

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6571752号
(P6571752)

(45) 発行日 令和1年9月4日(2019.9.4)

(24) 登録日 令和1年8月16日(2019.8.16)

(51) Int.Cl.

F 1

GO6F 17/16 (2006.01)	GO6F 17/16	D
GO6F 9/38 (2006.01)	GO6F 9/38	370C
GO6F 9/302 (2006.01)	GO6F 9/302	A
GO6F 9/315 (2006.01)	GO6F 9/315	S
GO6F 9/34 (2006.01)	GO6F 9/34	330

請求項の数 14 (全 21 頁) 最終頁に続く

(21) 出願番号	特願2017-502132 (P2017-502132)
(86) (22) 出願日	平成27年6月26日 (2015.6.26)
(65) 公表番号	特表2017-527886 (P2017-527886A)
(43) 公表日	平成29年9月21日 (2017.9.21)
(86) 國際出願番号	PCT/US2015/038013
(87) 國際公開番号	W02016/014213
(87) 國際公開日	平成28年1月28日 (2016.1.28)
審査請求日	平成30年6月8日 (2018.6.8)
(31) 優先権主張番号	62/029,039
(32) 優先日	平成26年7月25日 (2014.7.25)
(33) 優先権主張国・地域又は機関	米国(US)
(31) 優先権主張番号	14/486,326
(32) 優先日	平成26年9月15日 (2014.9.15)
(33) 優先権主張国・地域又は機関	米国(US)

(73) 特許権者	507364838 クアルコム、インコーポレイテッド アメリカ合衆国 カリフォルニア 921 21 サンディエゴ モアハウス ドラ イブ 5775
(74) 代理人	100108453 弁理士 村山 靖彦
(74) 代理人	100163522 弁理士 黒田 晋平
(72) 発明者	ルシアン・コドレスク アメリカ合衆国・カリフォルニア・921 21・サン・ディエゴ・モアハウス・ドラ イブ・5775

最終頁に続く

(54) 【発明の名称】ベクトルレジスタファイルにおいてデータインデックス方式アキュムレータを使用するベクトルプロセッサによるスカラ演算の並列化、関連する回路、方法およびコンピュータ可読媒体

(57) 【特許請求の範囲】

【請求項 1】

ベクトルプロセッサであって、

それが複数のアキュムレータを提供するように構成される、複数のベクトルレジスタを含むベクトルレジスタファイルと、

入力データベクトルを受信するための手段と、

1つまたは複数のベクトル演算を実行するための手段と

を備え、

前記1つまたは複数のベクトル演算を実行するための手段は、前記ベクトルプロセッサのベクトルレジスタファイルの複数の書き込みポートの書き込みポートごとに、

前記入力データベクトルの入力データ値にアクセスするための手段と、

前記入力データ値に基づいて、前記ベクトルレジスタファイル内の複数のベクトルレジスタのうちのベクトルレジスタを示すレジスタインデックスを決定するための手段と、

前記入力データ値に基づいて、前記ベクトルレジスタの複数のアキュムレータのうちのアキュムレータを示すアキュムレータインデックスを決定するための手段と、

前記レジスタインデックスおよび前記アキュムレータインデックスに基づいて、前記レジスタインデックスによって指示される前記ベクトルレジスタにおいてスカラ演算を実行するための手段であって、前記スカラ演算が前記アキュムレータインデックスによって指示される前記アキュムレータをインクリメントすることを含む、手段と、

前記レジスタインデックスによって指示される前記ベクトルレジスタに前記スカラ演

10

20

算の結果を書き込むための手段と
を備える、ベクトルプロセッサ。

【請求項 2】

前記ベクトルレジスタファイルは、それぞれが前記複数の書込みポートのうちの1つまたは複数の書込みポートを提供する、複数の物理レジスタファイルを含み、

前記入力データベクトルを受信するための手段は、前記入力データベクトルを、それぞれが前記複数の物理レジスタファイルのうちの1つの物理レジスタファイルに対応する、複数の入力データベクトルサブセットとして受信するように構成され、

前記1つまたは複数のベクトル演算を実行するための手段は、前記複数の物理レジスタファイルのうちの各物理レジスタファイルに対して前記1つまたは複数のベクトル演算を同時に実行することによって、前記1つまたは複数のベクトル演算を実行するように構成される、

請求項1に記載のベクトルプロセッサ。

【請求項 3】

前記レジスタインデックスを決定するための手段は、前記入力データ値の複数のビットの1つもしくは複数の論理右シフトを実行することによって、前記レジスタインデックスを決定するように構成されるか、または

アキュムレータインデックスを決定するための手段は、前記入力データ値の1つもしくは複数の上位ビットをゼロ(0)にマスクすることによって、前記アキュムレータインデックスを決定するように構成される、

請求項1に記載のベクトルプロセッサ。

【請求項 4】

前記複数の書込みポートの書込みポートごとの前記レジスタインデックスおよび前記アキュムレータインデックスのうちの1つまたは複数に基づいて、前記複数の書込みポートの対応する2つ以上の書込みポートのための2つ以上のスカラ演算をマージしてもよいと判断するための手段と、

前記2つ以上のスカラ演算を1つのマージされたスカラ演算に組み合わせるための手段とをさらに備え、

前記スカラ演算を実行するための手段は、前記マージされたスカラ演算を実行することによって前記スカラ演算を実行するように構成される、

請求項1に記載のベクトルプロセッサ。

【請求項 5】

前記スカラ演算をマージしてもよいと判断するための手段は、第1の書込みポートのための前記レジスタインデックスが前記複数の書込みポートの1つまたは複数の他の書込みポートのための前記レジスタインデックスと同一であると判断することによって、前記複数の書込みポートの前記対応する2つ以上の書込みポートのための前記2つ以上のスカラ演算をマージしてもよいと判断するように構成される、請求項4に記載のベクトルプロセッサ。

【請求項 6】

前記スカラ演算をマージしてもよいと判断するための手段は、前記第1の書込みポートのための前記アキュムレータインデックスが前記複数の書込みポートの前記1つまたは複数の他の書込みポートのための前記アキュムレータインデックスと同一であるとさらに判断することによって、前記複数の書込みポートの前記対応する2つ以上の書込みポートのための前記2つ以上のスカラ演算をマージしてもよいと判断するように構成される、請求項5に記載のベクトルプロセッサ。

【請求項 7】

集積回路(IC)、または

セットトップボックス、エンターテインメントユニット、ナビゲーションデバイス、通信デバイス、固定位置データユニット、モバイルロケーションデータユニット、モバイルフォン、セルラーフォン、コンピュータ、ポータブルコンピュータ、デスクトップコンピ

10

20

30

40

50

ユータ、携帯情報端末(PDA)、モニタ、コンピュータモニタ、テレビジョン、チューナ、ラジオ、衛星ラジオ、音楽プレーヤ、デジタル音楽プレーヤ、ポータブル音楽プレーヤ、デジタルビデオプレーヤ、ビデオプレーヤ、デジタルビデオディスク(DVD)プレーヤ、およびポータブルデジタルビデオプレーヤからなるグループから選択されるデバイスに組み込まれる、請求項1に記載のベクトルプロセッサ。

【請求項8】

ベクトルプロセッサ内のスカラ演算を並列化するための方法であって、前記ベクトルプロセッサは、それが複数のアキュムレータを提供するように構成される、複数のベクトルレジスタを含むベクトルレジスタファイルを含み、

前記ベクトルプロセッサの受信手段によって、入力データベクトルを受信するステップと、

前記ベクトルプロセッサのベクトルレジスタファイルの複数の書き込みポートの書き込みポートごとに、

前記ベクトルプロセッサのアクセス手段によって、前記入力データベクトルの入力データ値にアクセスするステップと、

前記ベクトルプロセッサの第1の決定手段によって、前記入力データ値に基づいて、前記ベクトルレジスタファイル内の複数のベクトルレジスタのうちのベクトルレジスタを示すレジスタインデックスを決定するステップと、

前記ベクトルプロセッサの第2の決定手段によって、前記入力データ値に基づいて、前記ベクトルレジスタの複数のアキュムレータのうちのアキュムレータを示すアキュムレータインデックスを決定するステップと、

前記ベクトルプロセッサの実行手段によって、前記レジスタインデックスおよび前記アキュムレータインデックスに基づいて、前記レジスタインデックスによって指示される前記ベクトルレジスタにおいてスカラ演算を実行するステップであって、前記スカラ演算が前記アキュムレータインデックスによって指示される前記アキュムレータをインクリメントすることを含む、ステップと、

