

2D-1

## スーパースカラプロセサの高速シミュレータによる評価

武田謙治、井上 淳、白川健治  
(株)東芝

## 1 はじめに

現在スーパースカラ型のマイクロプロセサは盛んに研究されている[1]。スーパースカラプロセサでは達成される高速化は演算器の数や性能および結合方式等に依存する。しかしハードウェアが複雑になる割には性能が向上しないので、達成される性能を考慮しつつハードウェア構成を必要最小限にする必要がある。本研究は最適なハードウェア構成を特定するために、シミュレータ上でプロセサモデルのハードウェア構成を仮想的に変更しながら、実際にプログラムを実行させて性能を評価する。

## 2 スーパースカラプロセサの基本モデル

評価の際には表1に示す基本モデルを想定した。レジスタポート数は演算器に対応した数だけであると仮定した。in-order issueのスーパースカラプロセサは既存のオブジェクトコードではout-of-order issueに対して性能が劣るが、コンパイル時にハードウェア構成に応じた最適化を行うことにより性能低下を十分に補うことができるし、ハードウェア構成が簡素になるという利点を持つ[1]。

命令発行方式：4命令同時発行(4ワード固定境界毎) in-order issue, out-of-order completion
演算器の数：ALU(パレルシフタ、アドレス計算を含む)2*個 整数乗算器 1個 分岐器1個 浮動小数点加算器 1*個(パイプライン構成：加減算のレーテンシーは2*) 浮動小数点乗除算器 1*個(パイプライン構成：乗算のレーテンシーは2*)
データキャッシュポート数：2*個(各々64ビット)
レジスタ数：整数32ビット×32個、浮動小数点32ビット×64個または64ビット×32個
パイプライン：5段パイプライン(フェッチ、デコード、実行、メモリアクセス、ライトバック)

表1 プロセサの基本モデル

## 3 シミュレータの概要

シミュレータはハードウェア構成を柔軟に変更できかつソフトウェアから見て実機に忠実な動作を保証する条件のもとに作成した。機能レベルでパイプラインを再現したシミュレータとすると共に、大規模なベンチマークを評価可能な高速動作を重視した設計とした。本シミュレータの動作原理は、5段のパイプライン中に、実行するインストラクションに各ステージでタグを付けてそれをデータフロー的に流すことにより動作する。タグにはデコードステージで付けられる命令番号、ソースレジスタ値、使用する機能ユニット識別子、実行ステージで付けられるライトバック値やレジスタ番号、次のステージでの処理内容等がある。ベンチマークが発生するシステムコールはシミュレータから呼び出す方法により実現をしたので、多くのプログラムがそのまま動作可能となった。

本シミュレータは現在開発中であり、キャッシュやブランチターゲットバッファは未実装であるので以下の評価ではメモリアクセスは全てキャッシュにヒットすると仮定した。本シミュレータ上では、おおむね1秒間に3万から4.5万命令を処理することが可能である。

## 4 コンパイラ

in-order issueのスーパースカラプロセサではコンパイラ側でスケジューリングによる最適化を行うことが必須となる。そこで本評価ではコンパイラを新たに作成して評価を行った。このコンパイラはループアンローリング、ソフトウェアスケジューリングを行なうのでプロセサの能力を十分に引き出すことができる。またターゲットとなるプロセサのレーテンシーや演算器数等が変更された場合にも、コンパイル時にパラメタとして与えて最適なソフトウェアスケジューリングを行うことが可能でありハードウェア構成に適したコードを生成する能力を持つ。

5 性能評価

ここでは評価の対象として整数系としてstanfordベンチマークを、浮動小数点系としてlivermoreループ(No.1からNo.14)を用いた。前者では各ベンチマーク実行時間の調和平均を用いた。

ALUとメモリポートの個数を変更した時の実行時間の比を図1に示す。図1はALUの個数が1個の時の実行時間を1とした時の実行時間の比である。図1に示す通りALUとメモリポート数は、整数系では2個づつ、浮動小数点系ではそれぞれ3個以上必要であることがわかる。また前者は命令間の依存性のためにALU演算でネックになり、後者では配列のアクセスのためのメモリポート数が重要であることがわかる。

