

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6920277号

(P6920277)

(45) 発行日 令和3年8月18日 (2021.8.18)

(24) 登録日 令和3年7月28日 (2021.7.28)

(51) Int. Cl.	F I
<b>G06F 9/38 (2006.01)</b>	G06F 9/38 370A
<b>G06F 9/34 (2006.01)</b>	G06F 9/34 320C
<b>G06F 9/302 (2006.01)</b>	G06F 9/34 330
<b>G06F 9/315 (2006.01)</b>	G06F 9/302 A
	G06F 9/315 S

請求項の数 8 (全 20 頁)

(21) 出願番号	特願2018-502231 (P2018-502231)	(73) 特許権者	507364838
(86) (22) 出願日	平成28年6月21日 (2016.6.21)		クアルコム、インコーポレイテッド
(65) 公表番号	特表2018-525731 (P2018-525731A)		アメリカ合衆国 カリフォルニア 921
(43) 公表日	平成30年9月6日 (2018.9.6)		21 サン ディエゴ モアハウス ドラ
(86) 国際出願番号	PCT/US2016/038487		イブ 5775
(87) 国際公開番号	W02017/014892	(74) 代理人	100108453
(87) 国際公開日	平成29年1月26日 (2017.1.26)		弁理士 村山 靖彦
審査請求日	令和1年6月5日 (2019.6.5)	(74) 代理人	100163522
(31) 優先権主張番号	14/805,456		弁理士 黒田 晋平
(32) 優先日	平成27年7月21日 (2015.7.21)	(72) 発明者	エリック・ウェイン・マハーアン
(33) 優先権主張国・地域又は機関	米国 (US)		アメリカ合衆国・カリフォルニア・921
早期審査対象出願			21-1714・サン・ディエゴ・モアハ
前置審査			ウス・ドライブ・5775・クアルコム・
			インコーポレイテッド
			最終頁に続く

(54) 【発明の名称】 広いデータ要素のためのレジスタのペアを用いた偶数要素演算および奇数要素演算を有する混合幅 SIMD 演算

(57) 【特許請求の範囲】

【請求項 1】

混合幅単一命令複数データ (SIMD) 演算を実行する方法であって、  
プロセッサによって、

第1のソースレジスタを備える第1のソースベクトルオペランドであって、前記第1のソースレジスタが第1のビット幅の第1のセットのソースデータ要素を備える、第1のソースベクトルオペランドと、

第2のビット幅の宛先データ要素を備える宛先ベクトルオペランドと  
を備える SIMD 命令を受信するステップであって、

前記第2のビット幅が前記第1のビット幅の2倍であり、

前記宛先ベクトルオペランドが、前記宛先データ要素の第1のサブセットを備える第1の宛先レジスタと、前記宛先データ要素の第2のサブセットを備える第2の宛先レジスタとを含む宛先レジスタのペアを備え、

前記第1のソースレジスタが、宛先レジスタの前記ペアに対応する単一のレジスタである、ステップと、

前記第1のセットのソースデータ要素の順序に基づいて、前記プロセッサにおいて前記 SIMD 命令を実行するステップであって、前記第1のセットのソースデータ要素の前記順序は、前記第1のセットのソースデータ要素と前記宛先データ要素との間のマッピングを提供するために割り当てられ、

前記第1のセットの偶数番号のソースデータ要素から、前記第1の宛先レジスタ内の前

記宛先データ要素の前記第1のサブセットを生成するステップと、

前記第1のセットの奇数番号のソースデータ要素から、前記第2の宛先レジスタ内の前記宛先データ要素の前記第2のサブセットを生成するステップと

を備える、ステップと

を備え、

前記第1のセットのソースデータ要素がそれぞれのSIMDレーン内にあり、

前記マッピングに従って、前記ソースデータ要素の各々から、前記それぞれのSIMDレーン、または前記それぞれのSIMDレーンに隣接するSIMDレーン内のそれぞれの宛先データ要素を生成する、方法。

【請求項2】

前記SIMD命令が、前記第1のセットの前記ソースデータ要素の二乗関数、左シフト関数、インクリメント、または一定値による加算のうちの1つである、請求項1に記載の方法。

【請求項3】

混合幅単一命令複数データ(SIMD)演算を実行する方法であって、

プロセッサによって、

第1のビット幅のソースデータ要素を備えるソースベクトルオペランドと、

宛先レジスタを備える宛先ベクトルオペランドであって、前記宛先レジスタが第2のビット幅の宛先データ要素を備える、宛先ベクトルオペランドと

を備えるSIMD命令を受信するステップであって、

前記第2のビット幅が前記第1のビット幅の半分であり、

前記ソースベクトルオペランドが、前記ソースデータ要素の第1のサブセットを備える第1のソースレジスタと、前記ソースデータ要素の第2のサブセットを備える第2のソースレジスタとを含むソースレジスタのペアを備え、

前記宛先レジスタが、ソースレジスタの前記ペアに対応する単一のレジスタである、ステップと、

前記宛先データ要素の順序に基づいて、前記プロセッサにおいて前記SIMD命令を実行するステップであって、前記宛先データ要素の前記順序は、前記ソースデータ要素と前記宛先データ要素との間のマッピングを提供するために割り当てられ、

前記ソースデータ要素の前記第1のサブセットから偶数番号の宛先データ要素を生成するステップと、

前記ソースデータ要素の前記第2のサブセットから奇数番号の宛先データ要素を生成するステップと

を備える、ステップと

を備え、

前記宛先データ要素がそれぞれのSIMDレーン内にあり、

前記マッピングに従って、前記それぞれのSIMDレーン、または前記それぞれのSIMDレーンに隣接するSIMDレーン内のソースデータ要素から、前記宛先データ要素の各々を生成する、方法。

【請求項4】

前記SIMD命令が、前記ソースデータ要素の右シフト関数である、請求項3に記載の方法

。

【請求項5】

プロセッサによって実行されると、前記プロセッサに混合幅単一命令複数データ(SIMD)演算を実行させる、前記プロセッサによって実行可能な命令を備える非一時的コンピュータ可読記憶媒体であって、

第1のソースレジスタを備える第1のソースベクトルオペランドであって、前記第1のソースレジスタが第1のビット幅の第1のセットのソースデータ要素を備える、第1のソースベクトルオペランドと、

第2のビット幅の宛先データ要素を備える宛先ベクトルオペランドと  
を備えるSIMD命令であって、

前記第2のビット幅が前記第1のビット幅の2倍であり、

前記宛先ベクトルオペランドが、前記宛先データ要素の第1のサブセットを備える第1の宛先レジスタと、前記宛先データ要素の第2のサブセットを備える第2の宛先レジスタとを含む宛先レジスタのペアを備え、

前記第1のソースレジスタが、宛先レジスタの前記ペアに対応する単一のレジスタである、SIMD命令と、

前記第1のセットのソースデータ要素の順序に基づいて、

前記第1のセットの偶数番号のソースデータ要素から、前記第1の宛先レジスタ内の前記宛先データ要素の前記第1のサブセットを生成するためのコードと、

前記第1のセットの奇数番号のソースデータ要素から、前記第2の宛先レジスタ内の前記宛先データ要素の前記第2のサブセットを生成するためのコードと

を備え、前記第1のセットのソースデータ要素の前記順序は、前記第1のセットのソースデータ要素と前記宛先データ要素との間のマッピングを提供するために割り当てられ、

前記第1のセットのソースデータ要素がそれぞれのSIMDレーン内にあり、

前記マッピングに従って、前記ソースデータ要素の各々から、前記それぞれのSIMDレーン、または前記それぞれのSIMDレーンに隣接するSIMDレーン内のそれぞれの宛先データ要素を生成するためのコードを備える、非一時的コンピュータ可読記憶媒体。

【請求項6】

前記SIMD命令が、前記第1のセットの前記ソースデータ要素の二乗関数、左シフト関数、インクリメント、または一定値による加算のうちの1つである、請求項5に記載の非一時的コンピュータ可読記憶媒体。

【請求項7】

プロセッサによって実行されると、前記プロセッサに混合幅単一命令複数データ(SIMD)演算を実行させる、前記プロセッサによって実行可能な命令を備える非一時的コンピュータ可読記憶媒体であって、

第1のビット幅のソースデータ要素を備えるソースベクトルオペランドと、

宛先レジスタを備える宛先ベクトルオペランドであって、前記宛先レジスタが第2のビット幅の宛先データ要素を備える、宛先ベクトルオペランドとを備えるSIMD命令であって、

前記第2のビット幅が前記第1のビット幅の半分であり、

前記ソースベクトルオペランドが、前記ソースデータ要素の第1のサブセットを備える第1のソースレジスタと、前記ソースデータ要素の第2のサブセットを備える第2のソースレジスタとを含むソースレジスタのペアを備え、

