

並列計算機ADENA

Parallel computer ADENA

ADENAシステムの実現
Implement of ADENA system

谷川 裕二 金子 克幸 廉田 浩
Tanikawa Yuji Kaneko Katsuyuki Kadota Hiroshi

松下電器産業 (株)
Matsushita Electric Industrial Co., Ltd.

あらまし 科学技術分野における大規模な2次元および3次元問題を扱うシミュレーション指向型のアーキテクチャを有する並列計算機ADENAについて報告する。まず、ADENAのネットワークの考え方の基礎となる、ADI法とADE操作について概説する。次に、ホスト計算機とADENAとを含む並列のハードウェア・アーキテクチャとシステム・ソフトウェアについて述べる。256PEを用いたこの計算機の実効的性能は1GFLOPS以上と予想される。

Abstract This paper will describe the architecture of a parallel computer, ADENA, which is applicable to various numerical analysis or simulations in two or three dimensions. The Network design of ADENA is based on ADI method and ADE operation, which will be discussed, first. And, parallel hardware architecture, relationship between host computer and ADENA machine and system software will be explained. The effective performance of the machine with 256PE's is expected to be over 1 GFLOPS.

1. はじめに

近年、科学技術の基礎と応用の幅広い研究・開発分野において、コンピュータ・シミュレーションの役割が益々大きくなり、より高度な利用へと進んで来ている。その中において、2次元及び3次元問題の大規模なシミュレーションの要望が強く望まれている。この大規模なシミュレーションが容易に実現できる意義は極めて大きく、産業技術において最適設計のための信頼できる方法を提供することが可能となる。

大規模なシミュレーションを行うためには、高速に効率良く処理を行う計算機が必要である。計算機の処理速度を向上させるアプローチには、ベクトル・パイプライン方式と並列処理方式とがある。

大規模なシミュレーションに対しては、最新のベクトル・パイプライン方式のスーパー・コンピュータをもってしても、計算速度および記憶容量において能力不足の

場合が多い。

そこで、標準的な演算能力を持つプロセッサを複数個用いた並列処理方式に着目し、並列処理で重要な演算処理の途中における演算エレメント間のデータ交換に特徴を持つ並列計算機で、特に科学技術計算において高い性能と汎用性を持つ並列計算機ADENA (Alternating Direction Edition Nexus Array: 交互方向編集アレイ)¹⁾⁻³⁾の開発を行なう。この計算機は、計算数学、シミュレーション・スキーム等の研究に基づいて、京都大学工学部数理工学科の野木達夫助教授が提案したシミュレーション指向型のアーキテクチャを有する並列計算機である。現在、同研究室と共同研究および共同開発を行っている。

この計算機は、ポアソン方程式などの2次元及び3次元の偏微分方程式の解法と原子及び分子の挙動を扱う粒子モデルの計算を対象にしている。

本報告では、並列計算機ADENAのアーキテクチャ

の基礎となるADI (Alternating Direction) 法とADE (Alternating Direction Edition) 操作とを概説し、数学的なアルゴリズムとこの計算機のアーキテクチャとの関係を明確にするとともに、開発中の並列計算機ADENAのシステム構成について述べる。

2. ADENAにおける計算方式

偏微分方程式の求解やFFTのような数値計算を計算機を用いて行う場合、計算機を高速に効率よく実行させるためには、計算機のアーキテクチャに適合したアルゴリズムが必要になる。一般に数値計算アルゴリズムとその応用は、それぞれの計算機のアーキテクチャをもとに研究されている。

偏微分方程式の解法を例にとれば、SLOR (Successive Line Over Relaxation) 法やICCG (Incomplete Conjugate Gradient) 法は逐次処理型やベクトル型の計算機に適した解法であり、ガウス・ザイデル法やPSOR (Point Successive Over Relaxation) 法は格子型のプロセッサ間結合を持つPAXやILLIAC IVに適した解法である。

