

“十一五”国家重点图书

计算机科学与技术学科前沿丛书

计算机科学与技术学科研究生系列教材（中文版）

---

# 超大规模集成电路 物理设计理论与算法

---

徐宁 洪先龙 编著

---



清华大学出版社



“十一五”国家重点图书

计算机科学与技术学科前沿丛书

计算机科学与技术学科研究生系列教材（中文版）

---

# 超大规模集成电路 物理设计理论与算法

---

徐宁 洪先龙 编著

---



清华大学出版社

北京

## 内 容 简 介

本书根据集成电路芯片物理设计的流程,首先介绍集成电路物理设计中的基础数学知识、数据结构和文件格式,然后对物理设计中的各个阶段进行详细的论述,包括电路系统的划分、布图规划和布局算法、网络布线算法、总体布线算法、通道布线算法、布图领域最新的相关技术以及三维集成电路布图等算法。

本书覆盖面广,内容由浅入深,满足了当前集成电路向 SoC、大规模和高性能发展的需求,既可作为高等院校计算机系和电子工程系从事 VLSI 设计和 VLSI CAD 的研究生的教材或教学参考书,也可作为从事 VLSI CAD 技术研究和开发人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

超大规模集成电路物理设计理论与算法/徐宁等编著. —北京:清华大学出版社,2009.9  
(计算机科学与技术学科前沿丛书. 计算机科学与技术学科研究生系列教材(中文版))  
ISBN 978-7-302-20828-0

I. 超… II. 徐… III. 超大规模集成电路—设计—研究生—教材 IV. TN470.2  
中国版本图书馆 CIP 数据核字(2009)第 156746 号

责任编辑:袁勤勇 李玮琪

责任校对:白 蕾

责任印制:李红英

出版发行:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260 印 张:16.75

字 数:390 千字

版 次:2009 年 9 月第 1 版

印 次:2009 年 9 月第 1 次印刷

印 数:1~2000

定 价:31.00 元

---

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。  
联系电话:010-62770177 转 3103 产品编号:031698-01

# 前言

21 世纪,世界全面进入信息时代。作为信息产业的支柱,超大规模集成电路(Very Large Scale Integration, VLSI)的设计和制造在推动经济发展、社会产业结构和生活方式的变革中的作用日益增长。

从 20 世纪 50 年代开始,集成电路制造技术经历了从小规模集成(SSI)、中规模集成(MSI)到大规模集成(LSI)阶段,乃至进入超大规模集成(VLSI)和甚大规模集成(Ultra Large Scale Integration, ULSI)阶段。尤其在过去的 30 年中,集成电路几乎完全遵循摩尔定律发展,即集成电路的集成度每隔 18 个月就翻一番。进入 20 世纪 90 年代以及 21 世纪以后,其设计规模由 VLSI、ULSI 向 G 规模集成(Giga-Scale Integration, GSI)的方向发展,于是,越来越多的功能,甚至是一个完整的系统都能够被集成到单个芯片之中。电子系统设计已从板上系统(System on Board, SoB)、多芯片模块(Multi-Chip Modules, MCM)进入到系统级芯片(System on Chip, SoC)时代。

集成电路的飞速发展体现出如下特点:特征尺寸越来越小,芯片面积越来越大,单片上的晶体管数目越来越多,时钟频率越来越高,电源电压越来越低,布线层数越来越多,I/O 引线越来越多。美国半导体工业协会 SIA 组织给出了 1997 年到 2009 年美国集成电路工艺发展趋势。随着集成度的提高,芯片内部晶体管数目越来越多,集成电路设计的复杂性越来越高,传统的手工设计和适应小规模的设计模式已经不再适用。为了设计复杂的大规模集成电路,人们越来越借助于电子设计自动化(EDA)工具。因此,越来越多的人致力于研究集成电路计算机辅助设计的算法,并且将这些算法集成到 EDA 工具中。

在集成电路的设计过程中,物理设计(或版图设计)是其中最费时的一步。它是 VLSI 设计过程中的一个重要环节,也是目前超大规模集成电路计算机辅助设计技术中最重要、最活跃的研究领域之一。它要把每个元件的电路表示转换成几何表示,同时,元件间的线网也被转换成几何连线图形。电路的几何表示叫版图,版图设计要符合与制造工艺有关的设计规则要求。版图设计可以有許多不同的模式,从对布局布线位置的限制和对布局模块的限制来分,可分为如下几种常用的模式:积木块模式(Building Block Layout, BBL)、标准单元模式(Standard Cell)、门阵列模式(Gate Array)和门海模式(Sea-of-Gates)等。版图设计自动化指根据电路性能和工艺水平的要求,自动完成电路元件或功能单元在芯片上的安置,实现它们所需要的互连关系,并尽可能使得电路占有的面积最小或性能最优。版图设计的输入是电路的元件说明和网表(可用原理图说明),其输出是设计好的版图。

由于物理设计的复杂性,整个版图过程往往被分为划分(Partition)、版图规划(Floorplanning)、布局(Placement)和布线(Routing)等若干子步骤进行,每一步骤完成整个物理设计的一个子功能。物理设计阶段划分为上面几个子阶段,必然导致整个物理设计过

