

VHDL を用いた論理回路の消費電力見積り方法の提案 ～32ビットアダー回路の消費電力についての考察～

木村 勉[†] 山本俊之^{††} 満田千秋^{††}
塩見彰睦^{††} 今井正治^{††} 引地信之^{†††}

[†] 豊田工業高等専門学校

^{††} 豊橋技術科学大学

^{†††} SRA

あらし

本論文では VHDL の処理系を用いて、CMOS 型集積回路の消費電力を評価する方法を提案する。まず、VHDL のネットリスト記述を VHDL 処理系に入力し、各基本セルの出力を計測することにより、消費電力を評価する方法について述べる。次にこの方法を用いて、実現方法(アーキテクチャ)が異なる数種類の 32 ビットアダーについて、実現方法とトグル数の関係を調査した結果について報告する。

和文キーワード 消費電力, トグル数の計測, VHDL, CMOS 回路

Power Consumption Estimation of Logic Circuits using VHDL

Tsutomu KIMURA[†] Toshiyuki YAMAMOTO^{††} Chiaki MITSUTA^{††}
Akichika SHIOMI^{††} Masaharu IMAI^{††} Nobuyuki HIKICHI^{†††}

[†] Toyota College of Technology

^{††} Toyohashi University of Technology

^{†††} Software Research Associates, Inc.

Abstract

This paper proposes a method to estimate the power consumption of CMOS logic circuits using VHDL. Where power consumption is estimated by measuring toggle counts of basic cells. Then a 32 bit adder has been implemented in different architectures and the toggle count in each implementation has been measured. Finally, the relation between implementations and toggle counts has been investigated.

Key words power consumption, toggle count measurement, VHDL, CMOS circuit

1 はじめに

集積回路技術の進歩によりハードウェアが大規模化し、その結果、集積回路内部での消費電力が大きくなってきた。また、小型化、軽量化が重要な課題である携帯用のアプリケーション機器などにおいては、バッテリーの負荷を軽減させるために、集積回路の消費電力を抑制する事が重要である。

ハードウェア/ソフトウェア協調設計手法を用いた特定用途向き集積化プロセッサ (ASIP) の設計において、ライブラリとして用意してある複数のモジュールの中から目的にあったモジュールを選択する場合について考える。この場合、例えば数種類の ALU モジュールの中から、速度や面積、消費電力などの条件を考慮して目的に最も適合するモジュールを選ぶためには、それぞれのモジュールの消費電力の正確な評価が必要である。

低消費電力の集積回路を設計するためには、集積回路内で消費される電力を出来るだけ正確に評価する必要がある。回路内部での消費電力を正確に評価するためには、SPICE などを用いた回路シミュレーションを行う事が望ましいが、この方法を大規模集積回路に適用すると膨大な計算時間が必要となる。したがって、この方法は現実的ではない。

これに代わる方法として、各ゲートのトグル確率をから消費電力を求める方法がある。トグル回数をを用いた消費電力の評価手法には、

- (1) ランダムパターン・シミュレーションによる手法、
- (2) 信号値遷移確率を利用した手法

等が知られている [1]。これらの手法は、回路シミュレーションによる方法と比較すると、消費電力の見積り精度は劣るものの、消費電力の評価を短時間でを行う事が可能であるという利点を持っている。しかし、トグル確率を用いる方法では、集積回路で実際に処理されるデータの性質を消費電力の評価に反映させる事が困難である。本論文では、VHDL の処理系を用いて実際のトグル回数を計測し、これに基づいて比較的精度の高い消費電力の見積りを短時間で得る方法を提案する。

ASIP の HW/SW 協調設計を行う場合、使用するアプリケーション・プログラムやデータがあらかじめ与えられていれば、これらの情報を用いてシミュレーションを行い、消費電力をより正確に見積る事が可能である。

本論文の構成は以下の通りである。まず第 2 章では、VHDL 処理系を用いて論理回路のトグル回数を評価する方法について述べる。次に第 3 章では、この方法を用いて、アーキテクチャの異なる 32 ビットアダー回路のトグル回数を調査した結果について述べる。第 4 章では、研究結果のまとめと今後の課題について述べる。

