

閾値ホッピング(V_{TH} -hopping)手法を用いた低電圧・低消費電力プロセッサ

野瀬 浩一, 平林 雅之, 川口 博, 李 誠洙*, 桜井 貴康

東京大学生産技術研究所

*梨花女子大学 情報通信学科

〒153-8505 東京都目黒区駒場 4-6-1

あらまし 本研究では、閾値をプロセッサの負荷に応じてソフトウェアが制御する閾値ホッピング(V_{TH} -hopping)を提案する。これは負荷の軽い時間は閾値を高くし、同時に周波数を下げることで動作時の無駄な消費電力を削減する方式である。閾値の制御は基盤バイアスを変化させることによって行う。

MPEG4 のエンコード部をシミュレーションした結果、電源電圧 0.5V のとき本手法を用いることで消費電力が 18% に削減された。また positive back-bias 手法と閾値ホッピングを組み合わせた小規模な RISC プロセッサを設計し、約 90% 消費電力を削減することが可能であることを示した。

キーワード 低消費電力, 低電圧, リアルタイムアプリケーション, positive back-bias

V_{TH} -hopping Scheme in Low-Voltage and Low-Power Processors

Koichi Nose, Masayuki Hirabayashi, Hiroshi Kawaguchi, Seongsoo Lee* and Takayasu Sakurai

Institute of Industrial Science, University of Tokyo

*Department of Information Electronics, Ewha University

4-6-1 Komaba, Meguro-Ku, Tokyo 153-8505, Japan

Abstract

A threshold voltage hopping (V_{TH} -hopping) scheme is proposed where V_{TH} is dynamically controlled through software depending on a workload. V_{TH} -hopping is shown to reduce the power to 18% of the fixed low- V_{TH} circuits in 0.5V supply voltage regime for multimedia applications. A positive back-gate bias scheme within V_{TH} -hopping is presented for the high-performance and low-voltage processors. The measurement result shows about 90% power reduction is possible by using V_{TH} -hopping.

key words Low-power, Low-voltage, real-time application, positive back-bias

1.はじめに

LSIのデバイスの微細化・集積化はムーアの法則にしたがって現在も着実に進んでいる。それに伴い、低消費電力・ゲート耐圧の観点から電源電圧(V_{DD})もまた同様に低電圧化が進んでいる。ITRS[1] (International Technology Roadmap for Semiconductors)によると、2011年には電源電圧が0.5V程度になると予想されている。電源電圧0.5Vのときの閾値と回路遅延、消費電力の関係を図1に示す。低電源電圧で高いパフォーマンスを維持するためにはトランジスタの閾値(V_{TH})を下げる必要がある。しかしながら閾値を下げるとリーク電流によるスタティック消費電力が指数関数的に増大する問題が発生する。従って、これらのトレードオフの解決が今後の低電圧LSI発展のカギを握るといえる。

また、従来のパフォーマンス重視の考え方から、バッテリー時間が重要となる携帯端末やコストが問題となるパッケージ技術など低消費電力に対する要求がますます増加している。しかしその一方で、特に動画などのリアルタイム処理を行うためには一定のレートで常に処理する必要もある。そのため、リアルタイム処理などでは要求される処理をしながらいかに低消費電力化を実現するかということが重要視されている。

この問題を解決するためにロジック部に従属接続したMOSFETを制御することで待機時のリーク電流を抑制するMTCMOS[1]や基盤バイアスによって待機時のリーク電流を削減するVTCMOS[2]回路などが提案されている。しかしながら、これらの手法では動作時のリーク電流が削減されない。一方、図2のようにクリティカルパスには低閾値のトランジスタ、それ以外には高閾値のトランジスタを配置することでリーク電流を定常的に抑制するDual- V_{TH} [3]手法も提案されている。しかしこの手法では、低閾値のトランジスタには常にリーク電流が流れてしまうため、電力削減効果が低減してしまう。

本研究ではこれらの問題を解決するために、閾値をプロセッサの負荷に応じて変化させる閾値ホッピング(V_{TH} -hopping)を提案する。これは負荷の軽い時間は閾値を高くし、同時に周波数を下げることで動作時の無駄な消費電力を削減する方式である。閾値の制御は基盤バイアスを変化させることによって行う。

