

科学研究費助成事業 研究成果報告書

平成 27 年 6 月 5 日現在

機関番号：14603

研究種目：基盤研究(A)

研究期間：2012～2014

課題番号：24240005

研究課題名(和文)高性能アクセラレーション基盤技術の研究

研究課題名(英文)A Research on Fundamental Technology for High Performance Acceleration

研究代表者

中島 康彦 (NAKASHIMA, YASUHIKO)

奈良先端科学技術大学院大学・情報科学研究科・教授

研究者番号：00314170

交付決定額(研究期間全体)：(直接経費) 21,500,000円

研究成果の概要(和文)：電力効率や性能見通しに難点があるマルチスレッディング機構を投入することなくデータ供給性能を一杯使う、低電力かつ性能見通しの良いアクセラレーション技術を創出した。従来型演算器アレイ型アクセラレータの弱点であるデータ伝搬オーバーヘッドを削減し、さらに、ベクトル演算機構としても利用可能な新しい多数演算器制御方式の12.5mm LSIを開発した。消費電力はわずかに0.88Watt、電力あたり性能は7.7GFlops/Wattに達した。

研究成果の概要(英文)：In contrast to traditional multi-threading platforms that are hard to improve power efficiency and hard to estimate performance, an acceleration technique that can fully utilize memory bandwidth and can easily fit to the performance and power model is established. A 12.5mm square prototype LSI of FU-array accelerator that can reduce the overheads of data movement and can work as vector mode has been designed. The power consumption is found to be only 0.88Watt and the performance per watt reaches 7.7GFlops/Watt.

研究分野：コンピュータ・アーキテクチャ

キーワード：演算器アレイ アクセラレータ ステンシル計算 ベクトル計算

1. 研究開始当初の背景

動作周波数が頭打ちになったプロセッサは、最近まで、数コアから構成されるマルチコアを経て、数十コアを集積するメニコアへ展開する動きを見せていた。しかし現在では、コア数増加ではなくアクセラレータ搭載の方向へ大きく舵を切っている。将来的に多数の高機能コアを集積する高並列処理機構が必要となることに疑いはないものの、現状では、多数の軽量コアを投入するメニコア方式とアクセラレータ方式には、学術的見地からも明らかな優劣がある。メニコアの一般的な実行モデルでは、並列化可能なフェーズでは多数の軽量コアを投入して並列処理を行い、そうでないフェーズでは単一コアによる処理を行う。問題は、並列化可能なフェーズにおける実行速度と、そうでないフェーズの実行速度のいずれについても、軽量メニコア方式ではその時代に達成可能な最高性能を本質的に享受できない点にある。そもそも高並列処理に適したフェーズでは、ベクトル演算に代表される SIMD 演算を動員するほうが高効率である。また、高並列処理に適さないフェーズでは、高度な分岐予測やハードウェアプリフェッチ機構を備えるスーパスカラ方式のほうが高性能を達成できる。特に、軽量メニコア方式は後者への対応能力に欠けるため、プログラム全体を高速化する際のボトルネックになりやすい。すなわち、高性能システムの実現を目指すプログラマにとって、汎用言語により統一的に記述したプログラムが低速な多数コア上で動作するモデルは満足は行かず。当面は、電力効率の高いアクセラレータと組み合わせるほうが、システム全体として高性能と高い電力効率を両立できる。また、ツールが提供する機能しか使えない制約があるとしても、必要な時のみアクセラレータを呼び出す方式のほうが、パス競合回避など不確定要素の多い並列プログラミングコストを削減できる。

2. 研究の目的

以上の課題を解決するための、3つの研究項目と目標を示す。

【1】限りあるデータ供給能力と演算器の関連付け(主にハードウェア構成)

