

引用格式:王秀云,刘军,任福继. TSVs 串扰故障分组测试和诊断策略[J]. 微电子学与计算机,2020,37(2):30-36. [WANG X Y,LIU J, REN F J. TSVs crosstalk fault grouping test and diagnostic strategy [J]. Microelectronics &Computer,2020,37(2):30-36.]

TSVs 串扰故障分组测试和诊断策略

王秀云^{1,2},刘 军^{1,2},任福继^{2,3}

(1 合肥工业大学 计算机与信息学院,安徽 合肥 230009; 2 合肥工业大学 情感计算与先进智能机器安徽省
重点实验室,安徽 合肥 230009; 3 德岛大学 工学院 智能信息系,日本 德岛 7708502)

摘 要: TSVs 串扰故障的测试和诊断对提高集成电路成品率有重要影响。为了减少 TSVs 测试和诊断时间,并且减少测试电路的面积开销,提出在信号接收端重用扫描单元的测试架构对 TSVs 串扰故障进行分组测试和诊断的新方案。该方案首先使用提出的 TSVs 分组算法,根据 TSVs 之间串扰影响距离,应用邻接矩阵求极大独立集对 TSVs 进行快速分组,使得每组内的 TSVs 不会发生串扰故障,并且最大化同组中 TSVs 的数量。分组完成后,使用提出的测试架构对同组内的 TSVs 进行并行测试,并且根据 TSVs 的测试响应,可以进一步诊断故障 TSVs。实验结果表明,所提测试方案有效地减少了测试和诊断时间,并且减少了面积开销。

关键词: TSVs; 串扰; 集成电路; 成品率; 扫描链

中图分类号: TP 391

文献标识码: A

文章编号: 1000—7180(2020)02—0030—07

TSVs crosstalk fault grouping test and diagnostic strategy

WANG Xiu-yun^{1,2}, LIU Jun^{1,2}, REN Fu-ji^{2,3}

(1 School of Computer and Information, Hefei University of Technology, Hefei 230009, China;

2 Anhui Province Key Laboratory of Affective Computing and Advanced Intelligent Machine,

Hefei University of Technology, Hefei 230009, China; 3 Department of Information Science & Intelligent
Systems, Faculty of Engineering, The University of Tokushima, Tokushima 7708502, Japan)

Abstract: Testing and diagnosis of TSVs crosstalk faults has an important impact on improving the yield of integrated circuits. In order to reduce TSVs testing and diagnostic time, and reduce the area overhead of the test circuit, a new scheme for group testing and diagnosis of TSVs crosstalk faults using a test architecture of the unilateral scan chain is proposed. The scheme firstly divides the TSVs into several groups according to the crosstalk influence distance between TSVs by the algorithm for finding the largest independent set in an adjacency matrix, so that the TSVs in each group do not have crosstalk faults and maximize TSVs in the same group. After grouping, the TSVs in the same group are tested and diagnosed in parallel using the proposed test architecture, and the faulty TSVs can be further diagnosed according to the test response of the TSVs. The experimental results show that the proposed test scheme effectively reduces the test and diagnosis time, and reduces the area overhead.

Key words: TSVs; crosstalk; integrated circuits; yield; scan chain

1 引言

随着半导体技术的进步,芯片的特征尺寸不断

缩小,互连线延迟和功耗的增加使得传统的摩尔定律已经很难继续发展下去^[1]。为了解决以上问题,以 TSVs (Through Silicon Vias) 为核心的 3D 集成技

术应运而生. 该技术通过 TSVs 将多个裸片在垂直方向上进行互连^[2-3], 缩短了芯片之间的互连线长度, 降低了延迟与功耗, 实现了高密度集成, 被认为是突破摩尔定律的有效途径^[4-7].

然而, 高密度集成可能导致 TSVs 之间发生串扰故障, 引起电路产生逻辑错误和时延故障, 从而影响信号传输的可靠性^[8-11], 使集成电路成品率下降. 因此, 绑定后 TSVs 串扰故障测试和诊断对提高集成电路成品率非常重要. 然而, 高密度集成使得 TSVs 不仅受到相邻 TSVs 的串扰影响^[12-14], 距离较远的 TSVs 也会产生影响^[15-16], 增加了 TSVs 串扰测试和诊断的难度.

