

## 専用プロセッサの動向\*

—アレイ・プロセッサの現状と課題—

吉岡 義朗\*\*

### 1. はじめに

ひとことアレイ・プロセッサといえば、ILLIAC IV が主役であったが、最近では、商用機種としても、CDC の STAR をはじめ、各種のアレイ・プロセッサが登場しはじめている。一方、事務計算分野では、ソート・プロセッサなども出現しており、計算機の多様化・専用化のさざしをみせている。

そこで、本文では、アレイ・プロセッサのように、特定のデータ処理形態に焦点を合わせて開発された専用プロセッサについて、その出現の背景をさぐってみるとともに、専用化のにない役であるアレイ・プロセッサについて、現状を概観してみることにする。

### 2. 出現の背景

Gain-Band 積が一定であるという法則は、計算機の場合にもあてはまっている。ひろいアプリケーションに対して高性能を有する汎用プロセッサを実現するのはむずかしくても、アプリケーションを限れば、同じコストで高い性能を出すことが可能である (Fig. 1)。

汎用か専用かは、古くからの議論であるが、最近商用計算機として、特定のデータ処理形式に焦点を合わせた専用プロセッサが次々と出現しはじめており、そ

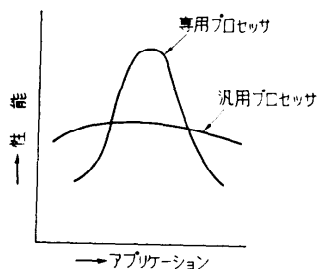


Fig. 1 Gain-Band Products of Processors.

\* A View of Special Purpose Processors—The States of Art of Array Processors—, by Yoshiro Yoshioka (Fujitsu Limited)

\*\* 富士通株式会社

れらを生みだす土壤が、ようやく育ってきたようである。その背景としては、

- (1) 計算機の市場が大きくなり、相対的に専用プロセッサの需要が大きくなってきたこと。
- (2) 汎用超高速プロセッサは、能力の限界に近づきつつあるが、専用プロセッサでは、この壁を突破できること。
- (3) 汎用プロセッサの方式は、ある程度方向が定まってきており、専用プロセッサを開発する技術的指向性が、生じつつあること。
- (4) 部品、回路技術が進歩した結果、素子のコストが低下し、ハードの専用化が行ないやすくなったこと。

が考えられる。

### 3. 専用プロセッサの種類

専用プロセッサの開発が成立するためには、

- (1) 専用といえども、専用処理の内容が、できるだけ多くのアプリケーションに適用可能であること。いいかえれば、ある程度の市場が見込めること。
- (2) 専用化することによって、専用化に見合うだけのコスト・パフォーマンスを上げられるような、技術的・方式的な解決策が存在すること。

が必要である。

現時点で、この条件を満たす処理内容として、科学計算分野では、アレイ処理が、事務計算分野では、ソート処理があり、それらに対して専用プロセッサが提供されはじめている。以下、専用プロセッサのにない役であるアレイ・プロセッサについて、現状を概観してみることにする。

### 4. アレイ・プロセッサ

#### 4.1 アレイ演算

ベクトルやマトリックスなどのようなアレイ (配列) 構造をしたデータの処理は、アレイ演算と呼ばれ、ア

レイ・プロセッサは、このアレイ演算を専用に実行すべく開発されたプロセッサである。

この演算の特徴は、

- (1) 同一の処理を多数のデータに施すこと。
- (2) データの構造が整っていること。

であり、並列処理に適した演算である。

アレイ・プロセッサは、この特徴を生かして、高度の並列処理、あるいはオーバラップ処理により高性能を実現しているのである。

計算機で数値計算を行なう場合、取扱いの便利さからデータをアレイ構造にして扱うことが多いので、アレイ・プロセッサの適用できる計算は非常に多い。

たとえば、科学計算問題の 75% 以上の問題に含まれているといわれる線型計算、偏微分方程式、数値積分、多項式計算などが、アレイ演算に導かれる。

アレイ・プロセッサの応用分野としては、

- (1) Matrix 演算一般
- (2) Matrix の固有値問題
- (3) 線型計画法
- (4) 気象計算
- (5) フィルタ設計
- (6) 構造解析
- (7) フーリエ解析

