

分岐アドレス予測機構の比較検討

3 F-5

中村 友洋, 吉瀬 謙二, 辻 秀典, 安島雄一郎, 田中 英彦
 東京大学大学院工学系研究科

1 はじめに

現在 21 世紀のマイクロプロセッサに関する研究が盛んに行なわれている。それらはより一層の並列性の抽出、メモリ・ウォールの解消、大規模な投機処理などを特徴とするものが多い。本稿では、大規模な投機処理を行なう上で、その性能に大きく影響を与える分岐処理、特に分岐先アドレスがレジスタ間接で計算される分岐命令の分岐先アドレス予測に関する調査・検討を行ない、Multi BTAC(Multi Branch Target Address Cache) の提案・評価を行なう。

2 目的と背景

現在のマイクロプロセッサではスーパースカラプロセッサが主流となっているが、21 世紀におけるマイクロプロセッサ・アーキテクチャを考えたとき、すでに性能向上に限界の見え始めているスーパースカラ方式以外のアーキテクチャを考える必要がある。21 世紀におけるデバイス技術を Moore の法則などから推定すると 10 年後の 2007 年には 1 チップでおよそ 350M 個ものトランジスタが入る。このような大規模なデバイスを利用できる場合のプロセッサアーキテクチャとして、様々な検討が行なわれている [2]。例えば多数の演算器を 1 チップに集積しそこにデータバスを構築して実行する大規模データバス方式 [3] があげられる。大規模データバス方式では大規模な命令の先行展開を特徴の 1 つとしており、高い分岐命令処理効率性が要求される。これまでのスーパースカラプロセッサにおける分岐予測機構の研究により、すでに分岐予測成功率は 90 % 台を得られている。しかし大規模な命令の展開を行なっていく上では、分岐命令の先のパスの展開において予測に基づいた 1 パスのみを展開するのでは予測ミスの際に大きなペナルティを受ける。この傾向は今後さらに強まる可能性がある。そこで分岐予測ミスによるペナルティが高い部分や分岐予測成功率の悪い部分については複数パスを展開する方式が研究されている [1]。このような展開を行なう際に、分岐先がレジスタ間接で計算される分岐命令のアドレス予測が必要になる。これまでは主に BTAC が使われてきたが、従来の BTAC では前回の分岐先アドレスを保存しておくだけなので、予測成功率はあまり高くない。そこで本稿では、より高い精度の分岐先アドレス予測機構の検討を行なう。

A Study of Branch Target Address Prediction
 Tomohiro NAKAMURA, Kenji KISE, Hidenori TSUJI,
 Yuichiro Ajima, Hidehiko TANAKA,
 Graduate school of Engineering, The University of Tokyo

3 コントロールフロー先行展開

大規模データバス方式ではコントロールフロー先行展開と呼ばれる機構により大規模な命令の先行展開を行ない、制御依存関係の解消された状態のデータバスを多数用意することが必要である。図 1 はコントロールフロー先行展開の動作を示したものである。図 1 の矢印はコントロールバスを示し、xxx は分岐確率情報である。

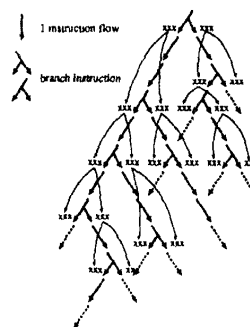


図 1: コントロールフロー先行展開イメージ図

分岐命令	分岐条件	アドレス計算	分岐方向
Bicc	条件付き	PC 相対	図 2(a)
CALL	無条件	PC 相対	図 2(a)
JMPL	無条件	レジスタ間接	図 2(b)
Ticc	条件付き	レジスタ間接	図 2(b)

表 1: 主な分岐命令の特徴

ここで SPARC V8 における分岐命令の種類について述べる。表 1 のように大きく分けて 4 種類の分岐命令があり、その分岐方向は図 2 のようになる。図 1 のような展開を行なう場合には、図 2 のように Taken/NotTaken の 2 通りのみならば、最大 2 パスの展開を行なえば実行すべきパスが必ず展開されることになるが、図 2(b) のように分岐先が複数の可能性を持つ場合には、2 パスの展開では必要なパスの展開が行なわれない可能性がある。

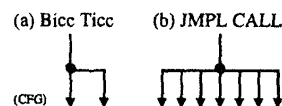


図 2: 分岐命令における制御の流れ

4 Multi BTAC

前節で述べたように、大規模なコントロールフローの展開を行なう場合、レジスタ間接ジャンプをまたぐ展開において、分岐先を複数の可能性の中から正しく予測することが必要である。そこで本研究では、複数の分岐先アドレスを保存する Multi BTAC について検討・評価を行なう。図 3 に示すように通常の BTAC では 1 つの分岐先のみを保存しているが、Multi BTAC では n 個

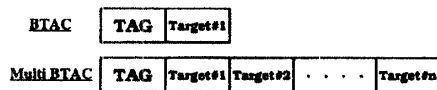


図3: BTACとMulti BTAC

の分岐先を保存する。このような Multi BTAC を multi level = n の Multi BTAC と呼ぶことにする。

5 性能評価

ここでは、SPEC92 などから cc1, compress, dhrystone, espresso, sc をサンプルプログラムとして SPARC V8 シミュレータによる 1 千万トレースを使用して評価を行なう。また JMPL 命令についてのみを評価対象とした。これは次のような理由による。

