

## 演算語長の再構成に基づくビジュアルフィードバック制御用並列プロセッサの構成と評価

### Design and Performance Evaluation of a Parallel Processor for Visual Feedback Control Based on the Reconfiguration of Word-Length

○藤岡 与周      苫米地 宣裕

○Yoshichika Fujioka and Nobuhiro Tomabechi

八戸工業大学

Hachinohe Institute of Technology

キーワード： 微小演算遅れ時間 (small delay time), 知能ロボットシステム (intelligent robot system), 多入力積和演算 (multi-operand multiply-addition), 演算語長 (word-length), 再構成 (reconfiguration)

連絡先： 〒031 青森県八戸市大字妙字大開 88-1八戸工業大学 工学部 電気工学科  
藤岡与周, Tel.: (0178)25-8063, Fax.: (0178)25-1430, E-mail: fujioka@hi-tech.ac.jp

#### 1. まえがき

自律的動作を目指した知能ロボット実現のためには、外界の環境の認識から柔軟なマニピュレータ制御に至るまで多種多様な処理が必要となる。これらの一連の処理は情報の流れが直列的であると共にセンサフィードバックが多いことから、外界環境の変化に高速に応答するためには、個々の処理に対してスループットの向上のみではなくセンサ信号が入力されてから制御出力が計算されるまでの、演算遅れ時間が小さい VLSI プロセッサの開発が重要な問題となる。この種の研究領域はロボットエレクトロニクス 1) ~ 3) と呼ばれており、これまでにいくつかのプロセッサ開発例が報告されている 4) ~ 9)。

本稿では、知能ロボットシステムにおいて必要となる種々の入力数の積和演算の遅れ時間を減少するため、多入力積和演算器の再構成という概

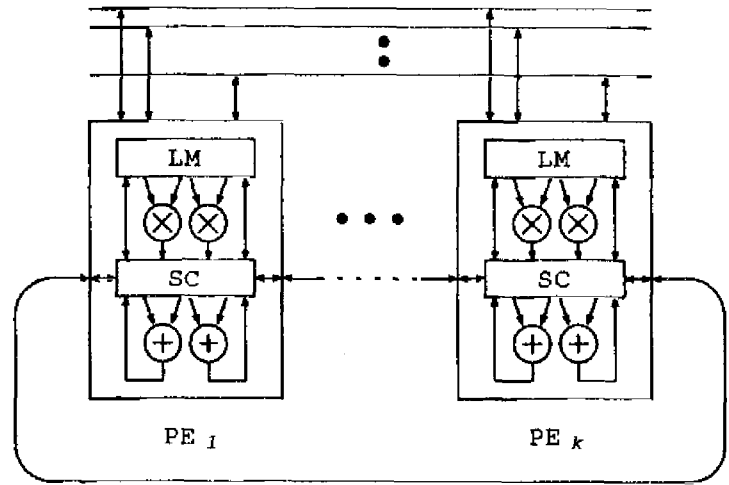
念を、演算語長の再構成という概念に拡張した新しい再構成可能並列プロセッサを提案している。多入力積和演算器の再構成は、複数個の要素プロセッサ (PE) を用いて所望の個数の乗算器を含む多入力積和演算器を、スイッチ回路の切換えによりダイナミックに形成できることを意味している。PE 間通信が、スイッチ回路の切換えによる乗算器や加算器等の各入出力間の直接接続に帰着されるため、PE 間通信のオーバーヘッドが減少し乗算器や加算器などの稼働効率が向上する特長がある。また、ビジュアルフィードバック制御においては 1 ビットから 32 ビット程度までの演算語長が必要となる。このため、所望とする演算語長に応じて乗算器や加算器の構成を変化させ、演算語長に反比例して演算器の個数を増化させることにより、高並列処理を可能としている。

知能ロボットのビジュアルフィードバック制御

における代表的な処理である、画像2値化、重心計算、マニピュレータ座標逆変換、マニピュレータ動的制御について評価を行なった結果、提案する再構成可能並列プロセッサを用いることにより、デジタル信号処理プロセッサ (DSP) を用いた並列プロセッサと比較して演算遅れ時間を約 1/6 に減少できることを明らかにしている。

## 2. 再構成可能並列プロセッサアーキテクチャ

計算量の多いデジタル制御には、複数の DSP からなる並列プロセッサが多く用いられている。しかし、DSP 間のデータ転送のためにハンドシェイクなどのソフトウェア手続きが必要であり、通信のオーバーヘッドが大きいため、演算遅れ時間を減少することが困難であるという問題がある。デジタル制御のための種々の多入力積和演算を高並列に実行し演算遅れ時間を減少するため、図 1 に示す再構成可能並列アーキテクチャを提案している<sup>7,8)</sup>。各 PE は 2つの乗算器と加算器、ローカルメモリ、スイッチ回路を備えている。このスイッチ回路は、乗算器と加算器間の直接接続を切換えて、所望の個数の乗算器を含む多入力積和演算器を再構成するためのものであり、リング状に結合されている。また、再構成された PE 間のデータ転送を容易とするため、各 PE は多重バスにより結合されている。一例として、4 個の乗算器を含む多入力積和演算器を再構成する場合を図 2 に示す。このように、スイッチ回路の接続を切替えて多入力積和演算器を再構成することにより、所望とする個数の乗算器を含む多入力積和演算器を備えた要素プロセッサ PE' を新たに複数の PE で再構成し、多入力積和演算をソフトウェアによる通信手順なしに高速に行うことができる。また、再構成された多入力積和演算器は、乗算器と加算器の入出力間が直接接続されているため、多入力積和演算に伴う PE 間通信は積和演算と同時に進行される。従って、PE 間通信



