計算機アーキテクチャ 116－3
（1996．1．26）

共有メモリ型並列計算機アーキテクチャ WOSMに関する考察

荒川 修
oarakawa@kanagawa.hitachi.co.jp
鳥取大学大学院
(株)日立製作所 汎用コンピュータ事業部
川村 尚生, 井上 倫夫, 小林 康浩
鳥取大学工学部知能情報工学科

従来の共有メモリ方式に比べ，より多くのプロセッサが接続可能なWOSM(Write Only Shared Memory)方式の並列計算機について検討を行った．メインメモリを分散しているため，小規模な結合網でプロセッサに対し十分なデータを供給できる．分散されたメモリを共有メモリに見せるため，結合網のリードアクセスパスとライトアクセスパスを分離し，ライトアクセスパスにマルチキャスト機能を設ける．見かけ上全てのプロセッサから共有メモリが等距離に見えるため，データ分割を最適化することなく一定の性能が得られる．

Discussion of WOSM : Parallel Processor Architecture of Shared Memory Type

Osamu Arakawa
oarakawa@kanagawa.hitachi.co.jp
Tottori Univ. JAPAN
Hitachi Ltd.
Takao Kawamura, Michio Inoue, Yasuhiro Kobayashi
Tottori Univ. JAPAN

We discuss WOSM (Write Only Shared Memory) that can connect more processors than those of the ordinary shared memory type. WOSM supplies enough data to processors with minimum interconnection, because WOSM has the distributed main memories. To give access to the distributed main memories like as a shared memory, we separate the read access paths and the write access paths, and we add a function of the multiple casting to the write access paths. The distributed shared memories are uniformly shared by all processors. Thus, the performance does not depend on the data allocation.

1. はじめに

RISC技術の登場によって年々飛躍的にマイクロプロセッサの性能が向上している．このようなプロセッサを使用し，極限まで高速演算能力を追求したハイエンド計算機の需要も多く，超並列計算機に期待が寄せられている．

しかし，超並列計算機よりもむしろ安価に購入でき，容易に使用できるコンピュータを望むユーザーも多い．汎用のRISCプロセッサを数十個結合した高並列計算機がこれにあたる．このような背景から，超並列計算機の研究だけでなく，RISCプロセッサを数十個接続した低価格で高性能な高並列計算機に対する研究の必要性が増している．

ユーザーが容易にプログラミングができる並列計算機の方式として，共有メモリ方式がある．シングルプロセッサシステムのマルチプロセスのプログラミングと同様にプログラミングでき，MPI等のメッセージパッシングの知識は不要である．4台

程度のプロセッサであればハードウェアも容易に実現できるため，多くの共有メモリ型並列計算機が製品化されている．しかしこの方式は，プロセッサの接続台数を増加し，より高性能にすることが難しい．多数のプロセッサから1つの共有メモリへアクセスが集中するため，共有メモリのスループット不足が発生し，性能向上の妨げとなるからである．このため，リニアに性能向上可能なプロセッサ結合台数は限られる．システム性能をリニアに向上しつつ，接続台数を増加させる方式が切望されている

一方，分散共有型並列計算機は，プロセッサとメモリ間のスループットによらず接続プロセッサを増加させることが可能であるが，他プロセッサユニットにあるデータを参照する場合のオーバヘッドが大きく，RISCプロセッサを使用した場合性能向上の妨げとなっており，このオーバヘッド隠蔽のための工夫が必要である．

我々は，リードアクセスとライトアクセスの性質の違いに注目し，プロセッサの接続可能台数を増加することが可能な並列計算機“砂丘”の研究を行ってきた．[1]“砂丘”は共有メモリを分散し，リードアクセスパスとライトアクセスパスを分離した．ライトアクセスパスについてはデータをブロードキャストする機能を持ち，分散されたメモリの内容を同一に保持するマルチリード・ワンライト方式を提案した．共有メモリに対し，リードアクセスの集中を回避し，より多数のプロセッサを接続可能とした．このマルチリード・ワンライト方式の欠点として，同一内容のメモリが多数存在するため，メモリが有効に使用できないという欠点があった．

