

## 研究課題別評価

1 研究課題名：制御フローコードとアドレス計算コードの分離による新しいプロセッサアーキテクチャの研究

2 研究者氏名：吉瀬 謙二

3 研究の狙い：

動作周波数と並列性の向上により、年率55%という高いプロセッサの性能向上が数十年に渡って維持されてきた。今後も同様の性能向上を維持するためには、10億個を超えるトランジスタという豊富なハードウェア資源を利用して、動作周波数と並列性をバランスよく向上させる努力が必要となる。現在の数GHzという動作周波数は、パイプライン段数を増やすことにより、高々2倍程度にしか向上しない。一方で、命令レベルあるいはスレッドレベルの並列性を利用することで従来の性能向上率を維持するためには、10年後にはサイクル当たり50個の命令を並列に実行する必要がある。一般的なサーバ計算機において汎用アプリケーションを走らせた際の命令レベル並列性は2程度であり、命令レベル並列性50を達成するためには約25倍の並列性を抽出しなければならない。

近年、プロセッサ性能の向上を目指してデータ値予測などのさまざまな投機技術による性能向上が報告されている。これらの努力に加えて、命令レベル並列性50を達成するためには、より根本的な所から、その土台を形成する新しいアーキテクチャを検討することが重要な意味を持つ。現在のスーパースカラプロセッサの主要な要素を大別すると、メモリからデータを取得して必要とするレジスタにデータを供給するメモリデータフロー、分岐命令による命令列の分断を解決しながら必要とされる十分な命令を実行機構に供給する制御フロー、データ授受のタイミングを取りながら高速に処理を進めるレジスタデータフローという3つの領域に分けることができる。高い命令レベル並列性を達成するために、分岐予測やキャッシュといった部分的な議論に加えて、これら3つの流れ(フロー)を効率良く扱う新しいプロセッサアーキテクチャを検討することが本研究のねらいである。特に、3つの流れを効率良く処理するために、従来の命令セットの枠組みにとらわれることなく、独自の命令形式を駆使したプロセッサアーキテクチャを検討する。

4 研究成果：

本研究の成果は、プロセッサ基盤ツールSimCore、分岐予測器の高性能化、カスケードALU (arithmetic logical unit)を用いることによる並列性の向上、独自の命令形式を用いる高性能プロセッサに大別できる。それぞれに関してまとめる。

4.1 プロセッサ基盤ツールSimCore

本研究を進めるための基盤ツールとして、プロセッサシミュレータSimCore Version 2.0を開発した。本ツールは、高速で、可読性が高く扱いやすいプロセッサシミュレータの実現を目指して設計・実装をおこなった。主な特徴は次のとおりである。(1)豊富な機能を提供する。(2)基本となる機能レベルシミュレータはC++で記述して、2800行というコンパクトな実装により実現する。(3)グローバル変数を排除して可読性と機能の向上を図る。(4)豊富な動作検証の機能を提供する。

SimCoreに施した高速化の有効性を確認するために、分岐予測あるいはデータキャッシュを組み込んだシステムのシミュレータを構築し、これらのシステムのシミュレーションにおいて、プロセッサコアの処理時間が大きな割合を占めること、プロセッサコアとしてのSimCoreの高速化がシステムシミュレーションの高速化に有効となることを確認した。機能レベルのシミュレータおよびクロックレベルシミュレータの一部はフリーソフトウェアとして公開しており、国内外の複数の大学で利用されている。本研究で取り組んでいるプロセッサ高速化のための要素技術の評価、独自の命令形式を用いるプロセッサの検討などにおいてもSimCoreが利用されている。

#### 4.2 分岐予測器の高性能化

命令発行幅の増大と命令パイプライン長の増大により、プロセッサ性能に与える分岐予測器の重要性が増している。予測精度の向上を目指して、新しい分岐予測方式の検討をおこなった。まず、極端な偏りのある分岐命令が存在することを示し、極端な偏りのある分岐命令を区別する方式を提案した。次に、方式をBimode分岐予測器に組み込む新しい分岐予測器としてBimode-Plus予測器を提案した。Biasテーブルと呼ばれるテーブルに、それぞれの分岐命令の過去の分岐方向が全て成立または不成立であるという情報を保存する。この情報を利用することで、ハードウェアのシンプルさを保ちながら、極端な偏りのある分岐命令の方向を高い精度で予測する。