この方法は動的に電源電圧と動作周波数を制御するDynamic V_{DD} Scaling (DVS)[4]に似ているが、DVSはダ

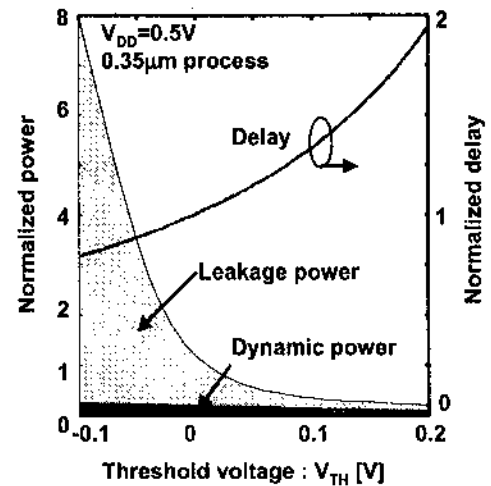


図1 Power and delay dependence on V_{TH}

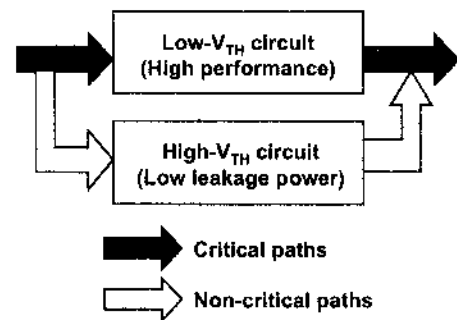


図2 Dual- V_{TH} scheme

イナミック電力が支配的であるときに有効な手段であるのに対し、本手法はスタティック電力が支配的である低電圧・低閾値の場合に有効な方法となる。

閾値ホッピングの有効性を示すために、MPEG4のエンコード部の消費電力削減効果をシミュレーションによって示した。また小規模なRISCプロセッサを設計し、基盤バイアスを用いた閾値ホッピングの効果を測定した。

2.閾値ホッピング (V_{TH} -hopping)によるリーク電力削減技術

消費電力と負荷(workload)の関係を図3に示す。最高周波数で常に動作させないと処理不可能になる負荷(最大負荷)を1とすると、それ以下の負荷(workload < 1)の処理においては動作周波数を下げて処理することが可能となる。また、同時に閾値をその周波数で動作する範囲内で上昇させることが可能となる。破線は負荷に応じて動作周波数のみを変化させた場合(閾値は一定)である。充放電電流によるダイナミック電力は動作周波数に比例するため、負荷が軽い時はその分の消費電

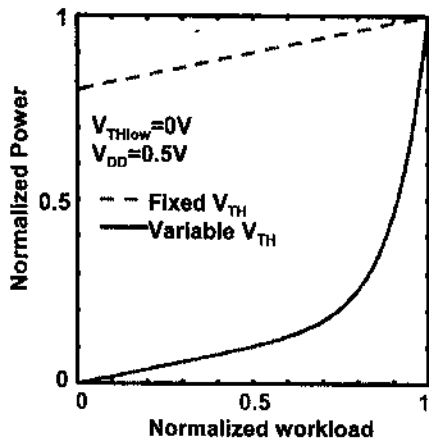


図3 Power dependence on workload

力が削減される。しかしながらリーク電流によるスタティック電力は動作周波数によらず一定であるため、スタティック電力が支配的となる低閾値領域では動作周波数の制御だけでは全体の消費電力を十分に削減することができない。一方、実線は負荷に応じて動作周波数および閾値を同時に変化させた場合を示している。動作周波数と閾値を同時に可変にすることにより消費電力を効果的に抑制できる。

3. 閾値ホッピング手法の概要

閾値ホッピングの概要を図4に示す。回路はコア部である Target processor と閾値および動作周波数を制御する Power control block で構成される。Power control block は dynamic V_{DD} scaling 向けに提案されたソフトウェアフィードバック方式[5]と同様なソフトウェアで制御することが可能である。このソフトウェアではリアルタイムアプリケーションにおいてプロセッサの処理状況と負荷に応じて動作周波数と閾値を切り替える制御信号を出力する。Target processor 内部について、閾値を Power control block から出力された V_{THlow_Enable} および V_{THhigh_Enable} 信号で制御する。また同時に動作周波数も f_{CLK} および $f_{CLK}/2$ にそれぞれ変化させる。負荷が重いときは V_{THlow_Enable} をオンさせ、プロセッサ内部の閾値を低くし(V_{THlow})、動作周波数を上げる(f_{CLK})。一方、負荷が軽いときはプロセッサ内部の閾値を高くし(V_{THhigh})、動作周波数を下げる($f_{CLK}/2$)ことで消費電力を軽減する。 V_{THlow} はプロセッサ内の回路がすべて f_{CLK} で動作可能となる範囲内で最大の閾値、 V_{THhigh} は $f_{CLK}/2$ で動作可能となる範囲内で最大の閾値である。

