再構成可能デバイスを用いた高信頼システムの設計

Design of Reliable Systems with Reconfigurable Devices

市原 英行（Hideyuki ICHIHARA, Ph.D.）
広島市立大学大学院 情報科学研究科 准教授
(Associate Professor, School of Information Sciences, Hiroshima City University)
IEEE, 電子情報通信学会
受賞：電子情報通信学会論文賞(2005) WRTLT2004 Best Paper Award (2005)
研究専門分野：計算機システム ディペンダブルコンピューティング
ストカスティックコンピューティング

あらまし 本研究では、再構成が可能な LSI の 1 つである FPGA（Field Programmable Gate Array）を利用して、低コストで高信頼なシステムを実現する方法を提案する。提案システムは、自律的に故障箇所を検出し、故障箇所を自ら修復または切り捨てることで、サービスを少しでも長く継続する。また、提案システムは複数の故障への対応や故障の種類に応じた回復処理を行う適応性も有しているため、限られたリソースを適切に利用することが可能である。これらの機能により提案システムは、人の手が届きにくく、稼働期間中の部品交換などが難しい遠隔地や宇宙空間での常駐システム（例えば軌道計算や通信制御システム、監視システムなど）に利用できると考えられる。さらに、提案システムを Dual-FPGA アーキテクチャを用いて実装することで、具体的なアプリケーションを対象にその機能や動作を検証し、その実現可能性を明らかにする。

１．はじめに

半導体技術の急速な進歩は集積回路（LSI）の高機能化、高速化、多様化などを可能とし、その結果、多くの LSI システムが我々の生活に深く関与している。当然ながらこれらの LSI システムは信頼できること（故障しないこと）が求められるが、その目的によって求められる信頼性は様々である。医療システムや金融システムのように、システムの障害が人命や経済活動に大きなインパクトを与えるため、高い信頼性を求められるものから、ゲームなどのアミューズメントシステムやコンシューマー向けの製品などのように、多少の障害であれば許容できるものまで様々存在する。当然ながら、前者のような高信頼システムは 2 重化や 3 重化などを行うことで高信頼を実現する場合が多いため実現コストは高くなり、後者の比較的低信頼なシステムは低コストでの実現が可能である。

本研究では、再構成可能な LSI の 1 つである FPGA（Field Programmable Gate Array）を利用して、低コストで高信頼なシステムを実現する方法の提案を行っている[1]-[4]。提案しているシステムは、無修理でシステムのアベイラビリティ（可用性）を高くすることできるため、人の手が届きにくく、稼働期間中の部品交換などが難しい遠隔地や宇宙空間での監視システムや常駐システム（例えば軌道計算や通信制御など）に利用できると考えられる。

以下では、これまでの研究動向を述べた後、提案システムの概要、そして Dual-FPGA アーキテクチャを用いた提案システムの実装と検証について述べる。

２．FPGA を用いた耐故障システム

FPGA とは SRAM 型の再構成可能デバイスであり、図 1 のように論理ブロック（Configurable Logic Block, CLB）と呼ばれるブロックが格子状に並び、それらが相互にスイッチマトリックスで接続された構造となっている。CLB の主な構成要素は SRAM で構成されたルックアップテーブル（LUT）とフリップフロップ（FF）である。LUT は SRAM に格納された構成情報によって真理値表を表現し、入力をアドレスとしてその真理値表を参照することで出力を決定する。

