

## (一) linux 常用命令

### 1、cd

“cd FileName” 是进入 FileName 文件夹。

“cd ..” 是进入上一层目录，主要那里是两个点。

### 2、ls

这是最基本的文件命令，用以显示某一个目录中的文件。可以在 ls 后加上所要观察的目录名称或文件的名称，如果你在下 ls 命令后没有接任何的目录名，它将会显示出目前目录中所有文件。

ls 可以带一些参数，给予用户更多相关的信息：

-a：在 UNIX 中若一个目录或文件名字的第一个字元为 "."，该文件为隐藏文件，使用 ls 将不会显示出这个文件的名字。如 cshell 的初始化文件 .cshrc，如果我们要察看这类文件，则必须加上参数 -a。

-l：这个参数代表使用 ls 的长(long)格式，可以显示更多的信息，如文件存取权，文件拥有者(owner)，文件大小，文件更新日期，或者文件链接到的文件、文件夹。

-F：给目录、可执行文件、符号链接以特殊的标记，在可执行文件后面加“\*”、目录后面加“/”、符号连接后面加“@”，以方便用户区别。

### 3、cp

cp 命令用于将一个或多个文件复制成另一个文件或者是将其复制到另一个目录去。

cp 有三种基本格式如下：

cp source target 将文件名为 source 的文件复制一份为文件名为 target 的文件。如果 target 文件不存在，则产生文件名为 target 的文件，如果 target 文件存在，缺省时自动覆盖该文件。

cp file1 file2...dir 将文件 file1 file2 ... 都以相同的文件名复制一份放到目录 dir 里面。

cp -r source target 拷贝整个目录，将 source 目录下的所有文件和文件夹复制到 target 目录下。

cp 命令常用的选项参数有：

-i：交互(interactive)模式，当文件名为 target 的文件存在时，缺省情况下该命令会自动将原来 target 的内容覆盖掉，加上改选项将询问用户是否覆盖。用户输入 y(yes)则执行复制的动作，否则放弃复制。

-r：复制文件夹，将该目录下所有的文件和文件夹都复制到另一个目录下。

### 4、mv

该命令重命名一个文件或文件夹，和 cp 命令一样，它也有三种基本格式：

mv source target 将 source 文件重命名为 target。

mv dir1 dir2 将 dir1 目录重命名为 dir2。

mv file1 file2...dir 将文件 file1 file2...都移动至目录 dir 中。

mv 命令常用的参数有：

-i：同 cp 命令，重命名时使用交互模式。

-f：强迫重命名文件和文件夹，覆盖原来的文件和文件夹，在执行重命名操作时不产生任何警告信息。

## 5、rm

**rm** 命令用于删除文件。除非系统中有被删除的文件的备份，否则是无法恢复的，所以用户在使用 **rm** 命令时应该特别小心，防止误删重要文件。

**rm** 命令的基本格式如下：

**rm file1 file2 file3 ...**          删除 file1,file2,file2...

**rm** 的常用选项有：

- f：强制删除。使用 **rm -f** 命令删除文件时，不管文件有没有写保护，都不产生任何警告信息。但是若该文件夹下面的文件夹是写保护的话，则无法删除文件夹和该文件夹下的文件。-f 选项必须慎用。
- i：交互模式，在删除文件或文件夹之前询问用户，确认之后才执行删除操作，否则放弃执行。
- r：删除文件夹与文件夹下的文件。

## 6、mkdir

**mkdir** 让用户在有写权限的目录下建立一个目录或多个目录的命令。

**mkdir** 命令的基本格式如下：

**mkdir dirname1 dirname2 ...**

## 7、cd/chdir

**cd/chdir** 命令让用户转移当前工作目录，**chdir** 命令的基本格式如下：

**chdir target\_dir** 将当前的工作目录变为 **target\_dir** 目录。若 **cd** 命令后面不带参数，则工作目录变为用户的 **HOME** 目录。

## 8、vi 的工作模式

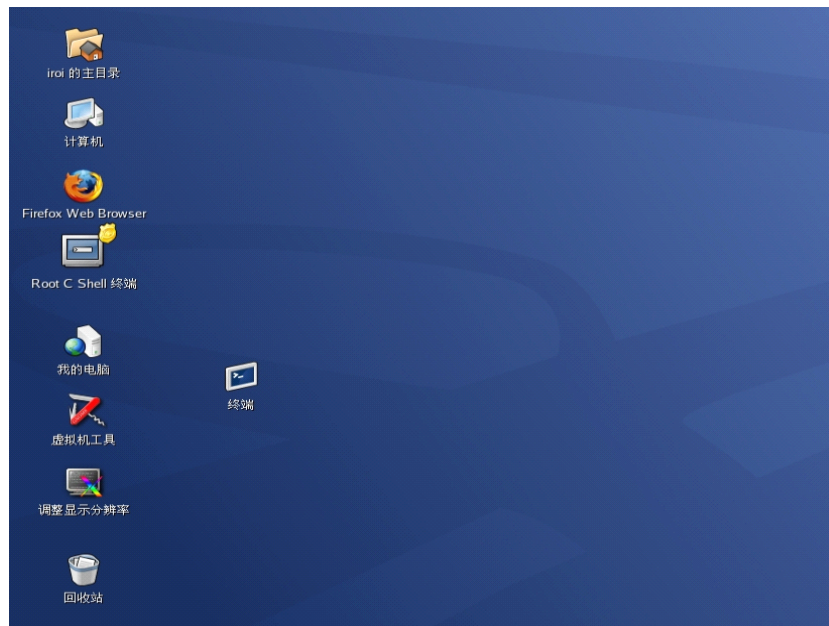
它有两种工作模式，命令模式和编辑模式。**vi** 在启动时先进入命令模式，当输入 **i**，**a**，**o** 等命令之后可进入编辑模式。除 **Esc** 之外的任何字符均将被看成是插入到编辑缓冲区中的字符，按 **Esc** 键之后，从编辑模式切换到命令模式。

在命令模式下使用“**:wq**”命令可存盘并退出。如果只需要退出程序，而不打算保存编辑的内容，可用下面的命令：

- :q                  在未作修改的情况下退出
- :q!                 放弃所有修改，退出编辑程序

下面介绍 cadence 软件的入门学习，原理图的创建、仿真，画版图和后仿真等一全套过程，本教程适合于初学着，讲得尽量详细和简单，按照给出的步骤可以完全的从头到尾走一遍，本教程以最简单的反相器为例。

## (二) 软件的启动



上图是桌面，在桌面上按右键，出现如下：



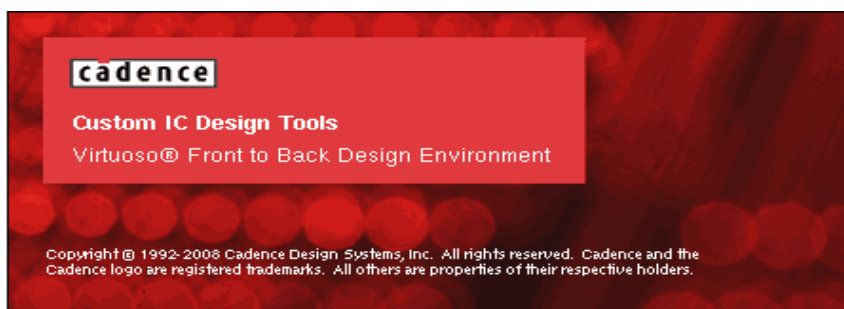
选择“打开终端”，得到如下：



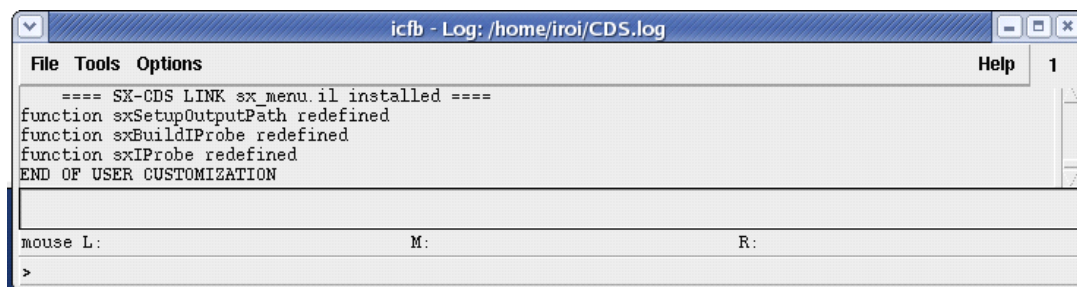
在此处输入：“cd my\_project”，按回车键后再输入“ic”，得到如下：



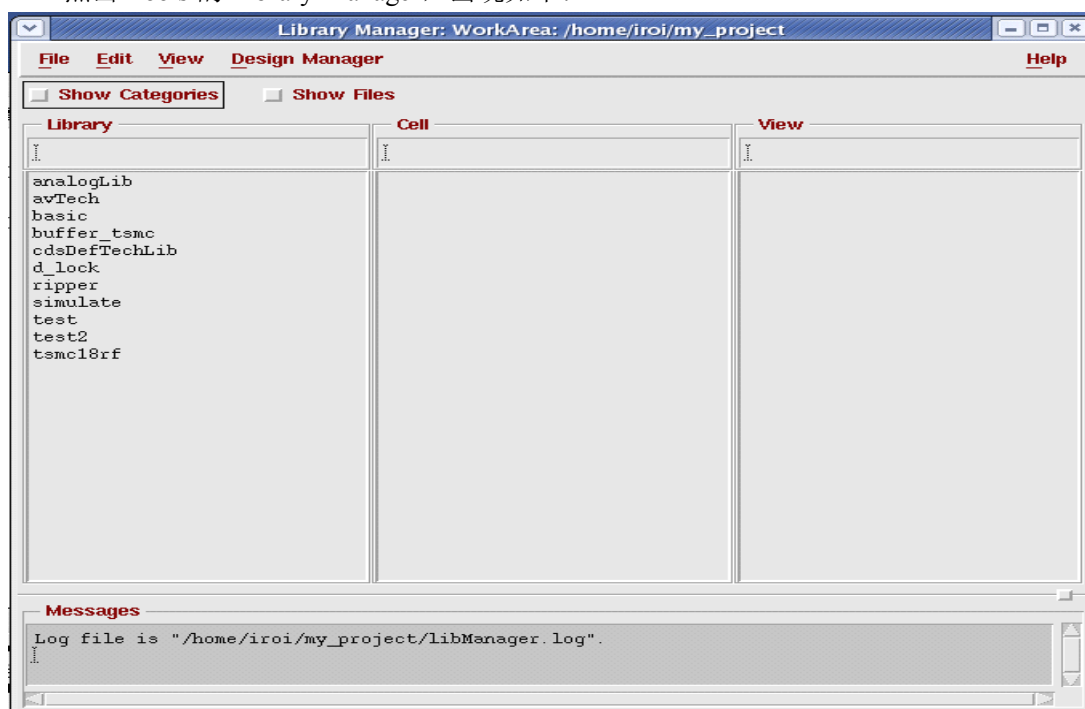
再回车，等一会后就出现了下图，也就是启动了软件：



启动后出现下图：（初次启动这里需要等一会儿）



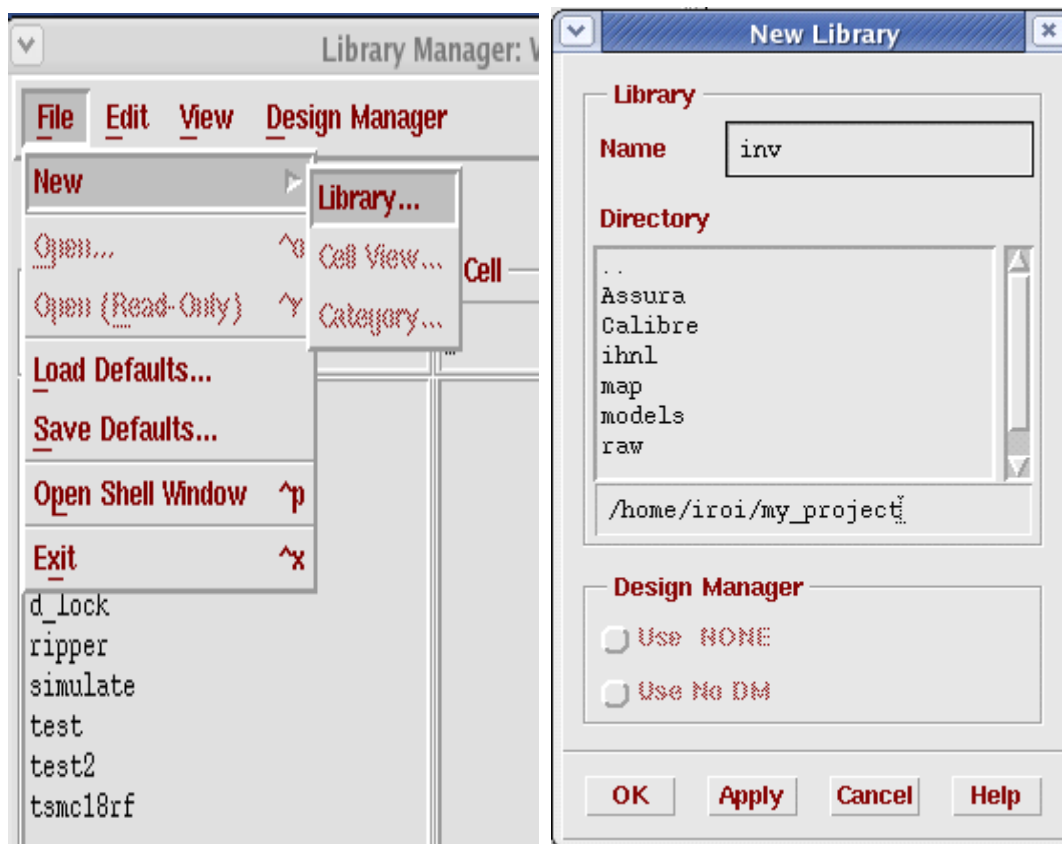
点击 Tools 的 Library Manager，出现如下：



上面显示的是文件管理窗口，可以看到文件存放的结构，其中 Library 就是文件夹，Cell 就是一个单元，View 就是 Cell 的不同表现形式。比如一个 mos 管是一个 Cell，但是 mos 管有原理图模型，有版图模型，有 hspice 参数模型，有 spectre 参数模型等，这就列举了 Cell 的 4 个 View。他们之间是树状的关系，即，Library 里面有多多个 Cell，一个 Cell 里面有多多个 View。应该保持一个好习惯就是每个工程都应该建立一个 Library。

### (三) 建立工程

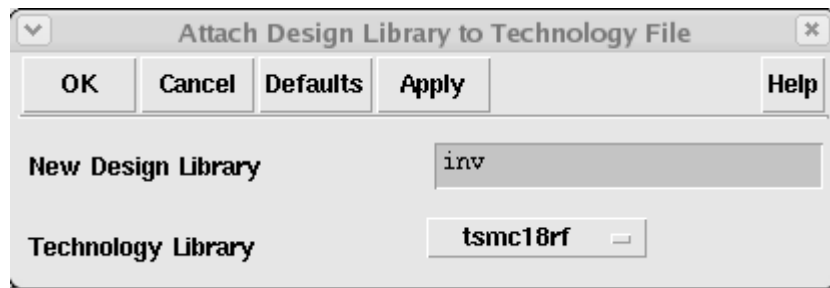
现在建立工程，新建一个 Library，如下左图，出现的对话框如下右图：



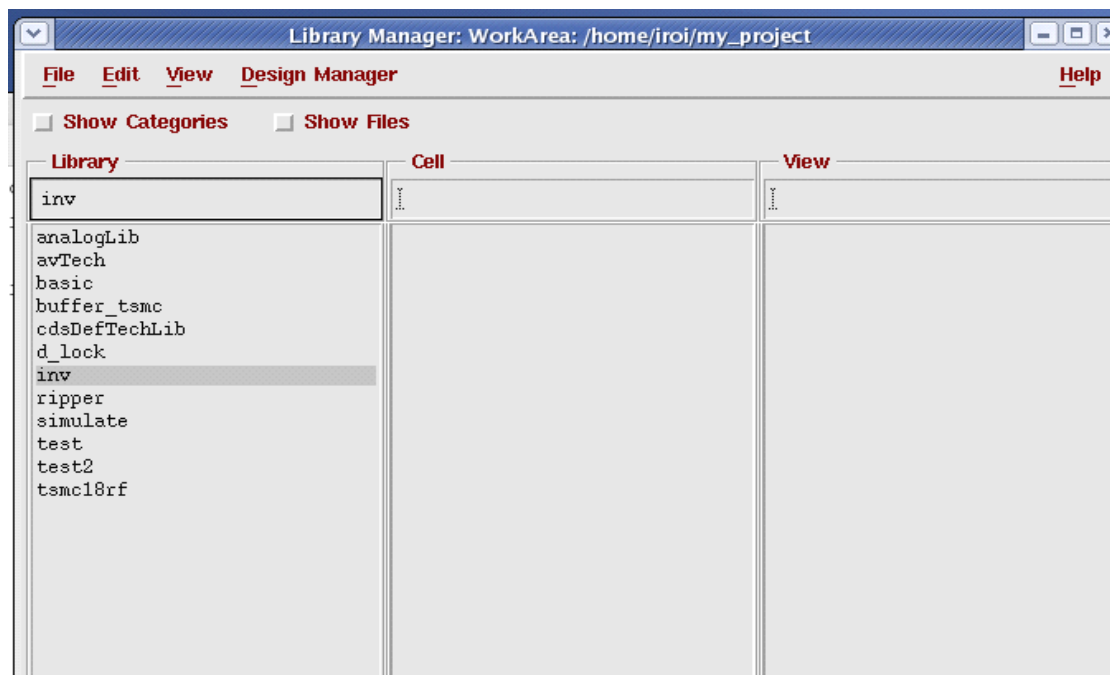
在上右图中选择合适的目录，这个目录很重要，统一选择/home/iror/my\_project。然后取新建的文件夹的名字，这里用的是 inv，建议用自己名字的拼音。以后的各种文件都在这个文件夹下。（这一点很重要，一个工程里面有很多文件，要是文件不管理好就会很乱）。OK 后出现下面对话框：



这个对话框是选择是否链接 techfile，如果只是原理图仿真而不用画版图，就选择 Dont need a techfile，这里我们要画版图，而且有工艺库，选择 Attach to an existing techfile，OK 后出现下面对话框：

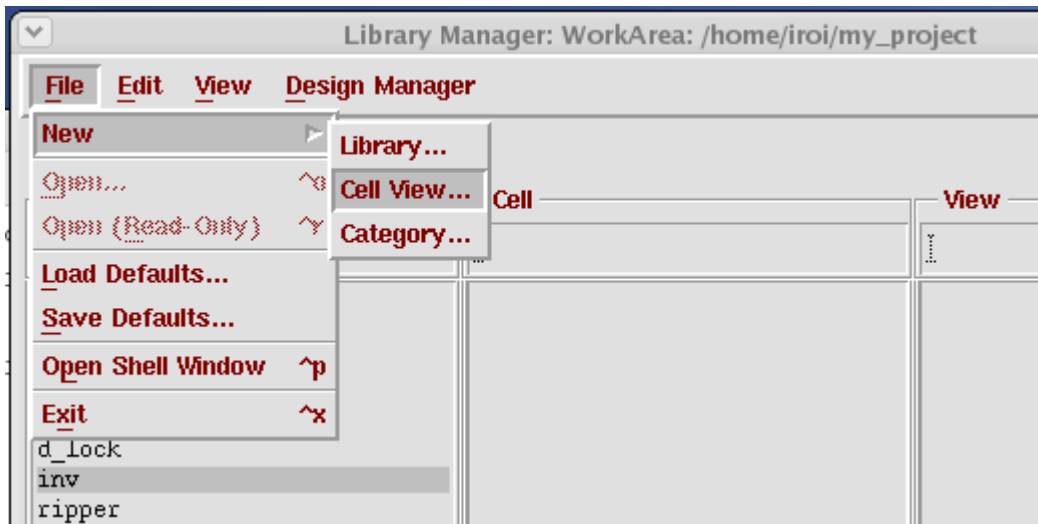


在 technology Library 选择 tsmc18rf, 我们使用的是这个工艺库。Inv 的文件夹就建好了, 在 Library Manager 就有它了, 如下图:

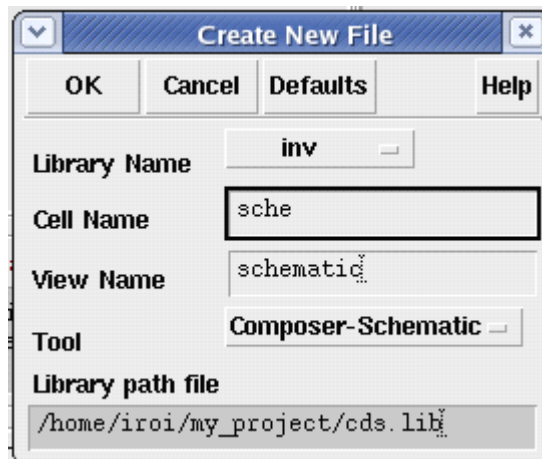


#### （四）画原理图

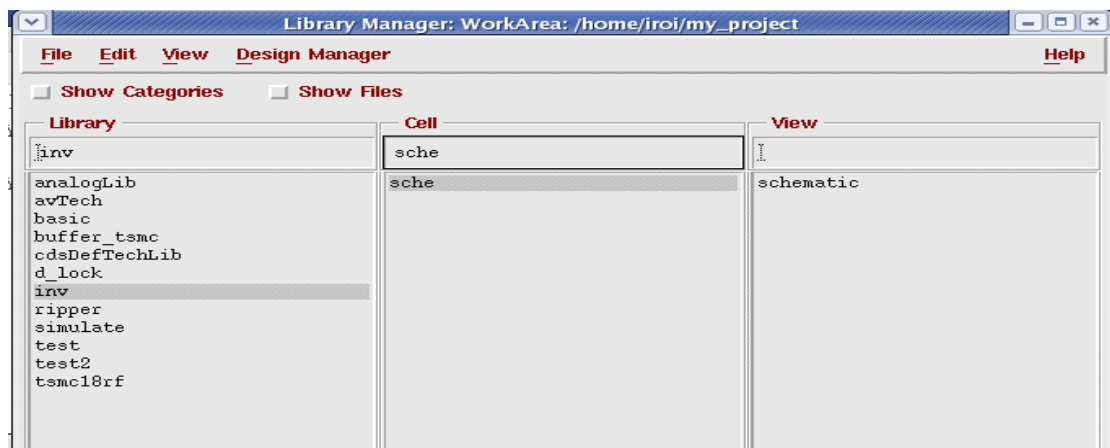
文件夹建好了后，我们要建立原理图，在 inv 的 Library 里面新建 Cell 如下：



然后出现新建 Cell 的对话框如下：

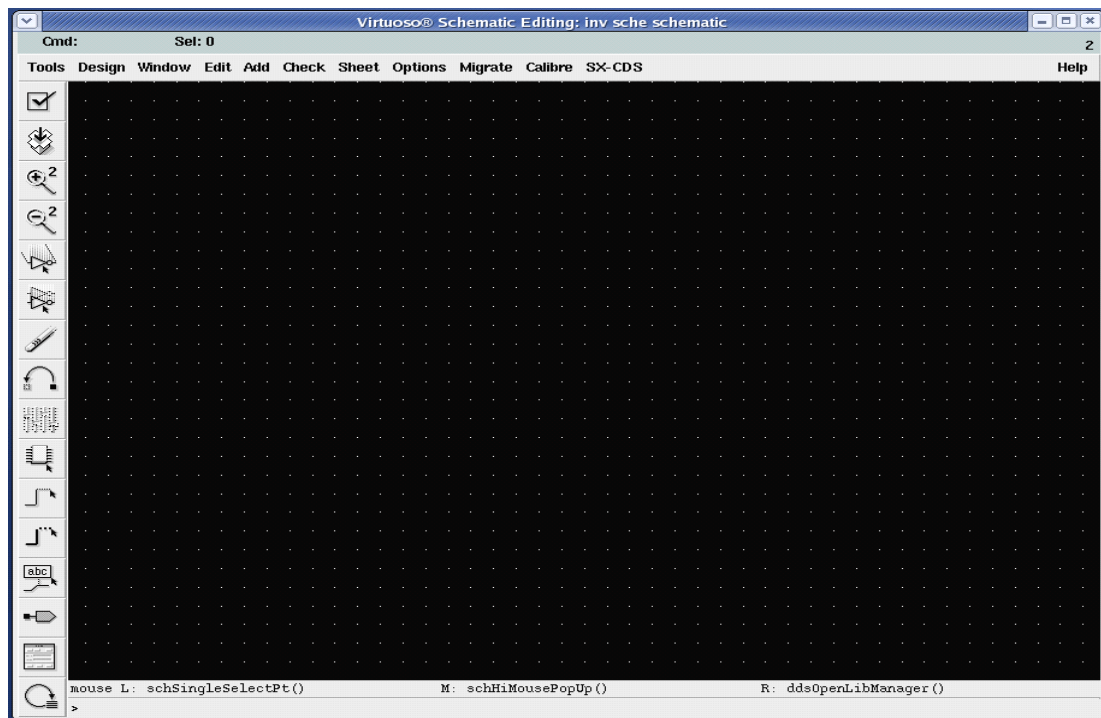


我这里取的名字是 sche 注意 View Name 是填的 schematic，Tool 用的是这个。点击 OK 之后发现 Library Manager 里面有如下变化：



