

Software Design and Implementation of Single Particle Soft Error Test System

Xiaoling Lai¹, Xiang Gao¹, Liang Jia¹, Xiaoping Huang², Hao Wu², Yangming Guo²

¹Academy of Space Technology (Xi'an), Xi'an Shaanxi

²School of Computer Science and Technology, Northwestern Polytechnical University, Xi'an Shaanxi
Email: yangming_g@nwpu.edu.cn, cowboy-gx@163.com

Received: Jun. 3rd, 2020; accepted: Jun. 16th, 2020; published: Jun. 24th, 2020

Abstract

The digital system of space vehicle is mainly composed of FPGA, CPU and other digital devices. Because it works in the space environment with a large number of radiation particles, single particle flip and other soft errors pose a great threat to the safety and reliability of the vehicle, resulting in the failure of navigation, measurement and control, communication and other functions. Especially with the increasing application of COTS devices in digital systems, it is urgent to establish a comprehensive and accurate method to detect the whole aircraft soft errors. This paper takes XC5VFX130T SRAM FPGA of Xilinx Company and TMS570LS3137 CPU of TI Company and their digital system as objects, designs the communication protocol between the monitoring FPGA and the upper computer based on the test system, and then the host computer software is implemented.

Keywords

FPGA, CPU, Single Event Effect, Single Particle Soft Error, Test System

一种单粒子软错误测试系统的软件设计与实现

赖晓玲¹, 高翔¹, 贾亮¹, 黄小平², 吴昊², 郭阳明²

¹中国空间技术研究院西安分院, 陕西 西安

²西北工业大学计算机学院, 陕西 西安

Email: yangming_g@nwpu.edu.cn, cowboy-gx@163.com

收稿日期: 2020年6月3日; 录用日期: 2020年6月16日; 发布日期: 2020年6月24日

摘要

空间飞行器数字系统主要由FPGA、CPU等数字器件构成, 由于其工作在含有大量辐射粒子的宇宙环

文章引用: 赖晓玲, 高翔, 贾亮, 黄小平, 吴昊, 郭阳明. 一种单粒子软错误测试系统的软件设计与实现[J]. 计算机科学与应用, 2020, 10(6): 1259-1266. DOI: 10.12677/csa.2020.106130

境下,单粒子翻转等软错误对飞行器的安全性、可靠性造成了很大威胁,造成导航、测控以及通讯等功能的失效。特别是随着COTS器件越来越多地应用到数字系统中,亟需建立全面并准确检测出整个飞行器软错误的方法。论文结合Xilinx公司的XC5VFX130T SRAM型FPGA和TI公司的TMS570LS3137型号CPU器件构建的测试系统,设计了监控FPGA和上位机之间的通讯协议,开展了上位机软件设计与实现。

关键词

FPGA, CPU, 单粒子效应, 单粒子软错误, 测试系统

Copyright © 2020 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

空间飞行器的导航、姿态控制以及空地通信等功能都是由数字系统来实现的[1]。空间环境中的辐射粒子很容易在数字系统中引起辐射效应,使得空间飞行器的安全受到严重威胁[2]。为了保证飞行器的长寿命和高可靠性,研究人员针对FPGA、CPU等器件提出了众多软错误检测方法,以便评估出这些器件的单粒子翻转敏感特性,为空间飞行器数字系统的设计选型和防护设计评价提供依据和支撑,对寻找预防和减缓空间粒子效应具有重要的价值。

论文围绕TI公司的TMS570LS3137 CPU和Xilinx公司的Virtex5 XC5VFX130T等两类新型COTS器件电离辐射摸底试验,在分析已有单粒子软错误检测方法的基础上构建了测试系统,设计了监控FPGA和上位机之间的通讯协议,开展了上位机软件的设计与实现。

2. 测试系统描述

辐照环境下的软错误测试系统用以控制试验过程,并且能够采集、传输、处理、保存试验数据[3]。一般将被测FPGA等芯片放在真空中,以避免加速器中射出的高能粒子到达芯片表面过程中能量衰减过多,导致了对外界环境和被测芯片之间的连线类型和数量的要求很苛刻。此外,为防止加速器对试验人员产生辐射伤害,实际操作中试验人员一般通过计算机对测试系统进行远程操作,即被测芯片距离试验控制室较远。

