

一种基于仿生原理的 Sobel 算子容错方法

吕 启¹ 窦 勇¹ 徐佳庆¹ 冯 雪²

(国防科技大学计算机学院 长沙 410073)¹ (中国人民解放军沈阳军区总医院 沈阳 110016)²

摘 要 提出一种基于仿生原理的 Sobel 算子容错方法。通过对蛋白质标记与识别、同类细胞替换、干细胞分化和异类细胞转换 4 种生物机制的模仿,设计了电子组织的结构,该结构具有层次化自修复的能力。用 MPI 编程的方式实现了根据 Sobel 算子定制的电子组织,并且通过故障注入实验验证了该方法对 Sobel 算子容错的可行性。

关键词 Sobel, 仿生, 容错

中图法分类号 TP302 **文献标识码** A

Bio-inspired Fault-tolerant Approach for Sobel Operator

LV Qi¹ DOU Yong¹ XU Jia-qing¹ FENG Xue²

(School of Computer, National University of Defense Technology, Changsha 410073, China)¹

(The General Hospital of Shenyang Military Command, PLA, Shenyang 110016, China)²

Abstract A bio-inspired fault-tolerant approach for Sobel operator was described in this paper. By imitating four biological principles, namely, match-based recognition in protein sorting, substitution among homogeneous cells, differentiation of stem cells, and conversion between heterogeneous cells, we designed the architecture of electronic tissue (eTissue), which supports hierarchical self-healing. We then implemented the eTissue architecture which is specific to Sobel operator by programming with MPI. Our fault-injection experiments prove the feasibility of this fault-tolerant approach.

Keywords Sobel, Bio-inspired, Fault-tolerant

1 概述

图像匹配技术是数字图像处理的一种重要技术,被广泛应用在公共安全与军事领域,如机场、地铁、银行等重要公共场所的安全监控和身份识别,侦察卫星对地面军事目标的识别等^[1]。然而,很多情况下,电子设备工作环境是十分恶劣的,系统可靠性成为一个至关重要的问题。以星载计算机为例,随着半导体技术的飞速发展,晶体管尺寸不断减小,空间环境中高能粒子的辐射效应造成的芯片瞬时故障率急剧增加,给其上的应用的正确运行带来了极大挑战。正因为如此,研究人员一直致力于寻找设计高可靠性电子设备的有效方法。

早在 20 世纪 50 年代, Von Neumann 就提出了通过冗余来提高系统可靠性的思想^[2]。此后,冗余作为容错的基本方法被广泛应用于可靠系统的设计中。冗余技术包括 3 大类:空间冗余、时间冗余和信息冗余。空间冗余是通过采用额外的资源来实现容错,最常用的是三模冗余(TMR);时间冗余则是通过重复计算再对结果进行表决,达到容错目的;信息冗余是通过添加冗余信息来实现检错和纠错,典型的检错码有

奇偶校验码和循环冗余码,典型的纠错码有 EDAC 和 ECC。然而,传统的容错技术给系统带来可靠性的同时也带来了巨大的开销:三模冗余将消耗倍于原系统的资源;时间冗余将使算法的执行时间增至原有执行时间的 3 倍以上;信息冗余技术则需要额外的编码解码器。

为了提高系统可靠性,人们努力寻找新的容错系统设计方法。生物体作为天然的容错系统,其自适应能力令人惊叹。近年来,计算机科研人员对生物体的自适应机制产生了浓厚的研究兴趣。通过对生物体自适应机制的学习和模仿,得到了许多仿生容错系统,典型代表有胚胎电子系统^[3]和 eDNA^[4-7]。它们均是对生物胚胎发育中的细胞分化和自我组织过程进行模仿,依赖于阵列中未分化的空闲细胞单元(胚胎干细胞)完成系统的自我修复。

