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

姚 睿,王友仁,于盛林,陈则王
(南京航空航天大学自动化学院,江苏南京 210016)

摘 要: 为提高太空恶劣环境中电子系统的可靠性,提出了一种具有芯片级在线修复能力的强容错三模冗余(TMR)系统结构及设计方法,可在不影响系统正常工作的前提下实现故障模块的在线修复.该系统采用 TMR 结构,可实时检测定位故障模块;模块采用组件备份法设计,故障发生时可通过备件切换法快速自修复,模块中每个故障组件均可通过进化进行修复;并通过异构冗余降低 2 个以上模块同时故障的概率.以具有片内三模冗余的三阶高密度双极性(HDB3)编码器系统设计为例,对系统结构和各种容错修复机制进行了验证,结果表明系统可靠性得到很大提高.

关键词: 航天器;电子设备;容错技术;进化硬件;三模冗余;HDB3 编码器
中图分类号: TP18 **文献标识码:** A **文章编号:** 0372-2112 (2010) 01-0177-07

Design and Experiments of Enhanced Fault-Tolerant Triple-Module Redundancy Systems Capable of Online Self-Repairing

YAO Rui, WANG You-ren, YU Sheng-lin, CHEN Ze-wang
(*College of Automation and Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu* 210016, *China*)

Abstract: A new system structure and design approach of triple-module redundancy (TMR) systems-on-chip with multi-ply online self-repair mechanisms is proposed in which fault module can be repaired without affectting the system's normal operation. The system is composed of three reconfigurable redundant modules and a control processor, in which fault modules can be detected autonomously; each module is made up of subassemblies with spare parts, and can recover from fault quickly by switching to the spare parts; meanwhile each subassembly in the module can be repaired through evolution; moreover, redundant circuits with different structures are applied to avoid the synchronous arrival of fault at more than 2 modules. The design method and all the self-repair mechanisms are proof-tested by a TMR HDB3 coder system-on-chip. It is shown that the reliability of the system has been enhanced greatly.

Key words: spacecraft; electronic equipment; fault tolerant technique; evolvable hardware; triple-module redundancy; high density bipolar of order 3 (HDB3) coder

1 引言

随着科学技术的快速发展,人类探索开发太空的步伐不断加快,各种空间飞行器的结构和功能越来越复杂,且寿命与可靠性要求越来越高.航天器运行在空间恶劣环境中(如电磁干扰、极端温度),易引起星载电子设备出现故障.另外,目前电子系统正朝着结构复杂化、高度集成化的片上系统(System-On-Chip, SOC)方向发展,传统冗余备份(如系统级、模块级、芯片级备份)容错设计方法已难以适应高可靠、强生存、长寿命要求,因此必须探索新的电子系统容错机制.

超大规模现场可编程门阵列(Field Programmable Gate Array, FPGA)具有通用性好、集成度高、硬件可重构等优点,目前已在航天器上得到广泛应用[1~3].但是,空间应用 FPGA 逻辑电路可能因空间粒子辐射等原因[4]产生各种瞬时性或永久性局部故障.近年来研究人员已开始研究 FPGA 片内冗余容错技术,如 FPGA 片内 N 模冗余、主备份冗余等.但是,这些容错设计方法基本还是采用传统的器件备份式冗余方案,资源利用率不高,系统容错能力不强.如何利用 FPGA 的可重构特性构建高可靠性容错系统,以及如何实现容错系统中故障模块的在线修复是当前该领域探索研究的前沿课题.进化硬件的出现为这一问题的解决提供了一种新的思路.

进化硬件(Evolvable Hardware,简城 EHW),也称演

收稿日期:2008-05-19;修回日期:2009-09-11
基金项目:国家自然科学基金(No.60374008,90505013);航空科学基金(No.2006ZD52044,04I52068)

化硬件或仿生硬件,是一种具有自组织、自适应和自修复特性的新型智能硬件[5],它将进化算法与可重构器件有机结合在一起,以进化算法特别是遗传算法作为全局搜索的主要工具,以现场可重构器件作为评估手段和实现载体,寻求在不依赖先验知识和人工干预的情况下,通过进化来获得满足给定要求的电路和系统结构,进而使系统自动、实时地调整其内部结构,以适应内部条件(如局部故障)和外部环境的变化[6].因此,特别适合空间应用中实现系统的在轨自修复[7,8],提高星载电子设备的长期可靠性、容错性能、抗电磁波损伤的生存能力.

本文将进化硬件与传统三模冗余(Triple-module Redundancy,简称 TMR)技术相结合,提出了一种具有芯片级在线修复能力的强容错 TMR 系统结构及设计方法,并以具有片内三模冗余的 HDB3 编码器设计为例进行验证.结果表明,该容错系统不仅具有强容错能力和高可靠性,而且可以在不影响系统正常工作前提下实现故障模块的在线修复.

2 基于进化硬件的自修复 TMR 系统结构及各种容错自修复机制

本文提出的自修复 TMR 系统的基本思想是采用进化硬件使 TMR 系统具有可修复性,同时采用多重容错修复机制进一步提高其可靠性.

