

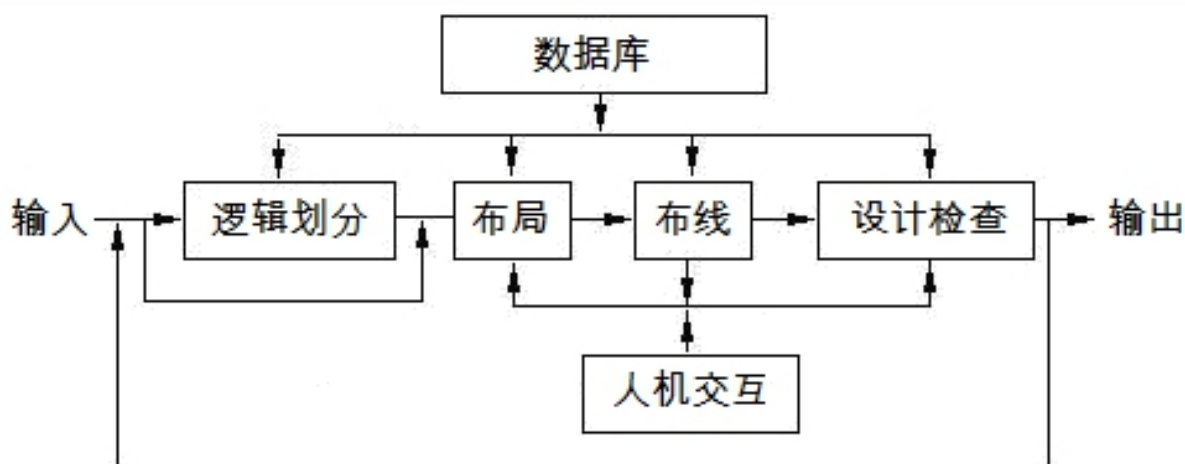
第五章 版图设计



- 集成电路版图设计 (layout) 其实际为电路物理实现的设计，又称为物理设计 (physical design)。版图设计的任务是将电路的逻辑描述形式转化为版图描述形式，将这种版图描述用于图形发生器即可产生生产芯片所需的掩膜 (MASK) 板，并通过MASK光刻实现版图到集成电路chip的物理转化。
- 以往人工设计版图的周期长、错误多、费用大。现在大多采用自动版图设计 (automatic layout) 技术，所以把物理设计也称为自动布图设计。
- 版图设计是一个组合规划问题，版图设计的约束条件、目标函数与设计方式、物理实现有直接的关联。
- 自动布图设计采用分级处理的方式，（布图或称逻辑划分Partition）是将电路按功能块进行逐级分级，直到便于设计；然后将划分后的电路子块以某种方式进行排列（布局placement），最后对排成的电路子块进行连线（布线Routing）；



物理设计流程框图



版图设计属于后端设计部分。

集成电路版图设计就是指将电路设计电路图或电路描述语言映射到物理描述层面，从而可以将设计好的电路映射到晶圆上生产。

版图是包含集成电路的器件类型，器件尺寸，器件之间的相对位置以及各个器件之间的连接关系等相关物理信息的图形，这些图形由位于不同绘图层上的图形构成。



从自动化程度划分版图设计方法：

1.全自动版图设计

利用计算机辅助设计工具以及电路的门级网表自动生成版图的设计方法。

2.半自动设计

在计算机上利用符号进行版图输入，符号代表不同层版图信息，再通过自动转换程序将符号转换成版图。

3.人工设计

主要应用在模拟电路版图，版图单元库设计，全定制数字集成电路设计中。

从版图设计类型分为：

1.标准版图设计

2.半定制版图设计

3.全定制版图设计

标准版图设计通常用于数字集成电路的标准单元库、输入输出单元库等。存储器的版图设计属于半定制版图设计，它的存储单元（例如RAM cell）的版图采用标准单元库的设计方法，其余部分则为不规则的版图设计。模拟与混合信号（analog mixed-signal, AMS）的版图设计以及射频电路的版图设计则属于全定制的版图设计。



常用文件格式

• CIF

caltech intermediate format, caltech中介格式。是另一种基本文本的掩模描述语言。

• GDSII

它是用来描述掩模几何图形的事实标准，是二进制格式，内容包括层和几何图形的基本组成。

• LEF

library exchange format, 库交换格式。它是描述库单元的物理属性，包括端口位置、层定义和通孔定义。它抽象了单元的底层几何细节，提供了足够的信息，以便允许布线器在不对内部单元约束来进行修订的基础上进行单元连接。

包含了工艺的技术信息，如布线的层数、最小的线宽、线与线之间的最小距离以及每个被选用cell, BLOCK, PAD的大小和pin的实际位置。cell, PAD的这些信息由厂家提供的LEF文件给出，自己定制的BLOCK的LEF文件描述经ABSTRACT后生成，只要把这两个LEF文件整合起来就可以了。

• DEF

design exchange format, 设计交换格式。它描述的是实际的设计，对库单元及它们的位置和连接关系进行了列表，使用DEF来在不同的设计系统间传递设计，同时又可以保持设计的内容不变。DEF与只传递几何信息的GDSII不一样。它还给出了器件的物理位置关系和时序限制等信息。



常用文件格式

• SDF

Standard delay format, 叫标准延时格式。是IEEE标准, 它描述设计中的时序信息, 指明了模块管脚和管脚之间的延迟、时钟到数据的延迟和内部连接延迟。

• TLF

Timing Library Format, TLF文件是描述cell时序的文件, 标准单元的rise time, hold time, fall time都在TLF内定义。时序分析时就调用TLF文件, 根据cell的输入信号强度和cell的负载来计算cell的各种时序信息。