程也是一个反复迭代求解的过程。随着工艺尺寸的不断减小和集成复杂度的增加,集成电路设计者面临全新的挑战,也给 EDA 研究人员提出了更高的要求。

本书对物理设计各个阶段的最新理论与算法进行了介绍,并将作者的研究成果融入到其中,可以使读者,特别是从事 VLSI CAD 领域的研究人员对集成电路物理设计领域的发展和研究趋势有一个全面的了解,从而为我国集成电路的发展做出应有的贡献。

作者

2009 年 5 月

# 目 录

|  |    |
|--|----|
| <b>第 1 章 超大规模集成电路布图问题、方法及版图设计自动化</b> ..... | 1  |
| 1.1 VLSI 设计流程 .....                        | 1  |
| 1.1.1 传统的 VLSI 设计流程 .....                  | 1  |
| 1.1.2 传统的布图设计过程 .....                      | 2  |
| 1.1.3 VLSI 设计流程的新趋势 .....                  | 3  |
| 1.1.4 VLSI 物理设计的新趋势 .....                  | 4  |
| 1.2 芯片费用和电性能的估计 .....                      | 5  |
| 1.3 布图模式 .....                             | 5  |
| 1.3.1 全定制版图模式 .....                        | 6  |
| 1.3.2 门阵列设计模式 .....                        | 8  |
| 1.3.3 标准单元设计模式 .....                       | 8  |
| 1.3.4 现场可编程门阵列 .....                       | 9  |
| 1.3.5 不同设计方法的比较 .....                      | 11 |
| 1.4 系统封装类型 .....                           | 11 |
| 参考文献 .....                                 | 14 |
| <b>第 2 章 VLSI 器件设计和制造的布图描述</b> .....       | 15 |
| 2.1 VLSI 制造工艺 .....                        | 15 |
| 2.2 设计规则 .....                             | 16 |
| 2.3 基本器件的版图实例 .....                        | 17 |
| 2.4 工艺制造中的其他因素 .....                       | 18 |
| 2.5 VLSI 版图的几何表示 .....                     | 21 |
| 2.5.1 CIF 格式 .....                         | 21 |
| 2.5.2 EDIF 格式 .....                        | 22 |
| 2.5.3 GDSII 格式 .....                       | 26 |
| 2.5.4 OASIS 格式 .....                       | 30 |
| 2.6 单元的拓扑描述与网表描述 .....                     | 33 |
| 参考文献 .....                                 | 35 |

|                                      |    |
|--------------------------------------|----|
| <b>第 3 章 VLSI 布图的数学基础和数据结构</b> ..... | 36 |
| 3.1 图的基本概念及其基本数据结构 .....             | 36 |
| 3.1.1 基本概念 .....                     | 36 |
| 3.1.2 图的基本数据结构 .....                 | 37 |
| 3.2 算法及其复杂度 .....                    | 39 |
| 3.2.1 算法问题及算法复杂性 .....               | 39 |
| 3.2.2 求解 NP-困难问题的方法 .....            | 40 |
| 3.3 解决布图问题的基本算法 .....                | 42 |
| 3.3.1 图论算法 .....                     | 42 |
| 3.3.2 确定性算法 .....                    | 50 |
| 3.3.3 随机算法 .....                     | 54 |
| 3.4 多目标优化 .....                      | 57 |
| 3.4.1 多目标优化问题的定义(MOP) .....          | 57 |
| 3.4.2 多目标优化方法 .....                  | 57 |
| 3.5 布图设计中涉及的基本数据结构 .....             | 63 |
| 3.5.1 版图数据的基本操作 .....                | 63 |
| 3.5.2 链表结构 .....                     | 64 |
| 3.5.3 基于 Bin 的结构 .....               | 65 |
| 3.5.4 邻接指针 .....                     | 66 |
| 3.5.5 角勾链 .....                      | 66 |
| 3.5.6 四叉树 .....                      | 70 |
| 3.5.7 各种版图数据结构的比较 .....              | 70 |
| 3.5.8 布图中模块和网表的数据结构 .....            | 71 |
| 3.5.9 树的数据结构 .....                   | 74 |
| 参考文献 .....                           | 77 |
| <b>第 4 章 电路系统的划分</b> .....           | 79 |
| 4.1 问题描述 .....                       | 79 |
| 4.2 代价函数和约束 .....                    | 80 |
| 4.3 解决划分问题的基本方法 .....                | 81 |
| 参考文献 .....                           | 86 |
| <b>第 5 章 布图规划</b> .....              | 88 |
| 5.1 问题描述 .....                       | 88 |
| 5.2 布图结构的表示方法 .....                  | 89 |
| 5.2.1 可二划分结构 .....                   | 89 |
| 5.2.2 不可二划分结构 .....                  | 90 |
| 5.3 解决布图规划问题的方法 .....                | 96 |

