

データ符号化による VLSI における低消費電力・高信頼データ伝送手法の 検討

小松 聡[†] 藤田 昌宏[†]

[†] 東京大学 大規模集積システム設計教育研究センター 〒113-0032 東京都文京区弥生 2-11-16

E-mail: †{komatsu,fujita}@ee.t.u-tokyo.ac.jp

あらまし 現在の VLSI システム設計において、消費電力は最も重大な設計制約のうちの 1 つである。加えて、VLSI システムのエラーに対する耐性もまた現在あるいは将来の微細化された VLSI テクノロジーにおいて重要な要素の一つである。本稿では、オンチップデータ伝送での、高信頼データ伝送方式に対する低消費電力符号の影響についての検討を行う。SPEC2000 ベンチマークを利用した実験により、提案された方法により最高 33% の信号遷移遷移削減を実現できることを示した。さらに、エンコーダ/デコーダ回路でのオーバーヘッドおよびバス信号の振幅最適化を考慮した評価を行い、低消費電力符号化と高信頼符号化を組み合わせることで、将来の VLSI システムにおいて有効なデータ伝送方式を実現可能であることを示した。

キーワード

Low Power and Fault Tolerant Encoding Methods for On-Chip Data Transfer

Satoshi KOMATSU[†] and Masahiro FUJITA[†]

[†] VLSI Design and Education Center, University of Tokyo Yayoi 2-11-16, Bunkyo-ku, Tokyo 113-0032, Japan

E-mail: †{komatsu,fujita}@ee.t.u-tokyo.ac.jp

Abstract Energy consumption is one of the most critical constraints in the current VLSI system designs. In addition, fault tolerance of VLSI systems is also one of the most important requirements in the current shrunk VLSI technologies. This paper presents an impact of the low power encoding on the fault tolerant data encoding methods in on-chip data transfer scheme. Experiments using SPEC2000 benchmark programs show that the proposed methods can reduce signal transitions by up to 33% on the bus with fault tolerance. Moreover, the results show that bus signal swing optimization can increase the effectiveness of the encoding method.

Key words

1. ま え が き

VLSI プロセス技術・デバイス技術の進歩に伴い、システム・オン・チップなどに代表されるように、多くの機能ブロック、メモリ、プロセッサなどの IP を含んだ非常に大規模なシステムが 1 つの VLSI チップに実現可能になった。また、それに伴い VLSI チップの電源電圧も縮小し、結果として電源ノイズ、クロストークノイズ、ソフトエラーなどの様々なノイズに対する耐性が低減してきている。結果的に、オンチップでのデータ伝送の正確性を完全に保証することが難しくなりつつあり、将来の微細テクノロジーで製造される VLSI システムにおいて、何らかの信頼性向上技術が導入される必要があると言える。

現在の高性能計算機システムでは、プロセッサとメモリとの

間のデータ伝送において、冗長性を利用したエラー訂正コード (Error Correction Code; ECC) が利用されている。同様に、VLSI システムが常に正常に動作することを保証するために、VLSI 内の配線でのデータ伝送におけるエラー耐性を高めることが必要不可欠になると考えられる。現在までに、配置配線、電源分布などの物理情報を利用したノイズマージンを向上させる技術が数多く研究されてきているが、それらの技術は将来の莫大な規模の VLSI システムにおいては、最適化のための計算量の面から現実的でなくなる可能性が高い。一方、エラー訂正による信頼性向上手法は、システムのエラー耐性をより上位の設計レベルから考慮することができるため、システム・オン・チップにおけるデータ伝送のエラー耐性向上のために非常に有用な手法であると考えられる。しかしながら、エラー訂正符号

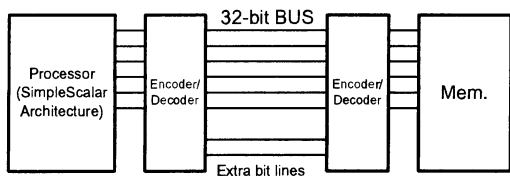


図1 対象とするアーキテクチャ
Fig.1 The target architecture.

