社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－ARC－175 (12)
2007／11／22

細粒度命令分解と少品種セルによる高信頼化アーキテクチャの提案

鈴木一範[†1] 中田尚[†1] 中西正樹[†1]
山下茂[†1] 中島康彦[†1]

近年のトランジスタ製造のプロセス微細化によって，トランジスタの性能ばらつきが無視できない問題となっている．ばらつきを抑える手法として，セル内のトランジスタを規則的に配置するという方法がある．本論文では，トランジスタを規則的に配置したセルを提案する．このセルはPMOSとNMOSを組み合わせたものが基本単位として構成されている．我々の提案したセルは，従来のセルに比べて壊れにくく，また故障を検知する機能を備えている．次に，このセルを組み合わせた高信頼性の演算器を提案した．提案した演算器と命令分解機構を組み合わせることにより従来にない高信頼化アーキテクチャが実現できると考えている．

A Highly Reliable Architecture with a Fine-grained Instruction Decomposition and a Small Variety of Standard Cells

Kazunori Suzuki,[†1] Takashi Nakada,[†1] Masaki Nakanishi,[†1]
Shigeru Yamashita[†1] and Yasuhiko Nakashima[†1]

Recently, process deviation causes transistor variation. It is known that transistor variation can be reduced by arranging transistors regularly. In this paper, we propose new standard cells in which transistors are arranged regularity in order to control variation. These cells are composed by pairs of PMOS and NMOS. The proposed cells are more robust against transistor faults and can also detect them. Next, we propese highly reliable arithmetic circuit by using our cells. We also show that highly reliable architecture can be constructed by this circuit and instruction decomposition circuit.

1. はじめに

近年のトランジスタ製造プロセスの微細化によって，チップに搭載されるトランジスタの集積度が急激に増加している．その一方でトランジスタの性能ばらつきが無視できない事態になっている．ばらつき増大はトランジスタの故障率増加に繋がるので，いかにしてばらつきに対処するかということが問題となっている．

この問題に対する一般的な解決策は，図1の（A）のように回路を多重化し，後ろに多数決回路を設置する方法[1]である．しかしこの方法では，(B) のように，故障率増加によって多重化した回路すべてに故障が発生した場合は対処できない．またLSIを製造する立場から見た，ばらつきに対処する方法として，トランジスタを規則的に配置するという方法があげられる．現在のLSIは主にメーカから与えられた基本セルを組み合わせて作成される．基本セルは不規則にトランジスタが配置されており，また種類が多いことから，結果として作成されるLSIもトランジスタが不規則に配置されているものが多い．

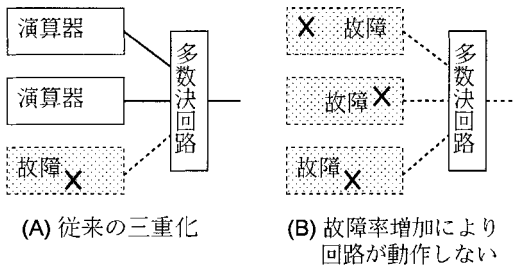

図1 従来の演算器の構成とその問題点

そこで我々はトランジスタが規則正しく配置されており，かつ以下の機能を持ったセルを提案する．

(1) 壊れにくいセル
(2) 壊れた場合，その故障を検出できるセル

また本稿では，このセルを用いて高い信頼性を持つ演算器を構成する手法を提案する．提案する演算器と，OROCHI[2]に搭載されている命令分解機構を使用す

†1 奈良先端科学技術大学院大学
Nara Institute of Science and Technology

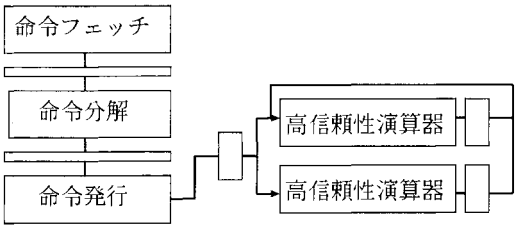

図 2 高信頼性演算器と命令分解による回路構成

ることで従来にない高い信頼性を持つアーキテクチャが実現可能だと考えている．その配置は図 2 のようになる．OROCHI の命令分解機能は現在はプロセッサ内に実装されている．将来的にはメモリを使用したマイクロプログラム方式によって実現される．メモリ上に実装することでトランジスタのばらつき問題をプロセッサ内の演算器のみに集中できる．

