



集成电路设计实习 VLSI Design Labs

# 单元实验一（第一次课）

基本门电路设计——反相器

## 实验目的及时间安排

---

- 掌握基本门电路的设计方法
- 熟悉Cadence的设计数据管理结构，以及定制设计的原理图输入、电路仿真、版图设计、版图验证工具的使用
- 电路结构和优化设计可以参考甘老师《集成电路原理与设计》
- 完成反相器的设计
- 设计时间：一次课完成反相器的设计

# 反相器的设计

---

- 设计目的：基于Chart 0.35um工艺完成一个具有逻辑反相功能的电路
- 设计要求：
  - 反相器的逻辑阈值在 $V_{dd}/2$ 附近，即噪声容限最大
  - 反相器的版图高度限制为13微米，电源和地线宽度各为2微米
  - 反相器宽度限制为mos器件不折栅
  - 为了给顶层设计留出更多的布线资源，版图中只能使用金属1和多晶硅作为互连线，输入，输出和电源、地线等pin脚必须使用金属1
  - 版图满足设计规则要求，并通过LVS检查
  - 为了满足以后复杂门电路设计的需要，要求反相器版图满足上、下、左、右并置排列的时候不违反设计规则

# 如何开始设计

---

- 作为我们的第一个设计，请按照后面的具体指导进行
- 设计过程：
  - 设置Cadence环境
  - 启动icfb
  - 建立自己的设计库
  - 用Virtuoso Schematic Composer 画电路图
  - 在Analog Design Environment中进行电路仿真
  - 用Virtuoso (XL) Layout Editer画版图
  - 利用diva工具进行版图验证和提取

# 1 熟悉设计环境

---

- 在本地目录中建立自己的工作目录，名称为姓名缩写加上学号后四位，例如js0459
- 将设计数据包dsnLab.tar.gz文件拷贝到自己的工作目录，可以用命令行cp命令，也可以用鼠标拖动+control键（注意一定在移动过程中按住control键）
- 利用gzip -d命令将设计数据解压，利用tar -xvf命令将数据解包，输入文件名的过程中按Esc键可以帮助自动完成相关输入
- 解压完成后可以看到dsnLab目录，为我们的设计目录，以后所有的设计均放在该目录中，利用rm命令可以删除tar文件
- 进入dsnLab目录中，lab1,lab2...等目录为单元实验目录，而logicLab和analogLab分别为数字和模拟综合实验目录，chrt35dg\_COREcell目录为数字半定制设计中使用的Chart 0.35um工艺标准单元库，chrt35dg\_SiGe为定制设计中使用的Chart 0.35um工艺的PDK包，即模拟库

## 2. 设计环境lab1

---

- 进入lab1目录，lab1为定制设计实验，该目录已经配置了定制设计环境
- 在file manager界面的菜单选择view—show hidden objects选项，可以看到包括隐藏文件（以.开头的文件）在内的所有目录和文件
- 其中的.cdsinit文件为cadence定制工具dfll的初始化文件，看看你的本地目录student下是否有该文件，如果没有可以将其拷贝或者链接到本地目录下，这样其他lab中的定制设计可以使用该文件
- cds.lib文件为dfll的库定义文件，打开该文件可以看到用include命令将chrt35dg\_SiGe目录中的一个库定义文件包括进来，这样可以利用该库文件定义的库进行扩展
- 在unix中（./）表示当前目录，（~/）表示本地目录

### 3. 启动Cadence (icfb &)

- 从你的工作目录启动Cadence定制工具dfll

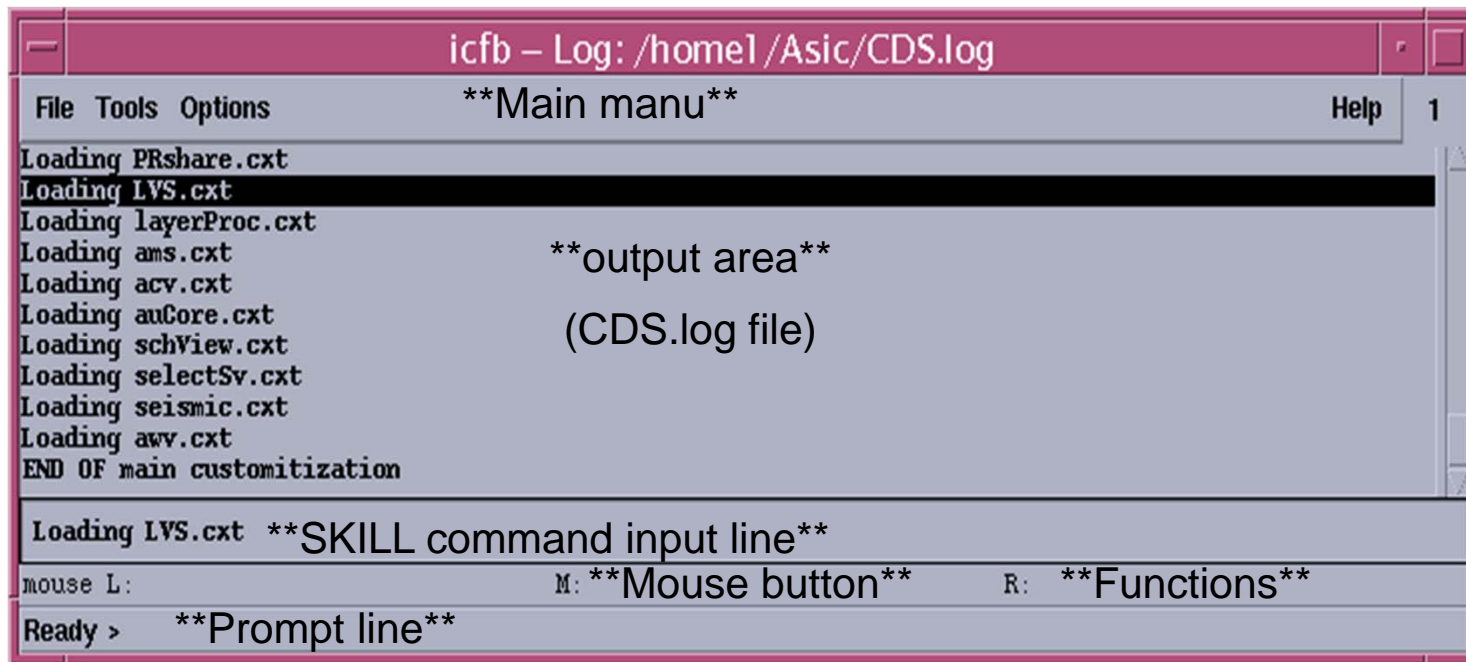
<prompt> icfb &

- icfb为可执行文件的名称，（&）符号表示让该软件在terminal的后台执行，即你还可以在该terminal中运行其他命令
- 工具启动后出现CIW: Command Interpreter Window

```
icfb - Log: /home1/Asic/CDS.log
File Tools Options      **Main manu**      Help 1
Loading PRshare.cxt
Loading LVS.cxt
Loading layerProc.cxt
Loading ams.cxt          **output area**
Loading acv.cxt          (CDS.log file)
Loading auCore.cxt
Loading schView.cxt
Loading selectSv.cxt
Loading seismic.cxt
Loading awv.cxt
END OF main customitization
Loading LVS.cxt **SKILL command input line**
mouse L:                M: **Mouse button**    R: **Functions**
Ready > **Prompt line**
```

# CIW

- 在CIW窗口中，你可以查看不同工具的运行状况，包括各种错误提示（如果设计中有错误，首先需要查看错误信息，这对我们确定问题很有帮助）
- CIW窗口为图形界面的dfII工具集成了cadence的定制设计工具，我们利用该平台开展定制设计



```
icfb - Log: /home1/Asic/CDS.log
File Tools Options      **Main manu**      Help  1
Loading PRshare.cxt
Loading LVS.cxt
Loading layerProc.cxt
Loading ams.cxt          **output area**
Loading acv.cxt          (CDS.log file)
Loading auCore.cxt
Loading schView.cxt
Loading selectSv.cxt
Loading seismic.cxt
Loading awv.cxt
END OF main customitization

Loading LVS.cxt **SKILL command input line**
mouse L:              M: **Mouse button**      R: **Functions**
Ready >      **Prompt line**
```



## 4. 建立自己的设计库

---

- 在dfII的tools菜单下启动library manager工具，可以帮助我们管理库中的设计数据
- Cadence启动后会自动加载cde.lib文件定义的库
- 在library manager中可以看到已经有了很多库，其中的chrt35dg\_IOcell和chrt35dg\_COREcell为cde.lib中定义的本目录下的库
- chrt35dg\_SiGe为模拟单元和工艺库，其中包括了定制设计所需的器件，还包括工艺信息
- analogLib库为dfII提供的器件库，而basic为dfII提供的基本符号库，我们主要用analogLib中的工艺无关器件和chrt35dg\_SiGe中的工艺相关器件构建电路
- cdsDefTechLib为dfII提供的缺省工艺库，我们不用

## 建立自己的Design Lib

- 第一步:

CIW->File->New->Library

- 第二步:

