## 修士論文

## 低消費電力高性能プロセッサに関する研究

平成 13 年 2 月 2 日

指導教官 桜井 貴康 教授

東京大学大学院 工学系研究科 電子工学専攻 96456

平林 雅之

## 第1章. 序論

#### 1.1. はじめに

マイクロプロセッサや半導体メモリなどに代表される大規模集積回路(VLSI)は,現在の高度情報化社会を支える基盤技術となっている.我々の身近なもので集積回路の恩恵を受けているものを考えてみると,テレビやゲームなどの家電製品をはじめとして,ラジオや携帯電話などの無線通信機器,電子手帳やウォークマンなどの携帯機器,カメラやビデオなど映像機器,多数のセンサや制御チップを搭載する車のエレクトロニクスなど,枚挙に暇がない.これらの個人向けの電子機器にとどまらず,医療サービス,カーナビゲーションシステム,交通システム,電子商取引,インターネット,移動体通信,電力供給システムなど,我々が陰に陽に享受する各種公共サービスの充実と拡大も集積回路の急速な発展の上に成り立っている.社会からの多種多様なニーズの存在によって,集積回路の持つ役割は今後ますます大きくなってゆくと思われる.

#### 1.2.集積回路の現在

現在の集積回路の主役は CMOS(Complementary MOS)回路である.この CMOS 回路技術を歴史的見地から見ると,その低消費電力性ゆえに,NMOS 回路やバイポーラ回路を抑えて LSI の主流の座を奪ったと見ることができる[1].NMOS から CMOS への移行は,より複雑なプロセスを用いることによる生産コストの上昇を招き,バイポーラから CMOS への移行は,回路の動作速度の低下を招いた.にもかかわらず CMOS は LSI の主流になったわけである.しかし近年,微細化加工技術の進歩によって回路の集積度は着実に向上を続け,一方でより複雑かつ高度なデジタルシステムを目指して回路規模が増大したことが,CMOS LSI にいくつかの新たな問題を引き起こした.そのひとつとして消費電力の問題がある.図 1.1 に ISSCC(世界固体回路会議)で発表されたマイクロプロセッサの消費電力のトレンド曲線を示す.70W という値は空冷下でのセラミックパッケージの限界であるが,100W を越えるプロセッサが登場するに至り,現在では消費電力の増大は"熱の危機"と呼ばれている.ハイエンドプロセッサの他に,近年注目を集めている携帯情報端末機器も低消費電力化を促す大きな要因となっている.これらはバッテリ駆動タイプのものが多く,長時間の使用に耐えるためには低電力化が必須となる.集積回路は現在,高速化と高機能化と同時に低消費電力化が大きな課題となっている.

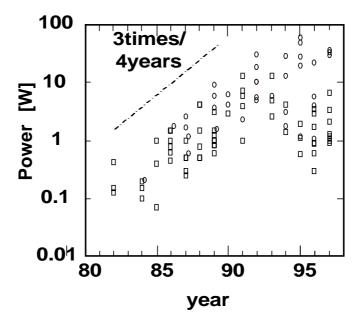


図 1.1 CMOS LSI における消費電力の推移 (ISSCC 発表の統計)

#### 1.3. CMOS 回路における消費電力と速度

CMOS 回路の消費電力・速度を検討するためにはまずその構成成分やパラメータ依存性などを示す必要がある. CMOS 回路の消費電力 P は式(1.1)のように表される.

$$P = afCV_{DD}^{2} + I_{SC}fV_{DD} + I_{IFAK}V_{DD}$$
 (1.1)

この式の第1項は充放電電流による消費電力の成分で a がスイッチング確率(活性化率) , f が動作周波数 , C が負荷容量 , そして VDD が電源電圧である . 第2項はゲートがスイッチングする時 , PMOS も NMOS もオンする瞬間があり , その時に流れる貫通電流成分である . 第3項はリーク電流による消費電力である . 消費電力成分の中で充放電電流成分とリーク成分は比較的簡単にパラメータ依存性を表現できるのに対し , 貫通電流成分の解析は課題の残る問題である . いくつかの解析式が提案されているがそれぞれに問題が残されており , 将来の低電源電圧化したときの予測がはっきりとわからなかった . この成分が将来どのように変化していくのかを知ることも消費電力を見積もる上で重要になっていくと考えられる .

消費電力の削減のためには式(1.1)から ,スイッチング確率を減らす ,負荷容量を減らす , そしてすべての項にかかってくる電源電圧を下げるという方法などが考えられる .

これらの解決法の中で,最も効果が大きいと考えられるのが電源電圧を下げるという方法である.しかしながら、電源電圧を下げることは駆動能力の低下による回路速度の低下を引き起こす。そこで回路性能を落とさずに電源電圧を下げる方法を考えると,閾値を低くする,並列処理をする,複数電源を使うなどの方法が考えられる。このうち最も一般的に行われている閾値を低くする方法について考える.

回路の遅延時間は一般的に次式のように表される[2].

$$D = \frac{k \cdot CV_{DD}}{\beta (V_{DD} - V_{TH})^{\alpha}} \tag{1.2}$$

よって、電源電圧を下げた時に遅延時間を維持するためには閾値を下げることが有効であることが分かる。

しかしながら, 閾値を下げると式(1.3)のような関係を持つリーク電流の増大につながってしまう.

$$I_{LEAK} \propto 10^{\frac{-V_{TH}}{S}} \tag{1.3}$$

つまり、閾値を下げるとリーク電流が指数関数的に増加してしまうのである。閾値を 0.7V から 0.1V に下げるとリーク電流は 5 , 6 桁増加してしまう。以前は充放電電流成分が消費電力のほとんどを占めていたため,閾値をある程度下げても全体の消費電力に影響はなかったが,電源電圧が下がってくるとこのリーク成分は重要な問題となってくる.その様子を模式的に示したものが図 1.2 である.この図からわかるとおり,低電源電圧領域では速度と消費電力の充放電電流成分とリーク電流成分との三者のトレードオフとなり,単に低電源電圧にすれば良いというわけではない.図中 A 点と B 点では遅延が同じなのにもかかわらず,消費される電力が異なる.つまりしきい値と電源電圧を適切に選ぶことが低消費電力化には重要である.従来はプロセスが決まれば閾値も一定に決まり,回路自体も一定電源電圧で動作させるのが一般的であったが,電源電圧や閾値を制御して,必要とされる速度を保ちながら低消費電力を実現させる方法も広く使われるようになってきている.

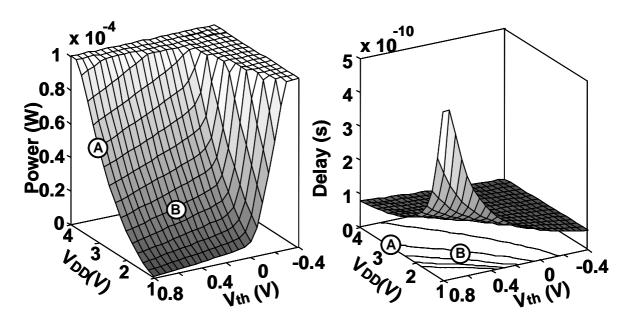


図 1.2 電源電圧・閾値と動作速度・消費電力のトレードオフの概念図.

#### 1.4. 電源電圧制御による低消費電力化

電源電圧を低くすれば消費電力は低くすることができる.しかし,図 1.2 にもあるように電源電圧を低くすると,回路の遅延が大きくなってしまう.そこで速度が必要なときには電源電圧を高くし,スタンバイ時など速度が必要ないときには電源電圧を低くするというVoltage Hopping という手法がある 本節ではこの Voltage Hopping を Run-time で行う Run-time Voltage Hopping の基礎について説明する.

#### 1.4.1.Run-time Voltage Hopping の基礎

汎用プロセッサなどでは、時間によってデータ処理量が大きく変動する.要求される処理量に応じ、プロセッサの動作周波数を変化させると同時に電源電圧をも変化させることで、低消費電力化を図ることが考えられている.これを実現するために多くの可変電源電圧制御アルゴリズムが提案されている[3-6].しかし、その中にはアプリケーションコンパイル時に電源電圧の割り当てをスケジューリングするものや,特化されたハードウェアを必要とするものが多く,必ずしもリアルタイムアプリケーションに実用できるとはいえない.というのはリアルタイムアプリケーションの多くはデータに依存しており,コンパイル時にはアプリケーションの実行に必要なパフォーマンスを予測することが困難だからである.また,特化されたハードウェアを必要とする場合は従来のハードウェアリソースを十分に利用することができず,実用化は難しい.

これに対して、run-time voltage hopping(VH)と呼ばれる可変電源制御方式 [7]は従来のハードウェアリソースに簡単な外付け制御回路を追加し、アプリケーションに電圧制御アルゴリズムを組み込むことによって実現することができる.VH は、アプリケーションのリアルタイム性を保証するだけでなく、従来のハードウェアリソースをそのまま低消費電化することができる.VH において電力制御アルゴリズムを組み込んだアプリケーションは、汎用プロセッサ上で実行する.アプリケーションの実行具合によって電力制御アルゴリズムは適切な電源電圧及び周波数を決定する.実際のプロセッサの電源電圧と動作周波数の変更は電力制御回路によって行われる.

VH の基本アイデアを図 1.3 に示す .このアプリケーションは 25 秒間で  $5\times10^8$  サイクルの演算が必要で、消費電力 P、ワークロード  $N_{\rm CYC}$ 、電源電圧  $V_{\rm DD}$  の関係は  $P=V_{\rm DD}{}^2\times N_{\rm CYC}$  x  $10^{-9}$  である。また,クロック周波数  $f_{\rm CLK}$  と最低電源電圧  $V_{\rm L}$  との関係は  $f_{\rm CLK}=V_{\rm L}\times10^7$  である.図

1.3(A)はプロセッサを一定電圧,一定周波数で動作させた場合であり,消費電力は 12.5J となっている.しかし図 1.3(A)をみてわかる通り,プロセッサはリアルタイムを保証する制限時間より前に処理を終わってしまっている.図 1.3(C)のように制限時間ぎりぎりで処理が終了するような一定電圧,一定周波数でプロセッサを動作させるのが理想的である.しかしリアルタイムアプリケーションでは処理するデータ列によってワークロードが変化するため,これを実現するのは難しい.

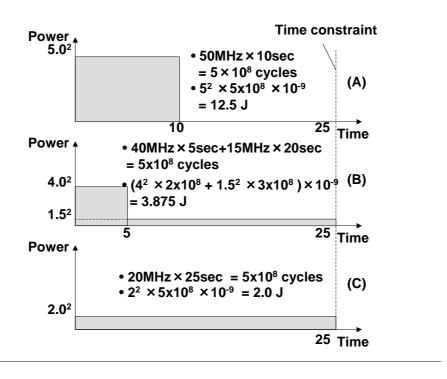


図 1.3 Voltage Hopping の概念図.

そこで図 1.4 のように制限時間をより細かい timeslot に分割する.あるタイムスロットでの  $f_{VAR}$  と  $V_{VAR}$  の算出方法は,まずパワー制御チップの内部カウンターから現在の時間  $T_C$  を読み込む.次に現在のタイムスロットにおけるターゲットタイム  $T_{TAR}$  と実行時間のワーストケース  $T_{fi}$  を算出する. $T_{TAR}$  と  $T_{fi}$  はそれぞれ  $T_{TAR} = \sum T_S - T_C$  、  $T_{fi} = T_{TD} + T_{S3}(f_{CLK}/f_1)$  として算出される.ただし  $f_i$  (i=1,2,3...)とする.ここで注意すべきは  $f_i$  がひとつ前のタイムスロットにおいて  $f_{VAR} = f_1$  のときは  $T_{fi} = T_{S3}(f_{CLK}/f_1)$ となる.算出された実行時間  $T_{fi}$  がターゲットタイム

 $T_{TAR}$ をこえない最小の周波数 $f_i$ がクロック周波数 $f_{VAR}$ と決まる.

図 1.3(B)によると,この方法で約70%の低消費電力化が実現されている.

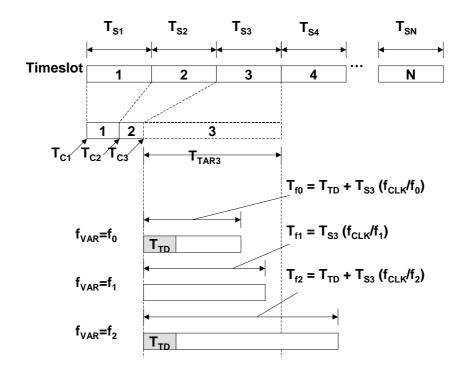


図 1.4 適用周波数の算出.

### 1.4.2.VH 実装上の問題点

VH は従来の CPU や DSP などのハードウェアに大きな変更を加えることなく実装することができるのが大きな利点である.Seongsoo Lee らによると,VH を実装すときに電源電圧や周波数を変化させるときの過渡遅延が低消費出力化に大きな影響をおよぼす[7]. 図 1.5 によると過渡遅延が小さければ小さいぼど VH によって低消費電力化することができる.現在のテクノロジーでは,この過渡遅延は DC-DC コンバータなどによって電圧を変化させるためμ秒のオーダかかってしまっている.高効率化のために,外部にあらかじめ複数の電源を用意し,チップ内部のスイッチで変化させることで 100n 秒のオーダで高速に変化させることが考えられている.電源電圧を高速に変化させたとき,従来の VLSI では見られなかっ

た問題が生じる可能性がある.

