

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4921665号  
(P4921665)

(45) 発行日 平成24年4月25日(2012.4.25)

(24) 登録日 平成24年2月10日(2012.2.10)

(51) Int.Cl.

F I

G 0 6 F 9/30 (2006.01)

G 0 6 F 9/30 3 5 0 F

請求項の数 4 (全 11 頁)

(21) 出願番号 特願2001-525517 (P2001-525517)  
(86) (22) 出願日 平成12年8月29日 (2000.8.29)  
(65) 公表番号 特表2003-510682 (P2003-510682A)  
(43) 公表日 平成15年3月18日 (2003.3.18)  
(86) 国際出願番号 PCT/US2000/023721  
(87) 国際公開番号 W02001/022216  
(87) 国際公開日 平成13年3月29日 (2001.3.29)  
審査請求日 平成19年8月29日 (2007.8.29)  
(31) 優先権主張番号 09/399,612  
(32) 優先日 平成11年9月20日 (1999.9.20)  
(33) 優先権主張国 米国 (US)

(73) 特許権者 591003943  
インテル・コーポレーション  
アメリカ合衆国 95052 カリフォル  
ニア州・サンタクララ・ミッション カレ  
ッジ ブレーバード・2200  
(74) 代理人 100064621  
弁理士 山川 政樹  
(72) 発明者 ドュロン, カロル  
アメリカ合衆国・95070・カリフォル  
ニア州・サラトガ・ハーレー ドライブ・  
18983

審査官 三坂 敏夫

最終頁に続く

(54) 【発明の名称】 プレディケーションを用いたマスクに基づくパックされたデータからのデータ要素の選択的書き込み

(57) 【特許請求の範囲】

【請求項 1】

並列処理装置を有するコンピュータにおいて実行される方法であって、

パックされたデータ・オペランドの1つのデータ要素を、現在選択されているデータ要素として選択するステップであって、前記パックされたデータ・オペランドのデータ要素は、書き込むためにそのデータ要素が選択されているかどうかを示す対応するマスク・データ要素を有する、ステップと、

前記現在選択されているデータ要素に対する現在選択されている記憶位置を示す値を記憶するステップと、

前記対応するマスク・データ要素の1つまたは複数のビットから、前記現在選択されているデータ要素用のプレディケイト値を決定するステップと、

前記並列処理装置が、前記プレディケイト値に基いて前記現在選択されているデータ要素を前記現在選択されている記憶位置に書き込む工程と、前記現在選択されている記憶位置を示す前記値を増分する工程と、前記パックされたデータ・オペランドの他のデータ要素を前記現在選択されているデータ要素として選択し、かつ前記対応するマスク・データ要素の1つ又はそれ以上のビットから前記データ要素用のプレディケイト値を決定する工程とを実行することによって、前記パックされたデータ・オペランドから前記現在選択されているデータ要素を選択的に書き込むステップと、

前記パックされたデータ・オペランドのデータ要素の最終要素が前記現在選択されているデータ要素になるまで前記ステップを繰返すステップと、そして

10

20

前記プレディケイト値に基いて前記現在選択されている記憶位置に前記現在選択されているデータ要素を書き込むステップとを含む方法。

【請求項 2】

並列処理装置を有するコンピュータにおいて実行される方法であって、

バックされたデータ・オペランドの 1 つのデータ要素を、現在選択されているデータ要素として選択するステップであって、前記バックされたデータ・オペランドのデータ要素は、書き込むためにそのデータ要素が選択されているかどうかを示す対応するマスク・データ要素を有する、ステップと、

前記現在選択されているデータ要素に対する現在選択されている記憶位置を示す値を記憶するステップと、

前記対応するマスク・データ要素の 1 つまたは複数のビットから、前記現在選択されているデータ要素用のプレディケイト値を決定するステップと、

前記並列処理装置が、前記プレディケイト値に基いて前記現在選択されているデータ要素を前記現在選択されている記憶位置に書き込む工程と、前記現在選択されている記憶位置を示す前記値を増分する工程と、前記バックされたデータ・オペランドを 1 つのデータ要素だけシフトし、かつ前記対応するマスク・データ要素の 1 つ又はそれ以上のビットから前記データ要素用のプレディケイト値を決定する工程とを実行することによって、前記バックされたデータ・オペランドから前記現在選択されているデータ要素を選択的に書き込むステップと、

