



**国家集成电路人才培养基地
培训资料(1)**

Cadence 入门教程

2006-7-19

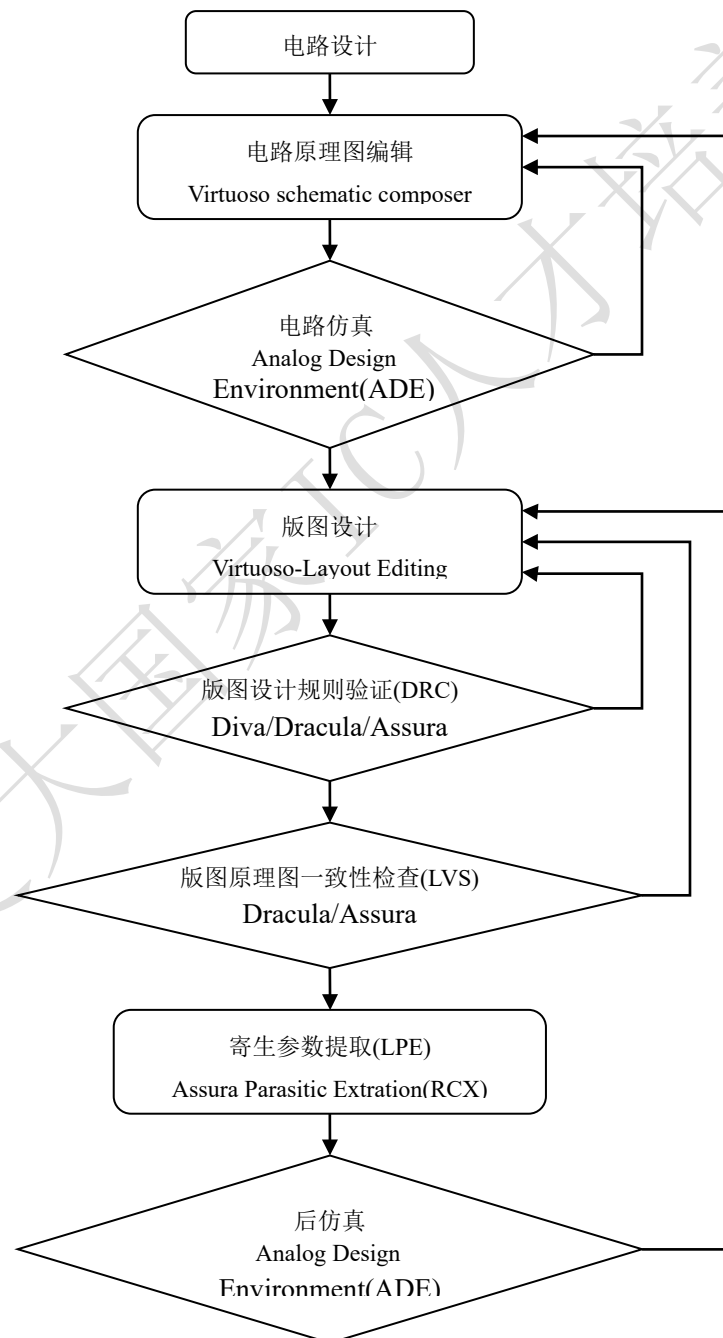
目 录

一、Cadence 概述.....	1
二、运行 Cadence.....	3
2.1 建立个人工作目录.....	3
2.2 启动 Cadence 之前的配置.....	3
2.3 拷贝相关的技术文件.....	3
2.4 启动 Cadence.....	4
2.5 建立个人工作库.....	4
三、电路图的输入——Composer.....	7
3.1 新建原理图.....	7
3.2 添加器件.....	8
3.3 连线.....	9
3.4 设置元件参数.....	10
3.5 放置端口.....	11
3.6 检查并存储.....	11
四、创建 Symbol——Composer.....	13
4.1 打开 inverter 原理图.....	13
4.2 创建 Symbol.....	13
4.3 编辑 Symbol 并保存.....	14
五、电路仿真——ADE.....	15
5.1 创建缓冲器原理图和 Symbol.....	15
5.2 创建仿真电路.....	16
5.3 打开仿真环境.....	17
5.4 设计变量.....	18
5.5 仿真设置.....	18
5.6 选择输出，并保存当前仿真设置.....	20
5.7 进行仿真并查看波形.....	20
5.8 仿真结果分析.....	21
六、版图设计——Virtuoso Layout Editing.....	26
6.1 新建 layout.....	26
6.2 参阅设计规则手册.....	27
6.3 设置 LSW 窗口可见层.....	27
6.3 画第一个反相器.....	28
6.3.1 画第一个晶体管——NMOS.....	28
6.3.2 画第二个晶体管——PMOS.....	31
6.3.4 信号线的连接.....	31
6.3.4 连接电源与地.....	32
6.3.5 做衬底接触.....	33
6.4 画第二个反相器.....	34
6.5 画缓冲器版图.....	35