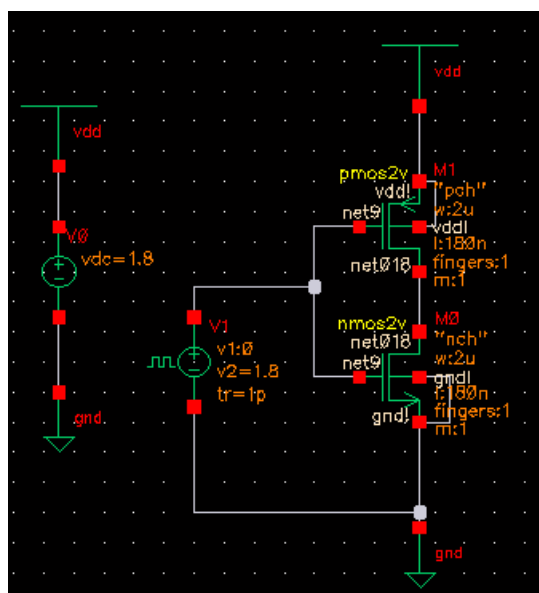
Cell 里面多了 sche，View 里面出来了 schematic，同时 schematic editing 窗口会打开，如果没有打开，可以双击 View 中的 schematic 打开，打开如下：





我们的原理图将在上面的面板中画，schematic editing 面板中的左边图标的意思我这里就不说了，可以看看其他版本的介绍。

下面开始介绍画原理图的步骤。下图是我们最终要画的原理图，一个简单的非门。



先对这个图进行说明。绿色的 vdd，它只是一个标号，说明每个 vdd 是连接在一起的，它并不提供电压和电流，真正的电源是 V0。同样，gnd 也是一样，只是一个连接的标记。图中的参数和网络标号是比较多的，不要看糊了。不管什么电路，所有 nmos 的 B 极（衬底）是接地的，所有 pmos 管的 B 极是接电源的。V1 是信号源，这里是个方波。

先介绍一些快捷键，快捷键以后会经常使用。

- i 是添加 instance （instance）
- f 是合适的显示所有内容 （fit）
- m 是移动 （move）



w 是连线 (wire)

q 看属性 (property)

p 添加引脚 (pin)

s 是保存 (save)

x 是检查并保存, 这个经常使用, 它会检查一些简单的连线错误。

鼠标上的前后滚轮是放大、缩小

e 进入 symbol 的内部电路

Ctrl+e 从 symbol 内部电路中退回

[ 是缩小

] 是放大

按住 shift 拖动 是复制添加

Delete 是删除

r 是 90 度旋转

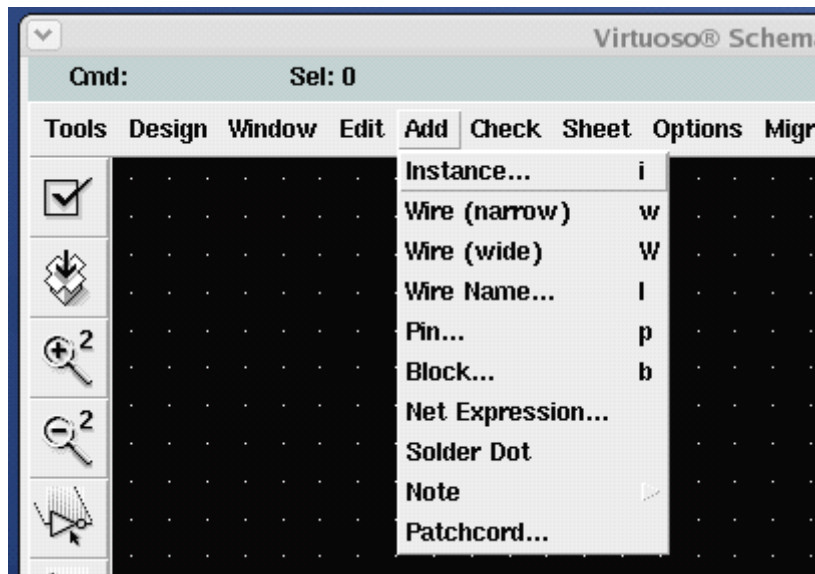
r 后再按 F3 可以选择左右翻转或者上下翻转

方向键当然可以上下左右移动

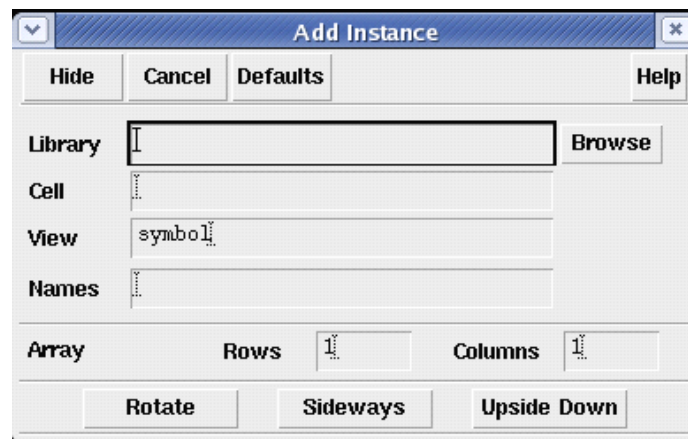
Esc 这个很重要, 是退出当前快捷方式, 要经常使用。

除非选择了另外的快捷键, 否则当前的快捷键一直存在, 所以经常用 esc。

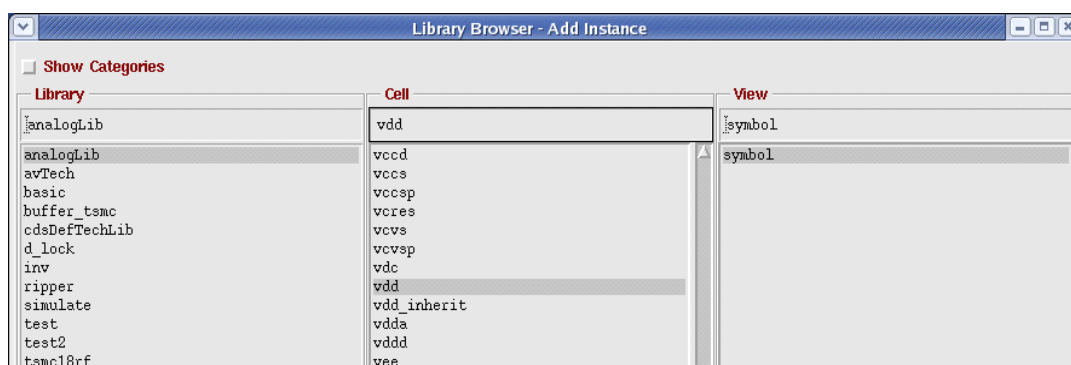
现在放置 vdd, 如下图, 点击 Add 的 Instance, 或者直接按键 i。



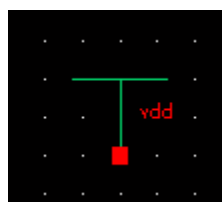
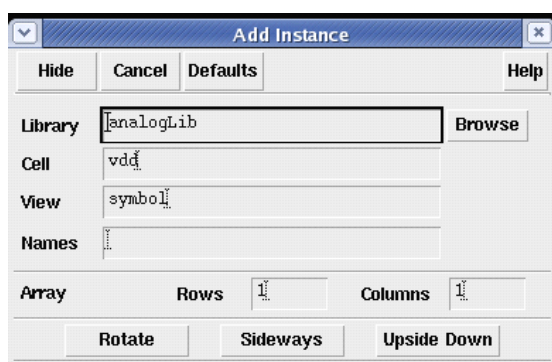
将出现下面的对话框:



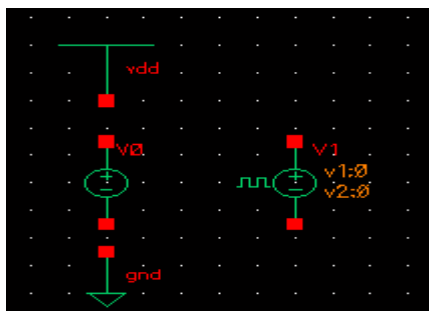
如果不知道库名和 Cell 名，就点击 Browse，出现如下：



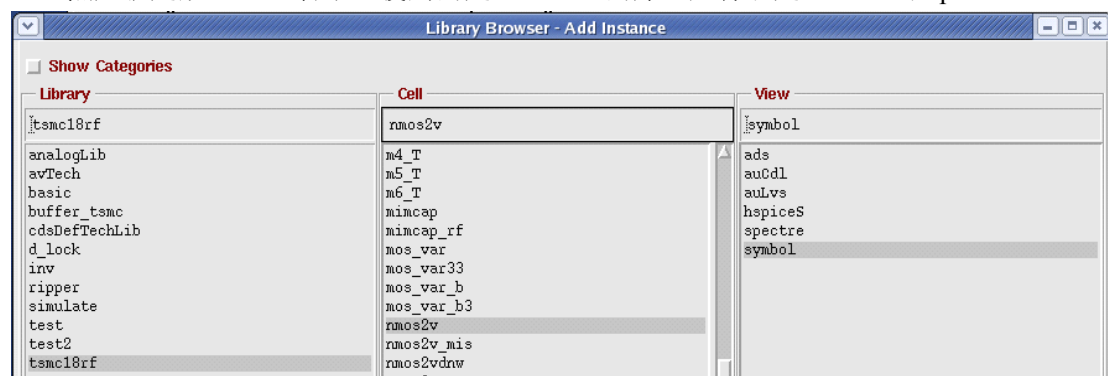
vdd 在 analogLib 库里面，找到 vdd，选择 View 为 symbol，然后将此窗口关掉，然后会发现，在 Add instance 窗口中已经将 vdd 的相关信息加入进来了，如下图，同时 vdd 已经悬浮在鼠标上，可以点击 Hide 将此窗口隐藏，然后可以随便放置 vdd 了。



同样的原理放置 gnd。记得使用 i 来添加 instance 的方法。vdc 是直流电压，也在 analogLib 库里面，同样放置，vpulse 是方波发生器，也在 analogLib 库里面，同样放置，放完后如下图：

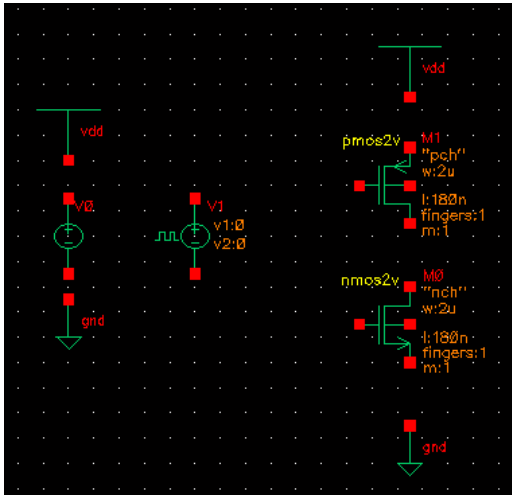


然后就是放置 mos 管了，使用的是 tsmc18rf 的库，元件名是 nmos2v 和 pmos2v

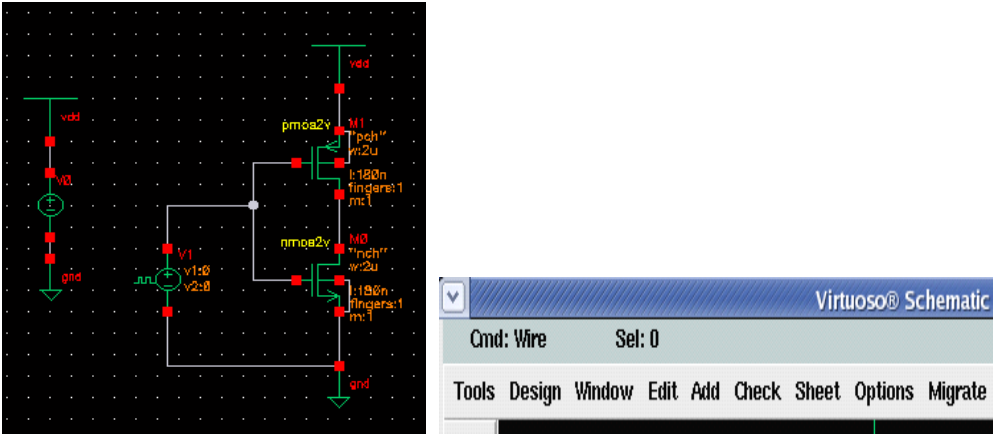


可以看得出来，库名是 tsmc18rf，库里面有很多元件，即很多 Cell，每一个 Cell 又有很

多模型。 所有的放置完后如下图：



按 w 键，开始连线，连线完后如下图



注意 mos 管的 B 极别忘了接。从左上角的 Cmd 可以看到当前使用的是什么命令。接下来就是要设置参数了。

单击 V0 将它选择，按 q 键，则弹出它的属性对话框如下：

Property	Value	Display
Library Name	analogLib	off
Cell Name	vdc	off
View Name	symbol	off
Instance Name	V0	off

User Property	Master Value	Local Value	Display
IvIgnore	TRUE		off

CDF Parameter	Value	Display
AC magnitude		off
AC phase		off
DC voltage	1.0	off
Noise file name		off
Number of noise/freq pairs	0	off
XF magnitude		off
PAC magnitude		off

V0 是直流电压，我们用的是 0.18um 的工艺库，电源电压是 1.8V，这里只要设置 DC voltage 为 1.8 就可以了，这个软件所有参数的设置都不用单位，比如 1.8V 只要写 1.8 就可以了，加了 V 反而错了。再如 10ns 只要些 10n 就可以了。点击 OK 退出。

然后点击 V1，按 q 键，打开 V1 的属性如下：

Property	Value	Display
Library Name	analogLib	off
Cell Name	vpulse	off
View Name	symbol	off
Instance Name	V1	off

Property	Value	Display
AC magnitude		off
AC phase		off
DC voltage	0 V	off
Voltage 1		off
Voltage 2	1.8 V	off
Delay time	1n s	off
Rise time	1p s	off
Fall time	1p s	off
Pulse width	1n s	off
Period	2n s	off
Frequency name for 1/period		off
Noise file name		off

这里需要设置的参数多些，方波嘛，有两个电压，这里设置的 voltage1 为 0，voltage2 为 1.8，后面的单位都是系统加上的。Delay time 是延时时间，rise time 是上升时间，fall time 是下降时间，pulse width 是脉冲宽度，也就是 voltage2 的持续时间，period 是方波的周期，这样，一个方波就被完全确定了。点击 OK 退出。

然后是 MOS 管的参数，如下：

Property	Value	Display
Library Name	tsmc18rf	off
Cell Name	pmos2v	value
View Name	symbol	off
Instance Name	M1	off

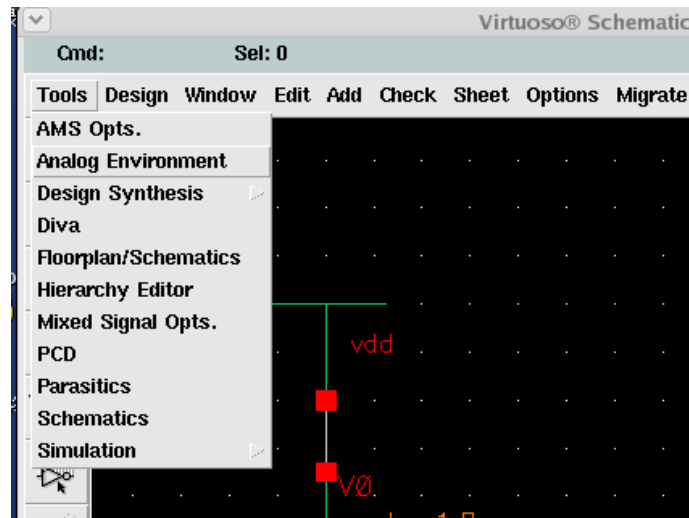
  

Property	Value	Display
Model name	pch	off
description	inal VT PMOS transistor	off
I (M)	180n M	off
w (M)	2u M	off
total_width(M)	2u M	off
Number of Fingers	1	off
Multiplier	1	off
total m	1	off

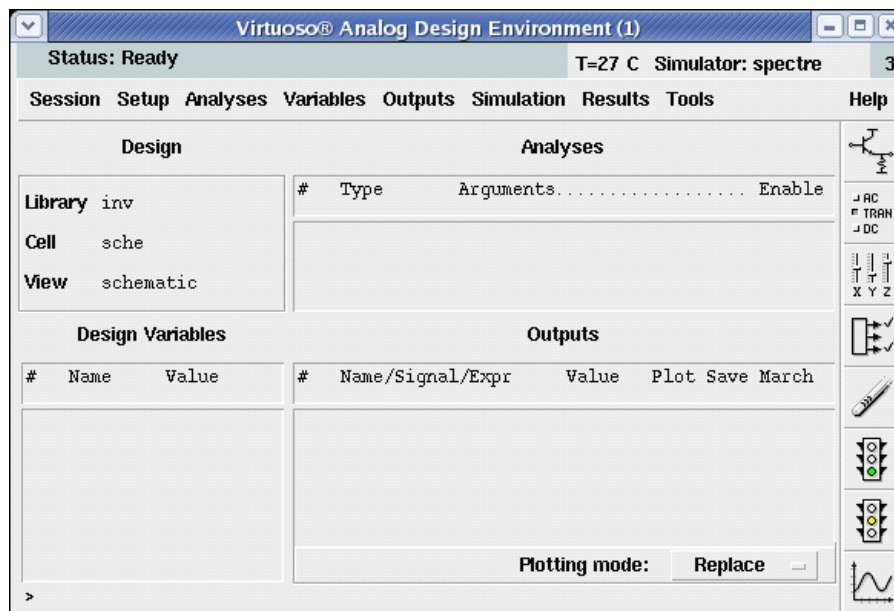
主要是设置宽和长，这里不改。  
到这里参数是设置就结束了，原理图就画完了，记得按 x 键检查并保存。

### （五）原理图仿真

打开仿真窗口，从原理图窗口中点击 Tools 的 Analog Environment，如下图：

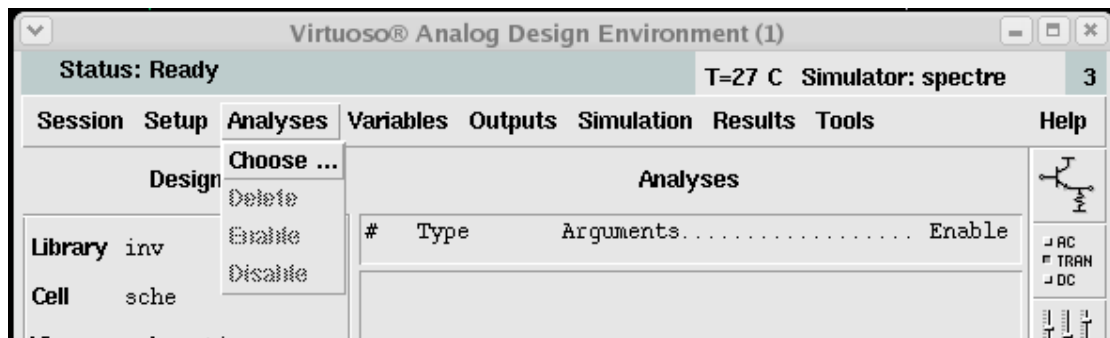


出现如下窗口：

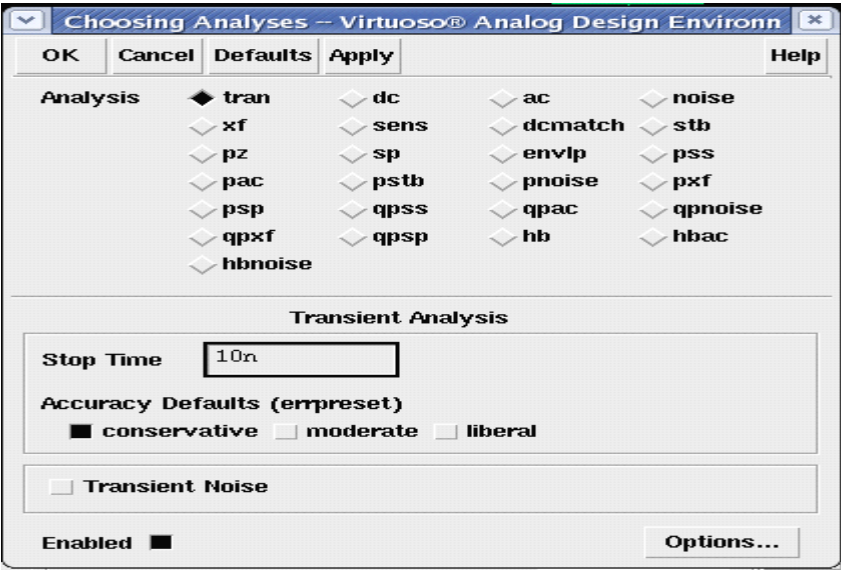


这里面是主要的，所有仿真都在这里设置，各个功能的介绍这里就不介绍了，参加其他教程。在我们的例子中，我们需要的是瞬态仿真。

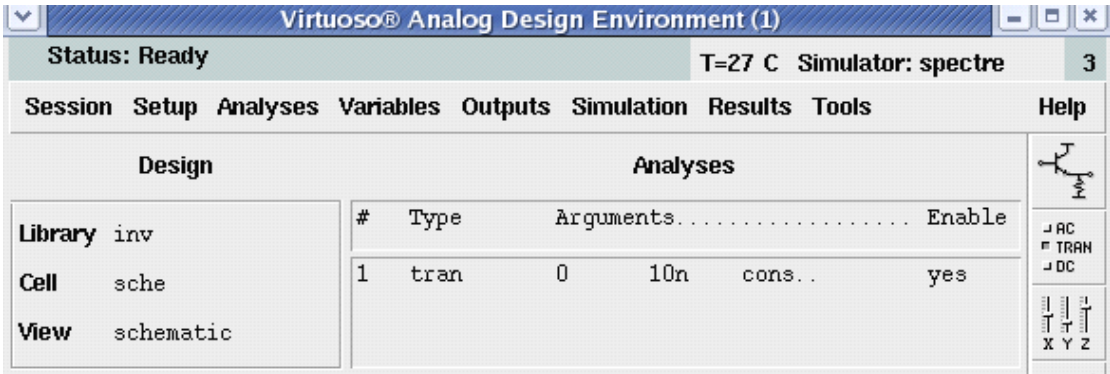
下面设置瞬态仿真，点击 Analyses 的 Choose...，如下图：



将出现下面的对话框:

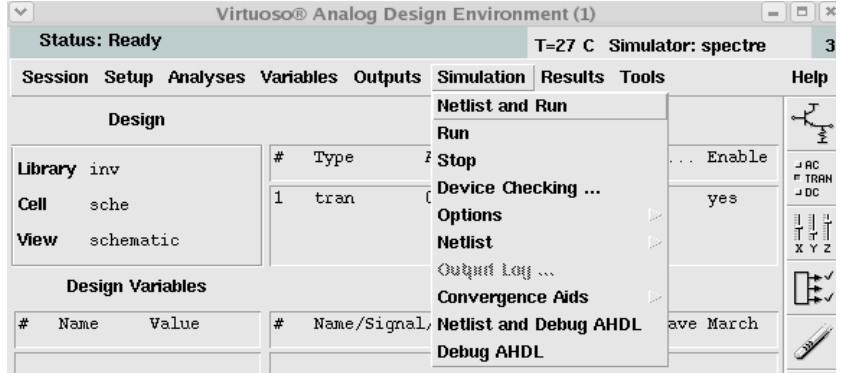


这里有很多的仿真模式，我们使用的瞬态仿真是 tran，其他的我也不介绍了，Stop time 填 10n，因为上面信号源的设置是 2ns 的周期，显示 5 个周期就差不多了。然后选中 conservative，表明是最高精度的分析，然后选中 Enabled，点击 OK 退出。可以发现仿真窗口发生了变化如下：



我们的仿真设置就这样好了，下面开始仿真。

点击 Simulation 的 Netlist and Run，如下图：



它有可能出现一个介绍性的东西，也有可能不出现，如果出现了点击 OK 就可以了，完了后会出现一个仿真的信息窗口，如下图：

```

/home/iroi/simulation/sche/spectre/schematic/psf/spectre.out
File Help 4

reltol = 100e-06
abstol(V) = 1 uV
abstol(I) = 1 pA
temp = 27 C
tnom = 27 C
tempeffects = all
errpreset = conservative
method = gear2only
lteratio = 10
relref = alllocal
cmin = 0 F
gmin = 1 pS

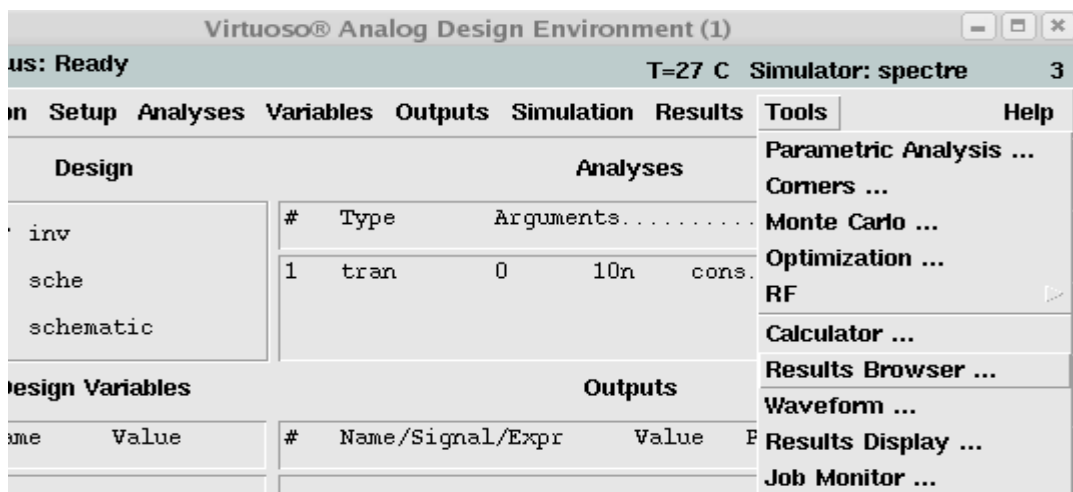
tran: time = 350 ps      (3.5 %), step = 100 ps      (1 %)
tran: time = 850 ps      (8.5 %), step = 100 ps      (1 %)
tran: time = 1.262 ns    (12.6 %), step = 100 ps      (1 %)
tran: time = 1.762 ns    (17.6 %), step = 100 ps      (1 %)
tran: time = 2.291 ns    (22.9 %), step = 87.88 ps    (879 m%)
tran: time = 2.791 ns    (27.9 %), step = 100 ps      (1 %)
tran: time = 3.307 ns    (33.1 %), step = 100 ps      (1 %)
tran: time = 3.807 ns    (38.1 %), step = 100 ps      (1 %)
tran: time = 4.291 ns    (42.9 %), step = 87.89 ps    (879 m%)
tran: time = 4.791 ns    (47.9 %), step = 100 ps      (1 %)
tran: time = 5.307 ns    (53.1 %), step = 100 ps      (1 %)
tran: time = 5.807 ns    (58.1 %), step = 100 ps      (1 %)
tran: time = 6.291 ns    (62.9 %), step = 87.89 ps    (879 m%)
tran: time = 6.791 ns    (67.9 %), step = 100 ps      (1 %)
tran: time = 7.307 ns    (73.1 %), step = 100 ps      (1 %)
tran: time = 7.807 ns    (78.1 %), step = 100 ps      (1 %)
tran: time = 8.291 ns    (82.9 %), step = 87.89 ps    (879 m%)
tran: time = 8.791 ns    (87.9 %), step = 100 ps      (1 %)
tran: time = 9.307 ns    (93.1 %), step = 100 ps      (1 %)
tran: time = 9.807 ns    (98.1 %), step = 100 ps      (1 %)
Number of accepted tran steps = 810
Initial condition solution time = 12.6331 ms.
Intrinsic tran analysis time = 276.685 ms.
Total time required for tran analysis 'tran' was 568.787 ms.

finalTimeOP: writing operating point information to rawfile.
modelParameter: writing model parameter values to rawfile.
element: writing instance parameter values to rawfile.
outputParameter: writing output parameter values to rawfile.
designParamVals: writing netlist parameters to rawfile.
primitives: writing primitives to rawfile.
subckts: writing subcircuits to rawfile.

```

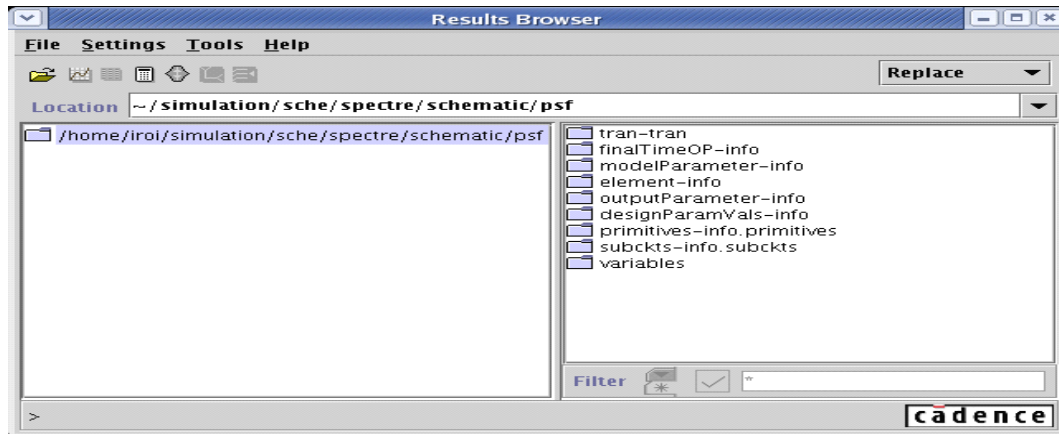
如果一切顺利就会出现上面这个东西，如果你的库没有安装好，会在这个里面报错。  
仿真结束了，然后就是看波形了。

点击 Tools 里面的 Results Browser，如下图：

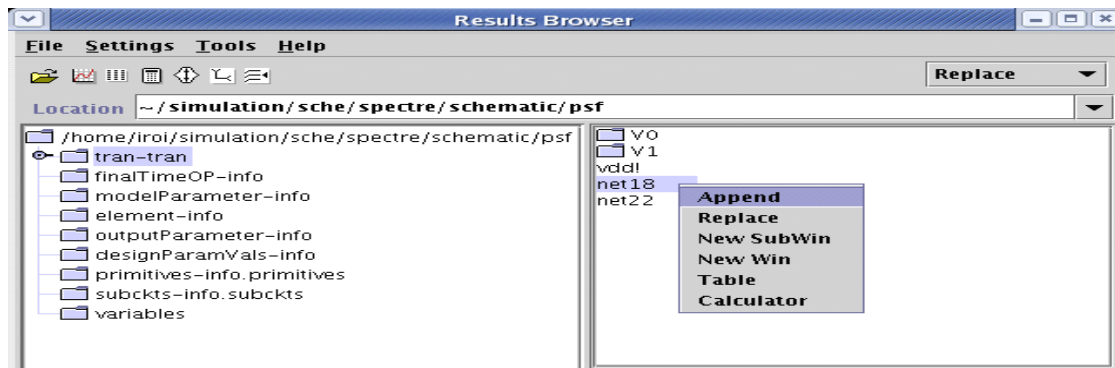


将出现如下图的 results 窗口：



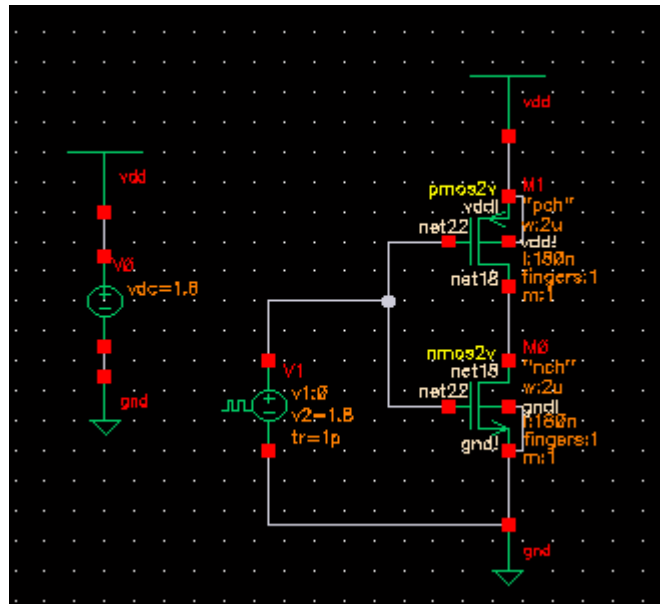


双击 tran-tran，出现如下：



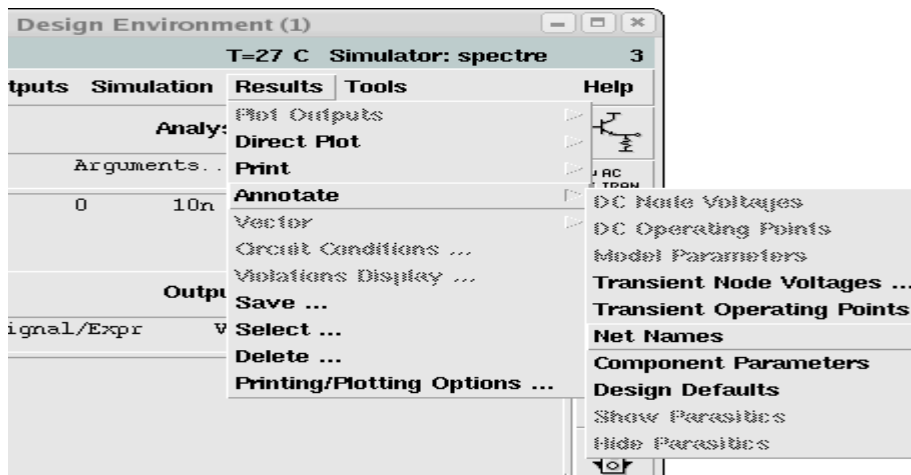
有个 net18 和 net22,当然这个名字可能各易。右键 net18 会出现上图的情况,其中 Append 表示在当前图形窗口中添加 net18 的图形, 如果当前没有图形窗口则显示 net18 的图形。其他的这里不介绍了。

下图中可以看到, net18 是非门的输出, net22 是输入, 也即信号。



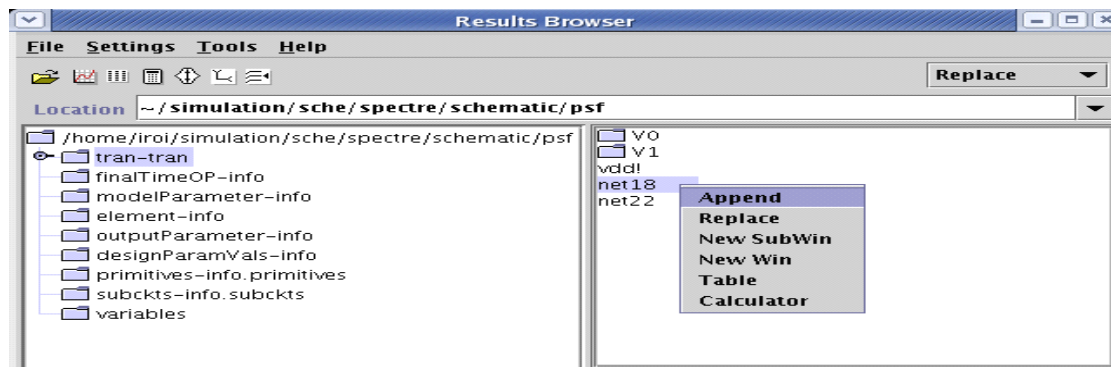
如果没有显示 net 信息, 这可以通过下面的方法显示。

点击 Results 的 Annotate 的 Net Names。

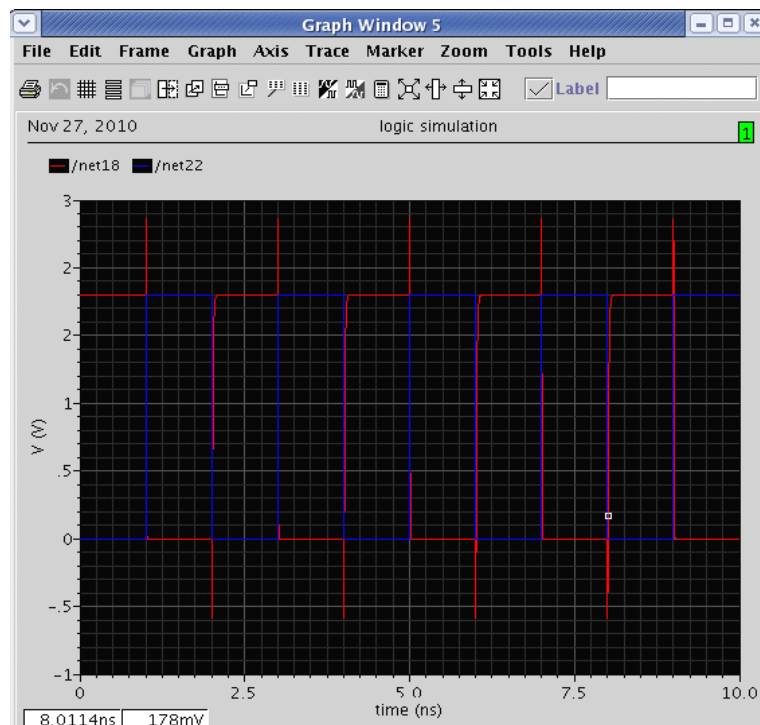


这里面是些显示的设置，具体这里不介绍了。


下面看波形：将 net18 和 net22 的波形都显示出来：

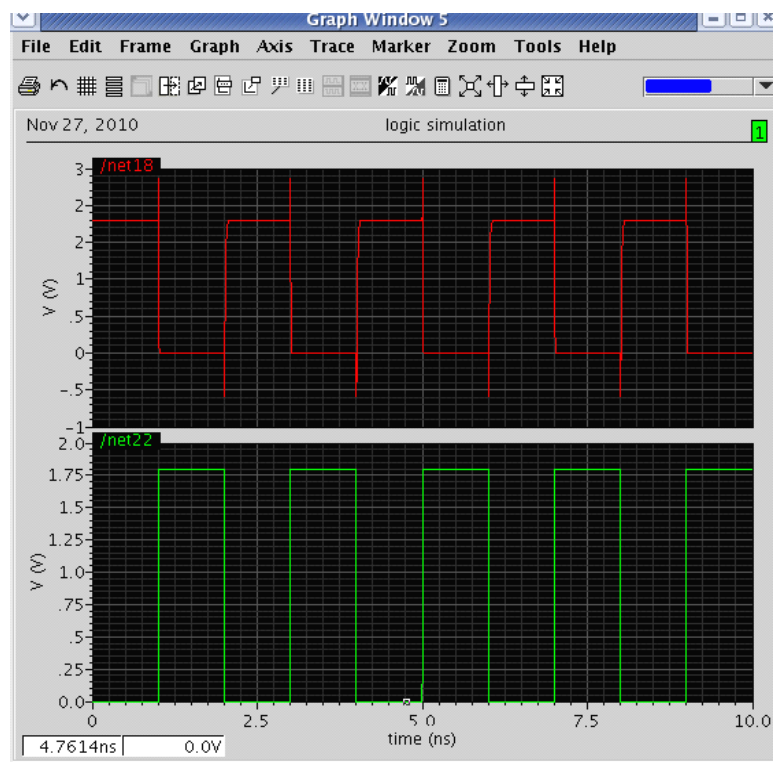


显示出来的波形如下：



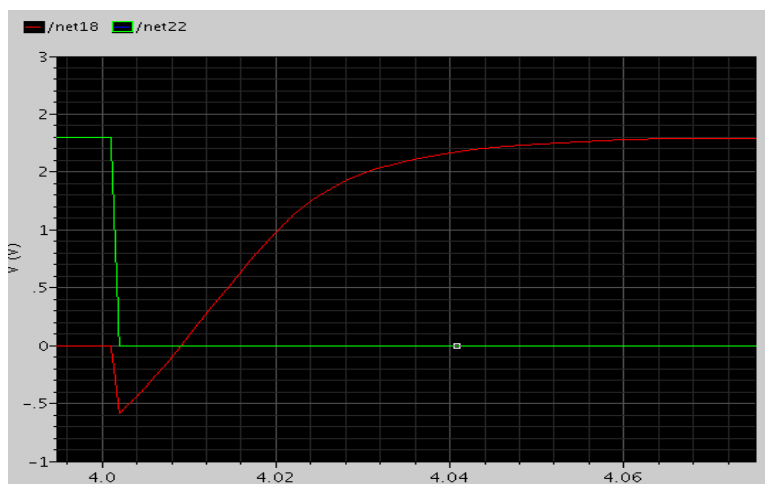
然后观察波形，看是否满足原理图的逻辑。

点击这个  可以得到下面的图形：



可以看得出，它符合一个非门的逻辑。

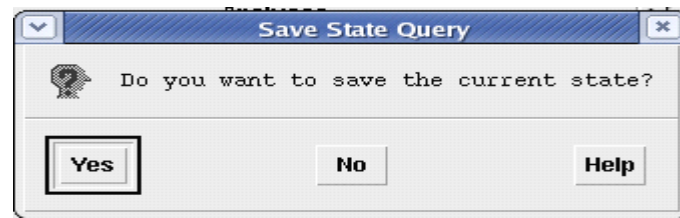
点击这个  可以将部分图形横向拉开，如下图：



可以看到，绿色的是输入的方波，下降沿的时间的 1ps，观测反相器的上升沿的时间。红色的输出在输入跳变时也有个相应的跳变，那是因为 mos 管 gd 之间的电容造成的，将输入信号耦合到了输出。如果输入信号的上升和下降时间设置慢些，这个现象就不是很明显了。

到此，原理图的仿真就完了。每次更改原理图后，仿真前都得 x（检查并保存），可以不 s（保存）。

在关闭仿真窗口 Analog Design Environment 时会提示是否保存当前状态, 选择保存就可以了:



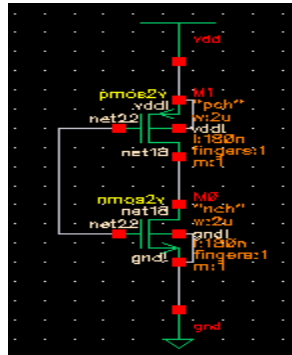
接下来还有一个, 也选 OK 就可以了。关闭过程有点慢, 要耐心等待。

## （六）生成 symbol

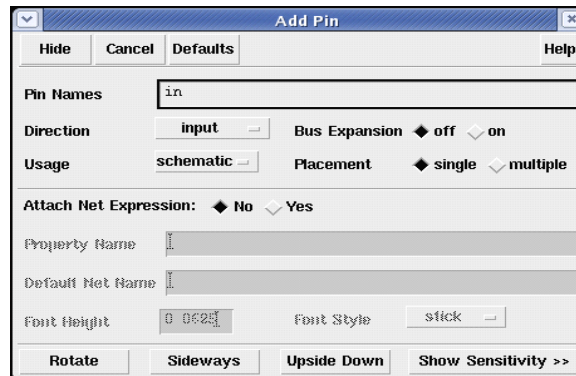
一般的，都会将一个模块做成一个 symbol，然后另外搭建测试电路，而且如果是要做版图，版图是不能将测试电路做进去的，也得将要做的版图部分做成一个模块，所以下面就将其改成 instance 的形式。

先将测试电路删掉，然后加入输入和输出的 pin。

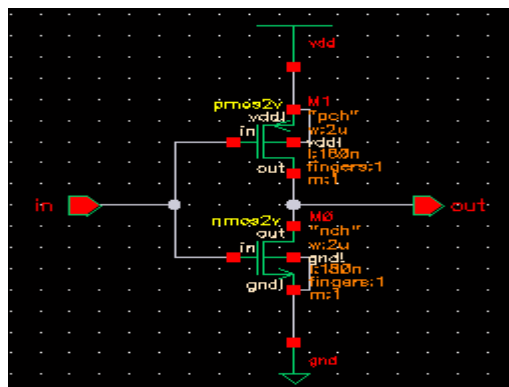
删掉后如下：



按键 p，弹出 pin 的对话框如下：

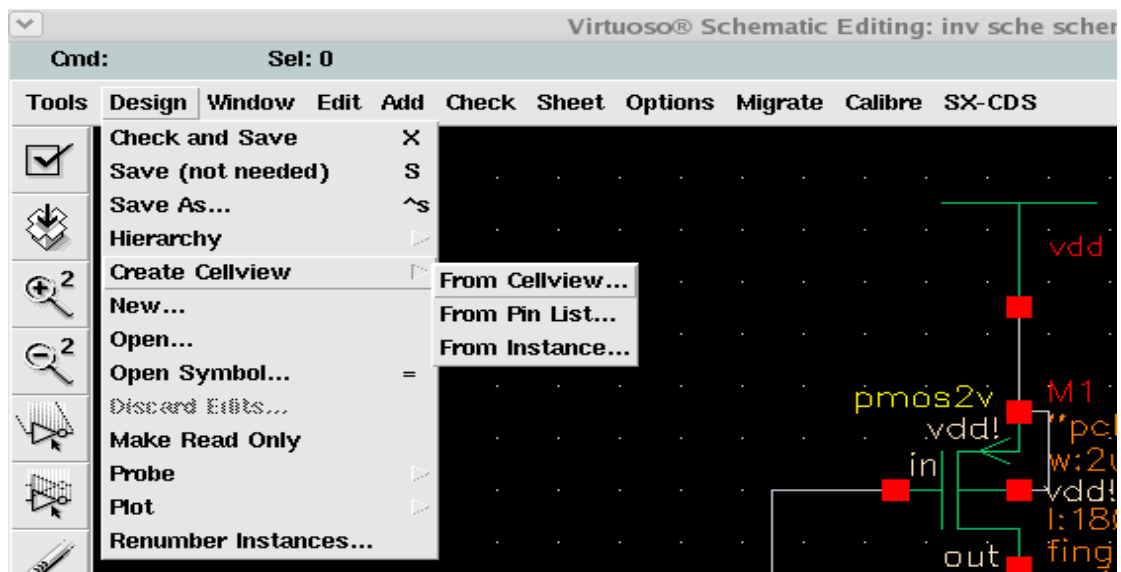


Pin Name 填 in，Direction 选 input，Attach Net Expression 选 no，然后点击 Hide，放置 pin 就可以了，同样放入 out，不过方向要选择 output。完了之后的图形如下：

