

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)公開番号

特開2022-94507  
(P2022-94507A)

(43)公開日 令和4年6月27日(2022.6.27)

(51)国際特許分類 F I テーマコード(参考)  
G 0 6 F 9/38 (2006.01) G 0 6 F 9/38 3 3 0 A 5 B 0 1 3

審査請求 未請求 請求項の数 6 O L (全20頁)

(21)出願番号	特願2020-207435(P2020-207435)	(71)出願人	000005223 富士通株式会社 神奈川県川崎市中原区上小田中4丁目1番1号
(22)出願日	令和2年12月15日(2020.12.15)	(74)代理人	100107766 弁理士 伊東 忠重
		(74)代理人	100070150 弁理士 伊東 忠彦
		(72)発明者	岡崎 亮平 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		Fターム(参考)	5B013 BB01

(54)【発明の名称】 演算処理回路及び演算処理方法

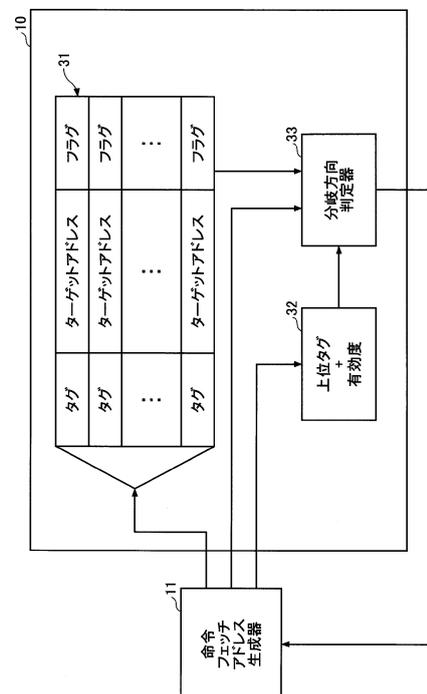
(57)【要約】

【課題】短いレイテンシで分岐予測可能な演算処理装置を提供する。

【解決手段】演算処理回路は、1つのフェッチラインに複数の命令が含まれるようにフェッチアドレスを順次生成する命令フェッチアドレス生成器と、1つのフェッチライン内でビット値が変化する命令アドレスビットを少なくとも1ビット含むようにフェッチアドレスから抽出されたビット列をインデックスとした複数のインデックスの各々に対して、1個の分岐先アドレスとフェッチアドレスの上位側のビット列である1個の第1のタグとを格納するアドレステーブルと、フェッチアドレスに応じてアドレステーブルから読み出された第1のタグと上位タグ格納部に格納されている第2のタグとが、フェッチアドレスの対応部分に一致する場合に、フェッチアドレスに応じてアドレステーブルから読み出された分岐先アドレスを命令フェッチアドレス生成器に供給する分岐方向判定部とを含む。

【選択図】図2

分岐予測機構の第1実施例の構成の一例を示す図



## 【特許請求の範囲】

## 【請求項 1】

1つのフェッチラインに複数の命令が含まれるようにフェッチアドレスを順次生成する命令フェッチアドレス生成器と、

前記1つのフェッチライン内でビット値が変化する命令アドレスビットを少なくとも1ビット含むように前記フェッチアドレスから抽出されたビット列をインデックスとした複数のインデックスの各々に対して、1個の分岐先アドレスと、前記ビット列よりも前記フェッチアドレスの上位側のビット列である1個の第1のタグと、を格納するアドレステーブルと、

前記フェッチアドレスを構成する複数個のビットのうち前記第1のタグよりも上位側のビット列である1個の第2のタグを格納する上位タグ格納部と、 10

前記フェッチアドレスに応じて前記アドレステーブルから読み出された前記第1のタグと前記上位タグ格納部に格納されている前記第2のタグとが、前記フェッチアドレスの対応部分に一致する場合に、前記フェッチアドレスに応じて前記アドレステーブルから読み出された前記分岐先アドレスを前記命令フェッチアドレス生成器に供給する分岐方向判定部と、

を含む演算処理回路。

## 【請求項 2】

前記命令フェッチアドレス生成器は所定のクロック信号に同期して前記フェッチアドレスを順次生成し、前記分岐方向判定部が前記クロック信号の第1のクロックサイクルで生成した第1のフェッチアドレスに応じて第1の分岐先アドレスを前記命令フェッチアドレス生成器に供給した場合、前記命令フェッチアドレス生成器は前記第1のクロックサイクルに続く第2のクロックサイクルにおいて前記第1の分岐先アドレスをフェッチアドレスとして生成する請求項1記載の演算処理回路。 20

## 【請求項 3】

前記1つのフェッチラインに含まれる前記複数の命令のうちで連続する2つの命令の命令アドレス間で変化するビットより1つ上のビットが、前記インデックスの最下位ビットである請求項1又は2記載の演算処理回路。

## 【請求項 4】

前記インデックスは4ビットである請求項1乃至3いずれか一項記載の演算処理回路。 30

## 【請求項 5】

前記フェッチアドレスを構成する前記複数個のビットのうち前記インデックスと比較して上位側にあり且つより長いビット列をインデックスとした複数のインデックスの各々に対して複数個のタグを有するアドレステーブルに基づいて分岐予測を行う予測器を更に含み、前記分岐方向判定部による第1の分岐予測に応じて前記命令フェッチアドレス生成器がフェッチアドレスを生成した後に前記予測器による第2の分岐予測が得られ、前記第2の分岐予測が前記第1の分岐予測と異なる場合に、前記命令フェッチアドレス生成器は前記第2の分岐予測に基づいてフェッチアドレスを生成し直す請求項1乃至4いずれか一項記載の演算処理回路。

## 【請求項 6】

1つのフェッチラインに複数の命令が含まれるようにフェッチアドレスを順次生成し、前記1つのフェッチライン内でビット値が変化する命令アドレスビットを少なくとも1ビット含むように前記フェッチアドレスから抽出されたビット列をインデックスとした複数のインデックスの各々に対して、1個の分岐先アドレスと、前記ビット列よりも前記フェッチアドレスの上位側のビット列である1個の第1のタグと、を格納するアドレステーブルから、前記フェッチアドレスに応じて前記第1のタグと前記分岐先アドレスとを読み出し、

前記フェッチアドレスを構成する複数個のビットのうち前記第1のタグよりも上位側のビット列である1個の第2のタグと、前記フェッチアドレスに応じて前記アドレステーブルから読み出された前記第1のタグとが、前記フェッチアドレスのそれぞれの対応部分に一 50

致する場合に、前記フェッチアドレスに応じて前記アドレステーブルから読み出された前記分岐先アドレスを前記フェッチアドレスとして用いる各段階を実行する演算処理方法。

【発明の詳細な説明】

【技術分野】

【0001】

本願開示は、演算処理回路及び演算処理方法に関する。

【背景技術】

【0002】

プロセッサの高速化技術のひとつである分岐予測技術では、実行される分岐命令に対して、分岐命令の分岐が成立する(TAKEN)又は成立しない(NOT\_TAKEN)のいずれかを示す分岐方向と、分岐命令のターゲットアドレス(分岐先アドレス)とを予測する。分岐命令の処理によって実際に分岐したか否かが判定される前に、予測に応じて、先行して投機的に後続命令の処理を進めていくことが可能となる。

10

【0003】

分岐予測が失敗した場合、先行して投機的に実行されていた後続命令の処理を全て破棄し、正しい後続命令について処理を実行し直さなければならないため、プロセッサの処理性能が低下する。分岐予測の精度を高めてプロセッサ処理性能の低下を避けるためには、計算量の多い複雑な分岐予測演算を実行することになるため、分岐予測を完了するまでにレイテンシ1よりも長いレイテンシが必要になってしまう。

20

【0004】

しかしながら分岐予測のレイテンシが長いと、命令フェッチの進行に間隙が空いてしまい、十分なスループットを維持できないという問題がある。例えば、分岐予測のレイテンシが3サイクルであるとする。第1の命令をフェッチした後、シーケンシャル方向の命令フェッチ(一定のアドレス増分での命令フェッチ)がその後の各サイクルにおいて実行されつつ、第1の命令に対する分岐予測演算が実行される。この第1の命令に対する分岐予測の結果が得られるのは、第3の命令をフェッチした後(即ち3サイクル経過後)である。分岐予測の結果が分岐成立を示す場合、シーケンシャル方向にフェッチしていた第2の命令及び第3の命令はキャンセルされ、分岐予測の結果が示すターゲットアドレスに対する命令フェッチが新たに実行される。この場合、分岐予測のレイテンシの長さに応じてスループットが低下してしまう。

30

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特開2018-063684号公報

【特許文献2】特開平05-143334号公報

【特許文献3】特表2017-509995号公報

【発明の概要】

【発明が解決しようとする課題】

【0006】

以上を鑑みると、短いレイテンシで分岐予測可能な演算処理装置が望まれる。

40

【課題を解決するための手段】

【0007】

演算処理回路は、1つのフェッチラインに複数の命令が含まれるようにフェッチアドレスを順次生成する命令フェッチアドレス生成器と、前記1つのフェッチライン内でビット値が変化する命令アドレスビットを少なくとも1ビット含むように前記フェッチアドレスから抽出されたビット列をインデックスとした複数のインデックスの各々に対して、1個の分岐先アドレスと、前記ビット列よりも前記フェッチアドレスの上位側のビット列である1個の第1のタグと、を格納するアドレステーブルと、前記フェッチアドレスを構成する複数個のビットのうち前記第1のタグよりも上位側のビット列である1個の第2のタグを