⊗: 乗算器 LM: ローカルメモリ PE: 要素プロセッサ  
⊕: 加算器 SC: スイッチ回路

Fig. 1 再構成可能並列アーキテクチャ

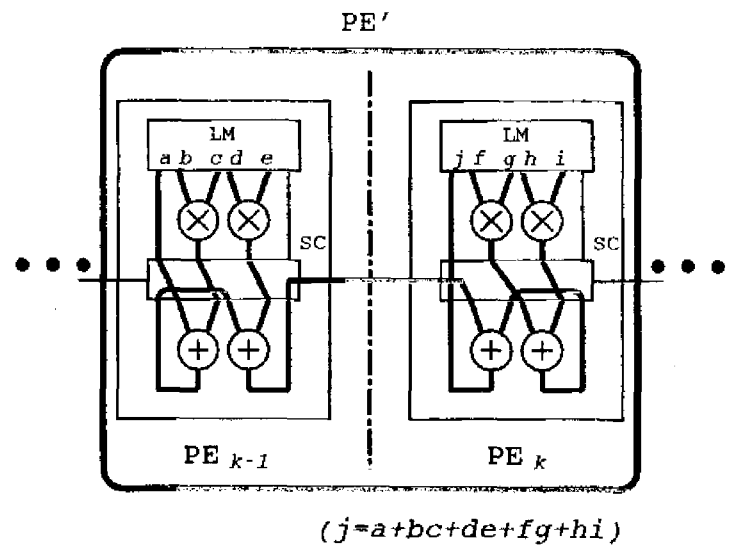


Fig. 2 多入力積和演算器の再構成

のオーバーヘッドが大幅に減少し、乗算器や加算器の稼働効率が向上するため、知能ロボット制御に必要な種々の多入力積和演算の遅れ時間を減少できる。

一例として、次式により表される積和演算の実行は以下ようになる。

$$c_i = \sum_{j=1}^4 a_{ij} b_j \quad (i = 1, \dots, 3) \quad (1)$$

再構成可能並列プロセッサにより並列性が最大となるように演算をおこなうためには、6 個の PE が必要となる。式 (1) で表される多入力積和演

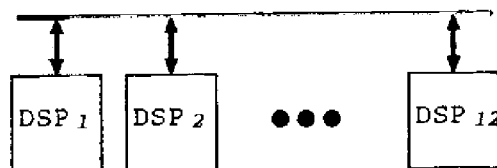


Fig. 3 DSPにより構成された並列プロセッサ

算器を備えた PE が、各プログラムステップ毎の動的なスイッチ回路の切換えにより再構成されるため、並列処理に伴う PE 間データ転送をソフトウェア手続きにより行なう必要がない。ここで、乗算器の演算時間を 1 クロックサイクルと定義すると、加算器の演算時間が乗算器の演算時間と比較して十分小さい場合には、乗算を 1 クロックサイクル、また乗算結果の加算を 1 クロックサイクルの合計 2 クロックサイクルで実行できる。

計算量の多いデジタル制御には、図 3 に示すような複数個の DSP により構成された並列プロセッサが用いられることが多い。多くの DSP は、乗算器、加算器、パラレル入出力ポートを各 1 つずつしか備えていないため、各 DSP が単一共有バス結合された構造となっている。式 (1) で表される積和演算を並列性が最大となるように実行するためには 12 個の DSP が必要となるが、合計 9 個の中間結果の DSP 間通信を、単一共有バスにより行なう必要がある。1 個のデータを PE 間で転送するためには、少なくとも 1 クロックサイクル必要であるとともに、単一共有バスで各 DSP が結合されているため同時に 1 つのデータしか転送することができない。従って、式 (1) で表される積和演算を行うためには、少なくとも 12 クロックサイクル必要となる。提案する再構成可能並列プロセッサを用いる場合と比較すると、演算遅れ時間は 6 倍以上となる。このように DSP 間の通信オーバーヘッドが大きいため、できるだけ DSP 間データ転送回数が少なくなるように並列処理を行なうことが多い。DSP 内部で乗算と加算をオーバーラップ処理可能である場合、式 (1)

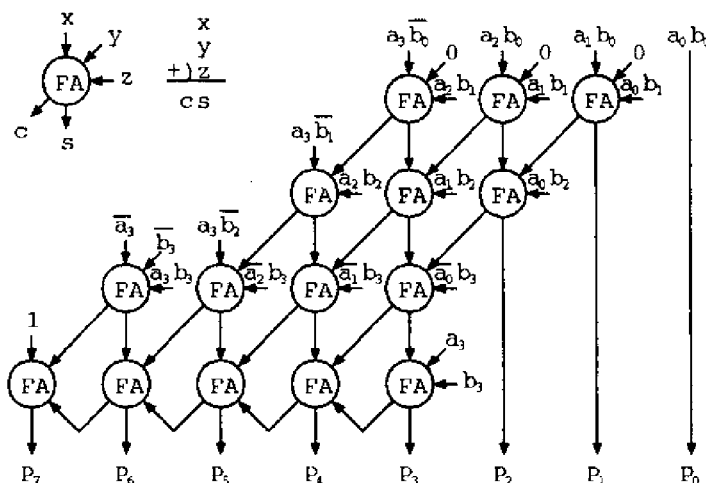


Fig. 4 Baugh-Wooley の 2 の補数配列形乗算器

により表される多入力演算は 3 個の DSP を用いて 5 クロックサイクルで実行される。データ転送回数が少なくなるように並列処理を行なう場合にはしばしば並列性が犠牲となるため、再構成可能並列プロセッサと比較すると、演算遅れ時間は 2.5 倍となる。従って、再構成可能並列プロセッサを用いることにより、複数個の DSP により構成された並列プロセッサを用いる場合と比べて演算遅れ時間を大幅に減少できることが明らかである。

