基于卷积编码的 SOC 测试响应压缩研究*

韩银和** 李华伟 李晓维

(中国科学院计算技术研究所先进测试技术实验室, 北京 100080; 中国科学院研究生院, 北京 100039)

Anshuman Chandra

(Synopsys, Inc, 700 E Middlefield Rd, Mountain View, CA 94043, USA)

摘要 提出一种单输出压缩方法. 首先提出了码率为 $n/(n-1)$ 、距离为 3 的卷积码的设计规则, 利用这些规则可得到卷积码的校验矩阵, 该校验矩阵的实现电路即是能够提供单输出压缩的响应压缩电路. 所设计的压缩电路可避免 2 个和任意奇数个错误位的混淆、避免一个未知位(X 位)对特征的掩盖. 利用概率论分析了未知位掩盖效应. 如果未知位分布具有聚簇特征, 那么提出的多重量校验矩阵设计算法能够大大降低未知位的掩盖效应. 最后用一些实验数据验证了所提出的压缩电路能够提供较强的未知位容忍能力和非常低的错误位混淆率.

关键词 SOC 测试 测试响应压缩 卷积码 错误位混淆率 未知位掩盖

测试在芯片整个周期(包含设计、制造、服务、其他系统)成本中所占的比重越来越大, 而且这种趋势还在加速发展^[1,2]. 测试压缩被认为是降低芯片测试成本的一个重要方法. 测试压缩一般包含两个方面: 测试激励压缩和测试响应压缩, 其对应的硬件结构如图 1 所示. 在图 1 结构中, 响应压缩电路通过数据压缩, 可将多扫描链的数据通过单个输出引脚传递到测试节点(可以是自动测试设备), 这就大大减少了存储向量的体积和传输时间.

响应压缩电路是当前研究比较多的一个领域^[2~16]. 这些研究可分为两大类: “和待测电路相关的压缩技术”^[3~5]和“和待测电路无关的压缩技术”^[6~16]. 第二类技术主要采用编码技术. Saluja^[9]首先构造了一个基于线性码校验矩阵的编码压

收稿日期: 2005-01-27; 接受日期: 2005-09-29

* 国家重点基础研究发展计划(2005CB321604 和 2005CB321605)、国家自然科学基金(批准号: 90207002, 60576031)和北京市重点科技项目(H020120120130)资助

** E-mail: yinhes@ict.ac.cn

图 1 内嵌式响应压缩电路设计示例

缩电路. Mitra^[10]提出了一种基于Hamming码校验矩阵的X-Compact压缩电路, 该电路能够诊断 1 个错误位, 文章^[10]还拓展了设计规则使得压缩电路能够检测出任意奇数个错误. Lumetta^[11,12]改造线性分组码, 提高了基于这些编码压缩电路对未知位的容忍能力. 上述压缩电路^[10,12], 均采用异或门实现, 文献[13]提出了一种新的编码压缩电路架构, 该压缩电路利用时序电路来实现线性码校验矩阵.

测试响应数据中会出现很多未知位^[10-16]. 文献[15,16]对一般数字电路带错误位的响应数据进行分析后提出了“少错误位的压缩模型”. 这种模型认为对于大多数测试向量而言, 出现的错误位比较少(一般小于 4 个). 本文提出的响应压缩电路也是基于少错误位压缩模型, 具有低混淆率和较强未知位处理能力.

1 改进的 $(n, n-1, m, d)$ 卷积码及对应的响应压缩电路设计

Elias^[17]在 1955 年第一次提出了卷积码. 由于卷积码具有很多优点, 使得它被广泛应用于通信系统、无线、卫星系统、移动通信等领域^[18].

