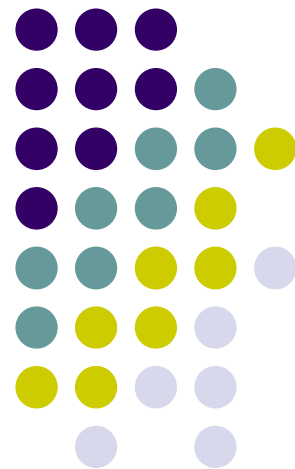


超大规模集成电路基础

Fundamental of VLSI

第十一章 芯片设计流程





数字IC设计流程

1

定义需求

制定芯片的具体指标

2

系统设计

用系统建模语言对各个模块描述

3

前端设计

RTL设计、RTL仿真、硬件原型验证、电路综合

4

后端设计

版图设计、物理验证、后仿真等



具体指标

物理指标

- 制作工艺
- 裸片面积
- 封装

性能指标

- 速度
- 功耗

功能指标

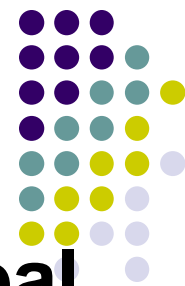
- 功能描述
- 接口定义

芯片设计的产业链的六个环节



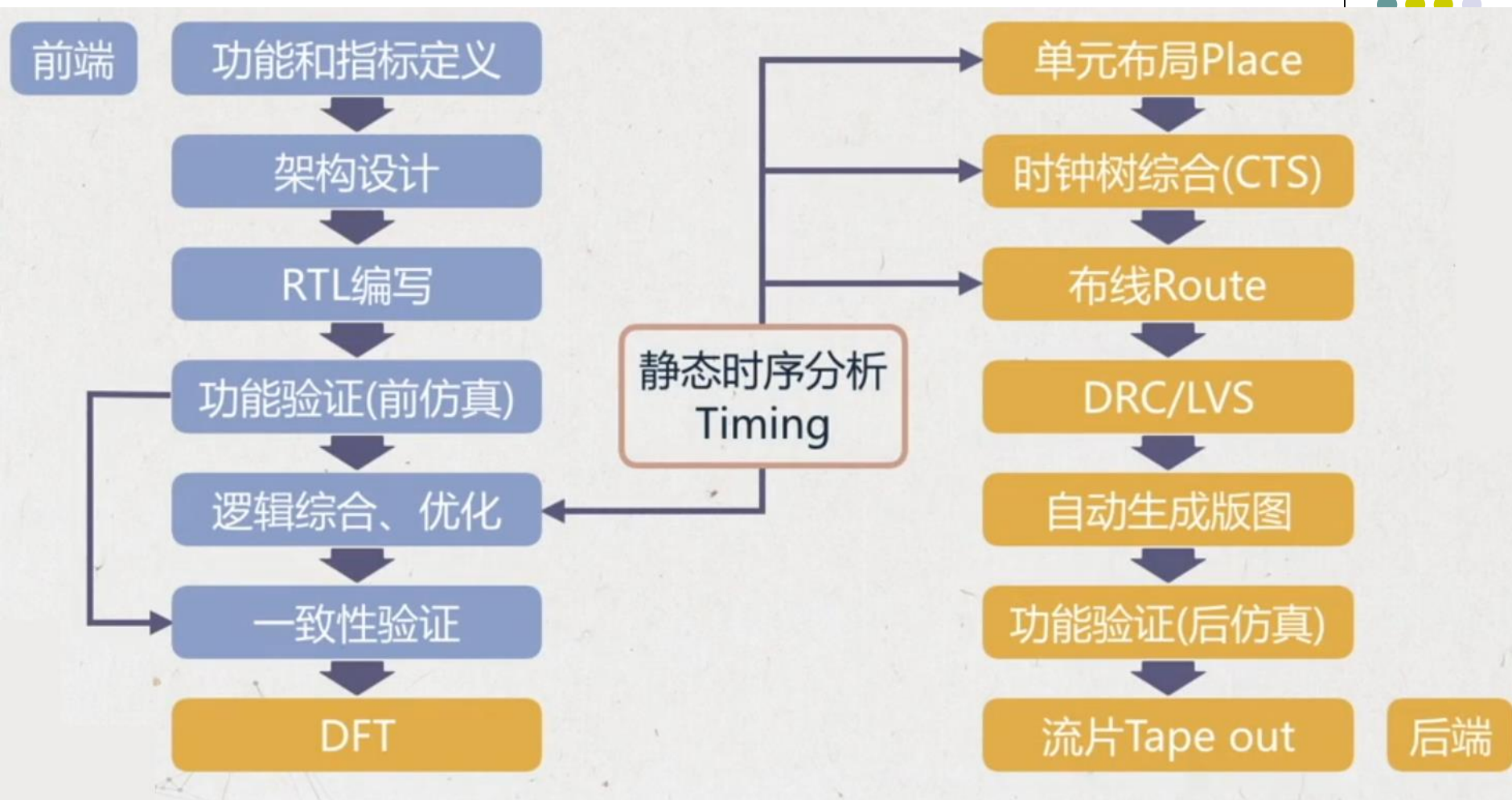
- 晶圆厂**Foundry**
- 无晶圆设计公司**Fabless**（只专注设计，没有制造业务的公司或者未拥有芯片制造工厂的**IC**设计公司）
- **EDA** 软件公司（自动化软件生产厂商，为集成电路设计提供软件支持）
- 设计服务软件公司**Design Service**（主要提供芯片后端设计服务）
- **IP**供应商（**IP**研发成本高，可复用性强）
- 集成设计与制造**IDM**

