

戦略的創造研究推進事業 CREST  
研究領域「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」  
研究課題「ソフトウェアとハードウェアの協調による  
組込みシステムの消費エネルギー最適化」

## 研究終了報告書

研究期間 平成17年10月～平成23年3月

研究代表者：高田 広章  
(名古屋大学大学院情報科学研究科、教授)

## § 1 研究実施の概要

情報家電や情報携帯端末などの組み込みシステムを対象として、サービス品質（リアルタイム性能、顧客満足度、信頼性など）を保証しつつ、消費エネルギーを最小限にするためのソフトウェア／ハードウェア協調最適化技術を開発した。本研究を通じて数多くの技術を開発したが、その中で特に重要な成果を以下に説明する。

### マルチパフォーマンスプロセッサの開発

従来、動的に電源電圧とクロック周波数を変更する DVFS (Dynamic Voltage/Frequency Scaling) 技術に関する研究は多々行われ、汎用計算機の分野においては、DVFS 技術は実用化されている。しかし、高いリアルタイム性能が求められる組み込みリアルタイムシステムにおいては実用化されていない。本研究では、DVFS プロセッサに代わる省エネルギープロセッサとしてマルチパフォーマンスプロセッサを開発した。マルチパフォーマンスプロセッサは、高速で消費エネルギーの大きい PE と低速で消費エネルギーの小さい PE を CPU コア内に複数搭載し、状況に応じて適切な PE を稼働させる。更に、キャッシュウェイを動的に選択する機能も有している。東芝社製の RISC プロセッサ MeP をベースにマルチパフォーマンスプロセッサのプロトタイプを設計し、その実用可能性についてポストレイアウトシミュレーションにより評価した。マルチパフォーマンスプロセッサは既存の DVFS プロセッサと比較して性能切り替えのオーバーヘッドを約 2 桁小さくすることを可能にした。また DVFS プロセッサと比較して低電圧動作時の電力効率が 30%以上良いことを HSPICE による解析で確認した。

### ULP ソフトウェア開発環境の開発

マルチパフォーマンスプロセッサを対象に、各タスクのリアルタイム性を保証しつつ、消費エネルギーを最小化するソフトウェアを開発するための環境 (ULP ソフトウェア開発環境) を開発した。ULP ソフトウェア開発環境は、アプリケーションタスクの QoS を適正化するソフトウェア工学技術、スクラッチパッドメモリを活用することによりメモリアクセスの消費エネルギーを最小化するコンパイラ／リアルタイム OS 協調技術、ハードウェア構成を変更するポイント (チェックポイント) を決定するプログラム解析技術、実行時にデッドラインまでの余裕時間 (スラック時間) を計算する技術、チェックポイントにおいて最適なハードウェア構成を決定するリアルタイム OS を中心とした DEPS (Dynamic Energy/Performance Scaling) 技術、命令セットシミュレーションを元に消費エネルギーを正確に見積もる技術などを含む。ITRON 仕様ベースのマルチプロセッサ向けリアルタイム OS「TOPPERS/FMP カーネル」や MeP 用ソフトウェア開発環境 (コンパイラ、シミュレータ等) を拡張したり、ツールを新たに開発したりすることにより、統合的なソフトウェア開発環境を構築した。

### 統合評価システムの開発と総合評価

最終年度には、本プロジェクトの成果を総合的に評価することを目的として、統合評価システムを開発した。開発した統合評価システムは、テレビ会議システム向け組み込みシステムのバーチャルプロトタイプ (シミュレーション環境) である。統合評価システムには、本研究プロジェクトで開発した技術の一部が実装されている。本統合評価システムによりテレビ会議システム・アプリケーションを実行したときの消費エネルギーを評価した結果、本研究の成果を適用することにより、プロセッサの消費エネルギーを 32.6%、オンチップメモリ (キャッシュとスクラッチパッドメモリ) の消費エネルギーを 73.8%、オフチップメモリの消費エネルギーを 37.2%削減可能であることを示した。また、本研究を通じて開発したすべての技術を適用する場合には、90%以上の消費エネルギー削減を達成できる見通しを得た。

## § 2. 研究構想

### (1) 当初の研究構想

情報家電や情報携帯端末などの組込みシステムを対象として、ソフトウェアとハードウェアの協調により、サービス品質(性能、計算精度、信頼性など)を保証しつつ、消費エネルギーを最小限にするための最適化技術を開発する。メモリアーキテクチャとコンパイラの協調や、低消費エネルギースケジューリング機構を持つマルチプロセッサリアルタイム OS などにより、消費エネルギーを100分の1に低減することを目標とする。

具体的には、以下に挙げる5項目を目標に研究を進める。

1. 見積もりツールおよびリアルタイム OS 向け見積もりモデルの開発(九州大学グループ)  
設計時のアーキテクチャ探索およびコード生成を支援する見積もりツールを開発する。また、実行時にリアルタイム OS が電力管理に利用できる見積もりモデルを開発する。アプリケーションにあわせてHWとSWを最適に切り分け、最適なハードウェア(ハードウェアアルゴリズムや回路構造)を選択することにより、消費エネルギーを1/4に削減する。また、見積もりに基づいて用途に特化した専用ハードウェアを設計することにより消費エネルギーを1/2に削減する。見積もりツールは、30%以下の誤差でQoSと消費エネルギーの値を見積もることを目標とする。
2. 低消費エネルギー化リアルタイム OS の開発(名古屋大学グループ)  
チップマルチプロセッサ(CMP)型の組込みシステムを対象として、消費エネルギーを最小化するリアルタイム OS を開発する。特に、動的に電源電圧と動作周波数を最適に制御するDVS(Dynamic Voltage Scaling)技術と、各コンポーネントの電力モード(休止モード、電源オフなど)を最適に管理するDPM(Dynamic Power Management)技術を柱とする。我々がこれまでに開発してきた $\mu$ ITRON準拠のRTOSであるTOPPERS/JSPカーネルに本技術を実装し、実用化の域にまで高めることを目指す。具体的数値目標は次の通りである。消費エネルギーを1/8に削減することを目標とする。  
(ア) フォールトトレラント技術を積極的に導入することにより電源電圧の下限を既存のDVS技術が想定する値の1/2以下にし、消費エネルギーを1/4に削減する。  
(イ) DPM技術とアーキテクチャ技術の協調によりハードウェアを停止させる機会を増大させると共に、消費エネルギーとQoSの見積もりに基づいてハードウェアモジュールへのタスク割り当てを最適に決定することにより消費エネルギーを1/2に削減する。
3. 低消費エネルギー化コンパイラの開発(名古屋大学グループ)  
メモリシステムの低消費エネルギー化を図る。特に、キャッシュメモリやスクラッチパッドメモリ(オンチップSRAM)を有効に活用するコンパイラ技術を開発する。キャッシュのヒット率を高めるだけでなく、メモリバンク毎に電源電圧や閾値電圧を最適に制御することにより、消費エネルギーを低減する。メモリアーキテクチャおよび目的コードを協調して最適化することにより、ペナルティの大きいオフチップへのアクセス回数を削減すると共に、メモリ自体が消費するエネルギーを削減することにより、システム全体のエネルギーを1/2に削減する。
4. 低消費エネルギー化ハードウェアの開発とそのチップ試作(九州大学グループ)  
CMPの冗長性を利用した低消費エネルギー化ハードウェアを開発する。具体的にはRTOSやアプリケーションプログラムによる電力管理(DPMやDVS)やコンパイラによる低消費エネルギー化を可能にするメモリシステムなどのハードウェアを開発する。平成19年度には個別階層のモデルおよび技術の評価のために提案手法を適用した組込み向け低電力マルチプロセッサのチップ試作、平成21年度には統合技術の評価のためのチップ試作を行う予定である。最終的には、シングルプロセッサに既存の省エネルギー化技術を適用した場合と比較して、消費エネルギーを1/100にすることを目標とする。

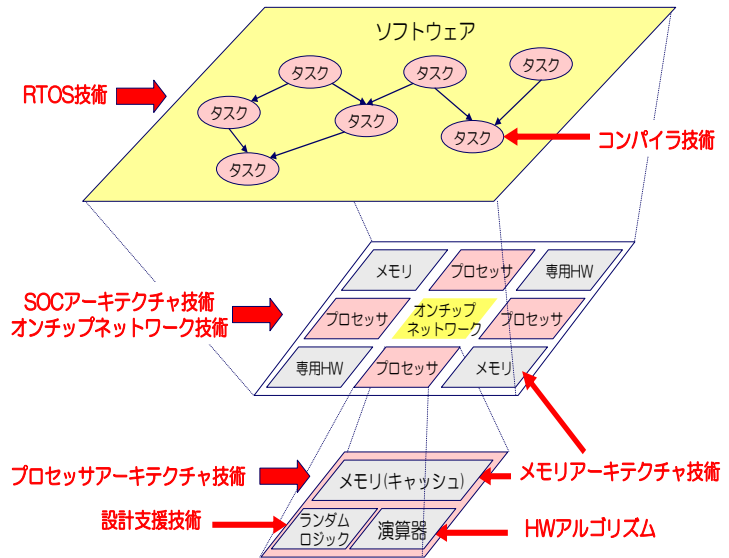
る。

5. ターゲットアプリケーションの解析と評価環境の構築（東芝グループ）

本研究でターゲットとしている 10mW クラスの組込みシステムに対して、消費エネルギープロファイル分析を行い、技術評価の観点として整理し評価環境の構築に役立てる。最終的に、実アプリケーションを用いた実証実験と評価を行う。実現するものは次の3項目である。(1) エネルギー測定ツール、(2) エネルギー分析ツール、(3) ターゲット組込みシステムの開発（周辺装置とアプリケーション）

上記のように、本研究では、回路設計からソフトウェアまでの複数の設計階層を統合的・体系的に扱う。各設計階層は、右に図示するように、以下の設計技術を含む。

- ソフトウェア
  - リアルタイム OS
  - コンパイラ
- アーキテクチャ
  - SOC
  - オンチップネットワーク
  - プロセッサ
  - メモリ
- 回路設計
  - 設計支援
  - HW アルゴリズム



全体的な研究計画は下記の表の通りである。最初の 2～3 年間で設計階層毎に要素技術を開発する。3 年目以降に個々の要素技術を統合する。

項目	平成 17 年度 (6 ヶ月)	平成 18 年度	平成 19 年度	平成 20 年度	平成 21 年度	平成 22 年度 (12 ヶ月)
理論の構築	←			→		
設計階層毎の方式検討	←	→				
設計階層毎の実装		←	→			
設計階層毎の評価			←	→		
階層統合技術の検討			←	→		
階層統合技術の実装				←	→	
階層統合技術の評価					←	→
実用性評価			←			→
全体のまとめ						←

## (2)新たに追加・修正など変更した研究構想

### 研究の範囲及び数値目標の修正 (H19年4月の領域ミーティングで報告)

ULP 領域ミーティングにおける議論を受け、H19年度以降、研究の範囲をプロセッサ、メモリ、および、ソフトウェアに限定した（アプリケーションに応じて専用ハードウェアを設計する技術や、オンチップネットワークを最適化する技術は研究の対象外）。これに伴い、数値目標を（100分の1から）60分の1に変更した。また、数値目標の比較ベースを、2005年時点で産業界で広く使われているシングルプロセッサシステムとソフトウェア開発技術とすることを確認した（アイデアだけ提案されていて産業界で使われていないものは比較ベースに含めない）。

### 新たな研究1：消費エネルギーに関するアルゴリズムの基礎理論 (名古屋大学グループ)

当初計画では想定されていなかったが、H19年度より名古屋大学グループにおいて「消費エネルギーに関するアルゴリズムの基礎理論」に関する研究を展開した（平成20年4月のULP 領域ミーティングで一部報告済み）。本報告書の「§2（3）達成状況」のスケジュール表（名古屋大学・立命館大学）においては、本研究は広い意味で「理論の構築」に含まれると解釈して記述した（元々はスケジューリング理論を想定）。そのため、「理論の構築」の実施期間は、当初計画より延長されている。

### 新たな研究2：省電力ソフトウェア設計手法とQoS要求分析手法 (東芝グループ)

平成19年度より東芝グループにおいて「省電力ソフトウェア設計手法」と「QoS要求分析手法」に関する研究を展開した。当初計画では、東芝グループは実用性評価を担当することになっていたが、単なる「評価」だけでなく、応用の視点から低消費エネルギー化技術を積極的に開発することにした。

### §3 研究実施体制

#### (1)「名古屋大学」グループ

##### ①研究参加者

氏名	所属	役職	参加時期
高田 広章	名古屋大学 大学院情報科学研究科 附属組込みシステム研究センター	センター長／教授	H17.10～H23.3
高木 直史	名古屋大学 大学院情報科学研究科 情報システム学専攻	教授	H17.10～H18.4
富山 宏之	同上	准教授	H17.10～H22.3 (H22.4より立命館大学グループに分離)
中村 一博	名古屋大学 大学院情報科学研究科 附属組込みシステム研究センター	助教	H18.5～H23.3
曾 剛	名古屋大学 工学研究科	研究員～特任助教～講師	H18.4～H23.3
横山 哲郎	南山大学 情報理工学部	研究員(名古屋大学)～講師	H19.6～H23.3
川島 裕崇	名古屋大学 大学院情報科学研究科 附属組込みシステム研究センター	M2～D3～研究員	H18.5～H23.3
渥美 紀寿	同上	研究員	H22.4～H23.3
奥村 朋子	同上	技術補佐員	H19.4～H23.3
菊地 武彦	名古屋大学 大学院情報科学研究科 情報システム学専攻	D1～3	H18.4～H20.9
図子 純平	同上	M1～2	H18.5～H20.3
片岡 寛貴	同上	M1～2	H18.5～H20.3
高瀬 英希	同上	M1～D2	H19.4～H23.3
立松 知紘	同上	M1～2	H21.4～H23.3
三輪 遼平	同上	M1	H22.4～H23.3

##### ②研究項目

- リアルタイム OS を中心としたエネルギー/性能の実行時最適化フレームワーク
- メモリの低消費エネルギー化を実現するコンパイラ/RTOS 技術(九州大学グループ、立命館大学グループと共同)
- 演算器の消費エネルギー削減するハードウェアアルゴリズム
- 消費エネルギーに関するアルゴリズムの基礎理論
- 統合評価システムの開発と評価(全グループの共同)

(2)「九州大学」グループ

① 研究参加者

氏名	所属	役職	参加時期
石原 亨	システム LSI 研究センター	准教授	H17.10～H23.3
松永 裕介	同上	准教授	H17.10～H18.4
佐藤 寿倫	同上	教授～学術研究員	H18.4～H23.3 (H21.4より福岡大学に異動のため職名変更)
久住 憲嗣	同上	准教授	H22.4～H23.3
室山 真徳	同上	助教	H17.10～H20.7
Goudarzi Maziar	同上	学術研究員	H17.10～H20.12
Gauthier Lovic Eric	同上	学術研究員	H21.3～H23.3
Zang Chengjie	同上	学術研究員	H21.5～H22.6
山口 誠一郎	大学院システム情報科学府 情報工学専攻	M2～D3	H17.10～H21.3
El Bourichi Adil	同上	M2～D3	H17.10～H21.3
徳永 将之	同上	M2	H18.4～H19.3
中村 隆二	同上	M2	H18.4～H19.3
坂本 良太	同上	M2	H18.4～H19.3
李 東勲	同上	M1～M2	H18.4～H20.3
山口 聖貴	同上	M1～M2	H18.4～H20.3
大山 裕一郎	同上	M1～M2	H18.4～H20.3
石飛 百合子	同上	M1～M2	H19.4～H21.3
松村 忠幸	同上	M1～M2	H19.4～H21.3
金田 裕介	同上	M1～M2	H20.4～H22.3
奥平 拓見	同上	M1～M2	H21.4～H23.3

② 研究項目

- メモリの低消費エネルギー化を実現するコンパイラ/RTOS 技術(名古屋大学グループ、立命館大学グループと共同)
- マルチパフォーマンスプロセッサ
- データメモリの省エネルギー技術
- プロセッサシステムの電力キャラクタライズと電力見積もりツール
- カナリア FF による電源電圧マージンの最適化
- 省電力メモリおよび省電力レジスタ回路とその制御技術
- 統合評価システムの開発と評価(全グループの共同)

(3)「東芝」グループ

①研究参加者

氏名	所属	役職	参加時期
深谷 哲司	ソフトウェア技術センター ソフトウェアイノベーション 戦略部	部長	H17.10～H23.3
片岡 欣夫	研究開発センター システム技術ラボラトリー	研究主幹	H17.10～H23.3
小林 良岳	ソフトウェア技術センター 先端ソフトウェア開発担当	主務	H18.4～H23.3
安東 孝信	同 ソフトウェア設計技術開発 担当	参事	H18.4～H23.3
神代 剛典	同上	主事	H19.1～H23.3
小原 俊逸	同上	主事	H20.11～H23.3
中里 竜	同上	主務	H21.4～H23.3
野末 浩志	ソフトウェア技術センター	参事	H17.10～H20.9
早乙女 正義	同上	主務	H18.4～H19.5
山田 真大	同上	主事	H19.6～H20.5
家村 慎也	同上	主事	H20.10～H20.6
進 博正	研究開発センター	主任研究員	H18.4～H21.9
酒井 良哲	ソフトウェア技術センター	主事	H19.1～H20.3
西岡 竜大	同上	主務	H19.6～H20.3

②研究項目

- QoS 要求適正化
- 低消費エネルギー能力評価技術の開発
- 統合評価システム用シミュレータとアプリケーションの開発
- 省電力ソフトウェア設計手法
- 統合評価システムの開発と評価(全グループの共同)

(4)「立命館大学」グループ

①研究参加者

氏名	所属	役職	参加時期
富山 宏之	立命館大学 理工学部 電子情報デザイン学科	教授	H22.4～H23.3

②研究項目

- メモリの低消費エネルギー化を実現するコンパイラ/RTOS 技術(名古屋大学グループ、九州大学グループと共同)
- 統合評価システムの開発と評価(全グループの共同)



## § 4 研究実施内容及び成果

### 4.1 リアルタイム OS を中心としたエネルギー/性能の実行時最適化フレームワーク (DEPS: Dynamic Energy/Performance Scaling) (名古屋大学グループ)

#### (1) 研究実施内容及び成果

組込みリアルタイムシステムの消費エネルギー/性能を動的(実行時)に最適化する **DEPS (Dynamic Energy/Performance Scaling) フレームワークを開発**した。DEPS とは我々が考案した造語であり、電源電圧と周波数を動的に変更する DVFS (Dynamic Voltage/Frequency Scaling) 技術を一般化した、新しい概念を意味する。より具体的には、従来は個別に適用されてきた以下の技術を、同時かつ統一的に適用する、リアルタイム OS を中心としたソフトウェア技術である。

- 動的に電源電圧と周波数を変動させる DVFS (Dynamic Voltage/Frequency Scaling)
- 動的に電源のオン/オフ制御を行う DPM (Dynamic Power Management)
- 動的にハードウェア構成(キャッシュサイズなど)を変化させる DHRC (Dynamic Hardware Reconfiguration)

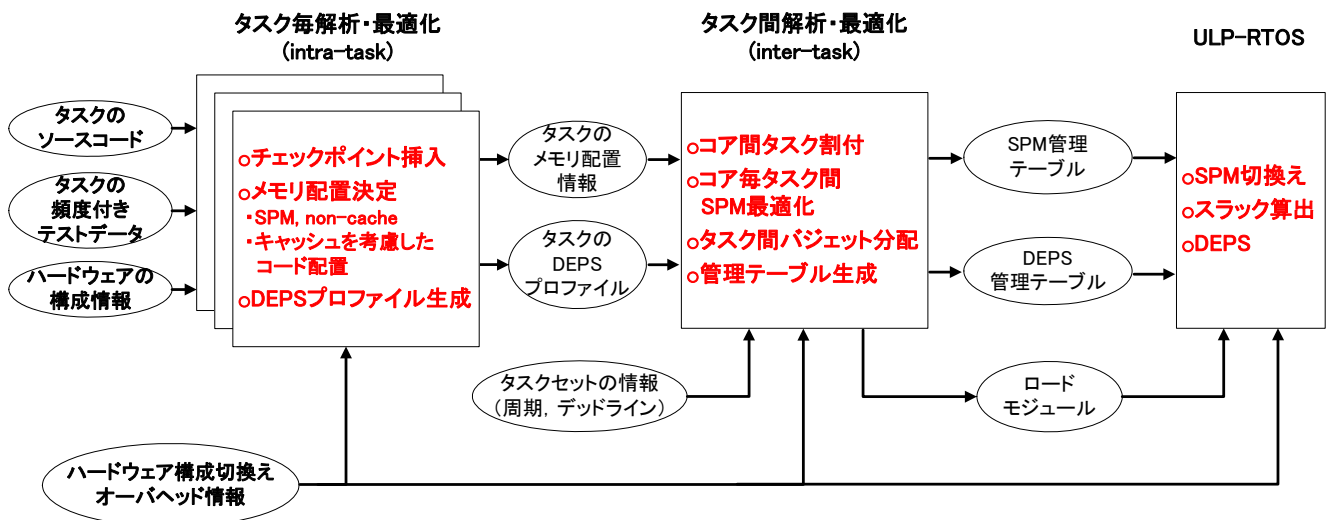
DVFS、DPM、DHRC のいずれも、その本質は消費エネルギーと性能との間に存在するトレードオフを最適化することであり、DVFS、DPM、DHRC はその実現手段に過ぎない。我々が提案した DEPS は、消費エネルギーと性能のトレードオフを最適化する方法 (DVFS、DPM、DHRC) が何であるかを問わず、消費エネルギーと性能のトレードオフを最適化する本質のみに着目した、汎用的なフレームワークである。

提案した DEPS フレームワークは、次に述べる組込みシステムを対象としている。1 つ以上の均質なプロセッサを持つ。プロセッサが複数の場合は、全プロセッサからアクセス可能な共有メモリを持つ。プロセッサは消費エネルギー/性能を実行時に変更できる仕組みを持つ (DVFS 等)。1 つ以上のタスクを持つ。タスクは周期的に実行される(あるいは既知の最小起動間隔で非周期的に実行される)。タスクはデッドライン制約を持つ。設計時にタスクはプロセッサに割り当てられ、実行時にプロセッサ間を移動(マイグレーション)しない。タスクは実行中に他のタスクと同期しない。タスクは静的に決定された優先度に従ってスケジューリングされる。

提案した DEPS フレームワークは、各タスクのデッドライン制約を満たした上で、平均消費エネルギーを最小化することを目的としている。

平成 21 年度までに、DEPS に関するスケジューリング理論の研究および予備実験を行った。例えば、シングルプロセッサを対象としたタスクスケジューリング手法[J-3][J-11]、マルチプロセッサへの拡張[I-8][C-67]、バッテリー長寿命化への拡張[J-16]などの研究成果は、論文誌あるいは権威ある国際会議に採択された。

最終年度である平成 22 年度は、DEPS フレームワークを実現するソフトウェア統合開発環境(以下、**ULP ソフトウェア開発環境**)を開発した。ULP ソフトウェア開発環境の全体像を次ページに図示する。開発した ULP ソフトウェア開発環境は 3 つのフェーズから構成され、各フェーズは複数の要素技術から構成される。



### 第1フェーズ:タスク毎解析・最適化

第1フェーズはタスク毎解析・最適化である。タスクのソースコード、タスクの頻度付きテストデータ、ハードウェアの構成情報、および、ハードウェア構成切り替えオーバーヘッド情報を入力とし、チェックポイント挿入、メモリ配置決定、DEPS プロファイル生成の3つの作業を行う。タスク毎解析・最適化の出力は、チェックポイントが挿入されたタスクのソースコード、タスクのメモリ配置情報、および、タスクの DEPS プロファイルである。

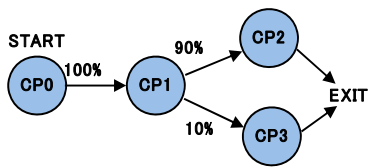
チェックポイントとはハードウェア構成を変更する可能性のある場所であり、その挿入箇所は、以下の基準に従って決定する。

- タスクの残り実行時間が大きく変わる箇所(理由:タスクの残り実行時間が短いとわかれば、HW を低速・小電力の構成に変更する。タスクの残り実行時間が長いパスを実行した時は、HW を高速・大電力の構成に変更する)
- タスクの性質が大きく変化する箇所(理由:キャッシュのサイズ/ウェイ数を変更する候補点となる)
- チェックポイントなしに実行する時間が長くなりすぎない適当な間隔(理由:動的なスラックを活用するため)

チェックポイントの挿入箇所を決定する実行トレースマイニングと呼ぶ手法を開発した。まず、頻度付きテストデータを用いてタスクを何度も実行し、大量の実行トレースを取得する。その後、実行トレース情報をもとにチェックポイント挿入箇所を決定する。実行トレースマイニングによりチェックポイントを挿入し、タスクの実行中にDVFSを行った場合、DVFSをまったく適用しない場合と比較して約60%(最大)、タスクの切替え時のみDVFSを適用した場合と比較して約18%(最大)、消費エネルギーを削減できた[C-85]。

タスクのソースコードにチェックポイントを挿入した後、タスクのコードとデータのメモリ配置を決定する。メモリ配置については § 4.2 で詳述する。

第1フェーズの最後で、DEPS プロファイルを生成する。DEPS プロファイルとは、有効な(パレート最適となり得る)ハードウェア構成の組合せが書かれたテーブルである。テーブルには、各組合せにおける最大実行時間(WCET)と平均消費エネルギー(RAEC)が書かれている。次ページに、タスクの制御フローとチェックポイント、および、そのタスクの DEPS プロファイルを例示する。



