

動的再構成のためのセル割り当て回路の設計

小関 豊 金杉昭徳

東京電機大学工学部 〒101-8457 東京都千代田区神田錦町 2-2

E-mail: kanasugi@d.dendai.ac.jp

あらまし 本論文は動的再構成のためのセル割り当てアルゴリズム, VHDL による設計, シミュレーションによる動作確認について述べる. また回路規模, 動作速度について考察する. 対象とするシステムは同一のセルの繰り返しから構成されるアレイシステムである. 88 個のセルを再構成する場合, 回路規模は約 33,000 ゲート, 1 回の再構成に要した時間は約 $12.9\mu\text{s}$ であった.

キーワード 動的再構成, フォールトトレランス, アレイプロセッサ

Design of Cell Assignment Circuit for Dynamic Reconstruction

Yutaka Koseki, Akinori Kanasugi

Department of Electronic Engineering, Tokyo Denki University
2-2 Kanda-Nishiki-cho, Chiyoda-ku, Tokyo, 101-8457 Japan

E-mail: kanasugi@d.dendai.ac.jp

Abstract This paper describes the cell assignment algorithm for dynamic reconstruction, the design by VHDL, and the verification by simulation. Moreover, the circuit scale and the operation of speed are investigated. The target is an array system which consists of repetitions of the same cell. In the example which reconstructs 88 cell, the circuit scales is about 33,000 gates and the reconstruction time is about $12.9\mu\text{s}$.

Keyword Dynamic Reconstruction, Fault Tolerance, Array Processor

1. はじめに

現在, VLSI は日常生活や社会活動などに大きく関わっている. このような VLSI の使用に大きく依存する社会では, VLSI に障害が起こった場合, 大きな損害を被る事になる[1]. このため, 障害が発生する確率ができるだけ小さい VLSI, すなわち信頼性の高い VLSI を作る事が重要である.

本研究では, 故障が検出された場合でも故障箇所を回避してシステムを再構成させることで, VLSI にフォールトトレランスを持たせることを目的とする. 再構成を行なうシステムとしては, 同一のセルの繰り返しから構成されているアレイ状のシステムで, 高速フーリエ変換(FFT)システムなどを対象として想定している.

フォールトトレランスを構築する基本的手法は冗長化である. 再構成による故障回避法では, 物理的に冗長なセルを用意し, 故障セルをそれぞれ置き換える. これまでメッシュ結合アレイ, ハイパキューブ結合アレイなどの様々な再構成可能アーキテクチャとそれに対する再構成法が提案されている[2][3]. 本研究ではより柔軟な再構成を効率よく実現することを目指している.

第 2 節では提案する再構成アルゴリズムについて,

第 3 節では再構成回路の設計手順を説明する. 第 4 節では設計した回路のシミュレーションを行い, 回路規模, 動作速度について考察する.

2. 再構成アルゴリズム

アルゴリズムの目的は同一のセルの繰り返しから構成されているアレイ状のシステム(高速フーリエ変換(FFT)システムなど)を, 不良セルを避けて良品セルを配置し再構成することにある. 再構成手順の概要は, あらゆる故障分布に適合するための共通の配置パターンを用意し, これを故障分布の故障箇所に適合するように修正して当てはめていく. この方法の特長は, 使用するセル位置を決定する代わりに, 冗長なセル位置を決定してから, 共通配置パターンと同じ順番でセルを当てはめる点にある. 一般に冗長なセル数は, 使用するセル数に比べて少ないので, 短い時間で決定できる.

共通配置パターンはすべてのセルが良品と仮定したときに最も望ましい配置として求める. 最も望ましい配置とは, 各セル間の距離が短く, セル, 配線が密集していない配置のことを言う. 図 1 は 8 個のセルから構成されている回路に冗長性を導入し, 共通配置パターンとした場合の例である.