前記宛先レジスタが、ソースレジスタの前記ペアに対応する単一のレジスタである、SIMD命令と、

前記宛先データ要素の順序に基づいて、

前記ソースデータ要素の前記第1のサブセットから偶数番号の宛先データ要素を生成するためのコードと、

前記ソースデータ要素の前記第2のサブセットから奇数番号の宛先データ要素を生成するためのコードと

を備え、前記宛先データ要素の前記順序は、前記ソースデータ要素と前記宛先データ要素との間のマッピングを提供するために割り当てられ、

前記宛先データ要素がそれぞれのSIMDレーン内にあり、

前記マッピングに従って、前記それぞれのSIMDレーン、または前記それぞれのSIMDレーンに隣接するSIMDレーン内のソースデータ要素から、前記宛先データ要素の各々を生成するためのコードを備える、非一時的コンピュータ可読記憶媒体。

【請求項8】

前記SIMD命令が、前記ソースデータ要素の右シフト関数である、請求項7に記載の非一時的コンピュータ可読記憶媒体。

【発明の詳細な説明】

## 【技術分野】

## 【0001】

本開示の態様は、少なくとも1つのベクトルのデータ要素が少なくとも1つの他のベクトルのデータ要素とは異なるビット幅である2つ以上のベクトルを伴う演算に関する。そのような演算は、混合幅演算(mixed-width operation)と呼ばれる。より詳細には、いくつかの態様は、少なくとも1つの第1のベクトルオペランドと第2のベクトルオペランドとを伴う混合幅単一命令複数データ(SIMD)(mixed-width single instruction multiple data)演算に関し、第1のベクトルオペランドまたは第2のベクトルオペランドのうちの少なくとも1つは、偶数または奇数のレジスタのペアに記憶され得るデータ要素を有する。

## 【背景技術】

10

## 【0002】

データ並列処理を利用する処理システムにおいては、単一命令複数データ(SIMD)命令が使用され得る。たとえば、データベクトルの2つ以上のデータ要素に対して同じまたは共通のタスクが実行される必要がある場合、データ並列処理が存在する。複数の命令を使用するのではなく、対応する複数のSIMDレーン内の複数のデータ要素に対して実行されるべき同じ命令を定義する単一のSIMD命令を使用することによって、2つ以上のデータ要素に対して並列に共通のタスクが実行され得る。

## 【0003】

SIMD命令は、ソースベクトルオペランドおよび宛先ベクトルオペランドなどの1つまたは複数のベクトルオペランドを含み得る。各ベクトルオペランドは2つ以上のデータ要素を含む。SIMD命令の場合、同じベクトルオペランドに属するすべてのデータ要素は、一般に、同じビット幅であり得る。しかしながら、いくつかのSIMD命令は混合幅オペランドを指定してよく、第1のベクトルオペランドのデータ要素は第1のビット幅のデータ要素であり得、第2のベクトルオペランドのデータ要素は第2のビット幅のデータ要素であり得、第1のビット幅と第2のビット幅は互いに異なる。混合幅オペランドを含むSIMD命令の実行には、いくつかの課題があり得る。

20

## 【0004】

図1A~図1Cは、混合幅オペランドを有するSIMD命令を実行するための従来の実装形態に伴う課題の例を示す。図1Aを参照すると、SIMD命令100を実行するための第1の従来の実装形態が示されている。SIMD命令100は、64ビット命令セットアーキテクチャ(ISA)をサポートする従来のプロセッサ(図示せず)によって実行され得ると仮定する。これは、SIMD命令100などの命令が、最大64ビットのビット幅を有するオペランドを指定し得ることを意味する。64ビットオペランドは、64ビットレジスタまたは32ビットレジスタのペアに関して指定され得る。

30

## 【0005】

SIMD命令100の目的は、ソースオペランド102の各データ要素に対して同じ命令を実行することである。ソースオペランド102は、0~7でラベル付けされた8個の8ビットデータ要素を備える64ビットベクトルである。ソースオペランド102は、単一の64ビットレジスタ、または32ビットレジスタのペアに記憶され得る。8個のデータ要素0~7の各々に対して実行される同じ命令または共通の演算は、たとえば、乗算、二乗関数、左シフト関数、インクリメント関数、加算(たとえば、命令内の一定値もしくは即値フィールドとの、または別のベクトルオペランドによって提供される値との)などであり得、その結果は、結果として得られる8個のデータ要素ごとに、8ビット以上、および最大16ビットのストレージを消費する可能性がある。これは、SIMD命令100の結果が、ソースオペランド102が消費し得るストレージ空間の2倍、すなわち2つの64ビットレジスタ、または32ビットレジスタの2つのペアを消費する可能性があることを意味する。

40

## 【0006】

SIMD命令100を実装するように構成された従来のプロセッサは、64ビットより大きいビット幅のオペランドを指定する命令を含まないので、SIMD命令100は、SIMD命令100Xと100Yとの2つの構成要素に分割され得る。SIMD命令100Xは、ソースオペランド102の偶数0、2

50

、4、および6でラベル付けされたデータ要素(または「偶数番号のデータ要素」)に対して実行されるべき共通の演算を指定する。SIMD命令100Xは、64ビット幅であり、それぞれが上位(H)8ビットおよび下位(L)8ビットで構成されるA、C、E、およびGでラベル付けされた16ビットデータ要素を含む、宛先オペランド104xを指定する。ソースオペランド102の偶数番号の8ビットデータ要素0、2、4、および6に対する共通の演算の結果は、宛先オペランド104xの16ビットデータ要素A、C、E、およびGに対応して書き込まれる。SIMD命令100YはSIMD命令100Xに類似しているが、SIMD命令100Yは、ソースオペランド102の奇数1、3、5、および7でラベル付けされたデータ要素(または「奇数番号のデータ要素」)に対する共通の演算を指定し、その結果は、やはりSIMD命令100Xの宛先オペランド104xと同様に64ビットオペランドである、宛先オペランド104yの16ビットデータ要素B、D、F、Hに書き込まれるという相違点がある。このようにして、SIMD命令100Xおよび100Yの各々は、1つの64ビット宛先オペランドを指定することができ、また、SIMD命令100Xおよび100Yは、ソースオペランド102のデータ要素0~7の各々に対する共通の演算の実行をともに達成することができる。しかし、SIMD命令100を実装するために必要な2つの別個の命令のために、コード空間が増大する。

10

【0007】

図1Bは、構成要素SIMD命令120Xおよび120Yの異なるセットを使用するSIMD命令100の第2の従来の実装形態を示す。SIMD命令120Xおよび120Yは、それぞれソースオペランド102の8ビットデータ要素0~7の各々に対する共通の演算を指定する。SIMD命令120Xは、結果の下位(L)8ビットが書き込まれる宛先オペランド124xを、宛先オペランド124xの対応する8ビット結果データ要素A~Hに指定する(一方、結果の上位(H)8ビットは破棄される)。同様に、命令120Yは、結果の上位(H)8ビットが書き込まれる宛先オペランド124yを宛先オペランド124yの対応する8ビットデータ要素A~Hに指定する(一方、結果の下位(L)8ビットは破棄される)。このSIMD命令100の第2の従来の実装形態では、2つの構成要素SIMD命令120Xおよび120Yのコード空間の増加も問題となる。さらに、理解され得るように、第2の従来の実装形態はまた、ソースオペランド102のデータ要素0~7の各々の上位(H)8ビット(たとえば、実行命令120X中)または下位(L)8ビット(たとえば、実行命令120Y中)のいずれかを計算および廃棄する際に電力の浪費を招く。

20

【0008】

図1Cは、図1AのSIMD命令100Xおよび100Yといくつかの点で類似している、SIMD命令140Xおよび140Yの構成要素のさらに別のセットを使用するSIMD命令100の第3の従来の実装形態を示す。相違点は、ソースオペランド102のデータ要素のうちの諸データ要素が各SIMD命令による操作を受けるという点である。より詳細には、偶数番号の8ビットデータ要素ではなく、SIMD命令140Xは、ソースオペランド102の下位4個のデータ要素0~3に対して実行されるべき共通の演算を指定する。結果は宛先オペランド144xの16ビットデータ要素A、B、C、Dに書き込まれる。しかしながら、SIMD命令140Xの実行は、宛先オペランド140Xの全64ビットにわたる下位4個の8ビットデータ要素(32ビットにまたがる)に対する演算の結果を広げること(spreading out)を含む。SIMD命令144yは同様であり、64ビット宛先オペランド144yの16ビットデータ要素E、F、G、Hにわたるソースオペランド102の上位4個の8ビットデータ要素4~7に対する演算の結果を広げることが指定する。第1および第2の従来の実装形態のようなコードサイズの増加は別として、第3の従来の実装形態において見られるようなこれらのデータ移動を広げることが、クロスバーのような追加のハードウェアを必要とする可能性がある。

