



基于 FPGA 的 DRAM 存储器容错实验 教学设计

田 毅¹, 马世耀², 陈庭康², 张博文¹, 史春蕾¹

(1. 中国民航大学 安全科学与工程学院, 天津 300300; 2. 中国民航大学 中欧航空工程师学院, 天津 300300)

摘要: 在安全应用场景中, 数据的完整性备受关注。作为计算机系统的核心记忆元部件, DRAM 的容错能力对数据完整性起着关键作用。该文提出了一种基于缩短 RS 纠错码并融合数据交织的存储器多位数据错误容错方法, 并设计了教学实验。在基础实验中指导学生进行硬件实现, 以促进掌握 FPGA 设计技术; 在拓展实验中, 要求学生在控制器电路中部署设计实现的容错电路, 并加入系统总线接口模块、故障注入模块和数据交织模块, 促进学生掌握系统设计和验证技术。通过创新实验教学内容和方法, 帮助学生更好地理解和掌握机载电子系统的容错设计方法, 从而提高学生的创新能力和工程实践能力。

关键词: 容错; 故障注入; 动态随机存取存储器; FPGA

中图分类号: TP302

文献标志码: A

DOI: 10.12179/1672-4550.20240233

Teaching Design of Fault Tolerance Experiments for DRAM Memory Based on FPGA

TIAN Yi¹, MA Shiyao², CHEN Tingkang², ZHANG Bowen¹, SHI Chunlei¹

(1. College of Safety Science and Engineering, Civil Aviation University of China, Tianjin 300300, China; 2. Sino-European Institute of Aviation Engineering, Civil Aviation University of China, Tianjin 300300, China)

Abstract: In security application scenarios, data integrity is of paramount concern. As the core memory component of computer systems, the fault tolerance capability of DRAM memory plays a critical role in data integrity. This paper proposes a fault-tolerant method for multi-bit data errors in memory, based on shortened RS error-correcting codes and integrated with data interleaving, and a corresponding teaching experiment is designed. In basic experiments, students are guided in hardware implementation to promote their mastery of FPGA design techniques. In extended experiments, they are required to deploy the designed and implemented fault-tolerant circuit within the controller circuit, and add a system bus interface module, a fault injection module, and a data interleaving module to promote their mastery of system design and verification techniques. By innovating experimental teaching content and methods, students can better understand and grasp the fault-tolerant design methods of airborne electronic systems, thereby improving their innovative and practical engineering capabilities.

Key words: fault tolerance; fault injection; DRAM; FPGA

随着国产大型客机成功研制及运行的推动, 航空电子系统研制进入快速发展期, 这要求学生在掌握航空电子系统基本原理的基础上, 进一步熟悉机载复杂系统安全性的设计要求, 培养学生的综合设计和验证能力, 而以往的教学则缺乏

相关的综合实验内容。航空电子系统设计技术是航空宇航科学与技术、安全科学与工程、计算机科学与技术、软件工程、电子科学与技术等多个学科的交叉融合, 需要开发专门的综合实验来贯穿多领域的教学内容, 这也是“新工科”理念下

收稿日期: 2024-04-28

基金项目: 天津市高等学校研究生教育改革研究计划 (TJYG135); 中国交通教育研究会教育科学研究课题 (JT2022YB304); 中国民航大学教改项目 (CAUC-2023-B1-001)。

作者简介: 田毅, 硕士, 副研究员, 主要从事机载电子硬件与飞行器适航技术方面的研究。E-mail: ytian@cauc.edu.cn

课程教学改革的必然要求^[1]。

1 项目教学设计

1.1 项目意义

航空领域属于高安全领域,对于系统设计有安全性要求。存储器是航空电子系统通用的元部件,是系统运行的记忆载体,其实现多借鉴通用电子领域的设计方式,但需要针对航空应用进行专门的安全性设计^[2]。FPGA 是集成电路中的一种半定制电路,集成了大量的可配置逻辑块(configurable logic block, CLB)、数字时钟管理模块(digital clock management, DCM)、嵌入式存储块等资源,具有可配置的特点,是客户化机载电子硬件设计的实现方式之一^[3]。同时,随着现代航空电子系统设计复杂度的提升,基于仿真的验证耗时过长,不利于及时发现问题进行定位。为提升验证效率,利用现场可编辑逻辑门阵列(field programmable gate array, FPGA)容量大、可灵活配置的特点,构建模拟测试台成为硬件加速验证的有效方式。