本論文では，マルチリード・ワンライト方式を発展させた方式としてWOSM(Write Only Shared Memory)方式を提案し，考察する．ライトアクセスとリードアクセスの性質の違いに加え，アルゴリズムに潜在する共有データのローカリティに注目する．“砂丘”と同様に，リード可能な領域を限定する．そして，ライトアクセスをブロードキャストではなくマルチキャスト可能とする．リード可能な領域が限定されているにもかかわらず，擬似的に共有メモリを実現する．WOSMは，十分なプロセッサへのデータ供給能力を持ち，従来4台程度であった共有メモリ型並列計算機に比べ実質的なプロセッサの接続台数を増加することができることを示す．

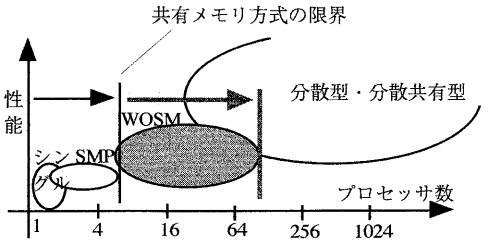


図1　WOSMの守備範囲

2. WOSM(Write Only Shared Memory)方式

2.1　方針

(1)共有メモリ方式の利点であるプログラミングの容易さを実現

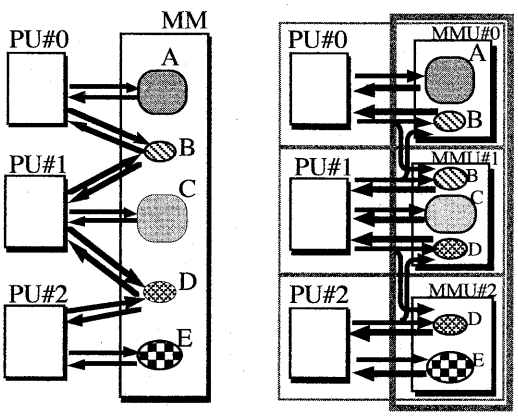
(2)共有データの授受を高速性に行えるようにする．

(3)プログラムの最適化を不要にするため全てのメモリが等距離に見えるようにする．

(4)また，RISCプロセッサで高い実効性能を得るために，リードアクセスレイテンシによるオーバヘッドを削減する．

2.2　WOSMのモデル

一般のプログラムにおいて，共有データの全ての領域は全てのプロセッサから一様にアクセスされるのではなく，アクセスのローカリティが存在する．この性質を利用して，共有データを分割して，メモリを分散できるはずである．図2に共有メモリ方式とWOSM方式のモデルを示す．図2(a)に共有メモリの例を示す．命令の実行を行うプロセッサやキャッシュメモリ等からなるPU(Processing Unit)3台と共有メモリがあり，4種類の演算データが存在する．データ領域A,C,Eはそれぞれ



(a)shared memory model　　(b)WOSM model

PU#0のみアクセス　PU#0とPU#1で共有

PU#1のみアクセス　PU#1とPU#2で共有

PU#2のみアクセス

PU : Processing Unit
MM : Main Memory
MMU : Main Memory Unit

図2　WOSMモデル

PU#0,PU#1,PU#2からのみアクセスされる．データ領域BはPU#0, PU#1から，データ領域DはPU#1, PU#2からアクセスされる．このように演算データが分離されている場合でも，共有メモリ方式では3つのPUから1つのメモリに対しアクセスが集中する．

一方，(b)のWOSMのモデルではメインメモリ(MM:Main Memory)をPU対応に3つに分割した．分割単位のみリードアクセス可能とし，ライトアクセスは結合網によって接続し全体に対して可能とする．共有している部分についてライトアクセスをマルチキャストする事によって同一内容を保持し，共有メモリを実現する．

ライトアクセスのみでリニアなアドレス空間を持ち，システムとして共有メモリを実現しているのでWOSM(Write Only Shared Memory)と呼ぶ．

2.3　WOSMの実現

(1)メモリの分散

共有データアクセスにローカリティがあることを考慮し，メインメモリを分散し，全てのプロセッサからのメモリアクセスが1ヶ所に集中するのではなく多数のメモリに分散するようにする．

(2)自ブロックのみリードアクセス可

プロセッサとメインメモリユニット(MMU)の組をブロックと定義する。プロセッサからは自ブロックのMMUのみにリードアクセスできる。他ブロックのプロセッサから自ブロックのMMUにはリードアクセスできない。

