

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6829838号
(P6829838)

(45) 発行日 令和3年2月17日(2021.2.17)

(24) 登録日 令和3年1月27日(2021.1.27)

(51) Int. Cl.	F I
G06F 17/16 (2006.01)	G06F 17/16 M
G06F 15/173 (2006.01)	G06F 15/173 683B
G06F 9/38 (2006.01)	G06F 9/38 370C
G06F 7/483 (2006.01)	G06F 9/38 370A
	G06F 7/483

請求項の数 14 (全 49 頁)

(21) 出願番号	特願2017-95803 (P2017-95803)	(73) 特許権者	515130201
(22) 出願日	平成29年5月12日(2017.5.12)		株式会社 Preferred Networks
(65) 公開番号	特開2018-194905 (P2018-194905A)		東京都千代田区大手町1丁目6番1号 大手町ビル
(43) 公開日	平成30年12月6日(2018.12.6)	(73) 特許権者	503359821
審査請求日	令和2年4月8日(2020.4.8)		国立研究開発法人理化学研究所 埼玉県和光市広沢2番1号
		(74) 代理人	100088155 弁理士 長谷川 芳樹
		(74) 代理人	100113435 弁理士 黒木 義樹
		(74) 代理人	100121980 弁理士 沖山 隆

最終頁に続く

(54) 【発明の名称】 演算装置及び演算システム

(57) 【特許請求の範囲】

【請求項1】

それぞれが第1ビット数を有する第1～第3入力データを入力し、演算モードを示すモード信号に応じて、それぞれが前記第1ビット数の半分の第2ビット数を有する第1～第8出力データを出力するマルチプレクサと、

前記第1出力データと前記第2出力データとの乗算を行う第1乗算器と、
前記第3出力データと前記第4出力データとの乗算を行う第2乗算器と、
前記第5出力データと前記第6出力データとの乗算を行う第3乗算器と、
前記第7出力データと前記第8出力データとの乗算を行う第4乗算器と、
を備え、

前記マルチプレクサは、
前記モード信号が前記第1ビット数を有するデータを用いた第1演算モードを示す場合には、前記第1入力データのうちの前記第2ビット数の上位データである第1上位データを前記第1出力データ及び前記第5出力データとし、前記第1入力データのうちの前記第2ビット数の下位データである第1下位データを前記第3出力データ及び前記第7出力データとして出力するとともに、前記第2入力データのうちの前記第2ビット数の上位データである第2上位データを前記第2出力データ及び前記第4出力データとし、前記第2入力データのうちの前記第2ビット数の下位データである第2下位データを前記第6出力データ及び前記第8出力データとして出力し、

前記モード信号が前記第2ビット数を有するデータを用いた第2演算モードを示す場合に

は、前記第 1 上位データを前記第 1 出力データ及び前記第 5 出力データとし、前記第 1 下位データを前記第 3 出力データ及び前記第 7 出力データとして出力し、前記第 2 上位データを前記第 2 出力データとし、前記第 2 下位データを前記第 4 出力データとして出力するとともに、前記第 3 入力データのうちの前記第 2 ビット数の上位データである第 3 上位データを前記第 6 出力データとし、前記第 3 入力データのうちの前記第 2 ビット数の下位データである第 3 下位データを前記第 8 出力データとして出力する、演算装置。

【請求項 2】

前記第 1 乗算器の乗算結果である第 1 乗算結果と前記第 2 乗算器の乗算結果である第 2 乗算結果との加算を行う第 1 加算器と、

前記第 3 乗算器の乗算結果である第 3 乗算結果と前記第 4 乗算器の乗算結果である第 4 乗算結果との加算を行う第 2 加算器と、

前記第 1 乗算結果、前記第 2 乗算結果、前記第 3 乗算結果、及び前記第 4 乗算結果の加算を行う部分加算器と、

をさらに備える、請求項 1 に記載の演算装置。

【請求項 3】

前記第 1 乗算器は、前記第 1 乗算器の乗算の途中結果である第 1 中間結果及び第 2 中間結果を前記第 1 乗算結果として出力し、

前記第 2 乗算器は、前記第 2 乗算器の乗算の途中結果である第 3 中間結果及び第 4 中間結果を前記第 2 乗算結果として出力し、

前記第 3 乗算器は、前記第 3 乗算器の乗算の途中結果である第 5 中間結果及び第 6 中間結果を前記第 3 乗算結果として出力し、

前記第 4 乗算器は、前記第 4 乗算器の乗算の途中結果である第 7 中間結果及び第 8 中間結果を前記第 4 乗算結果として出力する、請求項 2 に記載の演算装置。

【請求項 4】

それぞれが前記マルチプレクサ、前記第 1 乗算器、前記第 2 乗算器、前記第 3 乗算器、前記第 4 乗算器、及び前記部分加算器を有する複数の演算部と、

前記複数の演算部の前記部分加算器の加算結果の加算を行う第 3 加算器と、

前記第 1 加算器の加算結果である第 1 加算結果及び前記第 2 加算器の加算結果である第 2 加算結果と、前記第 3 加算器の加算結果である第 3 加算結果と、のいずれかを前記モード信号に応じて出力するセレクタと、

をさらに備え、

前記第 1 加算器は、前記複数の演算部の前記第 1 乗算結果及び前記第 2 乗算結果の加算を行い、

前記第 2 加算器は、前記複数の演算部の前記第 3 乗算結果及び前記第 4 乗算結果の加算を行い、

前記セレクタは、前記モード信号が前記第 1 演算モードを示す場合には、前記第 3 加算結果を出力し、前記モード信号が前記第 2 演算モードを示す場合には、前記第 1 加算結果及び前記第 2 加算結果を出力する、請求項 2 又は請求項 3 に記載の演算装置。

【請求項 5】

前記第 1 乗算結果の第 1 シフト量、前記第 2 乗算結果の第 2 シフト量、前記第 3 乗算結果の第 3 シフト量、及び前記第 4 乗算結果の第 4 シフト量を演算するシフト量演算回路をさらに備え、

前記複数の演算部のそれぞれは、前記第 1 シフト量に基づいて前記第 1 乗算結果をシフト処理する第 1 整列部と、前記第 2 シフト量に基づいて前記第 2 乗算結果をシフト処理する第 2 整列部と、前記第 3 シフト量に基づいて前記第 3 乗算結果をシフト処理する第 3 整列部と、前記第 4 シフト量に基づいて前記第 4 乗算結果をシフト処理する第 4 整列部と、をさらに備える、請求項 4 に記載の演算装置。

【請求項 6】

前記シフト量演算回路は、加算対象となる複数の対象データの指数部のうちの最大の指数部である最大指数を演算する最大値演算回路と、前記複数の対象データと前記最大指数

との差分をシフト量として演算する減算回路と、を備える、請求項 5 に記載の演算装置。

【請求項 7】

前記最大値演算回路は、前記複数の対象データの最上位ビットから最下位ビットに向かって順番に比較することによって、前記最大指数を演算する、請求項 6 に記載の演算装置。

【請求項 8】

前記第 1 演算モードは、倍精度演算モードであり、

前記第 2 演算モードは、単精度演算モードである、請求項 1 ~ 請求項 7 のいずれか一項に記載の演算装置。

【請求項 9】

前記第 1 演算モードは、単精度演算モードであり、

前記第 2 演算モードは、半精度演算モードである、請求項 1 ~ 請求項 7 のいずれか一項に記載の演算装置。

【請求項 10】

請求項 1 ~ 請求項 9 のいずれか一項に記載の演算装置であって、複数の前記演算装置を有する演算ユニットと、

前記演算ユニットを共有する複数のプロセッサと、
を備える演算システム。

【請求項 11】

前記複数のプロセッサは、行列演算を行う場合には、単一のプロセッサとして動作し、
行列演算以外の演算を行う場合には、個別のプロセッサとして動作する、請求項 10 に記載の演算システム。

【請求項 12】

前記複数のプロセッサは、1つのインストラクションで動作する SIMD 動作を行う、
請求項 10 又は請求項 11 に記載の演算システム。

【請求項 13】

前記複数のプロセッサのそれぞれを一意に識別可能なプロセッサ ID を含むメモリアドレスによってアクセス可能なメモリ空間を備え、

前記複数のプロセッサのそれぞれは、当該プロセッサのプロセッサ ID を含むメモリアドレスによって示されるメモリ領域にアクセス可能である、請求項 10 ~ 請求項 12 のいずれか一項に記載の演算システム。

【請求項 14】

前記複数のプロセッサはリング結合されており、

前記複数のプロセッサのそれぞれは、前記演算ユニットから受信したデータを前記リング結合を介して他のプロセッサに順に転送する、請求項 13 に記載の演算システム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、演算装置及び演算システムに関する。

【背景技術】

【0002】

従来、複数のデータに対し同一の演算を同時に行うことが可能な SIMD (Single Instruction Multiple Data) 演算器が知られている。例えば、SIMD 演算器を構成する各演算器に浮動小数点演算器を用いることにより、プロセッサの 1 コアにおいて並列浮動小数点演算が実現される。このような演算器では、供給可能なデータのビット数が一定であることから、例えば、2つの倍精度データに代えて、4つの単精度データを供給することにより、倍精度の 1 演算と単精度の 2 演算とを切り替えることが可能となる(例えば、特許文献 1 参照)。

【先行技術文献】

【特許文献】

10

20

30

40

50

【 0 0 0 3 】

【特許文献1】特表2002-528786号公報

【発明の概要】

【発明が解決しようとする課題】

【 0 0 0 4 】

倍精度データの1つの演算に用いられる演算器の回路規模は、単精度データの1つの演算に用いられる演算器の回路規模の4倍程度である。しかしながら、2つの倍精度データを用いた1つの演算に用いられる演算器を、4つの単精度データを用いた2つの演算に用いる場合、演算に用いられる回路素子の割合が低下する。このように、所定のビット数を有する2つのデータを用いた1つの演算を、半分のビット数を有する4つのデータを用いた2つの演算に切り替えると、全体の半分程度の回路素子が使用されないことになる。本技術分野では、演算装置の回路規模を拡大することなく、演算装置の演算性能のさらなる向上が望まれている。

【 0 0 0 5 】

本発明は、回路規模を拡大することなく、演算性能を向上可能な演算装置及び演算システムを提供する。

【課題を解決するための手段】

【 0 0 0 6 】

本発明の一側面に係る演算装置は、それぞれが第1ビット数を有する第1～第3入力データを入力し、演算モードを示すモード信号に応じて、それぞれが第1ビット数の半分の第2ビット数を有する第1～第8出力データを出力するマルチプレクサと、第1出力データと第2出力データとの乗算を行う第1乗算器と、第3出力データと第4出力データとの乗算を行う第2乗算器と、第5出力データと第6出力データとの乗算を行う第3乗算器と、第7出力データと第8出力データとの乗算を行う第4乗算器と、を備える。マルチプレクサは、モード信号が第1ビット数を有するデータを用いた第1演算モードを示す場合には、第1入力データのうちの第2ビット数の上位データである第1上位データを第1出力データ及び第5出力データとし、第1入力データのうちの第2ビット数の下位データである第1下位データを第3出力データ及び第7出力データとして出力するとともに、第2入力データのうちの第2ビット数の上位データである第2上位データを第2出力データ及び第4出力データとし、第2入力データのうちの第2ビット数の下位データである第2下位データを第6出力データ及び第8出力データとして出力する。マルチプレクサは、モード信号が第2ビット数を有するデータを用いた第2演算モードを示す場合には、第1上位データを第1出力データ及び第5出力データとし、第1下位データを第3出力データ及び第7出力データとして出力し、第2上位データを第2出力データとし、第2下位データを第4出力データとして出力するとともに、第3入力データのうちの第2ビット数の上位データである第3上位データを第6出力データとし、第3入力データのうちの第2ビット数の下位データである第3下位データを第8出力データとして出力する。

【 0 0 0 7 】

この演算装置では、モード信号が第2演算モードを示す場合には、第1上位データ及び第2上位データが第1乗算器に供給され、第1乗算器はこれらのデータの乗算を行う。また、第1下位データ及び第2下位データが第2乗算器に供給され、第2乗算器はこれらのデータの乗算を行う。また、第1上位データ及び第3上位データが第3乗算器に供給され、第3乗算器はこれらのデータの乗算を行う。また、第1下位データ及び第3下位データが第4乗算器に供給され、第4乗算器はこれらのデータの乗算を行う。このように、第1演算モードだけでなく、第2演算モードにおいても、第1～第4乗算器の全ての乗算器が演算を行う。その結果、演算装置の回路規模を拡大することなく、演算装置の演算性能を向上させることが可能となる。

【 0 0 0 8 】

演算装置は、第1乗算器の乗算結果である第1乗算結果と第2乗算器の乗算結果である第2乗算結果との加算を行う第1加算器と、第3乗算器の乗算結果である第3乗算結果と

第4乗算器の乗算結果である第4乗算結果との加算を行う第2加算器と、第1乗算結果、第2乗算結果、第3乗算結果、及び第4乗算結果の加算を行う部分加算器と、をさらに備えてもよい。この場合、第1加算器によって、第1上位データ及び第1下位データと、第2上位データ及び第2下位データと、のベクトル積が得られ、第2加算器によって、第1上位データ及び第1下位データと、第3上位データ及び第3下位データと、のベクトル積が得られる。また、部分加算器によって、第1入力データと第2入力データとの乗算結果が得られる。

【0009】

第1乗算器は、第1乗算器の乗算の途中結果である第1中間結果及び第2中間結果を第1乗算結果として出力してもよい。第2乗算器は、第2乗算器の乗算の途中結果である第3中間結果及び第4中間結果を第2乗算結果として出力してもよい。第3乗算器は、第3乗算器の乗算の途中結果である第5中間結果及び第6中間結果を第3乗算結果として出力してもよい。第4乗算器は、第4乗算器の乗算の途中結果である第7中間結果及び第8中間結果を第4乗算結果として出力してもよい。この場合、第1乗算器の乗算結果として第1中間結果及び第2中間結果が用いられることにより、第1乗算器の演算処理において、例えば桁上がり等による遅延を抑制することができる。これにより、第1乗算器の演算速度を向上させることが可能となる。同様に、第2乗算器、第3乗算器及び第4乗算器の演算速度を向上させることが可能となる。

【0010】

演算装置は、それぞれがマルチプレクサ、第1乗算器、第2乗算器、第3乗算器、第4乗算器、及び部分加算器を有する複数の演算部と、複数の演算部の部分加算器の加算結果の加算を行う第3加算器と、第1加算器の加算結果である第1加算結果及び第2加算器の加算結果である第2加算結果と、第3加算器の加算結果である第3加算結果と、のいずれかをモード信号に応じて出力するセレクタと、をさらに備えてもよい。第1加算器は、複数の演算部の第1乗算結果及び第2乗算結果の加算を行ってもよく、第2加算器は、複数の演算部の第3乗算結果及び第4乗算結果の加算を行ってもよい。セレクタは、モード信号が第1演算モードを示す場合には、第3加算結果を出力し、モード信号が第2演算モードを示す場合には、第1加算結果及び第2加算結果を出力してもよい。この場合、第1加算器及び第2加算器によって、第2ビット数を有するデータでの2つの行列演算結果が得られ、第3加算器によって、第1ビット数を有するデータでの1つの行列演算結果が得られる。そして、セレクタによって、モード信号が第1演算モードを示す場合には、第1ビット数を有するデータでの1つの行列演算結果が出力され、モード信号が第2演算モードを示す場合には、第2ビット数を有するデータでの2つの行列演算結果が出力される。このように、演算モードに応じた行列演算を行うことが可能となる。

【0011】

上記演算装置は、第1乗算結果の第1シフト量、第2乗算結果の第2シフト量、第3乗算結果の第3シフト量、及び第4乗算結果の第4シフト量を演算するシフト量演算回路をさらに備えてもよい。複数の演算部のそれぞれは、第1シフト量に基づいて第1乗算結果をシフト処理する第1整列部と、第2シフト量に基づいて第2乗算結果をシフト処理する第2整列部と、第3シフト量に基づいて第3乗算結果をシフト処理する第3整列部と、第4シフト量に基づいて第4乗算結果をシフト処理する第4整列部と、をさらに備えてもよい。第1乗算結果及び第2乗算結果が浮動小数点データである場合、第1乗算結果及び第2乗算結果を加算するためにはそれぞれの指数部を揃える必要がある。このため、第1シフト量で第1乗算結果をシフト処理し、第2シフト量で第2乗算結果をシフト処理することで、第1加算器での加算が可能となる。同様に、第3シフト量で第3乗算結果をシフト処理し、第4シフト量で第4乗算結果をシフト処理することで、第2加算器での加算が可能となる。

【0012】

シフト量演算回路は、加算対象となる複数の対象データの指数部のうちの最大の指数部である最大指数を演算する最大値演算回路と、複数の対象データと最大指数との差分をシ

フト量として演算する減算回路と、を備えてもよい。対象データが浮動小数点データである場合、複数の対象データを加算するためにはそれぞれの指数部を揃える必要がある。このため、複数の対象データの最大指数と各指数部との差分をシフト量とすることで、複数の対象データの指数部を揃えることが可能となる。

【 0 0 1 3 】

最大値演算回路は、複数の対象データの最上位ビットから最下位ビットに向かって順番に比較することによって、最大指数を演算してもよい。この場合、複数の対象データが最上位ビットから順に比較されるので、比較するビット数を減らすことができ、最大指数の演算を高速化することが可能となる。

【 0 0 1 4 】

第1演算モードは倍精度演算モードであってもよく、第2演算モードは単精度演算モードであってもよい。この場合、倍精度演算モード及び単精度演算モードでの演算が可能となる。このような演算装置においても、演算装置の回路規模を拡大することなく、演算装置の演算性能を向上させることが可能となる。

【 0 0 1 5 】

第1演算モードは単精度演算モードであってもよく、第2演算モードは半精度演算モードであってもよい。この場合、単精度演算モード及び半精度演算モードでの演算が可能となる。このような演算装置においても、演算装置の回路規模を拡大することなく、演算装置の演算性能を向上させることが可能となる。

【 0 0 1 6 】

本発明の別の側面に係る演算システムは、上述の演算装置であって、複数の演算装置を有する演算ユニットと、演算ユニットを共有する複数のプロセッサと、を備える。この演算システムでは、演算ユニットが複数の演算装置を備えるので、演算ユニットの回路規模を拡大することなく、演算ユニットの演算性能を向上させることが可能となる。

【 0 0 1 7 】

複数のプロセッサは、行列演算を行う場合には、単一のプロセッサとして動作してもよく、行列演算以外の演算を行う場合には、個別のプロセッサとして動作してもよい。この場合、行列演算を複数のプロセッサで並列処理することができるので、行列演算の演算速度を向上させることが可能となる。

【 0 0 1 8 】

複数のプロセッサは、1つのインストラクションで動作するSIMD動作を行ってもよい。この場合、複数のプロセッサが並列動作するので、処理の高速化が可能となる。

【 0 0 1 9 】

上記演算システムは、複数のプロセッサのそれぞれを一意に識別可能なプロセッサIDを含むメモリアドレスによってアクセス可能なメモリ空間を備えてもよい。複数のプロセッサのそれぞれは、当該プロセッサのプロセッサIDを含むメモリアドレスによって示されるメモリ領域にアクセス可能であってもよい。この場合、メモリ空間を拡張ことができ、演算システムで扱えるデータサイズを大きくすることが可能となる。

【 0 0 2 0 】

複数のプロセッサはリング結合されていてもよく、複数のプロセッサのそれぞれは、演算ユニットから受信したデータをリング結合を介して他のプロセッサに順に転送してもよい。この場合、複数のプロセッサのそれぞれは、演算ユニットから受信したデータをリング結合を介して他の要素プロセッサに順に転送（循環シフト）することができる。これにより、複数のプロセッサのそれぞれは、当該プロセッサが直接アクセスできないメモリ領域へのアクセス（unaligned access）を実行することが可能となる。

【 発明の効果 】**【 0 0 2 1 】**

本発明によれば、演算装置の回路規模を拡大することなく、演算装置の演算性能を向上させることができる。

【 図面の簡単な説明 】

10

20

30

40

50

【 0 0 2 2 】

【図 1】一実施形態に係る演算装置を含む演算システムの構成を概略的に示す図である。

【図 2】第 1 実施形態に係る演算装置の構成を概略的に示す図である。

【図 3】図 2 の演算装置に含まれる演算部の構成を概略的に示す図である。

【図 4】図 2 の演算装置の第 1 演算モードでの動作を説明するための図である。

【図 5】図 2 の演算装置の第 2 演算モードでの動作を説明するための図である。

【図 6】比較例の演算装置に含まれる演算部の構成を概略的に示す図である。

【図 7】図 3 の演算部の変形例を示す図である。

【図 8】第 2 実施形態に係る演算装置の構成を概略的に示す図である。

【図 9】図 8 の演算装置に含まれる演算部の構成を概略的に示す図である。

10

【図 10】図 9 の乗算器の構成を概略的に示す図である。

【図 11】シフト量演算回路の構成を概略的に示す図である。

【図 12】(a) は倍精度データの構成を示す図、(b) は単精度データの構成を示す図、(c) は半精度データの構成を示す図である。

【図 13】2 つのビット列から最大値を演算する最大値演算回路の一構成例を示す回路図である。

【図 14】図 13 の最大値演算回路の別の構成例を示す回路図である。

【図 15】複数のビット列から最大値を演算する最大値演算回路の構成例を示す図である。

【図 16】図 8 の演算装置の第 1 演算モードでの動作を説明するための図である。

20

【図 17】図 8 の演算装置の第 2 演算モードでの動作を説明するための図である。

【図 18】図 8 の演算装置の第 3 演算モードでの動作を説明するための図である。

【発明を実施するための形態】

【 0 0 2 3 】

以下、添付図面を参照しながら本発明の実施形態が詳細に説明される。図面の説明において、同一又は同等の要素には同一符号が用いられ、重複する説明は省略される。

【 0 0 2 4 】

図 1 は、一実施形態に係る演算装置を含む演算システムの構成を概略的に示す図である。図 1 に示される演算システム S は、行列演算及び他の演算を行うためのシステムである。演算システム S は、複数の要素プロセッサ（ここでは、4 つの要素プロセッサ P E 1 ~ P E 4 ）と、演算ユニット M A U と、を備える。

30

【 0 0 2 5 】

要素プロセッサ P E 1 ~ P E 4 は、単一の演算ユニット M A U を共有している。行列演算を行う際には、要素プロセッサ P E 1 ~ P E 4 が 1 つのプロセッサとして動作し、行列演算以外の演算を行う際には、要素プロセッサ P E 1 ~ P E 4 が個別のプロセッサとして動作する。行列演算以外の演算を行う際にも、要素プロセッサ P E 1 ~ P E 4 が 1 つのプロセッサとして動作してもよい。要素プロセッサ P E 1 ~ P E 4 が 1 つのプロセッサとして動作する場合、1 つのインストラクションで要素プロセッサ P E 1 ~ P E 4 が動作するので、要素プロセッサ P E 1 ~ P E 4 は S I M D 動作を行っているといみなされ得る。

【 0 0 2 6 】

40

要素プロセッサ P E 1 ~ P E 4 は、リング結合されている。具体的には、要素プロセッサ P E 1 及び要素プロセッサ P E 2、要素プロセッサ P E 2 及び要素プロセッサ P E 3、要素プロセッサ P E 3 及び要素プロセッサ P E 4、要素プロセッサ P E 4 及び要素プロセッサ P E 1 がデータバスでそれぞれ接続されている。要素プロセッサ P E 1 ~ P E 4 には、要素プロセッサを一意に識別可能なプロセッサ I D (0 0 , 0 1 , 1 0 , 1 1) が予め設定されている。

【 0 0 2 7 】

演算システム S には、メモリ空間が設定されている。メモリ空間のメモリアドレスは、プロセッサ I D を含む。本実施形態では、演算システム S は、4 つの要素プロセッサを含むので、例えばメモリアドレスの下位 2 ビットがプロセッサ I D に対応する。要素プロセ

50

ッサPE1～PE4のそれぞれには、例えば、10ビットのアドレスに対応するメモリ空間が割り当てられている。このため、演算システムSのメモリ空間は、12ビットに拡大されている。つまり、演算システムSのメモリ空間が4等分され、各メモリアドレスによって示されるメモリ領域には、当該メモリアドレスに含まれるプロセッサIDによって示される要素プロセッサがアクセスする。

