

博士学位論文

細粒度電源遮断制御および不揮発性素子を用いた

LSI の低消費電力化技術の研究

Low Power technology of LSI by Fine-Grain Power

Gating and Magnetic Tunnel Junction

芝浦工業大学大学院 理工学研究科

博士（後期）課程 機能制御システム専攻

工藤 優

# 目次

第1章	序論	1
第2章	研究背景	3
2.1.	消費電力	3
2.1.1	ダイナミック電力 ( $P_{dynamic}$ )	3
2.1.2	ショートサーキット電力 ( $P_{short-circuit}$ )	4
2.1.3	リーク電力 ( $P_{leakage}$ )	4
(a)	ゲートリーク電流 ( $I_{gate}$ )	5
(b)	サブスレッショルドリーク電流 ( $I_{sub}$ )	5
(c)	ジャンクションリーク電流 ( $I_{junc}$ )	5
(d)	GIDL(Gate Induced Drain Leakage)電流 ( $I_{GIDL}$ )	6
2.2.	トランジスタの微細化の傾向	7
2.3.	LSIの低消費電力化が求められる理由	8
2.4.	リーク電力の問題	9
2.5.	リーク電力削減技術	11
2.5.1	マルチ閾値電圧 (Dual- $V_{th}$ ) 技術	11
2.5.2	基板バイアス技術	12
2.5.3	パワーゲーティング (PG) 技術	14
(a)	細粒度 PG	15
(b)	不揮発性 PG	17
第3章	研究課題の定義	18
第4章	細粒度 PG におけるスリープ制御手法	19
4.1	Geyser-3	19
4.1.1	細粒度 PG の設計と実装	19
4.1.2	グラウンドバンス低減技術	25
4.2	スリープ制御手法	25
4.2.1	Whenever Idle Put to Sleep(WIPS)方式	25
4.2.2	Time-Based(TB)方式	26
4.2.3	Adaptive Time-Based(ATB)方式	27
4.2.4	従来手法の問題点	28
4.3	新しいスリープ制御手法の提案	29
4.3.1	リークモニタ回路	30
4.3.2	CuD 方式の制御方法	35
4.3.3	CuD 方式の利点	37
4.4	モデル式による評価	38

4.4.1	スリープ制御手法における消費エネルギー	38
4.4.2	各スリープ制御手法の比較	39
(a)	各スリープ制御手法での消費エネルギー	39
(b)	各パラメータの影響による比較	41
4.5	評価	42
4.6	考察とまとめ	61
第5章	不揮発性 PG 設計手法	63
5.1	リテンションフリップフロップ	63
5.1.1	SRAM 退避方式	64
5.1.2	バルーン方式	66
5.2	不揮発性メモリ	68
5.2.1	NAND フラッシュメモリ	68
5.2.2	抵抗変化型メモリ (ReRAM)	68
5.2.3	強誘電体メモリ (FeRAM)	68
5.2.4	STT-MRAM	69
(a)	アクティブ動作	74
(b)	ストア動作	74
(c)	スリープ動作	76
(d)	リストア動作	76
5.3	従来方式の分析	78
5.3.1	PSM-NVFF のレイアウト設計	78
5.3.2	PSM-NVFF の問題点	80
(b)	ストア動作のロバスト性	80
(c)	リストア動作のロバスト性	85
(d)	無駄な書き込みエネルギーの発生	86
5.4	新しい NVFF 回路の提案	88
5.4.1	Split-Store-Restore NVFF (SSR-NVFF)	88
(a)	ストア動作	88
(b)	リストア動作	89
5.4.2	Selective Storing NVFF (SS-NVFF)	93
(a)	初期化動作	93
(b)	ストア動作	93
(c)	リストア動作	94
5.5	評価	100
5.5.1	SSR-NVFF	100
(a)	ストア動作の不良率	100

(b)	ストア動作時のストア電流	101
(c)	回路性能の比較	103
5.5.2	SS-NVFF	104
5.6	考察とまとめ	107
第6章	考察	108
第7章	結論	110
第8章	今後の課題	112
	参考文献	113
	謝辞	118
	研究業績	

## 図目次

図 2.1.1	リーク電流の主成分.....	4
図 2.2.1	ムーアの法則と Intel 社製造マイクロプロセッサ[12].....	7
図 2.4.1	製造プロセスの微細化に伴う消費電力成分の推移[13].....	9
図 2.4.2	ダイナミック・リーク電力.....	10
図 2.5.1	Dual-V <sub>th</sub> .....	11
図 2.5.2	基板バイアス制御 (nMOS) .....	12
図 2.5.3	基板バイアス制御によるリーク電流および閾値電圧の変化.....	13
図 2.5.4	PG 技術.....	14
図 2.5.5	PG における時間的・空間的粒度.....	15
図 2.5.6	粗粒度 PG の実例 .....	16
図 2.5.7	PG における消費電力の推移.....	16
図 4.1.1	細粒度 PG におけるアーキテクチャ制御.....	20
図 4.1.2	PG 適用粒度による PG 制御タイミング .....	21
図 4.1.3	Geyser-3 のレイアウト図.....	22
図 4.1.4	Geyser-3 における各演算器のレイアウト図.....	24
図 4.2.1	TB 方式におけるスリープ制御.....	26
図 4.2.2	TB 方式の回路構成.....	27
図 4.2.3	ATB 方式におけるスリープ制御 .....	28
図 4.3.1	CuD 方式の回路構成.....	29
図 4.3.2	リークモニタの回路構成.....	31
図 4.3.3	電圧比較回路.....	32
図 4.3.4	リークモニタ回路のシミュレーション波形.....	32
図 4.3.5	リークモニタ回路のレイアウト図.....	34
図 4.3.6	チップ上の温度変化による PG 制御タイミング .....	36
図 4.3.7	演算器の BET とリークモニタ回路の遅延時間の関係性.....	36
図 4.5.1	Geyser-3 におけるリークエネルギー割合.....	42
図 4.5.2	各プログラムにおける演算器の Verilog シミュレーション結果 .....	43
図 4.5.3	各スリープ制御手法における演算器の消費エネルギー .....	44
図 4.5.4	各アプリケーションの消費エネルギーの算出方法 .....	44
図 4.5.5	各演算器の消費エネルギー.....	48
図 4.5.6	スリープ制御回路の消費エネルギー比較.....	49
図 4.5.7	アプリケーション毎の各演算器とスリープ制御回路の 消費エネルギーの割合 .....	52
図 4.5.8	回路環境による影響.....	56

図 4.5.9	グローバルばらつきによる各スリープ制御手法への影響 .....	58
図 4.5.10	グローバルばらつきによる演算器の BET とリークモニタ回路の 出力遷移時間への影響 .....	58
図 4.5.11	ランダムばらつきによるリークモニタ回路の 出力遷移時間への影響 ....	59
図 4.5.12	ランダムばらつきによる各スリープ制御手法の消費エネルギー .....	60
図 5.1.1	D-FF 回路図.....	64
図 5.1.2	SRAM 退避方式の概略図 .....	65
図 5.1.3	SRAM1bit のメモリセル.....	65
図 5.1.4	バルーン方式の D-FF 回路図 .....	67
図 5.2.1	MTJ 素子 .....	69
図 5.2.2	電流の向きによる MTJ の抵抗変化 .....	70
図 5.2.3	MTJ 素子の I-V 特性 .....	70
図 5.2.4	従来の PSM-NVFF の回路図 .....	72
図 5.2.5	PSM-NVFF の制御シーケンス.....	73
図 5.2.6	ストア動作 .....	75
図 5.2.7	リストア動作.....	77
図 5.3.1	PSM-NVFF レイアウト図.....	79
図 5.3.2	ストア電流の向きと電流の流れやすさ (HDPG) .....	81
図 5.3.3	ラッチ破壊の発生例.....	82
図 5.3.4	ストア時におけるストア電流経路.....	84
図 5.3.5	スレーブラッチサイズによるストア動作への影響 .....	85
図 5.3.6	無駄なストア動作の例.....	87
図 5.4.1	SSR-NVFF の回路図.....	90
図 5.4.2	SSR-NVFF の制御シーケンス .....	91
図 5.4.3	SSR-NVFF のストア動作.....	92
図 5.4.4	SSR-NVFF のストア電流の向きと流れやすさ .....	92
図 5.4.5	SS-NVFF 回路図 .....	96
図 5.4.6	SS-NVFF の制御シーケンス .....	97
図 5.4.7	MTJ へ書き込み動作 .....	98
図 5.4.8	選択回路へデータを格納 .....	99
図 5.5.1	プロセスばらつきにおけるストア動作の不良率 .....	101
図 5.5.2	PSM-NVFF と SSR-NVFF のストア電流比較 .....	102
図 4.5.3	16bit カウンタ回路 .....	105

## 表目次

表 3.1.1	演算器マクロの面積.....	22
表 3.5.1	各アプリケーションのスリープイベント解析.....	49
表 4.5.1	PSM-NVFF と SSR-NVFF の回路性能比較.....	103
表 4.5.2	SSNVFF の消費エネルギーおよび BET (PSM-NVFF で正規化) .....	106

# 第1章 序論

---

LSI (Large Scale Integrated Circuit : 大規模集積回路) は、IC (Integrated Circuit : 集積回路) のうち、その素子の集積度が 1000 個~10 万個程度のものである。デジタル/アナログ信号技術の要としてパソコン・OA 機器・ゲーム機器などの情報処理機器から、電子レンジ・洗濯機などの白物家電まで我々の生活に欠かせないものとなっている。また携帯電話をはじめとするモバイル情報端末の登場により、高性能化と高密度化が強く求められてきた。

これに伴い LSI のトランジスタの微細化技術は急激な躍進を遂げてきた。しかし一方で、LSI の発熱やバッテリーの長寿命化などの観点から、LSI の低消費電力化が大きな課題となってきた。特にトランジスタの微細化が進むことにより、回路が動作している時だけでなく待機時の消費電力の削減が低消費電力 LSI の大きな課題となっている。

リーク電力削減の技術としてマルチ閾値電圧 (Dual-Vth) [1,2]、逆バイアス基板印可 (RBB) [3,4]、パワーゲーティング (PG) [5,6,7]などが提案されている。Dual-Vth は回路の動作速度に影響のない経路のトランジスタに対して高閾値のトランジスタを用いることでリーク電力を削減する技術である。動作速度に影響のある経路 (クリティカルパス) のトランジスタには低閾値のトランジスタを用いるため、LSI の動作速度に影響が少ない特徴がある。しかし、トランジスタの実行的な閾値電圧を自由に変更できない場合、回路ごとの最適化設計が困難である。また、クリティカルパスのトランジスタのリーク電力は削減できない問題がある。さらに、製造時のマスク枚数の増大、プロセスステップの増加を伴うため製造コストが増大してしまう。RBB は基板バイアス効果により、基板に印可する電圧を変更することによって製造後のトランジスタの閾値を上げ、リーク電流を削減する技術である。Dual-Vth と比べ、回路が待機状態の時に回路全体のトランジスタの閾値を上げることができるため、より効果的なリーク削減が期待できる。しかし、基板の電圧を制御する必要があるため、電圧生成回路や制御回路などが必要になる。また、基板の電圧の充放電を繰り返す必要があるため、細かい粒度で制御する場合、エネルギーオーバーヘッドが問題になる恐れがある。一方、PG 技術は論理回路と電源 (または、グランド線) の間に高閾値のトランジスタを挿入することで、回路が待機時の時にリーク電流を遮断する技術である。回路が使用されていない時に効率よくリーク電力を削減できる技術として知られており、市販の CPU などにもすでに PG 技術が適用されている [8,9]。さらに、より細かい粒度ごとに区切り PG を適用することでリーク電力を削減することが期待できる。また、PG の対象となる回路内に、フリップフロップ (FF) などの記憶回路が存在する場合、PG 実行時に回路の内の記憶データが破壊されてしまう。しかし、不揮発性素子を用いること

で、記憶回路のデータを保持したまま PG が実行できる不揮発性 PG が実現できる。これにより、今まで PG が行えていなかった回路にも PG を適用することができるようになり、さらなる低消費電力 LSI の設計に繋がると期待されている。

以上を踏まえて本研究では LSI の消費電力のうちリーク電力に着目し、リーク削減に向けた PG 技術を研究テーマに上げ、より低消費電力を実現した LSI 設計手法の検討を行う。さらに、本研究で示した新しい回路を用いることで消費エネルギーの削減効果を求めるとともに、考察を行う。

本論文の構成は以下のとおりである。

第 2 章では本研究の研究背景について述べ、LSI の消費電力のうちリーク電力の削減が低消費電力化設計において重要な要因となることを示す。また、リーク電力を効果的に削減する方法として提案されている技術について説明し、それぞれの特徴と問題点を示す。

第 3 章では、第 2 章で述べた研究背景をもとに、著者が解くべき研究課題について述べる。

第 4 章では従来手法として提案されている細粒度 PG について述べ、従来の PG 制御手法の問題点を示す。また、より効果的な細粒度 PG の制御手法として著者が考案した新しい手法を提案し、その効果について評価を行う。さらに、従来手法と提案手法の性能を比較した結果をもとに、最も効果的にリーク電力を削減できる PG 制御の方法について考察を行う。

第 5 章では不揮発性 PG を実現するために、提案されている従来手法について述べる。さらに、従来技術の問題点を指摘し、新しい不揮発性 PG の回路を提案し、従来手法との比較評価を行う。また、提案手法を LSI 規模に適用した際の効果や課題について考察を述べる。

第 6 章では第 4 章、第 5 章で示した結果をもとに、細粒度 PG と不揮発性 PG を LSI に適用することについての考察を述べる。

第 7 章で結論を述べ、第 8 章にて今後の課題について述べる。

## 第2章 研究背景

---

本章では本研究で扱う CMOS 回路の消費電力の定義について述べる。さらに、トランジスタの微細化の動向について示し、低消費電力化を目指すうえでリーク電力が問題となってきたことを示す。それら踏まえた上で、これまでに報告されているリーク電力削減技術について述べる。

### 2.1. 消費電力

CMOS 回路の消費電力には平均電力とピーク電力の 2 つがある。ピーク電力は設計上想定される最大消費電力である。また平均電力はバッテリー寿命や発熱に関係しており、以下のように表される。

$$P_{\text{total}} = P_{\text{dynamic}} + P_{\text{short-circuit}} + P_{\text{leakage}} \quad (\text{式 2.1})$$

#### 2.1.1 ダイナミック電力 ( $P_{\text{dynamic}}$ )

ダイナミック電力とは、CMOS 回路の値が変化する際に、回路の充放電によって消費される電力のことである。ダイナミック電力の式は以下ようになる。

$$P_{\text{dynamic}} = CV_{\text{DD}}^2 f \alpha \quad (\text{式 2.2})$$

- C : 負荷容量
- $V_{\text{DD}}$  : 電源電圧
- f : 動作周波数
- $\alpha$  : スイッチング確率

### 2.1.2 ショートサーキット電力 ( $P_{\text{short-circuit}}$ )

CMOS回路がスイッチングする際に、pMOSとnMOSの両方がオン状態となる時間が一瞬存在する。このとき、電源電圧からグランドに直接電流が流れてしまう電流を貫通電流と呼ぶ。この貫通電流によって消費される電力をショートサーキット電力と呼ぶ。

$$P_{\text{short-circuit}} = I_s \Delta t_s V_{DD} f \alpha \quad (\text{式 2.3})$$

- $I_s$  : 貫通電流
- $\Delta t_s$  : スwitchング時間
- $V_{DD}$  : 電源電圧
- $f$  : 動作周波数
- $\alpha$  : スwitchング確率

### 2.1.3 リーク電力 ( $P_{\text{leakage}}$ )

回路が動作していないとき(待機状態)でも流れてしまう漏れ電流をリーク電流と呼ぶ。これによって消費される電力がリーク電力である。リーク電流は以下のように4つに分類することができる(図2.1.1)。[10,11,15]

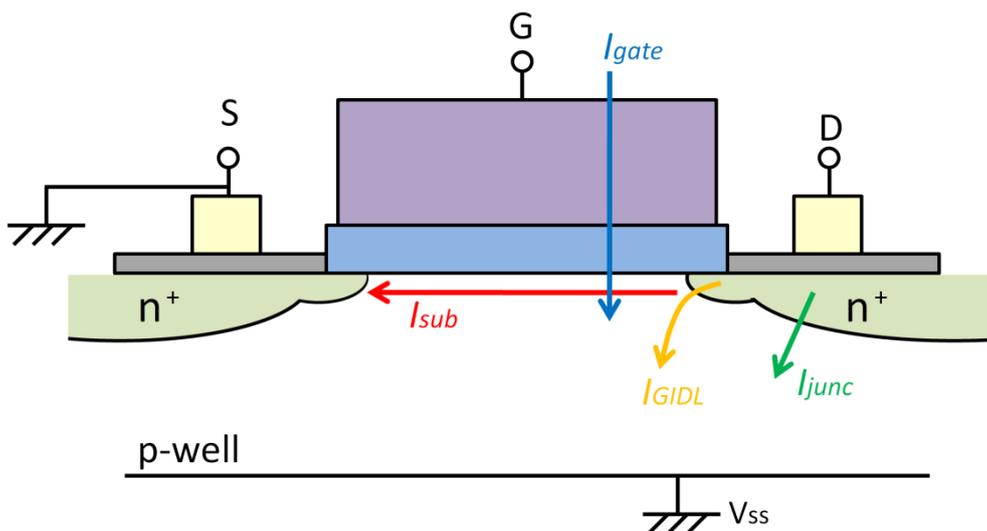


図 2.1.1 リーク電流の主成分

(a) ゲートリーク電流 ( $I_{\text{gate}}$ )

ゲート絶縁膜を介してゲートとチャンネル間に流れる電流のことで、式 1.4 の関係を持つ。

$$I_{\text{gate}} \propto \exp\left(-\frac{T_{\text{ox}}}{V_{\text{DD}}}\right) \quad (\text{式 2.4})$$

$V_{\text{DD}}$  : 電源電圧

$T_{\text{ox}}$ : ゲート絶縁膜厚

ゲートリーク電流はゲート絶縁膜  $T_{\text{ox}}$  が薄くなるにつれて指数関数的に増大する。LSI の製造プロセスが微細化するにつれ、ゲート絶縁膜は薄くなっている。

(b) サブスレッショルドリーク電流 ( $I_{\text{sub}}$ )

トランジスタがオフ状態の時にドレイン・ソース間に流れる電流である。この電流はトランジスタのしきい値の低下と温度上昇により指数関数的に増大する。

$$I_{\text{sub}} \propto W \times \exp\left(-\frac{V_{\text{th}}}{n \times U_{\text{T}}}\right) \quad (\text{式 1.5})$$

$W$  : トランジスタ幅

$V_{\text{th}}$  : トランジスタしきい値

$n$  : サブスレッショルド係数

$U_{\text{T}}$  : 熱電圧係数

(c) ジャンクションリーク電流 ( $I_{\text{junc}}$ )

トランジスタのドレインやソースの pn 接合部に流れる電流である。接合部に逆方向の電圧がかかると空乏層領域が形成されリーク電流が流れる。ジャンクションリーク電流は pn

接合の面積に比例して増加するが、微細化の進んだトランジスタでは、ゲートリーク電流やサブスレッショルドリーク電流に比べるとその電流量は大きくない。

#### (d) GIDL(Gate Induced Drain Leakage)電流 ( $I_{GIDL}$ )

ゲート電極下のドレイン端に高い電界がかかった際に、ドレインから基板へ流れる電流である。GIDL 電流の量はトランジスタの酸化膜の薄膜化と電源電圧の上昇に依存する。

## 2.2. トランジスタの微細化の傾向

近年、大規模集積回路である LSI の発展のスピードは目覚ましいものがある。今や LSI は、テレビや冷蔵庫、パソコンや洗濯機など、ありとあらゆる家電製品に搭載され人々の生活と密接に関わっており、これなしでは今日の社会はありえないだろう。また、携帯電話やポータブル音楽プレーヤー、タブレット PC などの携帯機器の普及により LSI の小型化、大容量化、高機能化の需要が高まり、現在進行形で日進月歩の進化を続けている。その進歩を支えているのは LSI の製造プロセスの微細化技術であり、高性能化、高密度化を可能にしている。

LSI は MOS(Metal Oxide Semiconductor)トランジスタと呼ばれる微細なスイッチとそれらを接続する金属配線によって構成されている。LSI に搭載されているトランジスタ素子の数は性能向上とともに今日まで増加し続けており、その速度は米 Intel 社の共同創設者である Gordon Moore 氏が 1965 年に提唱した「ムーアの法則」[12]を指標としている。これは「半導体に集積されるトランジスタの数は 18 カ月ごとに倍増する」というもので、これによると LSI の集積度は指数関数的に増加する。1975 年には LSI の複雑化を考慮してその倍増ペースを 24 ヶ月ごとに修正しているが、現在もこの法則に則した形で性能向上が図られている(図 2.2.1)。

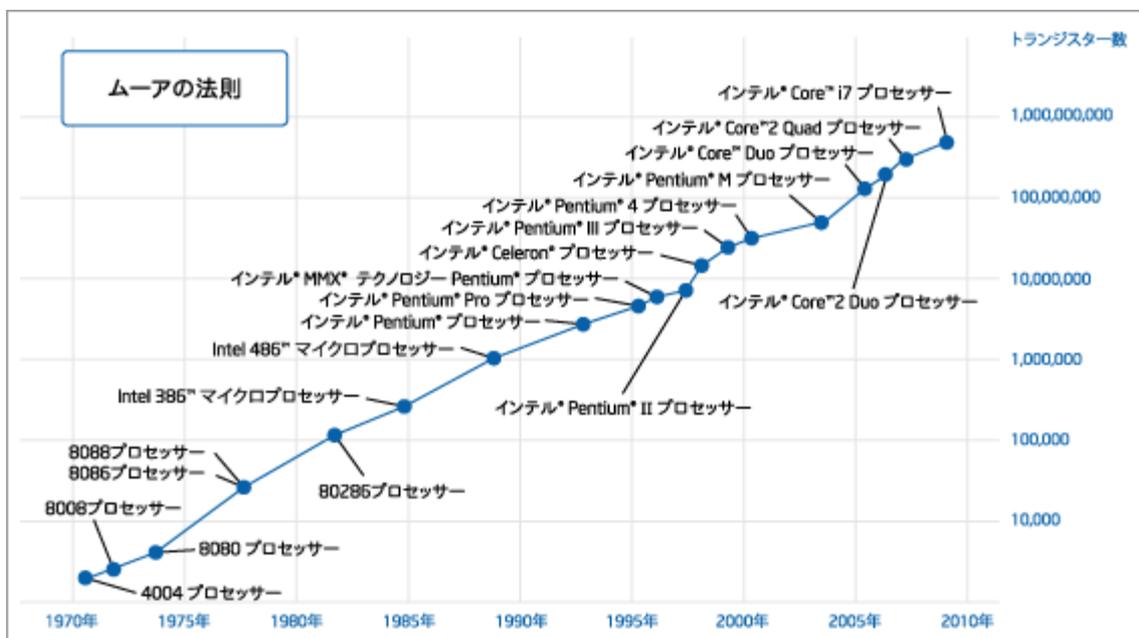


図 2.2.1 ムーアの法則と Intel 社製造マイクロプロセッサ[12]

## 2.3. LSI の低消費電力化が求められる理由

製造プロセスの微細化とともに LSI は高密度化と高性能化を実現し、大きな発展を遂げてきた。一方で、最近ではプロセスの微細化に伴う消費電力の増大が大きな問題として取り上げられている。消費電力として発散される熱エネルギーは LSI 自身の動作速度を低下させ、高温状態が続くことで物理的なダメージを与える。

そこで近年の非常に発熱量の高い LSI では、熱を効率的に排出する必要が生じている。最も一般的な方法は、パッケージの素材に放熱効果の高いセラミックを用いることである。セラミックパッケージのみで十分な放熱が行えない場合は、放熱ヒートシンクや冷却ファンなどの効果の高い冷却装置を併用することで対応を行う。しかしこれらの冷却装置は大型な物が多く、ノートパソコンや省スペースパソコンに使用する場合には困難となる場合があり、駆動部の増加によってシステム全体の信頼性も低下する。

LSI の発熱量は消費電力と動作速度に依存する。急速に性能を上げる近年のままのペースで CPU の動作速度を向上させていくと、近いうちに発熱量が冷却装置の限界を追い越してしまうため、性能向上が頭打ちとなってしまうことが予想されている。

また、ここ数年で爆発的に増加したモバイル機器やセンサ回路においては、バッテリーの長寿命化や低電力化が大きな課題となっている。デジタルカメラや携帯電話、スマートフォンやタブレット PC といった機器は、従来では考えられなかったような高度な機能を実現している。LSI を高速で動作させるためには多くの電力を使わなくてはならず、内蔵されている全ての LSI の消費電力はバッテリーの寿命に直結する。現在のバッテリーの大きさを維持しつつ、寿命をのばすためには製造技術の革新が必要であるが、それには開発期間とコストの増大が発生することはもはや避けられない。また、環境発電などで電源をまかなう従来の電源電圧よりもより低い条件で安定した動作が求められるセンサ回路の需要が高まっている。以上の理由から、LSI の低消費電力が求められ喫緊の課題となっている。

## 2.4. リーク電力の問題

一方で、LSIの消費電力の成分を見てみると、トランジスタの微細化に伴いリーク電力が大きな課題となってきた。図2.4.1は製造プロセスの微細化に伴うダイナミック電力とリーク電力の推移を示したものである[13]。プロセスの微細化に伴い、ダイナミック電力だけでなく、リーク電力が消費電力の大きな要素となった。その結果、図2.4.2で示すように、低消費電力化設計のためには、LSIが待機状態である時だけでなく、動作状態でもリーク電力が非常に大きな問題となってきた。そのため、LSIが動作している時、待機している時で、リーク電力を削減することにより、さらなる低消費電力化設計の実現が期待される。

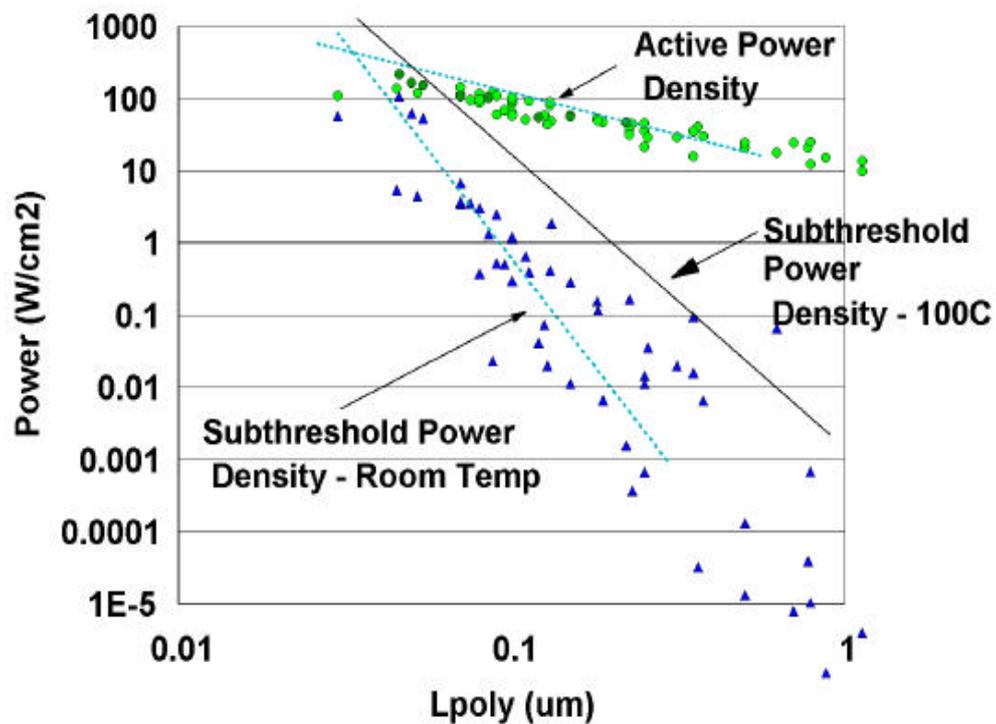


図 2.4.1 製造プロセスの微細化に伴う消費電力成分の推移[13]

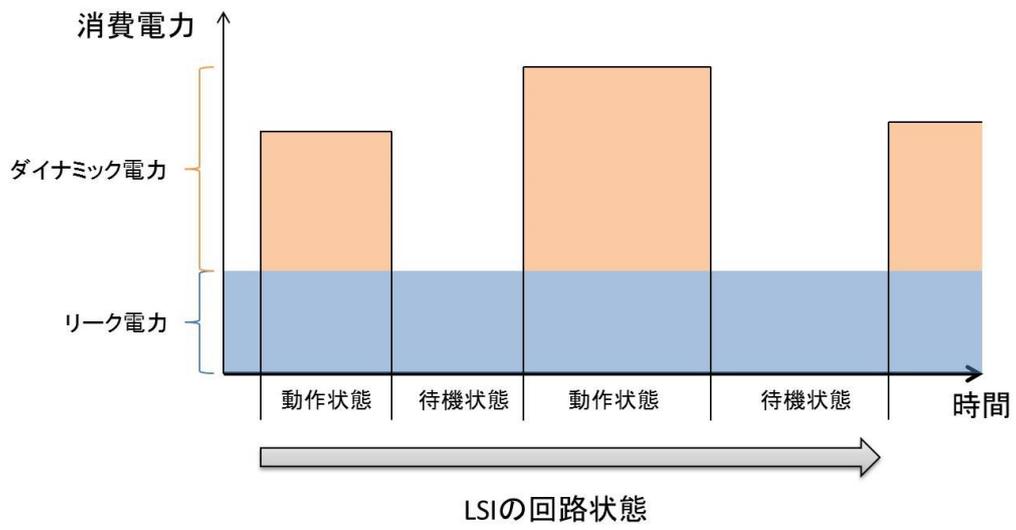


図 2.4.2 ダイナミック・リーク電力

## 2.5. リーク電力削減技術

### 2.5.1 マルチ閾値電圧 (Dual-Vth) 技術

Dual-Vth 技術は 2 種類のしきい値電圧を用いることで、高速動作かつリーク電力の削減を実現する技術である[14,16]。まず論理回路の内、動作速度に影響のある経路（クリティカルパス）のセルにしきい値電圧が低く設定されている低しきい値のセルを用いる（図 2.5.1）。これにより論理回路は高速動作が可能である。一方で、動作速度に影響のないセルにはしきい値を高く設定した高しきい値のセルを用いることで、動作速度に影響を与えずにリーク電力を削減することが可能である。動作速度の維持とリーク電力の削減の両立が比較的容易であるという特徴がある。しかし、回路設計の段階で低しきい値セルと高しきい値セルの動作速度とリーク電力の正確なライブラリ情報が必要である。そのため、動作速度とリーク電力の最適化のために、回路設計ごとに低しきい値と高しきい値の電圧をカスタマイズすることが困難である。また、クリティカルパスは必ず低しきい値セルを用いるため、リークを削減できないセルがどうしても存在してしまう。

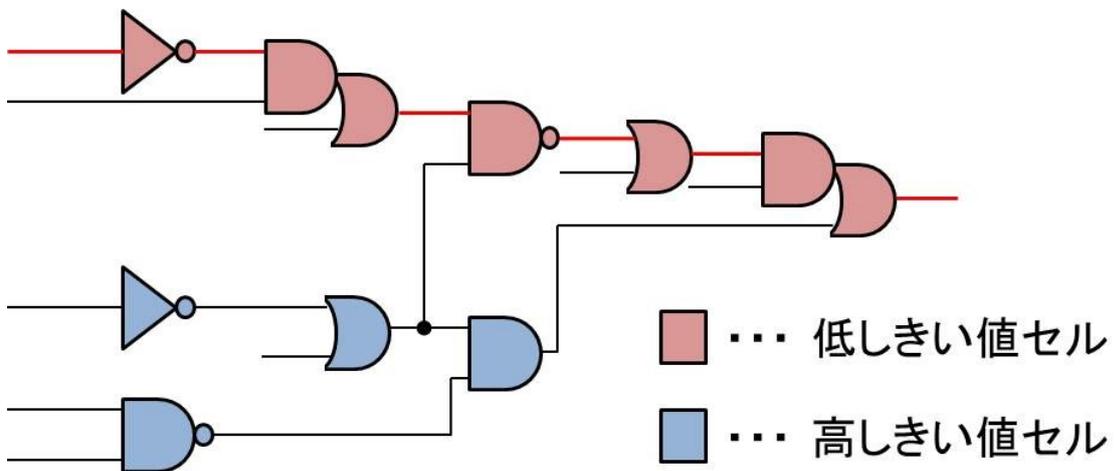


図 2.5.1 Dual-Vth

## 2.5.2 基板バイアス技術

基板バイアス制御技術は、トランジスタの基板電圧を変化させ、基板バイアス効果を生かせることにより、リーク電流を動的に変化させる技術である。図 2.5.2 に示すように、nMOS の基板電圧  $V_b$  には、通常ソース側の電圧  $V_s$  と同様に  $0V$ (pMOS の場合は電源電圧) が与えられている。この際、 $V_s$  と  $V_b$  との間で  $V_s > V_b$ (pMOS では  $V_s < V_b$ )となるような電界が生じると、トランジスタのしきい値電圧が高くなり、リーク電流が低減される。このように、トランジスタの電流が流れにくいように基板に電圧を印加することを逆バイアス基板印加(RBB: Reverse Body Bias)と呼ぶ。また、 $V_s < V_b$ (pMOS は  $V_s > V_b$ )になるように電圧を印加する順バイアス基板印加(FBB: Forward Body Bias)を行うことにより、しきい値電圧を低下させ、リーク電流を増加させることもできる(図 2.5.3)[19]。これにより、回路の待機時に RBB を行うことでリーク電流を削減し、回路の動作時には FBB を行うことで高速動作を可能にする[4]。

しかし、近年の LSI の回路設計ではトランジスタの高密度化を実現するため、セルごとといった細かい粒度で基板電圧を印可する Well を分離することは非常に困難であり、また実現するためには面積の大幅な増大が必要になってしまう。また基板バイアス制御を行うためには、論理回路の電源電圧とは別の電源線が必要になり、またその電源線による基板電圧の動的な電圧制御が求められる。このとき、電圧制御による基板電圧の充放電により、消費エネルギーが生じてしまう。そのため、基板バイアス技術をより細かい粒度や短い頻度で適用すると、基板バイアス制御に伴うエネルギーオーバーヘッドが大きな問題となってしまう。

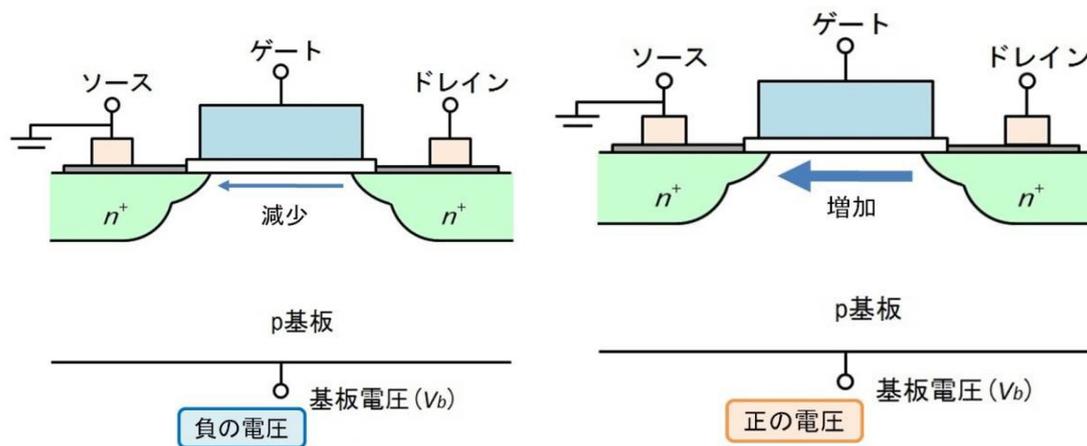


図 2.5.2 基板バイアス制御 (nMOS)

対象	基板電圧	リーク電流	閾値電圧
nMOS	$V_g = V_s$	---	---
	$V_g < V_s$	減少	増加
	$V_g > V_s$	増加	減少
pMOS	$V_g = V_s$	---	---
	$V_g < V_s$	増加	減少
	$V_g > V_s$	減少	増加

図 2.5.3 基板バイアス制御によるリーク電流および閾値電圧の変化

### 2.5.3 パワーゲーティング (PG) 技術

PG 技術は、パワースイッチ(PS)と呼ばれるリーク電流遮断用のトランジスタを論理回路と電源（もしくはグランド線）の間に挿入することにより、回路が使用されていない期間のリーク電流を削減する技術である[6,17,58]。pMOS トランジスタを電源線と仮想電源線の間に入挿するヘッダ型 PG (HDPG) と、グランド線と仮想グランド線の間に入挿するフッタ型 PG (FTPG) がある(図 2.4.4)[18]。HDPG では PG 適用回路が待機状態のとき、イネーブル信号 (EN) を '0' にすることで PS はオフし、リーク電流を遮断する。一方、動作状態のときには、EN を '1' にし PS はオンにすることで、PG 適用回路は通常動作を行うことができる。FTPG では EN を '0' にすることで PS をオフすることができる。また EN を '1' にすることで、PG 適用回路は通常動作が可能となる。PG 技術は市販の CPU などにも適用されてきている。さらに、より細かい粒度で PG を実行する細粒度 PG[21,22,23]や、不揮発性の素子を用いることで回路の状態を保持したまま PG が可能になる不揮発性 PG[24,25]などが提案されてきている。

