

## 科学研究費助成事業（科学研究費補助金）研究成果報告書

平成25年 5月31日現在

機関番号：12608

研究種目：若手研究（B）

研究期間：2010～2012

課題番号：22700046

研究課題名（和文） コア融合機能を持つメニーコアプロセッサに関する研究

研究課題名（英文） Research on cooperative many-core processors

研究代表者

吉瀬 謙二（KISE KENJI）

東京工業大学・大学院情報理工学研究科・准教授

研究者番号：50323887

研究成果の概要（和文）： コア融合機能を持つメニーコアプロセッサとして開発を進めている CoreSymphony として、効率的で現実的なアーキテクチャを提案し、FPGA をターゲットとする実装をおこなった。この研究プロジェクトの主な貢献は次の通りである。(1) 効率的で現実的なコア融合機能を持つメニーコアアーキテクチャとして CoreSymphony を提案した。既存のアウトオブオーダー実行のコアをベースにして、アプリケーションのバイナリの互換性を維持しつつ、プロセッサのフロントエンド部分での通信を大幅に排除している。(2) CoreSymphony を実現するために解決すべきマイクロアーキテクチャ的な問題点を明確にした。(3) マイクロアーキテクチャ的な問題点を解決する洗練された方式を示し、その実装を明らかにした。(4) FPGA をターゲットとし、4 コア構成の提案アーキテクチャを実装することで、その実現可能性を明らかにした。

研究成果の概要（英文）： We have proposed CoreSymphony architecture that is one of the cooperative core architectures. In this research project, we design and implement efficient and realistic CoreSymphony and run it on FPGA. The research project makes the following main contributions: (1) Proposing efficient and realistic CoreSymphony, the cooperative core architecture that forbids most of communication from front-end, adopts an out-of-order core as a baseline, and does not need binary modification. (2) Clarifying the problems on microarchitecture in order to realize CoreSymphony. (3) Designing and implementing CoreSymphony to solve these problems. (4) Showing possibility of the CoreSymphony architecture.

交付決定額

(金額単位：円)

	直接経費	間接経費	合計
2010年度	900,000	270,000	1,170,000
2011年度	1,300,000	390,000	1,690,000
2012年度	800,000	240,000	1,040,000
年度			0
年度			0
総計	3,000,000	900,000	3,900,000

研究分野：総合領域

科研費の分科・細目：情報学・計算機システム・ネットワーク

キーワード：計算機アーキテクチャ

## 1. 研究開始当初の背景

典型的なメニーコアプロセッサでは、アプ

リケーションが持つスレッドレベルの並列性を活用し、チップ上に配置される多数のコ

アにうまくタスクあるいはスレッドを割り当てることで処理の高速化を目指す。しかしながら、プログラムに存在する逐次処理がプロセッサの性能を制限する。

メニーコアプロセッサにおいてアプリケーションを実行している際に現れる部分的な逐次処理を高速化するアプローチとして、一時的に、複数のコアを協調動作させることで逐次実行能力の高い1つの仮想コアを作り出すコア融合と呼ばれる技術がある。そのようなコア融合の先駆的な研究としては CoreFusion および Federation がある。一方、我々が提案しているコア融合方式である CoreSymphony は、コアの独立性の維持、アーキテクチャ技術の連続性の維持、バイナリの連続性の維持という3つの挑戦的な課題を同時に達成することを目指しており、挑戦的なコア融合のアーキテクチャである。

## 2. 研究の目的

チップに搭載するコアの数を増やすことでプロセッサのピーク性能を向上できる。一方で、コアの数が多くなるに従って、高い実効性能の維持が困難になる。例えば、プログラムに存在する並列化が困難な処理（逐次処理）がプロセッサの性能を制限する。本研究では、幾つかの発行幅の狭いプロセッサコアを協調動作（コア融合）させることで発行幅の広い仮想コアを形成する。このコア融合を用いて逐次処理部分を加速するメニーコアプロセッサアーキテクチャの確立を目指す。

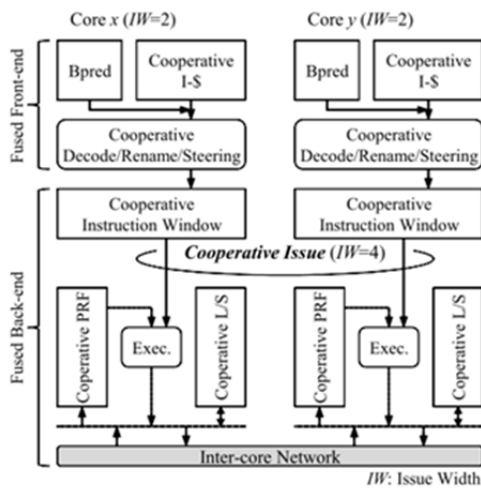


図1 コア融合の概要

本研究におけるコア融合とは、我々が CoreSymphony アーキテクチャとして提案しているものである。その概要を図1に示す。サイクルあたり2命令を実行する従来のコ