### 3. 演算語長の再構成

知能ロボットのビジュアルフィードバック制御においては、画像処理からマニピュレータ制御まで種々の処理が必要となる。これらの処理では多くの多入力積和演算が必要となるとともに、処理により演算語長が異なる。従来の再構成可能並列プロセッサは、演算語長を変化させることができなかったため、短い演算語長で十分な場合でも長い演算語長の乗算器や加算器を用いて処理を行なう必要があった。これに対し、必要となる演算語長に応じて乗算器や加算器を分割し、演算語長が短い場合にはそれに反比例して演算器の個数を増化させることができれば、ハードウェアの利用効率が向上するとともに高並列処理が可能となる。そこで、演算語長の再構成が可能な乗算器と加算器のアーキテクチャを提案する。この場合、乗算器

や加算器の他にローカルメモリやスイッチ回路も演算語長の再構成に対応させる必要がある。しかし、メモリやスイッチ回路は演算器のようなビット間の依存関係がないため、基本的に従来と同様のアーキテクチャとすることが可能である。クロスバススイッチなどによる任意のシフト機能を備えたスイッチ回路を追加することにより、メモリやスイッチ回路の効率的利用が可能となる。

### 3.1 乗算器

乗算器は、基本的に部分積生成と多入力加算により実現される。一例として、図4に4ビット×4ビットのBaugh-Wooleyの2の補数配列形乗算器の構成を示す。2の補数乗算器には、図4に示すように部分積に入力データ  $a, b$  の各ビット信号  $a_i$  や  $b_j$  の否定である  $\overline{a_i}$  や  $\overline{b_j}$  を使用することにより、単一種類の全加算器のみで多入力加算を行なう方法と、Pezarisの配列形乗算器などのように、部分積生成には  $\overline{a_i}$  や  $\overline{b_j}$  を用いない代わりに、正論理や負論理を組み合わせたいくつかの種類 of 全加算器を用いて多入力加算を行なう方法とに大きく分けることができる。Pezarisの配列形乗算器などを用いて演算語長を再構成する場合には、所望とする演算語長に応じて全加算器の構造も変化させる必要がある。乗算時間に占める多入力加算時間の割合が部分積生成時間に比べて大きいことを考慮すると、Pezarisの配列形乗算器などに基づくアーキテクチャでは全加算器の乗算時間が大幅に増化する可能性が高い。そこで、加算器の構造を変化させる必要のない Baugh-Wooley の配列形乗算器に基づいて、演算語長を再構成可能な乗算器を構成することとする。なお、所望とする演算語長に応じて部分積を生成する本方式は、加算順序を変化させても演算結果が変化しないため、高速乗算器に利用される Wallace トリーなどの概念の適用が容易と考えられるが、ここでは一例として配列形乗算器を採用している。

演算語長を再構成可能な8ビット×8ビット2の補数乗算器を図5に示す。この乗算器は、1個の8ビット×8ビット乗算器、2個の4ビット×4ビット乗算器、乗算器を2個と加算器を1個備える1個の4ビット多入力積和演算器の3種類の演算器を再構成可能である。このように、演算語長が1/2となると、乗算器の個数が2倍になることが、演算語長を再構成可能な乗算器の特長である。

図5の乗算器を用いた2個の4ビット×4ビット乗算器の再構成を図6に示す。8ビット入力  $a$  と  $b$  の下位4ビットの積は乗算結果の下位8ビットに、 $a$  と  $b$  の上位4ビットの積は乗算結果の上位8ビットにそれぞれ出力される。この場合、 $a$  の上位4ビットと  $b$  の下位4ビットの部分積と、 $a$  の下位4ビットと  $b$  の上位4ビットの部分積はともに0になるように部分積を生成している。また、2個の乗算器と1個の加算器を備えた1個の4ビット多入力積和演算器の再構成を図7に示す。2個の4ビット×4ビット乗算器を再構成する場合と異なり、 $a$  の上位4ビットと  $b$  の下位4ビットの積と  $a$  の下位4ビットと  $b$  の上位4ビットの積の和を演算可能である。この場合、 $a$  と  $b$  の上位4ビットの部分積と下位4ビットの部分積はともに0となるように部分積を生成している。

このように、演算語長を再構成する場合には4ビット入力データを8ビットの上位4ビットかまたは下位4ビットのいずれかにシフトする必要がある。このため、乗算器の入出力にそれぞれクロスバススイッチなどによるシフトを備える必要がある。また、所望とする演算語長や多入力積和演算の種類に応じて部分積を生成する回路を必要とするため、演算語長を再構成しない場合と比較して部分積生成の遅れ時間が増大すると考えられる。しかし、乗算器全体の遅れ時間に占める部分積生成時間の割合は、部分積の多入力加算の遅れ時間と比較すると十分小さい。したがって、演算語長

