论有限回溯测试模式产生方法

曾芷德,曹贺锋
(国防科技大学计算机学院,长沙 410073)

摘 要: 本文首先剖析了有限回溯测试模式产生(FBTPG)方法的实质,然后在深入分析三种 ATPG 系统的 C-B 曲线的实验数据的基础上,提出故障模拟对测试生成的综合调节效应,为 FBTPG 方法的有效性提供了理论依据. 最后以 ISCAS-85 和 ISCAS-89 电路为基础,给出了 FBTPG 与随机测试生成、确定性测试生成和商用 ATPG 系统 FlexTest 的实验比较结果,从而论证了 FBTPG 方法处理超大规模时序电路的有效性.

关键词: 有限回溯测试模式产生; 确定性测试生成; 故障模拟
中图分类号: TP306 **文献标识码**: A **文章编号**: 0372-2112 (2000) 11-0102-04

On Finite Backtracking Test Pattern Generation Method

ZENG Zhi-de, CAO He-feng
(Computer Institute, National University of Defense Technology, Changsha 410073, China)

Abstract: Based on the penetrating analysis of the characteristics in finite backtracking test pattern generation (FBTPG) and the three experimental C-B curves from three different ATPG systems, a new global adjusting effect about the influence of fault simulation on test generation is proposed. This theory is the theoretic base to prove the efficiency of FBTPG. Finally, ISCAS-85 circuits are used as a benchmark to run FBTPG, random test generator, deterministic test generator, and a commercial ATPG system (FlexTest) respectively and compare the results. The results prove that FBTPG method is more efficient in large scale sequential circuits.

Key words: finite backtracking test pattern generation; deterministic test generation; fault simulation

1 引言

确定性测试生成(ATPG)算法已有四十年的发展历史,由于其计算复杂性是随电路的规模和复杂性的增加呈指数膨胀的,目前已很难满足 VLSI 的应用需求. 国外文献[1]认为,对于大时序电路,在高档工作站上,ATPG 系统的运行时间一般要以星期计. 根据经验,这个"大"的含义应是万门量级以上. 曾用一个从国外引进的、号称目前世界上最先进的 ATPG 软件系统处理用户提供的十几万门的电路,在高档工作站上运行了一个星期,故障覆盖率只有 13 %,由此可见问题的严重性.

扫描设计能使此困难趋于缓和,但它是以多占芯片面积、延迟增大、测试应用开销增加为代价的,用户未必都欢迎. 再者,现有的 DFT(可测性设计)规则过严,难以用于高速电路设计. 例如上述十几万门的电路,6000 多个触发器经扫描设计规则检查确认只有 10 %可扫描. 这就是说,为了插入扫描链,必须对 90 %的触发器设计进行修改,用户很难接受. 由此看来,非扫描测试问题是回避不了的. 近十年来,国内外学者已广泛展开了对非扫描时序电路测试生成方法的研究. 有从并行算法入手的[1,2];也有利用网表寄存器传输级描述的[3]. 这些研究方向都值得坚持,但能否取得突破并使之适应大规模时序电路的复杂性,目前还难下结论.

能否对传统的 ATPG 算法进行改造,使之适应大规模时序电路的需求呢? 我们在文献[4]中对此进行了探索,提出一种把确定性算法与随机测试生成方法动态结合起来的有限回溯测试模式产生方法 FBTPG,并在三万门左右的 ASIC 测试生成中证明了它的有效性. 本文将在文献 4 和 5 的基础上,对 FBTPG 方法的一般规律进行较深入的理论探讨,并以 ISCAS-85 和 ISCAS-89 Benchmark 为基础,对 FBTPG 方法与确定性测试生成算法和随机测试生成方法进行实验比较研究,以证明 FBTPG 方法的普遍有效性.

2 关于 FBTPG 方法

FBTPG 方法的原型可以是任何一种确定性 ATPG 算法,该算法所采用的所有加速测试生成的策略,它都可以用,也应该用. 它与确定性 ATPG 算法的不同仅在于用上了被后者废弃的失败(aborted)故障码. 这一点使 FBTPG 方法发生了不同于确定性算法的质的变化.