七、设计规则检查(DRC)——Assura DRC.....	37
7.1 运行 Assura DRC	37
7.2 查找 DRC 错误并修改.....	38
7.3 一些高级 DRC 功能.....	39
7.3.1 屏蔽器件.....	39
7.3.2 屏蔽错误.....	40
7.3.3 把屏蔽掉的错误作为 DRC 运行参数.....	40
八、版图和原理图一致性检查—Assura LVS.....	42
8.1 LVS 前的准备	42
8.2 设置并运行 Assura LVS	43
8.3 查找 LVS 错误并修改.....	44
九、寄生参数提取(PE)——Assura RCX	47
9.1 运行 RCX 前的准备.....	47
9.2 设置并运行 RCX	47
9.3 查看 RCX 结果	49
十、后仿真(PS).....	51
10.1 运行 Hierarchy.....	51
10.2 对不带寄生参数的原理图进行仿真.....	53
10.3 对带寄生参数的原理图进行仿真.....	53

一、Cadence 概述

作为流行的 EDA 工具之一，Cadence 一直以来以其强大的功能受到广大 EDA 工程师的青睐。Cadence 可以完成整个 IC 设计流程的各个方面，如电路图输入(Schematic Input)、电路仿真(Analog Simulation)、版图设计(Layout Design)、版图验证(Layout Verification)、寄生参数提取(Layout Parasitic Extraction)以及后仿真(Post Simulation)。如图 1.1 所示，我们给出了一个简单的模拟集成电路设计流程，以及对应的 Cadence 工具。



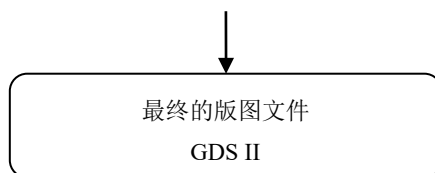


图 1.1 Cadence 模拟电路设计流程

本文针对以上流程，通过设计一个简单的缓冲器的原理图到最终的版图，对 Cadence 的 Composer, Analog Design Environment, Virtuoso, Assura 等各大功能模块逐一做介绍。

此外再介绍一点关于 Cadence 平台的知识。Cadence 开发了自己的编程语言 skill 以及相应的编译器，整个 Cadence 可以理解为一个搭建在 skill 语言平台上的可执行文件集。初学者对此可以不用理会，当用户深入后，可以用 skill 语言对 Cadence 进行扩展。

二、运行 Cadence

2.1 建立个人工作目录

在登陆工作站后，于桌面空白处右键，tools->Terminal，打开终端。默认是在当前登陆的用户目录下，由于实验室机器是多人使用，为了不致混乱，需要建立自己的工作目录。在终端中输入

```
mkdir wavehorse
```

其中 wavehorse 为自己命名的目录名称，你可以用自己名字的拼音或者其它来作为工作目录名称。然后进入自己的工作目录

```
cd wavehorse
```

