

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3723019号  
(P3723019)

(45) 発行日 平成17年12月7日(2005.12.7)

(24) 登録日 平成17年9月22日(2005.9.22)

(51) Int. Cl.<sup>7</sup>

F I

G06F 9/38

G06F 9/38 330B

G06F 9/42

G06F 9/42 320A

請求項の数 13 (全 24 頁)

(21) 出願番号	特願平11-276221	(73) 特許権者	000005223
(22) 出願日	平成11年9月29日(1999.9.29)		富士通株式会社
(65) 公開番号	特開2001-100993(P2001-100993A)		神奈川県川崎市中原区上小田中4丁目1番1号
(43) 公開日	平成13年4月13日(2001.4.13)	(74) 代理人	100074099
審査請求日	平成15年11月13日(2003.11.13)		弁理士 大菅 義之
		(74) 代理人	100067987
			弁理士 久木元 彰
		(72) 発明者	砂山 竜一
			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
		(72) 発明者	鵜飼 昌樹
			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

最終頁に続く

(54) 【発明の名称】 サブルーチンリターン相当の命令の分岐予測を行う装置および方法

(57) 【特許請求の範囲】

【請求項1】

サブルーチンコール相当の命令が検出されたとき、該サブルーチンコール相当の命令により指定されたレジスタ手段のレジスタ番号を、サブルーチンの戻り先アドレスを指定する情報として格納する格納手段と、

サブルーチンリターン相当の命令になり得る命令が検出されたとき、該サブルーチンリターン相当の命令になり得る命令により指定されるレジスタ番号と、前記格納手段に格納された前記戻り先アドレスを指定する情報としてのレジスタ番号とを比較し、比較結果を出力する比較手段と、

前記比較結果に基づいて、前記サブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令を識別する識別手段とを備えることを特徴とする分岐予測装置。

【請求項2】

前記レジスタ手段は、サブルーチンコール相当命令の命令により、戻り先のアドレスを格納する特定のレジスタ手段であることを特徴とする請求項1記載の分岐予測装置。

【請求項3】

サブルーチンの戻り先アドレスを指定する情報を格納するスタック手段と、サブルーチンコール相当の命令が検出されたとき、前記戻り先アドレスを指定する情報を前記スタック手段にプッシュするプッシュ手段と、

サブルーチンリターン相当の命令になり得る命令が検出されたとき、該サブルーチンリ

10

20

ターン相当の命令になり得る命令の分岐先アドレスを指定する情報と、前記スタック手段の先頭エントリに格納された前記戻り先アドレスを指定する情報とを比較し、比較結果を出力する比較手段と、

前記比較結果に基づいて、前記サブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令を識別する識別手段とを備える

ことを特徴とする分岐予測装置。

【請求項 4】

前記プッシュ手段は、前記サブルーチンコール相当の命令により指定されたレジスタ手段のレジスタ番号を、前記戻り先アドレスを指定する情報として前記スタック手段にプッシュし、前記比較手段は、前記サブルーチンリターン相当の命令になり得る命令により指定された分岐先アドレスレジスタ手段のレジスタ番号と、前記スタック手段の先頭エントリに格納されたレジスタ番号とを比較し、前記識別手段は、比較された2つのレジスタ番号が一致したとき、該サブルーチンリターン相当の命令になり得る命令を、前記サブルーチンリターン相当の命令であると識別することを特徴とする請求項 3 記載の分岐予測装置

10

【請求項 5】

前記識別手段は、前記分岐先アドレスレジスタ手段のレジスタ番号が特定のレジスタ手段に対応するとき、前記比較結果にかかわらず、前記サブルーチンリターン相当の命令になり得る命令を、前記サブルーチンリターン相当の命令であると識別することを特徴とする請求項 4 記載の分岐予測装置。

20

【請求項 6】

前記プッシュ手段は、前記レジスタ手段のレジスタ番号が特定のレジスタ手段に対応するとき、該レジスタ手段のレジスタ番号を前記スタック手段にプッシュしないことを特徴とする請求項 4 記載の分岐予測装置。

【請求項 7】

前記識別手段が、前記サブルーチンリターン相当の命令になり得る命令を前記サブルーチンリターン相当の命令であると識別し、該サブルーチンリターン相当の命令による分岐が実行されたとき、前記スタック手段をポップするポップ手段をさらに備えることを特徴とする請求項 3 記載の分岐予測装置。

【請求項 8】

分岐予測のための分岐履歴情報を格納する分岐予測テーブル手段をさらに備え、前記比較手段は、該分岐履歴情報が該分岐予測テーブル手段に登録されるとき、前記サブルーチンリターン相当の命令になり得る命令が指定するレジスタ番号と、前記戻り先アドレスを指定する情報としてのレジスタ番号とを比較することを特徴とする請求項 1 記載の分岐予測装置。

30

【請求項 9】

サブルーチンコールとサブルーチンリターンの対応関係を崩す可能性がある事象が発生したとき、前記格納手段に格納された情報を無効化する手段をさらに備えることを特徴とする請求項 1 記載の分岐予測装置。

【請求項 10】

分岐予測のための分岐履歴情報を格納する分岐予測テーブル手段をさらに備え、前記サブルーチンコール相当の命令の直後に位置する命令アドレスに戻らないサブルーチンリターン相当の命令が検出されたとき、検出されたサブルーチンリターン相当の命令の戻り先が異なることを示すフラグを該分岐予測テーブル手段に設定する設定手段をさらに備えることを特徴とする請求項 1 記載の分岐予測装置。

40

【請求項 11】

前記分岐予測テーブル手段は、前記サブルーチンの戻り先アドレスを格納するリターンアドレススタック手段を含み、分岐予測時に前記フラグが認識された場合、該リターンアドレススタック手段をポップし、ポップされた戻り先アドレスを予測分岐先として用いないことを特徴とする請求項 10 記載の分岐予測装置。

50

**【請求項 1 2】**

分岐予測のための分岐履歴情報を格納する分岐予測テーブル手段と、該分岐予測テーブル手段に登録されていないサブルーチンコール相当の命令が検出された場合には、所定のフラグをセットし、該登録されていないサブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令が検出された場合には、該所定のフラグをリセットして、該分岐予測テーブル手段では該対応するサブルーチンリターン相当の命令をサブルーチンリターン相当の命令としては認識しない制御を行う手段とをさらに備えることを特徴とする請求項 1 記載の分岐予測装置。

**【請求項 1 3】**

サブルーチンコール相当の命令が検出されたとき、該サブルーチンコール相当の命令により指定されたレジスタ手段のレジスタ番号を、サブルーチンの戻り先アドレスを指定する情報として格納する格納手段と、

サブルーチンリターン相当の命令になり得る命令が検出されたとき、該サブルーチンリターン相当の命令になり得る命令がレジスタ番号により指定するレジスタ手段が保持するレジスタ値と、前記格納手段に格納された前記戻り先アドレスを指定する情報としてのレジスタ番号が指定するレジスタ手段が保持するレジスタ値とを比較し、比較結果を出力する比較手段と、

前記比較結果に基づいて、前記サブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令を識別する識別手段とを

備えることを特徴とする分岐予測装置。

**【発明の詳細な説明】****【0001】****【発明の属する技術分野】**

本発明は、分岐予測機構を有する情報処理装置に係り、特に、サブルーチンリターンのための特別な命令が用意されていないアーキテクチャにおいて、サブルーチンリターン相当の命令の分岐予測を行う分岐予測装置およびその方法に関する。

**【0002】****【従来の技術】**

従来の命令処理装置においては、パイプライン処理やアウトオブオーダー処理のような技術を用いて、ある命令の実行の完了を待たずに後続の命令の実行を次々と開始することで、性能の向上が図られてきた。

**【0003】**

パイプライン処理において、先行する命令が、分岐命令のように後続の実行シーケンスを変更する命令である場合、分岐が成立するときに、分岐先の命令を実行パイプラインに投入する必要がある。そうしなければ、実行パイプラインが乱され、最悪の場合は、逆に性能が落ちてしまう。

**【0004】**

そこで、ブランチヒストリ（分岐予測テーブル）に代表される分岐予測機構を設けて、分岐の成立・不成立を予測することで、性能の向上が図られてきた。このような装置において、分岐の成立が予測された場合には、分岐命令の後に分岐先の命令が実行パイプラインに投入されるため、分岐が実際に成立したときに実行パイプラインが乱されることがない。

**【0005】**

また、サブルーチンリターン命令の分岐先（戻り先）は、その命令の性質上、実行の度に変わることが多い。これは、サブルーチンの呼び出し元であるサブルーチンコール命令の場所が実行の度に異なるためである。このような命令に対しては、リターンアドレススタックと呼ばれる専用の分岐予測機構を設けることで、性能の向上が可能となることが知られている。

**【0006】**

**【発明が解決しようとする課題】**

10

20

30

40

50

しかしながら、上述した従来の分岐予測機構には、以下のような問題がある。

CPU（中央処理装置）アーキテクチャの中には、サブルーチンコール・リターン命令対として、あらかじめ特別な命令を準備していないものがある。このようなアーキテクチャにおいて、リターンアドレススタックを採用して性能向上を図るためには、実行される分岐命令の中から、サブルーチンコール・リターン相当の命令対を動的に抽出する技術が必要となる。

#### 【0007】

ところが、従来の情報処理装置では、デコード時にサブルーチンコール・リターン命令であるか否かを静的に決定していたため、ハードウェアによる解釈と異なるプログラミングは好ましいものではなかった。この場合、好ましくないプログラミングにより、コール・リターン対の対応関係が、一旦、実際とずれてしまうと、リターンアドレススタックの特性上、それ以降の分岐先を次々と誤ることになる。このため、リターンアドレススタックが多段であればあるほど、性能を落としていた。