など多岐にわたっている。

これらの分野では、通常、アレイ処理の時間が、全体の処理時間の中で大きな割合を占めている。たとえば、マトリックスの計算では、計算時間は、マトリックスの次数  $n$  の 3 乗に比例して増大するし、気象予報計算では、5 日先の予報をするために、350 MIPS (Million Instructions Per Second) という膨大な超高速計算を必要とするといわれているほどである。応用分野からの、このような要求と、技術的な裏付けとが相まって、アレイ・プロセッサの登場をうながしたのであろう。

#### 4.2 アレイ・プロセッサの構成

アレイ・プロセッサとして超高性能を誇る ILLIAC IV は、256 個のプロセッシング・エレメント (PE, 最近 64 個になったと聞いている。) から成っているが、商用として登場しているのは、すべて単一プロセッサ (1 PE) 形式のものである。

アレイ・プロセッサの構成を、プログラムに見える PE 数とメモリの関係で分類すると、次のようになる。

- (1) プロセッサ・アレイ型
  - a. メモリ個有

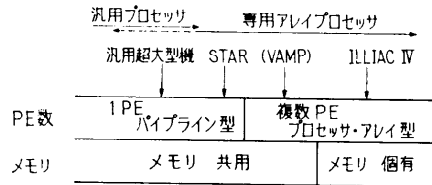


Fig. 2 Types of Array Processors.

#### b. メモリ共有

- (2) パイプ・ライン型

通常の計算機とこれらの関係は、Fig. 2 のようになる。

#### 4.3 プロセッサ・アレイ型

多数のプロセッシング・エレメント (PE) をアレイ状に配列した構成をとり、PE の並列動作によって高性能を実現するものである。アレイ演算に適しているゆえんは、PE がアレイ構成になっているからであって、アレイ処理用の特別な命令があるわけではない。この点、パイプ・ライン型とは、根本的に異なっている。

プロセッサ・アレイ型の思想は、SOLOMON COMPUTER にはじまり、ILLIAC IV で実現された<sup>1)~3)</sup>。ILLIAC IV では 256 個の PE に個有のメモリを持たせた形式をとっている。

並列処理による高速性は、PE の数と各 PE の性能に依存しているばかりでなく、メモリのデータ供給能力に大きく依存している。したがって、各 PE が個有のメモリを持っている方式は、直接的で、かつ最も能力を発揮できる形式である。ILLIAC IV では、この結果 200 MIPS (Million Instructions Per Second) という性能を発揮できるといわれている。

各 PE に、個有のメモリを持たせる方式は、このような利点を持っているが、次のような弱点をもっている。

- (1) アレイ・エレメントのシフト、回転、転置などに不向きである。
- (2) PE がメモリと一体になっているため、PE だけを、一次元的な処理機構とみなしにくいので、アレイの大きさが PE 数を越えたときの処理が複雑になる。
- (3) 演算の種類によって、並列処理を有効に生かすための、最適メモリ割当てが異なる。

これらの点の解決策として、PE とメモリを分離し、全 PE がメモリを共有する方式が提案されている (VAMP)<sup>4)</sup>。この場合、メモリは必要なデータ供給能

力を満たすために、多数のバンクに構成され、高度のインタリーブを行なうことになる。

この方式によれば、複数の PE を処理機構とみなして、一次元的な扱いができるので、上記の欠点はなくなるが、逆にメモリ・インタフェース部に多くの負担がかかることになる。

4.4 パイプ・ライン型

パイプ・ライン型は、プログラムに見えるプロセッシング・エレメントの数が一つである点からいって、通常の計算機の延長線上にある。アレイ演算関係のマクロ命令を主体とした（通常の）計算機であるということができる。

内部処理をはやめるためアレイ演算に対して、パイプ・ラインによるオーバラップ処理を行なうのが普通であるので、これをパイプ・ライン型と呼ぶことにする。ただ、高速化の手法は、必ずしも、パイプ・ライン制御に限らないし、また、パイプライン・ユニットを複数個並列に置くこともありうる。

