並列推論マシン PIE64 の相互結合網の作成および評価

高橋 栄一　　小池 汎平　　田中 英彦

{eiichi,koike,tanaka} @ mtl.t.u-tokyo.ac.jp

東京大学 工学部

概要

PIE64 は、記号処理を高速実行することを目的として開発を行なっている並列マシンである。PIE64 のアーキテクチャの特徴は、64 台のプロセシング・エレメントを、2 系統の均質な相互結合網で結合している点である。相互結合網は、並列計算機アーキテクチャにおける基本的なチョイスポイントのひとつであり、その性能や特性は、システム全体の処理能力と処理方式に直接反映する。本稿では、(1) 回線交換、(2) 多段網、(3) 動的負荷分散支援などの特徴を有する PIE64 の相互結合網の構成について検証し、相互結合網ハードウェアの実装方法の検討を行なう。

1　はじめに

PIE64[1] は、記号処理を高速実行することを目的として開発を行なっている並列知識処理マシンである。PIE64 が有する並列計算機アーキテクチャ上の大きな特徴は、プロセシング・エレメントである 64 台の推論ユニットを、2 系統の均質な相互結合網 [2] で結合している点である。

相互結合網は、並列計算機アーキテクチャにおける基本的なチョイスポイントのひとつであり、その性能や特性は、システム全体の処理能力と処理方式に直接反映する。

我々は、PIE64 の相互結合網として、

- 回線交換
- 多段網
- 動的負荷分散支援 [3]

などの特徴を有する相互結合網を 2 系統用意した。

相互結合網の実現には、構成単位となるスイッチングエレメントをゲートアレイで作成し、その LSI チップを用いて 64 ポート × 64 ポートの多段網を構成した。多段網は構造的には規則的であるが、他のトポロジと比較して配線パスが多くかつ集中しており実装が容易ではない。この点に関し PIE64 では、実装方法を工夫して解決した。

現在、PIE64 のハードウェアの開発は、完成した相互結合網ハードウェアのテストが終了し、推論ユニットの作成が進んでいる段階である。相互結合網のテストは、新たに開発したテスト用ハードウェア「タコ」[4] を用いることにより、効率的で実際の動作状態に近いレベルでの動作チェックとデバッグを行なうことができた。

本稿では、PIE64 の相互結合網の構成を検証し、相互結合網ハードウェアの実装方式の検討を行なう。

2　並列推論マシン PIE64

PIE64 は、Committed Choice 型言語 Fleng[5] およびオブジェクト指向言語 Fleng++[6] により記述された大規模知識処理ソフトウェアの高速実行を目的とする並列記号処理マシンである。PIE64 では並列処理技術の基本は相互結合網にあるものと考え、強力な相互結合網により多数台のプロセッサを結合するタイプのアーキテクチャを目指して研究開発を行なってきた。

Implementation and evaluation of interconnection network of PIE64

Eiichi TAKAHASHI, Hanpei KOIKE, Hideliko TANAKA

University of Tokyo, Faculty of Engineering.

PIE64のアーキテクチャ上の特徴は、推論ユニットと呼ぶ64台のプロセシングエレメントを、同一構成を持つ2系統のネットワークからなる相互結合網で結合している点である。PIE64の全体構成を図1に、相互結合網(1系統分)を図2に示す。

推論ユニットの構成を図3に示す。各機能ブロックは(カッコ内は図3中ので名称)、

- **ユニファイア / リデューサ (UNIRED)**[7]
  Flengの実行の中心であるユニフィケーション、リダクションを高速に処理する
- **ネットワークインタフェスプロセッサ (NIP)**[8]
  相互結合網を介して2台の推論ユニット間で行なわれるリモートデータアクセスなどの並列処理を支援する通信制御ハードウェアである
- **管理プロセッサ (SPARC)**
  負荷分散や実行のスケジューリングなどのゴール管理、システム述語の実行、分散ガベージコレクションを担当する
- **ローカルメモリ (Local Memory)**
  バンク分けされ管理プロセッサおよびUNIRED、NIPからアクセス可能なメモリである
- **ホストインタフェス (Host I/F)**
  ホストプロセッサであるワークステーションとPIE64とのインタフェースを行なう
- **I/O インタフェス (I/O I/F)**
  データベースマシンやグラフィックプロセッサなどを接続することを想定したインタフェースである