プログラムの特性が「演算量 < データ量」である場合、いかなる高速化手法も有効ではないため省電力実行が得策である。一方、(A)「演算量 > データ量」の性質を利用し、(B)データ供給性能を使い切る仕組みと、(C)大量の演算機構を装備することにより、演算量に比例する高性能を達成できる(演算能力は元々過剰なため、対ピーク性能を議論することに意味はない)。アクセラレータの本質はこの3つであり、データ供給性能を動的に目一杯使うために高度なマルチスレーディング機構を投入したのが GPU である。本研究は、電力効率や性能見通しに難点があるマルチスレーディング機構を投入することなくデータ供給性能を目一杯使う、低電力(電力効率 10 倍)

かつ性能見通しの良い(チューニングコスト半減)アクセラレーション技術の創出に取り組む。

【2】アプリケーションとアクセラレータの関連付け(主にバイナリトランスレータ)

アクセラレータを導入する場合、汎用 CPU のコンパイラ等ツールチェーンに対してアクセラレータ用命令列を出力させる改造が必要である。しかし、汎用 CPU の種類 × アクセラレータの種類だけのツールチェーンや演算ライブラリを愚直に開発するのは非現実的である。また、ベクトル演算を導入しようにも、ベクトル型計算機からの撤退が相次いでおり、新しい言語仕様への追従等、技術の継承が途絶えている。さらに、普及した汎用命令セットに対して独自にベクトル命令を拡張する方法は、企業間の利害関係が絡むため敷居が極めて高い。本研究申請は、迅速な導入を可能とするために、新命令セットとコンパイラを開発するのではなく、汎用 CPU の命令列から動的にアクセラレータ用命令列を生成するバイナリトランスレーション技術の創出に取り組む。

【3】ハードウェア機構とアクセラレータ用命令列の動的適合(主に動的チューニング機構)

本研究構想では、汎用 CPU の命令列から動的に生成したアクセラレータ用命令列が、多数の演算器と内蔵メモリを相互接続したハードウェア機構に写像された後、幅優先実行するベクトル演算方式、深さ優先実行する演算器アレイ方式、あるいは、複合方式やその他の方式の組み合わせにより実行される。しかし、前述のように、汎用 CPU の種類 × アクセラレータの種類だけの組み合わせに対応するためには、動的な挙動の把握とフィードバックによるチューニング技術が不可欠である。アクセラレータの内部動作状態をモニタリングし、バイナリトランスレーション方法、および、アクセラレータ用命令列の写像方法の両方に対してチューニングの指針を示す技術開発に取り組む。

3. 研究の方法

【24年度】

【1】限りあるデータ供給能力と演算器の関連付け

低電力かつ性能見通しの良いアクセラレータを構成するには、プログラムの特性に合わせて演算器と内蔵メモリの接続をある程度静的に決定した上で、実行時に動的な微調整を行う方式が有望である。アクセラレーション技術における静的命令スケジューリングの代表格であるベクトル演算方式(幅優先実行)には、十分なデータ供給能力が必要であるため、内蔵メモリに留保したデータを一挙に処理できる場合にのみ有効である。一方、もう1つの代表格である演算器アレイ方式(深さ優先実行)は、最内ループの入力と出力のみをメモリアクセスに対応付け、演算の中間データを全て演算器間ネットワークに収容することにより、必要最低限のメモリスループで最大限の演算性能を発揮させる機構である。データが

外部メモリに存在する場合に、細いデータ供給を最大限に生かしつつ演算に見合うだけの電力による処理が可能である。以上の議論から、24年度は、(1)ベクトル演算方式、(2)演算器アレイ方式、(3)メモリ空間の不規則参照を見通し良く安定的に制御する技術の各々をプログラム中でのように区別して組み合わせることが可能かについて調査を行った。