パイプ・ライン制御とは、ベルト・コンベアによる生産ラインと同じ考え方に基づくもので、一つの演算を複数の処理ステップに分割し、各ステップを別々のエレメントの処理に使用して、オーバラップによるスループットの向上をはかるものである。浮動小数点加算器を仮数部のプレシフト段、加算段、ポストノーマライズ段の3段にパイプ・ライン化するのは、簡単なパイプ・ラインの例である<sup>9)</sup>。

パイプ・ライン型は、プロセッサ・アレイ型が必然的に超超大型機になってしまうのに比べて、金物的にそれほど大きくはならないので、商用機向きである。

アレイ演算命令は、一般に三つのアレイ A, B, C を指定して、 $A * B \rightarrow C$  の形で行なわれている (\* はオペレーション)。

一次元アレイ (ベクトル) の指定は、

ベクトルの最初のエレメント・アドレス	B
エレメントの間隔	S
エレメント数	n

によって行なわれ、メモリのアドレス、B, B+S, B+2S, …, B+(n-1)S にあるエレメントが演算の

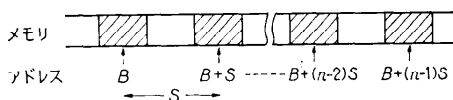


Fig. 3 Specification of a Vector in the Memory.

対象となる (Fig. 3)。エレメント数 n は、通常大きいので (逆に大きくないとアレイ・プロセッサの効果を発揮できないが)、演算はメモリ間で行なわれるのが普通である。

パイプ・ライン型は、アレイ演算を命令で指定するので、プロセッサ・アレイ型に比べて、単に対応するエレメント間の演算だけでなく、アレイ演算に特有な機能を持たせることが容易である。たとえば、

(1) マトリックスの乗算やコンボリューションのような複合演算機能。

(2) スパース・アレイの取扱い機能。

などを持たせることができる。

スパース・アレイとは、零要素を多く含んだアレイのことである。アレイ演算では、アレイの次数 (エレメント数) が高くなるにつれて、零要素が多くなり、非零要素が数パーセント以下であることも多い。

スパース・アレイを演算する場合、零要素を含んだまま扱うのでは、メモリ・スペースが不経済であるうえ、零要素の計算にむだな時間を費やしてしまう。

このため、一般には、非零要素のみをつめて扱っている。つめられた非零要素が、元のアレイのどこに位置するかは、インデックス表またはコントロール・ビット表によって示される (Fig. 4)。

アレイ・プロセッサに、スパース・アレイの処理機能を持たせることによって、単に演算の実行だけでなく、プログラムの記述を含めたひろい意味での、能率化をはかることができる。

パイプ・ライン型は、汎用プロセッサの延長線上に位置するわけであるが、汎用プロセッサに比べて高速化がはかる理由としては、次の点が考えられる。

(1) 命令の読出しが不要であること。

たとえば、 $A_i + B_i \rightarrow C_i (i=1, \dots, n)$  という演算を行なう場合、汎用プロセッサでは、

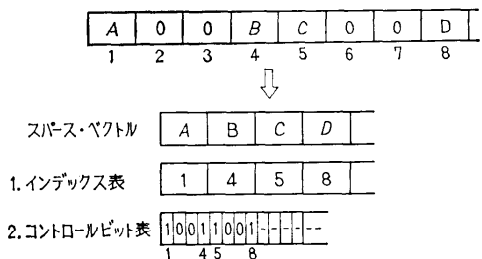
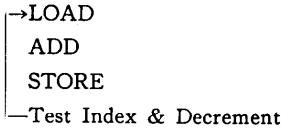


Fig. 4 Sparse Vector.



のように、4ステップのループを必要とするが、アレイ・プロセッサでは、1回の命令読出しだけでよい。ただ、高度の先行制御や、インストラクション・スタックを採用している大型機では、命令の読出しが、実行とオーバーラップするので、この利点は消えてしまう。

(2) ブランチ命令が不要であること。

超大型の汎用プロセッサでも、条件付きブランチ命令の先行制御は、そこでプログラムが二つに分岐するため、複雑でやっかいなものになる。一般には、ブランチ命令のたびに、処理の流れが中断してしまい、等価的に、ブランチに多くの時間がかかってしまう。