本文提出一种利用电子组织^[8]结构对 Sobel 算子容错的方法。电子组织结构模仿了生物体的自修复机制,根据图像匹配中的典型应用 Sobel 算子定制该结构,可实现对 Sobel 算子的层次化的容错。本方法不需要耗费传统的容错方法(如 TMR)的大量硬件资源,并且容错能力也大于胚胎电子系统和 eDNA。

到稿日期:2011-06-02 返修日期:2011-11-03 本文受国家 863 高技术研究发展计划项目基金(2008AA01A201),国家自然科学基金(60833004)资助。

吕 启(1987-),男,硕士生,CCF 会员,主要研究方向为计算机体系结构、仿生硬件, E-mail:lvqi2009@gmail.com;窦 勇(1966-),男,教授,博士生导师,主要研究方向为计算机体系结构、可重构计算等;徐佳庆(1982-),男,博士生,主要研究方向为计算机体系结构、仿生硬件;冯 雪(1982-),女,博士生,主要研究方向为数字医疗。

2 基于仿生原理的容错电子组织

2.1 电子组织的仿生原理

组织是生物体中一些来源和结构相同、行使一定功能的细胞群。电子组织中同样包含了若干种结构相同的细胞单元,这些细胞单元相互合作,完成特定的功能。电子组织主要模仿了以下4种细胞生物学原理:蛋白质的标记与识别、同类细胞的替换、成体干细胞的分化和异类细胞的转化。

蛋白质的标记与识别是细胞内或细胞间信息及物质传递的基本方式。通过对蛋白质进行标记,可将其运送到特定的细胞器或其他细胞。以蛋白质激素为例,蛋白质激素首先在内分泌细胞的核糖体上合成;当它到达内质网后,内质网将会对其进行标记,添加类似于邮政编码的氨基酸序列,该序列决定了蛋白质激素的最终目的地;再经过高尔基体的糖基化修饰,以胞吐的方式分泌到细胞的外部;随后,分泌到细胞外的蛋白质激素通过与特定受体相匹配而被靶细胞所识别^[9]。电子组织模仿了这种机制,对输入数据和中间结果进行标记,以保证数据能被特定的细胞单元所识别和处理。

多细胞生物体内有大量的同类细胞。以红细胞为例,人体血管中有大量红细胞,一旦某个红细胞死亡,剩余的红细胞就代替死亡细胞完成氧气的输送任务。类似地,电子组织中设置了若干同类细胞单元,当某个细胞单元失效后,剩余的同类细胞单元能接替失效细胞单元完成数据处理,保证了电子组织的正常运行。

在成体组织或器官中,许多细胞仍具有自我更新、分化、产生不同组织细胞的能力,这些具备有限分化能力的细胞被称为成体干细胞。造血干细胞就是一种典型的成体干细胞,它能够分化成红细胞、白细胞和血小板。当人体中某类血液细胞减少时,造血干细胞将进行分化,以弥补血液细胞的减少。电子组织引入了这种自修复机制,当系统中某类细胞单元严重不足时,成体干细胞进行分化,以实现对此类细胞单元的补充。

2010年11月,加拿大的E. Szabo等人在《自然》杂志上发表论文,他们通过对皮肤细胞重新编程,实现了皮肤细胞到血液细胞的转化^[10]。该发现改变了人类对细胞分化的认识,意味着通过对细胞的重新编程可以实现各种细胞间的相互转化。电子组织引入了这种异类细胞的转化机制,增强了电子组织的自我修复能力。

通过对以上4种生物学原理的模仿,电子组织具备了层次化的自我修复能力。

2.2 电子组织的结构

电子组织的体系结构如图1所示,它由输入控制单元、细胞管理器、细胞单元、数据回收单元和结果收集单元构成。各个单元之间通过消息传递进行交互。

输入控制单元:负责从存储器中取出输入数据,根据附在数据上的标记将输入数据分发到不同的细胞管理器。

细胞管理器:整个电子组织系统的核心单元。为简化细胞单元的功能,我们将同类细胞单元的管理功能集中到一个细胞管理器中。细胞管理器由数据FIFO、数据回收FIFO、细胞控制器、匹配表和细胞状态表构成,它主要具有以下两大功能。