SPEC、マルチメディア、サーバの領域を含むベンチマークプログラムを用いて、コンテキストスイッチと分岐予測精度の関係を評価した。予測精度は、機能レベルのソフトウェアシミュレータを利用した。これらの評価結果から、コンテキストスイッチが発生する実際のプロセッサの環境においても、Bimode-Plus予測器は、Bimode予測器に対して、6.1%から8.5%の予測ミスを削減できることを明らかにした。SPEC CINT2000の10本のプログラムを利用して、予測精度の向上とプロセッサの処理性能向上の関係を検討した。プロセッサ性能は、ソフトウェアシミュレータを用いて計測している。保守的な構成のBASEモデルと、大規模なハードウェア構成を想定するWIDEモデル(サイクル当たり8命令のフェッチ、デコード、完了をおこない、命令ウィンドウのサイズを約512エントリに設定)を定義し、それぞれのモデルにおける分岐予測器の影響を評価した。評価結果から、従来手法のBimode分岐予測器と比較して、WIDEモデルでは、Bimode-Plus分岐予測器により平均で3.2%、最大で8.5%の速度向上を達成できることを明らかにした。

#### 4.3 カスケードALUを用いることによる並列性の向上

レジスタデータフローにおける高速化として、カスケードALUの可能性の検討をおこなった。データ値予測を利用することで、真のデータ依存関係により生じるクリティカルパスを短縮することができる。同様に、演算をおこなうデータのビット長に注目し、短いビット長の命令が連続した場合に、これらの命令をカスケード接続されたALUを用いて高速に処理することで、クリティカルパスに必要なサイクル数を短縮できる。理想的な命令レベル並列性(Oracle IPC)の検討をおこない、命令ウィンドウのサイズを大きくすることで、抽出できる並列性が向上すること、また、カスケードALUを利用することで、利用しない場合と比較して、13%から19%の並列性が向上することを明らかにした。

#### 4.4 独自の命令形式を用いる高性能プロセッサ

分岐命令や分岐条件を計算するためのコードを制御フローコードと呼ぶことにする。また、制御フローコードを形成する個々の命令を制御フロー命令と呼ぶことにする。高い命令レベル並列性を達成するための土台となるアーキテクチャの構築を目指して、制御フローコードを分離する新しいプロセッサアーキテクチャの検討をおこなった。新しいプロセッサアーキテクチャの内部では、命令パイプラインの早い段階で制御フローコードを、それ以外の命令(データフローコード)と分離する。その後、制御フローコードは、それを専用に処理するハードウェアにより高速に処理される。制御フローコードを優先的に処理することで、分岐予測ミスのペナルティを軽減する。また、制御フローコードとそれ以外のコードとを分けることにより、それぞれの処理の流れに適した最適化を選択できるという利点が生じる。

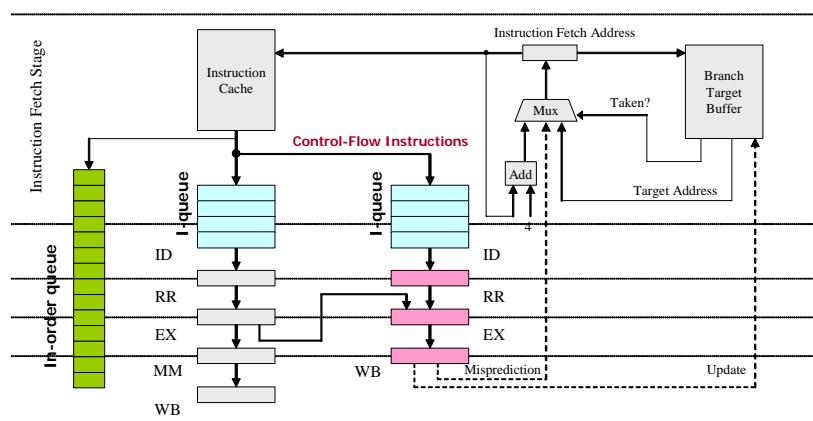


図 1 第1世代の制御フローコードを分離するプロセッサ

第1世代の制御フローコードを分離するプロセッサ(Super Instruction-Flow)の構成を図1に示す。命令フェッチのステージにおいて、制御フローコードかどうかの判定をおこない、適切な命令キューに挿入される。その後、それぞれのフローの特徴に適したパイプラインにおいて高速に命