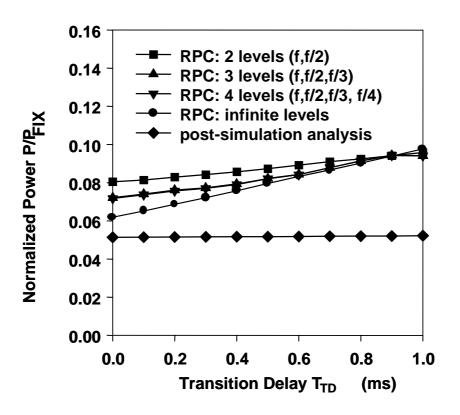


図 1.5 消費電力と過渡遅延の関係

( Seongsoo Lee and Takayasu Sakurai, "Run-time Voltage Hopping for low-power Real-time Systems" )

### 1.5. しきい値電圧制御による低消費電力化

VLSIの世界では、高速化やコスト削減のために微細化を続けてきた.それに伴って低しきい値のプロセスが広く用いられるようになっている.しきい値( $V_{TH}$ )が低くなると、回路速度があがる半面、式(1.3)にあるようにリーク電流が指数関数的に増加する.CMOS 回路の消費電力成分は式(1.1)にあるように主に次の 3 つである.

1. 充放電電流による消費電力

- 2. リーク電流による消費電力
- 3. 貫通電流による消費電力

低電圧低しきい値領域では、この3つの成分のうち第1項は電源電圧の2乗に比例するため、低電源電圧化が進むほど消費電力に占める割合は小さくなっていく、一方、第2項はしきい値が低くなるにつれて、指数関数的に増加することから CMOS 回路全体の消費電力に占める割合も指数関数で大きくなる、低電圧低しきい値領域ではリーク電流による消費電力が無視できない問題となっている、

スタンバイ時にリーク電流を削減する回路は従来から提案されており,従属接続した MOSFET を制御することでリークを切る MTCMOS[8]やクリティカルパス以外のトランジスタのしきい値をあらかじめ高く設計しておく Dual- $V_{TH}[9]$  ,さらに基板バイアスを用いて動的なしきい値電圧制御を行う VTCMOS(Variable-Threshold-Voltage CMOS)[10]などがある.本節では MTCMOS,VTCMOS,Dual- $V_{TH}$ の基本概念について説明する.

#### 1.5.1.MTCMOS

MTCMOS 回路の基本構成図を図 1.6 に示す.

MTCMOS 回路では論理回路部を低しきい値電圧で動作させ,スタンバイ時には高しきい値のスイッチによって電源線と論理回路を切り離す.これによってスタンバイ時のリーク電流が大幅に削減できるが,一方で動作時のリーク電流には全く効果がない.動作時のリーク電流による消費電力を抑えるためには動的にしきい値電圧を制御する必要がある.

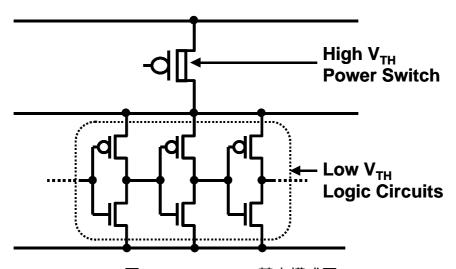


図 1.6 MTCMOS の基本構成図

#### **1.5.2.VTCMOS**

VTCMOS 回路の基本構成を図 1.7 に示す.

VTCMOS 回路では基板バイアス効果を利用してトランジスタのしきい値電圧を制御する.例えばトランジスタは, $V_{TH}$ が 0.15V になるように製造する.動作時には,基板にバイアスを掛けずに,低  $V_{TH}$  のまま回路を動作させる.スタンバイ時と  $I_{DDQ}$  テストの際は基板に負のバイアスを掛けて, $V_{TH}$  を例えば 0.55 V にする.そうすることでスタンバイ時などにリークによる消費電力を大幅に削減することができる.基板電位はポンプ回路で構成した基板電圧発生回路(SSB: Self-Adjusting Threshold Scheme)やスイッチ回路を使って制御する.

VTCMOS ではチップ上の基板コンタクトをすべて配線接続しなくてはならないため, 個々に近くの電源線に落としていた CMOS と比べてチップ面積が大きくなる可能性がある. あるいは,過去に設計した CMOS のレイアウトにそのような修正を後から加えることは一般的に困難となるので,過去の設計資産を活用できなくなる懸念がある.

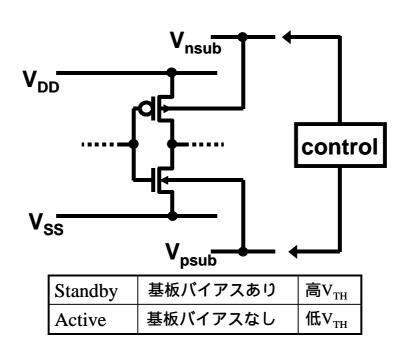


図 1.7 VTCMOS の基本構成図

#### 1.5.3.dual V<sub>TH</sub> テクノロジ

前述の MTCMOS や VTCMOS はスタンバイ時のリーク電流はカットできるものの,アクティブ時のリーク電流はカットできない.アクティブ時のリークもカットできる手法が dual  $V_{TH}$  デクノロジ(dual- $V_{TH}$ )[9]である.dual- $V_{TH}$ では速度のクリティカルな部分では,トランジスタのしきい値をリーク電流が許容できる範囲でできるだけ低下させてトランジスタの駆動力を増大させて高速化する.一方,スタンバイ時にできるだけ電力を抑えたい部分や,速度が要求されない部分には高いしきい値のトランジスタを用いることでリーク電流を抑える.このようにリーク電流削減重視の部位と,速度重視の部位とでトランジスタのしきい値を 2 種類使い分けていくやりかたを dual- $V_{TH}$  と呼んでいる.

dual-V<sub>TH</sub>の回路方式を図 1.8 に示す.前述のようにクリティカルパスには低しきい値,高速トランジスタを用い,ノンクリティカルパスには高しきい値,低速トランジスタを用いることで電力を削減している.

低しきい値を実現するためには、とくに PMOS に工夫が必要である.しきい値が低くなると、ゲート電圧を 0V とした場合に確実に電流を OFF しにくくなる.このためトランジスタの設計法やトランジスタを構成する材料は従来にはない工夫されていることもある.

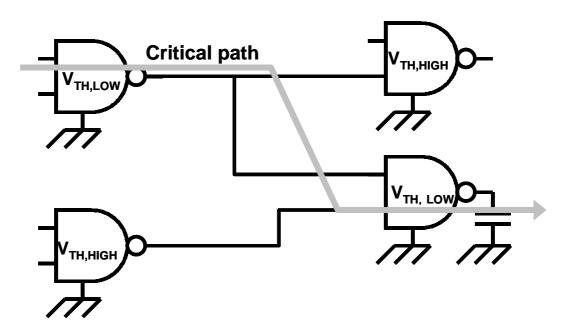


図 1.8 dual-V<sub>TH</sub> 回路方式

#### 1.6. 本論文の構成

第1章では本研究の背景を述べた.第2章では電源電圧制御による低消費電力化について述べる.電源電圧制御による低消費電力化として本研究では,低電圧下における SRAM の問題と乗算器における加算回路形式の検討について述べる.第3章ではしきい値制御による低消費電力化について述べる.しきい値制御による低消費電力化として,本研究では基板バイアス制御を可能にする自動設計手法, $V_{TH}$ -hopping によるリーク電流削減技術, $dual-V_{TH}$ の簡易設計手法と設計指針について述べる.最後に第4章で結論を述べる.

## 第2章. 電源電圧制御による低消費電力化

第1章で述べたように、CMOS 回路の消費電力は電源電圧の2乗に比例するため電源電圧を下げることは低消費電力化にとって非常に有効である。VH は高いパフォーマンスを必要とするときは高い電源電圧で回路を駆動し、高いパフォーマンスが必要ないときは低い電源電圧で消費電力を削減する。低電圧動作時は回路のしきい値に対して電源電圧が通常用いられるより 1/2 、1/3 程度とかなり低い状態で動作することになる。これにともなって適切な回路形式や動作に問題がでてくることが考えられる。そこで本章では、低電源電圧下における SRAM の問題と、VH に用いる乗算回路における加算回路形式について検討する。

### 2.1. 低電源電圧下における SRAM の問題

#### 2.1.1.SRAM の現状

SRAM はプロセッサ内のキャッシュメモリや携帯機器のメモリとして用いられている. DRAM と比較すると, SRAM セルの面積が大きいため小容量だが, リフレッシュ動作が不要なので待機時の電力が小さく, また相補のビット線によって高速アクセスが可能であるとい

う特徴がある . CPU の高速化と主記憶の大容量化に伴い , SRAM も高速大容量化の歴史を歩んできた .

最高速の SRAM は Bipolar や BiCMOS の SRAM であるが、消費電力の点で優れている CMOS SRAM が現在の主流である.特に最近急速に需要を増しているバッテリ駆動型携帯電子機器においては SRAM の低電力化が必須である.CMOS では低電源電圧化による電力削減が有効であるため、微細化技術や回路の改良による低電源電圧化が進んでいる.この状況をうけて低電圧化で高速動作を保証する回路技術は重要性を増している.VHを実装した場合、ハードウェアの電源電圧が回路動作中に変化し、しきい値に対して通常よりも低電圧で動作させる状態が存在する.そこで本節では電源電圧が動作中に変化したときに SRAM の動作が保証されるのかについて検討する.

#### 2.1.2.低電源電圧下における SRAM 動作の検討

検討に用いた SRAM の RAM セルは NMOS を負荷とする標準的なものであり,回路図を図 2.1(b)に示す.電源電圧が変化するときに正常に動作するかを確かめるため,電源電圧が高い状態から低い状態へ変化したとき,低い状態から高い状態へ変化したときの二つを考える.そして,それぞれについて書き込みと読み込みのふたつの動作について調べる. HSPICEによるシミュレーションの結果,電源電圧が低い状態から高い状態へ変化するときは動作異常が見られなかったが,高い状態から低い状態へ変化するときに異常な動作をする可能性がある 異常な動作をする可能性があるときの HSPICE シミュレーションによる波形を図 2.1(a)に示す.図 2.1(a)をみると,電源電圧が急激に低くなったときにビット線の電圧が電源電圧に追従しきれずに電源電圧よりも高い状態になってしまっている.これは,電源電圧が下がったときにビット線,基板間のキャパシタンスに蓄えられた電荷の逃げ道がないためにおきると思われる.

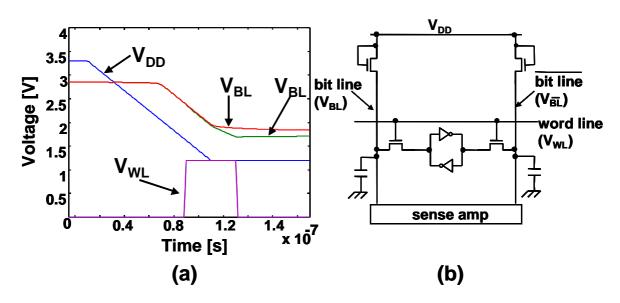


図 2.1 電源電圧を 3.3V から 1.2V に下げたときの SRAM (NMOS 負荷)動作 (a) 各ノードの動作時の波形 (b) SRAM (NMOS 負荷)の回路図

電源電圧よりもビット線の電位が高い状態のままセンスアンプで増幅しようとすると図 2.2 のようなことが起きる.図 2.2(a)にビット線の電位差とセンスアンプの出力電圧の関係を,図 2.2(b)にセンスアンプの回路図を示す.図 2.2(a)によると,ビット線の電位が電源電圧と等しいときにはビット線の電位差が 200mV 程度でセンスされているが,ビット線の電位が電源電圧よりも高いときには電位差が 600mV 程度ないとセンスされなくなってしまっている.センスアンプの役割はビット線の微小な電位差 V を感知し,大きなキャパシタンスを持つビット線をフルスイングさせなくてもセンスアンプによって出力電圧がフルスイングすることで高速化と低消費電力化を図ることである.ビット線の電圧が電源電圧よりも高い状態でセンスすると読み取りに必要な電位差が通常以上に必要となってしまい,図 2.2(c)のように読み取りにかかる時間が大幅に増加することになる.これがエラーの原因となる可能性がある.