では、冗長な信号の配線および符号化/復号化回路において、消費電力が増加してしまうと考えられる。

データ伝送の消費電力に関しては、バスに関連した消費電力がチップ全体の消費電力の大きな部分を占めており、将来の微細化されたテクノロジーにおいてはさらに比重が増すため、バスの消費電力削減低減は低消費電力 VLSI 設計において重要な位置を占めてきている。このような観点から、多くの低消費電力符号化手法が提案されている [2]~[7]。上で述べたように、一般的なエラー訂正・検出符号は追加の検査ビットを利用するため、その検査ビットにおける信号遷移がバス全体の信号遷移を増加させてしまうため、結果として消費電力を増大させてしまう。電力効率が高く、信頼性の高いデータ伝送方式を調査するために、Bertozi らはワードエラー率と消費エネルギーとのトレードオフを考慮しながら、様々なエラー訂正・検出符号の比較を行った [8]。しかし、そこでは低消費電力符号化の導入は行われていない。

本稿では、信頼性向上のためのエラー耐性の高いデータ伝送符号化において、低消費電力符号化をエラー訂正符号化に組み合わせた場合の、低消費電力符号化の与える影響についての検討を行う。最近の研究 [9] において、Sridhara らは低消費電力符号化、クロストークの影響を抑えた符号化、およびエラー訂正符号化を統一した符号化フレームワークを提案しているが、本稿では命令セットシミュレータを利用して、より現実的な状況下でのエラー訂正符号と低消費電力符号との結合についての評価を行う。また、符号化/復号化回路の評価を行い、ワードエラー率を一定に保つ条件下での、バス信号の信号振幅最適化を導入し、より効率的に両符号化を組み合わせる手法を提案する。

本稿は以下のように構成される。2 節では、本稿で利用するエラー訂正/検出符号と低消費電力符号について述べる。3 節では、SPEC2000 ベンチマークを利用した実験結果および回路レベルでの評価を示し、最後に 4 節でまとめとする。

2. データ伝送のための符号化手法

従来、VLSI チップ内のデータ伝送に関連して、信号遷移頻度の削減 (消費電力削減)、データ圧縮などの数多くの符号化手法が提案されている。本稿では、図 1 に示すようなプロセッサとオンチップメモリとの間のオンチップのメモリバスを対象として扱う。データは MIPS プロセッサと非常に似た構造を持つ SimpleScalar アーキテクチャ [12] のプロセッサからエンコーダを通過して 32 ビットバスといくつかの冗長ビットを持つバスへ送られる。その後、デコーダを通過してメモリへとデータが送られる。エラーがデコーダ側で検出された場合、エラーが訂正可能であった場合にはデコーダでエラーが訂正され、訂正不可能な場合には再度データの伝送が行われる。したがって、エ

ラーに対する耐性は次の 2 つの方式によって実現される。

- **Forward Error Correction:** エラーが訂正可能な場合、ECC を利用してデコーダがデータを訂正する。
- **Retransmission:** エラー訂正が不可能な場合、同じデータが再送される。本稿では、データの再送のためのリクエストの取り扱い手法については、扱わない。

以下で、本稿で利用するエラー訂正符号について述べる。

2.1 エラー訂正/検出符号

一般に、システムの信頼性を高めるために、エラー訂正符号 (Error Correction Code; ECC)、エラー検出符号 (Error Detection Code; EDC) が利用される。パリティ符号、 m -out-of- n 符号 [10]、Berger 符号 [11] などは EDC の一種である。パリティ符号は最も平易なエラー検出符号であり、1 ビットのパリティビットが偶パリティ (あるいは奇パリティ) を示し、1 ビットの双方向性エラーの検出が可能である。 m -out-of- n 符号は n ビットの全ての符号語がちょうど m 個の “1” を含んでいるため、全ての単方向性エラーの検出が可能である。Berger 符号もまた、データビットとデータビット中の “1” の数を示すチェックビットから構成されるため、全ての単方向性エラーの検出が可能である。一般的に、微細化された VLSI システムにおいて考慮されるべきノイズによるエラーは双方向性のエラーであると考えられるため、本稿でも双方向性のエラーを訂正/検出できる ECC/EDC を利用する。特に、パリティ符号、1 ビットエラー訂正符号 (Single Error Correction code; SEC コード)、1 ビットエラー訂正 2 ビットエラー検出符号 (Single Error Correction Double Error Detection code; SECDED コード) SEC コードと SECDED コードはともにハミング符号 [1] をもとした符号である。