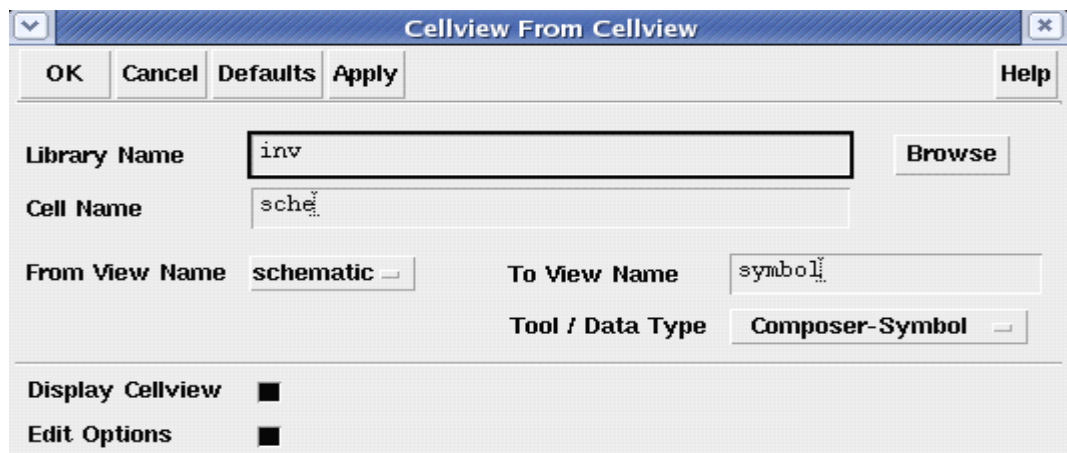


然后要把它做成一个 symbol，记得保存，不但要 x（检查并保存）也要 s（保存），否则 symbol 不一定同步。

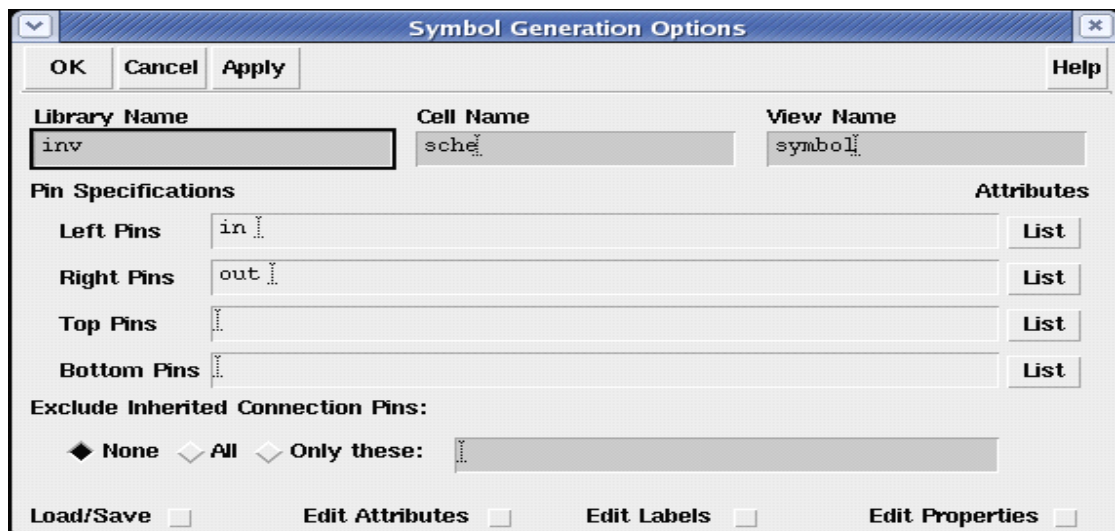
点击 Design 的 Create Cellview 的 From Cellview，如下图：



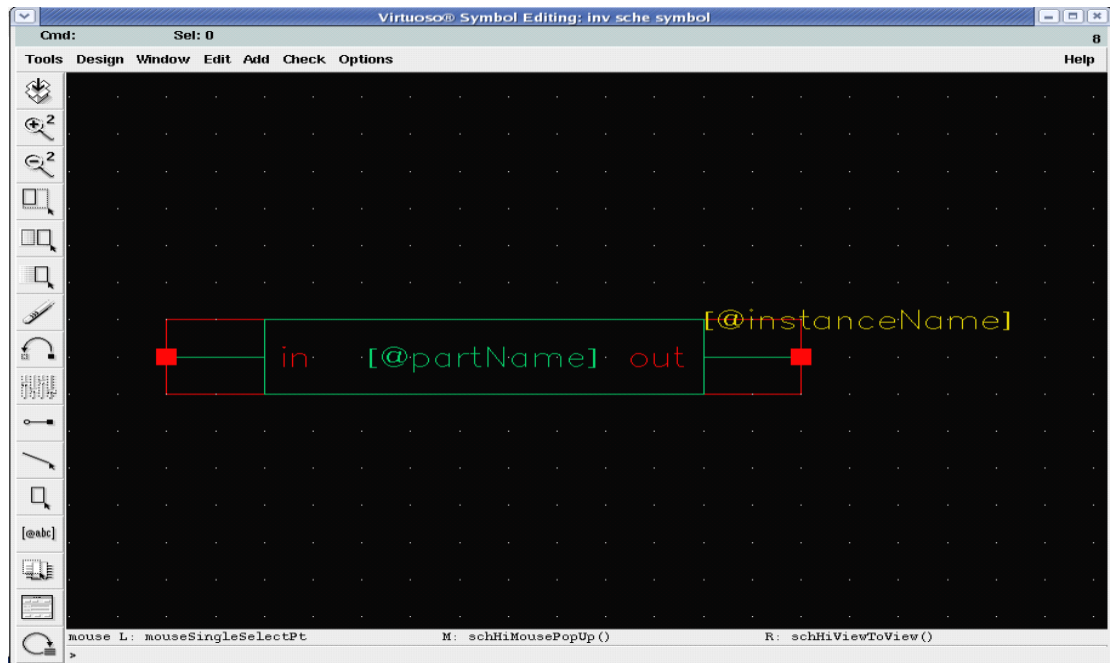
出现下图：



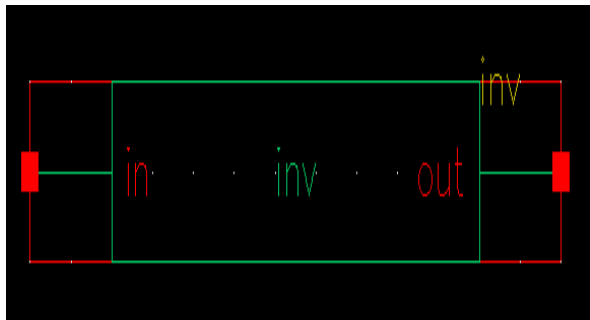
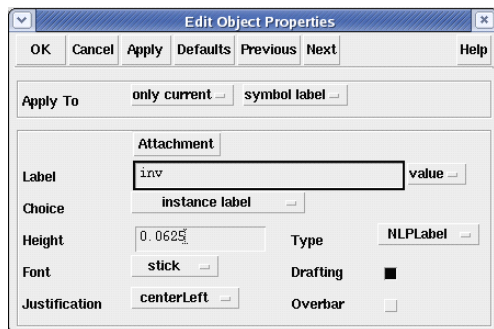
点击 ok，得下图：



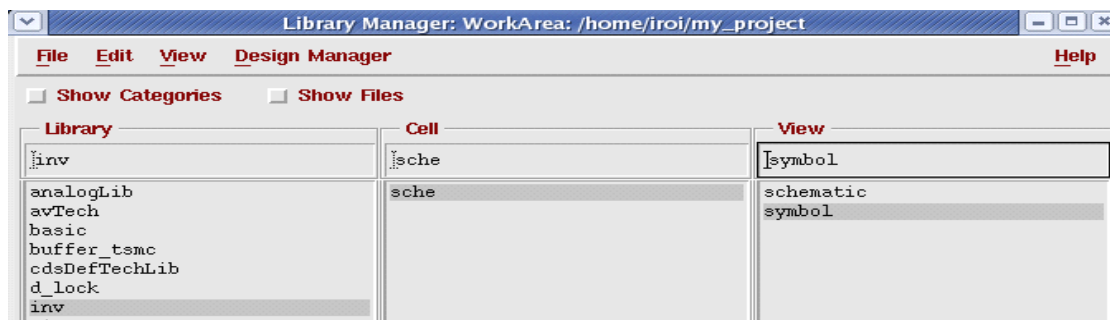
这里我们不需要更改，直接 OK，得下图：



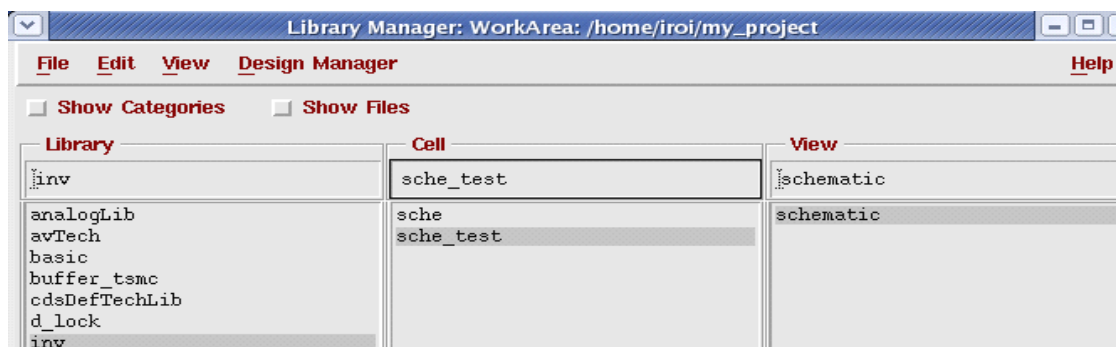
这就是封装后的 symbol，只有接口，其中 instance name 和 partname 可以更改，点击，按键 q，改为 inv。



这样，symbol 就建好了，记得保存。可以看到 Library Manager 里面多出来了一个 symbol。

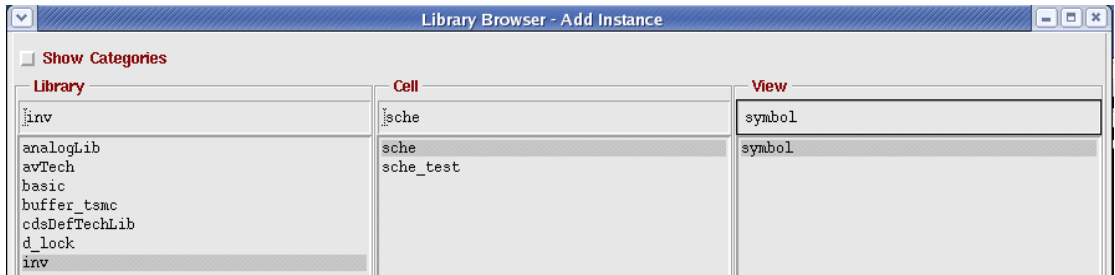


现在要建立测试电路，测试电路也是原理图，得另外新建一个 cell，建完后如下：

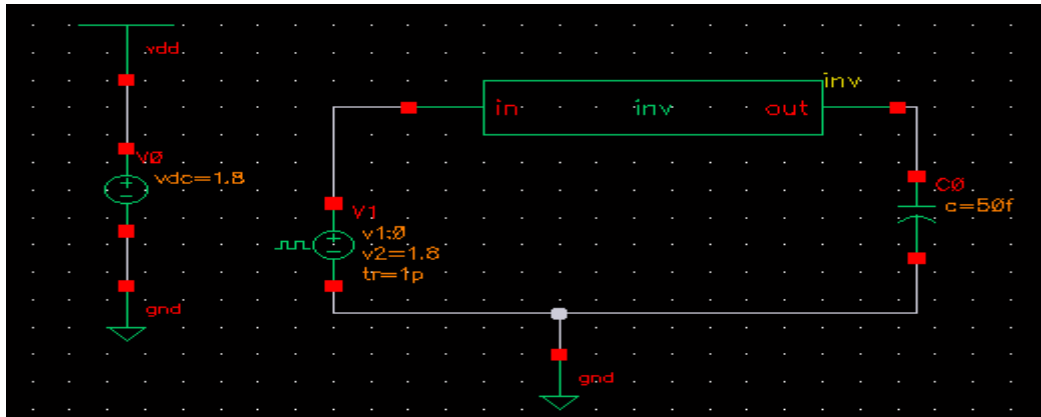




在测试电路 sche\_test 的 schematic 里面建立电路, 在这里要把我们刚生产的 inv 的 symbol 加进来, 过程和前面加 vdd 们的一样, 按键 i, browser, 然后选择 inv 的 symbol, 如下:



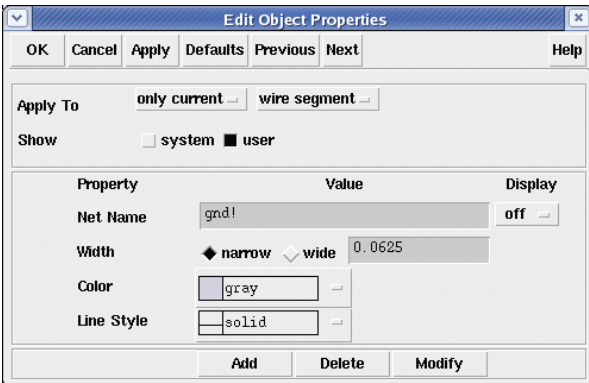
建立完了的测试电路如下:



在这里加了个负载电容, 容量为 50fF, f 是比 p 还小三个数量级的单位, 为 10 的-15 次方。其他的仿真和上面介绍的是相同的。

选中 inv, 按键 e, 点 OK, 可以进入其内部电路, ctrl+e 可以退回。

点击连接 gnd 的线, 然后按键 q, 看线的属性, 如下图:



可以发现, net name 是 gnd! 而不是 gnd, 这个很重要, 加! 说明是全局变量, 这样才能跟底层的 gnd! 连接在一起。同样, vdd!, 但是其他的都不是!

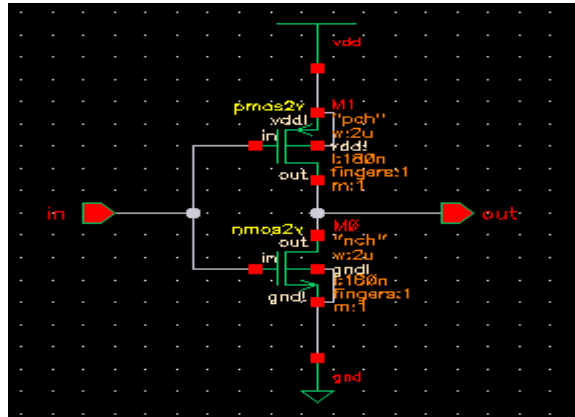
好了, 到这里原理图部分就介绍完了。下面介绍版图。

## （七）版图

### 7.1 生成版图

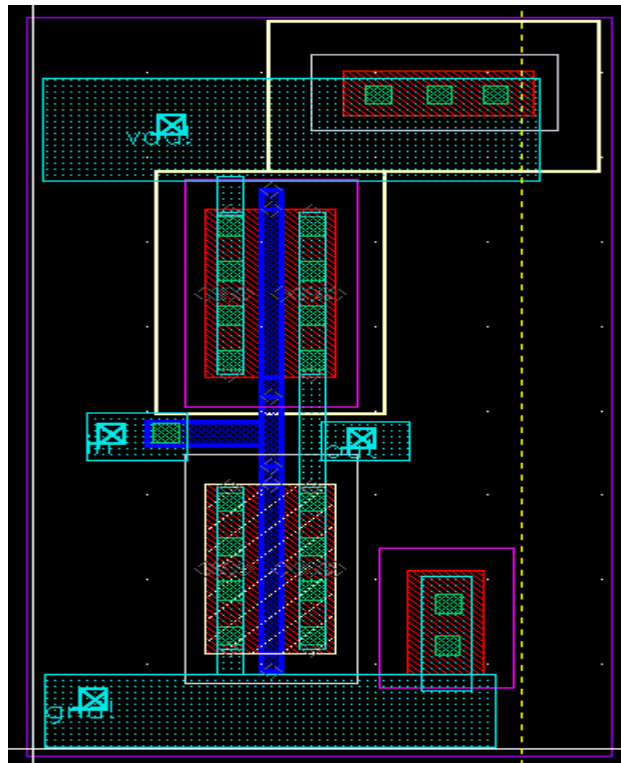
这里开始介绍版图。

我们是要画这个电路的版图：



其他的原理图都可以关掉。记得保存。

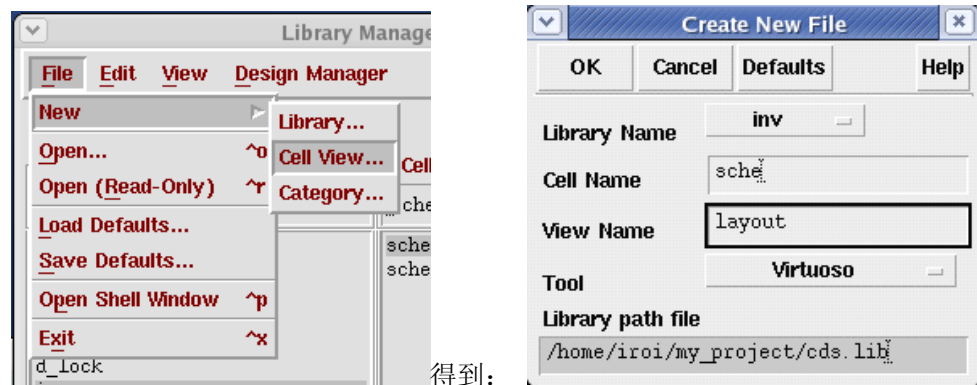
我们最终要完成的版图如下：



版图要注意文件结构，不要放错了，否则后面比较麻烦，现在文件结构如下：

Library Manager: WorkArea: /home/iroi/my_project		
File Edit View Design Manager Help		
<input type="checkbox"/> Show Categories <input type="checkbox"/> Show Files		
Library	Cell	View
inv	ishe	Ischematic
analogLib	sche	schematic
avTech	sche_test	symbol
basic		

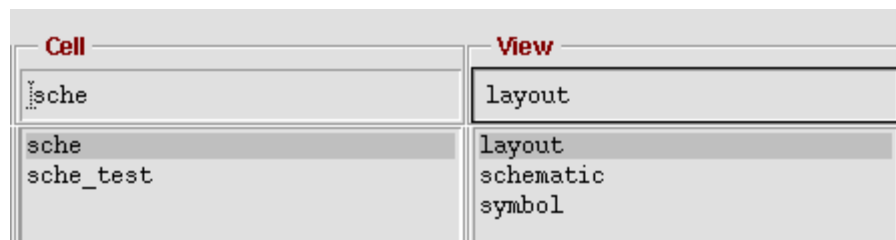
现在要新建一个版图的 Cell，最好是将版图文件和你要建立版图对应的原理图放在同一个 Cell 里面，过程如下：



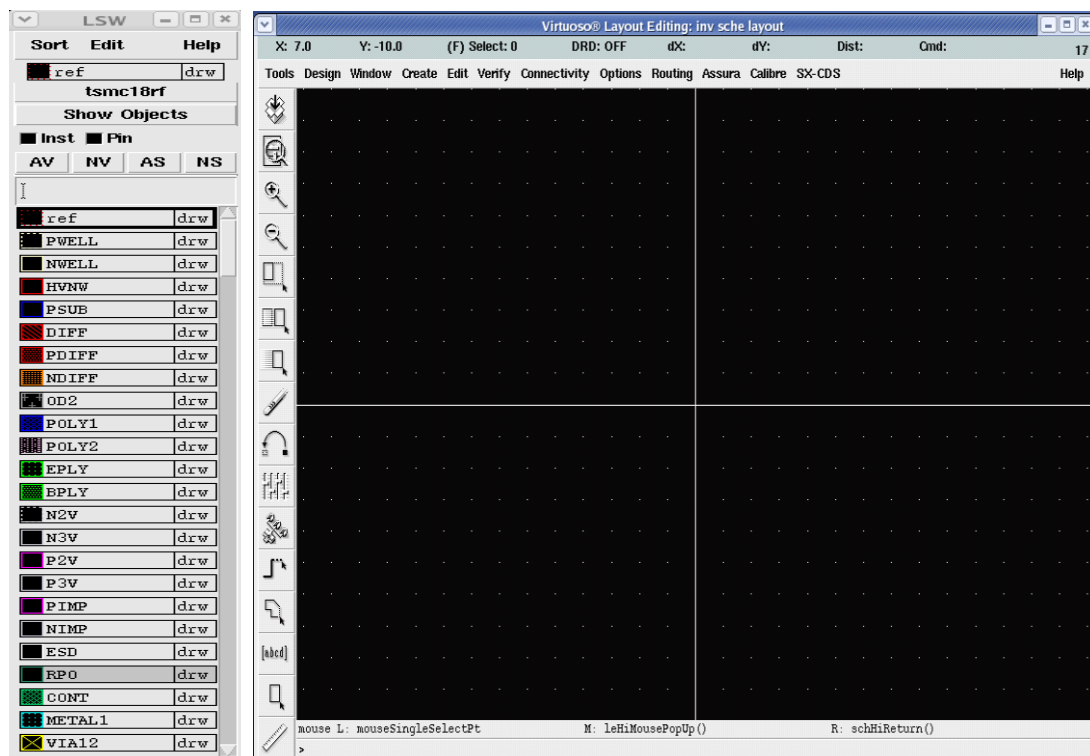
得到：

在上右图中，Cell Name 还是 sche 不变，但是 Tool 变了，改变了 Tool 后，view name 会自动改为 layout。点击 OK。会出现很多变化：

文件结构多了个 layout



而且出现了这两个：



左边表示版图的各个层，右边是画版图的区域。很有必要将左边的各层介绍介绍，已经假设你对 mos 管的物理结构比较了解。

PWELL 是 P 阱。

NWELL 是 N 阱。

DIFF 是扩散区，也叫有源区 active。

POLY1 是栅极。

PIMP 是选择进行 P 型扩散。

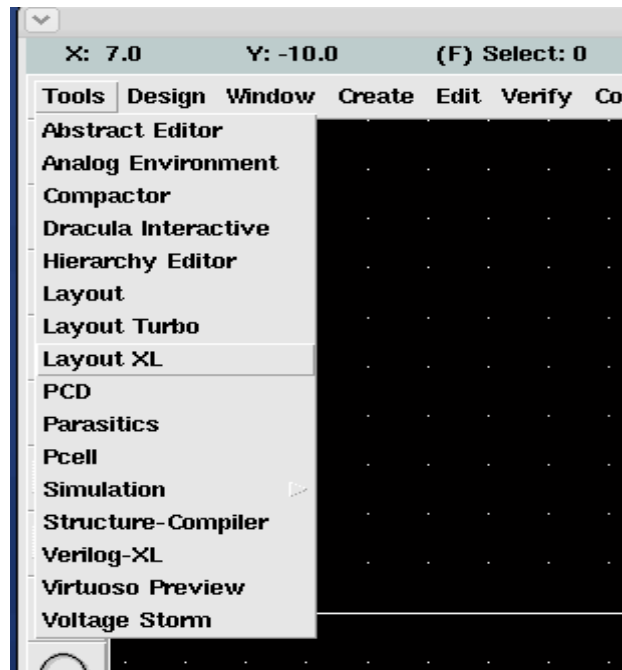
NIMP 是选择进行 N 型扩散。

CONT 是接触孔 contact。

MENTAL1 是金属 1 层。

其他的暂时用不到。关键是不同的工艺库，它的颜色和名字都不同，不能通用，所以这里比较困难。

我们使用另外一个工具来画版图，点击 tools 的 layout XL，如下图：



可能没有明显的变化，但是实质上是有变换的。

我们用软件自动生成的办法来做。点击 Design 的 Gen From Source，如下图：



得到下图：

**Layout Generation Options**

OK Cancel Defaults Help

**Layout Generation**

Generate: ☒ I/O Pins ☒ Instances ☒ Boundary  
☐ Transistor Chaining ☐ Transistor Folding ☐ Preserve Mappings

**I/O Pins**

Apply Pin Type Layer / Master Width Height Num Create

Defaults: Geometric ☒ METAL1 pn 0.23 0.23 1

Select: Number Selected: 0 Add a Pin

Term Name	Net Name	Pin Type	Layer / Master	Width	Height	Num	Create
"gnd!"	"gnd!"	"Geometric"	("METAL1" "pin")	0.23	0.23	1	
"in"	"in"	"Geometric"	("METAL1" "pin")	0.23	0.23	1	
"out"	"out"	"Geometric"	("METAL1" "pin")	0.23	0.23	1	
"vdd!"	"vdd!"	"Geometric"	("METAL1" "pin")	0.23	0.23	1	

Update Pin Type Layer / Master Width Height Num Create

in in Geometric ☒ NWELL pn 0.86 0.86 1

Pin Label Shape: ☒ Label ☐ Text Display ☐ None Pin Label Options...