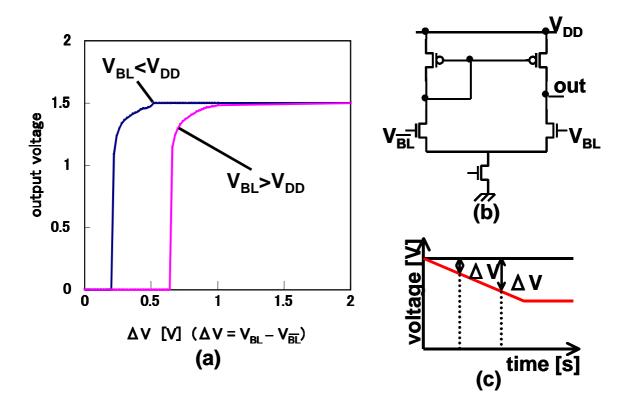


図 2.2 ビット線の電圧が電源電圧よりも高い状態でのセンスアンプの動作 (a)センスアンプの出力電圧とビット線の電位差 V の関係 (b)センスアンプの回路図 (c)ビット線の電位差 V とセンスに要する時間の関係

図 2.3 のように PMOS を負荷とし読み取り,書き込み時にプリチャージすることでビット線の電位が原因となるエラーを回避できる.したがって,VH に用いる SRAM は PMOS によってプリチャージするタイプのものが適当である.

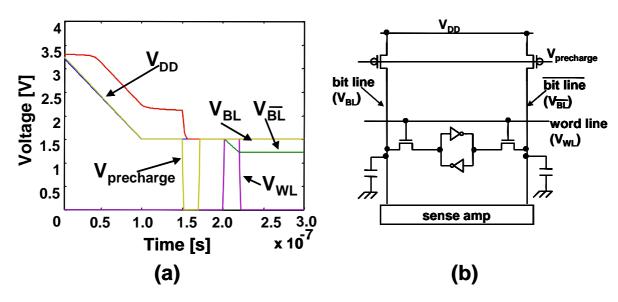


図 2.3 電源電圧を 3.3V から 1.2V に下げたときの SRAM (PMOS 負荷) 動作 (a) 各ノードの動作時の波形 (b) SRAM (PMOS 負荷) の回路図

### 2.2. 乗算器における加算回路形式の検討

マルチメディアデータをハードウェアで処理することを考えたとき,乗算がシステムの速度,消費電力に大きく影響する.乗算回路を低消費電力化することはシステム全体の低消費電力化にとって大きなインパクトとなる.第1章で述べたVH方式をシステムに適用したときは,回路動作中に電源電圧が大きく変化するため,従来からよく用いられている回路形式が乗算器にとって最適であるとは限らない.

M.I.Elmasry らによると図 2.4 をみてわかるように乗算器における消費電力の約 40% ~ 70% は array 状の加算器で消費される[11]. つまり乗算器に用いる加算器を低消費電力化すれば乗算器全体としても低消費電力を実現することができる. そこで本節では VH に用いるのに最適な加算器回路形式について検討する.

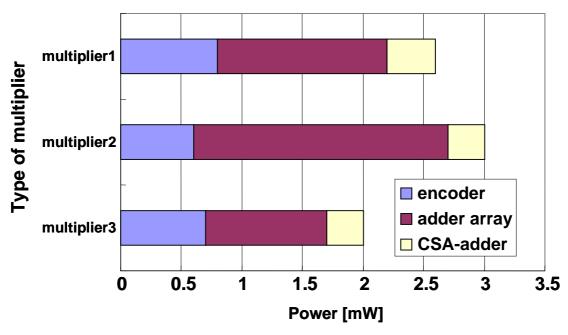


図 2.4 乗算器における消費電力の内訳

検討に用いる回路形式は CMOS Logic(CMOS), Complementary Pass-Transistor Logic (CPL), Transmission Gate Logic (TG)の3つであり、それぞれの形式でFull Adder(FA)を構成した[12-15]. 検討に用いる加算回路の回路図を表 2.1 に、測定に用いたチップ写真と測定回路を図 2.5 に示す.比較検討は FA の delay と power の 2 項目について行い、比較には次のような方法をとった.FA の入力に前状態として3つの入力信号を与えておく.そして入力信号を変化させると、それに追従して出力も変化する.これを後状態とする.前状態から後状態へ変化するのに要する transient time の最悪値を遅延とした.つまり FA は3入力であるから8通りの入力信号が考えられるので、前状態と後状態で8×8=64通りのワーストケースが遅延となる.消費電力については64通りの平均値を消費電力とした.

	CMOS	CPL	TG
回路図	A 4 B 4 A 4 A 4 B 4 C		
面積	48μm × 33μm	$73\mu m \times 33\mu m$	50μm × 33μm
総ゲート幅	160µm	160µm	156µm
トランジス タ数	28	28	20

表 2.1 比較に用いた加算回路形式

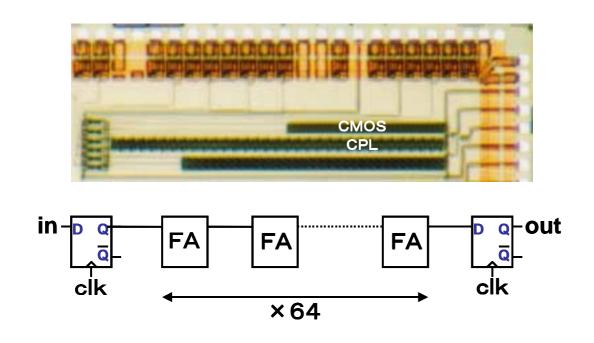


図 2.5 チップ写真と測定回路図

CMOS-FA,CPL-FA,TG-FA それぞれについて, HSPICE を用いたシミュレーションおよび チップ\*の実測を行う .デバイスのパラメータはすべてのゲート長をプロセスの最小値とする. ゲート幅については CMOS-FA と TG-FA は NMOS と PMOS のゲート幅の和が同じになる ようにし, CPL-FA は NMOS のゲート幅が和の 1/2 になるようにした. つまりそれぞれの回 路形式で面積をほぼ一定にした条件で比較検討を行った.この条件の下で delay が最小にな るようにパラメータを最適化する.

HSPICE によるシミュレーションは複数のプロセスについて行い .それぞれのプロセスに おいて最適化したパラメータを用いた、さらに今回用いた3種類の回路形式はどれもスタテ ィックな回路のため、プロセスによるしきい値のばらつきも考慮しなくてはならない、そこ で,それぞれのプロセスにおいて,ティピカルなしきい値から±0.1Vのばらつきがあると仮 定してシミュレーションを行い、しきい値をばらつかせた中で最悪の値をそのプロセスの消 費電力と遅延とする.

CMOS と CPL について実際に測定を行った.実測する際は一定以上の遅延がないと測定 が不可能なため,図 2.4 のように回路を組んだ.[16]クロックの立ち上がりで DFF から FA に 信号が入力され,64段のFAを信号が伝播する.1クロックサイクルの間に信号が出力段の DFF まで到達すれば出力は変化し,到達しなければ出力は変化しない.クロックサイクルを 変化させることによって , 64 段分の遅延を測定することができる . 消費電力については Vpp から Gnd に流れる電流を測定し,消費電力を計算した.

シミュレーションおよび実測の結果を図 2.6 , 図 2.7 に示す . 比較においては消費電力 , 遅延ともに,それぞれのプロセスで CMOS 形式を 1 として正規化して表示してある.

<sup>\*</sup> 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)の協力で行われたものである.

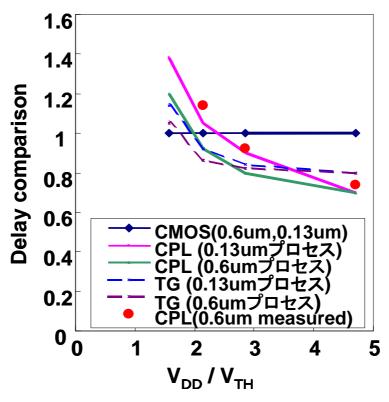


図 2.6 回路遅延と  $V_{DD}/V_{TH}$ の関係

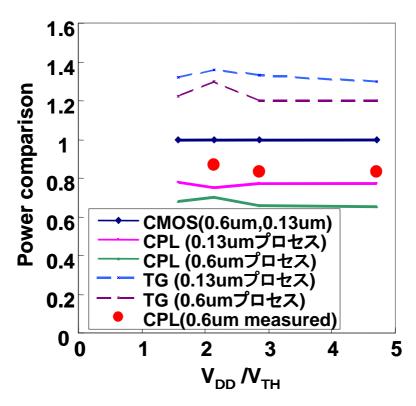


図 2.7 消費電力と V<sub>DD</sub>/V<sub>TH</sub> の関係

VH に用いることを考えると電源電圧を通常よりも下げて動作させることがあるため,  $V_{DD}/V_{TH}$  が 2~3 のところでも動作させることになる .しかも現在の高性能なプロセッサに VH を用いて MPEG4 などのリアルタイムアプリケーションを処理させる場合, $V_{DD}/2$  や  $V_{DD}/3$  などの低い電源電圧で動作させることがほとんどである . SH4 プロセッサによる MPEG-4 のリアルタイム処理では,低電圧で動作させる時間が全体の約 98%を占めるという実験結果も得られている .

図 2.6 を見ると,delay に関して  $V_{DD}/V_{TH}$  が 2 より小さいときは,従来からよく用いられている CMOS が最も遅延の少ない回路形式となっているが, $V_{DD}/V_{TH}$  が 2 以上では CPL が最も遅延の少ない回路形式となっている.図 2.7 に消費電力と  $V_{DD}/V_{TH}$  の関係を示す.シミュレーション結果をみるとわかる通り CPL と TG を比較すると TG は NMOS と PMOS を並列に組み合わせているため,NMOS のみで構成されている CPL に比べるとジャンクション容量が 2 倍となってしまう.そのため,TG は CPL に比べると低消費電力に向いていないと考えることができる.CPL は CMOS に比べてシミュレーションで 20%~30%,実測でも 10%低消費電力であることがわかる.

VH において 2 種類の電源電圧を用意して低消費電力化を図ることを考える . 2 種類の電源電圧  $V_{\rm DD\_HIGH}$  と  $V_{\rm DD\_LOW}$  は

$$f_{LOW} = \frac{1}{2} f_{HIGH} \tag{2.1}$$

つまり

$$Delay(V_{DD\_LOW}) = \frac{1}{2} Delay(V_{DD\_HIGH})$$
 (2.2)

を満たす  $V_{DD\_HIGH}$  と  $V_{DD\_LOW}$  とする.また前述のように SH4 プロセッサで MPEG4 のリアルタイム処理を実行したとき, $V_{DD\_HIGH}$  の状態は全体の実行時間のわずか 2% である.そのときの消費電力  $P_{TOTAL}$  は

$$\begin{split} P_{TOTAL} &= 0.02 P(V_{DD\_HIGH}) + 0.98 P(V_{DD\_LOW}) \\ &\propto 0.02 f_{HIGH} V_{DD\_HIGH}^2 + 0.98 f_{LOW} V_{DD\_LOW}^2 \end{split} \tag{2.3}$$

となる.式(2.3)、式(2.1)より P<sub>TOTAL</sub>は

$$P_{TOTAL} \propto 2V_{DD\_HIGH}^{2} + 49V_{DD\_LOW}^{2}$$
 (2.4)

となる.

式(2.4),図 2.6,図 2.7 から V<sub>DD HIGH</sub>/V<sub>TH</sub> と消費電力の関係を求め,それを図 2.8 に示す.

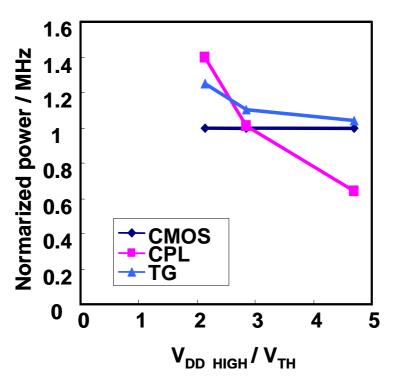


図 2.8 V<sub>DD HIGH</sub>/V<sub>TH</sub> と単位速度あたりの消費電力の関係

図 2.8 より単位速度あたりの消費電力は  $V_{DD\_HIGH}/V_{TH}$  が 3 以上の領域では CPL が最適な回路形式であり,3 以下の領域では CMOS が最適な回路形式となる.今後テクノロジーが進歩すると  $V_{DD}/V_{TH}$  は小さくなっていくと思われるため,CMOS を回路形式として用いるのが適当であると考えられる.

#### 2.3.まとめ

VH を用いる際の SRAM の問題点を考察した. VH を SRAM に用いる場合は NMOS によってプリチャージするタイプのものはセンスアンプでビット線の微小な電位差をセンスする際に誤動作を起こす可能性がある. したがって PMOS によってプリチャージするタイプが適している.

VH を用いる際に最適な乗算回路について検討した。乗算器の消費電力の 7 割から 8 割は乗算器を構成している加算回路で消費されるので,VH を用いる際に最適な加算回路形式について,シミュレーションおよび試作チップの測定により検討した。 $V_{DD\_HIGH}/V_{TH}$  が 3 以上

の領域では CPL が最適な回路形式であり 3 以下の領域では CMOS が最適な回路形式である.またテクノロジーが進歩すると  $V_{\rm DD}/V_{\rm TH}$  は小さくなる傾向があるので,今後も CMOS が広く用いられると思われる.

