

一种基于变长数据块相关性统计的 测试数据压缩和解压方法

欧阳一鸣, 成丽丽, 梁华国
(合肥工业大学计算机与信息学院, 安徽合肥 230009)

摘 要: 为了解决系统芯片(SoC)测试过程中自动测试设备(ATE)在存储空间以及带宽等方面所面临的问题, 本文提出了一种新的基于变长数据块相关性统计的测试数据压缩和解压方法. 以测试向量为单位, 先用算法确定一个具有最好相关性的数据块作为该向量的参考数据块, 再利用它与该向量中数据块的相关性进行压缩. 且每个向量的参考数据块长度相互独立. 其解压结构只需要一个有限状态机(FSM)、一个 5 位暂存器和一个与参考数据块等长的循环扫描移位寄存器(CSR)即可, 硬件开销小, 对 ISCAS 89 标准电路 Mintest 集的压缩结果表明, 本文提出方案较同类编码方法有更高的压缩效率.

关键词: 数据块相关性; 统计; 参考数据块; 数据压缩

中图分类号: TP302 **文献标识码:** A **文章编号:** 0372-2112 (2008) 02-0298-05

A New Test Data Compression and Decompression Technique Based on Statistic Relativity of Variable Length Data Block

OUYANG Yi ming, CHENG Li li, LIANG Hua guo
(School of Computer and Information, Hefei University of Technology, Hefei, Anhui 230009, China)

Abstract: We present a new test data compression and decompression method based on statistic relativity of variable length data block, to resolve the storage and bandwidth problems faced by the automatic test equipment (ATE) in system on a chip (SOC) during test. For each test vector, determine a data block with best relativity to be the referenced data block of it, then compress the vector by the relativity between its data blocks and the referenced data block. Besides, the length of referenced data blocks is independent from each other. Its decompression architecture requires only a finite state machine (FSM), a five bit storing device temporarily and a cyclical scan register (CSR) whose length is the same as the referenced data block, so the hardware cost in proposed scheme is small. Experimental results for ISCAS-89 benchmarks show that our new technique can provide a higher compression ratios than other compression algorithms.

Key words: relativity of data block; statistic; referenced data block; data compression

1 引言

随着系统芯片(SoC)的集成度越来越高, IP 核的增加及其多样性使得测试的复杂性越来越高. 测试数据量急剧增长, 这导致了传统外部自动测试设备(ATE)面临着存储空间不足、I/O 带宽有限以及测试时间过长等严峻的问题, 测试成本越来越高. 测试问题已经成为 SoC 发展所面临的一个主要问题.

测试数据压缩是解决上述问题的有效方法, 通过压缩可以减少数据的传输时间以及降低对 ATE 存储容量的要求. 目前, SoC 测试技术分为 BIST(内建自测试)技

术和 BOST(外建自测试)技术两种, BIST 通过在芯片内部集成测试向量生成电路以及测试控制和测试响应比较电路, 使得测试变得快速、高效^[1,2]; BOST 则将传统的 BIST 中的部分测试功能转移到被测试芯片之外, 在完成 BIST 的主要功能的同时又不需要将整个测试硬件放在芯片上, 通过对芯片内外测试源的优化分配, 能有效地降低测试成本, 如基于测试源划分的思想^[3], 将芯片上的所有测试芯核的测试向量统一压缩存放在 ATE 中, 由芯片上的测试向量生成与控制单元负责统一生成所需的测试向量, 利用自测试环境控制测试向量应用到每个测试芯核中. 为此, 常采用编码压缩技术^[4-11]来压

缩原测试集 T_D , 将压缩后的测试集 T_E 存放在 ATE 上, 测试时再通过解码器将 T_E 还原成 T_D , 这样可以降低对 ATE 的存储设备以及带宽的要求, 从而有效地降低了测试成本. 目前有许多不同的编码压缩方法, 它们使用不同的策略来压缩数据, 如: 交替连续长度码^[4]、FDR 码^[6]、PRL^[7]、Golomb 码^[8]、统计码^[9]、哈夫曼编码^[10] 以及 9C 编码^[11] 等.

