A Bio-Inspired Fault-tolerant Hardware System Supporting Hierarchical Self-healing

Jiaqing Xu, Yong Dou, Qi Lv

School of Computer, National University of Defense Technology,
Deya Road, 109#, Changsha, China, phone: +86 13786143965, e-mail: xujiaqing@nudt.edu.cn

Introduction

Fault tolerance has been a critical feature for reliable spaceborne electronic systems that run under hostile cosmic environments. Researchers are continuously looking for efficient ways to design more reliable fault-tolerant electronic systems. The basic principle in fault-tolerant system design is redundancy. Traditional fault-tolerant techniques include space redundancy, time redundancy, and information redundancy. However, these schemes lack flexibility compared with the self-healing capability of biological organisms.

In recent years, researchers have shown interest in how biological organisms survive through self-healing, which may open promising avenues for the design of fault-tolerant electronic system. Two main bodies of research within reconfigurable multicellular array architecture support self-healing: embryonics and eDNA. Embryonics was proposed by A. Stauffer and D.Mange in 1995, and developed by G.Tempesti *et al*. [1, 2]. In 2009, M.R.Reibel and J.Madsen proposed a novel bio-inspired architecture named eDNA [3, 4]. Both embryonics and eDNA imitate the development of the embryo, i.e., undergoing cellular differentiation and self-organization. Each cell of the embryonic array contains the complete DNA of the application; the function of a cell and communication among the cells are determined by the DNA and cell position (cell coordinates or cell numbers) in embryonic development. The relationship between data and cells are tightly-coupled (each data can be processed only by a certain cell). Self-healing in embryonics and eDNA is based on the imitation of the differentiation of embryonic stem cells; that is, they both need to identify a spare embryonic stem cell and recalculate their cell coordinates or cell numbers before restarting self-organization.

In this paper, we propose a novel bio-inspired fault-tolerant hardware system named electronic tissue (eTissue), which supports hierarchical self-healing. In the design of eTissue, we inherit two basic ideas from embryonics: all the cells contain the complete DNA of the application; and cellular differentiation is determined by the DNA and cell number. However, we pay more attention to the biological principles of adult organisms, especially their self-healing principles. In eTissue, we imitate the match-based recognition mechanism in protein sorting to loosely couple data and cells, i.e., each data can be recognized and processed by any homogeneous cell. This loosely-coupled relationship makes the replacement of cells more flexible. Substitution among homogeneous cells, differentiation of adult stem cells, and conversion between heterogeneous cells mechanisms endow eTissue with the capability of hierarchical self-healing.

The rest of this paper is organized as follows. We start with the introduction of biological principles in eTissue, and then depict the architecture, operational mechanism and self-healing schemes of eTissue in detail. Finally, the fault-injection experiments confirm the powerful self-healing capabilities of eTissue.

Biological principles in eTissue

Four important biological principles inspired the development of the eTissue architecture: protein sorting and recognition, substitution among homogeneous cells, differentiation of adult stem cells, and conversion between heterogeneous cells.

1) Protein sorting and recognition: protein sorting is the mechanism by which a cell transports proteins to the appropriate locations inside or outside the cell. For example, protein hormone is initially synthesized by the ribosome, then passed onto the endoplasmic reticulum and Golgi apparatus, where a signal sequence(tag) is added to the nascent protein. This signal sequence functions similar to a postal code, and determines the final destination of the protein, such as nuclear membranes, plasma membranes, or extracellular domains. After further processing, the protein hormone is secreted into the blood and it will be recognized by a particular receptor on target cells [5]. In

the design of eTissue, we imitate this principle by adding each incoming data and intermediate result with a tag so that data can be recognized by any of homogeneous cells.

2) Substitution among homogeneous cells: multicellular organisms contain numerous homogeneous cells. For example, large numbers of erythrocytes exist in our blood vessels; once a erythrocyte dies, its function is replaced by the remaining erythrocytes [5]. Similarly, numerous homogeneous cells are found in eTissue. Data with particular tags can be processed by any corresponding homogeneous cells, rather than a specific cell.