2 VHDL 処理系を用いた消費電力の評価

VHDL は IEEE の標準ハードウェア記述言語 (IEEE Std-1076) [2] であり、既に多数の処理系が開発され、ハードウェアや VLSI の設計に広く用いられている。また、VHDL は他の HDL よりも非常に高い記述能力を持っている。

VHDL では 1 つの entity (設計対象のインタフェースの宣言) に対して複数の architecture (実現方法の記述) を対応させる事が可能である。そこで、論理合成およびシミュレーション用とトグル計測用の 2 種類のライブラリを用意しておけば、それらを差し換えるだけで設計された論理回路のトグル数の計測を容易に行う事が出来る。この際、VHDL の処理系自体には何ら手を加える必要は無い。すなわち、既存の処理系をそのまま用いて消費電力 (トグル回数) の評価が行えるという利点がある。

2.1 回路モデル

この節では、VHDL を用いて消費電力を評価するための回路モデルについて述べる。一般に、CMOS 論理回路での電力の消費は次の 3 種類に分類出来る。

- (1) 充放電電流
- (2) 貫通電流
- (3) リーク電流

これらの電流の中で、(3) のリーク電流は他の電流と比較して、ごくわずかである。また、サブミクロン技術を用いた集積回路では、(2) の貫通電流は、(1) の充放電電流と比較して 10%~30% 程度であると報告されている [5]。しかし、貫通電流が流れるのは、基本セルの出力が 0 → 1 または 1 → 0 に変化する場合だけであるので、基本セルのトグルを観察すれば貫通電流を見積る事は可能である。

したがって、以下では、充放電電流の見積りを中心にして考える事にする。

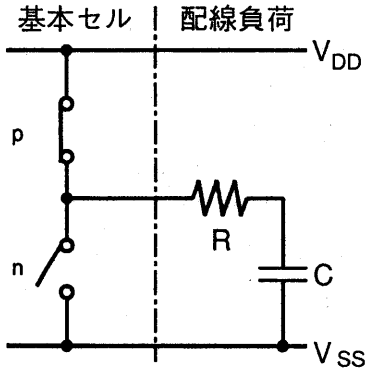


図 1: 論理回路の消費電力見積りモデル

ディープ・サブミクロン技術を用いた VLSI では、消費電力の中で充放電電流の割合がこれまでよりも高くなるので、充放電電流がますます支配的になると考えられるからである。本論文で用いる論理回路の消費電力見積りモデルを図 1 に示す。

同図で、 $p=OFF$, $n=ON$ の状態から $p=ON$, $n=OFF$ の状態に変化した時、すなわち論理回路の出力が $0 \rightarrow 1$ に変化(トグル)した時に、コンデンサ C へ V_{DD} から充電電流が流れる。リーク電流が無視出来ると仮定すると、放電電流の総和は、充電電流の総和と等しい。また、十分に長い時間の間に起きる $0 \rightarrow 1$ のトグルの回数と $0 \rightarrow 1$ のトグルの回数は高々 1 回しか違わない。そこで、以下では、コンデンサ C (配線負荷容量) に対する充電電流と直接関係のある $0 \rightarrow 1$ のトグル回数を測定する方法について議論する。また、 $0 \rightarrow 1$ のトグルを単に「トグル」と呼ぶ事にする。

2.2 VHDL を用いたトグル計測の方法

VHDL を用いてトグルを計測するために、トグル数の計測用のライブラリを用意する。出力の変化(トグル)の情報を出力するように記述した基本セル(図 2)を用いて、回路を VHDL でネットリスト記述する。

市販されている多くの処理系は EDIF のネットリストを VHDL に変換する機能を持っているので、論理合成可能な記述であれば、提案する手法でトグルの数の計測が可能である。

