

7E-3

連想プロセッサDREAM-Ⅱの性能評価

後藤 厚宏 田中 英彦 元岡 達

(東京大学 工学部)

1. はじめに

本研究の目的は、“計算機ハードウェア・アーキテクチャのレベルから非数値処理に適合したシステム”の構成法を検討することにある。

本研究では、現在の非数値処理における問題を、従来の汎用計算機アーキテクチャの質的不適合と量的不適合としてとらえた。質的にも、量的にも非数値処理(特に二値図形処理)に適合したシステムとして、多重モジュール構成連想プロセッサDREAM-Ⅱと汎用ホスト計算機とを結合した連想処理システムを提案し、システムを実際に製作した。

[1]

本報告では、二値図形処理への応用[2][3]における試作システムの性能について述べ、本アーキテクチャの評価をまとめる。

2. DREAM-Ⅱのアプローチ

DREAM-Ⅱのアプローチは以下のとおりである。

- ・応用向き専用ハードウェアの集合によって小規模処理モジュール(連想モジュール)を構成する。
- ・多数の処理モジュールを一次的に配置し、各モジュールの制御はマイクロプログラムレベルのSIMD制御とする。
- ・応用に合わせてDREAM-Ⅱの規模を拡張できるような柔軟性のあるシステム構成とする。
- ・処理アルゴリズムは書きかえ可能なマイクロプログラムで記述し、問題向きシステムの構成を容易にする。
- ・汎用ホスト計算機に結合することによって容易に連想処理システムが構成でき、ホスト計算機における非数値処理の負荷を軽減する。
- ・ホスト計算機からの指示は高レベルのコマンドとし、そのDREAM-Ⅱ内での処理をホスト計算機から独立させる。

3. 二値図形処理におけるDREAM-Ⅱの性能

(1) スループットの測定

二値図形の窓処理(輪郭抽出)について、DREAM-Ⅱのスループットを測定した。

本システムでは、ホスト計算機主記憶中に専用デ

ィスプレイ用のバッファ領域(256×192)をとっている。DREAM-Ⅱが処理する二値図形はこのバッファ領域にあり、DREAM-Ⅱは図形データを少しづつ取り出しながら処理を進め、処理結果をバッファに返す。

DREAM-Ⅱでは、各連想モジュールに対する1ワード(16bits)ずつのデータ入出力と、モジュール内での1ワードの処理とをオーバーラップさせることができる。3×3の窓処理の場合、画面の1ラインを処理するには、その上下の2ラインがあればよい。そこでDREAM-Ⅱの連想モジュール・アレイにおけるデータ入力、処理、出力をパイプライン化することによって、DREAM-Ⅱのスループットをホスト計算機側のバスのデータ転送速度まで高めることを目指した。

各モジュールに対して[図1]のように画面を分割した場合のモジュール数とスループットの関係を[図2]に示す。ここでPは窓処理の複雑さを示すパラメータであり、スループットの上限とは、単にバッファとDREAM-Ⅱの間でデータを往復させた場合の転送速度である。

ホスト計算機バスの転送速度は約0.7MB/secであり、スループットが最大になった時の処理時間は1画面(6KB)当たり約19msecであった。単にデータを往復させるのに必要な転送時間は約17msecである為、ほぼデータの転送時間内に処理が実行できていることがわかる。

スループットが最大(飽和)値に達するに必要なモジュール数は処理アルゴリズムによって異なるが、4近傍輪郭抽出でおおよそ2台、8近傍輪郭抽出において3~4台である。連続して数種の処理を実行した場合、及び多近傍処理の場合でも、4~16台あれば最大スループットが得られることがわかる。

(2) マイクロ命令の使用頻度

幾つかの処理プログラム[2]について、マイクロプログラムのステートメント数、及び実行ステップ数を調べた[図3]。これにより

- ・二次元アクセス記憶(TDAM)・ビット処理ハードウェアの利用率が高いこと
- ・制御命令の割合は全体の15%程度であり、大きなオーバーヘッドとはなっていないこと

