

基于地址分割的嵌入式存储器内建自修复方法

俞 洋, 李嘉铭, 乔立岩

(哈尔滨工业大学自动化测试与控制系, 黑龙江哈尔滨, 150001)

摘 要: 内建自修复技术是一种有效修复嵌入式存储器中失效单元的方法. 在传统的内建自修复过程中, 需要对故障地址进行多次的读写操作, 功耗比较大. 本文提出了一种基于地址分割的嵌入式存储器内建自修复方法. 该方法将故障地址分割成两部分, 对 BIRA 内部存储器的访问分两个步骤进行, 有效简化了地址比较过程, 降低了功耗. 仿真试验表明, 本文方法能够在实现存储器故障自修复同时显著降低修复与工作过程中产生的功耗.

关键词: 内建自修复; 内建自测试; 内建冗余分析

中图分类号: TP391.76 **文献标识码:** A **文章编号:** 0372-2112 (2010) 2A-169-05

Memory Built-In Self-Repair Method Based on Address Partitioning Strategy

YU Yang, LI Jia-ming, QIAO Li-yan

(Department of Automatic Test and Control, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China)

Abstract: To increase the memory reliability, the technology of memory built-in self-repair is growing rapidly. During the course of repairing, the BIRA SRAM has to be accessed continually, therefore the power dissipation is dramatically high. To solve this problem, a memory built-in self-repair method based on address partitioning strategy has been proposed. This method can decrease the power dissipation by simplify the address comparison course. Experimental results show that the proposed method can repair the defective cells in embedded memories and the power dissipation can be decreased simultaneously.

Key words: built-in self-repair; built-in self-test; built-in redundancy analysis

1 引言

随着深亚微米工艺到纳米工艺技术的发展, 以 IP 复用为主要设计方法的 SoC 技术得到了快速发展, 嵌入式存储器核被广泛用于 SoC 和其他超大规模集成电路的设计中. 根据 2005 年国际半导体协会公布的半导体技术发展指南 (International Technology Roadmap for Semiconductors, ITRS), 2001 年嵌入式存储器占 SoC 面积的 45%, 到 2014 年将提升到 94%^[1]. 由于 SoC 芯片中集成的存储器数目增多将使芯片层数增多、制造工艺更复杂并且增大了单元密度, 片上存储块的失效率将明显增大, 这成为 SoC 芯片成品率下降的一个主要原因. 器件复杂度的增加将使 SoC 的单位价格变得越来越昂贵. 如果由于片上存储模块存在部分失效单元就报废整个 SoC, 显然是浪费且不合理的, 因此如何修复存储器中的失效单元成为提高 SoC 成品率的关键技术之一.

为解决这一问题, 在早期的研究中, 人们通常采用外部测试设备定位故障位置, 并用激光和 FUSE 的方法

加以修复. 进入 SoC 阶段后, 由于芯片工艺流程的日益复杂以及这些方法在速度、成本、修复能力上的不足, 其应用受到很大的限制. 嵌入式存储器自修复 (MBISR, Memory Built-In Self-Repair) 技术由于强大的灵活性, 不仅能够修复制造故障, 而且能够修复应用现场中出现的故障, 已经逐渐成为保证存储器成品率 (提高约 2 - 10%) 和产品上市时间的事实标准^[2].

所谓存储器修复技术就是采用预置的冗余存储单元替代已检测到的故障存储单元, 从而完成修复目的的. 根据冗余单元形式的不同, 嵌入式存储器的修复策略主要有三类: 基于冗余行的修复策略、基于冗余列的修复策略和基于冗余字的修复策略. 文献[3]提出了对冗余列分段的软修复方法, 把冗余列分段, 细化冗余资源粒度, 在很大程度上提高了修复率, 但该方法需要较多的冗余列才能提高修复率, 面积开销较大. 文献[4]提出二维冗余修复方法, 在修复过程中采用冗余行和冗余列, 大大提高了修复效率和灵活性, 但也增加了冗余分析算法的复杂度. 文献[5]提出了基于冗余字的修复方

法,使用存储器同类型的冗余字作为冗余单元替换故障字,该方法具有较高的修复效率且简单容易实现,但该方法不能修复/列译码逻辑故障.文献[6]提出了一种将基于内容可寻址存储器作为冗余字替换故障字的方法,对高失效密度存储器具有较好的修复效果.文献[7]利用 LFSR 作为地址生成器设计了一种存储器内建自测试电路.