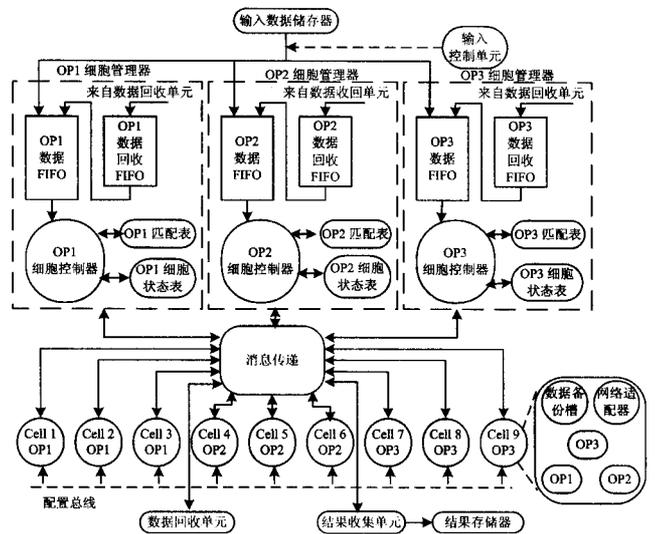


图1 电子组织结构示意图

数据到细胞单元的分派。细胞控制器根据数据的标记及匹配表和细胞状态表的内容将数据分派到细胞单元进行处理。

完成异类细胞间的转化。细胞管理器通过监控数据回收FIFO及细胞状态表,决定执行细胞间转化的时机。当其所属细胞单元过少时,细胞管理器向其他细胞管理器发出细胞间的转化的请求信号。如果其他细胞管理器有空闲的细胞单元,则响应该请求信号并将空闲细胞单元的控制权转让给请求的细胞管理器,细胞管理器再将该细胞配置成其所属细胞单元。

细胞单元:用于数据处理,能够完成加、减、乘、除和比较等算术逻辑操作,由数据备份槽、网络适配器和运算单元构成。数据备份槽用于备份细胞单元中待处理的数据,以防细胞单元失效导致的数据丢失;网络适配器不仅负责单元间的通讯,而且负责监控细胞单元的状态;运算单元完成数据的处理,细胞单元内部有多个运算单元,通过细胞分化表选取运算单元,进而决定细胞单元的种类。已分化的细胞单元称为体细胞单元,未分化的细胞单元称为成体干细胞单元。管理体细胞单元的细胞管理器称为体细胞管理器,管理干细胞单元的细胞管理器即为干细胞管理器。

数据回收单元:负责回收细胞管理器暂时无法处理的数据和故障细胞单元中的备份数据,并根据数据的标记将回收数据发送到对应细胞管理器的数据回收FIFO中。

结果收集单元:负责计算结果的收集,并将结果保存于存储器中。

2.3 电子组织的容错策略

电子组织拥有层次化的自我修复策略,该策略包括了同类细胞单元的替换、成体干细胞的分化和异类细胞单元的转化。

同类细胞单元的替换:当网络适配器监测到电子组织中的某个细胞单元发生故障时,网络适配器向主管该细胞单元的细胞管理器发出相应的细胞故障消息,并将保存在数据备份槽中的备份数据发送到数据回收单元。数据回收单元根据回收数据的标记将其发送到对应的细胞管理器的数据回收FIFO中,等待细胞管理器的重新分派。

干细胞单元的分化:当细胞管理器监测到其所属的细胞

单元过少(有两种情况会到导致细胞单元过少:一是用户分配细胞单元不足,不能与应用相适应;二是由于细胞单元失效,导致细胞单元数量减少)时,体细胞管理器会向干细胞管理器发出干细胞分化的请求。如果尚有干细胞未分化,干细胞管理器给该体细胞管理器发送一个响应信号,体细胞管理器再把该干细胞配置为其所属的细胞类型,以弥补其细胞数目的不足。