タスクの制御フローと  
チェックポイントの例

	CP0	CP1	CP2	CP3	WCET	AEC
config2	config2	config2	config1	config4	23.7	433.2
config3	config3	config3	config1	config4	28.3	345.1
config3	config4	config4	config2	config6	32.1	301.5
config4	config4	config4	config2	config6	35.2	273.8
config6	config6	config6	config4	config6	45.1	205.2

タスクのDEPSプロファイルの例

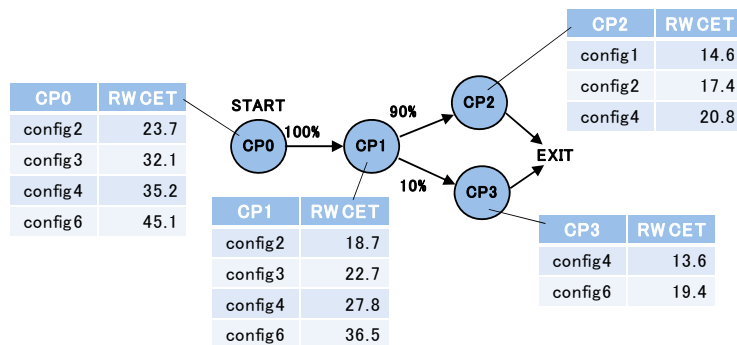
## 第2フェーズ:タスク間解析・最適化

第2フェーズはタスク間解析・最適化である。各タスクのチェックポイント挿入済みソースコード、各タスクの DEPS プロファイル、各タスクのメモリ配置情報、各タスクの周期とデッドライン、および、ハードウェアに関する情報を入力とし、タスクのコアへの割り付け、コア毎・タスク間の SPM (Scratch-Pad Memory) 最適化、コア毎・タスク間の実行バジレットの分配を行い、DEPS 管理テーブルと SPM 管理テーブルを出力する。

SPM 最適化と SPM 管理テーブル生成については § 4.2 で述べる。

タスクのコアへの割り付けとコア毎・タスク間の実行バジレットの分配は、招待講演論文[I-8]で提案した手法に基づいて行う(論文[I-8]の手法ではタスク内でコンフィギュレーションを切り替えることを想定していないので、それを拡張している)。本手法は整数線形計画法に基づいており、動的なスラック時間が発生しない状況においては最適なタスク割り付けを求めることができる。論文[I-8]の実験では、最大 86% の消費エネルギーを削減した。

第2フェーズで出力される DEPS 管理テーブルの例を下に示す。DEPS テーブルは、各タスクの各チェックポイントについて、有効な HW コンフィギュレーションと最大残り実行時間(RWCET)のリストを保持している。



チェックポイント毎のDEPS管理テーブル

## 第3フェーズ:ULP-RTOS

第3フェーズは、リアルタイム OS による実行時最適化である。

名古屋大学を中心として開発された組込みマルチプロセッサ向けリアルタイム OS 「TOPPERS/FMP カーネル」を拡張し、低消費エネルギー化のための機能を追加した。TOPPERS//FMP カーネルは、日本のデファクトスタンダードとなっている ITRON をベースとしたリアルタイム OS であり、研究代表者の高田広章が主宰する NPO 法人 TOPPERS プロジェクトより、

オープンソースソフトウェアとして公開されている。TOPPERS/FMP カーネルの完成度は高く、大手メーカーの実際の製品に搭載された実績もある。

元々の TOPPERS/FMP カーネルに対して追加した機能は、(a)スラック時間(デッドラインまでに余裕時間)を算出する機能、(b)DEPS 管理テーブルに基づいてチェックポイント毎に HW コンフィギュレーションを切り替える機能、(c)SPM 管理テーブルに基づいてタスク切り替え毎に SPM を切り替える機能である。スラック時間の算出についても、従来手法を改良し、高精度化と効率化を図った [C-78]。

DVFS 可能なプロセッサにおけるタスクスケジューリングに関する研究は数多く存在するが、本研究で開発した DEPS フレームワークは以下の点で優れている。

- DVFS だけでなく、DPM や DHRC も統合的に扱うことができる
- より現実的なシステムを想定している(利用可能な電源/周波数が有限、切り替えオーバーヘッド、性能と周波数は比例するとは限らない、など)
- 単なる理論研究でなく、実用レベルに近いリアルタイム OS を開発した

## (2)研究成果の今後期待される効果

本研究を通じて開発したリアルタイム OS は、研究期間終了後、NPO 法人 TOPPERS プロジェクトよりオープンソースソフトウェアとして公開することを予定している。これにより、日本中あるいは世界中の誰もが本研究成果を活用することができる。このリアルタイム OS は、デファクトスタンダードの ITRON 仕様をベースにしているため、産業界への技術移転もスムーズに行うことができると期待される。

## 4. 2 メモリの低消費エネルギー化を実現するコンパイラ/RTOS 技術(名古屋大学グループ、九州大学グループ、立命館大学グループ)

### (1)研究実施内容及び成果

メモリの低消費エネルギー化を実現するコンパイラ/RTOS 技術を開発した。頻繁に実行されるコードやデータを、キャッシュではなく、スラッシュパッドメモリ(SPM)に配置することで低消費エネルギー化を実現する。SPM は小容量なオンチップの SRAM であり、キャッシュに比べてアクセスあたりの消費エネルギーが小さい。

開発した SPM 活用技術を、§ 4. 1 で説明した ULP ソフトウェア開発環境に統合した。本技術は、3 つのフェーズから構成される。

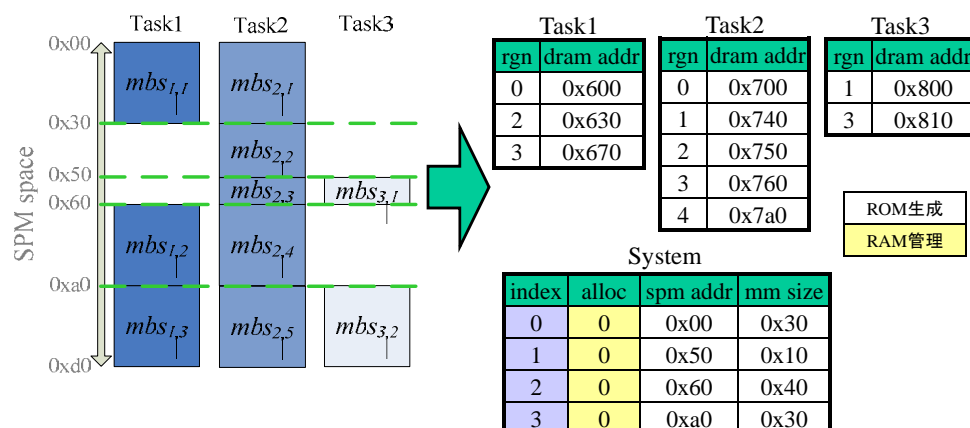
### 第 1 フェーズ:タスク毎メモリブロック割当て決定

第 1 フェーズでは、タスク毎に、SPM に割り当てるメモリブロックを決定する。この段階では、「実行時、タスク切り替えの度に SPM の全領域を切り替える」という悲観的な仮定のもとでメモリブロック割当てを決定し、第 2 フェーズ以降で是正する。メモリブロック割当ては整数線形計画法に基づいて行う。

### 第 2 フェーズ:タスク間メモリブロック配置決定と SPM 管理ブロック生成

第 2 フェーズでは、まず、タスク間での SPM 領域の実行時再配置を最小化するメモリブロックのアドレス配置を決定する(3 つのタスク間でメモリブロックの配置を行った例を左下に図示する)。配置先アドレスは静的に、かつ、一意に決定する。つまり、タスク実行中にメモリブロック配置を変更しない。メモリブロック配置を決定する際は、タスクのスケジューリング状況を解析し、SPM を切り替えるオーバーヘッドを正確に見積もる。メモリブロック配置決定も、整数線形計画法に基づいて行う。

決定したメモリブロック配置に従い、SPM 管理テーブルを生成する。SPM 管理テーブルの例を下に図示する。SPM 管理テーブルは、各タスクが有するタスク毎 SPM 管理テーブルと、システム全体の情報を管理するタスク間 SPM 管理テーブルの 2 種類存在する。



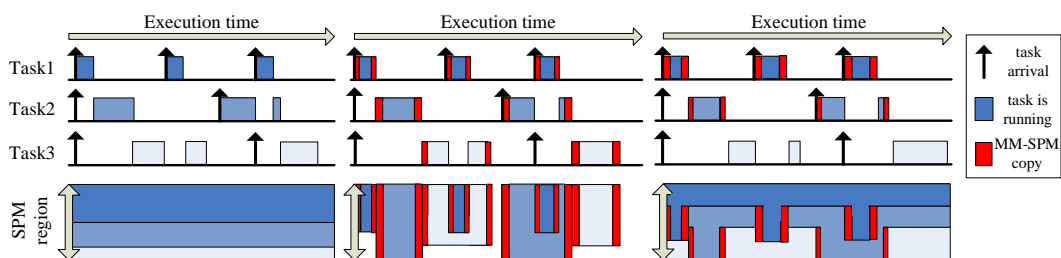
### 第 3 フェーズ:ULP RTOS による実行時再配置

実行時のタスク切り替えの際に、第 2 フェーズで生成した SPM 管理テーブルを用いて、リアルタイム OS が SPM の切り替えを行う。SPM 切り替え機構を、TOPPERS/FMP カーネルに実装した。アプ

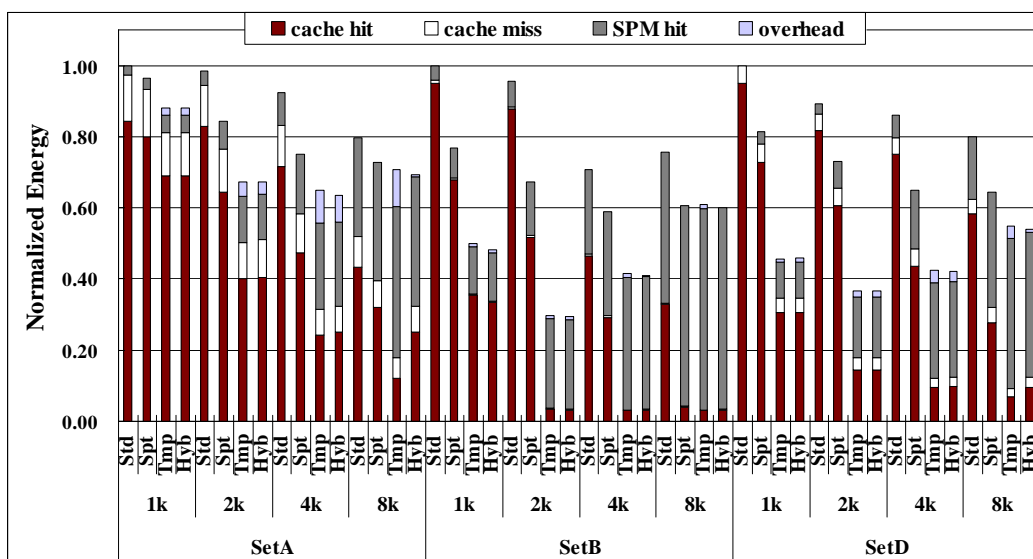
リケーションタスクの変更は不要であるため、ソフトウェア開発者に負担を強いることなく、メモリの消費エネルギーを削減することができる。

ULP ソフトウェア開発環境に実装した上記の SPM 活用技術は、個別に開発してきた複数の要素技術[J-12][C-74][C-81][C-83]を統合し、実用化に向けて改良したものである。これらの要素技術は論文誌あるいは権威ある国際会議に採択された他、多くの賞を受賞した[A-6][A-7][A-8][A-13][A-15][A-16]。これらの要素技術の中には、実装上の都合により、最終的な ULP ソフトウェア開発環境に統合されなかった(あるいは機能を制限した)ものも多数存在する。

例えば、論文[C-74]では、命令コードを対象として、タスク毎のメモリブロックの割り当てと、タスク間の SPM の割り当てを同時に最適化する手法を 3 種類(空間分割法、時間分割法、混合分割法)提案している(下図)。



左の図は空間分割法である。SPM 領域を静的に分割し、各領域をタスクに排他的に割り当てる。タスク切り替えの際に SPM を入れ替える必要がない。SPM 容量がタスクに対して相対的に大きい場合に有効である。中央の図は時間分割法である。タスク切り替えの度に SPM を入れ替える。SPM の容量が相対的に小さい場合に有効である。右の図が混合分割法である。混合分割法は空間分割法をベースにしているが、タスク切り替えの際に、低優先度のタスクの SPM 領域を、時間分割法により高優先度タスクが使用する。これにより、SPM 切り替えのオーバーヘッドを抑制しつつ、高優先度のタスクは大きな容量の SPM 領域を使用することができる。我々はこの 3 種類の手法を整数線形計画問題として形式的に定式化した。実験の結果、SPM 領域をタスクに均等に割り当てた場合と比較して、最大 73%の消費エネルギーを削減することができた。



SPM を活用する研究は世界中で多々行われているが、本研究は以下の点で優れている。

- 優先度ベース・スケジューリングを行うマルチタスク・リアルタイムシステムを対象とした研究は、本研究が世界初である
- 単なる理論研究でなく、リアルタイム OS に実装し、実用化に近いレベルまで完成度を高めた

## (2)研究成果の今後期待される効果

§ 4. 1と同様、本研究を通じて開発したリアルタイム OS は、研究期間終了後、NPO 法人 TOPPERS プロジェクトよりオープンソースソフトウェアとして公開することを予定している。これにより、日本中あるいは世界中の誰もが本研究成果を活用することができる。このリアルタイム OS は、デファクトスタンダードの ITRON 仕様をベースにしているため、産業界への技術移転もスムーズに行うことができると期待される。



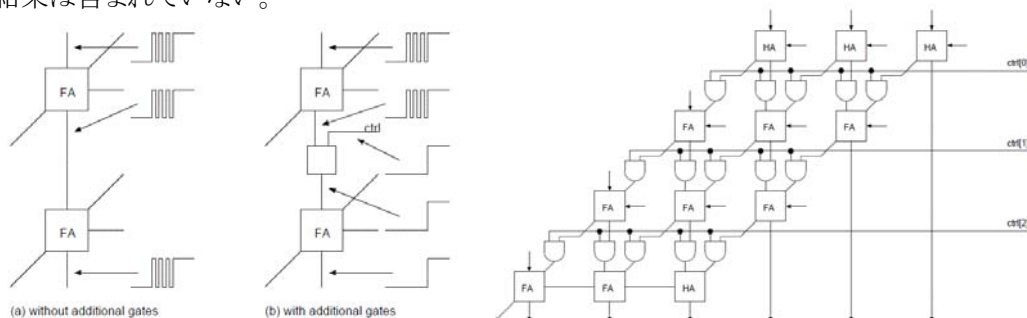
#### 4.3 演算器の消費エネルギー削減するハードウェアアルゴリズム(名古屋大学グループ)

##### (1)研究実施内容及び成果

これまでに、下記のような、整数除算の種々の実現方法を考慮したアプリケーションプログラムの消費エネルギー推定、部分加算における信号遷移回数の削減による配列型乗算器の低消費エネルギー化、オペランドの和を利用した小面積乗算器に関する研究を行った。

整数除算の種々の実現方法を考慮したアプリケーションプログラムの消費エネルギー推定：組み込み用プロセッサにおける整数除算の種々の実現方法を考慮した、アプリケーションプログラムの消費エネルギー推定手法を提案した。除算はマイクロプロセッサにおいて、加減乗算に比べ多くのサイクル数を要する演算である。また、除算の実現方法は複数存在し、その違いを考慮したアプリケーション全体の消費エネルギーの推定・比較が重要である。本推定手法では、まず整数除算の種々の実現方法において、除算一演算分の消費エネルギーをシミュレーションにより求める。その後、その結果を用いて、整数除算の種々の実現方法を考慮したアプリケーションプログラム全体の消費エネルギーを拡張 ILP (Instruction-Level Power Analysis, 命令レベル消費エネルギー解析) 手法により推定する。拡張 ILP 手法は、従来の ILP 手法を、実行ステージに複数サイクルを要する命令にも、除算の実現方法の特徴の違いによる消費エネルギーの変化にも対応できるように拡張したものである。除算の実現方法の特徴とは「除算をプロセッサに専用回路を付加して実現するかソフトウェアによって実現するか」や、「除算実行時間がオペランドによって異なるかどうか」などである。本拡張により、それぞれの除算実現方法におけるアプリケーション全体の消費エネルギーを ILP ベースの手法により推定・比較することが可能になった。

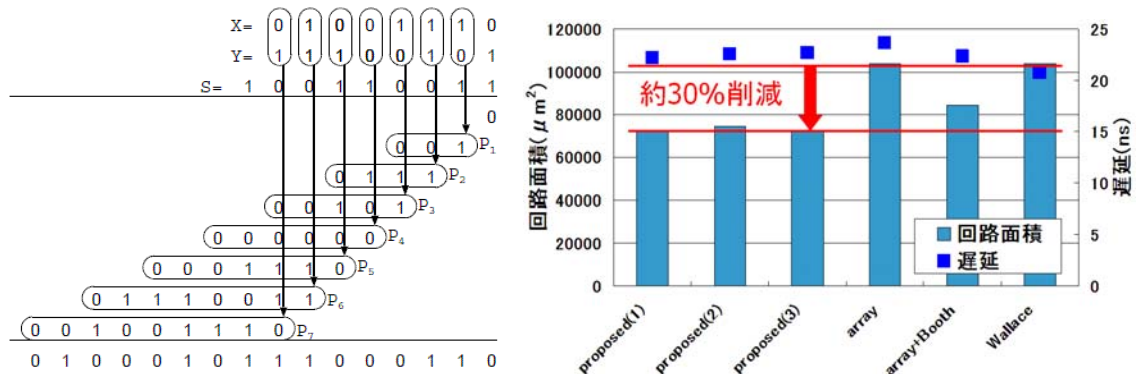
部分加算における信号遷移回数の削減による配列型乗算器の低消費エネルギー化：信号遷移回数を減らすことによって配列型乗算器の消費エネルギーを削減する手法を提案した。CMOS 回路では、信号が遷移する時に多くのエネルギーが消費される。演算を行ううえで有意な信号遷移は1演算あたり高々1回であり、2回以上の遷移は無駄なエネルギーを消費する原因となっている。乗算器の各信号では、信号遷移の伝搬が原因となり1回の演算で複数の信号遷移が起こっている。本手法は、演算過程において信号の値を一定期間固定することで無駄な遷移を削減するもので、不要な信号の伝搬を防ぐ。シミュレーションを行い消費エネルギーを見積もったところ、不要な信号の伝搬を防ぐことにより、32ビット配列型乗算器の消費エネルギーが約33%削減されるという結果がえられた。また、演算過程における信号の固定順序を変えたところ、消費エネルギーがさらに約20%削減できるという結果がえられた。研究実施後に、同じ手法に関する特許の存在が明らかになった。このため、最終的な演算器の消費エネルギー削減結果に、前述の乗算器のエネルギー削減結果は含まれていない。



オペランドの和を利用した小面積乗算器：並列乗算の最初のステップで生成される部分積のビット数を削減する手法を提案した。本手法では、乗数、被乗数のビットの値の組合せにより4つの場合を考え、1つを選択することで CPP (本手法におけるビット数を削減した部分積) の値を決定する。オペランドの和を利用することにより、CPP の総ビット数は通常の部分積の総ビット数の約半分となる。本手法は、符号なし乗算、符号つき乗算の両方に適用でき、本手法を用いた乗算器は従来の配列型乗算器やWallace 乗算器より約30%小面積であった。また、2ビット Booth の手法を



用いた乗算器より約10%小面積であった。このように本乗算器は、従来設計の代表的な乗算器と比較して、レイアウト面積の削減などが実現できており、リーク電力を少なくする目的で、とにかく小面積な乗算器が必要な状況での活用が期待される。



## (2)研究成果の今後期待される効果

オペランドの和を用いた乗算アルゴリズムは、他のハードウェアアルゴリズムとの組合せについては未検討である。オペランドの和を用いた乗算アルゴリズムのもつ特徴から、2点の改善が期待できる。1つは小面積性についての改善である。他の少ない素子数で乗算器を構成できるハードウェアアルゴリズムとの併用によりさらに小面積化が可能であると考えられる。もう1つは乗算以外の付加演算能力である。オペランドの和を用いた乗算器は、乗算器内部で加算を行っており、オペランドの積と和を並列に計算する能力を有している。他のハードウェアアルゴリズムとの併用により、より多くの付加演算のバリエーションが期待できる。また、この演算能力の活用法についても検討の価値があると考えられる。

#### 4. 4 消費エネルギーに関するアルゴリズムの基礎理論(名古屋大学グループ)

##### (1)研究実施内容及び成果

従来、アルゴリズムの良し悪しは主に時間計算量(計算時間)と空間計算量(メモリ使用量)により評価されてきた。本研究では、アルゴリズムの良し悪しを消費エネルギーの観点で評価するための理論的土台を築いた。

以下の前提で議論した。プロセッサは電源電圧とクロック周波数を連続的かつ上限なく変更することができる。電源電圧/クロック周波数を切り替えるオーバーヘッドはゼロである。電源電圧とクロック周波数は比例する。プログラムの実行サイクル数はクロック周波数に依存しない。消費電力は電源電圧の3乗に、消費エネルギーは電源電圧の2乗に比例する。プログラムはデッドライン制約を持つ。プログラムの実行時間は一定でなく、入力データに依存する。

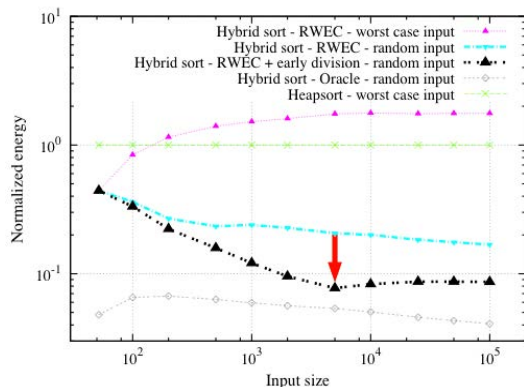
プログラムの実行中に電圧/周波数を変更しない静的電圧制御(SVS)の場合、デッドライン制約を満たし、平均消費エネルギーが最小となるアルゴリズムは、

$$(\text{最悪実行サイクル数})^2 \times \text{平均実行サイクル数}$$

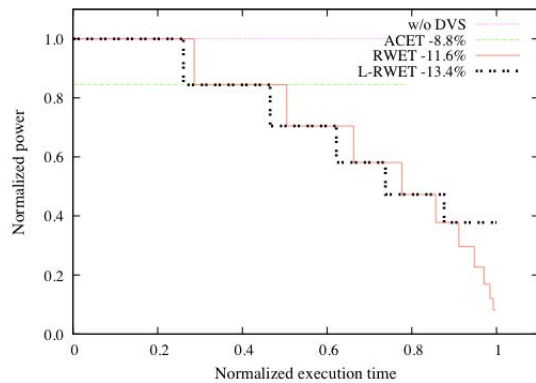
が最小となることを証明した[C-57]。つまり、理想的なDVFSプロセッサにおいては、「(最悪実行サイクル数)<sup>2</sup>×平均実行サイクル数」が、SVSアルゴリズムの消費エネルギーを評価する指標となる。理想的な仮定の下では、デッドラインを具体的に定めなくても消費エネルギーの大小を比較することが示されたのである。

また、上記の成果を元に、プログラムの実行中に電圧/周波数を変更可能な場合について、消費エネルギーの小さな整列アルゴリズムの研究を行った。種々の整列アルゴリズムの消費エネルギーを評価し、入力サイズがある程度以上大きい場合、ヒープソートは最悪消費エネルギーが小さく、混合整列法(最初は平均実行時間が短いクイックソートを用い、実行時間が長くなりそうであると判明したら、途中で最悪実行時間が抑えられるヒープソートを用いる)は平均消費エネルギーが小さくなる傾向であることを示した[C-57](次ページ左上の図)。

次に、DVSシステムにおいて動的エネルギー消費のより少ないプログラムを開発する方法論の1つを提示した。DVSシステムで実行されるプログラムに動的エネルギー消費の最適化が有効であるためには、残り予測実行時間の実行早期の正確な見積りを容易にすることが重要である。そのため、プログラムは何を計算するかに加えてどう計算するかをうまく指定する必要がある。しかし、プログラム改変によるエネルギー消費の最適化はプログラムのモジュール性を著しく損なう。われわれは、プログラムから独立し、エネルギー消費の最適化戦略を開発する手法を提案した[J-10]。提案手法により、エネルギー消費の最適化を行うときに元のプログラムの部分正当性が容易に保存され、元のプログラムおよびエネルギー消費の最適化を行うための評価戦略が独立してそれぞれモジュール性を有するようになった。遅延評価などのプログラミング言語の特徴や構成的アルゴリズム論における組化などを活用することにより半自動化が実現できたことが、本開発法の特徴の1つである。



整列アルゴリズムの正規化消費エネルギーの比較



Rabin-Karp 文字列検索アルゴリズムの消費電力

さらに、クイックソートと Rabin-Karp 文字列検索アルゴリズムについて、低消費エネルギー化のための改良を行った[J-10]。通常のクイックソートではスタックの使用量を抑えるために、小さい方から分割する(下図・左)。しかし、消費エネルギーの点では、実行早期に残り最悪実行時間が減少しないため、DVFS が有効に働かない。そこで、大きい方から分割する早期分割法を提案した(同・右)。早期分割法は実行早期に残り最悪実行時間が大幅に減少するため DVFS が有効に働き、消費エネルギーが小さくなる(上図・左)。同様の改良を Rabin-Karp 文字列検索アルゴリズムにも適用した。実験の結果、10%以上の消費エネルギー削減が得られた(上図・右)。

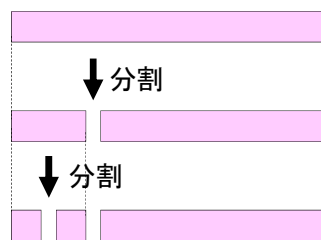


図. 小さいほうからの分割法 (通常の実装法)

