情報処理学会論文誌　コンピューティングシステム　Vol. 2　No. 1　158–172 (Mar. 2009)

演算加速機構を持つ オンチップメモリプロセッサの検討と電力性能評価

高橋睦史[†1,*1]　佐藤三久[†1]　高橋大介[†1]
朴泰祐[†1]　宇川彰[†1]　中村宏[†2,†1]
青木秀貴[†3]　澤本英雄[†3]　助川直伸[†3]

本稿では電力性能の向上に有効であるオンチップメモリプロセッサアーキテクチャSCIMAに，電力コストに有利な演算加速機構を導入することとし，その構成を検討し電力性能を評価する．演算加速機構としてベクトル型およびSIMD型の2種の方式を提案し，シミュレーションにより評価を行った結果，行列積演算および実アプリケーションであるQCD kernelにおいてはレジスタの要素数の差などの要因によりベクトル型がSIMD型の電力性能を上回り，全体としては主記憶バンド幅律速とならなければ，倍精度浮動小数点積和演算器（以降，FMA）の多いベクトル型がつねに優位であった．電力効率はベクトル型16FMAのときに最大となり，8コア時の電力効率は約1.58 GFLOPS/Wを示し，従来のプロセッサよりも高い電力効率を示すことが分かった．

Design and Power Performance Evaluation of On-chip Memory Processor with Arithmetic Accelerators

Chikafumi Takahashi,[†1,*1] Mitsuhisa Sato,[†1]
Daisuke Takahashi,[†1] Taisuke Boku,[†1]
Akira Ukawa,[†1] Hiroshi Nakamura,[†2,†1]
Hidetaka Aoki,[†3] Hideo Sawamoto[†3]
and Naonobu Sukegawa[†3]

In this paper, we design an on-chip memory processor with arithmetic accelerators, which is expected to be effective to improve power consumption. In addition, we evaluate power performance of the processor. We propose vector-type arithmetic accelerators and SIMD-type arithmetic accelerators into on-chip memory processor. The results of evaluation on our simulator indicate that the 4FMAs (Fused Multiply-Adders) Vector-type accelerator's performance exceeds the 4FMAs SIMD-type accelerator's on matrix multiplication and QCD kernel because of difference of the elements size of registers and so forth. The 16FMAs vector-type has advantage on almost all simulations excluding main memory bandwidth intensive benchmarks. Power effectivity is the maximum by vector-type 16FMAs, which indicates about 1.58 GFLOPS/W in 8 cores. It shows that the proposed architecture has advantage in power efficiency compared with existing processors.

1. はじめに

マイクロプロセッサはデバイス技術やアーキテクチャの改良により，その動作周波数を向上させてきた．その結果，現在では3 GHzを上回る動作周波数のプロセッサが現れるに至り，その高い動作周波数により高い性能を達成してきた．しかし代償としてプロセッサの消費電力は高くなり，集積度や実装に関して問題があることが明らかになってきた．最近では，比較的周波数を抑えたコアを1チップ内に集積したマルチコアプロセッサが使われるようになり，高性能システムにおいても低電力プロセッサを集積したBlueGene/L[1)]が注目されている．

本稿では高性能な演算処理が必要とされる科学技術計算を対象に，オンチップメモリとそれを効率的に活用する演算加速機構を用いることによって，より大きな演算あたりの電力効率が得られるプロセッサアーキテクチャについて検討し，その電力性能を評価する．

オンチップメモリプロセッサはプロセッサコアと同一チップ上にメモリを持つプロセッサである．チップ外の主記憶にアクセスするには相対的に大きな電力および時間的コストが必要となるが，オンチップメモリを利用することでオンチップメモリ・主記憶間のデータ転送をソフトウェアで効率的に制御でき，電力性能を改善できる．

他力，電力性能の改善方法としてプロセッサあたりの演算器を増やすことが考えられる．

†1 筑波大学
University of Tsukuba
†2 東京大学
The University of Tokyo
†3 株式会社日立製作所
Hitachi, Ltd.
*1 現在，株式会社ルネサステクノロジ
Presently with Renesas Technology Corporation

プロセッサコアより構成の単純な演算器を増やすことで，電力増加を最小限に抑えつつ性能を向上させることができる．そのためには効率的に演算器を利用するための機構が必要となる．

本稿では我々の提案しているオンチップメモリアーキテクチャSCIMA[2),3)]に演算加速機構を付加した高電力性能マルチコアプロセッサの構成を検討し，シミュレーションによる性能評価を行う．評価により各種アプリケーションにおいて演算加速機構の演算器方式やコア数などの各パラメータが実効性能と電力性能に与える影響を明らかにし，演算加速機構を持つオンチップメモリプロセッサの特性について考察する．

2. 演算加速機構を持つオンチップメモリプロセッサ

2.1 オンチップメモリ

オンチップメモリを用いたプロセッサアーキテクチャは数多く提案されている．我々はSCIMAと呼ぶアーキテクチャを提案している[2),3)]．本稿ではこのSCIMAを基本的なオンチップメモリアーキテクチャとして考える．SCIMAは高性能計算向けオンチップメモリプロセッサアーキテクチャであり，キャッシュと同じメモリ階層にSRAMを用いたオンチップメモリを実装する．オンチップメモリはメモリウェイごとに，すべてのデータ転送がソフトウェア制御可能な一時記憶か従来型のキャッシュとして使い分けることが可能である．オンチップメモリの利用には以下の利点がある．

(1) 演算に必要なデータを明示的に指定して転送することで，従来のキャッシュで生じていたような意図しないキャッシュミスや固定されたキャッシュラインサイズでのデータ転送で発生する不要なデータ転送を抑制する．

(2) 演算に必要となるより前にデータをオンチップメモリに転送しておくことでデータプリフェッチが明示的に行える．これはキャッシュプリフェッチとほぼ同等の機能である．しかしキャッシュプリフェッチとは異なり，プリフェッチしたデータが意図せずオンチップメモリから追い出されることはない．

(3) (2)によりオンチップメモリにあらかじめデータを転送しておくことで，オンチップメモリへのアクセスに関してアクセスするアドレスがコンフリクトしなければ，アクセスレイテンシが一定であることを保証できる．

(1)，(2)の利点により性能が向上し，加えて主記憶へのアクセスを削減することによって電力性能の向上も見込まれる．また，(3)によりアプリケーションプログラムの実装時に，オンチップメモリへのアクセスレイテンシを隠蔽するための最適化が容易になる．

演算加速機構をオンチップメモリプロセッサに導入した場合は，(3)によりプリロード命令を用いて，演算とオーバラップする効率的な命令スケジューリングが行える．また，従来研究で示されているように実効性能および電力性能はキャッシュを利用する場合よりも向上することが予想される[2),3)]．

本稿においてオンチップメモリは主記憶と直接接続することとする．またオンチップメモリは単層とし，演算加速機構と直接接続する．これは前述のとおり(3)によりアクセスレイテンシが予測可能とするためであり，また数十サイクル程度のレイテンシであれば，ソフトウェアにより十分レイテンシの隠蔽が可能であることから直接接続を想定する．これはハードウェアの複雑度を下げることにもつながる．

2.2 演算加速機構

多数の演算器を計算機に搭載する方法としては，Cell BE[4)]のように独立した命令ストリームで動作するシンプルな演算コアを持つ方法や，グラフィックアクセラレータやFPGAアクセラレータのように，プロセッサ外に付加される演算器を利用する方法などがある．

しかし，グラフィックプロセッサやFPGAの演算加速機構は特定の処理に関しては低い消費電力および回路コストで高い演算性能が得られる半面，その接続方式によっては大きなメモリバンド幅を得ることは難しく用途が限られる．高速なバスでアクセラレータを接続する方法として，AMD Torrenza[5)]のように，アプリケーションに特化したアクセラレータを「Coherent HyperTransport」と呼ばれるインターフェイスで汎用プロセッサに接続することも考えられる．

我々はすでに電力性能の向上が確かめられているSCIMAの利点を生かすため，プロセッサチップ内に演算加速機構を導入することとし，演算加速機構は既存のプロセッサコア（スカラコア）に付加される形で実装されることを想定する[6)]．このコアをプロセッサチップ内に複数配置し，前節で述べたオンチップメモリを共有する．各スカラコアはL1キャッシュを持ち，オンチップメモリに各コアの演算加速機構，L1キャッシュ，および主記憶が接続される．なお，オンチップメモリは，RAMとして使用可能な領域と，従来のL2キャッシュとして使用できる領域をWayごとに分割して，それぞれを同時に利用することができる．これはSCIMAの大きな特徴である．

以上より，検討するオンチップメモリプロセッサの基本構成を**図1**に示す．演算加速機構の実装方式としてはSIMD型演算加速機構およびベクトル型演算加速機構の2種類を検討する．

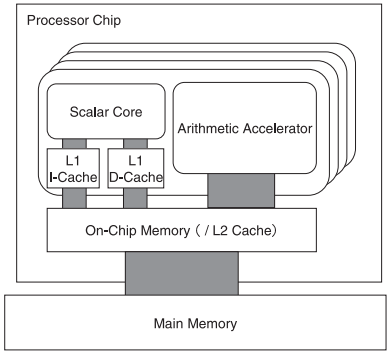


図 1　演算加速機構を持つオンチップメモリプロセッサの基本構成（4 コア時）
Fig. 1　Basic configuration of on-chip memory processor with arithmetic accelerators (a case of 4 cores).