30

40

【発明の概要】

【発明が解決しようとする課題】

【0009】

したがって、従来の実装形態の前述の欠点を回避する、混合幅SIMD命令の改良された実装形態が必要とされている。

【課題を解決するための手段】

【0010】

50

例示的な態様は、第1のビット幅のデータ要素を備える少なくとも1つのソースベクトルオペランドと、第2のビット幅のデータ要素を備える宛先ベクトルオペランドとを有する、混合幅単一命令複数データ(SIMD)命令に関連するシステムおよび方法を含み、第2のビット幅は、第1のビット幅の半分または2倍のいずれかである。これに対応して、ソースベクトルオペランドまたは宛先ベクトルオペランドのうちの1つは、第1のレジスタと第2のレジスタとの、レジスタのペアとして表される。他のベクトルオペランドは単一のレジスタとして表される。第1のレジスタのデータ要素は、単一のレジスタとして表される他のベクトルオペランドの偶数番号のデータ要素に対応し、第2のレジスタのデータ要素は、単一のレジスタとして表される他のベクトルオペランドのデータ要素に対応する。

【0011】

10

たとえば、例示的な態様は、混合幅単一命令複数データ(SIMD)演算を実行する方法に関連し、本方法は、プロセッサによって、第1のビット幅の第1のセットのソースデータ要素を備える少なくとも1つの第1のソースベクトルオペランドと、第2のビット幅の宛先データ要素を備える少なくとも1つの宛先ベクトルオペランドとを備えるSIMD命令を受信するステップであって、第2のビット幅は第1のビット幅の2倍である、ステップを備える。宛先ベクトルオペランドは、宛先データ要素の第1のサブセットを備える第1のレジスタと、宛先データ要素の第2のサブセットを備える第2のレジスタとを含むレジスタのペアを備える。第1のセットのソースデータ要素の順序に基づいて、本方法は、プロセッサにおいてSIMD命令を実行するステップであって、第1のセットの偶数番号のソースデータ要素から、第1のレジスタ内の宛先データ要素の第1のサブセットを生成するステップと、第1のセッ

20

【0012】

別の例示的な態様は、混合幅単一命令複数データ(SIMD)演算を実行する方法に関連し、本方法は、プロセッサによって、第1のビット幅のソースデータ要素を備える少なくとも1つのソースベクトルオペランドと、第2のビット幅の宛先データ要素を備える少なくとも1つの宛先ベクトルオペランドとを備えるSIMD命令を受信するステップを備え、第2のビット幅は第1のビット幅の半分である。ソースベクトルオペランドは、ソースデータ要素の第1のサブセットを備える第1のレジスタと、ソースデータ要素の第2のサブセットを備える第2のレジスタとを含むレジスタのペアを備える。宛先データ要素の順序に基づいて、本方法は、プロセッサにおいてSIMD命令を実行するステップであって、第1のレジスタ内のソースデータ要素の対応する第1のサブセットから偶数番号の宛先データ要素を生成するステップと、第2のレジスタ内のソースデータ要素の対応する第2のサブセットから奇数番号の宛先データ要素を生成するステップとを備える、ステップを含む。

30

【0013】

別の例示的な態様は、プロセッサによって実行されると、プロセッサに混合幅単一命令複数データ(SIMD)演算を実行させる、プロセッサによって実行可能な命令を備える非一時的コンピュータ可読記憶媒体に関連する。非一時的コンピュータ可読記憶媒体はSIMD命令を備え、SIMD命令は、第1のビット幅の第1のセットのソースデータ要素を備える少なくとも1つの第1のソースベクトルオペランドと、第2のビット幅の宛先データ要素を備える少なくとも1つの宛先ベクトルオペランドとを備え、第2のビット幅は第1のビット幅の2倍である。宛先ベクトルオペランドは、宛先データ要素の第1のサブセットを備える第1のレジスタと、宛先データ要素の第2のサブセットを備える第2のレジスタとを含むレジスタのペアを備える。第1のセットのソースデータ要素の順序に基づいて、非一時的コンピュータ可読記憶媒体は、第1のセットの偶数番号のソースデータ要素から、第1のレジスタ内の宛先データ要素の第1のサブセットを生成するためのコードと、第1のセットの奇数番号のソースデータ要素から、第2のレジスタ内の宛先データ要素の第2のサブセットを生成するためのコードとを含む。

40

【0014】

別の例示的な態様は、プロセッサによって実行されると、プロセッサに混合幅単一命令

50

複数データ(SIMD)演算を実行させる、プロセッサによって実行可能な命令を備える非一時的コンピュータ可読記憶媒体に関連し、非一時的コンピュータ可読記憶媒体はSIMD命令を備える。SIMD命令は、第1のビット幅のソースデータ要素を備える少なくとも1つのソースベクトルオペランドと、第2のビット幅の宛先データ要素を備える少なくとも1つの宛先ベクトルオペランドとを備え、第2のビット幅は第1のビット幅の半分である。ソースベクトルオペランドは、ソースデータ要素の第1のサブセットを備える第1のレジスタと、ソースデータ要素の第2のサブセットを備える第2のレジスタとを含むレジスタのペアを備える。宛先データ要素の順序に基づいて、非一時的コンピュータ可読記憶媒体は、第1のレジスタ内のソースデータ要素の対応する第1のサブセットから偶数番号の宛先データ要素を生成するためのコードと、第2のレジスタ内のソースデータ要素の対応する第2のサブセットから奇数番号の宛先データ要素を生成するためのコードとを含む。

10

【0015】

添付の図面は、本発明の態様の説明を補助するために提示され、態様の説明のためだけに提供され、本発明の限定ではない。

【図面の簡単な説明】

【0016】

【図1A】混合幅SIMD命令の従来の実装形態を示す図である。

【図1B】混合幅SIMD命令の従来の実装形態を示す図である。

【図1C】混合幅SIMD命令の従来の実装形態を示す図である。

【図2A】本開示の態様による、混合幅SIMD命令の例示的な実装形態を示す図である。

20

【図2B】本開示の態様による、混合幅SIMD命令の例示的な実装形態を示す図である。

【図2C】本開示の態様による、混合幅SIMD命令の例示的な実装形態を示す図である。

【図3A】混合幅単一命令複数データ(SIMD)演算を実行する方法を示す図である。

【図3B】混合幅単一命令複数データ(SIMD)演算を実行する方法を示す図である。

【図4】本開示の態様が有利に使用され得る例示的なワイヤレスデバイス400を示す図である。

【発明を実施するための形態】

【0017】

本発明の態様は、本発明の特定の態様を対象とする、以下の説明および関連する図面に開示されている。本発明の範囲から逸脱することなく、代替態様が考案され得る。さらに、本発明のよく知られている要素は、本発明の関連する詳細を不明瞭にしないように、詳細には記載されないか、省略される。

30

【0018】

「例示的」という用語は、本明細書では、「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。本明細書において「例示的」と記載されている任意の態様は、必ずしも他の態様よりも好ましいまたは有利であると解釈されるべきではない。同様に、「本発明の態様」という用語は、本発明のすべての態様が議論された特徴、利点、または動作モードを含むことを必要としない。

【0019】

本明細書で使用する用語は、特定の態様のみを説明する目的のためのものであり、本発明の態様の限定であることを意図しない。本明細書で使用する単数形「a」、「an」、および「the」は、コンテキストがはっきりと別段に指示しない限り、複数形も含むことが意図される。さらに、「備える(comprises)」、「備えている(comprising)」、「含む(includes)」、および/または「含んでいる(including)」という用語は、本明細書で使用されるとき、述べられた特徴、整数、ステップ、動作、要素、および/または構成要素の存在を明示するが、1つまたは複数の他の特徴、整数、ステップ、動作、要素、構成要素、および/またはそれらのグループの存在または追加を排除しないことが理解されよう。