10

#### 【0008】

図18は、このようなアーキテクチャにおいて使用される、サブルーチンコール・リターン命令対を含むプログラムの例を示している。

この例では、メインルーチンの中の命令“balr 14, 15”により、サブルーチンS1が呼び出され（Call 1）、さらにその中の命令“balr 15, 13”により別のサブルーチンS2が呼び出されている（Call 2）。そして、条件リターン命令“bcr 7, 15”により制御がサブルーチンS1に戻り（Return 2）、さらに無条件リターン命令“bcr 15, 14”によりメインルーチンに戻っている（Return 1）。

20

#### 【0009】

ここで、命令処理装置が、特定のオペレーションコード“balr”を、サブルーチンコール相当の命令として認識し、特定のオペレーションコードとオペランドを含む無条件分岐命令“bcr 15, x”（xは任意）を、サブルーチンリターン相当の命令として認識するものとする。

#### 【0010】

この場合、サブルーチンS2の中の命令“bcr 7, 15”は、サブルーチンリターン相当の命令として認識されずに、見落とされる。このため、従来のリターンアドレススタックでは、Call 2に対応するリターンがReturn 1であると認識してしまい、分岐予測に失敗する。実際には、Call 2に対応する正しいリターンは、Return 2である。

30

#### 【0011】

また、命令処理装置が、オペレーションコード“bcr”を含むすべての命令を単純にサブルーチンリターン相当の命令として認識すると、サブルーチンS2の中の単なる条件分岐命令である“bcr 4, 3”が、Call 2に対応するリターンであると認識されてしまう。したがって、この場合も、リターンアドレススタックがコール・リターン対を誤認識することが分かる。

#### 【0012】

このように、リターンアドレススタックを持つ情報処理装置においては、命令の実行時に、正しいサブルーチンコール・リターン命令対を認識することが重要になる。

40

#### 【0013】

本発明の課題は、サブルーチンリターンのための特別な命令が用意されていない情報処理装置において、サブルーチンリターン相当の命令を正しく認識する分岐予測装置およびその方法を提供することである。

#### 【0014】

##### 【課題を解決するための手段】

図1は、本発明の分岐予測装置の原理図である。本発明の第1の局面において、分岐予測装置は、格納手段1、比較手段2、および識別手段3を備える。

50

## 【 0 0 1 5 】

格納手段 1 は、サブルーチンコール相当の命令が検出されたとき、サブルーチンの戻り先アドレスを指定する情報を格納する。比較手段 2 は、サブルーチンリターン相当の命令になり得る命令が検出されたとき、そのサブルーチンリターン相当の命令になり得る命令の分岐先アドレスを指定する情報と、格納手段 1 に格納された戻り先アドレスを指定する情報とを比較し、比較結果を出力する。識別手段 3 は、その比較結果に基づいて、上記サブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令を識別する。

## 【 0 0 1 6 】

実行された（実行される）命令が、サブルーチンコール相当の動作をする命令であるとき、その命令により指定される戻り先アドレスや戻り先アドレスを保持するレジスタの情報等が、戻り先アドレスを指定する情報として格納手段 1 に格納される。

10

## 【 0 0 1 7 】

次に、実行された（実行される）命令が、サブルーチンリターン相当の動作をする命令である可能性があるとき、その命令により指定される分岐先アドレスや分岐先アドレスを保持するレジスタの情報等が、分岐先アドレスを指定する情報として選択され、比較手段 2 により、その情報が戻り先アドレスを指定する情報と比較される。

## 【 0 0 1 8 】

そして、識別手段 3 は、分岐先アドレスを指定する情報と戻り先アドレスを指定する情報が一致すれば、後者の命令を前者の命令に対応するサブルーチンリターン相当の命令であると識別し、それらが一致しなければ、後者の命令を前者の命令に対応するサブルーチンリターン相当の命令ではないと識別する。

20

## 【 0 0 1 9 】

このように、サブルーチンの戻り先アドレスを指定する情報を用いることで、正しいサブルーチンコール・リターン相当の命令対を動的に抽出することができる。したがって、コール・リターン対の対応関係が正しく認識され、その対応関係のズレが防止される。

## 【 0 0 2 0 】

また、本発明の第 2 の局面において、分岐予測装置は、スタック手段 4、プッシュ手段 5、比較手段 2、および識別手段 3 を備える。

スタック手段 4 は、サブルーチンの戻り先アドレスを指定する情報を格納する。プッシュ手段 5 は、サブルーチンコール相当の命令が検出されたとき、上記戻り先アドレスを指定する情報をスタック手段 4 にプッシュする。

30

## 【 0 0 2 1 】

比較手段 2 は、サブルーチンリターン相当の命令になり得る命令が検出されたとき、そのサブルーチンリターン相当の命令になり得る命令の分岐先アドレスを指定する情報と、スタック手段 4 の先頭エントリに格納された戻り先アドレスを指定する情報とを比較し、比較結果を出力する。識別手段 3 は、その比較結果に基づいて、上記サブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令を識別する。

## 【 0 0 2 2 】

サブルーチンコール相当の動作をする命令が検出されると、プッシュ手段 5 は、戻り先アドレスを指定する情報をスタック手段 4 にプッシュする。次に、サブルーチンリターン相当の動作をする命令である可能性がある命令が検出されると、比較手段 2 は、その命令の分岐先アドレスを指定する情報を、スタック手段 4 にプッシュされた戻り先アドレスを指定する情報と比較する。

40

## 【 0 0 2 3 】

そして、識別手段 3 は、分岐先アドレスを指定する情報と戻り先アドレスを指定する情報が一致すれば、後者の命令を前者の命令に対応するサブルーチンリターン相当の命令であると識別し、それらが一致しなければ、後者の命令を前者の命令に対応するサブルーチンリターン相当の命令ではないと識別する。

## 【 0 0 2 4 】

このように、サブルーチンの戻り先アドレスを指定する情報をスタック手段 4 にプッシュ

50

することで、第1の局面の分岐予測装置と同様に、コール・リターン対の対応関係が正しく認識され、その対応関係のズレが防止される。

【0025】

また、本発明の第3の局面において、分岐予測装置は、リターンアドレススタック手段6、比較手段2、および識別手段3を備える。

リターンアドレススタック手段6は、サブルーチンコール相当の命令が検出されたとき、サブルーチンの戻り先アドレスを格納する。比較手段2は、サブルーチンリターン相当の命令になり得る命令が検出されたとき、そのサブルーチンリターン相当の命令になり得る命令の分岐先アドレスと、リターンアドレススタック手段6に格納された戻り先アドレスとを比較し、比較結果を出力する。識別手段3は、その比較結果に基づいて、上記サブルーチンコール相当の命令に対応するサブルーチンリターン相当の命令を識別する。

10

【0026】

サブルーチンコール相当の動作をする命令が検出されると、その命令により指定された戻り先アドレスがリターンアドレススタック手段6にプッシュされる。次に、サブルーチンリターン相当の動作をする命令である可能性がある命令が検出されると、比較手段2は、その命令の分岐先アドレスを、リターンアドレススタック手段6にプッシュされた戻り先アドレスと比較する。

【0027】

そして、識別手段3は、分岐先アドレスと戻り先アドレスが一致すれば、後者の命令を前者の命令に対応するサブルーチンリターン相当の命令であると識別し、それらが一致しなければ、後者の命令を前者の命令に対応するサブルーチンリターン相当の命令ではないと識別する。

20

【0028】

このように、リターンアドレススタック手段6にプッシュされた戻り先アドレスを、直接、命令の分岐先アドレスと比較することで、第1の局面の分岐予測装置と同様に、コール・リターン対の対応関係が正しく認識され、その対応関係のズレが防止される。

【0029】

例えば、図1の格納手段1およびスタック手段4は、後述する図3のリンクスタック33、またはリターンアドレススタック35に対応する。また、例えば、図1の比較手段2と識別手段3は、後述する図11のEXNOR回路101、OR回路102、およびAND回路103、または後述する図17の比較回路151およびAND回路152に対応する。また、図1のプッシュ手段5は、後述する図12の制御回路に対応し、図1のリターンアドレススタック手段6は、図3のリターンアドレススタック35に対応する。

30

【0030】

【発明の実施の形態】

以下、図面を参照しながら、本発明の実施の形態を詳細に説明する。

命令処理装置においては、サブルーチンコール相当の命令により、戻り先のアドレスを格納するリンクレジスタが指定され、指定されたリンクレジスタを用いて、サブルーチンリターン相当の命令による分岐が行われる。

【0031】

サブルーチンコールまたはリターン相当の命令は、例えば、図2に示すように、オペレーションコード(OPコード)11、第1オペランド12、および第2オペランド13を含む。サブルーチンコール相当の命令においては、第1オペランド12が、リンクレジスタのレジスタ番号を表し、サブルーチンリターン相当の命令においては、第2オペランド13が、分岐先アドレスを格納するレジスタのレジスタ番号を表す。

40

【0032】

そこで、本実施形態においては、サブルーチンコール時に指定されたリンクレジスタのレジスタ番号を記録するリンクスタックを設ける。そして、リンクスタックに記録された番号のレジスタ内のアドレスを分岐先アドレスとして用いる分岐命令が出現したとき、その分岐命令をサブルーチンリターン相当の命令と認識する。

50

## 【0033】

このような制御によれば、リンクレジスタのレジスタ番号をリンク情報として用いて、サブルーチンコール相当の命令とサブルーチンリターン相当の命令を対応付けることができ、サブルーチンコール・リターン相当の命令対を動的に抽出することが可能になる。したがって、コール・リターン対の対応関係を正しく認識することができ、その対応関係のズレが防止されるので、リターンアドレススタックによる分岐予測の精度が向上する。

## 【0034】

例えば、図18の例の場合、コール命令に含まれるリンクレジスタの番号と、リターン命令に含まれる分岐先アドレスレジスタの番号を比較すれば、正しいコール・リターン対を認識することができ、分岐予測も成功する。

10

## 【0035】

Call 1の命令“balr 14, 15”の第1オペランドは、リンクレジスタの番号が“14”であることを表しており、Return 1の命令“bcr 15, 14”の第2オペランドは、分岐先アドレスレジスタの番号が“14”であることを表している。したがって、後者の命令は、Call 1に対応するリターン相当の命令であると認識される。