## 第3章. しきい値制御による低消費電力化

#### 3.1. しきい値制御の重要性

LSI デバイスの微細化は集積化がムーアの法則に従うように現在も着実に進んでいる.それに従い,低消費電力,耐圧の観点から電源電圧もまたスケーリングにしたがって低電圧化が進んでいる.ITRS[17](International Technology Roadmap for Semiconductors)によると,2011年には電源電圧(V<sub>DD</sub>)が 0.5V 程度になると予想されている.電源電圧 0.5V におけるしきい値と回路遅延,消費電力の関係を図 3.1 に示す.図 3.1 からわかるように,低電源電圧下ではパフォーマンスを維持するためにしきい値(V<sub>TH</sub>)を下げる必要がある.しかしながらしきい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してしまう.現在では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してしまう.現在ではリーク電流によるスタティック消費電力は充放電によるダイナミック消費電力の 10%程度と無視できるくらい小さいが,近い将来スタティック消費電力が無視できなくなり,リーク電流を削減することが必要となってくる.リーク電流を削減するにはしきい値をなんらかの手段で制御する,もしくは適切な値のしきい値を選択することが有効である.そこで本章ではしきい値制御による低消費電力化について述べる.

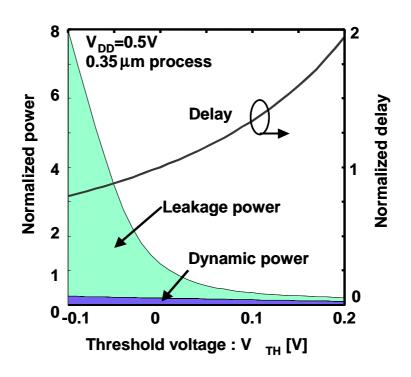


図 3.1 しきい値と回路遅延,消費電力の関係

### 3.2. 基板バイアス制御を可能にする自動設計手法

しきい値を制御するために基板バイアスを制御しようとすると,基板バイアス用の電源線が必要となる。自動配置配線によって基板バイアス制御が可能なVLSIの設計を行う場合,スタンダードセルライブラリに既存のものを用いることができず,図 3.2 のように  $V_{NSUB}$  線や  $V_{PSUB}$  線を加えた新たなスタンダードセルライブラリの作成,機能チェックおよびエラーの導入を行わなければならず容易ではない.そこで既存のスタンダードセルライブラリを用い,基板バイアス制御が可能な回路の設計方法を提案する.

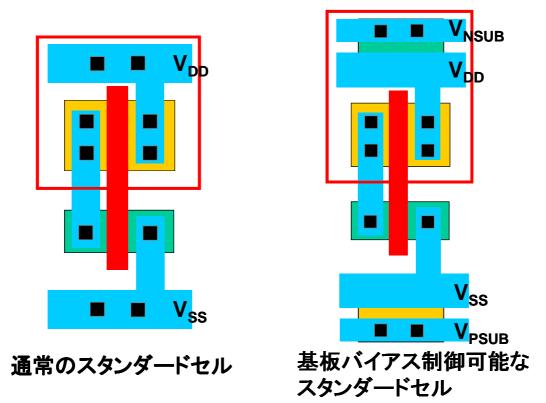


図 3.2 通常のスタンダードセルと基板バイアス制御可能なスタンダードセル

今回の設計に用いたツールは Synopsys Design Compiler [18], Avant! Apollo [19] , Cadence DF [20]である. 設計手法のフローチャートを図 3.3 に示す. 本設計手法はライブラリには既存のものを用いるので,HDL 記述から Synopsys による論理合成までは従来の LSI 設計と全く同じである.配置配線ツールによる自動配置配線もほぼ通常の設計を同じである.通常の自動配置配線の場合は回路コアの周辺に  $V_{DD}$ ,  $V_{SS}$  リングを作成するが,本設計手法ではそれに加えて基板バイアスを制御するために  $V_{PSUB}$  と  $V_{NSUB}$  リングを図 3.4 のように追加しなければならない.

自動配置配線によってレイアウトを作成後、このままでは基盤は  $V_{SS}$  に固定され、NWELL は  $V_{DD}$  に固定されてしまい基板バイアス制御ができないため、基板コンタクトを除去しなければならない。基板コンタクトの除去には SKILL スクリプト[21]を用い、自動的に行う。 SKILL スクリプトでは各レイヤの論理がとれるので、残すべきコンタクトレイヤと除去すべきコンタクトレイヤを定義し、その後除去すべきコンタクトのみをレイアウトから除去することになる。基板コンタクトを除去する SKILL スクリプトのメイン部分を図 3.5 に示す。

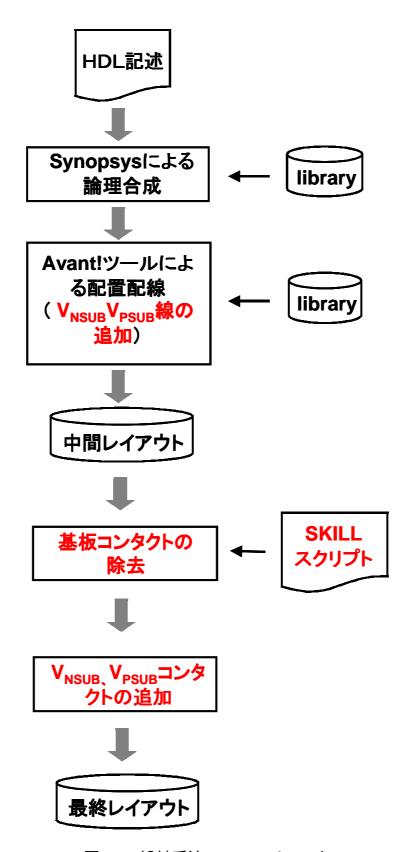


図 3.3 設計手法のフローチャート

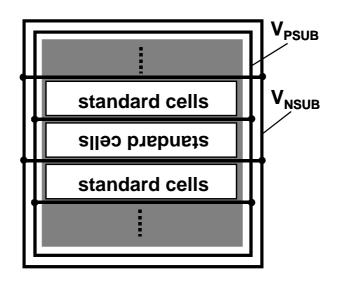
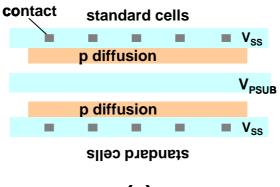


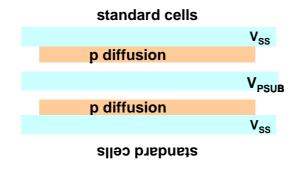
図 3.4 V<sub>PSUB</sub> と V<sub>NSUB</sub> 線の追加

図 3.5 基板コンタクト除去の SKILL スクリプトメイン部分

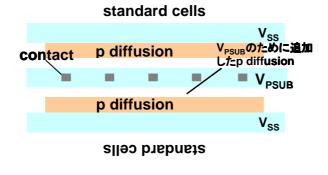
基板コンタクト除去後,P 基板を  $V_{PSUB}$  に,N 基板を  $V_{NSUB}$  に固定する.あらかじめ  $V_{PSUB}$  と  $V_{NSUB}$  線はスタンダードセルの上下に這わせてあるので,スタンダードセル列の間に P 拡散層と N 拡散層を追加し,コンタクトで  $V_{PSUB}$  と  $V_{NSUB}$  線をそれぞれ固定する.  $V_{PSUB}$  線と P 拡散層のコンタクトによる固定を図 3.6 に示す.



(a)



(b)



(c)

図 3.6 V<sub>PSUB</sub> 線と P 拡散層のコンタクトによる固定 (a)配置配線後のレイアウト

- (b) レイアウト(a)から基コンタクトを除去
- (c) レイアウト(b)に V<sub>PSUB</sub> を固定する拡散層とコンタクトを追加

 $V_{NSUB}$  と N 基板も図 3.6 と同様に行うことで , 既存のスタンダードセルライブラリを用いて , 基板バイアス制御が可能な回路レイアウトを得ることができる .

 $V_{PSUB}$  と  $V_{NSUB}$  線を追加することで基板バイアスを制御することが可能になる一方でレイアウトの面積は増加することになる。この面積オーバーラップはひとつのセルでみると図 3.7 のように 15% 程度であるが,チップ全体を考えるとメモリ部や入出力部など基板バイアス制御を用いない部分もあるため,面積オーバーラップは 5% 程度と考えられる.

本設計手法に基づいて設計されたレイアウトを図 3.7 に示す . 本レイアウト\*)は  $0.6\mu m$  , 3 層メタルプロセスにより設計され , レイアウトサイズは  $2.0 mm \times 2.1 mm$  である .

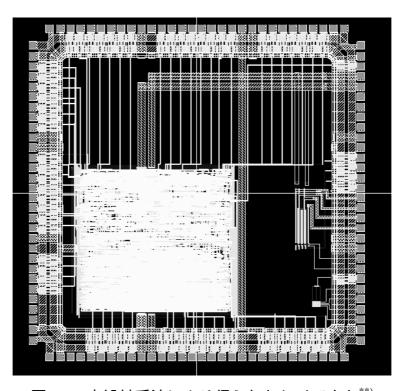


図 3.7 本設計手法により得られたレイアウト\*\*)

 $<sup>^{*)},^{**)}</sup>$ 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)の協力で行われたものである.

# 3.3. しきい値ホッピング(V<sub>ТН</sub>-hopping)によるリーク電流 削減技術

本章第1節で述べたように、VLSI は微細化にともなって高速化、低電圧化が進んでいるとともに、パフォーマンスの要求からしきい値を下げることが要求されている。図 3.1 にあるようにしきい値を下げると、リーク電流が指数関数的に増加し、それに伴ってリーク電流によるスタティックな消費電力も指数関数的に増加してしまう。今後もしきい値が下がっていくと電荷の充放電による消費電力よりもリーク電流による消費電力が深刻な問題となってくる。本節ではしきい値を動的に変化させることにより、リーク電流を削減するしきい値ホッピング(V<sub>TH</sub>-hopping)を提案する。

待機時にリーク電流を削減する回路は従来から提案されており、従属接続した MOSFET を制御することによりスタンバイ時のリーク電流を制御する MTCMOS[8]などがある。しかしながらこの方法ではアクティブ時のリーク電流は削減されない。また、クリティカルパス以外のトランジスタのしきい値をあらかじめ高く設計しておく dual  $V_{TH}[22]$ ではクリティカルパスのリーク電流は常に流れてしまうため、十分にリーク電流を削減することができない。

そこで本研究では MOS のしきい値をプロセッサの負荷に応じて変化させるしきい値ホッピング ( $V_{TH}$ -hopping)を提案する.この手法はプロセッサの負荷の軽いときにはしきい値を高くし,同時に周波数を下げることで無駄な消費電力を削減する方式である.この方法は動的に電源電圧を制御する dynamic  $V_{DD}$  scaling(DVS)[23]に似ているが,DVS はダ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・低しきい値の場合に有効な手法である.

図 3.8 にプロセッサの負荷と消費電力の関係を示す.低いしきい値 V<sub>TH\_LOW</sub> はプロセッサの負荷が重いときにのみ適用され,負荷が重くなく,それほど高いパフォーマンスが要求されないときは要求されるのに足るだけのパフォーマンスを実現できる程度までしきい値を高くすることで消費電力を削減する.図 3.8 の破線は周波数制御のみを行った場合であり,V<sub>TH</sub>-hopping を行った場合の方が明らかに消費電力の点で優れている.

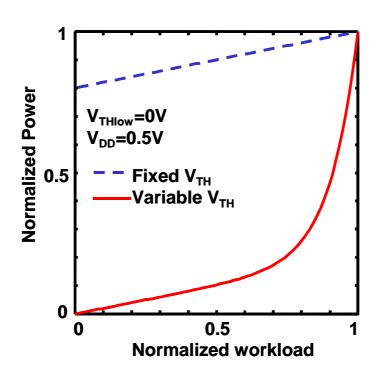


図 3.8 プロセッサの負荷と消費電力の関係

#### 3.3.1.V<sub>TH</sub>-hopping の概要

図 3.9 に本手法の概要を示す.回路はコア部である Target processor としきい値および周波数をコントロールする Power control block で構成される Power control block は DVS と同様のソフトウェアで制御することが可能である.Target processor 内部について,速度が問題となるクリティカルパスについては,そのしきい値を Power control block から出力された VThlow\_Enable および VThhigh\_Enable 信号で制御する.また同時に動作周波数も $f_{\text{CLK}}f_{\text{CLK}}/2$ に変化させる.負荷が重いときには VThlow\_Enable をオンさせ,クリティカルパスのしきい値を低くし $(V_{\text{THlow}})$ ,動作周波数を高くする.一方負荷が軽いときにはクリティカルパスのしきい値を高くし $(V_{\text{THhigh}})$ ,動作周波数を下げる $(f_{\text{CLK}}/2)$ ことで消費電力を削減する.