40

【0020】

さらに、多くの態様は、たとえば、コンピューティングデバイスの要素によって実行されるべきアクションのシーケンスに関して記述される。本明細書で説明する様々なアクシ

50

ョンは、特定の回路(たとえば、特定用途向け集積回路(ASIC))によって、1つもしくは複数のプロセッサによって実行されているプログラム命令によって、または両方の組合せによって実行され得ることが認識されよう。加えて、本明細書で説明するアクションのこれらのシーケンスは、実行時に関連するプロセッサに本明細書で説明する機能を実行させるコンピュータ命令の対応するセットを記憶しているコンピュータ可読記憶媒体の任意の形態内に完全に具体化されていると考えられ得る。したがって、本発明の様々な態様は、そのすべてが特許請求される主題の範囲内にあると考えられているいくつかの異なる形態において具体化されてもよい。さらに、本明細書で説明される態様の各々について、任意のそのような態様の対応する形態は、たとえば、記載されたアクションを実行する「ように構成された論理」として本明細書において記載され得る。

10

#### 【0021】

本開示の例示的な態様は、SIMDレーンを横切るデータ移動を回避し、コードサイズを低減する、混合幅SIMD演算の実装形態に関連する。たとえば、SIMD演算を2つ以上の構成要素SIMD命令に分解(たとえば、図1A～図1CにおけるSIMD命令100の従来の実行)するのではなく、例示的な態様は、1つまたは複数のベクトルオペランドをオペランドのペアとして指定する単一のSIMD命令を含み、これらはレジスタのペアの観点から表され得る。少なくとも1つのベクトルオペランド(ソースオペランドまたは宛先オペランドのいずれか)をレジスタのペアまたはレジスタペアとして指定することによって、2つ以上の構成要素の従来のSIMD命令の代わりに単一の例示的なSIMD命令が使用され得る。したがって、混合幅SIMD演算のコードサイズが低減される。

20

#### 【0022】

本開示においては、命令が1つまたは複数のレジスタに対して実行されるべき演算を指定する通常の命令フォーマットに従うために、レジスタに関してオペランドを表現することが参照される点に留意されたい。したがって、SIMD命令は、レジスタに関して表される1つまたは複数のオペランドに対して共通の演算が指定されるフォーマットのSIMD命令であり得る。したがって、本開示による例示的な混合幅SIMD命令は、単一のレジスタに関して表される少なくとも1つのベクトルオペランドと、レジスタのペアに関して表される少なくとも1つの他のベクトルオペランドとを含む。これらのレジスタへの言及は、例示的なSIMD命令を備えるプログラムによって使用される論理レジスタまたはアーキテクチャレジスタに関し得る。また、無制限に、物理レジスタファイルの物理レジスタにも関し得る。

30

#### 【0023】

したがって、レジスタファイルに結合されたプロセッサにおいて混合幅単一命令複数データ(SIMD)演算を実行する例示的な方法は、第1のビット幅のデータ要素を備える少なくとも1つの第1のベクトルオペランドと、第2のビット幅のデータ要素を備える少なくとも1つの第2のベクトルオペランドとを有するSIMD命令を指定することを含み得る。第1のベクトルオペランドはソースベクトルオペランドであり得、第2のベクトルオペランドは宛先ベクトルオペランドであり得る。それに対応して、ソースベクトルオペランドのデータ要素はソースデータ要素と呼ばれ得、宛先ベクトルオペランドのデータ要素は宛先データ要素と呼ばれ得る。

40

#### 【0024】

例示的な混合幅SIMD命令においては、ソースデータ要素と宛先データ要素との間に1対1対応が存在する。一般に、混合幅SIMD命令において指定された演算がソースデータ要素に対して実行されると、特定の対応する宛先データ要素が生成される。たとえば、宛先ベクトルオペランドを形成するためにソースベクトルオペランドを左シフトするための混合幅SIMD演算を考える。この例では、各ソースデータ要素は、ソースデータ要素の左シフトが実行されるときに、特定の宛先データ要素を生成する。

#### 【0025】

本開示の1つの例示的な態様では、宛先データ要素の第2のビット幅は、ソースデータ要素の第1のビット幅より小さく、具体的には半分のサイズであり得る。この態様では、宛

50



先ベクトルオペランドはレジスタのペアとして表され得、ソースベクトルオペランドは単一のレジスタとして表され得る。

【0026】

本開示の別の例示的な態様では、宛先データ要素の第2のビット幅は、ソースデータ要素の第1のビット幅より大きく、具体的には2倍のサイズであり得る。この態様では、ソースベクトルオペランドは単一のレジスタとして表され得、宛先ベクトルオペランドはレジスタのペアとして表され得る。

【0027】

ソースベクトルオペランドおよび宛先ベクトルオペランドのソースデータ要素とベクトルデータ要素との間の特定のマッピングをそれぞれ示すために、データ要素のビット幅がより小さいベクトルオペランドのデータ要素に順序が割り当てられる。たとえば、単一のレジスタとして表されるベクトルオペランドのデータ要素に順序が割り当てられる。順序に基づいて、偶数番号のデータ要素(たとえば、番号0、2、4、6などに対応する)および奇数番号のデータ要素(たとえば、番号1、3、5、7などに対応する)は、単一のレジスタとして表されるベクトルオペランドに対して識別される。他のベクトルオペランドのレジスタのペアは、第1のレジスタおよび第2のレジスタと呼ばれ、それぞれデータ要素の第1のサブセットおよび第2のサブセットを備える。したがって、単一のレジスタとして表されるベクトルオペランドの偶数番号のデータ要素には第1のサブセットまたは第1のレジスタのデータ要素との対応が割り当てられ、奇数番号のデータ要素には第2のサブセットまたは第2のレジスタのデータ要素との対応が割り当てられる。このようにして、指定されたSIMD演算の実行中に、対応する宛先データ要素を生成するために、ソースデータ要素に対してSIMDレーンを横切る大きなデータ移動が回避される。

【0028】

例示的な態様はまた、たとえば、第3のビット幅の第3のオペランドおよびそれ以上含む、3つ以上のベクトルオペランドを指定するSIMD演算に関連し得る。レジスタのペアとして表される宛先ベクトルオペランドを生成するために、混合幅SIMD命令用に単一のレジスタとしてそれぞれ表される、2つのソースベクトルオペランドが指定される一例が開示される。多くの他のそのような命令フォーマットは、本開示の範囲内で可能である。説明を簡単にするために、混合幅SIMD演算を実施するための例示的な態様を、オペランドのいくつかの例示的なSIMD命令およびビット幅に関連して説明するが、これらは単に説明のためのものであることに留意する。このように、本明細書で論じられる特徴は、混合幅ベクトル演算のための任意の数のオペランドおよびデータ要素のビット幅に拡張され得る。

【0029】

図2A～図2Cに、SIMD命令200、220、および240に関する例示的な態様が示されている。これらのSIMD命令200、220、および240の各々は、SIMD命令を実行するように構成されたプロセッサ(たとえば、図4に示されるプロセッサ402)によって実行され得る。より具体的には、これらのSIMD命令200、220、および240の各々は、1つまたは複数のソースベクトルオペランドおよび1つまたは複数の宛先ベクトルオペランドを指定することができ、ソースベクトルオペランドおよび宛先ベクトルオペランドは、レジスタ(たとえば、64ビットレジスタ)に関して表され得る。SIMD命令200、220、および240のソースベクトルオペランドおよび宛先ベクトルオペランドは、それぞれが1つまたは複数のSIMDレーンに該当する、対応するソースデータ要素および宛先データ要素を含む。SIMD命令の実行におけるSIMDレーンの数は、SIMD命令の実行において実行される並列演算の数に対応する。したがって、例示的なSIMD命令200、220、および240を実装するように構成されたプロセッサまたは実行ロジックは、SIMD命令200、220、および240によって指定された並列演算を実施するために必要とされるハードウェア(たとえば、多数の左/右シフタ、加算器、乗算器等を備える算術論理装置(ALU))を含むことができる。

【0030】

したがって、図2Aを参照すると、SIMD命令200の実行のための第1の例示的な態様が示されている。一例では、プロセッサは、64ビット命令セットアーキテクチャ(ISA)をサポート

10

20

30

40

50

トすることができる。SIMD命令200は、単一の64ビットレジスタに関して表されるソースベクトルオペランドのソースデータ要素に対して実行されるべき同じ演算または共通の命令を指定することができる。