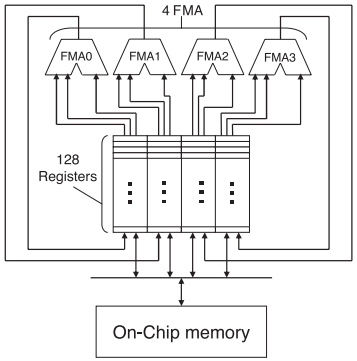


図 2　SIMD 型演算加速機構
Fig. 2　SIMD-type arithmetic accelerators.

2.2.1　SIMD 型演算加速機構

本稿で検討する SIMD 型演算加速機構を**図 2** に示す．この演算加速機構は 4 個の倍精度浮動小数点数積和演算器（Fused Multiply-Add，以降 FMA）を持ち，128 個の SIMD レジスタを持つ．1SIMD レジスタあたり 4 要素を持ち，演算時には 1 サイクルで 4 要素がそれぞれ対応する演算器へと送られる．SIMD 型演算加速機構では既研究[6] において，少ないレジスタ数ではスカラコアでのアドレス演算を行うことに起因するストールが発生することと，オンチップメモリへのアクセスレイテンシ隠蔽のために必要なループアンローリングを行う際にレジスタ数が不足することが判明した．そこで本 SIMD 型演算加速機構はロード・ストア命令用にインデックスレジスタを持ち，レジスタ数はループアンローリングに十分な 128 本とした．インデックスレジスタを用いることでアドレスを演算加速機構内に保持し，アドレスを再利用することやインデックスレジスタをベース値とした相対値アドレスによるロード・ストアに対応する．

SIMD 型演算加速機構の場合，比較的少ない要素を SIMD レジスタにロードして処理を行う．再利用される値をレジスタに保持しておき，最適化を図るレジスタブロッキングやレジスタ内要素間の交換命令を用いるなどベクトル型では行えない細かい最適化が可能である．SIMD 命令のオペランドは最大 3 オペランド入力 1 オペランド出力をサポートする．ただし，連続したベクトルを処理する場合，SIMD 型では 1 命令あたりの処理要素数がベクトル型より少ないため，ベクトル型演算加速機構と比較して単位演算あたりの命令発行数が最大で 32 倍になり，ベクトル型よりも不利となる．

電力の観点からは，ベクトル型に比べレジスタなどの機構が簡単なため，消費電力を抑えられる可能性がある．また，レジスタの再利用によりオンチップメモリと演算加速機構間のトラフィックを削減することで電力削減の効果が見込まれる．

Cell BE では，各 Synergistic Processor Element（SPE）は 128 ビット長 128 個のレジスタおよび 256 KB のローカルストアを持っているが，本稿で検討する SIMD 型演算加速機構では各アクセラレータは 256 ビット長 128 個のレジスタを持ち，8 MB のオンチップメモリを複数のアクセラレータで共有するという点で異なる．

2.2.2　ベクトル型演算加速機構

本稿で検討するベクトル型演算加速機構を**図 3** に示す．この演算加速機構は 16 個の FMA を持ち，32 個のベクトルレジスタを持つ．1 ベクトルレジスタは通常のベクトルプロセッサよりも少ない 16 要素 $\times 8 = 128$ 要素で構成される．1 サイクルに 1 ベクトルレジスタあたり 16 要素ずつ演算器に送り込まれ，これが 8 サイクル繰り返される．チェイニング機構を持つものとし，後続のベクトル演算のオペランドとして用いられる場合には，バイパスして後続の演算に供給される．なお，データ転送命令のアドレスはすべてスカラコア内で演算することとし，SIMD 型演算加速機構とは異なり特別なインデックスレジスタは用意せず，1 ロード・ストア命令ごとにアドレスをスカラコアから演算加速機構のロードストアユニットに渡すことを想定する．命令オペランドは SIMD 型と同様に最大 3 オペランド入力 1 オペランド出力をサポートする．

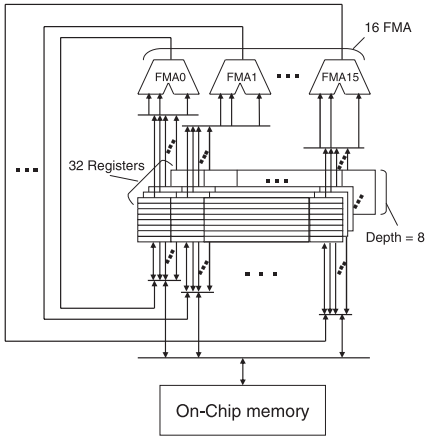


図 3　ベクトル型演算加速機構（16FMA 時）
Fig. 3　Vector-type arithmetic accelerators (a case of 16 FMAs).

本稿で想定する演算加速機構では L2 キャッシュ階層と同等のメモリ階層に置いたオンチップメモリにアクセスする．したがって従来のベクトル機とは異なりオンチップメモリへのデータ転送を行っておき，そこからのロード命令を発行する．また，オンチップメモリ上でのデータの再利用も行えるため，従来のベクトルプロセッサと比較して主記憶へのアクセスを削減することができる．また，再利用性を十分に活用できる場合，従来のベクトル機と異なりメインメモリとのバンド幅が小さい場合でも性能を発揮でき，結果として高い電力性能が得られることが見込まれる．

しかし，ベクトル演算器のベクトル長が大きい場合にはオンチップメモリに保持できるデータセットの数が少なくなり，再利用性を十分生かせない可能性がある．このため，オンチップメモリを組み合わせる場合にはベクトル長を大きくとることが難しい．この点で本稿で想定するベクトル型演算加速機構は通常のベクトルプロセッサと異なる．

なお，ベクトル型のアクセラレータを用いた場合の電力性能については，文献 7) でも議論されている．

本稿ではベクトル型と SIMD 型の演算加速機構について比較を行うが，両者のレジスタファイルの容量は異なっており，16FMA 時のベクトル型では 1 コアあたり 32 KB，4FMA の SIMD 型では 4 KB となっている．これはベクトル型と SIMD 型でレジスタ使用法が異なるため，結果として生じたものである．

最新のベクトルプロセッサである NEC の SX-9 においては，1CPU あたりのベクトルレジスタは 144 KB となっている．これに対し，SIMD 型の Intel 64 アーキテクチャの Xeon では 1 コアあたり 128 ビット長 16 個のレジスタを持ち，レジスタ容量は 256 B である．また，SIMD 型の Cell BE の SPE では 1 コアあたり 128 ビット長 128 個のレジスタを持ち，レジスタ容量は 2 KB となっている．このように，既存のプロセッサにおいては，SIMD 型はベクトル型よりもレジスタ容量が小さくなっていることが多いといえる．

これはいくつかの要因が考えられるが，大きな要因として演算器のレジスタ使用法の違いがあげられる．ベクトル型では通常，1 命令で演算器を複数回駆動する．これにより演算の行えるデータ要素の位置は各レジスタの中の特定の位置どうしの要素に限られるが，1 演算器の扱えるレジスタ要素数は大きくなる．対する SIMD 型では通常 1 命令で演算器を 1 回駆動するので，1 演算器の扱うレジスタ数が限られる．

本稿で検討する SIMD 型演算加速機構は 4 個の FMA を持ち，4 要素 128 個の SIMD レジスタを持つことから，FMA あたりのレジスタ容量は 1 KB となる．一方，ベクトル型演算加速機構は 16 個の FMA を持ち，通常のベクトルプロセッサよりも少ない 16 要素 $\times 8 = 128$ 要素で構成されたベクトルレジスタを 32 個持つことから，FMA あたりのレジスタ容量は 2 KB となる．

ここで，SIMD 型とベクトル型でレジスタ容量を同じにした場合，SIMD 型では 4 要素 256 個のレジスタを持つことになる．このようなレジスタ数の多い SIMD 型演算加速機構の実現は困難であると考えられ，ベクトルレジスタの要素数や個数をこれ以上少なくすることも現実的ではないと考えられるため，SIMD 型においてベクトル型よりも FMA あたりのレジスタ容量を小さくするのが妥当であると考えられる．

2.3　オンチップメモリと演算加速機構を用いるプログラム最適化

演算加速機構で演算を行う際には，いったんオンチップメモリに主記憶のデータを転送する必要があるが，そのデータ転送はソフトウェアで制御される．この手法については基本的な仕組みが同じであるので既存研究[2),3)] の手法を襲用することができる．オンチップメモリ上に転送し終わったデータに対しては演算加速機構による従来手法と同じ最適化が行える．

