

多値2線符号化に基づく高性能非同期データ転送VLSI

高橋 知宏[†] 水澤 一泰[†] 羽生 貴弘[†]

[†] 東北大学 電気通信研究所 〒980-8577 仙台市青葉区片平 2-1-1

E-mail: †{takahasi,mizusawa,hanyu}@ngc.riec.tohoku.ac.jp

あらまし モジュール間で双方向にデータ転送が可能な、1相2線符号化に基づく双方向非同期データ転送方式とその回路実現法を提案する。双方のデータと制御信号を全て同一配線上で多重化することで、配線数を増加させることなく2線のみで双方向データ転送が可能となる。加算によって多重化された符号が減算を用いることで復元することができる性質に着目し、同一の符号表現に複数の有効状態を割り当てることで、データの有効状態数の増加にもかかわらず信号の多値レベルの増加は最小限に抑えられている。この結果、配線あたりのスループットを大幅に向上させることができ、ビットあたりの消費電力量も大きく削減することができることを明らかにする。

キーワード 双方向データ転送, ネットワークオンチップ, システムオンチップ, ディレイインセンシティブ

High-Performance Asynchronous Data Transfer VLSI Based on Multiple-Valued Dual-Rail Encoding

Tomohiro TAKAHASHI[†], Kazuyasu MIZUSAWA[†], and Takahiro HANYU[†]

[†] Research Institute of Electrical Communication, Tohoku University

Katahira 2-1-1, Aoba-ku, Sendai 980-8577, Japan.

E-mail: †{takahasi,mizusawa,hanyu}@ngc.riec.tohoku.ac.jp

Abstract A full-duplex asynchronous communication scheme based on one-phase dual-rail encoding is proposed for on-chip high-speed communication. Since control signals and data from mutual modules are multiplexed using a multi-level dual-rail codeword, full-duplex data transfer without clocks can be realized by using only two wires; one wire per bit. As compared to uni-directional scheme, the increased level of multiple-valued signal is slight because different valid states are assigned on a common codeword. As a result, it is evaluated by using a 0.18- μm CMOS technology that the throughput and energy dissipation of the proposed asynchronous scheme attains a value of 1.18 Gb/s/wire and 0.78 pJ/bit, respectively, at a wire length of 1 mm.

Key words Duplex communication, Network-on-chip(NoC), System-on-a-chip(SoC), Delay-insensitive.

1. まえがき

近年の半導体微細化技術の進展に伴い、システムオンチップ (System on a Chip : SoC) に代表されるように複数の機能モジュールが1つのチップに集積されるようになる一方、クロックスキューやクロック分配に伴う消費電力の増加など配線に起因する問題が深刻となっており、従来の同期式制御ではチップの性能向上が困難となってきた。これらの問題を本質的に解決する一手法として、クロックを用いない非同期式制御が知られており、平均遅延による高速化や適応動作による低消費電力化などの効果が期待される [1]。とりわけ同期式制御で動作しているモジュール間のデータ転送を非同期式制御で行う局所同期大局非同期制御 (Globally Asynchronous Locally

Synchronous:GALS) の考え方は、同期式制御と非同期式制御の双方の利点を兼ね備える方式である [2]。しかしながら、非同期式制御でデータ転送を行う場合、要求応答処理に伴い原理的に制御信号を往復させなければならず1回のデータ転送にかかる遅延が大きくなってしまふ。またクロックに代わる制御信号をデータに付加する必要があり、配線数が増加してしまうことが問題となっている。非同期式制御の利点を最大限活かすためには、非同期式制御に伴うオーバーヘッドを極力削減することが望まれる。

著者らは、送信側と受信側の間で行われる送受信要求とその一致検出を同時に行うことで高速な非同期データ転送を可能とする、1相2線符号化に基づく非同期データ転送方式とその回路実現法を提案してきた [3], [4]。1相2線符号では送信側と受

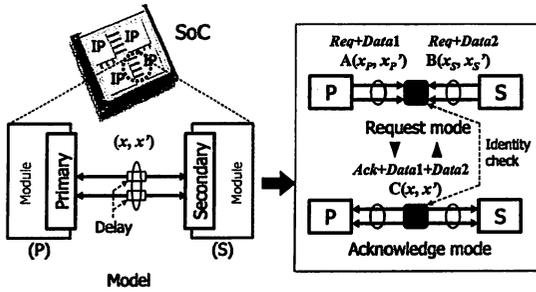


図1 チャネルモデルとプロトコル
Fig. 1 Channel model and protocol.