|              |                       |            |
|--------------|-----------------------|------------|
| 5.3.1        | 簇生长法 .....            | 96         |
| 5.3.2        | 解析法 .....             | 98         |
| 5.3.3        | 对偶图技术 .....           | 101        |
| 5.3.4        | 模拟退火算法 .....          | 105        |
| 5.3.5        | 其他方法 .....            | 108        |
| 5.4          | 高层次综合与布图结合技术 .....    | 109        |
| 5.4.1        | 传统的布图规划 .....         | 109        |
| 5.4.2        | 传统的高层次综合 .....        | 109        |
| 5.4.3        | 高层次与布图规划结合 .....      | 110        |
|              | 参考文献 .....            | 116        |
| <b>第 6 章</b> | <b>  布局 .....</b>     | <b>119</b> |
| 6.1          | 问题描述 .....            | 119        |
| 6.2          | 代价函数与约束 .....         | 120        |
| 6.2.1        | 线长估计 .....            | 120        |
| 6.2.2        | 总线长最小化 .....          | 123        |
| 6.2.3        | 最大割线数目最小化 .....       | 123        |
| 6.2.4        | 最大密度最小化 .....         | 123        |
| 6.2.5        | 性能最大化 .....           | 124        |
| 6.2.6        | 其他约束 .....            | 124        |
| 6.3          | 解决布局问题的方法 .....       | 125        |
| 6.3.1        | 基于划分的方法 .....         | 125        |
| 6.3.2        | 启发式最小割算法 .....        | 129        |
| 6.3.3        | 解析方法 .....            | 131        |
| 6.3.4        | 模拟退火 .....            | 137        |
| 6.3.5        | 神经网络 .....            | 138        |
| 6.3.6        | 遗传算法 .....            | 141        |
| 6.3.7        | 其他 .....              | 143        |
|              | 参考文献 .....            | 143        |
| <b>第 7 章</b> | <b>  线网布线 .....</b>   | <b>146</b> |
| 7.1          | 迷宫算法 .....            | 147        |
| 7.1.1        | 基本的迷宫算法——Lee 算法 ..... | 147        |
| 7.1.2        | 迷宫算法的改进 .....         | 150        |
| 7.1.3        | 迷宫算法的比较 .....         | 152        |
| 7.1.4        | 迷宫算法中提高布线效率的方法 .....  | 153        |
| 7.1.5        | 多端线网布线 .....          | 156        |
| 7.1.6        | 多层布线 .....            | 156        |
| 7.2          | 线探索法 .....            | 157        |



|            |                          |            |
|------------|--------------------------|------------|
| 7.3        | 布线顺序的影响及其处理 .....        | 158        |
| 7.4        | 整体布线 .....               | 160        |
| 7.5        | 电源地线线网布线 .....           | 162        |
| 7.6        | 层次式布线方法 .....            | 163        |
| 7.7        | 全芯片多级布线方法 .....          | 163        |
| 7.8        | 最新的研究成果 .....            | 164        |
|            | 参考文献 .....               | 165        |
| <b>第8章</b> | <b>总体布线</b> .....        | <b>167</b> |
| 8.1        | 总体布线问题 .....             | 167        |
| 8.1.1      | 总体布线图 .....              | 167        |
| 8.1.2      | 总体布线问题定义 .....           | 168        |
| 8.2        | 总体布线算法的分类 .....          | 169        |
| 8.3        | 总体布线图上的斯坦纳树算法 .....      | 171        |
| 8.3.1      | 基于最短路径的算法 .....          | 172        |
| 8.3.2      | 基于最小代价生成树的算法 .....       | 172        |
| 8.3.3      | 基于可分离性的算法 .....          | 173        |
| 8.3.4      | 基于非矩形边斯坦纳树的算法 .....      | 175        |
| 8.3.5      | Dreyfus-Wagner 算法 .....  | 175        |
| 8.3.6      | 最小最大权重边的斯坦纳树算法 .....     | 177        |
| 8.4        | 总体布线算法 .....             | 178        |
| 8.4.1      | 串行布线和拆线重布算法 .....        | 178        |
| 8.4.2      | 基于加权的斯坦纳树算法 .....        | 179        |
| 8.4.3      | 基于整数规划的方法 .....          | 180        |
| 8.4.4      | 基于网络流的总体布线算法 .....       | 183        |
| 8.4.5      | 基于拥挤度分析的并行层次迭代布线算法 ..... | 188        |
| 8.4.6      | 启发式总体布线 .....            | 190        |
| 8.5        | 其他最新算法 .....             | 192        |
| 8.5.1      | FGR .....                | 193        |
| 8.5.2      | BoxRouter .....          | 194        |
| 8.5.3      | MaizeRouter .....        | 195        |
| 8.5.4      | FastRoute .....          | 195        |
| 8.6        | 时钟布线技术 .....             | 197        |
| 8.6.1      | 时钟布线的研究对象 .....          | 198        |
| 8.6.2      | 时钟布线问题描述 .....           | 199        |
| 8.6.3      | 时钟布线算法简介 .....           | 199        |
|            | 参考文献 .....               | 203        |

