

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5358315号
(P5358315)

(45) 発行日 平成25年12月4日(2013.12.4)

(24) 登録日 平成25年9月6日(2013.9.6)

(51) Int. Cl.	F I
G06F 15/80 (2006.01)	G06F 15/80
G06F 9/38 (2006.01)	G06F 9/38 370A
G06F 9/30 (2006.01)	G06F 9/30 350F
G06F 9/34 (2006.01)	G06F 9/34 330
G06T 1/20 (2006.01)	G06F 9/30 370

請求項の数 2 (全 19 頁) 最終頁に続く

(21) 出願番号	特願2009-150019 (P2009-150019)	(73) 特許権者	000005326
(22) 出願日	平成21年6月24日 (2009.6.24)		本田技研工業株式会社
(65) 公開番号	特開2011-8416 (P2011-8416A)		東京都港区南青山二丁目1番1号
(43) 公開日	平成23年1月13日 (2011.1.13)	(74) 代理人	100064908
審査請求日	平成23年11月24日 (2011.11.24)		弁理士 志賀 正武
		(74) 代理人	100108578
			弁理士 高橋 詔男
		(74) 代理人	100146835
			弁理士 佐伯 義文
		(74) 代理人	100094400
			弁理士 鈴木 三義
		(74) 代理人	100107836
			弁理士 西 和哉
		(74) 代理人	100108453
			弁理士 村山 靖彦

最終頁に続く

(54) 【発明の名称】 並列計算装置

(57) 【特許請求の範囲】

【請求項1】

並列して演算処理を行う複数の演算プロセッサと、
前記演算プロセッサのそれぞれに制御信号を供給する命令実行制御部と、
を備え、
前記演算プロセッサが、
複数のデータ又は複数の演算結果を保持する記憶部と、
前記記憶部から読み出したデータに演算処理を施し、その結果を前記記憶部に供給する演算部と、
を備えるサブプロセッサを複数備え、
複数の前記サブプロセッサのうち少なくとも2つのサブプロセッサが同じ構造を有するものであり、同じ構成を有するサブプロセッサは同一構造サブプロセッサ群を構成し、

前記命令実行制御部は、
前記同一構造サブプロセッサ群に含まれるサブプロセッサに供給する制御信号を交換して供給する交換部
を備えることを特徴とする並列計算装置。

【請求項2】

前記サブプロセッサは、
前記演算部の一方の入力に接続され、書き込まれる情報を記憶するレジスタ部

を備え、

前記レジスタ部は、

通常時は前記演算部による演算結果を記憶し、記憶した演算結果を出力し、

前記同一構造サブプロセッサ群は、レジスタ参照部を備え、

前記レジスタ参照部は、前記同一構造サブプロセッサ群に含まれるそれぞれの前記サブプロセッサが備えるそれぞれの前記レジスタ部からの出力を、前記サブプロセッサの前記演算部の他方の入力に入力データを供給するレジスタのうち、少なくとも一部のレジスタからの入力として扱う際において、前記レジスタ部と前記入力データを供給するレジスタとの対応付けを変更することができるものであり、

前記命令実行制御部は、前記同一構造サブプロセッサ群の各サブプロセッサに供給する前記制御信号を交換する際には、交換前に、前記入力データを供給するレジスタのそれぞれに対応付けられていた、それぞれの前記レジスタ部に代えて、当該レジスタ部が属する前記サブプロセッサに交換前に供給されていた前記制御信号を交換後に供給されることとなる前記サブプロセッサが備える前記レジスタ部を、前記入力データを供給するレジスタにそれぞれ新たに対応付けるように、前記レジスタ参照部を制御するものであることを特徴とする請求項1記載の並列計算装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、並列計算装置に関する。

【背景技術】

【0002】

大容量の演算処理を行う電子計算装置には、画像処理用途に用いられるものがある。そのような電子計算装置が、リアルタイムに演算処理を行うシステムを例に示す。

自動車に搭載されたビデオカメラによって得られる映像に基づいて画像処理を行って、検出された結果をドライバーに示すことにより安全運転を補助するシステムがある。このようなシステムに用いられる電子計算装置（画像処理プロセッサ）は、ビデオカメラと一緒に自動車に搭載され、ビデオカメラからの映像（例えば640画素×480画素、10フレーム/秒）を分析する。例えば、歩行者検出をするような用途の分析に用いられる画像処理プロセッサには、100GOPS（Giga（ 10^9 ）operations per second）以上の演算能力が必要とされている。歩行者の検出精度を高めたり、或いは遠くの歩行者を検出したりするには、ビデオカメラが出力する映像信号から示される細部の状況を検出して、判定する必要がある。細部の状況を判定できる解像度を基準にすると、ビデオカメラの解像度を高め、映像信号を構成する画素数を増やす必要がある。それに伴って画像分析に必要な演算量は扱う画素数にほぼ比例して増大する。そのため、より高速な画像処理プロセッサが必要になる。

また、安全性を高めるには歩行者をより短時間で検出しなければならず、それには検出周期を短くし、1秒当たりに処理するフレーム数を増やす必要がある。フレーム数を増やすと、画像分析に必要な演算量は扱うフレーム数に比例して増大するので、画像処理プロセッサの更なる高速化が要求されている。

【0003】

一方で、車載環境では供給できる電力に限りがあるので、低消費電力（数ワット以下）で動作することが必須であり、消費電力の増大を無視してプロセッサを高速化することはできない。

近年、パーソナルコンピュータ（PC）などに使われる、汎用プロセッサ（汎用CPU）の性能が飛躍的に向上した（10Gflops程度）。それでも上記用途に必要とされる性能と比較すると、性能不足であり、かつ消費電力が数10W（ワット）にもなるので車載用途には使えない。

PC用の高性能画像処理専用のプロセッサとして、例えば米国nVIDIA社からGPGPU（General Purpose computing on Graphics Processing Unit）と呼ばれる半導体装置が市販さ

10

20

30

40

50

れている。この半導体装置に搭載されるプロセッサの公称演算性能は数100Gflosであり十分と推定されるが、消費電力が多大（数10W（ワット））であり、また画像処理専用のプロセッサであるが映像データを生成する用途向きであり、歩行者検出のような用途には不向きである。

【0004】

消費電力が数ワットと少ないプロセッサとしては、例えば、株式会社ルネサステクノロジ社の組み込み用プロセッサSH4シリーズなどがあるが、演算性能が2Gflops未満と低過ぎる。

これらの状況から、低消費電力で非常に高い演算処理性能を達成する計算機アーキテクチャを有する画像処理プロセッサが開発されている。その一例として、車載用画像処理を目的とする画像処理プロセッサがある。その画像処理プロセッサは、動作周波数を低く抑えた消費電力が非常に小さい要素プロセッサエレメント（要素PE）を、1個のLSIに多数個集積して並列処理する構成となっている。例えば、日本電気株式会社が開発したIMPCARなどである（例えば、非特許文献1）。

【先行技術文献】

【非特許文献】

【0005】

【非特許文献1】岡崎、昭倫、古賀、肥田野、「車載組み込み用画像認識プロセッサIMPCAR」、NEC技報 Vol. 60 No. 2/2007 p17 - 20,2007 .

【非特許文献2】Kyo, S.; Okazaki, S.; Arai, T., "An Integrated Memory Array Processor Architecture for Embedded Image Recognition Systems", Computer Architecture, 2005. ISCA apos;05. Proceedings. 32nd International Symposium on Volume , Issue , 4-8 June 2005 p134 - 145,2005 .