以降では、これらの符号化手法の概要を述べる。ここで、 e は各バス配線 1 本当たりのビットエラー発生率を示す。また、空間的・時間的なエラーの相関は無いものとする。 e を利用することで、符号化をしない 32 ビットデータのデータ伝送のワードエラー率 (PE) は次のように表される：

$$PE_{RAW} = 1 - (1 - e)^{32} \quad (1)$$

また、符号化を行わないデータ伝送では、エラーの訂正・検出をすることができないため、データの再送が発生する確率は 0 である。

2.1.1 パリティ符号

パリティ符号は最もシンプルなエラー検出符号の 1 つであり、エラーの発生したビット数が奇数 (例えばエラーが 1 ビット、3 ビット、5 ビット、...) の場合にエラーを検出することができる。逆に、エラーの数が偶数 (2 ビット、4 ビット、6 ビット、...) の場合にはエラーを検出することができず、ワードエラーとなってしまふ。1 ビットのパリティビットを持つパリティ符号の 32 ビットデータに対するワードエラー率は次のように表される：

$$PE_{PAR} = \sum_{i=1}^{16} \binom{33}{2i} e^{2i} (1 - e)^{33-2i} \quad (2)$$

パリティ符号では、デコーダがエラーを検出した場合、送信側に再送のリクエストを送ることで、正しいデータを受信する。

2.1.2 1ビットエラー訂正符号 (Single error correction (SEC) code)

本稿では、1ビットエラー訂正可能な (38, 32) ハミングコードを利用した。この符号では、任意の符号語間のハミング距離が3の倍数であるため、1ビットのエラーの訂正は可能であるが、2ビット以上のエラー検出はできない。(38, 32) ハミング符号を利用した SEC コードのワードエラー率は次のように表される：

$$PE_{SEC} = \sum_{i=2}^{38} \binom{38}{i} e^i (1-e)^{38-i} \quad (3)$$

SEC コードはエラー訂正不可能なエラーを検出することができないため、データの再送が発生する確率は0である。

2.1.3 1ビットエラー訂正 2ビットエラー検出符号 (Single error correction and double error detection (SECDED) code)

SECDED コードは SEC コードをもとに容易に拡張可能である。この符号では、任意の符号語間のハミング距離が4の倍数であり、1ビットのエラーを訂正でき、また4の倍数を除く偶数ビット (例えば、2ビット、6ビット、10ビット、...) のエラーを検出することが可能である。結果として、エラーの数が1を除く奇数および4の倍数の場合にエラーを検出することができない。132ビットデータ+7ビットパリティビットの SECDED コードのワードエラー率は次のように表される：

$$PE_{DED} = \sum_{i=1}^{19} \binom{39}{2i+1} e^{2i+1} (1-e)^{39-(2i+1)} + \sum_{i=1}^9 \binom{39}{4i} e^{4i} (1-e)^{39-4i} \quad (4)$$

2.2 低消費電力符号化手法

冒頭で述べたように、多くの低消費電力符号がすでに提案されている。それらの多くは、データ伝送に冗長性を導入することで、バス配線の信号遷移頻度を削減することで消費電力削減を実現している。本稿では、エンコーダ/デコーダ回路のオーバーヘッドを考慮して、最もシンプルなバス反転符号化[2]とそこから派生した符号化を利用する。以下にそれらの概要を述べる。

2.2.1 バス反転 (BI) 符号

ここでは、下記のような表現を用いる：

Data(t): 時刻 t における伝送されるデータの値
Bus(t): 時刻 t におけるバス上の符号化された値
 バス反転符号化[2]は M. R. Stan らによって提案された符号であり、1ビットの冗長ビットが N ビットのバスに付加される。**Data(t)** と **Bus(t-1)** のハミング距離が $N/2$ を越えた場合、**Data(t)** は元の値を反転され、 $\overline{\text{Data}(t)}$ へと符号化される。また、 $N/2$ を越えない場合には、もとの **Data(t)** へと符号化される。1ビットの冗長ビットは伝送された **Bus(t)** が反転されているか否かを示す。図2にバス反転符号化のエンコーダを示す。データバスに対してバス反転符号化を利用することで、信号遷移数を約10-20%削減することが可能である。

2.2.2 バス反転*(BI*)符号

バス反転*符号はバス反転符号の派生的一种である。バ

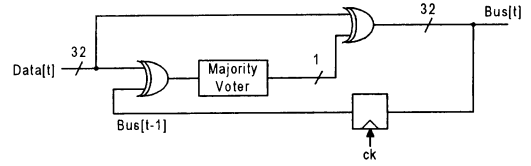


図2 バス反転符号化のエンコーダ
Fig. 2 The encoder of bus invert encoding.