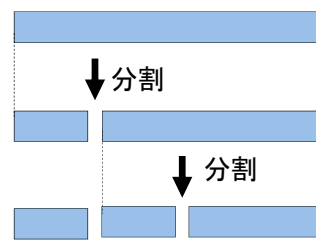


図. 大きいほうからの分割法 (提案手法)

これまで、複数の電力モードを持つシステムを省エネルギーに制御する様々なアルゴリズムが提案されてきた。しかし、我々の知る範囲において、デッドライン制約を持つ場合の具体的なアルゴリズムに対応する消費エネルギーに関する理論研究は他に例がない。本研究を通じて、我々は、平均/最悪実行時間が最適化されたアルゴリズムと平均/最悪消費エネルギーが最適化されたアルゴリズムは異なることを明らかにし、平均/最悪消費エネルギーが小さなアルゴリズムを開発するための理論的基盤を築いた。

## (2)研究成果の今後期待される効果

DEPS は、アプリケーション開発者にエネルギーと性能の関係を開示し、具体的なアーキテクチャを隠蔽する。したがって、DEPS システム上では、開示された情報を元に、使用されるアルゴリズムの消費エネルギー最適化を議論することが可能である。DEPS システム自体の研究とそのシステム上で消費エネルギー最適化が効きやすいアルゴリズムの研究は相補的な関係にある。本研究は、様々な省エネルギー技術が統合された DEPS システムにおいて、今後展開されるであろう低消費エネルギー・アルゴリズムに関する研究の、重要な基盤となると期待される。

#### 4. 5 マルチパフォーマンスプロセッサ(九州大学グループ)

##### (1)研究実施内容及び成果

###### ① 成果の位置づけや類似研究との比較

携帯型情報端末の市場拡大を背景にコンピュータシステムの省エネルギー化が非常に重要な課題となっている。2000年には斬新な省電力プロセッサとして動的に変圧プロセッサ(以下DVSプロセッサ)が製品化されたが、テストコストの問題やタイミング設計にかかる手間、オンチップDC-DCコンバータのコスト、電圧・周波数切替えのオーバーヘッドおよびこれに伴うリアルタイム性保証の困難さなどの理由から、DVSプロセッサがリアルタイム組込みシステムで利用された例は少ない。DVS制御を行う電力管理ソフトウェアの例として、SpeedStep(Intel)、PowerNow!(AMD)、LongRun(Transmeta)、SmartReflex(TI)、IEM(ARM)、XEC(Freescale Semiconductor)などがある。しかし、上記の電力管理ソフトウェアは何れもリアルタイムシステムを主なターゲットとはしていない。リアルタイムシステムとは、決められた制約時間内に与えられたタスクを完了するコンピュータシステムのことを意味する。あるいは、厳密に時間制約を守ることができなくても、タスクの最悪実行時間が制約時間よりも小さくなるように設計されたシステムのことを指す。例えばTransmeta社のLongRunは、プロセッサのサイクルあたりの実行命令数(IPC)の値を実行時に観測し、IPC値が小さい場合は、プロセッサに高い性能が求められていないと判断し、電圧を下げて省エネルギー化を図る。逆にIPC値が大きい場合は、プロセッサに高い性能が求められていると判断し電圧を挙げて高速化する。この方法はリアルタイム性を必ずしも保障しないため、リアルタイムシステムには適用が難しい。

本研究テーマでは、DVSプロセッサに代わる省エネルギープロセッサとしてマルチパフォーマンスプロセッサを開発し、その実用可能性についてポストレイアウトシミュレーションにより評価した。マルチパフォーマンスプロセッサは既存のDVSプロセッサと比較して性能切り替えのオーバーヘッドを約2桁小さくすることを可能にした。またDVSプロセッサと比較して低電圧動作時の電力効率が30%以上良いことをHSPICEによる解析で確認した。

###### ② 実施方法・実施内容

以下にマルチパフォーマンスプロセッサの詳細を述べる。マルチパフォーマンスプロセッサは同じ命令セットアーキテクチャを持つ複数の演算要素(以下PE)と動的に実効容量を変更できるキャッシュメモリとローカルメモリから構成される。各PEは異なる消費エネルギーと動作性能を持つ。一般には低消費エネルギーと高速動作はトレードオフの関係にあるため、高速で消費エネルギーの大きいPEと低速で消費エネルギーの小さいPEをCPUコア内に複数搭載し、状況に応じて適切なPEを稼働させる。同時には一つのPEのみが稼働する。使用しないPEへは信号供給とクロック供給を遮断する。キャッシュメモリやローカルメモリはPE間で共有する。ただし、同時に一つのPEのみが動作するため、メモリの入出力ポートはそれぞれ1ポートのみである。つまり、コストの大きいマルチポートメモリは不要である。タスクレベルの並列性が要求されるアプリケーションにはマルチパフォーマンスプロセッサコアを並列にバス接続することにより並列化する(図5.1参照)。

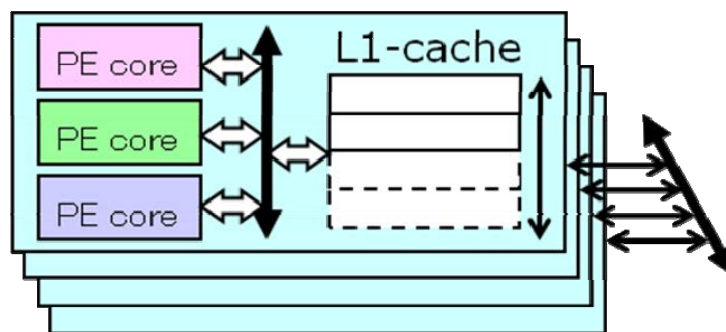


図 5.1 マルチパフォーマンスプロセッサの構成例

東芝社製の32/16ビット可変命令語長RISCプロセッサMePをベースにマルチパフォーマンス

プロセッサのプロトタイプを設計した。プロセステクノロジーは商用の 90nmCMOS プロセスを使用した。CPU コア部分は同一の回路を複製し、性能と消費エネルギーの異なる PE コアとして実現する。各 PE コアは RT レベルでは同一の回路記述であるが別々の電源電圧を使用してキャラクタライズした標準ライブラリを用いてそれぞれ設計する。それぞれ 1.0V、0.68V、0.52V を使って 3 種類の PE コアを設計した。MPU コア 0 には 3 種類の PE コアを、MPU コア 1 には 1.0V と 0.68V を使用する PE コアを、MPU コア 2 には 1.0V と 0.52V を使用する PE コアをそれぞれ搭載した。PE コアの切り替えはスクラッチパッドメモリの特定のアドレスに特定の値を書き込むことにより行う。メモリは 1.0V を使用し、すべての PE コアで共有する。ただし、PE は同時に一つしか稼働しないためメモリはすべてシングルポートで実現する。

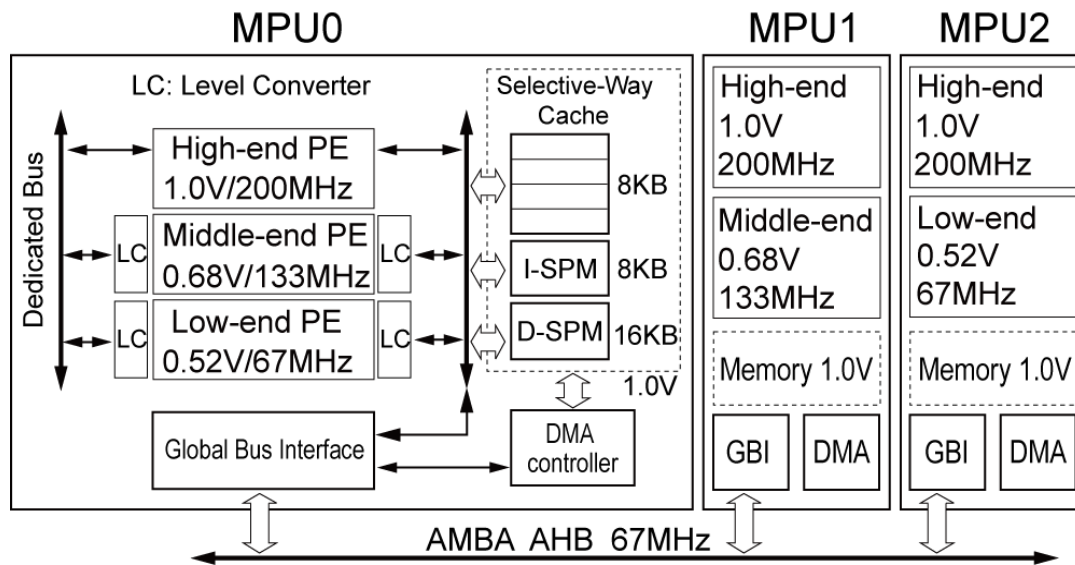


図 5.2 MeP をベースとしたマルチパフォーマンスプロセッサのプロトタイプ

オンチップメモリは 8K バイト 4 ウェイセットアソシアティブの命令キャッシュと 8K バイトの命令コード用スクラッチパッドメモリおよび 16K バイトのデータ用スクラッチパッドメモリを搭載する。命令キャッシュは 4 ウェイのうち、稼働させるキャッシュウェイを選択することが出来る。キャッシュウェイの選択はデータ用スクラッチパッドメモリの特定のアドレスに特定の値を書き込むことにより行う。つまり、プログラムの動作状況に応じてどのキャッシュウェイを使うかをプログラマが指定出来る。また、使用する PE を切替える際には、PE の内部状態(レジスタの値)を現在稼働中の PE から次に稼働する PE へ移動させる。汎用レジスタの値はローカルメモリを経由して移動させ、その他のレジスタの値は直接専用バスで移動させる(図 5.2 参照)。

### ③ 研究の成果

図 5.2 に示したプロセッサを設計し、PE コアの動的切り替えとキャッシュウェイの動的選択機能を含むすべての機能が正常に動作することをポストレイアウトシミュレーションにより確認した。

各 PE コアは使用するそれぞれの電源電圧に最適化されるため、従来型の DVS プロセッサよりも低電圧動作時のエネルギー効率が良い。図 5.3 は、1.0V で最適設計した PE コアの最長パスを抽出し、そのパスに対する電圧・遅延特性を SYNOPSIS 社の HSPICE で計測した結果である。最長パスは SYNOPSIS 社の静的遅延解析ツール(STA)で抽出した。1.0V 動作時には 5ns のパス遅延は、0.68V の電源電圧では 12.5ns まで増大する。この場合、プロセッサの動作周波数は 80MHz になる。筆者らの設計したマルチパフォーマンスプロセッサ(MPP)は 0.68V で 133MHz を実現する。さらに 0.52V まで電圧を下げるとパス遅延は 37ns になり、プロセッサは 27MHz でしか動作しない。つまり、従来型の DVS プロセッサは、0.52V では 27MHz でしか動作しない。多入力論理セルと高



い閾値電圧を使用する論理セルは低電圧動作時に遅延が急激に増大するためだと考えられる。また、クロックのソースから各フリップフロップへクロック信号が到達する時刻のばらつき(クロックスキューとして現れる)は電源電圧によって異なるため、1.0V で最適設計したプロセッサを 0.52V で動作させる場合にはクロックスキューによりプロセッサが正常動作しない可能性もある。上述したとおりマルチパフォーマンスプロセッサはそれぞれの電圧に最適化して設計しているため、低電圧動作時でも回路のクリティカルパス遅延はそれほど大きくならない。さらには、クロックツリーの設計も特定の電圧に特化して行うため、従来型の DVS プロセッサの設計と比較してクロックスキューの削減にかかる手間を小さく抑えられる。結果的に、マルチパフォーマンスプロセッサは従来の DVS プロセッサと比較すると、133MHz 動作では 31%、67MHz 動作では 34%エネルギー効率が良いことを確認した(図 5.3 参照)。

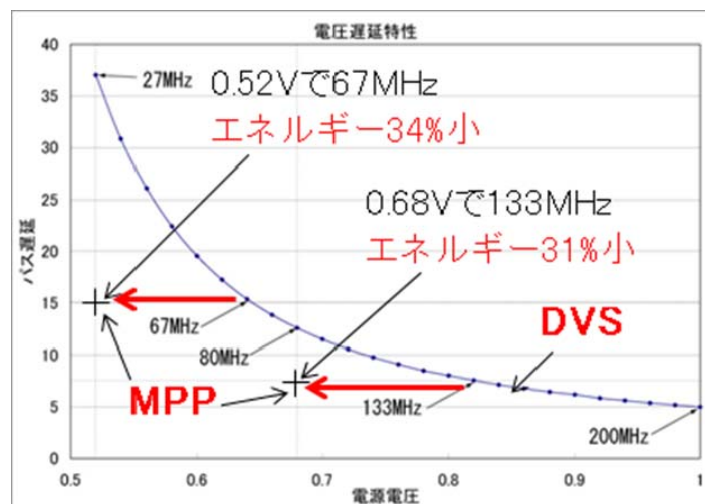


図 5.3 低電圧動作時のエネルギー効率

表 5.1 に PE コア切り替え時とキャッシュウェイ選択時のオーバーヘッドを示した。DVS プロセッサと比較すると切り替え時間、消費エネルギー共に約 2 桁小さくなることを確認した。命令キャッシュのウェイ選択も同程度のオーバーヘッドで実現可能であることを確認した。

Direction	Transition Time [ns]	Energy Consumption [nJ]
High speed → Middle speed	1,113	11.84
High speed → Low speed	1,290	11.25
Middle speed → High speed	968	13.04
Middle speed → Low speed	1,443	9.27
Low speed → High speed	1,205	13.39
Low speed → Middle speed	1,286	8.93
Cache Way Selection	690	10.35

東京大学 VDEC のチップ試作サービスを通じて、2008 年度に 65nm プロセスで単一電源電圧のシングルコア・マルチパフォーマンスプロセッサを試作し、簡単なベンチマークプログラムを用いたテストを行った結果、プロセッサの基本機能の正常動作と命令キャッシュのウェイ選択機能の正常動作を確認した。2009 年度に 0.18um プロセスを用いて試作した多電源電圧使用のマルチコア型マルチパフォーマンスプロセッサは電源の配線ミスにより正常動作が確認できなかった。2010 年度に、同チップ配線ミスを修正し、再試作を行った結果、単純なベンチマークプログラムの正常実行と DEPS 機能の正常動作を確認した。一部レベルシフト回路と論理素子との接続ミスにより、漏れ電流が発生する問題が生じているが、DEPS 機能の動作は正常であることを確認した。

## (2)研究成果の今後期待される効果

Intel 社が 80 コアを搭載するプロセッサを試作し、ハードウェアの実装技術の観点ではメニーコアの時代に突入した。しかし、80 の並列性が存在するアプリケーションは現状では少ない。一方で、組み込みコンピュータシステムに必要とされる機能は多様化し、同じアプリケーションでも、高い速度性能を要求する場面と低速動作が可能な場面が混在する。ピーク性能を維持したまま消費エネルギーを低減するためには、性能と消費エネルギーの観点からヘテロジニアスな PE を状況に応じて瞬時に切替えて使用するアーキテクチャが有効である。本研究で開発したマルチパフォーマンスプロセッサは DVS プロセッサと比較して性能の切り替えオーバーヘッドが約 2 桁小さい。また、低電圧動作時のエネルギー効率は DVS プロセッサより 30%以上良い。チップ面積をスレッドレベル並列化やタスクレベル並列化に使うだけでなく、「多性能化」に使用することにより大幅な省エネルギー化が実現できる可能性があることを確認した。マルチパフォーマンスプロセッサの応用に関しては名古屋大学グループの DEPS の報告に詳しく述べられている。詳細は DEPS の報告を参照されたい。

#### 4.6 データメモリの省エネルギー技術(九州大学グループ)

##### (1)研究実施内容及び成果

今日製品化されているほとんどの組み込みプロセッサは CPU コアと同じチップ上にキャッシュメモリとスクラッチパッドメモリを搭載している。スクラッチパッドメモリはキャッシュメモリと比較して消費電力が小さいため、頻りにアクセスされるプログラムコードやデータオブジェクトをスクラッチパッドに配置することによりプロセッサシステムの平均消費電力を低減できる。また、マルチコアプロセッサ(あるいはメニーコアプロセッサ)において同一チップ上に搭載されるプロセッサコア数が増大すると、コヒーレントキャッシュメモリではコヒーレンシ制御のための消費エネルギーが膨大になるため、もともと明示的なメモリ管理を行うスクラッチパッドメモリの方が消費電力の観点でメリットがさらに大きくなる。コヒーレンシ制御とは、マルチコアプロセッサにおいて、それぞれのプロセッサが共有データのコピーを持つ時、その複数のコピーデータ間で一貫性を常に保っておくためのデータ制御のことである。

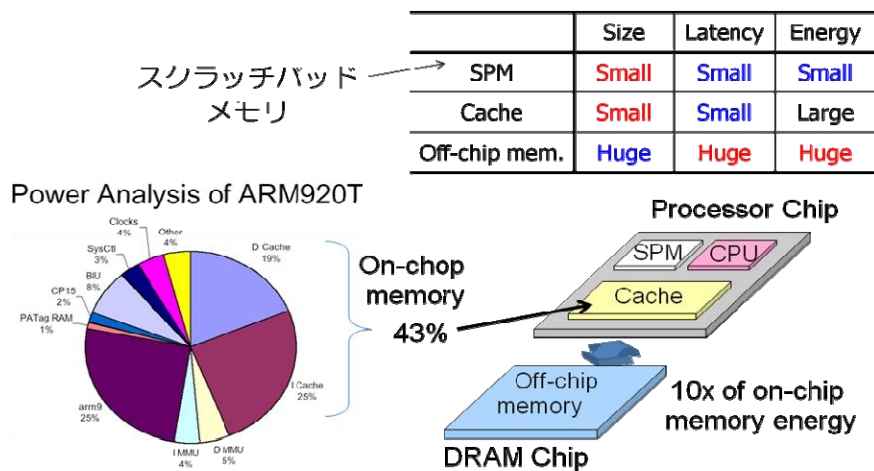


図 6.1 組み込みプロセッサの構成例

本研究では、マルチコアプロセッサにおいてそれぞれのプロセッサコアがローカルに搭載するスクラッチパッドメモリにデータオブジェクトを最適に配置しデータアクセスに伴う消費エネルギーを大幅に削減する手法を開発した。データオブジェクトの中でも特にスタックデータをスクラッチパッドメモリへ最適に配置する仕組みについて研究した。スタックデータはデータオブジェクトの中でも最もアクセスの多いオブジェクトであるため、スタックをスクラッチパッドメモリに配置できれば大幅な電力削減が見込まれる。しかし、スクラッチパッドメモリ(一般に SRAM)はオフチップメモリ(一般に DRAM)と比較してビット単価が高く、多くの組み込みプロセッサでは高々 16K バイト程度のスクラッチパッドメモリしか搭載しない。従って実用的なアプリケーションプログラムではスタック全体をスクラッチパッドに配置することは難しい。そこで、本研究では、関数がコールされる際にスタックフレームをスクラッチパッドと主記憶間で動的に入れ替える仕組みを開発した。スタックフレームの配置はコンパイル時に静的に決定する。スタックフレームの配置最適化の手順は以下のとおりである。

- スタックフレームのアクセス頻度を命令セットシミュレータを用いたプロファイリングにより解析
  - 最もアクセス頻度の高いスタックフレームを SPM に配置
  - 各スタックフレームの配置(スクラッチパッドメモリかメインメモリ)を決定してスタックアクセスに伴う消費エネルギーを最小化する問題を整数計画問題として定式化
  - 関数コールの前後でスタックフレームをメインメモリに Store/Load することが前提
  - 上記 Store/Load のエネルギーを最適化問題の中でコストとして考慮
  - 上記の整数計画問題を解くことによってコンパイル時にスタックフレームの配置を最適に決定
- 上記のスタック配置最適化手法を東芝社製の MeP プロセッサとその命令セットシミュレータを用いて評価した。比較のために以下の2つの既存手法(Cyclic と Static)と合わせて評価を行った。



- Cyclic: SPM が一杯になったら古いフレームから順に追い出す
- Static: スタックフレームをプログラム実行時には移動させない
- Ours: スタックの動的入れ替えを前提として、スタック配置を前項で述べた方法で最適化する

8つのベンチマークプログラム(AES, DES, mpeg2dec, MD5, rad2deg, patricia, Cubic, FFT)を用いた評価結果を図 6.2 に示す。グラフのエネルギー値(縦軸)はすべてのスタックフレームをメインメモリに配置した時の消費エネルギーで正規化されている。横軸はスクラッチパッドメモリ(abyte)を示している。すべてをメインメモリに配置した場合と比較するとスタックアクセスに伴うエネルギーを 90%以上削減できることを確認した。

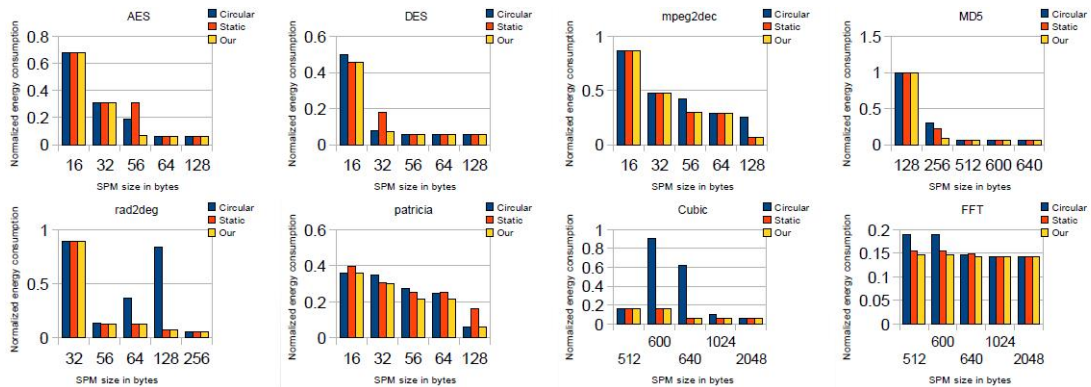


図 6.2 スタックフレーム最適配置結果

上記の結果はシングルタスクの結果であるが、マルチタスク環境ではタスク間でメモリ領域を適切にシェアすることによってスクラッチパッドメモリのアドレス空間を有効に利用できる。図 6.3 にスクラッチパッドの領域をタスク間でシェアする際の3通りの方法を示した。(a)は空間的な共有方法、(b)は時間的な共有方法、(c)はスクラッチパッドメモリを空間的共有部と時間的共有部に分けて(a)と(b)を適用する方法を示している。

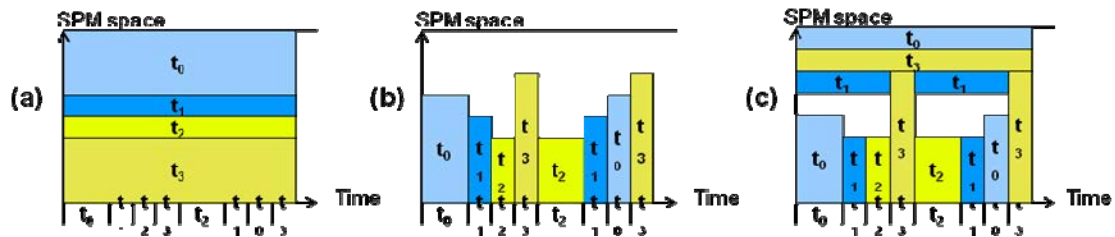


図 6.3 スクラッチパッドメモリの領域共有法

また、スクラッチパッドメモリ内に配置するデータオブジェクトの配置位置を最適化することにより、タスクスイッチの際にスクラッチパッドメモリとメインメモリ間で転送するデータ量を大幅に削減できる。例えば、図 6.4 に示すように、タスク  $t_0$  と  $t_1$  が切り替わる際に  $t_0$  が使用するメモリブロック  $\beta_0$  とタスク  $t_1$  が使用するメモリブロック  $\beta_1$  がアドレス上で大幅にオーバーラップしていると、スクラッチパッドメモリ内の多くのデータを入れ替える必要がある。そこで、本研究では、マルチタスク環境において、スクラッチパッドメモリ内に配置するデータオブジェクトの配置位置を最適化し、タスク切り替えの際にスクラッチパッドメモリとメインメモリの間で転送するデータ量を最小化する手法を開発した。

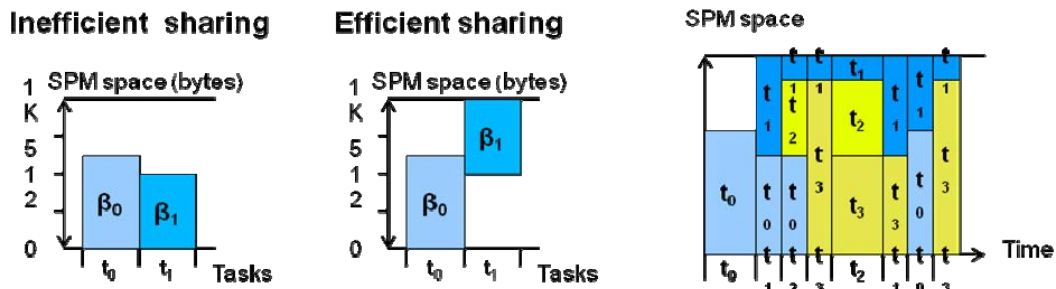


図 6.4 データ転送を最小化するデータ配置

東芝社製の MeP プロセッサとその命令セットシミュレータを用いてスラッシュパッドメモリへのデータ配置手法を評価した。配置対象とするデータオブジェクトはグローバル変数や定数データなどの静的変数とスタックフレームである。MiBench ベンチマークスイートから無作為に複数のアプリケーションプログラムを選択し、6種類のマルチタスクベンチマークプログラム(図 6.5 に示した set0~set5)を人工的に作成した。実験結果を図 6.5 に示す。図中の Space、Time、Hybrid はそれぞれ、図 6.3 の(a)、(b)、(c)に示す手法を適用した結果に対応している。“Task”は図 6.4 の右端に示した手法に対応する。図 6.4 に示した“Task”は各タスクが使用するデータオブジェクトに対して連続した1つの領域を割り当てるのに対して、“Block”では各タスクが使用するデータオブジェクトがスラッシュパッドメモリ内の複数の不連続なアドレス領域を使用する。