## 【0036】

また、Call 2の命令“balr 15, 13”の第1オペランドは、リンクレジスタの番号が“15”であることを表しており、Return 2の命令“bcr 7, 15”の第2オペランドは、分岐先アドレスレジスタの番号が“15”であることを表している。したがって、後者の命令は、Call 2に対応するリターン相当の命令であると認識される。

20

## 【0037】

次に、特別なサブルーチンコール・リターン命令対を準備していないアーキテクチャの一例を用いて、本実施形態における情報処理装置の動作をより詳細に説明する。このようなアーキテクチャは、例えば、ESA (enterprise systems architecture) / 390のPOO (principles of operation) により規定されている。

## 【0038】

サブルーチンコールとして使用可能な命令としては、サブルーチンリターン相当の命令による戻り先のアドレス(リンクアドレス)をレジスタに保存可能である命令が考えられる。このような命令には、bal、balr、bas、basr、basm等がある。

30

## 【0039】

また、サブルーチンリターンとして使用可能な命令としては、一般の分岐命令のほとんどが挙げられる。これらの中でも、特に、分岐先アドレスを1つのレジスタで指定する分岐命令、すなわち、RR形式命令が用いられやすい。RR形式命令には、bcr、basm等がある。もちろん、これらの命令は、通常の無条件分岐命令あるいは条件分岐命令としても用いられる。

## 【0040】

また、このようなアーキテクチャでは、出現頻度は低いものの、サブルーチンコール・リターン対を崩し得る命令も存在し得る。このような命令としては、例えば、lpswや、bc等のRX形式命令が挙げられる。また、一部の割り込み事象においても、サブルーチンコール・リターン対が崩されることがある。

40

## 【0041】

bcに代表されるRX形式の分岐命令は、リターン先アドレスを1つのレジスタのみで指定するとは限らず、特に、ディスプレイメントを指定する場合がある。他にも、リンクレジスタの値を書き換える等の処理により、戻り先アドレスが変更される場合もある。

## 【0042】

このような命令がサブルーチンリターンとして使用された場合、コール時にリターンアドレススタックに登録された戻り先アドレスが正しくならないので、リターン時にリターンアドレススタックを参照しないことが望ましい。その代わりに、通常に分岐命令と同じよ

50

うに、ブランチヒストリに登録されている予測分岐先を参照すれば、正しい戻り先アドレスが得られる可能性がある。

【0043】

また、lpswは、分岐先アドレスをレジスタで直接指定せず、オペランドが指すメモリ上のデータ列を分岐先アドレスとして用いる。このような命令列が出現した場合には、コール・リターン対の対応関係が崩れてしまう可能性がある。あるいはまた、割り込みが発生した場合にも、その割り込みの種類によっては、同様にコール・リターン対が崩れてしまうことがある。

【0044】

そこで、リターンアドレススタックに何らかの機構を組み込む必要がある。その一方法として、このような命令が実行されたときやこのような割り込みが発生したときに、リターンアドレススタックとリンクスタック双方のエントリをすべて消去してしまうことが考えられる。このような制御により、リターンアドレススタックの対応のズレが防止され、発生したズレにそれ以降の予測結果が引き摺られて性能が劣化することはなくなる。

10

【0045】

また、基本的な分岐命令がハードワイヤードで実現されているにもかかわらず、一部の分岐命令は、他の複雑な動作を伴うために、マイクロコードにより制御されている場合がある。このような複雑な分岐命令は、回路が複雑になるわりには得るものが少ない等の理由で、ブランチヒストリに登録しているメリットがない。このため、リターンアドレススタックも動作しない。

20

【0046】

しかしながら、前述したように、このような複雑な命令でも、サブルーチンコールもしくはリターン相当の命令になり得る場合には、その命令に対する対策を講じなければ、リターンアドレススタックの対応がずれてしまい、性能劣化の原因となる。

【0047】

そこで、ブランチヒストリに登録されていないサブルーチンコール相当の分岐命令が実行された後に、それに対応すると考えられるサブルーチンリターン相当の命令が検出されたとき、ブランチヒストリおよびリターンアドレススタックにおいて、そのサブルーチンリターン相当の命令をリターン相当の命令とは認識しないような制御を行う。

【0048】

また、プログラム指針書等により特定のレジスタをリンクレジスタとして使用することが推奨される場合等のように、特定のレジスタがリンクレジスタとして非常に多く使われることがある。このようなシステムでは、その特定のレジスタを用いる命令を、常に、サブルーチンコールまたはリターン相当の命令と認識することにする。これにより、リンクスタックのエントリを効率的に用いることができ、規模の小さなリンクスタックでも大きな効果が得られる。

30

【0049】

また、分岐命令において、分岐先アドレスのレジスタ番号として“0”を指定すると、分岐は行われない。このようなアーキテクチャでは、レジスタ番号“0”をリンク情報として、対応するサブルーチンリターン相当の命令を判別することは不可能である。そこで、サブルーチンコール相当の命令において、リンクレジスタの番号として“0”が指定されたときには、この命令をサブルーチンコール相当の命令として認識しないようにする。

40

【0050】

図3は、本実施形態における命令処理装置の構成図である。図3の命令処理装置は、命令フェッチ回路21、分岐予測機構22、デコーダ23、分岐先アドレス生成回路24、分岐命令実行処理回路25、および命令実行完了処理回路26を備え、命令をアウトオブオーダー方式で処理する。アウトオブオーダー処理方式を採用した命令処理装置においては、性能の向上を図るため、1つの命令実行の完了を待たずに、後続の命令列が順次複数のパイプラインに投入される。

【0051】

50

命令フェッチ回路 2 1 と分岐予測機構 2 2 は、命令フェッチパイプラインの回路に対応する。分岐予測機構 2 2 は、予測回路 3 1、比較回路 3 2、およびリンクスタック 3 3 を含み、予測回路 3 1 は、ブランチヒストリ 3 4 とリターンアドレススタック 3 5 を含む。

【 0 0 5 2 】

また、デコーダ 2 3、分岐先アドレス生成回路 2 4、分岐命令実行処理回路 2 5、および命令実行完了処理回路 2 6 は、命令実行パイプラインの回路に対応する。分岐命令実行処理回路 2 5 は、複数の R S B R (Reservation Station for BRanch) 3 6 を含む。

【 0 0 5 3 】

命令フェッチパイプラインは、命令アドレス発行サイクル ( I A )、テーブルサイクル ( I T )、バッファサイクル ( I B )、および結果サイクル ( I R ) を含み、命令実行パイプラインは、デコードサイクル ( D )、アドレス計算サイクル ( A )、実行サイクル ( X )、更新サイクル ( U )、およびライトサイクル ( W ) を含む。

10

【 0 0 5 4 】

R S B R 3 6 は、分岐命令を制御するための処理待ちのスタックである。分岐命令実行処理回路 2 5 は、アウトオブオーダー処理方式に従って、スタック内の処理可能なエントリを選択し、プログラムにより指示された順序とは異なる順序で、随時、分岐命令を実行することができる。

【 0 0 5 5 】

R S B R 3 6 で扱われる分岐命令のうち、b a l、b a l r (ただし、b a l r 1, 1 4 を除く)、b r a s、b a s、および b a s r は、サブルーチンコール相当の命令として扱われ、b c r、b s m、b a l r 1, 1 4 は、サブルーチンリターン相当の命令として扱われる。また、b a s s m は、サブルーチンコール相当の命令ではあるが、R S B R 3 6 では扱われない複雑命令である。

20

【 0 0 5 6 】

分岐命令実行処理回路 2 5 が分岐命令を実行した結果、分岐が生じることが判明した場合、分岐先の命令アドレスと分岐命令自身の命令アドレスが、対にしてブランチヒストリ 3 4 に登録される。そして、命令フェッチ回路 2 1 は、分岐命令をフェッチしたとき、次の命令のフェッチに先立ってブランチヒストリ 3 4 を検索し、分岐先を予測する。

【 0 0 5 7 】

また、デコーダ 2 3 がサブルーチンコール相当の命令を検出したとき、その命令のオペランドが表すリンクレジスタの番号がリンクスタック 3 3 にプッシュされ、対応する戻り先の命令アドレスがリターンアドレススタック 3 5 にプッシュされる。

30

【 0 0 5 8 】

また、デコーダ 2 3 がサブルーチンリターン相当の命令になり得る命令を検出すると、比較回路 3 2 は、リンクスタック 3 3 の先頭エントリに登録されているレジスタ番号と、検出された命令のオペランドが表す分岐先アドレスレジスタの番号を比較する。そして、2 つのレジスタ番号が一致すれば、その命令がサブルーチンリターン相当の動作をする命令であると判断し、比較結果を予測回路 3 1 に出力する。

【 0 0 5 9 】

このとき、リンクスタック 3 3 からレジスタ番号がポップされ、リターンアドレススタック 3 5 から対応する命令アドレスがポップされる。そして、ポップされた命令アドレスが、予測分岐先として命令フェッチ回路 2 1 に渡される。

40

【 0 0 6 0 】

リンクスタック 3 3 のエントリは、図 4 に示すように、リターンアドレススタック 3 5 のエントリと 1 対 1 に対応しており、これらの 2 つのスタックは、同時にプッシュ動作およびポップ動作を行う。ここでは、リンクスタック 3 3 のエントリには、4 ビットのレジスタ番号 0 : 3 が格納されており、リターンアドレススタック 3 5 のエントリには、3 2 ビットの分岐先アドレス 0 : 3 1 が格納されている。これらのスタックは、一般に、n 個 ( n - 1 ) のエントリからなる n 段のスタックとして実装される。

【 0 0 6 1 】

50

図5は、図3の命令処理装置において用いられる信号を示している。デコーダ23は、信号+D\_\_BALR、+D\_\_BAL、+D\_\_BRAS、+D\_\_BAS、+D\_\_BASR、+D\_\_BALR\_\_1E、+D\_\_BCR、+D\_\_BSM、+D\_\_BASSM、および+D\_\_OPC 8:15 を分岐命令実行処理回路25に出力する。