オンチップメモリを利用したプログラムは

(1)　利用するデータを選択して主記憶からオンチップメモリにデータを転送
(2)　オンチップメモリにあるデータを利用して演算
(3)　書き戻す必要のあるデータを主記憶に書き戻す

といった手順で実行される．このとき，データセットがオンチップメモリよりも大きい場合

は何らかのブロッキングを行う．この手順についてはベクトル型および SIMD 型の演算加速機構で共通の動作となる．データ転送は完了するまでにレイテンシがあるために，あらかじめ主記憶/オンチップメモリ間のデータ転送命令を発行しておくことで，このレイテンシを隠蔽できる．

なお，2.2.1 項でも述べたが，演算加速機構とオンチップメモリ間のデータ転送におけるレイテンシについても隠蔽するために先行してロード命令を発行する必要があり，その実現にはループアンローリングが基本的な実装方法となる．

オンチップメモリの利用はデータのアクセスパターンが予測可能であることが前提となる．アクセスパターンの予測困難なデータは，プログラミングもしくはコンパイル時にデータ転送の制御を明示することが難しく，基本的にオンチップメモリを用いた最適化は行えない．この場合は予測可能なデータのみをオンチップメモリを用いて最適化を行い，それ以外のデータは従来どおりにキャッシュを利用することで解決する．

オンチップメモリプロセッサ向けのプログラム最適化にはキャッシュと同様に時間的局所性を利用する方法と，オンチップメモリをストリームバッファ的に使用する方法がある．ストリームバッファとして利用する場合は各コアがオンチップメモリの異なる領域をそれぞれ利用してプリロードなどを行うことにより最適化を行う．時間的局所性を利用する場合は，あるコアがロードしたデータを各コア間で共有する方法と各コアが個別にデータをロードする方法が考えられる．前者の最適化では各コア間でデータを共有できた場合，後者と比較してメインメモリへのアクセスを削減することが可能である．しかし，同じオンチップメモリアドレスにアクセスが集中した場合は性能が低下する恐れがあるので，これを防ぐためにはアクセスタイミングの最適化が重要になる．

3. 性能評価環境

3.1 評価環境と評価用プログラム

想定するプロセッサアーキテクチャの実効性能と電力性能を評価するためにシミュレーションを行った．演算加速機構とオンチップメモリを付加するスカラコアとして低消費電力プロセッサである SH-4A プロセッサを想定し，GDB5.0 付属の SH エミュレータにクロックシミュレーション機能および演算加速機構シミュレーションのためにベクトル型および SIMD 型の演算加速機構を駆動するための命令セットを拡張したシミュレータを開発した．なお，命令ストリームはすべて SH-4A コアで処理されることとする．

データの依存関係は浮動小数点数の依存関係のみをシミュレーションし，それ以外の汎用レジスタなどにおけるデータは依存なく実行できるものと仮定する．また，命令分岐予測は 100%成立するとする．命令発行は SH-4A と同じくインオーダ型の Dual Issue スーパスケーラとし，演算加速機構の演算命令とロードストア命令は同時に発行できることとする．演算加速機構はロードストアユニットを 1 つ持つこととし，アクセスレイテンシは演算加速機構/オンチップメモリ間は 20 サイクル，オンチップメモリ/主記憶間は 100 サイクルとする．

シミュレーションはベクトル型と SIMD 型について，FMA 数をベクトル型は 16 個と SIMD 型は 4 個としてシミュレーションを行う（以降，“Vector-16FMA” および “SIMD-4FMA”）．またベクトル型と SIMD 型の演算器数による差異以外の特性の比較を行うために 4 個の演算器を持つベクトル型のシミュレーションもあわせて行う（以降，“Vector-4FMA”）．

なお，スカラコア（L1 キャッシュ），演算加速機構およびオンチップメモリはスイッチインタコネクトでの接続を想定し，そのインタコネクトのオンチップメモリ側のポートのバンド幅をシミュレーションのパラメータとした．また，オンチップメモリの Read/Write ポートは演算加速機構側 1 ポート，主記憶側 1 ポートとし，各コアから同時にはオンチップメモリにアクセスできないこととする．

評価用プログラムには DAXPY ループ，Livermore kernel 1，3，7[8]，$C = A \times B$ で表される N 次の倍精度行列積演算，および素粒子物理学の実アプリケーションである Lattice QCD プログラム[9] を使用した．それぞれのプログラムは C 言語により実装し，演算加速機構を利用する部分についてはベクトルもしくは SIMD 命令セットをインラインアセンブラでバイナリを挿入した．

DAXPY ループ，Livermore kernel および行列積演算についてはデータセットをオンチップメモリに転送できるよう各ベクトルを分割しオンチップメモリに転送し，転送したデータをさらに分割して各コアで演算した．行列積演算については，最内側ループ内でのデータセットがオンチップメモリ容量を上回ることから，タイリング[10] を用いて行列について 2 次元分割を行ってオンチップメモリへロード後，再び 1 次元もしくは 2 次元分割を行って各コアにスレッドとして割り当てた．この方法では 2.3 節で述べたように主記憶とのデータ転送量を減少させられるがコア間でアドレスコンフリクトが発生する可能性がある．本稿では実装上の都合によりコンフリクトを回避する最適化は特に行わない．

Lattice Quantum Chromodynamics Simulation（以下，QCD と呼ぶ）について概要を述べる．QCD とは，4 次元格子空間上で 3 つの色で模式的に表現される，グルーオンによるクオーク間の色交換の関係をシミュレーションするもので，素粒子物理学において実際に

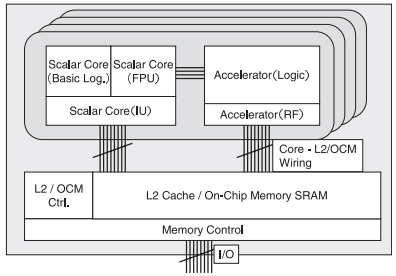


図 4　電力モデルにおける機能ブロック
Fig. 4　Functional blocks on the power estimation model.

用いられているアプリケーションである．実際の計算は BiCGStab[11] と呼ばれる iterative アルゴリズムとなっている．ソースには Aoki らによって作成されたベクトル計算機向けに最適化されたバージョン[12] を利用した．このプログラムでは X，Y，Z，T 次元方向の 4 重ループとなるアルゴリズムを，Z，T 次元についてループアンローリングを行うことによりベクトル長を大きくとる最適化が行われている．本評価ではこのプログラム中の実行時間の多くを占めるカーネルである MULT 部分のうち Z 方向演算のみ取り出して利用する．また，このプログラムでは X，Y，Z の 3 次元について並列化が行われているが，演算加速機構向けの最適化を行うために X 方向について 1 次元分割を行うよう並列化部分を改めた．

3.2　電力モデル

消費電力については現段階では本研究はアーキテクチャレベルの検討を行っているため，RTL 設計のデータの利用などによる詳細な見積りはできない．そこで本稿では想定するプロセッサについて機能ブロックごとに分割し，その機能ブロックの駆動回数あたりの消費電力単価を概算し，駆動回数を掛け合わせることで消費電力を求めることとする．

分割した機能ブロックのうちクロックツリーを除いた図を**図 4** に示す．まず各コア内のスカラコアについて命令発行ユニット（IU），基本論理，浮動小数点数演算ユニットに分割し，演算加速機構は論理とレジスタファイルに，オンチップメモリ部分は SRAM とコントロール部分とメモリインタフェースの 3 ブロックに分ける．そのほか，配線としてクロックツリー，演算加速機構/オンチップメモリ間配線，外部 I/O を考え，以上，計 11 ブロックに分割する．演算加速機構については乗算命令・ロードストア命令・それ以外の 3 種別に分けて駆動率を考える．スカラコア/オンチップメモリ間の配線は特に分けて考慮せずスカラコア部分の消費電力に含まれると仮定する．