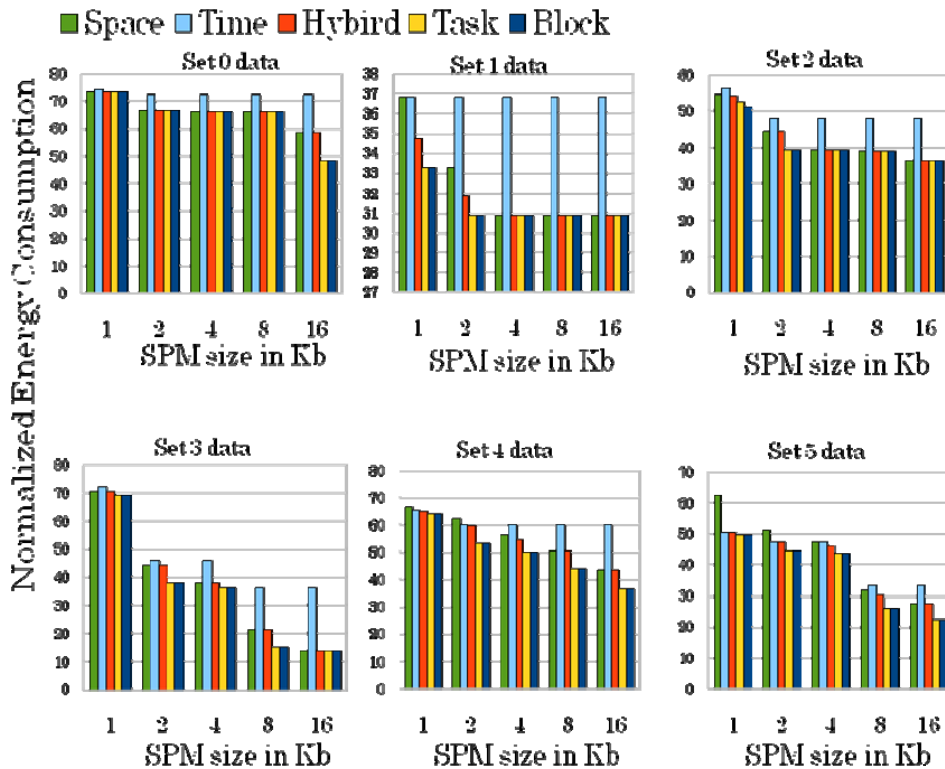


図 6.5 スラッシュパッドメモリへのデータ配置最適化の結果

グラフの縦軸はすべてのデータオブジェクトをメインメモリに配置した時の消費エネルギーで正規化した結果を示している。最も結果が良いケースでは、すべてのデータをメインメモリに配置した場合と比較して最大で消費エネルギーを 85%削減できることを確認した。

## (2)研究成果の今後期待される効果

チップに搭載されるプロセッサコア数が増大するにつれ、データコヒーレンシの観点からデータキャッシュメモリをオンチップメモリとして利用することが難しくなる。従って将来的にはスクラッチパッドメモリの存在価値がますます高くなることが予想される。特にデータ用のスクラッチパッドメモリを有効に利用する技術は性能と消費エネルギーの両方の観点で非常に重要な技術となる。本研究では、スタックフレームをスクラッチパッドメモリに配置する技術の開発に取り組んだが、実用化にはまだまだ多くの問題が残されている。今後は実用化へ向けた技術開発に取り組む予定である。

#### 4. 7 プロセッサシステムの電力キャラクタライズと電力見積もりツール(九州大学グループ)

##### (1)研究実施内容及び成果

###### ① 成果の位置づけや類似研究との比較

携帯型情報機器の普及と共に組込みシステムの低消費電力化に対する要求はますます強まっている。しかし、組込みソフトウェア設計の分野では低消費電力化に対する意識はそれほど高くない。デジタルシステムの電力見積もり手法は今日までに数多く提案されている。ソフトウェアの振る舞いが消費エネルギーに与える影響を見積もる最も正確かつ高速な方法は実チップの電力を測定することである。しかし、電力測定では、数 $\mu$ 秒オーダーの間に消費されるエネルギーを見積もることは難しいため、プロセッサ内部のホットスポット解析やアプリケーションプログラムのホットスポット解析が難しい。一方、ハードウェア設計初期段階の見積もり手法の多くはゲートレベルや RTL のシミュレーションをベースとしているため、大規模ソフトウェアの解析には膨大な時間を要する。また、RTLシミュレータなどのハードウェアシミュレータの使用はソフトウェア開発者にとって敷居が高い方法であると考えられる。命令セットシミュレータ(以下 ISS)を用いた電力見積もり手法も数多く提案されているが、電力見積もり機能を持つ ISS は一部のプロセッサに限られているため、この機能のサポートが無い場合は、専用のシミュレータと電力モデルを開発する必要がある。本研究で開発した手法は、対象とするプロセッサシステムに対して半自動で消費エネルギーのキャラクタライズを行うことを特徴とする。消費エネルギーのキャラクタライズにより生成された線形モデルは、専用開発されたサイクル精度 ISS や GNU のソフトウェアデバッガと組み合わせて使用することができる。

###### ② 実施方法・実施内容・成果

本研究では組込みソフトウェア分野の技術者が設計段階で消費エネルギーを意識できる環境の提供を目標として、命令セットシミュレータ(以下 ISS)を用いてプロセッサシステムの消費エネルギーを解析する研究に取り組んだ。アプリケーションソフトウェアから観測しやすい各種パラメータを用いてプロセッサの消費電力を半自動でモデル化するキャラクタライズ手法を開発した。パラメータには ISS から容易に抽出でき、かつ消費エネルギーに影響を与えるものを選択することが重要である。多くのプロセッサシステムでは、メモリアクセスに要するエネルギーが非常に大きな割合を占めるため、キャッシュメモリやスクラッチパッドメモリ、外部メモリへのアクセス回数をパラメータとして使用することはモデル精度を高める上で重要である。それ以外に CPU がストールを起こす原因をパラメータとして使用する。具体的には以下のパラメータを使用してプロセッサシステムの電力キャラクタライズを行った。

- 命令毎の実行回数
- 命令キャッシュミス回数
- データキャッシュミス回数
- ロードまたはストア命令が連続して実行される回数 (バーストアクセス)
- 分岐命令のうち分岐が不成立であった回数
- 分岐不成立と命令キャッシュミスが同時に発生する回数
- リードアフターライト (RAW) ハザードの発生回数
- データキャッシュミスと RAW ハザードが同時に発生する回数
- 乗算命令や除算命令などのマルチサイクル命令がキャッシュミスを引き起こす回数

上述のパラメータの値を独立に変化させるトレーニングベンチをパラメータ数だけ作成し、それらのトレーニングベンチを対象とするプロセッサシステム上で実行した時の消費エネルギーをポストレイアウトシミュレーションにより見積もる。この消費エネルギー値を  $E_i$  とする。ここで  $i$  はトレーニングベンチのインデックス番号である。同じトレーニングベンチを実行した際の上記パラメータの値を ISS により取得する。このパラメータ値を  $P_{ij}$  とする。ここで  $j$  はパラメータのインデックス番号である。ISS による消費エネルギーの見積もり値は  $E_i' = \sum c_j P_{ij}$  で見積もる。ここで、 $c_j$  はパラメータの係数である。このパラメータ係数は  $(E_i - E_i')^2$  の値を最小にするように最小二乗法を用いて回帰分析により決定する。つまり、ISS による見積もり値とポストレイアウトシミュレーションによる電力見積もり値の2乗誤差

が最小になるように  $c_j$  を求める。上述のフローにより求められる  $\sum c_j P_{ij}$  が消費エネルギーの線形近似式となる(図 7.1 左参照)。

上述のキャラクタライズフローをルネサステクノロジー社の M32R と SH3-DSP に適用したところ、モデルの平均誤差はポストレイアウトシミュレーションによる見積もり値と比較して約3%であった。このモデルの簡略版は上述のデータ配置最適化やコード配置最適化にも利用している。命令セットシミュレータ(以下 ISS)が出力するアドレストレースと上記のモデルを使用することにより、特定のプロセッサシステム上で動作するアプリケーションプログラムが消費する電力を高速に見積もることができる。ゲートレベルシミュレーションでは数時間かかる見積もりを数秒で行えることを確認した。

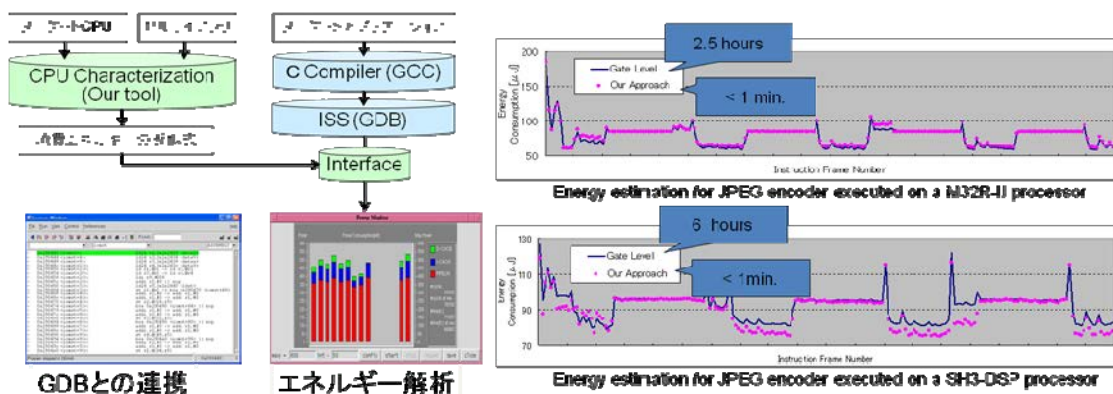


図 7.1 キャラクタライズフローと電力見積もり例

同様のキャラクタライズを上述のマルチパフォーマンスプロセッサにも適用した。マルチパフォーマンスプロセッサは東芝社の MeP をベースに  $0.18 \mu\text{m}$  プロセステクノロジーを使用して設計した。各プロセッサコア内に PE コアを 2 つ集積し、それぞれの PE コアは、1.8V と 1.0V で動作する。つまり、1.8V モードと 1.0V モードを切り替えて動作可能である。また、使用するキャッシュウェイを選択可能な 4ウェイセットアソシアティブの命令キャッシュを搭載している。

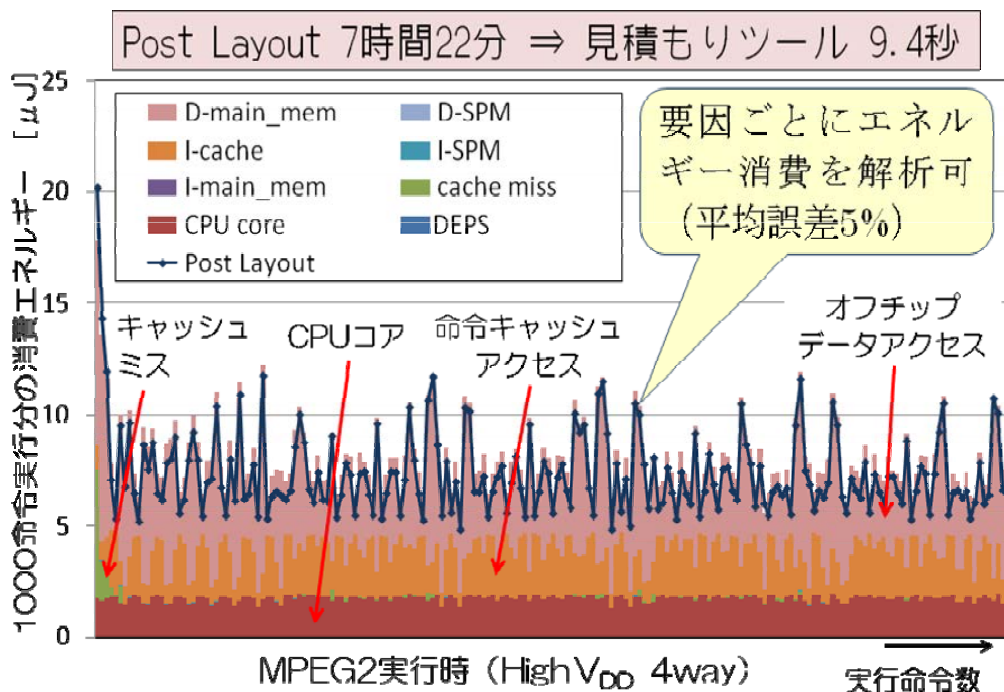


図 7.2 マルチパフォーマンスプロセッサの消費エネルギー解析結果

図7.2にマルチパフォーマンスプロセッサ上でMPEG2エンコーダを実行した際の消費エネルギーを示す。プロセッサは1.8Vの高速モードで、命令キャッシュのウェイは4つとも稼働させた。折れ線グラフがポストレイアウトシミュレーションによるエネルギー見積りの結果を示し、その下部の色塗りされた部分はISSから取得したパラメータ値と線形近似モデルにより見積もった消費エネルギー値を示している。ISSによる見積もり値はポストレイアウトシミュレーションによる見積もり値と比較して誤差が5%と非常に小さいことを確認した。また、ISSによる見積もりは、パラメータ毎の消費エネルギーを個別に計算できるため、消費エネルギー解析が容易である。例えば、オフチップメモリのアクセスで消費されたエネルギーやキャッシュアクセスによる消費エネルギーなどを個別に見積もることが可能である。これにより消費エネルギーのボトルネックを容易に特定できる。

## (2)研究成果の今後期待される効果

本研究テーマではCPUと主記憶などのメモリサブシステムを含むプロセッサシステムのみを対象として消費エネルギーのキャラクタライズを行ったが、本研究のアイデアは無線通信やストレージおよびディスプレイなどの消費エネルギーキャラクタライズへ拡張可能である。本研究で開発した手法の一部は、上記のような非LSI部品を含むさまざまな電子部品のエネルギーキャラクタライズ手法へと拡張し、民間企業との新たな共同研究として展開中である。



## 4.8 カナリアによる電源電圧マージンの最適化(九州大学グループ)

### (1)研究実施内容及び成果

正常動作保証のための設計マージンによる不要な消費エネルギーを削減する方式を開発した。

#### ①研究の背景

LSI においては、製造時のプロセス変動、動作時の電源の揺らぎや温度変動、これらのばらつきが遅延時間に影響を及ぼす。これらのパラメータが変動しても、タイミング制約を満足して LSI が正常に動作できるように、通常は右図のように電源電圧に余裕を持たせた設計が実施される。この設計マージンの設定には、上記パラメータの最悪条件を考慮する必要がある。あらゆるパラメータが最悪になる場合は非常に稀であり、上述のようにして決定された電源電圧は過大に高く設定されることになる。本来は必要の無いエネルギーを消費することになる。消費エネルギー100分の1の目標を達成するためには、あらゆる無駄なエネルギー消費を排除する必要がある、条件次第では不要な上記のエネルギーを削減することが必要である。動作状況に応じて最適な電源電圧を設定し、不要なエネルギー消費を排除出来る方式が求められる。

温度変動に  
配慮する  
マージン

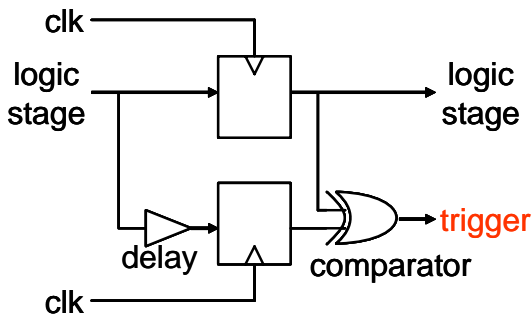
電源の  
揺らぎに  
配慮する  
マージン

プロセス  
変動に  
配慮する  
マージン

最良条件  
での電圧

#### ②実施方式

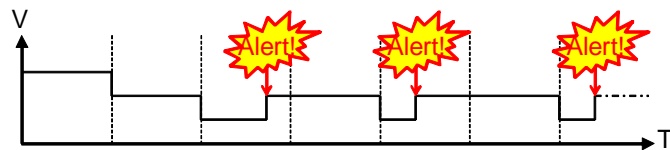
①で説明した課題を達成するためには、LSI の動作中にその遅延時間を測定し、タイミング制約を満足できる最低限の電源電圧を選択できる必要がある。遅延時間の絶対値を測定することは困難だが、上述の目的のためであれば絶対値は必要ではなく、タイミング制約を違反しないことを満足できれば十分である。上記の考察から左図に示すカナリア・フリップフロップ (FF) を考案した。



カナリア FF は通常の FF に加えて、冗長な FF、遅延素子、そして比較器で構成される。図中で上に配置される通常の FF をメイン FF、下に配置される冗長な FF をシャドウ FF と呼ぶことにする。二つの FF で同じ値を冗長に保持することになるが、シャドウ FF の手前には遅延素子があるため、前段の組み合わせ回路の出力は、メイン FF よりも遅れてシャドウ FF に到達することになる。二つの FF

の出力は比較に入力され、一致不一致が検出される。不一致はシャドウ FF が誤った値を保持していることを意味する。つまり、比較器によりシャドウ FF におけるタイミング制約違反を検出できる。これは、近い将来にメイン FF でもタイミング違反を生じる可能性があることを予測していることになる。

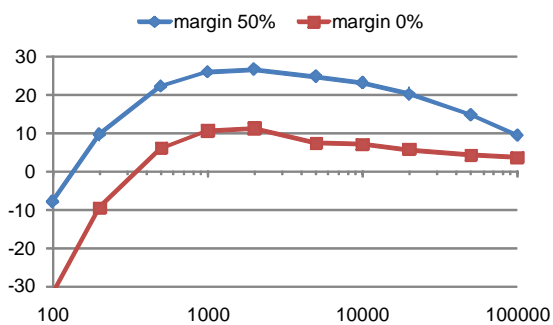
電源電圧が低くなると遅延時間が増大しタイミング違反を生じる。動的電圧制御機構 (DVS) とカナリア FF を組み合わせることで、右図のように徐々に電源電圧を下げれば、タイミング違反を起こさない最も低い電源電圧を知ることが可能になる。カナリア FF がタイミング違反を予報すると、現在よりも 1 レベル上の電源電圧に設定する。以上の方法で動作状況に応じて最適な電源電圧を設定し、不要なエネルギー消費を排除出来る。



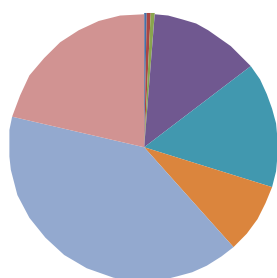
#### ③実施内容と成果

東芝グループより提供された MeP プロセッサのシミュレータ MPI を用いて、②で説明した方式を評価した。まず MPI 上でアプリケーションプログラムを実行し、実行トレースを採取する。一方、DVS とカナリア FF によるタイミング違反予測機構とを実装したシミュレータを独自に開発した。MPI で採取したトレースを後者のシミュレータで実行し、エネルギー削減効果を評価した。評価に際しては、電源電圧の変更に 100 クロックサイクルのオーバヘッドを要すると仮定している。まずシングルコア MeP で評価し、その結果を元にデュアルコア及びクアドコア MeP で評価した。

右図は、電源電圧を変更する間隔を 100～100K クロックサイクルで変えた時のエネルギー削減効果を示している。横軸が変更間隔で、縦軸がエネルギー削減割合である。図中の margin は④で説明した設計マージンの量を示している。企業で LSI 開発に携わっている研究者より得た知見によると、通常大凡 50～100%の設計マージンが設けられている。グラフより電源電圧の変更間隔が短いと、期待とは逆に消費エネルギーが増大していることが判る。これは上述のオーバーヘッドが原因である。実行時間が延長したために消費エネルギーが増大している。また、2000 クロックサイクル付近で消費エネルギーの削減効果が最大になることもわかる。2000 クロックサイクル時の削減割合は 26.7%である。以下では、電源電圧の変更間隔が 2000 クロックサイクルの場合について、評価結果を報告する。



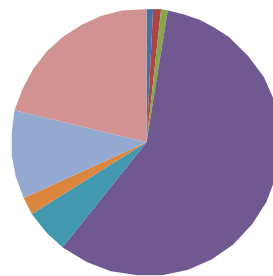
左図はプログラム実行時に選択された電源電圧の分布を示している。実行時間の 70%以上の期間で 1.084V 以下の電圧が選択されていることが判る。不用意に高い電源電圧を選択しないで、不要なエネルギー消費を排除できていることが確認できる。



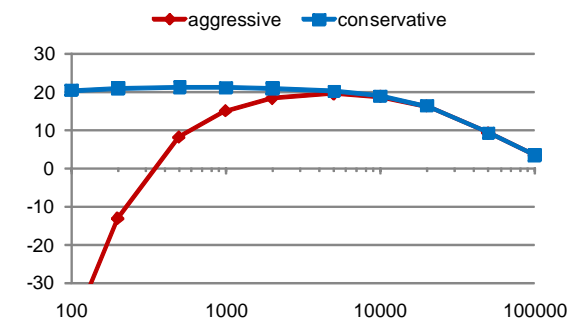
■ 1.340 V ■ 1.260 V ■ 1.228 V ■ 1.180 V  
■ 1.132 V ■ 1.084 V ■ 1.036 V ■ 0.988 V

右図はデュアルコア MeP の場合の結果である。同時に二つのプログラムが実行されていると、どちらか一方が高い電源電圧を要求すると、他方も同じ高い電圧で動作することになる。つまり、シングルコア時と比較すると、電源電圧を下げる機会が少なくなると予想される。右図よりこの予想が正しいことが確認出来る。全実行期間の 57.9%で比較的高い 1.180V を選択している。一方で 1.260V 以上の電圧は 2%未満の期間でのみ選択されており、不用意に高い電圧を選択していないことが確認できる。デュアルコア MeP でのエネルギー削減効果は、プログラムの組み合わせで変動するが、平均 20.5%であった。

右図はデュアルコア MeP の場合の結果である。同時に二つのプログラムが実行されていると、どちらか一方が高い電源電圧を要求すると、他方も同じ高い電圧で動作することになる。つまり、シングルコア時と比較すると、電源電圧を下げる機会が少なくなると予想される。右図よりこの予想が正しいことが確認出来る。全実行期間の 57.9%で比較的高い 1.180V を選択している。一方で 1.260V 以上の電圧は 2%未満の期間でのみ選択されており、不用意に高い電圧を選択していないことが確認できる。デュアルコア MeP でのエネルギー削減効果は、プログラムの組み合わせで変動するが、平均 20.5%であった。



■ 1.340 V ■ 1.260 V ■ 1.228 V ■ 1.180 V  
■ 1.132 V ■ 1.084 V ■ 1.036 V ■ 0.988 V



左図はクアドコア MeP の場合の結果である。まず赤線の aggressive に注目されたい。電圧変更間隔が小さくなるとエネルギー削減効果が失われることが判る。これは 4 つのコアが頻りに電圧を変更し、そのオーバーヘッドによる実行時間の増大が深刻になるためである。そこで、電源電圧の変更ポリシーに工夫をし、不用意に電圧を下げない様にした。その結果が青線の conservative である。広いクロックサイクル間隔でエネルギー消費を削減出来ている。

#### ④結果の位置づけ

本方式はシミュレーションによる評価で終了し、実チップへの実装には至らなかった。しかし、本方式が実用性の点で問題があるというわけではなく、寧ろ類似研究と比較して実用性が高いと思われる。最も近い研究はミシガン大学らによる Razor である。カナリア FF がデータパスに遅延を挿



入したのとは異なり, Razor ではクロックに遅延を挿入している. そのためホールド時間制約の解消が困難になるという問題が生じている. また Razor ではタイミング違反を予報するのではなく, 検出する. マイクロプロセッサのような内部状態を巻き戻す機構を持つ LSI では利用可能であるが, ほとんどの LSI にはそのような機構が存在しない. この点でもカナリア FF の方が実用的であると言える.

## (2)研究成果の今後期待される効果

エネルギー消費とディペンダビリティの間にはトレードオフがある. 高いディペンダビリティを提供するためには, 大きなエネルギー消費を要する. 現代社会では益々高いディペンダビリティが求められるようになってきており, 今後は両者のトレードオフを考慮した省電力化技術が求められるが, (1)の④で述べたように本方式は実用性が高いと期待され, 加えてディペンダビリティを獲得する方式への応用も期待できるため, ディペンダビリティを考慮したエネルギー消費削減技術への波及が期待される.

