

## 低消費電力プロセッサによるクラスタの検討

堀田 義彦<sup>†</sup> 佐藤 三久<sup>††</sup> 朴 泰祐<sup>††</sup>  
高橋 大介<sup>††</sup> 高橋 睦史<sup>†</sup> 中村 宏<sup>†††</sup>

近年、マイクロプロセッサの消費電力の上昇は著しいものがあり、クラスタなどにおいて高密度な実装が困難になっている。一方、従来、PDA やノート PC に使われていた低消費電力プロセッサは、高性能化しつつあり、これらのプロセッサを用いることによる高密度なクラスタを構築できることが期待できる。低消費電力のプロセッサによるクラスタを検討するにあたり、現在の主流なマイクロプロセッサや低消費電力 CPU の消費電力特性を評価した。ホール素子を用いた電力測定環境を構築し、いくつかのベンチマークプログラムにより実際に Pentium4, XScale, Crusoe の消費電力を計測した。キャッシュを有効利用することにより、総消費電力を低く抑える方法であることがわかった。また、実際に iPAQ による StrongARM と Crusoe を用いたクラスタを構築し、消費電力と性能の測定を行ない評価をした。

### Low Power Cluster using Low Power CPU

YOSHIHIKO HOTTA,<sup>†</sup> MITSUHIISA SATO,<sup>††</sup> TAISUKE BOKU,<sup>††</sup>  
DAISUKE TAKAHASHI,<sup>††</sup> CHIKAFUMI TAKAHASHI<sup>†</sup>  
and HIROSHI NAKAMURA <sup>†††</sup>

Recently, power consumption of high performance microprocessor is rapidly increasing, and it makes high-density packaging difficult when building clusters. On the other hands, low-power processor which is usually used in PDA and Note PC is getting faster. We expect "the low-power processor to be used for high-density and high-performance clusters."

In this paper, we examine the characteristics of power consumption of low-power processors. To measure the power consumption, we have built the environment to measure power by Hall device. We found that cache can aware program reduce the total power consumption. We built low-power clusters using iPAQ and Crusoe, and evaluated power consumption and performance for low-power cluster.

#### 1. はじめに

近年、マイクロプロセッサの性能は著しく向上しているが、高性能化とともに実装面において消費電力を低く抑えるニーズも増している。HPC の分野でプラットフォームとして主流になりつつあるクラスタなどの並列システムにおいて電力の急増による発熱量の増加のためにプロセッサは大きな冷却装置が必要になっている。これによりクラスタの実装密度を上げることができない場合も増えている。

一方で、従来 PDA やノート PC 向けの低消費電力

プロセッサの性能が向上してきている<sup>1)</sup>。本稿では、このような背景を踏まえ、低消費電力プロセッサを用いたクラスタについて、検討する。低消費電力プロセッサを用いることにより、ファンや冷却に対する制限が少なくなり実装密度の向上が期待できる。

このような問題に対処すべく、LosAlamos 研究所の Feng らによるグループは低消費電力スーパーコンピュータ「Green Destiny」を開発し評価を行なっている<sup>2)3)</sup>。コストや体積あたりの性能、電力あたりの性能が既存のスーパーコンピュータよりも高くできることが報告されている。

低消費電力プロセッサによるクラスタの検討にあたり、我々は、ホールデバイスを用いた電力計測システム環境を構築し、数種の実際の CPU を様々なベンチマークを実行し電力を計測した。さらに、実際にそれらのプロセッサを用いたクラスタを試作し、電力、性能について評価を行なった。

2 章ではプロセッサの電力の測定と電力特性について述べる。3 章ではプログラムの最適化による消費電

<sup>†</sup> 筑波大学大学院システム情報工学研究科  
Graduate School of Information and Sciences Engineering,  
University of Tsukuba

<sup>††</sup> 筑波大学 電子・情報工学系  
Institute of Information Science and Electronics, University  
of Tsukuba

<sup>†††</sup> 東京大学先端科学技術研究センター  
Research Center for Advanced Science and Technology,  
The University of Tokyo

力の削減について述べる。4章では低消費電力プロセッサを用いたクラスタの性能と消費電力について述べる。5章では結果からの考察を述べる。最後に6章で結論と今後の展望について述べる。

