

可寻址测试芯片测试结构自动分配算法研究^{*}

邵康鹏, 史 峥^{*}, 张培勇

(浙江大学 超大规模集成电路设计研究所, 浙江 杭州 310027)

摘要: 为了缓解由于可寻址测试芯片自动化设计方法的缺失所带来的设计效率低下、设计稳定性不足等问题, 针对可寻址测试芯片设计过程中的测试结构分配环节, 将线性规划应用到测试结构的自动分配算法中。通过整理现有的手动分配测试结构的方法, 得到了一套测试结构分配的基本规则; 同时, 将这些分配规则转换为多元一次不等式的数学表述, 从而构建了针对测试结构分配环节、基于线性规划的数学模型, 根据该数学模型, 可以发展出一个自动分配器以快速、自动地解决可寻址测试芯片的测试结构分配问题。研究表明, 基于线性规划的自动分配器可以在数分钟内完成上千个测试结构的自动分配; 同时, 自动分配器会在考虑各种测试结构分配的基本规则的前提下, 在测试单元空间利用率上得到一个最优化组合方案。

关键词: 可寻址测试芯片; 分配算法; 线性规划

中图分类号: TP312

文献标志码: A

文章编号: 1001-4551(2013)09-1147-06

Research of placement of test structure in addressable test chip

SAHO Kang-peng, SHI Zheng, ZHANG Pei-yong

(Institute of VLSI Design, Zhejiang University, Hangzhou 310027, China)

Abstract: In order to relieve the disadvantage in efficiency and reliability caused by lack of automatic way in addressable test chip design, aiming to the placement of the test structure, the linear programming was investigated. After researching the manual way in placement of test structure, a set of rules was summarized and expressed as multivariate inequality, and the mathematical model based on linear programming was established and developed into an automatic placer which will help the designer finish the placement of test structure quickly and automatically. The experimental result indicates that the automatic placer can address placement of thousands of test structure in minutes, and solution is optimized in area efficiency.

Key words: addressable test chip; placement; linear programming

0 引 言

随着 CMOS 工艺进入深亚微米水平, 为了保证芯片生产线上的成品率, 对测试芯片的需求变得越来越复杂, 为了分析产品的性能以及保证制造流水线的成品率, 需要在晶圆上部署越来越多的测试结构。目前, 芯片代工厂还是主要用传统的测试芯片来侦测流水线上的成品率状况。在传统的测试芯片中, 每个测试芯片的引脚需要单独地连接到 PAD 上。PAD 比较大, 这样导致了单位面积上 PAD 的数量无法达到相对比较多的数

量级, 而 PAD 的数量决定了实际可以进行测试的测试结构的数量, 所以传统测试芯片的面积利用率非常低。传统的测试芯片具有结构简单、设计方便等优点, 但传统测试芯片极低的面积利用率使其越来越难以满足如今为保证流水线的成品率所必需的数据条件。

为了提高测试芯片的面积利用率, 文献[1-3]的作者作了一些相关的尝试。文献[4]的作者则进一步描述了一种具有极高面积利用率的测试芯片的设计及制造方法——可寻址测试芯片。可寻址测试芯片采用了类似静态记忆体芯片的周围地址译码电路和堆叠摆放

收稿日期: 2013-01-17

基金项目: 国家自然科学基金资助项目(61204111)

作者简介: 邵康鹏(1988-), 男, 浙江温州人, 主要从事集成电路 DFM 及 EDA 方面的研究. E-mail: shaokp@vlsi.zju.edu.cn

通信联系人: 史 峥, 男, 副教授, 硕士生导师. E-mail: shiz@vlsi.zju.edu.cn

式的测试单元。每个测试单元纵向重叠摆放多个用于监测半导体生产工艺缺陷的测试结构,周围地址译码电路为每次测试唯一地确定一个测试结构,测试信号通过寻址,进入对应的测试结构进行测量。通过引入译码电路,基于同样数量的 PAD,可以测试更多的测试结构,从而大大提高了芯片的面积利用率。