4. 9 省電力メモリおよび省電力レジスタ回路とその制御技術(九州大学グループ)  
 (1)研究実施内容及び成果

マイクロプロセッサではキャッシュメモリやスクラッチパッドメモリなどのオンチップメモリの消費電力とフリップフロップのクロック変化に伴うレジスタ回路における消費電力が非常に大きい。図 9.1 に 0.18  $\mu\text{m}$  テクノロジを使用し独自に設計を行ったマイクロプロセッサ上でベンチマークプログラムを実行した際のマイクロプロセッサの消費電力内訳を示す。プロセッサはクロックゲーティングを適用して設計した。図 9.1 に示す通りマイクロプロセッサにおいてレジスタ回路とオンチップメモリの消費電力が全体の約 40 %を占めることがわかる。マイクロプロセッサの低消費電力化のためにはロジック部分と同様にメモリやレジスタ回路の低消費電力化が重要である。ロジック部分は比較的低い電源電圧を利用しやすいが、オンチップメモリの低電圧化は信頼性の劣化を招くため、単純な低電圧化は適用しにくい。本研究では、信頼性を損なうことなくオンチップメモリとレジスタ回路の低消費電力化を実現する技術の開発に取り組んだ。

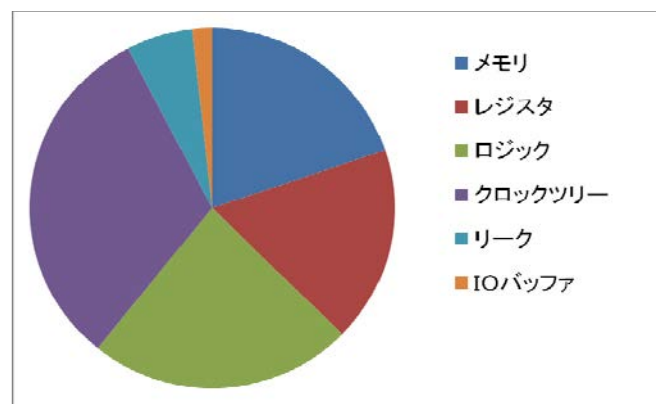


図 9.1 マイクロプロセッサの電力内訳

具体的にはオンチップメモリとレジスタ回路の低消費電力化に関して主に次の4つの技術を開発した。

● 1サイクルでアクセス可能な2階層キャッシュメモリ

1次キャッシュメモリよりも CPU に近い階層に小容量の0次キャッシュメモリを搭載することによりキャッシュメモリアクセスの省エネルギー化を実現可能であるが、0次キャッシュメモリを備える従来の2階層キャッシュメモリは、0次キャッシュメモリにミスヒットすると1次キャッシュメモリにアクセスするためには1サイクル以上のペナルティサイクルが必要であった。本研究で開発した2階層メモリは図 9.2 に示すように0次キャッシュメモリ (Small-cache) と1次キャッシュメモリ (Main-cache) にいずれも1サイクルでアクセス可能な仕組みを備える。

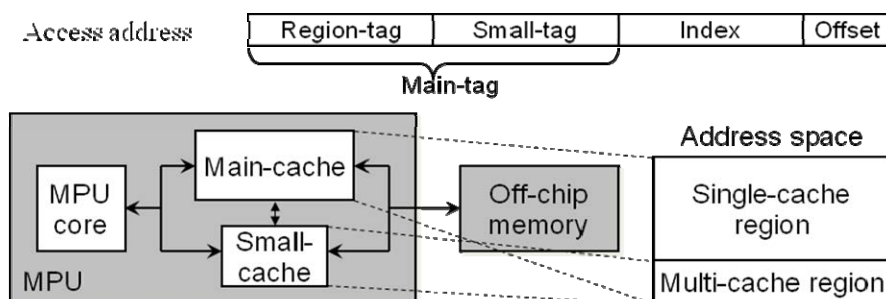


図 9.2 Single-Cycle Accessible Two-Level Cache Memory

従来型の1次階層のみのキャッシュメモリと比べて性能を落とすことなく0次キャッシュメモリ利用

によるキャッシュメモリアクセスの省電力化を実現した。この技術により、アプリケーションプログラムによっては、既存の0次キャッシュメモリを用いた手法と比較して消費電力を最大 10%削減できることをシミュレーションにより確認した。

● **2電源電圧オンチップメモリ(ハイブリッドメモリ)**

SRAM で構成されるオンチップメモリを2種類の異なる電圧を使用するブロックに分割し、低い電圧を割り当てたブロックにアクセスを集中させることにより平均メモリアクセス電力を削減する手法を開発した。従来から多電源電圧を使用するSRAMは多数提案されているが、単純にメモリの電源電圧を低下させるとノイズマージンが劣化し信頼性が低下する。そこで本研究では低い電圧を使用するメモリブロックのセルサイズを最適化し、低電圧下でもスタティックノイズマージンを劣化させない仕組みを開発した。SRAMのノイズマージンを改善するためにはSRAMセルのサイズを大きくする方法が有効であるが、単純にセルサイズを大きくするとメモリの面積増大が問題になる。そこで、全メモリ容量のうち約15%にあたるメモリブロックのみを低電圧化し大サイズメモリセルに置き換えることによりメモリ全体の面積増大を3%未満に抑えて平均電力を50%削減することに成功した。

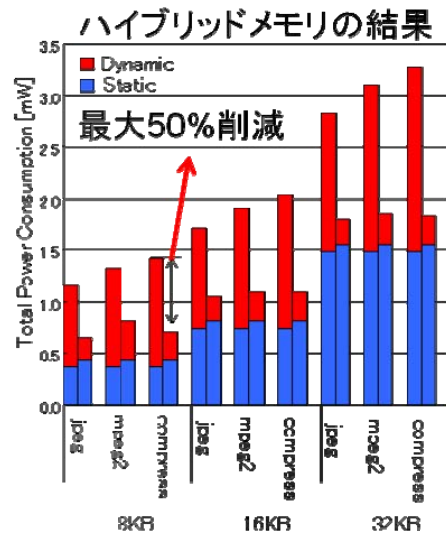
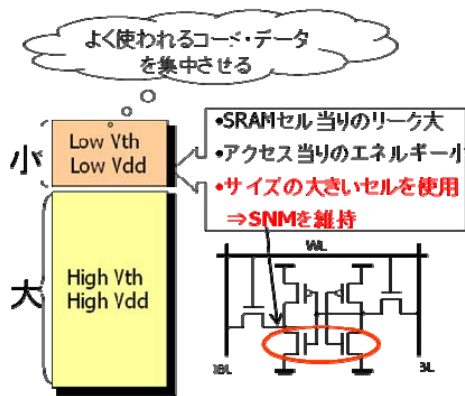
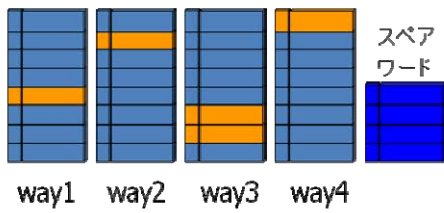


図 9.3 2電源電圧オンチップメモリ(ハイブリッドメモリ)の構成と結果

● **スペアメモリブロックの有効利用によるオンチップメモリの低消費電力化**

オンチップSRAMモジュールのワードごとのアクセス時間をチップ製造後に計測し、極端にアクセス遅延の大きいワードあるいはキャッシュラインをスペアメモリで置き換えることにより、メモリの性能を劣化させることなく平均消費電力を削減する手法を開発した。スペアメモリは従来から故障したメモリブロックを救済する目的で利用されていたが、本研究ではこれを応用し、プロセスばらつきの保障技術として活用した。最先端の製造技術を利用するとメモリのアクセス時間がメモリセルごとに大きく異なる可能性がある。メモリブロック全体を利用するためには最悪ケースに合わせてメモリアクセスサイクル時間を大きく設定する必要があった。本研究ではごく一部の不良ワード(アクセス時間の大きいワード)をスペアワードで置き換えることにより最悪ケースのメモリアクセス時間を大幅に改善することに成功した。逆に、メモリに供給する電源電圧を低くしても性能劣化とプロセスばらつきの影響を小さく抑えることができ、低電圧効果により平均アクセス電力を大幅に改善することができる。90nmプロセステクノロジーのSPICEモデルを利用したモンテカルロシミュレーションにより、5%のプロセスばらつきを仮定した場合に、わずか40ワード分のスペアワードを用意するだけで256×256bitのメモリのリーク電力を70%削減できることを確認した。将来的にメモリブロックのリーク電力の割合が動的電力と同等になると仮定すると(図9.4参照)、本手法によりメモリブロック全体の電力を35%削減できる計算になる。

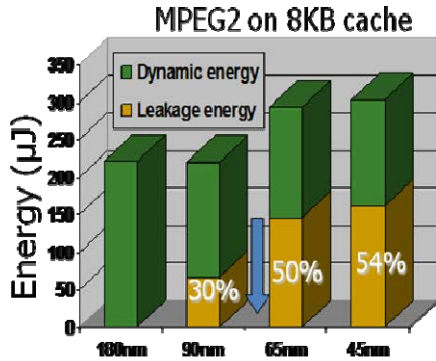
遅延の大きいラインをスペアワードで置き換え



高閾値電圧のプロセスを使用しても最悪ケースの遅延は増えない ⇒ リーク削減

メモリの合計電力を約35%削減(2/3)

ばらつきが5%の時リーク電力を70%削減



電力シェアの将来予想  
S. Rodriguez, et al, "Energy/power breakdown of pipelined nanometer caches (90nm/65nm/45nm/32nm)" ISLPED 2006.

図 9.4 スペアメモリワードの利用によるオンチップメモリの低消費電力化

● ゲーテッドフリップフロップ結合によるレジスタ回路の低消費電力化

レジスタ回路の低消費電力化のためにはその構成要素であるフリップフロップの低消費電力化が重要である。低消費電力フリップフロップとしてゲーテッドフリップフロップ(以下 GFF)が提案されている。GFF は現在記憶している値と新たに入力される値が異なる時だけフリップフロップにクロックを供給するため、フリップフロップ内部のクロックネットワークで消費される電力は極めて小さい。しかし従来型の GFF は1ビットごとにクロックを停止する回路を備えていたために、このクロック停止回路で消費される電力が大きいという問題があった。さらには、従来型の GFF は回路面積が標準的なフリップフロップの2倍近くあり、面積コストが増大するという問題もあった。本研究では1ビットのゲーテッドフリップフロップを複数ビット統合することによりクロック停止回路で消費する電力と面積のオーバーヘッドを大幅に削減する方法を開発した。商用の組込みプロセッサのレジスタ回路に、提案する統合 GFF を適用してシミュレーションにより評価した結果、標準的なフリップフロップで構成されるレジスタ回路と比較して 50%以上レジスタ回路の電力を削減できることを確認した(図 9.5 参照)。また、図 9.5 に示す通り、従来型の GFF を適用した場合と比較してもレジスタ回路の消費電力を平均で 27%、最大で 39%削減できることを確認した。

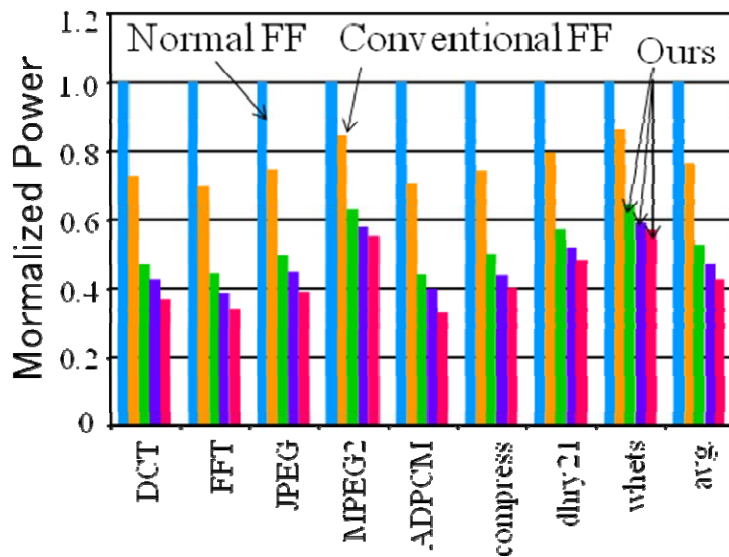


図 9.5 GFF の適用による電力削減

## (2)研究成果の今後期待される効果

本章ではメモリのハードウェアの省エネルギー技術についてのみ述べたが、これらのメモリをコンパイラやOSが有効に利用する技術に関しては十分に検討できていない。今後は、対象とするマルチコア・マルチタスクシステムにおいて、コンパイラとOSが上述のメモリを有効に活用し、システムとしてエネルギーを削減する技術を開発する。

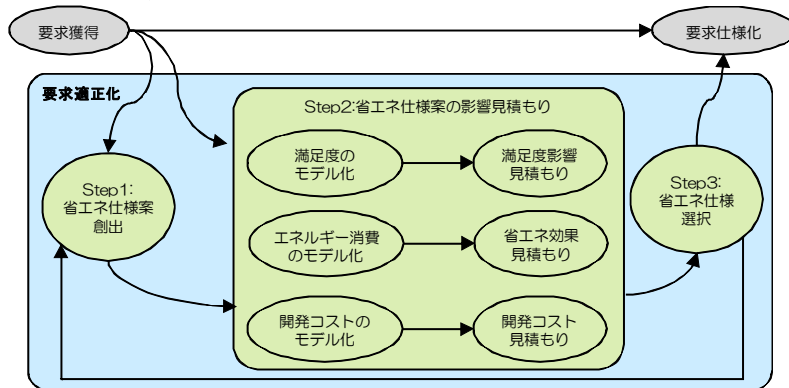
#### 4. 10 QoS 要求適正化(東芝グループ)

##### (1)研究実施内容及び成果

多くの電気・電子機器の製品開発において、省エネルギー化が求められるようになってきている。一般には構成デバイス自体の電力消費効率の改善や、アイドル時の部分的な電源断などの制御により解決を試みることが多い。しかしながらこれらはシステム全体から見た場合は部分的な削減に留まっている。家庭内の電力ネットワークの制御など、システム全体視点でのアプローチも存在してはいるが、それらはターゲットドメインに特化した取り組みであって、他のドメインへの流用が容易とはいえない。これらの結果、多くのシステム開発において、品質(特に高機能化や高性能化)が省エネルギー化よりも優先されてしまう傾向があった。このような問題に対し、上流工程からのシステム全体視点でのより汎用なアプローチが必要であると考えている。以上を背景に、要求工程から省エネルギー化を行うことを目的とする「QoS 要求適正化手法」を提案する。

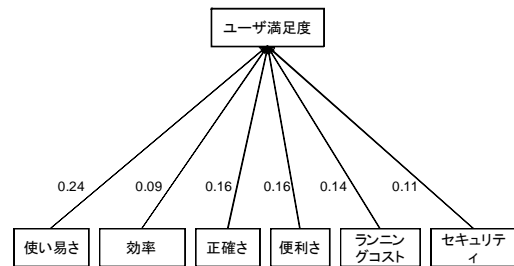
本手法の最大の特徴はユーザ満足度を考慮している点である。これによって、ユーザ満足度を損ねることなく、省エネルギー化を可能にする。もう一つの大きな特徴は、開発コストを考慮している点である。これによって、製品開発コストを現実的な範囲内に抑えることを可能にする。

本手法の概略のフローを以下に示す。本手法は一般的な要求工程における要求獲得と要求仕様化の間に位置づけられる。また本手法は大きく3つのステップから構成されている。



最初のステップは「省エネ仕様案の創出」である。このステップではチェックリストベースの発想法とブレインストーミングを用いて、開発対象における省エネルギー化のための仕様案を創出する。

2 つ目のステップは「省エネ仕様案の影響見積もり」である。ここでは省エネルギー化を行う上で考慮せねばならない様々な要素について、その影響を見積もる。現時点ではユーザ満足度、省エネルギー効果、開発コストの3要素についての影響を見積もることとしている。ユーザ満足度への影響見積もりでは、ゴール指向要求分析や、ユーザアンケートと一対比較などを用いながらユーザ満足度の分析を行い、ユーザ満足度木を構築する(右図はケーススタディでのユーザ満足度木の例)。そしてこれに基づき仕様案個々に対し評価を行う。省エネルギー効果の見積もりでは、開発対象の構成要素を基に作られたエネルギー消費モデルを構築し、それに基づき仕様案個々に対し評価を行う。開発コストの見積もりでは、ファンクションポイント法やオブジェクトポイント法を参考に仕様案個々に対する開発コストの見積もりを行う。そのためにはまず一通りの要求に基づいて概略のオブジェクト指向分析(ソフトウェアアーキテクチャ設計)を行い、ソフトウェアの概念モデルを構築する。そしてこれに基づき、仕様案個々に対しそれを実装した場合の開発コストを見積もる。



3 つ目のステップは「省エネ仕様の選択」である。ここでは、2 つ目のステップで得られた仕様案個々の評価値に基づいて、選択する仕様を決定する。選択は、まずユーザ満足度への影響がないものを選択し、そのうえでそれらに対し、省エネルギー効果と開発コストに基づく費用対効果を算出し、一定の基準をクリアしたものを選択することとする。このステップで最終的に選択された仕様案に対し、それらによる全体としての省エネルギー効果を計算し、目標の削減量に達していなかつ



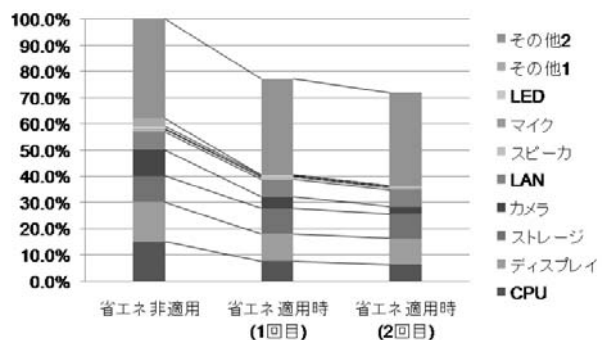
た場合、再度 1 つ目のステップから繰り返すこととする。

2 巡目以降の Step1 では、前のラウンドの Step1 の結果を参考に行うと効率を上げることができる。具体的には、前の繰り返しでは選択されなかった仕様案の弱点を見直すことや、多くの仕様案が創出されたチェックリスト上のキーワードに基づいて再度創出を行う、などで効率向上を図れる。

以上までに示した提案手法を用い、ネットブックのような小型ノート PC を用いたネットワーク TV 会議システムを対象にケーススタディを行い、効果の確認や手法のブラッシュアップを行った。以下の表は、結果の一例である。Step1 の部分には創出された仕様案が、Step2 の部分には各要素毎に見積もられた評価値が、Step3 の部分には取捨選択された後の消費エネルギー効果が、それぞれ個々の仕様案毎に示されている。

Step 1			Step 2				Step 3		Step 2					
番号	省エネ案	実現方法	Step 2.1	Step 2.2	Step 2.3	省エネ効果 (選択結果を反映)	省エネ効果 (選択結果を反映)	Step 2.2						
			沖による満足度	開発コスト	会議に占める時間の割合(%)			省エネ効果 (適用した場合)	時間の割合 (選択結果)	カメラ	10% CPU (エンコー)		4%	
								自分	他人	省エネ効果	自分	他人	省エネ効果	省エネ効果
1	自分が映像の送信を止めているときには電源切る(ミュートの映像版)	ボタンを押して On/Off を制御。カメラを切る。映像のエンコードも止める。	1.00	8	5%	0.8%	100%	0.8%	100%		0.5%	80%		0.16%
2	会議が停滞したらエンコード・デコードをいっしょに加減	音声が一定時間なかったら停滯と判断。デコードを半分に減らす。	1.00	31	10%	0.4%	0%	0.0%			30%	30%		0.1%
3	送る画がユーザー設定でビットレートを落とせる	色数を落とす。画像サイズを小さくする。	0.94	12	33%	1.4%	100%	1.4%			30%	30%		0.4%
.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....

本ケーススタディでの最終的な消費エネルギー削減は右のグラフのようになった。本ケーススタディでは提案手法を 2 巡行っており、グラフでは各反復の結果が示されている。1 巡目では、演算処理に関する部分で約 37.9%(システム全体としては約 25.1%)の省エネルギー化を、また 2 巡目では、演算処理に関する部分で約 40.2%(システム全体としては約 28.5%)の省エネルギー化を行える見込みであることを確認することができた。



## (2)研究成果の今後期待される効果

今回は QoS 要求適正化手法を提案し、ケーススタディによって効果の確認を行ったが、ケーススタディでは要求分析を行ったところまでで終わっており、省エネルギーの視点では、あくまで見積もり値としての効果の確認しかできていない。今後は実装までを伴う事例での効果の確認が必要になると考えている。また現時点の提案手法では、多くの意思決定すべき個所で、エンジニアのスキルやドメイン知識に依存してしまっている。エンジニアのスキルや知識になるべく依存しないよう、成熟度を高める必要があると考えている。加えて、ケーススタディではネットワーク TV 会議システムを対象としたが、ターゲットドメインが変われば、今回想定した 3 要素(ユーザ満足度への影響度、省エネルギー化の効果、開発コスト)以外の要素も検討に入れる必要が生じてくる場合もあると思われる。このような新たな要素の取り込みについても今後の検討課題である。

一方で、本手法の基本概念については、一般的な要求工程の一つのステップである要求調整にも応用可能ではないかと考えている。複数の要求から最適な要求(のセット)を選択するためには様々な側面でのトレードオフを分析し、調整する必要があると考えられる。本手法では省エネルギー化の視点を中心に、ユーザ満足度への影響や開発コストを考慮して調整を行う手法であると捉える事ができる。このことから、上述もしているが、手法内で分析や影響見積もりを行う要素の種類や組み合わせを増やすことで、要求調整へも適用可能になると考えている。ただしこの分析や影響見積もりに関しては、低コストで実践できるようになること、特定のドメイン知識やスキルにはなるべく依存しないようにすること、などの課題が残されていると考えている。

#### 4.11 統合評価システム用シミュレータの開発(東芝グループ)

##### (1)研究実施内容及び成果

###### ■本テーマの位置づけ

本研究の経緯にしたがって提案された種々の手法について有効性を定量的に測定できるような評価環境として、マルチコアシミュレータと評価用アプリケーションの構築を行った。本テーマは、研究年度全体において研究により提案された手法、および、試作された装置を計測し、研究の効果を評価することを年頭に実施した。

###### ■実施内容

###### 1) シミュレータ開発

シミュレータの開発に当たっては、九大で設計を進めているチップをシミュレーションレベルでも評価可能とすることが課題となった。なお、チップは MeP がベースとなっていたため、既存の MeP シミュレータを利用することとし、そこに機能の追加を行った(図 1)。具体的な機能追加項目を次に示す。

- コア間割り込み
- コア単位の動作周波数変更
- キャッシュ WAY 数変更
- トレース出力(メモリアクセス、データキャッシュ、命令キャッシュ、命令バッファ)
- 分岐ペナルティや DMA 転送のレイテンシ指定機能
- シミュレーション一時停止機能

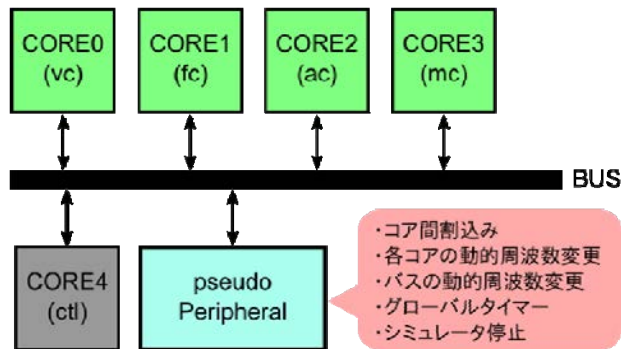


図 1 シミュレータの構成

これらの機能追加により、設計しているチップに近い動作を想定したシミュレーションを行うことが可能となった。また、シミュレーションで得た結果を、チェックポイントの設定や電力見積もりツールに必要な解析に用いやすくするための補助機能として、コア単位や単位時間のログ分割出力機能を作成した。

###### 2) 評価アプリケーション開発

本チームの省電力効果測定では、実際に運用を想定したアプリケーションを用いることが念頭に置かれた。そこで、図 2 に示す P&P 型のテレビ会議システムを題材に評価を進めることとし、研究 3 年度目から構築を開始した。なお、テレビ会議システムの機能はディスプレイ出力部やネットワーク部などの機能に分割されるが、この中でもっとも CPU に対する電力効果の測定のために、CPU 処理の差による影響が大きくなる映像や音声コーデックのエンコード・デコード処理を選択することとした。

評価アプリケーション構築開始当初は、MeP チップ上で動作するコーデックがなく、また MeP 向けの TOPPERS 環境にも制限があったため、図 3 に示すようなコーデ



図 2 テレビ会議システムの概要



図 3 テレビ会議システム (ver. 1)



ック処理のみを含むシステムの構築からスタートすることとした。この時点で、マルチコア環境で動作するビデオ会議システムが動作するようになった。

次に、第4年度目に TOPPERS/ASP カーネルの MeP 版が完成したことに伴い、一旦シングルコア構成に戻した上で、全てのコーデックを TOPPERS/ASP カーネル上で動作するように変更を行った(図5)。これにより、単一コアレベルでの電力評価が可能となった。また、QoS 要求適正化の効果を測定するために、映像のフレームレートを落とした紙芝居モードやミュート機能も組み込んだ。

そして最終年度では、想定する全ての評価が可能となるように、TOPPERS/FMP カーネルの上に、これまで構築した機能に加え、マルチタスク/並列タスク化したテレビ会議システムを構築した(図4)。

最終的には、図6に示されるように、各要素技術をシミュレーション環境上で評価することが可能となった。具体的には、九大が作成したチップ仕様に基づくハードウェア上で、TOPPERS/FMP カーネルおよび評価アプリケーションがプログラム解析やメモリアウト最適化技術によって電力を最適化した状態で動作することとなる。また、動作結果から得られるログを利用して、エネルギーの見積もりを行い、フィードバックすることも可能となった。



図5 テレビ会議システム(TOPPERS/ASP 対応)

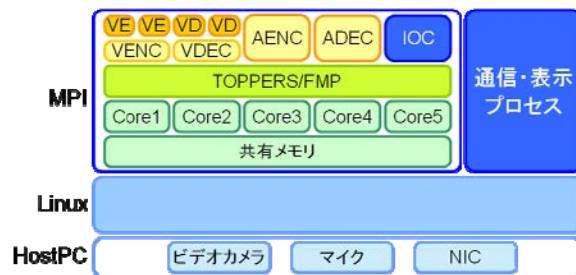


図4 テレビ会議システム(TOPPERS/FMP 対応)