2.2 启动 Cadence 之前的配置

Cadence 初次启动之前需要如下一些配置文件：

.cshrc 文件：有关一些 Cadence 必需的环境变量，如 Cadence 软件的路径及 license。

.cdsenv 文件：包含 Cadence 各种工具的一些初始设置。

.cds.lib 文件：用户库的管理文件，在第一次运行 Cadence 时会自动生成。

.cdsinit 文件：包含 Cadence 的一些初始化设置以及快捷键设置。

实际上，机房中我们已将各配置文件写好，只要在终端中执行

```
cds.setup
```

Cadence 的相关配置文件就已经自动设置完毕。如果用户在启动 Cadence 后，发现无法使用快捷键，则需要把.cdsinit 从 Cadence 的安装目录中拷贝到自己的工作目录下，在终端中输入：

```
cp /cad/cds/IC5141/tools/dfII/cdsuser/.cdsinit /home/bma/wavehorse
```

请将其中的/home/bma/wavehorse 换成自己的工作目录路径。

2.3 拷贝相关的技术文件

在我们设计电路过程中，需要各种技术文件，这些技术文件一般是由 Foundry 提供。在本教程中，需要如下文件：

TF 文件：TF(Technology File)文件一般由 Foundry 提供，包括了版图设计中的图层信息、符号化器件的定义以及一些针对 Cadence 工具的规则定义，还有版图转换成 GDSII 时所用到的层号的定义。本教程中需要的 tf 文件位于

```
/cad/smic018_tech/Design_Service_Technology/mixed-  
signal/LayoutTechnologyFile/SmicVTTF_LO_SRAM_MR_MM_HV_LC_018.tf
```

display.drf 文件：控制 Cadence 的版图显示。本教程中用到的 display.drf 文件位于

```
/cad/smic018_tech/Design_Service_Technology/mixed-  
signal/LayoutTechnologyFile/display.drf
```

请将以上文件拷贝到自己的工作目录下。

2.4 启动 Cadence

现在，我们可以启动 Cadence 了。在终端中输入命令

```
icfb&
```

出现 Cadence 初始界面，如图 2.1 所示。

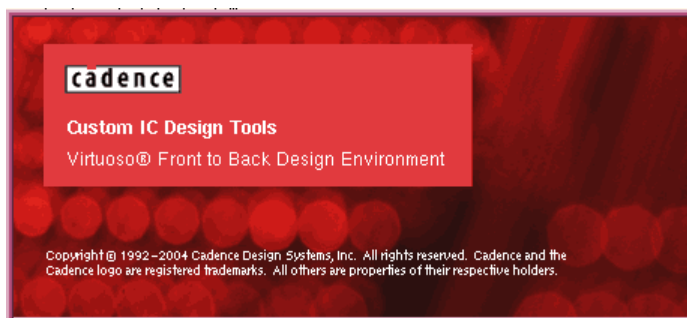


图 2.1 Cadence 初始界面

然后就会打开 Cadence 的主控窗口 CIW(Command Interpreter Window)。如图 2.2 所示：

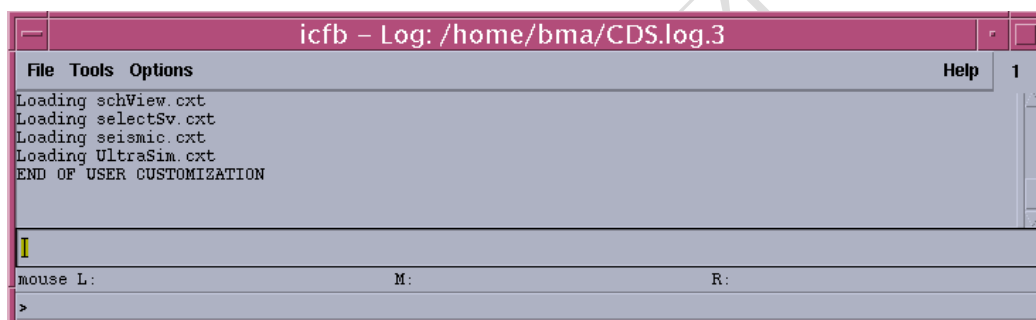


图 2.2 Cadence 主控窗口

这就是 Cadence 的集成设计环境，Cadence 的大部分工具都可以从这里打开。其中最上方是标题栏，第二行是菜单栏。中间部分是输出区域，许多命令的结果在这里显示。一些出错信息也在这里显示，要学会从输出区域中获取相应的信息。接下来一行是命令输入行。Cadence 的许多操作可以通过鼠标执行，也可以通过输入命令来执行。