【発明の概要】

【発明が解決しようとする課題】

【0006】

ところで、一般的に、並列処理プロセッサを実際に動作させた時に得られる実効性能は、そこに内蔵されている演算器（ALU）の総数と、それらの動作周波数との掛け算で得られる公称値に比べてかなり低い。実効性能が低下する主な原因は、演算器が必要とするデータを適当なタイミングで供給できなかつたり、或いは演算結果を直ぐに格納できなかつたりして演算器に遊びが生じるからである。並列処理プロセッサを実際に動作させたときに得られる実効性能を高めるには途切れなく演算器へデータを供給し、同時に演算結果を迅速に適当な場所へ格納できるようにする構成が必要とされる。

【0007】

多数の要素PEを並列動作させる方式にはSIMD（Single Instruction Multi Data）型、MIMD（Multi Instruction Multi Data）型及びVLIW（Very Long Instruction Word）型などのアーキテクチャがあるが、車載用画像処理用途では低コストと高いパフォーマンス性を併せ持つ必要があることから、SIMD型にVLIW型を組み合わせた前述のIMPCARの方式が優れている。

SIMD型画像処理プロセッサの構成例を図12に示す。SIMD型並列計算装置では、同じ命令で全ての要素PEが同時に動作するという構造上、各要素PEが一斉にデータ読み込んだり、一斉にデータを出力したりする。例えば、1つのLSIの中に128個の要素PEが在り、これらが100MHz（メガヘルツ）のクロックで動作して一斉に8ビットのデータを1つ要求したとすると、瞬間的に必要なデータ転送速度は、 $100\text{MHz} \times 1\text{B} \times 128 = 12.8\text{GB}$ （ギガバイト）/秒である。これは実現困難なデータ転送速度ではないが、外部回路も含めてコスト上昇と消費電力の増大を招いてしまう。そこで、図の構成では要素PE毎にローカルメモリを設け、それらと要素PEとの間を専用バスで接続している。このような構成にすることで、各要素PEとローカルメモリ間で必要なデータ転送速度は最大100MB（メガバイト）/秒になる。

【0008】

一方、処理すべきデータは先ずLSI外部の素子から供給され、処理結果はLSI外部へと取り出されなければならない。先に上げた例（640画素×480画素、10フレーム/秒）で入力画像が白黒で8bit/画素であるとする、入力データのデータ転送速度は $640 \times 480 \times 10 / s \times 1B = 3.1MB/秒$ であり、入力データ線を8本使えば3.1MHzのクロックで入力できる。この程度のデータ入力速度は容易に達成できる。また、一般的に画像処理では、出力データのデータ量は入力データと同程度なので、これも容易に出力できる。

【0009】

また、非特許文献1の技術によれば、IMPCARの要素PEは、4つのサブプロセッサ（以後SPE）を持つVLIW型である（図13参照）。4つのサブPEは機能分化している。すなわち、ローカルメモリとのデータ入出力又は加算を担当するサブPE（図中MEM/Add）、論理演算を担当するサブPE（同Logic）、算術論理演算を担当するサブPE（同ALU）、及び乗算を担当するサブPE（同Multiplier）である。4つのサブPEに共通に28個のジェネラルレジスタ（GR）を備える。この構成では次に示す3つの問題点がある。

一つ目の問題は、ローカルメモリとのデータ入出力が1つのサブPEでしか行えないので、メモリアクセスが集中するとデータ入出力が間に合わなくなったり（メモリデータのバンド幅の不足）、逆にメモリアクセスや加算が殆ど無い時は、このサブPEが遊んでしまったりする点である。

二つ目の問題は、1つのサブPEを乗算に固定しているが、画像処理において乗算を行う頻度はそれほど多くなく、このサブPE（同Multiplier）が遊んでしまう可能性が高いことである。これらの問題により、常に4つのサブPEをアクティブ（遊びが無い状態）に保ち続けることは難しい。

三つ目の問題は、28個のGRにある。

非特許文献2における5.3節によれば、サブPEで実行される命令のオペランドは、ソース2、ディスティネーション1の3オペランドである。GRは28個在るのでオペランド指定にそれぞれ5ビット必要になり、合計では15ビットになる。したがって、VLIW構成とするために4つのサブPEを合わせると、60ビット必要であり、命令コードを格納するメモリの容量が大きくなって、コストが上昇する要因になってしまう。

【0010】

発明は、上記問題を解決すべくなされたもので、その目的は、実効効率の高い並列計算装置を提供することを目的とする。

【課題を解決するための手段】

【0011】

上記問題を解決するために、請求項1に記載した発明は、並列して演算処理を行う複数の演算プロセッサ（例えば、実施の形態における演算プロセッサPE2）と、前記演算プロセッサのそれぞれに制御信号を供給する命令実行制御部（例えば、実施の形態における命令実行制御プロセッサPE-I3）と、を備え、前記演算プロセッサが、複数のデータ又は複数の演算結果を保持する記憶部（例えば、実施の形態におけるローカルメモリ11A～11D）と、前記記憶部から読み出したデータに演算処理を施し、その結果を前記記憶部に供給する演算部（例えば、実施の形態におけるALU12A～12C及び12DA）と、を備えるサブプロセッサ（例えば、実施の形態におけるサブプロセッサ（SPE）2A～2D）を複数備え、複数の前記サブプロセッサのうち少なくとも2つのサブプロセッサが同じ構造を有するものであり、同じ構成を有するサブプロセッサは同一構造サブプロセッサ群（例えば、実施の形態におけるサブプロセッサグループSPE2G）を構成し、前記命令実行制御部は、前記同一構造サブプロセッサ群に含まれるサブプロセッサに供給する制御信号を交換して供給する交換部（例えば、実施の形態における交換部33A～33C）を備えることを特徴とする並列計算装置である。

【0012】

請求項2に記載した発明は、前記サブプロセッサは、前記演算部の一方向の入力に接続され、書き込まれる情報を記憶するレジスタ部（例えば、実施の形態におけるAcc13A～13D）を備え、前記レジスタ部は、通常時は前記演算部による演算結果を記憶し、

10

20

30

40

50

記憶した演算結果を出力し、前記同一構造サブプロセッサ群は、レジスタ参照部（例えば、実施の形態におけるレジスタ参照部 15A～15C）を備え、前記レジスタ参照部は、前記同一構造サブプロセッサ群に含まれるそれぞれの前記サブプロセッサが備えるそれぞれの前記レジスタ部からの出力を、前記サブプロセッサの前記演算部の他方の入力にデータを供給するレジスタのうち、少なくとも一部のレジスタからの入力として扱う際において、前記レジスタ部と前記データを供給するレジスタとの対応付けを変更することができるものであり、前記命令実行制御部は、前記同一構造サブプロセッサ群の各サブプロセッサに供給する前記制御信号を交換する際には、交換前に、前記データを供給するレジスタのそれぞれに対応付けられていた、それぞれの前記レジスタ部に代えて、当該レジスタ部が属する前記サブプロセッサに交換前に供給されていた前記制御信号を交換後に供給されることとなる前記サブプロセッサが備える前記レジスタ部を、前記データを供給するレジスタにそれぞれ新たに対応付けるように、前記レジスタ参照部を制御するものであることを特徴とする請求項 1 に記載の並列計算装置である。

10

【発明の効果】

【0013】

請求項 1 から請求項 2 に記載した発明によれば、並列計算装置は、並列して演算処理を行う複数の演算プロセッサを備える。演算プロセッサでは、命令実行制御部がそれぞれに制御命令を供給する。サブプロセッサは、記憶部が複数のデータ又は複数の演算結果を保持する。演算部は、記憶部から読み出したデータに演算処理を施し、その結果を記憶部に供給する。

20

また、複数のサブプロセッサのうち少なくとも 2 つのサブプロセッサは同じ構造を有する。それらの同じ構造を有するサブプロセッサは、同一構造プロセッサ群を形成する。また、命令実行制御部は、交換部が同一構造サブプロセッサ群に含まれるサブプロセッサに供給する制御信号を交換して供給する。

これにより、記憶部に記憶されたデータを交換したり、参照するデータを切り換えて同様の処理をしたりするためにオペランドの異なる類似のプログラムを用意することもなく、データと処理の組み合わせを交換することができる。

【図面の簡単な説明】

【0014】

【図 1】本発明の第 1 実施形態を示す概略ブロック図である。

30

【図 2】第 1 実施形態における PE 2 の構成を示すブロック図である。

【図 3】第 1 実施形態における画像データを入力する場合のタイミングチャートである。

【図 4】第 1 実施形態における PE 2 から外部へデータを出力する場合のタイミングチャートである。

【図 5】第 1 実施形態における演算処理に入力される映像信号の例を示す図である。

【図 6】第 1 実施形態におけるアドレスレジスタの構成を示す図である。

【図 7】第 1 実施形態におけるローカルメモリへの画像データのマッピングを示す図である。

【図 8】第 1 実施形態における PE - I 3 を示す概略ブロック図である。

【図 9】第 1 実施形態における交換部 31 における命令選択状態を示す図である。

40

【図 10】第 1 実施形態におけるレジスタ参照部の動作を示す図である。

【図 11】第 2 実施形態における PE 2 の構成を示すブロック図である。

【図 12】従来技術による SIMD 型画像処理プロセッサの構成例のブロック図である。

【図 13】従来技術による要素演算プロセッサの構成例を示す。

【発明を実施するための形態】

【0015】

(第 1 実施形態)

図を参照し、並列計算装置の一実施形態について示す。

図 1 は、本発明の実施形態を示す概略ブロック図である。

この図に示される並列計算装置 1 は、複数の演算プロセッサ（「要素 PE」ともいう

50

)によって並列演算処理を行う。本実施形態の詳細な説明に先立ち、並列計算装置1の構成概要について説明する。

並列計算装置1における演算プロセッサ(PE)2-0~2-106(まとめて「演算プロセッサ(PE)2」という。)、及びPE2のそれぞれに制御命令を供給する命令実行制御プロセッサ(PE-I)3を含んで構成されている。