卷积码可以用一个四元组表征: (n, k, m, d) , 其中 n 是并行输出的数目, k 是并行输入的数目, m 是编码约束长度(在硬件电路实现中对应于存储单元的数目 memory size), d 是卷积码码字的最小距离. 卷积码上的最小距离定义为卷积码对应的所有码字之间距离的最小值:

$$d = \min\{d(y_1, y_2) | y_1 \neq y_2, y_1, y_2 \text{ 是码字}\} = \min\{w(y) | y \neq 0, y \text{ 是码字}\}.$$

这里, $d(\cdot, \cdot)$ 表示两个码字之间的距离, $w(\cdot)$ 表示码字的重量, 码字重量指码字中所含有 1 的数目. 如果 G_∞ 是卷积码的生成矩阵, 那么卷积码的校验矩阵 H_∞ 定义为 $G_\infty \cdot H_\infty^T = 0$. 如果使用 m -截断矩阵来表征卷积码, 那么相应的 m -截断校验矩阵定义为 $G_m \cdot H_m^T = 0$, 其中, H_m^T 是一个 $(m \times (n-k))$, $m \times n$ 矩阵. 在下文中,

如果我们不特指,提及的校验矩阵均是指 m -截断校验矩阵. 在 H_m 中, 每一个矩阵元素 h_i 是一个 $(n-k, n)$ 子矩阵, 其他列可以看作是第 1 列矩阵元素下移的结果, 如果矩阵 H_m 中第 1 列矩阵元素确定, 则整个校验矩阵 H_m 也就确定. H_m 的前 n 列可以表示为: $H = \{h_1^T, h_2^T, h_3^T, \dots, h_m^T\}^T$, 称为校验矩阵的基本阵.

文献[9]提出下面一个压缩电路设计引理:

引理 1^[9] 如果 H 是线性码 $(n, n-k, d)$ 的校验矩阵, 校验矩阵的实现电路可以作为测试响应压缩电路(记为 H -based 压缩电路). 且当测试响应距离小于 d 时, 该压缩电路不可能产生相同的码字. 该压缩电路有 n 个输入和 k 个输出. 压缩率为 n/k .

而卷积码 (n, k, m, d) 可以看作一个生成矩阵为 G_m , 校验矩阵为 H_m 的线性码¹⁾. 利用卷积码的校验矩阵就可以设计其所对应的响应压缩电路. 下面用一个例子, 说明如何由卷积码校验矩阵生成响应压缩电路. 图 2 是一个卷积码校验矩阵的基本阵:

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

图 2 校验矩阵的基本阵 H

图 3 给出了基于图 2 描述的校验矩阵 H_m 的一种压缩电路实现. 它由多个异或门和寄存器组成. 从结构上, 它可以分成两个部分: 异或网络和寄存器链. 异或网络的实现是基于校验矩阵的, 它可以使用下列三个映射规则从校验矩阵的基本阵中综合得出:

- (1) 校验矩阵的基本阵 H 每一列对应于异或网络的一个输入;
- (2) 校验矩阵的基本阵 H 每一行对应于异或网络的一个输出;

(3) 校验矩阵的基本阵 H 中每一行非零元素的分布决定了异或网络的结构. 如果 H 中第 i 行, 第 j 列元素为 1, 则第 j 个异或网络的输入通过异或门连接到第 i 个异或网络的输出上, 其中, 异或网络的第 1 个输出最远离寄存器链的输出端.

根据上面三个映射规则, 可得到图 2 中的校验矩阵基本阵所描述的异或网络的计算式:

$$\text{Out}_1 = \text{In}_1 \oplus \text{In}_2 \oplus \text{In}_3 \oplus \text{In}_5 \oplus \text{In}_6$$

1) Luerssen H G, Schmale W. Distance bounds for convolutional codes and some optimal codes. In: the Report available at <http://www.uni-oldenburg.de/math/personen/gluesing/OptCodes.pdf>

$$\text{Out}_2=\text{In}_1 \oplus \text{In}_4 \oplus \text{In}_5$$
$$\text{Out}_3=\text{In}_2 \oplus \text{In}_3 \oplus \text{In}_4$$
$$\text{Out}_4=\text{In}_1 \oplus \text{In}_2 \oplus \text{In}_3 \oplus \text{In}_6$$
$$\text{Out}_5=\text{In}_2 \oplus \text{In}_5 \oplus \text{In}_6$$