50

格納する上位タグ格納部と、前記フェッチアドレスに応じて前記アドレステーブルから読み出された前記第1のタグと前記上位タグ格納部に格納されている前記第2のタグとが、前記フェッチアドレスの対応部分に一致する場合に、前記フェッチアドレスに応じて前記アドレステーブルから読み出された前記分岐先アドレスを前記命令フェッチアドレス生成器に供給する分岐方向判定部とを含む。

【発明の効果】

【0008】

少なくとも1つの実施例によれば、短いレイテンシで分岐予測可能な演算処理装置が提供される。

【図面の簡単な説明】

【0009】

【図1】プロセッサの構成の一例を示す図である。

【図2】分岐予測機構の第1実施例の構成の一例を示す図である。

【図3】図2に示す分岐予測機構及び命令フェッチアドレス生成器の動作を示すフローチャートである。

【図4】フェッチラインの構成の一例を示す図である。

【図5】第2実施例に用いられる分岐予測器の構成の一例を示す図である。

【図6】分岐予測機構の第2実施例の構成の一例を示す図である。

【図7】分岐予測機構の第2実施例における命令フェッチの一例を示す図である。

【図8】分岐予測機構の第2実施例における命令フェッチの一例を示す図である。

【図9】第1予測器におけるエントリ更新処理の一例を示すフローチャートである。

【図10】分岐予測機構を備えたプロセッサを用いたシステム構成の一例を示す図である。

【発明を実施するための形態】

【0010】

分岐予測機構においては、通常、複数ウェイの各々に対してターゲットアドレスを格納したテーブルと、分岐方向を予測するテーブルとを備え、それぞれのテーブルをフェッチアドレスの一部（インデックス）により参照することで予測を行うことが多い。しかしながら、複数ウェイを有するテーブルを用いて予測を行うためには、インデックスにより各ウェイを並列に引き、それぞれのウェイがヒットか否かを調べ、複数のウェイがヒットである場合には1つのウェイを選択する、という各動作を実行する必要がある。これらの動作を実行するためには論理回路の論理段数が多くなり、長いレイテンシがかかってしまう。またレイテンシを短くしようとすると、高い動作クロック周波数を維持することが困難になる。

【0011】

ウェイの数が多いたことが論理段数の増加に繋がるので、ウェイの個数を1個のみにすれば、分岐予測演算に必要な論理段数の観点からはレイテンシ1での高速な分岐予測を実装することが可能である。しかしながら単純にウェイの個数を1個のみにしてしまったのでは、以下のようなテーブル格納エントリの衝突という問題が生じる。

【0012】

一般に、1つのフェッチアドレスにより読み出される1つのフェッチラインには複数（例えば8個）の命令が含まれており、これら複数の命令に対して纏めて分岐予測が実行される。例えば1つのフェッチラインに4バイト長の命令が8個含まれており、この8個の命令のうちで2個の命令が分岐命令である場合を考える。フェッチアドレスが0x1000であるとすると、対応フェッチラインに含まれる第1の命令乃至第8の命令の格納アドレスは、0x1000、0x1004、0x1008、0x100C、0x1010、0x1014、0x1018、及び0x101Cである。

【0013】

複数ウェイを用いる分岐予測機構では、通常、フェッチアドレスがシーケンシャル方向に所定固定量ずつ増加する毎に、1ずつ増加するインデックスを用いる。具体的には、上記

10

20

30

40

50

の例であれば、下位5ビット(bit 0 ~ bit 4)より上に位置する例えばbit 5 ~ bit 8(以降bit [8 : 5]と表記する)をインデックスとして用いる。フェッチアドレス0x1000に対するインデックスbit [8 : 5]は"0000"であり、シーケンシャル方向における次のフェッチアドレス0x1100に対するインデックスbit [8 : 5]は"0001"である。

**【0014】**

上述の例のように、フェッチアドレス0x1000に対応するフェッチラインに2個の分岐命令が含まれている場合であっても、テーブルに複数のウェイが設けられていれば、2個の分岐命令がそれぞれ異なるウェイに対応してテーブルに格納されることになる。従って、インデックス"0000"における複数のウェイを並列に参照することにより、2個の分岐命令のそれぞれに対して分岐予測結果を得ることができる。

10

**【0015】**

このような場合において、ウェイの個数を1個のみにしてしまうと、テーブル内でインデックス"0000"に対するエントリは1個のみとなる。従って、対応フェッチラインに含まれる2個の分岐命令のうちで、一方の命令はテーブル格納することができず、この命令についてレイテンシ1で予測を実行することはできない。以下に説明する本発明の実施例では上記の問題を解決するために、フェッチアドレスの一部を分岐予測に用いるインデックスとして用いる際に、1つのフェッチラインに対して少なくとも2つのインデックスが割り当てられるようにする。

**【0016】**

以下に、本発明の実施例を添付の図面を用いて詳細に説明する。以下の各図において、同一又は対応する構成要素は同一又は対応する番号で参照し、その説明は適宜省略する。

20

**【0017】**

図1は、プロセッサの構成の一例を示す図である。図1及び以下の同様の図において、各ボックスで示される各回路又は機能ブロックと他の回路又は機能ブロックとの境界は、基本的には機能的な境界を示すものであり、物理的な位置の分離、電気的な信号の分離、制御論理的な分離等に対応するとは限らない。各回路又は機能ブロックは、他のブロックと物理的にある程度分離された1つのハードウェアモジュールであってもよいし、或いは他のブロックと物理的に一体となったハードウェアモジュール中の1つの機能を示したものであってもよい。

30

**【0018】**

図1に示すプロセッサは、分岐予測機構10、命令フェッチアドレス生成器11、命令バッファ12、命令デコーダ13、及びレジスタリネーミング14を含む。プロセッサは更に、RSA (Reservation Station for Address generate) 15、RSE (Reservation Station for Execute) 16、RSF (Reservation Station for Floating) 17、及びRSBR (Reservation Station for Branch) 18を含む。プロセッサは更に、CSE (Commit Stack Entry) 19、オペランドアドレス生成器20、演算器21、演算器22、及びネクストプログラムカウンタ(PC) 23を含む。プロセッサは更に、固定小数点更新バッファ26、浮動小数点更新バッファ27、固定小数点レジスタ28、及び浮動小数点レジスタ29を含む。プロセッサは更に、2次命令キャッシュ102、1次命令キャッシュ103、及び1次データキャッシュ104を含んでよい。これらのキャッシュのうちの幾つかはプロセッサの外部に設けられていてもよい。2次命令キャッシュ102はメインメモリ101に接続されている。

40

**【0019】**

命令フェッチアドレス生成器11は、命令フェッチアドレスを格納するプログラムカウンタを内蔵し、当該プログラムカウンタの格納値を一定値ずつ増加させることにより、命令フェッチアドレスを順次生成する。1つのフェッチアドレスにより読み出される1つのフェッチラインには複数(例えば8個)の命令が含まれてよい。1つのフェッチラインに含まれる命令数は特に限定されるものではない。

**【0020】**

50

分岐予測機構 10 は、命令フェッチアドレス生成器 11 から出力された命令フェッチアドレスを受け取り、命令フェッチと並行して分岐予測を実行する。分岐予測機構 10 は、受け取った命令フェッチアドレスを基に分岐予測を行い、分岐成立又は分岐不成立を示す分岐方向と分岐先アドレス（ターゲットアドレス）とを命令フェッチアドレス生成器 11 に返す。予測された分岐方向が分岐成立であった場合、命令フェッチアドレス生成器 11 は次の命令フェッチアドレスとして分岐予測機構 10 によって予測された分岐先アドレスを選択する。分岐予測機構 10 の構成及び動作については後ほど説明する。

【 0 0 2 1 】

命令フェッチアドレス生成器 11 は、命令フェッチアドレスと命令フェッチリクエストとを 1 次命令キャッシュ 103 に発行する。これに応じて、1 次命令キャッシュ 103 から命令フェッチアドレスに対応するフェッチラインの命令が読み出され、読み出された命令が命令バッファ 12 に格納される。

10

【 0 0 2 2 】

命令バッファ 12 は、プログラムの命令順通り即ちインオーダーで格納した命令を命令デコーダ 13 に供給する。命令デコーダ 13 は、プログラムの命令順通り即ちインオーダーで命令のデコード処理を行い、インオーダーで命令を発行する。レジスタリネーミング 14 は、命令に示されるレジスタと物理的なレジスタとの対応関係を動的に切り替えることにより、命令の並列実効性を高める。この命令発行処理及びレジスタリネーミング処理を介して、キュー構造のストレージである R S A 15、R S E 16、R S F 17、又は R S B R 18 に、各命令を示すエントリが、デコードした命令の種類に従って作成される。

20

【 0 0 2 3 】

R S A 15 は、主記憶オペランドアドレスを生成してロード命令やストア命令を実行するために、作成されたエントリをプログラムの命令順に拘束されず即ちアウトオブオーダーで制御するリザベーションステーションである。R S A 15 による制御に基づいて、オペランドアドレス生成器 20 がアクセス先のアドレスを生成し、1 次データキャッシュ 104 の当該アドレスに対してロード命令やストア命令が実行される。ロード命令により読み出されたデータは、固定小数点更新バッファ 26 又は浮動小数点更新バッファ 27 の指定されたレジスタに格納される。

