

特大规模组合电路高速测试生成系统 ATGTA-1*

曾芷德

(国防科技大学计算机系 长沙 410073)

摘要 针对特大规模组合电路和全扫描设计电路提出了一种高速测试生成方法,并建成了相应的测试生成系统 ATGTA-1。该系统采用有限回溯测试模式产生方法生成测试码,采用 n (机器字长) 个测试码并行的单故障传播方法模拟验证测试覆盖。测试生成与故障模拟为 n 对 1 紧耦合集成方式。该系统运行 10 个 Benchmark 电路,取得了低测试长度、高故障覆盖、高效率的良好效果。

关键词 全扫描设计,有限回溯测试模式产生, n 对 1 紧耦合集成方式,测试码并行,单故障传播,特大规模组合电路

分类号 TN47, TN407

The High Speed Test Generation System ATGTA-1 for Very Large Scale Combination Circuit

Zeng Zhide

(Department of Computer, NUDT, Changsha, 410073)

Abstract This paper presents a high speed test generation method specifically for very large scale combination circuit and full scan-designed circuit. This method is used together with the finite backtracing test pattern generation method to generate test code, and then simulates to validate the fault covering by means of single fault propagating method with n (machine word) test code parallelised. The mode for test generation and fault simulation is n to 1 tight coupled integrating mode. With testing the method with 10 benchmark circuits, the result is good for low test pattern length, high fault coverage and high efficiency.

Key words full scan-design, finite backtracing test pattern generation, n to 1 tight coupled integrating mode, test code parallelism, single fault propagation, very large scale combination circuit

组合电路的测试生成和故障模拟方法之所以具有普遍意义,是因为它可用来解决全扫描 VLSI 的测试数据产生。从 80 年代中期以来,国内外学者就这一问题展开了广泛深入的研究。研究集中在两方面:一是如何对组合电路确定性测试生成算法进行加速,二是对组合电路如何使用新的快速故障模拟方法。前者以 SOCRATES^[1] 系统为代表。它在 FAN^[2] 算法基础上对唯一敏化和反向隐含方法提出了有效改进。该系统对 10 个 Benchmark 电路的处理达到了较好的效果。后者以 PPSFP^[3] 系统为代表。该系统借助理论分析和实验数据论证了测试码并行、单故障传播的故障模拟方法对于十万门量级的特大规模全扫描电路用随机码模拟的有效性。其它快速故障模拟方法^[4,5] 也给人们十分强烈的信息:组合电路随机码故障模拟所需 CPU 时间仅为随机码数目和电路规模的线性增长函数。而且前一条直线的斜率小于 1。由此看来,超大规模和特大规模全扫描设计的测试数据产生问题用随机码的快速故障模拟似乎已经解决了。但是,从应用角度看,留下了两个必须解决的难题:一是随机码序列太长,二是有些对随机码不敏感的电路,故障覆盖率难以满足要求。

关于测试序列数据长度,任何一家集成电路制造商都有严格限制。要想生成长度符合实用要求的测试码,大约只能借助确定性测试生成算法的优势,舍此恐难找更有效的测试数据压缩方法。至于对随机码不敏感的电路,则更得靠确定性测试生成方法提高故障覆盖率。

* 国家自然科学基金资助项目

1998年6月30日收稿

第一作者:曾芷德,男,1941年生,教授

但是,对于超大规模和特大规模组合电路,采用确定性算法生成测试也有二个难题:一是测试生成时间复杂性的 NP 完全性困扰,二是测试生成和故障模拟传统的 1对1 紧耦合集成方式使得测试生成系统无法有效地用测试码并行、单故障传播的快速故障模拟方法。

为了对付上面提到的难题,本文对特大规模组合电路提出了一种新的测试生成方法:采用有限回溯测试模式产生方法生成测试码,一次生成 k 个测试码,再配上 $n-k$ 随机码 (n 为机器字长, $k < n$), 然后进行 n 个测试码并行的单故障传播,以验证故障覆盖。我们在 ATGTA 系统^[6]中实现了此方法,形成了组合电路测试生成系统 ATGTA-1。该系统较大地提高了 ATGTA 系统对组合电路的处理效率,能高效率地为 10 个 Benchmark 电路产生高故障覆盖率的测试集,测试集长度完全满足实用要求。

