

パイプラインステージ統合によるプロセッサの消費エネルギーの削減

嶋田 創† 安藤 秀樹† 島田 俊夫†

近年のモバイルプロセッサにおいては、低消費エネルギーと高性能の両立が求められている。これらの要求に応えるため、現在では、DVS (dynamic voltage scaling) と呼ばれる方式が導入されている。DVS は、低クロック周波数での動作時に、電源電圧を低下させ、消費エネルギーを削減する。DVS は現在では有効な方式であるが、将来のプロセス技術においては、電源電圧の変動範囲が縮小し、有効性が低下する。これに対して我々は、低クロック周波数での動作時に、電源電圧を最大値に保ったまま複数のパイプラインステージを統合するパイプラインステージ統合 (PSU: pipeline stage unification) と呼ぶ方式を提案する。現在および将来のプロセス世代における DVS と PSU の効果を比較した結果、現在において、PSU は DVS に対して 11~14%程度消費エネルギーを削減できることが分かった。さらに、将来では DVS は大幅にその効果を落とすのに対し、PSU はその効果を維持し、その結果、約 10 年後には PSU は DVS に対して 27~34%と大きく消費エネルギーを削減できることが分かった。

Reducing Processor Energy Consumption with Pipeline Stage Unification

HAJIME SHIMADA,[†] HIDEKI ANDO[†] and TOSHIO SHIMADA[†]

Recent mobile processors are required to exhibit low-energy consumption as well as high performance. To satisfy these requirements, a method called dynamic voltage scaling or DVS is currently employed. DVS reduces energy consumption by decreasing the supply voltage when a processor runs at a low clock frequency. Although DVS is an effective method for reducing energy consumption, its effectiveness will be limited in future process generations because the variable supply voltage range will become small. As an alternative, we propose a method called pipeline stage unification or PSU, which unifies multiple pipeline stages when the processor runs at a low clock frequency, leaving the supply voltage at its maximum level. We compared PSU to DVS in terms of their effectiveness in current and future process generations. Our evaluations show that currently PSU reduces energy consumption only moderately (11-14%) more than DVS. Furthermore, in the future, DVS will significantly decrease its effectiveness, whereas PSU will maintain its effectiveness. As a result, PSU will reduce energy consumption by 27-34% more than DVS after about 10 years.

1. はじめに

近年のモバイルプロセッサでは、低消費エネルギーと高性能の両方が要求されている。この要求を満たすために、現在 DVS (dynamic voltage scaling) と呼ばれる方式が導入されている (たとえば、Transmeta Crusoe の LongRun^{1),2)}, Intel Mobile Pentium III,4 の SpeedStep^{3)~5)}, AMD Mobile Athlon 4 の PowerNow!⁶⁾). DVS はバッテリー持続時間要求やプロセッサ負荷に応じて、動的にクロック周波数と電源電圧を変更するものである。バッテリー持続時間要求が強い、与えられた負荷が低ければ、クロック周波数を低下

せ、消費電力を削減する。さらに、延びたクロックサイクル時間に信号の遅延を合わせ、電源電圧を低下させる。これにより、プログラム実行に要する消費エネルギーを削減する。

このように、DVS は消費エネルギーを削減する有効な手法であるが、プロセス技術の進歩に応じて将来、この有効性は減少すると我々は考える。その理由は以下の 2 点である。まず第 1 に、プロセス技術の進歩に応じて、電源電圧とトランジスタの閾値電圧はこれまでスケールアップされてきた。しかし、一般に閾値電圧のスケールアップは電源電圧のスケールアップに比べて緩やかである。加えて、今後、閾値電圧を大きく低下させると、サブスレッショルドリーク電流が劇的に増加するため、閾値電圧のスケールアップ速度は、さらに緩やかになると考えられる。トランジスタが正しく動作

† 名古屋大学大学院工学研究科
Graduate School of Engineering, Nagoya University

する電源電圧は閾値電圧によって下限が与えられるため、DVSにおける電源電圧の可変の範囲（正確には、最大の電源電圧に対する可変の範囲の割合）は減少する。第2に、今後のプロセス技術においては、DVSが過渡故障をより増加させる点である。先に述べたようにプロセス技術の進歩に応じて、電源電圧は下げられていくが、今後は、過渡故障が大きな問題となるレベルにまで下げられていく⁷⁾。過渡故障は電源電圧の低下に従って増加するため、電源電圧をより低下させるDVSでは、この問題は悪化する。このため、DVSによる電源電圧の低下は制限される。これら2つの点より、DVSの消費エネルギーの削減の要となる、電源電圧の可変の範囲は減少するため、DVSの有効性は減少することになる。

これに対して、我々はパイプラインステージ統合 (PSU: pipeline stage unification) と呼ぶ手法を提案する^{8)~10)}。PSUは単純だが効果的に消費エネルギーを削減することができる。PSUではDVSと同様に、プロセッサの消費エネルギーを削減するためにクロック周波数を低下させるが、DVSと異なり、電源電圧を低下させるのではなく、パイプラインレジスタをバイパスさせることによって複数のパイプラインステージを統合する。PSUによって消費エネルギーを削減できる理由は次の2点である。まず第1に、バイパスされるパイプラインレジスタへのクロックの供給を止めることにより、クロックドライバの総負荷を減少させることができる。これにより、消費電力が削減される。第2に、パイプラインステージの統合によりプロセッサのパイプラインが短くなる。これにより、プログラムの実行に必要なサイクル数が削減され、電力を消費する時間を短くすることができる。たとえば、フロントエンドパイプラインを短縮させることにより、分岐予測ミスペナルティは削減され、実行サイクル数は削減される。

残りの論文の構成は以下のとおりである。2章ではPSUの実装について述べ、3章ではDVSとPSUによる消費エネルギーの削減について解析的に分析する。4章では評価における仮定について説明し、5章で評価結果を提示し、その考察を行う。6章では関連研究について述べ、最後に、7章でこの論文についてまとめる。

2. PSUの実装

本章では、PSUの実装について述べる。2.1節でPSUを実現するための基本的な回路の変更を示す。PSUの大部分は、この節で説明する方法で実装でき

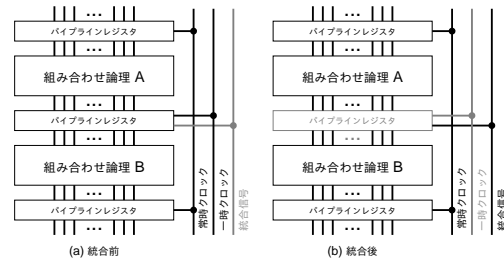


図1 クロック分配線の変更と統合信号の追加

Fig.1 Addition of clock distribution networks and unification signal.

