doi:10.11887/j.cn.201904013 http://journal.nudt.edu.cn

# 使用 FPGA 设计高可靠 SpaceWire 路由器*

姚 睿[1],王梅群[1],吴 军[2],胡 杰[1],李明莉[1]
(1. 南京航空航天大学 自动化学院,江苏 南京 211106;2. 北京控制工程研究所,北京 100190)

**摘 要**:为了提高系统的可靠性和可用性,提出一种基于静态随机存取存储器(Static Random-Access Memory,SRAM)型现场可编程门阵列(Field-Programmable Gate Array, FPGA)的 SpaceWire 路由器设计方法。路由器通过系统级三模冗余技术加固,采用基于位流重定位的动态部分刷新技术修复系统中发生的软故障,并提出一种基于工作输入和健康现态的实时状态同步方法,以确保故障模块修复后的状态与其他模块同步。因此,该系统能够进行错误掩蔽和自我修复。在 Xilinx Virtex-5 FPGA 开发板 ML507 上对所提出的路由器系统结构和设计方法进行实现和验证。实验结果表明,路由器的可靠性和可用性显著增加,且系统的实时性很好,能保证路由器在整个工作过程中提供正常服务而不会引起系统功能中断或延迟;位流重定位技术的采用将所需存储空间减少三分之二,同时也降低了原始位流本身故障的可能性。

**关键词**:SpaceWire 路由器;现场可编程门阵列;三模冗余;自修复;状态同步;位流重定位
**中图分类号**:TP274 **文献标志码**:A **文章编号**:1001-2486(2019)04-086-08

# Design of highly reliable SpaceWire routers based on FPGA

*YAO Rui*[1], *WANG Meiqun*[1], *WU Jun*[2], *HU Jie*[1], *LI Mingli*[1]
(1. College of Automation and Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China;
2. Beijing Institute of Control Engineering, Beijing 100190, China)

**Abstract**: A system design method for SpaceWire router based on SRAM FPGA(field-programmable gate array) was proposed to improve its reliability and availability. The router was hardened by system level triple modular redundancy; and the dynamic partial scrubbing technique based on bitstream relocation was adopted to repair the faulty module in case of soft errors. Meanwhile, a real-time state synchronization approach based on present input and healthy state was introduced to synchronize the repaired module's state with the other modules' after scrubbing. Hence the router is capable of error masking and self-healing. The proposed design method was implemented and verified on the Xilinx Virtex-5 FPGA develop kit ML507. Experimental results show that the reliability and availability of the router are increased significantly. And the router's real-time performance is satisfactory, such that it can offer normal service during the entire work process, without any interruption or delay in system functionality. Meanwhile, the amount of memory required is reduced to one-third of the original amount as a result of adopting the bitstream relocation technique, and the failure probability of the original bitstream is also greatly reduced.

**Keywords**: SpaceWire router; field-programmable gate array; triple modular redundancy; self-repair; state synchronization; bitstream relocation

空天任务中,主控计算机与外围设备或大容量存储器之间需要进行大量的数据交换,一些重要星上控制数据也需要通过一体化的总线传输,这对设备之间的信息交换和传输提出了较高要求。SpaceWire(SpW)是由欧空局推出的一种面向航天应用的专用通信标准,可提供高速(2~200 Mbit/s)、双向、全双工的数据链路[1]。通过 SpW 链路连接路由器和节点,可以灵活组建不同规模和拓扑结构的数据传输网络,提升航天电子系统的可扩展性、稳定性和规范化,降低系统组建成本。目前,很多空间任务已将 SpW 作为通信标准,一些空间仪器中也增加了 SpW 接口[2-3]。SpW 网络中,各节点通过路由器互联,实现数据链路的复用和数据交互。因此,SpW 路由器的可靠性在信息传输过程中十分重要。

当前商用 SpW 路由器通常由专用集成电路实现,如欧空局开发的 SpW-10X[4]、Aeroflex Gaisler 与 IMEC 联合开发的 GR718[5]。这些抗辐射路由器价格昂贵,且无法根据应用需求进行修改。现场可编程门阵列(Field-Programmable Gate

* 收稿日期:2018-04-10
基金项目:国家自然科学基金资助项目(61402226);国家部委基金资助项目(D020103)
作者简介:姚睿(1974—),女,河南邓州人,副教授,博士,硕士生导师,E-mail:yaorui@nuaa.edu.cn

Arrays, FPGAs),特别是静态随机存取存储器(Static Random-Access Memory, SRAM)型 FPGA,具有灵活性高、成本低、开发周期短等特点和硬件功能在轨升级的能力,越来越受空间工业的青睐[6]。然而,空天应用中 SRAM 型 FPGA 的逻辑电路可能发生各种故障,其数据通路和配置存储器均可能因辐射效应产生单粒子翻转(Single Event Upset, SEU),对系统可靠性造成严重威胁。因此,迫切需要采取措施提高基于 FPGA 的 SpW 路由器的可靠性。

目前,SpW 路由器的可靠性主要通过协议层的容错措施来保证,且针对专用集成电路实现的路由器 IP 核多采用硬件描述语言(Hardware Description Language, HDL)开发。然而,文献[7]中对路由器 IP 核的模拟 SEU 故障注入实验结果表明,SpW 协议不能确保 100% 的故障检测。因此,研究人员开始研究增强路由器 IP 故障检测和容错能力的措施,如:文献[8]通过故障注入实验评估路由器 IP 核中对 SEU 最敏感的组件,并对关键组件进行寄存器传输级(Register Transfer Leve, RTL)加固,如采用独热状态机、奇偶校验等;文献[9]采用具有错误检测和纠正功能的定制先入先出(First Input First Output, FIFO)存储电路来加固路由器的发送/接收 FIFO 单元;文献[10]采用冗余容错技术对 SpW 收发器进行加固。