## 2. マイクロプロセッサの消費電力の測定

### 2.1 電力測定環境

クラスタに用いるプロセッサを調査・評価するに当たって、現在ある各 CPU の電力消費特性を調べるために (株) シナジェティック社製 CT-30000 を用いた。この装置はホール素子、接続 BOX, A/D コンバータから構成されており、ATX 電源の各電圧の電線に流れる電流をダイナミックに測定が可能である。ホール素子の間に電線を通すだけで良く、取扱が容易であるのもこの装置の特徴である。この装置を用いてプログラム実行時の消費電力を測定した。また、瞬間的な電力だけではなく、総消費電力量も重要になるので瞬間の消費電力にサンプリング間隔との積をとり、積分を行なった。この総消費電力量の単位を [W・S] とする。これ以降、表で電力の単位をこの総消費電力の単位とする。

ATX 電源には+12V,-12V,+5V,-5V,+3.3V の各電圧があるが、測定の結果+12V,+5V,+3.3V 以外はほとんど電流が流れていないので測定の対象から外した。また Pentium4 では CPU への電力供給に+12V を変圧したものが使われている。他の CPU では+5V を変圧したものが使われている。

### 2.2 対象とした CPU

表 1 に測定対象のシステムの仕様を示す。測定を行なう CPU は、Pentium4, XScale, Crusoe の 3 つである。

XScale<sup>4)</sup> は StrongARM シリーズの上位互換 CPU で Intel が 2000 年に発表した低消費電力 CPU である。主な用途は PDA などの携帯デバイスである。現在は FPU (浮動小数点演算装置) が搭載されていないが、将来的には搭載されることも予定されている。

Transmeta 社の Crusoe TM-5800<sup>5)</sup> は VLIW アーキテクチャを採用している CPU である。この CPU は CMS (Code Morphing Software) により x86 命令を独自の VLIW 命令 (128bit) に変換し実行することによって x86 命令を使用する CPU との互換性を備えている。また、DVS (Dynamic Voltage Scaling)<sup>6)</sup> と呼ばれる動的に動作電圧を変化させる機能により、必要な負荷に応じてクロック周波数を変更し低消費電力化を実現している。

Pentium4 は現在の主流な CPU と低消費電力 CPU との比較のために測定を行った。表 2 に各システムの無負荷時の消費電力を示す。

### 2.3 ベンチマークプログラム

前述のシステムにおいていくつかのベンチマークを実行し、性能と電力消費の振舞を動的に確認した。測定に以下のベンチマークを用いた。

システム名	P4	XS	Crusoe
マザーボード	GA-81RX	NPWR	EB5
CPU	Pentium4	XScale	TM-5800
Clock	1.80GHz	733MHz	933MHz
L1 cache	8KB	64KB	64KB
L2 cache	512KB	none	512KB
Memory	512MB	256MB	256MB
Compiler	gcc3.2	gcc2.95.2	gcc3.2

表 1 測定 PC の仕様

	12V	5V	3.3V
P4	15W(CPU)	1W	3W
XS	0.5W	9W(CPU)	none
Crusoe	1W	6W(CPU)	0.3W

表 2 無負荷時の各システムの消費電力

- datascan
- dhrystone
- NPB (NAS Parallel Benchmark) IS
- matrix multiply ブロッキング版
- LU 分解ブロッキング版

#### 2.3.1 datascan

datascan はアクセスする領域のサイズを変化させ、キャッシュヒットしているかどうかを確認するプログラムである。このプログラムを各システム上で実行し、キャッシュヒット時とミスヒット時での性能の違いと電力差について確認した。図 1 から図 3 にそれぞれのプロセッサでのこのプログラムでの動的な電力の変化を示す。

Pentium4 の場合 (図 1)、プログラムを実行すると急激に消費電力が上昇する。消費電力が時間の経過とともに徐々に減少している。アクセス領域の増加とともにない、L1 キャッシュ、L2 キャッシュ、外部メモリと徐々に記憶階層から外れ、消費電力が減少するのがわかる。また、キャッシュに当たる時と当たらない時での電力差は数 W と少ない。キャッシュヒットしない場合は実行時間が大きくなるため、総消費電力はキャッシュにヒットし続ける方が電力は少なくて済む。

