

遅延故障を考慮したフォールトトレランス技術に基づく低消費電力方式

佐藤 寿倫^{†‡} 有田 五次郎[†][†]九州工業大学 情報工学部 知能情報工学科[‡]九州工業大学 マイクロ化総合技術センター

{tsato,arita}@ai.kyutech.ac.jp

<http://www.mickey.ai.kyutech.ac.jp/~tsato/cosmos/>

あらまし マイクロプロセッサの電力を削減できる新しい方式を提案する．本方式は遅延故障に対するフォールトトレランス技術を利用している．このフォールトトレランス技術は投機実行方式に基づいている．消費電力は電源電圧の自乗に比例するので，電圧を下げるのが非常に効果的である．しかし，同時にゲート遅延を増大させてしまい，その結果，クリティカルパスのタイミング制約を満足させるためにクロック周波数が低下してしまう．もし，遅延故障に対するフォールトトレランス技術が備わっていれば，必ずしもタイミング制約を満たす必要はない．これらの考察からわれわれは，投機実行機構を応用したフォールトトレランス技術を，遅延故障向けに提案する．シミュレーションの結果，本方式は有効であり，電力の削減が確認できている．

キーワード 低消費電力設計，フォールトトレランス，タイミング制約，分岐予測，投機実行

Improving Energy Efficiency via Timing Fault Tolerance

TOSHINORI SATO ITSUJIRO ARITA

KYUSHU INSTITUTE OF TECHNOLOGY

{tsato,arita}@ai.kyutech.ac.jp

<http://www.mickey.ai.kyutech.ac.jp/~tsato/cosmos/>

Abstract A novel technique to reduce power consumption is disclosed. It relies on a fault-tolerant mechanism for timing constraints based on speculative execution technique. Since power reduces quadratically with supply voltage, supply voltage reduction can result in substantial power savings. However, it also causes larger gate delay, and thus clock must be slow down in order not to violate timing constraints of critical paths. If any fault-tolerant mechanism is provided for timing faults, it is not necessary to keep the constraints. From these observations, we propose a fault-tolerant technique for timing failures, which efficiently utilizes the speculative execution mechanism and reduces power consumption. We evaluate our proposal using a cycle-by-cycle simulator and find its efficiency.

key words low power design, fault tolerance, timing constraints, branch prediction, speculative execution

1. はじめに

近年の PDA や携帯電話といったモバイル機器の隆盛により、高性能かつ電力利用効率の良いマイクロプロセッサが重要度を増している。例えば、Java 環境はすでに携帯電話上で動作可能であり、ゲームやモバイルトレーディングといったアプリケーションが提供されている。ここで問題になるのは、携帯端末の性能が向上すると、そこで消費される電力も同様に増加する傾向にあるということである。CMOS 回路の電力 P_{active} とゲート遅延 t_{pd} は以下の式により与えられる。

$$P_{active} = f C_{load} V_{dd}^2 \quad (1)$$

$$t_{pd} \propto \frac{V_{dd}}{(V_{dd} - V_{th})^\alpha} \quad (2)$$

ここで f はクロック周波数、 C_{load} は負荷容量、 V_{dd} は電源電圧で、 V_{th} はデバイスの閾値電圧である。 α はキャリアの速度飽和を与えるパラメータで、近年の MOSFET では 1.3~1.5 の値をとる⁴⁾。式 (1) から判るように、電源電圧を下げることで電力消費を小さくする最も有効な方法である。しかし式 (2) により、同時にゲート遅延が増加する、すなわち動作周波数が低下してしまうことが判る。このことは、結局マイクロプロセッサの性能を減じってしまうことになる。

プロセッサの性能を維持するためには、電源電圧を下げてでもクロック周波数を変えないことが必要となる。このことは遅延故障を発生させ、動作異常を引き起こしてしまう。しかし、仮に遅延故障に対するフォールトレランス機構が備わっていれば、プロセッサの異常な動作を避けることが可能である。言い換えれば、われわれは「タイミング制約を満たすことは諦め、代わりにタイミング違反に対するフォールトレランス機構を備える」ことを提案する。すでにこのポリシーをプロセッサの性能を向上する目的で利用し評価済みである⁷⁾。本稿では、このポリシーを電力利用効率改善の目的に適用する。