本文根据串扰影响距离提出了一种新的分组算法, 该算法使用邻接矩阵求极大独立集实现对 TSVs 分组. 本文同时提出了一种新的测试架构, 该测试架构仅在 TSVs 测试信号接收端重用扫描单元. 当 TSVs 存在串扰故障时, 使用扫描链移位输出 TSVs 测试响应, 以便诊断故障 TSVs.

2 TSVs 分组策略

一个 TSV 不仅受到相邻 TSVs 的串扰影响, 距离较远的 TSVs 也会产生串扰影响. 定义两个 TSVs 间会产生串扰影响的最小距离为单位 d , 最大距离为 kd ($k \in R^+$), 受串扰影响的 TSVs 为受害 (Victim) TSVs, 造成串扰影响的 TSVs 为攻击 (Aggressive) TSVs. 如图 1 所示, 在规则 TSVs 分布中, 假设 d 为两 TSVs 之间最小的距离, 当最大串扰影响距离为 $\sqrt{2}d$ 时, 一个受害 TSVs 会受到周围 8 个攻击 TSVs 的串扰影响. TSVs 串扰故障测试时, 每次只激活 1 个受害 TSV 和 8 个攻击 TSVs, 所需的测试时间长.

为了减少串扰故障测试时间, 根据串扰影响距离, 将不会发生串扰影响的 TSVs 划分到一组, 并且最大化同组内 TSVs 的数量, 使同组内的 TSVs 并行测试.

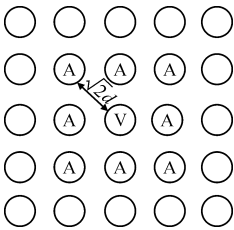


图 1 攻击 TSVs 和受害 TSVs 模型

本文提出使用邻接矩阵求极大独立集的方法对

TSVs 进行分组. 此方法首先根据串扰影响距离将 TSVs 之间的位置关系用邻接矩阵的形式表示. 若两个 TSVs 间不会发生串扰故障, 则矩阵的对应位置为“0”; 若两个 TSVs 间有可能发生串扰故障, 则矩阵的对应位置为“1”, 各 TSVs 自身与自身间的关系用“0”表示.

形成邻接矩阵后, 对此矩阵通过若干次平移变换转换为标准型, 得到极大独立集. 其算法的具体过程如图 2 所示:

输入: TSVs 分布信息, 最大串扰影响距离 kd
输出: 分组结果
步骤一: 根据 TSVs 分布信息和串扰影响距离 kd 构建邻接矩阵;

步骤二: 将邻接矩阵通过平移变换为标准型, 得到极大独立集, 将极大独立集中的 TSVs 划分为一组;

步骤三: 判断是否仍有 TSVs 未被划分到某分组中. 若有, 则执行步骤四; 若没有, 转向步骤五;

步骤四: 将已经分组的 TSVs 从邻接矩阵中删除, 得到新的邻接矩阵, 转向步骤二;

步骤五: 算法结束.

下面以图 3 为例, 阐述图 2 中分组算法在规则 TSVs 分布和不规则 TSVs 分布中的具体应用. 为方便描述, 图 3 中将各 TSVs 从 1 至 9 进行编号, 以便区别不同的 TSVs.

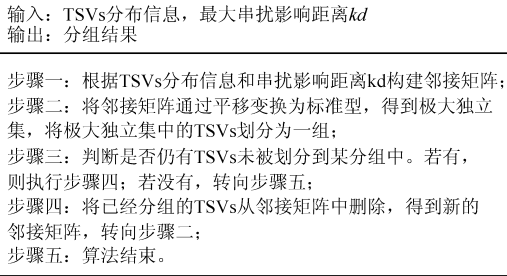
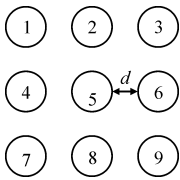
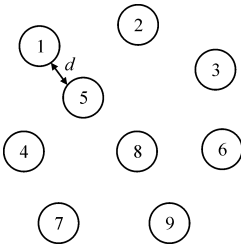


