

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6243000号
(P6243000)

(45) 発行日 平成29年12月6日 (2017. 12. 6)

(24) 登録日 平成29年11月17日 (2017. 11. 17)

(51) Int. Cl.

F I

G 0 6 F 17/16 (2006. 01)
G 0 6 F 9/38 (2006. 01)G 0 6 F 17/16 S
G 0 6 F 9/38 3 1 0 G

請求項の数 28 (全 40 頁)

(21) 出願番号 特願2016-500908 (P2016-500908)
 (86) (22) 出願日 平成26年3月7日 (2014. 3. 7)
 (65) 公表番号 特表2016-517570 (P2016-517570A)
 (43) 公表日 平成28年6月16日 (2016. 6. 16)
 (86) 国際出願番号 PCT/US2014/022162
 (87) 国際公開番号 W02014/164367
 (87) 国際公開日 平成26年10月9日 (2014. 10. 9)
 審査請求日 平成29年2月3日 (2017. 2. 3)
 (31) 優先権主張番号 13/798, 641
 (32) 優先日 平成25年3月13日 (2013. 3. 13)
 (33) 優先権主張国 米国 (US)

早期審査対象出願

(73) 特許権者 595020643
 クアアルコム・インコーポレイテッド
 QUALCOMM INCORPORATED
 アメリカ合衆国、カリフォルニア州 92
 121-1714、サン・ディエゴ、モア
 ハウス・ドライブ 5775
 (74) 代理人 100108855
 弁理士 蔵田 昌俊
 (74) 代理人 100109830
 弁理士 福原 淑弘
 (74) 代理人 100158805
 弁理士 井関 守三
 (74) 代理人 100194814
 弁理士 奥村 元宏

最終頁に続く

(54) 【発明の名称】 マルチモードベクトル処理を提供するためのプログラム可能データ経路構成を有するベクトル処理エンジン、ならびに関連ベクトルプロセッサ、システム、および方法

(57) 【特許請求の範囲】

【請求項 1】

ベクトルデータのマルチモードベクトル処理を行うように構成されたベクトル処理エンジン (VPE) であって、

複数の入力データ経路中で、複数のベクトルデータ入力サンプルセットを与えるように構成された入力読取り段と、

複数のベクトル処理ブロックを備える少なくとも1つのベクトル処理段と、前記複数のベクトル処理ブロックの中の各ベクトル処理ブロックは、

前記複数の入力データ経路の中の少なくとも2つの入力データ経路から、前記複数のベクトルデータ入力サンプルセットからの少なくとも2つのベクトルデータ入力サンプルセットを受信し、

前記少なくとも1つのベクトル処理段によって実行されるベクトル命令に従って、前記ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて少なくとも1つのベクトル結果出力サンプルセットを与えるように、前記少なくとも2つのベクトルデータ入力サンプルセットを処理し、ここにおいて、前記ベクトル処理ブロックは複数の乗算器を備え、前記ベクトル処理ブロックによる前記少なくとも2つのベクトルデータ入力サンプルセットの前記処理は、第1の構成における前記複数の乗算器を使用して前記少なくとも2つのベクトルデータ入力サンプルセットに対して第1の乗算演算を実施することを備え、前記プログラム可能データ経路構成は、第2の乗算演算を実施するために前記複数の乗算器を第2の構成に再構成するように、プログラム可能に構成され、前記第1の乗算演

10

20

算と前記第 2 の乗算演算は、異なるビット長の乗算演算である、

複数の出力データ経路の中の少なくとも 1 つの出力データ経路中で前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるように構成される、

前記複数のベクトル処理ブロックの各々から前記少なくとも 1 つのベクトル結果出力サンプルセットを受信するように構成された出力処理段と、

を備える V P E。

【請求項 2】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも 1 つのベクトル処理段によって実行される各ベクトル命令用に再構成されるように構成される、請求項 1 に記載の V P E。

10

【請求項 3】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令の各クロック周期ごとに、クロック周期単位で再構成されるように構成される、請求項 1 に記載の V P E。

【請求項 4】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、プログラム可能入力データ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記プログラム可能入力データ経路構成に基づいて前記複数の入力データ経路の中の前記少なくとも 2 つの入力データ経路から前記少なくとも 2 つのベクトルデータ入力サンプルセットを受信するように構成される、請求項 1 に記載の V P E。

20

【請求項 5】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、さらに、前記少なくとも 1 つのベクトル処理ブロック用のプログラム可能出力データ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記プログラム可能出力データ経路構成に基づいて前記複数の出力データ経路の中の前記少なくとも 1 つの出力データ経路中で前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるように構成される、請求項 4 に記載の V P E。

30

【請求項 6】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、さらに、前記少なくとも 1 つのベクトル処理ブロック用のプログラム可能ベクトル処理ブロックデータ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記プログラム可能ベクトル処理ブロックデータ経路構成に基づいて前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるために、前記少なくとも 2 つのベクトルデータ入力サンプルセットを処理するように構成される、請求項 5 に記載の V P E。

40

【請求項 7】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも 1 つのベクトル処理ブロック用のプログラム可能出力データ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記プログラム可能出力データ経路構成に基づいて前記複数の出力データ経路の中の前記少なくとも 1 つの出力データ経路中で前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるように構成される、請求項 1 に記載の V P E。

50

【請求項 8】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも 1 つのベクトル処理ブロック用のプログラム可能ベクトル処理ブロックデータ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記プログラム可能ベクトル処理ブロックデータ経路構成に基づいて前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるために、前記少なくとも 2 つのベクトルデータ入力サンプルセットを処理するように構成される、請求項 1 に記載の V P E。

【請求項 9】

前記少なくとも 1 つのベクトル処理ブロックは、少なくとも 1 つの乗算器ブロックおよび少なくとも 1 つの累算器ブロックのうちの少なくとも 1 つからなる、請求項 1 に記載の V P E。

【請求項 10】

前記少なくとも 1 つのベクトル処理段は、複数のベクトル処理段からなる、請求項 1 に記載の V P E。

【請求項 11】

前記少なくとも 1 つのベクトル処理段は、複数の乗算器ブロックからなる少なくとも 1 つの乗算ベクトル処理段および複数の累算器ブロックからなる少なくとも 1 つの累算ベクトル処理段からなり、

前記複数の乗算器ブロックの中の各乗算器ブロックは、

前記複数のベクトルデータ入力サンプルセットからの第 1 のベクトルデータ入力サンプルセットと第 2 のベクトルデータ入力サンプルセットとを、前記複数の入力データ経路の中の第 1 の入力データ経路および第 2 の入力データ経路から受信し、

前記少なくとも 1 つの第 1 のベクトル処理段によって実行されるベクトル命令に従って、前記乗算器ブロック用のプログラム可能乗算データ経路構成に基づいて複数の乗算出力データ経路の中のある乗算出力データ経路中でベクトル乗算出力サンプルセットを与えるために、前記第 1 のベクトルデータ入力サンプルセットを前記第 2 のベクトルデータ入力サンプルセットに乗算するように構成され、

前記複数の累算器ブロックの中の各累算器ブロックは、

複数の乗算出力データ経路の中の第 1 の乗算出力データ経路および第 2 の乗算出力データ経路から、それぞれ、第 1 の乗算出力サンプルセットと第 2 の乗算出力サンプルセットとを受信し、

少なくとも 1 つの第 2 のベクトル処理段によって実行されるベクトル命令に従って、前記累算器ブロック用のプログラム可能データ経路構成に基づいてベクトル累算結果サンプルセットを与えるために、前記第 1 の乗算出力サンプルセットを前記第 2 の乗算出力サンプルセットと累算し、

前記複数の出力データ経路の中の前記出力データ経路中で前記ベクトル累算結果サンプルセットを与える

ように構成される、請求項 1 に記載の V P E。

【請求項 12】

前記少なくとも 1 つのベクトル処理ブロックは、前記少なくとも 1 つのベクトル結果出力サンプルセットをベクトルレジスタに記憶するように構成されない、請求項 1 に記載の V P E。

【請求項 13】

各ベクトル処理ブロックは、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて異なるビット幅の前記複数のベクトルデータ入力サンプルセットを処理するようにさらに構成される、請求項 1 に記載の V P E。

【請求項 14】

各ベクトル処理ブロックは、前記少なくとも1つのベクトル処理段によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて符号付きの少なくとも1つのベクトルデータ入力サンプルセットまたは符号なしの少なくとも1つのベクトルデータ入力サンプルセットを処理するようにさらに構成される、請求項1に記載のVPE。

【請求項15】

前記少なくとも1つのベクトル処理段は、符号付きの演算命令からなる前記ベクトル命令を実行するように構成される、請求項1に記載のVPE。

【請求項16】

前記少なくとも1つのベクトル処理段は、符号なし演算命令からなる前記ベクトル命令を実行するように構成される、請求項1に記載のVPE。

【請求項17】

ベクトルデータのマルチモードベクトル処理を行うように構成されたベクトル処理エンジン(VPE)であって、

複数の入力データ経路中で、複数のベクトルデータ入力サンプルセットを与えるように構成された入力読取り手段と、

複数のベクトル処理手段を備える少なくとも1つのベクトル処理手段と、前記複数のベクトル処理手段の中の各ベクトル処理手段は、

前記複数の入力データ経路の中の少なくとも2つの入力データ経路から、前記複数のベクトルデータ入力サンプルセットからの少なくとも2つのベクトルデータ入力サンプルセットを受信し、

前記少なくとも1つのベクトル処理手段によって実行されるベクトル命令に従って、前記ベクトル処理手段用のプログラム可能データ経路構成に基づいて少なくとも1つのベクトル結果出力サンプルセットを与えるように、前記少なくとも2つのベクトルデータ入力サンプルセットを処理し、ここにおいて、前記ベクトル処理手段は複数の乗算器を備え、前記ベクトル処理手段による前記処理は、第1の構成における前記複数の乗算器を使用して前記少なくとも2つのベクトルデータ入力サンプルセットに対して第1の乗算演算を実施することを備え、前記プログラム可能データ経路構成は、第2の乗算演算を実施するために前記複数の乗算器を第2の構成に再構成するように、プログラム可能に構成され、前記第1の乗算演算と前記第2の乗算演算は、異なるビット長の乗算演算である、

複数の出力データ経路の中の少なくとも1つの出力データ経路中で前記少なくとも1つのベクトル結果出力サンプルセットを与えるように構成される、

前記複数のベクトル処理手段の各々から前記少なくとも1つのベクトル結果出力サンプルセットを受信するように構成された出力処理手段と、

を備えるVPE。

【請求項18】

ベクトル処理エンジン(VPE)中でベクトルデータを処理する方法であって、

入力処理段中の複数の入力データ経路中で、複数のベクトルデータ入力サンプルセットを与えることと、

少なくとも1つのベクトル処理段中の複数のベクトル処理ブロックにおいて、前記複数のベクトルデータ入力サンプルセットを処理することと、前記複数のベクトル処理ブロックの各々において、

前記複数の入力データ経路の中の少なくとも2つの入力データ経路から、前記複数のベクトルデータ入力サンプルセットからの少なくとも2つのベクトルデータ入力サンプルセットを受信することと、

前記少なくとも1つのベクトル処理段によって実行されるベクトル命令に従って、前記ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて少なくとも1つのベクトル結果出力サンプルセットを与えるように、前記少なくとも2つのベクトルデータ入力サンプルセットを処理することと、ここにおいて、前記ベクトル処理ブロックは複数の乗算器を備え、前記少なくとも2つのベクトルデータ入力サンプルセットを処理するこ

10

20

30

40

50

とは、第 1 の構成における前記複数の乗算器を使用して前記少なくとも 2 つのベクトルデータ入力サンプルセットに対して第 1 の乗算演算を実施することを備える、

複数の出力データ経路の中の少なくとも 1 つの出力データ経路中で前記少なくとも 1 つのベクトル結果出力サンプルセットを与えることと、を備える、

出力処理段中の前記複数のベクトル処理ブロックの各々から前記少なくとも 1 つのベクトル結果出力サンプルセットを受信することと、

第 2 の乗算演算を実施するために前記ベクトル処理ブロックの前記複数の乗算器を第 2 の構成に再構成するように前記複数の処理ブロックの各々用の前記プログラム可能データ経路構成を再構成することと、ここにおいて、前記第 1 の乗算演算と前記第 2 の乗算演算は、異なるビット長の乗算演算である、

を備える方法。

【請求項 19】

前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記プログラム可能入力データ経路構成に基づいて前記複数の入力データ経路の中の前記少なくとも 2 つの入力データ経路から前記複数のベクトルデータ入力サンプルセットからの前記少なくとも 2 つのベクトルデータ入力サンプルセットを受信することと、

前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能入力データ経路構成に基づいて前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるように前記少なくとも 1 つのベクトルデータ入力サンプルセットを処理することと、を備える、請求項 18 に記載の方法

【請求項 20】

前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、プログラム可能出力データ経路構成に基づいて前記複数の出力データ経路の中の前記少なくとも 1 つの出力データ経路中で前記少なくとも 1 つのベクトル結果出力サンプルセットを受信することを備える、請求項 18 に記載の方法。

【請求項 21】

前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用のプログラム可能ベクトル処理ブロックデータ経路構成に基づいて前記少なくとも 1 つのベクトル結果出力サンプルセットを与えるように前記少なくとも 2 つのベクトルデータ入力サンプルセットを処理することを備える、請求項 18 に記載の方法。

【請求項 22】

最小の 1 つのベクトル結果出力サンプルセットをベクトルレジスタに記憶しないことをさらに備える、請求項 18 に記載の方法。

【請求項 23】

前記少なくとも 2 つのベクトルデータ入力サンプルセットを処理することは、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて異なるビット幅の前記複数のベクトルデータ入力サンプルセットを処理することを備える、請求項 18 に記載の方法。

【請求項 24】

前記少なくとも 2 つのベクトルデータ入力サンプルセットを処理することは、前記少なくとも 1 つのベクトル処理段によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて符号付きの少なくとも 1 つのベクトルデータ入力サンプルセットまたは符号なしの少なくとも 1 つのベクトルデータ入力サンプルセットを処理することを備える、請求項 18 に記載の方法。

【請求項 25】

前記ベクトル処理ブロックの各々において前記少なくとも 2 つのベクトルデータ入力サンプルセットを処理することは、

10

20

30

40

50

前記少なくとも2つのデータ入力サンプルセットに対して実施された前記第1の乗算演算によって与えられた乗算サンプルセットを、以前に与えられた乗算サンプルセットに加算することをさらに備える、請求項18に記載の方法。

【請求項26】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも1つのベクトル処理段によって実行される前記ベクトル命令に従って、2つの8ビット×8ビット乗算器または1つの16×16ビット乗算器を形成するように前記複数の乗算器を構成するようにプログラムされる、請求項1に記載のVPE。

【請求項27】

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも1つのベクトル処理段によって実行される前記ベクトル命令に従って、1つの16ビット×16ビット乗算器または1つの24ビット×8ビット乗算器を形成するように前記複数の乗算器を構成するようにプログラムされる、請求項1に記載のVPE。

【請求項28】

前記複数のベクトル処理ブロックの各々は、前記複数の入力データ経路の中の前記入力データ経路のうちの少なくとも4つから、前記複数のベクトルデータ入力サンプルセットからの少なくとも4つのベクトルデータ入力サンプルセットを受信し、前記複数のベクトル処理ブロックの各々の前記複数の乗算器は、第1および第2の乗算器を備え、前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、

前記第1の乗算器の第1の入力に前記少なくとも4つのベクトル入力サンプルセットのうちの第1のベクトル入力サンプルセットを入力し、

前記少なくとも1つのベクトル処理段によって実行される前記ベクトル命令に従って、前記第1の乗算器の第2の入力に、前記少なくとも4つのベクトルデータ入力サンプルセットのうちの第2のベクトルデータ入力サンプルセット、または前記少なくとも4つのベクトルデータ入力サンプルセットのうちの第3のベクトルデータ入力サンプルセットを入力し、

前記第2の乗算器の第1の入力に前記少なくとも4つのベクトル入力サンプルセットのうちの第4のベクトル入力サンプルセットを入力し、

前記少なくとも1つのベクトル処理段によって実行される前記ベクトル命令に従って、前記第2の乗算器の第2の入力に、前記少なくとも4つのベクトルデータ入力サンプルセットのうちの前記第2のベクトルデータ入力サンプルセット、または前記少なくとも4つのベクトルデータ入力サンプルセットのうちの前記第3のベクトルデータ入力サンプルセットを入力する、

ように構成される、請求項1に記載のVPE。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願

[0001]__本出願は、2013年3月13日に出願し、参照によってその全体が本明細書に組み込まれている、「Vector Processing Engines Having Programmable Data Path Configurations For Providing Multi-Mode Radix-2^x Butterfly Vector Processing Circuits, And Related Vector Processors, Systems, And Methods」と題する米国特許出願第13/798,599号(Qualcomm整理番号123247)に関連する。

【0002】

[0002]__本出願は、2013年3月13日に出願し、参照によってその全体が本明細書に組み込まれている、「Vector Processing Carry-Save Accumulators Employing Redundant Carry-S

10

20

30

40

50

ave Format To Reduce Carry Propagation, And Related Vector Processors, Systems, And Methods」と題する米国特許出願第13/798,618号(Qualcomm 整理番号123248)にも関連する。

【0003】

[0003]__本開示の分野は、単一命令多重データ(SIMD)プロセッサと多重命令多重データ(MIMD)プロセッサを含む、ベクトルおよびスカラー演算を処理するためのベクトルプロセッサおよび関連システムに関する。

【背景技術】

【0004】

[0004]__ワイヤレスコンピューティングシステムは、デジタル情報領域において最も普及した技術の1つに急速になりつつある。技術の進歩により、ワイヤレス通信デバイスは、より小型でより強力になった。たとえば、ワイヤレスコンピューティングデバイスは一般に、小型で軽量な、ユーザが容易に持ち運べるポータブルワイヤレス電話、携帯情報端末(PDA)、およびページングデバイスを含む。より具体的には、セルラー電話およびインターネットプロトコル(IP)電話などのポータブルワイヤレス電話は、ワイヤレスネットワークを介して音声およびデータパケットを通信することができる。さらに、多くのそのようなワイヤレス通信デバイスが、他のタイプのデバイスを含む。たとえば、ワイヤレス電話機は、デジタルスチルカメラ、デジタルビデオカメラ、デジタルレコーダ、および/またはオーディオファイルプレーヤを含み得る。また、ワイヤレス電話は、インターネットにアクセスするために使用できるウェブインターフェースを含むことができる。さらに、ワイヤレス通信デバイスは、設計されたワイヤレス通信技術規格(たとえば、符号分割多元接続(CDMA)、広帯域CDMA(WCDMA(登録商標))、およびロングタームエボリューション(LTE(登録商標)))に従って高速ワイヤレス通信データを処理するための複合処理リソースを含み得る。したがって、これらのワイヤレス通信デバイスはかなりのコンピューティング能力を含む。

