

1 CMOS 电路设计方法及 EDA 工具的使用

1.1 概述

集成电路按实现方法可分为全定制集成电路、半定制全定制集成电路。

所谓全定制集成电路，是指按照用户的需求，从晶体管级开始设计，定义芯片上所有晶体管的几何图形和工艺规则，最后将设计结果交由 IC 厂家去进行掩模制造。全定制集成电路一般采用“自底向上”(Bottom-Up)的设计方法。

全定制设计方法的优点是芯片可以获得最优的性能，即面积利用率高、速度快、功耗低，而缺点是开发周期长，费用高，只适合大批量产品开发。

半定制集成电路分为基于门阵列设计法和基于标准单元设计法，这两种方法都是约束性的设计方法，其主要目的就是简化设计，以牺牲芯片性能为代价来缩短开发时间。目前一般应用较多的是基于标准单元的设计方法。半定制集成电路一般采用“自顶向下”(Top-Down)的设计方法。

CMOS 集成电路设计流程可分为逻辑设计和物理设计两个阶段。逻辑设计也称前端设计，这个阶段要根据需求，对于大规模数字集成电路，借助于硬件描述语言和相关 EDA 工具，完成整个逻辑电路的设计和仿真工作；对于基本单元电路或模拟电路，可以直接进行电路设计，然后完成仿真工作。物理设计也称为后端设计。这个阶段是将前端的逻辑设计转化为合格的版图的过程。

1.2 “自底向上”设计方法流程

标准单元和模拟集成电路设计一般采用“自底向上”的设计方法。“自底向上”的设计方法的设计流程如图 1.1 所示。

首先根据用户需求，确定晶体管级电路原理图；然后根据电路原理图，写出电路的 Spice 网表描述，利用晶体管级仿真工具 Spice 对设计的电路进行仿真；如果仿真结果满足的用户要求，接下来根据电路原理图进行版图设计，同时进行设计规则检查(DRC)、电学规则检查(ERC)和版图与电路图一致性检查(LVS)；如果版图设计阶段没有问题，接着进行版图寄生参数提取(LPE)，将提取出的带有寄生电阻、电容参数的网表用 Spice 进行后仿真，如果仿真结果满足要求，将版图送到代工厂(foundry)加工，如果不符合要求，修改版图后重新进行 LPE 和后仿真，如果还是不能满足要求，就要考虑修改或重新进行电路图设计。

下面以一个反相器为例说明标准单元或模拟集成电路设计的设计流程以及用到的 EDA 工具的使用方法。

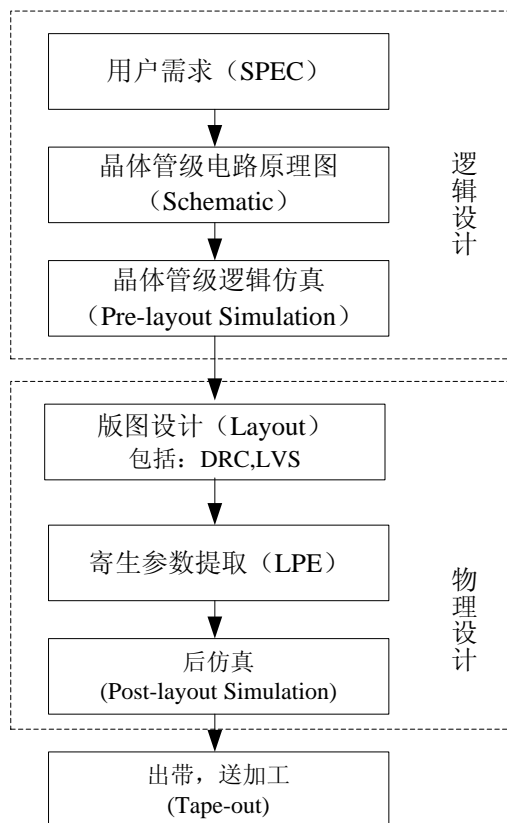


图 1.1 “自底向上”设计流程

1.2.1 电路图设计

一个反相器由一个 PMOS 管和一个 NMOS 管漏极相连而成。首先，按照前几章讲到的电路设计方法，根据具体的工艺参数，估算 PMOS 管和 NMOS 管的宽长比。晶体管的长度采用所选工艺的特征尺寸。以某代工厂 0.6 μm 工艺为例，特征尺寸为 0.6 μm ，计算得到 PMOS 管和 NMOS 管的宽长比分别为 6/1 和 3/1，那么 PMOS 管和 NMOS 管的宽度分别为 3.6 μm 和 1.8 μm 。设计的反相器电路草图如

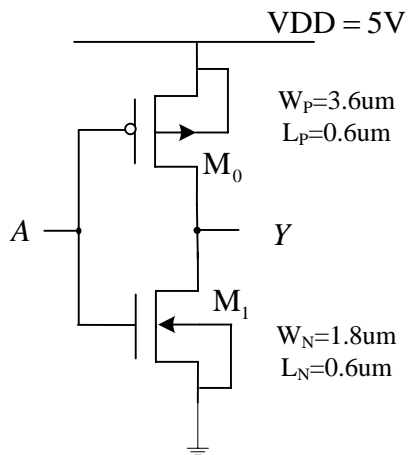


图 1.2 反相器设计电路草图

1.2.2 电路原理图输入

这一步是为电路仿真做准备，本例介绍用 Synopsys 公司的晶体管级仿真工具 Hspice 进行仿真。也可以用 Pspice、Spectre、SmartSpice 等工具，步骤略有不同。对于晶体管数目比较少的设计，直接可以在电路草图中标注节点人工写出 Hspice 的输入网表(netlist)，但是如果设计由较多晶体管，人工方法容易出错，所以这里采用先用 EDA 工具画出电路原理图(schematic)，然后通过 EDA 工具自动导出电路用于仿真的 Hspice 网表。

输入电路原理图的工具采用 Cadence 公司的 Composer-Schematic 工具。具体步骤如下：

(1) 建立库

① 在 Unix/Linux 控制台，启动 Cadence 平台。假设控制台提示符为 *piano%*

piano% icfb &

运行后 CIW 窗口显示如下：

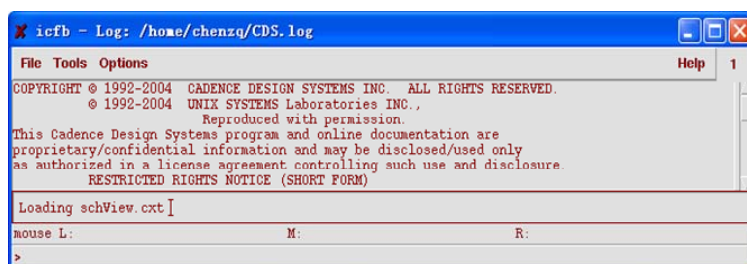


图 1.3 启动 cadence 平台

② 在 CIW 窗口菜单中依次选择 File-New-Library，新建一个名字为“INV”的库用于保存设计。在弹出窗口中输入库名：“INV”，技术文件（Technology File）项选择：“Don’t need a techfile”。按“OK”按钮建库。