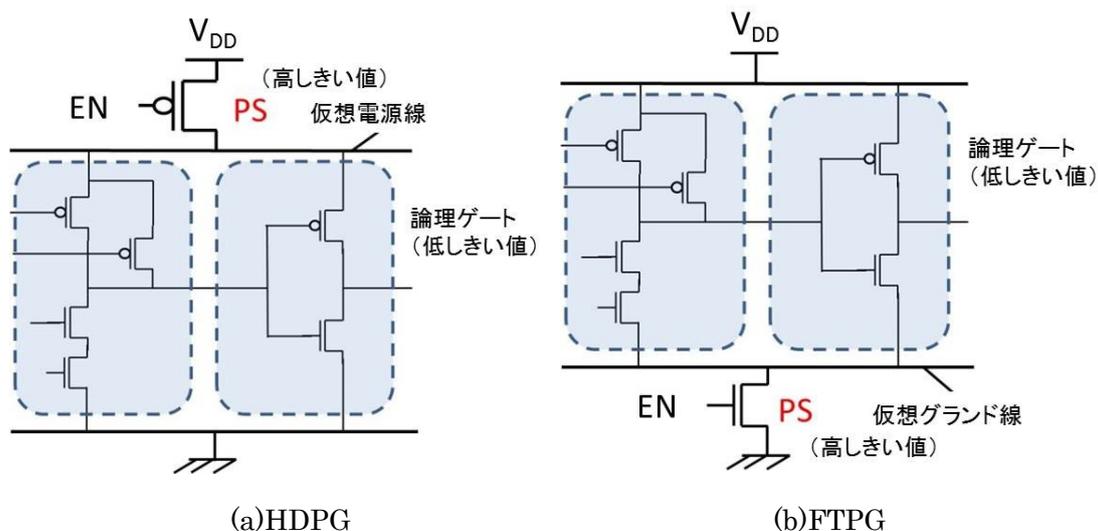
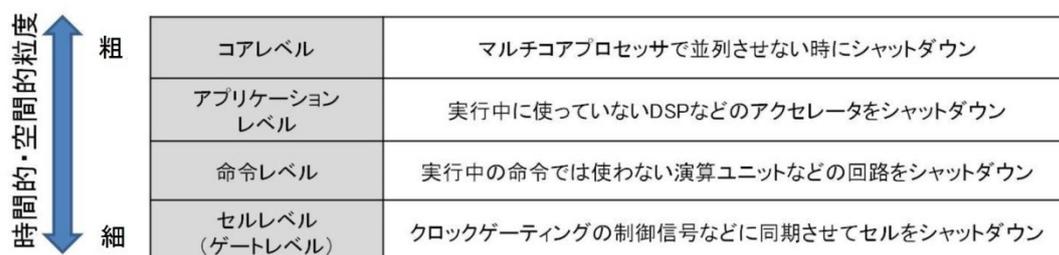


図 2.5.4 PG 技術

## (a) 細粒度 PG

PG は時間的・空間的観点から粗粒度と細粒度に分類することができる (図 2.5.5)。コアレベルやアプリケーションレベルの PG では、適用領域が広範囲に及ぶ (空間的粗粒度) ことや、シャットダウンの機会が OS やユーザーが実行するソフトウェアに依存する (時間的粗粒度) といった特徴を持つ。近年、SoC のマイクロプロセッサやマルチコアの CPU が当たり前となってきており [26]、図 2.5.6 に示すように、使用していないコアレベルの回路に対し PG を行うことで低消費電力を実現した CPU が実用化され始めている [27,28]。

これに対し、命令レベル、セルレベルの PG では、粒度の細かい領域 (空間的細粒度) に PG を適用した手法であり、粗粒度 PG と比べてより頻繁にシャットダウンを行うことができる (時間的細粒度)。粗粒度 PG では実行できない細かい時間で細粒度 PG は PS をオフにすることができるため、さらなるリーク電力の削減が実現できる。しかし、粗粒度 PG に比べて、PS をオフにしている時間が短いため、PS のオン・オフに伴う電力オーバーヘッドが問題となる (図 2.5.7) [29]。



時間的・空間的粒度 ↑粗 ↓細	コアレベル	マルチコアプロセッサで並列させない時にシャットダウン
	アプリケーションレベル	実行中に使っていないDSPなどのアクセラータをシャットダウン
	命令レベル	実行中の命令では使わない演算ユニットなどの回路をシャットダウン
	セルレベル (ゲートレベル)	クロックゲーティングの制御信号などに同期させてセルをシャットダウン

図 2.5.5 PG における時間的・空間的粒度

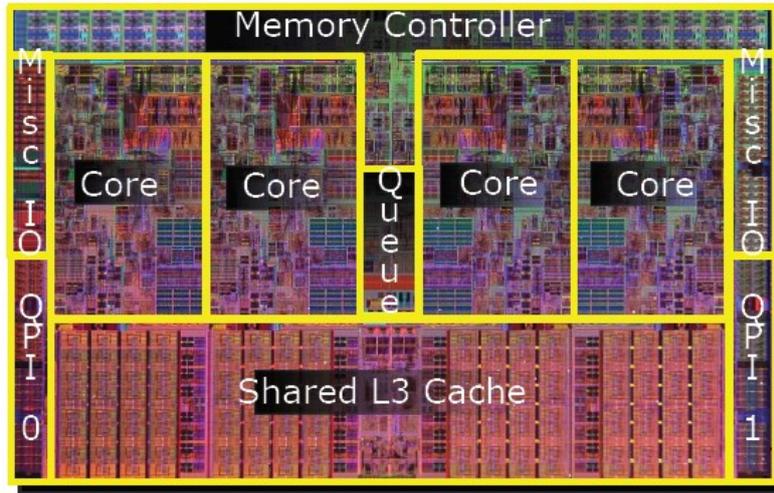


図 2.5.6 粗粒度 PG の実例

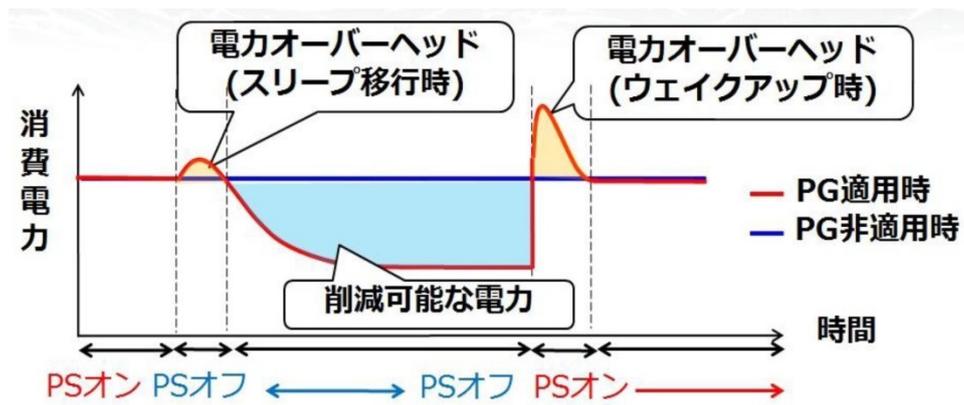


図 2.5.7 PG における消費電力の推移

## (b) 不揮発性 PG

PGの対象となる回路内に、フリップフロップ (FF) やラッチ等の記憶回路が含まれている場合、PGを行うと記憶回路に保持されていたデータが破壊されてしまう。そのため今までは、PGから復帰後にリセットを行い、また初期状態から動作をスタートさせる方法やPG前にFF中の記憶データをPG対象外のSRAM等のメモリへ退避させ、復帰後にFFにデータを戻すという方式が取られていた[30]。

しかし、近年、磁性体を用いた電子回路であるスピントロニクスが提案され始めた[31,32,33]。抵抗変化型メモリ (ReRAM) や、強誘電体メモリ (FeRAM)、STT-MRAMが提案され、不揮発性PGが可能になった。特に磁気トンネル接合 (MTJ : Magnetic Tunnel Junction) を使用するSTT-MRAMは、書き込みに必要な電圧が低く、また繰り返しの書き込みに対する耐性が高いという特徴がある。MTJを用いた不揮発性PGが提案されてきている。[34,35,36]

## 2.6. 研究課題の定義

上記までに、研究背景について述べ、低消費電力の LSI を設計する上で、リーク電力の削減が非常に重要な要因となっていることを示した。また、効果的なリーク電力の削減が期待できる技術として、細粒度 PG と不揮発性 PG について説明した。

しかし、細粒度 PG と不揮発性 PG には、以下のような課題が存在する。

### ● 細粒度 PG

細粒度 PG では、PS のオン・オフが頻繁に切り替わるため、PG のオーバーヘッドが問題となっている[29]。そのため、効果的な PG を制御するためのスリープ制御手法が提案されている[21]。しかし、それらのスリープ制御手法を回路で実現した際の、制御回路に必要なエネルギーのオーバーヘッドについては考慮されてこなかった。スリープ制御手法によっては、制御回路のエネルギーオーバーヘッドによって、細粒度 PG のリーク削減効果が大きく変わってきてしまう可能性がある。そのため、スリープ制御回路のエネルギーオーバーヘッドを小さくし、効果的な細粒度 PG を行うスリープ制御手法を求める必要がある。

### ● 不揮発性 PG

不揮発性 PG を実現するために、一般に記憶回路として使用されるフリップフロップ回路である D-FF を不揮発化した回路が提案されている[52]。この不揮発性フリップフロップ (NVFF) により、従来の回路設計を大きく変える必要が無いまま不揮発性 PG を実現することができる。しかし、この NVFF はシミュレーションのみで評価が行われており、シリコン上での測定はまだ行われていない。しかし、シリコン上で実装し、測定を行わなければ、プロセスのばらつきや回路面積のオーバーヘッドの影響などを正しく評価することができない。これらの影響は、NVFF が安定して動作するかどうかのロバスト性に大きくかわる。そのため、NVFF をシリコン上で評価し、NVFF のロバスト性を評価する必要がある。また、通常フリップフロップ回路には、誤り検出訂正機能 (ECC : Error Check and Correction) が付いていない。そのため、NVFF のロバスト性が低い場合には、原因を究明し、ロバスト性を高める必要がある。

以上の研究課題を解くべく、第 4 章では細粒度 PG の研究について述べ、第 5 章では不揮発性 PG について述べる。

## 第3章 細粒度 PG におけるスリープ制御手法

---

本章では、マイクロプロセッサの低消費電力設計として、細粒度 PG を用いた設計手法に関する研究成果について述べる。まず、細粒度 PG を適用したマイクロプロセッサとして著者らが設計し実装した **Geysler-3** について説明し、細粒度 PG が従来の PG に比べよりリーク電力を削減可能なことを示す。次に、細粒度 PG を制御し、パワースイッチ (PS) のオン・オフを行う際に、より効果的なリーク削減効果を得るためのスリープ制御手法が提案されている[21]。これら従来手法を説明するとともに、その問題点について議論する。さらに、この問題点を解決するために、新しいスリープ制御手法を提案する。細粒度 PG における各スリープ制御手法のエネルギー削減効果について、モデル式を立てた上で定量的に論ずる。以上を踏まえた上で、**Geysler-3** 上での各スリープ制御手法の消費エネルギーを評価し、著者が提案したスリープ制御手法の有効性を示す。さらに、回路環境およびトランジスタのばらつきによる影響についても議論する。

### 3.1 Geysler-3

**Geysler-3** は著者らが 65nm プロセスを用いて設計した MIPS R3000 のアーキテクチャを持つ 32bit のマイクロプロセッサである[37]。さらに、演算器として細粒度 PG を適用した ALU、シフト回路 (SHIFT)、乗算器回路 (MULT)、除算器回路 (DIV) の 4 つの回路を持つ。4 つの演算器はすべて組み合わせ回路のみで実現されているため、PG が可能である。また、動作周波数は 200MHz を想定して設計した。各演算器には PG を実装するために、細粒度 PG およびグラウンドバンスの低減技術の適用を行っている。以下にて、本チップで行った PG 技術の説明を行う。

#### 3.1.1 細粒度 PG の設計と実装

図 4.1.1 には細粒度 PG の適用領域および PG の制御方法について示す。**Geysler-3** のアーキテクチャではシングルパイプラインを採用しており、命令フェッチ (IF)、命令デコード (ID)、実行 (EX)、メモリアクセス (MEM)、レジスタ書き込み (WB) の 5 つのステージで構成されている。さらに、4 つの演算器それぞれに PG が適用されており、PS のイネーブル信号 (EN) も独立で制御されている。これにより、使用される演算器を IF ステージで先読みすることにより、EX ステージの時に使用される演算器以外の 3 つの演算器を演

算器選択回路によって PG することが可能である。そのため、マイクロプロセッサ全体が動作している間も、ランタイムパワーゲーティングにより無駄なリークエネルギーを削減することができる[23]。これにより、図 4.1.2 に示すように、粗粒度 PG に比べ細粒度 PG ではより頻繁に PG を行うことができ、さらなるリーク電力の削減が期待できる。しかし、粗粒度 PG では PG の制御がマイクロ秒からミリ秒オーダで求められるのに対し、細粒度 PG はナノ秒オーダでの PG の制御が必要となってくる。そのため、PS のオン・オフが頻繁に切り変わることにより、図 2.3.6 で示したように削減可能な消費エネルギーに対し、エネルギーオーバーヘッドの影響が顕著になってくる。もしスリープ制御時間が短く、回路のリークエネルギーの削減効果よりも PG のオーバーヘッドによるエネルギーが大きくなった場合、回路の消費エネルギーは増加してしまう[37]。この際、PG によるリークエネルギー削減効果が PG のエネルギーオーバーヘッドを上回るスリープ時間を損益分岐時間 (BET : Break Even Time) と呼ぶ。細粒度 PG ではこの BET を考慮したスリープ制御手法が求められる。Geysler-3 では Whenever Idle Put to Sleep (WIPS) 方式と呼ばれるスリープ制御手法を採用している。細粒度 PG のスリープ制御手法については 4.2 節にて詳しく説明する。

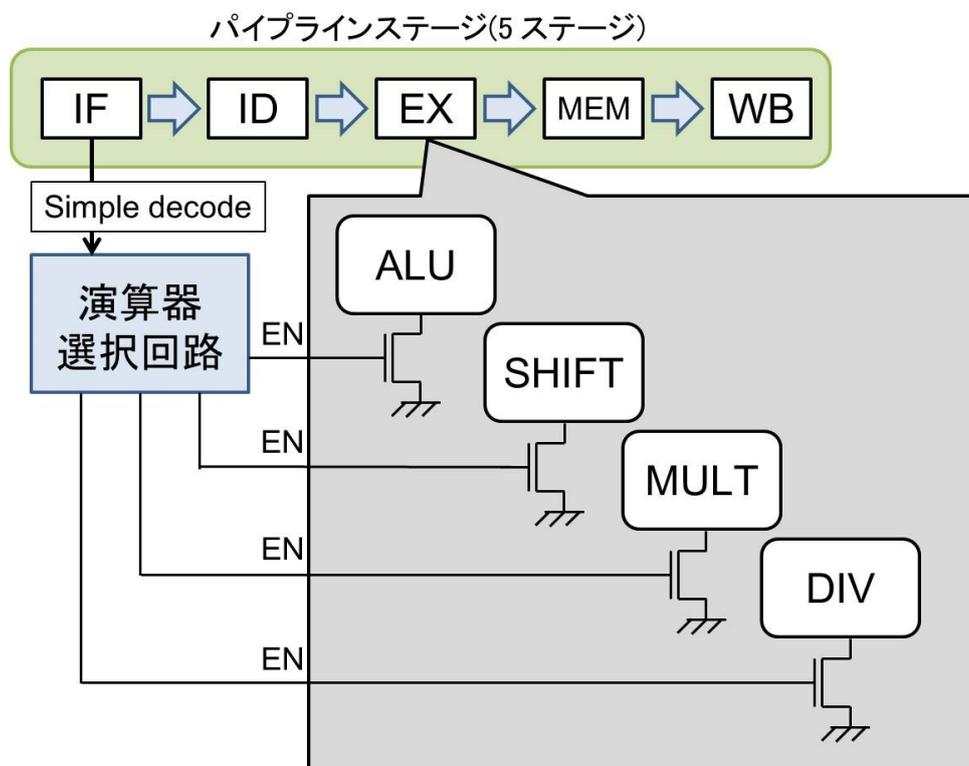


図 4.3.1.1 細粒度 PG におけるアーキテクチャ制御

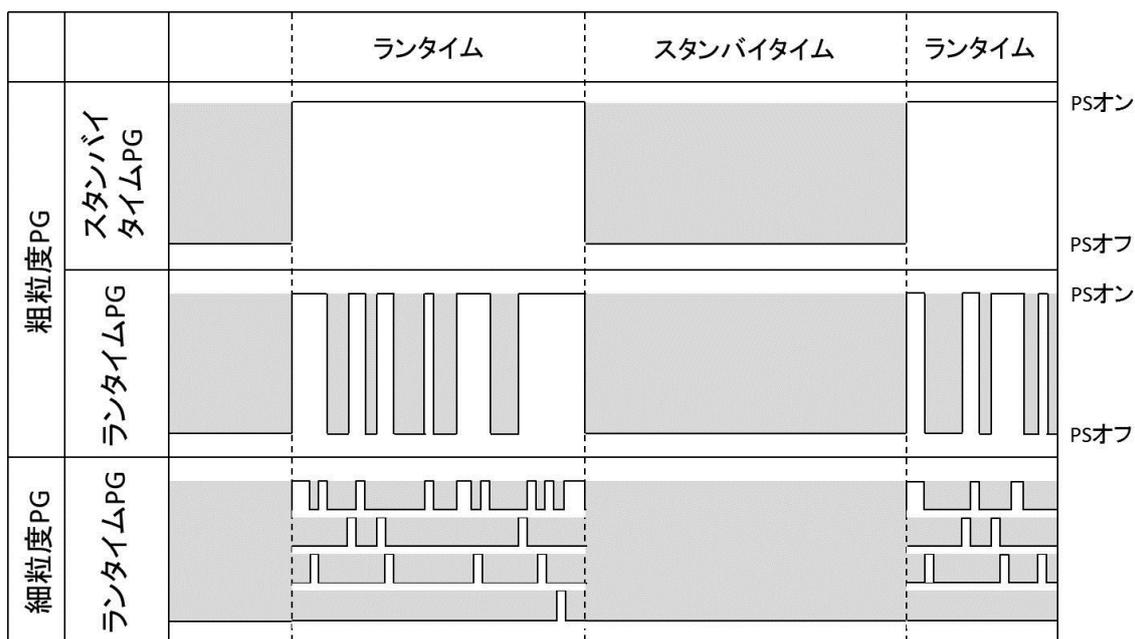


図 4.3.1.2 PG 適用粒度による PG 制御タイミング

さらに、実チップによる測定評価を行うため、Geyser-3 のレイアウト設計を行った。図 4.1.3 は Geyser-3 全体のレイアウト図である。チップの左半分がマイクロプロセッサのコア部分となっており、左下に 4 つの演算器が配置されている。また、図 4.1.4 は Geyser-3 に搭載するために、著者がレイアウト設計を行った演算器のレイアウト図である。組み合わせ回路のみで設計されている各演算器回路部は回路全体の中心部に配置されており、左右に PS が挿入されている。PS には nMOS を用いたフッタ型 PG (FTP) を採用している。これにより、各演算器回路部は PG 実行中、チップ内のグランド線から分離されリーク電流を遮断することができる。PG さらにその外周には、PS を駆動するための PS ドライバセル (PSD) およびアイソレーションセルが配置されている。また、表 4.1.1 は各演算器の面積を示す。

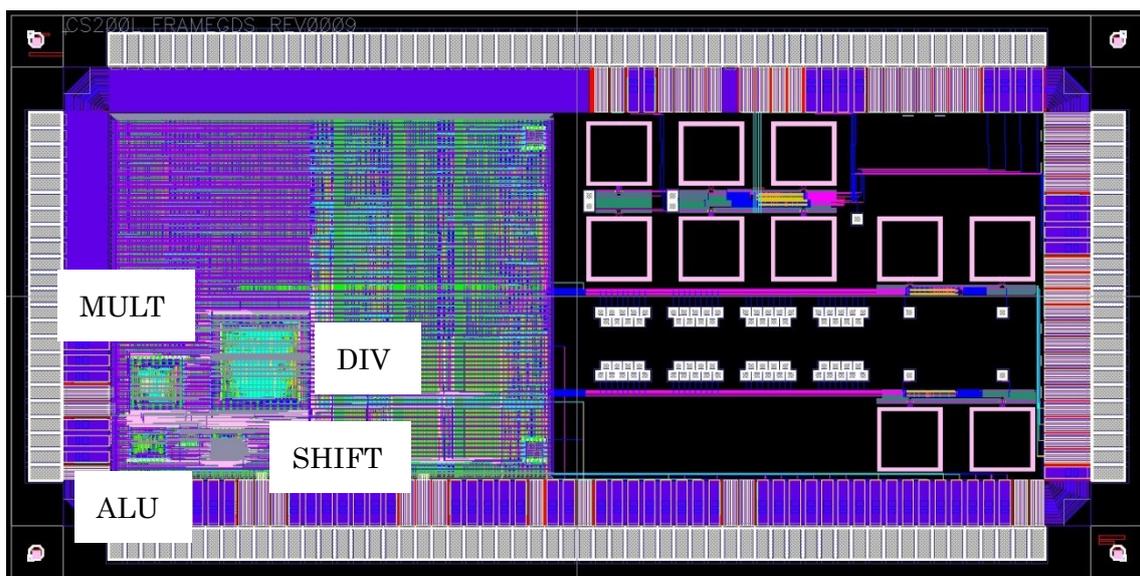
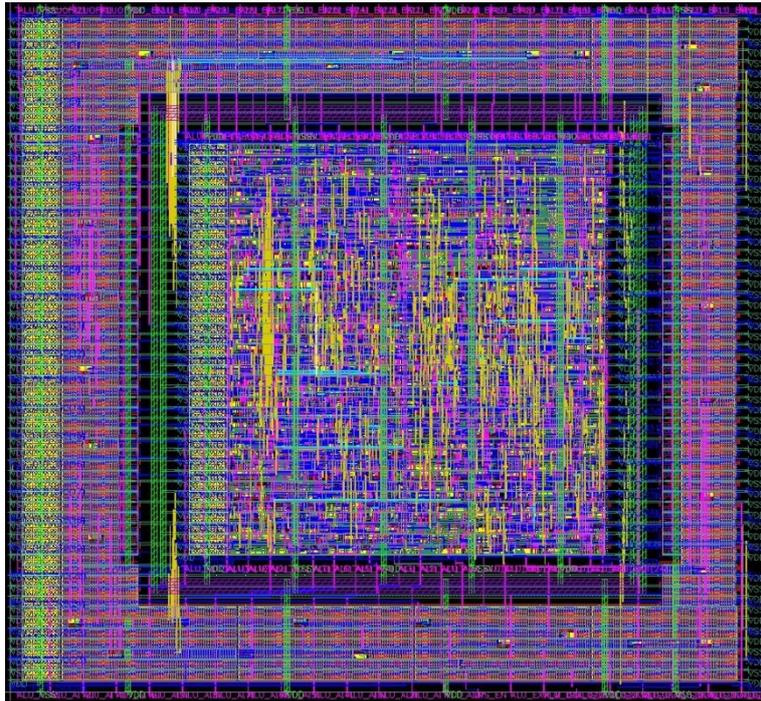


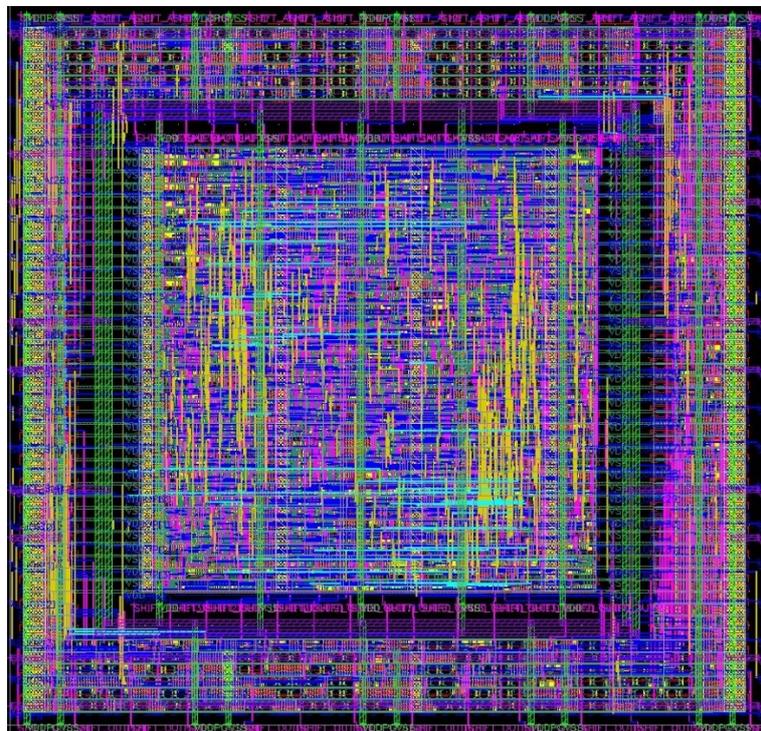
図 4.3.1.3 Geyser-3 のレイアウト図

表 4.3.1.1 演算器マクロの面積

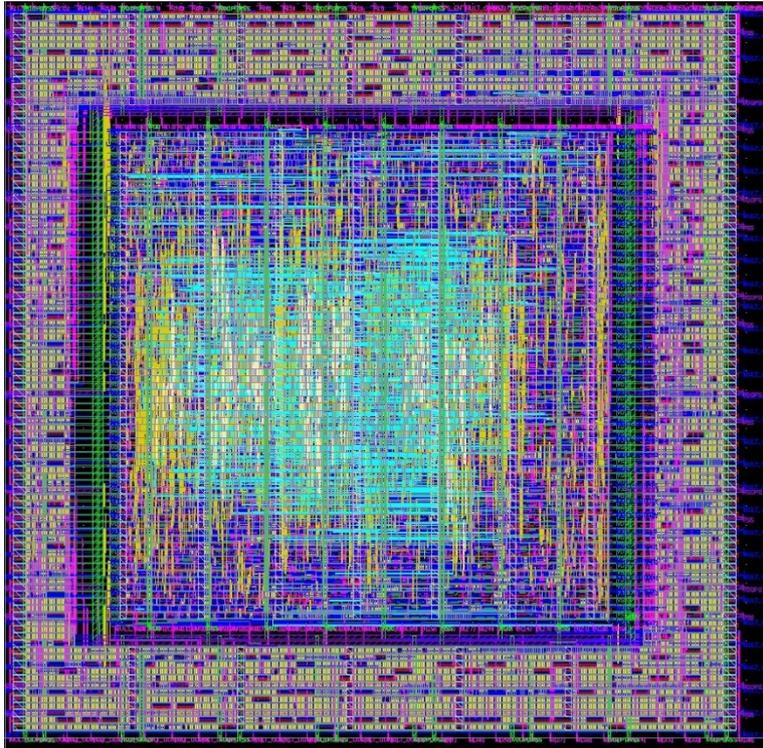
	ALU	SHIFT	MULT	DIV
面積 ( $\mu\text{m}^2$ )	14722	11875	35872	128737



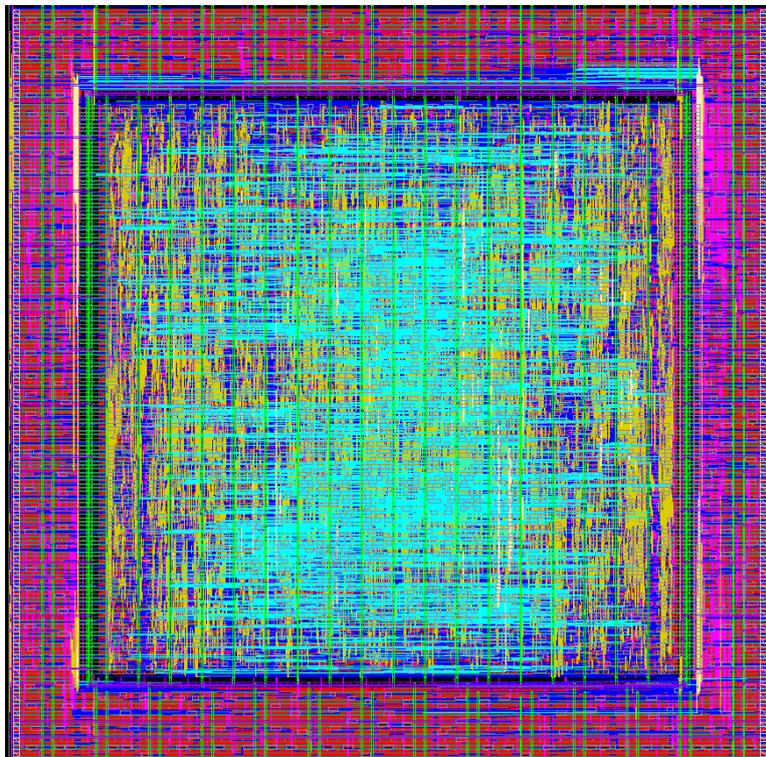
(a) ALU



(b) シフト回路



(c) 乗算器回路



(d) 除算器回路

図 4.3.1.4 Geyser-3 における各演算器のレイアウト図

### 3.1.2 グランドバンス低減技術

PG では PS をオフの状態からオンへ戻す際、すべての PS セルを同時にオンにしてしまうと、PG 回路内に溜まった電荷により、PS を通ってグラウンド線へ大量の電流が流れてしまう。しかし、チップ内のグラウンド線からチップ外のグラウンド線までにはパッケージのボンディングワイヤのインダクタ成分が存在するため、この急激な電流の流れにより、チップ内のグラウンド線の電圧が変化する現象が生じてしまう（グラウンドバンス）[38]。このグラウンドバンスによりグラウンド線の電圧が上昇してしまうと、PS をオンにした後の回路動作に不具合が生じてしまう恐れがある。そのため、PSD のセルのツリー構造を用いて、細かく分割した PS の個々のオンタイミングをずらすことでグラウンドバンスの電圧上昇を抑える必要がある。しかし、この PSD の最適解を求めることは困難になる。そこで本チップでは、遺伝的アルゴリズムを用いてもっともグラウンドバンスを抑えることのできる PSD のツリー構造を求め、レイアウトによる実装を行っている[39]。

## 3.2 スリープ制御手法

4.1.節で述べたように、細粒度 PG は頻繁に PS のオン・オフを繰り返すため、PS をオフしている時間（スリープ時間）が短い場合、PG のエネルギーオーバーヘッドにより演算器の消費エネルギーは増加してしまう恐れがある。そのため、スリープ時間が BET よりも短い（ショートスリープ）のかスリープ時間が BET よりも長い（ロングスリープ）のかを考慮したスリープ制御手法が必要となってくる。以下に、従来手法として提案されているスリープ制御手法について述べる。

### 3.2.1 Whenever Idle Put to Sleep(WIPS)方式

WIPS 方式は PG 適用回路が待機状態になり、PG が可能になると同時に PS をオフする方式である[22]。他のスリープ制御手法に比べ複雑な制御の必要がないため、数個～数十個の論理ゲートで実現できる。そのため、スリープ制御回路の面積および消費エネルギーのオーバーヘッドが非常に少ない特徴がある。しかし、BET を全く考慮していないため、PS のオン・オフが頻繁に切り替わる回路では、ショートスリープが頻繁に発生してしまい消費エネルギーが増大するという問題がある。

### 3.2.2 Time-Based(TB)方式

TB方式は、PG適用回路が待機状態になると、スリープ制御回路内のカウンタ回路がカウントを開始し、カウント数がBETを上回ると同時にPSのENをオフにする方式である[21]。図4.2.1にTB方式によるスリープ制御の例を示す。左の表は、演算器が待機状態にあり、PGが可能な時間とそれがアプリケーション実行中に何回発生するかを示したものである。TB方式ではBETよりも長い待機時間のスリープイベントのみPGが実行され、BETよりも短い待機時間のスリープイベントではPGが実行されない。これにより、ショートスリープによる消費エネルギーの増加を効果的に抑制することが可能である。

しかし、TB方式を実現するためには、スリープ制御回路にカウンタ回路および比較回路が必要になる(図4.2.2)。カウンタ回路はクロック信号(CLK)によりカウントを行う。また比較回路はカウンタ回路のカウント結果とBETの比較を行う。カウント結果がBETを上回った場合、出力信号であるENが変化する。PGのスリープ制御を実行する際、これらスリープ制御回路はダイナミックエネルギーを消費する。そのため、PGによるリークエネルギーの削減効果に対し、スリープ制御回路のエネルギーオーバーヘッドが顕著になってしまう恐れがある[40]。

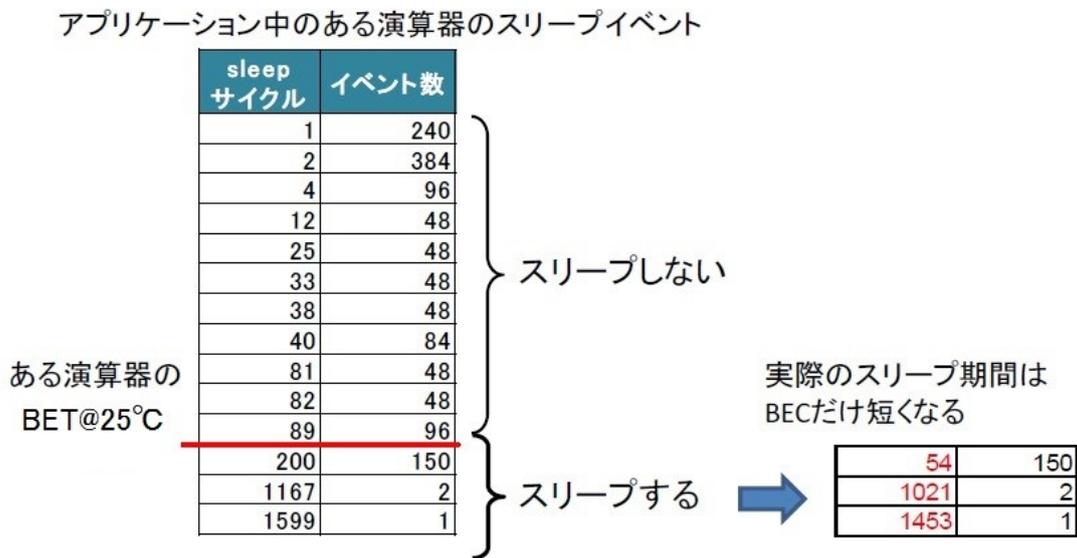


図 4.3.2.1 TB方式におけるスリープ制御

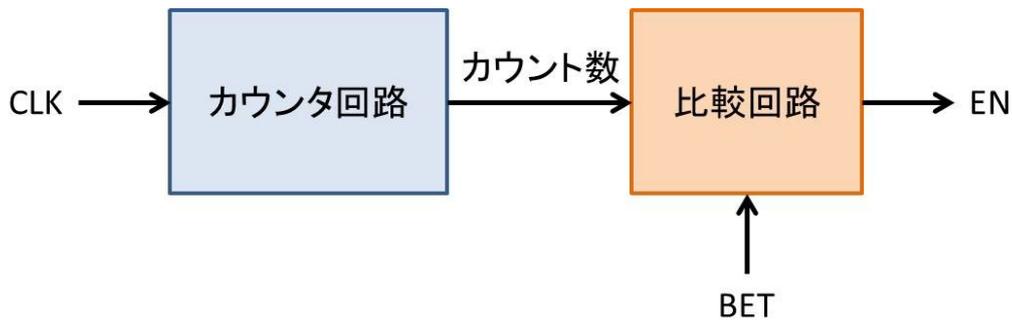


図 4.3.2.2 TB 方式の回路構成

### 3.2.3 Adaptive Time-Based(ATB)方式

TB 方式は PG 適用回路の演算器の待機時間が BET を上回った場合のみ PG を実行する。しかし、演算器の BET はチップ上の温度によって変化することが知られている[41]。そのため、TB 方式では一定の温度のときの BET を基にし、PG を行っている。一方で、チップ内に温度モニタを搭載し、温度モニタの結果をもとに演算器の BET を求め、その結果を基に、TB 方式による PG を実行する手法が提案されている[42]。この手法を ATB 方式と呼ぶ。図 4.2.3 に AT 方式によるスリープ制御の例を示す。チップ上の温度が低い場合、演算器のリーク電流が少なくなる。そのため、PG によるリークエネルギーの削減効果が少なくなり、BET は長くなる。この際、ATB 方式では、PG が実行されるイベント数は少なくなる。これにより、ショートスリープになるはずだったイベントを効果的に避けることが可能である。一方で、チップ上の温度が高温の場合、演算器のリーク電流が増大するため、PG によるリークエネルギーの削減効果が高くなり、BET が短くなる。これにより、ATB 方式では PG を実行するイベント数が増加し、TB 方式と比べ、より多くのリークエネルギーを削減することが可能になる。しかし、ATB 方式を実現するためにはスリープ制御回路として、カウンタ回路および比較回路の他に、温度モニタ回路とルックアップテーブル (LUT) が必要になる。ATB 方式を実現するための温度モニタ回路としてリークモニタ回路[43]が提案されている。しかし、TB 方式と比べ制御回路が複雑かつ大規模になってしまう問題がある。

アプリケーション中のある演算器のスリープイベントの長さと同数

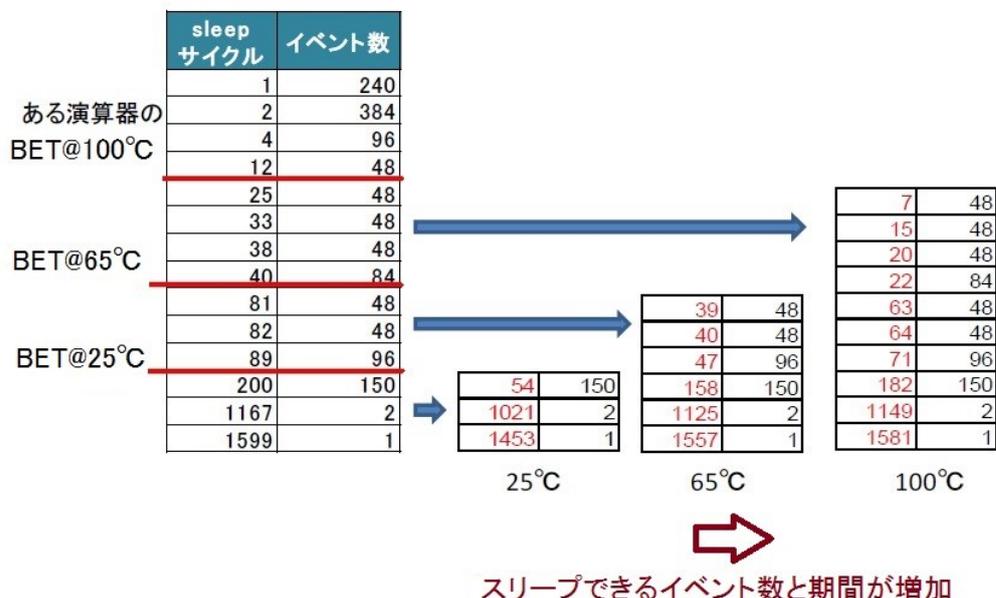


図 4.3.2.3 ATB 方式におけるスリープ制御

### 3.2.4 従来手法の問題点

WIPS 方式は BET についてまったく考慮しておらず、PG 適用回路が待機状態になるとすぐに PS をオフにする手法である。そのため、PS をオフしている時間が BET よりも短いショートスリープが頻繁に起こるアプリケーションで WIPS 方式を実行した場合、消費エネルギーが大幅に増大してしまう。一方、TB 方式や ATB 方式は、PG 適用回路が待機状態になってからの時間を測定することで効果的にショートスリープによる消費エネルギーの増大を防ぐことができる。しかし、TB 方式を実現するためにはカウンタ回路や BET とカウント数の比較回路が必要となる。さらに、ATB 方式にはチップ上の温度をリアルタイムで測定する温度モニタや LUT が必要となる。しかし、これらの回路はダイナミックエネルギーを消費して PG を制御する。この際、制御回路に必要な消費エネルギーは PG 適用回路の削減可能なリークエネルギーに対し、大きな問題となってしまふ。また、ATB 方式は TB 方式よりもリーク削減効果が期待できるが、温度モニタと LUT をより低消費エネルギーかつ小さな回路でチップ上に搭載するという課題が残っている。そのため、ATB 方式のように、チップ上の温度によって PS をオフするタイミングを生成するし、なおかつ制御回路に必要な消費エネルギーを抑えたスリープ制御手法が求められる。