【0031】

SIMD命令200において指定される同じ演算または共通の命令は、たとえば、8ビットのソースデータ要素に対する二乗関数、左シフト関数、インクリメント関数、一定値による加算などであり得(これは、8個の8ビット左シフタ、8個の8ビット加算器などの論理要素で実装され得る)、対応する8個の結果として得られる宛先データ要素を生成し、これは最大16ビットのストレージを消費し得る。図示されるように、SIMD命令200は、8個の8ビットデータ要素を備えるソースベクトルオペランド202を指定し得る。ソースベクトルオペランド202のこれらの8個の8ビットデータ要素には数値順が割り当てられ得、これは参照番号0~7で示されている。SIMD命令200の結果は、8個の16ビット宛先データ要素または128ビットをとともに使用して表され得、これは単一の64ビットレジスタに記憶することはできない。この問題を処理するためにSIMD命令200を2つ以上の命令に分解するのではなく(たとえば、図1A~図1Cに示されるSIMD命令100の従来の実装形態のように)、宛先ベクトルオペランドは構成要素ベクトルオペランドのペアとして指定される。構成要素宛先ベクトルオペランドのペアは、対応するレジスタ204x、204yのペアとして表され得る。レジスタのペアは、レジスタファイル内の連続する物理的位置に記憶される必要はなく、連続する論理レジスタ番号を有することさえできる点に留意されたい。このように、SIMD命令200は、構成要素ベクトルオペランドまたはレジスタ204x、204yのペア(たとえば、64ビットレジスタのペア)に関して表される宛先ベクトルオペランドと、単一のレジスタ202として表されるソースベクトルオペランド202とを指定する。

【0032】

さらに、ペアの第1のレジスタ204xとして表される第1の構成要素宛先ベクトルオペランドは、ソースベクトルオペランド202の偶数番号のソースデータ要素0、2、4、および6に対して実行されるSIMD命令200の結果の第1のサブセットを含む。これらの結果は、偶数番目のソースデータ要素0、2、4、および6への1対1の対応を有する宛先データ要素A、C、E、およびGによって示されており、これは、宛先データ要素A、C、E、およびGのこの例示的な配置において、結果に関してSIMDレーンを横切る大きな動きが回避されることを意味する。同様に、ペアの第2のレジスタ204yとして表される第2の構成要素宛先ベクトルオペランドは、ソースベクトルオペランド202の奇数番号のソースデータ要素1、3、5、および7に対して実行されるSIMD命令200の結果の第2のサブセットを含む。これらの結果は、奇数番号のソースデータ要素1、3、5、および7への1対1の対応を有する宛先データ要素B、D、F、およびHによって示されており、これもやはり、宛先データ要素B、D、F、およびHのこの例示的な配置において、結果に関してSIMDレーンを横切る大きな動きが回避されることを意味する。したがって、この場合、ソースベクトルオペランド202の偶数番号のソースデータ要素0、2、4、および6は、第1のレジスタ204xの宛先データ要素A、C、E、およびGに対応するか、またはそれを生成し、ソースベクトルオペランド202の奇数番号のソースデータ要素1、3、5、および7は、第2のレジスタ204yの宛先データ要素B、D、F、およびHに対応するか、またはそれを生成する。

【0033】

たとえばSIMDレーン0~7と呼ばれる、各レーンがそれぞれのソースデータ要素0~7を備える8個の8ビットSIMDレーンを考えると、対応する宛先データ要素A~Hを生成するために必要な移動量は、同じSIMDレーンまたは隣接するSIMDレーン内に含まれることが分かる。言い換えれば、第1のセットのソースデータ要素(たとえば、ソースデータ要素0~7)は、それぞれのSIMDレーンにあり、ソースデータ要素の各々から、宛先データ要素(たとえば、対応する宛先データ要素A~H)が、それぞれのSIMDレーン、またはそれぞれのSIMDレーンに隣接するSIMDレーンにおいて生成される。たとえば、SIMDレーン0、2、4、および6内の偶数番号のソースデータ要素0、2、4、および6はそれぞれ、宛先データ要素A、C、E、およびGを生成し、それらはそれぞれSIMDレーン0~1、2~3、4~5、および6~7内に含ま

れる。同様に、SIMDレーン0、2、4、および6内の奇数番号のソースデータ要素1、3、5、および7はそれぞれ、宛先データ要素B、D、F、およびHを生成し、それらはそれぞれSIMDレーン0~1、2~3、4~5、および6~7内に含まれる。

#### 【0034】

したがって、図2Aの第1の例示的な態様では、混合幅SIMD命令200は、(2つ以上の構成要素SIMD命令ではなく、1つのSIMD命令のみが使用されるので)命令空間またはコード空間の効率的な使用を含み、その実装形態または実行はSIMDレーンを横切る大きなデータ移動を回避する。

#### 【0035】

次に図2Bを参照すると、別の例示的な態様が、混合幅SIMD命令220に関連して示されている。SIMD命令220は、単一のレジスタ222として表される第1のソースベクトルオペランドと、単一のレジスタ223として表される第2のソースベクトルオペランドとの、2つのソースベクトルオペランドを含み、4個の16ビットのソースデータ要素の第1のセットおよび第2のセットをそれぞれ有する。SIMD命令220は、2つのソースベクトルオペランドに対する乗算(たとえば、丸めを伴う)などの同じまたは共通の演算を指定することができ、4個の32ビット結果を生成するために、(レジスタ222内の)第1のセットの4個の16ビットソースデータ要素が、(レジスタ223内の)第2のセットの対応する4個の16ビットソースデータ要素と乗算される(SIMD命令220の実装形態は、4個の16×16乗算器などの論理要素を含むことができる)。これらの4個の32ビット結果を記憶するために128ビットが必要とされるので、宛先ベクトルオペランドは、第1の構成要素宛先ベクトルオペランドと第2の構成要素宛先ベクトルオペランドとの、構成要素ベクトルオペランドのペアに関して指定される(これらは、それに応じて第1の64ビットレジスタ224xと、第2の64ビットレジスタ224yとして表され得る)。SIMD命令220はまた、第2のセットの対応するソースデータ要素を有する第1のセットのソースデータ要素の追加にも適用可能であり、対応する結果は、宛先データ要素ごとに16ビットより多くを消費し得る(32ビットすべてではないとしても)点に留意されたい。

#### 【0036】

図2Bにおいて、第1および第2のセットのソースデータ要素は、それぞれ0、1、2、3、および0'、1'、2'、3'として代表的に示されている順序を割り当てられる。第1のレジスタ224x内の第1の構成要素宛先ベクトルオペランドは、ソースオペランド222および223の偶数番号のソースデータ要素に対応するSIMD命令220の結果の第1のサブセット(32ビット宛先データ要素AおよびCとして示される)を保持し、同様に、第2のレジスタ224y内の第2の構成要素宛先ベクトルオペランドは、ソースオペランド222および223の奇数番号のソースデータ要素に対応するSIMD命令220の結果の第2のサブセット(32ビットデータ要素BおよびDとして示される)を保持する。この場合、第1のソースベクトルオペランド222および第2のソースベクトルオペランド223の偶数番号のソースデータ要素(0,0')および(2,2')は、それぞれ、第1の宛先ベクトルオペランド224xのデータ要素AおよびCを生成し、第1のソースベクトルオペランド222および第2のソースベクトルオペランド223の奇数番号のデータ要素(1,1')および(3,3')は、それぞれ、第2の宛先ベクトルオペランド224yのデータ要素BおよびDを生成することが分かる。

#### 【0037】

再び、図2Bの第2の例示的な態様においては、混合幅SIMD命令220は、2つ以上の構成要素SIMD命令ではなく、単一の混合幅SIMD命令を利用することによって、コード空間効率を達成することが分かる。さらに、この態様においても、SIMDレーンを横切る動きは最小限に抑えられることも分かる。一般に、第1のセットのソースデータ要素および第2のセットのソースデータ要素は、それぞれのSIMDレーンにあり、第1のセットのソースデータ要素の各ソースデータ要素、および第2のセットのソースデータ要素のうちの対応するソースデータ要素から、それぞれのSIMDレーン、またはそれぞれのSIMDレーンに隣接するSIMDレーン内の宛先データ要素を生成する。たとえば、第1のセットのソースデータ要素0~3(または、第2のセットのソースデータ要素0~3')を備える4個の16ビットSIMDレーン0~3を考

10

20

30

40

50

えると、それぞれ、対応する宛先データ要素A~Dを生成するための第1および第2のソースデータ要素のためのデータ移動は、同じSIMDレーンおよび多くとも隣接するSIMDレーンに含まれる(たとえば、SIMDレーン0および2における偶数番号のソースデータ要素(0,0')および(2,2')は、それぞれSIMDレーン0~1および2~4において宛先データ要素AおよびCを生成し、同様に、SIMDレーン1および3における奇数番号のソースデータ要素(1,1')、および(3,3')は、それぞれSIMDレーン0~1および2~4において宛先データ要素BおよびDを生成する)。