- クリティカルパスのない場所の制御方法としては,
- 1. クリティカルパスと同じしきい値( $V_{THlow}$ , $V_{THhigh}$ )を用いるようにし,動的にしきい値を変化させる.
- 2. あらかじめ高しきい値( $V_{THhigh}$ )を設定し ,常にリークが流れないようにする .( $V_{TH}$ -hopping + dual  $V_{TH}$ )

の 2 通り考えられる.2.の場合はアクティブ時の消費電力をより削減することが可能であるが,図 3.9 にあるようにしきい値を  $V_{TH\_noncrit}$  に設定し,ウェルを分離する必要がある.1.の場合はすべて critical paths のような制御をすればよいため, $V_{TH\_noncrit}$  を用意する必要がなくなる.2 つの手法についての比較は後述する.

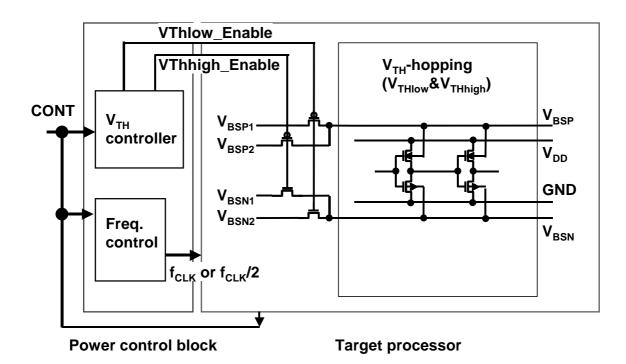


図 3.9 V<sub>TH</sub>-hopping の概要

このとき,より多くの  $V_{TH}$ ,f を用意し,更に消費電力を削減する方法も考えられる.そこでMPEG4 エンコード処理を  $V_{TH}$ -hopping で行ったときの消費電力の時間推移をシミュレーションで求め,その結果を図 3.10,図 3.11 に示す.図 3.10 は MPEG4 エンコードにおける消費電力の時間推移を,図 3.11 には動作周波数の時間推移を示す.3 種類以上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化,回路オーバーヘッドを考えると 2 種類の動作周波数,しきい値を用いるのが効果的であるといえる.動作周波数に  $f_{CLK}$ ,  $f_{CLK}$ 

また図 3.10,図 3.11 から明らかなように,一般的な動画の場合,約 94%の時間を低電力状態,つまり低周波数で動作させてもリアルタイム処理が可能である.

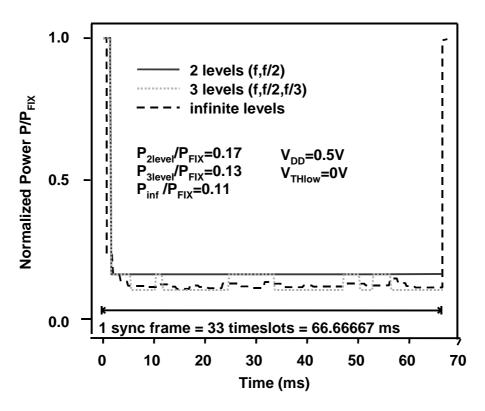


図 3.10 V<sub>TH</sub>-hopping を用いた MPEG4 エンコードにおける消費電力の時間推移

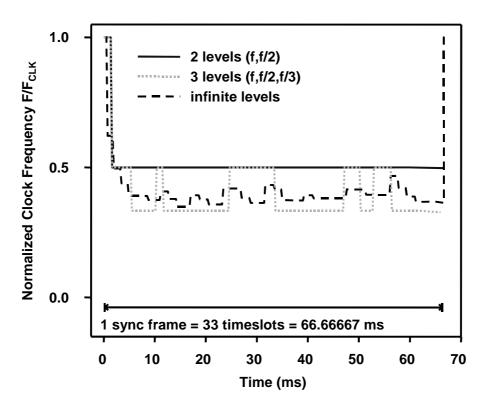


図 3.11 V<sub>TH</sub>-hopping を用いた MPEG4 エンコードにおける動作周波数の時間推移

図 3.12 に上述 2.0 手法 ,つまり  $V_{TH}$ -hopping に dual  $V_{TH}$ を付加した手法の概要を示す.  $V_{TH}$ -hopping においては各回路パスをクリティカルパスとノンクリティカルパスに分類せずにプロセッサ全体のしきい値をホッピングさせたが,本手法ではクリティカルパスとノンクリティカルパスに分類し,ノンクリティカルパスには高いしきい値を適用しホッピングしない. クリティカルパスには  $V_{TH}$ -hopping を適用し,プロセッサの負荷に応じてしきい値を制御する.制御方法は前述した  $V_{TH}$ -hopping のときと同様である.

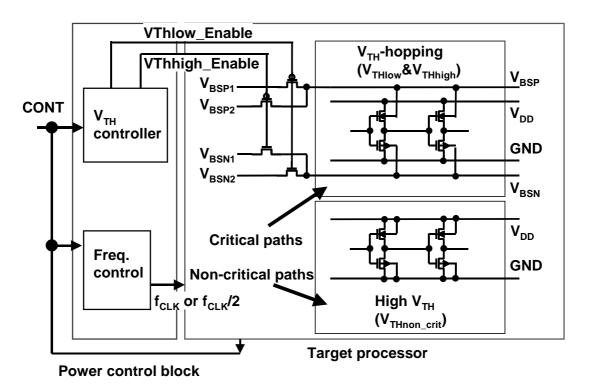


図 3.11 V<sub>TH</sub>-hopping に dual V<sub>TH</sub>を負荷した場合の概要

図 3.12 は MPEG4 エンコードを従来の single  $V_{TH}$ , dual  $V_{TH}$ , そして提案する  $V_{TH}$ -hopping で処理したときの消費電力の比較である. $V_{TH}$ -hopping を用いたときの消費電力は single  $V_{TH}$  の 18.5%に,また dual  $V_{TH}$  と比べても 27%に削減される. $V_{TH}$ -hopping に dual  $V_{TH}$  を付加した手法(上述 2.の  $V_{TH}$ -hopping + dual  $V_{TH}$ )の場合, $V_{TH}$ -hopping のみの手法(上述 1.の手法)に比べて消費電力は 1.5%改善されるが,ウェルの分離よる面積オーバーヘッド等を考慮すると, $V_{TH}$ -hopping に dual  $V_{TH}$  を導入する効果は小さく,手法 1.のようにクリティカルパスかどうかにかかわりなく Target processor 全体をホッピングする方法がもっとも効果的であるとわかる.

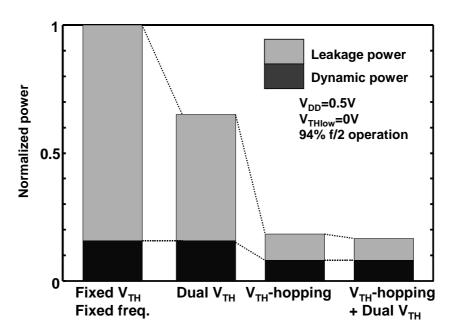


図 3.12 固定しきい値, Dual V<sub>TH</sub>, V<sub>TH</sub>-hopping, V<sub>TH</sub>-hopping + Dual V<sub>TH</sub> における 消費電力の比較

ここまでは基板のしきい値が低い( $V_{TH}=0V$ )ときも場合についてその有効性を述べてきた.この他に基板のしきい値が比較的高いときでも Forward-biasing で高速化を実現する [24]ときにこの方法が有効となる.基板に正のバイアスをかけるとしきい値が低くなることから動作速度は上昇する.このときの周波数を  $f_{CLK}$ 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)ときの周波数を  $f_{CLK}/2$  とすればより高速な回路動作を行うことができる.しかしこのとき,しきい値が低くなることによるリークの増加や,ドレイン-基板間に生じる順バイアスによって基板電流が生じるため消費電力が増大する.そこで  $V_{TH}$ -hoppingを用いることでそのスタティック電力を削減することが可能となる.

## 3.3.2.V<sub>TH</sub>-hopping の実測

基板バイアスを正にかけたときの  $V_{TH}$ -hopping について実測を行った.まず基板バイアスを正にかけていったときに遅延や消費電力がどう変化するのかを実際に測定するために基板バイアス制御が可能なインバータチェーンを設計した.設計したインバータチェーンの回路図を図 3.13 に示す.図をみてわかるとおり  $V_{FW}$ よって基板バイアスを制御できるようになっている.インバータチェーンの遅延,消費電力と基板バイアスの関係を図 3.14 に示す.

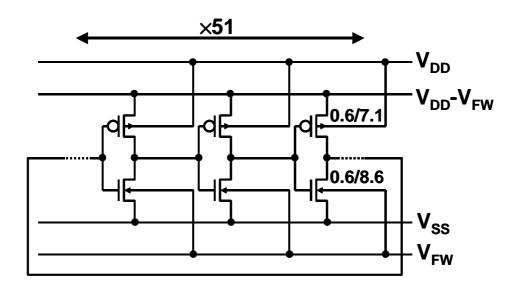


図 3.13 基板バイアス制御可能なインバータチェーン回路図

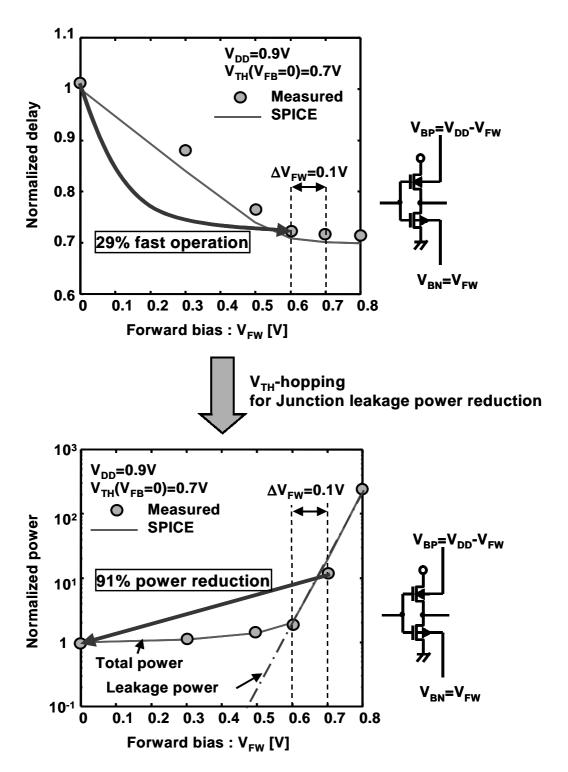


図 3.14 インバータチェーンの遅延,消費電力と基板バイアスの関係

基板バイアスを正にふっていくと、それにともなってバンドギャップ障壁が下がり、しきい値が低くなる. しきい値が低くなることで回路遅延も小さくなる.

図 3.14 を見てわかる通り, $V_{TH}$ -hopping において, $V_{FW}$ を 0.6V かけることで 29%高速化を実現している.消費電力については  $V_{FW}$  が 0.6V のときにはバイアスをかけていないときの数倍程度であるが,0.7V になると数十倍になってしまう.バイアスがゆらぐことを考慮すると 0.1V 程度のゆらぎは考えられるので,D-ストケースでは消費電力はバイアスをかけていないときの十倍以上になってしまう.このときに  $V_{TH}$ -hopping を用いることで,消費電力を削減する.高速動作が必要ないときは  $V_{FW}$ を 0V ,つまり基板バイアスをかけないことによって 91%消費電力を削減することができる.

 $V_{TH}$ -hopping によって消費電力および遅延を制御することができる 8bit RISC プロセッサを  $0.6\mu m$  プロセスにより試作した.チップ\*)写真を図 3.15 に示す.チップサイズは  $2.1mm \times 2.0mm$ ,  $V_{BS}$  selector のサイズは  $0.2mm \times 0.6mm$ , ゲート数は約 3000NAND ゲートである.図 3.16 に  $V_{FW}$  を 0.7V に固定したときのインバータチェーンの回路遅延,RISC プロセッサの回路遅延と電源電圧の関係を示す.

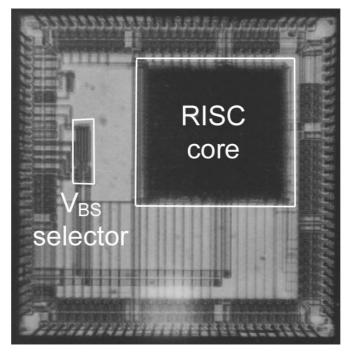


図 3.15 RISC プロセッサチップ写真

 $<sup>^{*)}</sup>$ 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)の協力で行われたものである.

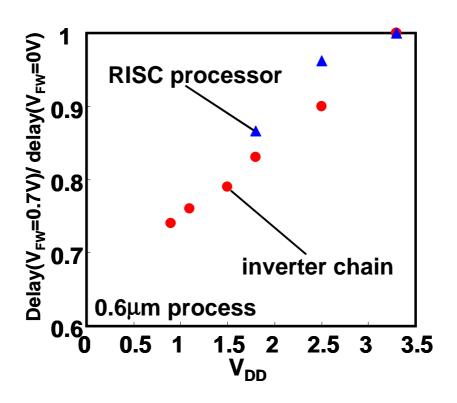


図 3.16 インバータチェーンの回路遅延, RISC プロセッサの回路遅延と 電源電圧の関係

回路遅延は  $V_{DD}$ - $V_{TH}$  に比例するため,電源電圧が低くなるにつれて  $V_{TH}$ -hopping の効果も大きくなる.図 3.16 の実測からもインバータチェーンと RISC プロセッサともに電源電圧が低くなるにつれて  $V_{TH}$ -hopping の効果が大きくなることがわかる.電源電圧 1.8V においておよそ 15% の高速化が RISC プロセッサにおいても可能である.