リアルタイムアプリケーション上での周波数・閾値の選択方法の概要を図5に示す。ここでは MPEG4 SP@L1

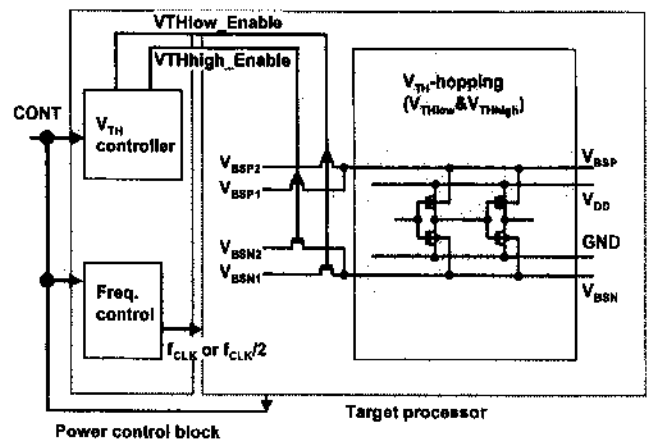


図4 Schematic diagram of V_{TH} -hopping

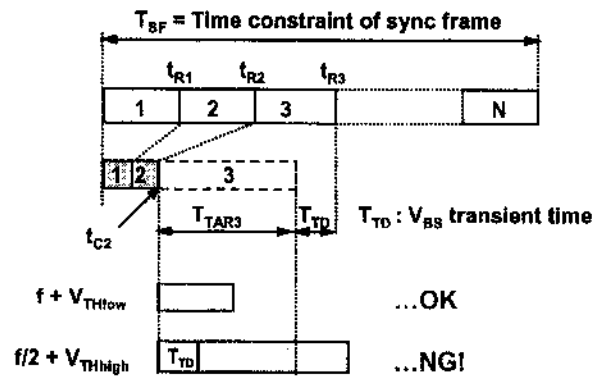


図5 Determination of frequency and threshold voltage

エンコードを例にして挙げる。MPEG4 は毎秒 15 フレームの処理を必要とする。ここで 1 枚の画像の処理をする時間(1/15 sec.)を 1 sync frame と呼ぶことにする。各 sync frame を N 分割し(それぞれを timeslot と呼ぶ)、それぞれの timeslot までに特定の処理を終えるように動作周波数や閾値を変化させる。図5では、timeslot2 までの処理は t_{R2} までに終了すれば良いのだが、負荷が軽い処理だったために t_{C2} で timeslot2 までの処理を終了してしまった場合を示してある。Timeslot3 を処理するための動作周波数・閾値は T_{TAR3} の時間で処理できるように設定すればよい(このとき、基盤バイアスを変化させる時間 T_{TD} を考慮する)。周波数が $f_{CLK}/2$ でも T_{TAR3} 以下で処理可能な場合は $f_{CLK}/2$ 、できない場合は f_{CLK} で動作させる。閾値は周波数に応じて、 $f_{CLK}/2$ なら V_{THhigh} を、 f_{CLK} なら V_{THlow} を選択する。ここで、前 timeslot と同じ閾値で動作させる場合は基盤バイアス変更のための時間 T_{TD} が必要ないという点に注意する。図5では timeslot2 の処理において V_{THlow} が選択されていたときを仮定している。