#### 【0038】

図2Cは、混合幅SIMD命令240に関する第3の例示的な態様を表す。混合幅SIMD命令200および220とは異なり、混合幅SIMD命令240のソースベクトルオペランドは、構成要素ベクトルオペランドのペアとして指定されるか、またはレジスタのペアとして表される。混合幅SIMD命令220は2つの別個のソースベクトルオペランドに含まれ、1つのソースベクトルオペランドのデータ要素が別のソースベクトルオペランドのデータ要素と相互作用する(たとえば、それと乗算される)ように指定されているため、混合幅SIMD命令240は混合幅SIMD命令220とは異なる点に留意されたい。一方、混合幅SIMD命令240においては、そうしないと2つの別個の命令が消費されてしまうので、構成要素ソースベクトルオペランドのペアが指定される。たとえば、SIMD命令240は、8個の8ビット宛先データ要素の結果を得るために、8個の16ビットソースデータ要素に対して実行されるべき16ビットから8ビットへの右シフト関数の共通の演算を含むことができる(SIMD命令240の実装形態は、8個の8ビット右シフトなどの論理要素を含むことができる)。しかしながら、8個の16ビットソースデータ要素は128ビットを消費するので、従来の実装形態は、2つの構成要素SIMD命令を使用して実行されるべきこの演算を分割することになる。一方、図2Cの例示的な態様では、第1のレジスタ242x内の第1の構成要素ソースベクトルオペランドと、第2のレジスタ242y内の第2の構成要素ソースベクトルオペランドとを備えるソースベクトルオペランドのペアが、SIMD命令240によって指定される。したがって、コード空間が効率的に使用される。

#### 【0039】

宛先ベクトルオペランドは、この場合は単一の64ビットレジスタ244として表され、SIMD命令240の結果である8個の8ビット宛先データ要素を備える。したがって、レジスタ244内の宛先ベクトルオペランドの宛先データ要素には順序が割り当てられ、これらの要素は参照番号0~7で示されている。構成要素ソースベクトルオペランドのペアのソースデータ要素(242x、242yのペアとして表される)は、ソースデータ要素A、C、E、およびGの第1のサブセットを備える第1のレジスタ242xが、それぞれレジスタ244内の宛先ベクトルオペランドの偶数番号の宛先データ要素0、2、4、および6に対応する結果を生成し、ソースデータ要素B、D、F、およびHの第2のサブセットを備える第2のレジスタ242yが、それぞれレジスタ244内の宛先ベクトルオペランドの奇数番号の宛先データ要素1、3、5、および7に対応する結果を生成するように配置される。

#### 【0040】

したがって、ソースベクトルオペランドが宛先ベクトルオペランドよりも広い場合でも、構成要素ソースベクトルオペランドのペアを指定するか、またはソースベクトルオペランドをレジスタのペアとして表すことによって、コード空間が有効に利用され得、SIMDレーンを横切るデータ移動が最小限に抑えられ得る。SIMD命令240の実行時にSIMDレーンを横切る移動も最小化される。一般に、宛先データ要素はそれぞれのSIMDレーンにあり、宛先データ要素の各々は、それぞれのSIMDレーン、またはそれぞれのSIMDレーンに隣接するSIMDレーン内のソースデータ要素から生成されることが分かる。たとえば、8個の宛先データ要素0~7に対応する8個の8ビットSIMDレーンを考えると、ソースデータ要素A、C、E、およびGは、SIMDレーン0、2、4、および6内の偶数番号の宛先データ要素に対応する結果を生成するために、それぞれSIMDレーン0~1、2~3、4~5、および6~7から移動し、ソースデータ要素B、D、F、およびHは、SIMDレーン1、3、5、および7内の偶数番号の宛先データ要素に対応する結果を生成するために、それぞれSIMDレーン0~1、2~3、4~5、および6~7から移動することが分かる。どちらの場合も、移動は2つのSIMDレーン内に含まれ

10

20

30

40

50

る。

【 0 0 4 1 】

したがって、態様は、本明細書において開示されたプロセス、関数、および/またはアルゴリズムを実施するための様々な方法を含むことが理解されよう。たとえば、図3Aに示されるように、ある態様は、たとえば図2A~図2Bによる、混合幅単一命令複数データ(SIMD)演算を実行する方法300を含むことができる。

【 0 0 4 2 】

ブロック302において、方法300は、プロセッサ(たとえば、以下で説明される図4のプロセッサ402)によって、おおよびたとえば図2Aを参照して、第1のビット幅(たとえば、8ビット)の第1のセットのソースデータ要素(たとえば、ソースデータ要素0~7)を備える少なくとも1つの第1のソースベクトルオペランド(たとえば、レジスタ202における)と、第2のビット幅(たとえば、16ビット)の宛先データ要素(たとえば、宛先データ要素A~H)を備える少なくとも1つの宛先ベクトルオペランド(たとえば、レジスタのペア204x、204yにおける)とを備えるSIMD命令(たとえば、SIMD命令200)を受信するステップを含み、第2のビット幅は第1のビット幅の2倍であり、宛先ベクトルオペランドは、宛先データ要素の第1のサブセット(たとえば、宛先データ要素A、C、E、G)を備える第1のレジスタ(たとえば、204x)と、宛先データ要素の第2のサブセット(たとえば、宛先データ要素B、D、F、H)を備える第2のレジスタとを含むレジスタのペアを備える。

【 0 0 4 3 】

ブロック303(ブロック304および306を含むものとして示される)において、方法300は、プロセッサ内で混合幅SIMD命令を実行するステップをさらに含む。具体的には、ブロック304においてソースデータ要素に割り当てられた順序(たとえば、0~7)を考えると、ブロック306は、プロセッサ内でSIMD命令を実行するステップを含む。さらに詳細においては、ブロック306は、並列に実行され得るブロック306aおよび306bの構成要素からなる。

【 0 0 4 4 】

ブロック306aは、第1のセットの偶数番号のソースデータ要素(たとえば、ソースデータ要素0、2、4、6)から、第1のレジスタ(たとえば、第1のレジスタ204x)内の宛先データ要素(たとえば、宛先データ要素A、C、E、G)の第1のサブセットを生成するステップを含む。

【 0 0 4 5 】

ブロック306bは、第1のセットの奇数番号のソースデータ要素(たとえば、ソースデータ要素1、3、5、7)から、第2のレジスタ(たとえば、第2のレジスタ204y)内の宛先データ要素(たとえば、宛先データ要素B、D、F、H)の第2のサブセットを生成するステップを含む。

【 0 0 4 6 】

一般に、方法300のSIMD命令は、第1のセットのソースデータ要素の二乗関数、左シフト関数、インクリメント、または一定値による加算のうちの1つであり得る。コード空間効率率は、方法300において単一のSIMD命令を利用することによって達成される。方法300においては、SIMDレーンを横切る移動も最小化され、第1のセットのソースデータ要素はそれぞれのSIMDレーン内にあり、方法300は、ソースデータ要素(たとえば、SIMDレーン0内のソースデータ要素0)のそれぞれから、それぞれのSIMDレーン(たとえば、SIMDレーン0)内の宛先データ要素(たとえば、宛先データ要素A)、またはそれぞれのSIMDレーンに隣接するSIMDレーン(たとえば、SIMDレーン1)を生成するステップを含む。

【 0 0 4 7 】

別個に示されていないが、方法300はまた、図2BのSIMD命令220を実装するための方法を含むことができ、この方法は、たとえば、ブロック302において、第1のビット幅の第2のセットのソースデータ要素(たとえば、レジスタ222および223内の第1および第2のソースベクトルオペランド)を備える第2のソースベクトルオペランドを受信するステップをさらに備え、第1のセットのソースデータ要素の順序は、第2のセットのソースデータ要素の順序に対応する点にも留意されたい。この場合、ブロック304において割り当てられた順序

に基づいて、ブロック306は、プロセッサ内でSIMD命令を実行するステップを含み、第1のセットの偶数番号のソースデータ要素、および第2のセットの偶数番号のソースデータ要素から、第1のレジスタ内の宛先データ要素の第1のサブセットを生成するためのブロック306aと、第1のセットの奇数番号のソースデータ要素および、第2のセットの偶数番号ソースデータ要素から、第2のレジスタ内の宛先データ要素の第2のサブセットを生成するためのブロック306bとを備える。この場合、SIMD命令は、第1のセットのソースデータ要素と、第2のセットの対応するソースデータ要素との乗算または加算であり得、第1のセットのソースデータ要素および第2のセットのソースデータ要素はそれぞれSIMDレーン内にあり、第1のセットのソースデータ要素の各ソースデータ要素と第2のセットのソースデータ要素のうちの対応するソースデータ要素とから、それぞれのSIMDレーン、またはそれぞれのSIMDレーンに隣接するSIMDレーン内の宛先データ要素を生成する。

10

【0048】

図3Bを参照すると、本明細書において開示されたプロセス、関数、および/またはアルゴリズムを実行するための別の方法が示されている。たとえば、図3Bに示されるように、方法300は、たとえば図2Cによる、混合幅単一命令複数データ(SIMD)演算を実行する別の方法を含む。