本稿は以下の構成となっている。2 節で電力利用効率を改善する手法を説明する。3 節で提案手法を評価する環境を述べ、4 節でシミュレーション結果を紹介する。5 節で関連研究をまとめたのち、6 節でまとめとする。

2. フォールトレランスに基づく電力削減方式

本稿での提案はある種の並列処理に基づいている。それは空間的冗長性である。各回路はメイン部とチェック部から構成される。メイン部は低レイテンシかつ高スループットという高性能を維持できるように設計されるが、遅延故障を発生する可能性を秘めている。チェック部は遅延故障を生じないように設計されており、メイン部の遅延故障を検出しそこからの回復を行う目的で使用される。実際はメイン部とチェック部は同一の回路で構成されている。しかし、それらに供給されるクロックの周波数が異なっている。電力利用効率の改善は以下のようにして

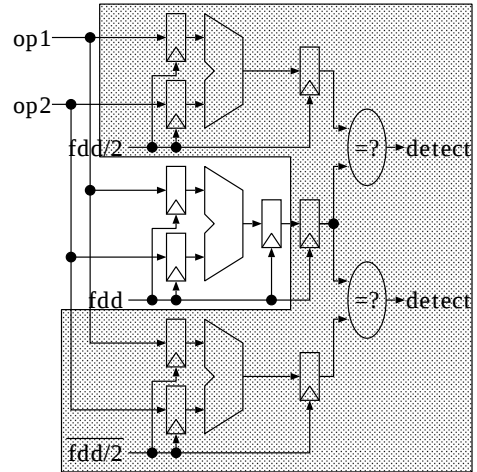


図1 提案手法を採用した ALU

達成できる。まず電源電圧が V_{dd} であるときメイン部は遅延故障を生じないと仮定する。式 (2) にしたがって、メイン部の最大動作周波数 f_{dd} は以下のように決定される。

$$f_{dd} \propto \frac{(V_{dd} - V_{th})^{1.3-1.5}}{V_{dd}} \quad (3)$$

ここで説明を容易にするために、式 (3) を

$$f_{dd} \propto V_{dd} \quad (4)$$

の様に単純化するが、これにより一般性が失われる心配は無い。電力利用効率を改善しようと思うと、電源電圧を V_{dd} よりも低い V_L に設定する必要がある。この結果、通常は動作周波数は V_{dd} で決まる f_L に低下するが、われわれはそれを f_{dd} に維持する。なぜなら、遅延故障を生じるのは非常に稀であると期待できるからである。チェック部は遅延故障を検出しなければならないので、電源電圧 V_L かつ動作周波数 f_L のもとで動作させる。つまりチェック部では遅延故障は発生しない。回路のスループットを維持するためには、必要であればチェック部を複製して複数用意すればよい。もし、電源電圧低下による電力削減の効果が、空間性冗長性が増すことによる電力増加の影響よりも大きければ、本提案方式は有効であるということになる。

以下では例を用いて提案手法を説明するが、いかなる組合せ回路に対しても適用可能である。図1は本方式を採用した ALU の例である。フォールトレランスの為の追加回路には網をかけてある。この ALU は電源電圧 V_{dd} かつ動作周波数 f_{dd} で動作していると仮定する。ここで電源電圧を例えば V_{dd} の半分の $V_L = \frac{V_{dd}}{2}$ に低下させることを目標とする。まず ALU を複製し合わせて3つ用意する。メイン ALU と呼ばれる一つの ALU は電源電圧 V_L かつ周波数 f_{dd} で動作させ、チェック ALU と呼ばれる残りの二つの ALU は電源電圧 V_L かつ周波数 $f_L = \frac{f_{dd}}{2}$ で動作させる。すなわちチェック ALU では遅延故障は発生しないので、メイン ALU における故障検