【0028】

要素プロセッサPE1～PE4のそれぞれは、当該要素プロセッサのプロセッサIDを含むメモリアドレスによって示されるメモリ領域（担当領域）へのアクセス（align access）を実行できるが、当該要素プロセッサのプロセッサIDとは異なるプロセッサIDを含むメモリアドレスによって示されるメモリ領域（非担当領域）へのアクセス（unaligned access）を実行できない。要素プロセッサPE1～PE4のそれぞれは、リング結合を介して他の要素プロセッサに順にデータ転送（循環シフト）する。これにより、要素プロセッサPE1～PE4のそれぞれは、非担当領域へのアクセスを実行することが可能となる。要素プロセッサPE1～PE4のそれぞれは、メモリアドレスによって示される領域に、演算ユニットMAUによる演算対象である入力データ、及び演算ユニットMAUによる演算結果である出力データを保持する。

【0029】

演算ユニットMAUは、例えば、複数の浮動小数点演算を並列化して行う。演算ユニットMAUは、例えば、ディープニューラルネットワーク（DNN：Deep Neural Network）、及びコンボリユーションアルニューラルネットワーク（CNN：Convolutional Neural Network）向けのプロセッサに適用され得る。演算ユニットMAUは、複数の演算装置（ここでは、演算装置1A～1D）を備える。演算装置1A～1Dのそれぞれは、演算ユニットMAUの演算の一部を行う装置である。要素プロセッサPE1は、演算装置1Aのレジスタにアクセス可能である。要素プロセッサPE2は、演算装置1Bのレジスタにアクセス可能である。要素プロセッサPE3は、演算装置1Cのレジスタにアクセス可能である。要素プロセッサPE4は、演算装置1Dのレジスタにアクセス可能である。

【0030】

（第1実施形態）

図2は、第1実施形態に係る演算装置の構成を概略的に示す図である。図2に示される演算装置1Aは、複数のデータに対し同一の演算を同時に行うことが可能なSIMD演算器である。なお、演算装置1A～1Dの構成は同等であるので、ここでは演算装置1Aについて説明する。

【0031】

演算装置1Aは、MODE信号（モード信号）に応じて、演算装置1Aの演算モードを第1演算モード及び第2演算モードのいずれかに切り替える。第1演算モードは、第1ビット数のデータを用いた演算を行うモードである。第2演算モードは、第2ビット数のデータを用いた演算を行うモードである。第2ビット数は、第1ビット数の半分のビット数である。MODE信号は、演算装置1Aの外部の装置（本実施形態では、要素プロセッサ）から供給される。MODE信号は、演算装置1Aの演算モードを示す信号である。MODE信号は、例えば1ビットの信号であり、第1演算モード及び第2演算モードのいずれかを示す。

【0032】

演算装置1Aが実行し得る演算モードとしては、例えば、倍精度演算モード、単精度演算モード、及び半精度演算モードが挙げられる。倍精度演算モードは、単精度のデータ（以下、「単精度データ」という。）のビット数の2倍のビット数のデータを用いて演算を行うモードである。単精度演算モードは、倍精度のデータ（以下、「倍精度データ」という。）のビット数の半分のビット数のデータを用いて演算を行うモードである。半精度演算モードは、倍精度データのビット数の4分の1であり、単精度データのビット数の半分のビット数のデータを用いて演算を行うモードである。32ビットを1ワードとする32ビットアーキテクチャでは、倍精度データのビット数は64ビット、単精度データのピッ

ト数は32ビット、半精度のデータ（以下、「半精度データ」という。）のビット数は16ビットである。つまり、第1演算モードが倍精度演算モードである場合、第2演算モードは単精度演算モードである。また、第1演算モードが単精度演算モードである場合、第2演算モードは半精度演算モードである。

【0033】

演算装置1Aは、複数の演算部（本実施形態では演算部10A～10D）と、レジスタ40と、加算部50と、を備える。複数の演算部10A～10Dのそれぞれは、演算装置1Aの演算の一部を行う回路である。

【0034】

図3は、図2の演算装置に含まれる演算部の構成を概略的に示す図である。なお、演算部10A～10Dの構成は同等であるので、ここでは演算部10Aについて説明する。図3に示されるように、演算部10Aは、データマルチプレクサ11（マルチプレクサ）と、乗算器12（第1乗算器）と、乗算器13（第2乗算器）と、乗算器14（第3乗算器）と、乗算器15（第4乗算器）と、レジスタ41と、レジスタ42と、レジスタ43と、を備える。データを送受信する2つの回路要素間は、送受信するデータのビット数に対応するバス幅のデータバスで接続されている。

【0035】

レジスタ41～43のそれぞれは、演算対象となる第1ビット数のビット列である入力データDIN1～DIN3（第1～第3入力データ）を記憶保持する回路である。第1ビット数は、第1演算モードで用いられるデータのビット数に相当し、例えば64ビットである。入力データDIN1～DIN3は、演算装置1Aの外部の装置（本実施形態では、要素プロセッサ）からレジスタ41～43にセットされる。入力データDIN1（第1入力データ）は、上位データdin1_h（第1上位データ）と、下位データdin1_l（第1下位データ）と、を含む。上位データdin1_hは、入力データDIN1のビット列のうちの最上位ビット（most significant bit：MSB）を含む上半分のビット列である。下位データdin1_lは、入力データDIN1のビット列のうちの最下位ビット（least significant bit：LSB）を含む下半分のビット列である。

【0036】

入力データDIN2（第2入力データ）は、上位データdin2_h（第2上位データ）と、下位データdin2_l（第2下位データ）と、を含む。上位データdin2_hは、入力データDIN2のビット列のうちのMSBを含む上半分のビット列である。下位データdin2_lは、入力データDIN2のビット列のうちのLSBを含む下半分のビット列である。入力データDIN3（第3入力データ）は、上位データdin3_h（第3上位データ）と、下位データdin3_l（第3下位データ）と、を含む。上位データdin3_hは、入力データDIN3のビット列のうちのMSBを含む上半分のビット列である。下位データdin3_lは、入力データDIN3のビット列のうちのLSBを含む下半分のビット列である。

【0037】

上位データdin1_h、下位データdin1_l、上位データdin2_h、下位データdin2_l、上位データdin3_h、及び下位データdin3_lは、第2ビット数のビット列である。第2ビット数は、第2演算モードで用いられるデータのビット数に相当し、例えば32ビットである。

【0038】

データマルチプレクサ11は、レジスタ41～43から出力されている入力データDIN1～DIN3を入力し、MODE信号に応じて、複数（ここでは8つ）の出力データdout1～dout8（第1～第8出力データ）を出力する回路である。複数の出力データdout1～dout8のそれぞれは、第2ビット数のビット列である。

【0039】

データマルチプレクサ11は、MODE信号が第1演算モードを示す場合には、上位データdin1_hを出力データdout1（第1出力データ）及び出力データdout5

(第5出力データ)として出力し、下位データ d_{in1_l} を出力データ d_{out3} (第3出力データ) 及び出力データ d_{out7} (第7出力データ)として出力する。また、データマルチプレクサ11は、MODE信号が第1演算モードを示す場合には、上位データ d_{in2_h} を出力データ d_{out2} (第2出力データ) 及び出力データ d_{out4} (第4出力データ)として出力し、下位データ d_{in2_l} を出力データ d_{out6} (第6出力データ) 及び出力データ d_{out8} (第8出力データ)として出力する。データマルチプレクサ11は、MODE信号が第1演算モードを示す場合には、入力データ D_{IN3} を受け取らない。

【0040】

データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データ d_{in1_h} を出力データ d_{out1} 及び出力データ d_{out5} として出力し、下位データ d_{in1_l} を出力データ d_{out3} 及び出力データ d_{out7} として出力する。また、データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データ d_{in2_h} を出力データ d_{out2} として出力し、下位データ d_{in2_l} を出力データ d_{out4} として出力する。また、データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データ d_{in3_h} を出力データ d_{out6} として出力し、下位データ d_{in3_l} を出力データ d_{out8} として出力する。

【0041】

乗算器12~15は、第2ビット数を有する2つのビット列の乗算を行う回路である。乗算器12は、出力データ d_{out1} と出力データ d_{out2} との乗算を行う。乗算器12は、例えば、ウォレスツリー (Wallace Tree) 回路を用いて2つのビット列の乗算を行い、乗算器12の乗算の途中結果である中間結果 m_{11} (第1中間結果) 及び中間結果 m_{12} (第2中間結果) を乗算結果 (第1乗算結果) として出力する。中間結果 m_{11} 及び中間結果 m_{12} は、第2ビット数のビット列である。中間結果 m_{11} 及び中間結果 m_{12} は、乗算器12の演算遅延を生じさせない値であり、例えば、桁上がりを生じさせない部分的な合計値である。乗算器12は、中間結果 m_{11} を整列部21に出力し、中間結果 m_{12} を整列部22に出力するとともに、中間結果 m_{11} 及び中間結果 m_{12} を部分加算器16に出力する。

【0042】

乗算器13は、出力データ d_{out3} と出力データ d_{out4} との乗算を行う。乗算器13は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器13の乗算の途中結果である中間結果 m_{21} (第3中間結果) 及び中間結果 m_{22} (第4中間結果) を乗算結果 (第2乗算結果) として出力する。中間結果 m_{21} 及び中間結果 m_{22} は、第2ビット数のビット列である。中間結果 m_{21} 及び中間結果 m_{22} は、乗算器13の演算遅延を生じさせない値であり、例えば、桁上がりを生じさせない部分的な合計値である。乗算器13は、中間結果 m_{21} を整列部23に出力し、中間結果 m_{22} を整列部24に出力するとともに、中間結果 m_{21} 及び中間結果 m_{22} を部分加算器16に出力する。

【0043】

乗算器14は、出力データ d_{out5} と出力データ d_{out6} との乗算を行う。乗算器14は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器14の乗算の途中結果である中間結果 m_{31} (第5中間結果) 及び中間結果 m_{32} (第6中間結果) を乗算結果 (第3乗算結果) として出力する。中間結果 m_{31} 及び中間結果 m_{32} は、第2ビット数のビット列である。中間結果 m_{31} 及び中間結果 m_{32} は、乗算器14の演算遅延を生じさせない値であり、例えば、桁上がりを生じさせない部分的な合計値である。乗算器14は、中間結果 m_{31} を整列部25に出力し、中間結果 m_{32} を整列部26に出力するとともに、中間結果 m_{31} 及び中間結果 m_{32} を部分加算器16に出力する。

【0044】

乗算器15は、出力データ d_{out7} と出力データ d_{out8} との乗算を行う。乗算器

15は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器15の乗算の途中結果である中間結果m41（第7中間結果）及び中間結果m42（第8中間結果）を乗算結果（第4乗算結果）として出力する。中間結果m41及び中間結果m42は、第2ビット数のビット列である。中間結果m41及び中間結果m42は、乗算器15の演算遅延を生じさせない値であり、例えば、桁上がりを生じさせない部分的な合計値である。乗算器15は、中間結果m41を整列部27に出力し、中間結果m42を整列部28に出力するとともに、中間結果m41及び中間結果m42を部分加算器16に出力する。

【0045】

演算部10Aは、第1演算モードでの演算のために、部分加算器（partial adder）16と、整列部17と、整列部18と、丸め処理部19と、丸め処理部20と、をさらに備える。

【0046】

部分加算器16は、第1演算モードの乗算結果を生成する回路である。具体的には、部分加算器16は、乗算器12～15の乗算結果の加算を行う。本実施形態では、部分加算器16は、中間結果m11、中間結果m12、中間結果m21、中間結果m22、中間結果m31、中間結果m32、中間結果m41、及び中間結果m42の加算を行う。部分加算器16は、部分加算器16の加算の途中結果である中間結果P11及び中間結果P12を加算結果として出力する。中間結果P11及び中間結果P12は、第1ビット数のビット列である。中間結果P11及び中間結果P12は、部分加算器16の演算遅延を生じさせない値であり、例えば、桁上がりを生じさせない部分的な合計値である。

【0047】

整列部17は、後述の加算器51（第3加算器）によって加算されるビット列の指数部を揃えるために、中間結果P11の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部17は、整列部17の処理結果を丸め処理部19に出力する。整列部18は、加算器51によって加算されるビット列の指数部を揃えるために、中間結果P12の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部18は、整列部18の処理結果を丸め処理部20に出力する。

【0048】

丸め処理部19は、整列部17によってシフト処理が行われた中間結果P11に対して丸め処理を行う回路である。丸め処理部19は、丸め処理部19の演算結果R11を加算部50（加算器51）に出力する。演算結果R11は、第1ビット数のビット列である。丸め処理部20は、整列部18によってシフト処理が行われた中間結果P12に対して丸め処理を行う回路である。丸め処理部20は、丸め処理部20の演算結果R12を加算部50（加算器51）に出力する。演算結果R12は、第1ビット数のビット列である。

【0049】

演算部10Aは、第2演算モードでの演算のために、整列部21～28と、丸め処理部31～38と、をさらに備える。

【0050】

整列部21は、後述の加算器52（第1加算器）によって加算されるビット列の指数部を揃えるために、中間結果m11の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部21は、整列部21の処理結果を丸め処理部31に出力する。整列部22は、加算器52によって加算されるビット列の指数部を揃えるために、中間結果m12の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部22は、整列部22の処理結果を丸め処理部32に出力する。整列部23は、加算器52によって加算されるビット列の指数部を揃えるために、中間結果m21の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部23は、整列部23の処理結果を丸め処理部33に出力する。整列部24は、加算器52によって加算されるビット列の指数部を揃えるために、中間結果m22の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部24は、整列部24の処理結果を丸め処理部34に出力する。

【 0 0 5 1 】

整列部 2 5 は、後述の加算器 5 3（第 2 加算器）によって加算されるビット列の指数部を揃えるために、中間結果 m 3 1 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 2 5 は、整列部 2 5 の処理結果を丸め処理部 3 5 に出力する。整列部 2 6 は、加算器 5 3 によって加算されるビット列の指数部を揃えるために、中間結果 m 3 2 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 2 6 は、整列部 2 6 の処理結果を丸め処理部 3 6 に出力する。整列部 2 7 は、加算器 5 3 によって加算されるビット列の指数部を揃えるために、中間結果 m 4 1 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 2 7 は、整列部 2 7 の処理結果を丸め処理部 3 7 に出力する。整列部 2 8 は、加算器 5 3 によって加算されるビット列の指数部を揃えるために、中間結果 m 4 2 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 2 8 は、整列部 2 8 の処理結果を丸め処理部 3 8 に出力する。

【 0 0 5 2 】

丸め処理部 3 1 は、整列部 2 1 によってシフト処理が行われた中間結果 m 1 1 に対して丸め処理を行う回路である。丸め処理部 3 1 は、丸め処理部 3 1 の演算結果 r 1 1 を加算部 5 0（加算器 5 2）に出力する。丸め処理部 3 2 は、整列部 2 2 によってシフト処理が行われた中間結果 m 1 2 に対して丸め処理を行う回路である。丸め処理部 3 2 は、丸め処理部 3 2 の演算結果 r 1 2 を加算部 5 0（加算器 5 2）に出力する。丸め処理部 3 3 は、整列部 2 3 によってシフト処理が行われた中間結果 m 2 1 に対して丸め処理を行う回路である。丸め処理部 3 3 は、丸め処理部 3 3 の演算結果 r 2 1 を加算部 5 0（加算器 5 2）に出力する。丸め処理部 3 4 は、整列部 2 4 によってシフト処理が行われた中間結果 m 2 2 に対して丸め処理を行う回路である。丸め処理部 3 4 は、丸め処理部 3 4 の演算結果 r 2 2 を加算部 5 0（加算器 5 2）に出力する。演算結果 r 1 1, r 1 2, r 2 1, r 2 2 のそれぞれは、第 2 ビット数のビット列である。

【 0 0 5 3 】

丸め処理部 3 5 は、整列部 2 5 によってシフト処理が行われた中間結果 m 3 1 に対して丸め処理を行う回路である。丸め処理部 3 5 は、丸め処理部 3 5 の演算結果 r 3 1 を加算部 5 0（加算器 5 3）に出力する。丸め処理部 3 6 は、整列部 2 6 によってシフト処理が行われた中間結果 m 3 2 に対して丸め処理を行う回路である。丸め処理部 3 6 は、丸め処理部 3 6 の演算結果 r 3 2 を加算部 5 0（加算器 5 3）に出力する。丸め処理部 3 7 は、整列部 2 7 によってシフト処理が行われた中間結果 m 4 1 に対して丸め処理を行う回路である。丸め処理部 3 7 は、丸め処理部 3 7 の演算結果 r 4 1 を加算部 5 0（加算器 5 3）に出力する。丸め処理部 3 8 は、整列部 2 8 によってシフト処理が行われた中間結果 m 4 2 に対して丸め処理を行う回路である。丸め処理部 3 8 は、丸め処理部 3 8 の演算結果 r 4 2 を加算部 5 0（加算器 5 3）に出力する。演算結果 r 3 1, r 3 2, r 4 1, r 4 2 のそれぞれは、第 2 ビット数のビット列である。

【 0 0 5 4 】

図 2 に戻って、演算装置 1 A の説明を続ける。加算部 5 0 は、加算器（final adder）5 1 と、加算器（final adder）5 2 と、加算器（final adder）5 3 と、マルチプレクサ 5 9（セレクタ）と、を備える。レジスタ 4 0 は、演算対象となる第 1 ビット数のビット列である入力データ D I N 0 を記憶保持する回路である。入力データ D I N 0 は、演算装置 1 A の外部の装置（本実施形態では、要素プロセッサ）からレジスタ 4 0 にセットされる。入力データ D I N 0 は、上位データ d i n 0 _ h と、下位データ d i n 0 _ l と、を含む。上位データ d i n 0 _ h、及び下位データ d i n 0 _ l は、第 2 ビット数のビット列である。レジスタ 4 0 は、上位データ d i n 0 _ h を加算器 5 2 に出力し、下位データ d i n 0 _ l を加算器 5 3 に出力するとともに、入力データ D I N 0 を加算器 5 1 に出力する。

【 0 0 5 5 】

加算器 5 1 は、第 1 演算モードの演算結果 D 0（第 3 加算結果）を生成する回路である。具体的には、演算部 1 0 A ~ 1 0 D の部分加算器 1 6 の加算結果と入力データ D I N 0

との加算を行う。本実施形態では、加算器 5 1 は、演算部 1 0 A ~ 1 0 D の演算結果 R 1 1 と、演算部 1 0 A ~ 1 0 D の演算結果 R 1 2 と、入力データ D I N 0 との加算を行う。加算器 5 1 は、加算器 5 1 の加算結果を演算結果 D 0 としてマルチプレクサ 5 9 に出力する。演算結果 D 0 は、第 1 ビット数のビット列である。

【 0 0 5 6 】

加算器 5 2 は、第 2 演算モードの演算結果 d 0 (第 1 加算結果) を生成する回路である。具体的には、加算器 5 2 は、各演算部 1 0 A ~ 1 0 D の乗算器 1 2 の乗算結果及び乗算器 1 3 の乗算結果と、上位データ d i n 0 _ h との加算を行う。本実施形態では、加算器 5 2 は、演算部 1 0 A ~ 1 0 D の演算結果 r 1 1 , r 1 2 , r 2 1 , r 2 2 と、上位データ d i n 0 _ h との加算を行う。加算器 5 2 は、加算器 5 2 の加算結果を演算結果 d 0 としてマルチプレクサ 5 9 に出力する。演算結果 d 0 は、第 2 ビット数のビット列である。

【 0 0 5 7 】

加算器 5 3 は、第 2 演算モードの演算結果 d 1 (第 2 加算結果) を生成する回路である。具体的には、加算器 5 3 は、各演算部 1 0 A ~ 1 0 D の乗算器 1 4 の乗算結果及び乗算器 1 5 の乗算結果と、下位データ d i n 0 _ l との加算を行う。本実施形態では、加算器 5 3 は、演算部 1 0 A ~ 1 0 D の演算結果 r 3 1 , r 3 2 , r 4 1 , r 4 2 と、下位データ d i n 0 _ l との加算を行う。加算器 5 3 は、加算器 5 3 の加算結果を演算結果 d 1 としてマルチプレクサ 5 9 に出力する。演算結果 d 1 は、第 2 ビット数のビット列である。なお、演算結果 d 0 及び演算結果 d 1 が組み合わせられて演算結果 (d 0 , d 1) が生成される。演算結果 (d 0 , d 1) は、演算結果 d 0 を上位ビットとし、演算結果 d 1 を下位ビットとするビット列である。つまり、演算結果 (d 0 , d 1) は、第 1 ビット数のビット列である。

【 0 0 5 8 】

マルチプレクサ 5 9 は、演算結果 D 0 及び演算結果 (d 0 , d 1) を入力し、MODE 信号に応じて、演算結果 D 0 と演算結果 (d 0 , d 1) とのいずれかを出力する回路である。マルチプレクサ 5 9 は、MODE 信号が第 1 演算モードを示す場合には、演算結果 D 0 を出力する。マルチプレクサ 5 9 は、MODE 信号が第 2 演算モードを示す場合には、演算結果 (d 0 , d 1) を出力する。

【 0 0 5 9 】

次に、図 2 ~ 図 4 を用いて演算装置 1 A ~ 1 D の第 1 演算モードでの動作を説明する。図 4 は、図 2 の演算装置の第 1 演算モードでの動作を説明するための図である。ここでは、第 1 演算モードとして、倍精度演算モードを用い、演算ユニット M A U が式 (1) に示される倍精度の行列演算を行う場合を一例として説明する。この行列演算は、倍精度データ A 0 ~ A 3、倍精度データ B 0 0 ~ B 0 3 , B 1 0 ~ B 1 3 , B 2 0 ~ B 2 3 , B 3 0 ~ B 3 3、及び倍精度データ C 0 ~ C 3 を用いた演算である。

【数 1】

$$(A_0 \ A_1 \ A_2 \ A_3) \times \begin{pmatrix} B_{00} & B_{01} & B_{02} & B_{03} \\ B_{10} & B_{11} & B_{12} & B_{13} \\ B_{20} & B_{21} & B_{22} & B_{23} \\ B_{30} & B_{31} & B_{32} & B_{33} \end{pmatrix} + (C_0 \ C_1 \ C_2 \ C_3) \ \dots \quad (1)$$

【 0 0 6 0 】

演算装置 1 A ~ 1 D はそれぞれ、式 (2) ~ (5) に示される行列演算を行う。

【数 2】

$$\begin{aligned} D_0 &= (A_0, A_1, A_2, A_3) \times (B_{00}, B_{10}, B_{20}, B_{30}) + C_0 \\ &= A_0 \times B_{00} + A_1 \times B_{10} + A_2 \times B_{20} + A_3 \times B_{30} + C_0 \ \dots \quad (2) \end{aligned}$$

【数 3】

$$D1 = (A0, A1, A2, A3) \times (B01, B11, B21, B31) + C1$$

$$= A0 \times B01 + A1 \times B11 + A2 \times B21 + A3 \times B31 + C1 \quad \dots \quad (3)$$

【数 4】

$$D2 = (A0, A1, A2, A3) \times (B02, B12, B22, B32) + C2$$

$$= A0 \times B02 + A1 \times B12 + A2 \times B22 + A3 \times B32 + C2 \quad \dots \quad (4)$$

【数 5】

$$D3 = (A0, A1, A2, A3) \times (B03, B13, B23, B33) + C3$$

$$= A0 \times B03 + A1 \times B13 + A2 \times B23 + A3 \times B33 + C3 \quad \dots \quad (5)$$

【0061】

演算装置 1 A ~ 1 D では、演算対象のデータが異なるものの、動作は同じであるので、ここでは演算装置 1 A について説明する。演算装置 1 A の演算部 1 0 A は、 $A0 \times B00$ の演算を行う。演算装置 1 A の演算部 1 0 B は、 $A1 \times B10$ の演算を行う。演算装置 1 A の演算部 1 0 C は、 $A2 \times B20$ の演算を行う。演算装置 1 A の演算部 1 0 D は、 $A3 \times B30$ の演算を行う。各演算部 1 0 A ~ 1 0 D では、演算対象のデータが異なるものの、動作は同じであるので、演算部 1 0 A の動作を主に説明する。