アが最大で4個融合することで、サイクルあたり8命令を実行できる高性能なコアとして見せかける仕組みである。

メニーコアプロセッサの定量的評価においては、従来から用いられているソフトウェアシミュレータのみの評価は莫大な時間を要する。このため、メニーコアプロセッサアーキテクチャ研究のインフラストラクチャとして FPGA を用いる現実的で洗練されたプロトタイプシステムを開発する。

## 3. 研究の方法

本研究の期間は平成22年度から24年度の3年間であり、次の項目を実施する。

初年度(平成22年度)は、効率的なアーキテクチャの開発をおこなう。また、メニーコアプロセッサシミュレータの構築を重点的に実施する。

平成23年度以降には、アーキテクチャの改良に加えて、大規模なシステム評価のためのFPGAカードを用いたハードウェアプロトタイプシステムの開発をおこなう。また、提案方式の有用性を明らかにする。

## 4. 研究成果

平成22年度は、アーキテクチャを評価するためのソフトウェアシミュレータの構築をおこなった。これはメニーコアプロセッサの挙動をサイクルレベルで模倣するシミュレータである。シミュレーション速度は重視せず、早期に正しく動作するシミュレータを構築することに重きを置いた。C++言語を用いて、逐次プログラムとして実装をおこなっている。また、効率的な CoreSymphony アーキテクチャの基本設計をおこなった。

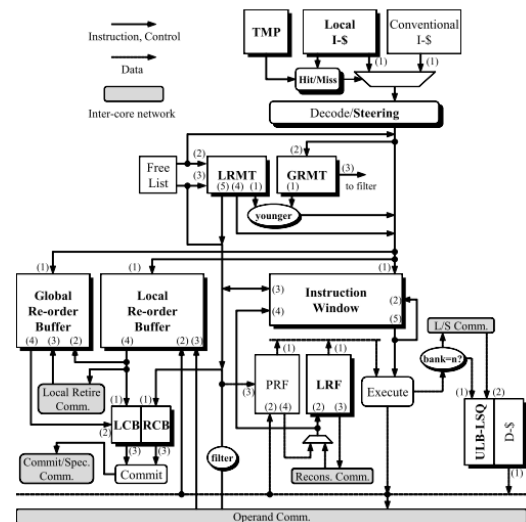


図2 提案アーキテクチャの構成

平成23年度は、コア融合機能を持つメニーコアプロセッサとして開発を進めているCoreSymphonyアーキテクチャの改良および部分的な実装をおこなった。複雑なハードウェアを必要としていたフロントエンド部を見直し、現実的な構成のフロントエンドの構成を提案した。具体的には、複数分岐予測器の一つであるTree-based Multiple Branch Predictorを用いることで分岐予測のハードウェアの軽減を実現した。また、プロセッサバックエンドについても、ハードウェアの複雑さを削減する仕組みの検討をおこなった。提案アーキテクチャの構成を図2に示す。プロセッサの実装に関しては、主に、プロセッサのフロントエンド部分をハードウェア記述言語にて記述し、FPGAをターゲットとして論理合成することでハードウェア規模と複雑さを評価した。その結果、コア融合の仕組みがプロセッサ全体の動作周波数に悪影響を与える可能性が低いことを明らかにした。

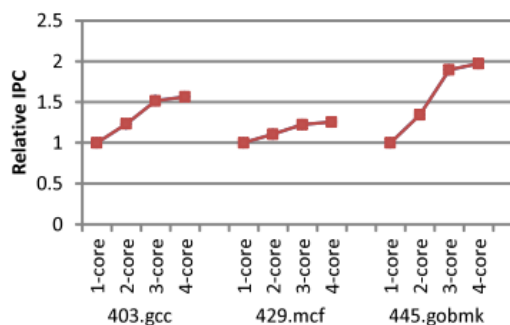


図3 コア融合による性能向上

最終年度の平成24年度は、コア融合プロセッサ全体として、大規模のFPGAボードをターゲットとして論理合成し、サンプルアプリケーションを動作させて、その論理を検証した。また、実装したコア融合プロセッサの動作速度およびハードウェアの複雑さを詳細に評価した。幾つかのベンチマークプログラムにおいて、融合するコアの数を1, 2, 4に変更させた場合の性能向上(Instructions Per Cycle)の評価結果を図3に示す。