图 2 TSVs 分组算法

首先执行步骤一, 将各 TSVs 之间的位置关系转换为邻接矩阵. 在图 3(a) 中假设最大串扰影响距离为 $\sqrt{2}d$, 则转换后的邻接矩阵为矩阵(1), 矩阵的



(a) 规则 TSVs 分布



(b)不规则 TSVs 分布

图 3 TSVs 分布

行序号和列序号对应着图 3(a)中 TSVs 的编号.

$$\begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \end{matrix} \quad (1)$$

接着执行步骤二,对矩阵(1)进行平移变换. 首先将矩阵(1)中的 2 项和 6 项作平移变换,再将 3 项和 7 项作平移变换,得到矩阵(2).

$$\begin{matrix} & 1 & 6 & 7 & 4 & 5 & 2 & 3 & 8 & 9 \\ \begin{matrix} 1 \\ 6 \\ 7 \\ 4 \\ 5 \\ 2 \\ 3 \\ 8 \\ 9 \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix} \end{matrix} \quad (2)$$

矩阵(2)是标准型,从而可知集合 {TSV1, TSV6, TSV7} 是邻接矩阵的极大独立集. 因此,将 TSV1、TSV6 和 TSV7 划分为一组.

当执行步骤三时,仍有 TSV2、TSV3、TSV4、TSV5、TSV8 和 TSV9 没有分组,因此执行步骤四.

在执行步骤四时,将 TSV1、TSV6 以及 TSV7 从矩阵中删除,得到新的邻接矩阵. 对此新的邻接矩阵重复执行步骤二、步骤三和步骤四. 依次得到极大独立集 {TSV4, TSV3, TSV9}、{TSV2, TSV8} 和 {TSV5}. 因此,将 TSV4、TSV3 和 TSV9 分为一组,将 TSV2 和 TSV8 分为一组,将 TSV5 分为一组. 至此,算法结束,最后的分组结果为 {TSV1, TSV6, TSV7}、{TSV4, TSV3, TSV9}、{TSV2, TSV8}、{TSV5}.

在 3(b)中,TSVs 的分布是不规则的,任然可以使用图 2 中的分组算法对 TSVs 进行分组. 假设图 3(b)中的最大串扰影响距离仍为 $\sqrt{2}d$, 根据串扰影响距离将各 TSVs 之间的位置关系转换为邻接矩阵

(3).

$$\begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{matrix} \quad (3)$$

对矩阵(3)顺次重复执行图 2 分组算法中的步骤二、步骤三和步骤四,依次得到极大独立集 {TSV9, TSV4, TSV3}、{TSV7, TSV5, TSV6}、{TSV2, TSV8} 和 {TSV1}. 因此,TSV9、TSV4、TSV3 分为一组,TSV7、TSV5、TSV6 分为一组,TSV2、TSV8 分为一组,TSV1 单独作为一组.

当 TSVs 分组完成后,使用测试架构对 TSVs 串扰故障进行分组测试和诊断. 测试时,一组 TSVs 为受害 TSVs,则其他组 TSVs 都为攻击 TSVs. 诊断时,通过 TSVs 测试信号接收端的扫描链移位输出各 TSVs 的响应信号,诊断故障 TSVs. 接下来,第三章将详细阐述本文提出的测试架构.

3 串扰故障测试架构

TSVs 串扰故障测试时,当攻击 TSVs 传输信号出现信号转换时,会对受害 TSVs 产生串扰影响,因此需要通过测试架构施加连续的两个测试模式.

如图 4 所示,图中展示了本文提出的测试架构. 图中,为了减少测试和诊断时间,同组中的 TSVs 连接到同一个寄存器,以便进行测试模式的广播,因此,当施加两个连续的测试模式时只需 2 个时钟周期(cycles). 在图 4 中,当测试模式传递到寄存器之前,通过 MUXes(多路选择器)的使能信号 $CNTi(i=1,2,\dots,n)$ 确定输入到寄存器的测试模式是 PA 模式(攻击 TSVs 模式)或 PV 模式(受害 TSVs 模式). PA 模式和 PV 模式由 TPG(测试模式生成器)生成,并且某组寄存器输入为 PV 模式后其他组的寄存器均为 PA 模式. 当寄存器输出的 PA 模式或 PV 模式传递到 TSVs 时,通过 TSVs 前的 MUXes,使能 TEST 信号的值为 0,使 TSVs 信号的输入端接受测试模式输入,进行串扰故障的检测.