**Boundary**

Layer: ☒ prBound dg Left: 0

Shape: Rectangle Bottom: 0

**Boundary Area Estimation**

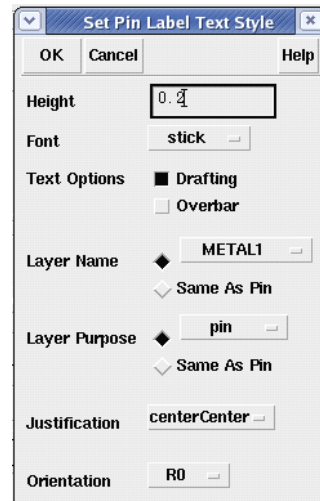
Utilization (%) 25 Aspect Ratio (W/H) 1

Area Calculation: PRBoundary Based

☐ Load Template File for Layout Generation

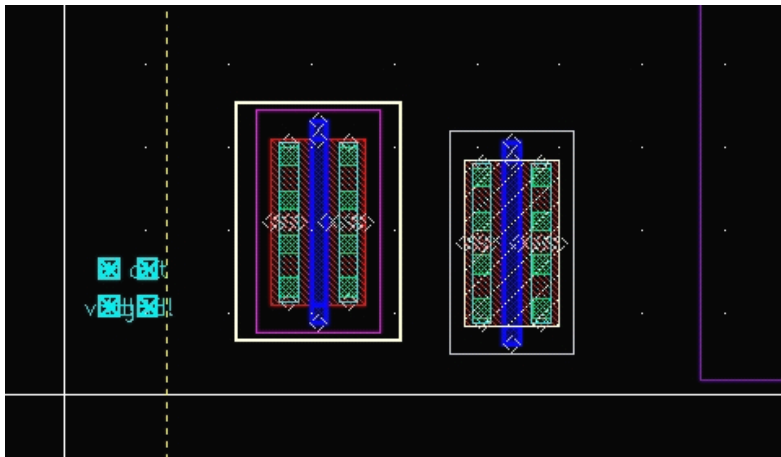
joke.lx Browse... Load

这个图中的设置很重要！这个是设置各个端口的。在 I/O pins 栏中，将 NWELL pn 改为 METAL1 pn，注意不要改错了，不是 METAL1 dg，然后点击 Apply，可以发现中间的 4 个端口参数都变了。另外在 Pin Label Shape 栏中，选中 Lable，点击后面的 Pin Label Options，打开下面对话框：



Height 为 0.2，这个是 Label 的大小，1 的话显示太大。Layer name 选择 METAL1。  
Layer Purpose 选择 pin。OK.

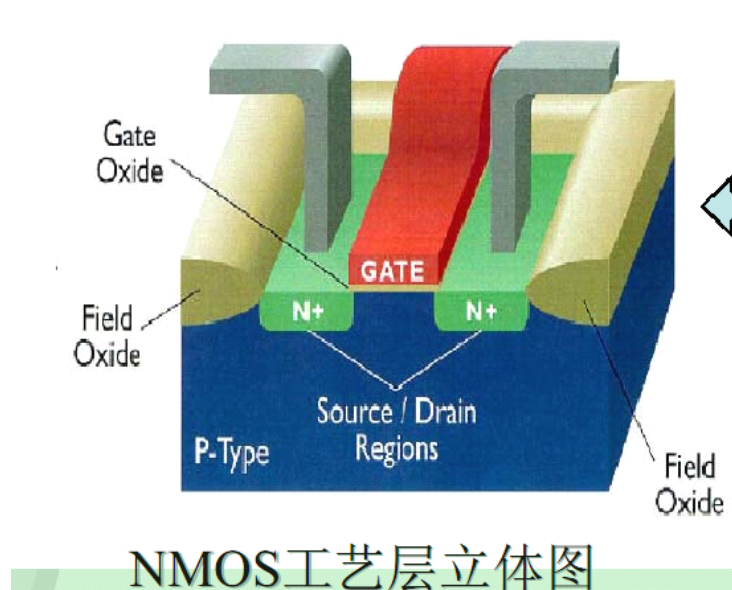
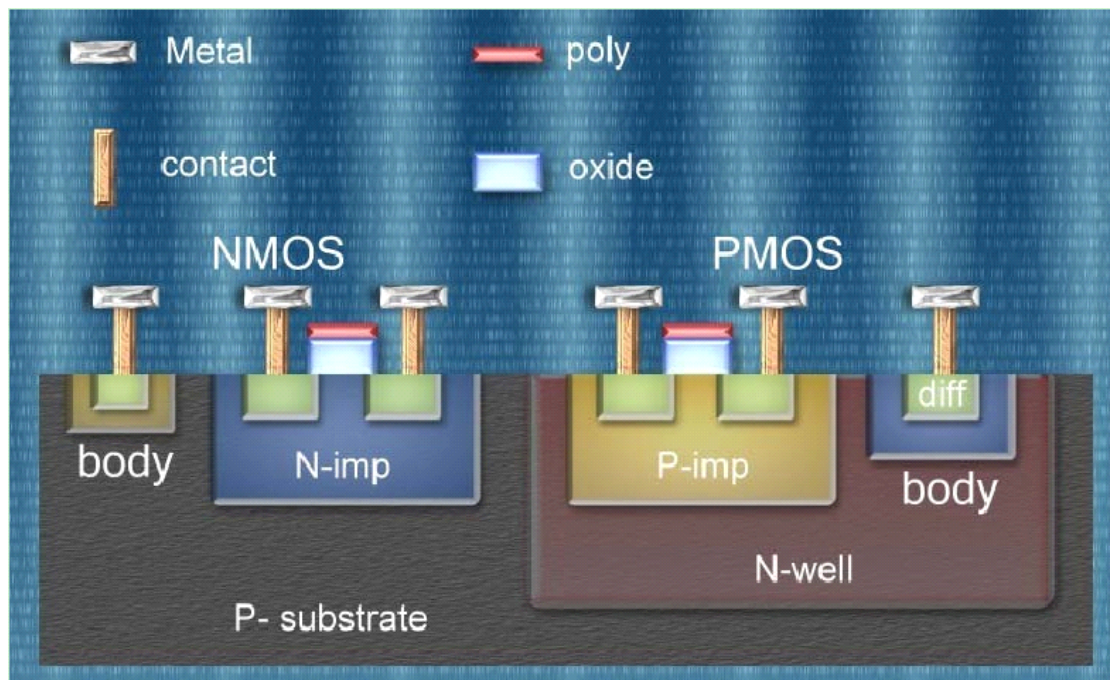
出现下面的版图：里面有 4 个引脚和两个 mos 管。



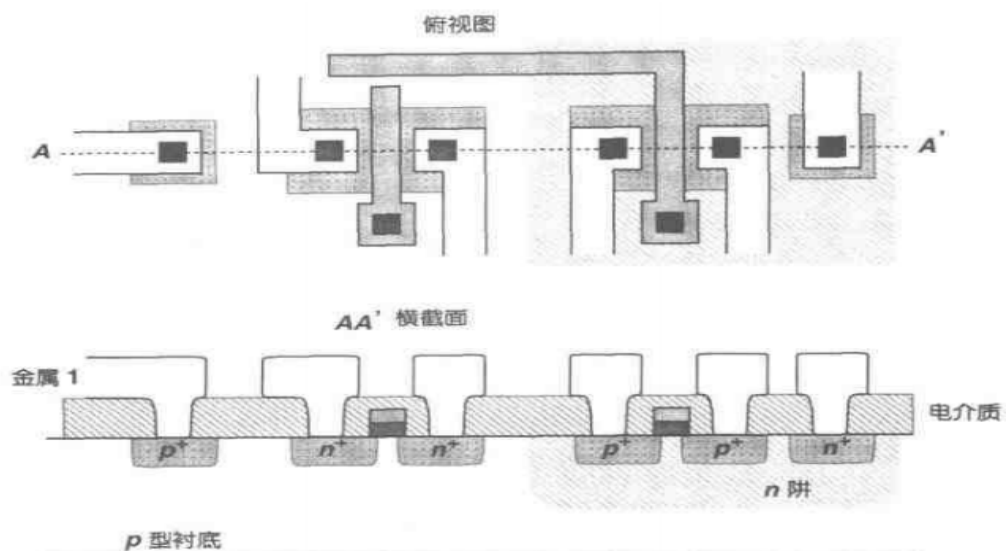
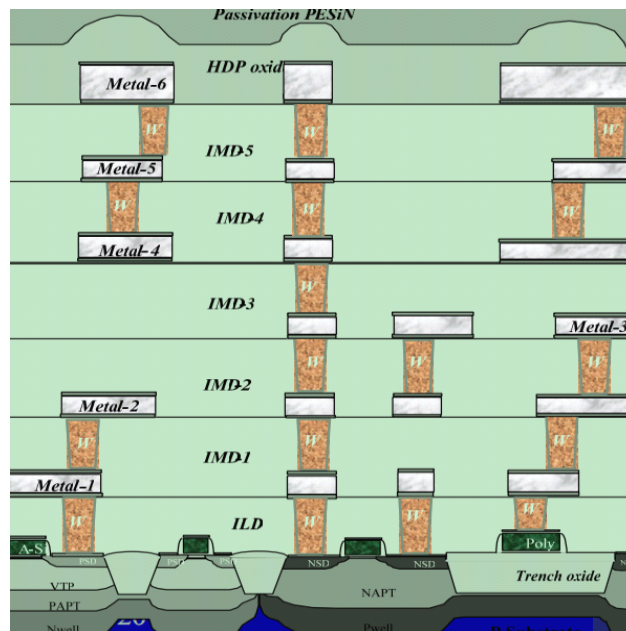
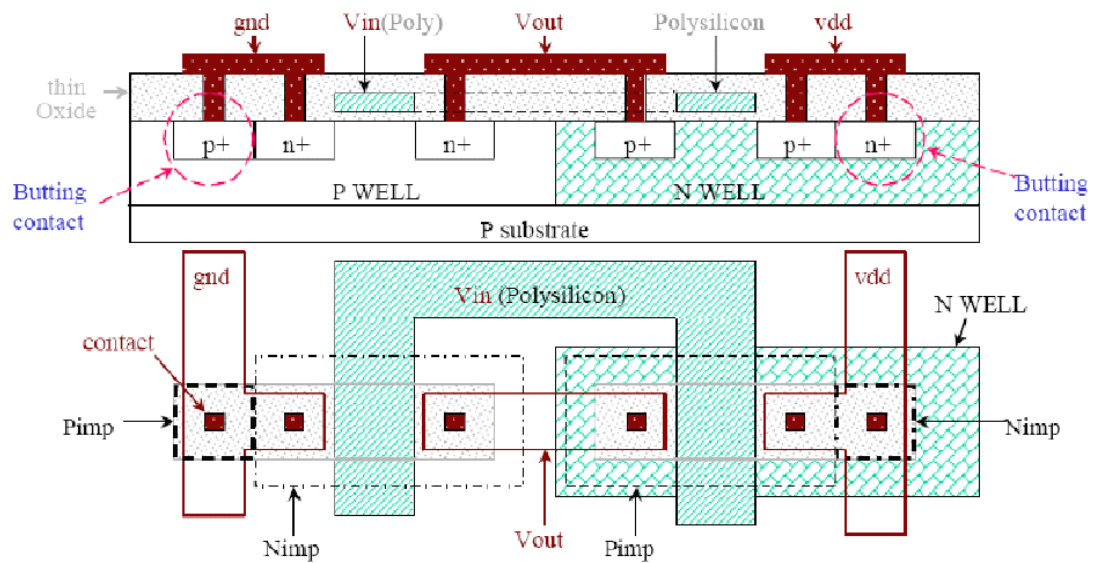


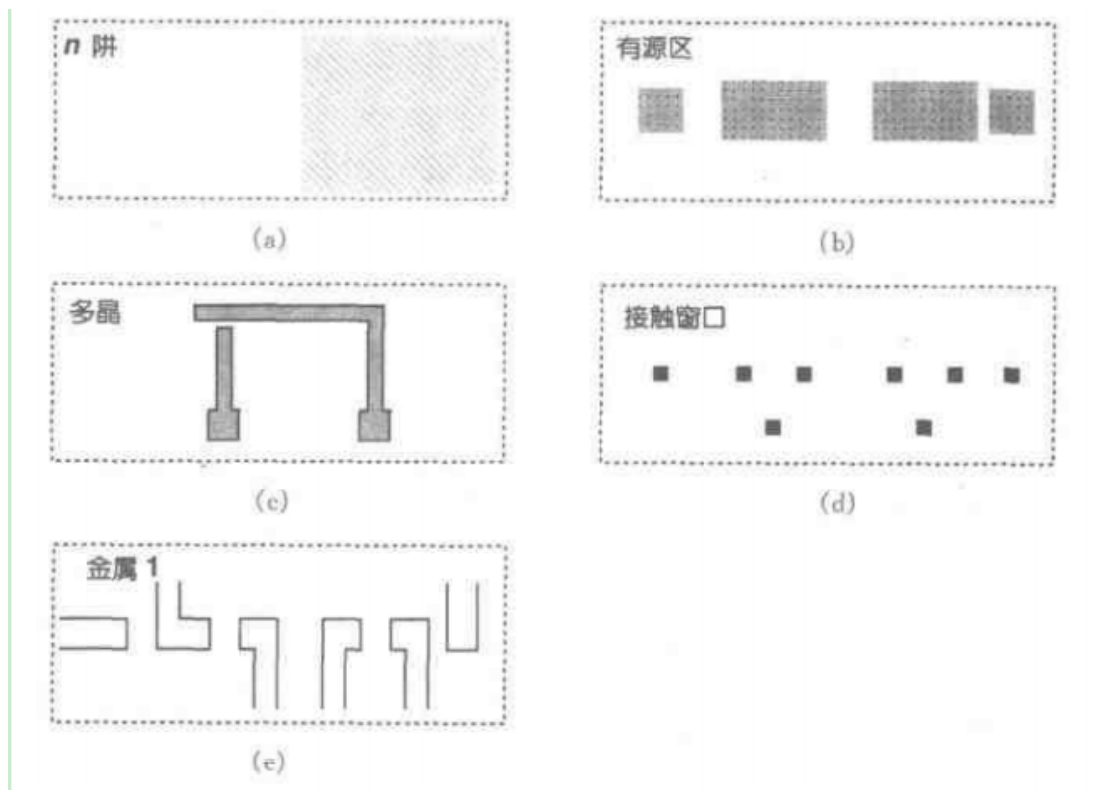
## 7.2 MOS 管结构介绍

下面要加入一些 mos 管结构的内容，完了后接着讲版图。为了能理解 mos 管的版图，我在这里贴出一些 mos 管版图的结构。每一个图都很有利于理解 mos 管的结构。









### 7.3 画版图

版图里面有的是要严格对齐，可以按键 **q** 在属性中更改 **xy** 坐标。

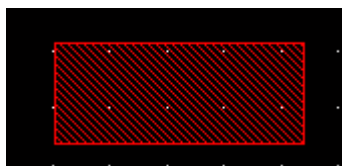
METAL1 有 **drw** 和 **pn** 之分，**drw** 是画金属层的，**pn** 是画引脚的，两个是同一个层的不同功能部分。

有些层必须相距一定的距离，有些层可以重叠，这个需要对各个层都了解。

虽然前面已经说过每个层干嘛用的，但是这里还是要再说一遍。



NWELL, N 阱, pmos 管做在 N 阱中。



DIFF 扩散区，或者是有源区，这个不分 P 型和 N 型。



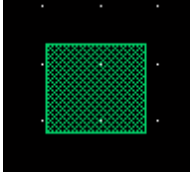
POLY1, 多晶, 做 mos 管的栅极。



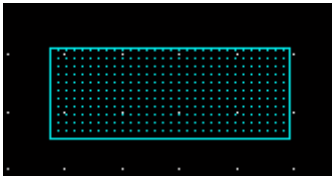
PIMP, 是说明里面的 DIFF 是 P 型扩散。跟 DIFF 联合使用。



NIMP, 是说明里面的 DIFF 是 N 型扩散。跟 DIFF 联合使用。



CONT, 连接孔, 这个是固定大小的矩形, 边长是 0.22um。

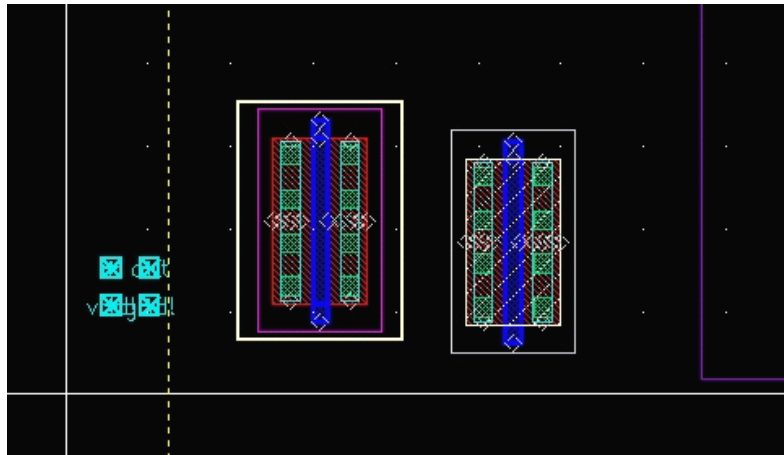


METAL1 drw, 金属 1 层, 画金属连线的。



METAL1 pn, 金属 1 层, 画金属引脚的。

下面接着讲版图:



先列出版图中使用的快捷键和某些设置。

Ctrl+z 放大

Shift+z 缩小

m 移动

q 属性

r 画矩形

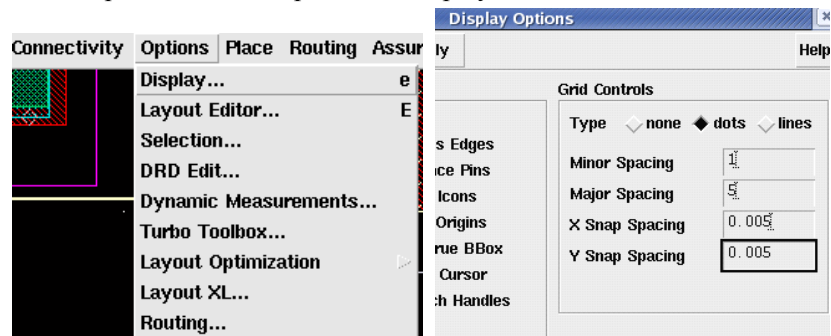
Shift+k 删除 ruler

c 复制

上、下、左、右 也可以使用

F4 可以在选中之间切换, 选中是整个方框还是选中某一边。

先设置一下 snap 的大小，在 Options 的 Display 中



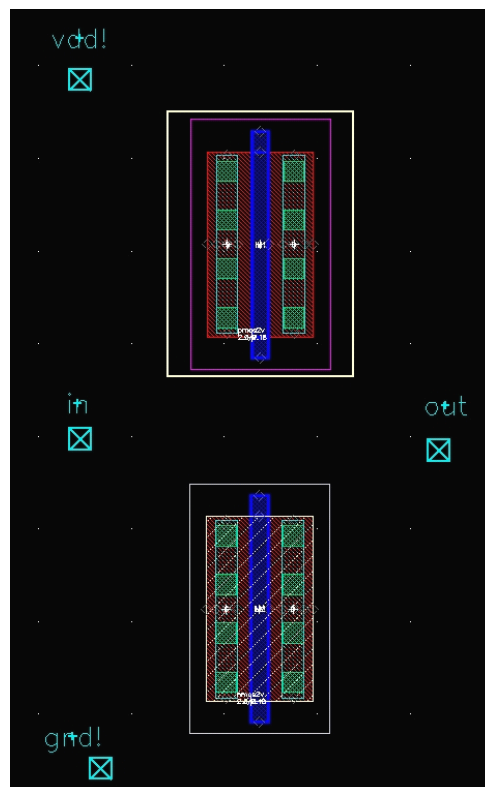
这是设置的是 0.005，有时候可能为了方便要改成另外的。



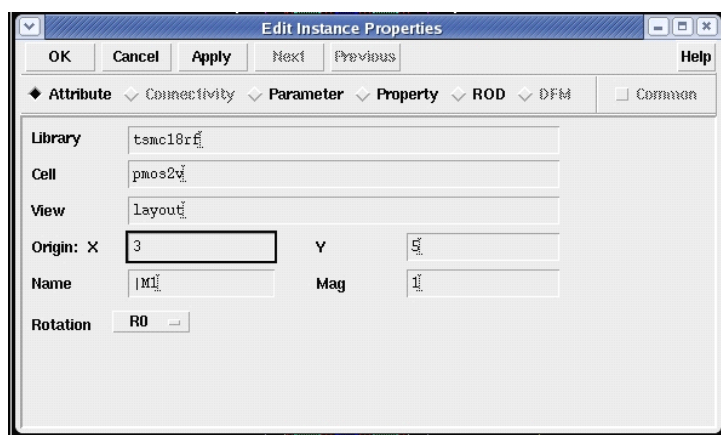
这个设置为 anyAngle 表明可以任意方向拖动，其他的有的是只有单方向拖动。OK 退出。

使用 m 键移动，可以先将紫色的框放一边，完了后再弄过来。（这一步最好要做，否则给后面带来不少麻烦）

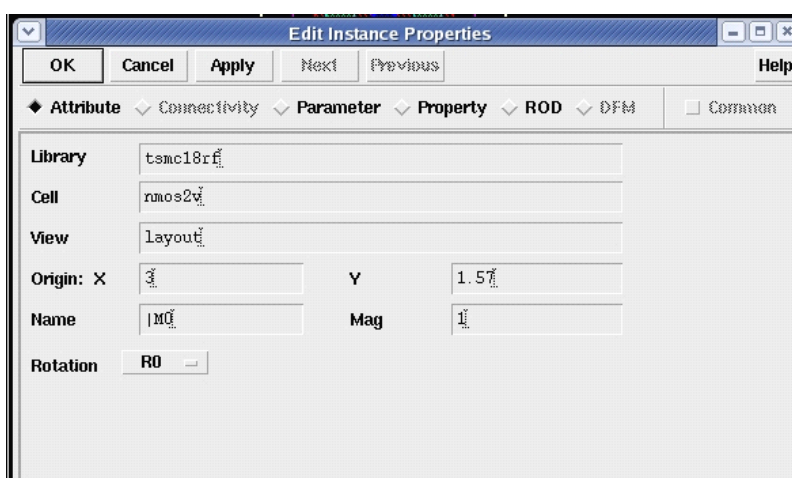
先通过移动，将他们的位置摆放好，如下图：



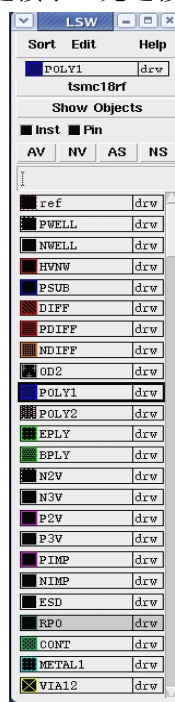
在这里，MOS 管的对齐是要求非常严格的，否则很容易在后面 DRC 检查错误。最好是使用坐标。选中 pmos 管，按 q 键，则出现它的属性窗口如下：



将 x 坐标改为 3，y 坐标可以随意。点击 OK。然后同样选择 nmos 管，按 q 键，将其 x 坐标改为 3，这样两个就严格对齐了，如下图：




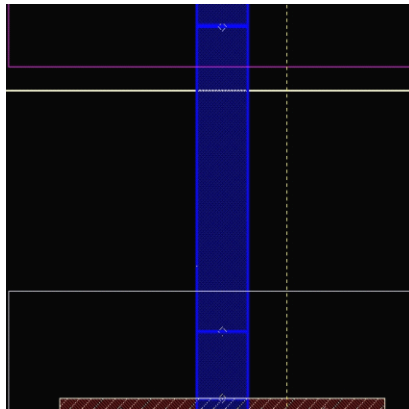
y 坐标是不要求的，上面那个值是否合适还不一定呢，也许到时候还要调整。然后就开始连接了，先连接栅极 poly，选择 poly 层：