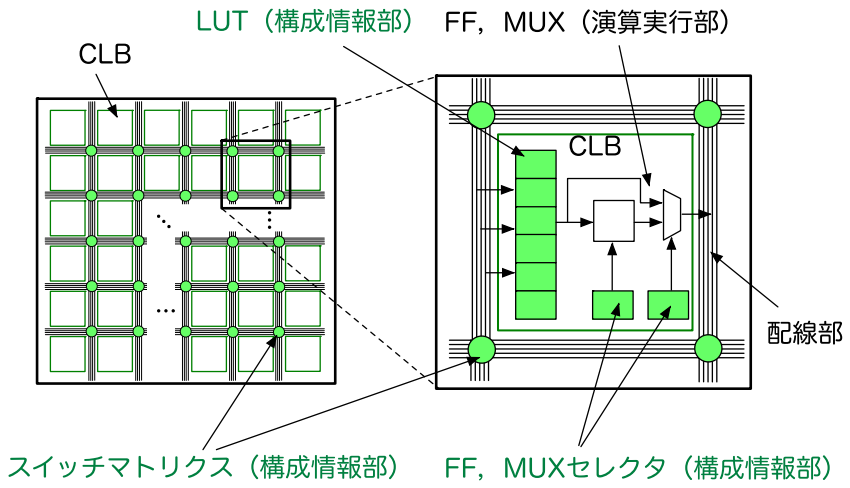

図１　ＦＰＧＡの構造

このLUTとFF, SRAMに格納された構成情報により、配線の接続を制御できるスイッチマトリックスを組み合わせることにより、目的に応じたアプリケーションを実現することが可能となる。

LUT やスイッチマトリックスなどの構成情報によりアプリケーションの構成を決定する部分を構成情報部、FF やマルチプレクサなどの構成情報部以外のアプリケーションを実現する部分を演算実行部と呼ぶ。FPGA 上にアプリケーションを実現する際には、FPGA 外部から回路構成情報（コンフィギュレーションデータ）を構成情報部であるSRAMにダウンロードして構成情報を更新する（再構成）。再構成時に、構成情報部内の回路構成情報は新しいものに更新され、演算実行部の FF の内容はリセットされる。これにより、コンフィギュレーションデータに応じた構成のアプリケーションが 、FPGA 上に実現される。

このような FPGA の再構成機能を利用した様々な耐故障システムが、いくつか提案されている。文献[1][5][6]では、予め FPGA 内部に予備の領域を設けておき、故障時にはその領域を利用するシステムが提案されている。このような耐故障システムは待機予備システムと呼ばれる。例えば、文献[5]では、構成情報が列方向で１つのグループを作っていることに着目し、このグループを１つの単位として FPGA を分割している。初期の構成情報では、予め予備グループを用意しておく。運用時には、グループごとに誤り検出を行い、誤りを検出すると、予備グループに故障グループの回路構成情報をシフトさせることで障害から回復する。

一方、文献[1][7][8]では、予備の領域をあらかじめ設けるのではなく、故障を検出すると、故障箇所を避けるように再構成を行う方式が提案されている。このような耐故障設計を漸次縮退システムと呼ぶ。一般的に漸次縮退システムは、待機冗長システムに比べて、正常時には FPGA のすべての面積を利用できるために、処理速度や能力が高いという特徴がある。この方式は筆者らが提案するシステムでも採用している。

自律的に FPGA の再構成を行うためには、再構成を制御するための機構がシステム内に必要となる。一般的には、外部にパーソナルコンピュータ（PC）や専用のコントローラを持つことで再構成を行うが、文献[9]では、2 つの FPGA を用いた Dual-FPGA アーキテクチャが提案されている。2 つの FPGA が互いの動作状況を監視し、相手のエラーを検出すると再構成させるためのコントローラを有する自律修復システムである。この方式は筆者らのシステムの実装に利用している。

３．故障状況対応型漸次縮退システム（提案システム）

提案システムは、自律的に障害から回復し動作を継続するための機構を有する漸次縮退システムであり、故障状況に対応してその回復動作を変えるものである。システムの特徴を以下に述べる。

３．１　ブロック単位のエラー検出と漸次縮退 [1]

提案システムの 1 つの特徴は、漸次縮退システムである点である。FPGA 上に故障が発生した際に、その箇所を特定・分離して残りの無故障部分で同じ機能を持つ回路を構成し、システムを継続して動作する。漸次縮退のために、図 2 のように FPGA の領域を複数の CLB からなる部分領域（ブロック）に分割する。また、各ブロックはエラー検出機構（二重化、符号化など）を備えており、どのブロックが故障したかを判断することができる。ブロック数を N とすると（図では N=9)、システム運用開始時には N ブロックを使用した回路構成で実行する。

ブロックの故障を検出すると漸次縮退を行う、つまり、図 3 のように、その故障ブロックを切り離し N–1 個のブロックでシステムを再構成する（回路構成 2)。これは故障ブロックを検出する度に行われ、システムで使用するブロックは再構成の度に減少していく。そして、サービスが継続できないところまで縮退すると、システムは停止する。

