

2D-2

VLIW型プロセッサへの
命令再構成機構の導入伊藤 善政 有田 隆也 曾和 将容
名古屋工業大学 電気情報工学科

1. はじめに

VLIW 計算機は、単一命令流により複数の処理ユニットを同時に制御するSIMD型並列計算機である。基本的なVLIW 計算機ではユニットの同期はクロックのみによってとっているため、ユニット間に複雑な同期または通信機構が必要ない。さらに、マルチポートレジスタファイルによってユニット間のデータ共有を行なっているのでユニットは密結合となり細粒度の並列性が引き出しやすい。本稿では、トレーススケジューリング等のアプローチではない、汎用計算を目的としたVLIW プロセッサの高性能化を目指した新しいアーキテクチャについて述べる。VLIW プロセッサに、遅延ユニット、同期ユニットを付加することによってプロセッサ内部で命令の再構成を動的に行なうものであり、これを動的再構成 VLIW と呼ぶ。

2. 動的再構成 VLIW

2.1 命令の再構成

VLIW 計算機では、トレーススケジューリングのようなスケジューリング法が有効に適用できない場合、データの従属関係からプログラム中に多くの NOP 命令を含みやすい。この結果、コード量が増え1命令語当たりの実行命令数つまり並列度が下がり性能が悪くなる。もし、基本ブロック同士をオーバーラップさせることができるなら全体の実行時間を短縮することが可能である。特に、ループの場合は何度も実行するため、ループのインスタンス間でオーバーラップができるならその効果が大きい。このループのオーバーラップを実現する方法の1つに、ソフトウェアパイプライン¹⁾がある。このソフトウェアパイプラインと同様の効果をハードウェアによって実現するのが、規定型再構成²⁾である。規定型再構成には、ソフトウェアパイプラインなどのようなソフトウェア最適化に比べコード量を減らせるという利点がある。規定型再構成は、基本ブロックのオーバーラップなどの静的なスケジューリング上の問題をハードウェアで解決するものである。

VLIW 計算機では、静的な要因の他にキャッシュミスやネットワーク遅延などのような実行時間の変動に対しても、他のプロセッサに比べて不利な点がある。たとえば、キャッシュミスは単一プロセッサの場合には、さげがたい最小の実行時間の損失であると考えられる。しかし、これが複数プロセッサの場合には、1つのプロセッサがキャッシュミスで停止した場合それとデータ依存関係にあるプロセッサも次々に待ちを生じるので、単一プロセッサよりも損失は大きい。さらにVLIW 計算機では、基本的には全ユニットが実行を終了しなければ次の命令語に進まないため、同一命令語中の一つの操作がキャッシュミスにより待ちを生じたとき関係のない他のユニットにまで不要な待ちを生じさせてしまうため、損失はかなり大きい。この不要な待ちをできるだけ減らし、さらに不必要な NOP 除去のために考えられたのが適応型再構成である。これらの規定型再構成と適応型再構成を組み合わせたものが提案する動的再構成 VLIW である。

2.2 規定型再構成

規定型再構成とは、コンパイラ等によりあらかじめ命令に付加した実行順序の情報に基づいて実行時に命令の再構成をおこなうものである。規定型再構成の概念を図1に示す。普通のプロセッサでは、命令を実行する時、図1(a)に示すようにフェッチ、実行と時間的に続けて実行される。それを規定型再構成では、図1(b)に示すようにフェッチ後数クロック遅らせて実行をする。この遅延ステップ数は、プログラム中に記述しておく。このような方法により命令が実行される順序をフェッチの順序とずらすことにより基本ブロック間でのオーバーラップが可能となる。このためプログラムは実際の実行順序にとらわれない順序で記述できる。その結果、命令の密度をあげることが可能となり NOP 命令を減らすことができる。このように基本ブロックのオーバーラップによってプログラム中に存在するループ等の命令実行の際に最も時間を要する部分の高速化をはかることができる。もし、これをソフトウェアパイプラインで実現するとループに入る時と出る時のつじつま合わせのためにプロローグ、エピローグといったコードを加える必要がある。これによって、実行は高速化されるがプログラムのコード量は増えてしまう。このようにソフトウェアパイプラインのような最適化実行を、プログラムのコード量をあまり変化させずに命令に付加情報を用いて実現する。

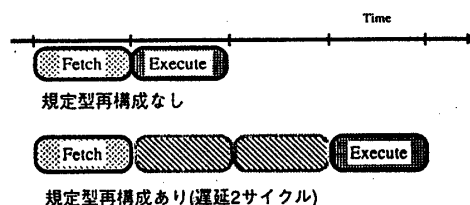


図1 規定型再構成の概念

2.3 適応型再構成

適応型再構成は、命令実行時に種々の要因により発生する命令実行時間の変動のために生じる不要な待ちをハードウェアで極力排除し実行時間の延長を減らすことを第1の目的としている。規定型再構成が主に静的な要因に対する高速化であったのに対し、適応型再構成は動的な要因に対する高速化をはかる。実行時に不規則に発生する実行時間の延長が、従来のVLIW 計算機では全ユニットに同じように働き本来必要のない待ちを生じさせる。適応型再構成では、プロセッサ内部においてユニット間の実行時間の差を最近提案されてきた高速な同期機構を用いて吸収することによって無駄な待ちを解消する。