• ALF

Advanced library format, 叫先进库格式。是一种用于描述基本库单元的格式。它包含电性能参数。

• PDEF

Physical design exchange format, 物理设计交换格式。它是SYNOPSYS公司用在前端和后端工具之间传递信息的文件格式。描述了与单元层次分组相关的互连信息。这种文件格式只有在使用SYNOPSYS公司的Physical Compiler工具才会用到, 而且.13以下工艺基本都会用到该工具。

• SPEF

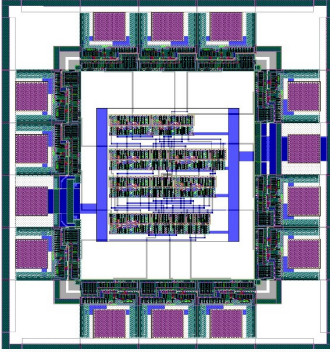
Standard parasitic exchange format, 叫标准寄生交换格式, 属于IEEE国际标准文件格式。从网表中提取出来的表示RC值信息, 是在提取工具与时序验证工具之间传递RC信息的文件格式。



- 两种主要的版图设计方式。
- 一种是几乎所有部件的高度都相等的标准化方式，如多单元（Polycell）或标准单元（Standard cell）方式，常通过自动布图设计系统的版图，如SPR（Standard Placement Routing）等实现；
- 另一种是针对各种不同大小的部件的方式，狭义来说就是积木块方式（Building Block）。这种方式允许存在像RAM、ROM、移位寄存器等规模很大的部件或子模块，因而适用的范围也较广，采用BBL（Building Block Layout）等方式实现，如BPR（Block Placement Routing）。

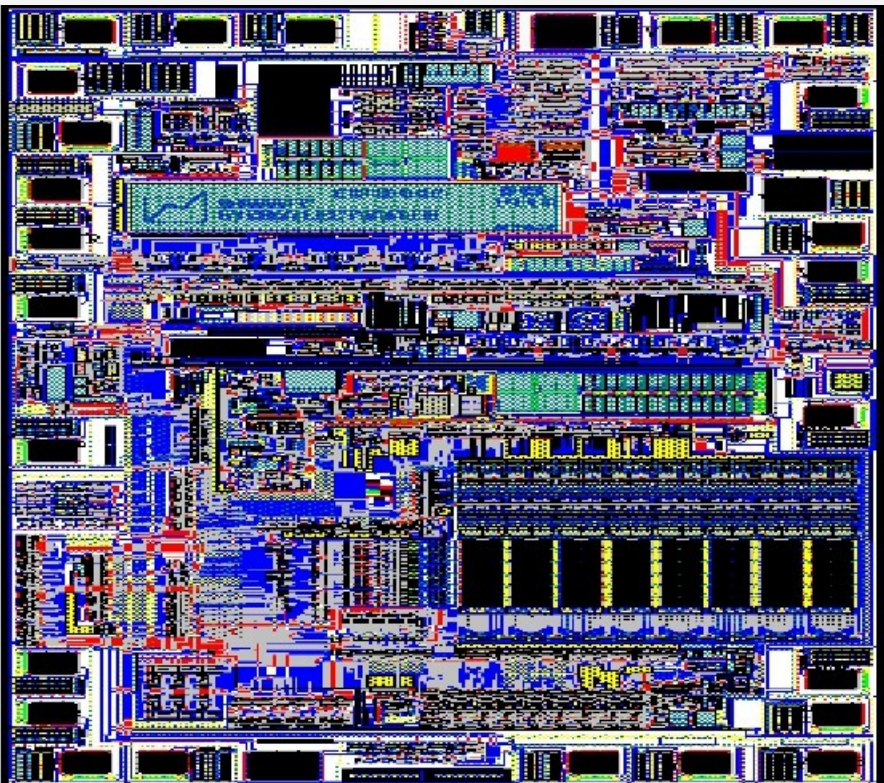


SPR和 BPR



Tanner Pro, SPR是利用stand cells和pad cells自动完成整个芯片的layout。

- 另外还有一个区块配置与绕线（BPR）工具用做区块的布局绕线，同样是依靠标准组件库进行工作，过程同SPR类似。通过BPR可以自由的定义版图布局与绕线，不会像SPR全部自动完成，提高了版图的利用率和芯片性能。



布图的分级

- Top down的布图设计一般都是分级设计，布图规划是是一个软件的划分过程，主要针对软模块（网表）；而布局是针对全部硬模块，并且是Bottom up的布图设计，它可以是分级设计，也可以不是分级设计。
- 在分级设计中，芯片由各级模块组成。芯片为最高一级模块，高一级模块由若干个低一级模块组成。最低一级的是基本单元。
- 一旦模块设计完成，有了具体的物理版图，它就是一个硬模块，也称为IP硬核（Hard core）。在分级设计过程中，待设计的模块也可为软模块，即软核（Soft core）。最低一级的是基本单元，可以是硬的，也可以是软的。硬核与具体Foundry的制造工艺结合紧密，软核在验证正确后，可以形成固核（Custom core），也可以形成多种的硬核。



§1.逻辑划分

- 布图规划过程与所用的算法和求解策略有关。一个典型的布图规划过程一般包括：Bottom up结群并产生结群树、Top down软模块布局、调整模块形状和确定模块的引线位置，目标是使芯片面积、总连线长度最小和优化输出结果。
- 布图规划分为物理分级构造、分级布图规划和详细布图构造三大部分，每个部分又分成若干过程。物理分级构造是一个Bottom up结群和估计模块面积的过程；分级布图规划则是个Top down软模块布局、布线区面积估计和分配、模块形状调整以及布局修正的过程；随后在详细布图构造中完成整个布局和布线。
- 无论用哪一种算法，布图规划过程一定是Top down和Bottom up多次交替迭代的过程。