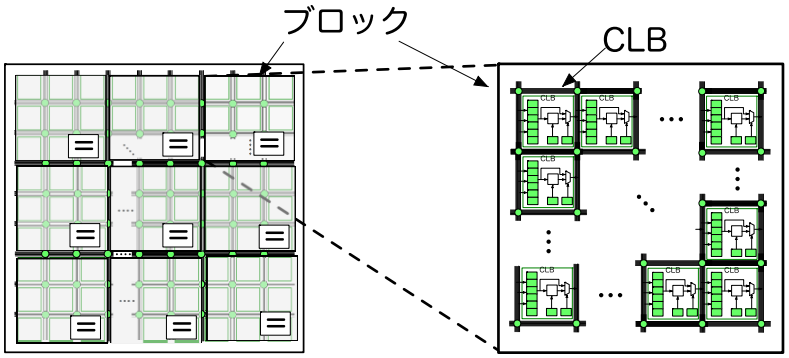

図２　ブロックに分割されたＦＰＧＡ

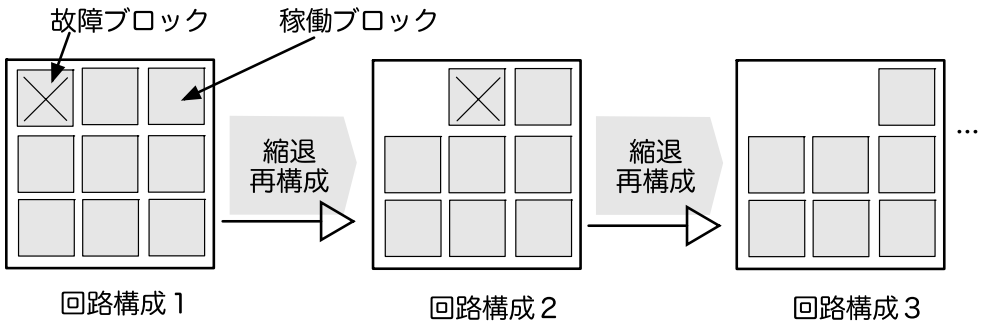

図３　漸次縮退再構成の様子

このとき、図 4 に示すように利用可能なハードウェアリソースの減少にしたがって、演算ユニットを別の時間に共有して利用することで同等の機能を実現する。この例では、回路構成 1 では 9 つのブロックを使えるため、9 つの演算ユニットを 9 つのブロックで実現しているが、回路構成 2 では 8 つのブロックしか使えないため、1 つの加算ユニットを 2 つの加算で共有している様子（色をつけた部分）を示している。回路構成 3 ではさらに共有が進み、1 つの加算ユニットを 3 つの加算で共有している。このような演算ユニットの共有は、性能低下が起こる可能性がある。この例では、回路構成 1 に対して回路構成 2 は 1 時刻、回路構成 3 は 2 時刻、演算時間が延びている様子を示している。

このような回路構成情報は、故障ブロックに応じてあらかじめ合成し、複数の回路構成を用意しておくものとする。ただし、この方式はメモリを多く必要とするため、故障状況に応じて適切な回路構成を選択して再構成することも検討している。

3．2　故障状況への対応 [2]

提案システムのもう 1 つの大きな特徴として、エラー発生の原因である故障の種類を推定し、回復のための適切な回路構成を選択できることが挙げられる。これにより、過剰な縮退再構成を避けることができる。本研究では、以下の 3 つの故障を想定している。

■ **演算実行部の一時故障**：放射線衝突などの影響で、演算中のデータが変化してしまう故障である。その誤った演算結果が FF に格納されるとエラーになる。放射線衝突などの影響は一時的なものであるため、エラーも一時的となる。このようなエラーをソフトエラーと呼ぶ。図 5 では、フリップフロップの値が放射線衝突によって 0 から 1 に変化した例である。

■ **構成情報部の一時故障**：LUT やスイッチマトリックスの SRAM の値が放射線衝突などにより変化することで回路構成情報が破損し、回路構成が変化してしまう故障である。正常なシステムでは構成情報は演算中には更新されないため、この故障状況が原因のエラーは、ソフトエラーと異なり再現性がある。そのため、再計算を行っても回復はできない。このようなエラーをファームエラーと呼ぶ。図 5 では、回路構成情報が変化することによって AND ゲートを構成していた LUT が XOR ゲートに変化した例である。