表 1 プロセッサ構成

フェッチ幅	4 命令
分岐予測	512 セット, 4 ウエイセットアソシアティブ BTB, 2048 エントリ bimodal 予測器, コミット時に更新, 8 エントリリターンアドレススタック, 3 サイクルミスペナルティ
命令ウインドウ	16 エントリ命令キュー, 8 エントリロードストアキュー
発火幅	4 命令
コミット幅	4 命令
演算器	4 iALU's, 1 iMUL/DIV, 2 Ld/St's, 4 fALU's, 1 fMUL/DIV
レイテンシ (全体/間隔)	iALU 1/1, iMUL 3/1, iDIV 20/19, Ld/St 2/1, fADD 2/1, fMUL 4/1, fDIV 12/12
レジスタファイル	32 本の 32 ビット整数レジスタ, 32 本の 32 ビット浮動小数点レジスタ
命令キャッシュ	16K, ダイレクトマップ, 32 バイトブロック, 6 サイクルミスペナルティ
データキャッシュ	16K, 4 ウエイセットアソシアティブ, 32 バイトブロック, 2 ポート, ライトバック, ノンブロッキング, 6 サイクルミスペナルティ
2 次キャッシュ	共用, 256K, 4 ウエイセットアソシアティブ, 64 バイトブロック, 48 サイクルミスペナルティ

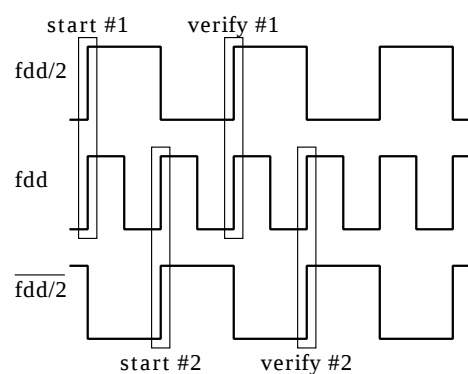


図 2 クロック信号

出に利用できる．このとき ALU が消費する電力は以下のように削減できている．メイン ALU が消費する電力は $f_{dd}C_{load}V_{dd}^2$ から $f_{dd}C_{load}(\frac{V_{dd}}{2})^2 = \frac{f_{dd}C_{load}V_{dd}^2}{4}$ に減少している．一方、二つのチェック ALU で消費される電力は、 $2 * \frac{f_{dd}}{2}C_{load}(\frac{V_{dd}}{2})^2 = \frac{f_{dd}C_{load}V_{dd}^2}{4}$ である．したがって全消費電力は、 $f_{dd}C_{load}V_{dd}^2$ から $\frac{f_{dd}C_{load}V_{dd}^2}{2}$ に低下している．さらに α として 1.3 ~ 1.5 の値を採用すれば、電力削減量はさらに大きくなる．

各 ALU に分配されるクロック信号は図 2 に示すとおりである、二つのチェック ALU に供給されるクロックは相補的な関係にあるので、チェック ALU は交互にメイン ALU をチェックすることになる．図 2 には連続する二つの演算がどのようにチェックされるかも説明されている．故障検出は、メイン ALU とチェック ALU から得られる二つの結果を比較することで実施される．もし両者が一致しなければ遅延故障が検出されたことになる．その場合には、何らかの方法でプロセッサの状態を正常に回復する必要がある．ここで、比較器で遅延故障が生じるわけにはいけないので、比較器は遅い周波数 f_L で動作させる．プロセッサの状態を故障発生前の状態に回復させるために、われわれは現在のマイクロプロセッサに既に備わっている投機実行のための機構を利用することを提案する．投機実行とは、分岐命令が現われた時に、分岐の方向が決定される前に予測に基づいて将来実行されると

表 2 ベンチマークプログラム

プログラム	入力
164.gzip	input.compressed
175.vpr	net.in arch.in
176.gcc	ccep.i
186.crafty	crafty.in
197.parser	test.in
252.eon	chair
255.vortex	lendian.raw
256.bzip2	input.random

期待できる命令を実行する方式である．したがって分岐予測に失敗した場合には、プロセッサの状態を投機開始前の状態に回復させなければならない．容易に判るように、これはいま必要としている故障からの回復と非常に類似している．つまり、遅延故障を生じた命令を分岐予測に失敗した命令であると読み替えることで、回復機構を実現できる．この方法を用いれば、遅延故障からの回復を実現するハードウェアに要するオーバーヘッドは全くないことになる．