電力はプロセスルールなどを考慮して，各機能ブロックについて想定より見積もったトランジスタ数や配線長をもとに論理量を導出し，電力単価と論理量を掛けたものに，論理ブロックについては機能ブロックを 1 回駆動することによって使用されるトランジスタの割合を表す活性化率を掛け，さらに機能ブロックの駆動率を掛け合わせることで求める．レジスタファイルなどのブロックでは駆動単位がポートアクセスになるなどするが，基本的に同様である．単位論理量あたりの電力単価を C_i，機能ブロックの論理量を L_i，活性化率を A_i，機能ブロックの駆動率（もしくは回数）を n_i とすると，電力 P は以下の式で求められる．

$$P = \sum_{i=1}^{N}((C_i \times L_i \times A_i) \times n_i) \quad (1)$$

スカラコアの論理量は，既存のプロセッサのゲート量より推定した．また，演算加速機構の論理量は，本プロセッサチップの上位論理設計からゲート量を求めた．論理部分の単位論理量あたりの電力単価は，既存のプロセスルールから 65 nm プロセスルール時のゲートスイッチ電力を推定し，長距離配線については配線長，配線本数から容量を想定して算出した．論理部分の活性化率は，対象論理の使用中は，各サイクルでスイッチの値が等しい確率で 4 通りに（0 から 0，0 から 1，1 から 0，1 から 1）移行するとして 50%とし，対象論理部分を使用する命令が発行されると活性化率が 50%になるとした．レジスタファイルについては，1 ポートアクセス時の電力を想定し，アクセスポート数に比例して活性化率が変化するとした．たとえば，ストア命令のようにリードポートを 1 つしかアクセスしない場合の活性化率は 100%，3 項演算命令のように 3 リードポートと 1 ライトポートをアクセスする場合の活性化率は 400%である．

プロセスルール 65 nm の SOTB（Silicon on Thin BOX）デバイスを用いることを想定し，プロセッサコアの電源電圧を 0.9 V，周波数は 2 GHz とした．また I/O である DDR3 インタフェースの電圧は 1.5 V とした．

表 1 に，消費電力に占める割合の大きい演算加速機構の論理とレジスタファイルについて，上記の方法により求めた電力単価（相対値），論理量，活性化率を示す．

なお，本稿ではプロセッサチップで生じる消費電力をシミュレートし，主記憶自体の消費電力は含まず，0/1 のスイッチングで発生するアクティブ電力についてのみ評価を行う．リーク電流による消費電力については，SOTB デバイスを用いることにより従来型のプロセスと比較してリーク電流を約 1/10 に削減できると見込まれるが[13]，温度などの条件により著しく値が変化し電力値の導出が困難であるため，現時点では評価には含めないことと

表 1　電力単価，論理量および活性化率
Table 1　Units of power consumption, amounts of logic and activated ratios.

	電力単価（相対値）	論理量（SIMD）	論理量（Vector）	活性化率
演算加速機構 論理	3 (/kGate)	210 (kGate/FMA)	210 (kGate/FMA)	50%
演算加速機構 レジスタファイル	10 (/port access)	1 (/port access)	1 (/port access)	100%〜400%

する．

4. 予備評価

詳細な評価と考察を行う前に，予備評価を行った．まず，想定するハードウェアコストが妥当であるか検証するために，想定する構成でのチップ面積を算出した．また，既存のプロセッサの電力性能を明らかにし，次章以降での評価の基準とする．

4.1 チップ面積

まず，現在想定している構成についてチップ面積が妥当な値になるか予備評価を行った．評価ではプロセスルール 65 nm の SOTB デバイスを用いることを想定し，論理量，配線量などから面積を算出した．なお，コア部分の面積はスカラコア数と演算器数に比例すると仮定し，スカラコア数と演算器数から S-RAM 面積，オンチップメモリ/スカラコア間スイッチ面積の増減を算出した．なお，I/O およびメモリインタフェースについては変化しないと想定した．算出結果を**表 2** に示す．

表 2 より，想定するプロセッサのチップ面積はベクトル型 16 コアを除いて，ほぼ 400 mm^2 に収まっていることが分かる．近年のプロセッサでは，Intel デュアルコア 64 ビット Xeon MP プロセッサ (codename “Tulsa”) のチップ面積が 435 mm^2 [14]，AMD クアッドコア Opteron プロセッサ (codename “Barcelona”) が 283 mm^2 と発表されており[15]，チップ面積から見たハードウェアコストとして，本研究で想定するプロセッサはベクトル型 16 コアの構成を除いて，ほぼ妥当な規模であると考えられる．

限られた面積で性能を向上させるためにはコア数を増加させる手法と FMA 数を増加させる手法が考えられる．同じ総 FMA 数ならばコア数よりもコアあたり FMA 数を増加させる手法のほうがスカラコアに関する相対的なコストが低減するため，電力性能あたりの面積などの観点で考えるとコアあたり FMA 数を増加させる手法が有利となる．したがって本稿では，ベクトル型としては Vector-16FMA に主眼を置いた評価を行う．

表 2　プロセッサチップ面積見積結果
Table 2　Result of chip area estimation.

Accelerator type		Vector			SIMD		
Cores		4	8	16	4	8	16
FMAs/cores		16	16	16	4	4	4
SCM size/chip		8 MB	8 MB	8 MB	8 MB	8 MB	8 MB
Chip area (mm^2)	Core (Scalar core + Accel.)	112.0	224.0	448.0	40.0	80.0	160.0
	SCM (S-RAM, SW, etc.)	115.0	148.3	215.0	96.3	123.3	177.5
	I/O (I/O, memory bus)	34.0	34.0	34.0	34.0	34.0	34.0
	Total	261.0	406.3	697.0	170.3	237.3	371.5
Chip side length (mm)		16.2	20.2	26.4	13.0	15.4	19.3

表 3　既存プロセッサの電力性能シート[1),16),17)]
Table 3　Power performance sheet of existing processors [1),16),17)].

CPU	Speed (GHz)	TDP (W)	PeakPerformance (GFLOPS)	PeakPerformance/TDP (GFLOPS/W)
(a) Xeon X5365	3.0	120	48.0	0.400
(b) Opteron 8360 SE	2.5	105	40.0	0.380
(c) BlueGene/L node ASIC	0.7	12	5.6	0.467

4.2 既存プロセッサの電力性能

本稿では電力性能を指標として検討するアーキテクチャの評価を行う．その評価に当たって，基準とするために既存のプロセッサの電力性能を机上計算で求めた．本稿では 1 W あたりの性能を示す GFLOPS/W を電力性能の指標の 1 つとして使用することとする．

表 3 は Intel Xeon X5365[16)]，AMD Opteron 8360 SE[17)] および BlueGene/L[1)] のプロセッサノード ASIC について，各プロセッサの動作速度，熱設計電力（Thermal Design Power．以降 TDP），理論ピーク性能および理論ピーク性能時のワースト電力と考えられる TDP1 W あたりのピーク性能（GFLOPS/W）を示している．表 3 より，3 種類のプロセッサは約 0.4 GFLOPS/W 前後の電力性能であることが分かった．

なお予備実験において Pentium4 と Opteron148 の実測に基づく電力性能と，同じくデータシートより求めた電力あたり性能を QCD を用いて比較したところ，どちらのプロセッサにおいても実測による電力性能がデータシートより求めた理論性能を下回った．これは QCD はメモリバンド幅に依存することから，メインメモリバンド幅律速により実効性能が理論ピーク性能より低くなったことによる性能低下が，性能低下による消費電力低下の影響を上回ったためと考えられる．以降，これらの値を参考に想定するアーキテクチャの評価を

行う．

5. 評 価 結 果

5.1 演算加速機構の基本特性

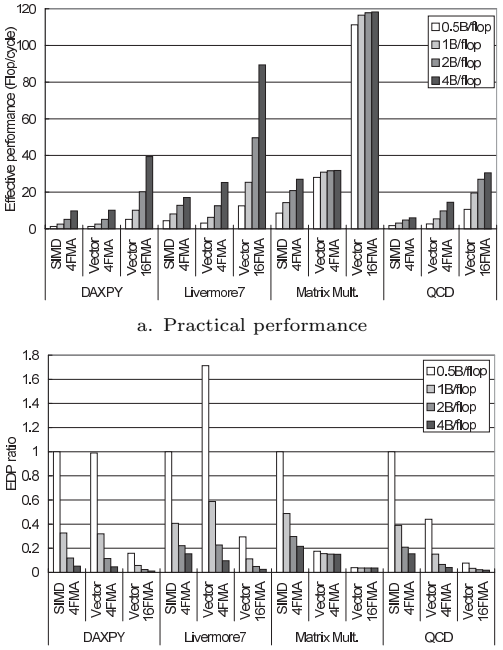
はじめに，各方式の基本性能を確認するためにすべてのデータがオンチップメモリにあると仮定して，演算加速機構とオンチップメモリ間のバスバンド幅をピーク演算性能あたりのデータ転送量（Byte/flop）を基準に変化させて評価を行った．演算コアは 4 コアと仮定し，DAXPY ループ，Livermore kernel 1，3，7 ではデータサイズ N = 200,000 を 10 回，行列積演算では行列サイズ 576×576 を 1 回，QCD ではデータサイズ $(\mathrm{NX}, \mathrm{NY}, \mathrm{NZ}, \mathrm{NT}) = (8, 2, 8, 16)$ の MULT ルーチンを 100 回実行した．これらのベンチマークのうち，DAXPY，Livermore7，行列積演算および QCD について結果を**図 5** に示す．実効性能は 1 サイクルあたりの演算数（flop/cycle）を，電力性能比は SIMD-4FMA 0.5 B/flop 時の性能を 1 として，各ベンチマークごとに正規化したものである．本稿では電力性能の指標としてエネルギー遅延積（Energy Delay Product：以降，EDP）を使用する．EDP は消費電力量に時間を乗じたものであり，電力あたりの性能を表す指標として広く用いられている．