3) Differentiation of adult stem cells: the adult stem cells have weaker differentiation capability compared with embryonic stem cells. Hematopoietic stem cell is a typical kind of adult stem cell, which can convert to erythrocytes, leukocytes, and thrombocytes. When one kind of blood cell decreases, hematopoietic stem cells can differentiate initiatively to make up for the reduction in blood cells. We introduce this self-healing mechanism in eTissue. When a serious shortage of somatic cells occurs, the adult stem cells can convert to the somatic cells required.

4) Conversion between heterogeneous cells: the conversion between heterogeneous cells is amazing remarkable phenomenon. In 2010, researchers from Canada confirmed the possibility of the conversion of human fibroblasts to multilineage blood progenitors, indicating that the conversion between heterogeneous cells is possible through the reprogramming of cell genes [6]. To enhance the self-healing capability of eTissue, we endow eTissue with the capability of conversion between heterogeneous cells by mimicking the same processes in humans.

Architecture and operational mechanism of eTissue

The architecture of the eTissue prototype system is depicted in Fig. 1. It consists of an input controller, cell managers, cells, and data recycle and result collection units. These modules are connected via a Network-on-chip (NoC). The details on the overhead and fault tolerance in communication are purely an NoC-related issue and will not be discussed here.

Input controller. This unit fetches input data from memory and sends the data to different cell managers according to the tag bound with the data.
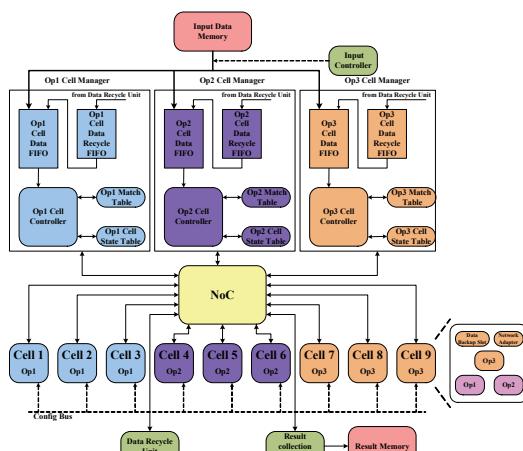
Cell manager. This module is the key component of eTissue, and each type of cell has a cell manager. To create a simple cell structure, the management function of cells is concentrated on this unit. The cell manager is composed of data FIFO, data recycle FIFO, cell controller, match table, and cell state table. It has two main functions: 1) Dispatch data to cells: The cell controller dispatches incoming data to the appropriate cells according to the data tag, the match table, and the cell state table. 2) Complete the conversion between heterogeneous cells: Once the cell manager detects the lack of owned cells by monitoring data recycle FIFO and cell state table, the cell manager sends cell conversion signals to other cell managers. If other cell managers have idle cells, they transfer the ownership of idle cells to the initiator, then the initiator sets idle cell functions identical to those of its own cells.

Cell. This module is a fine-grained processor that can handle simple arithmetic and logic operations. It consists of the data backup slot, network adapter, and processing units. The data backup slot saves incoming data in case the cell fails; the network adapter takes charge of monitoring the state of cells aside from communication; the processing unit is used to complete data processing. The type of processing unit is decided by the cellular differentiation table. If a cell is configured, we call it a somatic cell; otherwise it is a spare stem cell. When all operands have arrived, the cell initiates data processing.

Data recycle unit. This unit recycles incoming data in two cases: 1) when a cell fails, and 2) if there is no idle cell. Then, the recycled data are sent to corresponding data recycle FIFO in each cell manager according to the data tag, queued for the next dispatching.

Result collection unit. This module collects results and stores them in memory.