### 3.3 新しいスリープ制御手法の提案

TB 方式、ATB 方式は、WIPS 方式に比べてショートスリープによる消費エネルギーの増加を防ぐことができる反面、制御回路によるオーバーヘッドが増大してしまう欠点がある。そこで著者は、Charge up Delay (CuD) 方式と名付けた新しいスリープ制御手法を提案した[44,45]。本手法は、図 4.3.1 に示すように、演算器の PS トランジスタと EN 信号の間にリークモニタ回路を挿入することで、ATB 方式に類似した制御を行いつつ、制御回路のオーバーヘッドを大幅に削減することが可能である。以下に本手法で用いるリークモニタ回路の回路構成および制御方法について述べる。

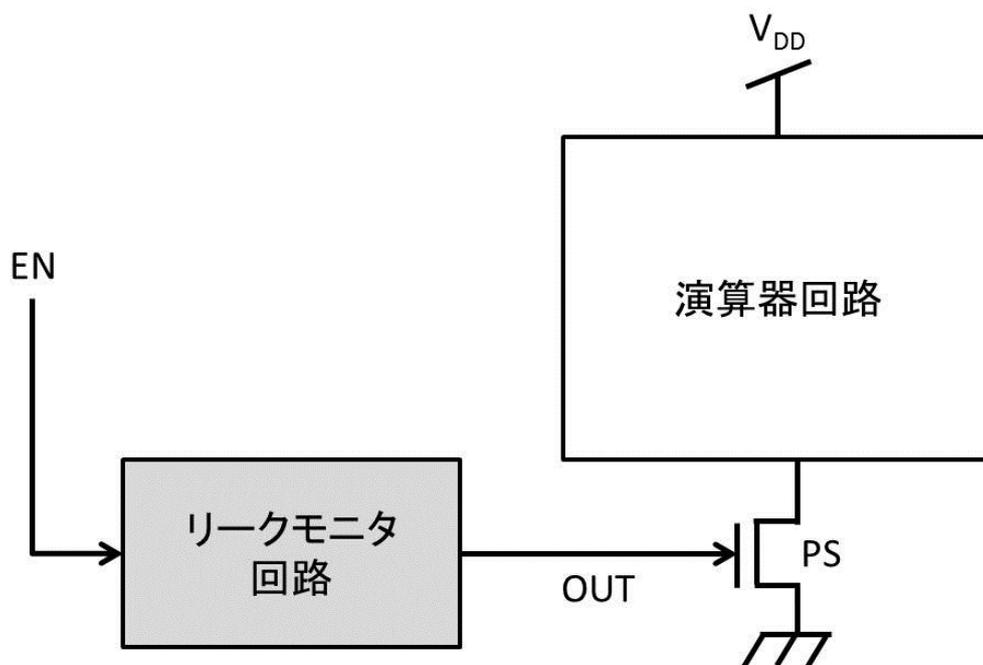


図 4.3.3.1 CuD 方式の回路構成

### 3.3.1 リークモニタ回路

リークモニタ回路はリーク電流をオンチップ上で測定するために提案された回路である[43]。CuD方式で用いるリークモニタ回路の回路構成を図4.3.2に示す。リークモニタ回路はリーク生成回路部と比較回路部の2つから構成される。リーク生成回路部には低しきい値のトランジスタが用いられている。高しきい値のnMOSトランジスタのEN信号が‘0’になると、リーク生成回路部から流れる電荷が仮想グランド線(VGND)に溜まることによりVGNDの電圧が緩やかに上昇する。この時のVGNDの電圧上昇速度はリーク生成回路部のpMOSのリーク電流量によって決定される。リークモニタ回路はこのVGNDの電圧上昇速度を利用してリーク電流量を測定する回路である。また本研究ではリーク生成回路の入力を‘1’に固定することでnMOSをオンにし、VGNDに流れるリーク電流はpMOSトランジスタだけのものを使用している。これは、CuD方式に適用した際に制御をよりシンプルな構造にするためである。しかし、リーク生成回路の入力を可変にすることで、nMOS、pMOSのリーク電流をそれぞれモニタすることも可能である。

また、電圧比較回路の回路構成を図4.3.3に示す。最もシンプルな電圧比較回路として図4.3.3(a)に示すようなカレントミラー型の電圧比較回路が知られている。VGNDの電圧が基準電圧(VREF)よりも低い場合、出力OUTは‘1’となる。また、VGNDがVREFを上回った場合、出力OUTは‘0’に遷移する。一方で、2重構造のカレントミラー型のレベルコンバータ[20]をもとに著者が提案した電圧比較回路を図4.3.3(b)に示す[44]。出力の遷移は(a)と同様であるが、2重構造によりVGNDが緩やかに上昇した場合でも、VGNDがVREFに達するまでの間に、V<sub>DD</sub>からグランド線へ流れる貫通電流を効果的に削減することができる。さらに、電圧比較回路の出力とEN信号を用いてnMOSによるグランド遮断を行うことで、出力が遷移した後の無駄なリーク電流も削減できる。より低消費電力なスリープ制御を実現できることから、本論文では(b)を採択した。また電源電圧としてV<sub>DD</sub>は1.2V、VREFは0.5Vを想定して設計した。

さらに、図4.3.4はリークモニタ回路の高温時と低温時のVGNDの電圧上昇速度のHSPICEシミュレーションを用いて求めた結果である。低温時ではVGNDの電圧の上昇は緩やかになる。このとき、VGNDの電圧がVREFを超えた時にリークモニタの出力は遷移し、PSをオフする。一方、高温時ではVGND電圧が急激に上昇していることがわかる。これにより、素早くVGND電圧はVREFを超え、PSがすぐにオフになる。

また、リークモニタ回路はゆっくりと電圧が上昇するVGNDのセンスアンプ回路になるため、アナログ回路に近い回路構造をとる。しかし、入力信号はデジタル信号であり、出力信号も比較回路の出力結果のインバータ出力になるため、デジタル信号として扱うことができる。そのため、リークモニタ回路はデジタル回路としてスリープ制御手法に用いることができる。

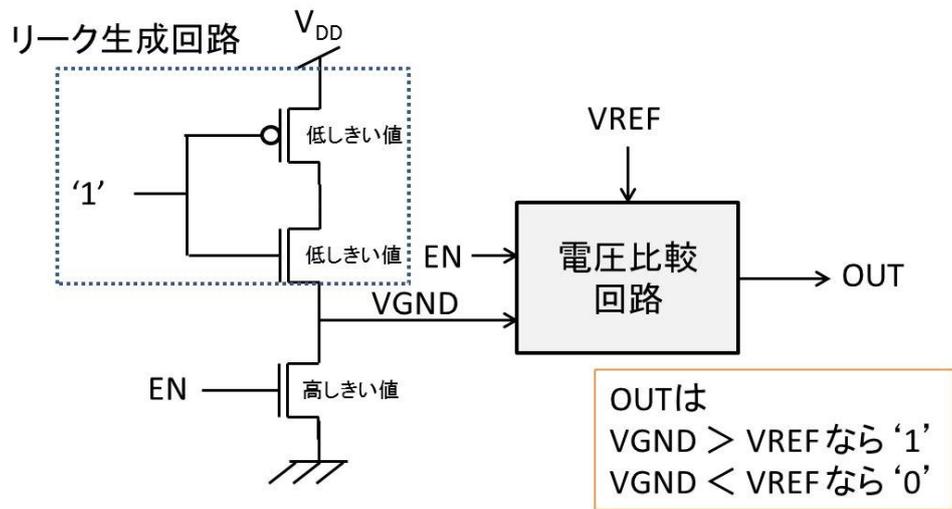
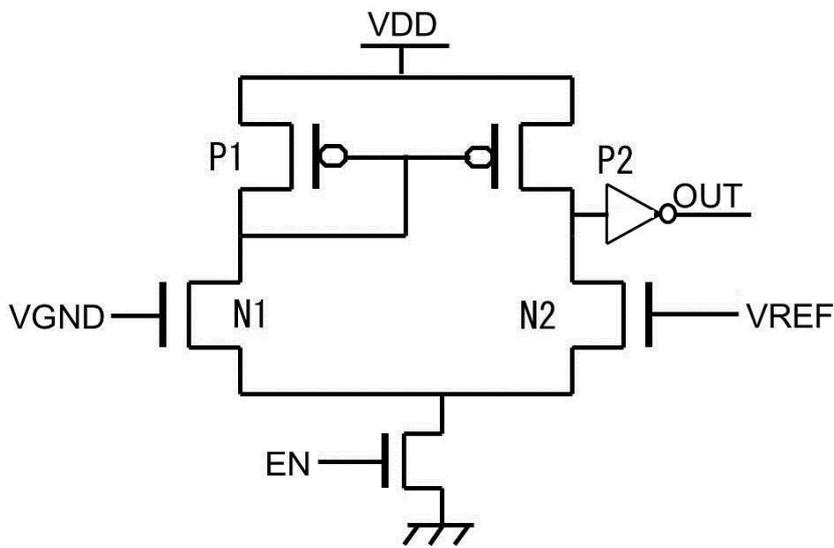
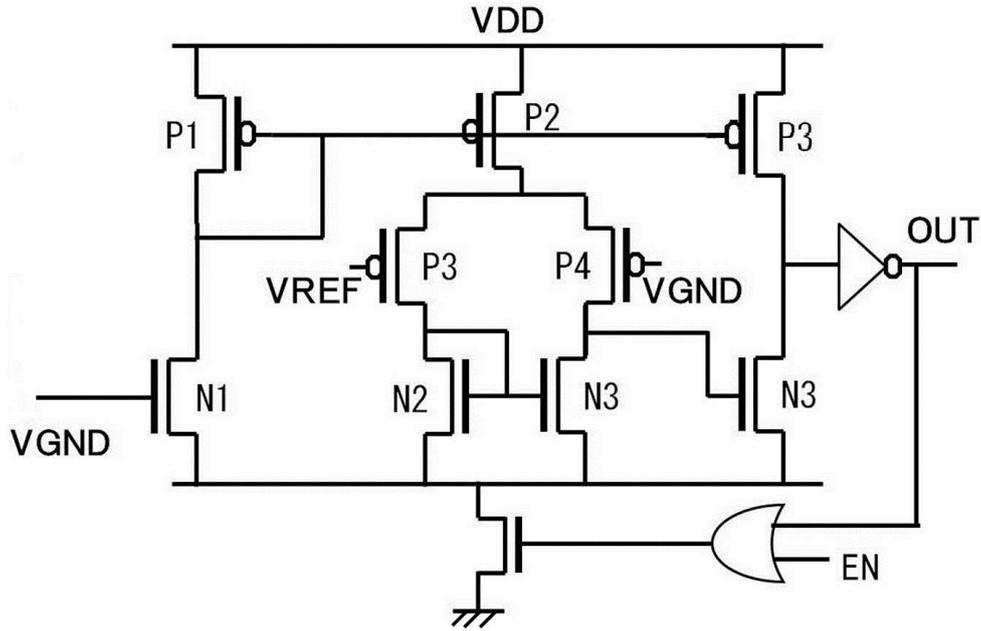


図 4.3.3.2 リークモニタの回路構成



(a) 一般的なカレントミラー型電圧比較回路



(b) 2重構造のカレントミラー型電圧比較回路

図 4.3.3.3 電圧比較回路

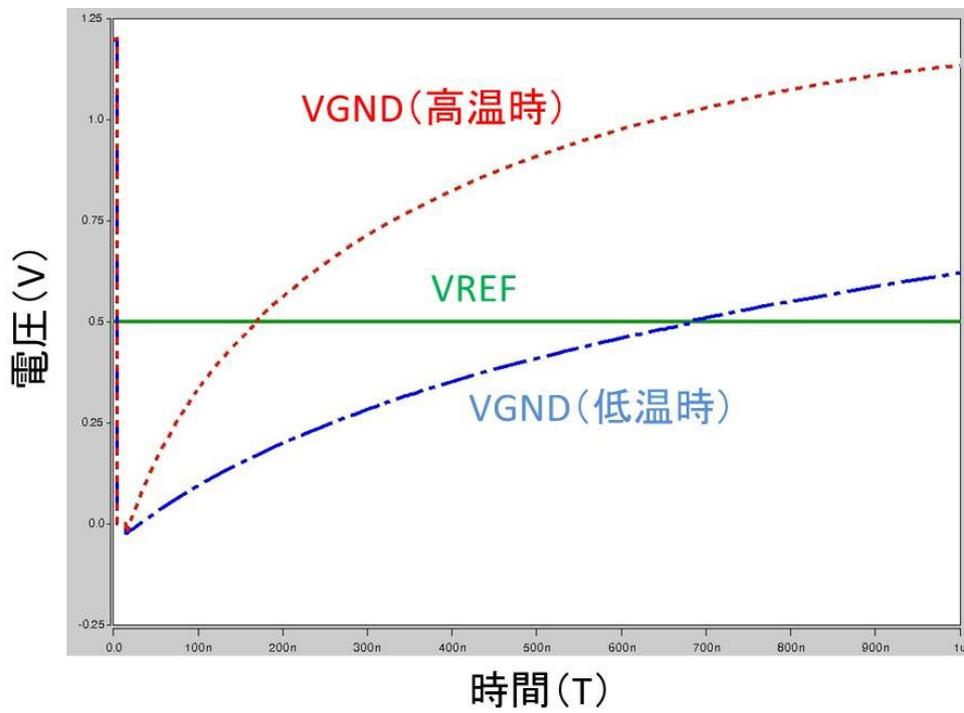


図 4.3.3.4 リークモニタ回路のシミュレーション波形

リークモニタ回路はリーク電流による信号線の電圧上昇を利用しており、電圧比較回路も使用しているため、アナログ回路となっている。そのため、回路内の容量や抵抗によってシミュレーションによる結果が変わってしまう可能性がある。また、トランジスタのばらつきによる影響も大きいと考えられる。そのため、65nm プロセスを用いてリークモニタ回路のレイアウトを行い、レイアウトデータから容量および抵抗の抽出を行った。図 4.3.5 はリークモニタ回路のレイアウト図を示している。上段が比較回路部となっており、下段がリーク生成回路部になっている。リークモニタ回路の面積は 16.4 $\mu\text{m}^2$  である。

また、このリークモニタ回路は以下の2つの特長があげられる。

- カウンタ回路に比べ消費エネルギーが少ない
- リーク電流によって出力の遅延時間が生成される

カウンタ回路が BET 分の時間を測定するために、ダイナミックエネルギーを常に消費するのに対し、リークモニタ回路は VGND に 1 回電荷を溜めるためのエネルギーと電圧比較回路が 1 度動作するエネルギーのみである。そのため、リークモニタ回路をスリープ制御に使用することで制御回路にて消費されるエネルギーを効果的に抑制することができる。また、温度によって出力の遷移時間が変化することを利用して、チップ上の温度の変化を自動で検知し、より効果的なリーク電力の削減が可能となる。次の 4.3.2 節にて、このリークモニタ回路を用いた CuD 方式の詳しい制御方法について述べる。

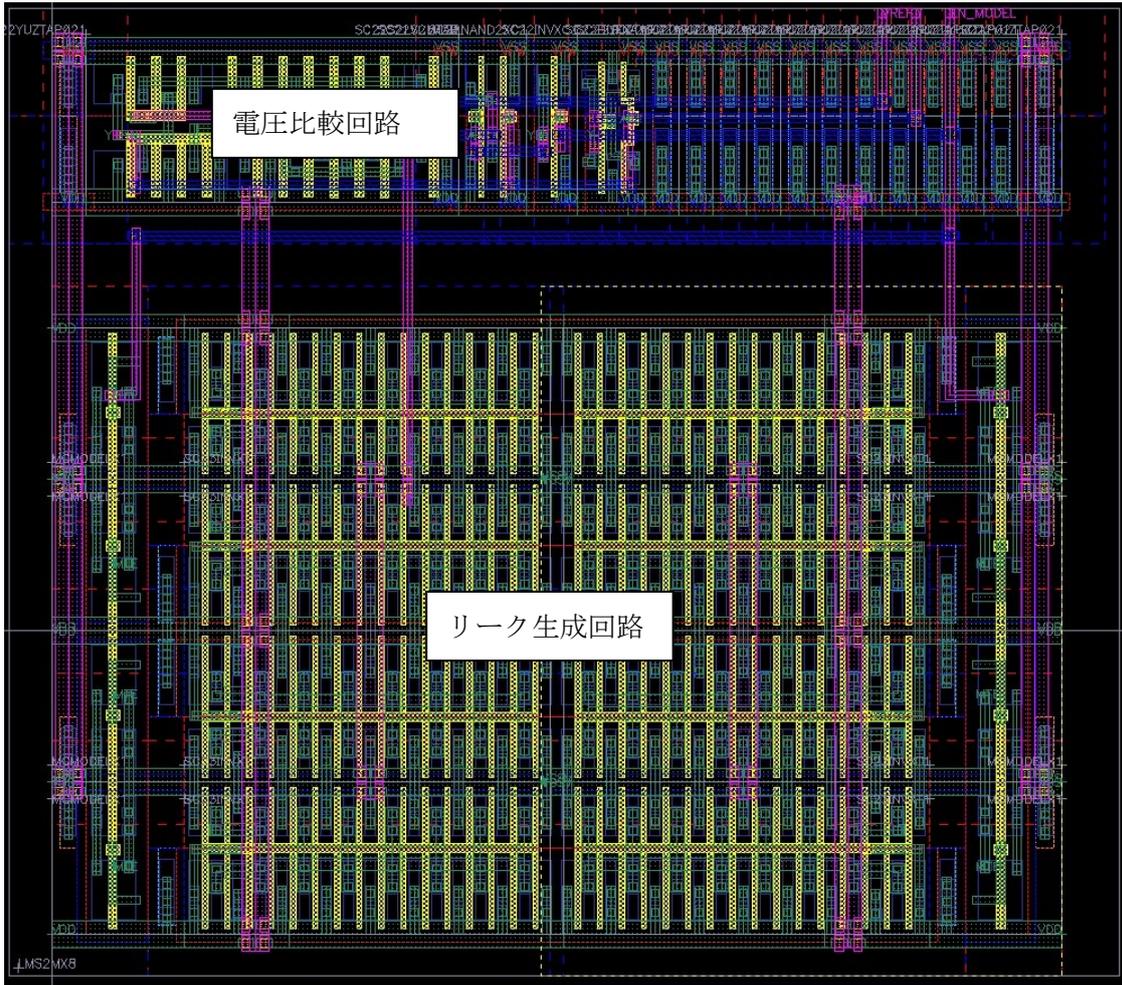
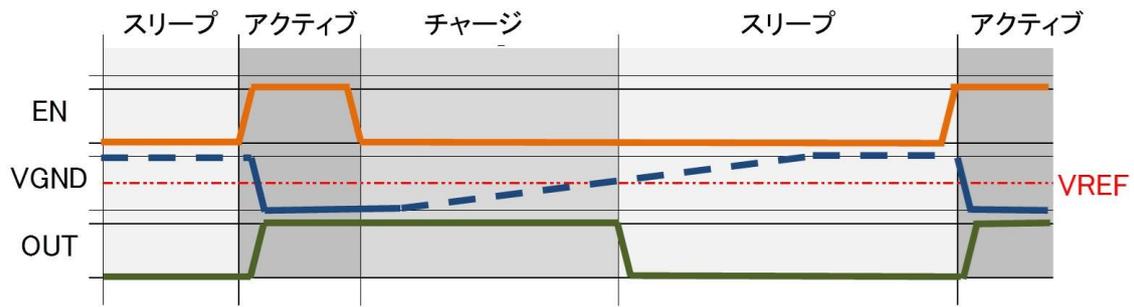


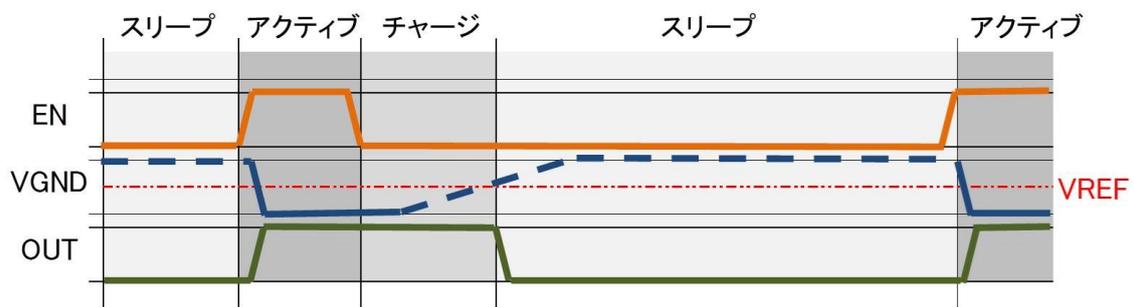
図 4.3.3.5 リークモニタ回路のレイアウト図

### 3.3.2 CuD 方式の制御方法

CuD 方式は演算器の EN 信号を受け取り、リークモニタ回路の出力の遷移時間を利用してスリープ制御を行う手法である。図 4.3.6 は CuD 方式におけるスリープ制御の制御シーケンスを示す。演算器がスリープ状態からアクティブ状態になると EN 信号が '0' から '1' に遷移し、リークモニタ回路内の高閾値の nMOS がオンになる。これにより、チップ上の温度に関係なく、VGND の電圧は素早く下がる。一方、演算器がアクティブ状態からスリープ状態になると EN 信号は '1' から '0' に遷移し、VGND に電荷が溜まり始める。この際、VGND の電圧上昇速度はリーク生成回路部のリーク電流量に依存するため、低温時ではゆっくりと電圧が上昇する。そのため、VGND の電圧が VREF に達し、出力 OUT が遷移するまでの遷移時間が長くなる。一方、高温時ではリーク生成回路部のリーク電流量が増大するため、VGND が VREF に達するまでの時間は短くなる。これにより、リークモニタ回路の出力 OUT の遷移時間は短くなる。このリークモニタ回路の出力遷移時間は、PG を適用した演算器の BET と非常に近い温度特性を示すことが知られている[41]。Geysler-3 で用いた各演算器の BET とリークモニタ回路の出力遷移時間の温度特性を図 4.3.7 に示す。今回使用するリークモニタ回路では出力遷移時間を演算器の中で最も BET が長かった ALU の 25°C の BET と同程度になるように設計した。これは、1 種類のリークモニタ回路で各演算器を制御するというもっともシンプルな CuD 方式を想定したためである。また、リークモニタ回路の出力遷移時間に ALU の 25°C の BET を想定したのは、汎用的にショートスリープでの PG をできるだけ抑えるためである。本手法では 4.3.1 節で述べたように、リークモニタ回路の pMOS のリーク電流のみを使用している。しかし、図 4.3.6 が示す通り、リークモニタ回路の遷移時間は各演算器の BET と非常に近い温度特性を示している。これにより、CuD 方式は制御回路の消費エネルギーを効果的に抑えながら、ATB 方式に非常に近い制御を実現することができる。



(a) 低温時



(b) 高温時

図 4.3.3.6 チップ上の温度変化による PG 制御タイミング

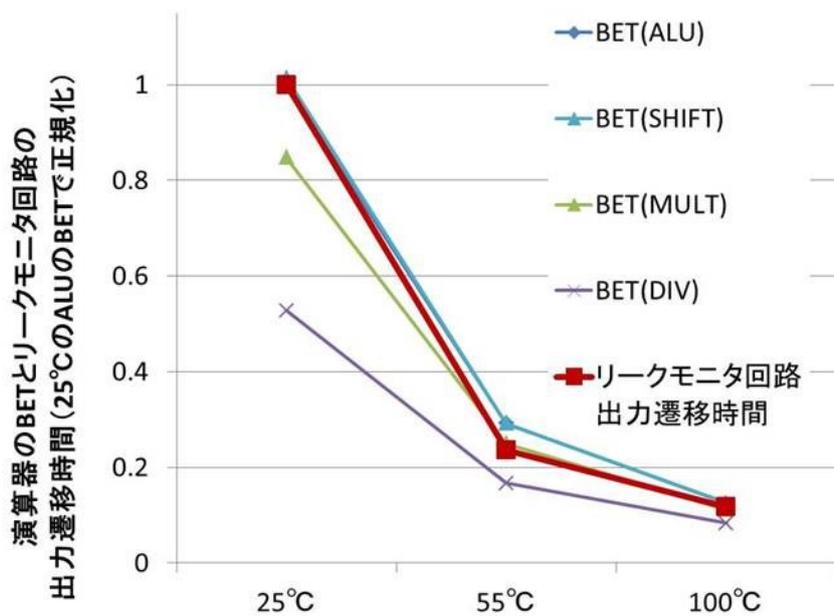


図 4.3.3.7 演算器の BET とリークモニタ回路の遅延時間の関係性

### 3.3.3 CuD 方式の利点

CuD 方式で用いるリークモニタ回路は **VGND** と **VREF** の電圧を比較し、その結果を使って直接 **PS** の **EN** 信号を制御する。この時、CuD 方式で 1 回の **PG** を制御するために必要なエネルギーは、**VGND** の電圧が **VREF** を超えるまでの電圧比較回路の 1 回動作エネルギーと **VGND** を **VREF** の電圧まで充電するためのエネルギーのみである。また、リークモニタ回路が **PS** をオフするタイミングは、**PG** 適用回路の **BET** と同程度にすることが可能である。さらに、チップ上の温度が変化した時、**PG** 適用回路の **BET** とリークモニタ回路が **PS** をオフするまでの時間は非常に近い温度特性を示す。

これにより、CuD 方式は従来手法に対し、以下の利点が挙げられる。

- スリープ制御回路の消費エネルギーを非常に小さく抑えられる
- 回路温度による **BET** の変化に応じて **PS** を自動で制御できる

以上より、CuD 方式を用いることで、チップ上の温度が低温時には **PG** とスリープ制御によるエネルギーヘッドを抑制し、高温時にはより効果的なリーク電流の削減が期待できる。

## 3.4 モデル式による評価

本章では各スリープ制御手法の消費エネルギーに対する解析モデルを導くとともに、種々の物理パラメータの各手法への影響度について議論する。

### 3.4.1 スリープ制御手法における消費エネルギー

スリープ制御手法を考慮した細粒度PGによる消費エネルギー $E_{PG}$ は以下の式で表すことができる。

$$E_{PG} = E_{OH} + E_{leak} + E_{control} \quad (\text{式 4.1})$$

$E_{OH}$ はPSのオン・オフ時に消費するエネルギーオーバーヘッドである。また $E_{leak}$ は回路のリークエネルギー、 $E_{control}$ はスリープ制御回路で消費するエネルギーを表す。さらに $E_{OH}$ は以下の式で表現できる。

$$E_{OH} = E_{PGOH} \times N \times r \quad (\text{式 4.2})$$

$E_{PGOH}$ はPGを1回行うために必要なエネルギーオーバーヘッドである。また $N$ はプログラム中の回路がPG可能になる回数（待機イベント数）、 $r$ は待機イベント数に対するPG実施率である。 $E_{PGOH}$ は主にPSをスイッチングするためのドライバやPGによって失われる回路の電荷によるエネルギーであり、

$$E_{PGOH} = C_{psd} \times VDD^2 + C_{logic} \times VDD^2 \quad (\text{式 4.3})$$

と表すことができ、スリープ時間に依存しない定数として考えることができる。 $C_{psd}$ はPSをスイッチングするためのドライバの静電容量であり、 $C_{logic}$ はPG適用回路の静電容量である。

さらに $E_{leak}$ と $E_{control}$ は、

$$E_{leak} = P_{onleak} \times T_{pson} + P_{offleak} \times T_{psoff} \quad (\text{式 4.4})$$

$$E_{\text{control}} = P_{\text{control}} \times T_{\text{idle-nonsleep}} \quad (\text{式 4.5})$$

と表すことができる。P<sub>onleak</sub> は PS がオンしている状態の回路のリーク電力、P<sub>offleak</sub> は PS がオフしている状態の回路のリーク電力、P<sub>control</sub> はスリープ制御回路で消費される電力である。また T<sub>pson</sub> はプログラム中の PS がオンしている時間であり、T<sub>psoff</sub> は PS がオフしている時間である。また T<sub>idle-nonsleep</sub> は回路が待機状態になり PG が可能になってから実際に PS がオフされるまでの時間である。この時、P<sub>offleak</sub> は P<sub>onleak</sub> に比べて非常に小さくなるため無視できるものとする。さらに、T<sub>pson</sub> および T<sub>idle-nonsleep</sub> は以下の式に表すことができる。

$$T_{\text{pson}} = \frac{1}{f_{\text{clk}}} \times n \times d + \frac{1}{f_{\text{clk}}} \times n \times (1 - d) \times s \quad (\text{式 4.6})$$

$$T_{\text{idle-nonsleep}} = \frac{1}{f_{\text{clk}}} \times n \times (1 - d) \times s \quad (\text{式 4.7})$$

n はプログラム実行にかかる総クロック数であり、d は n に対し、実際に回路が動作している割合（活性化率）を示す。また、s は回路が待機状態にある時間に対する、PS がオンしている時間の割合である。f<sub>clk</sub> は回路の動作周波数である。

### 3.4.2 各スリープ制御手法の比較

前述の節で述べた解析モデルをもとに各スリープ制御手法の消費エネルギーについて議論する。

#### (a) 各スリープ制御手法での消費エネルギー

PG を行わない手法（NonPG）では r=0、s=1 および P<sub>control</sub>=0 とみなすことができる。これにより NonPG 方式を適用した PG 適用回路の消費エネルギー（E<sub>PG(NonPG)</sub>）は以下の式になる。

$$E_{\text{PG(NonPG)}} = P_{\text{onleak}} \times \frac{1}{f_{\text{clk}}} \times n \quad (\text{式 4.8})$$

NonPG 方式ではプログラム実行時間の間、リークエネルギーは PS オン時のリーク電力に依存する。

また WIPS 手法では回路が待機状態になるとすぐに PS がオフされるため、 $r=1$ 、 $s=0$  となり、WIPS 方式を適用した PG 回路の消費エネルギー ( $E_{PG}(WIPS)$ ) は

$$E_{PG}(WIPS) = E_{PGOH} \times N + P_{onleak} \times \frac{1}{f_{clk}} \times n \times d \quad (\text{式 4.9})$$

と表すことができる。

さらに TB 方式は PS をオフするかを判断する間、カウンタ回路が動作しているため、

$$P_{control} = C_{count} \times V_{DD}^2 \times f_{clk} \quad (\text{式 4.10})$$

となる。 $C_{count}$  はカウンタ回路の静電容量である。このとき、回路全体の消費エネルギーは

$$E_{PG}(TB) = E_{PGOH} \times N \times r + P_{onleak} \left( \frac{n \times d}{f_{clk}} + \frac{n \times (1-d) \times s}{f_{clk}} \right) + C_{count} V_{DD}^2 \times n \times (1-d) \times s \quad (\text{式 4.11})$$

と表すことができる。また ATB 方式を実現するためにはオンチップ温度モニタや LUT によってさらにエネルギーが消費されてしまう。

一方、CuD 方式は VGND の電圧が VREF に達するまでの電荷とセンスアンプ回路のエネルギーが消費されるだけである。そのためスリープ制御回路で消費される電力は、

$$P_{control} = C_{VGND} \times VREF \times V_{DD} + P_{SA} \quad (\text{式 4.12})$$

となる。 $C_{VGND}$  はリークモニタ回路の VGND の配線容量であり、 $P_{SA}$  は電圧比較回路で消費される電力である。このとき、回路全体の消費エネルギーは

$E_{PG}(CuD)$

$$= E_{PGOH} \times N \times r + P_{onleak} \left( \frac{n \times d}{f_{clk}} + \frac{n \times (1 - d) \times s}{f_{clk}} \right) + \frac{(C_{VGND} \times VREF \times VDD + P_{SA}) \times n \times (1 - d) \times s}{f_{clk}}$$

(式 4.13)

と表すことができる。

## (b) 各パラメータの影響による比較

$P_{onleak}$ が増加する場合、NonPG方式の消費エネルギーは $P_{onleak}$ の増加量に比例する。一方、 $P_{onleak}$ が増大するとWIPSやTB、CuD方式はPSがオンしている間の消費エネルギーのみが増加する。この時、PSがオンしている時間はWIPS方式が最も短いため、 $P_{onleak}$ が増大していくとWIPS方式が最も消費エネルギーに優れたスリープ制御手法となる。

また活性化率 $d$ が小さい場合、NonPG方式と比べ、他のPG方式はPSがオンしている間のリークエネルギーである $E_{onleak}$ が小さくなる。しかし、 $d$ が大きくなるとPG方式の $E_{onleak}$ がNonPG方式の値に近づく。またTB方式とCuD方式を比較すると、TB方式は回路が待機状態になってからの時間がBETに達するまでの間カウントを行うため、カウント回数分のエネルギーが消費されてしまう。これに対し、CuD方式はリークモニタ回路を1回分動作させるだけの消費エネルギーで済む。

VDDが変化した場合、NonPG方式は $P_{onleak}$ が影響を受けるだけである。一方、他のPG手法では、 $E_{PGOH}$ がVDDの2乗の影響を受けるため、NonPG方式に比べパラメータへの影響が大きいと考えられる。さらにTB方式では $E_{control}$ もVDDの2乗で影響を受け、CuD方式でも1乗の影響を受ける。

動作周波数 $f_{clk}$ に対しては、NonPG方式は反比例の影響を受ける。そのため、動作周波数が高い場合にはNonPG方式がもっとも優れた手法になるが、動作周波数が低くなるほどに他の手法に比べ消費エネルギーの削減効果が低くなる。

## 4.1 評価

各スリープ制御手法による消費エネルギー削減効果を評価するため、4.1 節で述べた 65nm プロセスを用いたマイクロプロセッサである Geyser-3 を対象回路としシミュレーションでの評価を行った。また、nMOS を用いた FTPG 型の細粒度 PG を適用した ALU、シフト回路 (SHIFT)、乗算器回路 (MULT)、除算器回路 (DIV) の 4 つの演算器を持っている。標準の電源電圧は 1.2V とした。また、CuD 方式のための VREF には 0.5V を供給している。Geyser-3 全体のリーク電力の割合を図 4.5.1 に示す。ただし、キャッシュの消費電力は含まれていないものとしている。4 つの演算器のリーク電力は Geyser-3 全体の 60% を占める結果となった。

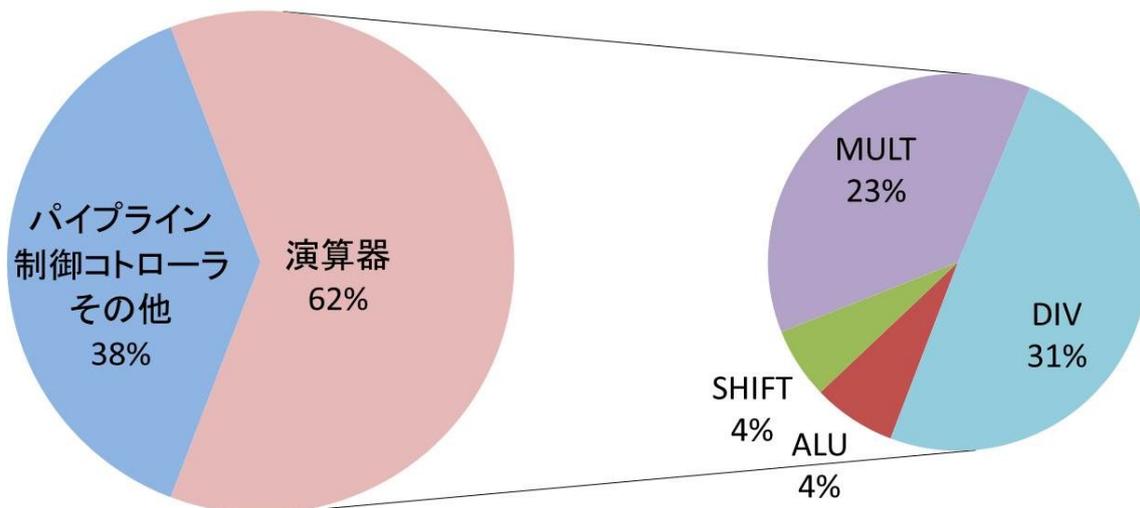


図 4.1.1 Geyser-3 におけるリークエネルギー割合

本論文では、ランタイムパワーゲーティングを用いてアプリケーション実行中の演算器のリークエネルギーおよびスリープ制御回路の消費エネルギーの評価を行う。しかし、Geyser-3 全体でアプリケーションを動かそうとする場合、例えば、100 個の要素に対するクイックソート・プログラムでは、実行に要するクロックサイクル数は 188,569 サイクルにも及ぶ[40]。これだけの膨大なクロックサイクル数を、マイクロプロセッサ全体の SPICE 記述に対して、回路レベルのシミュレーションを行うことは、回路規模およびシミュレーション時間の点から事実上不可能である。RTL レベルの Verilog シミュレーションでは上

記クロックサイクルのシミュレーションは可能であるが、正確な消費エネルギーの評価ができない。そこで、以下に示す方法を考案し、消費エネルギーを算出した。

まず、Geysler-3 全体の RTL モデルの Verilog シミュレーションを行うことでプログラム実行時に演算器が待機状態になる時間とイベント回数を解析し、演算器の各待機時間がプログラム実行中に何度発生しているかを求めた (図 4.5.2)。また、回路シミュレータ Hsim を用いて演算器の各待機時間に PG を実行した場合の消費エネルギーを求めた (図 4.5.3)。図 4.3.4 に示すように、演算器の待機時間ごとのイベント回数と待機時間ごとの消費エネルギーを掛け合わせて総和を取ることで、アプリケーション実行時の各演算器の消費エネルギーを算出した。さらに、演算器回路とスリープ制御回路の電源を分離して電流を解析することにより、両者それぞれの消費エネルギーを評価した。評価を行うアプリケーション [46] としてクイックソート (Qsort)、離散コサイン変換 (DCT)、JPEG エンコード (JPEG) の 3 つを用いた。