上述方法对于存储器都有较好的修复效果,但在修复和工作过程中,都要对用于保存故障地址的 BIRA (Built-In Redundancy Analysis)内部存储器进行多次的读/写访问,因此在修复和工作的过程中产生的功耗是相当高的.为了解决这个问题,本文提出了一种基于地址分割的嵌入式存储器内建自修复方法,将故障地址分割成两个部分,在存储器修复和工作过程中,对 BIRA 内部存储器的访问分两个步骤进行,有效降低了对 BIRA 内部存储器的访问次数,降低了功耗.

2 存储器内建自修复原理

一个完整的存储器内建自修复系统主要包括内建自测试(BIST, Built-In Self-Test)和内建冗余分析(BIRA, Built-In Redundancy Analysis)两个部分.其中,BIST 模块用于检测存储器中的故障单元并输出故障单元地址,BIRA 模块用于重构故障地址.典型的存储器内建自修复系统结构如图 1 所示.

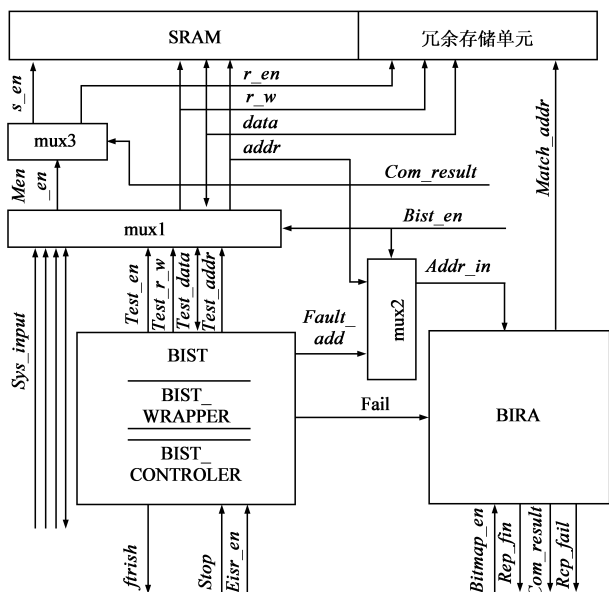


图1 存储器内建自修复系统结构框图

存储器内建自修复系统有两种工作模式:测试/修复模式和正常工作模式.在测试/修复模式下,系统上电启动,BIST 模块开始对待测存储器进行测试,在检测到故障单元后,BIST 模块给 BIRA 发送故障信号 Fail,将故障地址将送入 BIRA 模块,BIST 模块暂停检测.BIRA

模块接收到 Fail 信号后,对故障地址进行分析,判断该地址是否已经保存在 BIRA 内部存储器中,如果未存储,则在 BIRA 内部存储器的最末端保留该地址,并进行冗余分配,修复完成之后启动 BIST 继续进行测试;如果已经该故障地址已经存在,则继续进行 BIST.在正常工作模式下,BIST 电路处于不工作状态,要访问的地址直接被送入 BIRA 模块中,由 BIRA 模块判断是否为故障地址,若是则进行地址重映射,调用冗余存储单元,若不是则直接访问正常存储单元.

根据冗余单元形式的不同,嵌入式存储器的修复策略有三类:基于冗余行的修复策略、基于冗余列的修复策略和基于冗余字的修复策略^[8].随着嵌入式存储器容量的增加,故障单元的分布越来越分散,基于整行或者整列的修复会对系统开支造成很大负担,因此本文选用基于冗余字的修复策略.在每一次上电时,首先对存储器进行检测,如果检测到故障就产生修复信号,对嵌入式存储器故障地址进行重构,如图 2 所示.图中是一个 16 × 4 位存储器,当检测到存储单元 W1 有故障时,首先对该单元进行修复,即进行地址的重新映射,调用冗余单元 SW1 替换 W1.修复完成后,当对该存储器进行访问时,首先检测当前地址是否为故障已修复地址,若不是则访问 SRAM;若是则访问冗余存储单元.

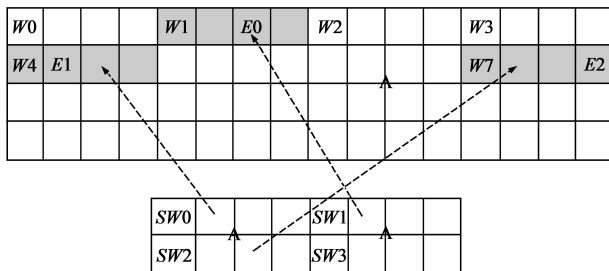


图2 基于冗余字的嵌入式存储器修复策略

3 基于地址分割的自修复技术

在存储器内建自修复的过程中,BIRA 模块是 MBISR 电路的核心部分,是在内建自测试电路基础上添加的一部分修复电路,通过将检测到的故障地址进行重构,以实现 MBISR 整体电路的离线修复功能.