次に浮動小数点加算器と乗算器の数を共に2個に増やした時と、浮動小数点加減乗算のレーテンシーを2から1に変更した時の評価を図2に示す。図中のFPU2というのは加算器と乗算器が2個づつあることを示している。図2より、演算器の数を増やすよりもレーテンシーを短縮した方が性能が上がる事がわかる。これは単位時間あたりの処理能力を高めるよりもレーテンシーの短縮の効果の方が高いことを示している。また浮動小数点の演算器を増やすならば、増やした演算器に十分なデータ量を供給可能にするために、バランス良くメモリポート数も増やさなければならないことがわかる。

今後はプランチャターゲットバッファを組み込んだり、命令発行方式を変えたり、より大規模なベンチマークテストを用いる等を行い、より進んだ評価を行う予定である。

演算器はバランス良く増やせば性能は向上するがチップ上での面積の増大は、例えばオンチップキャッシュの容量を圧迫してしまうので、トータルの性能を向上できるとは限らない。全体を見渡した設計する必要がある。

性能向上比 %

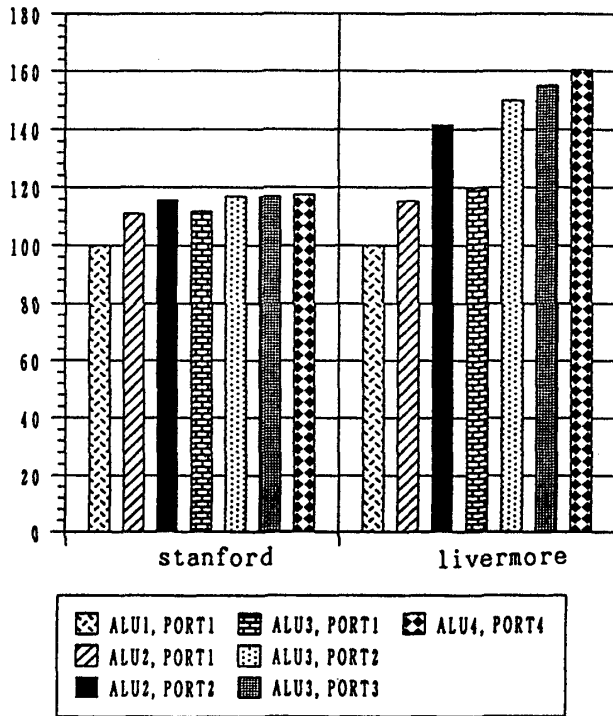


図1 ALU数とメモリポート数と性能向上比

性能向上比 %

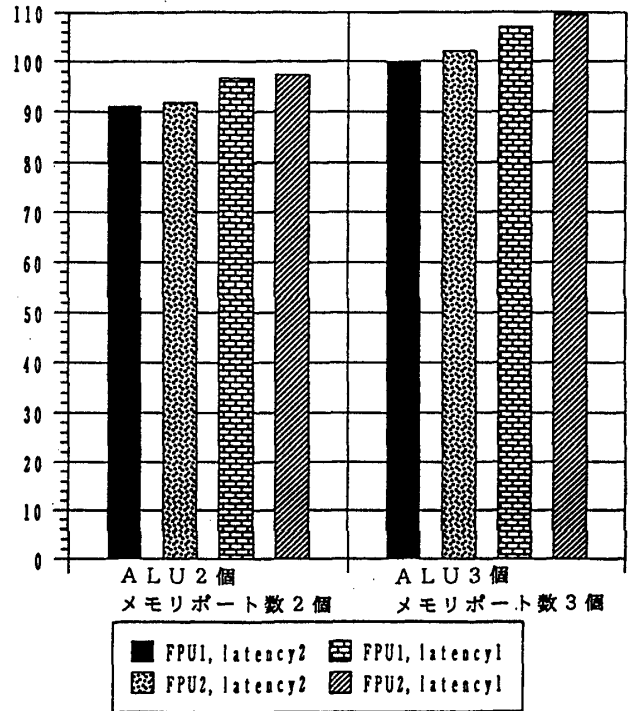


図2 FPU数とレーテンシーと性能向上比

6 まとめ

高速でハードウェア構成を可変にできるシミュレータを作成して様々な評価を行った。本評価によりin-order issue, out-of-order completion型のスーパースカラ型プロセッサにおいて、必要な演算器数や性能に指針を与えた。

参考文献

[1] Mike Johnson. Superscalar Microprocessor Design. 1991.