【0062】

倍精度データ A 0 は、上位データ $a0_h$ と、下位データ $a0_l$ と、を含む。上位データ $a0_h$ は、倍精度データ A 0 のビット列のうちの MSB を含む上半分のビット列であり、単精度データと同じビット数を有する。下位データ $a0_l$ は、倍精度データ A 0 のビット列のうちの LSB を含む下半分のビット列であり、単精度データと同じビット数を有する。倍精度データ B 0 0 は、上位データ $b00_h$ と、下位データ $b00_l$ と、を含む。上位データ $b00_h$ は、倍精度データ B 0 0 のビット列のうちの MSB を含む上半分のビット列であり、単精度データと同じビット数を有する。下位データ $a0_l$ は、倍精度データ B 0 0 のビット列のうちの LSB を含む下半分のビット列であり、単精度データと同じビット数を有する。

【0063】

演算部 1 0 A は、式 (6) に示されるように、倍精度データ A 0 及び倍精度データ B 0 0 のそれぞれを単精度データのビット数を有する 2 つのデータに分解することによって、倍精度データ A 0 及び倍精度データ B 0 0 の乗算を行う。

【数 6】

$$A0 \times B00 = (a0_h + a0_l) \times (b00_h + b00_l)$$

$$= a0_h \times b00_h + a0_l \times b00_h + a0_h \times b00_l + a0_l \times b00_l \quad \dots \quad (6)$$

【0064】

以下、具体的に説明する。レジスタ 4 1 には、倍精度データ A 0 が記憶されており、レジスタ 4 1 からデータマルチプレクサ 1 1 に入力データ DIN 1 として倍精度データ A 0 が供給されている。レジスタ 4 2 には、倍精度データ B 0 0 が記憶されており、レジスタ 4 2 からデータマルチプレクサ 1 1 に入力データ DIN 2 として倍精度データ B 0 0 が供給されている。レジスタ 4 3 には、有効なデータは記憶されていない。

【0065】

データマルチプレクサ 1 1 には、倍精度演算モード (第 1 演算モード) を示す MODE 信号が供給されている。このため、データマルチプレクサ 1 1 は、上位データ $a0_h$ を出力データ dout 1 及び出力データ dout 5 として出力し、下位データ $a0_l$ を出力データ dout 3 及び出力データ dout 7 として出力する。また、データマルチプレ

クサ 1 1 は、上位データ b 0 0 _ h を出力データ d o u t 2 及び出力データ d o u t 4 と
して出力し、下位データ b 0 0 _ l を出力データ d o u t 6 及び出力データ d o u t 8 と
して出力する。

【 0 0 6 6 】

そして、乗算器 1 2 は、上位データ a 0 _ h と上位データ b 0 0 _ h との乗算を行い、
中間結果 m 1 1 及び中間結果 m 1 2 を出力する。同様に、乗算器 1 3 は、下位データ a 0
_ l と上位データ b 0 0 _ h との乗算を行い、中間結果 m 2 1 及び中間結果 m 2 2 を出力
する。同様に、乗算器 1 4 は、上位データ a 0 _ h と下位データ b 0 0 _ l との乗算を行
い、中間結果 m 3 1 及び中間結果 m 3 2 を出力する。同様に、乗算器 1 5 は、下位データ
a 0 _ l と下位データ b 0 0 _ l との乗算を行い、中間結果 m 4 1 及び中間結果 m 4 2 を
出力する。 10

【 0 0 6 7 】

そして、部分加算器 1 6 は、中間結果 m 1 1、中間結果 m 1 2、中間結果 m 2 1、中間
結果 m 2 2、中間結果 m 3 1、中間結果 m 3 2、中間結果 m 4 1、及び中間結果 m 4 2 の
加算を行い、中間結果 P 1 1 及び中間結果 P 1 2 を出力する。そして、整列部 1 7 は、中
間結果 P 1 1 の仮数部に対してシフト処理を行い、整列部 1 7 の処理結果を丸め処理部 1
9 に出力する。同様に、整列部 1 8 は、中間結果 P 1 2 の仮数部に対してシフト処理を行
い、整列部 1 8 の処理結果を丸め処理部 2 0 に出力する。

【 0 0 6 8 】

そして、丸め処理部 1 9 は、整列部 1 7 によってシフト処理が行われた中間結果 P 1 1
に対して丸め処理を行い、演算結果 R 1 1 を加算器 5 1 に出力する。同様に、丸め処理部
2 0 は、整列部 1 8 によってシフト処理が行われた中間結果 P 1 2 に対して丸め処理を行
い、演算結果 R 1 2 を加算器 5 1 に出力する。 20

【 0 0 6 9 】

演算部 1 0 B、演算部 1 0 C、及び演算部 1 0 D においても、同様の演算が行われ、各
演算部 1 0 A ~ 1 0 D は、演算結果 R 1 1 及び演算結果 R 1 2 を加算器 5 1 に出力する。
また、レジスタ 4 0 には、倍精度データ C 0 が記憶されており、レジスタ 4 0 から加算器
5 1 に倍精度データ C 0 が供給されている。

【 0 0 7 0 】

そして、加算器 5 1 は、演算部 1 0 A ~ 1 0 D の演算結果 R 1 1 及び演算結果 R 1 2、
並びにレジスタ 4 0 から供給されている倍精度データ C 0 の加算を行い、加算器 5 1 の加
算結果を演算結果 D 0 としてマルチプレクサ 5 9 に出力する。このとき、整列部 2 1 ~ 2
8、丸め処理部 3 1 ~ 3 8、加算器 5 2、及び加算器 5 3 も演算を行っており、演算結果
(d 0 , d 1) がマルチプレクサ 5 9 に出力されている。マルチプレクサ 5 9 には、倍精
度演算モードを示す M O D E 信号が供給されているので、マルチプレクサ 5 9 は、演算結
果 D 0 を出力する。このようにして、倍精度の行列演算が行われる。 30

【 0 0 7 1 】

次に、図 2、図 3、及び図 5 を用いて演算装置 1 A ~ 1 D の第 2 演算モードでの動作を
説明する。図 5 は、図 2 の演算装置の第 2 演算モードでの動作を説明するための図である
。ここでは、第 2 演算モードとして、単精度演算モードを用い、演算ユニット M A U が式
(7) に示される単精度の行列演算を行う場合を一例として説明する。式 (7) に示され
る行列演算は、単精度データ a 0 ~ a 7、単精度データ b 0 0 ~ b 0 7、b 1 0 ~ b 1 7
, b 2 0 ~ b 2 7、b 3 0 ~ b 3 7、b 4 0 ~ b 4 7、b 5 0 ~ b 5 7、b 6 0 ~ b 6 7
, b 7 0 ~ b 7 7、及び単精度データ c 0 ~ c 7 を用いた演算である。 40

【数 7】

$$(a_0 \ a_1 \ \dots \ a_7) \times \begin{pmatrix} b_{00} & b_{01} & \dots & b_{07} \\ b_{10} & b_{11} & \dots & b_{17} \\ \vdots & \vdots & \ddots & \vdots \\ b_{70} & b_{71} & \dots & b_{77} \end{pmatrix} + (c_0 \ c_1 \ \dots \ c_7) \ \dots \quad (7)$$

【0072】

演算装置 1 A は、式 (8) 及び式 (9) に示される行列演算を行う。

【数 8】

$$d_0 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{00}, b_{10}, b_{20}, b_{30}, b_{40}, b_{50}, b_{60}, b_{70}) + c_0 \ \dots \quad (8)$$

【数 9】

$$d_1 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{01}, b_{11}, b_{21}, b_{31}, b_{41}, b_{51}, b_{61}, b_{71}) + c_1 \ \dots \quad (9)$$

【0073】

演算装置 1 B は、式 (10) 及び式 (11) に示される行列演算を行う。

【数 10】

$$d_2 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{02}, b_{12}, b_{22}, b_{32}, b_{42}, b_{52}, b_{62}, b_{72}) + c_2 \ \dots \quad (10)$$

【数 11】

$$d_3 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{03}, b_{13}, b_{23}, b_{33}, b_{43}, b_{53}, b_{63}, b_{73}) + c_3 \ \dots \quad (11)$$

【0074】

演算装置 1 C は、式 (12) 及び式 (13) に示される行列演算を行う。

【数 12】

$$d_4 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{04}, b_{14}, b_{24}, b_{34}, b_{44}, b_{54}, b_{64}, b_{74}) + c_4 \ \dots \quad (12)$$

【数 13】

$$d_5 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{05}, b_{15}, b_{25}, b_{35}, b_{45}, b_{55}, b_{65}, b_{75}) + c_5 \ \dots \quad (13)$$

【0075】

演算装置 1 D は、式 (14) 及び式 (15) に示される行列演算を行う。

【数 14】

$$d_6 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{06}, b_{16}, b_{26}, b_{36}, b_{46}, b_{56}, b_{66}, b_{76}) + c_6 \ \dots \quad (14)$$

【数 15】

$$d_7 = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7) \times (b_{07}, b_{17}, b_{27}, b_{37}, b_{47}, b_{57}, b_{67}, b_{77}) + c_7 \ \dots \quad (15)$$

【0076】

演算装置 1 A ~ 1 D では、演算対象のデータが異なるものの、動作は同じであるので、ここでは演算装置 1 A について説明する。演算装置 1 A の演算部 10 A は、 $a_0 \times b_{00} + a_1 \times b_{10}$ の演算、及び $a_0 \times b_{01} + a_1 \times b_{11}$ の演算を行う。演算装置 1 A の演算部 10 B は、 $a_2 \times b_{20} + a_3 \times b_{30}$ の演算、及び $a_2 \times b_{21} + a_3 \times b_{31}$ の演算を行う。演算装置 1 A の演算部 10 C は、 $a_4 \times b_{40} + a_5 \times b_{50}$ の演算、及び $a_4 \times b_{41} + a_5 \times b_{51}$ の演算を行う。演算装置 1 A の演算部 10 D は、 $a_6 \times b_{60} + a_7 \times b_{70}$ の演算、及び $a_6 \times b_{61} + a_7 \times b_{71}$ の演算を行う。

60 + a7 × b70 の演算、及び a6 × b61 + a7 × b71 の演算を行う。各演算部 10A ~ 10D では、演算対象のデータが異なるものの、動作は同じであるので、演算部 10A の動作を主に説明する。

【0077】

レジスタ41の上位ビットには、単精度データa0が記憶されており、レジスタ41の下位ビットには単精度データa1が記憶されている。レジスタ41からデータマルチプレクサ11に入力データDIN1として単精度データa0及び単精度データa1が供給されている。レジスタ42の上位ビットには、単精度データb00が記憶されており、レジスタ42の下位ビットには単精度データb10が記憶されている。レジスタ42からデータマルチプレクサ11に入力データDIN2として単精度データb00及び単精度データb10が供給されている。レジスタ43の上位ビットには、単精度データb01が記憶されており、レジスタ43の下位ビットには単精度データb11が記憶されている。レジスタ43からデータマルチプレクサ11に入力データDIN3として単精度データb01及び単精度データb11が供給されている。

10

【0078】

データマルチプレクサ11には、単精度演算モードを示すMODE信号が供給されている。このため、データマルチプレクサ11は、単精度データa0を出力データdout1及び出力データdout5として出力し、単精度データa1を出力データdout3及び出力データdout7として出力する。また、データマルチプレクサ11は、単精度データb00を出力データdout2として出力し、単精度データb10を出力データdout4として出力する。また、データマルチプレクサ11は、単精度データb01を出力データdout6として出力し、単精度データb11を出力データdout8として出力する。

20

【0079】

そして、乗算器12は、単精度データa0と単精度データb00との乗算を行い、中間結果m11及び中間結果m12を出力する。同様に、乗算器13は、単精度データa1と単精度データb10との乗算を行い、中間結果m21及び中間結果m22を出力する。同様に、乗算器14は、単精度データa0と単精度データb01との乗算を行い、中間結果m31及び中間結果m32を出力する。同様に、乗算器15は、単精度データa1と単精度データb11との乗算を行い、中間結果m41及び中間結果m42を出力する。

30

【0080】

そして、整列部21 ~ 28は、中間結果m11, m12, m21, m22, m31, m32, m41, m42の仮数部に対してシフト処理を行い、整列部21 ~ 28の処理結果を丸め処理部31 ~ 38に出力する。そして、丸め処理部31 ~ 34は、整列部21 ~ 24によってシフト処理が行われた中間結果m11, m12, m21, m22に対して丸め処理を行い、演算結果r11, r12, r21, r22を加算器52に出力する。同様に、丸め処理部35 ~ 38は、整列部25 ~ 28によってシフト処理が行われた中間結果m31, m32, m41, m42に対して丸め処理を行い、演算結果r31, r32, r41, r42を加算器53に出力する。

【0081】

演算部10B、演算部10C、及び演算部10Dにおいても、同様の演算が行われ、各演算部10A ~ 10Dは、演算結果r11, r12, r21, r22を加算器52に出力するとともに、演算結果r31, r32, r41, r42を加算器53に出力する。また、レジスタ40の上位ビットには、単精度データc0が記憶されており、レジスタ40の下位ビットには単精度データc1が記憶されている。そして、レジスタ40から加算器52に単精度データc0が供給されるとともに、加算器53に単精度データc1が供給されている。

40

【0082】

そして、加算器52は、演算部10A ~ 10Dの演算結果r11, r12, r21, r22、及びレジスタ40から供給されている単精度データc0の加算を行い、加算器52

50

の加算結果を演算結果 d_0 として出力する。また、加算器 53 は、演算部 10A ~ 10D の演算結果 r_{31} , r_{32} , r_{41} , r_{42} 、及びレジスタ 40 から供給されている単精度データ c_1 の加算を行い、加算器 53 の加算結果を演算結果 d_1 として出力する。そして、演算結果 d_0 及び演算結果 d_1 が組み合わされて演算結果 (d_0 , d_1) が生成され、マルチプレクサ 59 に供給される。このとき、部分加算器 16、整列部 17, 18、丸め処理部 19, 20、及び加算器 51 も演算を行っており、演算結果 D_0 がマルチプレクサ 59 に出力されている。マルチプレクサ 59 には、単精度演算モードを示す MODE 信号が供給されているので、マルチプレクサ 59 は、演算結果 (d_0 , d_1) を出力する。このようにして、2つの単精度の行列演算が行われる。

【0083】

次に、比較例の演算装置と比較して、演算装置 1A ~ 1D の作用効果を説明する。図 6 は、比較例の演算装置に含まれる演算部の構成を概略的に示す図である。図 6 に示される演算部 100 は、演算部 10A ~ 10D と比較して、レジスタ 43 を備えない点、データマルチプレクサ 11 に代えてデータマルチプレクサ 111 を備える点、整列部 21 ~ 28、及び丸め処理部 31 ~ 38 に代えて加算器 61 ~ 64、整列部 65 ~ 68、及び丸め処理部 69 ~ 72 を備える点、部分加算器 16 に代えて部分加算器 116 を備える点、並びに整列部 17, 18 及び丸め処理部 19, 20 に代えて整列部 117 及び丸め処理部 119 を備える点において主に相違する。

【0084】

データマルチプレクサ 111 には、MODE 信号が供給されない。このため、データマルチプレクサ 111 は、演算モードによらず、上位データ d_{in1_h} を出力データ d_{out1} 及び出力データ d_{out5} として出力し、下位データ d_{in1_l} を出力データ d_{out3} 及び出力データ d_{out7} として出力する。また、データマルチプレクサ 111 は、上位データ d_{in2_h} を出力データ d_{out2} 及び出力データ d_{out4} として出力し、下位データ d_{in2_l} を出力データ d_{out6} 及び出力データ d_{out8} として出力する。

【0085】

加算器 61 は、中間結果 m_{11} 、及び中間結果 m_{12} を加算することにより、乗算器 12 の乗算結果 m_1 を生成する回路である。加算器 61 は、乗算結果 m_1 を整列部 65 に出力する。加算器 62 は、中間結果 m_{21} 、及び中間結果 m_{22} を加算することにより、乗算器 13 の乗算結果 m_2 を生成する回路である。加算器 62 は、乗算結果 m_2 を整列部 66 に出力する。加算器 63 は、中間結果 m_{31} 、及び中間結果 m_{32} を加算することにより、乗算器 14 の乗算結果 m_3 を生成する回路である。加算器 63 は、乗算結果 m_3 を整列部 67 に出力する。加算器 64 は、中間結果 m_{41} 、及び中間結果 m_{42} を加算することにより、乗算器 15 の乗算結果 m_4 を生成する回路である。加算器 64 は、乗算結果 m_4 を整列部 68 に出力する。乗算結果 m_1 ~ m_4 のそれぞれは、第 2 ビット数のビット列である。

【0086】

整列部 65 は、部分加算器 116 によって加算されるビット列の指数部を揃えるために、乗算結果 m_1 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 65 は、整列部 65 の処理結果を丸め処理部 69 に出力する。整列部 66 は、部分加算器 116 によって加算されるビット列の指数部を揃えるために、乗算結果 m_2 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 66 は、整列部 66 の処理結果を丸め処理部 70 に出力する。整列部 67 は、部分加算器 116 によって加算されるビット列の指数部を揃えるために、乗算結果 m_3 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 67 は、整列部 67 の処理結果を丸め処理部 71 に出力する。整列部 68 は、部分加算器 116 によって加算されるビット列の指数部を揃えるために、乗算結果 m_4 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 68 は、整列部 68 の処理結果を丸め処理部 72 に出力する。

【0087】

10

20

30

40

50

丸め処理部 69 は、整列部 65 によってシフト処理が行われた乗算結果 m_1 に対して丸め処理を行う回路である。丸め処理部 69 は、丸め処理部 69 の演算結果 r_1 を部分加算器 116 に出力する。丸め処理部 70 は、整列部 66 によってシフト処理が行われた乗算結果 m_2 に対して丸め処理を行う回路である。丸め処理部 70 は、丸め処理部 70 の演算結果 r_2 を部分加算器 116 に出力する。丸め処理部 71 は、整列部 67 によってシフト処理が行われた乗算結果 m_3 に対して丸め処理を行う回路である。丸め処理部 71 は、丸め処理部 71 の演算結果 r_3 を部分加算器 116 に出力する。丸め処理部 72 は、整列部 68 によってシフト処理が行われた乗算結果 m_4 に対して丸め処理を行う回路である。丸め処理部 72 は、丸め処理部 72 の演算結果 r_4 を部分加算器 116 に出力する。演算結果 $r_1 \sim r_4$ のそれぞれは、第 2 ビット数のビット列である。

10

【0088】

部分加算器 116 は、演算結果 $r_1 \sim r_4$ の加算を行い、第 1 ビット数のビット列である加算結果 P_1 を出力する。整列部 117 は、後段の加算器（不図示）によって加算されるビット列の指数部を揃えるために、加算結果 P_1 の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 117 は、整列部 117 の処理結果を丸め処理部 119 に出力する。丸め処理部 119 は、整列部 117 によってシフト処理が行われた加算結果 P_1 に対して丸め処理を行う回路である。丸め処理部 119 は、丸め処理部 119 の演算結果 R_1 を後段の加算器に出力する。

【0089】

演算部 100 は、第 1 演算モードでは、演算部 10A ~ 10D と同様に、第 1 ビット数を有する 2 つのデータのそれぞれを第 2 ビット数を有する 2 つのデータに分解することによって、第 1 ビット数を有する 2 つのデータの乗算を行う。第 1 演算モードとして倍精度演算モードを用い、倍精度データ A_0 及び倍精度データ B_00 を用いた $A_0 \times B_00$ の演算を行う場合について説明する。図 4 と同様に、レジスタ 41 には、倍精度データ A_0 が記憶されており、レジスタ 41 からデータマルチプレクサ 111 に入力データ DIN_1 として倍精度データ A_0 が供給されている。レジスタ 42 には、倍精度データ B_00 が記憶されており、レジスタ 42 からデータマルチプレクサ 111 に入力データ DIN_2 として倍精度データ B_00 が供給されている。データマルチプレクサ 111 及び乗算器 12 ~ 15 の動作は、第 1 演算モードにおける演算部 10A ~ 10D のデータマルチプレクサ 111 及び乗算器 12 ~ 15 と同じであるので、説明を省略する。

20

30

【0090】

そして、加算器 61 は、中間結果 m_{11} 及び中間結果 m_{12} を加算することにより、乗算器 12 の乗算結果 m_1 を生成し、乗算結果 m_1 を整列部 65 に出力する。同様に、加算器 62 は、中間結果 m_{21} 及び中間結果 m_{22} を加算することにより、乗算器 13 の乗算結果 m_2 を生成し、乗算結果 m_2 を整列部 66 に出力する。同様に、加算器 63 は、中間結果 m_{31} 及び中間結果 m_{32} を加算することにより、乗算器 14 の乗算結果 m_3 を生成し、乗算結果 m_3 を整列部 67 に出力する。同様に、加算器 64 は、中間結果 m_{41} 及び中間結果 m_{42} を加算することにより、乗算器 14 の乗算結果 m_4 を生成し、乗算結果 m_4 を整列部 68 に出力する。

【0091】

そして、整列部 65 ~ 68 は、乗算結果 $m_1 \sim m_4$ の仮数部に対してシフト処理を行い、整列部 65 ~ 68 の処理結果を丸め処理部 69 ~ 72 に出力する。そして、丸め処理部 69 ~ 72 は、整列部 65 ~ 68 によってシフト処理が行われた乗算結果 $m_1 \sim m_4$ に対して丸め処理を行い、演算結果 $r_1 \sim r_4$ を部分加算器 116 に出力する。

40

【0092】

そして、部分加算器 116 は、演算結果 $r_1 \sim r_4$ の加算を行い、加算結果 P_1 を出力する。そして、整列部 117 は、加算結果 P_1 の仮数部に対してシフト処理を行い、整列部 117 の処理結果を丸め処理部 119 に出力する。そして、丸め処理部 119 は、整列部 117 によってシフト処理が行われた加算結果 P_1 に対して丸め処理を行い、演算結果 R_1 を後段の加算器に出力する。このようにして、 $A_0 \times B_00$ の演算結果 R_1 が得られ

50

る。

【 0 0 9 3 】

一方、第2演算モードでは、演算部100は、第2ビット数を有する4つのデータを用いて、2つの乗算を行う。第2演算モードとして単精度演算モードを用い、単精度データa0及び単精度データb00を用いた $a0 \times b00$ の演算と、単精度データa1及び単精度データb10を用いた $a1 \times b10$ の演算とを行う場合について説明する。レジスタ41の上位ビットには、単精度データa0が記憶されており、レジスタ41の下位ビットには単精度データa1が記憶されている。レジスタ41からデータマルチプレクサ11に入力データDIN1として単精度データa0及び単精度データa1が供給されている。レジスタ42の上位ビットには、単精度データb00が記憶されており、レジスタ42の下位ビットには単精度データb10が記憶されている。レジスタ42からデータマルチプレクサ11に入力データDIN2として単精度データb00及び単精度データb10が供給されている。

【 0 0 9 4 】

データマルチプレクサ111は、単精度データa0を出力データdout1及び出力データdout5として出力し、単精度データa1を出力データdout3及び出力データdout7として出力する。また、データマルチプレクサ111は、単精度データb00を出力データdout2及び出力データdout4として出力し、単精度データb10を出力データdout6及び出力データdout8として出力する。

【 0 0 9 5 】

そして、乗算器12は、単精度データa0と単精度データb00との乗算を行い、中間結果m11及び中間結果m12を出力する。同様に、乗算器13は、単精度データa1と単精度データb00との乗算を行い、中間結果m21及び中間結果m22を出力する。同様に、乗算器14は、単精度データa0と単精度データb10との乗算を行い、中間結果m31及び中間結果m32を出力する。同様に、乗算器15は、単精度データa1と単精度データb10との乗算を行い、中間結果m41及び中間結果m42を出力する。加算器61~64、整列部65~68、及び丸め処理部69~72の動作は、第1演算モードと同様であるので説明を省略する。

【 0 0 9 6 】

しかしながら、 $a0 \times b00$ の演算結果及び $a1 \times b10$ の演算結果を得るために、乗算器13及び乗算器14の乗算結果は用いられない。つまり、演算部100では、乗算器13及び乗算器14は有効な演算を行っていないといえる。

