

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2012-502367
(P2012-502367A)

(43) 公表日 平成24年1月26日(2012.1.26)

(51) Int.Cl.	F I	テーマコード (参考)
G06F 9/38 (2006.01)	G06F 9/38 330B	5B005
G06F 12/08 (2006.01)	G06F 12/08 511B	5B013
	G06F 12/08 511C	
	G06F 12/08 505Z	
	G06F 12/08 507Z	

審査請求 未請求 予備審査請求 未請求 (全 23 頁)

(21) 出願番号 特願2011-526253 (P2011-526253)
 (86) (22) 出願日 平成21年9月4日(2009.9.4)
 (85) 翻訳文提出日 平成23年5月9日(2011.5.9)
 (86) 国際出願番号 PCT/US2009/056136
 (87) 国際公開番号 W02010/028316
 (87) 国際公開日 平成22年3月11日(2010.3.11)
 (31) 優先権主張番号 12/205,429
 (32) 優先日 平成20年9月5日(2008.9.5)
 (33) 優先権主張国 米国 (US)

(71) 出願人 591016172
 アドバンスド・マイクロ・デバイス・
 インコーポレイテッド
 ADVANCED MICRO DEVI
 CES INCORPORATED
 アメリカ合衆国、94088-3453
 カリフォルニア州、サニペイル、ピィ・
 オウ・ボックス・3453、ワン・エイ・
 エム・ディ・プレイス、メイル・ストップ
 ・68 (番地なし)
 (74) 代理人 100108833
 弁理士 早川 裕司
 (74) 代理人 100111615
 弁理士 佐野 良太

最終頁に続く

(54) 【発明の名称】 疎及び密予測を伴うハイブリッド分岐予測デバイス

(57) 【要約】

【解決手段】

マイクロプロセッサにおける分岐予測のためのシステム及び方法。ハイブリッドデバイスは、命令キャッシュの各エントリ内の通常より小さい数より多くない分岐に対して疎キャッシュ内に分岐予測情報をストアする。iキャッシュラインが追加的な分岐を備えているあまり一般的でない場合に対しては、デバイスは、対応する分岐予測情報を密キャッシュ内にストアする。疎キャッシュの各エントリは、対応する命令キャッシュラインが追加的な分岐命令を含んでいるか否かを表示するビットベクタをストアする。この表示はまた、記憶のための密キャッシュ内のエントリを選択するために用いられてよい。第2の疎キャッシュは、第1の疎キャッシュから退去させられた全てのエントリをストアする。

【選択図】 図4

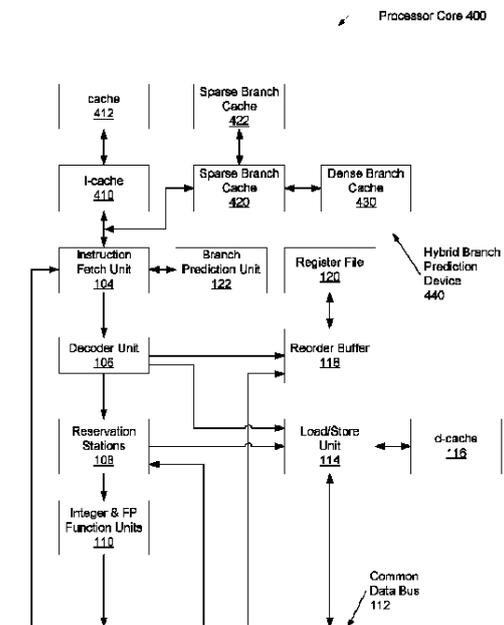


FIG. 4

【特許請求の範囲】**【請求項 1】**

命令キャッシュと、前記命令キャッシュのエントリに各々が対応する第 1 の複数のエントリを備えている第 1 の疎キャッシュとを備えたマイクロプロセッサであって、

前記第 1 の疎キャッシュの各エントリは、

第 1 の数より多くない分岐命令に対して分岐予測情報をストアし、

前記命令キャッシュの対応するエントリが前記第 1 の数より多い分岐命令を含むか否かの表示をストアするように構成されているマイクロプロセッサ。

【請求項 2】

第 2 の複数のエントリを備えている密キャッシュを更に備え、

前記第 2 の複数のエントリの各々は、前記第 1 の数の分岐命令内に含まれない追加的な分岐命令に対して分岐予測情報をストアするように構成されている、請求項 1 に記載のマイクロプロセッサ。

【請求項 3】

前記第 1 の疎キャッシュの全ての退去させられたエントリをストアするように構成されている第 2 の疎キャッシュを更に備えた、請求項 2 に記載のマイクロプロセッサ。

【請求項 4】

前記命令キャッシュ内のエントリ及び前記第 1 の疎キャッシュ内の対応するエントリは同一のアドレスによって選択される、請求項 3 に記載のマイクロプロセッサ。

【請求項 5】

前記疎キャッシュのエントリ内の前記ストアされた表示は、前記密キャッシュ内のエントリを選択するために利用されるビットベクタを備えており、各ビットは前記命令キャッシュのエントリ内の領域に対応している、請求項 4 に記載のマイクロプロセッサ。

【請求項 6】

前記分岐予測情報は前記命令キャッシュのエントリ内の対応する分岐命令のロケーションを表示するエンドポイントを備えている、請求項 5 に記載のマイクロプロセッサ。

【請求項 7】

前記分岐予測情報は分岐命令の種類及び分岐予測ユニットによって利用されるバイアス重み値を表示している、請求項 6 に記載のマイクロプロセッサ。

【請求項 8】

前記分岐予測情報は分岐予測ユニットによって利用される分岐目標アドレスを備えている、請求項 7 に記載のマイクロプロセッサ。

【請求項 9】

分岐予測を改善するための方法であって、

命令キャッシュのエントリ内の第 1 の数より多くない分岐命令に対して疎キャッシュのエントリ内に分岐予測情報をストアすることと、

前記命令キャッシュの対応するエントリが前記第 1 の数より多い分岐命令を含むか否かの表示を前記疎キャッシュの前記エントリ内にストアすることとを備えた方法。

【請求項 10】

前記第 1 の数の分岐命令内に含まれない追加的な分岐命令に対する分岐予測情報を密キャッシュ内にストアすることを更に備えた、請求項 9 に記載の方法。

【請求項 11】

前記第 1 の疎キャッシュの退去させられたエントリを第 2 の疎キャッシュ内にストアすることを更に備えた、請求項 10 に記載の方法。

【請求項 12】

前記命令キャッシュ内のエントリ及び前記第 1 の疎キャッシュ内の対応するエントリを同一のアドレスによって選択することを更に備えた、請求項 11 に記載の方法。

【請求項 13】

前記疎キャッシュのエントリ内の前記ストアされた表示は、前記密キャッシュ内のエントリを選択するために利用されるビットベクタを備えており、各ビットは前記命令キャッ

10

20

30

40

50

シュのエンタリ内の領域に対応している、請求項 1 2 に記載の方法。

【請求項 1 4】

前記分岐予測情報内に含まれる対応するエンドポイントによって前記命令キャッシュのエンタリ内の分岐命令のロケーションを表示することを更に備えた、請求項 1 3 に記載の方法。

【請求項 1 5】

分岐命令の種類及び分岐予測ユニットによって利用されるバイアス重み値を前記分岐予測情報内に表示することを更に備えた、請求項 1 4 に記載の方法。

【請求項 1 6】

前記分岐予測情報は分岐予測ユニットによって利用される分岐目標アドレスを備えている、請求項 1 5 に記載の方法。

10

【請求項 1 7】

命令キャッシュのエンタリに各々が対応する第 1 の複数のエンタリを備えている第 1 の疎キャッシュを備えたハイブリッド分岐予測デバイスであって、

前記第 1 の疎キャッシュの各エンタリは、

第 1 の数より多くない分岐命令に対して分岐予測情報をストアし、

前記命令キャッシュの対応するエンタリが前記第 1 の数より多い分岐命令を含むか否かの表示をストアするように構成されているハイブリッド分岐予測デバイス。

【請求項 1 8】

第 2 の複数のエンタリを備えている密キャッシュを更に備え、

20

前記エンタリの各々は、前記第 1 の数の分岐命令内に含まれない追加的な分岐命令に対して分岐予測情報をストアするように構成されている、請求項 1 7 に記載のハイブリッド分岐予測デバイス。

【請求項 1 9】

前記第 1 の疎キャッシュの全ての退去させられたエンタリをストアするように構成されている第 2 の疎キャッシュを更に備えた、請求項 1 8 に記載のハイブリッド分岐予測デバイス。

【請求項 2 0】

前記疎キャッシュのエンタリ内の前記ストアされた表示は、前記密キャッシュ内のエンタリを選択するために利用されるビットベクタを備えており、各ビットは前記命令キャッシュのエンタリ内の領域に対応している、請求項 1 9 に記載のハイブリッド分岐予測デバイス。

30

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

この発明はマイクロプロセッサに関し、更に詳しくは分岐予測メカニズムに関する。

【背景技術】

【0 0 0 2】

最新のマイクロプロセッサは 1 つ以上のプロセッサコア又はプロセッサを含むであろうし、各プロセッサはソフトウェアアプリケーションの命令を実行することができる。これらのプロセッサは典型的にはパイプライン化され、この場合、これらのプロセッサは 1 つ以上のデータ処理段階を含み、データ処理段階はこれらの間に位置する記憶要素（レジスタ及びアレイ）と共に直列に接続されている。1 つの段階の出力は、クロック周期又は位相を規定するクロックの遷移の間に次の段階の入力とされ、当該遷移は 1 つのクロックのほんの一瞬である。パイプラインは命令処理の部分部分が実行される任意の数の段階に分割されていてよいが、命令処理は一般的に、命令をフェッチすること(fetching)と、命令を復号化することと、命令を実行することと、実行された結果を命令によって指定される宛先にストアすることとを備えている。