We elucidate the operational mechanism of eTissue with a simple example. Suppose that the expression shown in Fig. 2(a) requires calculation. First, we obtain the corresponding data flow graph (DFG) from the expression and then add each incoming data and intermediate result with a tag, as illustrated in Fig. 2(c). The format of labeled data is <*operation_type, tag_number, data_value*> [Fig. 2(b)]. Operation_type indicates the operation type that the data will act in (1h for *op1*, 2h for *op2*, 3h for *op3*, 4h means the data is the result), and tag_number is a number used to distinguish different operands of identical operation type (*op1*, *op2*, or *op3*).

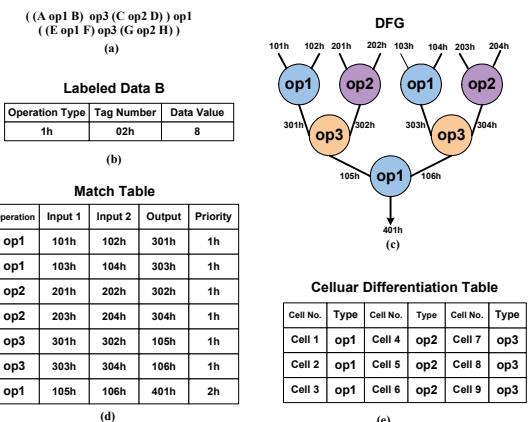**Fig. 1.** Architecture of the eTissue prototype system

Fig. 2. The constitution of eTissue's DNA

Then, we convert the tagged DFG into a match table [Fig. 2(d)]. Note that the priority of the data is used to represent data dependency between homogeneous operations. Fig. 2(c) shows that the operand of the last *op1* indirectly depends on the result of the first and second *op1*; thus, we assign 1 as the priority of the first and the second *op1*, and 2 as that for the last *op1*. The deadlock caused by low priority data occupying the cell can then be avoided through this priority technique. The cellular differentiation (defined by the user) table in Fig. 2(e) decides on the operation type of the processing unit in each cell. The match table [Fig. 2(d)] and cellular differentiation table [Fig. 2(e)] constitute the DNA of eTissue. Before eTissue begins to process input data, a procedure called differentiation is initiated, i.e., each cell sets the function of its processing unit according to the cellular differentiation table. After this, incoming data are dispatched and processed.

First, the input controller fetches input data from memory and sends them to different cell managers according to the tag bound with the data. Once the cell manager receives the data from the input controller, it obtains the tag of matched data from the match table through the tag of the input data, and then looks up the cell state table to determine whether a cell has already been assigned to the matched data. If so, the cell manager sends the input data to that cell and updates the cell state table. If not, the cell manager looks up the cell state table to identify an idle cell, assigns the input data to it, and updates the cell state table. If no idle cells remain, the cell manager looks up cell state table for low-priority cells (the cells occupied by low-priority data). If such a cell exists, the cell manager ejects low-priority data from it, assigns it with input data, and then updates the cell state table. If not, the input data is sent to the data recycle unit and queued for the next dispatching. When all operands have arrived, the cell initiates data processing and then sends the tagged result to the corresponding data FIFO in the cell manager, except for the final result, which should be sent to the result collection unit and stored in memory.

Hierarchical self-healing of eTissue

Self-healing schemes can be divided into two categories: static self-healing and dynamic self-healing. The self-healing scheme is dynamic if the system can accomplish self-healing without interrupting normal operations; otherwise, this is called static self-healing. Both embryonics and eDNA belong to the static self-healing scheme. By contrast, we present a hierarchical dynamic self-healing scheme that includes substitution among homogeneous cells, differentiation of adult stem cells, and conversion between heterogeneous cells. 1) substitution among homogeneous cells: once the network adapter detects a failing cell, it sends the data in the backup slot to the data recycle unit. Next, the data recycle unit shifts the data back to the corresponding cell manager according to the data tag. The cell manager receives this data and then dispatches them to other cells of the same operation type. 2) differentiation of adult stem cells: when a serious shortage of its own cells is detected (there are two

