

## 革新的電源制御による超低消費電力高性能システム LSI の構想

中村 宏<sup>†</sup> 天野 英晴<sup>††</sup> 宇佐美公良<sup>†††</sup> 並木美太郎<sup>††††</sup>  
今井 雅<sup>†</sup> 近藤 正章<sup>†</sup>

<sup>†</sup> 東京大学 先端科学技術研究センター 〒153-8904 東京都目黒区駒場 4-6-1

<sup>††</sup> 慶應義塾大学大学院 理工学研究科 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

<sup>†††</sup> 芝浦工業大学 工学部 情報工学科 〒135-8548 東京都江東区豊洲 3-7-5

<sup>††††</sup> 東京農工大学 大学院共生科学技術研究院 〒184-8588 東京都小金井市中町 2-24-16

あらまし 我々は、科学技術振興機構の戦略的創造研究推進事業（CRESTタイプ）の「情報システムの超低消費電力化を目指した技術革新と統合化技術」領域において、平成18年度より「革新的電源制御による次世代超低電力高性能システム LSI の研究」の研究課題を実施している。本プロジェクトは、回路実装、アーキテクチャ、システムソフトウェアの各階層が真に連携・協調し、革新的な電源制御を実現することで高性能システム LSI の消費電力を格段に低下させることを目指している。本稿では、本プロジェクトの構想と目標達成への戦略について述べると共に、現在実施している各研究項目の一部について概要を述べる。

キーワード システム LSI, 低消費電力, 電源制御, 階層間協調設計, データレジデントコンピューティング

## A Plan of Innovative Power Control for Ultra Low-Power and High-Performance System LSIs

Hiroshi NAKAMURA<sup>†</sup>, Hideharu AMANO<sup>††</sup>, Kimiyoshi USAMI<sup>†††</sup>, Mitaro NAMIKI<sup>††††</sup>,  
Masashi IMAI<sup>†</sup>, and Masaaki KONDO<sup>†</sup>

<sup>†</sup> Research Center for Advanced Science and Technology, The University of Tokyo  
4-6-1 Komaba, Meguro-ku, Tokyo, 153-8904 Japan

<sup>††</sup> Graduate School of Science and Technology, Keio University  
3-14-1 Hiyoshi, Kouhoku-ku, Yokohama, 223-8522 Japan

<sup>†††</sup> Department of Information Science and Engineering, Shibaura Institute of Technology  
3-7-5 Toyosu, Kohtoh-ku, Tokyo 135-8548, Japan

<sup>††††</sup> Department of Computer, Information and Communication Sciences, University of Agriculture and  
Technology  
2-24-16 Nakamachi, Koganei, 184-8588 Japan

**Abstract** We have started a research project of “Innovative Power Control for Ultra Low-Power and High-Performance System LSIs” since 2006 supported by Japan Science and Technology Agency as a CREST research program. The objective of this project is drastic power reduction of high-performance system LSIs. We are planning to achieve this goal by innovative power control realized by tight cooperation and cooptimization of various design levels including circuit, architecture, and system software. In this paper, we present the plan, the strategy, and the current status of our project.

**Key words** system LSI, low power consumption, power control, cooperative design, data resident computing

### 1. はじめに

携帯電話・端末、デジタルハイビジョン TV、カーエレクトロニクスなど、高性能なシステム LSI が我々の生活で多く利

用されているが、より高度で快適な情報化社会を実現するために、さらなる高性能化・高機能化が求められている。しかし、半導体微細化により VLSI に搭載可能なトランジスタ数は今後も増加するもののそれに伴い消費電力も増加するため、システ

ム LSI のさらなる高性能化は消費電力の面で限界にきている。また、これらのシステム LSI が社会の基盤技術として今後さらに飛躍的に普及することを考えると、限られた地球上のエネルギーでこの社会が要求する情報処理能力を実現するためには、従来よりも格段にシステム LSI を低消費電力化する必要がある。本研究の目的は、快適な高度情報化社会を支える高性能システム LSI のさらなる高性能化と低消費電力化を実現することである。