という機能を持つ。

3 PIE64の相互結合網

まず、PIE64の相互結合網の2つの大きな特徴である、

- 回線交換
- 多段網

について考察する。

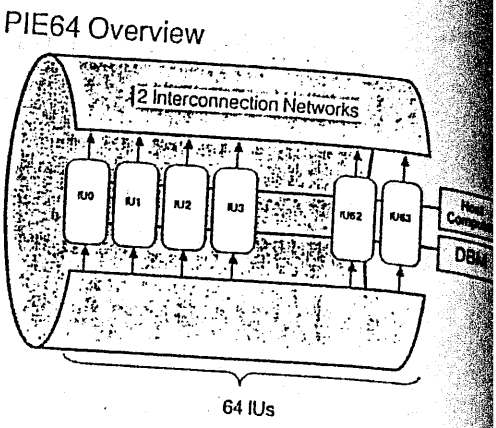


図 1: PIE64のアーキテクチャ

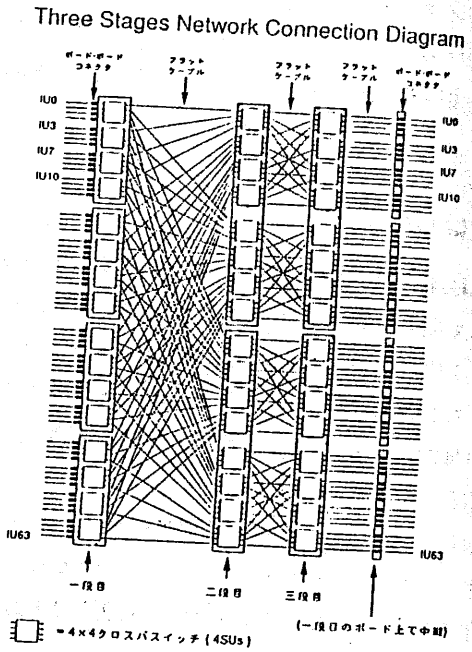


図 2: PIE64の相互結合網

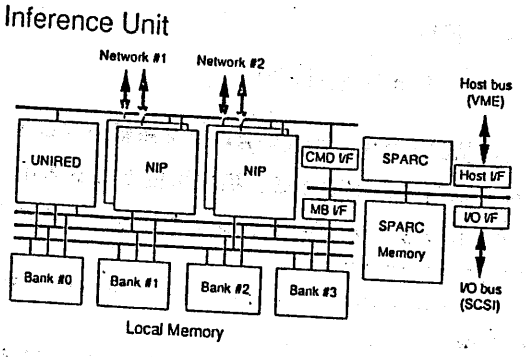


図 3: 推論ユニット

回線交換　PIE64で採用した回線交換方式をオルタナティブであるパケット交換方式と比較した場合、以下のような特徴が挙げられる。

- スイッチエレメントはデータを蓄積しないので、転送遅延は信号伝達遅延時間だけであり、原理的にはクロックや中継回数に依存しない
- 相互結合網を流れるのは経路制御情報とデータだけであり、パケットの順序制御など通信のための付加的な情報によるオーバーヘッドはない
- 機能レベルが原始的であり、相互結合網のレベルでデッドロックが生じることはない
- 接続を保ったままでの双方向通信が可能で、read-modify-write型のアクセスに有利である
- 接続のオーバーヘッドは大きいが、接続後の転送のコストは小さく、大きなデータ転送に有利である
- 同程度のデバイステクノロジを用いた場合、機構が単純なのでハードウェアの小型化、処理の高速化が期待できる
- 開発という観点から見た場合にも、設計およびデバッグが容易である
- 相互結合網を外側から見た場合、機能が原始的である分、通信制御ハードウェアであるNIPによる最適化が期待できる