【 0 0 9 7 】

これに対し、演算装置1A~1Dでは、MODE信号が第1演算モードを示す場合には、入力データDIN1の上位データdin1_h及び入力データDIN2の上位データdin2_hが乗算器12に供給され、乗算器12はこれらのデータの乗算を行う。また、入力データDIN1の下位データdin1_l及び入力データDIN2の上位データdin2_hが乗算器13に供給され、乗算器13はこれらのデータの乗算を行う。また、入力データDIN1の上位データdin1_h及び入力データDIN2の下位データdin2_lが乗算器14に供給され、乗算器14はこれらのデータの乗算を行う。また、入力データDIN1の下位データdin1_l及び入力データDIN2の下位データdin2_lが乗算器15に供給され、乗算器15はこれらのデータの乗算を行う。つまり、第1ビット数を有する1つのデータを、第2ビット数を有する2つのデータに分割して、乗算が行われる。

【 0 0 9 8 】

一方、演算装置1A~1Dでは、MODE信号が第2演算モードを示す場合には、入力データDIN1の上位データdin1_h及び入力データDIN2の上位データdin2_hが乗算器12に供給され、乗算器12はこれらのデータの乗算を行う。また、入力データDIN1の下位データdin1_l及び入力データDIN2の下位データdin2_lが乗算器13に供給され、乗算器13はこれらのデータの乗算を行う。また、入力デー

タDIN1の上位データdin1__h及び入力データDIN3の上位データdin3__hが乗算器14に供給され、乗算器14はこれらのデータの乗算を行う。また、入力データDIN1の下位データdin1__l及び入力データDIN3の下位データdin3__lが乗算器15に供給され、乗算器15はこれらのデータの乗算を行う。このように、第1演算モードだけでなく、第2演算モードにおいても、乗算器12~15の全ての乗算器が有効な演算を行う。その結果、演算装置1A~1Dの回路規模を拡大することなく、演算装置1A~1Dの演算性能を向上させることが可能となる。つまり、価格あたりの演算性能を向上させることが可能となる。

【0099】

また、比較例の演算装置では、第2演算モードにおいて、演算部100の乗算器12~15のうち、2つの乗算器が有効な演算を行っているのに対し、演算装置1A~1Dでは、演算部10A~10Dの乗算器12~15の全ての乗算器が有効な演算を行っている。このため、電力あたりの演算性能を向上させることが可能となる。言い換えれば、演算装置1A~1Dでは、比較例の演算装置に対し、演算装置1A~1Dを構成するトランジスタ等の回路素子の活用効率が向上するので、同じ演算性能を安価かつ低消費電力で提供することが可能となる。

【0100】

加算器52によって、入力データDIN1の上位データdin1__h及び下位データdin1__lと、入力データDIN2の上位データdin2__h及び下位データdin2__lと、のベクトル積が得られ、加算器53によって、入力データDIN1の上位データdin1__h及び下位データdin1__lと、入力データDIN3の上位データdin3__h及び下位データdin3__lと、のベクトル積が得られる。また、部分加算器16によって、入力データDIN1と入力データDIN2との乗算結果(第1ビット数を有する2つのビット列の乗算結果)が得られる。

【0101】

乗算器12は、乗算器12の乗算の途中結果である中間結果m11, m12を乗算結果として出力している。この中間結果m11, m12は、乗算器12の演算遅延を生じさせない乗算器12の乗算の途中結果である。このため、乗算器12の乗算結果として中間結果m11, m12が出力されることにより、乗算器12の演算処理において、例えば桁上がり等による遅延を抑制することができる。これにより、乗算器12の演算速度を向上させることが可能となる。同様に、乗算器13~15の演算速度を向上させることが可能となる。また、乗算の中間結果を用いることにより、後段の部分加算器16における加算のために、加算対象となるビット列の指数部を描える処理を省略し得るので、演算装置1A~1Dの演算速度を向上させることが可能となる。

【0102】

また、加算器52及び加算器53によって、第2ビット数を有するデータでの2つの行列演算結果が得られ、加算器51によって、第1ビット数を有するデータでの1つの行列演算結果が得られる。そして、マルチプレクサ59によって、MODE信号が第1演算モードを示す場合には、第1ビット数を有するデータでの1つの行列演算結果が出力され、MODE信号が第2演算モードを示す場合には、第2ビット数を有するデータでの2つの行列演算結果が出力される。このように、演算装置1A~1Dでは、2つの演算モードに応じた行列演算を行うことが可能となる。したがって、並列演算が必要となるDNN及びCNNに演算装置1A~1Dが適用された場合には、必要となる行列積に対して高い演算性能を実現することができる。また、演算装置1A~1Dに行列演算を行わせることにより、データマルチプレクサ11に供給するデータ数を減らすことができ、演算対象のデータを供給するためのデータバスのバンド幅を削減することが可能となる。

【0103】

第1演算モードが倍精度演算モードであり、第2演算モードが単精度演算モードである場合、演算部10A~10Dのそれぞれは、1個の倍精度の乗算器、及び4個の単精度の乗算器として再構成可能である。これにより、演算装置1A~1Dは、倍精度演算モード

及び単精度演算モードでの演算を行うことが可能となる。このような演算装置 1 A ~ 1 D においても、演算装置 1 A ~ 1 D の回路規模を拡大することなく、演算性能を向上させることが可能となる。

【 0 1 0 4 】

第 1 演算モードが単精度演算モードであり、第 2 演算モードが半精度演算モードである場合、演算部 1 0 A ~ 1 0 D のそれぞれは、1 個の単精度の乗算器、及び 4 個の半精度の乗算器として再構成可能である。これにより、演算装置 1 A ~ 1 D は、単精度演算モード及び半精度演算モードでの演算を行うことが可能となる。このような演算装置 1 A ~ 1 D においても、演算装置 1 A ~ 1 D の回路規模を拡大することなく、演算性能を向上させることが可能となる。

10

【 0 1 0 5 】

さらに、演算システム S では、演算ユニット MAU は、4 つの演算装置 1 A ~ 1 D を備えることにより、式 (1) に示されるような倍精度データの行列演算 (4 × 4 の行列ベクトル積) を行うことができる。演算ユニット MAU は、単精度演算モードでは式 (7) に示されるような 8 × 8 の行列ベクトル積を行うことが可能となる。この構成によれば、倍精度演算モードでは、それぞれが 5 2 ビットの仮数の乗算を行う 1 6 個の並列乗算器 (1 サイクルで 1 乗算を行う回路) を有する回路に相当する演算性能を実現することができる。単精度演算モードでは、それぞれが 2 3 ビットの仮数の乗算を行う 6 4 個の並列乗算器を有する回路に相当する演算性能を実現することができる。

【 0 1 0 6 】

また、要素プロセッサ PE 1 ~ PE 4 は、行列演算を行う場合には、単一のプロセッサとして動作するので、行列演算を要素プロセッサ PE 1 ~ PE 4 で並列処理することができる。これにより、行列演算の演算速度を向上させることが可能となる。

20

【 0 1 0 7 】

また、要素プロセッサ PE 1 ~ PE 4 は、1 つのインストラクションで動作する SIMD 動作を行うので、要素プロセッサ PE 1 ~ PE 4 が並列動作する。これにより、処理の高速化が可能となる。

【 0 1 0 8 】

演算システム S には、プロセッサ ID を含むメモリアドレスによってアクセス可能なメモリ空間が設定されている。要素プロセッサ PE 1 ~ PE 4 のそれぞれは、当該プロセッサ ID を含むメモリアドレスによって示されるメモリ領域にアクセス可能である。このため、メモリ空間を拡張することができ、演算システム S で扱えるデータサイズを大きくすることが可能となる。

30

【 0 1 0 9 】

要素プロセッサ PE 1 ~ PE 4 のそれぞれは、演算ユニット MAU から受信したデータをリング結合を介して他の要素プロセッサに順に転送 (循環シフト) することができる。これにより、要素プロセッサ PE 1 ~ PE 4 のそれぞれは、非担当領域へのアクセスを実行することが可能となる。

【 0 1 1 0 】

上記実施形態では、演算部 1 0 A ~ 1 0 D のそれぞれは、行列演算の専用回路として構成されている。つまり、乗算器 1 2 ~ 1 5 の乗算の中間結果を用いて加算を行っているので、乗算器 1 2 ~ 1 5 の乗算結果が得られない。図 7 に示されるように、変形例の演算部 1 0 A は、整列部 2 1 ~ 2 8 及び丸め処理部 3 1 ~ 3 8 に代えて、加算器 6 1 ~ 6 4 と、整列部 6 5 ~ 6 8 と、丸め処理部 6 9 ~ 7 2 と、を備える点で、上記実施形態の演算部 1 0 A と主に相違している。

40

【 0 1 1 1 】

加算器 6 1 ~ 6 4 は、比較例の演算部 1 0 0 の加算器 6 1 ~ 6 4 と比較して、乗算結果 m 1 ~ m 4 を部分加算器 1 6 にも出力する点において相違し、その余の処理については同じである。整列部 6 5 ~ 6 8 は、比較例の演算部 1 0 0 の整列部 6 5 ~ 6 8 と同じであるので、説明を省略する。丸め処理部 6 9 ~ 7 2 は、比較例の演算部 1 0 0 の丸め処理部 6

50

9 ~ 7 2 と比較して、演算結果の出力先において相違する。具体的には、丸め処理部 6 9 は、演算結果 r 1 を加算器 5 2 に出力する。丸め処理部 7 0 は、演算結果 r 2 を加算器 5 2 に出力する。丸め処理部 7 1 は、演算結果 r 3 を加算器 5 3 に出力する。丸め処理部 7 2 は、演算結果 r 4 を加算器 5 3 に出力する。

【 0 1 1 2 】

変形例の演算部 1 0 A によれば、加算器 6 1 ~ 6 4 によって、第 2 演算モードでの乗算結果 m 1 ~ m 4 が得られる。このため、演算部 1 0 A は、行列演算以外にも用いられ得る。

【 0 1 1 3 】

(第 2 実施形態)

図 8 は、第 2 実施形態に係る演算装置の構成を概略的に示す図である。図 8 に示される演算装置 1 A は、第 1 実施形態に係る演算装置 1 A と比較して、切り替え可能な演算モードの数、並びに、演算部 1 0 A ~ 1 0 D、及び加算部 5 0 に代えて、演算部 2 1 0 A ~ 2 1 0 D、及び加算部 2 5 0 を備える点において主に相違する。

【 0 1 1 4 】

第 2 実施形態に係る演算装置 1 A は、MODE 信号に応じて、演算装置 1 A の演算モードを第 1 演算モード、第 2 演算モード、及び第 3 演算モードのいずれかに切り替える。第 3 演算モードは、第 3 ビット数のデータを用いた演算を行うモードである。第 3 ビット数は、第 1 ビット数の 4 分の 1 であり、第 2 ビット数の半分のビット数である。MODE 信号は、例えば 2 ビットの信号であり、第 1 演算モード、第 2 演算モード、及び第 3 演算モードのいずれかを示す。演算装置 1 A が実行し得る演算モードとしては、例えば、倍精度演算モード、単精度演算モード、及び半精度演算モードが挙げられる。つまり、第 1 演算モードが倍精度演算モードである場合、第 2 演算モードは単精度演算モードであり、第 3 演算モードは半精度演算モードである。

【 0 1 1 5 】

図 9 は、図 8 の演算装置に含まれる演算部の構成を概略的に示す図である。演算部 2 1 0 A ~ 2 1 0 D は、同様の構成を有するので、ここでは、演算部 2 1 0 A の構成について、演算部 1 0 A との相違点を中心に説明する。図 9 に示されるように、演算部 2 1 0 A は、演算部 1 0 A と比較して、データマルチプレクサ 1 1 及び乗算器 1 2 ~ 1 5 に代えてデータマルチプレクサ 2 1 1 及び乗算器 2 1 2 ~ 2 1 5 を備える点、レジスタ 4 4 , 4 5 をさらに備える点において、主に相違する。

【 0 1 1 6 】

レジスタ 4 1 ~ 4 5 のそれぞれは、演算対象となる第 1 ビット数のビット列である入力データ D I N 1 ~ D I N 5 を記憶保持する。入力データ D I N 1 ~ D I N 5 は、演算装置 1 A の外部の装置 (本実施形態では、要素プロセッサ) からレジスタ 4 1 ~ 4 5 にセットされる。入力データ D I N 1 は、上述のように、上位データ d i n 1 _ h と、下位データ d i n 1 _ l と、を含む。上位データ d i n 1 _ h は、データ d i n ' 1 _ 1 及びデータ d i n ' 1 _ 2 を含み、下位データ d i n 1 _ l は、データ d i n ' 1 _ 3 及びデータ d i n ' 1 _ 4 を含む。データ d i n ' 1 _ 1 ~ d i n ' 1 _ 4 は、その順に入力データ D I N 1 のビット列の M S B から順に配列されている。入力データ D I N 2 ~ D I N 5 についても同様である。データ d i n ' 1 _ 1 ~ d i n ' 1 _ 4、データ d i n ' 2 _ 1 ~ d i n ' 2 _ 4、データ d i n ' 3 _ 1 ~ d i n ' 3 _ 4、データ d i n ' 4 _ 1 ~ d i n ' 4 _ 4、及びデータ d i n ' 5 _ 1 ~ d i n ' 5 _ 4 は、第 3 ビット数のビット列である。第 3 ビット数は、第 3 演算モードで用いられるデータのビット数に相当し、例えば 1 6 ビットである。

【 0 1 1 7 】

データマルチプレクサ 2 1 1 は、レジスタ 4 1 ~ 4 5 から出力されている入力データ D I N 1 ~ D I N 5 を入力し、MODE 信号に応じて、複数の出力データ d o u t 1 ~ d o u t 1 2 を出力する回路である。出力データ d o u t 1 ~ d o u t 1 2 のそれぞれは、第 2 ビット数のビット列である。

10

20

30

40

50

【0118】

MODE信号が第1演算モードを示す場合、及びMODE信号が第2演算モードを示す場合のデータマルチプレクサ211の動作は、データマルチプレクサ11の動作と同じである。データマルチプレクサ211は、MODE信号が第3演算モードを示す場合には、上位データdin1_hを出力データdout1及び出力データdout5として出力し、下位データdin1_lを出力データdout3及び出力データdout7として出力する。また、データマルチプレクサ211は、MODE信号が第3演算モードを示す場合には、上位データdin2_hを出力データdout2として出力し、下位データdin2_lを出力データdout4として出力する。

【0119】

また、データマルチプレクサ211は、MODE信号が第3演算モードを示す場合には、上位データdin3_hを出力データdout9として出力し、下位データdin3_lを出力データdout10として出力する。また、データマルチプレクサ211は、MODE信号が第3演算モードを示す場合には、上位データdin4_hを出力データdout6として出力し、下位データdin4_lを出力データdout8として出力する。また、データマルチプレクサ211は、MODE信号が第3演算モードを示す場合には、上位データdin5_hを出力データdout11として出力し、下位データdin5_lを出力データdout12として出力する。

【0120】

乗算器212~215は、第2ビット数を有する2つのビット列の乗算を行う回路である。乗算器212~215は、同様の構成を有するので、ここでは、乗算器212の構成について説明する。図10は、図9の乗算器の構成を概略的に示す図である。図10に示されるように、乗算器212は、データマルチプレクサ311と、乗算器312~315と、部分加算器316と、を備えている。

【0121】

データマルチプレクサ311は、出力データdout1, dout2, dout9を入力し、MODE信号に応じて、複数(ここでは8つ)の出力データdout'1~dout'8を出力する回路である。複数の出力データdout'1~dout'8のそれぞれは、第3ビット数のビット列である。出力データdout1は、データdout'1_h及びデータdout'1_lを含む。データdout'1_h及びデータdout'1_lは、その順に出力データdout1のビット列のMSBから順に配列されている。出力データdout2~dout12についても同様である。データdout'1_h~dout'12_h, dout'1_l~dout'12_lは、第3ビット数のビット列である。

【0122】

データマルチプレクサ311は、MODE信号が第1演算モード及び第2演算モードを示す場合には、データdout'1_hを出力データdout'1及び出力データdout'5として出力し、データdin'1_lを出力データdout'3及び出力データdout'7として出力する。また、データマルチプレクサ311は、MODE信号が第1演算モード及び第2演算モードを示す場合には、データdout'2_hを出力データdout'2及び出力データdout'4として出力し、データdout'2_lを出力データdout'6及び出力データdout'8として出力する。データマルチプレクサ311は、MODE信号が第1演算モード及び第2演算モードを示す場合には、出力データdout9を受け取らない。つまり、データマルチプレクサ311の第1演算モード及び第2演算モードでの動作は、データマルチプレクサ11の第1演算モードでの動作と同じである。

【0123】

データマルチプレクサ311は、MODE信号が第3演算モードを示す場合には、データdout'1_hを出力データdout'1及び出力データdout'5として出力し、データdin'1_lを出力データdout'3及び出力データdout'7として出

10

20

30

40

50

力する。また、データマルチプレクサ311は、MODE信号が第3演算モードを示す場合には、データdout'2_hを出力データdout'2として出力し、データdout'2_lを出力データdout'4として出力する。また、データマルチプレクサ311は、MODE信号が第3演算モードを示す場合には、データdout'9_hを出力データdout'6として出力し、データdout'9_lを出力データdout'8として出力する。つまり、データマルチプレクサ311の第3演算モードでの動作は、データマルチプレクサ11の第2演算モードでの動作と同じである。

【0124】

乗算器312~315は、第3ビット数を有する2つのビット列の乗算を行う回路である。乗算器312は、出力データdout'1と出力データdout'2との乗算を行う。乗算器312は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器312の乗算の途中結果である中間結果m'11及び中間結果m'12を乗算結果として出力する。中間結果m'11及び中間結果m'12は、第3ビット数のビット列である。中間結果m'11及び中間結果m'12は、乗算器312の演算遅延を生じさせない値であり、例えば、桁上りを生じさせない部分的な合計値である。乗算器312は、中間結果m'11を整列部321に出力し、中間結果m'12を整列部322に出力するとともに、中間結果m'11及び中間結果m'12を部分加算器316に出力する。

【0125】

乗算器313は、出力データdout'3と出力データdout'4との乗算を行う。乗算器313は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器313の乗算の途中結果である中間結果m'21及び中間結果m'22を乗算結果として出力する。中間結果m'21及び中間結果m'22は、第3ビット数のビット列である。中間結果m'21及び中間結果m'22は、乗算器313の演算遅延を生じさせない値であり、例えば、桁上りを生じさせない部分的な合計値である。乗算器313は、中間結果m'21を整列部323に出力し、中間結果m'22を整列部324に出力するとともに、中間結果m'21及び中間結果m'22を部分加算器316に出力する。

【0126】

乗算器314は、出力データdout'5と出力データdout'6との乗算を行う。乗算器314は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器314の乗算の途中結果である中間結果m'31及び中間結果m'32を乗算結果として出力する。中間結果m'31及び中間結果m'32は、第3ビット数のビット列である。中間結果m'31及び中間結果m'32は、乗算器314の演算遅延を生じさせない値であり、例えば、桁上りを生じさせない部分的な合計値である。乗算器314は、中間結果m'31を整列部325に出力し、中間結果m'32を整列部326に出力するとともに、中間結果m'31及び中間結果m'32を部分加算器316に出力する。

【0127】

乗算器315は、出力データdout'7と出力データdout'8との乗算を行う。乗算器315は、例えば、ウォレスツリー回路を用いて2つのビット列の乗算を行い、乗算器315の乗算の途中結果である中間結果m'41及び中間結果m'42を乗算結果として出力する。中間結果m'41及び中間結果m'42は、第3ビット数のビット列である。中間結果m'41及び中間結果m'42は、乗算器315の演算遅延を生じさせない値であり、例えば、桁上りを生じさせない部分的な合計値である。乗算器315は、中間結果m'41を整列部327に出力し、中間結果m'42を整列部328に出力するとともに、中間結果m'41及び中間結果m'42を部分加算器316に出力する。

【0128】

部分加算器316は、第2ビット数のビット列での乗算結果を生成する回路である。具体的には、部分加算器316は、乗算器312~315の乗算結果の加算を行う。本実施形態では、部分加算器316は、中間結果m'11、中間結果m'12、中間結果m'21、中間結果m'22、中間結果m'31、中間結果m'32、中間結果m'41、及び中間結果m'42の加算を行う。部分加算器316は、部分加算器316の加算の途中結

果である中間結果 m'_{11} 及び中間結果 m'_{12} を加算結果として出力する。中間結果 m'_{11} 及び中間結果 m'_{12} は、第2ビット数のビット列である。中間結果 m'_{11} 及び中間結果 m'_{12} は、部分加算器 316 の演算遅延を生じさせない値であり、例えば、桁上がりを生じさせない部分的な合計値である。

【0129】

整列部 321 は、後述の加算器 54 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{11} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 321 は、整列部 321 の処理結果を丸め処理部 331 に出力する。整列部 322 は、加算器 54 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{12} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 322 は、整列部 322 の処理結果を丸め処理部 332 に出力する。整列部 323 は、加算器 54 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{21} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 323 は、整列部 323 の処理結果を丸め処理部 333 に出力する。整列部 324 は、加算器 54 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{22} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 324 は、整列部 324 の処理結果を丸め処理部 334 に出力する。

【0130】

整列部 325 は、後述の加算器 55 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{31} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 325 は、整列部 325 の処理結果を丸め処理部 335 に出力する。整列部 326 は、加算器 55 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{32} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 326 は、整列部 326 の処理結果を丸め処理部 336 に出力する。整列部 327 は、加算器 55 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{41} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 327 は、整列部 327 の処理結果を丸め処理部 337 に出力する。整列部 328 は、加算器 55 によって加算されるビット列の指数部を揃えるために、中間結果 m'_{42} の仮数部に対してシフト処理（右ビットシフト）を行う回路である。整列部 328 は、整列部 328 の処理結果を丸め処理部 338 に出力する。

【0131】

丸め処理部 331 は、整列部 321 によってシフト処理が行われた中間結果 m'_{11} に対して丸め処理を行う回路である。丸め処理部 331 は、丸め処理部 331 の演算結果 r'_{11} を加算部 50（加算器 54）に出力する。丸め処理部 332 は、整列部 322 によってシフト処理が行われた中間結果 m'_{12} に対して丸め処理を行う回路である。丸め処理部 332 は、丸め処理部 332 の演算結果 r'_{12} を加算部 50（加算器 54）に出力する。丸め処理部 333 は、整列部 323 によってシフト処理が行われた中間結果 m'_{21} に対して丸め処理を行う回路である。丸め処理部 333 は、丸め処理部 333 の演算結果 r'_{21} を加算部 50（加算器 54）に出力する。丸め処理部 334 は、整列部 324 によってシフト処理が行われた中間結果 m'_{22} に対して丸め処理を行う回路である。丸め処理部 334 は、丸め処理部 334 の演算結果 r'_{22} を加算部 50（加算器 54）に出力する。演算結果 r'_{11} 、 r'_{12} 、 r'_{21} 、 r'_{22} のそれぞれは、第3ビット数のビット列である。

【0132】

丸め処理部 335 は、整列部 325 によってシフト処理が行われた中間結果 m'_{31} に対して丸め処理を行う回路である。丸め処理部 335 は、丸め処理部 335 の演算結果 r'_{31} を加算部 50（加算器 55）に出力する。丸め処理部 336 は、整列部 326 によってシフト処理が行われた中間結果 m'_{32} に対して丸め処理を行う回路である。丸め処理部 336 は、丸め処理部 336 の演算結果 r'_{32} を加算部 50（加算器 55）に出力する。丸め処理部 337 は、整列部 327 によってシフト処理が行われた中間結果 m'_{41}

1 に対して丸め処理を行う回路である。丸め処理部 337 は、丸め処理部 337 の演算結果 $r'41$ を加算部 50 (加算器 55) に出力する。丸め処理部 338 は、整列部 328 によってシフト処理が行われた中間結果 $m'42$ に対して丸め処理を行う回路である。丸め処理部 338 は、丸め処理部 338 の演算結果 $r'42$ を加算部 50 (加算器 55) に出力する。演算結果 $r'31$, $r'32$, $r'41$, $r'42$ のそれぞれは、第 3 ビット数のビット列である。

