

## 大規模並列セルラアレイプロセッサ: AAP-2

渡辺琢美 近藤利夫 北村美宏 杉山 吉

NTT LSI研究所

AAP-2は、65,536個の1ビットプロセッサのそれぞれに大容量のローカルメモリ(8kビット/プロセッサ)を付加したSIMD型セルラアレイプロセッサである。AAP-2では、従来の二次元アレイ接続のSIMD型マシンの欠点を克服するために、プロセッサ間ネットワークの可変構造、PE毎の各種修飾機能、高速データ転送機能等を採用し、高い規則性、単純性を保ったまま柔軟性の高いシステムを実現している。また、64Kのプロセッサアレイ部(1,024個のLSIと4,096個のローカルメモリチップ)をわずか0.2m<sup>3</sup>の容積に収納し、アレイ規模の拡張も容易な構成と成っている。現システムでは、16ビット整数加算の処理速度は2,520MOPSである。応用範囲は広く、特にLSI-CAD、画像処理、ニューラルネット・シミュレーション等の分野で高い性能が得られる。

A MASSIVELY PARALLEL CELLULAR ARRAY  
PROCESSOR: AAP-2

Takumi Watanabe Toshio Kondo Yoshihiro Kitamura Yoshi Sugiyama

NTT LSI Laboratories

3-1 Morinosato Wakamiya, Atsugi-shi, Kanagawa 243-01, Japan

This report describes the system design of the massively parallel cellular array processor, AAP-2, an SIMD machine. It consists of 65,536 one-bit processing elements (PE's) and 64MBytes of local memory. Special features include a variable structured PE interconnection networks, a high-speed data transfer mechanism and wideband modifiable operations in the PE. The array unit is compact and expandable, and can be assembled in a space of 0.2m<sup>3</sup> because of its simple and regular structure. On 16-bit integer data, addition can occur at 2,520 MOPS. The AAP-2 system can be used for a variety of applications including image processing, LSI-CAD problems and neural network simulations.

## 1. まえがき

最近、画像処理やLSI-CAD、さらにニューラル・コンピューティングの分野においてマルチプロセッサ型の大規模並列処理システムが注目を集めている。このシステムの実現には大量のハードウェア量を必要とするが、LSI技術の進歩により実用的な規模で実現可能となってきた。我々は、1978年以來、大規模で超高速なシステムを比較的容易に実現できる1ビット2次元SIMDプロセッサの特徴に着目し、VLSI技術を活用してその規則性・単純性を損なうことなく、一層の高性能化および柔軟性を達成することに主眼をおいたマルチプロセッサAAPの研究を進めてきた[1,2,3]。マルチプロセッサの特徴は、ハードウェア量の大半を占めるレイアウトが単純で規則的であるため、

(I) 新たに開発すべきLSIが1品種で済み、構成が規則的のため設計も容易である、  
(II) 上記LSIを規則的に並べるだけで容易にPEレイアウトを構成できる、  
(III) 同一LSIを大量に使用するので量産効果を生かすことができる、  
等により、極めて高いコスト性能比が得られることにある。

本報告では、VLSI技術を活用して新たに開発された大規模並列マルチプロセッサシステムAAP-2について述べる。まず、大規模1ビット2次元SIMDプロセッサの特徴について触れ、次にAAP-2のシステム構成、各種機能、性能、応用例等について述べる。

## 2. 大規模1ビット2次元SIMD型プロセッサの特徴と問題点

大規模1ビット2次元SIMDプロセッサの特徴としては、

- a. 語長の変化に強い、
  - b. ピーク性能が高い、
  - c. LSI化向きの構造、
- という長所がある反面、問題点としては、

a. プロセッサアレイの稼働率が低下しやすい、

b. 離れたプロセッサ間のデータ転送に時間がかかる、

などがあげられる。これらの問題点は1ビット2次元SIMDプロセッサの最大の長所である単純さと規則性の高さと裏腹の関係にある。AAP-2のアーキテクチャ設計においては、これらの長所を極力維持しつつ、問題点を解決するために、次の方針のもとに設計を行った。