2.1 自修复三模冗余系统结构

三模冗余是一种应用比较广泛的冗余容错方案,基本原理为:使用 3 个完全相同的模块执行相同任务,由表决器以三中取二原则对其输出进行表决,以确定整个系统的输出.TMR 系统可以容忍 1 个模块发生故障而不影响整个系统运行的正确性.

由于传统 TMR 系统各模块均不具备修复功能,因此本文利用进化硬件思想来设计具有在线修复能力的 TMR 系统,其结构框图如图 1 所示.

图 1 中,系统总体采用 TMR 结构,其中 A、B、C 均为可重构、可修复模块.为进行修复控制,增加了 1 个进化控制处理器(EHWCPU)和 1 个差错检测器.系统正常工作过程中,差错检测器检测到某一模块出现故障时,提示 EHWCPU 对其进行修复,EHWCPU 根据差错检测信息,控制双向开关屏蔽该模块,同时对其进行修复,修复后再将该模块重新插入容错系统,使系统恢复至初始容错能力.

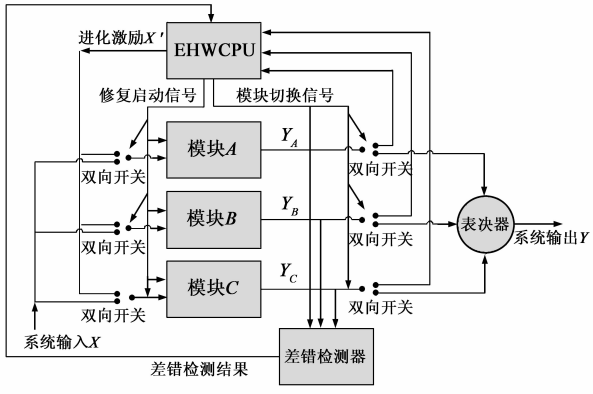


图1 自修复TMR系统结构

2.2 差错检测器设计

设 A、B、C 三个模块的输出分别为 Y_a、Y_b、Y_c,每个输出有 m 位,即 $Y_{am-1} \sim Y_{a0}$、$Y_{bm-1} \sim Y_{b0}$、$Y_{cm-1} \sim Y_{c0}$.检测器输出为 $Y: Y_{m-1} \sim Y_0$. F_a、F_b、F_c 分别为模块 A、B、C 的故障信号,F 为总故障信号,则

$$Y_i = Y_{ai}Y_{bi} + Y_{bi}Y_{ci} + Y_{ci}Y_{ai} \tag{1}$$

$$F_a = \sum_{i=0}^{m-1} \left(Y_{ai}\overline{Y}_{bi}\overline{Y}_{ci} + \overline{Y}_{ai}Y_{bi}Y_{ci} \right) \tag{2}$$

$$F_b = \sum_{i=0}^{m-1} \left(Y_{bi}\overline{Y}_{ai}\overline{Y}_{ci} + \overline{Y}_{bi}Y_{ai}Y_{ci} \right) \tag{3}$$

$$F_c = \sum_{i=0}^{m-1} \left(Y_{ci}\overline{Y}_{bi}\overline{Y}_{ai} + \overline{Y}_{ci}Y_{bi}Y_{ai} \right) \tag{4}$$

$$F = F_a + F_b + F_c \tag{5}$$

式(1)~(5)中,所有运算均为逻辑运算,且 F、F_a、F_b 和 F_c 为 1 表示出现故障.

2.3 提高系统可靠性的多种容错自修复机制

(1)模块的二重容错机制与修复方法

利用进化硬件可以实现故障模块的在线自修复,但是直接进化修复的电路规模普遍较小.随着电路规模的扩展,进化修复时间呈指数上升.为缩短修复时间,采用二重容错机制来设计模块电路.

模块电路由组件构成,每个组件均有自己的备件,以图 1 中模块 A 为例,其结构框图如图 2 所示.

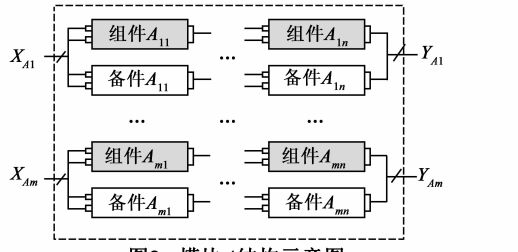


图2 模块A结构示意图

各模块均采用二重容错机制:第一重为备份组件切换修复,即当某个模块出现故障,如模块 A 的组件 A_{ij} 出现故障,可以通过进化算法,用备件 A_{ij} 替换,使模块 A 恢复正常,并把组件 A_{ij} 标记为故障组件;第二重为故障组件的进化修复,即 EHWCPU 可在空闲时对故障组件 A_{ij} 进行修复.采用二重容错机制不仅可大大缩短修复时间、提高修复成功率,而且可以提高各模块所能容忍故障的次数.

