

可変パイプライン構造による低消費電力技術を用いた 浮動小数点演算器の実装および評価

辻 拓郎[†] 佐々木 敬泰^{††} 市川 裕 二^{†††}
弘中 哲夫^{†††} 近藤 利夫^{††}

近年、プロセッサには高性能と低消費エネルギーの両立が求められている。しかし、現在広く用いられている DVS のような電源電圧に依存した低消費エネルギー手法は、将来的には電源電圧の低下による電源電圧変化幅の減少によってエネルギー削減効率が低下する。そこで我々は電源電圧に依存しないアーキテクチャレベルの低消費エネルギー手法として VSP を提案している。VSP は動的にパイプライン段数を変更して実行時間を削減すると同時に、LDS-cell と呼ぶ素子を用いてステージ統合時に増加する組合せ回路内のグリッチを緩和している。しかしこれまでの評価では整数演算 CPU のみを対象としており、FPU を搭載したより現実的なプロセッサでの評価は行われていなかった。そこで、本論文では VSP 技術を用いた FPU を設計し、評価を行った。その結果、整数演算 CPU では、VSP は DVS と比較して高性能かつ低消費エネルギーであったが、FPU では、VSP は DVS とほぼ同じ消費エネルギーで約 1.6 倍の性能であった。

Implementation and Evaluation of Low Energy Floating Point Unit by Variable Stages Pipeline Technique

TAKURO TSUJI,[†] TAKAHIRO SASAKI,^{††} YUJI ICHIKAWA,^{†††}
TETSUO HIRONAKA^{†††} and TOSHIO KONDO^{††}

Recently, in the field of mobile computing, the achievement of low energy computing and high performance computing is required simultaneously. DVS (Dynamic Voltage Scaling) is one current major technique to realize it. However, the lower the supply voltage becomes in the future, the less energy saving we get. So we propose VSP (Variable Stages Pipeline) technique which unifying pipeline stages in the use of a special D-flip flop cell called LDS-Cell which has an ability to act as a latch. We showed that the VSP processor can achieve lower energy computing and higher performance computing than the DVS processor. However, former evaluation was performed with only integer CPU. Therefore this paper designs the core modules of FPU with VSP technique, and evaluates its effectiveness. According to our evaluation results, the FPU with VSP technique can achieve 1.6 times higher performance with the almost same energy consumption as DVS.

1. はじめに

近年、モバイルコンピューティングの分野において低消費エネルギーと高性能の両立が要求されている。現在の代表的な低消費電力手法である DVS (Dynamic Voltage Scaling)¹⁾ は、動的に電源電圧と動作周波数を変化させることで消費エネルギーを削減する。し

かし、将来的な電源電圧の低下による電源電圧変化幅の減少によって消費エネルギーの削減効率が低下する。また、DVS は動作周波数の低下に比例して性能が低下する。それに対し、我々は電源電圧に依存しないアーキテクチャレベルの低消費エネルギー手法として VSP (Variable Stages Pipeline)^{2),3)} を提案している。VSP はパイプライン段数を動的に変化させることで、低消費エネルギー化を目指す手法である。VSP を整数演算 CPU に適用した場合、低消費電力モードにおいて、VSP は DVS の約 1/3 の消費エネルギーで約 2 倍の性能を得ることが可能である。また我々と同様にパイプライン段数を動的に変化させる手法の一つである PSU (Pipeline Stage Unification)⁴⁾⁵⁾⁶⁾ よ

[†] 三重大学大学院工学研究科

Graduate School of Engineering, Mie University

^{††} 三重大学工学部

Department of Engineering, Mie University

^{†††} 広島市立大学情報科学研究科情報工学専攻

Graduate School of Information Sciences, Hiroshima
City University

りも少ない消費エネルギーで同じ性能を得られた。

しかし、これまでの評価では整数演算 CPU のみを対象としており、浮動小数点演算ユニット (FPU) を搭載したより現実的なプロセッサでの評価は行われていなかった。FPU は整数演算 CPU に比べ、