5.1.1 実 効 性 能

各方式の基本構成である SIMD-4FMA と Vector-16FMA を比較すると，すべてのバンド幅において Vector-16FMA の実効および電力性能が SIMD-4FMA の性能を上回った．これは本評価ではピーク性能あたりのバスバンド幅で評価を行ったため，FMA 数を増やした場合でもバンド幅ボトルネックを生ずることなく性能が得られたためだと考えられる．そこで FMA 数によらない方式の特性を見極めるため，ここでは主に同じ FMA 数である SIMD-4FMA と Vector-4FMA の比較を行う．

まず，単純な計算ループである DAXPY では各バンド幅においてほぼ同じ性能を示した．この性能は，バンド幅より求められるベンチマークプログラムの理想性能にほとんど等しい値を示しており，これは Livermore1，3 においてもほぼ同様の結果となった．なお，Vector-16FMA については Vector-4FMA のほぼ 4 倍となる性能を示している．これらより基本的なベクトル処理においては演算加速機構の種類によらず，高い性能を発揮できることが分かる．

Livermore7 ではバンド幅が 2 B/flop 以下の場合に SIMD-4FMA の性能が Vector-4FMA の性能を上回っている．Livermore7 のループでは配列 $x[n]$ を求めるために配列 $u[n]$ から $u[n+6]$ までが使用される．$U[x] = \{u[x], u[x+1], u[x+2], u[x+3]\}$ とすると SIMD 型の



a. Practical performance

b. Power performance ratio

図 5　加速機構・オンチップメモリ間バンド幅による評価結果（4 コア）

Fig. 5　Evaluation result from effect of bandwidth between accelerators and on-chip memory (4 cores).

場合 4 要素を同時に演算できることからループ 4 回分がアンローリングされ演算では $U[n]$ から $U[n+6]$ が使用され，次のループでは $U[n+4]$ から $U[n+10]$ までが使用される．このとき $U[n+4]$ から $U[n+6]$ までは前のループで使用したデータを再利用できるためロードを削減できる．対するベクトル型ではベクトル長が 128 要素であるのでループイタレーション間でレジスタの再利用性がなくロードを削減できない．このためバンド幅ボトルネックとなる状況下では総データ転送量が少なくなる SIMD 型が有利になるものと考えられる．

行列積では，SIMD 型ではバンド幅の増加に従って徐々に性能が向上したが，ベクトル型では 0.5 B/flop バンド幅時においても，SIMD 型で最も性能の良い 4 B/flop 時の性能を上回った．これは両者のレジスタブロッキングサイズの差に起因するものと考えられる．SIMD

型ではレジスタが 4×128 要素のデータを保持するが，ベクトル型は 128×32 要素と，SIMD 型の 8 倍のデータをレジスタに保持できる．このため，ベクトル型は SIMD 型よりも行列積におけるレジスタブロッキングサイズを大きくとることが可能となり，レジスタにロードしたデータの再利用性が高まる．すなわち行列積では演算量あたりのロード数を少なくできることから，ベクトル型では低バンド幅でも高い性能を発揮できると考えられる．

QCD では，行列積よりもさらにベクトル型と SIMD 型の性能差が大きくなった．これは行列積で生じた問題に加えて，インデックスレジスタが不足したためスカラコアによるアドレス演算が多くなったことや，ベクトル版で行っている最内側ループでの 3×3 のアンローリングについて，SIMD 版では 20 サイクルのオンチップメモリへのアクセスレイテンシを隠蔽するためにはレジスタが不足するために 3×1 のアンローリングとせざるをえず，これによって再利用できるはずの計算結果を再計算する必要があるために，演算回数とデータロード回数が増加してしまったことが原因であると考えられる．

5.1.2 電 力 性 能

FMA 数が等しく実効性能が同じであっても電力性能では SIMD 型はベクトル型よりも命令発行数が多くなるために不利となるが，DAXPY における EDP を比較すると SIMD 型とベクトル型の電力性能はほぼ同じであった．すなわち命令発行数の差は電力性能に大きくは影響していないものと考えられる．Livermore7 では実効性能と同じく，EDP でも Vector-4FMA を下回っている．行列積では実効性能と同様，SIMD 型はベクトル型よりも EDP が大きい．また，各アーキテクチャでバンド幅増加による EDP 削減率は DAXPY が最も大きい．これは消費電力のうちクロックが占める割合が多いため，実効時間削減の効果を最も受けやすいからである．

ここで SIMD 型とベクトル型の消費電力の違いを見るため，行列積 4 B/flop 時の SIMD-4FMA，Vector-4FMA および Vector-16FMA の消費電力量内訳を**図 6** に示す．これは各ブロックの消費電力量を SIMD-4FMA の消費電力量全体を 1 として正規化したグラフである．消費電力の内訳は，スカラコア，演算加速機構ロジックのうち演算で消費された電力量，同じくロード/ストアで消費された電力量，同様の分類で演算加速機構のレジスタファイル，演算加速機構 = オンチップメモリ間配線，オンチップメモリ，およびクロックにより消費された電力量に分類されている．

SIMD-4FMA と Vector-4FMA を比較すると，Vector-4FMA で演算加速機構・オンチップメモリ間配線とオンチップメモリの消費電力量が大きく減っていることが分かる．これはベクトル型では SIMD 型と比較して多くの要素をレジスタ内に保持できるためレジスタブロッキングのブロックサイズが大きくなり，演算あたりのロードストア回数が少なくてすむためである．これにともない，演算加速機構の論理部分の消費電力やレジスタファイルの電力も減少している．また，スカラコアの消費電力量が減少しているのは，DAXPY では電力性能がほとんど変わらなかったことから，レジスタブロッキングサイズの差によりロードストア命令が少なくてすむことが主な要因であると考えられる．Vector-4FMA と Vector-16FMA の比較では Vector-16FMA が電力量を削減しているが，これはほぼクロック電力の削減，すなわち実行時間の短縮による効果であった．

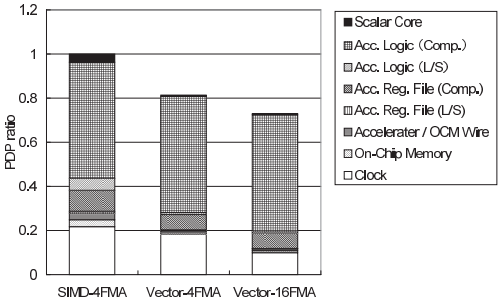


図 6　行列積演算 4 コア時における消費電力量内訳（4 B/flop 時）
Fig. 6　Power estimation detail on the matrix multiplication with 4 cores (a case of 4 B/flop).

なお，SIMD-4FMA と Vector-16FMA の比較では Vector-16FMA の方が電力性能が圧倒的に良いが，これは前述のとおり同 FMA 数でも有利であったベクトル型で，FMA 数がさらに増加したため実行時間が短縮され，EDP が大幅に減少したためだと考えられる．

5.2 主記憶バンド幅とスケーラビリティ

次にオンチップメモリ・主記憶間データ転送を含めたシミュレーションを行い，コア数のスケーラビリティに関して評価した．行列サイズ $N = 1{,}536 \times 1{,}536$ の倍精度行列積演算および N = 10,000,000 の DAXPY において，プロセッサコアの駆動周波数が 2.0 GHz のときに 1 ch 25.6 GB/s の DDR3 インタフェースを実装したと仮定して，各演算加速機構でチャネル数を 1 から 4 ch と変化させてシミュレーションを行った．なお演算加速機構とオンチップメモリ間のバンド幅は 4 B/flop とした．

5.2.1 電 力 性 能

まず，電力性能について評価を行った．実効性能は 1 サイクルあたりの演算数を，電力性能比は SIMD-4FMA 1 コア 25.6 GB/s 時の EDP を 1 として正規化したものである．評