ライブラリをトグル情報を出力するものと置き換え、ネットリスト記述された回路を処理系でシミュ

```

USE STD.TEXTIO.ALL;
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;

ENTITY AN2 IS
  PORT ( A : IN STD_LOGIC;
        B : IN STD_LOGIC;
        Z : OUT STD_LOGIC );
END AN2;

ARCHITECTURE toggle OF AN2 IS
  BEGIN
    an2_proc : PROCESS
      CONSTANT rise_time : TIME := 0.48 NS;
      CONSTANT fall_time : TIME := 0.77 NS;
      VARIABLE toggle_info : STRING(1 TO 5);
      VARIABLE tmp1, tmp2 : STD_LOGIC := 'U';
      FILE toggle_file : TEXT IS OUT "/dev/tty";

    BEGIN
      tmp1 := A AND B;

      IF tmp1 = '1' AND tmp2 = '0' THEN
        toggle_info := "an2:1";
        WRITE (toggle_file, toggle_info);
        Z <= tmp1 AFTER rise_time;
      ELSIF tmp1 = '0' AND tmp2 = '1' THEN
        toggle_info := "an2:0";
        WRITE (toggle_file, toggle_info);
        Z <= tmp1 AFTER fall_time;
      ELSE
        Z <= tmp1;
      END IF;

      tmp2 := tmp1;
      WAIT ON A, B;
    END PROCESS an2_proc;
  END toggle;

```

図 2: セルライブラリの例 (2 入力 AND)

レーションを行ない、出力されたトグル情報を集計することで消費電力を見積ることができる。(図 3)

トグルの記録は基本セルの種類ごとに集計する。これは VHDL ソース中の各基本セルは種類ごとにコンポーネントとして用意されたものを使用しているので、インスタンス化された個々の基本セルを識別する事が困難であったからである。

第 2 章でも述べたように CMOS 回路は負荷容量によって消費電力の大きさが決まる。このため、より正確な評価を行うためには負荷容量をも考慮し、個々の基本セルに負荷容量がどれくらいあるのかを調査しなければならない。このために、各基本セルに識別子を付けて、種類と負荷容量ごとにトグルの数を集計し、消費電力を見積る必要がある。しかし、今回は基本セルの種類ごとのトグルのみに着目して計測実験を行ったため、上記の解析は行っていない。

今回の実験ではトグル情報を出力するために、VHDL の TEXTIO パッケージを用いた。これは VHDL の標準パッケージであるので、すべての処理系で使用することが出来る。

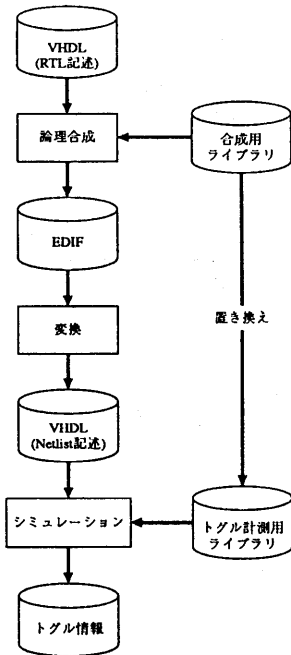


図 3: VHDL によるトグル数の計測の流れ

3 32 ビットアダー回路のトグル数の計測

第 2 章で述べた方法を用いて、アーキテクチャの異なる加算器のトグル数の計測を行った。ここではアーキテクチャ、実験方法及び結果について述べる。

3.1 アーキテクチャ

本研究では始めにアーキテクチャの異なる加算器ではどのようなトグル数の違いがあるかを調査した。今回は 32 ビットの加算を行うことを目的として、いくつかの加算器を用意した。用意した加算器は、

1. リップルキャリーアダー (RCA)
 - 32bit×1clk (RCA32)
 - 16bit×2clk (RCA16)
 - 8bit×4clk (RCA8)
 - 4bit×8clk (RCA4)
2. キャリールックアヘッドアダー (CLA)
 - 32bit×1clk (CLA32)
 - 16bit×2clk (CLA16)
 - 8bit×4clk (CLA8)
 - 4bit×8clk (CLA4)

