

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3866920号

(P3866920)

(45) 発行日 平成19年1月10日(2007.1.10)

(24) 登録日 平成18年10月13日(2006.10.13)

(51) Int. Cl.

G06F 9/34 (2006.01)

F I

G06F 9/34 330

請求項の数 19 (全 35 頁)

| | | | |
|---------------|-------------------------------|-----------|---------------------|
| (21) 出願番号 | 特願2000-562823 (P2000-562823) | (73) 特許権者 | 591016172 |
| (86) (22) 出願日 | 平成11年1月18日 (1999.1.18) | | アドバンスト・マイクロ・デバイス・ |
| (65) 公表番号 | 特表2002-521762 (P2002-521762A) | | インコーポレイテッド |
| (43) 公表日 | 平成14年7月16日 (2002.7.16) | | ADVANCED MICRO DEVI |
| (86) 国際出願番号 | PCT/US1999/001048 | | CES INCORPORATED |
| (87) 国際公開番号 | W02000/007097 | | アメリカ合衆国、94088-3453 |
| (87) 国際公開日 | 平成12年2月10日 (2000.2.10) | | カリフォルニア州、サニibel、ピィ・ |
| 審査請求日 | 平成18年1月17日 (2006.1.17) | | オウ・ボックス・3453、ワン・エイ・ |
| (31) 優先権主張番号 | 09/127,094 | | エム・ディ・プレイス、メイル・ストップ |
| (32) 優先日 | 平成10年7月31日 (1998.7.31) | | ・68 (番地なし) |
| (33) 優先権主張国 | 米国 (US) | (74) 代理人 | 100064746 |
| | | | 弁理士 深見 久郎 |
| | | (74) 代理人 | 100085132 |
| | | | 弁理士 森田 俊雄 |

最終頁に続く

(54) 【発明の名称】 命令のリタイアメント時に物理レジスタを選択的に自由にするよう構成されたプロセッサ

(57) 【特許請求の範囲】

【請求項1】

プロセッサにおいて物理レジスタを管理するための方法であって、前記方法は、

第1の物理レジスタを第1の命令動作の宛先オペランドに割当てるステップを含み、前記宛先オペランドは論理レジスタを識別し、前記方法はさらに、

前記第1の命令動作をリタイアさせるステップと、

前記リタイアさせるステップにตอบสนองして、アーキテクテッド再命名ブロックを前記第1の物理レジスタに対応する第1の物理レジスタ番号で更新するステップとを含み、前記第1の物理レジスタ番号はレジスタファイルにおける前記第1の物理レジスタを識別し、前記方法はさらに、

前記リタイアさせるステップにตอบสนองして、前記アーキテクテッド再命名ブロックから第2の物理レジスタ番号を読み出すステップを含み、前記第2の物理レジスタ番号は前記第1の物理レジスタ番号により前記アーキテクテッド再命名ブロック内の前記論理レジスタを表わすものからおきかえられ、前記方法はさらに、

前記更新するステップの後に、前記第2の物理レジスタ番号が前記アーキテクテッド再命名ブロック内の第2の論理レジスタを表わすことを決定するステップと、

前記第2の物理レジスタ番号が前記アーキテクテッド再命名ブロック内の前記第2の論理レジスタを表わすとの前記決定にตอบสนองして、前記第2の物理レジスタ番号に対応する第2の物理レジスタの自由化を禁止するステップとを含み、前記第2の物理レジスタ番号は前記レジスタファイルにおける前記第2の物理レジスタを識別する、方法。

10

20

【請求項 2】

前記第 2 の物理レジスタ番号が前記アーキテクテッド再命名ブロック内の前記第 2 の論理レジスタを表わさないとの決定にตอบสนองして、前記第 2 の物理レジスタを自由にするステップをさらに含む、請求項 1 に記載の方法。

【請求項 3】

前記更新するステップの後に、前記第 2 の物理レジスタ番号を前記アーキテクテッド再命名ブロックに記憶された前記物理レジスタ番号と比較し、前記第 2 の物理レジスタ番号が前記アーキテクテッド再命名ブロック内の第 2 の論理レジスタを表わすとの前記決定を行なうステップをさらに含む、請求項 2 に記載の方法。

【請求項 4】

前記禁止するステップが前記比較するステップにตอบสนองして行なわれ、前記第 2 の物理レジスタ番号における少なくとも 1 つの一致を検出する、請求項 3 に記載の方法。

【請求項 5】

前記自由にするステップが前記比較するステップにตอบสนองして行なわれ、前記第 2 の物理レジスタ番号における一致を検出しない、請求項 4 に記載の方法。

【請求項 6】

前記割当てするステップにตอบสนองして、前記第 1 の物理レジスタ番号をサイロに記憶するステップをさらに含む、請求項 1 に記載の方法。

【請求項 7】

前記リタイアさせるステップが、前記第 1 の物理レジスタ番号が前記サイロのヘッドに達すると行なわれる、請求項 6 に記載の方法。

【請求項 8】

プログラム命令における前記第 1 の命令動作の後に第 2 の命令動作をリタイアさせるステップを含み、前記第 2 の命令動作は前記第 2 の命令動作の宛先レジスタに割当てられた第 3 の物理レジスタを有し、前記宛先レジスタは前記第 2 の論理レジスタを含み、さらに

前記アーキテクテッド再命名ブロックを前記リタイアさせるステップにตอบสนองして前記第 3 の物理レジスタに対応する第 3 の物理レジスタ番号で更新するステップと、

前記リタイアさせるステップにตอบสนองして前記アーキテクテッド再命名ブロックから前記第 2 の物理レジスタ番号を读出すステップと、

前記第 2 の物理レジスタ番号を自由にするステップとをさらに含む、請求項 1 に記載の方法。

【請求項 9】

第 1 の物理レジスタを命令動作の宛先オペランドに割当てよう構成されたマップユニットを含み、前記マップユニットはさらに、いずれの物理レジスタが割当てのために自由であるかを示す自由リストを維持するよう構成され、前記マップユニットは前記自由リストから前記第 1 の物理レジスタを選択するよう構成され、さらに、

前記命令動作がリタイアしていることの指標および前記第 1 の物理レジスタに対応する第 1 の物理レジスタ番号を受けよう結合されたアーキテクチャ的再命名ブロックを含み、

前記第 1 の物理レジスタ番号はレジスタファイルにおける前記第 1 の物理レジスタを識別し、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアする前に、前記宛先オペランドにより識別された第 1 の論理レジスタに対応する第 2 の物理レジスタを識別する第 2 の物理レジスタ番号をもたらすよう構成され、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標にตอบสนองして前記第 2 の物理レジスタ番号をもたらし、前記第 2 の物理レジスタ番号は前記レジスタファイルにおける前記第 2 の物理レジスタを識別し、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標にตอบสนองして、前記第 1 の物理レジスタ番号を捕捉し、かつ前記第 1 の物理レジスタ番号を前記第 1 の論理レジスタに対応するものとして保持するよう構成され、前記アーキテクチャ的再命名ブロックは、前記第 2 の物理レジスタ番号が

10

20

30

40

50

さらに第 2 の論理レジスタに対応するか否かを判定するようさらに構成され、

前記マップユニットは、前記第 2 の物理レジスタが前記第 2 の論理レジスタを表わす場合、前記第 2 の物理レジスタ番号を前記自由リストに加えることを禁止するよう構成される、プロセッサ。

【請求項 10】

前記マップユニットが、前記第 2 の物理レジスタが前記第 2 の論理レジスタを表わさない場合、前記物理レジスタ番号を前記自由リストに加えるよう構成される、請求項 9 に記載のプロセッサ。

【請求項 11】

前記アーキテクチャ的再命名ブロックは、前記第 1 の物理レジスタ番号の捕捉の後に前記第 2 の物理レジスタ番号を前記アーキテクチャ的再命名ブロック内の前記物理レジスタ番号と比較することによって、前記第 2 の物理レジスタ番号が前記第 2 の論理レジスタにさらに対応するか否かを判定するよう構成される、請求項 9 に記載のプロセッサ。

10

【請求項 12】

前記アーキテクチャ的再命名ブロックが連想メモリを含む、請求項 11 に記載のプロセッサ。

【請求項 13】

前記マップユニットに結合されたサイロをさらに含み、前記サイロは、前記マップユニットによる前記命令動作への割当てに前記第 1 の物理レジスタ番号を記憶するよう構成される、請求項 9 に記載のプロセッサ。

20

【請求項 14】

前記サイロが前記アーキテクチャ的再命名ブロックにさらに結合され、前記サイロが前記命令動作のリタイア時に前記第 1 の物理レジスタ番号を与えるよう構成される、請求項 13 に記載のプロセッサ。

【請求項 15】

プロセッサにおいて物理レジスタを管理するための方法であって、前記方法は、第 1 の物理レジスタを命令動作の宛先オペランドに割当てするステップを含み、前記宛先オペランドは論理レジスタを識別し、前記方法はさらに、

前記命令動作をリタイアさせるステップと、

前記リタイアさせるステップに回答して、アーキテクチャ的再命名ブロックを前記第 1 の物理レジスタに対応する第 1 の物理レジスタ番号で更新するステップとを含み、前記第 1 の物理レジスタ番号はレジスタファイルにおける前記第 1 の物理レジスタを識別し、前記方法はさらに、

30

前記リタイアさせるステップに回答して、前記アーキテクチャ的再命名ブロックから第 2 の物理レジスタに対応する第 2 の物理レジスタ番号を読み出すステップを含み、前記第 2 の物理レジスタ番号は、前記第 1 の物理レジスタ番号により前記アーキテクチャ的再命名ブロック内の前記論理レジスタを表わすものからおきかえられ、前記第 2 の物理レジスタ番号は前記レジスタファイルにおける前記第 2 の物理レジスタを識別し、前記方法はさらに、

前記第 2 の物理レジスタを自由にするステップを含む、方法。

40

【請求項 16】

第 1 の物理レジスタを命令動作の宛先オペランドに割当てするよう構成されたマップユニットを含み、前記マップユニットは、いずれの物理レジスタが割当てのために自由であることを示す自由リストを維持するようさらに構成され、前記マップユニットは、前記自由リストから前記第 1 の物理レジスタを選択するよう構成されており、さらに、

前記命令動作がリタイアしているという指標および前記第 1 の物理レジスタに対応する第 1 の物理レジスタ番号を受けよう結合されたアーキテクチャ的再命名ブロックを含み、

前記第 1 の物理レジスタ番号はレジスタファイルにおける前記第 1 の物理レジスタを識別し、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアする前に、前記宛

50

先オペランドにより識別された第1の論理レジスタに対応する第2の物理レジスタを識別する第2の物理レジスタ番号をもたらすよう構成され、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標に応答して前記第2の物理レジスタ番号をもたらし、前記第2の物理レジスタ番号は前記レジスタファイルにおける前記第2の物理レジスタを識別し、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標に応答して前記第1の物理レジスタ番号を捕捉し、かつ前記第1の物理レジスタ番号を前記第1の論理レジスタに対応するものとして保持するよう構成されており、

前記マップユニットは、前記第2の物理レジスタ番号を前記自由リストに加えるよう構成される、プロセッサ。

10

【請求項17】

第1の物理レジスタを命令動作の宛先オペランドに割当てよう構成されたマップユニットを含み、前記マップユニットは、いずれの物理レジスタが割当てのために自由であることを示す自由リストを維持するようさらに構成され、前記マップユニットは前記自由リストから前記第1の物理レジスタを選択するよう構成されており、さらに、

前記命令動作がリタイアしているという指標および前記第1の物理レジスタに対応する第1の物理レジスタ番号を受けよう結合されたアーキテクチャ的再命名ブロックを含み、前記第1の物理レジスタ番号はレジスタファイルにおける前記第1の物理レジスタを識別し、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアする前に、前記宛先オペランドにより識別された第1の論理レジスタに対応する第2の物理レジスタを識別する第2の物理レジスタ番号をもたらすよう構成され、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標に応答して前記第2の物理レジスタ番号をもたらし、前記第2の物理レジスタ番号は前記レジスタファイルにおける前記第2の物理レジスタを識別し、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標に応答して、前記第1の物理レジスタ番号を捕捉し、かつ前記第1の物理レジスタ番号を前記第1の論理レジスタに対応するものとして保持するよう構成され、

20

前記マップユニットが、前記第2の物理レジスタ番号を前記自由リストに加えるよう構成されている、プロセッサと、

前記プロセッサに結合された周辺装置とを含み、前記周辺装置は、前記コンピュータシステムと別のコンピュータシステムとの間で通信するよう構成される、コンピュータシステム。

30

【請求項18】

前記アーキテクチャ的再命名ブロックが、前記第2の物理レジスタ番号が第2の論理レジスタにさらに対応するか否かを判定するようさらに構成され、前記マップユニットが、前記第2の物理レジスタが前記第2の論理レジスタを表わす場合、前記第2の物理レジスタ番号を前記自由リストに加えることを禁止するよう構成される、請求項17に記載のコンピュータシステム。

【請求項19】

第1の物理レジスタを命令動作の宛先オペランドに割当てよう構成されたマップユニットを含み、前記マップユニットは、いずれの物理レジスタが割当てのために自由であることを示す自由リストを維持するようさらに構成され、前記マップユニットは、前記自由リストから前記第1の物理レジスタを選択するよう構成されており、さらに、

40

前記命令動作がリタイアしているという指標と前記第1の物理レジスタに対応する第1の物理レジスタ番号とを受けよう結合されたアーキテクチャ的再命名ブロックを含み、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアする前に、前記宛先オペランドにより識別された第1の論理レジスタに対応する第2の物理レジスタを識別する第2の物理レジスタ番号をもたらすよう構成され、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアしているという前記指標に応答して前記第2の物理レジスタ番号をもたらし、前記アーキテクチャ的再命名ブロックは、前記命令動作がリタイアして

50

いるという前記指標に応答して、前記第1の物理レジスタ番号を捕捉し、かつ前記第1の物理レジスタ番号を前記第1の論理レジスタに対応するものとして保持するよう構成されており、

前記マップユニットは、前記第2の物理レジスタ番号を前記自由リストに加えるよう構成される、

第2のプロセッサをさらに含む、請求項17に記載のコンピュータシステム。

【発明の詳細な説明】

【0001】

【発明の分野】

この発明は、プロセッサの分野に関し、より特定的にはプロセッサのレジスタ再命名機能に関する。 10

【0002】

【関連技術の説明】

EP-A-0541216号は、独立した命令ストリームの各々に対してレジスタ再命名ユニットを含むデータ処理システムを開示する。このレジスタ再命名ユニットは、レジスタファイル内において処理状態レジスタを物理レジスタにマッピングするように、再命名動作を行なう。物理レジスタは、それを参照し得たすべての命令がうまく終了されると、自由にされる。

US-A-5805918号は、同じレジスタの別々の部分が、別々のセットアーキテクチャに属する命令によってアクセスされる、システムを開示する。 20

US-A-5546554号は、レジスタ再命名のためのプロセッサにおけるマッピングユニットを開示する。各物理レジスタは、3つの状態（自由、警告および準備完了）のうち1つを有し得る。

EP-A-0463628号は、レジスタ再命名のためのレジスタマッピングを開示する。

スーパースカラプロセッサにおいては、クロックサイクルごとに複数の命令を発行し実行し、かつ設計と一貫した可能な最高のクロック周波数を用いることによって高性能を達成することが試みられている。クロックサイクルごとに実行される命令の数を増加するための方法の1つは、「アウトオブオーダー」(out of order)実行である。アウトオブオーダー実行において、命令は、プログラムシーケンス（または「プログラム順序」）において特定されるものとは異なる順序において実行され得る。プログラムシーケンスにおいて互いに近いある命令は、それらの同時の実行を禁じる従属性を有しているかもしれないが、他方では、プログラムシーケンスにおける後の命令は前の命令に対する従属性を有していないかもしれない。したがって、アウトオブオーダー実行により、（平均して）同時に実行される命令の数を増加することによってスーパースカラプロセッサの性能が向上し得る。 30

【0003】

残念なことに、アウトオブオーダー実行では、プロセッサのハードウェアの複雑さがさらに増す。たとえば、プログラム順序において第1の命令の後の第2の命令は、第1の命令によって読出される記憶場所を更新することがある。換言すると、第2の命令の宛先オペランドが第1の命令の発信元オペランドの1つであり得る。適切にプログラムを実行するために、第1の命令は発信元オペランドとして、記憶場所に記憶された値を第2の命令の実行の前に受取らなければならない。同様に、第1および第2の命令が宛先オペランドとしてある特定の記憶場所を有している場合、第2の命令の結果は、第1および第2の命令の双方を実行した後に（かつ記憶場所を更新する第3の命令を実行する前に）記憶場所に記憶される値であるべきである。 40