■ **永久故障**：縮退故障などの物理的欠陥による FPGA そのものの故障である。永久故障が原因のエラーは、ファームエラーと同様にエラーに再現性があり、演算実行部に発生した場合と構成情報部に発生した場合を区別しない。このようなエラーをハードエラーと呼ぶ。図 5 では、信号線が断線し信号が伝播されない例である。

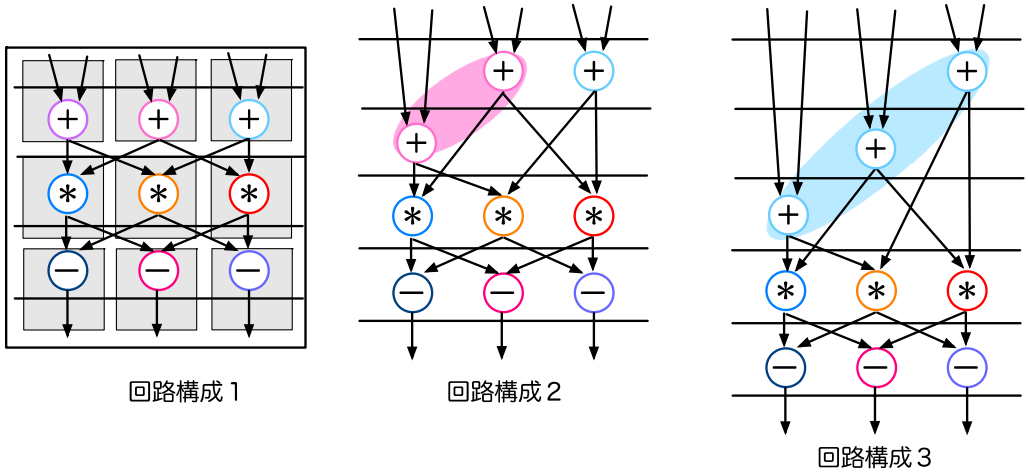

図 4　稼働ブロック数と性能の関係

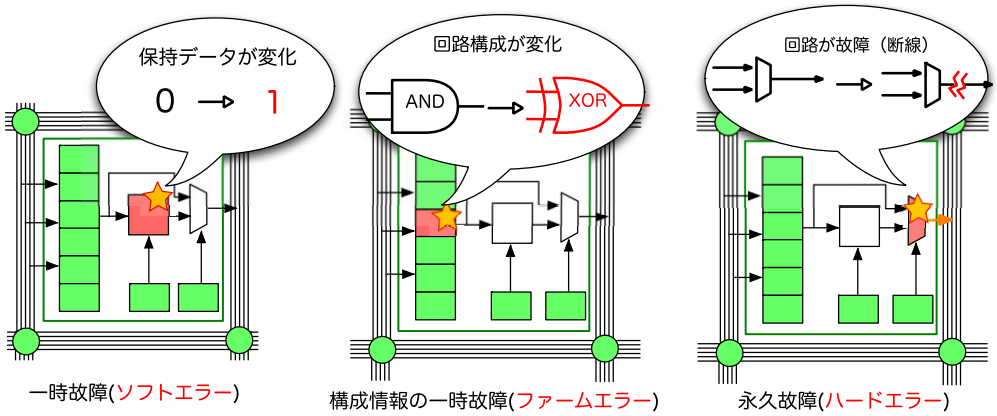

図５　ＦＰＧＡで起こる故障の分類

ソフトエラーの場合は、回路構成情報の変更の必要はなく、エラーを訂正するのみでよいため、演算情報のロールバックと再計算によって復帰する。一方、ファームエラーの場合、回路構成情報は正常と異なるが、FPGA そのものの構成機構（ハードウェア）は正常であるため、そのエラー発生箇所を分離する必要はなく、同一のコンフィギュレーションデータを用いて再構成することで縮退せずに、正常状態に回復することが可能である。ハードエラーの場合は、FPGA ハードウェアの故障のため、その発生ブロックを分離する縮退再構成によって、故障状態から回復する。