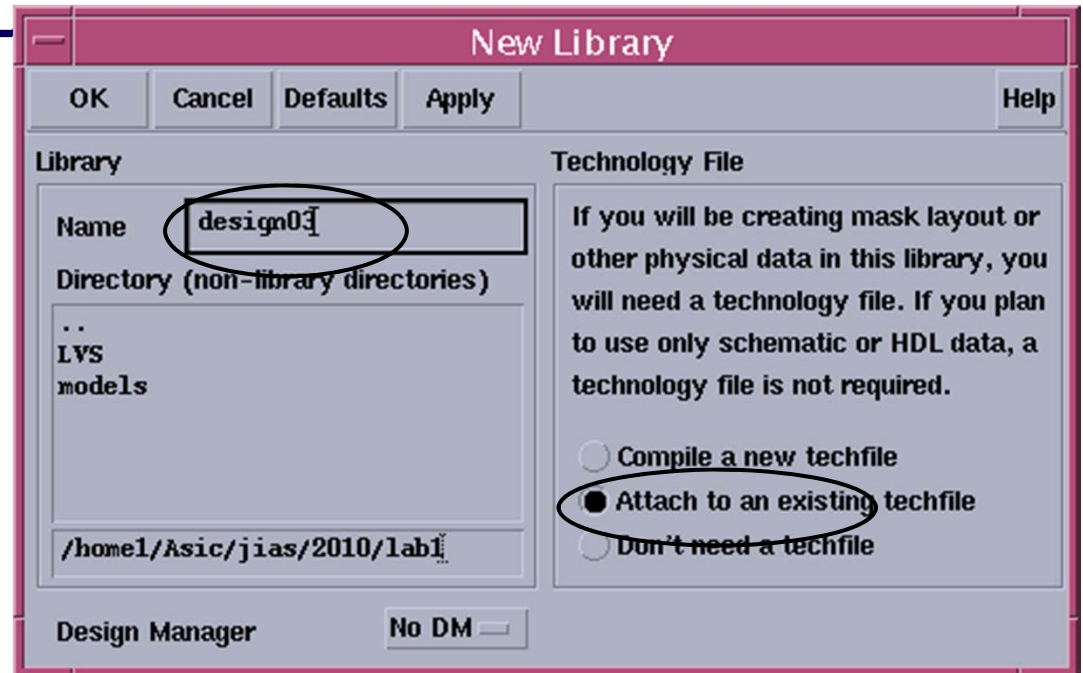
弹出“New Library”对话框

在“Name”项填写要建的design lib的名字，这里是“design03”，选择“Attach to an existing techfile”

- 第三步:

弹出” Attach Design Library to Technology File”对话框

在“Technology Library”中选择“chrt35dg\_SiGe”



## 5. 原理图输入，创建新的Schematic

- 建立了设计库，并把设计库附加到工艺库上，则可以在设计库中开始设计

- 第一步：

CIW->File->New->Cellview

- 第二步：

弹出“Create New File”对话框

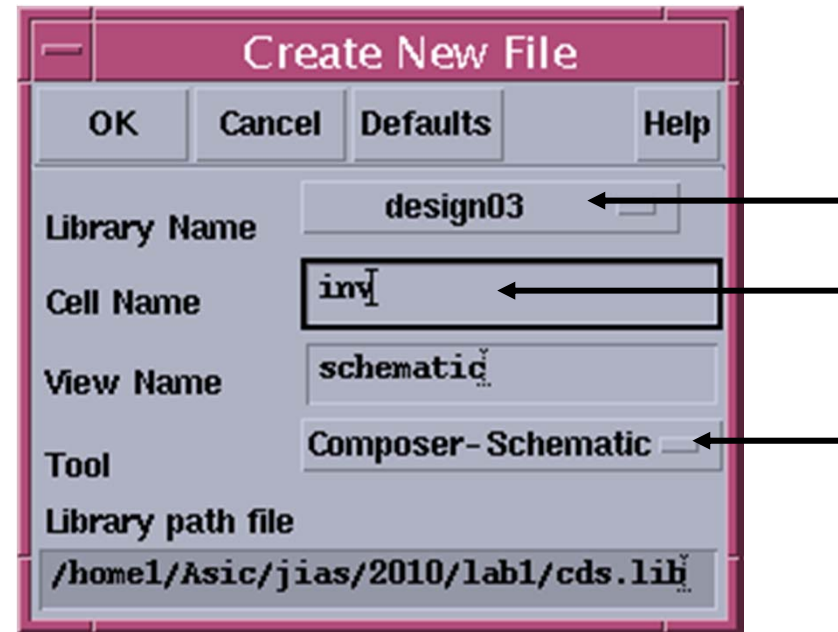
“Library Name”项选择“design03”

“Cell Name”项填入” inv”

“Tool”项选择” Composer-Schematic”

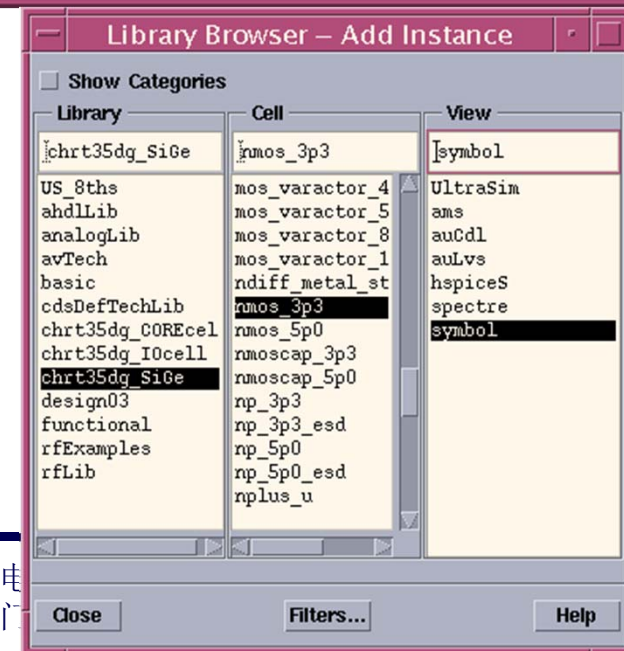
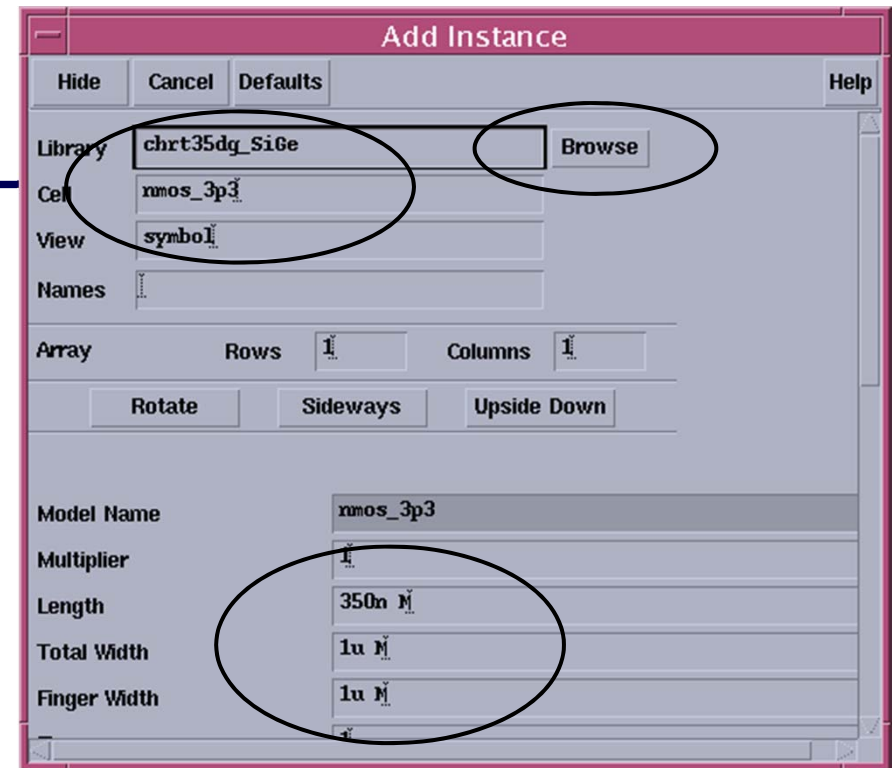
- “Tool”项确定后，相应的“View Name”项会出现内容,因而无需输入” Schematic”

- 点击“OK”后就进入Virtuso Schematic Composer



## 添加元件并编辑属性

- Schematic窗口: Add->Instance
- 在“Add Instance”中, 点击Browse
- 在“Library Browser”中(如下页所示), 选择需要添加的Instance:
  - Library: chrt35dg\_SiGe
  - Cell: nmos\_3p3
  - Cellview: symbol
- 编辑元件的属性
  - 修改MOSFET的Length, Width



## analogLib中的电源与全局节点

---

- 独立源：
  - 如vdc vpulse vpwl
- 受控源：
  - Vccs
- 全局节点
  - Vdd vcc gnd vdda vddd vss vssa vssd等
- 电路图中的电源和全局节点选用analogLib中的元件，其他元件选自chrt35dg\_SiGe

## 添加Pin



- 反相器中包括一个pmos和一个nmos器件，他们分别连接到电源/地和输入，输出信号之间
- 完成器件和电源的添加以后，需要定义输入和输出信号，在DFII中输入和输出信号用pin的方式定义，严格上说，电源和地也是pin，是输入输出属性的pin
  - Schematic窗口:Add->Pin
  - 指定” Pin names”
  - 选择 “Direction”