前記バックされたデータ・オペランドのデータ要素の最終要素が前記現在選択されているデータ要素になるまで前記ステップを繰返すステップと、そして

前記プレディケイト値に基いて前記現在選択されている記憶位置に前記現在選択されているデータ要素を書き込むステップとを含む方法。

【請求項 3】

並列処理装置を有するコンピュータにおいて実行される方法であって、

バックされたデータ・オペランドの 1 つのデータ要素を、現在選択されているデータ要素として選択するステップであって、前記バックされたデータ・オペランドのデータ要素は、書き込むためにそのデータ要素が選択されているかどうかを示す対応するマスク・データ要素を有する、ステップと、

前記現在選択されているデータ要素に対する現在選択されている記憶位置を示す値を記憶するステップと、

前記対応するマスク・データ要素の 1 つまたは複数のビットから、前記現在選択されているデータ要素用のプレディケイト値を決定するステップと、

前記並列処理装置が、前記プレディケイト値に基いて前記現在選択されているデータ要素を前記現在選択されている記憶位置に書き込む工程と、前記現在選択されている記憶位置を示す前記値を増分する工程と、前記バックされたデータ・オペランドの他のデータ要素を前記現在選択されているデータ要素として選択し、かつ前記対応するマスク・データ要素の 1 つ又はそれ以上のビットから、テスト・ビット命令を使用することにより前記データ要素用のプレディケイト値を決定する工程とを実行することによって、前記バックされたデータ・オペランドから前記現在選択されているデータ要素を選択的に書き込むステップと、

前記バックされたデータ・オペランドのデータ要素の最終要素が前記現在選択されているデータ要素になるまで前記ステップを繰返すステップと、そして

前記プレディケイト値に基いて前記現在選択されている記憶位置に前記現在選択されているデータ要素を書き込むステップとを含む方法。

【請求項 4】

バックされたデータ・オペランドの各バックされたデータ要素を対応する記憶位置に選

10

20

30

40

50

択的に書き込むための一つまたは複数の命令のセットをデコードするデコーダと、

前記対応する記憶位置に書き込むために前記バックされたデータ要素が選択されているか否かを示すバックされたデータ・マスク要素の1つまたは複数のビットに基づいて前記バックされたデータ要素の各々に対してプレディケイト値を決定するために前記デコーダに接続されたプレディケーション回路と、そして

前記デコーダに接続された実行ユニットであって、多重並列処理装置と結合バッファを有する実行ユニットと

を有するプロセッサにおいて、

前記プレディケーション回路が、データ要素用のプレディケイト値を決定し、

前記多重並列処理装置が、前記プレディケイト値に基づいて前記現在選択されているデータ要素を前記現在選択されている前記結合バッファの記憶位置に書き込む工程と、前記現在選択されている記憶位置を示す前記値を増分する工程と、前記バックされたデータ・オペランドの他のデータ要素を前記現在選択されているデータ要素として選択し、かつ前記対応するマスク・データ要素の1つ又はそれ以上のビットから前記データ要素用のプレディケイト値を決定する工程とを実行することによって、前記バックされたデータ・オペランドから前記現在選択されているデータ要素を選択的に前記結合バッファに記憶し、

前記実行ユニットが、前記バックされたデータ・オペランドのデータ要素の最終要素が前記現在選択されているデータ要素になるまで前記多重並列処理装置による処理を繰返し、そして

前記記憶した結合バッファを前記対応する記憶位置に書き込むことを特徴とするプロセッサ。

【発明の詳細な説明】

【0001】

(発明の背景)

(発明の分野)

本発明は、コンピュータ・システムの分野に関する。より詳細には、本発明は、プレディケーション(predication)を用いたマスクに基づくバックされたデータからのデータ要素の選択的書込みに関する。

【0002】

(背景情報)

コンピュータ技術は、加速し続ける速さで発展を続けている。コンピュータが、主としてワード・プロセッシング・アプリケーションやスプレッドシート・アプリケーションのために使用される単なるビジネス・ツールであった時代はもう終わった。現在では、マルチメディア・アプリケーションの発展に伴い、コンピュータ・システムは、テレビジョンおよび家庭用ステレオ・システムにより近い一般的な家庭電化製品となった。実際、適切に構成されたコンピュータ・システム上で実行されるマルチメディア・アプリケーションがテレビジョン・セット、ラジオ、ビデオ再生装置等のように機能するので、コンピュータ・システムと他の家庭用電化製品との間の境界線はあいまいになってきている。したがって、コンピュータ・システムの市場における人気は、それらが含むメモリ量と、そのようなマルチメディア・アプリケーションを実行することができる速度によって決定することがしばしばある。