また、並列計算装置1は、入出力プロセッサ(IOP)4、命令メモリ5、データ入力シフトレジスタ6、データ出力シフトレジスタ7及び外部メモリ9を備える。

【0016】

演算プロセッサ2は、それぞれが4個のサブプロセッサ(SPE)2A~2Dを有する。

10

SPE2A~2Dは、それぞれが異なる命令を実行するVLIW(Very Long Instruction Word)型の構成を有している。それぞれのPE2では、SPE2A~2Dが組み合わされた同じ構成で形成される。また、全てのPE2が有する107個のSPE2Aは、SIMD(Single Instruction Multi Data)型で構成され、全てのSPE2Aで同一の命令を実行する。また、SPE2B、SPE2C、SPE2Dについても同様に、それぞれがSIMD型で構成される。

それらのSPE2A~2Dは、構成の異なるSPEを組み合わせで構成される。演算プロセッサ2における演算処理を行うSPE2AからSPE2Cと、演算処理のほか外部との入出力などの構成を備えるSPE2Dの組み合わせを例にして説明する。なお、同じ構成を有するSPE2AからSPE2Cの組み合わせをサブプロセッサグループ2G

20

【0017】

PE-I3は、PE2の命令の実行順序を制御する。

PE-I3は、PE2のプログラムにおけるループ処理やサブルーチンコールなどの条件分岐を必要とする処理の制御を行う。PE-I3及びPE2の命令をアセンブラプログラムで記述すると、SPE2AからSPE2D及びPE-I3の命令の5命令を並列に実行するVLIW型の命令として記述される。

【0018】

IOP4は、PE2外部との入出力を行う。

SIMD+VLIW型の並列計算装置1で実行されるプログラムコードは、計算開始前にIOP4によって外部メモリ9から予め読み込まれ、PE-I3に付属する命令メモリ5に書き込まれる。その後、IOP4がPE-I3に計算開始信号を送ると、PE-I3は命令メモリから自分自身で実行する命令と、SPE2AからSPE2Dで実行すべき4個の命令とを読み出して演算を開始する。演算対象のデータはIOP4によって外部から取り込まれ、データ入力シフトレジスタ6を介して、PE2-0からPE2-106のプロセッサに分割して転送される。また、演算結果は、データ出力シフトレジスタ7を介してIOP4によって各PE2から読み出され、外部メモリ9へ転送され、PE2の外部に出力される。

30

【0019】

データ入力シフトレジスタ6及びデータ出力シフトレジスタ7は、並列計算装置1のPE2で演算を行うデータの入力及び出力を行う。データ入力シフトレジスタ6は、IOP4を介して入力されたデータをシフトレジスタの機能により、シリアルで順次入力されるデータに対応するPE2が配置されている位置までシフトする。また、データ出力シフトレジスタ7は、各PE2でそれぞれで演算処理された結果を、PE2が配置されている位置からIOP4までシフトして、IOP4を介して出力する。

40

このように、複数のPE2は、並列して演算処理及び入出力処理を行うことができる。

【0020】

図2は、本発明の並列計算装置1におけるPE2の構成を示すブロック図である。

この図に示される並列計算装置1は、PE2-kと、PE2-kを挟んで配置されるPE2-(k-1)とPE2-(k+1)、PE-I3、IOP4、命令メモリ5、データ

50

入力シフトレジスタ 6、データ出力シフトレジスタ 7 が示される。図 1 と同じ構成には同じ符号を付す。

【 0 0 2 1 】

PE 2 - k は、隣接する PE 2 - (k - 1) と PE 2 - (k + 1) と同じ構成を有することから、PE 2 - k (以下、隣接関係を特に示さない限り「PE 2」と示す。)を参照して構成を示す。

PE 2 は、4 個のサブプロセッサ (SPE) 2 A ~ 2 D のほか、レジスタ 2 M を有する。また、SPE 2 A から SPE 2 C は、同じ構成を有し、サブプロセッサグループ SPE 2 G を形成する。

【 0 0 2 2 】

レジスタ 2 M は、演算に必要なデータの一時記憶用の記憶領域である。レジスタ 2 M は、内部に 1 2 個のレジスタ (R4 ~ R15) を備える。レジスタ 2 M は、各 SPE から参照され、また書き込みが行われる。

【 0 0 2 3 】

SPE 2 A は、ローカルメモリー (LM) 1 1 A、ALU 1 2 A、Acc 1 3 A、セクタ (Sel) 1 4 A 及びレジスタ参照部 (Sel) 1 5 A を備える。同様に、SPE 2 B は、ローカルメモリー (LM) 1 1 B、ALU 1 2 B、Acc 1 3 B、セクタ (Sel) 1 4 B 及びレジスタ参照部 (Sel) 1 5 B を備える。SPE 2 C は、ローカルメモリー (LM) 1 1 C、ALU 1 2 C、Acc 1 3 C、セクタ (Sel) 1 4 C 及びレジスタ参照部 (Sel) 1 5 C を備える。

【 0 0 2 4 】

まず、SPE 2 A ~ 2 C の構成について SPE 2 A を代表して説明する。

SPE 2 A においてローカルメモリー 1 1 A は、例えば、512 x 8b (ビット) の構成を有するメモリであり、演算処理の入力データ及び演算結果を記録する。

Acc 1 3 A は、ALU 1 2 A の演算結果が書き込まれ、また、ALU 1 2 A ~ 1 2 D から参照されるアキュムレータである。

セクタ 1 4 A は、ALU 1 2 A への入力を選択する。

ALU 1 2 A は、入力されるデータに基づいて所定の演算処理を行う。ALU 1 2 A の一方の入力は、Acc 1 3 A からのデータが供給される。ALU 1 2 A の他方の入力は、セクタ 1 4 A により選択されたデータが供給される。

セクタ 1 4 A によって選択されるデータは、レジスタ 2 M、Acc 1 3 A と Acc 1 3 B と Acc 1 3 C と Acc 1 3 D、ローカルメモリー 1 1 A に記憶されたデータのいずれかである。

【 0 0 2 5 】

ALU 1 2 A による演算結果は通常は Acc 1 3 A に書き込まれるが、Acc 1 3 A のデータをレジスタ 2 M へ転送する命令を使って、レジスタ 2 M 内のレジスタ R4 ~ R15 のいずれかを選択して演算結果を書き込む。また、Acc 1 3 A のデータをメモリー 1 1 A へ転送する命令を使って、メモリー 1 1 A 内の記憶領域のいずれかを選択して書き込む。

レジスタ参照部 1 5 A は、PE - I 3 からの制御信号により、セクタ 1 4 A に出力するデータを選択する。選択できるデータは、Acc 1 3 A ~ Acc 1 3 C のデータである。セクタ 1 4 A における選択条件を変更することなく、レジスタ参照部 1 5 A の選択を変更することによって、Acc 1 3 A ~ Acc 1 3 C のデータを参照することが可能になる。

SPE 2 B ~ SPE 2 C は、SPE 2 A の構成と対応する構成を有する。

【 0 0 2 6 】

続いて、SPE 2 A ~ SPE 2 C と異なる構成を有する SPE 2 D について示す。

SPE 2 D は、ローカルメモリー 1 1 D、演算部 1 2、Acc 1 3 D 及びセクタ 1 4 D、データ入力バッファ (SIN_reg) 1 6、データ出力バッファ (SOUT_reg) 1 7、PE 出力レジスタ L (PE-out-L) 1 8、PE 出力レジスタ R (PE-out-R) 1 9 を備える。この

10

20

30

40

50

ように、S P E 2 Dは、他のS P E 2 A ~ S P E 2 Cと異なる独自の構成を有しており、レジスタ参照部を備えていないことも、前述のS P E 2 A ~ S P E 2 Cとの相違点になる。

【 0 0 2 7 】

S P E 2 Dが有する構成を、S P E 2 Aと対比して示す。

S P E 2 Dにおけるローカルメモリー11Dは、ローカルメモリー11Aに相当する。

演算部12Dは、A L U 1 2 D A、乗算器12DM、セクタ12DSを備える。A L U 1 2 D Aは、A L U 1 2 Aに相当し、出力はセクタ12DSを介してA C C 1 3 Dに入力される。乗算器12DMは、A L U 1 2 D Aと同じ入力信号が設定され、入力されるデータの乗算を行う。セクタ12DSは、A L U 1 2 D A又は乗算器12DMの演算結果を選択的にA c c 1 3 Dに出力する。A c c 1 3 Dは、A c c 1 3 Aに相当する。