- 布图规划的输入是一个有层次的网表，这个网表描述了功能方块（如RAM，ROM，ALU等），方块内的逻辑单元（如NAND，NOR，触发器等）和逻辑单元的接线端子。
- 网表是ASIC的逻辑描述；平面布局是ASIC的物理描述；因此，布图规划就是逻辑描述（网表）和物理描述（平面布局）之间的映射。



划分要求

- 逻辑划分的原则是：
 - (1) 逻辑功能块的功能完整性
 - (2) 连线尽量在模块内实现
- 每个模块完成一定的功能，即模块对应的子电路在电学上、逻辑上是相对独立的。
- 使每个模块的内部均有较强的连接关系，同时使模块间有尽量弱的连接关系。
- 一个电路系统 A_S ，它可以划分为 m 个子块， A_1, A_2, \dots, A_m ，满足条件：

$$\begin{cases} A_i \cap A_j = \Phi & i \neq j \quad i, j \in \{1, 2, 3, \dots, m\} \\ \bigcup_{i=1}^m A_i = A_S & \text{即 } A_1 \cup A_2 \cup \dots \cup A_m = A_S \end{cases}$$

- 一个划分出的子电路 A_i ，有对应的面积 $S(A_i)$ 及端子数 $E(A_i)$ 。每一个划分有一定的约束条件，即每个子电路的最大面积 S_{\max} 和最大端子数 E_{\max} ，所有划分要符合：



$$\begin{cases} S(A_i) \leq S_{\max} \\ E(A_i) \leq E_{\max} \end{cases}$$

简单连接度法

- 简单连接度法既是一个 Bottom up 的结群算法，也是一种最大结合-最小不结合在完成核心单元及其模块划分后的局部划分方法。
- 若用 $B(A_i)$ 表示单元 A_i 所有的外部连接的集合，则两个单元 A_i, A_j 之间的连接度（即连接数）为：

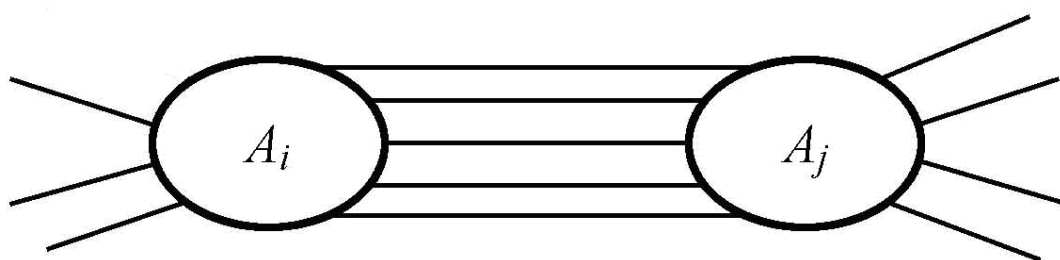
$$Con(A_i, A_j) = |B(A_i) \cap B(A_j)|$$

- A_i, A_j 之间的分离度（即无关连线之和）为：

$$Dis(A_i, A_j) = |B(A_i) \cup B(A_j) - Con(A_i, A_j)|$$



连接度和分离度的关系



$$\text{Con} (A_i, A_j) = 5$$

$$\text{Dis} (A_i, A_j) = 7$$

$$B(A_i) = 8$$

$$B(A_j) = 9$$

$$\text{Con}(A_i, A_j) = |B(A_i) \cap B(A_j)| = 5$$

$$B(A_i) \cup B(A_j) = 13$$

$$\begin{aligned} \text{Dis}(A_i, A_j) &= |B(A_i) \cup B(A_j) - \text{Con}(A_i, A_j)| \\ &= |13 - 5| = 7 \end{aligned}$$



简单连接度法

(鼠标滑过播放视频)

简单连接度法的算法过程

- (1) 任何一个电路都是由若干单元构成，在待划分的单元集A中取出一个单元a， $i=1$ ，第一次划分选取原则是a1具有与其它单元最少连线的单元，表示为 $Con(a1, A-a1)$ 最小，放入划分集 A_i 中， A_i 由 ϕ 变成 $\{a1\}$ ，A变成 $A - \{a1\}$ 。
- (2) 在A中取出一个单元a2，使 $Con(A_i, a2)$ 为最大，即最相关。如果存在两个以上连接度相同的单元，则选 $Dis(A_i, a2)$ 最小的单元，即a2与其它单元有弱的连接关系。
- (3) 检查
$$\begin{cases} S(A_i) \leq S_{max} \\ E(A_i) \leq E_{max} \end{cases},$$
 是否符合给出的面积和出线端的限制条件。
如满足条件则
$$\begin{cases} A_i = A_i + \{a2\} \\ A = A - \{a2\} \end{cases}$$

转(2)继续执行；若不满足条件，则转(4)。

简单连接度法的算法过程

(1) 任何一个电路都是由若干单元构成，在待划分的单元集 A 中取出一个单元 a ， $i=1$ ，第一次划分选取原则是 a_1 具有与其它单元最少连线的单元，表示为 $\text{Con}(a_1, A-a_1)$ 最小，放入划分集 A_i 中， A_i 由 ϕ 变成 $\{a_1\}$ ， A 变成 $A - \{a_1\}$ 。

(2) 在 A 中取出一个单元 a_2 ，使 $\text{Con}(A_i, a_2)$ 为最大，即最相关。如果存在两个以上连接度相同的单元，则选 $\text{Dis}(A_i, a_2)$ 最小的单元，即 a_2 与其它单元有弱的连接关系。