|                                 |     |
|---------------------------------|-----|
| <b>第 9 章 其他布图问题</b> .....       | 207 |
| 9.1 通孔优化问题 .....                | 207 |
| 9.1.1 通孔最少化 .....               | 207 |
| 9.1.2 带约束的通孔最小化问题 .....         | 208 |
| 9.1.3 二层 CVM 问题的图表示法 .....      | 209 |
| 9.1.4 无约束的通孔最小化 .....           | 212 |
| 9.1.5 交叉通道的 TVM 问题的最优化算法 .....  | 212 |
| 9.1.6 一般的 $k$ -TVM 问题的近似解 ..... | 213 |
| 9.1.7 基于拓扑解的布线 .....            | 213 |
| 9.1.8 通孔最少化算法 .....             | 214 |
| 9.1.9 统一通孔最少化和线长最小化层分配算法 .....  | 218 |
| 9.1.10 最少通孔和最小线长分层的无向图表示 .....  | 219 |
| 9.2 时延和功耗优化 .....               | 220 |
| 9.2.1 时延和功耗双重驱动布局算法 .....       | 220 |
| 9.2.2 延迟模型 .....                | 221 |
| 9.2.3 问题定义 .....                | 221 |
| 9.2.4 功耗 .....                  | 223 |
| 9.3 芯片热模型和均匀分布 .....            | 226 |
| 9.3.1 热问题分析 .....               | 226 |
| 9.3.2 功耗和热的关系 .....             | 227 |
| 9.3.3 在芯片设计的不同层次上考虑热问题 .....    | 228 |
| 9.3.4 热问题的解决方案与技术 .....         | 230 |
| 9.3.5 小结 .....                  | 234 |
| 9.4 布图设计的可制造性 .....             | 234 |
| 参考文献 .....                      | 235 |
| <b>第 10 章 三维芯片布图问题</b> .....    | 241 |
| 10.1 三维芯片对集成电路带来的影响 .....       | 241 |
| 10.2 三维芯片的布图规划表示方法 .....        | 243 |
| 10.3 三维芯片布局算法 .....             | 245 |
| 10.4 三维芯片布线与热通孔规划 .....         | 247 |
| 10.5 三维芯片的热模型及求解算法 .....        | 249 |
| 10.5.1 热模型 .....                | 249 |
| 10.5.2 热模型的计算 .....             | 251 |
| 10.6 小结 .....                   | 253 |
| 参考文献 .....                      | 253 |
| 附录 .....                        | 255 |

# 第 1 章

## 超大规模集成电路布图问题、方法及版图设计自动化

自从晶体管于 20 世纪 40 年代后期、集成电路于 20 世纪 60 年代初期发明以来,经历了小规模集成 (SSI)、中规模集成 (MSI)、大规模集成 (LSI)、超大规模集成 (VLSI) 和特大规模集成 (ULSI) 阶段,目前进入了片上系统 (System on Chip, SoC) 时代。半导体制造技术和计算机产业一直都在按摩尔定律 (每隔 18 个月性能翻一番) 以惊人的速度迅速发展,并创造了人类历史上的“数字文明”。

伴随着集成电路的发展,电子设计自动化 (Electronic Design Automatic, EDA) 技术已成为电子设计技术的核心。EDA 是指以计算机为工作平台,融合了应用电子技术、计算机技术、智能化技术最新成果而研制成的电子计算机辅助设计 (Computer-Aided Design, CAD) 通用软件包,极大地推动了集成电路的发展。尤其在版图设计自动化方面,已经有适用于不同设计版图模式的 EDA 工具。版图设计自动化设计的几何图形很简单,包括矩形、直角多边形和连接线等,但是,布图算法涉及图论、线性规划、非线性规划、组合优化、运筹学等,布图算法的实现则涉及版图数据结构和数据管理。因此,本书重点讨论版图设计自动化中的算法和实现技术。

### 1.1 VLSI 设计流程

#### 1.1.1 传统的 VLSI 设计流程

超大规模集成电路 (Very Large Scale Integration, VLSI) 设计周期分为以下几个步骤: 系统规范说明、功能设计、逻辑设计、电路设计、物理设计、设计验证、制造、封装和测试<sup>[1-2]</sup>。VLSI 设计可能会在一个步骤中或在几个步骤之间反复交替进行。VLSI 电路的设计流程如图 1-1 所示。

其中,系统规范说明包括系统功能、性能和物理尺寸。此外,还需要考虑选择设计模式和制造工艺。最终结果是确定芯片尺寸、工作速度、功耗和系统功能。

在功能设计中主要考虑系统的行为特性,常用的方法是时序图或者表示各子模块间关系的的关系图。利用这些信息可以改进整个设计过程或简化后续的设计步骤。

逻辑设计中可以得到一个表示系统功能的逻辑结构并反复测试其正确性。设计者通常用文本、原理图或逻辑图表示设计,有时也用布尔方程表示设计。在设计过程中,还要对该逻辑结构进行模拟以验证其正确性,并对其进行优化设计或称为逻辑最小化。

电路设计时要考虑逻辑部件的电路实现,包括速度和功耗。此外,还要注意各种元件的电性能。通常用详细的电路图来表示电路设计。

物理设计即版图设计,是 VLSI 设计中最费时的一步。物理设计要把每个元件的电路表示转换为几何表示,同时,元件间连接的线网也被转换成几何连线图形。电路的几何表示称为版图。版图设计要符合与制造工艺有关的设计规则要求。由于版图设计的复杂性,往往把版图设计分成若干子步骤进行,如图 1-2 所示。