图 1.4 建立库

(2) 画电路原理图

①在 CIW 窗口菜单中依次选择 File-New-Cellview，Library Name 选择上一步刚建立的 INV，Cell Name 输入要建立的单元的名字：“inverter”，Tool 选择“Composer-Schematic”，

这个工具用来建立电路原理图。设置完毕按“OK”进入 schematic 编辑窗口。

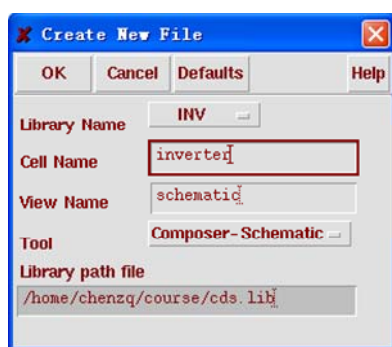


图 1.5 创建 schematic 文件

②在 schematic 编辑窗口菜单中选择 Add-instance。在 Add Instance 窗口中点击“Browse”按钮，弹出 Library Browser 窗口。在 Library Browser 窗口中，Library 选择“analogLib”（如果 analogLib 库不存在，在 cds.lib 中添加“DEFINE analogLib \$CDS_INST_DIR/tools/dfII/etc/cdslib/artist/analogLib”），Cell 项选择“PMOS4”，View 项选择“symbol”，选择后将鼠标在 schematic 窗口中点击一次，添加一个四端 PMOS 管。同样的方法一次添加“NMOS4,vdd,gnd”三个元件符号。

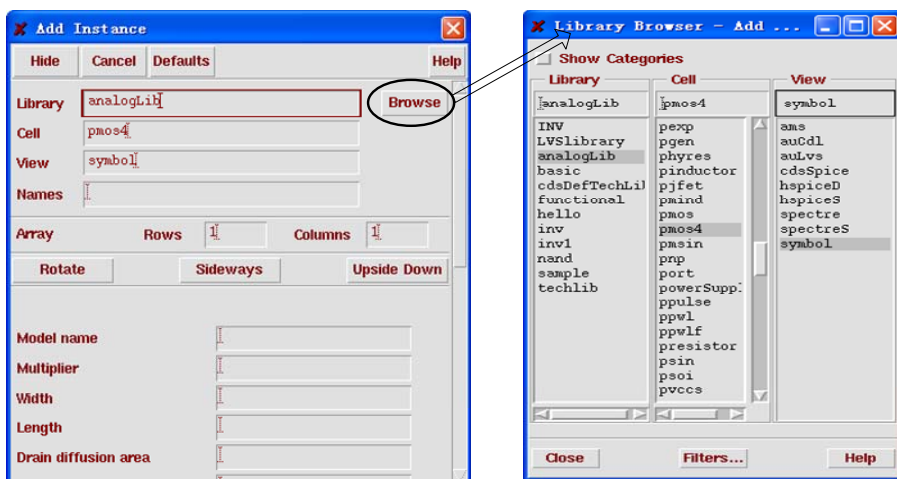


图 1.6 添加实例

③在 schematic 编辑窗口菜单中选择 Add-pin，添加输入引脚“A”和输出引脚“Y”。添加“A”引脚的方法：在弹出窗口中 Pin Names 中输入“A”，Direction 选择“input”，Attach Net Expression 选择“NO”；添加“Y”引脚的方法：在弹出窗口中 Pin Names 中输入“Y”，Direction 选择“output”，Attach Net Expression 选择“NO”。

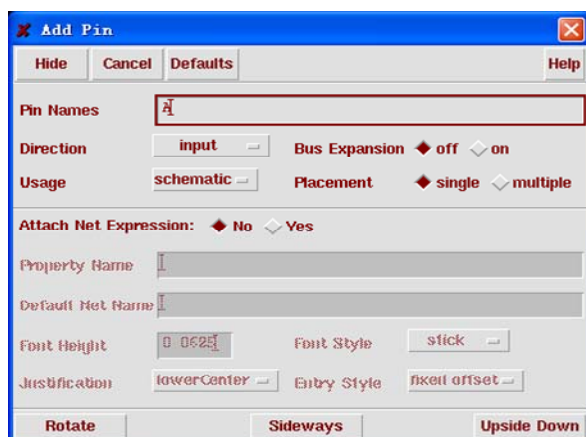


图 1.7 添加引脚

④然后在菜单中选择 Add-wire 将所有添加的部件按设计草图连接。最后输入 PMOS 管和 NMOS 管的宽度和长度，方法：选中要设置的 MOS 管，按键盘“q”键，调出 MOS 管的属性窗口，设置里面的 width 和 length 参数。完成的电路原理图如下图所示。

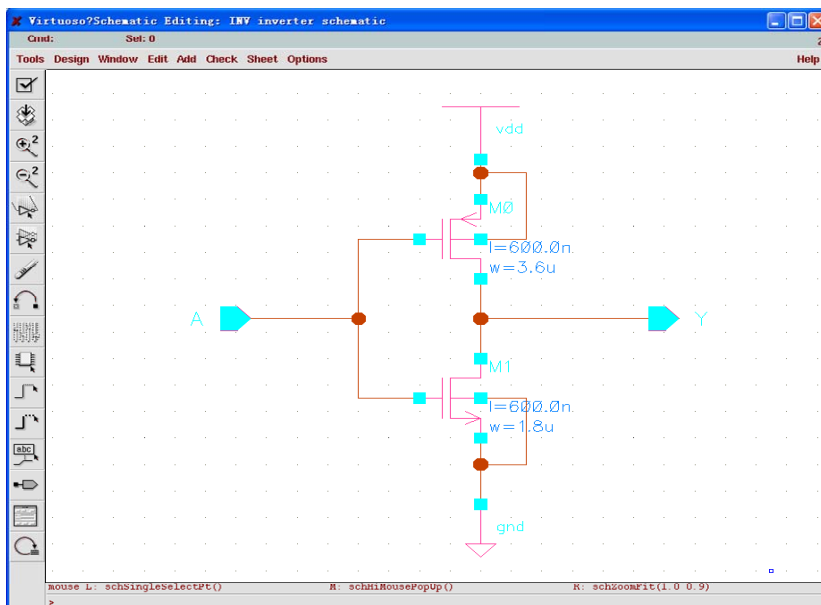


图 1.8 完成后的电路图

1.2.3 Hspice 仿真

- ① 导出 spice 子电路网表。在图 1.3 的窗口菜单中依次选择 File-Export-CDL。点击 Library Browse 选择上一步编辑的 schematic, 选择后的窗口如图 1.9 所示。Output File 和 Run Directory 项可以自行修改，这里 Output File 名称设为“inverter.sp”，Run Directory 默认为当前目录（icfb 运行目录），即导出的文件为当前目录下名称为 inverter.sp 的文件。
- ② 编辑导出的 inverter.sp 文件，加上激励和要进行的分析。如要对反相器电路进行瞬态分析，修改后的文件内容为：