为了充分发挥测试集中无关位多, 且各测试向量之间的无关位数量差异大的特点, 以及进一步提高数据压缩率, 本文提出了一种基于变长数据块相关性统计的测试数据压缩和解压方法, 运用统计^[9, 10] 和数据块相关^[7] 的思想, 针对每个测试向量, 用算法确定该向量的参考数据块 (以下用 $P_{X(k)}$ 表示), 使其和该向量中的各数据块相关性最好, 并且用它来对该向量进行编码得到的压缩码位数最少. 此方案利用每个测试向量内部数据块之间的相关性进行压缩, 突破了在整个测试集范围内操作所遇到的各数据块确定位之间相互约束而导致的相关性差的限制. 这种化整为零的方法大大提高了数据块间的相关性, 从而明显提高了压缩率. 该方案的解压体系结构只需要一个 FSM、一个 5 位暂存器和一个与 $P_{X(k)}$ 等长的循环扫描移位寄存器 (CSR) 即可, 此外该方法的实现与测试数据无关, 故具有普遍性.

2 变长数据块相关性统计压缩方法

2.1 方法思想

首先, 数据块相关是指不同的数据块之间存在着相容或是反向相容的关系. 数据块相容是指两个数据块对应位相同或是至少有一位是无关位; 反向相容就是两个数据块对应位相反或是至少有一位是无关位.

在测试集中, 不同的测试向量所包含的无关位数量有很大差异. 很明显, 当确定位较多时, 数据块长度越短相关性越好, 而无关位较多时, 数据块长短对相关性影响很小, 此时数据块长度越大压缩率越高, 故本文提出的方案中每个向量的 $P_{X(k)}$ 长度都由该向量自身特征决定, 因此压缩时必须确定 $P_{X(k)}$ 的长度区间, 当 k 值偏小时, 压缩率明显很低, 在本方案的压缩过程中, 取 k 的下限为 5, 随着 k 的增长, 压缩率随之增加, 但当 k 增加到一定程度时, 由于数据块之间的相关性变差, 压缩率的提高不再明显甚至不再提高, 并且随着 k 值的增加, 选择查找 $P_{X(k)}$ 算法的时间复杂度呈指数级增长, 这三个因素之间相互制约. 图 1 给出了 k 区间的上限与压缩率的增加量之间的关系图, 此处的压缩率增加量是指: 取当前 k 区间上限所得压缩率, 与取前一个上限所得压缩率的差值 (例如, k 区间上限为 10 时所对应的压缩率增加量就是: k 区间为 $[5, 10]$ 时所得的压缩率与 k 区间为 $[5, 9]$ 时所得的压缩率的差值). 由图可

知, 随着 k 区间上限的增长, 压缩率的增加量呈递减趋势; 图 2 是 k 区间的上限与 $P_{X(k)}$ 选择算法的时间复杂度之间的关系. 综合这两个图可知, 当 k 的取值在 5 到 20 之间时较为合理.