芯片设计的产业链的六个环节

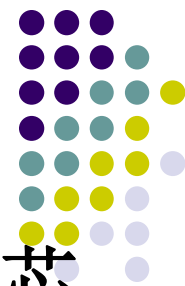


- **Foundry** 的公司代表有：**TSMC、Global Foundry、UMC、SMIC**
- **Fabless**的公司代表有：博通、高通、英伟达
EDA 公司三巨头：**Cadence、Synopsys、Mentor Graphics**
- **Design Service** 公司代表有：**VeriSilicon、alchip**等
- **IP** 供应商公司代表有：**RISC**的**IP**提供商**ARM**
- **IDM** 公司代表有：英特尔

数字芯片设计流程



功能定义指标



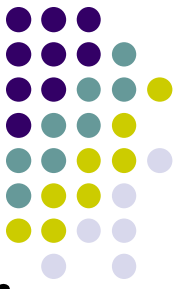
- 芯片规格，也就像功能列表一样，是客户向芯片设计公司（称为**Fabless**，无晶圆设计公司）提出的设计要求，包括芯片需要达到的具体功能和性能方面的要求。

算法或硬件架构设计与分析



- 完成芯片中数字部分的高层次算法或架构的分析与建模，为硬件提供一个正确的软件功能模型，更为重要的是，通过大量的高层次仿真和调试，为**RTL**实现提供总体性的设计指导。
- 工具：**MATLAB、C++、C、System C、System Verilog** 等。
- 特点：这部分工作至关重要，基本上奠定了整个芯片的性能和功耗的基础。这部分工作主要由具有通信、信号处理、计算机、软件专业背景的工程师完成，也有很多微电子专业背景的工程师参与。

RTL实现



- 使用硬件描述语言（**VHDL**，**Verilog HDL**，业界公司一般都是使用后者）将模块功能以代码来描述实现，也就是将实际的硬件电路功能通过**HDL**语言描述出来，形成**RTL**（寄存器传输级）代码。
- 工具：**Vim**、**Emac**。
- 特点：在书写 **Verilog HDL**、描述逻辑功能的同时，还要兼顾逻辑综合、**STA**、**P&R**、**DFX**、功耗分析等多方面因素。

仿真验证



- 检验编码设计的正确性，检验的标准就是第一步制定的规格。看设计是否精确地满足了规格中的所有要求。规格是设计正确与否的黄金标准，一切违反，不符合规格要求的，就需要重新修改设计和编码。设计和仿真验证是反复迭代的过程，直到验证结果显示完全符合规格标准。
- 语言：**Verilog**、**VHDL**
- 工具：**Mentor**公司的**Modelsim**、**Synopsys**公司的**VCS**、**Cadence**公司的**NC-Verilog**、**Verilog—XL**

