

# 基于 TSC695F 处理器的星载计算机 CPU 异常复位问题研究

邱源<sup>1,2</sup>, 纪丙华<sup>1,2</sup>, 沈奇<sup>1</sup>, 柳宜川<sup>1</sup>, 王志国<sup>1</sup>

(1. 上海航天电子技术研究所, 上海 201109; 2. 智能计算技术重点实验室, 上海 201109)

**摘要:** 针对基于 TSC695F 处理器的星载计算机模拟在轨飞行测试中 CPU 板异常复位问题进行研究; 通过使用逻辑分析仪捕捉过程数据, 定位问题发生部位, 发现导致星载计算机复位的根本原因是数据装载出错; 结合芯片驱动能力限度参数分析, 发现 CPU 板实际驱动负载的数量是决定 TSC695F 处理器是否复位的重要因素, 负载一旦超过芯片驱动能力上限, 即容易造成数据装载出错, 从而造成复位发生; 最后通过仿真和实验验证了理论分析的正确性, 并提出了改进方案。

**关键词:** 异常复位; 驱动能力; 验证

## Research on Abnormal Reset of on-board Computer CPU Based on TSC695F Processor

Qiu Yuan<sup>1,2</sup>, Ji Binghua<sup>1,2</sup>, Shen Qi<sup>1</sup>, Liu Yichuan<sup>1</sup>, Wang Zhiguo<sup>1</sup>

(1. Shanghai Aerospace Electronics Technology Research Institute, Shanghai 201109, China;

2. Key Laboratory of Intelligent Computing Technology, Shanghai 201109, China)

**Abstract:** The problem of abnormal reset of CPU board in on-orbit flight test based on TSC695F processor was studied. By using the logic analyzer to capture process data and locate the problem location, found that the root cause of the on-board computer reset is the data loading error. Combined with the chip drive capability limit parameter analysis, it was found that the actual driving load of the CPU board is an important factor determining whether the TSC695F processor reset. Once the load exceeded the upper limit of the chip driving capability, the data loading error would easily occur, which could result in a reset. Finally, the correctness of theoretical analysis was verified by simulation and experiment, and an improved scheme was proposed.

**Keywords:** abnormal reset; driving capability; verification

## 0 引言

宇宙空间充满了等离子体、电离气体和各种能量的带电粒子等多种形态的物质, 而且还有各种波长的电磁辐射, 所以长期工作于宇宙空间的星载计算机与普通商用计算机不同, 比如在处理器选型上, 星载计算机多采用了航天专用的 CPU 芯片, 如 P1750、TSC695、TSC697、POWER-PC 等<sup>[1-3]</sup>。这些芯片内部大多设计了 EDAC 校验、冗余设计、错误处理、地址保护等高可靠性措施。其中 1750 系列星载计算机和基于 TSC695F 系列星载计算机技术已经相对成熟<sup>[4-7]</sup>。因此星载计算机多采用国产 P1750 套片和 TSC695F 处理器完成星上运算, 尤其是 TSC695F 处理器已经作为标准模块被大多卫星所直接采用<sup>[8-12]</sup>。

TSC695F 处理器性能优异、安全可靠, 已经被业界广泛认可。然而, 实际应用中关于 TSC695F 处理器驱动能力的研究资料却相对匮乏, 因此在工程应用中缺少必要的

参考<sup>[13-17]</sup>。

针对某型卫星星载计算机成品在模拟在轨飞行测试中, 基于 TSC695F 处理器 CPU 板出现异常复位现象这一问题, 展开研究分析, 以找到造成 CPU 板异常复位的原因, 提出解决异常复位问题的具体措施, 为后续星载计算机单机产品开发提供必要参考<sup>[18-19]</sup>。

## 1 系统结构及原理

发生异常复位的星载计算机处理器板用到的器件主要有 TSC695F、EEPROM、SRAM、FPGA、1553B 总线等。由于该电路板上元器件较多, 不宜直接与主控 CPU 芯片相连, 为了提高系统的稳定性, 保证 CPU 运行稳定可靠, 电路设计时已经考虑到将芯片的驱动能力进行一定的均衡分布<sup>[20-24]</sup>。具体电路拓扑结构如图 1 所示。CPU 直接驱动 7 个芯片, 分别是 1 片 32 位的 SRAM、1 片 8 位的 SRAM、4 片 EEPROM 和 1 片 FPGA。其余 6 片 PROM、1 片 1553B 总线和 1 片隔离驱动芯片通过 FPGA 间接被 CPU 控制<sup>[25-27]</sup>。