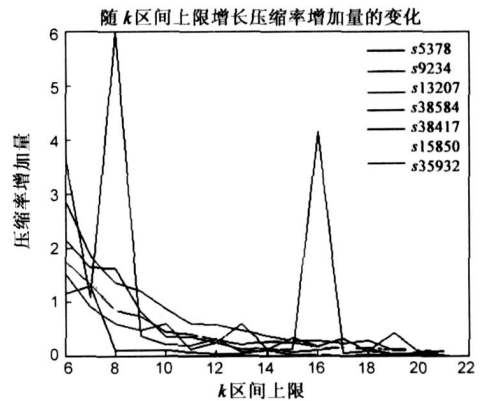


图 1 k 区间的上限与压缩率的增加量之间的关系图

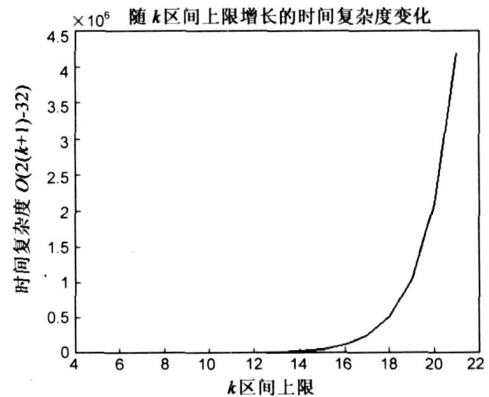


图 2 k 区间上限与参考数据块选择算法的时间复杂度之间的关系

方案具体描述: 针对每个测试向量, 在 k 的区间内穷举查找出其 $P_{X(k)}$, 此过程分两步完成: (1) 确定一个 k 值 ($5 \leq k \leq 20$), 将向量划为若干 k 位块, 末尾不足 k 位的用无关位补充, 在当前 k 位的所有二进制数据块中, 统计出它们与该向量中数据块相关性频率最高的一块, 计算出由该数据块作为参考对此向量编码得到的编码位数; (2) 比较 k 取不同值时得到的编码位数, 编码位数最少时对应的数据块, 便是该向量的参考数据块 ($P_{X(k)}$). 之后, 将表示 k 值的信息块 (因 $5 \leq k \leq 20$, 故用 5 位表示 k , 以下称为“ k 信息块”) 和各向量的 $P_{X(k)}$ 加在各向量编码的开始. 编码规则如下: 向量中的数据块与 $P_{X(k)}$ 相容用 ‘0’ 替换, 反向相容用 ‘10’ 替换, 而不相关时则在原数据块前面加上 ‘11’, 原数据块保留. 这就完成了一个测试向量的压缩, 对每个向量作上述循环, 最终达到对整个测试集 T_D 压缩.

2.2 算法描述

该方案不需要先对测试集进行优化或预处理, 而是直接面向原测试集, 设某测试集中向量个数为 m , 向

量长度为 $l, p_{x(k)} (0 \leq x \leq 2^k - 1, 5 \leq k \leq 20)$ 表示第 x 个长度为 k 的数据块(它是选择查找 $P_{X(K)}$ 时的过程变量, 经过统计和比较最终得出的最优情况便是该向量的参考数据块 $P_{X(K)}$), 对任意向量 $V_j (1 \leq j \leq m)$, 其所包含的数据块 $block_i$ 与 $p_{x(k)}$ 关系定义如下: 若 $block_i \oplus p_{x(k)} = 0(x) \dots 0(x)$, 则表示两块相容, 反之, 若 $block_i \oplus p_{x(k)} = 1(x) \dots 1(x)$, 则表示两位反向相容, 用 $C(p_{x(k)}, V_j)$ 、 $I(p_{x(k)}, V_j)$ 和 $R(p_{x(k)}, V_j) = C(p_{x(k)}, V_j) + I(p_{x(k)}, V_j)$ 分别表示向量 V_j 中与 $p_{x(k)}$ 相容、反向相容以及相关的数据块数目, C_{ij} 表示对一个向量编码后所得的编码位数, 具体描述如下:

- (1) 初始值 $j = 1$;
- (2) 初始值 $k = 5$;
- (3) 向量 V_j 包含的数据块数为 $block_num = \lceil l/k \rceil$ 将向量 V_j 划分成若干 k 位块 $block_1 block_2 \dots block_{\lceil l/k \rceil}$ 初始值 $x = 0$, 此时, $p_{x(k)}$ 初值为 $00 \dots 0$ (k 位全 0 数据块);
- (4) 初始值 $i = 1$;
- (5) 若 $block_i \oplus p_{x(k)} = 0(x) \dots 0(x)$, 则 $C(p_{x(k)}, V_j)++$, $R(p_{x(k)}, V_j)++$; 若 $block_i \oplus p_{x(k)} = 1(x) \dots 1(x)$, 则 $I(p_{x(k)}, V_j)++$, $R(p_{x(k)}, V_j)++$;
- (6) $i++$, 如果 $i \leq \lceil l/k \rceil$ 则跳至(5), 否则 $x++$, 若 $x \leq 2^k - 1$, 跳至(4);
- (7) 取 $p_{x(k)}$ 满足 $R(p_{x(k)}, V_j) = \max\{R(p_{x(k)}, V_j), 0 \leq x \leq 2^k - 1\} \wedge C(p_{x(k)}, V_j) \geq I(p_{x(k)}, V_j)$ (由于在编码

过程中, 相容时用“0”替换, 而反向相容时用“10”替换, 故当相关性频率相等时, 应该取相容频率高的情况);

- (8) $C_{ij} = 5 + k$, 如果 $block_i \oplus p_{x(k)} = 0(x) \dots 0(x)$, 则 $C_{ij}++$; 若 $block_i \oplus p_{x(k)} = 1(x) \dots 1(x)$, 则 $C_{ij} + 2 \Rightarrow C_{ij}$; 否则 $C_{ij} + 2 + k \Rightarrow C_{ij}, k++$, 如果 $k \leq 20$, 跳至(3);
- (9) 取 $C = \min\{C_{ij}, 5 \leq k \leq 20\}$ 对应的 $p_{x(k)}$ 作为该向量的 $P_{X(K)}$, 它对应的编码便是该向量的压缩码;
- (10) $j++$, 如果 $j \leq m$, 跳至(2), 否则结束.

2.3 编码实例

为了更直观的说明该方法, 在测试集中抽出一个测试向量进行编码举例, 由于测试集中向量长度过大, 截取一段数据串如下: $1x00x1110x1xxxx0x1x01xxx1xxx11x1x1000110111x0xxxxx0x1xx1xxxx0xx01x0xxxx(72 \text{ bits})$ 由该方法可知, 首先要找出它的 $P_{X(K)}$, 用其对该数据串进行编码, 执行 2.2 中算法的(2)~(9), 得到其 $P_{X(K)} = 11001111$ (此时 $k = 8$), 编码时在压缩码前加上 ‘01000’ (表示 $k = 8$, 即“ k 信息块”) 和 $11001111'$ ($P_{X(K)}$), 此时数据串被划分成若干 8 位块, 如图 3 所示, 第一块为 ‘ $1x00x111$ ’ 与 $P_{X(K)}$ 相容, 则用 ‘0’ 替换; 第二块为 ‘ $0x1xxxx0$ ’ 与 $P_{X(K)}$ 反向相容, 则用 ‘10’ 替换; 而第五块 ‘ 10001101 ’ 与 $P_{X(K)}$ 不相关, 则在其前面加上 ‘11’, 原数据块不动; 其他的情况均包含在上述情况中. 由结果可知, 该方案将原来的 72 位数据串压缩到只有 33 位, 从而达到了有效压缩测试集的目的.

	1x00x111	0x1xxxx0	x1x01xxx	1xxx11xx	10001101	11x0xxxx	xx0x1xx1	xxxx0xx0	1x0xxxx
本方案压缩结果:	0	10	0	0	110001101	0	0	10	0
	01000	11001111		01000110001101000100	(33bits)				
	(k 信息块)	($P_{X(K)}$)		(向量压缩码)					