- 利用頻度が時間的に偏る、
 - 命令の流れが整数系と大幅に異なる、
 - バス幅が大きい、
 - 低消費電力モード時に停止できる回路が少ない、
- などの違いがあるため、VSP 技術が FPU に有効であるか不明である。そこで本論文では、これまで評価に用いてきた整数演算 CPU のみの VSP プロセッサに付加できる FPU を設計し、従来手法のプロセッサと性能・消費エネルギーについて比較する。

本論文の構成は以下の通りである。まず第 2 節で関連研究について述べる。次に第 3 節で VSP と VSP 手法を用いたプロトタイププロセッサについて説明を行う。そして第 4 節で今回設計した FPU の詳細を説明し、第 5 節で評価し、第 6 節でまとめる。

2. 関連研究

本節では、プロセッサの低消費エネルギー手法に関連する研究について概括する。現在の主な低消費エネルギー手法として DVS が挙げられる。DVS は動的に電源電圧と動作周波数を制御することによって消費エネルギーを削減する。消費エネルギーは電源電圧の 2 乗に比例するため、電源電圧を低下させることは消費エネルギーを大きく削減することにつながる。しかし、LSI の電源電圧は年々低下しており、将来的に閾値電圧の制御などの問題から電源電圧を低下させることのできる変化幅は小さくなる。そのため DVS による消費エネルギーの削減効率は低下することになる。そこで、将来的に消費エネルギーの削減効率が低下する DVS に代わって、電源電圧に依存しないアーキテクチャレベルの低消費エネルギー手法が多く提案されている。アーキテクチャレベルの低消費エネルギー手法として、PSU や DPS (Dynamic Pipeline Scaling)⁷⁾ のように動的にパイプラインステージを変化させる手法がある。このプロセッサのパイプライン構造の例を図 1 に示す。PSU や DPS の手法は動作周波数に応じて動的にパイプラインステージを統合する。これによって以下の利点がある。

- 分岐ミスペナルティとデータ依存による待ちサイクルの削減によって実行時間を削減することが可能である。
- 使用しないパイプラインレジスタやユニットへの

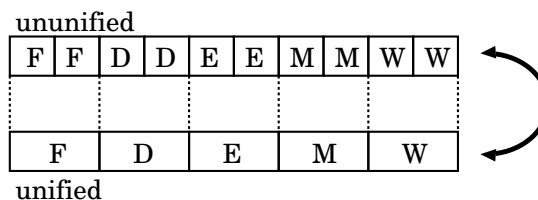


図 1 パイプライン段数の変更

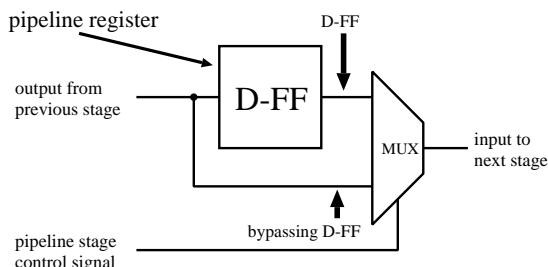


図 2 D-FF + MUX 型パイプラインレジスタの構成図

クロックを停止することでその部分の消費エネルギーを削減できる。

以上の利点によって低消費エネルギー化が可能である。また非同期のプロセッサについてパイプラインラッチコントローラを制御する⁸⁾ ことでパイプライン段数を削減し、低消費エネルギー化を実現することも可能である。従来の手法と VSP 手法の違いは VSP アーキテクチャにはパイプラインステージ統合時に増加する組合せ回路内のグリッチを緩和するセルを搭載していることにある。また従来の手法はプロセッサについて詳細な設計がなされていないが、VSP ではプロトタイププロセッサを詳細設計した上で、SPICE ネットリストを用いてトランジスタレベルの消費エネルギーの評価を行っている。PSU は図 2 に示すように D-FF と MUX によってパイプラインレジスタをバイパスすることでパイプラインステージを統合する。PSU には DVS を組み合わせたものも存在する。DVS と組み合わせることにより、より細かい電力制御が可能になる。VSP についても DVS と組み合わせることによりより細かい電力制御が可能になるが、本論文の主旨から外れるため今回は DVS との組合せについてはこれ以上議論しない。