[2]アプリケーションとアクセラレータの関連付け
バイナリトランスレーションは、プログラムの静的解析結果から(1)(2)(3)のいずれを重視するかを決定し、大枠に適した命令列を生成する方針で進めた。具体的には、(1)重視の場合、元の命令列をベクトル演算可能な形式に変換する。アドレスが規則的に変化するLD命令を起点とする命令をベクトル命令と見なし、書き込み先を配列に読み替えて中容量レジスタへの写像を可能とする。一方、(2)重視の場合、研究申請者の従来研究を応用し、限られたデータ供給能力に合わせた演算の流れを作る。(3)重視の場合、ギャザやスキャット機能を有するデータ転送機構を大容量メモリと中容量レジスタの間に配置し、(1)または(2)が無駄なくオーバーラップする命令列を生成する。

[3]ハードウェア機構とアクセラレータ用命令列の動的適合

本研究では低電力化を優先するためマルチスレッディングを使用しない。生成した命令列とハードウェア資源の間に過不足が生じた場合、アクセラレータ本体が命令列を幅・深さ方向に拡張・縮小したり、中間結果を一時保存する中容量レジスタを挿入・削除する等により高速省電力動作を維持する機構を研究する。ハードウェア構成に最適な命令写像を探索するために、内部状態モニタから得られる情報により問題箇所を特定し、以降の命令列生成や拡張・縮小方針に反映する機構を考案する。

[25年度以降]

[1]要素技術の集約と機能検証

まず、ホストにはパソコン + Linux 上にバイナリトランスレータを搭載し、アクセラレータ部分には前述の高精度シミュレータを使用するプロトタイプを開発した。システムコールからシミュレータ起動・終結までのフローを確立し、アクセラレータのハードウェア設計を開始する前に機能検証および性能見積りを行い、システム構成上の問題点やボトルネックを明らかにした。なお、シミュレータは単なる予備評価ツールではなく、ハードウェア設計時に Verilog 記述の妥当性を確認する期待値比較作業の際の期待値生成ツールとして極めて重要である。このためにも、高精度シミュレータはタイミングを含めてハードウェアの構成を忠実に再現する必要があった。同様の比較のために、既存 CPU の SIMD 命令や既存メニコア向け命令列を生成して評価した。

[2]ハードウェア開発と性能評価

次に、アクセラレータの機能検証が完了した時点

で、ハードウェア設計を開始した。具体的には、HDL 記述可能なレベルに高精度化したシミュレータを参考に Verilog 記述を進めた。これまでの FPGA/LSI 開発に際しても同じ手順を採用しておりノウハウを最大限に活用した。設計したハードウェア記述は、FPGA 上で動作確認を行った後に LSI 化の工程に投入した。LSI 化後の試験環境の構築や実機と組み合わせでの稼働についても、これまで利用してきた LSI 実験設備を使用して迅速に行った。FPGA 化や LSI 化が完了した時点で高精度シミュレータを FPGA/LSI 呼出し機構に入れ換え、実機動作検証および性能・電力評価を行った。

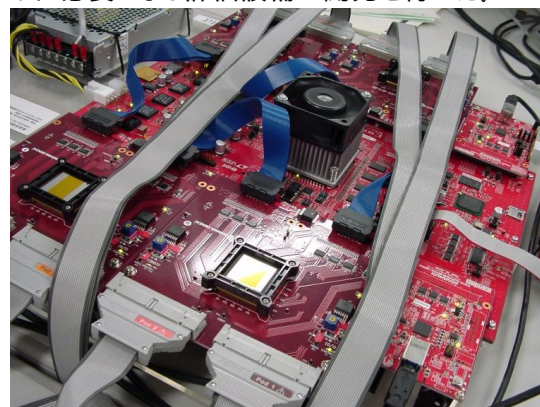
[3]アクセラレーション対象の拡大と自動化

現在、Intel 製 CPU を搭載するホストを利用するのが一般的である。しかし、より低電力な ARM 社製 CPU を搭載するホストを利用できる環境が整ってきた。より低電力な ARM ベースシステムへの移行も視野に入れた。また、ベクトル演算機構では、除算や平方根等、出現頻度が低くパイプライン処理に向かない演算は、複数サイクルを投入して比較的低速実行するのが一般的である。しかし、加減乗算がアクセラレータにより飛躍的に高速化されると、除算等の性能ボトルネックが相対的に際立つ。このため、パイプライン化が困難な演算についても、幅方向および深さ方向の両方に演算を自動展開できる機構を利用したアクセラレーション方法を開発した。