实验教学依托已构建好的存储系统电路,以模块化方式推进教学工作。为深化学生对航空电子系统安全设计要求的理解,学生需要学习基本的容错算法,并由学生自己实现容错电路模块。此外,学生可进一步拓展设计控制器接口,实现存储系统与片上总线的互联,并通过搭建故障注入验证平台,培养学生设计、调试和验证的能力。

1.2 项目的目标

实验教学旨在培养学生掌握相关专业知识,针对复杂工程问题设计和开发解决方案,使用现代工具开展工作。同时,激发学生自主学习的兴趣,为未来从事机载电子系统研制工作打下坚实的基础。

2 项目教学设计过程

本项目的内容分为基础和拓展两个部分。基础部分注重设计研发,要求学生了解基本的存储架构、典型容错算法及计算原理,并且完成容错算法模块的硬件实现;拓展部分注重系统搭建,要求学生了解动态随机存取存储器(dynamic random access memory, DRAM)存储单元的典型失效机理,掌握存储控制器接口的操作方法,并将容错模块加入控制器,设计总线接口电路、故

障注入电路和数据交织电路,模拟存储器故障并对故障造成的数据错误进行纠正。

2.1 实验系统架构

实验台选择基于 1 片 FPGA 与 4 片第三代双倍数据速率同步动态随机存取存储器(double-data-rate three synchronous dynamic random access memory, DDR3 SDRAM)构成的系统架构,如图 1 所示。FPGA 主芯片为 AMD Xilinx 公司的 kintex-7 系列 XC7K325TFFG900 芯片,存储器为 16 位的 MT41K256M16 型 DDR3 芯片。FPGA 端由测试激励产生模块、DDR3 控制器和物理层接口组成,其中 DDR3 控制器包含了 AXI(advanced microcontroller bus architecture)接口模块、指令转换模块、错误检测与纠正(error detection and correction, EDAC)模块、故障注入模块等。基础实验中,学生需要设计 EDAC 模块中的编码器和译码器,教师应指导学生掌握 FPGA 设计流程和硬件描述语言;拓展实验中,学生需要设计新的电路替换原有的测试激励产生模块、AXI 接口模块、EDAC 模块、故障注入模块,教师需指导学生掌握系统设计和验证方法。由于 DDR3 的故障类型多样,故障注入模块设计具有较高的灵活性,因此实验也培养了学生的创新能力。

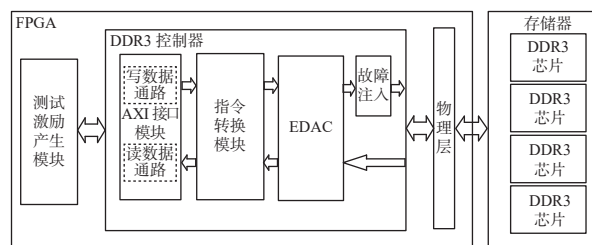


图 1 基于 FPGA 的存储实验台架构图

2.2 基础实验

2.2.1 典型容错方法

DRAM 存储器在实际航空应用中,在高空容易受到空间中的高能中子和质子影响,引发单粒子效应。DRAM 存储器的中子辐照试验表明^[4-5],由于存储器内部控制逻辑电路的临界电荷降低,控制逻辑部分对瞬态故障更加敏感,会呈现出行簇、列簇等独特的错误类型。在这些错误的影响下,最极限的情况是从一个存储芯片中读出的数据全部出错。因此在 EDAC 模块进行故障防护时,需要解决单芯片数据全部出错的问题,避免造成从存储芯片中读出的数据出错,引发系统失效。

存储器容错常采用EDAC技术^[6-7],传统的防护措施采用汉明码技术,可以检测并纠正一组数据中的一位错误,但是无法解决多位数据错误问题,而里德-所罗门(Reed-Solomon, RS)码作为多进制BCH码,可以纠正随机错误和突发错误,有效提高数据传输质量和存储器的可靠性,被广泛应用^[8-9]。

RS码采用多进制规则,编码和译码运算均在伽罗华域(Galois field, GF)中进行^[10]。通用编码参数 n 、 k 、 m 和 t 的含义如下:

1) n 表示一个RS码中码元的符号数,对于标准RS码 $n=(2^m-1)$;

2) k 表示有效的信息符号数;

3) m 表示码字中每个符号的比特数;