(3) アレイ処理に適した、高度のパイプ・ライン化を行なうことができること。

(4) メモリの高いデータ供給能力を有効に利用できること。

アレイ・データは、データ構造が規則的であるから、必要とされる多数のデータをメモリから(メモリへ)並列に読出す(書込む)ことが可能であり、高度にインタリーブされたメモリのデータ供給能力を有効に利用することができる。汎用プロセッサでも、メモリへのアクセス単位は次第に大きくなる傾向にあるが、ある程度以上大きくすると、汎用であるために、利点よりは返って損失が大きくなり、あまり大きくできない。

(5) アレイ処理に個々の機能を盛り込むことができること。

スパース・アレイの処理機能などを持たせることによって、アレイ処理全体としての能率を高めることができる。

このようなアレイ演算の特徴を生かすことによって、汎用プロセッサに比べて、数倍~数十倍の性能を出すことができる。Table 1 に、商用機種または実験機種として発表されているパイプ・ライン型アレイ・プロセッサとその特徴を示す。ミニコンに接続される小規模アレイ・プロセッサから、超大型機まで各種のアレイ・プロセッサが開発されている。

以下、各機種の概要をながめてみることにする。

4.5 STAR 100

Table 1. Announced Array Processors.

機種	ホスト・プロセッサ	接続レベル	ブランチ命令、インデックス命令などの有無
STAR	—	—	有
FACOM AP	FACOM 230-75	processor	"
IBM 2938	IBM 360/44, 65, 75	CH	無
UNIVAC AP	UNIVAC 1106, 1108, 1110	CH	"
VECTOR-1	sigmaz 2	processor	"
RAYTHEON ATP	RAYTHEON 703, 704, 706	CH (I/O)	"
NOVA	—	—	有

STAR とは、SString ARray Processor の略で、CDC がローレンス放射線研究所の特注で開発した超大型計算機である。1971 年秋に納入されるといわれている。

CDC は STAR を汎用プロセッサとして発表した。アレイ・プロセス専用であるという性格が強い。STAR 100 の性能は、IBM 370/195, CDC 7600 の性能 (10 MIPS 程度) をはるかに上回り、50~100 MIPS といわれている。その高性能の原動力は、

- (1) メモリの高いデータ供給能力
- (2) 三つのパイプライン・ユニット

から生まれている。

STAR のメモリは、標準 4 メガバイトの容量を持ち、独立に動作する 32 個のバンクから構成されている。1 個のバンクは、64 バイト/1.6  $\mu$ s のデータ供給能力を持っているので、全部がインタリーブして使用されるから、320 メガ語 (32 ビット)/ $\mu$ s という膨大なデータ供給能力を持っていることになる。

STAR のパイプ・ラインは、1 ステージ 40 ns で、動作する 2 個の浮動小数点演算ユニットと 1 個のストリング演算ユニットから成っていて、メモリ・バンド幅とのマッチングをとっている (Fig. 5)。

STAR の命令セットには、通常のレジスタ演算命令などのほかに、非常に強力なアレイ演算命令が用意されている。アレイ演算命令は、3 アドレス形式で、

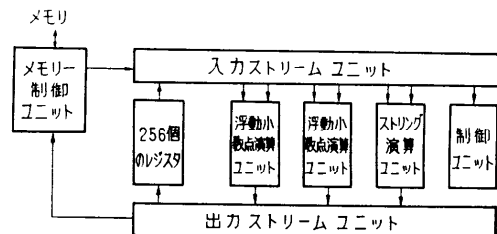


Fig. 5 The CPU Block Diagram of STAR.

32/64ビットの浮動小数点ベクトルを扱うことができ、次の4種類がある。

- (1) ベクトル演算命令。
- (2) スパース・ベクトル演算命令。
- (3) ベクトル間のテスト, 比較命令。
- (4) 最大値/最小値などのサーチ命令。

4.6 FACOM AP

FACOM AP は, FACOM 230-75 をホスト・プロセッサとするアレイ・プロセッサである。FACOM AP は, STAR と同程度の強力なアレイ演算命令を持っているほかに, 特権命令などを除く 75 CPU 命令を備えている。

慣性の大きい AP にとって不得意な, 割込処理, I/O 制御, システム制御などの処理 (あるいは, AP を使用しない JOB の処理) は, ホスト・プロセッサである 75 が分担し, AP は CPU との通信をできるだけ少なくして, アレイ処理に専念する形式をとっている。90 ns のパイプ・ラインを二つ備えており, 20~40 MIPS の性能を持っている。

