

一种相对游程长度编码方案

韩建华 詹文法 查怀志

(安庆师范学院教育学院多媒体与网络研究室 安庆 246133)

摘要 提出一种相对游程长度编码方案,以在不增加待编码数据中游程数量的情况下,达到减少待编码游程长度的目的,即通过缩短代码字长度来提高压缩效果。对 ISCAS89 部分基准电路的实验结果显示,提出的方案在压缩效率和解压结构方面都明显优于 Golomb 码、FDR 码、EFDR 等同类方案。

关键词 测试数据压缩,编码,内建自测试,外建自测试,代码字

中图分类号 TP306.2 **文献标识码** A

Coding Scheme of Relative Length of Runs

HAN Jian-hua ZHAN Wen-fa ZHA Huai-zhi

(Multimedia and Network Research Lab, School of Education, Anqing Normal College, Anqing 246133, China)

Abstract A coding scheme of relative length of runs was presented. It reduces the length of runs needing to be encoded by not increasing the number of runs in encoded test data. So the compression effect is enhanced by shortening the length of codeword. Experimental results on some ISCAS 89 benchmark circuits show that the proposed scheme obviously outperforms the traditional coding methods in the compression ratio and the implementation of decompression, such as Golomb, FDR, EFDR coding.

Keywords Test data compression, Encoding, Built-in self-test, Built-off self-test, Codeword

1 引言

随着集成电路技术的发展,可以在一个芯片中集成数以万计的晶体管,并且可以集成预先设计和经过验证的 IP,如存储器、微处理器、DSP(数字信号处理器)等。这种复杂的集成芯片已经成为能处理各种信息的集成系统,被称为片上系统或系统芯片(System-on-a-Chip, SoC)。SoC 大大降低了系统成本,缩短了设计周期,加快了产品面市时间,但是 SoC 产品的测试面临着越来越多的挑战。芯片的测试已经成为制约集成电路发展的一个“瓶颈”。已有大量的文献对集成电路的测试技术展开研究,主要有测试集紧缩技术(Test Set Compaction, TSC)^[1]、内建自测试(Built-In Self-Test, BIST)^[2]和外建自测试 3 种方法^[3-15]。

测试集紧缩技术^[1]通过自动测试模式生成算法(Auto Test Pattern Generation, ATPG)对测试立方体进行静态或动态压缩,通过减少测试立方体数量达到减少测试数据量的目的。该技术的优点是不需额外的硬件开销;其缺点是可能无法检测到某些非故障模型的故障。

内建自测试^[2]方法,依靠芯片自身的资源完成对芯片的测试。此方法将测试模式生成器 TPG、测试过程控制和测试响应评价功能模块嵌入在被测电路 CUT 上,摆脱了对 ATE 的依赖,减少了测试费用。但由于 BIST 生成的多是伪随机

测试向量,测试时通常存在着抗随机故障(Random Resistant Fault, RRF),因此 BIST 存在故障覆盖率不高、测试序列较长的弊端。虽然可以通过加权或采用混合模式的 BIST 等方法来进一步提高测试效率,但随着电路规模的扩大和 RRF 的增多,要付出的硬件开销将显著增加。

外建自测试^[3-15]方法又称为测试源划分技术,此方法将所需的测试向量经过压缩存储在 ATE 中。测试期间,通过片上的解压电路将其还原施加到被测电路上。它同样是将一些测试资源从 ATE 移入到芯片中,以达到减少测试数据量、缩短测试时间的目的,并允许使用低速 ATE 而不降低测试质量。该方法不需要了解被测设计(Design Under Test, DUT)的具体内部结构,可以很好地保护知识产权,因而得到广泛应用。

基于编码的方法是外建自测试中应用最广泛的技术之一。该方法将芯核供应商提供的测试集 T_D 压缩(编码)成一个更小的测试集 T_E ,并把它存储于 ATE 的存储器中。测试时,由芯片上的解码电路把 T_E 解码成 T_D 后再进行测试。测试数据编码压缩按照编码的原理可分为统计编码和游程编码两大类;统计编码中具有代表性的方法为统计码^[1]、Huffman 选择码^[1]以及 9 值编码等^[3,4],而游程编码方法包括 Golomb 码^[5]、FDR 码^[6]、EFDR^[7]等。文献[12]提出一种虚拟块游程编码,其通过减少游程数量的方法来减少待编码的测试数据,