#### あるプログラムにおけるスリープイベント (Shift@25 °C)

サイクル数	state@PG	"Short sleep" or "Long sleep"
5	Active	
1288	Sleep	Long sleep
2	Active	
81	Sleep	Short sleep
1	Active	
2	Sleep	Short sleep
1	Active	
179	Sleep	Long sleep
82	Active	
15	Sleep	Short sleep
6	Active	
1	Sleep	Short sleep
4	Active	
⋮	⋮	⋮



プログラム実行の流れ

BET=146

- ・BET以上 → Long sleep
- ・BET以下 → Short sleep

図 4.1.2 各プログラムにおける演算器の Verilog シミュレーション結果

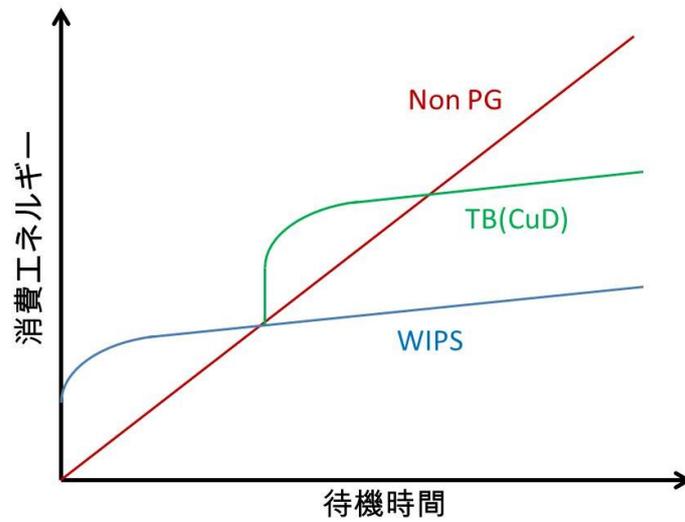


図 4.1.3 各スリープ制御手法における演算器の消費エネルギー

各アプリケーションの 待機イベント数		待機時間ごとの 演算器の消費エネルギー	
待機時間 (クロック数)	待機イベント数	待機時間 (クロック数)	演算器の消費 エネルギー(μJ)
1	240	1	0.9
2	384	2	1
4	96	4	1
12	48	12	1.4
25	48	25	1.7
33	48	33	1.9
38	48	38	2
40	84	40	2
81	48	81	2.9
82	48	82	2.9
82	96	82	3
200	150	200	4.4

×
=
アプリケーションごとの  
演算器の消費エネルギー

図 4.1.4 各アプリケーションの消費エネルギーの算出方法

図 4.5.5 には上記で述べた方法で算出した各演算器のリークエネルギーとスリープ制御回路の消費エネルギーのシミュレーション結果を示す。シミュレーション結果はアプリケーション毎に各温度での NonPG の消費エネルギーを元に正規化している。また、ATB 方式の結果にはオンチップ温度モニタ回路や LUT 回路の消費エネルギーは含まれておらず、カウンタ回路と比較回路のみとしている。また、表 4.5.1 には各プログラム実行時に、演算器で起こるスリープイベントに対し、BET よりも PS をオフしている時間が長いロングスリープと、BET よりも PS をオフしている時間が短いショートスリープの発生頻度を示している。

これらの結果より、各スリープ制御手法によるリークエネルギー削減効果には演算器ごとに以下のような特長がでることが分かった。

#### ● ALU

ALU では各アプリケーション、温度において NonPG が最も消費エネルギーが小さくなった。これは表 4.5.1 で示されているように、ALU では頻繁に回路が使用されるため、スリープイベントのうちロングスリープになるイベントは 1 度も無い。そのため、PG を実行してしまうとエネルギーは増大してしまうだけである。WIPS 方式はすべてのスリープイベントに対し PG を有効にしてしまうため、すべての条件において消費エネルギーを増大させてしまっている。これにより、チップ温度が低温である 25°C では、4.1 倍から 5 倍までエネルギーが増えてしまっている。一方、TB、ATB、CuD 方式では、PG が有効になる前に、ALU が再び使用されるため、PG によるエネルギーの増加を抑制できている。しかし、TB、ATB 方式では、スリープ制御回路で消費されるエネルギーが大きいため、25°C のような低温時では最大で 94% も消費エネルギーが増大してしまっている。一方で CuD 方式は、低温時でも NonPG に比べ 10% 以下に消費エネルギーが増加するにとどまっている。これは図 4.5.6 に示すように、PS をオフするまでの CuD 方式の消費エネルギーが、TB 方式に比べ非常に小さくなるためである。

#### ● SHIFT

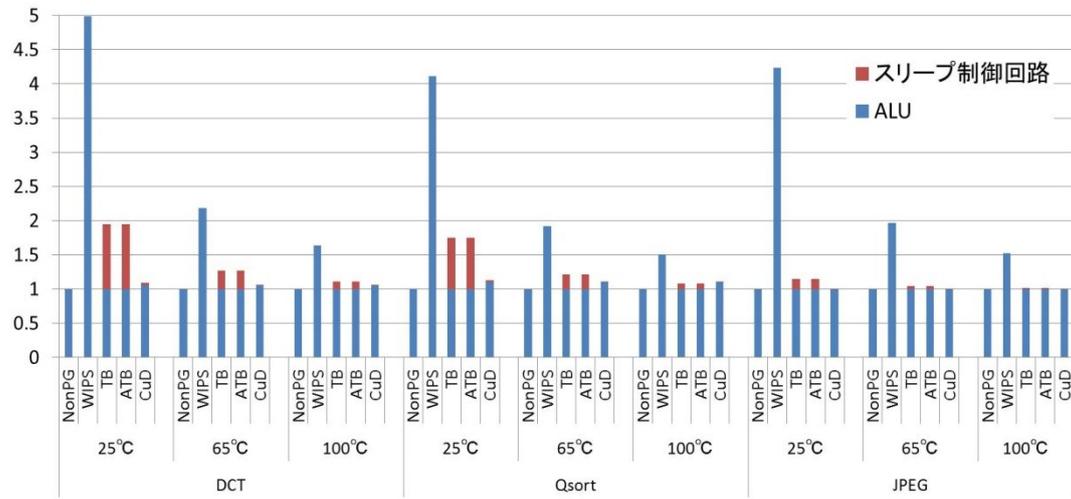
SHIFT では温度やアプリケーションによって各スリープ制御手法の優劣が変わってくる。25°C では、NonPG または CuD 方式が最も消費エネルギーが小さい結果となっている。特に CuD 方式は、DCT 実行時に最大で 5% 増加しただけである。これは、チップ上の温度が低いため、演算器回路のリークエネルギーが少なく、PG によるエネルギーオーバーヘッドやスリープ制御回路の消費エネルギーの影響が大きくなっているためである。そのため、WIPS 方式では、DCT と JPEG 実行時に 212%~250% まで消費エネルギーが増加している。しかし、Qsort 実行時には 50% に消費エネルギーを削減できている。また TB、ATB 方式では各アプリケーションで、184%~229% も消費エネルギーが増加している。一方、チップ温度が高い 100°C の場合では、NonPG に対し、各スリープ制御手法で消費エネルギーを削減することができている。特に WIPS 方式が消費エネルギーを小さくすることができ、最大で 12% にまで消費エネルギーが削減できた。

- MULT

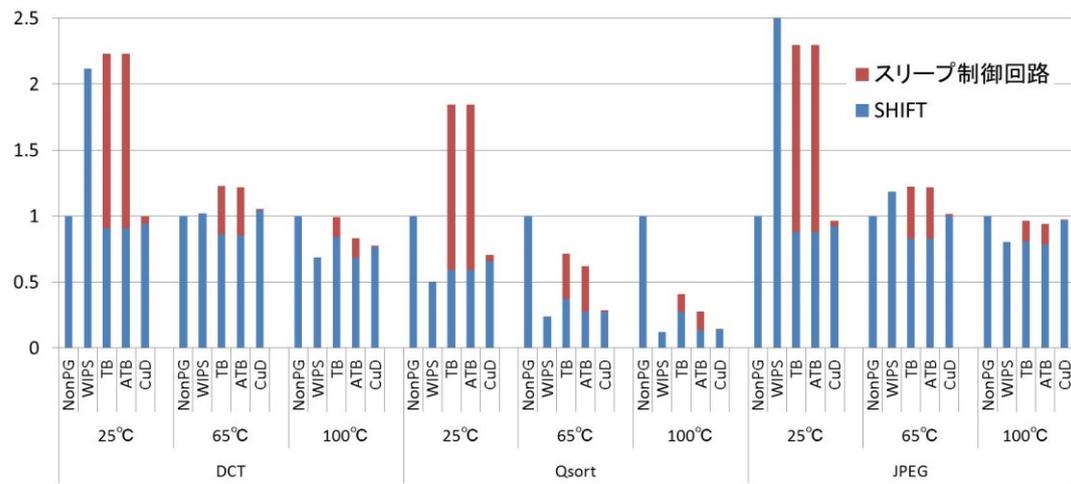
SHIFTと同様に、MULTでもチップ上の温度やアプリケーションによってスリープ制御手法の優劣が変わることが分かった。特に、Qsort、JPEG実行時では、温度に関係なく各スリープ制御手法で高いエネルギーの削減効果が得られた。CuD方式では58%~10%にまで消費エネルギーが削減できている。しかし、DCTを実行する場合は、チップ上の温度が65°Cの条件でも、WIPS方式では消費エネルギーの増大に繋がるということが分かった。また、TBやATB方式に比べ、CuD方式はスリープ制御回路の消費エネルギーが小さいため、低温時にはエネルギーの増加を抑え、高温時には高いエネルギー削減効果を実現することができた。

- DIV

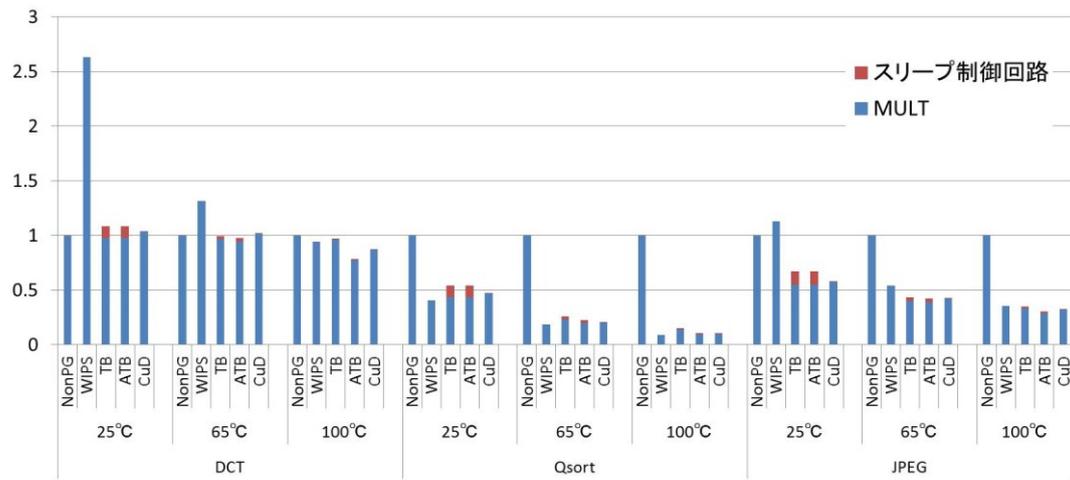
一方、DIVでは、温度やアプリケーションに依らず、各スリープ制御手法で高いエネルギー削減効果が得られた。これは、アプリケーションの種類に関係なくDIVが使用される頻度が極めて低いため、表4.5.1に示すように、ロングスリープの割合が他の演算器に比べ非常に高くなったからだと考えられる。そのため、DIVでは温度やアプリケーションに依らず、WIPS方式が最もエネルギーを小さくすることができ、最大で4%にまで消費エネルギーが削減できている。さらに、25°Cでも、32%~29%にまで消費エネルギーを削減できた。また、CuD方式も34%~4%までの範囲で消費エネルギーを削減できている。



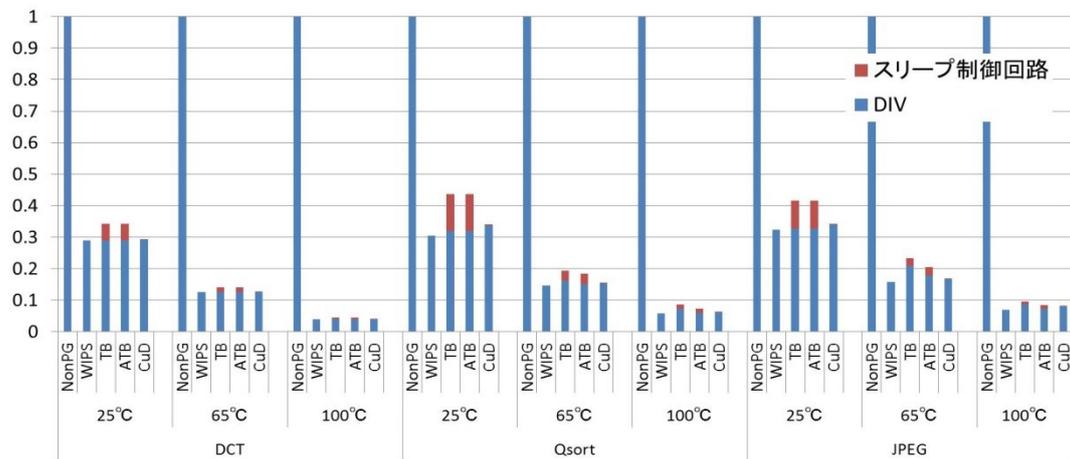
(c) ALU



(d) SHIFT



(e) MULT



(f) DIV

図 4.1.5 各演算器の消費エネルギー

表 4.1.1 各アプリケーションのスリープイベント解析

	温度	スリープ時間	ALU	SHIFT	MULT	DIV
	DCT	25°C	ショートスリープ	5532	1437	2538
ロングスリープ			0	7	7	1
65°C		ショートスリープ	5532	1236	2352	0
		ロングスリープ	0	208	193	1
100°C		ショートスリープ	5532	1008	2304	0
		ロングスリープ	0	436	241	1
	温度	スリープ時間	ALU	SHIFT	MULT	DIV
	Qsort	25°C	ショートスリープ	18937	24	23
ロングスリープ			0	211	86	41
65°C		ショートスリープ	18937	24	23	0
		ロングスリープ	0	211	86	41
100°C		ショートスリープ	18937	21	23	0
		ロングスリープ	0	214	86	41
	温度	スリープ時間	ALU	SHIFT	MULT	DIV
	JPEG	25°C	ショートスリープ	305647	124310	59760
ロングスリープ			0	157	194	661
65°C		ショートスリープ	305647	119009	56448	2.88
		ロングスリープ	0	5458	3506	766
100°C		ショートスリープ	305647	80491	55296	0
		ロングスリープ	0	43976	4658	1854

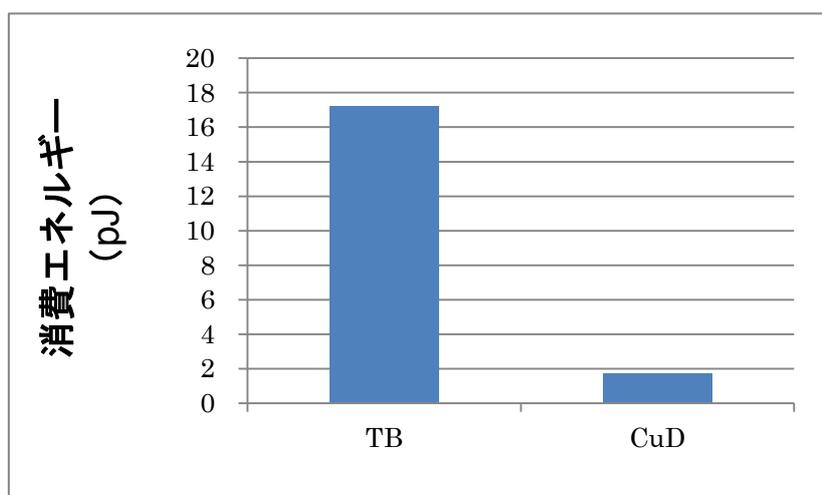


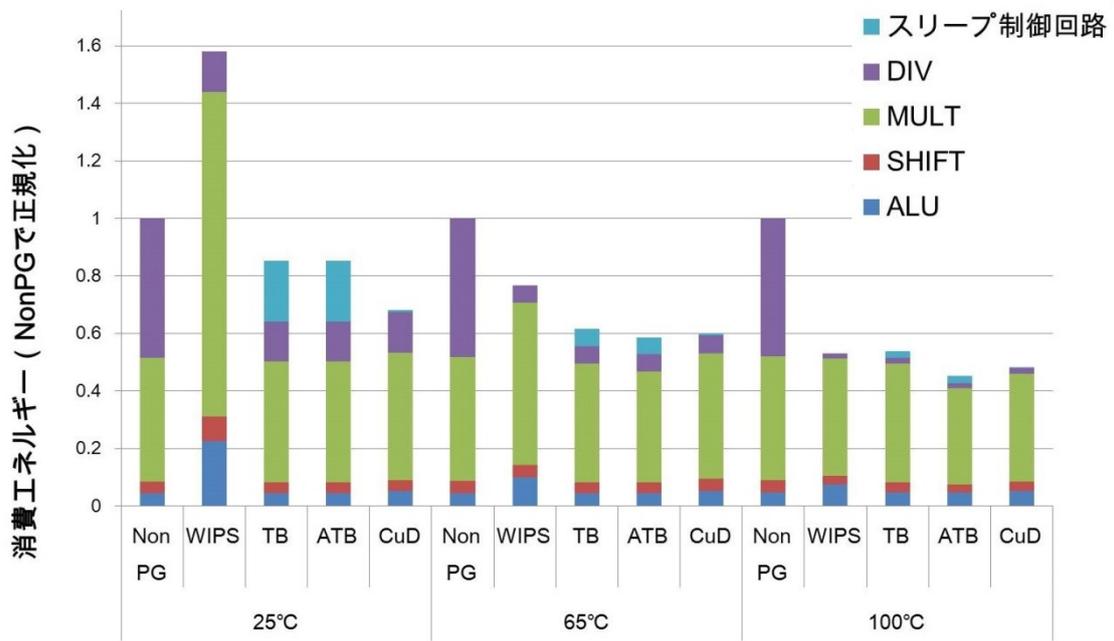
図 4.1.6 スリープ制御回路の消費エネルギー比較

また、図 4.5.5 で示したシミュレーション結果をもとに、Qsort、DCT、JPEG をそれぞれ実行したときのスリープ制御手法による各演算器のリークエネルギーとスリープ制御回路の消費エネルギーの合計を示す (図 4.5.7)。それぞれの消費エネルギーは NonPG 手法の消費エネルギーをもとに正規化している。

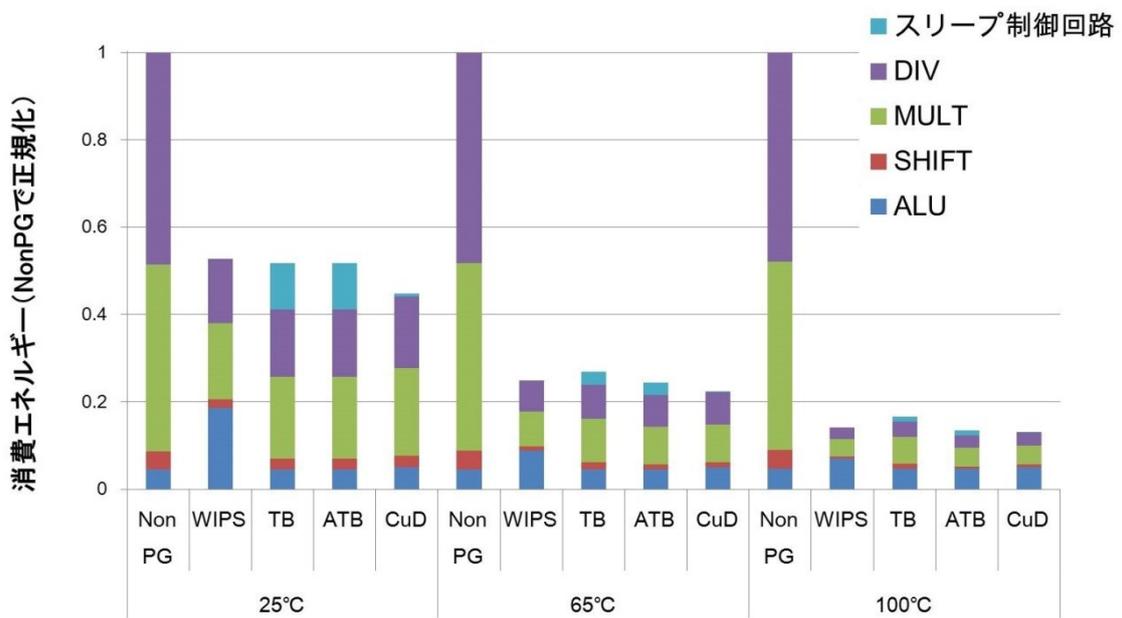
図 4.5.7 より、4つの演算器に各スリープ制御手法を適用した場合、各アプリケーションの消費エネルギーは以下ようになった。

- Qsort、JPEG 実行時では、25°C~100°Cの温度で CuD 方式が最も消費エネルギーを削減でき、最大で 13%にまで削減できた
- DCT 実行時では、25°Cの場合に CuD 方式が最も消費エネルギーを抑え、68%にまで削減できた
- 65°C、100°Cでは、ATB 方式が最も DCT の消費エネルギーを削減でき、最大で 45%にまで削減できた
- TB、ATB 方式のスリープ制御回路の消費エネルギーは最大で 21%を占める
- CuD 方式のスリープ制御回路の消費エネルギーは最大でも 2%に抑えられた

また、今回 ATB 方式は温度モニタや LUT をオンチップ上に搭載し、自動制御することが困難なため、外部から制御を行う想定をして評価を行った。この際、温度モニタや LUT を実装する際に必要な消費エネルギーが増大してしまう場合では、DCT の 65°Cや 100°Cにおいても、CuD 方式が最も消費エネルギーを削減できる可能性がある。



(a) DCT



(b) Qsort

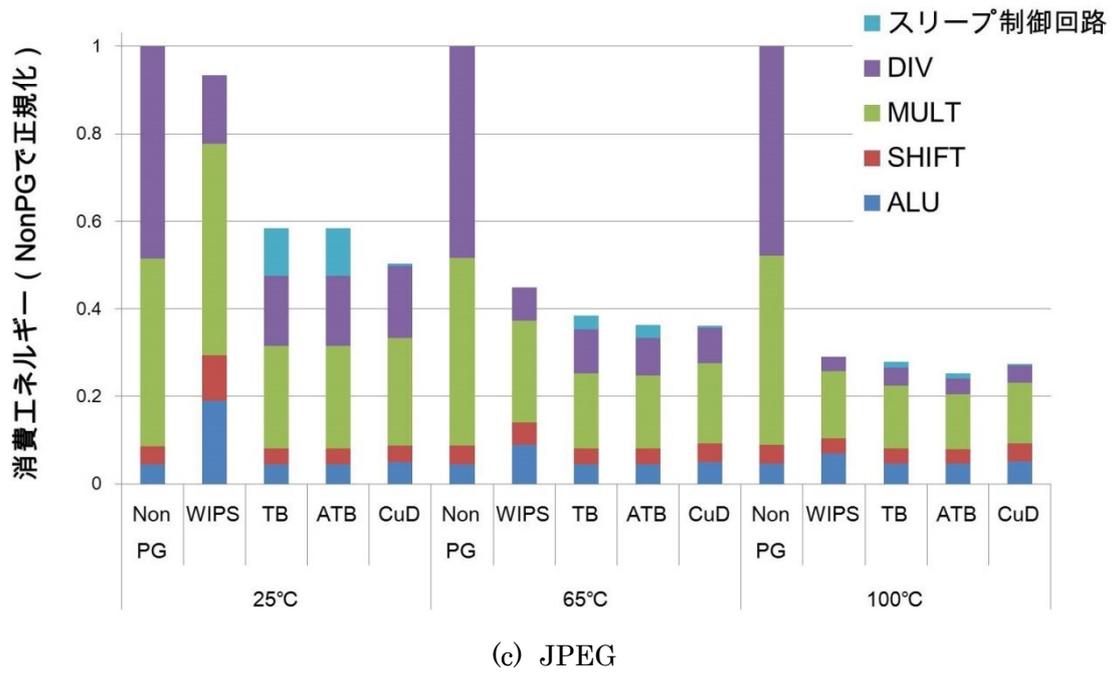


図 4.1.7 アプリケーション毎の各演算器とスリープ制御回路の消費エネルギーの割合

さらにマイクロプロセッサの演算器の消費エネルギーの分析結果から、リーク電流量、活性化率、VREF 電圧、電源電圧、動作周波数の観点でどのスリープ制御手法が優位性を示すか評価した (図 4.5.8)。図 4.5.8(a)は SHIFT と MULT の 25°C、65°C、100°Cでの温度におけるリーク電流量とプログラム実行時の消費エネルギーの関係を示している。また横軸のリーク電流量は SHIFT の 25°Cのリーク電流量で正規化している。温度が 100°Cの MULT のリーク電流は 25°Cの SHIFT の約 100 倍となっている。リーク電流量が少ない 25°Cの SHIFT の場合、WIPS、TB、ATB 方式の消費エネルギーが NonPG よりも増大している。WIPS 方式に関してはリーク電流量が小さいために、リークエネルギーの削減効果に対し、PG のオーバーヘッドが大きくなったためである。また、TB、ATB 方式は CuD 方式と同様にショートスリープ時の PG の実行を避けられるが、スリープ制御回路の消費エネルギーがリークエネルギーに対して大きくなってしまふ。一方、CuD 方式はスリープ制御回路の消費エネルギーを大幅に抑えながら、ショートスリープによるエネルギーの増大を招いていないため、リーク電流が少ない条件でもエネルギーの増大を起こしていない。また、MULT の 25°Cではエネルギーの増大を起こしているのは WIPS 方式のみである。これは SHIFT に比べ、MULT のリーク電流が多いが、回路面積も同様に大きくなっているため、演算器の静電容量も増大し、PG によるオーバーヘッドも大きくなってしまふ。そのため WIPS 方式は SHIFT と同じくエネルギーの増大につながってしまっている。一方、TB、ATB 方式に必要なカウンタ回路は PG 適用回路である演算器の面積に依存しないため、リーク電流に対し、スリープ制御回路で消費されるエネルギーが小さくなっている。さらに、SHIFT、MULT とともに 100°Cの高温時には、どのスリープ制御手法も NonPG 方式よりも消費エネルギーを削減できている。これは回路規模に対し、リーク電流が多くなることにより、PG のオーバーヘッドの影響が小さくなったためである。

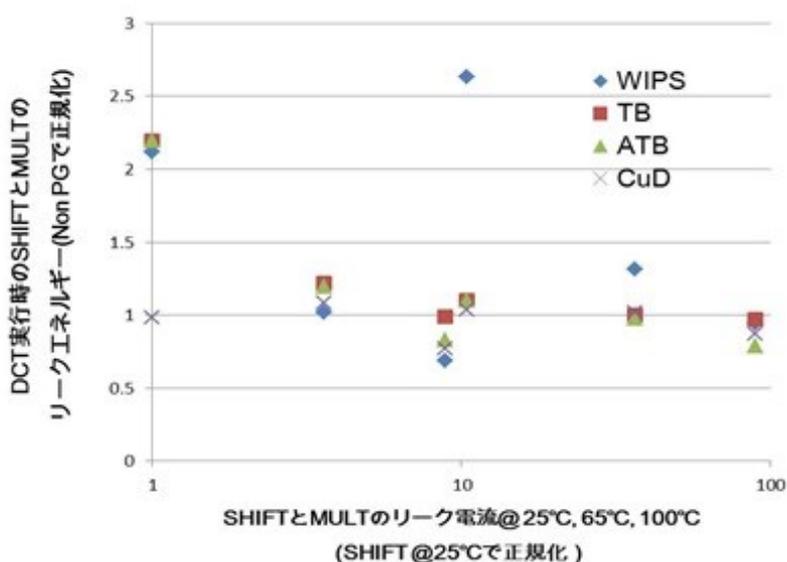
次に 4 つの演算器回路の各アプリケーションによる活性化率と消費エネルギーの関係を図 4.5.8(b)に示す。活性化率が 80%~90 %と高い回路では WIPS 方式は大幅なエネルギーの増大を起こしている。これは回路が頻繁に動作しており、PG が可能になってもすぐにまた使用されているためである。また TB、ATB、CuD 方式は高い活性化率の条件では WIPS 方式に比べエネルギーの増大を防いでいることが分かる。特に CuD 方式はスリープ制御回路で消費されるエネルギーが他の 2 つの手法よりも少ないため、消費エネルギーが小さい。また活性化率が 40%以下の回路では CuD 方式は安定して消費エネルギーを削減することができている。一方、CuD 方式以外のスリープ制御手法では NonPG 方式と比べ、消費エネルギーの削減効果が安定していない。これは活性化率のほかに、ショートスリープが起こる頻度や、スリープ制御回路の消費エネルギーと PG 適用回路のリーク電流の大小関係などの影響が出ているためだと考えられる。

図 4.5.7(c)は電源電圧が 1.2V の場合における VREF の電圧変化による消費エネルギーの影響について示している。また動作周波数は 200MHz としている。CuD 方式以外のスリー

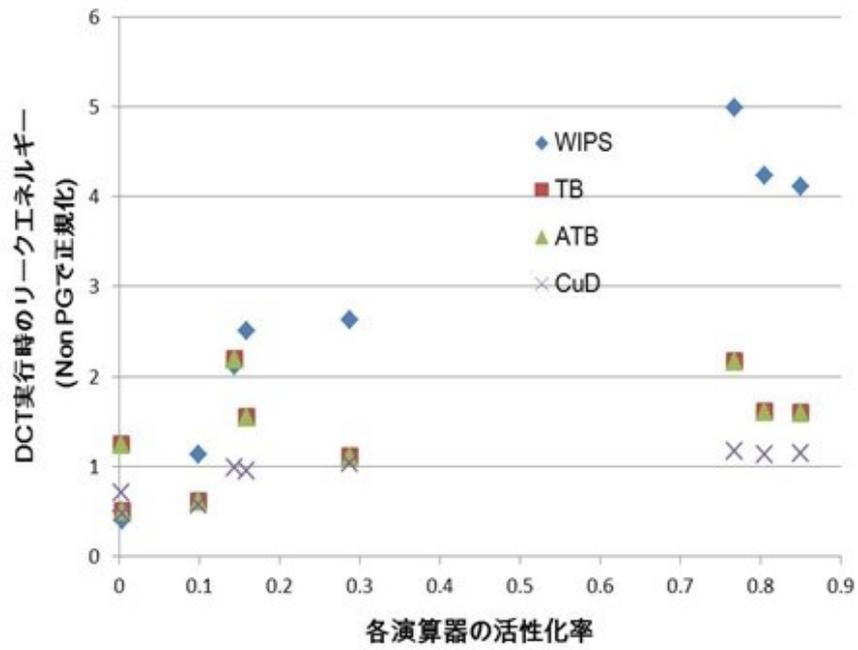
プ制御手法は VREF を用いていないため一定である。CuD 方式では VREF の電圧が変化することによってスリープ制御に必要な消費エネルギーが変化するとともに、スリープするタイミングが変わってしまう。しかし、今回想定した VREF の電圧 0.5V が 0.3V に変化しても DCT 実行時の消費エネルギーは全体の 10%程度しか増加しないことが分かった。

図 4.5.8(d)には電源電圧の影響について示す。電源電圧が上昇するとリーク電流が上昇する。また PG のオーバーヘッドも大きくなってしまう。しかし、PG 一回ごとのオーバーヘッドの増加よりも、常に消費されるリークエネルギーの増加により影響が大きいいため、電源電圧が上昇するに伴い、PG によるエネルギーの削減効果が高くなったと考えられる。

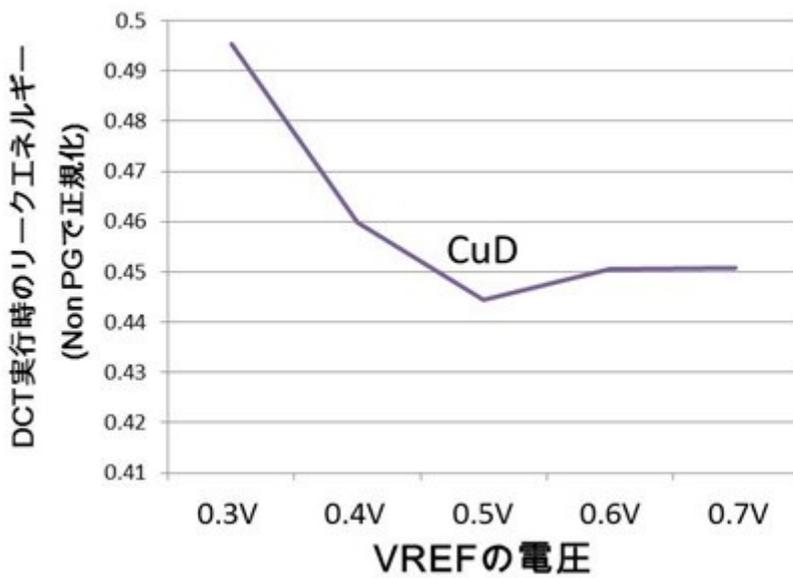
最後に図 4.5.8(e)に動作周波数と消費エネルギーの関係性について示す。電源電圧は 1.2V、VREF の電圧は 0.5V としている。今回評価に用いたマイクロプロセッサが 200MHz を最大動作周波数として設計しているため、200MHz から動作周波数を下げたときの影響を評価した。動作周波数が低くなるとアプリケーション実行中に PG を実行してられる時間が長くなる。一方、アプリケーション実行中に PG を行う回数は変わらないため、PG によるエネルギーオーバーヘッドは変化しない。これにより、ショートスリープが発生する頻度が低下するため、どのスリープ制御手法でも動作周波数が低くなるにつれて PG による消費エネルギー削減効果は高くなった。



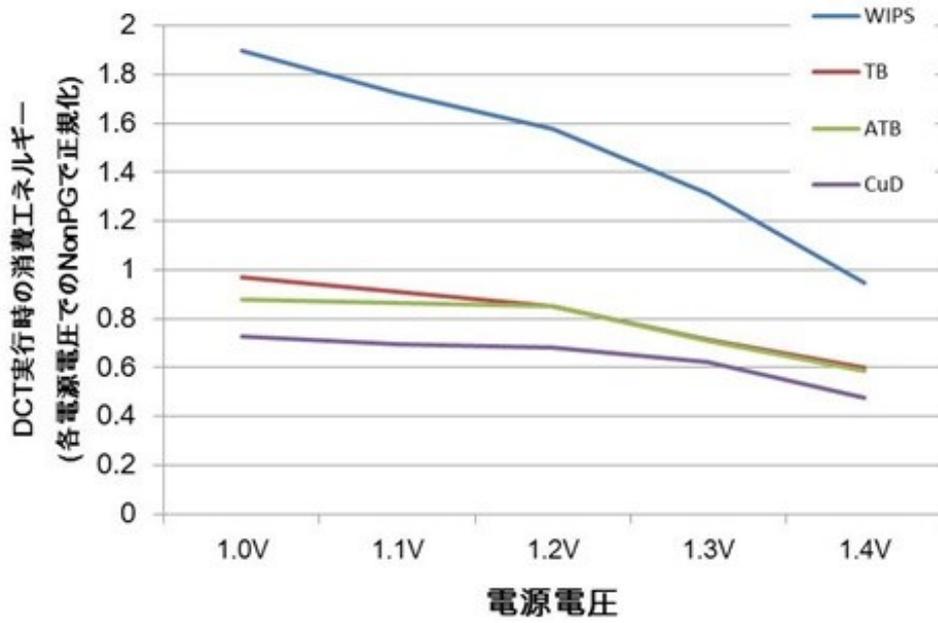
(a) リーク電流量



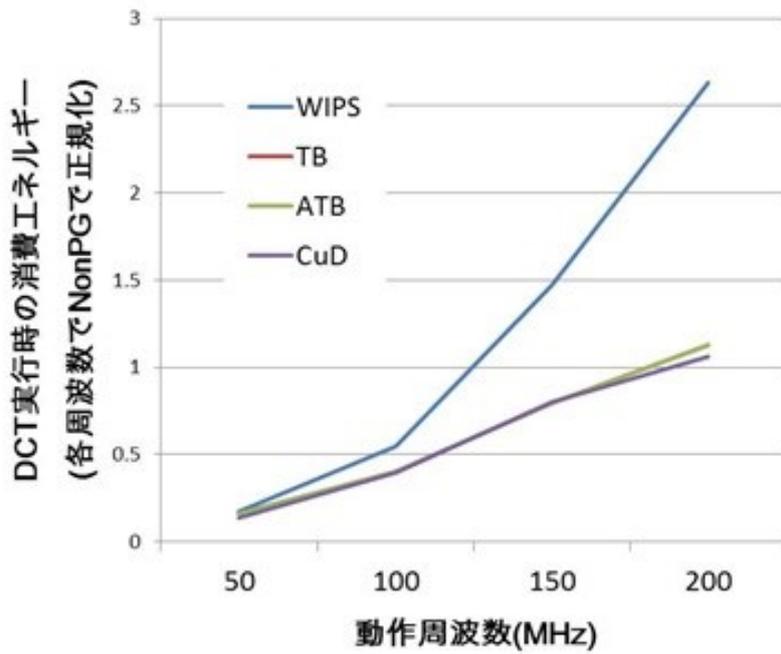
(b) 活性化率



(c) VREF



(d) 電源電圧



(e) 動作周波数

図 4.1.8 回路環境による影響

CuD 方式はリークモニタ回路の遷移時間で PG 適用回路の PS の制御タイミングを生成している。そのため、プロセスばらつきによるリークモニタの出力遷移時間の影響とそれに伴う、CuD 方式の消費エネルギーで削減効果の影響について評価を行った。

プロセスばらつきはチップ全体が同じ傾向にばらつくグローバルばらつきと、トランジスタの距離に関係なくランダムにトランジスタの特性が変化するランダムばらつきに分類することができる。