图 3 编码实例

3 解码体系结构及原理

本方案的解码体系结构的设计基于一个 FSM, 由它控制整个解码过程, 此外, 还有一个 K 位的 CSR、一个 5 位暂存器和三个计数器, 故其解码结构简单且硬件开销较小. 如图 4 所示, 在解压结构中, ‘en’ 和 ‘bit_in’ 分别是输入使能和位输入信号, 当 FSM 准备就绪时, 输入压缩的数据. 5 位暂存器用来保存每个向量的 k 信息块, 利用它将 k 信息块装载到 5 位计数器, ‘ren’ 则用来控制暂存器的时钟信号, 使其只在接收每个向量的 k 信息块时有效. K -bits CSR 用来保存各向量的 $P_{X(K)}$, 用于对与 $P_{X(K)}$ 相关的编码数据 ‘0’ 和 ‘10’ 进行译码. ‘flag1’ 是多路选择器的 0、1 通道选择信号, 控制 CSR 选择接收 ‘data_in’ 上的 $P_{X(K)}$ 或是其返回值. 三个计数器的功能分别如下: 其中, ‘dec1’、‘dec2’ 和 ‘dec3’ 分别被用作减一计数器, 而 ‘rs1’、‘rs2’ 和 ‘rs3’ 分别被用作指示计数器的复位 ‘0’ 状态. (1) 5 位计数器的功能

分为两部分, 初始时作为一个模 3 计数器, 用于在向量解码开始时, 控制将 5 位的 k 信息块移入到 5 位暂存器, 之后, 每当 ‘rs1’ 有效, ‘cen’ 变为有效, 暂存器将 k 信息块装载到该计数器中, 由它控制解码数据块的长度, 直到一个向量解码结束, 此时 ‘rs3’ 变为有效, 该计数器复位到初始的模 3 计数状态; (2) 由于本方案中每个向量生成的 $P_{X(K)}$ 长度不等, 则 CSR 的长度选取必须为 $P_{X(K)}$ 长度的最大值, 为了保证解码的正确, 用 5 位的 CSR 计数器控制每次解码时 CSR 循环 K 次, 使得当前 $P_{X(K)}$ 在 CSR 中处于解码输出状态; (3) $\lceil \log_2 l \rceil$ 位的向量长度计数器用来控制每个向量解压, 每解码一位做一次减一操作, 直到 ‘rs3’ 有效, 表示此向量解码结束, 这就屏蔽掉了编码时每个向量末可能补充的无关位; ‘flag2’ 是 MUX 的 0、1 通道选择信号. 同时 ‘dec3’ 控制扫描时钟 ‘scan_clk’ 是否有效, 整个解码过程中由 ‘scan_clk’ 控制解压数据是否进入扫描链.

解码器的工作原理:

(1) ‘en’ 为高, ‘dec1’ 和 ‘ren’ 分别为高, 控制 ‘bit_in’ 移入 k 信息块到 5 位暂存器, 直到 ‘rs1’ 为高时, ‘ren’ 变为低, k 信息块输入结束, 同时 ‘cen’ 变为高, k 信息被装载进入 5 位计数器。

(2) 接着 ‘dec1’ 和 ‘dec2’ 分别为高, ‘flag1’ 为低, 将 ‘bit_in’ 上的 $P_{X(K)}$ 移入 CSR, 直到 ‘rs1’ 为高, ‘cen’ 变为高, 暂存器中的数据再次装载进入 5 位计数器, 同时 ‘flag1’ 变为高, ‘en’ 变为低, 直到 ‘rs2’ 为高, CSR 中 $P_{X(K)}$ 循环到解码输出状态, 这时 ‘en’ 变为高;

(3) 若 ‘bit_in’ 输入为 0, 则 ‘en’ 变为低, ‘flag2’ 为 0, ‘dec1’、‘dec2’ 和 ‘dec3’ 分别为高电平, ‘data_in’ 连续输出 0 直到 ‘rs1’ 为高, ‘cen’ 变为高, 暂存器中的数据再次装载进入 5 位计数器, 同时 ‘dec3’ 变为低, 直到 ‘rs2’ 为高, CSR 中的 $P_{X(K)}$ 又回到解码输出状态, ‘en’ 再次变为高;

