C-12

ターゲットの変更が可能な命令スケジューラに関する研究
Research on the instructions scheduler, which can change a target

長谷川　光[†]　　**荒井　秀一**[†]　　**宮内　新**[†]
Hikaru Hasegawa　　Syuichi Arai　　Arata Miyauchi

1. はじめに

スーパースカラや VLIW プロセッサは複数の命令を並列実行することで, スループットを増加させ高速化を図る. しかしハザードの発生等により命令の並列実行ができない場合, 著しい性能の低化を引き起こしてしまうことから, これらのプロセッサにおいては命令のスケジューリングによりプログラム中から命令の並列性を抽出することが性能向上の鍵であるといえる.

スーパースカラプロセッサは動的にスケジューリングを実行するが, これらの処理はハードウェアで行なうため処理対象である命令ウィンドウを大きくしようとすると回路規模の増大を招きそれに伴った弊害が発生するといった欠点を持つ.

一方 VLIW ではソフトウェアによりスケジューリングが実行されるため回路規模の減少, ハードウェアの簡略化, 仕様の変更が比較的容易に行なえるなどスーパースカラにおける欠点を解消できる. 以上の理由により近年では VLIW プロセッサが注目されている.

2. 研究目的

スケジューリングは, 各プロセッサの持つ命令や資源を基に実行される. すなわちスケジューラは計算機のアーキテクチャに大きく依存する. そのため命令の追加, 拡張及びアーキテクチャの変更が行われ互換性が保てなくなった場合にはスケジューラの更新が必要となる. プロセッサの開発サイクルを考慮した際, 既存のスケジューラを新たな仕様に対応できることが望ましい.

そこで本研究においては, 異なるターゲットにおけるスケジューリング能力の強化を目的としスケジューラに対してターゲットとなるアーキテクチャ情報を与えることで, ターゲットの変更が可能になると共に各ターゲットマシンに特化した高いスケジューリング性能を発揮できる命令スケジューラの研究を行なう. そのためにスケジューラを実行する機構とターゲット情報を構成するための機構を切り離した構造にする (図 1).

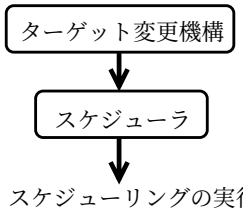

図 1: ターゲット変更機構とスケジューラ

特定用途の分野におけるアーキテクチャ仕様記述からの設計及びターゲット変更手法としては大阪大学により PEAS-III の研究が進められている [1] が現段階においては VLIW プロセッサを採用した設計はできない. 本研究では VLIW プロセッサにおけるスケジューラに注目し命令記述情報及び資源情報によって VLIW プロセッサに対応したスケジューリングを実行可能にする.

[†]武蔵工業大学大学院,　Musashi Instute of Technology

3. スケジューリング

3.1 スケジューラ部

本研究におけるスケジューリングアルゴリズムとしてはスーパーブロックスケジューリングを採用する [2]. この手法はプログラムの中で実行確率の最も高い部分のスケジューリングを実行するものである. この際分岐確率などのプロファイリングデータは既に与えられているものと仮定する.

3.2 ターゲットの変更

3.2.1 情報の抽出

スケジューラが命令をスケジューリングする際には以下の情報が必要であると言える.

- 命令が実行時において使用する資源
- 命令が実行を開始してから終了するまでの時間 (レイテンシ)
- ターゲットマシンが持つ最大並列度及び各種資源 (演算器の数や種類)

命令が使用する資源は, 命令の動作に関する情報より得ることができる. 実際のハードウェア設計は RTL により行われることが多いため命令の情報として RTL を用いるのもひとつの有効な手段である. このような命令動作情報に加えレイテンシ情報と資源情報を付加することで上記の要求を満たせる.

そこで本研究においてターゲット変更機構は図 2 のようにターゲットマシンが持つ全ての資源, 命令レイテンシ, 全命令の動作情報を入力としてこれらの中から資源情報及び命令情報を抽出することで, スケジューラが異なるターゲットマシンで実行が可能となる.

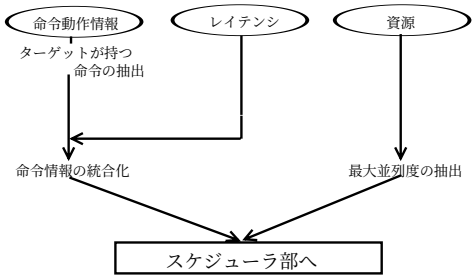

図 2: ターゲットの変更に必要な情報

3.2.2 命令記述統合化処理

命令の動作において類似した命令はほぼ同一の資源を使用して動作する. すなわちスケジューラは全ての命令の動作を完全に知る必要が無い. そのため命令を動作ごとに幾つかのグループに分割すれば命令情報解析処理における冗長性も情報量も少なくなり容易になる. 命令情報統合化は以下の手順で行なう.