上述技术仅对路由器部分组件进行加固,要确保基于 FPGA 的 SpW 路由器的可靠性,需要采取系统级保护措施。三模冗余(Triple Modular Redundancy, TMR)是一种应用广泛的经典容错方法,可屏蔽系统中一个模块发生的故障,但其无法纠正 SRAM 型 FPGA 的主要故障——配置存储器中 SEU 故障。虽然该故障可通过配置刷新(即定期刷新配置内存内容)进行纠正[11],但刷新过程将暂时中断系统功能。动态部分刷新(Dynamic Partial Scrubbing, DPS)可仅刷新配置存储器的一部分,而保持其余部分不变。因此,本文将系统级 TMR 和 DPS 技术结合,设计基于 FPGA 的 SpW 路由器,使其具有错误屏蔽和修复能力;并采用位流重定位技术减少位流文件的存储空间。

此外,DPS 虽可修复故障模块的功能,但无法确保修复后该模块中时序电路的状态与其他二个健康模块一致,因而无法使系统恢复至初始容错能力,可能因二次故障导致系统崩溃。实现 TMR 系统状态同步最简单的方法是重启整个系统。为避免重启系统,可采用回卷恢复[12]或前滚恢复[13]技术。故障发生时,回卷恢复将系统状态恢复至前一个无故障检测点备份的状态,前滚恢复直接将当前无故障冗余模块的状态拷贝至故障模块。虽然前滚恢复的性能优于回卷恢复,但其必须确保状态拷贝过程中无故障模块停止操作,否则后者状态可能发生新的变化。为此,文献[14]将前滚恢复和检测点技术结合,提出了一种改进的前滚恢复方法。在检测点发现故障时,首先缓存系统输入,并将无故障模块的状态拷贝至故障模块;恢复完毕再对缓存的输入进行处理。该方法通过使三个模块停止接收新的输入实现故障模块与健康模块的状态同步,但需对当前输入进行缓存,且状态恢复过程中系统并未真正对外提供服务,造成系统功能延迟。而且,该方法需要系统中所有寄存器的详细信息。为此,文献[15]和文献[16]分别提出了一种基于多扫描链的通用前滚恢复方法及其改进版本。然而,文献[14 – 16]的状态同步均需与检测点结合,且同步过程会造成系统功能的中断或延迟。因此,本文提出了一种基于工作输入和健康现态的 TMR 系统状态同步技术(Present-Input and Healthy-State based State Synchronization technique for TMR, PIHS3TMR),该方法无须设置检测点或缓存,且同步过程不会造成系统功能中断或延迟。

## 1　SpW 路由器结构和容错机制设计

### 1.1　路由器系统的总体结构

如图 1 所示,路由器的整体结构采用基于 SRAM 型 FPGA 的可编程片上系统结构。

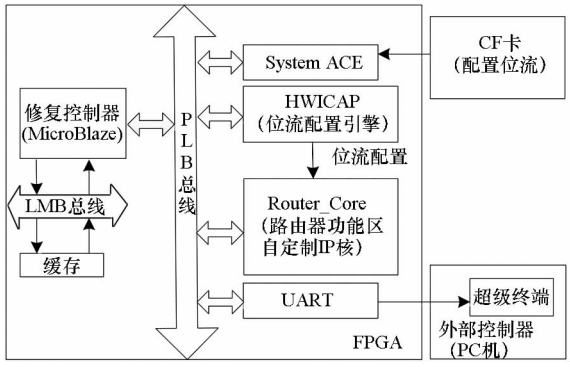



图 1　系统总体设计结构

Fig. 1　Overall structure of the system

其中软核处理器 MicroBlaze 用作修复控制器,缓存用于存储其指令和数据;自定制 IP 核 Router_Core 具有 TMR 结构,用于执行 SpW 路由器的主要功能;所有位流文件均存储于外部 CF 卡中。检测到一个模块故障时,通过 System ACE 控制器从 CF 卡读取相应的位流文件,并由位流

配置引擎 HWICAP 下载到 FPGA 的配置存储器中,从而修复故障模块。UART 用于实现 FPGA 与 PC 之间的通信。

### 1.2 Router_Core 的容错设计与故障修复机制

Router_Core 的结构如图 2 所示,由三个可重构模块(Reconfigurable Module,RM)组成,分别命名为 RMA、RMB 和 RMC;表决与检测模块不仅可对 RMA、RMB 和 RMC 的输出(即 $Y_a$,$Y_b$ 和 $Y_c$)进行表决得到系统输出 $Y$,且可进行故障检测和定位,并向修复控制器输出相应故障指示信号。