# 完成反相器的电路图

➤完成连线后，就得到了反相器的电路原理图

Pin, input型

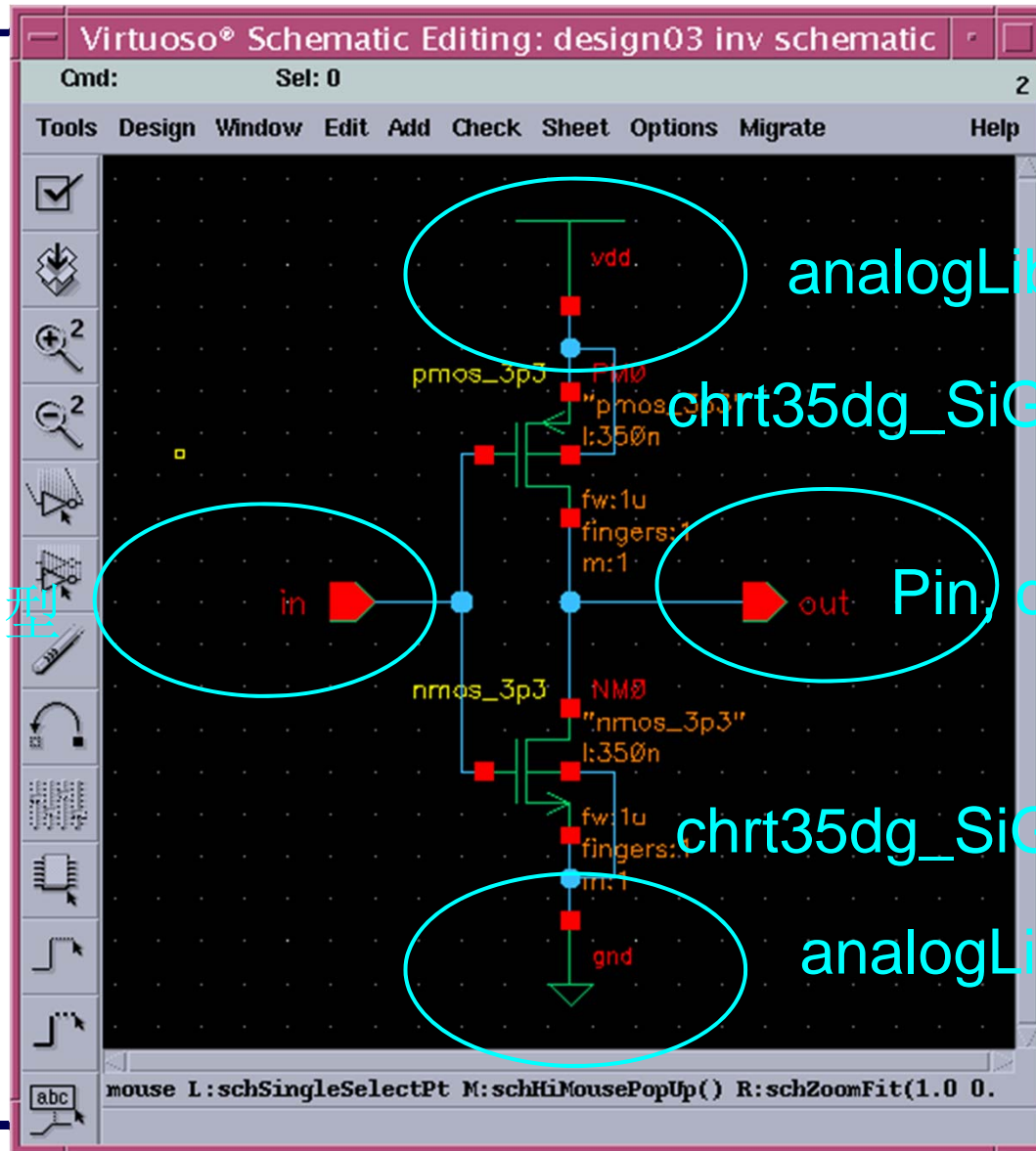
analogLib vdd

chrt35dg\_SiGe pmos\_3p3

Pin, output型

chrt35dg\_SiGenmos\_3p3

analogLib gnd



# Virtuso Schematic Composer常用操作

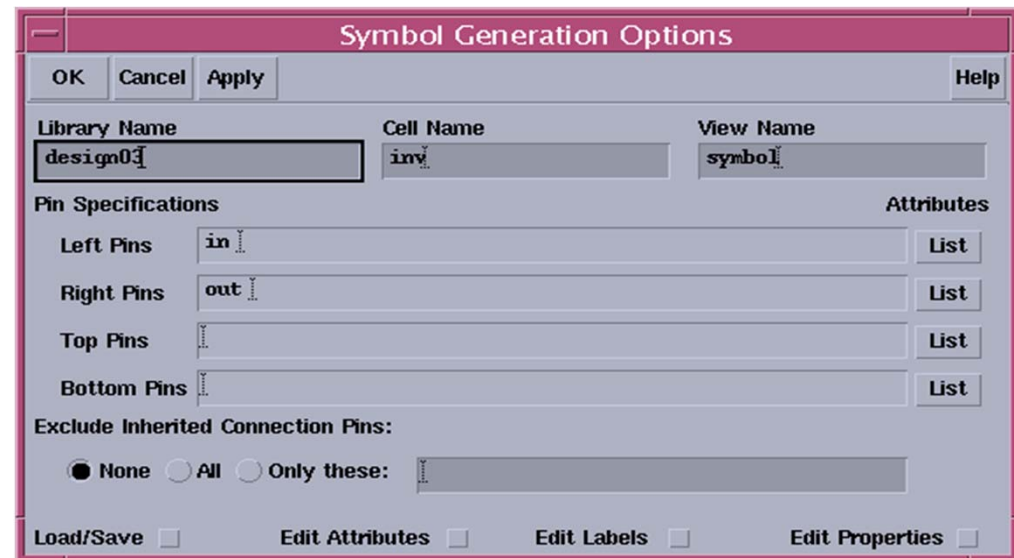
---

- Instance（添加元件，快捷键i）
- Pin（添加pin，快捷键p）
- Wire（连线，快捷键w）
- Check and Save（检查连接关系并保存，快捷键X）
- Copy（c）， Move（m）
- Zoom in by 2（]）， Zoom out by 2（[）， Fit（f）



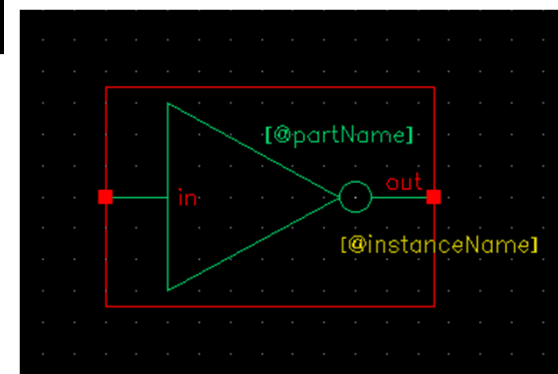
## 6. 创建反相器的符号图Symbol

- 为了使得设计好的反相器可以被更高层的设计例化为子电路，我们可以创建反相器的符号视图（symbol view）
- Schematic窗口中：Design-> Create cellview->From Cellview
  - 弹出“Cellview From Cellview”对话框，一般不改变默认值
- Symbol Generation Options窗口中
  - 默认Left Pins是“in”，Right Pins是“out”，即in端口在左，out端口在右，如果有必要，可以改变它们的位置



# Symbol的各部分内容

- Pin: 版图中的红点
- Pin name: 图中的in、out
- 如果需要修改Symbol的Pin, 必须同时修改Pin & Pin name
- 通过Symbol Editing窗口中的Add->Shape->Line(Circle,Arc...) 为了美观可以修改symbol形状, 我们暂时不修改