また、欠点としていくつかの点が挙げられるが、それらに対し次のように対処している。

- 「閉塞網を用いる場合、閉塞状態に対する何らかの対策が必要である」
  通信制御ハードウェアでタイムアウトを検出する。
- 「閉塞網を用いた場合、相互結合網が混雑してくると急激に閉塞率が上がりスループットの低下を招く」
  相互結合網を2系統使用することにより、「耐閉塞性」を高める。　また、ネットワークトラフィックを低減するような負荷分散ストラテジを用いる。
- 「推論ユニット間を結ぶ通信経路の物理的な距離が長くなると、信号伝達路としての品質が低下し、誤り制御などのために実質的な転送速度が低下する」
  推論ユニットは各々の間の物理的な距離が最短になるように配置する。

多段網　同様に、PIE64のネットワークトポロジとして採用したスイッチ結合型の多段網について、ハイパーキューブやメッシュ、クロスバなどの他のトポロジと比較した場合の特徴を挙げる。

- 「階層構造を持たない均質な相互結合網であり、すべてのプロセシングエレメント間の距離が等しい」
  ある種の問題を扱うのに有利な固定的な接続には向いてないので、必ずしも最適ではないが、広範囲な問題を扱うのに適している。　また、負荷分散を考える際の処理モデルが単純化され、動的な負荷分散処理のためのオーバーヘッドを抑えることができる。ハイパーキューブやメッシュでは、動的な負荷分散処理はオーバーヘッドが大きくなり、必ずしも有効性を保証できない。
- 「プロセシングエレメント間の平均距離が小さい」
  2つのプロセシングエレメント間の平均距離(転送経路を構成するアーク数、ただしスイッチノードも中継点と見なす)は、プロセシングエレメント数を n とすると、

  k 次元メッシュ ……$O(\sqrt[k]{n})$ $(k \leq 3)$
  ハイパーキューブ ……$O(\log_2 n)$
  多段網 ……$O(\log_k n)$ $(k = 2, 4, 8, \ldots)$
  クロスバ ……$O(1)$

  であり、多段網は比較的小さい(対数の底 k はスイッチノードのポート数)。特にハイパーキューブと比較した場合、回線交換の多段網ではデータ転送中は途中のスイッチノードでのバッファリングを行なわないため、データの転送遅延は純粋に信号の伝達遅延のみになる。　これに

よりデータの高速転送が可能になるが、 実現のためのハードウェア量、 特にプロセシングエレメントやスイッチ間を接続するパス数が増大する。 クロスバスイッチでは、 実現のためのハードウェア量がさらに増大する。

- 「シングルパスの閉塞網である」
  多段網としては最小のハードウェア量で構成でき、冗長経路を持たないのでルーティングが単純になる利点がある。 しかし、高負荷時に急激に閉塞率が上昇するので、この現象を抑えるような静的 / 動的負荷分散処理が必要になる。特に、ホットスポットの発生によって閉塞率が上昇している場合には負荷分散処理が有効である。

- 「ネットワーク構成 (ノード数) に対し拡張性がある」
  メッシュ、ハイパーキューブ、多段網などは同一のハードウェアを用いてプロセシングエレメント台数のバリエーションに対応することができる。 特にメッシュであれば、接続できるプロセシングエレメント台数に制限はない。 ハイパーキューブや多段網では、プロセシングエレメントの台数を増やすためには、ノードに対して特別な機構が必要となる。