セクタ14Dは、セクタ14Aと同じレジスタ2M、A c c 1 3 AとA c c 1 3 BとA c c 1 3 CとA c c 1 3 D、ローカルメモリー11Dに記憶されたデータのほかに、データ入力バッファ16を介して入力されるデータ入力シフトレジスタ6からの入力データ、隣接するP E 2 - (k - 1)やP E 2 - (k + 1)からの入力データがある。

【 0 0 2 8 】

データ入力バッファ16は、P E 2に入力されたデータを各P E 2にセットするデータ入力シフトレジスタ6からの入力を一時的に保持する。データ出力バッファ17は、演算された結果をP E 2から出力する際に、データ出力シフトレジスタ7に出力するデータを一時的に記憶し、データ出力シフトレジスタ7によって読み出しが行われる。

P E出力レジスタL18とP E出力レジスタR19は、隣接するP E 2に出力するデータを一時的に記憶する。

P E出力レジスタL18やP E出力レジスタR19に記憶されたデータは、隣接するP E 2におけるS P E 2 Dから参照される。

P E出力レジスタL18は、図の左隣に隣接するP E 2 - (k - 1)にデータを出力する。P E出力レジスタR19は、図の右隣に隣接するP E 2 - (k + 1)にデータを出力する。

このように、S P E 2 Dは、乗算を含んだ演算処理とP E 2外部とのデータ交換処理を選択的に処理できる。

【 0 0 2 9 】

以上に示したS P E 2 A ~ 2 Dにより、ローカルメモリー11A ~ 11Dは、各S P Eに分割して配置される。この図に示した例では、その容量は512 x 8b (ビット)の単位で4分割されているものとする。ローカルメモリー11A ~ 11Dの合計容量は、分割せずにとまどめて配置される2k x 8b (ビット)と同じ記憶容量であるが、分割して配置したことにより、それぞれのS P Eで同時にアクセスできるようになり、P E 2におけるローカルメモリーに対するデータ転送能力は、4倍になる。この構成により、全てのS P Eが待ち時間なしでメモリアクセスできるようになり、さらに、マルチポートメモリーを使う場合に生じるアクセスの衝突問題を回避できる。ただし、これらの利点と引き換えに、2k x 8b (ビット)のメモリーを1つ配置する場合よりも、ローカルメモリー11A ~ 11Dの配置に必要な面積は若干増大する。

【 0 0 3 0 】

ローカルメモリー11A ~ 11Dは、それぞれ内部にアドレスレジスタを備えている。S P E 2 A ~ 2 Dにおいて、ローカルメモリー11A ~ 11Dをアクセスする場合は、先ずアドレス情報を図示されないアドレスレジスタにセットし、次にメモリーへの書き込みか、或いはメモリーからの読み出しを行う。ローカルメモリー11A ~ 11Dにアクセスする度にアドレスレジスタを設定し直すのでは効率が悪くなる。そこで、通常メモリアクセス命令とは別に、アクセス後に自動的にアドレスレジスタに設定されたアドレスが1増える命令と、1減る命令とを用意する。このようなアドレスレジスタを用いることにより、メモリアクセス時に必要な命令数を減らすことができ、画像処理のようにメモリアクセスが多くなる処理では、メモリアクセスの効率を改善することができる。

【 0 0 3 1 】

また、全ての S P E には、A L U 1 2 A ~ 1 2 D A をそれぞれ備えているので、どの S P E においても算術論理演算処理ができる。更に、S P E 2 D には乗算器 (M u l) 1 2 D M を備えており、乗算処理が可能である。この構成により、各 S P E はそれぞれ算術論理演算処理を並列に行えることから、コンパイラによる命令割付の自由度が大幅に向上する。

【 0 0 3 2 】

また、各 S P E 2 は、ジェネラルレジスタ方式ではなく、アキュムレータ方式である。即ち、それぞれの A L U 1 2 A ~ A L U 1 2 D A の一方の入力は、アキュムレータ (A c c) に固定され、また、A L U 1 2 A ~ A L U 1 2 D A の出力先はそれぞれの A c c 1 3 A ~ 1 3 D である。A L U 1 2 A ~ A L U 1 2 D A の他方の入力だけが、命令に応じてデータの参照先を指定できる。

P E 2 は、演算に必要なデータの一時記憶用に用いられるレジスタ 2 M を備える。通常は、レジスタ 2 M から A L U 1 2 A ~ 1 2 D A へデータが供給される。このアーキテクチャにより、このレジスタを参照する命令に必要なオペランドが 4 ビットと少なくでき、4 つの S P E 2 A ~ 2 D を独立して制御する場合でも、合計 1 6 ビットで構成できるので非常にコンパクトになる。

【 0 0 3 3 】

続いて各 S P E におけるアキュムレータ (A c c) 選択制御処理について示す。

A c c 1 3 A ~ A c c 1 3 D は、各 A L U からの参照が可能な 4 つのレジスタ (レジスタ R0 ~ R3) である。A c c 1 3 A ~ A c c 1 3 D は、それぞれの S P E に分散して配置されるが、他の S P E から参照して読み出すことができる。

ただし、S P E 2 A ~ 2 C において、A c c 1 3 A ~ A c c 1 3 C と A c c 1 3 D とは、参照方法が異なる。

A c c 1 3 A ~ A c c 1 3 C は、レジスタ参照部 1 5 A ~ 1 5 C によって選択され、次に A c c 1 3 D やレジスタ 2 M などと共にセクタ 1 4 A ~ 1 4 D によって選択されて A L U 1 2 A ~ A L U 1 2 D A の片側に入力される。

A c c 1 3 D は、レジスタ 2 M などと共にセクタ 1 4 A ~ 1 4 D によって選択されて A L U 1 2 A ~ 1 2 D A の片側に入力される点が異なる。

【 0 0 3 4 】

セクタ 1 4 A ~ 1 4 D の選択により、各 A L U から参照されるレジスタ R3 は、常に A c c 1 3 D に対応しているが、レジスタ R0 ~ R2 は、レジスタ参照部 1 5 A ~ 1 5 C が設定される状態により参照先を変更できるようになる。そのため、レジスタ R0 ~ R2 は、レジスタ参照部 1 5 A ~ 1 5 C の設定によって A c c 1 3 A ~ A c c 1 3 C のいずれかに変化する。

レジスタ R4 ~ R15 へのデータの書き込みは、A c c 1 3 A ~ A c c 1 3 D からのデータ転送命令によって行われる。同一レジスタへの同時書き込みは、コンパイラで容易に回避できる。また、レジスタ R4 ~ R15 へのデータの書き込みは上記の命令でしか実行できないので、同一レジスタからの読み出しと書き込みが同時に起こる場合でも、データのバイパスを設けるなどでタイミング問題を容易に回避できる。

なお、演算部 1 2 D 並びに乗算器 1 2 D M への入力は、前述の A L U 1 2 D A の説明を参照する。

【 0 0 3 5 】

図を参照し、画像データを入出力する処理について示す。

図 3 は、画像データの入力処理のタイミングチャートである。

S P E 2 D には、P E 2 の外部とデータを入出力するための構成を有している。

入力画像データは、I O P 4 によって取り込まれ、データ入力シフトレジスタ (S e r i a l - i n) 6 にシフトされながら設定される。P E 2 がリセット (初期化) された直後 (時刻 t_{i0}) はデータ入力シフトレジスタ 6 が空であることを示すフラグ S i n - e m t y が 1 に、また、データ入力シフトレジスタ 6 にデータが準備できたことを示すフラグ S i n - r d y が 0 に

10

20

30

40

50

なっている。計算を開始するとIOP4はデータ入力シフトレジスタ6へデータを書き込む前にSin-EMPTYが1になるのを待つ(時刻 t_{i1})が、既にSin-EMPTYが1なので直ぐにデータ転送を始める(時刻 t_{i2})。

【0036】

一方、PE-I3はSin-rdyが1になるのを待ってウェイト状態になる(時刻 t_{i2})。PE-I3がウェイト状態になると同時に、全てのSPEがウェイト状態に入る。IOP4はデータ入力シフトレジスタ6に1ライン分のデータ転送を終了するとSin-rdyを1にする(時刻 t_{i4})。するとPE-I3はウェイト状態から抜け出して(時刻 t_{i5})、1クロックでデータ入力シフトレジスタ6からデータ入力レジスタ(SIN-reg)16へデータを転送すると同時にSin-rdyをクリアし、更にSin-EMPTYを1にしてIOP4にデータ入力シフトレジスタ6が空になったことを知らせる(時刻 t_{i8})。データ入力バッファ(SIN-reg)16にデータが用意できると、それをSPE2Dの処理により読み出されて、ローカルメモリに蓄積したり、或いは即座に計算処理に使ったりできる。