在测试架构中,当捕获测试响应时,将 TSVs 的输出信号与 Golden 信号(预期的输出信号)进行异或,得到 TSVs 串扰故障测试的响应信号,然后将同组中所有的响应信号相或. 若组中某个 TSVs 发生串扰故障时,经过或门后的寄存器的输出为 false,即该组中的 TSVs 存在串扰故障.

当 TSVs 存在串扰故障时,为了能够进一步诊断故障 TSVs,并且减少测试电路的面积开销,本文提出仅在 TSVs 测试信号的接收端重用扫描单元,定义这种只在一端使用扫描单元的结构称为单端扫描单元.当某组 TSVs 存在串扰故障时,通过扫描链将 TSVs 的响应信号移位输出,进行 TSVs 串扰故障诊断,以便确定故障 TSVs.

4 实验结果与分析

4.1 TSV 分组算法分析

本文使用 C++ 语言实现 TSVs 分组算法.如表 1 所示,表中列出了本文分组算法以及文献[9]和文献[10]分组算法的时间复杂度.

表 1 TSVs 分组算法的时间复杂度

work	本文	文献[9]	文献[10]
时间复杂度	$\frac{1}{2}n^2$	$\frac{1}{4}n^2$	n^2

从表 1 可以看出,本文分组算法的时间复杂度要优于文献[10]的时间复杂度,能够较快的实现分组.这是因为文献[10]分组算法需要将所有未被分组的 TSVs 遍历一遍,才能确定一个 TSVs 分组.然而,本文分组算法只需遍历转换后的矩阵的行或列,即每次遍历确定的 TSVs 数量多,因此,能够较快实现分组.从表中还可以看出,文献[9]分组算法的时间复杂度要优于本文,这是因为文献[9]分组算法使用四边形覆盖 TSVs 阵列,四边形的 4 个顶点每次确定同组中的 4 个 TSVs,因此,能较快的实现分组.然而,文献[9]分组算法只能用于规则 TSVs 分布.本文分组算法既可以用于规则 TSVs 分布也可以用于不规则 TSVs 分布,与文献[9]比较,更具有通用性.

4.2 测试架构硬件开销分析

本文测试架构中,各电路元器件使用 TSMC 90 nm 工艺.在 90 nm 工艺下,OR(或门)、XOR(异或门)、MUXes、D 触发器、TPG 的面积开销分别是 $3.528\ \mu\text{m}^2$ 、 $2.142\ \mu\text{m}^2$ 、 $5.64\ \mu\text{m}^2$ 、 $16.592\ \mu\text{m}^2$ 、 $196.86\ \mu\text{m}^2$.

为节约测试架构的面积开销,本文仅在 TSVs 测试信号的接收端使用封装在集成电路中的扫描链.在测试架构中,除了扫描链,上下裸片上分别需要一个 TPG,一个 TPG 生成 PA 模式和 PV 模式,另一个 TPG 生成 Golden 信号.在图 4 测试架构中,还可以看出每组 TSVs 需要两个由 D 触发器构成的寄存器,一个为每组 TSVs 存储 PA 模式或 PV

模式,另一个存储每组 TSVs 的测试结果.假设 n 表示 TSVs 的数量, g 表示 TSVs 组数,在测试架构中,除了扫描链、TPG 以及寄存器外,需要的 MUXes 的数量为 $n+g$ 、OR 的数量为 g 、XOR 的数量为 n ,其中 n 个 MUXes 用来选择 TSVs 信号接收端是功能输入或测试输入, g 个 MUXes 用来为每组 TSVs 选择 PA 模式或 PV 模式, n 个 XOR 用来将 TSVs 的输出信号与 Golden 信号进行异或, g 个 OR 用来将每组 TSVs 的响应信号相或.定义 A_{DFT} 表示本文方法封装在集成电路中测试架构的面积开销,则 $A_{\text{DFT}}=2\times\text{TPG}+(n+g)\times\text{MUXes}+g\times\text{OR}+2\times g\times\text{触发器}+n\times\text{XOR}=393.72+42.352g+7.782n$.