3. VSP(Variable Stages Pipeline)

本節では VSP について述べる。VSP は PSU と同様、パイプライン段数を動的に変更することで低消費エネルギー化を目指すアーキテクチャであるが、PSU とは異なり単純にステージ統合を行うのではなく、統

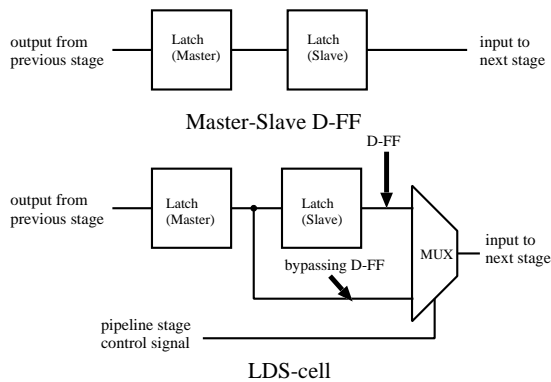


図 3 LDS-cell 型パイプラインレジスタの構成図

合により発生するグリッチの増加を抑制するための機能を有している。

3.1 Latch D-FF Selector - cell

この節では LDS-cell (Latch D-FF Selector - cell) について述べる。グリッチは組合せ回路内のゲートの遅延によって各ゲートに入力される信号の時刻がそろっていないために発生する。また、発生したグリッチによって新たなグリッチが発生するため、一般的に組合せ回路の規模が増大するとグリッチの量も増大する。VSP では統合されたパイプラインステージにラッチを挿入することでグリッチ増加を緩和する。ラッチを挿入する⁹⁾ ことでラッチ以前で発生したグリッチをラッチ以降に伝搬させず、ラッチ以降のデータ入力時刻を一致させることができるという効果が得られる。この効果によって統合されたパイプラインステージでのグリッチの緩和が可能になる。LDS-cell は 2 つのラッチと 1 つの MUX によって成されている。LDS-cell の構成を図 3 に示す。LDS-cell は図 2 の D-FF+MUX と同じトランジスタ数で実装可能である。

3.2 VSP プロセッサ

本論文で設計した FPU はこれまで評価に用いてきた整数演算 CPU のみの VSP プロセッサに付加できるものなので、本節ではまず整数演算 CPU について簡単に説明する。整数演算 CPU は MIPS R3000 と命令互換性を持つ。整数演算 CPU はパイプライン段数の違いによって High Speed (HS) モードと Low Energy (LE) モードの 2 つのモードを持つ。これら 2 つのモードの特徴は以下の通りである。

HS モード

- 9 段パイプラインである。
- デコードステージに gshare 分岐予測ユニット¹⁰⁾ を搭載しており、分岐ミスペナルティは 3 サイクルである。

- 無条件分岐は分岐予測ユニットにおいて 100% の分岐予測が可能である。
- インターロックと演算結果のフォワーディング機構を搭載している。

LE モード

- 3 段パイプラインであり、LDS-cell はグリッチの緩和を行うラッチとして動作する。
- HS モードの 1/4 の周波数で動作する。
- 分岐ペナルティやデータ依存によるインターロックが発生しない。
- 分岐予測ユニットやバイパスされて使用しなくなったパイプラインレジスタのクロックを止めることができる。