10

【0037】

一方、IOP4は1ライン分のデータを転送した後で他にすべき処理が無くなるとウェイト状態に入って、次のラインのデータを転送するためにSin-EMPTYが1になるのを待つ(時刻 t_{i7})。Sin-EMPTYが1になるとIOP4は再びデータ入力シフトレジスタ6へのデータ転送を開始する。

【0038】

Sin-EMPTYは、IOP4が適当なタイミングでクリアする(時刻 t_{i3})。この手順により、IOP4からのデータ書き込みとPE2での計算処理とが同期し、更にPE2での計算処理と画像データの入力作業とを並行させて実行できる。

20

【0039】

図4は、IOP4を介して外部へデータを出力する処理のタイミングチャートである。

PE2がリセットされた直後は、データ出力バッファ(SOUT-reg)17が空であることを示すフラグSOUT-EMPTYが1に、またSOUT-regにデータが準備できたことを示すフラグSOUT-rdyが0になっている(時刻 t_{o0})。計算を開始すると、IOP4はSOUT-rdyが1になるのを待ってウェイト状態になる(時刻 t_{o1})。

一方、PE-I3はSOUT-EMPTYが1になるのを待つウェイト状態になろうとするが、既にSOUT-EMPTYが1なのでウェイトには入らない(時刻 t_{o1})。そして、PE-I3がSOUT-EMPTYをクリアすると同時に、SPE2DがSOUT-regに出力データを蓄積し始める(時刻 t_{o3})。

30

出力すべきデータの準備できると、PE-I3はSOUT-rdyを1にする(時刻 t_{o4})。するとIOP4はウェイト状態から抜け出して(時刻 t_{o7})、1クロックでSOUT-regからSerial-outへデータを転送すると同時にSOUT-rdyをクリアし、更にSOUT-EMPTYを1にしてPE-I3にSOUT-regが空になったことを通知する(時刻 t_{o8})。その後、IOP4はSerial-outをシフトしながらデータを読み出して、LSIの外部へと転送する(時刻 t_{o8})。この手順により、PE2でのデータ処理と、IOP4によるLSI外部へのデータ出力とを同期させ、更にこれらの処理を並行して実行できる。

【0040】

40

この他に、隣り合ったPE2間でデータを転送するために、PE出力レジスタLレジスタ18とPE出力レジスタRレジスタ19が在る。左側即ちPE2の番号が小さいPE2へ渡したいデータは、SPE2DがそれをPE出力レジスタLレジスタ18に書くと、次の命令以降で左隣のSPE2Dが読み出せる。同様に、右側即ちPE2の番号が大きいPE2へ渡したいデータは、SPE2DがそれをPE出力レジスタRレジスタ19に書き込むと、次の命令以降で右隣のSPE2Dが読み出せる。

以上述べてきたように、図2に示す構成を用いることにより、ローカルメモリへのアクセスにおけるバンド幅の不足や各SPEを命令実行状態に保ち続けることに効果がある。

【0041】

さらに、画像処理用途に適用する際に、生じるローカルメモリへのアクセスにおけるバ

50

ンド幅の不足の問題を解決する技術について説明する。

図5は、本発明の実施形態における演算処理に入力される映像信号の例を示す。

入力される映像信号は、図に示されるように640画素×480画素を2次元に配列した画素で構成される画面によって示される白黒画像を表している。1画素は、8ビットで構成され、各画素の明るさに応じた階調が示される。

【0042】

続いて、画像処理特有の処理についての問題と対策について説明する。

並列計算装置1には、映像信号がビデオカメラから1ライン分の640画素を単位として、ラインごとに480回に分けてデータが入力される。入力されたデータは、PE2-0からPE2-106の107個のPE2に分割して格納され、各PE2が分担して演算処理をする。

10

ところで、画像処理では、特定の画素を定め、その画素に対し、2次元平面で示す座標軸方向に隣接する上下左右方向の画素のデータを参照することが多い。

頻繁に参照される上下左右の画素のデータを効率よくアクセスできないと、処理が煩雑になり処理速度が低下する。

最初に、水平方向に並んだ左右の画素を効率よく扱うための方法を説明する。

各PE2は、入力されたデータの内、分割された6画素分のデータを担当する(107×6=642)。しかし、自ら担当する6画素しかローカルメモリに格納しないと、頻繁に両隣のPE2との間でデータの受け渡しが発生して効率が悪くなる。そこで各PE2は、両隣が格納すべきデータを1画素分ずつ重複して格納することにする。各ローカルメモリは、1ライン当たり8画素を記憶することにする。

20

【0043】

続いて、垂直方向に並んだ上下の画素を効率よく扱うための方法を説明する。上下方向の画素は、ライン単位で扱われることからライン間の参照として説明する。

各SPEに配置されるローカルメモリ11A~11Dの容量は512×8b(ビット)なので、それぞれのSPEが備える1つのローカルメモリには、一度に64ライン(=512÷8(画素))分のデータを格納できる。PE2内にローカルメモリは、4個あるので、最大で256ライン分のデータを同時に格納できるが、1画面は480ラインなので1画面分のデータまでを格納することはできない。

画像処理では画像を狭い領域ごとに区切って演算対象にすることが多いため、入力されるデータを例えばSPE2Aのローカルメモリ11Aだけに順に格納すると、演算する時にSPE2Aのローカルメモリ11Aだけが頻繁に読み出され、メモリを分散配置して得られたデータ転送能力を生かせないことになる。

30

【0044】

また、一度に65ライン分以上のデータを格納しようとする、初めの64ライン分と次の64ライン分のデータを別々のメモリに格納しなければならず、データを格納するためのプログラムだけでなく、そのデータを使うプログラムも、条件判断処理を行うため複雑になってしまう。

このように、単にローカルメモリを分割して配置するだけでは、メモリアクセスのバンド幅を広げることではできず、また65ライン分以上のデータを格納して処理しようすると、処理プログラムが複雑化するという新たな問題が生じてしまう。

40

【0045】

ここで、ローカルメモリへアクセスするためのアドレスレジスタについて説明する。

図6は、アドレスレジスタの構成を示す図である。この図に示されるアドレスレジスタには、モード0~モード2として選択できる3つのモードを設定できる。

モード0では、ベースポイントBPと、4つのポイントAP0~AP3によって参照するメモリアドレスを定める。メモリアクセス命令は、そのオペランドにポイントAP0~AP3を一つ指定できる。例えばポイントAP0を指定するとポイントAP0の下位3ビットがメモリアドレスのビット2-0になり、ビット4-3は0固定、そしてBPの値がビット8-5になる。

50

【 0 0 4 6 】

メモリアドレスを自動的に1増加させる命令を実行する場合は、ローカルメモリをアクセスした後にポインタA P 0が1増加する。ポインタA P 0の値が、7から8になると、下位3ビットが全て0になるので、結局最初のアドレスに戻ることになる。このモードは、画像処理において、1ラインごとに処理する場合に便利である。ラインを1だけ移動する場合は、ベースポインタB Pの値を1だけ増減する。ベースポインタB PやポインタA P 0 ~ A P 3を1だけ増減する命令が用意されている。このモードでは、例えばポインタA P 0はデータの読み出し番地を、ポインタA P 1は一時的にデータを書き込む格納番地をそれぞれ保持し、読み書きする命令でポインタA P 0とポインタA P 1とをオペランド指定して切り替えてそれぞれの番地を切り換えてアクセスできる。

10

【 0 0 4 7 】

モード1の動作も同様であるが、ポインタA P 0とA P 2の下位4ビットが割り当てられているので、データを2ラインずつアクセスする場合に有用である。

モード2ではポインタA P 0の値がそのまま9ビットのアドレスに変換できるので、512B(バイト)のメモリ空間をリニアにアクセスできる。

【 0 0 4 8 】

図7は、本実施形態によるローカルメモリへの画像データのマッピング例を示す。

65ライン以上のデータを格納する場合にも処理プログラムを同一にするために、図に示すように画像データを1ライン毎にS P E 2 Aのローカルメモリ11A、S P E 2 Bのローカルメモリ11B、S P E 2 Cのローカルメモリ11Cと順に格納することにする。

20

【 0 0 4 9 】

最初に画像信号におけるライン0のデータをローカルメモリーに格納する場合の手順を示す。

【 0 0 5 0 】

I O P 4によって取り込まれた画像信号は、データ入力シフトレジスタ6によって、S P E 2 Dにおけるデータ入力バッファ16に設定される。設定された画像信号のデータをS P E 2 DにおけるA L U 1 2 D Aがデータ入力バッファ16から読み出すと、データはA c c 1 3 Dに書き込まれ、レジスタR3として参照できるようになる。S P E 2 AにおけるA L U 1 2 Aは、レジスタR3に書き込まれたデータを参照し、A c c 1 3 Aへ転送した後、ローカルメモリ11Aに書き込む。