XScale の場合 (図 2)、P4 とは異なり、プログラムの開始直後のキャッシュにヒットしている部分での電力の方が、メモリアクセスしている部分より少ない。これは、XScale の消費電力が非常に小さく、メモリの消費電力の方が大きいためである。よって、キャッシュにヒットしない場合は、実行時間も増大するために総消費電力は増大する。

Crusoe の場合、プログラム実行時に消費電力は無負荷時と比べて数 W 上昇する。キャッシュヒット時とミスヒット時で差は見られない。CPU の消費電力の減少とメモリアクセスによる消費電力の上昇がほぼ同じであり、5V の line を共有しているため、CPU の電力の変化を観測できない。よって、実行時間が短く

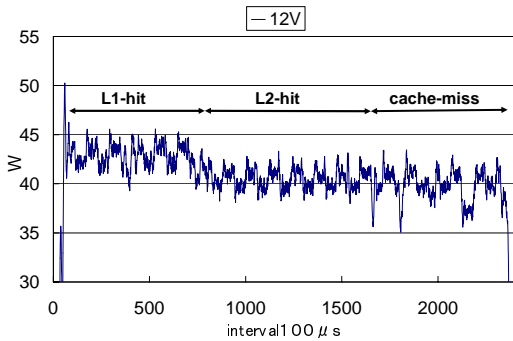


図 1 P4 12V datascan 実行時電力変化

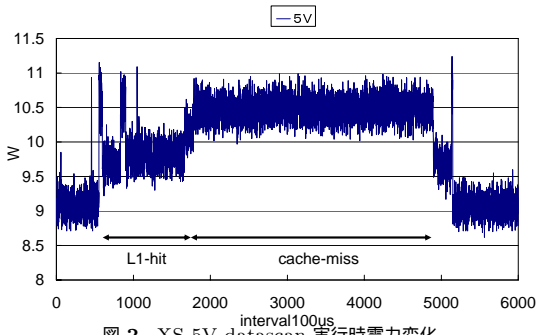


図 2 XS 5V datascan 実行時電力変化

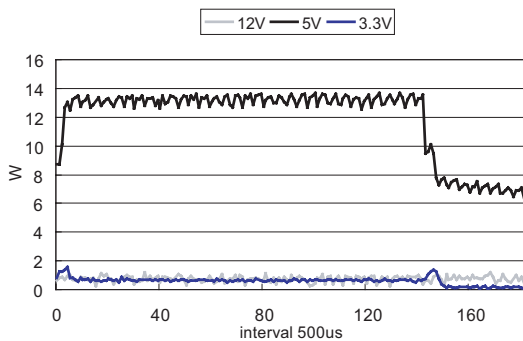


図 3 Crusoe datascan 実行時電力変化

なるキャッシュを有効利用することが省電力になる。  
2.3.2 dhrystone

dhrystone とは整数演算や文字列コピーを繰返し実行するベンチマークである。プログラム全体がキャッシュに収まる場合の各システムでのパフォーマンスと電力を比較した。図 4 に実行時間と電力のグラフを示す。P4 は高速に実行が可能である一方で総消費電力が最も多くなっている。XS はこのプログラムにおいて P4 との性能差は大きい。しかし、このようにキャッシュにヒットし続けるプログラムの場合 XS は低消費電力で実行できる。Crusoe では実行時間は P4 に及ばないが、総消費電力は約 4 分の 1 となっており電力あたりの性能では最も高い。