异类细胞单元的转化:当细胞数目过少而所有干细胞都已经分化完时,体细胞管理器向其他体细胞管理器发出细胞间的转化的请求信号。如果其他体细胞管理器有空闲的体细胞单元,则响应该请求信号并将空闲体细胞单元的控制权转让给请求的体细胞管理器。体细胞管理器再将该细胞配置成其所属细胞单元,从而完成异类细胞单元间的转化。

以上几种机制构成了一个三层的自我修复策略:第一层是同类细胞单元的替换;第二层是成体干细胞的分化;第三层是异类细胞单元的转化。该修复策略与人体血液系统的修复策略极为相似:当个别红细胞死亡后,剩余的红细胞可以替代死亡红细胞的功能;造血干细胞源源不断地生成新的红细胞对死亡红细胞进行补充;而对于白血病患者,或许在不久的将来就能够使用上利用自身皮肤细胞转化出来的血浆。

3 Sobel 边缘检测算子介绍

图像匹配包括了中值滤波、边缘检测、二值化和距离变换几个主要过程^[1],本文主要关注图像边缘检测技术。从本质上说,图像边缘是图像局部特性不连续性(灰度突变、颜色突变等)的反映,它标志着—个区域的终结和另一个区域的开始。图像边缘具有方向和幅度两个特征。通常,沿边缘走向的像素变化平缓,而垂直于边缘方向的像素变化剧烈。从这个意义上说,提取边缘的算法就是检出符合边缘特性的像素点的数学算子。通常,边缘像素点通过对各个像素点进行微分或求二阶微分来确定,一阶微分的峰值处和二阶微分的零点即对应图像的边缘点。

常用的图像边缘检测算子有 Sobel 算子、Laplace 算子、Canny 算子、Robert 算子、Prewitt 算子等。在精度要求不高的情况下,Sobel 算子是最为常用的一种算子。

Sobel 边缘检测算子的表达式为 $S(i, j) = \sqrt{f_x^2 + f_y^2}$ 。为计算方便,可用两个分量的绝对值之和来近似,即 $S(i, j) = |f_x| + |f_y|$ 。基于 Sobel 算子的边缘检测,是在图像空间上利用图 2 所示的两个方向上的模板与图像进行邻域卷积来完成的。

$$x \text{ 方向: } \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad y \text{ 方向: } \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

图 2 Sobel 算子卷积模板

根据图 2 的卷积模板得到 Sobel 算子的近似表达式为

$$S(i, j) = |f_x| + |f_y| \\ = |x_{i-1, j+1} + 2x_{i, j+1} + x_{i+1, j+1} - (x_{i-1, j-1} + 2x_{i, j-1} + x_{i+1, j-1})| + |x_{i+1, j-1} + 2x_{i+1, j} + x_{i+1, j+1} - (x_{i-1, j-1} + 2x_{i-1, j} + x_{i-1, j+1})| \quad (1)$$

不难发现,Sobel 边缘检测算子综合了每个像素点相邻 8 个点的灰度,而且接近模板中心的权值较大。Sobel 算子结合

了 Gaussian 平滑和微分,因此不但可以产生较好的边缘效果,而且对噪声具有一定的鲁棒性。

4 根据 Sobel 算法定制的容错电子组织

根据式(1)可以得到 Sobel 算子对应的数据流图(DFG),如图 3(a)所示。容易知道,对一个像素点进行处理,需要用到 9 个加法操作、2 个减法操作、4 个乘法操作和 2 个取绝对值(ABS)操作。在数据流图中,对每一种操作的输入数据和中间结果进行了标记。数据标记由操作类型和数据序号组成:操作类型用于指示数据需要进行何种操作(1 表示加法操作,2 表示减法操作,3 表示乘法操作,4 表示取绝对值操作,9 表示最后结果);数据序号用于区别进行同类操作的不同数据。例如,数据标记为 104h 的数据表示需要进行加法操作的第 4 个数据。