(I) プロセッサアレイの性能向上は、主にPE(Processing Element)機能の改良によって行う。

(II) 隣接プロセッサ間以外の接続の増設は原則として避ける。

これらの結果、PE構成は若干複雑になったものの、全体のハードウェア構成は1ビット2次元プロセッサの単純性・規則性をそのまま継承することに成功している。

アーキテクチャ上の主な特徴は次の3点である。

(I) プロセッサアレイの構造をダイナミックに変える可変構造。

(II) PE間をブロックで区切ることなくデータを転送する高速伝搬転送、伝搬演算。

(III) 伝搬経路短縮用の階層的バス転送路。

以下に具体的な説明を行う。

## 3. AAP-2のハードウェア構成

### 3.1 基本構成

AAP-2は、65,536個の1ビットプロセッサのそれぞれに大容量のローカルメモリ(8kビット/PE)を付加したマルチプロセッサ型の計算機である。ハードウェア量の大半を占めるレイアウトは8×8PEを含むCMOS-LSI [4]を1,024個用いることで構成している。本装置は基本的には2次元レイアウト接続のSIMD型マシンであり、PE間の通信ネットワークは、ハードウェア構成を簡素化するために単純な隣接結合のみで構成しているが、各種修飾機能、2系統のカスタムデータ転送路等の採用により、柔軟な構成となっている。各PEは2組(2ビット)

の独立したデータ転送ユニットを持ち、1つは8隣接PEと、他方は4隣接PEとそれぞれ独立に接続し、2系統のネットワークを構成している。ロカメモリとしてLSI毎に8個のバイトアクセスRAM(64kSRAM)が接続されており、各PEに8kビットのロカメモリが専用に割り当てられている。

### 3.2 システム構成

AAP-2のシステム構成を図1に示す。AAP-2はインターフェイス部を介してホストコンピュータと接続されており、必要なインストラクションおよびデータはIM(インストラクションメモリ)、DBMにそれぞれ格納される。制御部は演算を行うとともにアレイ部に必要な制御信号を供給する。アレイ部は、図2に示

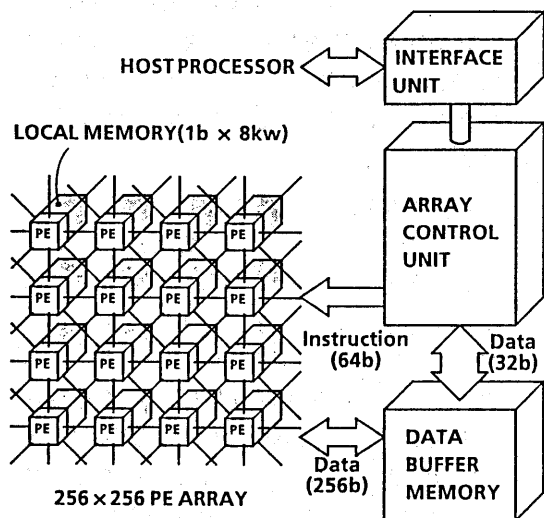


図1 AAP2システム構成

すようにLSIのそれぞれに64kビットのSRAMを8個ずつ搭載したモジュールを64個両面実装したPCB(アレイボード)16枚で構成されており、1枚のアレイボードで64×64のPEアレイ(4,096PE)を形成している。高密度実装形式の採用とLSI化によりアレイ部が非常にコンパクトに構成でき、わずか0.2m<sup>3</sup>の容積に65,536個の加算機と64Mバイトのロカメモリが収納可能となった。図3にAAP-2システムの写真を示す。