根据这些计算式得到图 2 中描述的校验矩阵对应的压缩电路:

图 3 基于卷积编码的响应压缩电路实现

2 卷积码响应压缩电路设计规则

文献[9~11]还给出了其他一些设计规则, 为了方便后续内容的引述, 我们将这些设计规则作为引理, 描述如下:

引理 2^[10,11] 对于一个线性码($n, n-k, d$), $H(k \times n)$ 是它的校验矩阵, 那么H-based压缩电路具有的检测错误位能力, 处理未知位能力二者有下列关系:

- (1) 它可以检测出测试响应数据中 $d-1$ 个错误位;

(2) 如果校验矩阵每一列的重量均为奇数, 那么它可以检测出任何奇数个错误位;

(3) 它可以检测出伴随着 u 个未知位的 e 个错误位, e 和 u 满足下列关系:
 $e+u \leq d-1$;

先前的实验调查已经说明当码字距离 d 为 3 的时候, 对于一般响应压缩电路已经足够. 本文下面的内容将着重于这种距离 d 为 3 的卷积码的设计. 首先, 介绍两个预备概念:

定义 1 对于两个 m 次多项式 $R_1(x)$ 和 $R_2(x)$, 则等价关系: $R_1(x) \equiv R_2(x)$ 当且仅当:

$$(I) \exists \sigma \mid R_1(x) = x^\sigma \times R_2(x), \quad 0 \leq \sigma \leq m-1 - \deg(R_2(x)),$$

或者

$$(II) \exists \sigma \mid R_2(x) = x^\sigma \times R_1(x), \quad 0 \leq \sigma \leq m-1 - \deg(R_1(x)).$$

这里 $\deg(\cdot)$ 是多项式中最大非零系数对应的次数.

定义 2 等价类和特征多项式; 我们定义一个“等价类”为所有互为等价的 m 次多项式的集合. 定义“特征多项式”为在一个等价类中, 最大非零次数为 $m-1$ 的多项式. 可以看出, 一个等价类中有且只有一个特征多项式. 正因为这种唯一性, 我们可以用特征多项式来表征整个等价类.

有了等价类和特征多项式的概念, 我们可以得出下面一个设计定理:

定理 1 对于卷积码 $(n, n-1, m, d)$, 在校验矩阵的基本阵中, 如果没有两列是等价的, 那么该编码的码字最小距离 d 有 $d \geq 3$.

证 在 $(n, n-1, m, d)$ 卷积码中, 校验矩阵的基本阵是 n 列多项式的集合 $H = \{C_1(x), C_2(x), C_3(x), \dots, C_n(x)\}$. 其中, $C_i(x)$ 是一个 $m-1$ 次的多项式. 这样, 该卷积码的校验矩阵可表示为: $H_m = \{C_1(x), C_2(x), C_3(x), \dots, C_n(x), xC_1(x), xC_2(x), \dots, xC_n(x), x^2C_1(x), x^2C_2(x), \dots, x^2C_n(x), \dots, x^{m-1}C_1(x), x^{m-1}C_2(x), \dots, x^{m-1}C_n(x)\}$. 其中, $x^i C_p(x)$ 是最大次数为 $m-1$ 的多项式, 如果乘积的结果多项式次数大于 $m-1$, 则大于的部分将被截去, 保持最大非零系数次数小于等于 $m-1$. 从等价的概念可以看出, 如果在 $\{C_1(x), C_2(x), C_3(x), \dots, C_n(x)\}$ 中没有两个多项式等价, 那么任意两个 $x^i C_p(x)$ 和 $x^j C_q(x)$ 不会相同, 这就是说在 H_m 中任意两列都线性无关. 因为卷积码也是线性码, 根据线性码距离计算公式¹⁾, 得到所设计的卷积码的最小距离 d 有 $d \geq 3$.