回路レベルで考えると、低消費電力化の唯一の手段は、不要不急のスイッチングをする素子の電源制御である。従来も、スタンバイ時（非動作時）の電源遮断や、動作モードに応じた DVFS による電圧制御はなされていたが、これらは時間的にも空間的にも粗い粒度で行われていた。しかし、高性能 VLSI のさらなる低消費電力化を実現するためには、動作時の電源制御をより積極的に、つまり時間的にも空間的にもより細粒度に実現する必要があると我々は考えている。不要不急のスイッチングがどれだけあるのかについては、以下の観察がある。高性能マイクロプロセッサはこれまで、搭載トランジスタ数は3年でほぼ4倍（いわゆるムーアの法則）、動作周波数も3年でほぼ2倍向上しているため、スイッチング動作は3年で8倍増加している。その一方で、実効性能は文献[1]によると年率約55%すなわち3年で約4倍の向上となっている。このことは、性能向上に寄与しない、すなわち不要不急のスイッチング動作が3年で2倍になっていることを意味している。我々は、この不要不急のスイッチング動作が消費する電力を、回路実装、アーキテクチャ・コンパイラ、システムソフトウェアが階層を越えて真に協調し、革新的な電源制御を行うことで削減することを目指している。

階層を超えた協調を実現するための統一概念として、データレジデントコンピューティングを我々は提唱している。データレジデントコンピューティングというのは抽象的な概念であり、データに対する処理を時間的・空間的に閉じ込めたものと言える。この概念により、不要不急のスイッチング動作を顕在化し、階層間で協調して低電力化のために活用することが可能になる。

この概念では、例えば VLSI アーキテクチャのあるべき姿としては、従来の処理スループット重視の下でいかに効果的にトランジスタ資源を投入するかという最適化ではなく、VLSI 内部で動作すべき真に必要なトランジスタ資源だけを活性化できるような最適化を目指すことになる。アーキテクチャは、設計階層においてソフトウェアとハードウェアのインタフェースとなる重要な抽象化レベルであるが、その意味は時代と共に変わってきている。つまり、メインフレームの時代にはハードウェア機能としての命令セットだけを考慮してソフトウェアを実現していたが、RISC の時代になると命令セットだけでなくパイプライン処理等の命令実行制御方式まで、つまりハードウェアによって実現される機能とタイミングの両方を考慮してソフトウェアが実現されるようになってきている。我々は、低消費電力という新しい目標に対しては、機能・タイミングだけでなく時空間上の活性度も考慮できるインタフェースとしてアーキテクチャを再検討し、このデータレジデントコンピューティングによる画期的な低消費電力化を実現したいと考えている。

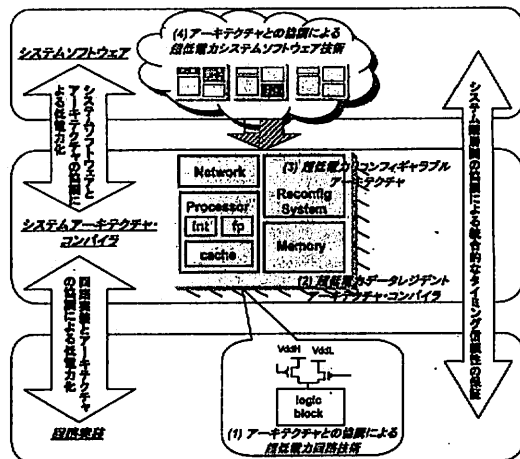


図1 研究項目と研究体制

## 2. 研究方針

1. 章で述べたように、回路実装、アーキテクチャ・コンパイラ、システムソフトウェアが階層を越えて真に協調し、VLSI 内部で動作すべき真に必要なトランジスタ資源だけを活性化する革新的な電源制御を行うことで、高性能システム LSI のさらなる高性能化と低消費電力化の実現を目指している。その目標を達成するために、図1に示すように以下の3つの技術開発を柱としている。