BIRA 模块中最主要部分为一个内部存储器,用于存储 BIST 检测到的故障地址.在测试/修复过程中,当 BIST 模块检测到故障时,BIST 模块暂停,故障地址被送入 BIRA 模块.在 BIRA 模块的内部存储器中,新的故障地址与已经保存的故障地址逐一比较,以判断该地址是否已经被保存,即该故障是否已经被修复.在正常工作过程中,要访问存储单元的地址被直接送入 BIRA 中,并且与 BIRA 内部存储器中存储的故障地址逐一比较以判断当前要访问的存储单元是否存在故障.若有

输出信号包括:

Rep_fin : 表示修复结束;

Rep_fail : 表示冗余单元已经耗尽, 修复失败;

Com_result : 表示片选信号, 当系统访问地址为故障地址时置 1, 访问冗余存储单元;

R_addr : 指出该故障地址由 BIRA 模块重新分配的冗余存储单元地址.

由于采用了图 4 的电路结构, 只是在 BIRA 内部存储器的每个字前增加了一位, 与常见的基于 CAM 存储器的 BIRA 模块相比, 片上面积开支更小. 当地址进入 BIRA 模块内时, 首先只对标志位为 1 的已存地址前 $n - k$ 位进行比较, 当不相同则输入下一地址, 有相同对前 $n - k$ 位相同的地址的后 k 位进行比较. 而常见的基于 CAM 存储器的 BIRA 方式只是将 n 位访问地址顺序的与已存地址经比较, 可以看出, 文中所提出的方案可大大降低 BIRA 过程中的功耗.

4 实验结果与分析

为验证上述 MBISR 电路的正确性, 我们通过仿真实验进行验证. 为检测其电路的故障检测能力并减少测试时间, 首先对一个 16×8 位的存储器进行仿真, 预置 8 个冗余存储单元. 随机注入 0 到 8 个故障来检测存储器的自修复能力, 结果证明了存储器是能够修复这些故障的, 又通过注入 8 个以上的故障, 结果证明了存储器是可以报错的. 因此, 当故障单元的数量不大于冗余单元的数量时, 本文方法能够提供 100% 的修复率.

为验证 BISR 模块在功耗方面的优点, 这里继续进行下列实验, 设待测存储器为 $128k \times 8$ 的存储器, 冗余单元 128 个. 此时存储器访问地址为 17 位地址, 冗余访问地址 7 位. 根据地址分割 BIRA 模块原理, 将这 17 位地址分割成 7 位和 10 位两部分. 实验中采用泊松分布模型分三次随机向存储器注入了 10 个、20 个和 50 个字故障, 即故障位置是随机的, 但故障分布采用泊松分布模型, 这与实际 SRAM 故障的分布情况比较接近. BIST 测试算法采用经典的 March C-算法, 算法描述如下:

$$\{ \parallel (W0); \uparrow (R0, W1); \uparrow (R1, W0); \downarrow (R0, W1); \downarrow (R1, W0); \parallel (R0) \}$$

其中: \uparrow 表示对存储器地址进行顺序操作, \downarrow 表示对其进行逆序操作. 该算法可有效检测到固定故障、跳变故障和耦合故障等, 是一种比较完备的 March 算法.

为保证可以在系统工作频率下进行测试, 设系统 BIRA 模块频率为 200MHz, 系统工作频率为 1MHz 对存储器进行测试. 表 1~3 分别给出了 10 个故障、20 个故障和 50 个故障时, 本文方法与传统的基于 CAM 的 BIRA 方法的地址比较次数. 从表中可以看出, 文中所采

用的新型 BIRA 模块可以大大降低存储器测试过程中故障地址的比较次数以及正常工作时访问地址的比较次数, 这样有效的降低了测试时的功耗.