到稿日期:2011-06-22 返修日期:2011-10-18 本文受安徽高校省级自然科学基金项目(KJ2011A198),安徽省自然科学基金(10040606Q42)资助。

韩建华(1962—),男,副教授,主要研究方向为通信、测试等,E-mail: hjh111101@sina.com;詹文法(1978—),男,博士,副教授,主要研究方向为测试数据压缩等;查怀志(1985—),男,硕士,讲师,主要研究方向为计算机应用。

具有一定的借鉴意义和实用价值。

编码压缩技术所用测试集 T_D 其测试模式数量较少,测试时间相对 BIST 更短;并且它不需要测试模式生成电路和响应压缩分析电路,其解压电路的成本低于 BIST;对于不支持 BIST 的 CUT 也可实现测试,具有广泛的应用性。

游程编码由于编码简单的特点在编码技术中应用最为广泛,但当前的游程编码编码时都基于游程本身的长度。传统的方法主要是通过减少代码字的平均长度来减少编码后的数据。显然如果能够减少待编码数据中游程的数量或长度,也能够对原始测试数据达到压缩的目的。文献[12]提出了减少待编码数据中游程的数量的方法,它有一定的实际意义。显然,通过减少待编码数据中游程的长度也能够压缩测试数据。基于这一思想,提出一种相对游程长度编码方案,其仅需编码相对游程长度,即编码当前游程与前一游程长度的差值,从而缩短了待编码的游程长度,减少了对应代码字长度,最终提高了压缩率。

2 相对游程长度编码算法

2.1 测试集的预处理过程

对于生成的测试集,先将其分为一个个游程分段,对相对应的相对游程长度编码,就是将测试集分为若干个 0 游程或者 1 游程的组合。测试集中包含“0”位、“1”位和无关位,在此无关位用“X”表示。形如“0000001”或者“00XX1”的游程,为 0 游程;同样,形如“1111110”或者“X11X10”的游程,为 1 游程。具体处理步骤如下:

- (a) 读取测试集中的第一位;
- (b) 若该位为 X,则继续读取下一位;
- (c) 当取到不为 X 的值时,记录该位的值为 i ,继续读取下一位;
- (d) 若该位与 i 相等或者为 X,则继续读取下一位;
- (e) 直到取到的值与 i 不相等时终止,并得到该位与该位之前的所有位为一个游程分段,该游程取决于 i 的值,若 $i=0$ 则为 0 游程, $i=1$ 则为 1 游程;
- (f) 读取下一位,重复(b);
- (g) 读到测试集最后一位时,结束。

通过上述步骤,若干游程分段重新拼成测试集 $\{Y_1, Y_2, Y_3, Y_4, \dots, Y_n\}$,在此期间对于无关位 X,实际上在编码时,是将其当作与 i 相等的值进行编码的,相应解码时获得的译码中的 X 都将被译为与其对应的 i 的值,因此在解码后的结果中是不存在无关位 X 的。

需要指出的是,可以对某些 X 位灵活赋值,使游程之间的长度差值落差更加缓和。比如,在一个长度为 m 位的游程后,如果出现一个长度为大约 $2m$ 位的无关位 X 时,就可以对其中的 X“做些手脚”。举例来讲,对于下面这部分测试集:

XX00XX00001XXXXXXXXXXXXXXXXXXXXXXXXX1

可以将其分为 2 个游程分段:

XX00XX00001 和 XXXXXXXXXXXXXXXXXXXXXXX1

但是,有没有更好的方法呢?答案是有的。我们可以对后一个游程中的中间的 1 个 X 赋值为 1,使其一分为二,这样就将原始测试集分为 3 个游程分段:

XX00XX00001,XXXXXXXXXXXX1 和 XXXXXXXXXXXX1

这样,后 2 个游程分段都与前一个游程分段相等,而按照

本文提出的编码方法,相等的情形的代码字是非常短的,这样可以进一步提高压缩效果。

2.2 编码

相对游程长度编码是一种不等间距的变长到变长的编码方式^[3-15]。所谓不等间距编码方式是指在分组时每个组的大小根据其出现的频率进行了适当的调整。表 1 是相对游程长度编码的编码表。第 1 列为组号,第 2 列为相邻游程相差位数,第 3 列为组前缀,第 4 列为尾部,最后 4 列为对应编码。U00 表示比前一游程短的 0 游程的码字;U01 表示比前一游程短的 1 游程码字;U10 表示比前一游程长的 0 游程码字;U11 表示比前一游程长的 1 游程码字。

表 1 相对游程长度编码表

分组	相差位数	组前缀	尾部	U00	U01	U10	U11
A ₁	0		0	0000	0100	1000	1100
	1	0	1	0001	0101	1001	1101
A ₂	2		00	001000	011000	101000	111000
	3	10	01	001001	011001	101001	111001
	4		10	001010	011010	101010	111010
	5		11	001011	011011	101011	111011
6	000		00110000	01110000	10110000	11110000	
A ₃	7		001	00110001	01110001	10110001	11110001
	8		010	00110010	01110010	10110010	11110010
	9	110	011	00110011	01110011	10110011	11110011
	10		100	00110100	01110100	10110100	11110100
	11		101	00110101	01110101	10110101	11110101
	12		110	00110110	01110110	10110110	11110110
	13		111	00110111	01110111	10110111	11110111
.....

从表 1 中,可以概括出这种相对游程长度编码具有如下特点:

- (a) 编码由游程位、增减位、组前缀和尾部组成;
- (b) 在每一个编码中组前缀和尾部都具有相同的位数,游程位和增减位各占一位;
- (c) 如果组号为 A_j ,那么组前缀的位数等于 j ,并且该组中共有 2^j 个成员。例如 A_2 中,组前缀为 2 位,共有 4 个成员; A_3 中,组前缀为 3 位,共有 8 个成员;
- (d) 组前缀的二进制值与该组中第一个成员的数值相等。例如 A_3 组中,前缀为 110,其对应十进制为 6,与该组中第一个成员 6 相等;
- (e) 相邻两组代码字长度相差 2 位;
- (f) 如果 A_k 代表最后一组, L_{\max} 为最长的游程数,那么二者满足如下关系: $(2^k - 3) < L_{\max} \leq (2^{k+1} - 3)$ 。例如, L_{\max} 为 15,那么 k 必须为 4 才能满足以上关系,这意味着最大长度 15 位于组 A_4 中,而 A_4 的中满足的游程长度为 $[14, 29]$ 。

已经得到测试集的若干个连续的游程分段 $\{Y_1, Y_2, Y_3, Y_4, \dots, Y_n\}$,接下来就开始编码。在对第一个游程分段 Y_1 编码前,先设定初始游程长度 $len^0 = 0$,那么第一个游程分段 Y_1 必然为增加位,并且 Y_1 的游程长度即为相差位数。获取相差位数之后,则按照表 1 所列的编码方法对其进行编码,并将游程分段 Y_1 的位数保存作为新的 len^0 。

对于其余的所有分段,按照下面方法编码:设待编码分段为 Y_m ,那么取 Y_m 的长度为 len^1 。在此之前的应该已有 len^0 的值,即为 Y_{m-1} 的长度。相差位数为 $len = len^1 - len^0$ 的绝对值 $|len|$,若 $len > 0$,则游程分段的码字为增加位的码字; $len <$

是位输入和使能信号,当有限状态机准备就绪时,输入编码的数据。信号 counter_in 是有限状态机到 $k+1$ 位计数器的位输出,shift 控制编码的后缀移入 $k+1$ 位计数器 1,dec1 控制 $k+1$ 位计数器 1 减一操作,load 指示装载 $k+1$ 位寄存器的内容到 $k+1$ 位计数器 1 中,rs1 表示 $k+1$ 位计数器 1 置位操作(即将该计数器的最低位置 1)。信号 out 是有限状态机的输出,受输出有效信号 v 控制,输出解压后的数据。