信側双方から送出される制御信号をモジュール間を接続する配線上で加算し、その加算結果を用いることで相手モジュールの状態を検出する仕組みとなっており、配線数を増加させることなく要求応答に対する処理ステップ数を減少させ高速化が可能となる。符号は線形和と単調変化に着眼しており、多値レベルの信号を用いても任意の遅延条件で正しくデータの有効状態を識別することができる。電流モード多値回路技術を活用することで、双方からの信号を電流レベルで表現して同一配線線に送出し、互いに結線されている配線線上で再び分流させることで、線形加算と加算結果の検出を同時に実行するものとなっており、提案方式を極めて簡単に実現できる。

本稿ではさらなる高性能化を指向し、送信側からのみならず受信側からもデータを転送する双方向非同期データ転送を実現する新しい1相2線符号について述べる。多値符号化を徹底活用し、受信側であったモジュールからのデータも配線線に多重化することで、配線数を2線のまま増やさずにデータの転送量を倍増させビットあたりの配線数を削減する。加算により多重化された信号から自身の信号を減算することで相手の信号を復元できることに着眼し、同一の符号表現に複数の有効状態を割当てて、双方向化に伴う多値レベルの増加を非常にわずかに抑えている。インタフェース回路の試作および評価を通して提案方式が実現可能であることを示し、性能比較を行い従来の非同期制御方式と比較して極めて高性能な非同期データ転送が実現できることを明らかにする。また、複数チャネルを用いたビット幅の広いデータ転送インタフェースの構成法についても述べる。

2. 双方向データ転送に拡張した1相2線符号

図1に双方向非同期データ転送のチャネルモデルとプロトコルを示す。双方向データ転送では送信側のデータ(Data1)のみならずこれまではデータを受け取るだけであった受信側からもデータ(Data2)を送出する。双方向データ転送では送信側と受信側の区別がなくなるが、ここでは説明の便宜上片側のモジュールをプライマリ(Primary:P)としもう片側をセカンダリ(Secundary:S)と呼ぶ。

提案方式ではプロトコルは双方のモジュールから状態を表す要求情報を送出し、その一致を検出することで非同期データ転

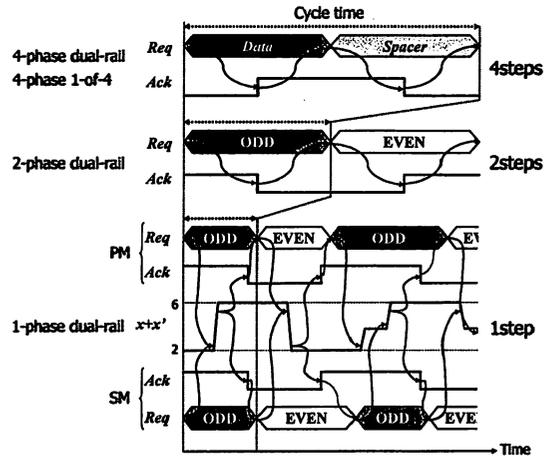


図2 非同期データ転送における処理ステップ数の比較
Fig. 2 Comparison of communication steps.

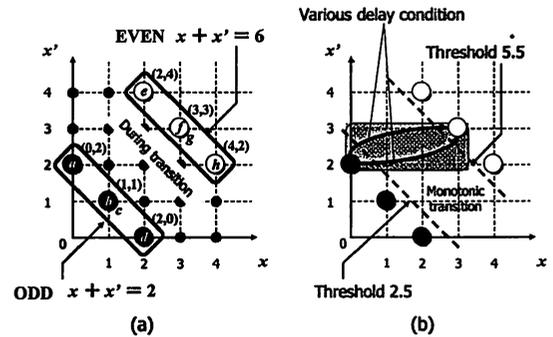


図3 符号の座標平面配置
Fig. 3 Assignment of codewords.