【 0 0 2 4 】

R S E 16 は、指定されたレジスタのデータに対して指定された固定小数点演算を実行するために、作成されたエントリをアウトオブオーダーで制御するリザベーションステーションである。R S E 16 は、実行対象の命令が参照するレジスタの依存性をチェックし、依存性のあるレジスタの更新状況や演算器 21 のうちの同じ実行資源を用いる命令の実行状況等から、実行資源である演算器 21 が命令を実行可能かどうかを判断する。実行可能であるなら、R S E 16 は、受け取った命令の順序によらずにアウトオブオーダーで各演算器 21 に対してレジスタ番号やオペランドアドレス等の命令実行に必要な情報を出力する。R S E 16 による制御に基づいて、演算器 21 が、固定小数点更新バッファ 26 及び固定小数点レジスタ 28 の指定されたレジスタのデータに対して指定された固定小数点演算を実行し、固定小数点更新バッファ 26 の指定されたレジスタに演算結果を格納する。

30

【 0 0 2 5 】

R S F 17 は、指定されたレジスタのデータに対して指定された浮動小数点演算を実行するために、作成されたエントリをアウトオブオーダーで制御するリザベーションステーションである。R S F 17 は、上記説明した R S E 16 と同様の動作により、アウトオブオーダーで各演算器 22 に対してレジスタ番号やオペランドアドレス等の命令実行に必要な情報を出力する。R S F 17 による制御に基づいて、演算器 22 が、浮動小数点更新バッファ 27 及び浮動小数点レジスタ 29 の指定されたレジスタのデータに対して指定された浮動小数点演算を実行し、浮動小数点更新バッファ 27 の指定されたレジスタに演算結果を格納する。

40

【 0 0 2 6 】

R S B R 18 は、分岐命令を実行するためのリザベーションステーションである。R S B

50

R 1 8 は、命令デコーダ 1 3 から分岐命令の種別を受け取ると共に、命令アドレスバッファから分岐命令アドレスを受け取り、分岐命令毎にそれぞれの信号を格納する。その後、R S B R 1 8 は、演算器 2 1 から分岐先アドレスや分岐条件となる演算の結果を受け取り、得られた演算結果が分岐条件を満たしていれば分岐成立、満たしていなければ分岐不成立の判断を行い、分岐方向を確定する。

【 0 0 2 7 】

また R S B R 1 8 では、上記演算結果が示す実際の分岐アドレス及び分岐方向と分岐予測時の分岐アドレス及び分岐方向とが一致するか否かの判断や、分岐命令の順序関係の制御も行う。R S B R 1 8 は、エン트리作成時に分岐予測時の分岐方向や分岐先アドレスを登録し、演算結果と予測との一致検証を実行後、演算結果によりエントリの置き換えを行う。具体的には、R S B R 1 8 は、命令デコーダ 1 3 から分岐命令の種別を受け取ると共に、分岐命令アドレスを受け取り、分岐命令毎にそれぞれの信号を格納する。

10

【 0 0 2 8 】

R S B R 1 8 は、実際の分岐命令の実行結果と分岐予測結果とが一致した場合、C S E 1 9 へ分岐命令の完了報告を出力すると共に、分岐予測機構 1 0 に完了した分岐命令の情報を供給する。R S B R 1 8 は、実際の分岐命令の実行結果と分岐予測結果とが一致しなかった場合、C S E 1 9 へ分岐命令の完了報告と共に後続命令のキャンセル要求を出力する。R S B R 1 8 は、実際の分岐命令の実行結果と分岐予測結果とが一致しなかった場合は更に、命令フェッチアドレス生成器 1 1 へ再命令フェッチ要求と再命令フェッチのアドレスとを出力し、更に、分岐予測機構 1 0 に完了した分岐命令の情報を供給する。

20

【 0 0 2 9 】

命令デコーダ 1 3 は更に、全てのデコードされた命令に命令識別子を割り振り、命令識別子を命令の順番通りに C S E 1 9 に送る。C S E 1 9 は、デコードされた命令の命令識別子を命令の実行順番通りに蓄積するキュー構造のストレージと、キューの情報と各処理パイプからの演算完了報告とに基づいてコミット処理を行う完了処理回路とに分けられる。デコーダで解読された命令の命令識別子は C S E 1 9 のキューに蓄えられ、処理パイプからの完了報告を待つ。各リザベーションステーションがアウトオブオーダーで実行した命令についての完了報告を C S E 1 9 に送ると、C S E 1 9 の完了処理回路が、キューに蓄えられた完了待ちの命令の中から完了報告に対応する命令をインオーダーで開放し、完了を確定させる。

30

【 0 0 3 0 】

C S E 1 9 から開放され完了が確定する命令については、対応する資源の更新を行なう。ロード命令や演算命令等の場合には、固定小数点更新バッファ 2 6 や浮動小数点更新バッファ 2 7 のデータを、固定小数点レジスタ 2 8 や浮動小数点レジスタ 2 9 に転送し、命令実行結果をソフトウェアからアクセス可能なレジスタに反映させる。また同時に、ネクストプログラムカウンタ 2 3 の値を、実行完了して開放される命令の次の命令のアドレスを指し示すように適切な増分だけ変化させる。なおネクストプログラムカウンタ 2 3 は、割り込み発生時等の特殊な場合に用いられるプログラムカウンタであり、プログラムの通常の進行中に用いられることはない。

【 0 0 3 1 】

図 2 は、分岐予測機構 1 0 の第 1 実施例の構成の一例を示す図である。図 2 に示す分岐予測機構 1 0 は、アドレステーブル 3 1、上位タグ格納部 3 2、及び分岐方向判定部 3 3 を含む。

40

【 0 0 3 2 】

図 2 に示される命令フェッチアドレス生成器 1 1 は、図 1 に示される命令フェッチアドレス生成器 1 1 であり、1つのフェッチラインに複数の命令が含まれるようにフェッチアドレスを順次生成する。命令フェッチアドレス生成器 1 1 がシーケンシャル方向に順次生成するフェッチアドレスにおいて、あるフェッチアドレスとその直後のフェッチアドレスとの差は所定の一定値（例えば 3 2 バイト）である。この場合 1つのフェッチラインの長さは、当該差に等しい長さ（例えば 3 2 バイト）である。例えば 3 2 バイト長である 1つの

50

フェッチラインには、例えば4バイト長の命令が8個含まれてよい。以下の説明においては、フェッチラインの長さが32バイトであり、4バイト長の命令が8個含まれる場合を例として用いるが、この例の構成は限定を意図するものではない。

【0033】

アドレステーブル31は、1つのフェッチライン内でビット値が変化する命令アドレスビットを少なくとも1ビット含むようにフェッチアドレスから抽出されたビット列をインデックスとした複数のインデックスの各々に対して、エントリを格納する。各インデックスに対するエントリは、1個の分岐先アドレスと、インデックスのビット列よりもフェッチアドレスの上位側のビット列である1個の第1のタグと、エントリが有効か否かを示すフラグを含む。

10

【0034】

アドレステーブル31のインデックスは、フェッチラインが32バイト長である場合、例えばフェッチアドレスの下位3ビット(bit0~bit2)より上に位置する例えばbit[6:3](bit3~bit6)である。アドレステーブル31に格納される第1のタグは、インデックスよりも上位側のビット列であり、例えばbit[10:7](bit7~bit10)である。

【0035】

アドレステーブル31は、命令フェッチアドレス生成器11が生成するフェッチアドレスに応じたエントリを出力する。具体的には、命令フェッチアドレス生成器11からフェッチアドレスのbit[6:3]がアドレステーブル31に供給されると、アドレステーブル31は、当該インデックスに対応した1個のターゲットアドレスと1個の第1のタグと1個のフラグとを出力する。出力された1個のターゲットアドレスと1個の第1のタグと1個のフラグとは、分岐方向判定部33に供給される。

20

【0036】

上位タグ格納部32は、フェッチアドレスを構成する複数個のビットのうち上記第1のタグよりも上位側のビット列である1個の第2のタグを格納する。この例では、第1のタグbit[10:7]よりも上位側のビットとして、例えばbit[15:11]が第2のタグであってよい。第2のタグbit[15:11]はアドレスビットが16ビットの場合であり、例えばアドレスビットが32ビットであれば、第2のタグはbit[31:11]であってよい。

30

【0037】

命令フェッチアドレス生成器11からフェッチアドレスのbit[15:11]が上位タグ格納部32に供給されると、上位タグ格納部32は、格納されている第2のタグと供給されたbit[15:11]とを比較し、比較結果を示す信号を出力してよい。比較結果を示す信号は分岐方向判定部33に供給される。代替的に、上位タグ格納部32は第2のタグを分岐方向判定部33に供給し、分岐方向判定部33が、命令フェッチアドレス生成器11から供給されたフェッチアドレスのbit[15:11]と上位タグ格納部32から供給された第2のタグとを比較してもよい。

【0038】

分岐方向判定部33は、命令フェッチアドレス生成器11から第1のタグに対応するフェッチアドレスの部分(bit[10:7])を命令フェッチアドレス生成器11から受け取る。分岐方向判定部33は、フェッチアドレスに応じてアドレステーブル31から読み出された第1のタグと上位タグ格納部32に格納されている第2のタグとが、前記フェッチアドレスのそれぞれに対応する部分に一致するか否かを判定する。分岐方向判定部33は、判定結果が一致を示す場合に、フェッチアドレスに応じてアドレステーブル31から読み出されたターゲットアドレスを命令フェッチアドレス生成器11に供給する。即ち、分岐予測機構10は、分岐予測演算の結果として分岐成立を予測し、ターゲットアドレスを命令フェッチアドレス生成器11に供給する。