【0062】

信号+D\_\_BALR、+D\_\_BAL、+D\_\_BRAS、+D\_\_BAS、+D\_\_BASR、+D\_\_BALR\_\_1E、+D\_\_BCR、+D\_\_BSM、および+D\_\_BASSMは、それぞれ、balr、bal、bras、bas、basr、balr 1, 14、bcr、bsm、およびbassmが検出されたときに、論理“1”となる。また、信号+D\_\_OPC 8:15 は、機械語命令のビット 8:15 のデータを表す。

10

【0063】

分岐命令実行処理回路25は、信号+BRHIS\_\_UPDATE\_\_SUBROUTINE\_\_CALL、+BRHIS\_\_UPDATE\_\_SUBROUTINE\_\_RTN、+BRHIS\_\_UPDATE\_\_CALL\_\_RTN\_\_REG 0:3、+BRHIS\_\_UPDATE\_\_BSM、および+D\_\_BASSMを分岐予測機構22に出力する。

【0064】

信号+BRHIS\_\_UPDATE\_\_SUBROUTINE\_\_CALLは、命令がサブルーチンコール相当の命令と判別されたときに論理“1”となり、信号+BRHIS\_\_UPDATE\_\_SUBROUTINE\_\_RTNは、命令がサブルーチンリターン相当の命令になり得る命令と判別されたときに論理“1”となる。また、信号+BRHIS\_\_UPDAT 20  
E\_\_CALL\_\_RTN\_\_REG 0:3 は、命令のオペランドにより指定されたレジスタ番号を表し、信号+BRHIS\_\_UPDATE\_\_BSMは、bsm命令の実行が完了したときに論理“1”となる。

【0065】

次に、図6から図17までを参照しながら、図3の命令処理装置の構成と動作についてより詳細に説明する。

まず、デコーダ23による命令デコード時に、図5に示した信号がRSBR36に入力され、サブルーチンコール相当の命令とサブルーチンリターン相当の命令になり得る命令が判別される。このうち、サブルーチンリターン相当の命令になり得る命令については、後述するリンクスタック33の回路により、サブルーチンリターンとのより厳密な対応関係 30  
が識別されることになる。

【0066】

図6は、RSBR36内の判別回路を示している。図6において、入力信号-D\_\_BALR\_\_1Eは、図5の信号+D\_\_BALR\_\_1Eの否定を表し、命令“balr 1, 14”がデコードされたときに論理“0”となる。AND回路41は、入力信号+D\_\_BALRと-D\_\_BALR\_\_1Eの論理積を、OR回路42に出力する。したがって、“balr 1, 14”以外のbalr命令がデコードされたとき、AND回路41の出力は論理“1”となる。

【0067】

OR回路42は、AND回路41からの出力信号と、入力信号+D\_\_BAL、+D\_\_BRAS、+D\_\_BASR、および+D\_\_BASの論理和を、信号+D\_\_SUBROUTINE\_\_CALLとして出力する。この信号+D\_\_SUBROUTINE\_\_CALLは、デコードされた命令がサブルーチンコール相当の命令であるときに論理“1”となるフラグとして用いられる。

40

【0068】

また、OR回路43は、入力信号+D\_\_BALR\_\_1E、+D\_\_BCR、および+D\_\_BSMの論理和を、信号+D\_\_SUBROUTINE\_\_RETURNとして出力する。この信号+D\_\_SUBROUTINE\_\_RETURNは、デコードされた命令がサブルーチンリターン相当の命令になり得る命令であるときに論理“1”となるフラグとして用いられる。

50

## 【 0 0 6 9 】

デコードされた命令が分岐命令であれば、通常は、デコード結果が R S B R 3 6 に登録される。この時点で、R S B R 3 6 には、サブルーチンコール・リターン判別結果を表すフラグと、リンクレジスタや分岐先アドレスレジスタの情報等が登録される。

## 【 0 0 7 0 】

E S A / 3 9 0 P O O のアーキテクチャでは、サブルーチンコール相当の命令になり得る命令（機械語命令）のビット 8 : 1 1 にリンクレジスタの番号が指定され、サブルーチンリターン相当の命令になり得る命令（機械語命令）のビット 1 2 : 1 5 に分岐アドレスレジスタの番号が指定される。そこで、ビット 8 : 1 5 のデータがこれらのレジスタの情報として登録される。

10

## 【 0 0 7 1 】

図 7 は、R S B R 3 6 内の登録回路を示している。図 7 において、入力信号 + R S B R \_ V A L I D は、対応する R S B R 3 6 が有効である間、論理“ 1 ”となる。ラッチ回路 5 1 は、入力信号 + D \_ O P C 8 : 1 5 の値をラッチし、信号 + R S B R \_ O P C 8 : 1 5 として出力する。

## 【 0 0 7 2 】

また、ラッチ回路 5 2 は、図 6 の判別回路が生成したフラグ + D \_ S U B R O U T I N E \_ C A L L と + D \_ S U B R O U T I N E \_ R E T U R N の値をラッチし、それぞれ、信号 + R S B R \_ S U B R O U T I N E \_ C A L L と + R S B R \_ S U B R O U T I N E \_ R E T U R N として出力する。

20

## 【 0 0 7 3 】

信号 + R S B R \_ V A L I D が論理“ 1 ”となると、これらの情報の登録が終了し、対応する R S B R 3 6 が有効である間、ラッチ回路 5 1、5 2 に登録された情報が保持される。

## 【 0 0 7 4 】

次に、分岐履歴情報の更新時において、R S B R 3 6 に登録されたサブルーチンコール・リターン判別結果とレジスタ情報が、その他の分岐履歴情報と同時に、分岐予測機構 2 2 へ送出される。サブルーチンコール相当の命令である場合は、リンクレジスタの番号がレジスタ情報として選択され、サブルーチンリターン相当の命令になり得る命令である場合は、分岐先アドレスレジスタの番号がレジスタ情報として選択される。

30

## 【 0 0 7 5 】

図 8 は、R S B R 3 6 内の選択回路を示している。図 8 において、AND 回路 6 1 は、図 7 の登録回路からの信号 + R S B R \_ S U B R O U T I N E \_ C A L L と + R S B R \_ O P C 8 : 1 1 の論理積を、OR 回路 6 3 に出力する。したがって、フラグ + R S B R \_ S U B R O U T I N E \_ C A L L が立っているときは、リンクレジスタの番号が AND 回路 6 1 から出力される。

## 【 0 0 7 6 】

また、AND 回路 6 2 は、図 7 の登録回路からの信号 + R S B R \_ S U B R O U T I N E \_ R E T U R N と + R S B R \_ O P C 1 2 : 1 5 の論理積を、OR 回路 6 3 に出力する。したがって、フラグ + R S B R \_ S U B R O U T I N E \_ R E T U R N が立っているときは、分岐先アドレスレジスタの番号が AND 回路 6 2 から出力される。

40

## 【 0 0 7 7 】

そして、OR 回路 6 3 は、AND 回路 6 1、6 2 の出力信号の論理和を、信号 + R S B R \_ C A L L \_ R E T U R N \_ R E G 0 : 3 として出力する。ここで、フラグ + R S B R \_ S U B R O U T I N E \_ C A L L と + R S B R \_ S U B R O U T I N E \_ R E T U R N が同時に立つことはないので、OR 回路 6 3 は、AND 回路 6 1、6 2 の出力信号を選択的に出力することになる。

## 【 0 0 7 8 】

信号 + R S B R \_ S U B R O U T I N E \_ C A L L、+ R S B R \_ S U B R O U T I N E \_ R E T U R N、および + R S B R \_ C A L L \_ R E T U R N \_ R E G 0 : 3 は、ぞ

50

れぞれ、図5の信号BRHIS\_\_UPDATE\_\_SUBROUTINE\_\_CALL、BRHIS\_\_UPDATE\_\_SUBROUTINE\_\_RTN、+BRHIS\_\_UPDATE\_\_CALL\_\_RTN\_\_REG 0:3として、分岐予測機構22へ送出される。

【0079】

ところで、前述したように、分岐命令(サブルーチンリターン相当の命令を含む)において、分岐先アドレスレジスタの番号として“0”を指定すると、分岐は行われない。また、逆に、デコード時にサブルーチンコール相当の命令と判別された命令でも、リンクレジスタの番号として“0”が指定された場合には、この命令をサブルーチンコール相当の命令として識別しない方がよい。

【0080】

そこで、図9のような識別回路を分岐予測機構22に設けて、送られてきたレジスタ番号が“0”でないときにのみ有効となる制御信号を生成する。図9において、NAND回路71は、信号+BRHIS\_\_UPDATE\_\_CALL\_\_RTN\_\_REG 0:3の4つのビットの否定の論理積を求め、その論理積の否定を信号+SBR TN\_\_LINK\_\_REG\_\_VALとして出力する。

【0081】

したがって、この出力信号は、信号+BRHIS\_\_UPDATE\_\_CALL\_\_RTN\_\_REG 0:3が表すレジスタ番号が“0”でないときにのみ、論理“1”となり、リンクレジスタが有効であることを表す。この信号を用いたリンクスタック33の制御については、後述することにする。

【0082】

分岐命令が分岐しないことを表す分岐先アドレスレジスタの番号として、“0”以外の特定の番号が用いられた場合も、図9と同様の回路により同様の制御信号が生成される。

【0083】

また、サブルーチンコールとして使用可能なb a s s m命令については、ハードワイヤードではなく、マイクロコードにより実現されるため、ブランチヒストリ34には登録されず、分岐履歴情報の更新時に情報が送られてこない。その代わりに、デコード時に生成された図5の信号+D\_\_BASSMが分岐予測機構22に送出される。

【0084】

そこで、図10のような識別回路を分岐予測機構22に設けて、b a s s m命令に関する制御を行う。ここでは、b a s s m命令に対応するリターン命令はb s mのみであるものと仮定している。