然而,可寻址测试芯片的设计复杂度阻碍了其推广。可寻址测试芯片由两部分组成:基本测试单元以及外围电路。设计可寻址测试芯片的困难之处在于外围复杂的译码电路设计以及测试结构的分配问题。可寻址测试芯片的基本测试单元可以通过堆叠的摆放方式,将不同掩模层的测试结构放在同一个测试单元中。同时,尺寸较小但属于同一个掩模层的测试结构也可以放在同一个基本测试单元中,只要保证基本测试单元的面积足够即可。

文献[5]的作者在对可寻址测试芯片的自动化设计领域做了一些尝试:他们将可寻址测试芯片的设计分为外围译码电路的设计以及可寻址测试芯片基本单元的设计。可寻址测试芯片的基本单元实际上就是测试结构的一个容器。基本单元设计的复杂性主要在于在该阶段,设计者需要将成千上万个测试结构合理地分配到可寻址测试芯片的基本测试单元中。而在分配的过程中,需要考虑诸如测试结构的掩模层冲突、基本测试单元的容量对测试结构尺寸大小的约束、基本测试单元绕线资源对于测试结构引脚数量的约束等问题。在分配的过程中,一个最基本的限制就是在同一个基本单元中,有相同掩模层的不同的测试结构不能出现重叠的情况。文献[5]通过作者的直觉来进行基本单元的分配,同时手动计算基本单元中各个测试结构的坐标。显而易见,这样做效率低下,同时基本不可能得到一个足够优化的结果。同时,如果测试结构作了部分修改,设计者很有可能需要重新进行手动分配和坐标计算,而这在工程应用中是无法接受的。

所以,设计者需要一个工具可以帮助他们自动完成可寻址测试芯片基本单元的设计,而且在相对较短的时间内得到一个优化的分配结果。本研究将会基于线性规划对可寻址测试芯片的基本单元设计阶段建立数学模型,以实现可寻址测试芯片基本单元设计的自动化。

1 问题描述

文献[4]将可寻址测试芯片分为外围译码电路以及基本测试单元,可寻址测试芯片原理如图 1 所示。基本测试单元中含有测试结构。通过引入译码电路,测试芯片只需要少量的 PAD 就可以管理大量的测试结构,大大增

加了测试芯片的面积利用率。同时,如果引入层叠式的设计模式(占用了不同掩模层的测试结构可以重叠地放置在一起)则可以进一步优化测试芯片的面积利用率。

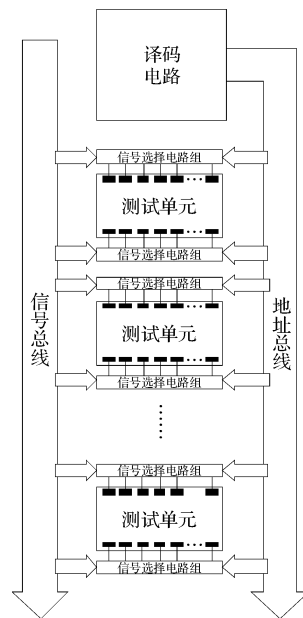


图 1 可寻址测试芯片原理

基本测试单元的分配问题是要将大量的测试结构合理地分配到如图 1 所示的测试单元中,并确定其具体坐标,测试单元版图如图 2 所示。基本测试单元的分配阶段的目标是得到一个分配方案,使每个测试单元的利用率尽可能高。在分配的过程中,需要考虑的问题有:①测试结构的掩模层。测试结构所占据的掩模层会各不相同,所以在测试单元中,每个测试结构只允许占据对应的掩模层,对于占据不同掩模层的测试结构,允许被分配到同一测试结构中,并采用堆叠的摆放方式,测试单元纵向切面示意图如图 3 所示;②测试结构的大小。测试结构的尺寸大小同样存在差异,有的测试结构需要占据整个测试单元,而有的测试结构只需要占据测试单元的一部分,如图 2 所示;③测试结构所需的绕线资源。每个测试单元可以提供的绕线资源是由分布在测试单元两边的伪终端数量决定,每个测试结构所需的绕线资源也不尽相同,所以每个测试单元中,测试结构的绕线资源总和不能超过测试结构所可以提供的绕线资源。