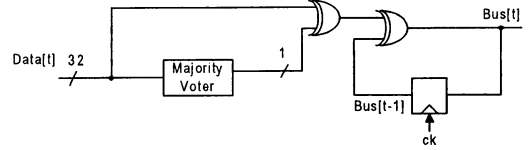


図3 バス反転*符号化のエンコーダ
Fig. 3 The encoder of bus invert* encoding.

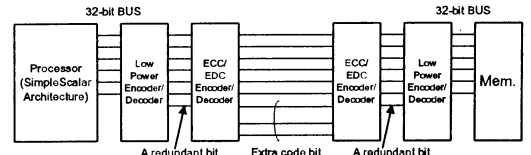


図4 エラー訂正/検出符号と低消費電力符号との組み合わせを利用したアーキテクチャ
Fig. 4 The architecture which exploits the combination of ECC/EDC code and low power code.

ス反転符号と同様に、1ビットの冗長ビットが付加され、**Data(t)** 中の“1”の数が $N/2$ を越えた場合 **Data(t)** は $\overline{\text{Data}(t)} \oplus \text{Bus}(t-1)$ へと符号化される。 $N/2$ を越えない場合には、**Data(t)** は $\text{Data}(t) \oplus \text{Bus}(t-1)$ へと符号化される。図3はバス反転*符号のエンコーダを示す。

バス反転符号、バス反転*符号ともに非常にシンプルであり、他の低消費電力符号化と比較してハードウェアオーバーヘッドが非常に小さい。

2.3 エラー訂正/検出符号と低消費電力符号との組み合わせ

図4はエラー訂正/検出符号と低消費電力符号との組み合わせを利用したアーキテクチャを示す。信頼性を低下させないために、ECCあるいはEDCのエンコーダの出力はバスを介して直接 ECC/EDC のデコーダに接続されていなければならない。エンコーダ/デコーダで追加の電力が消費されてしまうが、非常に大きな容量を持つバスの信号遷移頻度を削減することにより、信号伝送自体の電力を全体として削減することが可能である。

2.3.1 各符号化手法のワードエラー率

エラー訂正に関する4種類の符号化 (符号化なし (RAW)、パリティコード (PAR)、1ビットエラー訂正符号 (SEC)、1ビットエラー訂正2ビットエラー検出符号 (SECDED)) と消費電力削減に関する3種類の符号化 (符号化なし (no-enc)、バス反転符号 (BI)、バス反転*符号 (BI*)) を組み合わせることにより、12種類のデータ伝送手法について検討を行う。表1はそれぞれの符号のバス配線の数を示す。式(1)、式(2)、式(3)、式(4)、表1より、エラー訂正と低消費電力が組み合わせられたそれぞれ

表 1 12 種類の符号のそれぞれのバス配線の数

Table 1 The number of bus lines in the 12 kinds of codes

	RAW	PAR	SEC	DED
no-enc	32	33	38	39
BI	33	34	39	40
BI*	33	34	39	40

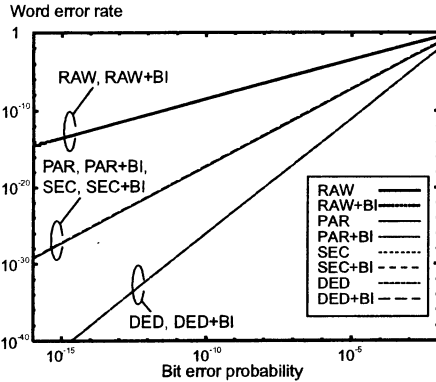


図 5 それぞれの符号におけるビットエラー率とワードエラー率の関係
Fig. 5 Relation between word error rate of each encoding method and bit error rate.