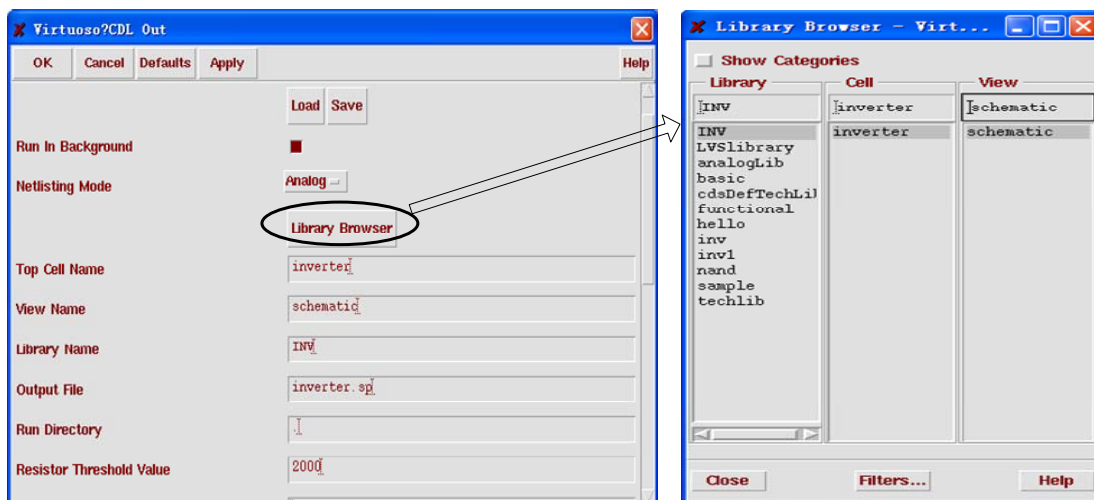


图 1.9 导出网表设置

Inverter Circuit *第一行为注释行
 .options list node post
 .lib 'library/spice06.l' TT * 指定仿真用的库文件
 .GLOBAL gnd! vdd! * 定义全局变量
 Vdd vdd! 0 5 * 设置 vdd!电压为 5V
 Vss gnd! 0 0 *设置 gnd! 电压为 0V

* Library Name: INV 导出的子电路修改后保留的部分
 * Cell Name: inverter
 * View Name: schematic

.SUBCKT inverter A Y
 *.PININFO A:I Y:I
 MM0 Y A vdd! vdd! PM W=3.6u L=600.0n
 MM1 Y A gnd! gnd! NM W=1.8u L=600.0n
 .ENDS

X1 in out inverter *外电路调用子电路，节点 in 对应子电路 A，节点 out 对应子电路 Y
 vin in 0 pulse 0 5 2n 0n 0n 5n 10n *输入波形
 .tran 200p 10n *瞬态分析
 .print tran v(in) v(out) *输出 in 和 out 节点电压
 .end * 程序结束标志

- ③ 运行 `hspice inverter.sp > inverter.lis` 仿真，如出错，在 `inverter.lis` 文件查找错误的原因。
- ④ 观察波形，运行 `awaves inverter` 查看仿真波形，评估设计是否达到要求。如图 1.10 所示

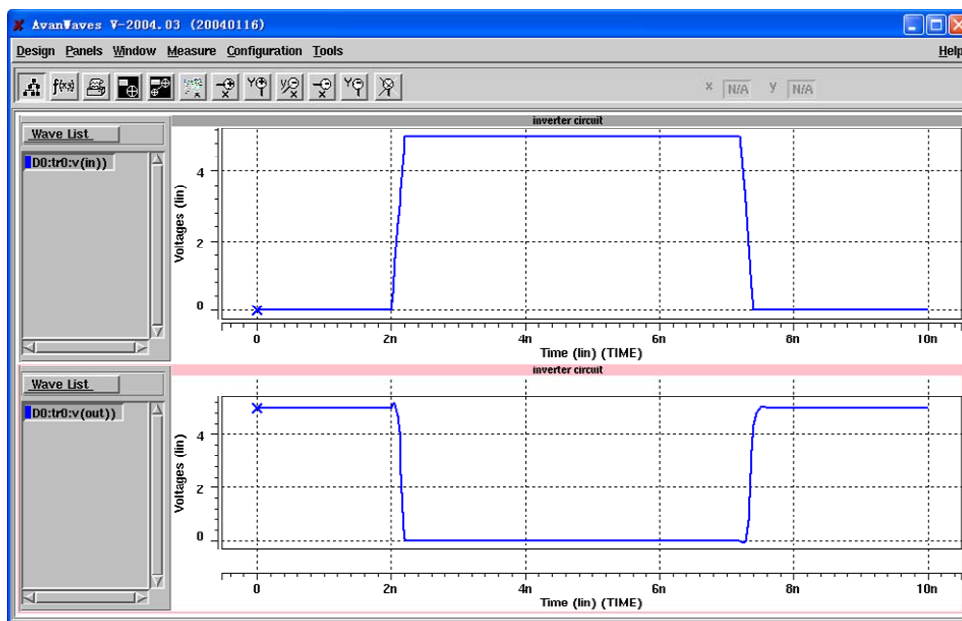


图 1.10 仿真波形图

1.2.4 版图设计

版图设计利用 cadence 公司的 Virtuoso 工具

1.2.4.1 准备工作

(1) 技术文件绑定

要建立的是一个版图文件，保存版图文件的库必须包括技术文件。上一节建立“INV”库的时候 Technology File 项选择了“Don't need a techfile”，也就是说，已经建立的“INV”没有包含技术文件。所以需要将 INV 库绑定到 techlib 库中（techlib 库已经实现导入了技术文件，导入的方法这里不做详细说明）