【0133】

このように、乗算器 212 ~ 215 は、演算部 10A のデータマルチプレクサ 11、乗算器 12 ~ 15、部分加算器 16、整列部 21 ~ 28、及び丸め処理部 31 ~ 38 と同様の構成を有している。なお、データマルチプレクサ 211 と各乗算器 212 ~ 215 のデータマルチプレクサ 311 とは、1 つのデータマルチプレクサで置き換えられてもよい。

【0134】

図 8 に戻って、演算装置 1A の説明を続ける。加算部 250 は、加算部 50 と比較して、マルチプレクサ 59 に代えてマルチプレクサ 259 を備える点、加算器 54 ~ 57 をさらに備える点において主に相違する。以下、加算部 50 との相違点について説明する。

【0135】

レジスタ 40 に保持される入力データ $DIN0$ は、第 3 ビット数のビット列であるデータ $din0_1 \sim din0_4$ を含む。レジスタ 40 は、データ $din0_1$ を加算器 54 に出力し、データ $din0_2$ を加算器 55 に出力し、データ $din0_3$ を加算器 56 に出力し、データ $din0_4$ を加算器 57 に出力する。

【0136】

加算器 54 は、第 3 演算モードの演算結果 $d'0$ を生成する。本実施形態では、加算器 54 は、演算部 210A, 210B の乗算器 212 ~ 215 の演算結果 $r'11$, $r'12$, $r'21$, $r'22$ と、データ $din'0_1$ との加算を行うことによって、演算結果 $d'0$ を生成する。加算器 54 は、演算結果 $d'0$ をマルチプレクサ 259 に出力する。

【0137】

加算器 55 は、第 3 演算モードの演算結果 $d'1$ を生成する。本実施形態では、加算器 55 は、演算部 210A, 210B の乗算器 212 ~ 215 の演算結果 $r'31$, $r'32$, $r'41$, $r'42$ と、データ $din0_2$ との加算を行うことによって、演算結果 $d'1$ を生成する。加算器 55 は、演算結果 $d'1$ をマルチプレクサ 259 に出力する。

【0138】

加算器 56 は、第 3 演算モードの演算結果 $d'2$ を生成する。本実施形態では、加算器 56 は、演算部 210C, 210D の乗算器 212 ~ 215 の演算結果 $r'11$, $r'12$, $r'21$, $r'22$ と、データ $din0_3$ との加算を行うことによって、演算結果 $d'2$ を生成する。加算器 56 は、演算結果 $d'2$ をマルチプレクサ 259 に出力する。

【0139】

加算器 57 は、第 3 演算モードの演算結果 $d'3$ を生成する。本実施形態では、加算器 57 は、演算部 210C, 210D の乗算器 212 ~ 215 の演算結果 $r'31$, $r'32$, $r'41$, $r'42$ と、データ $din0_4$ との加算を行うことによって、演算結果 $d'3$ を生成する。加算器 57 は、演算結果 $d'3$ をマルチプレクサ 259 に出力する。なお、演算結果 $d'0 \sim d'3$ が組み合わせられて演算結果 ($d'0$, $d'1$, $d'2$, $d'3$) が生成される。演算結果 ($d'0$, $d'1$, $d'2$, $d'3$) は、演算結果 $d'0 \sim d'3$ をその順に上位ビットから配列したビット列である。つまり、演算結果 ($d'0$, $d'1$, $d'2$, $d'3$) は、第 1 ビット数のビット列である。

【0140】

マルチプレクサ 259 は、演算結果 $D0$ 、演算結果 ($d0$, $d1$)、及び演算結果 ($d'0$, $d'1$, $d'2$, $d'3$) を入力し、MODE 信号に応じて、演算結果 $D0$ と演算結果 ($d0$, $d1$) と演算結果 ($d'0$, $d'1$, $d'2$, $d'3$) とのいずれかを出力する回路である。マルチプレクサ 259 は、MODE 信号が第 1 演算モードを示す場合に

は、演算結果 D0 を出力する。マルチプレクサ 259 は、MODE 信号が第 2 演算モードを示す場合には、演算結果 (d0, d1) を出力する。マルチプレクサ 259 は、MODE 信号が第 3 演算モードを示す場合には、演算結果 (d'0, d'1, d'2, d'3) を出力する。

【0141】

演算装置 1A は、シフト量演算回路 260 をさらに備える。図 11 は、シフト量演算回路の構成を概略的に示す図である。図 11 に示されるシフト量演算回路 260 は、各整列部におけるシフト処理のシフト量を演算する回路である。シフト量演算回路 260 は、指数部抽出回路 261 と、最大値演算回路 262A ~ 262D と、減算回路 263A ~ 263D と、を備える。

10

【0142】

指数部抽出回路 261 は、入力データ DIN0 ~ DIN5 から指数部を抽出する回路である。指数部抽出回路 261 には、レジスタ 40 から入力データ DIN0 が供給され、演算部 210A ~ 210D から入力データ DIN1 ~ DIN5 が供給されている。

【0143】

図 12 (a) ~ (c) に示されるように、各入力データ DIN0 ~ DIN5 が倍精度データ、単精度データ、及び半精度データのいずれを有するかに応じて、指数部の数及び位置が異なる。倍精度データのビット数が 64 ビットである場合、倍精度データでは、MSB は符号を示し、続く 11 ビットが指数部、残りの 52 ビットが仮数部である。単精度データのビット数が 32 ビットである場合、単精度データでは、MSB は符号を示し、続く 8 ビットが指数部、残りの 23 ビットが仮数部である。半精度データのビット数が 16 ビットである場合、半精度データでは、MSB は符号を示し、続く 5 ビットが指数部、残りの 10 ビットが仮数部である。

20

【0144】

入力データ DIN0 ~ DIN5 が倍精度データである場合、入力データ DIN0 ~ DIN5 のそれぞれは 1 つの指数部を有する。入力データ DIN0 ~ DIN5 が単精度データである場合、入力データ DIN0 ~ DIN5 のそれぞれは 2 つの指数部を有する。入力データ DIN0 ~ DIN5 が半精度データである場合、入力データ DIN0 ~ DIN5 のそれぞれは 4 つの指数部を有する。指数部抽出回路 261 は、MODE 信号に応じて指数部を抽出し、抽出した指数部に基づいて、加算対象となる指数部を最大値演算回路 262A ~ 262D、及び減算回路 263A ~ 263D に出力する。

30

【0145】

最大値演算回路 262A ~ 262D は、加算対象となる複数の対象データの指数部のうちの最大の指数部である最大指数を演算する回路である。最大値演算回路 262A ~ 262D は、指数部抽出回路 261 から供給された複数の指数部のうちの最大の指数部を最大指数とする。最大値演算回路 262A ~ 262D は、例えば、複数の対象データの MSB から LSB に向かって順番に比較を行うことによって、最大値を有するビット列を特定し、特定したビット列の値を最大指数とする。最大値演算回路 262A ~ 262D は、最大指数を減算回路 263A ~ 263D に出力する。減算回路 263A ~ 263D は、最大値演算回路 262A ~ 262D によって演算された最大値から、指数部抽出回路 261 から供給された指数部を減算し、減算結果をシフト量として整列部に出力する。シフト量演算回路 260 の詳細な動作については後述する。

40

【0146】

ここで、図 13 ~ 図 15 を用いて最大値演算回路 262A ~ 262D の構成例を説明する。図 13 は、2 つのビット列から最大値を演算する最大値演算回路の一構成例を示す回路図である。図 14 は、図 13 の最大値演算回路の別の構成例を示す回路図である。図 15 は、複数のビット列から最大値を演算する最大値演算回路の構成例を示す図である。最大値演算回路 262A ~ 262D は、同様の構成を有するので、ここでは、最大値演算回路 262A の構成について説明する。

【0147】

50

図13に示される最大値演算回路262Aは、2つの数値のうちの最大値を算出する回路である。この例では、2つの数値を示すビット列E0, E1は、4ビットのビット列である。最大値演算回路262Aは、ビット列E0, E1のうちの最大値を有するビット列E_{max}を出力する。ビット列E0は、ビットE0(0), E0(1), E0(2), E0(3)をその順に配列したビット列である。ビットE0(0)がMSBであり、E0(3)がLSBである。同様に、ビット列E1は、ビットE1(0), E1(1), E1(2), E1(3)をその順に配列したビット列である。ビットE1(0)がMSBであり、E1(3)がLSBである。同様に、ビット列E_{max}は、ビットE_{max}(0), E_{max}(1), E_{max}(2), E_{max}(3)をその順に配列したビット列である。ビットE_{max}(0)がMSBであり、E_{max}(3)がLSBである。

10

【0148】

最大値演算回路262Aは、ビット生成部361a~361dと、マスク信号生成部362a~362dと、を備える。ビット生成部361aは、ビットE_{max}(0)を生成する。具体的には、ビット生成部361aは、ORゲートによって構成される。ビット生成部361aは、ビットE0(0), E1(0)のOR演算を行って、演算結果をビットE_{max}(0)として出力する。

【0149】

ビット生成部361bは、ビットE_{max}(1)を生成する。具体的には、ビット生成部361bは、2つのANDゲート及びORゲートによって構成される。ビット生成部361bは、ビットE0(1)とマスク信号mask0(0)とのAND演算、及びビットE1(1)とマスク信号mask1(0)とのAND演算をそれぞれ行う。そして、ビット生成部361bは、2つのAND演算の演算結果をOR演算し、演算結果をビットE_{max}(1)として出力する。

20

【0150】

ビット生成部361cは、ビットE_{max}(2)を生成する。具体的には、ビット生成部361cは、2つのANDゲート及びORゲートによって構成される。ビット生成部361cは、ビットE0(2)とマスク信号mask0(1)とのAND演算、及びビットE1(2)とマスク信号mask1(1)とのAND演算をそれぞれ行う。そして、ビット生成部361cは、2つのAND演算の演算結果をOR演算し、演算結果をビットE_{max}(2)として出力する。

30

【0151】

ビット生成部361dは、ビットE_{max}(3)を生成する。具体的には、ビット生成部361dは、2つのANDゲート及びORゲートによって構成される。ビット生成部361dは、ビットE0(3)とマスク信号mask0(2)とのAND演算、及びビットE1(3)とマスク信号mask1(2)とのAND演算をそれぞれ行う。そして、ビット生成部361dは、2つのAND演算の演算結果をOR演算し、演算結果をビットE_{max}(3)として出力する。

【0152】

マスク信号生成部362aは、マスク信号mask0(0), mask1(0)を生成する。具体的には、マスク信号生成部362aは、2つのORゲートによって構成される。マスク信号生成部362aは、ビットE0(0)とビットE_{max}(0)の反転とをOR演算し、演算結果をマスク信号mask0(0)として出力する。マスク信号生成部362aは、ビットE1(0)とビットE_{max}(0)の反転とをOR演算し、演算結果をマスク信号mask1(0)として出力する。

40

【0153】

マスク信号生成部362bは、マスク信号mask0(1), mask1(1)を生成する。具体的には、マスク信号生成部362bは、2つのORゲート及び2つのANDゲートによって構成される。マスク信号生成部362bは、ビットE0(1)とビットE_{max}(1)の反転とをOR演算し、さらにその演算結果とマスク信号mask0(0)とをAND演算し、その演算結果をマスク信号mask0(1)として出力する。同様に、

50

マスク信号生成部 362b は、ビット $E_1(1)$ とビット $E_{max}(1)$ の反転とを OR 演算し、さらにその演算結果とマスク信号 $mask_1(0)$ とを AND 演算し、その演算結果をマスク信号 $mask_1(1)$ として出力する。

【0154】

マスク信号生成部 362c は、マスク信号 $mask_0(2)$, $mask_1(2)$ を生成する。具体的には、マスク信号生成部 362c は、2つの OR ゲート及び2つの AND ゲートによって構成される。マスク信号生成部 362c は、ビット $E_0(2)$ とビット $E_{max}(2)$ の反転とを OR 演算し、さらにその演算結果とマスク信号 $mask_0(1)$ とを AND 演算し、その演算結果をマスク信号 $mask_0(2)$ として出力する。同様に、マスク信号生成部 362c は、ビット $E_1(2)$ とビット $E_{max}(2)$ の反転とを OR 演算し、さらにその演算結果とマスク信号 $mask_1(1)$ とを AND 演算し、その演算結果をマスク信号 $mask_1(2)$ として出力する。

【0155】

マスク信号生成部 362d は、マスク信号 $mask_0(3)$, $mask_1(3)$ を生成する。具体的には、マスク信号生成部 362d は、2つの OR ゲート及び2つの AND ゲートによって構成される。マスク信号生成部 362d は、ビット $E_0(3)$ とビット $E_{max}(3)$ の反転とを OR 演算し、さらにその演算結果とマスク信号 $mask_0(2)$ とを AND 演算し、その演算結果をマスク信号 $mask_0(3)$ として出力する。同様に、マスク信号生成部 362d は、ビット $E_1(3)$ とビット $E_{max}(3)$ の反転とを OR 演算し、さらにその演算結果とマスク信号 $mask_1(2)$ とを AND 演算し、その演算結果をマスク信号 $mask_1(3)$ として出力する。

【0156】

このように、図13の最大値演算回路 262A では、ビット列 E_{max} が MSB から順に1ビットずつ算出される。マスク信号 $mask_0$, $mask_1$ が MSB から LSB に向かって順に伝搬しているため、最大値演算回路 262A の遅延はこれらのマスク信号 $mask_0$, $mask_1$ が通過するゲート数によって律速される。なお、比較対象のビット列 E_0 , E_1 は、4ビットのビット列であるため、マスク信号生成部 362d は省略され得る。また、比較対象のビット列が4ビットよりも大きい場合には、ビット数に応じてビット生成部 361d 及びマスク信号生成部 362d がさらに追加される。

【0157】

図14に示される最大値演算回路 262A は、2つの数値のうちの最大値を算出する回路である。この例では、2つの数値を示すビット列 E_0 , E_1 は、4ビットのビット列である。最大値演算回路 262A は、図13に示される最大値演算回路 262A と比較して、マスク信号生成部 362a , 362c を備えない点、並びに、ビット生成部 361b , 361d 及びマスク信号生成部 362b , 362d の構成において主に相違する。

【0158】

ビット生成部 361b は、3つの AND ゲート及び2つの OR ゲートによって構成される。ビット生成部 361b は、ビット $E_0(0)$, $E_0(1)$ を AND 演算し、ビット $E_1(0)$, $E_1(1)$ を AND 演算する。ビット生成部 361b は、ビット $E_0(1)$, $E_1(1)$ を OR 演算し、その演算結果とビット $E_0(0)$ の反転とビット $E_1(0)$ の反転とを AND 演算する。そして、ビット生成部 361b は、3つの AND 演算の演算結果をさらに OR 演算し、演算結果をビット $E_{max}(1)$ として出力する。

【0159】

ビット生成部 361d は、6つの AND ゲート及び3つの OR ゲートによって構成される。ビット生成部 361d は、ビット $E_0(2)$, $E_0(3)$ を AND 演算し、ビット $E_1(2)$, $E_1(3)$ を AND 演算する。ビット生成部 361d は、ビット $E_0(3)$, $E_1(3)$ を OR 演算し、その演算結果とビット $E_0(2)$ の反転とビット $E_1(2)$ の反転とを AND 演算する。そして、ビット生成部 361d は、3つの AND 演算の演算結果をさらに OR 演算する。ビット生成部 361d は、OR 演算の演算結果とマスク信号 $mask_0(1)$, $mask_1(1)$ との AND 演算、ビット $E_0(3)$ の反転とマスク信

号 $mask0(1)$ との AND 演算、ビット $E1(3)$ の反転とマスク信号 $mask1(1)$ との AND 演算をさらに行い、3つの AND 演算の演算結果を OR 演算して演算結果をビット $E_{max}(3)$ として出力する。

【0160】

マスク信号生成部 362b は、8つの AND ゲート及び2つの OR ゲートによって構成される。マスク信号生成部 362b は、ビット $E1(0)$ 、 $E1(1)$ の AND 演算結果の反転とビット $E0(0)$ とを AND 演算し、その演算結果とビット $E0(0)$ 、 $E0(1)$ の AND 演算結果とをさらに AND 演算する。そして、マスク信号生成部 362b は、ビット $E1(0)$ の反転とビット $E0(1)$ とを AND 演算し、ビット $E1(0)$ の反転とビット $E1(1)$ の反転とを AND 演算する。そして、マスク信号生成部 362b は、3つの AND 演算結果を OR 演算し、その演算結果をマスク信号 $mask0(1)$ として出力する。

【0161】

また、マスク信号生成部 362b は、ビット $E0(0)$ 、 $E0(1)$ の AND 演算結果の反転とビット $E1(1)$ とを AND 演算し、その演算結果とビット $E1(0)$ 、 $E1(1)$ の AND 演算結果とをさらに AND 演算する。そして、マスク信号生成部 362b は、ビット $E0(0)$ の反転とビット $E1(1)$ とを AND 演算し、ビット $E0(0)$ の反転とビット $E0(1)$ の反転とを AND 演算する。そして、マスク信号生成部 362b は、3つの AND 演算結果を OR 演算し、その演算結果をマスク信号 $mask1(1)$ として出力する。

【0162】

マスク信号生成部 362d は、12個の AND ゲート及び3つの OR ゲートによって構成される。マスク信号生成部 362d は、マスク信号 $mask0(1)$ 、 $mask1(1)$ の OR 演算を行い、その演算結果の反転とマスク信号 $mask0(1)$ との AND 演算を行う。また、マスク信号生成部 362d は、ビット $E1(2)$ 、 $E1(3)$ の AND 演算結果の反転とビット $E0(2)$ とを AND 演算し、その演算結果とビット $E0(2)$ 、 $E0(3)$ の AND 演算結果とをさらに AND 演算する。そして、マスク信号生成部 362d は、ビット $E1(2)$ の反転とビット $E0(3)$ とを AND 演算し、ビット $E1(2)$ の反転とビット $E1(3)$ の反転とを AND 演算する。そして、マスク信号生成部 362d は、4つの AND 演算結果を OR 演算し、その演算結果とマスク信号 $mask0(1)$ 、 $mask1(1)$ の OR 演算結果との AND 演算を行い、その演算結果をマスク信号 $mask0(3)$ として出力する。

【0163】

マスク信号生成部 362d は、マスク信号 $mask0(1)$ 、 $mask1(1)$ の OR 演算結果の反転とマスク信号 $mask1(1)$ との AND 演算を行う。また、マスク信号生成部 362d は、ビット $E0(2)$ 、 $E0(3)$ の AND 演算結果の反転とビット $E1(3)$ とを AND 演算し、その演算結果とビット $E1(2)$ 、 $E1(3)$ の AND 演算結果とをさらに AND 演算する。そして、マスク信号生成部 362d は、ビット $E0(2)$ の反転とビット $E1(3)$ とを AND 演算し、ビット $E0(2)$ の反転とビット $E0(3)$ の反転とを AND 演算する。そして、マスク信号生成部 362d は、4つの AND 演算結果を OR 演算し、その演算結果とマスク信号 $mask0(1)$ 、 $mask1(1)$ の OR 演算結果との AND 演算を行い、その演算結果をマスク信号 $mask1(3)$ として出力する。

【0164】

このように、図14の最大値演算回路 262A では、ビット列 E_{max} が MSB から順に2ビットずつ算出される。複数ビットをまとめて処理することによって、マスク信号 $mask0$ 、 $mask1$ が通過するゲート数を図13の最大値演算回路 262A よりも減らすことができる。これにより、図13の最大値演算回路 262A と比較して、最大値演算回路 262A の遅延を低減することができ、最大値演算回路 262A の処理効率を向上させることが可能となる。

10

20

30

40

50

【 0 1 6 5 】

なお、比較対象のビット列 E 0 , E 1 は、4 ビットのビット列であるので、マスク信号生成部 3 6 2 d は省略され得る。また、比較対象のビット列 E 0 , E 1 が 4 ビットよりも大きい場合には、2 ビットごとに、ビット生成部 3 6 1 c , 3 6 1 d 及びマスク信号生成部 3 6 2 d がさらに追加される。

【 0 1 6 6 】

図 1 5 に示される最大値演算回路 2 6 2 A は、複数の数値のうちの最大値を算出する回路である。この例では、最大値演算回路 2 6 2 A は、8 つのビット列 E 0 ~ E 7 のうちの最大値を有するビット列を出力する。最大値演算回路 2 6 2 A は、複数の最大値演算部（ここでは、最大値演算部 2 6 2 a ~ 2 6 2 g ）を備える。最大値演算部 2 6 2 a ~ 2 6 2 g のそれぞれは、2 つの数値のうちの最大値を算出する回路であり、例えば、図 1 3 又は図 1 4 に示される最大値演算回路 2 6 2 A と同様の構成を有している。

【 0 1 6 7 】

最大値演算部 2 6 2 a は、ビット列 E 0 , E 1 のうちの最大値を有するビット列を出力する。最大値演算部 2 6 2 b は、ビット列 E 2 , E 3 のうちの最大値を有するビット列を出力する。最大値演算部 2 6 2 c は、ビット列 E 4 , E 5 のうちの最大値を有するビット列を出力する。最大値演算部 2 6 2 d は、ビット列 E 6 , E 7 のうちの最大値を有するビット列を出力する。最大値演算部 2 6 2 e は、最大値演算部 2 6 2 a から出力されたビット列と最大値演算部 2 6 2 b から出力されたビット列とのうちの最大値を有するビット列を出力する。最大値演算部 2 6 2 f は、最大値演算部 2 6 2 c から出力されたビット列と最大値演算部 2 6 2 d から出力されたビット列とのうちの最大値を有するビット列を出力する。最大値演算部 2 6 2 g は、最大値演算部 2 6 2 e から出力されたビット列と最大値演算部 2 6 2 f から出力されたビット列とのうちの最大値を有するビット列を出力する。

【 0 1 6 8 】

このように、最大値演算回路 2 6 2 A では、複数の最大値演算部がツリー状に接続されている。これにより、複数のビット列のうちの最大値を有するビット列が出力される。

【 0 1 6 9 】

次に、図 1 6 を用いて演算装置 1 A ~ 1 D の第 1 演算モードでの動作を説明する。図 1 6 は、図 8 の演算装置の第 1 演算モードでの動作を説明するための図である。ここでは、第 1 演算モードとして、倍精度演算モードを用い、演算ユニット M A U が式 (1) に示される倍精度の行列演算を行う場合を一例として説明する。演算装置 1 A ~ 1 D では、演算対象のデータが異なるものの、動作は同じであるので、ここでは演算装置 1 A について説明する。また、演算部 2 1 0 A ~ 2 1 0 D では、演算対象のデータが異なるものの、動作は同じであるので、ここでは演算部 2 1 0 A について説明する。

【 0 1 7 0 】

演算装置 1 A の演算部 2 1 0 A は、 $A 0 \times B 0 0$ の演算を行う。倍精度データ A 0 は、上位データ $a 0 _h$ と、下位データ $a 0 _l$ と、を含む。上位データ $a 0 _h$ は、データ $a ' 0 _1$ 及びデータ $a ' 0 _2$ を含み、下位データ $a 0 _l$ は、データ $a ' 0 _3$ 及びデータ $a ' 0 _4$ を含む。データ $a ' 0 _1 \sim a ' 0 _4$ は、その順に倍精度データ A 0 のビット列の M S B から順に配列されている。倍精度データ B 0 0 は、上位データ $b 0 0 _h$ と、下位データ $b 0 0 _l$ と、を含む。上位データ $b 0 0 _h$ は、データ $b ' 0 0 _1$ 及びデータ $b ' 0 0 _2$ を含み、下位データ $b 0 0 _l$ は、データ $b ' 0 0 _3$ 及びデータ $b ' 0 0 _4$ を含む。データ $b ' 0 0 _1 \sim b ' 0 0 _4$ は、その順に倍精度データ B 0 0 のビット列の M S B から順に配列されている。データ $a ' 0 _1 \sim a ' 0 _4$ 、及びデータ $b ' 0 0 _1 \sim b ' 0 0 _4$ は、半精度データと同じビット数を有する。