るが、例外的に取り扱わなければならない部分が存在する。2.2節ではこれについて述べる。最後に、2.3節では、さらなるエネルギー削減のためのPSUとDVSの併用について述べる。

2.1 PSUのための回路

図1にPSUに関連する信号線とパイプラインレジスタとの結線関係を示す。説明を簡単にするために、2ステージの統合を例としている。図1に示すように、常時クロック、一時クロック、統合信号と呼ぶ3本の信号線が存在する。まず、クロックは統合時の動作が異なる常時クロックと一時クロックの2系統に分割される。常時クロックは統合をするかしないかにかかわらず、つねに動作しているクロックである。一方、一時クロックは統合を行ったときには動作を停止させるクロックであり、統合を行わないときのみ動作する。動作時の周波数と位相は常時クロックと同一である。統合信号は統合を指示する信号である。

図1(a)はステージを統合していない状態を、図1(b)は統合した状態を示す。図中の黒い部分は動作、あるいは、アサートされている部分を示し、灰色の部分は動作していない、あるいは、アサートされていない部分を示す。図1(a)に示すように、通常のパイプラインでは、隣接する組合せ論理回路AとBは、それらの回路の間のパイプラインレジスタが動作しているため、異なったステージとして動作する。一方、図1(b)ではこのパイプラインレジスタは動作せず、2つの組合せ論理回路は1つのステージとして動作する。

ここで、パイプラインレジスタには、フロントエンド、実行コア、バックエンド間にある命令ウィンドウなどデカップリング用の記憶素子を含まないことを注意しておく。これらは、パイプラインレジスタと同じくステージをつなぐ記憶素子であるが、複数の命令の状態を記憶する機能が必要なので、バイパスさせることはできない。

パイプラインレジスタをバイパスさせるには、2つ

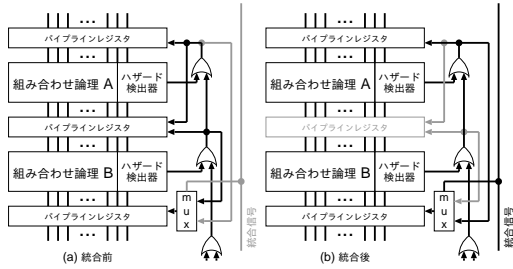


図2 インターロック回路
Fig. 2 Interlock logic.

の方法が考えられる。1つめの方法は、統合時にクロックとは無関係に信号が通過するようパイプラインレジスタの論理を構成することである。透過型のラッチでパイプラインレジスタを実現する場合、この論理の変更は容易である。2つめの方法は、パイプラインレジスタの後方にマルチプレクサを配置し、パイプラインレジスタの出力と前方のステージの出力を統合信号により選択する方法である。この方法は、パイプラインレジスタをどのような回路で構成しても適用可能である。

インターロック回路にもわずかな変更が必要である。図2に変更したインターロック回路を示す。図2(a)に統合していない状態での信号の流れを、図2(b)に統合した状態での信号の流れを示す。

一般に、あるステージのストール信号がアサートされるのは、そのステージでハザードを検出するか、または、後方のステージのストール信号が真のときである。図2に示した例では、 $S(A)$ と $H(A)$ を、それぞれ、ステージAのストール信号、ハザード信号とすると、 $S(A)$ は以下の式により与えられる：

$$S(A) = H(A) + S(B) \tag{1}$$

ステージAとBを統合する場合、統合後のステージのストール信号 $S(A+B)$ は、AまたはBがストールするときアサートされる必要があるので、以下のようになる：

$$S(A+B) = S(A) + S(B) \tag{2}$$

式(1)を式(2)に代入すると、

$$S(A+B) = H(A) + S(B) \tag{3}$$

となり、結局、 $S(A)$ と等しいことが分かる。

ところで、ストール信号によりステージを実際にストールさせるためには、そのステージの前方にあるパイプラインレジスタの更新を抑止し、後方のパイプラインレジスタへ渡す信号をNOPに変更する必要がある。組合せ回路Aの前方のパイプラインレジスタへ渡す信号は、統合前後で、それぞれ、 $S(A)$ 、 $S(A+B)$ なので、式(1)、(3)より変更の必要がないことが分か

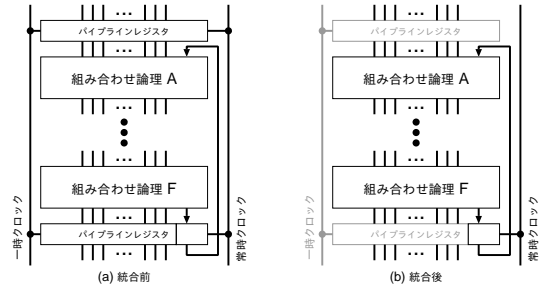


図3 ループをなす信号パス中のパイプラインレジスタ
Fig. 3 Pipeline register within a looped path.

る。一方、組合せ回路Bの後方のパイプラインレジスタへ渡す信号については、統合するかどうかに応じて $S(B)$ と $S(A+B)$ を切り替える必要がある。このためにマルチプレクサが必要である。

以上では2ステージ統合の場合についてのみ述べたが、一時クロック、統合信号、および、関連する回路を複数用意して適切に動作させることにより、さらに多くのステージの統合を行えるように、統合するステージ数に比例する程度の複雑さで拡張可能である。

2.2 ループをなす信号パスにおけるパイプラインレジスタ

前節で述べたように、統合するステージ間のパイプラインレジスタは基本的にバイパスさせるが、例外がある。ループをなす信号パスにおいて、後方のステージへ信号を出力するパイプラインレジスタは、統合するステージ間であってもバイパスさせない。図3に例を示す。図3(a)は、ステージFの出力がパイプラインレジスタを通り、後方のステージAに入力され、再びステージFに戻るといふ、ループをなす信号パスを持つパイプラインを示している。ステージFとその次のステージを統合する際、図3(b)に示すように、統合されるステージ間のパイプラインレジスタは基本的にバイパスさせるが、ループをなす信号パス上にある部分はバイパスさせない。パイプラインレジスタをバイパスさせないことにより、ループをなす信号パスは正しいタイミングを維持できる。このようなループをなす信号パスの例として、プログラムカウンタの読み出しと更新、命令ウィンドウやロードストアキューの割当てにおけるヘッド/テールポインタの更新、命令発行とそれによる命令のwakeup、命令の実行結果のフォワーディングなどがあげられる。

図4に、命令の実行結果のフォワーディングのタイミングを例として示す。単純なDLX型の5段のパイプライン¹¹⁾において、実行ステージ(Exec)とメモリアクセスステージ(Mem)を統合した場合を考

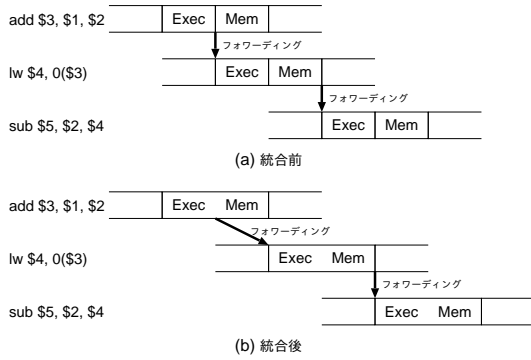


図4 実行結果のフォワーディングのタイミング
Fig.4 Timing of result forwarding.

える。Exec と Mem ステージの間のパイプラインレジスタは基本的にバイパスされるが、実行結果のフォワーディングの部分ではバイパスされない。図4 (a) および (b) は、それぞれ統合前および統合後のタイミングを示している。図4 (b) に示すように、Exec ステージで得られた add 命令の結果は、バイパスさせない Exec と Mem ステージ間のパイプラインレジスタで保持され、次のサイクルに、正しいタイミングで Exec ステージの lw 命令に渡される。一方、Mem ステージで得られる lw 命令の結果は、通常のタイミングで Exec ステージの sub 命令に渡される。統合後も、add 命令のレイテンシは1サイクルであるが、lw 命令のレイテンシは統合により、1サイクルに短縮されることに注意されたい。

2.3 PSU と DVS の組合せ

PSU は f_{max} を最大のクロック周波数、 $U (\geq 2)$ を統合されるステージ数としたとき、 f_{max}/U の点 (以下、スイッチングポイントと呼ぶ) ごとに段階的に消費エネルギーを削減する。このことは、PSU は DVS と異なり、クロック周波数のスケールングに対して連続的にエネルギー消費を削減することはできないことを意味している。この欠点を補うためには、PSU と DVS を組み合わせればよい。つまり、各スイッチングポイントではステージを統合し、スイッチングポイントの間は DVS を適用する。より具体的には、まず、電源電圧とクロック周波数を、最大値から最初のスイッチングポイント ($f_{max}/2$) まで減少させていく。次に、電源電圧を最大値に戻し、2 ステージを統合する。同様に、次のスイッチングポイント ($f_{max}/3$) まで電源電圧は最大値から減少させていく。このプロセスをクロック周波数を減少させている間、繰り返す。クロック周波数を上昇させるときはこの反対を行う。5章で述べるが、スイッチングポイントにおいては、PSU は