【0004】

一般的に、命令は1つ以上の発信元オペランドと1つ以上の宛先オペランドを有し得る。発信元オペランドは、（宛先オペランドである）1つ以上の結果を生成するために命令定義に従って操作すべき入力値である。発信元および宛先オペランドは、プロセッサの外部のメモリ場所に記憶されるメモリオペランドであるかもしれないが、またはプロセッサ内に含 50

まれるレジスタ記憶場所に記憶されるレジスタオペランドであるかもしれない。プロセッサにより用いられる命令セットアーキテクチャはいくつかのアーキテクテッドレジスタを規定する。これらのレジスタは命令セットアーキテクチャにより、存在していることが規定され、命令はアーキテクテッドレジスタを発信元および宛先オペランドとして用いるようにコード化され得る。命令は、命令のオペランドフィールドにおけるレジスタ番号（またはレジスタアドレス）を介してある特定のレジスタを発信元または宛先オペランドとして特定する。レジスタ番号はアーキテクテッドレジスタのうちの選択されたレジスタを一意に識別する。発信元オペランドは発信元レジスタ番号によって識別され、宛先オペランドは宛先レジスタ番号によって識別される。

【 0 0 0 5 】

アーキテクテッドレジスタに加えて、いくつかのプロセッサは、命令実行の間に一時的な結果を保持するのに用いることができるさらなるマイクロアーキテクテッドレジスタを規定する。たとえば、いくつかのプロセッサはマイクロコーディング技法を用いて最も複雑な命令を取扱う。マイクロコードルーチンは、複雑な命令に回答して実行され、複数のより単純な命令を含む。マイクロコードルーチンは、複雑な命令を実行すると同時に一時的な結果を生じ得る。これらのマイクロアーキテクテッドレジスタ（または一時レジスタ）はさらなるレジスタ番号に割当てられ、一時レジスタをアーキテクテッドレジスタから一意に識別する。ここではアーキテクテッドレジスタと一時レジスタとは併せて論理レジスタと称する。

【 0 0 0 6 】

アウトオブオーダー実行を用いるプロセッサは、レジスタオペランドに関して上記のハザードを経験し得る。このようなハザードに対処するための方法はレジスタ再命名である。レジスタ再命名において、プロセッサは1セットの物理レジスタを実装する。物理レジスタの数は、プロセッサの命令セットアーキテクチャおよびマイクロアーキテクチャによって特定される論理レジスタの数より大きい。命令が発行されると、物理レジスタは命令の宛先レジスタオペランドに割当てられる。割当てられた物理レジスタを識別する物理レジスタ番号は各宛先オペランドに対して与えられ、どの物理レジスタが論理レジスタに対応するのかを示す指標はプロセッサによって維持される。論理レジスタを発信元オペランドとして有する後の命令には、適当な発信元オペランドを讀出すための対応する物理レジスタ番号が与えられる。各命令の宛先オペランドを記憶するように異なる物理レジスタを割当てることによって、命令はその宛先オペランドを任意の順序で自由に更新することができるが、これは異なる物理記憶場所が更新されるからである。

【 0 0 0 7 】

残念なことに、物理レジスタ番号を宛先オペランド命令に割当て、これらの物理レジスタ番号を、その宛先オペランドを発信元オペランドとして有する後の命令に与えるプロセスは複雑かつ遅いものであり得る。スーパースカラプロセッサにおいて特に困難であるのは、命令の宛先オペランドへの物理レジスタ番号の割当てと、これらの命令と同時にレジスタ再命名ハードウェアを通過している後の従属する命令に物理レジスタ番号を与えること等である。より高い周波数で動作し、なおかつクロックサイクルごとに複数の命令の再命名を取扱うことのできるレジスタ再命名構造が望ましい。

【 0 0 0 8 】

レジスタ再命名は、命令が例外条件を経験する際に困難となる。ここで用いられる例外とは、後の命令を破棄することおよび命令フェッチが異なるアドレスで開始されることが必要となる命令の実行におけるエラーのことを指して言う。たとえば、分岐予測誤りは例外条件である。プロセッサは、分岐予測を行なって、条件付き分岐命令の後の命令を投機的にフェッチし、発行し、実行し得る。その予測が誤りである場合、分岐命令の後の命令は破棄され、分岐命令の実行に応じて命令がフェッチされる。さらなる例外条件には、メモリオペランドのアドレスに対するアドレス変換エラーおよび他のアーキテクチャ的またはマイクロアーキテクチャ的エラー条件が含まれ得る。

【 0 0 0 9 】

レジスタ再命名が、例外によって後で破棄される命令に適用されていることがあり得るため、論理レジスタの物理レジスタへのマッピングは、例外を経験している命令と一致する状態に回復しなければならない。言換えると、論理レジスタの物理レジスタへのマッピングは、(プログラム順序において)例外を経験している命令の前の命令の実行を反映しているべきであり、例外を経験している命令の後の命令の実行を反映すべきではない。例外にตอบสนองしてフェッチされる命令が利用可能になるとすぐにレジスタ再命名ハードウェアを通過することができるように、レジスタ再命名マップの回復が迅速であることが望ましい。新しくフェッチされた命令がレジスタ再命名ハードウェアに到達した際にレジスタ再命名マップの回復が依然として行なわれている場合、新しくフェッチされた命令はレジスタ再命名マップが回復されるまで停止されなければならない。プロセッサの性能はこれによ

10

【0010】

さらに、レジスタ再命名ハードウェアは一般的に、一旦対応する論理レジスタが後の命令に対応する値にコミットされるとある特定の命令の宛先オペランドに前に割当てられた物理レジスタを再利用するための機構を含む。物理レジスタを可能な限り効率よく用いるのが望ましく、また、一旦後の状態が対応する論理レジスタにコミットされた際に物理レジスタを自由にするための正確な方法を提供することが望ましい。

【0011】**【発明の概要】**

上述の課題は、レジスタ再命名ハードウェアを含むマップユニットを用いたプロセッサによってその大部分が解決される。

20

第1の局面によれば、本発明は、請求項10に記載されるプロセッサを提供する。

【0014】

マップユニットは、状態コード(condition code)結果および整数結果の双方を記憶するのに同じ物理レジスタを用いることによって物理レジスタ使用の効率を向上してもよい。物理レジスタを識別する物理レジスタ番号は、状態コードレジスタおよび整数レジスタの双方に対して記録される。再命名として再利用するために物理レジスタを正しく自由にするために、マップユニットは、1つ以上の命令動作をリタイアすることに対応してコミットされる論理レジスタのセットに対応するエントリを更新する前にアーキテクテッド再命名ブロックから前の再命名をポップする(pop)。ポップされた物理レジスタ番号を自由にする前に、ポップされた物理レジスタ番号は、アーキテクテッド再命名ブロックにより維持される更新されたアーキテクチャ的状态に対してカムされる(cammed)。カム一致が検出された場合、ポップされた物理レジスタは自由にされない。有利には、物理レジスタのより効率のよい使用を行なうことができ、また、物理レジスタを正確に自由にすることもできる。状態コードレジスタおよび整数レジスタ再命名に関して物理レジスタをより効率よく用いることに加えて、レジスタを自由にするための本発明の機構を用いた他の最適化が可能である。たとえば、発信元物理レジスタ番号を宛先物理レジスタ番号としてコピーすることによってレジスタ間移動を達成してもよい。本発明の機構により、そのような場合においても物理レジスタ番号が不注意に自由にされることを防ぐことができる。

30

マップユニットの上述の実施例は、さまざまな技術を採用してレジスタ再命名機構の性能および効率を向上させる。

40

第2の局面によれば、本発明は、請求項1に記載される、プロセッサにおいて物理レジスタを管理するための方法を提供する。

【0015】

概して、プロセッサにおける物理レジスタを管理するための方法が企図される。第1の物理レジスタは、命令動作の宛先オペランドに割当てられる。この宛先オペランドは論理レジスタを識別する。命令動作はリタイアされる。命令動作のリタイアにตอบสนองして、アーキテクテッド再命名ブロックが第1の物理レジスタに対応する第1の物理レジスタ番号で更新される。また、命令動作のリタイアにตอบสนองして、第2の物理レジスタ番号がアーキテクテッド再命名ブロックから読出される。第2の物理レジスタ番号は、第1の物理レジスタ

50

番号によってアーキテクテッド再命名ブロック内の論理レジスタを表わすものからおきかえられる。第2の物理レジスタの自由化は、第2の物理レジスタ番号がアーキテクテッド再命名ブロック内の第2の論理レジスタを表わすということの決定に应答して、禁止される。

【0016】

プロセッサが、マップユニットおよびアーキテクチャ的再命名ブロックを含んで企図される。マップユニットは、第1の物理レジスタを命令動作の宛先オペランドに割当てよう構成される。さらに、このマップユニットは、いずれの物理レジスタが割当てのために自由にされるかを示す自由リストを維持するよう構成される。マップユニットは、この自由リストから第1の物理レジスタを選択するよう構成される。命令動作がリタイアしているという指標および第1の物理レジスタに対応する第1の物理レジスタ番号を受けよう結合されて、アーキテクチャ的再命名ブロックは、命令動作がリタイアしているという指標に应答して宛先オペランドにより識別された第1の論理レジスタに前に対応していた第2の物理レジスタを識別する第2の物理レジスタ番号を与えるよう構成される。アーキテクチャ的再命名ブロックは、第1の物理レジスタ番号を捕捉し、かつその第1の物理レジスタ番号を、命令動作がリタイアしているという指標に应答して、第1の論理レジスタに対応するものとして保持するよう構成される。またさらに、アーキテクチャ的再命名ブロックは、第2の物理レジスタ番号が第2の論理レジスタにさらに対応するか否かを決定するようさらに構成される。マップユニットは、第2の物理レジスタが第2の論理レジスタを表わす場合、第2の物理レジスタ番号を自由リストに加えることを禁止するよう構成される。

10

20

【0017】

プロセッサにおける物理レジスタを管理するための方法が企図される。第1の物理レジスタは、命令動作の宛先オペランドに割当てられる。この宛先オペランドは論理レジスタを識別する。命令動作はリタイアされる。命令動作のリタイアに应答して、アーキテクテッド再命名ブロックが第1の物理レジスタに対応する第1の物理レジスタ番号で更新される。また、命令動作のリタイアに应答して、第2の物理レジスタ番号がアーキテクテッド再命名ブロックから読出される。第2の物理レジスタ番号は、第1の物理レジスタ番号によってアーキテクテッド再命名ブロック内の論理レジスタを表わすものからおきかえられている。第2の物理レジスタは自由にされる。

30

【0018】

プロセッサが企図される。このプロセッサは、マップユニットおよびアーキテクチャ的再命名ブロックを含む。マップユニットは、第1の物理レジスタを命令動作の宛先オペランドに割当てよう構成され、さらに、いずれの物理レジスタが割当てのために自由にされるかを示す自由リストを維持するよう構成される。マップユニットは、この自由リストから第1の物理レジスタを選択するよう構成される。命令動作がリタイアしているという指標および第1の物理レジスタに対応する第1の物理レジスタ番号を受けよう結合されて、アーキテクチャ的再命名ブロックは、命令動作がリタイアしているという指標に应答して宛先オペランドにより識別された第1の論理レジスタに前に対応していた第2の物理レジスタを識別する第2の物理レジスタ番号を与えるよう構成される。さらに、アーキテクチャ的再命名ブロックは、命令動作がリタイアしているという指標に应答して、第1の物理レジスタ番号を捕捉し、かつその第1の物理レジスタ番号を第1の論理レジスタに対応するものとして保持するよう構成される。マップユニットは、第2の物理レジスタ番号を自由リストに加えるよう構成される。

40

【0019】

プロセッサおよびI/O装置を含むコンピュータシステムが企図される。プロセッサはマップユニットおよびアーキテクチャ的再命名ブロックを含む。マップユニットは、第1の物理レジスタを命令動作の宛先オペランドに割当てよう構成され、さらに、いずれの物理レジスタが割当てのために自由にされるかを示す自由リストを維持するよう構成される。マップユニットは、この自由リストから第1の物理レジスタを選択するよう構成される

50

。命令動作がリタイアしているという指標および第1の物理レジスタに対応する第1の物理レジスタ番号を受けよう結合されて、アーキテクチャ的再命名ブロックは、命令動作がリタイアしているという指標に回答して宛先オペランドにより識別された第1の論理レジスタに前に対応していた第2の物理レジスタを識別する第2の物理レジスタ番号を与えるよう構成される。さらに、アーキテクチャ的再命名ブロックは、命令動作がリタイアしているという指標に回答して、第1の物理レジスタ番号を捕捉し、かつその第1の物理レジスタ番号を第1の論理レジスタに対応するものとして保持するよう構成される。マップユニットは、第2の物理レジスタ番号を自由リストに加えるよう構成される。プロセッサに結合されて、I/O装置は、このコンピュータシステムと、I/O装置が結合される別のコンピュータシステムとの間で通信を行なうよう構成される。

10

【0020】

この発明の他の目的および利点は、以下の詳細な説明を読み、添付の図面を参照することにより明らかになるであろう。

【0021】

【詳細な説明】

この発明はさまざまな変更および代替の形態の影響を受け得るが、その特定の実施例が図面において例によって示され、ここに詳細に説明される。しかしながら、その図面および詳細な説明は、この発明を、開示される特定の形態に限定することが意図されるものではなく、むしろ、添付の請求の範囲によって規定されるこの発明の精神および範囲内に含まれるすべての変更、均等物および代替例を網羅することが意図される。

20

【0022】

図1を参照して、プロセッサ10の一実施例のブロック図が示される。他の実施例も可能であり考慮される。図1の実施例において、プロセッサ10はラインプレディクタ(predictor)12、命令キャッシュ(Iキャッシュ)14、アライメントユニット16、分岐履歴テーブル18、間接アドレスキャッシュ20、戻りスタック22、復号ユニット24、プレディクタミス復号ユニット26、マイクロコードユニット28、マップユニット30、マップサイロ32、アーキテクチャ的再命名ブロック34、一对の命令キュー36A-36B、一对のレジスタファイル38A-38B、一对の実行コア40A-40B、ロード/ストアユニット42、データキャッシュ(Dキャッシュ)44、外部インターフェイスユニット46、PCサイロおよびリダイレクトユニット48ならびに命令TLB(ITB)50を含む。ラインプレディクタ12は、ITB50、プレディクタミス復号ユニット26、分岐履歴テーブル18、間接アドレスキャッシュ20、戻りスタック22、PCサイロおよびリダイレクトブロック48、アライメントユニット16ならびにIキャッシュ14に接続される。Iキャッシュ14はアライメントユニット16に接続される。アライメントユニット16はさらに、プレディクタミス復号ユニット26および復号ユニット24に接続される。復号ユニット24はさらに、マイクロコードユニット28およびマップユニット30に接続される。マップユニット30は、マップサイロ32と、アーキテクチャ的再命名ブロック34と、命令キュー36A-36Bと、ロード/ストアユニット42と、実行コア40A-40Bと、PCサイロおよびリダイレクトブロック48とに接続される。命令キュー36A-36Bは互いに接続され、かつ、それぞれ対応の実行コア40A-40Bおよびレジスタファイル38A-38Bに接続される。レジスタファイル38A-38Bは互いに接続され、かつそれぞれ対応の実行コア40A-40Bに接続される。実行コア40A-40Bはさらに、ロード/ストアユニット42と、データキャッシュ44と、PCサイロおよびリダイレクトユニット48とに接続される。ロード/ストアユニット42は、PCサイロおよびリダイレクトユニット48と、Dキャッシュ44と、外部インターフェイスユニット46とに接続される。Dキャッシュ44はレジスタファイル38に接続され、外部インターフェイスユニット46は外部インターフェイス52に接続される。ここで文字が後に続く参照番号によって示される要素は、参照番号のみによって集合的に示される。たとえば、命令キュー36A-36Bは集合的に命令キュー36と称することにする。