【0005】

[0005]__ワイヤレスコンピューティングデバイスは、より小型でより強力になるにつれて、ますますリソースの制約を受けるようになる。たとえば、画面サイズ、利用可能なメモリおよびファイルシステム空間の量、ならびに入出力能力の量は、小さいデバイスのサイズによって制限されることがある。さらに、電池サイズ、電池によって供給される電力の量、および電池の寿命も制限される。デバイスのバッテリー寿命を増すための1つのやり方は、より少ない電力を消費するプロセッサを設計することである。

【0006】

[0006]__この点において、ベクトルプロセッサを含むベースバンドプロセッサが、ワイヤレス通信デバイス用に利用され得る。ベクトルプロセッサは、ベクトル、すなわちデータアレイに働きかけるハイレベル演算を提供するベクトルアーキテクチャを有する。ベクトル処理は、ベクトル命令を一度フェッチすることと、次いで、1つのデータセットに対してベクトル命令を実行し、次いで、ベクトル内の後続要素のためにベクトル命令をフェッチし直し、復号するのとは反対に、データ要素アレイ全体にわたってベクトル命令を複数回実行することとを伴う。このプロセスにより、他の要因の中でも、各ベクトル命令は、より少ない回数しかフェッチされる必要がないので、プログラムを実行するのに必要とされるエネルギーが低減される。ベクトル命令は、同時に複数のクロック周期にわたって長いベクトルに対して動作するので、簡易な順序ベクトル命令ディスパッチを用いて、高度の並列性が達成可能である。

【0007】

[0007]__図1は、ワイヤレスコンピュータデバイスなどのコンピューティングデバイス中で利用され得る例示的なベースバンドプロセッサ10を示す。ベースバンドプロセッサ10は、特定のアプリケーション用の関数固有ベクトル処理を提供するのに各々が専用である複数の処理エンジン(PE)12を含む。この例では、6つの別個のPE12(0)

10

20

30

40

50

~ 1 2 (5) がベースバンドプロセッサ 1 0 中に設けられる。P E 1 2 (0) ~ 1 2 (5) は各々、共有メモリ 1 6 から P E 1 2 (0) ~ 1 2 (5) に与えられる固定 X ビット幅のベクトルデータ 1 4 用のベクトル処理を提供するように構成される。たとえば、ベクトルデータ 1 4 は 5 1 2 ビット幅であり得る。ベクトルデータ 1 4 は、X のより小さい倍数のビット幅のベクトルデータサンプルセット 1 8 (0) ~ 1 8 (Y) (たとえば、1 6 ビットおよび 3 2 ビットサンプルセット) 中で定義されてもよい。このようにして、P E 1 2 (0) ~ 1 2 (5) は、高度の並列性を達成するように、P E 1 2 (0) ~ 1 2 (5) に並列に与えられる複数のベクトルデータサンプルセットに対するベクトル処理を提供することが可能である。各 P E 1 2 (0) ~ 1 2 (5) は、ベクトルデータ 1 4 上で処理されるベクトル命令の結果を記憶するためのベクトルレジスタファイル (V R) を含み得る。

10

【 0 0 0 8 】

[0008]__図 1 のベースバンドプロセッサ 1 0 中の各 P E 1 2 (0) ~ 1 2 (5) は、特定のタイプの固定演算を効率的に実施するように特に設計された、特定の専用回路要素とハードウェアとを含む。たとえば、図 1 のベースバンドプロセッサ 1 0 は、別個の広帯域符号分割多元接続 (W C D M A) P E 1 2 (0)、1 2 (1) とロングタームエボリューション (L T E) P E 1 2 (4)、1 2 (5) とを含むが、これは、W C D M A および L T E は異なるタイプの特殊な演算を伴うからである。したがって、別個の W C D M A 固有 P E 1 2 (0)、1 2 (1) と L T E 固有 P E 1 2 (4)、1 2 (5) とを設けることによって、P E 1 2 (0)、1 2 (1)、1 2 (4)、1 2 (5) の各々は、高効率な演算のための、W C D M A および L T E 用の頻繁に実施される関数に固有である、特殊な専用回路要素を含むように設計され得る。この設計は、より多数の無関係な演算を、比較的無効率的にはあるがサポートするために柔軟であるように設計された、より一般的な回路要素とハードウェアとを含むスカラー処理エンジンとは対照的である。

20

【 0 0 0 9 】

[0009]__ベースバンドプロセッサ中に関数固有 P E を設けることは、頻繁に実行される特殊な演算には有利である。ただし、関数固有 P E を設けると、別個の関数固有 P E が各々、特殊な回路要素とメモリとを含むので、ベースバンドプロセッサ用に必要とされる面積および電力が増大する場合がある。

【 発明の概要 】

30

【 0 0 1 0 】

[0010]__本明細書で開示する実施形態は、マルチモードベクトル処理を提供するためのプログラム可能データ経路構成を有するベクトル処理エンジン (V P E) を含む。関連ベクトルプロセッサ、システム、および方法についても開示する。本明細書で開示する V P E は、ベクトル処理段によって実行されるベクトル命令に従ってベクトルデータを処理するように構成された少なくとも 1 つのベクトル処理段を含む。各ベクトル処理段は、実行されるベクトル命令に基づいてベクトルデータを処理するように構成された 1 つまたは複数のベクトル処理ブロックを含む。ベクトル処理ブロックの非限定的例は、乗算器および累算器である。ベクトル処理ブロックは、ベクトル処理ブロックのプログラム可能データ経路構成に基づいて、異なるタイプの命令向けの異なる関数を異なるモードで行うことが可能である。このようにして、V P E には、実行されるベクトル命令に従って、異なるタイプの関数を実行するように再プログラム可能なデータ経路構成が与えられ得る。

40

【 0 0 1 1 】

[0011]__その結果、より少ない V P E が、所望の技術特質を提供することができ、したがって、ベースバンドプロセッサ中の面積を節約するとともに、スカラー処理エンジンと比較して、より少ないレジスタ書込みおよびより速い命令実行というベクトル処理利点を依然として保有する。ベクトル処理ブロック用のデータ経路構成はまた、V P E 中で、異なる特殊なベクトル演算の実行を異なるモードでサポートするように、V P E におけるベクトル命令実行中にプログラムされ、再プログラムされ得る。

【 0 0 1 2 】

50

【0012】__本明細書で開示するマルチモードベクトル処理のためのプログラム可能データ経路構成を有するVPEは、固定関数を提供するための固定データ経路構成を含むだけであるVPEとは区別可能である。本明細書で開示するベクトル処理のためのプログラム可能データ経路構成を有するVPEは、たとえばデジタル信号プロセッサ(DSP)中に設けられるようなスカラー処理エンジンとも区別可能である。スカラー処理エンジンは、異なるタイプの非固定関数を実施するが、ベクトル命令実行中の中間結果もレジスタファイルに書き込むための柔軟な共通回路要素と論理とを利用し、そうすることによって、追加電力を消費し、ベクトル命令実行回数を増大させる。

【0013】

【0013】__この点について、一実施形態では、VPEが提供される。VPEは、ベクトルデータのマルチモードベクトル処理を行うように構成される。VPEは、複数の入力データ経路の中のある入力データ経路中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセットを与えるように構成された入力読取り段を備える。VPEは、複数のベクトル処理ブロックを備える少なくとも1つのベクトル処理段も備える。複数のベクトル処理ブロックの中の各ベクトル処理ブロックは、複数の入力データ経路の中の少なくとも1つの入力データ経路から、複数のベクトルデータ入力サンプルセットからの少なくとも1つのベクトルデータ入力サンプルセットを受信するように構成される。複数のベクトル処理ブロックの中の各ベクトル処理ブロックは、少なくとも1つのベクトル処理段によって実行されるベクトル命令に従って、各ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて、少なくとも1つのベクトル結果出力サンプルセットを与えるように、少なくとも1つのベクトルデータ入力サンプルセットを処理するようにさらに構成される。複数のベクトル処理ブロックの中の各ベクトル処理ブロックは、複数の出力データ経路の中の少なくとも1つの出力データ経路中で少なくとも1つのベクトル結果出力サンプルセットを与えるようにさらに構成される。VPEは、複数のベクトル処理ブロックの各々から少なくとも1つのベクトル結果出力サンプルセットを受信するように構成された出力処理段も備える。

【0014】

【0014】__別の実施形態では、ベクトルデータのマルチモードベクトル処理を行うように構成されたVPEが提供される。VPEは、複数の入力データ経路の中のある入力データ経路中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセットを与えるように構成された入力読取り段手段を備える。VPEは、少なくとも1つのベクトル処理段手段も備える。少なくとも1つのベクトル処理段手段は、複数のベクトル処理手段を備える。複数のベクトル処理手段の中の各ベクトル処理手段は、複数の入力データ経路の中の少なくとも1つの入力データ経路から、複数のベクトルデータ入力サンプルセットからの少なくとも1つのベクトルデータ入力サンプルセットを受信するように構成される。複数のベクトル処理手段の中の各ベクトル処理手段は、少なくとも1つのベクトル処理段によって実行されるベクトル命令に従って、各ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて、少なくとも1つのベクトル結果出力サンプルセットを与えるように、少なくとも1つのベクトルデータ入力サンプルセットを処理するようにさらに構成される。複数のベクトル処理手段の中の各ベクトル処理手段は、複数の出力データ経路の中の少なくとも1つの出力データ経路中で少なくとも1つのベクトル結果出力サンプルセットを与えるようにさらに構成される。VPEは、複数のベクトル処理手段の各々から少なくとも1つのベクトル結果出力サンプルセットを受信するように構成された出力処理手段も備える。

【0015】

【0015】__別の実施形態では、ベクトル処理エンジン(VPE)中でベクトルデータを処理する方法が提供される。この方法は、入力処理段中の複数の入力データ経路の中のある入力データ経路中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセットを与えることを備える。方法は、少なくとも1つのベクトル処理段中の複数のベクトル処理ブロックにおいて、複数のベクトルデータ入力サンプルセットを処理することも備える。

方法は、複数のベクトル処理ブロックの各々において、複数の入力データ経路の中の少なくとも1つの入力データ経路から、複数のベクトルデータ入力サンプルセットからの少なくとも1つのベクトルデータ入力サンプルセットを受信すること、少なくとも1つのベクトル処理段によって実行されるベクトル命令に従って、各ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて、少なくとも1つのベクトル結果出力サンプルセットを与えるように、少なくとも1つのベクトルデータ入力サンプルセットを処理することも備える。方法は、複数の出力データ経路の中の少なくとも1つの出力データ経路中で少なくとも1つのベクトル結果出力サンプルセットを与えることをさらに備える。方法は、出力処理段中の複数のベクトル処理ブロックの各々から少なくとも1つのベクトル結果出力サンプルセットを受信することも備える。

10

【0016】

[0016] 別の実施形態では、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、ベクトルデータメモリからベクトルデータ入力サンプルセットを受信するように構成された少なくとも1つのベクトルレジスタを備える。ベクトルプロセッサは、ベクトル結果出力サンプルセットを記憶するように構成された少なくとも1つのベクトル結果レジスタも備える。ベクトルプロセッサは、複数のベクトル処理段を備える少なくとも1つのVPEも備える。複数のベクトル処理段は各々、少なくとも1つのベクトルレジスタから少なくとも1つのベクトルデータ入力サンプルセットを受信し、少なくとも1つのベクトルレジスタからの少なくとも1つのベクトルデータ入力サンプルセットを、各ベクトル処理段によって実行されるベクトル命令に従って、ベクトル処理段の各々のための少なくとも1つのデータ経路構成に基づいて、少なくとも1つのベクトル結果出力サンプルセットに処理し、少なくとも1つのベクトル結果出力サンプルセットを、少なくとも1つのベクトル結果レジスタに記憶されるように、少なくとも1つのベクトルレジスタに与えるように構成される。

20

【図面の簡単な説明】

【0017】

【図1】 [0017] 固有アプリケーション向けの関数固有ベクトル処理を提供するのに各々が専用である、複数のベクトル処理エンジン(VPE)を含む例示的なベクトルプロセッサの配線図。

【図2】 [0018] VPE中に設けられた共通回路要素およびハードウェアが、別個のVPEを設ける必要なく、複数のアプリケーションまたは技術のために特定のタイプのベクトル演算を高効率に実施するための複数のモードでプログラムされ得るように、プログラム可能データ経路構成を有する共通VPEを含む例示的なベクトルプロセッサの配線図。

30

【図3】 [0019] 図2のVPEに設けられる例示的なベクトル処理段の配線図であって、ベクトル処理段のうちのいくつかは、プログラム可能データ経路構成を有する例示的なベクトル処理ブロックを含む。

【図4A】 [0020] 図2および図3の例示的なベクトルプロセッサ中に含まれるプログラム可能データ経路構成を有する少なくとも1つのベクトル処理ブロックの例示的なベクトル処理を示すフローチャート。

【図4B】 [0021] プログラム可能データ経路構成を各々が有するとともに、図2および図3の例示的なベクトルプロセッサにおける異なるベクトル処理段において提供される乗算器ブロックおよび累算器ブロックの例示的なベクトル処理を示すフローチャート。

40

【図5】 [0022] 図2および図3のVPEのベクトル処理段に設けられる複数の乗算器ブロックのより詳細な配線図であって、複数の乗算器ブロックは各々、複数の乗算器ブロックが、特定の異なるタイプのベクトル乗算演算を実施するための複数のモードでプログラムされ得るように、プログラム可能データ経路構成を有する。

【図6】 [0023] 8ビット×8ビットベクトルデータ入力サンプルセットおよび16ビット×16ビットベクトルデータ入力サンプルセットについての乗算演算を行うようにプログラムされることが可能なプログラム可能データ経路構成を有する、図5の複数の乗算器ブロックの中のある乗算器ブロックの内部構成要素の配線図。

50

【図 7】[0024] 図 2 および図 3 の V P E 中の乗算器ブロックおよび累算器ブロックの一般化配線図であって、累算器ブロックは、桁上げ伝搬を低減するのに冗長桁上げ保存フォーマットを利用する桁上げ保存累算器構造を利用する。

【図 8】[0025] 図 2 および図 3 の V P E 中に設けられる、図 7 の累算器ブロックの例示的な内部構成要素の詳細な配線図であって、累算器ブロックが、冗長桁上げ保存フォーマットで、特定の異なるタイプのベクトル累算演算を実施するための複数のモードでプログラムされ得るように、累算器ブロックはプログラム可能データ経路構成を有する。

【図 9 A】[0026] 二重 24 ビット累算器モードを提供するためにプログラムされたデータ経路構成を有する図 8 の累算器ブロックの配線図。

【図 9 B】[0027] 単一 40 ビット累算器モードを提供するためにプログラムされたデータ経路構成を有する図 8 の累算器ブロックの配線図。

【図 9 C】[0028] 16 ビット桁上げ保存加算器と、それに続く単一 24 ビット累算器モードを提供するためにプログラムされたデータ経路構成を有する図 8 の累算器ブロックの配線図。

【図 10】[0029] 本明細書で開示する実施形態による、プログラム可能データ経路構成を有する V P E を含むベクトルプロセッサを含むことができ、したがって V P E 中の共通回路要素およびハードウェアが、別個の V P E を提供する必要なく、複数のアプリケーションまたは技術のために特定のタイプのベクトル演算を効率的に実施するように設計された専用回路要素として作用するようにプログラムされ得る、例示的なプロセッサベースのシステムのブロック図。

【発明を実施するための形態】

【0018】

[0030] ここで図面を参照すると、本開示のいくつかの例示的な実施形態が記載される。「例示的な」という言葉は、本明細書中で、「例、事例、または例示として働くこと」を意味するように使用される。本明細書で「例示的な」として記載される任意の実施形態は、必ずしも、他の実施形態よりも好ましいまたは有利であると解釈されるべきでない。

【0019】

[0031] 本明細書で開示する実施形態は、マルチモードベクトル処理を提供するためのプログラム可能データ経路構成を有するベクトル処理エンジン (V P E) を含む。関連ベクトルプロセッサ、システム、および方法についても開示する。本明細書で開示する V P E は、ベクトル処理段によって実行されるベクトル命令に従ってベクトルデータを処理するように構成された少なくとも 1 つのベクトル処理段を含む。各ベクトル処理段は、実行されるベクトル命令に基づいてベクトルデータを処理するように構成された 1 つまたは複数のベクトル処理ブロックを含む。ベクトル処理ブロックの非限定的例は、乗算器および累算器である。ベクトル処理ブロックは、ベクトル処理ブロックのデータ経路構成に基づいて、異なるタイプの命令向けの異なるベクトル演算を異なるモードで行うことが可能である。ベクトル処理ブロックのデータ経路構成は、実行される特定のベクトル命令に従って、異なるモードで特殊なベクトル処理演算を行うためにベクトルデータを異なるやり方で処理するようにプログラム可能であり、再プログラム可能であり（すなわち、再配線され）得る。

【0020】

[0032] このようにして、V P E には、実行されるベクトル命令に従って異なるタイプのベクトル演算を実行するように再プログラム可能なデータ経路構成が与えられ得る。その結果、より少ない V P E が、ベクトルプロセッサ中で所望のベクトル処理演算を行うことができ、したがって、ベクトルプロセッサにおける面積を節約するとともに、スカラー処理エンジンに勝る、より少ないレジスタ書込みおよびより速いベクトル命令実行回数というベクトル処理利点を依然として保有する。ベクトル処理ブロック用のデータ経路構成はまた、V P E 中で、異なる特殊なベクトル演算の実行を異なるモードでサポートするように、V P E におけるベクトル命令実行中にプログラムされ、再プログラムされ得る。

【0021】

【0033】__本明細書で開示するマルチモードベクトル処理のためのプログラム可能データ経路構成を有するVPEは、固定関数を提供するための固定データ経路構成を含むだけであるVPEとは区別可能である。本明細書で開示するベクトル処理のためのプログラム可能データ経路構成を有するVPEは、たとえばデジタル信号プロセッサ(DSP)中に設けられるようなスカラー処理エンジンとも区別可能である。スカラー処理エンジンは、異なるタイプの非固定関数を実施するが、ベクトル命令実行中の中間結果もレジスタファイルに書き込むための柔軟な共通回路要素と論理とを利用し、そうすることによって、追加電力を消費し、ベクトル命令実行回数を増大させる。

【0022】

【0034】__この点において、図2は、ベクトル処理エンジン(VPE)22とも呼ばれる、例示的なベクトル処理ユニット22を含むベースバンドプロセッサ20の配線図である。ベースバンドプロセッサ20およびそのVPE22は、半導体ダイ24中に設けられ得る。図3で始まる、後でより詳しく論じる本実施形態では、ベースバンドプロセッサ20は、プログラム可能データ経路構成を有する共通VPE22を含む。このように、VPE22は、ベースバンドプロセッサ20中に別個のVPEを設ける必要なく、異なる演算モードで異なる特定のタイプのベクトル演算を行うようにプログラムされ、再プログラムされ得る共通回路要素とハードウェアとを含む。VPE22はまた、ベースバンドプロセッサ20中に別個のVPEを設ける必要なく、複数のアプリケーションまたは技術のために、一般的な算術演算を高効率的に実施するためのベクトル算術モードでプログラムされ得る。