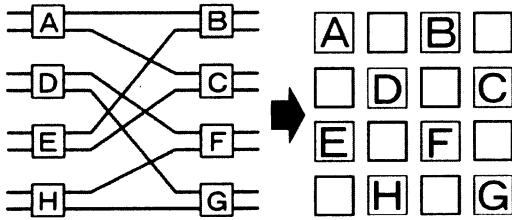


図1 共通配置パターンの例

図1の共通配置パターンではAからHまでのアルファベットが書かれているセルが回路の動作に使用するセルを表し、なにも描かれていないセルが予備の冗長なセルを表している。

故障分布は、正常に動作しない不良セルの分布である。故障分布の例を図2に示す。

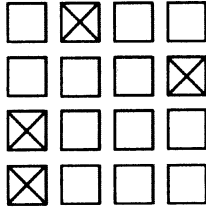


図2 故障分布の例

故障分布の例では、斜線が描かれているセルが故障セルを表し、なにも描かれていないセルが正常に動作する良品セルを表している。

アルゴリズムの計算量を減らすために、共通配置パターンおよび故障分布を1次元に変換する。図3のようにジグザグの矢印に沿って2次元マップを1次元マップに変換する。

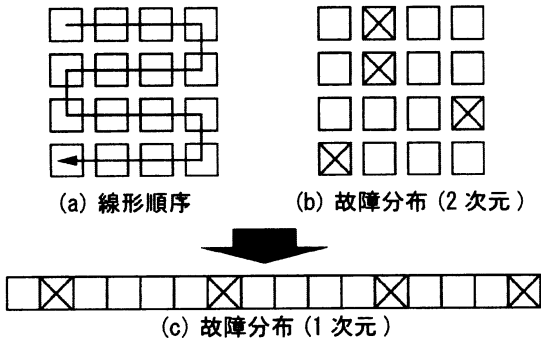


図3 一次変換の例

共通配置パターンを当てはめるための手順を以下にまとめる。

[Step 1]

良品セル数 P 、使用するセル数 N 、冗長セル数 R とする。冗長セル数 R は良品であるが使用しないセルのこと、つまり

$$R = P - N$$

となる

- $R < 0$ なら、システム構成不可になり終了
- $R = 0$ なら、すぐに[Step 3]へ
- $R > 0$ なら、[Step 2]へ

[Step 2]

冗長セルを決定する。

Scp : 共通配置上で i 番目までの冗長セルの総和

S : 故障分布上で i 番目までの故障セルの総和

としたとき、両者の差が最小になるように冗長セルを配置する。そのためのアルゴリズムを下に示す。

$$S = 0$$

$$Scp = 0$$

for ($i = 0 \sim M$) { * M は扱うビット長

if (共通配置上の i 番目が冗長セル) {

$$Scp = Scp + 1$$

if (故障分布の i 番目が良品セル)

if ($S < Scp$ and $R > 0$) {

i 番目を冗長セルに指定

$$S = S + 1$$

$$R = R - 1$$

} else { *故障分布の i 番目が不良セル

$$S = S + 1$$

if ($S > Scp$ and 既に冗長指定している) {

・直前の冗長指定を解除

$$S = S - 1$$

$$R = R + 1$$

}

}

[Step 3]