将 Sobel 算子的数据流图进行转化,就得到了其匹配表,如图 3(b)所示。该匹配表描述了 Sobel 算子的运算过程。其中,优先级用来区分有数据依赖关系的同类操作(Priority 越小,优先级越高),通过标记同类操作的优先级可以防止由于低优先级数据占据细胞单元而造成的系统死锁。

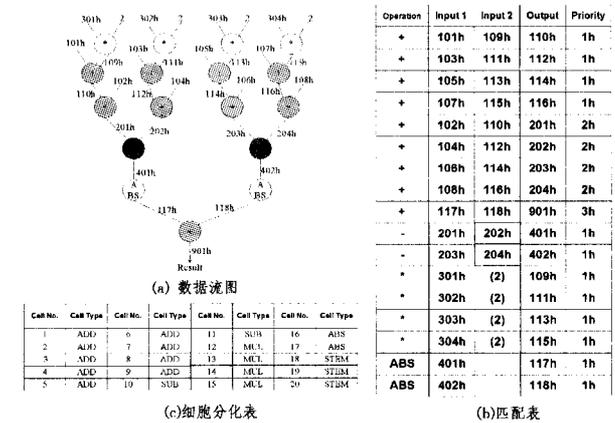


图 3 Sobel 算子数据流图和 DNA

随后,我们得到图 3(c)所示的细胞分化表,它决定了电子组织内部各类细胞单元的数量,细胞单元根据细胞分化表对其运算单元进行配置。细胞分化表与匹配表共同构成了电子组织的 DNA。

在进行数据处理前,电子组织根据 DNA 进行分化,不同类型的体细胞管理器形成各自的细胞状态表,如图 4 所示。细胞状态表包括了细胞编号、第一操作数、第二操作数、优先级,以及 Active、Available、State 几个标志位。Active 表明了细胞是否为有效,出故障时置为 0;Available 表明细胞管理器是否对该细胞拥有控制权,它在异类细胞转化时用到;State 表明了细胞拥有操作数的状态,细胞空闲时 State 为 0,一个操作数到达时为 1,两个操作数到达时为 2。

下面介绍由 Sobel 算子定制的电子组织对数据的处理过程。输入控制单元从存储器中取出输入数据,根据附着在数据上的标记将数据分派给不同的细胞管理器(操作类型为 1,分发给加法细胞管理器;为 3 时分发给乘法细胞管理器)。细胞管理器执行图 5 所示的数据匹配和分派算法,将输入数据发送到所属的细胞单元进行处理。首先,细胞管理器从数据 FIFO 中读出数据,根据输入数据的标记从匹配表中读出与其相匹配的操作数标记、输出数据标记和优先级等信息;查询细

胞状态表,判断是否已经为其匹配操作数分配了细胞单元(State 等于 1)。如果存在这样的细胞单元,则将输入数据发送给对应细胞单元进行数据处理,并更新细胞状态表;如果没有查找到匹配的细胞单元,但存在空闲细胞单元,则为输入数据分配一个空闲细胞单元并更新细胞状态表,等待其他操作数的到达;如果没有空闲细胞单元,则查找是否有低优先级的数据占据了细胞单元。如果存在这样的低优先级数据,则将其驱逐到数据回收单元中,将该细胞单元重新分配给高优先级的输入数据,并更新细胞状态表;如果以上情况均不满足,则将输入数据直接送到数据回收单元,等待下一次的分派。当细胞单元所需的操作数全部达到后,细胞单元开始进行数据处理(ABS 操作只需要得到一个操作数就开始计算,乘法操作是与 2 相乘,也只需要一个操作数就开始计算),并根据处理结果的标记将其发送到相应的细胞管理器进行后续处理;一旦得到最终结果,则将结果发送到结果收集单元并保存到存储器中。