本稿では，上記の機能を持ったセルについて提案，評価を行い，セルがトランジスタの故障に対して頑健であること，またこのセルを組み合わせることでどのような性質を持つ演算器が構成可能なのかを示す．

本稿の構成は以下の通りである．まず，2 章ではセルが高い信頼性を持つために必要な機能について説明する．次に 3 章で，我々が提案するセルについて述べる．4 章では HSPICE によるシミュレーションの評価結果を示し，提案したセルが必要な機能を備えていることを示す．5 章で演算器の構成方法について述べ，最後に 6 章でまとめを述べる．

2. 目標とされるセルの機能

本章では，我々の提案したセルが持ち備えている 3 つの機能について説明する．

2.1 壊れにくいセル

図 3 の（A）は，従来のセルの一例を示している．このセルの特徴として，PMOS や NMOS が直列あるいは並列に配置されている．この場合，直列に配置されている MOS の片方が壊れた場合は動作しない．図では，NMOS のどれか 1 つが壊れた場合がそれに該当する．一方（B）は，我々の提案したセルでありPMOS と NMOS を組み合わせたものを基本単位をしてセルが構成されている．PMOS と NMOS を組み合わせることにより，PMOS と NMOS の片方が壊れた場合であっても動作する．つまり，我々の提案したセルは従来のセルより壊れにくい．ただし PMOS と NMOS の両方が壊れた場合は，回路は動作しない．以下，本稿ではセルの故障を，PMOS と NMOS の両方が壊れ，物理的に断線している状態と定義する．

2.2 故障を検出できるセル

現在はセルが壊れたということを検出するために，回路を三重化し，その後ろに多数決回路を設置すると

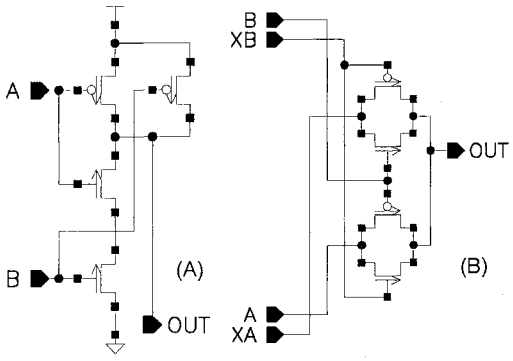

図 3 従来のセルの構成（A）および提案したセルの構成（B）

いう方法が主流である．しかし，我々が提案するセルでは出力信号を調べることで故障を検出できる．

提案したセルでは入力信号，出力信号ともに正論理と負論理の両方を使用する．これにより，従来 2 入力 1 出力だった回路が 4 入力 2 出力となる．このセルにおいて，2.1 節で述べたような故障が発生した場合，正論理と負論理の出力で同じ値が出力される．つまり，出力信号の正論理と負論理の値を調べることで回路内に故障が発生したかどうかが判定できる．以下，正論理と負論理で同じ値が出力されることを「**不正値を出力する**」と定義する．

2.3 検出機能が演算器内に埋め込まれている

我々が提案しているセルには，故障を検出する機能が演算器に含まれている．2.2 節で述べたように，我々のセルは入出力とも正論理と負論理に分かれている．この正論理，負論理に対する入力信号が正しい場合，正しい結果を出力する．しかし入力信号が両方とも 1，もしくは両方とも 0 の場合，不正値を出力する．これは，演算器内の論理回路の任意の段で故障が発生し，2 本の出力信号が不正値だった場合，その不正値を入力信号として使用する次の段の 2 つの出力信号も不正値となることを示している．不正値の出力が繰り返されることによって，最終的には演算器の出力信号における任意のビットが不正値を出力する．つまり，回路の任意の段で故障が発生したという情報が回路を通じて最終段の出力結果まで伝搬する．故障したという情報が最終段まで伝搬するので，従来のように論理回路の任意の段に故障検出回路を設置して故障を検知する必要がない．よって，従来の回路より回路面積を小さくすることが可能であると考えられる．