40

【0039】

分岐方向判定部33は、上記判定結果が一致を示している場合であっても、フェッチアド

50

レスに応じてアドレステーブル 3 1 から読み出されたフラグが無効を示す場合、ターゲットアドレスを命令フェッチアドレス生成器 1 1 に供給しない。当該フラグは、アドレステーブル 3 1 のエントリが有効であるか無効であるかを示す例えば 1 ビットのデータであり、当該フラグが無効を示す場合には、アドレステーブル 3 1 に格納されるエントリを分岐予測に用いることは適切でない。

#### 【 0 0 4 0 】

なお、例えば図 1 に示す R S B R 1 8 から供給される完了した分岐命令の情報に基づいて、実際の分岐命令の実行結果と分岐予測結果とが一致しなかった場合に、アドレステーブル 3 1 のエントリが更新されてよい。例えば分岐不成立と予測されたフェッチラインに対して R S B R 1 8 からの情報が分岐成立を示す場合等に、アドレステーブル 3 1 の対応するインデックスにおけるエントリ（ターゲットアドレス等）が新たに登録されてよい。また例えば分岐成立と予測されたフェッチラインに対して R S B R 1 8 からの情報が分岐不成立を示す場合等に、アドレステーブル 3 1 の対応するインデックスにおけるフラグがリセット（無効状態に設定）されてよい。

10

#### 【 0 0 4 1 】

また上位タグ格納部 3 2 には、格納される第 2 のタグに関して有効度を示す飽和カウンタが設けられている。例えば R S B R 1 8 からの完了分岐命令の情報によれば実際に分岐成立した命令が第 2 のタグに対応する命令アドレスのものである場合には、飽和カウンタを + 1 してよい。この際、飽和カウンタ値が既に上限値である場合には、値が + 1 されることはなく上限値に維持される。また例えば R S B R 1 8 からの完了分岐命令の情報によれば実際に分岐成立した命令が第 2 のタグに対応しない命令アドレスのものである場合には、飽和カウンタを - 1 してよい。また例えば R S B R 1 8 からの完了分岐命令の情報によれば実際に分岐成立した命令が第 2 のタグに対応しない命令アドレスのものであり且つ飽和カウンタ値が下限値に等しい場合には、完了分岐命令の命令アドレスにより第 2 のタグを更新してよい。この更新作業と共に、有効度を示す飽和カウンタを上限値に設定し、アドレステーブル 3 1 の全フラグをリセットしてよい。

20

#### 【 0 0 4 2 】

上記のように構成された分岐予測機構 1 0 においては、アドレステーブル 3 1 のウェイ数を 1 個とすることにより、複数のウェイがヒットした場合にウェイを選択する処理が不要になる。従って分岐予測演算において実行される処理の論理段数を削減することができ、高速な分岐予測演算が可能になる。

30

#### 【 0 0 4 3 】

また分岐予測機構 1 0 においては、アドレステーブル 3 1 のインデックスが例えば 4 ビットであることにより、アドレステーブル 3 1 のエントリ数が 1 6 個となる。エントリ数が 1 6 個程度と少ない数とすることにより、アドレステーブル 3 1 からインデックスに応じたエントリを読み出す論理演算処理を、少ない論理段数で実現することが可能となる。このようにして、アドレステーブル 3 1 からの高速読み出しを実現することができる。

#### 【 0 0 4 4 】

また分岐予測機構 1 0 においては、タグとして用いるべきビット列の上位側の一部は、上位タグ格納部 3 2 に第 2 のタグとして格納し、分岐予測機構 1 0 全体で 1 つだけ保持している。このような構成とせずに、アドレステーブル 3 1 に格納されるタグのビット数が多い構成にすると、アドレステーブル 3 1 から読み出されたタグに対して分岐方向判定部 3 3 が照合判定をするために必要な処理回路の論理段数が多くなってしまう。図 2 に示す分岐予測機構 1 0 が想定するような命令列は、アドレス空間が比較的狭い範囲の命令列であり、上位ビットの変動は少ない。従って、タグとして用いるべきビット列の上位側の一部は、上位タグ格納部 3 2 に第 2 のタグとして格納し、分岐予測機構 1 0 全体で 1 つだけ保持することにすれば十分である。このようにすれば、アドレステーブル 3 1 からインデックスに対応するエントリを読み出す動作と同時に、上位タグ格納部 3 2 に格納される第 2 のタグを照合する動作を並行して実行することが可能になり、タグの照合にかかる時間が短くなる。

40

50

## 【 0 0 4 5 】

分岐予測機構 10 を上述のような構成とすることにより、分岐予測演算にかかる時間を短くして、分岐予測演算を 1 レイテンシ内に完了させることができる。このことは具体的には以下の動作を意味する。まず命令フェッチアドレス生成器 11 は所定のクロック信号に同期してフェッチアドレスを順次生成している。分岐方向判定部 33 がクロック信号の第 1 のクロックサイクルで生成した第 1 のフェッチアドレスに応じて分岐成立を予測し、第 1 の分岐先アドレスを命令フェッチアドレス生成器 11 に供給したとする。この場合、命令フェッチアドレス生成器 11 は、クロック信号の第 1 のクロックサイクルに続く第 2 のクロックサイクルにおいて、上記第 1 の分岐先アドレスをフェッチアドレスとして生成する。即ち、命令フェッチアドレス生成器 11 は、第 1 のクロックサイクルで生成したフェッチアドレスに対する分岐予測結果に基づいて、直後に続く第 2 のクロックにおいてフェッチアドレスを生成することができる。

10

## 【 0 0 4 6 】

分岐予測演算を 1 レイテンシ内に完了させることにより、分岐予測演算結果が分岐を示す場合に、フェッチされた命令をキャンセルする必要がなくなり、効率的なフェッチ動作を実現することができる。即ち、分岐予測結果に影響しない高いスループットを維持することができる。

## 【 0 0 4 7 】

図 3 は、図 2 に示す分岐予測機構 10 及び命令フェッチアドレス生成器 11 の動作を示すフローチャートである。なお図 3 及び以降の図において、フローチャートに記載された各ステップの実行順序は一例にすぎず、本願の意図する技術範囲が、記載された実行順番に限定されるものではない。例えば、A ステップの次に B ステップが実行されるように本願に説明されていたとしても、A ステップの次に B ステップを実行することが可能なだけでなく、B ステップの次に A ステップを実行することが、物理的且つ論理的に可能である場合がある。この場合、どちらの順番でステップを実行しても、当該フローチャートの処理に影響する全ての結果が同一であるならば、本願に開示の技術の目的のためには、B ステップの次に A ステップが実行されてもよいことは自明である。A ステップの次に B ステップが実行されるように本願に説明されていたとしても、上記のような自明な場合を本願の意図する技術範囲から除外することを意図するものではなく、そのような自明な場合は、当然に本願の意図する技術範囲内に属する。

20

30

## 【 0 0 4 8 】

ステップ S 1 において、命令フェッチアドレス生成器 11 がフェッチアドレスを分岐予測機構 10 に通知する。具体的には、フェッチアドレスの bit [ 6 : 3 ] がアドレステーブル 31 に供給され、フェッチアドレスの bit [ 10 : 7 ] が分岐方向判定部 33 に供給され、フェッチアドレスの bit [ 15 : 11 ] が上位タグ格納部 32 に供給される。

## 【 0 0 4 9 】

ステップ S 2 において分岐予測機構 10 が予測成立を示す場合 ( Y E S )、ステップ S 3 において、命令フェッチアドレス生成器 11 は分岐予測機構 10 から供給されたターゲットアドレスで次の命令フェッチを実行する。ステップ S 2 において分岐予測機構 10 が予測成立を示さない場合 ( N O )、命令フェッチアドレス生成器 11 は、ターゲットアドレスを受け取ることなく、ステップ S 3 においてシーケンシャル方向の次の命令アドレスで命令フェッチを実行する。

40

## 【 0 0 5 0 】

図 4 は、フェッチラインの構成の一例を示す図である。図 4 を用いて、図 2 に示されるアドレステーブル 31 におけるエントリ登録を説明する

図 4 において、32 バイト長である 1 つのフェッチラインに、各々が 4 バイト長である 8 個の命令 i 1 乃至 i 8 が含まれている。このフェッチラインをフェッチするためのフェッチアドレスは 0 x 1 0 0 0 である。このときフェッチラインに含まれる命令 i 1 乃至 i 8 の格納アドレスはそれぞれ、0 x 1 0 0 0、0 x 1 0 0 4、0 x 1 0 0 8、0 x 1 0 0 C、0 x 1 0 1 0、0 x 1 0 1 4、0 x 1 0 1 8、及び 0 x 1 0 1 C である。

50

## 【 0 0 5 1 】

図 4 に示される例では、この 8 個の命令  $i_1$  乃至  $i_8$  のうちで、2 個の命令  $i_5$  及び  $i_8$  が分岐命令である。図 4 において矢印 J 1 で示されるように命令  $i_5$  の分岐先は命令  $i_8$  のアドレスであり、矢印 J 2 で示されるように命令  $i_8$  の分岐先は命令  $i_1$  のアドレスである。