定义 A_{SC} 表示扫描链的面积, A_{TSV} 表示 TSVs 的面积, $[A_{\text{DFT}}/(A_{\text{TSV}}+A_{\text{SC}})]\times 100\%$ 表示本文测试架构相对面积开销的计算公式.在 90 nm 工艺下,一个扫描单元的面积是 $22.232\ \mu\text{m}^2$,一个 TSV 的面积是 $15\times 15\ \mu\text{m}^2$,因此,在集成电路中 A_{SC} 的值为 $n\times 22.232\ \mu\text{m}^2$, A_{TSV} 的值为 $n\times 15\times 15\ \mu\text{m}^2$.

假设本文方法、文献[9]方法和文献[10]方法对 TSVs 分得的组数均为 $g=4$,如图 5 所示,图中分别展示了本文、文献[8]、文献[9]以及文献[10]测试方法的相对面积开销.从图中可以看出,本文方法的相对面积开销从 4.04% 到 3.18%,文献[8]的相对面积开销从 13.5% 到 9.70%,文献[9]的相对面积从 5.93% 到 4.53%,文献[10]的相对面积从 17.93% 到 17.37%.从以上比较可知,文献[8]和文献[10]的面积开销要远远大于本文和文献[9],这是由于在文献[8]和文献[10]中没有使用扫描链结构,当捕获 TSVs 响应信号时需要增加额外的电路设备,导致文献[8]和文献[10]测试架构的面积开销大.从图 5 中也可以看出,本文方法的相对面积开销比文献[9]低,这是由于在文献[9]测试架构中在 TSVs 两端都使用了 BSCs,并且对 BSCs 进行修改,需要额外的电路设备,增加了面积开销.

接下来,本文分析了随着 TSVs 分组数的增大,提出的测试架构在面积开销上的变化.从图 4 的测试架构可以看出,每增加一组 TSVs,需要增加一个 MUX 和一个寄存器,其面积开销增大 $22.232\ \mu\text{m}^2$.当需要检测的 TSVs 的数量为 256,组数 $g=4$ 时,本文测试架构的面积开销为 $2\ 555.32\ \mu\text{m}^2$.当需要检测的 TSVs 的数量仍为 256,组数 $g=8$ 时,本文测试架构的面积开销为 $2\ 719.728\ \mu\text{m}^2$,其面积开销仅增大了 6.43%.因此,TSVs 组数的增大对测试

架构的面积开销影响很小.