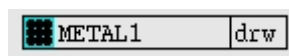
选择其中的 POLY1 层：



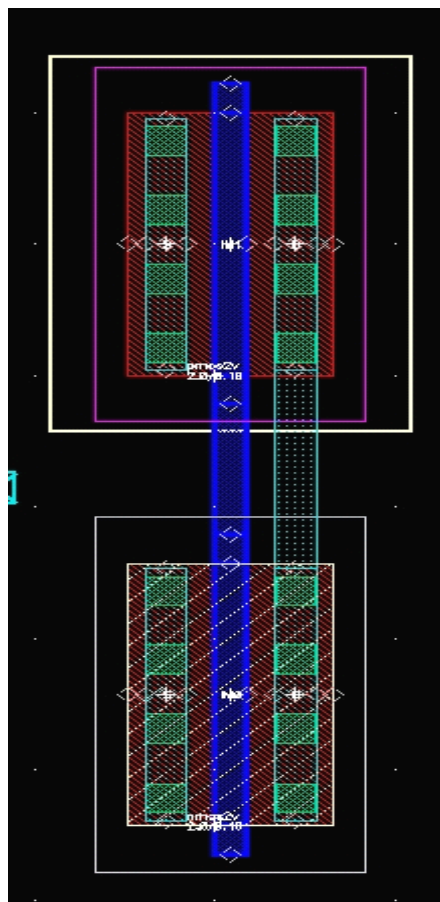
然后选择画方框：，将两个的栅连接起来，并用坐标将其对齐：



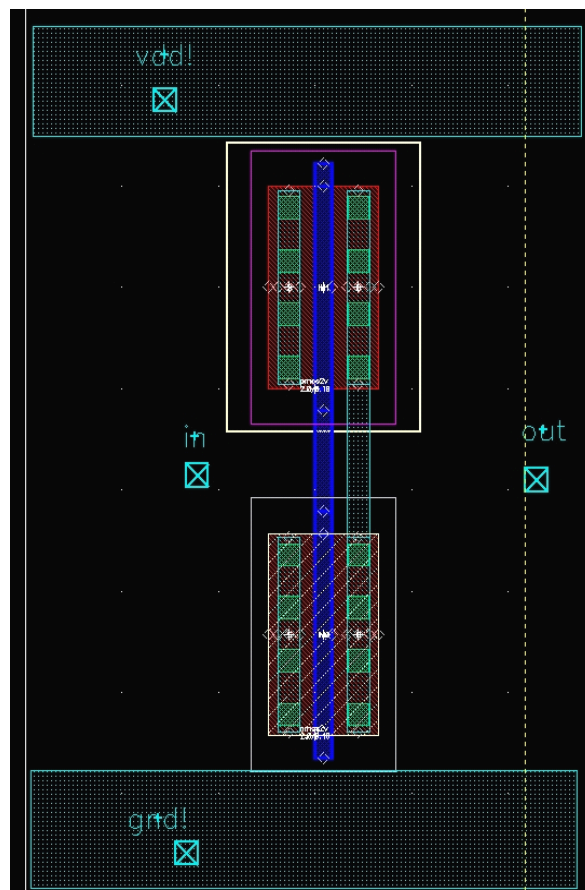
同样的方法将金属层连接好，金属层为：METAL 1 drw



这个没办法用坐标，只能细心的对齐了：

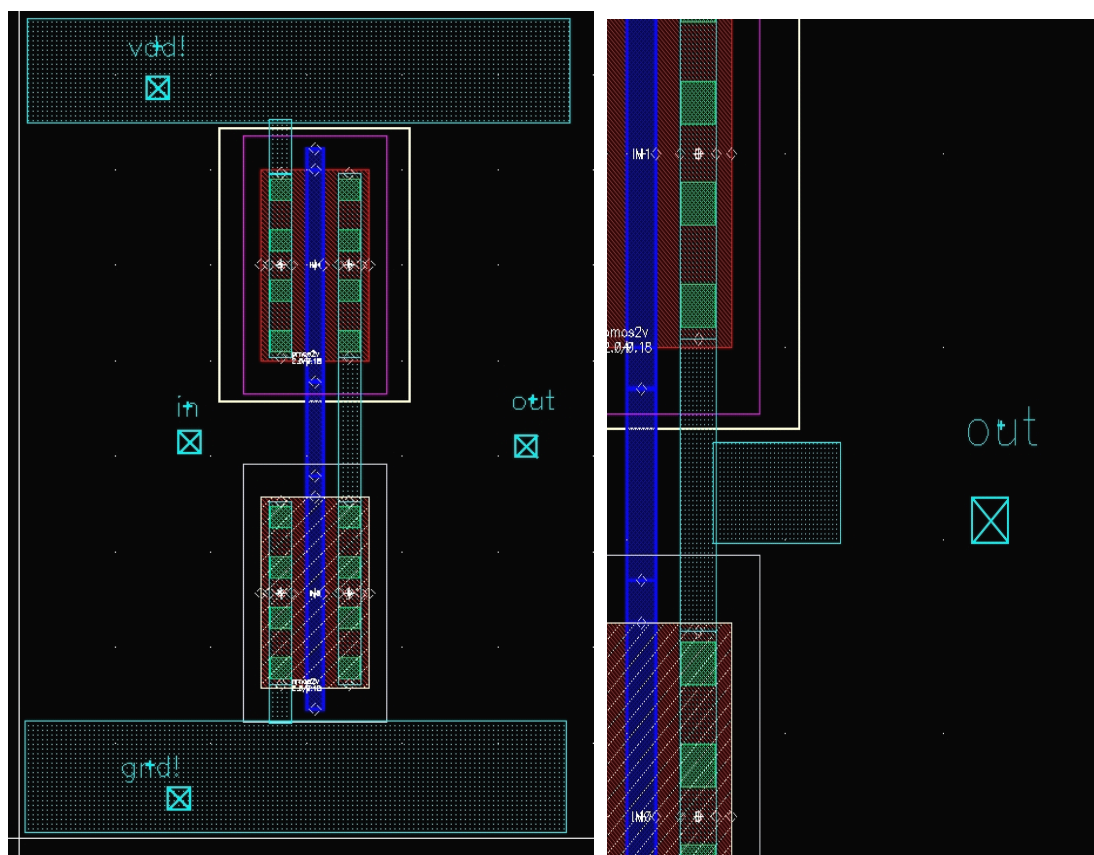


连接输出



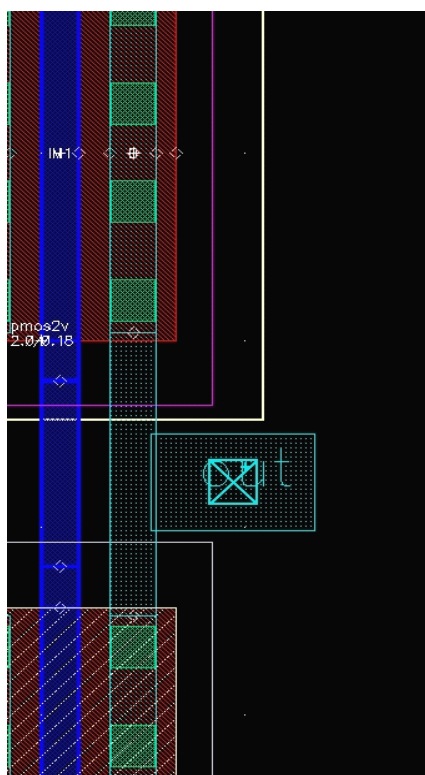
画出电源线和地线



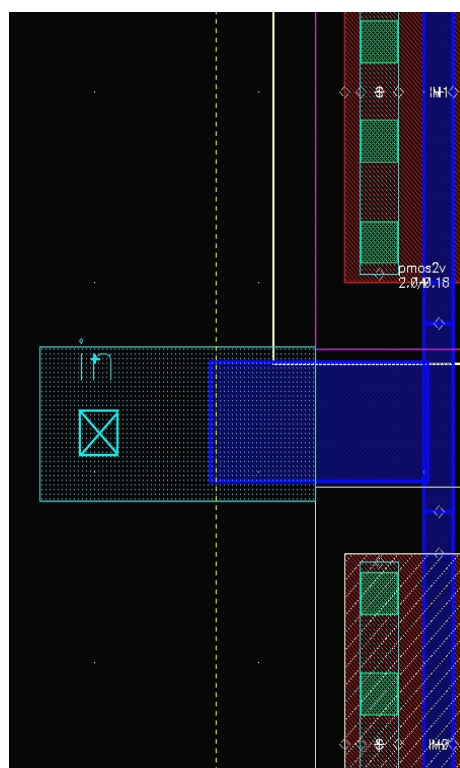


将 PMOS 源极接电源，NMOS 源极接地

画输出端口



将输出引脚放置上去，一定要把 out 也放在金属上



画输入端，并把 in 放上去，注意当前的是哪一层

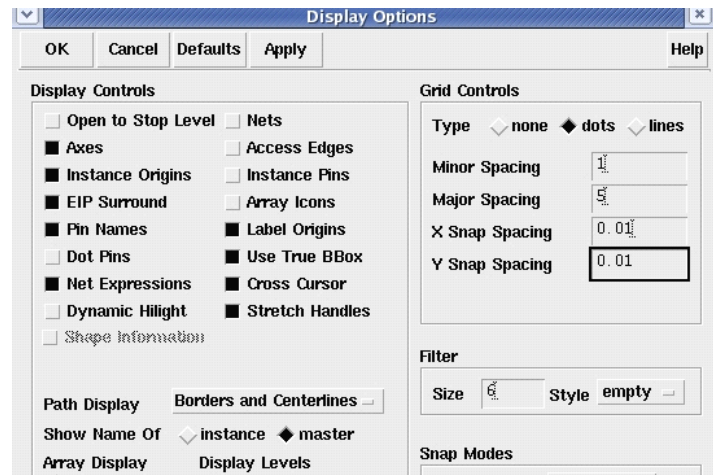


上面的图中可以看到，输入端金属层和栅极有重叠的部分，但是两个还没有连接起来，要用个过孔将两者连接，过孔是 CONT 层：



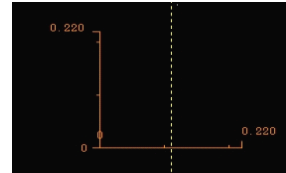
的  $0.22 \times 0.22$ ，所以这里要用的尺子 ruler：

先将 snap 的大小改下，否则画 0.22 不好控制，我们现在的精度是 0.005，改为 0.01，前面介绍有，或者按 e 键得到：

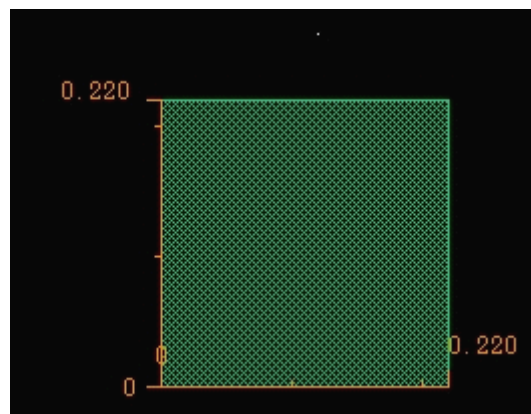


X 和 Y 都改为 0.01 就可以了。

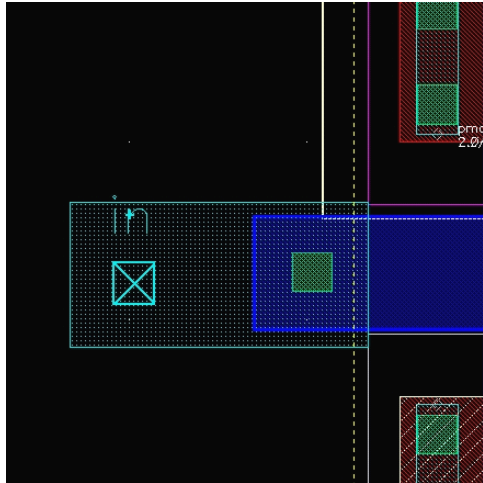
使用 ruler 画：最好尽量放大一点，好控制些。



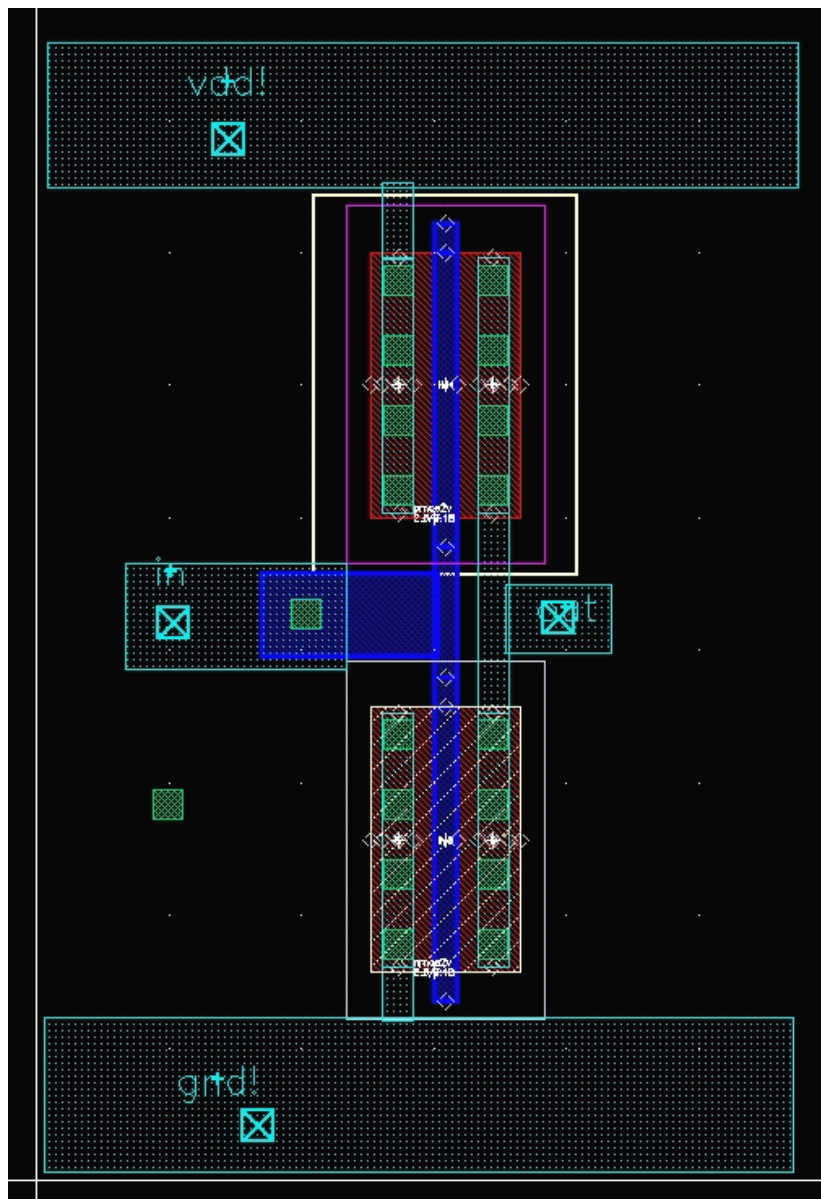
然后在 CONT 层，画下图：



再按 shift+k 键就得到了  $0.22 \times 0.22$  的过孔，再把它复制到那个重叠的地方去：

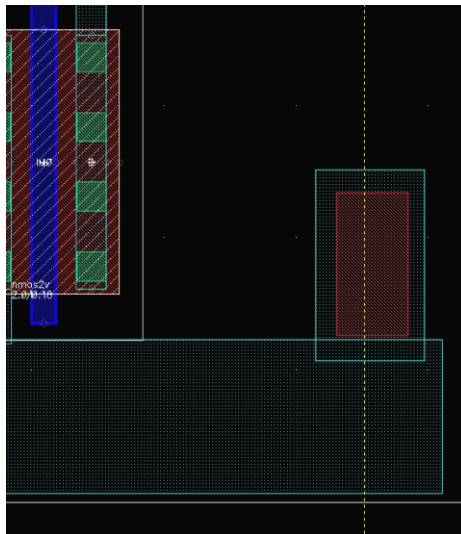


到现在为止，图示如下：

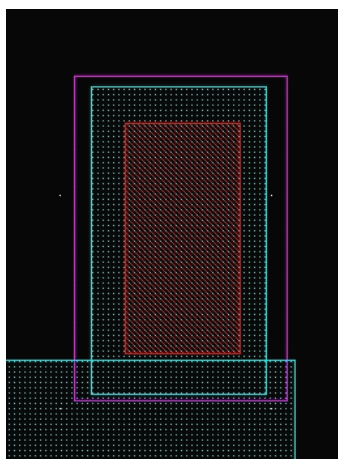


还有就是 NMOS 的衬底接地和 PMOS 的衬底接电源没画。想想他们的衬底都是接在哪里的？一端接地或电源，另一端呢？

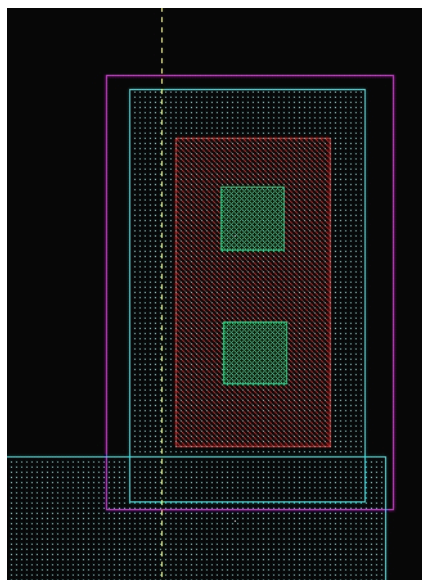
画金属层和扩散区（也叫有源区），扩散区是 DIFF 如下：



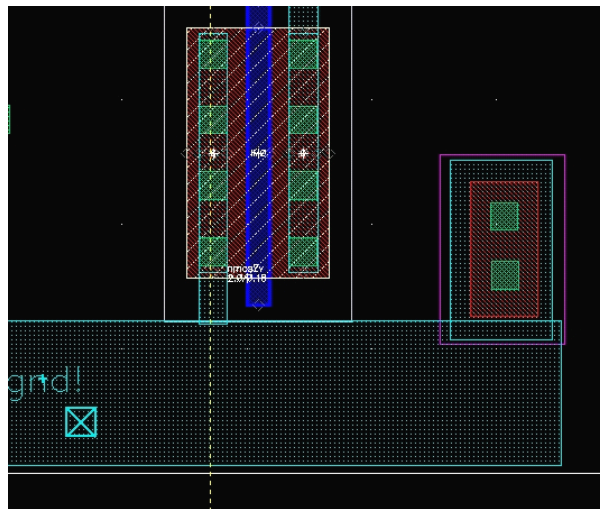
然后画 P 型扩散(为什么是 P 型扩散,不是 N? ),P 型扩散是 PIMP




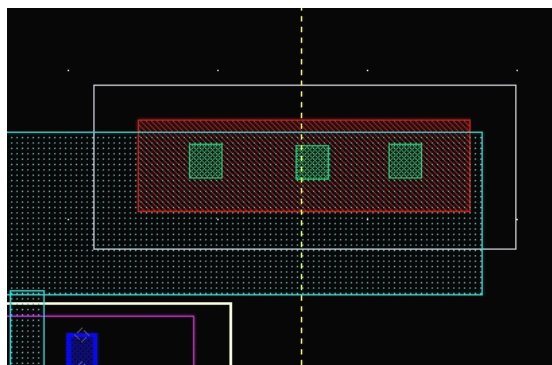
然后将过孔复制到上面，如下图：



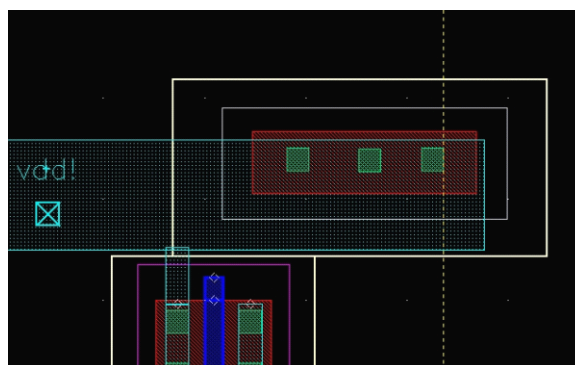
现在局部图如下：



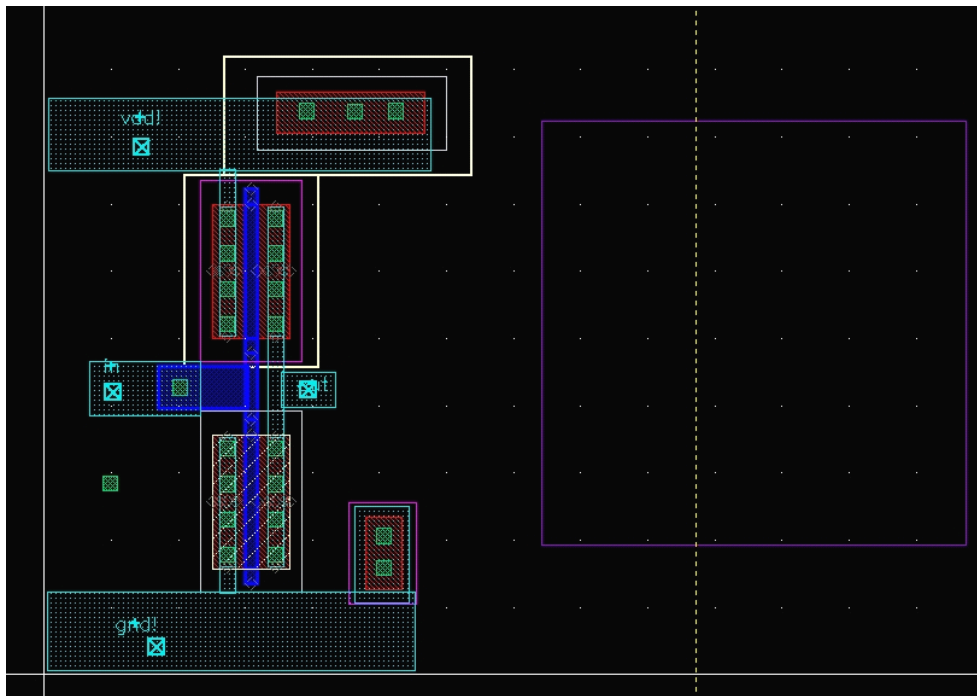
下面做 PMOS 管的衬底，其实衬底并不需要单独拉出来做，可以直接放置在电源线或地线上，PMOS 的衬底就直接放置在电源线上。同样，先放置扩散区 DIFF,再画 N 型扩散框（为什么这里是 N 型扩散？）表征 N 型扩散的是 NIMP ,并把过孔复制过来，如下图：



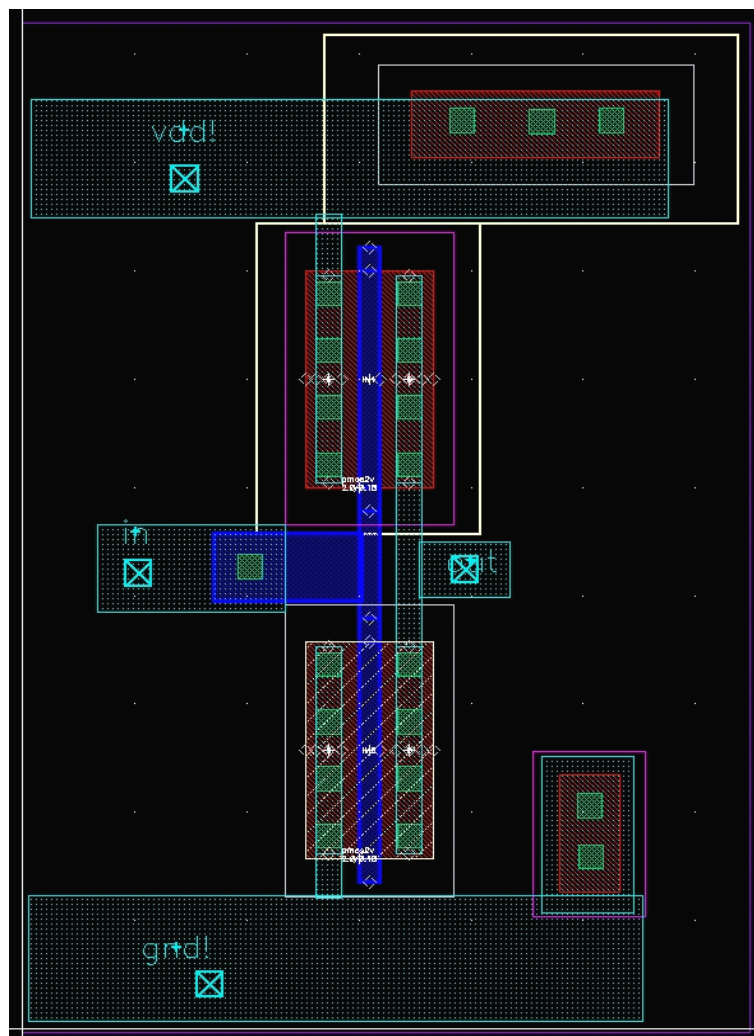
然后要画 N 阱,这个 N 阱必须要跟 PMOS 管的 N 阱连接着(为什么?)。N 阱是 NWELL,结果如下图：