また，不正値が出力される場合，弱い 0 や弱い 1 といった値が出力されることがある．電源電圧が 1.8V の場合，弱い 0 は 0.2 から 0.5V，弱い 1 は 0.8V から 1.4V の間の電圧を示す．我々の提案したセルは，このような弱い 0 あるいは弱い 1 が入力信号として与えられた場合でも，その信号を 0 あるいは 1 と認識して正しく動作する．

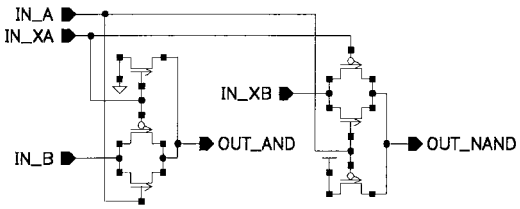

図 4 land

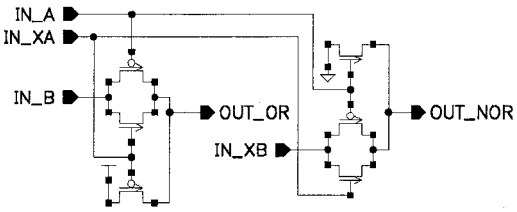

図 5 lor

2.4 従来のセルとの違い

従来のセルおよび回路との違いとして以下の点があげられる．

まず，回路面積が従来の回路より大きいという点がある．これは使用することができるセルが限られているため，回路を構成するために必要なセルの数が増加するからである．しかし，プロセス微細化によって搭載されるトランジスタ数が増加しているので問題はない．

また，回路面積が増大することによって配線が長くなり，データ遅延が増大する可能性がある．しかし，トランジスタを規則的に配置することで，極端に長い配置配線が排除されると考えられる．

3. 提案するセルの構造

本章では 2 章で述べた機能を持つセルを構造するための基本アイデア，および各セルの具体的構造について述べる．

3.1 基本アイデア

まず，各セルは入出力それぞれに正論理及び負論理を持ち，演算は常に正論理と負論理の両方で行う．これは AND や OR といった従来の回路の 2 重化とみなすことができる．

また，エラー訂正のために上記のセルを 2 重化して回路を構成する．回路構成の点だけで見た場合，上記回路は従来の回路を 4 重化したものとなっている．しかしセル自体が効率化されており，1 つのセル内で論理が共有されているので，4 重化より小さい面積で回路を構成できると考えられる．

さらに，弱い 0 あるいは弱い 1 を入力信号に使用することで，多数決回路を用いる必要がなく，故障検出回路を従来より小さい面積で構成できる．

3.2 各セルの具体的構造

図 4 は AND を出力する land，図 5 は OR を出力する lor，図 6 は EOR を出力する leor である．それぞれ入力信号を 4 本，出力信号を 2 本持つ．

図 7 はセレクト回路にあたる lsel2 である．この回路は S に入力される信号から A と B のどちらを出力するかを決める．IN_S=1，IN_XS=0 の場合，A の値を S1 に出力し XA の値を XS1 に出力する．逆に IN_S=0，IN_XS=1 の場合は B を S1 に，XB を XS1 に出力する．