Cell No.	Active	Available	State	Input1	Input2	Priority
1						
2						
3						
4						
5						
6						
7						
8						
9						

Cell No.	Active	Available	State	Input1	Input2	Priority
12						
13						
14						
15						

Cell No.	Active	Available	State	Input1	Input2	Priority
16						
17						

Cell No.	Active	Available	State	Input1	Input2	Priority
10						
11						

Cell No.	Active	Available	State	Input1	Input2	Priority
18						
19						
20						

图 4 各类细胞管理器的细胞状态表

我们采用 MPI 编程的方式实现了根据 Sobel 算子定制的电子组织。每个单元用一个进程来表示,各个进程分别表示输入控制单元、加法细胞管理器单元、减法细胞管理器单元、乘法细胞管理器单元、绝对值细胞管理器单元、干细胞管理器单元、结果收集单元以及各类细胞单元。进程之间通过 MPI 消息的方式进行交互。每一类运算都有一个数据 FIFO 和数据回收 FIFO,采用循环队列的方式实现。

```

BEGIN:
  For each incoming labeled data do {
    S1: Look up the Match Table to get the tag of input data's match data;
    S2: if (match data is already in some cell) then { //Look up Cell State Table for match data;
      Send input data to this cell;
      Update Cell State Table; }
    S3: else
    S3.1: if (some cell is idle) then { //Look up Cell State Table for idle cell;
      Send input data to this idle cell;
      Update Cell State Table; }
    S3.2: else
    S3.2.1: if (some low-priority cell exists) then { //Look up Cell State Table for low-priority cell;
      Eject data in low-priority cell;
      Send input data to this cell;
      Update Cell State Table; }
    S3.2.2: else
      Send input data to Data Recycle FIFO;
  };
END

```

图 5 细胞管理器的数据匹配与分派算法

若某一细胞出错,它会给其对应的细胞管理器发送一个 Cell_To_CellManager_DeadNotify 的消息以通知细胞管理器该细胞单元出错,随后将出错单元的备份数据用 Cell_To_CellManager_Recycle_Data 类型的消息送回细胞管理器。细胞管理器在收到细胞出错消息后,将细胞状态表中该细胞的 Active 属性置为 0,随后再将收到的回收数据加到数据回收

FIFO 中。

采用 2.3 节所述的自修复策略对 Sobel 算子进行容错处理。同类细胞替换的实现比较简单,若某个细胞出错,回收数据会被分派到同种类的其他细胞进行计算。干细胞分化与异类细胞转化类似,若某一类细胞初始分配不当或者失效细胞数目过多,会向干细胞管理器或其它体细胞管理器发出一个 CellManager_To_OtherCellManager_Request 类型的转化请求消息,干细胞管理器或其它细胞管理器查看细胞状态表,如果有空闲细胞,则释放对该细胞的控制权(置 Available 为 0),将细胞编号通过 CellManager_To_OtherCellManager_Response 消息发送给请求者。随后,细胞管理器根据收到的细胞编号,向该细胞发送 CellManager_To_Cell_Config 类型的消息,将其配置为其所属细胞单元。通过这样的方式,根据 Sobel 算子定制的电子组织就具备了 3 个层次的容错能力。

5 实验结果

我们通过几组故障注入实验来验证基于电子组织结构的 Sobel 算子的容错方法的可行性。为简便起见,我们只在加法单元进行故障注入。初始时,各类细胞的数目为加法细胞单元 9 个、减法细胞单元 2 个、乘法细胞单元 4 个、绝对值细胞单元 6 个、干细胞单元 6 个。每次实验中,运行了 1000 次 Sobel 算子操作,得到的运行时间如表 1 所列。