本報告の並列計算機ADENAのアーキテクチャは、大規模なシミュレーションの数値解法アルゴリズムの一つである『ADI法』を処理するのに最適なシステム構成を備えている。

このADI法は、多次元問題を首尾よく解くための基本的原理を与えている。すなわち、無条件的収束性/安定性を持つセミ・インプリシット・スキームであり、計算量もオーダ的にエクスプリシット・スキームと同程度の単純性を持つ。多次元処理においても、1次元処理を繰返し行うもので、この1次元方向にインプリシット性を含んでいる。

まず、このADI法を説明するために、2次元ポアソン方程式に対する境界値問題の数値解法を考える。ポアソン方程式を次式に示す。

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = F \quad (1)$$

この偏微分方程式を計算機で解くために、差分化を用いて、次式のように書き直す。説明を簡単にするために $F=0$ として以下説明する。

$$(u_{i-1,j} - 2u_{i,j} + u_{i+1,j}) + (u_{i,j-1} - 2u_{i,j} + u_{i,j+1}) = 0 \quad (2)$$

この差分方程式を解く反復解法を構成するために、

カッコで括られた第1項と第2項に示すように添字の i 方向および j 方向の差分に着目する。それぞれの方向の差分問題をインプリシット的に解く2段階構成とする。

$$u_{i-1,j}^{(n+1/2)} - (2+r)u_{i,j}^{(n+1/2)} - u_{i+1,j}^{(n+1/2)} = -u_{i,j-1}^{(n)} + (2-r)u_{i,j}^{(n)} - u_{i,j+1}^{(n)} \quad (3-1)$$

$$u_{i,j-1}^{(n+1)} - (2+r)u_{i,j}^{(n+1)} + u_{i,j+1}^{(n+1)} = -u_{i-1,j}^{(n+1/2)} + (2-r)u_{i,j}^{(n+1/2)} - u_{i+1,j}^{(n+1/2)} \quad (3-2)$$

ここで r は加速パラメータである。この2式を結果が収束するまで交互に繰返し演算することにより、定常解を求めることができる。この処理を『ADI法』と言う。(3-1) 式の右辺は、前演算ステップで求めた結果であるので、定数 $b_{i,j}$ と置くと、

$$u_{i-1,j} - 2u_{i,j} + u_{i+1,j} = b_{i,j} \quad (4)$$

と書き直すことができる。これは、次の順方向、逆方向の2つの逐次計算によって解ける(消去と代入)。

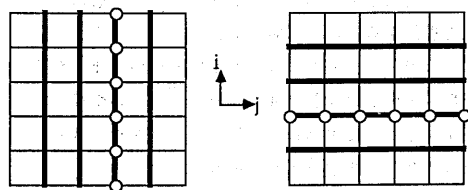
$$\left. \begin{aligned} L_{i,j} &= \frac{1}{2+r-L_{i-1,j}} \\ M_{i,j} &= \frac{M_{i-1,j}-b_{i,j}}{2+r-L_{i-1,j}} \end{aligned} \right\} \quad (5)$$

$$u_{i,j} = L_{i,j} u_{i+1,j} + M_{i,j} \quad (6)$$

(3-2) 式も同様に解くことができる。

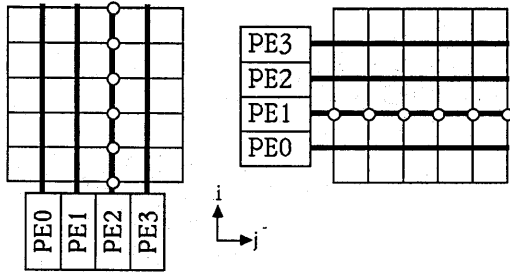
このADI法の処理に、単一の演算エレメント (PE: Processing Element) を用いて、この2式を交互に繰返し演算を行う場合を考える。図2.1に示すように、(3-1) 式では i 方向の1次元配列に対して順次処理が行われ、(3-2) 式では j 方向の1次元配列に対して順次処理が行われる。

