

令和 4 年 6 月 1 日現在

機関番号：12601

研究種目：基盤研究(B) (一般)

研究期間：2019～2021

課題番号：19H04077

研究課題名(和文) 省電力・省面積・高効率を実現するユニバーサルなコアプロセッサIPの開発

研究課題名(英文) Development of Universal Processor Core IP with Area and Power Efficiency

研究代表者

入江 英嗣 (Hidetsugu, Irie)

東京大学・大学院情報理工学系研究科・准教授

研究者番号：50422407

交付決定額(研究期間全体)：(直接経費) 13,400,000円

研究成果の概要(和文)：本課題ではプロセッサの電力効率や性能を飛躍的に向上させる新しいコンピュータアーキテクチャ「STRAIGHT」の研究開発を行った。STRAIGHTは独自のコンピュータアーキテクチャで、従来のRISCコンピュータとは異なり、レジスタ上書きが発生しない特徴を持つ。本課題では命令セット策定からコンパイラ、マイクロアーキテクチャ、実チップまで総合的に開発を進め、TSMC28nmプロセスで製作された試作チップで高性能実行を確認した。この成果は、多くのユーザおよび企業が試用/実用できるようにオープン化されている。

研究成果の学術的意義や社会的意義

本研究はCPUをはじめとするあらゆるプロセッサの効率化に資する新しいコンピュータアーキテクチャとして、オペランドを距離で表現する命令セットの具体例を明らかにし、あたらしいクラスのコンピュータアーキテクチャの基礎技術確立および性能・設計上の優位点を明らかにしたものである。成果には実用的な命令セット仕様、C言語およびC++言語に対応したコンパイラ、実行モデルシミュレータ、さらに28nmプロセスによる高性能実チップが含まれ、基本アセットは公開にされて使用可能な状態となっている。

研究成果の概要(英文)：In this project, we conducted research and development of a new computer architecture, "STRAIGHT," which dramatically improves the power efficiency and performance of processors. STRAIGHT is a unique computer architecture that differs from conventional RISC computers in that it does not cause register overwriting. In this project, we have comprehensively developed STRAIGHT from the instruction set, compiler, micro-architecture, and actual chip, and confirmed high-performance execution on a prototype chip fabricated in TSMC's 28nm process. The results are open to many users and companies for trial/practical use.

研究分野：コンピュータシステム

キーワード：プロセッサ アーキテクチャ コンパイラ マイクロアーキテクチャ グリーンコンピューティング ASIC FPGA チップ試作

科研費による研究は、研究者の自覚と責任において実施するものです。そのため、研究の実施や研究成果の公表等については、国の要請等に基づくものではなく、その研究成果に関する見解や責任は、研究者個人に帰属します。

## 1. 研究開始当初の背景

情報化社会の進展はコンピュータ、特にその中核であるプロセッサ性能の成長によって支えられている。近年では、この成長戦略で重要な位置を占めてきた半導体微細化の影響の鈍化が喫緊の課題として知られている。一方で、アプリケーションとして、深層学習やブロックチェーンなど一定の需要を見込める専用計算が存在感を増しており、様々な新規プレーヤによる独自システム・オン・チップ(SoC)開発を動機付けている。SoCを実用的に駆動するためには、ホストとなるCPUコアが不可欠である。しかし、商用CPUコアのライセンスは新規プレーヤに敬遠される傾向にあり、これがIntel社、ARM社などのデファクトチップが現在、市場的には優勢にもかかわらず、RISC-VのようなオープンCPUアーキテクチャが社会から強く求められ、勢力圏を塗り替えようとしている理由の一つとなっている。

しかしながら、これまでの知見を得て洗練されているとはいえRISC-Vの基本設計は従来RISCアーキテクチャであり、性能を向上させようとする爆発的に複雑さが増加し、面積や消費電力に影響することが知られるようになった。本来高性能CPUの計算能力および実行効率は高く、アクセラレータでは一部のワークロードしか効率化できないことから、SoCの効果を高めるためには高性能CPUコアは重要である。しかし高性能RISCアーキテクチャが抱える複雑さのために、高性能化のノウハウを握っているプレーヤは少なく、ライセンス料や発熱量、デザインの困難さなどの点で、活気のあるチップ開発分野での高性能CPUコアの浸透は妨げられている。

一方、国内外で提案された優れたいくつかのマイクロアーキテクチャから着想をえて我々のグループが開発しているSTRAIGHTアーキテクチャでは、新しい命令セットとコンパイラによりこの複雑さを取り除くことに成功している。このアーキテクチャは、従来用いられている論理レジスタの代わりに、値を生成する命令が何命令前か、という距離の形でオペランドを表現する。この表現形式に従って生成されたコードにはデータ逆依存が含まれない特徴があり、従来に比べて遙かに簡易なハードウェアでも性能を落とさずに実行が可能となる。開発は初期段階が終了し、その独創性と発展性は高く評価され、国際トップカンファレンスMCIROに採択され、国内チップメーカを始めとする多くの企業から高い注目を集めている。