40

【0 0 0 3】

理想的には、クロック周期毎にパイプラインの各段階に対する命令の有用な実行が生み

50

出される。しかし、パイプライン内のストール(stall)が当該パイプライン段階の間に行われるべき有用なワークを生じさせないことがある。幾つかのストールは数クロック周期を持続させることがあり、そしてプロセッサ性能を著しく低下させ得る。ストールの幾つかの例は、データキャッシュ又は命令キャッシュのミス、命令間のデータ依存性、及び予測を誤った分岐命令のような制御フロー予測ミスを含む。

【 0 0 0 4 】

プロセッサ性能に対するストールのネガティブな影響は、パイプライン段階を重複させることによって低減され得る。更なる技術は命令の順序外実行(out-of-order execution)を可能にすることであり、それによりデータ依存ストールの低減が支援される。また、スーパー标ケラアーキテクチャ(superscalar architecture)を伴うコアは、動的なスケジューリングに基きクロック周期毎に変化する数の命令を発行する。しかし、数クロック周期のストールは、それら全てのストール周期を隠すことを妨げるであろう順序どおりのリタイアメント(in-order retirement)に起因して、それでもなおプロセッサの性能を低下させる。そこで、性能損失を低減するための他の方法は、多重周期ストールの発生を低減することである。1つのそのような多重周期ストールは、分岐命令のような制御フロー命令の予測ミスである。

【 0 0 0 5 】

分岐命令は、条件付きの又は無条件の、及び直接の又は間接の、のような多くの種類を備えている。条件付き分岐命令は、命令ストリームをどのバスが取り入れるかの決定を実行する。命令内で符号化されているであろう指定の条件が満たされていないと分岐命令が決定した場合、その分岐命令は取り込まれるべきではないとみなされ、そしてプログラム順序における次のシーケンシャル命令が実行される。一方で、指定の条件が満たされていると分岐命令が決定した場合には、その分岐命令は取り込まれるべきであるとみなされる。その結果、プログラム順序内の次のシーケンシャル命令ではなく、むしろ分岐目標アドレスに配置される命令である後続の命令が実行される。無条件分岐命令は常に取り込まれる条件付き分岐命令とみなされる。試験する命令内には指定の条件はなく、そして後続の命令の実行は常にシーケンシャル順序とは異なるシーケンスで発生する。

【 0 0 0 6 】

また、分岐目標アドレスは、当該分岐命令それ自身内にストアされているであろうオフセットによって、プログラムカウンタ(PC)レジスタ内にストアされる線形アドレス値に対して指定され得る。この種の分岐目標アドレスは直接的であると称される。分岐目標アドレスはまた、レジスタ又はメモリ内の値によって指定することができ、この場合、レジスタ又はメモリロケーションが分岐命令内にストアされているであろう。この種の分岐目標アドレスは間接的であると称される。更に、間接的分岐命令においては、分岐目標アドレスを指定しているレジスタは異なる値でロードされ得る。

【 0 0 0 7 】

無条件間接的分岐命令の例は、プログラムコード内にサブルーチンを実装するために使用されることができ且つ分岐目標アドレスを供給するためにリターンアドレススタック(RAS)を用いることができる手続きコール及びリターンを含む。他の例は、スイッチケース命令文(switch-case statement)を実装するために用いられ得る間接的ジャンプ命令であり、スイッチケース命令文はC++及びJavaのようなオブジェクト指向プログラムにおいて一般的である。

【 0 0 0 8 】

条件付き分岐命令の例は、プログラムコード内にループを実装するために用いられ得る分岐命令である(例えば「フォー(for)」及び「ホワイル(while)」構成)。条件付き分岐命令は、取り込まれるとみなされるべき指定の条件を満たす必要がある。満たされた条件の例は、指定のレジスタがその時点でゼロのストア値を保持していることであり得る。指定のレジスタは当該条件付き分岐命令において符号化される。この指定のレジスタは、ソフトウェアアプリケーションコード内の命令によってループ内でデクリメントされるそのストア値を有することができる。指定のレジスタの出力は、専用のゼロ検出組み合わせ論

10

20

30

40

50

理に入力されるであろう。

【0009】

また、複数の条件付き分岐命令は互いにいくらかの依存関係を有していることがある。例えば、プログラムは次のような単純なケースを有しているかもしれない。

```
if ( value == 0 ) value = 1 ;  
if ( value == 1 )
```

【0010】

上記ケースを実装するために用いられるであろう条件付き分岐命令は、条件を予測する正確性を改善するために用いられ得るグローバル履歴を有しているであろう。1つの形態では、予測は2ビットカウンタによって実装され得る。次に分岐予測を更に詳細に説明する。

10

【0011】

最新のマイクロプロセッサは、分岐命令の条件の結末を決定し且つ分岐目標アドレスを決定するために多重化クロック周期を必要とすることがある。特定のパイプライン内で実行中の特定のスレッドに対して、分岐命令が復号化されるまで分岐命令又は後続の命令によっては有用なワークが実行されないことがあり、そして後で条件の結末及び分岐目標アドレスの両方が知られることになる。これらのストール周期はプロセッサの性能を低下させる。

【0012】

ストールよりもむしろ、予測は命令がフェッチされた直後に条件付き分岐条件及び分岐目標アドレスから作成され得る。いつ予測が整ったかに関する正確な段階は、パイプライン実装に依存している。分岐条件を予測するために、メモリからの、例えば命令キャッシュ(iキャッシュ)からの命令をフェッチするために用いられるPCは、分岐予測論理を索引付けるために用いられ得る。PCを用いる早期の組み合わせによる予測スキームの1つの例は、スコット・マクファーリン(Scott McFarling)の1993年の文献、「組み合わせ分岐予測器」、デジタル・ウエスタン・リサーチ・ラボラトリ・テクニカル・ノートTN-36("Combining Branch Predictors", Digital Western Research Laboratory Technical Note TN-36)に記載されているジーセレクト(gselect)分岐予測方法であり、当該文献はその全部を参照のためにここに組み込まれる。PCにストアされる線形アドレスは、ハッシング(hashing)機能におけるグローバル履歴レジスタ内にストアされる値と組み合わせられるであろう。ハッシング機能の出力及びPCは、パターン履歴テーブル(PHT)、分岐目標バッファ(BTB)その他の予測テーブルを索引付けるために用いられ得る。取り込まれた又は取り込まれなかった予測よりはむしろ、その時点での分岐命令の分岐目標アドレス情報でのグローバル履歴レジスタの更新は、条件付き分岐方向予測(即ち、取り込まれ又は取り込まれなかった結末予測)及び、BTB予測又は間接的目標アレイ予測のような間接的分岐目標アドレス予測の両方の予測精度を高めることができる。多くの異なるスキームが分岐予測メカニズムの種々の形態に含まれ得る。

20

30

【0013】

高い分岐予測精度は、より電力効率が良いで且つより高性能なマイクロプロセッサに貢献する。予測された命令ストリームからの命令は、分岐命令の実行に先立ち投機的に(speculatively)実行されてよく、そしてどんな場合でも分岐命令の実行に先立ちプロセッサのパイプライン内に置かれる。予測された命令ストリームが正しい場合には、クロック周期毎に実行される命令の数は有利に増加させられる。しかし、予測された命令ストリームが正確でない場合(即ち1つ以上の分岐命令が不正確に予測されている場合)には、不正確に予測された命令ストリームからの命令はパイプラインから破棄され、そしてクロック周期毎に実行される命令の数は減少させられる。

40

【0014】

多くの場合、分岐予測メカニズムは、特定の分岐命令に対するより正確な挙動をなすために、分岐命令の先行する実行の履歴を備えている。そのような分岐予測履歴は典型的には、分岐命令に対応するデータを記憶装置内に維持している必要がある。また、目標アド

50

レス予測において用いられる全部の又は一部の分岐目標アドレスをストアするために、分岐目標バッファ (B T B) が用いられることがある。履歴及びアドレス情報を備えている分岐予測データが記憶装置から退去させられ、あるいは失われた場合には、分岐命令のためのデータを後で再生成する必要がある。

【 0 0 1 5 】

上述の問題の 1 つの解決法は、分岐予測記憶装置の規模を大きくすることであろう。しかし、分岐予測記憶装置の規模を大きくすることは、ゲート領域の顕著な増大及び分岐予測メカニズムの規模の顕著な拡大を必要とするであろう。その結果、ゲート領域及び電力消費を減少させるために分岐予測記憶装置の規模を小さくすることによって、分岐の挙動に関する有益なデータは退去させられるかもしれず、そして再生成されなければならない。

10

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 1 6 】

上述に鑑み、ゲート数の顕著な増大又は分岐予測メカニズムの規模の顕著な拡大を必要とせずに分岐予測能力を向上させるための効果的な方法及びメカニズムが要望されている。

【 課題を解決するための手段 】

【 0 0 1 7 】

マイクロプロセッサにおける分岐予測のためのシステム及び方法が検討される。

20

【 0 0 1 8 】

1 つの実施形態では、マイクロプロセッサ内のハイブリッド分岐予測デバイスが提供される。命令キャッシュ (i キャッシュ) の各エントリ内の予め定められた数の分岐に対する分岐予測情報に対して記憶装置を割り当てるよりもむしろ、ハイブリッドデバイスは、命令キャッシュの各エントリ内の通常よりも小さい数より多くない分岐に対して第 1 の記憶装置内に分岐予測情報をストアする。 i キャッシュラインが追加的な分岐を備えているあまり一般的でない場合に対しては、デバイスは、対応する分岐予測情報を第 2 の記憶装置内にストアする。

【 0 0 1 9 】

また、疎キャッシュ及び密キャッシュを備えたハイブリッド分岐予測デバイスが検討される。疎キャッシュの各エントリは、命令キャッシュのエントリ又はキャッシュラインに対応する。これらの対応するエントリは、命令フェッチ要求の間にプロセッサから送られる同一のアドレスによって選択されてよい。疎キャッシュのエントリの各々は、命令キャッシュライン内の予め定められた数より多くない分岐命令に対して分岐予測情報をストアする。また、各エントリは、この命令キャッシュラインが追加的な分岐命令を含んでいてよいか否かの表示、例えばビットベクタをストアする。この表示はまた、追加的な分岐に対する分岐予測情報をストアするための密キャッシュ内のエントリを選択するために用いられてよい。ハイブリッド分岐予測デバイスはまた、第 1 の疎キャッシュから退去させられたエントリをストアするための第 2 の疎キャッシュを備えている。全てのエントリはストアされ、分岐予測情報は失われずあるいは後で再構築される必要がない。