(3) 检查
$$\begin{cases} S(A_i) \leq S_{\max} \\ E(A_i) \leq E_{\max} \end{cases},$$

是否符合给出的面积和出线端的限制条件。

如满足条件则
$$\begin{cases} A_i = A_i + \{a_2\} \\ A = A - \{a_2\} \end{cases}$$

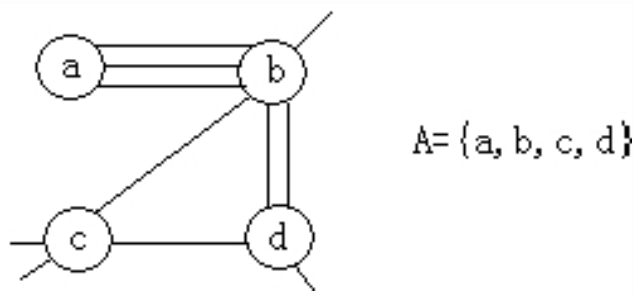
转(2)继续执行；若不满足条件，则转(4)。

(4) 检查 $A=\phi$

若不满足条件， $i=i+1$ ，转(1)产生新的划分集；否则结束划分过程，输出划分结果。



简单连接度划分示例



$A = \{a, b, c, d\}$

解：设有待划分集合 $A = \{a, b, c, d\}$

$\text{Con}(a, A-a) = 3$

$\text{Con}(b, A-b) = 6$

$\text{Con}(c, A-c) = 2$

$\text{Con}(d, A-d) = 3$

(1) $\because \text{Con}(c, A-c)$ 最小，将其选出，则 $A_i = \{c\}$ ， $A = \{a, b, d\}$ ；

(2) 虽然 d 与 b 与 A_i 的连接度相同，但 $\text{Dis}(A_i, d) < \text{Dis}(A_i, b)$ ，选中 d；

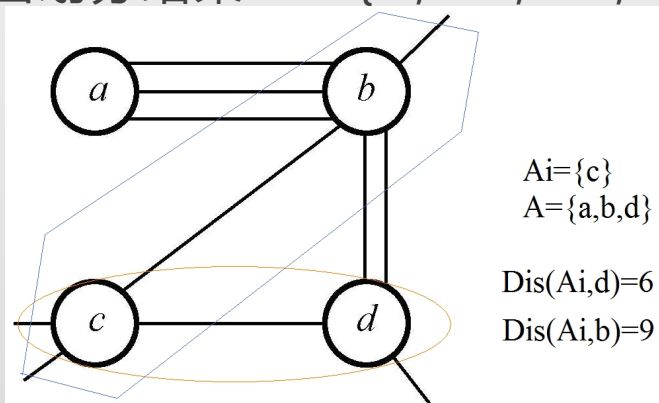
(3) 检查，测定满足条件，即 S、E 均满足条件（面积和边界连线端子的约束）。 $A_i = \{c, d\}$ ， $A = \{a, b\}$ ；

(4) 选 b， $\because \text{Con}(A_i, a) = 0$ ；

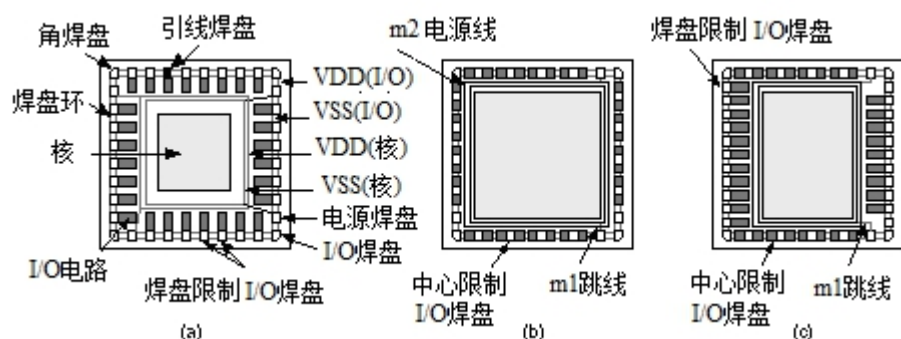
(5) 检查，若满足条件，有 $A_i = \{c, d, b\}$ ， $A = \{a\}$ ；

