

5W-1

Datarol プロセッサの設計

上田 哲郎 谷口 倫一郎 雨宮真人
(九州大学 総合理工学研究科)

1 はじめに

純粹な関数型言語は、関数モジュールが明確になっており、副作用が存在しない。したがって関数型言語のプログラム構造は、高並列の処理を記述する上で特に魅力的である。

関数型言語の高並列性に注目し、これを効率的に実行できるマシンとしてデータフローモデルに基づいた並列処理コンピュータのアーキテクチャがこれまでもいくつか提案されてきたが、実際に、ハードウェアにインプリメントするには、種々の問題を抱えている。筆者らは、新しいデータフローモデルのアーキテクチャとして Datarol プロセッサアーキテクチャを提案している。これは、Datarol と呼ぶマルチスレッドコントロールフローをコンパイラによって抽出し、それに沿って並列処理実行を制御する機構を有するプロセッサである。本論文では、Datarol マルチプロセッサについて、そのネットワーク構成を中心に述べる。

2 Datarol プロセッサの構成及び制御機構

Datarol プロセッサの基本構成を図1に示す。

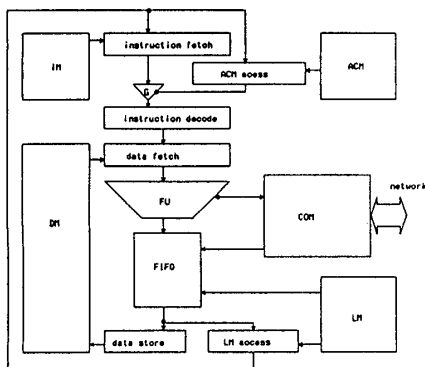


図1 Datarol プロセッサの構成

Datarol プロセッサは、循環パイプライン上に構成されたステージ間でのパケット交換という形で実行制御を行う。data memory(DM)は、およそ1k個のレジスタファイルからなり、関数活性体毎に一つのレジスタファイルを割り当てる。各命令には、作業環境を区別するために、このレジスタファイル名(instance)が付加される。instruction memory(IM)は Datarol code を保持している。link memory(LM)は、次命令を取り出すためのリンケージ情報を保持している。発火制御メモリACMは、ペアオペランドチェックを行うためのメモリである。FUは命令の実行部であり、その実行結果はFIFOキューに入る。communication unit(com)は、processing element(PE)と interconnection network とのインターフェースである。

3 Datarol マルチプロセッサ

プロセッサは、interconnection network に接続されている。またプロセッサは、複数のクラスタに分けられる。各クラスタは、同一の関数本体を有しており、クラスタ内では自動負荷分散を行う。また、関数本体は、予め異なったクラスタ間に分散されているため、プログラムで明示的に他のクラスタの関数本体を呼び出すことにより、クラスタ間での、負荷分散を実現する。

3-1 コミュニケーションユニット

他PEとの通信は、com を介して行われる。FUが com へ処理を任せると、そのレスポンスが返ってくるまでPEは、他のスレッドを扱うことが可能である。

他PEに処理を任せるために次の命令が用意されている。i)他PEへの newins(nins) の要求(call命令), ii)nins要求を受け取ったPEからの nins の送り返し, iii)引数の受渡し(link命令), iv)結果値の送り返し(return命令)。以上4つの場合に各々 control code を割り当てる。i)の場合は、比較的負荷の小さいPEに要求が発せられる必要(自動負荷分散機能)があるが、この機能はネットワークが有する。com は send ユニットの、receive ユニットの、退避用レジスタファイルからなる。退避用レジスタファイルは、call命令のレスポンスとして、nins が返されるまで、call命令のパケットが持っていた情報を退避させるためのものである。send receive 各ユニットは、入力パケットの control code にしたがって処理を行う。

3-2 ネットワーク

ネットワークに要求される機能は、

- 可変なクラスタ構造を実現できる
 - クラスタ内のPEに負荷分散を行う
 - 任意のPE間での接続が可能である。
- の3点である。またPE間の通信要求は、それぞれのPEで任意の時間に発生する。

以上のことから、Datarol マルチプロセッサに用いるネットワークとして、多段の閉塞網を用い、網内の各スイッチノードに上の要求を満たすための機能を追加した。

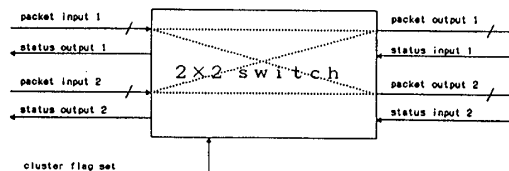


図2 パケット交換スイッチ

ネットワークには、図2の様な2*2の交換スイッチを用いる。各スイッチは、転送パケットのペケットアドレスに従って、交換と直通の二つの、状態があり、2つの入力パケットの出力ポートでの競合は、スイッチ内の待行列で対処する。スイッチの詳しい構成を図3に示す。