- Multi BTAC を利用する必要があるのは分岐先がレジスタ間接で指定されるもののみである
- そのような分岐命令は JMPL と Ticc であるが Ticc はトラップ命令であり、さらに出現頻度も低い

図4は multi level を 1~16 までの範囲で Multi BTAC の性能 (分岐先アドレスの予測成功率) を示したものである。ここでは BTAC のエントリ数は 10240, フルアソシアティブ, LRU の条件でシミュレーションを行ない、multi level 以外の要因による性能への影響が極力なくなるようにした。これによれば、multi level = 3

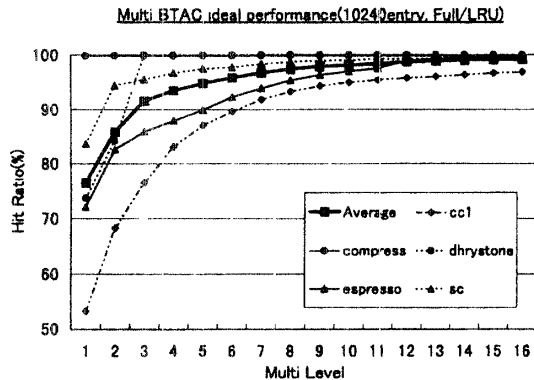


図4: Multi BTAC の multi level 別性能

で平均して 90 % 以上の予測成功率を得ており、通常の BTAC (multi level = 1 の Multi BTAC) に比べて 15 % 以上の性能向上が見られた。ハードウェアへの実装の可能性などを考慮して、以下では multi level = 3 を上限としてシミュレーションを行なう。

図5, 図6は BTAC エントリ数の違いによる性能差を示したものである。各線は multi level = 1~3 のもので、multi level = 2 では平均で 86 %, 3 では 91 % 程度の予測成功率が得られる。BTAC がダイレクトマップである図6とフルアソシアティブである図5のいずれにおいても BTAC エントリ数が 256 もしくは 512 エントリではほぼ性能向上が飽和する。また両者では BTAC エントリ数が少ないときには性能に若干の差があるが、ある程度のエントリ数が確保できる場合には大きな差はない。

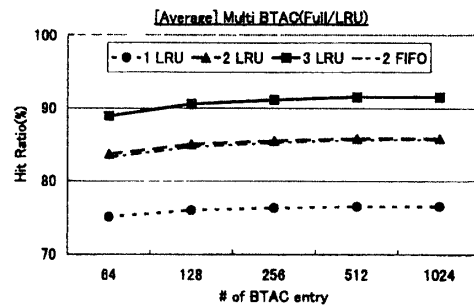


図5: BTAC エントリ数と性能 (フルアソシアティブ)

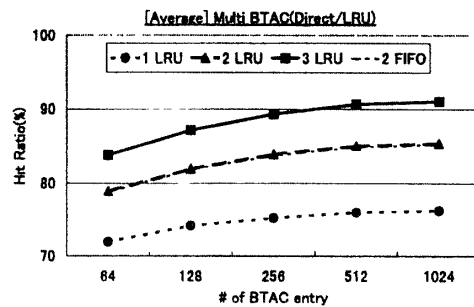


図6: BTAC エントリ数による性能 (ダイレクトマップ) 4way セットアソシアティブについても調査を行なったが、両者の中間的な性能であった。また、分岐先アドレスの置き換え方式についても LRU と FIFO で大きな差はみられなかった。

6 おわりに

Multi BTAC を使い、2~3 個の分岐先アドレスを保存することで通常の BTAC に比べて 10~15 % の予測成功率向上が見られた。512 エントリの BTAC よりも 256 エントリの 2 level Multi BTAC を利用の方がコストパフォーマンスが高いことがわかった。大規模コントロールフロー展開を行なうような今後のプロセッサにおいて、レジスタ間接ジャンプ命令に Multi BTAC を利用することの有効性が示せた。

謝辞

本研究の遂行にあたり、日本学術振興会の特別研究員制度 (「プログラム解析に基づく次世代マイクロプロセッサアーキテクチャの研究」), 文部省科学研究費 (一般研究 (B) 課題番号 07458052 「大規模アータバスプロセッサの研究」) のご支援を頂きました。ここに感謝の意を表します。

参考文献

- [1] A.K.Uht and V.Sindagi, "Disjoint Eager Execution: An Optimal Form of Speculative Execution", 28th International Symposium on Microarchitecture, pp.313-325, Dec 1995.
- [2] 坂井修一, "オンチップマルチプロセッシングに関する初期的検討", 情処研究会 ARCH 122-7, vol. 97, no. 15, pp.33-38, Feb 1997.
- [3] 中村友洋, 吉瀬謙二, 辻秀典, 安島雄一郎, 田中英彦, "大規模アータバスプロセッサの構想", 情処研究会 ARCH 124-3, vol. 97, no. 61, pp.13-18, Jun 1997.