推论 1 如果校验矩阵 H_m 按照定理 1 构造, 那么约束长度 m 存在下面的限制:

$$m \geq \log_2(n) + 1.$$

1) 见第 688 页脚注

证 非等价列多项式的重量可以从 1 变化到 m . 对于重量为 ω , 最高非零项次数为 $m-1$ 的特征多项式, 除去最高次数 $m-1$, 其他还应该有 $\omega-1$ 个非零系数次数待定. 这些待定的非零系数次数可以看作是从 $\{0, 1, \dots, m-2\}$ 集合中取 $\omega-1$ 个元素进行组合, 这样组合的数目为: $C_{m-1}^{\omega-1}$. 根据这个结论, 所有次数为 $m-1$ 的多项式中所含的等价类数目为

$$n \leq 1 + C_{m-1}^1 + C_{m-1}^2 + C_{m-1}^3 + C_{m-1}^4 + \dots + C_{m-1}^{m-1} = 2^{m-1},$$

$$\therefore m \geq \log_2(n) + 1.$$

定理 2 对于卷积码($n, n-1, m, d$), 如果校验矩阵的基本阵 H 按照定理 1 设计, 且基本阵 H 中每一列多项式的重量均为奇数, 那么根据该卷积码的 H_m -based 压缩电路可以检测出两个错误位和任意奇数个错误位.

证 可以从引理 2 的第(1)和第(2)个结论得知.

定理 3 如果卷积码校验矩阵 H_m 设计满足定理 1, 且 H_m 每一列的重量均相等, 那么基于该校验矩阵 H_m 实现的 H_m -based 压缩电路可以检测出伴随着一个未知位的单错误位.

证 该定理可以从引理 2 的第(3)结论得出.

尽管定理 3 可使得设计的编码压缩电路具有一定的处理未知位的能力, 然而在实际应用中, 这些能力是不足的, 需要加强. 文章第 4 节将讨论有关未知位容忍能力.

3 未知位容忍能力分析

在芯片可测性设计后, 响应输出数据中可能有很多未知位^[10-16], 定理 3 给出了基于卷积码校验矩阵的压缩电路处理未知位的基本能力. 然而, 在实际中, 这种能力是不足的. 需要设计新的容忍未知位的能力. 容忍未知位意味着需要从提高错误位的被检测出概率角度去考虑未知位的处理能力. “检测概率” P 定义为伴随着未知位时, 压缩电路通过输出检测出错误位的概率^[11].

可以用概率论的知识来分析错误位的“检测概率”. 如果一个卷积码($n, n-1, m, d$)的校验矩阵是 H_m , 该校验矩阵是 $(m, m \times n)$ 矩阵. 假设校验矩阵中所有列多项式的重量均为 ω , 这样如果假设 H_m 中每一个位置取值为 1 的概率相等且等于 P_1 , 那么 $P_1 = \omega/m$. 如果输入信息字为 I , 输出码字为 C , 那么压缩过程可以表示为: $C = H_m \times I^T$. 其中 I 是一个 $m \times n$ -位字、 C 是一个 m -位的压缩码字. 不失一般性, 假设单错误位 E 仅可能分布在 I 的前 n 位(第一拍输入字). 设从扫描链卸载出的响应数据中未知位的密度为 P_x , 假设未知位是等概率分布的, 那么 P_x 也是响应数据中未知位出现的概率. 使用 $\bar{P}_j$ 表示通过比较输出压缩码字的第 j 位不能检测出错误位 E 的概率.