(1) 回路実装とアーキテクチャの協調による低電力化  
素子レベルで考えれば、低消費電力化の唯一の手段は不要不急のスイッチングをする素子の電源制御である。低電力化の機会を増やすためには、電源制御の空間的・時間的粒度は細かいほうが良い。しかし、空間的粒度には回路実装上限界があり、電源電圧の変更には必ず時間を要することから時間的粒度にも限界がある。一方、性能面では、アーキテクチャレベルの高速化手法が活用する局所性に電源制御の粒度をあわせることが重要である。そこで、回路実装とアーキテクチャレベルの協調により、処理の局所性の粒度（データレジデントコンピューティングの粒度）を両者であわせることで電源制御を最適化し、低消費電力と高性能化の両方を実現する。この協調においては、回路レベルで電源制御に要する遅延時間が性能に与える影響を抑えることも重要であり、アーキテクチャから回路に対し電源の先行制御を指示することでその遅延時間を隠蔽可能とするなどの両者の密な協調が必要となる。

(2) システムソフトウェアとアーキテクチャの協調による低電力化  
大規模システム LSI はもはやシングルタスク環境ではなくマルチタスク環境で動作している。資源競合が動的に発生するマルチプロセス環境下では、前項のプロセス単位での電源管理機構だけでは限界があり、プロセス全体の消費電力を考慮した資源割当てが、超低電力かつ高性能な計算機システムの実現に必要となる。そこで、システムソフトウェアとアーキテクチャが協調して、回路実装技術が提供する電源制御機構を用いることを

目指す。この協調を実現するために、アーキテクチャ側は、システムソフトウェアからの電源制御が効果的となるように、電源制御に必要な情報を十分にシステムソフトウェア側に提供でき、システムソフトウェアが観測可能な事象から電源制御方針を決定容易なアーキテクチャの実現を目指す。システムソフトウェア側は、提供される情報を元に電源制御機構を有効に活用して電力を削減し、しかも高速処理を可能とするようなプロセススケジューリング、プロセス単位での資源管理の研究を行う。

(3) システム階層間の協調による統合的なタイミング信頼性の保証

プロセス・電圧・温度ばらつきやクロストークに起因するタイミングのばらつきは、微細化に伴い深刻化するため、回路レベルでタイミング制約違反（フォールト）を完全排除するのは性能的にも電力的にも損失が大きい。例えば、ラッチ間最悪遅延よりサイクル時間を長くする必要のある従来の設計手法では、多くの場合にサイクル時間より短い時間で動作が完了しており、必要以上の電力を消費して無駄に高速処理をさせている。微細化に伴いタイミングのばらつきが深刻化すると、この性能・消費電力両面での損失はさらに大きくなる。そこで、回路レベルでの完全なフォールトアボイダンス (fault avoidance) は諦め、下位層のフォールトを上位層でマスクする、階層間協調によるフォールトトレランスを実現する。高性能・低消費電力・高信頼性を達成する上で重要なのは下位層で可能な限りのアボイダンスを目指すことであり、下位層のアボイダンスでは性能・電力的な損失が大きいフォールトのみを上位層に伝播させること、および、下位層でのフォールト発生頻度を上位層から可能な限り低減すること、の2点が重要となる。

このような、システム階層の協調に基づく低消費電力高信頼なシステム LSI を実現するための基礎技術として、図 1 に示すように以下の 4 項目の研究を行う。

- アーキテクチャ協調型超低電力回路技術（芝浦工業大学・宇佐美グループ）
- 超低電力データレジダントアーキテクチャ・コンパイラ（東京大学・中村グループ）
- 超低電力動的リコンフィギュラブルアーキテクチャ（慶應義塾大学・天野グループ）
- 超低電力を実現するアーキテクチャ協調型システムソフトウェア（東京農工大学・並木グループ）

### 3. 回路実装とアーキテクチャの協調による低電力化

現在、半導体の微細化により深刻になる動作時のリーク電力の削減を、回路実装とアーキテクチャの協調により実現する手法の検討を行っている。その検討状況を述べる。

3.1 ランタイムパワーゲーティングによる低電力回路技術  
リーク電力を低減する技術として、パワースイッチを用いて、動作中の回路を時間的にも空間的にも細粒度にスリープ制御するランタイムパワーゲーティング (Run Time Power Gating, 以下、RTPG) がある [2]。この論文ではゲーティッドクロックのイネーブル信号に着目し、同じイネーブル信号によって制御される論理ゲートの集合 (PG-ドメイン) を単位として、パワ-