在 CIW 窗口菜单依次选择 Tools-Technology File Manager，窗口中选择“attach”，弹出窗口如图 1.11 所示。Design Library 选择已建立的库“INV”，要绑定到的库 Technology Library 选择包含技术文件的库“techlib”

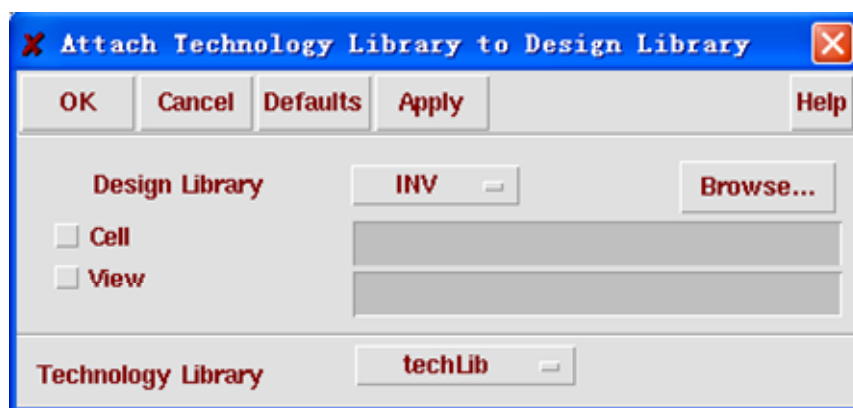


图 1.11 绑定库

(2) 准备 DRC, LVS 用规则文件

将代工厂提供的用于 DRC、Extract 和 LVS 用的规则文件复制到 techLib 库中，供设计过程中调用。我们绑定的 techlib 库中已经存在这些规则文件，所以不用拷贝了。

后面需要这些文件的地方会自动调用。

1.2.4.2 建立版图文件

在 CIW 窗口菜单或者 Library Manager 窗口菜单依次选择 File-New-Cellview，弹出窗口如图 1.12 所示。

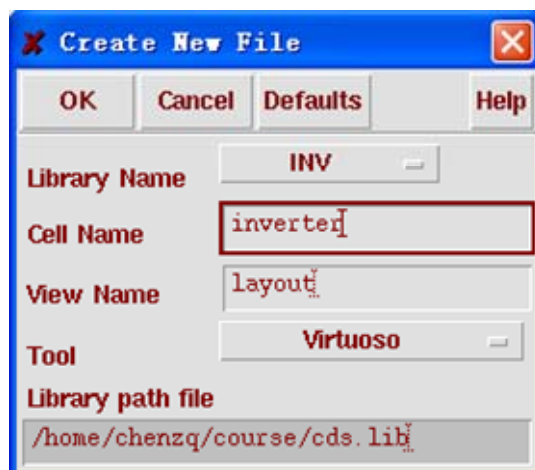


图 1.12 新建版图文件

Library Name 项选择已经包含 schematic 的库“INV”，Tool 中选择 virtuoso，然后点击“OK”按钮，打开 virtuoso editing 窗口，如图 1.13 所示。

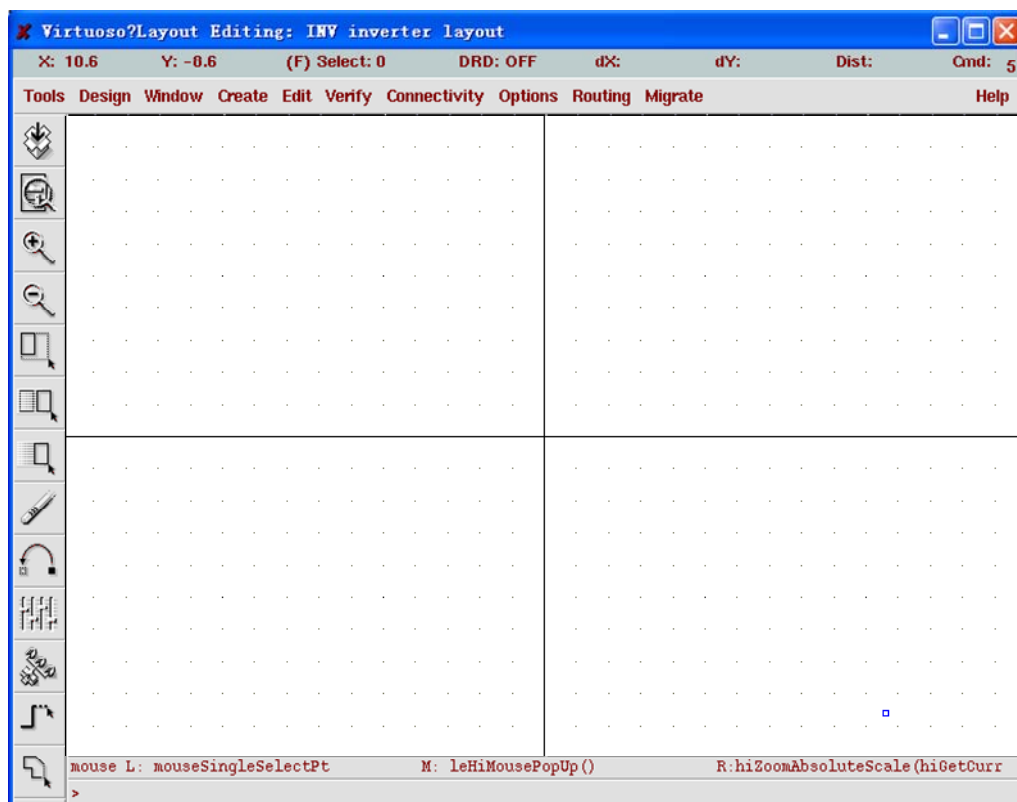


图 1.13 版图编辑窗口

版图视窗打开后，掩模版图窗口显现。视窗由三部分组成：**Icon menu** , **menu banner** , **status banner**.