DVS より多くの消費エネルギーを削減できる。したがって、PSU に DVS を組み合わせれば、すべての周波数において、PSU は DVS より多くの消費エネルギーを削減できる。

なお、本論文では、PSU と DVS を組み合わせた場合の評価は行っていない。

3. 消費エネルギー削減量の解析

本章では、DVS と PSU による消費エネルギーの削減について解析的に説明する。一般に、プログラムの実行の際に消費するエネルギー E は以下の式で表される：

$$E = P \times T_{ex} \quad (4)$$

ここで、 P は消費電力、 T_{ex} は実行時間である。 P と T_{ex} は以下の式で与えられる：

$$P = a \times f \times C \times V_{DD}^2 \quad (5)$$

$$T_{ex} = \frac{N}{IPC \times f} \quad (6)$$

ここで、 f はクロック周波数、 C はスイッチするノードの全容量、 a はアクティビティファクタ、つまり、各ノードの平均スイッチング確率、 N は実行命令数、 IPC は1サイクルあたりの実行命令数の平均である。

3.1 DVS による消費エネルギー削減

式 (5) と (6) を式 (4) に代入し、以下に示す消費エネルギー $E(f, V_{DD})$ が得られる：

$$E(f, V_{DD}) = \frac{a \times N \times C \times V_{DD}^2}{IPC} \quad (7)$$

通常モードでは最大のクロック周波数 f_{max} と最大の電源電圧 V_{DDmax} で動作し、低消費電力モードでは f_{low} と V_{DDlow} で動作する DVS プロセッサを考える。通常モードの消費エネルギーで正規化した低消費電力モードにおける消費エネルギーは以下の式となる：

$$E_{DVS,n}(f_{low}, V_{DDlow}) = \frac{IPC_{max}}{IPC_{low}} \times \left(\frac{V_{DDlow}}{V_{DDmax}} \right)^2 \quad (8)$$

ここで、 IPC_{max} と IPC_{low} はそれぞれ通常モードと低消費電力モードでの IPC である。プロセッサ以外のデバイスも、プロセッサのクロック周波数の低下に応じて動作速度を低下させるという単純な仮定をおくと、 IPC_{low} は IPC_{max} に等しい。よって、次の式が導かれる：

$$E_{DVS,n}(f_{low}, V_{DDlow}) = \left(\frac{V_{DDlow}}{V_{DDmax}} \right)^2 \quad (9)$$

上式より、消費エネルギーの削減は電源電圧の低下によってのみ達成されることが分かる。また、電源電圧の低下によるエネルギーの削減は、電源電圧の比の

2乗の関数となっていることが分かる。2乗の関数のため、消費エネルギーは劇的に削減される。同時に、式(9)は、将来のプロセス技術ではDVSの効果は急速に減少することを意味している。将来のプロセス技術では、 V_{DDlow}/V_{DDmax} は現在のプロセス技術ほど小さくできない。なぜなら、トランジスタが確実に動作するためには、 V_{DDlow} は、閾値電圧の2倍以上が必要であるが¹²⁾、1章で述べたように、閾値電圧は電源電圧より緩やかにしかスケールされないからである。DVSの電源電圧の可変範囲の比を大きくするために、閾値電圧をより低下させるという選択肢も考えられる。しかし、閾値電圧を低下させるとリーク電流が劇的に増加するため、この選択肢は得策ではない。さらに、1章で述べたように、今後のプロセス技術においては、DVSによる電源電圧の低下は、通常よりも過渡故障を増加させることになる。この過渡故障への対策という面からも電源電圧を低下させることは制限され、DVSの効果は減少する。

3.2 PSUによる消費エネルギー削減

2章で述べたように、PSUは一時クロックのドライバを停止することによって消費電力を削減する。 U ステージ統合(以下、これを統合度 U と呼ぶ)でプロセッサが動作しているとき、理想的にはクロックドライバが消費する電力は $1/U$ となる。また、通常のプロセッサと同じく、総消費電力はクロック周波数の低下率に比例して削減する。したがって、クロック周波数 f_{low} で動作する、統合度 U のPSUプロセッサの消費電力は以下の式で表される：

$$P_{PSU}(f_{low}, U) = \left(P_{total} - P_{clock} + \frac{P_{clock}}{U} \right) \times \frac{f_{low}}{f_{max}} \quad (10)$$

ここで、 P_{total} と P_{clock} はそれぞれ、通常モードにおけるプロセッサの総消費電力とクロックドライバの消費電力である。

式(4)を用い、通常モードで正規化した消費エネルギーは以下の式で表される：

$$E_{PSU, n}(f_{low}, U) = \frac{P_{PSU}(f_{low}, U) \times T_{ex}(f_{low}, U)}{P_{total} \times T_{ex}(f_{max}, 1)} \quad (11)$$

ここで、 $T_{ex}(f, U)$ はクロック周波数 f と統合度 U における実行時間である。 $T_{ex}(f_{max}, 1)$ は通常モードにおける実行時間であることを注意されたい。式(6)と(10)を式(11)に代入することにより、以下の式を得ることができる：

表1 ベンチマーク

Table 1 Benchmarks.

ベンチマーク	入力	実行命令数
compress95	bigtest.in	95M
gcc	genoutput.i	84M
go	2stone9.in	75M
ijpeg	specmun.ppm	450M
li	train.lsp	183M
m88ksim	ct.in	100M
perl	scrabbl.in	80M
vortex	vortex.in	80M

$$E_{PSU, n}(f_{low}, U) = \frac{IPC_{max}}{IPC_{low}} \times \left\{ 1 - k \times \left(1 - \frac{1}{U} \right) \right\} \quad (12)$$

ただし、

$$k = \frac{P_{clock}}{P_{total}} \quad (13)$$

である。

式(12)から分かるように、消費エネルギーはIPCの向上に反比例する(パイプラインが短縮されるため、 $IPC_{max} < IPC_{low}$ であることに注意)。また、消費エネルギーの削減は、クロックドライバにより消費される電力が、プロセッサの総消費電力のどれだけの割合を占めているかということに依存する。この割合は、近年の高速なプロセッサでは非常に大きく(たとえば、Alpha 21264では32%¹³⁾)、この傾向は深いパイプラインや小さいクロックスキューなどの達成のために将来も続き、PSUは消費エネルギーを大きく削減できると期待できる。

4. 評価環境

3章の式(12)から分かるように、PSUの消費エネルギーにはIPCと統合度 U とクロックドライバの消費電力の割合 k が関係する。本章では最初にIPCの評価環境を示す。次に、具体的にパイプラインを仮定し、どのように統合するか述べる。最後に、クロックドライバの消費電力の割合 k に関する仮定について説明する。

4.1 シミュレーション環境

SimpleScalar Tool Set¹⁴⁾中のout-of-order実行シミュレータを用いて、パイプラインのステージ数を変化させ、IPCを測定した。命令セットはSimpleScalar PISAである。表1に示すように、ベンチマークプログラムとして、SPECint95の8本を用いた。ベンチマークプログラムのバイナリはgcc ver.2.7.2.3を用い、-O6 -funroll-loopsのオプションでコンパイルし作成した。シミュレーション時間が過大にならないよ

表 2 プロセッサの構成
Table 2 Processor configuration.