此外还有一个 What'new 窗口，介绍 Cadence 新版本特性，不必理会，双击左上角将其关闭。

2.5 建立个人工作库

Cadence 是以库来组织文件的。为了使我们的工作和系统自带库区别，我们需要建立自己的工作库。有两种方法来建立新库，一是通过菜单栏 Tools->Library Manager 打开库管理器，另一种是通过 File->New->Library 来建立新库。这里我们用第一种方法建立新库。单击菜单栏 Tools->Library Manager，会打开 LM(Library Manager)窗口，如图 2.3 所示。

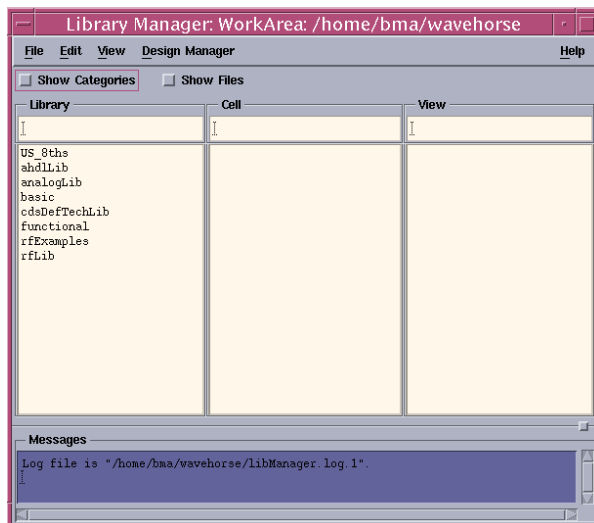


图 2.3 LM 窗口

该窗口列出了当前已有的库。点击 File->New->Library，打开 New Library 窗口，如图 2.4 所示。

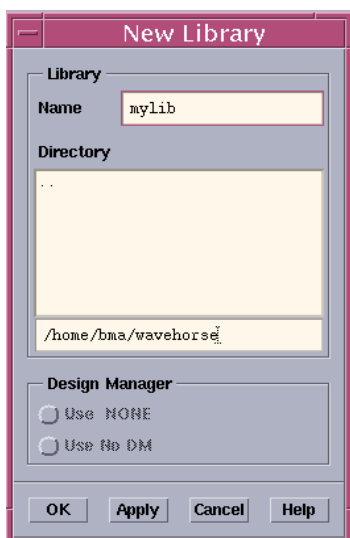


图 2.4 新建库窗口

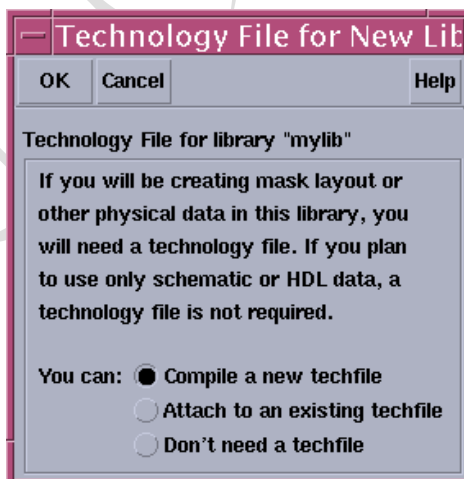


图 2.5 Technology File 设置窗口

在 Name 一栏输入要新建的库名，如 mylib，然后单击 OK 确定。出现 Technology File 设置窗口，如图 2.5 所示。如果不做版图设计的话，就不需要 tf 文件。这里我们选择第一项 Compile a new techfile，单击 OK 确定。出现 Load Technology File 窗口，如图 2.6 所示。

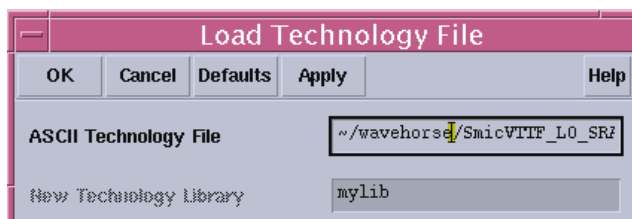


图 2.6 设置 tf 文件路径

这里我们要输入 tf 文件所在路径，例如这里我输入的是

/cad/smic018_tech/Design_Service_Technology/mixed-signal/LayoutTechnologyFile/SmicVTTF_LO_SRAM_MR_MM_HV_LC_018.tf