4.7 IBM 2938<sup>5)</sup>

IBM 2938 アレイ・プロセッサは, 360/44 に接続されるモデル I と, 65, 75 に接続されるモデル II とがあり, システム構成上は, チャンネルと同一レベルに置かれ, ソフト的には一種の I/O とみなされる。

したがって, 2938 は I/O を起動すると同じ START I/O 命令によって, CPU から起動され, メモリー上に用意されている 2938 専用のチャンネル・プログラムを実行する。実行の終了は, I/O 割込みによって CPU に報告される。

2938 は, メイン・メモリー上の三つのアレイ (X, Y, U) に対して, 次の演算を行なう。

$$Y' = (Y) + U * X$$

\*: オペレーション, (Y): 命令で有無を指定する。

命令の種類は, Table 2 に示す 12 種類で, ブランチ命令, インデックス命令などはない。

データの形式は, 16 ビットの固定小数点データと, 32 ビットの浮動小数点データの 2 形式であり, 長精度はない。これは 2938 が主として, 地震波解析を対象にしているためかと思われる。

2938 には, 200 ns のパイプ・ラインのほかに, プログラムからは見えない 32 語 (32 ビット) 分のバッファ・メモリーが, 2 組用意されており, X および Y のバッファとして使われている。

Table 2. The Instruction Set of IBM 2938

	名 称	演 算 内 容
1	Vector Move Convert	$y_i \leftarrow x_i$
2	Vector Floating-Point to Fixed Point Move	---
3	Vector Element by Element Multiplication	$y_i \leftarrow \{y_i\} + u_i \cdot x_i$
4	Vector Element by Element Sum	$y_i \leftarrow u_i + x_i$
5	Scalar Multiplication	$y_i \leftarrow \{y_i\} + u \cdot x_i$
6	Signed Squared Array	$y_i \leftarrow \{y_i\} + x_i \cdot  x_i $
7	Sum of Squares	$y \leftarrow \{y\} + \sum_{i=1}^n x_i \cdot  x_i $
8	Sum of Vector Elements	$y \leftarrow \{y\} + \sum_{i=1}^n x_i$
9	Vector Inner Product	$y \leftarrow \{y\} + \sum_{i=1}^n x_i \cdot x_i$
10	Partial Matrix Multiplication	$y_i \leftarrow \{y_i\} + \sum_{j=1}^n x_j \cdot u_{(i-1)P+j}$
11	Convolving Multiplication	$y_i \leftarrow \{y_i\} + \sum_{j=1}^n u_j \cdot x_{i+j-1}$
12	Convolving Addition	$y_i \leftarrow \{y_i\} + \sum_{j=1}^n  u_j + x_{i+j-1} $

$u, x, y$ : スカラ  
 $u_i, x_i, y_i$ : ベクトル・エレメント  
 $i=1 \sim N, N$  はベクトル  $U, X$  または  $Y$  の全エレメント数  
 $P$ : ベクトル  $Y$  のエレメント数

4.8 UNIVAC AP

UNIVAC AP は, UNIVAC 1106, 1108, 1110 に接続されるアレイ・プロセッサで, IBM 2938 とほとんど同じ仕様である。IBM 2938 と同様に, 地震波の解析を対象にしているようである。250 ns のパイプ・ラインを採用しているとのことである。

4.9 VECTOR-I

VECTOR-I は, カルフォルニア大学 (ディビス) で開発された研究用アレイ・プロセッサで, XDS の Sigma 2 に接続して使用される。

VECTOR-I の接続は, Fig. 6 に示すように, Sigma 2 に, 4K バイトのメモリーを 3 バンク追加し, それに接続される形式をとっている。

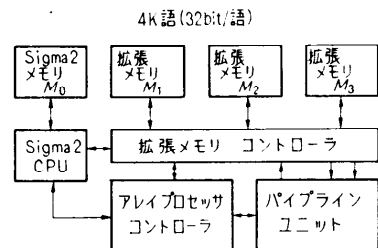


Fig. 6 The Block Diagram of Vector-I.