核心处理器 CPU 即是采用的 ATMEL 公司生产的 TSC695F 芯片, 该芯片是一款 32 位处理器, 芯片内部包含 EDAC 校验功能, 2 个异步串口, 5 个外部输入的中断控制器和 1 个 TAP 控制器用于 JTAG 接口的测试电路操作顺

收稿日期: 2020-06-14; 修回日期: 2020-07-14。

基金项目: 国家自然科学基金(61671037)。

作者简介: 邱源(1982-), 男, 上海人, 博士生, 高级工程师, 主要从事星载计算机设计方向的研究。

纪丙华(1992-), 男, 河南周口人, 研究生, 主要从事星载计算机设计方向的研究。

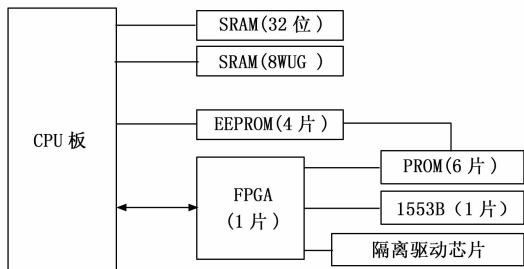


图 1 CPU 板系统框图

序。同时该芯片还提供直接访问 PROM 和 RAM 等内存的控制接口。

## 2 异常复位现象描述与分析

### 2.1 异常现象描述

基于 TSC695F 处理器的星载计算机在卫星总体单位进行接口测试时，按照既定的操作程序，单机正常运行了一段时间后，CPU 检测到单机 EDAC 错误中断，然后对软件执行软复位操作，随即 CPU 出现异常复位现象。为了故障复现，对星载计算机进行多次重复试验，单机正常工作一段时间后均会发生异常复位现象，并且复位发生前正常工作的时间从 4 分钟至数小时不等。

为了找到导致星载计算机异常复位的原因，提取 CPU 软件故障处理入口地址数据进行分析，发现造成错误的原因是硬件问题导致数据装载出错。针对数据装载出错的情况，TSC695 器件手册中给出了这类问题的具体说明，如表 1 所示。

表 1 TSC695F 器件手册关于数据装载错误故障说明

陷阱/错误	同步/异步	优先级	陷阱类型(11)	输出信号观测	注释
数据访问	同步	10	0x09	MEXC	数据装载错误： ① 控制/数据/地址总线奇偶校验错误；② 访问受保护或无效存储区；存储器不可纠正错误；③ 总线超时/错误；④ 系统寄存器访问冲突。

从表 1 可知，数据装载错误具体产生的原因有 4 种可能情况。为了进一步排查错误的具体原因，通过逻辑分析仪对数据时序关系进行捕捉分析，如图 2 所示。

由图 2 可知，MEXCb 信号是处理器检测到数据装载出错时的输出标识信号，即低电平表示数据发生错误。可以看到，发生错误的地址信号 MHOLDb 对应的起始地址为“02001940”。这个地址和 TSC695F 运用自身故障检测机制检出的错误地址完全一致。

发生错误时的程序代码如下所示：

```

200192c: 90 04 80 10 add %l2, %l0, %o0
2001930: 95 2a 20 02 sll %o0, 2, %o2
2001934: d2 05 40 0a ld [%l5 + %o2], %o1
2001938: d0 06 00 13 ld [%i0 + %l3], %o0
    
```

