

一种双游程编码的测试数据压缩方案

商进^{1,2}, 张礼勇¹

(1. 哈尔滨理工大学 测控技术与通信工程学院, 黑龙江 哈尔滨 150080;

2. 黑龙江工程学院 电子工程系, 黑龙江 哈尔滨 150050)

摘要: SOC 芯片测试中一个主要的挑战就是处理大量的测试数据. 为了减少芯片测试中的测试数据, 提出了一种双游程的编码方案, 采用变长到变长的编码方式对 0 游程和 1 游程进行编码. 该算法在编码时同时考虑 0 游程和 1 游程, 大大减少了测试数据中短游程的数量, 同时文中给出了一种基于有限状态机的解压缩算法的实现方案. 理论分析和实验结果证明该方案具有高压缩率、硬件实现简单等特点.

关键词: 测试数据压缩; 解压; 双游程编码

中图分类号: TP391.76 **文献标志码:** A **文章编号:** 1007-2683(2010)04-0019-04

Scheme of Test Data Compression Based on Dual-run-length Code

SHANG Jin^{1,2}, ZHANG Li-yong¹

(1. School of Measure-control Technology and Communication Engineering, Harbin University of Science and Technology, Harbin 150080, China;

2. Department of Electronic Engineering, Heilongjiang Institute of Technology, Harbin 150050, China)

Abstract: One of the major challenges in testing a system-on-a-chip is dealing with a large volume of test data. To reduce the volume of test data, dual-run-length code is proposed in this paper, a variable-to-variable run-length code based on encoding both runs of 0s and 1s. Both runs of 0s and 1s are considered in this method so as to reduce the number of short runs, and the circuit structure of decoder with FSM is proposed. Theoretical analysis and experimental results show that this schemes is a very efficient compression method.

Key words: test data compression; decompression; dual-run-length code

0 引言

随着科技的进步, 集成电路的集成度有了很大的提高, 在单个芯片上集成的晶体管数目越来越多. 集成度的提高导致测试芯片时的测试数据急剧增加. 在存储模式测试中, 测试数据需要存储在自动测试仪器(ATE)中, 然后再传递到测试电路上. 大量的测试数据不仅增加了测试时间而且也超出了自动测

试仪器的内存容量, 使测试数据难以全部存储在自动测试仪器有限的内存中. 同时, 随着测试速度、带宽和内存的增加, 自动测试仪器的成本也急剧增加. 因此, 为了减小测试时间和测试成本, 有必要减少测试数据量.

测试数据的减少可以通过内建自测试(Built-in Self Test, BIST)、测试紧缩(Test Set Compaction, TSC)和测试数据压缩技术(Test Data Compression, TDC)实现. BIST 技术是通过芯片本身所带有的测

收稿日期: 2010-02-11

基金项目: 黑龙江省教育厅科学技术研究项目(10551Z0007)

作者简介: 商进(1974—), 男, 博士研究生, E-mail: shjin@Sohu.com;

张礼勇(1939—), 男, 教授, 博士生导师.

试模式生成器,直接在芯片上生成测试向量,但芯片需要进行预先设计以便为 BIST 测试做好准备. 测试紧缩技术通过紧缩部分带有关位 (Don't care bits) 测试立方,来减少测试向量的数量,但该技术对非模型的物理故障检测率较低. 测试数据压缩技术用压缩的形式在自动测试仪器中存储测试数据,不需要了解被测设计 (Design Under Test, DUT) 的内部结构,可以很好的保护知识产权,得到了广泛应用.

编码压缩方法是测试数据压缩最常用的方法之一. 这种技术的代表有 Golomb 码^[1],统计码^[2],游程码^[3],霍夫曼码^[4],FDR 码^[5],MTC 码^[6],九值码^[7]和字典码^[8]等. 这些编码方法都可以取得较好的压缩效果,其中以 FDR 码及其变种码的效果更为突出. 根据 FDR 码的编码思想,提高压缩效率的关键是在测试集中有尽可能多的连续的 0,因此通常采用差分的方法对原始测试集进行处理,这样在解码时除了有限状态机外,相应的解压电路还需要有循环扫描寄存器 (Cyclical Scan Registers, CSR) 来解压差分向量,无疑增加了硬件成本.

通过对测试数据进行分析,可观察到对于许多电路来说,1 游程出现的频率和 0 游程同样重要. 这样对 0 和 1 都进行编码将会有更高的压缩率. 对此本文提出了双游程编码,来对测试数据进行编码.

1 双游程码的编码原理

双游程编码表如表 1 所示,第一列为组号,第二列为游程,第三、第四列为组前缀和后缀,最后两列为完整的 0 游程和 1 游程的代码字,由表 1 看出,随着组数的增加,每个分组中游程的容量依次递增,这样的分组更符合实际数据中游程的分布.