cases that may cause this phenomenon, one is when cells die because of faults, and the other is when cells are initially allocated improperly by the user), the cell manager notifies stem cell managers. If the stem cell managers have undifferentiated stem cells, they transfer the ownership of stem cell to the initiator, and then the initiator sets stem cell functions identical to those of its own cells. 3) conversion between heterogeneous cells: when a serious shortage of somatic cells occurs and undifferentiated stem cells are uesed out, the cell manager notifies other somatic cell managers. If other somatic cell managers have idle cells, they transfer the ownership of idle cell to the initiator, and then the initiator sets idle cell functions identical to those of its own cells. This strategy is similar to that observed in humans. When an erythrocyte dies, it is first replaced by the remaining erythrocytes (homogeneous cells); if a shortage of erythrocytes remains, the erythrocytes can be converted from hematopoietic stem cells. Suppose that these methods fail to work, the conversion between heterogeneous cells (e.g., from fibroblasts) is then initiated.

The fault-tolerant capability of eTissue is superior to traditional space redundancy techniques owing to its bio-inspired self-healing strategy. The *M*-of-*N* system is a typical space redundancy system. If the proper operating modules are less than *M*, the entire system loses its capacity for fault tolerance. However, in an eTissue with *N* cells, even if *N-1* cells fail, the system can still operate properly and obtain accurate results..

Experiments and results

We have implemented the eTissue prototype system with 10 cells using Verilog HDL, and evaluated it in Modelsim 6.2 through three sets of fault-injection experiments. We deployed a simple 3*3 mean filter algorithm in eTissue. The calculation of each pixel is given by the following expression

$$P'[x,y]=(P[x-1,y-1]+P[x-1,y]+\cdots+P[x+1,y+1])/9 \ . \ (1)$$

Equation (1) indicates that calculating each pixel requires 8 addition operations and 1 division operation. To simplify the experiments, we only injected faults to add cells. Table 1 summarizes the results of the fault-injection experiments (the number in the last column is the number of cycles used to calculate 4 pixels).

Experiment A. To verify the substitution among homo-geneous cells, we disabled the differentiation of adult stem cells and conversion between cells in experiment *A*. The initial numbers of healthy add cells and divide cells are 6 and 4, respectively. Table 1[*A*] shows that as the reducing of healthy add cells from 6 to 2, the number of cycles elapsed increases from 4985 to 10382. However, even when only two add cells remain, eTissue can still complete the execution because of the substitution among homogeneous cells.

Experiment B. To study the effectiveness of the conversion between cells (differentiation of adult stem cells and the conversion between heterogeneous cells both belong to this kind of case; we select conversion between heterogeneous cells to evaluate this situation). The cell

manager initiates this conversion under two conditions: (a) the accumulation of unsettled data in recycle FIFO exceeds threshold 2; (b) the number of cells that the cell manager owns is less than two. During the operational procedure of eTissue, the add cell manager requests idle cell from the divide cell manager when either of the above-mentioned conditions is satisfied. In experiment *B*, the initial number of cells and the frequency of fault-injection are the same as those in experiment *A*. The results show that eTissue can achieve better performance at the same fault-injection rates compared with experiment *A* because of the introduction of the mechanism of conversion between heterogeneous cells.

Experiment C. Both the differentiation of adult stem cells and conversion between heterogeneous cells were enabled in experiment *C* to verify three-level self-healing strategy. Two add cells are replaced by two stem cells compared with experiment B. In the current topology of the NoC, the distance between the add cell manager and stem cell manager is farther than that between the add cell manager and divide cell manager. Thus, the costs of converting a stem cell is appreciably higher than those of converting an idle divide cell, as shown in Table 1[*C*].