【0003】

当業者ならば、期待されるようになったオーディオおよびビデオの生き生きしたレンダリングを実現するために、マルチメディア・アプリケーションと通信アプリケーションは、少数のビットで表される大量データを処理することを必要とすることを理解するだろう。例えば、3Dグラフィックをレンダリングするために、個別データ・アイテムの比較的大きな集合(例えば、8ビットのデータ)を同様に処理しなければならない。

【0004】

このようなアプリケーションによって必要とされる1つの共通したオペレーションは、データ・アイテムの集合から選択されたデータ・アイテムをメモリに書き込むことである。

所与のデータ・アイテムがメモリに書き込まれるべきかどうかはマスクによって決まる。データの移動する選択バイトに対する1つの手法は、一連のテスト、分岐、および書込みの命令を使用する。この手法によれば、対応するデータ・アイテムに対して1つまたは複数のマスク・ビットがテストされ、バイトをメモリに書き込むか、または書込みをバイパスするために分岐が使用される。しかし、この手法は、分岐の予測ミスのために性能を犠牲にするという難点がある。

#### 【0005】

この分岐の予測ミスによる不利益を回避するために、パックされたデータのある記憶位置から別の記憶位置に書き込むためのSIMD「バイト・マスク・ライト(Byte Mask Write)」命令をサポートするために、単一命令多重データ処理(SIMD)プロセッサ・アーキテクチャが使用される(1998年3月31日出願の米国特許出願第09/052,802号参照のこと)。図1は、SIMDアーキテクチャでSIMDバイト・マスク・ライト命令を実施するための専用並列回路を示すブロック図である。図1は、MM1と標識され、第1のオペランドSRC1 100で示されている第1のSIMDレジスタからレジスタによって指定されるメモリ位置106に整数データを表す最大64ビットを移動するSIMDバイト・マスクド・クォッドワード移動命令(MASKMOVQ)を示す。その際、MM2と標識され、第2のオペランドSRC2 102によって示される第2のSIMDレジスタに置かれるバイト・パックド・データ・マスク(byte packed data mask)を使用する。レジスタMM1 100のバイト110と114は、レジスタMM2 102に記憶されているマスクのバイト108と112によって書き込み許可される。

#### 【0006】

図1に示すように、このSIMDバイト・マスク・ライト命令は、パックされたデータ・アイテムの各バイトを並列に処理するために、プロセッサ内に専用回路を必要とする。この専用回路の並列であるという性質によって、比較的良好なプロセッサ処理能力が達成されるにも関わらず、この専用回路は貴重なダイ・エリアを必要とし、グラフィカルおよび類似のタイプの処理にのみ使用されるだけである。

#### 【0007】

(発明の簡単な概要)

プレディケーションを用いたマスクに基づくパックされたデータからデータ要素を選択的に書き込むための方法および装置が開示される。本発明の一実施態様では、並列処理装置で、パックされたデータ・オペランドのデータ要素に対して下記の事項が実施される。すなわち、そのデータ要素が対応する記憶位置に書き込まれるように選択されているかどうかを示す、対応するパックされたデータ・マスク要素の1つまたは複数のビットから、そのデータ要素のプレディケイト値を決定すること、およびそのプレディケイト値に基づいてそのデータ要素を対応する記憶位置に記憶することである。

#### 【0008】

本発明の特徴および利点は、以下の本発明の詳細な説明から明らかになる。

#### 【0009】

(発明の詳細な説明)

以下の説明では、説明のために、本発明の完全な理解を実現する目的で多くの詳細が記載されている。しかし、当業者には、本発明を実施するためにこれらの具体的な詳細が必ずしも必要ではないことが明らかになる。他の事例では、本発明を不明瞭にしないように、よく知られた電子構造および電子回路をブロック図形式で示す。

#### 【0010】

図2は、本発明の一実施形態によるコンピュータ・システム200の一実施形態を示すブロック図である。コンピュータ・システム200は、プロセッサ210、記憶装置220、およびバス215を備える。プロセッサ210は、バス215によって記憶装置220に結合されている。さらに、いくつかのユーザ入力/出力装置240(例えば、キーボード、マウス)もバス215に結合されている。プロセッサ210は、CISC、RISC、VLIW、またはハイブリッド・アーキテクチャ等の任意のタイプのアーキテクチャの

中央処理装置を表す。さらに、プロセッサ 210 は、1 つまたは複数のチップ上で実施することもできる。バス 215 は、1 つまたは複数のバス 210 (例えば、AGP、PCI、ISA、X-Bus、VESA 等) およびブリッジを表す。この実施形態は単一プロセッサのコンピュータ・システムに関して説明するが、本発明は、多重プロセッサのコンピュータ・システムで実施することも可能である。