为此,软错误测试系统分为监控FPGA、被测FPGA、CPU和上位机等部分。测试系统的总体结构如图1所示。图1中,上位机试验时一般处于真空室外,在辐照过程中由控制室的计算机远程控制,监控FPGA通过用户数据报协议(UDP)与PC进行数据传输,以达到测试环境要求的高速可靠的数据交互和板间最少连线的双重目的。

上位机作为测试系统控制中心与数据存储与分析平台,设置测试条件与参数;向下位机发送测试模式切换、测试过程控制等指令;静态测试模式下,回读被测FPGA的配置数据并与参照文件比较;实时统计FPGA动态测试以及CPU测试的翻转数量;保存每一次测试的数据和结果,用于后续数据分析。

监控FPGA通过网口与上位机直接相连,其主要功能有接收上位机下发的测试控制指令并执行相应操作;CPU测试模式下给被测FPGA提供时钟;将测试结果通过网口上传给上位机。

监控FPGA主要由以下模块组成:

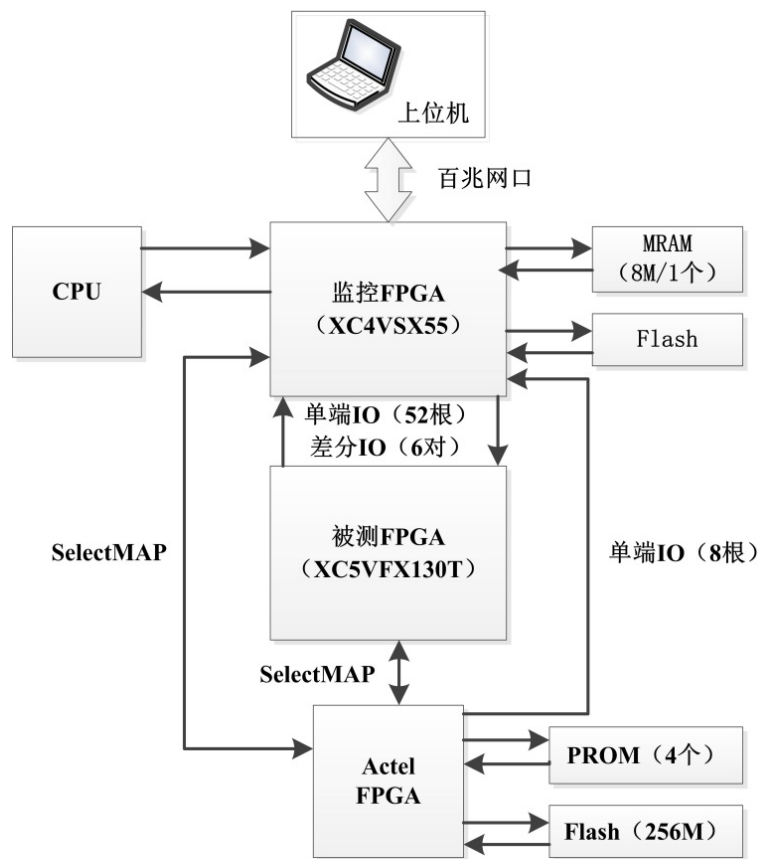


Figure 1. General structure of soft error test system
图 1. 软错误测试系统总体结构

1) 网口驱动模块，上位计算机与测试板的通讯接口。上位计算机通过网口向测试板发送各种控制命令；测试板通过网口向上位计算机传回测试状态与测试数据。采用 UDP 通讯协议实现。

2) 静态测试模块，采用“一次回读”的方式，即辐照一段时间后再对被测 FPGA 通过 SelectMAP 口回读，并区分出配置存储器和 BRAM 的配置信息。

3) 动态测试模块，为 BRAM 和触发器动态测试提供输入时钟，并接收测试输出。与静态测试“先辐照后统计”不同的是，在被测 FPGA 被辐照的同时，该模块统计翻转位数，即“边辐照边统计”。

FPGA 软错误检测方法以测试电路的形式在被测 FPGA 中实现。FPGA 测试模式下，被测 FPGA 的功能主要有静态测试模式下，通过 Select MAP 口将配置数据发给监控 FPGA；动态测试模式下，通过信号线将测试电路的输出结果上传给监控 FPGA。而在 CPU 测试模式下，被测 FPGA 的功能是接收来自 CPU 的测试数据并将其转发给监控 FPGA [4] [5]。