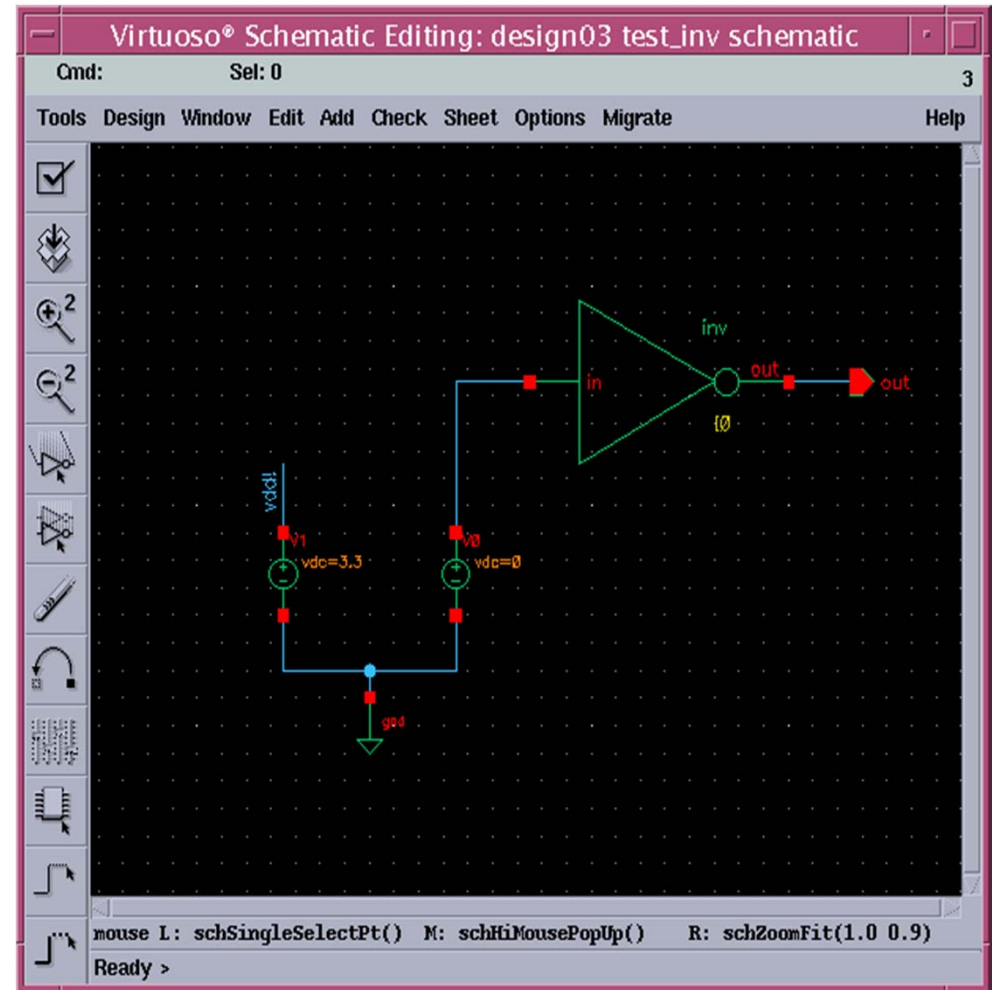
## 7. Inv的仿真

---

- 自此，我们已经完成了反相器的电路原理图设计，并生成了该电路的symbol，在library manager中可以看到你的设计库中的该单元有schematic和symbol两个view，view是同一个设计的不同形式
- 下面我们利用电路仿真来对电路的功能和性能进行验证和优化，定制设计中使用spice仿真器进行电路仿真，即输入电路网表和器件模型，spice进行计算，输出计算结果，显示为波形，设计者可以根据波形判断电路的特性
- dfll中的集成仿真环境ADE（analog design environment）可以调用spice仿真器进行电路仿真以及其他分析
- 我们首先构建被仿真电路inv的仿真电路，然后利用ADE进行仿真
- 对于反相器我们可以利用spice直流扫描功能得到其传输特性曲线，观察其噪声容限等直流特性；然后进行瞬态仿真观察其速度和功耗等时序特性（即甘老师书上第四章内容）

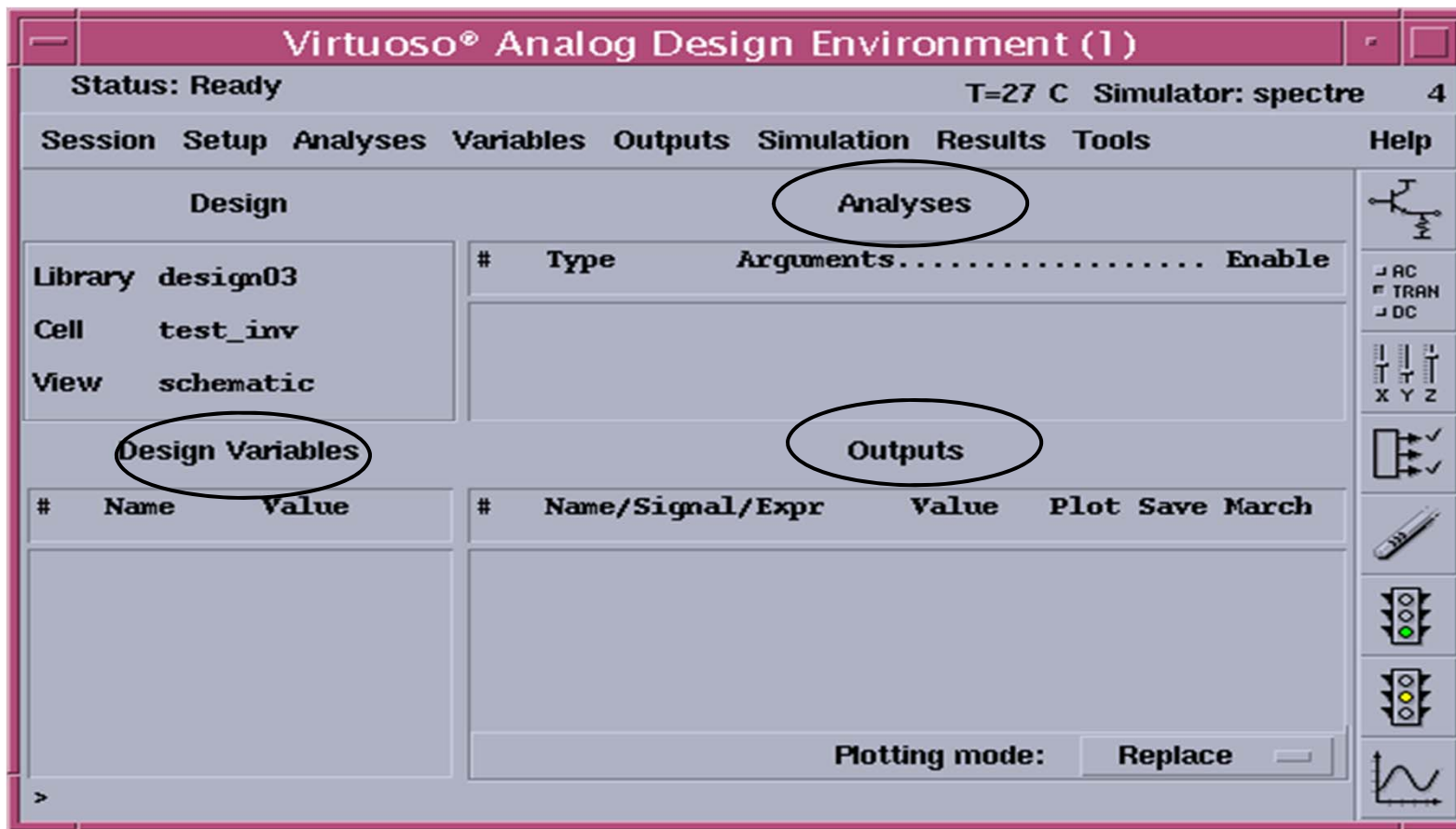
# Inv的仿真

- 在设计库中新建一个schematic, test\_inv, 作为仿真的电路, 在其中例化inv, 并增加激励源, 我们要通过spice仿真观察inv对激励信号的输出响应是否满足设计要求
- 左边的直流源V1给vdd供电, 线上加上label“Vdd!”表示这根线和全局变量Vdd相连
  - 其中的vdc=3.3 v可以通过选中该元件后按“q”键编辑属性得到
- 右边的直流源V0加到inv输入端, 用于直流扫描



# Inv的仿真

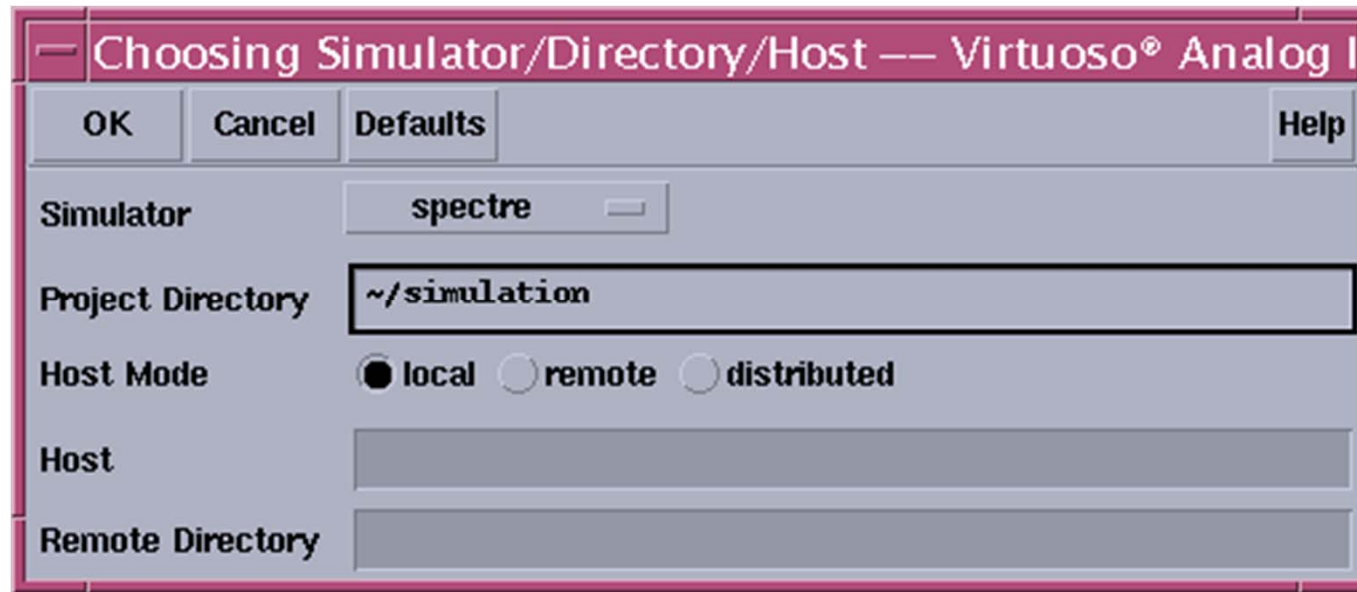
- 启动Analog Design Environment
  - Schematic窗口: Tools->Analog Environment



## 仿真的设置

---

- Simulator设置: Analog Design Environment窗口中
  - Setup->Simulator/Directory/Host
    - Simulator默认会选择spectre