## 【 0 0 5 2 】

命令フェッチアドレス生成器 11 は、フェッチアドレス  $0 \times 1000$  で図 4 に示されるフェッチラインをフェッチすると共に、フェッチアドレス  $0 \times 1000$  を分岐予測機構 10 に供給して分岐予測演算を実行させる。分岐予測機構 10 では、フェッチアドレス  $0 \times 1000$  に応答して、命令  $i_1$  乃至  $i_8$  のうちでシーケンシャル方向に並ぶ順番で最初の分岐命令である命令  $i_5$  に対する分岐予測を行うことが期待される。フェッチアドレス  $0 \times 1000$  のインデックス部分である  $bit[6:3]$  は "0000" であるので、アドレステーブル 31 におけるインデックス "0000" の位置に命令  $i_5$  に対するエントリが格納されることになる。例えば、図 1 に示す R S B R 18 から供給される完了した分岐命令  $i_5$  の情報に基づいて、アドレステーブル 31 に命令  $i_5$  に関するエントリが作成されてよい。

10

## 【 0 0 5 3 】

なお命令  $i_5$  のアドレス  $0 \times 1010$  の  $bit[6:3]$  は "0010" である。しかしながらフェッチアドレス  $0 \times 1000$  に応答して命令  $i_5$  に対する分岐予測を行うことが期待されるので、上述のように、インデックス "0000" の位置に命令  $i_5$  に対するエントリが格納されることになる。

20

## 【 0 0 5 4 】

命令  $i_8$  に対する分岐予測演算が実行されるのは、命令  $i_5$  に対する分岐予測結果が分岐成立を示し命令フェッチアドレス生成器 11 が命令  $i_8$  のアドレスに等しいターゲットアドレスで命令フェッチをしたときである。命令  $i_8$  のアドレス  $0 \times 1010$  のインデックス部分である  $bit[6:3]$  は "0011" であるので、アドレステーブル 31 におけるインデックス "0011" の位置に命令  $i_5$  に対するエントリが格納されることになる。例えば、図 1 に示す R S B R 18 から供給される完了した分岐命令  $i_8$  の情報に基づいて、アドレステーブル 31 に命令  $i_8$  に関するエントリが作成されてよい。

## 【 0 0 5 5 】

図 2 に示されるアドレステーブル 31 においては、各インデックスに対して 1 つのタグしか設けられておらず、ウェイ数が 1 となっている。このようにウェイ数が 1 であるにも関わらず、アドレステーブル 31 には、1 つのフェッチラインに含まれる 2 個又はそれ以上の個数の命令に対してエントリを登録することができる。これは、1 つのフェッチライン内でビット値が変化する命令アドレスビットを少なくとも 1 ビット含むように選択されたビット列をインデックスとして用いているためである。

30

## 【 0 0 5 6 】

例えば、命令アドレスの  $bit[5:2]$  をインデックスとして用いてもよい。この場合、1 つのフェッチラインに含まれる 8 個の命令の各々が格納される命令アドレスは、その  $bit[5:2]$  のビットパターンが互いに異なる。従って、理論的には 1 つのフェッチラインに含まれる 8 個の命令全てについて、アドレステーブル 31 にエントリを登録することができる。

40

## 【 0 0 5 7 】

但し連続する 2 つの命令が両方共に分岐命令である頻度は低いので、連続する 2 つの命令を両方ともアドレステーブル 31 に登録することに対する必要性は低い。従って、上述の例のようにインデックスとしては、命令アドレスの  $bit[6:3]$  を用いれば十分である。即ち、1 つのフェッチラインに含まれる複数の命令のうちで連続する 2 つの命令の命令アドレス間で変化するビットより 1 つ上のビットが、インデックスの最下位ビットであってよい。このようなインデックス構成とすることにより、より広い命令アドレス範囲をアドレステーブル 31 でカバーできることになり、効率的な分岐予測が可能になる。

50

## 【 0 0 5 8 】

図 5 は、第 2 実施例に用いられる分岐予測器の構成の一例を示す図である。第 2 実施例においては、図 2 に示す第 1 実施例の分岐予測機構 1 0 (以降第 1 予測器と呼ぶ場合がある)に加え、更に、図 5 に示される第 2 予測器 1 1 0 を用いる。

## 【 0 0 5 9 】

第 2 予測器 1 1 0 は、分岐予測機構 1 0 よりも複雑な分岐予測演算を実行し、分岐予測機構 1 0 よりも長いレイテンシでの分岐予測演算を行う。第 2 予測器 1 1 0 は例えば、レイテンシ 3 (動作クロックの 3 サイクルをかけて)の分岐予測を実行する。第 2 予測器 1 1 0 の構成は特に限定されないが、図 5 に示す例では、アドレステーブル 4 1、バイモダル予測部 4 2、予測生成部 4 3、フリップフロップ 4 4、フリップフロップ 4 5、及びフリップフロップ 4 6 を含む。

10

## 【 0 0 6 0 】

図 5 に示される命令フェッチアドレス生成器 1 1 は、図 1 に示される命令フェッチアドレス生成器 1 1 であり、前述のように、1つのフェッチラインに複数の命令が含まれるようにフェッチアドレスを順次生成する。第 1 の実施例と同様に、以下の説明においては、フェッチラインの長さが 3 2 バイトであり、4 バイト長の命令が 8 個含まれる場合を例として用いるが、この例の構成は限定を意図するものではない。

## 【 0 0 6 1 】

アドレステーブル 4 1 は、フェッチアドレスを構成する複数個のビットのうち、アドレステーブル 3 1 のインデックスと比較して上位側にあり且つより長いビット列をインデックスとした複数のインデックスの各々に対して複数個のタグを有する。ここで「比較して上位側にあるビット列」とは、アドレステーブル 4 1 のインデックスであるビット列の最下位ビットがアドレステーブル 3 1 のインデックスの最下位ビットよりも上位側にあることを意味する。

20

## 【 0 0 6 2 】

命令フェッチアドレス生成器 1 1 から、例えばフェッチアドレスのうちの `bit [ 1 3 : 5 ]` が、フリップフロップ 4 4 を介してアドレステーブル 4 1 に供給される。アドレステーブル 4 1 には、このフェッチアドレスの `bit [ 1 3 : 5 ]` に対応する各々が 9 ビット幅である 5 1 2 個のインデックスにそれぞれ対応して 5 1 2 個のエントリが格納されており、1つのエントリには例えば 8 個のタグが対応付けられていてよい。即ち、アドレステーブル 4 1 は 8 ウェイで 5 1 2 エントリの構成であってよい。

30

## 【 0 0 6 3 】

各タグは、アドレステーブル 4 1 のインデックスのビット列よりもフェッチアドレス上位側のビット列である。この例では、インデックスである `bit [ 1 3 : 5 ]` よりも上位側のビットとして、例えば `bit [ 3 1 : 1 4 ]` がタグであってよい。一つ一つのタグに対応して、1個の分岐先アドレスと、当該分岐先アドレスが有効か否かを示すフラグとが、アドレステーブル 4 1 に格納されている。

## 【 0 0 6 4 】

アドレステーブル 4 1 は、命令フェッチアドレス生成器 1 1 が生成するフェッチアドレスに応じた分岐先アドレス及びフラグを出力する。具体的には、フェッチアドレスの `bit [ 1 3 : 5 ]` がアドレステーブル 4 1 にインデックスとして供給されると、アドレステーブル 4 1 は、当該インデックスに対応した 8 個のタグと、8 個のターゲットアドレスと、8 個のフラグとを出力する。アドレステーブル 4 1 から出力された 8 個のタグと、8 個のターゲットアドレスと、8 個のフラグとは、フリップフロップ 4 6 に格納される。

40

## 【 0 0 6 5 】

バイモダル予測部 4 2 は、フェッチアドレスの `bit [ 1 3 : 5 ]` に対応する各々が 9 ビット幅である 5 1 2 個のインデックスにそれぞれ対応して 5 1 2 個の飽和カウンタが設けられていてよい。バイモダル予測部 4 2 は、命令フェッチアドレス生成器 1 1 が生成するフェッチアドレスに応じた飽和カウンタ値を出力する。具体的には、フェッチアドレスの `bit [ 1 3 : 5 ]` がバイモダル予測部 4 2 にインデックスとして供給されると、

50

バイモーダル予測部 4 2 は、当該インデックスに対応した 1 個の飽和カウンタ値を出力する。アドレステーブル 4 1 から出力された 1 個の飽和カウンタ値はフリップフロップ 4 5 に格納される。

#### 【 0 0 6 6 】

飽和カウンタは例えば 2 ビットのカウンタである。飽和カウンタの値として「 0 0 」が強い分岐不成立、「 0 1 」が弱い分岐不成立、「 1 0 」が弱い分岐成立、「 1 1 」が強い分岐成立を意味する。飽和カウンタ値が「 0 0 」又は「 0 1 」のときには分岐不成立と判定し、飽和カウンタ値が「 1 0 」又は「 1 1 」のときには分岐成立と判定する。飽和カウンタ値は、対応する分岐命令の実際の演算結果が分岐成立であれば + 1 され、対応する分岐命令の実際の演算結果が分岐不成立であれば - 1 される。但し上限値「 1 1 」以上に値は増大せず、下限値「 0 0 」以下に値は減少しない。このバイモーダル予測部 4 2 では、過去に分岐成立が連続した場合には今回も分岐が成立する可能性が高く、過去に分岐不成立が連続した場合には今回も分岐が不成立である可能性が高いという事実を分岐予測の原理として用いている。