故障模块在线修复流程为:系统正常工作过程中EHWCPU对 F 进行监控,若 F 为1,表明系统出现故障,并根据 F_a、F_b 和 F_c 的值判断哪个模块故障,然后存储故障模块编号,启动故障模块的第一重修复,即组件切换修复.故障模块被修复成功后,EHWCPU一方面继续监控 F 状态,另一方面启动第二重修复,即修复组件切换过程中被标记的故障组件.

(2)异构冗余容错

传统三模冗余系统由于各模块结构相同,在特定环境中易出现相关错误而导致多个模块同时出错,因此为避免2个以上模块同时出错而引起系统工作异常,在自修复TMR系统中,还采用异构冗余设计方法以进一步提高系统的可靠性.

异构三模冗余系统设计基本思想:利用进化算法进化出三个能够实现需求功能的电路(未进行非相似度评价);然后每进化成功一个电路,进行一次非相似度评价,保留三个非相似度最大的电路并应用,当三个异构电路有一个出错时,系统即对故障电路进行屏蔽,待修复成功后再重新投入运行.

3　系统可靠性分析

3.1　自修复 TMR 系统可靠性模型

假设图1所示的自修复TMR系统具有3种状态:正常状态(即 A、B、C,3个模块均正常工作)、修复状态(有1个模块故障,且正在被修复)和失效状态(有2个以上模块故障),并且假定在相当小时间 Δt 内,2个或2个以上模块同时进行状态转移的概率为 Δt 的高阶无穷小,则该系统的状态空间马尔科夫模型如图3所示.

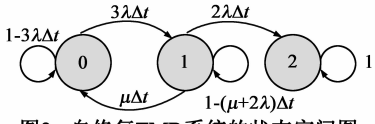


图3　自修复TMR系统的状态空间图

图3中,状态0为正常状态,状态1为修复状态,状态2为失效状态.初始情况下系统处于状态0,当系统开始工作后,有可能向状态1转移.假设每个模块的失效率为 λ,模块 A、B、C 中任意一个(记为 M_f)发生故障时,系统均进入状态1,因此在 Δt 时间内,由状态0向状态1转移的概率为 $3\lambda\Delta t$,则停留在状态0的概率是 $1-3\lambda\Delta t$.

在状态1下,若 M_f 再次发生故障,则仍处于状态1;若除 M_f 外的另一模块发生故障,则系统失效,进入状态2,因此在 Δt 时间内由状态1向状态2转移的概率为 $2\lambda\Delta t$;同时由于模块 M_f 正在被修复,若修复成功则进入状态0,假设 M_f 的修复率为 μ,则在 Δt 时间内由状态1向状态0转移的概率为 $\mu\Delta t$;同时可得系统仍停留在状态1的概率为 $1-(\mu+2\lambda)\Delta t$.

状态2为2个或2个以上模块发生故障的失效状态,假定这种情况下不进行修复,则系统进入状态2后仍停留在该状态的概率为1.

3.2　自修复 TMR 系统可靠性评价的理论依据

令 $p_0(t)$和 $p_0(t+\Delta t)$、$p_1(t)$和 $p_1(t+\Delta t)$、$p_2(t)$和 $p_2(t+\Delta t)$分别代表系统在 t 时刻和 $t+\Delta t$ 时刻处于状态0、状态1、状态2的概率.据图3可列出自修复TMR系统状态转移矩阵为

$$\boldsymbol{T}=\begin{bmatrix}1-3\lambda & \mu & 0\\ 3\lambda & 1-(\mu+2\lambda) & 0\\ 0 & 2\lambda & 1\end{bmatrix} \tag{6}$$

状态方程的系数矩阵为

$$\boldsymbol{A}=\boldsymbol{T}-\boldsymbol{I}=\begin{bmatrix}-3\lambda & \mu & 0\\ 3\lambda & -(\mu+2\lambda) & 0\\ 0 & 2\lambda & 0\end{bmatrix} \tag{7}$$

状态方程为

$$\begin{bmatrix}\dot{p}_0(t)\\ \dot{p}_1(t)\\ \dot{p}_2(t)\end{bmatrix}=\begin{bmatrix}-3\lambda & \mu & 0\\ 3\lambda & -(\mu+2\lambda) & 0\\ 0 & 2\lambda & 0\end{bmatrix}\begin{bmatrix}p_0(t)\\ p_1(t)\\ p_2(t)\end{bmatrix} \tag{8}$$

已知初始条件 $p_0(t)=1, p_1(t)=0, p_2(t)=0$.