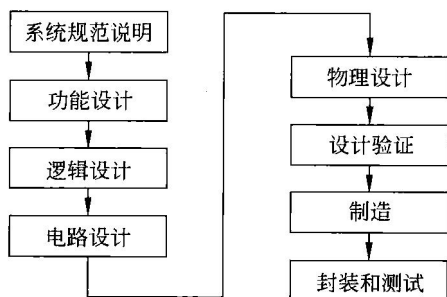


图 1-1 VLSI 设计流程

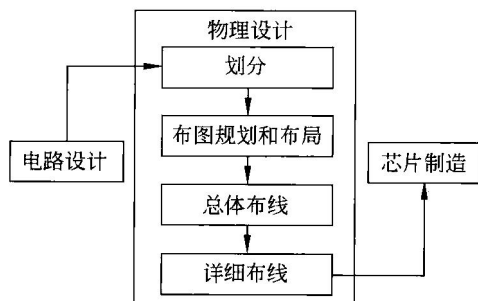


图 1-2 物理设计过程

设计验证也称版图验证,它确保版图设计完成后所得到的几何图形满足制造工艺要求和符合系统的设计规范。

芯片制造过程包括芯片准备、杂质注入、扩散和光刻等工艺。

在完成芯片制造后,要进行封装和测试。安置在印制电路板上的芯片可封装成双列直插式或引脚阵列式。用于多芯片模块的芯片可以不封装。

用户根据不同产品的性能要求、生产量的大小和设计周期的长短,有多种布图模式可以选择。VLSI 设计的布图模式主要包括积木块自动布图(Building Block Layout, BBL)、标准单元、门阵列、门海设计模式和 FPGA(Field Programmable Gate Array)。实现相同的功能,FPGA 是最快的方法,BBL 布图模式则最慢,然而 FPGA 占用的芯片面积最大、性能最差。反之 BBL 布图模式因为单元(或模块)的类型、大小和单元布局都很灵活,故能够设计出高性能的芯片。但由于 BBL 布图设计非常复杂,它的设计周期比其他布图模式设计周期要长,且自动化过程最难,因此要耗用更多的计算资源。

### 1.1.2 传统的布图设计过程

布图设计(Layout Design)也称为 VLSI 物理设计(Physical Design)。其输入是电路的元件说明和网表,输出是设计好的版图。即根据电路和工艺要求完成芯片上单元或功能块的安置,实现它们之间所需要的互连。

由于布图设计的复杂性,它又分为以下几个步骤:划分、布图规划和布局、总体布线、详细布线。布图设计流程如图 1-2 所示。

一个芯片可以包含数以百万计的晶体管,当今的 CPU 芯片已达到上 10 亿个晶体管。划分过程是将各部件分组形成多个模块(Block),各模块的大小和在芯片上的精确位置由布图规划和布局算法确定。布图规划和布局算法的目标是最小化芯片的面积同时满足约束条件,如模块间无重叠且完全互连等。总体布线和详细布线组成了复杂的布线过程,总体布线

确定一个线网的大致走线,它将各线网合理地分配到各布线区域中,以确保尽可能高的布通率;详细布线则是最终产生线网在芯片上的实际走线,及生成各线网的几何版图。压缩的任务是从各个方向上压缩芯片的版图,以期将芯片的总面积减小。

在整个集成电路设计过程中,物理设计是与产品研制和生产直接相关的一个设计过程,直接关系到芯片设计的周期、生产成本和产品质量。这一步骤也是以往人工设计中耗时最多和错误率最高的设计过程之一。因此,它也是近年来 EDA 工具中发展最快和自动化程度最高的领域之一。而且,随着 VLSI 向深亚微米,甚至纳米推进,它也是受工艺影响最大,面临的机遇和挑战最多的领域之一。

### 1.1.3 VLSI 设计流程的新趋势

随着 VLSI 向深亚微米,甚至纳米推进,集成电路的规模越来越大,系统更复杂,同时也带来了许多新的问题<sup>[2]</sup>。

(1) 互连线延迟增加:器件的尺寸变小的速度比互连线快,互连线的延迟已经大大超过门的延迟,它已经占整个路径延迟的 60%左右了,其解决方案是在芯片级互连线上插入缓冲器,因此必须在设计的早期就要进行规划。

(2) 互连线面积增加:器件占芯片面积的 60%~70%,互连线占面积的 30%~40%。在提供更多的金属连线层时,并不能降低互连线的面积,反而会使层间的通孔数增加,从而会阻碍上层的布线。

(3) 需求阶段就必须考虑后期设计的需要:在设计阶段的早期就必须对物理信息进行比较准确的估计。在功能设计阶段考虑芯片模块的规划;为后期阶段提供时序验证和时序约束。