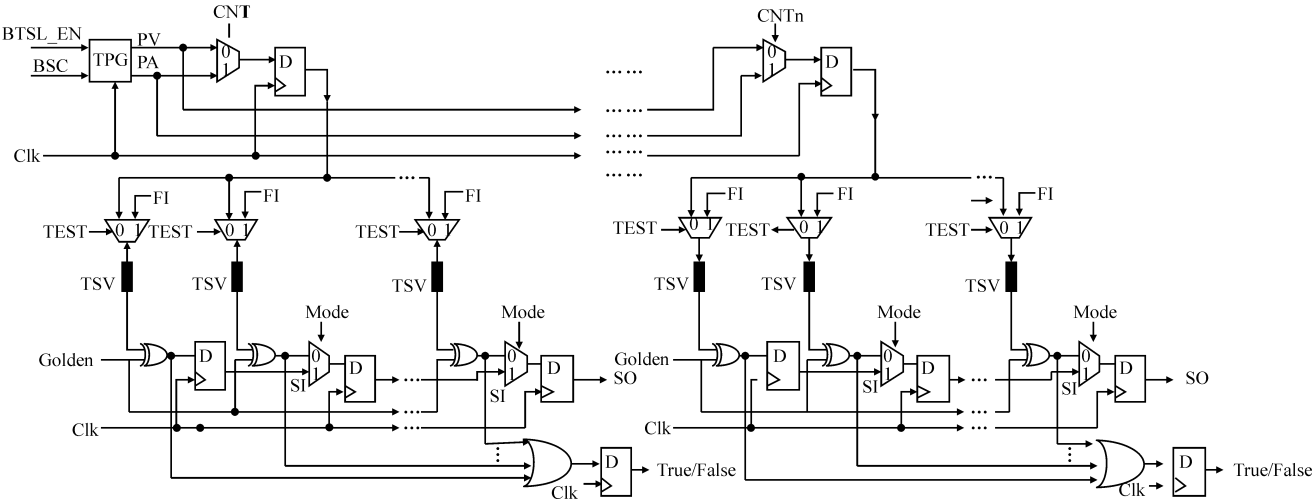


图 4 测试架构

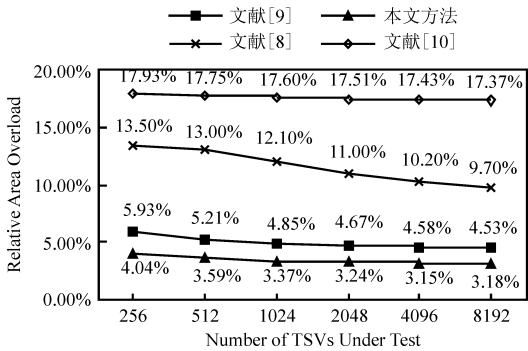


图 5 测试架构相对面积开销

4.3 TSVs 测试和诊断时间分析

常见的 TSVs 串扰故障有 4 种类型,分别是正尖峰故障、负尖峰故障、上升延时故障和下降延时故障. 本文提出的测试架构可以检测 TSVs 之间是否存在串扰故障以及存在哪种类型的串扰故障,此外还可以诊断故障 TSVs. 表 2 分别列出了本文方法、文献[8]方法、文献[9]方法以及文献[10]方法执行 TSVs 串扰故障检测过程的测试和诊断时间,在表 2 中的第二项为各文章中测试时间的计算公式,第三项为各文章中诊断时间的计算公式.

表 2 测试和诊断时间

work	Test Cycles	DiagnosisCycles
[8]	$20\sqrt{n}$	$20n$
[9]	$6g+n+1$	$12g+4n$
[10]	$6g+n+1$	$6g+n+1$
本文	$12g+4$	最好 $12g+4$ 最坏 $12g+n+4$

从表 2 可以看出,当检测集成电路中 TSVs 串

扰故障时,本文方法的测试和诊断时间要优于文献[8]方法、文献[9]方法以及文献[10]方法. 这是由于在文献[8]方法中,测试时间由 TSVs 的行数(C)和列数(R)决定. 测试时每次只能按行进行测试,且只有偶数列或奇数列的 TSVs 是受害 TSVs;诊断时,需要按列将测试响应移出. 此外,文献[8]将测试模式分为 2 个集合,每个集合有 5 个测试向量. 假设 $C=R=\sqrt{n}$, 得测试时间为 $2\times 2\times 5\times R=20\sqrt{n}$, 其诊断时间为 $20n$. 因此,文献[8]方法的测试和诊断时间长.

在文献[9]和文献[10]中,TSVs 的串扰故障按分组进行检测. TSVs 测试时,文献[9]和文献[10]的测试时间都是 $6g+n+1$. TSVs 诊断时,文献[9]可以确定 TSVs 产生串扰故障的类型,因此,诊断时间较长,诊断时间为 $12g+4n$,而文献[10]的诊断时间为 $6g+n+1$. 由于在文献[9]和文献[10]中,无论是否发生串扰故障,都需要将每个 TSVs 的响应信号移出,使得 TSVs 串扰故障检测的测试和诊断时间都与 TSVs 的数量成正比. 在集成电路中 TSVs 数量庞大,因此,文献[9]方法和文献[10]方法对 TSVs 串扰故障检测的测试和诊断时间长.