基本测试单元的分配问题划分成两个阶段:全局分配以及局部分配。全局分配是将测试结构分配到每个基本测试单元中且满足约束条件,而局部分配则是计算每个基本单元中测试结构的具体坐标。当完成全局分配之后,每个基本测试单元中测试结构的数量已经很少,并且确定存在一种摆放方式使测试结构全部分布到基本测试结构的相应掩模层且不违反之前提到的诸多约束。所以研究者可以简单的采用动态规划的

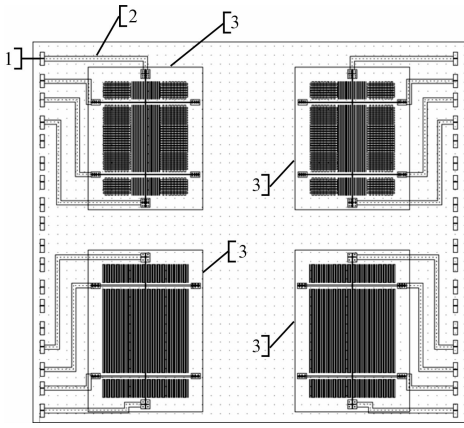


图2 测试单元版图

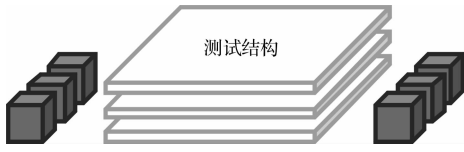


图3 测试单元纵向切面示意图

方法,遍历所有的摆放方式,通过相应的剪枝来快速地找到合理的解。本研究之后的讨论将会集中在基本测试单元的分配问题中的全局分配阶段。

测试结构全局分配类似于装箱问题。装箱问题的一个最简单的解决方案是使用 First fit 算法(遍历容器序列,将部件放入第一个合适的容器中)。First fit 算法可以快速地解决问题,它的时间复杂度是 $\Theta(n \log n)$ 。First fit 算法的不足之处在于通常情况下无法得到最优解,尤其是在测试结构比较复杂的情况下。当引入一些约束条件时,则会进一步恶化。文献[6]提出了一个组合启发式算法来解决一个基本的三维装箱问题,但没有考虑装箱问题的诸多约束条件。而对于可寻址测试芯片基本测试单元设计阶段的测试结构分配问题,需要考虑很多约束条件,如测试结构的掩模层约束,测试结构之间的摆放规则约束,每个基本测试单元的绕线资源约束等等。同时,设计者在分配过程中会有很多限制条件,如某些测试结构需要被放置在一起,而有些测试结构则不能同时出现在一个基本测试单元中。对于多约束的问题,空间规划算法不太有效,而启发式算法存在局限性较大的问题。文献[7]则进一步提出了使用遗传算法解决多约束的规划问题是一个行之有效的解决方案。

线性规划^[8-9]是最优化问题中的重要领域之一。很多运筹学中的实际问题都可以用线性规划来表述。本研究的后续部分将尝试线性规划应用到可寻址测试芯片的基本单元设计中,通过线性规划来构建一个稳定并且具有良好扩展性的数学模型。

2 问题建模

综合上一章中对于可寻址测试芯片的全局分配问题的分析,可以进一步将可寻址测试芯片的分配问题抽象成以下表述现实:现有基本测试单元 $B = \{b_1, b_2, \dots, b_n\}$,以及测试结构 $I = \{i_1, i_2, \dots, i_m\}$,找到一个对于 I 的分割 $S = \{S_j | 1 \leq j \leq n\}$ 且 $S_1 \cup S_2 \cup \dots \cup S_n \in I$, 如果 $\forall i_j \in S_j, i_j$ 必须满足基本测试单元 b_j 中相应的一些约束条件。

线性规划问题是目标函数和约束条件都是线性的最优化问题。使用线性规划来构建数学模型的关键在于成功使用一元线性多项式表述问题的优化目标以及使用线性不等式表述问题的约束条件。

2.1 确定变量语义

选择问题都可以转换成 0—1 问题,因为对于选择,只会有选中与不中两种可能,每个测试结构在基本测试单元中的分配情况可以通过一个变量非常方便地表示。如下所示的分配矩阵可以用来表示测试结构在基本测试单元中的分配情况:

$$X = \begin{cases} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{cases}, x_{ij} = \begin{cases} 0 \\ 1 \end{cases} \quad (1)$$