### 2.3.3 NPB

実用的なアプリケーションで性能と消費電力を比較

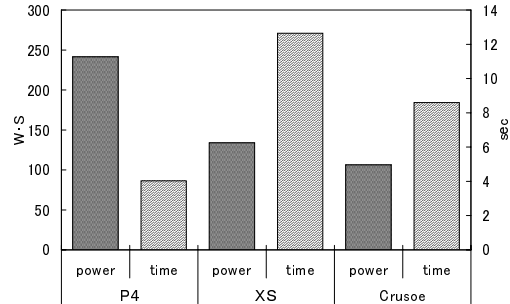


図 4 dhrystone 測定結果

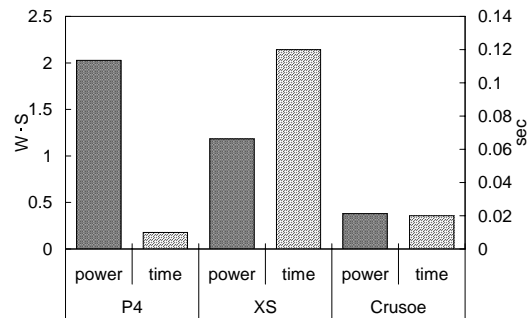


図 5 IS CLASS=S 測定結果

するために逐次版の NPB の IS で評価を行なった。IS はソートの実行部分のみで電力の評価を行なった。表 3 に IS の各クラスでの総実行時間とソート実行時間の結果を示す。XS はソートの実行時間だけならば P4 の 10 倍程度でクロック周波数の差を考慮すると妥当な結果であるが、総実行時間は大きくなる。これは、IS のベンチマークは前半部分でソートに必要なデータを生成しているが、XS はデータの生成などのメモリアクセスが頻繁に起こるような処理の性能が低いためにこのような差が生じている。

一方、Crusoe では高性能かつ低消費電力でプログラムを実行できているのが確認できる。図 5 に IS の CLASS=S でのソートの実行時間と総消費電力を示す。このグラフからも Crusoe が性能と電力において優位となっている。

CLASS	S	W
P4	0.01 (0.077)	0.21 (1.004)
XS	0.12 (9.487)	7.51 (158.15)
Crusoe	0.02 (0.100)	0.60 (2.150)

表 3 逐次版 IS のソート実行時間 (括弧内は全体実行時間 [sec])

## 2.4 各 CPU の電力消費特性

以下に各 CPU の電力特性をまとめる。

—Pentium4 は高性能であるが瞬間的な電力は非常に高いものであった。また、プログラムがキャッシュヒットしている時としていない時で、数 W の違いが

見られた。Pentium4の電力特性としては、より高速にプログラムを実行することで全体の総消費電力が少なくなる傾向がある。瞬間の電力は急激に上昇するが、総消費電力は少なくなる。

—XScale では、キャッシュヒット時の消費電力の方が少なミスヒット時より少なくなっているこれは、CPUの消費電力の上下よりもメモリの消費電力の上下の方がはるかに大きくなるのが原因である。また、ミスヒット時にヒット時と比べて急激に性能が低下することも総消費電力が増大する原因となっている。キャッシュが有効利用され、整数演算の多いプログラムの実行においてのみ、低消費電力でプログラムを実行できる。

—Crusoe では実行時に消費電力が数 W しか上昇しない。CPUの消費電力の減少とメモリアクセスによる消費電力の上昇がほぼ同じであり、ATXの5Vのlineを共有しているため電力の大きな変化は観測できない。最大で12Wぐらゐと消費電力も少なく、電力あたりの性能も出ている。電力あたりの性能が非常に高く、冷却の必要がないほど瞬間の消費電力が低い。

### 3. プログラム最適化による低消費電力

これまで述べたように、低消費電力CPUではなるべくキャッシュを有効利用することが、消費電力削減につながるということがわかった。そこでキャッシュブロッキングを行なうことによってキャッシュを有効に利用する時の電力の変化を確認した。対象としたプログラムは、単純な行列積とLU分解のプログラムである。

#### 3.1 matrix multiply

このプログラムでの測定はXSのみ扱うデータをintegerで行なった。ここまでの測定の中で低消費電力CPUの利点を生かすためには実行時間を短くする必要があるのでわかっている。低消費電力CPUの場合キャッシュから外れた場合急激に性能が低下するので性能低下を防ぐためにキャッシュを有効利用する手法としてキャッシュブロッキングを適用した。これは、空間的局所性を向上させることによりキャッシュを利用する割合を多くし、性能を向上させる手法である。このブロッキングのサイズを変化させ、実行時間と消費電力の変化を確認した。行列のサイズは1024である。

図6にCrusoeにおける各ブロッキングサイズでの実行時間と総消費電力を示す。表4に各CPUでのブロッキングを行わない場合と行なう場合の実行時間を示す。すべてのCPUで効果があるが、特にXSとCrusoeで効果が大きい。これは、ブロッキングを行なう場合、計算を実行している間、ほとんどキャッシュにヒットしているのが原因である。一方でP4では実行時間が減少はするが約半分と減少の割合が少ない。これは、Crusoeなどではキャッシュミスによる性能低