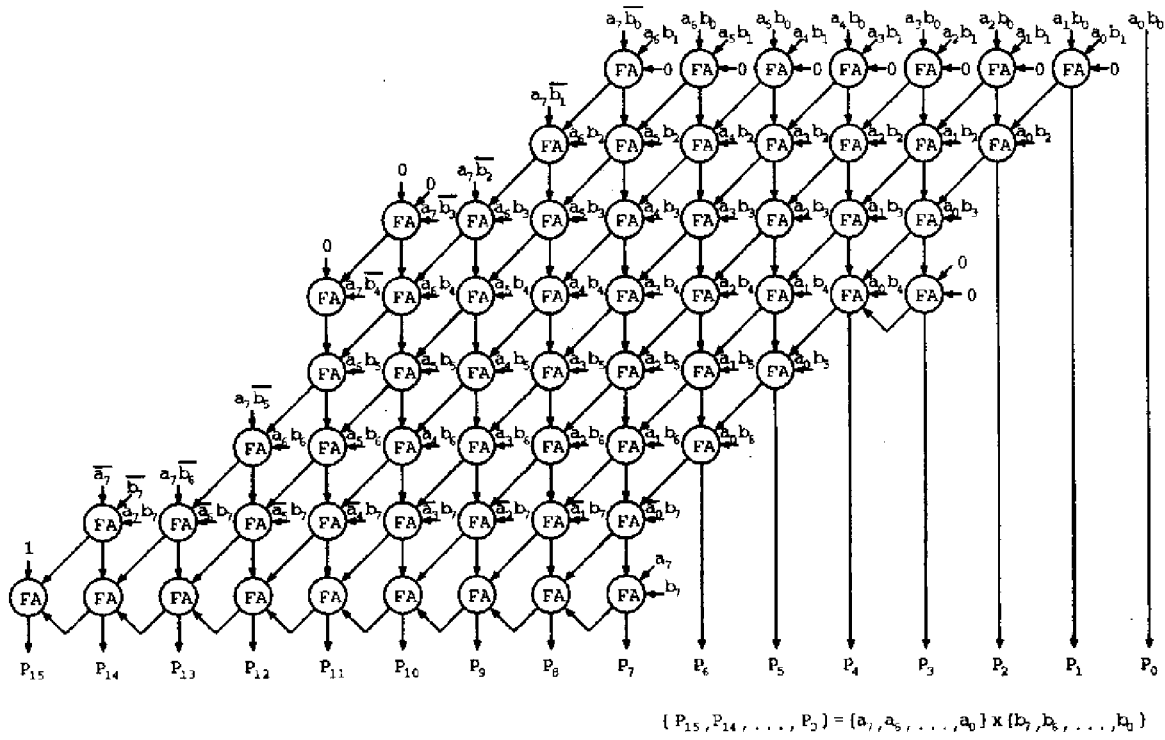


Fig. 5 演算語長を再構成可能な2の補数乗算器

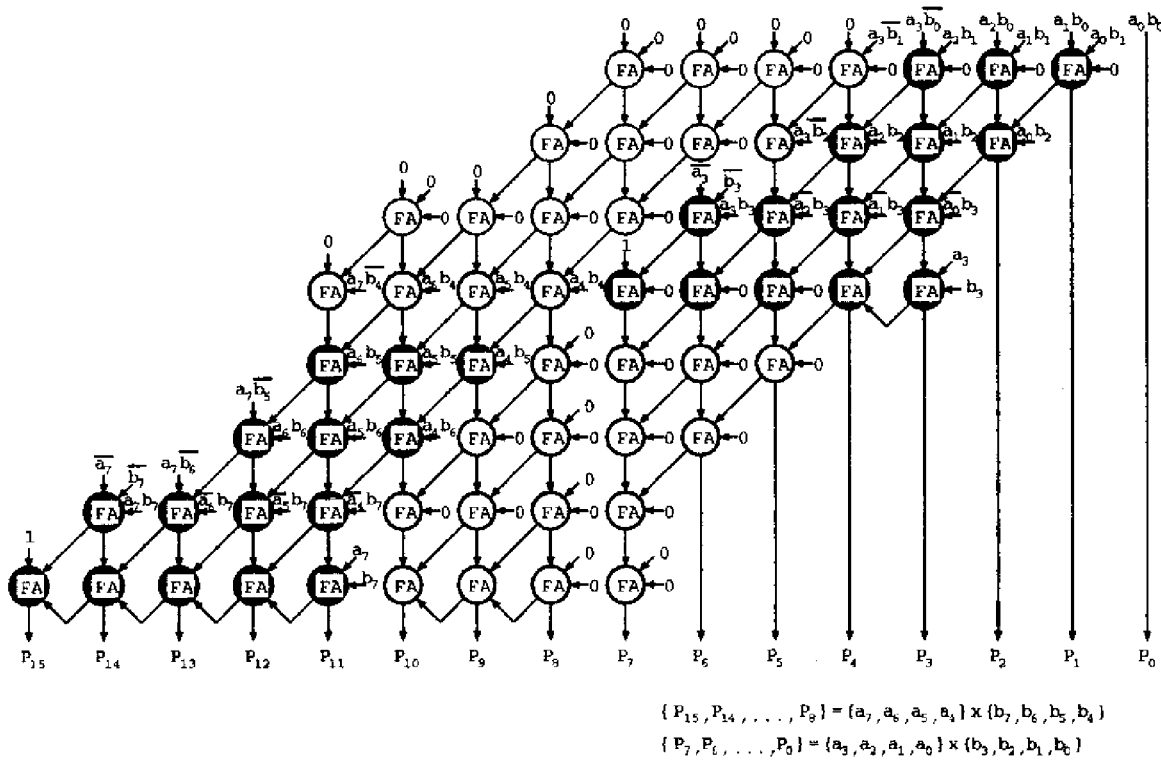


Fig. 6 2個の4ビット×4ビット乗算器の再構成

を再構成可能とすることによる乗算時間の増加はそれほど問題にならないと考えられる。

32ビット×32ビット乗算器の演算語長を再構成する場合の演算器の種類と個数を表1に示す。演算語長に反比例して乗算器の個数が増加す

るのみならず、乗算結果を個別に出力したり、あるいはその和を出力することができる。演算語長を再構成可能な乗算器の特長である。この結果、所望とする演算語長と入力数の多入力積和演算器を容易に再構成可能である。