压缩字第 j 位可以用下式计算得到:

$$C_j = [h_j, \dots, h_1, 0, \dots, 0] \times I^T,$$

$[h_j, \dots, h_1, 0, \dots, 0]$ 是 H_m 中的第 j 行向量. 这样, 只有未知位分布在前 $j \times n - 1$ 位时才可能掩盖错误位(因为其他 $(m-j) \times n$ 均为零). 于是

$$\bar{P}_j = P_1 \times \{1 - [P_X \times (1 - P_1) + (1 - P_X)]^{j \times n - 1}\} + (1 - P_1),$$

化简该式得到

$$\bar{P}_j = 1 - P_1 \times (1 - P_1 \times P_X)^{j \times n - 1}.$$

单个错误位完全被掩盖的概率为

$$\bar{P} = \prod_{j=1}^m [1 - P_1 \times (1 - P_1 \times P_X)^{j \times n - 1}].$$

通过检查整个 m 位输出压缩字, 可得到单个错误位的检测概率为

$$P = 1 - \left\{ \prod_{j=1}^m [1 - P_1 \times (1 - P_1 \times P_X)^{j \times n - 1}] \right\}.$$

如果需要计算伴随着 u 个未知位的单错误位被检测的概率, 那么代入: $P_1 = \frac{\omega}{m}$ 和

$$P_X = \frac{u}{mn} \text{ 得到}$$

$$P = 1 - \left\{ \prod_{j=1}^m \left[1 - \left(\frac{\omega}{m} \right) \times \left(1 - \frac{\omega}{m} \times \frac{u}{mn} \right)^{j \times n - 1} \right] \right\}.$$

我们没有办法进一步对该式进行化简. 观察该式, 可以看出单个错误的检测概率和参数 P_1 和 P_X 紧密相关. 而 P_1 和校验矩阵中列向量的重量 ω 及压缩电路的存储单元个数 m (编码的约束长度) 有关. 下面将通过分析这些参数之间的关系, 来确定如何选择适当的 ω 和 m , 从而得到尽可能大的错误位检测概率.

在图 4 中的前两个例子中, 检测概率都是随着列多项式重量的增加而增加的. 然而, 在第三个例子中, 只有当重量 $\omega = 9$ 时, 检测概率才能取得最大值(最大值为: 71.5%). 而且 ω 取值对检测概率的影响效果很显著. 对于图 4 中的三个例子, 可以发现当重量 $\omega = [7 \sim 11]$ 时, 单个错误位的检测概率比较大. 图 4 中的曲线还描述了检测概率和未知位密度的关系, 它指导我们什么时候需要选择使用一些可测性设计逻辑去减少未知位. 可以看出, 当未知位的密度控制在 0.15% (对应于图中 $u=10$ 的曲线), 只要我们合理选择重量和存储单元个数, 就可以将未知位的掩盖概率降低到 0.01% 以下. 但是, 如果未知位的密度高达 0.78% 时 (对应于图 4 中 $u=50$ 曲线), 无论选择何种重量, 最小不确定掩盖概率也会达到 29%. 在这种情况下, 就需要增加额外的可测性设计逻辑去减少不确定的数目.

图 4 检测概率(P)和重量(weight)分布的关系

图 5 分析检测概率和存储单元数之间的关系. 可以看出随着存储单元数的增加, 检测概率也在增加. 这样, 选择较大的存储单元个数可以帮助我们提高单个错误位的检测概率. 当然增加存储单元数时要考虑压缩电路的面积开销, 其间的折衷是设计压缩电路时要注意考虑的一个问题.

图 5 检测概率(P)随存储单元数(memory size)的变化曲线

4 多重量校验矩阵设计算法

在实际芯片中未知位往往是聚簇分布的^[15,16], 也就是说大量的未知位只会分布在少数几个扫描输出上. 因此, 可以针对这种实际中的聚簇效应提出解决办法来更加有效地解决未知位掩盖的问题. 分析可以看出: 未知位的掩盖概率和未

知位在卷积码 H_m -based压缩电路的存储单元中时间有关,而未知位在存储单元中的时间又和校验矩阵的列多项式有关.定义一个参数“势” τ 来刻画校验矩阵的基本阵中某一列中1的分布, τ 可由下式计算得到:

$$\tau_i = \sum_{l=0}^{l=m-1} (l) (C_i^l = 1),$$

C_i^l 表示校验矩阵的基本阵中第*i*列第*l*个元素. τ_i 可以理解为校验矩阵的基本阵中第*i*列多项式中所有非零系数的对应的次数之和. τ_i 值越小,则未知位扩散到的存储单元和输出越接近,它们对其他错误位的掩盖可能性就越小.