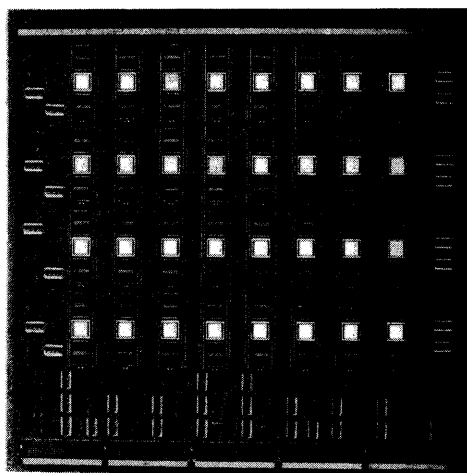


図2 AAP2ボード

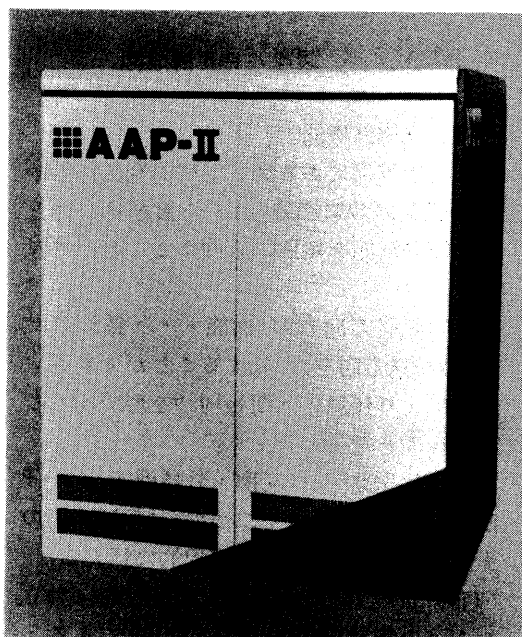


図3 AAP2システム

### 3.3 各種基本機能

アレイ部を構成するPEは制御レジスタを兼ねる1b×144Wのレジスタファイルを持つ演算部と2つのデータ転送部とからなる。演算部とデータ転送部の独立性は高く、2つのデータ転送路を独立に使用しながら演算部内で閉じた演算を行うことが可能である。図4にPEの構成図を示す。以下にアレイ部の基本機能を述べる。

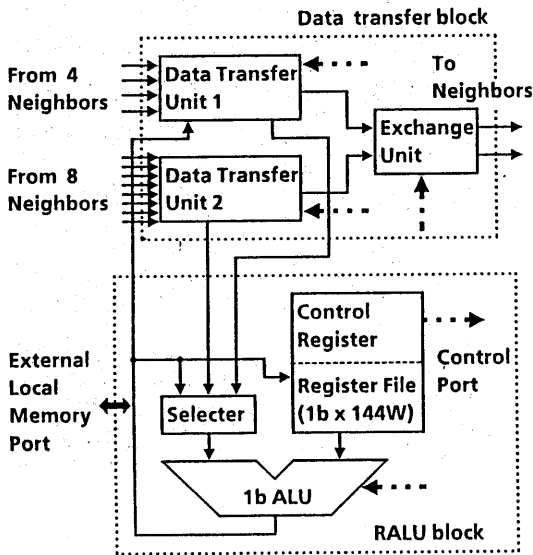


図4 PE構成図

(a) 各種修飾機能

PE内演算部にある8ビット構成の制御レジスタや活性制御レジスタに必要なデータを書き込むことで各PEの機能を個別に修飾することができる。

(i) 転送方向プログラム機能・・・各PEごとにデータ転送方向をプログラムすることができる。これにより任意のPE間ネットワークを形成することが可能となる。

(ii) 転送路間データ交換・分岐機能・・・PE内交換ユニットによって二つのデータ転送路間のデータ交換・分岐がPE毎に選択的に行える。

(iii) ALU機能修飾機能・・・PE毎に独立にALUの演算機能の設定が可能である。

(iv) 活性制御機能・・・演算部の活性制御レジスタを設定することにより、演算結果を選択的にレジスタファイルに格納できる(活性化されない場所のデータは前の値が保存される)。

(b) データ伝搬転送機能 (リプル転送)

PE間データ転送の高速化機能であり、データを一定時間伝搬させることで、高速にデータ転送を行う機能である。通常のPE間のデータ転送は、隣接接続バスを用いたシフトによって行うので、データはクロック毎に次々にPE間を移動

していく。一方、伝搬転送は、データ転送ユニットのレジスタをバイパスしてデータを転送する(必要なタイミングでレジスタに取り込む)ので、単位時間あたりデータ転送路の遅延によって決定する距離だけデータが転送される。

(c) 2方向、4方向への一括データ転送

データ転送ユニットIではPE内で隣接PEからの入力データのOR処理が行え、一方向だけでなく、水平、垂直さらに4方向へのデータ転送が一度に行える。4方向への一括データ転送と上記のデータ拡散転送機能を組合せると、図形のフィリング(塗り潰し)が高速に行える。

(d) 一括OR演算機能

全PE間でOR演算を行ない、結果を制御部へ転送する機能を持っている。これは主にALU部での演算結果によってプログラムの流れを変えるために用い、256×256PEレイの場合、3クロックで実行できる。