の符号について、バス配線の数を変更することで、容易にワードエラー率を求めることができる。図 5 にそれぞれの符号のワードエラー率を示す。ここで、バス反転符号とバス反転*符号との間にはバス配線の数に違いが無い、両者の間にワードエラー率の違いが無い。したがって、この図では、バス反転*符号については図示していない。さらに、バス反転符号の有無によるワードエラー率の相違はほとんど無いことがわかる。例えば、PAR と PAR+BI はほとんど同じ信頼度を持つ。また、PAR と SEC についても両者のワードエラー率はほとんど同一である。これは、ビットエラー率が小さいため、PAR 符号と SEC 符号において、ワードエラー率は 2 ビットのエラーの発生する確率にほぼ等価であるためである。

2.4 エラーモデルとバス振幅最適化

本稿では、VLSI バス配線上のエラーのモデルとして、ノイズの振幅が CMOS ゲートの閾値を越えた際にエラーが発生するというモデルを前提とする。さらに単純化のため、CMOS ゲートの閾値を $V_{BUS}/2$ とし、ノイズはガウス分布を持つとする。これらの前提は、Hegde らが [13] にて利用した仮定と同一である。これらの仮定のもと、ビットエラー率は次のように表すことができる：

$$e = Q\left(\frac{V_{BUS}}{2\sigma_N}\right) \quad (5)$$

この式で、 $Q(x)$ は標準ガウス分布関数であり、 σ_N^2 はノイズ電圧の分散である。

このエラーモデルと前述の式 (1)、式 (2)、式 (3)、式 (4)、を利用することで、ワードエラー率を求めることが可能である。また、ワードエラー率を一定の値に保ちながらバスの信号振幅を最適化することが可能である。例えば、図 5 から、 $PE_{RAW}(e)$ は $PE_{ENC}(e)$ より大きいことがわかるが、その場合、 $PE_{RAW}(e_{raw})$ と $PE_{ENC}(e_{enc})$ が等しく、かつ、

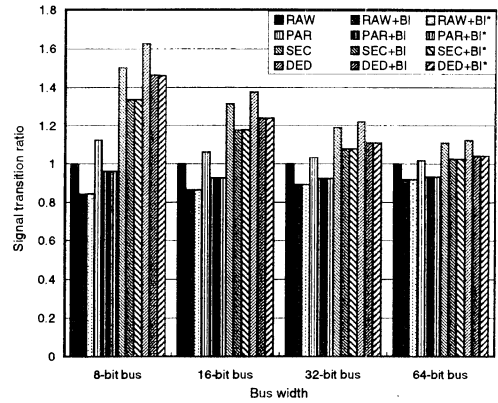


図 6 8, 16, 32, 64 ビットのランダムデータに対するそれぞれの符号の符号化なしと比較した信号遷移頻度の比
Fig. 6 Signal transition ratio of each encoding method for 8-, 16-, 32-, and 64-bit bus.

$e_{raw} < e_{enc}$ となるような e_{raw} と e_{enc} が存在する。そのような観点から、次のようにバス信号振幅を低減することができる：

$$V_{BUS,enc} = V_{BUS,raw} \frac{Q^{-1}(e_{enc})}{Q^{-1}(e_{raw})} \quad (6)$$

CMOS 回路の消費電力が信号振幅の 2 乗に比例するため、バス信号振幅の最適化は消費電力削減のために非常に有効である。通常の VLSI システムにおいて、 V_{BUS} は何通りかの離散値から選択される必要があるが、ここではそれぞれの符号において任意の連続値から選択することができると仮定する。

3. 実験・評価

図 6 にそれぞれの符号化を行ったデータ伝送と符号化なしのデータ伝送との間での信号遷移頻度の比を示す。入力データは、完全にランダムなデータ系列とし、バス幅は 8, 16, 32, 64 ビットのそれぞれについて実験を行った。この図より、ECC/EDC 符号を適用した際に、よりビット幅の大きなバスにおいて信号遷移数が小さくなっていることがわかる。これは、ビット幅の大きなバスでは、冗長ビット数が相対的に小さくなるからである。例えば、SEC 符号では、8 ビットバスで 4 ビット、16 ビットバスで 5 ビット、32 ビットバスで 6 ビットの冗長ビットが必要となる。また、入力データが完全にランダムなデータ系列であるため、バス反転符号とバス反転*符号とがほとんど同じ値となっている。

3.1 実験環境

低消費電力かつ高信頼性な符号化の効果を評価するために、これらの符号化アルゴリズムについて C 言語での実装を行った。また、命令セットシミュレータとして SimpleScalar [12] アーキテクチャシミュレータを利用し、いくつかのベンチマークに対してメモリアクセスのトレースを行った。SPEC2000 ベンチマークのうち、gzip, gcc, mcf, ammp, art, bzip2, equake, parser の 8 種類のベンチマークについて、標準パラメータを利用して 100,000,000 サイクルのメモリアクセスのトレース結果を抽出し、利用した。

表 2 各ベンチマークでの信号遷移頻度の比

Table 2 Signal transition ratio with respect to binary encoding for each benchmark.