令が実行される。プロセッサのアーキテクチャステートは、主に、R0からR31までの整数レジスタセット、F0からF31までの浮動小数点レジスタセット、C0からC31までの制御フローレジスタセットから構成される。機械的に、オペランドとして制御フローレジスタを利用する命令が制御フローコードと判別される。幾つかのベンチマークプログラムを利用して、手作業で制御フローコードを分離するプロセッサのアセンブラを構築し、その性能を評価した。評価には、クロックレベルのソフトウェアシミュレータを利用している。この結果から、典型的な5ステージのスカラプロセッサと比較して、制御フローコードを分離するプロセッサが26%の速度向上を達成することを明らかにした。

図1に示した第一世代の構成では、1つのプログラムカウンタおよび命令キャッシュを想定していたために、それぞれのフローの先見能力が制限されてしまう。この弱点を克服するために、第2世代のスーパー命令フローアーキテクチャ(図2)の検討をおこなった。この構成では、プログラムを制御フローコード、メモリ参照のためのアドレス計算をおこなうアドレス計算コード、それら以外のデータフローコードに分離する。また、それぞれのフローを格納する個別の命令キャッシュを持つ。また、個別のプログラムカウンタを有し、必要に応じてフローをフォーク、ジョインする。ベクトル積などのベンチマークプログラムを用いた検討から、この方式の高い性能を確認している。一方、大規模なアプリケーションを用いた詳細評価は今後の課題である。

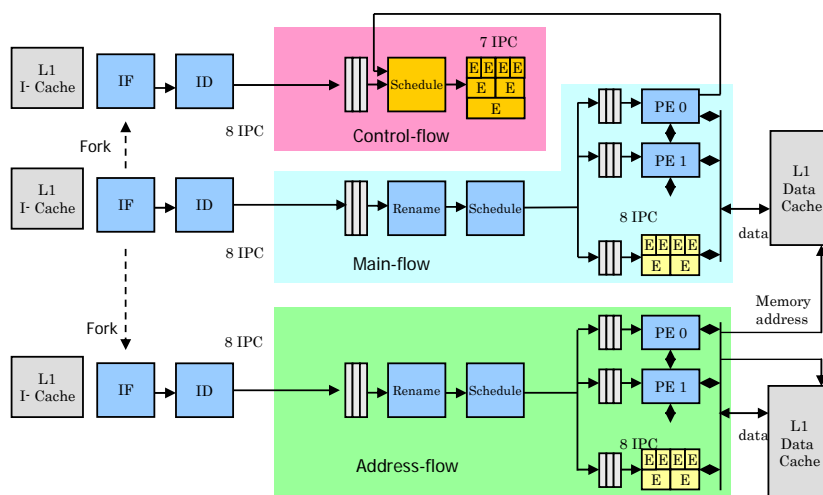


図 2 第2世代の制御フローコードを分離するプロセッサ

## 5 自己評価:

本研究では、高い命令レベル並列性の利用を目指して、要素技術の開発および、独自の命令形式を採用し、制御フローコードとメモリフローコードとに命令流を分離するアーキテクチャの検討をおこなった。プロセッサ基盤ツールSimCoreは、国内外の幾つかの大学において利用されている実績がある。今後も、プロセッサ要素技術やアーキテクチャを評価するための共通インフラのひとつとして開発を続けていく。分岐予測器の性能評価に関しては、予測精度の向上とそれによる

プロセッサ性能の向上を明らかにすることができた。しかしながら、今後の高性能プロセッサでは、1サイクル当たり多数の予測をおこなう現実的な方式の開発、トレースキャッシュとの融合などが不可欠となる。これら、異なる視点からの研究を積極的に進めていく必要がある。高い命令レベル並列性を達成するための基盤技術の確立を目指して、独自の命令形式を用いるスーパー命令フローアーキテクチャの検討をおこなった。第1世代、第2世代のアーキテクチャを提案し、その可能性を示すことができた。一方、その有効性を明確にするためには大規模なアプリケーションを対象とする詳細評価が必要となる。また、本アーキテクチャに適したコンパイラ技術に関して十分な検討が必要となる。本研究課題では、高い命令レベル並列性を抽出するプロセッサアーキテクチャの開発を目指して、研究を進めてきた。しかしながら、プロセッサシミュレータの開発および、分岐予測器の開発といった部分に予定以上の時間を費やしたことで、新しいプロセッサアーキテクチャの評価を十分におこなうことができなかつた点は残念である。提案したスーパー命令フローアーキテクチャという土台をしっかりとしたものへと育て、その上にいくつもの技術を積み上げていくことで高い命令レベル並列性の利用を実現することが今後の課題である。