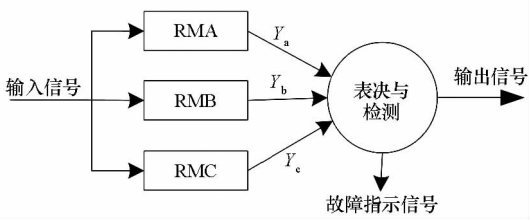



图 2 Router_Core 的结构图

Fig. 2 Structure diagram of the Router_Core

图 2 中,RMA、RMB 和 RMC 是三个执行相同路由功能的独立模块;某模块发生故障时,TMR 结构将通过多数表决屏蔽该模块。然而,TMR 系统在任何时刻均只能容忍一个模块的故障;当两个模块同时故障时,系统将输出错误结果。为了使系统能从软错误中恢复,本文采用 DPS 技术。任一模块发生软错误时,表决与检测模块将检测到它并向修复控制器发送故障指示信号;修复控制器将通过 System ACE 从 CF 卡读取相应位流文件以修复故障模块。由于本文系统具有自修复能力,下文将其称为可修复 TMR(Repairable TMR,RTMR)。

DPS 技术虽可恢复系统中组合电路的逻辑,但无法确保修复后模块中时序电路的状态与其他两个模块一致。若不进行状态同步,修复后模块将无法跟上无故障模块的步伐,无法使系统恢复至原始容错能力。因此,本文提出一种称作 PIHS3TMR 的基于工作输入和健康现态的 TMR 系统状态同步技术来实现状态同步,并确保系统在状态同步过程中提供正常服务。

另外,为节省位流存储空间,本文采用位流重定位技术,只存储一个功能模块的部分位流文件,在修复过程中进行修改并配置到另外两个模块中,可减少三分之二的位流存储空间,并降低原始位流故障的可能性。

### 1.3 状态同步技术

#### 1.3.1 时序逻辑电路的模型

时序电路具有记忆功能,其输出由电路当前输入和现态共同决定。大部分时序电路可建模为如图 3 所示的由组合逻辑电路(Combinational Logic Circuits,CLCs)和触发器(Flip-Flops,FFs)组成的状态机。

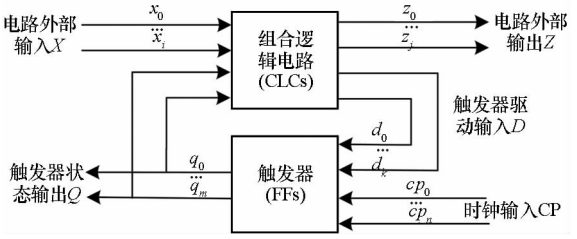



图 3 时序电路模型

Fig. 3 Model of the sequential logic circuit

图 3 中,$X=(x_0,x_1,\cdots,x_i)$ 和 $Z=(z_0,z_1,\cdots,z_j)$ 分别为电路外部输入和输出;$D=(d_0,d_1,\cdots,d_k)$ 为触发器的驱动输入;$Q=(q_0,q_1,\cdots,q_m)$ 为电路的状态,即触发器的状态;$CP=(cp_0,cp_1,\cdots,cp_n)$ 为时钟脉冲。$X$、$Z$、$D$、$Q$ 之间的关系可用式(1)~(3)描述。

$$Z^n=F_1(X^n,Q^n) \tag{1}$$

$$D^n=F_2(X^n,Q^n) \tag{2}$$

$$Q^{n+1}=F_3(D^n,Q^n) \tag{3}$$

其中:$n$ 和 $n+1$ 分别代表两个相邻的离散时刻 $t_n$ 和 $t_{n+1}$,$Z^n$、$D^n$ 和 $X^n$ 分别表示 $t_n$ 时刻 $Z$、$D$ 和 $X$ 的值;$Q^n$ 表示 $t_n$ 时刻电路的状态,称作现态;$Q^{n+1}$ 表示 $t_{n+1}$ 时刻电路的状态,称作次态。式(1)为电路输出的表达式,称作输出方程;式(2)为触发器输入的控制方程,称作驱动方程;式(3)为电路次态与其现态及电路当前工作输入的关系,称作状态方程或次态方程。显然,$Z$ 和 $D$ 由组合逻辑实现,故 $t_n$ 时刻的输出仅取决于该时刻的输入;而 $Q$ 由触发器实现,故其 $t_{n+1}$ 时刻的输出 $Q^{n+1}$ 由当前输入 $D^n$ 和现态 $Q^n$ 共同确定。

使用 DPS 技术虽可恢复 TMR 系统中故障模块的电路,但不能保证修复后模块的现态与无故障模块一致。因此,必须进行状态同步。

#### 1.3.2 状态同步技术 PIHS3TMR

为了确保无故障模块正常工作并在状态同步过程中提供正常服务,提出了如图 4 所示的实时自修复 TMR 系统状态同步方法 PIHS3TMR。

图 4 中,将 TMR 系统中每个模块(记作模块 $y$)的状态机进行改造,在系统工作输入 $X$ 和时钟输入的基础上,增加了健康现态输入 $Q_h$($q_{0h}$ ~ $q_{mh}$)和状态同步控制信号 $SC_y$ 以及二选一多路复用器;状态输入 $Q$($q_0$ ~ $q_m$)可由 $SC_y$ 控制在本模块现态 $Q_y$($q_{0y}$ ~ $q_{my}$)和健康模块的现态 $Q_h$($q_{0h}$ ~