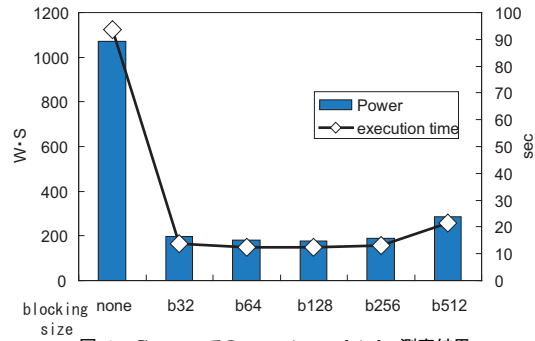


図6 Crusoeでのmatrix multiply測定結果

bsize	none	32	64	128	256	512
P4	16.5	8.8	8.3	7.7	11.7	14.3
XS	966.0	34.0	31.4	54.7	64.0	65.3
Crusoe	93.7	13.7	12.4	12.3	13.0	21.3

表4 ブロッキングサイズ毎の実行時間 [sec]

下が大きいことを示している。この結果から、低消費電力CPUにとっては、低消費電力化には、実行時間短縮が効果的である。キャッシュ有効利用は、低消費電力CPUにとっては実行時間短縮に効果的である。

#### 3.2 LU分解

ブロッキングされたLU分解のプログラム<sup>7)</sup>を用いて、電力と性能を測定した。このプログラムはP4とCrusoeで測定を行なった。表5に性能と電力の結果を示す。ブロッキングを行なうことにより約3倍性能が向上する。性能に反比例するように消費電力はブロッキング行なうと3分の1に削減できている。前述の行列積と同じように、キャッシュのヒット率を上げることにより実行時間を減らすことが低消費電力化への有効なアプローチであることが確認できた。

	w/o blocking	with blocking
Crusoe	149.9(56.2)	53.6(174.5)
P4	270.3(138.4)	92.2(442.8)

表5 LU分解における消費電力(括弧内は性能 [MFlops])

### 4. 低消費電力CPUを用いたクラスタの構築

低消費電力CPUを用いた2種類のクラスタを実際に構築し、評価した。表6にそれぞれの仕様を示す。

#### 4.1 Crusoeクラスタ

NPBのIS, LUで性能を計測した。表7にISでの2node, 4nodeにおける性能と消費電力を示す。この結果と表5や図3とを比較すると、クラスタにしても性能が低下している。また、台数が増えても性能に変化がなく、結果として総消費電力も大きくなっている。低消費電力化のためには、このような通信が多いアプリケーションでは逐次での実行が良いと思われる。表8は、LUでのCrusoeクラスタの性能と消費電力を2node, 4nodeの場合とP4との比較をまとめたもので

クラスタ名	Crusoe クラスタ	iPAQ クラスタ
CPU	TM-5800	Strong ARM-1110
Clock	933MHz	206MHz
Memory	256MB	64MB
OS	RedHat8.0	Familiar v0.6.3
kernel	2.4.18	2.4.18
Network	100BASE	10BASE
Nodes	4	8

表 6 構築した cluster の仕様

ある。

この結果から、局所性の高いプログラムでは、大幅な性能向上がわずかな消費電力の増加によって実現できているのがわかる。

#### 4.2 ブロッキングによる低消費電力化

キャッシュの有効利用が低消費電力クラスタにおいてどれくらい影響があるのかを確認するために matrix multiply を並列化し、サイズが 1024 の行列にブロッキングを行ない、キャッシュの有効利用が低消費電力クラスタにおいてどれくらい影響があるのかを調べた。

表 9 に各ノード数でのブロッキングサイズ毎の実行時間を示す。single の結果は MPI を使用していない完全な逐次版での結果である。図 7 にブロッキングを施さなかったプログラムでの総消費電力を示す。図 8 にブロッキングサイズを 128 にしたプログラムでの総消費電力を示す。これらの結果から、並列化によってわずかな総消費電力の上昇で、大幅な性能向上が実現できていることが確認できる。このことからクラスタにおいてもブロッキングは有効な手法であり、大幅な低消費電力化には不可欠なものとなる。