(4) 综合:其优点是降低出错率和减少设计时间,缺点是增加芯片面积。

(5) 早期物理设计规划的好处:提高早期阶段估计的准确性,降低裸片尺寸,提高成品率。

由图 1-3 可以看到,新的 VLSI 设计流程更加复杂,需要各个层次之间的多次交互,这些是进入纳米工艺时必须考虑的,也是设计满足最终需要的保证。

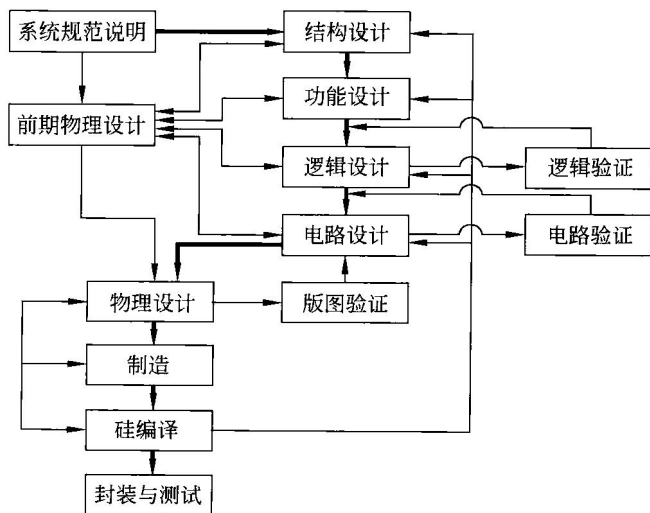


图 1-3 新的 VLSI 设计流程图

### 1.1.4 VLSI 物理设计的新趋势

新的 VLSI 设计流程对 VLSI 物理设计也提出了新的要求<sup>[2]</sup>。在该阶段必须考虑如下问题:

(1) 在设计的前期阶段就考虑互连线延迟。对全局信号线和总线进行早期规划,从而降低它们的长度,将它们布在金属层的最上层,以便使单位长度上的延迟降低。

(2) 低功耗设计。

(3) 基于信号完整性问题。在设计的不同阶段,结合布线信息,对设计进行有效的分析和指导,避免出现信号完整性问题。

(4) 可制造设计(Design For Manufacture,DFM)。对于物理设计,主要采用的 DFM 设计手段就是冗余通孔(Multi-Via)、连线散布(Wire Spreading)、金属填充(Metal Fill)等。冗余通孔就是尽可能采用多个通孔,从而避免一旦通孔的加工出现问题,导致整条线的电学特性出现较大偏差。连线散布是尽量把互连线之间的间距拉开,均匀分布,避免出现局部区域连线过密。金属填充是为了在化学机械抛光(CMP)时保证芯片内各区域互连线的性能一致性,在金属密度较低的区域加入一些浮空(或接地)的金属线,以满足金属密度的设计需求。

由图 1-4 可以看到,时序必须在物理设计的划分和布图规划/布局阶段进行考虑,通过多次反复达到设计要求,这也使得优化目标和约束增多,且有些目标或约束是相互矛盾的,这就必须在一定条件下进行折中,从而导致求解过程更加复杂。

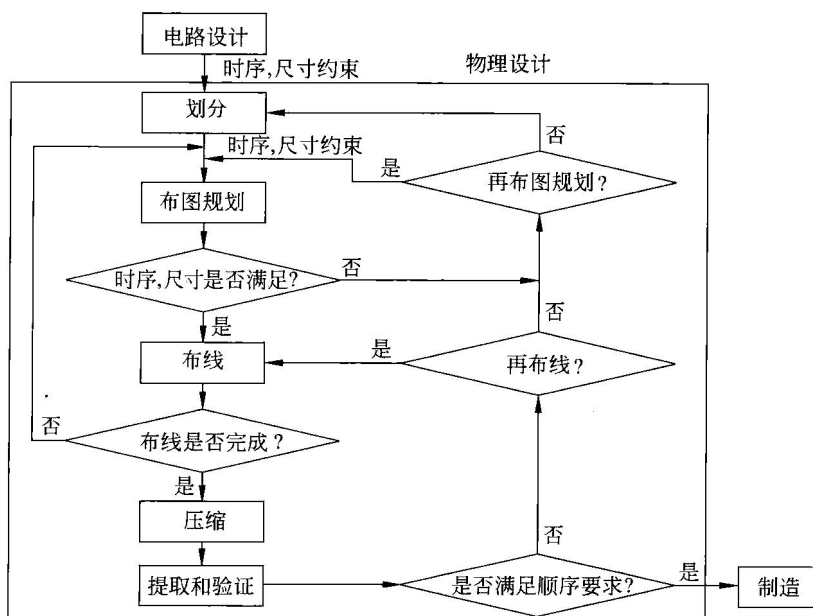


图 1-4 新的 VLSI 物理设计流程图

## 1.2 芯片费用和电性能的估计

VLSI 版图设计是一个十分复杂的过程,即使把它分成若干子过程,它仍然是一个很困难的任务。但用户又希望尽量缩短产品进入市场的时间。根据不同产品性能要求、生产产量的大小和设计周期的长短,有多种布图模式可供选择。就用户而言,总是希望在满足电性能要求的前提下,尽可能降低每个芯片的设计和研制费用,尽可能快地使产品进入市场。因此他需要综合考虑芯片的电性能要求、芯片的复杂性、所需设计周期(即设计费用)、芯片制造成品率(即制造费用)及可实现性等,选择一个合适的布图模式。