#### 【0011】

別の装置に加え、1 つまたは複数のネットワーク・コントローラ 255、TV 放送単一受信機 260、ファックス/モデム 245、ビデオ・キャプチャ・カード 235、およびオーディオ・カード 250 を任意選択でバス 215 に結合することができる。ネットワーク・コントローラ 255 は、1 つまたは複数のネットワーク接続 (例えば、イーサネット (登録商標) 接続) を表す。ネットワークからのトラフィックをネットワーク・コントローラ 255 で受信する記憶装置 220 および媒体は、1 つまたは複数の機械可読媒体を代表している。したがって、機械可読媒体は、機械 (例えば、コンピュータ) が読み取れる形式で情報を記憶または伝送するための任意の機構を含む。例えば、機械可読媒体は、読取り専用メモリ (ROM)、ランダム・アクセス・メモリ (RAM)、磁気ディスク記憶媒体、光学記憶媒体、フラッシュ・メモリ・デバイス、電子・光学・音響または他の形式の伝搬信号 (例えば、搬送波、赤外線信号、デジタル信号等) を含む。

#### 【0012】

TV 放送信号受信機 260 は、TV 放送信号を受信するための装置を表し、一方、ファックス/モデム 245 は、データを表すアナログ信号を受信および/または送信するためのファックスおよび/またはモデムを代表している。イメージ・キャプチャ・カード 235 は、イメージをデジタル化するための 1 つまたは複数の装置 (すなわち、スキャナ、カメラ等) を代表している。オーディオ・カード 2590 は、音声を入力および/または出力するための 1 つまたは複数の装置 (例えば、マイクロフォン、スピーカ、磁気記憶装置、光学記憶装置等) を代表している。グラフィック・コントローラ・カード 230 もバス 215 に結合されている。グラフィック・コントローラ 230 は、イメージを生成するための 1 つまたは複数の装置 (例えば、グラフィック・カード) を代表している。

#### 【0013】

図 2 は、記憶装置 220 がデータ 224 およびプログラム・コード 222 を記憶しているということも示す。データ 224 は、1 つまたは複数の形式で記憶されているデータ (例えば、バックされたデータ・フォーマット) を代表している。プログラム・コード 222 は、本発明の任意および/またはすべての技術を実施するために必須のコードを表す。具体的には、プログラム・コード 222 は、プレディケーションと並列データ処理を使用したマスク (例えば、SIMD バイト・マスク書込み命令のエミュレーション) に基づくバックされたデータからのデータ要素の効率的な選択的書込みをプロセッサ 210 に実施させる 1 つまたは複数の命令を含んでいる。当然ながら、記憶装置 220 は、本発明を理解するために必須ではない付加的なソフトウェア (図示せず) を好適に含んでいる。

#### 【0014】

図 2 は、プロセッサ 210 がデコーダ 216、一組のレジスタ 214、実行ユニット 212、内部バス 211、およびプレディケーション回路 213 を含むことをも示す。当然ながら、プロセッサ 210 は、本発明を理解するために必須ではない付加的な回路も含んでいる。デコーダ 216、レジスタ 214、実行ユニット 212、およびプレディケーション (predication) 回路 213 は、内部バス 211 によって相互に結合されている。デコーダ 216 は、プロセッサ 210 によって受信された命令を制御信号および/またはマイクロコード・エントリポイントに復号するために使用される。これらの制御信号および/またはマイクロコード・エントリポイントに応答して、実行ユニット 212 は適切なオペレーションを実施する。デコーダ 216 は、任意の数の様々な機構を使用して実施することができる (例えば、参照用テーブル、ハードウェア実施、PLA 等)。

#### 【0015】

デコーダ 216 は、プロセッサ 210 にマスクに基づくバックされたデータからのデータ

10

20

30

40

50

要素の選択的書込みをさせるための1つまたは複数の命令をサポートするための復号回路218を含んでいるように示してある。さらに、一実施形態では、プロセッサ210は、クロック・サイクルごとにそれらの1つまたは複数の命令を完結させることができるスーパースカラ・パイプライン・プロセッサである(すべてのデータ依存性およびパイプライン・フリーズを無視して)。これらの命令の他に、プロセッサ210は、複数の新しい命令および/または既存の汎用プロセッサにある命令と類似の複数の命令を実行可能にすることもできる。例えば、一実施形態では、プロセッサ210は、カリフォルニア州サンタクララのIntel社製、Pentium(登録商標)IIIプロセッサ等の既存プロセッサによって使用されるIntelアーキテクチャ命令と互換性のある命令セットをサポートする。