逻辑综合



- 将 **RTL** 代码映射为与工艺库相关的网表。
- 工具：
 - Synopsys的**Design Compiler**、**Behavioral Compiler**、**DC-Expert**
 - Cadence的**buildgates**、**Envisia Ambit®**
 - Mentor的**Leonardo**
- 特点：从芯片生产的角度来看，在该步骤之前，所有的工作都可近似看做一个理想的，与现实无关。而从逻辑综合以及后续所有的工作都将与工艺的物理特性、电特性等息息相关。逻辑综合工具的功能主要是将**VerilogHDL**格式的文本映射为网表格式的文本，因此，它的功能等同于文本编译器。

逻辑综合



- 特点：逻辑综合过程中，整个文本格式的编译过程是在给定的人为约束条件下进行的，通过这些约束和设定的目标来指导工具完成 **Compiler** 的工作。所以，逻辑综合过程可以看成是一个多目标（频率、面积、功耗）多约束的工程优化问题。该步骤中，通常会插入 **DFT**、**clock gating** 等。该步骤中通常加入 **Memory**、各种 **IP** 等。为了在各种工艺库以及 **FPGA** 原型验证平台之间有一个更方便的移植，注意适当处理这些 **Memory**、**IP** 等的接口。该步骤中也可加入 **I/O**、**PLL** 等。

STA静态时序分析



- 从静态分析的角度，保证设计中所有的路径，满足内部时序单元对建立时间和保持时间的要求。即无论起点是什么，信号都可以被及时地传递到该路径的终点，并且在电路正常工作所必需的时间段内保持恒定。
- 工具：
 - **Synopsys的Prime Time。**
 - **Cadence的Pearl、Tempus**
 - **Mentor的SST Velocity**
- 特点：从逻辑综合开始，基本上每做一步大的调整，都会完成一次 **STA** 分析，以保证每步都能实现时序收敛。

静态时序分析



- 特点： **PrimeTime** 成为了 **Signoff** 的重要工具。所用到的 **SDC** 同逻辑综合；通常设计中会存在大量的违例路径， **STA** 要修大量的 **setup**、**hold** 等，如何修这些违例，可以体现工作经验的重要性。此外，如果是前端修 **timing** 违例，一般会修的很快，但是会带来一个重大的问题，代码被前端修改后是否存在新的 **bug**，还需重新仿真确认，仿真会消耗掉数以月计的时间，所以除非万不得已，不会找前端修 **timing**。

一致性验证



- **RTL** 代码和逻辑综合后的网表都可以抽象为两幅由节点和边构成的图，一致性验证阶段采用了类似于直接比较两幅图是否一致的方法，来确定逻辑综合生成的网表是否正确。
- 工具：**Formality**、**Conformal**
- 输入文件：**RTL** 代码、**netlist**（逻辑综合输出）
- 输出文件：**match**（两张图节点是否一致）和**verify**（计算得出两张图功能是否一致）的报告。
- 前端设计的结果就是得到了芯片的门级网表电路。