## 3.4. dual-V<sub>TH</sub>の簡易設計手法と設計指針

ハイパフォーマンスな VLSI では,遅延を抑えるためにしきい値は今後も低くなってくると思われる.VLSI の消費電力は前述のように電荷の充放電によるダイナミック消費電力とリーク電流によるリーク消費電力に分けることができる.現在のプロセスにおいて,リーク電力はダイナミック消費電力に比べて10%程度と小さく,大きな問題とならないが,今後,より高いパフォーマンスを実現するために,しきい値は現在よりも低くなっていくと考えることができる.しきい値が低くなるとそれに伴ってリーク電流が増加するが,その増加は図

からわかるように指数関数的に増加する.そうなると現在問題とならないリーク電流が大きな問題となってくる.リーク電流を削減する手法として,MTCMOS[8]や VTCMOS[10]が提案されているが,これらの手法はスタンバイ時のリーク電流はカットできるが,アクティブ時のリーク電流はカットすることができない. $dual-V_{TH}$  は回路をクリティカルパスとノンクリティカルパスに分け,クリティカルパスには低いしきい値のデバイスを用い,ノンクリティカルパスには高いしきい値のデバイスを用いることで,高いパフォーマンスを維持しながら,リーク電流による消費電力を削減する手法である.本節では  $dual-V_{TH}$  の簡易設計手法を述べた後, $dual-V_{TH}$  の設計指針として,しきい値の最適化やしきい値の組み合わせについて述べる.

### 3.4.1.dual-V<sub>TH</sub>の簡易設計手法

dual-V<sub>TH</sub>の設計を行う際に HDL を用いるとすると,特別なツールやライブラリが必要となる.本節では HDL を用いた上位設計において広く使われている Synopsys Design Compiler (SDC)[18]を用いた簡易な設計手法について述べる.

図 3.17 に本簡易設計手法のフローチャートを示す .本設計手法においては ,まず dual- $V_{TH}$  用のライブラリを作成し , その後は従来通りに SDC を用いて回路を合成することができる .

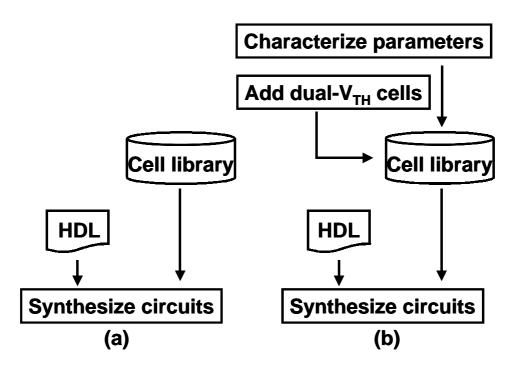


図 3.17 設計のフローチャート (a)SDC による通常の CMOS 回路設計手法 (b)SDC による dual-V<sub>TH</sub> 設計手法

dual-V<sub>TH</sub>に用いるライブラリには少なくとも 2 種類のセルが必要になる.たとえばインバータセルを例にとると高いしきい値のトランジスタを用いた INV(high)と.低いしきい値を用いた INV(low)の 2 種類のセルが必要となる.低いしきい値を用いたセルは遅延が小さく論理段数の多いバスにも用いることができるが,消費電力は大きくなってしまう.高いしきい値を用いたセルは,遅延は大きくなってしまうものの,消費電力は低く抑えることができる.つまり dual-V<sub>TH</sub>では,タイミング制約が許す限り,高いしきい値のセルを用いた方が低消費電力である.SDC において消費電力は dynamic power と leakage power がパラメータとして用いられ power 最適化を計ることができる.

SDC による論理合成にはセルの遅延情報が必要となる . SDC において扱われる遅延パラメータを図.3.18 に示す .

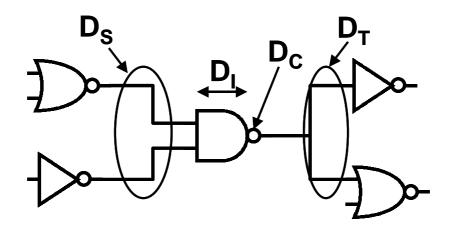


図 3.18 Synopsys Design Compiler における遅延パラメータ

セルの遅延, $D_{TOTAL}$ ,は式(3.1)のようになっている.

$$D_{TOTAL} = D_S + D_I + D_T + D_C (3.1)$$

それぞれのパレメータの意味は

D<sub>s</sub>: スロープ遅延,入力ピンでの状態遷移が遅いために,入力ピンから出力ピンに発生する 遅延.

D<sub>1</sub>: 固有遅延.入力ピンから出力ピンへのセル内部の遅延.

D<sub>T</sub>: 遷移時間.ネット上で状態遷移が完了するまでの時間.ネットのファンアウトにおいてセルの遅延を計算するパラメータとしても使用される.

D<sub>c</sub>:接続遅延.フライト遅延とも呼ばれる.相互接続ネットワークを介して状態遷移を伝播させるための時間,つまりドライブする出力ピンの信号遷移のあと,入力ピンで電圧が変化するまでの所要時間.

#### である.

セルの遅延情報に関しては各セルの SPICE ファイルを作成し, SPICE シミュレーションによって遅延パラメータを抽出した.

遅延と消費電力のパラメータを書き換えたセルを加えた.lib フォーマットのファイルを library compiler [25]でコンパイルし、.db フォーマットのファイルにすることで dual- $V_{TH}$ ライブラリを得ることができる.それ以降の手順は従来の上位設計と同じ手順で行うことができる.

### 3.4.2.セルにおけるしきい値の組み合わせ

dual-V<sub>TH</sub>では少なくとも 2 種類のしきい値をもつ MOSFET を用いる .例えば NOR セルを 例にとると図 3.19 のようにいくつかの組み合わせが考えられる .V<sub>TH LOW</sub> と V<sub>TH HIGH</sub> の 2 種類 のしきい値がそれぞれのトランジスタに適用され得るので,考えられ得る組み合わせは 2<sup>4</sup>, つまり 16 通りとなる. もし NOR だけでなくそれぞれのセルに 16 通りの組み合わせが存在 したとすると dual- $V_{TH}$  に用いるセルライブラリのセル数は通常の CMOS に比べ、16 倍とな ってしまい,効率的に論理合成するに現実的なサイズではない,16通りの組み合わせの中に はセルライブラリに加えても意味をなさないものも存在するので、それらの組み合わせをま ず取り除くことを考える.図中セル(c)はセル(a)に比べ大きな遅延をもち,さらに同等の電力 を消費するため,取り除くことができる.このように V<sub>TH LOW</sub> と V<sub>TH HIGH</sub> が直列に接続されて いるトランジスタの消費電力は  $V_{THLOW}$  にほぼ等しく,さらに遅延は大きいためライブラリ に追加する必要はない.またセル(e)とセル(f)は入力ピンに対して対象なため,どちらか一方 を加えればよい.このように不要なセルを取り除くと,ライブラリに加えることで高いパフ オーマンスを維持したまま,より低消費電力化を計ることができるのはセル(a),(b),(d),(e), (g), (h)の 6 通りのセルのみとなる. しかし 6 通りのセルがそれぞれのセルに存在すればセル ライブラリのサイズは6倍となり,セルライブラリのサイズとしてはまだ大きいと考えられ る.つまり,セル数が増えることによってより消費電力化を図ることができる一方で SDC による論理合成時間の増加[26]やレイアウトなどのエラーの導入に要する時間の増加という 問題が起きる.

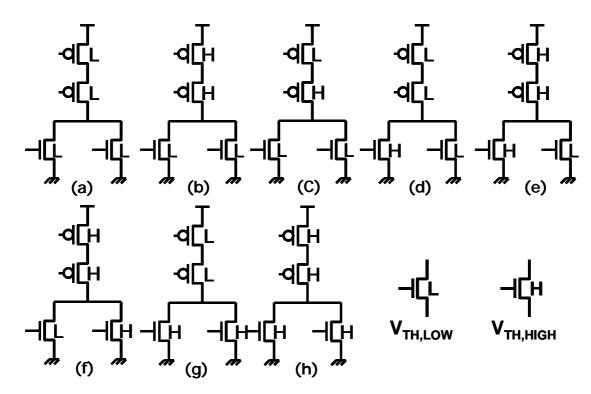


図 3.19 NAND セルにおけるしきい値の組み合わせ

もし,ライブラリセルがしきい値の組み合わせをすべて用意しなくても,すべてのトランジスタにしきい値の高いものを用いたものと,すべて低いしきい値の低いトランジスタを用いたセルの2種類でよいならば,ライブラリセルの数は従来のライブラリの2倍のサイズで済むことになり現実的である.

図 3.20 にしきい値を組み合わせたセルを用いたセルライブラリと,すべてのトランジスタに高いしきい値を用いたセルとすべてに低いしきい値を用いたセルの 2 種類のセルのみをふくむセルライブラリの比較を示す。比較には 8 ビット RISC プロセッサを SDC で合成した.ライブラリのテクノロジーは電源電圧 0.5V, $V_{TH,LOW}$ が 0V, $V_{TH,HIGH}$  には 0.1V,0.2V,0.3V,0.4V の 4 種類を用意し,それぞれのライブラリで合成して,消費電力を比較した.

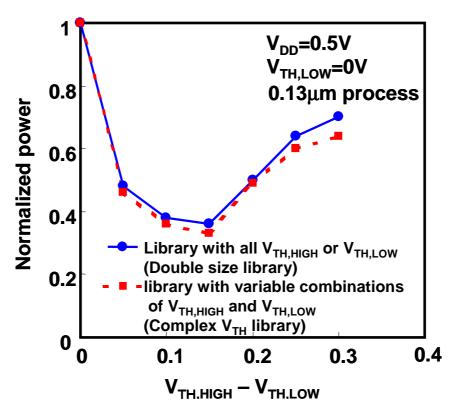


図 3.20 しきい値の組み合わせによる低消費電力化効率の比較

しきい値を組み合わせたセルライブラリを用いた方が低消費電力ではあるが,その効果は2種類のセルを用意したライブラリと比べて,4~7%しか改善しておらず,セル数が増加によって合成時間が増加することなどを考慮すると,2種類のセルを用意すれば十分であることがわかる.

### 3.4.3.dual-V<sub>TH</sub> におけるしきい値の最適化

 $V_{TH,LOW}$ がシステムパフォーマンスの要求から、つまりタイミング制約から決定されても、 $V_{TH,HIGH}$ は唯一に定まるものではない。本節では dual- $V_{TH}$ におけるしきい値の最適化について数式から導いた論理値と、SDC を用いたシミュレーション結果の 2 つについて述べる。本節の数式で用いられる変数および定数は表 3.1 に示す。

Notation	Meaning
a	switching activity
$L_{d}$	logic depth of critical path
f	given clock frequency
С	load capacitance par gate
α	velocity saturation index
$I_0$	drain current when V <sub>GS</sub> =V <sub>TH</sub> at lowest temperature
$T_{min}$	lowest operation temperature
T <sub>max</sub>	highest operation temperature
T	temperature
m	temperature exponent of mobility
n	subthreshold slope factor
K	coefficient of delay
$\Delta V_{TH}$	peak-to-peak V <sub>TH</sub> variation through process
κ	temperature coefficient of $V_{TH}$
k	boltzman constant
q	electron charge
$V_{TH,max}$	highest V <sub>TH</sub> in operation temp. and process variation range
$V_{TH,min}$	lowest V <sub>TH</sub> in operation temp. and process variation range
$I_{ON}$	drain current when $V_{GS}=V_{DD}$ at lowest temp.
	and highest V <sub>TH</sub> corner in process variation
$I_{OFF}$	leakage current at lowest temp.
	and highest V <sub>TH</sub> corner in process variation
$I_{LEAK}$	leakage current at highest temp.
	and lowest V <sub>TH</sub> corner in process variation
p	the ratio of the gates which is in critical paths

表 3.1 Notations used in this paper

まず dual- $V_{TH}$  ではない ,通常の VLSI における遅延と消費電力の論理値を計算する .ショートチャネル MOSFET のレイン電流[27]は

$$I_{D} = \begin{cases} I_{0} \left(\frac{e}{\alpha}\right)^{\alpha} \left(\frac{q}{nkT}\right)^{\alpha} \left(\frac{T}{T_{\min}}\right)^{\alpha-m} (V_{GS} - V_{TH})^{\alpha} & (V_{GS} \ge V_{TH} + \frac{\alpha nkT}{q}) \\ I_{0} \left(\frac{T}{T_{\min}}\right)^{\alpha-m} \cdot \exp\left(\frac{q(V_{GS} - V_{TH})}{nkT}\right) & (V_{GS} \le V_{TH} + \frac{\alpha nkT}{q}) \end{cases}$$
(3.2)

で表される.

