

可重构电子系统芯片级在线自主容错方法研究^{*}

袁 鹏, 王友仁, 张 砦

(南京航空航天大学 自动化学院, 南京 210016)

摘 要: 可重构电子系统芯片固定型故障的传统容错设计往往采用集中式控制方法, 存在测试时间长、硬件资源利用率低、对外部控制器依赖性高等问题。因此, 设计了一种具有分布式自主容错能力的可重构细胞阵列, 通过将细胞内部查找表输出与参考值进行比较的方式进行循环检测, 并利用冗余存储单元对故障查找表进行修复。以四位并行乘法器为例进行仿真验证, 实验结果表明, 新型可重构阵列的自主容错设计方法, 比现有设计的硬件开销小, 修复时间短, 容错能力强, 且设计复杂度不受阵列规模影响。

关键词: 可重构电子系统; 分布式控制; 细胞阵列; 芯片级在线自主容错

中图分类号: TP302.8 **文献标志码:** A **文章编号:** 1001-3695(2012)06-2172-04

doi:10.3969/j.issn.1001-3695.2012.06.045

Research on chip-level autonomous on-line fault-tolerant method for reconfigurable electronic systems

YUAN Peng, WANG You-ren, ZHANG Zhai

(College of Automation Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract: The traditional fault-tolerant designs for reconfigurable electronic systems on-chip stuck-at fault mainly use the centralized control methods, which usually cause long testing time, low hardware resources utilization rate, and high dependence on external controller. To solve these problems, this paper designed a novel reconfigurable cell array with distributed autonomous fault-tolerant ability, each cell could implement cyclic self-detection by comparing LUTs' output with reference value, and used redundant storage unit to repair fault LUTs. Chose a 4 bits parallel multiplier as an example, the experiment result demonstrates that, compared with the existing design methods, this novel reconfigurable array has less hardware cost, shorter self-repair time, better fault-tolerance ability, and the design complexity isn't influenced by the array scale.

Key words: reconfigurable electronic systems; distributed control; cell array; autonomous on-line fault-tolerance at chip-level

0 引言

随着集成电路特征尺寸的缩小、工作频率的提升和电源电压的降低, 芯片故障率大幅提高, 保障芯片高可靠性的容错设计研究成为亟待解决的问题。可重构硬件作为可编程器件具有集成度高、结构灵活、开发周期短及研发成本低等优点, 为电子系统开展先进容错设计方法研究提供了硬件基础, 现已广泛应用于航天航空、核电监控等恶劣环境下的电子系统中。目前, 以 FPGA 为代表的可重构硬件的重构过程往往需要软件与外部处理器结合进行集中式控制^[1~3], 不仅重构计算复杂、重构时间长, 且系统的软硬件成本也大大增加。而在分布式自主控制的方式下, 控制逻辑分布在各重构单元中, 容错算法较简单; 重构速度快。因此, 开展分布式自主容错型可重构硬件结构设计及容错机制的研究, 对提高电子系统可靠性与硬件利用率、减少体积与重量具有重要意义, 已成为新型可重构硬件的研究热点之一。

传统的可重构硬件芯片级容错设计主要针对可配置逻辑模块和互连资源两部分。可配置逻辑模块是可重构硬件的基

本逻辑执行单元, 目前包括主流 FPGA 在内的多数可重构硬件都是基于查找表(LUT)技术, 而 LUT 存储单元故障是造成可配置逻辑模块故障的主要原因^[4]。针对逻辑模块故障的容错方法主要有以下几类。第一类是基于硬件冗余的方法。最常用的是三模冗余(TMR)^[5,6], 其实现简单, 但无法定位故障且硬件开销大; 为了实现故障定位, 可采用双模比较及并发故障检测(DWC-CED)的方法^[7], 在基本双模冗余比较的基础上增加了并发故障检测模块, 实现了故障的模块级定位, 但只能处理单故障。第二类是基于软件和外部处理器通过重布局布线实现的方法。系统在检测到故障后, 将故障资源信息存入数据库, 在随后的逻辑映射和布局布线时, 根据数据库中的故障信息避开故障资源。这类方法不需要增加冗余电路, 灵活性好, 但需要较强的计算资源且运行时间较长。第三类是基于内建自测试(BIST)的方法^[8~11]。测试向量的产生与响应分析由芯片自身硬件完成, 降低了测试成本, 但多数 BIST 方法是离线的^[10]。目前在线 BIST 主要采用 roving STARs 的方法^[11], 利用可重构硬件的动态重构特性进行循环检测, 但仍需要结合外部处理器进行控制, 无法完全由芯片自主实现。第四类方法是基