10

#### 【0016】

実行ユニット212は、複数の命令を並列に処理することができる複数の並列処理装置219A~219N(例えば、並列処理装置)を含んでいる。本発明の一実施形態では、実行ユニット212は、64ビットの書込みを実行することができるまでデータを記憶しておくための結合バッファ270も含んでいる。この結合バッファ270は、バッファが記憶されるべきデータを蓄積している間、バス215を別の目的で使用することを可能にすることによって、バス215を最大限に活用することを可能にする。しかし、64ビットと異なる幅を有する結合バッファ270またはバス215は本発明の範囲内にあるということ、これらは、処理装置の特定の細部において相当する相違点を生じさせることがあるということを理解されたい。

20

#### 【0017】

レジスタ214は、制御/状態情報、数字データ、浮動小数点データ、パックされたデータ(数字および/または浮動小数点)、およびプレディケイト(predicate:述語)・データを含む情報を記憶するためのプロセッサ210上の記憶領域を表す。本発明の態様は、パックされたデータに関して動作するために記述された1つまたは複数の命令であり、また、1つまたは複数のこれらの命令を使用する方法であることが理解されよう。本発明のこれらの態様により、プレディケイト・データを記憶するために使用される記憶領域は重要ではない。本明細書では、図2を参照して説明した1つまたは複数のコンピュータ・システムを含めてデータを処理するための任意の機械を示すためにデータ処理システムという用語を使用する。

30

#### 【0018】

本発明の一実施形態では、プレディケーション回路213は、プロセッサ216によってサポートされる命令セットのうちの任意の命令のために使用することができる一般的なプレディケーション回路である。プレディケーションは、すべての命令の条件付き実行を可能にする。したがってこの技術は、分岐自体の除去を可能にし、それによって予測ミスによる不利益を解消する。1998年1月26発行のMicroprocessor ReportのPeter Song著、「Demystifying the EPIC and IA-64」と、1991年12月18日発行のMicroprocessor ReportのBrian Case著、「ARM Architecture Offers High Code Density」を参照されたい。一般的なプレディケーション・モデルが使用される場合の一実施形態が説明されているが、代替の実施形態は特定の命令(例えば、条件付き移動のタイプの命令)だけに関するプレディケーションをサポートするようにすることもできる。さらに、一実施形態は、マスクが、データ要素が書き込まれるマスクをデータ要素と同じサイズにすることを可能にするために使用されるように説明されているが、一方、代替の実施形態では、マスクは異なるサイズであってよい(各マスクが単一ビットから構成されることを含めて)。

40

#### 【0019】

図3は、本発明の一実施形態による流れ図である。ブロック305では、選択的に書き込まれるべきパックされたデータ・オペランドの現在選択されているデータ要素に対する書き込み可能な記憶位置を識別する初期値が決定される。ブロック305から、プロセス

50

はブロック 3 1 0 に進む。

【 0 0 2 0 】

ブロック 3 1 0 では、現在選択されているデータ要素に対するプレディケイト値が決定される。本発明の一実施形態では、プレディケイト値は、テスト・ビット (「t b i t」) 命令を使用することによって決定される。この実施形態によれば、t b i t 命令は、選択的に書き込まれるべきパックされたデータ・オペランドの現在選択されているデータ要素に対応するパックされたマスク・データ要素のビットをテストするために使用される。このプレディケイト値は、様々な記憶媒体 (例えば、専用プレディケイト・レジスタ、汎用レジスタ等) に記憶することができる。

【 0 0 2 1 】

図 3 のブロック 3 2 0 では、ブロック 3 1 0 で決定されたプレディケイト値に関して判断が行われる。ブロック 3 2 0 では、プレディケイト値が真であればプロセスの流れはブロック 3 3 0 に進む。そうでない場合、プロセスの流れはブロック 3 5 0 に進む。

【 0 0 2 2 】

ブロック 3 3 0 では、パックされたデータ・オペランドの現在選択されているデータ要素が記憶される (図 2 に関しては、現在選択されているデータ要素は結合バッファに記憶することができる)。ブロック 3 3 0 の後、プロセスはブロック 3 5 0 に進む。

【 0 0 2 3 】

図 3 のブロック 3 5 0 では、現在選択されているデータ要素が、処理されるべきパックされたデータ・オペランドの最後のデータ要素であるかどうかについてもう 1 つの判断がなされる。現在選択されているデータ要素が処理されるべき最後のデータ要素である場合、プロセスの流れはこの手順を終了するように進む。そうでない場合、プロセスの流れはブロック 3 5 0 からブロック 3 6 0 に進む。