式中: x_{ij} —第 j 个测试结构是否被分配到了第 i 个基本测试单元,1—测试结构被分配到对应的基本测试单元中。

一个测试结构可以分配到不同的基本测试单元中,可以用下面的不等式来限制每个测试结构的重复次数:

$$\sum_{i=1}^n x_{ij} \leq r \quad (2)$$

式中: r —第 j 个测试结构需要放置的次数。

当然,每个测试结构的分配信息可以通过一个向量来表示: $ITEM_i = (x_{1i}, x_{2i}, \dots, x_{ni})$ 。对于每个基本测试单元,也可以通过一个向量来表示分配结果: $BLOCK_i = (x_{i1}, x_{i2}, \dots, x_{im})$ 。所以,分配矩阵还可以表示成如下的形式:

$$X = (ITEM_1^T \quad ITEM_2^T \quad \dots \quad ITEM_m^T) = \begin{pmatrix} BLOCK_1 \\ BLOCK_2 \\ \vdots \\ BLOCK_n \end{pmatrix} \quad (3)$$

可寻址测试芯片全局分配的目标是将尽可能多的测试结构分配到基本测试单元中,所以数学模型的优

化目标是:

$$\text{Maximize: } P = \sum x_{ij} \quad (4)$$

2.2 构建约束方程

对于可寻址测试芯片的分配问题,其最基本的约束条件是掩模层约束:测试结构只能占据特定的掩模层,同时,在同一层掩模层,不同的测试结构不能有重叠的情况,即掩模层冲突。所以,在同一层掩模层的测试结构的面积总和不能超过基本测试单元的尺寸大小。对于每个测试结构,可以用一个向量来记录该测试结构在每一个掩模层的大小信息:

$$W = (w_1 \quad w_2 \quad \cdots \quad w_k), w_i = \begin{cases} 0 \\ b_{box} \end{cases} \quad (5)$$

式中: k —掩模层的数量。

向量中的元素指示了该测试结构的组件在当前掩模层中的尺寸大小,当测试结构占用了某个掩模层,则掩模层向量中对应的元素应为测试结构实际的尺寸大小。如果考虑到设计规则(Design Rule),可以对实际的尺寸大小作一次扩展,以防止测试结构的距离太小。因此,下面的掩模层信息矩阵可以用来指示所有测试结构在对应掩模层上的尺寸信息:

$$WM = \begin{pmatrix} W_1 \\ W_2 \\ \vdots \\ W_m \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1k} \\ w_{21} & w_{22} & \cdots & w_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mk} \end{pmatrix} \quad (6)$$

式中: w_{ij} —第 i 个测试结构在第 j 层掩模的尺寸信息。

同时,不同的基本测试单元可以有不同的掩模层,同样可以用一个向量来表示每个基本测试单元的掩模层信息: $L_i = (l_{i1} \quad l_{i2} \quad \cdots \quad l_{ik})$ 。所以,对于基本测试单元的掩模层信息矩阵如下所示:

$$L = \begin{pmatrix} L_1 \\ L_2 \\ \vdots \\ L_n \end{pmatrix} = \begin{pmatrix} l_{11} & l_{12} & \cdots & l_{1k} \\ l_{21} & l_{22} & \cdots & l_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nk} \end{pmatrix} \quad (7)$$

可寻址测试芯片的掩模层约束条件是在基本测试单元中,每个掩模层上,测试结构的面积总和不能超过基本测试单元的物理尺寸。综上所述,可寻址测试芯片分配问题的掩模层约束条件可以用如下的不等式表示:

$$X \times WM \leq L \quad (8)$$

综上所述,可寻址测试芯片的测试结构全局分配问题的一个最基本的数学模型可以表述为如下的形式:

$$\text{Variables: } X = \{x_{ij} | 1 \leq i \leq n, 1 \leq j \leq m\} \quad (9)$$

$$\text{Maximize: } P = \sum x_{ij} \quad (10)$$

$$\text{Subject to: } X \times W \leq L \quad (11)$$

$$\sum_{i=1}^n x_{ij} \leq r_j \quad (12)$$

迄今,之前描述的只是基于可寻址测试芯片的测试结构分配问题的一个最基本的数学模型。但实际上,更多的情况下,在测试结构的分配过程中,往往会有许多额外的约束要求,以得到合理的分配结果。线性规划另外一个优点就是其扩展性:根据实际情况的不同,线性规划的数学模型可以很方便地进行扩展,以满足设计需求。