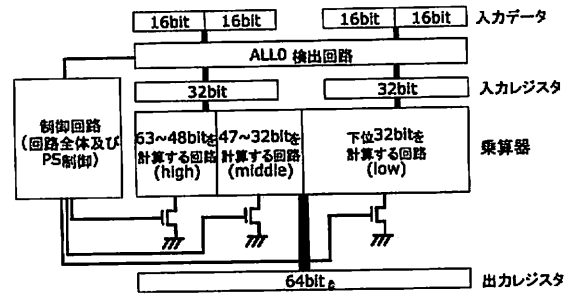


図 2 RTPG を適用した乗算器の構成図

ゲーティングすることが可能な回路構成が提案され、その適用範囲の広さと電力削減効果を示されている。しかし、RTPG を用いた場合には、パワースイッチ回路を挿入することによるスリープ制御対象回路の遅延時間増加、パワースイッチを制御するために必要となる電力増加、および、回路をスリープ状態から動作状態に復帰させるのに必要となるウェイクアップ時間などが問題となる可能性がある。そこで、これらの評価を実機で行うために、チップの設計・試作を行った [3]。

#### 3.1.1 試作チップの概要

RTPG を適用する回路として 32bit x 32bit 乗算器を用いた。乗算器は、入力値によっては演算を行わない部分回路が多数存在する。そこで、演算を行わない部分回路をスリープさせることにより、部分回路でのリーク電力を低減させることができる。この乗算器の構成を図 2 に示す。この乗算器は、パワーゲーティングの状態として以下の 4 つをもつ。各状態においてパワーゲーティングする部分回路を変える事により、走行中の乗算器において効率的に電力を低減できることが特徴である。

状態 1 被演算数の大きさが共に 17bit 以上の場合：この状態では乗算器は回路全体が使用されているのでパワーゲーティングは行われない。

状態 2 片方の被演算数の大きさが 16bit 以下の場合：出力値の上位 16bit は常に '0' であり、演算を行っていないため、上位 16bit を演算する部分回路 (high) をパワーゲーティングする。

状態 3 被演算数の大きさが共に 16bit 以下の場合：この場合、出力値の上位 32bit を演算する部分回路 (high + middle) がパワーゲーティングされる。

状態 4 乗算回路自体が使用されない場合：この場合、乗算器自身が使用されていないので乗算器の回路全体 (high + middle + low) をパワーゲーティングすることが可能である。

試作には ASPLA90nm プロセス (6 層 Cu 配線) を用いた。ASPLA90nm には  $V_t$  の差により、高速動作・高リーク電流の HP トランジスタ、低速動作・低リーク電流の MP トランジスタがあり、乗算器と ALLO 検出回路の論理ゲートには HP トランジスタを、パワースイッチ、ホルダーセルと上記以外の論理ゲートには MP トランジスタをそれぞれ用いた。

試作したチップの写真を図 3 に示す。チップには、RTPG を適用した乗算器と適用しなかった乗算器がそれぞれ 2 つ実装されている。乗算器自体の面積は、 $226.8 \times 544.32 \mu\text{m}^2$  である。

#### 3.1.2 評価結果と検討事項

実際に計測した結果を 2 つの表に示す。表 1 は、ダイナミッ

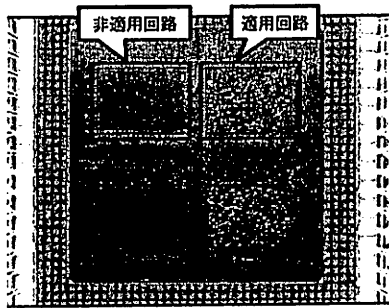


図3 RTPGを適用した乗算器のチップ写真

ク電流とリーク電流をあわせた総電流、表2は、入力値を一定にすることでリーク電流だけを測定した結果である。それぞれ20℃と85℃で測定しており、周波数は100MHz、電源電圧は1.0Vとしている。なお、表中の括弧内の値は、状態1に対する削減率を表す。

表1 総電流の実測値 (mA)

	状態1	状態2	状態3
20℃	5.79	5.67 (-2.0%)	4.85 (-16.2%)
85℃	6.50	6.36 (-2.1%)	5.40 (-16.9%)