自ブロックのプロセッサは、他ブロックのMMUへはリードアクセスはできない．

(3)全ブロックに対しライトアクセス可

ライトアクセスは全ブロックのメモリすなわちメインメモリ全体にアクセス可能とする．ライトアクセスのみでシステム全体としてリニアな単一アドレス空間をもつ．

(4)ブロード/マルチキャストライト機能

ライトアクセスにブロードキャストまたはマルチキャストする機能を設ける．

(1)により，最小限の結合網でプロセッサに対し十分なデータ供給することができ，プロセッサの性能を最大限引き出すことができる．そして，(2)により，リードアクセスパスの結合網による遅延を小さくすることができ，リードアクセスレイテンシを最小限にできる．(3)により，リニアな単一アドレス空間を持つ．(4)により，複数の領域に同じ内容が存在し，プロセッサから見ると，必要なデータが他プロセッサと共有されているように見え，演算に必要なデータにアクセスできる．すなわち、リードアクセスレイテンシの小さい共有メモリが実現できた。

ライトアクセスは，メインメモリ全体に対し可

能であるため，ライトアクセスレイテンシはリードアクセスレイテンシに比べ大きくなるが，ライトアクセスレイテンシの増大はシステム性能に大きく影響しない．

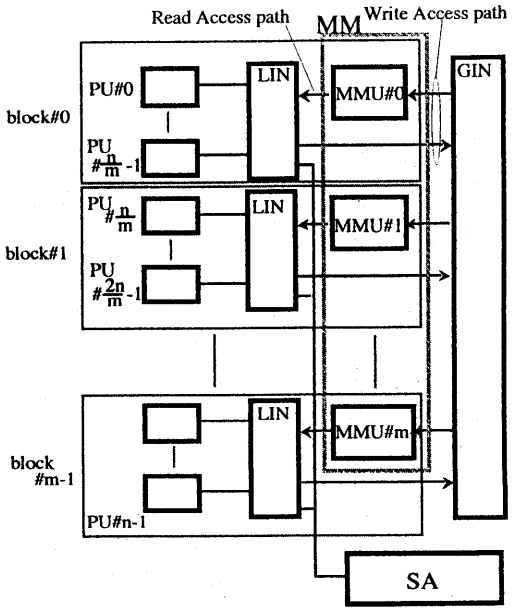
2.4 WOSMのハードウェア構成

図3にWOSMの構成を示す．WOSMはMM(Main Memory)を分割する．分割した1つのユニットをMMU(Main memory Unit)と呼ぶ．ライトアクセスは結合網を通じ，全てのMMUにアクセス可能である．1つのMMUにn/m個のプロセッサをLIN(Local Interconnection Network)を通じて接続する．ライトパスはGIN(Global Interconnection Network)によってMMUと接続する．LINとGINに接続し，GINとMMUを接続する．プロセッサはキャッシュとローカルメモリを持つことができる．

図4 (a)にWOSMのメモリマップの1例を示す．アドレス0からLAB(Local memory Area Bottom)まではLM(Local Memory)の領域である．LMは，PU上に実装され，それぞれのプロセッサに対応しローカルにアクセスされる．このエリアは主にOSやアプリケーションプログラムを格納する．LMは必須ではなくMMにOSやアプリケーションプログラムを格納しても良い．

MMの領域は，プロセッサは自ブロックのMMUのみリードアクセス可能である．一方ライトアクセスパスはMM領域全体にアクセス可能である．ブロック#0のプロセッサ(PU#0～PU#n/m-1)はMMU#0に対しリードアクセス，ライトアクセス共に可能である．しかし，MMU#1～MMU#m-1についてはライトアクセスのみ可能であるが，リードアクセスは許さない．ブロック#1のプロセッサ(PU#n/m～PU#2n/m-1)についても同様である．GINはブロード/マルチキャストモードを持つ．図3はクラスタ型の並列計算機の構成を示したが，m=nの構成でも良い．

全LINを経由し全てのPUをSA(System Area)に接続する．SAは全てのプロセッサからアクセス可能な共有資源で，I/OやI/Oバッファ，プロセッサ間の同期処理を行うための機構を持つ．



PU : Processing Unit
MM : Main Memory
MMU : Main Memory Unit
LIN : Local Interconnection Network
GIC : Global Interconnection Network
SA : System Area