## 2. 研究の目的

本研究の中核をなす学術的問いは、「爆発的に増加しようとしているカスタムSoC向けのホストコアとして、STRAIGHTアーキテクチャの性能・効率および設計容易性は合致するか、そして結果として活気ある分野のコンピューティングを後押しできるか？」である。この問いは更に、「実アプリケーションに新アーキテクチャが与える実際の効果はどの程度か？」と「多様なプレーヤにアーキテクチャを使って貰うためにどうすればよいか？」の問いに分けられる。

これらの問いを明らかにするため、本研究では実アプリケーションが実行可能となるまでSTRAIGHTアーキテクチャのエコシステムの開発を進展させ、この上で最適化技術を明らかにし、性能評価を行う。また、新しいアーキテクチャについて研究・商用コミュニティを広げるため、ハードウェア設計、チップ試作を行い、成果はソフトコアIPやリファレンスデザインの形でオープンコアIPとする。

## 3. 研究の方法

独自アーキテクチャ上で実アプリケーションを行うには多くの技術レイヤーでの開発が必要であり、性能向上を得るための最適化はさらに多くの技術レイヤーが複雑に絡み合った課題となる。そこで本課題では開発を以下の6つの小課題に分け、各レイヤーを並行して進めるとともに、成果を段階的に確定させながら研究を進める。

### 小課題1. 実アプリケーション実行のためのミドルウェア実装

STRAIGHT 評価のために複雑な実アプリケーションを実行するためには新しいアーキテクチャのためのコンパイル基本アルゴリズムのみならず、スタンダードライブラリ中に含まれるインラインアセンブリやOSとの連携を行うシステムコールなど多くのコードを実装する必要がある。本課題ではまずこの実装を集中的に行い、実アプリケーション性能を早期に明らかにするための環境を整備する。実装順序としては、まず、SPECベンチマークや深層学習アルゴリズムなどから複数本の実証用実アプリケーションを選択し、その実行に必要な実装から優先して行う。

また、STRAIGHTアーキテクチャはハードウェア簡略化の代わりにコンパイル時の命令配置によりコードクオリティが大きく変化するため、この最適化技術を明らかにする。

### 小課題2. 実アプリケーション実行を最適化するプロセッサパラメタの探索

STRAIGHTは従来RISCと同様のデバイスパラメタ構成でも、必要電力を削減した上で実行性能を20%近く向上させることが判明している。さらに、ハードウェアの複雑性が低減されているため、現行のプロセッサパラメタとは異なるパラメタ最適値があり、従来RISCの限界を上回る有

効な性能向上を得ることができると予想されている。そこで、小課題 2 ではシミュレータを用いて、実アプリケーション実行を対象としたパラメタ探索を行う。

#### 小課題 3. 発展型 STRAIGHT マイクロアーキテクチャの提案

メモリバイパシング、改良型ストアキューなどから、最も性能に有効な技術を導入し、その効果の評価する。もし、小課題 2 の時点で既に既存アーキテクチャを上回る性能が得られている場合には、IP 公開（小課題 4）の時期を早めることを優先する。

#### 小課題 4. FPGA 用 STRAIGHT アーキテクチャ IP（ソフトコア）公開

FPGA 上で動作可能な STRAIGHT アーキテクチャ RTL 記述を開発し、ソースコードおよび基本コンパイラをオープンとする。また、ソフトコアとしての性能・面積上の利点を示す。

#### 小課題 5. STRAIGHT1/2 アーキテクチャの開発

STRAIGHT アーキテクチャはレジスタ番号ではなく命令間距離でオペランドを指定するため、従来のコンピュータ命令とは異なる命令コーディング最適化が可能かつ有効である。端的には、差分指定ということから想像できるとおり、本方式は非常にビットあたりの情報効率が高い。我々の検討の結果、無理なく命令長を 16 ビットに圧縮可能であることが分かっており、すでに命令マッピングは終了している。これは命令サイズがそのまま半分になることを意味しており、命令キャッシュを大きくできない組み込み用途や、命令キャッシュ容量が不足するサーバ用途の CPU にとって、機種選定の際の決め手となる性質である。コンパイラ対応、マイクロアーキテクチャ対応を行い、シミュレータおよび FPGA 実装でこの効果を確認する。

#### 小課題 6. STRAIGHT チップ試作

小課題 2 および 4 より得られるハードウェア記述を元に ASIC チップを製造し、その性能を実証する。65nm プロセス、5mm 角のチップを想定した実装を行う。チップの規模的に予算が不足する場合には、アーキテクチャの核となる部分の専用チップ化を優先して実証する。