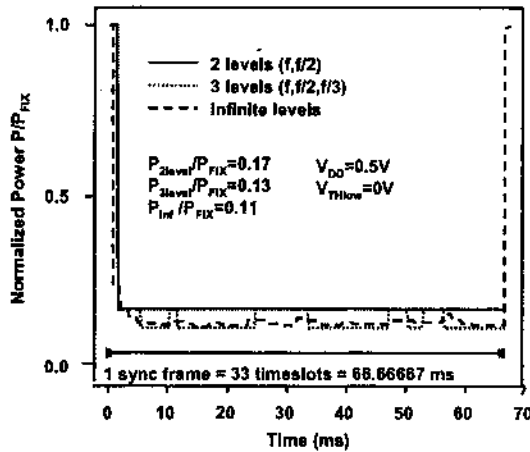


図 6 Power transition of V_{TH} -hopping

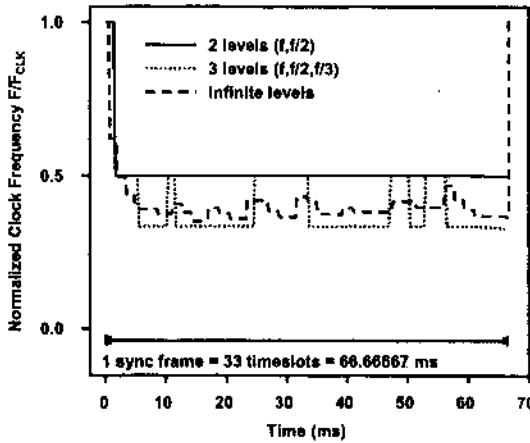


図 7 Frequency transition of V_{TH} -hopping

このとき、より多くの周波数を用意し切り替えることで更に消費電力を軽減する方法も考えられる。そこで閾値ホッピングを用いて MPEG4 エンコード処理を行ったときの消費電力の時間推移のシミュレーションを行った(図 6)。同期に関する問題や外部システムとのインターフェイスなどを考慮し、周波数を f_{CLK} , $f_{CLK}/2$, $f_{CLK}/3$, ..., f_{CLK}/N と N 種類の中で切り替えるとする。3 種類以上 ($N \geq 3$) の動作周波数を用意した場合、2 種類の場合よりも消費電力は削減できるがその改善は 6% までと小さく、複数用意することによる制御の複雑性や回路オーバーヘッドを考慮すると、2 種類の動作周波数・閾値を用いるのが最も効率的である。

また図 7 から明らかなように、一般的な動画の場合、約 94% の時間を f_{CLK} よりも低い周波数で動作させても real-time 処理が可能である(しかし f_{CLK} は最も重い処理が必要な画像パターンが来た場合に必要である)。MPEG2 decoding や VSELP voice codec などのアプリケーションでも同様の傾向がある。

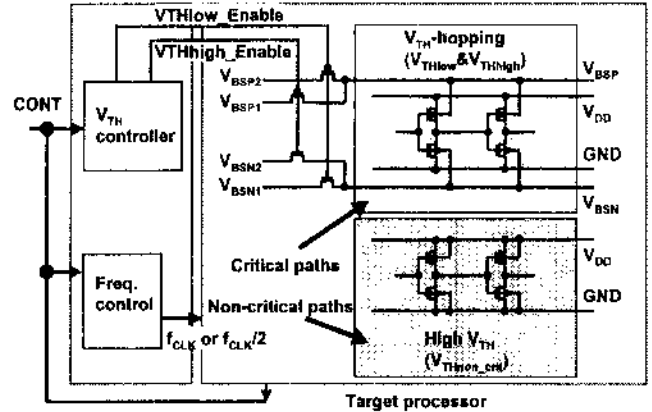


図 8 Schematic diagram of design which combines V_{TH} -hopping and dual- V_{TH}

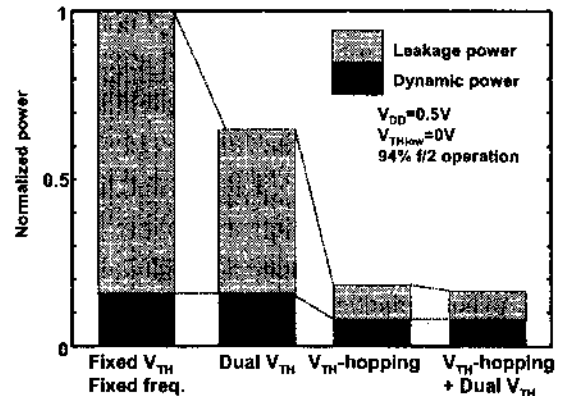


図 9 Power comparison among single fixed V_{TH} , dual- V_{TH} and V_{TH} -hopping