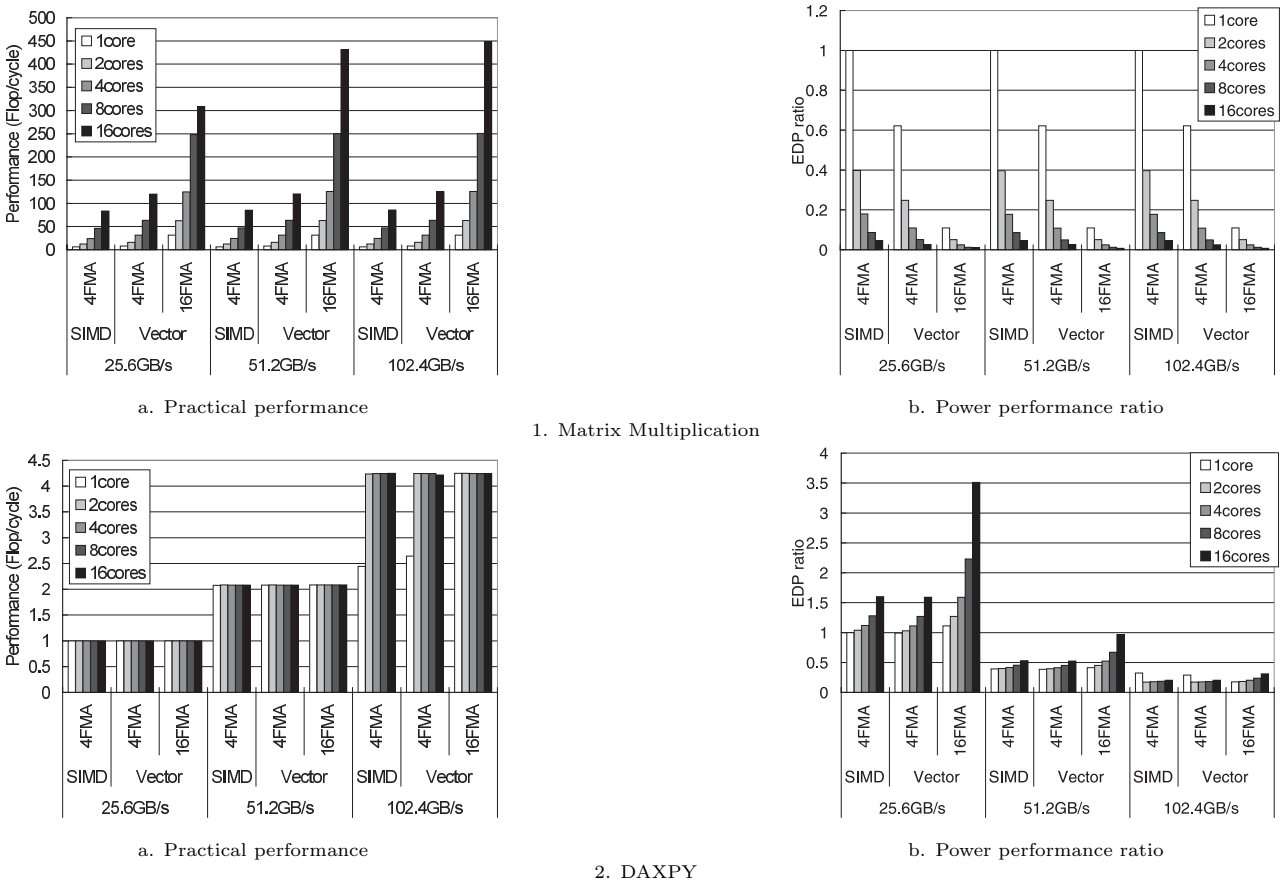


図 7　メモリバンド幅別スケーラビリティ評価結果
Fig. 7　Scalability result from effect of bandwidth between on-chip memory and main memory.

価結果を**図 7** に示す．行列積の実効性能を見ると Vector-16FMA を除き演算器律速となり，バンド幅増加による実効性能の向上にほぼ関係なく良好なスケーラビリティを示した．102.4 GB/s 時に 1 コアと比較して 16 コアとしたときの性能向上率は SIMD-4FMA が約 13.4 倍，Vector-4FMA が約 15.1 倍，Vector-16FMA が約 14.2 倍であった．このときの EDP は実効性能とほぼ反比例となっており，コア数の 1 から 16 への増加により EDP は同様にそれぞれ約 90.6%，93.6%，93.9% 削減された．なお，この評価においても FMA の多い Vector-16FMA の性能が最も良く，102.4 GB/s 時の Vector-16FMA で実効および電力性能が最良となった．

DAXPY では主記憶メモリバンド幅律速となり，ほぼすべての条件でバンド幅により実効性能が決定され，バスバンド幅 102.4 GB/s 時の 4FMA での評価において演算加速機構/オ

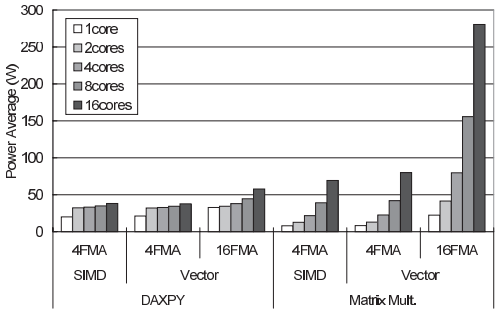


図 8　コア数別平均消費電力
Fig. 8　Average power consumption by the number of cores.

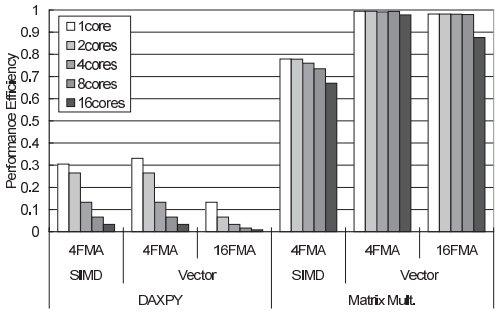


図 9　コア数別性能効率
Fig. 9　Performance efficiency by the number of cores.

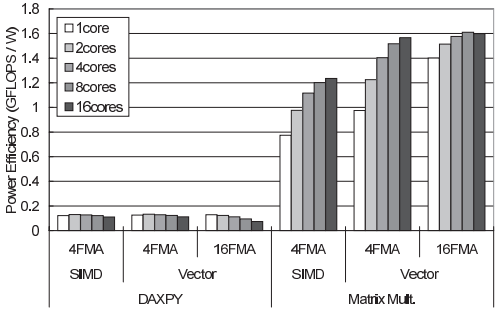


図 10　コア数別電力効率
Fig. 10　Power efficiency by the number of cores.

ンチップメモリ間バンド幅律速により実効性能が制限される以外，コア数増加による性能向上はほとんどなかった．このときの電力性能は FMA が多いほど不利であり，25.6 GB/s・16 コア時には SIMD-4FMA と比較して Vector-16FMA の EDP は約 2.19 倍にもなった．ただし，この比率はバンド幅増加により減少しており，102.4 GB/s 時には同様の比較で約 1.51 倍まで減少した．なお，電力性能が最良となったのは演算加速機構/オンチップメモリ間とオンチップメモリ/主記憶間のバンド幅が最も近くなった Vector-4FMA 2 コア時であった．

5.2.2　性能効率および電力効率

続いて，これまでの評価で最も電力性能の良かった演算加速機構 = オンチップメモリ間バンド幅 4 B/flop，オンチップメモリ/主記憶バンド幅 102.4 GB/s 時の消費電力，実効性能および電力効率について評価を行う．**図 8** に図 7 における平均消費電力を，**図 9** にこのときのプロセッサのピーク性能あたりの実効性能（以降，性能効率）を示す．図 8 より，多くの条件において平均消費電力は 100 W を下回ることが分かる．対して Vector-16FMA では 8 コア時に約 155.7 W，16 コア時には約 280.5 W となり 100 W を大幅に超過している．この条件は近年のプロセッサの TDP を大きく上回るものであり[16),17)]，ハードウェアコスト的に問題があると考えられる．これらの条件のケースを除けば，性能効率 66%以上となっているときでも平均消費電力は 100 W 以下となっているので，おおむね問題ないと考えられる．

次に同様の条件における消費電力あたりの実効性能（以降，電力効率）を求めた．**図 10** に電力効率を示す．図 10 を見ると，バンド幅律速となり性能効率の低い DAXPY では約 0.1 GFLOPS/W となっているが，性能効率の高かった行列積演算では SIMD 型の 1，2 コアおよび Vector-4FMA の 1 コアを除いて 1.0 GFLOPS/W 以上の電力効率となっており，表 3 で示した従来のプロセッサの性能を大きく上回っている．消費電力 100 W 以下では Vector-16FMA 8 コアのときに電力効率が最大となり，そのときの値は約 1.58 GFLOPS/W であった．

6. 考　　察

6.1　SIMD 型およびベクトル型による特性差

今回の評価で用いた典型的なベクトル処理のうち DAXPY や Livermore1，3 では FMA

数が同じならばSIMD型とベクトル型で実効性能と電力性能に大きな差はなかったが，行列積演算とQCDではベクトル型がSIMD型を上回る結果となった．この要因としては以下のような理由があげられる．

(1) レジスタにロードしたデータを再利用する際に，レジスタ容量の違いからレジスタブロッキングサイズがベクトル型はSIMD型よりも8倍大きくなり，結果として演算あたりのロード/ストア回数，すなわち必要となる単位時間あたりデータ転送量（B/Flop）が小さくなる．