4) t 表示码字可以纠正的错误数, $t=(n-k)/2$ 。

在嵌入式存储系统中,数据位宽通常是32位。为了在存储系统中应用RS(n, k, m)码,需针对数据位宽对标准RS码进行缩减。具体来说,从标准RS码组中提取出所有高 λ 个符号为0的码字,组成缩短码集合。然后,从缩短码集合中删除每个码字的高 λ 个符号,从而构成缩短RS($n-\lambda, k-\lambda, m$)码组。同时,缩短RS($n-\lambda, k-\lambda, m$)码的纠错检错能力不低于对应的标准RS(n, k, m)码^[11]。

2.2.2 RS码编码器原理与设计

基于GF(2^4)域中的本原多项式^[12],如式(1)所示,构造缩短RS(8, 4, 4)的生成矩阵 G_s ,如式(2)所示,其中 α 为GF(2^4)域中的本原元。

$$P(x) = x^4 + x + 1 \quad (1)$$

$$G_s = \begin{bmatrix} 1 & 0 & 0 & 0 & \alpha^{10} & \alpha^{13} & \alpha^{13} & \alpha^{11} \\ 0 & 1 & 0 & 0 & \alpha & \alpha^{11} & \alpha^5 & \alpha^{11} \\ 0 & 0 & 1 & 0 & \alpha & \alpha^7 & \alpha^8 & \alpha^8 \\ 0 & 0 & 0 & 1 & \alpha^{13} & \alpha^6 & \alpha^3 & \alpha^{10} \end{bmatrix} \quad (2)$$

$u=[u_0, u_1, u_2, u_3]$ 为一组16位的信息符号,编码器以 u 作为输入,根据 $c=u \otimes G_s$ 求得32位的发送向量 $c=[u_0, u_1, u_2, u_3, r_0, r_1, r_2, r_3]$,其中 $\otimes$ 表示GF(2^4)域中的乘法运算。校验位 $r=[r_0, r_1, r_2, r_3]$ 的计算如式(3)所示,其中 $\oplus$ 表示GF(2^4)域中的加法运算。

$$\left. \begin{aligned} r_0 &= (\alpha^{10} \otimes u_0) \oplus (\alpha \otimes u_1) \oplus (\alpha \otimes u_2) \oplus (\alpha^{13} \otimes u_3) \\ r_1 &= (\alpha^{13} \otimes u_0) \oplus (\alpha^{11} \otimes u_1) \oplus (\alpha^7 \otimes u_2) \oplus (\alpha^6 \otimes u_3) \\ r_2 &= (\alpha^{13} \otimes u_0) \oplus (\alpha^5 \otimes u_1) \oplus (\alpha^8 \otimes u_2) \oplus (\alpha^3 \otimes u_3) \\ r_3 &= (\alpha^{11} \otimes u_0) \oplus (\alpha^{11} \otimes u_1) \oplus (\alpha^8 \otimes u_2) \oplus (\alpha^{10} \otimes u_3) \end{aligned} \right\} \quad (3)$$

学生使用硬件描述语言,结合式(3)完成编码电路的设计。设计完成的编码电路结构如图2所示,利用乘法器和加法器完成有限域中的 r_0 的计算,其余校验位计算的电路结构与之类似。

2.2.3 RS码译码器原理与设计

接收到的向量 y_s 可以表示为发送向量 c 和错误图样 e 之和,即 $y_s=c \oplus e$ 。如果错误图样 e 不为0,则向量 y_s 的伴随式 s 可以表示为 y_s 与缩短RS码校验矩阵 H_s 的转置相乘得到的值,即 $s=y_s \otimes H_s^T=e \otimes H_s^T$, H_s 如式(4)所示。在这种情况下,需要对接收向量 y_s 进行译码操作,以消除 e 对于信息符号的影响,从而恢复原始的信息符号 u 。

$$H_s = \begin{bmatrix} \alpha^{10} & \alpha & \alpha & \alpha^{13} & 1 & 0 & 0 & 0 \\ \alpha^{13} & \alpha^{11} & \alpha^7 & \alpha^6 & 0 & 1 & 0 & 0 \\ \alpha^{13} & \alpha^5 & \alpha^8 & \alpha^3 & 0 & 0 & 1 & 0 \\ \alpha^{11} & \alpha^{11} & \alpha^8 & \alpha^{10} & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