基于上面对势和未知位掩盖率关系的认识,为了尽可能地降低未知位对特征的影响,我们提出了下列算法:

多重量校验矩阵设计算法.重量数为 w

- (1) 首先对扫描链的输出根据其输出数据中的未知位密度进行排序;
- (2) 根据未知位密度阈值集合 $\{X_1, X_2, \cdots, X_w\}$,将扫描链划分成 w 个集合 $\{S_1, S_2, \cdots, S_w\}$, S_1 中的扫描链的未知位密度都大于 X_1 , S_2 中的扫描链的未知位密度都大于 $X_2, \cdots$,直到所有扫描链划分完毕;
- (3) 取重量 ω 的集合为: $\{\omega_1, \omega_2, \cdots, \omega_w\}$, 其中 $\omega_1 < \omega_2 < \cdots < \omega_w$;
- (4) i 取值从 1 到 w :
- a) 根据 τ 值,从小到大生成校验矩阵基本阵列向量,这些列向量的重量为 ω_i ;
- b) 将这些列向量对应到 S_i 中的扫描链.

表 1 应用多重量校验矩阵设计算法后未知位掩盖概率的减少

实例	编码压缩电路参数			X 位密度(X 位数/ 总响应数)	X 位分布	随机校验矩阵 设计算法	多重量校验矩阵 设计算法
	n	m	d				
实例 1	100	52	5	0.003%	$I_1(90\%), \text{other}(10\%)$	421	32
实例 2	100	52	5	0.003%	$I_1(30\%), I_2(30\%), I_3(10\%),$ $I_4(15\%), \text{other}(15\%)$	3274	543
实例 3	100	64	5	0.03%	$I_1(90\%), \text{other}(10\%)$	2104	152
实例 4	100	64	5	0.03%	$I_1(30\%), I_2(30\%), I_3(10\%),$ $I_4(15\%), \text{other}(15\%)$	13431	1762
实例 5	100	48	5	0.3%	$I_1(90\%), \text{other}(10\%)$	63173	1035
实例 6	100	48	5	0.3%	$I_1(30\%), I_2(30\%), I_3(10\%),$ $I_4(15\%), \text{other}(15\%)$	425340	5120

表 1 列出了利用该算法的一些实验结果.实验中采用了双重量算法, S_1 集合中列向量的重量选择 3, S_2 集合中列向量的重量选择为 9.实验实例 1,2 模拟了未知位密度比较小的芯片;实例 3,4 模拟了未知位密度相对于实例 1,2 稍高的芯片;实例 5,6 模拟了两款未知位密度比较高的芯片.每一种实例模拟 10^8 次.所有这

些实例的未知位均假设呈聚簇分布, 实例 1, 3, 5 中, 大部分未知位(90%)聚集在一个扫描输出上, 实例 2, 4, 6 中, 大部分未知位(85%)聚集在 5 个扫描输出上. 第 5 列列出了采用类似于文献[15, 16]中的随机校验矩阵设计算法综合得到压缩电路, 经过电路模拟后错误位被未知位掩盖数目, 而第 6 列列出了采用多重量校验矩阵设计后得到的压缩电路错误位被未知位掩盖的数目模拟结果. 可以看出, 对于实例 1, 3, 5 来说, 由于未知位集中于一个输出上, 本文提出的算法效果特别好, 被掩盖的错误位可以有两个数量级上的减少. 对于实例 2, 4, 6 效果虽不如 1, 3, 5 明显, 但也有较大程度地减少.