从表 1 中可以看出,双游程编码分别对 0 和 1 进行编码,由组前缀和后缀组成,每个完整编码的组前缀和后缀相同,如果组数为 A_n ,则组前缀为 $n+1$ 位,后缀为 $n+1$ 位,相邻组之间的代码字相差 2 位.

下面举例来说明双游程码的编码过程:

编码前,确定的测试集 $T = 01\ 0000001\ 110\ 00000001\ 111111110\ 0001$,共 35 位,编码后, $T_D = 0100\ 001001\ 1001\ 001011\ 110100\ 0010$,共 30 位. 如果采用 FDR 编码,则 $T_F = 01\ 110000\ 0000\ 110011\ 000000000000000000\ 0110$,共 40 位,编码后的测试向量与原测试向量相比节省了 5 位,与 FDR 编码方法相比,节省了 10 位.

通常一个测试集中除了需要编码的确定位 0 和 1 外,还存在大量的无关位 X ,无关位的值不影响故障覆盖率,因此,在测试数据压缩过程中,可以对无关位指定为有利的数值,提高测试压缩率. 在 FDR 码中,将测试集中的无关位全部指定为 0,在文 [10] 中, X 位是按照所处的 X 串两边的确定位的值来指定的. 只有在 X 位的左右边界都为确定串 1 时,该 X 串才被指定为 1,其余情况都指定为 0,该方法并没有达到最好的填充效果. 在本方案中,采用下面的填充方法:在双游程编码中,有游程 0 和游程 1 两种游程. 在一个 X 游程中所遇到的第一个确定位决定了这个游程中所有 X 应该填充的值. 例如,这一位为 0,则在这个游程中所有 X 被填充为 0,否则,将被填充为 1.

表 1 双游程码编码表

组	游程长度	编码组成		完整编码	
		组前缀	后缀	0 游程编码	1 游程编码
A_1	1		00	0100	1000
	2		01	0101	1001
	3	01(10)	10	0110	1010
	4		11	0111	1011
	5		000	001000	110000
A_2	6		001	001001	110001
	7		010	001010	110010
	8	001(110)	011	001011	110011
	9		100	001100	110100
	10		101	001101	110101
A_3	11		110	001110	110110
	12		111	001111	110111
	13	0001(1110)	0000	00010000	11100000
	⋮			⋮	⋮
	⋮			⋮	⋮

2 压缩效率分析

下面研究双游程编码的压缩效率. 根据表 1 可知,双游程编码根据游程的长度将数据分为 $A_1, A_2, \dots, A_n$ 组. 其中 n 由最长的游程 $L_{\max}$ 确定, n 和 $L_{\max}$ 应满足如下关系式: $n = \log_2(L_{\max} + 4) - 2$,对于 A_i 组,其代码字长度为 $2i + 2$,所能表示的游程范围为 $[2^{i+1} - 3, 2^{i+2} - 4]$

不失一般性,设被测数据中某一位置 0 出现的概率为 p ($0 \leq p \leq 1$),则该位置 1 出现的概率为 $1 -$

p . 根据双游程的编码规则,长度为 l 的 0 游程出现的概率为 $p^l(1-p)$, 长度为 l 的 1 游程出现的概率为 $(1-p)^l p$, 对于任一游程 l , 其属于组 k 的概率 p 为

$$P(i, k) = \sum_{i=2^{k+1}-3}^{i=2^{k+2}-4} [p^i(1-p) + (1-p)^i p] = \frac{p^{2^{k+1}} - p^{2^{k+2}}}{p^3} + \frac{(1-p)^{2^{k+1}} - (1-p)^{2^{k+2}}}{(1-p)^3}$$

双游程码的平均游程长度为

$$\bar{C} = \sum_{k=1}^{\infty} (2k+2) P(i, k)$$

待压缩数据的平均游程长度为

$$\lambda = 1 + \sum_{i=1}^{\infty} ip^i(1-p) + 1 + \sum_{i=1}^{\infty} ip(1-p)^i = \frac{1}{p(1-p)}$$

双游程码的压缩效率为

$$\beta = \frac{\lambda}{\bar{C}} = \frac{1}{p(1-p) \sum_{k=1}^{\infty} (2k+2) P(i, k)}$$

由于 FDR 码和双游程码都存在无限求和的因素,因此在计算时可以取 2 048 项作为近似逼近,通过计算可以知道, p 较大时,双游程码的压缩效果要略好于 FDR 码,而 p 较小时,双游程码的压缩效果要明显好于 FDR 码,这是由于 FDR 码是以 0 游程为基础编码,而双游程码对 0 游程和 1 游程同时进行编码。