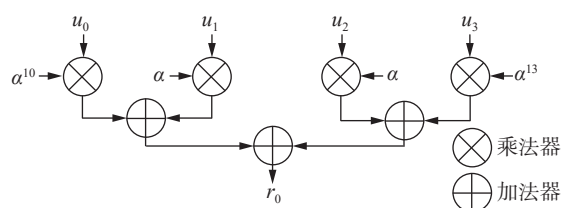


图2 编码电路结构图

对于存储系统,译码器设计时需要考虑译码延时、译码复杂度等问题,采用查找表译码算法^[13],并结合缩短RS码的特点进行改进。

缩短码译码时,需要借助对应标准码的查找表,为了便于描述,对标准码码元 y 的符号段作以下定义:设标准码码元 $y=[y_0, y_1, \dots, y_{n-1}]$,其中元素 $y_i \in GF(2^4)$;将 y 的后 $n-k$ 个符号 $y_p=[y_k, y_{k+1}, \dots, y_{n-1}]$ 定义为 y 的校验符号段。此外,定义 $y_r=[y_\lambda, y_{\lambda+1}, \dots, y_{k-1}]$ 为 y 的缩短码信息符号段, $y_l=[y_0, y_1, \dots, y_{k-\lambda-1}]$ 为 y 的缩短码辅助译码段。同时,定义 e_l 为 y_l 的错误图样, e_r 为 y_r 的错误图样, e_p 为 y_p 的错误图样^[14]。

针对 $n \geq 2k-\lambda$ 的情况,在缩短RS码译码时,利用 e_l 构造 T_L 查找表,如表1所示,其中 H 为标准RS码的校验矩阵,表中只包含 y_l 发生 δ 个错误的所有错误图样及其伴随式,其中 $1 \leq \delta \leq t-\beta-1$, $\beta=[t/2]$ 。若 $t-\beta-1 < 1$,译码时无需构造 T_L 表。构造 T_R 查找表,如表2所示,表中只包含 y_r

发生 γ 个错误的所有错误图样及其伴随式, 其中 $1 \leq \gamma \leq \beta$ 。

表 1 T_L 查找表

序号	错误图样	伴随式
1	e_1^1	$s_1 = [e_1^1, 0_{1 \times \lambda}, 0_{1 \times (n-k)}] \otimes H^T$
2	e_1^2	$s_2 = [e_1^2, 0_{1 \times \lambda}, 0_{1 \times (n-k)}] \otimes H^T$
$\vdots$	$\vdots$	$\vdots$
M	e_1^M	$s_M = [e_1^M, 0_{1 \times \lambda}, 0_{1 \times (n-k)}] \otimes H^T$

表 2 T_R 查找表

序号	错误图样	伴随式
1	e_r^1	$s_1 = [0_{1 \times \lambda}, e_r^1, 0_{1 \times (n-k)}] \otimes H^T$
2	e_r^2	$s_2 = [0_{1 \times \lambda}, e_r^2, 0_{1 \times (n-k)}] \otimes H^T$
$\vdots$	$\vdots$	$\vdots$
N	e_r^N	$s_N = [0_{1 \times \lambda}, e_r^N, 0_{1 \times (n-k)}] \otimes H^T$

y_r 和 y_p 分别对应 y_s 的缩短码信息符号段和校验符号段, 因此, y_s 的错误图样可表示为 $e = [e_r, e_p]$ 。若 e 的汉明重量 $\omega(e) \leq t$, 且 $1 \leq \omega(e_r) \leq \beta$, $[0_{1 \times \lambda}, e] \otimes H^T = s$, 则存在唯一 i ($1 \leq i \leq N$), 使得 $e_r = e_r^i$, $e_p = s \oplus s_i$, 满足 $\omega([e_r^i, s \oplus s_i]) \leq t$ ^[13]。

如果接收到的向量 y_s 的信息部分 y_r 发生至少 $\beta+1$ 个错误, 那么需要进行纠错处理。首先将 y_s 的高位补 λ 个 0 符号得到新的向量 $y_1 = [0_0, 0_1, \dots, 0_{\lambda-1}, y_{\lambda}, y_{\lambda+1}, \dots, y_{k-1}, y_k, \dots, y_{n-1}]$, 接下来将 y_1 循环右移 $k-\lambda$ 个符号 $y_2 = [y_{n-k+\lambda}, y_{n-k+\lambda+1}, \dots, y_{n-1}, 0_0, 0_1, \dots, 0_{\lambda-1}, y_{\lambda}, y_{\lambda+1}, \dots, y_{k-1}, y_k, y_{k+1}, \dots, y_{n-k+\lambda-1}]$, 此时 y_2 的 y_1 部分至多发生 $\beta-1$ 个错误, 可利用 T_L 进行译码。