CPU 软错误检测方法以测试程序的形式在被测 CPU 中实现。CPU 的功能主要有实现寄存器、SRAM、Flash 等资源的测试程序；通过串口与上位机直接相连，在 CPU 测试过程中向上位机传输测试相关的信息。

软错误测试系统在实际辐照试验中运行时，涉及到上位机软件与监控 FPGA(下位机)之间的指令发送与接收、数据收发等通讯操作。考虑到测试过程中有大量的测试数据上传，使用端口号分别为 5000、5001 何 5002 的三个端口，设计测试系统的上位机与监控 FPGA 之间基于 UDP 的通讯协议。

1) 5000 端口主要用于上位机下发测试控制指令给监控 FPGA。网口下发的 4 字节长度的指令结构如

表 1 所示。

Table 1. Command structure of network ports

表 1. 网口下发指令结构

有效标志		指令数据	
EB	90	XX	XX

“EB 90”为指令的有效标志，其后面是两个字节的指令数据。具体指令及其含义如表 2 所示。其中前 2 条指令用于在 FPGA 测试模式与 CPU 测试模式之间切换，第 3~12 条指令用于 FPGA 测试模式，其余用于 CPU 测试模式。

Table 2. Instruction issued by 5000 port and its meaning

表 2. 5000 端口下发指令及其含义

序号	指令	含义
1	EB 90 55 55	切换到 FPGA 测试模式
2	EB 90 55 AA	切换到 CPU 测试模式
3	EB 90 00 AA	确认静态测试
4	EB 90 00 BB	确认 BRAM 测试
5	EB 90 00 CC	确认触发器测试
6	EB 90 DD AA	开始静态测试
7	EB 90 DD BB	开始 BRAM 测试
8	EB 90 DD CC	开始触发器测试
9	EB 90 00 33	暂停静态测试
10	EB 90 EE BB	暂停 BRAM 测试
11	EB 90 EE CC	暂停触发器测试
12	EB 90 FF EE	结束 FPGA 任一测试模式
13	EB 90 32 32	开始寄存器测试(全 1)
14	EB 90 33 33	开始寄存器测试(全 0)
15	EB 90 30 30	开始 SRAM 测试(全 1)
16	EB 90 31 31	开始 SRAM 测试(全 0)
17	EB 90 36 36	开始 Flash 测试(全 1)
18	EB 90 37 37	开始 Flash 测试(全 0)
19	EB 90 38 38	开始定时器及全功能测试
20	EB 90 34 34	结束 CPU 任一测试模式
21	EB 90 02 02	复位 CPU

2) 5001 端口用于监控 FPGA 向上位机发送测试指令的响应帧或者测试数据。响应帧和测试数据的长度都是 15 个字节，具体结构如表 3 所示。其中，每一帧最后的填充字节不被上位机解析。

Table 3. 5001 port frame format and its meaning**表 3.** 5001 端口帧格式及其含义

序号	响应帧或数据帧	含义
1	EB 90 55 00 XX XX XX XX XX XX XX XX XX XX XX	指令执行成功, XX 为填充字节
2	EB 90 55 11 XX XX XX XX XX XX XX XX XX XX XX	指令执行失败, XX 为填充字节
3	EB 90 AA C3 XX XX XX XX XX XX XX XX XX XX X	监控 FPGA 通知上位机, 即将有动态测试数据上传, XX 为填充字节
4	EB C3 WW WW WW WW WW WW WW WW XX XX	FPGA 动态测试数据, WW 为当前 SEU 累计值, XX 为填充字节
5	EB C3 DQ YY YY YY ZZ ZZ SS XX XX XX XX XX XX	CPU 测试数据, Q 为测试模式序号, YY 为 SEU 地址, ZZ 为对应地址的内容, SS 为前 6 个字节的校验和, XX 为填充字节

3) 5002 端口用于 FPGA 静态测试中, 监控 FPGA 上传回读数据。由于回读的配置数据约为 49 Mb, 使用网口进行数据打包上传, 一包数据为 1028 个字节(1024 个字节为数据), 一次回读总计需要上传 6010 包。上传的数据格式如表 4 所示。