例如,一些测试结构会被希望分配到相同的基本测试单元中。那么本研究提出的数学模型可以通过添加一个约束方程即可满足要求。因为测试结构需要被分配到同一个基本测试单元中,两个测试结构的分配信息是相同的,可以表述成如下的等式:

$$ITEM_i - ITEM_j = 0 \quad (13)$$

将其展开可以表示为如下形式:

$$\begin{cases} x_{1i} - x_{1j} = 0 \\ x_{2i} - x_{2j} = 0 \\ \vdots \\ x_{ni} - x_{nj} = 0 \end{cases} \quad (14)$$

再者,由于测试结构的测试方法多不相同,每个基本测试单元中应该分配同种类型的测试结构,而不同类型的测试结果是不允许分配到同一个基本测试单元格中。假设 i, j 是两个不同类型的测试结构,那么在同一个基本测试单元 k 中, x_{ki} 和 x_{kj} 不能同时为 1。所以,数学模型中需要添加如下约束不等式来获得理想的分配效果:

$$ITEM_i + ITEM_j \leq 1 \quad (15)$$

即:

$$\begin{cases} x_{1i} + x_{1j} \leq 1 \\ x_{2i} + x_{2j} \leq 1 \\ \vdots \\ x_{ni} + x_{nj} \leq 1 \end{cases} \quad (16)$$

以此类推,数学模型中还可以添加诸如绕线资源等一系列约束条件。

3 优化策略

前几章已经为可寻址测试芯片的测试结构全局分配问题构建了一个基本的数学模型,在该数学模型的基础上可以利用 GLPK (GNU Linear Programming Kit)^[10-11] 构建一个简易的用于可寻址测试芯片的自动分配工具。

显而易见,GLPK 的求解速度与待求解的问题规模相关。假设待求解的问题包含了 n 个测试结构, m 个基本测试单元以及 k 层掩模,在不考虑额外的约束

条件,一个最基本的数学模型将会有 $n \times m$ 个变量,与此同时,需要构建 $m \times k$ 个层约束不等式和 n 个变量约束不等式。

但通常情况下,很多测试结构会有相同的掩模层和相似的尺寸大小。所以,研究者可以利用一些策略来减小数学模型的规模:

(1)将在相同掩模层并且尺寸大小相似的测试结构组成一个大的测试结构。

(2)将在相同掩模层并且尺寸大小相似的测试结构当作相同的个体看待。

(3)将在不同掩模层并且尺寸大小相似的测试结构组成一个大的测试结构。

通过以上的措施,数学模型的规模会得到有效地减少。

4 实验数据

4.1 分配效果

由于可寻址测试芯片设计规则的限制,测试结构的分配不是随意的,一些测试结构是不能同时分配到一个测试单元中。其中最主要的规则是:

(1)采用四端测量方法的测试结构和采用二端测量方法的测试结构不能同时分配到一个测试单元中。

(2)对于同一个测试单元来说,每个测试结构在 X 方向上的尺寸与测试单元在 X 方向上的尺寸的比值 $w = x_i/x_b$ (尺寸比值)。 $w > 1/2$ 和 $w < 1/2$ 的测试结构不能分配到一个测试单元中,因为可寻址测试芯片对于这两种测试结构会采取不同的布线策略。

测试单元中分配了采用四端测量法的测试结构如图5所示。图5的测试单元在不同掩模层的分配效果示意图如图6、图7所示。图4的测试单元的3维示意图如图7所示。该测试单元在 Metal1 及以下掩模层的版图示意如图5所示,从图中可以看出,分配时,4个尺寸较小的测试结构被组合在一起,已尽量多的利用测试单元的资源。该测试单元在 Metal2 掩模层的版图示意如图6所示。在 Metal2 层,分配了两个相对较大的测试结构, Metal4 掩模层的分配情况与 Metal2 类似。该测试单元总共分配了8个采用四端测量法的测试结构,使用的掩模层包括 Metal1, Metal2, Metal4。该测试单元可以提供的布线资源为8个,分配到该测试单元中的每个测试结构都需要1个布线资源,所以没有足够的布线资源可以提供给更多的测试结构,因此无法进一步分配占用 Metal3 掩模层的测试结构到该测试单元中。同时,从图中可以看到,该测试单元中只出现了尺寸比值小于1/2的测试结构。上文所提到的数学模型中的约束条件很好地将在 X 方向上尺寸