为了定量地分析和比较不同设计模式的费用和电性能,下面给出两个表示芯片费用和电性能的公式。

每个芯片的总费用可用下式表示

$$C_T = C_D/N + C_P/(y \times n) \quad (1.1)$$

其中: $N$ 是总产量, $C_D$ 是设计及制版费, $C_P$ 是每圆片的制造费用, $n$ 是每圆片上的芯片数, $y$ 是圆片成品率,即圆片上是成品的芯片数占总芯片数的百分比。式中的第一项表示分摊到每个芯片上的设计费用。第二项表示每个芯片的制造费用。

用芯片的可能最大延迟来表示芯片的电性能。芯片的最大延迟可用下式近似地表示

$$T_{pd} = T_{pd0} + V_L(C_w + C_g)/I_p \quad (1.2)$$

其中: $T_{pd0}$ 是电路器件本身的延迟, $V_L$ 是最大电压, $I_p$ 是尖峰时的电流, $C_w$ 是连线电容, $C_g$ 是扇出栅电容。

从式(1.1)看,对批量大的产品(即 $N$ 很大),第一项很小,对 $C_T$ 起主要作用的是第二项。为了降低 $C_P$ ,必须设法提高 $y$ 和 $n$ ,即减小芯片面积,进一步提高集成度以增加每圆片上的芯片数 $n$ 和成品率 $y$ 。对小批量产品(即 $N$ 很小),第一项起主要作用,即分摊在每个芯片上的设计和制版费用占总芯片费用的大部分。此时,主要是要降低 $C_D$ ,即尽可能缩短设计周期。从式(1.2)看,影响芯片性能的主要是 $C_w$ 和 $C_g$ 。要提高性能就要注意减小连线长度,减小芯片面积。式(1.1)和式(1.2)可作为选择布图模式的依据。

## 1.3 布图模式

布图阶段是一个非常复杂的过程,即使把它划分为很多子过程,其每一步也是非常困难的。在上市时间和成品率的双重压力下,必须采用严格的模式和设计周期,从而降低在布图阶段的复杂性。

本节介绍用于产生电路物理表示的布图方法,这些不同的布图方法大致可以分为两类:全定制布图和半定制布图。在全定制布图中,电路中的不同模块,在满足各模块互不重叠的条件下,可以任意摆放在硅晶片上的任何位置;而对于半定制布图模式,电路中的一些模块已经预先设计和摆放在硅晶片上的特定位置上了。

布图模式的选择依赖于很多因素,包括芯片的类型、成本和上市时间等。全定制布图主要用于大批量生产的芯片,需要设计者完成所有电路的设计,因此需要大量人力物力,灵活性好但开发效率低下。如果设计较为理想,全定制能够比半定制的专用集成电路

(Application Specific Integrated Circuit, ASIC) 芯片运行速度更快。半定制使用库里的标准逻辑单元(Standard Cell),设计时可以从标准逻辑单元库中选择 SSI(门电路)、MSI(如加法器、比较器等)、数据通路(如 ALU、存储器、总线等)、存储器甚至系统级模块(如乘法器、微控制器等)和 IP 核,这些逻辑单元已经布局完毕,而且设计得较为可靠,设计者可以较方便地完成系统设计。

### 1.3.1 全定制版图模式

版图设计方法可以从不同角度对它进行分类。如按设计自动化程度来分,可将版图设计方法分成手工设计和自动设计两大类。手工设计方法中又可分成基于几何图形的交互图形编辑方法和基于符号的交互图形编辑方法。手工设计的设计周期很长,每人每天可设计 5~10 个器件,且很容易出错。因此在设计完成后,总是需要一个版图验证软件进行设计正确性的检查。版图验证包括设计规则检查(DRC)、电学规则检查(ERC)、版图参数提取(LPE)、版图和原理图对照检查(LVS)等。但手工设计方法可使芯片面积比较小,芯片电性能比较好。根据式(1.1)和式(1.2),它比较适用于大批量生产的产品,如存储器、CPU 芯片等。

如按对布局布线位置的限制和对布局模块的限制来分,则可把设计方法分成全定制(Full-Custom)和半定制(Semi-Custom)两大类。我们也把它们称作全定制和半定制设计模式。全定制设计模式对模块(或单元)和布局位置没有限制,而半定制设计模式对单元的高度、电源线位置和单元引线端的引出方向都有一定的限制。并且单元也只能放置在规定的区域内或必须按行放置。半定制设计模式又可分成若干小类。下面我们介绍几种常用的设计模式的特点、技术内容和应用前景。