命令発行幅		8
RUU		64 エントリ
LSQ		32 エントリ
メモリポート		8
機能ユニット	整数 ALU	8
	整数乗除算	4
	浮動小数点 ALU	8
	浮動小数点乗除算	4
分岐予測	予測方式	gshare
	履歴	6 ビット
	インデックス	13 ビット
	BTB	2048 エントリ/4-way
キャッシュ	RAS	16 エントリ
	L1 命令	64KB/32B ライン/1-way
	L1 データ	64KB/32B ライン/1-way
メモリ	L2 統合	2MB/64B ライン/4-way
	初期参照レイテンシ	64
TLB	バースト転送間隔	2
	命令	16 エントリ/4-way
	データ	32 エントリ/4-way
ミスレイテンシ		128

うにするために、命令ミックス、関数の出現頻度など、特徴をほぼ維持しつつ、それぞれのベンチマークプログラムへの入力を調整した。

表 2 に、シミュレーションにおいて仮定したプロセッサの構成を示す。プロセッサは近年のプロセッサと同様に、深いパイプラインを持つと仮定した。メモリアクセス時間は、プロセッサのクロック周波数の低下に比例して遅くなると仮定した。そのため、プロセッサのクロック周波数によらず、メモリアクセスのサイクル数は一定である。

4.2 パイプラインの仮定

図 5 (a) に示すように、評価においては、Pentium 4¹⁵⁾ のパイプラインとほぼ等しいパイプラインをベースとした。図中の各ステージの動作については、以下のとおりである。

- NextPC : 分岐予測による次の PC の決定
- Fetch : 命令キャッシュからの命令フェッチ
- Drive1 : フェッチした命令のデコーダへの転送
- Alloc : RUU (register update unit) , LSQ (load/store queue) の割当て
- Rename : レジスタリネーミング
- Queue : RUU への書き込み
- Schedule : 命令スケジューリング
- Dispatch : RUU からの発行
- Register : レジスタ読み出し
- Exec : 命令の実行
- Flags : フラグの書き込み

- BrnChk : 分岐命令の実行結果と分岐予測の比較
- Drive2 : 分岐結果のフロントエンドへの転送
- Cache : データキャッシュアクセス
- Writeback : RUU へのライトバック
- Commit : 実行結果のコミット

図 5 (b) と図 5 (c) にそれぞれ 2 ステージおよび 4 ステージの統合したパイプラインを示す。表 3 に、これらのパイプラインにおける命令の実行レイテンシ、分岐予測ミスペナルティ、キャッシュヒットレイテンシを示す。表の 2 行目に示すように、1, 2, 4 ステージを統合した場合、それぞれ最大クロック周波数の 100%, 50%, 25% で動作する。

なお、整数/浮動小数点除算と平方根演算については、同一資源を繰り返し使用し完全なパイプライン化はされておらず、ステージの統合はできないと仮定した。レイテンシはそれぞれ、20, 12, 24 サイクルとした。

4.3 クロックの消費電力の割合の仮定

プロセッサの総消費電力に対するクロックの消費電力の割合 k は、プロセッサの設計によって異なる。文献 13), 16)~18) によれば、その範囲は 18%~40% である。そこで、5 章の評価においては、特に断わりのない限り、これらの値のほぼ中央値である 30% と仮定した。また、クロックドライバが消費する電力は、駆動するパイプラインレジスタ数に比例すると仮定し、単純に統合度 U に反比例するとした。この仮定はおおまかではあるが、以下の理由で我々の評価にお

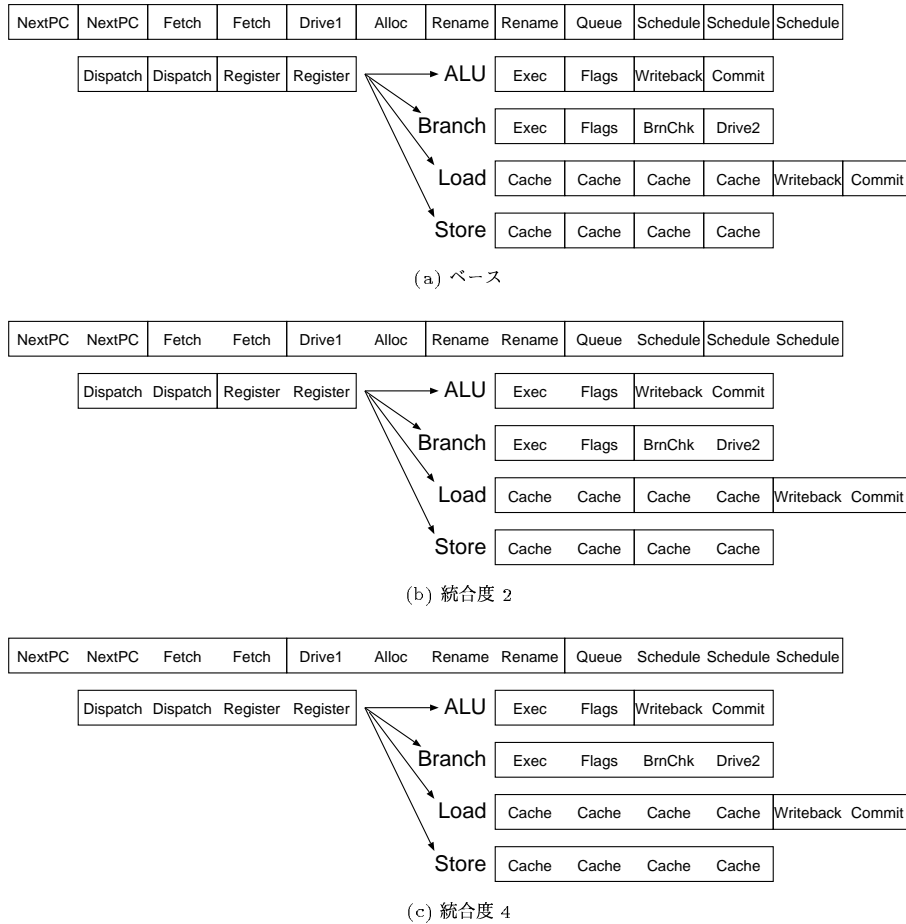


図 5 仮定する PSU のパイプライン
Fig.5 Assumed PSU pipeline.

表 3 PSU において仮定する実行レイテンシ, キャッシュヒットレイテンシ, 分岐予測ミスペナルティ
Table 3 Assumptions of execution latencies, cache hit latencies, and branch misprediction penalty in PSU processor.

統合されるステージ数		1	2	4
クロック周波数		100%	50%	25%
実行レイテンシ	整数乗算	3	2	1
	浮動小数点 ALU	2	1	1
	浮動小数点乗算	4	2	1
L1 キャッシュヒットレイテンシ		4	2	1
L2 キャッシュヒットレイテンシ		16	8	4
分岐予測ミスペナルティ		20	10	5

いては妥当であると考ええる。一般に、クロックは多段のネットワークにより分配するが、クロックドライバの消費電力のほとんどは、最終段のドライバで消費される（たとえば、Intel Itanium 2 の階層化クロックネットワークの場合、88%を最終段のドライバが消費している¹⁶⁾）。また、最終段のドライバの負荷は、およそ、そのファンアウトであるパイプラインレジスタ

の数に比例すると考えられる。

5. 評価結果

本章では、まず、現在の 180 nm のプロセス技術におけるエネルギー削減の評価結果を示す。次に、プロセッサの総消費電力に対するクロックが消費する電力の割合の違いにより、消費エネルギー削減量がどのよ

表 4 180 nm プロセス技術における DVS と PSU のクロック周波数と電源電圧の仮定

Table 4 Assumptions of clock frequencies and supply voltages for DVS and PSU in the 180 nm technology.