Table 1. Results of eTissue fault-injection experiments

Experiment	#Initial Add Cells	#Initial Stem Cells	#Initial Div Cells	#Fault-Injection in Add Cells	#Remaining Add Cells	#Remaining Stem Cells	#Remaining Div Cells	Cycles
A	6	0	4	0	6	0	4	4985
	6	0	4	2	4	0	4	5609
	6	0	4	4	2	0	4	10382
B	6	0	4	0	8	0	2	4732
	6	0	4	2	6	0	2	5270
	6	0	4	4	4	0	2	6203
C	4	2	4	0	8	0	2	4819
	4	2	4	2	6	0	2	5355
	4	2	4	4	4	0	2	6287

Conclusions

In this paper, we have presented a new bio-inspired fault-tolerant hardware system name eTissue. This prototype system consists of a number of cell managers and cells connected via an NoC. It imitates four biological principles, i.e., match-based recognition, substitution among homogeneous cells, differentiation of adult stem cells, and conversion between heterogeneous cells. Our eTissue derives the loosely-coupled relationship between data and cells from the imitation of the match-based recognition mechanism in protein sorting; this relationship equips eTissue with flexible cell replacement capability. Inspired by the substitution among homogeneous cells and conversion between cells, we proposed a three-level dynamic self-healing strategy. Fault-injection experiments show that eTissue has promising self-healing capabilities.

References

1. **Stauffer A., Mange D., Sanchez E., et al.** Embryonics: towards new design methodologies for circuits with biological–like properties // International Workshop on Logic and Architecture Synthesis. – Grenoble, 1995. – P. 299–306.
2. **Mange D., Sanchez E., Stauffer A.,Tempesti G., et al.** Embryonics: a new methodology for designing field–programmable gate arrays with self–repair and self–replicating properties // IEEE Transactions on Very Large Scale Integration (VLSI) System, 1998. – Vol. 6. – No. 3. – P. 387–399.
3. **Boesen M. R., Madsen J.** eDNA: a bio–inspired reconfigurable hardware cell architecture supporting self–organisation and self–healing // NASA/ESA Conference on Adaptive Hardware and Systems. – San Francisco, 2009. – P. 147– 154.
4. **Boesen M. R., Schleuniger P., Madsen J.** Feasibility study of a self–healing hardware platform // The 6th International Symposium on Applied Reconfigurable Computing. – Bangkok, Thailand, 2010. – P. 29–41.
5. **Alberts B., Johnson A., Lewis J., Raff M., Roberts K., Walter P.** Molecular Biology of the Cell, 5th ed. – Garland Science, 2008. – 1601 p.
6. **Szabo E., Rampalli S., et al.** Direct conversion of human fibroblasts to multilineage blood progenitors // Nature, 2010. – Vol. 468. – P. 521–526.

Jiaqing Xu, Yong Dou, Qi Lv. A Bio-Inspired Fault-tolerant Hardware System Supporting Hierarchical Self-healing // Electronics and Electrical Engineering. – Kaunas: Technologija, 2012. – No. 4(120). – P. 103–106.

This paper utilizes biological principles to design bio-inspired fault-tolerant system name eTissue. It imitates four biological principles: the match-based recognition mechanism in protein sorting equips eTissue a flexible cell replacement capability; substitution among homogeneous cells, differentiation of adult stem cells, and conversion between heterogeneous cells mechanisms compose a three-level dynamic self-healing strategy. Fault-injection experiments show that eTissue has promising self-healing capabilities. Ill. 2, bibl. 6, tabl. 1 (in English; abstracts in English and Lithuanian).

Jiaqing Xu, Yong Dou, Qi Lv. Biologiniu pagrindu sukurta klaidoms tolerantiška savigydė sistema // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2012. – Nr. 4(120). – P. 103–106.

Projektuojant klaidoms tolerantiškas sistemas, pavadintas eTissue, naudojami keturi biologiniai principai: atitikimais pagrįstas atpažinimo mechanizmas rūšiuojant baltymus suteikia eTissue gebėjimą lanksčiai pakeisti ląsteles; homogeninių ląstelių tarpusavio apsikeitimo, suaugusių kamieninių ląstelių diferenciacijos ir heterogeninių ląstelių tarpusavio pasikeitimo mechanizmai sukuria trijų lygių dinaminę savigydos strategiją. Klaidų įvedimo mechanizmai parodė, kad eTissue teikia daug žadančių savigydos galimybių. Il. 2, bibl. 6, lent. 1 (anglų kalba; santraukos anglų ir lietuvių k.).