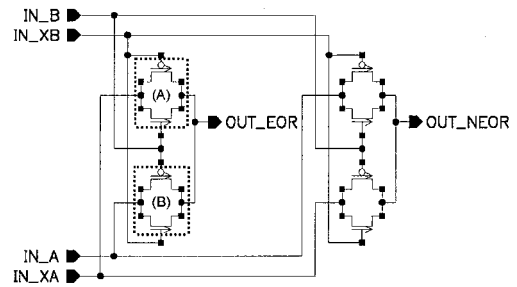

図 6 leor

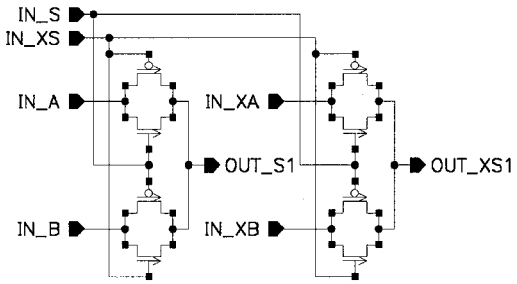

図 7 lsel2

表 1 セルが故障した場合

A	XA	B	XB	EOR1	EOR2
1	0	1	0	0	0
1	0	0	1	1	0
0	1	1	0	1	1
0	1	0	1	0	1

4. セルの性能評価

本章では，land，lor，leor，lsel2 の各セルについて HSPICE を使用し，様々な入力パターンを与えた場合の出力信号について評価する．なお，表中の*0 および*1 は弱い 0 と弱い 1 を示している．

4.1 故障が発生した場合

本節では，回路に故障が発生した場合でも，正しい

値あるいは不正値のどちらかが出力されることを示す．つまり，正論理と負論理両方とも反転した値が出力されることはない．

例として図6の（A）と（B）が故障した場合を考える．（A）が故障した場合の出力信号をEOR1とし，（B）が故障した場合の信号をEOR2とする．また，入力信号のパターンがA，XA，B，XB=［1，0，1，0］，［1，0，0，1］，［0，1，1，0］，［0，1，0，1］の4つを繰り返している時の出力について評価する．

表1に出力結果を示す．回路が故障しているにもかかわらずEOR1は通常のEORと同じ値を出力している．しかしEOR2では間違った値を出力しているパターンが存在する．これは（A）と（B）に対する入力値に依存する．PMOSのゲートに対して0が，NMOSのゲートに対して1が入力されている場合はPMOS，NMOSともスイッチが入った状態になり，ソースに入力された値をそのまま出力する．しかしPMOSのゲートに対して1が，NMOSのゲートに対して0が入力された場合，その直前までソースに入力された値を出力する．表1の場合では，EOR1に対するA，XA，B，XB=［0，1，1，0］のときがそれに該当し，この直前の入力，A，XA，B，XB=［1，0，0，1］の時のPMOSとNMOSに対する入力信号1を出力する，同様にEOR2に対するA，XA，B，XB=［1，0，0，1］のときも，その直前の入力信号0を出力する．

つまり，回路が故障した場合であっても出力信号が正しい場合と間違っている場合が存在する．出力信号が正しい場合は問題はない．しかし，間違った出力信号の場合は不正値を出力する．表1の場合では，EOR2に対するA，XA，B，XB=［1，0，0，1］のときがその場合に該当する．

4.2 不正値が入力された場合

本節では，故障が発生した場合でもその情報が後で検出できる形で伝搬されることを示す．表2に不正値が入力された場合の出力結果を示す．

まず，landについて説明する．入力信号のパターンのうち不正値を含む入力信号に対して，出力信号が2通り存在する．1つは不正値を出力する場合であり，6パターンが該当する．もう1つは正論理と負論理の出力信号が異なる場合であり，表2のA，XA，B，XB=［0，0，0，1］など6パターンが該当する．この6パターンのうちA，XA，B，XB=［0，0，0，0］とA，XA，B，XB=［1，1，1，1］を除く4パターンは，片方に対する入力信号が正しく，出力信号がその片方の入力信号に依存する場合である．