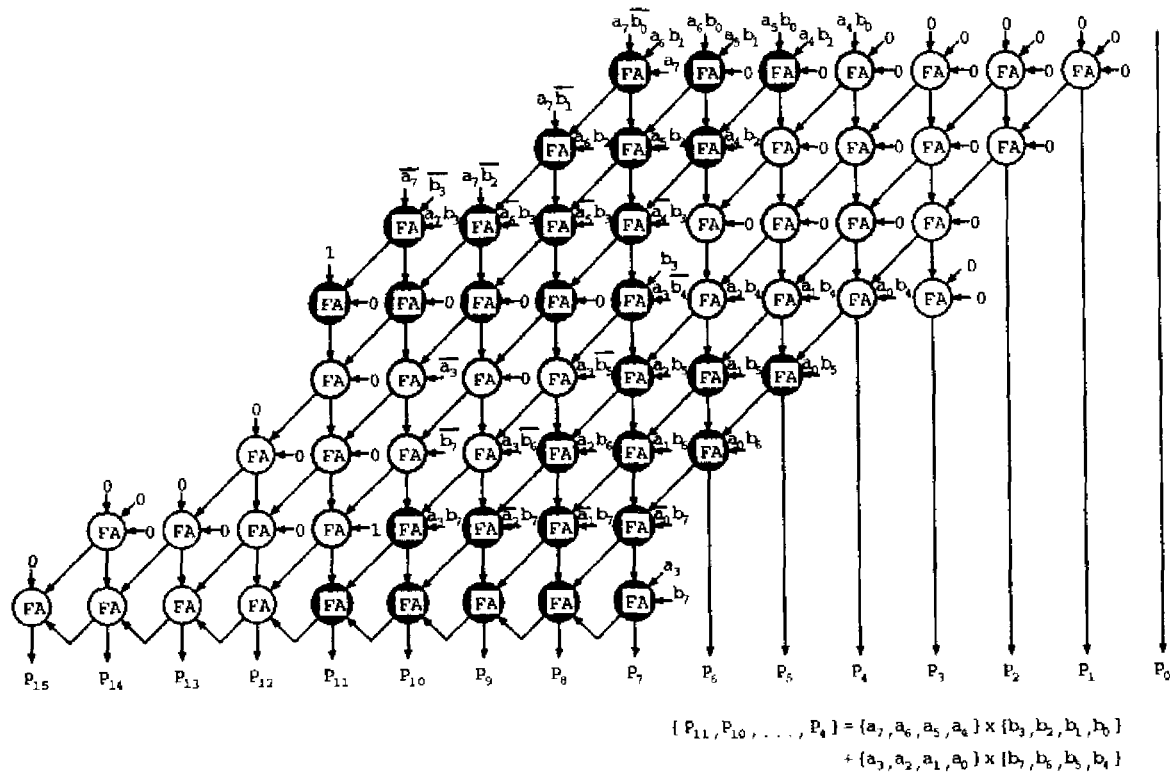


Fig. 7 2個の乗算器と1個の加算器を備えた多入力積和演算器の再構成

Table 1 32ビット×32ビット乗算器の演算語長の再構成

演算語長	演算器の種類	個数
32ビット	乗算器	1
16ビット	乗算器	2
16ビット	2個の乗算器と1個の加算器による多入力積和演算器	1
8ビット	乗算器	4
8ビット	2個の乗算器と1個の加算器による多入力積和演算器	2
8ビット	4個の乗算器と3個の加算器による多入力積和演算器	1
4ビット	乗算器	8
4ビット	2個の乗算器と1個の加算器による多入力積和演算器	4
4ビット	4個の乗算器と3個の加算器による多入力積和演算器	2
4ビット	8個の乗算器と7個の加算器による多入力積和演算器	1

### 3.2 加算器

加算器の演算語長を再構成する最も簡単な方法は、リップルキャリ加算器を用いる方法であるが、演算遅れ時間が非常に大きい。これに対し、高速加算器であり、かつ演算語長を再構成可能な加算器として、バイナリルックアヘッドキャリ (BLC) 加算器があげられる。4ビット BLC 加算器の構成を図8に示す。図8に示すように、BLC 加算器においては、まず入力データの1ビット目と2ビット目、3ビット目と4ビット目の和の中間結果をそれぞれ並列に計算したのち、それらの結果に基づき4ビット分の加算結果を得ている。従って、これらの2ビットデータの和の中間結果を直接利用すると、入力データの下位2ビットの和と上位2ビットの和を独立に得ることが容易である。同様に、入力データの各ビット毎の和もまた容易に得ることが可能と考えられる。このように、BLC 加算器において演算語長を再構成するためには、中間結果を最終段のモジュール S0 に入力する必要がある。