30

40

【 図面の簡単な説明 】

【 0 0 2 0 】

【 図 1 】 図 1 はプロセッサコアの 1 つの実施形態の一般化されたブロック図である。

【 0 0 2 1 】

【 図 2 】 図 2 は分岐予測ユニットの 1 つの実施形態を示す一般化されたブロック図である。

【 0 0 2 2 】

【 図 3 】 図 3 は i キャッシュ記憶配置の 1 つの実施形態を示す一般化されたブロック図である。

【 0 0 2 3 】

50

【図4】図4はハイブリッド分岐予測を伴うプロセッサコアの1つの実施形態を示す一般化されたブロック図である。

【0024】

【図5】図5は分岐予測ユニットの1つの実施形態を示す一般化されたブロック図である。

【0025】

【図6】図6は疎キャッシュ記憶配置の1つの実施形態を示す一般化されたブロック図である。

【0026】

【図7】図7は密分岐キャッシュ記憶配置の1つの実施形態を示す一般化されたブロック図である。

【0027】

【図8】図8は分岐予測の効果的な改良のための方法の1つの実施形態のフロー図である。

【0028】

本発明が種々の修正及び代替的な形態を許容し得る一方で、具体的な実施形態が例示の目的で図面に示されており、またここに詳細に説明されている。しかし、図面及びそれに対する詳細な説明は、開示された特定の形態に本発明を限定することを意図しているのではなく、むしろ本発明は、添付の特許請求の範囲により画定されるような本発明の精神及び範囲内にある全ての修正、均等なもの及び代替案を網羅するものであることが理解されるべきである。

【発明を実施するための形態】

【0029】

以下の説明においては、本発明の完全な理解をもたらすために多くの特定の詳細が示されている。しかし、当業者であれば、これらの特定の詳細なしに本発明を実施し得ることを認識すべきである。幾つかの例では、周知の回路、構造及び技術は、本発明を不明瞭にすることを避けるために、詳細には示されていない。

【0030】

図1を参照すると、順序外実行を行うプロセッサコア100の一般化されたブロック図の1つの実施形態が示されている。コア100は所定の命令セットアーキテクチャ(ISA)に従う命令を実行するための回路を含む。例えば、x86命令セットアーキテクチャが選択されてよい。代替的には、任意の他の命令セットアーキテクチャが選択されてよい。1つの実施形態では、コア100は単一プロセッサ構成内に含まれていてよい。もう1つの実施形態では、コア100は多重化プロセッサ構成内に含まれていてよい。他の実施形態では、コア100は多重化ノードシステムの処理ノード内の多重化コア構成内に含まれていてよい。

【0031】

命令キャッシュ(iキャッシュ)102はソフトウェアアプリケーションに対する命令をストアすることができ、またデータキャッシュ(dキャッシュ)116は命令によって実行される計算で用いられるデータをストアすることができる。一般的に言うと、1つのキャッシュは1つ以上のブロックをストアすることができ、各ブロックは、図示しないシステムメモリ内の対応するアドレスにストアされるデータのコピーである。ここで用いられる「ブロック」は、連続したメモリロケーションにストアされる一連のバイトであり、コヒーレンシ(coherency)目的のための単位として取り扱われる。幾つかの実施形態では、ブロックはキャッシュ内での割り当て及び割り当て解除(allocation and deallocation)の単位でもある。1つのブロック内のバイトの数は、設計上の選択に従って変化してよく、また任意のサイズのものであってよい。例としては、32バイト及び64バイトのブロックがよく用いられる。

【0032】

キャッシュ102及び116は、図示されるように、プロセッサコア100内に集積化

10

20

30

40

50

されていてよい。代替的には、キャッシュ102及び116は、所望に応じてバックサイドキャッシュ構成又はインライン構成でコア100と結合されていてよい。更には、キャッシュ102及び116はキャッシュの階層(hierarchy)として実装されていてよい。1つの実施形態では、キャッシュ102及び116は各々L1及びL2キャッシュ構造を代表する。他の実施形態では、キャッシュ102及び116は、L3キャッシュ構造として実装されるもう一つのキャッシュ(図示せず)を共有してよい。代替的には、キャッシュ102及び116の各々はL1キャッシュ構造を代表し、そして共有されるキャッシュ構造はL2キャッシュ構造であってよい。所望に応じて他の組み合わせが可能でありまた選択されてよい。

【0033】

キャッシュ102及び116並びに任意の共有されるキャッシュの各々は、対応するキャッシュ制御器に結合されるキャッシュメモリを含んでいてよい。コア100が多重化コアシステム内に含まれている場合には、メモリ制御器(図示せず)は、パケットを経路付け、データ処理のためにパケットを受け取り、そしてコア100内の論理によって用いられる内部クロックにパケットを同期させるために用いられてよい。また、多重化コアシステムにおいては、多重化プロセッサの多重化キャッシュ内にメモリブロックの多重化コピーが存在してよい。従ってメモリ制御器内には、キャッシュコヒーレンシ回路が含まれていてよい。与えられたブロックは1つ以上のキャッシュ内にストアされていてよく、またキャッシュコピーの1つはメモリシステム内のコピーに対して修正されているであろうから、コンピューティングシステムは多くの場合にキャッシュとメモリシステムの間のコヒーレンシを維持する。コヒーレンシは、所定のコヒーレンシプロトコルに従ってブロックの他のキャッシュコピーによってブロックに対する更新が反映される場合に維持される。種々の具体的なコヒーレンシプロトコルが周知である。

【0034】

命令フェッチユニット(IFU)104は、iキャッシュミスが無い場合にクロック周期毎にiキャッシュ102から多重化命令をフェッチしてよい。IFU104は、iキャッシュ102からフェッチすべき次の命令のアドレスに対するポインタを保持するプログラムカウンタ(PC)レジスタを含んでいてよい。分岐予測ユニット122がIFU104に結合されていてよい。ユニット122は、命令ストリームのフローを次のシーケンシャル命令を実行することから変化させる命令の情報を予測するように構成されてよい。予測情報の例は、次のシーケンシャル命令が実行されるべきかあるいは命令ストリームにおける他のロケーション内の命令が次に実行されるべきかを決定する条件が満たされているか否かの予測を備えた1ビット値を含んでいてよい。予測情報の他の例は、次のシーケンシャル命令とは異なる実行すべき次の命令のアドレスであってよい。実際の結末及び予測が正しかったか否かの決定は、後でのパイプライン段階において生じてよい。また、代替的な実施形態においては、IFU104及びユニット122を2つの別のユニットとして実装するよりもむしろ、IFU104がユニット122を備えていてよい。

【0035】

復号器ユニット106は、多重化フェッチ命令のオペコード(opcodes)を復号化する。復号器ユニット106は、並べ替えバッファ118のような順序どおりのリタイアメントキュー(in-order retirement queue)におけるエントリを予約ステーション108内及びロード/ストアユニット114内に割り当ててよい。予約ステーション108におけるエントリの割り当ては発送(dispatch)とみなされる。予約ステーション108は、複数の命令がそれらのオペランドが利用可能になるまで待機する命令キューとして機能することができる。オペランドが利用可能であり且つハードウェア資源も利用可能である場合、命令は予約ステーション108から整数及び浮動小数点機能ユニット110又はロード/ストアユニット114へ順序外(out-of-order)に発行されてよい。機能ユニット110は、加算、減算、掛け算、割り算、及び平方根のようなコンピュータ計算のための演算論理ユニット(ALU)を含んでいてよい。分岐命令の結末を決定しまた計算された結末を予測された値と比較するための論理が含まれていてよい。一致しない場合には、予測ミスが発生

10

20

30

40

50

し、そして分岐命令の後の後続の命令は除去される必要があり、また新たなフェッチが正しいPC値で実行される必要がある。

【0036】

ロード/ストアユニット114はメモリアクセス命令を実行するためのキュー及び論理を含んでいてよい。また、正しい最も若いストア命令からの受信され転送されたロード命令データ又はバイパスデータを確保するために、照合論理がロード/ストアユニット114内にあってよい。

【0037】

機能ユニット110及びロード/ストアユニット114からの結果は、共通データバス112上に提示されてよい。結果は並べ替えバッファ118に送られてよい。ここで、それ自身の結果を受け取り、リタイアメントに対して標識付けられ、且つ先頭キュー(head-of-the-queue)である命令は、その結果をレジスタファイル120へ送ってよい。レジスタファイル120は、プロセッサコア100の汎用レジスタのアーキテクチャ状態を保持することができる。1つの実施形態では、レジスタファイル120は32個の32ビットレジスタを含んでいてよい。次いで、並べ替えバッファ内の命令は順序どおりにリタイアさせられてよく、そしてその先頭キューポインタは、プログラム順序における後続の命令に対して調節されてよい。

【0038】

共通データバス112上の結果は、結果を待っている命令のオペランドに値を転送するために、予約ステーションへ送られてよい。これらの待機している命令がそれらのオペランドに対する値を有しており且つハードウェア資源が当該命令を実行するために利用可能である場合には、これらの命令は、予約ステーション108から機能ユニット110又はロード/ストアユニット114内の適切なハードウェア資源に対して順序外で発行されてよい。共通データバス112上の結果は、制御フロー予測情報及び/又はPC値を更新するために、IFU104及びユニット122へ経路付けられてよい。