(e) 階層構造バス転送機能

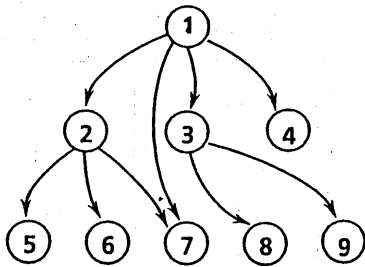
通常のPE間接続に加え、各PEは、PEレベル、チップレベル、およびボードレベルの3レベルのバスによって接続されている。バスは、LSIおよびLSIが実装されているプリント基板において、各PE、LSIの行および列毎にそれぞれ設けられている。チップレベルのバスは、7PE、ボードレベルのバスは、6個のLSIをスキップすることになる。このバスを用いてリプル転送を行なうことでPE行または列に沿った加算や放送等の命令が高速化される。

3.4 プロセッサ間通信

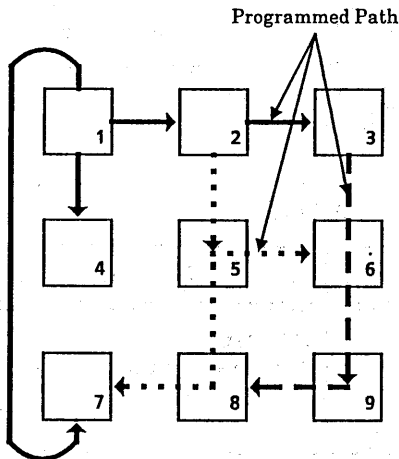
2次元結合のマルチプロセッサ型計算機はネットワークの柔軟性に欠け、応用範囲が狭いと言われているが、AAP-2では基本的には2次元結合形態をとりながらも上述の各種機能を使用して任意の接続ネットワークが形成可能となっている。隣接接続を利用した通常の転送(シフト)命令の他に次の2種類の転送方式が可能である。

(a) カラムデータ転送路による結合

カラム転送路とリプル転送機能を用い、データ転送路 I、II を 2 層ルータのように制御して加算機間の経路を設定する。この時、ある加算機間結合のデータ転送路として設定された PE が他の加算機間結合のソースまたはターゲットになっている場合はシケンサルに処理する。この結合方法は加算機間が直接結合されるため転送先アドレスや転送方向などの制御に関する処理が不要なため高速に転送が可能である。しかし、他のネットワークの転送路となった PE は同時にはデータ転送できないため、並列度が低く、加算機間の結合が多い場合には効率が低下する。図 5 は、(a) に示すネットワークを加算機アレイ上で実現した例 (同図 (b)) を示している。



(a)



(b)

図 5 任意のネットワークの形成

(b) バックデータ転送

データと送先きのアドレスをバックとして PE 間の通常のデータ転送路を用いて転送を行う機能である。この機能はマクロ命令としてサポートされている。

2.5 その他の機能

(a) パーティアルアレイ

論理アレイ空間を PE のローカルメモリ方向に折り畳むことによって物理サイズを越える規模の問題についても処理可能となる。AAP-2 用言語 (AAPL) でアレイサイズを宣言するだけで、AAPL コンパイラが自動的に折り畳み用のインストラクションを発生する。したがって、ユーザは物理的なアレイのサイズを気にすることなくコーディングできる。

(b) 放送機能

全 PE に同じデータを放送する場合は、ALU で演算を行ない、高速に PE 全体への放送が実現する。また行や列ごとの放送の場合は上記のリプル転送とハイパス機能を用いることで高速に放送することが可能である。

3.6 システムの拡張性

システムの拡張性は AAP-2 のようなシステムにとって重要な問題である。AAP2 はアレイ部の PCB の接続方法に工夫を加え、特別な設計変更なしに容易にシステム規模の拡張ができる。これは、図 6 に示すボード間の物理的接続図からわかるように、ボード間の接続はアレイの端の部分を除いて規則的であり、布線距離の局所的な増大がない構成をとっているからである。現在のシステムでは 1 ラックで 256 × 512 PE アレイまで拡張可能であり、それ以上の拡張もラック間をコネクタケーブルで接続するだけで実現できる。このときも、ボード間接続の大幅な変更やラック間のケーブル長の増大の問題は生じない。