送を行う。図1のように要求モードでは、双方のモジュールからデータと要求情報を重畳させ、2線符号A、Bとして送出し、応答モードでは双方の要求情報が一致したか否かを判定し、その結果を2線符号Cとして返信する。双方で要求情報一致を検出し非同期動作の依存関係を保ちつつも、双方のモジュールは時間的に同時並行で動作できるため、図2のように要求応答に対する処理ステップ数を減少させることができ高速化が実現できる。

1相2線符号は表1(a)に示すように、「データ」(論理値:「0」or「1」)はディファレンシャル信号で、「要求情報」(「奇数」or「偶数」)はコモンモード信号で定義される。ここでプライマリのデータ信号をPIN、要求信号をPREQとし、セカンダリのデータ信号をSIN、要求信号をSREQとする。データと要求情報は符号成分の線形加算によって多重化し、プライマリとセカンダリからそれぞれA(x_P, x'_P)、B(x_S, x'_S)として送出され、双方の符号もまた成分の線形加算によって重畳される。

$$A(x_P, x'_P) = (x_D + x_R, x'_D + x'_R)$$

$$B(x_S, x'_S) = (x_D + x_R, x'_D + x'_R)$$

表 1 1 相 2 線符号 (双方向データ転送用)

Table 1 1-phase dual-rail encoding for full-duplex communication.

(a) Definition

Data (x_D, x'_D)	Req (x_R, x'_R)
"0"	(0,1) ODD (0,0)
"1"	(1,0) EVEN (1,1)

(b) Encoding process

Primary		Transmission lines		Secondary		
PREQ	PIN	A(x_P, x'_P)	C(x, x')	B(x_D, x'_D)	SIN	SREQ
ODD	"0"	(0,1)	$\rightarrow a(0,2) \leftarrow$	(0,1)	"0"	ODD
	"0"	(0,1)	$\rightarrow b(1,1) \leftarrow$	(1,0)	"1"	
	"1"	(1,0)	$\rightarrow c(1,1) \leftarrow$	(0,1)	"0"	
EVEN	"1"	(1,0)	$\rightarrow d(2,0) \leftarrow$	(1,0)	"1"	EVEN
	"0"	(1,2)	$\rightarrow e(2,4) \leftarrow$	(1,2)	"0"	
	"0"	(1,2)	$\rightarrow f(3,3) \leftarrow$	(2,1)	"1"	
	"1"	(2,1)	$\rightarrow g(3,3) \leftarrow$	(1,2)	"0"	
EVEN	"1"	(2,1)	$\rightarrow h(4,2) \leftarrow$	(2,1)	"1"	EVEN

(c) Decoding process

Primary		Secondary			
POUT	(x, x') - (x_D, x'_D)	C(x, x')	(x, x') - (x_D, x'_D)	SOUT	
"0"	0<1 (0,1)	$\leftarrow a(0,2) \rightarrow$	(0,1)	0<1	"0"
"1"	1>0 (1,0)	$\leftarrow b(1,1) \rightarrow$	(0,1)	0<1	"0"
"0"	0<1 (0,1)	$\leftarrow c(1,1) \rightarrow$	(1,0)	1>0	"1"
"1"	1>0 (1,0)	$\leftarrow d(2,0) \rightarrow$	(1,0)	1>0	"1"
"0"	2<3 (2,3)	$\leftarrow e(2,4) \rightarrow$	(2,3)	2<3	"0"
"1"	3>2 (3,2)	$\leftarrow f(3,3) \rightarrow$	(2,3)	2<3	"0"
"0"	2<3 (2,3)	$\leftarrow g(3,3) \rightarrow$	(3,2)	3>2	"1"
"1"	3>2 (3,2)	$\leftarrow h(4,2) \rightarrow$	(3,2)	3>2	"1"

$$C(x, x') = (x_P + x_S, x'_P + x'_S)$$

表 1(b) は双方向データ転送における符号の有効状態をまとめており、データと要求情報の組合せによって a~h の 8 つの有効状態が存在する。データの符号は論理値の値によらず成分の和が一定なのに対し、要求情報の符号は「奇数」と「偶数」で成分の和が 2 変化するため、双方の要求一致は符号の和 $x + x'$ を観測すればよい。プライマリとセカンダリの双方の要求情報が「奇数」なら和は最小値 2 をとり、「偶数」なら和は最大値 6 となる。状態遷移の際、符号の和は最小値と最大値の間を単調変化することから、要求の一致はしきい値 2.5 と 5.5 のしきい値演算を用いることで検出できる。