将式(8)展开后的状态方程为

$$\begin{cases}\dot{p}_0(t)=-3\lambda p_0(t)+\mu p_1(t)\\ \dot{p}_1(t)=3\lambda p_0(t)-(\mu+2\lambda)p_1(t)\\ \dot{p}_2(t)=2\lambda p_1(t)\end{cases} \tag{9}$$

对式(9)进行拉氏变换,得线性方程组

$$\begin{cases}sp_0(s)-p_0(0)=-3\lambda p_0(s)+\mu p_1(s)\\ sp_1(s)-p_1(0)=3\lambda p_0(s)-(\mu+2\lambda)p_1(s)\\ sp_2(s)-p_2(0)=2\lambda p_1(s)\end{cases} \tag{10}$$

由式(10)可解出

$$\begin{cases}p_0(s)=\dfrac{3\lambda}{s^2+(\mu+\lambda)s+6\lambda^2}\\ p_1(s)=\dfrac{s+\mu+2\lambda}{s^2+(\mu+\lambda)s+6\lambda^2}\\ p_2(s)=\dfrac{6\lambda^2}{s[s^2+(\mu+\lambda)s+6\lambda^2]}\end{cases} \tag{11}$$

对式(11)进行拉氏反变换,得系统的可靠性为:

$$\begin{aligned}R(t)&=p_0(t)+p_1(t)\\ &=\frac{\mu+5\lambda+\sqrt{\lambda^2+\mu^2+10\mu\lambda}}{2\sqrt{\lambda^2+\mu^2+10\mu\lambda}}e^{\frac{\mu+5\lambda-\sqrt{\lambda^2+\mu^2+10\mu\lambda}}{2}t}\\ &\quad-\frac{\mu+5\lambda-\sqrt{\lambda^2+\mu^2+10\mu\lambda}}{2\sqrt{\lambda^2+\mu^2+10\mu\lambda}}e^{\frac{\mu+5\lambda+\sqrt{\lambda^2+\mu^2+10\mu\lambda}}{2}t}\end{aligned} \tag{12}$$

3.3　自修复 TMR 系统可靠性分析及与传统 TMR 系统的对比

已知传统TMR系统的可靠性为

$$R_{TMR}(t)=-2e^{-3\lambda t}+3e^{-2\lambda t} \tag{13}$$

为了分析方便,设 $\mu = k\lambda$,由式(12)可得自修复 TMR 系统的可靠性为

$$R(t) = \frac{k+5+\sqrt{k^2+10k+1}}{2\sqrt{k^2+10k+1}} e^{\frac{k+5-\sqrt{k^2+10k+1}}{2}\lambda t} - \frac{k+5-\sqrt{k^2+10k+1}}{2\sqrt{k^2+10k+1}} e^{\frac{k+5+\sqrt{k^2+10k+1}}{2}\lambda t} \tag{14}$$

为便于比较,分别取 $k = 10, 100, 1000, 10000, 100000$,不同情况下自修复 TMR 系统及传统 TMR 系统的可靠性随 λt 变化曲线如图 4 所示(图中 ETMR 表示自修复 TMR).

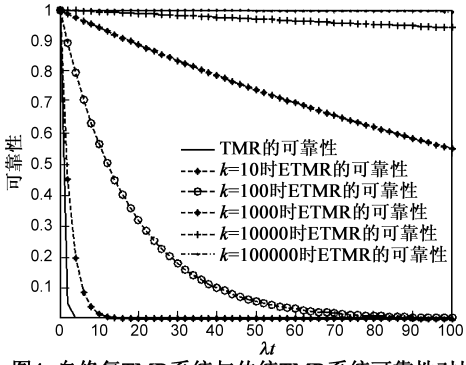


图4 自修复TMR系统与传统TMR系统可靠性对比

由图 4 可知,当故障率 λ 一定时,随着使用时间的增加(即 λt 的增大),系统可靠性将逐渐降低.而且,修复率 μ 越低(即 k 值越小),可靠性下降越快.

当 λ 和 t 一定时,自修复 TMR 系统的可靠性普遍高于传统 TMR 系统.而且修复率 μ 越高(即 k 值越大),可靠性也越高.如,当 $\lambda t = 1$ 时传统 TMR 系统可靠性约为 0.3;而此时自修复 TMR 系统可靠性:$k = 10$ 时约为 0.68,$k = 100$ 时约为 0.94,$k > 1000$ 时可达 0.99 以上.特别是,当 $k = 100000$,$\lambda t = 100$ 时自修复 TMR 系统可靠性仍能保持 0.99 以上.

当 λ 值一定时,k 的大小取决于修复时间.修复时间越长,μ 越小,k 也越小.如,设系统每 2 年发生一次故障,即 $\lambda = 1/(2\times365\times24)$次/h 时:若修复时间为 2h,则 k 为 8760;若修复时间为 20h,则 k 降为 876;若修复时间为 200h,则 k 仅为 87.6.

因此对于自修复 TMR 系统,在提高生存寿命的同时提高其可靠度,必须尽可能地提高修复率 μ(即降低修复时间),并减小故障率 λ.

4 试验结果与分析

下面以具有片内三模冗余的 HDB3 编码器系统设计为例,验证系统设计方法及各种容错修复机制.