【0039】

図2は分岐予測ユニット200の1つの実施形態を示している。命令のアドレスはレジスタプログラムカウンタ210(PC210)内にストアされる。1つの実施形態では、アドレスは32ビット又は64ビット値であってよい。グローバル履歴シフトレジスタ240(GSR240)は、条件付き分岐命令の末尾の予測結果の最新の履歴を含んでいてよい。1つの実施形態では、GSR240は1エン트리10ビットシフトレジスタであってよい。他の実施形態では、シフトレジスタGSR240は異なる数のビットを含んでいてよい。GSR240にストアされている情報は、グローバル履歴を用いることによって、その時点での条件付き分岐命令の条件が満たされているか否かを予測するために用いられてよい。例えば1つの実施形態では、GSR240は、プログラム実行における最後の10個の条件付き分岐命令の1ビットの取り込まれた/取り込まれなかった結果を保持する10ビットシフトレジスタであってよい。1つの実施形態では、論理「1」は取り込まれた結末を示してよく、また論理「0」は取り込まれなかった結末を示してよく、あるいは逆であってもよい。また代替的な実施形態においては、GSR240は、分岐毎基準(per-branch basis)に対応する又は分岐履歴のテーブル内で結合された分岐履歴に対応する情報を用いてよい。これらの実施形態においては、分岐予測を作成するために用いられるグローバル履歴情報を提供するために、1つ以上の分岐履歴テーブル(BHT)が用いられてよい。

【0040】

十分なアドレスビット(即ちPC210にストアされているその時点での分岐命令のPC)がその時点での分岐命令を指定するために用いられる場合には、これらのビットのハッシュは、GSR240にストアされているグローバル履歴と共に、いずれかの要素単独でよりも有益な予測情報を有することができる。1つの実施形態では、PCの低位の10ビットはGSRの10ビットでハッシュされてよい。代替的な実施形態では、PCの低位の10ビット以外のビット、そして場合によっては非連続のビットがGSRのビットでハ

10

20

30

40

50

ッシュされてよい。また、G S R 2 4 0 の多重化された部分部分は P C 2 1 0 で別々にハッシュされてよい。例えば、2 6 ビット G S R 2 4 0 は、第 1 のハッシング機能において P C 2 1 0 と組み合わせられるその低位の 1 3 ビットと、第 2 のハッシング機能において P C 2 1 0 と組み合わせられる高位の 1 3 ビットとを有してよい。2 つの別々の出力は次いで、予測論理及びテーブル 2 6 0 へ伝達されてよい。多くのそのような代替が可能でありまた検討される。

【 0 0 4 1 】

1 つの実施形態では、P C ビット及び G S R ビットのハッシングは、ビットの連結 (concatenation) を備えてよい。代替的な実施形態では、ハッシング機能の結果は P C ビットの G S R ビットとの排他的 O R を含んでよい。ハッシング論理 2 5 0 内で実行される他のハッシング機能は、予測論理 2 6 0 内に含まれる 1 つ以上のパターン履歴テーブル (P H T) 及び / 又は他のテーブルに対する索引を決定するために用いられてよい。1 つの実施形態においては、予測論理 2 6 0 内で B T B を索引付けるために、P C が単独で用いられてよい。ここで用いられているように、文字が続く参照番号によって参照される要素は、当該番号単独によって集合的に参照されることがある。例えば、予測テーブル 2 6 2 a ~ 2 6 2 d は集合的に予測テーブル 2 6 2 として参照されることがある。

10

【 0 0 4 2 】

1 つの実施形態では、1 つの予測テーブル 2 6 2 は条件付き分岐のための P H T であってよく、この場合、P H T の各エントリは 2 ビットカウンタを保持してよい。特定の 2 ビットカウンタは、条件付き分岐命令結果 (即ち取り込まれ又は取り込まれていない) の過去の挙動に基いてインクリメント及びデクリメントされてよい。予め定められたスレッシュホルド値に一旦到達すると、ストアされている予測は、取り込まれた及び取り込まれていないの 1 ビット予測値の間でフリップしてよい。2 ビットカウンタのシナリオでは、P H T の各エントリは以下の 4 つの状態、即ち、取り込まれないことを強く予測する、取り込まれないことを予測する、取り込まれることを強く予測する、及び取り込まれることを予測する、のうちの 1 つを保持することができ、4 つの状態において、各状態は 1 ビットの取り込まれる / 取り込まれないの予測値に対応している。

20

【 0 0 4 3 】

代替的には、1 つ以上の予測テーブル 2 6 2 は、既に論じられたようにハッシング論理 2 5 0 の出力によって索引付けられてよく、そしてこれらのテーブル 2 6 2 は、1 つの実施形態では学習された (trained) 4 ビットの重みをストアしてよい。1 つの実施形態では、2 つのテーブル 2 6 2 が同一のクロック周期内で索引付けられてよい。1 つの実施形態では、これら 2 つのテーブル 2 6 2 はハッシング論理 2 5 0 からの同一の出力によって索引付けられてよい。代替的な実施形態では、これら 2 つのテーブル 2 6 2 は、ハッシング論理 2 5 0 内の別個のハッシング機能のそれぞれの出力から索引付けられてよい。読み出される対応する 2 つの重みは、2 ビットによって符号拡張されて (sign-extended) おり、そして予測論理 2 6 0 内の論理によって互い加算されてよい。代替的には、これら 2 つの符号拡張された値はまた、合算されそしてバイアス重みに加えられてよい。バイアス重みもまた 2 ビットによって符号拡張されており、このバイアス重みは分岐予測記憶装置の読み出しであってよく、これについては後で更に詳細に説明する。条件付き分岐命令の予測された方向を表示するために、合計 6 ビットの最も重み付けの大きいバイト (most-significant-byte) (M S B) が用いられてよい。

30

40

【 0 0 4 4 】

1 つの実施形態では、これらのテーブル 2 6 2 はハッシング論理 2 5 0 からの出力によってアクセスされてよく、ハッシング論理 2 5 0 からの出力は、論理 2 5 0 の入力としての分岐予測記憶装置からの分岐数 (例えば 1, 2, 等) を含む。これらの分岐数は、同一クロック周期内で予測されつつある別個の分岐命令を区別するために用いられてよい。例えば、テーブル 2 6 2 の行アクセスは、ハッシュされていない G S R 2 4 0 の一部分、例えば 1 つの実施形態では G S R 2 4 0 の低位の 7 ビットを用いてよい。これらのテーブル 2 6 2 の出力は、1 つ以上のその選択線に対してハッシュ出力を用いるマルチプレクサに

50

よって減少され得る。論理 250 内のこのハッシュ機能への入力は、PC 210、GSR 240 の分離された部分、例えば 1 つの実施形態では 23 乃至 7 のビット範囲、及び対応する分岐数を含んでいてよい。

【0045】

予測（例えば、取り込まれる / 取り込まれない、又は分岐目標アドレス、あるいは両方）が一旦決定されたならば、その値は GSR 240 内に投機的にシフトされてよい。1 つの実施形態では、取り込まれる / 取り込まれないの値だけが GSR 240 内にシフトされる。他の実施形態では、分岐目標アドレスの一部が GSR 240 内に移される。GSR 240 をどのように更新するかは決定は、更新論理 230 内で実行される。後のパイプライン段階において予測ミスが決定された場合には、この 1 つ以上の値は正しい結末で修復されてよい。しかし、このプロセスはまた、その時点でパイプライン内で生じている (in flight) 分岐予測ミス及び正しい PC からの命令の再フェッチに起因して、フェッチされている命令を終了させることを取り入れている。これらのステップは、プロセッサ性能の妨げになるパイプラインストールを含む。

10

【0046】

1 つの実施形態では、予測論理及びテーブル 260 内の PHT 又は他の論理からの 1 ビットの取り込まれる / 取り込まれないの予測は、i キャッシュを索引付けそして同時に GSR 240 を更新するのに次の PC を用いることを決定するために用いられてよい。例えば 1 つの実施形態では、予測が取り込まれた場合、分岐目標アドレスが次の PC を決定するために用いられてよい。予測が取り込まれなかった場合には、次のシーケンシャル PC が次の PC を決定するために用いられてよい。

20

【0047】

1 つの実施形態では、更新論理 230 は GSR 240 を更新する方法を決定してよい。例えば、条件付き分岐がグローバル履歴更新を要求している場合には、更新論理 230 は、1 ビットの取り込まれる / 取り込まれないの予測ビットを GSR 240 の最新の位置にシフトさせることを決定してよい。代替的な実施形態では、分岐は GSR に対する値を提供しなくてよい。他の実施形態では、GSR 240 及び PC 210 のハッシュは予測論理 260 内の間接的予測アレイを索引付けるために用いられ得るので、分岐目標アドレスの一部は、例えば間接的分岐のために GSR 240 に入力されてよい。分岐目標アドレスにおける例えば 1 つの実施形態では 4 ビットの当該部分は、分岐目標アドレスの中間部分から選択されてよい。分岐目標アドレスの高位のビットは、メモリ内の同様のロケーションに置かれる命令間では頻繁には変化しないであろうし、また低位のビットは、メモリ内の異なる線、ワード、及びバイトへのアクセスに起因して高い頻度で変化する。例えば 1 つの実施形態では、分岐目標アドレスのビット 7 乃至 4 が用いられてよい。

30

【0048】

代替的な実施形態では、更新論理 230 は GSR 240 内へのシフト動作を実行しなくてよく、むしろ分岐目標アドレスの 1 つ以上のビットと GSR 240 の 1 つ以上の選択されたビットとの排他的 OR を実行してよい。次いで、GSR 240 の選択されたビットは排他的 OR の結果で置換される。例えば、分岐目標アドレスの 3 ビット、例えば分岐目標アドレスのビット 7 乃至 5 と GSR 240 の最新の 3 ビットとの排他的 OR の結果が GSR 240 の最新の 3 ビットと置き換わってよい。更新論理 230 の他の実施形態が実装されてよい。いずれの場合にも、間接的な分岐命令の分岐目標アドレスの一部は、これらの命令の発生をその時点での分岐予測メカニズムと関連付ける試みにおいて、グローバル履歴を更新するために用いられてよい。与えられた分岐又は分岐の種類が GSR に対して価値をもたらすか否かは、設計上の選択である。

40

【0049】

更新論理 230 の各実装においては、GSR 内にストアされた新たなグローバル履歴は、条件付き分岐方向予測（即ち、取り込まれる / 取り込まれないの結末予測）及び間接的分岐目標アドレス予測、例えば BTB 予測又は間接的目標アレイ予測の両方の正確性を高めることができる。正確性の向上は、ダイ面積、電力消費、及びクロック周期増大に対し