3 解码器的设计

双游程编码技术是基于 FDR 编码技术而来的,因此解码器的设计和 FDR 码的解码器设计是类似的,都是基于一个有限状态机的设计,二者之间一个重要的区别是双游程编码的最小游程长度是 1,而且有两个游程 0 和 1. 图 1 所示为该解码器的框图. 该解码器由一个有限状态机(FSM)、一个 $k+1$ 位计数器、一个 $\log_2(k+1)$ 位计数器、一个 SR 锁存器和一个异或门组成. 其中 bit-in 是数据输入端,需解码的数据通过此端移入 FSM, en 是使能端, en 输出 1 表示解码器已准备就绪,等待接受需解码的数据. counter-in 是 FSM 向 $k+1$ 位计数器输入编码数据的通道, v 为标志位, $v=1$ 表示输出数据有效, $v=0$, 输出数据无效, shift 控制编码数据移入 $k+1$ 位计数器, dec1 控制 $k+1$ 位计数器作减 1 操作, rs1 为 $k+1$

位计数器复位标志位. $\log_2(k+1)$ 位计数器作为前缀和后缀的长度计数用,以识别组数. inc 和 dec2 控制该计数器的加 1 和减 1 操作, rs2 为该计数器的复位标志位. SR 锁存器用来存储待解码数据的游程类型, 输出 0 表示 0 游程, 输出 1 表示 1 游程. 异或门用来控制输出数据, data 是 FSM 输出的解码数据, FSM 在解码时,仅根据 0 游程解码,因此,如果游程是 1, 异或门将起取反作用,将 0 转变为 1.

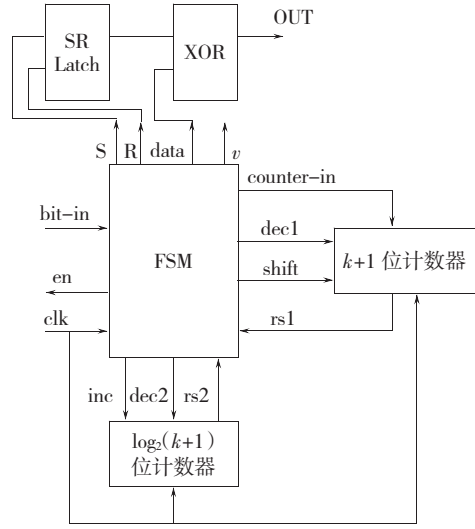


图1 解码器结构框图

该解码器的主要工作原理如下:

1) 首先, FSM 发出使能信号 en 为 1, 先读取前缀信息, 前缀信息由 bit-in 和 counter-in 移入 $k+1$ 位计数器, inc 和 shift 为高电平, $\log_2(k+1)$ 位计数器开始加 1 计数, 若 bit-in = 0, 此过程重复到 bit-in = 1 停止, 反之, 若 bit-in = 1, 则该过程持续到 bit-in = 0 停止. 前缀输入结束, FSM 向 $k+1$ 位计数器移入一个 1, 此过程中, v 为低电平, 表示输出无效.

2) FSM 输出 0 至 data 端, dec1 为 1, $k+1$ 位计数器作减 1 操作, 直到 $k+1$ 位计数器为 0, rs1 为高电平, 此过程中 v 为高电平, 表示输出有效. 前缀解码结束.

3) 由 bit-in 和 counter-in 输入后缀信息到 $k+1$ 位计数器, $\log_2(k+1)$ 位计数器控制后缀的长度, dec2 为 1, 每进入一位数据 $\log_2(k+1)$ 位计数器作减 1 操作, 直到 $\log_2(k+1)$ 位计数器为 0, rs2 为 1, 后缀输入结束.

4) $k+1$ 位计数器作减法操作, 控制 data 输出 0, 直到计数器为 0, 此时 rs1 为高电平, 最后, data 输出一个 1, 此过程中 v 为高电平, 表示输出有效, 后缀解码结束.

4 实验结果

为验证本文算法的有效性,将其应用于 ISCAS89 标准电路,先压缩预先计算的测试集,将压缩后的测试集送入待测电路,通过解码器对压缩数据解码.用 C 语言编写一个随机数据发生器作为芯片测试集,其中,压缩率的计算方法为

$$\text{压缩率} = \frac{T_D - T_E}{T_D} \times 100\%$$

其中: T_D 为原测试集, T_E 为压缩后的测试集. 比较结果如表 2 所示.