然而,在本文方法中,虽然与文献[9]方法和文献[10]方法相似,TSVs 串扰故障按分组进行检测,但与文献[9]方法和文献[10]方法不同的是,为减少 TSVs 串扰故障测试时间,本文将同组中的响应信号相或,测试结果只输出 false 或 true,即移出测试结果时只需 1 个时钟周期,不需要将每个 TSV 的响应信号移出,大大减少了 TSVs 串扰故障检测的测试时间. 由于广播两个连续的测试模式需要 2 个时

钟周期,加上捕获响应需要 1 个时钟周期,因此,测试一种类型的串扰故障需要 $3g+1$ 个时钟周期,测试 4 种类型的串扰故障需要 $4 \times (3g+1) = 12g+4$ 个时钟周期. 从本文方法测试时间的计算公式可知,本文方法的 TSVs 测试时间与 TSVs 的数量没有直接关系,因此测试时间短. 此外,TSVs 故障诊断时,若各组 TSVs 测试输出都为 true,则诊断时间与测试时间相同,得到最佳的诊断时间 $12g+4$,即没有故障 TSVs,因此不需要定位故障 TSVs;若各组 TSVs 测试输出都为 false,则需要将所有 TSVs 的响应信号移位输出,此过程需要 n 个时钟周期,因此得到最坏的诊断时间 $12g+n+4$.

为了更明显地展示本文方法在测试时间和诊断时间上的优势,本文将提出方法的测试时间和诊断时间与文献[9]方法作进一步比较. 假设本文方法和文献[9]方法分得的组数 $g=4$,如图 6 和图 7 所示,图 6 展示了本文方法和文献[9]方法在测试时间上的比较,图 7 展示了本文方法和文献[9]方法在诊断时间上的比较.

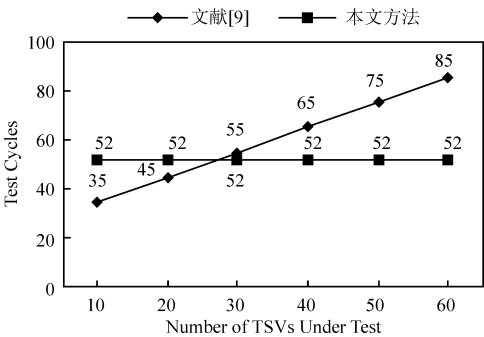


图 6 本文与文献[9]测试时间比较

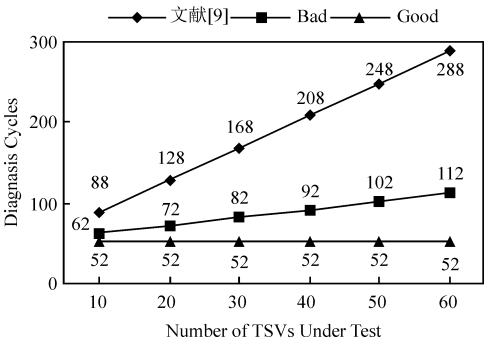


图 7 本文与文献[9]诊断时间比较

从图 6 可以看出,本文方法的测试时间与 TSVs 的数量无关. 当 TSVs 的数量大于 30 时,本文方法的测试时间总是比文献[9]的测试时间少,并且随着 TSVs 数量的增大,本文方法在测试时间上的优

势会越来越明显. 从图 7 可以看出,当 TSVs 之间没有串扰故障时,本文方法的最好(Good)诊断时间是常量,与 TSVs 的数量无关;当每组 TSVs 都存在故障 TSVs 时,本文方法的最坏(bad)诊断时间与 TSVs 的数量成正比. 从图 7 也可以看出,当 TSVs 的数量大于 10 时,本文方法的诊断时间总是比文献[9]方法的诊断时间少,并且随着 TSVs 数量的增加,本文方法在诊断时间上的优势会越来越明显.

4 结束语

本文提出了一种在信号接收端重用扫描单元的 TSVs 串扰故障分组测试和诊断的新方案,该方案首先将 TSVs 的位置信息转换为邻接矩阵,然后根据 TSVs 串扰影响距离,使用邻接矩阵求极大独立集对 TSVs 进行分组,使得每组内的 TSVs 不会发生串扰故障,并且最大化同组中 TSVs 的数量,然后使用提出的测试架构对 TSVs 进行测试. 实验结果表明,与现有 TSVs 串扰故障分组测试和诊断方案相比,本文方案能够降低 50% 的 TSVs 分组时间,并且适用于 TSVs 规则分布和不规则分布. 此外,提出的测试架构能够使 TSVs 串扰故障测试的面积开销最大降低 81.69%,测试时间最大减少 38.82%,诊断时间最大减少 81.94%.

