

## マルチプロセッサ性能評価システムの開発 (1)

## 7U-4 ～ 知識ベースマシンMu-Xへの適用 ～

浅野 滋博 石村 多喜二 柴山 茂樹 \*伊藤 文英  
(株)東芝 総合研究所 \*ICOT

## 1. はじめに

われわれは、知識ベース処理の処理系と、知識ベース処理に適したハードウェアの実験のために知識ベースマシンMu-Xを開発している。[1]

Mu-Xではすでに様々な機能をインプリメントし、ハードウェア、ソフトウェアの両方の評価を進めている。

評価の基準となるのは演算時間や、メモリアクセス回数であるが、これらのデータを取るためにはソフトウェアによる方法[2]と専用のハードウェアによる方法がある。

ソフトウェアによる方法はソフトウェアの挙動が細かく取れる反面、ディスクや、OSの振舞い、メモリアクセス、並列の効果などは処理に掛かる時間のために測定しにくいという欠点がある。

一方、専用のハードウェアを使用した方法は細かな挙動は追えないものの、ほとんど演算に掛ける時間に影響を与えずに測定ができるので、並列性や処理ネックの測定に適している。

われわれは以上の二つの方法を測定に適した使い分けをする事とした。

Mu-Xでは実験機という性格からそのプロセッサエレメントとして汎用のVMEボードを使用しており、ハードウェアで評価データを取るような機能はサポートされていない。そこで、われわれはMu-Xにサポートされていなかったハードウェアによる測定機能を追加するために性能評価ボードを開発した。

## 2. 性能評価ボードに要求される項目

性能評価ボードで測定したい項目をあげると次のようになる。

①プログラムのどの部分がよくアクセスされているかが知りたい。

実際にプログラムを走らせてみて、よくアクセスされる部分があればそのループを最適化することによりプログラムの実行速度を向上させることができる。

②データのどの部分がよくアクセスされるかが知りたい。

よくアクセスされる部分はデータ構造の工夫などで実行速度の向上をはかる場合、大きな役割をしめる。

③あるタスクの実行時間が知りたい。

CPUごとの実行時間を測定してどれぐらい並列度が出ているかを測定することができる。

④ディスクの処理にかかっている時間を知りたい。

OSも含めたディスクの処理にかかっている時間を測定することができる。

⑤システムダウンの原因を知りたい。

複雑化したソフトウェアではソフトのバグによりシステムダウンがおこる。とくにMu-Xのようなマルチプロセッサ環境では原因をつきとめるのは大変である。システムダウンまでの履歴を調べることができればバグの原因を発見するのが容易になる。

⑥どのプロセッサがどんな処理を行っているかがある瞬間について知りたい。

マルチプロセッサ環境で他のプロセッサがどんな処理を行っているかを知るのは容易ではない。ソフトウェアで実行履歴を残すとどうしても処理時間がかかってしまいプログラムの挙動が変わってしまうかもしれない

## 3. 性能評価ボードの設計方針

2章であげた要求を満たすハードウェアは、次のような理由からなるべく簡単な機能であるほうがよい。

## 1) 実装の形態

性能評価ボードは各CPU毎に1枚と、各CPUから共通に読み書きできる共有メモリの位置に1枚置く必要があるので、ボードが小さいほど望ましい。

## 2) 柔軟性

様々な測定を行うには機能が単純な方がよい。

ハードウェアの機能を単純化すれば、評価データの分析をするソフトウェアの処理は複雑になるが、評価データの分析は演算が終了してから行うので実行時間に影響を与えない。

以上のような理由から性能評価ボードは単純でしかも様

Development of a Performance Evaluation System in Multiprocessor Environment(1)