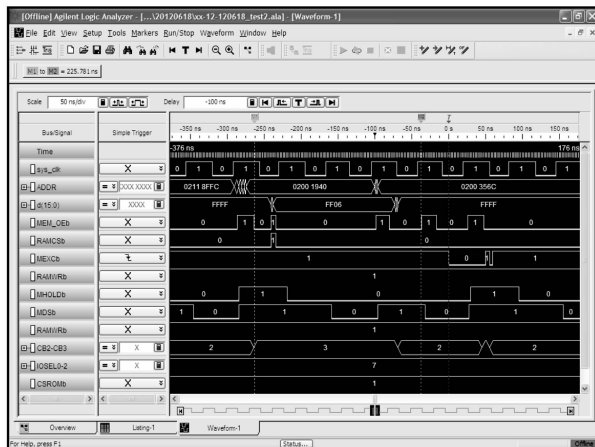


图 2 数据装载错误时序图

```

200193c: 80 a2 40 08 cmp %o1, %o0
2001940:22 80 00 06 be,a 2001958 <main+0x2a4>
2001944:a0 04 20 01 inc %l0
printf("Error at %9x*n", (int)sig + (i+k) * 4);
2001948:90 15 a2 c0 or %l6, 0x2c0, %o0
200194c:40 00 00 40 call 2001a4c <printf>
2001950:92 05 40 0a add %l5, %o2, %o1
    
```

通过对比程序的执行代码发现，正确的程序区地址和数据应是：

```
ADDR = 0X02001940 DATA(15:0) =0X0006;
```

但从逻辑分析仪看到却是：

```
ADDR = 0X02001940 DATA(15:0) =0X FF06。
```

因此可以推测数据装载错误是引起星载计算机异常复位的原因。

为了更准确地说明问题发生的原因，现对测试过程中地址和数据波形进行抓取，部分试验结果如图 3~4 所示。

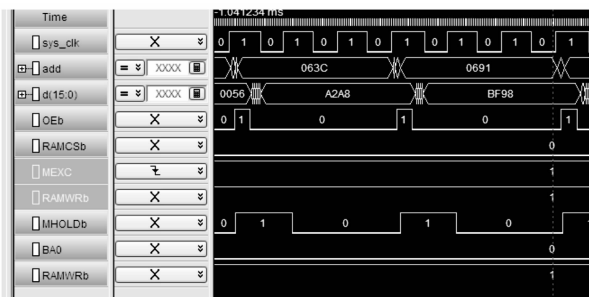


图 3 正确装载（地址 02000691，正确数据应为 BF98）

从图 3 中可以看到，在星载计算机正常工作时，地址变量 addr 为 0691 时对应的内存数据 data [15: 0] 是 BF98，处理器数据装载正确。当发生异常复位现象时，同一个地址捕捉到的内存数据如图 4 所示，读取地址变量 addr 为 0691 的内存数据，发现该地址内的数据变为 BF10，再次故障复现时该地址的数据又变为 B198。

通过多次的故障复现，总结出异常复位发生时 有 3 个



图 4 错误装载 (地址 02000691, 错误数据 BF10 和 B198, 正确数据应为 BF98)

特点:

1) 数据发生错误地址不固定。在多次问题复现试验过程中, 在“02001940”、“02000650”、“02000691”地址均发生过数据错误, 因此可排除是某个固定地址引起异常复位。

2) 发生错误的数不确定。在多次问题复现试验过程中, 发生的错误数据有“EE06” (正确应为“0006”), “FF00” (正确应为“0000”), “BF10” (正确应为“BF98”), “B198” (正确应为“BF98”) 等多种数据, 因此可确定不是某几个数据线错误造成异常复位。

3) 发生错误时 4 个基片不是全错。在进行多次问题复现试验时, 4 个八位数据均错的现象从未发生, 这就意味着 SRAM (32 位 SRAM 是由 4 个 8 位的 SRAM 组成) 中并非所有的基片都不可以正常工作, 因此 CPU 对 SRAM 访问时序没有问题。

综上所述, 可以断定 CPU 数据装载时, 数据位的多位误码, 导致了“存储器不可纠正的错误 (EDAC)”。最终导致 CPU 软件进入了错误故障模式, 而 CPU 软件对这类错误的处理方式就是让软件重新运行, 具体的现象就是星载计算机复位。

### 2.2 异常复位分析

基于以上分析, CPU 直接对 SRAM 进行访问控制, 在对 SRAM 读访问控制时序完全正确的情况下, SRAM 的其中一个基片却输出了错误数据, 因此可以推测 CPU 提供给 SRAM 读访问控制信号的驱动能力存在问题, 为了进一步分析探究异常复位的原因和机理, 现对与 CPU 处理器直接连接即由 CPU 驱动的芯片数量进行统计分析, 统计结果如表 2 所示, 表 2 给出了由 CPU 直接驱动的芯片种类与数量。由表 2 可知, CPU 直接驱动 7 片集成电路芯片, 其余 8 片则通过 FPGA 间接驱动。