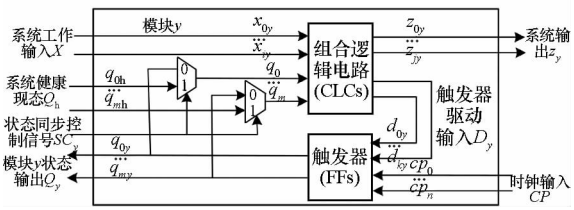



图 4　状态同步技术 PIHS3TMR

Fig. 4　PIHS3TMR, state synchronization technique

$q_{mh}$)之间选择,如式(4)所示。

$$Q = SC_y \cdot Q_h + \overline{SC_y} \cdot Q_y \tag{4}$$

正常工作过程中 $SC_y$ 为 0,选择本模块现态 $Q_y$ 作为 CLC 的输入,所以模块 $y$ 根据自己的状态机独立工作。状态同步过程中 $SC_y$ 为 1,选择健康现态 $Q_h$ 作为 CLC 的输入,可根据系统工作输入和健康现态立即构造出模块 $y$ 的次态,确保模块 $y$ 的状态与其他模块同步。此外,在整个电路恢复和状态同步过程中,无故障模块一直正常工作,为外界提供服务,无须中断系统的运行。

图 4 中健康现态可由三个模块的状态表决得到,亦可直接使用任一无故障模块的现态。由于为每个 FF 的输出插入表决器将加大面积开销和关键路径延迟,本文直接采用相邻模块的现态作为健康现态,如式(5)~(7)所示。

$$Q_{h1} = Q_2 \tag{5}$$

$$Q_{h2} = Q_3 \tag{6}$$

$$Q_{h3} = Q_1 \tag{7}$$

其中,$Q_{hi}$($i$ = 1,2,3)代表模块 $i$ 的健康现态。

### 1.4　位流重定位技术

执行 DPS 时,传统设计流程为某一部分重构区域(Partially Reconfigurable Region,PRR)产生的部分重构位流(Partially Reconfiguration Bitstream,PRB)不能重定位到其他 PRR。如图 5 所示,PRR1 的位流 adder_1 无法配置到 PRR2 和 PRR3 中[17]。

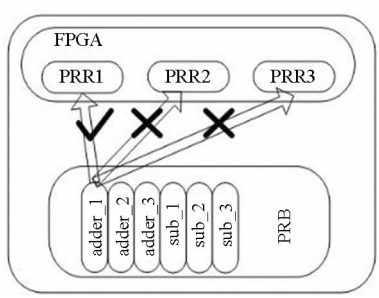



图 5　DPS 系统的位流配置

Fig. 5　Bitstream configuration of DPS system

为实现位流重定位,应遵循以下两个步骤。

**步骤 1**:设计时确保 PRR 完全相同,要求如下。

1)可重构资源的数量相同;

2)可重构资源的相对布局一致;

3)代理逻辑的相对位置一致;

4)代理逻辑和静态区域之间互联的相对路由路径一致;

5)静态区域的信号布线不能穿过动态区域。

**步骤 2**:重定位之前必须修改配置位流。关键是修改帧地址和循环冗余校验(Cyclic Redundancy Check,CRC)值。帧地址为 16 进制命令字(30002001)后第一个字,表示 PRR 的起始地址。CRC 值为 16 进制命令字(30000001)后第一个字,用于检查位流的有效性。创建重定位位流时,必须先用目标 PRR 的帧地址替换原始 PRR 的帧地址;然后根据式(8)重新计算 CRC 值并替换原 CRC 值。

$$x^{32} + x^{28} + x^{27} + x^{26} + x^{25} + x^{23} + x^{22} + x^{20} + x^{19} + x^{18} + x^{14} + x^{13} + x^{11} + x^{10} + x^9 + x^8 + x^6 + 1 \tag{8}$$

## 2　实验结果和讨论

使用 Xilinx Virtex - 5 FPGA 开发套件 ML507,以具有 4 个端口的 SpW 路由器为例来验证所提出的系统结构和自修复机制。

### 2.1　SpW 路由器的典型结构

如图 6 所示,SpW 路由器通常包含若干 SpW 端口和一个交换矩阵。前者为所有 SpW 节点和路由器提供接口,后者用于实现各端口间数据的交换。

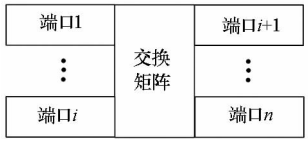



图 6　SpW 路由器的典型结构

Fig. 6　General block diagram of the SpW router

端口通常由状态机、发送器、接收器、发送 FIFO 和接收 FIFO 组成。其中状态机控制端口的所有操作;接收器对链路数据进行解码并将其传输至接收 FIFO,再由接收 FIFO 传输至主机接口;发送 FIFO 将来自主机接口的用户数据传输至发送器,再由发送器编码后传输至链路。