	CLASS S	CLASS W	CLASS A
2node	41.2(0.18)	95.4(2.60)	547.1(19.6)
4node	131.0(0.14)	240.9(2.80)	960.6(21.01)

表 7 Crusoe クラスタ NPB IS 電力 (括弧内は実行時間 [sec])

	CLASS S	CLASS W	CLASS A
2node	52.8(78.1)	2510.8(179.8)	17674.7(177.3)
4node	159.1(92.6)	2764.8(292.5)	17069.5(338.4)
P4	14.2(420.5)	3015.5(308.3)	19532.4(286.4)

表 8 Crusoe クラスタ NPB LU 電力 (括弧内は性能 [Mop/s])

bsize	none	32	64	128	256	512
single	93.7	13.7	12.4	12.3	13.0	21.3
2node	108.6	6.1	5.4	5.4	6.0	9.8
3node	109.1	6.1	5.4	5.4	6.4	9.9
4node	54.6	3.4	3.0	3.0	3.6	5.2

表 9 並列版ブロッキング行列積実行時間 [sec]

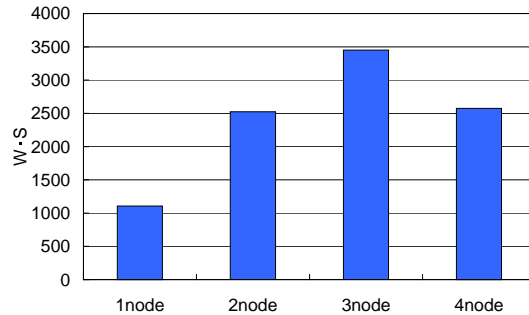


図 7 blocking なしでの Crusoe クラスタ総消費電力

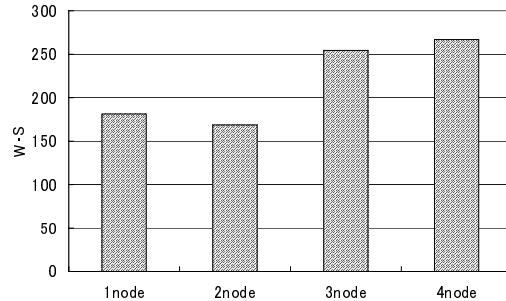


図 8 bsize=128 での Crusoe クラスタ総消費電力

#### 4.3 iPAQ クラスタ

Compaq の iPAQ でクラスタを構築した。iPAQ には CPU は XScale ではなく、StrongARM が使用されている。クラスタ構築にあたり、debian をベースに作られている Familiar と呼ばれる Linux を内部のフラッシュメモリにインストールした。iPAQ はその構成上直接電流を測ることができないため、交流電源で測定を行なった。また、iPAQ には FPU が備わっていないため、性能を測定する際に浮動小数演算が含まれない、NPB の IS と knapsack 問題を選択した。

表 10 に iPAQ クラスタでの IS と knapsack 問題の性能結果を示す。IS の性能はソートのみにかかった実行時間である。この結果から、IS では 1node では XScale とほぼ同じ性能であるが、台数が増えるにつれ性能が低下している。このように通信を多く伴うプログラムでは、ネットワークの性能の低さが全体の性能低下に大きく影響して並列の効果が得られない。

一方、knapsack 問題では、台数が増える毎に性能が向上している。複数台の場合は、ある 1node がリクエストを出しその他の node で探索を行なっている。このために 2node の時は 1node の時と性能が変わらない。

交流電流の電力のデータでは iPAQ8 台で通常時、消費電力は約 30W である。いずれのプログラムでも実行時には約 45W まで上昇した。このことから、iPAQ クラスタの消費電力は非常に小さいものとなっている。

しかし、性能が P4 の 10 分の 1 以下であるために結果として総消費電力は P4 より大きくなり、残念ながら有効な結果は得られなかった。

	IS CLASS=S	IS CLASS=W	knapsack
single	0.13	6.92	538.24
2node	1.12	22.48	538.59
4node	1.27	28.29	183.99
8node	8.69	38.99	98.09

表 10 iPAQ クラスタでの実行時間 [sec]