表 2 几种压缩方案效果比较

电路	Golomb 码 压缩率 ^[1]	FDR 码 压缩率 ^[2]	9C 码压 缩率 ^[3]	VIHC 码压 缩率 ^[4]	双游程码 压缩率
S5378	37.11	47.98	51.64	51.78	52.08
S9234	45.25	43.61	50.91	47.25	48.13
S13207	79.74	81.3	82.31	83.51	82.85
S15850	62.82	66.21	66.38	67.94	69.57
S38417	28.37	43.37	60.63	53.36	61.86
S38584	57.17	60.93	65.53	62.28	63.69
AVG	51.74	57.23	62.90	61.02	63.03

从表 2 可以看出,双游程码的压缩效率要好于另外四种编码方案.本方案与 Golomb 码的平均差值压缩率为 11.29%,与 FDR 码的平均差值压缩率为 5.8%,与 9C 码的平均差值压缩率为 0.13%,与 VIHC 码的平均差值压缩率为 2.01%,这主要是因为测试集中除了有大量的 0 游程外,还有大量的 1 游程,同时对 0 游程和 1 游程进行编码可以获得更高的压缩率.因此,上述 4 种压缩方案压缩率要低于双游程码方案.综合考虑到压缩率、解码复杂度等因素,双游程编码是数据压缩算法中一种较理想的方案.

5 结 语

为了降低 SOC 芯片测试中的成本问题,本文提出了一种双游程编码的测试数据压缩方法.它以 FDR 码为基础,根据测试数据中 0 和 1 同时大量存在这个事实,针对 FDR 码仅对 0 游程进行编码的缺点,提出了对 0 游程和 1 游程同时进行编码,有效地

减少了测试数据集中短游程的数量,提高了压缩率.该方案的解码器独立于被测电路,无需对原测试集作差分变换,节省了大量的硬件开销.与同类编码方案相比,具有很大的优越性.

参 考 文 献:

- [1] KCHAKRABARTY A C. System-on-a-Chip Test Data Compression and Decompression Architectures based on Golomb codes [J]. IEEE Transactions on CAD of Integrated Circuits and System, 2001, 20(3): 355 - 368.
- [2] JAS A, GHOSH D J, TOUBA N. Scan Vector Compression/Decompression using Statistical Coding [J] // Proc. VLSI Test Symp, San Diego, CA, April 1999: 114 - 120.
- [3] JAS A, ToubA N. Test vector decompression via cyclical scan chains and its application to testing core-based designs [C] // Proc. Int. Test Conf., Washington, DC, October, 1998: 458 - 464.
- [4] NOURANI M, TEHRANIPOUR M. RL-Huffman Encoding for Test Compression and Power Reduction in Scan Application [J]. ACM Trans. Des. Autom. Electron. Syst., 2005, 10(1): 91 - 115.
- [5] CHANDRA A, CHAKRABARTY K. Test Data Compression and Test Resource Partitioning for System-on-a-chip using Frequency-directed run-length (FDR) Codes [J]. IEEE Trans. Comput., 2003, 52(8): 1076 - 1088.
- [6] ROSINGER P, GONCIARI P, Al-Hashimi B, et al. Simultaneous Reduction in Volume of Test Data and Power Dissipation for System-on-a chip [J]. Electron. Lett., 2001, 37(24): 1434 - 1436.
- [7] TEHRANIPOUR M, NOURANI M, CHAKRABARTY K. Nine-coded Compression Technique for Testing Embedded cores in SoCs [J]. IEEE Trans. Very Large Scale Integr. Syst, 2005, 13(6): 719 - 731.
- [8] LI L, CHAKRABARTY K. Test Data Compression using dictionaries with Fixed Length Indices [J] // Proc. VLSI Test Symp. Napa Valley, CA, April. 2003: 219 - 224.
- [9] 彭喜元, 俞洋. 基于变游程编码的测试数据压缩算法 [J]. 电子学报, 2007, 35(2): 197 - 201.
- [10] EL-MALEH A, AL-ABAJI R. Extended Frequency-directed Run Length Code with Improved Application to System-on-a-chip Test Data Compression [J] // Proc. 9th IEEE Int. Conf. Electronics, Circuits and Systems, Dubrovnik, Croatia, September, 2002: 449 - 452.
- [11] GONCIARI P T, AL-HASHIMI B, NICOLICI N. Synchronization Overhead in SoC Compressed Test [J]. IEEE Trans. Very Large Scale Integr. Syst., 2005, 13(1): 140 - 153.
- [12] GONCIARI P T, AL-HASHIMI B M. Variable-length Input Huffman Coding for System-on-a-chip Test [J]. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, 2003, 22(6): 783 - 796.

(编辑: 温泽宇)