

並列オブジェクト指向トータルアーキテクチャ A-NET

6P-8

— ルータの設計 —

茂木 久 吉永 努 馬場 敬信

(宇都宮大学工学部)

1. はじめに

我々は、並列オブジェクト指向概念に基づいたトータルアーキテクチャ A-NET (Actors NETWORK)^[1,2]を開発中である。A-NET 計算機は多数のノード(要素プロセッサとルータの対)が、ネットワーク状に結合された MIMD 型の並列計算機である。ノードの構成要素であるルータは、ネットワークでの通信を担当する。また異なる応用分野に対して、適切なネットワークトポロジを選択できるように、ルータはトポロジに独立である必要がある。

本稿では、ルータの設計方針やハードウェア構成を中心に述べる。

2. 設計方針

A-NET 計算機におけるルータの設計方針を以下に示す。

1) トポロジ独立性の実現

ルータにおいて、ネットワークトポロジに依存する部分は、メッセージの経路選択と動的オブジェクトの生成先を決定する部分である。特に経路選択の部分は、トポロジ変更の際の柔軟性と共に処理の高速性を要求されることから、その機能をいかに実現するかが問題となる。

A-NET 計算機で採用を予定しているトポロジは、2次元、3次元のメッシュ、トーラス、バイナリ n キューブ、2分木、4分木、X トリーなどである。ルータは、これらのトポロジを実現するのに必要な6個のノード間通信用ポートを持つ。また、その他のトポロジについても、ポート数の制限を満たす範囲で構成できる。

2) 高い通信能力の実現

プログラム実行の高速化のためには、ネットワークの通信能力が重要になる。高い通信能力を得るための方法を以下に示す。

① 通信路の性能

8ビット幅の並列伝送を行い、20MB/secの転送速度を目標とする。

② データ処理時間の短縮

パケットの経路選択、転送をハードウェアで制御することにより、メッセージのネットワーク通過時間を短くする。

③ 並列動作の実現

通信路の重複しない複数のデータ移動、データ処理等を並列に実行し、性能向上を図る。

3. 機能

ルータの接続する通信路には次の2種類がある。

・ノード-ノード間結合

プログラム実行時のオブジェクト間でのメッセージ交換やオブジェクトの動的生成に使用される。

・ホスト-ノード間結合

システム初期化、オブジェクトの静的割り付け、入出力やデバッグ等に用いられる。

ルータの機能を、メッセージ交換とオブジェクト割り付けに大きく分けて、以下に説明する。

1) メッセージ交換

メッセージには、A-NET L^[3]で記述される通常メッセージと、システム初期化やルータのデバッグ時に使用されるルータ用メッセージがある。共にメッセージサイズが16語を超える場合はパケット分割される。このパケット最大サイズは、複数のA-NET Lプログラムの実行結果から、多くのメッセージがパケット分割を行う必要が無い大きさを選択したものである。

ルータ用メッセージは、初期設定用とプログラム実行時に分けられる。システムの初期化時には、トポロジの設定メッセージ、ローカルOS^[4]のロード開始メッセージ、ユーザオブジェクトの静的割り付け開始メッセージ、ノードの接続確認用メッセージなどがあり、ホストから各ルータに送信し、プログラムの実行準備を行う。ここで接続確認用メッセージとは、ホスト側から任意のノードの各ポートの隣接ノードID、あるいは未接続の情報を求めるためのメッセージである。全ノードについての情報を得れば、ホスト上で実際のネットワークの状況を把握でき、トポロジ変更時のネットワークの接続誤りによるプログラム実行の異常を回避することができる。プログラム実行時のルータ用メッセージとしては、ルータデバッグ用メッセージ、パケット再送のためのメッセージなどがある。デバッグ用メッセージにより、ルータ内の各種情報の確認、設定などを行う。特に、各ルータ内に存在するパケットを取り出すことで、ネットワーク内の全メッセージや、パケットの移動状況が確認できる。

2) オブジェクト割り付け

オブジェクトコード転送時の局所メモリ-共有メモリ間でのデータ移動、パケット分割等のオーバーヘッドを取り除くため、オブジェクト生成には回線交換方式を用いる。このため、オブジェクトのDMA転送に先立ち転送路を確保する必要がある。転送路を設定することで、中継ノードで経路選択する必要やバッファリングする必要がなく速やかに転送できる。

オブジェクトの割り付けには、静的なもの動的なものがある。前者は、ホストからノードに対する、ローカルOSとユーザオブジェクトの割り付けであり、

後者は、オブジェクトを動的に生成できるクラス⁽³⁾の存在するノード(クラスノード)から生成先ノードに対する割り付けである。動的オブジェクトはクラスノードの近傍から最初に生成させる。これは、生成依頼元オブジェクトとクラスと動的オブジェクトの間のメッセージ交換の局所性や、オブジェクト転送の際の通信コストを考慮したためである。このため、処理系は、ユーザオブジェクトを静的に割り付ける際、クラス周辺のノードに、余裕を持たせて割り付けておく必要がある。