【0023】

【0035】__図3で始まる、ベクトルマルチモード処理のためにVPE22中に設けられるプログラム可能データ経路構成について論じる前に、図2のベースバンドプロセッサ20の構成要素について、最初に記載する。この非限定的例におけるベースバンドプロセッサ20は、512ビットベクトルプロセッサである。ベースバンドプロセッサ20は、VPE22がベースバンドプロセッサ20中でベクトル処理を行うのをサポートするための追加構成要素を、VPE22に加えて含む。ベースバンドプロセッサ20は、ベクトルユニットデータメモリ(LMEM)32からベクトルデータ30を受信し、記憶するように構成されたベクトルレジスタ28を含む。たとえば、ベクトルデータ30はXビット幅であり、「X」は、設計選択に従って定義される(たとえば、512ビット)。ベクトルデータ30は、ベクトルデータサンプルセット34に分割され得る。たとえば、ベクトルデータ30は、256ビット幅であってよく、より小さいベクトルデータサンプルセット34(Y)~34(0)を備えることができ、ここでベクトルデータサンプルセット34(Y)~34(0)の一部は16ビット幅であり、ベクトルデータサンプルセット34(Y)~34(0)のうちのそれ以外は32ビット幅である。VPE22は、高度の並列性を達成するように、VPE22に並列に与えられるいくつかの選ばれた乗算ベクトルデータサンプルセット34(Y)~34(0)に対して、ベクトル処理を行うことが可能である。ベクトルレジスタ28は、VPE22がベクトルデータ30を処理するとき生成される結果を記憶するようにも構成される。いくつかの実施形態では、VPE22は、より速いベクトル命令実行時間を可能にするようにレジスタ書き込みを低減するために、中間ベクトル処理結果をベクトルレジスタ28に記憶しないように構成される。この構成は、スカラー処理DSPなど、中間結果をレジスタに記憶するスカラー処理エンジンによって実行されるスカラー命令とは反対である。

【0024】

【0036】__図2のベースバンドプロセッサ20は、ベクトル命令の条件付き実行において使用するために、条件をVPE22に与えるように、およびアップデートされた条件をベクトル命令実行の結果として記憶するように構成された条件レジスタ36も含む。ベースバンドプロセッサ20はまた、累算レジスタ38と、グローバルレジスタ40と、アドレスレジスタ42とを含む。累算レジスタ38は、VPE22によって、累算された結果を、ベクトルデータ30に対していくつかの特殊な演算を実行した結果として記憶するのに

使われるように構成される。グローバルレジスタ40は、VPE22によってサポートされるいくつかのベクトル命令用のスカラーオペランドを記憶するように構成される。アドレスレジスタ42は、ベクトルユニットデータメモリ32からベクトルデータ30を取り出し、ベクトル処理結果をベクトルユニットデータメモリ32に記憶するために、ベクトルロードによってアドレス指定可能なアドレスを記憶し、VPE22によってサポートされる命令を記憶するように構成される。

【0025】

[0037] 引き続き図2を参照すると、本実施形態におけるベースバンドプロセッサ20は、VPE22によって行われるベクトル処理に加え、ベースバンドプロセッサ20中でスカラー処理を行うためのスカラープロセッサ44（「整数ユニット」とも呼ばれる）も含む。高効率な演算のために実行される命令のタイプに基づいて、ベクトル演算とスカラー命令演算の両方をサポートするように構成されたCPUを設けることが所望され得る。本実施形態では、スカラープロセッサ44は、非限定的例として、32ビット縮小命令セットコンピューティング(RISC)スカラープロセッサである。スカラープロセッサ44は、この例では、スカラー命令処理をサポートするための論理演算ユニット(ALU)46を含む。ベースバンドプロセッサ20は、プログラムメモリ50から命令をフェッチし、フェッチした命令を復号し、フェッチした命令を、命令タイプに基づいて、スカラープロセッサ44に、またはベクトルデータ経路49を通してVPE22に向けるように構成された命令ディスパッチ回路48を含む。スカラープロセッサ44は、スカラー命令を実行するときにスカラープロセッサ44によって使用するための汎用レジスタ52を含む。スカラー命令実行のためにスカラープロセッサ44によってアクセスするために、メインメモリから汎用レジスタ52にデータを与えるように、整数ユニットデータメモリ(DMEM)54がベースバンドプロセッサ20中に含まれる。DMEM54は、非限定的例として、キャッシュメモリであり得る。ベースバンドプロセッサ20は、メモリコントローラデータ経路59を通る、メインメモリへのアクセスを求めるベクトル命令をスカラープロセッサ44が実行しているとき、汎用レジスタ52からメモリアドレスを受信するように構成されたメモリコントローラレジスタ58を含むメモリコントローラ56も含む。

【0026】

[0038] 図2のベースバンドプロセッサ20の例示的な構成要素について記載してきたので、共通回路要素およびハードウェアをもつ、複数の動作モードを提供するVPE22およびそのプログラム可能データ経路構成に関するさらなる詳細についてここで論じる。この点において、図3は、図2のVPE22の例示的な配線図を示す。図3に示すように、および図4～図8Cにおいて以下でより詳細に説明するように、VPE22は、プログラム可能データ経路構成で構成され得る例示的なベクトル処理ブロックを有する複数の例示的なベクトル処理段60を含む。後でより詳しく論じるように、ベクトル処理ブロック中に設けられたプログラム可能データ経路構成により、特定の回路およびハードウェアが、図2のベクトルユニットデータメモリ32から受信されたベクトルデータ30に対する異なる特定のベクトル処理演算の実施をサポートするようにプログラムされ、再プログラムされる。

【0027】

[0039] たとえば、いくつかのベクトル処理演算は一般に、ベクトルデータ30の乗算、続いて、乗算されたベクトルデータ結果の累算を必要とし得る。そのようなベクトル処理の非限定的例には、ワイヤレス通信アルゴリズムのための高速フーリエ変換(FFT)演算を実施するために一般に使われる、フィルタリング演算、相関演算、ならびに基数2および基数4のバタフライ演算があり、ここで一連の並列乗算が、続いて乗算結果の一連の並列累算が行われる。やはり図7～図9Cに関連して後でより詳しく論じるように、図2のVPE22は、桁上げ保存累算器において冗長桁上げ保存フォーマットを提供するための桁上げ保存累算器をもつ融合乗算器というオプションも有する。桁上げ保存累算器中で冗長桁上げ保存フォーマットを提供することにより、桁上げ伝搬経路と、各累算ステップ中の桁上げ伝搬加算演算とを提供する必要をなくすることができる。

【 0 0 2 8 】

[0040]__この点において、さらに図 3 を参照して、V P E 2 2 の M 0 乗算ベクトル処理段 6 0 (1) について最初に記載する。M 0 乗算ベクトル処理段 6 0 (1) は、プログラム可能データ経路構成を各々が有する、任意の所望の数の乗算器ブロック 6 2 (A) ~ 6 2 (0) の形の複数のベクトル処理ブロックを含む第 2 のベクトル処理段である。乗算器ブロック 6 2 (A) ~ 6 2 (0) は、V P E 2 2 中でベクトル乗算演算を実施するように設けられる。複数の乗算器ブロック 6 2 (A) ~ 6 2 (0) は、最大 1 2 個の乗算ベクトルデータサンプルセット 3 4 (Y) ~ 3 4 (0) の乗算を行うために、M 0 乗算ベクトル処理段 6 0 (1) 中に互いと並列に配置される。本実施形態では、「A」は 3 に等しく、この例では、M 0 乗算ベクトル処理段 6 0 (1) に 4 つの乗算器ブロック 6 2 (3) ~ 6 2 (0) が含まれることを意味する。乗算ベクトルデータサンプルセット 3 4 (Y) ~ 3 4 (0) は、V P E 2 2 中の第 1 のベクトル処理段 6 0 (0) である入力読取り (R R) ベクトル処理段に設けられた複数のラッチ 6 4 (Y) ~ 6 4 (0) へのベクトル処理のために V P E 2 2 中にロードされる。本実施形態では、V P E 2 2 中に 1 2 個のラッチ 6 4 (1 1) ~ 6 4 (0) があり、本実施形態では「Y」が 1 1 に等しいことを意味する。ラッチ 6 4 (1 1) ~ 6 4 (0) は、ベクトルレジスタ 2 8 (図 2 参照) から取り出された乗算ベクトルデータサンプルセット 3 4 (1 1) ~ 3 4 (0) をベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) としてラッチするように構成される。この例では、各ラッチ 6 4 (1 1) ~ 6 4 (0) は 8 ビット幅である。ラッチ 6 4 (1 1) ~ 6 4 (0) は各々、乗算ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) を、総計 9 6 ビット幅のベクトルデータ 3 0 (すなわち、各々、1 2 ラッチ × 8 ビット) を求めてラッチするようにそれぞれ構成される。

10

20

【 0 0 2 9 】

[0041]__引き続き図 3 を参照すると、複数の乗算器ブロック 6 2 (3) ~ 6 2 (0) は、ベクトル乗算演算を行うために、ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) のいくつかの組合せを受信することが可能なように構成され、ここにおいて「Y」は、この例では 1 1 に等しい。乗算ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) は、V P E 2 2 の設計に従って、複数の入力データ経路 A 3 ~ A 0、B 3 ~ B 0、および C 3 ~ C 0 中で与えられる。ベクトルデータ入力サンプルセット 6 6 (3) ~ 6 6 (0) は、図 3 に示すように入力データ経路 C 3 ~ C 0 に対応する。ベクトルデータ入力サンプルセット 6 6 (7) ~ 6 6 (4) は、図 3 に示すように入力データ経路 B 3 ~ B 0 に対応する。ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (8) は、図 3 に示すように入力データ経路 A 3 ~ A 0 に対応する。複数の乗算器ブロック 6 2 (3) ~ 6 2 (0) は、ベクトル乗算演算を行うために、それぞれ、受信ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) を、複数の乗算器ブロック 6 2 (3) ~ 6 2 (0) へと設けられた入力データ経路 A 3 ~ A 0、B 3 ~ B 0、C 3 ~ C 0 に従って処理するように構成される。

30

【 0 0 3 0 】

[0042]__図 4 および図 5 に関連して後でより詳しく論じるように、図 3 の乗算器ブロック 6 2 (3) ~ 6 2 (0) 中に設けられたプログラム可能内部データ経路 6 7 (3) ~ 6 7 (0) は、異なるデータ経路構成を有するようにプログラムされ得る。これらの異なるデータ経路構成は、各乗算器ブロック 6 2 (3) ~ 6 2 (0) へと設けられた特定の入力データ経路 A 3 ~ A 0、B 3 ~ B 0、C 3 ~ C 0 に従って乗算器ブロック 6 2 (3) ~ 6 2 (0) に与えられる特定の受信ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) の異なる組合せおよび / または異なるビット長の乗算を可能にする。この点において、複数の乗算器ブロック 6 2 (3) ~ 6 2 (0) は、ベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) を、ベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) の特定の組合せを掛け合わせた乗算結果を備えるベクトル結果出力サンプルセットとして与える。

40

【 0 0 3 1 】

50

[0043] __たとえば、乗算器ブロック62(3)~62(0)のプログラム可能内部データ経路67(3)~67(0)は、図2のベースバンドプロセッサ20の命令ディスパッチ48中のベクトル命令デコーダから与えられる設定に従ってプログラムされ得る。本実施形態では、乗算器ブロック62(3)~62(0)の4つのプログラム可能内部データ経路67(3)~67(0)がある。ベクトル命令は、VPE22によって実施されるべき特定タイプの演算を指定する。したがって、VPE22は、同じ共通回路要素を用いて高効率的に異なるタイプのベクトル乗算演算を行うために、乗算器ブロック62(3)~62(0)のプログラム可能内部データ経路67(3)~67(0)を構成するようにプログラムされ、再プログラムされ得る。たとえば、VPE22は、乗算器ブロック62(3)~62(0)のプログラム可能内部データ経路67(3)~67(0)を、命令ディスパッチ48中の命令パイプライン中のベクトル命令の復号に従って、実行される各ベクトル命令ごとに周期かけるクロック周期ベースで構成し、再構成するようにプログラムされ得る。したがって、VPE22中のM0乗算ベクトル処理段60(1)が、ベクトルデータ入力サンプルセット66を毎クロック周期に処理するように構成されている場合、その結果、乗算器ブロック62(3)~62(0)は、命令ディスパッチ48中の命令パイプライン中のベクトル命令の復号に従って、毎クロック周期でベクトル乗算演算を実施する。

10

【0032】

[0044] __乗算器ブロック62は、実数乗算と複素乗算とを実施するようにプログラムされ得る。引き続き図3を参照すると、あるベクトル処理ブロックデータ経路構成において、乗算器ブロック62が、2つの8ビットベクトルデータ入力サンプルセット66を掛け合わせるように構成され得る。1つの乗算ブロックデータ経路構成では、乗算器ブロック62が、2つの16ビットベクトルデータ入力サンプルセット66を掛け合わせるように構成されてよく、これらのセットは、8ビットベクトルデータ入力サンプルセット66の第1のペアに、8ビットベクトルデータ入力サンプルセット66の第2のペアを乗算して形成される。これについては、図6に示し、以下でより詳細に説明する。やはり、乗算器ブロック62(3)~62(0)中でプログラム可能データ経路構成を提供することにより、乗算器ブロック62(3)~62(0)が、VPE22における面積を削減するとともに、可能な、所望のベクトル処理演算を実施するために、ベースバンドプロセッサ20中により少ないVPE22が設けられるようにするために、異なるタイプの乗算演算を実施するように構成され、再構成され得るという柔軟性がもたらされる。

20

30

【0033】

[0045] __図3に参照を戻すと、複数の乗算器ブロック62(3)~62(0)は、プログラム可能出力データ経路70(3)~70(0)中のベクトル乗算出力サンプルセット68(3)~68(0)を、次のベクトル処理段60または出力処理段のいずれかに与えるように構成される。ベクトル乗算出力サンプルセット68(3)~68(0)は、複数の乗算器ブロック62(3)~62(0)によって実行されるベクトル命令に基づき、プログラムされた構成に従って、プログラム可能出力データ経路70(3)~70(0)中で与えられる。この例では、プログラム可能出力データ経路70(3)~70(0)中のベクトル乗算出力サンプルセット68(3)~68(0)は、以下で論じるように、累算のためにM1累算ベクトル処理段60(2)に与えられる。VPE22のこの特定の設計では、複数の乗算器ブロック62(3)~62(0)、続いて、ベクトルデータ入力の乗算、それに続く乗算結果の累算を要請する特殊なベクトル命令をサポートするための累算器を設けることが所望される。たとえば、FFT演算を行うのに一般に使われる、基数2および基数4のバタフライ演算は、一連の乗算演算、それに続く乗算結果の累算を含む。ただし、VPE22中に設けられるベクトル処理ブロックのこれらの組合せは例示であり、限定的でないことに留意されたい。プログラム可能データ経路構成を有するVPEは、ベクトル処理ブロックを有する1つまたは他のどの数のベクトル処理段を含むように構成されてもよい。ベクトル処理ブロックは、設計およびVPEによってサポートされるように設計された特定のベクトル命令に従って、どのタイプの演算を実施するように設けられ

40

50

てもよい。

【 0 0 3 4 】

[0046]__引き続き図 3 を参照すると、本実施形態では、ベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) は、M 1 累算ベクトル処理段 6 0 (2) である次のベクトル処理段中に設けられた複数の累算器ブロック 7 2 (3) ~ 7 2 (0) に与えられる。複数の累算器ブロック 7 2 (A) ~ 7 2 (0) の中の各累算器ブロックは、2 つの累算器 7 2 (X) (1) および 7 2 (X) (0) (すなわち、7 2 (3) (1)、7 2 (3) (0)、7 2 (2) (1)、7 2 (2) (0)、7 2 (1) (1)、7 2 (1) (0)、および 7 2 (0) (1)、7 2 (0) (0)) を含む。複数の累算器ブロック 7 2 (3) ~ 7 2 (0) は、ベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) の結果を累算する。図 7 ~ 図 9 C に関連して後でより詳しく論じるように、複数の累算器ブロック 7 2 (3) ~ 7 2 (0) は桁上げ保存累算器として設けられてよく、ここにおいて桁上げ積は本質的に、累算演算が完了されるまで、累算プロセス中は保存され、伝搬されない。複数の累算器ブロック 7 2 (3) ~ 7 2 (0) は、複数の累算器ブロック 7 2 (3) ~ 7 2 (0) 中で冗長桁上げ保存フォーマットを提供するために、図 4 および図 5 の複数の乗算器ブロック 6 2 (3) ~ 6 2 (0) と融合されるという選択肢も有する。複数の累算器ブロック 7 2 (3) ~ 7 2 (0) 中で冗長桁上げ保存フォーマットを提供することにより、複数の累算器ブロック 7 2 (3) ~ 7 2 (0) の各累算ステップ中に、桁上げ伝搬経路と桁上げ伝搬加算演算とを提供する必要をなくすることができる。M 1 累算ベクトル処理段 6 0 (2) およびその複数の累算器ブロック 7 2 (3) ~ 7 2 (0) を、ここで図 3 を参照して提起する。

【 0 0 3 5 】

[0047]__図 3 を参照すると、M 1 累算ベクトル処理段 6 0 (2) 中の複数の累算器ブロック 7 2 (3) ~ 7 2 (0) は、累算器出力サンプルセット 7 6 (3) ~ 7 6 (0) (すなわち、7 6 (3) (1)、7 6 (3) (0)、7 6 (2) (1)、7 6 (2) (0)、7 6 (1) (1)、7 6 (1) (0)、および 7 6 (0) (1)、7 6 (0) (0)) を次のベクトル処理段 6 0 または出力処理段のいずれかにおいて与えるために、プログラム可能出力データ経路 7 4 (3) ~ 7 4 (0) (すなわち、7 4 (3) (1)、7 4 (3) (0)、7 4 (2) (1)、7 4 (2) (0)、7 4 (1) (1)、7 4 (1) (0)、および 7 4 (0) (1)、7 4 (0) (0)) 中のベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) を、プログラム可能出力データ経路構成に従って累算するように構成される。この例では、累算器出力サンプルセット 7 6 (3) ~ 7 6 (0) は、A L U 処理段 6 0 (3) である出力処理段に与えられる。たとえば、後でより詳しく論じるように、累算器出力サンプルセット 7 6 (3) ~ 7 6 (0) は、非限定的例として、図 2 のベースバンドプロセッサ 2 0 中のスカラープロセッサ 4 4 中の A L U 4 6 にも与えられてよい。たとえば、A L U 4 6 は、より一般的な処理演算において使われるべき、V P E 2 2 によって実行される特殊なベクトル命令に従って、累算器出力サンプルセット 7 6 (3) ~ 7 6 (0) をとることができる。