10

#### 【 0 0 6 7 】

以下に、図 5 に示す第 2 予測器 1 1 0 の動作を、動作クロックのサイクル毎に説明する。まず動作クロックの第 1 サイクルにおいて命令フェッチアドレス生成器 1 1 からのフェッチアドレスがフリップフロップ 4 4 に格納される。第 2 サイクルにおいて、アドレステーブル 4 1 から読み出されたデータをフリップフロップ 4 4 に格納すると共に、バイモーダル予測部 4 2 から読み出された飽和カウンタ値をフリップフロップ 4 5 に格納する。第 3 サイクルにおいて、予測生成部 4 3 が、フリップフロップ 4 5 に格納された飽和カウンタ値とフリップフロップ 4 6 に格納されたデータとに基づいて分岐予測を行う。

20

#### 【 0 0 6 8 】

具体的には、予測生成部 4 3 は、フリップフロップ 4 6 から供給された 8 個のタグのうちで対応フラグが有効を示すタグとフェッチアドレスの対応部分とを照合し、一致（即ちヒット）するか否かを判定する。予測生成部 4 3 は、対応する飽和カウンタ値が分岐成立を示す場合に、タグが一致したウェイに対応するターゲットアドレスを命令フェッチアドレス生成器 1 1 に供給する。即ち、第 2 予測器 1 1 0 は、分岐予測演算の結果として分岐成立を予測し、ターゲットアドレスを命令フェッチアドレス生成器 1 1 に供給する。飽和カウンタ値が分岐不成立を示す場合、又は、タグの一致するウェイが存在しない場合には、第 2 予測器 1 1 0 は分岐不成立を予測する。タグが一致したウェイが複数個存在する場合には、一致ウェイに対応する複数の命令アドレスのうちでフェッチアドレスからシーケンシャル方向に進んだときに最初に現れる命令アドレスに対応するウェイを選択する。

30

#### 【 0 0 6 9 】

図 6 は、分岐予測機構の第 2 実施例の構成の一例を示す図である。分岐予測機構の第 2 実施例では、図 6 に示すように図 2 に示す第 1 予測器 1 0 と図 5 に示す第 2 予測器 1 1 0 とを用い、第 2 予測器 1 1 0 の分岐予測結果により第 1 予測器 1 0 による分岐予測結果を修正することで、より精度の高い分岐予測を行う。

#### 【 0 0 7 0 】

動作クロックの第 1 サイクルで命令フェッチアドレス生成器 1 1 からフェッチアドレスが第 1 予測器 1 0 と第 2 予測器 1 1 0 とに供給されると、命令フェッチアドレス生成器 1 1 がレイテンシ 1 で予測結果を生成する。この第 1 予測器 1 0 が生成した予測結果は、セレクタ 5 4 を介して命令フェッチアドレス生成器 1 1 に供給される。これにより、命令フェッチアドレス生成器 1 1 は、第 1 予測器 1 0 による予測結果をレイテンシ 1 で受け取り、次のフェッチアドレス生成に用いることができる。なおセレクタ 5 4 は、予測結果比較部 5 1 から受け取る選択制御信号が例えば値 0 の場合に第 2 予測器 1 1 0 による予測結果を選択し、選択制御信号が例えば値 1 の場合に第 1 予測器 1 0 による予測結果を選択する。選択制御信号は、通常、第 1 予測器 1 0 による予測結果を選択する値（例えば 1 ）に設定されている。

40

#### 【 0 0 7 1 】

50

動作クロックの第2サイクルで、第1予測器10の予測結果を示すデータはフリップフロップ52に格納される。更に動作クロックの第3サイクルで、第1予測器10の予測結果を示すデータはフリップフロップ53に格納されると共に予測結果比較部51に供給される。またこの第3サイクルにおいて、第2予測器110がレイテンシ3で予測結果を生成し、この予測結果が予測結果比較部51に供給される。予測結果比較部51は、第1予測器10による予測結果と第2予測器110による予測結果とを比較し、両者が一致する場合にはセクタ54に供給する選択制御信号の値を1に設定する。両者が一致しない場合には、予測結果比較部51は、セクタ54に供給する選択制御信号の値を0に設定する。

#### 【0072】

10

従って、第1予測器10による予測結果と第2予測器110による予測結果とが一致する場合には、第1予測器10による予測結果は追認されたことになり、第1予測器10による予測結果に基づいて発行されたフェッチアドレスはそのまま維持される。一方、第1予測器10による予測結果と第2予測器110による予測結果とが一致しない場合には、第1予測器10による予測結果は否定されたことになる。このとき第1予測器10による予測結果に基づいて既に発行されたフェッチアドレスはキャンセルされ、第2予測器110による予測結果に基づいて改めてフェッチアドレスが発行される。

#### 【0073】

このように図6に示される第2実施例では、第1予測器10による第1の分岐予測に応じて命令フェッチアドレス生成器11がフェッチアドレスを生成した後に第2予測器110による第2の分岐予測が得られる。第2の分岐予測が第1の分岐予測と異なる場合に、命令フェッチアドレス生成器11は第2の分岐予測に基づいてフェッチアドレスを生成し直す。この構成により、第1予測器10により短いレイテンシ(例えばレイテンシ1)での効率的な命令フェッチを可能にすると共に、第2予測器110による長いレイテンシ(例えばレイテンシ3)での信頼性の高い分岐予測を実現することができる。即ち、第1予測器10が正しく分岐予測する場面では効率的な命令フェッチを実現することができると共に、第1予測器10が正しく分岐予測できない場面では効率は落ちるが第2予測器110による信頼性のある分岐予測を実現することができる。

20

#### 【0074】

例えば第1予測器10が第1サイクル乃至第3サイクルで分岐不成立を予測し、第2予測器110も第1サイクル乃至第3サイクルで分岐不成立を予測したとする。この場合、命令フェッチアドレス生成器11は、第2サイクル及び第3サイクルでシーケンシャル方向のフェッチアドレスを生成する。第3サイクルにおいて、第3サイクルのフェッチアドレスに関する予測結果が第1予測器10から命令フェッチアドレス生成器11に通知される。その結果、命令フェッチアドレス生成器11は、第4サイクルでもシーケンシャル方向のフェッチアドレスを生成する。

30

#### 【0075】

また例えば第1予測器10が第1サイクル乃至第3サイクルで分岐不成立を予測し、第2予測器110が第1サイクル、第2サイクル、及び第3サイクルでそれぞれ分岐不成立、分岐不成立、及び分岐成立を予測したとする。この場合、命令フェッチアドレス生成器11は、第2サイクル及び第3サイクルでシーケンシャル方向のフェッチアドレスを生成する。第3サイクルにおいて、第1サイクルのフェッチアドレスに関する予測結果が第2予測器110から命令フェッチアドレス生成器11に通知される。その結果、第1予測器10による予測結果が第2予測器110による予測結果で訂正されることになり、命令フェッチアドレス生成器11は、第4サイクルにおいてターゲットアドレスをフェッチアドレスとして生成する。

40

#### 【0076】

また例えば第1サイクル、第2サイクル、及び第3サイクルで第1予測器10がそれぞれ分岐成立、分岐不成立、及び分岐不成立を予測し、第2予測器110がそれぞれ分岐不成立、分岐不成立、及び分岐成立を予測したとする。即ち、第1サイクルでのフェッチアド

50

レスに対する分岐予測が、第 1 予測器 1 0 及び第 2 予測器 1 1 0 の両方共に分岐成立であったとする。この場合の命令フェッチシーケンスが図 7 に一例として示される。

【 0 0 7 7 】

図 7 は、分岐予測機構の第 2 実施例における命令フェッチの一例を示す図である。図 7 において、各フェッチアドレスに対する分岐予測動作がパイプラインとして実行される様子が示されており、パイプラインの各ステージにおける動作は以下の通りである。

A : 命令フェッチアドレスを決定し分岐予測機構に供給

T : アドレステーブルからデータを取り出し予測対象命令用の演算実行

M : 分岐予測

B : 分岐予測結果転送

R : 後続する予測対象命令用の演算実行

なお上記各ステージは、レイテンシ 3 である分岐予測機構の動作に対応しており、第 2 実施例の例における第 2 予測器 1 1 0 が、上記のパイプラインに相当する分岐演算を実行してよい。

【 0 0 7 8 】

動作クロックの第 1 サイクル C 1 において、フェッチアドレス i 1 が発行され命令フェッチが実行されると共に、フェッチアドレス i 1 に対する分岐予測が開始される。フェッチアドレス i 1 に対する 6 個のステージ A 乃至 R の演算はそれぞれ対応する 6 個のサイクルで順次実行される。図 7 に示される例では、第 1 予測器 1 0 により第 1 サイクル C 1 において分岐成立の予測結果が得られる場合を想定しており、分岐先のターゲットアドレスは i 2 5 1 である。第 2 サイクル C 2 において、フェッチアドレス i 2 5 1 が発行され命令フェッチが実行されると共に、フェッチアドレス i 2 5 1 に対する分岐予測が開始される。