双方向化にあたって重要な点は和の性質のみならず、双方の信号が多重化された和から自身の信号の差を取れば相手の信号を復元することができる差の性質を活用することである。

$$(x_S, x'_S) = (x - x_P, x' - x'_P)$$

$$(x_P, x'_P) = (x - x_S, x' - x'_S)$$

データはディファレンシャル信号で定義されているので、減算により得られた符号の成分同士を大小比較することによって、相手のデータを復号することができる。表 1(c) は各有効状態における復号処理をまとめたものである。ここで、プライマリで復号される信号は POUT, セカンダリで復号された信号は

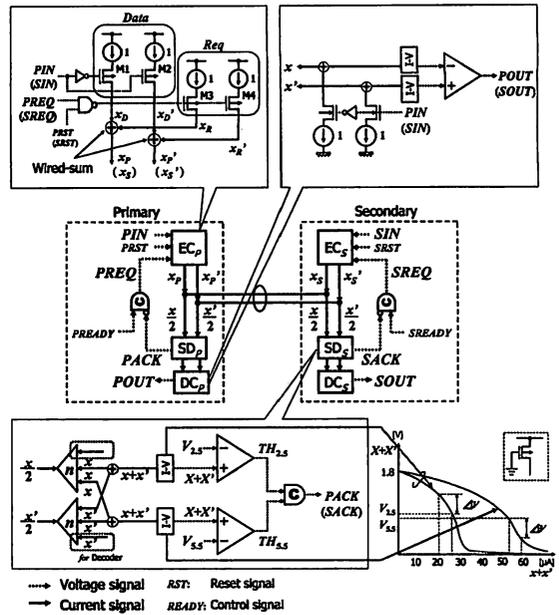


図 4 インタフェース回路の構成

Fig. 4 Circuit diagrams of interface.

SOUT としている。

図 3(a) に符号の有効状態を座標平面上に配置したものを示す。有効状態 b と c および f と g がそれぞれ同一の符号表現で表されていることがわかる。差の性質を用いることで同一の符号表現に複数の有効状態を割り当てることができ、そのため双方向化に伴う和 $x + x'$ の多値レベルの増加は単方向の時と比較して高々 1 である。これはハードウェア実現の際に、双方向化によるオーバーヘッドが非常にわずかで済むことを意味し、高性能化に大きく貢献する。さらに図 3(b) のように状態遷移の際、2 線の遅延がばらついた場合でも、2 線の遷移の軌跡が描く領域内には出発点と到達点の有効状態しか含んでいないことがわかる。これは双方向化しても単方向のときと同様に、任意の遅延条件において一意に有効状態を決定できることを示している。

3. 双方向非同期データ転送インタフェースの回路構成

電流モード多値回路は 1 線上に多レベルの信号を重畳でき、線形加算が結線のみで実現できるという特長を有している。この特長に着目し、双方からの信号を電流レベルで表現して同一配線に送出し、互いに結線されている配線上で再び分流させることで、線形加算と加算結果の検出を同時に実行でき、提案プロトコルが極めて簡単に実現できる。

図 4 に双方向非同期データ転送インタフェースのブロック図と回路構成を示す。双方のインタフェースは符号化部 (EC)、状態検出部 (SD)、復号化部 (DC) から成り、伝送路で結ばれている。符号化部では 1bit の入力データ信号 PIN (SIN) と要求信号 PREQ (SREQ) によって制御トランジスタを切り換え電流の線形加算により 2 線符号 A(x_P, x'_P) (B(x_S, x'_S)) に対応

表2 性能比較

Table 2 Comparison of performance.