收稿日期: 2011-11-03; **修回日期:** 2011-12-12 **基金项目:** 航空科学基金资助项目(2011ZD52050); 南京航空航天大学基本科研业务费专项科研资助项目(NS2010086)

作者简介: 袁鹏(1987-), 男, 硕士研究生, 主要研究方向为可重构硬件容错技术(yuan_nuaa_2005@163.com); 王友仁(1963-), 男, 教授, 博导, 主要研究方向为电子系统自主容错、故障诊断预测、信号处理等; 张砦(1980-), 男, 讲师, 博士研究生, 主要研究方向为数字仿生硬件与应用技术。

于新型仿生容错机制的可重构硬件结构。Canham 等人^[12]和 Szász 等人^[13]提出了一种具有自修复能力的胚胎电子系统,具有自主重构、修复快速等优点,但冗余资源开销大、利用率低;Zhang 等人^[14]和 Canham 等人^[15]结合生物免疫机制,设计了一种新型可重构硬件结构,通过设置免疫层电路可实现芯片级自测试与自修复,但免疫层消耗硬件巨大,且需要进行离线免疫学习。

在面向航空航天、核电监测等对自检测、自修复能力有特殊要求的环境中,现有可重构硬件容错方法的自主性、硬件利用率和快速性难以满足需求,需要研究分布式在线自主容错设计方法。本文采用分布式容错机制,设计了一种具有在线自主容错能力的可重构阵列,阵列中的细胞能够针对自身逻辑模块内部的 LUT 存储单元故障进行检测与修复。仿真实验表明,故障检测过程能够在线循环执行,不影响细胞在阵列中的正常工作,故障修复速度快,容错性能得到提升。

1 可重构阵列结构

1.1 阵列体系结构

为了实现可重构阵列的分布式自主容错,本文设计了一种基于二维细胞矩阵的可重构阵列结构,如图 1 所示。细胞是阵列中的基本可重构单元,每个细胞由细胞功能单元和开关块构成。细胞功能单元是细胞的逻辑执行单元,内部设置的检测电路能够对可配置逻辑模块进行在线循环检测,修复电路可以对 LUT 存储单元故障进行修复。细胞之间通过开关块进行数据通信。

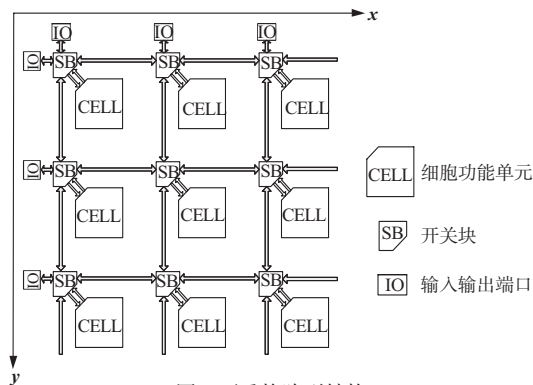


图1 可重构阵列结构

1.2 开关块结构

开关块是基于多路器设计的,每个开关块有四个通道,如图 2(a) 所示。每个通道由五个 4 输入 1 输出多路器构成,如图 2(b) 所示,每个方向的输入可以连接到其他三个方向的输出和细胞功能单元的输入,而细胞功能单元的输入可以连接到 N、E、S、W 四个方向。多路器的配置数据有三位,最高位为多路器使能位,低两位用来选择连接到输出端的输入信号。

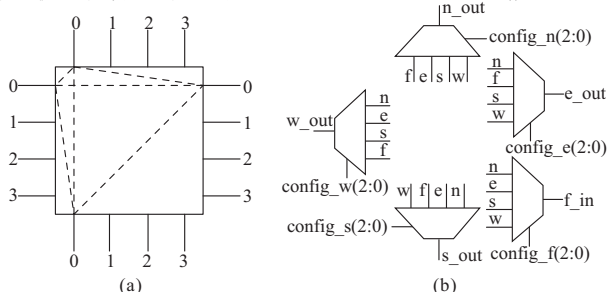


图2 布线开关块结构

1.3 细胞功能单元

在传统的可重构硬件中,以典型的基于 LUT 的 FPGA 为例,逻辑功能由可配置逻辑模块实现,每个逻辑模块由查找表、控制逻辑单元以及触发器单元构成。由于在辐射等恶劣环境下,可配置逻辑模块中的查找表很可能会发生存储单元的固定‘0’或固定‘1’故障,从而引发系统错误,而传统的可配置逻辑模块结构无法满足自主容错的需求。