【 0 0 3 6 】

[0048]__図 3 に参照を戻すと、累算器ブロック 7 2 (3) ~ 7 2 (0) のプログラム可能入力データ経路 7 8 (3) ~ 7 8 (0) および / またはプログラム可能内部データ経路 8 0 (3) ~ 8 0 (0) は、乗算器ブロック 6 2 (3) ~ 6 2 (0) から累算器ブロック 7 2 (3) ~ 7 2 (0) に与えられた、異なる組合せおよび / またはビット長のベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) を受信するように再構成されるようにプログラムされ得る。各累算器ブロック 7 2 は、2 つの累算器 7 2 (X) (1)、7 2 (X) (0) からなるので、プログラム可能入力データ経路 7 8 (A) ~ 7 8 (0) は、7 8 (3) (1)、7 8 (3) (0)、7 8 (2) (1)、7 8 (2) (0)、7 8 (1) (1)、7 8 (1) (0)、および 7 8 (0) (1)、7 8 (0) (0) として図 3 に示されている。同様に、プログラム可能内部データ経路 8 0 (3) ~ 8 0 (A) は、8 0 (3) (1)、8 0 (3) (0)、8 0 (2) (1)、8 0 (2) (0)、8 0 (1) (1)

、80(1)(0)、80(0)(1)、80(0)(0)として図3に示されている。累算器ブロック72(3)~72(0)中にプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)を設けることについては、図8~図9Cに関連して後でより詳しく論じる。このようにして、累算器ブロック72(3)~72(0)のプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)に従って、累算器ブロック72(3)~72(0)は、累算されたベクトル乗算出力サンプルセット68(3)~68(0)のプログラムされた組合せに従って、累算器出力サンプルセット76(3)~76(0)を与えることができる。やはり、これにより、累算器ブロック72(3)~72(0)が、VPE22中での面積を削減するとともに、可能性としては、所望のベクトル処理演算を実施するのに、より少ないVPE22がベースバンドプロセッサ20中に設けられるようにするために、プログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)のプログラミングに基づいて異なるタイプの累算演算を実施するように構成され、再構成され得るという柔軟性が与えられる。

【0037】

[0049]__たとえば、ある累算器モード構成では、2つの累算器ブロック72のプログラム可能入力データ経路78および/またはプログラム可能内部データ経路80は、非限定的例として、単一40ビット累算器を提供するようにプログラムされ得る。これについては、図9Aに示し、以下でより詳細に説明する。別の累算器モード構成では、2つの累算器ブロック72のプログラム可能入力データ経路78および/またはプログラム可能内部データ経路80は、非限定的例として、二重24ビット累算器を提供するようにプログラムされ得る。これについては、図9Bに示し、以下でより詳細に説明する。別の累算器モード構成では、2つの累算器ブロック72のプログラム可能入力データ経路78および/またはプログラム可能内部データ経路80は、16ビット桁上げ保存加算器、それに続く単一24ビット累算器を提供するようにプログラムされ得る。これについては、図9Cに示し、以下でより詳細に説明する。乗算演算と累算演算の特定の異なる組合せも、乗算器ブロック62(3)~62(0)および累算器ブロック72(3)~72(0)(たとえば、16ビット累算を用いる16ビット複素乗算、および16ビット累算を用いる32ビット複素乗算)のプログラミングに従って、VPE22によってサポートされ得る。

【0038】

[0050]__累算器ブロック72(3)~72(0)のプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)は、図2のベースバンドプロセッサ20の命令ディスパッチ48中のベクトル命令デコードから与えられた設定に従ってプログラムされ得る。ベクトル命令は、VPE22によって実施されるべき特定タイプの演算を指定する。したがって、VPE22は、累算器ブロック72(3)~72(0)のプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)を、命令ディスパッチ48中の命令パイプライン中のベクトル命令の復号に従って実行される各ベクトル命令用にプログラムし直すように構成され得る。ベクトル命令は、VPE22の1つまたは複数のクロック周期にわたって実行し得る。また、この例では、VPE22は、累算器ブロック72(3)~72(0)のプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)を、ベクトル命令の各クロック周期ごとに、クロック周期単位でプログラムし直すように構成され得る。したがって、たとえば、VPE22中のM1累算ベクトル処理段60(2)によって実行されるベクトル命令が、ベクトル乗算出力サンプルセット68(3)~68(0)を毎クロック周期に処理する場合、その結果、累算器ブロック72(3)~72(0)のプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)は、ベクトル命令の実行中、各クロック周期に再構成され得る。

【 0 0 3 9 】

[0051] 図 4 A および図 4 B は、例示的なベクトル処理のさらなる説明を与えるための、図 2 および図 3 の V P E 2 2 中の乗算器ブロック 6 2 (A) ~ 6 2 (0) および累算器ブロック 7 2 (A) (1) ~ 7 2 (0) (0) の例示的なベクトル処理を示すフローチャートである。図 4 A は、図 2 および図 3 の例示的な V P E 中に含まれるプログラム可能データ経路構成を有する、乗算器ブロック 6 2 (A) ~ 6 2 (0)、累算器ブロック 7 2 (A) (1) ~ 7 2 (0) (0)、または両方のいずれであってもよい一般化ベクトル処理ブロックの例示的なベクトル処理を示すフローチャートである。図 4 B は、各々がプログラム可能データ経路構成を有するとともに図 2 および図 3 の例示的な V P E 2 2 中の異なるベクトル処理段中に設けられる、乗算器ブロック 6 2 (A) ~ 6 2 (0) および累算器

10

【 0 0 4 0 】

[0052] この点において、図 4 A に示すように、V P E 2 2 のプロセスは、入力処理段 6 0 (0) 中の複数の入力データ経路の中のある入力データ経路 (A 3 ~ C 0) 中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセット 3 4 (Y) ~ 3 4 (0) を受信することを含む (ブロック 8 1)。ベクトル処理は次に、ベクトル処理ブロック 6 2 (A) ~ 6 2 (0) および / または 7 2 (A) (1) ~ 7 2 (0) (0) 中の複数の入力データ経路 A 3 ~ C 0 から、ベクトルデータ入力サンプルセット 3 4 (Y) ~ 3 4 (0) を受信することを備える (ブロック 8 3)。ベクトル処理は次に、ベクトル処理段 6 0 (1)、6 0 (2) によって実行されるベクトル命令に従って、ベクトル処理ブロック 6 2 (A) ~ 6 2 (0)、7 2 (A) (1) ~ 7 2 (0) (0) 用のプログラム可能データ経路構成 6 7 (A) ~ 6 7 (0)、7 0 (3) ~ 7 0 (0)、7 8 (A) (1) ~ 7 8 (0) (0)、8 0 (A) (1) ~ 8 0 (0) (0)、7 4 (A) (1) ~ 7 4 (0) (0) に基づいて、ベクトル結果出力サンプルセット 6 8 (A) ~ 6 8 (0)、7 6 (A) (1) ~ 7 6 (0) (0) を与えるように、ベクトルデータ入力サンプルセット 3 4 (Y) ~ 3 4 (0) を処理することを含む (ブロック 8 5)。ベクトル処理は次に、出力データ経路 7 0 (A) ~ 7 0 (0)、7 4 (A) (1) ~ 7 4 (0) (0) 中でベクトル結果出力サンプルセット 6 8 (A) ~ 6 8 (0)、7 6 (A) (1) ~ 7 6 (0) (0) を与えることを含む (ブロック 8 7)。ベクトル処理は次に、出力処理段 6 0 (3) において、ベクトル処理段 6 0 (1)、6 0 (2) からベクトル結果出力サンプルセット 6 8 (A) ~ 6 8 (0)、7 6 (A) (1) ~ 7 6 (0) (0) を受信することを含む (ブロック 8 9)。

20

30

【 0 0 4 1 】

[0053] 図 4 A に関連して上述したベクトル処理における各処理段 6 0 (0) ~ 6 0 (3) は、並列化ベクトル処理のために同時に起こり、ここにおいてベクトル処理ブロック 6 2 (A) ~ 6 2 (0)、7 2 (A) (1) ~ 7 2 (0) (0) のプログラム可能データ経路構成 6 7 (A) ~ 6 7 (0)、7 0 (3) ~ 7 0 (0)、7 8 (A) (1) ~ 7 8 (0) (0)、8 0 (A) (1) ~ 8 0 (0) (0)、7 4 (A) (1) ~ 7 4 (0) (0) は、各クロック周期と同程度に頻繁に再プログラムされ得ることに留意されたい。上述したように、これにより、ベクトル処理ブロック 6 2 (A) ~ 6 2 (0)、7 2 (A) (1) ~ 7 2 (0) (0) は、異なるベクトル命令用の異なる演算を、効率的に、および共通ベクトル処理ブロック 6 2 (A) ~ 6 2 (0)、7 2 (A) (1) ~ 7 2 (0) (0) の使用により実施することができるようになる。

40

【 0 0 4 2 】

[0054] 図 4 B は、乗算演算、それに続く累算演算を伴う、ベクトル命令のための、図 3 の V P E 2 2 中の乗算器ブロック 6 2 (A) ~ 6 2 (0) および累算器ブロック 7 2 (A) (1) ~ 7 2 (0) (0) の例示的なベクトル処理を示すフローチャートである。たとえば、FFT ベクトル演算は、乗算演算と、それに続く累算演算を伴う。図 4 B のフローチャートは、図 4 A で上述した、V P E 2 2 の例示的な一般化ベクトル処理のさらなる

50

例示的な詳細を与える。この点において、ベクトル処理は、入力処理段 6 0 (0) 中の複数の入力データ経路の中のある入力データ経路 (A 3 ~ C 0) 中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセット 3 4 (Y) ~ 3 4 (0) を受信することを含む (ブロック 9 3) 。ベクトル処理は次いで、複数の乗算器ブロック 6 2 (A) ~ 6 2 (0) 中の複数の入力データ経路 A 3 ~ C 0 からベクトルデータ入力サンプルセット 3 4 (Y) ~ 3 4 (0) を受信することを含む (ブロック 9 5) 。ベクトル処理は次いで、ベクトル処理段 6 0 (1) によって実行されるベクトル命令に従って、乗算器ブロック 6 2 (A) ~ 6 2 (0) 用のプログラム可能データ経路構成 6 7 (A) ~ 6 7 (0) 、 7 0 (3) ~ 7 0 (0) に基づき、複数の乗算出力データ経路 7 0 (A) ~ 7 0 (0) 中の乗算出力データ経路 7 0 (A) ~ 7 0 (0) 中で乗算ベクトル結果出力サンプルセット 6 8 (A) ~ 6 8 (0) を与えるように、ベクトルデータ入力サンプルセット 3 4 (Y) ~ 3 4 (0) を乗算することを含む (ブロック 9 7) 。ベクトル処理は次に、複数の累算器ブロック 7 2 (A) (1) ~ 7 2 (0) (0) 中の複数の乗算出力データ経路 7 0 (A) ~ 7 0 (0) から乗算ベクトル結果出力サンプルセット 6 8 (A) ~ 6 8 (0) を受信することを含む (ブロック 9 9) 。ベクトル処理は次に、第 2 のベクトル処理段 6 0 (2) によって実行されるベクトル命令に従って、累算器ブロック 7 2 (A) (1) ~ 7 2 (0) (0) 用のプログラム可能データ経路 7 8 (A) (1) ~ 7 8 (0) (0) 、 8 0 (A) (1) ~ 8 0 (0) (0) 、 7 4 (A) (1) ~ 7 4 (0) (0) 構成に基づき、ベクトル累算結果サンプルセット 7 6 (A) (1) ~ 7 6 (0) (0) を与えるように、乗算ベクトル結果出力サンプルセット 6 8 (A) ~ 6 8 (0) を一緒に累算することを含む (ブロック 1 0 1) 。ベクトル処理は次いで、出力データ経路 7 4 (A) (1) ~ 7 4 (0) (0) 中でベクトル累算結果サンプルセット 7 6 (A) (1) ~ 7 6 (0) (0) を与えることを含む (ブロック 1 0 3) 。ベクトル処理は次いで、出力処理段 6 0 (3) 中の累算器ブロック 7 2 (A) (1) ~ 7 2 (0) (0) からベクトル結果出力サンプルセット 7 6 (A) (1) ~ 7 6 (0) (0) を受信することを含む (ブロック 1 0 5) 。

【 0 0 4 3 】

[0055]__プログラム可能データ経路構成を有するベクトル処理ブロックを利用する、図 3 の例示的な V P E 2 2 および図 4 A および図 4 B のベクトル処理の概観について記載したので、記述の残りでは、図 5 ~ 図 9 C において、これらのベクトル処理ブロックのより例示的な、非限定的詳細について記載する。

【 0 0 4 4 】

[0056]__この点において、図 5 は、図 3 の V P E 2 2 の M 0 乗算ベクトル処理段 6 0 (1) 中の複数の乗算器ブロック 6 2 (3) ~ 6 2 (0) のより詳細な配線図である。図 6 は、図 5 の乗算器ブロック 6 2 の内部構成要素の配線図である。図 5 に示すように、特定の入力データ経路 A 3 ~ A 0 、 B 3 ~ B 0 、 C 3 ~ C 0 に従って乗算器ブロック 6 2 (3) ~ 6 2 (0) によって受信されるベクトルデータ入力サンプルセット 6 6 (1 1) ~ 6 6 (0) が示されている。図 6 に関連して後でより詳しく論じるように、この例における乗算器ブロック 6 2 (3) ~ 6 2 (0) の各々は、4 つの 8 ビット × 8 ビット乗算器を含む。図 5 に参照を戻すと、この例における乗算器ブロック 6 2 (3) ~ 6 2 (0) の各々は、被乗数入力「 A 」を、被乗数入力「 B 」または被乗数入力「 C 」のいずれかで乗算するように構成される。乗算器ブロック 6 2 中で掛け合わせられ得る被乗数入力「 A 」、および「 B 」または「 C 」は、図 5 に示すように、どの入力データ経路 A 3 ~ A 0 、 B 3 ~ B 0 、 C 3 ~ C 0 が乗算器ブロック 6 2 (3) ~ 6 2 (0) に接続されるかによって制御される。被乗数セレクト入力 8 2 (3) ~ 8 2 (0) が、被乗数入力「 B 」それとも被乗数入力「 C 」が被乗数入力「 A 」で乗算されるために選択されるかを選択するために、各乗算器ブロック 6 2 (3) ~ 6 2 (0) 中のプログラム可能内部データ経路 6 7 (3) ~ 6 7 (0) を制御するための入力として、各乗算器ブロック 6 2 (3) ~ 6 2 (0) に与えられる。このようにして、乗算器ブロック 6 2 (3) ~ 6 2 (0) は、それらのプログラム可能内部データ経路 6 7 (3) ~ 6 7 (0) が、必要に応じて異なる乗算演算を行うように再プログラムされるという能力を備えている。

【 0 0 4 5 】

[0057]__引き続き図 5 を参照すると、乗算器ブロック 6 2 (3) を一例として使うと、入力データ経路 A 3 および A 2 が、それぞれ入力 A H および A L に接続される。入力 A H は、被乗数入力 A の上位ビットを表し、A L は、入力被乗数入力「 A 」の下位ビットを意味する。入力データ経路 B 3 および B 2 は、それぞれ入力 B H および B L に接続される。入力 B H は、被乗数入力「 B 」の上位ビットを表し、A L は、入力被乗数入力「 B 」の下位ビットを表す。入力データ経路 C 3 および C 2 は、それぞれ入力 C I および C Q に接続される。入力 C I は、この例では入力被乗数入力「 C 」の実数ビット部分を表す。C Q は、この例では入力被乗数入力「 C 」の虚数ビット部分を表す。図 6 に関連して後でより詳しく論じるように、被乗数セクタ入力 8 2 (3) は、この例では、乗算器ブロック 6 2 (3) のプログラム可能内部データ経路 6 7 (3) が、被乗数入力「 A 」に対する 8 ビット乗算を被乗数入力「 B 」それとも被乗数入力「 C 」で実施するように構成されるか、または乗算器ブロック 6 2 (3) が、被乗数入力「 A 」に対する 1 6 ビット乗算を被乗数入力「 B 」それとも被乗数入力「 C 」で実施するように構成されるかも制御する。

10

【 0 0 4 6 】

[0058]__引き続き図 5 を参照すると、乗算器ブロック 6 2 (3) ~ 6 2 (0) は、それらのプログラム可能内部データ経路 6 7 (3) ~ 6 7 (0) の構成に基づいて、乗算演算の桁上げ「 C 」および和「 S 」ベクトル出力サンプルセットとしてベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) を各々が生成するように構成される。図 7 ~ 図 9 C に関連して後でより詳しく論じるように、ベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) の桁上げ「 C 」および和「 S 」は融合され、桁上げ「 C 」および和「 S 」が、複数の累算器 7 2 (3) ~ 7 2 (0) 中で冗長桁上げ保存フォーマットを提供するように、冗長桁上げ保存フォーマットで複数の累算器 7 2 (3) ~ 7 2 (0) に与えられることを意味する。後でより詳しく論じるように、複数の累算器 7 2 (3) ~ 7 2 (0) 中で冗長桁上げ保存フォーマットを提供することにより、複数の累算器 7 2 (3) ~ 7 2 (0) によって実施される累算演算中に桁上げ伝搬経路と桁上げ伝搬加算演算とを提供する必要をなくすることができる。

20

【 0 0 4 7 】

[0059]__ベクトル乗算出力サンプルセット 6 8 (3) ~ 6 8 (0) を、それらのプログラム可能内部データ経路 6 7 (3) ~ 6 7 (0) の構成に基づいて乗算演算の桁上げ「 C 」および和「 S 」ベクトル出力サンプルセットとして生成する乗算器ブロック 6 2 (3) ~ 6 2 (0) の例を、図 5 に示す。たとえば、乗算器ブロック 6 2 (3) は、桁上げ C 0 0 と和 S 0 0 とを 8 ビット乗算用の 3 2 ビット値として、桁上げ C 0 1 と和 S 0 1 とを 1 6 ビット乗算用の 6 4 ビット値として生成するように構成される。他の乗算器ブロック 6 2 (2) ~ 6 2 (0) も、この例では同じ能力を有する。この点において、乗算器ブロック 6 2 (2) は、桁上げ C 1 0 と和 S 1 0 とを 8 ビット乗算用の 3 2 ビット値として、桁上げ C 1 1 と和 S 1 1 とを 1 6 ビット乗算用の 6 4 ビット値として生成するように構成される。乗算器ブロック 6 2 (1) は、桁上げ C 2 0 と和 S 2 0 とを 8 ビット乗算用の 3 2 ビット値として、桁上げ C 2 1 と、和 S 2 1 とを 1 6 ビット乗算用の 6 4 ビット値として生成するように構成される。乗算器ブロック 6 2 (0) は、桁上げ C 3 0 と和 S 3 0 とを 8 ビット乗算用の 3 2 ビット値として、桁上げ C 3 1 と和 S 3 1 とを 1 6 ビット乗算用の 6 4 ビット値として生成するように構成される。

30

40

【 0 0 4 8 】

[0060]__図 5 の乗算器ブロック 6 2 中に設けられるプログラム可能データ経路構成のより例示的な詳細を説明するために、図 6 が挙げられる。図 6 は、8 ビット × 8 ビットベクトルデータ入力サンプルセット 6 6 と、1 6 ビット × 1 6 ビットベクトルデータ入力サンプルセット 6 6 とを乗算することが可能なプログラム可能データ経路構成を有する、図 3 および図 4 の乗算器ブロック 6 2 の内部構成要素の配線図である。この点において、乗算器ブロック 6 2 は、この例では 4 つの 8 × 8 ビット乗算器 8 4 (3) ~ 8 4 (0) を含む。任意の所望の数の乗算器 8 4 が設けられてよい。第 1 の乗算器 8 4 (3) が、8 ビット