図 4.5.9 は、ばらつきなしの場合 (Typical) に対し、グローバルばらつきによってチップ全体のしきい値電圧 ( $V_{th}$ ) が nMOS、pMOS がともに  $\pm 10\%$  同じ方向に変化した場合の各スリープ制御手法の消費エネルギーを示す。TB 方式と ATB 方式は  $25^{\circ}\text{C}$  の場合同じ制御を行うため、消費エネルギーは等しい。 $V_{th}$  が  $-10\%$  になると PG 適用回路のリーク電流が増加するため、NonPG と比べて各スリープ制御手法でのエネルギー削減効果が高くなる。一方、 $V_{th}$  が  $+10\%$  に増加するとリーク電流は減少する。そのため、ショートスリープによるエネルギーの増大が顕著になる。また、TB 方式、ATB 方式が温度ごとに定めた BET に対して制御を行うのに対し、CuD 方式はリークモニタ回路の出力遷移時間でスリープ制御を行う。この際、グローバルばらつきは PG 適用回路とリークモニタ回路両方に影響を及ぼす。図 4.5.10 に示すように、グローバルばらつきによる演算器の BET の変化とリークモニタ回路の出力遷移時間は似た特性を示すため、CuD 方式ではグローバルばらつきによる影響を考慮した制御が可能になる。これにより、他のスリープ制御手法に比べ、 $V_{th}$  が増加したときのエネルギーの増大を抑制できたのだと考えられる。

また、グローバルばらつきでは nMOS と pMOS のしきい値電圧が別方向に変化する可能性がある。しかし、リークモニタ回路は pMOS のリーク電流のみを利用しているため、nMOS と pMOS が別々に変化した場合、CuD 方式ではそのばらつきに対応することが不可能である。しかし、nMOS、pMOS それぞれのリーク電流をモニタするためには、制御回路がより複雑な回路になってしまい消費エネルギーや面積のオーバーヘッドが増加してしまう。そのため、CuD 方式では、チップ製造後に VREF の電圧を変化させることで、CuD 方式の制御タイミングの最適化を行うことを想定している。

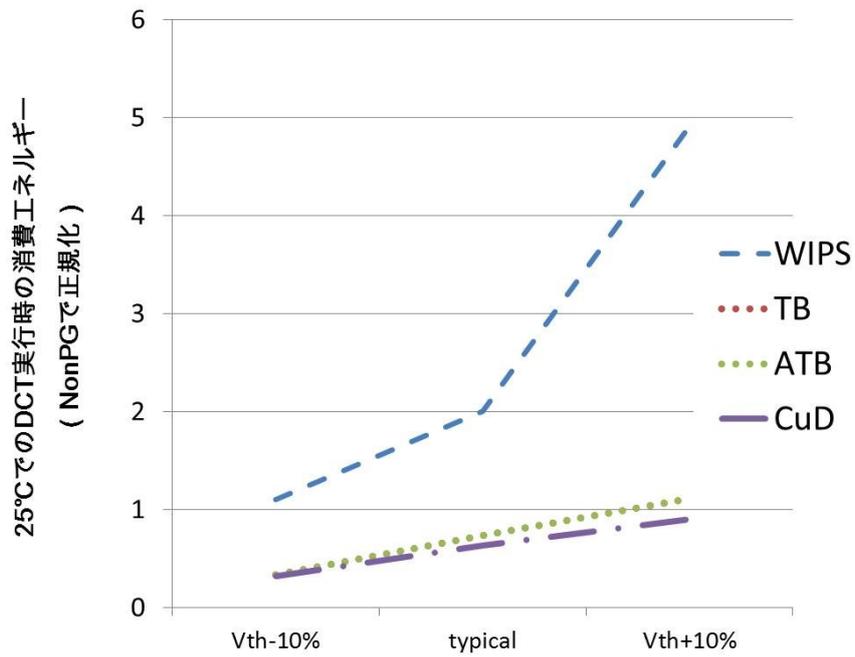


図 4.1.9 グローバルばらつきによる各スリープ制御手法への影響

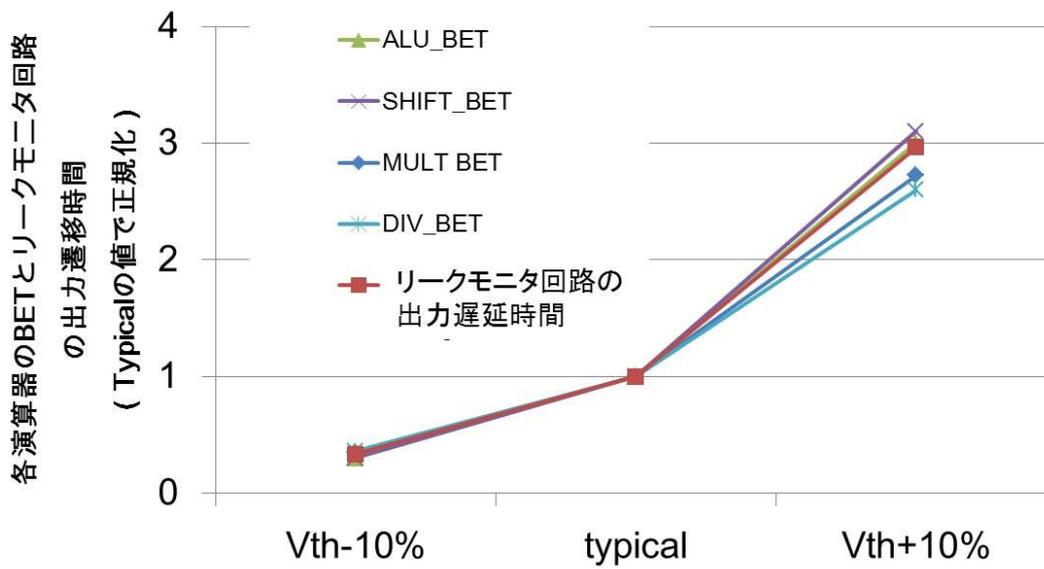


図 4.1.10 グローバルばらつきによる演算器の BET とリークモニタ回路の出力遷移時間への影響

トランジスタのしきい値電圧は正規分布でばらつくことが知られている[15]。シミュレーション条件として、標準偏差 $\sigma$ が平均のしきい値電圧より 3.3%変動するに相当する正規分布を想定した。図 4.5.11 は上記の条件で 3000 回のモンテカルロ・シミュレーションを行った際のリークモニタ回路の出力遷移時間のばらつきを示す。ばらつきなし (Typical) と比べて、遅延時間が-28%から 22%の範囲で変化した。また図 4.5.12 にリークモニタ回路の出力遷移時間が-28%から 22%まで変化した場合の CuD 方式および他のスリープ制御手法への消費エネルギーの影響を示す。TB 方式、ATB 方式では PS をオフするタイミングはカウンタ回路を用いて測定しているため、ランダムばらつきに関係なく一定の制御が可能である。そのため、図 4.5.12 では WIPS、TB、ATB 方式での消費エネルギーは一定としている。また、ランダムばらつきによる影響は平均化されるため、演算器のリーク電流はほとんど変化しない。そのため、CuD 方式以外のスリープ制御手法はランダムバラつきによる影響は非常に少ないものと考えられる。また、CuD 方式はリークモニタ回路の出力遷移時間が大きく変化したのに対し、25°C の DCT 実行した場合、出力遷移時間が-28%減少した場合に消費エネルギーが 18%増加するにとどまる結果となった。そのため、他のスリープ制御手法より消費エネルギーを削減できた。

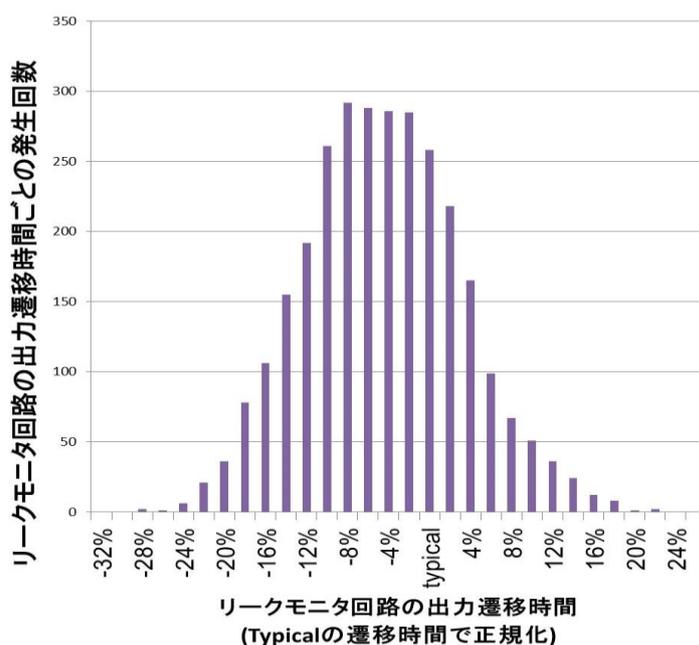


図 4.1.11 ランダムばらつきによるリークモニタ回路の出力遷移時間への影響

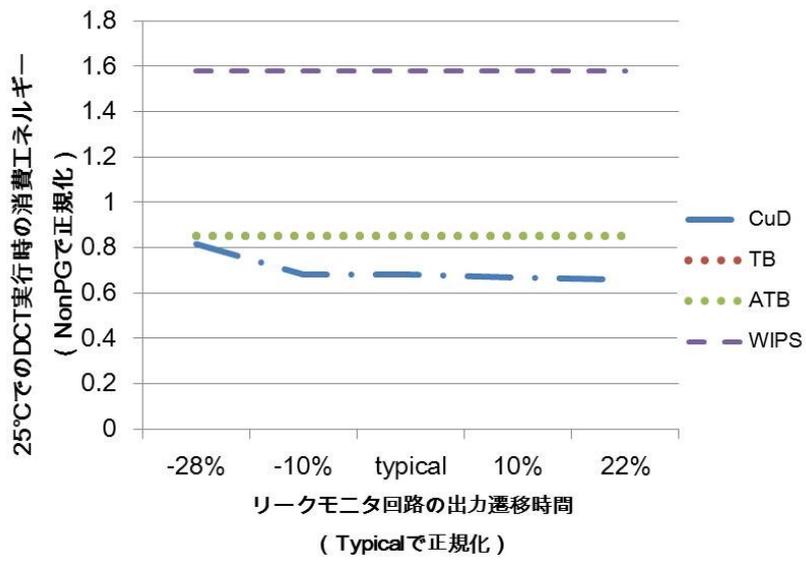


図 4.1.12 ランダムばらつきによる各スリープ制御手法の消費エネルギー

## 4.2 考察とまとめ

第4章では、マイクロプロセッサのさらなる低消費電力化を実現するために、細粒度 PG を適用して、ランタイムパワーゲーティングによるアプリケーション実行中のリークエネルギーの削減効果の評価を行った。細粒度 PG を制御するためのスリープ制御手法として、従来の手法に対し、著者が提案した手法を使用することにより、スリープ制御回路を含めた演算器全体の消費エネルギーを最も削減できることを示した。また、従来手法および提案手法による消費エネルギーのモデル式を構築し、適用回路が変化した場合の消費エネルギーをシミュレーションで求めることでその影響を示した。さらに、提案手法におけるばらつきの影響についても調査した。

また、本論文では 1 種類のスリープ制御手法を演算器に適用することを想定し、消費エネルギーの評価を行った。しかし、4.5 節で述べたように、適用する演算器ごとにそれぞれスリープ制御手法が存在することが分かった。そのため、各演算器に別のスリープ制御手法を適用する際には以下のように適用した場合が最もエネルギーを削減できると考えられる。

- ALU

アプリケーションに依らず、頻繁に回路が使用されるため、PG を適用しない NonPG が最も適している。しかし、CuD 方式の消費エネルギーは最大でも 10% の増加にとどまった。

- SHIFT・MULT

スリープ制御回路の消費エネルギーが小さく、PG によるオーバーヘッドを効果的に抑えられる CuD 方式が、低温時から高温時まで効果的に消費エネルギーを抑えられるため最も適している。これにより、CuD 方式は 5% までしか消費エネルギーが増加することは無い反面、最大で 10% にまで消費エネルギーを削減できている。

- DIV

アプリケーションに依らず、使用頻度が極端に低いため、WIPS 方式が最も適している。これにより、32%~4% にまで消費エネルギーを削減できている。また CuD 方式も 34%~4% にまで消費エネルギーを削減できた。

また、チップ上の温度によっても、適したスリープ制御手法が変わることが分かった。しかし、温度ごとに適したスリープ制御手法を行うためには、スリープ制御回路を搭載し、動的に切り替える機能のほかに、チップ上の温度をモニタする回路が必要となってくる。これらの制御回路の消費エネルギーや面積のオーバーヘッドも問題となる可能性がある。そのため、温度ごとにスリープ制御手法を変える場合にはこれらのオーバーヘッドを小さく抑える工夫をした回路を考案する必要がある。

さらに、本論文では pMOS のリーク電流を利用したリークモニタ回路を用いた。しかし、pMOS と nMOS の両方のリーク電流を使ったリークモニタ回路を用いることで、nMOS、pMOS のしきい値電圧が別方向にばらついた場合に、より適したスリープ制御が可能になると考えられる。しかし、nMOS と pMOS のしきい値電圧が別々に変化した場合、各リーク電流から BET を推察する回路が新たに必要になる。これにより、スリープ制御回路が複雑になり、CuD 方式のようにスリープ制御回路の消費エネルギーを小さくできない可能性がある。

## 第4章 不揮発性 PG 設計手法

---

第4章では、PGを細粒度に適用することでよりリークエネルギーをより削減する技術について述べ、回路が動作時のリークエネルギーを削減するランタイムパワーゲーティングの削減効果について示した。しかし、細粒度PGの評価に用いたマイクロプロセッサは、組み合わせ回路のみで構成されている演算器にだけPGが適用されており、それ以外の回路に対しては適用されていない。これは、一般に記憶回路として使用されるフリップフロップ回路であるD-FFやラッチセルなどにPGを適用した場合、PGの実行中に回路内の記憶情報が破壊されてしまい、PG実行前と実行後で回路状態が変わってしまうためである。これにより、D-FFやラッチセルを含む順序回路ではPGを適用した場合、PGを行うたびに回路の初期化を行うなど、組み合わせ回路に比べ複雑な制御が必要になる。一方で、順序回路を含めたマイクロプロセッサ全体のPGが可能になれば、回路の待機時におけるリーク電力をさらに削減することができ、さらなる低消費電力設計が期待できる。

そのため、記憶回路のデータがPGを行っても保持されるよう、リテンションフリップフロップ (Retention Flip Flop) や不揮発性メモリなどを用いることで、順序回路を含めたすべての回路でPGを行う不揮発性PGが提案されてきている。本章では、不揮発性PGを実現するために、従来研究として提案されているリテンションフリップフロップと不揮発性メモリを説明するとともに、不揮発性の記憶回路であるNVFF (Nonvolatile D-FF) を回路設計し、実チップで評価した。さらに、実チップ上での評価から判明した従来手法の問題点について言及し、それらを解決するための新しいNVFFの回路を提案する。また、シミュレーション評価を用いて、従来手法との比較を行い、提案手法の優位性を示すとともに不揮発性PGの今後の課題について述べる。

### 5.1 リテンションフリップフロップ

マイクロプロセッサ内の制御回路などの順序回路では、記憶回路としてD-FFやラッチセルが主に用いられている。図5.1.1は一般的なD-FFの回路図を示す。D-FF回路はマスターラッチとスレーブラッチから構成される。クロック信号 (CLK) が '0' のとき、入力信号 (D) が変化しても出力信号 (Q) は変化しない。また、CLK が '0' から '1' に遷移すると、QはDと同じ値になり、再びCLKが '0' から '1' に遷移するまでQの値は保持される。しかし、このD-FFはPGを行うと内部のデータを保持しておくことができない。そのため、D-FFやラッチセルをPG実行後もデータが保持できるようにすれば、記憶回路

を含めた回路全体を PG することが可能になる。そのための手法として提案されているリテンションフリップフロップの手法について以下に述べる。

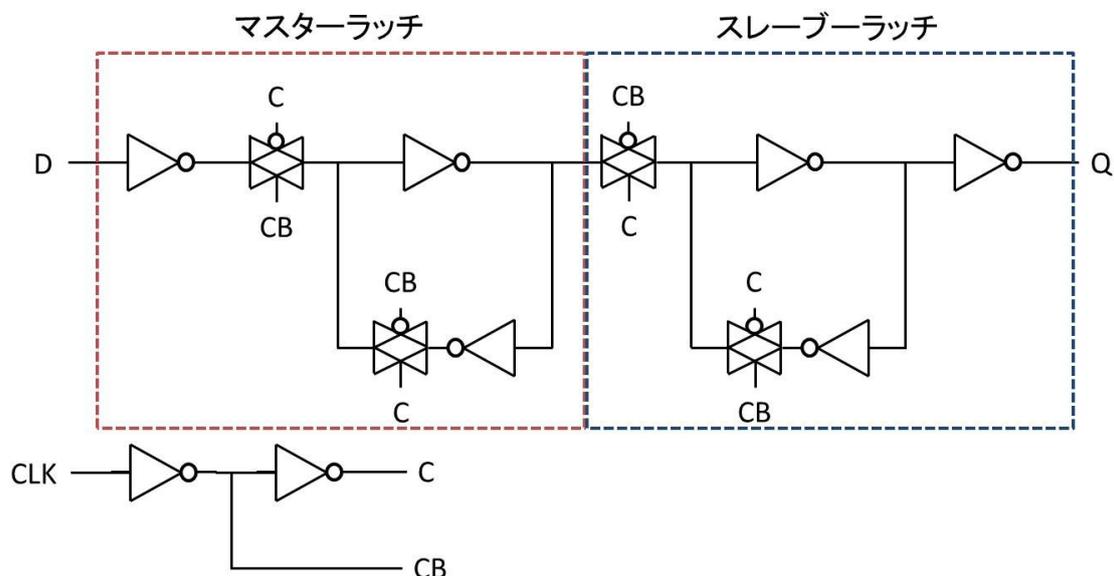


図 5.1.1 D-FF 回路図

### 5.1.1 SRAM 退避方式

SRAM 退避方式は、回路内で保持すべき記憶回路の情報を PG の適用外である外部メモリに一度退避し、PG 終了後に再び記憶回路へ書き込む方法である[30]。図 5.1.2 に SRAM 退避方式の概略図を示す。外部メモリにはリーク電力が少なく、高速動作が可能である SRAM が用いられる。また図 5.1.3 は SRAM の 1bit の記憶回路の回路図である。WL 線で書き込み、読み出しの制御を行い、BL、BLB 線でデータを受け取ることができる。SRAM 退避方式では、まず PG 適用回路が待機状態になると書き込み制御回路を通じて SRAM にデータを書き込む。この SRAM は PG の適用外にあるため、PG 実行中も SRAM 内のデータが失われることはない。次に、PG が終了し、回路が使用可能な状態になると SRAM から読み出し制御回路を通じてレジスタに元のデータが読み込まれる。これにより、PG 適用回路は PG 前と同じ回路状態で動作を始めることが可能である。また、レジスタとして既存の D-FF などをそのまま使用可能であるという利点がある。

しかし、SRAM は PG の適用外のため、SRAM のリーク電力が必ず生じてしまう。そのため、SRAM のリーク電力を抑える工夫や、保持するデータを選別する必要がある。また、データを保持すべきレジスタの個数分のビット情報を SRAM とやりとりする必要があるため、並列処理などの工夫を行わなければ、データの読み書きに膨大なクロック数が必要に

なり、時間的なオーバーヘッドを生ずる。また、データの読み書きで生ずる消費エネルギーもオーバーヘッドとなる。

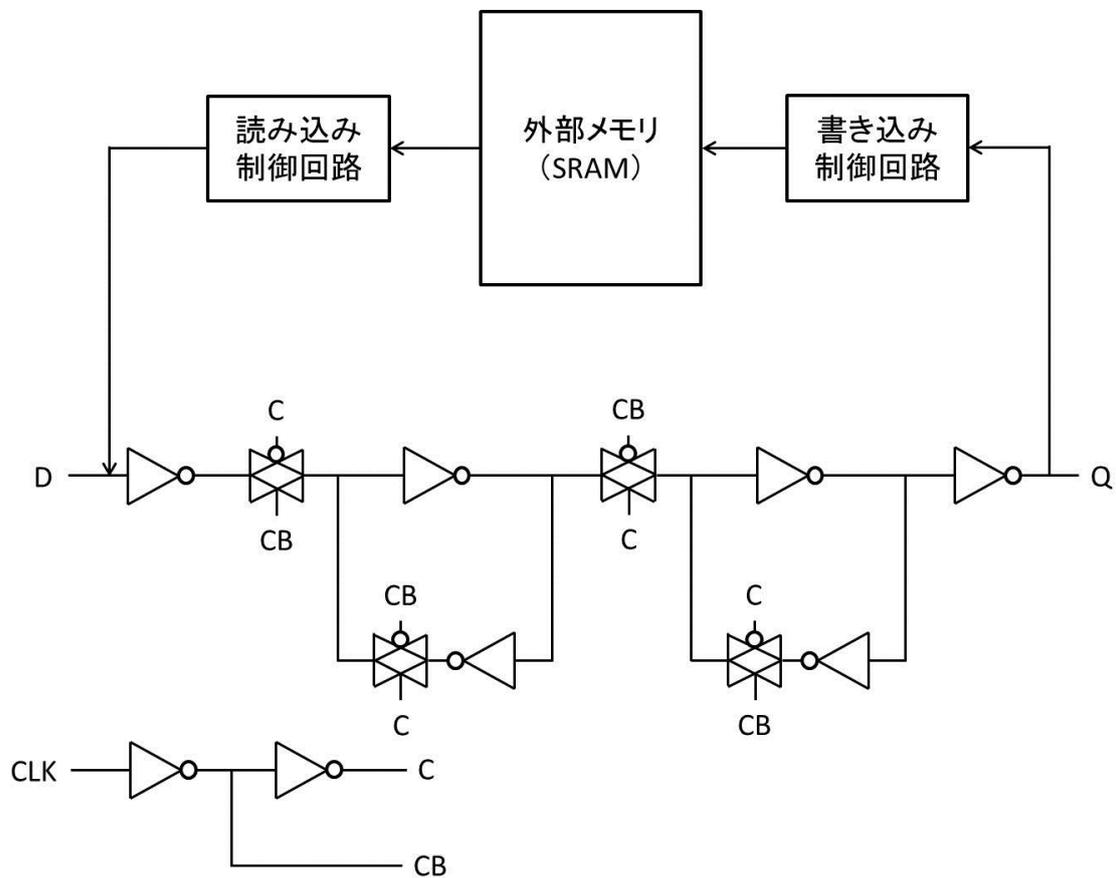


図 5.1.2 SRAM 退避方式の概略図

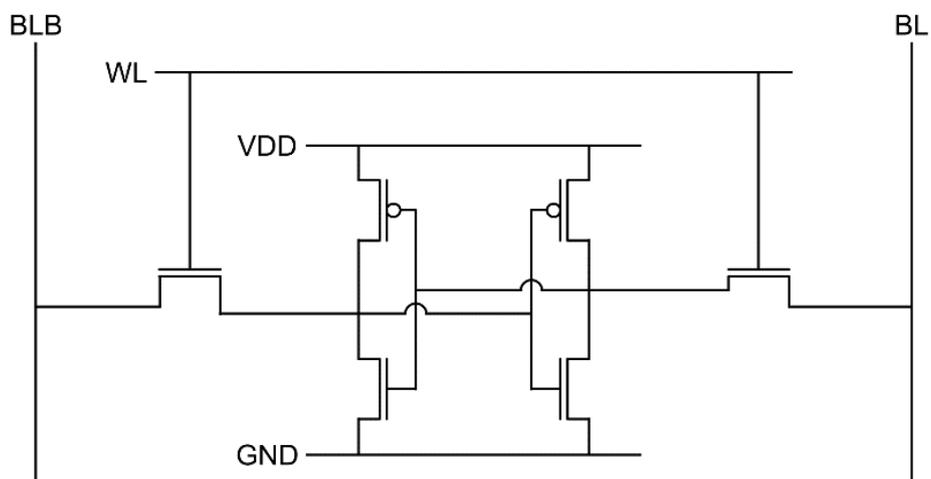


図 5.1.3 SRAM1bit のメモリセル

## 5.1.2 バルーン方式

バルーン方式は **D-FF** の回路内にデータを保持する領域を用意する方式である[47]。図 5.1.4 はバルーン方式を適用した **D-FF** の回路図である。この回路では、**PG** が可能になると **D-FF** のスレーブラッチのビット情報をバルーンラッチと呼ばれるラッチセルに退避する。また、**PG** が終了し、回路が動作可能になると逆にバルーンラッチのデータをスレーブラッチへ書き込むことで、元の回路状態へ復帰する。さらに、バルーンラッチのトランジスタだけを高しきい値にし、その他のトランジスタを低しきい値にすることで、リーク電力を削減しつつ、**D-FF** の動作速度を高速に設計することができる。また、**SRAM** 退避方式と比べ、すべての **D-FF** のデータを同時に退避させることができるため、退避時のデータの読み書きを高速に行うことが可能である。

しかし、通常の揮発性 **D-FF** に比べてトランジスタ数が増加してしまう。また、たとえ高しきい値のトランジスタを用いても、バルーンラッチのトランジスタ数分だけ、削減できないリーク電力が存在する。さらに、ナノスケールのトランジスタでは回路面積の削減のため、トランジスタの基板電圧の印加は、セルの外部で共通で行うことが主流である。そのため、1つのセル内で **PG** 適用領域と **PG** 非適用領域が混在するバルーン方式では、セル内で基板を分離するなど非常に複雑なセル設計が必要になると考えられる。

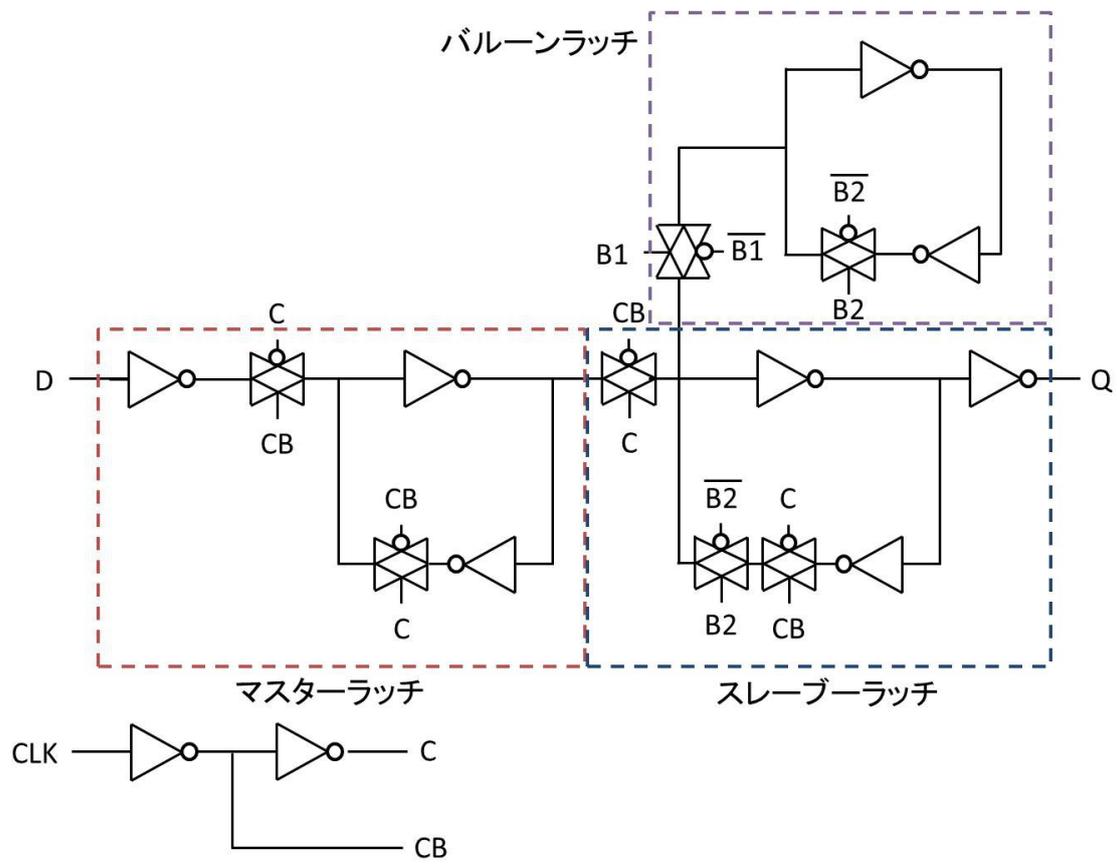


図 5.1.4 バルーン方式の D-FF 回路図

## 5.2 不揮発性メモリ

不揮発性を実現するメモリとして、スピン注入磁化反転 (Spin Transfer Torque : STT) と呼ばれるデータ書き換え技術を用いた MRAM である STT-MRAM や、相変化メモリ (PCM)、抵抗変化型メモリ (ReRAM) [32]が提案されている。

### 5.2.1 NAND フラッシュメモリ

不揮発性メモリの代表的なものとして NAND フラッシュが挙げられる。フラッシュメモリは、通常の MOS トランジスタと異なり、絶縁体中にフローティングゲートを持つ MOS トランジスタを使用する。このフローティングゲートに電荷を貯めると、まわりが絶縁体に囲まれているため、電荷が抜けることがなく不揮発性メモリを実現することができる。また、NAND フラッシュメモリは NAND 型の論理回路を用いるフラッシュメモリであり、従来のフラッシュメモリに対し、セル面積を抑えて集積度を大きく上げることができる。

しかし NAND フラッシュメモリはデータの書き込み、読み出しに時間がかかるという欠点があり、マイクロプロセッサ全体の記憶回路のデータを保持し、PG を行う不揮発性 PG には不向きである。

### 5.2.2 抵抗変化型メモリ (ReRAM)

ReRAM はフラッシュメモリと同様に電源がなくともデータを保持できる不揮発性メモリのひとつである。また DRAM 並みの動作速度で書き込み、読み出し動作が可能である。

ReRAM は印加する電圧によって大きく抵抗値が変化する電界誘起巨大抵抗変化と呼ばれる現象を利用しており、この際の抵抗値の高低差を利用してデータを保持する。記憶素子を 20 nmφ 程度で製造でき、動作速度も高速だが、データを書き込む際に必要な電圧が 3V と高い電圧を用意する必要がある。また、繰り返し書き込みを行える回数が少ないという欠点がある。

### 5.2.3 強誘電体メモリ (FeRAM)

FeRAM は強誘電体を利用した不揮発性メモリである。強誘電体とは、電圧を印可することで物質内の自発分極の方向を自由に变化させ、電源を遮断してもその分極方向を保持できる誘導体のことである。FeRAM も DRAM 並みの速度で動作させることが可能であり、さらにデータの書き込みに必要な消費エネルギーが非常に少なくすむ。しかし、強誘電体の素子面積が巨大になってしまうという問題があり、不揮発性 PG を行う際の面積オーバーヘッドが大きな問題となる。

## 5.2.4 STT-MRAM

STT-MRAMは書き込みや読み出しの消費エネルギーで他の2つのメモリに優れていることが知られており、オンチップメモリ[49]やフリップフロップ[56]に用いる技術として多くの研究が行われている。この STT-MRAM の重要な技術の 1 つとして、磁気トンネル接合 (MTJ : Magnetic Tunnel Junction) が挙げられる (図 5.2.1)。

この MTJ 素子は上下にある固定層とフリー層と呼ばれる 2 つの強磁性層とその間の MgO バリア層から構成される。MTJ 素子は固定層とフリー層の磁化の向きにより抵抗が変化する特性がある。磁化の向きが平行の場合 (P : Parallel)、MTJ 素子の抵抗は低抵抗になり、磁化の向きが反並行の場合 (AP : Anti-Parallel)、MTJ 素子の抵抗は高抵抗になる。

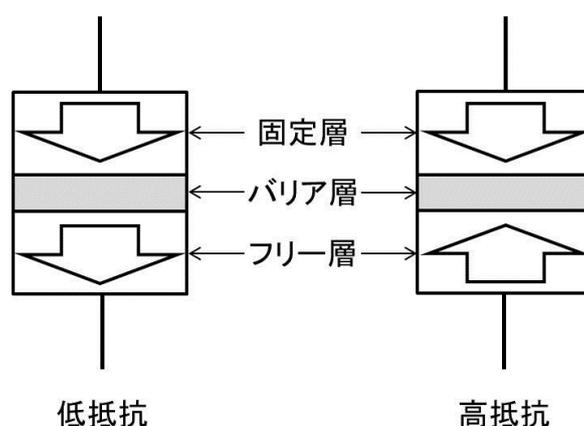
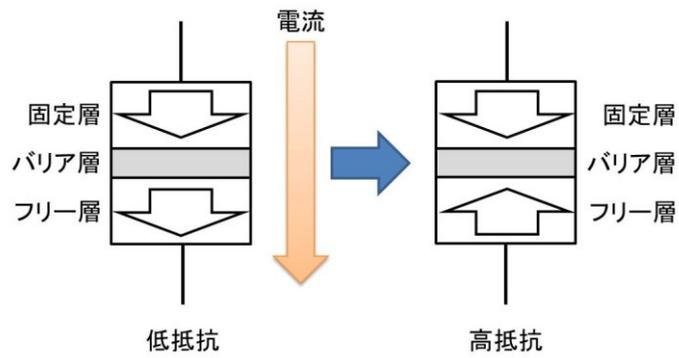


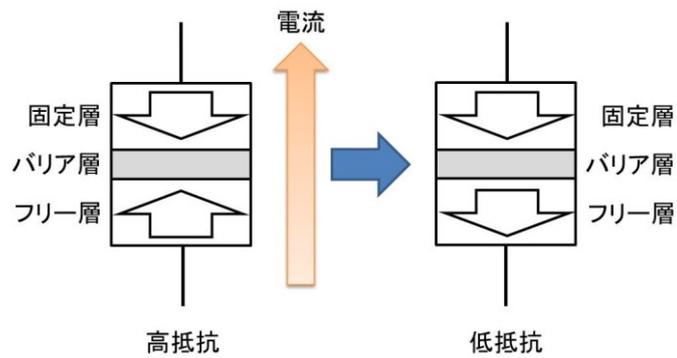
図 5.2.1 MTJ 素子

図 5.2.2 に示すように、MTJ 素子の抵抗値は、一定の電流を MTJ 素子に流すことで変化させることができる。また、その抵抗値は流す電流の向きによって、低抵抗か高抵抗に変化する。MTJ 素子が低抵抗の場合、固定層からフリー層へ電流を流すことで、抵抗値が高抵抗へ変化させることができる。また、MTJ 素子が高抵抗の場合、フリー層から固定層へ電流を流すことで抵抗値は低抵抗へ変化する。抵抗の変化による MTJ 素子の I-V 特性を図 5.2.3 に示す。I<sub>cP</sub>→AP は低抵抗から高抵抗へ書き換えるために必要なストア電流を示す。また、I<sub>cAP</sub>→P は高抵抗から低抵抗へ書き換えるために必要なストア電流である。

MTJ 素子の書き込みには 1V 程度の電圧で十分なため、不揮発性メモリのために高い電圧線を用意する必要がない。また素子の体積を数十 nmφ と比較的小さくでき、繰り返し書き込みの耐性も強い。



(a) 低抵抗から高抵抗への変化



(b) 高抵抗から低抵抗への変化

図 5.2.2 電流の向きによる MTJ の抵抗変化

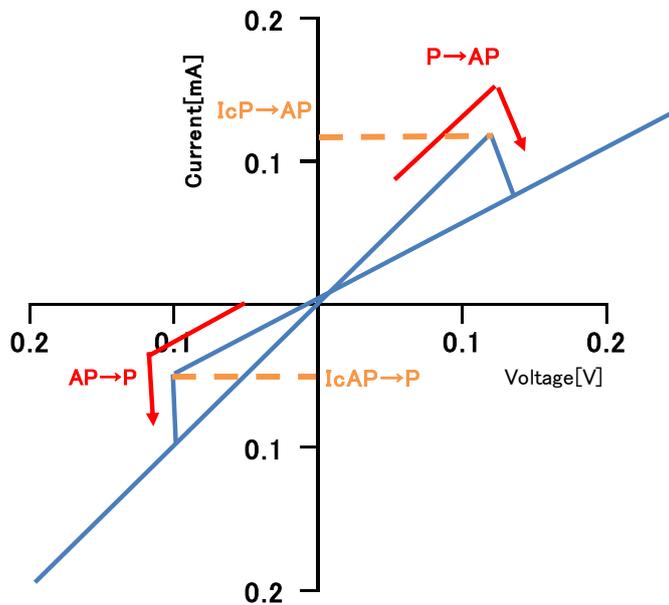


図 5.2.3 MTJ 素子の I-V 特性

この MTJ 素子を用いた不揮発性 D-FF (NVFF : Nonvolatile Flip Flop) がいくつも提案されている[54,55]。

[31]では、2つの MTJ 素子を D-FF のマスターラッチに加えた手法が提案されている。この手法ではインバータの nMOS のソース側と MTJ 素子が繋がっており、毎クロックで MTJ に書き込みが行われる。しかし、クロックのたびに MTJ 素子への書き込みが行われるため、消費エネルギーが増大してしまう。また、[51]では別の回路として、マスターラッチのインバータの pMOS 側に 2つの MTJ を配置した手法が提案されている。しかし、この手法も前述した回路と同じ問題を抱えている。

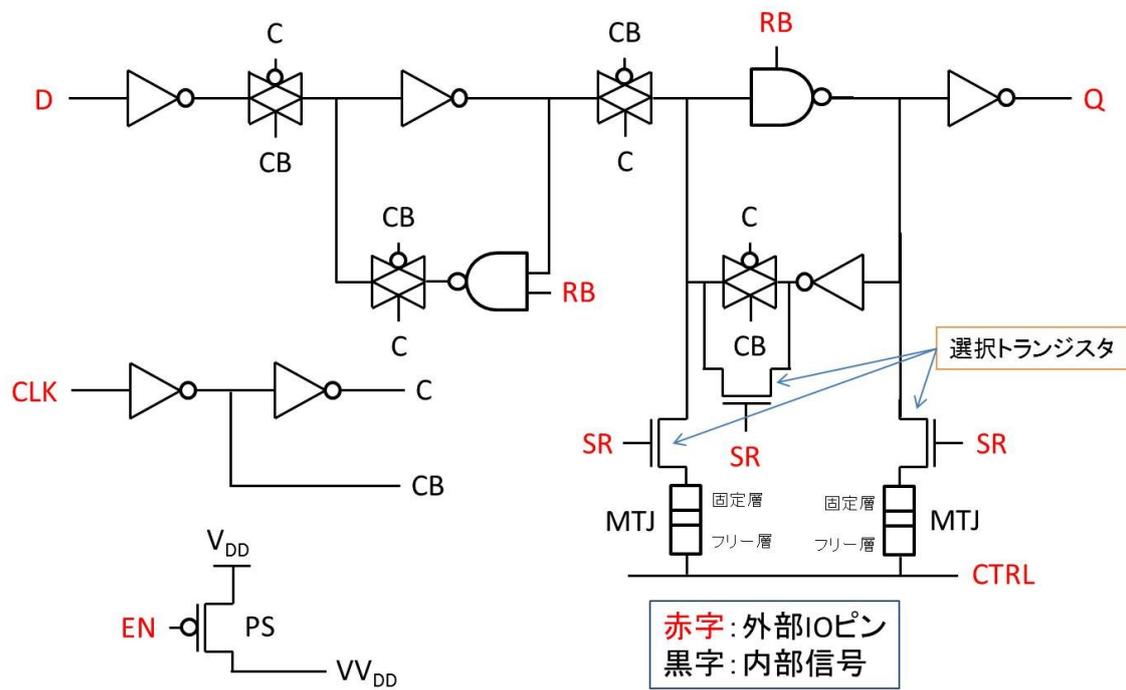
一方、[52]では、MTJ 素子を含む比較回路を通常の D-FF のスレーブラッチに付与している。この手法では、PG が実行される前に MTJ 素子にデータを書き込み、PG が終了したのを検知すると、2つの MTJ 素子の抵抗値の差を利用して、元のデータが D-FF に読み込まれる。PG が実行される場合にのみ、MTJ 素子への書き込みが行われるため、無駄な消費エネルギーを削減することができる。しかし、比較回路による面積のオーバーヘッドが大きいという問題がある。