前記ベクトルプロセッサの書き込み手段によって、前記レジスタインデックスによって指示される前記ベクトルレジスタに前記スカラ演算の結果を書き込むステップと

を行うことによって1つまたは複数のベクトル演算を実行するステップとを含む、方法。

【請求項9】

前記ベクトルレジスタファイルは、それが前記複数の書き込みポートのうちの1つまたは複数の書き込みポートを提供する、複数の物理レジスタファイルを含み、

前記入力データベクトルは、それが前記複数の物理レジスタファイルのうちの1つの物理レジスタファイルに対応する、複数の入力データベクトルサブセットを含み、

前記1つまたは複数のベクトル演算を実行するステップは、前記複数の物理レジスタファイルのうちの各物理レジスタファイルに対して前記1つまたは複数のベクトル演算を同時に実行するステップを含む、

請求項8に記載の方法。

【請求項10】

前記レジスタインデックスを決定するステップは、前記入力データ値の複数のビットの1つまたは複数の論理右シフトを実行するステップを含むか、または

前記アキュムレータインデックスを決定するステップは、前記入力データ値の1つまたは複数の上位ビットをゼロ(0)にマスクするステップを含む、

請求項8に記載の方法。

【請求項11】

前記ベクトルプロセッサの第3の決定手段により、前記複数の書き込みポートの書き込みポートごとの前記レジスタインデックスおよび前記アキュムレータインデックスのうちの1つまたは複数に基づいて、前記複数の書き込みポートの対応する2つ以上の書き込みポートのための2つ以上のスカラ演算をマージしてもよいと判断するステップと、

10

20

30

40

50

前記ベクトルプロセッサの組合せ手段により、前記2つ以上のスカラ演算を1つのマージされたスカラ演算に組み合わせるステップと

をさらに含み、

前記スカラ演算を実行するステップは、前記マージされたスカラ演算を実行するステップを含む、

請求項8に記載の方法。

【請求項 1 2】

前記複数の書込みポートの前記対応する2つ以上の書込みポートのための前記2つ以上のスカラ演算をマージしてもよいと判断するステップは、第1の書込みポートのための前記レジスタインデックスが前記複数の書込みポートの1つまたは複数の他の書込みポートのための前記レジスタインデックスと同一であると判断するステップを含む、請求項11に記載の方法。

10

【請求項 1 3】

前記複数の書込みポートの前記対応する2つ以上の書込みポートのための前記2つ以上のスカラ演算をマージしてもよいと判断するステップは、前記第1の書込みポートのための前記アキュムレータインデックスが前記複数の書込みポートの前記1つまたは複数の他の書込みポートのための前記アキュムレータインデックスと同一であると判断するステップをさらに含む、請求項12に記載の方法。

【請求項 1 4】

ベクトルプロセッサに、請求項8から13のいずれか一項に記載の方法を実行させるコンピュータ実行可能命令を記憶する、非一時的コンピュータ可読記録媒体。

20

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

優先権主張

本出願は、参照によりその全体が本明細書に組み込まれる、2014年7月25日に出願された"PARALLELIZATION OF SCALAR OPERATIONS BY VECTOR PROCESSORS USING DATA- INDEXED ACCUMULATORS IN VECTOR REGISTER FILES, AND RELATED CIRCUITS, METHODS, AND COMPUTER-READABLE MEDIA"と題する米国仮特許出願第62/029,039号の優先権を主張する。

【0 0 0 2】

30

本出願は、参照によりその全体が本明細書に組み込まれる、2014年9月15日に出願された"PARALLELIZATION OF SCALAR OPERATIONS BY VECTOR PROCESSORS USING DATA- INDEXED ACCUMULATORS IN VECTOR REGISTER FILES, AND RELATED CIRCUITS, METHODS, AND COMPUTER-READABLE MEDIA"と題する米国特許出願第14/486,326号の優先権を主張する。

【0 0 0 3】

本開示の技術は一般には、ベクトルプロセッサを使用する並列データ処理に関する。

【背景技術】

【0 0 0 4】

最新のコンピュータプロセッサが直面する1つの種類の計算タスクは、入力データに基づいていくつかのアキュムレータのうちの1つにおいてスカラ演算を実行することを伴い、入力データの値が、どのアキュムレータが各スカラ演算のターゲットであるかを決定する。この種類の計算タスクの非限定的な例が、ヒストグラム生成である。ヒストグラムを生成するために、プロセッサは、(たとえば、入力データ内に各データ値が現れる回数をカウントすることによって)入力データ内の個々のデータ値またはデータ値範囲に関する累積的な発生頻度を計算する。このようにして、入力データ内のデータ値の全体的な分布を特定することができ、その分布を用いて、その分布の視覚表現を生成することができる。

40

【0 0 0 5】

デジタル画像内の色調値ごとのいくつかのピクセルをプロットすることによってデジタル画像内の色調分布を図示するために、画像処理においてヒストグラムが頻繁に使用され

50

る。たとえば、デジタル画像は、それぞれが8ビット強度値を有するピクセルを含む場合がある。したがって、デジタル画像からヒストグラムを生成するには、それぞれがデジタル画像のピクセルのための取り得る強度値のうちの1つに対応する256(すなわち、 2^8)個のアキュムレータをプロセッサが使用する必要がある場合がある。プロセッサは、デジタル画像の各ピクセルを調べ、そのピクセルのための強度値を決定するために演算を実行する。その後、ピクセルのための強度値を用いて、どのアキュムレータがインクリメントされるべきであるかを決定する。

【0006】

各データ値の処理が、入力としてデータ値を受信することと、データ値に従ってアキュムレータの値を検索することと、検索された値に関して実行されるスカラ演算に基づいて新たな値をアキュムレータに書き込むこととを伴うので、ヒストグラム生成のような計算タスクは、膨大な計算量になる場合がある。したがって、各データ値は、処理するのに複数のプロセッサロックサイクルを必要とする場合がある。さらに、入力データの処理は、アキュムレータを更新する帯域幅の利用可能性によってさらに制限される場合がある。たとえば、アキュムレータが記憶されるデータキャッシュが、各プロセッサロックサイクル中に、限られた数の読み出しパスおよび/または書き込みパスしか提供しない場合がある。

10

【0007】

この種類の計算タスクを最適化するための1つの手法は、複数命令複数データ(MIMD)技法を用いてスカラ演算を並列化するためにマルチコア処理を使用することを伴う。この手法の下では、マルチコアプロセッサの各処理スレッドがアキュムレータのプライベートセットを提供し、入力データの1つのセクションを処理する。その後、すべての処理スレッドが、入力データの個々の部分に関する処理を完了した後に、処理スレッドそれぞれのアキュムレータが「削減される」、すなわち、集約される。しかしながら、この手法は結果として、処理スレッド間の依存性問題および/またはメモリ衝突を引き起こす場合があり、さらなる処理クラスタが使用されても、最小限の性能向上しか与えない場合がある。

20

【発明の概要】

【課題を解決するための手段】

【0008】

詳細な説明において開示される態様は、ベクトルレジスタファイルにおいてデータインデックス方式アキュムレータを使用してベクトルプロセッサによるスカラ演算を並列化することを含む。関連する回路、方法、およびコンピュータ可読媒体も開示される。この関連で、一態様では、ベクトルプロセッサが、スカラ演算を並列化するための単一命令複数データ(SIMD)機能を提供するように構成される。ベクトルプロセッサは、複数のベクトルレジスタを提供するベクトルレジスタファイルを含む。各ベクトルレジスタは複数のアキュムレータに論理的に細分される。複数のベクトルレジスタ内のアキュムレータの合計数は、予想される入力データ内の取り得るデータ値の数に対応する。また、ベクトルレジスタファイルは、各プロセッサロックサイクル中にベクトルレジスタに複数の書き込みを行うことができるようになる複数の書き込みポートを提供する。スカラ演算を並列化できるようにするために、ベクトルプロセッサは、入力データベクトルを受信するように構成される。その後、ベクトルプロセッサは、ベクトルレジスタファイルの書き込みポートごとに入力データ値(たとえば、入力データベクトルのサブセット)にアクセスするためにベクトル演算を実行する。入力データ値ごとに、レジスタインデックスおよびアキュムレータインデックスが決定される。合わせて、レジスタインデックスおよびアキュムレータインデックスは、適切なアキュムレータへのマッピングとしての役割を果たす場合もあり、レジスタインデックスはアキュムレータを含むベクトルレジスタを示し、アキュムレータインデックスは、ベクトルレジスタ内の特定のアキュムレータを示す。したがって、入力データ値が、作用を受けるアキュムレータを決定するという点で、アキュムレータは「データインデックス方式」とみなすことができる。スカラ演算が、レジスタインデックスによって指示されるベクトルレジスタにおいて実行され、具体的なスカラ演算はレジスタインデッ

30

40

50

クスおよびアキュムレータインデックスに基づく。このようにして、ベクトルプロセッサは、処理帯域幅を増やすために、ベクトルレジスタファイルのデータ並列化能力を利用することができ、それにより、全体的な処理能力を高めることができる。