50

ベクトルデータ入力サンプルセット 66A[H] (入力被乗数入力「A」の上位ビットである)を受信し、ベクトルデータ入力サンプルセット 66A[H]を、8ビットベクトルデータ入力サンプルセット 66B[H] (入力被乗数入力「B」の上位ビットである)または8ビットベクトルデータ入力サンプルセット 66C[I] (入力被乗数入力「C」の上位ビットである)と乗算するように構成される。被乗数として乗算器 84(3)に与えている8ビットベクトルデータ入力サンプルセット 66B[H]または8ビットベクトルデータ入力サンプルセット 66C[I]のいずれかを選択するように構成されたマルチプレクサ 86(3)が設けられる。マルチプレクサ 86(3)は、本実施形態では、乗数セクタ入力 82中の上位ビットである被乗数セクタビット入力 82[3]によって制御される。このようにして、マルチプレクサ 86(3)および被乗数セクタビット入力 82[3]は、8ビットベクトルデータ入力サンプルセット 66B[H]それとも8ビットベクトルデータ入力サンプルセット 66C[I]が、受信するベクトルデータ入力サンプルセット 66A[H]と乗算されるかを乗算器 84(3)が制御するためのプログラム可能内部データ経路 67[0]構成を提供する。

【0049】

[0061]__引き続き図6を参照すると、他の乗算器 84(2)~84(0)も、第1の乗算器 84(3)用に設けられるものと同様のプログラム可能内部データ経路 67[2]~67[0]を含む。乗算器 84(2)は、被乗数入力「A」の下位ビットである8ビットベクトルデータ入力サンプルセット 66A[L]と乗算されるべき、8ビットベクトルデータ入力サンプルセット 66B[H]または8ビットベクトルデータ入力サンプルセット 66C[I]のいずれかをプログラム可能内部データ経路 67[1]中で与えるための、プログラム可能構成を有するプログラム可能内部データ経路 67[2]を含む。選択は、本実施形態では、マルチプレクサ 86(2)によって、乗数セクタ入力 82中の被乗数セクタビット入力 82[2]に従って制御される。乗算器 84(1)は、8ビットベクトルデータ入力サンプルセット 66A[H]と乗算されるべき、被乗数入力「B」の下位ビットである8ビットベクトルデータ入力サンプルセット 66B[L]、または被乗数入力「C」の下位ビットである8ビットベクトルデータ入力サンプルセット 66C[Q]のいずれかをプログラム可能内部データ経路 67[1]中で与えるようにプログラム可能なプログラム可能内部データ経路 67[1]を含む。選択は、本実施形態では、マルチプレクサ 86(1)によって、乗数セクタ入力 82中の被乗数セクタビット入力 82[1]に従って制御される。さらに、乗算器 84(0)は、8ビットベクトルデータ入力サンプルセット 66A[L]と乗算されるべき、8ビットベクトルデータ入力サンプルセット 66B[L]または8ビットベクトルデータ入力サンプルセット 66C[Q]のいずれかをプログラム可能内部データ経路 67[0]中で与えるようにプログラム可能なプログラム可能内部データ経路 67[0]を含む。選択は、本実施形態では、マルチプレクサ 86(0)によって、乗数セクタ入力 82中の被乗数セクタビット入力 82[0]に従って制御される。

【0050】

[0062]__引き続き図6を参照すると、上述したように、乗算器 84(3)~84(0)は、異なるビット長乗算演算を実施するように構成され得る。この点において、各乗算器 84(3)~84(0)は、それぞれビット長乗算モード入力 88(3)~88(0)を含む。この例では、各乗算器 84(3)~84(0)は、それぞれ、プログラム可能データ経路 90(3)~90(0)、91、および92(3)~92(0)の構成を制御する入力に従って、8ビット×8ビットモードでプログラムされ得る。各乗算器 84(3)~84(0)は、それぞれ、プログラム可能データ経路 90(3)~90(0)、91、および92(3)~92(0)の構成を制御する入力に従って、16ビット×16ビットモードと24ビット×8ビットモードとを含む、より大きいビット乗算演算の一部を行うようにもプログラムされ得る。たとえば、各乗算器 84(3)~84(0)が、プログラム可能データ経路 90(3)~90(0)の構成に従って8ビット×8ビット乗算モードで構成される場合、ユニットとしての複数の乗算器 84(3)~84(0)は、2つの個々

10

20

30

40

50

の 8 ビット× 8 ビット乗算器を、乗算器ブロック 6 2 の一部として備えるように構成され得る。各乗算器 8 4 (3) ~ 8 4 (0) が、プログラム可能データ経路 9 1 の構成に従って 1 6 ビット× 1 6 ビット乗算モードで構成される場合、ユニットとしての複数の乗算器 8 4 (3) ~ 8 4 (0) は、単一 1 6 ビット× 1 6 ビット乗算器を乗算器ブロック 6 2 の一部として備えるように構成され得る。乗算器 8 4 (3) ~ 8 4 (0) が、プログラム可能データ経路 9 2 (3) ~ 9 2 (0) の構成に従って 2 4 ビット× 8 ビット乗算モードで構成される場合、ユニットとしての複数の乗算器 8 4 (3) ~ 8 4 (0) は、1 つの 1 6 ビット× 2 4 ビット× 8 ビット乗算器を乗算器ブロック 6 2 の一部として備えるように構成され得る。

【 0 0 5 1 】

[0063]__引き続き図 6 を参照すると、この例における乗算器 8 4 (3) ~ 8 4 (0) は、1 6 ビット× 1 6 ビット乗算モードで構成されるものとして示されている。1 6 ビット入力 9 4 (3)、9 4 (2) および入力桁上げ 9 6 (3)、9 6 (2) が、それぞれ、各乗算器 8 4 (3)、8 4 (2) によって生成される。1 6 ビット入力 9 4 (1)、9 4 (0) および入力桁上げ 9 6 (1)、9 6 (0) が、それぞれ、各乗算器 8 4 (1)、8 4 (0) によって生成される。1 6 ビット入力 9 4 (3)、9 4 (2) および入力桁上げ 9 6 (3)、9 6 (2) はまた、入力 9 4 (3) ~ 9 4 (0) と入力桁上げ 9 6 (3) ~ 9 6 (0) を加え合わせるように、1 6 ビット和の入力 9 4 (1)、9 4 (0) および入力桁上げ 9 6 (1)、9 6 (0) とともに 2 4 ビットの 4 : 2 圧縮器 1 0 9 に与えられる。加算された入力 9 4 (3) ~ 9 4 (0) および入力桁上げ 9 6 (3) ~ 9 6 (0) は、プログラム可能データ経路 9 1 がアクティブであるとともに入力 9 4 (3) ~ 9 4 (0) および入力桁上げ 9 6 (3) ~ 9 6 (0) とゲートされるとき、1 6 ビット× 1 6 ビット乗算モードで、単一和 9 8 と単一桁上げ 1 0 0 とをもたらす。プログラム可能データ経路 9 1 は、2 4 ビットの 4 : 2 圧縮器 1 0 9 に与えられるように、1 6 ビットワードとして、組み合わされた入力 9 4 (3)、9 4 (2) をもつ第 1 の AND ベースのゲート 1 0 2 (3) によって、および 1 6 ビットワードとして、組み合わされた入力桁上げ 9 6 (3)、9 6 (2) をもつ第 2 の AND ベースのゲート 1 0 2 (2) によってゲートされる。プログラム可能データ経路 9 1 はまた、2 4 ビットの 4 : 0 圧縮器 1 0 9 に与えられるように、1 6 ビットワードとして、組み合わされた入力 9 4 (1)、9 4 (0) をもつ第 3 の AND ベースのゲート 1 0 2 (1) によって、および 1 6 ビットワードとして、組み合わされた入力桁上げ 9 6 (1)、9 6 (0) をもつ第 4 の AND ベースのゲート 1 0 2 (0) によってゲートされる。乗算器ブロック 6 2 が 1 6 ビット× 1 6 ビットまたは 2 4 ビット× 8 ビット乗算モードで構成される場合、プログラム可能出力データ経路 7 0 [0] には、ベクトル乗算出力サンプルセット 6 8 [0] が、圧縮された 3 2 ビット和 S 0 および 3 2 ビット桁上げ C 0 部分積として与えられる。

【 0 0 5 2 】

[0064]__乗算器ブロック 6 2 中の乗算器 8 4 (3) ~ 8 4 (0) が 8 ビット× 8 ビット乗算モードで構成される場合、プログラム可能出力データ経路 7 0 [1] 構成は、圧縮なしで、1 6 ビット入力 9 4 (3) ~ 9 4 (0) および部分積としての対応する 1 6 ビット入力桁上げ 9 6 (3) ~ 9 6 (0) として提供される。乗算器ブロック 6 2 中の乗算器 8 4 (3) ~ 8 4 (0) が 8 ビット× 8 ビット乗算モードで構成される場合、プログラム可能出力データ経路 7 0 [1] は、圧縮なしで、1 6 ビット入力 9 4 (3) ~ 9 4 (0) およびベクトル乗算出力サンプルセット 6 8 [1] としての対応する 1 6 ビット入力桁上げ 9 6 (3) ~ 9 6 (0) として提供される。乗算器ブロック 6 2 の乗算モードに依存するベクトル乗算出力サンプルセット 6 8 [0]、6 8 [1] は、実行されるベクトル命令に従って、和および桁上げ積の累算のために、累算器ブロック 7 2 (3) ~ 7 2 (0) に与えられる。

【 0 0 5 3 】

[0065]__プログラム可能データ経路構成を有する、図 4 および図 5 の乗算器ブロック 6 2 (3) ~ 6 2 (0) について記載したので、冗長桁上げ保存フォーマットで構成された

10

20

30

40

50

累算器ブロック 72(3) ~ 72(0) と融合されるべき、VPE 22 中の乗算器ブロック 62(3) ~ 62(0) の特徴について、図 7 に関連してここで概説する。

【0054】

【0066】__この点について、図 7 は、図 2 および図 3 の VPE 中の乗算器ブロックおよび累算器ブロックの一般化配線図であり、ここにおいて、累算器ブロックは、桁上げ伝搬を低減するのに冗長桁上げ保存フォーマットを利用する桁上げ保存累算器構造を利用する。上述し、図 7 に示したように、乗算器ブロック 62 は、被乗数入力 66[H] および 66[L] を乗算し、少なくとも 1 つの入力 94 と少なくとも 1 つの入力桁上げ 96 を、ベクトル乗算出力サンプルセット 68 としてプログラム可能出力データ経路 70 中で与えるように構成される。各累算ステップ用の累算器ブロック 72 中に桁上げ伝搬経路と桁上げ伝搬加算器とを設ける必要をなくするために、プログラム可能出力データ経路 70 中のベクトル乗算出力サンプルセット 68 中の少なくとも 1 つの入力 94 および少なくとも 1 つの入力桁上げ 96 が、少なくとも 1 つの累算器ブロック 72 への冗長桁上げ保存フォーマットで融合される。言い換えると、ベクトル乗算出力サンプルセット 68 中の桁上げ 96 が、ベクトル入力桁上げ 96 として桁上げ保存フォーマットで累算器ブロック 72 に与えられる。このようにして、ベクトル乗算出力サンプルセット 68 中の入力 94 および入力桁上げ 96 は、本実施形態では複合ゲート 4:2 圧縮器である累算器ブロック 72 の圧縮器 108 に与えられ得る。圧縮器 108 は、入力 94 および入力桁上げ 96 を、それぞれ、以前の累算ベクトル出力 112 および以前のシフトされた累算ベクトル出力桁上げ 117 と一緒に累算するように構成される。以前のシフトされた累算ベクトル出力桁上げ 117 は本質的に、累算演算中の保存された桁上げ累算である。

【0055】

【0067】__このようにして、単一最終桁上げ伝搬加算器のみが、受信入力桁上げ 96 を、累算器ブロック 72 によって生成された累算の一部として、入力 94 に伝搬するために、累算器ブロック 72 中に設けられることを求められるのではない。累算器ブロック 72 中の各累算ステップ中に桁上げ伝搬加算演算を実施することに関連した電力消費が、本実施形態では低減される。また、累算器ブロック 72 中の各累算ステップ中に桁上げ伝搬加算演算を実施することに関連したゲート遅延も、本実施形態ではなくなる。

【0056】

【0068】__引き続き図 7 を参照すると、圧縮器 108 は、冗長な形での入力 94 および入力桁上げ 96 を、それぞれ、以前の累算ベクトル出力 112 および以前のシフトされた累算ベクトル出力桁上げ 117 と累算するように構成される。シフトされた累算ベクトル出力桁上げ 117 は、次の受信入力 94 および入力桁上げ 96 の次の累算が圧縮器 108 によって実施される前に、累算ベクトル出力桁上げ 114 をシフトすることによって、圧縮器 108 ビットによって生成された累算ベクトル出力桁上げ 114 によって生成される。最終的なシフトされた累算ベクトル出力桁上げ 117 は、累算器ブロック 72 に設けられた単一最終桁上げ伝搬加算器 119 によって最終累算ベクトル出力 112 に加算され、最終的なシフトされた累算ベクトル出力桁上げ 117 を、最終累算ベクトル出力 112 を最終累算器出力サンプルセット 762 の補数表現にコンバートするために桁上げ累算を伝搬する。最終累算ベクトル出力 112 は、プログラム可能出力データ経路 74 中で累算器出力サンプルセット 76 として与えられる(図 3 参照)。

【0057】

【0069】__乗算器ブロック 62 と、冗長桁上げ保存フォーマットで構成された累算器ブロック 72 の融合を示す図 7 について説明したので、累算器ブロック 72(3) ~ 72(0) に関する、より例示的な詳細について、図 8 に関連してここで概説する。以下で説明する図 9A ~ 図 9C は、共通回路要素およびハードウェアを用いて異なるベクトル累算演算を行うための異なる累算モードにおける、冗長桁上げ保存フォーマットで構成された累算器ブロック 72(3) ~ 72(0) のより例示的な詳細を与える。

【0058】

【0070】__図 8 は、図 3 の VPE 22 中に設けられる累算器ブロック 72 の例示的な内部

構成要素の詳細な配線図である。上述するとともに後でより詳しく論じるように、累算器ブロック72は、プログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)を有して構成され、そうすることによって累算器ブロック72は、特定の異なるタイプのベクトル累算演算を実施するように設計された専用回路要素として作用するようにプログラムされ得る。たとえば、累算器ブロック72は、符号付きおよび符号なし累算演算を含む、いくつかの異なる累算および加算を行うようにプログラムされ得る。異なるタイプの累算演算を行うように構成された累算器ブロック72中のプログラム可能入力データ経路78(3)~78(0)および/またはプログラム可能内部データ経路80(3)~80(0)の具体例が、後で論じる図9A~図9Cに示される。また、累算器ブロック72は、低減された組合せ論理で高速累算演算を行うために、桁上げ伝搬を回避または低減するために、冗長桁上げ算術を行うための桁上げ保存累算器72[0]、72[1]を含むように構成される。

10

【0059】

[0071]__累算器ブロック72の例示的な内部構成要素が、図8に示される。そこに示されているように、本実施形態における累算器ブロック72は、第1の入力和94[0]および第1の入力桁上げ96[0]と、第2の入力和94[1]および第2の入力桁上げ96[1]とを、一緒に累算するために乗算器ブロック62から受信するように構成される。図8に関連して、入力和94[0]、94[1]および入力桁上げ96[0]、96[1]は、ベクトル入力和94[0]、94[1]およびベクトル入力桁上げ96[0]、96[1]と呼ばれる。上で記載するとともに図6に示すように、本実施形態におけるベクトル入力和94[0]、94[1]およびベクトル入力桁上げ96[0]、96[1]は各々、長さが16ビットである。この例における累算器ブロック72は、2つの24ビット桁上げ保存累算器72[0]、72[1]として設けられ、各々が、共通要素番号をもつ同様の構成要素を含み、「[0]」は桁上げ保存累算器72[0]用に指定され、「[1]」は桁上げ保存累算器72[1]用に指定される。桁上げ保存累算器72[0]、72[1]は、ベクトル累算演算を同時に実施するように構成され得る。

20

【0060】

[0072]__図8の桁上げ保存累算器72[0]を参照すると、ベクトル入力和94[0]およびベクトル入力桁上げ96[0]は、プログラム可能内部データ経路80[0]の一部として設けられたマルチプレクサ104(0)中の入力である。負のベクトル入力和94[0]'と負のベクトル入力桁上げ96[0]'とを必要とする累算演算のための、マルチプレクサ104(0)への入力としての入力107(0)に従って負のベクトル入力和94[0]'と負のベクトル入力桁上げ96[0]'とを生成する、排他的ORベースのゲートからなり得る否定回路106(0)も設けられる。マルチプレクサ104(0)は、ベクトル命令復号の結果として生成されたセクタ入力110(0)に従って、圧縮器108(0)に与えられるべきベクトル入力和94[0]およびベクトル入力桁上げ96[0]または負のベクトル入力和94[0]'および負のベクトル入力桁上げ96[0]'のいずれかを選択するように構成される。この点において、セクタ入力110(0)は、累算器ブロック72によって実施されるように構成された累算演算に従って、桁上げ保存累算器72[0]のプログラム可能入力データ経路78[0]を、ベクトル入力和94[0]およびベクトル入力桁上げ96[0]、または負のベクトル入力和94[0]'および負のベクトル入力桁上げ96[0]'のいずれかを、圧縮器108(0)に与えるようにプログラム可能にさせる。

30

40

【0061】

[0073]__引き続き図8を参照すると、本実施形態における桁上げ保存累算器72[0]の圧縮器108(0)は、複合ゲート4:2圧縮器である。この点において、圧縮器108(0)は、冗長桁上げ保存演算において和と桁上げとを累算するように構成される。圧縮器108(0)は、現在のベクトル入力和94[0]およびベクトル入力桁上げ96[0]、または現在の負のベクトル入力和94[0]'および負のベクトル入力桁上げ96[0]'を、圧縮器108(0)への4つの入力として、以前の累算されたベクトル入力

50

和 9 4 [0] およびベクトル入力桁上げ 9 6 [0]、または累算された負のベクトル入力
 和 9 4 [0] ' および負のベクトル入力桁上げ 9 6 [0] ' と一緒に累算するように構成
 される。圧縮器 1 0 8 (0) は、累算ベクトル出力和 1 1 2 (0) と、累算ベクトル出力
 桁上げ 1 1 4 (0) とを、累算器出力サンプルセット 7 6 (3) ~ 7 6 (0) を与えるよ
 うに、プログラム可能出力データ経路 7 4 [0] 中の累算器出力サンプルセット 7 6 [0]
 として与える (図 3 参照)。累算ベクトル出力桁上げ 1 1 4 (0) は、各累算ステップ
 中にビット幅成長を制御するための、シフトされた累算ベクトル出力桁上げ 1 1 7 (0)
 を与えるように、累算演算中にビットシフト 1 1 6 (0) によってシフトされる。たとえ
 ば、本実施形態におけるビットシフト 1 1 6 (0) は、冗長桁上げ保存フォーマットでの
 、圧縮器 1 0 8 (0) に融合されるバレルシフトである。このようにして、シフトされた
 累算ベクトル出力桁上げ 1 1 7 (0) は本質的に、累算器 7 2 [0] によって実施される
 累算演算中に、累算ベクトル出力和 1 1 2 (0) に伝搬される必要なく、保存される。こ
 のようにして、累算器 7 2 [0] 中の各累算ステップ中に桁上げ伝搬加算演算を実施する
 ことに関連した電力消費およびゲート遅延が、本実施形態ではなくなる。