表 2 CPU 板驱动芯片数量

序号	输出芯片名称	输入芯片名称	驱动片数	驱动总片数
1	CPU	EEPROM	4	7 片
2	CPU	SRAM (32 bit 和 8 bit 各一片)	2	
3	CPU	FPGA	1	
4	FPGA	1553B	1	8 片
5	FPGA	PROM	6	
6	FPGA	164 245	地址数据各 1	

### 2.3 驱动能力复算和波形分析

为了进一步探究 CPU 的驱动能力与负载的匹配性, 就需要结合手册分析这些与 CPU 直接相连的芯片输入电容。因此通过查阅与 CPU 直接连接的几种芯片的手册, 获取到相关芯片电容参数和计算结果如表 3 所示。由表 3 可知, CPU 负载输入电容值 (地址总线) 总数为 193pF, 输入/输出电容值 (数据总线) 总数为 103pF。

表 3 CPU 负载芯片电容表

序号	芯片名称	单片输入电容值 pF (地址总线)	驱动片数	总输入电容值 pF (地址总线)	单片输入/输出电容值 pF (数据总线)	总输入/输出电容值 pF (数据总线)
1	EEPROM	32	4	128	14	56
2	32 位 SRAM	45	1	45	25	25
3	8 位 SRAM	10	1	10	12	12
4	FPGA	10	1	10	10	10
5	总计			193		103

现以 FPGA 的输入/输出电容值 (10pF) 作为一个芯片的基本单位来估算驱动芯片数量。由于 SRAM, EEPROM 内部都为多基片封装, 内部数据为 8 位拼接, 所以对于 CPU 来讲可等效于驱动一个芯片, 因此, CPU 相当于驱动 10.3 个 FPGA 芯片的基本单元, 驱动能力可以满足要求。而对于地址线来讲, CPU 相当于驱动 19.3 个 FPGA 芯片的基本单元, 容性负载过大致使驱动能力不足。

在问题复现时用示波器抓到 SRAM 读操作时的地址线波形, 如图 5 所示。其中 C2 为 nOE 信号, 表示 CPU 发起一次读控制操作, 该信号为低电平有效; C3 和 C4 分别为 CPU 给出的地址线 A [9] 和 A [10], 可以看出 A [9] 和 A [10] 在本次读操作时均发生了地址跳变, 但 A [9] 的地址线爬坡过程相对平缓, 而 A [10] 则出现了一道坎, 表明地址线 A [10] 上的分布电容较大, CPU 的驱动电流不足, 因而出现地址跳变延时。同时, A [9] 和 A [10] 整个上升过程周期也比较长, 大约为 20 ns, 而 nOE 信号上升时间则只有大约 9 ns。

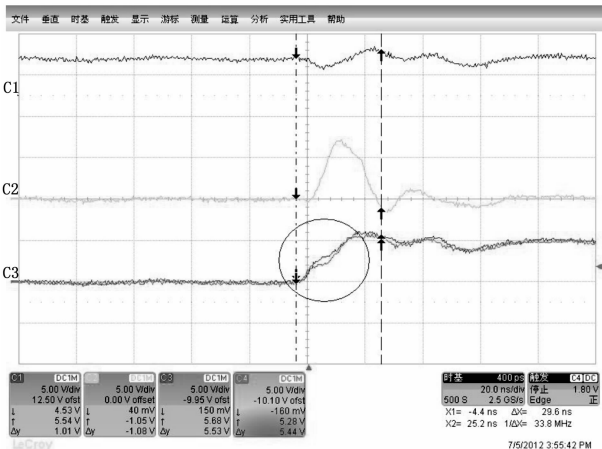


图 5 CPU 读操作时地址爬坡图

为了充分验证分析的合理性,同时对 CPU 板 PCB 进行了仿真。将 PCB 参数、分层结构、层厚、层介质参数等交给专业人员进行仿真,并对地址线、数据线进行复核核算。对 CPU 地址线 A [10] 的仿真结果如图 6 所示,地址线在上升和下降过程中均存在非单调变化的特性,这与实测波形基本一致。