1 有限回溯测试模式产生方法 FBTPG

理论上已经证明,大规模电路测试生成的时间复杂性一般是 NP 完全性问题^[7],对组合电路的实验结果表明,测试生成时间随电路规模的增大呈平方至立方的指数增长^[8]。由此看来,现有的组合电路确定性测试生成算法将无力处理特大规模(十万门量级)电路。为此,对特大规模组合电路,我们建议采用有限回溯测试模式产生方法 FBTPG 生成其测试集。FBTPG 的基本思想已在文献[6]的 ATGTA 系统中进行了初步描述,它对四万门规模的 VLSI 的有效性也由 ATGTA 所证实。关于 FBTPG 方法一般规律的理论探讨,我们将另文讨论。本节仅仅讨论它能否处理十万门量级的特大规模组合电路。为此,我们需要分析 FBTPG 的时间复杂性与确定性算法有何不同。

ATGTA-1 系统用的确定性测试生成算法是 G-F 二值算法^[9],FBTPG 方法是以 G-F 二值算法为母算法的。G-F 二值算法生成一个指定目标故障的测试码所需时间。

$$T_1 = t_1 + t_2 + t_3 \quad (1)$$

式(1)中 t_1 为预处理时间。它包括计算故障电路中每条引线的置 0 置 1 代价。计算故障级距 L_{fv} 和已选定的输出扇入锥中所有引线的树头级距 L_{hv} , 以及对锥中引线进行跳变分析所需的时间^[10]。 t_1 在 T_1 中所占的比例很小,它是电路规模的线性增长函数。即复杂性为 $O(N)$ 。

t_2 为建立测试码方程,即故障敏化所需时间。一般有:

$$t_2 = p \cdot m_i \cdot L_{ij} \cdot t_0 \quad (2)$$

式(2)中, t_0 为故障敏化处理一个功能块所需的平均时间,它与电路规模无关。 p 为与目标故障逻辑相关的初级输出 PO 数, m_i 为目标故障到第 i ($i = 1, 2, \dots, p$) 个 PO 的通路数, L_{ij} 为从目标故障线沿第 j ($j = 1, 2, \dots, m_i$) 条道路到达第 i 个 PO 所经的逻辑级数。平均来看,可以认为 p, m_i, L_{ij} 都是电路规模增长的线性函数,因此, t_2 的最高复杂性为 $O(N^3)$ 。

t_3 为解方程,即反向追踪、相容性运算所需时间。当解方程出现矛盾赋值时,最坏情况下需要对各种赋值选择进行穷举搜索。随着电路规模的增大,这种穷举搜索消耗的 CPU 时间是指数膨胀的。我们不妨假设一种最简单情况:方程中包含 N 个括号连乘,每个括号都有两项供选择,于是遍历所有选项组合的数目为 2^N ,可见 t_3 的复杂性最小为 $O(2^N)$ 。

综上所述可见,对超大规模和特大规模电路, t_3 的复杂性将支配 T_1 的复杂性。而对 FBTPG 方法来说,由于限定对任何目标故障只进行有限次的回溯,因此,不可能出现穷举搜索, t_3 的复杂性将远远小于 $O(2^N)$, t_2 的复杂性也将远远小于 $O(N^3)$,这就为它处理特大规模电路奠定了基础。有人可能会问,确定性算法在执行时不也是限制回溯次数吗?二者有何不同呢?

