

# Dynamic Standby Prediction for Leakage Tolerant Microprocessor Functional Units

著者： Ahmed Youssef, Mohab Anis and Mohamed Elmasry

出典： *Proc. of the 39th Annual IEEE/ACM International Symposium on Microarchitecture, pp.371-384, Year2006*

発表者： 所属：本多・近藤研究室 学籍番号：0953018 氏名：徳園智哉

## 1 概要

近年、トランジスタ微細化によるリーク電力の増加が問題となってきている。65nm プロセスでのプロセッサの全消費電力に対するリーク電力による割合が50%を越えている。そこで、リーク電力を低減させるために、使用されていない Functional units(FU) の電力供給を断つ方法が研究されてきた。しかしそれらの方法には、実装や低減効率の面で問題があった。

本研究では、FU が使用されない時間 SP(Standby period) を予測して電力供給を断つ手法 DSSG(Dynamic sleep signal generator) を提案する。また、提案手法による有用性を、予測精度と電力消費についてシミュレーションして評価する。

## 2 Breakeven point

FU への電力を断つ回路を動作させるのにも電力を消費する。この動作のための電力消費量が FU 休止による消費電力低減量と等しくなった時点を Breakeven point(BEP) という(図1)。BEP はリーク電力の低減効果を考える上で、重要な値となる。

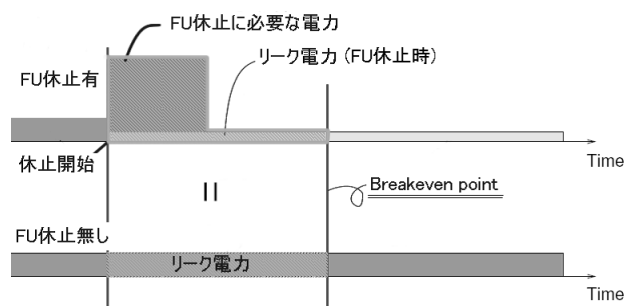


図 1: Breakeven point

## 3 既存手法と問題点

リーク電力を抑えるには、FU が使われていないときに以下の操作を行う(図2)。

1. アイドル信号が FU から Sleep Signal Generator に送られる。

2. 休止信号を生成し Leakage Control Mechanism に渡す。

3. FU をスリープさせる。

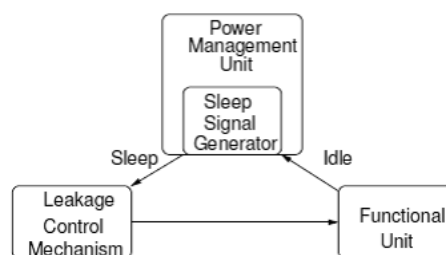


図 2: リーク電力低減操作

この休止信号をどのように生成するかについて、様々な研究がなされてきた。

コンパイラベースの手法では、アプリケーションのコードを分析し、FU が使われない期間を特定し休止信号を生成した。しかしこの手法を利用するにはリコンパイルが必要であった。また、ハードウェア仕様が変更されると対処できないという欠点があった。

FU のアイドルとなった時間があらかじめ決めていた長さ(基準点)に達したとき、休止信号を生成する SSSG(Static sleep signal generator) では、SSSG 自身の電力オーバーヘッドが低いという特徴があるが、SP が BEP に届かない場合があった。

## 4 DSSG の動作

DSSG は内在するカウンターに FU を休止させた時間が BEP に届いたかどうかを記録し、基準点を動的に変更する。アプリケーションが FU を頻繁に使い、FU がアイドルする時間が BEP より短い場合、基準点を伸ばし休止信号を生成しないようにする。一方稀にしか使われない場合は基準点を短くし、休止信号の生成を早める。そのような DSSG の動作を 6 つの動作領域(図3)で説明する。

1. FU 使用中.DSSG 待機状態。

2. FU のアイドル開始。

アイドル時間が基準点に達したら休止信号生成。

3. 休止させた時間 > BEP の場合にカウントする。

FU が駆動したら 1 へ。

## 5.2 DSSG の回路実装と電力消費

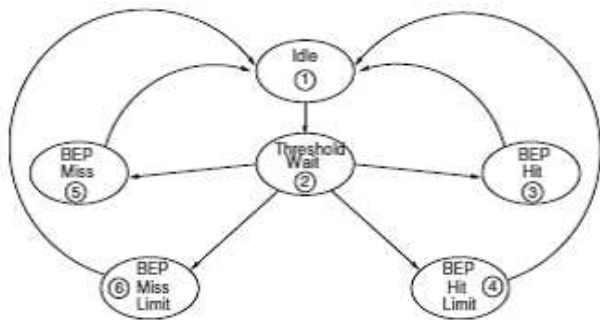


図 3: DSSG state machine.