3. 評価環境

SimpleScalar/Alpha ツールセット (ver.3.0a)²⁾を利用して、サイクルレベルの精度を持つシミュレータを構築した．ベースとなるプロセッサモデルはレジスタ更新ユニット⁸⁾を利用して乱実行を実現するスーパースカラプロセッサであり、その構成は表 1 に示す通りである．

評価のためのベンチマークには SPECint2000 を使用した．表 2 にプログラムとその入力をまとめてある．使用した目的コードはミシガン大学から提供されているものである．252.eon を除いて各プログラムは、最初の 10 億命令をスキップし、その後につづく 1 億命令を詳細に実行した．ただし nop 命令は数に入っていない．

4. シミュレーション結果

本節で、2 節で紹介した例を用いた予備評価の結果を紹介する．メイン部一つとチェック部二つで回路を構成する場合には、以下の関係を満足しなければならない．

$$\frac{1}{2}f_{dd} \leq f_L < f_{dd}$$

したがって、チェック部の動作周波数には $\frac{1}{2}f_{dd}$ を選ぶことにする。この場合に電力削減量が最大になるからである。評価は、遅延故障が発生する確率を 0～30%の間で変化させ、そのときのプロセッサ性能を求めることで行う。現実には、 f_L が決まれば故障が生じる確率も決定されることに注意されたい。より現実的な環境下での評価は将来の課題である。図 3 にプロセッサ性能をまとめる。プロセッサ性能の指標には、サイクル当たりの完了命令数 (IPC: committed instructions per cycle) を用いる。すでに述べたように nop 命令は数に入っていない。各プログラムにおいて、提案モデルの性能をベースモデルの性能で標準化したものが図示してある。容易に判るように、故障確率が大きくなるとプロセッサ性能は著しく低下している。この主な原因は、分岐予測に用いられる回復機構では、故障からの回復に必要なペナルティが非常に大きいためである。近い将来マイクロプロセッサに実装されると予測されている⁵⁾命令再発行機構⁶⁾を、故障からの回復に利用すれば、この性能低下を十分補うことが出来ると想像できる。この検証は将来の課題である。

図 4 に電力消費量をまとめた。消費電力に実行時間を掛け合わせることで算出している。やはりベースモデルで標準化している。図より、故障確率がおおよそ 10% 以下であれば $\alpha = 2$ の場合でさえ電力消費量削減が確認できる。

5. 関連研究

上野ら⁹⁾は整数 ALU に可変レイテンシパイプライン (VLP: variable latency pipeline) を適用している。各演算に対して最長パスが短くなるように二つの回路を用意することで、1GHz で動作しながら実効レイテンシがほぼ 1 となる ALU を実現している。われわれの提案は、VLP に大きく影響されて考案されたものである。彼らは SPECint92 ベンチマークを用いて VLP を評価している。しかし、インオーダー実行のスカラプロセッサを用いており、現在主流の動的命令スケジューリングを行うスーパースカラプロセッサ上での効果は評価できていない。本稿では、4 命令並列のスーパースカラプロセッサ上で評価を行った。また彼らは、フォールトトレランス技術や消費電力削減の検討は行っていない。

Chandrakasan ら³⁾は並列処理を利用して、スループットを維持しつつ消費電力削減を達成している。二つの同一回路を用意することで、もとの回路の動作周波数の半分の周波数で動作させる。したがってスループットは維持される。動作周波数が低くなれば電源電圧を下げる事が可能である。この場合には、並列度向上による電力増加の影響よりも、電源電圧低下による電力削減の効果が大きい。われわれの提案も並列処理に基づく方式である。しかし、スループットだけでなくレイテンシも維持していることが彼らの提案と異なる。つまり、われわれの方式を用いれば、十分なデータ並列性の無いようなア

プリケーションにも効果を得ることが出来る。

DIVA¹⁾は空間冗長性に基づいた商用マイクロプロセッサ向けフォールトトレランス技術である。メインプロセッサの実行結果を動的にチェックするためのシンプルなチェッカープロセッサが付け加えられている。エラーからの回復には、分岐予測失敗からの回復に用いられている機構を流用している。つまり DIVA はハードウェアだけで実現できる方式である。DIVA における問題点は、メインプロセッサとチェッカープロセッサでコンテキストを共用するために、レジスタファイルとキャッシュに追加のポートが必要なことである。これにより、メインプロセッサの回路的な複雑度が増し、動作速度を低下させる恐れがある。加えて、DIVA では消費電力削減は検討されていない。