【0085】

図10において、AND回路81は、ラッチ回路83の出力とNAND回路84の出力の論理積をOR回路82に出力し、OR回路82は、入力信号+D\_\_BASSMとAND回路81の出力信号の論理和をラッチ回路83に出力する。ラッチ回路83は、実質的にS e t - R e s e tフリップフロップの動作を行い、OR回路82の出力信号をラッチして、NAND回路84に出力する。

【0086】

また、NAND回路84は、図5の信号+BRHIS\_\_UPDATE\_\_BSSMと、図9の制御信号+SBR TN\_\_LINK\_\_REG\_\_VALと、ラッチ回路83の出力信号の論理積の否定を、信号-SBR TN\_\_BASSM\_\_BSSM\_\_RTN\_\_VALIDとして出力する。この信号-SBR TN\_\_BASSM\_\_BSSM\_\_RTN\_\_VALIDは、論理“0”のとき、実行されたb s m命令が上記b a s s m命令に対応するリターン命令であることを表す。

【0087】

このような識別回路によれば、分岐するb a s s m命令が実行された場合には、信号+D\_\_BASSMが論理“1”となり、ラッチ回路83の出力も論理“1”となる。そして、ラッチ回路83の出力と図9の信号+SBR TN\_\_LINK\_\_REG\_\_VALが論理“1”である間に、b s m命令の実行が完了して信号+BRHIS\_\_UPDATE\_\_BSSMが

10

20

30

40

50

論理“1”となると、実行された b s m 命令は上記 b a s s m 命令に対応するリターン命令であると識別される。

【0088】

このとき、信号 - S B R T N \_ B A S S M \_ B S M \_ R T N \_ V A L I D が論理“0”となるため、AND回路81の出力も論理“0”となる。また、信号 + D \_ B A S S M も論理“0”であるため、ラッチ回路83の出力も論理“0”となる。

【0089】

このように、ラッチ回路83の出力信号は、b a s s m 命令および b s m 命令が検出されたことを表す所定のフラグとして用いられ、分岐する b a s s m 命令が検出されたときにセットされ、それに対応する b s m 命令が検出されたときにリセットされる。

10

【0090】

また、信号 - S B R T N \_ B A S S M \_ B S M \_ R T N \_ V A L I D と同時に、不図示の信号 + S B R T N \_ B A S S M \_ B S M \_ R T N \_ V A L I D も生成される。この信号 + S B R T N \_ B A S S M \_ B S M \_ R T N \_ V A L I D は、信号 - S B R T N \_ B A S S M \_ B S M \_ R T N \_ V A L I D の否定に対応し、論理“1”のとき、実行された b s m 命令が上記 b a s s m 命令に対応するリターン命令であることを表す。

【0091】

こうして識別された b a s s m 命令に対応する b s m 命令は、ブランチヒストリ34あるいはリターンアドレススタック35上では、リターン相当の命令とは認識されなくなる。なぜなら、b a s s m 命令自身がコール相当の命令として登録されていないからである。

20

【0092】

分岐予測機構22は、分岐命令実行処理回路25から送られてきた信号と、図9および図10の識別回路により生成された特別な制御信号とを用いて、サブルーチンコール・リターン相当の命令を判別する。

【0093】

図11は、分岐予測機構22内の判別回路を示している。図11において、入力信号 - B R H I S \_ U P D A T E \_ S U B R O U T I N E \_ R T N は、図5の信号 + B R H I S \_ U P D A T E \_ S U B R O U T I N E \_ R T N の否定に対応する。

【0094】

また、入力信号 + R T N \_ L I N K \_ R E G \_ S T K 0 0 : 3 は、リンクスタック33の先頭エントリに保持されたレジスタ番号を表す。入力信号 + S B R T N \_ L I N K \_ R E G \_ E Q \_ E は、図5の信号 + B R H I S \_ U P D A T E \_ C A L L \_ R T N \_ R E G 0 : 3 がレジスタ番号“14”を表すとき、論理“1”となり、それ以外の番号を表すとき、論理“0”となる。

30

【0095】

AND回路91は、図5の信号 + B R H I S \_ U P D A T E \_ S U B R O U T I N E \_ C A L L と図9の信号 + S B R T N \_ L I N K \_ R E G \_ V A L の論理積を、AND回路92に出力する。AND回路92は、AND回路91の出力信号と信号 - B R H I S \_ U P D A T E \_ S U B R O U T I N E \_ R T N の論理積を、信号 + B R \_ C O M P \_ S U B R O U T I N E \_ C A L L として出力する。

40

【0096】

この信号 + B R \_ C O M P \_ S U B R O U T I N E \_ C A L L は、分岐予測機構22において、サブルーチンコール相当の命令を表すフラグ(サブルーチンコールフラグ)として用いられる。このフラグが論理“1”のとき、分岐命令実行処理回路25で実行された命令はサブルーチンコール相当の命令であると判別される。実行された命令がリンクレジスタとして番号“0”のレジスタを指定していた場合は、このフラグが論理“0”となり、サブルーチンコール相当の命令ではないと判別される。

【0097】

また、EXNOR回路101は、図5の信号 + B R H I S \_ U P D A T E \_ C A L L \_ R T N \_ R E G 0 : 3 と、信号 + R T N \_ L I N K \_ R E G \_ S T K 0 0 : 3 を比

50

較し、2つの信号の排他的論理和の否定を出力する。OR回路102は、EXNOR回路101の出力信号と信号+SBRTN\_LINK\_REQ\_EQ\_Eの論理和を出力する。

【0098】

そして、AND回路103は、図5の信号+BRHIS\_UPDATE\_SUBROUTINE\_RTNと、図9の信号+SBRTN\_LINK\_REQ\_VALと、図10の信号-SBRTN\_BASSM\_BSM\_RTN\_VALIDと、OR回路102の出力信号の論理積を、信号+BR\_COMP\_SUBROUTINE\_RTNとして出力する。

【0099】

この信号+BR\_COMP\_SUBROUTINE\_RTNは、分岐予測機構22において、サブルーチンリターン相当の命令を表すフラグ(サブルーチンリターンフラグ)として用いられる。このフラグが論理“1”のとき、分岐命令実行処理回路25で実行された命令はサブルーチンリターン相当の命令であると判別される。この判別動作は、対応する分岐履歴情報がブランチヒストリ34やリターンアドレススタック35に登録される前に行われる。

10

【0100】

EXNOR回路101、OR回路102、およびAND回路103からなるサブルーチンリターン判別回路は、図3の比較回路32に対応する。この判別回路によれば、EXNOR回路101により、実行されたサブルーチンリターン相当の命令になり得る命令の分岐先アドレスレジスタの番号がリンクスタック33の先頭エントリと比較され、それらが一致したとき、実行された命令がサブルーチンリターン相当の命令であると判別される。

20

【0101】

ただし、前述したように、分岐予測機構22では、bassm命令に対応するbsm命令をリターン相当の命令として扱わないので、信号-SBRTN\_BASSM\_BSM\_RTN\_VALIDにより、AND回路103の出力が抑制されている。

【0102】

また、慣習的に、番号“14”のレジスタは、サブルーチンリターンにおける分岐先アドレスレジスタとして用いられることが多い。そこで、このレジスタが分岐先アドレスレジスタとして指定された場合は、信号+SBRTN\_LINK\_REQ\_EQ\_EQ\_Eを用いて、EXNOR回路101の比較結果にかかわらず、実行された命令をサブルーチンリターン相当の命令とみなしている。

30

【0103】

サブルーチンリターン相当の命令であることを表す分岐先アドレスレジスタの番号として、“14”以外の特定の番号が用いられた場合も、図11と同様の回路により同様の制御が行われる。

【0104】

こうして生成されたサブルーチンコールフラグとサブルーチンリターンフラグを用いて、リンクスタック33は、図12のような制御回路によりプッシュ動作とポップ動作を行う。ここでは、リンクスタック33は2つのエントリからなるものとし、入力信号+RTN\_LINK\_REQ\_STK0\_0:3と+RTN\_LINK\_REQ\_STK1\_0:3は、それぞれ、先頭エントリ(エントリ0)と2番目のエントリ(エントリ1)に保持されたレジスタ番号を表している。

40

【0105】

また、入力信号-SBRTN\_LINK\_REQ\_EQ\_EQ\_Eは、図11の信号+SBRTN\_LINK\_REQ\_EQ\_EQ\_Eの否定に対応し、入力信号BRHIS\_UPDATE\_TAKENは、分岐命令による分岐が実行され、分岐履歴情報が更新されるときに、論理“1”となる。

【0106】

まず、AND回路111は、これらの2つの信号の論理積を出力する。AND回路112は、図11のフラグ+BR\_COMP\_SUBROUTINE\_CALLとAND回路1

50

11の出力信号の論理積を、動作信号+PUSH\_\_RTN\_\_STACK\_\_LINK\_\_REGとして出力する。この信号は、リンクスタック33およびリターンアドレススタック35のプッシュ動作の指示に用いられ、サブルーチンコール相当の命令が実行されて分岐履歴情報が更新されるときに、論理“1”となる。

【0107】

また、AND回路113は、図11のフラグ+BR\_\_COMP\_\_SUBROUTINE\_\_RTNとAND回路111の出力信号の論理積を、動作信号+POP\_\_RTN\_\_STACK\_\_LINK\_\_REGとして出力する。この信号は、リンクスタック33およびリターンアドレススタック35のポップ動作の指示に用いられ、サブルーチンリターン相当の命令が実行されて分岐履歴情報が更新されるときに、論理“1”となる。

10

【0108】

ここで、リンクレジスタの番号として“14”を指定するサブルーチンコール相当の命令と、分岐先アドレスレジスタの番号として“14”を指定するサブルーチンリターン相当の命令は、常にコール・リターン命令対を形成するものとする、リンクスタック33を用いなくてもこれらの対応関係を抽出することができる。

【0109】

そこで、このような場合にリンクスタック33を動作させないように、信号-SBRTN\_\_LINK\_\_REQ\_\_EQ\_\_Eを用いて、プッシュ動作およびポップ動作の動作信号を抑制している。これにより、リンクスタック33のエントリの浪費が防止され、少ない段数でも効率的な動作が可能になる。