Table 4. Return data structure**表 4.** 回传数据结构

包头标志		有效数据			包序号	
EB	91	WW	...	WW	XX	XX

由于 XC5VFX130T 的回读配置数据为 1,538,320 字(49,226,240 bits), 每包数据为 1024 个字节(8192 bits), 无法整除。因此, 最后一包数据只有 64 个字节有效数据, 其余 960 个字节需要补为全 FF 的填充字节, 这部分不参与回读比对。

3. CPU 软件设计

CPU 测试模式下, 寄存器测试程序是用汇编语言实现的, 而 SRAM 和 Flash 的测试程序是以 C 语言实现的。下面介绍 CPU 的主体程序设计, 主体程序不仅包括上述测试程序, 还有对整个系统的初始化工作。图 2 表示的是 CPU 主体程序。

CPU 首先上电启动, 然后系统完成对串口、锁相环(Phase Locked Loop, PLL)、外部存储器接口(External Memory Interface, EMIF)等资源的初始化工作。其中, PLL 和 CPU 的时钟配置有关, 而 EMIF 用于 CPU 和监控 FPGA 之间交换数据。当监控 FPGA 收到上位机发来数据的同时, 将该数据转发给 CPU, CPU 判断收到的是不是一条测试指令。如果不是, 则忽略并继续等待接收测试指令; 如果是测试指令, 则主体程序流程继续。如果该指令的目的是让 CPU 开始测试, 这里假定该指令为“开始 SRAM 测试(全 1)”(EB 90 30 30)或者“开始 SRAM 测试(全 0)”(EB 90 31 31), 那么下面 CPU 主控程序进入 SRAM 测试子程序。CPU 先后完成对 SRAM 和 DMA 的初始化, SRAM 的初始化工作就是写入初值(全 0 或者全 1), 而 DMA 的初始化是为测试过程中接收大量测试数据做准备。辐照开启后, DMA 周期性地、按地址由低到高的顺序扫描 SRAM, 如果有软错误发生, CPU 将测试数据通过 EMIF 接口发给监控 FPGA, 监控 FPGA 随即又通过网口将其上传至上位机处理。

上述 SRAM 测试程序以及其他资源的测试程序本质都是一个循环, 待 DMA 扫描到被测资源的最高地址后, 又重新从最低地址继续扫描。因此 CPU 在一次扫描结束后, 会判断上位机是否下发了测试结束指令(EB 90 34 34)。如果是, 则退出测试子程序, 进入 CPU 主体程序, 继续等待接收新的测试指令; 否则, 继续开始当前被测资源的下一次扫描。

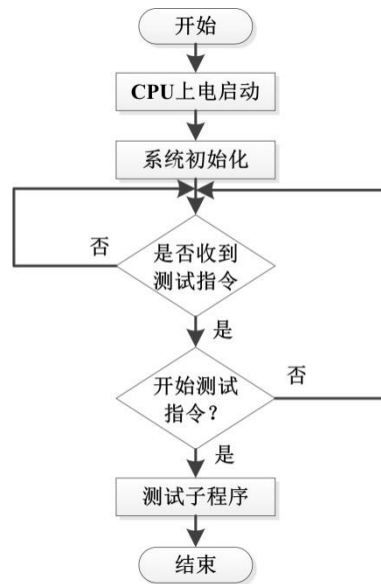


Figure 2. Flow chart of CPU software
图 2. CPU 软件的流程图

4. 上位机软件设计与实现

4.1. 上位机软件需求分析

测试系统上位机完成“测试过程控制”、“测试信息实时统计”和“测试结果保存”三大功能，各功能具体说明如下：

1) 测试过程控制

首先上位机应该可以在 FPGA 测试模式和 CPU 测试模式之间来回切换。若成功进入 FPGA 测试模式，上位机先选择具体的测试模式(如静态测试、BRMA 测试、Flip-flop 测试等)，并确认该模式。接下来可以对下位机发出“测试开始”、“测试暂停”、“测试结束”等命令。对于 FPGA 测试模式来说，需要特别指出的是，若要选择“静态测试”模式，在确认该测试模式前，需要指定回读文件存储目录。