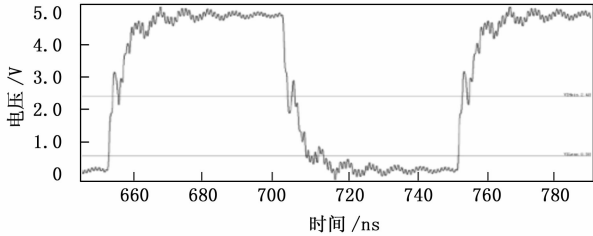


图 6 CPU 地址线 A [10] 仿真图

与此同时为了验证分析的合理性,拆除 EEPROM 后对 CPU 板多次长时间测试,均无异常复位现象发生,表明异常复位问题是由于地址线驱动不足引起的 SRAM 读操作异常引起的。

### 3 解决方案及验证

为了解决 CPU 板驱动能力不足的问题,需要对电路进行修改完善。在不改变处理器对各个负载芯片的控制逻辑条件下,要使电路能正常工作,则需要找到一种能够在处理器芯片和负载芯片之间充当驱动媒介的芯片。

经过选用元器件的初步筛选,发现规格为 B54ACS164245SRH 的多用途双向收发器具有双向异步通信、信号缓冲、电压转换以及冷备份电路保护的功能。其电路结构如图 7 所示,首先, B54ACS164245SRH 芯片的输入输出端都有单独的供电端口,具有输入阻抗高,驱动电流小的优势,并且可通过方向控制端 (DIR) 和使能控制端 (OE) 灵活配置芯片信息流方向 (信息流从 A 总线向 B 总线发送,或从 B 总线向 A 总线发送) 和和工作时间。此外,该规格芯片具备冷备份的功能 (Cold spare),当电源输入电压为零时,输入端 buffers 和输出端 buffers 处于至少 1 M 欧姆的高阻状态,因此 B54ACS164245SRH 还可作为冷备份保护电路,起到输入输出电气隔离和降低功耗等作用。

TSC695F 处理器的星载计算机 CPU 板异常复位问题是由 CPU 处理器驱动能力不足造成的,而规格为 B54ACS164245SRH 是采用 CMOS 工艺制作的一种具有 16 比特可双向传输数据的器件,具有输阻抗高,带负载能力强的优点。因此如果在 CPU 板和负载 EEPROM 之间通过使用 B54ACS164245SRH 间接通信,则就可以解决 CPU 板直接驱动负载过多,驱动能力不足的问题。

在 CPU 板和 4 片负载 EEPROM 芯片之间通过插入 B54ACS164245SRH 电路作为总线驱动器,系统电路如图 8 所示,通过利用 B54ACS164245SRH 输入阻抗高,带负载能力强的特点,实现 CPU 处理器驱动能力的扩展的作用,以提高系统的稳定性和可靠性。

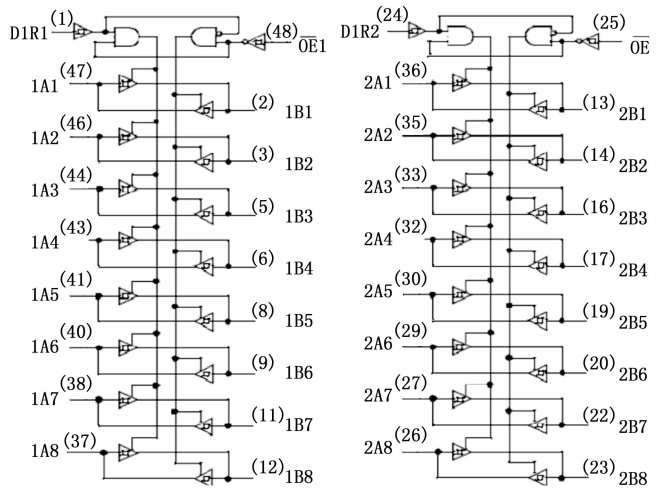


图 7 B54ACS164245SRH 电路结构图

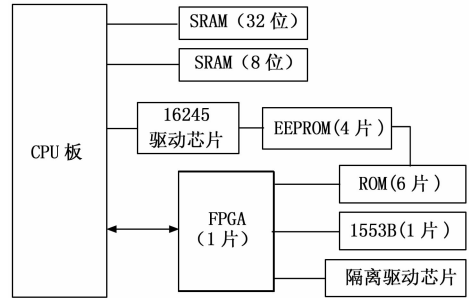


图 8 增加 164245 芯片后 CPU 板系统框图

经测算,修改后 CPU 驱动地址线相当于驱动 8 个 FPGA 芯片的基本单元,一片 164245 芯片驱动数据线相当于驱动 6.2 个 FPGA 芯片的基本单元,故而可以满足驱动能力的要求。