たとえば、図2 (a) の様なプログラムがありこれを実行したとする。実行時間の変動が命令B 2とD 3に生じ、B 2、D 3の実行時間がそれぞれ1サイクル伸びたとすると図2 (b) のような命令の実行が考えられる。すると全体の実行時間は、図2 (b) からわかるように2サイクル伸びたことになる。しかし、ユニット間にデータ従属などの関係がないユニットは待つ必要がない。そのため、図2 (c) の様な実行となるように制御する。ここでは、B 2とD 3の実行に影響を受けるのはC 3のみとする。この場合、実行時間の延長は命令の並び替えによって吸収されたため全体の実行時間は変動がなかった時と比べ1サイクル増えるだけとなる。このように、命令の実行タイミングを動的にずらすことによって結果的に、実行時間の短縮をすることができる。このようなことをシフトレジスタと同期機構によって実現するのが適応型再構成である。適応型再構成には、同期機構を使用するためコード量の圧縮といった第2の効果もある。これは、本来VLIWにおいてNOP命令を挿入するのは命令の実行順序を保証するためであったが、同期機構を備えた動的再構成VLIWでは、同期機構によって実行順序を保証すれば良く、不要なNOP命令はなくても良いからである。

2. 4 動的再構成VLIWの構成

動的再構成VLIWは、従来のVLIWプロセッサに動的再構成を行なうため、適応型再構成で実行順序を保証するための同期ユニットと2つの再構成の結果に基づいてVLIWを作り直す遅延ユニットを付け加えたものである。基本的な構成を図3に示す。図3においてフェッチユニットと下部にある実行ユニットとの間にあるのが実際に命令の再構成を行なう遅延ユニットであり、右側にあるのが全体の実行の制御を行なう同期ユニットである。遅延ユニットは、現在のところシフトレジスタで構成することが有効であると考えている。同期ユニットについては、細粒度並列実行を支援する高速同期機構として、一般化静的順序制御機構³⁾あるいは重複可能なバリア型同期機構⁴⁾の採用を検討している。

3. シミュレータによる動作確認

動的再構成VLIWのシミュレータによって動的再構成の動作を確認をした。表1に規定型再構成を用いた場合のパルスートプログラムに関してのコード量と実行時間に関する評価をソフトウェアパイプラインによる場合と比較して示す。

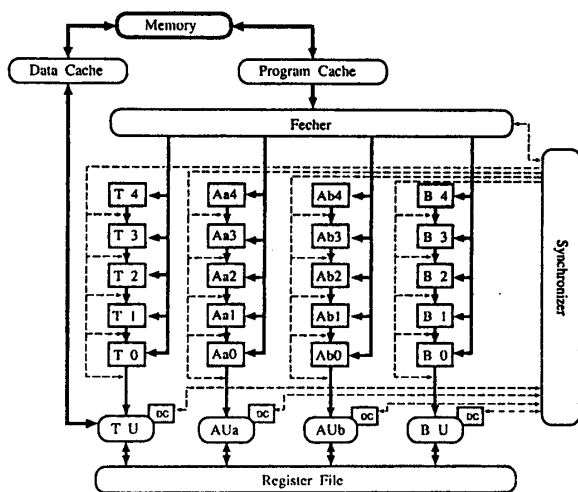


図3 動的再構成VLIWの基本構成

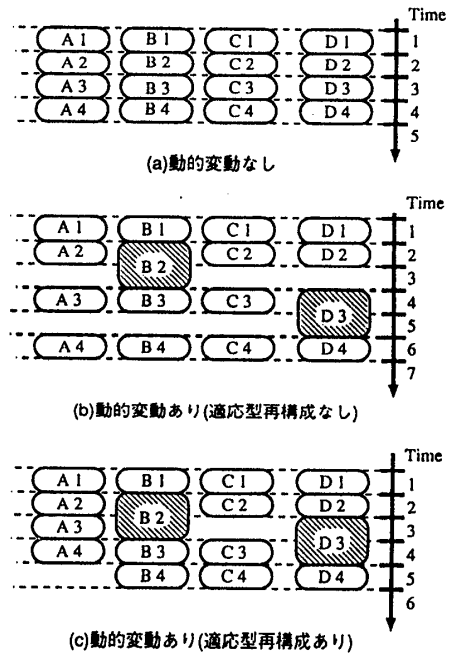


図2 適応型再構成の原理

	コード量	実行時間
ノーマル	9	648
ソフトウェアパイプライン化	10	458
規定型再構成	8	458

表1. パルスートプログラムによるシミュレーションの結果

このサンプルでは、規定型再構成を使用することにより、もとのプログラムに対しコード量においては約10%、実行時間においては約30%圧縮がされている。また、20%少ないコード量でソフトウェアパイプラインと同じ実行時間を達成できることも示された。

4. まとめ

従来のVLIW計算機では、コード量や動的な実行時間変動に関して他の方式のプロセッサに対し不利であった。そこで、遅延ユニットと同期ユニットを付加することによって実行時間の動的変動に対応し、基本ブロックのオーバーラップにより実行時間を短縮し、そしてコード量を減少できるアーキテクチャ提案した。今後は、同期機構として適切なものを検討し、それにあったプログラム作成手法を確立していく予定である。

参考文献

- 1) M. Lam, "Software pipelining: An effective scheduling technique for VLIW Machines", Proc. ACM SIGPLAN '88 Conference on Programming Language Design and Implementation, pp. 318-328, 1988.
- 2) 加藤工明、有田隆也、曾和将容、"長命令語 (LIW) コンピュータにおける命令実行遅延方式"、電子情報通信学会論文誌、Vol. J74-D-I, No. 9, pp. 613-622、1991.
- 3) 高木浩光、有田隆也、曾和将容、"問題が持つ先行関係のみを保証する高速な静的実行順序制御機構"、情報処理学会論文誌、Vol. 32、No. 12、pp. 1583-1592、1991.
- 4) 高木浩光、有田隆也、曾和将容、"重複可能なバリア型同期のためのスケジューリングアルゴリズムとその性能"、1991年並列/分散/協調処理に関する大沼サマー・ワークショップ、CPSY91-15、1991.