このように、3 つのエラーを適切に見分けることができれば、過剰な縮退を避けることができ、システム運用開始時の性能を維持しながらシステムの寿命を延ばすことができる。しかし一般に、検出されたエラーからは 3 つのうちのどのエラーかを判別することはできない。したがって、このシステムでは 3 つのエラーレベルを設定し、各レベルで異なるエラーへの回復処理を試行し、エラーから復帰できない場合は、次のエラーレベルに進むことを繰り返す。エラー想定と回復処理の状態遷移図を図 6 に示す。各レベルで想定するエラーと対応するエラーの検出時の対応は次のようになる。

■ エラーレベル 0：ソフトエラーを想定し、ロールバック（再計算）で回復（EL0）

■ エラーレベル 1：ファームエラーを想定し、同一再構成後、ロールバックで回復（EL1）

■ エラーレベル 2：ハードエラーを想定し、縮退再構成とロールバックで回復（EL2）

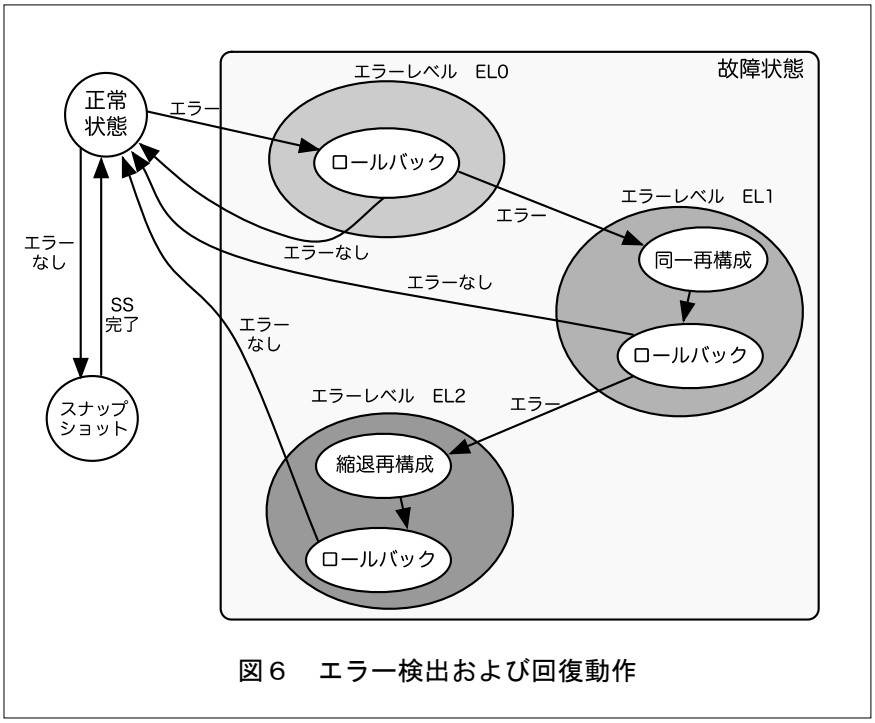

図６　エラー検出および回復動作

ここでは、発生する故障の状況がソフトエラー、ファームエラー、ハードエラーの順で頻度が高いと仮定している。エラーレベルにしたがって、システムはエラーから回復するまで、EL0, EL1, EL2 の順に回復処理を行う。

４．Dual-FPGA アーキテクチャによる実装

提案するシステムの有効性を検証するために、文献[9]で提案されている Dual-FPGA アーキテクチャを参考にして、相互再構成型のシステムとして実装を行った[3][4]。本実装におけるアーキテクチャを図 7 に示す。主な機能は FPGA-main（以下、*Fm* ）で実装され、再構成等を補助する機能は FPGA-support（以下、*Fs* ）で実装されている。さらに、*Fm* は演算部、制御部というモジュールからなり、*Fs* は SS 保持部、監視部というモジュールからなる。いずれのモジュールも、機能単位でさらに分割されており、それぞれでエラー検出機能を備えている。