	4-phase dual-rail encoding	4-phase 1-of-4 encoding	2-phase dual-rail encoding	1-phase dual-rail encoding	1-phase dual-rail encoding [4]
Communication channel	Duplex				Uni-directional
Number of communication steps	4	4	2	1	1
Number of wires	6	5	6	2	2
Circuit realization	Binary-CMOS	Binary-CMOS	Binary-CMOS	MVCML	MVCML
Cycle time [ns]	1.78	2.00	0.88	0.85	0.94
Circuit delay	1.08	1.28	0.53	0.47	0.50
Wire delay	0.70	0.72	0.35	0.38	0.44
Throughput [Gb/s/wire]	0.19	0.20	0.38	1.18	0.53
Power dissipation [mW]	3.20	1.66	3.42	1.83	1.40
Energy dissipation / Cycle [pJ/bit]	2.85	1.78	1.56	0.78	1.33
Area [μm^2]	2070	2027	1843	2517	2171

HSPICE simulation with a 0.18- μm CMOS technology. Supply voltage=1.8V. Wire length=1mm.

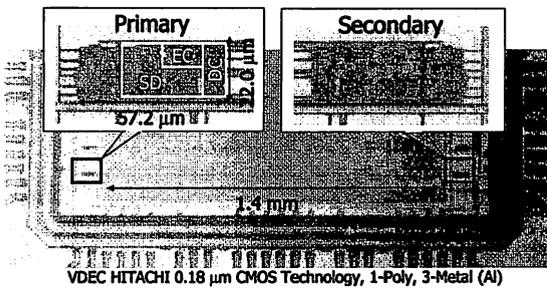


図5 試作チップ写真

Fig. 5 Photomicrograph of a test chip.

する電流信号が生成される。双方からの電流信号は伝送路上で重畳され結線により $C(x, x')$ が得られ、これが双方に分流される。状態検出部では2種類のしきい演算を用いて要求一致を検出する。分流された電流はカレントミラーにより復元および複製され、2線を線形加算することにより電流信号の和 $x + x'$ を得る。電流信号 $x + x'$ は電流-電圧変換回路によって電圧信号 $X + X'$ に変換される。もし、電圧信号 $X + X'$ がしきい値電圧 $V_{2.5}$ よりも小さければ、しきい演算の出力電圧 ($TH_{2.5}, TH_{5.5}$) は (0, 0) となる。もし、電圧信号 $X + X'$ がしきい値電圧 $V_{5.5}$ よりも大きければ、しきい演算の出力電圧 ($TH_{2.5}, TH_{5.5}$) は (1, 1) となる。 ($TH_{2.5}, TH_{5.5}$) はC素子の入力となり、C素子の出力が要求一致の結果、すなわち応答信号 $PACK(SACK)$ である。ここで、C素子とは状態保持素子であり、両方の入力が0なら出力は0に1なら1に変化するが、それ以外の組合せの入力では出力は前の値を保持する回路である。復号化部では自身のデータを表す入力信号 $PIN(SIN)$ で制御トランジスタを切り換えることで、2線電流信号 (x, x') から自身の2線電流信号を引き算している。そして、その結果を比較回路で大小比較をすることで相手のデータを出力データ信号 $POUT(SOUT)$ として復号できる。制御信号 $PREADY, SREADY$ は2値CMOSモジュールが次のデータを送る準備ができたことをイ

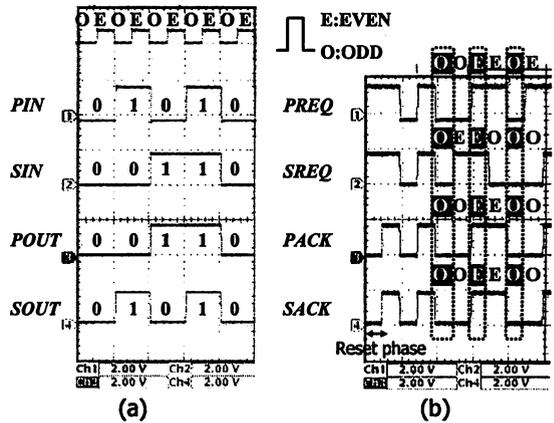


図6 測定波形：(a) 入出力データ信号、(b) 要求応答信号

Fig. 6 Measured waveforms: (a) input and output signals, (b) request and acknowledge signals.