空きセルに共通配置パターンと同じ順序で使用するセルを配置していく。

以上の3つのステップからセルの再構成を行なう。再構成アルゴリズムを用いて、前に示した図1の共通配置パターンと図2の故障分布を再構成した場合の例を図4に示す。

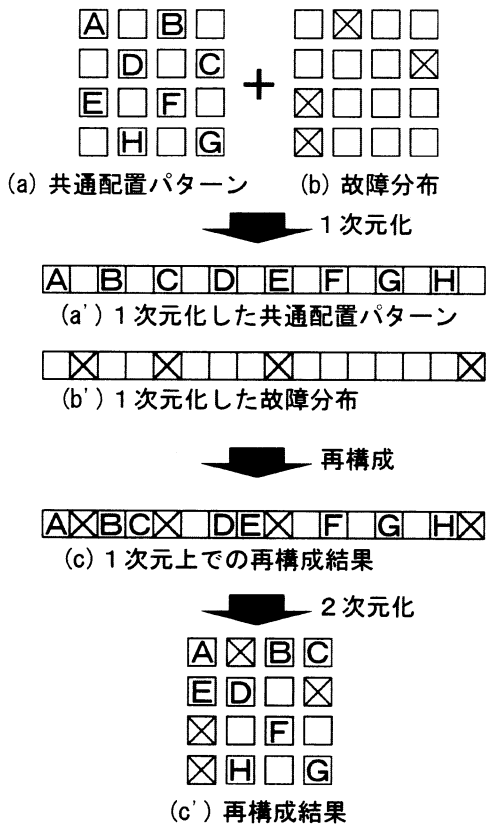


図4 再構成の例

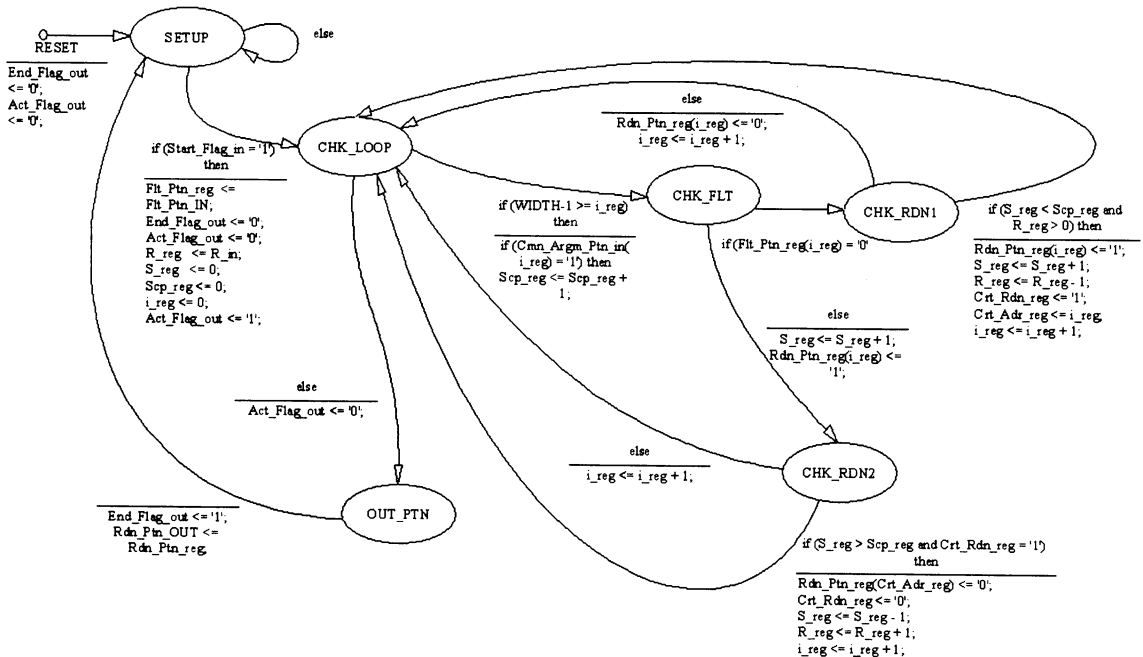


図5 状態遷移図

3. 回路の設計

前節で説明した再構成アルゴリズムのハードウェア化を行った。はじめに回路動作を表す状態遷移図を作成し、次に状態遷移図を参考にしてVHDLを記述した。最後に、記述したVHDLから論理合成ツールを使いPCでゲート・レベルのネットリストを生成し、回路を自動合成した。