30

40

50

【0023】

図1の実施例において、プロセッサ10は、可変バイト長、複雑命令セット計算(CISC)命令セットアーキテクチャを用いる。たとえば、プロセッサ10は、x86命令セットアーキテクチャ(IA-32とも称される)を用いてもよい。他の実施例では、固定長命令セットアーキテクチャおよび簡約化命令セット計算(RISC)命令セットアーキテクチャを含む他の命令セットアーキテクチャを用いてもよい。図1に示される特徴のいくつかはそのようなアーキテクチャにおいては省いてもよい。

【0024】

ラインプレディクタ12は、Iキャッシュ14のためのフェッチアドレスを生成するように構成され、さらに、命令動作のラインに関する情報をアライメントユニット16に与えるように構成される。一般的に、ラインプレディクタ12は、プロセッサ10によって前に投機的にフェッチされた命令動作のラインと、ラインのフェッチの際に選択すべき各ラインに対応する1つ以上の次のフェッチアドレスとを記憶する。一実施例では、ラインプレディクタ12は、1Kエントリを記憶するように構成され、その各々は命令動作の1つのラインを規定する。ラインプレディクタ12は、たとえば256エントリの4つのバンクにバンク分けされてもよく、その各々は所望であれば二重ポート(dual porting)なしに同時読出および更新を可能にするものである。

10

【0025】

ラインプレディクタ12は、次のフェッチアドレスをIキャッシュ14に与え、対応する命令バイトをフェッチするようにする。Iキャッシュ14は、命令バイトを記憶するための高速キャッシュメモリである。一実施例によれば、Iキャッシュ14は、たとえば、64バイトキャッシュラインを用いる256キロバイトの四重セット連想編成(four way set associative organization)を含んでいてもよい。しかしながら、どのIキャッシュ構造でも適当であろう。さらに、次のフェッチアドレスは入力としてラインプレディクタ12に戻され、命令動作の対応するラインに関する情報をフェッチするようにする。次のフェッチアドレスは、PCサイロおよびリダイレクトユニット48に報告される例外条件に回答してITB50により与えられるアドレスによってオーバーライドされ得る。

20

【0026】

ラインプレディクタによって与えられる次のフェッチアドレスは、(ラインが非分岐命令で終わる場合)ライン内の最終命令に連続するアドレスであり得る。代わりに、次のフェッチアドレスは、ラインの終わりにくる分岐命令のターゲットアドレスであってもよい。さらに別の選択肢では、ラインは戻り命令によって終わっていてもよく、この場合には次のフェッチアドレスは戻りスタック22から引出される。

30

【0027】

フェッチアドレスに回答して、ラインプレディクタ12は、フェッチアドレスにおいて始まる命令動作のラインに関する情報をアライメントユニット16に与える。アライメントユニット16は、Iキャッシュ14からフェッチアドレスに対応する命令バイトを受け、与えられた命令動作情報に応じて命令バイトを選択して1セットの発行位置とする。より特定的には、ラインプレディクタ12は、ライン命令動作内の各命令に対してシフト量をもたらし、かつ、ラインを構成する命令動作のセットに命令をマッピングする。1つの命令は複数の命令動作に対応していてもよく、このため、その命令に対応するシフト量は命令バイトを選択して複数の発行位置にするよう用いられてもよい。1つの発行位置は、ライン内の各々の可能な命令動作に対して与えられる。一実施例では、命令動作のラインは、最大6の命令に対応する最大8の命令動作を含み得る。一般的に、ここで用いられるように、命令動作のラインとは、復号ユニット24に同時に発行される命令動作の一群を指して言う。命令動作のラインは、マイクロプロセッサ10のパイプラインを通して1ユニットとしての命令キュー36へと進む。命令キュー36に記憶される際、個別の命令動作はいかなる順序で実行されてもよい。

40

【0028】

復号ユニット24(および、命令キュー36までの後のパイプラインステージ)内の発行

50

位置は、これらのパイプラインステージ内のハードウェアのためのライン内の命令動作のプログラム順序を規定する。アライメントユニット16によって発行位置に位置合わせされる命令動作は、命令キュー36A - 36B内に記憶されるまでその発行位置に留まる。したがって、第1の発行場所は、第1の発行場所内の命令動作がプログラム順序において同時に第2の発行位置内にある命令動作の前である場合に、第2の発行場所の前のものとして言及され得る。同様に、第1の発行場所は、第1の発行場所内の命令動作がプログラム順序において同時に第2の発行場所内にある命令動作の後である場合に、第2の発行場所の後のものとして言及され得る。また、発行場所内の命令動作は、ライン内の他の命令動作の前であるまたは後であるとして言及され得る。

【0029】

ここで用いられるように、命令動作（またはROP）は、実行コア40A - 40B内の実行ユニットが単一のエンティティとして実行するように構成されるところの動作のことである。単純な命令は単一の命令動作に対応し得るが、より複雑な命令は複数の命令動作に対応し得る。より複雑な命令のいくつかは、マイクロコードユニット28内でマイクロコードルーチンとして実現されてもよい。さらに、非CISC命令セットを用いる実施例では、各命令に対して単一の命令動作を用いてもよい（すなわち、命令と命令動作とはこのような実施例においては同義であり得る）。ある特定の一例では、ラインは、最大6の命令に対応する最大8の命令動作を含み得る。さらに、そのある特定の一例では、分岐命令が検出されるとラインを6の命令および/または8の命令動作未満で終わらせることがある。要望に応じて、ラインに対する命令動作に関するさらなる制限を行なってもよい。

【0030】

ラインプレディクタ12によって生成された次のフェッチアドレスは、分岐履歴テーブル18、間接アドレスキャッシュ20および戻りスタック22に経路付けられる。分岐履歴テーブル18は、次のフェッチアドレスによって識別されるラインを終わらせ得る条件付き分岐命令に対する分岐履歴をもたらす。ラインプレディクタ12は、分岐履歴テーブル18により与えられる予測を用いて、ラインを終わらせる条件付き分岐命令がテークン（taken）として予測されるべきであるかまたはノットテークン（not taken）として予測されるべきかを判定してもよい。一例では、ラインプレディクタ12は、テークンまたはノットテークンを選択するようを用いるべき分岐予測を記憶していてもよく、分岐履歴テーブル18は、ラインプレディクタの予測を打消しかつ異なる次のフェッチアドレスが選択されるようにし得る、より正確な予測をもたらすのに用いられる。間接アドレスキャッシュ20は、頻繁に変わる間接分岐ターゲットアドレスを予測するのに用いられる。ラインプレディクタ12は、次のフェッチアドレスとして、前に生成された間接ターゲットアドレスを記憶してもよい。間接アドレスキャッシュ20は、対応するラインが間接分岐命令によって終わる場合にはラインプレディクタ12によって与えられる次のフェッチアドレスをオーバーライドしてもよい。さらに、命令動作のライン内の最終命令の後のアドレスは、そのラインがサブルーチン呼出し命令によって終わる場合には戻りスタック22にプッシュされてもよい。戻りスタック22は、その一番上に記憶されるアドレスを、戻り命令によって終わるラインに対する潜在的な次のフェッチアドレスとしてラインプレディクタ12に与える。

【0031】

次のフェッチアドレスおよび命令動作情報を上記のブロックに与えることに加えて、ラインプレディクタ12は、次のフェッチアドレスおよび命令動作情報をPCサイロおよびリダイレクトユニット48に与えるように構成される。PCサイロおよびリダイレクトユニット48は、フェッチアドレスおよびライン情報を記憶し、例外の際に命令フェッチをリダイレクトすることおよび命令の順序正しいリタイアメントを行なうことの実行責任がある。PCサイロおよびリダイレクトユニット48は、プロセッサ10内で未決着（outstanding）であり得る命令動作の複数のラインに対応するフェッチアドレスおよび命令動作情報を記憶するための循環バッファ（circular buffer）を含んでいてもよい。命令のライン

10

20

30

40

50

をリタイアする際、PCサイロおよびリダイレクトユニット48は、それぞれ、条件付き分岐および間接分岐の実行に応じて分岐履歴テーブル18および間接アドレスキャッシュ20を更新してもよい。例外を処理する際、PCサイロおよびリダイレクトユニット48は、例外を起こす命令の後の戻りスタック22からのエントリをページしてもよい。また、PCサイロおよびリダイレクトユニット48は、例外を起こす命令の指標をマップユニット30、命令キュー36およびロード/ストアユニット42に経路付け、これによって、これらのユニットが例外を起こす命令の後の命令を取消し、かつこれに応じて投機的状態を回復し得るようにする。

【0032】

一実施例では、PCサイロおよびリダイレクトユニット48は、各命令動作にシーケンス番号(R)を割当て、プロセッサ10内で未決着である命令動作の順序を識別する。PCサイロおよびリダイレクトユニット48は、ラインを有する各々の可能な命令動作にRを割当ててもよい。ラインが最大数の命令動作より少ない数を含んでいる場合、割当てられたRのいくつかはそのラインに対して用いられないことになる。しかしながら、PCサイロおよびリダイレクトユニット48は、Rの次のセットを命令動作の次のラインに割当てるように構成されていてもよく、これによって、割当てられたが使用されていないRが、命令動作の対応するラインがリタイアされるまで使用されないままとなる。この態様で、所与のラインに割当てられるRの一部は、プロセッサ10内のラインを識別するように用いられ得る。一実施例では、最大8のROPがラインに割付けられ得る。したがって、各ライン内の第1のROPには、8の倍数であるRが割当てられ得る。使用されないRは、適宜自動的にスキップされる。

【0033】

前述の説明では、ラインプレディクタ12が次のアドレスを予測し、命令動作のラインに対する命令動作情報を与えることを説明した。この動作は、各フェッチアドレスがラインプレディクタ12においてヒットする限り行なわれる。ラインプレディクタ12においてミスが検出された際には、アライメントユニット16は対応する命令バイトをIキャッシュ14からプレディクタミス復号ユニット26に経路付ける。プレディクタミス復号ユニット26は、欠けているフェッチアドレスによって特定されるオフセットにおいて始まる命令を復号化し、命令動作のラインおよび次のフェッチアドレスを生成する。プレディクタミス復号ユニット26は、プロセッサ10がその目的で設計されているところのどんな制限をも命令動作のラインに強要する(たとえば、命令動作の最大数、命令の最大数、分岐命令に対する終了など)。あるラインの復号化を完了する際、プレディクタミス復号ユニット26は、その情報をラインプレディクタ12に記憶するため与える。プレディクタミス復号ユニット26が、命令が復号化されるにつれ命令をディスパッチするように構成されてもよいことに注意されたい。代わりに、プレディクタミス復号ユニット26は、命令情報のラインを復号化して、これをラインプレディクタ12に記憶するため与えてもよい。その後、欠けているフェッチアドレスはラインプレディクタ12において再度試みられ、ヒットが検出されることもある。さらに、ラインプレディクタ12におけるヒットが検出され、Iキャッシュ14におけるミスが起こることがある。対応する命令バイトは外部インターフェイスユニット46を介してフェッチされ、Iキャッシュ14に記憶されてもよい。

【0034】

一実施例では、ラインプレディクタ12およびIキャッシュ14は物理アドレス指定(physical addressing)を用いる。しかしながら、例外を検出した際に、PCサイロおよびリダイレクトユニット48は論理(または仮想)アドレスが供給されることになる。したがって、リダイレクトアドレスは、ラインプレディクタ12への表示のためにITB50によって変換される。また、PCサイロおよびリダイレクトユニット48は、相対分岐ターゲットアドレスなどPC相対計算において用いるための仮想先読みPC値を維持する。各ラインに対応する仮想先読みPCは、ITB50によって変換され、対応する物理アドレスがラインプレディクタ12によって生成される物理フェッチアドレスと一致すること

を確認する。不一致が起こった場合、ラインプレディクタ12は正しい物理アドレスで更新され、正しい命令がフェッチされる。PCサイロおよびリダイレクトユニット48はさらに、保護境界などを越えたフェッチに関連する例外を取扱う。PCサイロおよびリダイレクトユニット48はまた、最も最近リタイアされた命令のアドレスを示すリタイアPC値を維持する。

【0035】

復号ユニット24は、上述のとおり、複数の発行位置においてアライメントユニット16から命令動作を受けるように構成される。復号ユニット24は、並列して各発行位置に位置合わせされる命令バイトを復号化する(命令バイトに対応するどの命令動作がある特定の発行位置において生成されるべきであるかということを示す指標とともに)。復号ユニット24は、各命令動作に対して発信元および宛先オペランドを識別し、実行コア40A-40Bによって用いられる命令動作符号化を生成する。また、復号ユニット24は、マイクロコードで実現される命令に対してマイクロコードユニット28からマイクロコードルーチンをフェッチするように構成される。

10

【0036】

ある特定の一例によれば、プロセッサ10により以下の命令動作が支援されている。すなわち、整数、浮動小数点加算(マルチメディアを含む)、浮動小数点乗算(マルチメディアを含む)、分岐、ロード、ストアアドレス生成およびストアデータである。各命令動作は、最大で2つの発信元レジスタオペランドおよび1つの宛先レジスタオペランドを用い得る。ある特定の一例によれば、単一の宛先レジスタオペランドが整数ROPに割当てられ、整数結果および条件コード(condition code)(またはフラグ)更新の双方を記憶するようにしてもよい。対応する論理レジスタはともに、整数動作のリタイアメントの際に対応するPRを受けることになる。ある命令は、同じタイプの2つの命令動作を生成して2つの宛先レジスタを更新し得る(たとえば、ESPおよび特定された宛先レジスタを更新するROP)。

20

【0037】

復号化された命令動作と発信元および宛先レジスタ番号とはマップユニット30に与えられる。マップユニット30は、各命令動作の各々の宛先レジスタオペランドおよび発信元レジスタオペランドに物理レジスタ番号(PR)を割当てることによってレジスタ再命名を行なうように構成される。物理レジスタ番号はレジスタファイル38A-38B内のレジスタを識別する。さらに、マップユニット30は各命令動作にキュー番号(IQ)を割当て、命令動作を記憶するよう割当てられる命令キュー36A-36B内の場所を識別する。マップユニット30はさらに、命令動作の発信元オペランドに割当てられる各物理レジスタ番号を更新する命令のキュー番号を与えることによって各命令動作に対する従属性の指標をもたらす。マップユニット30は、物理レジスタ番号および命令により、マップサイロ32を、各命令動作に割当てられる番号(および対応する論理レジスタ番号)に更新する。さらに、マップサイロ32は、命令のラインの前の論理レジスタに対応する先読み状態と、PCサイロに関する命令のラインを識別するRとを記憶するように構成されてもよい。上述のPCサイロと同様に、マップサイロ32は、エントリの循環バッファを含んでいてもよい。各エントリは、命令動作の1ラインに対応する情報を記憶するように構成されていてもよい。

30

40

【0038】

マップユニット30およびマップサイロ32はさらに、PCサイロ48からリタイア指標を受けるとともに構成される。命令動作のラインをリタイアする際、マップサイロ32は、ラインに割当てられる宛先物理レジスタ番号および対応する論理レジスタ番号をアーキテクチャ的再命名ブロック34に伝達して記憶するようにする。アーキテクチャ的再命名ブロック34は、各論理レジスタに対応する物理レジスタ番号を記憶し、各論理レジスタに対するコミットされたレジスタ状態を表わす。新しい物理レジスタ番号での対応する論理レジスタの更新の際にアーキテクチャ的再命名ブロック34から変位される物理レジスタ番号は、後の命令への割付のために物理レジスタ番号の自由リストに戻される。一実施例

50

では、物理レジスタ番号を自由リストに戻す前に、その物理レジスタ番号はアーキテクチャ的再命名ブロック 3 4 内の残っている物理レジスタ番号と比較される。ある物理レジスタ番号が変位された後にも依然としてアーキテクチャ的再命名ブロック 3 4 内に表わされている場合、その物理レジスタ番号は自由リストに追加されない。このような実施例は、同じ物理レジスタ番号が 1 つより多い命令の結果を記憶するよう用いられる場合に用いてもよい。たとえば、x 8 6 命令セットアーキテクチャを用いる実施例では、浮動小数点オペランドを記憶するのに十分大きい物理レジスタをもたらしてもよい。このように、どんな物理レジスタをも、いかなるタイプのオペランドを記憶するのにも用いることができる。しかしながら、整数オペランドおよび条件コードオペランドは所与の物理レジスタ内の空間を完全に利用しない。そのような実施例では、プロセッサ 1 0 は、ある命令の整数結果および条件コード結果の双方を記憶するように単一の物理レジスタを割当ててもよい。物理レジスタに対応する条件コード結果を上書きする命令の後のリタイアメントによって同じ整数レジスタが更新されないこともあるため、物理レジスタは新しい条件コード結果をコミットする際に自由でないかもしれない。同様に、物理レジスタに対応する整数レジスタを更新する命令の後のリタイアメントにより、条件コードレジスタが更新されないこともあり、このため、物理レジスタは新しい整数結果をコミットする際に自由ではないかもしれない。