本文设计的细胞功能单元结构如图 3 所示。可配置逻辑模块内部的基本逻辑执行单元包括四个功能查找表、一个 D 触发器及若干多路器和门电路,可以实现组合逻辑和时序逻辑功能。自测试执行单元包括查找表 X (LUT-X)、计数器、存储器以及查找表 X 诊断器。为了实现循环自测试,由计数器控制给查找表 X 写入与待测功能查找表相同的配置数据,然后以查找表 X 的输出作为参考值,将其与待测功能查找表的输出进行比较,以此来判断可配置逻辑模块中的功能查找表是否发生故障。当对所有功能查找表检测完一遍后,自动转入对查找表 X 进行诊断的流程,查找表 X 自诊断完成后,计数器清 0,开始下一轮检测。在自修复方面,可配置逻辑模块中的每个功能查找表都配有修复模块,一旦检测到故障,比较器发出的故障信号会触发相应的修复电路对故障查找表进行修复。

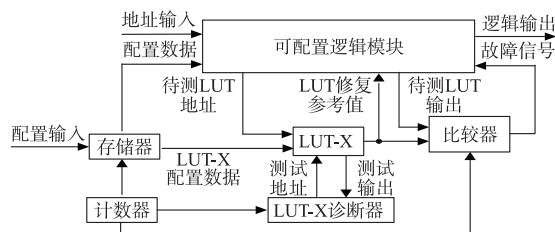


图3 具有容错功能的细胞功能单元结构

2 可重构阵列自主容错机制

可重构阵列主要针对可配置逻辑模块中 LUT 存储单元的固定型故障进行自主容错,以下将分别详细介绍故障自测试和自修复方法。

2.1 故障自测试方法

2.1.1 功能查找表的自检测方法

针对 LUT 存储单元的固定型故障,传统的查找表测试方法大多是离线测试,在线的方法一般有两种。一种是双模冗余比较,每一个查找表都需要备份;另一种是将待测查找表的配置数据先转移到其他冗余查找表中,将待测查找表空闲出来后用测试向量进行诊断。后者需要将待测查找表的配置数据和输入信号进行转移,测试速度相对较慢,且会降低系统的工作性能。

为了解决上述问题,本文采用将功能查找表的输出与参考值进行比较的方法,原理如图 4 所示。设置了一个专门用于检测的查找表 X,将查找表 X 的输出作为参考值,对四个功能查找表进行循环比较。检测过程中,功能查找表保持正常工作。在比较之前先给查找表 X 写入和待测功能查找表相同的配置数据,同时将待测功能查找表的地址输入连接到查找表 X 的地址输入端(通过计数器的 count 信号来控制),然后比较二者的输出,若输出一致,则说明无故障;若输出不一致,则判定为待测查找表当前地址对应的存储单元发生故障。

功能查找表的检测过程在线自主地循环执行,不需要用测

的存储单元发生了固定‘1’故障,触发查找表 X 故障信号 self_test_fail。

2) 仿真结果分析 由图 7 可知,当写入故障测试配置“0001”后,在测试地址“11”时(test0_11 阶段),由于查找表 X 输出的结果 test_lutx = ‘1’(正确值应为‘0’),说明此处发生了固定‘1’故障,self_test_fail 信号置‘1’,指示查找表 X 发生故障。

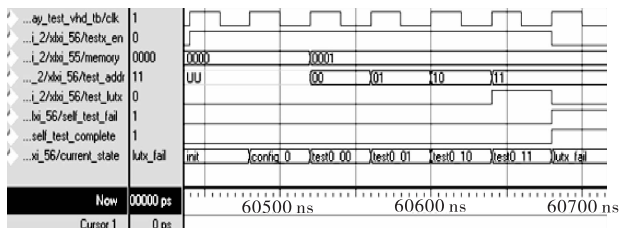


图7 查找表X故障仿真时序

3.2 容错性能分析

1) 硬件开销分析 本文设计的可重构硬件结构在 Virtex 2 系列 XC2V8000 的 FPGA 芯片上进行仿真综合,以综合后消耗的 XC2V8000 FPGA 芯片的 slice 数量作为硬件开销分析的参考数据。由于采用分布式容错机制,可重构阵列所有细胞结构相同,因此可以用单个细胞的硬件开销来说明分析。硬件资源开销情况如表 2 所示,其中硬件开销增量是指本文所设计的容错细胞单元相比于基本细胞单元(无容错能力)所增加的硬件资源百分比,用来衡量自主容错所需的附加硬件资源的情况。