【0009】

別の態様では、ベクトルレジスタファイルを含むベクトルプロセッサが提供される。ベクトルレジスタファイルは、それぞれが複数のアキュムレータを提供するように構成される複数のベクトルレジスタを含む。また、ベクトルレジスタファイルは、複数の書込みポートを提供するように構成される。ベクトルプロセッサは、入力データベクトルを受信するように構成される。複数の書込みポートの書込みポートごとに、ベクトルプロセッサは、入力データベクトルの入力データ値にアクセスするために1つまたは複数のベクトル演算を実行するように構成される。ベクトルプロセッサはさらに、複数の書込みポートの書込みポートごとに、入力データ値に基づいて、複数のベクトルレジスタのうちのベクトルレジスタを示すレジスタインデックスを決定するように構成される。また、ベクトルプロセッサは、複数の書込みポートの書込みポートごとに、入力データ値に基づいて、ベクトルレジスタの複数のアキュムレータのうちのアキュムレータを示すアキュムレータインデックスを決定するように構成される。ベクトルプロセッサはさらに、レジスタインデックスおよびアキュムレータインデックスに基づいて、レジスタインデックスによって指示されるベクトルレジスタにおいてスカラ演算を実行するように構成される。

10

【0010】

別の態様では、入力データベクトルを受信するための手段を備えるベクトルプロセッサが提供される。ベクトルプロセッサはさらに、ベクトルプロセッサのベクトルレジスタファイルの複数の書込みポートの書込みポートごとに、入力データベクトルの入力データ値にアクセスするための手段を備える。また、ベクトルプロセッサは、複数の書込みポートの書込みポートごとに、入力データ値に基づいて、ベクトルレジスタファイル内の複数のベクトルレジスタのうちのベクトルレジスタを示すレジスタインデックスを決定するための手段を備える。ベクトルプロセッサはさらに、複数の書込みポートの書込みポートごとに、入力データ値に基づいて、ベクトルレジスタの複数のアキュムレータのうちのアキュムレータを示すアキュムレータインデックスを決定するための手段を備える。ベクトルプロセッサはさらに、複数の書込みポートの書込みポートごとに、レジスタインデックスおよびアキュムレータインデックスに基づいて、レジスタインデックスによって指示されるベクトルレジスタにおいてスカラ演算を実行するための手段を備える。

20

【0011】

別の態様では、ベクトルプロセッサにおけるスカラ演算を並列化するための方法が提供される。本方法は、ベクトルプロセッサによって、入力データベクトルを受信するステップを含む。本方法はさらに、ベクトルプロセッサのベクトルレジスタファイルの複数の書込みポートの書込みポートごとに、入力データベクトルの入力データ値にアクセスするために1つまたは複数のベクトル演算を実行するステップを含む。また、本方法は、複数の書込みポートの書込みポートごとに、入力データ値に基づいて、ベクトルレジスタファイル内の複数のベクトルレジスタのうちのベクトルレジスタを示すレジスタインデックスを決定するために1つまたは複数のベクトル演算を実行するステップを含む。本方法はさらに、複数の書込みポートの書込みポートごとに、入力データ値に基づいて、ベクトルレジスタの複数のアキュムレータのうちのアキュムレータを示すアキュムレータインデックスを決定するために1つまたは複数のベクトル演算を実行するステップを含む。本方法はさらに、複数の書込みポートの書込みポートごとに、レジスタインデックスおよびアキュムレータインデックスに基づいて、レジスタインデックスによって指示されるベクトルレジスタにおいてスカラ演算を実行するために1つまたは複数のベクトル演算を実行するステップを含む。

30

【0012】

別の実施形態では、コンピュータ実行可能命令を記憶する非一時的コンピュータ可読媒体が提供される。コンピュータ実行可能命令は、ベクトルプロセッサに、入力データベク

40

50

トルを受信させる。コンピュータ実行可能命令はさらに、ベクトルプロセッサに、ベクトルプロセッサのベクトルレジスタファイルの複数の書き込みポートの書き込みポートごとに、入力データベクトルの入力データ値にアクセスするために1つまたは複数のベクトル演算を実行させる。また、コンピュータ実行可能命令は、ベクトルプロセッサに、複数の書き込みポートの書き込みポートごとに、入力データ値に基づいて、ベクトルレジスタファイル内の複数のベクトルレジスタのうちのベクトルレジスタを示すレジスタインデックスを決定するために1つまたは複数のベクトル演算を実行させる。コンピュータ実行可能命令はさらに、ベクトルプロセッサに、複数の書き込みポートの書き込みポートごとに、入力データ値に基づいて、ベクトルレジスタの複数のアキュムレータのうちのアキュムレータを示すアキュムレータインデックスを決定するために1つまたは複数のベクトル演算を実行させる。10。コンピュータ実行可能命令はさらに、ベクトルプロセッサに、複数の書き込みポートの書き込みポートごとに、レジスタインデックスおよびアキュムレータインデックスに基づいて、レジスタインデックスによって指示されるベクトルレジスタにおいてスカラ演算を実行するために1つまたは複数のベクトル演算を実行させる。

【図面の簡単な説明】

【0013】

【図1】スカラ演算を並列化するためのベクトルプロセッサを含むことができる例示的なコンピュータプロセッサのブロック図である。

【図2】スカラ演算の並列化のための図1のベクトルプロセッサ内の例示的な通信フローおよび論理要素のブロック図である。20

【図3】入力データ値を処理する、図1のベクトルプロセッサの例示的な結果を示す図である。

【図4】ベクトルレジスタファイルにおいてデータインデックス方式アキュムレータを使用して図1のベクトルプロセッサによるスカラ演算を並列化するための例示的なプロセスを示す流れ図である。

【図5】図1のベクトルプロセッサによるスカラ演算をマージするためのさらなる例示的な演算を示す流れ図である。

【図6】図1のベクトルプロセッサを含むことができる例示的なプロセッサベースシステムのブロック図である。

【発明を実施するための形態】

【0014】

ここで図面を参照しながら、本開示のいくつかの例示的な態様が説明される。「例示的」という語は、本明細書において「一例、事例、または例示としての役割を果たすこと」を意味するために使用される。「例示的」として本明細書において説明されるいずれの態様も、必ずしも他の態様よりも好ましいか、または有利であると解釈されるべきではない。

【0015】

詳細な説明において開示される態様は、ベクトルレジスタファイルにおいてデータインデックス方式アキュムレータを使用してベクトルプロセッサによるスカラ演算を並列化することを含む。関連する回路、方法、およびコンピュータ可読媒体も開示される。この関連で、一態様では、ベクトルプロセッサが、スカラ演算を並列化するための単一命令複数データ(SIMD)機能を提供するように構成される。ベクトルプロセッサは、複数のベクトルレジスタを提供するベクトルレジスタファイルを含む。各ベクトルレジスタは複数のアキュムレータに論理的に細分される。複数のベクトルレジスタ内のアキュムレータの合計数は、予想される入力データ内の取り得るデータ値の数に対応する。また、ベクトルレジスタファイルは、各プロセッサクロックサイクル中にベクトルレジスタに複数の書き込みを行うことができるようになる複数の書き込みポートを提供する。スカラ演算を並列化できるようになるために、ベクトルプロセッサは、入力データベクトルを受信するように構成される。その後、ベクトルプロセッサは、ベクトルレジスタファイルの書き込みポートごとに入力データ値(たとえば、入力データベクトルのサブセット)にアクセスするためにベクトル40

演算を実行する。入力データ値ごとに、レジスタインデックスおよびアキュムレータインデックスが決定される。合わせて、レジスタインデックスおよびアキュムレータインデックスは、適切なアキュムレータへのマッピングとしての役割を果たす場合もあり、レジスタインデックスはアキュムレータを含むベクトルレジスタを示し、アキュムレータインデックスは、ベクトルレジスタ内のアキュムレータを示す。したがって、入力データ値が、作用を受けるアキュムレータを決定するという点で、アキュムレータは「データインデックス方式」とみなすことができる。スカラ演算が、レジスタインデックスによって指示されるベクトルレジスタにおいて実行され、具体的なスカラ演算はレジスタインデックスおよびアキュムレータインデックスに基づく。このようにして、ベクトルプロセッサは、処理帯域幅を増やすために、ベクトルレジスタファイルのデータ並列化能力を利用することができます、それにより、全体的な処理能力を高めることができる。

【0016】

これに関連して、図1は、スカラ演算を並列化するための例示的なベクトルプロセッサ12を含むコンピュータプロセッサ10の概略図である。以下にさらに詳細に説明されるように、ベクトルプロセッサ12は、本明細書において開示される例示的なベクトル処理演算を含むベクトル処理演算を提供する他の特定の例示的な回路および機能とともに、実行ユニット14およびベクトルレジスタファイル16を含む。コンピュータプロセッサ10およびそのベクトルプロセッサ12は、数ある素子の中でも、既知のデジタル論理素子、半導体回路、処理コアおよび/またはメモリ構造のいずれか1つ、またはその組合せを含む場合がある。