10

【 0 0 3 9 】

さらに、マップユニット 3 0 およびマップサイロ 3 2 は、P C サイロ 4 8 から例外指標を受けよう構成される。例外を起こす命令動作を含むラインの後の命令動作のラインは、マップサイロ 3 2 内で無効として印付けされる。命令動作の後のラインに対応する物理レジスタ番号は、リタイアメントのための対応するラインの選択の際に自由にされる（アーキテクチャ的再命名ブロック 3 4 は無効化された宛先レジスタで更新されない）。さらに、マップユニット 3 0 によって維持される先読みレジスタ状態は、例外を起こす命令に対応する先読みレジスタ状態に復元される。

20

【 0 0 4 0 】

命令動作のライン、発信元物理レジスタ番号、発信元キュー番号および宛先物理レジスタ番号は、マップユニット 3 0 によって割当てられるキュー番号に従って命令キュー 3 6 A - 3 6 B に記憶される。一実施例によれば、命令キュー 3 6 A - 3 6 B は対称であり、いかなる命令をも記憶できる。さらに、ある特定の命令動作に対する従属性は、いずれかの命令キューに記憶される他の命令動作に関して起こり得る。マップユニット 3 0 はたとえば、命令動作のラインを命令キュー 3 6 A - 3 6 B のうちの一方に記憶し、その後続く命令動作のラインを命令キュー 3 6 A - 3 6 B の他方に記憶し得る。命令動作は、少なくとも、命令動作が従属している前の命令動作が実行されレジスタファイル 3 8 A - 3 8 B を更新するまで命令キュー 3 6 A - 3 6 B 内に残る（そして命令動作は実行のためスケジューラされる）。一実施例では、命令動作はリタイアされるまで命令キュー 3 6 A - 3 6 B 内に残る。

30

【 0 0 4 1 】

命令キュー 3 6 A - 3 6 B は、ある特定の命令動作を実行のためにスケジューラされる際に、そのある特定の命令動作がどのクロックサイクルにおいてレジスタファイル 3 8 A - 3 8 B を更新することになるのかを定める。実行コア 4 0 A - 4 0 B 内の異なる実行ユニットが異なる数のパイプラインステージ（よって異なるレイテンシ）を用い得る。さらに、ある命令は他のものよりパイプライン内でより長いレイテンシを経験し得る。したがって、（クロックサイクルの数で）ある特定の命令動作に対するレイテンシを測定するカウントダウンが生成される。命令キュー 3 6 A - 3 6 B は、更新が起こるまで、特定された数のクロックサイクルだけ待ち、次に、そのある特定の命令動作に従属している命令動作をスケジューラしてもよいことを示す。各命令キュー 3 6 A - 3 6 B は、その命令キュー内の命令動作に対するカウントダウンを維持し、カウントダウンの満了の際に従属する命令動作をスケジューラすることを内部で許す。さらに、命令キューは、カウントダウンの満了の際に他方の命令キューに指標を与える。その後、他方の命令キューは従属する命令動

40

50

作をスケジュールすることができる。このような他方の命令キューへの命令動作完了の送信を遅延することにより、レジスタファイル 38A - 38B が、実行コア 40A - 40B の一方により与えられる結果を他方のレジスタファイルに伝搬することが可能となる。レジスタファイル 38A - 38B の各々は、プロセッサ 10 によって用いられる物理レジスタのセットを実装しており、実行コア 40A - 40B の一方によって更新される。この更新は次に、他方のレジスタファイルに伝搬される。命令キュー 36A - 36B は、ある命令を一旦その従属性が満足された（すなわちキュー内のその順序に関してアウトオブオーダーである）際にスケジュールし得ることに注意されたい。

【0042】

命令キュー 36A からスケジュールされる命令動作は、レジスタファイル 38A からの発信元物理レジスタ番号に従って発信元オペランドを読み出し、実行のために実行コア 40A に伝達される。実行コア 40A は、命令動作を実行し、レジスタファイル 38A 内の宛先に割当てられる物理レジスタを更新する。命令動作によっては宛先レジスタを有しておらず、実行コア 40A はこの場合には宛先物理レジスタを更新しない。さらに、実行コア 40A は命令動作の R および（もしあれば）命令動作に関する例外情報を PC サイロおよびリダイレクトユニット 48 に報告する。命令キュー 36B、レジスタファイル 38B および実行コア 40B も同様に動作し得る。

10

【0043】

一実施例では、実行コア 40A および実行コア 40B は対称のものである。各実行コア 40 はたとえば、浮動小数点加算ユニット、浮動小数点乗算ユニット、2つの整数ユニット、分岐ユニット、ロードアドレス生成ユニット、ストアアドレス生成ユニットおよびストアデータユニットを含んでいてもよい。実行ユニットの他の構成も可能である。

20

【0044】

宛先レジスタを有さない命令動作には、ストアアドレス生成、ストアデータ動作および分岐動作がある。ストアアドレス/ストアデータ動作は、ロード/ストアユニット 42 に結果をもたらす。ロード/ストアユニット 42 は、メモリデータ動作を行なうために D キャッシュ 44 にインターフェイスをもたらす。実行コア 40A - 40B は、命令のアドレスオペランドに基づいて、それぞれ、ロード ROP およびストアアドレス ROP を実行してロードおよびストアアドレスを生成する。より特定的には、ロードアドレスおよびストアアドレスは、その生成の際に（直接、実行コア 40A - 40B と D キャッシュ 44 との接続を介して）実行コア 40A - 40B によって D キャッシュ 44 に与えられてもよい。D キャッシュ 44 をヒットするロードアドレスにより、データが D キャッシュ 44 からレジスタファイル 38 に経路づけられることにつながる。一方、ヒットするストアアドレスはストアキューエントリを割付けられる。その後、ストアデータがストアデータ命令動作（これはストアデータをレジスタファイル 38A - 38B からロード/ストアユニット 42 へ経路づけるのに用いられる）によりもたらされる。ストア命令がリタイアされる際、データは D キャッシュ 44 内に記憶される。また、ロード/ストアユニット 42 は、（外部インターフェイス 46 を介して）後のキャッシュフィールドのために D キャッシュ 44 をミスするロード/ストアアドレスを記憶し、かつ欠けているロード/ストア動作を再度試みるためのロード/ストアバッファを含んでいてもよい。ロード/ストアユニット 42 はさらに、ロード/ストアメモリ従属性を取扱うよう構成される。

30

40

【0045】

次に、図 2 を参照して、マップユニット 30、マップサイロ 32 およびアーキテクチャ的再命名ブロック 34 の一実施例のブロック図が、プロセッサ 10 の一実施例によるその間の相互接続を強調して示される。要望に応じて、付加的な、置換のまたはより少ない相互接続を用いる他の実施例も可能であり考慮される。

【0046】

復号ユニット 24 は、ROP 情報バス 60 に接続され、ROP 情報バス 60 はさらに、マップユニット 30 およびマップサイロ 32 の双方に接続される。命令動作のライン（または ROP のライン）に関する情報が復号ユニット 24 によって ROP 情報バス 60 上にも

50

たらされる。ライン内の各 R O P に対し、復号ユニット 2 4 は少なくとも以下のもの、すなわち、有効指標と、R O P が宛先レジスタを書込むかどうかの指標と、R と、論理宛先レジスタ番号と、論理発信元レジスタ番号（最大 2 まで）をもたす。マップユニット 3 0 は宛先 I Q を各 R O P に割当て、宛先 P R を宛先レジスタを書込む各 R O P に割当てる。マップユニット 3 0 は、割当てられた P R および I Q をマップサイロ 3 2 へ宛先 P R / I Q バス 6 2 上に与える。さらに、マップユニット 3 0 は、現先読みレジスタ状態をマップサイロ 3 2 へ現先読みレジスタ状態バス 6 4 上に与える。一般的に、「先読みレジスタ状態」という用語は、プログラムシーケンスの実行におけるある特定の時点において（すなわち、プログラムシーケンスにおけるそのある特定の時点の前の各命令を実行した後およびプログラムシーケンスにおけるそのある特定の時点の後の各命令を実行する前）論理レジスタの状態（すなわちその中に記憶される値）を識別することを指して言う。現先読みレジスタ状態は、マップユニット 3 0 によって処理されている R O P のラインの前の論理レジスタに対応する物理レジスタのセットを識別する。換言すると、現先読みレジスタ状態は、各論理レジスタに対応する物理レジスタ番号を記憶する。さらに、本実施例において、現先読みレジスタ状態は、識別される物理レジスタを更新する命令の I Q と、I Q が依然として有効であるか（すなわち命令がまだリタイアされていないか）を示す有効ビットとを含む。マップサイロ 3 2 は、R O P のラインに対するエントリを割付け、現先読みレジスタ状態およびマップユニット 3 0 によって与えられる割当てられた P R および I Q を記憶する。また、マップサイロ 3 0 は、どの R O P が有効であり、どの R O P が論理レジスタを更新し、どの論理レジスタが R O P 情報バス 6 0 からこれらの R O P によって更新されるかを捕捉してもよい。例示的なマップサイロエントリが以下に示される（図 9）。

【 0 0 4 7 】

一般的に、ここで言及される「サイロ」は、命令、命令動作または命令動作のラインに対応する情報を記憶するための構造である。サイロは情報をプログラム順序に保ち、命令がプログラム順序においてリタイアされるにつれ（例外条件がない場合）情報はサイロの一番上（または末端）からサイロの一番下（またはヘッド）へと論理的に移動する。ここで用いられるように、命令がリタイアされるのは、命令の結果がアーキテクチャの状態にコミットされる際である（たとえば、命令の宛先に割当てられる物理レジスタ番号によりアーキテクチャ的再命名ブロック 3 4 の更新を可能にすることによって、または命令に対応するストアデータにより D キャッシュ 4 4 の更新を可能にすることによって）。

【 0 0 4 8 】

マップサイロ 3 2 は、リタイア有効ライン 6 6 上のリタイア有効信号と、例外情報バス 6 8 上の例外有効指標および R とを受けよう接続される。リタイア有効ライン 6 6 および例外情報バス 6 8 は P C サイロ 4 8 に接続される。アサートされたリタイア有効信号に応答して、マップサイロ 3 2 は、サイロのヘッドにおけるエントリからアーキテクチャ的再命名ブロック 3 4 へリタイアレジスタ / P R バス 7 0 上に、リタイアされたレジスタ情報を与える。より特定的には、リタイアレジスタ / P R バス 7 0 は、更新すべき論理レジスタ番号および対応する物理レジスタ番号を伝達してもよい。本実施例では、R O P のリタイアメントは完全なラインに対して同時に起こる（すなわち、P C サイロ 4 8 は、マップサイロ 3 2 および P C サイロ 4 8 のヘッドにおけるライン内の各 R O P が一旦実行するのに成功するとリタイアメントを信号で知らせる）。したがって、本実施例では最も古いラインをリタイアするための信号が用いられてもよい。他の実施例では、部分的リタイアメントに備えていてもよく、または個別の命令動作を介して記憶を編成してもよく、この場合、リタイアメントは命令動作などによって起こり得る。

【 0 0 4 9 】

アーキテクチャ的再命名ブロック 3 4 は、リタイアレジスタ / P R バス 7 0 上の特定される論理レジスタに対応するエントリを更新する前に、これらの論理レジスタに対応する現物理レジスタ番号を読み出す。換言すると、アーキテクチャ的再命名ブロック 3 4 から変位されている物理レジスタ番号（「前の物理レジスタ番号」）は、アーキテクチャ的再命

10

20

30

40

50

名ブロック34からポップされる。アーキテクチャ的再命名ブロック34は、前のPRを前のPRバス72上にもたらし、前のPRバス72はマップユニット30に接続され、リタイアレジスタ/PRバス70上にもたえられるPRにより、特定された論理レジスタエントリを更新する。

【0050】

一般的に、前のPRは、PRの自由リストに追加される資格がある(また、後のROPの宛先レジスタへの割当の資格がある)。しかしながら、本実施例では、プロセッサ10は、物理レジスタ共用技術を用いて、物理レジスタ使用の効率を向上する。たとえば、物理レジスタは、整数値および条件コード値(またはフラグ値)の双方を記憶するよう割当てられてもよい。物理レジスタ記憶部のある部分は整数値を記憶し、別の部分は条件コード値を記憶する。したがって、たとえば、PRが割当てられた整数レジスタの更新の際に前のPRがポップされた際に、PRはその中に記憶される条件コードを依然として表わし得る(またその逆もある)。アーキテクチャ的再命名ブロック34は、前のPRを更新されたアーキテクチャ的状态と比較して、どのレジスタが実際に自由にされる資格があるのかを判定する(図2において、PRを前のPRバス72から捕捉し、捕捉された番号をアーキテクチャ的再命名ブロック34に戻すレジスタ75によって表わされるが、他の実施例では1つのクロックサイクルにおいて更新および比較が行なわれてもよい)。たとえば、アーキテクチャ的再命名ブロック34は、論理レジスタに対応するPRを記憶するために連想メモリ(CAM: content addressable memory)を用いてもよい。アーキテクチャ的再命名ブロック34は、前のPRバス72上に伝達される各PRに対応してカム一致バス74上にカム一致信号を伝達してもよい。マップユニット30は、対応するカム一致信号がアサートされていない場合には前のPRバス72上に特定されるレジスタを自由にしてもよい。有利には、物理レジスタ使用はより効率のよいものであってもよく、なおかつ物理レジスタは正確に自由にされてもよい。他の考慮される実施例では、別個の物理レジスタが、命令動作に応答して更新される各論理レジスタに割当てられてもよいことに注意されたい。

【0051】

前のPRがアーキテクチャ的再命名ブロック34からポップされる際に自由にされない場合には、前のPRによって依然として表わされている論理レジスタを更新する命令のその後のリタイアメントによって、前のPRの自由化につながり得ることに注意されたい。その後のリタイアメントの際、カム一致は検出されないかもしれない。

【0052】

ここで用いられるように、物理レジスタが「自由」であるのは、再命名ハードウェアによって処理されている命令の宛先オペランドへの割当に利用可能である場合である。本実施例では、物理レジスタは、物理レジスタが割当てられる論理レジスタを更新する後の命令のリタイアメントの際に自由にされる。他の実施例では代替の態様でレジスタを自由にしてもよい。

【0053】

あるライン内の1つ以上の命令動作が同じ論理レジスタを更新し得ることに注意されたい。したがって、マップサイロ32またはアーキテクチャ的再命名ブロック34のうちの1つが、論理を含みリタイアされる論理レジスタを走査して各論理レジスタに対する最も古い更新を識別し(すなわち、プログラム順序において最終の更新)、その最も古い更新に対応する物理レジスタ番号をアーキテクチャ的再命名ブロック34に記憶する。より新しい更新は、上述の態様と同様に自由にされてもよい(すなわち、カムされ、一致が起らなければ自由にされる)。

【0054】

マップサイロ32はまたPCサイロ48から例外指標を受けてもよい。PCサイロ48は、例外有効信号をアサートし、例外を経験している命令動作のRをマップサイロ32に例外情報バス68を介して与えてもよい。マップサイロ32は、(ライン内の各ROPに対して一定であるRの部分を用いて)例外を経験している命令動作を含むROPのライ

10

20

30

40

50

ンに対応するサイロエントリを選択する。マップサイロ32は、選択されたエントリ内に記憶される現先読みレジスタ状態をマップユニット30へ回復先読みレジスタ状態バス76上に与える。マップユニット30は先読みレジスタ状態を回復された状態に復元する。さらに、マップサイロ32は、ライン内の、例外を経験するROPの前の、ROPの論理レジスタ番号、PR およびIQ をもたらず。マップユニット30は、もたらされたPR およびIQ により、復元された先読み状態を更新する。有利には、先読み状態は迅速に回復される。例外条件に回答してフェッチされる命令は、再命名の迅速な回復によりマップユニット30に到達する際に再命名されてもよい。