表2 リーク電流の実測値 (mA)

	状態1	状態2	状態3
20℃	1.01	0.87 (-13.8%)	0.80 (-20.7%)
85℃	1.90	1.60 (-15.7%)	1.40 (-26.3%)

これらの表からわかるように、RTPGによる電力削減効果を実チップを用いて確認することができた。しかしながら、その削減効果は期待通りではない。もしパワースイッチが理想的に働けば、状態3ではリーク電流を50%削減できるはずなのに、実測では85℃でも25%ほどの削減に留まっている。この原因として、パワースイッチをドライブするためのバッファツリーを高リーク電流のHPトランジスタを用いて構成したこと、パワースイッチに用いたMPトランジスタ自体にも多少リーク電流が流れること、などが、シミュレーションによる追評価からわかっている。また、これらの要因とも関連するが、パワースイッチ自体をon/offする電力的なオーバーヘッドもそれなりに大きく、このオーバーヘッドとパワーゲーティングによるリーク電力の低減効果が同じになる消費電力上の損益分岐点となるスリープ期間 (Eoverthead-period) は約10サイクル程度になることもわかった。現在、これらの課題を解決すべく回路構成に改良を加え評価・検討している。

### 3.2 リーク電力削減のための細粒度命令スケジューリング手法

前節で述べた回路技術と協調して電力削減を目指すアーキテクチャ技術として現在

- RTPGを適用したマイクロプロセッサの試作 [4]
- RTPGによる電力削減効果を最大化するマイクロアーキテクチャ技術の検討 [5]

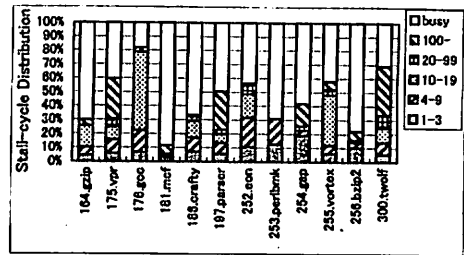


図4 ALU部のストールサイクル時間の分類

の2つの研究を実施中である。前者は、RTPG技術をマイクロプロセッサに実際に適用し、その電力削減効果を実証することを目指している。またこのプロセッサの上でOSを動作させ、システムソフトウェアによる電力削減手法を実機上で示すためのプラットフォームとしても利用する予定である。現在検討していることの詳細は文献 [4] に譲り、本稿では後者に関する検討状況を述べる。

#### 3.2.1 動作時リーク電力削減の課題

効率的な実行時リーク電力削減のためには、時間的・空間的に細粒度に電源電圧供給制御を行うことが望ましく、オーバーヘッドが小さなパワーゲーティング手法においても、モードの切り替え時には性能およびエネルギー面において、ある程度のオーバーヘッドが生じる。そこで、リーク電力削減効果を最大化するためには、パワーゲーティングの時間粒度をできるだけ大きくするマイクロアーキテクチャ技術が必要となる。これは、処理を空間的・時間的に閉じ込め、処理を行う際はできるだけ一度に大量の処理を、またストール時にはできるだけ長い間ストールする、というデータレジデントコンピューティングの実現を目指すものである。

まず、実際のプログラム実行中にどの程度の時間ALUがアイドルであるかを評価した。そもそもアイドル時間が短ければRTPGによる電力削減効果はさほど期待できないからである。図4は、SPEC2000整数ベンチマークの各プログラムについて、ALUのアイドルサイクル1回の長さ毎に、それらが全実行時間に占める割合を示している。4命令同時発行 superscalar方式、ALUは2つ搭載するモデルで評価を行った。図中、例えば“1-3”は、1回のアイドル時間の長さが1, 2, および3サイクルのものが、合計で実行時間中の何%を占めていたかを示している。

図4より、ALUがbusyである時間の割合はプログラムによってはそれほど多くなくRTPGによる電力削減の機会是十分にある一方で、20サイクル以下のような比較的短い期間のアイドルが占める割合が多いこともわかる。3.1節で述べたようにEoverthead-periodが10サイクル前後であると仮定すると、このままではパワーゲーティングによるリーク電力削減効果は十部には得られないこともわかる。