它们有二点主要不同:

① 确定性算法执行时,限制回溯次数是一种防止单个故障过度消耗 CPU 时间的不得已的策略。与算法的完备性是不相容的。一般来说,回溯次数太少会降低故障覆盖率,而 FBTPG 方法中的回溯次数是执行过程的控制参数,其多少与故障覆盖率高低的对应关系不像确定性算法那样严格。

④ 在确定性算法中,需进行故障模拟的测试码是指一定能检测目标故障的码子。失败故障(aborted fault)没有求出测试码,不需要进行故障模拟。这使确定性算法处理失败故障的时间白白浪费了。而 FBTPG 方法则把在限定回溯次数之内产生的码子都作为被测电路的测试码进行故障模拟,避免了上

述浪费。由此可见, 确定性算法着眼于是否求出了单个目标故障的测试码, 而FBTPG方法着眼于被测电路的目标故障集的故障覆盖率。二者的出发点是不一样的。

那么用FBTPG方法所生成的码子的故障覆盖效果会不会与随机码差不多, 此问题已由文献6的理论分析和实验结果作了初步回答。ATGTA系统采用FBTPG方法生成的测试码, 用一对一的紧耦合集成方式执行同时故障模拟, 确定每个码子的故障覆盖。其中测试生成速度和故障覆盖率可与SOCRATES系统相媲美。这表明FBTPG方法在推导测试码的精确性方面决非随机码所及。

本文与PPSFP方法一样, 也采用测试码并行的单故障传播方法^[11]进行故障模拟, 但测试码由FBTPG方法提供。后面对10个Benchmark电路的实验结果表明, 本文所达到的故障覆盖率比用随机码的PPSFP高, 与SOCRATES系统相当, 测试序列长度比PPSFP短得多, 完全可满足实用要求。而系统的总运行速度与SOCRATES相比也是高的。

综上所述可见, FBTPG方法实质上是用确定性算法的精确性来减少随机码的盲目性; 用随机码的高速度来避开确定性算法的指数复杂性。因此, 用它处理10万门量级的特大规模全扫描电路是完全有可能的。

2 测试生成 (TG) 与故障模拟 (FS) 的集成方式

TG与FS传统的集成方式是一对一的紧耦合方式, 即二者在同一系统中, 采用交替运行方式。每生成一个测试码, 立即执行FS, 并删除所覆盖故障, 此过程一直重复执行, 直到目标故障表被扫描一遍为止。这种方式有两大优点: 一是TG的无效计算量少; 二是所生成的测试码数量少, 冗余测试码很少。例如对10个Benchmark电路, 单个电路测试码最多不会超过300。但有一大问题是无法采用测试码并行的快速故障模拟方法。对于特大规模电路, 此问题也是十分严重的。例如SOCRATES系统, 对C7752的FS时间为158秒, 按最低的复杂性 $O(N^2)$ 计算, 对10万门规模的组合电路执行FS将大约需要27小时, 而PPSFP的估算是半小时。因此, 当用FBTPG方法处理特大规模的组合电路时, 必须设法用上测试码并行单故障传播的快速故障模拟方法。为此, 有必要改变TG与FS传统的1对1紧耦合方式, 本文建议采用 n (机器字长) 对1紧耦合集成方式, 即每生成 n 个测试码之后才执行FS, 删除 n 个测试码所检测的故障, 之后重复上述过程, 直到目标故障表被扫描一遍为止。

上述集成方式可保证ATGTA-1既能用FBTPG方法, 又能用快速FS。但有可能使TG做一些重复工作降低效率。为此, 我们采用 $k+(n-k)$ 模式一次只生成 $k(k+n)$ 个FBTPG测试码, 再配上 $n-k$ 随机码去执行FS。 k 可根据需要改变, 以兼顾TG的效率、故障覆盖率和测试集的大小。实践表明, $k+(n-k)$ 的TG模式和 n 对1紧耦合集成方式可使ATGTA-1系统获得相对最佳效果。

3 实验结果

下述实验结果中的时间为SUN20工作站的CPU秒。

实验1: 1对1方式与32对1方式性能比较

TG与FS1对1的紧耦合集成方式与32对1紧耦合集成方式运行10个Benchmark电路效果分别列在表1中。两种方式的TG都是FBTPG方法。在32对1方式下, TG用的是 $32+0$ 模式, 即32个测试码全都是FBTPG码。故障模拟用的是文献[11]所描述的方法。该方法综合了国际上目前比较有效的快速故障模拟技术, 如测试码并行、单故障传播、划分无扇出区等等。在1对1方式下, 用的是ATGTA中实现的同时故障模拟, 但对数据结构进行了优化, 使效率平均提高了2.3倍。