デバイスのばらつきと温度のばらつきを考慮したしきい値は式(3.2)で表され, $T_{min}$ は最低動作温度, $V_{Thmin}$ と  $V_{Thmax}$ はプロセスのばらつきによるしきい値の最小値と最大値である. つまり  $V_{TH}$ がプロセスによってばらつくしきい値の幅である.

$$V_{TH \min} = V_{TH \max} - \Delta V_{TH} - \kappa \Delta T \tag{3.3}$$

最低動作温度時に回路遅延は最悪値をとる .回路遅延の最悪値を式(3.4)に示す . $I_{ON}$  は  $V_{GS}$  =  $V_{DS}$  =  $V_{DD}$  のときのドレイン電流であり ,  $I_{OFF}$  は  $V_{GS}$  = 0 ,  $V_{DS}$  =  $V_{DD}$  のときのドレイン電流である .

$$t_d = K \frac{CV_{DD}}{I_{ON} - I_{OFF}} \tag{3.4}$$

式(3.4)はしきい値は $V_{Thmax}$ のときに最悪値をとる.回路遅延は式(3.2)を式(3.4)に代入し,

$$t_{d} = \begin{cases} K \frac{CV_{DD}}{I_{0} \left(\frac{e}{\alpha}\right)^{\alpha} \left(\frac{q}{nkT_{\min}}\right)^{\alpha} \left(V_{DD} - V_{TH \max}\right)^{\alpha} - I_{0} \cdot \exp\left(-\frac{qV_{TH \max}}{nkT_{\min}}\right)} & (V_{TH \max} > -\frac{\alpha nkT_{\min}}{q}) \\ K \frac{CV_{DD}}{I_{0} \left(\frac{e}{\alpha}\right)^{\alpha} \left(\frac{q}{nkT_{\min}}\right)^{\alpha} \left\{\left(V_{DD} - V_{TH \max}\right)^{\alpha} - \left(-V_{TH \max}\right)^{\alpha}\right\}} & (V_{TH \max} < -\frac{\alpha nkT_{\min}}{q}) \end{cases}$$
(3.5)

となる.式(3.5)とクリティカルパスの論理段数,Ldを用いて動作周波数fを表すと

$$f = \frac{1}{L_d \cdot t_d} \tag{3.6}$$

となる.

一方,消費電力は最も高い動作周波数で最悪値をとる.VLSI の消費電力はダイナミック消費電力  $P_D$ ,とリーク消費電力  $P_{LEAK}$  の和で表され,VLSI の消費電力 P は式(3.7)のようになる.

$$P = P_D + P_{LEAK} = afCV_{DD}^2 + I_{LEAK \max}V_{DD}$$
(3.7)

 $I_{LEAK}$ はリーク電流であり式(3.8)に示し、しきい値がばらつくことを考慮すると  $V_{THmin}$  おいて最悪値をとる.

$$I_{LEAK} = \begin{cases} I_0 \left(\frac{T_{\text{max}}}{T_{\text{min}}}\right)^{\alpha - m} \cdot \exp\left(-\frac{qV_{TH \text{ min}}}{nkT_{\text{max}}}\right) & (V_{TH \text{ min}} > -\frac{\alpha nkT_{\text{max}}}{q}) \\ I_0 \left(\frac{e}{\alpha}\right)^{\alpha} \left(\frac{q}{nkT_{\text{max}}}\right)^{\alpha} \left(\frac{T_{\text{max}}}{T_{\text{min}}}\right)^{\alpha - m} \left(-V_{TH \text{ min}}\right)^{\alpha} & (V_{TH \text{ min}} < -\frac{\alpha nkT_{\text{max}}}{q}) \end{cases}$$
(3.8)

式(3.8)と式(3.6)を式(3.7)に代入することで消費電力 P が求められる.

 ${
m dual}$ - ${
m V}_{
m TH}$ においてはクリティカルパスには  ${
m V}_{
m TH, LOW}$  適用され,ノンクリティカルパスには  ${
m V}_{
m TH, HIGH}$  が適用される . ${
m dual}$ - ${
m V}_{
m TH}$ において消費される電力を求めるためにはクリティカルパス が全回路の占める割合 p を考慮する必要がある.クリティカルパスが全回路の占める割合 p を用いて消費電力  ${
m P}$  を表すと

$$P = P_D + (pI_{LEAK.LOW} + (1-p)I_{LEAK.HIGH})V_{DD}$$
(3.9)

#### となる.

プロセッサ内のクリティカルパスの分布ばらつきや温度ばらつきを考慮し,式(3.9)を用いて dual- $V_{TH}$  の消費電力を求めた結果を図 3.21 ,図 3.22 に示す.それぞれの図には異なるテクノロジーを仮定している.図 3.21 では  $V_{TH,LOW}$  が 0.1V ,電源電圧が 1.5V であり ,図 3.22 では  $V_{TH,LOW}$  が-0.1V ,電源電圧が 0.5V である.

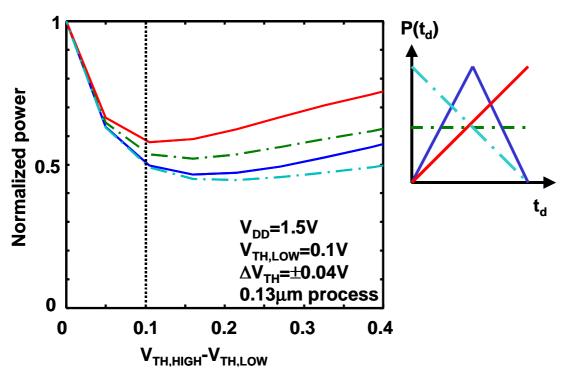


図 3.21 論理計算によるしきい値の最適化 (V<sub>DD</sub> = 1.5V, V<sub>TH,LOW</sub> = 0.1V)

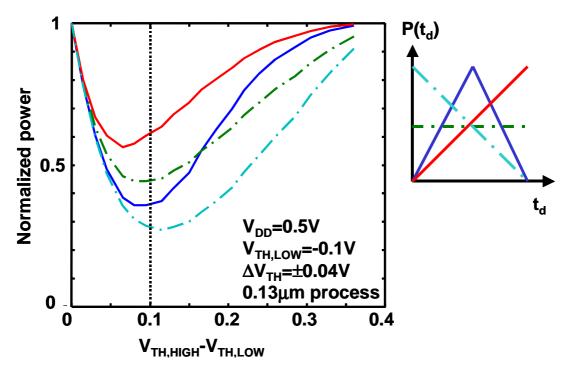


図 3.22 論理計算によるしきい値の最適化 (V<sub>DD</sub> = 0.5V, V<sub>TH,LOW</sub> = -0.1V)

図 3.21 , 図 3.22 をみると , 異なるテクノロジーを用い , 異なる電源電圧を用いている . さらにプロセッサ内のパスの遅延分布が異なっているにもかかわらず同じ傾向を示しており , しきい値の選択には最適値が存在することを示している . つまり式(3.10)はパスの遅延分布 , 電源電圧によらない式である .

$$V_{TH,HIGH} - V_{TH,LOW} = 0.1 (3.10)$$

式(3.10)を満たすようにしきい値を選択すれば,回路のパフォーマンスを損なわずに消費電力を最適化することができる.

つまり  $V_{TH,HIGH}$  と  $V_{TH,LOW}$  の差があまり大きいと  $V_{TH,HIGH}$  がリークを完全にカットするものの,遅延が大きいためすべてのセル中で  $V_{TH,HIGH}$  のセルが使われる数が少なくなり結局  $dual-V_{TH}$  の効果は薄くなる.一方, $V_{TH,HIGH}$  と  $V_{TH,LOW}$  の差が小さいと  $V_{TH,HIGH}$  のセルがカットするリーク電流が小さく,結果として  $dual-V_{TH}$  の効果は薄くなってしまう.そこで最適値が存在するわけである.

SDC によるシミュレーションの結果も図 3.21,図 3.22 と同じ傾向を示している . SDC によるシミュレーションでは 8 ビット RISC プロセッサを前述の手法によって dual- $V_{TH}$  を用いて論理合成し,消費電力を求める . その結果を図 3.23 に示す . 論理合成の条件は電源電圧が 0.5 V ,  $V_{TH,LOW}$  は-0.1 V , 0 V , 0.1 V の 3 種類である .

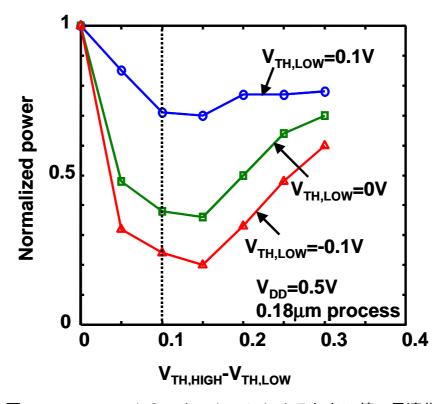


図 3.23 Synopsys シミュレーションによるしきい値の最適化

図 3.23 をみてわかる通り, $V_{TH,LOW}$  が-0.1V のときには 80%電力を削減することができ, $V_{TH,LOW}$  が 0.1V でも dual- $V_{TH}$  を用いることで 30%電力を削減することができる.図 3.21,図 3.22 に加えて図 3.23 では異なるしきい値についても式(3.10)をみたしている.Synopsys によるシミュレーションでも,式(3.10)を満たす点でほぼ消費電力は最適値をとり,論理計算とよく合っていることがわかる.

### 3.5. まとめ

本章ではしきい値による低消費電力化を述べた.

しきい値を基板バイアスによってダイナミックに制御する  $V_{TH}$ -hopping という手法を提案した.

 $V_{TH}$ -hopping ではしきい値をリアルタイムで制御する必要があるので,基板バイアスを制御する  $V_{PSUB}$ ,  $V_{NSUB}$  が新たに必要となる.そこで基板バイアス制御を可能とする簡易な設計手法を提案した.ライブラリに手を加えず,配置配線後のレイアウトから基板コンタクトをSKILL スクリプトで除去し, $V_{PSUB}$ ,  $V_{NSUB}$  を加えることで簡易な設計を実現した.

 $V_{TH}$ -hopping の有効性を MPEG4 のエンコードを用いて示した.シミュレーションによると  $V_{TH}$ -hopping によっておよそ 80%電力を削減することができることがわかった.

 $0.6\mu \mathrm{m}$  プロセスによって実際に基板バイアス制御が可能なインバータチェーンを試作した. $V_{\mathrm{TH}}$ -hopping によって,電源電圧  $0.9\mathrm{V}$ ,基板にフォワードバイアスを  $0.6\mathrm{V}$  かけることで要求されるパフォーマンスがバイアスをかけていないときのパフォーマンスにくらべて 30%高くても対応できること.バイアス電圧がゆらぐことを考慮するとフォワードバイアスを  $0.7\mathrm{V}$  かけたときまで考慮しなくてはならず,そのときの消費電力はバイアスをかけていないときの十倍以上になってしまう.このとき  $V_{\mathrm{TH}}$ -hopping をもちい,高いパフォーマンスが要求されないときは基板バイアスをかけないことで,約 90%電力を削減できることを,インバータチェーンを実測することによって示した.

さらに  $V_{TH}$ -hopping が可能な 8bit RISC プロセッサを試作した. $0.6\mu m$  プロセス,チップ 面積は  $2.0mm \times 2.1mm$ .測定結果はインバータチェーンと同様の傾向を示し, $V_{TH}$ -hopping が インバータチェーンのような小規模な回路だけでなく,RISC のような大規模な回路にも適用可能であること,さらに本研究で提案した設計手法が正しいことがわかった.

Synopsys Design Compiler を用いた簡易な dual- $V_{TH}$  設計手法を提案した.本設計手法はセルライブラリのセルに高いしきい値のセルと低いしきい値のセルを用意し,遅延は SPICE シミュレーションによりパラメータを抽出する.消費電力に関しても遅延と同様に SPICE からパラメータを抽出する.パラメータ抽出後は従来と同じ設計手法により回路を合成することができる. dual- $V_{TH}$  では,高いしきい値と低いしきい値のトランジスタを用いるので,セル内のそれぞれのトランジスタにどちらのしきい値を適用するかで,さまざまな組み合わせが考えられるが,すべて高いしきい値のトランジスタを用いたセルとすべて低いしきい値のトランジスタを用いたセルを用意すれば,さまざまな組み合わせを考えたセルライブラリに比

べて 4~8%消費電力が劣化するだけであり,効率的である.高いしきい値を低いしきい値の組み合わせには最適値が存在し、二つのしきい値の差が 0.1V になるように選択するのがよい. さらにその最適値は電源電圧,パスの遅延のばらつき,しきい値によらないことがわかった.

結論 第4章

# 第4章.結論

### 4.1.結論

本研究では大きく分けて,電源電圧制御による低消費電力化としきい値制御による低消費電力化の2つについて述べた.

電源電圧制御による低消費電力化においては次のことがわかった.