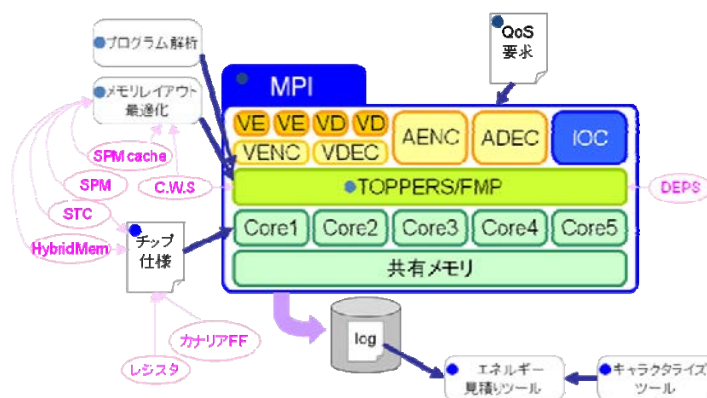


図6 各要素技術とシミュレーション環境との関係

## (2)研究成果の今後期待される効果

本シミュレータ技術によって、電力見積りに利用可能なレベルまで、チップの動作を詳細に取得することが可能となった。実チップ稼動前に電力面での詳細な評価が行えるようになったことで、消費電力まで想定した設計に結びつくことが効果として考えられる。

#### 4. 12 省電力ソフトウェア設計手法(東芝グループ)

##### (1)研究実施内容及び成果

マルチパフォーマンスプロセッサは、動作スピードやキャッシュ構成など実行時に設定可能な複数の動作モードを有する。省電力プラットフォームが制御可能な動作モードを提供するとエネルギー効率を高める余地が増えるが、性能を引き出すにはソフトウェアの特性に応じて細やかに動作モードを制御する必要があり、ソフトウェア開発者の負担が大きくなる。近年、組み込みシステム開発ではソフトウェア開発のコストを無視できないので、省電力プラットフォームの性能を引き出しつつ開発コストを抑える技術の開発が課題である。

この研究テーマでは、組み込みシステムのプラットフォームの省エネ性能を最大限に引き出す、応用ソフトウェアの実装技術を開発してきた。

プロジェクト最終目標の組み込みシステムの超低消費エネルギー化には、システムを構成するプラットフォームと応用ソフトウェアの協調設計および協調動作が欠かせない。省電力プラットフォームが多様な省エネ機構を提供しても、省エネ機構を活かした応用ソフトウェアの開発に多大なコストがかかれば、省電力プラットフォームは産業界では普及しない。そこで、産業界の立場から省電力プラットフォームの能力を活かす応用ソフトウェアの効率的な実装技術を開発している。

研究上のアプローチは次の通りである。応用ソフトウェアの動作プロファイリングの結果をもとに省電力プラットフォームの制御パラメータを最適化する。省電力プラットフォームに依存する制御コードをアスペクト指向プログラミングの手法でモジュール化してアプリケーションへ組み込む。

上記のアプローチを DVFS 制御へ適用して評価し国内特許（1 件）を出願した。  
進 博正（株式会社東芝）、「マイクロプロセッサ制御装置ならびにその方法およびプログラム」、P2007-237818、2007 年 9 月 13 日

##### (2)実施内容

この研究では、アスペクト指向プログラミング（以下、AOP: Aspect-Oriented Programming）と機械学習を併用するアプローチを取る。AOP によりアプリケーションから ULP プラットフォームに依存する制御コードを分離しモジュール化し、機械学習によりアプリケーションと ULP プラットフォームの特性を考慮した最適制御を実現する。この成果を利用すると「プラットフォーム独立なアプリケーションとプラットフォーム依存の ULP アスペクトから低消費エネルギーなアプリケーションを自動生成」できる。

ULP アスペクトは、ULP プラットフォーム（OS やプロセッサなど、アプリケーションの実行基盤）とアプリケーションの特性を記述すると共に、ULP プラットフォームに依存する制御コードをモジュール化する（図 7）。たとえば動画プレーヤー向けの DVFS (Dynamic Voltage Frequency Scaling) 制御を考えると、アプリケーションの特性はフレーム生成タスクの完了デッドライン（例えば 30ms）とタスクの時間計測における開始点と終了点などとなる。プラットフォームの特性は DVFS 制御で選択可能な動作スピードや電圧となる。

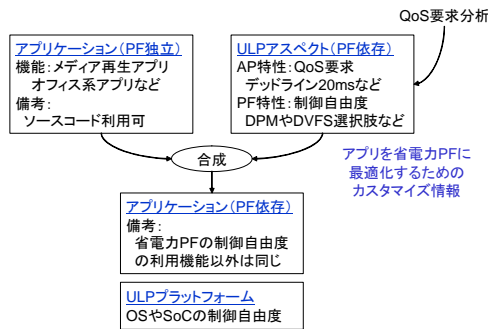


図 7 ULP アスペクトの概要

計算負荷が変動する周期タスクを模擬するアプリケーションにより DVFS 制御の実験を行った。このアプリケーション (図 8) は、起動するとデッドライン付きの周期タスク task() を繰り返し実行する。周期タスクは 80%の確率で短いタスクとなり、20%の確率で長いタスクとなる。長いタスクは短いタスクの約 2.3 倍 (=7/3) の計算量となる。タスク計算量の長短はスイッチ文の何れに入るかの制御フローの相違となる。

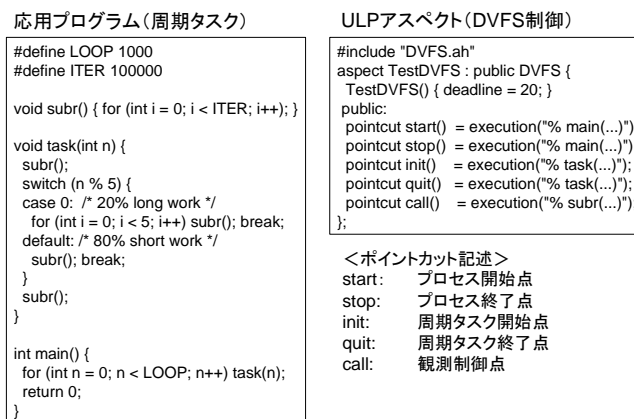


図 8 アプリケーションと ULP アスペクト

ULP アスペクトは抽象アスペクト DVFS を継承した具象アスペクト TestDVFS である。抽象アスペクト DVFS は、プロセスの開始 (start) と終了 (stop)、タスクの開始 (init) と終了 (quit)、観測と制御点 (call) における共通の処理内容 (AOP のアドバイス) を記述する。DVFS 制御の本質的な制御はすべてここに記述される。また共通前処理や後処理など、DVFS 制御を行う際に必ず必要となる基本的な処理もこれに相当する。ポイントはプラットフォームに依存したそれらの複雑な制御が全て抽象アスペクトとして隠蔽されていることにあり、プログラマは単にそのアスペクトを継承して利用するという宣言を行うだけで、複雑な DVFS 制御が利用可能になるのである。

具象アスペクトは、プロセスの開始と終了、タスクの開始と終了、観測と制御点が、現在のアプリケーションの何処に対応するかをポイントカット構文で記述し、タスクの完了期限 (deadline) をコンストラクタ内の代入文で記述する。今回のアプリケーションはただ一つのタスクを持つ例題であり、そのタスクの完了期限が 20ms であるのでこのような記述になっている。完了期限の異なる複数のタスクが動作するアプリケーションの場合は、タスクごとに具象アスペクトを準備することで、完了期限のみが異なる DVFS 制御を簡単に

組み込むことができるようになる。

アプリケーションと ULP アスペクトをアスペクト指向プログラミング言語のコンパイラでコンパイルすると、ULP アスペクトのポイントカット構文で指定したアプリケーションの関数の前後にアドバイスを挿入したアプリケーションが生成される。アプリケーションの実行中にアドバイスに入るとコールスタックを取得してアプリケーションの実行状態を識別する。この例では、観測制御点が関数 `subr()` の前後となるが、タスクの長短に応じて呼び出し位置が異なるので、状態の違いとして識別できる。ULP アスペクトのポイントカット構文の記述すなわち「観測制御点を何処にするか」が開発者のチューニングする自由度となる。

このアプリケーションと ULP アスペクトを用いた実験結果を示す。図 9 の有向グラフは制御装置が実際に観測するアプリケーションの状態遷移系である。制御装置はアプリケーションの実行状態の一部しか観測しないので非決定的な状態遷移系となる。状態に対応するノードはアプリケーションのコールスタックで識別される。遷移に対応するアークは、観測した状態間の隣接関係を示す。遷移のアークはアプリケーションが実行中を意味し、図のラベルにはアプリケーションの経過サイクル数と分岐確率を示した。たとえば、タスク開始点が状態 `n0` で、次の状態 `n1` へは確率 100%で遷移し経過サイクル数は 166 サイクルとなる。状態 `n1` の次は確率 20%と確率 80%で状態 `n2` と状態 `n5` へ分岐し経過サイクル数は共に約 40 万サイクルとなる。長いタスクに対応するパスは約 300 万サイクルとなり、短いタスクに対応するパスは約 120 万サイクルとなる。タスクの完了デッドラインを 20ms とすると、短いタスクを実行する場合は平均 60MHz の動作スピードで間に合うが、長いタスクを実行する際は平均 150MHz 以上の動作スピードが必要となる。各ノードに示した動作スピードは、強化学習の一手法である Q 学習で得た報酬総和の期待値が最大の最適方策である。短いタスクを実行する場合に 67MHz（一部 133MHz）で処理し、長いタスクに入ると 200MHz へ動作スピードを増す妥当な動作スピード割当を学習している。

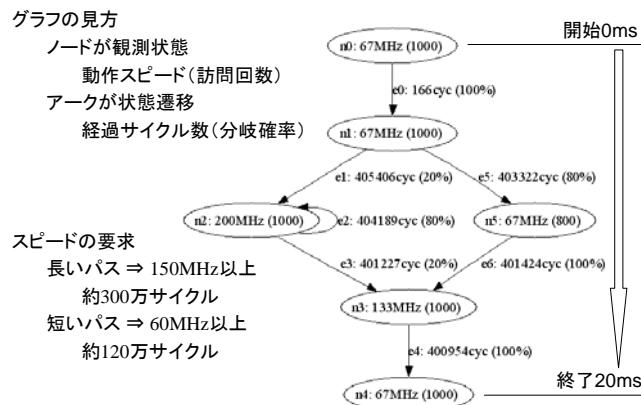


図 9 確率的な状態遷移系

ULP アスペクトによる Q 方式（報酬総和期待値に基づく方式）と R 方式（残余計算量に基づく方式）による DVFS 制御の動的な振る舞いを以下に示す（図 10、図 11）。以下のグラフは周期タスクの繰り返し回数を横軸とし、縦軸に 10 周期あたりの消費エネルギー平均値とデッドラインミス確率をプロットしている。なお、消費エネルギーは消費電力の時間積分値としたが、消費電力は動作スピード三乗の見積り値を利用したので、消費エネルギーの絶対値には意味がない。消費エネルギーを相対的に比較する基準値として、全タスクを一定の動作スピード（67MHz、133MHz、200MHz）で処理した場合の消費エネルギー見積り値をプロットした。

いずれの DVFS 制御方式も学習前は最低スピードを選択するように初期値を設定している。

周期タスクを繰り返す間に適当な動作スピードを学習し、デッドラインミスを抑えながら消費エネルギーを抑える制御が機能している。消費エネルギーとデッドラインミス率の値の変動が安定するループ回数は、Q方式は約300回以降、R制御方式は約100回以降となる。特徴量の変動が安定する領域は、制御パラメータの学習が大よそ収束した領域と見なせる。学習スピードの面でR方式がQ方式の約三倍早いのは、学習パラメータがQ方式はR方式の三倍（動作スピードの選択肢）多いことが理由と考えられる。DVFS制御に限定するとR方式はQ方式より収束性が早く、各状態への到達時刻を考慮したQ方式より柔軟な動作スピードの調整が可能であり、消費エネルギーやデッドラインミス率の低さ面から優れている。ただし、R方式はDVFS制御に特化した制御方式なのでDPM制御自由度の学習には利用できない。

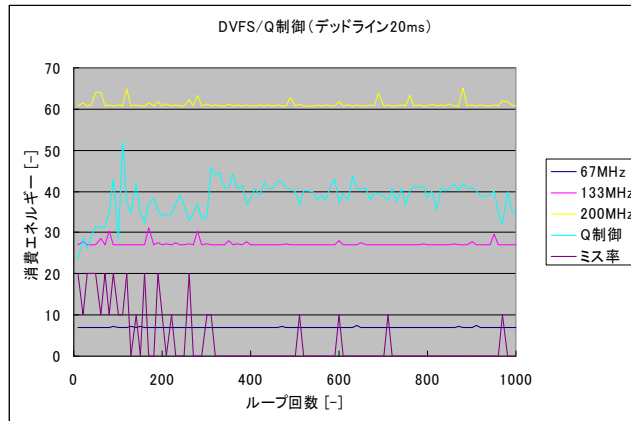


図 10 DVFS/Q 制御の実験結果

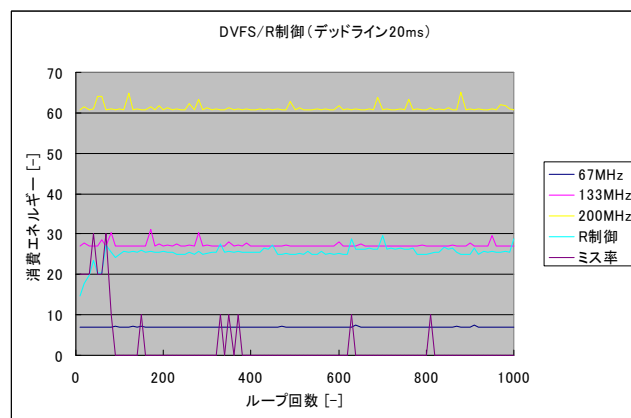


図 11 DVFS/R 制御の実験結果

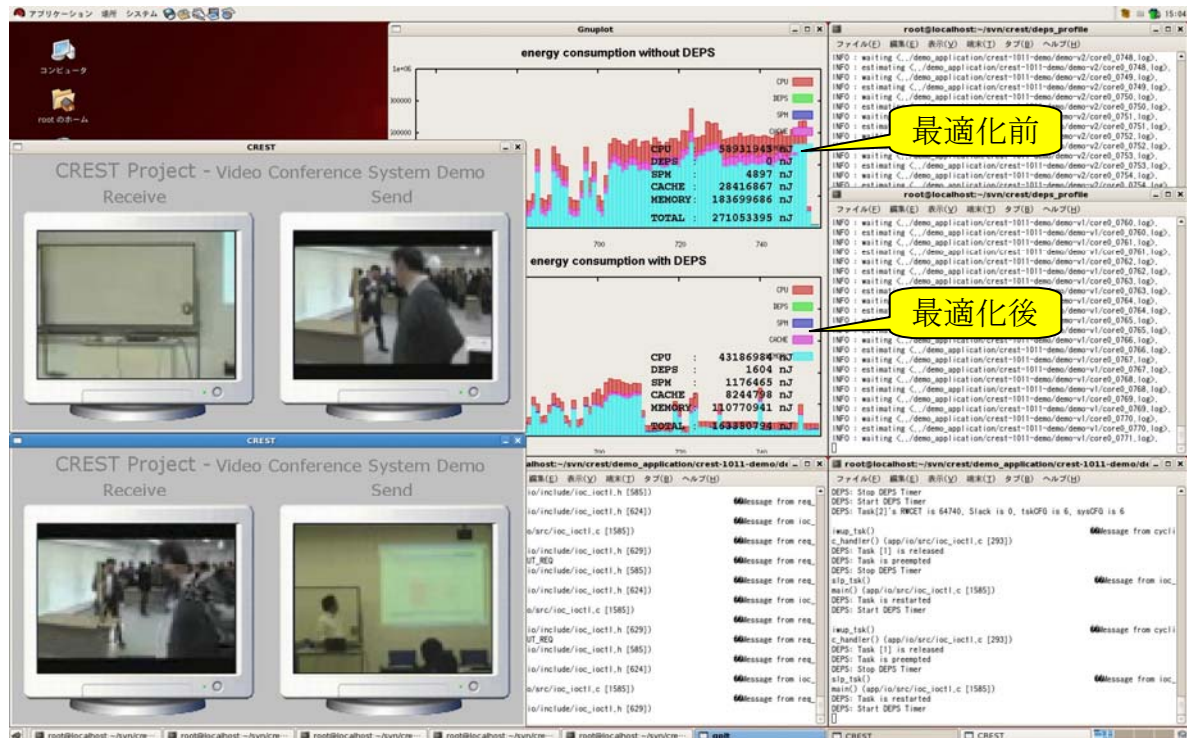
(2)研究成果の今後期待される効果

今後開発されるマルチパフォーマンスプロセッサの低消費エネルギー化の効果を最大限に発揮するソフトウェア開発が容易になり、システム全体としての低消費エネルギー化を推進することが期待される。

#### 4.13 統合評価システムの開発と評価(全てのグループ)

##### (1)研究実施内容及び成果

最終年度である平成 22 年度に、全グループが共同で、統合評価システムを開発し、本研究プロジェクトの統合評価を行った。統合評価システムには、本研究プロジェクトで開発した種々の技術が統合されている。開発した統合評価システムのスクリーンショットを示す。



統合評価システムは、東芝グループが開発したシミュレータ (§ 4.11) をベースとしている。このシミュレータは、元々は標準的な東芝社製 MeP プロセッサの命令セット・シミュレータであったが、九州大学グループが開発したマルチパフォーマンスプロセッサ (§ 4.5) を模擬するように改良されている。具体的には、コア毎に電圧/周波数を動的に変更する機能と、動的にキャッシュウェイを選択する機能が追加されている。

本シミュレータは、九州大学が開発した電力見積りツール (§ 4.7) と連携する。シミュレータはコア毎に実行命令のログを出力する。電力見積りツールは、シミュレータが出力したログを受け取り、一定時間間隔毎に消費エネルギーを計算して出力する。これにより、システム設計者は、実際にチップやプロトタイプボードが完成する前に、ホスト計算機上でシステムの消費エネルギーを評価することができる。また、ハードウェアコンポーネント毎(例えば、CPU コア、命令キャッシュ、命令 SPM など)の消費エネルギーを計算するので、消費エネルギーのボトルネックを容易に解析することができる。また、一定の時間間隔毎に消費エネルギーが計算されるので、タスク毎の消費エネルギーを算出することも可能である。

東芝グループが開発したテレビ会議システム・アプリケーション (§ 4.11) に対し、名古屋大学グループが中心となって開発した ULP ソフトウェア開発環境 (§ 4.1) を適用して、低消費エネルギー化を図った。現時点で適用した要素技術を次ページの表(「現時点」の列)に示す。



要素技術	現時点	今後の目標	試作チップの今後の目標
MPP コア(コア切替え+可変キャッシュウェイ)	✓	✓	✓
カナリア FF			
DEPS	✓	✓	✓
実行トレースマイニング	✓	✓	✓
スラック回収	✓	✓	✓
キャッシュウェイ選択			
低消費エネルギーアルゴリズム			
バッテリー長寿命化スケジューリング			
マルチコア+DPM		✓	✓
QoS 要求適正化	✓	✓	✓
キャッシュ用メモリ配置	命令	✓	✓
	データ		
SPM メモリ配置	命令	✓	✓
	データ		✓
SCATL キャッシュ			
ハイブリッドメモリ		✓	
製造ばらつきの考慮		✓	
低電力レジスタ		✓	
HW アルゴリズム			

本統合評価システムにより評価した結果、テレビ会議システム・アプリケーションを実行したときの消費エネルギーを 44.5%削減することができた(1.8 分の 1)。内訳は、プロセッサの消費エネルギーの削減率が 32.6%、オンチップメモリの削減率が 73.8%、オフチップメモリの削減率が 37.2%であった。

消費電力を大幅に削減することができたものの、当初の目標である 60 分の 1 には遠く及んでいない。主な理由は以下の通りである。

- 現時点の統合評価システムには、実装上の技術的な問題あるいは工数の都合により、本研究を通じて開発したすべての要素技術が実装されている訳ではない(例えば、カナリア FF を命令セット・シミュレータに実装するのは非常に困難)。
- 実装した要素技術の中にも、最適化能力を落として実装しているものがある(実行トレースマイニング、スラック回収アルゴリズムなど)。
- 「最適化前」のアプリケーションも QoS 要求適正化は行われているため、「最適化前」の値(基準値)も厳密には最適化前ではない。つまり、真の削減率は 44.5%よりも大きい。

成果の今後期待される効果



研究期間終了後も継続して統合評価システムの拡張・改良を進めている。また、平成 23 年度より試作チップを使った評価も予定している。前ページの表において、「今後の目標」に統合評価システムの目標（実装を予定している要素技術）を、「試作チップの今後の目標」に試作チップを使った評価の目標を示している。

#### 4. 14 総合評価と今後の見通し(全てのグループ)

##### (1)研究実施内容及び成果

§ 4.13 で説明したように、現在の統合評価システムには、本研究を通じて開発した要素技術のごく一部しか実装していない。そのため、統合評価システムによる評価では、消費エネルギーの削減率は44.5%に留まっていた。

本研究を通じて開発した主要な要素技術を以下の表にまとめる。開発した要素技術毎に、当該技術によりエネルギーが削減されるコンポーネント(プロセッサ、オンチップメモリ、オフチップメモリ)と削減率、その成果の主な発表論文などを記している。削減率は、ベンチマークプログラムや各種パラメータを変更して実験した結果、最も削減率が大きかった場合の値を記している。矢印(→)の前の数値は現状の削減率を表しており、矢印の後の数値は、現在の最先端の製造テクノロジーを用いた場合に得られるであろう削減率(予測)を表している(注:現在設計したプロセッサは、約3世代前の90ナノテクノロジーを使っている)。

技術	効果	対象と削減率			主な研究グループ	主要な発表論文	補足
		プロセッサ	オンチップメモリ	オフチップメモリ			
MPPコア+カナリアFF+DEPS	電力削減	50% → 65%			九大/ 名大	[J-18] [C-84] [I-8]	MPPコアのスケーラビリティが5倍 カナリアが1/3を活用 DEPSがスケーラビリティの3/4を活用 削減率=(1-1/5*(1-1/3))*3/4
実行トレースマイニング+DEPS		18%				[C-85]	
スラック回収+DEPS		60%				[C-78]	
MPPコア+キャッシュウェイ選択	電力削減	10%			九大	[J-18] [C-49]	
MPPコア+低エネルギーアルゴリズム	エネルギー削減	10%			名大	[J-10] [C-57]	
MPPコア+バッテリー長寿命化スケジューリング	バッテリー長寿命化	23%			名大	[J-16]	厳密には、本手法は消費エネルギーを削減するのではなく、バッテリーの特性を利用して、バッテリーを長寿命化する。
マルチコア+DPM	ピーク性能向上 電力削減	50% → 67%			九大	[C-84] [C-4]	
QoS要求適正化	処理量削減 信頼性要求適正化	40%			東芝	[C-87]	
SPM/キャッシュ用コード/データ配置	電力削減 性能向上		46%		九大	[J-4]	
SPM用コード/データ配置タスク間最適化	電力削減 性能向上		46%		名大/ 九大	[C-74] [C-83]	
SCATLキャッシュ	電力削減 性能向上		15%		九大	[J-13]	
ハイブリッドメモリ	電力削減		35%		九大	[J-14]	
製造ばらつきの考慮	電力削減		50%		九大	[J-7] [J-17]	
低消費電力レジスタ	電力削減 性能向上	10%			九大	[C-73] [C-82]	
合計		97.24% (1/36) ↓ 98.71% (1/78)	99.75% (1/405) ↓ 99.88% (1/867)	85.69% (1/7) ↓ 85.69% (1/7)			

表の最終行には、すべての要素技術の削減率を単純に積算した場合の削減率を示している。もちろん、各要素技術の評価で用いたベンチマークプログラムやパラメータは統一されていないため、単純に積算することは適当ではないが、数値目標の達成度を測る目安として提示している。この目安を用いると、プロセッサの消費エネルギーの削減率が 97.24% (最適化前の約 36 分の 1)、オンチップメモリ(キャッシュと SPM)の削減率が 99.75% (約 405 分の 1)、オフチップメモリの削減率が 85.69% (約 7 分の 1)となる。仮に、最適化前のプロセッサの消費エネルギーがシステム全体の 50%、オンチップメモリとオフチップメモリの消費エネルギーが全体の 25%と仮定すると、システム全体の消費エネルギーの削減率は 94.98% (約 20 分の 1)となる。また、最先端のテクノロジーを想定した場合の見通しは、プロセッサの消費エネルギーの削減率が 98.71% (約 78 分の 1)、オンチップの削減率が 99.88% (約 867 分の 1)、オフチップの削減率が 85.69% (約 7 分の 1)、システム全体の削減率が 95.75% (約 24 分の 1)となる。

以上のように、プロセッサとオンチップメモリに関しては、非常に高い削減率を達成している。しかし、オフチップメモリ(DRAM)については、DRAM を極力使わないための技術開発は行ってきたものの、DRAM 自身の消費エネルギーを削減する技術開発は行っていなかった。そのため、オフチップメモリの消費エネルギーの削減率は、プロセッサやオンチップメモリと比較して相対的に低く留まっている。その結果、「最適化前にオフチップメモリの消費エネルギーがシステム全体の 25%を占める」という前提においては、本研究成果の適用後はオフチップメモリの消費エネルギーが支配的となり、システム全体の消費エネルギーは目標値(60 分の 1)に達しない。しかし、DRAM 自体の消費エネルギーが(他の研究グループの技術開発により)削減されているのであれば、システム全体の消費エネルギー削減率はより高くなる可能性がある。