クロック周波数	電源電圧	
	DVS	PSU
100%	1.65 V	1.65 V
50%	1.35 V	1.65 V
25%	1.10 V	1.65 V

うに変化するかを示す。最後に、将来のプロセス技術における DVS と PSU の消費エネルギーの比較結果を示す。

5.1 現在のプロセス技術における消費エネルギーの比較

本節では、現在の 180 nm のプロセス技術における PSU による IPC の向上を評価し、次に PSU と DVS の消費エネルギーを比較する。表 4 に評価におけるクロック周波数と電源電圧の関係の仮定を示す。DVS における各仮定は、Transmeta Crusoe TM5400¹⁹⁾ の対応するデータをもとにしている。なお、PSU での電源電圧はクロック周波数によらず一定である。

図 6 に、各ベンチマークにおける、PSU による IPC の向上率とその平均を示す。各ベンチマークの 2 本の棒はそれぞれ統合度 2 ($f=50\%$) と統合度 4 ($f=25\%$) の場合の IPC 向上率である。積層された棒は下からそれぞれ分岐予測ミスペナルティ、L1 キャッシュヒットレイテンシ、L2 キャッシュヒットレイテンシ、命令の実行レイテンシの短縮による寄与である。IPC は $f=50\%$ 時に平均 48%、 $f=25\%$ 時に平均 95% 向上した。特に、jpeg、m88ksim、vortex を除くベンチマークでは、IPC が大幅に向上することが分かる。IPC が大きく向上すれば、3.2 節で示した式 (12) から分かるように、消費エネルギーの大幅な削減が期待できる。

IPC の向上の内訳について説明する。分岐予測ミスペナルティの減少は、vortex を除くすべてのベンチマークにおいて、最も大きく寄与している。平均では、IPC の向上のうち 63% が分岐予測ミスペナルティの減少によるものであった。特に、gcc、go、li では、IPC の向上のうち 70% 以上が分岐予測ミスペナルティの減少によるものであった。次に IPC の向上への影響が大きかったものは、L1 キャッシュのヒットレイテンシの減少である。平均で IPC の向上のうち 27% を占めた。この効果は、どのベンチマークにおいても同程度の IPC の向上として現れた。L2 キャッシュのヒットレイテンシの減少は、平均で IPC の向上のうち 9% と少ないが、vortex においては、IPC の向上の 44% と、最も大きな割合を占めた。これに対して、命令の実行

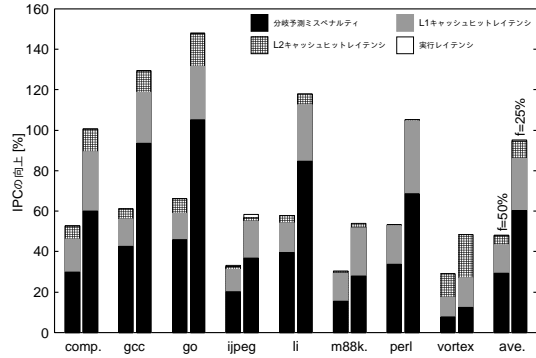


図 6 PSU による IPC の向上

Fig. 6 IPC improvement through PSU.

レイテンシの変化が IPC に及ぼしたものは、ほとんどなかった。これは、SPECint95 では整数演算命令が命令の大部分を占めるが、実行レイテンシが短くなる整数演算命令が少ないためである。

図 7 に、図 6 と 3.2 節の式 (12) から得られる PSU の消費エネルギーと、表 4 と 3.1 節の式 (9) から得られる DVS の消費エネルギーの評価結果を示す。縦軸は、通常モードにおける消費エネルギーで正規化した消費エネルギーである。各ベンチマークにおける 2 本の棒グラフはともに、PSU における正規化消費エネルギーである。左の棒グラフは $f=50\%$ における正規化消費エネルギーを、右の棒グラフは $f=25\%$ における正規化消費エネルギーを示す。また、2 本の横線 (破線) は、DVS における正規化消費エネルギーを示している。上下の破線は、それぞれ $f=50\%$ と $f=25\%$ の場合である。なお、3.1 節の式 (9) から分かるように、DVS においては、エネルギーの削減は電源電圧の減少のみによって達成されるため、正規化消費エネルギーはどのベンチマークにおいても等しい。

図より、 $f=50\%$ における PSU の消費エネルギーは、すべてのベンチマークにおいて DVS よりも少ないことが分かる。正規化消費エネルギーは DVS で平均 0.67 である一方 PSU では平均 0.57 であり、PSU を用いることで DVS より 14% 削減された。 $f=25\%$ においては、PSU の DVS に対する優位さは小さくなり、PSU は一部のベンチマーク (jpeg、m88ksim、vortex) で DVS より多くのエネルギーを消費している。これは、DVS のエネルギー消費は電源電圧の減少の 2 乗で削減されるが、PSU による IPC の増加はそれほど急速ではないからである。図 6 から分かるように、上記の 3 本のベンチマークでは IPC の向上率が少ない。しかしながら、平均ではエネルギー消費は 0.44 (DVS) から 0.39 (PSU) へと、DVS より 11% 削減されてい

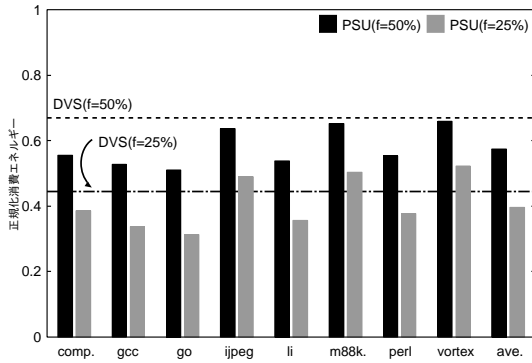


図7 180 nmのプロセス技術における $f=50\%$ と $f=25\%$ における正規化消費エネルギー

Fig. 7 Normalized energy consumption for $f=50\%$ and $f=25\%$ in the 180 nm technology.

る。現在のプロセス技術における PSU の優位性はこの程度にとどまっているが、5.3.2 項ではこの優位性は将来のプロセス技術ではより大きくなることを示す。

5.2 クロックの消費電力の割合が異なる場合

3.2 節の式 (12) により、PSU によって削減される消費エネルギーは、プロセッサの消費する電力に対する、クロックが消費する電力の割合 k に依存していることを示した。これまで k を 30% と仮定してきたが、 k はプロセッサの設計により変化する。文献 (13), (16)~(18) の例では、 k は 18% から 40% と幅がある。この節では、 k を変化させたときの消費エネルギーの評価結果を示す。

図 8 に評価結果を示す。表 4 に示した 180 nm のプロセス技術における仮定をしている。グラフの横軸はクロックの消費電力の割合であり、縦軸は、通常モードにおける消費エネルギーで正規化した消費エネルギーである。クロックの消費する電力の割合ごとに 2 本の棒グラフがある。両棒グラフはともに、PSU における正規化消費エネルギー（ベンチマーク平均）である。左の棒グラフは $f=50\%$ における正規化消費エネルギーを、右の棒グラフは $f=25\%$ における正規化消費エネルギーを示している。また、2 本の横線（破線）は、DVS における正規化消費エネルギーを示している。上下の破線は、それぞれ $f=50\%$ と $f=25\%$ における正規化消費エネルギーを示す。図より、クロックの消費電力の割合が減少すると、DVS に対する PSU の優位性は減少するが、すべての条件において、PSU は DVS よりも消費エネルギーを削減可能なことが分かる。PSU が最も不利となる 20% の場合においても、 $f=50\%$ において 9%、 $f=25\%$ において 2% 消費エネルギーを削減可能である。

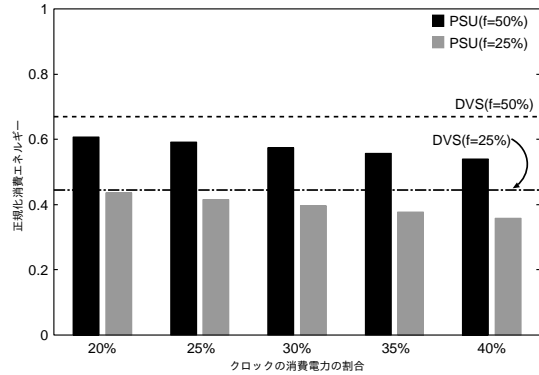


図8 様々なクロックの消費電力の割合に対する消費エネルギーの削減

Fig. 8 Energy consumption reduction for various clock power rates.