表 1 本文方法与传统方法的故障地址比较次数比较 (10 个故障)

	本文方法	基于 CAM 的 BIRA
测试/修复模式	36	45
正常工作模式	7	10

表 2 本文方法与传统方法的故障地址比较次数比较 (20 个故障)

	本文方法	基于 CAM 的 BIRA
测试/修复模式	146	190
正常工作模式	15	20

表 3 本文方法与传统方法的故障地址比较次数比较 (50 个故障)

	本文方法	基于 CAM 的 BIRA
测试/修复模式	855	1275
正常工作模式	35	50

5 结论

在嵌入式存储器的自修复过程中, 需要对用于存储故障地址的 BISR 内部存储器进行多次的读/写访问, 因此在修复和工作的过程中产生的功耗是相当高的. 为了解决这个问题, 本文提出了一种基于地址分割的嵌入式存储器内建自修复方法. 该方法将地址分成两个部分进行比较, 并增加标志位来减少某些不必要的地址比较过程, 因此有效降低了功耗. 仿真实验表明, 该方法能够自动修复一定数量的存储器故障, 并且能对超出其修复范围时能够报错; 故障地址的比较次数较传统的基于 CAM 的自修复方法有明显的降低, 有效降低了测试和工作过程中的功耗, 从而提高了整个存储器及修复电路的可靠性. 在硬件上, 本文方法只在 BIRA 内部存储器中为每个字增加了一个存储单元, 与常见的基于 CAM 存储器的 BIRA 过程相比, 片上面积开支更小. 因此基于地址分割的嵌入式存储器内建自修复方法是有效的存储器修复方法, 具有一定的实用价值.

参考文献:

- [1] Semiconductor Industry Association (SIA). International technology roadmap for semiconductors (ITRS) [DB/OL]. [Http://www.sia-online.org/home.cfm](http://www.sia-online.org/home.cfm), 2005.
- [2] 陈国斌. 基于 SoC 的嵌入式存储器故障修复策略[J]. 半导体技术, 2006, 31(4): 260 - 263.
Chen Guo-bin. Embedded memory fault repair strategies based on SOC[J]. Semiconductor Technology, 2006, 31(4): 260 - 263.
- [3] 付祥, 王达, 李华伟, 胡瑜, 李晓维. 一种嵌入式存储器的内建自修复机制[A]. 第四届中国测试学术会议论文集[C]. 河北秦皇岛, 2006. 15 - 19.

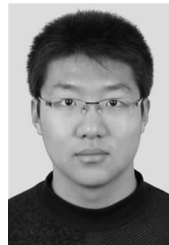
- [4] Jin-Fn Li, Jen-Chieh Yeh, Rei-Fu Huang, Chen-Wen Wu. A built-in self-repair design for RAMs with 2-D redundancy[J]. IEEE Transactions on VLSI Systems, 2005, 13(6): 742 - 745.
- [5] K Sawada, T Sakurai, Y Uchino, K Yamada. Built-in self repair circuit for high density ASMIC[A]. Proceedings of the IEEE Custom Integrated Circuits Conference[C]. San Diego, CA, 1989. 26. 1/1 - 26. 1/4.
- [6] 谢远江, 王达, 胡瑜, 李晓维. 利用内容可寻址技术的存储器 BISR 方法[J]. 计算机辅助设计与图形学报, 2009, 21(4): 467 - 473.
Xie Yuanjiang, Wang Da, Hu Yu, Li Xiaowei. Memory BISR based on content addressable memory[J]. Journal of Computer-Aided Design & Computer Graphics, 2009, 21(4): 467 - 473.
- [7] 王丽, 施玉霞, 王友仁. 一种嵌入式存储器内建自测试电路设计[J]. 计算机测控与控制, 2008, 16(5): 624 - 626.
Wang Li, Shi Yuxia, Wang Youren. Design of embedded memory built-in self-test circuits [J]. Computer Measurement & Control. 2008, 16(5): 624 - 626.
- [8] Shyue-Kung Lu, Chun-lin Yang, Han-wen Lin. Efficient BISR techniques for word-oriented embedded memories with hierarchical redundancy[A]. Proceedings of the 5th IEEE/ACIS In-

ternational Conference on Computer and Information Science and 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse [C]. Honolulu, HI, 2006. 355 - 360.

作者简介:



俞 洋 女, 1979 年生于黑龙江省大庆市, 哈尔滨工业大学自动化测试与控制系讲师, 博士. 主要研究方向为 SoC 测试与故障诊断技术等. E-mail: yuyanghit53@163.com



李嘉铭 男, 1986 年生于黑龙江省哈尔滨市, 在读硕士生. 主要研究方向为嵌入式存储器可测性设计与修复技术.