ここまではプロセッサコア全体の閾値をホッピングする方法を述べてきたが、その他に Dual- V_{TH} と組み合わせる方法(閾値ホッピング+dual V_{TH})も考えられる。つまり、クリティカルパスでない部分はあらかじめ高閾値($V_{TH, noncrit}$)トランジスタで設計し、リーク電流が定常的に流れないようにする。そして閾値ホッピングは低閾値のトランジスタを使用する必要があるクリティカルパス部のみに対して行うという方法である。この場合はアクティブ時の消費電力はより削減されるが、図 8 にあるように閾値を $V_{TH, noncrit}$ に設定し、ウェルを分離することによる面積オーバーヘッドが発生する。Dual- V_{TH} を組み合わせない場合は $V_{TH, noncrit}$ を用意する必要がなくなる。これらの手法を用いたときの消費電力削減効果の比較について次に述べる。

図 9 は MPEG4 エンコードを従来の single V_{TH} , dual- V_{TH} [3], そして提案する閾値ホッピングを用いて処理したときの消費電力の比較である。閾値ホッピングを用いたときの消費電力は周波数および閾値を固定

した場合(Fixed V_{TH} , Fixed freq.)の 18.5%に、また dual- V_{TH} に比べても 27%に削減される。閾値ホッピングに dual- V_{TH} を付加した手法(上述の閾値ホッピング + dual V_{TH})の場合、閾値ホッピングのみの場合に比べて消費電力の削減効果は 1.5%改善されるが、上述のとおり、ウェル分離によるオーバーヘッド等を考慮すると閾値ホッピングに dual- V_{TH} を付加する効果は小さい。したがって critical path であるかどうかにかかわらず Target processor 全体の閾値をホッピングする方法が最も効果的であることがわかる。

閾値ホッピングと dual- V_{TH} を組み合わせた場合の削減効果が 1.5%しかない理由として、それぞれのゲートのリーク電流はすでに閾値ホッピングによって削減されているのでクリティカルパス以外の部分に高い閾値を用いても大きな差が発生しないため、ということが挙げられる。

4. 閾値ホッピングを用いた RISC プロセッサの設計

前節では基盤の閾値が低い($V_{THlow}=0V$ 付近)ときの場合についてその有効性を述べた。この他に基盤の閾値が比較的高い場合、高速化を実現するために基盤バイアスを正にバイアスする positive back-bias 手法[6]を用いる場合にも閾値ホッピングが有効となる。基盤に正のバイアスをかけると閾値が低くなることから動作速度は上昇する。このときの動作周波数を f_{CLK} とし、バイアスをかけていない(または負にバイアスをかける)ときの動作周波数を $f_{CLK}/2$ とすればより高速な回路が設計できる。しかしながら positive back-bias を付加した場合、閾値が低くなることによるリーク電流の増加や、ドレイン(ソース)-基盤間に生じる順バイアスによる基盤電流(junction leakage)といったスタティック電力が発生するため、消費電力が増大する。そこで閾値ホッピングを併用することでスタティック電力を削減する。

閾値ホッピング手法の効果を測定するため、基盤バイアスを変更可能な RISC プロセッサを設計した。図 10 はそのチップ写真である。チップは $0.6\mu m$ CMOS テクノロジーで設計されている。閾値ホッピングを用いることによる回路オーバーヘッド(基盤バイアス用信号線 V_{BSP}, V_{BSN} および閾値ホッピングのための基盤バイアス選択回路(V_{BS} selector)など)は 14%であった。RISC コア部の面積は $2.1mm \times 2.0mm$ 、 V_{BS} selector 部の面積は $0.2mm \times 0.6mm$ である。