4. 研究成果

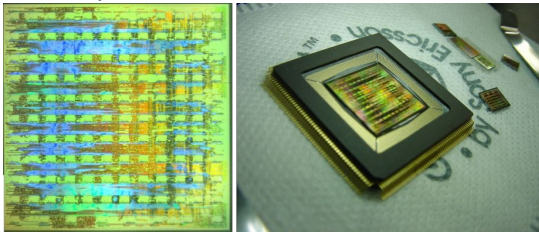
本研究の実施項目は大きく3つの柱からなる。

[1] 限りあるデータ供給能力と演算器の関連付け(主にハードウェア構成)では、電力効率や性能見通しに難点があるマルチスレッディング機構を投入することなくデータ供給性能を一杯使う、低電力(電力効率10倍)かつ性能見通しの良い(チューニングコスト半減)アクセラレーション技術の創出を目指した。H24年度は、従来型演算器アレイ型アクセラレータの弱点であるデータ伝搬オーバーヘッドを削減し、さらに、ベクトル演算機構としても利用可能な新しい多数演算器制御方式を考案し、アプリケーションの整備から、ハードウェア実行モデル(詳細なレジスタレベルのシミュレータ)の開発までを実施した。また、LSI化のために必要となる評価設備の開発を行った。H25



年度は、従来型演算器アレイ型アクセラレータの弱点であるデータ伝搬オーバーヘッドを削減し、さらに、ベクトル演算機構としても利用可能な新しい多数演算器制御方式を考案し、アプリケーションの整備から、ハードウェア実行モデル(FPGA実機上で動作中)の開発までを実施した。また、LSI化のために必要となる評価設備の開発を行った。

H26年度は、従来型演算器アレイ型アクセラレータの弱点であるデータ伝搬オーバーヘッドを削減し、さらに、ベクトル演算機構としても利用可能な新しい多数演算器制御方式の12.5mm LSIを開発し、評価ボード上で正常動作を確認した。消費電力はわずかに0.88Watt、電力あたり性能は7.7GFlops/Wattに達した。



[2] アプリケーションとアクセラレータの関連付け(主にバイナリトランスレータ)では、迅速な導入を可能とするために、新命令セットとコンパイラを開発するのではなく、汎用CPUの命令列からアクセラレータ用命令列を生成するバイナリトランスレーション技術の創出に取り組んだ。H24年度は、Intel-CCを使用した命令列生成手法、および、コードから多数演算器制御情報を生成するアルゴリズムおよび生成ツールの開発を行った。H25年度は、アプリケーションプログラムがレジスタレベルシミュレータ上で動作するとともに、構築が完了したFPGAプラットフォームでも動作した。H26年度は、バイナリトランスレータの開発を完了し、さらに、ステンシル計算に最適化したパラメタライズドライブラリの開発を行った。現在、アプリケーションプログラムが、LSI上で動作している。

[3] ハードウェア機構とアクセラレータ用命令列の動的適合(主に動的チューニング機構)では、幅優先実行するベクトル演算方式、深さ優先実行する演算器アレイ方式、あるいは、複合方式やその他の方式の組み合わせにより実行するチューニング技術に取り込んだ。H25年度に、Intel-CCの出力コードを元にアクセラレータコードを生成するツールを開発し、本機能を一部組み込んでいる。H26年度は、グラフ処理への適用手法に関して研究を進め、トランザクショナルユニットの追加による高性能化の可能性を示した。

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12件すべて査読有り)

Yuuki Shibata, Takanori Tsumura, Tomoaki Tsumura and Yasuhiko Nakashima: "An Implementation of Auto-Memoization