Benchmark	RAW	RAW +BI	RAW +BI*	PAR	PAR +BI	PAR +BI*	SEC	SEC +BI	SEC +BI*	DED	DED +BI	DED +BI*
ammp	1	0.889	0.620	1.05	0.938	0.666	1.266	1.155	0.874	1.314	1.204	0.920
art	1	0.903	0.760	1.036	0.939	0.791	1.212	1.115	0.948	1.246	1.149	0.979
bzip2	1	0.954	0.737	1.072	1.026	0.809	1.406	1.358	1.116	1.470	1.424	1.187
quake	1	0.810	0.646	1.051	0.859	0.690	1.308	1.116	0.917	1.357	1.168	0.963
gcc	1	0.870	0.698	1.055	0.924	0.754	1.300	1.168	0.993	1.352	1.220	1.044
gzip	1	0.985	0.784	1.067	1.052	0.856	1.368	1.353	1.166	1.427	1.413	1.227
mcf	1	0.784	0.626	1.045	0.828	0.668	1.264	1.045	0.869	1.308	1.088	0.911
parser	1	0.945	0.899	1.062	1.007	0.967	1.335	1.279	1.314	1.391	1.334	1.374
Average	1	0.892	0.721	1.055	0.947	0.775	1.307	1.199	1.024	1.358	1.250	1.076

3.2 実験結果

表 2 に各符号における信号遷移頻度の比を示す。この表で、信号遷移頻度の比とは、エラー訂正/検出および低消費電力化の符号化を全く行わない元のデータの信号遷移数に対する各符号の信号遷移数の比である。言い替えると、信号遷移頻度の比が小さければ小さいほど信号遷移頻度の削減が大きいということになる。一方、信号遷移頻度の比が 1 を越えている場合には、ECC/EDC 符号による冗長性の増加により信号遷移数が増加し、低消費電力符号化を行っても元のデータの信号遷移数より大きくなってしまったケースである。また、この表の最下行は全ベンチマークの平均値を示す。この表から、PAR 符号、SEC 符号、DED 符号では、その冗長性から信号遷移頻度が増加していることがわかる。他方、PAR+BI*符号、SEC+BI*符号、DED+BI*符号などの低消費電力高信頼性符号では、信号遷移頻度を削減することが可能であり、それぞれ 33%、13%、9%の信号遷移削減となっている。特に、信号遷移削減という観点から見ると、バス反転*符号が SimpleScalar アーキテクチャに対して適していると言える。

次に、各符号のエンコーダ/デコーダ回路によるオーバーヘッドについて、消費電力、回路面積の面から評価を行った。回路面積を見積るために、RTL での回路設計を行い、商用の論理合成ツールを利用してゲートレベルのネットリストを生成した^(注1)。ターゲットテクノロジーは 1.8[V] 0.18 μ m CMOS テクノロジーであり、動作周波数の制約を 100[MHz] とした。ワードエラー率については、元の符号化を行わないデータ伝送において $PE_{raw} = 1.0 \times 10^{-14}$ となるように σ_N を設定し、式 6 を用いて各符号におけるバスの信号振幅 (V_{BUS}) を設定した。また、エンコーダ/デコーダ回路については、符号化/復号化プロセスでの信頼性を損なわないために、動作電圧は 1.8[V] のままとした。消費電力については、ゲートレベル回路について PowerCompiler を利用して見積りを行った。これらの結果を表 3 に示す。本稿の主目的がエンコーダ/デコーダ回路の最適化ではないため、それらは論理設計、物理設計の両レベルにおいて十分に最適化されていない。

この表に示すように、よりシンプルな符号のエンコーダ/デコーダ回路のオーバーヘッド (面積、消費電力) が小さく、また、図 5 で高い信頼度を持つ符号ではバスの信号振幅が大きく削減

表 3 それぞれのエンコーダ/デコーダ回路の見積り

Table 3 Estimation of the codec circuit for each encoding method.