(4) 若 ‘bit_in’ 输入为 1, 则 ‘en’ 继续为高, 接收下一位, 若为 0, 则情况基本同(3), 不同的是 ‘data_in’ 连续输出 1; 否则接收到 1, 则 ‘en’ 继续为高, 此时 ‘flag2’ 为 1, ‘dec1’、‘dec2’ 和 ‘dec3’ 分别为高, ‘bit_in’ 上的数据直接被移入扫描链, 直到 ‘rs1’ 为高, ‘cen’ 变为高, 暂存器中的数据再次被装载进入计数器, ‘dec3’ 变为低, 同时 ‘en’ 变为低, 直至 ‘rs2’ 为高, ‘en’ 再次有效, 继续接收下个码字进行解压;

(5) 直到 ‘rs3’ 为高, 则一个向量解压结束, 5 位计数器复位到初始的模 3 计数状态, 返回(1)继续解压下个向量。

下面给出此解压结构中有限状态机的状态转换图(图 5), 它包括 10 个状态, S_0 为初始状态, 其中 $S_0 \sim S_1$ 以及 S_1 的循环是读入 k 信息块到暂存器的过程; $S_1 \sim S_2$ 过程是暂存器中的数据装载进入 5 位计数器, $S_2 \sim S_3 \sim S_4$ 以及 S_3 和 S_4 的循环过程是读入 $P_{X(K)}$ 并且使其在 CSR 内循环到解码输出状态, 同时完成计数器的又一次装载; $S_4 \sim S_5$ 开始接收待解压的码字; $S_5 \sim S_6$ 以及 S_6 的循环是解压相容块过程; $S_5 \sim S_7 \sim S_8$ 以及 S_8 的循环是反向相容块的解压过程; $S_5 \sim S_7 \sim S_9$ 以及 S_9 的循环是不相关块的解压过程; S_6 、 S_8 、 S_9 三个状态循环结束后回到 S_4 , 再次将暂存器中的数据装载进入计数器, 同时等待 CSR 内的 $P_{X(K)}$ 回到解码输出状态, 接下来到达 S_5 状态, 接收下个码字继续解码, 直到 ‘rs3’ 信号有效, 回到 S_1 状态, 即一个向量解码结束, 接着解压下个向量。

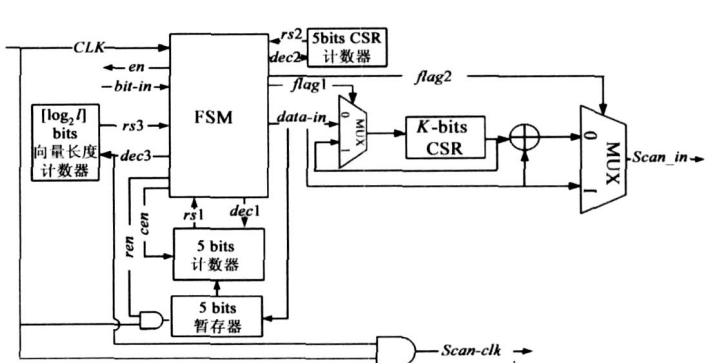
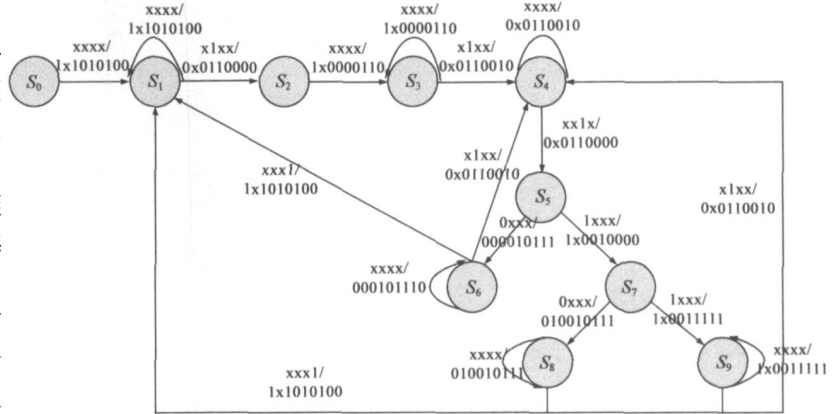