5.3 将来のプロセス技術における消費エネルギー

本節では、将来のプロセス技術における DVS と PSU の消費エネルギーの削減量を比較する。

5.3.1 将来の DVS プロセッサにおける電源電圧のスケールリングの見積り

将来における DVS による消費電力の削減量を見積もるには、プロセス技術の進歩に応じて、どのように電源電圧が変化するか、また、低下させたクロック周波数に対して電源電圧をどこまで減少させることができるかを調べる必要がある。

最大クロック周波数で動作する通常モードでの電源電圧 V_{DDmax} については、これまで発表されたプロセッサの電源電圧のトレンド²⁰⁾、および、TSMC と UMC が発表している電源電圧^{21), 22)} より求めた。その結果、以下の式が得られた：

$$V_{DDmax} = 0.0381 \times Technology^{0.7171} \quad (14)$$

ここで、 $Technology$ [nm] は、プロセス技術の最小加工寸法である。

一方、低消費電力モードでの電源電圧に関しては、 V_{DDmax} と同様の方法で見積もるだけの十分なデータはなかった。そこで、以下のようにして求めた。

一般に、電源電圧と閾値電圧が与えられたとき、それらとゲートが動作する最大クロック周波数 f との間に、以下に示す関係がある（たとえば、文献 23)）：

$$f \propto \frac{(V_{DD} - V_{th})^2}{V_{DD}} \quad (15)$$

ここで、 f を通常モードでのクロック周波数で正規化した低消費電力モードでのクロック周波数とし、比例定数を a とおくと、上式は次のように表すことができる：

$$f = a \times \frac{(V_{DD} - V_{th})^2}{V_{DD}} \quad (16)$$

比例定数 a は $f = 100\%$ のときの電源電圧 V_{DDmax} (式 (14)) と V_{th} から得ることができる:

$$a = \frac{V_{DDmax}}{(V_{DDmax} - V_{th})^2} \quad (17)$$

一方 V_{th} は V_{DDmax} と同様の方法で見積もり、以下の式を得た:

$$V_{th} = 0.0226 \times Technology^{0.5111} \quad (18)$$

比例定数 a を用い、低消費電力モードにおける V_{DD} を式 (16) より求めることができる:

$$V_{DD} = \frac{A + \sqrt{A^2 - 4a^2 V_{th}^2}}{2a} \quad (19)$$

ただし、

$$A = 2aV_{th} + f \quad (20)$$

である。

実際の電源電圧は、上式で定まる V_{DD} にある程度のマージンが必要と思われる。種々の理由で、電源電圧が低いほどこのマージンを大きくとる必要がある。たとえば、式 (15) から分かるように、プロセスのばらつきによる V_{th} のばらつきのクロック周波数への影響は、 V_{DD} が低いほど大きい。また、 V_{DD} が低いほどノイズの影響を受けやすい。そこでこのマージンを M とし、式 (19) を以下のように書き直す:

$$V_{DD} = M \times \frac{A + \sqrt{A^2 - 4a^2 V_{th}^2}}{2a} \quad (21)$$

ここで、 M は通常モードのとき 1、低消費電力モードのとき $M > 1$ であり、プロセス世代によらず一定とする。

本研究で関心のある $f=50\%$ 、 $f=25\%$ の M を計算するため、Crusoe TM5400 (180 nm プロセス) と TM5800 (130 nm プロセス) の f と V_{DD} の関係のデータを用いた。これらのデータと式 (17)、(18) より与えられる a と V_{th} を式 (21) に代入することにより、 M を計算することができる。表 5 に計算結果をまとめる^{*}。これより、 M を以下のように決定した:

$$M = \begin{cases} 1.00, & f = 100\% \\ 1.24, & f = 50\% \\ 1.40, & f = 25\% \end{cases} \quad (22)$$

以上まとめると、通常モードの電源電圧 V_{DDmax} は、式 (14) より得られる。低消費電力モードで $f=50\%$ 、 $f=25\%$ での電源電圧は、式 (17)、(18)、(22) を式 (21)

^{*} V_{DDmax} には、データの一貫性を保つため式 (14) から得られる値ではなく、実際の値を用いた。また、TM5800 の $f=50\%$ における電源電圧は残念ながら入手できなかった。

表 5 式 (21) のマージン M の計算結果

Table 5 Calculation results of margin M in equation (21)

正規化した クロック周波数	TM5400		TM5800	
	V_{DD}	M	V_{DD}	M
100%	1.65 V	1.00	1.30 V	1.00
50%	1.35 V	1.24	—	—
25%	1.10 V	1.39	0.90 V	1.41

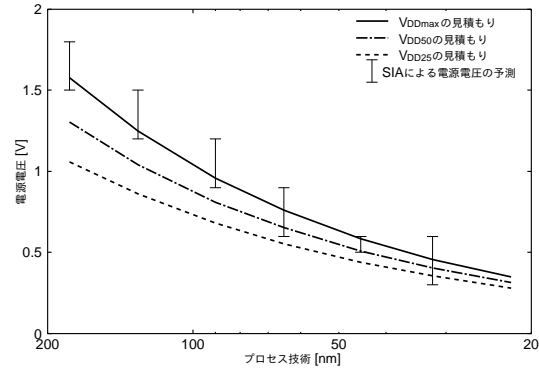


図 9 将来の DVS プロセッサの V_{DDmax} と V_{DDmin} の見積り結果

Fig. 9 Estimation of V_{DDmax} and V_{DDmin} in the future DVS processors.

に代入して得られる。

図 9 に V_{DDmax} 、 V_{DD50} 、 V_{DD25} の見積り結果を示す。ここで、 V_{DD50} 、 V_{DD25} は、それぞれ、 $f=50\%$ 、 $f=25\%$ のときの電源電圧である。図には、同時に SIA 予測²⁴⁾ の電源電圧の範囲も示している。図から分かるように、 V_{DD50}/V_{DDmax} と V_{DD25}/V_{DDmax} はプロセスが進むに応じて増加していく。このことは、将来 DVS の有効性が減少していくことを意味している。

5.3.2 将来のプロセス技術における消費エネルギーの比較

前項で見積もった V_{DDmax} 、 V_{DD50} 、 V_{DD25} を用いて、将来における DVS と PSU の消費エネルギーを比較する。表 6 に前項で見積もった各プロセス世代におけるクロック周波数と電源電圧の関係をまとめる。

図 10 に、各プロセス世代において、通常モードの消費エネルギーで正規化した消費エネルギー (ベンチマーク平均) を示す。まず、DVS の有効性はプロセス世代の進歩に応じて低下していくことが分かる。 $f=50\%$ の場合はおよそ 3~4%/世代、 $f=25\%$ の場合は 5~7%/世代の割合で消費エネルギーは増えていく。この結果、プロセス世代が現在の 180 nm からおよそ 10 年後の 32 nm の世代になると、消費エネルギーの削減率は、 $f=50\%$ の場合は 33% から 21%、 $f=25\%$ の場合は 55% から 40% に減少する。削減率が単調に減

表 6 将来のプロセス技術における DVS と PSU のクロック周波数と電源電圧の関係の仮定
Table 6 Assumptions of clock frequencies and supply voltages in the future technology for DVS and PSU.