【0049】

ブロック352において、方法350は、プロセッサ(たとえば、プロセッサ402)によって、第1のビット幅(たとえば、16ビット)のソースデータ要素(たとえば、ソースデータ要素A~H)を備える少なくとも1つのソースベクトルオペランド(たとえば、レジスタ242x、242y)における)と、第2のビット幅(たとえば、8ビット)の宛先データ要素(たとえば、宛先データ要素0~7)を備える少なくとも1つの宛先ベクトルオペランド(たとえば、レジスタ244における)とを備えるSIMD命令(たとえば、SIMD命令240)を受信するステップを含み、第2のビット幅は第1のビット幅の半分であり、ソースベクトルオペランドは、ソースデータ要素の第1のサブセット(たとえば、宛先データ要素0、2、4、6)を備える第1のレジスタ(たとえば、第1のレジスタ242x)と、ソースデータ要素の第2のサブセット(たとえば、宛先データ要素1、3、5、7)を備える第2のレジスタ(たとえば、第2のレジスタ242y)とを含むレジスタのペアを備える。

20

【0050】

ブロック354において、宛先データ要素に順序が割り当てられ、ブロック356において、SIMD命令が実行される。ブロック356は、サブブロック356aおよび356bを含み、これらはまた、並列に実行され得る。

30

【0051】

ブロック356aは、第1のレジスタ内のソースデータ要素の対応する第1のサブセット(たとえば、ソースデータ要素A、C、E、G)から、偶数番号の宛先データ要素(たとえば、宛先データ要素0、2、4、6)を生成するステップを含む。

【0052】

ブロック356bは、第2のレジスタ内のソースデータ要素の対応する第2のサブセット(たとえば、ソースデータ要素B、D、F、H)から、奇数番号の宛先データ要素(たとえば、宛先データ要素1、3、5、7)を生成するステップを含む。

40

【0053】

例示的な態様では、方法350のSIMD命令は、ソースデータ要素の右シフト関数であってもよく、宛先データ要素は、それぞれのSIMDレーン(たとえば、SIMDレーン0~7)内にあり、それぞれのSIMDレーン(たとえば、SIMDレーン0)、またはそれぞれのSIMDレーンに隣接するSIMDレーン(たとえば、SIMDレーン1)内のソースデータ要素(たとえば、ソースデータ要素A)から、宛先データ要素(たとえば、宛先データ要素0)の各々を生成する。

【0054】

図4を参照すると、例示的な態様によるワイヤレスデバイス400の特定の例示的な態様のブロック図である。ワイヤレスデバイス400は、たとえば図3Aの方法300および図3Bの方法350による、例示的な混合幅SIMD命令の実行をサポートおよび実装するように構成され得

50

る(たとえば、実行ロジックを含む)プロセッサ402を含む。図4に示されるように、プロセッサ402は、メモリ432と通信することができる。プロセッサ402は、例示的なSIMD命令のどのオペランドが表されるかに関してレジスタ(たとえば、論理レジスタ)に対応する物理レジスタを保持するレジスタファイル(図示せず)を含み得る。いくつかの態様では、レジスタファイルにはメモリ432からデータが供給され得る。図示されていないが、1つもしくは複数のキャッシュまたは他のメモリ構造もワイヤレスデバイス400に含まれ得る。

【0055】

図4は、プロセッサ402およびディスプレイ428に結合された、ディスプレイコントローラ426も示している。コード/デコード(コーデック)434(たとえば、オーディオおよび/または音声コーデック)は、プロセッサ402に結合することができる。(モデムを含んでもよい)ワイヤレスコントローラ440などの他の構成要素も示されている。スピーカ436およびマイクロフォン438は、コーデック434に結合することができる。図4は、ワイヤレスコントローラ440をワイヤレスアンテナ442に結合することも示している。特定の態様では、プロセッサ402、ディスプレイコントローラ426、メモリ432、コーデック434、およびワイヤレスコントローラ440は、システムインパッケージデバイスまたはシステムオンチップデバイス422に含まれる。

【0056】

特定の態様では、入力デバイス430および電源444は、システムオンチップデバイス422に結合される。さらに、ある特定の態様では、図4に示されるように、ディスプレイ428、入力デバイス430、スピーカ436、マイクロフォン438、ワイヤレスアンテナ442、および電源444は、システムオンチップデバイス422の外部に位置する。ただし、ディスプレイ428、入力デバイス430、スピーカ436、マイクロフォン438、ワイヤレスアンテナ442、および電源444の各々は、インターフェースまたはコントローラのような、システムオンチップデバイス422の構成要素に結合することができる。

【0057】

図4は、ワイヤレス通信デバイスを示しているが、プロセッサ402およびメモリ432はまた、セットトップボックス、音楽プレーヤ、ビデオプレーヤ、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、携帯情報端末(PDA)、固定位置データユニット、通信デバイス、またはコンピュータに統合され得る点に留意されたい。さらに、ワイヤレスデバイス400の少なくとも1つまたは複数の例示的な態様は、少なくとも1つの半導体ダイに統合され得る。

【0058】

当業者であれば、情報および信号は、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得ることを理解するであろう。たとえば、上記の説明を通じて参照され得るデータ、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得る。

【0059】

さらに、当業者は、本明細書において開示された態様に関連して説明された様々な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ステップが、電子ハードウェア、コンピュータソフトウェア、またはその両方の組合せとして実装され得ることを理解するであろう。ハードウェアおよびソフトウェアのこの互換性を明確に示すために、種々の例示的な構成要素、ブロック、モジュール、回路、およびステップについて、上記では概してそれらの機能に関して説明してきた。そのような機能がハードウェアとして実装されるか、またはソフトウェアとして実装されるかは、特定のアプリケーションおよびシステム全体に課される設計制約に依存する。当業者は、特定のアプリケーションごとに様々な方法で説明した機能を実装し得るが、そのような実装形態の決定は本発明の範囲から逸脱するものと解釈されるべきではない。

【0060】

本明細書において開示された態様に関連して説明された方法、シーケンス、および/ま

10

20

30

40

50

たはアルゴリズムは、ハードウェア、プロセッサによって実行されるソフトウェアモジュール、またはその2つの組合せにおいて直接的に実施され得る。ソフトウェアモジュールは、RAMメモリ、フラッシュメモリ、ROMメモリ、EPROMメモリ、EEPROMメモリ、レジスタ、ハードディスク、リムーバブルディスク、CD-ROM、または当技術分野で知られている任意の他の形態の記憶媒体内に存在してもよい。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、かつ記憶媒体に情報を書き込むことができるように、プロセッサに結合される。あるいは、記憶媒体は、プロセッサに一体化され得る。

#### 【 0 0 6 1 】

したがって、本発明のある態様は、(たとえば、上述の方法300および350による、図2A ~ 図2CのSIMD命令を実装するための)混合幅SIMD命令を実装するための方法を具体化するコンピュータ可読媒体(たとえば、非一時的コンピュータ可読記憶媒体)を含むことができる。したがって、本発明は図示された例に限定されず、本明細書に記載の機能を実行するための任意の手段が本発明の態様に含まれる。

10

#### 【 0 0 6 2 】

前述の開示は本発明の例示的な態様を示しているが、添付の特許請求の範囲によって規定される本発明の範囲から逸脱することなしに、本明細書に様々な変更および修正が行われ得る点に留意されたい。本明細書に記載される本発明の態様に従う方法クレームの機能、ステップ、および/または行為を、任意の特定の順序で実施する必要はない。さらに、本発明の要素は、単数形で記載され、特許請求される場合があるが、単数形への制限が明示的に言及されない限り、複数形が意図される。

20

#### 【符号の説明】

#### 【 0 0 6 3 】

100 SIMD命令  
 100X SIMD命令  
 100Y SIMD命令  
 102 ソースオペランド  
 104x 宛先オペランド  
 104y 宛先オペランド  
 120X SIMD命令  
 120Y SIMD命令  
 200 SIMD命令  
 202 ソースベクトルオペランド  
 202 単一のレジスタ  
 204x 第1のレジスタ  
 204y 第2のレジスタ  
 220 SIMD命令  
 220 混合幅SIMD命令  
 222 単一のレジスタ  
 222 ソースオペランド  
 222 第1のソースベクトルオペランド  
 223 単一のレジスタ  
 223 ソースオペランド  
 223 第2のソースベクトルオペランド  
 224x 第1のレジスタ  
 224x 第1の宛先ベクトルオペランド  
 224y 第2のレジスタ  
 224y 第2の宛先ベクトルオペランド  
 240 SIMD命令  
 240 混合幅SIMD命令  
 242x 第1のレジスタ