4. ハードウェア構成

ルータの各機能を実現するためのハードウェア構成を図1に示す。

①通信制御装置

クロスバネットワークのスイッチング制御を行う。また、クロスバネットワークにつながる各機能部との、データ転送や経路選択などの要求の調停を行う。経路選択は、宛先ノードのID(16bit)から、送出可能なポート、あるいは自ノード宛であるかの情報を求めることで行う。転送パケットは、送出ポートのうち、現在使用していないポートから送出し、使用可能ポートがなければ、パケットバッファに格納する。

現在、プログラム可能な機能素子を用いて、ハードウェアでメッセージの経路選択とオブジェクトの生成先決定の機能を実現することを検討している。

②メッセージ送信装置

自ノードのPEが共有メモリの出力キューに書き込んだメッセージを、経路選択し、宛先ノードに送出する。もし16語より大きければ、パケット分割を行う。

③メッセージ受信装置

自ノード宛のパケットを受信して、共有メモリの入力キューに書き込み、PEに割り込みをかける。分割されたパケットならば、共有メモリ上の受信バッファでメッセージに再構成して入力キューに格納する。また、動的オブジェクト生成のための問い合わせメッセージならば、自ノードに割り付け可能かどうかの確認を行い、その結果をクラスノードに返答する。その際、割り付け可能ならばPEを止めて、クラスノードからのオブジェクト転送を待つ。

また、システム設定用メッセージ、ルータデバッグ用メッセージなどの、ルータ内で処理すべきメッセージの解釈実行を行う。以上のように、メッセージ受信装置は、比較的処理が複雑であり、その動作に柔軟性を持たせる必要があるため、マイクロプログラムで実現する。

④DMAコントローラ(DMAC)

オブジェクトの転送の際、クラス側ノードのDMACは局所メモリからポートへ送出し、生成先ノードのDMACはポートから局所メモリにオブジェクトを格納する。中継ノードのDMACはこのオブジェクト転送には関与しない。

⑤ポート

各ポートは8ビットのデータにパリティを1ビット付加し、3線ハンドシェイク方式の並列伝送を行う。

また、1パケットを格納するためのバッファを持つ。オブジェクト割り付けのためのDMA転送の際には、データは、バッファには格納されずに、予め設定されていた転送路を移動する。

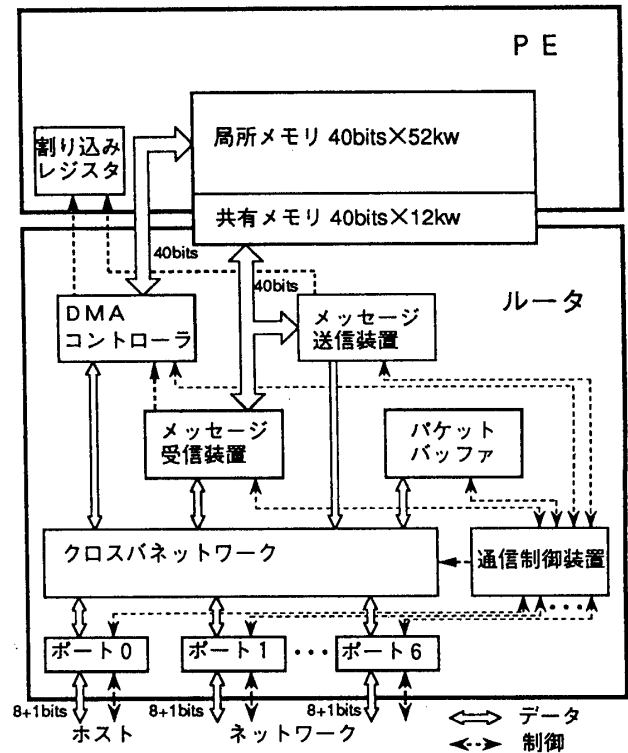


図1 ルータのハードウェア構成

5. おわりに

以上A-NET計算機のルータの設計方針、その機能を実現するためのハードウェア構成について述べた。

今後、ハードウェアの詳細設計とルータのデバッグ用メッセージの確定、ノード数が増加した場合のネットワークの構成方法やトポロジ可変性の実現方法などを検討し、本年度中に2ノードのプロトタイプを製作したい。

参考文献

- [1] 馬場 他: "並列オブジェクト指向トータルアーキテクチャA-NET", 並列処理シンポジウム JSPP '89, A4-1(1989)
- [2] 吉永 他: "並列オブジェクト指向トータルアーキテクチャA-NETの開発方針", 情報処理学会第40回全国大会, 1L-3(1990)
- [3] 岩本 他: "並列オブジェクト指向言語A-NET Lの言語処理系", 情報処理学会第38回全国大会, 4P-1(1989)
- [4] 吉永 他: "並列オブジェクト指向トータルアーキテクチャA-NETのローカルOS", 情報処理学会第45回オペレーティングシステム研究会報告, 89-OS-45-5(1989)