整数演算 CPU では HS モードにおいて DVS, PSU, VSP のサイクル数は同じになる。消費電力については PSU が少し大きいですが、VSP と DVS はほぼ同じ値となる。LE モードにおいて、VSP は DVS の約 1/3 の消費エネルギーで約 2 倍の性能を得ることが可能であり、PSU よりも少ない消費エネルギーで同じ性能を得ることができた。

4. FPU

前節より、VSP は低消費電力化に有効であることがわかる。しかし、これは整数演算 CPU のみを評価しているため、FPU を搭載したより現実的なプロセッサでも効果があるかどうか不明である。

整数演算 CPU と FPU では以下のような違いがある。すなわち、

- 利用頻度が時間的に偏っているため、低消費電力化の効果が不明、
- バス幅が大きいため、パイプラインを統合することによりグリッチが多く発生し、消費電力が増加する可能性がある、
- 整数演算部では分岐予測器やフォワーディング回路の一部など、低消費電力モード時に停止できる回路が存在するが、FPU にはそのような回路はない。そのため、回路全体の電流を低減できる DVS の方が有利になる可能性がある、

などである。そこで、本研究では、第一次評価として浮動小数点演算器の演算パイプライン部分を詳細設計し、性能及び消費電力評価を行う。

4.1 設計する FPU の概要

我々は VSP の評価基盤として、MIPS R3000 の命令互換プロセッサを設計し、評価を行ってきた。そこで、上記プロセッサと結合して利用できるように、R3000 用浮動小数点演算ユニットである R3010 互換の浮動

表 1 サイクル数の違い

	本論文	R3010
ADD/SUB	4(1)	2(2)
MUL	4(1)	4(2)
DIV	12(1)	12(3)

小数点演算ユニット (FPU) を設計する。FPU により実行される基本演算は以下の通りである。

- FPU レジスタとメモリとの間のロード/ストア命令
- FPU と CPU レジスタ間の転送
- 浮動小数点加算, 減算, 乗算, 除算, および変換命令などの算術演算
- 浮動小数点比較

この演算の内, 本論文で実装を行った FPU とオリジナルの R3010 FPU との違いは, 浮動小数点加減算命令, 乗算命令, 除算命令の実行に必要なサイクル数と, 排他的なアクセスを必要とするサイクル数である。排他的なアクセスを必要とするサイクル数とは, パイプラインに命令を投入した後, 次の命令を投入することができないサイクル数のことである。本論文で実装した FPU は完全にパイプライン化されているので排他的なアクセスを必要とするサイクル数は 1 である。以下にそれをまとめた表 1 を示す。表 1 の括弧内の数字は, 排他的なアクセスを必要とするサイクル数を表す。

4.2 FPU の実装

本節では, FPU の詳細設計について述べる。本研究では, VSP 手法を用いた MIPS R3010 命令互換浮動小数点ユニットを Verilog-HDL を用いて詳細設計を行う。浮動小数点ユニットは文献 11) の実装を採用した。また, 性能比較のため, DVS 方式, PSU 方式の浮動小数点ユニットも設計する。設計した FPU の詳細なデータを表 3 に, FPU 加減算器を図 4 に, FPU 乗算器を図 5 に, FPU 除算器を図 6 に示す。また, 実装予定である FPU 全体の構成を図 7 に示す。

5. 評価

本節では前節で詳細設計を行った VSP を用いた FPU の評価を行う。

5.1 消費エネルギー

本研究の目的は低消費エネルギーと高性能の両立である。そこで本節では消費エネルギーと性能について定義を行う。まず性能はアプリケーションの実行時間を指標とする。今回議論するのはパイプラインプロセッサであるため, アプリケーション実行のためのサイクル数にクロック周期を掛けることで実行時間を求