これに対し、[53]では、既存のスレーブラッチに存在するインバータループを使用してリストア動作を行う手法を提案している。図 5.2.4 にこの手法によるリセット機能付の NVFF の回路図を示す。この手法では、通常の D-FF に対し、2つの MTJ の素子と選択トランジスタとして 3つのトランジスタを追加するだけで NVFF を実現することが可能である。また、HDPG の場合には nMOS を選択トランジスタとして使用し、FTPG の場合には pMOS を使用することで、PG の方式に関係なく NVFF を実現することができる。[53]ではこの選択トランジスタ 1個と MTJ 1個の組み合わせを pseudo-spin-MOSFETs (PSM) と呼んでいるため、本論文では、この手法による NVFF を PSM-NVFF と呼ぶことにする。

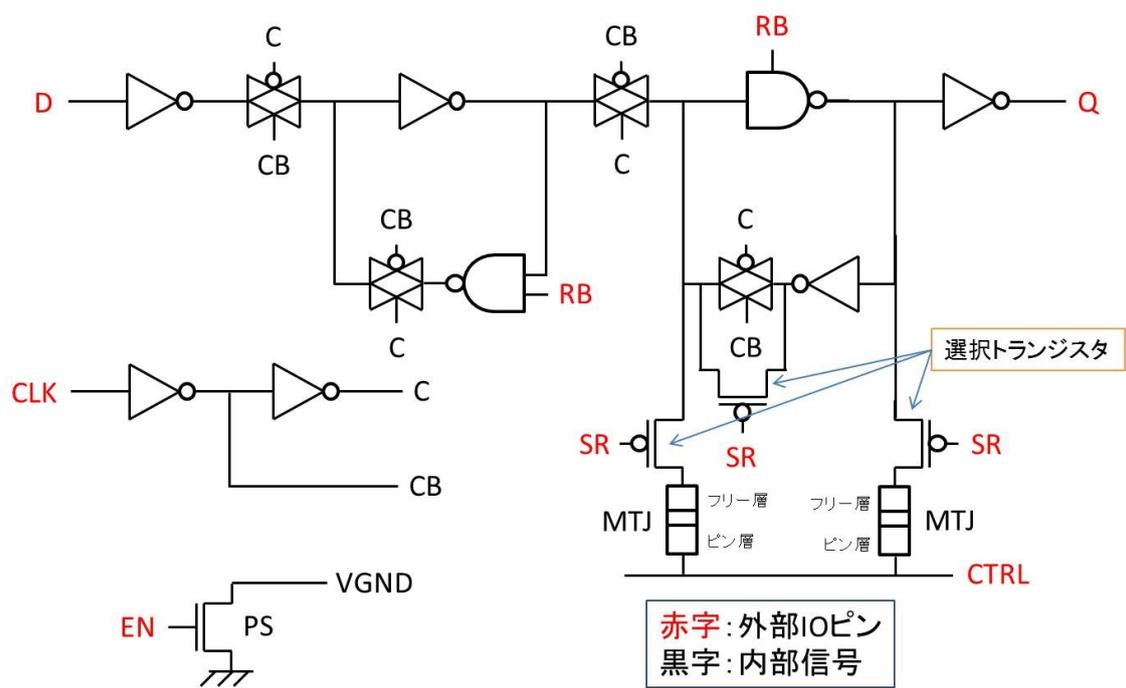
図 5.2.5 には PSM-NVFF の制御シーケンスを示す。PSM-NVFF を用いて不揮発性 PG を行う際には、

- アクティブ動作
- ストア動作
- スリープ動作
- リストア動作

以上の 4つの動作を行う必要がある。

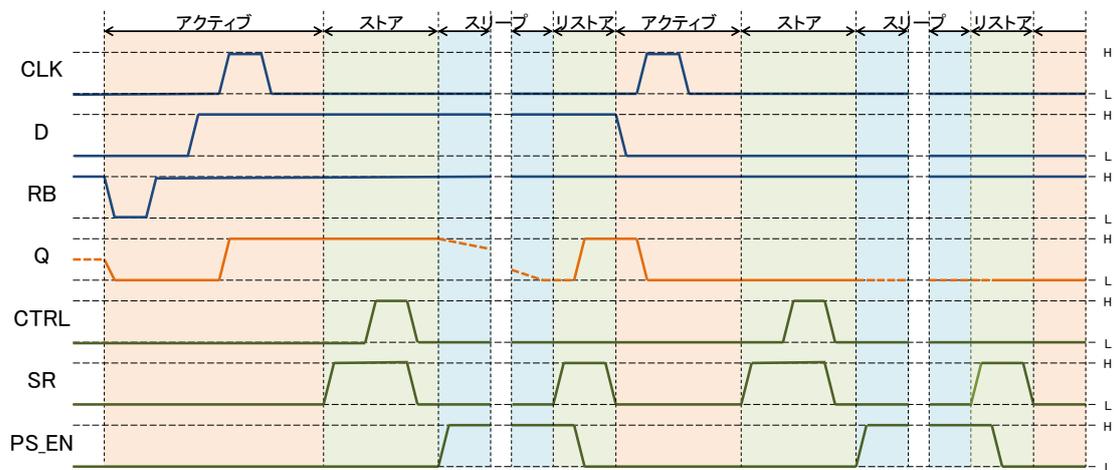


(a) HDPG

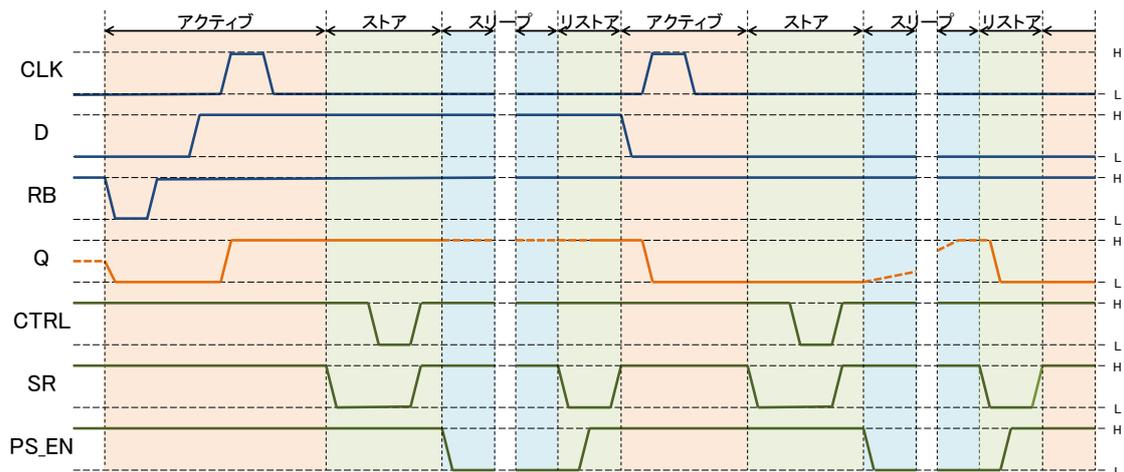


(b) FTPG

図 5.2.4 従来の PSM-NVFF の回路図



(a) HDPG



(b) FTPG

図 5.2.5 PSM-NVFF の制御シーケンス

各動作時における制御方法を以下に説明する。

### (a) アクティブ動作

HDPG の場合、アクティブ動作時には、選択トランジスタの信号である SR 信号と PS のイネーブル信号である EN を ‘0’ にする。(FTPG の場合、SR 信号と EN 信号を ‘1’) これにより、PS はオンになり、選択トランジスタはオフの状態となる。この際、PSM-NVFF は通常の D-FF と同じ動作が可能となる。また、RB を ‘0’ にすることで、非同期で出力 Q を ‘0’ に初期化することができる。

### (b) ストア動作

PG 適用回路が動作状態から待機状態になると、アクティブ動作をしていた PSM-NVFF はストア動作へと移行する。ストア動作では、PSM-NVFF 内のデータの情報を MTJ に書き込む動作を行う。図 5.2.6 に HDPG での PSM-NVFF でのストア動作について示す。まず、EN 信号は ‘0’ の状態のまま SR 信号は ‘0’ から ‘1’ へ遷移する。この際、MTJ に接続されている CTRL 線は ‘0’ である。この時、スレーブラッチのループ部のうち、‘1’ を保持している側の MTJ に選択トランジスタから CTRL 線に向かって電流が流れる。この際、MTJ には固定層からフリー層へ電流が流れるため、MTJ は高抵抗になる。

次に、CTRL 線を ‘1’ にすることで、スレーブラッチのループ部のうち、‘0’ を保持している側の MTJ に CTRL 線から選択トランジスタへ方向の電流が流れる。この時、MTJ にはフリー層から固定層へ電流が流れるため、MTJ は低抵抗になる。これにより、MTJ への書き込みが行える。最後に、SR 信号と CTRL 線を ‘0’ にすることでストア動作は終了する。

ストア動作の流れ

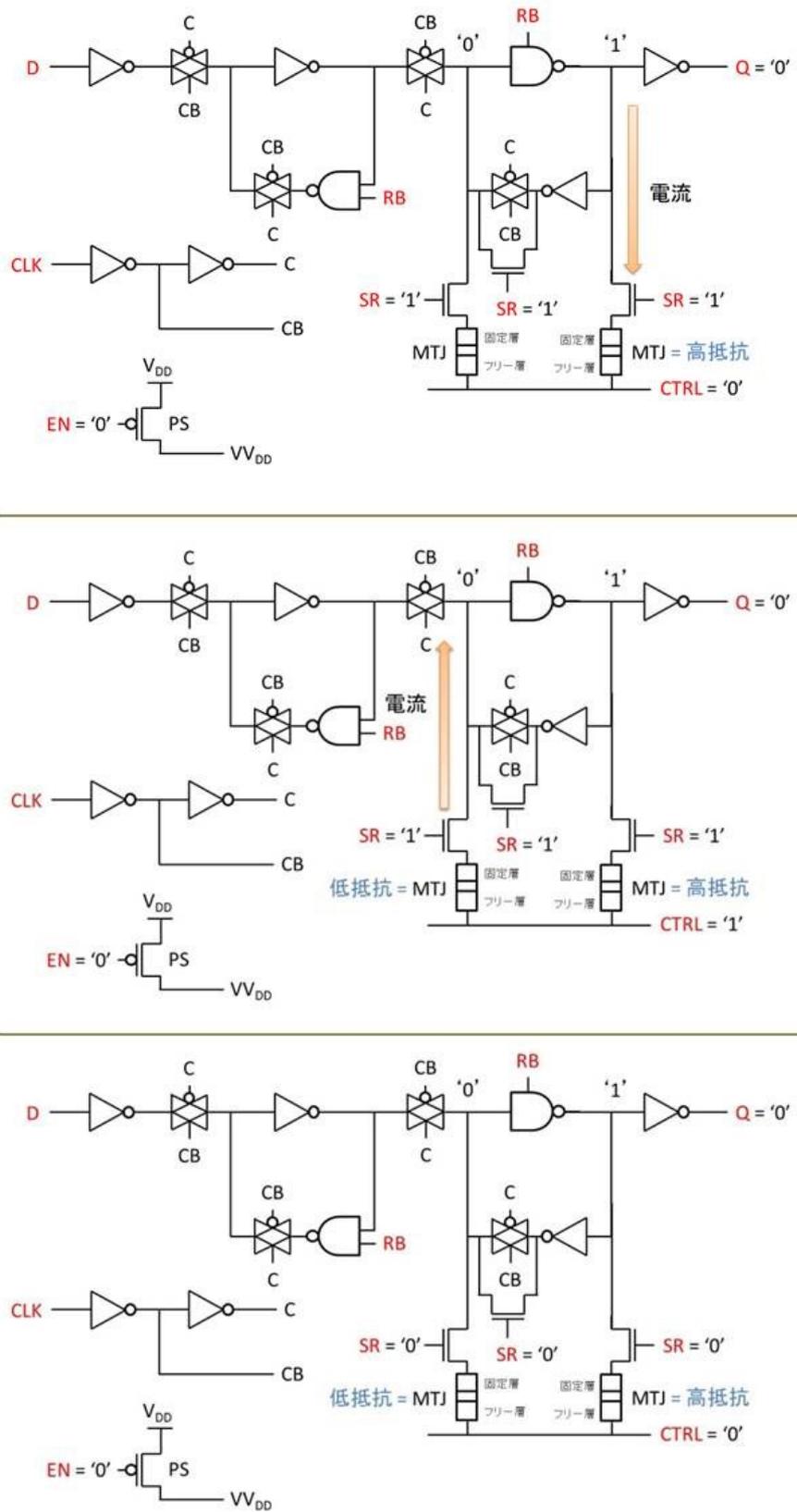


図 5.2.6 ストア動作

### (c) スリープ動作

ストア動作により、MTJへPSM-NVFF内のデータの書き込みが行えたため、PSM-NVFFのPGが可能となった。スリープ動作ではEN信号を‘0’から‘1’にすることで、PSをオフにし、電源遮断を行う。これにより、スリープ動作中のリーク電力を削減することができる。

### (d) リストア動作

スリープ動作が終了すると、次にリストア動作により、PSM-NVFFのデータをPG前の状態に戻す必要がある。図5.2.7にリストア動作の流れを示す。まず、SR信号を‘1’にし、選択トランジスタをオンする。次に、EN信号を‘0’にする。この時、CTRL線の値は‘0’である。そのため、スレーブラッチのループ部はそれぞれ選択トランジスタからCTRL線に向かって電流が流れる。しかし、MTJの抵抗値の違いにより、スレーブラッチの左右で流れる電流は異なる。この時の流れる電流の違いにより、高抵抗のMTJが接続されている側のスレーブラッチは‘1’となり、低抵抗のMTJが接続されている側は‘0’となる。これにより、PG前の状態にPSM-NVFFのデータを読み出すことができる。最後にSRを‘0’にし、選択トランジスタをオフすることで、リストア動作が終了すると、PSM-NVFFはアクティブ動作が可能となり、通常のD-FFと同じ動作を行うことができる。

リストア動作の流れ

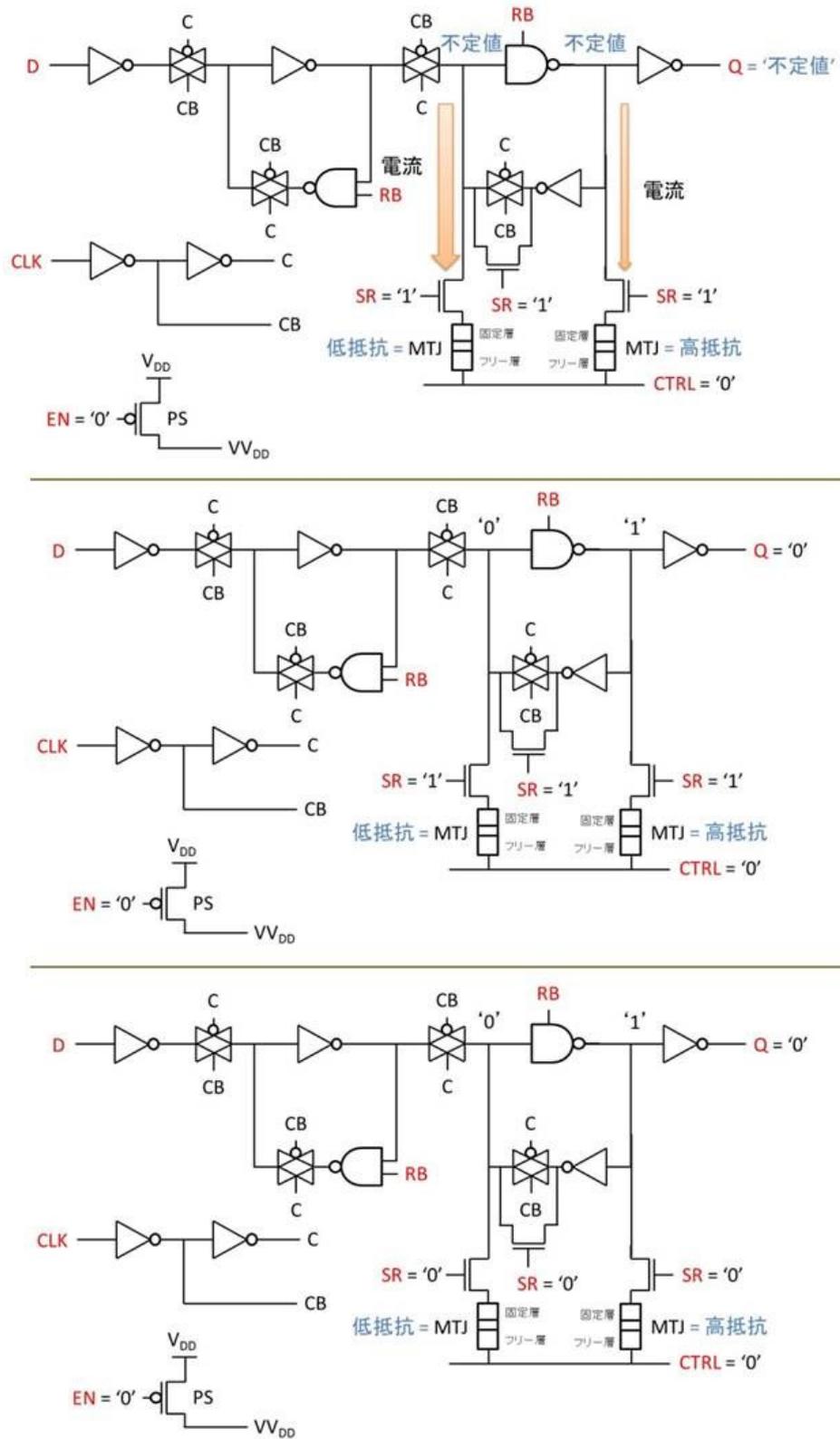


図 5.2.7 リストア動作

PSM-NVFF は、MTJ 素子へのストアエネルギーを削減できるほか、トランジスタ数を抑えられる特徴がある。また、他の不揮発性メモリに比べ、動作速度が速いかつ低電圧でデータの読み出し、書き込みが可能である。さらに、他の不揮発性素子に対し、MTJ は繰り返しの書き込みに強いという特徴がある。マイクロプロセッサ等に対し、不揮発 PG を適用した場合、D-FF のデータの書き込み、読み出しにミスが生じてしまうとマイクロプロセッサ全体の不良動作に繋がってしまう。そのため、PSM-NVFF はマイクロプロセッサ等の回路で行う不揮発 PG において、PSM-NVFF は優れた手法であると言える。そこで本章では、この PSM-NVFF に着目し、分析と議論を行う。

## 5.3 従来方式の分析

先行研究[53]では、PSM-NVFF の評価をシミュレーションのみで行っており、シリコン上での測定は行っていない。また、その他の NVFF の評価もそのほとんどがシミュレーションでのみ行われている。しかし、ストア、リストア動作のロバスト性やトランジスタのプロセスばらつきを評価するにはシリコン上での測定が必要不可欠になってくる。以上のことより、本節では PSM-NVFF の設計し、シリコン上での測定を行った。さらに、測定結果を元に、新しく得られた知見について述べる。

### 5.3.1 PSM-NVFF のレイアウト設計

図 5.3.1 は 65nm プロセスを用いた FPG 方式の PSM-NVFF のレイアウト図である。PSM-NVFF のマスターラッチは回路の左側に配置し、通常の NV-FF と同様のトランジスタサイズを使用している。一方、スレーブラッチおよび選択トランジスタは MTJ 素子へのストア電流を確保するために、トランジスタの W のサイジングを行っている。また、MTJ 素子はトランジスタの上層の配線層と配線層の間に挿入することができるため、面積の影響は小さく抑えることができている。今回用いた MTJ 素子は直径 65nm であり、MTJ の抵抗値の変化の大きさの比率であるトンネル磁気抵抗 (TMR) が 106%、抵抗面積積 (抵抗×面積) が  $5.5\Omega\mu\text{m}^2$  である。また標準の電源電圧は 1.2V とし、動作周波数は 50MHz を想定し測定した。

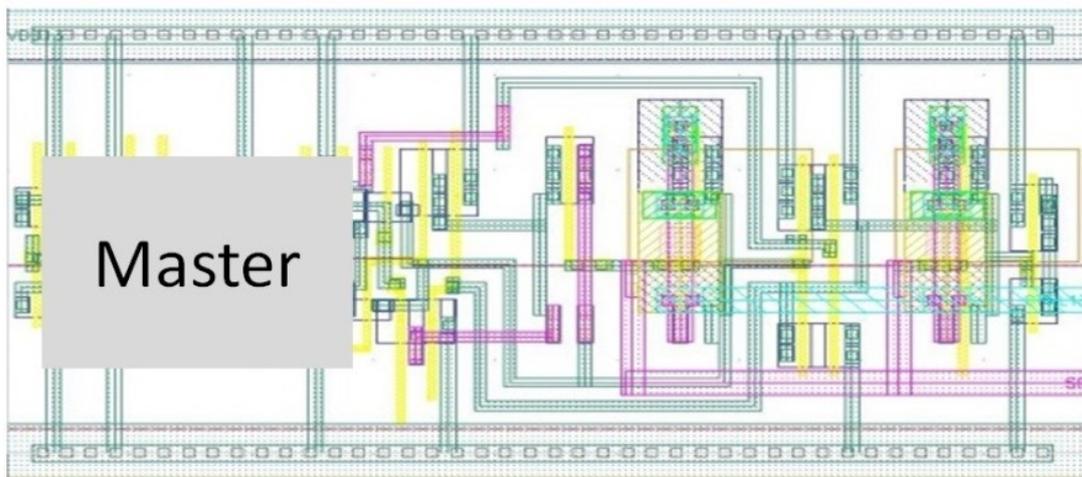


図 5.3.1 PSM-NVFF レイアウト図

### 5.3.2 PSM-NVFF の問題点

5.3.1 節で述べたように、PSM-NVFF を設計し、シリコン上で評価を行ったことにより、不揮発 PG を実現するうえで、PSM-NVFF の問題点が明らかになった。PSM-NVFF の解決すべき問題点について以下の 3 つを示す。

- ストア動作のロバスト性
- リストア動作のロバスト性
- 無駄な書き込みエネルギーの発生

不揮発 PG を行うための NVFF は、メモリのように Error Correction Code (ECC) などを使ったデータのエラー検知による自動修復機能を搭載することが困難である。そのため、ストア動作、リストア動作の動作不良による影響が大きい。よって、ストア動作、リストア動作それぞれにおいて高いロバスト性が求められる。また、不揮発性 PG において、現在の MTJ では抵抗値を書き換えるために必要なエネルギーが PG によるリークエネルギー削減効果と比べても非常に大きいものとなっている。つまり、MTJ への書き込み回数が増えると不揮発性 PG における BET も長くなってしまふ。それぞれの問題点について詳しい説明を以下に述べる。

#### (a) ストア動作のロバスト性

PSM-NVFF ではストア動作時のロバスト性が大きな問題となってくる。ストア時に動作不良を起こす主な原因として以下の 2 つが挙げられる。

- MTJ 素子を流れるストア電流が少ない
- ストア時にスレーブラッチの論理値が反転する (ラッチ破壊)

MTJ 素子を流れる電流が MTJ 素子の磁化を反転させるために必要な電流よりも少ない場合、ストア動作は失敗する。しかし、選択トランジスタと MTJ で 1 つの回路として考えた場合、図 5.3.2 に示すように、電流の向きによって電流の流れやすさが異なってくる。選択トランジスタである nMOS のドレイン側に MTJ が接続されている左側の場合、nMOS のソース側はグラウンドと直接つながっている。そのため、nMOS は十分な駆動力を発揮できる。しかし、nMOS トランジスタのソース側に MTJ が接続されている場合、MTJ の抵抗値によってソースの電圧は上がってしまう。これにより、nMOS の駆動力は大きく低下してしまう。これにより、同じ選択トランジスタを使用していても流れやすい電流の向きと流れにくい電流の向きが生じる。また、図 5.2.3 で MTJ 素子の I-V 特性を示したように、MTJ 素子の抵抗値を変化させる方向によっても必要な電流値は異なってくる。PSM-NVFF の場合、より大きな電流が求められる低抵抗から高抵抗へ MTJ 素子を書き換えるストア電流 ( $I_{cP \rightarrow AP}$ ) は、図 5.3.2 の流れにくい方向の電流である。つまり、PSM-NVFF では、流れにくい方向の電流で、より大きな電流が求められる。そのため、 $I_{cP \rightarrow AP}$  の電流を確

保するために、選択トランジスタを非常に大きくする必要はある。しかし、その際、 $I_{cAP} \rightarrow P$  の電流では、必要以上の電流が流れてしまい、無駄なエネルギーが消費されてしまう。また回路面積への影響も大きくなる。

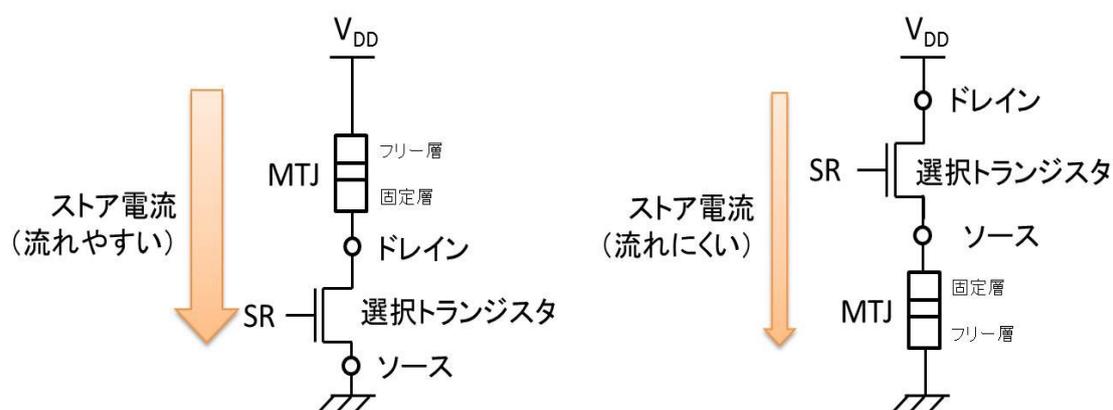


図 5.3.2 ストア電流の向きと電流の流れやすさ (HDPG)

さらに、PSM-NVFF では、ストア動作中にスレーブラッチの論理値が反転してしまい、保持していたデータが破壊されてしまうラッチ破壊という問題も生じてしまう恐れがある。図 5.3.3 にラッチ破壊が発生する例を示す。CTRL 線が '1' の場合、選択トランジスタ側からスレーブラッチ側へストア電流が流れる。この際、MTJ 側の選択トランジスタの駆動力に対し、スレーブラッチ側の駆動力が小さいとスレーブラッチのループ部の論理値が反転してしまう。これにより、MTJ へ正常な書き込みが行われる前にスレーブラッチ内の論理値が反転してしまうため、正しいストア動作を行うことができなくなる。また、今回の PSM-NVFF では、マイクロプロセッサなどでも使用できるように、Reset 機能付きの D-FF を基にした NVFF を設計した。そのため、スレーブラッチに NAND セルを用いており、左右のトランジスタ数がアンバランスになっている。そのため、ストア電流が流れる向きによってはスレーブラッチのノードの電圧が大きく変化してしまい、論理値の反転が起こりやすい。図 5.3.4 にはストア動作を行う際に流れる電流の経路を示す。このとき、ラッチ破壊を防ぐためには、トランジスタのオン時の抵抗と MTJ の抵抗に対し、以下の式を満たすように、スレーブラッチのトランジスタサイズを大きく必要がある[57]。

$$R_{MTJ1} + R_{TR1} \gg R_{TR3} + R_{TR11} \quad (\text{式 5.1})$$

$$R_{MTJ2} + R_{TR2} \gg R_{TR21} + R_{TR20} + R_{PS} \quad (\text{式 5.2})$$

$$R_{MTJ2} + R_{TR2} \gg R_{TR22} \quad (\text{式 5.3})$$

$R_{TR1}$  および  $R_{TR2}$ 、 $R_{TR3}$  は選択トランジスタのオン抵抗を示す。また、 $R_{TR11}$  および  $R_{TR20}$ 、 $R_{TR21}$ 、 $R_{TR22}$  はスレーブラッチのトランジスタのオン抵抗を示す。さらに、 $R_{MTJ1}$  および  $R_{MTJ2}$  は MTJ 素子の抵抗値である。MTJ 素子の抵抗値は使用するプロセスに依存するため、固定値であるとする。そのため、ラッチ破壊を防ぐには、選択トランジスタとスレーブラッチのトランジスタのトランジスタサイズを使って上記式を満たす必要がある。そのためには、MTJ 素子と接続されている選択トランジスタのサイズを小さくすることで、左辺の抵抗値を大きくし、スレーブラッチのトランジスタのサイズを大きくすることで、右辺の抵抗値を小さくすることが効果的である。しかし、上記で述べたように、MTJ 素子の磁気を反転させるためのストア電流を確保するためには、選択トランジスタサイズを大きくすることが必要である。そのため、スレーブラッチのトランジスタサイズをより大きくし、右辺の抵抗値を小さくしなければ、ラッチ破壊が頻繁に発生してしまう。これによりスレーブラッチの面積も増大してしまう。

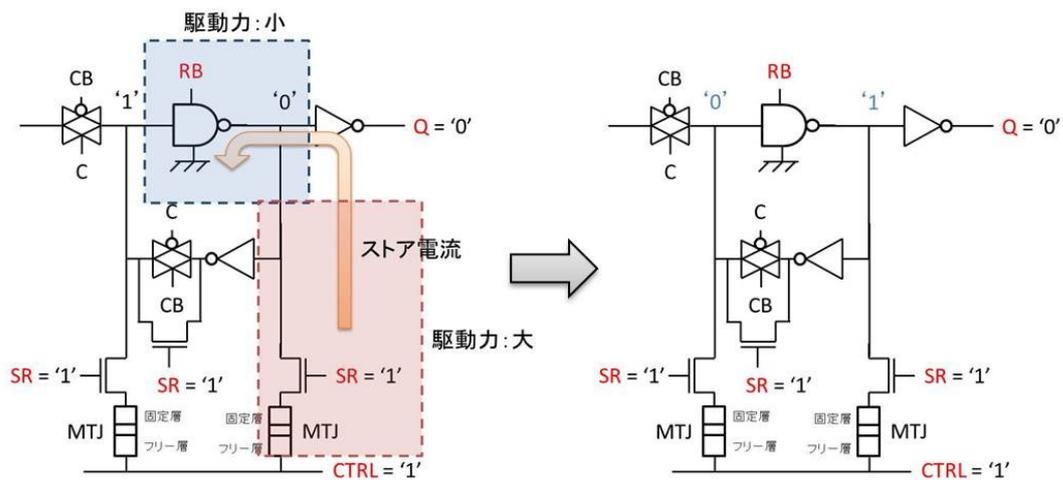
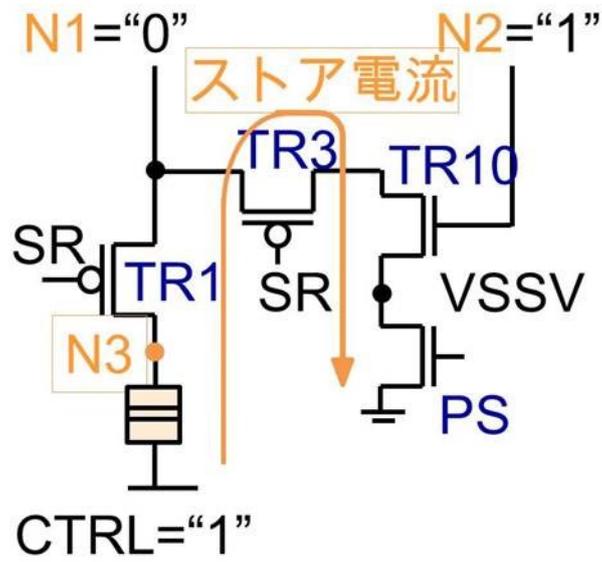
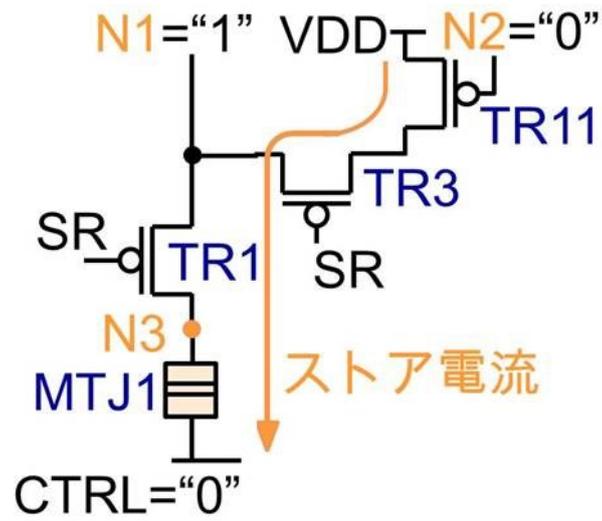


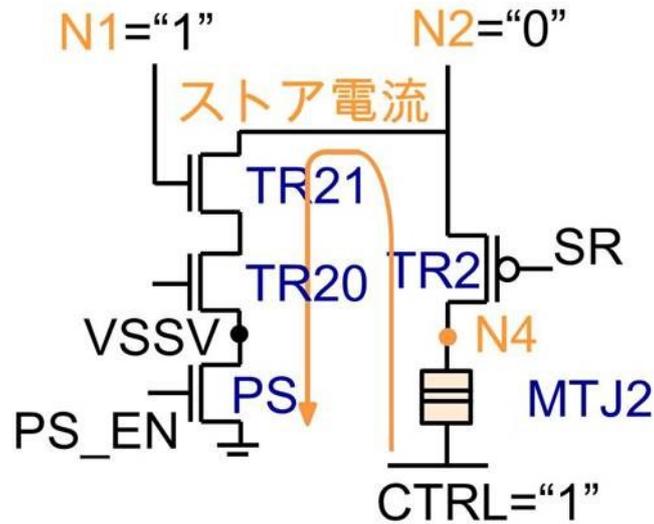
図 5.3.3 ラッチ破壊の発生例



(a) 経路 1



(b) 経路 2



(c) 経路 3

図 5.3.4 ストア時におけるストア電流経路

さらに、シリコン上でラッチ破壊の発生頻度を測定するために、スレーブラッチおよび選択トランジスタのサイズが異なる PSM-NVFF を 3 つ設計し、実チップでの測定を行った。測定結果を図 5.3.5 に示す。まず、スレーブラッチのトランジスタサイズは同じ 65nm プロセスの通常の D-FF のスレーブラッチのサイズを 1 としたときの値を示している。測定結果より、通常と同程度のスレーブラッチのサイズでは 26%もの割合で、ラッチ破壊が発生してしまうことが分かる。また、トランジスタサイズを 3 倍近くまで大きくしても、ラッチ破壊が起こる可能性があることを示している。以上のことから、PSM-NVFF では、ストア動作のロバスト性に大きな課題があり、これにより PSM-NVFF の面積の増大に繋がることが分かる。

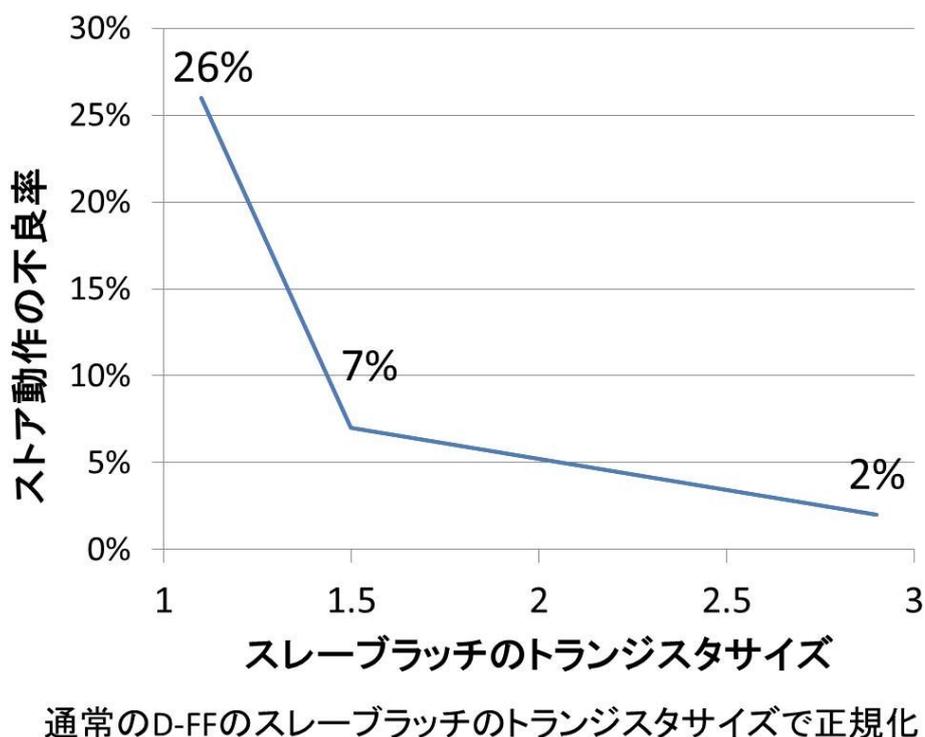


図 5.3.5 スレーブラッチサイズによるストア動作への影響

### (b) リストア動作のロバスト性

リストア動作は、スレーブラッチに繋がれた 2 つの MTJ 素子の抵抗差を利用して行われる。そのため、MTJ 素子の TMR が低い場合、リストアの動作不良が発生する可能性がある。また、今回使用した MTJ 素子の TMR は 106%程度である。しかし、PSM-NVFF では、MTJ 素子を選択トランジスタのソース側に接続するようにしている。この際、MTJ 素子の抵抗分圧によって、選択トランジスタのソース側の電圧が低抵抗の MTJ 素子と高抵抗の MTJ 素子で異なってくる。MTJ 素子が高抵抗の場合、ソース側の電圧が大きく変化する。これにより、逆バイアス効果が生じ、選択トランジスタの駆動能力が大きく低下する。よって、MTJ 素子と選択トランジスタの組み合わせでの駆動能力で比較した場合、MTJ 素子が低抵抗の場合と高抵抗の場合で、TMR 以上の抵抗差が生じる。以上のことから、PSM-NVFF はリストア動作に高いロバスト性があると言える。