- **演算部**：アプリケーション実装部であり、定期的に演算のスナップショット（SS）をとり、ロールバック（RB：再計算）等に備える。SS のデータは SS 保持部に転送される。演算部から RB の指示を受けたとき、あるいは、再構成により初期状態で復帰したとき、SS 保持部から SS データを呼び戻し、エラー発生以前の状態への復旧を試みる。
- **制御部**：演算部、SS 保持部、監視部のエラーの監視と、副 FPGA *Fs* の再構成を制御する。演算部のソフトエラーを推定したとき（EL0 の状態でエラーを検出したとき）は、演算部および SS 保持部に RB を指示する。演算部のファームエラー、ハードエラーを推定したとき（EL1, EL2 の状態でエラーを検出したとき）、*Fs* の監視部に *Fm* の再構成を依頼する。
- **SS 保持部**：制御部の指示に従いながら演算部から SS データを受け取り、それを保持する。主 FPGA *Fm* の再構成時などに、保持している SS データを演算部に転送する。
- **監視部**：制御部のエラーを監視する。制御部のエラーおよび制御部からの再構成依頼の受信時に、主 FPGA *Fm* を再構成し、制御部へ *Fm* を復旧させるための指示を送る。

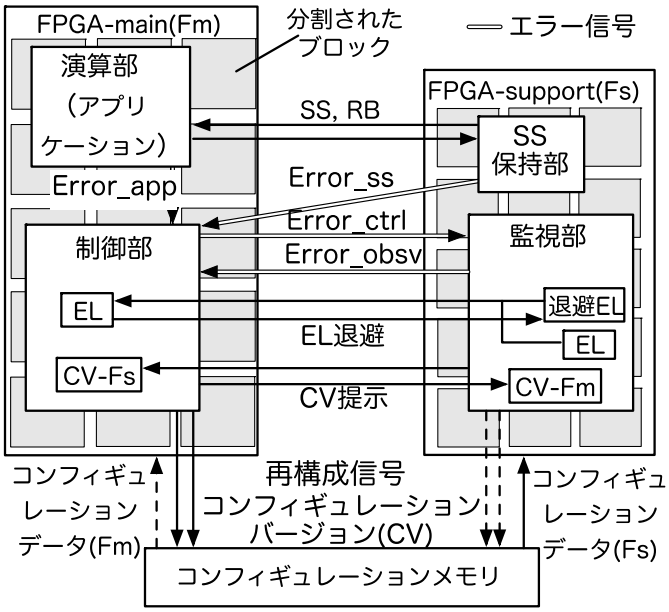

図７　Dual-FPGA アーキテクチャによるシステムの実装

このように、演算部を主とし、演算部とその動作に密接に関わる制御部を *Fm* に、演算の SS を保持する SS 保持部と再構成を行う監視部を *Fs* で実現している。これによって演算部が利用できるリソースを最大限確保し、再構成による回復時の制御を簡略化している。それぞれのモジュールは、その機能を二重化による比較や符号化を施すことでエラーを検出する。エラーの検出は、演算部と SS 保持部では、演算結果や保持している SS データの比較によって行い、制御部と監視部では、状態や出力信号を比較することで行う。比較結果が一致しない場合は、エラー信号を出力する。

５．実験結果

設計したシステムの動作を検証するために、システムの FPGA 部分を 2 台の FPGA 評価ボード ATLYS Board（Digilent 社、デバイス名：XC6SLX45、パッケージ：CSG324）を利用し、コンフィギュレーションメモリと再構成の実行部分を PC で実装した。クロック発振器は 100MHz のクロック信号を生成する。FPGA-FPGA 間の通信には FPGA が備える VHDCI コネクタと 40 ピンの端子を持つ vMod ケーブル（Digilent 社）を利用し、FPGA-PC 間の通信にはシリアル通信（USB ケーブル、RS232C、115,200bps）を使用し、再構成の実行には DigilentJTAG Config Utility を利用した。このツールによる FPGA の再構成には 8.0 秒必要となる。演算部には、一次元拡散方程式を解く演算回路を実装して実験を行った。

このシステムに対して各モジュールでのエラーを仮定し、回復処理が行われる様子を確かめた。その結果、(1)演算部と SS 保持部の情報が正常にスナップショットやロールバックされていること、(2)故障状況に対応した回復処理が正しく行え、再構成の実行を行う制御部と監視部内の故障状況にも対応できていること、そして、(3)漸次縮退による再構成が正しく行えていることが確認できた。