1. 全命令の動作記述にレイテンシ情報を付加する
2. 記述が同じ部分をまとめていく
3. 記述が異なっても同じ資源を使用している場合にはひとつにまとめる (図 3)
4. 記述が同じでもレイテンシが異なる場合にはまとめない

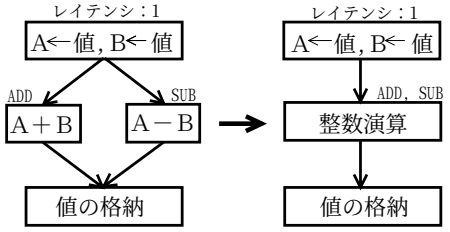

図 3: 命令記述の統合化

上記のように統合化された命令情報は使用する資源によって各命令グループ固有のポイントが割り振られ, レイテンシとともにスケジューラに渡されることになる.

3.2.3 資源情報

一方, 資源情報を基にして資源予約テーブルを構成する. これにより命令の最大並列度が得られることになる. スケジューラは実行時に解析された命令情報とこのテーブルとを参照しながら命令を配置していく.

4. 実験

本研究の効果を DLX 命令セットを持つ仮想 VLIW プロセッサを用いて確認する. この際命令の生成による性能への影響を回避するために,C 言語で記述したプログラムを既存の DLX Compiler によって得られるアセンブリコードを入力として用いることで評価を行なう. DLX Compiler は VLIW に対応した命令のスケジューリング機構は持っていない. 評価時における条件は以下のようなものであるとする.

1. 命令のレイテンシは以下の様仮定にする
   -LOAD 系命令：2
   -整数乗除算：2
   -整数乗除算：2
   -浮動小数点演算：2
   -分岐成立時：2
   -その他：1
2. 最大 4 命令が同時実行可能
3. 命令の実行による例外は一切発生しない

この時, ターゲット変更機構に対して DLX 命令セットの動作情報を与えバブルソート, 文字列照合, 画像フィルタのプログラム (表 1) をスケジューラに対して入力する.

表 1: スケジューリング対象となったプログラム

	基本ブロック内平均命令数	スケジューリング前並列度
バブルソート	5.7	0.8
文字列照合	3.3	0.9
画像フィルタ	16.3	1.0

それにより得られる並列度及び実行サイクル数における性能向上度について計算機上で測定を行った. その結果スケジューラは DLX の全命令を認識し, 対象となる 3 種のプログラムのスケジューリングが可能となり表 2 に示すような効果が得られた.

表 2: スケジューリングの効果

	並列度	性能向上度 (%)
バブルソート	1.3	28
文字列照合	1.1	16
画像フィルタ	2.3	56

バブルソート及び文字列照合に関してはスケジューリングの効果が低い. 一方画像フィルタに関しては並列度が約 2 倍となり比較的高い効果が得られている. しかし得られた並列度は高々2 程度であることからスケジューラの性能自体はあまり高くないといえる.

一方, ターゲット変更ユニットによって解析された命令情報がスケジューラに認識されており, また認識された DLX の命令列が VLIW 命令としてスケジューリングされていることから, 本スケジューラのターゲットを DLX 仮想プロセッサ用に変更することができたといえる

5. おわりに

本研究では,VLIW 型プロセッサの異なるターゲットマシンにおいてスケジューリング可能な命令スケジューラシステムの研究を行なっている. ターゲット変更機構に DLX 命令セットにおける命令動作情報などを与え, これを解析することでスケジューラが DLX 命令を認識し実行することが可能となり,16〜56 %程度の性能を得ることができた.

今後の課題としては更に多くの命令セットを持つプロセッサにおいて本スケジューラを動作させ本研究の有効性を検証するとともに, スケジューラの更なる性能向上を図る予定である.

参考文献

[1] 小林真輔, 三田健太朗, 武内良典, 今井正治：”PEAS-III システムのコンパイラ生成系とその評価 ”, 電子情報通信学会技術研究報告 (2002)

[2] Richard E. Hank, Scott A. Mahlke,Roger A. Bringmann, JohnC. Gyllenhaal,and Wenmel W.Hwu : ”Super block formation using static program analysis”, in proceedings of the 26th Annual ACM/IEEE International Symposium on Micro architecture (1993)