4.1 在线进化修复平台

EHW 在线进化修复实验平台如图 5 所示.平台采用 CPU 和 FPGA 的协处理环境.其中进化控制处理器由主处理器 CPU 执行,三个模块及差错检测器在 Virtex FPGA 上实现.Virtex FPGA 具有局部动态重构能力,即仅重构 FPGA 上部分资源而其他部分仍正常工作.因此可以在不影响系统正常运行情况下,仅对故障模块所在区域进行修复.局部动态重构通过基于 Java 的应用程序接口 JBits 进行操作.

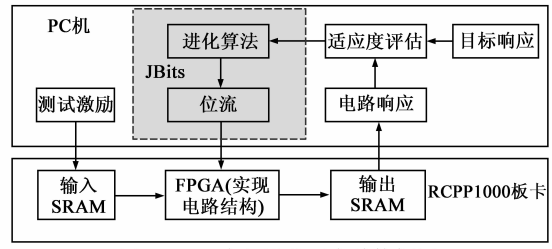


图5 EHW内部进化平台结构框图

4.2 容错系统设计实例及二重容错修复实验

HDB3 码(三阶高密度双极性码)是基带电信设备之间进行基带传输的主要码型之一,其主要特点是易于提取时钟、不受直流特性影响、具有自检能力、连 0 串不超过 3 个等.

4.2.1 HDB3 编码器模型

HDB3 码是 AMI(Alternate Mark Inversion)码的改进型.AMI 码是用交替极性的脉冲表示码元 1,用无脉冲表示码元 0.为了防止电路长时间出现无脉冲状态,HDB3 码的编码规则为:

(1)将消息码变换成 AMI 码.

(2)插 V:若码流中连 0 码个数不超过 3 个,则保持 AMI 码形式不变;若出现 4 个或 4 个以上连 0 时,则将第 4 个 0 变为与前一非 0 符号同极性的符号,用 V 表示.

(3)插 B:若相邻 V 间非 0 符号的个数为偶数,则将当前 V 的前一非 0 符号后的第一个 0 变为 B,且 B 的极性与该非 0 符号相反.

在 FPGA 上实现 HDB3 编码器时,为简化电路结构,采用图 6 所示模型,即直接在消息码基础上,首先依据 HDB3 编码原则进行插 V 和插 B 操作,最后完成单极性信号到双极性信号转换.

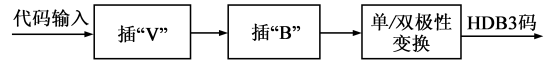


图6 HDB3编码器模型

4.2.2 三个冗余模块的区域规划与染色体编码方法

三个冗余模块和差错检测器在同一片 FPGA 上实现,差错检测模块共使用 12 个查找表(Look-Up Table,简称 LUT).每个模块限定的进化区域为 14×26 个 LUT 阵列(包括插 V、插 B 和极性转换三个部分的组件及其备件),三个模块共使用 42×26 个 LUT.

由于在组件切换修复实验中,每个模块各组件和备件的位置和功能固定,因此只需对组件间连线进行编码,采用分段编码方法,将三个模块编码为一条染色体,其中每个模块占用 8 位,染色体长度共 24 位.

4.2.3　故障模块的组件切换修复实验

采用模拟故障注入法,即通过系统 CPU 发出命令,强行向 FPGA 上进化区域某个 LUT 注入固定于 0 或 1 的故障(直接将该 LUT 的输出设置为 0 或 1),直至某个模块出现故障($F=1$),系统会自动检测到故障模块并启动修复程序.

向工作正常的三模冗余 HDB3 编码器电路的一个工作组件随机注入 stuck-at 故障,使用备件切换法修复故障.共做 30 次实验,在 100 代内全部修复成功.每次进化时间如表 1 所示.

表 1　HDB3 编码器的在线修复时间

次序	1	2	3	4	5	6	7	8	9	10
时间(s)	15	11	11	20	11	21	20	20	11	21
次序	11	12	13	14	15	16	17	18	19	20
时间(s)	30	11	20	11	20	11	20	11	11	10
次序	21	22	23	24	25	26	27	28	29	30
时间(s)	11	10	10	10	10	10	20	10	10	10

表 1 中,平均修复时间(Mean Time To Repair 简称 MTTR)为 14.2s.可见使用组件切换法可以快速在线修复故障,使自修复 TMR 系统的三个模块均正常工作.

4.2.4　组件的进化修复实验

在 4.2.3 小节中,系统修复过程并未对故障组件进行修复,因此各组件仅能容忍 1 次故障.为进一步提高容错次数,在 EHWCPU 空闲时对故障组件进行修复.

下面以 HDB3 编码器中极性转换组件的进化修复为例,验证组件的进化修复能力.

经插 V、插 B 后,1、V、B 和 0 已分别用双相码 01、11、10 和 00 表示.极性转换电路作用是将 1 和 B 交替用 +1、-1(01、11)表示,V 变成与前一非 0 符号相同的符号,因此需要一个二进制计数器 f 来计 1 和 B 的个数,并根据 f 的奇偶进行极性转换.设该电路输入为 b_1b_0,输出为 t_1t_0,则真值表如表 2 所示.