この i 方向および j 方向の1次元配列に対する演算は、それぞれ j 方向および i 方向に独立して処理することが可能である。そこで、図2.2に示すように、複数の演



(a) i 方向の演算 (b) j 方向の演算

図2.1 ADI法

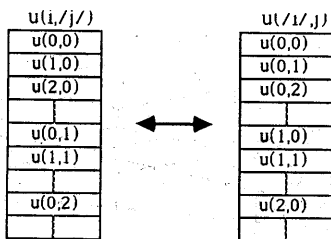


(a) i方向の演算 (b) j方向の演算
図2.2 ADI法の並列処理

算エレメントを用いて並列処理を行うことができる。この場合、各演算エレメントは、図2.3に示すように、i方向とj方向の1次元配列を処理することになる。全体でi方向を処理してj方向の処理にうつる（あるいはその逆）場合、方向の違う1次元配列の組に編集し直すが必要になる。この編集操作を『ADE』という。この並列処理によるADI法とADE操作とを実現するためのハードウェア構成を図2.4に示す。これは、N個の1次元に配置された演算エレメントと、N×N個の2次元に配置されたバッファ・メモリ（BM: Buffer Memory）によって構成され、i方向およびj方向からの演算は1次元に配置された各演算エレメントと2次元に配置されたバッファ・メモリとの接続方向を切り換えることにより交互に並列処理することができる。

この2次元用のシステムは、京都大学工学部ですでに『ADENA-I』としてプロトタイプが製作されている。

この2次元用システムの考え方をそのまま3次元用システムに拡張すると、2次元に配置された演算エレメントと3次元に配置されたバッファ・メモリが必要であり、2次元配置の演算エレメントと3次元配置のバッファ・メモリとの接続も3方向必要となる。この3方向からの接続をもった3次元に配置されたバッファ・メモリを構



(a) i方向配列 (b) j方向配列

図2.3 ADE操作

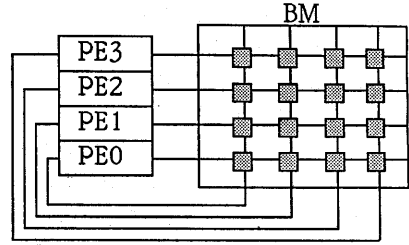


図2.4 ADENA-Iの構成

成するのは、演算エレメントの数が増すにつれて、ハードウェアが複雑でかつ大規模なものとなり、実現することは困難である。

この3次元用システムを構成する場合の問題点を明らかにするために、2次元問題と同様に、3次元問題の偏微分方程式の数値解法を考える。3次元の偏微分方程式を次式に示す。

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = F \quad (6)$$

この偏微分方程式を計算機で解くために、差分化を行って、次式のように書き直す。説明を簡単にするために $F=0$ として以下説明する。

$$\begin{aligned} & (u_{i-1,j,k} - 2u_{i,j,k} + u_{i+1,j,k}) \\ & + (u_{i,j-1,k} - 2u_{i,j,k} + u_{i,j+1,k}) \\ & + (u_{i,j,k-1} - 2u_{i,j,k} + u_{i,j,k+1}) \\ & = 0 \quad (7) \end{aligned}$$

この差分方程式は、カッコで括られた第1項、第2項と第3項に示すようにi、jおよびk方向の成分に着目すると、i、jおよびk方向の1次元の差分方程式に書き直すことができる。2次元問題の(4)式と同様に、前演算ステップの結果を定数bとして、それぞれi、jおよびk方向の1次元の式として表現すると次の3つの式が得られる。(rは加速パラメータ)

$$\begin{aligned} & u_{i-1,j,k} - (2+r)u_{i,j,k} + u_{i+1,j,k} \\ & = b_{i,j,k} \quad (6-1) \end{aligned}$$

$$\begin{aligned} & u_{i,j-1,k} - (2+r)u_{i,j,k} + u_{i,j+1,k} \\ & = b_{i,j,k} \quad (6-2) \end{aligned}$$

$$\begin{aligned} & u_{i,j,k-1} - (2+r)u_{i,j,k} + u_{i,j,k+1} \\ & = b_{i,j,k} \quad (6-3) \end{aligned}$$