【 0 0 6 2 】

[0074]__追加の後続ベクトル入力 和 9 4 [0] およびベクトル入力桁上げ 9 6 [0]、
 または負のベクトル入力 和 9 4 [0] ' および負のベクトル入力桁上げ 9 6 [0] ' は、
 現在の累算ベクトル出力和 1 1 2 (0) および現在の累算ベクトル出力桁上げ 1 1 7 (0)
 と累算され得る。ベクトル入力 和 9 4 [0] およびベクトル入力桁上げ 9 6 [0]、ま
 たは負のベクトル入力 和 9 4 [0] ' および負のベクトル入力桁上げ 9 6 [0] ' は、
 ベクトル命令復号の結果として生成された和桁上げセレクタ 1 2 0 (0) に従って、プロ
 グラム可能内部データ経路 8 0 [0] の一部としてのマルチプレクサ 1 1 8 (0) によ
 って選択される。現在の累算ベクトル出力和 1 1 2 (0) および現在のシフトされた累算
 ベクトル出力桁上げ 1 1 7 (0) は、桁上げ保存累算器 7 2 [0] が、アップデートされた累
 算ベクトル出力和 1 1 2 (0) と、累算ベクトル出力桁上げ 1 1 4 (0) とを与えるた
 めに、入力として圧縮器 1 0 8 (0) に与えられ得る。この点において、和桁上げセレクタ
 1 2 0 (0) は、累算器 7 2 [0] のプログラム可能内部データ経路 8 0 [0] を、累算
 器ブロック 7 2 によって実施されるように構成された累算演算に従って、ベクトル入力
 和 9 4 [0] と、ベクトル入力桁上げ 9 6 [0] とを圧縮器 1 0 8 (0) に与えるように
 プログラム可能にさせる。マルチプレクサ 1 1 8 (0) に、桁上げ保存累算器 7 2 [0] に
 おける累算の動作タイミングを制御するための保持状態入力 1 2 6 (0) に従って、累算
 ベクトル出力和 1 1 2 (0) およびシフトされた累算ベクトル出力桁上げ 1 1 7 (0) の
 現在の状態を保持させるために、保持ゲート 1 2 2 (0)、1 2 4 (0) も本実施形態で
 は設けられる。

【 0 0 6 3 】

[0075]__引き続き図 8 を参照すると、桁上げ保存累算器 7 2 [0] の累算ベクトル出力
 和 1 1 2 (0) およびシフトされた累算ベクトル出力桁上げ 1 1 7 (0)、ならびに桁上
 げ保存累算器 7 2 [1] の累算ベクトル出力和 1 1 2 (1) およびシフトされた累算ベ
 クトル出力桁上げ 1 1 7 (1)、ならびには、それぞれ、制御ゲート 1 3 4 (0)、1 3 6
 (0) および 1 3 4 (1)、1 3 6 (1) によってゲートされる。制御ゲート 1 3 4 (0)
 、1 3 6 (0) および 1 3 4 (1)、1 3 6 (1) は、それぞれ、圧縮器 1 0 8 (0)
 、1 0 8 (1) に戻される、累算ベクトル出力和 1 1 2 (0) およびシフトされた累算
 ベクトル出力桁上げ 1 1 7 (0) と、累算ベクトル出力和 1 1 2 (1) およびシフトされた
 累算ベクトル出力桁上げ 1 1 7 (1) とを制御する。

【 0 0 6 4 】

[0076]__図 8 の累算器ブロック 7 2 の累算器 7 2 [0]、7 2 [1] のプログラム可能
 入力データ経路 7 8 [0]、7 8 [1] およびプログラム可能内部データ経路 8 0 [0]
 、8 0 [1] を要約すると、累算器ブロック 7 2 は異なるモードで構成され得る。累算器
 ブロック 7 2 は、図 8 に示す共通累算器回路要素を用いた特定のベクトル処理命令に従
 って、異なる累算演算を行うように構成され得る。共通累算器回路要素を用いた特定のベ

10

20

30

40

50

トル処理命令に従って異なる累算演算を行うように構成される累算器ブロック 72 の例を、以下の図 9 A ~ 図 9 C に示す。

【 0 0 6 5 】

[0077]__この点において、図 9 A は、図 8 の同じ累算器ブロック 72 の配線図である。この例では、累算器ブロック 72 は、二重 24 ビット累算器モードを提供するようにプログラムされた、プログラム可能入力データ経路 78 [0]、78 [1] と、プログラム可能内部データ経路 80 [0]、80 [1] とを有する。図 9 A の累算器ブロック 72 中の各桁上げ保存累算器 72 [0]、72 [1] は、24 ビット累算器を提供するように構成される。累算器 72 [0]、72 [1] の 24 ビット累算容量は、圧縮器 108 (0)、108 (1) のビット容量によって与えられる。図 8 に関連した累算器 72 [0]、72 [1] の考察により、図 9 A の累算器 72 [0]、72 [1] によって行われる個々の 24 ビット累算について説明する。累算ブロック 72 において二重 24 ビット累算を行うためのバイ累算器 72 [0]、72 [1] によって実施される累算の一般的なデータ経路が、それぞれ、図 9 A の、プログラム可能累算データ経路 132 (0) および 132 (1) 中に示されている。

【 0 0 6 6 】

[0078]__引き続き図 9 A を参照して、桁上げ保存累算器 72 [0] の例示的な構成要素を記載するが、桁上げ保存累算器 72 [1] にも等しく適用可能である。本実施形態では、桁上げ保存累算器 72 [0] の累算ベクトル出力和 112 (0) およびシフトされた累算ベクトル出力桁上げ 117 (0)、ならびに桁上げ保存累算器 72 [1] の累算ベクトル出力和 112 (1) およびシフトされた累算ベクトル出力桁上げ 117 (1)、ならびには、それぞれ、制御ゲート 134 (0)、136 (0) および 134 (1)、136 (1) によってゲートされる。制御ゲート 134 (0)、136 (0) および 134 (1)、136 (1) は、それぞれ、圧縮器 108 (0)、108 (1) に戻される、累算ベクトル出力和 112 (0) およびシフトされた累算ベクトル出力桁上げ 117 (0) と、累算ベクトル出力和 112 (1) およびシフトされた累算ベクトル出力桁上げ 117 (1) とを制御する。ベクトル命令の復号から、制御ゲート 134 (0)、136 (0) と 134 (1)、136 (1) の両方に与えられた制御入力 138 (0)、138 (1) は、それぞれ、累算ベクトル出力和 112 (0) と、シフトされた累算ベクトル出力桁上げ 117 (0) とを制御し、累算ベクトル出力和 112 (1) およびシフトされた累算ベクトル出力桁上げ 117 (1) は、それぞれ、圧縮器 108 (0)、108 (1) に戻される。制御入力 138 (0)、138 (1) および制御ゲート 134 (0)、136 (0) は、累算器 72 [0]、72 [1] が、図 9 B および図 9 C に関連して後でより詳しく論じるように、別個の累算演算それとも組み合わせられた累算演算を各々が実施するようにプログラムされるかを制御する。したがって、制御入力 138 (0)、138 (1) ならびに制御ゲート 134 (0)、136 (0) および 134 (1)、136 (1) は、本実施形態では、それぞれ、累算器 72 [0]、72 [1] のプログラム可能内部データ経路 80 [0]、80 [1] の一部を形成する。

【 0 0 6 7 】

[0079]__図 8 に参照を戻すと、累算器ブロック 72 のプログラム可能内部データ経路 80 [0]、80 [1] はまた、個々の累算器 72 [0]、72 [1] の 24 ビット容量よりも大きい累算演算を実施するようにプログラムされ、かつ / または再プログラムされ得る。この点において、累算器 72 [0]、72 [1] のプログラム可能内部データ経路 80 [0]、80 [1] は、単一ベクトル累算演算において一緒に利用されるべき両方の累算器 72 [0]、72 [1] を提供するようにプログラムされ得る。累算器 72 [0]、72 [1] は、累算器 72 [0]、72 [1] の個々のビット累算容量よりも大きいビット容量の単一累算演算を行うようにプログラムされ得る。累算器 72 [0]、72 [1] のプログラム可能内部データ経路 80 [0]、80 [1] は、桁上げ保存累算器 72 [0] に、圧縮器 108 (0) から、次の桁上げ出力 (N C O) 137 (0) としてオーバーフロー桁上げ出力を伝搬させるように構成され得る。N C O 137 (0) は、次の桁上げ

入力 (NCI) 139 (1) として、桁上げ保存累算器 72 [1] 中の圧縮器 108 (1) に与えられ得る。図 8 に関連して上で記載したように、本実施形態では、40 ビット累算を行うための累算器 72 [0]、72 [1] に、24 ビット桁上げおよび和累算への 24 ビットオーバーフロー桁上げ伝搬を行わせるための、累算器 72 [0]、72 [1] 中でプログラム可能内部データ経路 80 [0]、80 [1] として提供されるこの桁上げ伝搬構成能力。

【0068】

[0080] この点において、図 9 B は、図 8 の同じ累算器ブロック 72 の配線図である。図 9 B において、累算器ブロック 72 が、単一累算演算モードで構成されて示されている。図 9 B において、累算器 72 [0]、72 [1] は、累算器ブロック 72 中で単一累算演算を行うために構成されたプログラム可能内部データ経路 80 [0]、80 [1] を有する。累算器ブロック 72 の単一累算器モードのこの例において、累算器 72 [0]、72 [1] は、単一 40 ビット累算ベクトル出力和 112 とシフトされた累算ベクトル出力桁上げ 117 とを累算する。単一累算ベクトル出力和 112 は、プログラム可能出力データ経路 74 [0]、74 [1] 中で累算器出力サンプルセット 76 として与えられる累算ベクトル出力和 112 (0)、112 (1) からなる (図 3 も参照)。累算ベクトル出力和 112 (0) は、単一 40 ビット累算ベクトル出力和 112 の最下位ビットを備える。累算ベクトル出力和 112 (1) は、単一 40 ビット累算ベクトル出力和 112 の最上位ビットを備える。同様に、シフトされた累算ベクトル出力桁上げ 117 は、シフトされた累算出力桁上げ 117 (0)、117 (1) からなる。シフトされた累算ベクトル出力桁上げ 117 (0) は、単一シフト累算ベクトル出力桁上げ 117 の最下位ビットを備える。累算ベクトル出力和 114 (1) は、単一 40 ビット累算ベクトル出力桁上げ 114 の最上位ビットを備える。累算ベクトル出力和 114 [0] およびシフトされた累算ベクトル出力桁上げ 117 (0) は、プログラム可能出力データ経路 74 [0] 中に設けられる (図 3 参照)。

【0069】

[0081] 引き続き図 9 B を参照すると、累算器ブロック 72 が単一累算モード (たとえば、40 ビット累算) で構成されるときに累算器 72 [0]、72 [1] によって実施される累算演算の一般的なデータ経路が、プログラム可能累算データ経路 146 として示されている。この点において、累算器ブロック 72 は、上で記載したように、ベクトル入力 94 とベクトル入力桁上げ 96 とを受信する。桁上げ保存累算器 72 [0] は、場合によっては、ベクトル入力 94 [0] およびベクトル入力桁上げ 96 [0]、または負のベクトル入力 94 [0] ' および負のベクトル入力桁上げ 96 [0] の累算からの、累算ベクトル出力和 112 (0) と、累算ベクトル出力桁上げ 114 (0) との最下位ビットを累算する。桁上げ保存累算器 72 [1] は、場合によっては、ベクトル入力 94 [0] およびベクトル入力桁上げ 96 [0]、または負のベクトル入力 94 [0] ' および負のベクトル入力桁上げ 96 [0] の累算からの、累算ベクトル出力和 112 (1) と、累算ベクトル出力桁上げ 114 (1) との最上位ビットを累算する。

【0070】

[0082] 引き続き図 9 B を参照すると、単一累算ベクトル出力和 112 と、累算ベクトル出力桁上げ 114 とを与えるように累算器 72 [0]、72 [1] をプログラムするために、累算器 72 [0]、72 [1] のプログラム可能内部データ経路 80 [0]、80 [1] は、単一累算演算を行うようにプログラムされる。この点において、桁上げ保存累算器 72 [0] の圧縮器 108 (0) の NCI 137 (0) および桁上げ保存累算器 72 [1] の圧縮器 108 (1) の NCI 139 (1) は、累算器ブロック 72 中に単一累算器 (たとえば、40 ビット累算器) を設けるために構成される。桁上げ保存累算器 72 [1] の NCI 139 (1) は、NCI 制御入力 142 (1) を用いて NCI ゲート 140 (1) によってゲートされる。このようにして、累算器ブロック 72 中の累算器 72 [0]、72 [1] が、桁上げ保存累算器 72 [0] から桁上げ保存累算器 72 [1] へのオーバーフロー桁上げ伝搬を利用して単一累算演算を行うことを所望されるとき、NCI 制

10

20

30

40

50

御入力 142 (1) は、桁上げ保存累算器 72 [1] のプログラム可能内部データ経路 80 [1] の一部としてアクティブにされ得る。NCI 制御入力 142 (1) は、NCI ゲート 140 (1) に、圧縮器 108 (0) から圧縮器 108 (1) へのオーバーフロー桁上げ伝搬を許可させる。NCI 制御入力 142 (1) は、圧縮器 108 (0) に、圧縮器 108 (1) への NCI 139 (1) として NCO 137 (0) を生成させるように、桁上げ保存累算器 72 [0] 中の圧縮器 108 (0) の桁上げ伝搬入力 144 (0) にも結合される。これらの構成により、桁上げ保存累算器 72 [1] は、単一累算ベクトル出力和 112 と、累算ベクトル出力桁上げ 114 とを与え得るように、ベクトル入力 94 [1] と、ベクトル入力桁上げ 96 [1] とを累算することができるようになる。

【0071】

10

[0083] 累算器ブロック 72 中の桁上げ保存累算器 72 [1] は、図 9 B に示すように、NCI 139 (0) および NCI 制御入力 142 (0) によってゲートされる NCI ゲート 140 (0) も含むことに留意されたい。ただし、桁上げ保存累算器 72 [0] が単一累算演算の最下位ビットを累算するので、NCI 制御入力 142 (0) と NCI 139 (0) の両方は、本実施形態では論理「0」に結び付けられる。桁上げ保存累算器 72 [1] の圧縮器 108 (0) は、桁上げ伝搬入力 144 (1) も含むが、桁上げ伝搬入力 144 (1) は、本実施形態では、累算器 72 (1) に NCO 12 (1) を生成させないために、論理「0」に結び付けられることにも留意されたい。桁上げ保存累算器 72 [1] は、本実施形態では、次の累算器への桁上げ伝搬を実施する必要はなく、というのは、本実施形態において設けられる桁上げ保存累算器 72 [1] を超える、累算器ブロック 72

20

【0072】

[0084] 図 9 C は、図 8 の同じ累算器ブロック 72 の別の累算器モードの配線図である。この点において、図 9 C は、桁上げ保存加算器と、それに続く、累算器ブロック 72 中で別の累算モードを提供するための単一累算器を提供するための、プログラムされたデータ経路構成を有する、図 8 の累算器ブロック 72 の概略図である。いくつかの累算演算は、複合算術を実施するための余剰加算器を必要とし得る。図 9 C において、累算器 72 [0]、72 [1] は、16 ビット桁上げ保存加算器と、それに続く単一 24 ビット累算器を提供するために構成されたプログラム可能内部データ経路 80 [0]、80 [1] を有する。

30

【0073】

[0085] 図 9 C を参照すると、累算器ブロック 72 が桁上げ保存加算器と、それに続く単一累算器中で構成されるときに累算器 72 [0]、72 [1] によって実施される累算の一般的なデータ経路が、プログラム可能データ経路 148 によって示されている。この点において、和桁上げセレクト 120 (0) は、マルチプレクサ 118 (0) に、ベクトル入力 94 [1] と、ベクトル入力桁上げ 96 [0] とを制御ゲート 134 (0)、136 (0) に与えさせるように、ベクトル命令復号の結果として生成される。制御入力 138 (0) は、制御ゲート 134 (0)、136 (0) に、ベクトル入力 94 [1] と、ベクトル入力桁上げ 96 [1] とを圧縮器 108 (0) へ与えさせるように、桁上げ保存累算器 72 [0] のプログラム可能内部データ経路 80 [1] をプログラムするようにアクティブにされる。ベクトル入力 94 [1] およびベクトル入力桁上げ 96 [1] は、累算ベクトル出力和 112 (0) と、累算ベクトル出力桁上げ 114 (0) とを与えるように、桁上げ保存累算器 72 [0] の圧縮器 108 (0) によって、ベクトル入力 94 [0] およびベクトル入力桁上げ 96 [1] と累算される。累算ベクトル出力和 112 (0) およびシフトされた累算ベクトル出力桁上げ 117 (0) は、桁上げ保存加算器を設けるように、プログラム可能出力データ経路 74 [0] (図 3 も参照) 中で累算器出力サンプルセット 76 [0] として与えられる。シフトされた累算ベクトル出力桁上げ 117 (0) は、24 ビット累算器としての、累算ベクトル出力和 112 (1) と、シフトされた累算ベクトル出力桁上げ 117 (1) とを、プログラム可能出力データ経路 74 [1] (図 3 も参照) 中で累算器出力サンプルセット 76 [1] として与えるようにベクトル

40

50

入力 94 [1] およびベクトル入力桁上げ 96 [1] に累算されるために、マルチプレクサ 104 (1) によって圧縮器 108 (1) に与えられるように、プログラム可能データ経路 148 中に示される桁上げ保存累算器 72 [1] にも与えられる。

【 0074 】

[0086] 図 2 および図 3 の V P E 22 を含むが、それに限定されない、ベクトル処理のためであるとともに、本明細書において論じた概念および実施形態によるプログラム可能データ経路構成を有する V P E は、どのプロセッサベースのデバイス中に設けられても、統合されてもよい。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、固定ロケーションデータユニット、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 (P D A)、モニタ、コンピュータモニタ、テレビジョン、チューナー、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 (D V D) 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。

【 0075 】

[0087] この点において、図 10 は、プロセッサベースのシステム 150 の例を示す。この例では、プロセッサベースのシステム 150 は、各々が 1 つまたは複数のプロセッサまたはコア 154 を含む、1 つまたは複数の処理装置 (P U) 152 を含む。P U 152 は、非限定的例として、図 2 のベースバンドプロセッサ 20 であってよい。プロセッサ 154 は、非限定的例として、図 2 に挙げたベースバンドプロセッサ 20 のようなベクトルプロセッサであってよい。この点において、プロセッサ 154 は、図 2 の V P E 22 を含むが、それに限定されない V P E 156 も含み得る。P U 152 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 154 に結合されたキャッシュメモリ 158 を有し得る。P U 152 は、システムバス 160 に結合され、プロセッサベースのシステム 150 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し得る。よく知られているように、P U 152 は、システムバス 160 を介してアドレス、制御、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、P U 152 は、スレーブデバイスの例として、メモリコントローラ 162 にバストランザクション要求を通信することができる。図 10 には示されていないが、複数のシステムバス 160 が提供されてよく、ここにおいて、各システムバス 160 は異なる組織を構成する。