### (c) 無駄な書き込みエネルギーの発生

ストア動作では選択トランジスタをオンにし、MTJ 素子に電流を流すことで書き込みを行うことができる。しかし、PSM-NVFF では前回 MTJ に書き込みを行ったデータを記憶しておくことができない。そのため、図 5.3.6 のように、ストア動作を行っても MTJ の抵抗値が同じ変わらない場合が存在する。しかし、MTJ 素子の抵抗値が変化するかどうかに関係なく選択トランジスタがオンしている間はストア電流が流れてしまう。これにより、無駄なストアエネルギーが発生してしまう。不揮発性 PG ではこの MTJ 素子に書き込むために必要なストアエネルギーが、PG を行う際の非常に大きなオーバーヘッドとなるため、BET の増大に繋がってしまう。そのため、無駄なストア動作はできる限り避けるべきである。

ストア動作の流れ

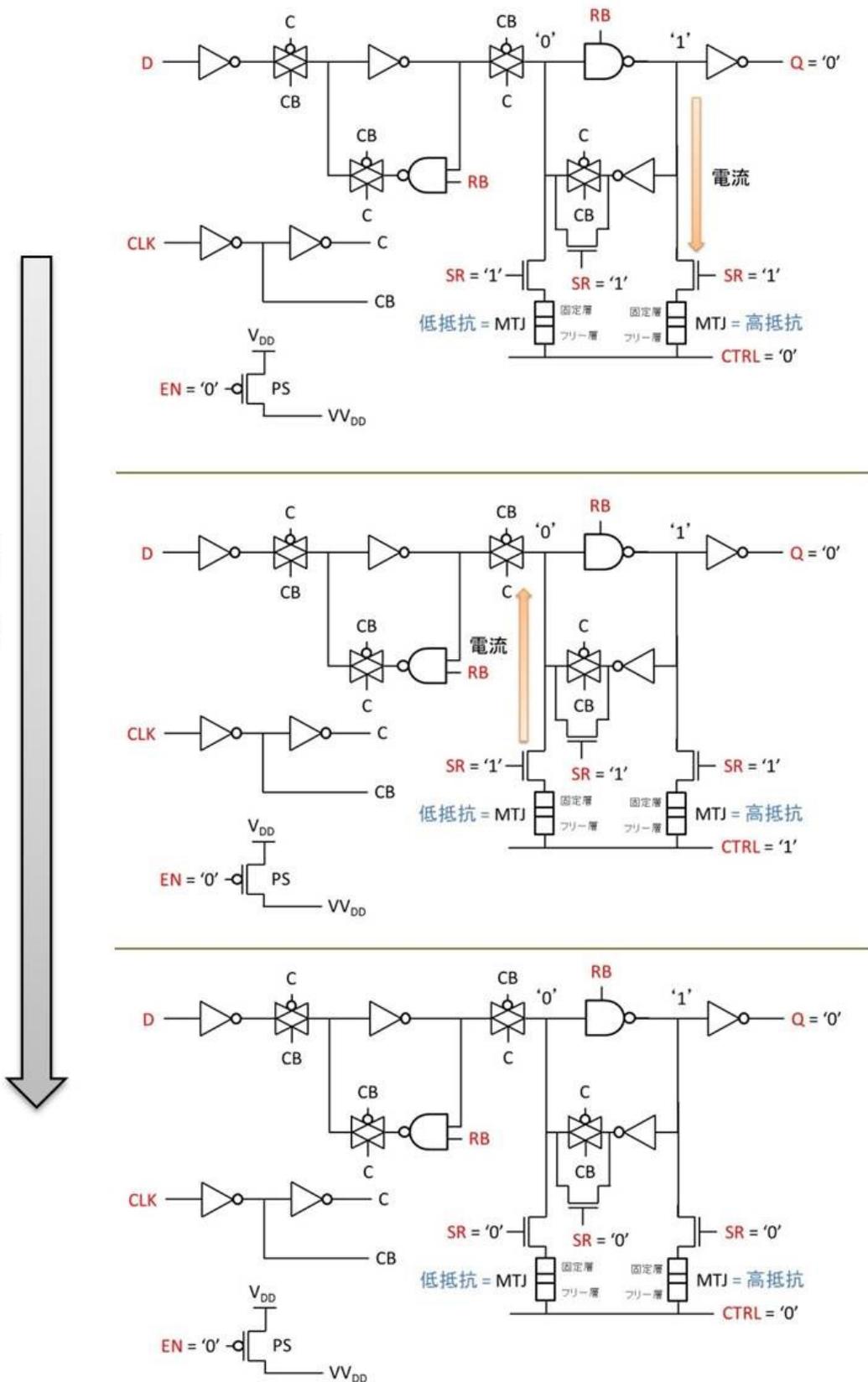


図 5.3.6 無駄なストア動作の例

## 5.4 新しい NVFF 回路の提案

5.3.2節で述べた PSM-NVFF の問題点から、著者は2種類の新しい NVFF 回路を提案する。本節では各提案回路の回路構成を示し、その優位性について述べる。さらに、次の 5.5 節にて、既存の PSM-NVFF との比較評価を行い、提案手法の性能について定量的な評価を示す。

### 5.4.1 Split-Store-Restore NVFF (SSR-NVFF)

SSR-NVFF は既存の PSM-NVFF において、重大な問題となっているストア動作によるラッチ破壊の発生を効果的に防ぎ、スレーブラッチの回路面積を削減が期待できる NVFF 回路である[57]。ヘッダ型(HDPG)およびフッタ型(FTPG)の SSR-NVFF の回路図を図 5.4.1 に示す。また、図 5.4.2 は SSR-NVFF の制御シーケンスである。SSR-NVFF は既存の PSM-NVFF の回路図を基にし、スレーブラッチの N1、N2 のノードをそれぞれインバータで受け、その出力を左右反対の選択トランジスタおよび MTJ 素子へ接続している。SSR-NVFF でのストア動作とリストア動作を以下に示す。

#### (a) ストア動作

ストア動作時の制御について図 5.4.3 に示す。ストア動作ではまず、SR1 信号を '0' にし、ストア動作の選択トランジスタ (TR1、TR2) をオンにする。また CTRL 線は '0' のため、TR2 から CTRL 線の方向に電流が流れる。これにより、MTJ 素子の抵抗値は高抵抗に変化する。次に、CTRL 線の電圧を '1' にする。CTRL 線か TR1 の方向に電流が流れ、MTJ 素子の抵抗値は低抵抗になる。最後に TR1、TR2 をオフにすることでストア動作を行うことができる。また、次に PS をオフにすることで、PSM-NVFF と同じ制御でスリープ動作を行える。

さらに、選択トランジスタと MTJ を組み合わせた際のストア電流の流れやすさを図 5.4.4 に示す。PSM-NVFF に対し、SSR-NVFF ではストア動作の選択トランジスタとして pMOS を用いている。これにより、PSM-NVFF に対し、ストア電流の流れやすい方向が逆になる。つまり、より多くの電流が必要な IcP→AP のストア電流は、選択トランジスタと MTJ を流れやすい方向の電流となる。よって、PSM-NVFF に比べ、SSR-NVFF ではストア動作の選択トランジスタのサイズを小さくすることができる。

また、ストア電流はスレーブラッチの N1、N2 ノードをゲート受けしたインバータと CTRL 線の間で流れる。そのため、スレーブラッチのトランジスタサイズに依らず、ラッチ破壊が起こることはない。これにより、スレーブラッチのトランジスタサイズは通常動作に必要なサイズがあれば問題ない。

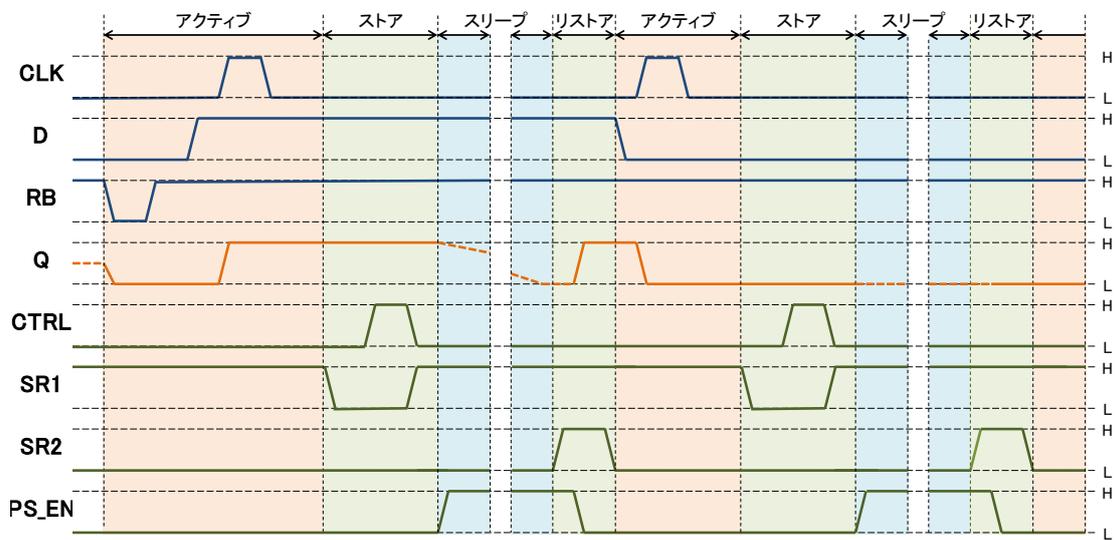
## (b) リストア動作

リストア動作では SR2 の信号を '0' から '1' にすることで、nMOS の選択トランジスタ (TR3、TR4、TR5) をオンにする。また、CTRL 線は '0' にしておくため、PSM-NVFF と同じく CTRL 線から nMOS の選択トランジスタを通して、スレーブラッチのループ部へ電流が流れる。また、MTJ 素子の抵抗値の差を利用して、スレーブラッチの N1、N2 のノードの電圧はストア動作前の状態へと復帰することができる。

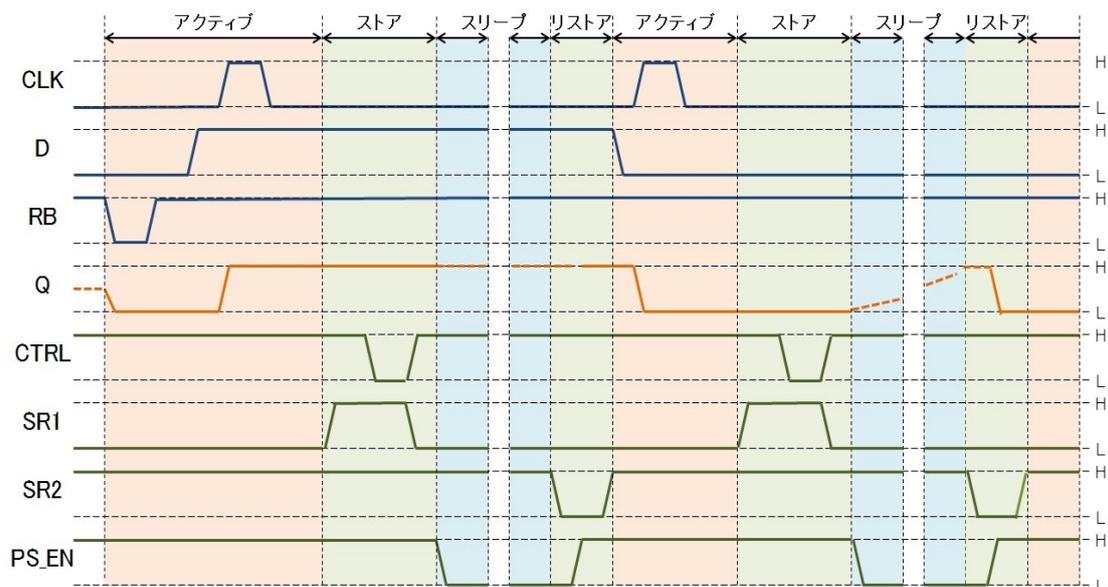
また、5.3.2 節にてリストア動作のロバスト性を述べたように、リストア動作のロバスト性が高いため、リストア動作の選択トランジスタはストア動作の選択トランジスタよりも小さくすることができる。これにより、TR3、TR4、TR5 のトランジスタサイズは小さいものとなる。

以上のことから、SSR-NVFF では、リストア動作だけでなくストア動作でも安定した動作を実現することができる。また、スレーブラッチおよび選択トランジスタのサイズを小さくできることから、PSM-NVFF と比べトランジスタ数は多くなるものの、回路面積では SSR-NVFF の方が小さくなることが期待できる。さらに、ストア電流の経路は Reset 機能付による NAND セルを経由しないため、インバータセルと選択トランジスタのサイジングだけで設計を行うことができるため、設計コストを小さく抑えることもできる。





(a) HDPG



(b) FTPG

図 5.4.2 SSR-NVFF の制御シーケンス

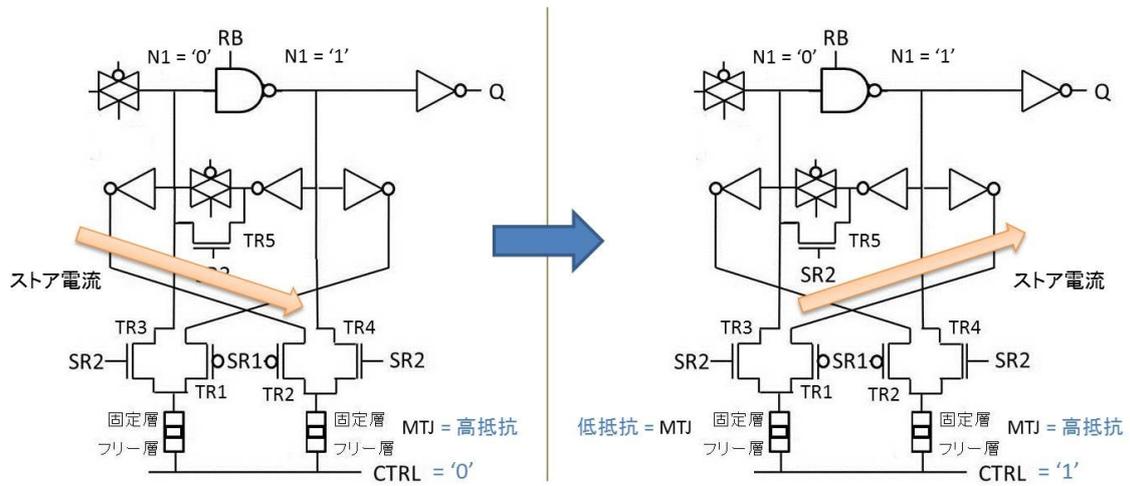


図 5.4.3 SSR-NVFF のストア動作

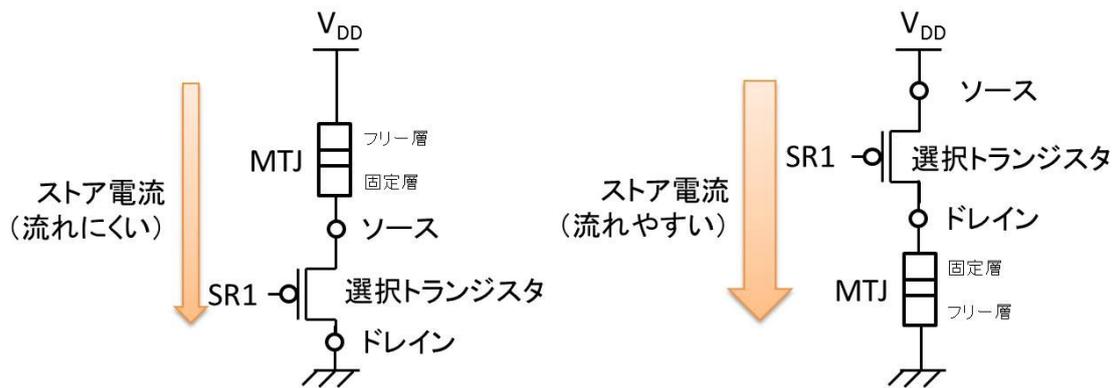


図 5.4.4 SSR-NVFF のストア電流の向きと流れやすさ

## 5.4.2 Selective Storing NVFF (SS-NVFF)

既存の PSM-NVFF では、PG が実行されるたびにストア動作が実行されてしまうという問題が挙げられる。PG によるリーク削減効果に対し、MTJ 素子へのストアエネルギーは非常に大きなオーバーヘッドとなっている。これにより、MTJ 素子へのストアエネルギーが NVPG を適用したときの BET の大きな要素となってしまう。よって、MTJ 素子への書き込み頻度を抑えることで、NVFF を用いた不揮発性 PG におけるエネルギーオーバーヘッドを大幅に削減することができ、BET を短くすることが期待できる。

スレーブラッチの回路状態が前回 PG を行った時と同じ場合には、MTJ 素子への書き込みを行う必要はない。そのため、スレーブラッチに保持しているデータと最後に MTJ 素子に書き込んだデータを比較し、異なっていた場合にのみ、ストア動作を自動で実行する回路を提案した (図 5.4.5) [50]。本提案の回路を Selective Storing NVFF (SS-NVFF) と名付けた。SS-NVFF における制御シーケンスを図 5.4.6 に示す。SS-NVFF では、不揮発性 PG を実行するために以下の動作を行う必要がある

- 初期化動作
- アクティブ動作
- ストア動作
- スリープ動作
- リストア動作

アクティブ動作およびスリープ動作は、PSM-NVFF と同じ制御で動作させることができる。その他の初期化動作、ストア動作、リストア動作について以下に詳しく説明する。

### (a) 初期化動作

PSM-NVFF に対し、SS-NVFF では FF を初期化するのと同じタイミングで、MTJ の書き込みを 1 度行う必要がある。そのため、SR1 と SR2 が接続されている選択トランジスタをオンにし、CTRL 信号を '0' から '1' にすることで左右の MTJ 素子へ書き込みを行う (図 5.4.7)。さらに、Save 信号を '1' にすることで、スレーブラッチで保持しているデータを選択回路内のラッチへ格納する。これにより、最初のデータを選択回路が記憶することができる。初期化動作を実行した後は、PSM-NVFF と同様にアクティブ動作を実行できる。

### (b) ストア動作

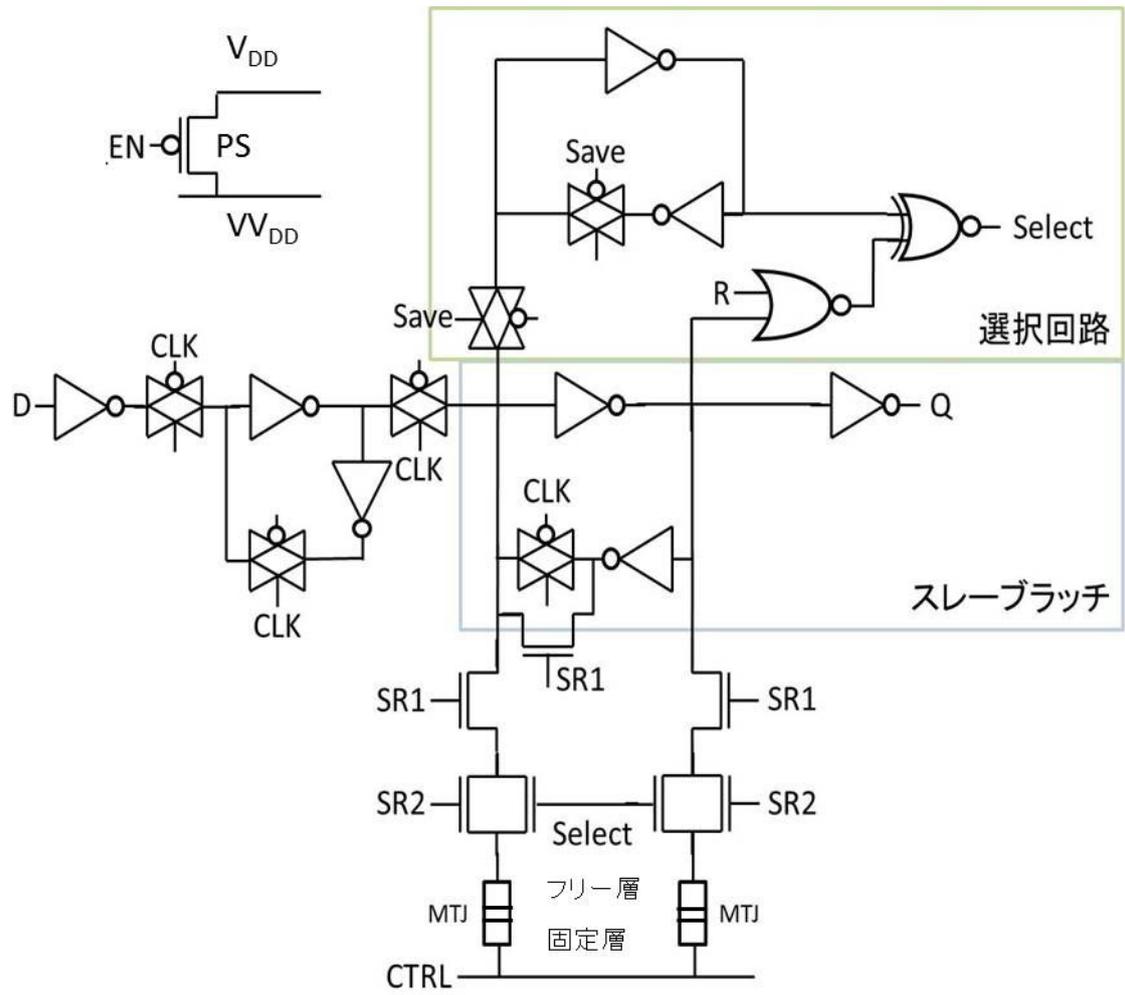
ストア動作時には、SR1 信号が接続されている選択トランジスタのみをオンにし、SR2 が接続されているトランジスタはオフにする。このとき、選択回路はスレーブラッチのデータと選択回路内のラッチのデータを比較する。お互いのデータが異なっている場合には、

Select 信号は '1' となり、図 5.4.7 と同じようにストア電流が流れ、MTJ 素子への書き込みが行われる。しかし、スレーブラッチと選択回路で同じデータを保持していた場合には、Select 信号は '0' のままであり、ストア電流は流れない。これにより、ストア動作前とストア動作後で MTJ 素子の抵抗が同じになるようなストア電流が流れることを防ぐことができる。ストア動作後は SS-NVFF 回路全体を PG することが可能となる。

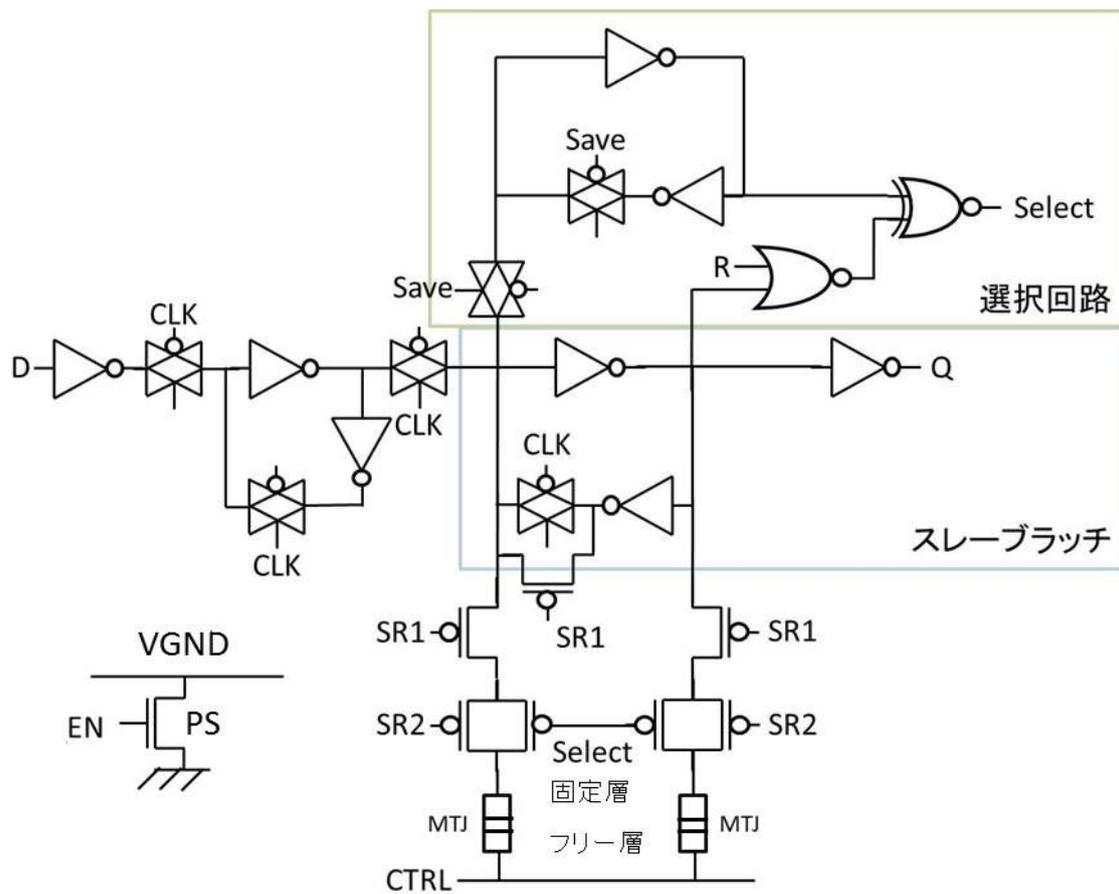
### (c) リストア動作

リストア動作では、SR1、SR2 に接続されているトランジスタをオンにすることで、PSM-NVFF と同じ原理で、回路内のデータを読み出すことができる。しかし、PSM-NVFF と同じリストア動作を行った後、図 5.4.8 で示した動作を行い、スレーブラッチのデータを選択回路部へ格納する必要がある。これにより、最後に MTJ 素子へ書き込みを行ったデータを選択回路は保持することができる。

よって、SS-NVFF では、MTJ 素子への書き込みの選択をセル単体だけで自動で行うことが可能である。しかし一方で、選択回路にインバータや伝送ゲートの他に、Exclusive NOR セルを用いているため、PSM-NVFF に比べてトランジスタ数が増えてしまっている。そのため、不揮発性 PG を行う場合には、PG 適用回路全体の面積に対して D-FF のセル数が少ない回路や、D-FF のデータが書き換わる頻度が低い回路に適用することが望ましいと考えられる。



(a) HDPG



(b) FTPG

図 5.4.5 SS-NVFF 回路図

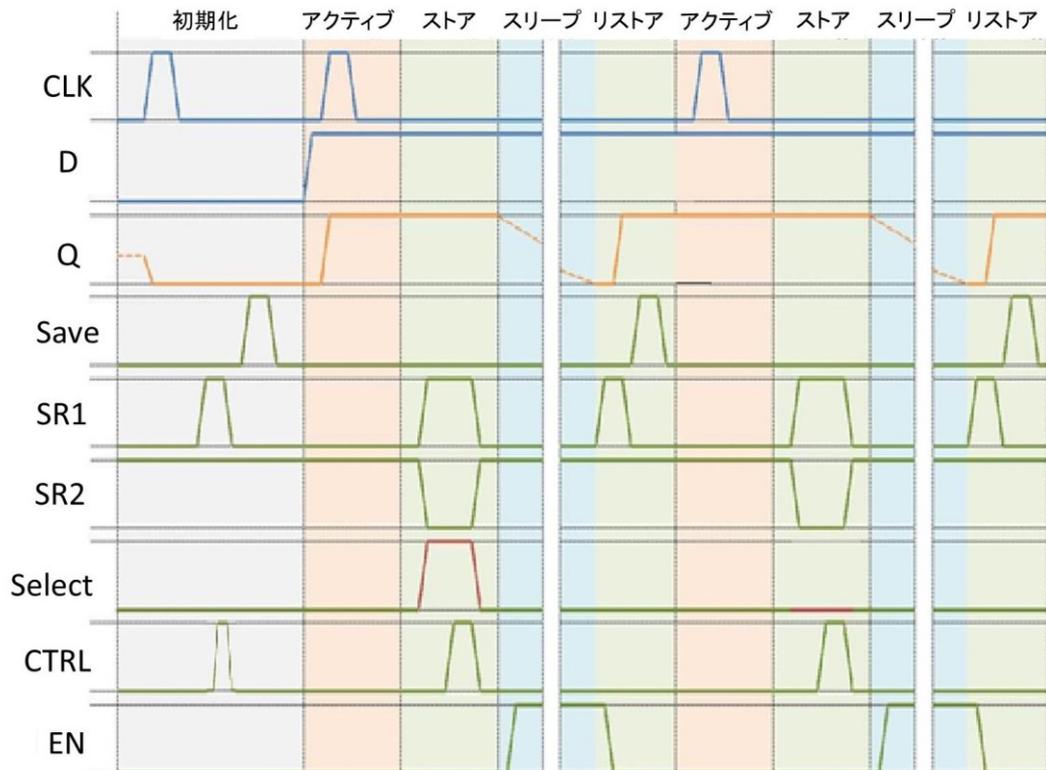


図 5.4.6 SS-NVFF の制御シーケンス

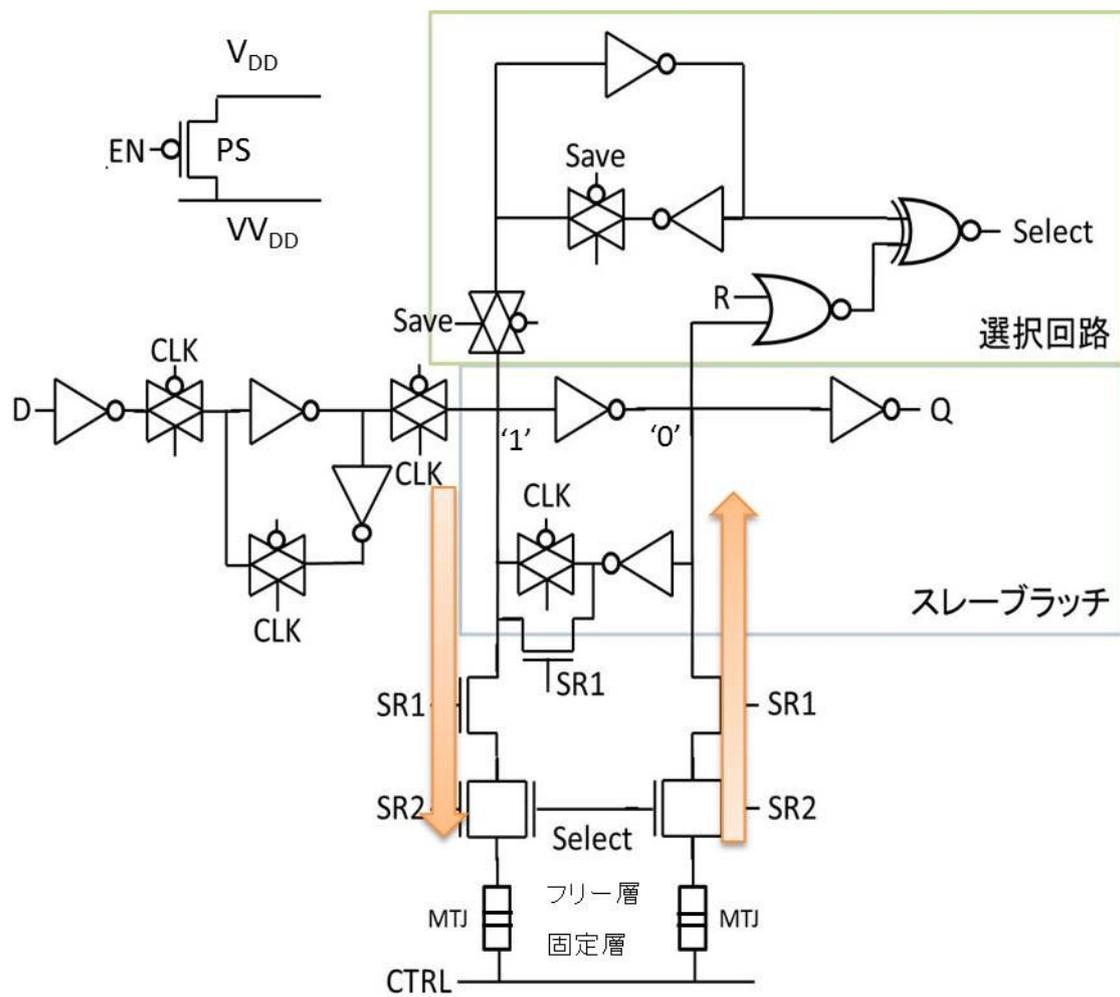


図 5.4.7 MTJ へ書き込み動作

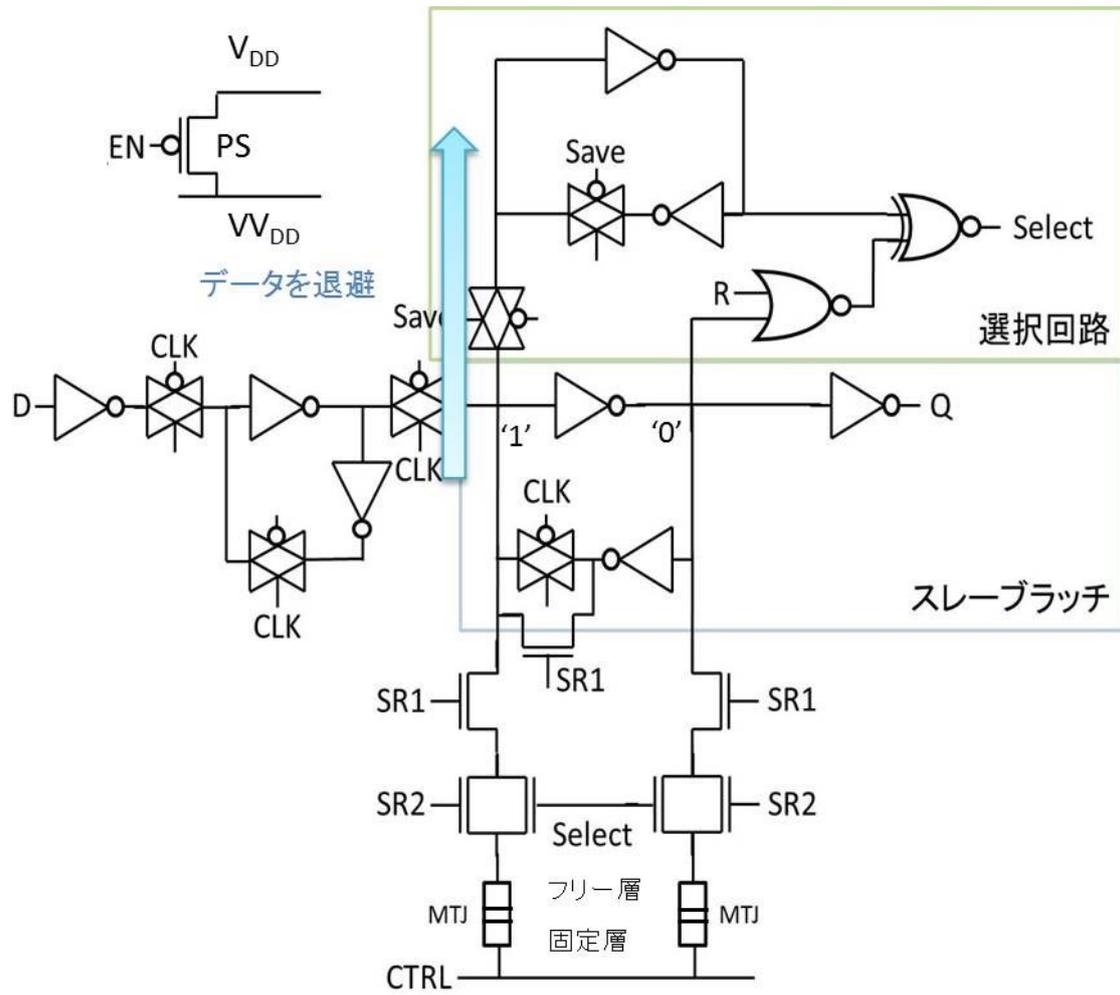


図 5.4.8 選択回路へデータを格納

## 5.5 評価

本節では、著者らが提案した2つのNVFFと従来手法であるPSM-NVFFを、実際に65nmプロセスを用いて設計し、回路シミュレータ(HSPICE)を用いて比較評価を行った。また、電源電圧は1.2Vとした。さらに、通常のD-FFと比較し、D-FFをNVFFに置き換えた際の面積および動作速度へのオーバーヘッドを求めた。

### 5.5.1 SSR-NVFF

上記で述べたように、提案手法としてSSR-NVFFの設計を行い既存手法のPSM-NVFFとの比較を行った。まず、PSM-NVFFとSSR-NVFFにおいて、ストア電流 $I_{cP \rightarrow AP}$ と $I_{cAP \rightarrow P}$ を十分に確保できるように選択トランジスタおよびスレーブラッチのトランジスタのサイジングを行った。さらに、PSM-NVFFとSSR-NVFFで、通常のD-FFのスレーブラッチのサイズから、ラッチ破壊が起こらなくなるスレーブラッチのサイズを求めた。

#### (a) ストア動作の不良率

PSM-NVFFとSSR-NVFFのストア動作時のラッチ破壊への耐性を評価するため、スレーブラッチのトランジスタサイズとストア動作の不良率の関係性を求めた。PSM-NVFFとSSR-NVFFに対し、選択トランジスタサイジングは、ストア電流 $I_{cP \rightarrow AP}$ と $I_{cAP \rightarrow P}$ を十分に確保できるようにした。また、ストア動作時のラッチ破壊の起こりやすさを評価するために、トランジスタしきい値電圧に対しプロセスばらつきが生じると想定し、モンテカルロ・シミュレーションを1000回行った。また、標準偏差 $\sigma$ に対し、 $3\sigma$ のとき標準のしきい値電圧から $\pm 10\%$ 変化すると想定した。PSM-NVFFとSSR-NVFFに対し、スレーブラッチのトランジスタサイズを通常のD-FFのサイズから大きくした際のストア動作の不良率を図5.5.1に示す。スレーブラッチのトランジスタサイズは、通常のD-FFのトランジスタサイズをもとに正規化を行っている。また、ストア動作の不良率は各スレーブラッチのトランジスタサイズで、1000回のシミュレーションのうち、どれだけラッチ破壊が発生したかを示したものである。