5 混淆率方面的一些实验结果

混淆是响应压缩中需要解决的一个问题. 混淆是指带错误位的响应数据经过压缩后和正确响应数据经过压缩后的压缩结果完全一致, 从而使得测试者将错误响应结果误解为正确. 在测试中混淆情况发生的次数越少越好. 卷积码 H_m -based 压缩电路的混淆情况来源于两个方面: (1) 异或网络压缩带来的混淆; (2) 不同拍错误位在寄存器链上的相互抵消而带来的混淆. 如上文所述, 卷积码可以看作是校验矩阵为 H_m 的线性码. 这样, 压缩电路压缩过程可以看作是从 $m \times n$ 位信息空间映射到 m 位压缩码字空间的过程. 如果假设错误位分布是等概率的, 那么压缩电路的混淆率为

$$P(\text{混淆}) = \frac{m \times n \text{位空间中非码字元素个数} - 1}{m \times n \text{位字空间元素个数}} = \frac{2^{n \times m - m} - 1}{2^{n \times m}} = \frac{2^{(n-1) \times m} - 1}{2^{n \times m}} \approx \left(\frac{2^{(n-1)}}{2^n} \right)^m = 2^{-m}.$$

观察表达式可以发现, 混淆率只和压缩电路中的存储单元个数有关系, 和压缩电路的压缩率是没有关系的. 这一点启示我们通过设计比较大的存储单元数来减少混淆率.

当存储单元个数比较小的情况下, 混淆率还和校验矩阵中列向量的重量 ω 有关. 表 2 列出了一些用来评估 ω 的影响的实验结果. 压缩电路按照定理 2 设计, 因为定理 2 不能保证 4 个错误位的混淆. 所以, 在每一次模拟时, 电路中插入 4 个

表 2 压缩电路混淆率分析

m	100 个输入 4 个错误位			200 个输入 4 个错误位		
	$\omega=3$	$\omega=5$	$\omega=7$	$\omega=3$	$\omega=5$	$\omega=7$
30	3722	542	98	1780	323	112
40	360	43	33	242	75	45
50	55	24	11	63	25	8
60	32	12	0	5	19	3
70	7	4	0	1	2	5
80	3	8	0	0	0	0
90	0	2	1	0	0	0
100	0	1	0	0	0	0

错误位, 以评估 4 错误位的混淆率. 针对每一个设计进行 10^8 次故障插入和模拟, 表 2 中列出了经过 10^8 次模拟后, 发生混淆的次数. 通过观察可以发现, 在存储单元个数比较小的情况下, 增加 ω 可以减少混淆率, 而且这种减少的效果比较显著. 当 m 取值在 70 以上时, 错误位的混淆率则低于 10^{-8} .

也可以使用HOPE^[19]通过电路模拟的方法来评估压缩电路的混淆率. 因为本文的方法是针对全扫描设计的, 首先将时序电路改造为全扫描电路, 使用组合电路测试向量生成工具MINTTEST^[20]来产生 100%故障覆盖率的测试向量. 在全扫描化的时序电路中内置压缩电路. 为了简化模拟过程, 在模拟过程中去除了基准电路中的寄存器, 并且只针对基准电路部分(不包含压缩电路)进行并行单故障模拟. 对于表 3 中的所有电路, 采用了满足定理 2 的 62×62 校验矩阵基本阵来生成压缩电路. 表 3 中可以发现对于ISCAS'89 几个大电路实例, 都是无混淆的, 这一点充分显示了所设计的压缩电路在混淆率方面的优越性能.

表 3 应用卷积码压缩电路后, 对于固定型单故障的混淆率评估

电路	故障数	向量数	检测出故障数	混淆率
S13207	9815	236	9815	0%
S15850	11727	126	11727	0%
S35932	39094	16	39094	0%
S38417	31180	99	31180	0%
S38584	36305	136	36305	0%

6 结语

本文提出了一种基于卷积码理论的测试响应压缩电路. 为了取得最大压缩率, 本文提出了单输出压缩方法. 如果文中提出的一些设计定理应用到压缩电路设计中, 那么所设计的压缩电路可以提供低混淆率, 并能提供处理未知位能力. 通过一个多重量校验矩阵设计算法, 可以大大提高压缩电路对未知位的容忍能力. 相对于其他压缩电路而言, 本文提出的压缩电路面积会稍微大一些. 然而, 在现代芯片的设计中, 特别是在当前超大规模 SOC 设计中, 对于面积的要求并不十分严格, 因此本文提出的方法特别适合应用于这类芯片.