针对完善后的 CPU 处理器电路进行多次测试,均未发生异常复位现象。在星在计算机整机联测时,CPU 数据收发正常,未再发生数据装载错误的问题。

### 4 结束语

通过对基于 TSC695F 处理器的星载计算机 CPU 板异常复位问题进行研究,发现造成星载计算机单机的处理器板异常复位的问题是所挂负载超过了 CPU 的驱动能力。因此,为了提升星载计算机的可靠性,必须结合芯片驱动能力限度对电路综合考虑分析,实验证明通过加入 16 245 芯片可有效拓展 CPU 驱动能力,所以在一定程度上可以考虑通过加入驱动芯片拓展 CPU 的驱动能力解决驱动能力不足的问题。

#### 参考文献:

[1] 黎 黎,乔德治,何世民,等.星载计算机软件通用测试平台的设计与实现 [J]. 微电子学与计算机,2019,36 (3): 23-27.  
 [2] 李兴伟,白 博,周 军.基于 FPGA 的立方星可重构星载处理系统研究 [J]. 计算机测量与控制,2018,26 (8): 172-176.

- [3] 姜 宏, 杨孟飞, 于广良, 等. 一种面向星载计算机的功能级功耗估计方法 [J]. 空间控制技术与应用, 2015, 41 (3): 48-52.
- [4] 申 奥. 高可靠并行星载计算机软件容错技术研究 [D]. 上海交通大学, 2013.
- [5] 王 挥, 潘海燕, 沙李鹏. 嵌入式星载计算机故障注入系统 [J]. 计算机测量与控制, 2011, 19 (10): 2335-2336, 2343.
- [6] 高万里, 赵 诣, 刘 磊, 等. 基于 COTS SoC 的星载微型 GNSS 接收机硬件设计 [J]. 计算机测量与控制, 2019, 27 (1): 259-262.
- [7] 徐 楠, 李朝阳, 王兆琦, 等. 高轨卫星星载计算机优化设计与实现 [J]. 中国空间科学技术, 2020, 40 (1): 94-100.
- [8] 高万里, 赵 诣, 李向红, 等. 多模卫星导航星载终端仿真平台设计 [J]. 计算机测量与控制, 2019, 27 (3): 235-239.
- [9] 王 平, 孙 宁, 李华旺, 等. 创新一号小卫星星载计算机控制系统设计 [J]. 计算机工程, 2006 (18): 255-257.
- [10] 何小青. 基于 SOC 和 SIP 的微纳卫星星载计算机设计 [D]. 西安: 西安电子科技大学, 2018.
- [11] 孙兆伟, 刘 源, 邢 雷, 等. 面向多任务的可重构星载计算机设计 [J]. 系统工程与电子技术, 2011, 33 (6): 1407-1414.
- [12] 沈怡颺. 卫星控制系统软件可靠性设计及工程应用研究 [D]. 上海: 上海交通大学, 2016.
- [13] 王翠莲, 李 寅, 周 东, 等. 基于 FPGA 的星内高速路由实现方案 [J]. 计算机测量与控制, 2019, 27 (3): 164-167.
- [14] 钱 晨. 基于 SystemC 的微纳卫星星载计算机建模和故障注入研究 [D]. 南京: 南京理工大学, 2017.
- [15] 黎映相, 刘学文. 基于 PXI 总线的星载机仿真系统研究 [J]. 计算机测量与控制, 2019, 27 (12): 157-160.
- [16] 刘 鹏, 董 超, 杨政委, 等. 一种基于单片机和 FPGA 的星载定位指向控制器设计 [J]. 空间电子技术, 2016, 13 (6): 14-18.
- [17] 王远程. 星载计算机导星数据处理单元研究 [D]. 哈尔滨: 哈尔滨工业大学, 2016.
- [18] 关 宁, 田文波, 何津松, 等. 星载计算机模块间时钟同步方法 [J]. 计算机工程与设计, 2016, 37 (3): 629-634.
- [19] 胡志丹. 星载计算机中商用处理器容错关键技术研究 [D]. 长沙: 国防科学技术大学, 2009.
- [20] 王阳元. 微电子科学技术和集成电路产业技术讲座 (第五章): 我国集成电路产业和微电子科学技术发展的历史机遇和目标 [J]. 中国集成电路, 2003 (5): 90-93.
- [21] 吴汉鹏. 星载计算机系统单粒子效应仿真方法研究 [D]. 西安: 西安电子科技大学, 2019.
- [22] 高万里, 赵 诣, 李向红, 等. 多模卫星导航星载终端仿真平台设计 [J]. 计算机测量与控制, 2019, 27 (3): 235-239.
- [23] 钱 晨. 基于 SystemC 的微纳卫星星载计算机建模和故障注入研究 [D]. 南京: 南京理工大学, 2017.
- [24] 贾文涛, 张春元, 付 剑, 等. 一种高可靠双机温备星载计算机的设计与实现 [J]. 计算机研究与发展, 2010, 47 (S1): 127-132.
- [25] 包依勤. 面向电压监测终端的远程升级加密通讯方法 [J]. 计算机测量与控制, 2019, 27 (5): 178-182.
- [26] 李宏亮, 文 梅, 张春元, 等. 高可用实时系统中故障检测及故障恢复技术的研究 [J]. 计算机工程与科学, 1999 (6): 81-85.