以下给出 $n \geq 2k-\lambda$ 情况下, 基于查找表的缩短 RS 码通用译码算法伪代码。

input: y_s, H, T_L, T_R

$y_1 = [0_{1 \times \lambda}, y_s]$

$s = y_1 \otimes H^T$

$e = [0_{1 \times (k-\lambda)}, s]$

$i = 1$

while $\omega(e) > t$ and $i \leq N$ do

$e = [e_r^i, s \oplus s_i]$

$i = i+1$

end while

if $i \leq N$

then return $y_s \oplus e$

end if

$y_2 = [0_{1 \times \lambda}, y_s] \gg (k-\lambda)$

$s = y_2 \otimes H^T$

$j = 1$

while $\omega(e) > t$ and $j \leq M$ do

$e = [s \oplus s_j, e_1^j]$

$j = j+1$

end while

if $j \leq M$

then return $y_s \oplus e$

else return “failure”

end if

实验采用 RS(8, 4, 4) 码, 则 $t=2$, $\beta=1$, 译码时, T_R 表包含 y_r 发生 1 个错误的所有错误图样及其伴随式, T_L 表不包含错误图样和伴随式。实验中要求学生使用硬件描述语言按照伪代码开展设计。设计完成的译码电路结构如图 3 所示, 将 y_s 和经过循环右移的 y_s 输入到伴随式计算模块, 得到对应的伴随式 s 后, 将 s 分别输入比较电路 1 和比较电路 2。比较电路 1 完成译码算法中第一个 while 循环条件中的并行比较操作, 比较电路 2 完成译码算法中第二个 while 循环条件中的并行比较操作。错误图样和对应的伴随式输入到译码输出模块, 根据比较电路的结果完成译码输出。

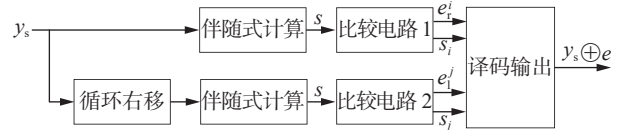


图 3 译码电路结构图

2.3 拓展实验

实验选用的 DDR3 存储器芯片位宽为 16 bit, 存储容量为 4 Gb, 有 8 个 bank, 用 18 根地址线进行寻址。DDR3 控制器管理着刷新、数据读写、激活、预充电等命令的发出^[15]。实验中提供了可编辑的 DDR3 控制器设计代码, 要求学生基于 AXI4 片上总线协议改进控制器接口电路, 实现存储器数据读写操作。要求学生梳理物理层和指令转换模块的数据通路接口逻辑, 添加故障注入模块。此外, 要求学生分析数据交织策略, 设计数据交织模块, 完成对极限错误的纠正。

2.3.1 DDR 控制器接口设计

实验中要求学生使用硬件描述语言按照 AXI4 接口协议规范开展设计。设计完成的 AXI4 接口结构如图 4 所示。AXI4 接口将主机与从机的通信接

口划分为5个独立的通道,分别为:读地址通道、读数据通道、写地址通道、写数据通道及写响应通道。5个独立的通道都使用一组握手信号来控制数据和控制信息的传输。通过读写控制状态机实现对命令和数据的缓存,命令和数据有序存入接收先入先出队列(first input first output, FIFO)和发送FIFO,供指令转换模块调用。

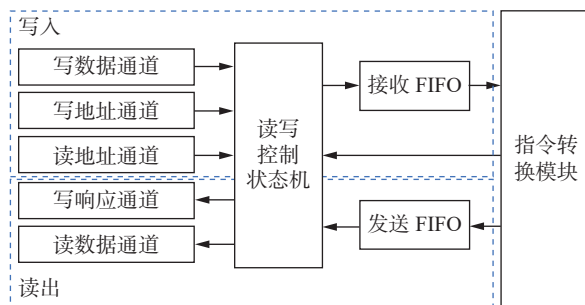


图4 AXI4接口电路结构图

2.3.2 故障注入模块设计

存储器故障类型多样,需要将故障造成的存储数据错误进行归类。实验中典型的数据错误形式包括单bit错误、多bit错误、单行错误、单列错误、单bank错误、多bank错误等^[16]。根据存储器数据错误的特点,用故障注入的方式模拟错误发生。

实验中给出的参考做法是对EDAC编码模块输出的编码数据进行改写。按照预先设定故障发生的地址,对数据进行改写,然后将改写后的数据写入存储器中,此时存储器中的数据即为发生错误的数据。在读取过程中,读到的数据是故障注入后的错误数据,将错误数据输入EDAC译码模块进行译码操作,实现错误的纠错。在实际实验中引导学生开展创新思考,以便提出更多的设计方法。