#### 6 研究総括の見解:

命令レベル並列性50に挑戦するために、その土台となる新しいアーキテクチャについて研究した。具体的にはプロセッサ基盤ツールSimCore、分岐予測器の高性能化、カスケードALUによる並列性向上など要素技術の開発とその効果確認、および独自の命令形式をもち制御フローコードとメモリフローコードとに命令流を分離する新しいスーパー命令フローアーキテクチャを提案とその可能性の提示などの成果を挙げた。また、コンピュータアーキテクチャに関する権威ある国際学会 ACM SIGARCH シンポジウムにおいて独創的な分岐予測器Bimode-Plusを提案し、その性能コンテストにおいて優秀な成績を収めた。研究期間の制約からすべての評価を終えることができなかったようだが、今後の研究の進展を大いに期待している。

#### 7 主な論文等:

##### 論文

1. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「Bimode-Plus分岐予測器の提案」 情報処理学会論文誌コンピューティングシステム, Vol.46, No. SIG 7(ACS 10), pp.85-102, 2005年6月
2. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「SimCore/Alpha Functional Simulatorの設計と実装」 電子情報通信学会論文誌, Vol. J88-D-I, No.2, pp. 143-154, 2005年2月
3. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「PCクラスタを用いたN-queens問題の求解」 電子情報通信学会論文誌レター, Vol. J87-D-I, No.12, pp. 1145-1148, 2004年12月
4. Kenji Kise, Takahiro Katagiri, Hiroki Honda, and Toshitsugu Yuba, 「A Super Instruction-Flow Architecture」 International Symposium on Low-Power and High-Speed Chips (COOL Chips VII), pp. 279-290, April 2004
5. Kenji Kise, Hiroki Honda, and Toshitsugu Yuba, 「SimAlpha Version 1.0: Simple and Readable

Alpha Processor Simulator」Lecture Note in Computer Science (LNCS), Vol. 2823, pp. 122-136, Springer-Verlag, September 2003

6. 吉瀬謙二: 特集 新世代マイクロプロセッサアーキテクチャ, 「タイルプロセッサ」(解説記事), 情報処理, Vol. 46, No. 10, pp. 1131-1137, 2005年10月
7. Kenji Kise, Takahiro Katagiri, Hiroki Honda, and Toshitsugu Yuba, 「Toward A Common Emulation Infrastructure with Large-Scale FPGA」, Workshop on Architecture Research using FPGA Platforms (WARFP2005), February 2005
8. Kenji Kise, Takahiro Katagiri, Hiroki Honda, and Toshitsugu Yuba, 「The Bimode++ Branch Predictor」IWIA-2005, IEEE Computer Society Press, January 2005
9. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「極端な偏りを利用するBimode++分岐予測器の提案」情報処理学会研究報告 2005-ARC-161, pp. 57-62, 2005年1月
10. Kenji Kise, Takahiro Katagiri, Hiroki Honda, and Toshitsugu Yuba, 「The SimCore/Alpha Functional Simulator」Workshop on Computer Architecture Education (WCAE-2004) held in conjunction with the ISCA-31, pp. 128-135, June 2004
11. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「SimCore/Alpha Functional Simulatorの設計と評価」, 情報処理学会研究報告 2004-ARC-156, pp. 31-36, 2004年2月
12. Kenji Kise, Takahiro Katagiri, Hiroki Honda, and Toshitsugu Yuba, 「A Super Instruction-Flow Architecture for High Performance and Low Power Processors」IWIA-2004, IEEE Computer Society Press, January 2004
13. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「Bimode-Plus分岐予測器の提案」電子情報通信学会技術研究報告CPSY-2003-10, pp. 25-30, 2003年8月
14. 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣, 「制御フローコードを分離するプロセッサアーキテクチャの提案」情報処理学会研究報告 2002-ARC-150, pp. 101-106, 2002年11月

#### 特許

1. 発明者: 吉瀬謙二, 発明の名称: 「不安定状態を利用する予測器, プロセッサ」, 出願人: 電気通信大学, 出願日: 2004年12月3日
2. 発明者: 吉瀬謙二, 発明の名称: 「極端な偏りに基づく分岐予測器, それを組み込んだプロセッサ及びハードウェア予測器」, 出願人: 科学技術振興機構, 出願日: 2003年7月7日
3. 発明者: 吉瀬謙二, 発明の名称: 「制御フローコードの分離によるプロセッサの実現方法及びそれを用いたマイクロプロセッサ」, 出願人: 科学技術振興機構, 出願日: 2002年11月20日

受賞 なし

招待講演 なし