6. おわりに

本稿でわれわれは、フォールトトレランス機構を電力利用効率改善の目的に利用することを提案した。予備評価の結果から、遅延故障を生じる確率が小さければ、プロセッサ性能は若干低下するが、電力消費量は大幅に削減可能であることが確認できた。

将来の検討課題の一つは、故障からの回復に要するペナルティを削減することである。われわれは既に、本稿で評価した命令破棄方式はプロセッサ性能に重大な悪影響を与えることを確認している⁶⁾。一方、命令再発行方式を用いたときのペナルティは極めて小さいことが判っている⁶⁾。このことから、現在われわれは命令再発行方式を用いて本稿の提案を評価中である。プロセッサ性能低下を小さくしつつ、電力利用効率の大幅な改善が得られることが確認できつつある。

他の検討課題には、フォールトトレランス機構を導入するために必要なハードウェア規模を削減することがあげられる。VLP で提案されている 2 サイクルレイテンシ ALU は、チェック ALU として利用可能である。これを用いればハードウェア規模を大幅に削減できる。われわれは VLP を性能向上だけでなくフォールトトレランス技術への応用に向けて検討中である。

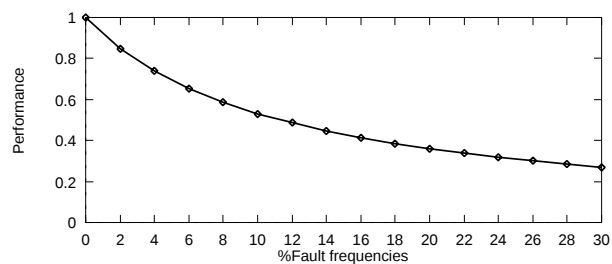
さらに、動作周波数と遅延故障発生確率の関係も検討する必要がある。このモデルが作成できれば、現実のマイクロプロセッサに用いた場合の有効性が評価できると期待している。

謝 辞

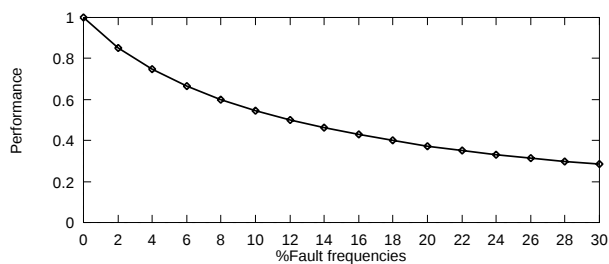
本研究の一部は、株式会社東芝 セミコンダクタ社の支援によるものです。

参 考 文 献

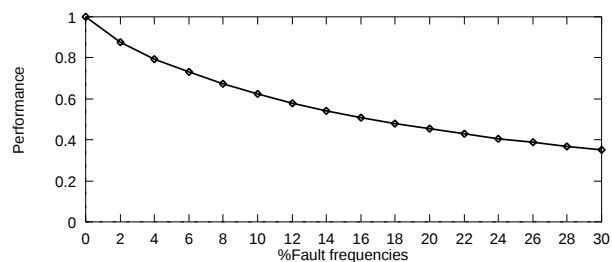
- 1) T.M.Austin, "DIVA: a reliable substrate for deep submicron microarchitecture design," 32nd International Symposium on Microarchitecture, 1999.
- 2) D.Burger and T.M.Austin, "The SimpleScalar tool set, version 2.0," ACM SIGARCH Computer