SSR-NVFFでは、ゲート受けしたインバータからMTJ素子へ電流を流しているため、通常のD-FFのスレーブラッチと同じトランジスタサイズを用いてもラッチ破壊は起こらなかった。一方、PSM-NVFFでは、5.3.2節で述べたように、スレーブラッチのトランジスタサイズを小さくすることで、ラッチ破壊の頻度は高くなる傾向を示した。また、FTPGに比べHDPGの方が、ラッチ破壊が起こりやすいという結果となった。これは、HDPGがpMOSをPSに用いており、nMOSに比べpMOSのトランジスタのオン抵抗が高いためだ

と考えられる。また、今回のシミュレーションでラッチ破壊が起こらないような PSM-NVFF を設計するためには、スレーブラッチのトランジスタサイズを通常の D-FF の 6 倍のサイズまで大きくしなければならないことがわかった。

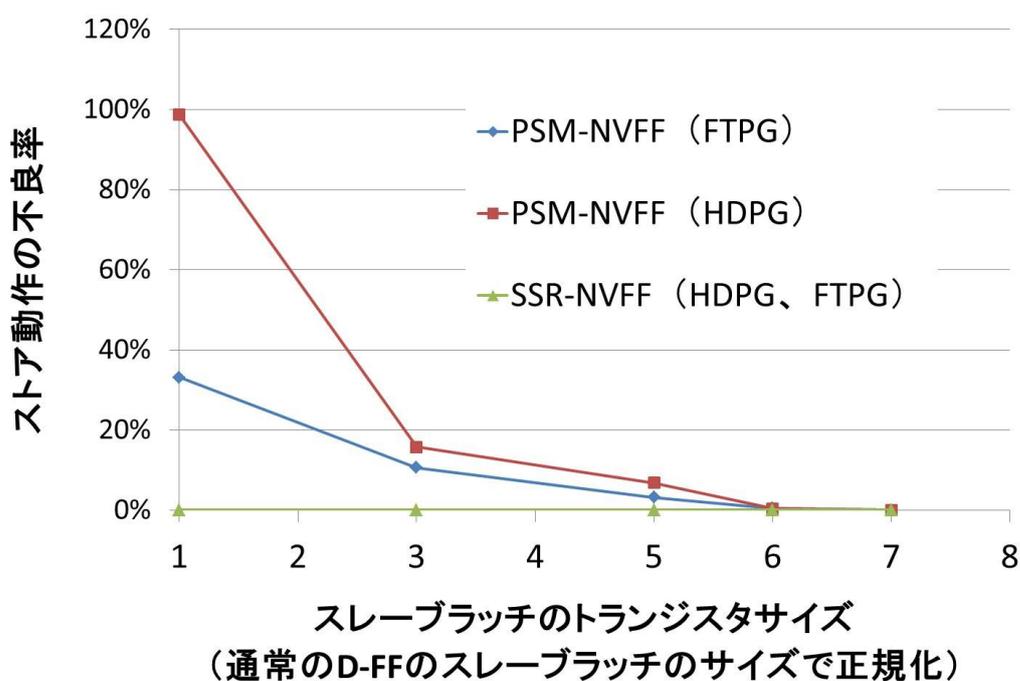


図 5.5.1 プロセスばらつきにおけるストア動作の不良率

## (b) ストア動作時のストア電流

PSM-NVFF と SSR-NVFF において、5.3.2 節および 5.4.1 節で述べたストア動作を行った際の MTJ 素子に流れるストア電流を、回路シミュレーションを用いて求めた。図 5.5.2 に HDPG および FTPG を適用した PSM-NVFF と SSR-NVFF のシミュレーション結果を示す。また、SSR-NVFF のスレーブラッチのトランジスタサイズは通常の D-FF と同程度とし、PSM-NVFF のスレーブラッチのトランジスタサイズは、ストア動作の不良動作が発生しないよう、通常の D-FF の 6 倍のサイズを想定した。

PSM-NVFF では AP→P への書き込みの方が、ストア電流が流れやすい。そのため、IcP →AP を確保するために選択トランジスタのサイズを大きくすると、AP→P への書き込み時に必要以上の電流が流れてしまう。これは、無駄なエネルギーが消費されるだけでなく、MTJ 素子に高い電圧が印加されることで、MTJ 素子のバリア層が破壊される原因になりか

ねない。一方、SSR-NVFF ではストア動作時に使用する選択トランジスタを変えることによって、 $I_{cAP \rightarrow P}$  よりも  $I_{cP \rightarrow AP}$  の方向の電流が流れやすくなっている。そのため、より多くの電流が必要な  $I_{cAP \rightarrow P}$  の電流を確保するように、選択トランジスタのサイジングを行っても AP→P 方向の書き込み時に、無駄なストア電流が流れるのを防ぐことができた。

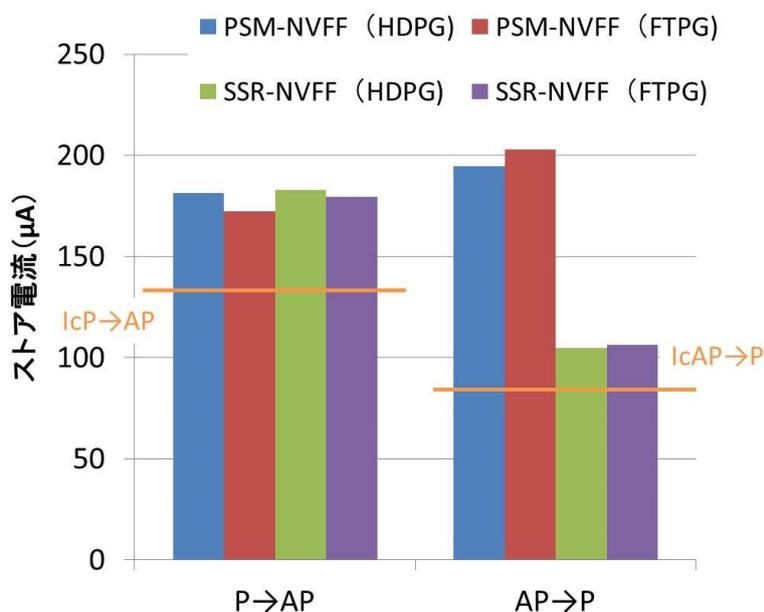


図 5.5.2 PSM-NVFF と SSR-NVFF のストア電流比較

### (c) 回路性能の比較

さらに PSM-NVFF と SSR-NVFF の回路性能の比較を行うために、それぞれの NVFF の面積およびクロックが立ち上がってから出力 Q が遷移するまでの時間 (C-Q 遅延時間)、ダイナミックエネルギー、ストアエネルギー、リストアエネルギーのシミュレーション評価を行った (表 5.5.1)。ダイナミックエネルギーは、出力 Q が '1' から '0' に遷移するときと、'0' から '1' へ遷移するときの消費エネルギーの平均をとった。またストアエネルギー、リストアエネルギーは、ストア動作、リストア動作が十分に行える時間として 50ns の間、選択トランジスタをオンにした時の消費エネルギーをシミュレーションで求めたものである。

回路面積は PSM-NVFF と比べて、SSR-NVFF が 55%にまで削減できる結果となった。これは、回路構成に必要なトランジスタ数は SSR-NVFF の方が多いが、スレーブラッチおよび選択制御回路のトランジスタサイズを効果的に削減できたためである。また、C-Q 遅延時間も 18%削減することができた。これもまた、スレーブラッチのトランジスタサイズを抑えたことにより、回路内の寄生容量が減ったためだと思われる。さらに、ストア・リストアエネルギーも PSM-NVFF と比べ削減できることがわかった。ストアエネルギーの削減は AP→P 方向の書き込みを行う際の無駄な電流を削減できたことによる影響だと考えられる。また、SSR-NVFF ではストアとリストアでそれぞれ別の選択トランジスタを用いている。そのため、リストアに用いる選択回路はリストアが実行できるトランジスタサイズがあれば十分なため、PSM-NVFF に比べ消費エネルギーを削減することができた。

表 5.5.1 PSM-NVFF と SSR-NVFF の回路性能比較

	PSM-NVFF (A)	SSR-NVFF (B)	比率 (B/A)
面積 [ $\mu\text{m}^2$ ]	172×203	180×106	0.55
C-Q 遅延時間 [ps]	280	230	0.82
ダイナミックエネルギー [pJ]	0.17	0.14	0.84
ストアエネルギー [pJ]	9.02	6.13	0.68
リストアエネルギー [pJ]	3.84	2.23	0.58

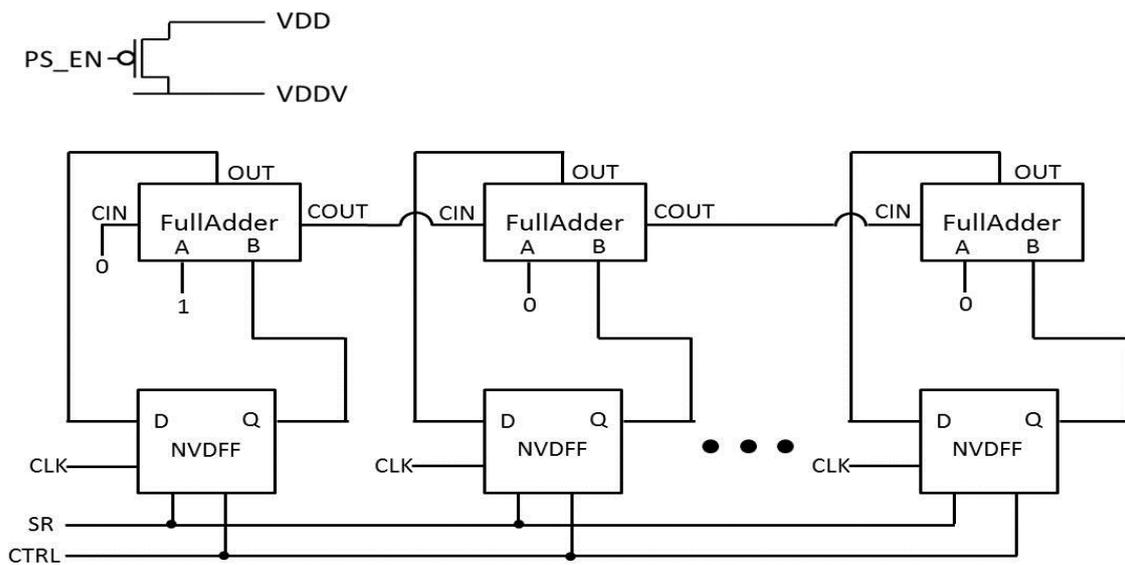
## 5.5.2 SS-NVFF

SS-NVFF の評価を行うため、16bit カウンタ回路と ISCAS'89 ベンチマークから 2 つの回路を採択し、シミュレーションを行った。SS-NVFF の設計では 65nm プロセスを想定しており、電源電圧は 1.2V としている。PSM-NVFF と SS-NVFF を適用した 16bit カウンタ回路の回路図を図 5.5.3 に示す。

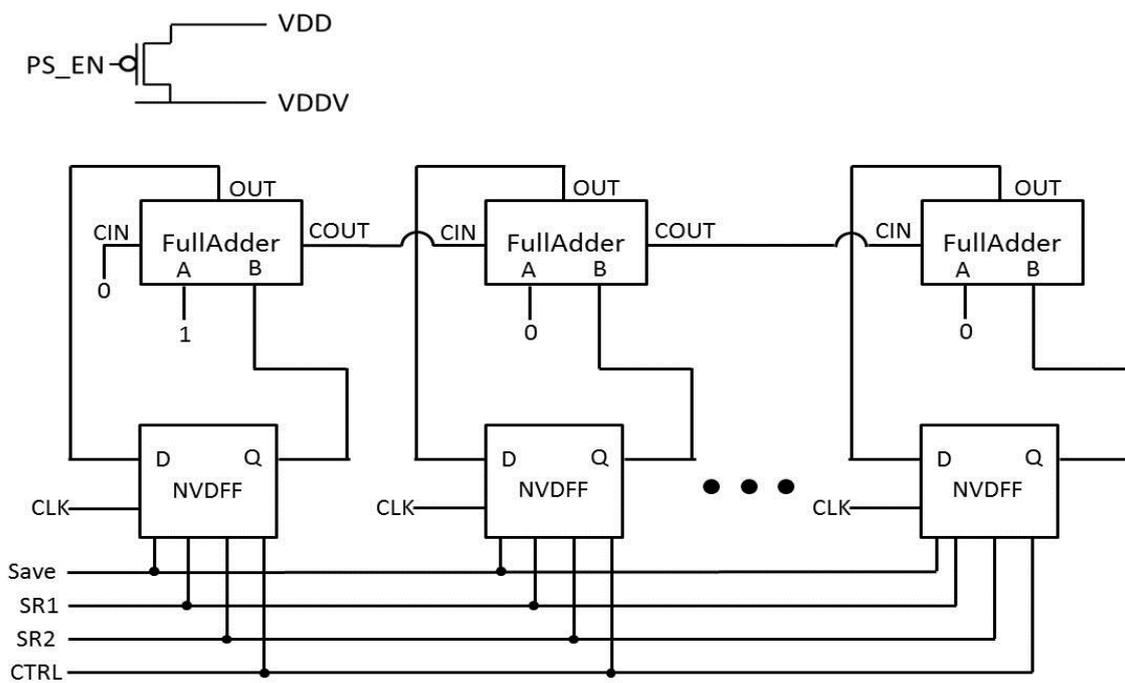
また、表 5.5.2 は SS-NVFF のシミュレーション結果を示す。ダイナミックエネルギーは出力が '1' から '0' へ遷移したときと、'0' から '1' へ遷移した時の平均の消費エネルギーである。また、ストアエネルギーはストア動作を行うため、選択トランジスタ (SR、SR1) を 50ns の間オンにし、ストア動作を行っている時に消費されるエネルギーを評価した。

ダイナミックエネルギー、ストアエネルギー、BET は PSM-NVFF の値で正規化している。16bit カウンタ回路では、図 4.5.3 に示すように、16 個の NVFF が使用されており、FullAdder セル 16 個と合わせて 32 個のセルのみで構成されている。そのため、回路全体に対する FF のセル数の割合は 50%となっている。一方、ISCAS'89 のベンチマーク回路の s1494 と s5378 ではそれぞれ NVFF が 6 個と 179 個使用されている。しかし、s5387 では、その他に使用しているセル数も多いため、FF 数の割合はそれぞれ 6%と 9%となっている。

ダイナミックエネルギーは、PSM-NVFF と比べ、SS-NVFF では 2~13%増加する結果となった。これは、ストア動作を行う必要があるかを自動で判断するために追加した回路が大きくなったため、セル内のトランジスタ数や負荷容量が増加したためである。しかし、回路全体のセル数に対する FF の割合が低い s1494 の回路ではこのダイナミックエネルギーの増加を 2%に抑えられている。また、カウンタ回路では回路の半数を FF が占めているが、ダイナミックエネルギーの増加は 7%となっている。これはカウンタ回路の FF のスイッチング確率が低いためだと考えられる。一方、ストアエネルギーは 15~29%にまで削減することができた。無駄な MTJ の書き込みを抑制することで、3 つの回路すべてで MTJ 書き込み頻度を 30%以下に抑えられていることを示している。これにより、PSM-NVFF に対して、SS-NVFF では PG を適用した回路の BET を 14~28%にまで削減することができた。



(a) PSM-NVFF



(b) SS-NVFF

図 4.5.3 16bit カウンタ回路

表 5.5.2 SSNVFF の消費エネルギーおよび BET (PSM-NVFF で正規化)

ベンチマーク 回路	回路内の FF 数	全セル数の内 FF の割合	ダイナミック エネルギー	ストア エネルギー	BET
16bit カウンタ	16	50%	1.07	0.26	0.26
s1494	6	6%	1.02	0.15	0.14
s5378	179	9%	1.13	0.29	0.28

## 5.6 考察とまとめ

第 5 章では、マイクロプロセッサなどで必ず必要であるレジスタ回路を含んだ回路全体に PG を適用するための手法について述べ、MTJ 素子と呼ばれる磁気素子を使用した不揮発性 PG に着目し、研究を行った。

従来手法として提案されていた、PSM-NVFF を実際に設計し、ウェハ上で測定を行うことで、この NVFF 回路の問題点を新たに明らかにした。さらに、それらの問題点を解決する新しい NVFF 回路を 2 つ提案した。

ストア動作のロバスト性を高める NVFF では、通常の D-FF のスレーブラッチのサイズを用いてもラッチ破壊が発生しないことを示せた。さらに、トランジスタ数は増えるが、スレーブラッチおよび選択トランジスタのサイズを効果的に小さくできるため、全体の回路面積を 55%まで削減できることが分かった。さらに、効果的な電流経路でストア動作を行えるため、ストアエネルギーも 68%まで削減することができた。

MTJ 素子が同じ抵抗値のままになってしまう無駄なストア動作を防ぐ NVFF 回路では、NVFF 回路のトランジスタ数の増加にともない、動作エネルギーが 2~13%ほど増加することが分かった。しかし一方で、データを記憶するための消費エネルギーは 15~29%にまで減らすことができた。

従来手法である PSM-NVFF の問題点を大きく 2 つ挙げ、新しく提案した NVFF でそれぞれの問題を解決することが可能であることを示せた。しかし、今回挙げた問題点は、不揮発性 PG において、常に同時に生じている問題だと考えられる。そのため、2 つの提案手法の回路の機能を統合した、新しい NVFF 回路が強く求められると考えられる。

## 第5章 考察

---

第4章では、細粒度 PG を適用することにより、マイクロプロセッサが動作をしているランタイムでもリークエネルギーを削減できることを示した。これにより、細粒度 PG を用いることで動作中のリーク電力の削減が期待できる。しかし、4.4 節、4.5 節でも述べたように、動作周波数によってエネルギー削減効果が大きく変わる。今回は評価対象として動作周波数が 200MHz のマイクロプロセッサを想定した。しかし、動作周波数が 1GHz 以上の CPU などの回路では細粒度 PG による消費エネルギーの削減が期待できないと考えられる。一方で、動作周波数が 1MHz 以下でも十分なセンサ回路や医療用デバイスなどでは、リーク電力の削減効果が高まると考えられる。また、微細化によりトランジスタサイズの小さいプロセスではリーク電流が多くなる。このような場合にも、細粒度 PG を適用することによるリーク電力を削減できると考えられる。

また、チップ上の温度が高くなる場合も、リーク電流が多くなる。このため、チップ上の温度をモニタし、温度が高くなった場合にのみ、細粒度 PG を有効にすることで、効果的に PG を制御できると考えられる。

さらに、今回の細粒度 PG は、演算器のみに適用した。これは、演算器回路が組み合わせ回路のみで構成されており、PG を行う前後で保持しておくデータが無いためである。しかし、レジスタを含む回路にも細粒度 PG を適用することでさらなるリーク電力が削減できると考えられる。しかし、レジスタは PG を行うことで保持しているデータが失われてしまうため、レジスタを不揮発化するなどの工夫が必要になる。また、不揮発化したレジスタを含む回路を細粒度 PG するには、データを保持するために必要なエネルギーオーバーヘッドも考慮する必要がある。このエネルギーオーバーヘッドが大きい場合、消費エネルギーが増大する可能性がある。そのため、エネルギーオーバーヘッドを小さくする工夫や、オーバーヘッドを考慮したスリープ制御が必要になると考えられる。

また、第5章では、新しい不揮発性 D-FF (NVFF) を提案し、マイクロプロセッサなどにおいて必要不可欠な D-FF を NVFF に置き換えることで、レジスタを含む回路でも不揮発性 PG を効果的に実行できることを示した。また、新しい NVFF が高いロバスト性を示すことが分かった。しかし、D-FF は誤り検出訂正機能 (ECC) を持たないため、誤ったデータを保持してしまうと回路の不良動作に繋がってしまう。そのため、データ保持のためにマージンを持った回路設計が必要となる。しかし、正しいデータが記憶されているか確認する (Verify) 機能が実現できれば、正しいデータを保持できなかったレジスタにのみ、再度書き込みを行うことができる。これにより、回路設計でのマージンを無くすことができ、無駄な書き込みエネルギーを抑制できるため、回路面積や消費エネルギーのオーバー

ヘッドを減らせると考えられる。

さらに、回路面積に対しレジスタの数が多い回路では、NVFF のデータを保持するときのエネルギーオーバーヘッドが大きくなるため、不揮発性 PG の効果が小さくなる。そのため、エネルギーオーバーヘッド小さくするために、回路に含まれるレジスタのうち、記憶すべきレジスタとデータが失われても不良動作に繋がらないレジスタを取捨選択する方法が考えられる。このためには、システムレベル、アーキテクチャレベルでの回路設計が必要になる。

また、NVFF を用いて不揮発性 PG を細粒度に適用した場合、データを保持するエネルギーオーバーヘッドが問題となる。このエネルギーオーバーヘッドが削減するリークエネルギーに対し大きくなってしまうと損益分岐時間 (BET) が長くなるためである。これにより、不揮発性 PG を適用できるのは、回路が使われず待機している時間が長い回路に限られてしまう。しかし、このデータを保持するためのエネルギーオーバーヘッドは MTJ を書き換えるために必要なエネルギーが主な要因である。そのため、回路設計だけでこのエネルギーオーバーヘッドを小さくすることは困難であると考えられる。そのため、MTJ 素子に用いられる素材などを工夫することにより、書き込みエネルギーを大きく削減する工夫などが強く求められる。MTJ 素子への書き込みエネルギーが大幅に削減できた場合、PG を行っている時間が短くとも消費エネルギーの削減が期待できる。これにより、より多くの機会でリーク電力を削減することができると予想される。

## 第6章 結論

---

本論文では、トランジスタの微細化とともに大きな課題となっているリーク電力削減技術について概説した。さらに、マルチコアのマイクロプロセッサやモバイル向けの LSI などへの適用が期待されることから、PG 技術に着目し、PG を適用する回路の粒度を下げることで PG を行う機会をより多く得る手法と、PG による記憶回路のデータの破壊を防ぎ、より多くの領域で PG を実行する手法の 2 つの観点から、さらなるリーク電力削減効果を示した。

PG を適用する領域をコアレベルではなく、セルレベルで行う細粒度 PG を実装したマイクロプロセッサを開発し、ランタイムパワーゲーティングの有効性を示した。さらに、PG の粒度を細かくすることで生じる問題を取り上げ、従来手法では解決できない問題点を明確にした。また、その問題を解決するための、新しい PG の制御手法を提案した。正確なシミュレーション評価を行うため、レイアウトにより回路設計を行い、対象回路の配線抵抗・容量の抽出を行った。提案手法と従来手法の性能の評価を行うため、細粒度 PG を実装したマイクロプロセッサ上でアプリケーションを実行した際の消費エネルギーを評価した。これにより、提案手法が従来手法に比べ、非常に少ない消費エネルギーで PG を制御することができ、なおかつ効率的に PG によるリークエネルギーの削減が行えることが分かった。

さらに、提案手法と従来手法を用いた場合の消費エネルギーのモデル化を行い、各手法の特徴を定量的に評価した。電源電圧やリーク電流、回路の活性化率など回路環境による各手法への影響について評価し、さらに、プロセスばらつきによる影響についても調査を行った。以上のことから、従来手法に対する提案手法の有用性について示した。

PG を行っても保持するデータが失わない記憶回路技術について述べ、PG を適用する領域を広げることで得られるリーク削減効果について議論し、磁気素子を用いた不揮発性フリップフロップの有用性を示した。また不揮発性フリップフロップとして提案されている回路を取り上げ、レイアウト設計を行うことで、シリコン上での測定を行った。さらに、測定から得られた知見をもとに、従来手法のロバスト性について議論し、PG を適用する際の問題点を指摘した。この問題を解決するために、PG に対する記憶データにロバスト性を持つ回路と、データを記憶するために無駄なエネルギーが消費されるのを防ぐ回路の 2 つを提案した。

記憶データにロバスト性を持つ不揮発性フリップフロップの回路では、回路のトランジスタサイズとデータを保持する際の動作不良率の関係性を示し、従来手法に対してセルの回路面積を小さくできた。さらに、データの記憶時に消費されるエネルギーも削減できる

ことが分かった。

また、不揮発性フリップフロップがすでに同じデータを保持している場合、データの書き込みを行わないことで無駄なエネルギーの消費エネルギーを抑える回路を考案した。提案手法と従来手法の回路を 16bit カウンタ回路とベンチマーク回路に適用し、PG を行った際の性能についてシミュレーションで評価した。従来手法に比べ、提案手法は回路面積が増大してしい、動作エネルギーが 2~13%ほど増加することが分かった。一方で、データを記憶するための消費エネルギーは 15~29%にまで減らすことができた。

## 第7章 今後の課題

---

細粒度 PG を用いたマクロプロセッサの低消費電力設計および、スリープ制御については、以下の課題が残されている。

既存のスリープ制御および本研究で提案したスリープ制御手法の評価は、マイクロプロセッサでのアプリケーション実行を想定したシミュレーションで行っている。これらの手法を適用したマイクロプロセッサを設計し、実装した上で、実チップによる測定が必要である。

不揮発性フリップフロップを用いた不揮発性 PG 回路の設計については、以下の課題が残されている。

既存手法の不揮発性フリップフロップはシリコン上での測定を行ったが、本論文で提案した不揮発性フリップフロップは未実装であり、シミュレーションでの評価を行っている。MTJ 素子やプロセスのトランジスタのばらつきによる影響が考えられるため、シリコン上で測定し、評価する必要がある。さらに、不揮発性フリップフロップを用いたマイクロプロセッサを設計し、実チップでの測定による評価が求められる。また、マイクロプロセッサへ不揮発性フリップフロップを用いた場合、MTJ 素子への書き込みエネルギーがパワーゲーティングのオーバーヘッドとして問題になると考えられる。そのため、書き込みエネルギーのさらなる低減化を図るための回路技術やアーキテクチャレベルでの制御技術が必要である。

さらに、マイクロプロセッサなどに対し、細粒度 PG と不揮発性 PG を両方適用することにより、マイクロプロセッサが動作時と待機時を合わせてさらなるリーク電力の削減が期待できる。

## 参考文献

---

- [1] V.Sundararajan and K.Parchi, "Low power synthesis of dual threshold voltage CMOS VLSI circuits," Proc.1999 Int.Symp.on Low Power Electronics and Design, pp.139-144, Aug.1999.
- [2] M.Hirabayashi, K.Nose, and T.Sakurai, "Design methodology and optimization strategy for dual-V<sub>th</sub> scheme using commercially available tools," Proc.2001 ISLPED, pp.283-286, Aug.2001
- [3] James W. Tschanz et al., "Adaptive Body Bias for Reducing Impacts of Die-to-Die and Within-Die Parameter Variations on Microprocessor Frequency and Leakage," IEEE Journal of Solid-State Circuits, Vol. 37, no. 11, pp. 1396-1402, Feb. 2002.
- [4] Takahiro Kuroda et al., "A 0.9V 150MHz 10mW 4mm<sup>2</sup> 2-D discrete cosine transform core processor with variable-threshold-voltage scheme," IEEE Journal of Solid-State Circuits, Vol. 31, no. 11, pp. 1770-1779, Nov. 1996.
- [5] J.Kao, S.Narendra, A.Chandrakasan, "Subthreshold Leakage Modeling and Reduction Techniques," ICCAD, 2002, pp. 141-149.
- [6] W. Hu, et al., "Godson-3b1500: A 32nm 1.35GHz 40W 172.8Gflops 8-core processor," Proc. 2013 IEEE International Solid-State Circuits Conference, pp.54-55, 2013.
- [7] A. Lungu, et al, "Dynamic power gating with quality guarantees," Proc. the 14th International Symposium on Low Power Electronics and Design, pp. 377-382, 2009.
- [8] Y. Kanno et al., "Hierarchical power distribution with 20 power domains in 90-nm low-power multi-CPU processor," Proc. 2006 IEEE International Solid-State Circuits Conference, pp.540-541, 2006.
- [9] R. Kumar and G. Hinton, "A family of 45nm IA processors", Proc. 2009 IEEE International Solid-State Circuits Conference, pp.58-59, 2009.
- [10] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand, "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," Proceedings of the IEEE, vol. 91, no. 2, pp. 305-327, 2003.
- [11] S. Narendra and A. Chandrakasan, "Leakage in Nanometer CMOS Technologies," Springer, 2006.
- [12] S.Narendra, D.Blaauw, A.Devgan, F.Najm, "Leakage issues in IC design:trends, estimation, and avoidance," Proc. o/ICCAD, 2003.

- [13] David E. Lackey, Paul Rs. Zuchowski, Juergen Koehl, "Designing Mega-ASICs in Nanotechnology", DAC, no 46.1, pp 770-775, June 2003
- [14] S. Dropsho et al., "Managing static leakage energy in microprocessor functional units," Proc. 35th Annual IEEE/ACM International Symposium on Microarchitecture, pp. 321-332, 2002.
- [15] N. Weste and D. Harris, "CMOS VLSI Design," 4th ed., Addison-Wesley, p.243, 2011
- [16] V.Sundararajan and K.Parchi, "Low power synthesis of dual threshold voltage CMOS VLSI circuits," Proc.1999 Int.Symp.on Low Power Electronics and Design, pp.139-144, Aug.1999.
- [17] A. Youssef, et al, "Dynamic standby prediction for leakage tolerant microprocessor functional units", Proc. IEEE/ACM International Symposium on Microarchitecture, pp.371-384, 2006.
- [18] 宇佐美 公良, "ゲーティング技術の最新動向", 信学技報 VLD2011-4, pp.19-24, 2011年5月.
- [19] Koichiro Ishibashi, Nobuyuki Sugii, Shiro Kamohara, Kimiyoshi Usami, Hideharu Amano, Kazutoshi Kobayashi, Cong-Kha Pham "A Perpetuum Mobile 32bit CPU on 65nm SOTB CMOS Technology with Reverse-Body-Bias Assisted Sleep Mode," Vol.E98-C No.7, pp.536-543. 2015.
- [20] 大崎 勇士, et al., "低電圧デジタル LSI のためのレベルコンバータ回路, " 電子情報通信学会技術研究報告, ICD, pp.133-138, 2010年7月
- [21] Z. Hu, et al., "Microarchitectural techniques for power gating of execution units", Proc. the 2004 International Symposium on Low Power Electronics and Design, pp.32-37, 2004.
- [22] N. Seki, et al., "A fine grain dynamic sleep control scheme in MIPS r3000", Proc. 26th IEEE International Conference on Computer Design, pp.612-617, 2008.
- [23] D. Ikebuchi, et al., "Geysers-1: A MIPS R3000 CPU core with fine grain runtime power gating," Proc. the 2009 IEEE Asian Solid-State Circuits Conference, pp.281-284, 2009.
- [24] M. Hayashikoshi, et al, "Normally-off MCU Architecture for Low-power Sensor Node," ASP-DAC 2014.
- [25] D. Flynn, "Power Gating and State Retention Applied to SOC Standby Power Management," in A. Baschirotto et al. (eds.), Frequency References, Power Management for SoC, and Smart Wireless Interfaces: Advances in Analog Circuit Design 2013, pp 209-226, Springer, 2013.

- [26] J. Hart et al., "3.6GHz 16-core SPARC SoC processor in 28nm," Proc. 2013 IEEE International Solid-State Circuits Conference (in presentation slide), pp. 48-49, 2013.
- [27] J. Koppanalil et al., "A 1.6 GHz dual-core ARM Cortex A9 implementation on a low power high-k metal gate 32nm process," Proc. 2011 International Symposium on VLSI Design, Automation and Test, pp. 1-4, 2011.
- [28] S. Sawant et al., "A 32nm Westmere-EX Xeon enterprise processor," Proc. 2011 IEEE International Solid-State Circuits Conference, pp. 74-75, 2011.
- [29] K. Usami, et al., "Adaptive Power Gating for Function Units in a Microprocessor," IEEE International Symposium on Quality Electronic Design (ISQED), pp.29-37, Mar. 2010.
- [30] M. Keating et al, "Low Power Methodology Manual For System-on-Chip Design", Springer, 2007.
- [31] Zhao, et al, "Spin-MTJ based non-volatile flip-flop," IEEE Int. conf. on Nanotechnology, 2007.
- [32] H. Li and Y. Chen, Nonvolatile Memory Design – Magnetic, Resistive and Phase Change, CRC Press, 2012.
- [33] D. Chabi, et al, "Ultra Low Power Magnetic Flip-Flop Based on Checkpointing/Power Gating and Self-Enable Mechanisms," IEEE Transactions on Circuits and Systems—I: Regular Papers, Vol. 61, No. 6, pp.1755-1765, June 2014.
- [34] S. Sugahara and J. Nitta, "Spin-transistor Electronics: An Overview and Outlook," Proc. IEEE, vol.98, no.12, December 2010.
- [35] N. Sakimura, et al, "Nonvolatile Magnetic Flip-Flop for Standby-Power-Free SoCs," IEEE Journal of Solid-State Circuits, Vol. 44, No. 8, pp.2244-2250, August 2009.
- [36] N. Sakimura, et al, "A 90nm 20MHz Fully Nonvolatile Microcontroller for Standby-Power-Critical Applications," ISSCC2014.
- [37] M. Kondo, et al., "Design and Evaluation of Fine-Grained Power-Gating for Embedded Microprocessors", Design, Automation and Test in Europe Conference and Exhibition (DATE 2014), Mar. 2014.
- [38] S. Kim, et al, "Understanding and Minimizing Ground Bounce during Mode Transition of Power Gating Structures", Proc. ISLPED, 2003, pp. 22-25.
- [39] K. Usami, M. Miyauchi, M. Kudo, K. Takagi, H. Amano, M. Namiki, M. Kondo, H. Nakamura, "Unbalanced Buffer Tree Synthesis to Suppress Ground Bounce for Fine-grain Power Gating," International Symposium on System-on-Chip, Tampere, Finland, Oct. 28-29, 2014.
- [40] K. Usami, et al., "Adaptive Power Gating for Function Units in a Microprocessor,"

- IEEE International Symposium on Quality Electronic Design (ISQED), pp.29-37, Mar. 2010.
- [41] K. Usami, et al., "On-chip Detection Methodology for Break-Even Time of Power Gated Function Units," International Symposium on Low Power Electronics and Design (ISLPED 2011), pp241-246, Aug. 2011.
- [42] K. Usami, and M. Kudo, et al., "Design and Control Methodology for Fine Grain Power Gating based on Energy Characterization and Code Profiling of Microprocessors," Proc. IEEE/ACM Asia and South Pacific Design Automation Conference (ASP-DAC'14), pp.843-848, Jan. 2014.
- [43] S. Koyama, and K. Usami, et al., "Design and Analysis of On-chip Leakage Monitor using an MTCMOS circuit," The 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC'08), pp.205-208, Jul. 2008.
- [44] 工藤 優, 宇佐美 公良, "仮想グランド線電圧の自動検出による細粒度パワーゲーティング制御", 電子情報通信学会 VLD 研究会 (デザインガイア), 2012 年 11 月.
- [45] M. Kudo, K. Usami, "Sleep Control Using Detection of Virtual Ground Voltage for Fine-Grain Power Gating", The 28th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC'13), Jun. 30 – Jul. 3, 2013.
- [46] M. Guthaus et al., "Mibench: A free, commercially representative embedded benchmark suite," Proc. 2001 International Workshop on Workload Characterization, pp.3-14, Dec. 2001.
- [47] H. Mahmoodi-Meimand and K. Roy, "Data-retention flip-flops for power-down applications," in Proceedings of the IEEE International Symposium on Circuits and Systems , pp. II677– II680, May 2004.
- [48] L. Torres, et al, "Trends on the application of emerging nonvolatile memory to processors and programmable devices," ISCAS2013.
- [49] H. Noguchi et al, "Highly Reliable and Low-Power Nonvolatile Cache Memory with Advanced Perpendicular STT-MRAM for High-Performance CPU," Symposium on VLSI Circuits, 2014.
- [50] M. Kudo, K. Usami, "MTJ Based Non-Volatile Flip Flop to Prevent Useless Store Operation", The 30th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC'15), Jun. 29 - Jul. 2, 2015.
- [51] G. D. Pendina et al, "Ultra Compact Non-volatile Flip-Flop for Low Power Digital Circuits Based on Hybrid CMOS/Magnetic Technology," PATMOS2011.
- [52] T. Na, et al, "High-performance Low-power Magnetic Tunnel Junction Based

Non-volatile Flip-flop,” ISCAS2014.

- [53] S. Yamamoto, Y. Shuto and S. Sugahara, “Nonvolatile Delay Flip-flop using Spin-transistor Architecture with Spin Transfer Torque MTJs for Power-gating Systems,” *Electronics Letters*, Vol. 47 No. 18, Sept. 2011.
- [54] T. Kawahara, et al, “2 Mb SPRAM (Spin-Transfer Torque RAM) with Bit-by-Bit Bi-Directional Current Write and Parallelizing-Direction Current Read,” *IEEE Journal of Solid-State Circuits*, Vol. 43, No. 1, pp.109-422, January 2008.
- [55] K. Lee, et al, “Unified Embedded Non-Volatile Memory for Emerging Mobile Markets,” ISLPED 2014.
- [56] N. Sakimura, et al, “Nonvolatile Magnetic Flip-Flop for Standby-power-free SoCs,” CICC 2008.
- [57] 宇佐美 公良, 工藤 優, 平賀 啓三, 屋上 公二郎, “ストア／リストア動作のロバスト性を向上させた MTJ 利用不揮発性フリップフロップ回路” 応用物理学会春季学術講演会, 2016 年 3 月.
- [58] M. Lueders, et al, “Architectural and Circuit Design Techniques for Power Management of Ultra-low-power MCU Systems,” *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, Vol.22, No.11, pp.2287-2296, Nov. 2014.

## 謝辞

本博士論文の審査にあたり、主査を引き受けて下さった芝浦工業大学 工学部情報工学科の宇佐美公良教授に、心からの感謝の意を表します。研究の進め方から論文の書き方など、終始ひとかたならぬご指導を賜りました。最後まで本当に有難うございました。

また、審査委員を快く引き受けて下さった芝浦工業大学 工学部 電子工学科 佐々木昌浩准教授、情報工学科 大倉典子教授、木村昌臣教授、および、ソニー（株）の屋上公二郎博士に、深くお礼を申し上げます。

研究を進めていく上で多大なるご支援をいただいた、ソニー（株）の平賀啓三氏に、心から感謝いたします。

最後に本研究の課程で、宇佐美研究室と一緒に研究を行い、活発な議論をさせていただいた、宮内誠氏、安西幸夫氏に、深く感謝いたします。