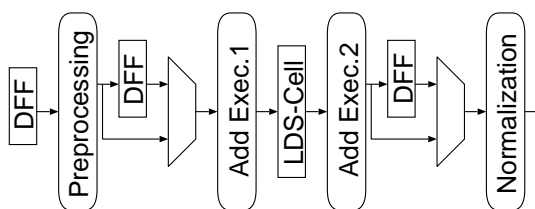


図 4 FPU 加減算器

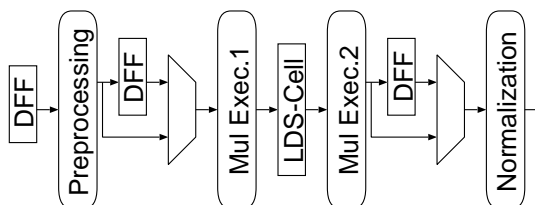


図 5 FPU 乗算器

めることができる。次に消費エネルギーはアプリケーションを実行するために必要な消費エネルギーと定義する。アプリケーションを実行していない間の消費エネルギーは議論しない。CMOS で構成されたプロセッサの消費エネルギー E は消費電力 P 実行時間 T を用いて以下の式によって求められる。

$$E = P \times T = St \times C \times V^2 \times G \times cycle \quad (1)$$

St はゲートのスイッチング確率, C は容量 (ゲート容量, 配線容量を含む), V は電源電圧, G はゲート数, $cycle$ はアプリケーションを実行するサイクル数である。式 (1) から分かるように消費エネルギーは動作周波数に依存しない。これが消費エネルギーと消費電力の違いである。本研究ではなるべく性能を低下させずにアプリケーションを実行するために必要なエネルギーを削減することが目的である。

5.2 評価方法

FPU の評価には姫野ベンチマークを用いた。姫野ベンチマークは非圧縮流体解析コードの性能評価を行うために開発されたもので, ポアソン方程式をヤコビの反復法で解く場合の主要なループの処理速度を計るものである。このベンチマークは浮動小数点の加減算と乗算のみを使用しているため, 除算の評価は行っていない。また今回は, 命令デコーダが未実装であるためプログラムをそのまま Verilog-HDL シミュレータ上で実行することができない。そのため, SimpleScalar を拡張して, クロックサイクル毎の浮動小数点命令およびオペランドを出力するよう変更した。そして, そのデータを NanoSim 用の入力ファイル形式に変更して消費電力見積もりを行った。設計及び評価に用いた環境を表 2 に示す。

設計した FPU は Design Compiler で Hitachi

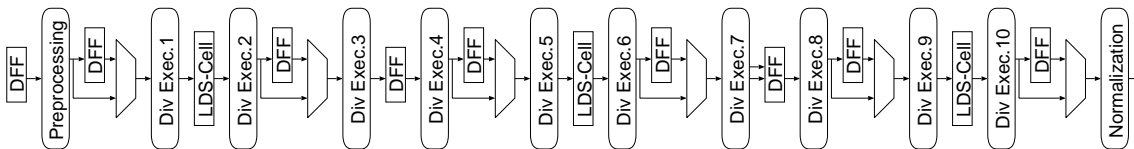


図 6 FPU 除算器

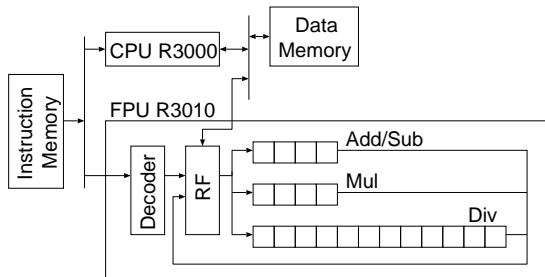


図 7 FPU 全体

表 2 設計・評価環境

作業工程	メーカー	ツール名
論理シミュレーション	Synopsys	VCS 7.0
論理合成	Synopsys	Design Compiler 2003.03
消費電力見積もり	Synopsys	NanoSim 2003.03
トレースデータ作成	-	SimpleScalar 3.0

0.18 μ m CMOS テクノロジーを用いて論理合成した。論理合成によって得られた演算器のトランジスタ数と評価環境をまとめて表 3 に示す。演算器名の下のは演算器のトランジスタ数である。それぞれの演算器の消費エネルギーは NanoSim を用いて測定した。また、使用しないパイプラインレジスタのためのクロック信号は停止させた。