PIE64 相互結合網の付加機能　次に、PIE64 の相互結合網が持つその他の特徴について考察する。

- **動的負荷分散支援 (図 4)**
  通信に使用していない転送線とスイッチを用いてコンパレータを構成し、各推論ユニットの負荷を示すデータをこのコンパレータで比較する。最小の負荷量を報告した推論ユニットを相互結合網が記憶し、この最小値は通信を行なっていない各推論ユニットに配られる。負荷最小の推論ユニットへの接続は相互結合網が行なう。この方式は単純な機能であるが、システム中での負荷の最小値の取得、および最小負荷推論ユニットへの接続を瞬時に実現し有効である。通信に使用していない資源を利用しているので、通信処理へのオーバーヘッドはない。負荷量として

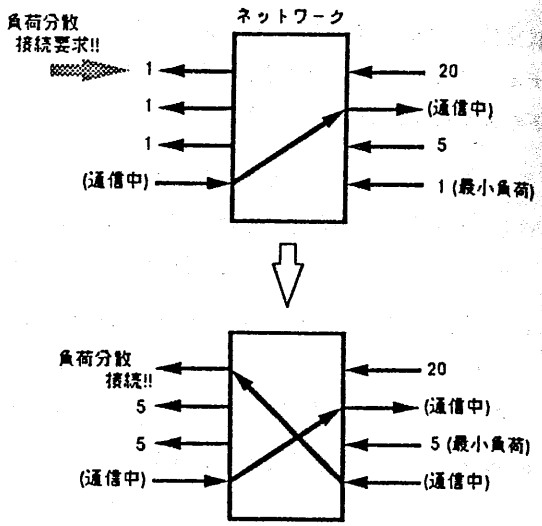


図 4: 動的負荷分散支援機能

どんな値を用いるかがこの機能を利用する上での鍵となる。

- **拡張性 (図 5)**
  ビットスライス構成により、8 ビットずつビット幅を拡張することができる。 また、4 段までの多段構成をサポートしており、最大 256 台の推論ユニットが接続できる。 これにより推論ユニットの台数、および推論ユニット間の転送データ幅に自由度を持たせることができる。

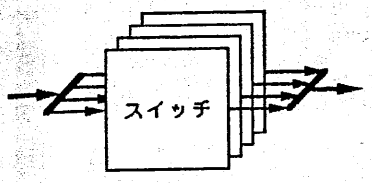
- **二重化された相互結合網**
  2 系統の相互結合網を使用することにより閉塞率を抑えることができる。また、各推論ユニットの負荷情報を 2 次元で表現することができ、よりきめの細かい動的負荷分散支援が可能になる。

4　相互結合網の実装

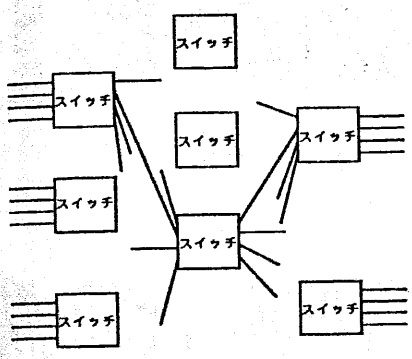
前節では、PIE64 が採用した相互結合網の構成に関して考察を行ない、その妥当性を検討した。その結果、定性的な評価においては妥当性が確認されたものの、定量的な判断が必要な部分についてはある程度実機での評価が必要となると思われる。

本節では、PIE64 の相互結合網の実現について、問題点およびその解決法を述べる。

以下、



(a) ビットスライス構成



(b) 多段構成

図 5: 相互結合網の拡張性

1. 基本構成
2. 推論ユニットの物理配置
3. 相互結合網の実装
4. スイッチノードの配置
5. 組立

の順に説明する。

基本構成　まず、PIE64 の基本構成として、

- 実用規模の並列マシンという目標に従い 64 台の推論ユニットを作成する。これは、むやみに推論ユニット台数を増やすよりも、高性能の推論ユニットを用意してシステム全体としても実用的な性能を達成しようという主旨からである。
- 相互結合網は、閉塞率を抑えスループットを向上するために 2 系統用意する。