図3 WOSMの構成

2.5 WOSMのメモリ配置

WOSMのアドレスマップとデータの配置方法について述べる．

図4にWOSMアドレスマップの1例を示す．図1の例のようなA～Eのデータを MMUにマッピングする．領域A, C, Eは共有する必要のないデータであり，領域AはPU#0のみがアクセスする領域であるのでMMU#0，同様に領域Cも同様にPU#1のみがアクセスするのでMMU#1，領域EはPU#2のみがアクセスするのでMMU#2にそれぞれ割り当てる．領域BはPU#0とPU#1で共有するデータであり，MMU#0とMMU#1に割り当てる．領域Dも同様にPU#1とPU#2で共有するデータであり，MMU#1とMMU#2に割り当てる．領域Bにライトする場合、MMU#0とMMU#1にマルチキャストを行う。領域D

も同様である。

ライトするアドレスとして、図4(a)のWrite accessに領域A～Eにアクセスするアドレスの領域を示している。MSTからMMBの領域のどのアドレスにアクセスするかによって、MMUのある一つにライトするか複数のMMUにアクセスするか選択できる。MMU#0の領域Aにライトしたい場合、MSTからMST+MUS-1のアドレス領域にライトする。同様にMMU#0とMMU#1で共有されている領域Bにライトする場合、MST+2・MUSからMST+3・MUS-1の領域にライトする。ライトアクセスは、GINでマルチキャストとなりMMU#0とMMU#1にアクセスされる。

ライトアクセスをブロードキャストまたはマルチキャストすることにより，同じ内容のデータが複数箇所に存在することになる．メモリの割り当てを工夫することにより，重複するメモリ領域を削減できる．重複するメモリ量は分散記憶型並列計算機と同程度である．分散記憶型並列計算機でもメッセージによって転送された他のプロセッサのデータをコピーして持っているため，同程度のメモリ領域の増大が発生する．

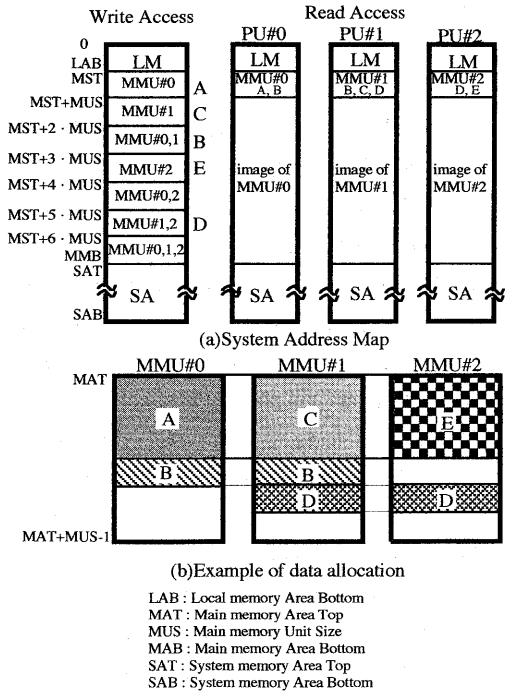


図4　WOSMアドレスマップの１例

3. 評価

WOSMがプロセッサ数を増加してもリニアに性能向上できることを示すために性能を机上計算し、従来の共有メモリ方式と分散共有メモリ方式で比較する．ベンチマークは図5に示したDAXPYと陽解法による拡散方程式の求解プログラムで行う．

各並列計算機の仕様を図6のように仮定する．プロセッサは100MHzで動作する．メモリアクセスレイテンシは，共有メモリ方式で結合網の遅延を考慮し300[ns]とし，分散共有メモリ方式でローカルアクセスの場合200[ns]，グローバルアクセスの場合1[μs]とし，WOSMは200[ns]とする．メモリアクセス中、プロセッサの命令は停止すると仮定した．

図7にDAXPYの評価結果を示す．このベンチマ

```
      DO 10 K=1, N
        Y(K)=A*X(K)+Y(K)
10    CONTINUE
```

(a) DAXPY

```
      DO 10 K=1, N
        X(K)=(X(K-M)+X(K-1)+X(K+1)+X(K+M))/4
10    CONTINUE
```

(b)拡散方程式陽解法

図5　ベンチマークプログラム

	共有メモリ	分散共有	WOSM
プロセッサ	動作周波数　100MHz 整数命令　1命令実行/サイクル 浮動小数点(8B)　1命令実行/サイクル		
キャッシュ	ラインサイズ　64B ライトスルー		
メモリレイテンシ	300[ns] 30サイクル	ローカル 200[ns] 20サイクル	200[ns] 20サイクル
		グローバル 1[μs] 100サイクル	