ンタフェースへ伝える信号である。制御信号は全て「奇数」のときはLow、「偶数」のときはHighと定義する。また $PRST, SRST$ はリセット信号でありLowのときはリセット、Highのときは動作を意味する。

しきい演算には高駆動能力を有する差動対回路を用いた電圧比較回路が有用であるが、入力としきい値の電圧差は電源電圧により制約されるため、多値レベルが大きくなるとレベル間の電圧差が小さくなりその高速性が十分に享受できないという問題がある。そこで、前段に挿入する電流-電圧変換回路を、電圧比較回路のしきい値近辺で広い電圧差が取れるように個々に設計を施す。双方向化による多値レベルの増加が高々1であること、そしてこの電流-電圧変換回路の個別設計により、双方向化による遅延の増加はなく高速な検出を可能としている。

双方向化に伴うハードウェアの増加は、従来は受信側であったセカンダリの符号化部の一部と、従来は送信側であったプラ

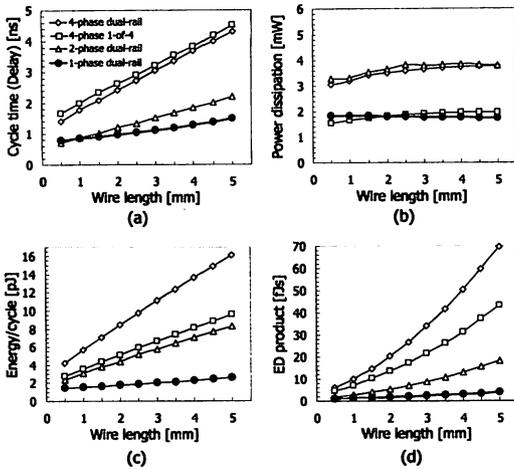


図7 長配線時の性能比較: (a) サイクルタイム, (b) 消費電力, (c) 消費電力量/サイクル, (d) ED 積

Fig. 7 Comparisons of performances: (a) Cycle time, (b) Power, (c) Energy per cycle, and (d) Energy-delay product versus wire length.

イマリの復号化部のみであり、配線数は増加していない。これより、非常にわずかなハードウェアの増加で双方向非同期データ転送インタフェースが実現できる。

4. 性能評価

双方向非同期データ転送方式の動作検証を行うために、0.18- μm CMOS プロセスを用いてインタフェース回路の試作を行った。図5はテストチップの写真である。図6(a) 双方の要求信号に対する入出力データの波形を示したものである。「奇数」と「偶数」のどちらの状態においても、双方から入力されたデータが反対側の出力としてそれぞれ正しく得られていることがわかる。図6(b)は双方の要求信号に対する応答信号の波形を示す。非同期動作を検証するため、双方の要求信号が変化するタイミングを意図的にずらしている。双方の要求が「奇数」もしくは「偶数」に揃ったときのみ、双方の応答が同時に要求一致を検出しており、双方の要求が異なる場合は前の状態を保持（待機）していることがわかる。これにより、所望の動作を確認し提案方式が実現可能であることを実証した。

表2は従来の符号化（4相2線符号、2相2線符号、1-of-4符号）[5]–[7]に基づく双方向非同期データ転送チャンネルと、1相2線符号化に基づく双方向非同期データ転送チャンネルの比較をまとめたものである。提案方式は多値2線符号化の考案により配線数を増加させずに、要求応答処理のステップ数を削減するだけでなく双方からのデータ転送を可能としたことで、配線あたりのスループットは1.18 Gb/s/wireに達し、これは従来方式である4相2線符号化方式の6.2倍である。また、ハードウェアの増加がわずかであることからビットあたりの消費電力量は0.78 pJ/bitとなり、4相2線符号化方式の27%にまで削減できた。配線数は1本/bitで、これは同期制御と同等の配

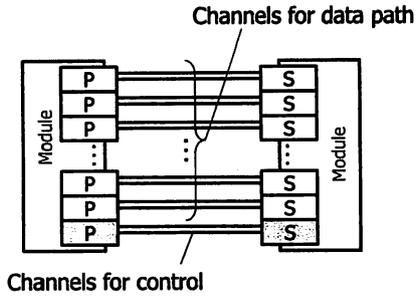


図8 複数チャンネルを用いたインタフェース
Fig. 8 Multi-channel interface model.