(6) 选 a 并检查，测定满足条件，则 $A_i = \{c, d, b, a\}$ ， $A = \emptyset$ ；

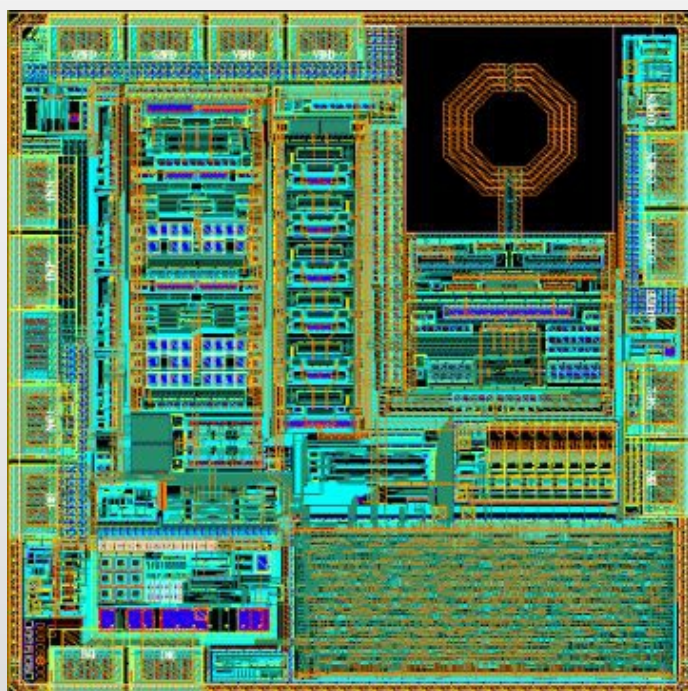
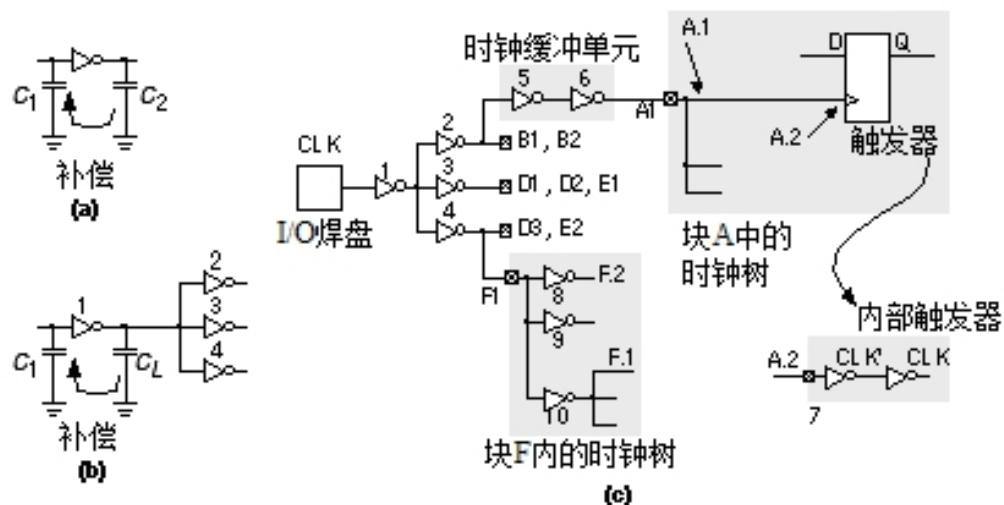
(7) 输出划分结果 $A_i = \{c, d, b, a\}$ 。



I/O和电源规划



时钟规划



§2.布局

VLSI在完成布图规划之后，就可以开始在规划好的方块内进行逻辑单元的布局。

布局比布图规划更适合自动化处理。布局就是要在满足有关约束条件下对已划分后的 n 个组件寻找一个最优放置方案，以使某一个或多个目标函数为最优或较优，这需要制定一定的测量技术和算法规则。

布局一般在一个平面上实现。



布局的目标函数与约束条件

①几何要求：

首先应满足一些严格的设计要求（如在布局前规定一些单元的相对位置是何种结构，如多元胞结构是一种以标准单元为基础的等高等宽结构），其次所有单元在平面上不允许重迭，单元之间要划出通道用于布线，最终所需的面积要尽可能小。

②拓扑要求：

具有强连接（连线较多）的一些单元之间距离要求。如有最长线网长度的限制，则在同一线网中的单元要尽量靠近安置，最终要求总连线长度最短或各通道区的布线分布要均匀，使之有利于布线。经验表明，采用均匀布线通道效果较好。



•理想的布局目标

- 保证100%能完成布线
- 关键线网的最小化延迟
- 使芯片上放置的单元尽可能紧密

•优化的目标

- 最小化能量耗散
- 最小化信号间的交叉

•量化的具体目标

- 总的互连线长度的最小化估计
- 满足关键线网的时钟需要
- 最小化互连线拥挤情况

这些目标是一些具体的评估和测量标准，在某种程度是妥协和折中的结果。

因此在完成布图规划和布局之后，还需要预估块内和块间电容，这样可以更精确的估计每个逻辑单元的驱动电容，从而达到逻辑综合优化。

实际上，何为布局最佳只是个相对的概念。通常的布局需要进行人机交互处理，把人的经验代入进行布局。

一般先进行初始布局，然后进行布局的迭代改善。



- 为了简化布局，将平面划分为方格的组成方式，这些格子用占位符

$$P = \{p_1, p_2, \dots, p_N\}$$

的集合描述，
网表对象

$$M = \{m_1, m_2, \dots, m_r\}$$

映射在占位符上。每一个

$$m_i \in M$$

对象与一组信号 s_i 相关。



距离树

- 为了评估布局的质量，需要能够实施评价的量化指标。布局决定了布线的连线总长度，距离树是一个近似的指标，是一种互连线长度的定量方式，它和互连线的最终长度有密切关系而且容易计算。
- 在两维空间中，两点之间的距离通常用两点间距离的欧几里德距离公式表示，而在集成电路的连线通常是横线和竖线而不采用斜线，因此求距离不能用欧几里德距离公式，而用曼哈顿距离表示。
- 反映了一个线网的所有节点的结构图被称为树，树也是线网中各节点间距离的体现形式之一。

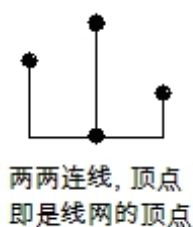
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$d = |x_1 - x_2| + |y_1 - y_2|$$



几种常用的树

- 最小生成树
- 最小斯坦纳树
- 最小链
- 矩形半周长



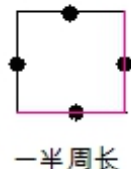
(a)



(b)



(c)



(d)

- 最小生成树
- 其顶点是线网的所有顶点的最小长度树。
- 最小斯坦纳树
- 其顶点是包含线网的所有点的最小长度树，线网本身的连线包含在树内。
- 最小链
- 从源点开始，连接所有的点，除起点和终点的连接度为1之外，其余点连接度为2，并以最小长度连接所有点。
- 矩形半周长
- 包围线网的所有顶点的矩形的半个周长。