表 2　极性转换模块的真值表

输入			输出		输入			输出	
f	b_1	b_0	t_1	t_0	f	b_1	b_0	t_1	t_0
0	0	0	0	0	1	0	0	0	0
0	0	1	1	1	1	0	1	0	1
0	1	0	1	1	1	1	0	0	1
0	1	1	0	1	1	1	1	1	1

采用 4×6 进化区域,将 LUT 和 D 触发器作为基本进化单元统一编码,染色体长度为 36 位,其中 26 位表示连线,10 位表示 LUT 功能.

图 7(a)为进化成功未注入故障的极性转换电路,电路中共用 5 个逻辑门和 1 个 D 触发器.下面每次向其中 1 个逻辑门注入固定于 0 的故障,分别进行进化修复实验.

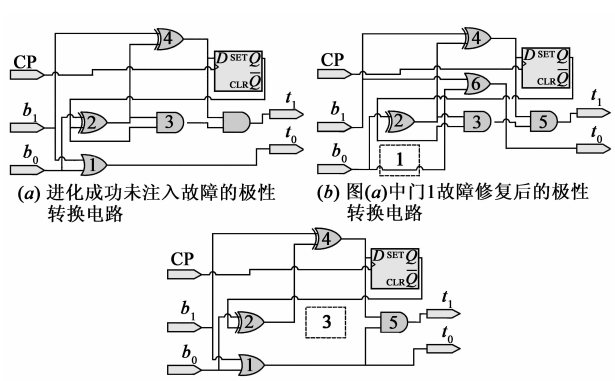
(1)向门 1 注入固定于 0 的故障

共做了 20 次实验,在 20000 代内全部修复成功,且由于修复是在原染色体基础上进行,每次修复得到的电路结构完全相同,如图 7(b)所示.

图 7(b)中,修复后电路绕过了故障门 1(图中虚线框所示位置),用门 6 取代原来门 1 所实现的功能,每次修复所需时间如表 3 所示.

表 3　图 7(a)中门 1 故障的在线修复时间

次序	1	2	3	4	5	6	7	8	9	10
时间(min)	18	75	1	4	22	24	282	56	84	36
次序	11	12	13	14	15	16	17	18	19	20
时间(min)	17	92	11	8	23	9	37	42	6	4



(a) 进化成功未注入故障的极性转换电路　(b) 图(a)中门1故障修复后的极性转换电路

(c) 图(a)中门3故障修复后的极性转换电路

图7　无故障及门1、3故障修复后的极性转换电路

表 3 中,MTTR 为 42.6min,是组件切换法的 180 倍.HDB3 编码器其他两个组件的电路更复杂,因此其 MTTR 将更长,若将所有模块作为整体进行进化修复,其 MTTR 将呈几何指数增长,这也是采用组件备份法设计该系统的重要原因之一.

(2)向门 3 注入固定于 0 的故障

共做了 20 次实验,在 20000 代内全部修复成功,每次修复所得电路结构完全相同,如图 7(c)所示.

图 7(c)中,修复后电路绕过了故障门 3(图中虚线框所示位置),用门 4 输出替代原门 3 输出,实现同样的功能,每次修复所需时间如表 4 所示.

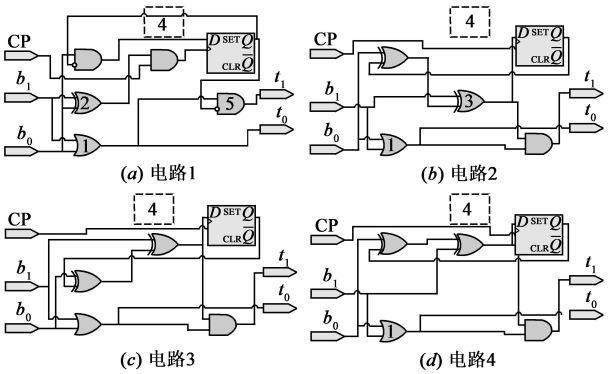
表 4　图 7(a)中门 3 故障的在线修复时间

次序	1	2	3	4	5	6	7	8	9	10
时间(s)	65	169	24	24	64	24	72	40	40	136
次序	11	12	13	14	15	16	17	18	19	20
时间(s)	16	96	43	24	80	32	24	24	72	112

表 4 中,MTTR 为 59s.

(3)向门 4 注入固定于 0 的故障

共做了 10 次实验,在 20000 代内全部修复成功,而每次修复所得电路结构不完全相同,其中 4 种典型电路结构如图 8 所示.



(*a*) 电路1 (*b*) 电路2

(*c*) 电路3 (*d*) 电路4

图8 图7(*a*)中门4故障修复后的极性转换电路

图 8 中,修复后电路均绕过了故障门 4(图中虚线所示位置),每次修复所需时间如表 5 所示.