交换矩阵包含地址识别和矩阵路由两部分。前者解析包头所包含的目的地址,后者调控数据包的传输路径。

### 2.2　状态同步技术的结果

以图 7 所示 SpW 端口的状态机为例,验证所

提出状态同步技术的有效性。

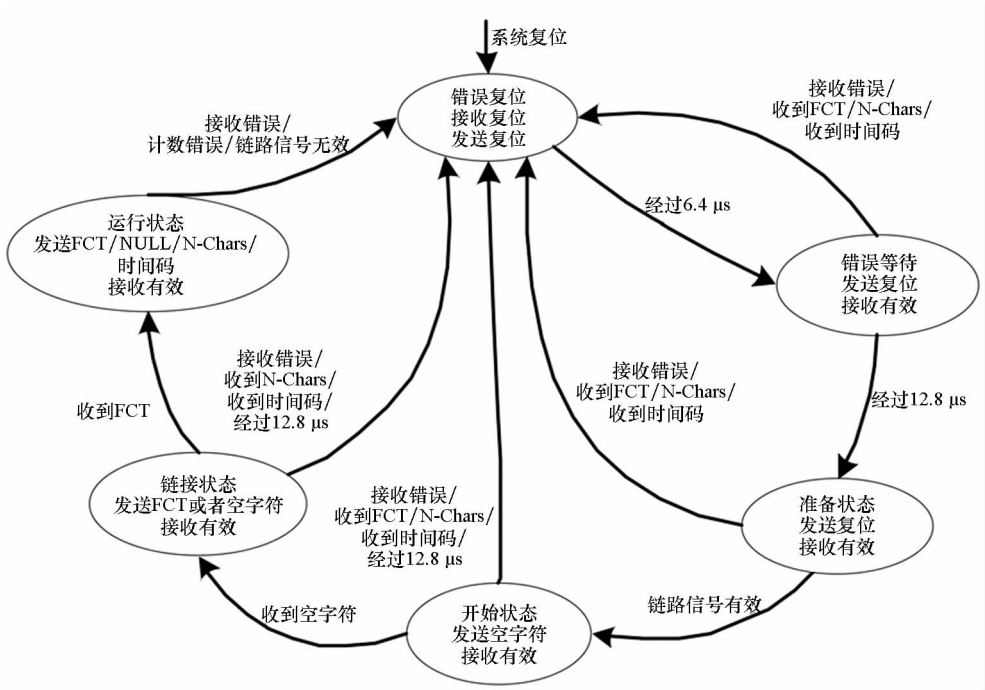



图 7　链路状态转换图

Fig. 7　State transition diagram of the state machine

TMR 结构的 SpW 端口上电时状态机处于“错误复位”状态，发送端与接收端的链路接口将自动连接，并最终处于“运行”状态。某一模块发生错误时，将对其进行 DPS 纠正错误，之后其状态机进入“错误复位”状态，但此时其他模块仍处于“运行”状态并正常运行。三个模块共享链路并与同一个节点进行通信，因此没有握手信号使该模块与节点再次自动建立连接。若不进行状态同步，该模块将一直处于“错误复位”状态，无法转换为“运行”状态。PIHS3TMR 技术可使修复后模块根据工作输入和健康现态进行状态转换，实现与无故障模块的状态同步。采用 HDL 语言设计该状态机的 PIHS3TMR 系统，在 Xilinx ISE 软件中的仿真结果如图 8 所示。

图 8 中，clk 为时钟输入，rst、sc 和 state 分别为复位信号、状态同步控制信号和状态输出信号；active 表示建立链路连接的“运行”状态。所有以“sanmo”为前缀的信号表示健康状态，以“yimo”为前缀的信号表示修复模块（记为 $RM_y$）的状态；矢量信号的显示格式为“$n$′hxxx”，其中“$n$”是原始矢量的位数，“hxxx”是其十六进制值。由图 8 可见，在第 1 ~ 3 个时钟周期，三个模块均处于“运行”状态并正常工作，因此其状态相同，均为“3′h5”。在第 4 个时钟周期，$RM_y$ 的复位信号有效（模拟修复后 $RM_y$ 的初始状态），其状态（yimo/statem/state）为“3′h0”（错误复位），与其他模块不一致。由于采用了状态同步技术，第 5 个时钟周期，$RM_y$ 的 sc 有效，故其状态立即根据健康现态（sanmo/statem/state）从“3′h0”跳到“3′h5”，与其他模块一致；且 active 再次有效。第 6 ~ 7 个时钟周期，三个模块均正常工作。可见 PIHS3TMR 的理想状态同步时间仅为 1 个时钟周期。尤为重要的是，状态机在整个工作过程中均提供正常服务，不会中断系统功能。

## 2.3　SpW 路由器的实现结果

使用 VHDL 语言分别设计 PIHS3TMR 结构和

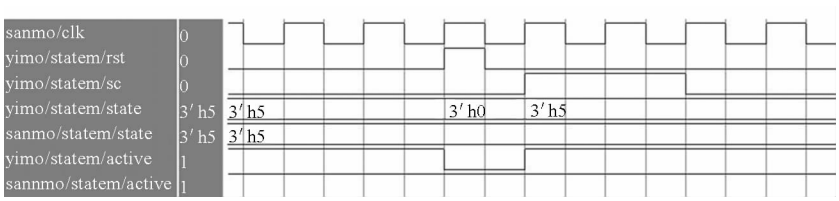



图 8　状态机的 PIHS3TMR 状态同步仿真结果

Fig. 8　Simulation results of the PIHS3TMR design for the state machine

传统 TMR 结构的四端口 SpW 路由器,二者实现结果比较见表 1。

由表 1 可见,与传统 TMR 相比,PIHS3TMR 所用 BRAM/FIFO 资源数量相同;所用寄存器和查找表资源的数量分别增加了约 13.7% 和 41.8%,但所用资源占可用资源总量的比例依然很少(分别为 6% 和 13%)。所用资源数量增加的主要原因在于 PIHS3TMR 结构的状态同步需要增加多路选择器。由于 FPGA 有大量的资源冗余,所以 PIHS3TMR 的资源代价是可接受的。

**表 1　SpW 路由器的实现结果比较**

Tab. 1　Implementation results comparison for the SpW router

| 分类 | 可用资源数 | 传统 TMR | | PIHS3TMR | |
|---|---|---|---|---|---|
| | | 已使用 | 利用率/% | 已使用 | 利用率/% |
| 寄存器 | 44 800 | 2300 | 5 | 2614 | 6 |
| 查找表 | 44 800 | 4198 | 9 | 5954 | 13 |
| BRAM/FIFO | 148 | 8 | 5 | 8 | 5 |