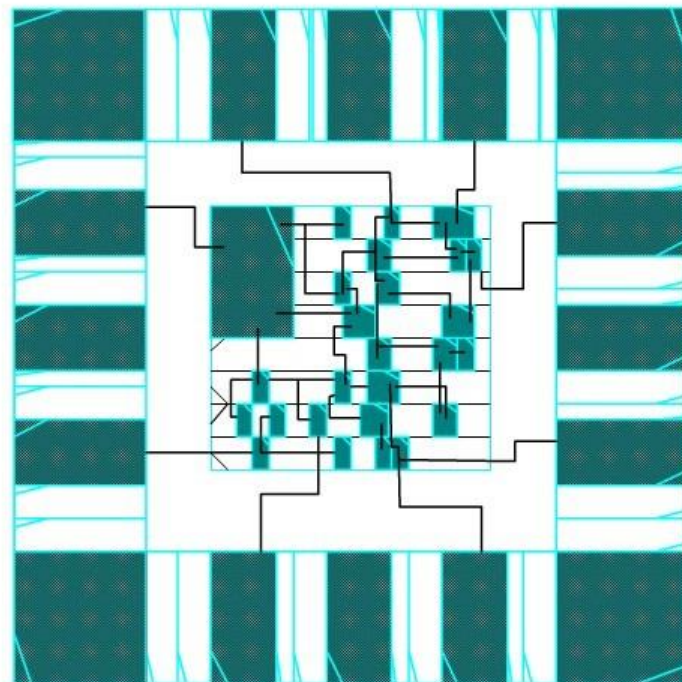
Place And Route



布局布线主要是通过**EDA工具**来完成的

- 芯片布图 (RAM, ROM等的摆放、芯片电源网络配置、I/O PAD摆放)
- 标准单元的布局
- 时钟树综合
- 布线

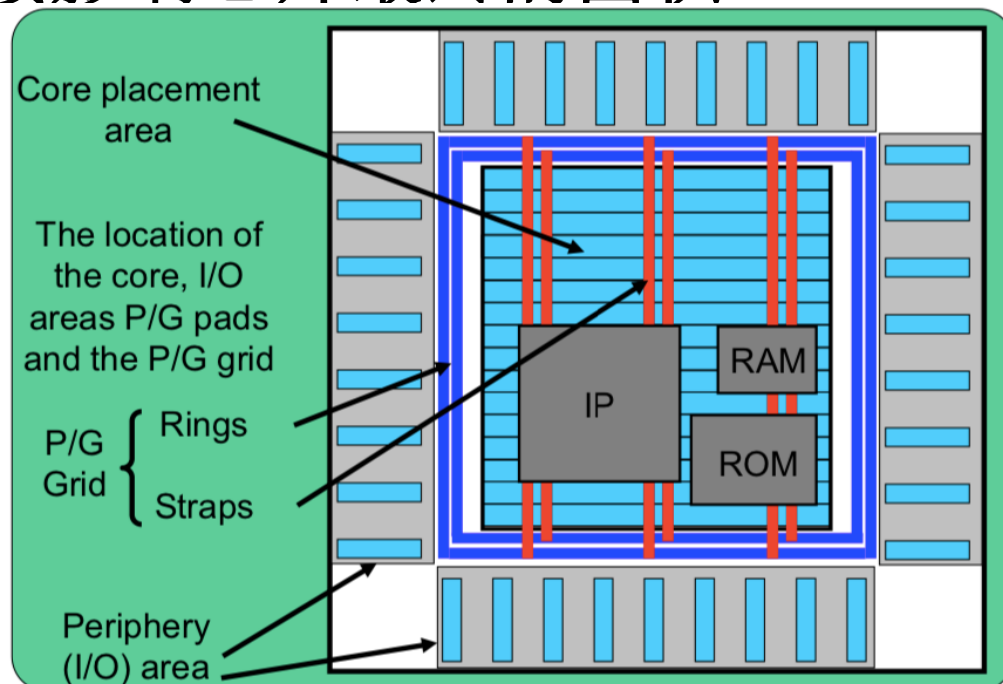
Routing



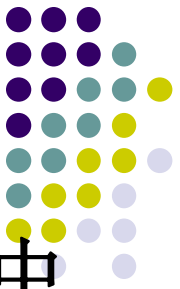
Floor Plan布局规划



- 将 **Design** 导入后端工具后，检查输入文件是否缺少或者有错误。确保 **Design** 正常 **initial** 后就可以进行调整 **Floor Plan**。**Floor Plan** 主要目的是要确定 **design** 的形状大小、**IO Pin** 的位置、所有**macro**的摆放，布局规划完成芯片的总体规划，将直接影响芯片最终的面积。