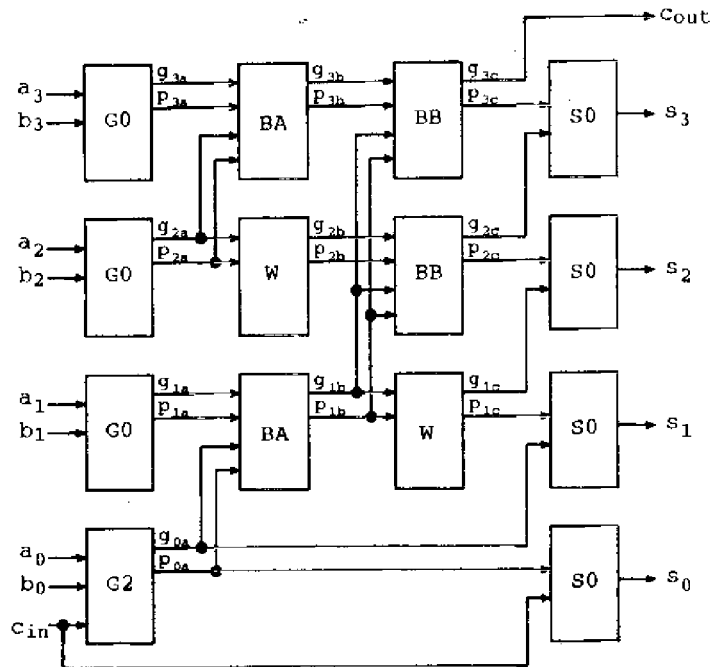


Fig. 8 バイナリルックアヘッドキャリ加算器

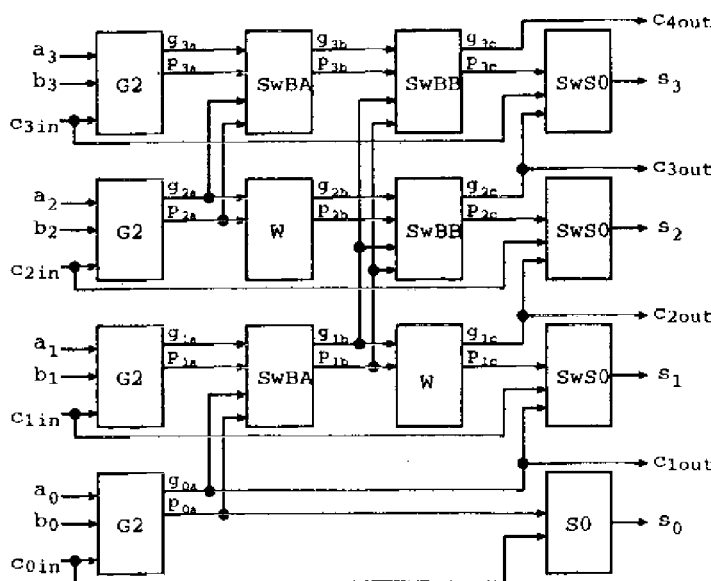


Fig. 9 演算語長を再構成可能な BLC 加算器

演算語長を再構成可能な BLC 加算器の構成を図 9 に示す。1 ビット毎に加算結果を得る場合のため、キャリーの入出力はともに 4 ビットずつ必要となる。また、中間結果を最終段のモジュールに入力するため、モジュール SwBA と SwBB にはバイパス回路が追加されている。さらに、最終段のモジュールには、キャリー入力を選択するためのスイッチが備えられている。これにより、1

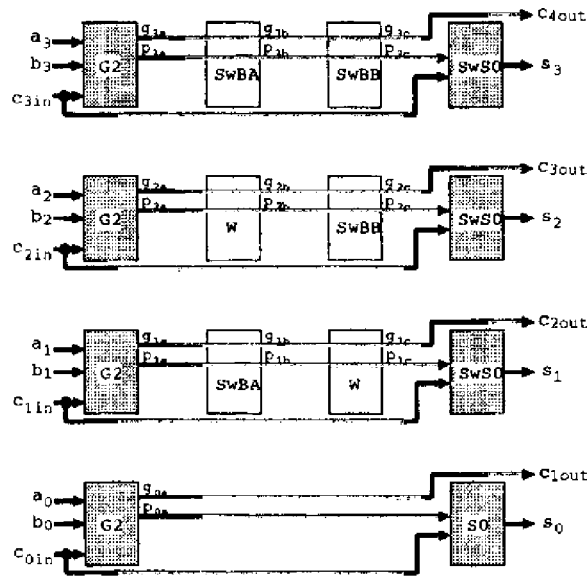


Fig. 10 1 ビット加算器の再構成

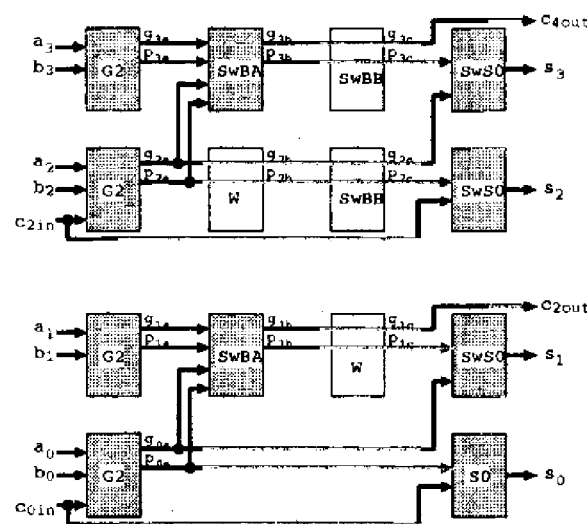


Fig. 11 2 ビット加算器の再構成

個の 4 ビット加算器、2 個の 2 ビット加算器、4 個の 1 ビット加算器（全加算器）を再構成可能である。

図 9 の BLC 加算器による 1 ビット加算、2 ビット加算、4 ビット加算をそれぞれ図 10、図 11、図 12 に示す。中間モジュールをバイパスすることにより、種々の演算語長を再構成可能である。また、3 2 ビット BLC 加算器の演算語長を再構成する場合の演算器の種類と個数を表 2 に示す。演算語長を再構成可能な乗算器と同様に、演算語長に反比例して加算器の個数が増加するため、演算