50

て無視し得る影響で達成することができる。

【0050】

ソフトウェアアプリケーション命令は、図1のiキャッシュ102のような命令キャッシュ内に種々の方法でストアされ得る。例えば、図3は4ウェイセットアソシアティブキャッシュ構成(4-way set associative cache organization)を用いて命令がストアされるiキャッシュ記憶配置300の1つの実施形態を示している。ISAに依存する可変長命令であってよい命令338は、4ウェイセットアソシアティブキャッシュ330内のキャッシュラインのデータ部分又はブロックデータであってよい。1つの実施形態では、キャッシュラインの命令338は64バイトを備えていてよい。代替的な実施形態では、異なるサイズが選ばれてよい。

10

【0051】

命令338の連続的なバイト内にストアされていてよい命令は、1つ以上の分岐命令を含んでいてよい。幾つかのキャッシュラインは少しの分岐命令を有していてよく、また他のキャッシュラインは多くの分岐命令を有していてよい。キャッシュライン毎の分岐命令の数には一貫性がない。従って、対応するキャッシュラインのための分岐予測情報の記憶装置は、全ての分岐に対する情報を提供するために、多数の分岐情報がキャッシュライン内にストアされることを前提とする必要がある。しかし、そのような記憶装置は大きなダイ面積を必要とするであろうし、また電力消費を増大させるであろう。キャッシュライン毎の小さな数であってよい共通ケースの数の分岐命令の情報及び非共通ケースに対するキャッシュライン内の追加的な分岐の表示のための主記憶装置を有するハイブリッド記憶装置は、より効率的であろう。ハイブリッド記憶装置スキーム内には、主記憶装置によって表示される追加的な分岐のための補助記憶装置があってよい。そのようなハイブリッド記憶装置スキームの更なる詳細は後述する。

20

【0052】

キャッシュ330の4ウェイの各々はまた状態情報334を有しており、状態情報334はキャッシュラインの有効ビット及び他の状態情報を備えていてよい。例えば、状態フィールドは、MOESIスキーム内の状態のような対応するキャッシュブロックの状態を特定するために用いられる符号化されたビットを含んでいてよい。また、ブロック状態334内のフィールドは、退去(eviction)のための、最後に使用されてから最も長い時間が経過した(Least Recently Used)(LRU)情報を表示するために用いられるビットを含んでいてよい。LRU情報は、キャッシュセット332内のどのエントリが最も長い間参照されなかったのかを表示するために用いることができ、またキャッシュ制御器によって採用されるキャッシュ交換アルゴリズムと併せて用いることができる。

30

【0053】

プロセッサコアからキャッシュ330に提示されるアドレス310は、対応するキャッシュセット332を選択するためにブロックインデックス318を含んでいてよい。1つの実施形態では、ブロック状態334及びブロックタグ336が、同一アレイ内の連続したビット内よりはむしろ別個のアレイ内にストアされていてよい。ブロックタグ336は、選択されたキャッシュセット332内で4キャッシュラインのどれがアクセスされている最中であるかを決定するために用いられてよい。また、アドレス310のオフセット320がキャッシュライン内の具体的なバイト又はワードを表示するために用いられてよい。

40

【0054】

次に図4を参照すると、ハイブリッド分岐予測を伴うプロセッサコア400の1つの実施形態の一般化されたブロック図が示されている。図1の回路部分に対応する回路部分には同一の番号が付されている。iキャッシュサブシステムに対するキャッシュ階層の最初の2つのレベルは、iキャッシュ410及びキャッシュ412として明示的に示されている。キャッシュ410及び412は、1つの実施形態では、それぞれL1キャッシュ構造及びL2キャッシュ構造として実装されてよい。1つの実施形態では、キャッシュ412は、命令及びデータの両方をストアするスプリット第2レベルキャッシュであってよい。

50

代替的な実施形態では、キャッシュ 4 1 2 は、2 つ以上のコアの間で共有されるキャッシュであってよく、またメモリ制御器内にキャッシュコヒーレンシ制御回路を必要とする。他の実施形態では、L 3 キャッシュ構造がチップ上又はチップ外にあってよく、そしてキャッシュ 4 1 2 よりもむしろ L 3 キャッシュが多重化コア間で共有されてよい。

【 0 0 5 5 】

前述したように、i キャッシュ 4 1 0 からフェッチされつつあるアドレスの有益な割合のために、対応する i キャッシュライン内には少数の分岐命令のみが含まれていてよい。一般的に言って、多くのアプリケーションコードの大きな割合に対して、分岐は i キャッシュライン内に疎にのみ見出される。従って、特定の i キャッシュラインに対応する分岐予測情報の記憶は、多数の分岐に対して情報をストアするための回路を割り当てる必要はないであろう。例えば、ハイブリッド分岐予測デバイス 4 4 0 は、分岐予測ユニット 1 2 2 によって用いられるべき分岐予測情報をストアするためのダイ面積及び回路をより効率的に割り当てることができる。疎分岐キャッシュ 4 2 0 は、i キャッシュライン毎に予め定められた通常の数分の分岐命令に対する分岐予測情報をストアすることができる。i キャッシュ 4 1 0 内の各キャッシュラインは、対応するエントリを疎分岐キャッシュ 4 2 0 内に有してよい。1 つの実施形態では、通常の数分の分岐は、i キャッシュ 4 1 0 内の各 6 4 バイトキャッシュラインに対して 2 分岐であってよい。i キャッシュ 4 1 0 内の各ラインのための疎な数の分岐に対してのみ予測情報をストアすることによって、i キャッシュ 4 1 0 内の各ラインに対して予め定められた最大数の分岐に対する情報を含む記憶装置において、キャッシュ 4 2 0 のサイズを大幅に低減することができる。ダイ面積要求、容量性負荷、及び電力消費が各々低減され得る。

10

20

【 0 0 5 6 】

1 つの実施形態では、i キャッシュ 4 1 0 は 4 ウエイセットアソシアティブキャッシュとして組織化されてよい。疎分岐キャッシュ 4 2 0 は直接マッピングされたキャッシュとして組織化されてよい。疎分岐キャッシュ 4 2 0 内の各エントリは i キャッシュ 4 1 0 内のキャッシュラインに対応してよい。疎分岐キャッシュ 4 2 0 の各エントリは、i キャッシュ 4 1 0 の対応するライン内の予め定められた疎な数の分岐命令、例えば 1 つの実施形態では 2 分岐、に対応する分岐予測情報を備えていてよい。分岐予測情報は後で更に詳細に説明される。代替的な実施形態では、異なる数の分岐命令が疎であると決定されてよく、また i キャッシュ 4 1 0 内のラインのサイズは異なるサイズであってよい。キャッシュ 4 2 0 は、IFU 1 0 4 から i キャッシュ 4 1 0 へ送られるのと同じ線形アドレスによって索引付けられてよい。i キャッシュ 4 1 0 及びキャッシュ 4 2 0 は両方とも、キャッシュライン境界に対応する線形アドレス内のビットのサブセットによって索引付けられてよい。例えば 1 つの実施形態では、線形アドレスはリトルエンディアン (little-endian) バイト順序での 3 2 ビットを備えていてよく、また i キャッシュ 4 1 0 内のラインは 6 4 バイトを備えていてよい。従って、キャッシュ 4 1 0 及び 4 2 0 は各々、ビット 6 で終了する線形アドレスの同一部分によって索引付けられてよい。

30

【 0 0 5 7 】

疎分岐キャッシュ 4 2 2 は、キャッシュ 4 2 0 から退去させられたラインをストアするためにコア 4 0 0 において利用されてよい。キャッシュ 4 2 2 はキャッシュ 4 2 0 と同じキャッシュ構成を有してよい。i キャッシュ 4 1 0 からラインが退去させられてキャッシュ 4 1 2 内に置かれた場合、その対応するキャッシュ 4 2 0 内のエントリは、キャッシュ 4 2 0 から退去させられてキャッシュ 4 2 2 内にストアされてよい。このように、先に退去させられたキャッシュラインがキャッシュ 4 1 2 からキャッシュ 4 1 0 に戻った場合には、このキャッシュライン内の分岐に対する対応する分岐予測情報もまた、キャッシュ 4 2 2 からキャッシュ 4 2 0 へと戻される。従って、対応する分岐予測情報が再構築される必要はない。分岐予測情報を再構築するためのプロセスの不在により、プロセッサ性能が向上し得る。

40

【 0 0 5 8 】

より密に詰め込まれた分岐情報を含むアプリケーションコード内の領域に対しては、i

50

キャッシュ 4 1 0 内のキャッシュラインは、疎な数の分岐より多くを含んでいてよい。疎分岐キャッシュ 4 2 0 の各エントリは、i キャッシュ 4 1 0 内の疎な数の分岐を超えた追加的な分岐の表示をストアしてよい。追加的な分岐が存在する場合には、対応する分岐予測情報は密分岐キャッシュ 4 3 0 内にストアされてよい。

【 0 0 5 9 】

密分岐キャッシュ 4 3 0 の各エントリは、i キャッシュ 4 1 0 内のラインの一部に対応してよい。例えば、i キャッシュ 4 1 0 は 6 4 バイトのラインを有してよく、また密分岐キャッシュ 4 3 0 の各エントリは 6 4 バイトラインの 8 バイト部分に対応してよい。疎分岐キャッシュ 4 2 0 の各エントリは、対応する 6 4 バイトラインのどの 8 バイト部分が追加的な分岐を有してよいかを指定するベクタのビット(vector of bits)を有してよく、この場合、その 6 4 バイトラインは通常のスparseな数の分岐より多くを有してよい。ベクタビットの数及びi キャッシュラインの提示された部分の対応するサイズは、アプリケーションコード内に密に詰め込まれた分岐の領域の挙動を効率的に捕捉するために選択されてよい。