Mechanism on ARM-based Superscalar Processor", Proc. Int'l Symp. on System-on-Chip 2014, SoC2014, DOI=10.1109/ISSOC.2014.6972435, pp.1-8 (2014)

Yoshikazu Inagaki, Shinya Takamaeda-Yamazaki, Jun Yao, Yasuhiko Nakashima: "Performance Evaluation of a 3D-Stencil Library for Distributed Memory Array Accelerators", Proc. 2nd Int'l Workshop on Computer Systems and Architectures, CSA'14, DOI=10.1109/CANDAR.2014.100, pp.388-393 (2014)

Takanori TSUMURA, Yuuki SHIBATA, Kazutaka KAMIMURA, Tomoaki TSUMURA, Yasuhiko NAKASHIMA: "Hinting for Auto-Memoization Processor based on Static Binary Analysis", Proc. 2nd Int'l Workshop on Computer Systems and Architectures, CSA'14, DOI=10.1109/CANDAR.2014.49 pp.426-432 (2014)

Jun YAO, Yasuhiko NAKASHIMA, Naveen DEVISETTI, Kazuhiro YOSHIMURA, Takashi NAKADA: "A Tightly Coupled General Purpose Reconfigurable Accelerator LAPP and Its Power States for HotSpot-Based Energy Reduction", IEICE Trans., E97-D-12, DOI=10.1587/transinf.2014PAP0025, pp.3092-3100 (2014)

Jun Yao, Yasuhiko Nakashima, Mitsutoshi Saito, Yohei Hazama, Ryosuke Yamanaka: "A Flexible, Self-Tuning, Fault-Tolerant Functional Unit Array Processor", IEEE Micro, 34-6, DOI=10.1109/MM.2014.92, pp.54-63 (2014)

Tanvir Ahmed, Jun Yao, and Yasuhiko Nakashima: "A Two-Order Increase in Robustness of Partial Redundancy Under a Radiation Stress Test by Using SDC Prediction", IEEE Transactions on Nuclear Science, Vol61, Issue4, DOI=10.1109/TNS.2014.2314691, pp.1567-1574 (2014)

Tanvir AHMED, Jun YAO, Yuko HARA-AZUMI, Shigeru YAMASHITA, and Yasuhiko NAKASHIMA: "Selective Check of Data-Path for Effective Fault Tolerance", IEICETrans., Vol.J96-D, No.8, DOI=10.1587/transinf.E96.D.1592, pp.1592-1601 (2013)

Kazutaka KAMIMURA, Ryosuke ODA, Tatsuhiro YAMADA, Tomoaki TSUMURA, Hiroshi MATSUO, Yasuhiko NAKASHIMA: "A Speed-Up Technique for an Auto-Memoization Processor by Reusing Partial Results of Instruction Regions", Proc. 3rd Int'l. Conf. on Networking and Computing(ICNC'12), DOI=10.1109/ICNC.2012.17, pp.49-57 (2012)

吉村和浩, 中田尚, 中島康彦, 北村俊明: “異種命令セットアーキテクチャを持つ高電力効率 SMT プロセッサの開発”, 電子情報通信学会論文誌 D, Vol.J95-D, No.6, pp.1334-1346 (2012)

中田尚, 吉村和浩, 下岡俊介, 大上俊, Devisetti Venkatarama Naveen, 中島康彦: “画像処理向け線形アレイアクセラレータの性能評価”, 情報処理学会論文誌コンピューティングシステム, ACS38, Vol.5, No.3, pp.74-85 (2012)

岩上拓矢, 吉村和浩, 中田尚, 中島康彦: “時分割実行機構による演算器アレイ型アクセラレータの効率化”, 情報処理学会論文誌コンピューティングシステム, ACS39, Vol.5, No.4, pp.13-23 (2012)

齊藤光俊, 下岡俊介, Devisetti Venkatarama Naveen, 大上俊, 吉村和浩, 姚駿, 中田尚, 中島康彦: “線形演算器アレイ型アクセラレータを備えた高電力効率プロセッサの開発”, 電子情報通信学会論文誌 D, Vol.J95-D, No.9, pp.1729-1737 (2012)