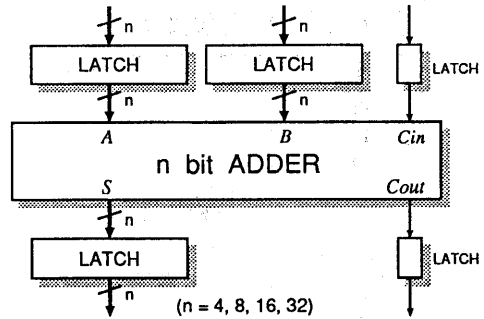


図 4: 加算モジュールの構成

の 8 種類である。

図 4 に回路図を示す。

ここで、ラッチを含めた回路全体を加算モジュールと呼び、ラッチを除いた部分を加算回路と呼ぶことにする。

CLA4 は 4bit のキャリールックアヘッド回路 (図 5) を用いて、実現した (図 6) [6]。8bit では CLA4 を 2 つ用意し、下位の Cout を上位の Cin に接続する RCA 接続で構成した。

CLA16 は図 7 に示すような回路を組み [6]、CLA32 は CLA8 と同じように 2 つの CLA16 を RCA 接続で構成した。また、RCA 回路を図 8 に示す。

加算回路 (RCA, CLA) の入出力信号線にはラッチをおき、32bit の加算が出来るようにした。(図 4) 16bit 以下の加算モジュールでは次のような命令をキャリーフラグを初期化した後、必要な回数だけ繰り返すことになる。

```
LD A, Data(a) ; A ← Data(a)
LD B, Data(b) ; B ← Data(b)
ADC           ; S ← A + B + Cr
LD mem, S    ; mem ← S
```

ラッチから加算回路への信号入力へのタイミングは同時に行われるようにして、入力のタイミングのズレによる余分なトグルが行われなかったようにした。それぞれの回路の基本セル数を表 1 に示す。