## 2.4　位流重定位技术的结果

在 ML507 开发板上实现了所提 SpW 路由器的系统结构,并分别采用位流重定位技术(本文方法)和非位流重定位技术(传统方法)进行 DPS,所需部分位流文件数量和存储器空间大小如表 2 所示。由表 2 可见,传统方法所需部分位流文件数量是本文方法的 3 倍。由于每个部分位流文件的大小为 133 Kbit,因此本文方法可以节省 266 Kbit,即大约 66.7% 的存储空间;同时也降低了原始位流本身故障的可能性。

**表 2　位流重定位和非位流重定位技术的比较**

Tab. 2　Comparison of bitstream relocation and non-bitstream-relocation techniques

| | 位流文件数量 | 存储空间大小 |
|---|---|---|
| 位流重定位技术 | 1 | 133 Kbit |
| 非位流重定位技术 | 3 | 399 Kbit |

目前仅考虑 SEU 之类软故障的纠正,每个模块仅需一个位流文件;未来考虑硬故障修复时,每个模块均需多个位流文件以应对 FPGA 不同位置的硬故障,位流重定位技术的优势将更明显。

DPS 所需时间可用式(9)描述,其中 $T_{cf}$ 为读取位流文件并将其存储至 System ACE 缓冲区所需时间,$T_{icap}$ 为利用位流文件刷新配置存储器所需时间,$T_{add}$ 为修改帧地址和 CRC 的额外时间开销。

$$T_{rec} = T_{cf} + T_{icap} + T_{add} \tag{9}$$

$T_{cf}$ 的值可由式(10)[18] 计算,其中 $L$ 为以位(bit)为单位的位流文件大小,0.634 18 为 System ACE 控制器以 MByte/s 为单位的平均带宽值,$T_{cf}$ 的单位为 ms。

$$T_{cf} = L \times 10^{-3} / 0.634\ 18 \tag{10}$$

$T_{icap}$ 的值可由式(11)计算,其中 $L'$ 是以字节(Byte)为单位的位流文件大小,$CLK$ 是以 Hz 为单位的内部配置访问端口(Internal Configuration Access Port, ICAP)时钟频率,$ICAP_{宽度}$ 是以 Byte 为单位的 ICAP 宽度。

$$T_{icap} = L' / (CLK \times ICAP_{宽度}) \tag{11}$$

本文系统的时钟频率为 100 MHz(周期为 10 ns),位流文件大小为 133 Kbit,因此据式(10)和式(11)可算出 $T_{cf}$ 和 $T_{icap}$ 的值分别约为209.72 ms 和 42.56 μs。由于修改帧地址和 CRC 所需时间仅为几十纳秒(几个时钟周期),故与 $T_{cf}$ 和 $T_{icap}$ 相比,$T_{add}$ 可忽略。因此,DPS 时间约为 209.76 ms。

## 2.5　可靠性和可用性分析

### 2.5.1　系统可靠性模型

MicroBlaze 和表决与检测模块未来可进一步采取措施加固,故分析系统可靠性时,暂时忽略其故障影响,仅考虑 Router_Core 核的可靠性。

假设图 2 所示 RTMR 结构的 Router_Core 核有三种状态——正常状态(三个模块均正常工作),修复状态(存在故障模块并正在修复)和故障状态(存在两个或更多故障模块),并假定在相当短的时间 $\Delta t$ 内,两个或多个模块同时发生状态转换的概率是 $\Delta t$ 的高阶无穷小,可以忽略,则该系统的马尔可夫状态空间模型可用图 9 表示。

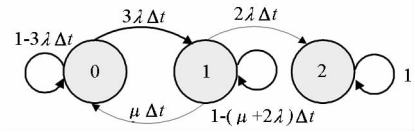



图 9　自修复 TMR 系统状态空间图

Fig. 9　Self-repair TMR system state space diagram

图 9 中,0 表示正常状态,1 表示修复状态,2 表示失效状态;$\lambda$ 为每个模块的故障率;$\mu$ 为修复率;$\Delta t$ 为时间间隔。则 RTMR 系统的可靠度如式(12)[6] 所示。

$$R(t) = \frac{k+5+\sqrt{k^2+10k+1}}{2\sqrt{k^2+10k+1}} e^{-\frac{k+5-\sqrt{k^2+10k+1}}{2}\lambda t} - \frac{k+5-\sqrt{k^2+10k+1}}{2\sqrt{k^2+10k+1}} e^{-\frac{k+5+\sqrt{k^2+10k+1}}{2}\lambda t} \tag{12}$$