を決定した。

特に前節で述べた回線交換の多段網を実現するために、4×4 でデータ幅 8 ビットのクロスバスイッチをゲートアレイ (Switching Unit、SU) を用いて開発した。この SU は、

- 多段構成をサポートする分散型のルータ
- 8 ビットずつのビットスライス構成でデータ幅拡張可能
- 動的負荷分散機能を支援するコンパレータ内蔵
- 内部診断用のスキャンパス

などの機能を持つ。推論ユニット内の処理が 32 ビットで行われるので、相互結合網のデータ幅も 32 ビットとした。

推論ユニットの物理配置　PIE64 の相互結合網はスイッチ結合型であり、周りを推論ユニットが取り巻くような論理構成を有する。従って、物理的にも図 6のように相互結合網を中央に、推論ユニットを周囲に配置するのが合理的である。

相互結合網の実装　相互結合網の基板 (図 7) は、推論ユニットボードのバックプレーンにあたる位置と、これらに対し箱の上面や底面にあたる位置に配置する。2 系統の相互結合網は、同形状に組み立てたものを上下に併置する。また、基板間の接続は基板間コネクタで行うのが確実であるが、

- 基板間で結線すべき信号線の本数が多く、通常の密度のコネクタが使用できない
- 特殊な高密度のコネクタを使用すると、取り付ける部分での基板上の配線が困難になる
- 相互結合網のチェックは全体に対してのみ可能である

また、基板上の配線に対し、

- 信号線への雑音の回り込みやクロストークが問題となりやすい
- 32 ビット分の信号線の遅延特性を揃えるのは困難である

などの問題点が存在する。

これに対し PIE64 の相互結合網では、スイッチ間の接続をすべてフラットケーブルを用いて実現した。この方式は、

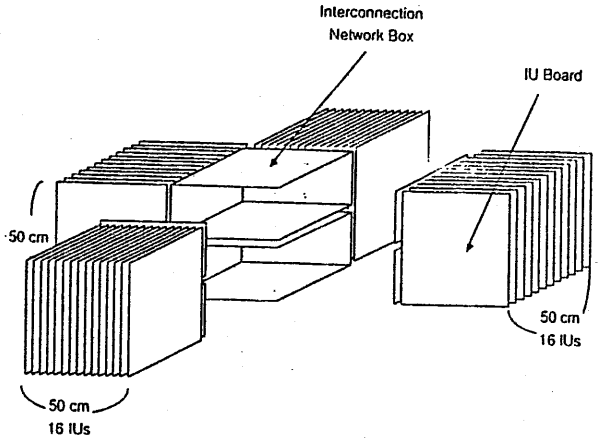


図 6: PIE64 の実装

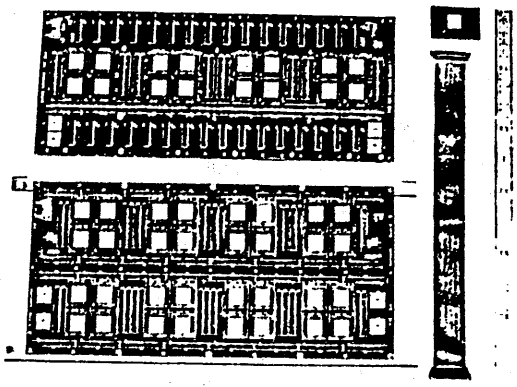


図 7: 相互結合網基板

- フラットケーブルの伝達特性は安定している
- スイッチ間の遅延を全接続に対して揃えることができる
- 各スイッチが独立しているので、チェックしやすい
- スイッチの配置を工夫することにより、 最長のフラットケーブルを最短にすることができる
- 組立に手間がかかる

などの利点を持つ。逆に、接続の正しさや接続部分に対する信頼性が低いという指摘があるが、専用のメンテナンス機構「タコ」(次段落参照) を用いた効率的なテスト環境によりこの問題を解決した。