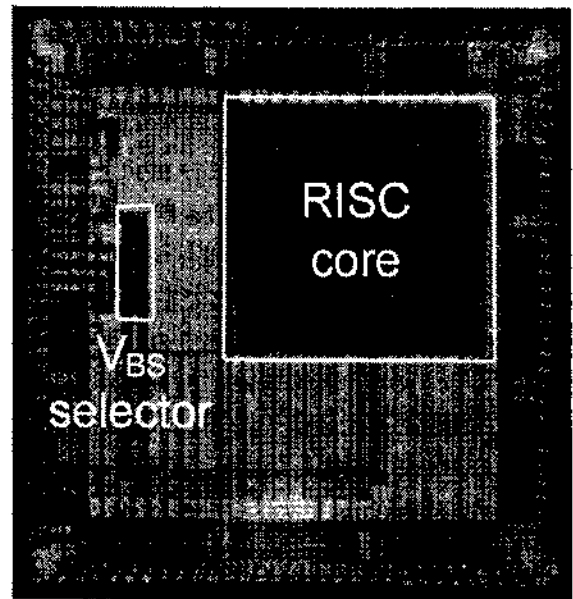


図 10 Microphotograph of RISC processor

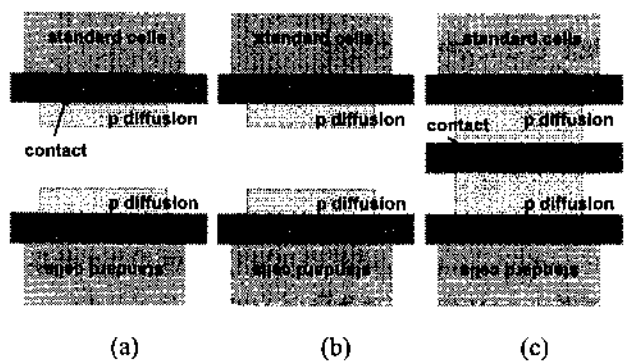


図 11 Place and route using conventional standard cells

コア部の設計においてはスタンダードセルを用いて設計した。通常のスタンダードセルは基盤・ウェルコンタクトは電源線(V_{DD}, GND)につながれているため、このままでは基盤バイアスを変化させることができない。本研究では通常のスタンダードセルを一つ一つ変更することなく、また従来の配置配線ツール[7]を用いて基盤バイアスを変更可能なレイアウトにする手法を提案する。

図 11 にその手順を示す。まず、通常のスタンダードセルを用いて配置配線を行う。このとき図 11(a)のように、基盤バイアス用信号線(V_{BSP}, V_{BSN})を通せるように、適当なスタンダードセル間の間隔をあけて配置する。これは配置配線ツールのパラメータを調整することで実行できる。次に、SKILL スクリプト[8]を用いて電源線上にある基盤・ウェルコンタクトを除去する(図 11(b))。最後に図 11(c)のようにセル間のウェル領域の隙間を埋

め、 V_{BSP} , V_{BSN} 線を配し、基盤・ウェルコンタクトおよびコンタクトのための拡散層を追加する。この手法を用いることでスタンダードセルを変更することなく基盤バイアス用信号線を追加することが可能となる。

図 12 にこの回路の測定結果および SPICE シミュレーション結果を示す。 V_{FW} は positive back-bias 電圧である。 ΔV_{FW} は positive back-bias 電圧のばらつきを考慮したもので、ここでは $0.1V$ (V_{DD} の $\pm 5.5\%$ で妥当である) に設定した。positive back-bias をかけたときの遅延の最悪値は V_{FW} がばらつきの中で最も低いとき(閾値が最も高い)、接合リーク電流の最大値は V_{FW} がばらつきの中で最も高いときにおこる。測定した結果、遅延は $V_{FW}=0.6V$ のとき、positive back-bias をかけなかった場合と比較して 29% 高速動作となった。一方、このとき接合リーク電流の最大値 ($V_{FW}=0.7V$ のとき) は急激に増大するが、 V_{TH} -hopping 手法を利用し、低い周波数動作のとき positive back-bias をかけないようにすることにより、positive back-bias をかけた場合に比べて消費電力を 91% 削減することが可能となる。

5. まとめ

本研究ではソフトウェアを用いてプロセッサの負荷に応じて閾値と動作周波数を動的に変化させる閾値ホッピング (V_{TH} -hopping) 手法を提案した。本手法の効果を電源電圧 $0.9V$ の低電圧動作の MPEG4 アプリケーション上で調べたところ消費電力が 18% に低減された。本手法は特に低電源電圧・低閾値設計、つまりリークによる消費電力が支配的になる領域において有効である。

閾値ホッピングを使用可能なレイアウトにするための設計手法を提案した。それを用いて設計した小規模 RISC プロセッサを測定し、positive back-bias をかけることで 29% 高速化し、その時に問題となる接合リーク電流を閾値ホッピングで抑制することが可能であることを示した。