这样，MOS 管的版图就画完了，如下：（那个空地方的绿色的过孔是没用的，那个只是为了画 0.22\*0.22 的过孔时用的，后面的过孔都是从它复制过来的，用完后删除掉就可以了）



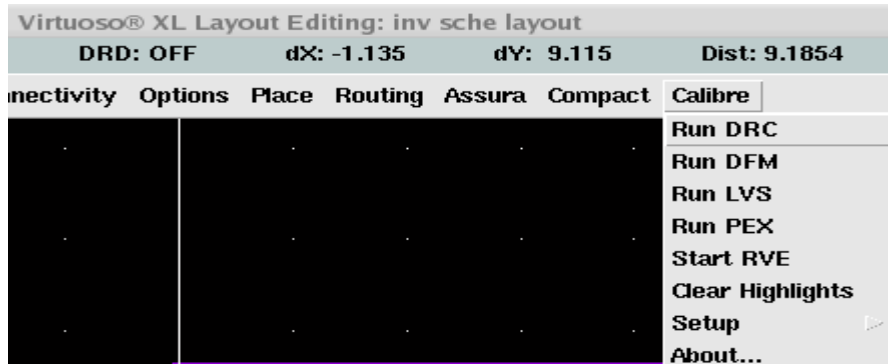
再将紫色的方框拉伸移位后得到下图：



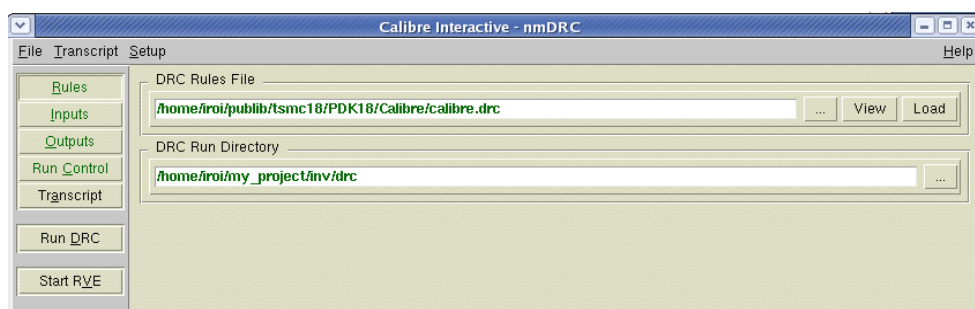
到这里，整个版图就画完了，再下来就是介绍 DRC 检查了。

## (八) DRC 检查

Calibre 软件的介绍我这里也只是简单介绍，详细的参见其他教程。  
然后进行 DRC 检查，点击 Calibre 的 Run DRC



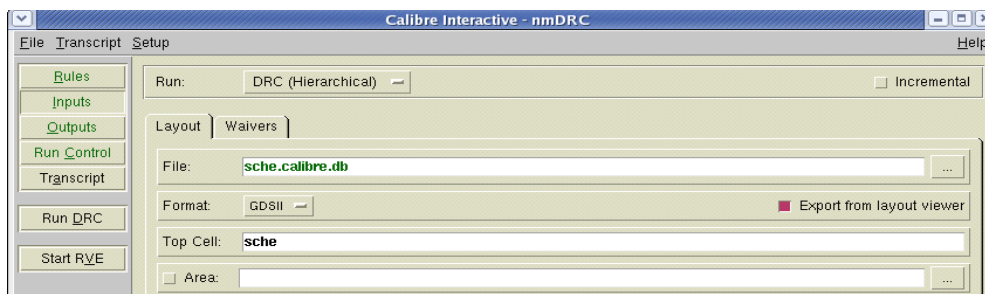
打开如下窗口：刚打开时要 load runset file，这里没有就点击 cancel。



在 Rules 标签中的 DRC Rules File 中加入 rules file，tsmc 公司提供的是 Calibre.drc。所在的目录就是上图中所示的目录。

然后在 DRC Run Directory 中选择输出文件存放目录。这里要注意，DRC 检查会产生很多文件，为了文件存放清晰，DRC 强烈建议在 inv 目录下新建个文件夹 drc，这样生成的所有文件都在 drc 文件夹下。

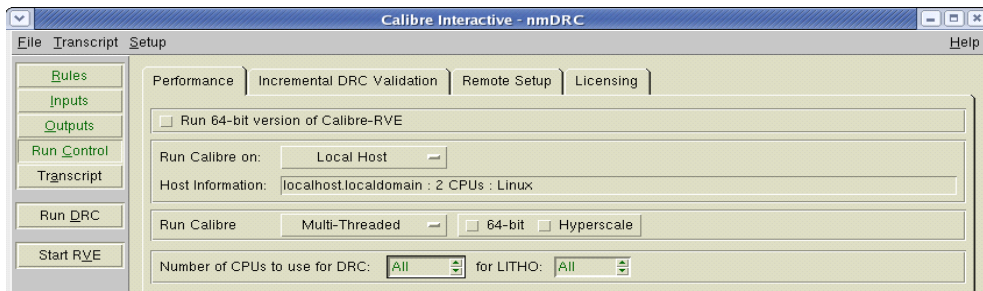
然后是 inputs 标签中，如果选中 Export from layout viewer，则软件自动生成网表，网表的名字就是 File 中的 sche.calibre.db，如果没选中，软件将调用 sche.calibre.db 文件。这里我们不改。如果颜色是红色，就说明设置有错误。



Outputs 标签中的我们也不改。

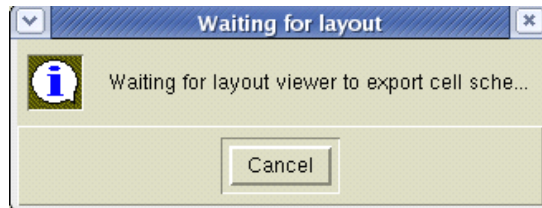
Run Control 标签中，如下：



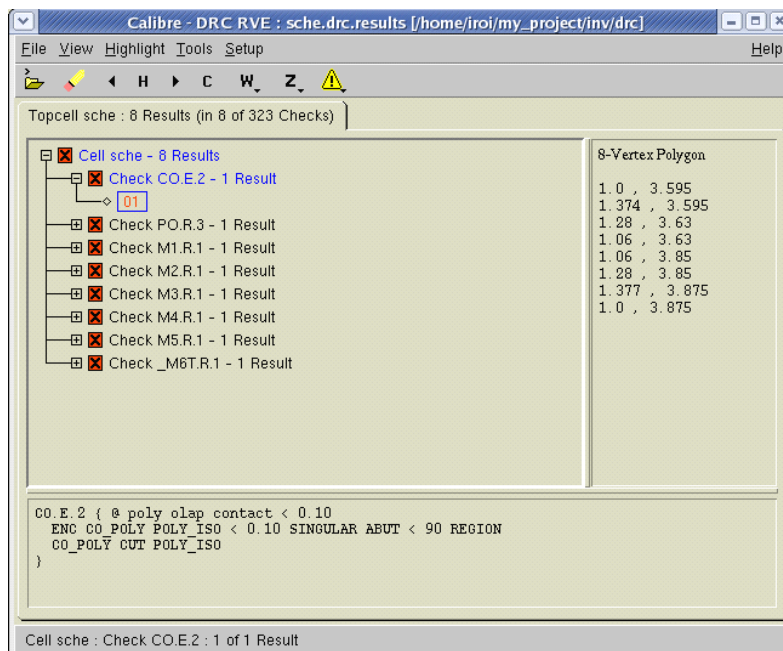


有的电脑是是多核 CPU，所以 Run Calibre 中选中 Multi-Threaded。

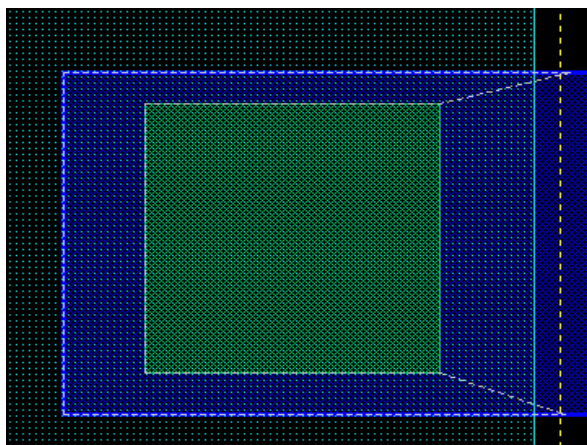
然后点击 Run DRC。它会出一个对话框：（等一下就是了，不要点击 Cancel）



完了后会出现错误信息窗口 RVE，如果没有出现，点击左边的 Start RVE。

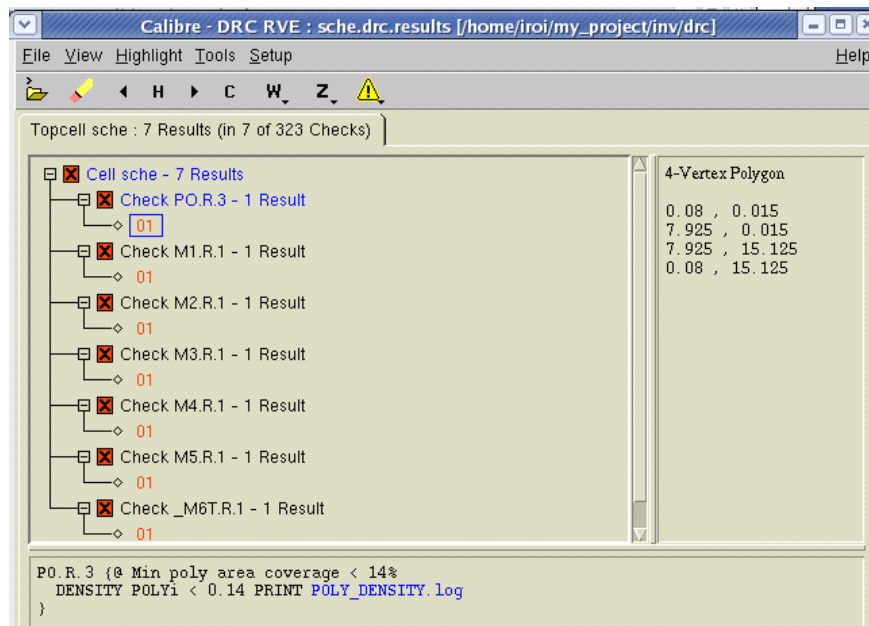


这里会显示错误信息，错误原因，双击某个错误，下面会出现错误原因，也会在版图中高亮显示：



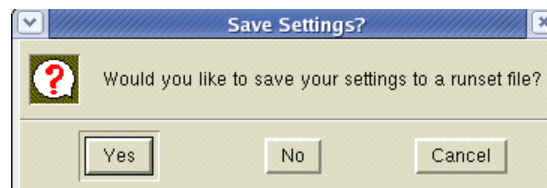


很明显，contact 和 poly1 的边距太小，改后即可。再 Run DRC 完了之后应该还有 7 个错误，如下：



分别是说，poly1 的面积小于 14%，M1 到 M6 的面积小于 30%，这个需要在空余的地方增大他们的面积。M2-M6 没有使用，我们这里就随便画了。其实这个可以到后面来处理，因为它不影响后面的 LVS 检查。我们也留在后面讲这个问题。

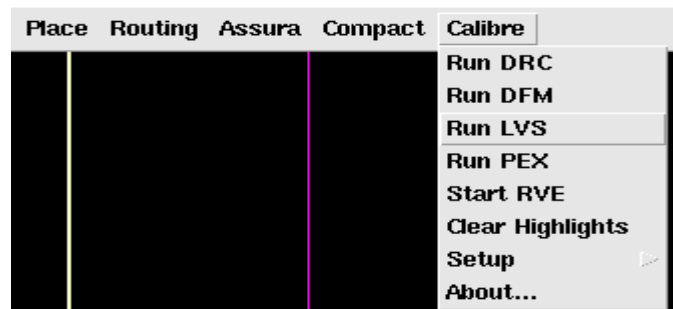
关闭 DRC 时，会提示保存 runset file



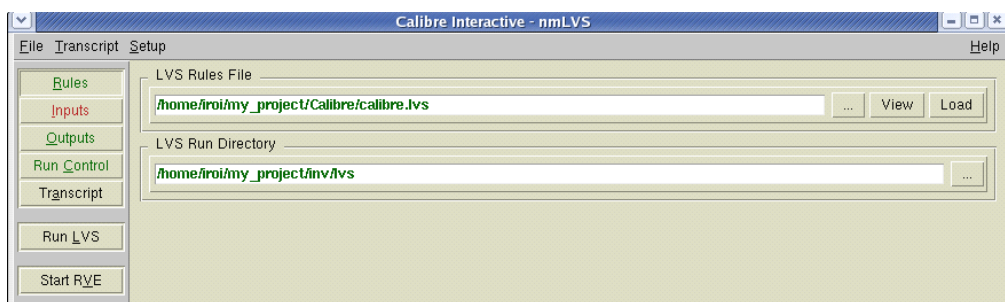
点击 yes，取个名字保存在 DRC 目录下就可以了。

### (九) LVS 检查

DRC 检查就是这样，现在进行 LVS 检查，LVS 是原理图和版图对应的检查，看两者是否一致。点击 Calibre 的 Run LVS。

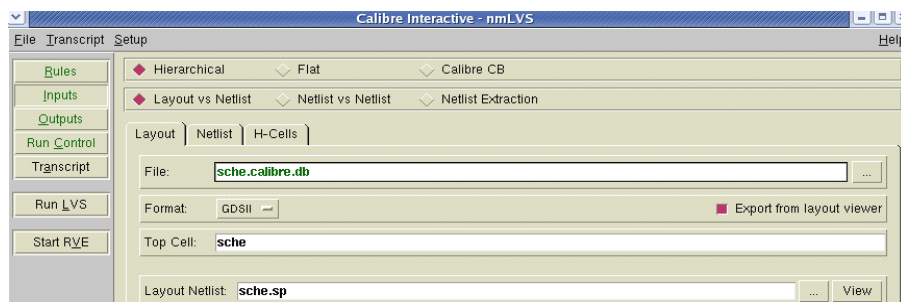


同样，启动没有 runset file，选择 cancel 打开如下：



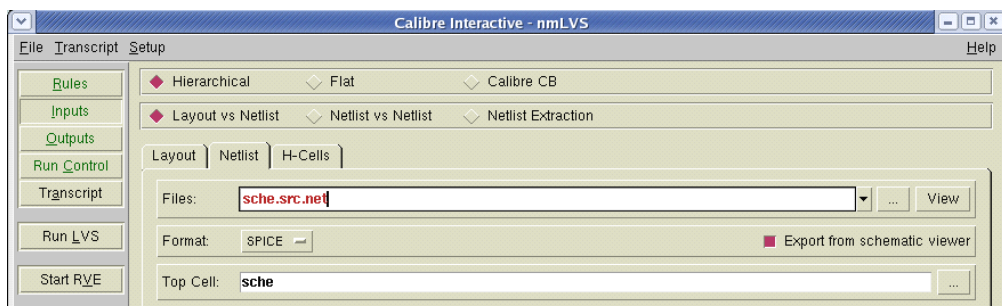
装载 rules file，选择 lvs 目录，还是强烈建议，在那 inv 目录下新建一个 lvs 目录，它将生成很多文件，也不要使用 drc 目录。

同样在 inputs 标签中：



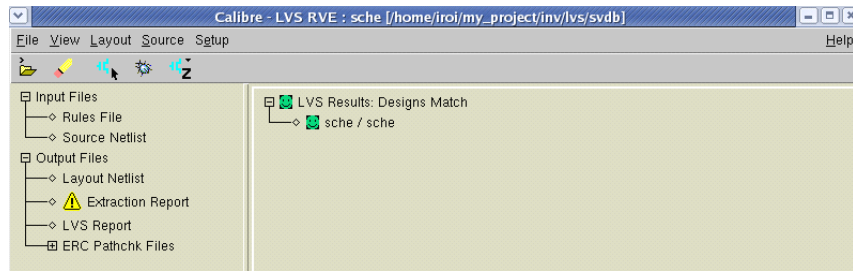
Layout 选项中 Export from Layout viewer 功能同上面相同。

Netlist 选项中第一次选中 Export from Layout viewer



Layout 对应的是版图的信息，Netlist 是原理图的信息。

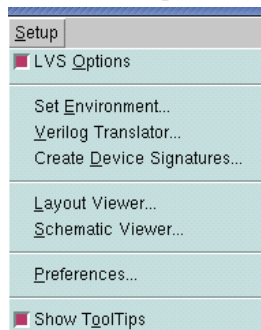
Outputs 这里不改，Run Control 像上面一样改成多核的。然后点击 Run LVS。



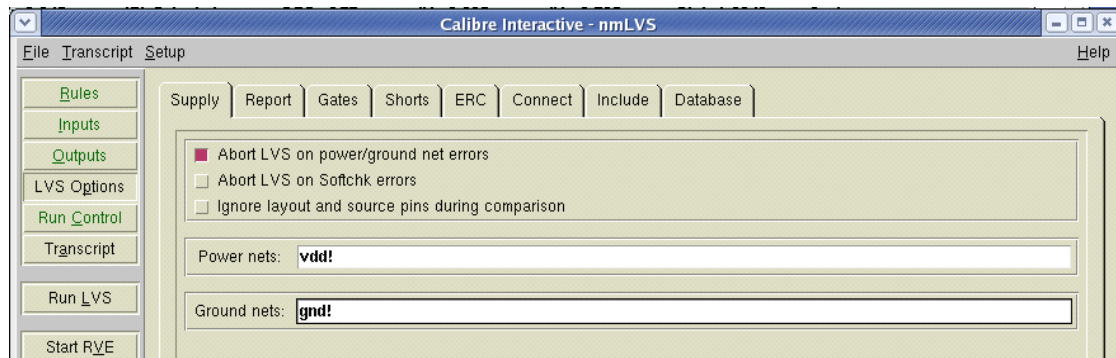
发现有警告，双击警告信息，得到：

```
WARNING: Invalid PATHCHK request "GROUND && ! POWER": no POWER nets present, operation aborted.
WARNING: Invalid PATHCHK request "POWER && ! GROUND": no POWER nets present, operation aborted.
WARNING: Invalid PATHCHK request "! POWER && ! GROUND": no POWER nets present, operation aborted.
```

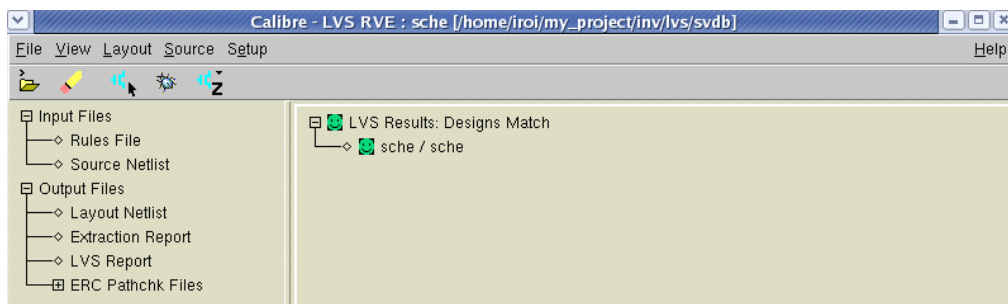
电压和地有问题。在 setup 里面选中 LVS Option



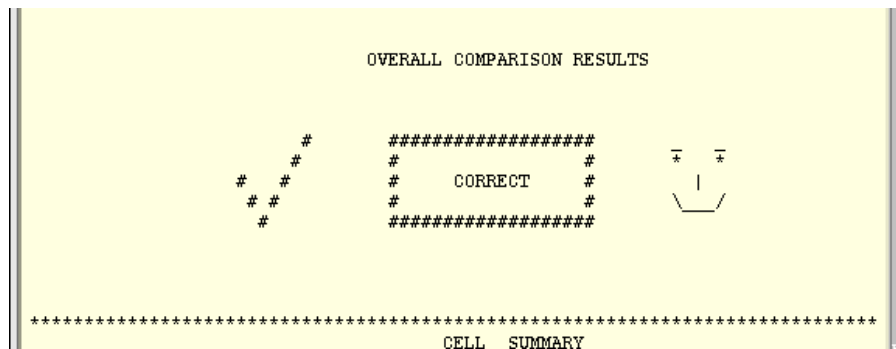
填写如下，其他的不改。（记得有！）



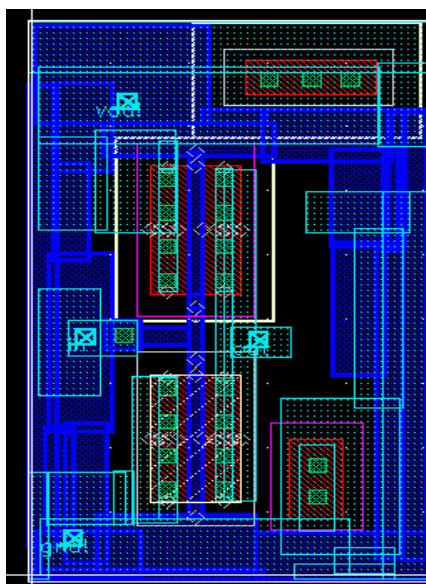
再次运行 Run LVS，得到：



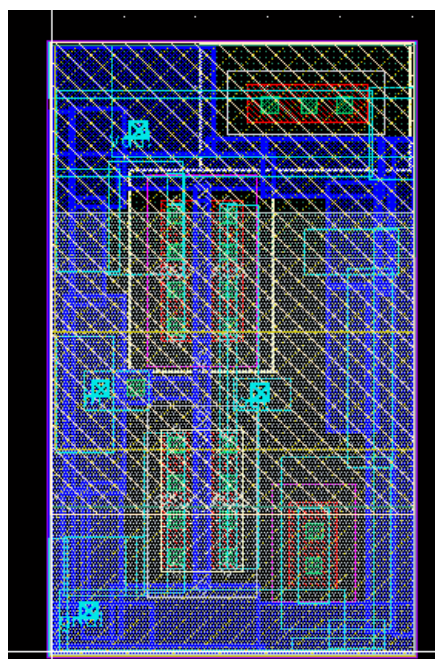
警告没有了，而且会出现对勾和笑脸，如下：



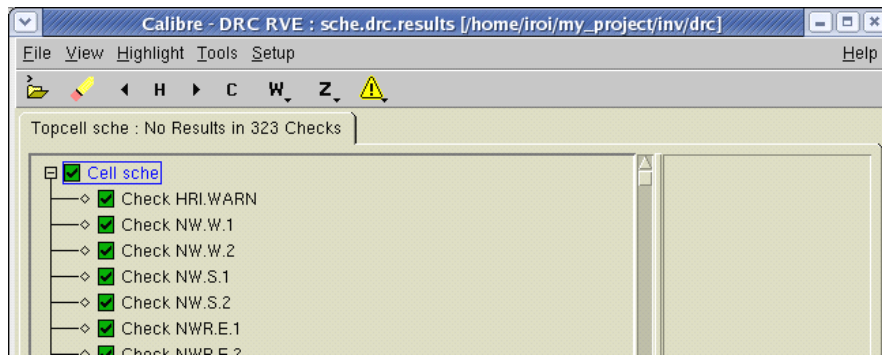
LVS 就结束了，退出时保存 runset file。  
然后就是要把前面几个的 DRC 错误改过来。