電源電圧について Hitachi 0.18 μ m CMOS テクノロジーを用いているため 1.80V と設定した。DVS の LE モードの電源電圧については、文献 4) において、スケールリング則を用いて予測される 22nm プロセスでの電源電圧の割合を 0.18 μ m プロセスに適応して 1.44V とした。

5.3 評価結果

各モードにおける実行時間と消費エネルギーの評価結果を図 8、図 9 に示す。それぞれ DVS の値を 1.0 とした時の比である。図 8、図 9 より HS モードにおいて DVS, PSU, VSP のサイクル数、消費電力共にほぼ同じ値となるのがわかる。また、LE モードにおいては、PSU と VSP のサイクル数は DVS の 64% まで削減されるが、VSP の消費電力は DVS とほぼ同じ値となった。一方、PSU においては消費電力が DVS より増加している。これらの実行結果から LE モード

表 3 FPU の詳細

	演算	mode	stage [段]	frequency [MHz]	voltage [V]
DVS	加減算 20281 Tr	HS	4	80	1.80
		LE	4	20	1.44
	乗算 102727 Tr	HS	4	80	1.80
		LE	4	20	1.44
	除算 155630 Tr	HS	12	80	1.80
		LE	12	20	1.44
PSU VSP	加減算 22560 Tr	HS	4	80	1.80
		LE	1	20	1.80
	乗算 105335 Tr	HS	4	80	1.80
		LE	1	20	1.80
	除算 165920 Tr	HS	12	80	1.80
		LE	3	20	1.80

において、VSP は DVS とほぼ同じ消費エネルギーで約 1.6 倍の性能を得ることが可能であると言える。この結果を文献 3) で行った整数演算 CPU と比較する。整数演算 CPU では、サイクル数、消費エネルギー共に VSP も PSU も DVS より大幅に減少した。これに対し、FPU ではサイクル数は VSP も PSU も同程度 DVS より減少しているが、消費エネルギーは VSP は DVS とあまり変わらず、PSU は DVS より増加している。これは、パイプラインを統合したことによる低消費電力効果よりも、グリッチの増加による電力消費の方が大きくなったためと考えられる。つまり、FPU は整数演算 CPU と比較して 1 ステージ内の回路が複雑で大きいため、グリッチによる電力消費の影響がより強く現れたものと考えられる。

6. 結論と今後の展望

本稿では、電源電圧に依存しない低消費エネルギー手法として提案している VSP を FPU に適用した。その結果、低消費エネルギーモードにおいて、VSP は DVS とほぼ同じ消費エネルギーで約 1.6 倍の性能を得ることが可能であり、PSU より少ない消費エネルギーで同じ性能を得ることができた。

今後の展望としては、(1) 浮動小数点命令用デコーダを設計し、整数演算部と結合してプロセッサ全体の性能/消費電力評価を行う、(2) 姫野ベンチマーク以外のベンチマークプログラムを用いた評価、(3) 今回は単精度のみを扱ったので整数演算 CPU と同じバス幅