分析表1数据可知, 32对1方式的TG时间按10个电路平均计算, 为1对1方式的2.4倍, FS时间平均只有1对1方式的8.7%, 总的运行时间平均降低7.4%, 故障覆盖率平均降低0.12%, 测试码数量平均增加84.6%。由此可见32对1方式总的效果比1对1方式好, 由于测试码总数量仍然很少, 故增加一点测试码可不予考虑。

实验2: 32对1方式性能的改善

实验1中, 32对1方式的TG时间增加, 表明一次性所生成的32个测试码中, 含有重复无效码。为避

免重复码, 我们采用 $1+31$ 的TG模式, 即一次只生成一个FBTPG码, 再配31个随机码进行故障模拟。表2给出了在此TG模式下, TG与FS采用32对1集成方式时, 对10个Benchmark电路的运行结果。

表1 两种耦合集成方式比较

Tab. 1 Comparison between two couple integration fashion

电路名	1对1方式					32对1方式				
	时间 (s)			测试码数	故障覆盖率%	时间 (s)			测试码数	故障覆盖率%
	TG	FS	TO			TG	FS	TO		
C432	1.47	1.27	2.74	74	99.25	1.82	0.14	1.96	108	99.25
C499	11.17	1.50	12.67	63	98.96	13.94	0.10	14.04	126	98.32
C880	1.46	3.54	5.00	79	100.00	1.92	0.21	2.13	128	100.00
C1355	9.59	6.13	15.72	108	99.50	17.82	0.55	18.37	224	99.50
C1908	23.92	21.16	45.08	148	99.48	53.92	1.34	55.26	240	99.34
C2670	20.97	39.65	60.36	147	95.72	38.00	2.05	40.05	308	95.50
C3540	37.21	98.8	136.01	244	96.25	71.93	5.53	77.46	384	96.14
C5315	19.83	135.78	155.61	186	98.96	43.51	7.23	50.74	281	98.78
C6288	13.84	34.21	48.05	30	99.56	113.02	15.64	128.66	84	99.56
C7552	78.72	381.06	459.78	276	98.21	158.81	15.90	174.71	467	98.29

表2 $1+31$ 的TG模式Tab. 2 TG made of $1+31$

电路名称	时间 (s)			测试码数	故障覆盖率%	未测故障数
	TG	FS	TO			
C432	0.31	0.22	0.53	352	99.25	4
C499	2.10	0.22	2.32	576	98.96	8
C880	0.36	0.41	0.77	640	100	0
C1355	2.33	0.95	3.28	864	99.50	8
C1908	13.82	2.93	16.21	1344	99.53	10
C2670	24.86	4.29	29.15	2688	95.72	120
C3540	17.03	8.75	25.78	1440	96.22	138
C5315	11.29	8.28	20.11	960	98.94	60
C6288	30.67	16.11	46.78	96	99.56	34
C7552	82.14	24.93	107.07	3904	98.30	138

分析表2与表1的1对1方式的结果可知, 在 $1+31$ 的TG模式下, 32对1方式的TG时间比1对1方式平均降低20.4%, 总运行时间平均降低69.1%, 二者的故障覆盖率几乎相同。这说明32对1方式的效果已明显优于1对1方式。其总运行速度与文献[1]中SOCRATES系统第一步加随机码和不加随机码的两种情况的速度(见文献1的表1和表2)相比, 也是快的。

从表2还得出, 在 $1+31$ 的TG模式下, 32对1方式的测试码数量平均增加了7.8倍。按实用要求, 测试码数目稍多了一些, 例如C2670和C7552已超过了2K。此问题可采用 $k+(n-k)$ 的TG模式解决。

实验3: $k+(n-k)$ 的TG模式中, k 对测试码数量的影响

表3就C2670和C7552两个对随机码不敏感电路, 给出了TG与FS为32对1方式时, 采用 $k+(n-k)$ TG模式, k 对测试码数目影响的实验结果。从表中数据可见: 测试码数目随着 k 从1变到6迅速减少,