10

【 0 0 6 0 】

1 つの実施形態では、密分岐キャッシュ 4 3 0 は 4 ウエイセットアソシアティブキャッシュとして組織化されてよい。従って密分岐キャッシュ 4 3 0 は、6 4 バイトラインの 8 バイト部分毎に 4 つの分岐命令に対する分岐予測情報をストアすることができる。6 4 バイトライン内の 8 つの 8 バイト部分でもって、密分岐キャッシュ 4 3 0 は、単一の 6 4 バイトライン内の 3 2 個の分岐命令に対する分岐予測情報をストアすることができる。この実施形態では、疎分岐キャッシュ 4 2 0 と密分岐キャッシュ 4 3 0 のハイブリッドな組み合わせが、i キャッシュ 4 1 0 内の 6 4 バイトラインに対する 3 4 個の分岐命令の分岐予測情報をストアすることができる(2 つの疎分岐と 3 2 個の密分岐)。キャッシュ 4 1 0、4 1 2、4 2 0、4 2 2、及び 4 3 0 のサイズ、結合性(associativity)、及び密度は、対象とするアーキテクチャ上で動作することになるアプリケーションに対する性能と電力消費の間での最良のトレードオフをもたらすように選択されてよい。

20

【 0 0 6 1 】

図 5 を参照すると、分岐予測ユニット 5 0 0 の一般化されたブロック図の 1 つの実施形態が示されている。図 2 の回路部分に対応する回路部分には同一の番号が付されている。ここでは、ストアされたハイブリッド分岐予測情報は、予測論理及びテーブル 2 6 0 とハッシング論理 2 5 0 へ伝達されてよい。1 つの実施形態では、ハイブリッド分岐予測情報は、疎分岐キャッシュ 4 2 0 及び 4 2 2 並びに密分岐キャッシュ 4 3 0 のような i キャッシュとは別のキャッシュ内にストアされてよい。1 つの実施形態では、この情報は、同じクロック周期内で予測されつつある分岐命令を区別するための分岐数、特定の条件付き分岐命令の種類その他を表示する分岐種類情報、テーブル 2 6 0 内のテーブルエントリと合算されるべき重み情報、対応するキャッシュライン内の分岐命令の最終バイトに対するポインタのような付加的なアドレス情報、及び対応する分岐目標アドレス情報を含んでいてよい。

30

【 0 0 6 2 】

既に論じられたように、ストアされたハイブリッド分岐情報からのバイアス重み情報が、論理及びテーブル 2 6 0 内のテーブルからの 2 つの対応する符号拡張重み値と合算されてよい。このバイアス重みもまた合算に先立ち符号拡張されていてよく、そして 6 ビット合計の最も重み付けの大きいバイト(MSB)は、条件付き分岐命令の予測された方向を表示するために用いられてよい。

40

【 0 0 6 3 】

図 6 は分岐予測情報がストアされる疎キャッシュ記憶配置 6 0 0 の 1 つの実施形態を示している。1 つの実施形態では、キャッシュ 6 3 0 は直接マッピングされたキャッシュとして組織化されてよい。予め定められた疎な数のエントリ 6 3 4 は、直接マッピングされたキャッシュ 6 3 0 内のキャッシュラインのデータ部分内にストアされてよい。1 つの実施形態では、疎な数は 2 であると決定されてよい。各エントリ 6 3 4 は、i キャッシュ 4

50

10の対応するライン内の特定の分岐に対する分岐予測情報をストアしてよい。対応するライン内に疎な数の分岐を超えて追加的な分岐が存在してよいことの表示が、密分岐表示636内にストアされる。

【0064】

1つの実施形態では、各エントリ634は、有効ビットを備えている状態フィールド640と他の状態情報とを備えていてよい。エンドポイントフィールド642は、iキャッシュ410のライン内の対応する分岐命令の最後のバイトに対する表示をストアしてよい。例えば、対応する64バイトのiキャッシュラインに対して、エンドポイントフィールド642は、64バイトのいずれかを指し示すために6ビットを備えていてよい。このポイント値は、iキャッシュ410及び疎分岐キャッシュ420の両方を索引付けるために用いられる線形アドレス値に添付されていてよく、また全体のアドレス値は分岐予測ユニット500へ送られてよい。

10

【0065】

予測情報フィールド644は、分岐予測ユニット500内で用いられるデータを備えていてよい。例えば、特定の分岐命令が直接的なものであり、間接的なものであり、条件付きなものであり、無条件なものであり、又は他のものであることを表示するために、分岐種類情報が伝達されてよい。また、予測論理及びテーブル260における重みテーブル内の重みと合算されてよいバイアス重みがストアされてよい。分岐予測ユニット500に関して既に論じられたように、1つの実施形態では、4ビットの重みは、予測テーブル260内の2つの別々のテーブルからの読み出しであってよい。これらのテーブルは、PC210及びGSR240を結合するハッシング機能の出力によって索引付けられてよい。読み出される対応する2つの重みは、2ビットによって符号拡張され、そして同じく2ビットにより符号拡張されているフィールド644からのバイアス重みに加えられてよい。6ビット合計の最も重み付けの大きいバイト(MSB)は、条件付き分岐命令の予測された方向を示すために用いられてよい。

20

【0066】

対応する部分的分岐目標アドレス値がアドレスフィールド646内にストアされてよい。通常の場合、分岐命令それ自身と同じページ内に分岐目標が置かれていることが見出されるであろうから、部分的分岐目標アドレスのみが必要とされてよい。1つの実施形態では、1ページは4KBを備えていてよく、そして12ビットの分岐目標アドレスがフィールド646内に記憶される必要があるだけである。より小さいフィールド646は、ダイ面積、容量負荷、及び電力消費を低減することを更に支援する。フィールド646にストアされているのに対して付加的なビットを必要とする分岐目標のためには、別個のページ外アレイが利用されてよい。

30

【0067】

密分岐表示フィールド636はビットベクタを備えていてよく、この場合、当該ベクタの各ビットは、iキャッシュ410の対応するライン内の一部分に対して追加的な分岐が存在する可能性を表示する。例えば、フィールド636は8ビットベクタを備えていてよい。各ビットはiキャッシュ410の64バイトライン内の別個の8バイト部分に対応してよい。1つの実施形態では、最初のビットはiキャッシュ410の対応するライン内の最初の8バイトのデータに対応してよい。ベクタ内の各次に続くビットは、キャッシュライン内の次の連続する8バイトに対応してよい。1つの実施形態では、フィールド636内の1つ以上のセットビットは、疎な数の分岐を超えて追加的な分岐が64バイトキャッシュライン内に存在し得ることを表示してよい。フィールド636内の特定のセットビットは、8バイトの対応するグループ内に追加的な分岐が存在することを示してよい。この場合、密分岐キャッシュ430が索引付けられる。1つの実施形態では、密分岐キャッシュ430は疎分岐キャッシュ420と並行して索引付けられてよいが、電力を節約するために、フィールド636内でセットビットが見出される場合にのみキャッシュ430が索引付けられてもよい。

40

【0068】

50

図7は密分岐キャッシュ記憶配置700の1つの実施形態を示している。1つの実施形態では、4ウェイセットアソシアティブキャッシュ構成を用いて追加的な分岐予測情報がストアされる。キャッシュ730のキャッシュセット732内の4つのキャッシュラインの各々は、iキャッシュ410の64バイトキャッシュライン内の8バイト領域に対応している。キャッシュ730の4ウェイの各々は状態情報734を有している。状態情報734は、有効ビットと、対応するキャッシュラインの他の状態情報とを備えている。

【0069】

プロセッサコアからキャッシュ730に提示されるアドレス310は、対応するキャッシュセット732を選択するために、ブロックインデックス318及び、オフセット320の一部を伝達している。密分岐キャッシュ430は、iキャッシュ410及び疎分岐キャッシュ420とは異なるアドレス可能な境界を利用してよい。例えば1つの実施形態では、キャッシュ410及び420は64バイト境界を伴うデータにアドレスすることができ、また密分岐キャッシュ430は8バイト境界を伴うデータにアドレスすることができる。従って、オフセット320の一部、例えば合計6ビットのうちの3ビットがキャッシュ730を索引付けるために追加的に必要になるであろう。1つの実施形態では、ブロック状態734及びブロックタグ736は、同一のアレイ内の連続するビット内よりはむしろ、別個のアレイ内にストアされている。ブロックタグ736は、選択されたキャッシュセット732内で4つのキャッシュラインのどれがアクセスされている最中であるかを決定するために用いられている。分岐予測情報はブロックデータ738の連続するビット内にストアされている。

【0070】

1つの実施形態では、各ブロックデータ738は、iキャッシュ410のライン内の対応する分岐命令の最後のバイトに対する表示をストアするエンドポインタフィールド740を備えている。例えば、対応する64バイトのiキャッシュラインに対して、エンドポインタフィールド740は、対応する8バイト領域内の任意のバイトを指し示すために3ビットを備えている。このポインタ値は、iキャッシュ410及び疎分岐キャッシュ420の両方を索引付けるために用いられる線形アドレス値に添付されている。また全体のアドレス値は分岐予測ユニット500へ送られている。

【0071】

1つの実施形態では、予測情報フィールド742は、疎キャッシュ420内の分岐予測フィールド644と同様のデータを備えている。この情報は分岐予測ユニット500内で用いられている。特定の分岐命令が直接的なものであり、間接的なものであり、条件付きなものであり、無条件なものであり、又は他のものであることを示すために、分岐種類情報が伝達されている。また、予測論理及びテーブル260における重みテーブル内の重みと合算されてよいバイアス重みがストアされている。同様に、分岐目標アドレスフィールド744は、疎キャッシュ420内のアドレスフィールド646と同様のデータを備えている。通常の場合、分岐命令それ自身と同じページ内に分岐目標が置かれていることが見出されるであろうから、部分的分岐目標アドレスのみが必要とされている。1つの実施形態では、1ページは4KBを備えている。そして全部で32ビットではなく12ビットの分岐目標アドレスがフィールド744内に記憶される必要があるだけである。より小さいフィールド744は、ダイ面積、容量負荷、及び電力消費を低減することを更に支援する。フィールド744にストアされているのに対して付加的なビットを必要とする分岐目標のためには、別個のページ外アレイが利用されている。

【0072】

図8は分岐予測の効果的な改良のための方法800を示している。方法800は、代替的な実施形態を導き出すために当業者によって修正されている。また、この実施形態におけるステップは逐次的な順序で示されている。しかし、幾つかのステップは図示されているのとは異なる順序で起こり、幾つかのステップは同時に実行されており、幾つかのステップは他のステップと組み合わせられており、そして幾つかのステップは他の実施形