作成した再構成回路の状態遷移図を図5に示す。この状態遷移図は6つの状態を取る。初めにリセットされることで状態"SETUP"となり、開始信号が入力されるのを待つ。開始信号が入力されると、状態"CHK_LOOP"へ移動し再構成が完了したかをチェックする。再構成の途中の場合は、状態"CHK_FLT","CHK_RDN1","CHK_RDN2"を移動し再構成を行っていく。最後のセルまで再構成を行った場合は、状態"OUT_PTN"へ移動し再構成結果と終了信号を出力して、再び状態"SETUP"へ戻る。

図6に設計した回路のブロックダイアグラムを示す。入力には故障分布、クロック、リセット、再構成の開始信号の4つである。出力は再構成結果と再構成の終了信号の2つである。共通配置パターンは頻繁に変化するものではないので回路内にデータを内蔵することにした。回路の動作は、初めに入力された故障分布から使用できるセルの数を求め、それからアルゴリズムにしたがって再構成可能かを判断し、再構成を行なう。

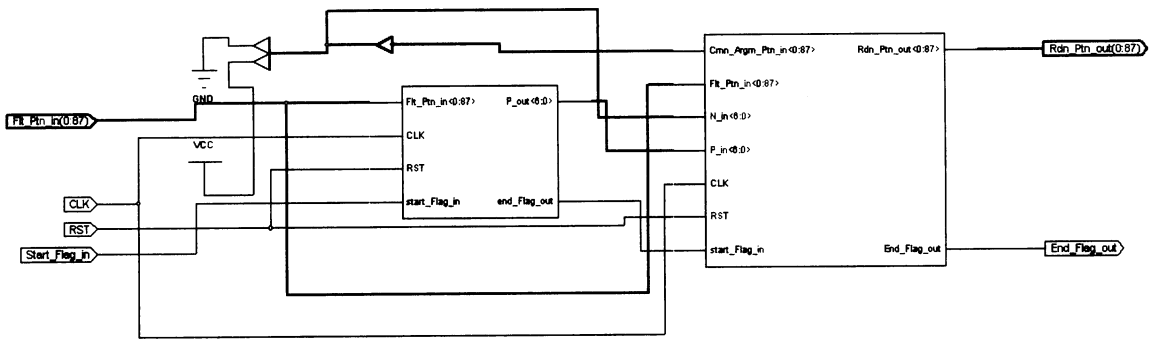


図 6 ブロックダイアグラム

4. シミュレーション結果

設計した回路のシミュレーションを行った。論理合成には Xilinx 社の ISE WebPACK 6.2.03i を使用し、ターゲットデバイスは Xilinx 社の Spartan-III XC3S50 を想定した。論理シミュレーションには Model Technology 社の ModelSim XE II 5.7g を使用した。

シミュレーションの対象とするシステムは FFT システムでブロックダイアグラムを図 7 に示す。FFT システムを共通配置パターンに変換したものを図 8 に示す。この例では、88 個のセルの中から 48 個のセルを使用する。図 9 に線形順序を図 10 に故障分布を示す。この場合では故障セルは 20 個としてある。図 11 にアルゴリズムを適用した場合の再構成結果を示す。

シミュレーション結果から、設計した回路は所望の動作を行なうことが確認することができた。図 12 にシミュレーション結果の例を示す。再構成結果は '1' が冗長セル、故障セルによる使用しないセルを表し、'0' が使用するセルを表している。シミュレーション上での最高動作速度は約 55MHz で、再構成の開始信号が入力されてから、再構成が行われ終了信号が出力されるまでに約 12.9 μ s の時間がかかった。

回路規模は約 33,000 ゲートであった。現在の VLSI は数十万ゲート超であることを考慮すると、十分に実用的な規模であると考えられる。また、再構成は回路の電源が ON になった時や、ある適度の時間を置いて定期的に行われることを想定すれば、12.9 μ s という速度は十分に実用的な速度であると考えられる。