例えば，A，XA，B，XB=［0，0，0，1］という入力信号はBに対する入力信号が正しい場合でありB=0とみなせる．ANDは片方の入力が0の場合，もう片方の入力にかかわらず0を出力し，逆にNANDは片方の入力が0の場合1を出力する．この入力信号に対してAND=*0，NAND=1という値が出力されているので，これは正しい値が出力されているといえる．他にもA，XA，B，XB=［0，1，0，0］の場合は入力信号のうちA=0とみなせるのでその出力信号であるAND=0，NAND=1も正しい値である．つまり，この4パターンは2つある入力信号のうち片方だけ正しい場合の正しい出力信号であるといえる，

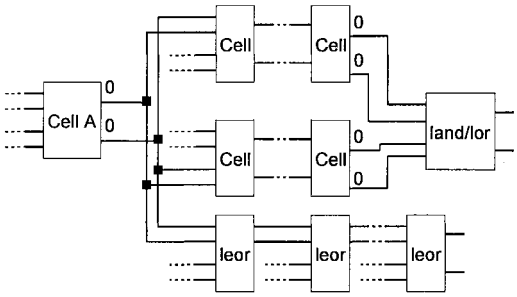

図8 land/lorの入力パターンチェック

lorも同様の出力結果となり，A，XA，B，XB=［0，0，0，0］とA，XA，B，XB=［1，1，1，1］を除き正しい値を出力する．正論理と負論理の出力信号が異なる場合も，片方の入力信号が1である場合の値を出力する．

しかし，land，lor両方とも入力信号がA，XA，B，XB=［0，0，0，0］とA，XA，B，XB=［1，1，1，1］の場合は，入力信号のA，Bが両方とも間違った値であり，出力信号も間違った値を出力しなければならない．しかし，出力信号の結果だけでは入力信号が間違っていることを判定できない．そこでこの2つの入力信号パターンを検出するために，図8のように回路を設計する．

図8について説明する．まずCell Aで不正値が出力される．次に，出力された不正値が分岐され2つ以上の異なるセルに入力される．land，およびlor以外のセルに不正値が入力された場合，出力信号も不正値となる．よって，不正値が後続のセルに伝搬される．最後に，分岐したセルの不正値がlandもしくはlorに入力される．以上の場合にのみland/lorにA，XA，B，XB=［0，0，0，0］もしくはA，XA，B，XB=［1，1，1，1］の信号パターンが入力される．以下，Cell Aのように出力信号が分岐し，再び収束してland，もしくはlorの入力信号になるようなセルを「**再収斂セル**」と呼ぶ．

これを回避するために，再収斂セルの出力信号をleorに接続する．再収斂セルは1つの演算器内に複数個存在するので，ツリー状に構成したleorを用意し，そのツリーの入力信号として再収斂セルの出力信号を与える．ツリー状のleorの最終段から出力される値が正負論理とも同じ場合は故障は発生していない．しかし，正論理と負論理の値が異なる場合は，再収斂セルのどれかに不正値が入力されたと判定できる．

表 2　不正値が入力された場合

				land		lor		leor		lsel2(S=0)		lsel2(S=1)	
A	XA	B	XB	AND	NAND	OR	NOR	EOR	NEOR	S1	XS1	S1	XS1
0	0	0	0	*0	*1	*1	*0	*0	*0	0	0	0	0
0	0	0	1	*0	1	*1	1	0	0	0	1	0	0
0	0	1	0	1	*1	1	*0	0	0	1	0	0	0
0	0	1	1	1	1	1	1	0	0	1	1	0	0
0	1	0	0	0	1	0	0	*1	*1	0	0	0	1
0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	1	0	0	1	1	0	1	0	1	0	0	1
0	1	1	1	0	1	1	1	*0	*0	1	1	0	1
1	0	0	0	0	0	1	0	*1	*1	0	0	1	0
1	0	0	1	0	1	1	0	1	0	0	1	1	0
1	0	1	0	1	0	1	0	0	1	1	0	1	0
1	0	1	1	1	1	1	0	*0	*0	1	1	1	0
1	1	0	0	0	0	0	0	1	1	0	0	1	1
1	1	0	1	0	*1	0	*0	1	1	0	1	1	1
1	1	1	0	*0	0	*1	0	1	1	1	0	1	1
1	1	1	1	*0	*1	*1	*0	*1	*1	1	1	1	1