スイッチ内の各ブロックについて説明する。

i) status generatorは、その出力ポートに接続されている2つのスイッチから、ステータス情報を受け取って、自スイッチのステータスを生成し、その入力ポートに接続されている2つのスイッチにステータス情報を送る。ステータス情報とは、そのスイッチに接続されている、PEの負荷の状態を伝搬したものである。ステータスは<無、青、黄、赤>、の4つの状態を持つ。無とは、そのPEが使用されていない状態を示しており、その後の色は、PEの負荷の程度を示している。もちろん赤の方が、PEの負荷は大きい。生成されるステータスは、入力ステータスの、負荷の小さい方になる。

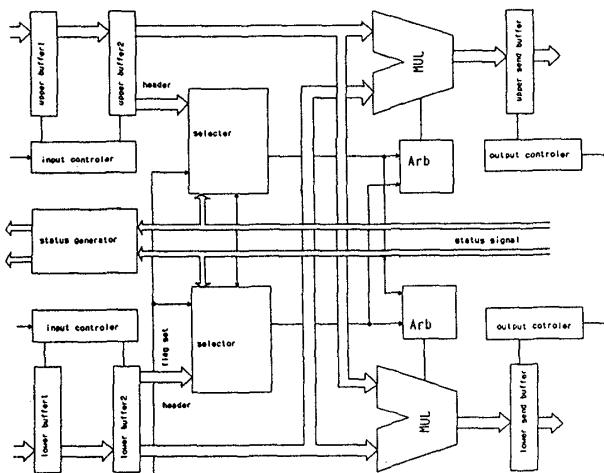


図3 パケット交換スイッチの構成

ii) selectorは、そのスイッチが、クラスタ内のスイッチであるかどうかを示すフラグを持っている。このフラグは、プロセッサにプログラムをロードする際にセットされる。負荷分散を必要としないパケット転送では、入力パケットのペケットアドレスを解釈して、目的の出力ポートのarbiterに、send要求を出す。負荷分散を必要とする場合は、①クラスタ内を示すフラグが立っている場合は、入力されたステータスにしたがって負荷の小さい側の出力ポートを選択する。入力されたステータスが同じ場合には、出力ポートの選択が偏らないように、フリップフロップスイッチを用いて、2つの出力ポートを交互に選択できるようにする。②クラスタ外の場合は、パケットアドレスにしたがう。selectorは、二つの入力ポートにそれぞれ用意されており、二つのselector間で情報を交換して、できるだけ出力ポートが競合しないように調停する。

iii) arbiterは、それぞれの出力ポートにあり、sel

ectorからの、send要求を受け取る。二つの要求が競合した場合には、調停を行う。このときもやはりsend要求の受領が一方に偏らないようにフリップフロップスイッチを用いて、交互に要求を受け取るようになっている。

例えば図4のように4つのPEで1クラスタを形成している場合を考える。PE0, PE1, PE2, PE3の負荷の程度がそれぞれ青、黄、無、無であった場合、スイッチ(1,0)のステータスは青、スイッチ(1,1)のステータスは無、になる。これらのスイッチからステータスを受け取るスイッチ(0,0)、スイッチ(0,1)のステータスは、共に無となる。

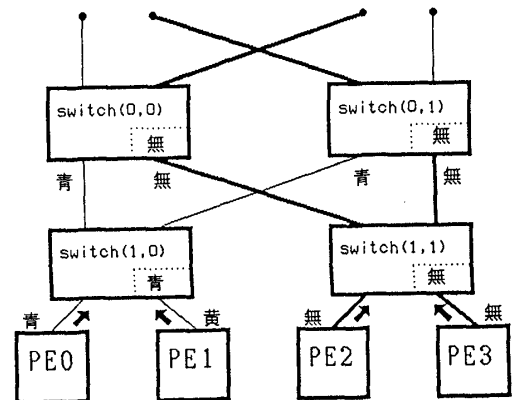


図4 ネットワークの動作

今、スイッチ(0,0),(0,1)に負荷分散モード(cc=0)でのパケット入力があった場合、共に負荷の小さいスイッチ(1,1)へパケットをsendする。スイッチ(1,1)では、パケットの一方をPE2へ、他方をPE3へ送る。PE2,3のステータスは、青に変わり、これを接続されているスイッチへ伝搬する。このようにして、クラスタ内の負荷の小さいPEが選ばれ負荷の分散が可能である。

4 今後の課題

今後は、ソフトウェアでDatarolプロセッサのシミュレータを記述し、精密な性能評価を行う。シミュレーション内容は、insの使用量、FIFOキューの詰まり具合、プロセッサ間通信の頻度、ネットワークによる負荷分散の確認、通信オーバーヘッドの全体の処理に及ぼす影響などである。さらに、以上の、シミュレーション結果を踏まえた上で、要求されるマシンサイクル、ネットワークサイクルを設定し、実際のハードウェアの設計、実装を行う予定である。

参考文献

- [1] 雨宮真人 "超多重並列処理のためのプロセッサアーキテクチャ" 「コンピュータアーキテクチャ」シンポジウム 昭和63年5月
 [2] 上田哲郎等 "datarolプロセッサアーキテクチャについて" 情報処理学会第38回全国大会 平成元年3月