比例小于1/2的测试结构和尺寸比例大于1/2的测试结构进行了隔离。

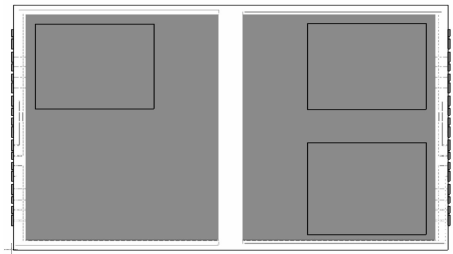


图4 测试单元版图

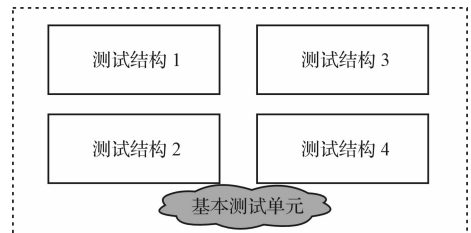


图5 Metal1 分配结果示意图

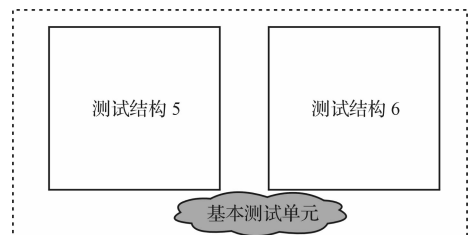


图6 Metal2/3 分配结果示意图

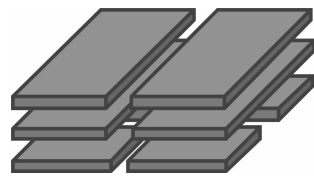


图7 测试单元三维示意图

4.2 运行效率

基于可寻址测试芯片的设计自动化的研究相对较少,也没有解决可寻址测试芯片设计中遇到的问题的针对性算法,所以本研究仅列出了在研究过程中一些运行数据,可以与现有可寻址测试芯片手动/半自动设计的效率做一个直观比对。对于一个包含1024个测试结构,同时无需考虑测试单元层叠摆放需求,文献[5]中提出的可寻址测试芯片半自动化的设计方法可以让分配阶段消耗的时间由超过20天压缩至2~3天。而基于上文提出的数学模型设计的自动分配器的运行效率如表1所示,其中第一行为测试结构的数量,第二行则是自动分配的运行时间。

表1 运行时间统计

数量	500	1 000	1 500	2 000	2 500	3 000
时间/s	0.07	0.72	2.47	4.09	20.2	40.9

通过表 1 的数据可以发现,当分配问题的规模增大时,基于线性规划的自动分配算法的性能退化比较严重。这是因为本研究提出的线性规划的数学模型中的变量的语义代表测试结构是否分配到对应的测试单元中,变量必须为 0 或 1。因此本研究在使用 GLPK 实现文章中提到的数学模型时,采用了 GLPK 的整形规划解法器(MIP Solver)。相较于常规的线性规划解法器(LP Solver),MIP Solver 的性能要差很多。同时,在规模较大时,容易陷入死迭代。所以在问题的规模比较大时,可以考虑将问题分解成几个小问题,以得到较好的运行性能。虽然这样不会得到一个全局优化的结构,但通过引入合理的分批策略,可以使分配结果不至于退化太多。

4.3 分配结果统计

4.1 节已经展示了自动分配算法结果中局部测试单元的版图示意。从示意图中可以看到,自动分配算法的分配结果已经达到了预想的效果。接下来,笔者将展示每个测试单元的空间利用率的统计信息。

其中,实验的样本来源于 246 种不同类型的测试结构模板,这些测试结构模板覆盖了采用四端测量法的测试结构和采用二端测量法的测试结构。基于这些模板,赋予不同的特征尺寸,共生成了 13 249 个具体的测试结构用于分配。分配结果的统计数据如表 2、表 3 所示。