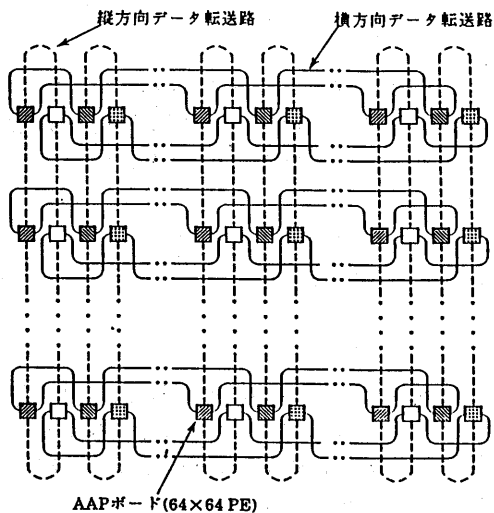


図6 ボード間の物理的接続図

#### 4. ソフトウェア

AAP-2を動作させるためにはホスト側、アレイ側加算機が必要である。ホスト側加算機はホスト計算機上で行ったほうが効率的な処理、およびAAP2の起動・停止等を記述する。アレイ側プログラムは、二次元アレイ構造の計算機上での一般的なデータ処理とAAP2特有の機能を効率良く表現するために開発された加算機言語AAPL (AAP Language)を用いる[5]。AAPLの特徴は、

(1) APLをベースにした言語でAPL記号を文字列で表現している。例えば、a, b, cがアレイ変数の時、 $a=b*c$ でアレイの各要素の乗算が表現できる。

表1 AAPLの基本命令とマクロ命令の例

命令の種類		命令	説明
基本	演算	\$AND, \$OR	論理演算 シフト(データ転送)
	データ転送 処理制御	\$SFTN, \$SFTW \$GOTO, \$CALL \$DOWHILE, \$CASE	
マクロ	演算	+, -, *, /	算術演算 論理演算 演算修飾(疑似MIMD処理)
	データ転送	\$ANDH, \$MAXH \$MIMD, \$MERGE \$RIPLN, \$RIPLP	

(ii) PE内で閉じて行う演算とPE間データ転送の交絡した記述が可能。

(iii) パーチャル・アレイの概念の導入により、任意のビット長のデータや物理空間を越えたアレイを扱うことが可能。

等である。表1はAAPLの基本命令とマクロ命令の一例である。

#### 5. 基本性能

表2にAAP-2の基本処理速度を示す。また、AAP-2の基本的な演算と同じ機能の演算をIBM3090上で実現した場合の処理時間を比較

表2 AAP2の基本性能

(16ビット整数)

(A:アレイデータ、S:スカラデータ)

演算	演算時間(μsec)	処理速度(Mop/s)
A + A	26	2520
A * A	116	564
A * S	154	426
\$BROADCAST S <sup>*1)</sup>	38	1742
\$MAXH A <sup>*2)</sup>	194	338
\$SUMH A <sup>*3)</sup>	161	406

\*1): スカラデータをアレイ全体へ放送

\*2): 列毎の最大値

\*3): 列毎の総和

を表3に示す。これからわかるように、AAP-2は基本的な二次元演算において、逐次型の大型汎用計算機と比べて数10～数1000倍処理能力があることがわかる（データ規模をAAP-2の物理アドレスである256×256に設定）。

表3 AAP2処理速度比較

演算	マシン名		
	IBM-3090	IBM-3090(VF) <sup>*2</sup>	AAP2
四則演算	120msec (1)	6~30msec (4~20)	0.03~0.8msec (150~4000)
放送 (全配列要素 への代入)	40msec (1)	4msec (10)	0.04msec (1000)
配列要素 全ての加算	30msec (1)	3~7msec (4~10)	1.42msec (20)
マスク演算	30~120msec (1)	6~8msec (4~20)	0.42msec (50~250)
配列要素 のシフト	70~80msec (1)	5~20msec (4~16)	0.044msec (1500)
行/列毎の 加算	10~50msec (1)	5~8msec (1~10)	0.32msec (30~150)

\*1 256×256配列、32ビット整数

\*2 ベクトルプロセッサ付き

\* カッコ内はIBM-3090からみた速度比

## 6. 各種応用

AAP-2は、問題の持つ並列性を利用して処理を行うことで、非常に高い処理能力を発揮する。以下にAAP-2の特長を生かした主な応用例を示す。

(1) 配置・・・256×256のPEアレイを8×256の32ビットの加算ユニット(PU)アレイとすることで、最急降下法に基づく交換法の並列処理が実現できる。各PUにモジュール(Mi)を割り当て、PUの中にはMiの位置情報とMiに接続するネットに含まれるモジュールR(Mi)に関する位置情報とそのネット識別番号が格納されている。まず交換モジュールMを全PUに放送し、MとMiとが交換される場合の総配線減少量Diを各モジュール毎に並列に計算する。次にDiの最大値を計算し、最大のDiを与えるモジュールMjとの交換を行い、それに伴うデータの更新を行う。以上の処理を所定回数繰り返す。AAP-2上で並列処理を行うことにより、モジュール数をN