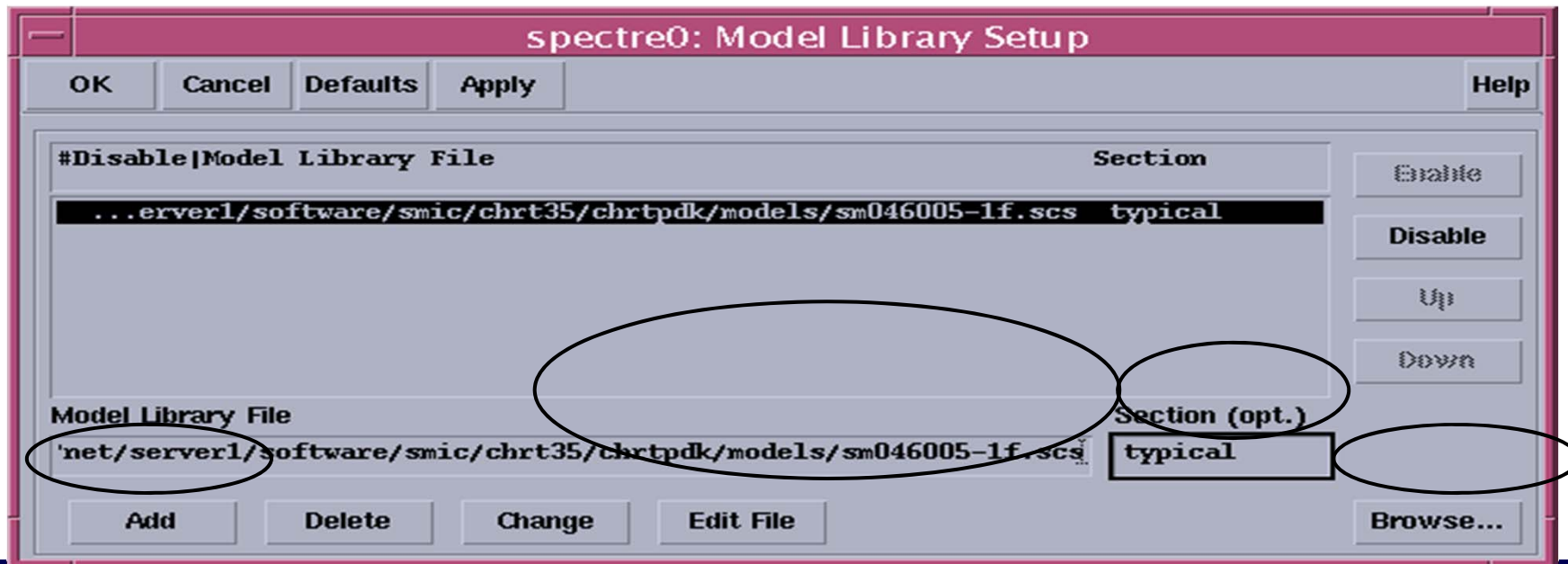


# 仿真的设置

## ● Model设置: Analog Design Environment窗口中

### ● Setup->Model Libraries

- Model Library File: 使用Browse找到模型文件（通常在models目录下）
- Section: typical
- Add
- 多余的库删除



# 仿真的设置

## ● Analysis 设置

### ● Analysis->Choose

● Analysis: dc

### ● Saving Variables

● Component Parameter

● Component Name: /V0

● Parameter Name: dc

### ● Sweep Range

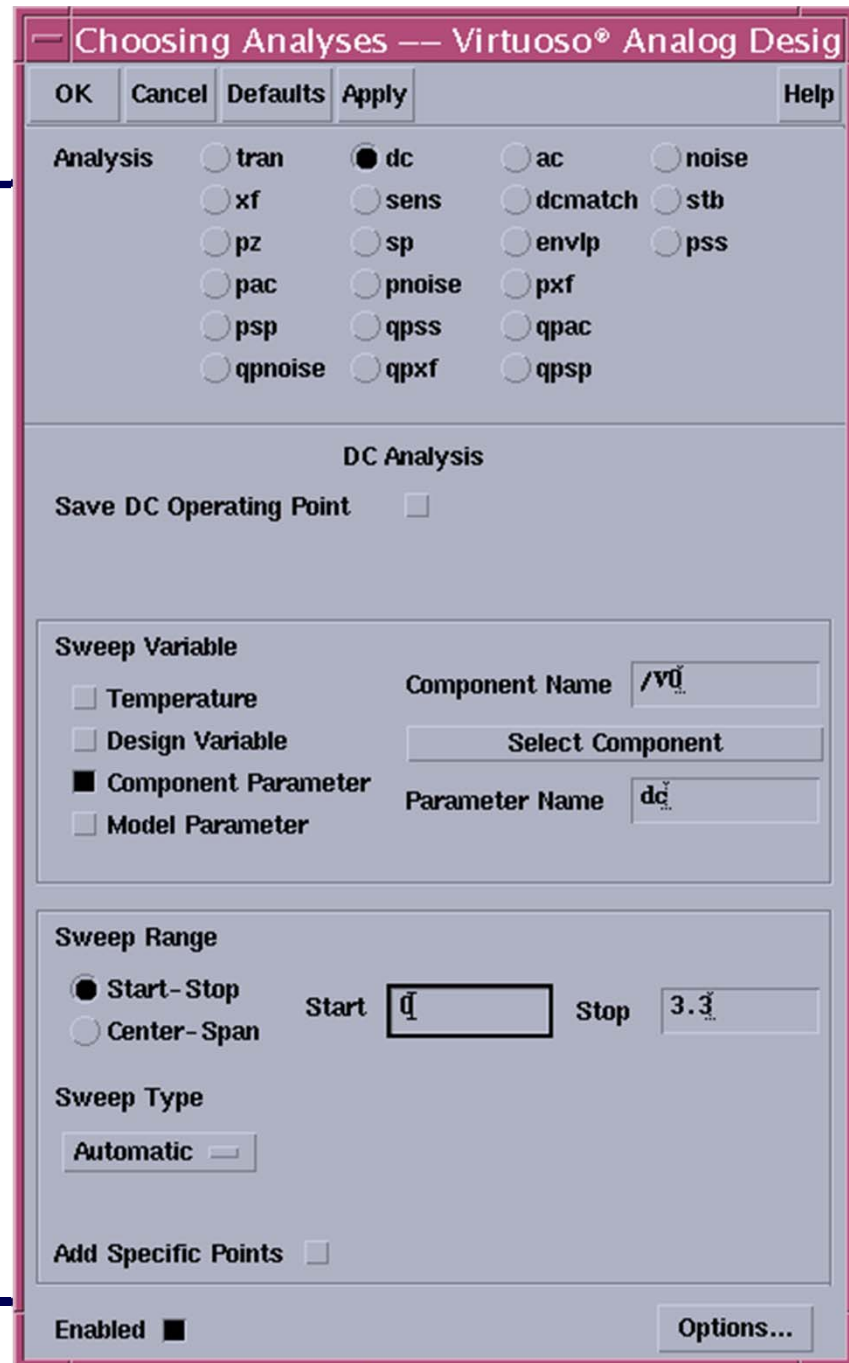
● Start-Stop

● Start: 0

● Stop: 3.3

● Enabled

● OK





# 仿真的设置

---

- 输出结果的显示

- Output->To Be Plotted->Select On Schematic

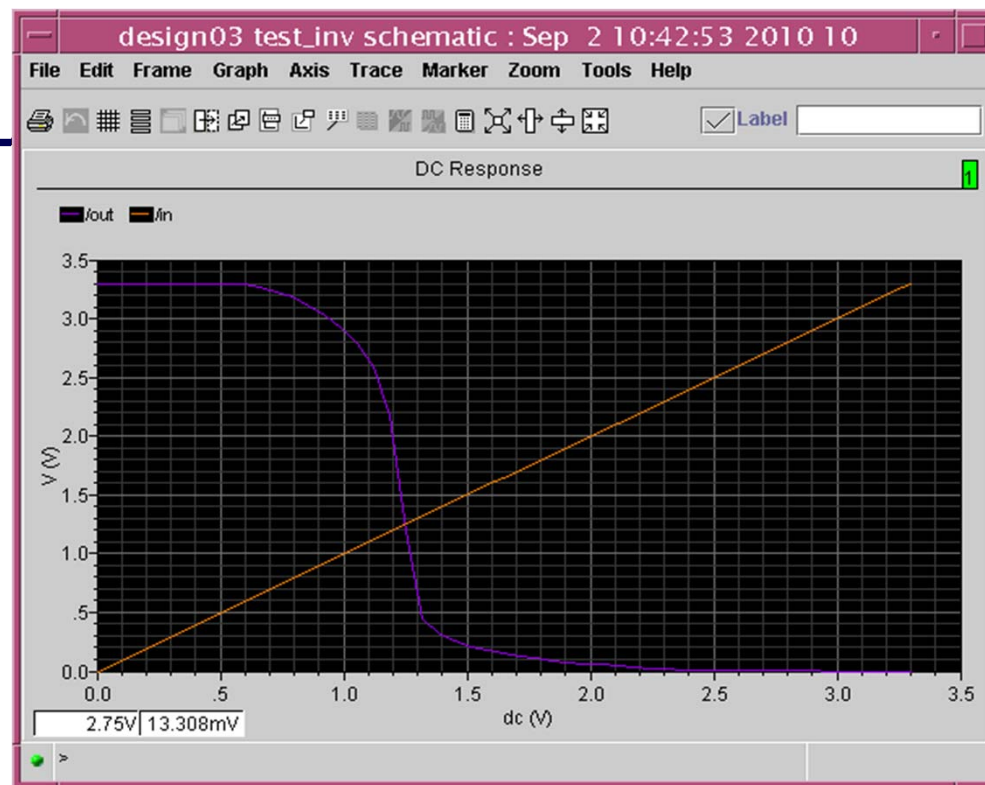
- 在schematic中选择要观测的连线或端点
    - 为便于观测，可以给线加上label