30

【 0 0 5 1 】

続いて、S P E 2 Aのローカルメモリ11Aにライン0のデータを書き込むプログラムを使って、次のライン1のデータをS P E 2 Bのローカルメモリ11Bに書き込む手順を説明する。

図8は、本発明の実施形態におけるP E - I 3を示す概略ブロック図である。この図に示されるP E - I 3は、内部に交換部31を備える。

P E - I 3は、命令メモリ5に登録されたプログラムに記述された命令を分解し、S P E 2 A用、S P E 2 B用、S P E 2 C用の命令(OPコード)と、その命令の交換を行うローテーション命令を抽出する。ローテーション命令はP E - I 3で実行される。S P E 2 A用、S P E 2 B用、S P E 2 C用の命令(OPコード)を、それぞれs p e - a用コード、s p e - b用コード、s p e - c用コードと示し、ローテーション命令をR O T命令と示す。

40

P E I 3における交換部31は、これらのOPコードを、必要に応じてローテーション(交換)する。

【 0 0 5 2 】

交換部31は、命令セレクト部32、命令選択部33A~33C、命令デコード部34A~34Cを備える。

交換部31における命令セレクト部32は、入力されるR O T命令に応じて、命令選択部33A~33Cの入力選択を制御する。命令セレクト部32は、内部に2ビットのカウ

50

ンタを備え、ROT命令が実行されるとカウンタの値を1ずつ増加させる。命令セレクト部32は、カウンタの値が2のときにROT命令が実行されると値を0に戻し、0～2の範囲で変化させる。命令セレクト部32は、そのカウンタの値に応じて、命令選択部33A～33Cの入力を切り換えるセレクト信号を出力する。

命令選択部33A～33Cは、入力されるspe-a用コード、spe-b用コード、spe-c用コードを命令セレクト部32が出力するセレクト信号に応じて切り換える。

命令デコーダ部34A～34Cは、入力される命令コードに応じて各SPE2A～2Cの制御信号を生成し出力する。

【0053】

交換部31において命令選択部33Aは、spe-a用コードがI0入力に、spe-b用コードがI2入力に、spe-c用コードがI1入力に入力される。命令選択部33Bは、spe-a用コードがI1入力に、spe-b用コードがI0入力に、spe-c用コードがI2入力に入力される。命令選択部33Cは、spe-a用コードがI2入力に、spe-b用コードがI1入力に、spe-c用コードがI0入力に入力される。

命令選択部33A～33Cは、入力されるセレクト信号の値0～2に応じて、対応するI0入力、I1入力、I2入力の各入力端子に入力されるコードを出力する。

【0054】

図を参照し、図8に示した交換部31によって各SPEに供給される制御信号を交換する処理を説明する。

図9は、交換部31における命令選択状態を示す図である。

命令セレクト部32が出力するセレクト信号が0のとき、SPE2Aの制御信号にはspe-a用コードが、SPE2Bの制御信号にはspe-b用コードが、SPE2Cの制御信号にはspe-c用コードが出力される。ROT命令が実行され、セレクト信号が1になると、SPE2Aの制御信号にはspe-c用コードが、SPE2Bの制御信号にはspe-a用コードが、SPE2Cの制御信号にはspe-b用コードが出力される。さらにROT命令が実行され、セレクト信号が2になると、SPE2Aの制御信号にはspe-b用コードが、SPE2Bの制御信号にはspe-c用コードが、SPE2Cの制御信号にはspe-a用コードが出力される。次にROT命令が実行されるとセレクト信号の値は0に戻る。

【0055】

命令セレクト部32におけるセレクト信号が0の状態、SPE2Aがspe-a用コードにしたがってライン0の読み込みを行った後に、PE-I3でROT命令を実行すると、セレクト信号が1になる。そのためspe-a用として記述されたプログラムのOPコードであっても、SPE2Bで実行されるようになる。つまり、ライン0を読み込んだプログラムと全く同じプログラムを実行してライン1のデータをSPE2Bのローカルメモリ11Bに書き込むことができる。同様に、ライン1の読み込み終了後にROT命令を実行すると、ライン2のデータがSPE2Cのローカルメモリ11Cに書き込むことができる。

次に、ROT命令を実行してデータを読み込むと、SPE2Aでライン0のデータがライン3のデータで上書きされることになる。それが不都合な場合は、PE-I3でROT命令を行うのと同時にSPE2AでベースポイントBP(図6)を1増やす命令を実行しておく。このようにすることで、ROT命令を実行しながら単純にループするプログラムで、図7のようにデータを格納することができる。

【0056】

なお、SPE2DとPE-I3の命令は、常にspe-d用コードとpe-i用コードであり変化しない。また、SPE2A～SPE2Cをローテート(交換)するためには、これらが全く同じ構成でなければならない。したがって、PE2の外部とのデータ入出力に必要な構成や、乗算器などは全てSPE2Dに集中して配置する。

【0057】

次に、図7のように格納された画像データを使って、画像処理が効率的に行えることを

10

20

30

40

50

説明する。例えば3画素×3画素の窓を有するデジタルフィルタを画像に適用する場合を示す。

最初の処理では、ライン0（上段）、ライン1（中段）、ライン2（下段）からそれぞれ3画素ずつのデータ（合計9画素）を読み出して処理する。ライン0のデータはSPE2A、ライン1のデータはSPE2B、ライン2のデータはSPE2Cの各ローカルメモリ11A～11Cにそれぞれ格納されていると仮定する。

SPE2A～2Cでは、各ラインのデータから3画素ずつ分散して読み出せる。つまり、拡大されたメモリバンド幅を有効に使うことができる。1ライン分の処理が終わって次のラインの処理に移ると、ライン1が上段、ライン2が中段、ライン3が下段になる。1ライン分の処理の最後で、ROT命令を実行すれば、spe-a用コードがSPE2Bで、spe-b用コードがSPE2Cで、spe-c用コードがSPE2Aでそれぞれ実行されるようになるので、順次切り換えられるラインへの移動が簡単に行える。

【0058】

つまり、上段のデータにアクセスする命令はspe-a用のコードとして記述されているが、データが格納されているローカルメモリーは、SPE2Bにおけるローカルメモリー11Bである。spe-a用のコードが、ローカルメモリー11Bが含まれるSPE2Bで実行されるので、結局ライン1のデータが上段のデータとして処理することになるからである。ただし、spe-c用のコードについては注意が必要で、これはSPE2Aで実行されるが、そのままではライン0をアクセスしてしまう。そこで、ROT命令と同時にSPE2AにおいてベースポインタBPを1増加する命令を実行しておく。このように、ROT命令とベースポインタBPを増加させる命令を加えて、単純にループするプログラムで、図7のように格納されたデータを使った画像処理ができる。

【0059】

最後に、図を参照し、図2の3つのレジスタ参照部15A～15Cの機能について説明する。

図10は、レジスタ参照部の動作を示す図である。

SPEをローテーションしてもプログラムが正しく動作するためには、spe-a用のコードが実行されているSPEのAccが、他のSPEからは、レジスタR0として読み出せることが必要になる。同様に、spe-b用のコードが実行されているSPEのAccがレジスタR1、spe-c用のコードが実行されているSPEのAccがレジスタR2として読み出せることが必要になる。

そこで、レジスタ参照部15A～15Cは、図に示すように、命令セレクト部32が出力するセレクト信号の値に応じてAcc13A～13Cを選択してレジスタR0～R2とする。

命令セレクト部32が出力するセレクト信号が0のとき、レジスタR0にはAcc13Aが、レジスタR1にはAcc13Bが、レジスタR2にはAcc13Cが参照される。セレクト信号が1のとき、レジスタR0にはAcc13Bが、レジスタR1にはAcc13Cが、レジスタR2にはAcc13Aが参照される。セレクト信号が2のとき、レジスタR0にはAcc13Cが、レジスタR1にはAcc13Aが、レジスタR2にはAcc13Bが参照される。

【0060】

（第2実施形態）

図を参照し、並列計算装置の一実施形態について示す。

図11は、第2実施形態におけるPE2の構成を示すブロック図である。図2と同じ構成には同じ符号を付す。

この図に示される並列計算装置1aは、PE2a、PE-I3a、IOP4、命令メモリ5、データ入力シフトレジスタ6、データ出力シフトレジスタ7が示される。図1と同じ構成には同じ符号を付す。また、PE2aは、同じ構成を有する複数のPE2a-0～PE2a-106を代表する。また、IOP4、命令メモリ5は、記載が省略されている。

10

20

30

40

50

【 0 0 6 1 】

PE 2 a は、図 2 における PE 2 に相当し、一部異なる構成を有している。PE 2 a は、SPE 2 A a ~ SPE 2 D a とレジスタ 2 M を有する。