参 考 文 献

1 Semiconductor Industry Association (SIA). International Technology Roadmap for Semiconductors (ITRS), 1999

2 Xiang D, Gu S, Sun J, et al. A cost-effective scan architecture for scan testing with non-scan test power and test application cost. In: Proc of Design Automation Conference. New Orleans: ACM Press, 2003. 744—747

3 Ivanov A, Tsuji B, Zorian Y. Programmable space compactors for BIST. IEEE Trans Comput, 1996, 45(12): 1393—1405^[DOI]

4 Chakrabarty K, Murray B T, Hays J P. Optimal zero-aliasing space compaction of test responses. IEEE

- Trans Comput, 1998, 47(11): 1171—1187[DOI]
- 5 Bhattacharya B B, Dmitriev A, Goessel M. Zero-aliasing space compaction of test responses using a single periodic output. IEEE Trans Comput, 2003, 52(12): 1646—1651[DOI]
 - 6 Wohl P, Waicukauskis J A, Williams T W. Design of compactors for signature-analyzers in built-in self-test. In: Proc of International Test Conference. Baltimore: IEEE Press, 2001. 54—63
 - 7 Barnhart C, Brunkhorst V, Distler F, et al. Extending OPMISR beyond 10x scan test efficiency. IEEE Des Test Comput, 2002, 19(5): 65—73[DOI]
 - 8 Wohl P, Huisman L. Analysis and design of optimal combinational compactors. In: Proc of VLSI Test Symposium. Napa Valley: IEEE Press, 2003. 101—106
 - 9 Saluja K K, Karpovsky M. Testing computer hardware through data compression in space and time. In: Proc of International Test Conference. Philadelphia: IEEE Press, 1983. 83—88
 - 10 Mitra S, Kim K S. X-compact: An efficient response compaction technique. IEEE Tran Comput-Aided Des Integr Circuits Syst, 2004, 23(3): 421—432[DOI]
 - 11 Patel J H, Lumetta S S, Reddy S M. Application of Saluja-Karpovsky compactors to test responses with many unknowns. In: Proc of VLSI Test Symposium. Napa Valley: IEEE Press, 2003. 107—112
 - 12 Lumetta S S, Mitra S. X-codes: Error control with unknowable inputs. In: Proc of International Symposium on Information Theory. Pacifico: IEEE Press, 2003. 102
 - 13 Wang C, Reddy S M, Pomeranz I, et al. On compacting test response data containing unknown values. In: Proc of International Conference on Computer-Aided Design. San Jose: IEEE Press, 2003. 855—862
 - 14 Pomeranz I, Kundu I S, Reddy S M. On output response compression in the presence of unknown output values. In: Proc of Design Automation Conference. New Orleans: IEEE Press, 2002. 255—258
 - 15 Rajski J, Tyszer J, Wang C, et al. Convolutional compaction of test responses. In: Proc of International Test Conference. Charlotte: IEEE Press, 2003. 745—754
 - 16 Han Y, Xu Y, Li H, et al. Test resource partitioning based on efficient responses compaction for test time and tester channels reduction. In: Proceedings of Asian Test Symposium. Xi'an: IEEE Press, 2003. 440—445
 - 17 Elias P. Coding for Noisy Channels. IRE International Convention Record, 1955. 37—46
 - 18 Forney G D. Convolutional codes I: Algebraic structure. IEEE Trans Inf Theory, 1970, 16: 720—738
 - 19 Lee H K, Ha D S. HOPE: An efficient parallel fault simulator for synchronous sequential circuits. In: Proc of Design Automation Conference. Anaheim: IEEE Press, 1992. 336—340
 - 20 Hamzaoglu I, Patel J H. Test set compaction algorithms for combinational circuits. In: Proc of IEEE International Conference on Computer-Aided Design. San Jose: IEEE Press, 1998. 283—289