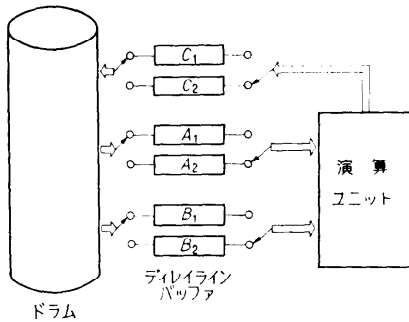


Fig. 7 The Block Diagram of NOVA.

VECTOR-I の命令は、非常に豊富であるが、機能的には、ベクトル演算、論理演算、シフト処理、ソート処理の4種からなっていて、命令でハードの各機能ブロックの結合を直接指定することによって、各種の演算をサポートしている。

パイプ・ラインの速度は  $1 \mu s$  で、メモリ・サイクルと同じである。

#### 4.10 RAYTHEON ATP

RAYTHEON ATP (Array Transform Processor) は、RAYTHEON の 703, 704, 706 に I/O として接続される。16, 32, 48 ビットの固定小数点データに対するフーリエ変換を主体とする 10 数種のアレイ演算命令を持っている。

#### 4.11 NOVA<sup>7)</sup>

NOVA は、ドラムを用いたアレイ・プロセッサであり、この分野では、特色のある実験機である。

ベクトル演算は、ドラムの二つのトラック上に配列したベクトルに対して行なわれ、結果は、第3のトラックに格納される。NOVA では、2式のディレイラインを設け、演算とドラムの読出し/書き込みを並列に処理している (Fig. 7)。

NOVA は、コア・メモリの代わりに安いドラム(ディスク)を使用してコアを用いた CPU 相当の高い性能が得られることを示したものである。

### 5. アレイ・プロセッサの今後の課題

アレイ・プロセッサは、今後の計算機専用化の一大方向であることに違いはないが、まだ緒についたばかりであり、今後の課題は多い。

- (1) アレイ・プロセッサの持つべき機能と役割の確立

アレイ処理の専用化といっても、極度に目的をしばったものから、ある程度の汎用性を持たせたものまで、各種のレベルがある。現在はまだ市場の形成段階にあり、市場の要求するアレイ・プロセッサの機能と役割の確立は、今後を待たなければならない。

- (2) 汎用機との接点の確立

パイプ・ライン型アレイ・プロセッサは、汎用プロセッサの延長上にあり、システム構成上、システム規模上、両者の接点をどこにおくか。

- (3) ソフトの開発

アレイ・プロセッサを使用する言語、アプリケーション・プログラムの開発などは、現在ではまだ特殊品の域を出ないが、市場の増大と相まって、組織的に行なう必要があろう。

- (4) アレイ・プロセッサ向き解法の開発

計算機の出現により、問題の解法が大きく変化したように、アレイ・プロセッサの特性を利用した問題の解法も、開発されるべきであろう。

#### 参考文献

- 1) Daniel L. Slotnick, W. Carl Borck and Robert C. McReynolds; The SOLOMON Computer. Proc. FJCC, 22, Dec., 1962, pp. 97-107.
- 2) G. H. Barnes, R. M. Brown, M. Kato, D. J. Kuck, D. L. Slotnick, R. A. Stones; The ILLIAC IV Computer. IEEE Trans. on C, Vol. 17, No. 8, Aug., 1968, pp. 746-757.
- 3) D. J. Kuck; ILLIAC IV Software and Application Programming, IEEEEC, Vol. 17, No. 8, Aug., 1968, pp. 758-770.
- 4) D. N. Senzig, R. V. Smith; Computer Organization for Array Processing. Proc. FJCC, 27, pt. 1, 1965, pp. 117-128.
- 5) S. F. Anderson, J. G. Earle, R. E. Goldschmit, D. M. Powers; The IBM System/360 Model 91: Floating-Point Execution Unit. IBM JOURNAL, JAN. 1967, pp. 34-52.
- 6) J. F. Ruggiero, D. A. Coryell; An auxiliary processing system for array calculations. IBM SYST J, No. 2, 1969, pp. 118-135.
- 7) Joseph E. Wirsching; NOVA; alist-oriented computer. Datamation, Vol. 12, No. 12, Dec. 1966, pp. 41-43.

(昭和46年5月12日受付)