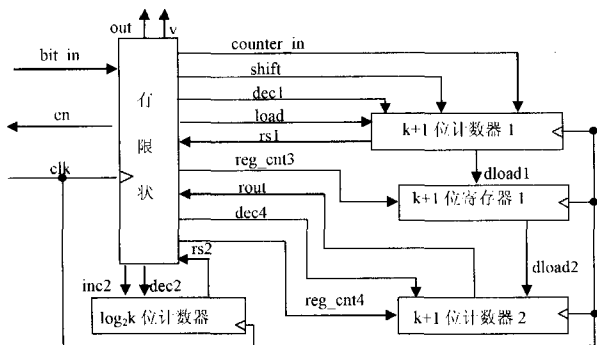


图3 相对游程长度编码解压结构框图

这个解压器的主要工作原理如下:

(a)首先,有限状态机发出使能信号 en 为 1,然后 shift 和 inc 均为高电平,同时读取 bit_in 的值,若不跳变(即由 0→1 或由 1→0), $\log_2 k$ 位计数器开始加一计数,此过程一直重复到 bit_in 的值发生跳变(即由 0→1 或由 1→0)。

(b)接下来,输入后缀进入 $k+1$ 位计数器 1。 $\log_2 k$ 位计数器控制后缀的长度,dec2 控制该计数器做减一计数,当计数器为 0 时,返回 rs2 高电平,后缀输入结束。

(c) $k+1$ 位寄存器 1 通过 dload2 将数据复制给 $k+1$ 位寄存器 2。

(d) $k+1$ 位计数器 2 做减一操作,同时 $k+1$ 位计数器 1 根据前缀的跳变情况做加一操作或减一操作,此过程一直重复到 $k+1$ 位计数器 2 的值为 4(即二进制“100”)。

(e)reg_cnt1 置为高电平,通过 dcnt 复制 $k+1$ 位计数器 1 的值到 $k+1$ 位寄存器 1 中。

(f) $k+1$ 位计数器 1 做减一计数,同时在有限状态机的控制下, v 输出高电平,out 输出低电平,此过程一直重复到 $k+1$ 位计数器 1 的内容为 4(即二进制“100”),再次让 out 和 v 都输出高电平。至此,该游程解压结束。

(g)重复步骤(a)到步骤(f),直到所有数据全部解压。

5 实验结果及分析

该部分将通过实验数据来验证本方案的效果。为了增加实验结果的可比性,本文采用美国 Duke 大学提供的 MinTest 产生的测试向量集,分别对 ISCAS89 基准电路中几个规模较大的时序电路进行了实验。

表2 本方案的压缩效果

电路名称	原始数据位数(bit)	压缩后数据位数(bit)	压缩率(%)
S5378	23866	15229	36.19
S9234	39432	22499	42.94
S13207	165436	31815	80.77
S15850	77112	27383	64.49
S35932	28224	5807	79.43
S38417	164835	76557	53.56
S38584	199240	79131	60.28
平均			59.67

下面将给出本方法的实验结果。这里使用的实验电路是 ISCAS 89 基准电路中较大的几个时序电路,实验结果如表 2 所列。第 1 列为电路名称,第 2 列为原始测试数据位数,第 3 列为压缩后的数据位数,第 4 列为压缩率。

从表 2 中可以看出,本方案对 ISCAS89 基准电路中规模较小的几个电路 S5378 和 S9234 实验结果较差,对规模较大的几个电路 S13207, S15850, S35932, S38417 和 S38584 实验结果较好。这主要是因为电路 S13207, S15850, S35932 的 Mintest 测试集中,尤其是测试集后面的部分,含有大量很长的游程分段。由于其数量很多,长游程分段连续出现在所难免,因此在这里相对游程长度编码就起到了非常好的压缩效果。而对于 S5378 和 S9234,其中的大部分游程段都比较短而且没有规律,尤其开始数行确定位非常之多,所以用该方法压缩效果并不是很好。由于在实际的电路中,电路规模都比较大,且无关位非常多,占测试集的 90% 以上^[1],因此本方案不失为一种有效方案。