### 4. 研究成果

小課題 1 のコンパイラ開発は多くのコード開発を要したが当初予定よりも順調にすすみ C 言語のみならず C++コンパイラの開発が可能となるなど、予想以上の成果を得た。SPEC CPU ベンチマークに含まれる実アプリケーションがコンパイル可能となった結果、生存変数の多いプログラムでは当初予想されたとおり冗長な命令が多く生成される傾向が観測された。これに対し、STRAIGHT コンパイラが生成するコードクオリティを向上させる最適化技術開発を前倒して実施した。この結果冗長命令削減、コーナーケース対策、Aggressive Spill、Callee Save Register 導入、Path Equalization など STRAIGHT 独自のコンパイラ最適化技術の基礎を確立し、生成されるコードクオリティを大幅に高めることに成功した。これらの結果は国内外の会議に採択されている。また、このコードクオリティの想定以上の改善の結果、STRAIGHT プロセッサの優位性は、研究計画当初の DSA プロセッサの安価ホストコア候補としてだけでなく、メインストリームの高性能 CPU や高効率のハイエンドエッジ CPU としても発揮できることが明らかとなった。

小課題 2 のプロセッサパラメタ探索ではクラスタ計算機を用いて網羅的にシミュレーションを行い STRAIGHT アーキテクチャのスケラビリティをはじめとする新しい特徴が設計ポイントをどのように変化させるかを計測した。小課題 1 でのコードクオリティ向上の結果、現行プロセッサと同様のパラメタ同士の比較でも遜色ない性能を発揮すると同時に、現行プロセッサを上回るフロントエンド幅や ROB 規模の構成において特に効果が高いことが確認された。本課題採択以降、商用高性能 CPU でもこれらのパラメタ拡大が進んでおり、このようなパラメタに優位性をもつコンピュータアーキテクチャとして STRAIGHT の優位性は更に高まっている。また、当初計画よりもハードウェア実装が進展したことを受け、シミュレータのみでなく ASIC 実装を想定した設計データを用いて、面積・電力を考慮した探索を行った。

小課題 3 では実アプリケーション評価を受けて主にスケジューラ構成の検討を行ない、STRAIGHT 命令セットに親和性の高い cyclic スケジューラの詳細設計と評価を行った。

小課題 4 は想定よりも早く進み、プロセッサ全体の RTL 記述を早い段階で利用可能となった。STRAIGHT アーキテクチャが FPGA 上でソフトコアとして利用する場合もフロントエンド幅のポート数の面で優位性があることを示し、国際学会に採択された。

小課題 5 では、具体的な命令セット仕様を STRAIGHT lite として策定した。さらに派生型のアーキテクチャとして、距離指定型アーキテクチャの設計空間を広げる複数 RP アーキテクチャの研究へと進展した。

これら当初想定以上の進展と、プロセッサ優位性の確認から、小課題 6 は当初予定のプロセスノードよりも微細な 28nm プロセスを想定して設計を行った。製造されたチップ上においてプロ

グラムの実行が確認されている。

小課題 1-6 の成果のうち最適化コンパイラアルゴリズムは特許化され、命令セット仕様、アセンブラ、基本コンパイラ、シミュレータ、RTL は公開されて利用可能となっている。

5. 主な発表論文等

〔雑誌論文〕 計1件（うち査読付論文 1件/うち国際共著 0件/うちオープンアクセス 0件）

1. 著者名 野村 隼人, 入江 英嗣, 坂井 修一	4. 巻 12
2. 論文標題 適応型Stubbornキャッシュマネジメント手法の提案	5. 発行年 2019年
3. 雑誌名 情報処理学会論文誌コンピューティングシステム	6. 最初と最後の頁 76-86
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計14件（うち招待講演 2件/うち国際学会 4件）

1. 発表者名 Susumu Mashimo, Akifumi Fujita, Reoma Matsuo, Seiya Akaki, Akifumi Fukuda, Toru Koizumi, Junichiro Kadomoto, Hidetsugu Irie, Masahiro Goshima, Koji Inoue and Ryota Shioya
2. 発表標題 An Open Source FPGA-Optimized Out-of-Order RISC-V Soft Processor
3. 学会等名 Int. Conf. on Field-Programmable Technology (国際学会)
4. 発表年 2019年

1. 発表者名 Tomoki Nakamura, Toru Koizumi, Yuya Degawa, Hidetsugu Irie, Suichi Sakai and Ryota Shioya
2. 発表標題 T-SKID: Timing Skid Prefetcher
3. 学会等名 The Third Data Prefetching Championship (in conjunction with ISCA 2019) (国際学会)
4. 発表年 2019年