(2) 1レジスタあたりの要素数について，ベクトル型はSIMD型の32倍あるため，たとえばデータ転送命令でスカラコアでのアドレス演算が必要となったときに最大32倍のスカラ命令が必要となるなどスカラ命令の発行数が多くなり，拡張命令の発行間隔が短くスカラコアに命令を発行する余裕の少ないSIMD型ではこのスカラ命令の多さがパイプラインストールに直結する．

(3) FMAあたりのレジスタ容量はベクトル型のほうが多いために，ベクトル型用プログラムはSIMD型用プログラムと比較して命令発行間隔が長くなり，各種レイテンシ隠蔽のための所要レジスタ数が少なくて済む．対するSIMD型ではより多くのレジスタが必要となる．本研究でのSIMD型においてレジスタ数128本という想定では，DAXPYのような単純な演算では問題とならなかったが，QCDでは特にレジスタ数が不足し，演算加速機構/オンチップメモリ間の20サイクルレイテンシを完全に隠蔽することができなかった．

このように，主にレジスタ容量の違いによりベクトル型はSIMD型よりも性能的に有利となることが多い．QCDにおいては，(1)〜(3)の問題のほかにも，インデックスレジスタを用いた相対値指定によるロード命令について，相対値が指定可能な値を超えることから相対値を用いたアドレス指定ができず，スカラコアですべてのアドレスを演算する必要があったことや，そもそもインデックスレジスタが不足することによって，アドレスをスカラコアに保持せざるをえなかったこともSIMD型の性能がベクトル型に及ばなかった原因となった．また，ベクトル型はレジスタ要素間演算などをサポートしないという制限をつけることでFMAを多く持つが，多くの評価ではこのFMAの多さがSIMD型と比較して実効性能および電力性能が高くなる大きな要因となった．以上のような理由により，本研究の評価では多くの評価で16FMAおよび4FMAのベクトル型の性能が4FMAのSIMD型の性能を上回った．

しかしながら，Livermore7のようにレジスタにロードしたデータを再利用できる場合や，今回のベンチマークには含まれなかったが複素数演算などでレジスタ内要素交換が生かせるような状況下であればSIMD型演算加速機構の利点が現れると考えられる．なお，この点を評価することを目的として本稿ではQCDによる評価を試みたが，SIMD型の性能がベクトル型の性能を上回ることはなかった．これはQCDの場合はSIMD型においてレジスタ内要素交換による演算と交換を必要としない演算をともに多用する必要があり，そのような実装ではオーバヘッドが非常に大きくなるため，SIMD型においてもベクトル型と同様に複素数を分離してデータを確保してベクトル計算にする必要があったことが主な要因であると考えられる．

本稿での評価では演算加速機構を駆動するためのSIMD命令およびベクトル命令は，コンパイラを開発できていないためにハンドコンパイルによる実装であったため，限られたアプリケーションによる評価で，限られた最適化しか行えなかった．しかし，本研究の評価結果から，ベクトル化が容易であるプログラムにおいてはベクトル型が高い性能を得られると考えられる．逆にベクトル化の行いにくいようなアプリケーションではSIMD型が有利になると予想されるが，本稿では十分な検証が行えなかった．また，本稿の評価で用いた電力単価についてはベクトル型とSIMD型のFMAあたりの電力コストは異ならないという見積りに基づいており，ハードウェアコストを正しく反映できていない可能性もある．加えて，5.1節においてピーク演算量あたりバンド幅（B/flop）というパラメータ下で比較したため，ベクトル型とSIMD型ではオンチップメモリの絶対的なバンド幅も異なっていたが，電力コストを含めたハードウェアコストの見積りも不十分である．今後はこの点を含めて，SIMD型がベクトル型を上回る可能性に着目して検証する必要があると考えている．

6.2 消費電力

本稿では演算器律速となる行列積演算と，主記憶バンド幅律速となるDAXPYの2つの傾向を持つプログラムで評価を行ったが，演算器律速の場合でも電力性能はほぼ低下せず，主記憶バンド幅増強が電力性能の向上に有効であることが分かった．ただし，チップ上に搭載可能なトランジスタ数や外部I/Oピン数には物理的な制約が生じ，DDR3のチャネル数はピン数の制限（1 ch 約250本ほど）から4チャネルが物理的限界であると考えられる．この制限により，今回の想定ではオンチップメモリ/主記憶バンド幅が最も大きくなる4チャネル構成が最も電力性能が良い構成であると考えられる．

5.2.2項の評価では，平均消費電力や，平均消費電力から求めた電力効率で評価を行った．プロセッサにおいては許容される電力が定義されているため，瞬間消費電力が重要な制限となるが，本評価ではその瞬間消費電力に関する評価が行えていない．この点については，

消費電力シミュレーションで必要とされる分解能を明らかにしたうえで，瞬間消費電力による評価を行うことでより確かな結果が得られるものと考えられる．

6.3　キャッシュ機構の想定

本研究での評価はオンチップメモリに対象を絞り，キャッシュとの比較は行っていない．オンチップメモリの有効性は従来研究で明らかであり[18),19)]，演算加速機構と組み合わせた場合でも従来研究と同様にデータ転送量の削減とともに，実行時間短縮の 2 つの効果により電力性能の向上が見込まれる．一方，演算加速機構とキャッシュの組合せを考えた場合，データ転送量の増大によるデメリットのほかに，特にベクトル型において 1 ロード/ストア命令で複数のキャッシュラインにアクセスするため，1 アクセスごとにキャッシュミスが頻発する可能性がある．またベクトルのリストアクセスが行われた際に同一のキャッシュウェイへの連続アクセスによるキャッシュラインコンフリクトが生じた場合などを考慮しなければならず，実現は難しい．実現可能としても，必要ハードウェア量の増加とそれにともなうアクセスレイテンシの増大が懸念される．加えて，キャッシュ機構にはプリフェッチが必須であると思われるが，その実現方法も同様の理由で実装が難しい．この点において，複雑な機構の必要ないオンチップメモリは演算加速機構を親和性が高く，アクセス時の動作も必ず同じで一定のレイテンシとなることから従来のキャッシュと比較して優位であると考えられる．

7. ま と め

本稿では電力性能の向上に有効であるオンチップメモリプロセッサアーキテクチャSCIMAに多数の演算器を備えた演算加速機構を導入することを検討し，SIMD 型演算加速機構とベクトル型演算加速機構について実効性能と電力性能を指標に評価を行った．

シミュレーションによる評価の結果，すべてのデータがオンチップメモリにあると仮定した場合，同じ FMA 数であれば DAXPY や Livermore1 のような典型的なベクトル処理では実効性能と電力性能において SIMD 型演算加速機構とベクトル型演算加速機構はほぼ同等の性能であったが，行列積演算および QCD ではレジスタ内の要素数の差などによりベクトル型は SIMD 型より実効および電力性能が勝る結果となった．Livermore7 ではループイタレーション間のレジスタ再利用により演算加速機構/オンチップメモリ間のバンド幅が 2 B/flop 以下の場合に SIMD 型の性能が同 FMA 数のベクトル型の性能を上回った．しかし，Vector-16FMA はすべての条件で SIMD-4FMA の性能を上回った．

オンチップメモリ/主記憶間のデータ転送を含めた評価では，演算器律速である行列積演算では良好なスケーラビリティを示し，特に FMA 数の多いベクトル型が良好な電力性能を示した．主記憶バンド幅律速となる DAXPY では FMA 数が多いほど電力性能は低下したが，主記憶バンド幅の増加により性能の低下率は緩和した．また，電力効率は Vector-16FMA のときに最大となり，8 コア時の電力効率は約 1.58 GFLOPS/W であった．

今後は実用に近い多様なアプリケーションでの評価，およびオンチップメモリの代替としてキャッシュが使用された場合の評価を行い，演算加速機構の構成の評価をするとともに演算加速機構を持つオンチップメモリプロセッサの優位性を明らかにしたい．

謝辞　本研究の一部は文部科学省「次世代 IT 基盤構築のための研究開発」プロジェクト「低電力高速デバイス・回路技術・理論方式の研究開発」による．

参 考 文 献

1) Adiga, N., et al.: An overview of the Blue Gene/L supercomputer, *Proc. SC2002*, pp.1–22 (2002).
2) 中村　宏，近藤正章，大河原英喜，朴　泰祐：ハイパフォーマンスコンピューティング向けアーキテクチャSCIMA，情報処理学会論文誌：ハイパフォーマンスコンピューティング，Vol.41, No.SIG 5(HPS 1), pp.15–27 (2000).
3) Kondo, M., Fujita, M. and Nakamura, H.: Software-controlled on-chip memory for high-performance and low-power computing, *ACM SIGARCH Computer Architecture News*, Vol.30, No.3, pp.7–8 (2002).
4) Hofstee, H.P.: Power Efficient Processor Architecture and The Cell Processor, *Proc. HPCA'05*, pp.258–262 (2005).
5) AMD: Torrenza: Leading the Industry in Open Collaboration. http://enterprise.amd.com/us-en/AMD-Business/Technology-Home/Torrenza.aspx
6) 高橋睦史，佐藤三久，高橋大介，朴　泰祐，宇川　彰，中村　宏，青木秀貴，澤本英雄，助川直伸：演算加速機構を持つオンチップメモリプロセッサの検討と電力性能評価，情報処理学会研究報告，2008-ARC-174, pp.37–42 (2007).
7) Lemuet, C., Sampson, J., Collard, J.-F. and Jouppi, N.P.: The potential energy efficiency of vector acceleration, *Proc. SC2006* (2006).
8) McMahon, F.H.: The Livermore Fortran Kernels: A Computer Test Of The Numerical Performance Range, Technical report, Lawrence Livermore National Laboratory (1986).
9) Aoki, S., Burkhalter, R., Kanaya, K., Yoshie, T., Boku, T., Nakamura, H. and Yamashita, Y.: Performance of lattice QCD programs on CP-PACS, *Parallel Computing*, Vol.25, pp.1243–1255 (1999).
10) Lam, M.S., Rothberg, E.E. and Wolf, M.E.: The cache performance and optimizations of Blocked Algorithms, *Proc. ASPLOS-IV*, pp.63–74 (1991).