【 0 0 7 9 】

更に第 3 サイクル C 3 において、シーケンシャル方向でフェッチアドレス i 2 5 1 に後続するフェッチアドレス i 2 5 2 が発行され命令フェッチが実行されると共に、フェッチアドレス i 2 5 2 に対する分岐予測が開始される。この第 3 サイクル C 3 において、第 1 サイクルのフェッチアドレス i 1 に対する分岐予測結果がステージ M (図 7 においてハッチングされたステージ) において得られる。

【 0 0 8 0 】

この例では、第 2 予測器 1 1 0 が第 3 サイクルにおいて分岐成立を予測する場合を想定しているので、第 1 サイクルにおいて第 1 予測器 1 0 が予測した分岐成立が追認されることになる。従って、既に発行された第 2 サイクル C 2 のフェッチアドレス i 2 5 1 及び第 3 サイクル C 3 のフェッチアドレス i 2 5 2 はそのまま維持される。更に、次の第 4 サイクル C 4 において、シーケンシャル方向でフェッチアドレス i 2 5 2 に後続するフェッチアドレス i 2 5 3 が発行され命令フェッチが実行される。

【 0 0 8 1 】

図 8 は、分岐予測機構の第 2 実施例における命令フェッチの別の一例を示す図である。図 8 には、第 1 サイクル、第 2 サイクル、及び第 3 サイクルで第 1 予測器 1 0 がそれぞれ分岐成立、分岐不成立、及び分岐不成立を予測し、第 2 予測器 1 1 0 がそれぞれ分岐不成立、分岐不成立、及び分岐不成立を予測した場合が示されている。即ち、第 1 サイクルでのフェッチアドレスに対する分岐予測について、第 1 予測器 1 0 による予測結果が分岐成立であり、第 2 予測器 1 1 0 による予測結果が分岐不成立であった場合である。

【 0 0 8 2 】

動作クロックの第 1 サイクル C 1 において、フェッチアドレス i 1 が発行され命令フェッチが実行されると共に、フェッチアドレス i 1 に対する分岐予測が開始される。図 8 に示される例では、第 1 予測器 1 0 により第 1 サイクル C 1 において分岐成立の予測結果が得られる場合を想定しており、分岐先のターゲットアドレスは i 2 5 1 である。第 2 サイクル C 2 において、フェッチアドレス i 2 5 1 が発行され命令フェッチが実行されると共に、フェッチアドレス i 2 5 1 に対する分岐予測が開始される。

10

20

30

40

50

## 【 0 0 8 3 】

更に第3サイクルC3において、シーケンシャル方向でフェッチアドレスi251に後続するフェッチアドレスi252が発行され命令フェッチが実行されると共に、フェッチアドレスi252に対する分岐予測が開始される。この第3サイクルC3において、第1サイクルのフェッチアドレスi1に対する分岐予測結果がステージMにおいて得られる。

## 【 0 0 8 4 】

この例では、第2予測器110が第3サイクルにおいて分岐不成立を予測する場合を想定しているので、第1サイクルにおいて第1予測器10が予測した分岐成立が否定されることになる。従って、既に発行された第2サイクルC2のフェッチアドレスi251及び第3サイクルC3のフェッチアドレスi252はキャンセルされ、先読みで実行されていた命令実行シーケンスは取り消される。次の第4サイクルC4においては、シーケンシャル方向で第1サイクルC1のフェッチアドレスi1に後続するフェッチアドレスi2が発行され命令フェッチが実行される。

## 【 0 0 8 5 】

図9は、第1予測器におけるエントリ更新処理の一例を示すフローチャートである。第1実施例において分岐予測機構として第1予測器10のみが用いられる場合には、前述のようにアドレステーブル31のエントリ登録及びリセット並びに上位タグ格納部32の更新は、RSBR18からの情報に基づいて行われてよい。第2実施例においては第1予測器10と第2予測器110とが用いられているので、第2予測器110による予測結果に基づいて第1予測器10の登録、更新、及びリセットを行ってよい。

## 【 0 0 8 6 】

第2予測器110による予測が分岐成立を示す状況が発生すると(ステップS11)、ステップS12の処理が実行される。ステップS12において、第1予測器10は、第2予測器110により分岐成立と予測された命令アドレスの上位部分と上位タグ(上位タグ格納部32に格納される第2のタグ)とが一致するか否かを判定する。両者が一致する場合、処理はステップS13に進む。

## 【 0 0 8 7 】

ステップS13において、第1予測器10は、分岐成立と予測された命令に関するエントリを、アドレステーブル31において対応フェッチアドレスが指し示すインデックス位置に作成する。具体的には、第1予測器10は、当該インデックス位置において対応フェッチアドレスのbit[10:7]を第1のタグとして格納し、予測されたターゲットアドレスを格納し、フラグを有効に設定する。第1予測器10は更に、上位タグ格納部32の有効度(飽和カウンタ値)を+1する。

## 【 0 0 8 8 】

ステップS12において上位タグが一致しないと判定された場合、処理はステップS14に進む。S14において、第1予測器10は、上位タグ格納部32が保持する有効度(飽和カウンタ値)は0であるか否かを判定する。なおこの例において飽和カウンタは2ビットであり、上限値が3(=11)であり、下限値が0(=00)である。有効度が0である場合、処理はステップS15に進む。

## 【 0 0 8 9 】

ステップS15において、第1予測器10は、対応フェッチアドレスの上位ビット(例えばbit[15:11])を上位タグ格納部32に第2のタグとして格納することにより、上位タグを更新する。第1予測器10は更に、アドレステーブル31の全フラグをリセットすると共に、分岐成立と予測された命令に関するエントリを、アドレステーブル31において対応フェッチアドレスが指し示すインデックス位置に作成する。具体的には、第1予測器10は、当該インデックス位置において対応フェッチアドレスのbit[10:7]を第1のタグとして格納し、予測されたターゲットアドレスを格納し、フラグを有効に設定する。第1予測器10は更に、上位タグ格納部32の有効度(飽和カウンタ値)を上限値3(=11)に変更する。

## 【 0 0 9 0 】

10

20

30

40

50

ステップ S 1 4 において有効度が 0 でないと判定された場合、処理はステップ S 1 6 に進む。ステップ S 1 6 において、第 1 予測器 1 0 は、新たなエントリを作成することなく、上位タグ格納部 3 2 に格納される有効度（飽和カウンタ値）を - 1 する。

【 0 0 9 1 】

図 1 0 は、分岐予測機構を備えたプロセッサを用いたシステム構成の一例を示す図である。図 1 0 に示されるシステムは、メモリ 6 1、CPU (Central Processing Unit) 6 2、メモリ 6 3、CPU 6 4、及びインターコネクト制御部 6 5 を含む。CPU 6 2 及び 6 4 は、例えば図 1 に示すようなプロセッサ構成を有し、上記説明した第 1 実施例又は第 2 実施例による分岐予測機構を内蔵する。インターコネクト制御部 6 5 は、外部装置との入出力制御を行う。図 1 0 に示されるような複数のプロセッサとメモリとを含む例えばサーバ装置等のシステムにおいて、上記説明した分岐予測機構を用いることができる。

10

【 0 0 9 2 】

以上、本発明を実施例に基づいて説明したが、本発明は上記実施例に限定されるものではなく、特許請求の範囲に記載の範囲内で様々な変形が可能である。

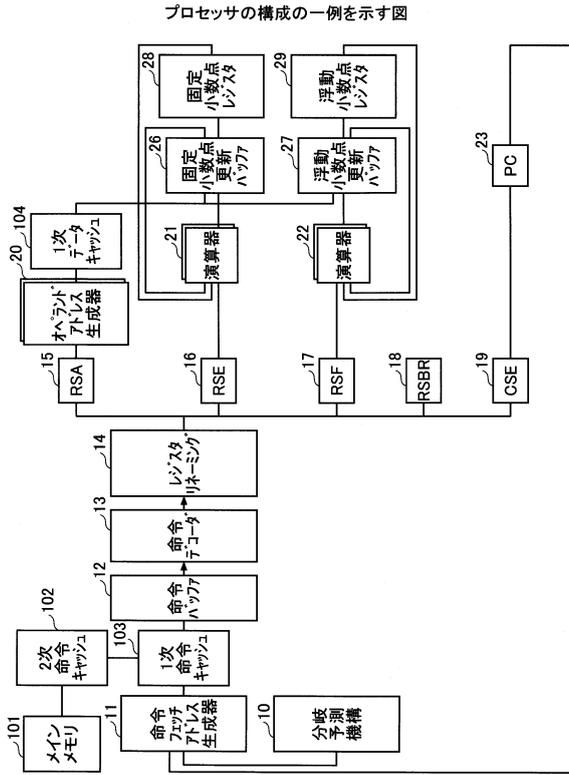
【 符号の説明 】

【 0 0 9 3 】

1 0	分岐予測機構	
1 1	命令フェッチアドレス生成器	
1 2	命令バッファ	
1 3	命令デコーダ	20
1 4	レジスタリネーミング	
1 5	R S A	
1 6	R S E	
1 7	R S F	
1 8	R S B R	
1 9	C S E	
2 0	オペランドアドレス生成器	
2 1	演算器	
2 2	演算器	
2 3	ネクストプログラムカウンタ	30
2 6	固定小数点更新バッファ	
2 7	浮動小数点更新バッファ	
2 8	固定小数点レジスタ	
2 9	浮動小数点レジスタ	
1 0 1	メインメモリ	
1 0 2	2 次命令キャッシュ	
1 0 3	1 次命令キャッシュ	
1 0 4	1 次データキャッシュ	
3 1	アドレステーブル	
3 2	上位タグ格納部	40
3 3	分岐方向判定部	
4 1	アドレステーブル	
4 2	バイモダル予測部	
4 3	予測生成部	
5 1	予測結果比較部	
5 4	セレクタ	