若成功进入 CPU 测试模式，上位机同样先选择具体的测试模式(如寄存器测试、SRAM 测试、Flash 测试、计数器测试、全功能测试等)，然后选择并打开正确的串口以连接 CPU，并指定测试结果存储目录。与 FPGA 测试模式不同的是，CPU 测试模式下，主要有“测试开始”、“测试结束”和“CPU 复位”这三种命令。其中需要特别说明的是，试验人员可以实时看到 CPU 通过串口打印到上位机上的测试信息，若发现 CPU 因为受到 SEFI 影响或其他原因跑飞了，可以点击“CPU 复位”按钮，以恢复 CPU 程序的正常执行。

2) 测试信息实时统计

上位机可以实时统计每次辐照测试过程中的软错误数量并记录实验时长，而且可以累加统计当前测试模式下多次测试的软错误数量。对于 FPGA 的静态测试来说，上位机的界面上可以实时显示回读进度。

3) 测试结果保存

上位机可以保存与每一次测试有关的信息，包括剂量率、总注量、LET 值、离子种类、硅中射程、软错误数量、实验时长、实验开始时间以及结束时间等。另外，CPU 各测试模式下，上位机可以在后台自动保存 CPU 实时上传的软错误信息，例如寄存器测试中，若当前扫描到的寄存器发生了软错误，那么上位机会保存 CPU 上传的该寄存器的地址和值。

4.2. 上位机软件架构设计

上位机软件基于 Qt 5.7.0 框架实现, 主要使用 QUdpSocket 和 QtSerialPort 等 Qt 封装好的类。在 FPGA 静态测试和 CPU 测试过程中, 涉及到网口大量数据上传以及频繁的文件读写操作, 为了避免上位机的主界面假死而影响用户体验, 以及更严重的 UDP 丢包问题, 上位机采用多线程的方式, 将所有界面操作都放进主线程(界面线程), 而 UDP 数据接收和文件读写被放进了另外两个子线程(工作线程)。界面线程和工作线程之间的变量传递利用了 Qt 的信号 - 槽(Signal-Slot)机制。

上位机软件主要由“参数设置”模块、“测试控制”模块、“测试结果”模块和“数据处理”模块组成。

1) “参数设置”模块的功能是在本次测试开始前, 输入当前辐照设备的各种参数条件, 包括剂量率、总注量、LET 值、离子种类和硅中射程, 方便测试数据记录及脱机数据分析。

2) “测试控制”模块是整个上位机软件的核心模块。由于 FPGA 和 CPU 的测试原理不同, 该模块在两种测试模式下也有所不同。

对于 FPGA 测试来说, 若选择的是“静态测试”, 那么在确认模式前, 需要先指定回读文件存储的目录。根据配置存储器测试流程, 静态测试模式确认成功后, 开始测试, 辐照一段时间后, 暂停测试的同时, 对 FPGA 进行回读, 回读结束后立即与参照配置文件进行异或比较, 获得软错误数量。测试暂停后, 可以选择继续下一次测试或者结束当前测试模式。

上位机软件在接收 5002 端口数据的工作线程中, 声明了一个作用类似于缓冲区的 QQueue 队列, 将每一包数据先与参照配置文件中对应配置帧进行比较并统计软错误, 然后这一包数据入列。待回读结束时, 再循环出列将数据写入到文件中, 直到队列为空为止。由于队列是“先入先出”, 所以最后生成的回读文件中的配置信息也是按照包序号从小到大的顺序排列的。该设计思想也在一定程度上避免了丢包的问题; 若选择的是“BRAM 测试”或者“Flip-flop 测试”, 同样需要先确认模式, 然后开始测试, 在测试过程中, 若有软错误发生, 下位机会给上位机上传当前软错误数量的累计值。和“静态测试”模式一样, 暂停测试后, 既可以继续开始下一次测试, 也可以结束当前模式的测试。

对于 CPU 测试来说, 只需要在下拉列表中选择想要的测试模式。辐照试验开始前, 需确保 PC 已经通过串口与 CPU 正确连接, 并且指定测试结果的存储目录, 如果这串口未打开或者目录未指定, 上位机会弹出警告以提示用户。测试一旦开始, 上位机会弹出新的窗口, 该窗口仅绘制了一张表格用于模拟寄存器、SRAM 或 Flash, 行表示地址, 列表示具体的某一位。