11) van der Vorst, H.A.: BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.*, Vol.13, pp.631–644 (1992).
12) Aoki, S., Ishikawa, K.I., Iwasaki, Y., Kanaya, K., Kaneko, T., Kuramashi, Y., Tsutsui, N., Ukawa, A. and Yoshie, T.: Lattice QCD on Earth Simulator, *Nucl. Phys. B* (*Proc. Suppl.*), 129, 130, pp.859–861 (2004).
13) Shimizu, K. and Hiramoto, T.: Mobility Enhancement in Uniaxially Strained (110) Oriented Ultra-Thin Body Single- and Double-Gate MOSFETs with SOI Thickness of Less Than 4 nm, *Electron Devices Meeting, 2007, IEDM Technical Digest, IEEE International*, pp.715–718 (2007).
14) Rusu, S., Tam, S., Muljono, H., Ayers, D. and Chang, J.: A Dual-Core Multi-Threaded Xeon Processor with 16 MB L3 Cache, *Solid-State Circuits, 2006 IEEE International Conference Digest of Technical Papers*, pp.315–324 (2006).
15) Dorsey, J., Searles, S., Ciraula, M., Johnson, S., Bujanos, N., Wu, D., Braganza, M., Meyers, S., Fang, E. and Kumar, R.: An Integrated Quad-Core Opteron Processor, *Solid-State Circuits Conference, 2007, ISSCC 2007, Digest of Technical Papers, IEEE International*, pp.102–103 (2007).
16) Intel: Quad-Core Intel Xeon Processor 5300 Series Datasheet. http://download.intel.com/design/Xeon/datashts/31556903.pdf
17) AMD: Quad-Core AMD Opteron Processors Fast Facts. http://www.amd.com/us-en/assets/content_type/DownloadableAssets/FINAL_Fast_Facts_43410B.pdf
18) 高橋睦史，近藤正章，朴　泰祐，高橋大介，中村　宏，佐藤三久：HPC 向けオンチップメモリプロセッサアーキテクチャSCIMA の SMP 化の検討と性能評価，情報処理学会論文誌：コンピューティングシステム，Vol.44, No.SIG6(ACS 1), pp.76–86 (2003).
19) Takahashi, C., Kondo, M., Boku, T., Takahashi, D., Nakamura, H. and Sato, M.: SCIMA-SMP: On-chip memory processor architecture for SMP, *WMPI '04: Proc. 3rd workshop on Memory performance issues*, ACM, pp.121–128 (2004).

(平成 20 年 7 月 23 日受付)
(平成 20 年 11 月 17 日採録)

高橋　睦史（正会員）

昭和 55 年生．平成 14 年筑波大学第三学群情報学類卒業．平成 17 年より平成 19 年まで筑波大学産学官連携研究員．平成 20 年筑波大学大学院博士課程システム情報工学研究科修了．博士（工学）．同年（株）ルネサステクノロジ入社．組み込み向けプロセッサの設計開発に従事．

佐藤　三久（正会員）

昭和 34 年生．昭和 57 年東京大学理学部情報科学科卒業．昭和 61 年同大学院理学系研究科博士課程中退．同年新技術事業団後藤磁束量子情報プロジェクトに参加．平成 3 年通産省電子技術総合研究所入所．平成 8 年新情報処理開発機構並列分散システムパフォーマンス研究室室長．平成 13 年より筑波大学大学院システム情報工学研究科教授．平成 19 年より同大学計算科学研究センターセンター長．理学博士．並列処理アーキテクチャ，言語およびコンパイラ，計算機性能評価技術，グリッドコンピューティング等の研究に従事．IEEE，日本応用数理学会各会員．

高橋　大介（正会員）

昭和 45 年生．平成 3 年呉工業高等専門学校電気工学科卒業．平成 5 年豊橋技術科学大学工学部情報工学課程卒業．平成 7 年同大学大学院工学研究科情報工学専攻修士課程修了．平成 9 年東京大学大学院理学系研究科情報科学専攻博士課程中退．同年同大学大型計算機センター助手．平成 11 年同大学情報基盤センター助手．平成 12 年埼玉大学大学院理工学研究科助手．平成 13 年筑波大学電子・情報工学系講師．平成 16 年同大学大学院システム情報工学研究科講師．平成 18 年同助教授，平成 19 年同准教授．博士（理学）．並列数値計算アルゴリズムに関する研究に従事．平成 10 年度情報処理学会山下記念研究賞，平成 10 年度，平成 15 年度情報処理学会論文賞各受賞．日本応用数理学会，ACM，IEEE，SIAM 各会員．

朴　　泰祐（正会員）

昭和 35 年生．昭和 59 年慶應義塾大学工学部電気工学科卒業．平成 2 年同大学大学院理工学研究科電気工学専攻後期博士課程修了．工学博士．昭和 63 年慶應義塾大学理工学部物理学科助手．平成 4 年筑波大学電子・情報工学系講師，平成 7 年同助教授，平成 16 年同大学大学院システム情報工学系助教授，平成 17 年同教授，現在に至る．超並列計算機アーキテクチャ，ハイパフォーマンスコンピューティング，クラスタコンピューティング，グリッドに関する研究に従事．平成 14 年度および平成 15 年度情報処理学会論文賞受賞．日本応用数理学会，IEEECS 各会員．

宇川　彰

昭和 24 年生．昭和 51 年東京大学大学院理学系研究科博士課程物理学専攻単位取得退学．昭和 52 年理学博士（東京大学）．昭和 56 年東京大学原子核研究所助教授．昭和 59 年筑波大学物理学系助教授，平成 2 年より同教授．平成 10 年より筑波大学計算物理学研究センター長．平成 16 年より同計算科学研究センター長．平成 18 年より筑波大学学長特別補佐．計算素粒子物理学，計算科学のための並列計算機の開発，計算科学全般の推進に従事．平成 6 年仁科記念賞．日本物理学会会員．

中村　宏（正会員）

昭和 60 年東京大学工学部電子工学科卒業．平成 2 年同大学大学院工学系研究科電気工学専攻博士課程修了．工学博士．同年筑波大学電子・情報工学系助手．同講師，同助教授，平成 8 年東京大学先端科学技術研究センター助教授，平成 20 年より東京大学大学院情報理工学系研究科准教授．この間，平成 8〜9 年カリフォルニア大学アーバイン校客員助教授．高性能・低消費電力プロセッサのアーキテクチャ，ハイパフォーマンスコンピューティング，ディペンダブルコンピューティング，ディジタルシステムの設計支援の研究に従事．情報処理学会より論文賞（平成 5 年度），山下記念研究賞（平成 6 年度），坂井記念特別賞（平成 13 年度）各受賞．IEICE，IEEE，ACM 各会員．

青木　秀貴（正会員）

昭和 47 年生．平成 9 年京都大学大学院工学研究科情報工学専攻修士課程修了．同年（株）日立製作所入社．スーパコンピュータの研究開発，プロセッサアーキテクチャの研究に従事．平成 19 年より 1 年間，カリフォルニア大学バークレー校 Visiting Industrial Fellow.

澤本　英雄

昭和 28 年生．昭和 53 年京都大学大学院工学研究科電子工学専攻修士課程修了．同年（株）日立製作所入社．汎用コンピュータ，RISC プロセッサ，スーパコンピュータの開発に従事．平成 18 年 4 月より国家プロジェクトの次世代スーパコンピュータ開発に従事．平成 19 年 10 月より省電力プロジェクト推進室室長．サーバおよびデータセンタの省電力化プロジェクトに従事.

助川　直伸

昭和 42 年生．平成 4 年東京大学大学院工学系研究科電子工学専攻修士課程修了．同年（株）日立製作所入社．スーパコンピュータ，並列計算機向け論理方式の研究および開発に従事.