20

【0110】

次に、AND回路114は、図5の信号+BRHIS\_\_UPDATE\_\_CALL\_\_RTN\_\_REG 0:3と動作信号+PUSH\_\_RTN\_\_STACK\_\_LINK\_\_REGの論理積を出力し、AND回路115は、信号+RTN\_\_LINK\_\_REG\_\_STK1 0:3と動作信号+POP\_\_RTN\_\_STACK\_\_LINK\_\_REGの論理積を出力する。

【0111】

また、OR回路116は、AND回路114、115の出力信号の論理和を、信号+SET\_\_RTN\_\_LINK\_\_REG\_\_STK0 0:3として出力する。この信号は、リンクスタック33の先頭エントリにセットされるレジスタ番号を表す。

【0112】

ここで、動作信号+PUSH\_\_RTN\_\_STACK\_\_LINK\_\_REGと+POP\_\_RTN\_\_STACK\_\_LINK\_\_REGが同時に論理“1”になることはないので、OR回路116は、AND回路114、115の出力信号を選択的に出力することになる。したがって、プッシュ動作の場合は、サブルーチンコール相当の命令により指定されたリンクレジスタの番号がセットされ、ポップ動作の場合は、リンクスタック33の2番目のエントリに保持されたレジスタ番号がセットされる。

30

【0113】

また、AND回路117は、信号+RTN\_\_LINK\_\_REG\_\_STK0 0:3と動作信号+PUSH\_\_RTN\_\_STACK\_\_LINK\_\_REGの論理積を、信号+SET\_\_RTN\_\_LINK\_\_REG\_\_STK1 0:3として出力する。この信号は、リンクスタック33の2番目のエントリにセットされるレジスタ番号を表し、プッシュ動作の場合は、リンクスタック33の先頭エントリに保持されたレジスタ番号に一致する。

40

【0114】

図13は、リンクスタック33内でレジスタ番号を保持するラッチ回路を示している。図13において、入力信号-PUSH\_\_POP\_\_RTN\_\_LINK\_\_REG\_\_STKは、プッシュ動作またはポップ動作が終了したときに論理“1”となる。

【0115】

ラッチ回路121は、図12の信号+SET\_\_RTN\_\_LINK\_\_REG\_\_STK0 0:3を先頭エントリとしてラッチし、それを図12の信号+RTN\_\_LINK\_\_REG\_\_STK0 0:3として出力する。また、ラッチ回路122は、図12の信号+SE

50

T\_RTN\_LINK\_REG\_STK1 0:3 を2番目のエントリとしてラッチし、それを図12の信号+RTN\_LINK\_REG\_STK1 0:3 として出力する。

【0116】

信号-PUSH\_POP\_RTN\_LINK\_REG\_STKが論理“1”となると、これらのエントリへのレジスタ番号の登録が終了し、この信号が論理“0”になるまで、登録されたレジスタ番号が保持される。

【0117】

ところで、前述したlpsw命令(複雑命令)は、サブルーチンコール・リターン命令のいずれにもなり得るため、コール・リターンの対応関係を崩すものと考えられる。あるいはまた、割り込みが発生した場合、それが割り込み処理の後に元のプログラムに戻らない種類の割り込みであれば、やはり、コール・リターンの対応関係を崩すものと考えられる。

10

【0118】

そこで、このような事象(命令や割り込み等)が発生した場合には、その命令や割り込みの実行時に、リンクスタック33とリターンアドレススタック35のエントリをすべてクリアして、保持された情報を無効化する。

【0119】

図14は、分岐予測機構22内の無効化回路を示している。図14において、入力信号+MICRO\_PURGE\_RTN\_ADRS\_STKは、リンクスタック33とリターンアドレススタック35のエントリをクリアする信号であり、コール・リターンの対応関係を崩すような命令や割り込みが発生したときに、論理“1”となる。

20

【0120】

NOR回路131は、図12の動作信号+PUSH\_RTN\_STACK\_LINK\_REGと+POP\_RTN\_STACK\_LINK\_REG、および信号+MICRO\_PURGE\_RTN\_ADRS\_STKの論理和の否定を、図13の信号-PUSH\_POP\_RTN\_LINK\_REG\_STKとして出力する。

【0121】

したがって、信号+MICRO\_PURGE\_RTN\_ADRS\_STKが論理“1”となると、信号-PUSH\_POP\_RTN\_LINK\_REG\_STKが論理“0”となり、図13のラッチ回路121、122に保持されたレジスタ番号がクリアされる。

30

【0122】

また、サブルーチンコールに対応する戻り先、すなわち、サブルーチンコール相当の命令の直後の命令アドレスに戻らないサブルーチンリターン相当の命令を認識した場合、ランヒストリ34に、そのサブルーチンリターン相当の命令の戻り先が異なることを示すフラグを設定することができる。

【0123】

図15は、RSBR36において、このようなフラグを生成する回路を示している。図15において、入力信号+D\_BCは、デコーダ23によりオペレーションコード“bc”が検出されたときに、論理“1”となる。入力信号-D\_DISP\_EQ\_0は、命令により指定されたディスプレイメントが0でないときに、論理“1”となる。

40

【0124】

また、入力信号+D\_BR\_EQ\_E、+D\_XR\_EQ\_Eは、それぞれ、命令により指定されたベースレジスタ、インデックスレジスタの番号が“14”のときに、論理“1”となる。これらの信号は、デコーダ23からRSBR36に出力される。

【0125】

OR回路141は、信号+D\_BR\_EQ\_Eと+D\_XR\_EQ\_Eの論理和を表す信号を出力し、AND回路142は、信号+D\_BCと-D\_DISP\_EQ\_0、およびOR回路141の出力信号の論理積を、信号+D\_BC\_GIDDY\_RTNとして出力する。

50

## 【 0 1 2 6 】

また、ラッチ回路 1 4 3 は、OR 回路 1 4 1 からの信号 + D \_ B C \_ G I D D Y \_ R T N をラッチし、信号 + R S B R \_ B C \_ G I D D Y \_ R T N として出力する。この信号は、対応する R S B R 3 6 が有効である間、ラッチ回路 1 4 3 に保持され、サブルーチンリターン相当の命令の戻り先が異なることを示すフラグとして用いられる。

## 【 0 1 2 7 】

このフラグ + R S B R \_ B C \_ G I D D Y \_ R T N は、信号 + B R H I S \_ U P D A T E \_ B C \_ G I D D Y \_ R T N として分岐予測機構 2 2 に送出され、図 1 6 に示すように、ブランチヒストリ 3 4 のエントリのフラグ G I D D Y \_ R T N に設定される。

## 【 0 1 2 8 】

図 1 6 のブランチヒストリ 3 4 のエントリには、フラグ G I D D Y \_ R T N 以外に、分岐命令アドレス I A R、分岐先アドレス T I A R、およびフラグ C A L L、R T N が格納されている。フラグ C A L L、R T N は、それぞれ、サブルーチンコールフラグ、サブルーチンリターンフラグに対応する。

## 【 0 1 2 9 】

例えば、ディスプレイメント d が 0 でないような分岐命令 “ b c m . d ( 1 4 ) ” がデコードされた場合、信号 + D \_ B C \_ G I D D Y \_ R T N が論理 “ 1 ” となり、フラグ + R S B R \_ B C \_ G I D D Y \_ R T N が立つ。したがって、ブランチヒストリ 3 4 にこの分岐命令が登録されたとき、対応するフラグ G I D D Y \_ R T N に論理 “ 1 ” が格納される。

## 【 0 1 3 0 】

予測回路 3 1 による分岐予測時に、このフラグ G I D D Y \_ R T N が立っていた場合、リターンアドレススタック 3 5 はリターン命令を予測したときと同様のポップ動作を行うが、予測回路 3 1 は、リターンアドレススタック 3 5 からポップされた分岐先アドレスではなく、ブランチヒストリ 3 4 に登録された分岐先アドレスを、予測分岐先アドレスとして出力する。したがって、ブランチヒストリ 3 4 により予測された分岐先の命令フェッチが行われ、リターンアドレススタック 3 5 による予測結果は廃棄される。

## 【 0 1 3 1 】

上述した実施形態においては、リンクスタック 3 3 に登録されたリンクレジスタの番号と、実行された（実行される）命令の分岐先アドレスレジスタの番号を比較することで、その命令がサブルーチンリターン相当の命令であるか否かを判別している。これ以外の他の実施形態として、リンクスタック 3 3 を用いず、リターンアドレススタック 3 5 に登録されている戻り先アドレスと、実行された（実行される）命令の分岐先アドレスを比較することで、同様の判別を行うことも考えられる。

## 【 0 1 3 2 】

この方法では、上述した b c 命令等のように、戻り先がコール命令の直後とならないリターン相当の命令が出現すると、認識されるコール・リターン対の対応関係が崩れてしまい、リターンアドレススタック 3 5 の本来の性能が発揮できない。しかし、リンクスタック 3 3 を新たに設ける必要がないというメリットがある。

## 【 0 1 3 3 】

図 1 7 は、分岐予測機構 2 2 内でこのような判別を行う回路を示している。図 1 7 において、信号 + B R H I S \_ U P D A T E \_ T I A R は、サブルーチンリターン相当の命令になり得る命令の分岐先アドレスを表し、R S B R 3 6 から送出される。

## 【 0 1 3 4 】

比較回路 1 5 1 は、この信号 + B R H I S \_ U P D A T E \_ T I A R をリターンアドレススタック 3 5 の先頭エントリ（エントリ 0）と比較し、それらが一致すると論理 “ 1 ” の信号を出力する。ここでは、リターンアドレススタック 3 5 は、n 段のスタックとして示されている。AND 回路 1 5 2 は、図 5 の信号 + B R H I S \_ U P D A T E \_ S U B R O U T I N E \_ R T N と比較回路 1 5 1 の出力信号の論理積を、図 1 2 の信号 + B R \_ C O M P \_ S U B R O U T I N E \_ R T N として出力する。