【0017】

図2で開始するベクトルレジスタファイル16を使用するスカラ演算の並列化のための本開示における特定の回路と、ベクトルプロセッサ12によって提供されるように構成されるベクトル処理演算とを論じる前に、図1のコンピュータプロセッサ10の構成要素が最初に説明される。先に言及されたように、ベクトルプロセッサ12は、ベクトルレジスタ18(0)～18(X)を提供するベクトルレジスタファイル16を含む。ベクトルレジスタ18(0)～18(X)は、ベクトルプロセッサ12によってSIMDのように実行されるベクトル命令による作用を受けるデータを記憶することができる。いくつかの態様において、ベクトルレジスタファイル16は、複数の物理レジスタファイル(図示せず)を用いて実現することができ、各物理レジスタファイルはプロセッサクロックサイクルあたり複数の読み出しポート20(0)～20(Y)と、複数の書き込みポート22(0)～22(Z)とを提供する。物理レジスタファイルは、たとえば、高速ランダムアクセスメモリ(RAM)を含むことができる。ベクトルレジスタファイル16は、それぞれが自らの読み出しポートおよび書き込みポートを提供する2つ以上の物理レジスタファイルを含むことができる。たとえば、ベクトルレジスタファイル16は、512ビット幅とすることができます、それぞれがプロセッサクロックサイクルあたり8つの読み出しポートおよび4つの書き込みポートを提供する、4つの128ビット幅物理レジスタファイルとして実現することができる。ベクトルプロセッサ12の例示される要素をサポートするために、コンピュータプロセッサ10および/またはベクトルプロセッサ12は、明確にするために図1では省略される他の付加構成要素を含むことができる。

【0018】

引き続き図1を参照すると、この態様におけるコンピュータプロセッサ10は、ベクトルプロセッサ12によって提供されるベクトル処理に加えて、コンピュータプロセッサ10においてスカラ処理を提供するスカラプロセッサ24も含む。このようにして、コンピュータプロセッサ10は、極めて効率的な演算のために、実行される命令のタイプに基づいて、ベクトル命令演算およびスカラ命令演算の両方をサポートするように構成することができる。いくつかの態様では、スカラプロセッサ24は、非限定的な例として、32ビット縮小命令セットコンピューティング(RISC: reduced instruction set computing)スカラプロセッサとすることができます。スカラプロセッサ24は、この例では、スカラ命令処理をサポートするための算術論理演算ユニット(ALU)26を含む。スカラプロセッサ24は、スカラ命令を実行するときにスカラプロセッサ24によって使用するための汎用レジスタ(GPR)28も含むことができる。汎用レジスタ28は、矢印30によって示されるように、ALU26と通信可能に接

10

20

30

40

50

続することができる。スカラプロセッサ24はさらに、矢印34によって示されるように、ALU26に通信可能に結合されるデータキャッシュ32を含むことができる。データキャッシュ32は、ALU26によって頻繁に使用されるデータを高速に記憶および検索するために使用される場合がある。

【0019】

コンピュータプロセッサ10はさらに、矢印40によって示されるように、プログラムメモリ38から命令をフェッチするように構成される命令ディスパッチ回路36を含む。命令ディスパッチ回路36は、フェッチされた命令を復号することができる。命令のタイプに基づいて、命令ディスパッチ回路36は、フェッチされた命令を、スカラデータバス42を介してスカラプロセッサ24に送るか、またはベクトルデータバス44を通してベクトルプロセッサ12に送ることができる。

10

【0020】

先に論じられたように、コンピュータプロセッサ10が直面する場合がある1つの種類の計算タスクは、データインデックス方式アキュムレータ(すなわち、入力データの値が、どのアキュムレータがスカラ演算のターゲットであるかを決定するアキュムレータ)においてスカラ演算を実行することを伴う。スカラプロセッサ24を用いてこれらの計算タスクを実行する結果として、準最適なプロセッサ性能になる場合がある。一例として、各入力データ値は、スカラプロセッサ24が処理するのに複数のプロセッサクロックサイクルを必要とする場合がある。入力データの処理は、アキュムレータを更新するためにスカラプロセッサ24が利用可能な帯域幅によってさらに制限される場合がある。たとえば、アキュムレータは、各プロセッサクロックサイクル中に限られた数の読み出しポート(図示せず)および/または書き込みポート(図示せず)しか提供しない場合があるデータキャッシュ32内に記憶される場合がある。

20

【0021】

この関連で、図1のベクトルプロセッサ12は、ベクトルレジスタファイル16におけるデータインデックス方式アキュムレータを使用するスカラ演算の並列化を提供するように構成される。スカラ演算の並列化中のベクトルプロセッサ12内の例示的な通信フローおよび論理要素を例示するために、図2が提供される。例示するために、ベクトルプロセッサ12によって実行されることになる計算タスクは、それぞれが8ビット強度値を有するピクセルを含むデジタル画像のためのヒストグラムを生成することであると仮定する。

30

【0022】

図2の例では、ベクトルプロセッサ12のベクトルレジスタファイル16は、ベクトルレジスタ18(0)～18(31)と呼ばれる、全部で32個のベクトルレジスタ18を含む。ベクトルレジスタ18はそれぞれ8個のアキュムレータ("ACC")46に論理的に細分され、ベクトルレジスタ18(0)はアキュムレータ46(0)～46(7)を含み、ベクトルレジスタ18(1)はアキュムレータ46(8)～46(15)を含むなどであり、最後に、ベクトルレジスタ18(31)はアキュムレータ46(24)～46(255)を含む。アキュムレータ46(0)～46(255)はそれぞれ、0～255の強度を有するデジタル画像内のピクセルの総数を指示する値を記憶する。したがって、この例において、ベクトルプロセッサ12によって並列化されることになるスカラ演算は、アキュムレータ46(0)～46(255)のうちの1つまたは複数に記憶される値をインクリメントすることを含む。いくつかの態様では、並列化されることになるスカラ演算は、アキュムレータ46(0)～46(255)のうちの1つまたは複数に記憶される値に関して実行されることになるさらなるおよび/または異なる算術演算および/または論理演算を含む場合があることを理解されたい。

40

【0023】

この例におけるベクトルレジスタファイル16は、全部で4つの書き込みポート22(0)～22(3)を提供し、それにより、各プロセッサクロックサイクル中にベクトルレジスタ18のうちの最大4つを更新できるようにする。書き込みポート22(0)～22(3)は、図1の書き込みポート22(0)～22(Z)に対応し、書き込みポート22(0)～22(Z)の機能に対応する機能を提供する。ベクトルレジスタファイル16のいくつかの態様は、図2において例示されるより多くの、また

50

は少ない書込みポート22を提供する場合がある。いくつかの態様では、図2に示されるベクトルレジスタファイル16は、それぞれが書込みポート22を提供する、複数の物理レジスタファイル(図示せず)のうちの1つに対応する場合がある。したがって、ベクトルレジスタファイル16は、図2には示されないさらなる物理レジスタファイルおよびさらなる書込みポート22をさらに備える場合がある。

【0024】

ベクトルプロセッサ12は、ヒストグラムが生成されることになるデジタル画像内のピクセル強度値のセットを表す入力データベクトル19を受信する。この例では、入力データベクトル19は、0～15の番号を付された16個の値を含むベクトルである。いくつかの態様によれば、入力データベクトル19は、16個の8ビット値からなる128ビットベクトルとすることができる。入力データベクトル19は、非限定的な例として、入力ストリーム(図示せず)として受信される場合があるか、またはベクトルプロセッサ12がアクセス可能であるレジスタまたは他のメモリ(図示せず)内に記憶される場合がある。いくつかの態様では、入力データベクトル19は、図2に示されるより多くの、もしくは少ないビットおよび/または多くのもしくは少ない値を含む場合がある。

【0025】

書込みポート22(0)～22(3)ごとに、ベクトルプロセッサ12は、対応するマルチプレクサ論理ブロック48(0)～48(3)を提供する。いくつかの態様では、マルチプレクサ論理ブロック48(0)～48(3)は、本明細書において説明される演算を実行するためのベクトル命令を規定するマイクロコードとして実現される場合がある。各マルチプレクサ論理ブロック48は、入力として、入力データベクトル19のサブセットを受信し、そのサブセットから、プロセッサクロックサイクル中に処理するための1つの入力データ値50を選択する。図2の例では、マルチプレクサ論理ブロック48はそれぞれ、入力データベクトル19から4つの値を受信し、4つの値から1つの入力データ値50を選択する。したがって、入力データベクトル19全体を、ベクトルプロセッサ12によって4プロセッサクロックサイクルにおいて処理することができる。