スイッチノードの配置　PIE64 の相互結合網は、スイッチノード (4 つの SU で構成される 32 ビットクロスバスイッチ) を実装した 6 枚の基板を直方体の箱状に配置し、ステージ間を接続するフラットケーブルをその箱の中に収納する形で実現する。基板上のスイッチノードの配置には自由度があるが、

- 1 段目のスイッチエレメントは推論ユニットボードのバックプレーンとなる側面の基板上に配置し推論ユニットとの接続を固定する
- 必要な最長のフラットケーブルの長さを最短にする
- ステージ間のシャッフルが 4 つに分割できることから、 分割されたケーブル群が入り交じらないようにする

という条件に従い決定する。

図 8: 相互結合網の組み立て

組立　組立 (図 8) には困難が予想されたが、

- 彩色して各ケーブルの論理的な位置を明確にするとともに、 誤配線を防ぐ
- オープンなスペースであらかじめケーブルの配置を決定しておき、 そのままラック内に移動して、コネクタの接続を行う

などの対策を講じた結果、非常に効率よく作業を進めることができた。図 9 に、1 系統分の相互結合網が組上がった様子を写真で示す。

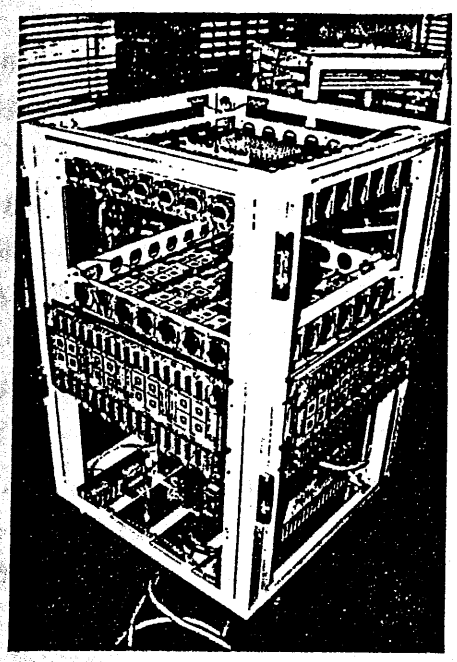


図 9: 相互結合網 (完成時)

5 相互結合網のデバッグおよびメンテナンス

PIE64 の相互結合網のテストやデバッグを行う方法について述べる。

相互結合網はスイッチエレメントごとにテストすることができるが、

- 基本的にスイッチ素子であるので監視すべき端子数が多い (1 ボート 38 本 × 8 ボート)
- 32 ビット幅のデータ線は双方向端子である
- 実働状態と同じ 10MHz のクロックを使用してのテストが必要

などの理由から、通常のロジックアナライザではテストが困難である。

そこで、PIE64 のホストインタフェースおよびクロックジェネレータを組み込みスイッチエレメントのテストに必要な 38 ビット × 8 本のプローブを持つ PIE64 の相互結合網専用ハードウェアデバッガ「タコ」を開発した。

この「タコ」は、

- 推論ユニット内の NIP の機能を一部エミュレーションすることができ、これにより 10MHz のクロックで相互結合網を駆動することが可能となる
- 相互結合網上を流れるデータをモニタする

という 2 つの機能を持ち、これにより

- SU チップのテスト
- スイッチエレメントのテスト
- フラットケーブルのテスト
- 相互結合網基板のテスト
- 多段接続状態でのテスト
- 相互結合網のモニタ