首先,FBTPG 方法追求的目标不再是推导单个目标故障的"真"测试码. 而是求整个目标电路的高故障覆盖率的测试集. FBTPG 认为,单个目标故障的"假"测试码如果能检测其它故障,则对整个电路来说仍是"真"测试码. 因此,传统的"真"

收稿日期:1999-09-17;修回日期:2000-03-16
基金项目:国家自然科学基金(No. 69773035)

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

“假”测试码概念必须更新. 曾做过一个实验,对 ISACS-89 的 S35932,限制回溯次数 60 次,分别用四值测度[6]和二值测度引导 FBTPG 方法生成大约 1 千拍(时钟周期)测试序列. 四值测度产生的 1006 拍输入序列中,只有两个是传统意义上的“真”测试码,而二值测度产生的 1017 拍输入序列中,有 63 个“真”测试码. 如果按确定性算法的概念,毫无疑问是二值结果优于四值. 但用 FlexTest 故障模拟器验证的结果是:四值的 1006 拍码子的故障覆盖率为 89. 61 %,除去不可测故障,只剩下 11 个故障未检测. 而二值的 1017 拍码子只有 70. 71 %的故障覆盖率,未测故障有 7280 个之多. 可见四值结果要好得多. 此实验表明,再拘泥于传统意义上的“真≠假”测试码概念是没有意义的.

其次,FBTPG 方法通过较小的回溯次数的控制,只对求测试涉及的搜索空间进行有限的搜索,剩下未搜索到而又必须搜索的空间由随机码解决. 这样做既避免了计算复杂性的指数膨胀;又可把 ATPG 算法的精确性与随机码的高速度动态地结合起来. 本文后面的实验证实,随机码对时序电路的故障覆盖效果大部分是很差的. 但当按确定性 ATPG 思想为随机码动态地插入某些约束条件,或者说在确定性算法所创建的测试模式框架内随机地给未定值的初级输入赋值时,便能够收到意想不到的良好效果. 从这个意义上说,可以认为 FBTPG 方法是一种动态约束随机模式产生方法,不过其约束条件是按确定性算法思想动态产生的.

由上可见,FBTPG 方法实质上是要在较小的回溯次数控制下,让确定性 ATPG 与随机测试生成动态结合以产生高故障覆盖率的测试集. 为适应超大规模时序电路的复杂性作准备. 从本文后面的实验结果可以看出,此目标是可以达到的. 下面用定性分析与对实验结果的定量分析相结合的方法为 FBTPG 的有效性提供理论依据.

3　故障模拟对测试生成的综合调节效应(简称 GAFSTG 效应)

如果用确定性 ATPG 逐个故障地生成测试,不加故障模拟,那么显而易见的是:故障覆盖率与回溯次数之间的函数曲线(简称 C-B 曲线或 C-B 函数)是单调不降的. 但当把故障模拟集成进来之后, C-B 曲线的单调不降特性将被破坏. 例如:假设有故障集 $F=\{f_i \mid i=1,2,3,4\}$,用 B_i 表示生成 f_i 的真测试码 T_i 时所需的回溯次数,且有 $B_1 > B_2$; B_3、$B_4 \gg B_1$,即很难直接生成 T_3 和 T_4. 又设 T_1 可另检测 f_2; T_2 可另检测 f_3 和 f_4. 如果限制 ATPG 的回溯次数 B 满足 $B_2 < B < B_1$,则可生成 T_2,检测 f_2、f_3、f_4;如果限 $B > B_1$,则只能够生成 T_1,检测 f_1 和 f_2,覆盖率反而下降. 这种人为假设的结果有无普遍性呢? 表 1 的实验数据作了肯定回答.