#### 1. 交互图形编辑器

这是一种手工设计方法,也是一种传统设计方法。设计者将手工设计好的版图草图用一个交互图形编辑器(Interactive Graphic Editor)输入至计算机并进行编辑。设计者也可直接在屏幕上绘制版图。编辑器提供有插入、移动、删除、复制、拉伸等命令,并辅以开窗、缩放、窗口移动等显示命令和文件处理命令,使设计者方便而自由地在屏幕上绘制和编辑版图。虽然此方法设计效率低,设计时间长,但由于用此方法设计可得到高集成度和高性能的芯片,因而仍广泛用于大批量产品的设计。如目前大量使用的存储器、CPU、家用电器的芯片都是使用手工设计和交互图形编辑方法设计的。

大部分 CAD 公司都提供实用的交互图形编辑器,如 CADENCE 和 MENTOR 等都有功能完善、性能良好的编辑工具。我国自行开发的 PANDA 系统也包括一个交互图形编辑器。除了工作站版本外,有些公司还专门提供微机上的版本。TANNER 就是一个广泛使用的微机上的交互图形编辑器。

由于手工设计方法不可避免地总是要产生错误,有必要在版图编辑后进行版图验证。上面已经提到,版图验证包括设计规则检查、电学规则检查、版图与原理图对照检查以及电路网表和版图寄生参数提取和后模拟。CAD 公司在提供交互图形编辑软件同时,也提供版图验证软件。CADENCE 的 DRACULA 就是一个十分著名的版图验证软件。设计规则检查软件用以检查各种违反工艺设计规则的错误,如连线宽度小于最小线宽规定,连线间距小



于最小间距等。电学规则检查用以检查各种连接性错误,如短路、开路、非法连接等。版图提取软件通过器件和网表的提取,将物理版图还原成电路图甚至逻辑图,然后与原始输入的原理图进行对照,以检查逻辑正确性。版图的寄生参数提取软件将用户事先未考虑的寄生电阻和电容计算出,并把它们合并入电路图,再次进行电路模拟以检查电路功能的正确性。一个性能良好的版图验证软件可以将设计错误消灭在芯片制造之前,确保芯片的正确性和一定的成品率。

## 2. 符号法版图设计方法

符号法版图设计方法(Synbolic Layout Approach)是另一种人工设计方法。它使用晶体管、通孔和连线的符号进行输入和编辑并产生一个拓扑版图,然后再根据给定的设计规则转换成物理版图。不同工艺的晶体管、通孔和连线的符号以及它们相应的物理版图由符号法版图编辑器事先制作好并存放在单元库中。器件尺寸和线宽可根据给定的设计规则而变化。直接由拓扑版图转换而成的物理版图往往有较多的冗余空间。因此,符号法版图编辑器中还提供一个压缩功能,以优化版图面积。图 1-5 给出了一个 CMOS 二输入端与非门符号法版图设计的例子。它对应的物理版图参见第 2 章的图 2-5。

符号法版图设计方法保持了交互图形编辑方法有较高布图密度和灵活性的优点。且由于设计规则是由符号法版图编辑器维持的,用户在操作时不需要考虑,因而较大幅度降低了设计工作量。但由于目前还没有高性能的二维压缩工具,布图密度低于手工设计的交互图形编辑方法。MAGIC 是一个著名的符号法版图编辑器。它是由美国加州大学 BERKELEY 分校开发的。它除了有编辑功能和压缩功能外,还提供了自动布线工具。有些实用系统把符号法和交互图形编辑集成在一个系统中,如我国的 PANDA 系统。用户可以用几何图形,也可以用器件符号进行版图编辑。

## 3. 积木块自动布图

积木块自动布图(Building Block Layout)又叫做任意形状单元布图,简称 BBL。BBL 布图模式的版图如图 1-6 所示。限于实现的困难,目前的 BBL 模式中的模块都为矩形。但它们可被安置在芯片的任何位置上。模块版图可以是手工设计,也可以是用其他设计方法,如门阵列、标准单元或者是 BBL 方法本身设计的。如果是用其他自动设计方法设计的模块

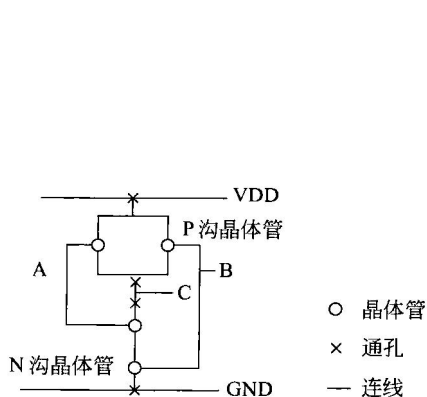


图 1-5 符号法版图例子

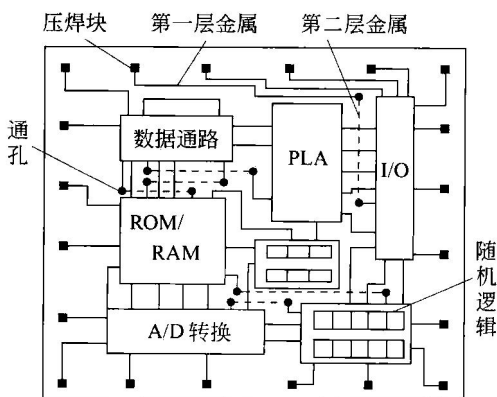


图 1-6 BBL 版图结构