PE 2 a における SPE 2 A a ~ SPE 2 D a は、図 2 における SPE 2 A ~ SPE 2 D に相当するが、セクタ 1 4 A ~ セクタ 1 4 D 及びレジスタ参照部 1 5 A ~ 1 5 C に代え、セクタ 1 4 X A ~ セクタ 1 4 X D を備える。

セクタ 1 4 X A ~ セクタ 1 4 X C は、セクタ 1 4 A ~ セクタ 1 4 C とレジスタ参照部 1 5 A ~ 1 5 C を一体化した構成にあたり入力される信号を選択制御信号に応じて選択する。セクタ 1 4 X A に入力される信号は、Acc 1 3 A ~ Acc 1 3 D、ローカルメモリ 1 1 A 及びレジスタ 2 M であり、PE - I 3 a からの選択制御信号に応じて入力される信号を切り換える。セクタ 1 4 X B ~ 1 4 X C についても、セクタ 1 4 X A と同様である。

また、セクタ 1 4 X D は、入力される信号が、Acc 1 3 A ~ Acc 1 3 D、ローカルメモリ 1 1 D、レジスタ 2 M 及びデータ入力バッファ (SIN_reg) 1 6 からの情報並びに隣接する PE 2 a からの入力信号であり、PE - I 3 a からの選択制御信号に応じて入力される信号を切り換える。

【 0 0 6 2 】

PE - I 3 a は、図 2 における PE - I 3 に相当し、交換部 3 1 a を備える。交換部 3 1 a は、命令セレクト部 3 2 と選択制御部 3 5 A ~ 3 5 D を備える。

命令選択デコード部 3 5 A ~ 3 5 D は、図 8 における命令選択部 3 3 A ~ 3 3 D と命令デコード部 3 4 A ~ 3 4 D をそれぞれ合わせた構成を備え、入力される spe - a 用コード、spe - b 用コード、spe - c 用コードを、命令セレクト部 3 2 が出力するセレクト信号に応じて切り換えて、入力される命令コードに応じて各 SPE 2 A ~ 2 C を制御する制御信号を生成し出力する。

【 0 0 6 3 】

上記の構成とすることにより SPE 2 A ~ 2 C 内におけるレジスタ R0 ~ R3 の選択を簡素化し、SPE 内のセクタ 1 4 X A ~ 1 4 X C の 1 段で切り換えるようになる。

第 1 実施形態に示した ROT 命令によって、SPE 2 A 用、SPE 2 B 用、SPE 2 C 用の命令 (OP コード) を交換し、各 SPE の ALU 1 2 A ~ 1 2 C に供給する選択処理を同じように実施できる。

【 0 0 6 4 】

以上に示した実施形態により、複数のサブ PE を持つ VLIW 型アーキテクチャを採用した並列計算機において、ローカルメモリをサブ PE 毎に分割して配置することで、ローカルメモリへの多重アクセス問題を回避しながら、メモリとサブ PE 間のデータ転送能力を高めることができる。さらに、2 つ以上のサブ PE の構成を全く同じにし、サブ PE の動作を制御する制御信号を交換することによって、ローカルメモリに格納されている全てのデータを短時間 (1 クロック) で交換したのと同じ効果を出せる。これらの特徴により、サブ PE とローカルメモリ間のデータ転送能力が大幅に向上し、実効演算性能が高い並列処理プロセッサを提供できる。

【 0 0 6 5 】

本発明の実施形態によれば、並列計算装置 1 は、並列して演算処理を行う複数の演算プロセッサ (PE) 2 を備える。演算プロセッサ 2 では、命令実行制御部 3 がそれぞれに制御命令を供給する。サブプロセッサ SPE 2 A は、記憶部 1 1 A が複数のデータ又は複数の演算結果を保持する。ALU 1 2 A は、記憶部 1 1 A から読み出したデータに演算処理を施し、その結果を記憶部に供給する。

また、サブプロセッサは、複数のサブプロセッサのうち少なくとも 2 つのサブプロセッサが同じ構造を有する。それらの同じ構造を有するサブプロセッサは、同一構造プロセッサ群を形成する。また、命令実行制御部は、交換部が、同一構造サブプロセッサ群に含まれるサブプロセッサに供給する制御信号を交換して供給する。

これにより、記憶部に記憶されたデータを交換したり、参照するデータを切り換えて同

10

20

30

40

50

様の処理をしたりするためにオペランドの異なる類似のプログラムを用意することなく、1命令の処理により、データと処理の組み合わせを交換することができる。

【0066】

なお、本発明は、上記の各実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲で変更可能である。

例えば、本発明の説明では1個の並列計算装置1(LSI)に内蔵される演算プロセッサ-2の個数を107としたが、本発明はこれに制限されるものではなく、1以上の演算プロセッサ-2を内蔵する計算機に適用できる。また、VLIW型で並列化するサブPEの個数を各演算プロセッサ-2ごとに4としたが、本発明はこれに制限されるものではなく、複数のSPEを持つシステムに適用できる。さらに、ROT命令により、OPコードを交換するサブPEの個数を3としたが、本発明はこれに制限されるものではなく、複数のサブPEのOPコードが交換されるシステムに適用できる。

10

【0067】

また、本発明の説明では全ての演算プロセッサ-2でSIMD型を構成し、1個の命令制御プロセッサによって制御されるとしたが、本発明はこれに制限されるものではなく、演算プロセッサ-2を複数のグループに分割し、それぞれに制御プロセッサを配置するような、SIMD型とMIMD型の中間のアーキテクチャにも適用できる。また、図11に示す第2実施形態の構成では、各SPEのOPコードを交換した後でOPコードをデコードしているが、制御信号の本数がOPコードのビット数よりも少ないような場合には、OPコードをデコードした後で交換しても、本発明と同様の効果を得ることができる。

20

【0068】

また、図2に示した構成では説明を分かり易くするために、Acc13A~Acc13Cの選択を2段にしているが、必要な回路素子の数を減らし、同時に回路の動作速度を上げるためには、図11に示した構成のように1段で構成する方が望ましい。その場合は図11に示すように、図8と同じ命令セレクト信号を用いて、セクタ14XA~14XDのデータ選択信号そのものを交換する。

【符号の説明】

【0069】

2 演算プロセッサ(P E)

2 A、2 B、2 C、2 D サブ演算プロセッサ(S P E、サブプロセッサ)

30

2 G サブ演算プロセッサグループ(同一構造プロセッサ群)

3 命令実行制御プロセッサ(P E - I、命令実行制御部)

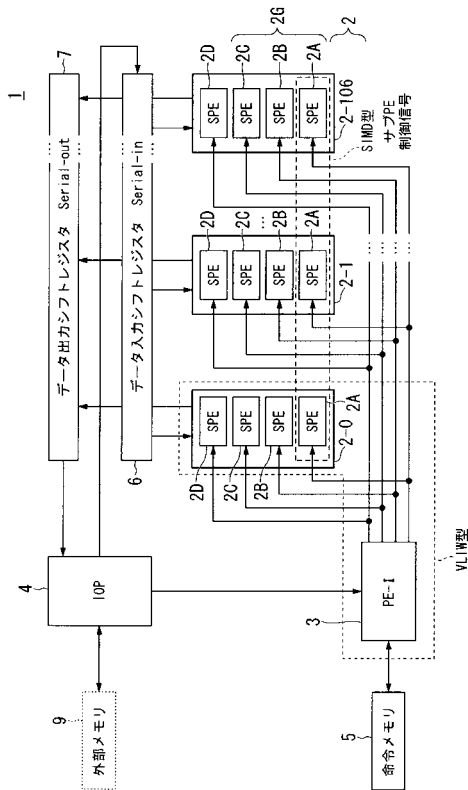
1 1 A、1 1 B、1 1 C、1 1 D ローカルメモリー(L M、記憶部)

1 2 A、1 2 B、1 2 C、1 2 D A A L U (演算部)

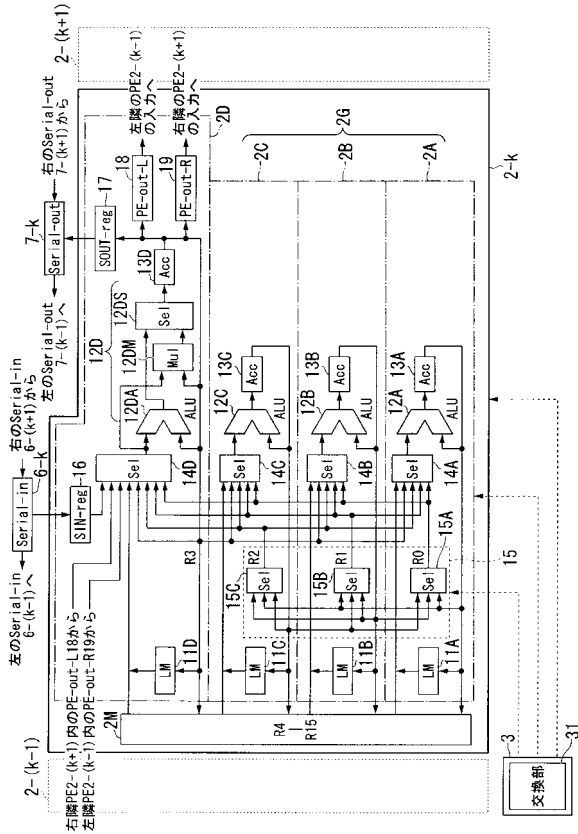
1 3 A、1 3 B、1 3 C、1 3 D A c c (レジスター部)