3.2 実験方法及び結果

実験は以下の項目を基にして行なった。

1. VHDL の処理系は Synopsys の VHDL Analyzer, VHDL Simulator の Ver 3.1b を使用。

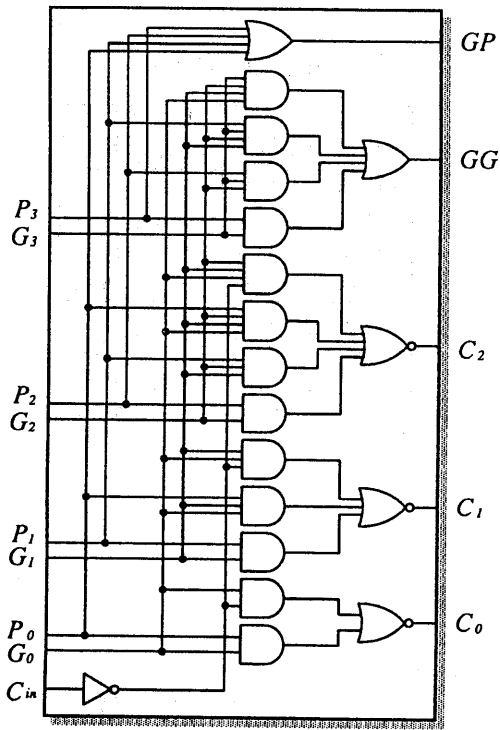


図 5: 4bit キャリールックアヘッド回路

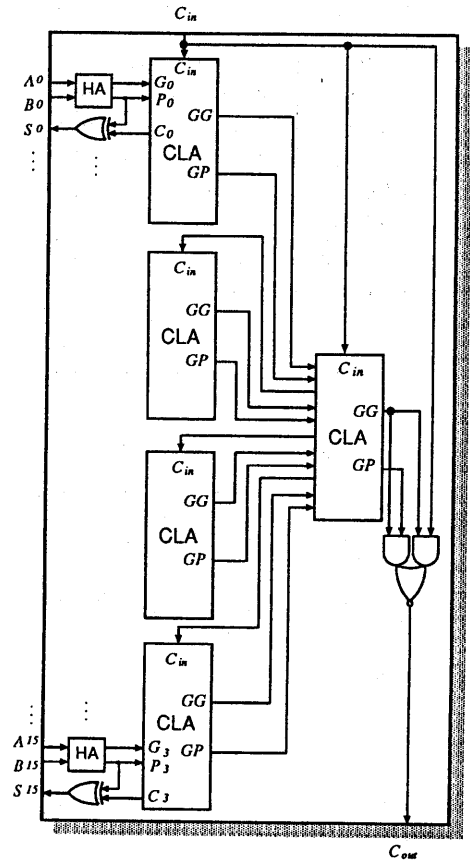


図 7: CLA16 加算回路

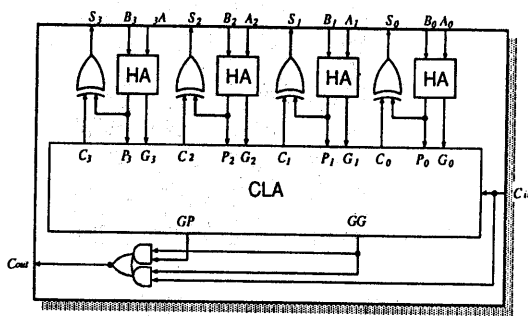


図 6: CLA4 加算回路

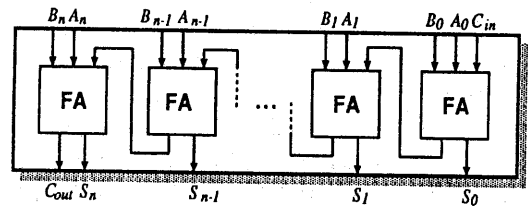


図 8: RCA 回路

表 1: 各加算モジュールの基本セル数

Module Type	Basic Cell		
	Adder	Latch	Total
RCA32	160	98	258
RCA16	80	50	130
RCA8	40	26	66
RCA4	20	14	34
CLA32	330	98	428
CLA16	165	50	215
CLA8	76	26	102
CLA4	38	14	52

表 2: 各回路のトグル数

Module Type	Toggle		
	Adder	Latch	Total
RCA32	7848	2422	10270
RCA16	7425	2489	9914
RCA8	6930	2570	9500
RCA4	6238	2772	9010
CLA32	7811	2422	10233
CLA16	8232	2489	10721
CLA8	7188	2570	9758
CLA4	7111	2772	9883

- セルライブラリは Synopsys のシステムに添付されている CLASS を使用。
- 入力データは 32 ビット一様乱数の対 100 組。
- トグル計測は 0 と 0 の加算で初期化を行ってから、基本セル単位で計測。

制御回路はその構成方法により、トグルの数が変化する [7] が、今回の実験では制御回路については考慮せずに加算モジュールのみを計測した。

各回路のトグル数及びトグル率を表 2、表 3 に示す。ここで Adder は加算回路、Latch はラッチ回路、Total は回路全体のトグル数およびトグル率を示す。トグル率は基本セル単位である。また、図 9 にトグル数と回路の関係を示す。

基本セル数は 4bit を基本にして、ほぼ 2ⁿ 倍となっている。CLA 回路の場合、8bit と 16bit に差があるのは、加算回路の構成方法が変わったためである。

表 3: 各回路のトグル率

Module Type	Toggle Rate (%)		
	Adder	Latch	Total
RCA32	49.05	24.71	39.81
RCA16	46.41	24.89	38.13
RCA8	43.31	24.71	35.98
RCA4	38.99	24.30	33.13
CLA32	23.67	24.71	23.91
CLA16	24.95	24.89	24.93
CLA8	23.64	24.71	23.92
CLA4	23.39	24.30	23.64