图 4 解压结构

状态转换输入、输出顺序: bit_in, rs1, rs2, rs3/en, data_in, ren, cen, flag1, flag2, dec1, dec2, dec3,



注: S_0 为初始状态

图 5 FSM 状态转换图

从状态转换图来看, 转换过程中的 $S_3 \sim S_4$ 以及 S_4 的循环过程就是当 $P_{X(K)}$ 不足 20 位的情况下, CSR 需要补充循环的次数, 目的使 CSR 中的数据循环回到解码状态, 保证下个码字的解码正确, 这使得解码过程中部分时间被浪费, 但是由于在具体的实验中大部分向量的 $P_{X(K)}$ 的长度较大, 故测试时间开销不会很大。对上述解压结构进行了对应的仿真与综合实验, 硬件仿真结果表明, 该解码器能够正确完成本文方案的解码功能。

4 实验结果

为了证明本文方案的有效性, 下面给出了本方案的实验结果, 实验电路是 ISCAS-89 标准电路中的最大的几个时序电路, 测试集是由 Mintest ATPG 测试生成工具生成的, 与 FDR 码^[6]、统计码^[9]、9C 编码^[11]以及 PRL^[7]等现有方法实验选取的测试集相同, 我们也将与这些方法实验结果(各方案的压缩率)作比较, 如表 1 所示。

从实验结果可见, 本文方案的最终压缩率均优于其他几种方法, 并且表 1 中统计码、9C 编码以及 PRL 方法的实验结果都选择了每个电路各自最优情况下的压缩率, 故本文的方案实现了对原测试集的有效压缩, 同时在本方案中各向量生成 $P_{X(K)}$ 的长度范围上限为 20, 故解压结构中只需要 20 位的 CSR 存放 $P_{X(K)}$ 即可, 硬件

开销较小. 本文对硬件解
码电路运用 Synopsys 公
司的 Design Compiler 综合工
具进行综合, 得出其晶体
管单元个数和电路面积
(μm^2) 两部分开销, 如表 1
最后两列所示.

5 结束语

本文提出了一种新
颖的变长数据块相关性
统计方法, 结合了统计和数据块相关的思想, 充分发挥
二者的优点, 将统计的思想应用到每个测试向量中, 并
且各向量的 $P_{X(K)}$ 长度完全独立, 这样使得统计的结果
明显高于文献[8, 9]中的在整个测试集中对定长数据块
的统计. 同时此方法充分利用了向量中非相邻数据块
的相关性, 相对于文献[7]中的方法, 选择出了一个更
优的数据块作为参考数据块, 避免了频繁更换参考数
据块的现象, 此方案有效地压缩了测试集并且减少了
测试应用时间. 它的解压结构也很简单, 相对于很多现
有的方法, 其硬件开销较小, 同时与大部分现有的方法
相比, 本文方案具有更好的测试数据压缩率, 并且拥有
较好的整体测试性能, 是一种有效的 SoC 测试数据压缩
/ 解压方法.