【 0 1 7 1 】

演算部 2 1 0 A は、式 (1 6) に示されるように、倍精度データ A 0 及び倍精度データ B 0 0 のそれぞれを半精度データのビット数を有する 4 つのデータに分解することによって、倍精度データ A 0 及び倍精度データ B 0 0 の乗算を行う。

10

20

30

40

50

【数 1 6】

$$\begin{aligned}
 A0 \times B00 &= (a0_h + a0_l) \times (b00_h + b00_l) \\
 &= a0_h \times b00_h + a0_l \times b00_h + a0_h \times b00_l + a0_l \times b00_l \\
 &= (a'0_1 + a'0_2) \times (b'00_1 + b'00_2) + (a'0_3 + a'0_4) \times (b'00_1 + b'00_2) \\
 &\quad + (a'0_1 + a'0_2) \times (b'00_3 + b'00_4) + (a'0_3 + a'0_4) \times (b'00_3 + b'00_4) \\
 &= a'0_1 \times b'00_1 + a'0_1 \times b'00_2 + a'0_1 \times b'00_3 + a'0_1 \times b'00_4 + \\
 &\quad a'0_2 \times b'00_1 + a'0_2 \times b'00_2 + a'0_2 \times b'00_3 + a'0_2 \times b'00_4 + \\
 &\quad a'0_3 \times b'00_1 + a'0_3 \times b'00_2 + a'0_3 \times b'00_3 + a'0_3 \times b'00_4 + \\
 &\quad a'0_4 \times b'00_1 + a'0_4 \times b'00_2 + a'0_4 \times b'00_3 + a'0_4 \times b'00_4 \quad \dots \quad (16)
 \end{aligned}$$

【0 1 7 2】

以下、具体的に説明する。レジスタ 4 1 には、倍精度データ A 0 が記憶されており、レジスタ 4 1 からデータマルチプレクサ 2 1 1 に入力データ D I N 1 として倍精度データ A 0 が供給されている。レジスタ 4 2 には、倍精度データ B 0 0 が記憶されており、レジスタ 4 2 からデータマルチプレクサ 2 1 1 に入力データ D I N 2 として倍精度データ B 0 0 が供給されている。レジスタ 4 3 ~ 4 5 には、有効なデータは記憶されていない。

【0 1 7 3】

データマルチプレクサ 2 1 1 には、倍精度演算モードを示す M O D E 信号が供給されている。このため、データマルチプレクサ 2 1 1 は、上位データ a 0 __ h 及び上位データ b 0 0 __ h を乗算器 2 1 2 に出力する。また、データマルチプレクサ 2 1 1 は、下位データ a 0 __ l 及び上位データ b 0 0 __ h を乗算器 2 1 3 に出力する。また、データマルチプレクサ 2 1 1 は、上位データ a 0 __ h 及び下位データ b 0 0 __ l を乗算器 2 1 4 に出力する。また、データマルチプレクサ 2 1 1 は、下位データ a 0 __ l 及び下位データ b 0 0 __ l を乗算器 2 1 5 に出力する。

【0 1 7 4】

乗算器 2 1 2 では、データマルチプレクサ 3 1 1 には、倍精度演算モードを示す M O D E 信号が供給されている。このため、データマルチプレクサ 3 1 1 は、データ a ' 0 __ 1 及びデータ b ' 0 0 __ 1 を乗算器 3 1 2 に出力し、データ a ' 0 __ 2 及びデータ b ' 0 0 __ 1 を乗算器 3 1 3 に出力し、データ a ' 0 __ 1 及びデータ b ' 0 0 __ 2 を乗算器 3 1 4 に出力し、データ a ' 0 __ 2 及びデータ b ' 0 0 __ 2 を乗算器 3 1 5 に出力する。そして、各乗算器 3 1 2 ~ 3 1 5 において乗算が行われ、その中間結果 m ' 1 1 , m ' 1 2 , m ' 2 1 , m ' 2 2 , m ' 3 1 , m ' 3 2 , m ' 4 1 , m ' 4 2 を部分加算器 3 1 6 が加算し、中間結果 m 1 1 及び中間結果 m 1 2 を出力する。つまり、乗算器 2 1 2 では、a 0 __ h × b 0 0 __ h の演算が半精度データを用いて行われ、中間結果 m 1 1 及び中間結果 m 1 2 が出力される。

【0 1 7 5】

同様に、乗算器 3 1 3 では、a 0 __ l × b 0 0 __ h の演算が半精度データを用いて行われ、中間結果 m 2 1 及び中間結果 m 2 2 が出力される。同様に、乗算器 3 1 4 では、a 0 __ h × b 0 0 __ l の演算が半精度データを用いて行われ、中間結果 m 3 1 及び中間結果 m 3 2 が出力される。同様に、乗算器 3 1 5 では、a 0 __ l × b 0 0 __ l の演算が半精度データを用いて行われ、中間結果 m 4 1 及び中間結果 m 4 2 が出力される。乗算器 2 1 3 ~ 2 1 5 の動作も乗算器 2 1 2 の動作と同様であるので、説明を省略する。以降の処理は、演算部 1 0 A と同じであるので、説明を省略する。

【0 1 7 6】

ここで、シフト量演算回路 2 6 0 の第 1 演算モードでの動作を説明する。演算装置 1 A では、式 (2) に示される演算が行われるので、式 (2) の右辺の各項を加算器 5 1 において加算するために、各項の指数部が揃えられる必要がある。シフト量演算回路 2 6 0 は、各項のシフト量を演算する。以下、具体的に説明する。

【0 1 7 7】

指数部抽出回路 261 には、倍精度演算モードを示す MODE 信号が供給されている。このため、指数部抽出回路 261 は、演算部 210A ~ 210D からの入力データ DIN1, DIN2 のそれぞれと、入力データ DIN0 と、から指数部を抽出する。具体的には、指数部抽出回路 261 は、倍精度データ A0 ~ A3、倍精度データ B00, B10, B20, B30、及び倍精度データ C0 の指数部を抽出する。

【0178】

そして、指数部抽出回路 261 は、倍精度データ A0 の指数部と倍精度データ B00 の指数部との和を演算することで、 $A0 \times B00$ の指数部を算出する。同様に、指数部抽出回路 261 は、倍精度データ A1 の指数部と倍精度データ B10 の指数部との和を演算することで、 $A1 \times B10$ の指数部を算出する。同様に、指数部抽出回路 261 は、倍精度データ A2 の指数部と倍精度データ B20 の指数部との和を演算することで、 $A2 \times B20$ の指数部を算出する。同様に、指数部抽出回路 261 は、倍精度データ A3 の指数部と倍精度データ B30 の指数部との和を演算することで、 $A3 \times B30$ の指数部を算出する。そして、指数部抽出回路 261 は、 $A0 \times B00$ の指数部、 $A1 \times B10$ の指数部、 $A2 \times B20$ の指数部、 $A3 \times B30$ の指数部、及び倍精度データ C0 の指数部を最大値演算回路 262A に出力する。また、指数部抽出回路 261 は、 $A0 \times B00$ の指数部、 $A1 \times B10$ の指数部、 $A2 \times B20$ の指数部、 $A3 \times B30$ の指数部をそれぞれ異なる減算回路 263A に出力する。

【0179】

最大値演算回路 262A は、 $A0 \times B00$ の指数部、 $A1 \times B10$ の指数部、 $A2 \times B20$ の指数部、 $A3 \times B30$ の指数部、及び倍精度データ C0 の指数部から最大の指数部を、最大指数として算出（特定）する。そして、最大値演算回路 262A は、各減算回路 263A に最大指数を出力する。

【0180】

そして、各減算回路 263A は、最大指数と各乗算結果の指数部との差を演算する。具体的には、 $A0 \times B00$ の指数部が供給されている減算回路 263A は、最大指数から $A0 \times B00$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 263A は、算出したシフト量を、演算部 210A の整列部 17, 18 に出力する。同様に、 $A1 \times B10$ の指数部が供給されている減算回路 263A は、最大指数から $A1 \times B10$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 263A は、算出したシフト量を、演算部 210B の整列部 17, 18 に出力する。

【0181】

また、 $A2 \times B20$ の指数部が供給されている減算回路 263A は、最大指数から $A2 \times B20$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 263A は、算出したシフト量を、演算部 210C の整列部 17, 18 に出力する。同様に、 $A3 \times B30$ の指数部が供給されている減算回路 263A は、最大指数から $A3 \times B30$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 263A は、算出したシフト量を、演算部 210D の整列部 17, 18 に出力する。

【0182】

各整列部 17 は、シフト量演算回路 260 から供給されたシフト量に基づいて、中間結果 P11 の仮数部に対してシフト処理を行う。同様に、各整列部 18 は、シフト量演算回路 260 から供給されたシフト量に基づいて、中間結果 P12 の仮数部に対してシフト処理を行う。

【0183】

次に、図 17 を用いて演算装置 1A ~ 1D の第 2 演算モードでの動作を説明する。図 17 は、図 8 の演算装置の第 2 演算モードでの動作を説明するための図である。ここでは、第 2 演算モードとして、単精度演算モードを用い、演算ユニット MAU が式 (7) に示される単精度の行列演算を行う場合を一例として説明する。演算装置 1A ~ 1D では、演算対象のデータが異なるものの、動作は同じであるので、ここでは演算装置 1A について説明する。また、演算部 210A ~ 210D では、演算対象のデータが異なるものの、動作

は同じであるので、ここでは演算部 210A について説明する。

【0184】

演算装置 1A の演算部 210A は、 $a_0 \times b_{00} + a_1 \times b_{10}$ の演算と、 $a_0 \times b_{01} + a_1 \times b_{11}$ の演算と、を行う。単精度データ a_0 は、単精度データ a_0 のビット列のうちの上半分のビット列であるデータ a'_{0_h} と、単精度データ a_0 のビット列のうちの下半分のビット列であるデータ a'_{0_l} と、を含む。単精度データ a_1 , b_{00} , b_{10} , b_{01} , b_{11} についても同様である。データ a'_{0_h} , a'_{0_l} , a'_{1_h} , a'_{1_l} , b'_{00_h} , b'_{00_l} , b'_{10_h} , b'_{10_l} , b'_{01_h} , b'_{01_l} , b'_{11_h} , b'_{11_l} は、半精度データと同じビット数を有する。

10

【0185】

演算部 210A は、式(17)に示されるように、単精度データ a_0 , a_1 及び単精度データ b_{00} , b_{10} のそれぞれを半精度データのビット数を有する 2 つのデータに分解することによって、単精度データ a_0 及び単精度データ b_{00} の乗算と、単精度データ a_1 及び単精度データ b_{10} の乗算と、を行う。

【数17】

$$\begin{aligned} a_0 \times b_{00} + a_1 \times b_{10} &= (a'_{0_h} + a'_{0_l}) \times (b'_{00_h} + b'_{00_l}) + (a'_{1_h} + a'_{1_l}) \times (b'_{10_h} + b'_{10_l}) \\ &= a'_{0_h} \times b'_{00_h} + a'_{0_h} \times b'_{00_l} + a'_{0_l} \times b'_{00_h} + a'_{0_l} \times b'_{00_l} + \\ &\quad a'_{1_h} \times b'_{10_h} + a'_{1_h} \times b'_{10_l} + a'_{1_l} \times b'_{10_h} + a'_{1_l} \times b'_{10_l} \quad \dots \quad (17) \end{aligned}$$

【0186】

同様に、演算部 210A は、式(18)に示されるように、単精度データ a_0 , a_1 及び単精度データ b_{01} , b_{11} のそれぞれを半精度データのビット数を有する 2 つのデータに分解することによって、単精度データ a_0 及び単精度データ b_{01} の乗算と、単精度データ a_1 及び単精度データ b_{11} の乗算と、を行う。

【数18】

$$\begin{aligned} a_0 \times b_{01} + a_1 \times b_{11} &= (a'_{0_h} + a'_{0_l}) \times (b'_{01_h} + b'_{01_l}) + (a'_{1_h} + a'_{1_l}) \times (b'_{11_h} + b'_{11_l}) \\ &= a'_{0_h} \times b'_{01_h} + a'_{0_h} \times b'_{01_l} + a'_{0_l} \times b'_{01_h} + a'_{0_l} \times b'_{01_l} + \\ &\quad a'_{1_h} \times b'_{11_h} + a'_{1_h} \times b'_{11_l} + a'_{1_l} \times b'_{11_h} + a'_{1_l} \times b'_{11_l} \quad \dots \quad (18) \end{aligned}$$

【0187】

以下、具体的に説明する。レジスタ 41 の上位ビットには、単精度データ a_0 が記憶されており、レジスタ 41 の下位ビットには単精度データ a_1 が記憶されている。レジスタ 41 からデータマルチプレクサ 211 に入力データ DIN1 として単精度データ a_0 及び単精度データ a_1 が供給されている。レジスタ 42 の上位ビットには、単精度データ b_{00} が記憶されており、レジスタ 42 の下位ビットには単精度データ b_{10} が記憶されている。レジスタ 42 からデータマルチプレクサ 211 に入力データ DIN2 として単精度データ b_{00} 及び単精度データ b_{10} が供給されている。レジスタ 43 の上位ビットには、単精度データ b_{01} が記憶されており、レジスタ 43 の下位ビットには単精度データ b_{11} が記憶されている。レジスタ 43 からデータマルチプレクサ 211 に入力データ DIN3 として単精度データ b_{01} 及び単精度データ b_{11} が供給されている。レジスタ 44 , 45 には、有効なデータは記憶されていない。

40

【0188】

データマルチプレクサ 211 には、単精度演算モードを示す MODE 信号が供給されている。このため、データマルチプレクサ 211 は、単精度データ a_0 及び単精度データ b_{00} を乗算器 212 に出力する。また、データマルチプレクサ 211 は、単精度データ a_1 及び単精度データ b_{10} を乗算器 213 に出力する。また、データマルチプレクサ 211 は、単精度データ a_0 及び単精度データ b_{01} を乗算器 214 に出力する。また、データマルチプレクサ 211 は、単精度データ a_1 及び単精度データ b_{11} を乗算器 215 に出力する。

50

【0189】

乗算器212では、データマルチプレクサ311には、単精度演算モードを示すMODE信号が供給されている。このため、データマルチプレクサ311は、データa'0__h及びデータb'00__hを乗算器312に出力し、データa'0__1及びデータb'00__hを乗算器313に出力し、データa'0__h及びデータb'00__1を乗算器314に出力し、データa'0__1及びデータb'00__1を乗算器315に出力する。そして、各乗算器312～315において乗算が行われ、その中間結果m'11, m'12, m'21, m'22, m'31, m'32, m'41, m'42を部分加算器316が加算し、中間結果m11及び中間結果m12を出力する。つまり、乗算器212では、 $a0 \times b00$ の演算が半精度データを用いて行われ、中間結果m11及び中間結果m12が出力される。

【0190】

同様に、乗算器313では、 $a1 \times b10$ の演算が半精度データを用いて行われ、中間結果m21及び中間結果m22が出力される。同様に、乗算器314では、 $a0 \times b01$ の演算が半精度データを用いて行われ、中間結果m31及び中間結果m32が出力される。同様に、乗算器315では、 $a1 \times b11$ の演算が半精度データを用いて行われ、中間結果m41及び中間結果m42が出力される。乗算器213～215の動作も乗算器212の動作と同様であるので、説明を省略する。以降の処理は、演算部10Aと同じであるので、説明を省略する。

【0191】

ここで、シフト量演算回路260の第2演算モードでの動作を説明する。演算装置1Aでは、式(8)及び式(9)に示される演算が行われる。このため、式(8)の右辺の各項を加算器52において加算するために、各項の指数部が揃えられる必要がある。同様に、式(9)の右辺の各項を加算器53において加算するために、各項の指数部が揃えられる必要がある。シフト量演算回路260は、各項のシフト量を演算する。以下、具体的に説明する。

【0192】

指数部抽出回路261には、単精度演算モードを示すMODE信号が供給されている。このため、指数部抽出回路261は、演算部210A～210Dからの入力データDIN1～DIN3のそれぞれと、入力データDIN0と、から指数部を抽出する。具体的には、指数部抽出回路261は、単精度データa0～a7、単精度データb00, b01, b10, b11, b20, b21, b30, b31, b40, b41, b50, b51, b60, b61, b70, b71、及び単精度データc0, c1の指数部を抽出する。

【0193】

そして、指数部抽出回路261は、単精度データa0の指数部と単精度データb00の指数部との和を演算することで、 $a0 \times b00$ の指数部を算出する。同様に、指数部抽出回路261は、 $a1 \times b10$ の指数部、 $a2 \times b20$ の指数部、 $a3 \times b30$ の指数部、 $a4 \times b40$ の指数部、 $a5 \times b50$ の指数部、 $a6 \times b60$ の指数部、及び $a7 \times b70$ の指数部を算出する。そして、指数部抽出回路261は、各乗算結果の指数部、及び単精度データc0の指数部を最大値演算回路262Aに出力する。また、指数部抽出回路261は、各乗算結果の指数部を互いに異なる減算回路263Aに出力する。

【0194】

同様に、指数部抽出回路261は、式(9)に示される右辺の各乗算結果の指数部を算出し、各乗算結果の指数部及び単精度データc1の指数部を最大値演算回路262Bに出力する。また、指数部抽出回路261は、各乗算結果の指数部を互いに異なる減算回路263Bに出力する。

【0195】

最大値演算回路262Aは、 $a0 \times b00$ の指数部、 $a1 \times b10$ の指数部、 $a2 \times b20$ の指数部、 $a3 \times b30$ の指数部、 $a4 \times b40$ の指数部、 $a5 \times b50$ の指数部、 $a6 \times b60$ の指数部、 $a7 \times b70$ の指数部、及び単精度データc0の指数部から最大

の指数部を、最大指数として算出（特定）する。そして、最大値演算回路 2 6 2 A は、各減算回路 2 6 3 A に最大指数を出力する。

【 0 1 9 6 】

そして、各減算回路 2 6 3 A は、最大指数と各乗算結果の指数部との差を演算する。具体的には、 $a_0 \times b_{00}$ の指数部が供給されている減算回路 2 6 3 A は、最大指数から $a_0 \times b_{00}$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 2 6 3 A は、算出したシフト量を、演算部 2 1 0 A の整列部 2 1, 2 2 に出力する。同様に、 $a_1 \times b_{10}$ の指数部が供給されている減算回路 2 6 3 A は、最大指数から $a_1 \times b_{10}$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 2 6 3 A は、算出したシフト量を、演算部 2 1 0 A の整列部 2 3, 2 4 に出力する。

10

【 0 1 9 7 】

また、 $a_2 \times b_{20}$ の指数部が供給されている減算回路 2 6 3 A は、最大指数から $a_2 \times b_{20}$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 2 6 3 A は、算出したシフト量を、演算部 2 1 0 B の整列部 2 1, 2 2 に出力する。このように、各減算回路 2 6 3 A は、供給されている乗算の乗算結果（中間結果）をシフト処理する整列部に、算出したシフト量を出力する。最大値演算回路 2 6 2 B 及び減算回路 2 6 3 B の動作も同様であるので、説明を省略する。

【 0 1 9 8 】

そして、各整列部は、シフト量演算回路 2 6 0 から供給されたシフト量に基づいて、中間結果の仮数部に対してシフト処理を行う。

20

【 0 1 9 9 】

次に、図 1 8 を用いて演算装置 1 A ~ 1 D の第 3 演算モードでの動作を説明する。図 1 8 は、図 8 の演算装置の第 3 演算モードでの動作を説明するための図である。ここでは、第 3 演算モードとして、半精度演算モードを用い、演算ユニット MAU が式 (1 9) に示される単精度の行列演算を行う場合を一例として説明する。

【 数 1 9 】

$$(a'0 \ a'1 \ \dots \ a'15) \times \begin{pmatrix} b'00 & b'01 & \dots & b'0F \\ b'10 & b'11 & \dots & b'1F \\ \vdots & \vdots & \ddots & \vdots \\ b'F0 & b'F1 & \dots & b'FF \end{pmatrix} + (c'0 \ c'1 \ \dots \ c'15) \ \dots \quad (19)$$

【 0 2 0 0 】

演算装置 1 A は、式 (2 0) ~ 式 (2 3) に示される行列演算を行う。

【 数 2 0 】

$$d'0 = (a'0, a'1, \dots, a'15) \times (b'00, b'10, \dots, b'F0) + c'0 \ \dots \quad (20)$$

【 数 2 1 】

$$d'1 = (a'0, a'1, \dots, a'15) \times (b'01, b'11, \dots, b'F1) + c'1 \ \dots \quad (21)$$

【 数 2 2 】

$$d'2 = (a'0, a'1, \dots, a'15) \times (b'02, b'12, \dots, b'F2) + c'2 \ \dots \quad (22)$$

【 数 2 3 】

$$d'3 = (a'0, a'1, \dots, a'15) \times (b'03, b'13, \dots, b'F3) + c'3 \ \dots \quad (23)$$

【 0 2 0 1 】

50

演算装置 1 A ~ 1 D では、演算対象のデータが異なるものの、動作は同じであるので、ここでは演算装置 1 A について説明する。演算装置 1 A の演算部 2 1 0 A は、 $a' 0 \times b' 0 0 + a' 1 \times b' 1 0 + a' 2 \times b' 2 0 + a' 3 \times b' 3 0$ の演算、 $a' 0 \times b' 0 1 + a' 1 \times b' 1 1 + a' 2 \times b' 2 1 + a' 3 \times b' 3 1$ の演算、 $a' 0 \times b' 0 2 + a' 1 \times b' 1 2 + a' 2 \times b' 2 2 + a' 3 \times b' 3 2$ の演算、及び $a' 0 \times b' 0 3 + a' 1 \times b' 1 3 + a' 2 \times b' 2 3 + a' 3 \times b' 3 3$ の演算を行う。演算装置 1 A の演算部 2 1 0 B は、 $a' 0 \times b' 4 0 + a' 1 \times b' 5 0 + a' 2 \times b' 6 0 + a' 3 \times b' 7 0$ の演算、 $a' 0 \times b' 4 1 + a' 1 \times b' 5 1 + a' 2 \times b' 6 1 + a' 3 \times b' 7 1$ の演算、 $a' 0 \times b' 4 2 + a' 1 \times b' 5 2 + a' 2 \times b' 6 2 + a' 3 \times b' 7 2$ の演算、及び $a' 0 \times b' 4 3 + a' 1 \times b' 5 3 + a' 2 \times b' 6 3 + a' 3 \times b' 7 3$ の演算を行う。

【0 2 0 2】

演算装置 1 A の演算部 2 1 0 C は、 $a' 0 \times b' 8 0 + a' 1 \times b' 9 0 + a' 2 \times b' A 0 + a' 3 \times b' B 0$ の演算、 $a' 0 \times b' 8 1 + a' 1 \times b' 9 1 + a' 2 \times b' A 1 + a' 3 \times b' B 1$ の演算、 $a' 0 \times b' 8 2 + a' 1 \times b' 9 2 + a' 2 \times b' A 2 + a' 3 \times b' B 2$ の演算、及び $a' 0 \times b' 8 3 + a' 1 \times b' 9 3 + a' 2 \times b' A 3 + a' 3 \times b' B 3$ の演算を行う。演算装置 1 A の演算部 2 1 0 D は、 $a' 0 \times b' C 0 + a' 1 \times b' D 0 + a' 2 \times b' E 0 + a' 3 \times b' F 0$ の演算、 $a' 0 \times b' C 1 + a' 1 \times b' D 1 + a' 2 \times b' E 1 + a' 3 \times b' F 1$ の演算、 $a' 0 \times b' C 2 + a' 1 \times b' D 2 + a' 2 \times b' E 2 + a' 3 \times b' F 2$ の演算、及び $a' 0 \times b' C 3 + a' 1 \times b' D 3 + a' 2 \times b' E 3 + a' 3 \times b' F 3$ の演算を行う。各演算部 1 0 A ~ 1 0 D では、演算対象のデータが異なるものの、動作は同じであるので、演算部 1 0 A の動作を主に説明する。