表 1 是用 G-F 二值算法[7]得到的未检测故障随回溯次数的变化规律. 结果表明,有 5 个电路的 C-B 函数曲线已不是单调变化的. 表 2 的数据性质与表 1 相同,不过是用 FBTPG 方法得到的. 表 3 是不用故障模拟时的 ATPG 结果. 比较表 1 和表 2 的 C-B 函数可知:

表 1　G-F 二值算法的 C-B 函数

回溯数 / 电路名	0	2	6	8	20	40	60	80	100
C432	12	5	5	5	4	4	4	4	4
C499	88	107	103	171	151	116	93	100	100
C880	26	3	0	0	0	0	0	0	0
C1355	425	424	264	264	162	98	98	98	98
C1908	667	381	133	158	122	117	106	106	106
C2670	696	520	493	490	145	139	139	139	139
C3540	264	215	186	188	158	160	159	160	160
C5315	257	161	116	95	79	81	77	79	76
C6288	77	80	50	54	44	40	42	42	42
C7552	1000	1020	811	747	330	207	194	194	194
平均	351	292	216	217	120	96	91	92	92

表 2　FBTPG 方法的 C-B 函数

回溯数 / 电路名	0	2	6	8	20	40	60	80	100
C432	15	6	5	6	4	4	4	4	4
C499	42	44	50	47	36	32	34	48	35
C880	7	1	0	0	0	0	0	0	0
C1355	129	115	29	31	19	14	18	14	10
C1908	83	78	36	36	36	48	32	30	29
C2670	487	466	453	448	137	130	132	130	133
C3540	156	164	148	154	143	149	146	145	145
C5315	130	110	95	91	80	83	79	76	76
C6288	34	38	34	34	38	38	38	38	38
C7552	563	528	469	443	245	193	198	201	197
平均	165	155	132	129	74	69	68	69	67

表 3　无故障模拟的 G-F 二值算法的 C-B 函数

回溯数 / 电路名	0	4	8	20	60	100
C432	65	30	8	5	5	5
C499	605	571	569	559	409	354
C880	287	7	0	0	0	0
C1355	1262	1168	824	623	545	545
C1908	1834	610	359	274	218	213
C2670	2077	1240	1204	721	709	709
C3540	2005	907	707	420	785	360
C5315	2633	1290	994	609	317	266
C6288	7634	7382	7259	7039	6689	6550
C7552	5841	4487	3923	2449	1491	1387
平均	2424	1769	1585	1270	1117	1039

在集成了故障模拟的确定性 ATPG 系统中,大约有一半电路的 C-B 曲线失去了单调性;而在由 FBTPG 方法构成的 ATPG 系统中,则几乎所有电路的 C-B 曲线都失去了单调性. 只有特别易测的电路(例如 C880)例外. 此现象突出地表明,在 ATPG 系统中,存在着 GAFSTG 效应,并且在由 FBTPG 方法构成的 ATPG 系统中表现得更为明显.

为了弄清 GAFSTG 效应的实质,让我们分别对表 1～表 3 进行比较分析. 根据表 1～表 3 的未测故障数可以推算出:当回溯次数为 0 时,三种情况的平均故障覆盖率分别为 87. 64 %,94. 80 %和 37. 08 %;当回溯次数为 100 时,则分别为

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

96.17 %,97.98 %和 76.83 %. 由此可见,GAFSTG 效应使 ATPG 系统的 *C-B* 曲线向纵轴(*C* 轴)正方向移动,回溯次数越小时移动量越大,而 FBTPG的移动量又比确定性算法大. 例如,回溯次数为 0 时,表 1 相对表 3 的移动量为 50.56 %,表 2 对表 3 为 57.72 %;而当回溯次数为 100 时,表 1 和表 2 的移动量分别为 19.34 %和 21.15 %. GAFSTG 效应的另一作用是使 *C-B* 曲线的变化趋于平缓. 例如,无故障模拟(表 3)时的曲线,回溯次数由 0 增至 100 时,覆盖率增加 39.75 %,而 *G-F* 二值算法(表 1)和 FBTPG 方法(表 2)则只分别增加 8.53 %和 3.18 %.

FBTPG方法的 *C-B* 曲线的起点高,变化平缓这个事实十分明确地告诉我们,GAFSTG 效应使 FBTPG 方法能在从 0 开始的较低的回溯次数范围内很快接近最高的故障覆盖率. 表 2 显示,当回溯次数达到 40 时,平均故障覆盖率已接近最高值 98.45 %(根据文献 5 记载).