等がわかる。

4. 評価・検討

本システムでは、

- ・連想モジュール内の二値図形処理向き機能
(二次元アクセス記憶、ビット処理ハードウェア)
 - ・図形データの連想モジュールへの効率のよい負荷分割
(ホスト計算機に対するワード単位の二次元アクセス機能、スキャン・モードによるデータ転送)
 - ・連想モジュールにおけるデータ処理とデータ転送とのパイプライン化
(処理と転送のオーバーラップ機能・柔軟性のあるモジュールの制御)
- により、高いスループットを得ることができた。

残された課題について以下に述べる。

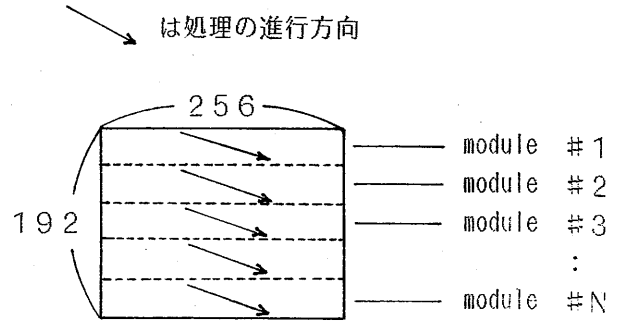
DREAM-11では、連想モジュール内でのデータ操作；データ転送と処理のパイプライン化；モジュールのアクティビティ制御等、処理アルゴリズムの全てを同一レベルのマイクロプログラムで記述している。これはマイクロプログラムの開発に職人芸的な慣れを必要とした。

より高度な処理を可能とする為にも、処理プログラムの記述に階層性を取り入れる必要がある。

<文 献>

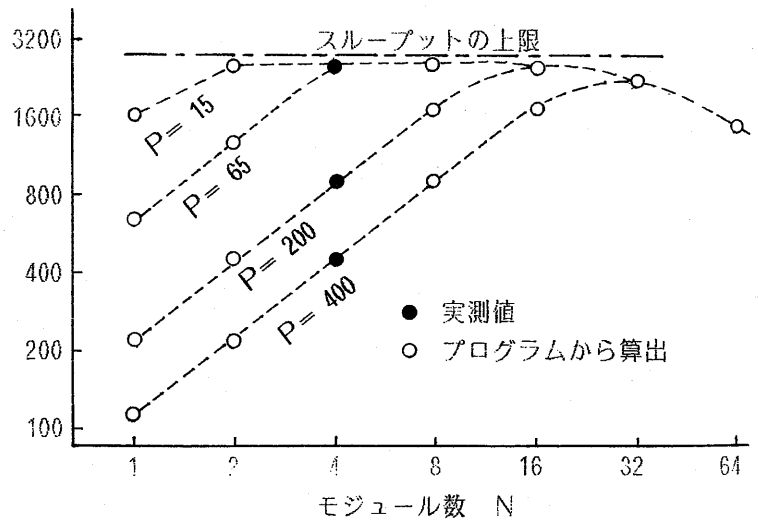
- [1]. 後藤 他、“多重モジュール構成連想プロセッサDREAM-11のハードウェア”，信学技報EC79-64

- [2]. 藤森 他、“連想プロセッサによるLSIマスクパターン傷検出システム”，第23回情報全大
- [3]. 大金 他、“手書き文字認識前処理アルゴリズムと連想プロセッサによるその評価”，第23回情報全大



[図1] 負荷分割

スループット (bits/msec)



[図2] スループットとモジュール数

実行ステップ数 カッコ内 ステートメント数	連想モジュール・アレイ			システム制御部	
	TDAM アクセス	ビット処理 ハードウェア	レジスタ演算 その他	Jump JSR	その他
平滑化・ 輪郭抽出	30% (22)	0.5% (2)	53% (47)	12% (14)	4.5% (7)
屈折点抽出・ 傷判定	20% (29)	22% (19)	44% (92)	10% (39)	4% (14)

[図3] マイクロ命令の使用頻度