表 1 电子组织故障注入实验结果

实验	故障注入次数	运行时间(s)
0	0	8.104
(无故障注入)		
1	2	12.331
(同类细胞替换)	4	17.565
2	2	15.106
(干细胞分化)	4	19.170
3	2	15.186
(异类细胞转化)	4	20.329
4	2	14.622
(三种自修复策略)	4	18.618

在实验 0 中不注入故障,得到的运行时间为 8.104s。实验 1 中,仅采用同类细胞替换策略,对加法细胞单元故障注入次数分别为 2 次和 4 次。基于该策略,Sobel 算子操作在每组实验中仍可以得到正确结果,但由于细胞替换的额外通信开销,运行时间分别增加到 12.331s 和 17.565s。实验 2 中,仅采用干细胞分化的策略,加法细胞单元出错个数仍为 2 个和 4 个。由于需要进行的通信比同类细胞替换更多,故消耗的时间比实验 1 略多。实验 3 是为了验证异类细胞转化,在实验中仅开启此容错策略,由表 1 可发现其运行时间与干细胞分化相当。实验 4 中,同时开启同类细胞替换、干细胞分化和异类细胞转化 3 种机制,在相同故障注入情况下,运行时间为 14.622s 和 18.618s。通过这几组故障注入实验,可以得出结论:在细胞出现故障情形下,电子组织能够通过这样一种层次化的自修复策略达到容错目的。

结束语 本文提出了一种基于仿生原理的 Sobel 算子容错方法。通过对生物体自修复机制的模仿,电子组织具备了层次化的自修复能力。我们用 MPI 编程的方式实现了根据 Sobel 算子定制的电子组织,并用几组故障注入实验证明了本容错方法的可行性。

(下转第 299 页)

从表4可以看出,对于不同的时序电路,本方案相对于其它几种方案都有不同程度的改善。本方案与混合定变长码的平均差值压缩效率为8.48%;与混合定变长虚拟块方案的平均差值压缩效率为4.30%;与共游程码方案的平均差值压缩效率为4.82%。这些数据充分说明了本方案的有效性。

结束语 为了进一步降低测试成本,本文提出了一种相对游程长度编码的测试数据压缩方案。本方案独立于被测试电路。因此,其特别适用于没有结构信息的IP核的测试,与同类编码方案比较,本方案特别适用于测试集中游程长度相差较小的情况。

参 考 文 献

[1] Touba N A. Survey of test vector compression technique[J]. IEEE Design & Test of Computers, 2006, 23(4): 294-303

[2] Zhan Wen-fa, Ma Jun, Du Shan-shan, et al. A LFSR Reseeding Scheme Based on Division by 2 to the Power of Integer[J]. International Journal of Digital Content Technology and its Applications, 2010, 4(9): 88-96

[3] Zhan Wen-fa, Liang Hua-guo, Jiang Cui-yun, et al. A Scheme of Test Data Compression Based on Coding of Even Bits Marking and Selective Output Inversion [J]. Computers and Electrical Engineering, 2010, 36(5): 969-977

[4] Zhan Wen-fa, Liang Hua-guo, Shi Feng, et al. Test data compression scheme based on variable-to-fixed-plus-variable-length coding[J]. Journal of Systems Architecture, 2007, 53(11): 877-887

[5] Chandra A, Chakrabarty K. Test data compression and decompression based on internal scan chains and Golomb coding[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2002, 21(6): 715-722

[6] Chandra A, Chakrabarty K. Test data compression and test resource partitioning for system-on-a-chip using frequency-directed run-length(FDR) codes[J]. IEEE Transactions on Computers, 2003, 52(8): 1076-1088

[7] El-Maleh A H. Test data compression for system-on-a-chip using extended frequency-directed run-length code[J]. IET Computers & Digital Techniques, 2008, 2(3): 155-163

[8] 梁华国, 蒋翠云. 基于交替与连续长度码的有效测试数据压缩和解压[J]. 计算机学报, 2004, 27(4): 548-554

[9] 韩银和, 李晓维, 徐勇军, 等. 应用 Variable-Tail 编码压缩的测试资源划分方法[J]. 电子学报, 2004, 32(8): 1346-1350