最後に、前ページの表に基づいて積算した削減率の誤差について検証する。統合評価システム (§ 4.13)に実装した要素技術に限って、そのエネルギー削減率を前ページの表に倣って積算すると、プロセッサの削減率は 69.25% (約 3.3 分の 1)、オンチップメモリの削減率は 80.57% (約 5.1 分の 1)、オフチップメモリの削減率は 25% (1.3 分の 1)となる。一方、実際(統合評価システムで評価した場合)の消費エネルギー削減率は、プロセッサが 32.6%、オンチップメモリが 73.8%、オフチップメモリが 37.2%であった。プロセッサについては、積算した削減率は実際の削減率よりも過大であったが、オフチップの積算値は実際の値よりも過少であった。積算値には誤差があるものの、本研究の達成度を測る目安としては有用であると考えている。

#### 成果の今後期待される効果

現時点において、統合評価システムには一部の研究成果しか実装していないため、消費エネルギーを 60 分の 1 にするという数値目標が達成されたことを統合評価システム上で示すことはできていないが、本研究で得られたすべての成果を総合的に適用すれば数値目標を達成できる可能性を示せたと考えている。引き続き統合評価システムの拡張・改良を進め、研究成果を実証する予定である。

## § 5 成果発表等

(1)原著論文発表 (国内(和文)誌 1件、国際(欧文)誌 17件)

1. 著者、論文タイトル、掲載誌 巻、号、発行年

- [J-1] M. Goudarzi and T. Ishihara, "Value-dependence of SRAM leakage in deca-nanometer technologies", IEICE Electronics Express, Vol.5, No.1, pp.23-28, Jan. 2008
- [J-2] Shingo Watanabe, Akihiro Chiyonobu, and Toshinori Sato, "A Low-Power Instruction Issue Queue for Microprocessors", IEICE Transactions on Electronics, Vol.E91-C, No.4, pp.400-409, Apr. 2008.
- [J-3] G. Zeng, H. Tomiyama, and H. Takada, "Dynamic Power Management for Embedded System Idle State in the Presence of Periodic Interrupt Services," IPSJ Trans. on System LSI Design Methodology, vol. 1, pp. 48-57, Aug. 2008.
- [J-4] Yuriko Ishitobi, Tohru Ishihara and Hiroto Yasuura, "Code and Data Placement for Embedded Processors with Scratchpad and Cache Memories," Journal of Signal Processing Systems, Nov. 2008 (online edition).
- [J-5] H. Takase, H. Tomiyama, G. Zeng, and H. Takada, "Energy Efficiency of Scratch-Pad Memory in Deep Submicron Domains: An Empirical Study," IEICE Electronics Express, vol. 5, no. 23, pp. 1010-1016, Dec. 2008.
- [J-6] Maziar Goudarzi, Tohru Ishihara, and Hiroto Yasuura, "A Software Technique to Improve Lifetime of Caches Containing Ultra-Leaky SRAM Cells Caused by Within-Die Vth Variation," Elsevier Journal of Microelectronics, 39(12), pp. 1797-1808, Dec. 2008 (online edition).
- [J-7] Maziar Goudarzi, Tadayuki Matsumura and Tohru Ishihara, "Way-Scaling to Reduce Power of Cache with Delay Variation", IEICE Trans. Fundamentals, Vol.E91-A, No.12, pp. 3576-3584, Dec. 2008.
- [J-8] Maziar Goudarzi, Tohru Ishihara, Hamid Noori, "Software-Level Instruction-Cache Leakage Reduction using Value-Dependence of SRAM Leakage in Nanometer Technologies," Trans. on HiPEAC (Transactions on High-Performance Embedded Architectures and Compilers), vol.3, issue 4, Nov. 2008..
- [J-9] Maziar Goudarzi, Tohru Ishihara, "Process-Variation-Aware Instruction Rescheduling to Reduce Leakage in Nanometer Instruction Caches," CSI (Computer Society of Iran) Journal on Computer Science and Engineering, in press.
- [J-10] 横山哲郎, 今井敬吾, 曾剛, 富山宏之, 高田広章, 結縁祥治, "動的電圧制御システムにおける評価戦略選択に基づく高効率消費エネルギー関数型プログラミング," 情報処理学会論文誌プログラミング, vol. 2, no. 2, pp. 54-69, 2009年3月.
- [J-11] Gang Zeng, Hiroyuki Tomiyama, and Hiroaki Takada, "A Generalized Framework for Energy Savings in Hard Real-Time Embedded Systems," IPSJ Transactions on System LSI Design Methodology, vol. 2, pp.167-179, Aug. 2009.  
doi:10.2197/ipsjtsldm.2.167
- [J-12] Hideki Takase, Hiroyuki Tomiyama, and Hiroaki Takada, "Partitioning and Allocation of Scratch-Pad Memory in Priority-Based Multi-Task Systems," IPSJ Transactions on System LSI Design Methodology, vol. 2, pp.180-188, Aug. 2009.  
doi:10.2197/ipsjtsldm.2.180
- [J-13] Seiichiro Yamaguchi, Yuriko Ishitobi, Tohru Ishihara, and Hiroto Yasuura, "Single-Cycle-Accessible Two-Level Caches and Compilation Technique for Energy Reduction", IPSJ Trans. System LSI Design Methodology, vol.2, pp.189-199, Aug. 2009.  
doi:10.2197/ipsjtsldm.2.189

- [J-14] Tadayuki Matsumura, Tohru Ishihara, and Hiroto Yasuura, “An Optimization Technique for Low-Energy Embedded Memory Systems”, IPSJ Trans. System LSI Design Methodology, vol. 2, pp. 239-249, Aug. 2009.  
doi:10.2197/ipsjtsldm.2.239
- [J-15] Tetsuo Yokoyama, “Reversible Computation and Reversible Programming Languages,” Electronic Notes in Theoretical Computer Science, vol. 253, Issue 6, pp. 71-81, Mar. 2010.  
doi:10.1016/j.entcs.2010.02.007
- [J-16] Tetsuo Yokoyama, Gang Zeng, Hiroyuki Tomiyama, and Hiroaki Takada, “Static Task Scheduling Algorithms Based on Greedy Heuristics for Battery-Powered DVS Systems,” IEICE Transactions on Information and Systems, vol. E93-D, no. 10, pp. 2737-2746, Oct. 2010.
- [J-17] Maziar Goudarzi, Tohru Ishihara, “SRAM Leakage Reduction by Row/Column Redundancy Under Random Within-die Delay Variation,” IEEE Trans. VLSI Systems, vol. 18, no. 12, pp. 1660-1671, Dec. 2010 .  
doi:10.1109/TVLSI.2009.2026048
- [J-18] T. Ishihara, “A Multi-Performance Processor for Reducing the Energy Consumption of Real-Time Embedded Systems,” IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E93-A, No.12, pp. 2533-2541, Dec. 2010.

(2)その他の著作物(総説、書籍など)

- [M-1] 石原 亨, “ソフトウェアに対する電力見積もりと電力削減技術,” 電子情報通信学会 基礎・境界ソサイエティ誌「ファンダムレビュー」, Vol.2, No.3、2009年1月
- [M-2] 高田広章, “ソフトウェアとハードウェアの協調による組込みシステムの消費エネルギー最適化,” 情報処理, Vol.51, No.7, pp. 846-854, Jul 2010.

(3)国際学会発表及び主要な国内学会発表

①招待講演 (国内会議5件、国際会議11件)

1. 発表者(所属)、タイトル、学会名、場所、月日

- [I-1] 石原亨(九州大学), 富山宏之(名古屋大学), “低消費電力化ソフトウェア技術,” 組込みソフトウェアシンポジウム, 東京, 2005年10月.
- [I-2] T. Ishihara (Kyushu Univ.), “Energy-Efficient Embedded System Design at 90nm and Below - A System-Level Perspective -,” International Workshop on Advanced Low Power Systems, Cairns, Australia, July, 2006.
- [I-3] M. Goudarzi, T. Ishihara and H. Yasuura, “Variation Aware Compilation for Improving Energy-Efficiency of Nanometer Processor Caches,” Workshop on Compiler-Assisted SoC Assembly (CASA’06), October, 2006
- [I-4] T. Ishihara and M. Goudarzi (Kyushu Univ.), “System-Level Techniques for Estimating and Reducing Energy Consumption in Real-Time Embedded Systems”, International SoC Design Conference, pp.67-72, Oct. 2007.
- [I-5] 石原 亨(九州大学), “今さら聞けないローパワー ～ 教えます。現場で使えるローパワー設計 ～ ,” Electronic Design and Solution Fair 2008 (EDSFair2008) , 2008年1月.
- [I-6] 石原 亨(九州大学), “ソフトウェアの消費エネルギー解析と最適化技術,” 第21回回路とシステム(軽井沢)ワークショップ論文集, pp.343-348, 2008年4月
- [I-7] Maziar Goudarzi, Tohru Ishihara (Kyushu Univ.), “Redundancy Techniques for SRAM

Leakage Reduction in Presence of Within-Die Delay Variation,” in Proc. of International SoC Design Conference, Nov. 2008.

- [I-8] G. Zeng, T. Yokoyama, H. Tomiyama, H. Takada (Nagoya Univ.), and T. Ishihara (Kyushu Univ.), “A Generalized Framework for Energy Savings in Real-Time Multiprocessor Systems,” In Proc. of International SoC Design Conference (ISOCC), Invited paper, LG Electronics best paper award, pp. 44-49, Busan, Korea, Nov. 2008.
- [I-9] Tetsuo Yokoyama (Nagoya Univ.), “Reversible computation and reversible programming languages,” In Preliminary Proceedings of Reversible Computation (RC), p.17, York, UK, March 22, 2009.
- [I-10] Tohru Ishihara (Kyushu Univ.), “Real-Time Dynamic Voltage Hopping on MPSoCs,” 9th International Forum on Embedded MPSoC and Multicore, Savannah, Georgia, USA, Aug. 2009.
- [I-11] 高田広章 (名古屋大学), “組込みシステムの消費エネルギー最適化,” 情報処理学会アルゴリズム研究会, 2009年11月.
- [I-12] Tohru Ishihara (Kyushu Univ.), “Real-Time Power Management for a Multi-Performance Processor,” International SoC Design Conference, Busan, Korea, Nov. 2009.
- [I-13] 石原亨 (九州大学), “組込みシステムの消費エネルギー見える化と省エネルギー技術,” LSI と システムのワークショップ 2010, 北九州, 2010年5月18日
- [I-14] Hiroaki Takada (Nagoya Univ.), “Challenges of Hard Real-Time Operating Systems - Multiprocessor Support and Energy Consumption Optimization -,” ARTIST Summer School Europe 2010, Autrans, France, Sep. 2010.
- [I-15] Hiroyuki Tomiyama (Ritsumeikan University), “Efficient Utilization of Scratch-Pad Memory in Preemptive Multi-Task Systems,” Workshop on Compiler-Assisted System-On-Chip Assembly (CASA) in conjunction with Embedded Systems Week (ESWEEK), Scottsdale, AZ, USA, Oct. 2010.
- [I-16] L. Gauthier and T. Ishihara (Kyushu Univ.), “Compiler Assisted Energy Reduction Techniques for Embedded Multimedia Processors,” Asia-Pacific Signal and Information Processing Association (APSIPA) Annual Summit and Conference 2010, Biopolis, Singapore, Dec. 2010.

②口頭発表 (国内会議 50 件、国際会議 42 件)

1. 発表者(所属)、タイトル、学会名、場所、月日

- [C-1] 今井陽平, 本田晋也, 富山宏之, 高田広章 (名古屋大学), “メモリ配置最適化によるウェイ予測キャッシュの低消費電力化,” 電子情報通信学会 VLD/ICD 研究会, 沖縄, 2006年3月.
- [C-2] 李東勲, 石原亨, 室山真徳, 安浦寛人 (九州大学), Farzan Fallah (米富士通研), “マイクロプロセッサのエネルギー消費特性抽出とソフトウェアデバッガを用いた消費エネルギー見積もり,” 組込技術とネットワークに関するワークショップ ETNET2006, 2006年3月.
- [C-3] S. Yamaguchi, M. Muroyama, T. Ishihara, H. Yasuura (Kyushu Univ.), “Exploiting Narrow Bitwidth Operations for Low Power Embedded Software Design”, Workshop on Synthesis And System Integration of Mixed Information Technologies, 2006. 04.
- [C-4] T. Sato (Kyushu Univ.) and A. Chiyonobu (Kyushu Institute of Tech.), “Multiple Clustered Core Processors,” Proc. of 13th Workshop on Synthesis and System Integration of Mixed Information Technologies, pp.262-267, Nagoya, Apr., 2006.
- [C-5] 山口 誠一郎, 室山 真徳, 石原 亨, 安浦 寛人 (九州大学), “オペランドのビット

幅を考慮したソフトウェアレベル消費エネルギー削減手法,” 電子情報通信学会技術研究報告, Vol.106, No.31, pp.13-18, 松山, 2006年5月.

- [C-6] S. Yamaguchi, M. Muroyama, T. Ishihara, and H. Yasuura (Kyushu Univ.), “Exploiting Narrow Bitwidth Operations for Low Power Embedded Software Optimization,” Proc. of the International PhD Student Workshop on SOC, Taiwan, Jul. 2006.
- [C-7] D. Lee, T. Ishihara, M. Muroyama, H. Yasuura (Kyushu Univ.) and F. FALLAH (Fujitsu Lab. of America), “An Energy Characterization Framework for Software-Based Embedded Systems”, IEEE Workshop on Embedded Systems for Real-Time Multimedia, 2006.10.
- [C-8] 関子純平, 富山宏之, 高田広章(名古屋大学), 井上弘士(九州大学), “Drowsy キャッシュにおけるモード切替アルゴリズムの評価,” 情報処理学会 ARC 研究会, 2006年11月.
- [C-9] 曾剛, 富山宏之, 高田広章(名古屋大学), “Power Management for Idle Time in the Presence of Periodic Interrupt Services,” 情報処理学会 ARC/EMB 研究会, 2007年1月.
- [C-10] M. Goudarzi, T. Ishihara and H. Yasuura (Kyushu Univ.), “A Software Technique to Improve Yield of Processor Chips in Presence of Ultra-Leaky SRAM Cells Caused by Process Variation”, Asia and South Pacific Design Automation Conference, Yokohama, Japan, Jan., 2007.
- [C-11] M. Goudarzi, T. Ishihara and H. Yasuura (Kyushu Univ.), “Ultra-Leaky SRAM Cells Caused by Process Variation: Detection and Leakage Suppression at System-Level”, International CSI Computer Conference, Tehran, Iran, Feb., 2007.
- [C-12] 松村忠幸, 室山真徳, 石原亨, 安浦寛人(九州大学), “プロセスばらつきを考慮した SRAM セルの低消費電力設計手法”, 電子情報通信学会総合大会, AS-3-1, 名古屋, 2007年3月.
- [C-13] 石飛百合子, 石原亨, 室山真徳, 安浦寛人(九州大学), “プロセッサベース組込みシステムの省エネルギー化を目的としたコード配置手法”, 電子情報通信学会総合大会, AS-3-2, 名古屋, 2007年3月.
- [C-14] 山口誠一郎, 室山真徳, 石原亨, 安浦寛人(九州大学), “プロセッサベース組込みシステムの低消費電力化を目的としたクロックゲーティング命令に関する検討”, 電子情報通信学会総合大会, A-3-18, 名古屋, 2007年3月.
- [C-15] T. Sato (Kyushu Univ.) and Y. Kunitake (Kyushu Institute of Tech.), “A Simple Flip-Flop Circuit for Typical-Case Designs for DFM,” Proc. of International Symposium on Quality Electronic Design, pp.539-545, San Jose, USA, Mar. 2007.
- [C-16] 佐藤 寿倫 (九州大学), “カナリア・フリップフロップを利用する省電力マイクロプロセッサの評価”, 先進的計算基盤システムシンポジウム(SACSIS), pp.227-234, 2007年5月.
- [C-17] 佐藤 寿倫 (九州大学), 国武 勇次 (九州工業大学), “カナリア・フリップフロップを利用するDVS方式の改良”, 情報処理学会研究報告, 2007-ARC-173, Vol.2007, No.55, pp.43-48, 2007年5月.
- [C-18] G. Zeng, H. Tomiyama, and H. Takada (名古屋大学), “Power Optimization for Embedded System Idle Time in the Presence of Periodic Interrupt Services,” In Proc. of International Embedded Systems Symposium (IESS), Springer IFIP 231, pp. 241-254, Irvine, CA, USA, May-June 2007.
- [C-19] 石飛 百合子, 石原 亨, 安浦 寛人 (九州大学), “スクラッチパッドメモリを考慮したコード配置最適化による組込みシステムの低消費エネルギー化”, 情報処理学会 DA シンポジウム, pp.85-90, 2007年8月.



- [C-20] T. Sato (九州大学) and Y. Kunitake (九州工業大学), "Exploiting Input Variations for Energy Reduction", 17th International Workshop on Power and Timing Modeling, Optimization and Simulation, pp.384-393, Sep. 2007.
- [C-21] 石飛 百合子, 石原 亨, 安浦 寛人 (九州大学), "組込みシステムの低消費エネルギー化を目的としたコード配置最適化とその ILP モデルの提案", 電気関係学会九州支部連合大会, pp.206, 2007年9月.
- [C-22] 山口 誠一郎, 室山 真徳, 石原 亨, 安浦 寛人 (九州大学), "小容量オンチップメモリの有効利用による組込みシステムの低消費エネルギー化", 電気関係学会九州支部連合大会講演論文集, pp.207, 2007年9月.
- [C-23] M. Goudarzi and T. Ishihara (九州大学), "Variation-Aware Instruction Rescheduling: Reducing Instruction Cache Leakage in Processor-based Embedded Systems," 電気関係学会九州支部連合大会, page. 11-2P-05, 2007年9月.
- [C-24] Y. Ishitobi, T. Ishihara, and H. Yasuura (九州大学), "Code placement for Reducing the Energy Consumption of Embedded Processors with Scratchpad and Cache Memories", Proc. of 2007 IEEE/ACM/IFIP Workshop on Embedded Systems for Real-Time Multimedia, pp.13-18, Oct. 2007.
- [C-25] M. Goudarzi, T. Matsumura, and T. Ishihara (九州大学), "Taking Advantage of Within-Die Delay-Variation to Reduce Cache Leakage Power Using Additional Cache-Ways", International Workshop on Dependable Embedded Systems (WDES), Oct. 2007.
- [C-26] T. Matsumura, Y. Ishitobi, T. Ishihara, M. Goudarzi, and H. Yasuura (九州大学), "A Hybrid Memory Architecture for Low Power Embedded System Design", Proceedings of the Workshop on Synthesis And System Integration of Mixed Information Technologies, pp.56-62, Oct. 2007.
- [C-27] T. Sato (九州大学) and Y. Kunitake (九州工業大学), "Critical Issues Regarding A Variation Resilient Flip-Flop", 14th Workshop on Synthesis and System Integration of Mixed Information Technologies, pp.280-286, Oct. 2007.
- [C-28] 松村 忠幸, 石飛 百合子, 石原 亨, 安浦 寛人 (九州大学), "コード配置変更によるハイブリッドローカルメモリの消費エネルギー最小化", 情報処理学会研究報告, 2007-SLDM-131, pp.25-30, Oct. 2007.
- [C-29] S. Watanabe, A. Chiyonobu (九州工業大学) and T. Sato (九州大学), "Indirect Tag Search Mechanism for Instruction Window Energy Reduction", 7th International Conference on Computer and Information Technology, pp.841-846, Oct. 2007.
- [C-30] G. Zeng, H. Tomiyama, and H. Takada (名古屋大学), "A Generalized Framework for Energy and Performance Tradeoff," 組込みシステムシンポジウム(ESS)論文集, pp. 74-81, 2007年10月.
- [C-31] 高瀬英希, 曾剛, 富山宏之, 高田広章 (名古屋大学), "リーク電力を考慮したスクラッチパッドメモリの有効性の評価," 組込みシステムシンポジウム(ESS)論文集, pp. 82-89, 2007年10月.
- [C-32] 石飛 百合子, 石原 亨, 安浦 寛人 (九州大学), "組込みプロセッサのエネルギー消費を最小化するコード配置最適化問題の ILP モデル", 電気情報通信学会技術研究報告, VLD2007-75, Vol.107, No.334, pp.31-36, Nov. 2007.
- [C-33] 山口 誠一郎, 室山 真徳, 石原 亨, 安浦 寛人 (九州大学), "マルチタスク組込みアプリケーションの低消費エネルギー化のためのメモリ管理技術", 電子情報通信学会技術研究報告, Vol.107, No.334, pp.25-29, Nov. 2007.
- [C-34] 川島祐崇, 中村一博, 高木一義, 高木直史 (名古屋大学), "部分積加算における信号遷移回数の削減による配列型乗算器の低消費エネルギー化設計," 情報処理学会研究報告, システム LSI 設計技術, 2007-SLDM-132, pp.~103-108, Nov. 2007.

- [C-35] G. Zeng, H. Tomiyama, and H. Takada (名古屋大学), “A Dynamic Algorithm for Energy Savings in DEPS Framework,” 情報処理学会 EMB 研究会, 2007 年 12 月.
- [C-36] G. Zeng, H. Tomiyama, and H. Takada (名古屋大学), “A Software Framework for Energy and Performance Tradeoff in Fixed-Priority Hard Real-Time Embedded Systems,” In Proc. of International Conference on Embedded and Ubiquitous Computing (EUC), Springer Lecture Notes in Computer Science (LNCS) 4808, pp. 13-24, Taipei, Taiwan, Dec. 2007.
- [C-37] J. Zushi, G. Zeng, H. Tomiyama, H. Takada (名古屋大学), and K. Inoue (九州大学), “Improved Policies for Drowsy Caches in Embedded Processors,” In Proc. of International Symposium on Electronic Design, Test and Applications (DELTA), pp. 362-367, Hong Kong, China, Jan. 2008.
- [C-38] M. Goudarzi, T. Ishihara, H. Noori (九州大学), “Variation-Aware Software Techniques for Cache Leakage Reduction using Value-Dependence of SRAM Leakage due to Within-Die Process Variation,” Int’l Conf. on High Performance Embedded Architectures & Compilers (HiPEAC’08), LNCS 4917, pp. 224-239, Jan. 2008.
- [C-39] 山口誠一郎、大山裕一郎 (九州大学)、国武勇次 (九州工業大学)、松村忠幸、石飛百合子、山口聖貴、李東勲、金田裕介 (九州大学)、舟木敏正 (九州工業大学)、室山真徳、石原亨、佐藤寿倫 (九州大学)、“負荷変動に瞬時適応可能なマルチパフォーマンストップセッサの設計と評価”、情報処理学会研究報告, 2007-SLDM-134, 2008 年 3 月.
- [C-40] 高瀬英希, 富山宏之, 高田広章 (名古屋大学), “マルチタスク環境におけるスクラッチパッドメモリ領域活用法,” 組込技術とネットワークに関するワークショップ (ETNET), 2008 年 3 月.
- [C-41] S. Watanabe (九州工業大学) and T. Sato (九州大学), “Uncriticality-directed Low-power Instruction Scheduling”, IEEE Computer Society Annual Symposium on VLSI, pp.69-74, Apr. 2008.
- [C-42] Tetsuo Yokoyama (Nagoya Univ.), Holger Bock Axelsen and Robert Glueck (Univ. of Copenhagen), “Principles of a Reversible Programming Language,” ACM International Conference on Computing Frontiers, May 2008.
- [C-43] 高瀬英希, 富山宏之, 高田広章 (名古屋大学), “プリエンティブなマルチタスク環境におけるスクラッチパッドメモリ領域分割法,” 情報処理学会 EMB/SE 研究会, 2008 年 6 月.
- [C-44] T. Ishihara, S. Yamaguchi, Y. Ishitobi, T. Matsumura, Y. Kunitake, Y. Oyama, Y. Kaneda, M. Muroyama and T. Sato (Kyushu Univ.), “AMPLE: An Adaptive Multi-Performance Processor for Low-Energy Embedded Applications,” IEEE Symposium on Application Specific Processors, pp.83-88, Jun. 2008.
- [C-45] Tetsuo Yokoyama (Nagoya Univ.), Holger Bock Axelsen and Robert Glueck (Univ. of Copenhagen), “Reversible Flowchart Languages and the Structured Reversible Program Theorem,” International Colloquium on Automata, Languages and Programming, July 2008.
- [C-46] H. Takase, H. Tomiyama, G. Zeng, and H. Takada (Nagoya Univ.), “Energy Efficiency of Scratch-Pad Memory at 65 nm and Below: An Empirical Study,” In Proc. of International Conference on Embedded Software and Systems (ICCESS), pp. 93-97, Chengdu, China, July 2008.
- [C-47] M. Goudarzi and T. Ishihara (Kyushu Univ.), “Row/Column Redundancy to Reduce SRAM Leakage in Presence of Random Within-Die Delay Variation,” International Symposium on Low Power Electronics and Design (ISLPED), pp.93-98, Aug. 2008.
- [C-48] 松村忠幸、石原亨、安浦寛人 (九州大学)、“コード配置とメモリ構成の同時最適化による省電力化手法,” DA シンポジウム 2008 論文集, pp13-18, 2008 年 8 月.