30

40

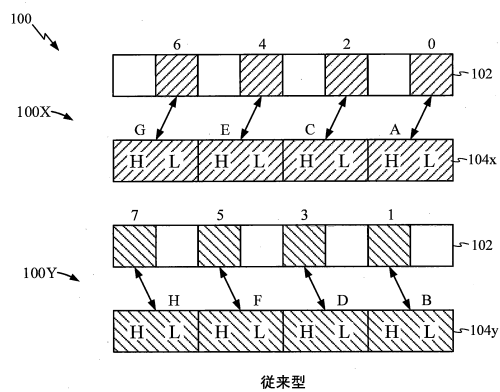
50



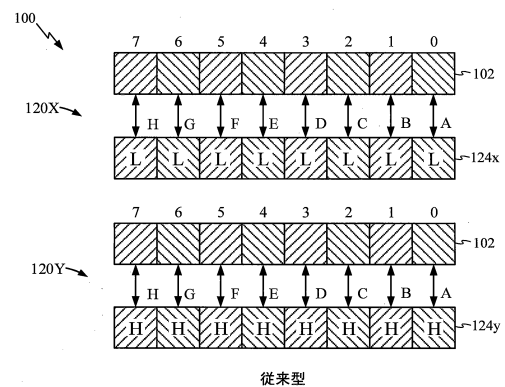
242y 第2のレジスタ  
 244 レジスタ  
 300 方法  
 350 方法  
 400 ワイヤレスデバイス  
 402 プロセッサ  
 422 システムオンチップデバイス  
 426 ディスプレイコントローラ  
 428 ディスプレイ  
 430 入力デバイス  
 432 メモリ  
 434 コーダ/デコーダ(コーデック)  
 436 スピーカー  
 438 マイクロフォン  
 440 ワイヤレスコントローラ  
 442 ワイヤレスアンテナ  
 444 電源

10

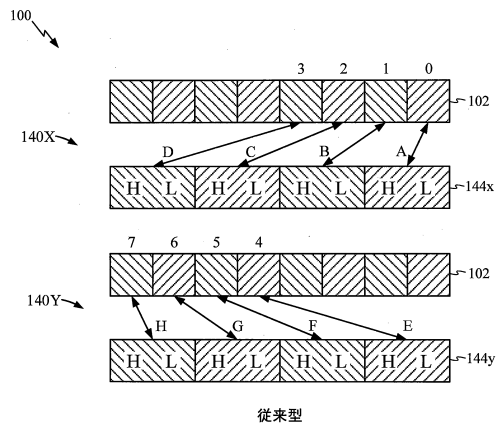
【図 1 A】



【図 1 B】



【図 1 C】



【図 2 A】

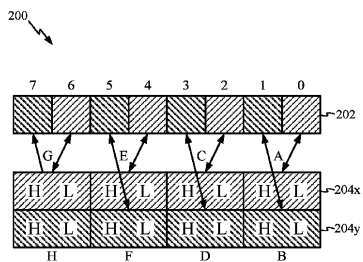
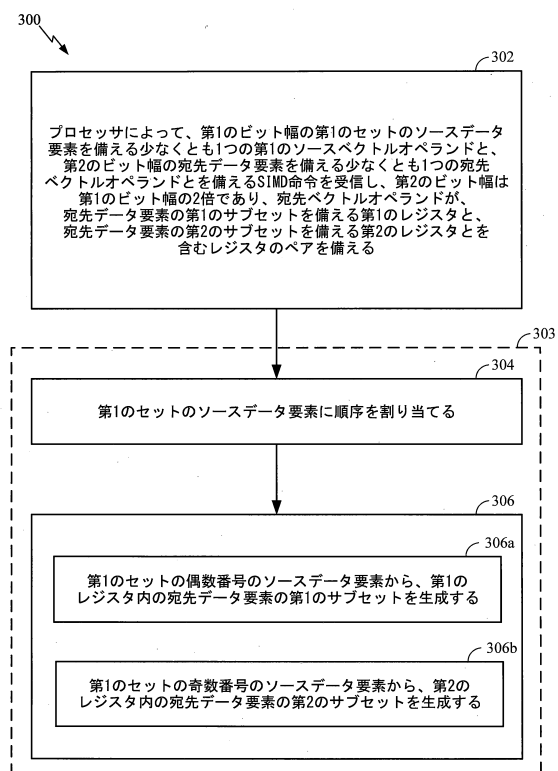


FIG. 2A

【図 3 A】



【図 2 B】

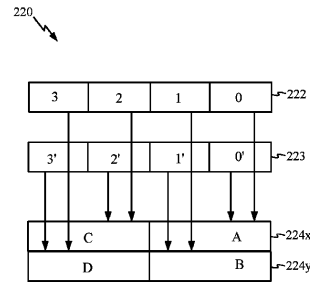


FIG. 2B

【図 2 C】

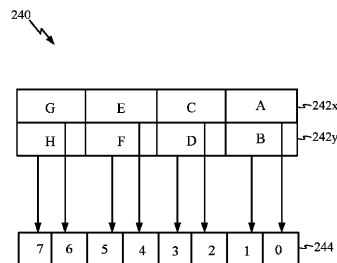
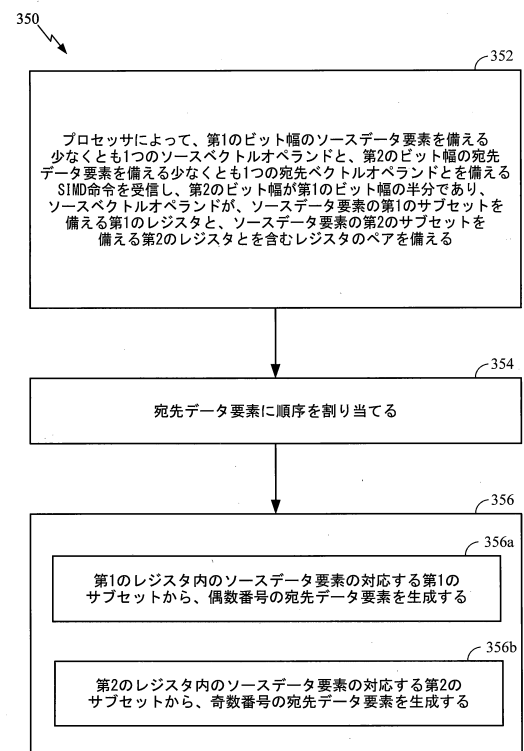
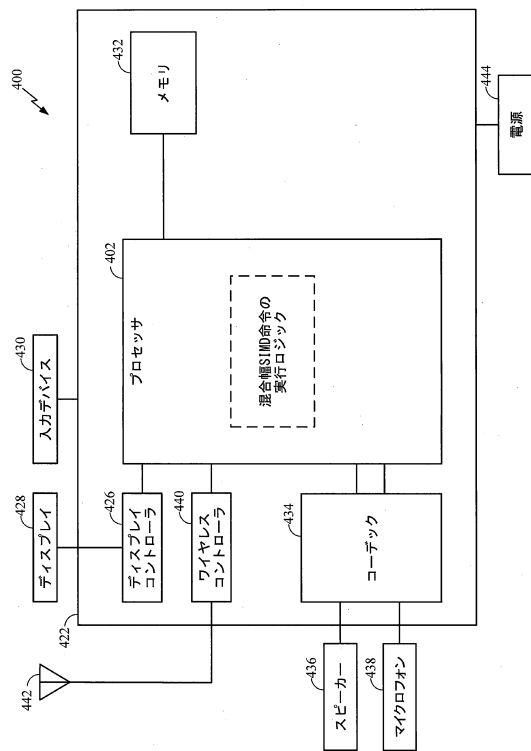


FIG. 2C

【図 3 B】



【図4】



---

フロントページの続き

(72)発明者 アジャイ・アナント・イングル  
アメリカ合衆国・カリフォルニア・９２１２１－１７１４・サン・ディエゴ・モアハウス・ドライ  
ヴ・５７７５・クアルコム・インコ ポレイテッド

審査官 井上 宏一

(56)参考文献 特開２００５－１７４２９５（ＪＰ，Ａ）  
米国特許出願公開第２００７／００３３３８１（ＵＳ，Ａ１）  
米国特許第０７２１９２１２（ＵＳ，Ｂ１）

(58)調査した分野(Int.Cl.，ＤＢ名)  
G 0 6 F 9 / 3 8  
G 0 6 F 9 / 3 4  
G 0 6 F 9 / 3 0 2  
G 0 6 F 9 / 3 1 5