1. 発表者名 中村 朋生, 塩谷 亮太, 入江 英嗣, 坂井 修一
2. 発表標題 キャッシュ圧縮による余剰領域を利用した仮想プリフェッチ・バッファ
3. 学会等名 cross-disciplinary Workshop on Computing Systems, Infrastructures, and Programming
4. 発表年 2019年

1. 発表者名 出川 祐也, 中村 朋生, 渋江 陽人, 入江 英嗣, 坂井 修一
2. 発表標題 フェーズ検出と事前学習を利用したプリフェッチャ動的切り替え手法の検討
3. 学会等名 cross-disciplinary Workshop on Computing Systems, Infrastructures, and Programming
4. 発表年 2019年

1. 発表者名 小泉 透, 塩谷 亮太, 入江 英嗣, 坂井 修一
2. 発表標題 リネームレスアーキテクチャに適するオペランド表現
3. 学会等名 cross-disciplinary Workshop on Computing Systems, Infrastructures, and Programming
4. 発表年 2019年

1. 発表者名 光野 聡志, 小泉 透, 門本 淳一郎, 入江 英嗣, 坂井 修一
2. 発表標題 Out-of-Order STRAIGHTソフトプロセッサの実装と評価
3. 学会等名 電子情報学会技術研究報告
4. 発表年 2020年

1. 発表者名 出川 祐也, 中村 朋生, 小泉 透, 塩谷 亮太, 入江 英嗣, 坂井 修一
2. 発表標題 タイミング調整を行うプリフェッチャに対して投機ミスが与える影響の解析
3. 学会等名 情報処理学会研究報告システム・アーキテクチャ
4. 発表年 2019年

1. 発表者名 小泉 透, 入江 英嗣, 坂井 修一
2. 発表標題 パイプライン化された演算器を生かした低レイテンシ指数関数実装
3. 学会等名 情報処理学会研究報告システム・アーキテクチャ
4. 発表年 2019年

1. 発表者名 中村 朋生, 小泉 透, 出川 祐也, 稲岡 航大, 有馬 裕一郎, 塩谷 亮太, 入江 英嗣, 坂井 修一
2. 発表標題 RISC-V上で実行するSPEC CPU 2017のSimulation Point解析
3. 学会等名 情報処理学会研究報告システム・アーキテクチャ
4. 発表年 2019年

1. 発表者名 門本 淳一郎, 入江 英嗣, 坂井 修一
2. 発表標題 オープンソースなマルチポートメモリコンパイラの検討
3. 学会等名 情報処理学会研究報告システム・アーキテクチャ
4. 発表年 2019年

1. 発表者名 Hayato Nomura, Tomoki Nakamura, Toru Koizumi, Hidetsugu Irie, Shuichi Sakai
2. 発表標題 Preliminary Discussion of a Time Stride Prefetching
3. 学会等名 IEEE Symp. on Low-Power and High-Speed Chips and Systems (国際学会)
4. 発表年 2019年

1. 発表者名 Hidetsugu Irie
2. 発表標題 Towards Ultra Flexible Computers:a Novel CPU Architecture and an Interconnect Technology
3. 学会等名 Int. Conf. for for Top and Emerging Computer Scientists (keynote) (招待講演) (国際学会)
4. 発表年 2019年

1. 発表者名 佐藤 幸紀, 橋本 政朋, 入江英嗣, 小林悠記, 丸岡晃
2. 発表標題 今後のテクノロジーロードマップ
3. 学会等名 Japan Consortium for the Reconfigurable-hardware Next generation 2019秋
4. 発表年 2019年

1. 発表者名 入江 英嗣
2. 発表標題 STRAIGHT: レジスタリネーミングのないハザードレスアーキテクチャ
3. 学会等名 情報科学技術フォーラム, トップコンファレンスセッション (招待講演)
4. 発表年 2019年

〔図書〕 計0件

〔出願〕 計1件

産業財産権の名称 コンパイラ、コンパイル装置、およびコンパイル方法	発明者 入江 英嗣, 坂井 修一, 小泉 透, 中江 哲史, 福田 晃史	権利者 同左
産業財産権の種類、番号 特許、特願2019-095442	出願年 2019年	国内・外国の別 国内

〔取得〕 計0件



〔その他〕

<https://www.mtl.t.u-tokyo.ac.jp/~irie/publication>

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
研究分担者	坂井 修一  (Sakai Shuichi)  (50291290)	東京大学・大学院情報理工学系研究科・教授   (12601)	
研究分担者	塩谷 亮太  (Shioya Ryota)  (10619191)	東京大学・大学院情報理工学系研究科・准教授   (12601)	
研究分担者	門本 淳一郎  (Kadomoto Junichiro)  (10909386)	東京大学・大学院情報理工学系研究科・助教   (12601)	

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件

8. 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関
---------	---------