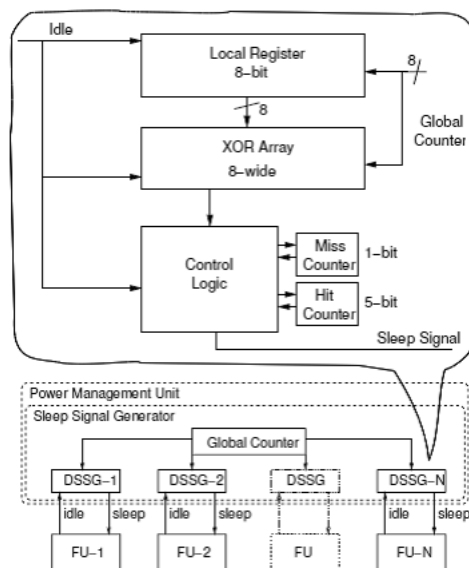


図 5: DSSG の実装

4.3 が一定数 (hit-limit) カウントされたとき、定義しておいた長さ (step-) だけ基準点を短くする。

(次から休止信号生成が早まる)

5. 休止させた時間 < BEP の場合にカウント。

6.5 が一定数 (miss-limit) カウントされたとき、定義しておいた長さ (step+) だけ基準点を長くする。

(次から休止信号生成を遅らせる)

## 5 実験と評価

### 5.1 DSSG の精度

SPEC2000 ベンチマークを用いた整数, 浮動小数点それぞれの Multiply, Divide についての予測精度結果を図 4 に示す。SSSG と比べても, DSSG は BEP を変動させても高精度を維持出来ることが分かる。精度は次のように計算している。

$$Accuracy(\%) = \frac{Number\ of\ hits}{Number\ of\ hits + Number\ of\ misses} \quad (1)$$

(Number of hits : FU の SP が BEP を越えた回数)

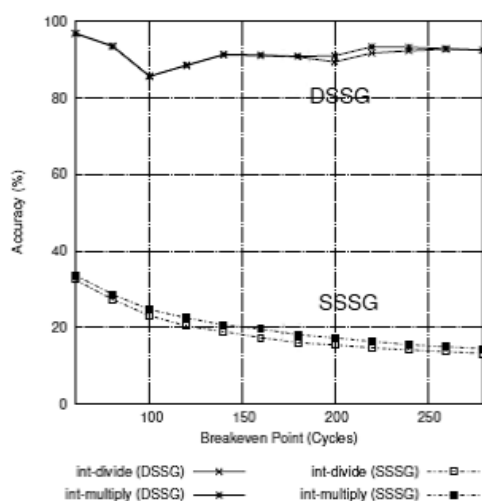


図 4: Int,FP-Multiply/Divide での DSSG,SSSG 精度

DSSG の有用性を評価するために, DSSG 1 回路がどれだけの電力を消費するかという情報が必要である。DSSG 実装は図 5 のように FU に 1:1 で取り付けるため, DSSG 1 回路の消費電力が高いと全体の消費電力が上昇してしまう可能性もある。130nmCMOS に DSSG を実装し, 1GHz のクロック周波数での動作を Spice でシミュレーションした結果, DSSG のアイドル時に  $141 \mu W$ , 動作時に  $356 \mu W$ , グローバルカウンターで  $360 \mu W$  の電力を消費していることが分かった。デュアルコア UltraSPARC マイクロプロセッサの 64bitALU 消費電力がおよそ  $400mW$  であることを考えても, DSSG での電力消費はプロセッサの電力消費に僅かな影響しかない。

## 6 まとめ

DSSG による FU の SP は, 80~ 98% という高精度で予測することが可能である。アプリケーションの動作が変動したとしても精度は維持され, SP が BEP を越える時のみ休止信号が生成される。これにより BEP に届かないのに休止してしまう既存手法の欠点を補う事ができた。パワーゲーティング [1] などの低消費電力設計に応用した場合, それらは低減効率を最大限に発揮することができる。

## 参考文献

- [1] Seidai Takeda, Toshihiro Kashima, "Analysis for factors that affect power dissipation for Multiplier applying Run Time Power Gating" IEICE Tech. Rep., vol. 106, no. 549, VLD2006-154, pp. 81-85, March 2007.