表3 モジュールの状態と転送モード
Table 3 Transmission mode.

Primary	Secondary	Mode	
<i>P</i> MODE (<i>CTR</i> ₁)	<i>S</i> MODE (<i>CTR</i> ₂)		<i>STB</i>
Transmission	Transmission	Duplex	H
Transmission	Idle	Uni-directional	H
Idle	Transmission		L
Idle	Idle	Idle	L

Transmission:0
Idle:1
H: High (1)
L: Low (0)

線数で非同期データ転送を実現できることを意味する。単方向方式と比較をすると、ハードウェアの追加により30%程度消費電力が増加しているものの、配線あたりのスループットを2倍以上に引き上げることができ、結果としてビットあたりの消費電力量を59%に削減することができた。

図7は長配線時におけるデータ転送の特性を示したものである。提案方式は要求応答処理のステップ数が少ないので、配線が長くなったときのサイクルタイム（遅延）の増加量が小さい。要求応答処理のステップ数は配線容量の充放電回数にも影響することから、消費電力量においても同様の傾向が見られる。この結果、遅延と消費電力量の積であるED積で評価すると、提案方式は従来方式である4相2線符号化方式の1/18となり、提案方式は極めて高性能な非同期データ転送ができることがわかる。

5. 複数チャンネルを用いたインタフェースの構成

図8に複数チャンネルを用いたビット幅の広いデータ転送インタフェースの構成を示す。提案している双方向非同期データ転送チャンネルはビットシリアル転送を基準としているため、このチャンネルを複数並列に配置することでより帯域幅の広いデータバスを構成することができる。データバスの幅は、モジュール間でやりとりされるデータ転送量の応じてチャンネル数を増減させることで調整する。

インタフェースには双方のモジュール間データ転送の状態に応じて3つのモード、双方向モード、単方向モード、休止モードがある。プライマリとセカンダリはデータを送信している

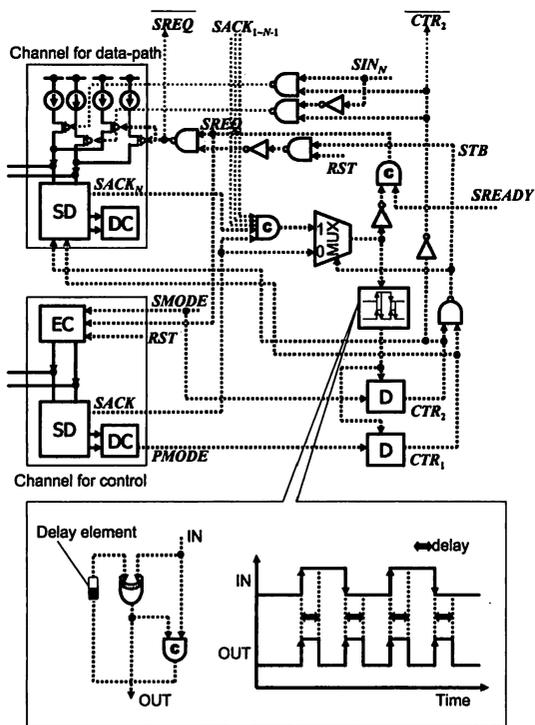


図9 モード切替回路

Fig. 9 Circuit diagrams of control circuit.

状態'であるか否かを、それぞれ送信状態および休止状態として互いに送りあい、双方の状態に応じて3つのモードを表3のように切り替える。この切替制御はデータパスを構成するチャンネルとは別に用意したモード切替用のチャンネルによって制御される。

モード切替用チャンネルはデータパスに用いているチャンネルと全く同じ構成であるが、論理値の「0」の代わりに送信状態を論理値「1」の代わりに休止状態を相手のモジュールに転送する。