10

20

30

40

50

## 【 0 1 3 5 】

図 1 7 の判別回路は、図 1 1 に示したサブルーチンリターン相当の命令の判別回路の代用となり、リンクスタック 3 3 のエントリを参照しなくても、サブルーチンリターンフラグを生成することができる。したがって、この場合、リンクスタック 3 3 は不要となる。

## 【 0 1 3 6 】

以上説明した実施形態においては、主として、リンクスタック 3 3 およびリターンアドレススタック 3 5 を 2 段のスタックとしているが、任意の段数のスタックを用いた場合も、同様の制御を行うことができる。また、レジスタ番号や命令アドレス以外にも、サブルーチンの戻り先アドレスを指定する任意の情報を比較することで、サブルーチンコール・リターン命令対を認識することができる。

10

## 【 0 1 3 7 】

## 【 発明の効果 】

本発明によれば、リターンアドレススタック等の分岐予測機構を有する情報処理装置において、正しいサブルーチンコール・リターン命令対を動的に抽出することができる。したがって、分岐予測機構におけるコール・リターン対の対応関係のズレが防止され、サブルーチンリターン相当の命令の分岐予測の精度が向上する。

## 【 図面の簡単な説明 】

【 図 1 】 本発明の分岐予測装置の原理図である。

【 図 2 】 命令コードを示す図である。

【 図 3 】 命令処理装置の構成図である。

20

【 図 4 】 リンクスタックとリターンアドレススタックの対応関係を示す図である。

【 図 5 】 命令処理装置で用いられる信号を示す図である。

【 図 6 】 第 1 の判別回路を示す図である。

【 図 7 】 登録回路を示す図である。

【 図 8 】 選択回路を示す図である。

【 図 9 】 第 1 の識別回路を示す図である。

【 図 1 0 】 第 2 の識別回路を示す図である。

【 図 1 1 】 第 2 の判別回路を示す図である。

【 図 1 2 】 制御回路を示す図である。

【 図 1 3 】 ラッチ回路を示す図である。

30

【 図 1 4 】 無効化回路を示す図である。

【 図 1 5 】 フラグ生成回路を示す図である。

【 図 1 6 】 ブランチヒストリに登録されるエントリを示す図である。

【 図 1 7 】 第 3 の判別回路を示す図である。

【 図 1 8 】 サブルーチンコール・リターン命令対を示す図である。

## 【 符号の説明 】

- 1 格納手段
- 2 比較手段
- 3 識別手段
- 4 スタック手段
- 5 プッシュ手段
- 6 リターンアドレススタック手段
- 1 1 オペレーションコード
- 1 2、1 3 オペランド
- 2 1 命令フェッチ回路
- 2 2 分岐予測機構
- 2 3 デコーダ
- 2 4 分岐先アドレス生成回路
- 2 5 分岐命令実行処理回路
- 2 6 命令実行完了処理回路

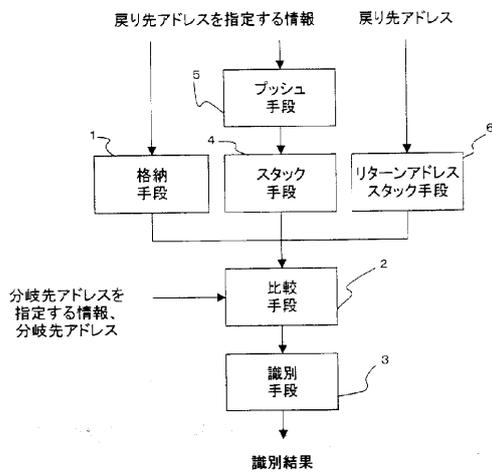
40

50

- 3 1 予測回路
- 3 2、1 5 1 比較回路
- 3 3 リンクスタック
- 3 5 リターンアドレススタック
- 3 6 R S B R
- 4 1、6 1、6 2、8 1、9 1、9 2、1 0 3、1 1 1、1 1 2、1 1 3、1 1 4、1 1 5、1 1 7、1 4 2、1 5 2 AND回路
- 4 2、4 3、6 3、8 2、1 0 2、1 1 6、1 4 1 OR回路
- 5 1、5 2、8 3、1 2 1、1 2 2、1 4 3 ラッチ回路
- 7 1、8 4 NAND回路
- 1 0 1 EXNOR回路
- 1 3 1 NOR回路