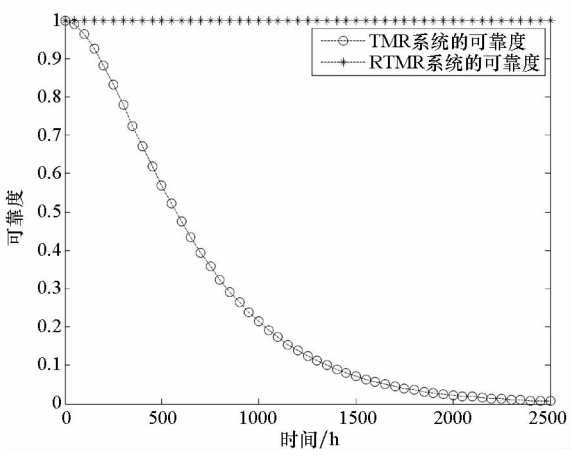
式中,$\lambda$ 为单个模块的故障率,即每小时失效次

数，$k=\mu/\lambda$；$\mu$ 为修复率，即每小时成功修复次数。传统 TMR 系统的可靠度[6]为：

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

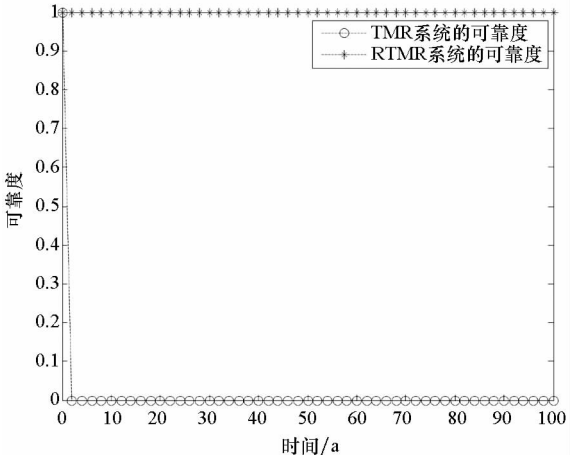
2.5.2 系统可靠性分析

系统的平均修复时间（Mean-Time-To-Repair，MTTR）包括 DPS 时间和状态同步时间。由于 PIHS3TMR 的状态同步仅需 1 个时钟周期，故其 *MTTR* 约等于 DPS 时间，即约为 209.76 ms，可得修复率 $\mu = 3600/MTTR \approx 1.72 \times 10^4$ 次/h。假设 Virtex－5 器件工作于地球同步轨道（Geostationary Earth Orbit，GEO）（36 000 km），则典型日 SEU 故障率为 $3.8 \times 10^{-10}$ 次/（bit · d）[19]，因此 SpW 路由器每个模块的 SEU 故障率为 $\lambda = 3.8 \times 10^{-10} \times 133 \times 10^3 \times 24 \approx 1.21 \times 10^{-3}$ 次/h；进而可得 $k = \mu/\lambda \approx 1.42 \times 10^7$。因此，由式（12）和式（13）可得 RTMR 结构（本文结构）和传统 TMR 结构的 SpW 路由器的可靠度，如图 10 所示（这里仅考虑软故障）。




（a）0～2500 h 的可靠度对比

（a）Reliability from 0 h to 2500 h




（b）0～100 a 的可靠度对比

（b）Reliability from 0 a to 100 a

图 10 TMR 和 RTMR 结构 SpW 路由器可靠度对比

Fig. 10 Reliability comparison of the SpW routers with TMR structure and RTMR structure

由图 10 可见，随着持续工作时间的增加，传统 TMR 系统的可靠度呈指数规律递减；持续工作 2300 h 后，其可靠度快速递减至 0。而本文 RTMR 系统的可靠度近似保持不变；若仅考虑软故障，甚至持续工作 100 a 后，其可靠度仍可达到约 0.999 6。这种高可靠度特别适合要求高可靠性和长寿命的空间应用。

2.5.3 系统可用性分析

可用度可定义[7]为：

$$Av = \frac{1}{1 + (MTTD + MTTR)/MTTF} \tag{14}$$

其中，平均故障间隔时间（Mean-Time-To-Failure，MTTF）定义为 SpW 一个模块持续无故障操作的时间，平均检测时间（Mean-Time-To-Detect，MTTD）定义为配置存储器发生错误到该错误被检测到的时间间隔。无错误发生时，*MTTD* 和 *MTTR* 均为 0，故 *Av* 为 1，意味着 FPGA 器件是完全可用的。配置存储器中发生 SEU 时，若设计对其敏感，将会发生错误，则表决与检测模块将会立即检测到该错误，因此 *MTTD* 可以忽略。所以，对本文系统，式（14）可以简化为

$$Av = \frac{1}{1 + MTTR/MTTF} \tag{15}$$

本文系统的 *MTTR* 约为 209.76 ms，即 $5.827 \times 10^{-5}$ h，$MTTF = 1/\lambda = 1/(1.213 \times 10^{-3})$ h，因此由式（15）可得其可用度约为 0.999 999 9，非常接近于 1。而对于传统 TMR 结构，当两个以上模块发生故障时，*MTTR* 为无穷大，因此其可用度为 0。

## 3 结论

首次将系统级三模冗余、动态部分刷新、位流重定位技术，以及基于工作输入与健康现态的实时状态同步方法 PIHS3TMR 结合，提出了一种基于 SRAM 型 FPGA 的高可靠 SpaceWire 路由器设计方法。PIHS3TMR 技术的引入保证了路由器工作过程中始终提供正常服务，不会引起系统功能中断或延迟；且状态同步时间仅为 1 个时钟周期，有助于减少平均修复时间。位流重定位技术的采用将所需存储空间减少三分之二，同时也降低了原始位流本身故障的可能性；当每个模块需要多个位流以应对不同位置硬故障时，其优势更为明显。

经三模表决后路由器数据交换更可靠，且路由器自身硬件可自修复，可靠性和可用性高。Virtex－5 器件工作于 GEO 时（典型日 SEU 故障率为 $3.8 \times 10^{-10}$ 次/（bit · d））,若仅考虑软故障，

持续工作 100 a 后,其可靠度仍可达到约0. 999 6,且其可用度约为 0. 999 999 9,非常接近于 1。

所提系统结构和设计方法是通用的,可用于设计其他需要高可靠性、好实时性、小批量和易于升级的系统。目前仅对 SpW 路由器核心功能区进行了 TMR 加固,未来将进一步研究修复控制器和表决与检测模块的加固。