- [27] 李 欣, 禹霁阳, 牛跃华, 等. 基于分级缓存加速的高可靠高速星载固存设计 [J]. 计算机测量与控制, 2019, 27 (10): 186-191.
- [19] Watson J D, Crick F H C. A structure for deoxyribose nucleic acid [J]. Nature, 1953, 171 (4356): 737-738.
- [20] Wu J H, Liao X F, Yang B. Image encryption using 2D Henon-Sine map and DNA approach [J]. Signal Processing, 2018, 153: 11-23.
- [21] Goldberger A L, Luis A N, Leon G, et al. Physiobank, physiokit, and physionet: components of a new research resource for complex physiologic signals [J]. Circulation, 2000 (101): 215-220.
- [22] Enayatifar R, Abdullah A H, Ismail F I, et al. Image encryption using a synchronous permutation-diffusion technique [J]. Optics & Lasers in Engineering, 2017, 90: 146-154.
- [23] Shalini S, Priti M, Piyush K S, et al. Fast and Secure Medical Image encryption based on non linear 4D logistic and DNA sequence (NL4DM-DNA) [J]. Journal of Medical Systems, 2019, 43: 267.
- [24] Sun S L. Chaotic image encryption scheme using two-by-two deoxyribonucleic acid complementary rules [J]. Optical Engineering, 2017, 56 (11): 1-10.
- [25] Belazi A, Abd E A A, Belghith S. A novel image encryption scheme based on substitution-permutation network and chaos [J]. Signal Process, 2016, 128 (11): 155-170.

(上接第 196 页)

- [12] Gong L H, Deng C Z, Pan S M, et al. Image compression-encryption algorithms by combining hyperchaotic system with discrete fractional random transform [J]. Optics and Laser Technology, 2018, 103: 48-58.
- [13] 李玉珍, 金 鑫, 赵 耿, 等. 基于 Zigzag 变换与混沌的彩色图像加密方案 [J]. 计算机工程与设计, 2016, 37 (8): 2002-2006.
- [14] 贺温平. 对基于 Zigzag 变换与混沌的彩色图像加密方案的破译 [J]. 计算机应用与软件, 2019, 36 (10): 323-329.
- [15] Wang X Y, Sun H H. A chaotic image encryption algorithm based on zigzag-like transform and DNA-like coding [J]. Multimedia Tools and Applications, 2019, 78: 34981-34997.
- [16] Narendra K P, Vinod P, Krishan K S. Diffusion-substitution based gray image encryption scheme [J]. Digital Signal Processing, 2013, 23 (3): 894-901.
- [17] Alireza J, Wu X W, Vallipuram M. Comments on the security of "diffusion-substitution based gray image encryption" scheme [J]. Digital Signal Processing, 2014, 32: 34-36.
- [18] Ye G D, Jiao K X, Pan C, et al. An effective framework for chaotic image encryption based on 3D logistic map [J]. Security and Communication Networks, 2018: 1-11.