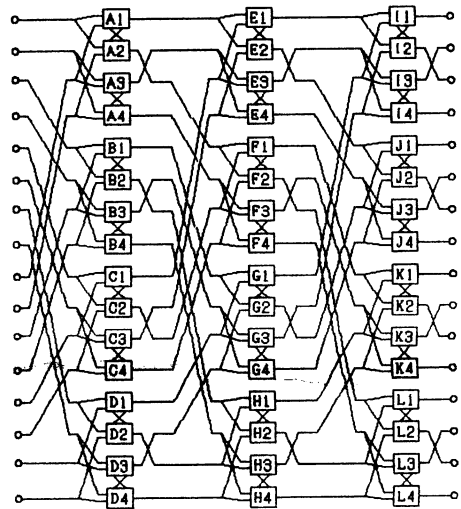


図 7 FFT システムのブロックダイアグラム

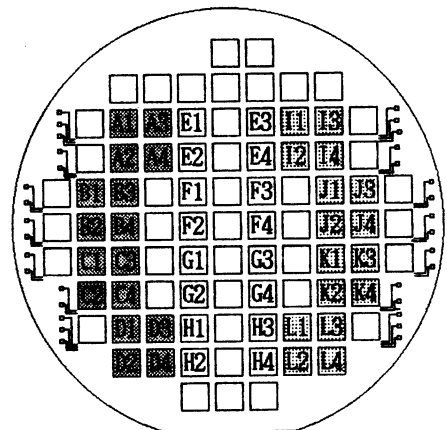


図 8 共通配置パターン

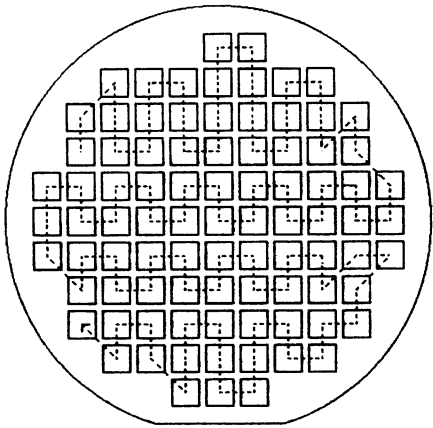


図9 線形順序

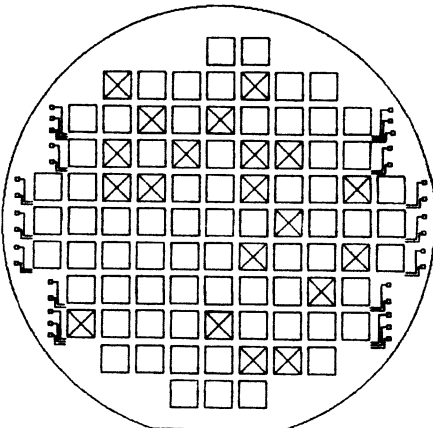


図10 故障分布

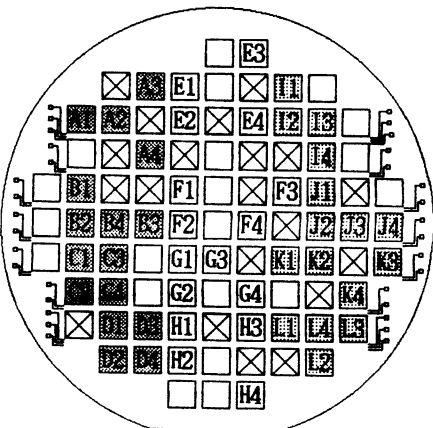


図11 配置結果

5. まとめ

再構成アルゴリズムのハードウェア化を行った。論理合成、論理シミュレーションの結果から、所望の動作を確認することができた。また、回路規模は約33,000ゲート、1回の再構成にかかった時間は約12.9 μ sであった。