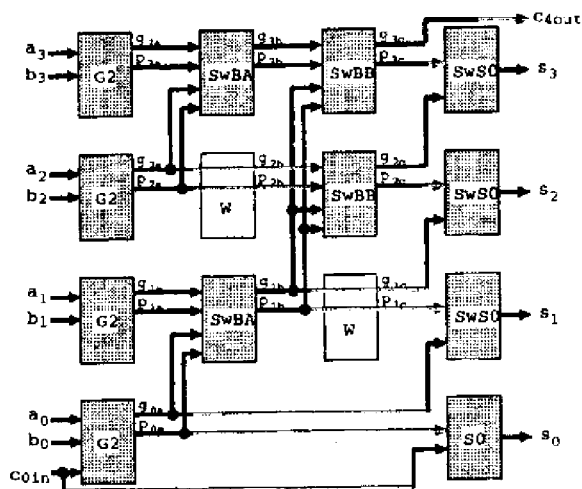


Fig. 12 4ビット加算器の再構成

Table 2 32ビットBLC加算器の演算語長の再構成

演算語長	演算器の種類	個数
32ビット	加算器	1
16ビット	加算器	2
8ビット	加算器	4
4ビット	加算器	8
2ビット	加算器	16
1ビット	加算器	32

語長が短い場合には高並列処理が可能である。

#### 4. ビジュアルフィードバック制御への応用

知能ロボットのビジュアルフィードバック制御には、カメラからの画像取り込みからマニピュレータ制御に至るまで種々の処理が必要となる。ここでは、簡単な例題として、画像2値化、2値化画像における対象物体の重心計算、対象物体の3次元座標からマニピュレータの各関節変位を計算する座標逆変換、マニピュレータの動特性を補正する動的制御を考えることにする。ただし、極めて限定された作業環境を設定し、重心から対象物体の3次元座標が容易に得られると仮定している。

表3に示すように、8ビットから32ビット程度の演算語長が各処理において必要となる。さらに、重心計算には2値化画像の対象物体の面積計

Table 3 ビジュアルフィードバック制御における各処理の主な演算語長

処理の種類	演算語長
画像2値化	8ビット
重心計算	27ビット
座標逆変換	32ビット
動的制御	32ビット

Table 4 性能評価

		(μsec)	
	再構成可能	複数個の DSP	
画像2値化	273	1092	
重心計算	316	2186	
座標逆変換	11	143	
動的制御	12	34	
合計	612	3458	

算が含まれる。2値化画像では各画素は1ビットデータとして表現できるため、面積計算は1ビットデータの加算により実現される。512画素×512画素の2値化画像での面積計算には最大で18ビットの演算語長が必要となるが、最初から18ビット程度の演算語長を必ずしも必要としない。例えば、1ビットデータの加算を並列に実行して2ビットの中間結果をまず求めた後、2ビットデータの加算を並列に実行して4ビットの中間結果を求めることができる。同様にして8ビット、16ビット、32ビットと演算語長を段階的に増化させることが可能である。そこで、BLC加算器の演算語長を1ビットから必要に応じて増化させることにより、1ビットデータの加算時には32倍の個数の加算器を用いた高並列処理が可能となる。

6個のPEを5本の多重バスで結合した再構成可能並列プロセッサと、12個のDSPを単一共有バス結合した並列プロセッサによるビジュアルフィードバック制御の演算遅れ時間の比較を表4に示す。マニピュレータの座標逆変換と動的制御の演算性能は、従来の再構成可能並列プロセッサ