## 参考文献(References)


[1] 朱晓燕, 陶利民, 张伟功, 等. 面向卫星数据系统的 SpaceWire 应用模型仿真研究[J]. 小型微型计算机系统, 2015, 36(3): 616 – 620.
ZHU Xiaoyan, TAO Limin, ZHANG Weigong, et al. SpaceWire application model simulation for satellite data systems[J]. Journal of Chinese Computer Systems, 2015, 36(3): 616 – 620. (in Chinese)

[2] 闫梦婷, 安军社, 龚泉铭. SpaceWire 总线的双路由单元性能评价方法[J]. 国防科技大学学报, 2017, 39(1): 86 – 91.
YAN Mengting, AN Junshe, GONG Quanming, et al. Performance evaluation method for dual-route unit based on SpaceWire bus[J]. Journal of National University of Defense Technology, 2017, 39(1): 86 – 91. (in Chinese)

[3] Gibson D, Parkes S, Mcclements C, et al. SpaceWire-D on the castor spaceflight processor [C]//Proceedings of International SpaceWire Conference, 2014.

[4] AT7910E:SpW-10X SpaceWire router[EB/OL]. (2008 – 08 – 10)[2018 – 04 – 02]. http://pdfl. alldatasheet. com/datasheet-pdf/view/257040/ATMEL/AT7910E. html.

[5] Ekergarn J, Habinc S, Ringhage F, et al. GR718 – radiation-tolerant 18 × SpaceWire router based on the DARE 180 nm library[C]//Proceedings of International SpaceWire Conference, 2014.

[6] 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, 38(1): 177 – 183.
YAO Rui, WANG Youren, YU Shenglin, et al. Design and experiments of enhanced fault-tolerant triple-module redundancy systems capable of online self-repairing[J]. Acta Electronica Sinica, 2010, 38(1): 177 – 183. (in Chinese)

[7] Tarrillo J, Chipana R, Chielle E, et al. Designing and analyzing a SpaceWire router IP for soft errors detection[C]// Proceedings of Latin American Test Workshop, 2011: 1 – 6.

[8] Tarrillo J, Altieri M, Kastensmidt F L. Improving error detection capability of a SpaceWire router IP [C]// Proceedings of European Conference on Radiation and ITS Effects on Components and System, 2011: 501 – 506.

[9] Petri E, Saponara S, Tonarelli M, et al. Mitigating radiation effects on ICs at device and architectural levels: the SpaceWire router case study [C]//Proceedings of IEEE International Symposium on Industrial Electronics, 2007: 3310 – 3315.

[10] Taube S, Petrovic V, Krstic M. Fault tolerant implementation of a SpaceWire interface [C]//Proceedings of IEEE International Conference on Electronics, Circuits and Systems, 2015: 604 – 609.

[11] 郑晓云, 陶淑苹, 冯汝鹏, 等. SRAM 型 FPGA 抗单粒子翻转技术研究[J]. 电子测量技术, 2015, 38(1): 59 – 63.
ZHENG Xiaoyun, TAO Shuping, FENG Rupeng, et al. Research on anti-single-particle flipping technology of SRAM FPGA [J]. Electronic Measurement Technology, 2015, 38(1): 59 – 63. (in Chinese)

[12] Pradhan D K, Vaidya N H. Roll-forward and rollback recovery: performance-reliability trade-off [J]. IEEE Transactions on Computers, 1994, 46(3): 372 – 378.

[13] Xu J, Randell B. Roll-forward error recovery in embedded real-time systems [C]//Proceedings of International Conference on Parallel and Distributed Systems, 1996: 414 – 421.

[14] Yu S Y, McCluskey E J. On-line testing and recovery in TMR systems for real-time applications [C]//Proceedings of International Test Conference, 2001: 240 – 249.

[15] Ebrahimi M, Miremadi S G, Asadi H. ScTMR: a scan chain-based error recovery technique for TMR systems in safety-critical applications[C]//Proceedings of Design, Automation & Test in Europe, 2011.

[16] Ebrahimi M, Miremadi S G, Asadi H, et al. Low-cost scan-chain-based technique to recover multiple errors in TMR systems [J]. IEEE Transactions on Very Large Scale Integration Systems, 2013, 21(8): 1454 – 1468.

[17] 姚睿, 何坤, 朱萍, 等. 使用位流重定位与差异配置在线演化数字系统[J]. 国防科技大学学报, 2017, 39(3): 69 – 76.
YAO Rui, HE Kun, ZHU Ping, et al. Online evolution of the digital system on bitstream relocation and discrepancy configuration[J]. Journal of National University of Defense Technology, 2017, 39(3): 69 – 76. (in Chinese)

[18] 刘春红. 基于 Virtex 5 的 DRP SoC 自重构系统设计与性能评估[D]. 西安: 西安电子科技大学, 2013.
LIU Chunhong. Design and performance evaluation of SoC DRP self reconfigurable system based on Virtex 5 [D]. Xi′an: Xi′an Electronic and Science University, 2013. (in Chinese)

[19] Xilinx. DS192:radiation-hardened, space-grade Virtex – 5QV family data sheet: overview(v1. 6)[EB/OL]. (2018 – 01 – 11)[2018 – 04 – 02]. http://www. Xilinx. com/support/documentation/data _ sheets/ds192 _ V5QV _ Device _ Overview. pdf.