Icon menu (图标菜单)缺省时位于版图图框的左边，列出了一些最常用的命令的图标，要查看图标所代表的指令，只需要将鼠标滑动到想要查看的图标上，图标下方即会显示出相应的指令。

menu banner (菜单栏),包含了编辑版图所需要的各项指令，并按相应的类别分组。几个常用的指令及相应的快捷键列举如下：

Zoom In	-----放大 (z)	Zoom out by 2	----- 缩小 2 倍(Z)
Save	----- 保存编辑(f2)	Delete	----- 删除编辑(Del)
Undo	----- 取消编辑(u)	Redo	-----恢复编辑 (U)
Move	----- 移动(m)	Stretch	----- 伸缩(s)
Rectangle	-----编辑矩形图形(r)	Polygon	----- 编辑多边形图形(P)
Path	----- 编辑布线路径(p)	Copy	-----复制编辑 (c)

status banner (状态显示栏)，位于 **menu banner** 的上方，显示的是坐标、当前编辑指令等状态信息。

在版图视窗外的左侧还有一个层选择窗口（**Layer and Selection Window LSW**）。

LSW 视图的功能：

- 1) 可选择所编辑图形所在的层；
- 2) 可选择哪些层可供编辑；
- 3) 可选择哪些层可以看到。

以下仅列出绘制我们这个版图所需的最少版图层次。

层次名称	说明
nwell	N 阱
active	有源区
pimp	P 型注入掩膜
nimp	N 型注入掩膜
contact	引线孔，连接金属与多晶硅/有源区
metal1	第一层金属，用于水平布线，如电源和地
via	通孔，连接 metal1 和 metal2
metal2	第二层金属，用于垂直布线，如信号源的 I/O 口
poly	多晶硅，做 mos 的栅

下图是修改后的 LSW。



图 1.14 LSW

1.2.4.3 绘制版图

(1) 画 PMOS 的版图

- 1) 画出有源区

在 LSW 中, 点击 active (dg), 注意这时 LSW 顶部显示 active 字样, 说明 active 层为当前所选层次。然后点击 icon menu 中的 rectangle, 在 vituoso editing 窗口中画一个宽为 $4u$, 长为 $3.6u$ 的矩形。这里我们为了定标, 必须得用到标尺。点击菜单 windows-Create Ruler 即可得到。清除标尺点击菜单 windows-Clear All Ruler。如果你在绘制时出错, 点击需要去除的部分, 然后点击 delete icon。

2) 画栅

在 LSW 中, 点击 poly (dg), 画矩形。与有源区的位置关系如下图:

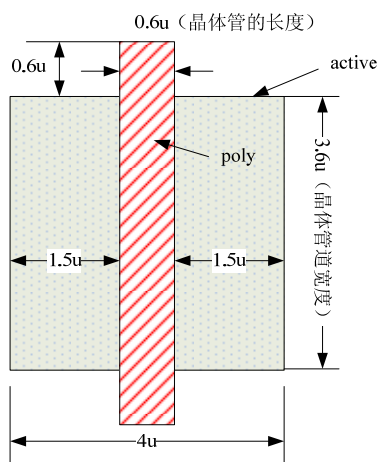


图 1.15 有源区和栅

3) 画整个 PMOS

为了表明我们画的是 PMOS 管, 我们必须在刚才图形的基础上添加一个 pmp 层, 这一层将覆盖整个有源区至少 $0.6u$ 。接着, 我们还要在整个管子外围画上 nwell, 它覆盖有源区至少 $1.8u$ 。如下图所示:

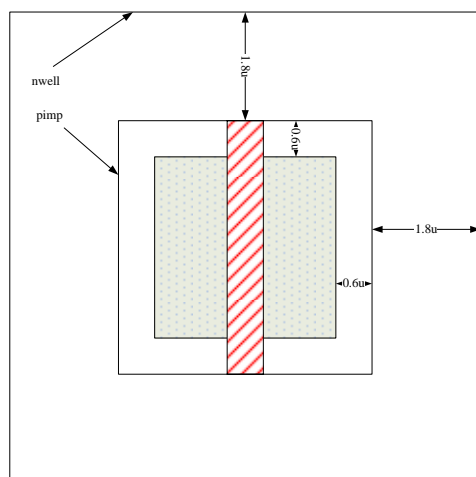


图 1.16 P 型注入区域和阱

4) 连接接触

在版图设计过程中, 要用到 contact 和 via 完成不同层的连接。由于 contact 和 via 和其它层之间要满足一定的设计规则, 而版图设计中又需要大量的连接接触, 如果对每一个接触都单独设计接触孔是一个费时的过程。所以可以将不同层之间的连接接触孔设计为一个单独

的实例，在设计中只需要调用相应接触孔实例即可。一般导入技术文件库中，已经包含了已经设计好的接触孔实例单元，在版图设计中直接调用即可。如果技术文件库中没有这些接触孔实例单元，我们可以根据设计规则自行建立不同层之间的接触孔实例单元。

在我们已经建立的 techLib 库中，psdco 是连接 p1imp 层与 metal1 层的接触孔，nsdco 是连接 n1imp 与 metal1 层的接触孔，p2co 是连接 poly1 层与 metal1 层的接触孔，m1m2 是连接 metal1 层与 metal2 层的接触孔。在设计中，根据需要连接到层次，选择相应接触孔即可。