由于 SRAM 和 Flash 的地址很大, 弹窗不显示所有地址, 当有软错误发生时, 才会将该错误的地址及其对应的内容在表格中更新, 这不仅避免了因为表格初始化耗时长且不方便实时观测而导致对用户体验的影响, 而且增强了软件本身的健壮性, 防止软件崩溃。如果被测资源发生了软错误, CPU 会上传错误所在的地址以及被测资源中该地址内容的值, 上位机软件在统计软错误数量的同时, 会在弹窗中对应的行和列同步显示错误。

针对测试过程中, CPU 被测资源被循环扫描可能引起错误重复统计的问题。软件设计中声明一个 QMap 容器, 该容器的键是地址, 值是该地址对应的内容, 该容器初始为空。当 CPU 扫描到某一地址时, 若发现 QMap 容器中没有值为该地址的键, 那么插入“地址 - 初始值”的键值对, 接着将 CPU 上传的改地址对应的值 value1 与该地址在 QMap 容器中对应的值 value2 比较, 若两者一样, 则上位机丢弃该数据, 否则在后台自动保存该数据, 并在主界面更新软错误数量的同时, 在弹窗中标注软错误发生的位置, 最后将该地址在 QMap 容器中对应的值更新为 value1, 这样就避免了对软错误的重复统计。与 FPGA 测试不同的是, CPU 测试中没有“暂停测试”的功能, 上位机软件可以直接结束测试, 可以选择继续开始下

一次测试，也可以选择切换其他的测试模式，但是在切换模式之前必须保存当前模式下的测试数据，否则会弹出警告提示保存。在辐照试验中，CPU 常常会因为 SEFI 或者其他原因而跑飞，因次在 CPU 测试模式下，上位机软件增加了“CPU 复位”的功能。

3) “测试结果”模块中，“总 SEU”表示当前测试模式下多次测试的软错误累计值；“当前 SEU”代表本次测试实时统计的软错误数量；“实验时长”是从测试开始时计时，直到测试暂停或者结束时停止计时；“实验开始时间”为测试开始的时间戳，而“实验结束时间”为测试暂停或者结束的时间戳。

4) “数据处理”模块的作用是在表格中记录当前测试模式下每一次“开始-暂停”或者“开始-结束”的测试信息，包括剂量率等测试条件、“测试结果”模块中的信息以及测试过程产生的中间文件名，若测试模式切换，表格的内容清空。对于 FPGA 测试来说，需要先指定一个目录，然后才能保存这些测试记录；而对于 CPU 测试，直接保存即可。

除了上述 4 个主要模块，上位机软件还设计了一个状态栏，用于显示测试过程各种信息提示，起到类似日志的作用。

5. 小结

论文从测试需求出发，对单粒子软错误测试系统进行功能定义以及模块划分，设计了整个系统的通讯协议。在对上位机软件需求分析的基础上，提出了软件架构设计和实现。应用表明，所设计的上位机软件，可以完成向测试系统下位机下发测试指令、实时监测被测器件翻转情况以及保存辐照试验数据等主要功能，大大降低了辐照试验的操作难度，具有很好的实用性，可以满足应用需求。

基金项目

国家某重点科研项目(MJ-2016-S-42, MJ-2018-S-34)，国防基础科研计划项目，国网浙江省电力有限公司科技项目(2020 年)，陕西省创新能力支撑计划项目(2019PT-03)。

参考文献

- [1] 李杨. 国外航天器电子系统的几个重要发展趋势[J]. 航天器工程, 2014, 23(6): 1-6.
- [2] 邢克飞. 星载信号处理平台单粒子效应检测与加固技术研究[D]: [博士学位论文]. 长沙: 国防科学技术大学, 2007.
- [3] 孙鹏. 抗单粒子翻转 SRAM-based FPGA 测试系统研究与设计[D]: [硕士学位论文]. 上海: 复旦大学, 2010.
- [4] 谢楠. 宇航用 FPGA 单粒子效应及监测方法研究[D]: [硕士学位论文]. 西安: 西安电子科技大学, 2011.
- [5] 宋凝芳, 朱明达, 潘雄. SRAM 型 FPGA 单粒子效应试验研究[J]. 宇航学报, 2012, 33(6): 836-842.