表 3　弱い 0 もしくは弱い 1 が入力された場合

				land		lor		leor		lsel2(S=0)		lsel2(S=1)	
A	XA	B	XB	AND	NAND	OR	NOR	EOR	NEOR	S1	XS1	S1	XS1
*0	*0	0	1	0	1	*1	*1	*0	*0	0	1	*0	*0
*0	*0	1	0	*1	*1	1	0	*0	*0	1	0	*0	*0
0	1	*0	*0	0	1	*0	*0	*1	*1	*0	*0	0	1
1	0	0	0	*0	*0	1	0	*1	*1	*0	*0	1	0
*1	*1	0	1	0	1	0	*0	*1	*1	0	1	*1	*1
*1	*1	1	0	0	*0	1	0	*1	*1	1	0	*1	*1
0	1	*1	*1	0	1	*1	*1	*0	*0	*1	*1	0	1
1	0	*1	*1	*1	*1	1	0	*0	*0	*1	*1	1	0

次に leor について説明する．このセルは入力信号パターンが正しい場合は正しい出力信号を示し，間違った入力信号パターンの場合は EOR と NEOR で不正値を出力する．

最後に lsel2 について説明する．この回路では，間違った入力信号パターンに対する出力結果が 2 種類存在する．1 つめは不正値を出力する場合である．この場合は，land，lor と同様に入力信号が間違っているということが分かる．2 つめは正論理と負論理の出力信号が異なる場合である．表 2 の A，XA，B，XB，S，XS=[0，0，0，1，0，1] の場合の出力 S1=0，XS1=1 がそれに該当する．しかし，lsel2 は S=0，XS=1 の場合は B と XB を出力するよう設計されているので，A と XA が同じ値であっても出力には反映されない．よって，この出力は正しい出力といえる．

4.3　弱い 0 もしくは弱い 1 が入力された場合

本節では，不正値として弱い 0 あるいは弱い 1 が入力された場合でも強い 0 あるいは強い 1 の不正値が入力された場合と同じ結果を出力することを示す．

表 3 に結果を示す．land，lor，leor，lsel2 それぞれの出力結果は，入力信号が正しい値の場合の出力結果と基本的に等しく，一部は出力が強い値から弱い値に変化している．これは，故障が発生したことを示す弱い値が入力信号として与えられたとしても，出力信号として不正値が出力されることで故障が次の段へ伝搬されるということを示している．

5.　演算器の構成方法

4 節の結果を踏まえて，次のような回路構成を持つ演算器を考える．

- 回路はセルの 2 重化構成とする
- ある適切な論理回路ごとにセレクタを挿入する
- 最終段の後ろに leor を設置し，出力結果を 2 ビットに集約させる

回路 2 重化について 3.1 節で説明したので省略する．挿入されるセレクタは 2 重化した回路の出力信号がそれぞれ入力信号として与えられている．2 重化した回路の片方の出力が不正値であっても，もう片方が正しい入力の場合は正しい値のペアを出力する．以下，このセレクタを lsel2 detect と呼ぶ．lsel2 detect の構成を図 9 に示し．不正値が入力された場合の出力結果を表 4 に示す．

lsel2 detect を用いた回路の修復を図 10 を用いて説明する．図中の論理回路 A1 と論理回路 A2 は同じ回路であり，B1 と B2 も同じ回路である．今 A1 で故障が発生し，不正値が出力されたとする．A2 は正しい値を出力している．これは表 4 の A，XA，B，XB=[0,

