

Verilog HDL で記述する RISC-V 命令セットの アウトオブオーダー実行プロセッサ

藤浪 将[†] 眞下 達[‡] 吉瀬 謙二[‡]

[†]東京工業大学 情報工学科 [‡]東京工業大学 大学院情報理工学研究所

1 はじめに

教育, 研究分野において, オープンソースのプロセッサは有用である. しかし, 現状ではオープンソースの Out-of-Order 実行プロセッサは数が少ない. そこで我々は, 教育や研究に用いることを想定した Out-of-Order 実行を行うプロセッサ, rv-000 を開発した.

教育, 研究用プロセッサは自由かつ簡単に使える必要があるため, フリーの命令セットを用いていることや, プロセッサがオープンソースであること, コンパイラ及びアセンブラが利用可能であることが重要である.

我々はこれらの条件を満たす命令セットとして RISC-V[1] を選択した上で, rv-000 をオープンソースとして公開する予定である. RISC-V の Out-of-Order 実行プロセッサとして, BOOM[2] が存在するが, Chisel という独自の言語で記述されており導入コストが高い. 一方, 我々は rv-000 を標準的な Verilog HDL で記述した.

2 仕様及び実装

2.1 プロセッサ仕様

Out-of-Order は実装に多様性があるため, 公開にあたっては十分なドキュメントが提供されることが望ましい. そのため, rv-000 は書籍 *Modern Processor Design*[3] に沿って仕様を策定し, 理解を容易にする.

まず基本的な仕様について述べる. rv-000 は, アドレス, データともに 32bit の, 2 命令同時発行を行う Out-of-Order 実行プロセッサである. 例外や割り込み処理, システムレジスタなどは未実装であるため, OS を動作させることはできない. パイプラインについては, 図 1 に概略図を示す.

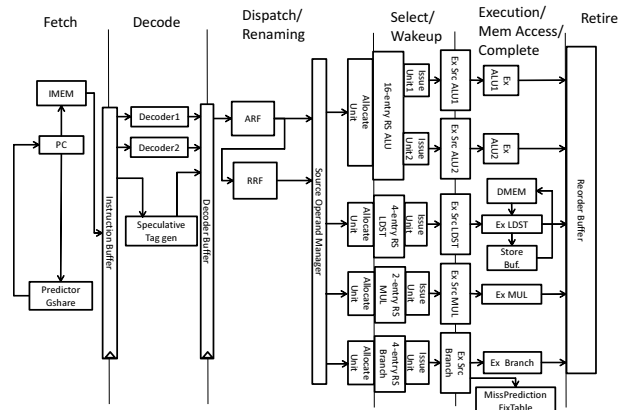


図 1: パイプライン

表 1: RS/演算器仕様

	ALU	LDST	分岐	乗算
RS エントリ数	16	4	4	2
演算器数	2	1	1	1
演算器レイテンシ	1	2	1	1

最後に Out-of-Order 実行の仕様について述べる. レジスタリネーミングを行う方式を採用し, 簡単のため, 分岐命令及びロード・ストア (LDST) 命令は In-Order 実行を行う. 分岐予測器には Gshare 方式を採用する. ストアバッファは, 32 エントリを持つ, 優先度付きの連想検索が可能なメモリであり, ロード命令を優先的に実行する. リザーベーションステーション (以下 RS と表記する), 演算器の仕様を表 1 に示す. RS は演算器ごとに用意する分散型を採用した.

2.2 実装

rv-000 の総ソースコード行数は 8558 行である. Vivado(2015.2) を用いて, rv-000 と, 同じ命令セットを持つ vscale[4] を, Virtex-7 XC7VX485T-2FFFG1761C 向けに論理合成した. vscale は, 3 段パイプラインを備えた In-Order 実行プロセッサである. 同時命令発行数は 1 であり, 分岐予測器は備えていない.