Shigehiro Asano, Takiji Ishimura, Shigeki Shibayama TOSHIBA R&D Center

Fumihide Itoh ICOT

々な測定ができるように設計した。

#### 4. 性能評価ボードの機能

以上に述べたような要求により、性能評価ボードの機能は次の二つに絞った。

##### ◎カウンタ機能

指定されたアドレス範囲のアクセスをカウントする機能。アドレス範囲の大きさ、開始アドレスを柔軟に指定できる。また、R/Wの指定もできる。

この機能により2章の①、②の要求が満たされる。

##### ◎トレース機能

指定されたアドレス範囲がアクセスされたときに、その瞬間のタイマの値とアドレス/データをトレースデータとしてメモリにストアする。

この機能はCPUの実行履歴をとる機能で様々な用途が考えられる。

例えば③の目的で、実行するタスクの最初と最後にアドレス範囲を指定しておけば実行を開始したときと終了したときのトレースデータのタイマの値を見れば実行時間は分かる。ディスクなどの処理時間を測定する④の目的でも同様な方法で測定が出来る。

また、システムダウンするまでの履歴を知りたい⑤の場合はアドレス範囲をユーザプログラム空間全体に設定しておけばシステムダウンするまでの実行履歴がわかる。

並列処理のある瞬間の様子が知りたい⑥のような場合は各性能評価ボードのタイマを合わせておけばトレースデータとタイマの値によって各CPUが何を行っているかを知ることができる。

ところで、性能評価ボードは各CPU毎にあるので、ボード毎にあるタイマを合わせなければ⑥のような目的には使えないが、M u - Xには全てのCPUから共通に読めるタイマがあるのでこのタイマを読みに行くときの各性能評価ボードのトレースデータから各性能評価ボードのタイマとシステムに共通なタイマのずれが計算できる。

このように、二つの機能で目標にあげた全ての測定は可能となる。

#### 5. 性能評価ボードの構成

以下図に従って性能評価ボードの構成について説明する。図1は性能評価ボードのブロック図である。

性能評価ボードはVME/VMXバスのトランザクションをモニタしてデータを取るようになっている。

TRACE MEMORYはカウンタ機能の時はカウンタとして使われ、トレース機能の時はトレースデータをストアするための使用される。

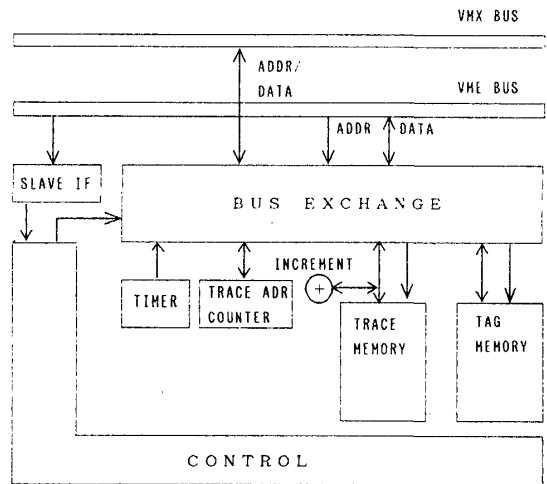


図1 性能評価ボードブロック図

TAG MEMORYはアドレス範囲を設定するもので、カウンタ機能の時はどのアドレス範囲がTRACE MEMORYのどの番地に割り当てられるかが設定される。このTAG MEMORYのアドレス変換機構はキャッシュのdirect mappingの方式と同様である。

INCREMENTはカウンタ機能でカウンタの値をインクリメントするのに使用される。

TRACE ADR COUNTERはトレース機能のTRACE MEMORYのアドレスを更新するのに使用される。

TIMERは、トレース機能でその値がバス上のアドレス/データとともにストアされる。

BUS EXCHANGEはこれらの資源のデータパスの切り替えに使用され、CONTROLが制御を行っている。性能評価ボードの動作モードや、アドレス範囲の設定はVMEバスのSLAVE IFをつうじて行われる。

#### 6. まとめ

性能評価ボードはVMEのダブルハイトのボードに実装することができた。

性能評価ボードは単純な機能であるが様々な測定が可能になるように設計されている。今後は、性能評価ボードで収集された結果を分析するソフトウェアを作成し、M u - Xの評価を進めていく。

#### 参考文献

- [1] 酒井ほか, "知識ベースマシンM u - X (1) ~ (5)" 第37回情報処理全国大会講演論文集
- [2] 沖ほか, "マルチプロセッサ性能評価システムの開発 (2)" 第38回情報処理全国大会講演論文集