とすれば、AAP-2上で並列処理を行うことにより、総配線減少量Diの計算は1/Nに、Diの最大値の計算は(logN)/Nに短縮される。

(2) 配線・・・AAP-2のデータ転送ユニットは一種のハードウェアとみなすことができる。従って高速に配線径路を見付けることができる。例えば、迷路法のように二次元的に探索を行って径路を見出す場合、PEを配線格子に対応させ、4方向一括のデータ転送機能を用いると、波紋が広がっていくような二次元的な探索が同時（並列）に行なわれる。また、データ間の乗換機能を用いて二層配線が実現される。さらに、ハードウェアの特徴と問題の持つ並列性を利用して新しい並列配線アルゴリズムも提案されている[6]。

(3) 論理シミュレーション・・・アレイ上に論理回路とその結線情報をそのままマッピングし、各々ユニット毎に、信号の入力と、各PEで同時に演算を行な演算モード、および信号の伝搬を行う転送モードを繰返し、論理回路をシミュレートすることができる。論理ゲート間の結線は、上記の配線により行ない、転送路をプログラムしておく。この方法の最大の特徴は、アレイ上にマッピングされた全論理ゲートの演算、ゲート間の信号伝達を、実際の論理回路の動作と同様に、並列に処理することにある。この方法は、AAP-2の2系統のデータ転送路とそれらの交換機能を共に利用することで高速化を図っている。

(4) ニュラルネット・シミュレーション・・・AAP-2はニューロコンピュータとしても強力な力を備えている。各ニューロンあるいは、ニューロン間の結合をPEに割り当てることで65,000個の加算を効率良く動作させ、高い性能を得ることができる[7]。

(5) その他・・・画像処理、図形処理等の規則性が高く、語長の短い大規模データ処理において特に高い性能を発揮する[8]。

## 7. まとめ

大規模並列セルアレイプロセッサAAP-2システムについて述べた。実験結果から、AAP-2の基本命令で、大型汎用計算機(IBM-3090)の数10~数100倍の処理速度が得られることを示した。PE内の各種修飾機能と高速データ転送機能の採用により、1ビット2次元SIMDプロセッサの特長である高い規則性と単純性を保ったまま、柔軟性のある、高性能プロセッサを開発できた。大規模なメモリを持ったシングルプロセッサアレイから成るAAP-2はコスト性能比の面で特にLSI化に向いており、LSI-CADをはじめ各種画像処理やニューラルネット・シミュレーション等、幅広い応用分野で高い性能を発揮する。

### [謝辞]

日頃ご指導頂く須藤カスタム化技術研究部長、中島、武谷両主幹研究員に深く感謝します。

### [参考文献]

- [1]T. Sudo et al., "An LSI Adaptive Array Processor", ISSCC Dig.Tech.Papers, pp.122-123,1982.
- [2]T. Kondo et al., "A Large Scale Cellular Array Processor:AAP-1", Proc. Computer Science Conf.'85 ACM,Mar.1985.
- [3]T. Kondo et al., "Pseudo MIMD Array Processor - AAP2", IEEE 13th Int'l Symposium on Computer Architecture, pp. 330-337, 1986.
- [4]近藤他, "高並列処理用セルアレイプロセッサ LSI-AAP2," 信学技報 SDM87-104, ICD87-62, pp.25-30,1987.
- [5]近藤他, "二次元アレイプロセッサ(AAP2)とプログラミング言語," 電子情報通信学会論文誌, vol. J71-D, no.8, pp.1399-1406, 1988.
- [6]T. Watanabe et al., "A Parallel Adaptive Routing Algorithm and its Implementation on a Two-Dimensional

Array Processor", IEEE Trans on CAD Vol. CAD-6, No.2, pp.241-250, Mar 1987.

[7]T. Watanabe et al. "Neural Network Simulation on A Massively Parallel Cellular Array Processor:AAP-2", to be published in Proc. of IJCNN-89.

[8]T. Kondo et al. "Kanji Character Recognition Unit with Hand-Scanner Using SIMD Processor," Visual Communications and Image Processing '88, vol.1001, pp.476-482, 1988.