この3式を結果が収束するまで交互に繰返し演算することにより、定常解を求めることができる。

これより、図2.5に示すように、i方向の演算に対してjとk方向に、j方向の演算に対してkとi方向に、

k方向の演算に対してiとj方向に並列処理することができる。したがって、3次元用のシステムは、2次元に配置された演算エレメントと3次元に配置されたバッファ・メモリとより構成される。

図2.5に示すように、3方向からの演算を2次元の場合と同じような構成で行うと、3次元配置のバッファ・メモリに対して3方向からの接続手段が必要となる。この3次元配置のバッファ・メモリの構造を簡単にすることを考える。そこで、演算エレメントで演算処理されるデータに注目する。図2.6に示すように、jk面からki面へ、ki面からij面へ、そしてij面からjk面へのADE操作におけるデータの移動を示すと、

$$PE_{/k,i/}(j) \leftarrow PE_{/j,k/}(i) \quad (9-1)$$

$$PE_{/i,j/}(k) \leftarrow PE_{/k,i/}(j) \quad (9-2)$$

$$PE_{/j,k/}(i) \leftarrow PE_{/i,j/}(k) \quad (9-3)$$

となる。ここで、 $PE_{/i,j/}(k)$ は、2次元に配列されたi行j列の演算エレメントのk番目のデータを示す。

そこで上記の3式の添字i, j, kを、(9-1)式では $x=i, y=j, z=k$ と、(9-2)式では $z=i, x=j, y=k$ と、(9-3)式で、 $y=i, z=j, x=k$ と置き変えると、3式は全て

$$PE_{/z,x/}(y) \leftarrow PE_{/y,z/}(x) \quad (10)$$

となる。これは、2次元配列のz列目のN個の各演算エレメントがそれぞれx番目のデータをz行x列目の演算エレメントに転送することを示す。

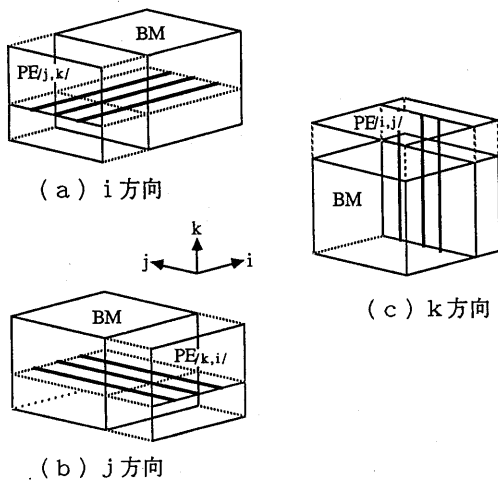


図2.5 3次元ADI法

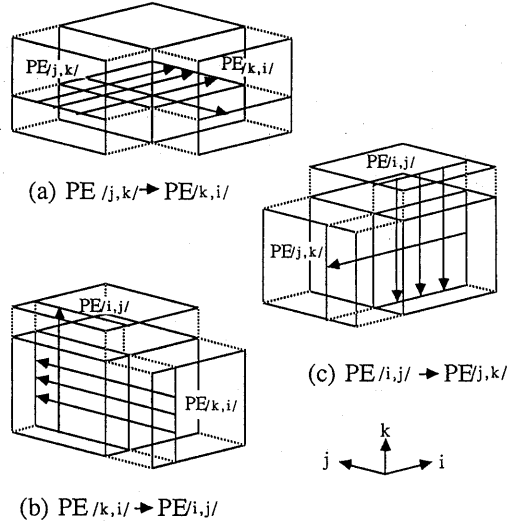


図2.6 3次元ADE操作

以上より、3次元用のシステムでは、2次元の演算エレメントと3次元のバッファ・メモリとを備え、各演算エレメントは3次元のバッファ・メモリを介して(10)式を満たすように3次元のバッファ・メモリに対して2方向からの接続を持ったネットワークを構成する。このネットワークをADENAネットワークと呼ぶ。