Placement布局

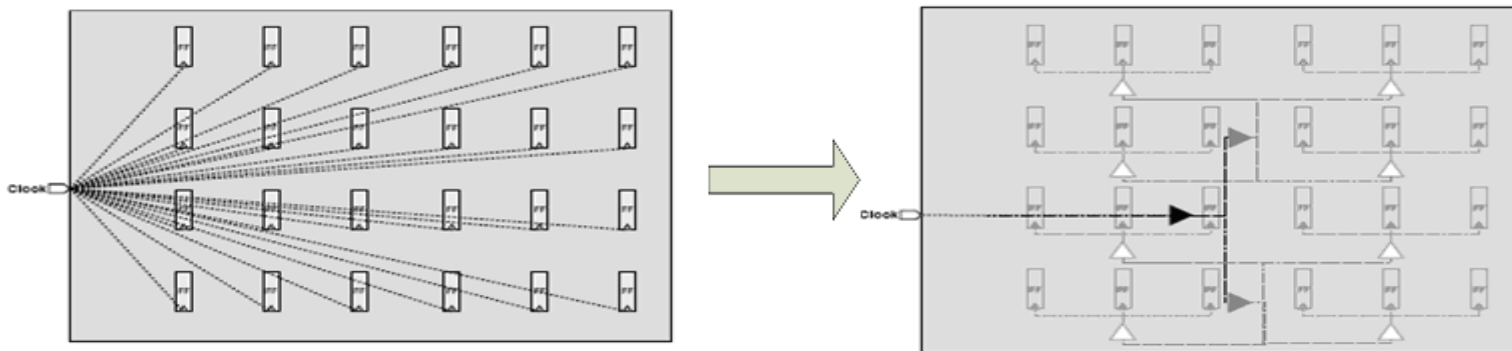


- 用**EDA**工具将电路中各个基本单元在芯片中进行自动摆放
- 工具：
 - **Synopsys**的**Astro**、**Physical Compiler**、**IC Compiler**
 - **Cadence** 的 **Encounter** 、 **PKS** 、 **Silicon Ensemble**、**Design Planner**
- 整个**placement**过程基于**Timing**和**Congestion**及**Power Optimization**进行，不同的条件下**placement**的结果和质量，也是不一样的。**placement**的好坏，直接决定了时钟树的质量和布线，最终体现为影响芯片的性能

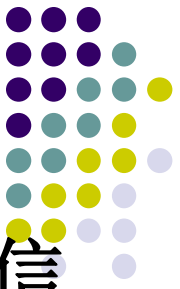
CTS时钟树综合



- 芯片中的时钟网络要驱动电路中所有的时序单元，所以时钟源端门单元带载很多，其负载延时很大并且不平衡，需要插入缓冲器减小负载和平衡延时,时钟信号线需要单独布线。
- 工具：
 - Synopsys的Clock Tree Compiler
 - Cadence的CT-Gen
- 时钟树综合的目的：低skew、低clock latency

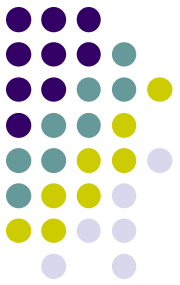


Routing布线



- **CTS** 之后整个芯片的大体结构已定。要将信号线通过金属连接起来,布线就是将前端提供的网表(**netlist**), 实现成版图(**layout**)。布线过程主要完成以下几个目标:
 - 布线过程中会考虑 **DRC** 和 **LVS**, 布线后不会有 **DRC/LVS violations**
 - 布线过程中不会导致 **timing** 变差, 也不会引入新的信号完整性问题。
 - 考虑 **DFM** (**Design for manufacturability**), 例如: **multi-cut via**, 线宽和线间距。

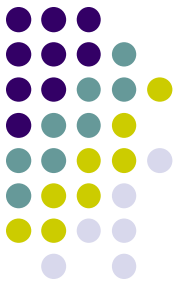
DRC (Design Rule Check)



- Design Rule: 由于制造工艺与电路性能等原因，对版图设计有一定要求，比如说，线宽不能低于最低线宽，N阱间应当具有一定间距，每一层金属应当具有一定密度等。