【0055】

さらに、例外に回答して、この例外を経験するROPに続いてROPに割当てられた物理レジスタは自由にされる。マップサイロ32は、マップユニット30への自由PR バス78上で自由にされるPR を伝達する。一実施例では、マップサイロ32は、自由にされるPR を1クロックサイクルごとに1ラインの割合で与えるよう構成され得る。さらに、物理レジスタが割当てられたROPはリタイアされなかったため、物理レジスタは、カムするためにアーキテクチャ的再命名ブロック32に伝達される必要はない。

【0056】

ここで図3を参照して、マップユニット30の一実施例のブロック図が示される。他の実施例も可能であり、企図される。図3の実施例では、マップユニット30は、レジスタ走査ユニット80、IQ / PR 制御ユニット82、先読みレジスタ状態84、仮想/物理レジスタマップユニット86、自由リスト制御ユニット88および自由リストレジスタ90を含む。レジスタ走査ユニット80は、バス60A(図2に示すROP情報バス60の一部)により復号ユニット24から発信元および宛先レジスタ番号(および各々に対する有効指標)を受けよう接続される。レジスタ走査ユニット80は、宛先レジスタ番号および発信元仮想レジスタ番号を仮想/物理レジスタマップユニット86に伝えるよう構成される。IQ / PR 制御ユニット82はバス60B(図2に示すROP情報バス60の一部)に接続され、宛先レジスタ番号およびその宛先レジスタ番号に対応する有効指標を受け。命令キュー36Aおよび36Bは末端ポインタバス92により末端ポインタを与え、各キューのいずれのエントリが現在そのキューの末端であることを示す。さらに、IQ / PR 制御ユニット82は宛先PR / IQ バス62に接続される。仮想/物理レジスタマップユニット86は、回復先読みレジスタ状態バス76、および先読みレジスタ状態84に接続され、これはさらに現先読みレジスタ状態バス64に接続される。またさらに、仮想/物理レジスタマップユニット86は、命令キュー36Aおよび36Bへの発信元/宛先PR およびIQ バス94によってライン内の各ROPに対して発信元PR 、発信元IQ 、宛先PR およびIQ を与えるよう接続される。自由リスト制御ユニット88は次の自由PR バス96および割当てられたPR バス99を介してIQ / PR 制御ユニット82に接続され、かつ自由リストレジスタ90に接続される。さらに、自由リスト制御ユニット88は前のPR バス72、カム一致バス74、および自由PR バス78に接続される。

【0057】

図3の実施例では、マップユニット30は2段階のパイプライン設計を用いてレジスタの再命名を行なう。第1段階では、レジスタ走査ユニット80が仮想レジスタ番号を各発信元レジスタに割当てて。並行して、IQ / PR 制御ユニット82はIQ を(命令キュー36Aおよび36Bによって与えられた末端ポインタに基づいて)各ROPに割当て、PR を宛先レジスタを有するROPに割当てて。第2段階では、仮想/物理レジスタマップユニット86は仮想レジスタ番号を物理レジスタ番号に(現先読み状態および割り当てられたPR に基づいて)マッピングし、IQ / PR 制御ユニット82によって割当てられた物理レジスタ番号を対応のROPの発行位置に経路づけする。

【0058】

レジスタ走査ユニット80によって割当てられた仮想レジスタ番号は、物理レジスタ番号に対する発信元を識別する。たとえば、本実施例では、発信元レジスタに対応する物理レ

10

20

30

40

50

ジスタ番号は先読みレジスタ状態 84 (前にマップユニット 30 により処理された ROP のラインに対応する更新を反映するもの)か、ROP のライン内の前の発行位置 (前の ROP の宛先オペランドが発信元オペランドと同じ、すなわちライン内従属性が存在する場合)のいずれかから引出され得る。言換えれば、発信元レジスタ番号に対応する物理レジスタ番号は、ライン内従属性が検出されない限り、先読みレジスタ状態 84 により維持される物理レジスタ番号である。レジスタ走査ユニット 80 はライン内従属性検査を効果的に行なう。他の実施例により、所望に応じて他の発信元オペランドの発信元を与えることもできる。

【0059】

物理レジスタ番号のマッピングからのライン内従属性検査 / 宛先物理レジスタ割当てを複数のパイプライン段階に分けることにより、各段階はより高い周波数で動作され得る。したがって、図 3 に示すマップユニット 30 の実施例は、発信元物理レジスタ番号の決定と並行してライン内従属性検査および宛先物理レジスタ割当てを行なう他の実施例と比べて、より高い周波数で動作することが可能である。仮想レジスタ番号を用いることにより、機能の分離が可能になり、また図 8 を参照して以下に説明するように、発信元物理レジスタ番号の比較的簡単かつ効率的なマッピングが可能になる。

【0060】

IQ / PR 制御ユニット 82 は、命令キュー 36A および 36B のうち 1 つの末端ポイントで始まる命令キュー番号を割当てする。言換えれば、ライン内の第 1 の ROP は選択された命令キューの末端ポイントを IQ として受け、その他の ROP は末端ポイントから増加する順に IQ を受ける。制御ユニット 82 はあるライン内の ROP の各々を同じ命令キュー 36A および 36B に割当て、ROP の次のラインを他の命令キュー 36A および 36B に割付ける。制御ユニット 82 は命令キュー 36A および 36B に割付けられた ROP の番号の指標を ROP 割付けバス 98 を介して伝達する。これにより、受ける側の命令キューは、そのキューに対する ROP の割付けを反映するようにその末端ポイントを更新し得る。

【0061】

制御ユニット 82 は自由リスト制御ユニット 88 から自由 PR のセットを受ける。この自由 PR のセットは、命令動作のライン内の宛先レジスタに割当てられる。一実施例では、プロセッサ 10 は論理レジスタの更新数を 1 ラインから 4 ライン内に制限する (すなわち、プレディクタミス復号ユニット 26 が 5 番目の論理レジスタの更新に遭遇すると、ラインはその前の命令において終わる)。したがって、自由リスト制御ユニット 88 は自由リスト 90 から 4 つの PR を選択し、その選択されたレジスタを次の自由 PR バス 96 によって制御ユニット 82 に伝達する。制御ユニット 82 は、割当てられた PR バス 99 を介していずれの PR が実際に割当てられたかということで応答し、自由リスト制御ユニット 88 は割当てられた物理レジスタをこの自由リストから削除する。他の実施例では、ライン内の更新の数に対してさまざまな制限が採用され、これは全く制限されないもの (すなわち各 ROP が更新され得るもの) も含み得る。

【0062】

自由リスト制御ユニット 88 は、物理レジスタの自由化を管理し、かつ後の命令に割当てするためにレジスタを選択するよう構成される。自由リストレジスタ 90 は、たとえば、各物理レジスタに対応するあるビットを記憶し得る。このビットがセットされていると、対応するレジスタが自由にされる。このビットがクリアされていると、対応するレジスタは現在割当てられている (すなわち自由でない) ということである。自由リスト制御ユニット 88 は自由リストを走査し、制御ユニット 82 に伝達するためのレジスタを選択する。たとえば、自由リスト制御ユニット 88 は自由リストレジスタ 90 の各端部から初めの 2 つの自由レジスタについて走査し、本実施例において設けられた 4 つのレジスタの迅速な選択を可能にし得る。これらの走査は、各端部からの 2 つのピック・ワン動作 (pick one operations) (他方の前に行なわれ、割当てられた物理レジスタを自由リストから取除く動作) として行なわれ得る。

10

20

30

40

50

【 0 0 6 3 】

自由リスト制御ユニット 8 8 は前の P R バス 7 2 を介してアーキテクチャ的再命名ブロック 3 4 からポップされた前の物理レジスタ番号を受ける。その後、前の物理レジスタ番号の各々に対応するカム一致信号がカム一致バス 7 4 上で受けられる。対応のカム一致信号がデアサートされる、前の P R の各々が、自由リスト制御ユニット 8 8 によって自由リストに加えられる。さらに、自由 P R バス 7 8 上で受けられた物理レジスタ番号は無条件で自由リストに加えられる。

【 0 0 6 4 】

先読みレジスタ状態 8 4 は、仮想 / 物理レジスタマップユニット 8 6 に呈される R O P のラインに対応する更新の前に、先読みレジスタ状態に記憶される。より具体的には、先読みレジスタ状態 8 4 が、各論理レジスタに対応する物理レジスタ番号と、(本実施例では)宛先レジスタとして割当てられた物理レジスタ番号を有する R O P に対応する命令キュー番号とを記憶する。各クロックサイクルで、先読みレジスタ状態 8 4 は現先読みレジスタ状態を現先読みレジスタ状態バス 6 4 によりマップサイロ 3 2 に伝達する。仮想 / 物理レジスタマップユニット 8 6 は、P R の発信元が先読みレジスタ状態 8 4 であることを示す仮想レジスタ番号を有する各発信元レジスタに対して先読みレジスタ状態 8 4 により示されるような対応の論理レジスタの P R および I Q を供給する。仮想レジスタ番号が前の発行位置を示す発信元レジスタには、制御ユニット 8 2 により割当てられた対応の P R および I Q が供給される。さらに、仮想 / 物理レジスタマップユニット 8 6 は、R O P のラインによって特定された論理宛先レジスタと制御ユニット 8 2 により割当てられた宛先 P R / I Q とに従って、先読みレジスタ状態 8 4 を更新する。

【 0 0 6 5 】

仮想 / 物理レジスタマップユニット 8 6 はさらに、例外条件 (上記) に応答して、回復先読みレジスタ状態バス 7 6 上でマップサイロ 3 2 により与えられる回復先読みレジスタ状態を受けよう構成される。仮想 / 物理レジスタマップユニット 8 6 は、レジスタ走査ユニット 8 0 および I Q / P R 制御ユニット 8 2 からの入力に従って生成された次の先読みレジスタ状態をマップサイロ 3 2 により与えられた回復先読み状態でオーバライドし得る。

【 0 0 6 6 】

なお、本実施例では、I Q は各発信元オペランドに対して経路づけられ、いずれの命令キューエントリに対応の R O P が従属しているかを示す。命令キュー 3 6 A および 3 6 B は、対応する命令キューエントリにおける R O P の完了を待った後、従属 R O P をスケジューリングして実行する。

【 0 0 6 7 】

次に図 4 を参照して、レジスタ走査ユニット 8 0 の一実施例を表わすブロック図が示される。他の実施例も可能であり企図される。図 4 の実施例では、レジスタ走査ユニット 8 0 は、R O P のライン内の各発行位置に対応する走査ユニットを含む。たとえば、本実施例は、8 つの発行位置に対応する 8 つの走査ユニット 1 0 0 A ~ 1 0 0 H を含む (これより多いものまたは少ないものも実現され得るが) 。走査ユニット 1 0 0 A ~ 1 0 0 H は仮想先読みレジスタ状態を伝えるための連続鎖として、また発信元および宛先レジスタ番号を受けよう並列で接続される。

【 0 0 6 8 】

各走査ユニット 1 0 0 A ~ 1 0 0 H は、図 3 に示す R O P バス 6 0 A の一部に接続される。より具体的には、各走査ユニット 1 0 0 A ~ 1 0 0 H は、対応する発行位置における R O P の発信元および宛先レジスタ番号を受けよう接続される。したがって、バス 6 0 A A は走査ユニット 1 0 0 A に接続され、発行位置ゼロ (すなわち R O P のライン内のプログラム順序で 1 番目の R O P) に対して発信元および宛先レジスタ番号を与える。同様に、バス 6 0 A B は走査ユニット 1 0 0 B に接続され、発行位置 1 に対して発信元および宛先レジスタ番号を与える。その他のバス 6 0 A C ~ 6 0 A H は、図示したように、順に残りの発行位置に対応する発信元および宛先レジスタ番号を与える。各走査ユニット 1 0

10

20

30

40

50

0A～100Hは、各発信元レジスタに対して発信元仮想レジスタ番号を与えるように構成され、この番号は後で仮想/物理レジスタマップユニット86に伝えられる。

【0069】

走査ユニット100Aは仮想現先読みレジスタ状態を受けるとして結合される。この仮想現先読みレジスタ状態は、各論理レジスタに対応する仮想レジスタ番号を含む。仮想現先読み状態内の仮想レジスタ番号は、PR（およびIQ）の発信元が先読みレジスタ状態84であることを示す。x86命令セットアーキテクチャを採用する一実施例において、たとえば、先読みレジスタ状態84は、アーキテクテッド整数レジスタに対応する8つのレジスタと、条件コードビットの群（群分けは、さまざまな命令によってどれが更新されるかに従って選択され、たとえば0ビットを1つの群とし、Cビットを別の群、さらにZ, A, PおよびSビットを第3の群とする）に対応する3つのレジスタと、アーキテクテッド浮動小数点/MMXレジスタに対応する8つのレジスタと、浮動小数点条件コードレジスタに対応する1つのレジスタと、一時マイクロコードレジスタに対応する8つのレジスタとを含む。

10

【0070】

走査ユニット100Aは発信元レジスタ番号に基づいて仮想現先読みレジスタ状態からの発信元仮想レジスタ番号を割当てる。さらに、発行位置ゼロのROPが宛先レジスタを更新すると、走査ユニット100Aは対応する論理宛先レジスタに対して新しい仮想レジスタ番号を挿入することにより仮想現先読み状態を更新する。この新しい仮想レジスタ番号は、対応する論理レジスタについてのPR およびIQ の発信元が発行位置ゼロであることを示す。元の仮想レジスタ番号の代わりに新しい仮想レジスタ番号が挿入された更新先読みレジスタ状態が、走査ユニット100Bに伝えられる。

20

【0071】

走査ユニット100Bは走査ユニット100Aから更新された先読みレジスタ状態を受取り、更新された先読みレジスタ状態からの仮想レジスタ番号を発信元レジスタ番号に割当てる。さらに、走査ユニット100Bは、発行位置1におけるROPが論理レジスタを更新すると、走査ユニット100Aにより与えられる更新された先読みレジスタ状態へ発行位置1を示す新しい仮想レジスタ番号を挿入する。走査ユニット100C～100Hもまた同様に、先行する走査ユニットにより与えられる更新された先読みレジスタ状態に回答して、発行位置3～8におけるROPの発信元レジスタに対してそれぞれ仮想レジスタ番号を割当て、宛先レジスタ番号がもしあればそれに従って、更新先読みレジスタ状態を更新する。

30

【0072】

したがって、ライン内の前のROPがROPの発信元レジスタ番号により特定された論理レジスタを更新すると、前の発行位置を示す仮想レジスタ番号が割当てられる。そうでなければ、対応する論理レジスタについての現先読みレジスタ状態を示す仮想レジスタ番号が割当てられる。言換えれば、前の命令ラインに従属する（現先読みレジスタ状態を介した）ライン内従属は、仮想レジスタ番号により示される。

【0073】

走査ユニット100Hにより与えられたこの更新先読みレジスタ状態は次の仮想先読みレジスタ状態であり、これは、走査ユニット100A～100Hの各々により与えられる発信元仮想レジスタ番号および宛先レジスタ番号とともに、仮想/物理レジスタマップユニット86に伝達される。仮想/物理レジスタマップユニット86はその後、ROPの後のラインに対して先読みレジスタ状態84を更新するために、ROPのラインに対応する次の先読みレジスタ状態を生成し得る。

40

【0074】

なお、一実施例では、走査ユニット100A～100Fの各々は、整数レジスタ、一時レジスタおよび条件コードレジスタを取扱う整数/一時走査回路と、浮動小数点レジスタおよび浮動小数点条件コードレジスタを取扱う浮動小数点走査回路とを含む。したがって、各発信元および宛先レジスタが浮動小数点レジスタであるか整数レジスタであるかの指標

50

は、レジスタ番号とともに経路づけられ、また、どの仮想レジスタ番号をレジスタに割当てるか、または新しい仮想レジスタ番号で置換えるかを決定するのに用いられる。

【 0 0 7 5 】

次に図 5 を参照して、仮想レジスタ番号の符号化の例を表 1 0 2 が示される。他の符号化も可能であり、企図される。具体的には、たとえば、表 1 0 2 に示す最上位ビット (M S B) の論理状態は表に示すものから反転され得る。また他の符号化も同様に可能である。