表2 硬件资源开销

项目	slices	LUT	FFs
基本细胞单元	82	106	48
容错细胞单元	153	232	223
硬件开销增量/%	86.6(8 以 slices 为标准)		

采用本文设计,在对每个功能查找表只设一个冗余存储位的情况下,硬件资源开销增加 86.6%,能使单个细胞最大容错四次。

若要增加细胞的容错次数,只需给修复模块增加故障地址存储单元和冗余存储位。如此一来,比设置一个冗余存储位的方案只增加少量的硬件开销(表 3),就能使细胞的容错次数上限增加一倍。由此可见,本文采用的容错方法对于多位 LUT 存储单元故障具有更好的容错能力。

表3 不同容错次数的硬件开销

	容错细胞单元(slices)	硬件开销增量/%	细胞最大容错次数
每个 LUT 容错 1 次	153	86.6	4
每个 LUT 容错 2 次	179	118.3	8

2) 时间开销分析 本文方法的容错过程分为功能查找表的检测、修复以及查找表 X 的自诊断三个阶段。设这三个阶段的时间开销分别是 $T_{\text{lut-test}}$ 、 $T_{\text{self-repair}}$ 和 $T_{\text{x-diagnosis}}$ 。首先,每个功能查找表的检测依次需要进行两个步骤:给查找表 X 写配置数据、打开比较模块进行判断,分别耗时 $3T_{\text{clk}}$ 和 $1T_{\text{clk}}$,因此 $T_{\text{lut-test}} = 3T_{\text{clk}} + 1T_{\text{clk}} = 4T_{\text{clk}}$;其次修复阶段耗时两个时钟周期,即 $T_{\text{self-repair}} = 2T_{\text{clk}}$;最后在对四个功能查找表测试完成后,进入查找表 X 的自诊断阶段,该阶段耗时 $T_{\text{x-diagnosis}} = 12T_{\text{clk}}$ 。由于本文采用的是分布式的容错方法,所以单个细胞的容错时间就是整个可重构硬件的容错时间,因此总时间开销为

$$T = 4 \times (T_{\text{lut-test}} + T_{\text{self-repair}}) + T_{\text{x-diagnosis}} = 36T_{\text{clk}}$$

本文方法的仿真实验中设置的时钟频率为 25 MHz,周期 $T_{\text{clk}} = 40 \text{ ns}$,则整个可重构硬件完成一次检测的总时间开销 $T = 1440 \text{ ns}$,即 $1.44 \mu\text{s}$ 。实验结果表明,相比其他容错方法的时间开销,本文的设计具有明显的优势(表 4)。

3) 容错性能对比 表 4 列出了几种可重构硬件逻辑模块容错方法的性能对比结果(系统时钟频率都设定为 25 MHz)。三模冗余的方法实现简单^[6],但硬件开销增加了约两倍,且无法定位故障;双模冗余及并发故障检测^[7]的方法能实现模块级故障定位,硬件开销相对 TMR 有所减少,但是只能处理单故障;roving STARS^[11]是较常用的在线检测方法,但在测试区域循环移动的过程中,重构时间相对较长,且需要外部处理器控制;免疫胚胎电子^[14,15]应用了免疫学的概念,专门设置了一层免疫细胞用来对故障进行分布式自诊断,但诊断前需要较长时间的学习过程。本文设计的结构采用了分布式的容错机制,可配置逻辑模块能够在线进行自测试与自修复,时间开销很小,且容错时间不随阵列规模的增大而增加。

表4 容错性能对比

容错方法	特点	时间开销	硬件开销增量/%	备注
TMR	功能模块三备份	-	217.3	硬件开销大,无法定位故障
DWC-CED	双模备份	-	172.4	只容忍单故障
roving STARS	在线、外部控制	536 ms	-	时间随规模增大而增加
免疫胚胎电子	分布式、自主控制	600 μs	> 100	需要离线状态下免疫学习
本文设计	分布式、在线、自主控制	1.44 μs	86.6	可在线自主检测及修复多位 LUT 存储单元故障

