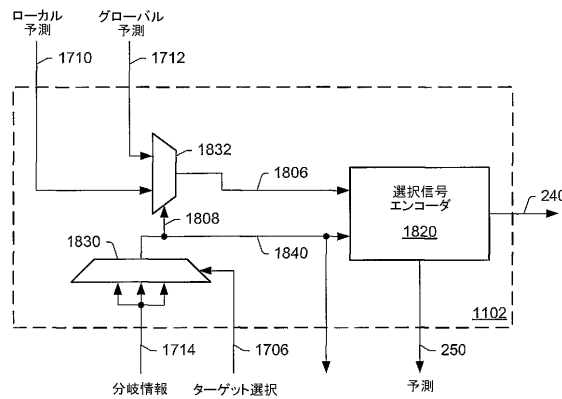
【 図 16 】



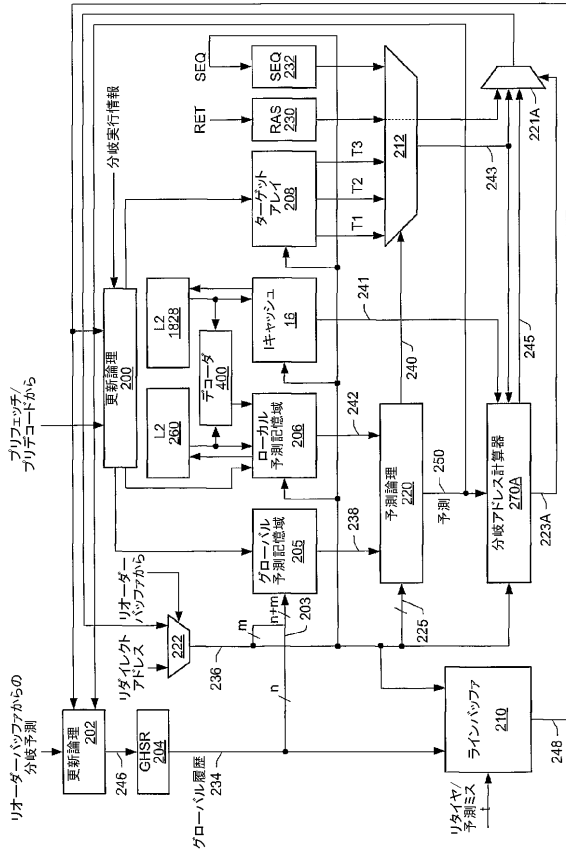
【 図 17 】



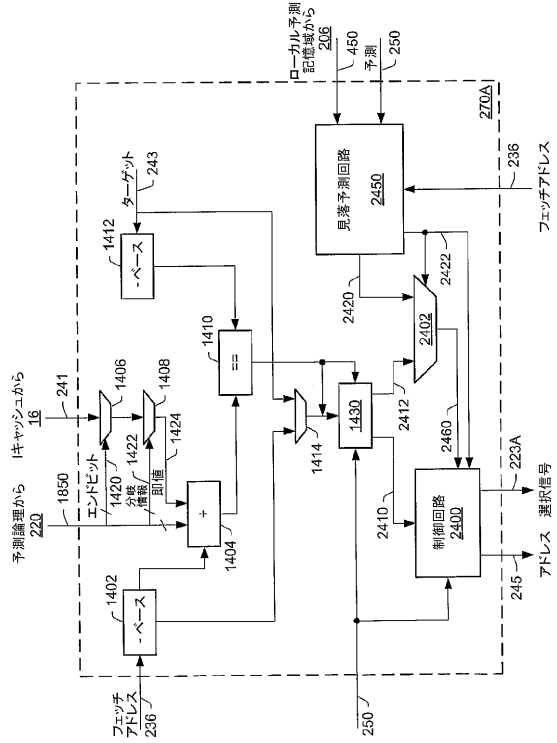
【 図 18 】



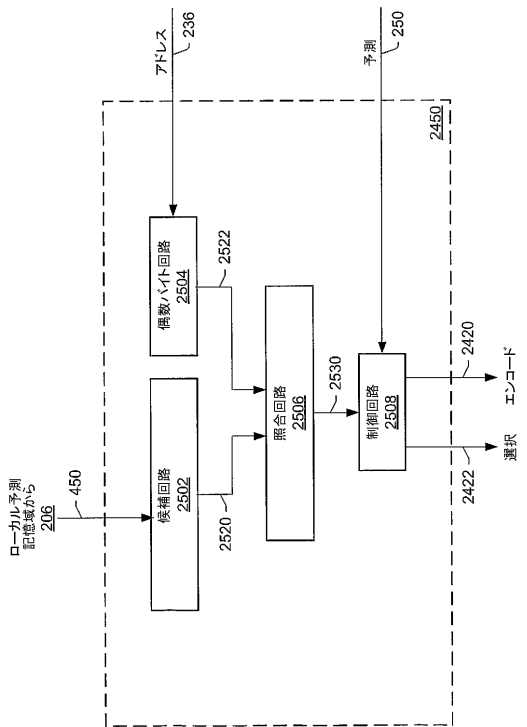
【図 23】



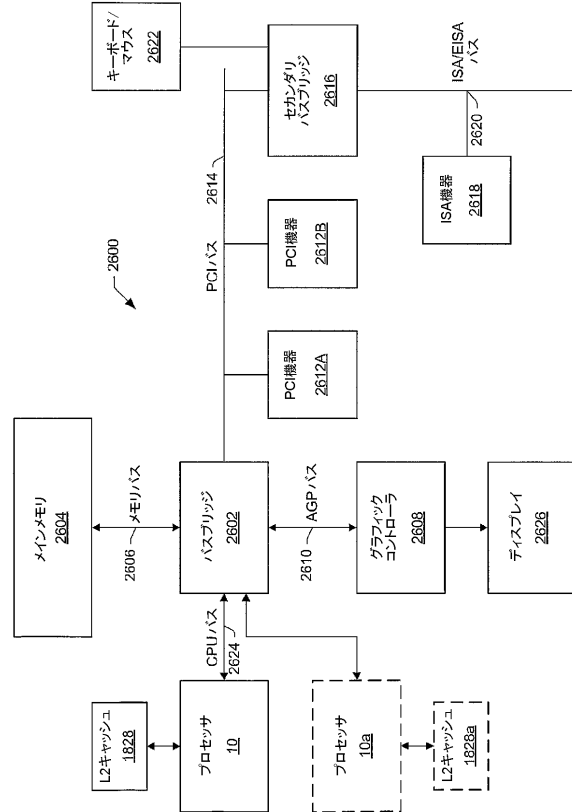
【図 24】



【図 25】



【図 26】



フロントページの続き

(74)代理人 100108604

弁理士 村松 義人

(72)発明者 ジェラルド ディー . ズラスキー

アメリカ合衆国、テキサス州 78726、オースティン、ミード パーク コウブ 8009

(72)発明者 ジェイムズ エス . ロバーツ

アメリカ合衆国、イリノイ州 60004、アーリントン ハイ츠、ナンバー2、パートリッジ
コート 1621

審査官 後藤 彰

(56)参考文献 特開2001-249806(JP,A)

国際公開第99/22293(WO,A1)

特表2001-521241(JP,A)

特開平7-200399(JP,A)

特開平5-120013(JP,A)

特開平3-63726(JP,A)

特開平1-239638(JP,A)

Chris H. Perleberg 外1名, "Branch Target Buffer Design and Optimization", IEEE Trans
action on Computers, 米国, 1993年 4月, 第42巻, 第4号, p. 396-412

(58)調査した分野(Int.Cl., DB名)

G06F 9/38

G06F 12/08