

マルチコアプロセッサの決定論的機能シミュレータのRISC-V対応と高速化

情報科学科 坂 一哲

指導教員：佐々木 敬泰

1 はじめに

コンピュータの性能に対する要求は年々高くなっており、これまで以上に高性能で低消費電力なプロセッサの開発が求められている。これらのプロセッサは設計が非常に複雑で、検証に膨大な時間やコストがかかるという問題がある。

プロセッサ検証では、ベンチマークプログラム全体の実行が望ましい。しかし、ハードウェア記述言語 (Hardware Description Language: HDL) をもとに行う HDL シミュレータの実行速度は低速であるため、ベンチマークプログラムの核となる部分までを高速なソフトウェアモデル上で実行する協調検証フレームワークが提案されている。マルチコアプロセッサの検証を行う場合、シミュレータが並列化されていないとコア数に比例して実行時間が増加するという問題がある。一方で、一般に並列プログラムの挙動は非決定的であり、試行毎に振る舞いに変化する。この挙動の変化は、同じ条件で繰り返し実行する必要のあるプロセッサ検証では容認できない。

このような問題を解決するために本研究室では、各コアが並列に動作し、かつ実行を繰り返しても結果が変化しない決定論的性質を持つシミュレータを提案している [1]。これを現在産学両方で広く使われている命令セットである RISC-V に対応させれば、様々な研究で利用可能になると考えられる。そこで本研究では、決定論的並列シミュレータの RISC-V 対応に必要な機構を調査し、その実装法を提案し、評価を行う。

2 提案手法

既存のシミュレータの問題を解決するために、以下のような手法を提案する。

2.1 概要

本研究では、文献 [1] の機能シミュレータを RISC-V に対応させることを目指す。文献 [1] で採用している MIPS32R2 命令セットと本研究の対象である RISC-V 命令セットでは命令や例外の処理、ABI 等が異なる。単純な命令は簡単な書き換え作業で対応できるが、同期命令やメモリ管理は MIPS32R2 と RISC-V で全く異なるため、それらについて新たに実装法を考える必要がある。

2.2 AMO 命令のための同期機構

アトミックメモリ操作 (Atomic Memory Operation, AMO) 命令は、メモリに関する複数の操作を他のコア等から見たときに一つの操作に見えるように行う RISC-V の命令である。MIPS32R2 には、LL/SC という同期機構が存在するが、AMO 命令に相当するものは存在しない。したがって、AMO 命令のための同期機構は新たに考える必要がある。

本研究で取り扱うシミュレータでは、ストア命令は未解決のメモリアクセスが決まった順番で行われ、互いに干渉しないように同期を行う。同期が完了してから、他のコアにメモリアクセスを許可するまでに行われるメモリ操作はアトミックになるという性質がある。この性質を利用することで、既存の実装と共通の仕組みを使って少ない実装量で AMO 命令が実装可能である。

表 1 基本命令の実行速度 (10 回の平均値)

	MIPS32R2	RISC-V
実行時間 (秒)	21.3	21.0
命令実行数	107M	93.7M
1 秒あたり命令実行数	5.02M	4.46M

2.3 仮想メモリ機構

仮想メモリとは限られたメモリ空間を仮想化し、メモリ容量を超えたアドレスを取り扱えるようにする機構である。通常時は TLB (Table Look-aside Buffer) に保存されたエントリをもとにアドレスの変換が行われるが、TLB に該当する行が無いとき、また仮想メモリの設定に変更が生じた時にはメモリ上に置かれたテーブルをもとに TLB の内容を更新する。MIPS32R2 では TLB の更新をソフトウェアで行う。一方で、RISC-V では TLB の更新はハードウェアで行われる。ソフトウェアは、RISC-V の要求するデータ構造をメモリ上に構築しておく必要がある。本研究で取り扱うシミュレータでは、システムコールエミュレーションを行う場合はシミュレータ側でテーブルの管理を行わなければならない。実装法として、スタートアップルーチンに組み込む方法とシミュレータの初期化時点で行う方法がある。

2.4 システムコールエミュレーション

本研究で取り扱うシミュレータには、システムコールエミュレーションを行うための仮想的デバイスがあり、そのデバイスのメモリ領域に値を書き込むことでシステムコールが実行される。MIPS32R2 が 32 ビット命令であるのに対して、本研究で実装するのは 64 ビット命令であるため、どの領域に何を書き込むべきか等の規約を新たに考える必要がある。

3 評価

現在実装が完了している基本命令を使って、フィボナッチ数を計算するプログラムを作成し、MIPS32R2 版と速度の比較を行った。結果は表 1 のとおりである。実行時間が変化していないことから、性能を落とすことなく RISC-V への対応ができていくことがわかる。

4 おわりに

本研究では決定論的性質を持つ機能レベルプロセッサシミュレータを RISC-V に対応させるために、MIPS32R2 と RISC-V の差異の調査と、MIPS32R2 にはない命令や機構の実装法を提案した。また、先行研究 [1] の同期機構の性質を活かすことでアトミック命令が容易に実装できることを明らかにした。今後はさらに実装を進めてゆき、性能の裏付けを行うとともに、AMO 命令の同期機構を改善する等の高速化手法も研究していく。

参考文献

- [1] 萱室, 他: マルチコアプロセッサの効率的な設計検証に向けたプロセッサシミュレータの並列化, 信学技報, Vol.117, No.278, pp.53-58(2017).