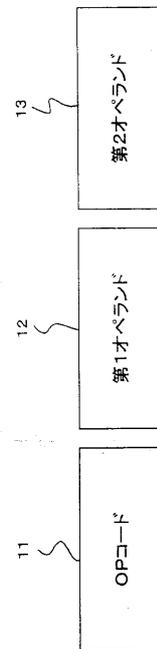
【 図 1 】

本発明の原理図



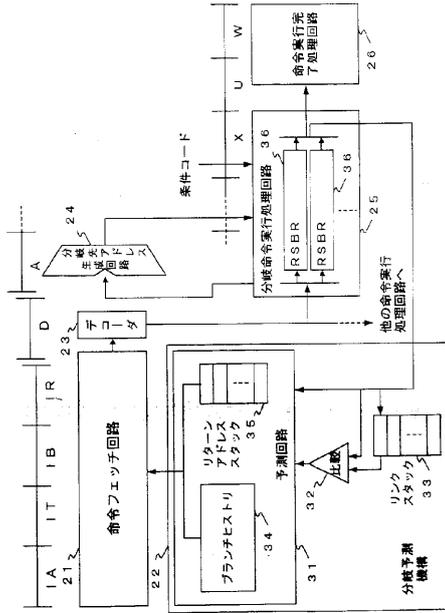
【 図 2 】

命令コードを示す図



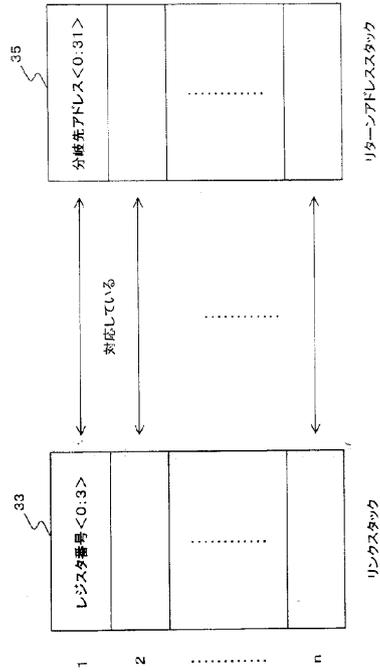
【図3】

命令処理装置の構成図



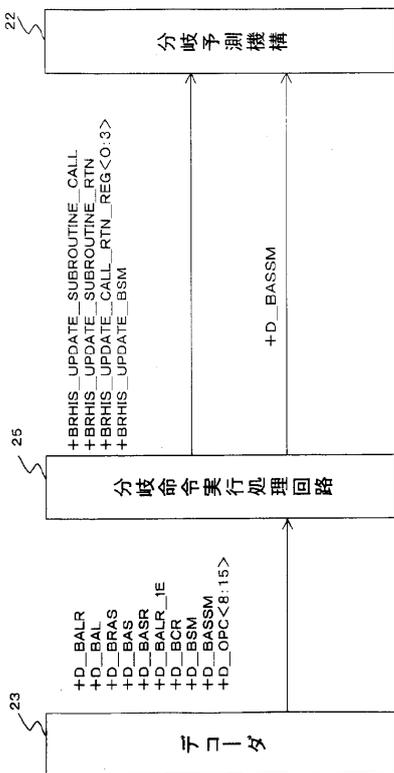
【図4】

リンクスタックとリターンアドレススタックの対応関係を示す図



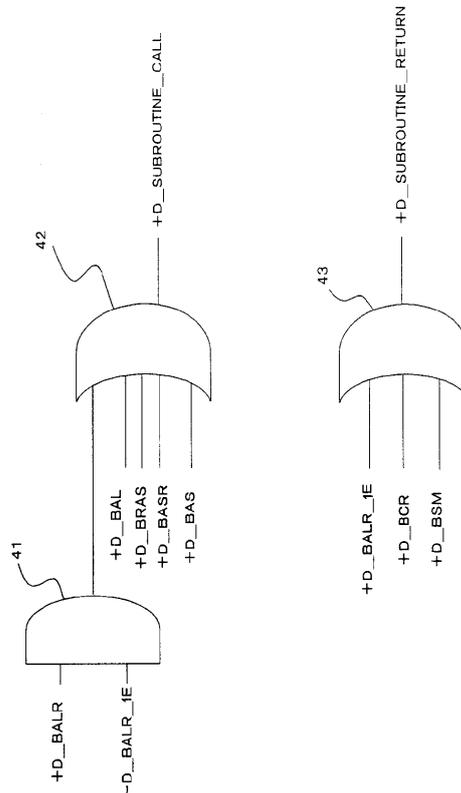
【図5】

命令処理装置で用いられる信号を示す図



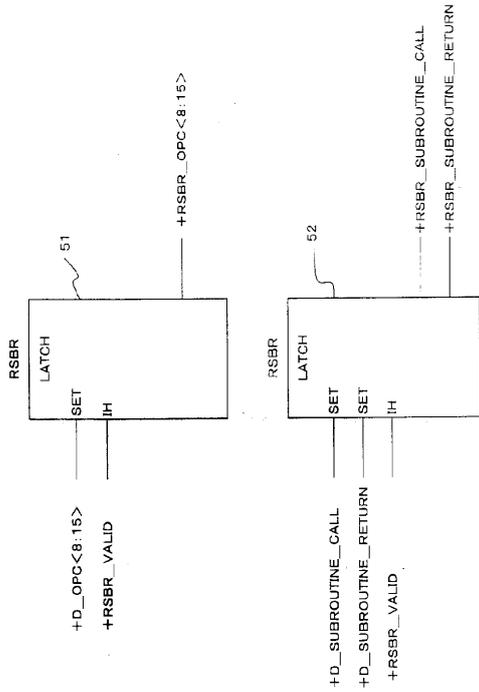
【図6】

第1の判別回路を示す図



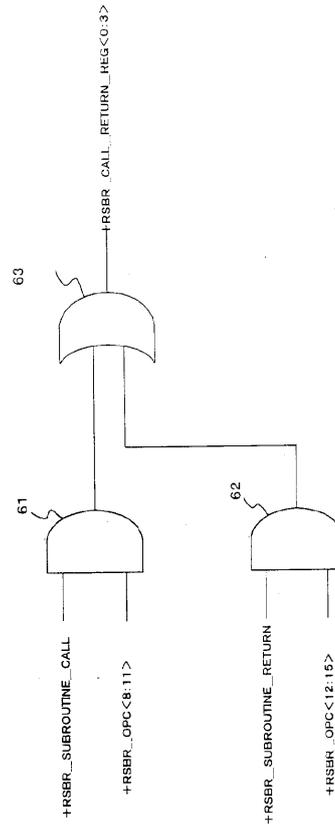
【 図 7 】

登録回路を示す図



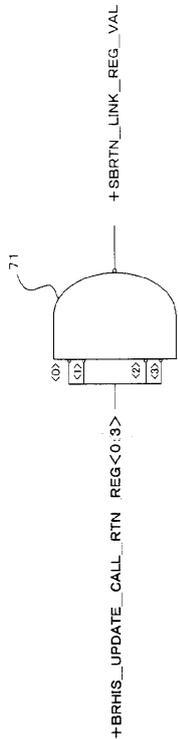
【 図 8 】

選択回路を示す図



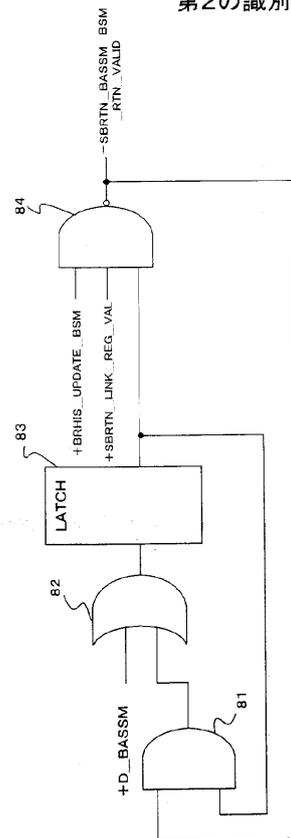
【 図 9 】

第1の識別回路を示す図



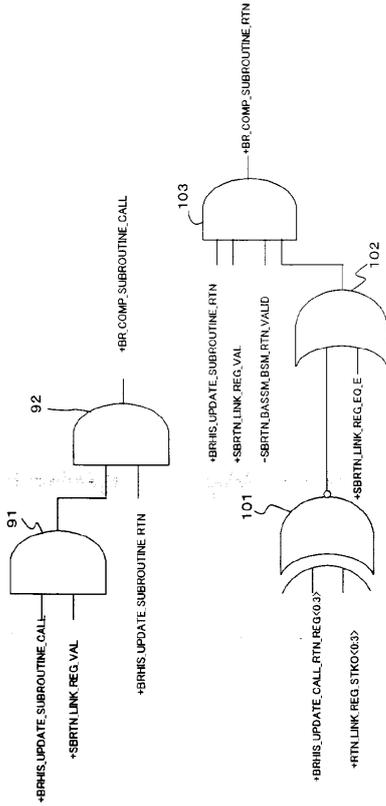
【 図 10 】

第2の識別回路を示す図



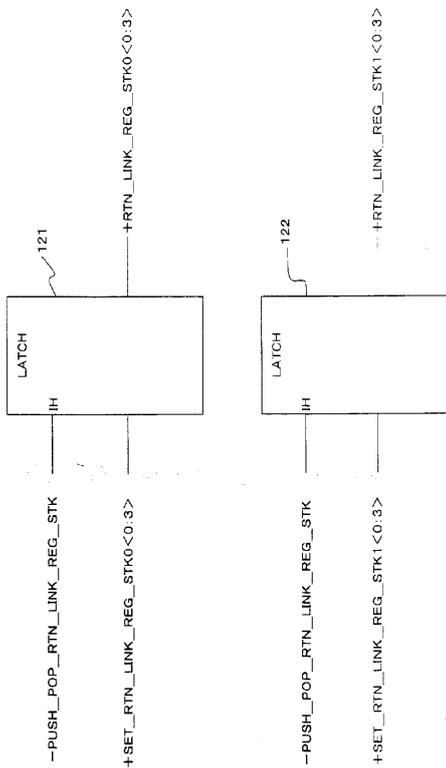
【 図 1 1 】

第2の判別回路を示す図



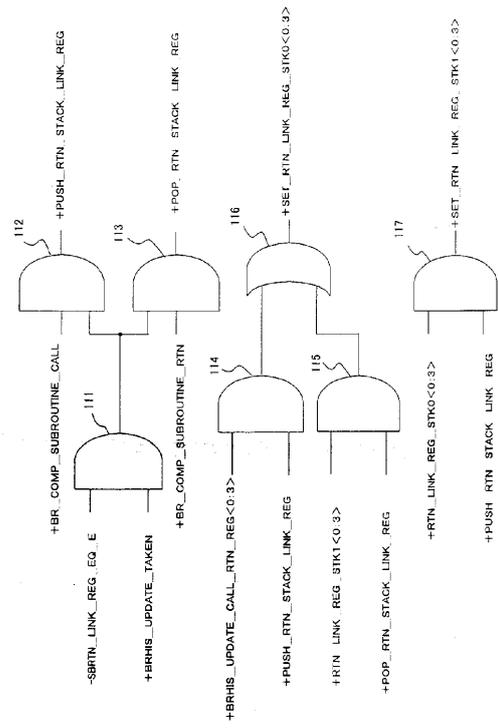
【 図 1 3 】

ラッチ回路を示す図



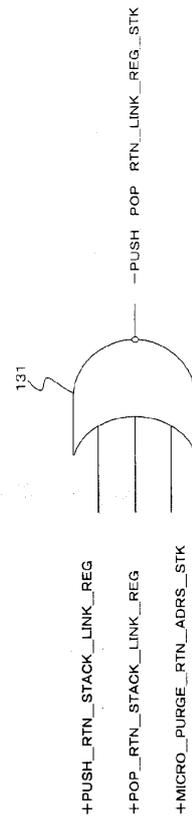
【 図 1 2 】

制御回路を示す図



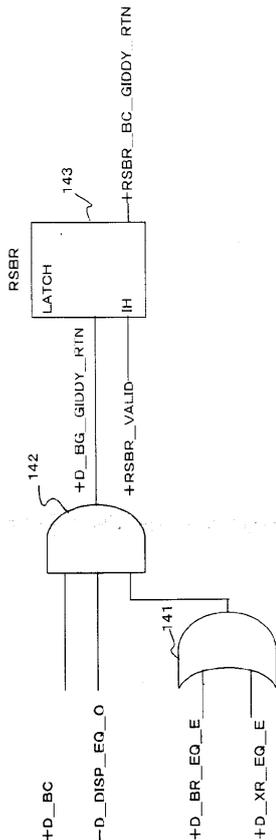
【 図 1 4 】

無効化回路を示す図



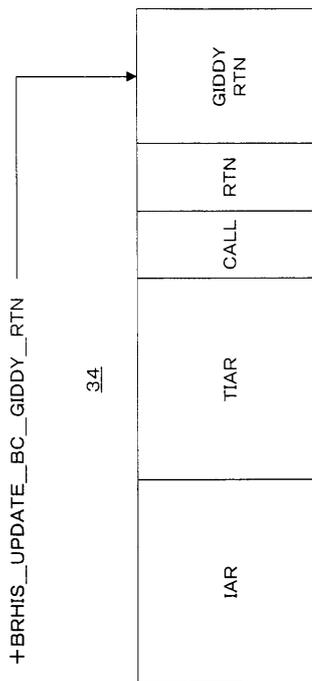
【 図 1 5 】

フラグ生成回路を示す図



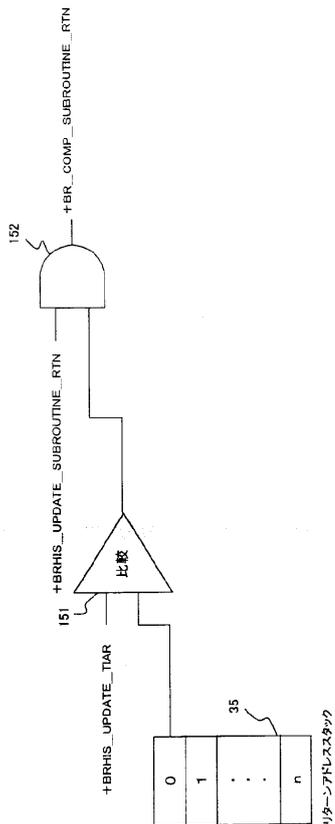
【 図 1 6 】

ブランチヒストリに登録されるエントリを示す図



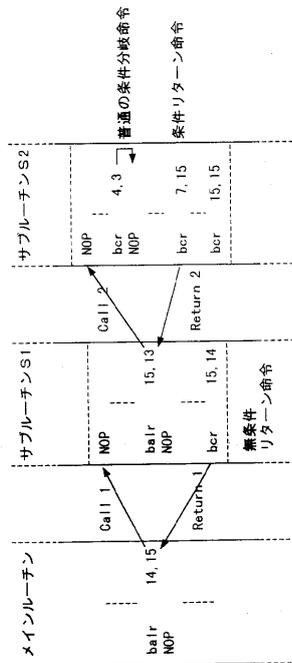
【 図 1 7 】

第3の判別回路を示す図



【 図 1 8 】

サブルーチンコール・リターン命令対を示す図



---

フロントページの続き

(72)発明者 井上 愛一郎  
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

審査官 後藤 彰

(56)参考文献 特開平11-259298(JP,A)  
特開平9-218786(JP,A)  
特開平6-59888(JP,A)  
特開平5-120013(JP,A)

(58)調査した分野(Int.Cl.<sup>7</sup>, DB名)  
G06F 9/38 - 9/42