【 0 0 2 4 】

図 3 のブロック 3 6 0 では、記憶位置に関連付けられた値が増分される。本発明の一実施形態では、この値は汎用レジスタ内に置かれる。しかし、記憶位置に関連付けられた値は、様々な記憶媒体内に置くこともできる。次に、図 3 のブロック 3 7 0 では、パックされたデータ・オペランドは 1 つのデータ要素の幅だけシフトされる (例えば、パックされたデータ・オペランドの最上位データ要素を現在選択されているデータ・アイテムであると仮定すると、パックされたデータ・オペランドは左シフトされ、その結果、次に下位のデータ要素が最上位データ要素、すなわち現在選択されているデータ要素になる)。次いでこのプロセスの流れは、3 7 0 からブロック 3 1 0 に戻り、最後のデータ要素が処理されるまでこのプロセスは繰り返される。

【 0 0 2 5 】

パックされたデータ・オペランドの現在選択されているデータ要素を選択するためにシフティングが用いられる一実施形態を説明したが、代替の実施形態は別の技術 (例えば、ポインタ、指示を可能にするサポート命令 (下記の t b i t 命令を参照のこと) 等) を使用することができる。さらに、図 3 は順次処理を例示するものと解釈することができるが、オペレーションを異なる順序で実行することができ (例えば、コンパイル・スケジューリング、ループ展開、ハードウェアの o u t - o f - o r d e r 実行、等)、様々なオペレーションを並列で実行することができ、かつ / または様々な他のオペレーションの流れの所々に挿入することができることも理解されたい。

【 0 0 2 6 】

本発明の一実施形態は、概念的に以下のようにとらえることができる。

R 2 = アドレス・レジスタ

R 3 = 6 4 ビットのデータ・レジスタ

R 4 = 対応するマスク要素を有する M a s k   R e g i s t e r ( x 0 、 x 1 、 . . . 、 x 7 )

R 3 に記憶されているパックされたデータ・オペランドから選択的にバイト・サイズ・データ要素を書き込むためのコード・シーケンス

10

20

30

40

50

【表 1】

命令 グループ				
1)	p1,p2=tbit r4,x0			
2)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x1
3)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x2
4)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x3
5)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x4
6)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x5
7)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x6
8)	<p1>storel r2=r3	r2=r2+1	shiftr r3=r3<<8	p1,p2=tbit r4,x7
9)	<p1>storel r2=r3			

10

## 【 0 0 2 7 】

レジスタ R 2 は、記憶位置に関連付けられた値を含んでいる。レジスタ R 3 は、この例では、8つのデータ要素に分割される64ビットのパックされたデータを含んでいる。レジスタ R 4 は、パックされたマスク・オペランドを、R 3 の8つのデータ要素に対応する x 0 ~ x 7 の8つのマスク要素と共に含んでいる。

20

## 【 0 0 2 8 】

命令の第1のグループで、コード・シーケンスは t b i t 命令を使用してプレディケイト値を決定する。この実施形態では、t b i t はパックされたマスク要素の最上位ビット x 0 をテストする。このビットが1の場合、プレディケイト p 1 は真にセットされ、そうでない場合、p 1 は偽にセットされる。代替の実施形態は、ビット値が0の場合にプレディケイト p 1 を真に等しくセットし、反対に、ビット値が1の場合にプレディケイト p 1 を偽にセットする場合がある。t b i t 命令は、対応するパックされたマスク要素の任意のビットをテストすることができることを理解されたい。

30

## 【 0 0 2 9 】

命令の第2のグループでは、R 4 の値に関連付けられる記憶位置にこの例では1バイトを条件付きで記憶するために、以前記憶されたプレディケーション値が使用される。同様に、命令の第2のグループを通じて R 2 は増分され、R 3 は右に8ビットだけシフトされ、プレディケイト p 1 は、パックされたマスク要素 x 1 が x 0 の代わりに使用されることを除いて、命令の第1のグループと同様の方法でリセットされる。多重並列処理装置は、命令の第2のグループのすべての4つの命令を1つまたは複数の同じクロック・サイクルで並列処理することを可能にする。命令の第3から第7のグループは、命令の各グループが様々なパックされたマスク要素を使用することを除いて、命令の第2のグループと同様の方法で続行する。命令の第9のグループは、R 4 内の値に関連付けられた記憶位置に最後のバイトを条件付きで記憶するためにプレディケーションを使用する。この例では、シーケンスは9クロック・サイクルを要する。しかし、クロック・サイクルの量は、プロセッサの実施態様および/またはパックされたデータ・オペランドのデータ要素の数によって異なる場合がある。