【 0 0 7 6 】

表 1 0 2 は仮想レジスタ番号の符号化を表わし、ここで M S B が、対応する P R / I Q に対する発信元が先読みレジスタ状態 8 4 によって維持される現先読み状態であるか、ライン内の前の発行位置に割当てられた宛先 P R / I Q であるかを決定する。たとえば、M S B がクリアされていれば、仮想レジスタ番号は発信元が現先読み状態であることと、仮想レジスタ番号の最下位ビット (L S B) は現先読み状態内の論理レジスタ番号であることを示している。M S B がセットされていると、仮想レジスタ番号は発信元が前の発行位置であることを示している。この場合の L S B は前の発行位置番号となる。

【 0 0 7 7 】

表 1 0 2 に示す符号化を採用する一実施例では、走査ユニット 1 0 0 A に与えられた仮想現先読み状態は、M S B がクリアされており、与えられる対応の論理レジスタ番号が L S B である、符号化を含む。走査ユニット 1 0 0 A ~ 1 0 0 H により更新先読みレジスタ状態中に挿入された新たな仮想レジスタ番号は、M S B がセットされており、挿入する走査ユニットに対応する発行位置番号が L S B で与えられる、符号化を含む。

【 0 0 7 8 】

次に図 6 を参照して、走査ユニット 1 0 0 A ~ 1 0 0 H の各々の一実施例において採用され得る、整数 / 一時走査回路 1 1 0 の一実施例の一部が示される。他の実施例も可能であり、企図される。図 6 の実施例では、対応する R O P の宛先レジスタ番号は宛先バス 1 1 2 によって伝達され、対応する R O P の第 1 の発信元 (S R C 1) レジスタ番号は S R C 1 バス 1 1 4 により伝達され、また対応する R O P の第 2 の発信元 (S R C 2) レジスタ番号は S R C 2 バス 1 1 6 により伝達される。バス 1 1 2、1 1 4 および 1 1 6 はバス 6 0 A の一部を含み、(たとえば、整数 / 一時走査回路 1 1 0 が走査ユニット 1 0 0 A の一部であれば、バス 1 1 2、1 1 4 および 1 1 6 はバス 6 0 A の部分となる)。宛先バス 1 1 2 は複数の宛先識別子ユニット (たとえば図 6 に示すユニット 1 1 8 A および 1 1 8 B) に接続される。宛先識別子ユニット 1 1 8 A および 1 1 8 B の各々は伝達 / 書込 (p a s s / w r i t e) 制御ユニット (たとえば宛先識別子ユニット 1 1 8 A および 1 1 8 B にそれぞれ接続された伝達 / 書込制御ユニット 1 2 0 A および 1 2 0 B) に接続される。各伝達制御ユニット 1 2 0 A および 1 2 0 B はさらに、先行する走査ユニット 1 0 0 A ~ 1 0 0 H により与えられた仮想レジスタ入力と、後続の走査ユニット 1 0 0 A ~ 1 0 0 H への仮想レジスタ出力とに接続される (たとえば、伝達 / 書込制御ユニット 1 2 0 A は仮想 E A X 入力バス 1 2 2 A および仮想 E A X 出力バス 1 2 4 A に接続され、伝達 / 書込制御ユニット 1 2 0 B は仮想 E B X 入力バス 1 2 2 B および仮想 E B X 出力バス 1 2 4 B に接続される)。さらに、S R C 1 バス 1 1 4 は、複数の S R C 1 識別子ユニット (たとえば図 6 に示す S R C 1 識別子ユニット 1 2 6 A および 1 2 6 B) に接続される。S R C 2 バス 1 1 6 は、同様に、複数の S R C 2 識別子ユニット (たとえば図 6 に示す S R C 2 識別子ユニット 1 2 8 A および 1 2 8 B) に接続される。S R C 1 / S R C 2 識別子ユニット 1 2 6 A および 1 2 6 B または 1 2 8 A および 1 2 8 B の各々は、図 6 に示す対応のスイッチ 1 3 0 A から 1 3 0 D のイネーブル制御に接続される。スイッチ 1 3 0 A および 1 3 0 B は仮想 E A X 入力バス 1 2 2 A に接続された入力を有し、一方スイッチ 1 3 0 C および 1 3 0 D は仮想 E B X 入力バス 1 2 2 B に接続された入力を有する。スイッチ 1 3 0 A および 1 3 0 C は S R C 1 仮想レジスタバス 1 3 2 A に接続された出力を有し、一方スイッチ 1 3 0 B および 1 3 0 D は S R C 2 仮想レジスタバス 1 3 2 B に接続された出力を有する。

【 0 0 7 9 】

10

20

30

40

50

通常、整数ノ一時走査回路110は対応の発行位置におけるROPの整数ノ一時発信元レジスタに対して仮想レジスタ番号を割当て、対応する発行位置における整数宛先レジスタに対して新たな仮想レジスタ番号を挿入する。宛先識別子ユニット、伝達ノ書込制御ユニット、SRC1識別子ユニットおよびSRC2識別子ユニットは、論理整数、一時または条件コードレジスタの1つに対応する。たとえば、宛先識別子ユニット118A、伝達ノ書込制御ユニット120A、SRC1識別子ユニット126AおよびSRC2識別子ユニット128AはEAXレジスタに対応する。同様に、宛先識別子ユニット118B、伝達ノ書込制御ユニット120B、SRC1識別子ユニット126BおよびSRC2識別子ユニット128BはEBXレジスタに対応する。宛先識別子ユニット、伝達ノ書込制御ユニット、SRC1識別子ユニットおよびSRC2識別子ユニットの他のセット(図示せず)は、論理整数レジスタ、一時レジスタまたは条件コードレジスタの他のものに対応する。以下に、EAXレジスタハードウェアについて述べる。EBXレジスタハードウェアはEBXレジスタと同様に動作し、他の同様のハードウェア(図示せず)は残りのレジスタに対して同様に動作する。

10

【0080】

宛先識別子ユニット118Aは宛先レジスタバス112上の宛先レジスタ番号がEAXレジスタを選択するか否かを判定する。したがって、宛先識別子ユニット118AはEAXが選択されたか否かを判定するために宛先レジスタ番号を復号化し、この復号化は、宛先レジスタ番号が有効であると示す有効信号と、宛先レジスタ番号が整数ノ一時ノ条件コードレジスタである(すなわち浮動小数点レジスタではない)ことを示す整数信号とで分類される(qualified)。宛先レジスタがEAXレジスタであれば、宛先識別子ユニット118Aは伝達ノ書込制御ユニット120Aに信号で知らせ、整数ノ一時走査回路110が仮想EAX出力バス124A上で採用される発行位置に対応する仮想レジスタ番号を挿入する。そうでなければ、宛先識別子ユニット118Aは伝達ノ書込制御ユニット120Aに、仮想EAX入力バス122Aにより仮想EAX出力バス124Aへと与えられた仮想レジスタ番号を伝えるよう信号で知らせる。

20

【0081】

同様に、SRC1識別子ユニット126Aは、SRC1レジスタバス114上のSRC1レジスタ番号がEAXが選択されたか否かを判定するためにSRC1レジスタ番号を復号化し、かつこの復号化を、SRC1レジスタ番号が有効であることを示す有効信号とSRC1レジスタ番号が整数ノ一時ノ条件コードレジスタであることを示す整数信号とで分類することにより、SRC1レジスタバス114上のSRC1レジスタ番号がEAXレジスタを選択するか否かを決定する。EAXがSRC1であると選択されると、SRC1識別子ユニット126Aはスイッチ130Aを活性化し、仮想EAX入力バス122AによりSRC1仮想レジスタバス132A上に与えられる仮想レジスタ番号を駆動する。SRC識別子ユニット128AはSRC1識別子ユニット126Aと同様であるが、SRC2レジスタバス116上に与えられたSRC2レジスタ番号において動作し、これに応じてSRC2仮想レジスタバス132Bを駆動するようにまたは駆動しないようにスイッチ130Bを制御する。

30

【0082】

この形式で、更新先読み状態が次の走査ユニットへ(124Aおよび124Bなどの仮想出力バスを介して)伝えられることができ、また発信元仮想レジスタ番号が(SRC1仮想レジスタバス132AおよびSRC2レジスタ番号バス132Bを介して)割当てられ得る。なお、整数ノ一時走査回路110は、仮想整数ノ一時入力の各々を伝達ノ書込制御ユニット120A(および他の伝達ノ書込制御ユニット)に経路づけることにより、レジスタ間の移動演算(move operations)を取扱うように変更され得る。現発行位置におけるROPがレジスタ間の移動であることを示すために、さらなる信号が送られ得る。SRC1レジスタ番号および移動信号(move signal)にตอบสนองして、SRC1レジスタ番号に対応する仮想整数ノ一時入力は、宛先レジスタに対応する仮想整数ノ一時出力へと経路づけられる。さらに、対応する発行位置におけるROPは命令キュー36Aおよび36Bへ

40

50

伝達することが禁止され(たとえばその有効ビットがリセットされ)、これは、発信元仮想レジスタ番号を宛先仮想レジスタ番号として経路づけることによりこの移動が達成されるからである。このような実施例において、リタイアした物理レジスタを自由にする前にそれらをカムすることにより、宛先PR を不注意で早く自由化してしまうのを防ぐことができる。

【0083】

ここで図7を参照して、走査ユニット100A~100Hの各々の一実施例において採用され得る、浮動小数点走査回路140の一実施例の一部が示される。この他の実施例も可能であり、企図される。走査回路140は、整数/一時操作回路110と同様、SRC1レジスタバス114およびSRC2レジスタバス116を受け、SRC1仮想レジスタバス132AおよびSRC2仮想レジスタバス132B上で仮想レジスタ番号を駆動し得る。図7に示す部分は仮想stI入力(x86命令セットアーキテクチャにおいて規定されたst0~st7のレジスタのうちの1つであり、 $0 < I < 7$)を受け、仮想stI出力を回路140が採用される発行位置「現発行位置」に対して与える。SRC1レジスタバス114に接続されたSRC1識別子ユニット142Aは、SRC1レジスタ番号がstIレジスタを選択しているか否か(すなわちレジスタ番号がstIであり、レジスタが浮動小数点レジスタであり、さらにSRC1レジスタが有効であるか否か)を判定する。SRC1レジスタ番号がstIレジスタを選択している場合、SRC1識別子ユニット142Aは、仮想stI入力バス144によりSRC1仮想レジスタバス132A上へ与えられた仮想レジスタ番号を駆動するようスイッチ130Eを制御する。同様に、SRC2レジスタバス116に接続されたSRC2識別子ユニット142Bは、SRC2レジスタ番号がstIレジスタを選択しているか否かを判定し、仮想stI入力バス144によりSRC2仮想レジスタバス132Bに与えられた仮想レジスタ番号を駆動するようスイッチ130Fを選択的に制御する。

【0084】

さらに、図7に示す浮動小数点走査回路114の部分は、仮想stI出力バス146上のレジスタstIに対応する出力仮想レジスタ番号を与える。さまざまな入力仮想レジスタ番号が出力バス146上の出力仮想レジスタ番号として選択可能であり、これは復号ユニット24により与えられたさまざまな制御信号によって制御される。x86浮動小数点命令はこの浮動小数点レジスタセットをスタックとして扱う。st0はこのスタックの一番上にあるレジスタであり、st1は一番上の次にくるものである、などとなる。特定の命令により、スタックはプッシュされる(現st0レジスタをst1レジスタにするなど)またはポップされる(現st1レジスタをst0レジスタにするなど)。またさらに、交換命令がサポートされ、これがスタックレジスタの一番上(st0)とその他のレジスタのうちの1つとをスワップする。

【0085】

この仮想stI出力の選択により、スイッチ130Gから130Kおよび入力仮想レジスタバス144、148、150、152および154のセットを採用することによりこれらのシチュエーションの多くを取扱う試みがなされる。現発行位置にあるROPが浮動小数点スタックまたはstIレジスタに個別に影響を与えなければ、この伝達信号はスイッチ130G(仮想stI入力バス144と仮想stI出力バス146との間に接続されている)にアサートされ、仮想stI入力が仮想stI出力として与えられる。一方、現発行位置にあるROPがstIレジスタを更新すれば、書込信号がスイッチ130J(バス152とバス146との間に接続されている)にアサートされ、この現発行位置(バス152により伝達される)に対応する仮想レジスタ番号は仮想stI出力バス146上で送信される。現発行位置におけるROPがスタックプッシュまたはポップを起こすと、対応する信号がスイッチ130K(バス154とバス146との間に接続されている)および130I(バス152とバス146との間に接続されている)にそれぞれアサートされる。現発行位置に与えられる更新先読み状態内のstI-1およびstI+1レジスタに対応する仮想レジスタ番号はこれにより与えられる。最後に、現発行位置の2つ前の発行位

10

20

30

40

50

置から入力された仮想 $st0$ はバス 148 により与えられる。現 ROP が FEXC 命令の後半部分であれば、EXCH 信号がスイッチ 130H にアサートされ、現発行位置の 2 つ前の発行位置からの $st0$ に対応する仮想レジスタ番号は仮想 stI 出力として選択される。

【0086】

なお、FEXC 命令の前半を取扱うために、 $st0$ に対応する浮動小数点走査回路 140 の部分は前の走査ユニットからの仮想 stI 入力の各々を含み、いずれかのレジスタを仮想 $st0$ 出力として任意に選択することができる。さらに、浮動小数点状態レジスタのトップ・オブ・スタック (TOS) フィールドおよび浮動小数点タグワードも同様に浮動小数点操作により影響される。TOS およびタグワードについての先読み値も同様にプッシュ、ポップおよび交換を通じて伝搬され得る。TOS およびタグワードの現先読みコピーは、先読みレジスタ状態 84 に維持され得る。さらに、各発行位置に対応する TOS およびタグワードの値は、例外回復のためにマップサイロ 32 に記憶され得る (そして最後の値が先読みレジスタ状態 84 に更新され得る)。またさらに、レジスタ走査ユニット 80 は (タグワードで示されるように) 無効であるレジスタの使用を検出し、かつ後に例外を取扱うためにレジスタを用いた ROP による例外に気づくことができる。

【0087】

なお、整数 / 一時走査回路 110 および浮動小数点走査回路 140 は、整数から浮動小数点への、および浮動小数点から整数への移動を同様に取扱うことができる。ROP の発信元レジスタが整数であると示されれば、整数 / 一時走査回路 110 は発信元仮想レジスタ番号を与える。一方、ROP の発信元レジスタが浮動小数点であると示されれば、浮動小数点走査回路 140 が発信元仮想レジスタ番号をもたらず。宛先レジスタが浮動小数点であると示されれば、新しい浮動小数点仮想レジスタ番号が浮動小数点走査回路 140 により更新先読みレジスタ状態中に与えられる。一方、宛先レジスタが整数であると示されれば、新しい整数仮想レジスタ番号が整数 / 一時走査回路 110 によって更新先読みレジスタ状態中に与えられる。したがって、浮動小数点から整数レジスタへの移動を有する発行位置には、浮動小数点発信元仮想レジスタ番号が割当てられ、宛先レジスタ番号は整数を示す。一方、整数から浮動小数点レジスタへの移動を有する発行位置は整数発信元仮想レジスタ番号が割当てられ、宛先レジスタ番号は浮動小数点を示す。

【0088】

ここで図 8 を参照して、仮想 / 物理レジスタマップユニット 86 の一実施例のブロック図が示される。他の実施例も可能であり、企図される。図 8 の実施例では、仮想 / 物理レジスタマップユニット 86 はソース IQ_{mux160} と、次の先読み IQ_{mux162} と、発信元 PR_{mux164} と、次の先読み PR_{mux168} と、トラップ IQ_{mux170} と、トラップ PR_{mux172} とを含む。発信元 IQ_{mux160} は、先読みレジスタ状態 84 からの各論理レジスタに対応する現先読み IQ を受けるように、また IQ / PR 制御ユニット 82 により割当てられた宛先 IQ を受けるように、接続される。次の先読み IQ_{mux162} も同様に、現先読み IQ および宛先 IQ を受けるよう接続される。発信元 IQ_{mux160} の出力は命令キュー 36A および 36B にパイプラインで接続され、一方次の先読み IQ_{mux162} の出力はトラップ IQ_{mux170} の入力として接続される。トラップ IQ_{mux170} は、回復先読みレジスタ状態バス 76A (回復すべき IQ を運ぶ回復先読みレジスタ状態バス 76 の一部) にさらに接続される。発信元 PR_{mux164} は、先読みレジスタ状態 84 からの各論理レジスタに対応する現先読み PR を受けるように、また IQ / PR 制御ユニット 82 により割当てられた宛先 PR を受けるように接続される。次の先読み PR_{mux168} も同様に、現先読み PR および宛先 PR を受けるように接続される。発信元 PR_{mux164} の出力は命令キュー 36A および 36B にパイプライン接続される。次の先読み PR_{mux168} の出力はトラップ PR_{mux172} に接続され、これは回復先読みレジスタ状態バス 76B (回復すべき PR を運ぶ回復先読みレジスタ状態バス 76 の一部) にさらに接続される。レジスタ走査ユニット 80 により割当てられた発信