【0026】

先に論じられたように、ヒストグラムを生成するとき、入力データ値50はそれぞれ、どのアキュムレータ46がインクリメントされるべきであるかを指示する。たとえば、入力データ値50(0)が255(255のピクセル強度を指示する)である場合には、ベクトルプロセッサ12は、対応するアキュムレータ46(255)をインクリメントさせるべきである。アキュムレータ46は、ベクトルレジスタ18(0)～18(31)内に記憶されるので、どの特定のベクトルレジスタ18がインクリメントされるべきアキュムレータ46を含むかを判断するのに、ベクトルプロセッサ12は入力データ値50を復号する必要がある。したがって、図2に示されるようなベクトルプロセッサ12は、マルチプレクサ論理ブロック48(0)～48(3)ごとにレジスタデコーダ("REG DEC")52(0)～52(3)を提供する。いくつかの態様では、レジスタデコーダ52(0)～52(3)は、本明細書において説明される演算を実行するためのベクトル命令を規定するマイクロコードとして実現される場合がある。

【0027】

レジスタデコーダ52(0)～52(3)はそれぞれ、対応するマルチプレクサ論理ブロック48(0)～48(3)からそれぞれの入力データ値50(0)～50(3)を受信し、レジスタインデックス54(0)～54(3)をそれぞれ生成するために、入力データ値50(0)～50(3)に関する演算を行う。レジスタインデックス54(0)～54(3)はそれぞれ、インクリメントされるべきアキュムレータ46を含むベクトルレジスタ18のうちの1つを示す。図2の例では、レジスタデコーダ52はそれぞれ、0～31の範囲内の値を有するレジスタインデックス54を生成するために、対応する入力データ値50に関する演算を行う。この例における入力データ値50はそれぞれ8ビット値であるので、レジスタインデックス54を生成することは、各入力データ値50の8ビットを3ビットだけ論理的に右にシフトすることによって達成することができる。レジスタインデックス54を生成するために、さらなるおよび/または異なる演算がレジスタデコーダ52によって実行される場合があることを理解されたい。

10

20

30

40

50

【0028】

図2において見られるように、ベクトルレジスタ18(0)～18(31)はそれぞれ8個のアキュムレータ46を含む。したがって、レジスタインデックス54によって識別されるベクトルレジスタ18内のどの特定のアキュムレータ46がインクリメントされるべきであるかを判断するために、ベクトルプロセッサ12は入力データ値50を復号することがさらに必要である。この関連で、ベクトルプロセッサ12は、マルチプレクサ論理ブロック48(0)～48(3)ごとにアキュムレータデコーダ("ACC DEC")56(0)～56(3)を提供する。いくつかの態様は、アキュムレータデコーダ56(0)～56(3)が、本明細書において説明される演算を実行するためのベクトル命令を規定するマイクロコードとして実現されることを規定する場合がある。

【0029】

アキュムレータデコーダ56(0)～56(3)は、対応するマルチプレクサ論理ブロック48(0)～48(3)からそれぞれの入力データ値50(0)～50(3)を受信し、アキュムレータインデックス58(0)～58(3)をそれぞれ生成するために、入力データ値50(0)～50(3)に関する演算を行う。アキュムレータインデックス58(0)～58(3)はそれぞれ、レジスタインデックス54の対応する1つによって識別されるベクトルレジスタ18のうちの1つ内のアキュムレータ46のうちの1つを示す。この例では、アキュムレータインデックス58がそれぞれ、レジスタインデックス54のうちの対応する1つによって識別されるベクトルレジスタ18のうちの1つ内の8個のアキュムレータ46のうちのいずれがインクリメントされるべきであるかを指示する、0～7の範囲内の値(すなわち、3ビット値)でなければならない。したがって、アキュムレータインデックス58を生成することは、各入力データ値50の上位5ビットをマスクすることによって(たとえば、論理AND演算を実行することによって)達成することができる。アキュムレータインデックス58を生成するのに、アキュムレータデコーダ56によってさらなるおよび/または異なる演算が実行される場合があることを理解されたい。

【0030】

レジスタインデックス54およびアキュムレータインデックス58が生成されると、ベクトルプロセッサ12は、ベクトルレジスタ18に関するスカラ演算を適切に実行するために、ベクトル命令(図示せず)を合成することができる。図2の例では、実行されている計算タスクがヒストグラム生成である場合、ベクトルプロセッサ12は、レジスタインデックス54および対応するアキュムレータインデックス58に基づいて、アキュムレータ46をインクリメントするためにベクトル命令を実行することができる。いくつかの態様では、複数のアキュムレータ46をインクリメントすることは、ベクトルレジスタ18のうちの1つのベクトルレジスタのレジスタ値を検索するための演算と、結果値を生成するために、レジスタ値と、アキュムレータインデックス58のうちの対応するアキュムレータインデックスとに基づいて、スカラ演算(この例では、加算)を実行するための演算と、結果値をベクトルレジスタ18に記憶するための演算とを含むことができる。いくつかの態様が、結果値が、この例において論じられる以外のさらなるおよび/または異なるスカラ演算によって生成される場合があることを規定する場合があることを理解されたい。スカラ演算は、たとえば、結果値を生成するために、入力データ値50(0)～50(3)のうちの1つの一部として含まれる重み値(図示せず)がベクトルレジスタ18のうちの1つのベクトルレジスタのレジスタ値に加算される、重み付きヒストグラムを計算することを含む場合があることをさらに理解されたい。

【0031】

先に論じられたように、図2の例では、各プロセッサクロックサイクル中にベクトルプロセッサ12によって4つのスカラ演算(書込みポート22(0)～22(3)ごとに1つ)を達成することができる。状況によっては、ベクトルプロセッサ12は、スカラ演算のうちの1つまたは複数をマージしてもよいと判断することができる。たとえば、入力データ値50のうちの2つ以上が異なる場合があるが、ベクトルレジスタ18のうちの1つ内のアキュムレータ46のうちの2つ以上が単一のプロセッサクロックサイクル中にインクリメントされることが必要な場合があるという点で、値が十分に近い場合がある。入力データ値50のうちの2つ以上が同じである場合もあり、その場合、アキュムレータ46のうちの同じアキュムレータが

10

20

30

40

50

、プロセッサクロックサイクル中に複数回インクリメントされることが必要な場合がある。したがって、ベクトルプロセッサ12は、いくつかの態様では、スカラ演算のうちの1つまたは複数が、単一のマージされたスカラ演算にマージしてもよいか否かを判断するための衝突マージ論理ブロック60を提供することができる。いくつかの態様では、衝突マージ論理ブロック60は、本明細書において説明される演算を実行するためのベクトル命令を規定するマイクロコードとして実現される場合がある。

【0032】

いくつかの態様は、衝突マージ論理ブロック60が、書込みポート22のうちの2つ以上に10 対応するレジスタインデックス54の2つ以上が同一であると判断する場合があることを規定する場合がある。そのような場合には、ベクトルプロセッサ12は、一致するレジスタインデックス54によって指示されるベクトルレジスタ18内の複数のアキュムレータ46に関して実行されるべきスカラ演算を1つのマージされたスカラ演算にマージするベクトル命令を合成することができる。いくつかの態様では、衝突マージ論理ブロック60は、一致するレジスタインデックス54に対応するアキュムレータインデックス58のうちの2つ以上も同一であるとさらに判断する場合がある。それに応じて、ベクトルプロセッサ12は、一致するアキュムレータインデックス58によって指示されるアキュムレータ46に関して実行されるべきスカラ演算をマージされたスカラ演算にマージする(たとえば、アキュムレータ46を2つ以上だけインクリメントする)ベクトル命令を合成することができる。

【0033】

先に言及されたように、いくつかの態様では、図1のベクトルレジスタファイル16は、20 それぞれが複数の書込みポート22のうちの1つまたは複数の書込みポートを提供する、複数の物理レジスタファイル(図示せず)として実現される場合がある。そのような態様では、ベクトルプロセッサ12は、複数の物理レジスタファイルのそれぞれに対して図2に関して先に説明された演算を同時に実行するように構成することができ、その結果として、より高度な並列化が行われる。非限定的な例として、入力データベクトル19は、それぞれが複数の物理レジスタファイルのうちの1つに対応する場合がある、複数の入力データベクトルサブセット(図示せず)として受信される場合がある。その際、ベクトルプロセッサ12は、入力データベクトルサブセットを同時に処理するために、各物理レジスタファイルの各書込みポート22に対して同時にベクトル演算を実行することができる。このようにして、複数の物理レジスタファイルをそれぞれ、入力データの一部を並列処理して、入力データのその部分に対応するヒストグラムにするための個別の「レーン」として使用することができる。ベクトルプロセッサ12は、いくつかの態様において、その後、入力データの全体に関するヒストグラムを作成するために、各物理レジスタファイルのベクトルレジスタ18内の対応するアキュムレータ46を総計することができる。