- 注意

- 电压值：选择Net，而不是元件的Terminals！！
  - 电流值：选择Terminal
    - 点击元件：该元件的所有Terminal
    - 点击某个Terminal：此端电流

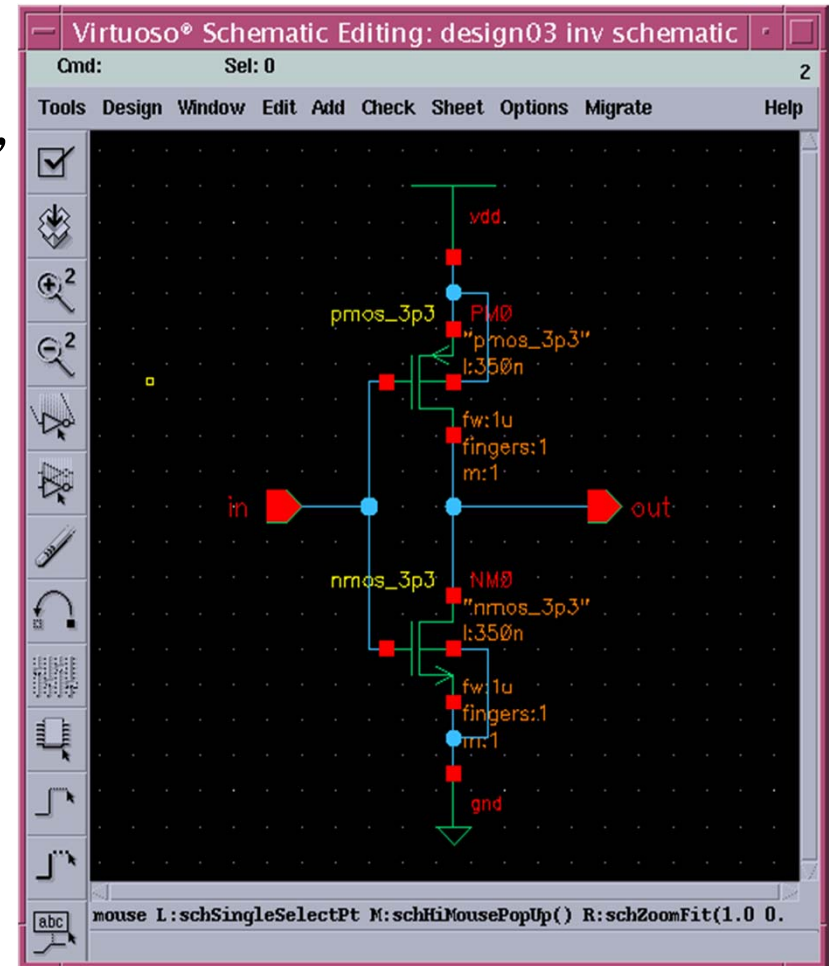
# 仿真

- 运行仿真
  1. 检查并保存Schematic
  2. simulation-> Netlist And Run
  3. 仿真结束，显示波形
- 这是反相器的直流扫描的波形，是输出电压在不同输入电压下的电平值，即我们在cmos课中学习的反相器的输入输出特性曲线
- 输入信号和输出信号交点即为inv的阈值电平，如果器件的电流相同，该电平应该为Vdd/2
- 可以用逻辑阈值点计算噪声容限



## 8. 确定mos管尺寸

- 调整pmos管和nmos管的尺寸比例，观测对直流特性中的阈值电平的变化，即电路的噪声容限的影响
- 根据设计要求，按照反相器阈值点定义的噪声容限最大原则定下mos管尺寸



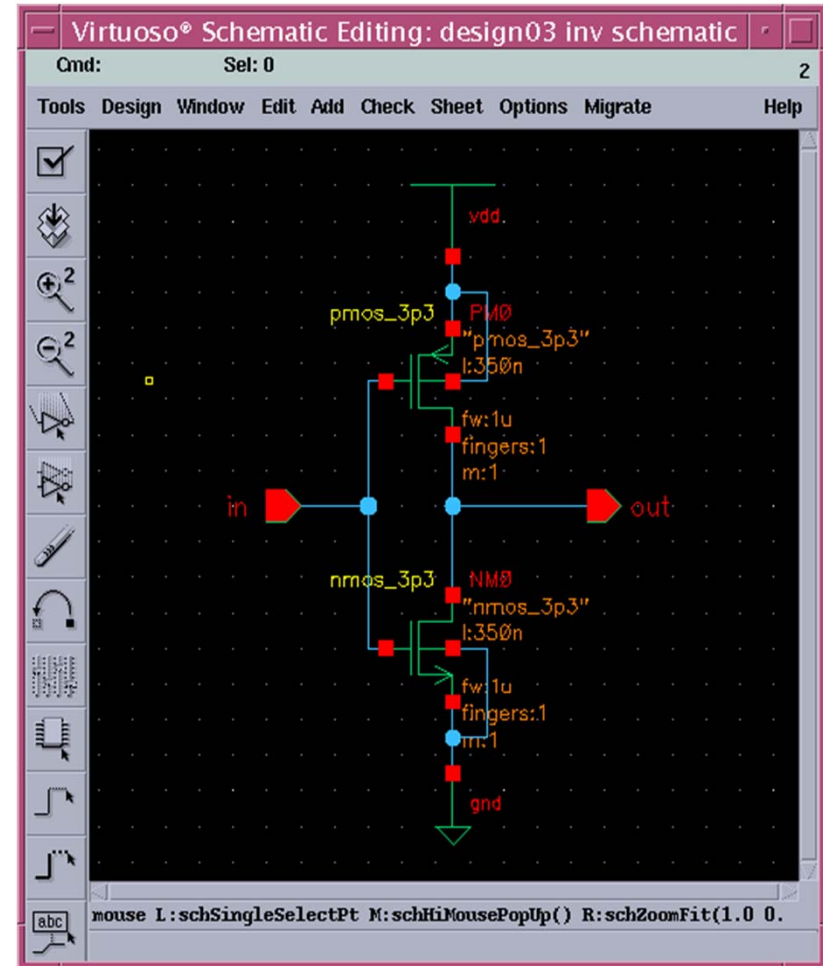
## 9. 瞬态分析

---

- 进行瞬态扫描，观测mos管的大小对延迟时间的影响
  - 扫描类型为tran
  - 把直流源V2换成瞬态源vpulse
- 设置瞬态源的脉宽和周期时间
- 设置瞬态分析时间
- 进行瞬态分析并观察输出波形，利用波形观察器中的标尺测量延迟时间
- 观察vdc节点的电流波形，利用波形观察器中的计算器工具可以计算电源输出的平均电流，乘以电压就得到该时间段内的平均功耗
- 自此，就完成了电路设计，下面可以进行inv电路的版图设计

# 11. 由电路图产生初始版图

- VirtuosoXL为cadence的连接关系驱动定制版图设计工具
- 启动Virtuoso XL
  - 第一步：打开inv的schematic view
  - 第二步： Schematic窗口： Tools-> Design Synthesis->Layout XL
    - Create New
    - OK
    - OK， 弹出Virtuoso XL窗口
  - 在Virtuoso XL窗口中， Design->Gen From Source， 弹出Layout Generation Options对话框（下页）