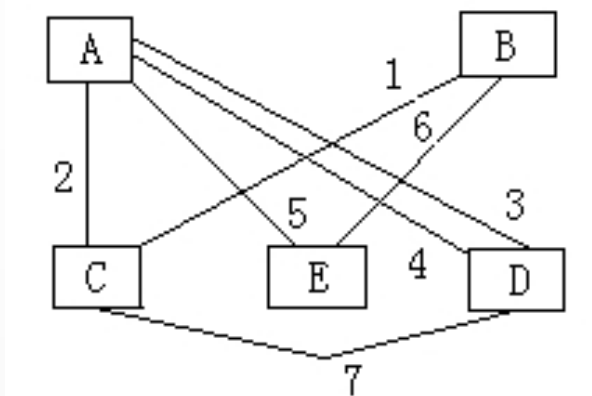


初始布局

- 初始布局就是在满足约束条件的情况下，按照一定的目标，把单元安放在芯片的特定位置上，一般的说，初始布局主要是为了以后的布局迭代改善提供一个较好的初始条件。有些芯片面积冗余量较大，如门阵列电路，其性能无特殊要求，初始布局也可以作为安置单元的一种手段。



单元布局描述



- 其布局描述为：

cell A 2, 3, 4, 5

cell B 1, 6

cell C 1, 2, 7

cell D 3, 4, 7

cell E 5, 6

- 由这个描述立即知道，A与B无连接，A与C只有2号线网连接，A与D为3、4号线网连接，A与E为5号线网连接，B与C为1号线网连接，B与D无连接，B与E为6线网连接，C与E无连接，D与E无连接。

$Cell(A, B) = \{\}$ $Cell(A, C) = \{2\}$

$Cell(A, D) = \{3, 4\}$ $Cell(A, E) = \{5\}$

$Cell(B, C) = \{1\}$ $Cell(B, D) = \{\}$

$Cell(B, E) = \{6\}$ $Cell(C, D) = \{7\}$

$Cell(C, E) = \{\}$ $Cell(D, E) = \{\}$



• 选择规则

主要以未安置单元与已安置单元的连接度为依据。

单元间的连线是确定选择函数的一个重要参数。同时，选择函数与未安置单元和已安置单元的连接状况有关。

• 安置规则

一般总是将待安置单元放在已安置单元的邻域的空位上。这些空位的集合称为候选单元位置集 S_p 。

究竟安放在哪个位置，需要采用上述的四种距离树中的一种来进行估算。



安置规则的布局

17	18	19	20
13	14 ^A	15	16
9	10 ^B	11 ^C	12
5	6	7	8
1	2	3	4

(a)

- 对图 (a) 的布局图，现在已经安置了三个单元A、B、C，下面再安置单元，则一定选择与A、B、C最有关联的单元。设有D单元为需要安置，按安置规则，D单元一定是安放在A、B、C的邻域内，那么候选单元位置集 $S_p = \{6, 7, 9, 12, 13, 15, 18\}$ 。
- 以最小生成树为计算规则进行安置，单元应安置在所有线网总距离为最小的位置上。若D单元与 $\{A, B, C\}$ 有两个线网，一个是与A、B连接，一个是与B、C连接。显然D单元只能安放在 S_p 中的位置上，一共有7个位置。



17	18	19	20
13	14 ^A	15	16
9	10 ^B	11 ^C	12
5	6	7	8
1	2	3	4

(a)

17	18	19	20
13	14 ^A	15	16
9 ^D	10 ^B	11 ^C	12
5	6	7	8
1	2	3	4

(b)

- 以最小生成树为判据：
- ①放在6号位 最小生成树长度=4
- ②放在7号位 最小生成树长度=5
- ③放在9号位 最小生成树长度=4
- ④放在12号位 最小生成树长度=5
- ⑤放在13号位 最小生成树长度=5
- ⑥放在15号位 最小生成树长度=4
- ⑦放在18号位 最小生成树长度=5
- 显然应该将D单元安放在 {6, 9, 15} 位置上，这样才符合最小生成树长度最小的要求。从均匀性讲，如图 (b)所示，D安放在15号位最合理；但通常要尽量将单元靠边放，为以后的其它线网提供方便，所以D应该放在9号位。安放完D单元后，产生新的邻域 $S_p = \{5, 6, 7, 12, 13, 15, 18\}$ ，可以继续安放新的单元。



§3. 布线

- 布线就是在满足工艺规则（如线宽、线间距、布线层数等）和电学规则（如不同线网间的可靠绝缘等）情况下，完成给定的信号线网连接，并使布线结果最佳。
- 通常将布线分成总体布线和详细布线两部分；另外根据对象的不同又可分为面向线网的布线和面向通道的布线；根据布线区域也可以分为门间布线与通道布线。
- 布线的分类是有相关性的，总体布线主要是面向整个线网的布线，而详细布线更多的是指通道布线和通道内的面向线网的布线；一般情况下，除了门间布线是单元（或模块）内部的布线外，其余都是指单元（或模块）外部的布线。



门间布线

- 以行式结构为例，除了有专门的布线通道用于布线外，还有阵列单元的连线与单元阵列中门与门间的连线。门间布线主要指的是非通道区的布线。
- 任何一个自动布图设计系统都必须有一定的单元库的支持，单元库内的单元由三个方面的数据描述：单元的名称和符号；单元的端点描述；单元的版图连接描述。