2.3.3 数据交织设计

对于4片16位存储芯片的存储系统,数据总线位宽为64bit,将信息编码写入存储系统,一个时钟沿并行写入64bit,其中32bit信息位,32bit校验位。缩短RS(8,4,4)码的纠错上限为2个符号错误。为纠正一片存储芯片中16位数据的错误,将32位数据位和对应的32位校验位采用如图5所示的交织方式存储在4片DRAM芯片中,图中 $u_0 \sim u_3$, $r_0 \sim r_3$ 为一组编码数据, $r_0 \sim r_3$ 为 $u_0 \sim u_3$ 的校验符号; $u_4 \sim u_7$, $r_4 \sim r_7$ 为一组编码数据, $r_4 \sim r_7$ 为 $u_4 \sim u_7$ 的校验符号。这样,即使单片存储

芯片出现故障,仍然可以通过其他芯片中存储的信息来消除故障,确保数据在读出时能够得到正确的纠正。基于此,需要设计数据交织模块对编码器输出的数据进行交织,以及数据解交织模块对译码器输入的数据进行解交织。

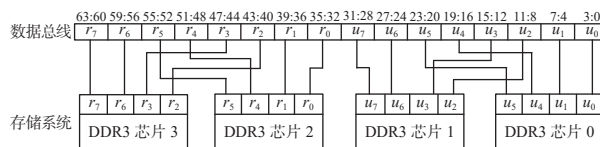


图5 数据交织状态图

DDR3控制器负责命令和数据的转换,物理层负责控制器与存储器之间信号时序的控制。物理层与控制器之间数据位宽是物理层与DDR3存储器之间数据位宽的两倍。EDAC电路结构如图6所示,利用4组RS(8,4,4)编码器和译码器实现EDAC模块设计。

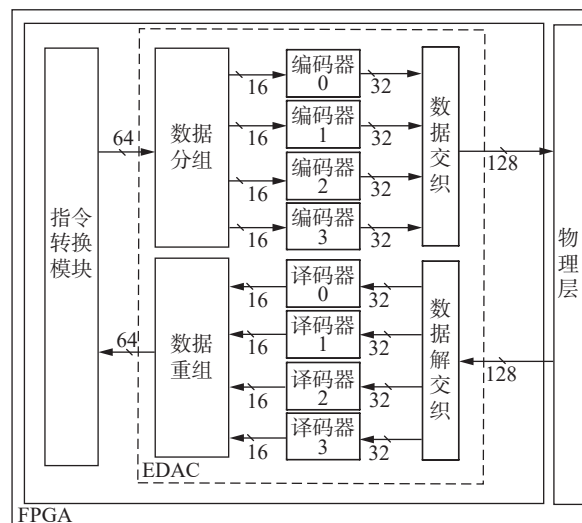


图6 EDAC电路结构图

3 实验验证

3.1 建立验证平台

实验使用Siemens EDA公司的Modelsim仿真工具和AMD Xilinx公司的Vivado工具完成实验的仿真验证和FPGA板级验证。利用Xilinx提供的DDR3存储器软模型,学生可以创建包含DDR3控制器和DDR3存储器模型的工程,使用Vivado和Modelsim联合仿真的方式对设计模块进行仿真验证。利用教学实验室中搭载了AMD Xilinx Kintex-7系列的xc7k325tffg900-2型FPGA开发板,借助集成逻辑分析器(integrated logic analyzer, ILA)抓取信号波形,对设计模块进行

FPGA 验证。

3.2 验证结果

仿真验证时, 利用测试激励产生模块, 通过 AXI 接口, 发送数据读写指令。当 AXI 发送写入指令时, 写入的数据有序进入 EDAC 模块进行数据编码。当 AXI 发送读出指令时, 读出的数据进入 EDAC 模块进行数据译码, 之后正确输出数据到 AXI 接口。EDAC 模块的编码器和译码器输入输出信号仿真波形如图 7 所示。encoder_in 为 64 位的编码输入信号, 输入数据值为 64'h0123456789abcdef。64 位数据分为 4 组 16 位的数据 16'h0123、16'h4567、16'h89ab、16'hcdef, 进入 4 个编码器, 输出 4 组 32 位编码数据 32'hbd0f0123、32'h36314567、32'h886089ab、32'h035ecdfe。