表 5 图 7(*a*)中门 4 故障的在线修复时间

次序	1	2	3	4	5	6	7	8	9	10
时间(min)	220	615	36	59	131	536	91	362	478	93

表 5 中,MTTR 为 262.1min,远远大于门 1 和 3 故障所需的 MTTR,原因是它在图 7(*a*)中的位置比较关键,输出同时又作为触发器输入和门 5 的一个输入.

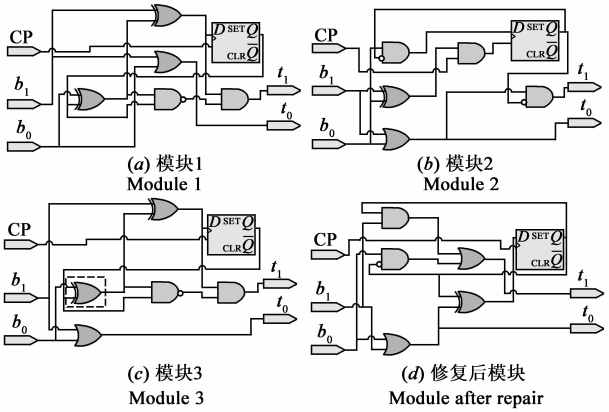
门 2 和门 5 的故障可类似修复,这里不再赘述.

由本节实验结果可知,通过进化可以实现故障组件的在线修复,修复所需 MTTR 与故障资源在电路中地位有关,即越关键的资源修复所需 MTTR 越大.

4.3 异构冗余容错实验

以实现异构三模冗余极性转换组件电路为例,设计出的三模异构电路典型结构如图 9(*a*)~(*c*)所示.

由图 9 可知,模块 1、2、3 中所使用的 5 种逻辑门(*F*1&*F*2, *F*1&(~*F*2), *F*1^*F*2, ~(*F*1&*F*2), *F*1|*F*2)的个数以及位置不同,即存在异构.



(*a*) 模块1 Module 1 (*b*) 模块2 Module 2

(*c*) 模块3 Module 3 (*d*) 修复后模块 Module after repair

图9 设计出的三个异构模块结构及修复模块结构图

系统正常工作时,强制图 9(*c*)中虚线框所示的门故障,通过进化修复可得到与模块 1、2 异构的修复模块,如图 9(*d*)所示.修复后的模块绕过了故障位置.

可见采用 EHW 不仅可以自动设计具有不同结构的电路,而且在系统故障时还可以通过 EHW 获得异构的修复电路.

4.4 各种容错修复机制对系统可靠性的贡献及资源代价分析

4.4.1 各种容错修复机制对系统可靠性的贡献

使用组件切换法可以快速修复系统故障,大大提高系统可靠性.例如 4.2.3 中 HDB3 编码器系统的 MTTR 为 14.2s,修复率 μ 约为 253.5 次/h.假设单个模块(不包含备件)的平均故障间隔时间(Mean Time Between Failure,简称 MTBF)为每年 1 次,则其失效率 $\lambda' = 1/(365\times24)$次/h,则包含备件的单个模块故障率 $\lambda = 2/(365\times24)$次/h.因此 $k = \mu/\lambda \approx 1110330$,由式(15)可得该系统使用 5 年后的可靠性仍约为 1.0000;同理可得在不同的 MTBF 下自修复 TMR 系统与传统 TMR 系统在使用 5 年、20 年、100 年之后的可靠性分别如表 6~8 所示.

表 6 不同 MTBF 下 5 年后系统的可靠性

MTBF	1 年	6 个月	3 个月	2 个月	1 个月
ETMR	1.0000	1.0000	0.9999	0.9998	0.9997
TMR	0.0010	0.0000	0.0000	0.0000	0.0000

表 7 不同 MTBF 下 20 年后系统的可靠性

MTBF	1 年	6 个月	3 个月	2 个月	1 个月
ETMR	0.9999	0.9998	0.9996	0.9992	0.9988
TMR	0.0000	0.0000	0.0000	0.0000	0.0000

表 8 不同 MTBF 下 100 年后系统的可靠性

MTBF	1 年	6 个月	3 个月	2 个月	1 个月
ETMR	0.9995	0.9989	0.9979	0.9957	0.9936
TMR	0.0000	0.0000	0.0000	0.0000	0.0000

由表 6~8 可知,MTBF 为 1 年时:自修复 TMR 系统工作 5 年后可靠性为传统 TMR 系统的 1000 倍;工作 100 年后可靠性仍可达 0.9989,而传统 TMR 系统的可靠性近似为 0.即使 MTBF 为 1 个月的情况下,自修复 TMR 系统工作 100 年后仍具有 0.99 以上的可靠性.这种特性在要求长寿命和高可靠性的空间应用中特别有用.

另外,使用进化修复方法对故障组件进行修复,在 FPGA 可用资源允许情况下,理论上可以使系统容忍任意多次故障.

采用异构容错设计方法,可以降低多个模块同时出现故障的概率,从而进一步提高系统的可靠性.