## ● Layout Generation部分

- 选中I/O Pins, Instances, Transistor Chaining, Transistor Folding

## ● I/O Pins部分

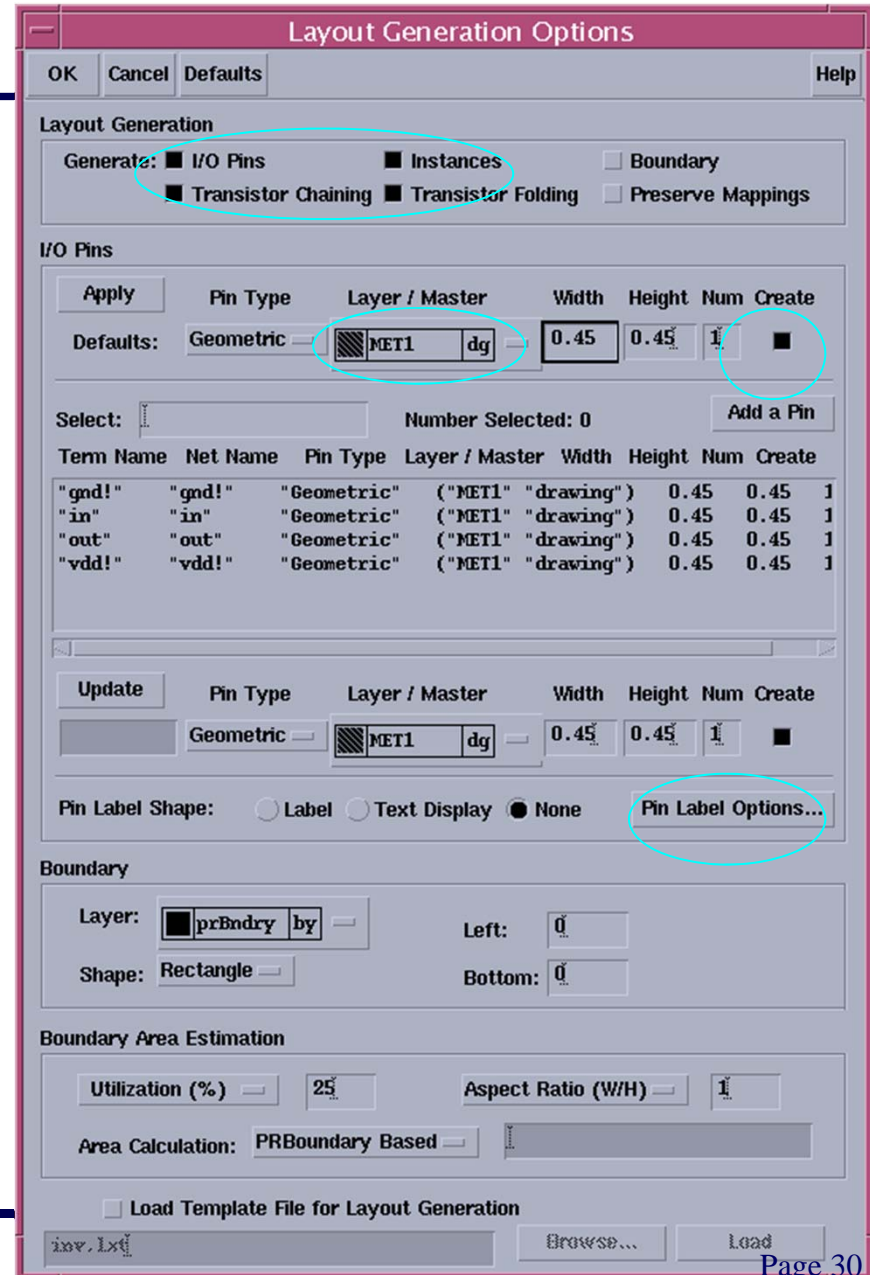
### ● Defaults

- Layer/Master选择MET1层
- 选中Create
- Apply

### ● Pin Label Shape

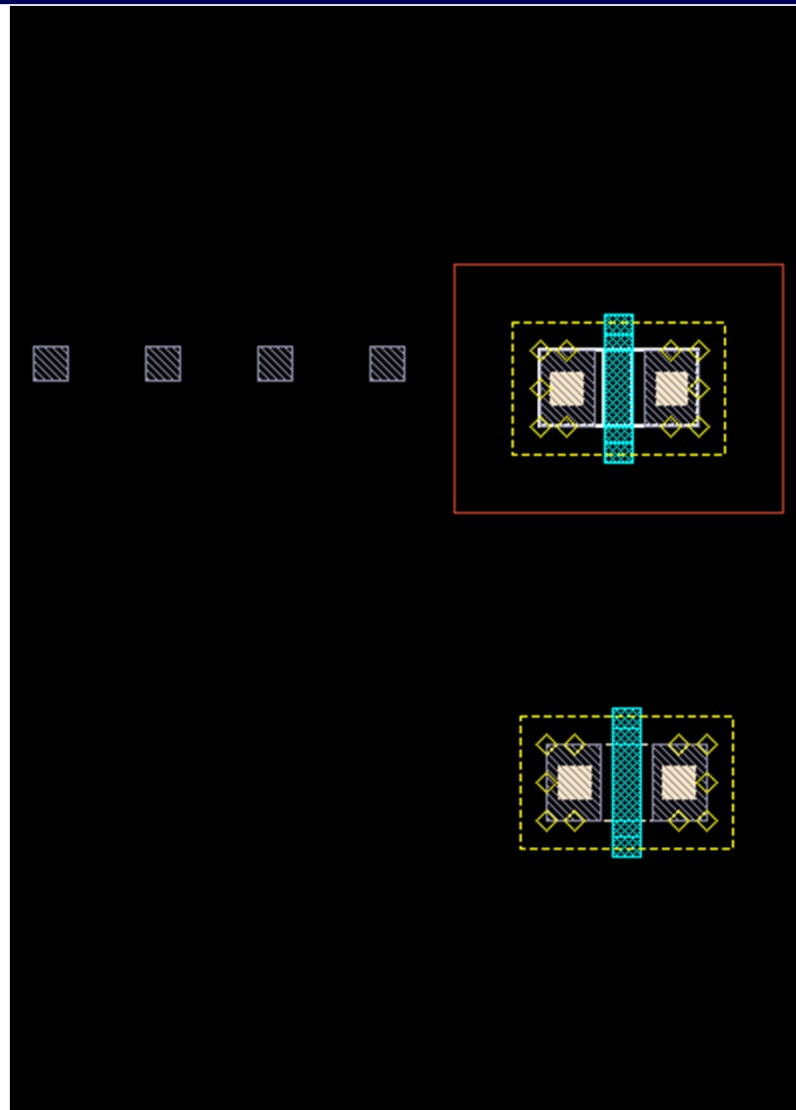
- 点击Pin Label Options, 在弹出的对话框中选中Layer Name后面的Same As Pin

## ● 点击OK后出现下页图, 按照前面的设计要求进行版图设计



## Inv的版图设计

- 在此基础上添加电源线vdd，地线gnd（可参考下页快捷键）
- 按照电路图用相应的层把mos管的各端连线画出来
- 把vdd和N阱相连（用chrt35dg\_siGe库中的M1\_NWELL），把gnd和衬底相连（用chrt35dg\_siGe库中的M1\_PSUB）
- 按照标准单元的画法，单元有统一的高度（这里是13um），统一的vdd和gnd走线宽度（2um）和位置（vdd走线在单元的最上端，gnd在最下端）



## Virtuoso Layout Editor常用操作

---

- Instance（添加元件，快捷键i）
- Rectangle（画矩形，快捷键r）， Polygon（画多边形，快捷键P）， Path（画长连线，快捷键p）
- Copy（c）， Move（m）， Stretch（s）
- Merge（把多个相互重叠的图形合并成一块，M）
- Create Ruler（添加标尺，快捷键k）， Clear All Rulers（K）
- Descend（X）， Return（B）
- Zoom in by 2（^z）， Zoom out by 2（Z）， Fit（f）
- Option→Display（e）中可以设置一些参数
- 在命令过程中可以利用**F3键**打开该命令相关的选项，帮助我们调整命令参数（很有帮助！）



## 12. 版图的验证DRC

- 在Layout窗口中，Verify->DRC

- 在Rules File一项中，填入divaDRC.rul，若该文件不在启动目录下，还应写上路径

- 选中Rules Library选项

- Switch Names一项中选择：

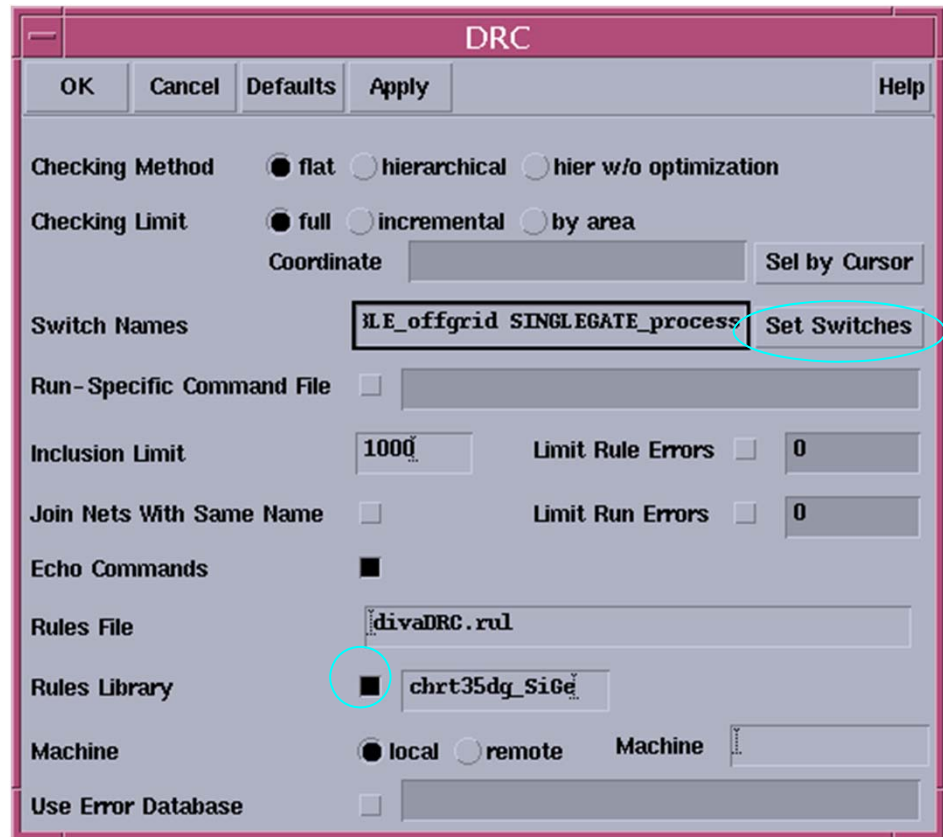
2P4M;

DISABLE\_offgrid;

SINGLEGATE\_process;