プロセス	DVS			PSU		
	100%	50%	25%	100%	50%	25%
180 nm	1.58 V	1.29 V	1.06 V	1.58 V	1.58 V	1.58 V
130 nm	1.25 V	1.04 V	0.86 V	1.25 V	1.25 V	1.25 V
90 nm	0.96 V	0.81 V	0.68 V	0.96 V	0.96 V	0.96 V
65 nm	0.76 V	0.65 V	0.56 V	0.76 V	0.76 V	0.76 V
45 nm	0.58 V	0.51 V	0.44 V	0.58 V	0.58 V	0.58 V
32 nm	0.46 V	0.41 V	0.36 V	0.46 V	0.46 V	0.46 V
22 nm	0.35 V	0.32 V	0.28 V	0.35 V	0.35 V	0.35 V

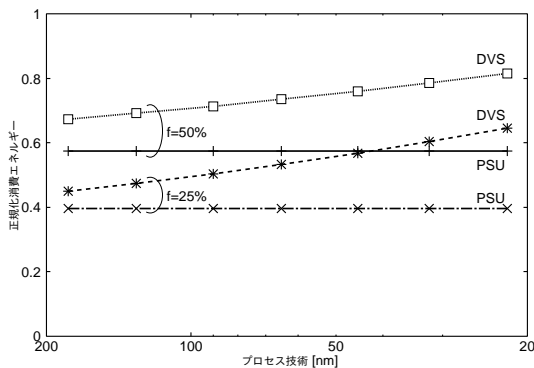


図 10 将来のプロセス技術における $f=50\%$ と $f=25\%$ における正規化消費エネルギー

Fig. 10 Normalized energy consumption at $f=50\%$ and $f=25\%$ in the future technology.

少していく理由は、前項で述べたように、通常モードにおける電源電圧に対する低消費電力モードでの電源電圧の割合が増加していくためである。

これに対して、PSU の消費エネルギーはプロセス世代の進歩に関係なく一定である。その結果、32 nm のプロセス技術においては、PSU は DVS よりも $f=50\%$ において 27%、 $f=25\%$ において 34% 多くの消費エネルギーを削減できる。この理由は、PSU の消費エネルギーの削減が、電源電圧の低下に依存しているのではなく、動的なアーキテクチャの変更によって達成されているためである。

6. 関連研究

Albonesi は、動的にプロセッサの資源のサイズとクロック周波数を変更し、実行時間を削減する方法を提案した²⁵⁾。一般に、資源サイズと実現可能なクロック周波数は、トレードオフの関係にある。このトレードオフの最適点を求め、プログラムの実行時間を削減する。

Bahar らは、プログラムの実行中に同時発行命令数

が大きく変化することに着目し、必要に応じて動的に命令発行論理と機能ユニットの一部を停止させ、消費電力を削減することを提案した²⁶⁾。

Manne らは分岐予測ミスしたバスからフェッチした命令を減らして消費電力を削減することを提案した²⁷⁾。具体的には、分岐予測の信頼性が低い分岐命令が連続してフェッチされた場合、その後の命令のフェッチを停止することにより、分岐予測ミスしたバスからフェッチした命令数を削減する。

Canal らは、データ、アドレス、命令がある特別のエンコーディングによって圧縮し、パイプラインの活動量を減少させ、消費電力を削減する方法を提案した²⁸⁾。

クロックの消費電力が多いことに着目し、その削減を試みた研究には、クロック分配線を最適化するもの²⁹⁾や、クロック分配線の一部を低電圧化するもの³⁰⁾や、クロックゲーティングを工夫するもの^{31),32)}があげられる。

7. まとめ

本論文では、PSU と呼ぶ消費エネルギーを削減する手法を提案し、現在と将来のプロセス技術において現在の手法である DVS と比較した。PSU はクロック周波数が減少したときに、電源電圧を最大値に保ったまま、複数のパイプラインステージを統合することによって消費エネルギーを削減する手法である。PSU を実装するために、それぞれ常時クロック、一時クロックと呼ぶ複数のクロック分配線と、ステージ統合を行うという信号を送る統合信号が必要となる。消費電力の削減は、パイプラインステージ統合時に、一時クロックのドライバの動作を停止することによって達成される。また、PSU を適用するとパイプラインが短くなるため、分岐予測ミスペナルティやキャッシュヒットレイテンシが短縮され、クロック周波数の低下による性能低下を抑えることが可能である。この 2 つの効果

により、PSU は従来の DVS よりも効果的に消費エネルギーを削減することができる。将来のプロセス技術では電源電圧の変動範囲が減少するため、DVS の効果は減少する一方、電源電圧に依存せずにアーキテクチャの変更によって消費エネルギーを削減する PSU は、その効果を維持することができる。

評価の結果、現在では PSU は DVS よりも $f=50\%$ のとき 14%、 $f=25\%$ のとき 11% 多く消費エネルギーを削減できる。この改善率はプロセス技術の進歩にともなって増加していく。たとえば、10 年後には、改善率は $f=50\%$ のとき 27%、 $f=25\%$ のとき 34% に増加する。このように、PSU は将来のモバイルプロセッサにおいては、より魅力的なエネルギー削減手法となる。

より深いパイプラインがプロセッサの性能をさらに向上させるという予測がいくつかの文献で述べられている^{33)~35)}。パイプラインがより深くなれば、全消費電力に対するクロックの消費電力の割合は増加する。この結果、統合できるステージ数も増加する。このため、PSU の有効性は本論文で述べたよりもさらに大きくなると考えられる。