【 0076 】

[0088] 他のマスタおよびスレーブデバイスが、システムバス 160 に接続され得る。図 10 に示すように、これらのデバイスは、例として、メモリシステム 164 と、1 つまたは複数の入力デバイス 166 と、1 つまたは複数の出力デバイス 168 と、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 170 と、1 つまたは複数のディスプレイコントローラ 172 とを含み得る。メモリシステム 164 は、メモリコントローラ 162 によってアクセス可能なメモリ 165 を含み得る。入力デバイス 166 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 168 は、限定はしないが、オーディオ、ビデオ、他の視覚的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 170 は、ネットワーク 174 との間でデータの交換を可能にするように構成された任意のデバイスであり得る。ネットワーク 174 は、限定はしないが、ワイヤードまたはワイヤレスネットワーク、プライベートまたは公共ネットワーク、ローカルエリアネットワーク (L A N)、ワイドローカルエリアネットワーク (W L A N)、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であってよい。ネットワークインターフェースデバイス 170 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され得る。

【 0077 】

[0089] C P U 152 は、1 つまたは複数のディスプレイ 178 に送られる情報を制御

10

20

30

40

50

するため、システムバス160を介してディスプレイコントローラ172にアクセスするようにも構成され得る。ディスプレイコントローラ172は、1つまたは複数のビデオプロセッサ180を介して表示されるべき情報をディスプレイ178に送り、1つまたは複数のビデオプロセッサ180は、表示されるべき情報をディスプレイ178に好適なフォーマットに処理する。ディスプレイ178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。

【0078】

[0090]__本明細書で開示する二重電圧ドメインメモリバッファの実施形態とともに説明する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は、電子的なハードウェアとして、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り実行される命令として、またはこれら両方の組合せとして実装され得ることが、当業者にはさらに理解されよう。本明細書に記載されるアービタ、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チップに採用され得る。本明細書で開示するメモリは、どのタイプおよびサイズのメモリでもよく、所望されるどのタイプの情報も記憶するように構成され得る。この互換性を明確に示すために、様々な例示的な構成要素、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性に関して記載された。そのような機能性がどのように実装されるかは、特定の用途、設計選択、および/または全体的なシステムに課された設計制約に依存する。当業者は、各特定の用途について様々な様式で記載された機能性を実装することができるが、そのような実装決定は、本開示の範囲からの逸脱を引き起こすと解釈されるべきでない。

【0079】

[0091]__本明細書に開示された実施形態に関連して記載された、様々な例示の論理ブロック、モジュール、および回路は、本明細書に記載の機能を実施するよう設計された、プロセッサ、DSP、特定用途向け集積回路（ASIC）、FPGAまたは他のプログラマブル論理デバイス、個別ゲートまたはトランジスタロジック、個別ハードウェア構成要素、またはその任意の組合せを用いて実装または実施され得る。プロセッサはマイクロプロセッサであり得るが、代替として、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、またはステートマシンであり得る。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との組合せ、複数のマイクロプロセッサ、DSPコアと連携する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も実装され得る。

【0080】

[0092]__本明細書で開示する実施形態は、ハードウェアにおいて、およびハードウェアに記憶された命令において実施されてよく、たとえば、ランダムアクセスメモリ（RAM）、フラッシュメモリ、読取り専用メモリ（ROM）、電氣的プログラマブルROM（EPROM）、電氣的消去可能プログラマブルROM（EEPROM（登録商標））、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コンピュータ可読記憶媒体中に常駐し得る。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出し、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。代替形態において、記憶媒体はプロセッサと一体であってもよい。プロセッサおよび記憶媒体はASIC内に存在し得る。ASICは遠隔局中に常駐し得る。代替として、プロセッサおよび記憶媒体は、個別構成要素として遠隔局、基地局、またはサーバ中に存在してよい。

【0081】

[0093]__また、本明細書の例示的な実施形態のいずれかにおいて記載された動作ステップは、例および説明を提供するために記載されたものであることにやはり留意されたい。説明された動作は、例示された順序以外の多くの異なる順序で実施されてもよい。さらに、単一の動作ステップで説明される動作は、実際には、多くの異なるステップで実施され

得る。加えて、例示的な実施形態において議論された１つまたは複数の動作ステップは組み合わされ得る。フローチャート図に示した動作ステップは、当業者には容易に明らかになるように、多数の様々な修正を受け得ることを理解されたい。情報および信号は、多種多様な技術および技法のいずれかを使用して表され得ることも、当業者には理解されよう。たとえば、上記の説明全体にわたって言及され得るデータ、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、磁界もしくは磁性粒子、光場もしくは光子、またはそれらの任意の組合せによって表され得る。

【 0 0 8 2 】

[0094]__本開示の前の記載は、当業者が本開示を製造または使用することを可能にするように提供される。本開示に対する様々な変更形態が、当業者には容易に明らかとなり、本明細書に規定の一般的な原理が、本開示の精神または範囲から逸脱することなく、他の変形形態に適用され得る。したがって、本開示は、本明細書に記載される例および設計に限定することを意図しておらず、本明細書に開示される、原理および新規の特徴と一致する最も広い範囲が与えられるべきである。

以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

[C 1]

ベクトルデータのマルチモードベクトル処理を行うように構成されたベクトル処理エンジン (V P E) であって、

複数の入力データ経路の中のある入力データ経路中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセットを与えるように構成された入力読取り段階と、

複数のベクトル処理ブロックを備える少なくとも１つのベクトル処理段階と、前記複数のベクトル処理ブロックの中の各ベクトル処理ブロックは、

前記複数の入力データ経路の中の少なくとも１つの入力データ経路から、前記複数のベクトルデータ入力サンプルセットからの少なくとも１つのベクトルデータ入力サンプルセットを受信し、

前記少なくとも１つのベクトル処理段階によって実行されるベクトル命令に従って、前記ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて少なくとも１つのベクトル結果出力サンプルセットを与えるように、前記少なくとも１つのベクトルデータ入力サンプルセットを処理し、

複数の出力データ経路の中の少なくとも１つの出力データ経路中で前記少なくとも１つのベクトル結果出力サンプルセットを与えるように構成される、

前記複数のベクトル処理ブロックの各々から前記少なくとも１つのベクトル結果出力サンプルセットを受信するように構成された出力処理段階と、を備える V P E 。

[C 2]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも１つのベクトル処理段階によって実行される前記ベクトル命令に基づいて再構成されるように構成される、 C 1 に記載の V P E 。

[C 3]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも１つのベクトル処理段階によって実行される各ベクトル命令用に再構成されるように構成される、 C 2 に記載の V P E 。

[C 4]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも１つのベクトル処理段階によって実行される前記ベクトル命令の各クロック周期において再構成されるように構成される、 C 2 に記載の V P E 。

[C 5]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、プログラム可能入力データ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも１つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記プログラム可能入力データ経路構成に

10

20

30

40

50

基づいて前記複数の入力データ経路の中の前記少なくとも1つの入力データ経路から前記少なくとも1つのベクトルデータ入力サンプルセットを受信するように構成される、C 1 に記載のV P E。

[C 6]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、さらに、前記少なくとも1つのベクトル処理ブロック用のプログラム可能出力データ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記プログラム可能出力データ経路構成に基づいて前記複数の出力データ経路の中の前記少なくとも1つの出力データ経路中で前記少なくとも1つのベクトル結果出力サンプルセットを与えるように構成される、C 5 に記載のV P E。

10

[C 7]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、さらに、前記少なくとも1つのベクトル処理ブロック用のプログラム可能ベクトル処理ブロックデータ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記プログラム可能ベクトル処理ブロックデータ経路構成に基づいて前記少なくとも1つのベクトル結果出力サンプルセットを与えるために、前記少なくとも1つのベクトルデータ入力サンプルセットを処理するように構成される、C 6 に記載のV P E。

20

[C 8]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも1つのベクトル処理ブロック用のプログラム可能出力データ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記プログラム可能出力データ経路構成に基づいて前記複数の出力データ経路の中の前記少なくとも1つの出力データ経路中で前記少なくとも1つのベクトル結果出力サンプルセットを与えるように構成される、C 1 に記載のV P E。

30

[C 9]

前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成は、前記少なくとも1つのベクトル処理ブロック用のプログラム可能ベクトル処理ブロックデータ経路構成からなり、

前記複数のベクトル処理ブロックの各々は、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記プログラム可能ベクトル処理ブロックデータ経路構成に基づいて前記少なくとも1つのベクトル結果出力サンプルセットを与えるために、前記少なくとも1つのベクトルデータ入力サンプルセットを処理するように構成される、C 1 に記載のV P E。

40

[C 1 0]

前記少なくとも1つのベクトル処理ブロックは、少なくとも1つの乗算器ブロックおよび少なくとも1つの累算器ブロックのうちの少なくとも1つからなる、C 1 に記載のV P E。

[C 1 1]

前記少なくとも1つのベクトル処理段階は複数のベクトル処理段階からなる、C 1 に記載のV P E。

[C 1 2]

前記少なくとも1つのベクトル処理段階は、複数の乗算器ブロックからなる少なくとも1つの乗算ベクトル処理段階および複数の累算器ブロックからなる少なくとも1つの累算ベクトル処理段階からなり、

50

前記複数の乗算器ブロックの中の各乗算器ブロックは、

前記複数のベクトルデータ入力サンプルセットからの第1のベクトルデータ入力サンプルセットと第2のベクトルデータ入力サンプルセットとを、前記複数の入力データ経路の中の第1の入力データ経路および第2の入力データ経路から受信し、

前記少なくとも1つの第1のベクトル処理段階によって実行されるベクトル命令に従って、前記乗算器ブロック用のプログラム可能乗算データ経路構成に基づいて複数の乗算出力データ経路の中のある乗算出力データ経路中でベクトル乗算出力サンプルセットを与えるために、前記第1のベクトルデータ入力サンプルセットを前記第2のベクトルデータ入力サンプルセットに乗算するように構成され、

前記複数の累算器ブロックの中の各累算器ブロックは、

複数の乗算出力データ経路の中の第1の乗算出力データ経路および第2の乗算出力データ経路から、それぞれ、第1の乗算出力サンプルセットと第2の乗算出力サンプルセットとを受信し、

前記少なくとも1つの第2のベクトル処理段階によって実行されるベクトル命令に従って、前記累算器ブロック用のプログラム可能データ経路構成に基づいてベクトル累算結果サンプルセットを与えるために、前記第1の乗算出力サンプルセットを前記第2の乗算出力サンプルセットと累算し、

前記複数の複数の出力データ経路の中の前記出力データ経路中で前記ベクトル累算結果サンプルセットを与えるように構成される、C1に記載のVPE。

[C13]

前記少なくとも1つのベクトル処理ブロックは、最小の1つのベクトル結果出力サンプルセットをベクトルレジスタに記憶するように構成されない、C1に記載のVPE。

[C14]

各ベクトル処理ブロックは、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて異なるビット幅の前記複数のベクトルデータ入力サンプルセットを処理するようにさらに構成される、C1に記載のVPE。

[C15]

各ベクトル処理ブロックは、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて符号付きの少なくとも1つのベクトルデータ入力サンプルセットまたは符号なしの少なくとも1つのベクトルデータ入力サンプルセットを処理するようにさらに構成される、C1に記載のVPE。

[C16]

前記少なくとも1つのベクトル処理段階は、符号付きの演算命令からなる前記ベクトル命令を実行するように構成される、C1に記載のVPE。

[C17]

前記少なくとも1つのベクトル処理段階は、符号なし演算命令からなる前記ベクトル命令を実行するように構成される、C1に記載のVPE。

[C18]

ベクトルデータのマルチモードベクトル処理を行うように構成されたベクトル処理エンジン(VPE)であって、

複数の入力データ経路の中のある入力データ経路中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセットを与えるように構成された入力読取り段階手段と、

複数のベクトル処理手段を備える少なくとも1つのベクトル処理段階手段と、前記複数のベクトル処理手段の中の各ベクトル処理手段は、

前記複数の入力データ経路の中の少なくとも1つの入力データ経路から、前記複数のベクトルデータ入力サンプルセットからの少なくとも1つのベクトルデータ入力サンプルセットを受信し、

前記少なくとも1つのベクトル処理段階手段によって実行されるベクトル命令に従っ

10

20

30

40

50

て、前記ベクトル処理手段用のプログラム可能データ経路構成に基づいて少なくとも1つのベクトル結果出力サンプルセットを与えるように、前記少なくとも1つのベクトルデータ入力サンプルセットを処理し、

複数の出力データ経路の中の少なくとも1つの出力データ経路中で前記少なくとも1つのベクトル結果出力サンプルセットを与えるように構成される、

前記複数のベクトル処理手段の各々から前記少なくとも1つのベクトル結果出力サンプルセットを受信するように構成された出力処理段階手段と、を備えるVPE。

[C19]

ベクトル処理エンジン(VPE)中でベクトルデータを処理する方法であって、

入力処理段階中の複数の入力データ経路の中のある入力データ経路中で、ベクトルアレイの幅の複数のベクトルデータ入力サンプルセットを与えることと、

少なくとも1つのベクトル処理段階中の複数のベクトル処理ブロックにおいて、前記複数のベクトルデータ入力サンプルセットを処理することと、前記複数のベクトル処理ブロックの各々において、

前記複数の入力データ経路の中の少なくとも1つの入力データ経路から、前記複数のベクトルデータ入力サンプルセットからの少なくとも1つのベクトルデータ入力サンプルセットを受信することと、

前記少なくとも1つのベクトル処理段階によって実行されるベクトル命令に従って、前記ベクトル処理ブロック用のプログラム可能データ経路構成に基づいて少なくとも1つのベクトル結果出力サンプルセットを与えるように、前記少なくとも1つのベクトルデータ入力サンプルセットを処理することと、

複数の出力データ経路の中の少なくとも1つの出力データ経路中で前記少なくとも1つのベクトル結果出力サンプルセットを与えることと、を備える、

出力処理段階中の前記複数のベクトル処理ブロックの各々から前記少なくとも1つのベクトル結果出力サンプルセットを受信することと、を備える方法。

[C20]

前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に基づいて、前記複数のベクトル処理ブロックの各々用の前記プログラム可能データ経路構成を再構成することをさらに備える、C19に記載の方法。

[C21]

前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記プログラム可能入力データ経路構成に基づいて前記複数の入力データ経路の中の前記少なくとも1つの入力データ経路から前記複数のベクトルデータ入力サンプルセットからの前記少なくとも1つのベクトルデータ入力サンプルセットを受信することと、

前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能入力データ経路構成に基づいて前記少なくとも1つのベクトル結果出力サンプルセットを与えるように前記少なくとも1つのベクトルデータ入力サンプルセットを処理することと、を備える、C19に記載の方法。

[C22]

前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、プログラム可能出力データ経路構成に基づいて前記複数の出力データ経路の中の前記少なくとも1つの出力データ経路中で前記少なくとも1つのベクトル結果出力サンプルセットを受信することを備える、C19に記載の方法。

[C23]

前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用のプログラム可能ベクトル処理ブロックデータ経路構成に基づいて前記少なくとも1つのベクトル結果出力サンプルセットを与えるように前記少なくとも1つのベクトルデータ入力サンプルセットを処理することを備える、C19に記載の方法。

[C24]

最小の1つのベクトル結果出力サンプルセットをベクトルレジスタに記憶しないことをさらに備える、C 1 9に記載の方法。

[C 2 5]

前記少なくとも1つのベクトルデータ入力サンプルセットを処理することは、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて異なるビット幅の前記複数のベクトルデータ入力サンプルセットを処理することを備える、方法C 1 9。

[C 2 6]

前記少なくとも1つのベクトルデータ入力サンプルセットを処理することは、前記少なくとも1つのベクトル処理段階によって実行される前記ベクトル命令に従って、前記ベクトル処理ブロック用の前記プログラム可能データ経路構成に基づいて符号付きの少なくとも1つのベクトルデータ入力サンプルセットまたは符号なしの少なくとも1つのベクトルデータ入力サンプルセットを処理することを備える、C 1 9に記載の方法。

[C 2 7]

前記入力処理段階における前記複数の入力データ経路の中の前記入力データ経路中で、第2の幅の前記ベクトルアレイの第2の複数のベクトルデータ入力サンプルセットを与えることと、

前記少なくとも1つのベクトル処理段階中の前記複数のベクトル処理ブロックにおいて前記第2の複数のベクトルデータ入力サンプルセットを処理することと、前記複数のベクトル処理ブロックの各々において、

前記複数の入力データ経路の中の前記第2の少なくとも1つの入力データ経路から、前記第2の複数のベクトルデータ入力サンプルセットからの少なくとも1つのベクトルデータ入力サンプルセットを受信することと、

前記少なくとも1つのベクトル処理段階によって実行される第2のベクトル命令に従って、前記ベクトル処理ブロック用の第2のプログラム可能データ経路構成に基づいて第2の少なくとも1つのベクトル結果出力サンプルセットを与えるように、前記第2の少なくとも1つのベクトルデータ入力サンプルセットを処理することと、

前記複数の出力データ経路の中の前記少なくとも1つの出力データ経路中で前記第2の少なくとも1つのベクトル結果出力サンプルセットを与えることと、を備え、

前記出力処理段階中の前記複数のベクトル処理ブロックの各々から前記第2の少なくとも1つのベクトル結果出力サンプルセットを受信することと、をさらに備える、C 1 9に記載の方法。

[C 2 8]

前記第1の幅とは異なる前記第2の幅の前記ベクトルアレイの前記第2の複数のベクトルデータ入力サンプルセットを、前記入力処理段階における前記複数の入力データ経路の中の前記入力データ経路中で与えることを備える、C 2 7に記載の方法。

[C 2 9]

前記複数のベクトル処理ブロック中で前記複数のベクトルデータ入力サンプルセットを処理することは、前記複数の乗算器ブロック中で前記複数のベクトルデータ入力サンプルセットを処理することを備え、

前記複数のベクトル処理ブロック中で前記第2の複数のベクトルデータ入力サンプルセットを処理することは、前記複数の累算器ブロック中で前記第2の複数のベクトルデータ入力サンプルセットを処理することを備える、C 2 7に記載の方法。

[C 3 0]

前記複数の乗算器ブロックでの前記複数のベクトルデータ入力サンプルセットの処理は、前記複数の累算器ブロック中で前記第2の複数のベクトルデータ入力サンプルセットを処理することを備える、前記複数のベクトル処理ブロックでの前記第2の複数のベクトルデータ入力サンプルセットの前記処理と同時に起こる、C 2 9に記載の方法。