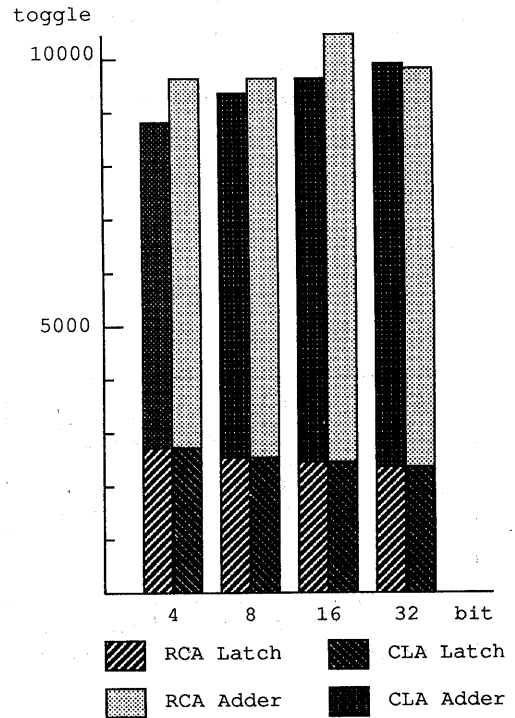


図 9: トグル数と回路の関係

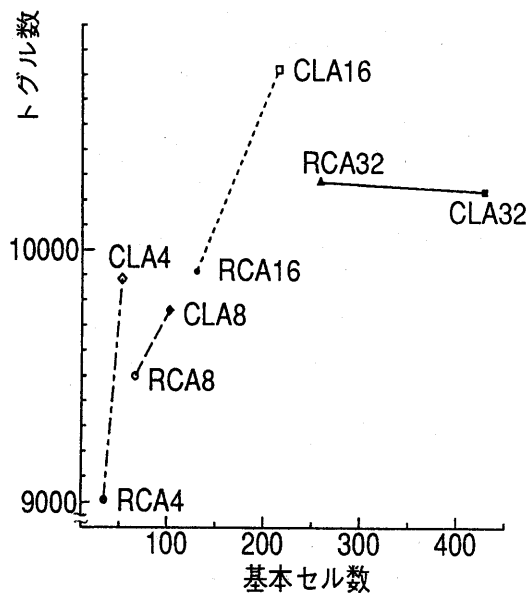


図 10: トグル数と基本セル数

る。これは CLA8 は CLA4 を RCA 接続しているが、CLA16 は図 7 に示すような構成にしたためである。また、ラッチ回路はキャリーフラグのラッチが含まれているため、完全な 2^n 倍ではない。

RCA, CLA ともにすべてにおいてラッチのトグル数はほとんど同じであった。

加算回路は図 9 から RCA ではビット幅が大きくなると、トグル数も多くなる。しかし、CLA は 16bit 加算回路が最大となっている。

トグル率では、ラッチ回路がすべての回路において約 24% になっている。

加算回路は CAL が約 24% に対して RCA は約 38%~約 49% である。32bit 加算回路では、RCA, CLA のトグル数が逆転して CLA の方がやや小さくなっている。図 10 に基本セル数とトグル数の関係を示したが、両者には単純な比例関係は無い。

4 考察

ラッチ回路において、トグル数がビット幅が大きくなれば若干少なくなるのは、ビット幅が 4, 8, 16bit 場合、クロック数だけ繰り返して加算を行な

い、32bit の加算を実現するときにキャリーフラグのラッチにトグルが起こるからである。

加算回路を見ると、RCA はキャリーの伝搬があるために、ビット数が大きくなるとトグルの数も大きくなる。CLA ではキャリーの伝搬が無いためにそれほど差はない。

32bit の加算回路を比較すると、CLA の方が RCA よりもトグルが小さくなっている。これは先ほど述べたキャリーの伝搬に原因があると思われる。32bit 加算の場合、RCA と CLA でのトグル数の差はほとんどないので、回路の面積と速度のトレードオフということになる。

図 10 から、アーキテクチャによって基本セル数とトグル数の間には大きく変化することがわかる。

最小のトグル数は RCA4 で、加算回路の部分だけ見ると RCA32 に比べて約 20% ほど小さい。低消費電力の観点からは RCA4 が有効であるが、実行速度が遅くなるという問題がある。汎用の集積回路の場合はどのようなアプリケーションが使用されるかわからないので、高速な回路を使用するが、アプリケーションが固定されている ASIP などでは、ある一定の時間内に処理が終了すればよいものがあるので、処理が間に合うようにすれば、低速でも消費電力が小さい回路を選択することが考えられる。