为了验证本方案的效果,将本方案与国内同期成果进行比较,实验结果如表 3 所列。

表3 相对游程长度编码与其它算法比较

电路名称	本方案 (%)	Golomb (%)	FDR (%)	EFDR (%)	交替连续码 (%)
S5378	36.19	40.70	48.19	50.81	45.12
S9234	42.94	43.34	44.88	45.89	42.79
S13207	80.77	74.48	78.67	79.38	80.43
S15850	64.49	41.77	52.87	56.29	65.13
S35932	79.43	4.69	10.19	45.63	79.06
S38417	53.56	44.12	54.53	52.35	56.52
S38584	60.28	54.86	52.85	62.91	60.57
平均	59.67	43.42	48.88	56.18	58.54

第 1 列为电路名称,第 2 列为本方案压缩效果,第 3 列为 Golomb 码压缩效果,第 4 列为 FDR 码压缩效果,第 5 列为 EFDR 码压缩效果,第 6 列为交替连续码压缩效果^[8]。可以发现,对于 7 种不同的时序电路,本方案具有很强的稳定性和较高的压缩效率。本方案与 Golomb 码方案的平均差值压缩效率(两种算法针对 7 种不同电路的压缩差值的平均值)为 16.25%;与 FDR 码的平均差值压缩效率为 10.79%;与 EFDR 码的平均差值压缩效率为 3.49%;与交替连续码^[8]的平均差值压缩效率为 1.125%。

为了进一步验证本方案的实际效果,将 ISCAS89 基准电路中规模较大的几个电路的跳变故障集与国内其它同类方案进行了比较,所有方案均采用了利于自己的最优无关位的赋值方式,结果如表 4 所列。采用跳变故障集的原因主要是因为跳变故障集原始测试数据量大、所包含无关位多,这两个特点类似于实际的工业设计。

表4 相对游程长度编码与国内同类算法比较

电路名称	原始数据位数(bit)	本方案	混合定变长变长 ^[11]	混合定变长虚拟块 ^[12]	共游程码 ^[13]
S5378	635586	86.70	79.35	89.76	83.52
S9234	1206867	82.88	83.77	86.35	78.48
S13207	5044248	91.01	78.64	84.42	88.67
S15850	4975344	94.85	83.42	85.27	89.53
S38417	28203842	97.12	85.92	89.68	86.98
S38584	44947224	97.05	87.63	88.31	93.51
平均		91.60	83.12	87.30	86.78

从表4可以看出,对于不同的时序电路,本方案相对于其它几种方案都有不同程度的改善。本方案与混合定变长码的平均差值压缩效率为8.48%;与混合定变长虚拟块方案的平均差值压缩效率为4.30%;与共游程码方案的平均差值压缩效率为4.82%。这些数据充分说明了本方案的有效性。

结束语 为了进一步降低测试成本,本文提出了一种相对游程长度编码的测试数据压缩方案。本方案独立于被测试电路。因此,其特别适用于没有结构信息的IP核的测试,与同类编码方案比较,本方案特别适用于测试集中游程长度相差较小的情况。

参 考 文 献

[1] Touba N A. Survey of test vector compression technique[J]. IEEE Design & Test of Computers, 2006, 23(4): 294-303

[2] Zhan Wen-fa, Ma Jun, Du Shan-shan, et al. A LFSR Reseeding Scheme Based on Division by 2 to the Power of Integer[J]. International Journal of Digital Content Technology and its Applications, 2010, 4(9): 88-96

[3] Zhan Wen-fa, Liang Hua-guo, Jiang Cui-yun, et al. A Scheme of Test Data Compression Based on Coding of Even Bits Marking and Selective Output Inversion[J]. Computers and Electrical Engineering, 2010, 36(5): 969-977

[4] Zhan Wen-fa, Liang Hua-guo, Shi Feng, et al. Test data compression scheme based on variable-to-fixed-plus-variable-length coding[J]. Journal of Systems Architecture, 2007, 53(11): 877-887

[5] Chandra A, Chakrabarty K. Test data compression and decompression based on internal scan chains and Golomb coding[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2002, 21(6): 715-722

[6] 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 Transactions on Computers, 2003, 52(8): 1076-1088