其实就是在空白的地方加 POLY1 和 METAL1，记得在加完这两个后要进行 DRC 检查，因为可能加的时候违反了设计规则。然后加 METAL2 到 METAL6 就不要紧了。

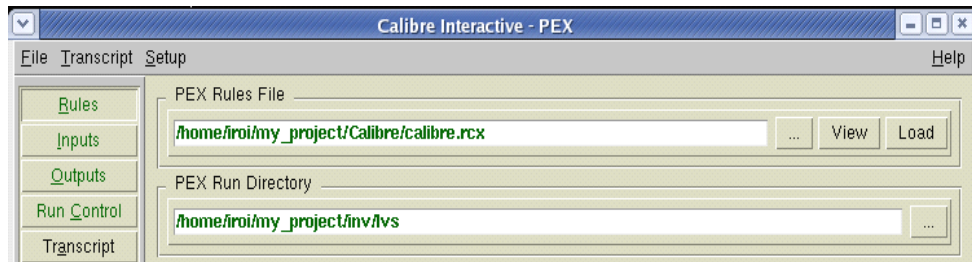


这是加了金属层后的最后的版图，再进行 DRC 检查，结果就没有错误了，如下：

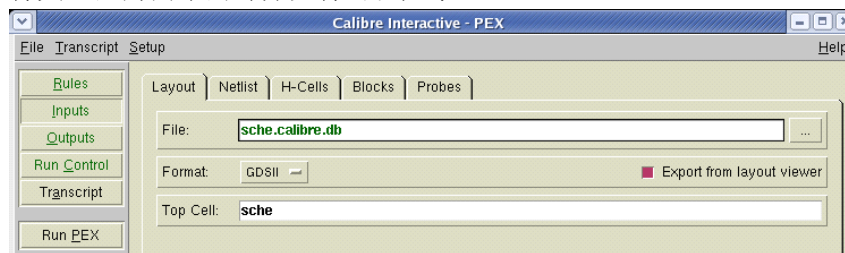


## (十) PEX 参数提取

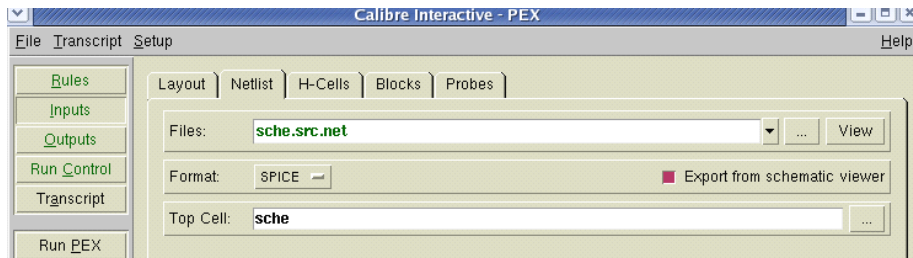
接下来进行 PEX 参数提取。点击 Calibre 的 Run PEX。



加入 PEX 的 rules file，tsmc 公司提供的是 calibre.rcx。这次的目录跟 LVS 目录相同，都是 lvs 文件夹，因为两个的部分文件可以共享。

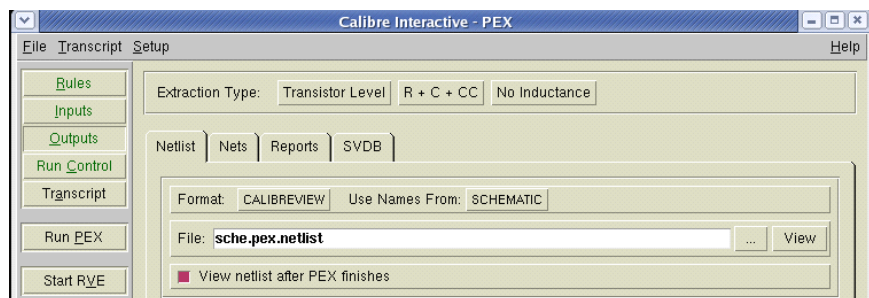


上面是 Layout 的。下面是 Netlist 的。



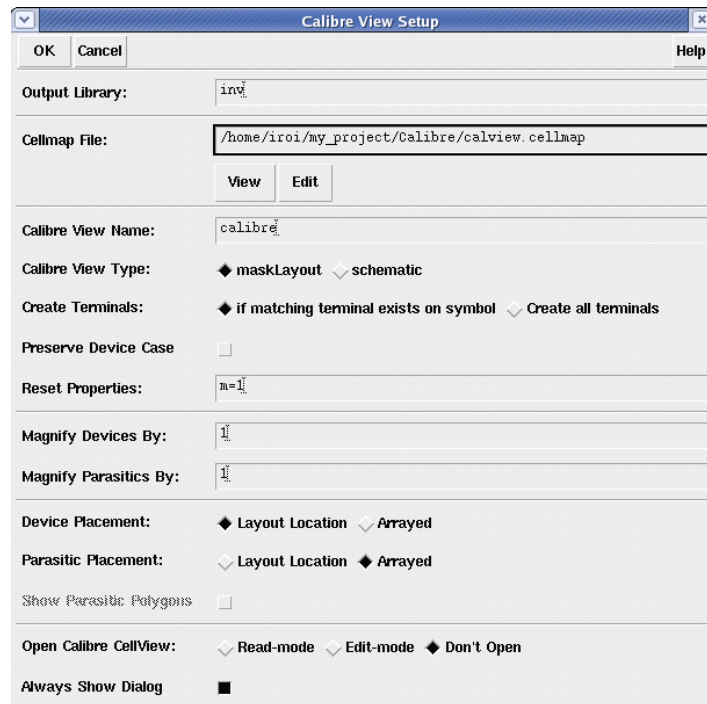
如果是跟 lvs 同一个目录，则上面两个文件都已经生成了，Export from layout viewer 和 Export from schematic viewer 就可以不选中了，这样节省了时间。

Output 的设置如下：Format 设置为 CALIBREVIEW，其他的设置为与图中相同即可。

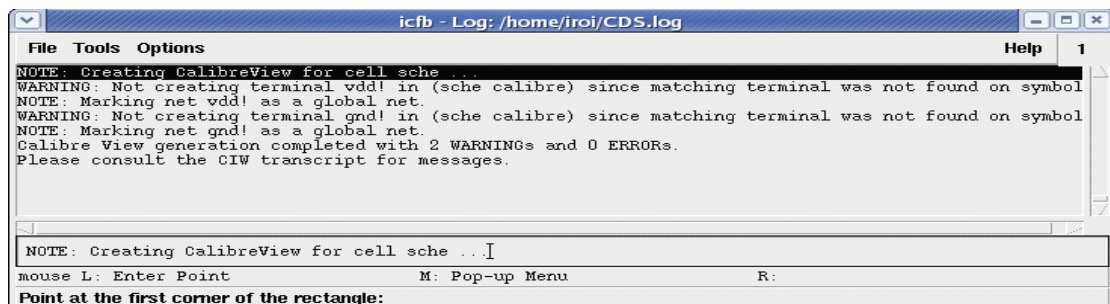
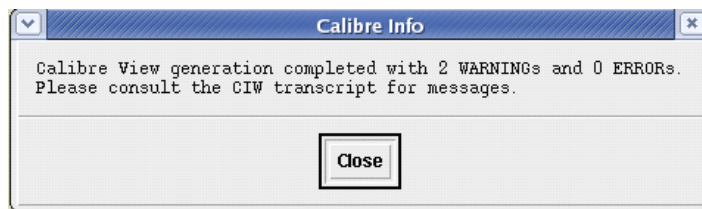


Run Control 设置为多核。点击 Run PEX。

过一会儿会出现下面的窗口：

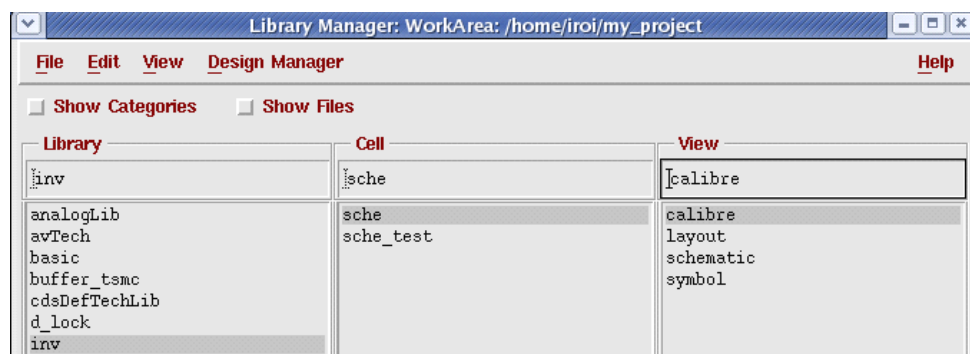


这里关键是要输入 Cellmap File 的路径，然后在 Calibre View Name 里面输入要生成的名字。点击 ok。会出现如下警告：



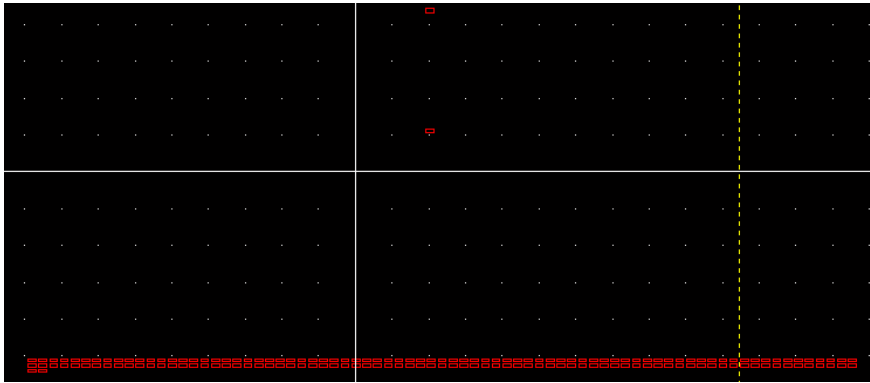
可以看到警告内容，我也不知道是否有错。

然后就会在 Library Manager 里面看到多了个 View: calibre

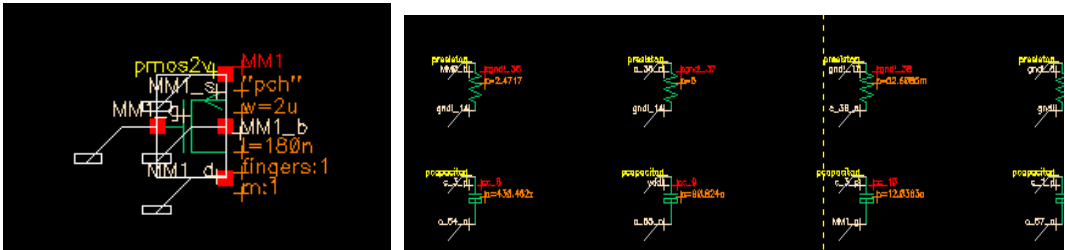




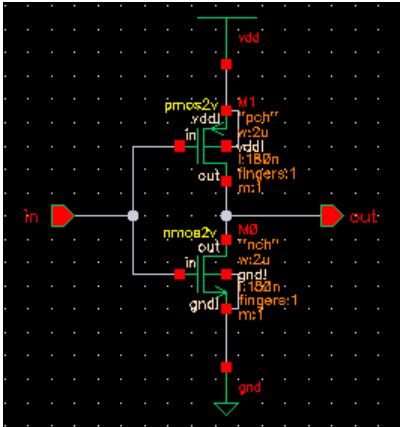
这样，版图提取就完成了，版图提取后的文件就是 calibre。  
双击 View 中的 calibre 就可以看到：



上面两个红色的方框就是两个 mos 管，下面那么多都是提取出来的寄生参数。放大看如下：



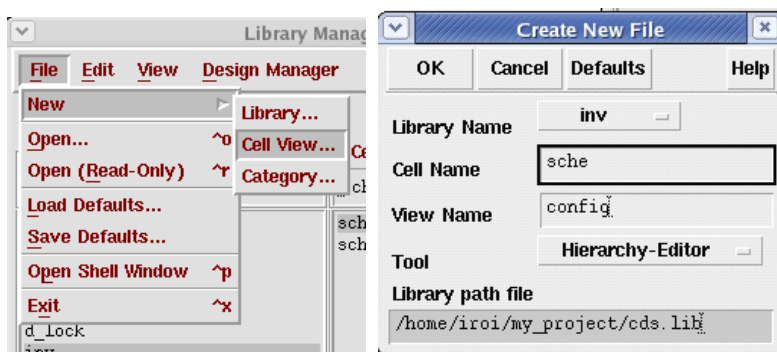
注意，这个是非门的版图，而测试电路不在里面：



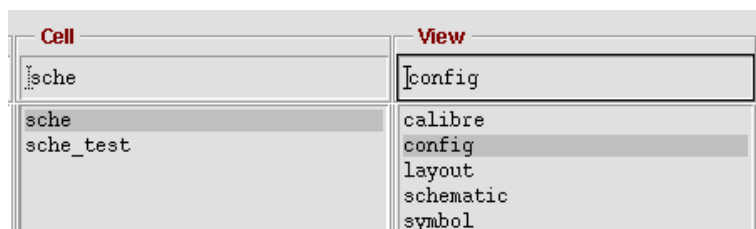
### (十一) 后仿真

接下来就是后仿，要生成用于后仿的文件。

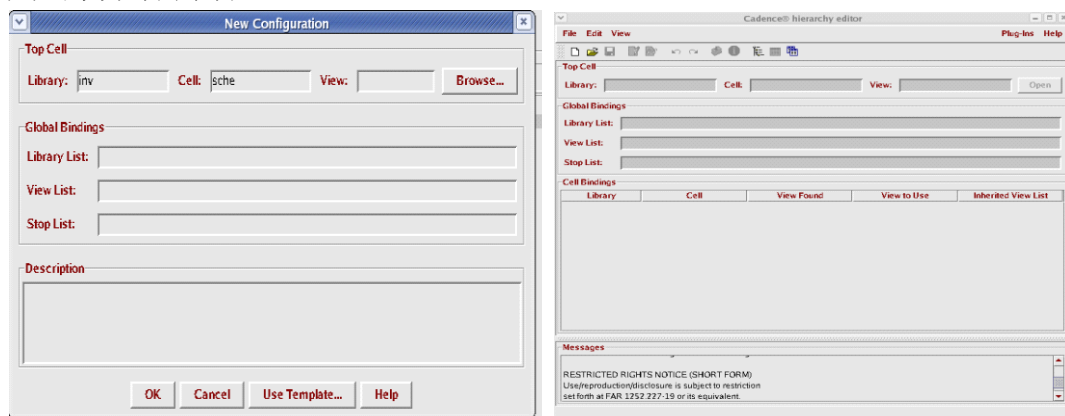
新建 Cell:



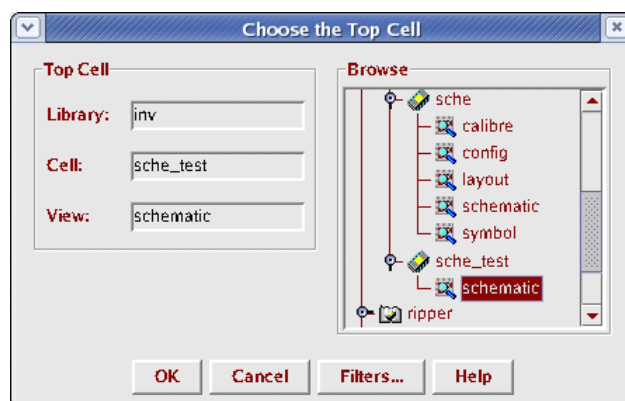
注意上右图中，Cell name 还是要在 sche 中，Tool 改为 Hierarchy-Editor，View Name 就会自动变为 config。点击 Ok。可以看到多了个文件：



同时出来另外两个窗口：



点击左边的窗口的 Browse...将出现下面窗口：

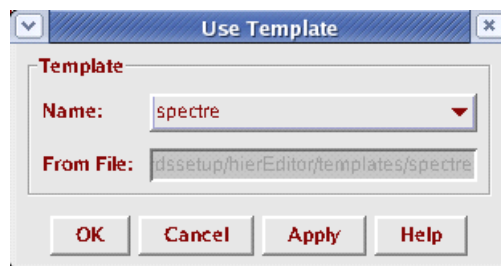


选择 sche\_test 中的 schematic，点击 ok。不要选 sche 中的 schematic，因为顶层文件是 sche\_test。

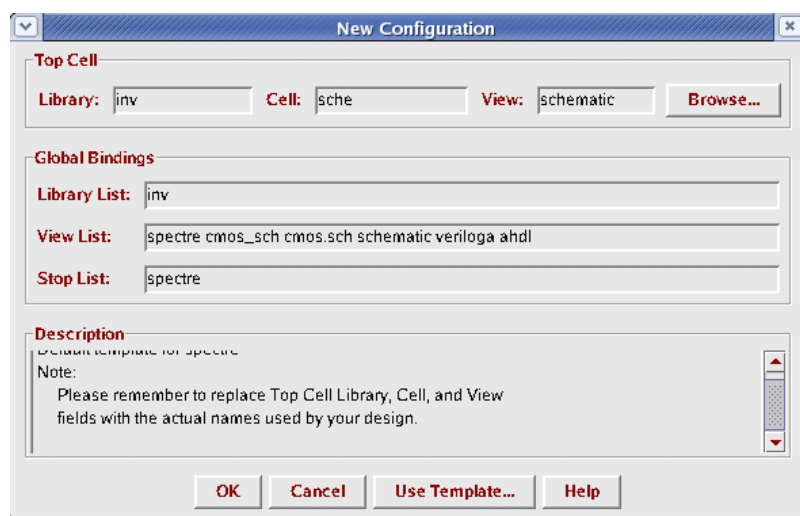
然后点击



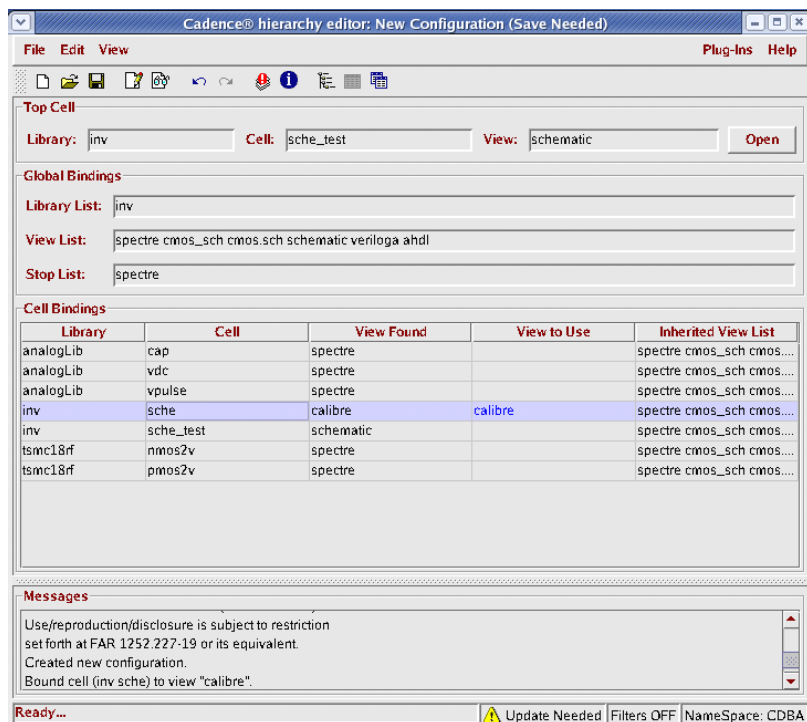
中的 Use Template...得到:



在 Name 中选择 spectre，点 OK，得到:

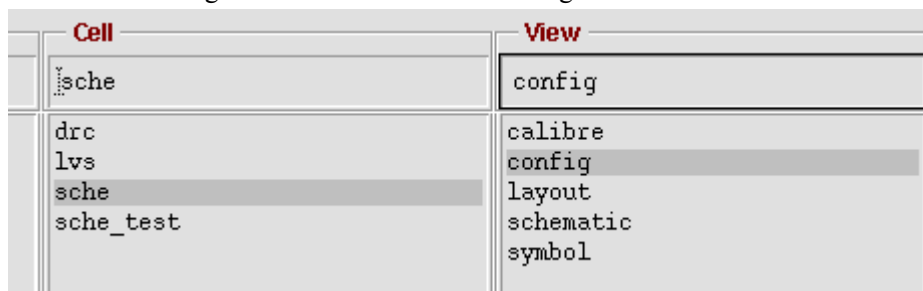


然后在上图中将 Library List 改为 library 的名字 inv。点击 OK 得到:

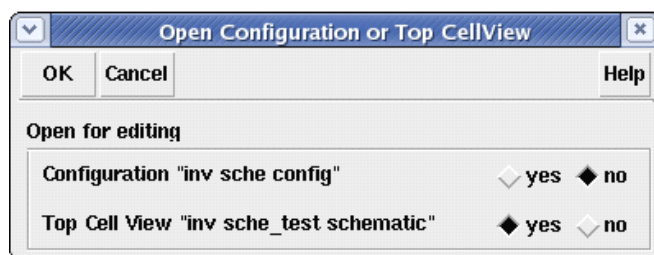


在所示的那行的 View to Use 列输入 PEX 生成的文件名，这里是 calibre。保存之后可以关掉。

然后可以打开 config 看看，双击 View 中的 config:

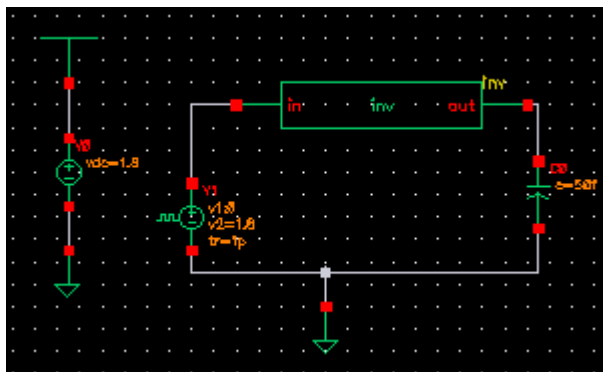


出现如下:

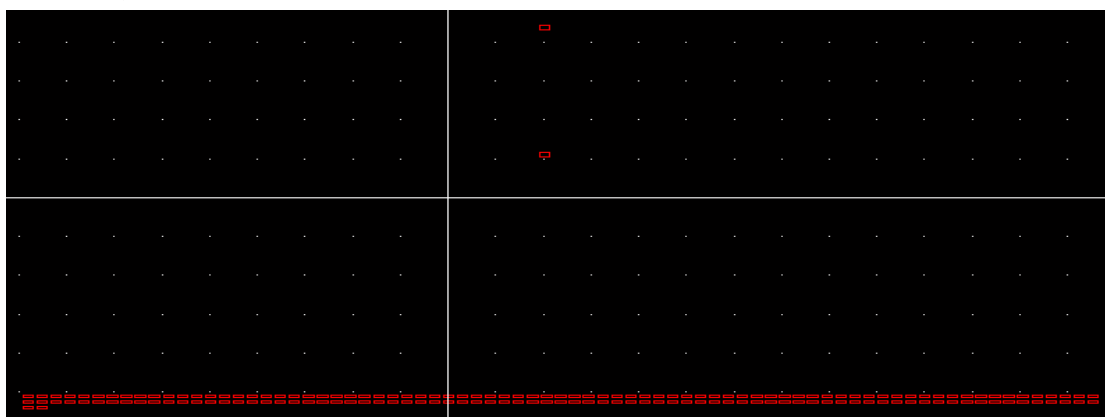


可以两个都选 yes，看看打开的是什么，我这里就只打开下面的，上面的你打开了就熟悉。

打开后就是下面这个图:



看起来好像没什么哈，但是选中 inv，按键 e 进入其内部，得到:



应该眼熟吧，用的是提取版图后的图做的仿真，这样就把各种参数的影响全部考虑进去了。然后接下来的仿真就和原理图的仿真是一样的了，整个过程就结束了。

所有过程结束了，建议多走两遍，熟悉下流程，学习下软件，里面的每一步都是有原因的，想想作用在哪里。