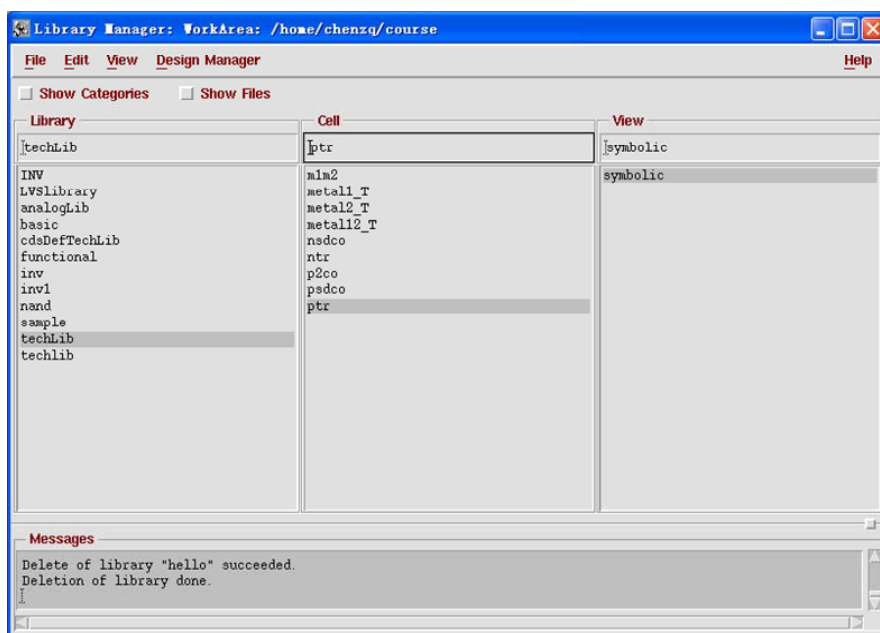


图 1.17 techLib 库中包含的实例单元

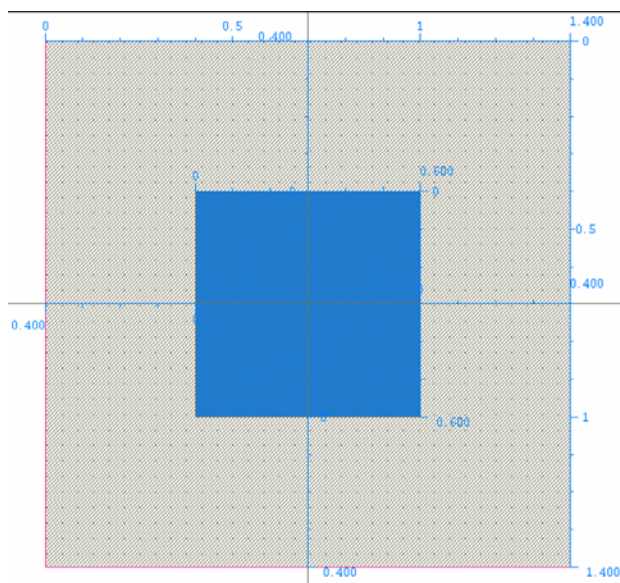


图 1.18 接触孔规则

图 1.18 为一个接触孔的设计规则。如果为 psdco 实例，该实例外围是由 $1.4\mu\text{m} \times 1.4\mu\text{m}$ 的 active 和 metal1 层重叠，中间的图形为 $0.6\mu\text{m} \times 0.6\mu\text{m}$ 的 contact，contact 与外围图形之间的距离为 $0.4\mu\text{m}$ ；我们导入的库中 nsdco 实例与 psdco 相同，外围是 active 与 metal1 层重叠；

如果是 p2co 实例，外围是 poly1 与 metal1 重叠；如果没 m1m2 实例，外围是 metal1 与 metal2 重叠，中间为 via 层。

① 画电源金属连线，宽度为 3u。将其放置在 PMOS 版图的最上方。

② 完成有源区（源区和漏区）的连接。在源区和漏区上添加 psdco 实例。调用实例方法：编辑窗口菜单依次选择 Create-instance，或者用快捷键“i”，弹出窗口中选择 techLib 库，cell 选择 psdco，view 选择 symbol，鼠标在版图窗口点击即可添加 psdco。

由于设计规则中 contact 与 poly1 的间距至少为 0.6um，而 psdco 中 contact 与外围间距为 0.4um，所以 psdco 与 poly1 之间的间距至少为 0.2um。

将 PMOS 管源极用 metal1 连接到电源上。

③ 衬底连接。PMOS 的衬底（nwell）必须连接到 vdd。nwell 与 vdd 要通过 N 型有源区与 metal 连接，调用 techLib 库中的 nsdco 实例。然后在添加的 nsdco 外围画一层 nimp，nsdco 与 nimp 间距为 0.6um。

如果 DRC 通过，PMOS 的版图绘制完成。布线完毕后的 PMOS 管版图如图 1.19 所示。接下来绘制 NMOS 管版图

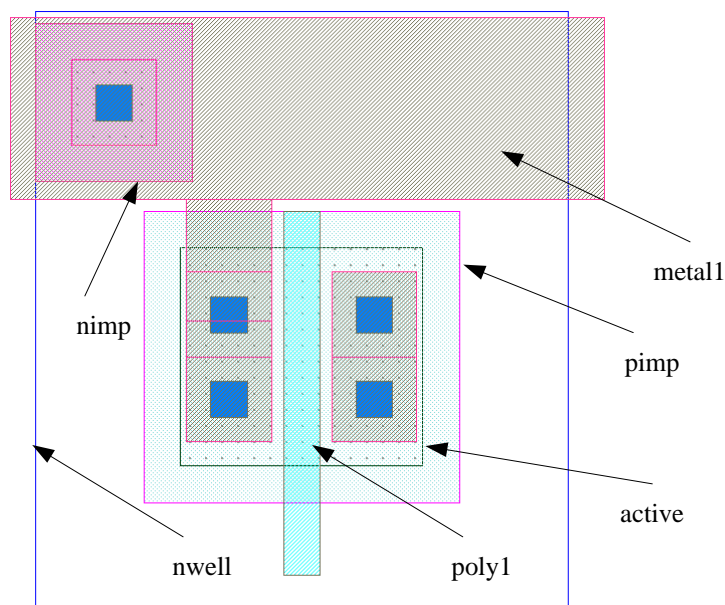


图 1.19 PMOS 管版图

(2) 画 NMOS 管

绘制 NMOS 管的步骤同 PMOS 管基本相同。区别是：1) NMOS 管直接制造在 P 衬底上，不需要画阱；2) 画 NMOS 管时有源区 active 被 nimp 覆盖；3) NMOS 管与衬底连接的有源区 active 覆盖的是 pimp。4) NMOS 管的宽度为 1.8u，长度为 0.6u。完成后的 NMOS 管版图如图 1.20 所示

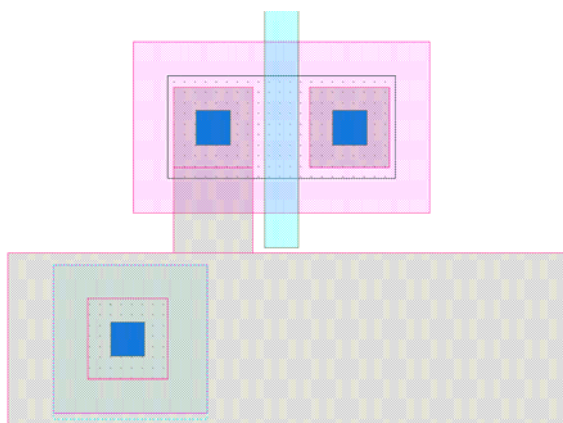


图 1.20 NMOS 版图

(3) 完成整个反相器版图

- 1) 将上面完成的两块版图以多晶硅为基准将两图上下对齐，然后，将任意一个版图的多晶硅延长和另外一个的多晶硅相交。
- 2) 输入：为了与外部电路连接，我们需要用到 metal1。但 poly1 和 metal1 不能直接相连，因此在多晶硅边放置一个接触孔，调用实例 p2co 即可。
- 3) 输出：先将两版图右边的 metal1 连起来（任意延长一个的 metal1，与另一个相交）。

(4) 作标签（这一步可省略）

- 1) 在 LSW 中选择层次 text (d1)，点击 create/label，在弹出窗口中的 label name 中填入 vdd！并将它放置在版图中相应的位置上。
- 2) 按同样的方法创制 gnd!、A 和 Y 的标签。