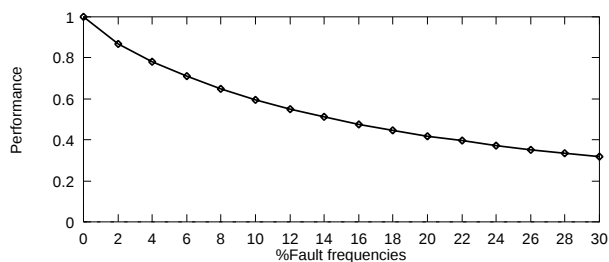
(i) 164.gzip



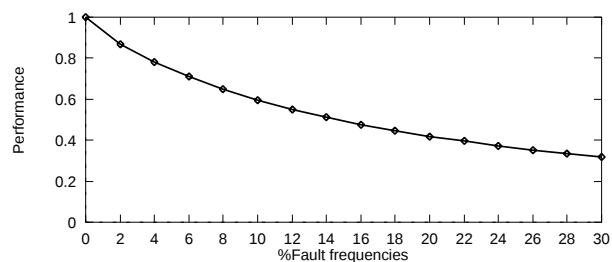
(ii) 175.vpr



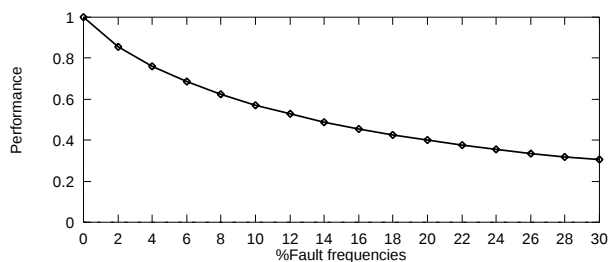
(iii) 176.gcc



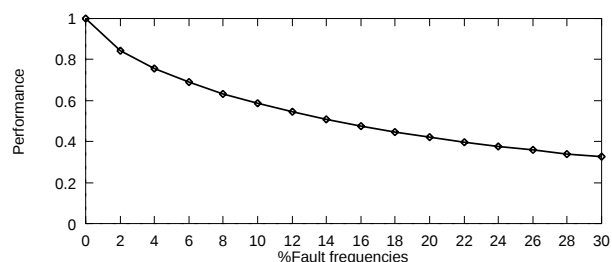
(iv) 186.crafty



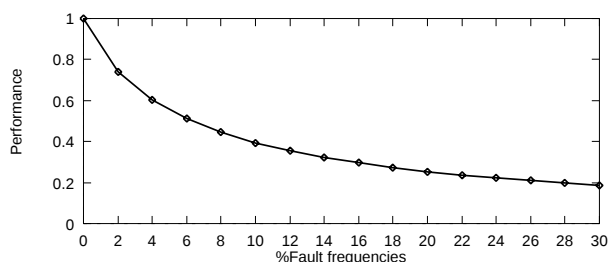
(v) 197.parser



(vi) 252.eon



(vii) 255.vortex



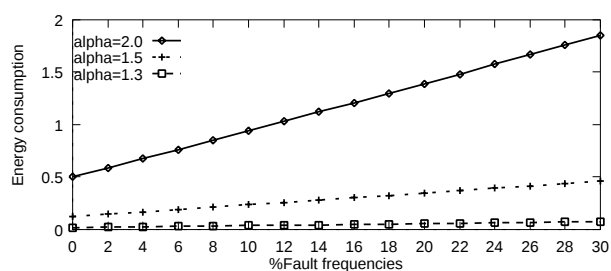
(viii) 256.bzip2

図 3 プロセッサ性能

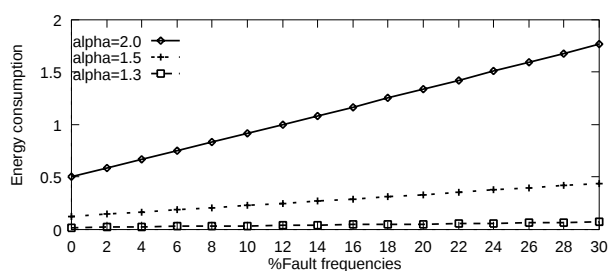
Architecture News, vol.25, no.3, 1997.

- 3) A.P.Chandrakasan and R.W.Brodersen, "Minimizing power consumption in digital CMOS circuits," Proceedings of IEEE, vol.83, no.4, 1995.
- 4) T.Hiramoto and M.Takamiya, "Low power and low voltage MOSFETs with variable threshold voltage controlled by back-bias," IEICE Transactions on Electronics, vol.E83-C, no.2, 2000.