40

## 【 0 0 3 0 】

したがって、本発明は、専用の S I M D バイト・マスク・ライト回路の使用を必要としない。むしろ、プレディケーションの使用は、条件付き分岐の回避、したがって分岐の予測ミスによる性能損失の回避を可能にする。多重並列処理装置は、改善されたプロセッサ処理能力を可能にする（例えば、本発明の一実施形態は、上記各グループの命令を並列に処理するために必要な数の並列処理装置を有する）。この並列という特性とプレディケーシ

50



ョンによって、本発明は、バイト・マスク・ライトの専用SIMD回路の実装に匹敵する性能を有することができる。さらに、表1にある命令、したがってそれらをサポートするためのプロセッサの回路は、使用の分野が遥かに狭い特化/専用バイト・マスク書込み命令回路（例えば、グラフィックス処理）と比較して、より多目的に使用される。

【0031】

本発明は、処理システムにおいて様々なレベルに実装することができる。例えばプロセッサ・アーキテクチャによって、上記のコード・シーケンスは手動で符号化されプロセッサに供給される、単一命令をコンパイルするコンパイラによって生成される、単一命令の受信に回答してプロセッサ内のデコーダによって生成される等々が可能である。

【0032】

図4は、本発明の一実施形態によるコンピュータ・システムの選択された一部を示す図である。コンピュータ・システムのこの選択された一部は、プロセッサ410、記憶装置420、およびバス415を含んでいる。図4のプロセッサ410は、本発明を理解するためには必須でない付加的な回路を含んでいる。

【0033】

プロセッサ410は、2つの異なる命令セット（例えば、64ビットの命令セットと32ビットの命令セット）を実行するように設計されている。本明細書では、プロセッサによって外部から受信される命令を示すために、マクロ命令という用語を使用する。1つの具体的な実施形態では、第1の命令セット410の命令を制御信号および/またはマイクロコード・エントリポイントに復号するためにデコーダ416が使用される。この場合、第1の命令セット410の命令は、プロセッサ410によって外部から受信された場合はマクロ命令である。さらに、第2の命令セット420の命令（例えば、IA32Intel Architecture命令）を、変換後にデコーダ416によって処理される第1の命令セット410に変換するためにマイクロコード・コンバータ417が使用される。すなわち、第2の命令セット420の少なくとも特定のマクロ命令は、第1の命令セット410からの命令で構成される第1のレベルのマイクロコードに変換される。この場合、第1の命令セット410からの命令は、プロセッサ410に対して内部的に生成される場合はマイクロ命令である。一実施形態では、この変換はマイクロコードROMを使用して達成される。例えば、一実施形態では、単一SIMDバイト・マスク・ライトマイクロ命令をマイクロコード・コンバータ417に供給することができ、そこでプレディケーションを用いたマスクに応じたパックされたデータからのデータ要素を選択的に書き込むために、第1の命令セット410からの適切な命令（ここではマイクロ命令として動作する）に変換される。さらに、データ要素を選択的に書き込むための第1の命令セット410からの同じ命令を、デコーダ416によって直接的にマクロ命令として受信することもできる。

【0034】

以上、本発明を具体的な例示の実施形態を参照しながら説明した。しかし、この開示の特典を受ける人物には、本発明のより広範な趣旨および範囲から逸脱せずに、これらの実施形態に対して様々な修正形態および変更を行うことができることが明らかになる。したがって、本明細書および図面は、限定的なものではなく説明的なものとみなされるべきである。

【図面の簡単な説明】

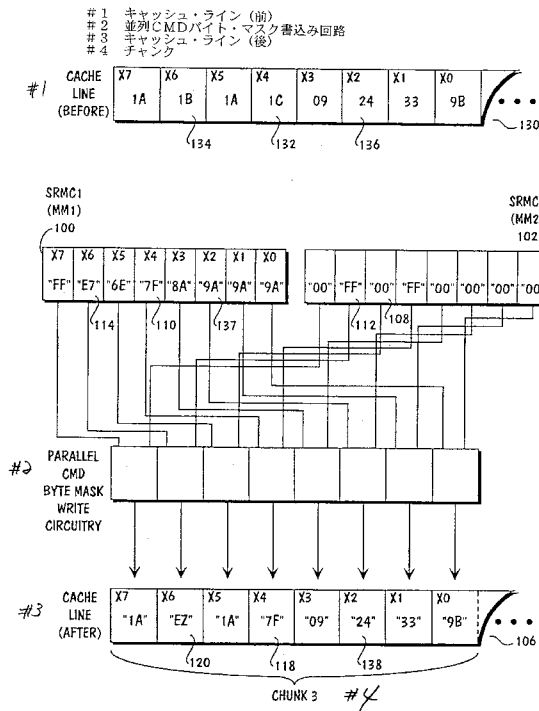
【図1】 SIMDアーキテクチャでSIMDバイト・マスク・ライト命令を実施するための専用並列回路を示すブロック図である。