GAFSTG效应产生的根源是目标故障的完全测试集之间存在交集. 例如,假设 TS_i 和 TS_j 分别是目标故障 f_i 和 f_j 的完全测试集. 如果 $TS_i \cap TS_j = \phi$,或虽然 $TS_i \cap TS_j = TS_{ij} \neq \phi$,但 ATPG 为 $f_i(f_j)$ 生成的 $T_i(T_j) \notin TS_{ij}$,则故障模拟对 f_i 和 f_j 的测试生成发挥不了调节作用. 但如果 $T_i(T_j) \in TS_{ij}$,则综合调节效应至少可使 ATPG 省去为 $f_j(f_i)$ 生成 $T_j(T_i)$ 的时间. 如果 T_i 和 T_j 都容易生成,则故障模拟的调节作用可能还不那么明显;但如果 T_i 和 T_j 之一很难生成,则故障模拟就不仅是节省了 ATPG的时间,而且还提高了故障覆盖率.

4 关于 FBTPG方法有效性的实验结果

(1) 实验 1:FBTPG与随机码的比较

表 4 给出了对 ISCAS-89 Benchmark 中 22 个规模较小的电路,分别用 FBTPG码和随机码进行故障模拟的结果. 从表中看出,绝大部分电路随机码的覆盖率太低,满足不了应用要求. 而 FBTPG方法的覆盖率率与文献 8 相比,大部分电路都略高一些. 至于测试码长度较长,是因为上述序列是对全部故障一次性生成的,未经故障模拟筛选.

(2) 实验 2:FBTPG方法与确定性算法的比较

表 1 和表 2 的结果已足以表明 FBTPG方法优于确定性算法. 回溯次数越小,FBTPG方法的覆盖率比确定性算法高得越多. 回溯次数为 100 时,还高 1.81 %. 我们也做过另一个实验,让两种方法分别按表 1 和表 2 方式,各自取不同的回溯次数,以达到最优的结果为准,则二者的覆盖率几乎相同,但 FBTPG方法所用时间只有 G-F 二值算法的 70 %,测试码数目增加了三分之一. 由此可见,FBTGP 方法的运行速度确实比确定性算法高.

(3) 实验 3:FBTPG方法处理超大规模时序电路的效果

表 5 给出了 FBTPG方法处理 ISCAS-89 中 7 个最大的电路的结果. 为便于比较,同时给出了目前进入国内市场的商用 ATPG软件中性能最好的 FlexTest 的处理结果. FBTPG的覆盖率是 FlexTest 模拟给出的. 运行时间已把 FlexTest 的模拟时间加进去了. 由于运行控制条件相同,因此二个系统的覆盖率是可比的. 运行时间前 3 个电路是可比的,后 4 个电路不可比,因二个系统采用松耦合方式,FBTPG分段生成测试,看覆盖率与 FlexTest 相当时便停止.

表 4 FBTPG与随机码的比较

电路名	FBTPG码		10000 拍随机码的	故障总数
	长度	覆盖率(%)	覆盖率(%)	
S208	361	8.29	8.29	217
S298	2343	85.71	73.05	235
S344	849	96.20	94.15	308
S349	1033	95.71	93.71	332
S382	2561	89.47	12.28	399
S386	2996	82.29	36.72	384
S400	5378	88.21	11.79	404
S420	212	6.15	5.27	455
S444	3855	90.30	15.61	474
S526	4335	81.08	11.89	555
S526*n*	4335	81.37	11.93	553
S641	964	87.57	85.87	465
S713	966	84.17	81.41	581
S820	8866	61.29	41.53	850
S832	7983	60.00	40.57	870
S838	99	5.16	4.30	931
S953	26	8.34	8.34	1079
S1196	1831	99.76	92.43	1242
S1238	2464	94.69	87.31	1355
S1423	4723	93.68	37.29	1515
S1488	8434	93.61	33.98	1486
S1494	5733	91.30	33.33	1506

表 5 FBTPG方法与 FlexTest 的比较

电路名	FBTPG			FlexTest		
	码长度	覆盖率(%)	运行时间(*m*)	码长度	覆盖率(%)	运行时间(*m*)
S5378	5062	74.65	50.9	4971	74.36	69.3
S9234.1	2006	12.35	99.8	2791	12.69	256.8
S13207.1	6054	9.32	128.7	947	9.41	133.7
S15850.1	32053	59.72	1598.1	102047	48.79	4168.1
S35932	1006	89.61	25.9	1078	89.39	409.6
S38417.1	4066	3.64	245.3	545	3.67	310.3
S38584.1	50250	72.54	3011.5	42451	72.53	4325.5

表 5 结果说明,FBTPG方法对超大规模时序电路生成测试序列的效率已优于 FlexTest.