故障覆盖率几乎未变, C2670的总运行时间也未变, C7552的总运行时间增加了约13%。

表3 $k+(n-k)$ 的 TG 模式

Tab.3 TG mode of $k+(n-k)$

K	C2670			C7552		
	测试码数	故障覆盖率%	TO (S)	测试码数	故障覆盖率%	TO (S)
1	2688	95.72	29.35	3936	98.30	107.07
2	1536	95.72	29.46	2528	98.30	113.51
3	1024	95.72	29.32	1824	98.26	113.09
4	768	95.72	28.75	1534	98.28	116.81
5	640	95.50	29.65	1344	98.25	122.50
6	544	95.50	29.86	1088	98.26	121.85
7	512	95.57	30.40	991	98.26	125.65
8	476	95.57	31.29	893	98.24	138.18
9	416	95.50	32.01	859	98.24	146.71
10	412	95.36	33.49	793	98.23	137.75

实验3表明, TG 与 FS 采用 n 对1的耦合集成方式的测试数据生成系统中, TG 应采用 $k+(n-k)$ 模式的 FBTPG 码, 以便通过 k 值变化, 使测试序列长度满足实用要求, 而总运行时间和故障覆盖率与 $k=1$ 时相比, 不会起大的变化。

4 结论

从本文的理论分析和实验结果可以得出下列结论: 采用 $k+(n-k)$ 的 TG 模式生成 k 个 FBTPG 码, 再配以 $n-k$ 个随机码, 按 TG 与 FS 的 n 对1集成方式进行 n 个测试码并行的单故障传播快速故障模拟方法, 是一种较好的组合电路测试数据生成策略选择。由于 FBTPG 方法避开了确定性算法的 NP 完全性困扰, 充分利用了确定性算法的精确性和随机码的高速度, 因此, 本文提出的方法完全有可能处理十万门量级的特大规模组合电路和全扫描设计电路。

关于 FBTPG 方法, 本文表1中1对1方式下的 TG 时间表明, 它对10个 Benchmark 电路生成测试码的效果比 SOCRATES 系统一点不差; 本文表3的数据说明, FBTPG 方法处理对随机码不敏感的电路的综合效果远非随机码可比。而表2的结果表明, 当把 FBTPG 方法与随机码相结合时, 效果更加令人鼓舞。因此, 我们有理由相信, FBTPG 方法一定能在特大规模组合电路的测试数据生成中发挥重要作用。

参考文献

- Schulz M H, Trischler E, Sarfert T M. SOCRATES: a highly efficient automatic test pattern generation system. In: Proc IEEE Int Test Conf, Washington, 1987, 1016~1025
- Fujwara H, Shimono T. On the Acceleration of Test Generation Algorithms. IEEE Trans. on Computers (C-32), 1983, 1137~1144
- Waicukauskis J A, Eichelberger E B, Forlenza D O, Lindbloom E, Thomas M cCarthy. Fault Simulation for Structured VLSI. VLSI Syst. Design, Dec. 1985, 20~32
- Antreich K, Schulz M. Accelerated Fault Simulation and Fault Grading in Combinational Circuits. IEEE Trans. on Computer-Aided Design, vol. CAD-6, September, 1987, 704~712
- 石茵, 魏道政. 平行码临界路径跟踪法中的加速技术. 全国 CFTC6 论文集, 哈尔滨, 1995, 178~184
- 曾芷德. ASIC 测试生成和可测性分析系统 ATGTA. 计算机学报, 1998, 21(5), 448~455
- Fujwara H. 逻辑测试和可测性设计. 北京: 计算机技术编辑部, 1990
- Lala P K. Fault tolerant and fault testable hardware design. London: PHI Inc, 1985
- 盛运焕, 曾芷德等. 功能块级大型数字电路测试生成的 GF 二值算法. 计算机学报, 1983, 6(1), 32~41
- 曾芷德编著. 数字系统测试与可测性. 长沙: 国防科技大学出版社, 1992
- 刘胜利, 曾芷德. 组合电路故障模拟加速方法的研究与实现. 桂林, 全国第十届 CAD&CG 会议论文集, 1998: A1~A5