図5は、パワースイッチ自体の動作に時間的なオーバーヘッドはないと仮定し、Eoverthead-period以上のストールのみをモード切り替えの対象とした際に、Eoverthead-periodが0, 5, 10, 20サイクルの4通りであったと仮定した場合の、ALU部の実行時間の分類を示したものである。この図では、実行時間

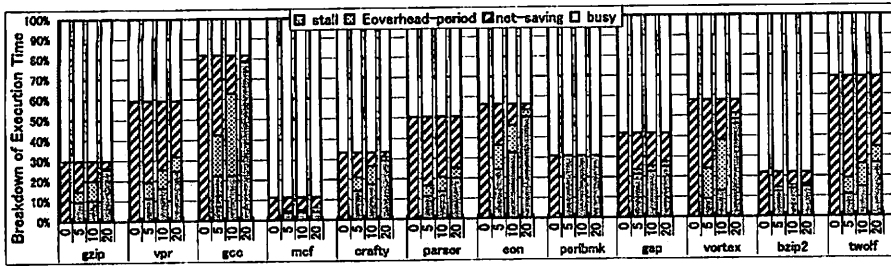


図5 ALU部にパワーゲーティングを適用した場合の実行時間の内訳

は通常モード時のストール (stall)、スリープモード時のオーバヘッドサイクル (Eoverhead-period) すなわちスリープモードには入ったが電力を削減できないサイクル、スリープモード時にリーク電力を削減できたサイクル (net-saving)、および演算を行っていたサイクル (busy) の4つに分類している。図より、Eoverhead-periodが0であればnet-savingの部分が占める割合が大きいためリーク電力を大きく削減できるが、Eoverhead-periodが大きくなるとbusy以外のアイドルサイクルの中でnet-savingの占める割合が減少し、リーク電力を効率的に削減できないことがわかる。Eoverhead-periodが5サイクル程度でも大きくnet-savingが減少してしまうものが多いことから、一回のアイドル時間が長くなるような実行方式が必要であると考えられる。

### 3.2.2 提案する命令実行制御

そこで、一回のアイドル時間が長くなるような命令実行制御の1つとして、データキャッシュミスが生じた際の命令実行を制御することで、net-savingの時間を最大化する手法を検討している [5]。

近年のマイクロプロセッサは、データキャッシュミス発生時にも依存がなく実行可能な命令を処理するノンブロッキングキャッシュを採用するものが多い。しかし、キャッシュミス解決のためのデータ転送中に新たにキャッシュミスが生じ、キャッシュミスによるデータ転送要求が積み重なると、プロセッサはストールによるデータ転送要求が積み重なると、プロセッサはストールと交互に表れ、効率的にパワーゲーティングを行うことができない。そこで、キャッシュミスによるデータ転送要求が2つ以上積み重なった場合は、実行可能な命令がある場合でも、ALUやFPUなどの演算器部をスリープ状態にすることを提案する。

このようにすると実行可能な命令がある場合でもスリープ状態にしてしまうため性能が低下する可能性もあるが、回路技術と協調したリーク電力の削減のためには、このような「めりはり」型の命令実行制御方式が有望と考える。また、今回はL1キャッシュにこの手法を適用する。これによりL2キャッシュレテンシ (通常は10サイクル程度) に相当する時間単位で処理をまとめられることが期待でき、3.1節で得られた10サイクル程度のEoverhead-periodとの親和性が高いからである。

評価したところ、上記手法により、Eoverhead-periodが10以下の場合には全実行時間におけるnet-savingの割合を、最高で12%、平均でも2%程増加することができた。一方でEoverhead-periodが20の場合にはnet-savingをほとんど増加させること

ができないこともわかった。この結果から、当初の予想通り、L1のキャッシュミスという10サイクル相当の時間単位で処理を効果的にまとめることができ、回路技術によって決定されるEoverhead-periodに応じて効果的にリーク電力を削減できることがわかった。

## 4. システムソフトウェアとアーキテクチャの協調による低電力化

### 4.1 統計情報に基づく実行時最適化

プログラムの実行に関して種々の最適化が強く求められてきているが、とりわけ実行時における最適化の重要性が増してきている。一般に実行時、つまり動的な最適化手法を静的な最適化手法と比べた場合の利点は、データセットの違いや、ハードウェア構成などの違いからくる動的な振る舞いに対処することが可能であり、効率的な実行を提供できるという点にある。