(5) 添加 Pin

为了进行版图提取，还要给版图文件标上端口，这是 LVS 的一个比较的开始点。在 LSW 窗口中，选中 metal1(pn)层；然后在 Virtuoso 环境菜单中选择 Create-Pin，弹出窗口（图 1.21 所示）中 mode 选择 shape pin。

填上端口的名称“A”（Terminal Names 和 Schematic 中的名字一样），I/O Type 输入引脚选 input，输出引脚选 output，电源和地引脚选 input，也可选 inputOutput。填好可以直接在版图中画上矩形端口。在版图的输入、输出、电源、地等处画上 pin，和编辑版图其它层次一样。这些端口仅表示连接关系，并不生成加工用的掩模板，只要求与实际版图上铝线接触即可，没有设计规则。

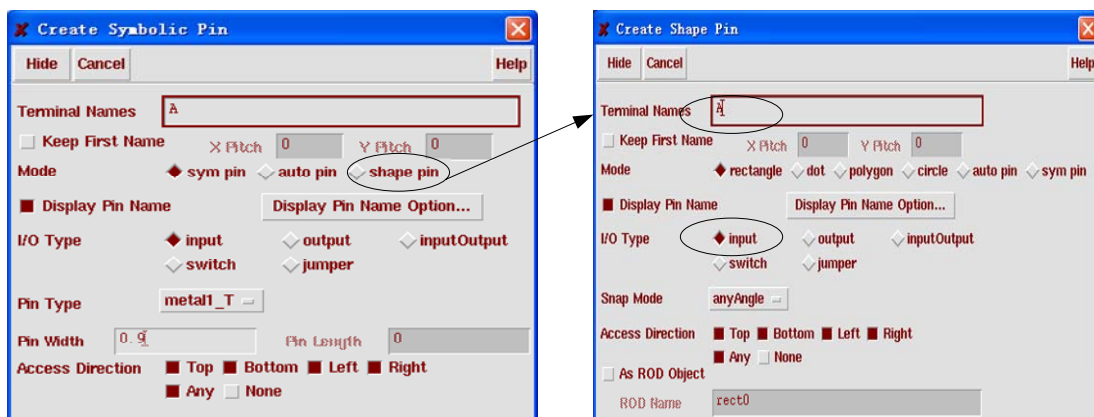


图 1.21 添加 Pin

完成后整个的版图

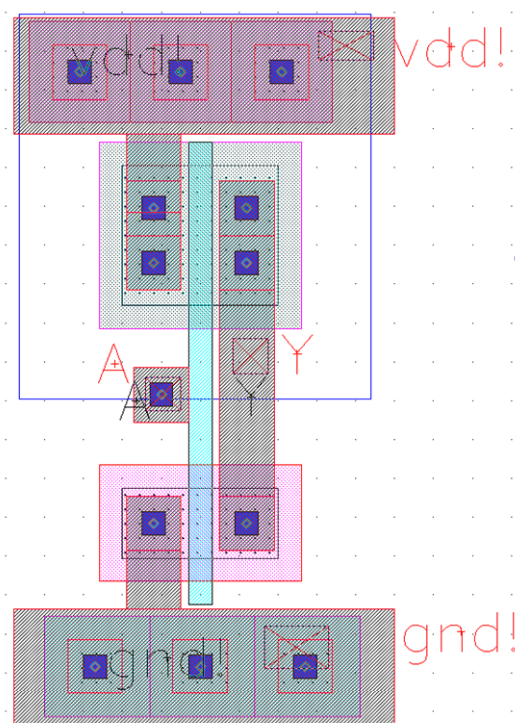


图 1.22 反相器版图

至此，我们已经完成了整个反相器的版图的绘制。下一步将进行 DRC 检查，以检查版图在绘制时是否有同设计规则不符的地方。

1.2.4.4 设计规则检查（DRC）

将代工厂提供的 DRC 规则文件（如：divaDRC.rul）复制到 techLib 库中，在 virtuso layout editing 窗口菜单依次选择 Verify-DRC，弹出窗口如图 1.23 所示。

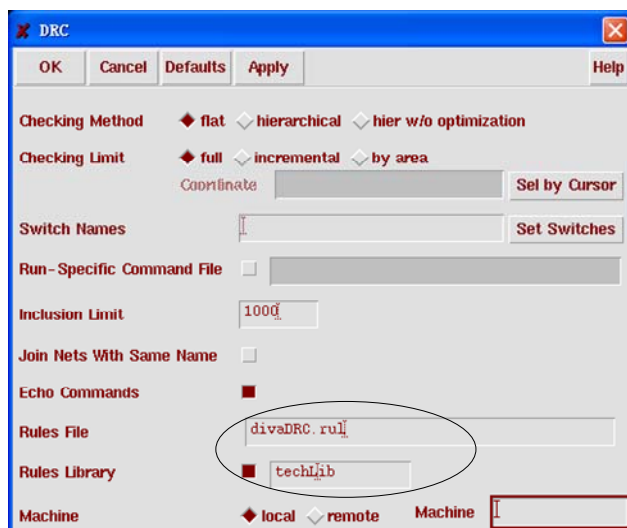


图 1.23 DRC 窗口

将弹出窗口中 Rules Library 设置为 techLib，Rules File 设置为复制的 DRC 规则文件，如：divaDRC.rul。然后点击“OK”按钮开始执行 DRC。如果设计中某区域违反了设计规则，该区域会高亮显示，同时 CIW 窗口的运行日志会显示具体违反了哪条规则，修正错误再重新执行 DRC，直到运行日志显示没有错误。

另外也可以选择 Verify-Markers-Find 菜单来帮助找错。单击菜单后会弹出一个窗口，在这个窗口中单击 apply 就可以显示第一个错误。同样，可以选择 Verify-Markers-Explain 来看错误的原因提示。选中该菜单后，用鼠标在版图上出错了，并且需要了解原因的地方单击就可以了。也可以选择 Verify-Markers-Delete 把这些错误提示删除。

1.2.4.5 版图提取

在进行 LVS 之前，首先要进行版图提取(Extract)，版图提取可以识别出版图表示的器件信息，如果哪些是 PMOS 管，哪些是 NMOS 管，哪些是电阻、电容，以及这些器件的连接关系等等。将提取出来的信息再与电路图进行比较。