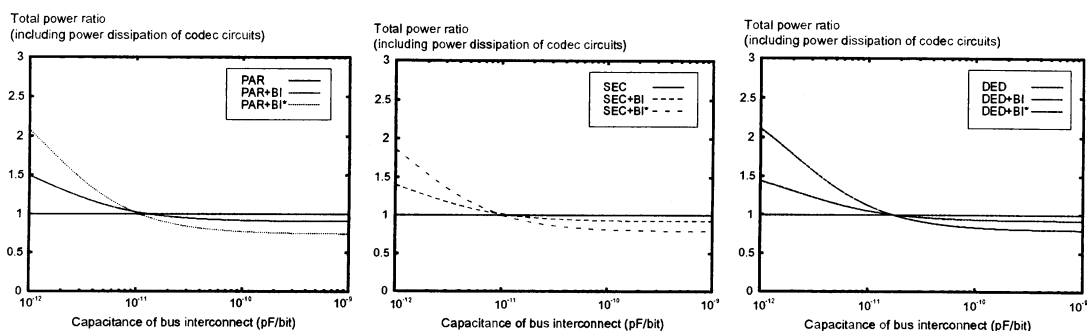
Encoding Method	Area (μm^2)		Power (μW)		Scaling $\frac{V_{BUS,enc}}{V_{BUS,raw}}$
	Enc.	Dec.	Enc.	Dec.	
RAW	0	0	0	0	1
RAW+BI	13378	6412	1038	357	1
RAW+BI*	16941	10252	1125	532	1
PAR	7103	7395	459	476	0.71
PAR+BI	15562	8866	1219	554	0.71
PAR+BI*	19114	12715	1307	751	0.71
SEC	10252	14898	724	1133	0.71
SEC+BI	18662	16346	1449	1209	0.71
SEC+BI*	22207	21602	1546	1362	0.71
DED	11542	18516	835	1383	0.59
DED+BI	23488	19942	1582	1441	0.59
DED+BI*	13636	25218	1662	1615	0.59

されている。

図 7 に、バス配線およびエンコーダ/デコーダ回路の両者の消費電力を含んだ消費電力比を示す。ここで、横軸はバス配線の 1 本当たりの容量である。この図で、“total power ratio” は 1.8[V] の信号振幅のバスを各符号を用いてバスの信号振幅を $V_{BUS,enc}$ へ低減させた場合と全く符号化を行わずにデータ伝送を行った場合との比である。この図から、バス配線の容量が小さい場合、符号化を行わないデータ伝送が最適であり、逆にバス配線の容量が大きい場合、DED+BI*符号が最適であることがわかる。先に述べたように、エンコーダ/デコーダ回路は十分に最適化されていないため、それらの回路での消費電力が大きくなってしまっていると考えられる。したがって、将来の微細化されたシステム LSI では十分に最適化された回路を利用し、DED+BI*符号のような比較的複雑な符号を利用することで、効率的に信頼性の高いデータ伝送を行うことができると考えられる。

図 8 に各高信頼符号における低消費電力符号の影響を示す。図 8(a) において、PAR+BI 符号と PAR+BI*符号の消費電力は PAR 符号の消費電力で規格化されている。また、図 8(b),(c) においても同様に SEC 符号、DED 符号の消費電力によって規格化されている。これらの図から、バス配線の容量が 10[pF] 以上という状況で、バス反転*符号が効率的に消費電力を削減できていることがわかる。実際には、0.18 μ m テクノロジーにお

(注 1) : この目的のために、シノプシス社の DesignCompiler, PowerCompiler を利用した。



(a) Total power ratio normalized by PAR. (b) Total power ratio normalized by SEC. (c) Total power ratio normalized by DED.

図 8 各高信頼符号における低消費電力符号の影響

Fig. 8 Total power reduction including power dissipation of codec circuits normalized by no-low power encodings.

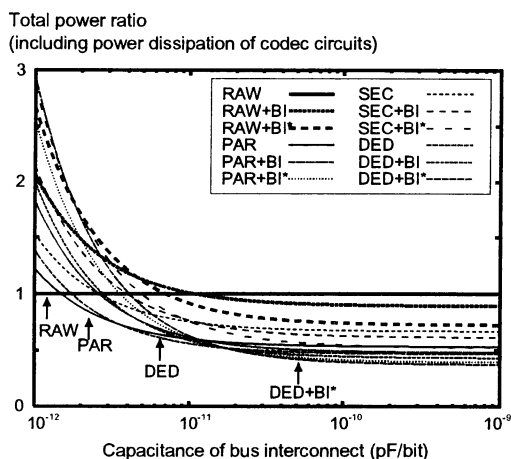


図 7 エンコーダ/デコーダ回路を含んだ消費電力の比

Fig. 7 Total power reduction including power dissipation of codec circuits by using each encoding method.