FPGAをターゲットに論理合成をおこない、必要となるハードウェア資源を求めた結果を図4に示す。これは、レジスタファイルのためのハードウェア量をまとめたもので、既存の方式で高速化する右端の場合と比較して、提案方式において大幅にハードウェア量

が削減できることが示されている。

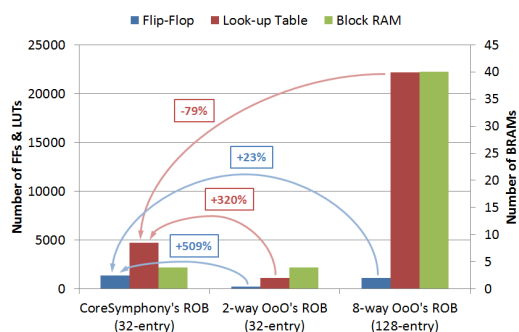


図4 レジスタファイルのハードウェア量

この研究プロジェクトの主な貢献は次の通りである。

効率的で現実的なコア融合機能を持つメニーコアアーキテクチャとしてCoreSymphonyを提案した。従来のアウトオブオーダー実行のコアをベースにして、アプリケーションのバイナリの互換性を維持しつつ、プロセッサのフロントエンドでの通信の大部分を排除している。

CoreSymphonyを実現するために解決すべきマイクロアーキテクチャ的な問題点を明確にした。

マイクロアーキテクチャ的な問題点を解決する洗練された方式を示し、その実装について明らかにした。

FPGAをターゲットとして、4コア構成の提案アーキテクチャを実装し、その実現可能性を明らかにした。

## 5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文] (計 2件)

- ① Tomoyuki Nagatsuka, Yoshito Sakaguchi, Takayuki Matsumura, Kenji Kise, CoreSymphony: An Efficient Reconfigurable Multi-core Architecture, COMPUTER ARCHITECTURE NEWS, 査読有, Vol. 39, No. 4, pp. 32-37, DOI: 10.1145/2082156.2082165 (September 2011).
- ② 若杉祐太, 坂口嘉一, 吉瀬謙二, 協調可能スーパースカラCoreSymphony, 情報処理学会論文誌コンピューティングシステム, 査読有, Vol. 3, No. 3, pp. 67-87 (September 2010).

[学会発表] (計 6件)

- ① 永塚智之, 吉瀬謙二, 効率的な CoreSymphony アーキテクチャの実装, 情報処理学会第 75 回全国大会, 於 東北大学 川内キャンパス(2013年3月8日発表) (March 2013).
- ② Tomoyuki Nagatsuka, Yoshito Sakaguchi, Kenji Kise, CoreSymphony Architecture, ACM International Conference on Computing Frontiers, pp.249-252, 於 Cagliari イタリア(2012年5月16日発表) DOI: 10.1145/2212908.2212945 (May 2012).
- ③ 永塚智之, 吉瀬謙二, CoreSymphony における命令キャッシュの効率化, 情報処理学会第 74 回全国大会, 於 名古屋工業大学 御器所キャンパス(2012年3月6日発表) (March 2012).
- ④ 永塚智之, 坂口嘉一, 松村貴之, 吉瀬謙二, CoreSymphony の実現に向けた高性能フロントエンドアーキテクチャ, 情報処理学会研究報告 2011-ARC-195, 於 沖縄県立博物館・美術館(2011年4月13日発表), pp.1-8 (April 2011).
- ⑤ 永塚智之, 吉瀬謙二, CoreSymphony における分岐予測器の分散化, 情報処理学会第 73 回全国大会, 於 東京工業大学 大岡山キャンパス(2011年3月2日発表), Vol.1, No. 1H-5, pp.59-60 (March 2011).
- ⑥ 若杉祐太, 坂口嘉一, 吉瀬謙二, CoreSymphony アーキテクチャ, 先進的計算基盤システムシンポジウム SACSIS2010 論文集, 於 奈良県新公会堂(2010年5月27日発表), pp.53-62 (May 2010).

[その他]

受賞

- ① 永塚智之: 情報処理学会第 73 回全国大会 学会推奨卒業論文 「コア融合プロセッサのフロントエンドアーキテクチャに関する研究」 (May 2011).

## 6. 研究組織

### (1) 研究代表者

吉瀬 謙二 (KISE KENJI)

東京工業大学・大学院情報理工学研究科・  
准教授

研究者番号: 50323887

### (2) 研究分担者

なし

### (3) 連携研究者

なし