10

20

30

40

50

元仮想レジスタ番号は、選択制御として発信元 I Q $\text{mux}160$ および発信元 P R $\text{mux}164$ に与えられる。次の仮想先読み状態は、次の先読み I Q $\text{mux}162$ および次の先読み P R $\text{mux}168$ に選択制御として与えられる。P C サイロ 48 からのトラップ制御はトラップ I Q $\text{mux}170$ およびトラップ P R $\text{mux}172$ に対して選択制御を与える。

【0089】

通常、発信元 I Q $\text{mux}160$ および発信元 P R $\text{mux}164$ は、レジスタ走査ユニット 80 により与えられた対応する発信元仮想レジスタ番号に回答して各 R O P の各発信元オペランドに対して発信元 I Q および P R を選択する。 $\text{mux}160$ は、たとえば $\text{mux}160$ に対して図示したとおり入力を受けよう接続され、かつ対応する発信元仮想レジスタ番号を選択制御として受ける mux の並行なセット（各 R O P の各発信元レジスタに対して 1 つ）として実現され得る。同様に、 $\text{mux}164$ は、 $\text{mux}164$ に対して図示したとおり入力を受け、かつ対応する発信元仮想レジスタ番号を選択制御として受けるよう結合された mux の並行なセット（各 R O P の各発信元レジスタに対して 1 つ）として実現され得る。発信元仮想レジスタ番号が、現先読み状態が I Q / P R に対する発信元であることを示す場合、発信元仮想レジスタ番号に含まれる論理レジスタ番号が用いられて、先読みレジスタ状態 84 により与えられた I Q および P R のうち 1 つを選択する。一方、発信元仮想レジスタ番号が前の発行位置を示す場合、この発行位置番号が用いられて制御ユニット 82 により割当てられた宛先 I Q / P R のうち 1 つを選択する。制御ユニット 82 は、たとえば、各発行位置に対応する宛先 I Q および P R をもたらし得る。一方、制御ユニット 82 は制限された数の I Q および P R （1 ライン中の R O P の数より少ない）をもたらし得る。このような実施例では、論理は発信元仮想レジスタ番号および宛先レジスタ有効指標において各 R O P に対して行なわれ、宛先 I Q / P R の組合せのうち 1 つを選択し得る。

【0090】

同様に、次の先読み I Q $\text{mux}162$ および次の先読み P R $\text{mux}168$ は、レジスタ走査ユニット 80 により与えられた対応する次の仮想先読み状態に回答して各論理レジスタに対して I Q および P R を選択する。 $\text{mux}162$ は、たとえば、 $\text{mux}162$ に対して図示されたとおり入力を受けよう接続され、かつ対応する仮想レジスタ番号を選択制御として受ける mux （各論理レジスタについて 1 つ）の並列セットとして実現され得る。同様に、 $\text{mux}168$ は、 $\text{mux}168$ に対して図示されたとおり入力を受けよう接続され、かつ対応する仮想レジスタ番号を選択制御として受ける、 mux （各論理レジスタについて 1 つ）の並列セットとして実現され得る。仮想レジスタ番号が、現先読み状態が特定の論理レジスタの I Q / P R に対する発信元であることを示す場合、この仮想レジスタ番号中に含まれる論理レジスタ番号が用いられて、先読みレジスタ状態 84 により与えられた I Q および P R のうち 1 つを選択する。一方、仮想レジスタ番号が発行位置を示す場合、発行位置番号が用いられて制御ユニット 82 により割当てられた宛先 I Q / P R のうち 1 つを選択する。

【0091】

図 8 に示す仮想レジスタ番号に基づいて I Q と P R との選択を物理的に分離することにより、利点が達成され得る。比較的多数のバスが比較的少量の回路（すなわち $\text{mux}160$ 、 162 、 164 および 168 で表わされる mux ）に経路づけられる。したがって、仮想 / 物理レジスタマップユニット 86 により占有される領域の量は、先読みレジスタ状態 84 および I Q / P R 制御ユニット 82 からのバスにより支配され得る。I Q および P R は別々の値であるので、これらの値を別々の mux のセットへと経路づけることにより、仮想 / 物理レジスタマップユニット 86 が占有する領域が減少し得る。一実施例では、I Q におけるビット数および P R におけるビット数はほぼ同じであり得る。したがって、仮想 / 物理レジスタマップユニット 86 の一方側の I Q バスのワイヤリングおよび仮想 / 物理レジスタマップユニット 86 の他方側の P R バスのワイヤリングは、公平に対照的なレイアウトとなり、両側のワイヤリング空間が効果的に用いられ得

10

20

30

40

50

ることになる。さらに、命令キュー 36 A および 36 B は、スケジューリング部分（現 ROP が従属する ROP がいつ完了したか、したがって ROP の現セットの 1 つがいつスケジュールされ得るか、を判定するための IQ を受ける部分）と、命令記憶部（PR および命令の実行に用いられる他の情報は記憶するが、IQ は記憶しない部分）とに物理的に分けられ得る。したがって、命令キュー 36 A および 36 B は、スケジューリング部分を IQ mux の近傍に、また命令記憶部分を PR mux の近傍に配置して物理的に構築され、さらに領域の節約を向上させる。

【0092】

図 8 に示す仮想ノ物理レジスタマップユニット 86 は、例外条件に回答して、先読みレジスタ状態 84 に対する次の先読みレジスタ状態のマッピングをさらに取扱う。トラップ IQ mux 170 は、トラップが実行時に認識されると用いられ、マップサイロ 32 からの回復先読みレジスタ状態に対応する IQ を経路づけ、また mux 162 によって与えられた次の先読み状態をオーバーライドする。PC サイロ 48 はトラップに mux 170 上での選択制御として信号を送り得る。同様に、回復先読みレジスタ状態内の PR は、トラップ信号に回答してトラップ PR mux 172 を通じて選択され得る。なお、この他の実施例では、例外条件から回復するための他の方法も採用され得る。たとえば、例外条件はリタイアメント時に取り扱われ得る。

10

【0093】

次に図 9 を参照して、マップサイロ 32 内のエントリ（すなわち ROP のあるラインに対応するエントリ）に記憶された情報を表わす表 180 が示される。他の実施例も可能であり、企図される。

20

【0094】

表 180 に示すように、このラインに対応する R が R（ライン部分）フィールドに記憶される。この記憶された R は、PC サイロ 48 により ROP のラインに割当てられた R のライン部分である。このライン部分はライン内の各 ROP について同じであり、一方 R のオフセット部分はある特定の ROP のライン内の発行位置を識別する。したがって、サイロエントリ（全体としてこのラインに対応する）は、R が記憶される ROP に対する R のライン部分を比較することによって、例外を経験している ROP に関連づけられ得る。

【0095】

さらに、そのライン中のどの ROP が有効であるかの指標はマップサイロエントリのライン（Line）フィールド内の有効 ROP に記憶される。たとえば、この指標は 1 ROP ごとに 1 ビットであり得る。ビットがセットされていると、そのライン内の対応の ROP は有効である。ビットがクリアされていると、そのライン内の対応する ROP は無効である。またさらに、どの ROP が宛先論理レジスタを有するかという指標は ROP レジスタ書込フィールドに記憶される。これもまた、指標は 1 ROP ごとに 1 ビットであり得る。ビットがセットされれば、ライン内の対応する ROP は宛先レジスタを更新する。ビットがクリアであれば、ライン内の対応する ROP は宛先レジスタを更新しない。どの ROP が宛先レジスタを有するかという指標は、割当てられた PR および割当てられた IQ のうちどれが以下に述べるような例外イベントにおいて回復状態の部分となるかを決定するのに用いられる。

30

40

【0096】

宛先レジスタを有する ROP に割当てられた PR および IQ は、それぞれエントリの割当てられた PR フィールドおよび割当てられた IQ フィールドに維持される。さらに、各宛先レジスタの論理レジスタ番号が論理レジスタ番号フィールドに記憶される。これらの論理レジスタ番号が用いられて、回復先読み状態内のどの論理レジスタが、以下に述べるように、割当てられた PR および割当てられた IQ を受けるかが決定される。さらに、ラインのリタイアメントが成功すると、論理レジスタ番号および対応する PR はアーキテクチャ的再命名ブロック 34 に運ばれて記憶される。

【0097】

50

ライン内のどの ROP が条件コードレジスタを更新するかという指標は、CC 書込フィールドに記憶される。更新される条件コードの各部分は、CC 書込フィールド内のビットによって別々に表わされ、1 セットのビットは、ROP レジスタ書込フィールド内に示される各レジスタ書込みと関連づけられ得る。対応する ROP の PR および IQ (割当てられた PR および IQ フィールドに記憶される) は、例外が検出されると、先読みレジスタ状態内の先読み条件コードレジスタを回復するのに用いられ得る。さらに、CC 書込フィールドは割当てられた PR フィールド内のどの PR が、ROP のラインのリタイアメントが成功するとアーキテクチャ条件コード状態に対応するかを示す。上述したように、本実施例では、整数レジスタ更新および対応の条件コード更新の両者を記憶するために単一の物理レジスタが用いられる。

10

【0098】

マップサイロエントリに対応する ROP のラインのディスパッチングの前に先読みレジスタ状態 84 内に記憶された現先読みレジスタ状態は、エントリの現先読みレジスタ状態フィールドに記憶される。この現先読みレジスタ状態は、ライン内の例外のイベントにおいて先読みレジスタ状態 84 を回復するための基準としての役割を果たす。

【0099】

最後に、各発行位置に対応する FP TOS および有効ビットは、エントリの FP TOS および有効ビットフィールド内に記憶される。例外を経験する ROP に対応する FP TOS および有効ビットは、例外が検出されると、先読みレジスタ状態 84 内の先読み FP TOS およびタグワード内に復元される。

20

【0100】

ここで図 10 を参照して、論理レジスタに対する先読みレジスタ状態 84 の一実施例によって採用され得る先読みレジスタ状態エントリ 182 の一例のブロック図が示される。エントリ 182 は、有効フィールド 184 と、IQ フィールド 186 と、PR フィールド 188 とを含む。有効フィールド 184 は IQ フィールド 186 が有効であるか否かを示す。IQ は、対応する ROP が命令キュー 36A および 36B から完成されるまでは有効である。したがって、先読み状態 84 は IQ が完成しているロード/ストアユニット 42 および実行コア 40 からの指標を受けることができる。一致が検出されると、先読み状態 84 は有効指標をリセットし得る。この有効指標は、たとえばセットされていると有効でありクリアされていると無効であることを示す、1 ビットであり得る。IQ フィールド 186 は論理レジスタを最も最近更新した ROP の IQ を記憶し、PR フィールド 188 は、ROP の宛先レジスタとして割付けられた物理レジスタの PR を記憶する。

30

【0101】

次に図 11 を参照して、PC サイロ 48 によって信号で知らされる例外条件に回答したマップサイロ 32 の動作を表わすフローチャートが示される。この他の実施例も可能であり、企図される。図 11 に示すステップは理解しやすいように特定の順序で示されるが、適切であればいかなる順序も採用され得る。さらに、ステップは、マップサイロ 32 内で採用される組合せ論理 (combinatorial logic) において並列で行なわれ得る。

【0102】

40

ステップ 190 に示されるように、マップサイロ 32 は、マップサイロ 32 に記憶されたエントリの R (ライン部分) フィールドに対して PC サイロ 48 により与えられた R をカムする。PC サイロ 48 により与えられた R は特定の ROP を識別する。しかしながら、マップサイロ 32 に対する検査の目的で、R のライン部分がカムされる。与えられた R より最近のエントリ (プログラム順序で例外の後の命令) はマップサイロ 32 内で取消される。取消されたエントリの割当てられた PR フィールドに記憶された PR は自由にされる。一実施例では、取消されたエントリの PR は、1 クロックサイクルごとに 1 エントリの割合で、複数のクロックサイクルにわたって自由にされる。カムが一致を示すサイロエントリが、選択されたマップサイロエントリである。

【0103】

50

選択されたマップサイロエントリに記憶された現先読みレジスタ状態（すなわち例外を経験する ROP を含む ROP のラインの前の現先読み状態が、現先読みレジスタ状態 8 2 の先読み状態に復元される（ステップ 1 9 2））。さらに、先読みレジスタ状態 8 4 における FP TOS および有効ビットは、例外を経験する ROP の発行位置に対して選択されたマップサイロエントリに記憶された値へと復元される（ステップ 1 9 2）。

【 0 1 0 4 】

選択されたマップサイロエントリ内の ROP レジスタ書込フィールドは、例外を経験する ROP の前の書込へマスクされる。言換えれば、例外を経験する ROP の後の ROP レジスタ書込はマスクされて取除かれ、それらがマスキングの後に書込として現われないようにする。（マスキング後の）残りの書込は走査されて、各レジスタへの一番最近の書込が検出される（すなわち、残りの書込のうち 2 つ以上が同じレジスタに対するものであれば、より最近の書込が保持される）。現先読み状態はこの結果を用いて更新される（ステップ 1 9 6）。ステップ 1 9 2 およびステップ 1 9 6 は、回復先読みレジスタ状態を先読みレジスタ状態 8 4 に送信する前にマップサイロ 3 2 において行なわれ得る。代替的には、現先読みレジスタ状態は先読みレジスタ状態 8 4 に復元され、またその後ステップ 1 9 6 に関して更新され得る。

10

【 0 1 0 5 】

またさらに、選択されたマップサイロエントリにおける ROP レジスタ書込フィールド（すなわちステップ 1 9 6 のマスキング前の元の値）は、例外を経験する ROP の後のレジスタ書込へとマスクされる。言換えれば、例外を経験する ROP の前のレジスタ書込はマスクして取除かれる。残りのレジスタ書込の PR は自由にされる（ステップ 1 9 8）。図 1 1 に示すフローチャートは、例外に回答して先読み状態を回復するための迅速な方法を有利に提供し得る。

20

【 0 1 0 6 】

なお、上記において、発信元仮想レジスタ番号という用語および仮想発信元レジスタ番号という用語を用いることができる。これらの用語は同じ意味であることが意図される。さらに、上述したように、プロセッサ 1 0 により採用された命令セットアーキテクチャにおいて具体化された各命令が単一の命令動作へとマッピングする実施例は、ここに規定される命令動作の意味の範囲内で企図される。

【 0 1 0 7 】

次に図 1 2 を参照して、バスブリッジ 2 0 2 を介してさまざまなシステム構成要素に結合されたプロセッサ 1 0 を含むコンピュータシステム 2 0 0 の一実施例のブロック図が示される。この他の実施例も可能であり、企図される。図示されたシステムでは、メインメモリ 2 0 4 がメモリバス 2 0 6 を介してバスブリッジ 2 0 2 に結合され、グラフィックコントローラ 2 0 8 が AGP バス 2 1 0 を介してバスブリッジ 2 0 2 に結合される。最後に、複数の PCI 装置 2 1 2 A および 2 1 2 B が PCI バス 2 1 4 を介してバスブリッジ 2 0 2 に結合される。電氣的インタフェースを 1 つ以上の EISA または ISA 装置 2 1 8 に EISA / ISA バス 2 2 0 を介して適応させるために、二次的バスブリッジ 2 1 6 をさらに設けてもよい。プロセッサ 1 0 は外部インタフェース 5 2 を介してバスブリッジ 2 0 2 に結合される。

30

40

【 0 1 0 8 】

バスブリッジ 2 0 2 は、プロセッサ 1 0 と、メインメモリ 2 0 4 と、グラフィックコントローラ 2 0 8 と、PCI バス 2 1 4 に取付けられた装置との間にインタフェースをもたらす。バスブリッジ 2 0 2 に接続された装置の 1 つから命令（operation）を受取ると、バスブリッジ 2 0 2 はその命令のターゲット（たとえば特定の装置、または PCI バス 2 1 4 の場合にはターゲットは PCI バス 2 1 4 上にある）を識別する。バスブリッジ 2 0 2 はその命令をターゲットの装置へと経路づける。バスブリッジ 2 0 2 は通常、発信元装置またはバスにより用いられるプロトコルからターゲット装置またはバスにより用いられるプロトコルへと命令を変換する。