An Out-of-Order Processor of RISC-V ISA written in Verilog HDL

Masashi FUJINAMI[†], Susumu MASHIMO[‡] and Kenji KISE[‡]

[†]Department of Computer Science

Tokyo Institute of Technology

[‡]Graduate School of Information Science and Engineering

Tokyo Institute of Technology

表 2: リソース使用量及び動作周波数

	vscale	rv-000
Slice LUTs	2402	48488
Slice Registers	1073	14196
BRAM	0	2
DSP	0	16
Frequency[MHz]	136.4	53.46

合成は、命令メモリ及びデータメモリを除いて行った。合成結果として、リソース使用量及び動作周波数を表 2 に示す。vscale と比較して rv-000 のリソース使用量は非常に多く、BRAM 使用量が少ないが、これは 2 命令発行の Out-of-Order プロセッサの場合、メモリに必要なリード・ライトポート数が多くなるため、ほとんどのメモリはロジックを用いて生成されていることが要因の一つとなっている。また動作周波数に関して、rv-000 は vscale の半分以下となっているが、性能低下の原因となるため、動作周波数を上げることは今後の課題である。

3 性能評価及び考察

3.1 IPC, 実行時間

表 3 に、vscale と rv-000 で各アプリケーションを実行した際の IPC と相対性能を示す。イタリックで示したものは、ロード・ストア命令や分岐命令を極力排した最高性能測定用アプリケーションであり、アセンブリ言語で記述した。その他のアプリケーションについては C 言語で記述し、gcc(5.2.0) を用いて、最適化オプション-O2 をつけてコンパイルした。

結果から、rv-000 は多くのアプリケーションで vscale に対して高い IPC を達成している。しかし、fib や sort_3 では逆に IPC が低下している。また、行列積計算 matmul では IPC 比が大きいが、これは vscale では乗算器をロジックで実装しており、rv-000 では 1 サイクルで乗算できるのに対し、vscale では 32 サイクルかかるためである。

3.2 考察

rv-000 は vscale と比較してパイプラインの段数が深く、分岐予測失敗時のペナルティーが大きい。fib や sort_3 で rv-000 が vscale よりも性能が悪いのは、表 4 に示す通り他アプリケーションよりも分岐が多く、分岐予測失敗が多いためであると考えられる。

RISC-V コンパイラなどの開発環境は、RISC-V 公式サイトにドキュメントが豊富に用意されており、簡単に構築することが可能となっている。そのため、RISC-V

表 3: IPC

Application	vscale	rv-000	relative
fib	0.868	0.812	0.935
matmul	0.259	1.149	4.437
sort_3	0.857	0.779	0.909
hanoi	0.934	1.101	1.179
stencil	0.963	1.169	1.214
<i>simplecalc</i>	0.956	1.829	1.913
<i>noops</i>	0.990	1.918	1.937

表 4: 分岐予測内訳

Application	分岐命令数	予測ミス率
fib	1019	0.55
matmul	102	0.34
sort_3	454	0.45
hanoi	112	0.42
stencil	44	0.25

は教育、研究用プロセッサの命令セットとして適していると言える。

また、Verilog HDL で記述された、RISC-V 命令セットの Out-of-Order 実行プロセッサは未だ公開されておらず、新規性があると言える。ソースコードについては、プロセッサの中身が理解しやすいように 10000 行以内で記述した。

4 おわりに

本稿では、我々が開発した Out-of-Order 実行を行うプロセッサ、rv-000 の仕様及び実装について説明し、その性能に関する評価を行った。今後の課題としては、実際に FPGA 上で動作させることや例外や割り込みに対応して OS の動作を保証すること、プロセッサの性能を向上させること、除算、浮動小数点演算などの命令をサポートすること、オープンソースとして公開するために、ソースコードのリファクタリングや rv-000 に関するドキュメントを用意すること、などが考えられる。

参考文献

- [1] RISC-V(online), 参照先 (<http://riscv.org/>) (2016.01.07).
- [2] BOOM(online), 入手先 (<https://github.com/ucb-bar/riscv-boom>) (2016.01.07).
- [3] John Paul Shen, Mikko H. Lipasti: *Modern Processor Design: Fundamentals of Superscalar Processors*, (2013).
- [4] vscale(online), 入手先 (<https://github.com/ucb-bar/vscale>) (2016.01.07).