(10)式に示す演算エレメント間の接続を2回用いると、

$$PE_{/k,m/} \leftarrow PE_{/j,i,k/} \leftarrow PE_{/i,j/} \quad (11)$$

となる。これは、任意の2つの演算エレメント間でデータ転送ができることを意味する。この2回転送は、2次元のADI法の解法および粒子モデルの演算に用いることができる。

この3次元用のシステム『ADENA-II』の構成を図2.7に示す。

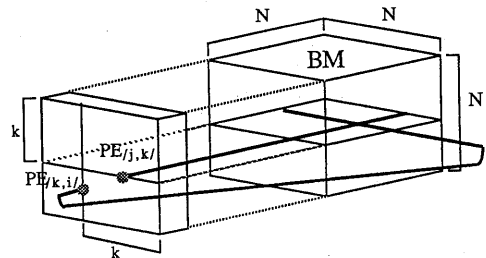


図2.7 ADENA-IIの構成

3. ADENAシステムの構成

本報告の並列計算機ADENAは、各演算エレメントが独立した命令メモリ及びデータ・メモリを持つMIMD (Multiple Instruction Stream Multiple Data Stream) 型の並列計算機で、 16×16 個の2次元配置の演算エレメントと $16 \times 16 \times 16$ 個の3次元配置のバッファ・メモリとより構成され、並列処理記述言語により、3次元モデルで 100^3 、2次元モデルで 1000^2 程度の格子点数で、各格子点あたり20パラメータを持つシミュレーション・モデルを処理することができ、1GFLOPS以上の処理能力を予想する。

本報告の並列計算機ADENAシステムの特徴を次に示す。

- 1) 256 (16×16) プロセッサ・システム
- 2) オリジナル64ビット浮動小数点演算ユニット
- 3) $100 \times 100 \times 100$ 規模のシミュレーション
- 4) 並列処理ステートメントを含むFORTRAN
- 5) UNIX環境下で動作

以下で、ハードウェアおよびソフトウェアのシステム構成について説明する。

3.1 ハードウェアの構成

本システムは、図3.1に示すように、3つのユニットより構成される。

- 1) ホスト計算機
- 2) インターフェース・ユニット
- 3) ADENAキューブ

以下に、各ユニットの構成を簡単に説明する。

i) ホスト計算機 (EWS)

ホスト計算機は、プログラムの作成、データの保存、結果の表示、システム全体の同期制御および演算処理におけるシリアル演算を行なうものである。

本システムのホスト計算機には、32ビット・マイクロ・プロセッサとIEEE形式の倍精度浮動小数点演算を行う浮動小数点演算ユニットとを用いた汎用のエンジニアリング・ワークステーション (EWS) を採用する。

ホスト計算機の機能を次に示す。

- 1) 『UNIX』オペレーティング・システム
- 2) 32ビット・システム・バス

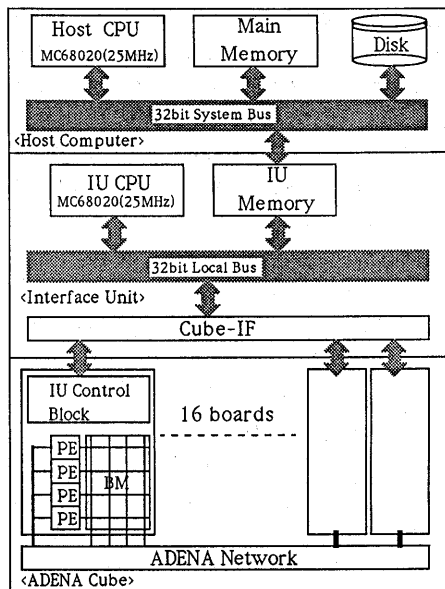


図3.1 並列計算機ADENAの構成

- 3) 3次元カラー・グラフィック・ディスプレイ
- 4) 大容量ハード・ディスク

ii) インターフェース・ユニット (IU)