いて、10[pF] は数 cm に対応するため、非現実的ではあるが、エンコーダ/デコーダ回路が最適化されることでこれらの符号化方式が有効となる下限が下がると期待できる。また、将来の微細化されたテクノロジーでは、論理ゲートと長距離配線との消費電力の差がさらに大きくなると予想され、エンコーダ/デコーダ回路の消費電力の制約が緩和されと考えられる。一方、非常に大きなバス配線の容量を持つオフチップバスなどに対しては、これらの符号が有効であることは明らかである。

4. ま と め

本稿では、高信頼符号に対して低消費電力符号化の与える影響についての検討を行った。SPEC2000 ベンチマークを利用して、アルゴリズムレベルのシミュレーションを行い、高信頼低消費電力符号で信号遷移数が最大 33%削減できることを示した。さらに、エンコーダ/デコーダ回路でのオーバーヘッドおよびバス信号の振幅最適化を考慮した評価を行った。実験結果より、低消費電力符号化と高信頼符号化を組み合わせることで、将来の VLSI システムにおいて有効なデータ伝送方式を実現可

能であることを示した。

謝辞 本研究は東京大学大規模集積システム設計教育研究センターを通し、株式会社日立製作所およびシノプシス株式会社の協力で行われたものである。

文 献

- [1] R. Hamming, "Error detecting and error correcting codes," *Bell System Technical Journal*, Vol. 29, pp. 147-160, 1950.
- [2] M. R. Stan, W. P. Burleson, "Bus-invert coding for low power I/O," *IEEE Transactions on Very Large Scale Integration Systems*, pp. 49-58, Mar. 1995.
- [3] L. Benini, G. De Micheli, E. Macii, D. Sciuto, C. Silvano, "Asymptotic zero-transition activity encoding for address buses in low-power microprocessor-based systems," *IEEE 7th Great Lakes Symposium on VLSI*, pp. 77-82, Mar. 1997.
- [4] L. Benini, G. De Micheli, E. Macii, D. Sciuto, C. Silvano, "Address bus encoding techniques for system-level power optimization," *Design Automation and Test in Europe*, pp. 861-866, 1998.
- [5] L. Benini, A. Macii, E. Macii, M. Poncino, R. Scarsi, "Synthesis of low-overhead interfaces for power-efficient communication over wide buses," *Design Automation Conference '99*, pp. 128-133, Jun. 1999.
- [6] P. Sotiriadis, A. Wang, A. Chandrakasan, "Transition pattern coding: An approach to reduce energy in interconnect," *European Solid-State Circuit Conference*, Sep. 2000.
- [7] S. Ramprasad, N. R. Shanbhag, I. N. Hajj, "A coding framework for low-power address and data buses," *IEEE Transactions on Very Large Scale Integration Systems*, pp. 212-221, Vol. 7, No. 2, 1999.
- [8] D. Bertozzi, L. Benini, G. D. Micheli, "Low power error resilient encoding for on-chip data buses," *Design Automation and Test in Europe 2002*, pp. 102-109, 2002.
- [9] S. R. Sridhara and N. R. Shanbhag, "Coding for System-on-Chip Networks: A Unified Framework," *Design Automation Conference 2004*, pp. 103-106, 2004.
- [10] T. V. Ramabadran, "A coding scheme for m -out-of- n codes," *IEEE Transactions on Communications*, pp. 1156-1163, Vol. 38, No. 8, August, 1990.
- [11] J. M. Berger, "A note on error detecting codes for asymmetric channels," *Information and Control*, Vol. 4, pp. 68-73, Mar. 1961.
- [12] <http://www.simplescalar.org/>
- [13] R. Hegde, N. R. Shanbhag, "Toward achieving energy efficiency in presence of deep submicron noise," *IEEE Transactions on Very Large Scale Integration Systems*, pp. 379-391, Vol. 8, No. 4, Aug. 2000.