図6　性能評価の前提条件

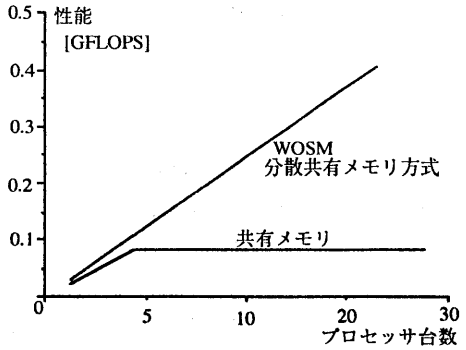


図7　DAXPYの性能評価結果

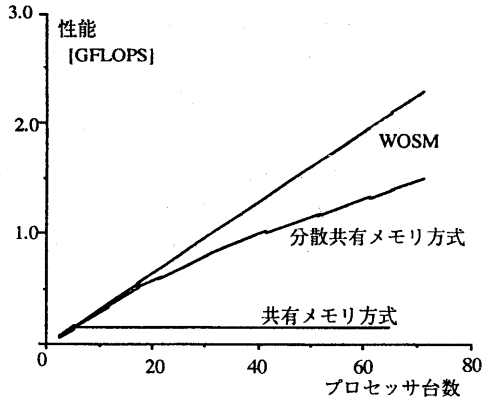


図8　拡散方程式の性能評価結果

ークは100%並列に実行でき，データもプロセッサ対応に分割できる．共有メモリ方式は，プロセッサ台数が少ない場合でもリードアクセスレイテンシが大きい影響で性能が低くなっており，プロセッサ台数が多い場合共有メモリのスループットの制限によりプロセッサ数を増加させても性能が飽和している．リードアクセスレイテンシが同じ分散共有メモリとWOSMは同じ性能となる．

図8は拡散方程式の性能評価結果である．DAXPYと同様，共有メモリ方式では性能が低くなる．分散共有メモリ方式は，グローバルメモリをリードする際のオーバヘッドにより，WOSMより性能が低下している．

WOSMは，他ブロックに同一内容のコピーが発生するためデータ領域のメモリ量が増加する．今回評価を行ったベンチマークではDAXPYは100%データ分割できるためメモリ使用量は増大しないが，拡散方程式の例では10%増加する．

4. おわりに

従来の共有メモリ方式に比べ、より台数を向上できる方式であるWOSMを提案した．結合網の増大を押さえつつ共有メモリからプロセッサへの十分なスループットを確保でき，リードアクセスレイテンシを低く押さえることができる．WOSMは共有メモリ方式と分散記憶方式の長所を合わせ持った方式である．

今後の課題として，WOSMの実現性について検討を行う．共有データのユーザプログラムでの定義方法，プログラム実行時のメモリ割り当て方法，GIN，LINの実現性，同期排他処理，キャッシュコヒーレンシ制御等である．

また，様々なベンチマークについて評価を行い，性能低下の要因となりうるライトアクセスパスのトラフィック量やデータ領域のメモリ量を見積もる。

参考引用文献

[1]　荒川修，橋本正巳，井上倫夫，小林康浩：
資源共有型並列計算機“砂丘”，
情処学計算機アーキテクチャ研究会，CA-85-1，
pp1-6，(1990.11.21)

[2] I.O.Mahgoub and A.K. Elmagarmid : Performance analysys of a generalized class of m-level hierarchical multiprocessor systems, IEEE Trans. Parallel Distrib. Syst., vol.3,pp129-138,Mar.1992.

[3] Kifung C. Cheung, Gurindar S. Sohi, Kewal K. Saluja, Dhiraj K. Pradhan: Design and Analysis of a Gracefully Degrading Interleaved Memory System, IEEE Trans.Computers,vol.39 No.1,Jan 1990.

[4] Allan Gottlieb, Ralph Grishman, Clyde P. Kruskal, Kevin P. Mcauliffe, Larry Rudolph, Marc Snir: The NYU Ultracomputer - Designing an MIMD Shared Memory Parallel Computer, IEEE Trans.Computers,vol.32 No.2,Feb. 1983.