インターフェース・ユニットは、ホスト計算機のサブ・モジュールとしてホスト計算機の内部に構成され、ホスト計算機とADENAキューブとの間でデータの転送およびADENAキューブの動作制御を行う。

このインターフェース・ユニットは、次の3つのブロックより構成される。

- 1) IUプロセッサ部 (IU_CPU)
- 2) IUメモリ部 (IU_MEM)
- 3) キューブ・インターフェース部 (Cube_IF)

IUプロセッサ部は、ホスト計算機と同一の32ビット・マイクロ・プロセッサを用いたCPUモジュールで、ホスト計算機からのコマンドに従って、ホスト計算機のシステム・バスとは独立したローカル・バスを介してキューブ・インターフェース部よりADENAキューブに対し、データ転送および制御を行うことができる。

ホスト計算機とのデータおよびコマンドの送受は、IUメモリ部を介して行なわれる。IUメモリ部は、ホスト計算機のシステム・バスとインターフェース・ユニットのローカル・バスとの2つのバスからアクセスすることが可能な2ポートRAMモジュールである。

同時に複数の演算エレメントに対するデータ転送を行うために、演算エレメントの2次元配置に対応した選択パターンをあらかじめキューブ・インターフェース部のRAMに記憶する。IUプロセッサ部は、実行時に、対応した選択パターンを示すRAMのアドレスを指定することにより、同時に複数の演算エレメントの選択を容易に行うことができる。

iii) ADENAキューブ

ADENAキューブは、2次元に配置された演算エレメントと3次元に配置されたバッファ・メモリおよびADENAネットワークより構成される。

ADENAキューブの構成は、図3.2に示すように、インターフェース・ユニットのキューブ・インターフェース部のIU制御ブロックと16個の1次元に配置された演算エレメントと16×16個の2次元に配置されたバッファ・メモリを1枚の基板上に配置し、この基板を16枚組合せて、マザー・ボードにより各基板間の接続を行なうことにより、システムを実現している。

各演算エレメントは、独立に64ビットの浮動小数点演算を行いながら、バッファ・メモリを介してデータ転送を行う。演算エレメントの構成は、図3.3に示すように、独立した64Kワード/24ビットのローカル命令メモリLIM(Local Instruction Memory)と1Mワード/72ビットのローカル・データ・メモリLDM(Local Data Memory)、浮動小数点演算ユニットFPU(Floating Point Arithmetic Unit)およびデータ転送制御ユニットTCU(Data Transfer Control Unit)より構成される。

浮動小数点演算ユニットは、64Kワード/24ビットの命令バスと1Mワード/72ビットのデータ・バスが分離され命令フェッチとデータ・アクセスを同時処理

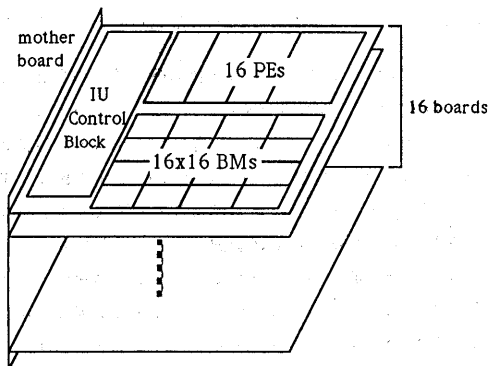


図3.2 ADENAキューブの構成

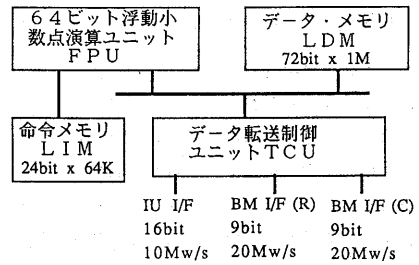


図3.3 演算エレメントの構成

し、46種類のRISCタイプの命令セットを有し、5段のパイプライン構成により64ビットの浮動小数点演算の加減乗算を100nsec、除算を400nsecで処理することができる。