【0034】

図3は、入力データベクトル64の入力データ値62を処理する、図2のベクトルプロセッサ12の例示的な結果を示す。明確にするために、図3は、図2のベクトルレジスタファイル16の書込みポート22のうちの1つを介してのスカラ演算に対応する、単一の入力データ値62が処理されることのみを示す。しかしながら、ベクトルプロセッサ12は、図2のベクトルレジスタファイル16の複数の書込みポート22に関して、図3によって例示される処理の複数のインスタンスを並列に実行できることを理解されたい。

【0035】

図3の例では、ベクトルプロセッサ12のベクトルレジスタファイル16は、ベクトルレジスタ18(0)～18(31)と呼ばれる、全部で32個のベクトルレジスタ18を含む。ベクトルレジスタ18はそれぞれ8個のアキュムレータ("ACC")46に論理的に細分され、ベクトルレジスタ18(0)はアキュムレータ46(0)～46(7)を含むなどであり、最後に、ベクトルレジスタ18(31)はアキュムレータ46(248)～46(255)を含む。アキュムレータ46はそれぞれ、処理が始まる前に、ゼロ(0)の値に初期化される。

【0036】

図3において、ベクトルプロセッサ12は入力データベクトル64を受信し、入力データベ

10

20

30

40

50

クトル64の8ビット値66(0)～66(3)を含むサブセットをマルチプレクサ論理ブロック68に提供することができる。マルチプレクサ論理ブロック68は、入力データ値62として、値66のうちの1つ(この例では、値66(3))を選択する。図3において見られるように、入力データ値62は、250のピクセル強度値を表す、1111 1010の2値(10進数において250)を有する。その後、入力データ値62は、入力データ値62を適切なベクトルレジスタ18および適切なアキュムレータ46にマッピングするために、レジスタデコーダ("REC DEC")70およびアキュムレータデコーダ("ACC DEC")72に提供される。

【 0 0 3 7 】

この例では、レジスタデコーダ70は、入力データ値62の複数のビットを3ビットだけ論理的に右にシフトすることによって、入力データ値62のためのレジスタンディックス74を生成する。その結果は、図3において見られるように、0001 1111の2値(10進数において31)を有するレジスタンディックス74である。したがって、レジスタンディックス74は、ベクトルレジスタ18(31)が、ベクトルプロセッサ12によってインクリメントされるべきアキュムレータ46を含むことを指示する。アキュムレータデコーダ72は、入力データ値62の上位5ビットをマスクすることによって、アキュムレータインディックス76を生成する。これは、入力データ値62および2値111(10進数において7)に関する論理AND演算を実行することによって達成される。結果として生じるアキュムレータインディックス76は、0000 0010の2値または10進数において2を有する。アキュムレータインディックス76に基づいて、ベクトルプロセッサ12は、ベクトルレジスタ18(31)の第3のアキュムレータ46(0から始めて、右から数える)がスカラ演算のターゲットであると判断することができる。図3に示されるように、これは、アキュムレータ46(250)に対応する。それに応じて、ベクトルプロセッサ12は、スカラ演算を実行する(この場合には、アキュムレータ46(250)を1だけインクリメントする)ベクトル命令を実行する。

【 0 0 3 8 】

図4は、ベクトルレジスタファイル16内のアキュムレータ46を使用して、図2のベクトルプロセッサ12によるスカラ演算を並列化するための例示的なプロセスを示す流れ図である。明確にするために、図4を説明する際に図2および図3の要素が参照される。図4において、ベクトルプロセッサ12が入力データベクトル64を受信することから演算が始まる(ブロック78)。その後、ベクトルプロセッサ12は、ブロック80によって示されるように、ベクトルプロセッサ12のベクトルレジスタファイル16の複数の書き込みポート22の書き込みポート22(0)～22(Z)ごとに演算を実行するためにベクトル命令を実行する。ブロック80において例示される演算は、複数の書き込みポート22のそれぞれに対して同時に実行される場合があることを理解されたい。

【 0 0 3 9 】

ベクトルプロセッサ12は最初に、入力データベクトル64の入力データ値62にアクセスする(ブロック82)。先に論じられたように、入力データ値62は、入力データベクトル64のサブセットを表すことができる。入力データ値62に基づいて、ベクトルプロセッサ12は、ベクトルレジスタファイル16内の複数のベクトルレジスタ18のうちのベクトルレジスタ18(31)を示すレジスタンディックス74を決定する(ブロック84)。いくつかの態様では、ベクトルプロセッサ12は、入力データ値62の複数のビットの1つまたは複数の論理右シフトを実行することによって、レジスタンディックス74を決定することができる(ブロック86)。また、ベクトルプロセッサ12は、入力データ値62に基づいて、ベクトルレジスタ18の複数のアキュムレータ46のうちのアキュムレータ46(250)を示すアキュムレータインディックス76を決定する(ブロック88)。いくつかの態様は、ベクトルプロセッサ12が、入力データ値62の1つまたは複数の上位ビットをゼロ(0)にマスクことによって、アキュムレータインディックス76を決定すること(ブロック90)を規定することができる。入力データ値62に基づいて、レジスタンディックス74および/またはアキュムレータインディックス76を決定するために、ベクトルプロセッサ12によってさらなるおよび/または異なる演算が実行される場合があることを理解されたい。

【 0 0 4 0 】

10

20

30

40

50

ベクトルプロセッサ12は、その後、レジスタインデックス74およびアキュムレータインデックス76に基づいて、レジスタインデックス74によって指示されるベクトルレジスタ18(31)においてスカラ演算を実行する(ブロック92)。非限定的な例として、スカラ演算は、アキュムレータインデックス76によって指示されるアキュムレータ46をインクリメントする演算を含むことができる。いくつかの態様では、スカラ演算は、アキュムレータ46に関するさらなるおよび/または他の算術および/または論理演算を含む場合がある。たとえば、スカラ演算は、入力データ値62の一部として含まれる重み値(図示せず)に従って、インクリメントされるアキュムレータ46内の重み付きヒストグラムを計算することを含むことができる。

【0041】

10

図2に関して先に論じられたように、ベクトルプロセッサ12は、単一のプロセッサクロックサイクル中に実行されるべきスカラ演算をマージすることによって、いくつかの状況において性能を改善することができる。この関連で、図5が、スカラ演算をマージするためのさらなる例示的な演算を示すために提供される。図5を説明する際に、明確にするために図2および図3の要素が参照される。図5に示される演算は、図4のブロック88および90の演算後に行うことのできる理解されたい。

【0042】

図5において、ベクトルプロセッサ12は、複数の書込みポート22のうちの書込みポート22(0)～22(Z)ごとのレジスタインデックス54およびアキュムレータインデックス58の1つまたは複数に基づいて、複数の書込みポート22の対応する2つ以上の書込みポート22(0)～22(Z)のための2つ以上のスカラ演算をマージしてもよいと判断する(ブロック94)。いくつかの態様では、ベクトルプロセッサ12は、第1の書込みポート22(0)～22(Z)のためのレジスタインデックス54が、複数の書込みポート22の1つまたは複数の他の書込みポート22(0)～22(Z)のためのレジスタインデックス54と同一であると判断することによって、2つ以上のスカラ演算をマージしてもよいと判断することができる(ブロック96)。いくつかの態様は、ベクトルプロセッサ12が、第1の書込みポート22(0)～22(Z)のためのアキュムレータインデックス58が複数の書込みポート22の1つまたは複数の他の書込みポート22(0)～22(Z)のためのアキュムレータインデックス58と同一であるとさらに判断することによって、2つ以上のスカラ演算をマージしてもよいと判断することができる(ブロック98)ことを規定する。

20

【0043】

30

ベクトルプロセッサ12は、次に、2つ以上のスカラ演算を1つのマージされたスカラ演算に組み合わせる(ブロック100)。非限定的な例として、同じアキュムレータ46を1だけインクリメントする2つのスカラ演算をマージして、アキュムレータ46を2だけインクリメントするマージされたスカラ演算にすることができる。ベクトルプロセッサ12は、その後、マージされたスカラ演算を実行することによってスカラ演算を実行する(ブロック102)。

【0044】

40

本明細書において開示される態様による、ベクトルレジスタファイルにおいてデータインデックス方式アキュムレータを使用してベクトルプロセッサによるスカラ演算を並列化することは、任意のプロセッサベースデバイスにおいて提供することができるか、または任意のプロセッサベースデバイスに組み込むことができる。例としては、限定はしないが、セットトップボックス、エンターテインメントユニット、ナビゲーションデバイス、通信デバイス、固定位置データユニット、モバイルロケーションデータユニット、モバイルフォン、セルラーフォン、コンピュータ、ポータブルコンピュータ、デスクトップコンピュータ、携帯情報端末(PDA)、モニタ、コンピュータモニタ、テレビジョン、チューナ、ラジオ、衛星ラジオ、音楽プレーヤ、デジタル音楽プレーヤ、ポータブル音楽プレーヤ、デジタルビデオプレーヤ、ビデオプレーヤ、デジタルビデオディスク(DVD)プレーヤ、およびポータブルデジタルビデオプレーヤがある。