などに使用できる。

タコの操作はワークステーション上で対話的に行なうことができるように X ウインドウ上に相互結合網のテスト支援環境を作成した。

実際のテストは、

- ホストから対話的にタコを動作させ、基本的な機能を確認する
- バッチ的に動作させ、システマティックに全状態のテストを行う

という手順で、非常に効率よくテストを行うことができた。

6 おわりに

本稿では、PIE64 の相互結合網の構成について概説し、ハードウェアの実装方式の検討を行なった。

まず、PIE64 の相互結合網の構成とその特徴である

- 回線交換
- 多段網
- 2 系統
- 動的負荷分散支援

について考察し、PIE64 の相互結合網として適切であることを検証した。

次に、PIE64 の相互結合網の実装方式である

- 相互結合網を中央に配置する
- 多段網のステージ間接続にフラットケーブルを用いる

について検討し、その実現の過程を報告した。

最後に、相互結合網のテスト、デバッグ、メンテナンス方式について述べ、相互結合網のテスト用ハードウェア「タコ」を用いた効率的なテストについて紹介した。

今後の課題として次のような項目が挙げられる。

- 「信号伝送路としての特性評価」
  信号伝送路として、ビット化けやクロストークなどのエラーがフォールト発生する確率を長時間のランニングテストなどで評価する。
- 「データ転送能力に対する定量的な評価」
  実用的な規模のプログラムの実行中に起こるリモートアクセス特性を考慮したデータ転送能力の評価を、動的負荷分散の戦略などと関連付けて行なう。

謝辞

相互結合網の構成要素となるゲートアレイ、SUの開発に当たり、多大なる御支援を賜わった富士通研究所 人工知能第三研究室の服部室長、並びに、久門氏、三宅氏に深謝いたします。

また、ハードウェアの作成やチェックに協力して頂きました研究生の松本氏(日本IBM)と宮木氏(日立)、PIE64の相互結合網作成に当たり、必ず議論に参加し多くの貴重なアイデアを提供してくださった修士2年の日高君他、修士1年の中田君と毛利君、その他田中研究室の皆さんに感謝します。

さらに、相互結合網基板を作成して下さいました日立化成工業株式会社の山林氏、PIE64のラックの設計および作成、「タコ」の基板の作成を行なって下さいましたヨシキ電子株式会社の中野氏、橋本氏に感謝致します。

なお本研究は，文部省特別推進研究No.62065002の一環として行なわれている。

参考文献

[1] 小池 汎平, 田中 英彦: "並列推論エンジン PIE64", 並列コンピュータアーキテクチャ, bit 臨時増刊, Vol.21, No.4, 1989, pp. 488-497.

[2] Koike H., Takahashi E., Yamauchi T. and Tanaka H.: *The High Performance Interconnection Network of Parallel Inference Machine PIE64*, Computer Architecture Symposium IPS Japan, 1988.

[3] 坂井, 小池, 田中, 元岡, "動的負荷分散を行なう相互結合網の構成", 情処論, Vol.27, No.5, 1986.

[4] 日高, 高橋, 小池, 清水, 田中, "PIE64 のネットワークメンテナンス, ホストインタフェース, クロック分配機構: タコ", 第 40 回情報処理学会全国大会, May 1990.

[5] Nilsson, M. and Tanaka, H.: *Fleng Prolog - The Language which turns Supercomputers into Prolog Machines.* In Wada, E.(Ed.): Proc. Japanes Logic Programming Conference. ICOT, Tokyo. June 1986. p209-216. Also in Wada, E.(Ed.): Logic Programming '86, Springer LNCS 264. p170-179.

[6] 中村, 小中, 田中, "並列論理型言語 FLENG に基づいたオブジェクト指向型言語 FLENG++", 日本ソフトウェア科学会, オブジェクト指向計算に関するワークショップ WOOC'89, 1989.

[7] 島田, 下山, 清水, 小池, 田中: "推論プロセッサ UNIREDII の命令セット", 計算機アーキテクチャ研究会 79-5, 情報処理学会, Nov. 1989.

[8] 清水, 小池, 田中: "並列推論マシン PIE64 の推論ユニット間通信", 計算機アーキテクチャ研究会 79-4, 情報処理学会, Nov. 1989.