表 2 采用四端测量法测试单元统计数据

掩模层	Metal 1	Metal 2	Metal 3	Metal 4	Metal 5
利用率/(%)	100	75.9	75.9	39.01	1.79

表 3 采用四端测量法测试单元统计数据

掩模层	Metal 1	Metal 2	Metal 3	Metal 4	Metal 5
利用率/(%)	100	83.82	98.77	61.03	12.25

从表 2、表 3 中可以看出测试结构的分布情况,测试结构掩模层的需求主要集中在 Metal1、Metal2、Metal3,其中 Metal1 的需求量最大。可寻址测试芯片测试结构分配问题最基本的约束条件是掩模层约束:每个测试结构只能出现在对应的掩模层中,例如一个含有 Metal1 的测试结构必须被分配到测试单元的 Metal1 掩模层中。由于测试单元数量的限制,包含了 Metal1 掩模层的测试结构并没有全部分配到测试单元中。上文提到的线性规划的数学模型的优化目标是将尽可能多的测试结构分配到测试单元中,以使测试单元的利用率最高,所以在分配含有 Metal1 掩模层的测试结构时,自动分配算法会寻求一种最大化测试单元空间利用率的测试结构组合方案。从表 2、表 3 的数据中可以发现,对于包含 Metal1 掩模层的测试结构,恰好存在可以完全利用 Metal1 掩模层空间的分配方案。

5 结束语

在自动分配算法的帮助下,可寻址测试芯片的测试结构分配可以很方便地完成。假如一个项目有几千个测试结构,往往需要设计者花费数天时间来完成测试结构的分配。但在自动分配算法的帮助下,测试结构的分配可以快速地并且不需要人为参与地完成。另外,自动分配算法在极大提高了可寻址芯片设计效率的同时,可以显著提升分配结果的稳定性。

参考文献(References):

- [1] DOONG K Y Y, BORDELON T J, HUNG L J, et al. Field-configurable test structure array(FC-TSA): enabling design for monitor, model and manufacturability, Semiconductor Manufacturing[J]. *IEEE Transactions on*,2008,21(2):169-179.
- [2] SMITH B, ARRIORDAZ A, KOLAGUNTA V, et al. A novel biasing technique for addressable parametric arrays[J]. *Semiconductor Manufacturing, IEEE Transactions on*, 2009,22(1):134-145.
- [3] HESS C, SQUCCIARINI M, SHIA Y, et al. High density test structure array for accurate detection and localization of soft fails[C]. *Microelectronic Test Structures,2008, ICMTS 2008, IEEE International Conference on*,2008:131-136.
- [4] 郑勇军.用于物理失效分析的改进型可寻址测试芯片及制作方法:中国,201010612475.8[P].2012-10-03.
- [5] ZHANG Bo, PAN Wei-wei, ZHENG Yong-jun, et al. A fully automated large-scale addressable test chip design with high reliability[C]. *European Conference on Circuit Theory and Design(ECCTD)*. Sweden:[s. n.],2011:61-64.
- [6] 张德富,魏丽军,陈青山,等.三维装箱问题的组合启发式算法[J]. *软件学报*,2007,18(9):2083-2089.
- [7] 何大勇,查建中,姜义东.遗传算法求解复杂集装箱问题方法研究[J]. *软件学报*,2001,12(9):1380-1385.
- [8] SCHRIJVER A. *Theory of Linear and Integer Programming* [M]. Wiley,1998.
- [9] CORMEN T H, CHARLES E, RIVEST R L, et al. *Introduction to Algorithms*[M]. Second Edition. MIT Press and McGraw-Hill,2001.
- [10] CERO R, IBM Software Group. *The GNU Linear Programming Kit, Part 1: Introduction to linear optimization* [DB/OL]. [2006-08-08]. <http://www.ibm.com/developerworks/linux/library/l-glpk/>.
- [11] WEST A J, MODAL S, PATRA D, et al. A novel high speed automatic layout system to place and route test structures for parametric test capability [C]. *Microelectronic Test Structures, 2008, ICMTS 2008, IEEE International Conference on*. Edinburgh:[s. n.],2008:71-75.