一方で、こういった最適化を行うためには計算機の振る舞いを解析・理解し、どのような場合にどういった最適化を行うかというモデルを立てることが必要となる。しかし、計算機システムは年々加速的に複雑化してきており、計算機の振る舞いを定性的に理解することが非常に困難になってきている。

そこで我々は、このような問題に対処しつつ種々の最適化を可能にするために、計算機システムの振る舞いを定量的に解析し実行時に最適化を行う手法を提案している [6]。

図6に提案手法の概要を示す。まず、あらかじめ最適化対象の各モード (例えば、取りうる各周波数・電圧) での各種アプリケーション実行中のパフォーマンスカウンタの値と性能を取得する。それを基に、モードを変更した際に性能がどう変化するかを予測する上で参照すべきカウンタと性能予測式を統計的に学習する。学習結果はライブラリに保存され、コンパイラはターゲットとなるアプリケーションのソースコードに対し、パフォーマンスカウンタの値を用いて、性能を予測するためのライブラリコールを挿入する。また、予測された性能に基づき、モードを変更するためのランタイムコードも挿入する。

得られたカウンタ値を独立変数とし、異なるモードへ変更した際の性能比率を従属変数として重回帰分析を行うことで、性能を予測するための回帰式を求める。それを用い、現在のカウンタの値から次のフェーズにおける各モードでの性能を予測し、次のフェーズでは最適と予測されたモードで実際に実行を行う。なお、用いた統計手法の詳細については、文献 [6] を参照されたい。

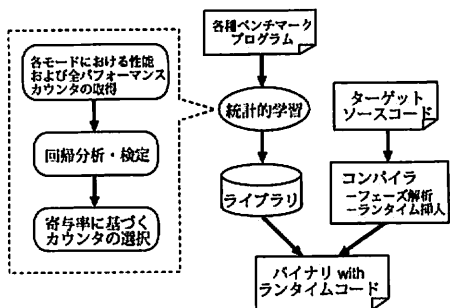


図 6 統計情報に基づく実行時最適化手法の概要

実際に Pentium M 760 プロセッサを搭載する PC に適用して、性能制約下で出来るだけ低い電源電圧・周波数を選択して低消費電力化を図る実験を行った。その結果、L2 STM (Level 2 store misses) と L2 TCM (Level 2 total cache misses) のカウンタが選択され、この 2 つのカウンタを用いたモデリングが性能制約下での低消費電力化に極めて有効であることが確認できた。

現在は、ランタイムコードをソースコード中に埋め込んでいるが、計算機システムの振舞いを実行時にプラットフォーム非依存でモデル化するこの手法は、システムソフトウェアによる最適化にそのまま自然に適用することができるため、今後システムソフトウェアと協調した手法の確立を目指す予定である。

#### 4.2 省電力システムのための QEMU を用いたシステム評価環境の実現

ハードウェア技術の向上により、非常に細粒度の電力制御技術が利用可能になってきており、既に実用化されている周波数・電圧制御手法だけでなく、計算機内部のユニット単位での電力制御も可能になると考えられる。これらの技術をシステムソフトウェアから活用してシステム全体の省電力化を目指す場合、主記憶に対する電源制御は非常に有効であることが期待される。そこで、内部ユニット単位での電源管理機能が追加された主記憶として「複数のブロックに等分され、ブロック単位で電源の ON・OFF を管理できる主記憶」を想定し、未使用ブロックの電源を OFF にすることで、そのブロックの待機電力を削減し省電力化の実現を考える。