- 以门阵列的版图结构形式进行介绍。

- ① 符号表示：



- 用这样一个形式描述三输入与非门（NAND3）。

- ② 端点描述：对应NAND3的端点描述可以是NAND3，即

- in: A, B, C out: O

- ③ 版图连接描述：对于门阵列结构，芯片结构已固定。单元、电源、地线的位置也是固定的，因此逻辑单元的描述只涉及除电源、地线以外的其他线网端子的引线连接。

- 由于线网是由横线与竖线组成的，所以基本连接描述就是对这些横线段与竖线段的描述，有如下约定：

- （a）线段描述。横线段由三个数据组成：纵坐标、起点横坐标、终点横坐标。竖线段也由三个数据组成：横坐标、起点纵坐标、终点纵坐标。可见，同样三个数据，区别就是描述横线的第一个数据前有个“-”号。对于一个线网，总是由横线段与竖线段组成。因此一个线网可由多组三位（符号）数据连接而成。

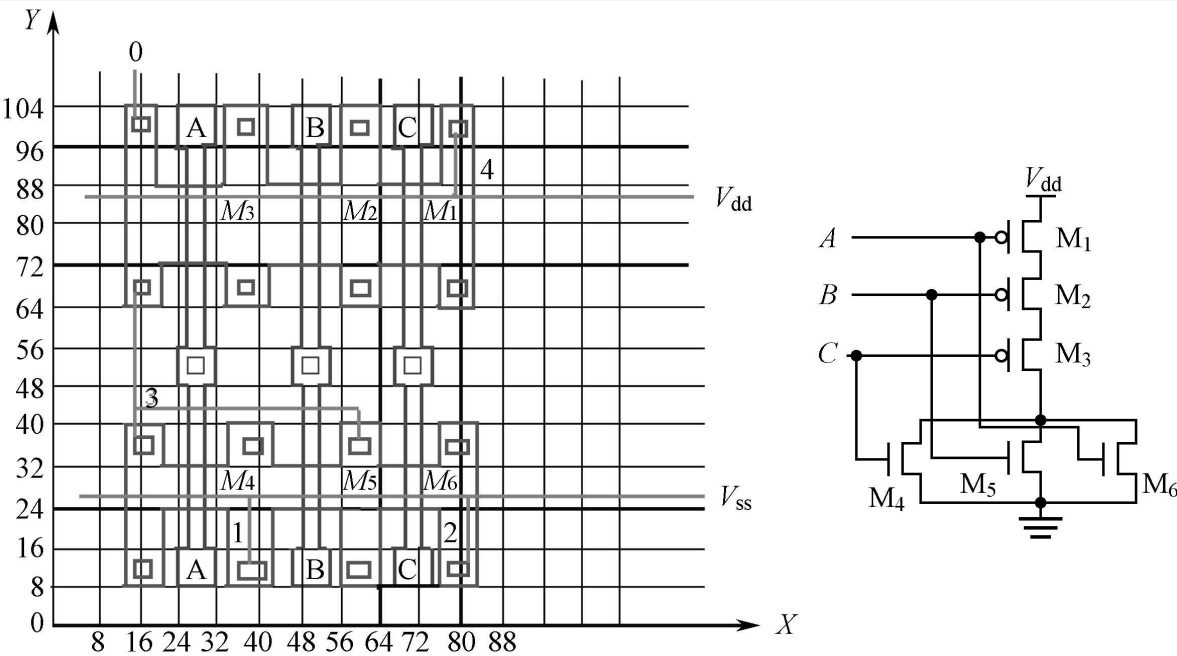
- （b）输入/输出描述的格式为：

- 输入：INPUT 端子号，上出线端坐标，下出线端坐标；

- 输出：OUT 端子号，上出线端坐标，下出线端坐标。

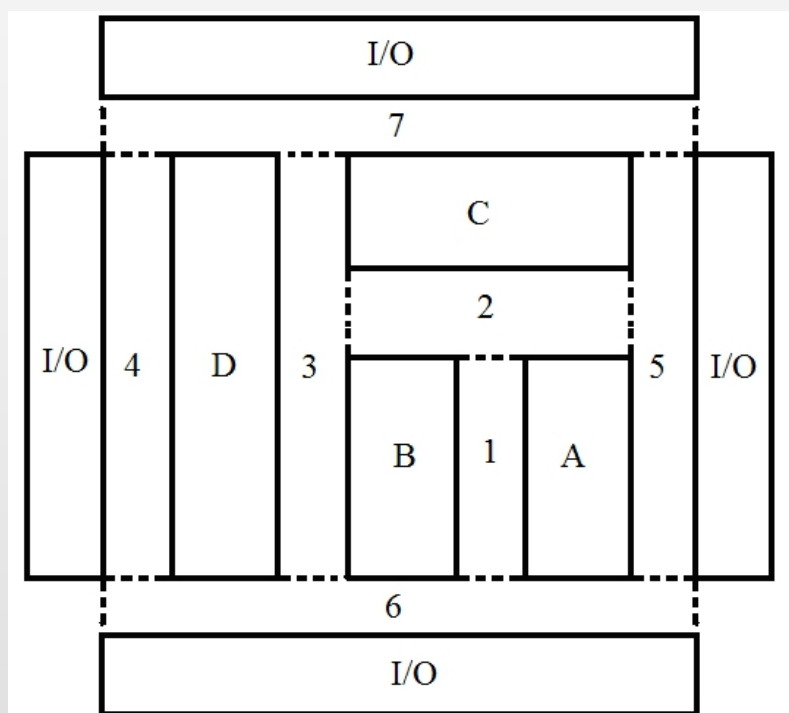


- 端子描述：
 - in: A, B, C
 - out: O
- 版图连接
 - CELL NOR3
 - 38, 12, 28
 - 80, 12, 28
 - 16, 36, 68, 60, 36, 44, -44, 16, 60
 - 80, 84, 100
 - INPUT, A, 28, 28, B, 50, 50, C, 70, 70
 - OUT, O, 16, 16
 - END CELL NOR3



