

# ヘテロジニアスマルチコアプロセッサのRISC-V移植に関する研究

柏森 風介 指導教員：佐々木 敬泰

## 1 はじめに

コンピュータの省電力化，高速化のために複数の異なるプロセッサコアを組み合わせるヘテロジニアスマルチコアプロセッサ (Heterogeneous Multicore Processor : HMP) が提案されているが，HMPはその設計に非常に大きな労力が必要となる．その問題を解決するためにHMPの自動設計ツールとしてFabHeteroが提案されている．しかし，この自動設計ツールは現在は商用ではほとんど利用されていない古い命令セットにしか対応していない．そこで本研究ではこの自動設計ツールを最新の命令セットであるRISC-Vに対応させることを目標に，命令セットの変更に伴い新たに実装する必要のある命令の実装方法を提案し，その命令を実行するLSUを作成・評価を行った．

## 2 関連研究

### 2.1 FabHetero

前述の通り，FabHeteroはヘテロジニアスマルチコアプロセッサの自動設計ツールである．現在，FabHeteroはプロセッサコア生成ツールであるFabScalar，キャッシュ自動設計ツールであるFabCache，およびバスの設計ツールであるFabBusから構成されている [1]．

### 2.2 AnyCore

FabScalarの採用している命令セットは古い命令セットであるため，最新の命令セットであるRISC-Vへの対応が求められている．そこでFabScalarをベースとして，RISC-Vプロセッサコア自動設計ツールであるAnyCore [2] が開発されているが，マルチコアプロセッサとして動作させるために必要な同期命令は未実装である．

### 2.3 RISC-V

RISC-Vは非営利団体であるRISC-V基金によって開発・運営されているフリーの命令セットであり，近年開発された最新の命令セットであること，仕様がオープンになっているオープンアーキテクチャであること，および機能ごとに命令をモジュールとして分割し，目的にあった命令のみを実装することができるモジュラー式という方式を採用していることが特徴である．

## 3 提案手法

FabHeteroをRISC-Vに対応させるには，FabHeteroのプロセッサコア生成ツールを現在利用しているFabScalarからAnyCoreに置き換えることで実現できる．

しかし現在のAnyCoreではマルチコア化するために必要な命令セットである同期命令がサポートされていない．そのため同期命令の実装方法を考案する必要がある．RISC-Vにおいて同期命令はRV32A拡張モジュールにまとめられており，LR/SC命令とAMO命令の二種類の命令がサポートされる [3]．

### 3.1 LR/SC 命令

LR/SC命令は不可分なメモリアクセスの実現に利用され，LR命令で値のロード及び監視アドレスの設定，SC命令で非介入のストアを保証する．FabScalarにも同種のLL/SC命令があるため，LR/SC命令の実装はLL/SC命令の実装を参考にした．

### 3.2 AMO 命令

AMO命令はAtomic-Memory-Operationの略称であり，この命令はメモリの指定したアドレスから値を取得し，その値に二項演算を行い，再び同じアドレスに演算結果を格納する．実行する二項演算としては表1がある．RISC-Vは命令セットの定義であり，実装方法はハードウェア設計者に委ねられている．そこで，本研究では以下のような実装方法を考案した．AMO命令の実装には図1のような専用のAMO命令実行モジュールを作成した．そして，各コアはそのモジュールに対してAMO命令要求を行い，命令の実行結果を受け取るというような実装を行った．

命令	動作
AMOSWAP	スワップ (値の入れ替え)
AMOADD	加算
AMOAND	論理積
AMOOR	論理和
AMOXOR	排他的論理和
AMOMAX	符号付き最大値
AMOMAXU	符号なし最大値
AMOMIN	符号付き最小値
AMOMINU	符号なし最小値

表1 AMO命令一覧

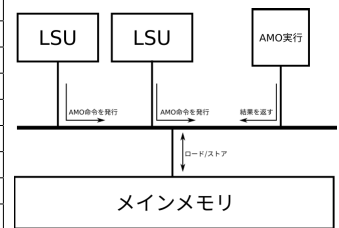


図1 AMO実行モジュール

## 4 提案手法の評価

今回の手法を実装したLSUをSystemVerilogで記述しシミュレーションにより動作を確認した．LR/SC命令では他のスレッドからの介入がある場合に失敗することを確認し，AMO命令ではメモリへの値の格納およびLSUへの結果の返答が成功していることを確認できた．

## 5 おわりに

本研究ではFabHeteroへAnyCoreを組み込むための前段階として，AnyCoreに対して実装する必要のある命令の実装方法を提案し，評価を行った．今後は本手法の面積評価，及び本手法を用いたAnyCoreのマルチコア対応を行う予定である．

## 参考文献

- [1] T. Nakabayashi, et al. FabHetero: An Environment for Developing Diverse Heterogeneous Multi-core Processors., 2nd International Symposium for Sustainability by Engineering at MIU, pp.288-291 (2012)
- [2] R. Chowdhury, et al. AnyCore: A Synthesizable RTL Model for Exploring and Fabricating Adaptive Superscalar Cores., Proceedings of the 2016 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS'16), pp. 214-224 (2016)
- [3] A. Waterman, et al. The RISC-V Instruction Set Manual Volume I: User-Level ISA Document Version 2.2, <https://riscv.org/wp-content/uploads/2017/05/riscv-spec-v2.2.pdf>, (2021-12-13)