10

20

30

40

50

態ではなくてよい。図示される実施形態では、プロセッサはブロック 802 において命令をフェッチする。プログラムカウンタ内にストアされる線形アドレスは、命令データの連続するバイトをフェッチするために i キャッシュ 410 へ伝達されてよい。i キャッシュ 410 内のキャッシュラインのサイズに応じて、プログラムカウンタの全部の内容が i キャッシュ 410 へ伝達されなくてもよい。またブロック 804 において、同じアドレスが疎分岐キャッシュ 420 へ伝達されてよい。

【0073】

L1 疎分岐キャッシュ 420 においてキャッシュヒットがない場合（条件ブロック 806）には、対応するキャッシュラインがブロック 808 において L2 疎分岐キャッシュから検索され (retrieved) てよい。ミスが発生した場合には、i キャッシュ 410 内の対応するキャッシュラインはどこにもないであろうから、第 2 レベルキャッシュ 412 から検索される必要がある。一旦 L1 疎分岐キャッシュ 420 内の要求されたラインがアクセスされると、条件ブロック 810 において、疎分岐エントリが有効エントリに対して照合される。1つの実施形態では、疎分岐エントリはコード実行の間に順次埋められてよい。従って、疎分岐キャッシュ 420 内の有効エントリ間には無効エントリの「ギャップ」はないであろう。無効エントリが検出されると、i キャッシュ 410 の対応するライン内にはもはや分岐命令がないことを表示してよい。無効エントリが発見されない場合には、i キャッシュ 410 内の対応するラインは、少なくとも最大数の疎分岐命令を有しているであろう。この対応する i キャッシュライン内の予め定められた数の疎分岐よりも多くの分岐命令がある場合には、疎分岐キャッシュ 420 内にストアされている密分岐表示 636 内のビットベクタが当該表示をもたらすことができる。

10

20

【0074】

有効な疎分岐エントリがない場合（条件ブロック 810）には、i キャッシュ 410 のライン内にいずれの分岐命令もないか、あるいはコード実行の間に分岐命令が未だ検出されていないことになる。この場合には、フロー制御はブロック 802 へ戻る。他の場合には、最初の疎分岐エントリが照合される。疎分岐エントリは順次埋められ且つ少なくとも 1つのエントリは有効であるから、最初の疎分岐エントリは有効である必要がある。疎分岐エントリは並行して照合されてよく、例えば各疎分岐エントリの状態フィールド 640 内の有効ビットがセットされているか否かが検出される。ここでフローを逐次的に説明する。その時点での疎分岐エントリは、その時点でのエントリが有益な分岐予測情報を有しているかどうか（条件ブロック 812）を決定するために照合されるその対応する有効ビットを有している。有益な情報がストアされていると有効ビットが表示している場合には、ブロック 814 において、疎分岐キャッシュ 420 は、エンドポイントフィールド 642、分岐予測情報フィールド 644、及び分岐目標アドレスフィールド 646 内にストアされている対応する情報をプロセッサに伝達してよい。次いで、次の疎分岐エントリがその有効ビットを照合される。

30

【0075】

予め定められた疎な数の分岐エントリの全てが未だ照合されていない場合（条件ブロック 816）には、方法 800 のための制御フローはブロック 812 へ戻る。他の場合には、密分岐表示フィールド 636 内の密ビットベクタがセットビットに対して照合されてよい（条件ブロック 818）。このベクタ内に 1つ以上のビットがセットされている場合には、ブロック 820 において、密分岐キャッシュ 430 は、i キャッシュ 410 及び疎分岐キャッシュ 420 にアクセスするために用いられたよりもっと下位のビットでプログラムカウンタ値の一部によって索引付けられてよい。このことは、密分岐表示フィールド 636 内の各ビットが例えば 64 バイトの i キャッシュラインより小さい 8 バイトのような領域に対応するであろうから、より細かい精度によってアドレス可能な密キャッシュに起因し得る。

40

【0076】

密分岐キャッシュ 430 がセットアソシアティブである場合には、アドレスのタグフィールド及び密分岐キャッシュのフィールド 734 内の有効ビットは、プロセッサへ送るべ

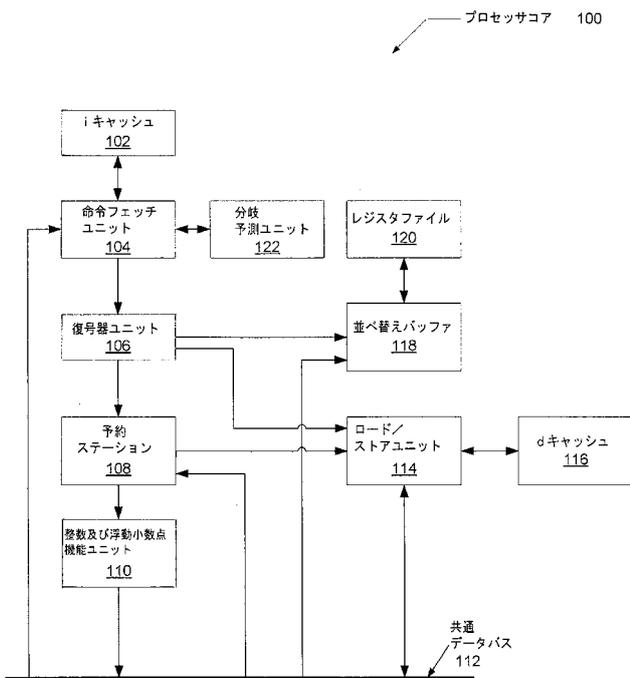
50

き対応するキャッシュラインを決定するために用いられてよい。このキャッシュラインは、エンドポイントフィールド740、分岐予測情報フィールド742、及び分岐目標アドレスフィールド744内にストアされている情報を備えていてよい。方法800の制御フローは次いでブロック802へ戻る。同一クロック周期内で並行して予測される分岐命令の数は、分岐予測ユニット500内の回路に依存するであろう。例えば1つの実施形態では、疎分岐キャッシュ420は、1クロック周期において2つの疎分岐命令に対する分岐予測情報を分岐予測ユニット500へ送ることが可能であろう。密分岐キャッシュ430からの分岐予測情報は、後続のクロック周期において送られてよい。クロック周期毎に分岐予測情報によって示される密分岐の数、及びクロック周期毎に予測される密分岐の数は、選択される実施形態に依存する。

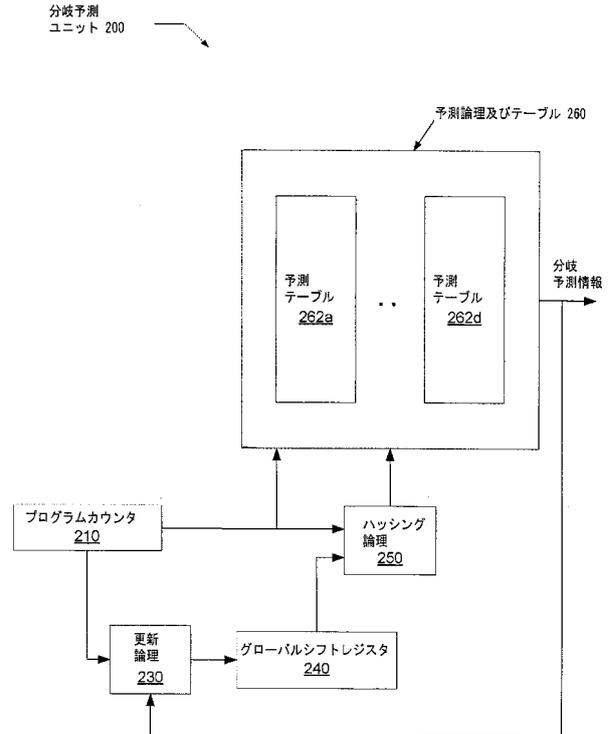
【0077】

上述の実施形態はかなり詳細に説明されてきたが、上述の開示を完全に理解するならば、多くの変形及び修正が当業者にとって明らかであろう。後述の特許請求の範囲は、そのような全ての変形及び修正を包含するものとして解釈されることが意図されている。