版图的完成后，就可以提取了，在版图编辑环境下选择 Verify-extract。弹出窗口如图 1.24。填好提取文件库和文件名后，单击 OK。

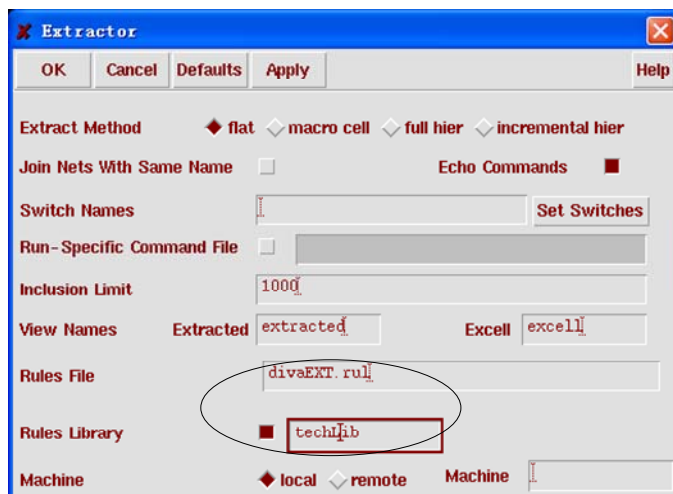


图 1.24 版图提取设置

执行完成后，打开 Library Manager 窗口在库“INV”下，inverter 单元中增加了一个文件 extracted 类型就是提取出来的文件，可以用打开版图文件同样的方式打开它。图 7 中，是提取出来的版图，可以看到提取出来的器件和端口。

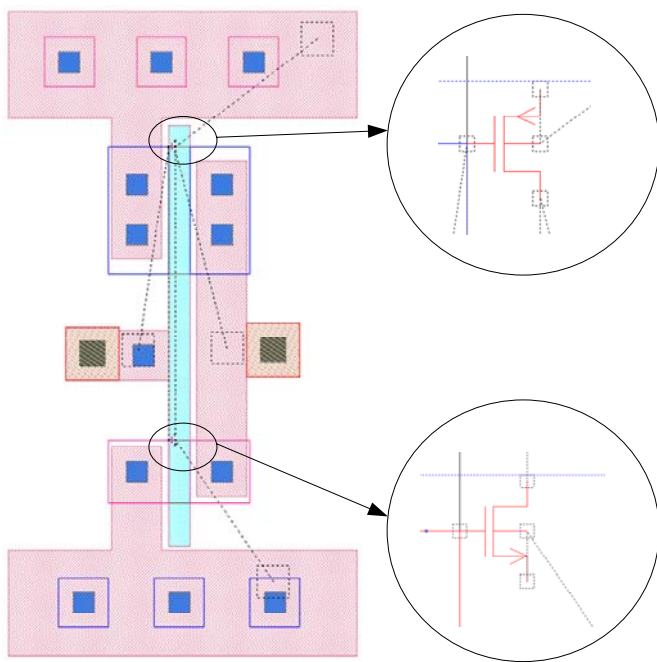


图 1.25 提取出的版图

在版图和线路图的准备工作完成后就可以进行 LVS 了。

1.2.4.6 版图与电路图一致性检查 (Layout vs Schematic,LVS)

版图通过 DRC 检查，只能说明版图没有违反设计规则，而不能说明版图描述的器件逻辑的正确性。所以经过 DRC 和 Extract 之后，还要进行 LVS，验证从版图提取出的连接关系是否与 1.2.2 节所输入的电路原理图是否一致。

在 virtuso layout editing 窗口菜单依次选择 Verify-LVS，弹出窗口设置图 1.26 所示。分别填好 schematic 和 extracted 文件所在的库和单元名称以及类型，也可以通过 Sel by Cursor 按钮选择正在打开的窗口；Rules Library 和 Rules File 项分别填入用于 LVS 的技术文件所在的库和文件名称；将优先级 Priority 设置为 20。然后点击 Run 按钮。

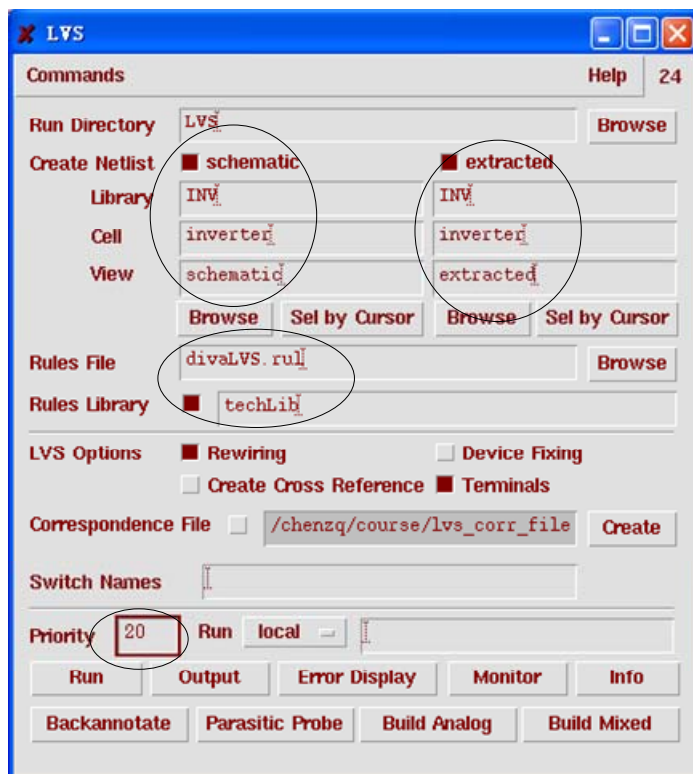


图 1.26 LVS 设置

当系统提示分析成功后，点击 output 按钮查看比较结果。输出文件部分内容如下：

Net-list summary for /home/chenzq/course/LVS/layout/netlist

```
count
4          nets
4          terminals
1          pmos
1          nmos
```

这一部分给出了从版图中提取的信息

Net-list summary for /home/chenzq/course/LVS/schematic/netlist

```
count
4          nets
4          terminals
1          pmos
1          nmos
```

这一部分给出了电路图中包括对信息。以上两部分的相应元素对应的数目一致说明版图的信息和电路图一致。

Terminal correspondence points

```
N3      N5      A
N0      N2      Y
N2      N1      gnd!
N1      N0      vdd!
```

The net-lists match.

	layout	schematic
	instances	
un-matched	0	0
rewired	0	0
size errors	0	0
pruned	0	0
active	2	2
total	2	2

这部分说明版图和电路图中的实例完全一致。

	nets	
un-matched	0	0
merged	0	0
pruned	0	0
active	4	4
total	4	4

这部分说明版图和电路图中的连线完全一致。

	terminals	
un-matched	0	0
matched but different type	1	1
total	4	4

这部分说明版图和电路图中的端子完全一致。

当输出文件中分析的数据完全一致时，说明版图设计没有问题。至此，反相器版图设计验证完成。