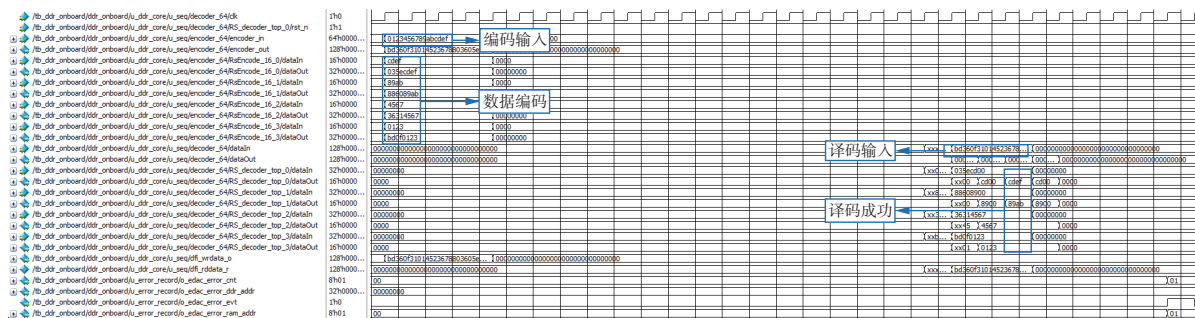


图 7 仿真验证波形图

仿真验证通过后, 利用 Vivado 工具对硬件设计进行编译, 加入 ILA 工具抓取关键波形信号, 启动开发板并烧写程序, 如图 8 所示, 利用 ILA 抓取到 FPGA 板级验证波形。

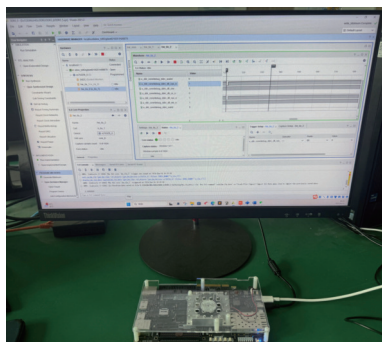


图 8 FPGA 板级验证实物图

4 结束语

按照课程培养目标的需要, 引入新的容错方法, 设计了基于 FPGA 的 DRAM 存储器容错教学实验。实验涉及的硬件模块包括 DDR3 存储系

4 组 32 位输出数据进行交织处理得到 128 位的 encoder_out 信号, 并将 encoder_out 输出到物理层。为了验证 EDAC 的功能, 通过故障注入将写入 DDR3 SDRAM 0 的数据改写为 0, 模拟出极限情况下单芯片数据全部出错。从 DDR3 SDRAM 0 中读出的数据经过解交织模块被分到了译码器 0 和译码器 1。输入译码器 0 的数据 32'h035ecd00, 经过译码器输出 16'hcdef; 输入译码器 1 的数据 32'h88608900, 经过译码器输出 16'h89ab, 数据错误被成功纠正, EDAC 设计正确; 编码器输入数据正确, AXI 接口功能正确; EDAC 输出数据进行了正确的交织, 数据交织设计正确; 译码器输入数据成功进行了数据改写, 故障注入设计正确。

统、EDAC 电路、故障注入电路等。其中, EDAC 模块包含了数据编码和译码模块, 针对缩短 RS 码, 提出通用查找表译码算法, 实现对行簇和列簇错误的纠正。故障注入模块可以模拟存储器故障导致的存储器数据错误, 从而直观地展现了存储器故障导致数据错误进而引发系统失效的过程。同时, 学生可深入了解存储体系结构, 熟悉硬件设计实现, 加深对机载系统安全性设计工作的理解, 巩固在航电系统设计时所必备的硬件设计和系统验证知识, 实现课程学习的目标。

参考文献

- [1] 田毅, 刘畅, 马世耀, 等. 基于工程能力培养的“航空电子系统设计技术”课程建设研究[J]. 科技风, 2022(18): 137-139.
- [2] FAA. Commercial off-the-shelf airborne electronic hardware assurance methods-phase 3-commodity memories: DOT/FAA/TC-16/40[R]. Washington: FAA, 2017.
- [3] 张华忠. 基于民航应用特色的数字电子技术研究[J].

数字技术与应用, 2017, 35(12): 73-74.