4.4.2 各种容错修复机制的资源代价分析

由于自修复 TMR 系统各模块的每个组件均有备件,因此在不考虑组件进化修复所需资源冗余,其资源代价为传统 TMR 系统 2 倍.表面上看,这是本文所设计系统的缺点,但是考虑到集成电路和 SOC 系统设计的技术现状和发展趋势,自修复 TMR 系统的优势还是远远大于其缺点.一方面,半导体技术的发展使得集成电路的集成度一直按 Moore 定律不断提高,但设计生产率的差距却在不断加大;另一方面,系统复杂性的提高使得可靠性随之下降,利用资源冗余实现设计效率和性能的提高无疑是片上系统设计的一个发展方向.FPGA 实现的片上系统往往有相当数量的资源冗余,自修复 TMR 系统虽然资源代价较高,但是可提高对 FPGA 内部资源的利用率,而且用这种相对少的资源代价换来的是航天器在外太空等恶劣环境下长期使用的稳定性、可靠性的极大提高,为传统 TMR 系统所无法企及.

在自修复 TMR 系统中,若对每个组件进行修复,资源代价会相应提高,但是 EHW 可通过优化电路结构以减少资源代价;同时,故障组件的修复也可利用 FPGA 内部空闲区域进行,而不必为每个组件提供一定比例的冗余资源,从而减少资源代价.

5 结论

本文提出了一种基于进化硬件与异构冗余技术的容错系统设计方法,采用多种容错机制提高系统的可靠性和容错能力:

(1)进化硬件的引入使传统 TMR 系统具有了芯片级在线自修复功能;

(2)三模冗余技术和组件切换法能实现并发差错检测和实时故障容错;

(3)故障组件的在线进化修复提高了系统可以容忍的故障次数,保证了设备的长期可靠性和可用性;

(4)采用异构冗余设计方法降低共模干扰情况下多个模块同时出现故障的概率.

因此,基于进化硬件的电子系统具有长寿命、高可靠性和强容错能力,这种容错系统设计思想值得进一步研究探索,从而为空间恶劣环境条件下高可靠复杂电子系统的设计提供有效方法.

参考文献:

[1] 杜文志,谭维炽.航天器中 FPGA 在系统局部重构设计研究[J].中国空间科学技术,2005,25(5):10 – 16.
Du Wen-zhi. Research on FPGA fault-tolerant method by ISP partial reconstruction[J]. Chinese Sace Science and Technology,2005,25(5):10 – 16.(in Chinese)

[2] Vladimirova T, Wu X F. On-board partial run-time reconfiguration for pico-satellite constellations[A]. Proc. of the 1st Conference on Adaptive Hardware and Systems (AHS'2006)[C]. Istanbul, Turke, 2006. 262 – 269.

[3] 陈雪芹,张迎春,耿云海,等.基于 IMM/EA 的卫星姿态控制系统重构容错控制[J].系统工程与电子技术,2007,29(5):774 – 777.
Chen Xue-qin, Zhang Ying-chun, Gen Yun-hai, et al. IMM/EA-based on-orbit reconfigurable rault-tolerant control for satellite attitude control system[J]. Systems Engineering and Elect ronics, 2007, 29(5):774 – 777.(in Chinese)

[4] Fernanda Lima, Luigi Carro, Ricardo Reis. Designing fault tolerant systems Into SRAM2 based FPGAs[A]. 40th Design Automation Conference[C]. Anaheim, CA, 2002. 650 – 655.

[5] 王友仁,姚睿,朱开阳,等.仿生硬件理论与技术的研究现状与发展趋势分析[J].中国科学基金,2004(5):273 – 277.
Wang You-ren, Yao Rui, Zhu Kai-yang, et al. The present sate and future trends in bio-inspired hardware research[J]. Bulletin of National Natural Science Foundation of China, 2004(5):273 – 277.(in Chinese)

[6] Yao X, Hugichi T. Promises and callenges of evolvable hardware[J]. IEEE Trans on Systems Man and Cybernetics-Part C: Applications and Reviews, 1999, 29(1):87 – 97.

[7] James M H. Fault-tolerant sensor systems using evolvable hardware[J]. IEEE transactions on instrumentation and measurement, 2006, 55(3):846 – 853.

[8] Gregory V L, Jason D L. Evolutionary based techniques for fault tolerant field programmable gate arrays[A]. Proc. of 2nd IEEE International Conference on Space Mission Challenges for Information Technology[C]. Pasadena, California, USA: IEEE, 2006. 553 – 560.

作者简介:

姚　睿　女,1974 年 9 月生于河南邓州,博士,副教授.主要研究兴趣为进化硬件理论与技术,计算机测控系统,智能电路,认知无线电技术.　E-mail: yaorui@nuaa.edu.cn

王友仁　男,1963 年出生,博士,教授.主要研究领域为仿生硬件理论与技术、电路测试与故障诊断、计算机测控系统.

于盛林　男,1941 年出生,教授.主要研究领域为测试信号分析与处理,在线检测与故障诊断.