データ転送制御ユニットは、浮動小数点演算ユニットが演算した結果を効率良く転送するための機能を有し、バッファ・メモリに対する9ビット/ワードの2つの転送ポートとインターフェース・ユニットに対する16ビット/ワードの入出力ポートを備え、各ポートは20Mバイト/秒の転送を行なうことができる。

ローカル・データ・メモリのデータ形式は、64ビットの浮動小数点データと8ビットのECCコードより構成される。浮動小数点データ形式は、ホスト計算機のデータ形式と同様に、IEEEの倍精度データ形式である。

浮動小数点演算ユニットは、演算結果で転送するデータの場合、ローカル・データ・メモリに書き込むと同時にデータ転送制御ユニットに対してデータ転送要求を発生する。このデータ転送要求が発生されるとデータ転送制御ユニットは、バッファ・メモリに対する一方の転送ポートより、要求が発生した個数のデータを順次ローカル・データ・メモリより読み出し送出する。この送出されたデータは、3次元配置されたバッファ・メモリに書き込まれる。データ転送制御ユニットはこのデータをもう一方の転送ポートより順次読み込み、ローカル・データ・メモリに書き込む。これにより、転送に要する時間を演算処理の時間とオーバー・ラップさせることができる。

3.2 ソフトウェアの構成

ADENAシステムのソフトウェア構成は、図3.1に示すシステム構成に対応して、図3.4に示すように、ホスト計算機、インターフェース・ユニットおよびADENAキューブの3つの環境を提供する。

以下、各ユニットについて説明する。

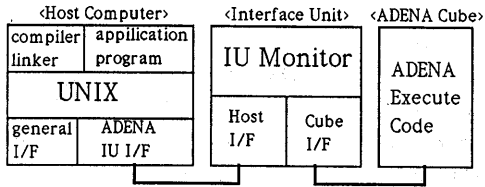


図3.4 ADENAソフトウェアの構成

1) ホスト計算機 (EWS)

ホスト計算機のオペレーティング・システムOSとしては、ソフトウェアの開発環境が整備され移植性に優れた『UNIX』を採用する。

このUNIXの開発環境を利用し、並列処理記述が可能な『ADETRAN』(ADENA FORTRAN)コンパイラ、アセンブリ言語、リンカおよびローダ等を開発する。

さらに、アプリケーション・プログラムおよびライブラリ関数の開発も行なう。

ホスト計算機でコンパイルされた実行コードおよびデータは、ローダによりUNIXのデバイス・ドライバを介して、インターフェース・ユニットに転送される。

ii) インターフェース・ユニット (IU)

インターフェース・ユニットは、ホスト計算機からの制御コマンドおよびデータを受け取り、ADENAキューブの各演算エレメントに対してデータ転送および演算実行制御を行うIUモニタを備えている。

iii) ADENAキューブは、ホスト計算機でコンパイル、リンクされた実行プログラムを各演算エレメントにおいて並列実行させる。

並列処理記述が可能な『ADETRAN』⁴⁾は、野木達夫助教授が提案したもので、FORTRAN77をもとに、ADENAシステムに特有な並列動作やデータ転送等に関する処理ステートメントを付加させた科学技術計算用言語である。