- [4] BAK G, LEE S, LEE H, et al. Logic soft error study with 800-MHz DDR3 SDRAMs in 3x nm using proton and neutron beams[C]//2015 IEEE International Reliability Physics Symposium. Monterey: IEEE, 2015: SE. 3.1-SE. 3.5.
- [5] PARK M, JEON S, BAK G, et al. Soft error study on DDR4 SDRAMs using a 480 MeV proton beam[C]//2017 IEEE International Reliability Physics Symposium (IRPS). Monterey: IEEE, 2017: SE-3.1-SE-3.6.
- [6] 贺振江, 刘曦, 王小珂. DDR 存储器单粒子翻转试验及加固设计研究进展[J]. 微电子学与计算机, 2022, 39(10): 111-117.
- [7] ASTHANA P, KHANNA G, SANKHYAN S, et al. Design of reed-Solomon encoder for error detection in DRAM cells[C]//AIJR Proceedings. New Delhi: AIJR Publisher, 2021: 551-556.
- [8] 胡涛, 马秀荣. 新型 RS-BCH 级联码[J]. 电光与控制, 2023, 30(5): 58-60.
- [9] 齐艺珂. 存储器抗单粒子效应二维码与 RS 码的研究[D]. 绵阳: 西南科技大学, 2019.
- [10] 刘梦欣. 基于 FPGA 的 RS 编译码研究与设计[D]. 太原: 中北大学, 2020.
- [11] SHU L, DANIEL J. COSTELLO. 差错控制编码[M]. 晏坚, 何元智, 潘亚汉, 译. 北京: 机械工业出版社, 2007.
- [12] 孙正. 基于 RS 码的抗多位翻转 SRAM 设计[D]. 哈尔滨: 哈尔滨工业大学, 2011.
- [13] 李梅, 包小敏, 武登杰, 等. RS 码的查表译码算法[J]. 西南大学学报(自然科学版), 2015, 37(11): 170-174.
- [14] 刘爱莲. 纠错编码原理及 MATLAB 实现[M]. 北京: 清华大学出版社, 2013.
- [15] 唐成武. 基于 AXI4 总线的 DDR3 高速存储接口系统设计[D]. 武汉: 华中科技大学, 2020.
- [16] SRIDHARAN V, LIBERTY D. A study of DRAM failures in the field[C]//SC' 12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. Salt Lake City: IEEE, 2012: 1-11.

编辑 葛晋

(上接第 110 页)

- [2] 李双寿, 杨建新. 金属工艺学实习教材[M]. 5 版. 北京: 高等教育出版社, 2023.
- [3] 马鹏举, 王亮, 胡殿明. 工程实践教学现状分析与对策研究[J]. 高等工程教育研究, 2011(1): 143-147.
- [4] 高琪, 李颖, 张飞. 基于工程能力培养的“金工实习”教学改革[J]. 实验室研究与探索, 2015, 34(1): 234-237.
- [5] 焦波. 美国威斯康星大学麦迪逊分校《能源系统实验》课程的特点与启示[J]. 创新创业理论研究与实践, 2021, 92(20): 49-52.
- [6] 栗茂林, 崔琦, 宋超英, 等. 项目驱动式教学在机械制造实习课程中的探索实践[J]. 中国现代教育装备, 2020(21): 90-93.
- [7] 李津津, 叶佩青. 新工科背景下贯通式项目制研究型综合实践教学模式探讨[J]. 中国大学教学, 2020(10): 58-61.
- [8] 张海光, 何岚岚, 李文彬, 等. 面向大规模学生在线实践直播教学模式探究: 以“云上金工实习”为例[J]. 实验室研究与探索, 2023, 42(8): 225-230.
- [9] 叶回春, 沈连娟. 当前先进制造技术下的工程实践教学研究[J]. 实验技术与管理, 2018, 35(4): 37-40.
- [10] PULLEN K R. Flywheel energy storage, in storing energy[M]. 2nd ed. Amsterdam: Elsevier, 2022.
- [11] 王明菊, 王辉. 飞轮储能的原理及应用前景分析[J]. 能源与节能, 2021(4): 27-28.
- [12] 薛飞宇, 梁双印. 飞轮储能核心技术发展现状与展望[J]. 节能, 2020, 39(11): 119-122.
- [13] AMIRYAR M E, PULLEN K R. A review of flywheel energy storage system technologies and their applications[J]. Applied Sciences, 2017, 7(3): 286.
- [14] 赵萌, 杜平, 赵天骄, 等. 基于实践教学的立式飞轮储能装置设计应用[J]. 机电工程技术, 2022, 51(9): 10-14.
- [15] 赵萌, 杜平, 张秀海, 等. 新型磁悬浮飞轮储能实验装置的设计及应用[J]. 现代机械, 2023(3): 11-15.

编辑 钟晓