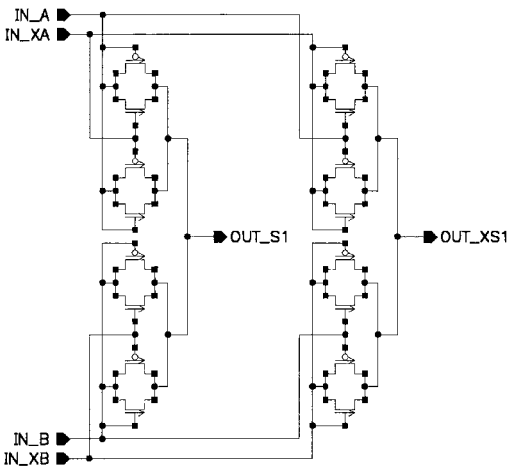

図 9 lsel2 detect

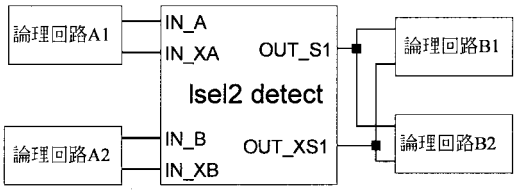

図 10 lsel2 detect による回路復旧

0, 0, 1], [0, 0, 1, 0], [1, 1, 0, 1], [1, 1, 1, 0] のどれかに該当し，いずれの場合も B および XB の値が出力される．同様に A2 が故障した場合でも，A および XA の値が出力される．また，lsel2 detect を用いた修復回路は従来の多数決回路より少ないトランジスタ数で作成することができ，回路の面積が小さくなる．

しかし，A1 と A2 両方で故障が発生した場合，不正値が出力され伝搬するので演算器の最終段の出力は不正値となる．最終段の出力信号が不正値かどうかを調べるために，再収斂セルと同様にツリー状の leor を使用する．

この演算器の問題点として，1 つの論理回路の複数段で故障が発生した場合，故障を検出できない可能性があげられる．任意の段のセルで故障が発生すると，不正値が後続の論理回路へ伝搬するが，途中で別の故障したセルを通過する際，正しい正論理と負論理の値へ戻る場合と，正負が逆転する場合が考えられる．正しい値へ戻った場合は問題ないが，逆転した場合は検出できない．

表 4 不正値が入力された場合 (lsel2 detect)

A	XA	B	XB	S1	XS1
0	0	0	0	*0	*0
0	0	0	1	0	*1
0	0	1	0	*1	0
0	0	1	1	*1	*1
0	1	0	0	0	*1
0	1	0	1	0	1
0	1	1	0	*0	*0
0	1	1	1	*0	1
1	0	0	0	*1	0
1	0	0	1	*0	*0
1	0	1	0	1	0
1	0	1	1	1	*0
1	1	0	0	*1	*1
1	1	0	1	*0	1
1	1	1	0	1	*0
1	1	1	1	*1	*1

6. ま と め

近年プロセス微細化によるトランジスタの性能ばらつきが問題となっている．本稿では，このばらつきを抑える手法として PMOS と NMOS を規則的に配置したセルを提案した．このセルは特徴として，壊れにくい，壊れた場合その故障が検出できる，故障検出機能を演算器内に埋め込むことができる，という点があげられる．このセルを組み合わせることで回路自身に自己修復機能を備わった演算器を構成することが可能となる．また，命令分解機構を使用することにより，ばらつきの影響を演算器のみに絞ることができる，以上から，少品種セルで構成された演算器と細粒度命令分解を組み合わせることで，従来にない高い信頼性を持つアーキテクチャが実現可能であると考えられる．

現在は上記のセルについて HSPICE を使ってシミュレーションを行い評価を行っている段階である．今後は，回路のレイアウトを行い，回路面積の具体的な評価や，VDEC を通じて上記の演算器で構成されたチップの試作を行う予定である．

謝辞 本研究の一部は科学研究費補助金（基盤研究 (B) 課題番号 19300012）による．また，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，シノプシス株式会社，日本ケイデンス株式会社，ローム (株) および凸版印刷 (株) の協力で行われたものである．

参 考 文 献

1) J Von Newmann: *"Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components"*, pp.43–98, Princeton Univ. Press (1956).
2) 島田貴史，田畑猛一，北村俊明：“異種命令セット同時実行プロセッサとしての OROCHI の構成”，情処研報，2006-ARC-170，pp.55–60 (2006).