【 0 1 0 9 】

50

PCIバス214に対するISA/EISAにインタフェースをもたらすことに加えて、二次的バスブリッジ216は所望に応じて付加機能をさらに組入れることができる。たとえば、一実施例では、二次的バスブリッジ216はPCIバス214の所有権を調停するためのマスタPCIアービタ(図示せず)を含む。二次的バスブリッジ216から外部の、または二次的バスブリッジ216に一体化された、入力/出力コントローラ(図示せず)をコンピュータシステム200内に含んで、キーボードおよびマウス222ならびにさまざまなシリアルポートおよびパラレルポートに対して、所望に応じて動作上のサポートを提供することもできる。他の実施例においては、プロセッサ10とバスブリッジ202との間に、外部キャッシュユニット(図示せず)をさらに外部インタフェース52に結合させてもよい。代替的には、この外部キャッシュはバスブリッジ202に結合されてもよく、外部キャッシュに対するキャッシュ制御論理はバスブリッジ202内に一体化され得る。

10

【0110】

メインメモリ204は、アプリケーションプログラムが記憶され、そこからプロセッサ10が主に実行するメモリである。好適なメインメモリ204は、DRAM(ダイナミックランダムアクセスメモリ)を含み、好ましくは複数のバンクのSDRAM(同期DRAM)を含む。

【0111】

PCI装置212Aおよび212Bは、たとえば、ネットワークインタフェースカード、ビデオアクセレレータ、オーディオカード、ハードもしくはフロッピーディスクドライブまたはドライブコントローラ、SCSI(小型コンピュータシステムインタフェース)アダプタ、およびテレフォニーカードなどのさまざまな周辺機器を表わす。同様に、ISA装置218は、モデム、サウンドカード、およびGPIBまたはフィールドバスインタフェースカードなどの種々のデータ収集カードなど、さまざまなタイプの周辺機器を表わすものである。

20

【0112】

グラフィックコントローラ218は、文字または画像のディスプレイ226上での表現(rendering)を制御するために設けられる。グラフィックコントローラ208は従来技術で一般に公知である典型的なグラフィックアクセレレータを採用して、メインメモリ204へおよびメインメモリ204から効果的にシフトされ得る3次元データ構造を表現し得る。グラフィックコントローラ208はしたがって、AGPバス210のマスタであってもよく、そこでバスブリッジ202内のターゲットインタフェースへのアクセスを要求しかつそれを受けることができ、よってメインメモリ204へのアクセスを得ることができる。専用グラフィックバスはメインメモリ204からのデータの迅速な再生に適應する。特定の動作に対して、グラフィックコントローラ208はさらに、AGPバス210上でのPCIプロトコルトランザクションを生成するようさらに構成され得る。バスブリッジ202のAGPインタフェースはしたがって、AGPプロトコルトランザクションならびにPCIプロトコルターゲットおよびイニシエータトランザクションの両者をサポートするための機能性を含み得る。ディスプレイ226はそこに画像または文字を表示し得る何らかの電子ディスプレイである。好適なディスプレイ226は、陰極線管(「CRT」)、液晶ディスプレイ(「LCD」)、などを含む。

30

40

【0113】

なお、上記の説明ではAGP、PCIおよびISAまたはEISAバスが例として用いられたが、所望に応じて、いかなるバスアーキテクチャで代用してもよい。さらに、コンピュータシステム200はさらなるプロセッサ(コンピュータシステム200のオプション構成要素として示されるプロセッサ10aなど)を含む多重プロセッシングコンピュータシステムであり得る。プロセッサ10aはプロセッサ10と同様である。より具体的には、プロセッサ10aはプロセッサ10の同一の複製物であり得る。プロセッサ10aは(図12に示すような)プロセッサ10と外部インターフェイス52を共有しても、または独立したバスを介してバスブリッジ202に接続されてもよい。

50

【0114】

上記の開示に従って、レジスタ再命名スキームを採用するプロセッサが示された。一実施例では、この再命名スキームは仮想レジスタ番号を用いて複数の段階に分割される。有利には、このスキームを用いて高周波数動作が可能になり得る。別の実施例では、ROPの各ラインに対応する現先読み状態を格納し(siloing)、またライン内のレジスタ書込に気づくことにより、例外からの迅速な回復が行なわれる。さらに別の実施例では、再命名レジスタの自由化が管理され、同じ再命名レジスタが1つ以上の論理レジスタと対応できるようにする。これにより、効率的な物理レジスタの使用が採用され得る。

【0115】

ここに記載した実施例に対しては、多くの変形および変更がなされ得る。本発明は、前掲の請求の範囲内に収まるそのような変形および変更のすべてを含むことが意図される。

10

【図面の簡単な説明】

【図1】 プロセッサの一実施例のブロック図である。

【図2】 図1に示されるプロセッサの一実施例による、図1に示されるマップユニット、マップサイロおよびアーキテクチャ的再命名ブロックの間の相互接続を強調するブロック図である。

【図3】 図1および図2に示されるマップユニットの一実施例のブロック図である。

【図4】 図3に示されるレジスタ走査ユニットの一実施例のブロック図である。

【図5】 仮想レジスタ番号のために用いることができる1つの符号化を示す表の図である。

20

【図6】 図4に示される走査ユニットの一実施例の一部分のブロック図である。

【図7】 図4に示される走査ユニットの一実施例の別の部分のブロック図である。

【図8】 図3に示される仮想/物理レジスタマップユニットの一実施例のブロック図である。

【図9】 図1および図2に示されるマップサイロの一実施例において記憶される情報を示す表の図である。

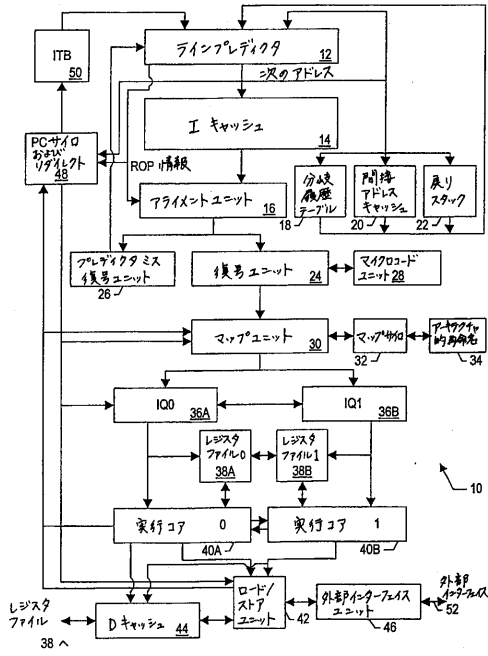
【図10】 論理レジスタのための例示的な先読みレジスタ状態を示すブロック図である。

【図11】 図1、図2および図3に示されるマップユニットの一実施例による先読み状態を復元することを示すフローチャートの図である。

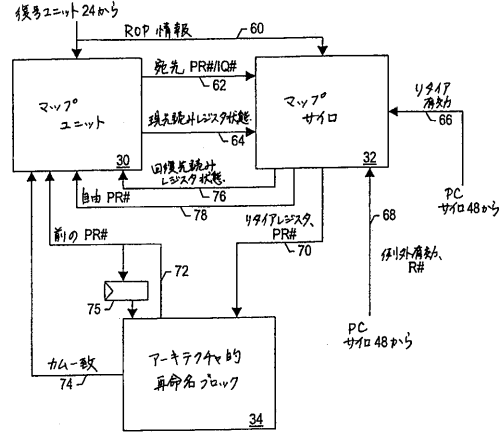
30

【図12】 図1に示されるプロセッサを含むコンピュータシステムの一実施例のブロック図である。

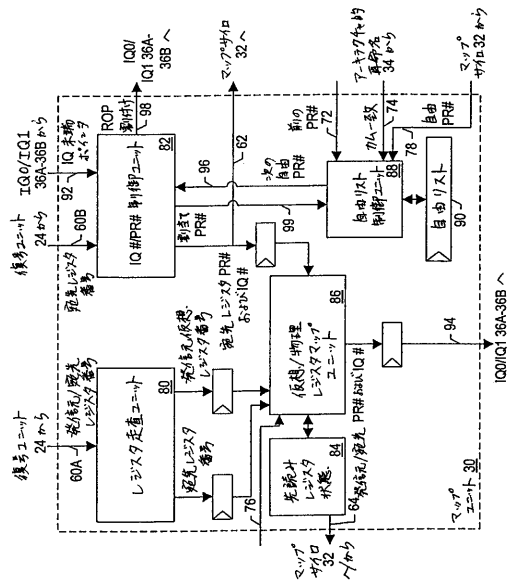
【図1】



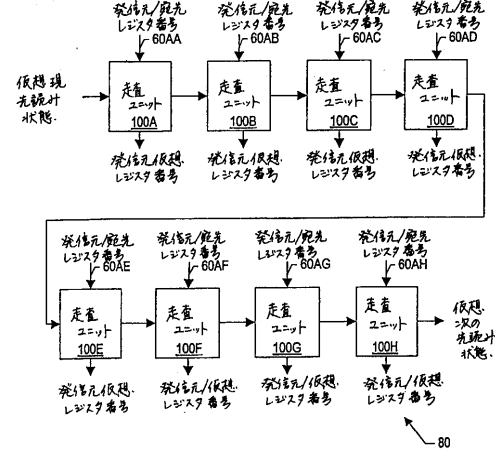
【図2】



【図3】



【図4】

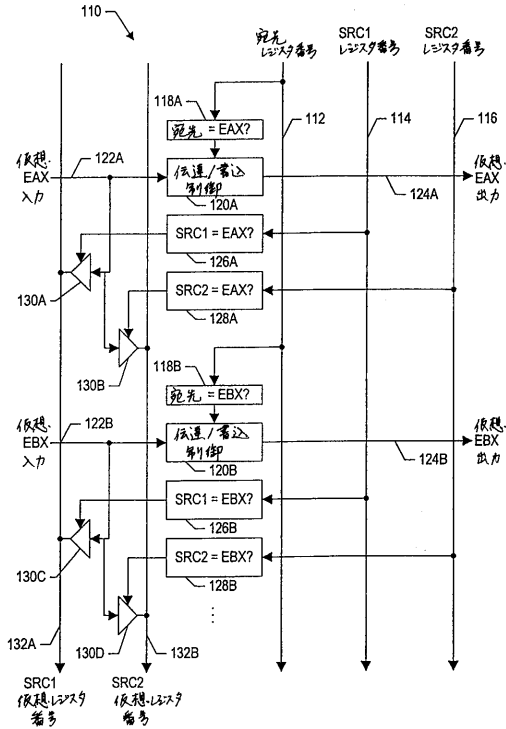


【図5】

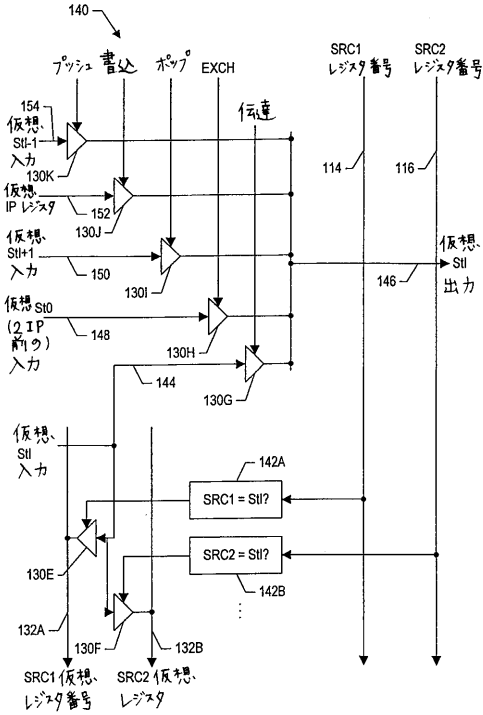
| MSB | LSBs |
|-----|----------|
| 0 | 論理レジスタ番号 |
| 1 | 発行位置番号 |

102

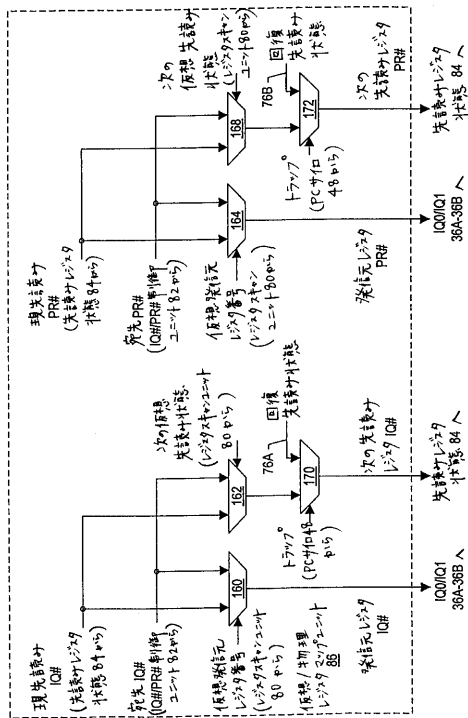
【図 6】



【図 7】



【図 8】



【図 9】

| |
|------------------------|
| R# (ライン部分) |
| ライン内有効 ROP |
| ROP レジスタ書込 |
| 割り当てられた PR# |
| 割り当てられた IC# |
| 論理レジスタ番号 |
| CC 書込 |
| 現失音のレジスタ状態 |
| FPTOS および有効ビット (各実行位置) |

【図 10】

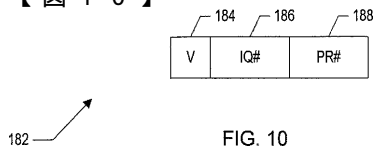
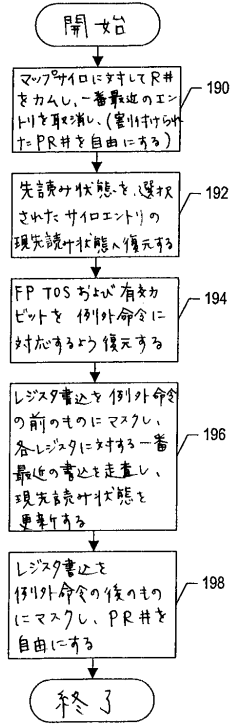
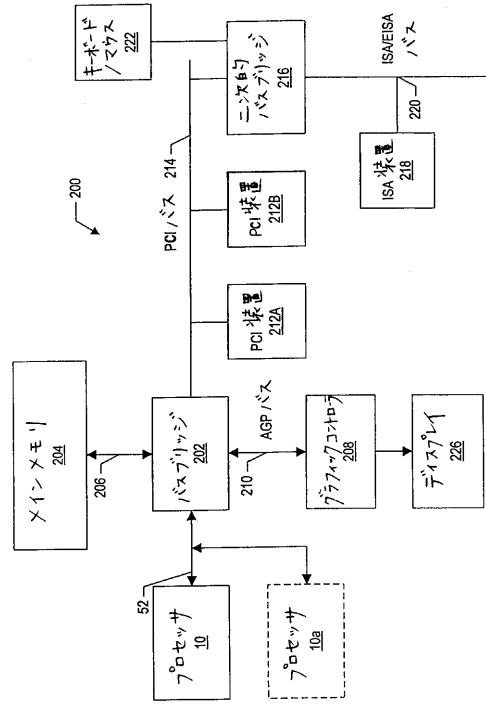


FIG. 10

【 図 1 1 】



【 図 1 2 】



フロントページの続き

(74)代理人 100083703

弁理士 仲村 義平

(74)代理人 100091409

弁理士 伊藤 英彦

(74)代理人 100096781

弁理士 堀井 豊

(74)代理人 100096792

弁理士 森下 八郎

(72)発明者 ウィット, デイビッド・ビィ

アメリカ合衆国、78759 テキサス州、オースティン、パスファインダー・ドライブ、6318

審査官 後藤 彰

(56)参考文献 特表2000-506639(JP, A)

特開平7-295813(JP, A)

特表平6-501805(JP, A)

特開平5-241830(JP, A)

国際公開第97/22921(WO, A1)

(58)調査した分野(Int.Cl., DB名)

G06F 9/30 - 9/355