5 结论

本文首先分析了 FBTPG方法既不同于确定性测试生成算法,也不同于随机测试生成的本质. 然后用 ISCAS-85 和 ISCAS-89 对三种方法作了大量实验,通过对实验结果的深入比较分析,发现了三种方法的 C-B 曲线的变化规律,由此总结提出了 GAFSTG效应,为 FBTPG方法的有效性提供了理论依据. 实验结果也表明,FBTPG方法为非扫描超大规模时序电路产生测试序列的效果远非随机码可比,速度和故障覆盖率超过了确定性 ATPG系统 FlexTest. 由此可见,FBTPG方法所追求的目标:以较小的回溯次数为超大规模时序电路产生高故障覆盖率的测试集,是有可能实现的.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

参考文献:

[1] P. Banerjee. Parallel algorithms for VLSI computer-aided design [J]. Englewood Cliffs, NJ: Prentice Hall, 1994, 477 - 532.

[2] B. Ramkumar, P. Banerjee. Proper test: a portable patallel test generator for sequential circuits [J]. IEEE Trans. On CAD. 1997, 16(5): 555 - 569.

[3] F. Corno, U. Glaser, P. Prinetto, M. S. Recorda, H. T. Vierhaus and M. Violante. Sym fony: A hybrid topological-symbolic ATPG exploring RT-Level information [J]. IEEE Trans. On CAD. 1999, 18(2): 191 - 202.

[4] 曾芷德. ASIC 测试生成和可测性分析系统 ATGTA [J]. 计算机学报, 1998, 21(5): 448 - 455.

[5] 曾芷德. 特大规模组合电路测试数据产生方法研究 [J]. 系统工程与电子技术, 1999, 21(6): 1 - 5.

[6] 曾芷德, 盛运焕. 论四值动态代价分析方法 [J]. 国防科技大学学报, 1988, 10(1): 1 - 7.

[7] 盛运焕, 曾芷德等. 功能块级大型数字电路测试生成的 G-F 二值算法 [J]. 计算机学报, 1983, 6(1): 32 - 41.

[8] T. M. Niermann, W. Cheng, J. H. Patel. PROFFS: A fast. memory-efficient sequential circuit fault simulator [J]. IEEE Trans. on CAD, 1992, 11(2): 198 - 207.

作者简介:



曾芷德　1941 年生, 1965 年毕业于中国人民解放军军事工程学院, 现任国防科技大学计算机学院教授. 主要研究领域为数字系统容错与故障诊断.



曹贺锋　1976 年生, 1997 年获国防科技大学计算机专业学士学位, 现在国防科技大学攻读计算机专业硕士学位. 研究领域为测试生成、并行计算.

(上接第 139 页)

参考文献:

[1] S. Mallat, and W. L. Hwang. Singularity detection and processing with wavelets [J]. IEEE Trans. Information Theory, 1992, 38(2): 617 - 643.

[2] M. J. Kirton, and M. J. Uren. Noise in solid-state microstructures: a new perspective on individual defects, interface states and low-frequency (1/f) noise [J]. Advances in Physics, 1989, 38(4): 367 - 468.

[3] B. B. Mandelbrot and H. W. Van Ness. Fractional Brownian motions, fractional noises and applications [J]. SIAM Rev., 1968, 10: 422 - 436.

[4] B. Pilgram, and D. T. Kaplan. A comparison of estimators for 1/f noise [J]. Physica D, 1998, 114: 108 - 112.

[5] J. Feder. Fractals. New York and London: Plenum Press. 1988: 21.

[6] Du Lei, and Zhuang Yiqi. Microscopic Simulation on Trapping 1/f Noise in MOS Devices [M]. In proc. Noise in Physical Systems and 1/f Noise, Hong Kong, 1999.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net