5 まとめ

本論文で VHDL を用いた消費電力の見積り方法について述べた。この方法を用いることにより、より正確な見積りが可能であると思われる。

従来の観点で消費電力を考慮すると、回路規模の大きいものほど消費電力が大きくなると捉えてしまう。しかし、本研究で実験した結果、一概に大きい回路ほど消費電力が大きくなるとは限らないことが分かった。これは各基本セルに対して同じ割合の電力を消費すると捉えてしまっていたからである。しかし、実際にはアーキテクチャによってトグル率が大きく変化することが分かった。

ASIP のようにアプリケーションやデータがある程度分かっている場合はそれらを用いて消費電力の評価を行うのが良いと思われる。

また、この見積り方法を利用すれば、セルライブラリの工夫で大規模な回路の一部分の消費電力を見積る、といったことも可能である。つまり、あるモジュールを評価してどのようなアーキテクチャが目的

に合うのかを見極める手段として用いて、HW/SW 協調設計に利用することが可能である。

6 今後の研究

今後は他の演算回路(乗算器, 除算器など)の評価を行い, いくつかの制御方法を用いて, 制御方法とトグル数の関係についても検討する。

さらに, 負荷容量も考慮し, より正確な消費電力の評価を行う。

また, 今回の実験では CLA において 4bit と 8bit, 16bit と 32bit の間でトグル数の逆転現象が起こっている。これらの原因についても解明していきたい。

謝辞

本研究を進めるにあたり, 御助言をして下さった鶴岡工業高等専門学校佐藤 淳氏, 並びに豊橋技術科学大学 VLSI 設計研究室の諸兄に感謝致します。また, ツールを御提供して下さいました日本シノプシス社, メンターグラフィックスジャパン株式会社, 研究環境の一部を提供して頂いた株式会社サイエンス・クリエイトに感謝致します。

なお本研究の一部は財団法人 人工知能財団 (5AI 第 235 号-11), 及び財団法人堀情報科学振興財団の研究助成による。

参考文献

- [1] 上田祐彰・樹下行三: “CMOS 論理回路の信号値遷移確率の評価法について”, 情報処理学会研究報告 設計自動化 72-22, 1994.
- [2] IEEE : “IEEE Standard VHDL Language Reference Manual”, Std 1076-1993, IEEE, 1994.
- [3] IEEE : “IEEE Standard Multivalued Logic System for VHDL Model Interoperability (Std_logic_1164)”, Std 1164-1993, IEEE, 1994.
- [4] Lipsett, R., Schaefer, C., and Ussery, C. : “VHDL: Hardware Description and Design”, Kluwer Academic Publishers, 1989.
- [5] 平田昭夫・小野寺秀俊・田丸啓吉: “CMOS 論理ゲートにおける貫通電流による消費電力の定式化”, 電子情報通信学会総合大会論文集 A-124, 1995.
- [6] 楠 菊信・高木 茂: “デジタルシステム-基礎論から実用設計まで-”, 朝倉書店, 1987.
- [7] 秋田純一・浅田邦博: “低消費電力状態遷移回路のためのコード割当の評価”, 電子情報通信学会総合大会論文集 A-108, 1995.
- [8] 富沢孝・松山泰男 訳: “CMOS VLSI 設計の原理 システムの視点から”, 丸善, 1988.
- [9] Rouatbi, F., Haroun, B., Al-Khalili, A.J., : “Power Estimation Tool for Sub-Micron CMOS VLSI Circuit”, Proc. of ICCAD-92, pp.204-209, Nov. 1992.
- [10] Jiing-Yuan Lin, Tai-Chien Liu and Wen-Zen Shen : “A Cell-Based Power Estimation in CMOS Combinational Circuits”, Proc. of ICCAD-94, pp.304-309, Nov. 1994.