この時、ブロック分割された主記憶で省電力化を達成するには、OS などシステムソフトウェアが使用ブロック数を少なく抑えるメモリ管理を行う必要がある。つまり、各ブロックの使用状況の把握、使用ブロック数を少なく抑えるためのメモリ割当て、場合によってはあるブロックに割り当てられた領域を他のブロックにマイグレーションする、などの処理が要求される。しかし、メモリ管理の内容によっては、時間的にも電力的にも無視できないオーバーヘッドが発生する。また、各ブロックの電源 ON・OFF 操作でもオーバーヘッドは発生すると考えられる。そのため、オーバーヘッドとのトレードオフを含めた省電力メモリ管理手法の設計・評価環境が必要となる。そこでそのような設計・評価環境の実現を目指し、命令・メモリアクセストレーサの構築を QEMU を改造することでやっている。実行された機械語、アクセスされた仮想アドレス、物理アドレス、アクセス時の CPU 特権レベル、アクセスしたプロセスの PID (Linux 系 OS の場合) がトレース対象である。

現在、x86 用 CUI 版 Fedora Core5 からのトレースにも成功しており、Celeron1.0GHz、主記憶 384MB の実機で電源投入から Linux のログイン画面表示までのトレース採取に約 7800 秒を要した。しかしその実行時間の大部分は計 215GB にもおよぶトレースデータの保存に要しており、出力先を /dev/null と指定した場合には約 600 秒で終了する。実機ではログイン画面表示まで約 60 秒を要するので I/O のオーバーヘッドを仮に隠蔽できたとすると、実機の約 1/10 の速度という非常に高速なトレース採取を実現できることになる。

## 5. おわりに

本稿では、平成 18 年度より開始した「革新的電源制御による次世代超低電力高性能システム LSI の研究」の研究プロジェクトの構想と目標達成への戦略、および、現在実施している各研究項目について、特に回路技術とアーキテクチャ協調を中心に述べた。紙面の都合で割愛したが、そのほかにも動的リコンフィギュラブルプロセッサ MuCCRA-1 の設計と試作 [8] を行い、今後このプロセッサへも RTPG の適用やシステムソフトウェアからのコンテキスト制御など、階層間の協調による低電力化を推し進める予定である。

回路実装・アーキテクチャ・システムソフトウェアが、階層を超えて真に協調することで新しいブレークスルーをもたらし、革新的な電源制御による格段な低消費電力化を本プロジェクトで実現したい。

謝辞 各研究グループの、大学院学生を含む参加メンバーには定例の打ち合わせで多くの有益な議論に参加してもらい、また本プロジェクトの推進に大きく貢献していただいております。ここに感謝申し上げます。

## 文 献

- [1] J.Hennessy and D.Patterson, "Computer Architecture A Quantitative Approach", Morgan Kaufmann Publisher, 2006
- [2] K.Usami and N.Ohkubo "A Design Approach for Fine-grained Run-Time Power Gating using Locally Extracted Sleep Signals", Proc. of ICCD'06, Oct. 2006.
- [3] 武田清大, 香嶋俊裕, 大久保直昭, 白井利明, 宇佐美公良. "走行時パワーゲーティングを適用した低消費電力乗算器の物理設計と試作", 電子情報通信学会技術研究報告 VLD2006-74, pp.13-18, 2006 年 11 月.
- [4] 関直臣, 長谷川掲平, 天野英晴, 大久保直昭, 武田清大, 香嶋俊裕, 白井利明, 宇佐美公良, 近藤正章, 中村宏, "MIPS R3000 における細粒度動的スリープ制御方式の提案", 情報処理学会研究報告 ARC-173 / 電子情報通信学会技術研究報告 ICD2007, 2007 年 5 月.
- [5] 近藤正章, 中村宏, "リーク電力削減のための細粒度命令スケジューリング手法の検討", 情報処理学会研究報告 ARC-170, pp.49-54, 2006 年 11 月.
- [6] H.Sasaki, Y.Ikeda, M.Kondo, and H.Nakamura, "An Intra-Task DVFS Technique based on Statistical Analysis of Hardware Events", Proc. of Computing Frontiers 2007, May 2007.
- [7] 松尾和弥, 佐藤未来子, 並木美太郎, "省電力システムのための QEMU を用いたシステム評価環境の実現", 先進的計算機システムシンポジウム SACSIS2007, May 2007.
- [8] 長谷川掲平, 堀聡, 中村拓郎, 西村隆, 佐野徹, 加藤勝, 齊藤正太郎, 天野英晴, "動的リコンフィギュラブルプロセッサ MuCCRA-1 の実装と評価", 先進的計算機システムシンポジウム SACSIS2007, May 2007.