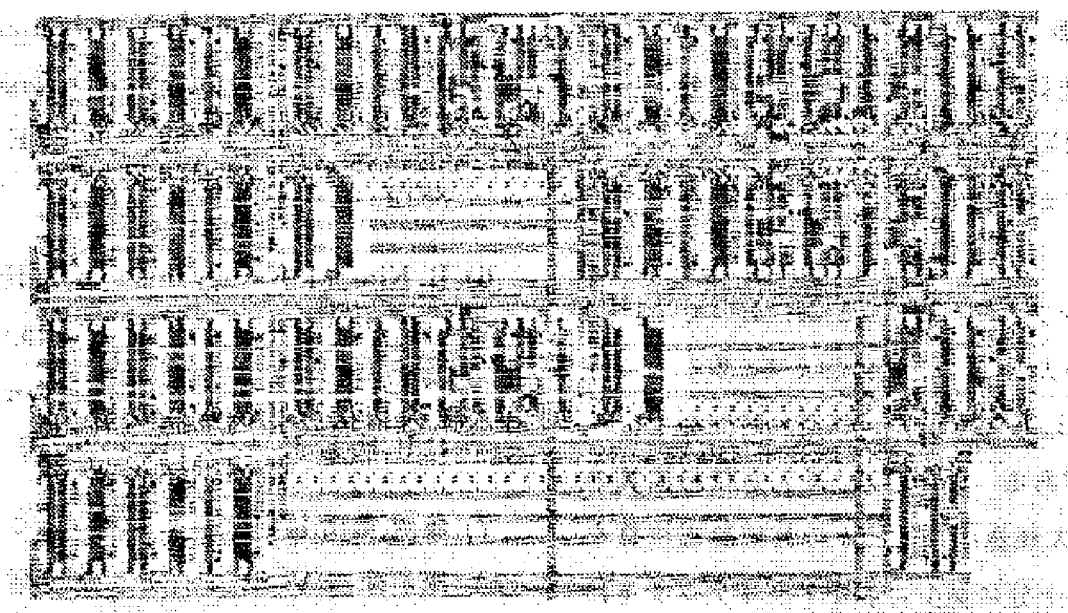


Fig. 13 演算語長を再構成可能な4ビットBLC加算器のチップレイアウト

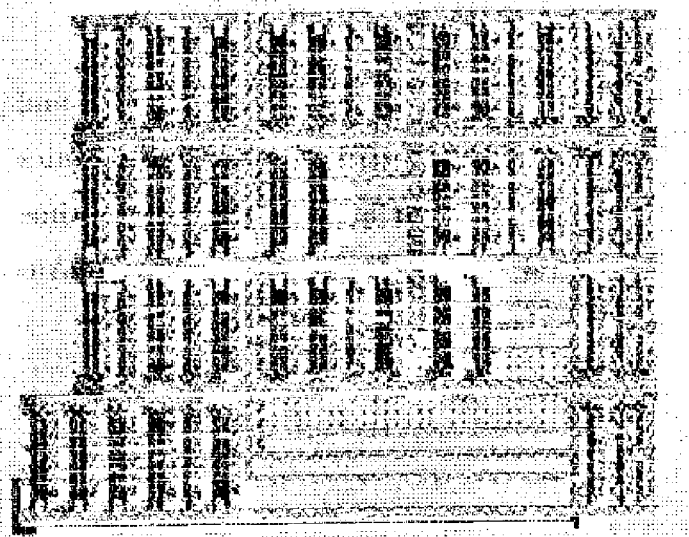


Fig. 14 従来の4ビットBLC加算器のチップレイアウト

によるものと等しい。また、種々の演算語長を必要とする画像2値化や重心計算については、従来の再構成可能並列プロセッサの性能に加えて、演算語長を再構成可能としたことによる性能向上が大きく寄与している。

このように、演算語長を再構成可能とすることにより、演算遅れ時間を約1/6程度に減少することができる。しかし、演算語長を再構成可能とした結果ハードウェア量が膨大になるとすれば、

総合的な演算性能が向上したとはいえない。そこで、4ビットBLC加算器を例にとり、演算語長の再構成を行なう場合と従来の場合とのチップレイアウト評価を1 $\mu$ m CMOS設計ルールに基づいて行なった結果をそれぞれ図13と図14に示す。この結果、演算語長の再構成を行なう場合とそうでない場合とのチップ面積はそれぞれ53240 $\mu$ m<sup>2</sup>と31501 $\mu$ m<sup>2</sup>となり、演算語長の再構成を行なう場合従来と比べて約1.6倍のチップ面積を必要とすることが明らかとなった。ビジュアルフィードバック制御の演算遅れ時間が約1/6となるのに対し、演算器のハードウェア増加量が約1.6倍であるため、総合性能という観点からみても演算語長を再構成可能な並列プロセッサアーキテクチャが優れているといえる。以上より、演算語長を再構成可能な並列プロセッサは、ビジュアルフィードバック制御などのように種々の演算語長を必要とし、かつ高速応答性が要求される知能集積システムの実現に有用である。

## 5. むすび

知能ロボットのビジュアルフィードバック制御を複雑な環境で実現するためには、非常に多くの

処理が必要となる。再構成可能並列プロセッサは、PE 間データ転送のオーバーヘッドが小さいため、DSP による並列処理と比較すると、多くの PE を用いた高並列処理が容易である。さらに、演算語長を再構成することにより、より高並列な処理が可能となる。

ビジュアルフィードバック制御においては、種々の入力数の多入力積和演算に加えて、多くの条件分岐を含む処理などもまた演算遅れ時間を十分減少する必要がある。このため、多入力積和演算のみならず、最大値選択など種々の算術論理演算をも遅れ時間を十分に減少可能なアーキテクチャに拡張する必要がある。

また、種々の知能ロボット制御に要求される性能を達成するためには、PE 数やネットワーク構造を適切に選択するとともに、演算器稼働効率が高く演算遅れ時間が小さい並列処理を実現するソフトウェアを作成するための開発環境が重要である。このため、ロボット制御などに必要となる種々の入力数の多入力算術演算を効率よく実行するためのハイレベルシンセシスやスケジューリングアルゴリズムの開発が今後重要な課題となる。

## 参考文献

- 1) 亀山,樋口: “ロボットと VLSI コンピュータ”, 日本ロボット学会誌, 6, 4, pp. 332-338 (1988).
- 2) 樋口: “ロボットエレクトロニクス”, 信学誌, 71, 5, pp. 502-504 (1988).
- 3) 今井: “ASIC 技術の基礎と応用”, 電子情報通信学会, pp. 118-140 (1994).
- 4) M. Kameyama, T. Matsumoto, H. Egami and T. Higuchi: “A special-purpose LSI for inverse kinematics computation,” Trans. IEICE, E 74, 11, pp. 3829-3837 (1991).
- 5) S. Kittichaikoonkit, M. Kameyama and T. Higuchi: “Design of a matrix multiply-addition VLSI processor for robot inverse dynamics computation,” Trans. IEICE, E 74, 11, pp. 3819-3828 (1991).
- 6) B. Kim, M. Kameyama and T. Higuchi: “Parallel VLSI processors for robotics using multiple bus interconnection Networks,” Trans. IEICE, E 75-A, 6, pp. 712-719 (1992).
- 7) 藤岡, 亀山, 樋口: “冗長マニピュレータ制御用座標変換 VLSI プロセッサ”, 信学論, J 75-D-I, 10, pp. 909-916 (1992).
- 8) 藤岡, 亀山, 苦米地: “再構成可能並列プロセッサと知能ロボット制御への応用”, 日本ロボット学会誌, 13, 6, pp. 112-119 (1995).
- 9) P. Sadayappan, Y.L.C. Ling, K.W. Olson and D.E. Oriin: “A restructurable VLSI robotics vector processor architecture for real-time control,” IEEE Trans. on Robotics and Automation, 5, 5, pp. 583-599 (1989).