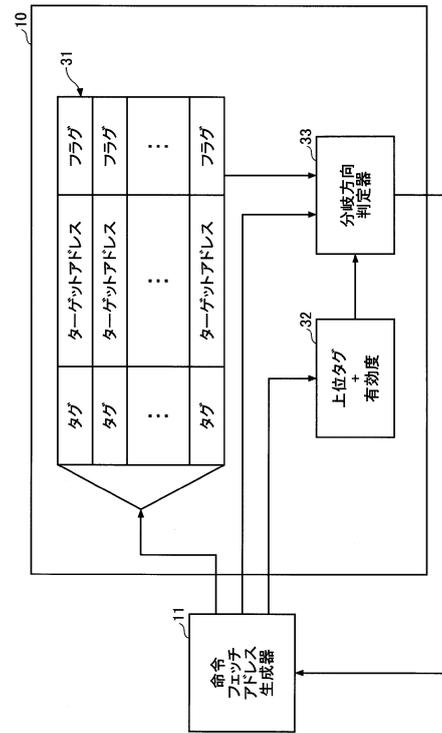
【 図 面 】

【 図 1 】



【 図 2 】

分岐予測機構の第1実施例の構成の一例を示す図

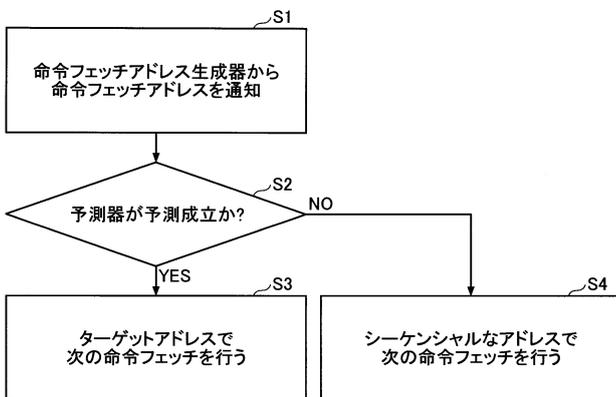


10

20

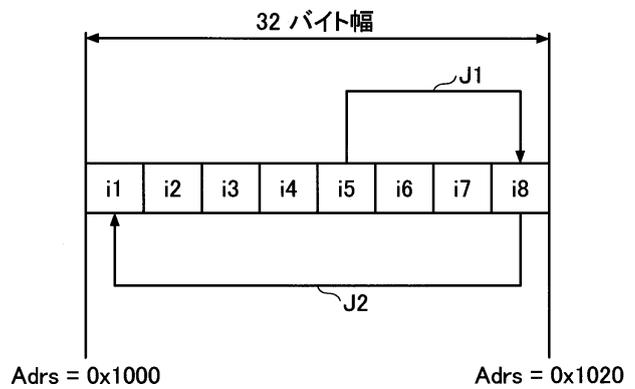
【 図 3 】

図2に示す分岐予測機構及び命令フェッチアドレス生成器の動作を示すフローチャート



【 図 4 】

フェッチラインの構成の一例を示す図



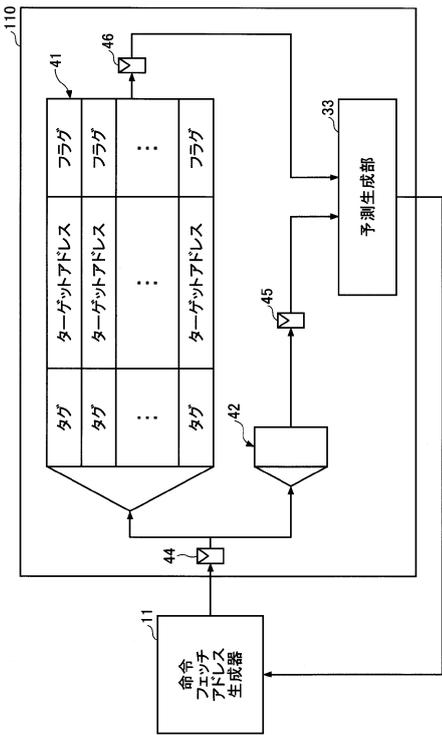
30

40

50

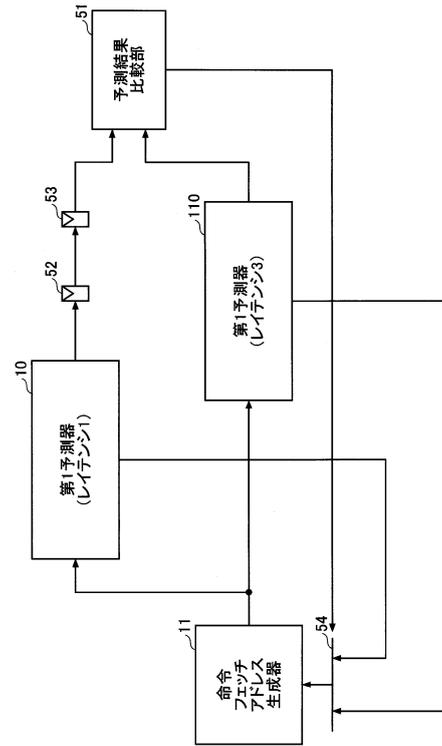
【 図 5 】

第2実施例に用いられる分岐予測器の構成の一例を示す図



【 図 6 】

分岐予測機構の第2実施例の構成の一例を示す図

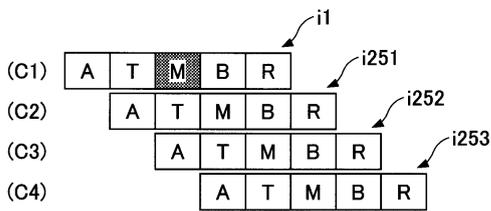


10

20

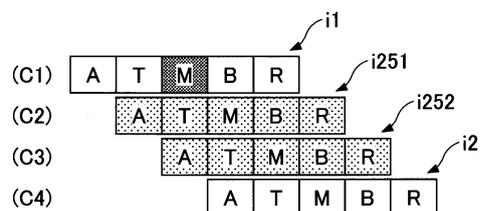
【 図 7 】

分岐予測機構の第2実施例における命令フェッチの一例を示す図



【 図 8 】

分岐予測機構の第2実施例における命令フェッチの一例を示す図



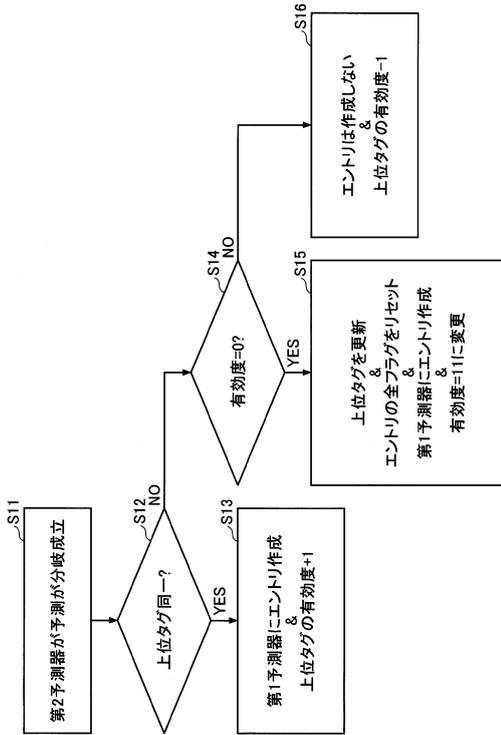
30

40

50

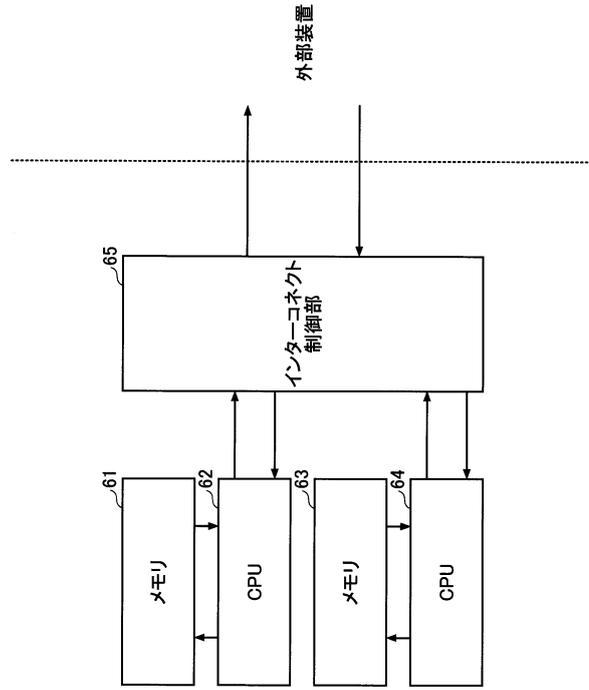
【 図 9 】

第1予測器におけるエントリ更新処理の一例を示すフローチャート



【 図 10 】

分岐予測機構を備えたプロセッサを用いたシステム構成の一例を示す図



10

20

30

40

50