- [C-49] 石飛百合子、石原亨、安浦寛人 (九州大学) 、 “キャッシュウェイ割当てとコード配置最適化による組込みプロセッサの省エネルギー化,” DA シンポジウム 2008 論文集, pp19-24, 2008 年 8 月.
- [C-50] 山口誠一郎、石飛百合子、室山真徳、石原亨、安浦寛人 (九州大学) 、 “組込みシステムの消費エネルギー削減を 目的とした不均一キャッシュアーキテクチャ,” DA シンポジウム 2008 論文集, pp25-30, 2008 年 8 月.
- [C-51] 横山 哲郎 (名古屋大学), Holger Bock Axelsen, Robert Glueck (Univ. of Copenhagen). 単射関数のクリーン可逆シミュレーション (Working Paper). 日本ソフトウェア科学会第 25 回大会, 2008 年 9 月.
- [C-52] 横山 哲郎, 今井 敬吾, 曾 剛, 富山 宏之, 高田 広章, 結縁 祥治 (名古屋大学), 動的電圧制御システムにおけるエネルギー効率的なプログラムの導出, 第 71 回情報処理学会プログラミング研究会, 2008 年 10 月.
- [C-53] Gang Zeng, Tetsuo Yokoyama, Hiroyuki Tomiyama, Hiroaki Takada (名古屋大学), “Practical Energy-Aware Rate Monotonic Task Scheduling for DVS-Enabled Multiprocessor,” 組込みシステムシンポジウム (ESS), pp. 23-30, 2008 年 10 月.
- [C-54] 横山 哲郎, 曾 剛, 富山 宏之, 高田 広章 (名古屋大学). DVS システムにおけるアルゴリズムレベルのエネルギー消費の解析と最適化, 組込みシステムシンポジウム (ESS), pp. 31-40, 2008 年 10 月.
- [C-55] S. Yamaguchi, T. Ishihara, H. Yasuura, “A Single Cycle Accessible Two-Level Cache Architecture for Reducing the Energy Consumption of Embedded Systems,” In Proc. of International SoC Design Conference (ISOCC), Busan, Korea, Nov. 2008.
- [C-56] G. Zeng, T. Yokoyama, H. Tomiyama, H. Takada (Nagoya Univ.), and T. Ishihara (Kyushu Univ.), “A Generalized Framework for System-Wide Energy Savings in Hard Real-Time Embedded System,” In Proc. of International Conference on Embedded and Ubiquitous Computing (EUC), pp. 206-213, Shanghai, China, Dec. 2008.
- [C-57] T. Yokoyama, G. Zeng, H. Tomiyama, and H. Takada (Nagoya Univ.), “Analyzing and Optimizing Energy Efficiency of Algorithms on DVS Systems: A First Step towards Algorithmic Energy Minimization,” In Proc. of Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 727-732, Yokohama, Japan, Jan. 2009.
- [C-58] 石飛百合子、石原亨、安浦寛人 (九州大学) 、 “Non-uniform Selective Way Cache の動的制御による組込みプロセッサの省エネルギー化,” 電子情報通信学会技術研究報告, Vol.108, No. 463, pp.13-18, 2009 年 3 月.
- [C-59] 山口誠一郎、石原亨、安浦寛人 (九州大学) 、 “シングルサイクルアクセス可能な二階層キャッシュアーキテクチャ,” 電子情報通信学会技術研究報告, Vol. 108, No. 463, pp. 19-24, 2009 年 3 月.
- [C-60] 立松知紘, 横山哲郎, 菊地武彦, 富山宏之, 高田広章 (名古屋大学), “組込みシステムのタスク内 DVFS のための実行トレースマイニング,” 電子情報通信学会 VLD 研究会, 2009 年 3 月.
- [C-61] T. Sato, S. Watanabe, “Uncriticality-directed Scheduling for Tackling Variation and Power Challenges,” 10th International Symposium on Quality Electronic Design, March 2009.
- [C-62] Tomoo Yokoyama (Aoyama Gakuin Univ.), Tetsuo Yokoyama (Nagoya Univ.), “Functoriality in Reversible Circuits (work in progress),” In Preliminary Proceedings of Reversible Computation (RC), pp. 68-72, York, UK, March 22, 2009.
- [C-63] H. Takase, H. Tomiyama, and H. Takada (Nagoya Univ.), “Allocation of Scratch-Pad Memory in Multi-Task Systems,” In Proc. of International Symposium on VLSI Design, Automation and Test (VLSI-DAT), pp. 68-71, Hsinchu, Taiwan, Apr. 2009.

- [C-64] T. Yokoyama, G. Zeng, H. Tomiyama, and H. Takada (Nagoya Univ.), “Heuristics for Static Voltage Scheduling Algorithms on Battery-Powered DVS Systems,” In Proc. of International Conference on Embedded Software and Systems (ICCESS), pp. 265-272, Hangzhou, Zhejiang, China, May 2009.
- [C-65] Gang Zeng (名古屋大学), Shinpei Kato (東京大学), Tetsuo Yokoyama, Hiroyuki Tomiyama, and Hiroaki Takada (名古屋大学), “Task Migration for Energy Savings in Multiprocessor Real-Time Systems,” 電子情報通信学会 VLD 研究会, 2009 年 5 月.
- [C-66] 高瀬英希, 富山宏之, 高田広章(名古屋大学), “組込みリアルタイムシステムにおけるスラッシュパッドメモリ管理技術,” 情報処理学会 SLDM 研究会, 2009 年 5 月.
- [C-67] G. Zeng, T. Yokoyama, H. Tomiyama, and H. Takada (Nagoya Univ.), “Practical Energy-Aware Scheduling for Real-Time Multiprocessor Systems,” In Proc. of International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), pp. 383-392, Beijing, China, Aug. 2009.
- [C-68] 奥平 拓見, 石原 亨, 井上 弘士(九州大学), “ゲーテッドフリップフロップの多ビット統合によるレジスタ回路の低消費電力化,” DA シンポジウム論文集, pp. 25-30, Aug. 2009.
- [C-69] 安東 孝信, 中里 竜, 池田 信之, 高田 沙都子, 深谷 哲司(東芝), “ユーザ満足度を損ねずにエネルギー消費を極小化させる要求分析手法”, 情報科学技術フォーラム, 東北工業大学(仙台), 2009 年 09 月
- [C-70] Lovic Gauthier, Tohru Ishihara, “Optimal stack frame placement and transfer for energy reduction targeting embedded processors with scratch-pad memories,” In Proc. of IEEE Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia), pp. 116-125, Nov. 2009.
- [C-71] 川島裕崇, 高木直史(名古屋大学), “オペランドの和を用いた並列乗算器の消費エネルギー評価”, 信学技報, VLSI 設計技術研究会, 2009 年 12 月.
- [C-72] 安東 孝信, 中里 竜, 深谷 哲司(東芝), “ユーザ満足度を損ねずに省エネルギーを目指す要求適正化手法”, 情報処理学会ウィンターワークショップ, 倉敷市芸文館(倉敷), 2010 年 01 月.
- [C-73] Takumi Okuhira, Tohru Ishihara (Kyushu Univ.), “Unification of Multiple Gated Flip-Flops for Saving the Power Consumption of Register Circuits,” In Proc. of International Conference on Embedded Systems and Intelligent Technology (ICESIT), Feb. 2010.
- [C-74] Hideki Takase, Hiroyuki Tomiyama, and Hiroaki Takada (Nagoya Univ.), “Partitioning and Allocation of Scratch-Pad Memory for Priority-Based Preemptive Multi-Task Systems,” In Proc. of Design Automation and Test in Europe (DATE), pp. 1124-1129, Dresden, Germany, Mar. 2010.
- [C-75] 川島裕崇, 中村一博, 高木一義, 高木直史 (名古屋大学), “セル遅延モデルを用いた算術演算回路の信号遷移回数見積もり手法,” 電子情報通信学会 VLD 研究会, 2010 年 3 月.
- [C-76] C. Zang and T. Ishihara (Kyushu Univ.), “An Implementation of Energy Efficient Multi-Performance Processor for Real-Time Applications,” in Proc. of International Conference on Green Circuits and Systems, pp. 211-216, June, 2010.
- [C-77] 立松知紘, 高瀬英希, 曾剛 (名古屋大学), 富山宏之, 高田広章 (名古屋大学), “実行トレースマイニングを用いたタスク内 DVFS に有効なチェックポイント抽出手法,” 情報処理学会 EMB 研究会, 函館, 2010 年 8 月.
- [C-78] 三輪遼平, 高瀬英希, 曾剛 (名古屋大学), 富山宏之 (立命館大学), 高田広章 (名古屋大学), “組込みシステムにおける低消費エネルギー志向の効率的なスラック時間の導出,” 情報処理学会 EMB 研究会, 函館, 2010 年 8 月.

- [C-79] 中里竜, 安東孝信, 深谷哲司(東芝), “ユーザ満足度を損ねずに省エネルギーを目指す要求適正化手法 ～ 開発コストを考慮した省エネルギー仕様選択 ～”, 電子情報通信学会 SS 研究会, 旭川, 2010 年 8 月.
- [C-80] L. Gauthier, T. Ishihara (Kyushu Univ.), and H. Takada (Nagoya Univ.), “Stack Frames Placement in Scratch-Pad Memory for Energy Reduction of Multi-task Applications,” DA シンポジウム論文集, pp.171-176, 豊橋, 2010 年 9 月.
- [C-81] Lovic Gauthier, Tohru Ishihara (Kyushu Univ.), Hideki Takase (Nagoya Univ.), Hiroyuki Tomiyama (Ritsumeikan Univ.), Hiroaki Takada (Nagoya Univ.), “Placing Static and Stack Data into a Scratch-Pad Memory for Reducing the Energy Consumption of Multi-task Applications,” In Proc. of Workshop on Synthesis and System Integration of Mixed Information Technologies (SASIMI), pp. 7-12, Taipei, Taiwan, Oct. 2010.
- [C-82] T. Okuhira and T. Ishihara (Kyushu Univ.), “A Power Efficient Unified Gated Flip-Flop,” In Proc. of Workshop on Synthesis and System Integration of Mixed Information Technologies (SASIMI), pp. 424-429, Taipei, Taiwan, Oct. 2010.
- [C-83] Lovic Gauthier, Tohru Ishihara (Kyushu Univ.), Hideki Takase (Nagoya Univ.), Hiroyuki Tomiyama (Ritsumeikan Univ.), Hiroaki Takada (Nagoya Univ.), “Minimizing Inter-Task Interferences in Scratch-Pad Memory Usage for Reducing the Energy Consumption of Multi-Task Systems,” In Proc. of International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), pp. 157-166, Scottsdale, AZ, USA, October 2010.
- [C-84] Yoshimi Otsuka (Fukuoka Univ.), Toshinori Sato (Fukuoka Univ. / Kyushu Univ.), Takahito Yoshiki, Takanori Hayashida (Fukuoka Univ.), MultiCore Energy Reduction Utilizing Canary FF, 10th International Symposium on Communications and Information Technologies, pp.922-927, October 2010.
- [C-85] Tomohiro Tatematsu, Hideki Takase, Gang Zeng (Nagoya Univ.), Hiroyuki Tomiyama (Ritsumeikan Univ.), Hiroaki Takada (Nagoya Univ.), “Checkpoints Extraction Using Execution Traces for Intra-Task DVFS in Embedded Systems,” In Proc. of International Symposium on Electronic Design, Test and Applications (DELTA), pp. 19-24, Queenstown, New Zealand, Jan. 2011.
- [C-86] 中里竜, 安東孝信, 深谷哲司(東芝), “ユーザ満足度を損ねずに省エネルギーを目指す要求適正化手法”, 情報処理学会ウィンターワークショップ, pp.33-34, ラフォーレ修善寺(伊豆市), 2011 年 01 月.
- [C-87] Takanobu Ando, Ryu Nakazato, Tetsuji Fukaya (Toshiba), “Requirement Analysis for Reducing Energy Consumption without Degrading User Satisfaction”, In Proc. of The Tenth IASTED International Conference on Software Engineering, pp. 33-40, Innsbruck, Austria, Feb 2011.
- [C-88] 吉木崇人(福岡大学), 佐藤寿倫(九州大学), 林田隆則(福岡大学), タイミングエラー予報 FF を利用するマルチコアプロセッサのパワーマネージメント, 情報処理学会九州支部 火の国情報シンポジウム, CD-ROM, 6 pages, 2011 年 3 月.
- [C-89] 川島裕崇, 曾剛, 渥美紀寿, 立松知紘, 高田広章(名古屋大学), “DEPS フレームワークにおける最悪実行時間と平均消費エネルギーのタスク内解析手法,” 電子情報通信学会 VLD 研究会, 沖縄, 2011 年 3 月.
- [C-90] Hideki Takase, Gang Zeng, Hiroataka Kawashima, Noritoshi Atsumi, Tomohiro Tatematsu(名古屋大学), Lovic Gauthier, Tohru Ishihara(九州大学), Yoshitake Kobayashi, Shunitsu Kohara, Takenori Koshiro(東芝), Hiroyuki Tomiyama(立命館大学), Hiroaki Takada(名古屋大学), “An Energy Optimization Framework for Embedded Applications,” 電子情報通信学会 CPSY/DC/情報処理学会 SLDM/EMB 研究会(組込み技術とネットワークに関するワークショップ), 宮古島, 2011 年 3 月.

- [C-91] 川島裕崇, 曾剛, 渥美紀寿, 立松知紘, 高瀬英希, 高田広章, “DEPS プロファイルの評価法とそれを利用したチェックポイント選定,” 電子情報通信学会 CPSY/DC/情報処理学会 SLDM/EMB 研究会(組込み技術とネットワークに関するワークショップ), 宮古島, 2011 年 3 月.
- [C-92] Toshinori Sato (九州大学), Takahito Yoshiki, and Takanori Hayashida (福岡大学), Multicore Power Management Utilizing Error-Predicting Flip-flop, 4th International Workshop on Multi-Core Computing Systems, to appear, June 2011.

③ポスター発表 (国内会議 4 件、国際会議 5 件)

1. 発表者(所属)、タイトル、学会名、場所、月日

- [P-1] 大山裕一郎、室山真徳、石原亨、佐藤寿倫、安浦寛人(九州大学)、“低消費エネルギーシステムのための適応型マルチパフォーマンスプロセッサ”、電子情報通信学会総合大会ポスターセッション、ISS-P-61、2007 年 3 月
- [P-2] Y. Oyama, T. Ishihara, T. Sato and H. Yasuura (九州大学), “A Multi-Performance Processor for Low Power Embedded Applications”, COOL Chips X IEEE Symposium on Low-Power and High-Speed Chips, Vol.1, pp.138, Apr. 2007.
- [P-3] 片岡寛貴 (名古屋大学), “除算に着目した消費電力最適な組み込みプロセッサ構成の決定手法,” 組込みシステム技術に関するサマーワークショップ, Aug. 2007.
- [P-4] 菊地武彦 (名古屋大学), “ソフトウェアとハードウェアの協調による組込みシステムの消費エネルギー最適化,” 組込みシステム技術に関するサマーワークショップ, Aug. 2007.
- [P-5] M. Goudarzi, T. Matsumura and T. Ishihara (Kyushu Univ.), “Cache Power Reduction in Presence of Within-Die Delay Variation using Spare Ways”, IEEE Annual Symposium on VLSI (ISVLSI), Apr. 2008.
- [P-6] M. Goudarzi and T. Ishihara (Kyushu Univ.), “Instruction Cache Leakage Reduction by Changing Register Operands and Using Asymmetric SRAM Cells”, Great Lakes Symposium on VLSI (GLSVLSI), May. 2008.
- [P-7] T. Matsumura, T. Ishihara and H. Yasuura (Kyushu Univ.), “Simultaneous optimization of memory configuration and code allocation for low power embedded systems”, ACM Great Lakes Symposium on VLSI (GLSVLSI), pp.403-406, May. 2008.
- [P-8] 高瀬英希(名古屋大学), 小原俊逸, 深谷哲司(東芝), Lovic Gauthier, 石原亨(九州大学), 富山宏之(立命館大学), 高田広章(名古屋大学), “ソフトウェアとハードウェアの協調による組込みシステムの消費エネルギー最適化,” ポスター発表, 第 12 回組込みシステム技術に関するサマーワークショップ (SWEST12), 豊橋, 2010 年 9 月.
- [P-9] T. Ishihara and L. Gauthier (Kyushu Univ.), “Compiler Assisted Energy Reduction Techniques for Embedded Processors” Poster Presentation at University Booth of DATE 2011, Grenoble, France, March 2011.

(4)知財出願

①国内出願 (1 件)

1. “マイクロプロセッサ制御装置ならびにその方法およびプログラム”、進 博正、株式会社東芝、2007 年 9 月 13 日、P2007-237818.

②海外出願 (0 件)

(5)受賞・報道等

## ①受賞

- [A-1] 2006 年 IEEE 福岡支部 学生研究奨励賞、李東勲、九州大学、“An Energy Characterization Framework for Software-Based Embedded Systems,” Proc. of the 2006 IEEE/ACM/IFIP Workshop on Embedded Systems for Real Time Multimedia (ESTIMedia 2006), pp.59-64.
- [A-2] 最優秀論文賞受賞、佐藤 寿倫、九州大学、“カナリア・フリップフロップを利用する省電力マイクロプロセッサの評価’ ”, 先進的計算基盤システムシンポジウム (SACSIS), pp.227-234, 2007 年 5 月.
- [A-3] IEEE 福岡支部 学生研究奨励賞、石飛百合子、九州大学、“Code placement for Reducing the Energy Consumption of Embedded Processors with Scratchpad and Cache Memories”, IEEE/ACM/IFIP Workshop on Embedded Systems for Real-Time Multimedia, pp.13-18, Oct. 2007.
- [A-4] 平成 19 年度連合大会講演奨励賞、Maziar Goudarzi、九州大学、“Variation-Aware Instruction Rescheduling: Reducing Instruction Cache Leakage in Processor-based Embedded Systems,” 電気関係学会九州支部連合大会, page11-2P-05, 2007 年 9 月.
- [A-5] LSI IP デザイン・アワード MeP 賞、山口誠一郎、大山裕一郎(九州大学)、国武勇次(九州工業大学)、松村忠幸、石飛百合子(九州大学)、山口聖貴、李東勲(九州大学)、舟木敏正(九州工業大学)、金田裕介、室山真徳、石原亨、佐藤寿倫(九州大学)、“計算負荷の変動に瞬時適応可能なマルチパフォーマンスプロセッサ”2008 年 4 月.
- [A-6] 平成 20 年度情報処理学会コンピュータサイエンス領域奨励賞、高瀬英希(名古屋大学)、“マルチタスク環境におけるスクラッチパッドメモリ領域活用法”, 2008 年 7 月 12 日.
- [A-7] 平成 20 年度情報処理学会 SLDM 研究会優秀論文賞、高瀬英希(名古屋大学)、“マルチタスク環境におけるスクラッチパッドメモリ領域活用法”, 2008 年 8 月 26 日.
- [A-8] 平成 20 年度情報処理学会 SLDM 研究会優秀発表学生賞、高瀬英希(名古屋大学)、“マルチタスク環境におけるスクラッチパッドメモリ領域活用法”, 2008 年 8 月 26 日.
- [A-9] Intel Asia Academic Forum 2008 Best Research Award, Tohru Ishihara, “Software and Hardware Approaches for Reducing the Energy Consumption of Embedded Systems”, Oct. 2008.
- [A-10] SoC Design Group of IEEE Best Paper Award, International SoC Design Conference (ISOCC), Maziar Goudarzi, Tohru Ishihara (Kyushu Univ.), Nov. 2008,
- [A-11] LG Electronics Co., Ltd. Best Paper Award, International SoC Design Conference (ISOCC), G. Zeng, T. Yokoyama, H. Tomiyama, H. Takada (Nagoya Univ.), and T. Ishihara (Kyushu Univ.), Nov. 2008
- [A-12] 石原亨, 平成 21 年度科学技術分野の文部科学大臣表彰 若手科学者賞, 2009 年 4 月 14 日
- [A-13] 高瀬英希, 第 140 回情報処理学会 SLDM 研究会優秀発表学生賞, “組込みリアルタイムシステムにおけるスクラッチパッドメモリ管理技術,” 2009 年 8 月 26 日.
- [A-14] 山口誠一郎, 第 140 回情報処理学会 SLDM 研究会優秀発表学生賞, “シングルサイクルアクセス可能な二階層キャッシュアーキテクチャ,” 2009 年 8 月 26 日
- [A-15] 高瀬英希, 平成 21 年度 IPSJ 論文船井若手奨励賞, 2010 年 3 月.
- [A-16] Lovic Gauthier, Tohru Ishihara (Kyushu University), Hideki Takase (Nagoya University), Hiroyuki Tomiyama (Ritsumeikan University), Hiroaki Takada (Nagoya University), “Placing Static and Stack Data into a Scratch-Pad Memory for Reducing the Energy Consumption of Multi-task Applications,” SASIMI 2010 Outstanding Paper Award, 2010 年 10 月.
- [A-17] 石原亨, 情報処理学会 長尾真記念特別賞, 2010 年 5 月

## (6)成果展開事例



### ①実用化に向けての展開

- マルチコア向け DEPS/コンパイラ技術の一部を発展させ、NEDO の公募研究「極低電力回路・システム技術開発(グリーン IT プロジェクト)」研究開発項目[7]「低消費電力メニーコア用アーキテクチャとコンパイラ技術」に採択され、現在実施中(代表は九州大学の井上弘土准教授。九州大学、立命館大学、電気通信大学、株式会社フィックスターズ、株式会社トプスシステムズとの共同研究)。
- マルチパフォーマンスプロセッサとその DEPS 制御技術のコンセプトを発展させ、内閣府の最先端・次世代研究開発支援プログラムの中で研究を実施中。研究タイトルは「環境エネルギーを使用する情報通信機器の組み込みプロセッサアーキテクチャと OS 制御による最適エネルギー管理技術の開発」。代表は九州大学の石原亨准教授(現在、京都大学准教授)。

### § 6 研究期間中の主なワークショップ、シンポジウム、アウトリーチ等の活動

年月日	名称	場所	参加人数	概要
2008 年 9 月 16、17 日	第 1 回名古屋大学組込みシステム研究センターシンポジウム	名古屋大学 (名古屋市)	346 名(延べ人数)	「ソフトウェアとハードウェアの協調による組み込みシステムの消費エネルギー最適化」という題目で講演(富山宏之)
2010 年 9 月 15、16 日	第 2 回名古屋大学組込みシステム研究センターシンポジウム	名古屋大学 (名古屋市)	287 名(延べ人数)	「ソフトウェアとハードウェアの協調による組み込みシステムの消費エネルギー最適化」という題目で講演(高瀬英希)

### § 7 結び

本プロジェクトでは、組み込みシステムを対象として、計算機システム(のプロセッサとメモリシステム)の消費エネルギーを 60 分の 1 に低減するための技術の研究開発に取り組んできた。§ 4 で述べたように、オフチップメモリまで含めた消費エネルギーを 60 分の 1 にするメドはつけられなかったが、オンチップ部分(プロセッサ部とオンチップメモリ)に関しては、60 分の 1 に低減する技術的なメドをつけることができた。

しかしながら、本プロジェクトの最も大きい意義は、DEPS(Dynamic Energy/Performance Scaling)という、計算機システムの消費エネルギー削減のために今後重要となるであろうコンセプトを提案し、そのコンセプトに基づいて、ハードウェア IP(マルチパフォーマンスプロセッサ)とソフトウェア開発環境、RTOS で構成される DEPS フレームワークを提案したことである。DEPS フレームワークは、本プロジェクトで研究開発した様々な技術を統合するものであり、このようなスケールの大きいフレームワークを提案することは、CREST のような大規模な研究プロジェクトでなければなしえなかったものだと考えている。

また、これまで消費エネルギーを意識していなかったソフトウェア開発者や研究者に、低消費エネ

ルギーアルゴリズムや QoS 要求適正化技術の提案によって、消費エネルギー削減に貢献する手段(手掛かり)を与えたことも、本プロジェクトの大きな貢献である。

オフチップメモリまで含めたシステムの消費エネルギーを 60 分の 1 にできなかった理由としては、次のような理由が考えられる。

- 本プロジェクトでは、オンチップ回路の消費エネルギー削減に努力してきたが、一方で、オフチップメモリ(具体的には SDRAM)の消費エネルギー削減に関する研究にはあまり力を入れてこなかった。DEPS に対応できる SDRAM を開発すれば、オフチップメモリの消費エネルギーに関しても、本プロジェクトの成果を活用して削減することができる。
- §4 の計算では、オフチップメモリの消費エネルギーをシステム全体の 25% と仮定しているが、実際には、この数値は、オンチップメモリのサイズによって大きく変化する。そのため、大きいサイズのオンチップメモリを使用することで、オフチップメモリの消費エネルギー比率が低下し、結果としてシステムの消費エネルギー削減率が大きくなる。

本プロジェクトの今後の展開として、まずは、やり残した数多くの研究テーマについて取り組むことが挙げられる。本プロジェクトでは、研究期間の前半で個別技術の研究に取り組み、後半ではそれらを統合する研究開発に注力した。後半の統合過程で、前半で研究した個別技術の不足点が数多く出てきたが、それに取り組む時間が取れず、数多くの研究テーマが残った。DEPS フレームワークができたことにより、これらの個別技術の研究には、(例えば科研費などを利用した)大学での一般的な研究スキームで取り組めるものと考えている。

それと並行して、本プロジェクトの開発成果を産業展開するための活動にも力を入れて取り組んでいく。具体的には、開発した RTOS は、完成度を上げた上で TOPPERS プロジェクトからオープンソース化する。ソフトウェア開発環境の産業展開については、組込みソフトウェアの開発ツールメーカーとの協業が不可欠で、すでに協議を開始しているが、ツールメーカー(中小企業がほとんど)の独自資金で開発することは難しく、研究成果を産業展開するための公的資金の申請を計画している。最大の課題は、DEPS に対応するハードウェアの開発である。これに関しては、かなり多額の資金を必要とするため、半導体メーカー等に本プロジェクトの成果の有用性をアピールする活動を続けていく必要がある。

CREST のスキームに対する要望としては、企業の積極的な参加が見込める形態にして欲しいと考えている。具体的には、CREST の研究資金により、企業に属する研究者の人件費が支払えないという制約は、本プロジェクト遂行上の大きな障害となった。特にソフトウェア技術に関する研究は、かかる経費の大部分が人件費であり、この制約により、企業に属する研究者が本プロジェクトに割ける時間が短くなってしまったことは否定できない。