この言語では、図3.5に示すように、ホスト計算機により処理されるシリアル処理記述とADENAキューブで処理される並列記述を同一ファイル内に記述することが出来る。このプログラム記述において、(1) SENDステートメントは、ホスト計算機からADENAキューブへのデータ転送を示す。(2) pdo ステートメントは、j k面におけるi方向の1次元の演算を示す。(3) passステートメントは、ADENAキューブ内でj k面よりk

```

gsubroutine cul(a,b)
real a(N,N,N),b(N,N,N),a(/,/),b(/,/),a(/,/),b(/,/),
send i,j,k=1,16
    a(i,j,k)=a(i,j,k) (1)
endsend
pdo j,k=1,16
    do 10 i=1,16
        a(i,j,k)=a(i,j,k)+b(i,j,k) (2)
    10 continue
endpdo
pass i,j,k=1,16
    a(i,j,k)=a(i,j,k) (3)
endpass
pdo k,i=1,16
    do 20 j=1,16
        a(i,j,k)=a(i,j,k)-b(i,j,k) (4)
    20 continue
endpdo
receive i,j,k=1,16
    a(i,j,k)=a(i,j,k) (5)
endreceive

```

図3.5 ADETRANソース

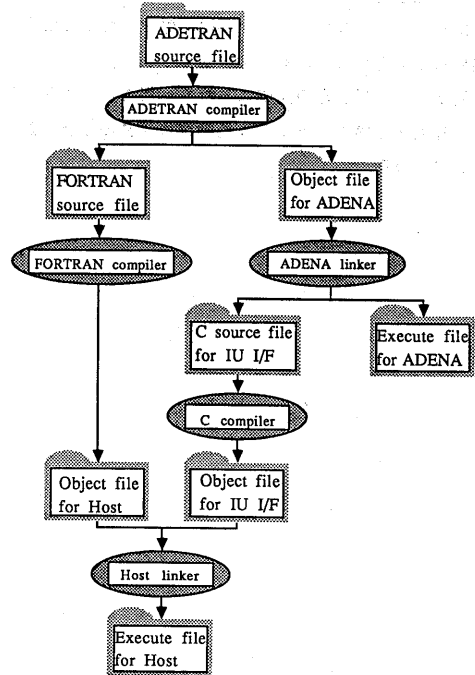


図3.6 ADETRAN.

i面へのデータ転送を示す。(4) pdo ステートメントは、k i面におけるj方向の1次元の演算を示す。最後の、(5) receive ステートメントは、ADENAキューブよりホスト計算機へのデータ転送を示す。

ADETRANコンパイラの処理フローを図3.6に示す。ADETRANのソース・ファイルは、プリ・コンパイラにおいて、シリアル処理記述と並列処理記述とに分離する。ホスト計算機が処理するシリアル処理記述部分は、ホスト計算機の有するFORTRANコンパイラを利用してコンパイルする。一方、並列処理記述部分は、ADETRANコンパイラでコンパイルを行い、ホスト計算機とのデータ転送制御ステートメントは『C』言語のソース形式で出力し、ホスト計算機のCコンパイラでコンパイルして、FORTRANコンパイルの結果とリンクする。

4. おわりに

本報告の並列計算機ADENAシステムは、実装上の制約により256 (16×16) 個の演算エレメントを持つシステム構成で開発を行っている。今後は、アプリケーション・プログラムおよびライブラリの開発とシステムの処理能力の評価を行う。さらに上位のシステムの開発とADENAネットワークの特徴を生かした応用の開発などを行なっていく予定である。

最後に、本報告書の完成にあたり、指導いただきました京都大学工学部の野木達夫助教授に深甚の謝意を表すものである。

参考文献

- 1) T.Nogi, Parallel Machine ADINA, in Computing Methods in Applied Sciences and Engineering, V, eds.R. Glowinsky and J.L.Lines, North-Holland,1982, 103-122
- 2) T.Nogi, The ADENA Computer, in International Symposium on Applied Mathematics and Information Science, Kyoto University, 1984,7/9-16.
- 3) T.Nogi, Parallel Computation, Patterns and Waves - Qualitative Analysis of Nonlinear Differential Equations -, p.279-318(1986)
- 4) 野木：並列言語ADETRAN, 第1回数値流体力学シンポジウム講演論文(文部省重点領域研究『数値流体力学』) P343-346(1987)
- 5) 若谷, 野木：並列計算機用言語ADETRAN処理系の開発 情報処理学会第36年全国大会p153(1988)