今後は微細化に伴う低電力化やゲート耐圧の問題の解決のため、更なる低電源電圧化は必須となる。また、高いパフォーマンスを必要とするアプリケーションでは閾値を下げることも重要となる。本手法はそれらの要求を満たす技術として重要となってくると考えられる。

謝辞

本チップ試作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し ローム(株)および凸版印刷(株)

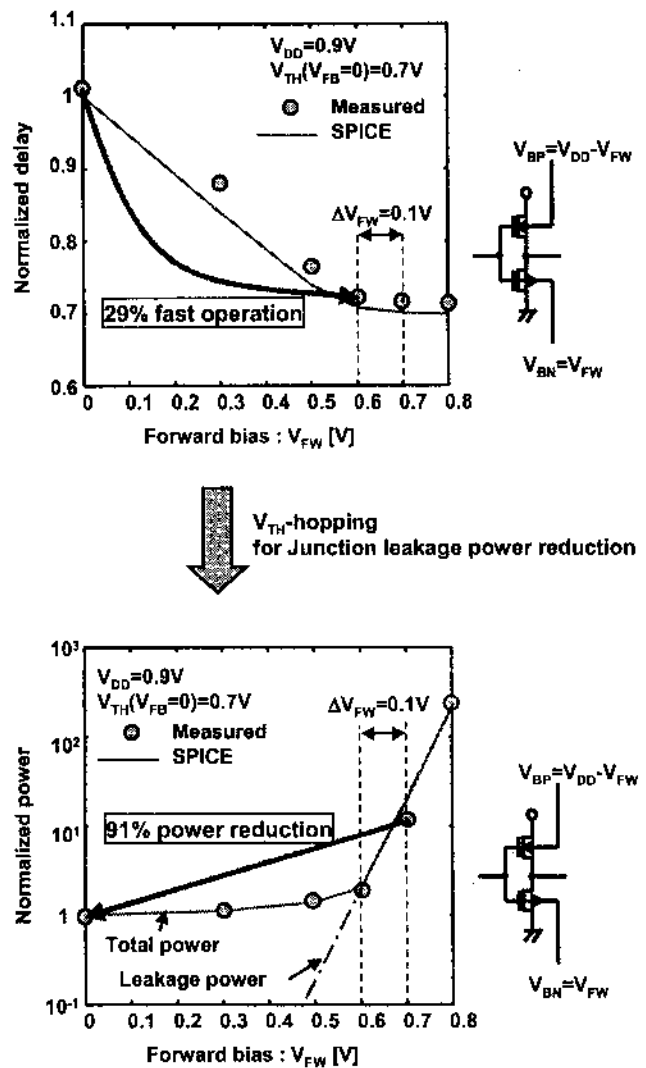


図 12 Delay and power of V_{TH} -hopping with positive back-gate bias

の協力で行われたものである。

本研究は未来開拓プロジェクトの一環として行われたものである。

参考文献

- [1] Shin'ichiro Mutoh, et al, "1-V power supply high-speed digital circuits technology with multithreshold-voltage CMOS," *IEEE J. Solid-State Circuits*, vol. 30, pp.847-854, Aug. 1995.
- [2] T. Kuroda, et al, "A 0.9-V 150-MHz, 10-mW, 4mm², 2-D discrete cosine transform core processor with variable threshold-voltage (VT) scheme," *IEEE J. Solid-State Circuits*, vol. 31, pp.1770-1778, Nov. 1996.
- [3] Q. Wang and S. Vrudhula, "Static power optimization of deep submicron CMOS circuits for dual vt technology," in *International Conference on Computer-Aided Design*, pp.490-494, 1998.
- [4] A. Chandrakasan, V. Gutnik and T. Xanthopoulos, "Data driven signal processing: an approach for energy efficient

- computing," *Proceedings of ISLPED'96*, pp.347-352, 1996.
- [5] S. Lee and T. Sakurai, "Run-time voltage hopping for low-power real-time systems," *IEEE/ACM Proc. DAC*, pp.806-809, June, 2000.
- [6] C. Wann et al, "CMOS with active well bias for low-power and RF/analog applications," *Dig. Tech. Papers Symp. VLSI Tech.*, pp.158-159, 2000.
- [7] Apollo user guide, Avant! co., 1998.
- [8] Diva interactive verification reference manual, Cadence Design Systems Inc., 1997.