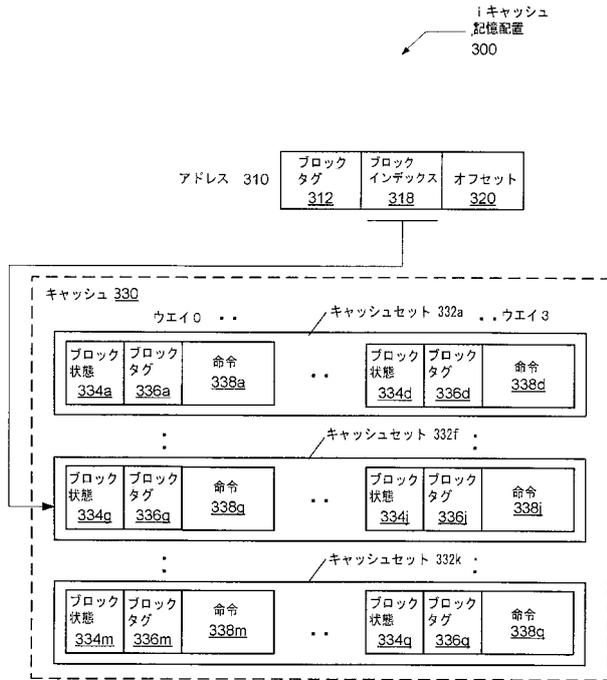
【図1】



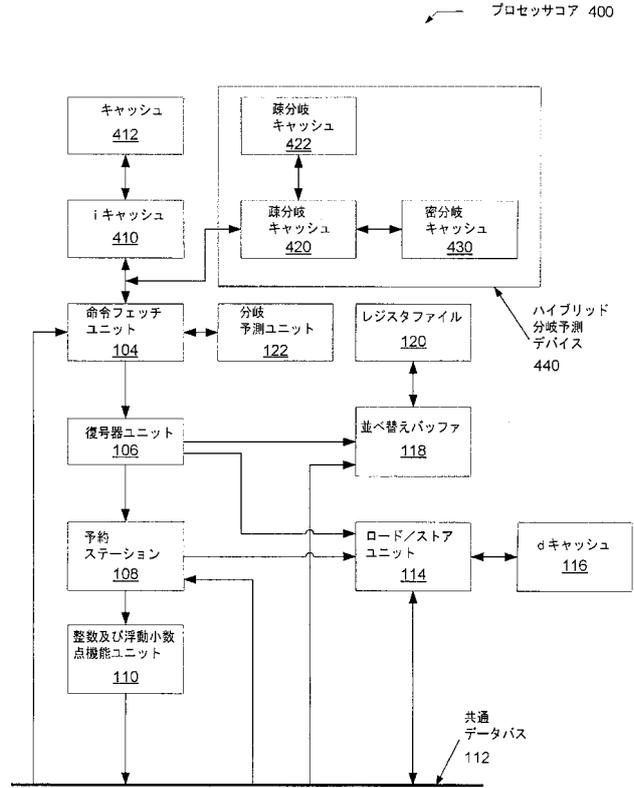
【図2】



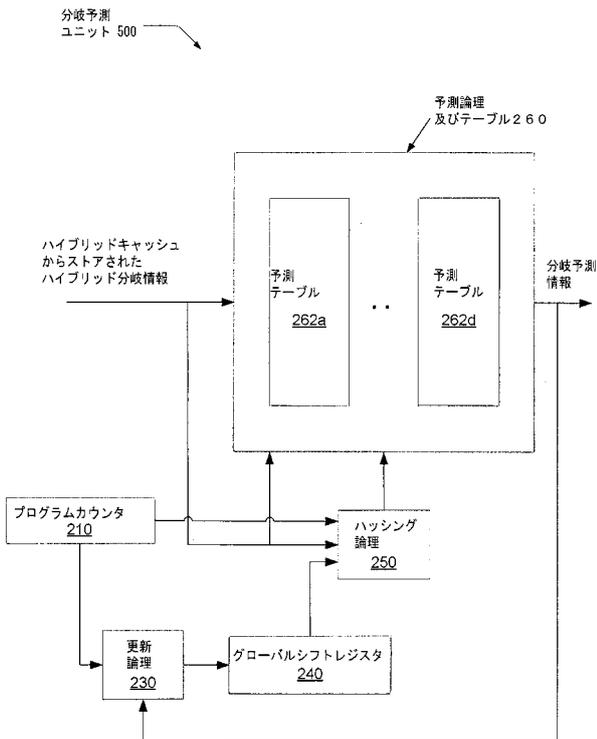
【図 3】



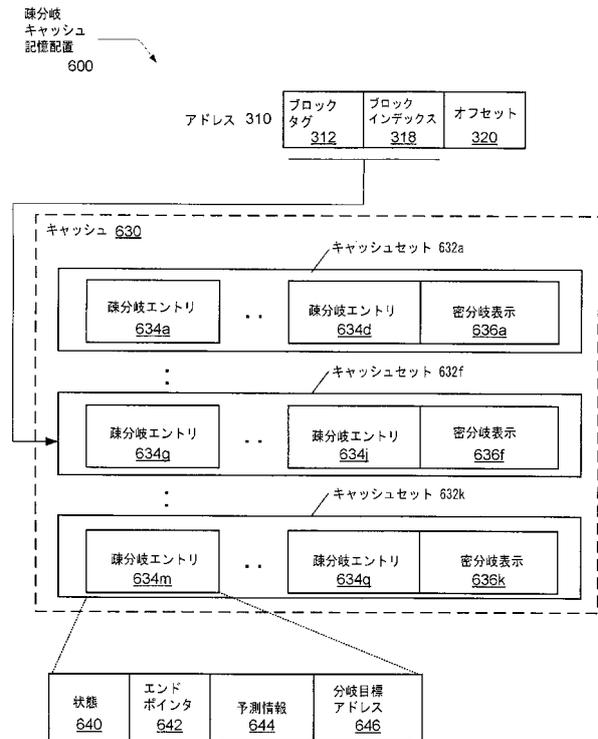
【図 4】



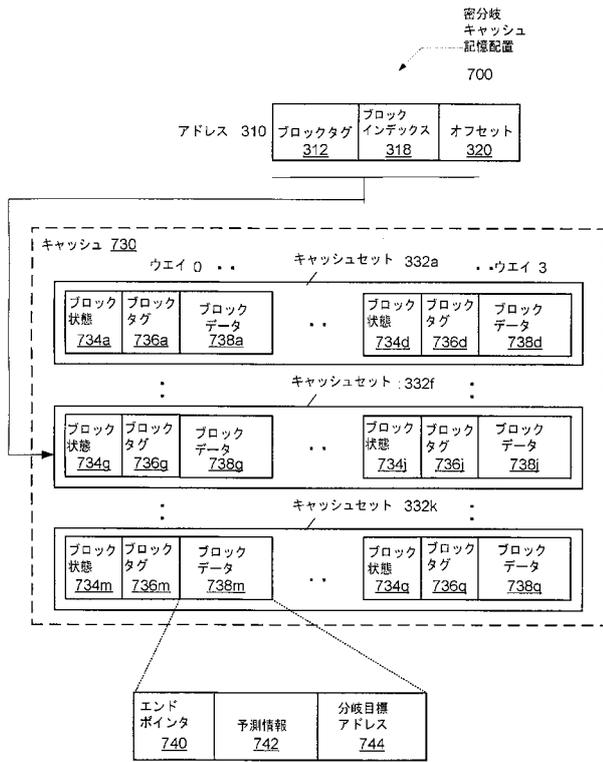
【図 5】



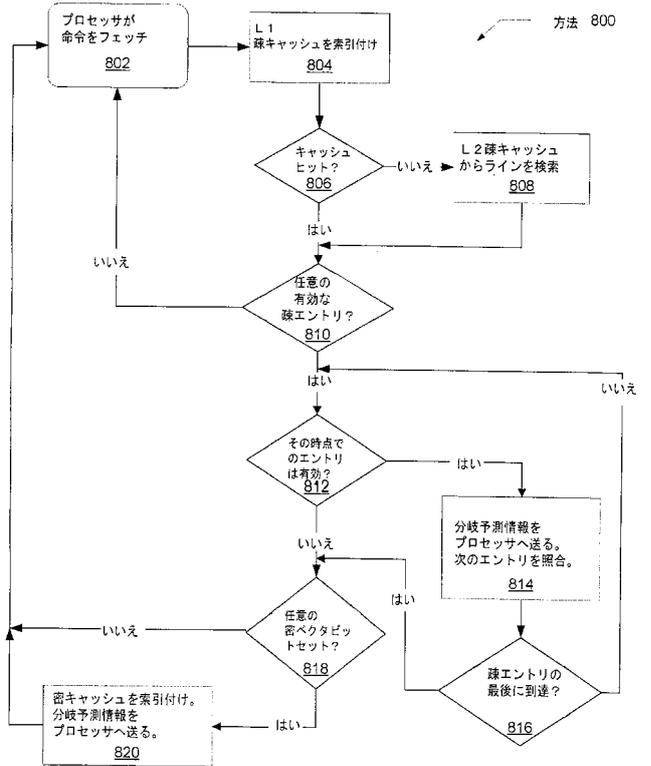
【図 6】



【 図 7 】



【 図 8 】



【国際調査報告】

INTERNATIONAL SEARCH REPORT

 International application No
 PCT/US2009/056136

A. CLASSIFICATION OF SUBJECT MATTER INV. G06F9/38		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, INSPEC, IBM-TDB, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 99/23553 A1 (ADVANCED MICRO DEVICES INC [US]) 14 May 1999 (1999-05-14) page 2, line 1 - page 3, line 11	1-20
A	US 6 425 075 B1 (STILES DAVID R [US] ET AL) 23 July 2002 (2002-07-23) column 3, line 20 - column 4, line 29	1-20
A	WO 03/017091 A1 (ADVANCED MICRO DEVICES INC [US]; ZURASKI GERALD D [US]; ROBERTS JAMES) 27 February 2003 (2003-02-27) page 1, line 38 - page 2, line 15	1-20
A	EP 0 798 632 A2 (SUN MICROSYSTEMS INC [US] SUN MICROSYSTEMS INC) 1 October 1997 (1997-10-01) column 2, line 26 - line 58	1-20
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents :		
A document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) *O* document referring to an oral disclosure, use, exhibition or other means *P* document published prior to the international filing date but later than the priority date claimed		*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. *Z* document member of the same patent family
Date of the actual completion of the international search 3 November 2009		Date of mailing of the international search report 16/11/2009
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax (+31-70) 340-3016		Authorized officer Thibaudeau, Jean

3

INTERNATIONAL-SEARCH REPORT

Information on patent family members

International application No

PCT/US2009/056136

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9923553	A1	14-05-1999	EP 1025487 A1 09-08-2000
			JP 2001522082 T 13-11-2001
			US 5974542 A 26-10-1999
US 6425075	B1	23-07-2002	NONE
WO 03017091	A1	27-02-2003	CN 1535415 A 06-10-2004
			EP 1410177 A1 21-04-2004
			JP 3798404 B2 19-07-2006
			JP 2005500616 T 06-01-2005
			TW 591530 B 11-06-2004
			US 6854050 B1 08-02-2005
EP 0798632	A2	01-10-1997	US 7024545 B1 04-04-2006
			DE 69715280 D1 17-10-2002
			DE 69715280 T2 07-08-2003
			JP 10055276 A 24-02-1998

フロントページの続き

(81)指定国 AP(BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP(AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW

(特許庁注：以下のものは登録商標)

1 . J A V A

(74)代理人 100162156

弁理士 村雨 圭介

(72)発明者 ジェラルド ディー・ブラスキ ジュニア .

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 コーヴ 8009

(72)発明者 ジェイムズ ディー・ダングス

アメリカ合衆国、テキサス州 78735、オースティン、ゲーンズ ランチ ループ #523
4424

(72)発明者 アンソニー エックス・ジャーヴィス

アメリカ合衆国、マサチューセッツ州 01720、アクトン、ハモンド ストリート 95

Fターム(参考) 5B005 JJ13 MM02 MM03 MM05 NN23 NN42

5B013 AA01 BB02