4 结束语

为了解决传统可重构硬件容错方法硬件开销大、测试时间长及外部集中式容错控制算法复杂等不足,本文设计了一种具有在线自主容错能力的可重构阵列结构,主要针对细胞功能单元内部 LUT 存储单元的固定型故障,采用了分布式容错机制,阵列由完全相同的细胞组成,容错分布在每个细胞内实现。细胞功能单元内部设置的检测模块可对逻辑模块进行循环检测,在完成检测后,检测模块对自身也进行诊断,整个测试过程自主循环执行,不影响细胞的正常工作;修复模块能对可配置逻辑模块内的 LUT 故障存储单元进行屏蔽,并利用冗余存储位进行修复,未发生故障的 LUT 存储单元仍能被正常使用。实验表明,相比其他可重构硬件容错方法,本文方法的容错能力得到了提升,能处理可配置逻辑模块的多位 LUT 存储单元故障,容错性能与目前的一些设计相比,硬件开销更小、容错速度更快、冗余硬件资源利用率更高。

参考文献:

- [1] 殷进勇,顾国昌,吴艳霞.可重构系统中实时任务容错调度算法[J].计算机应用研究,2009,26(5):1729-1732.
- [2] 王平,孙宁,李华旺,等.小卫星星载容错计算机控制系统软硬件设计[J].宇航学报,2006,27(3):412-415.
- [3] 彭和平,时晨,赵元富,等.面向空间应用的双核容错微处理器的研究与实现[J].宇航学报,2007,28(1):188-193.
- [4] UROS L, ANTON B, FRANC N. Automated SEU fault emulation using partial FPGA reconfiguration[C]//Proc of the 13th IEEE International Symposium on Design and Diagnostics of Electronic Circuits and Systems. 2010:24-27.

(上接第 2175 页)

- [5] 姚睿,王友仁,于盛林,等. 具有在线修复能力的强容错三模冗余系统设计及实验研究[J]. 电子学报,2010,27(1):177-183.
- [6] KASTENSMIDT F L, STERPONE L, CARRO L, *et al.* On the optimal design of triple modular redundancy logic for SRAM-based FPGAs [C]//Proc of Conference on Design, Automation and Test in Europe. Washington DC:IEEE Computer Society,2005:1290-1295.
- [7] LIMA F, CARRO L, REIS R. Designing fault tolerant systems into SRAM-based FPGAs[C]//Proc of the 40th Annual Design Automation Conference. New York:ACM Press, 2003:650-655.
- [8] 郝国锋,王友仁,张岩,等. 可重构硬件内建自测试与容错机制研究[J]. 仪器仪表学报,2011,32(4):856-862.
- [9] ARGYRIDES C, ZARANDI H, PRADHAN D K. Multiple SEU tolerance in LUTs of FPGAs using protected schemes[C]//Proc of European Conference on Radiation and Its Effects on Components and Systems. 2008:325-330.
- [10] KUMAR T N, CHONG C W. An automated approach for locating multiple faulty LUTs in an FPGA [J]. **Microelectronics Reliability**,

2008,48(11-12):1900-1906.

- [11] HSU C L, CHEN T H. Built-in self-test design for fault detection and fault diagnosis in SRAM-based FPGA [J]. **IEEE Trans on Instrumentation and Measurement**,2009,58(7):2300-2315.
- [12] CANHAM R, TYRRELL A M. An embryonic array with improved efficiency and fault tolerance [C]//Proc of NASA/DoD Conference on Evolvable Hardware. Washington DC:IEEE Computer Society,2003:265-272.
- [13] SZÁSZ C, CHINDRIS V, HUSI G. Embryonic systems implementation with FPGA-based artificial cell network hardware architectures [J]. **Asian Journal of Control**,2010,12(2):208-215.
- [14] ZHANG X, DRAGFFY G, PIPE A G, *et al.* Artificial innate immune system:an instant defence layer of embryonics[J]. **Artificial Immune Systems**,2004,3239:302-315.
- [15] CANHAM R, TYRRELL A M. A learning, multi-layered, hardware artificial immune system implemented upon an embryonic array [C]//Proc of the 5th International Conference on Evolvable Systems:from Biology to Hardware. Berlin:Springer-Verlag,2003:174-185.