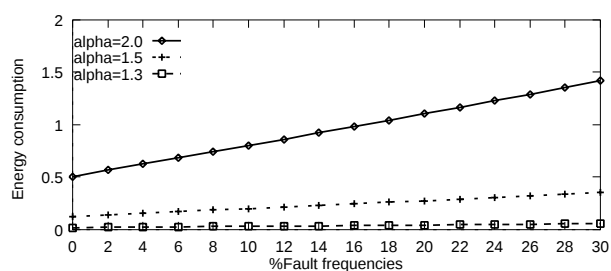
- 5) Intel Corporation, "Inside the NetBurst micro-architecture of the Intel Pentium 4 processor," White paper, 2000.
- 6) T.Sato, "Data dependence speculation using data address prediction and its enhancement with instruction reissue," 24th Euromicro Conference, Workshop on Digital System Design: Architectures, Methods and Tools, 1998.



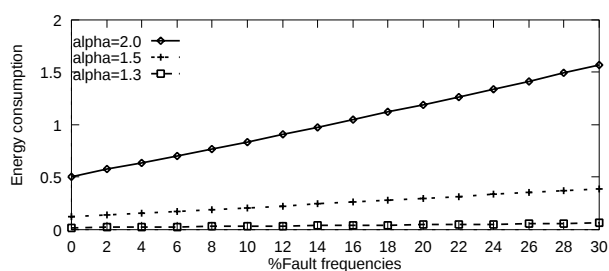
(i) 164.zip



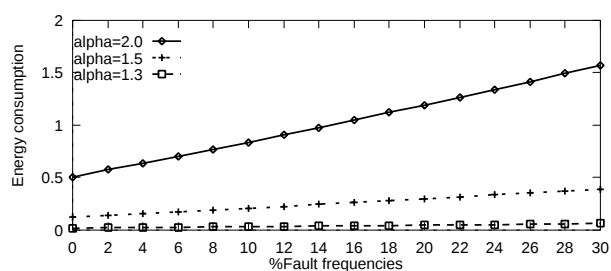
(ii) 175.vpr



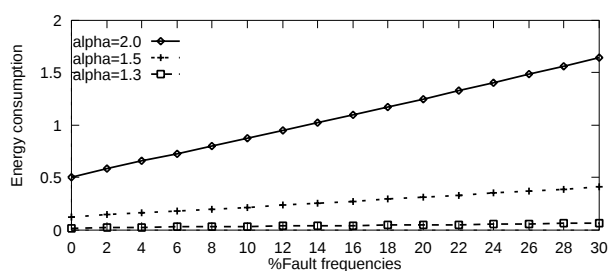
(iii) 176.gcc



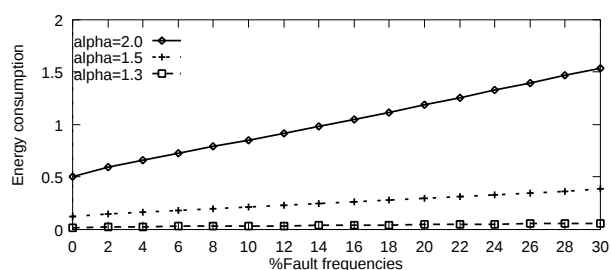
(iv) 186.crafty



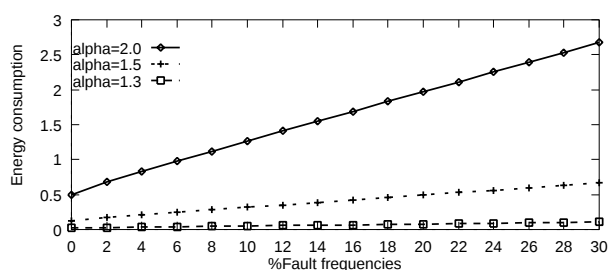
(v) 197.parser



(vi) 252.eon



(vii) 255.vortex



(viii) 256.bzip2

図 4 電力消費量

- 7) T.Sato and I.Arita, "Give up meeting timing constraints, but tolerate violations," Cool Chips IV, 2001.
- 8) G. S. Sohi, "Instruction issue logic for high-performance, interruptible, multiple functional unit, pipelined computers," IEEE Transactions on Computers, vol.39, no.3, 1990.

- 9) 上野喜代治, 幾見宣之, 近藤勝久, 森順治, 平野勝士: "Variable Latency Pipeline (VLP) を用いた 1GHz ALU データパス," 信学技法, ICD97-1, 1997.