【0045】

この関連で、図6は、図1～図3に示されるベクトルプロセッサ(VP)12を利用することが

50

できる、プロセッサベースシステム104の一例を示す。この例では、プロセッサベースシステム104は、それぞれが1つまたは複数のプロセッサ108を含む、1つまたは複数の中央処理ユニット(CPU)106を含む。1つまたは複数のプロセッサ108は、図1～図3のベクトルプロセッサ(VP)12を含むことができる。CPU106はマスタデバイスとすることができます。CPU106は、一時的に記憶されたデータへの高速アクセスのためにプロセッサ108に結合されるキャッシュメモリ110を有する場合がある。CPU106は、システムバス112に結合され、プロセッサベースシステム104内に含まれるマスタデバイスとスレーブデバイスとを相互結合することができる。よく知られているように、CPU106は、システムバス112を介してアドレス情報、制御情報、およびデータ情報を交換することによって、これらの他のデバイスと通信する。たとえば、CPU106は、スレーブデバイスの一例として、メモリコントローラ114にバストランザクション要求を通信することができる。10

【0046】

他のマスタデバイスおよびスレーブデバイスをシステムバス112に接続することができる。図6に示されるように、これらのデバイスは、例として、メモリシステム116、1つまたは複数の入力デバイス118、1つまたは複数の出力デバイス120、1つまたは複数のネットワークインターフェースデバイス122、および1つまたは複数のディスプレイコントローラ124を含むことができる。入力デバイス118は、限定はしないが、入力キー、スイッチ、音声プロセッサなどを含む、任意のタイプの入力デバイスを含むことができる。出力デバイス120は、限定はしないが、オーディオインジケータ、ビデオインジケータ、他の視覚インジケータなどを含む、任意のタイプの出力デバイスを含むことができる。ネットワークインターフェースデバイス122は、ネットワーク126との間のデータ交換を可能にするよう構成される任意のデバイスとすることができます。ネットワーク126は、限定はしないが、有線ネットワークまたはワイヤレスネットワーク、プライベートネットワークまたは公衆ネットワーク、ローカルエリアネットワーク(LAN)、ワイドローカルエリアネットワーク(WLAN)、およびインターネットを含む、任意のタイプのネットワークとすることができます。ネットワークインターフェースデバイス122は、所望の任意のタイプの通信プロトコルをサポートするように構成することができる。メモリシステム116は、1つまたは複数のメモリユニット128(0～N)を含むことができる。20

【0047】

CPU106はまた、1つまたは複数のディスプレイ130に送信される情報を制御するために、システムバス112を介してディスプレイコントローラ124にアクセスするように構成することができる。ディスプレイコントローラ124は、1つまたは複数のビデオプロセッサ132を介して、表示されるべき情報をディスプレイ130に送信し、ビデオプロセッサ132は、表示されるべき情報を、ディスプレイ130に適したフォーマットになるように処理する。ディスプレイ130は、限定はしないが、陰極線管(CRT)、液晶ディスプレイ(LCD)、プラズマディスプレイなどを含む任意のタイプのディスプレイを含むことができる。30

【0048】

本明細書において開示される態様に関して説明される種々の例示的な論理ブロック、モジュール、回路、およびアルゴリズムは、電子ハードウェア、またはメモリもしくは別のコンピュータ可読媒体に記憶され、プロセッサもしくは他の処理デバイスによって実行される命令、またはその両方の組合せとして実現できることを、当業者はさらに理解されよう。本明細書において説明されるマスタデバイスおよびスレーブデバイスは、例として、任意の回路、ハードウェア構成要素、集積回路(IC)、またはICチップにおいて利用することができる。本明細書において開示されるメモリは、任意のタイプおよびサイズのメモリとすることができます、所望の任意のタイプの情報を記憶するように構成することができる。この互換性を明確に説明するために、種々の例示的な構成要素、ブロック、モジュール、回路、およびステップは、概してそれらの機能に関してこれまで説明してきた。そのような機能がどのように実現されるかは、特定の適用例、設計上の選択、および/またはシステム全体に課された設計制約によって決まる。当業者は、説明された機能を特定の適用例ごとに様々な方法で実現することができるが、そのような実装形態の決定は、本開示の4050

範囲からの逸脱を引き起こすものと解釈されるべきではない。

【0049】

本明細書において開示される態様に関して説明される種々の例示的な論理ブロック、モジュール、および回路は、プロセッサ、デジタルシグナルプロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)もしくは他のプログラマブル論理デバイス、個別ゲートもしくはトランジスタロジック、個別ハードウェア構成要素、または本明細書において説明される機能を実行するように設計されるそれらの任意の組合せを用いて実現または実行することができる。プロセッサは、マイクロプロセッサである場合があるが、代替形態では、プロセッサは、任意の従来のプロセッサ、コントローラ、マイクロコントローラ、またはストートマシンである場合がある。プロセッサはまた、コンピューティングデバイスの組合せ、たとえば、DSPとマイクロプロセッサの組合せ、複数のマイクロプロセッサ、DSPコアと連携した1つもしくは複数のマイクロプロセッサ、または任意の他のそのような構成として実現される場合もある。

10

【0050】

本明細書において開示される態様は、ハードウェアにおいて、また、ハードウェアに記憶された命令において具現される場合があり、命令は、たとえば、ランダムアクセスメモリ(RAM)、フラッシュメモリ、読み取り専用メモリ(ROM)、電気的プログラマブルROM(EPROM)、電気的消去可能プログラマブルROM(EEPROM)、レジスタ、ハードディスク、リムーバブルディスク、CD-ROM、または当技術分野において知られている任意の他の形態のコンピュータ可読媒体内に存在する場合がある。例示的な記憶媒体は、プロセッサが記憶媒体から情報を読み取り、記憶媒体に情報を書き込むことができるよう、プロセッサに結合される。代替形態では、記憶媒体は、プロセッサと一緒にできる。プロセッサおよび記憶媒体は、ASIC内に存在する場合がある。ASICは、リモート局内に存在する場合がある。代替形態では、プロセッサおよび記憶媒体は、個別構成要素として、リモート局、基地局、またはサーバ内に存在する場合がある。

20

【0051】

本明細書の例示的な態様のいずれかにおいて説明される動作ステップは、例および考察を提供するために説明されることにも留意されたい。説明される動作は、図示されるシーケンス以外の多数の異なるシーケンスにおいて実行される場合がある。さらに、単一の動作ステップにおいて説明される動作は、実際にはいくつかの異なるステップにおいて実行される場合がある。さらに、例示的な態様において論じられる1つまたは複数の動作ステップが組み合わせられる場合がある。当業者には容易に明らかになるように、流れ図に示される動作ステップは、多数の異なる変更を受ける場合があることを理解されたい。当業者は、情報および信号が様々な異なる技術および技法のいずれかを使用して表される場合があることも理解されよう。たとえば、上記の説明全体を通して参照される場合があるデータ、命令、コマンド、情報、信号、ビット、シンボル、およびチップは、電圧、電流、電磁波、磁場もしくは磁性粒子、光場もしくは光学粒子、またはそれらの任意の組合せによって表される場合がある。

30

【0052】

本開示の上述の説明は、当業者が本開示を実施するかまたは使用することを可能にするために提供される。本開示に対する種々の変更が、当業者には容易に明らかになり、本明細書において規定される一般原理は、本開示の趣旨または範囲を逸脱することなく、他の変形形態に適用することができる。したがって、本開示は、本明細書において説明される例および設計に限定されるものではなく、本明細書において開示される原理および新規の特徴と一致する最も広い範囲を与えられるべきである。

40

【符号の説明】

【0053】

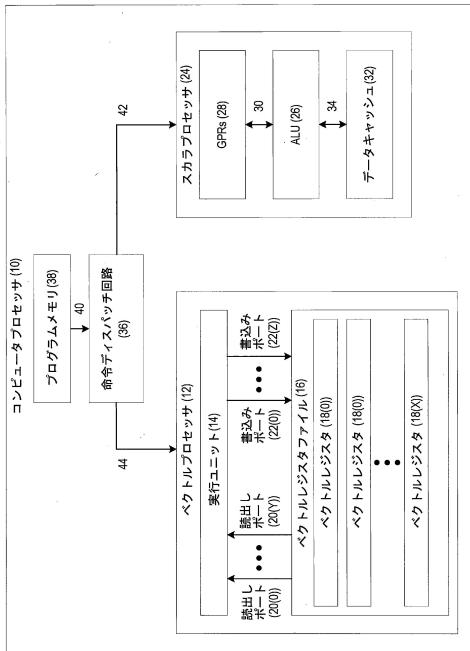
- 10 コンピュータプロセッサ
- 12 ベクトルプロセッサ
- 14 実行ユニット