【0 2 0 3】

以下、具体的に説明する。レジスタ 4 1 には、上位ビットから順に半精度データ $a' 0$ 、半精度データ $a' 1$ 、半精度データ $a' 2$ 、及び半精度データ $a' 3$ が記憶されている。レジスタ 4 1 からデータマルチプレクサ 2 1 1 に入力データ $DIN 1$ として半精度データ $a' 0 \sim a' 3$ が供給されている。レジスタ 4 2 には、上位ビットから順に半精度データ $b' 0 0$ 、半精度データ $b' 1 0$ 、半精度データ $b' 2 0$ 、及び半精度データ $b' 3 0$ が記憶されている。レジスタ 4 2 からデータマルチプレクサ 2 1 1 に入力データ $DIN 2$ として半精度データ $b' 0 0$ 、 $b' 1 0$ 、 $b' 2 0$ 、 $b' 3 0$ が供給されている。

【0 2 0 4】

レジスタ 4 3 には、上位ビットから順に半精度データ $b' 0 1$ 、半精度データ $b' 1 1$ 、半精度データ $b' 2 1$ 、及び半精度データ $b' 3 1$ が記憶されている。レジスタ 4 3 からデータマルチプレクサ 2 1 1 に入力データ $DIN 3$ として半精度データ $b' 0 1$ 、 $b' 1 1$ 、 $b' 2 1$ 、 $b' 3 1$ が供給されている。レジスタ 4 4 には、上位ビットから順に半精度データ $b' 0 2$ 、半精度データ $b' 1 2$ 、半精度データ $b' 2 2$ 、及び半精度データ $b' 3 2$ が記憶されている。レジスタ 4 4 からデータマルチプレクサ 2 1 1 に入力データ $DIN 4$ として半精度データ $b' 0 2$ 、 $b' 1 2$ 、 $b' 2 2$ 、 $b' 3 2$ が供給されている。レジスタ 4 5 には、上位ビットから順に半精度データ $b' 0 3$ 、半精度データ $b' 1 3$ 、半精度データ $b' 2 3$ 、及び半精度データ $b' 3 3$ が記憶されている。レジスタ 4 5 からデータマルチプレクサ 2 1 1 に入力データ $DIN 5$ として半精度データ $b' 0 3$ 、 $b' 1 3$ 、 $b' 2 3$ 、 $b' 3 3$ が供給されている。

【0 2 0 5】

データマルチプレクサ 2 1 1 には、半精度演算モードを示す $MODE$ 信号が供給されている。このため、データマルチプレクサ 2 1 1 は、半精度データ $a' 0$ 、 $a' 1$ 、半精度データ $b' 0 0$ 、 $b' 1 0$ 、及び半精度データ $b' 0 1$ 、 $b' 1 1$ を乗算器 2 1 2 に出力する。また、データマルチプレクサ 2 1 1 は、半精度データ $a' 2$ 、 $a' 3$ 、半精度データ $b' 2 0$ 、 $b' 3 0$ 、及び半精度データ $b' 2 1$ 、 $b' 3 1$ を乗算器 2 1 3 に出力する。また、データマルチプレクサ 2 1 1 は、半精度データ $a' 0$ 、 $a' 1$ 、半精度データ $b' 0 0$ 、 $b' 1 0$ 、及び半精度データ $b' 0 1$ 、 $b' 1 1$ を乗算器 2 1 2 に出力する。また、データマルチプレクサ 2 1 1 は、半精度データ $a' 2$ 、 $a' 3$ 、半精度データ $b' 2 0$ 、 $b' 3 0$ 、及び半精度データ $b' 2 1$ 、 $b' 3 1$ を乗算器 2 1 3 に出力する。

' 0 2 , b ' 1 2、及び半精度データ b ' 0 3 , b ' 1 3 を乗算器 2 1 4 に出力する。また、データマルチプレクサ 2 1 1 は、半精度データ a ' 2 , a ' 3、半精度データ b ' 2 2 , b ' 3 2、及び半精度データ b ' 2 3 , b ' 3 3 を乗算器 2 1 5 に出力する。

【 0 2 0 6 】

乗算器 2 1 2 では、データマルチプレクサ 3 1 1 には、半精度演算モードを示す M O D E 信号が供給されている。このため、データマルチプレクサ 3 1 1 は、半精度データ a ' 0 及び半精度データ b ' 0 0 を乗算器 3 1 2 に出力し、半精度データ a ' 1 及び半精度データ b ' 1 0 を乗算器 3 1 3 に出力し、半精度データ a ' 0 及び半精度データ b ' 0 1 を乗算器 3 1 4 に出力し、半精度データ a ' 1 及び半精度データ b ' 1 1 を乗算器 3 1 5 に出力する。そして、各乗算器 3 1 2 ~ 3 1 5 において乗算が行われ、その中間結果 m ' 1 1 , m ' 1 2 , m ' 2 1 , m ' 2 2 , m ' 3 1 , m ' 3 2 , m ' 4 1 , m ' 4 2 が出力される。乗算器 2 1 3 ~ 2 1 5 の動作も乗算器 2 1 2 の動作と同様であるので、説明を省略する。

【 0 2 0 7 】

ここで、シフト量演算回路 2 6 0 の第 3 演算モードでの動作を説明する。演算部 2 1 0 A では、式 (2 0) ~ 式 (2 3) に示される演算が行われる。このため、式 (2 0) の右辺の各項を加算器 5 4 において加算するために、各項の指数部が揃えられる必要がある。同様に、式 (2 1) の右辺の各項を加算器 5 5 において加算するために、各項の指数部が揃えられる必要がある。式 (2 2) の右辺の各項を加算器 5 6 において加算するために、各項の指数部が揃えられる必要がある。式 (2 3) の右辺の各項を加算器 5 7 において加算するために、各項の指数部が揃えられる必要がある。シフト量演算回路 2 6 0 は、各項のシフト量を演算する。以下、具体的に説明する。

【 0 2 0 8 】

指数部抽出回路 2 6 1 には、半精度演算モードを示す M O D E 信号が供給されている。このため、指数部抽出回路 2 6 1 は、演算部 2 1 0 A ~ 2 1 0 D からの入力データ D I N 1 ~ D I N 5 のそれぞれと、入力データ D I N 0 と、から指数部を抽出する。具体的には、指数部抽出回路 2 6 1 は、半精度データ a ' 0 ~ a ' 1 5、半精度データ b ' 0 0 ~ b ' 0 3 , b ' 1 0 ~ b ' 1 3 , . . . , b ' F 0 ~ F 3、及び半精度データ c ' 0 ~ c ' 3 の指数部を抽出する。

【 0 2 0 9 】

そして、指数部抽出回路 2 6 1 は、半精度データ a ' 0 の指数部と半精度データ b ' 0 0 の指数部との和を演算することで、 a ' 0 x b ' 0 0 の指数部を算出する。同様に、指数部抽出回路 2 6 1 は、式 (2 0) に示される右辺の各乗算結果の指数部を算出する。そして、指数部抽出回路 2 6 1 は、各乗算結果の指数部、及び半精度データ c ' 0 の指数部を最大値演算回路 2 6 2 A に出力する。また、指数部抽出回路 2 6 1 は、各乗算結果の指数部を互いに異なる減算回路 2 6 3 A に出力する。

【 0 2 1 0 】

また、指数部抽出回路 2 6 1 は、式 (2 1) に示される右辺の各乗算結果の指数部を算出し、各乗算結果の指数部及び半精度データ c ' 1 の指数部を最大値演算回路 2 6 2 B に出力する。また、指数部抽出回路 2 6 1 は、各乗算結果の指数部を互いに異なる減算回路 2 6 3 B に出力する。同様に、指数部抽出回路 2 6 1 は、式 (2 2) に示される右辺の各乗算結果の指数部を算出し、各乗算結果の指数部及び半精度データ c ' 2 の指数部を最大値演算回路 2 6 2 C に出力する。また、指数部抽出回路 2 6 1 は、各乗算結果の指数部を互いに異なる減算回路 2 6 3 C に出力する。指数部抽出回路 2 6 1 は、式 (2 3) に示される右辺の各乗算結果の指数部を算出し、各乗算結果の指数部及び半精度データ c ' 3 の指数部を最大値演算回路 2 6 2 D に出力する。また、指数部抽出回路 2 6 1 は、各乗算結果の指数部を互いに異なる減算回路 2 6 3 D に出力する。

【 0 2 1 1 】

最大値演算回路 2 6 2 A は、各乗算結果の指数部、及び半精度データ c ' 0 の指数部から最大の指数部を、最大指数として算出 (特定) する。そして、最大値演算回路 2 6 2 A

は、各減算回路 263A に最大指数を出力する。

【0212】

そして、各減算回路 263A は、最大指数と各乗算結果の指数部との差を演算する。具体的には、 $a'0 \times b'00$ の指数部が供給されている減算回路 263A は、最大指数から $a'0 \times b'00$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 263A は、算出したシフト量を、演算部 210A の乗算器 212 の整列部 321, 322 に出力する。同様に、 $a'1 \times b'10$ の指数部が供給されている減算回路 263A は、最大指数から $a'1 \times b'10$ の指数部を減算し、その減算結果をシフト量として算出する。そして、減算回路 263A は、算出したシフト量を、演算部 210A の乗算器 212 の整列部 323, 324 に出力する。

10

【0213】

このように、各減算回路 263A は、供給されている乗算の乗算結果（中間結果）をシフト処理する整列部に、算出したシフト量を出力する。最大値演算回路 262B ~ 262D 及び減算回路 263B ~ 263D の動作も同様であるので、説明を省略する。

【0214】

そして、整列部 321 ~ 328 は、シフト量演算回路 260 から供給されたシフト量に基づいて、中間結果 $m'11$, $m'12$, $m'21$, $m'22$, $m'31$, $m'32$, $m'41$, $m'42$ の仮数部に対してシフト処理を行い、整列部 321 ~ 328 の処理結果を丸め処理部 331 ~ 338 に出力する。そして、丸め処理部 331 ~ 334 は、整列部 321 ~ 324 によってシフト処理が行われた中間結果 $m'11$, $m'12$, $m'21$, $m'22$ に対して丸め処理を行い、演算結果 $r'11$, $r'12$, $r'21$, $r'22$ を加算器 54 に出力する。同様に、丸め処理部 335 ~ 338 は、整列部 325 ~ 328 によってシフト処理が行われた中間結果 $m'31$, $m'32$, $m'41$, $m'42$ に対して丸め処理を行い、演算結果 $r'31$, $r'32$, $r'41$, $r'42$ を加算器 55 に出力する。乗算器 213 ~ 215 においても、同様の演算が行われる。

20

【0215】

演算部 210B、演算部 210C、及び演算部 210D においても、同様の演算が行われる。各演算部 210A ~ 210D の乗算器 212, 213 は、演算結果 $r'11$, $r'12$, $r'21$, $r'22$ を加算器 54 に出力するとともに、演算結果 $r'31$, $r'32$, $r'41$, $r'42$ を加算器 55 に出力する。各演算部 210A ~ 210D の乗算器 214, 215 は、演算結果 $r'11$, $r'12$, $r'21$, $r'22$ を加算器 56 に出力するとともに、演算結果 $r'31$, $r'32$, $r'41$, $r'42$ を加算器 57 に出力する。また、レジスタ 40 には、上位ビットから順に半精度データ $c'0$ 、半精度データ $c'1$ 、半精度データ $c'2$ 、及び半精度データ $c'3$ が記憶されている。そして、レジスタ 40 から加算器 54 に半精度データ $c'0$ が供給され、加算器 55 に半精度データ $c'1$ が供給され、加算器 56 に半精度データ $c'2$ が供給され、加算器 57 に半精度データ $c'3$ が供給されている。

30

【0216】

そして、加算器 54 は、演算部 210A ~ 210D の乗算器 212, 213 の演算結果 $r'11$, $r'12$, $r'21$, $r'22$ 、及び半精度データ $c'0$ の加算を行い、加算器 54 の加算結果を演算結果 $d'0$ として出力する。また、加算器 55 は、演算部 210A ~ 210D の乗算器 212, 213 の演算結果 $r'31$, $r'32$, $r'41$, $r'42$ 、及び半精度データ $c'1$ の加算を行い、加算器 55 の加算結果を演算結果 $d'1$ として出力する。同様に、加算器 56 は、演算部 210A ~ 210D の乗算器 214, 215 の演算結果 $r'11$, $r'12$, $r'21$, $r'22$ 、及び半精度データ $c'2$ の加算を行い、加算器 56 の加算結果を演算結果 $d'2$ として出力する。また、加算器 57 は、演算部 210A ~ 210D の乗算器 214, 215 の演算結果 $r'31$, $r'32$, $r'41$, $r'42$ 、及び半精度データ $c'3$ の加算を行い、加算器 57 の加算結果を演算結果 $d'3$ として出力する。

40

【0217】

50

そして、演算結果 $d'0 \sim d'3$ が組み合わされて演算結果 ($d'0, d'1, d'2, d'3$) が生成され、マルチプレクサ 259 に供給される。このとき、部分加算器 316、部分加算器 16、整列部 17, 18、丸め処理部 19, 20、整列部 21~28、丸め処理部 31~38、及び加算器 51~53 も演算を行っており、演算結果 $D0$ 及び演算結果 ($d0, d1$) がマルチプレクサ 259 に出力されている。マルチプレクサ 259 には、半精度演算モードを示す MODE 信号が供給されているので、マルチプレクサ 259 は、演算結果 ($d'0, d'1, d'2, d'3$) を出力する。このようにして、4つの半精度の行列演算が行われる。

【0218】

第2実施形態に係る演算装置 1A~1D、及びそれらを備える演算システム S においても、上述した第1実施形態に係る演算システム S、及び演算装置 1A~1D と同様の効果が奏される。

【0219】

第2実施形態に係る演算装置 1A~1D では、加算器 51 によって、第1ビット数を有するデータでの1つの行列演算結果が得られ、加算器 52 及び加算器 53 によって、第2ビット数を有するデータでの2つの行列演算結果が得られ、さらに加算器 54~57 によって、第3ビット数を有するデータでの4つの行列演算結果が得られる。そして、マルチプレクサ 259 によって、MODE 信号が第1演算モードを示す場合には、第1ビット数を有するデータでの1つの行列演算結果が出力され、MODE 信号が第2演算モードを示す場合には、第2ビット数を有するデータでの2つの行列演算結果が出力され、MODE 信号が第3演算モードを示す場合には、第3ビット数を有するデータでの4つの行列演算結果が出力される。このように、演算装置 1A~1D では、3つの演算モードに応じた行列演算を行うことが可能となる。

【0220】

半精度演算モードの1演算に用いられる演算器の回路規模は、倍精度演算モードの1演算に用いられる演算器の16分の1以下である。しかし、上述の比較例の演算装置と同様に、倍精度演算モードの1演算を半精度演算モードの4演算に切り替えると、倍精度演算モードの1演算を行う回路規模あたり、半精度演算モードでの演算が4つしか行えない。このため、倍精度演算モードの1演算を半精度演算モードの4演算に切り替えると、倍精度演算モードの1演算に用いられるトランジスタ等の回路素子の4分の1程度の回路素子しか使用されない。これに対し、上記構成の演算装置 1A~1D によれば、半精度演算モード及び単精度演算モードであっても、演算性能を2~4倍に向上させることができ、電力効率も改善することができる。

【0221】

さらに、演算システム S では、演算ユニット MAU は、4つの演算装置 1A~1D を備えることにより、式(1)に示されるような倍精度データの行列演算 (4×4 の行列ベクトル積)を行うことができる。演算ユニット MAU は、単精度演算モードでは式(7)に示されるような 8×8 の行列ベクトル積、半精度演算モードでは式(19)に示されるような 16×16 の行列ベクトル積を行うことが可能となる。この構成によれば、倍精度演算モードでは、それぞれが52ビットの仮数の乗算を行う16個の並列乗算器(1サイクルで1乗算を行う回路)を有する回路に相当する演算性能を実現することができる。単精度演算モードでは、それぞれが23ビットの仮数の乗算を行う64個の並列乗算器を有する回路に相当する演算性能を実現することができる。半精度演算モードでは、それぞれが10ビットの仮数の乗算を行う256個の並列乗算器を有する回路に相当する演算性能を実現することができる。

【0222】

第1演算モードでは、加算器 51 が演算部 210A~210D の中間結果 $P11, P12$ を加算するためには、それぞれの指数部を揃える必要がある。第2演算モードでは、加算器 52 が演算部 210A~210D の中間結果 $m11, m12, m21, m22$ を加算するためには、それぞれの指数部を揃える必要があり、加算器 53 が演算部 210A~2

10 D の中間結果 m_{31} , m_{32} , m_{41} , m_{42} を加算するためには、それぞれの指数部を揃える必要がある。同様に、第3演算モードでは、加算器 54 ~ 57 が演算部 210 A ~ 210 D の乗算器 212 ~ 215 の中間結果 m'_{11} , m'_{12} , m'_{21} , m'_{22} , m'_{31} , m'_{32} , m'_{41} , m'_{42} を加算するためには、加算対象となるデータの指数部を揃える必要がある。シフト量演算回路 260 によって、加算対象となるデータの最大指数と各指数部との差分をシフト量として演算することによって、加算対象となるデータの指数部を揃えることが可能となる。

【0223】

最大値演算回路 262 では、複数の比較対象のデータが MSB から順に比較されるので、比較するビット数を減らすことができ、最大指数の演算を高速化することが可能となる。

【0224】

以上、本発明の実施形態について説明したが、本発明は上記実施形態に限られない。

【0225】

例えば、演算装置 1A ~ 1D は、データフローグラフをマッピングできる再構成可能な演算器として利用されてもよい。

【0226】

演算装置 1A ~ 1D は、レジスタ 40 ~ 45 を備えなくてもよい。演算装置 1A ~ 1D の外部からデータマルチプレクサ 11 に入力データ D_{IN0} ~ D_{IN3} が直接供給されてもよく、演算装置 1A ~ 1D の外部からデータマルチプレクサ 211 に入力データ D_{IN0} ~ D_{IN5} が直接供給されてもよい。

【0227】

また、演算装置 1A ~ 1D の演算モードとして、倍精度演算モード、単精度演算モード、及び半精度演算モードが挙げられているが、これらに限られない。第1演算モードの演算で用いられるデータの第1ビット数が、第2演算モードの演算で用いられるデータの第2ビット数の2倍であればよい。また、第3演算モードの演算で用いられるデータの第3ビット数が、第2演算モードの演算で用いられるデータの第2ビット数の半分であればよい。例えば、4倍精度演算モード、及び1/4精度演算モード等が採用されてもよい。例えば、第1演算モードとして4倍精度演算モードが採用され、第2演算モードとして倍精度演算モードが採用され、第3演算モードとして単精度演算モードが採用され得る。また、第1演算モードとして単精度演算モードが採用され、第2演算モードとして半精度演算モードが採用され、第3演算モードとして1/4精度演算モードが採用され得る。採用される演算モードの精度を低下させることにより、演算装置 1A ~ 1D の演算性能のさらなる向上が可能となる。

【0228】

さらに、演算装置 1A ~ 1D において採用される演算モードは異なってもよい。例えば、演算装置 1A ~ 1C は、単精度演算モードと半精度演算モードを採用し、演算装置 1D は、倍精度演算モードと単精度演算モードを採用してもよい。この場合、演算ユニット MAU としての実効的な演算精度を、演算装置 1A ~ 1C の演算精度よりも向上させることができる。また、アプリケーションで必要な演算精度を維持しつつ、高い演算性能を実現できる。

【0229】

さらに、演算装置 1A ~ 1D は、MODE 信号に応じて、演算装置 1A ~ 1D の演算モードを第1演算モード及び第3演算モードのいずれかに切り替え可能であってもよい。例えば、演算装置 1A ~ 1D の演算モードとして、倍精度演算モード及び半精度モードが採用されてもよい。この場合、第2演算モード用の回路（整列部 21 ~ 28、丸め処理部 31 ~ 38、加算器 52, 53）を省略することができる。これにより、回路規模を小さくことができ、性能向上及び省電力化が可能となる。さらに、この構成において、演算装置 1A ~ 1D において採用される演算モードは異なってもよい。この場合、広いアプリケーションレンジで、高い性能と省電力性とを両立させることができる。

【0230】

上記実施形態では、演算装置1A～1Dのそれぞれは、4つの演算部を備えているが、演算部の数は、演算目的に応じて適宜変更され得る。このため、演算装置1A～1Dのそれぞれは、少なくとも1つの演算部を備えていればよく、任意の複数個の演算部を備えてもよい。演算装置1A～1Dのそれぞれは、演算部とは別に1又は複数の倍精度乗算器をさらに備えてもよい。

【0231】

第1実施形態の演算装置1A～1Dは、シフト量演算回路260を備えていてもよい。

【0232】

乗算器312～315は、乗算器212～215の構成をさらに備えていてもよい。この場合、演算装置1A～1Dは、第3ビット数の半分の第4ビット数のデータを用いた第4演算モードをさらに行うことが可能となる。さらに、各乗算器に乗算器212～215の構成が繰り返し適用されてもよい。これにより、所望の精度での演算が可能となる。

【0233】

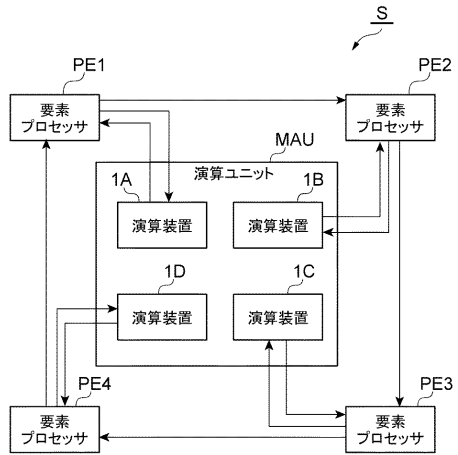
上記実施形態及び各変形例において、各演算部10A～10Dは、レジスタ44をさらに備えてもよい。レジスタ44は、演算対象となる第1ビット数のビット列である入力データDIN4を記憶保持する回路である。入力データDIN4は、上位データdin4_hと、下位データdin4_lと、を含む。入力データDIN4は、演算装置1A～1Dの外部の装置（本実施形態では、要素プロセッサ）からレジスタ44にセットされる。上位データdin4_h、及び下位データdin4_lは、第2ビット数のビット列である。レジスタ44は、保持している入力データDIN4をデータマルチプレクサ11に出力する。この場合、データマルチプレクサ11に4つの第1ビット数のデータが供給される。データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データdin1_hを出力データdout1として出力し、下位データdin1_lを出力データdout3として出力する。また、データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データdin2_hを出力データdout2として出力し、下位データdin2_lを出力データdout4として出力する。また、データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データdin3_hを出力データdout5として出力し、下位データdin3_lを出力データdout7として出力する。また、データマルチプレクサ11は、MODE信号が第2演算モードを示す場合には、上位データdin4_hを出力データdout6として出力し、下位データdin4_lを出力データdout8として出力する。この構成によれば、演算部10A～10Dのそれぞれは、第2ビット数のビット列の乗算を行う4つの乗算器として機能する。このため、演算装置1A～1Dは行列演算以外の演算に用いられ得る。演算部210A～210Dについても同様に構成され得る。

【符号の説明】

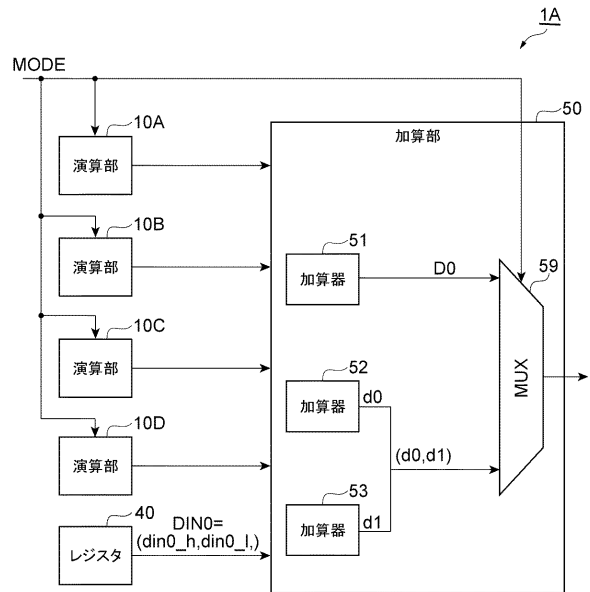
【0234】

1A～1D 演算装置、10A～10D、210A～210D 演算部、11、211、311 データマルチプレクサ（マルチプレクサ）、12、212、312 乗算器（第1乗算器）、13、213、313 乗算器（第2乗算器）、14、214、314 乗算器（第3乗算器）、15、215、315 乗算器（第4乗算器）、16、316 部分加算器、50、250 加算部、51 加算器（第3加算器）、52 加算器（第1加算器）、53 加算器（第2加算器）、59、259 マルチプレクサ（セレクタ）。