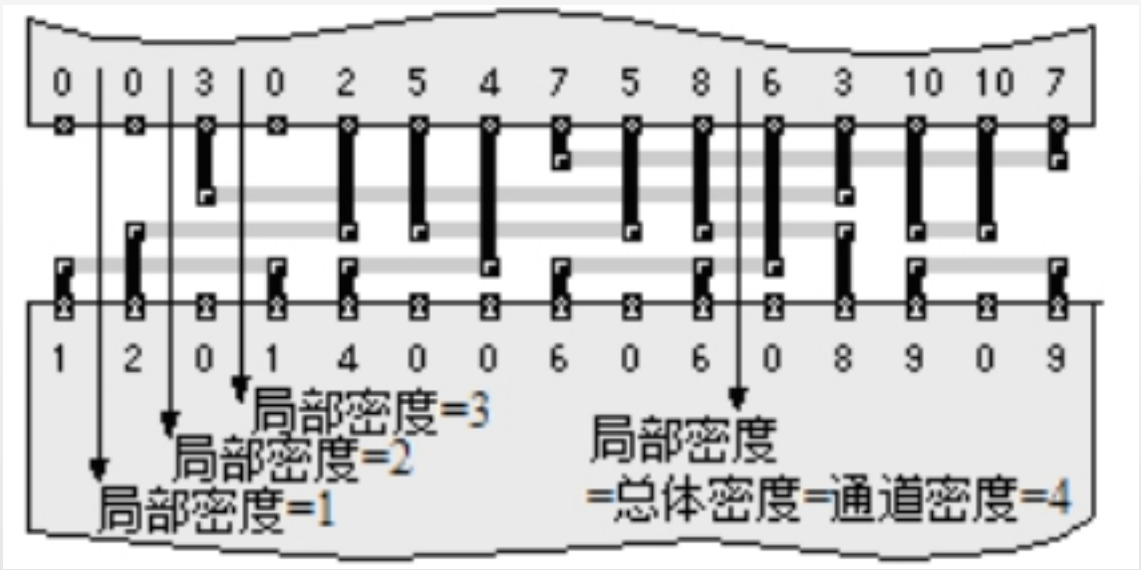
布线区与通道区

- 布线区是一个总体的概念，是指芯片中除电路单元所占区域以外的能进行布线的区域。通道区指的是被电路单元布局分割后出现的各个分离的矩形区域，如图 5.31 所示的 1, 2, 3, 4, 5, 6, 7 的矩形，每个矩形叫做一个布线通道。显然布线区就是这些矩形的集合 $\{1, 2, 3, 4, 5, 6, 7\}$ 。



二边通道布线模型

- 所谓二边通道指的是通道具有二边平行边界的通道。在给定通道的上下边界上等距的分布着各个线网的端子。除0以外的相同端子号代表是同一线网，把各出线端按号加以连接就构成了线网，0号端子表示在该位置是空的引线端。左右两端的引线用对应的端子号标注，其出线的上下位置是自由的，即该端点可以用导线延伸。除固定门阵列外，通道宽度可根据线网的多少调节宽度。



双层布线的严格分层

- 对于二边通道布线，可以看出若用单层布线，势必出现线的交叉。为了防止交叉有两种方法：①用多晶作竖线，用金属作横线。因多晶上的 SiO_2 绝缘，又称为“一层半”布线；②用双层金属布线，两层金属间为绝缘层。其中一层用于实现竖线，另一层用于实现横线，两层金属用通孔连接，这也是曼哈顿布线。



垂直约束图

(VCG - Vertical Constrained Graph)

- 因两边通道排列的是均匀间隔的出线端，因此在同一水平坐标处就有上、下边出现不同线网的引线端。为了防止同列的竖直线重叠，则水平线所处的上下位置关系必须符合一定的约束条件。采用一个有向图来表示这种约束关系，称为垂直约束图VCG。

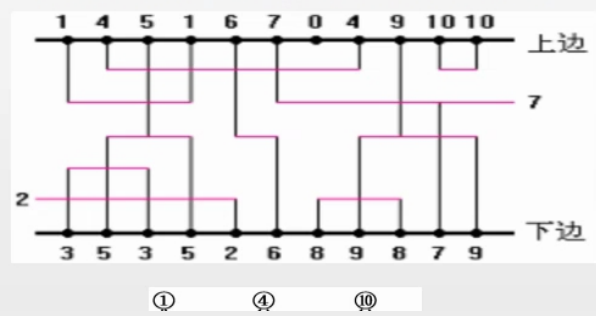


二边通道的垂直约束图

(鼠标滑过播放视频)

二边通道的垂直约束图的画法

- 先取下边没有的端子为VCG第一层，显然是①④⑩线网；
- 找①④⑩下的且同时也在上边有出线端子的项，显然是⑤⑨⑦，作为VCG第二层；
- 再找在⑤⑨⑦下且同时也在上边有出线端子的项，显然是⑥，作为VCG第三层；
- 再找⑥下且同时也在上边有出线端子的项，这时没有符合的线网；
- 上边的端子均填满VCG后，则将下边剩余的线网按关系填入VCG的最下面的一层。



二边通道的垂直约束图的画法

- 先取下边没有的端子为VCG第一层，显然是①④⑩线网；
- 找①④⑩下的且同时也在上边有出线端子的项，显然是⑤⑨⑦，作为VCG第二层；
- 再找在⑤⑨⑦下且同时也在上边有出线端子的项，显然是⑥，作为VCG第三层；
- 再找⑥下且同时也在上边有出线端子的项，这时没有符合的线网；
- 上边的端子均填完VCG后，则将下边剩余的线网按关系填入VCG的最下面的一层。