参考文献:

- [1] 梁华国, 聚贝勒, 海伦布昂特, 汉斯 耶西姆, 冯特利西. 一种基于折叠计数器重新播种的确定自测试方案[J]. 计算机研究与发展, 2001, 38(8): 931- 938.
Liang Hua Guo, Hellebrand S, Wunderlich H-J. A deterministic BIST scheme based on reseeding of folding counters[J]. Journal of Computer Research and Development, 2001, 38(8): 931- 938. (in Chinese)
- [2] 梁华国, 蒋翠云. 使用双重种子压缩的混合模式自测试[J]. 计算机研究与发展, 2004, 41(1): 214- 220.
Liang Hua guo, Jiang Cui yun. Mixed mode BIST using Bi seed compression[J]. Journal of Computer Research and Development, 2004, 41(1): 214- 220. (in Chinese)
- [3] A Chandraand, K Chakrabarty. Test resource partitioning for system on a chip based on test data compression and on chip decompression[J]. IEEE Design and Test of Computers, 2001, 18(5): 80- 91.
- [4] 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): 549- 554.
Liang Hua guo, Jiang Cui yun. Efficient test data compression and decompression based on alternation and run length codes [J]. Chinese Journal of Computers, 2004, 27 (4): 548- 553. (in Chinese)

表 1 实验结果

电路名称	原测试集 T_D	FDR 码	统计码	9C 编码	PRL	变长数据块相关性统计法	本方案硬件开销	
							number of cell	total cell area(μm^2)
S5378	23754	48.02%	55.1%	51.64%	54.97%	58.23%	232	9258.41
S9234	39273	43.59%	54.2%	50.91%	54.03%	55.76%	232	9258.41
S13207	165200	81.30%	77.0%	82.31%	83.11%	84.36%	236	9470.59
S15850	76986	66.22%	66.0%	66.38%	66.54%	70.38%	236	9470.59
S38417	164736	43.26%	59.0%	60.63%	53.25%	60.81%	244	9892.28
S38584	199104	60.91%	64.1%	65.53%	66.89%	69.13%	244	9892.28

第一列是电路名称, 第二列是原测试集 T_D 的大小, 第三、四、五、六列分别是 FDR 码^[6]、统计码^[9]、9C 编码^[11]以及 PRL^[7]方法对相同电路的 Mintest 集实验的压缩率, 之后则是本文方案所得到的压缩率.

- [5] 韩银和, 李晓维, 徐勇军, 李华伟. 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.
Han Yir he, Li Xiaor wei, Xu Yong jun, Li Hua wei. Test resource partitioning using variable tail code[J]. Acta Elelectronica Sinica, 2004, 32 (8): 1346- 1350. (in Chinese)
- [6] A Chandra, K Chakrabarty. 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] Xiaoyu Ruan, Rajendra Katti. An efficient data independent technique for compressing test vectors in Systems on a Chip [A]. IEEE Proceedings of the 2006 Emerging VLSI Technologies and Architectures, ISVLSI2006 [C]. Washington, DC: IEEE Computer Society, 2006. 153- 158.
- [8] A Chandra, K Chakrabarty. System on a Chip Test data compression and decompression architectures based on Golomb Codes[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2001, 20(3): 355- 368.
- [9] A Jas, J G Dasidar, N A Touba. Scan vector compression/ decompression using statistical coding [A]. Proceeding of 17th IEEE VLSI Test Symposium [C]. Dana Point, California, 1999. 114- 121.
- [10] A Jas, J G. Dasidar, N A Touba. An efficient test vector compression scheme using selective huffman coding [J]. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, 2003, 22(6): 797- 806.
- [11] Mohammad Tehranipoor, Mehrdad Nourani, K Chakrabarty. Nine coded compression technique for testing embedded cores in SoCs [J]. IEEE Transactions on VLSI systems, 2006, 13 (6): 719- 731.

作者简介:

欧阳一鸣 男, 1963 年生于安徽合肥, 副教授. 主要研究方向为嵌入式系统综合与测试、人工智能与模式识别等. E-mail: oyymbox@163.com

成丽丽 女, 1984 年生于安徽巢湖, 硕士研究生. 主要研究方向为嵌入式系统综合与测试.

梁华国 男, 教授, 1959 年生于安徽合肥, 博士生导师, 中国计算机学会容错计算专业委员会委员. 主要研究方向为嵌入式系统综合与测试, 数字系统设计自动化.