図9にインタフェースの制御回路を構成をセカンダリを例にして示す。ここで、 $PMODE$ および $SMODE$ はそれぞれプライマリおよびセカンダリのデータ転送状態を表す制御信号である。 $PMODE$ および $SMODE$ は、「奇数」と「偶数」の両エッジで立ち上がるパルスを生成する回路からの制御信号によってDフリップフロップで保持されそれぞれ CTR_1 および CTR_2 となり、これらの制御信号によって表3に従い次の状態からモードの切替を行う。 CTR_1 , CTR_2 の両方が0の双方向モードのときは回路のどの部分も動作しており、どちらのモジュールもデータの送信および受信動作を行っている。 CTR_1 , CTR_2 の片方が1の単方向モードのときは、状態検出部では CTR_1 , CTR_2 によって単方向方式のしきい演算を行うように回路構成が切り替わり、さらに休止するモジュールの方は自身のデータ (x_D, x'_D) の送信を切断し (0,0) にする。 CTR_1 , CTR_2 の両方が1の休止モードのときは、制御信号 STB が0となり要求情報 (x_R, x'_R) が強制的に「奇数」の状態である (0,0) にな

る。このとき伝送路に流れる電流は完全にカットオフされる。ただし、モード切替用チャンネルは常に動作しており双方のデータ転送状態を常に検出できるようになっている。

6. まとめ

配線数を増加することなく双方からデータ転送が可能となる1相2線符号を提案した。差の性質に着目し同一の符号表現に複数の有効状態を割当ててことで、多値レベルの増加を極めて少なく抑えることが可能となり、結果として配線あたりのスループットの向上およびビットあたりの消費電力量の低減を同時に達成した。インタフェース回路を構成し試作チップにより提案方式を実証するとともに、その有用性を定量的に評価し従来方式と比較して高性能な非同期データ転送が可能であることを示した。

提案方式は配線遅延のばらつきの大きいグローバルなデータ転送に特に有効であることから、チップ内ネットワーク (Network on a Chip: NoC) への適用 [8] や、配線遅延のばらつきがとりわけ顕著となるアプリケーション、例えば LDPC デコーダ [9] などに適用すれば、平均遅延による高速化など非同期制御の利点を大きく享受できる高性能なプロセッサが実現できるものと期待される。

文献

- [1] J. V. Woods, P. Day, S. B. Further, J. D. Garside, N. C. Paver, and S. Temple, "AMULET1: An asynchronous ARM microprocessor," *IEEE Transactions on Computers*, Vol.46, pp.385-398, April 1997.
- [2] J. Mutersbach, T. Villiger, and W. Fitchner, "Practical Design of Globally Asynchronous Locally Synchronous Systems," *Proc. of the 6th IEEE International Symposium on Asynchronous Circuits and Systems*, pp.52-59, April 2000.
- [3] 高橋知宏, 羽生貴弘, 亀山充隆, "双方向同時制御に基づく非同期データ転送方式とそのVLSI実現," 信学論 (C), Vol.J87-C, No.5, pp.459-468, May 2004.
- [4] T. Takahashi and T. Hanyu, "Control Signal Multiplexing Based Asynchronous Data Transfer Scheme Using Multiple-Valued Bidirectional Current-Mode Circuits," *Journal of Multiple-Valued Logic and Soft-Computing*, Vol.11, No.5-6, pp.499-517, 2005.
- [5] J. Sparso and S. Furber, "Principles of Asynchronous Circuit Design," *Kluwer Academic Publisher*, 2001.
- [6] M.E. Dean, T.E. Williams, and D.L. Dill, "Efficient Self-Timing with Level-Encoded 2-Phase Dual-Rail (LEDR)," *Advanced Research in VLSI*, pp.55-70, MIT Press, 1991.
- [7] W. J. Bainbridge and S. B. Furber, "Delay Insensitive System-on-Chip Interconnect using 1-of-4 Data Encoding," *Proc. 7th IEEE Int. Symp. on Asynchronous Circuits and Systems*, pp.74-82, 2001.
- [8] M. Amde, T. Felicijan, A. Efthymiou, D. Edwards and L. Lavagno, "Asynchronous on-chip networks," *IEE Proc. -Computers & Digital Techniques*, Vol.152, No.2, pp.273-283, March 2005.
- [9] C. J. Howland and A. J. Blanksby, "Parallel Decoding Architectures for Low Density Parity Check Codes," *Proc. of the IEEE International Symposium on Circuits and Systems*, pp.IV-742-745, 2001.