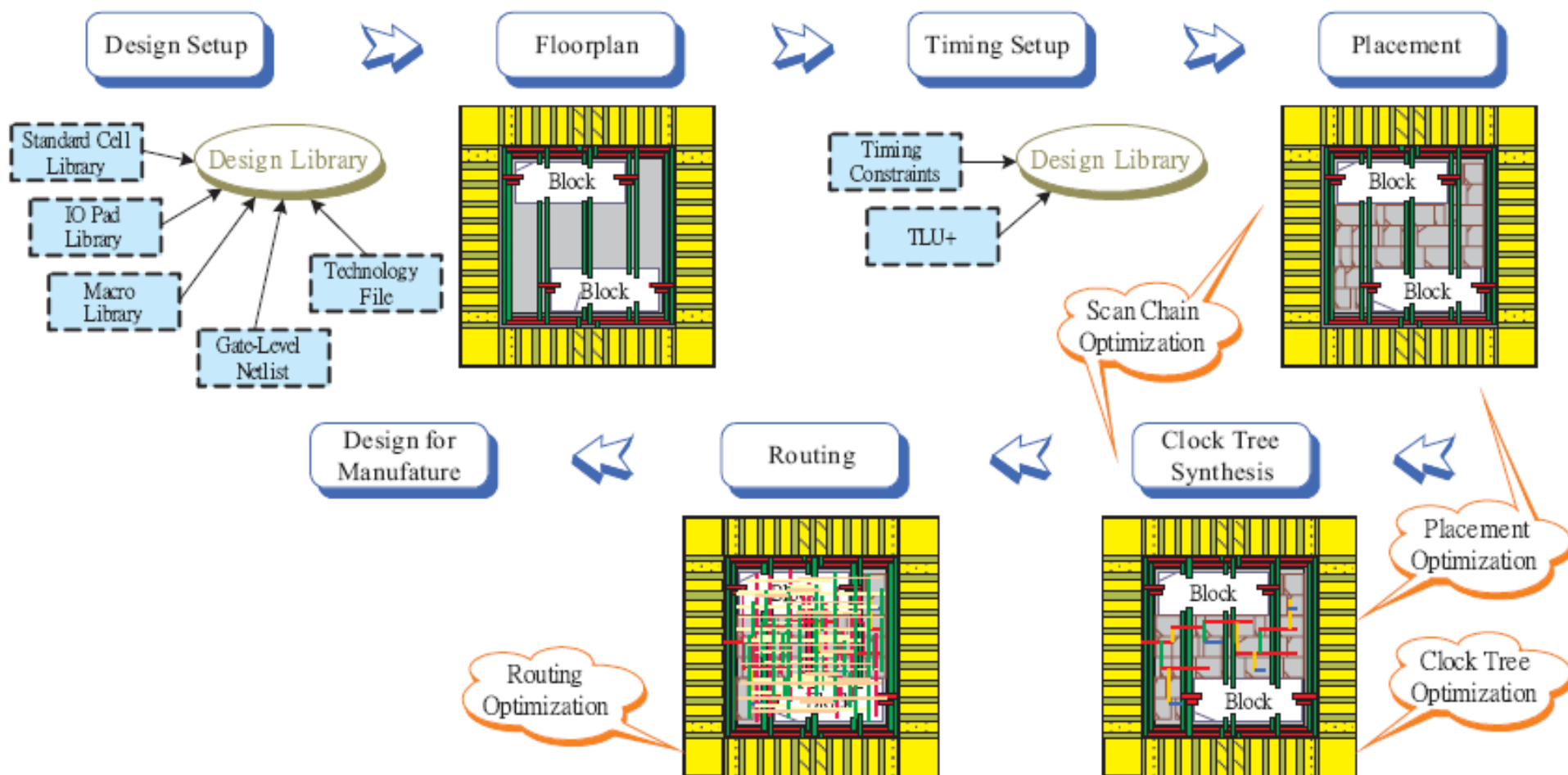
LVS (layout versus schematic)



- LVS: 检查版图文件功能与原有电路设计功能的一致性。LVS软件根据标准单元库设计者提供的cdl网表文件从版图中提取电路网表。



布局布线流程



设计库



- 库是物理版图、电路图、时序和功能模型、晶体管电路描述的集合。
- 标准单元

包括反相器、与门、寄存器、选择器、全加器等多种基本单元，每一个标准单元对应着多个不同尺寸(W/L)、不同驱动能力的单元电路，而且不同驱动强度电路都是基本尺寸或最小尺寸的整倍数。单元库的多样性可以有效提高综合工具和自动布局布线工具的效率，同时也使得设计者可以更加自由地在性能、面积、功耗和成本之间进行优化。

设计库



- 工艺文件
定义了布局布线的规则
- 宏单元
ROM、RAM、IP Core
- IO Pad单元
IO、Pad
- Gate-Level Netlist
前端综合的网标

设计库



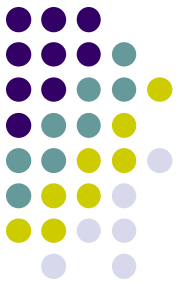
- **Timing Constraints 时序约束**

时序约束信息包括与电路延时相关信息，包括：
建立时间、保持时间、时钟的约束（寄存器-寄存器之间的路径约束）、输入延时的约束、输出延时的约束、组合逻辑的约束

- **TLU+寄生RC查找表**

ICC使用网表的几何形状以及该文件来计算金属线的寄生参数的，包括电阻、电容、耦合电容等

DFM (Design For Manufacturing)



- DFM：可制造性设计
- DFM步骤在整个布局布线流程以后开始，主要目的是通过一些技术处理防止芯片在物理制造过程中出现问题，造成芯片不能工作。DFM的目的在于提高良率。

DFM主要考虑以下效应：

- 天线效应
- Metal liftoff效应
- Metal over-etching效应