実装したシステムの面積や動作周波数を表 1 に示す。各モジュールの面積は、それぞれ独立して合成することで求めたスライス数であり、そのため、合計は全体のものと一致しない。動作周波数からは今回想定している 100MHz で十分に動作することがわかる。面積からは、FPGA-support（*Fs* ）は FPGA-main（*Fm* ）の 10%未満で実現できており、また、FPGA-main の制御部も演算部の 20%程度であった。よって、本システムを実現するために必要なオーバヘッドは、それほど大きくないと考えられる。さらに、SS 保持部は演算部の内容に依存する可能性があるが、制御部と監視部はエラーからの回復を命令する FSM であるため、演算部の内容に依存しないことを考慮すると、アプリケーションの規模が大きくなるほど、このオーバヘッドの比率は小さくなると考えられる。

表１　実装結果

	FPGA-main	*Fm*:演算部	*Fm*:制御部
面　積	1,545	1,463	36
動作周波数 [MHz]	137.053	－	－
	FPGA-support	*Fs*:SS 保持部	*Fs*:監視部
面　積	134	71	13
動作周波数 [MHz]	220.473	－	－

６．まとめ

本研究では、再構成が可能な LSI の 1 つである FPGA を利用して、低コストで比較的高信頼なシステムを実現する方法の提案を行った。また、実際に Dual-FPGA アーキテクチャによって動作する相互再構成可能なシステムを実装し、エラー状況に応じた回復処理が可能であることや、再構成の実行を行う制御部と監視部内の故障状況にも対応できること、再構成後に各モジュールが適切な状態に遷移できていることを確認した。実装によって示された提案システムの面積や動作周波数は、システムの実現可能性が示している。

今後の課題としては、コントローラ（FSM）である制御部や監視部での漸次縮退の実現や、任意のアプリケーションに対応した回路構成情報を自動的に合成するシステムの開発、さらには、無停止性を指向したシステムへの応用などが挙げられる。このような課題をクリアしていくことで、提案するシステムを遠隔地や宇宙空間で利用することが可能になると考えられる。

再構成可能デバイスを用いた高信頼システムの設計

Design of Reliable Systems with Reconfigurable Devices

参考文献

[1] R. Noji, S. Fujie, Y. Yoshikawa, H. Ichihara, T. Inoue, “Reliability and Performance of FPGA-Based Fault Tolerant Systems,” IEEE Proc. WRTLT, pp.245–253, 2009.

[2] R. Noji, S. Fujie, Y. Yoshikawa, H. Ichihara, T. Inoue, “An FPGA-based Fail-soft System with Adaptive Reconfiguration,” IEEE Proc. IOLTS, pp.127–132, 2010.

[3] 竹内宏和, 岩垣剛, 市原英行, 井上智生, ”SRAM 型 FPGA を用いた故障状況対応型システムのリカバリ機構に関する考察,” 信学技報, pp. 1–6, 2012 年 3 月.

[4] 森拓馬, 大元将一, 岩垣剛, 市原英行, 井上智生, ”Dual-FPGA アーキテクチャに基づく相互再構成型耐故障システムの実装,” 信学技報 (DC2013-90), Vol. 113, No. 430, pp. 67–72, 2014 年 2 月.

[5] W. Huang, E. J. McClaskey, “Column-Based Precompiled Configuration Techniques for FPGA Fault Tolerance,” IEEE Proc. FCCM, pp.137–146, 2001.

[6] 岡田, 喜多, 塩谷, 五島, 坂井, “耐永久故障 FPGA アーキテクチャ,” 信学技報, vol.110, no.2, CPSY2010-7, pp.33–37, 2010.

[7] 阿部, 青木, 樋口, 鹿股, “故障の検出および回復が可能な FPGA アーキテクチャ,” 電子情報通信学会技術研究報告. FTS, pp.1–8, 2000.

[8] Y. Nakamura, K. Hiraki, “ Highly Fault-Tolerant FPGA Processor by Degrading Strategy, ” IEEE Proc. IEEE PRDC, pp. 75–78, 2002.

[9] S. Mitra, W. Huang, N. Saxena, S. Yu, E. J. McCluskey, “Reconfigurable Architecture for Autonomous Self-Repair,” Design and Test of Computers, pp.228–240, 2004.

この研究は、平成２２年度ＳＣＡＴ研究助成の対象として採用され、平成２３～２５年度に実施されたものです。