[10] 彭喜元, 俞洋. 基于变游程编码的测试数据压缩算法[J]. 电子学报, 2007, 35(2): 197-201

[11] 詹文法, 梁华国, 时峰, 等. 混合定变长码的测试数据压缩方案[J]. 计算机学报, 2008, 31(10): 1826-1834

[12] 詹文法, 梁华国, 时峰, 等. 一种混合定变长虚拟块游程编码的测试数据压缩方案[J]. 电子学报, 2009, 37(8): 1837-1841

[13] 詹文法, 梁华国, 时峰, 等. 一种共游程码的测试数据压缩方案[J]. 计算机研究与发展, 2008, 45(10): 1646-1653

[14] 许川佩, 董祥健. 一种交替游程编码的 SOC 测试数据压缩方法[J]. 计算机工程与应用, 2010, 46(25): 57-60

[15] 商进, 张礼勇. 一种混合相容数据块的测试数据压缩方案[J]. 哈尔滨理工大学学报, 2011, 16(6): 93-96

(上接第290页)

为简便起见,我们将同类细胞的控制和管理功能集中在细胞管理器中,使得细胞管理器成了可靠性单点。下一步工作中,将研究实现一种分布式的结构,亦即将细胞管理器的集中管理功能分散到各个细胞单元中去,通过细胞单元的内部信息以及细胞单元间的信息交换实现数据的正确分派及系统的自我修复。

参 考 文 献

[1] 李俊丰. 图像匹配应用的硬件加速技术研究[D]. 长沙:国防科技大学计算机学院, 2006

[2] Neumann J V. Probabilistic logics and the synthesis of reliable organisms from unreliable components[M]//Shannon C E, McCarthy J. Automata Studies. Princeton: Princeton Univ Press, 1954: 43-98

[3] Stauffer A, Mange D, Sanchez E, et al. Embryonics: towards new design methodologies for circuits with biological-like properties [C]//Proceedings of 1995 International Workshop on Logic and Architecture Synthesis. Grenoble, France, 1995: 299-306

[4] Boesen M R, Madsen J. eDNA: a bio-inspired reconfigurable hardware cell architecture supporting self-organisation and self-healing[C]// Proceedings of 2009 NASA/ESA Conference on Adaptive Hardware and Systems (AHS2009). San Francisco,

USA, 2009: 147-154

[5] Boesen M R, Schleuniger P, Madsen J. Feasibility study of a self-healing hardware platform[C]//Proceedings of the 6th International Symposium on Applied Reconfigurable Computing (ARC2010). Bangkok, Thailand, 2010: 29-41

[6] Boesen M R, Madsen J, Pop P. Application-Aware Optimization of Redundant Resources for the Reconfigurable Self-Healing eDNA Hardware Architecture[C]// Proceedings of 2011 NASA/ESA Conference on Adaptive Hardware and Systems (AHS2011). San Diego, USA, 2011: 66-73

[7] Boesen M R, Madsen J, Keymeulen D. Autonomous distributed self-organizing and self-healing hardware architecture-the eDNA concept[C]// Proceedings of the 2011 IEEE Aerospace Conference. Big Sky, Montana, USA, 2011: 1-13

[8] Xu Jia-qing, Dou Yong, Lv Qi, et al. eTissue: a bio-inspired match-based reconfigurable hardware architecture supporting hierarchical self-healing and self-evolution [C]// Proceedings of 2011 NASA/ESA Conference on Adaptive Hardware and Systems (AHS2011). San Diego, USA, 2011: 311-318

[9] Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell (The 5th edition)[M]. Oxford: Garland Science, 2008

[10] Szabo E, Rampalli S, et al. Direct conversion of human fibroblasts to multilineage blood progenitors[J]. Nature, 2010, 468(7323): 521-526