[学会発表] (計27件中7件のみ記載)

Anna Zhang, Jun Yao, Yasuhiko Nakashima: “Lowering the Complexity of k-means Clustering by BFS-dijkstra method for Graph Computing”, IEEE Symposium on Low-Power and High-Speed Chips, Yokohama Bunka Center (Japan Yokohama), 2015年04月13日 ~ 2015年04月15日

Jun Yao, Yasuhiko Nakashima, Kazutoshi Kobayashi, Makoto Ikeda, Wei Xue, Tomohiro Fujiwara, Ryo Shimizu, Masakazu Tanomoto, Yangtong Xu, Xinliang Wang, Weimin Zheng: “XStenciler: a 7.1GFLOPS/W 16-Core Coprocessor with a Ring Structure for Stencil Applications”, IEEE Symposium on Low-Power and High-Speed Chips, Yokohama Bunka Center (Japan Yokohama), 2015年04月13日 ~ 2015年04月15日

Shohei Takeuchi, Thi Hong Tran, Shinya Takamaeda, Yasuhiko Nakashima: “A Parameterized Many Core Simulator for Design Space Exploration”, IEEE Symposium on Low-Power and High-Speed Chips, Yokohama Bunka Center (Japan Yokohama), 2015年04月13日 ~ 2015年04月15日

枝元正寛, 高前田伸也, 姚駿, 中島康彦: “データムービングボトルネックを解決するためのインテリジェントメモリスステムの検討”, 信学技報CPSY2014-91, 機械振興会館(東京都港区), 2014年12月01日 ~ 2014年12月02日

竹内昌平, 高前田(山崎)伸也, 姚駿, 中島康彦: “次世代アプリケーションのための包

括的なアーキテクチャ探索環境の検討”, 信学技報CPSY2014-89, 機械振興会館(東京都港区), 2014年12月01日 ~ 2014年12月02日

田ノ元正和, 高前田(山崎)伸也, 姚駿, 中島康彦: “メモリネットワークベースアクセラレータを用いた畳み込みニューラルネットワーク処理”, 信学技報CPSY2014-82, ビーコンプラザ(大分県別府市), 2014年11月26日 ~ 2014年11月28日

清水怜, 田ノ元正和, 高前田(山崎)伸也, 姚駿, 中島康彦: “メモリネットワークベースアクセラレータの試作と評価”, 信学技報CPSY2014-81, ビーコンプラザ(大分県別府市), 2014年11月26日 ~ 2014年11月28日

[図書] (計1件)

[編著] 中島康彦: “OHM大学テキストシリーズ コンピュータアーキテクチャ”, オーム社, ISBN:978-4-274-21253-6 2012

[産業財産権]

出願状況(計2件)

名称: データ供給装置及びデータ処理装置

発明者: 中島康彦, 姚駿

権利者: 奈良先端大

種類: 特許, PCT/JP2013/057503

出願年月日: 2013年3月15日

国内外の別: 外国

名称: メモリ内蔵アクセラレータの構成方法

発明者: 中島康彦, 高前田伸也

権利者: 奈良先端大

種類: 特許, 特願 2015-079552

出願年月日: 2015年4月8日

国内外の別: 国内

[その他]

ホームページ等

<http://arch.naist.jp/>

6. 研究組織

(1) 研究代表者

中島 康彦 (NAKASHIMA YASUHIKO)

奈良先端科学技術大学院大学・情報科学研究科・教授

研究者番号: 00314170

(2) 研究分担者

姚 駿 (YAO JUN)

奈良先端科学技術大学院大学・情報科学研究科・准教授

研究者番号: 40567153

(3) 研究分担者

原 祐子 (HARA YUKO)

奈良先端科学技術大学院大学・情報科学研究科・助教

研究者番号: 20640999