現在のVLSIの規模を考慮すると、十分に実用的な規模であると考えられる。また再構成は頻繁に行われるものでないことから速度も十分に実用的と考えられる。

参考文献

- [1] 当麻喜弘, 南谷崇, 藤原秀雄, “フォールトトレラントシステムの構成と設計”, 槇書店, 1991.
- [2] 高浪五男, 松野浩嗣, “WSIにおけるフォールトトレランス”, 信学誌, vol.81, no.9, pp.888-892, 1998.
- [3] 山下公一, 馬場文雄, “ウェーハスケールFFTプロセッサおよびウェーハメモリ”, 信学技報, WSI91-1, 1991.
- [4] A.Kanasugi, Y.Koseki, “A Control Circuit for Reconstruction of Wafer Scale Integrated Circuit”, Proc. of 2004 ICEP, pp.47-52, 2004.

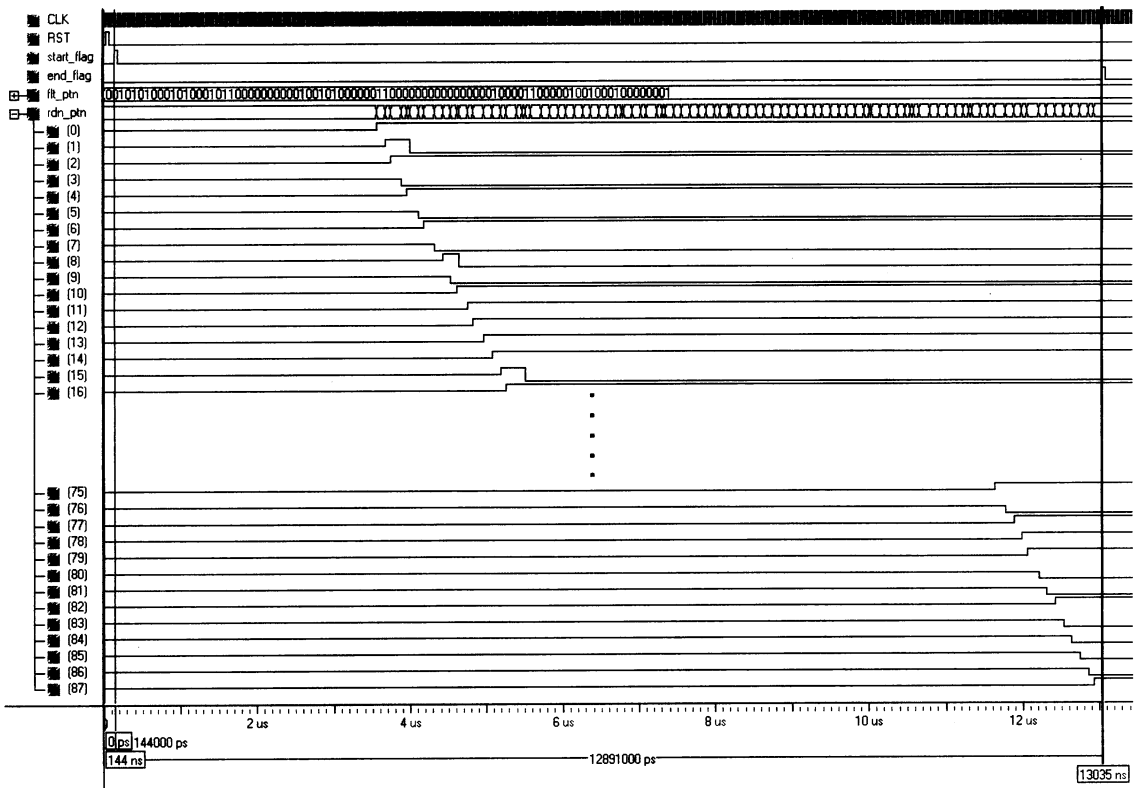


図 12 シミュレーション結果の例