10

20

30

40

【図 1】

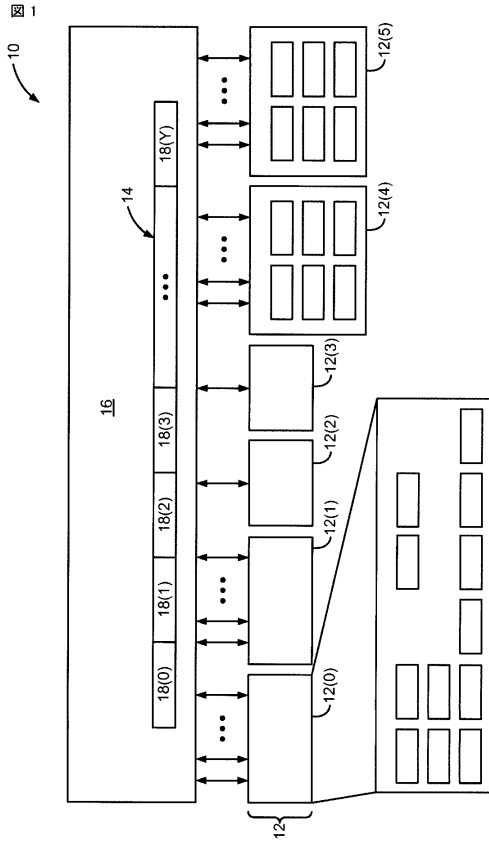


FIG. 1

【図 2】

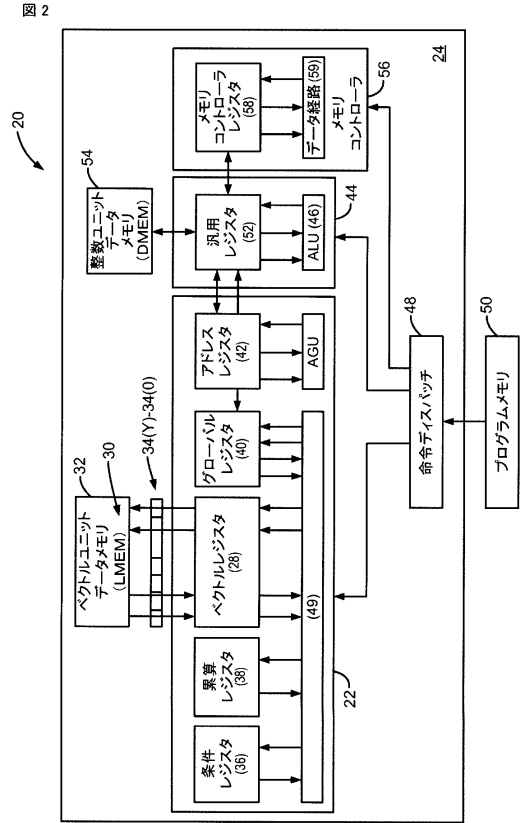


FIG. 2

【図 3】

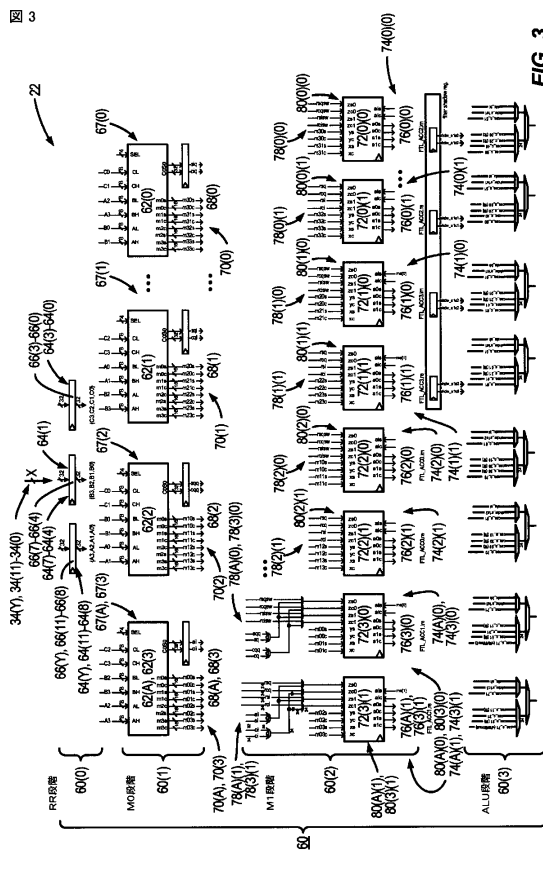


FIG. 3

【図 4 A】

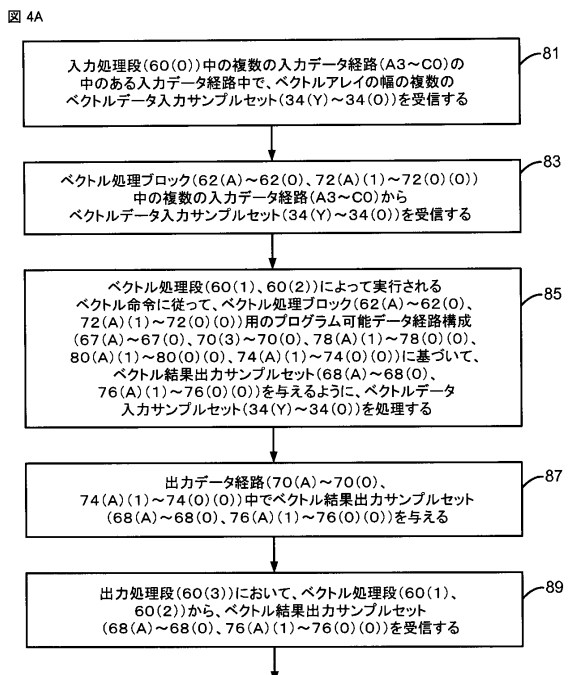


FIG. 4A

【図 4 B】

図 4B

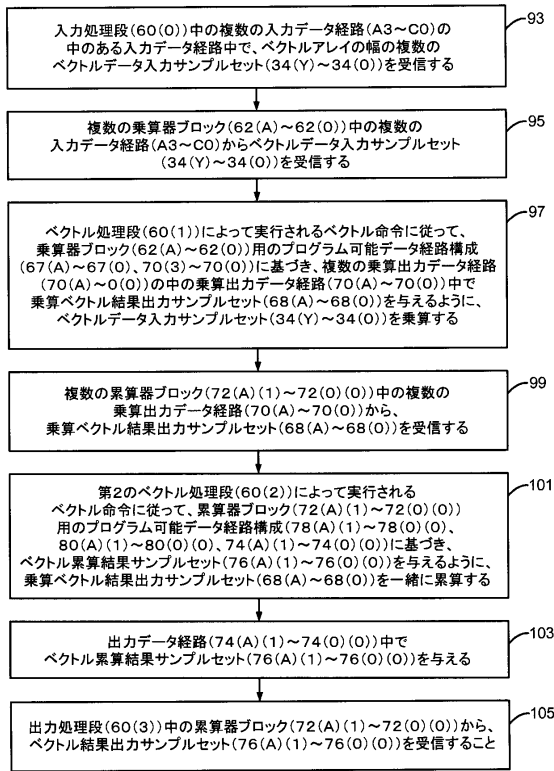


FIG. 4B

【図 5】

図 5

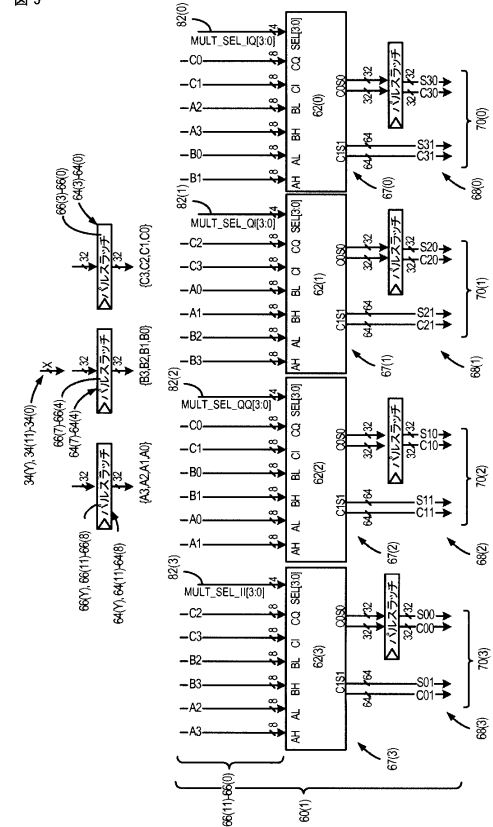


FIG. 5

【図 6】

図 6

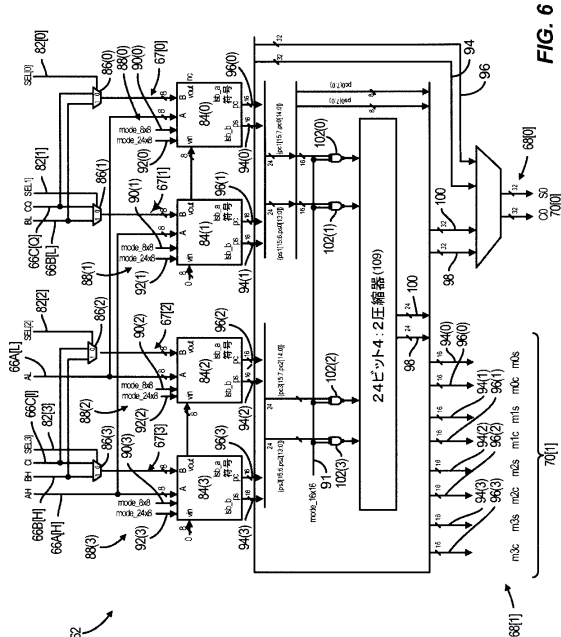


FIG. 6

【図 7】

図 7

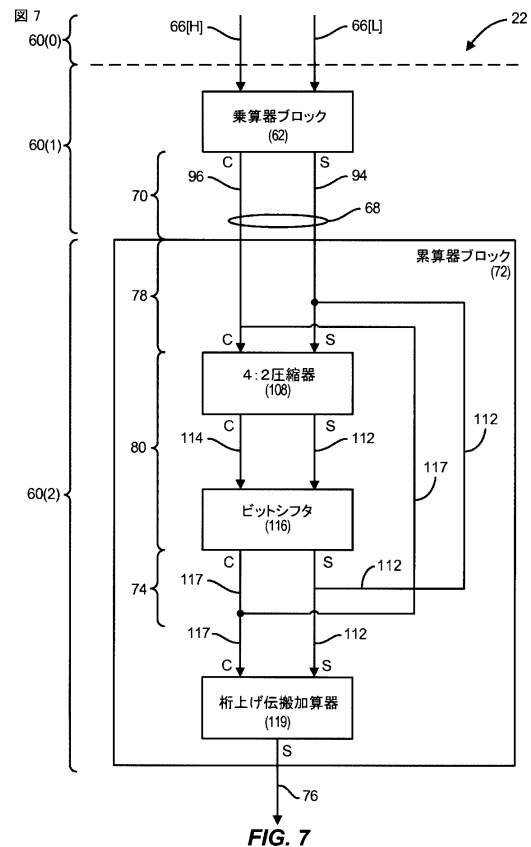


FIG. 7

【図 8】

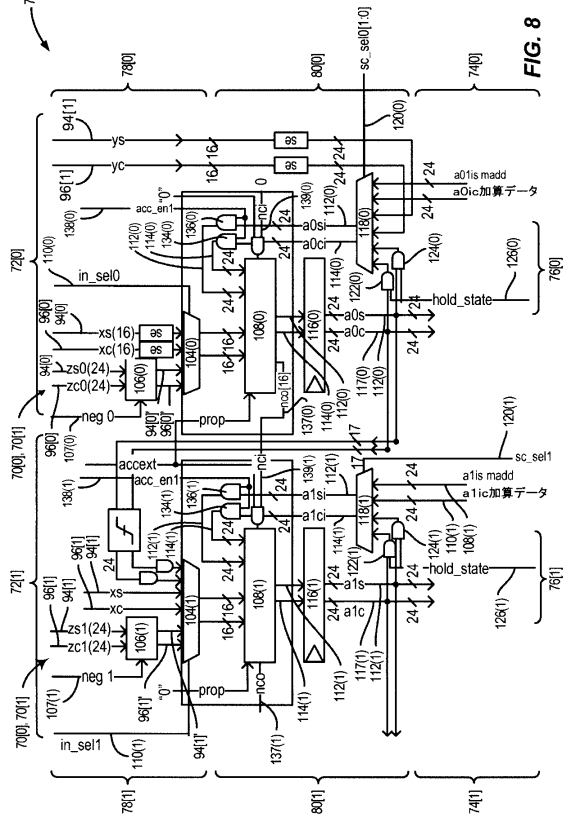


FIG. 8

【図 9 A】

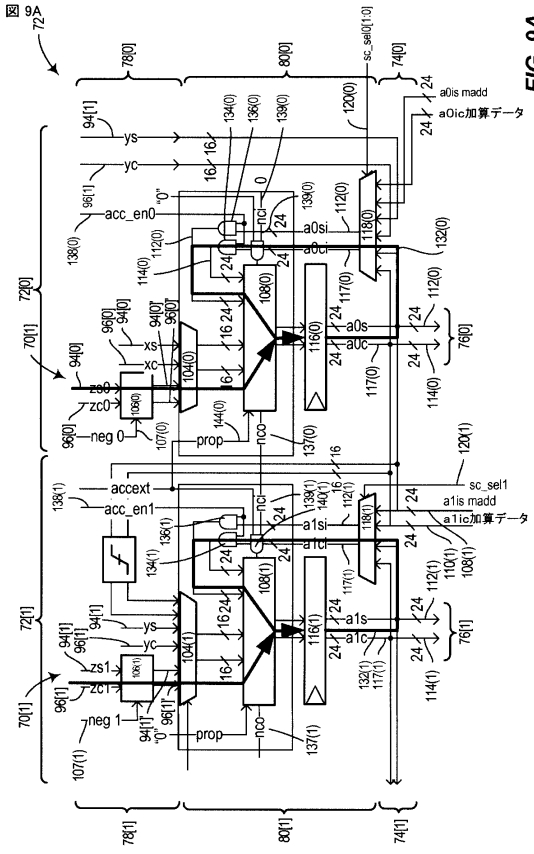


FIG. 9A

【図 9 B】

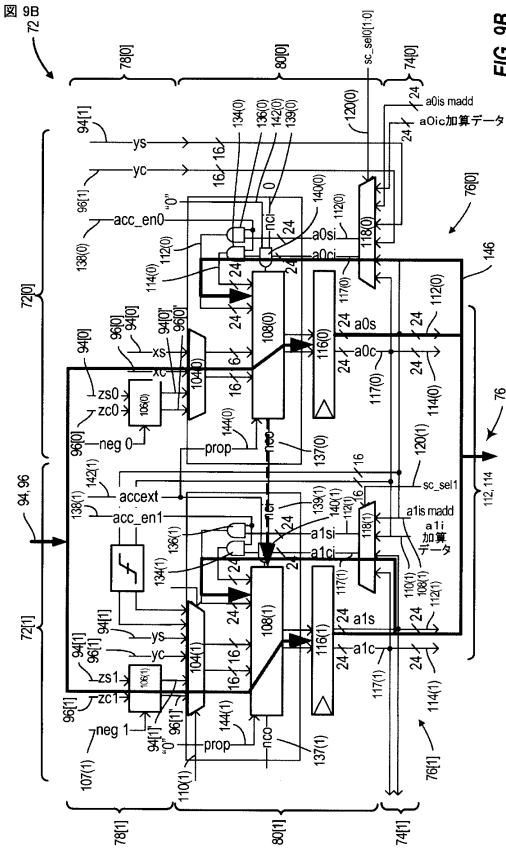


FIG. 9B

【図 9 C】

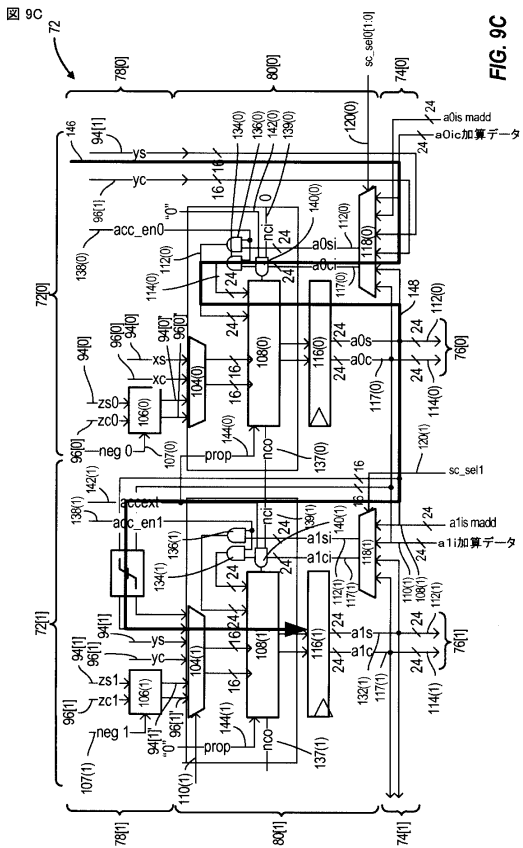
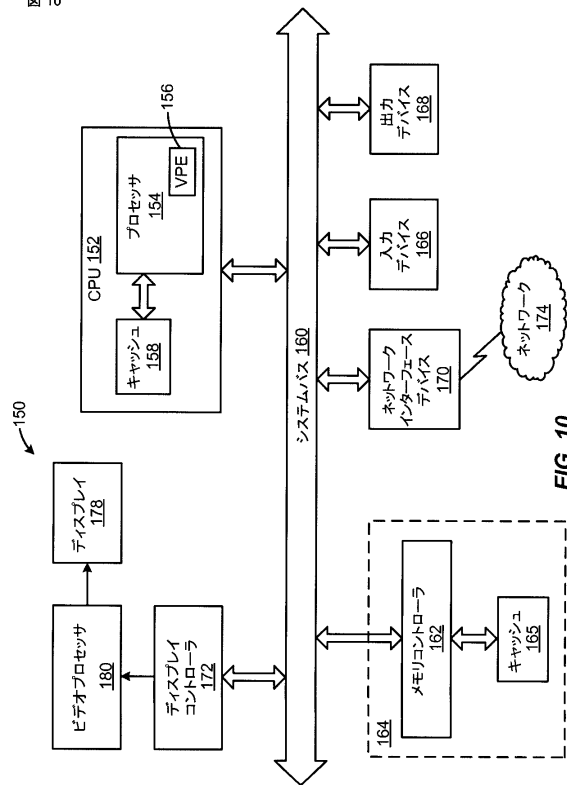


FIG. 9C

【図 10】

図 10



フロントページの続き

(72)発明者 カーン、ラヘル

アメリカ合衆国、カリフォルニア州 92121、サン・ディエゴ、モアハウス・ドライブ 57
75

審査官 田中 幸雄

(56)参考文献 特表2009-505215(JP,A)

特開2009-181509(JP,A)

特開2006-39790(JP,A)

米国特許出願公開第2007/0185953(US,A1)

(58)調査した分野(Int.Cl., DB名)

G06F 17/16

G06F 9/38