[7] El-Maleh A H. Test data compression for system-on-a-chip using extended frequency-directed run-length code[J]. IET Computers & Digital Techniques, 2008, 2(3): 155-163

[8] 梁华国, 蒋翠云. 基于交替与连续长度码的有效测试数据压缩和解压[J]. 计算机学报, 2004, 27(4): 548-554

[9] 韩银和, 李晓维, 徐勇军, 等. 应用 Variable-Tail 编码压缩的测试资源划分方法[J]. 电子学报, 2004, 32(8): 1346-1350

[10] 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

[11] 詹文法, 梁华国, 时峰, 等. 混合定变长码的测试数据压缩方案[J]. 计算机学报, 2008, 31(10): 1826-1834

[12] 詹文法, 梁华国, 时峰, 等. 一种混合定变长虚拟块游程编码的测试数据压缩方案[J]. 电子学报, 2009, 37(8): 1837-1841

[13] 詹文法, 梁华国, 时峰, 等. 一种共游程码的测试数据压缩方案[J]. 计算机研究与发展, 2008, 45(10): 1646-1653

[14] 许川佩, 董祥健. 一种交替游程编码的SOC测试数据压缩方法[J]. 计算机工程与应用, 2010, 46(25): 57-60

[15] 商进, 张礼勇. 一种混合相容数据块的测试数据压缩方案[J]. 哈尔滨理工大学学报, 2011, 16(6): 93-96

(上接第290页)

为简便起见,我们将同类细胞的控制和管理功能集中在细胞管理器中,使得细胞管理器成了可靠性单点。下一步工作中,将研究实现一种分布式的结构,亦即将细胞管理器的集中管理功能分散到各个细胞单元中去,通过细胞单元的内部信息以及细胞单元间的信息交换实现数据的正确分派及系统的自我修复。

参 考 文 献

[1] 李俊丰. 图像匹配应用的硬件加速技术研究[D]. 长沙:国防科技大学计算机学院, 2006

[2] Neumann J V. Probabilistic logics and the synthesis of reliable organisms from unreliable components[M]//Shannon C E, McCarthy J. Automata Studies. Princeton: Princeton Univ Press, 1954: 43-98

[3] Stauffer A, Mange D, Sanchez E, et al. Embryonics: towards new design methodologies for circuits with biological-like properties [C]//Proceedings of 1995 International Workshop on Logic and Architecture Synthesis. Grenoble, France, 1995: 299-306

[4] Boesen M R, Madsen J. eDNA: a bio-inspired reconfigurable hardware cell architecture supporting self-organisation and self-healing[C]//Proceedings of 2009 NASA/ESA Conference on Adaptive Hardware and Systems(AHS2009). San Francisco,

USA, 2009: 147-154

[5] Boesen M R, Schleuniger P, Madsen J. Feasibility study of a self-healing hardware platform[C]//Proceedings of the 6th International Symposium on Applied Reconfigurable Computing (ARC2010). Bangkok, Thailand, 2010: 29-41

[6] Boesen M R, Madsen J, Pop P. Application-Aware Optimization of Redundant Resources for the Reconfigurable Self-Healing eDNA Hardware Architecture[C]//Proceedings of 2011 NASA/ESA Conference on Adaptive Hardware and Systems(AHS2011). San Diego, USA, 2011: 66-73

[7] Boesen M R, Madsen J, Keymeulen D. Autonomous distributed self-organizing and self-healing hardware architecture-the eDNA concept[C]//Proceedings of the 2011 IEEE Aerospace Conference. Big Sky, Montana, USA, 2011: 1-13

[8] Xu Jia-qing, Dou Yong, Lv Qi, et al. eTissue: a bio-inspired match-based reconfigurable hardware architecture supporting hierarchical self-healing and self-evolution [C]//Proceedings of 2011 NASA/ESA Conference on Adaptive Hardware and Systems(AHS2011). San Diego, USA, 2011: 311-318

[9] Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell(The 5th edition)[M]. Oxford: Garland Science, 2008

[10] Szabo E, Rampalli S, et al. Direct conversion of human fibroblasts to multilineage blood progenitors[J]. Nature, 2010, 468(7323): 521-526