参 考 文 献

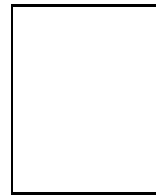
- 1) Deifendroff, K.: Transmeta Unveils Crusoe, *Microprocessor Report*, Vol.14, Archive 1, pp.15-16 (2000).
- 2) Transmeta Corporation: *Crusoe Processor Model TM5800 Product Brief* (2001).
- 3) Intel Corporation: *Mobile Pentium III Processor in BGA2 and Micro-PGA2 Packages Datasheet* (2001).
- 4) Intel Corporation: *Mobile Intel Pentium III Processor-M Datasheet* (2002).
- 5) Intel Corporation: *Mobile Intel Pentium 4 Processor-M Datasheet* (2002).
- 6) Advanced Micro Devices, Inc.: *Mobile AMD Athlon 4 Processor Model 6 CPGA Data Sheet* (2001).
- 7) Anghel, L. and Nicolaidies, M.: Cost Reduction and Evaluation of a Temporary Faults Detecting Technique, *Proc. 2000 Design, Automation, and Test in Europe*, pp.591-598 (2000).
- 8) 嶋田 創, 安藤秀樹, 島田俊夫: 低消費電力化のための可変パイプライン, 情報処理学会研究報告, 2001-ARC-145, pp.57-62 (2001).
- 9) Shimada, H., Ando, H. and Shimada, T.: Pipeline Stage Unification for Low-Power Consumption, *Int. Symp. on Low-Power and High-Speed Chips (COOL Chips V) Presentation Slides*, pp.194-200 (2002).
- 10) 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, 先進的計算基盤システムシンポジウム SACSIS2003, pp.283-290 (2003).
- 11) Hennessy, J.L. and Patterson, D.A.: *Computer Architecture: A Quantitative Approach, 2nd Edition*, Morgan Kaufmann Publishers Inc. (1996).
- 12) Semiconductor Industry Association: *International Technology Roadmap for Semiconductors 2001 Process Integration, Devices, Structures* (2001).
- 13) Gowan, M.K., Biro, L.L. and Jackson, D.B.: Power Considerations in the Design of the Alpha 21264 Microprocessor, *the 35th Design Automation Conf.*, pp.726-731 (1998).
- 14) Burger, D. and Austin, T.M.: The SimpleScalar Tool Set, Version 2.0, Technical Report CS-TR-97-1342, University of Wisconsin-Madison Computer Sciences Dept. (1997).
- 15) Glaskowsky, P. N.: Pentium 4 (Partially) Previewed, Vol.14, Archive 8, pp.1-4 (2000).
- 16) Anderson, F.E., Wells, J.S. and Berta, E.Z.: The Core Clock System on the Next-Generation Itanium Microprocessor, *2002 IEEE International Solid-State Circuits Conf. Visual Supplement to the Digest of Technical Papers*, pp.110-111 (2002).
- 17) Clark, L.T., Hoffman, E.J., Miller, J., Biyani, M., Liao, Y., Strazdus, S., Morrow, M., Velarde, K.E. and Yarch, M.A.: An Embedded 32-b Microprocessor Core for Low-Power and High-Performance Applications, *IEEE Journal of Solid-State Circuits*, Vol.36, No.11, pp.1599-1608 (2001).
- 18) Gronowski, P.E., Bowhill, W.J., Preston, R.P., Gowan, M.K. and Allmon, R.L.: High-Performance Microprocessor Design, *IEEE Journal of Solid-State Circuits*, Vol.33, No.5, pp.677-686 (1998).
- 19) Laird, D.: *Crusoe Processor Products and Technology*, Transmeta Corporation (2000).
- 20) Butts, J.A. and Sohi, G.S.: A Static Power Model for Architecture, *Proc. 33rd Annual Int. Symp. on Microarchitecture*, pp.191-201 (2000).
- 21) Diefendorff, K.: TSMC Sets Sights on #1, *Microprocessor Report*, Vol.14, Archive 6, pp.17-21 (2000).
- 22) <http://www.umc.com/>
- 23) Mudge, T.: Power: A First-Class Architectural Design Constraint, *IEEE Computer*, Vol.34, No.4, pp.52-58 (2001).
- 24) Semiconductor Industry Association: *International Technology Roadmap for Semiconduc-*

- tors 2000 Update Process Integration, Devices, Structures (2000).
- 25) Albonesi, D.H.: Dynamic IPC/Clock Rate Optimization, *Proc. 25th Annual Int. Symp. on Computer Architecture*, pp.282-292 (1998).
 - 26) Bahar, R.I. and Manne, S.: Power and Energy Reduction Via Pipeline Balancing, *Proc. 28th Annual Int. Symp. on Computer Architecture*, pp. 218-229 (2001).
 - 27) Manne, S., Klauser, A. and Grunwald, D.: Pipeline Gating: Speculation Control For Energy Reduction, *Proc. 25th Annual Int. Symp. on Computer Architecture*, pp.132-141 (1998).
 - 28) Canal, R., Gonzalez, A. and Smith, J.E.: Very Low Power Pipelines using Significance Compression, *Proc. 33rd Annual Int. Symp. on Microarchitecture*, pp.181-190 (2000).
 - 29) Xi, J.G. and Dai, W.W.-M.: Buffer Insertion and Sizing Under Process Variations for Low Power Clock Distribution, *Proc. Design Automation Conf. 1995*, pp.491-496 (1995).
 - 30) Pangjun, J. and Sapatnekar, S.S.: Clock Distribution Using Multiple Voltages, *Proc. Int. Symp. on Low-Power Electronics and Design 1999*, pp. 145-150 (1999).
 - 31) Donno, M., Ivaldi, A., Benini, L. and Macii, E.: Clock-Tree Power Optimization based on RTL Clock-Gating, *Proc. Design Automation Conf. 2003*, pp.622-627 (2003).
 - 32) Farrahi, A.H., Chen, C., Srivastava, A., Tellez, G. and Sarrafzadeh, M.: Activity-Driven Clock Design, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.20, No.6, pp.705-714 (2001).
 - 33) Hartstein, A. and Puzak, T.R.: The Optimum Pipeline Depth for a Microprocessor, *Proc. 29th Annual Int. Symp. on Computer Architecture*, pp.7-13 (2002).
 - 34) Hrishikesh, M.S., Jouppi, N.P., Farkas, K.I., Burger, D., Keckler, S.W. and Shivakumar, P.: The Optimal Useful Logic Depth Per Pipeline Stage is 6-8 FO4, *Proc. 29th Annual Int. Symp. on Computer Architecture*, pp. 14-24 (2002).
 - 35) Sprangle, E. and Carmean, D.: Increasing Processor Performance by Implementing Deeper Pipelines, *Proc. 29th Annual Int. Symp. on Computer Architecture*, pp.25-34 (2002).

(平成 15 年 5 月 7 日受付)

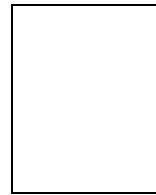
(平成 15 年 7 月 5 日採録)

嶋田 創 (学生会員)



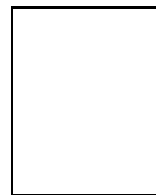
1976 年生。1998 年名古屋大学工学部情報工学科卒業。2000 年名古屋大学大学院工学研究科情報工学専攻博士課程前期課程修了。2003 年名古屋大学大学院工学研究科電子情報学専攻博士課程後期課程満了。現在、名古屋大学大学院工学研究科電子情報学専攻研究生。電子情報通信学会/情報処理学会 2003 年先進的計算基盤システムシンポジウム優秀学生論文賞受賞。計算機アーキテクチャの研究に従事。

安藤 秀樹 (正会員)



1959 年生。1981 年大阪大学工学部電子工学科卒業。1983 年大阪大学大学院修士課程修了。京都大学工学博士。1983 年三菱電機(株)LSI 研究所。ISDN 用デジタル信号処理 LSI, 第 5 世代コンピュータ・プロジェクトの推論マシン用プロセッサの設計に従事。1991 年 Stanford 大学客員研究員。1997 年名古屋大学大学院工学研究科電子情報学専攻講師。1998 年名古屋大学助教授。1998 年~2001 年東京大学大学院理学系研究科助教授併任。1998 年, 2002 年情報処理学会論文賞, 電子情報通信学会/情報処理学会 2003 年先進的計算基盤システムシンポジウム優秀学生論文賞受賞。計算機アーキテクチャ, コンパイラの研究に従事。ACM, IEEE, 電子情報通信学会会員。

島田 俊夫 (正会員)



1968 年東京大学工学部計数工学科卒業。1970 年東京大学大学院修士課程修了。同年電子技術総合研究所入所。1993 年より名古屋大学大学院工学研究科電子情報学専攻教授。人工知能向き言語, LISP マシン, データフロー計算機の研究に従事。最近はマイクロプロセッサのアーキテクチャやチップ内並列処理の研究を行っている。1988 年度市村賞, 1994 年度情報処理学会論文賞, 1995 年度注目発明賞, 2001 年度情報処理学会論文賞, 電子情報通信学会/情報処理学会 2003 年先進的計算基盤システムシンポジウム優秀学生論文賞受賞。工学博士。