## 5. 考 察

### 5.1 CPU の消費電力

本稿では低消費電力 CPU である ARM を使用している iPAQ と Crusoe を使用してクラスタを構築し評価を行なった。

Crusoe クラスタでは、性能も大きく向上し全体の総消費電力も小さくすることができた。

科学技術計算の分野であれば、プログラムが並列化してある場合が多く、性能が低くても低消費電力な CPU をたくさん接続しクラスタを構築することは性能、電力両方の面で高性能で高消費電力な Pentium のようなプロセッサを用いるより性能が良い場合がある。

iPAQ クラスタでは元々の CPU の性能が低く、またネットワークの性能も悪いために有効な結果は得られなかった。

### 5.2 冷却・実装

Pentium4 は高速に実行が可能であるが、Crusoe に比べ瞬間の消費電力が非常に大きくなり、瞬間の消費電力が大きいため発生する瞬間の熱量が多くなる。よって、定常的に冷却をしなければならず、大きな冷却装置が必要になる。これが、高性能 CPU を高密度で実装する場合の最大の問題である。一方、Crusoe や XScale はどれだけプログラムを実行しても、CPU 自体にはファンなどの冷却装置が必要ないため、高密度な実装が可能になる。また、これらの低消費電力 CPU は使用されない時は通常時より電力を下げるできるので、定常的な電力を大幅に削減できる。これらは、高密度で低消費電力なクラスタを構築する時に重要なことである。

### 5.3 プログラム最適化による低消費電力化

行列積の結果や LU 分解の結果より低消費電力 CPU におけるキャッシュブロッキングは低消費電力化に有効であるということが確認できた。このようにキャッシュを有効に使用することで大幅な低消費電力が実現できる。今後は他のアプリケーションに関してもブロッキングを施し低消費電力化を測ることが必要である。また、ソフトウェアで自動的にブロッキングを施すことができるような機構を備えれば、ユーザに意識させることなく低消費電力化が可能になるであろう。

### 5.4 ネットワークインタフェースの消費電力

通信に関しては、ネットワークカードの消費電力が CPU の消費電力に比べても小さいので全体の消費電力に影響を与えることはほとんどない。ちなみにカードなしの時には無負荷時の消費電力が 1W 強減少する。カードをさした場合、定常的にこの消費電力を消費し、通信を行なっても消費電力の変化は観測できなかった。

## 6. おわりに

本研究では、Pentium4 や XScale, Crusoe の消費電力をダイナミックに測定することにより各 CPU の電力特性を評価した。各 CPU の電力特性として、Pentium4 は非常に性能が高いが、瞬間の消費電力も非常に高くなるため Pentium4 には大きな冷却装置が必要になる。低消費電力 CPU は瞬間の消費電力は低いために冷却装置が必要ない。また、キャッシュを有効に利用することで大幅に電力を削減できる。

また、低消費電力なプロセッサをクラスタに用いることによって、Pentium4 よりも性能が高く、総消費電力も低くすることが実現できる可能性があることを示した。

ソフトウェアからのアプローチとして、キャッシュブロッキングを行いプログラムを最適化することで大幅に総消費電力を削減できることを示した。

## 謝 辞

本研究（の一部）は科学技術進行事業団・戦略的創造研究「低消費電力化とモデリング技術によるメガスケールコンピューティング」による。また、様々な御助言をいただいた CREST チームの方々に感謝します。

## 参 考 文 献

- 1) Simon Segars, "Low Power Design Techniques for Microprocessors", ISSC, Feb. 2001.
- 2) W. Feng, M. Warren, E. Weigle, "The Bladed Beowulf: A Cost-Effective Alternative to Traditional Beowulfs", IEEE Cluster 2002, September 2002.
- 3) M. Warren, E. Weigle, W. Feng, "High-Density Computing: A 240-Node Beowulf in One Cubic Meter", SC2002, November 2002.
- 4) Intel "Intel XScale<sup>TM</sup> Microarchitecture Technical Summary", Intel Corporation, 2000.
- 5) "Crusoe Processor Model TM5800 Product Brief", Transmeta Corp, Feb. 2003.
- 6) Johan Powelse, Koen Langendoen, Henk Sips "Dynamic Voltage Scailing on a Low-Power Microprocessor", UbiCom-Technical Report, 2000.
- 7) 寒川 光, "RISC 超高速化プログラミング技法", 共立出版, p.113-148, 1995.