【図2】 本発明の一実施形態によるコンピュータ・システムを示すブロック図である。

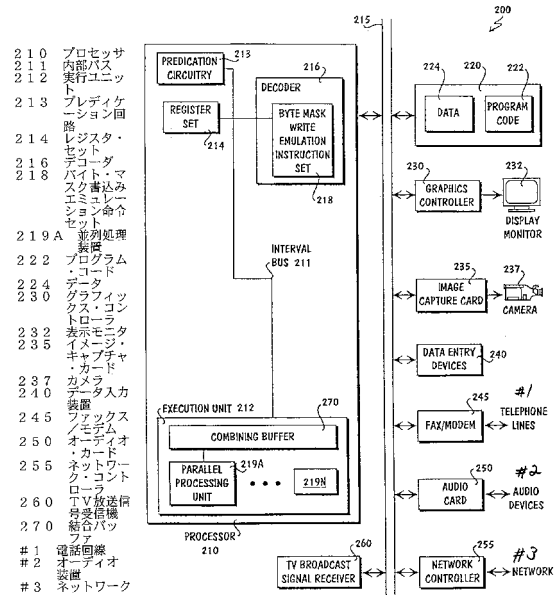
【図3】 本発明の一実施形態による流れ図である。

【図4】 本発明の一実施形態によるコンピュータ・システムの選択された部分を示す図である。

【図 1】

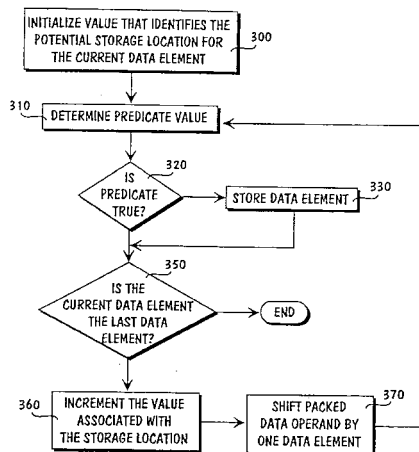


【図 2】

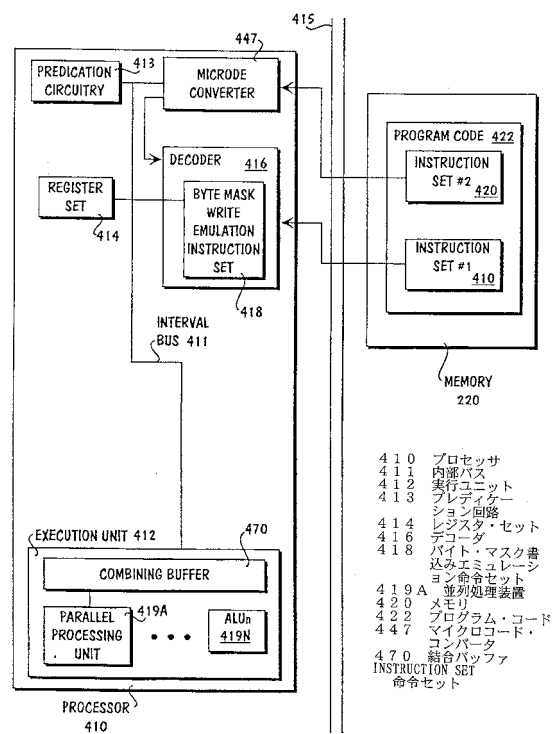


【図 3】

300 現在のデータ要素に対する、記憶位置を識別する初期値  
310 プレディケート値を決定する  
320 プレディケート値は真か  
330 データ要素を記憶する  
340 現在のデータ要素は最後のデータ要素か  
350 その記憶位置に関連付けられている値を増分する  
360 パックされたデータ・オペランドを1つのデータ要素だけシフトする



【図 4】



---

フロントページの続き

- (56)参考文献 特開平 0 2 - 2 0 4 8 6 1 ( J P , A )  
特開昭 6 0 - 0 5 9 4 6 9 ( J P , A )  
特開平 1 1 - 0 5 3 1 8 9 ( J P , A )  
特表 2 0 0 2 - 5 1 2 3 9 8 ( J P , A )  
特開平 0 9 - 1 9 8 2 3 1 ( J P , A )

- (58)調査した分野(Int.Cl. , D B 名)

G06F 9/30