50

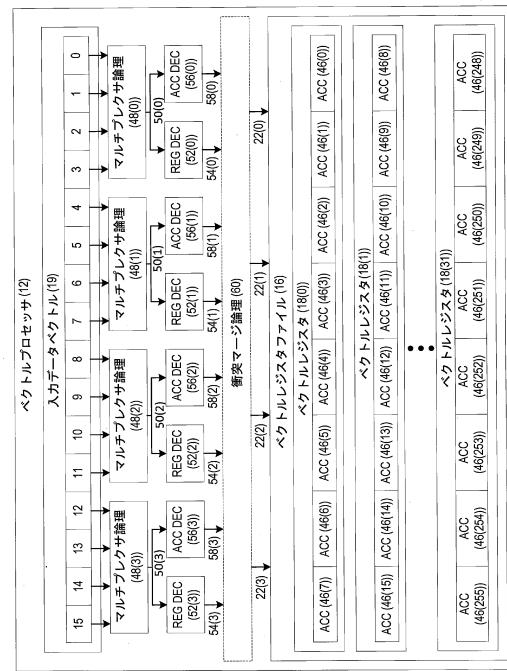
16 ベクトルレジスタファイル	
18 ベクトルレジスタ	
18(0) ベクトルレジスタ	
18(1) ベクトルレジスタ	
18(31) ベクトルレジスタ	
18(X) ベクトルレジスタ	
19 入力データベクトル	
20(0) 読出しポート	10
20(Y) 読出しポート	
20 書込みポート	
22(0) 書込みポート	
22(3) 書込みポート	
22(Z) 書込みポート	
24 スカラプロセッサ	
26 算術論理演算ユニット(ALU)	
28 汎用レジスタ(GPR)	
30 矢印	
32 データキャッシュ	
34 矢印	
36 命令ディスパッチ回路	20
38 プログラムメモリ	
40 矢印	
42 スカラデータバス	
44 ベクトルデータバス	
46 アキュムレータ	
46(0) アキュムレータ	
46(7) アキュムレータ	
46(8) アキュムレータ	
46(15) アキュムレータ	
46(248) アキュムレータ	30
46(250) アキュムレータ	
46(255) アキュムレータ	
48 マルチプレクサ論理ブロック	
48(1) マルチプレクサ論理ブロック	
48(3) マルチプレクサ論理ブロック	
50 入力データ値	
50(0) 入力データ値	
50(3) 入力データ値	
52 レジスタデコーダ	
52(0) レジスタデコーダ	40
52(3) レジスタデコーダ	
54 レジスタインデックス	
54(0) レジスタインデックス	
54(3) レジスタインデックス	
56 アキュムレータデコーダ	
56(0) アキュムレータデコーダ	
56(3) アキュムレータデコーダ	
58 アキュムレータインデックス	
58(0) アキュムレータインデックス	
58(3) アキュムレータインデックス	50

60	衝突マージ論理ブロック	
62	入力データ値	
64	入力データベクトル	
66(0)	8ビット値	
66(3)	8ビット値	
68	マルチプレクサ論理ブロック	
70	レジスタデコーダ	
72	アキュムレータデコーダ	
74	レジスタインデックス	
76	アキュムレータインデックス	10
104	プロセッサベースシステム	
106	中央処理ユニット(CPU)	
108	プロセッサ	
110	キャッシュメモリ	
112	システムバス	
114	メモリコントローラ	
116	メモリシステム	
118	入力デバイス	
120	出力デバイス	
122	ネットワークインターフェースデバイス	20
124	ディスプレイコントローラ	
126	ネットワーク	
128(0)	メモリユニット	
128(N)	メモリユニット	
130	ディスプレイ	
132	ビデオプロセッサ	

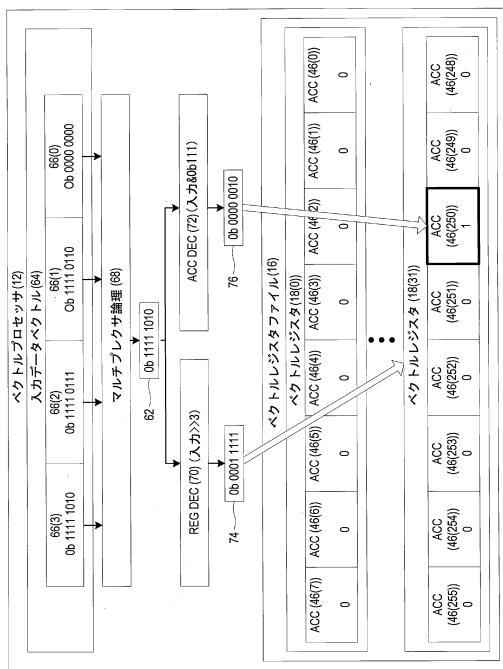
【図1】



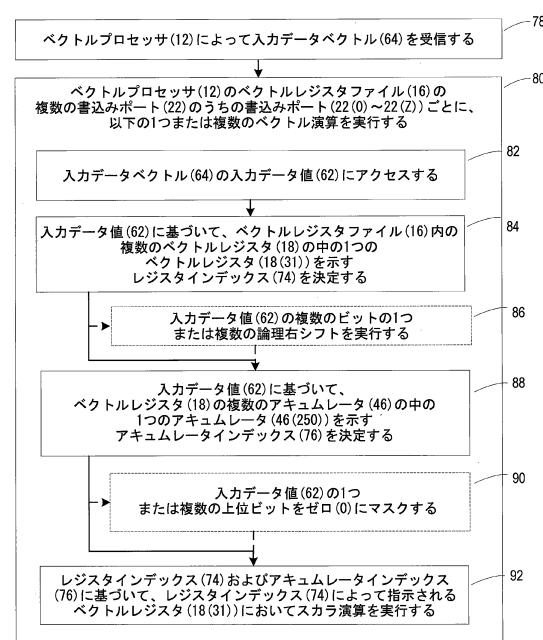
【図2】



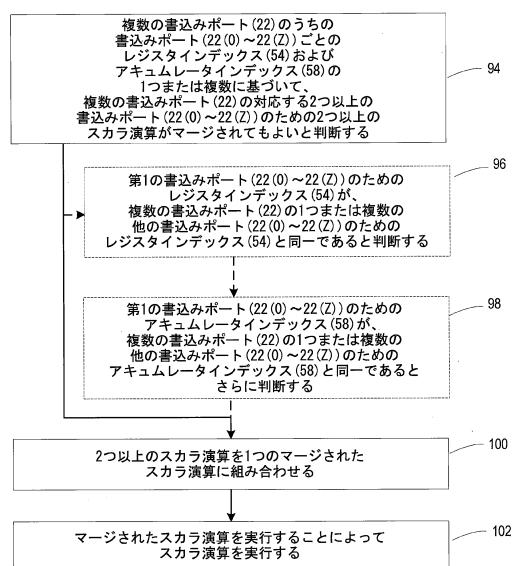
【 四 3 】



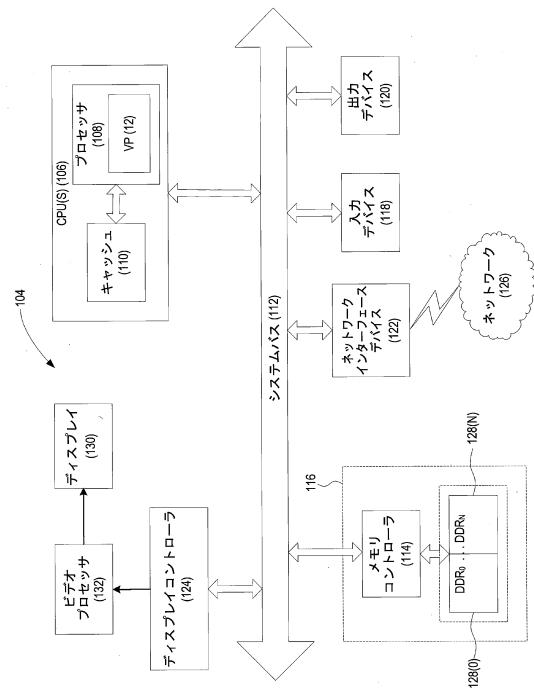
【図4】



【図5】



【図6】



フロントページの続き

(51)Int.CI.	F I
G 0 6 F 9/345 (2006.01)	G 0 6 F 9/38 3 7 0 A
G 0 6 F 9/355 (2006.01)	G 0 6 F 17/16 S
	G 0 6 F 9/345 A
	G 0 6 F 9/355 3 2 0

(72)発明者 エリック・ウェイン・マハーン
アメリカ合衆国・カリフォルニア・92121・サン・ディエゴ・モアハウス・ドライブ・577
5

審査官 井上 宏一

(56)参考文献 米国特許出願公開第2009/0276606(US, A1)
特開平 2-50258 (JP, A)

(58)調査した分野(Int.CI., DB名)

G 0 6 F	1 7 / 1 6
G 0 6 F	9 / 3 0 - 9 / 3 5 5
G 0 6 F	9 / 3 8