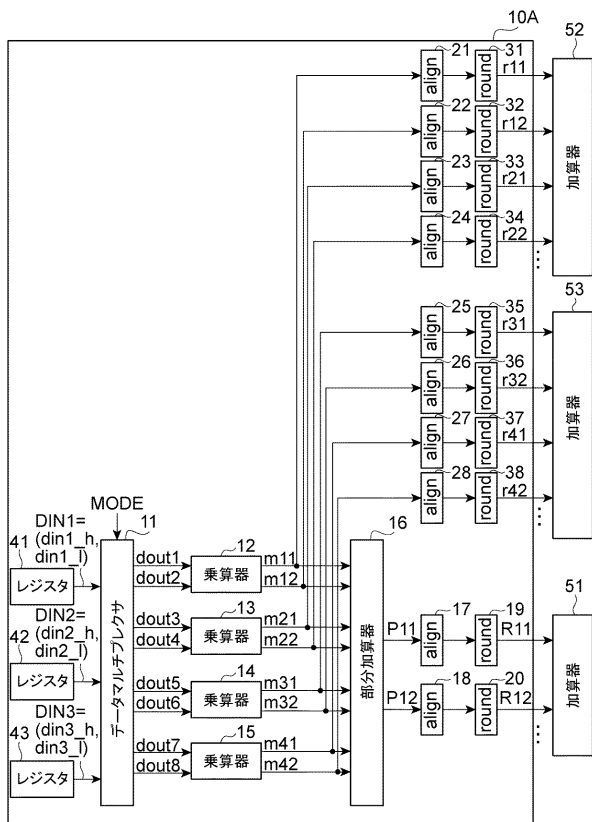
【図1】



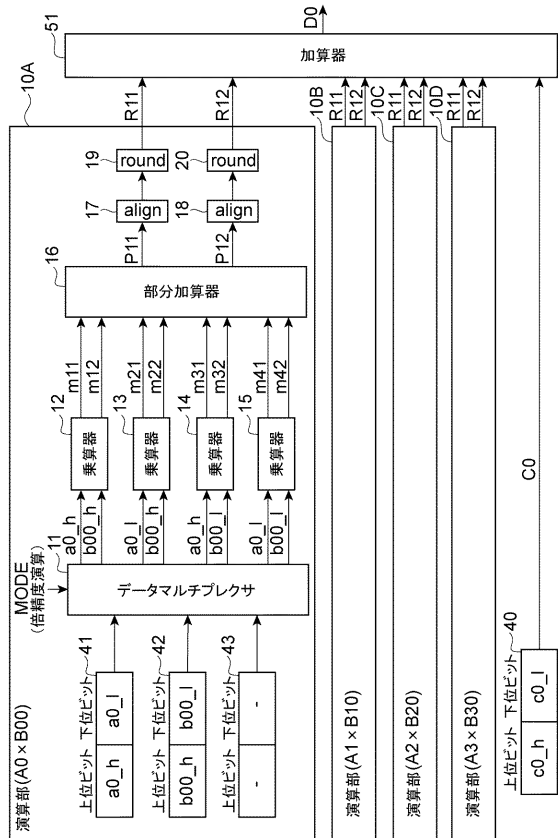
【図2】



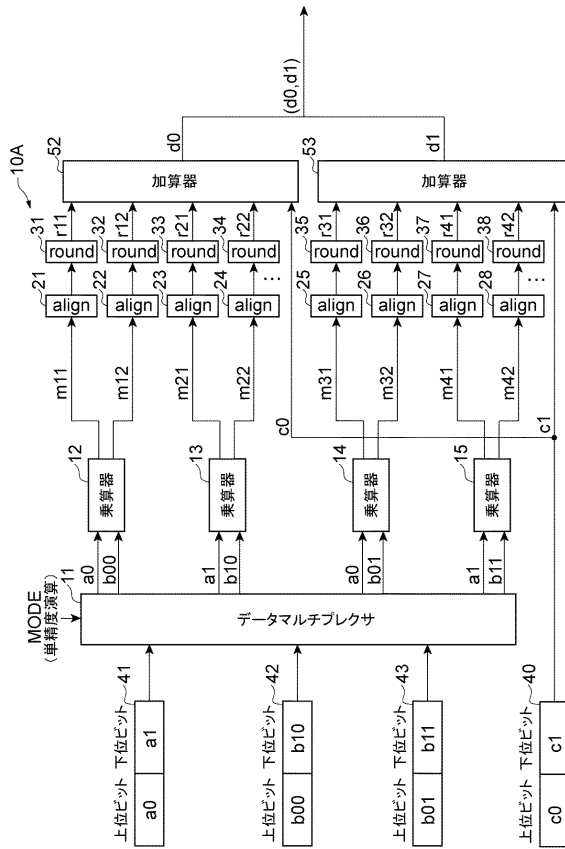
【図3】



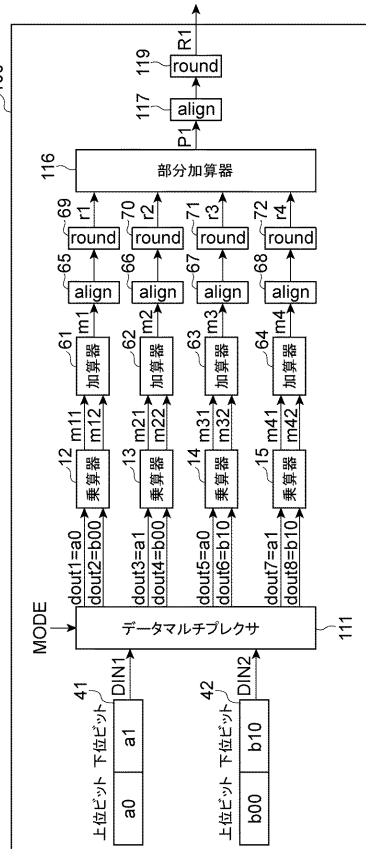
【図4】



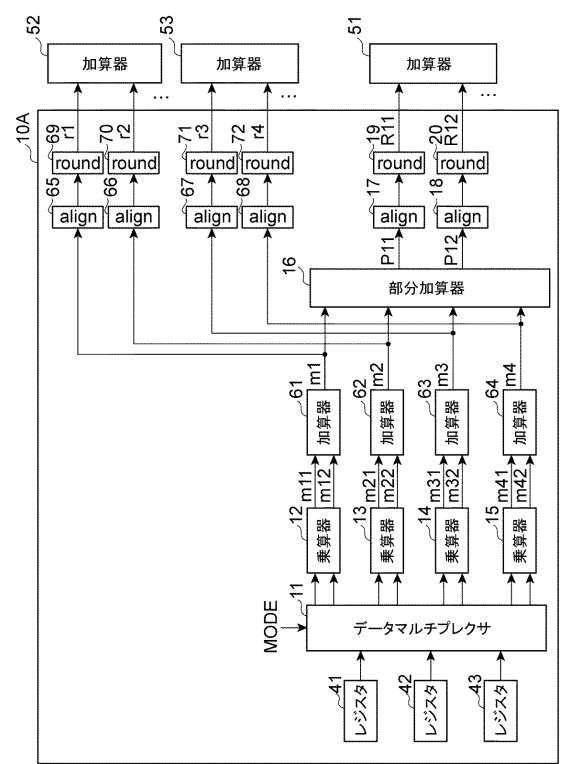
【図5】



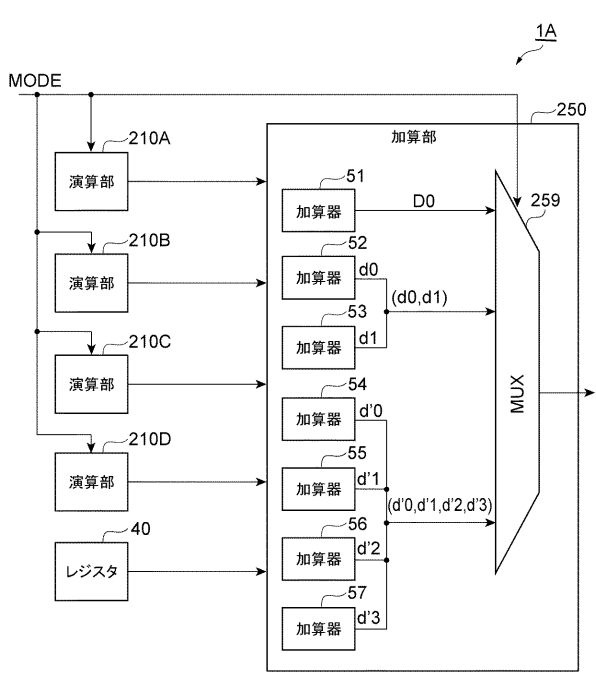
【図6】



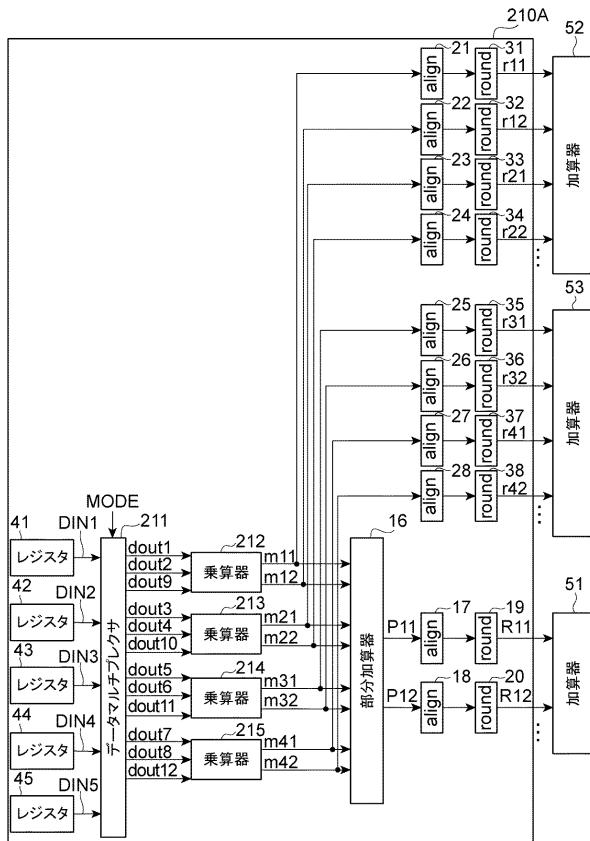
【図7】



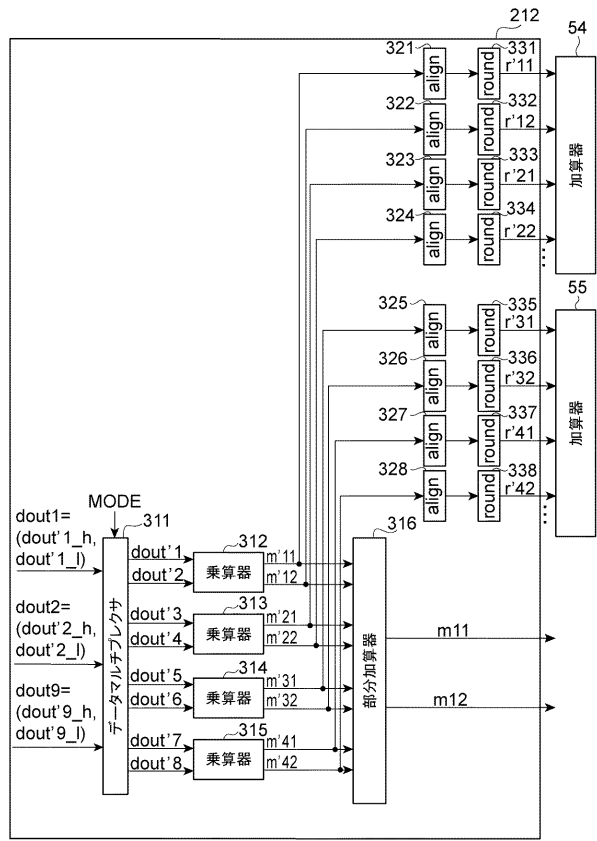
【図8】



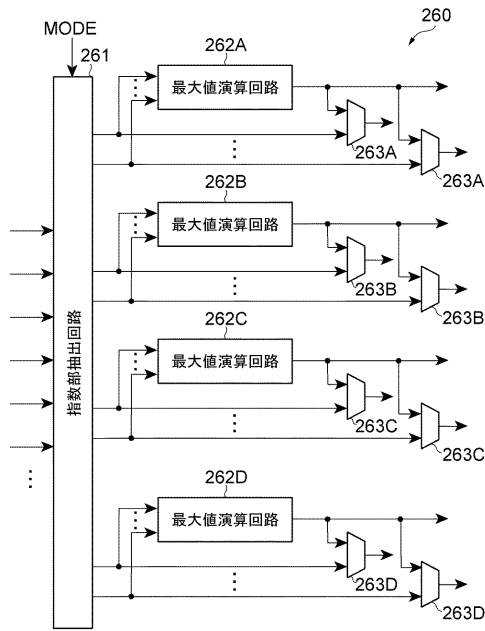
【図 9】



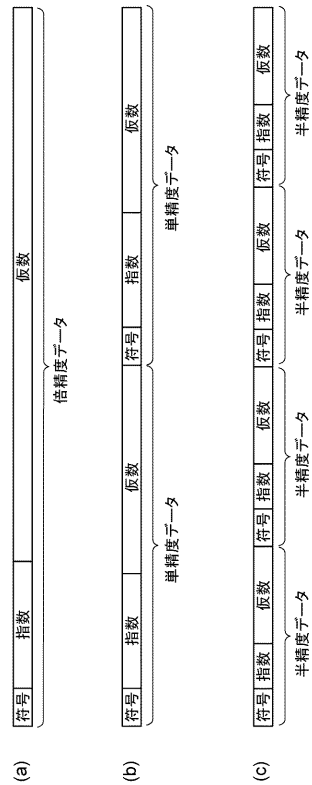
【図 10】



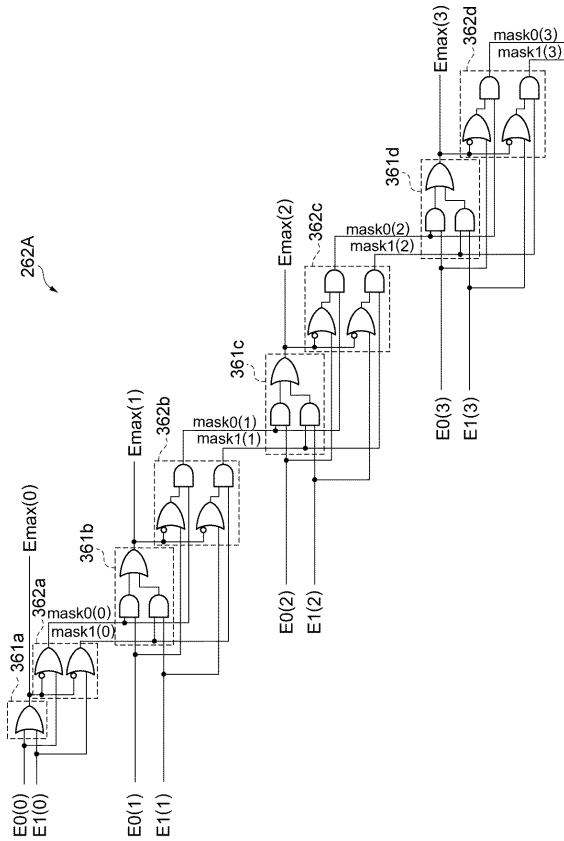
【図 11】



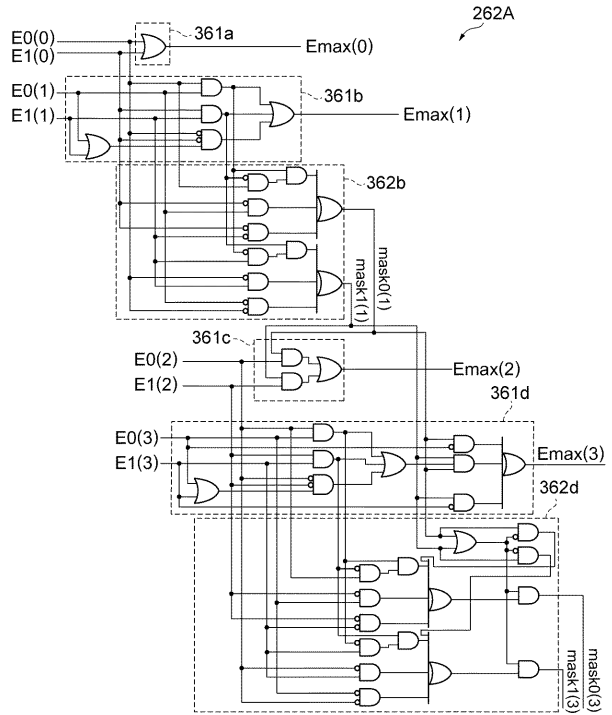
【図 12】



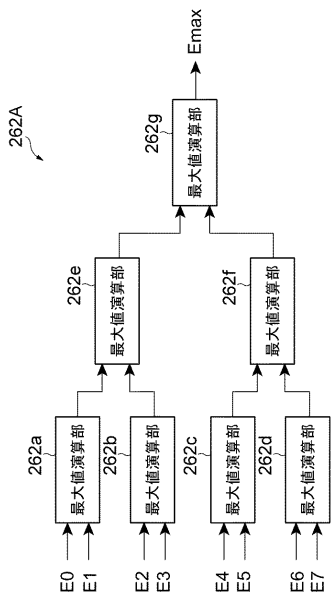
【図 13】



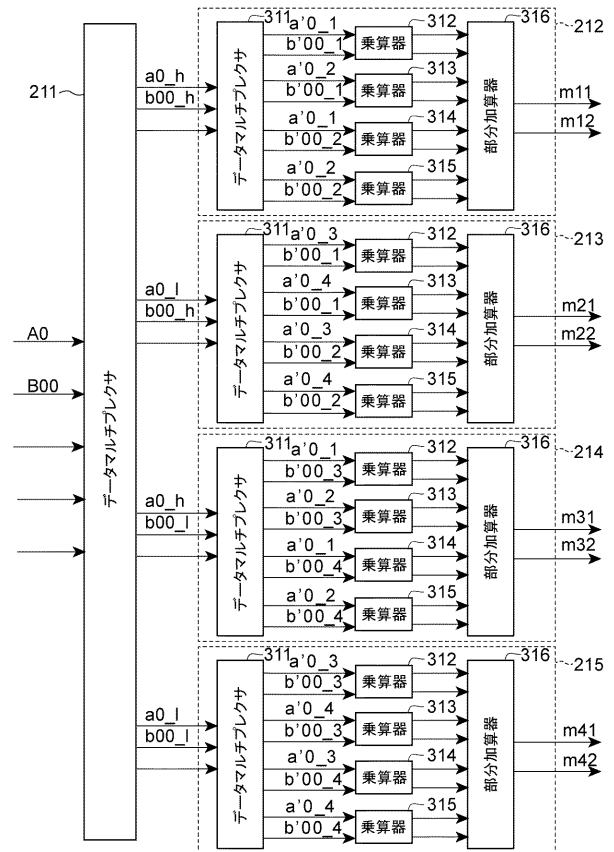
【図 14】



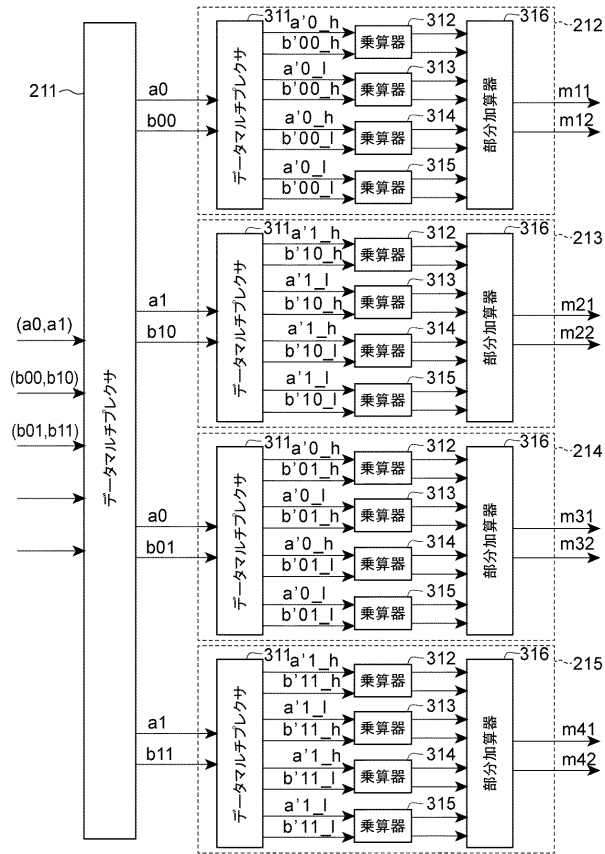
【図 15】



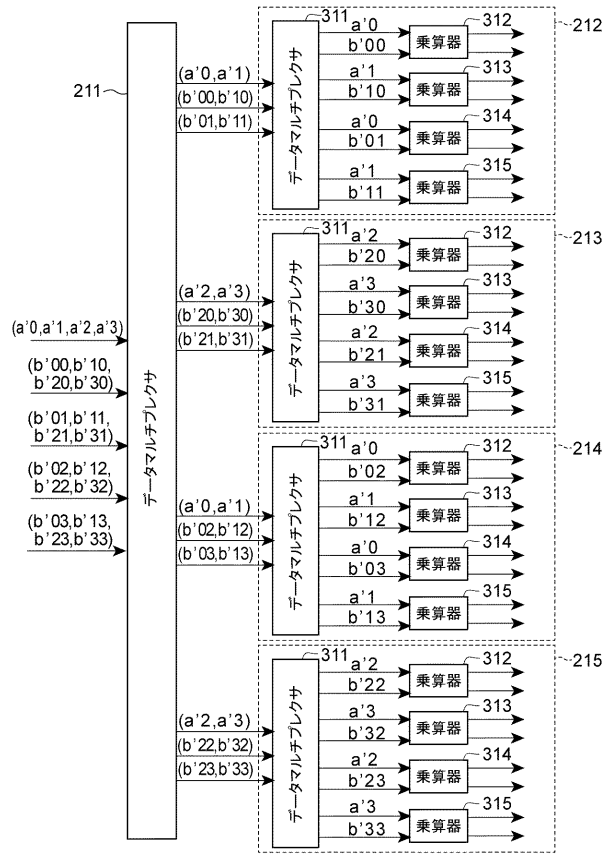
【図 16】



【図 17】



【図 18】



フロントページの続き

(74)代理人 100171583

弁理士 梅景 篤

(72)発明者 牧野 淳一郎

埼玉県和光市広沢 2 番 1 号 国立研究開発法人理化学研究所内

(72)発明者 村主 崇行

埼玉県和光市広沢 2 番 1 号 国立研究開発法人理化学研究所内

(72)発明者 坪内 美幸

埼玉県和光市広沢 2 番 1 号 国立研究開発法人理化学研究所内

(72)発明者 名村 健

東京都千代田区大手町 1 丁目 6 番 1 号 大手町ビル 2 階 株式会社 Preferred Networks 内

審査官 桜井 茂行

(56)参考文献 特開平 3 - 2 8 2 9 2 6 (J P , A)

米国特許第 5 5 9 0 3 6 5 (U S , A)

特開平 4 - 8 4 2 1 9 (J P , A)

米国特許第 5 5 1 5 5 2 0 (U S , A)

米国特許第 8 0 5 1 1 2 3 (U S , B 1)

(58)調査した分野(Int.Cl. , D B 名)

G 0 6 F 1 7 / 1 6

G 0 6 F 7 / 4 8 3

G 0 6 F 9 / 3 8

G 0 6 F 1 5 / 1 7 3