- VH を用いる際の SRAM の問題点を考察した .VH を SRAM に用いる場合は NMOS によってプリチャージするタイプのものはセンスアンプでビット線の微小な電位差をセンスする際に誤動作を起こす可能性がある . したがって PMOS によってプリチャージするタイプが適していることがわかった .
- VH を用いる際に最適な乗算回路について検討した.乗算器の消費電力の7割から8割は乗算器を構成している加算回路で消費されるので,VH を用いる際に最適な加算回路

結論 第4章

形式について ,シミュレーションおよび試作チップの測定により検討した . $V_{DD\_HIGH}/V_{TH}$ が 3 以上の領域では CPL が最適な回路形式であり ,3 以下の領域では CMOS が最適な回路形式である . またテクノロジーが進歩すると  $V_{DD}/V_{TH}$  は小さくなる傾向があるので , 今後も CMOS が広く用いられると思われる .

しきい値制御による低消費電力化においては次のことがわかった.

- V<sub>TH</sub>-hopping ではしきい値をリアルタイムで制御する必要があるので,基板バイアスを制御する V<sub>PSUB</sub>,V<sub>NSUB</sub> が新たに必要となる.そこで基板バイアス制御を可能とする簡易な設計手法を提案した.ライブラリに手を加えず,配置配線後のレイアウトから基板コンタクトを SKILL スクリプトで除去し, V<sub>PSUB</sub>,V<sub>NSUB</sub> を加えることで簡易な設計を実現した.
- $V_{TH}$ -hopping の有効性を MPEG4 のエンコードを用いて示した . シミュレーションによると  $V_{TH}$ -hopping によっておよそ 80%電力を削減することができることがわかった .
- 0.6μm プロセスによって実際に基板バイアス制御が可能なインバータチェーンを試作した.
- V<sub>TH</sub>-hopping によって,電源電圧 0.9V,基板にフォワードバイアスを 0.6V かけることで約 30%高速化できること,フォワードバイアスをかけた状態からかけない状態にすることで約 90%電力を削減できることをインバータチェーンの実測により示した.
- V<sub>TH</sub>-hopping が可能な 8bit RISC プロセッサを試作した . 0.6μm プロセス , チップ面積は 2.0mm×2.1mm . 測定結果はインバータチェーンと同様の傾向を示し , 本研究で提案した 設計手法が正しいこと , さらにインバータチェーンのような小規模な回路ばかりではなく , RISC プロセッサのような大規模な回路にも V<sub>TH</sub>-hopping が適用可能であることがわかった .
- Dual-V<sub>TH</sub>では,高いしきい値と低いしきい値のトランジスタを用いるので,セル内のそれぞれのトランジスタにどちらのしきい値を適用するかで,さまざまな組み合わせが考えられるが,すべて高いしきい値のトランジスタを用いたセルとすべて低いしきい値のトランジスタを用いたセルを用意すれば,さまざまな組み合わせを考えたセルライブラリに比べて4~8%消費電力が劣化するだけであり,効率的であることがわかった.
- 高いしきい値を低いしきい値の組み合わせには最適値が存在し,二つのしきい値の差が

結論 第4章

0.1V になるように選択するのがよいということがわかった.

### 4.2. 今後の課題

•  $V_{TH}$ -hopping はリーク電力が大きいところで効力を発揮する.本研究では  $V_{TH}$ -hopping の効果を実測するのに,しきい値が0.7V と高い領域のみにとどまった.しきい値が低い領域での実測による  $V_{TH}$ -hopping の効果を検証することが課題として残されている.

● V<sub>TH</sub>-hopping はプロセッサの負荷をソフトウェアで監視し ,負荷に応じてしきい値をソフトウェアでコントロールする .ソフトウェアを実装し ,V<sub>TH</sub>-hopping の効果を検証することが課題として残されている .

# 参考文献

- [1] T.Sakurai (Ed.), "Low-Power High-Speed LSI Circuits and Technology", Realize Inc., 1998.
- [2] T. Sakurai and A. R. Newton, "Alpha Power Law MOSFET Model and its Applications to CMOS Inverter Delay and other formulas", *IEEE Journal of Solid-State Circuits*, vol.25, no.2, pp.584-594, April, 1990.
- [3] A.Chandrakasan, S. Sheng, R. Brodersen, "Low-Power CMOS Digital Design", *IEEE Journal of Solid-State Circuits*, 27(4), Apr.1992.
- [4] A.Chandrakasan, R.Brodersen, "LOW POWER DIGITAL CMOS DESIGN", Kluwer Academic Publishers, 1995.
- [5] A.Chandrakasan, V.Gutnik, and T. Xanthopoulos, "Data driven signal processing: an approach for energy efficient computing," *Proceedings of International Symposium on Low Power Electronics and Design (ISLPED*'96), pp.347-352, 1996.
- [6] J.Goodman, A.Chandrakasan, A.P.Dancy, "Design and implementation of a scalable encryption processor with embedded variable DC/DC converter," *Design Automation Conference*, 1999. Proceedings.36<sup>th</sup>, 1999, pp.855-860.
- [7] Seongsoo Lee and Takayasu Sakurai, "Run-time Power Control Scheme using Software Feedback Loop for Low Power Real-time Application," *Proceedings of ASP-DAC'00*, 2000.
- [8] S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S, Shigematsu, and J. Yamada," A 1-V Power Supply High-Speed Digital Circuits Technology with Multithreshold-Voltage CMOS", *IEEE Journal of Solid-State Circuits*, Vol.30, No.8, pp847-854, August, 1995.
- [9] James T. Kao, and Ananthe P. Chandrakasan, "Dual-Threshold Voltage Techniques for Low-Power Digital Circuits", *IEEE Journal of Solid-State Circuits*, Vol.35, No.7, pp.1009-1018, July, 2000.

- [10] T. Kuroda, T. Mita, T. Fujita, F. Yamane, F. Sano, M. Norishima, M. Murota, M. Kato, M. Kinugawa, M. Kakumu, and T. Sakurai, "A 0.9V 150MHz 10mW 4mm<sup>2</sup> 2-D Discrete Cosine Transform Core Processor with Variable-Threshold Voltage Scheme", *IEEE International Solid-State Conference, Digest of Technical Papers*, pp.166-167, 1996.
- [11] Issam S. Abu-Khater, Abdellatif Bellaouar and M.I.Elmasry,"Circuit Techiques for CMOS Low-Power High-Performance Multipliers" *IEEE Journal of solid-state circuits*, Vol31, No.10, pp.1535-1546, Oct 1996.
- [12] Madhuban Kishor, Basabi Bhaumik, Jose Pineda de Gyvez and Marcel Pilgrom, "Comparison of Logic families in Deep Submicron Technology for Low Power and Low Voltage Applications", *IEEE Journal of solid-state circuits*, Vol32, No.10, pp.1435-1442, Oct, 1997.
- [13] Reto Zimmermann and Wolf Fichtner, "Low-Power Logic Styles; CMOS Versus Pass-Transistor Logic", *IEEE Journal of solid-state circuits*, Vol32, No.7, pp.1079 1089, July 1997.
- [14] Akilesh Parameswar, Hiroyuki Hara, Takayasu Sakurai, "A Swing Restored Pass-Transistor Logic-Based Multiply and Accumulate Circuit for Multimedia Applications", *IEEE Journal of solid-state circuits*, Vol31, No.6, pp.804 809, June, 1996.
- [15] Reto Zimmermann and R. Gupta, "Low-Power Logic Styles; CMOS Versus Pass-Transistor Logic", *Proceedings of 22<sup>nd</sup> European Solid-State Conference*, pp.112-115, Sept, 1996.
- [16] D.J.Radak, L.W.Linholn, "On-Chip Propagation Delay Measurement", *IEEE Custom Integrated Circuits Conference*, pp.579-581, 1987.
- [17] International Technology Roadmap for Semiconductors, 1998 update.
- [18] Synopsys 1999.05 Reference Manual, Synopsys Inc., 1999.
- [19] Apollo user guide, Avant! Co., 1998.
- [20] Cadence Design Framework Reference Manual, Cadence Design Systems Inc., 1997.
- [21] Diva interactive verification reference manual, Cadence Design Systems Inc., 1997.
- [22] James T. Kao and Aanantha P. Chandrakasan, "Dual-Threshold Voltage Techniques for Low-power Digital Circuits" *IEEE Journal of solid-state circuits*, Vol.35, No.7, pp.1009-1018, 2000.
- [23] S. Lee and T. Sakurai, "Run-time voltage hopping for low-power real-time systems," *IEEE/ACM Proc. DAC*, pp.806-809, June 2000.
- [24] Clement Wann, Jay Harrington, Rebecca Mih, Serge Biesemans, Kevin Han, Robert Dennard, Odin Prigge, Chuan Lin, Reinhard Hahnkopf and Bomy Chen, "CMOS with Active Well Bias for

- Low-Power and RF/Analog Applications" Symposium on VLSI Technology Digest of Technical Papers, pp158-159, 2000.
- [25] Synopsys Library Compiler Reference Manual, Ver 1997.01, Synopsys Inc., 1997.
- [26] Nguyen Minh Duc and Takayasu Sakurai, "Compact yet High-Performance (CyHP) Library for Short Time-to-Market with New Technologies", Proceeding of Asia and South Pacific Design Automation Conference 2000, pp.475-480, 2000.
- [27] Kouchi Nose and Takayasu Sakurai, "Optimization of  $V_{DD}$  and  $V_{TH}$  for Low-Power and High Speed Applications", Proceeding of Asia and South Pacific Design Automation Conference 2000, pp.469-474, 2000.

# 本研究に関する発表

- [1] 平林 雅之,桜井 貴康,"低消費電力乗算器における加算回路形式の検討",2000年秋期第61回応用物理学会学術講演会.
- [2] 平林 雅之,桜井 貴康, "基板バイアス制御を可能にする自動設計手法",2000年電子情報通信学会ソサエティー大会.
- [3] Koichi Nose, Masayuki Hirabayashi, Hiroshi Kawaguchi, Seongsoo Lee and Takayasu Sakurai, "V<sub>TH</sub>-hopping Scheme for 82% Power Saving in Low-voltage Processors", 2001 Custom Integrated Circuits Conference. (accepted)
- [4] Masayuki Hirabayashi, Koichi Nose and Takayasu Sakurai, "Design Methodology and Optimization Strategy for Dual-V<sub>TH</sub> Scheme using Commercially Available Tools", International Symposium on Low Power Electronics and Design 2001. (under investigation)

# 謝辞

本修士論文研究を行うにあたり、お忙しい中最先端の技術者としてさらに指導教官として 広い視野と鋭い発想で様々なご指導ご鞭撻を賜った生産技術研究所第3部 桜井貴康教授に 深く感謝いたします.研究を進めていく上で学術的な見地からばかりではなく、産業的な見 地からのアイディアやご指摘、問題提起など私にとって大変有意義で貴重な経験となりまし た.

研究環境の整備や私の研究知識に関する稚拙さ故の質問や相談などに快くのってくださった川口博技官,稲垣賢一技官に深く感謝いたします.

また研究をはじめ、多方面で多々お世話となりました野瀬浩一氏をはじめとする研究室諸氏にも深く感謝いたします.

本研究におけるチップ試作は東京大学大規模集積システム設計教育センターを通しローム(株)および凸版印刷の協力で行われたものです。ここに感謝いたします。

最後に私事ではあるが 私生活の支えとなってくれた両親ならびに妻亮子に感謝いたします.

# 目次

第1章.	序論	1
1.1.	はじめに	1
1.2.	集積回路の現在	2
1.3.	CMOS 回路における消費電力と速度	3
1.4.	電源電圧制御による低消費電力化	5
1.4	.1. Run-time Voltage Hopping の基礎	5
1.4	.2. VH 実装上の問題点	7
1.5.	しきい値電圧制御による低消費電力化	8
1.5	.1. MTCMOS	9
1.5	.2. VTCMOS	10
1.5	.3. dual V <sub>TH</sub> テクノロジ	11
1.6.	本論文の構成	12
第2章.	電源電圧制御による低消費電力化	13
2.1.	低電源電圧下における SRAM の問題	13
2.1	.1. SRAM の現状	13
2.1	.2. 低電源電圧下における SRAM 動作の検討	14
2.2.	乗算器における加算回路形式の検討	17
2.3.	まとめ	24
第3章.	しきい値制御による低消費電力化	26
3.1.	しきい値制御の重要性	26
3.2.	基板バイアス制御を可能にする自動設計手法	27
3.3.	しきい値ホッピング(V <sub>TH</sub> -hopping)によるリーク電流削減技術	33
3.3	.1. V <sub>TH</sub> -hopping の概要	34
3.3	.2. V <sub>TH</sub> -hopping の実測	39
3.4.	dual-V <sub>TH</sub> の簡易設計手法と設計指針	43
3.4	.1. dual-V <sub>TI</sub> の簡易設計手法	44

3.4.	2. セルにおけるしきい値の組み合わせ	47
3.4.	3. dual-V <sub>TH</sub> におけるしきい値の最適化	49
3.5.	まとめ	54
第4章.	結論	5 <i>6</i>
4.1.	結論	5 <i>6</i>
4.2.	今後の課題	58
参考文南	伏	59
本研究に関する発表		62
謝辞		63