3 1 交換部

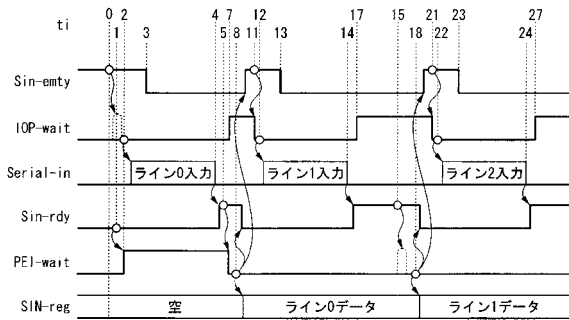
【図1】



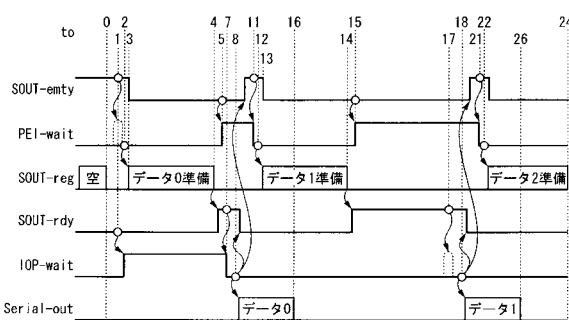
【図2】



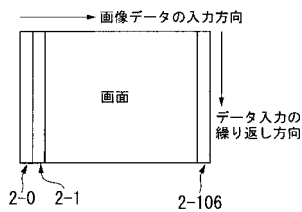
【図3】



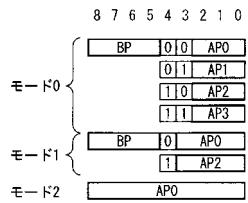
【図4】



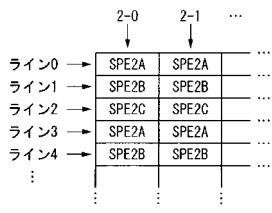
【図5】



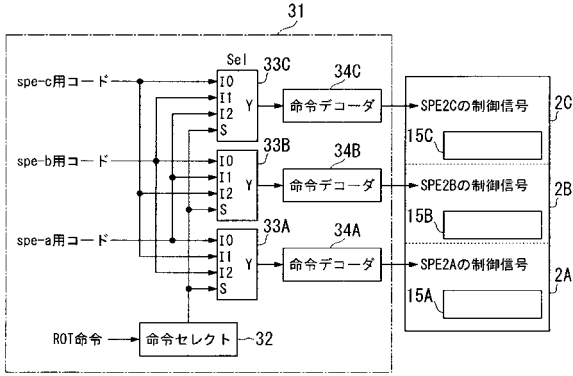
【図6】



【図7】



【図 8】



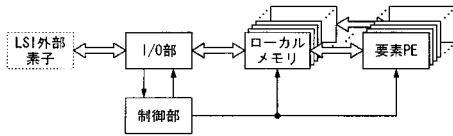
【図 9】

命令セレクト	0	1	2
SPE2Cの制御信号	spe-c用コード	spe-b用コード	spe-a用コード
SPE2Bの制御信号	spe-b用コード	spe-a用コード	spe-c用コード
SPE2Aの制御信号	spe-a用コード	spe-c用コード	spe-b用コード

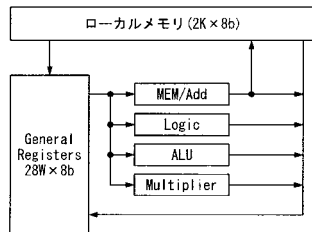
【図 10】

命令セレクト	0	1	2
R2	Acc13C	Acc13A	Acc13B
R1	Acc13B	Acc13C	Acc13A
R0	Acc13A	Acc13B	Acc13C

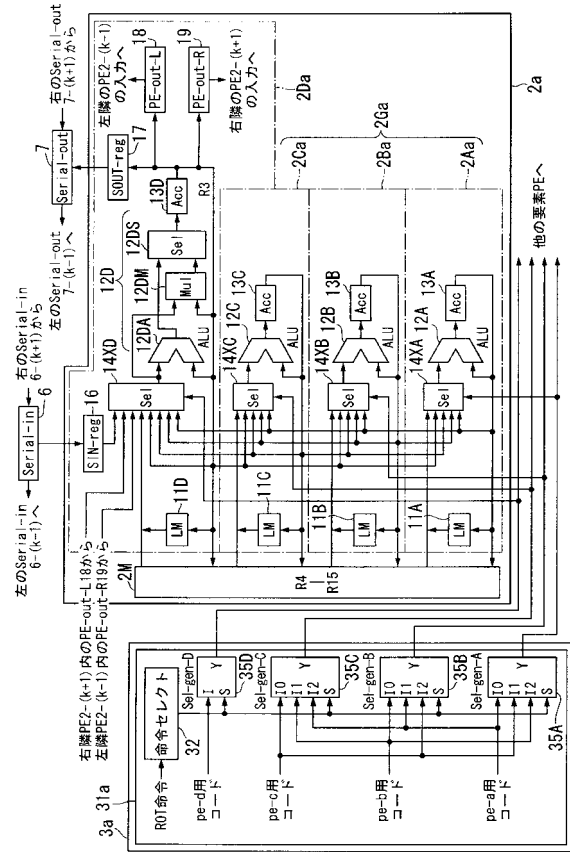
【図 12】



【図 13】



【図 11】



フロントページの続き

(51)Int.Cl. F I
G 0 6 T 1/20 B

(72)発明者 豊田 新次郎
埼玉県和光市本町 8 番 1 号 株式会社ホンダ・リサーチ・インスティテュート・ジャパン内

(72)発明者 宮川 宣明
埼玉県和光市本町 8 番 1 号 株式会社ホンダ・リサーチ・インスティテュート・ジャパン内

審査官 清木 泰

(56)参考文献 特開 2 0 1 0 - 2 7 1 7 9 9 (J P , A)
特開 2 0 1 0 - 2 5 7 2 0 0 (J P , A)
特開 2 0 0 0 - 3 0 5 7 8 1 (J P , A)
特開 2 0 0 2 - 2 5 1 2 8 4 (J P , A)
特開 2 0 0 4 - 0 5 4 5 8 5 (J P , A)
国際公開第 2 0 0 6 / 0 4 9 3 3 1 (W O , A 1)
細田寛人, 車載用画像認識プロセッサIMAPCAR, N E C 技報, 日本, 日本電気株式会社, 2 0 0 6 年 1 1 月 2 0 日, 第59巻, 第5号, Pages:22-25
西脇大輔, 外 4 名, 安心・安全を実現する自動車向け画像認識技術, 映像情報インダストリアル, 日本, 産業開発機構(株), 2 0 0 7 年 2 月 1 日, 第39巻, 第2号, Pages:61-69
岡崎信一郎, 高並列プロセッサIMAPCARと車載画像認識への適用, 電子情報通信学会技術研究報告, 日本, 社団法人電子情報通信学会, 2 0 0 7 年 5 月 1 7 日, 第107巻, 第57号, (PRMU2007-1~26), Pages:53-58
京昭倫, 高並列アレイ型プロセッサIMAPCARのアーキテクチャとその技術展望, 電子情報通信学会技術研究報告, 日本, 社団法人電子情報通信学会, 2 0 0 7 年 5 月 2 4 日, 第107巻, 第76号, (ICD2007-17~36), Pages:97-102
Shorin Kyo et al., An Integrated Memory Array Processor Architecture for Embedded Image Recognition Systems, Proceedings of 32nd International Symposium on Computer Architecture (ISCA'05), IEEE, 2 0 0 5 年 6 月 4 日, Pages:134-145

(58)調査した分野(Int.Cl., D B 名)

G 0 6 F 1 5 / 8 0

G 0 6 F 1 5 / 1 6 - 1 5 / 1 7 7

G 0 6 F 9 / 3 0 - 9 / 4 2

G 0 6 T 1 / 2 0