参考文献:

[1] MOSSA S F, ELKEELANY O S A, HASAN S R. Grouped through silicon vias for lower Ldi/dt drop in three-dimensional integrated circuit[J]. Circuits Devices & Systems Iet, 2016, 10(1):44-53.

[2] 刘军,董鹏,任福继. 基于对角线的硅通孔容错设计[J]. 微电子学与计算机, 2018, 35(11):79-84.

[3] RAB M T, BAWA A A, TOUBA N A. Reducing Cost of Yield Enhancement in 3-D Stacked Memories Via Asymmetric Layer Repair Capability[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2014, 22(9):2017-2024.

[4] DING Y, CHEN Z, CHEN Q, et al. Highly conformal polyimide liner deposition in high-aspect-ratio through-silicon-vias [J]. Micro & Nano Letters, 2016, 11(5):253-255.

[5] LIU X, ZHU Z, YANG Y, et al. Low-Loss Air-Cavity Through-Silicon Vias (TSVs) for High Speed Three-Dimensional Integrated Circuits (3-D ICs)[J]. IEEE Microwave and Wireless Components Letters, 2016, 26(2):89-91.

- [6] YU C, CUI X, ZHOU S. A prebond TSV test scheme using oscillator[C]// IEEE International Conference on Solid-state & Integrated Circuit Technology. IEEE, 2017:987-989.
- [7] SHEN K, XIANG D, JIANG Z. Cost-Effective Test Optimized Scheme of TSV-Based 3D SoCs for Pre-Bond Test. [C]// IEEE Computer Society Annual Symposium on VLSI. IEEE, 2014:208-213.
- [8] HUANG Y J, LI J F, CHOU C W. Post-bond test techniques for TSVs with crosstalk faults in 3D ICs [C]// IEEE Int. Symp. VLSI Design Autom. Test (VLSI-DAT). Hsinchu, Taiwan, 2012:1-4.
- [9] HSU W H, KOCHTE M A, LEE K J. Built-In Test and Diagnosis for TSVs With Different Placement Topologies and Crosstalk Impact Ranges [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2017, 36(6):1004-1017.
- [10] PASCA V, ANGHEL L, BENABDENBI M. Kth-Aggressor Fault (KAF)-based Thru-Silicon-Via Interconnect Built-In Self-Test and Diagnosis[J]. Journal of Electronic Testing, 2012, 28(6):817-829.
- [11] 陈思远, 范鑫, 蒋剑飞, 等. 一种 TSV 阵列的串扰抑制设计方法[J]. 微电子学与计算机, 2018, 410(07): 25-28.
- [12] RAJSKI J, TYSZER J. Fault diagnosis of TSV-based interconnects in 3-D stacked designs[C]// IEEE International Test Conference (ITC). Anaheim, CA, USA, 2013:1-9.
- [13] YAO W, PAN S, ACHKIR B, et al. Modeling and application of multi-port TSV networks in 3-D IC[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2013, 32(4):487-496.
- [14] RAJSKI J, TYSZER J. Fault diagnosis of TSV-based interconnects in 3-D stacked designs[C]// 2013 IEEE International Test Conference (ITC). IEEE, 2013: 1-9.
- [15] TAOUIL M, MASADEH M, HAMDIOUI S, et al. Interconnect test for 3D stacked memory-on-logic [C]// Conference on Design. European Design and Automation Association, 2014:1-6.
- [16] HSU W H, KOCHTE M A, LEE K J. 3D-IC test architecture for TSVs with different impact ranges of crosstalk faults[C]// IEEE Int. Symp. VLSI Design Autom. Test (VLSI-DAT). Hsinchu, Taiwan, 2016: 1-4.

作者简介:

王秀云 女, (1992-), 硕士研究生, 研究方向为集成电路故障测试与修复等. E-mail: w1576322508@gmail.com

刘 军 男, (1978-), 副教授. 研究方向为数字系统设计自动化、集成电路测试、嵌入式系统等.

任福继 男, (1959-), 博士, 教授. 研究方向为人工智能、情感机器人、自然语言理解等.