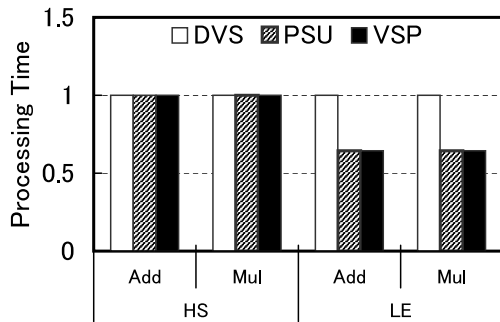


図 8 LE モードサイクル数比

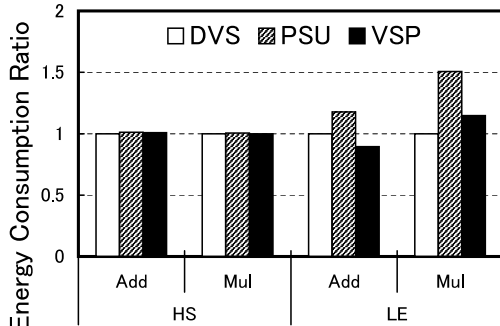


図 9 消費エネルギー比

だったので、よりグリッチの影響が深刻化すると考えられる倍精度への拡張を行う、(4) 非パイプライン部への DVS 手法の適用、等である。

謝辞 本研究は東京大学大規模集積システム設計教育研究センターを通じ、シノプシス株式会社ならびに株式会社日立製作所の協力で行われたものである。また、本研究で実装に用いた浮動小数点演算器を提供して頂いた京都大学大学院 情報学研究科 越智裕之助教授に謝意を表す。

参 考 文 献

- 1) J. Pouwelse, K. Langendoen, and H. Sips: "Dynamic voltage scaling on a low-power microprocessor", 7th ACM Int. Conf. on Mobile Computing and Networking (Mobicom), pp.251-259, July 2001.
- 2) Yuji Ichikawa, Takahiro Sasaki, Tetsuo Hironaka, Toshiaki Kitamura, and Toshio Kondo: "Low Energy Consumption by a Variable Stages Pipeline Technique," International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC2004), pp.6C1L-4-1-4.
- 3) 市川裕二, 佐々木 敬泰, 弘中 哲夫, 北村 俊明, 近藤 利夫: "可変パイプライン手法によるプロトタイプ低消費エネルギープロセッサの設計", 電

- 子情報通信学会技術報告, CPSY2004-21, No.241, pp.7-12, 2004.
- 4) 嶋田創, 安藤秀樹, 島田俊夫: "パイプラインステージ統合: 将来のモバイルプロセッサのための消費エネルギー削減技術", 2003 年先進的計算基盤システムシンポジウム SACSIS 2003, pp.283-290, 2003.
- 5) Hajime Shimada, Hideki Ando, and Toshio Shimada: "Pipeline Stage Unification: A Low-Energy Consumption Technique for Future Mobile Processors", The International Symposium on Low Power Electronics and Design 2003, pp.326-329, August 2003.
- 6) 嶋田創, 安藤秀樹, 島田俊夫: "パイプラインステージ統合とダイナミック・ボルテージ・スケールリングを併用したハイブリッド消費電力削減機構", 2004 年先進的計算基盤システムシンポジウム SACSIS 2004, pp.11-18, 2004.
- 7) Koppanalil, J., Ramrakhyan, P., Desai, S., Vaidyanathan, A. and Rotenberg, E.: "A Case for Dynamic Pipeline Scaling", Proc. of Int. Conf. on Compilers, Architecture, and Synthesis for Embedded Systems 2002, pp.1-8, 2002.
- 8) Efthymiou, A. and Garside, J. D.: "Adaptive Pipeline Depth Control for Processor Power-Management", Proc. of Int. Conf. on Computer Design 2002, pp. 454-457, 2002.
- 9) Enric Musoll, Jordi Cortadella: "Low-Power Array Multipliers with Transition-Retaining Barriers", Proc. International Workshop on Power, Timing Modeling Optimization and Simulation, pp.227-238, October 1995.
- 10) 吉瀬謙二, 片桐孝洋, 本多弘樹, 弓場敏嗣: "高性能プロセッサのための代表的な分岐予測の実装と評価", Technical Report UEC-IS-2003-2, 電気通信大学 大学院情報システム学研究科, 2003.
- 11) 越智裕之, 鈴木達也, 松永静香, 河野陽一, 津田孝夫: "IEEE754 準拠単精度浮動小数点除算器 IP ライブラリの開発", 電子情報通信学会技術報告, VLD2002-82, No.476, pp.205-210, 2004.