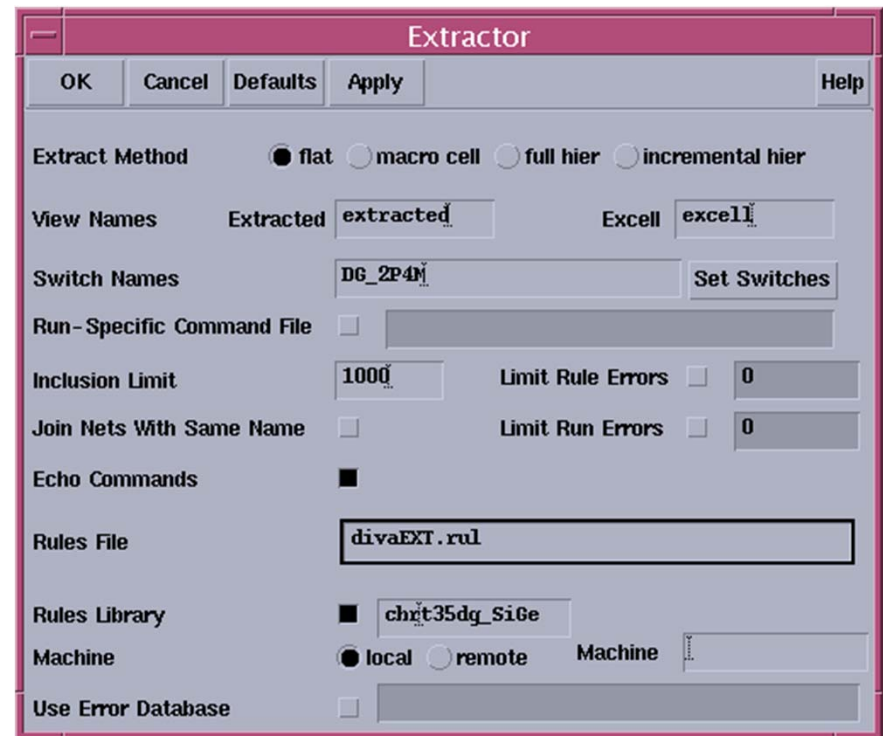
- OK

- 在CIW中可以看到DRC的结果，按说明改掉图中的error，直到Total errors found为0



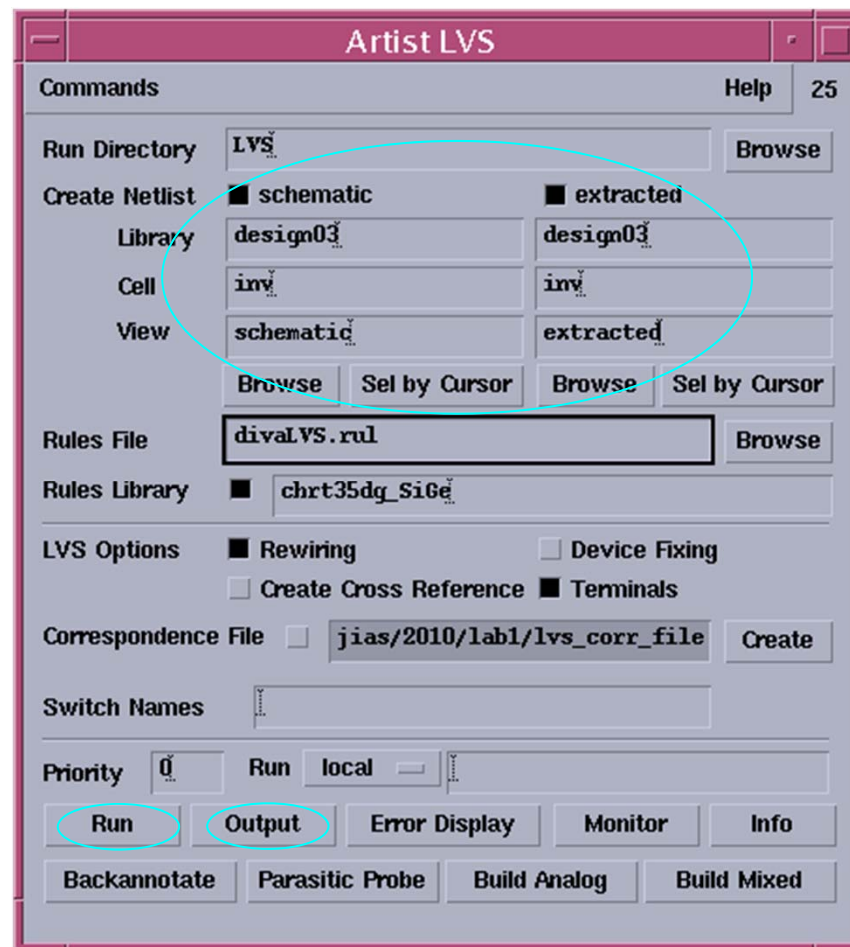
## 13. 版图的器件提取和LVS检查

- 在做LVS之前，要把版图中的管子信息和pin的信息提取出来，这就是Extract
- 在Layout窗口中，Verify->Extract
  - 在Rules File一项中，填入divaEXT.rul，若该文件不在启动目录下，还应写上路径
  - 选中Rules Library选项
  - Switch Names选项中，选择DG\_2P4M
  - OK
- 在CIW中可以看到Extract是否成功，一般情况下Total errors found都为0
- 执行的结果是cell inv产生了一个extracted view



# LVS

- 在Layout窗口中，Verify->LVS
  - Library和Cell中分别填入库名字 design03和单元名字inv，View中对应于schematic的填入schematic，对应于extracted的填入extracted（这些信息也可以通过Browse或者Sel by Cursor得到）
  - 在Rules File一项中，填入 divaLVS.rul，若该文件不在启动目录下，还应写上路径
  - 点击Run，运行几秒后会出现“Analysis Job Succeeded”的提示
  - 点击Output，观看结果。改正版图中的错误，重新Extract，LVS，直至出现“The net-lists match.”（下页）



# LVS通过的报告和版图

```

/home1/Asic/jias/2010/lab1/LVS/si.out
File Help 28
@(#)CDS: LVS.exe version 5.1.0 06/20/2007 02:37 (cicsun11) $

Command line: /cds3/IC5141/IC5141USR5/tools.sum4v/dfII/bin/32bit/LVS.exe -dir /home1/Asic/jias/2010/lab1/LVS -l -s -t /home1/
Like matching is enabled.
Net swapping is enabled.
Using terminal names as correspondence points.

CHRT 0.35um Diva LVS Rules

Net-list summary for /home1/Asic/jias/2010/lab1/LVS/layout/netlist
count
4 nets
4 terminals
1 pmos_3p3
1 nmos_3p3

Net-list summary for /home1/Asic/jias/2010/lab1/LVS/schematic/netlist
count
4 nets
4 terminals
1 pmos_3p3
1 nmos_3p3

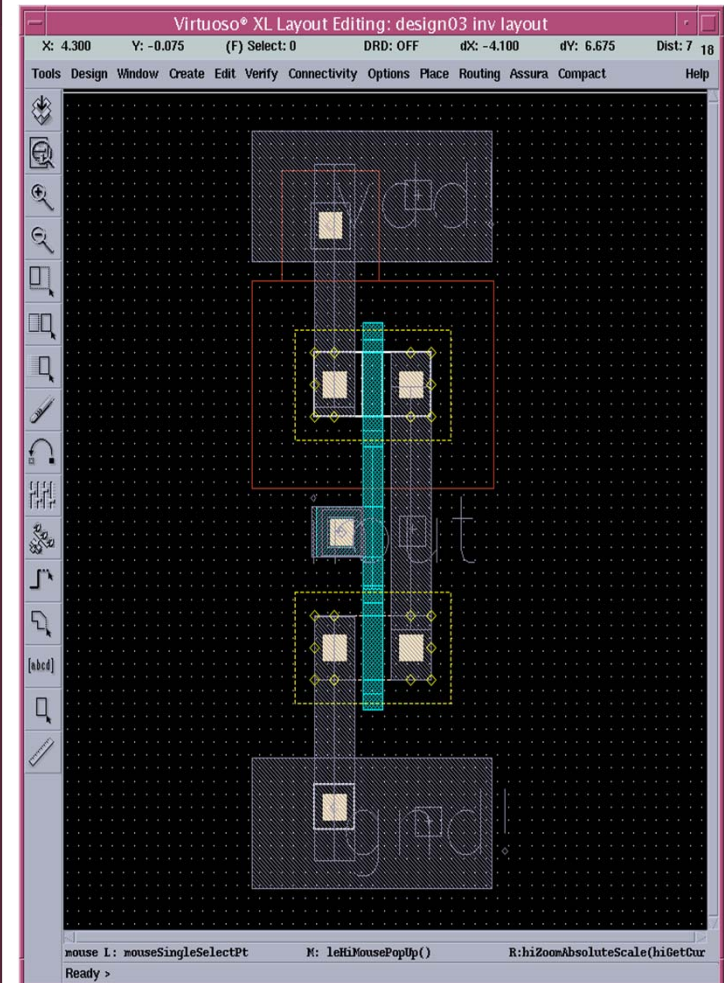
Terminal correspondence points
N1 N1 gnd!
N3 N4 in
N2 N3 out
N0 N0 vdd!

Devices in the rules but not in the netlist:
npoly_u_1k nmos_5p0 pmos_5p0 RF_nmos RF_nmos_BN RF_pmos vpmp lpmp
vpmp_EPI npn_3v npn_5v np_5p0 np_3p3 pn_5p0 pn_3p3 nwp np_3p3_esd
np_5p0_esd pwbmp bmp Schottky_diode pplus_u nplus_u nwell npolyf_u
npolyf_u_sige nploy_u_1k rm1 rm2 rm3_D6_3M rm3_4M rm4_D6 rm4_SiGe pip
sinker_cap mim_2ff mim_4ff inductor Mos_Varactor pn_EPI_Varactor

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

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



# INV设计总结

- 至此，我们完成了一个反相器的设计的全过程
- 在设计过程中，我们学习了电路原理图输入，电路仿真并根据仿真的结果对电路尺寸进行了优化设计
- 对设计好的电路，我们面向Chart 0.35um工艺进行了定制的版图设计，并通过了设计规则检查和LVS检查，说明该版图可以正确加工并实现原理图功能
- 后面还可以进行寄生参数提取和后仿真，得到更接近流片结果的仿真结果，这些内容我们以后学习