确定后，就会建立名为 mylib 的新库，Cadence 会在当前的工作目录下自动生成一个新目录 mylib 以存放和库 mylib 相关的文件。

三、电路图的输入——Composer

本章将通过画一个 CMOS 反相器来简单的介绍电路图设计流程。

3.1 新建原理图

类似于新建一个库,有两种方法可以新建原理图,一是通过库管理器,另一种是通过 CIW 菜单新建。这里我们直接通过 CIW 来新建原理图。

在 CIW 窗口中, File->New->CellView, 弹出新建对话框, 如图 3.1 所示。

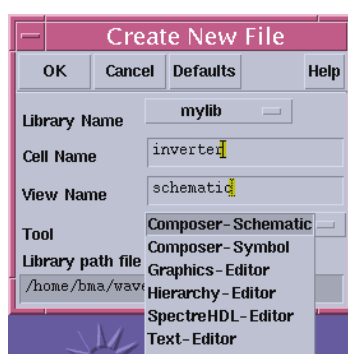


图 3.1 新建原理图

于 Library Name 栏选择自己的工作库, 如 mylib, 在 Cell Name 栏输入原理图名字, 如 interver, 于 Tool 栏选择电路编辑工具 Composer-Schematic, 此时 View Name 栏自动变为 schematic。最后单击 OK。这样就会弹出 Composer 主界面。如图 3.2 所示。

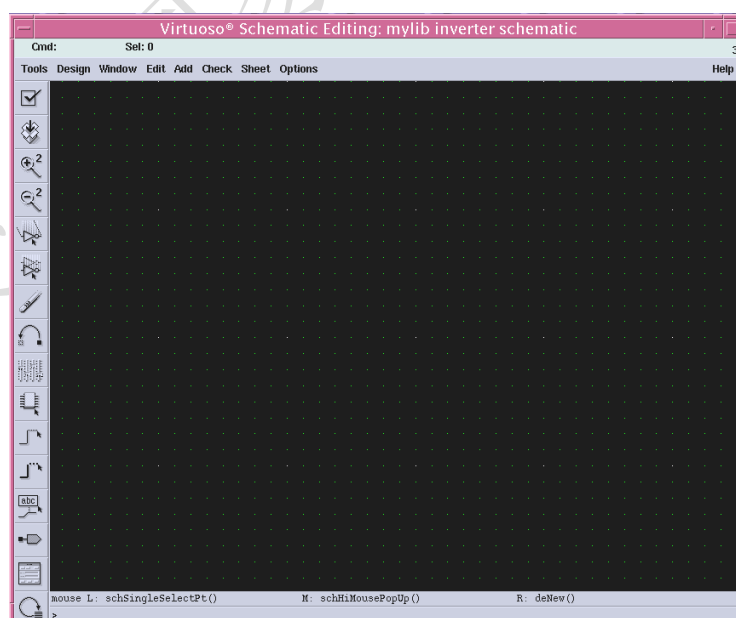


图 3.2 Composer 主界面

Composer 主界面包括: 标题栏, 菜单栏, 工具栏, 状态栏(第二行), 提示区(就是最底下那行)以及最大的那个工作区。标题栏和菜单栏没什么好说的, 状态栏会提示当前的命令

以及所选择的物体个数，提示区会告诉你当前应该做什么事。作为初学者，在设计电路过程中应该要仔细阅读提示区中的信息。此外，

注意：①Composer 中的多数命令会一直保持，直到你调用其它命令替代它或者按 Esc 取消，尤其是在执行 delete 命令时，忽视这一点很可能会误删除，一定要多加小心！Composer 的 Undo 操作默认只能进行一次(可以在 CIW 窗口的 Option->User Preferences 中修改，最多可以是 10)。所以每完成一个命令，记着按 ESC 取消当前命令。

②点击工具栏的 zoomin 和 zoomout 按钮可以放大缩小电路图。键入快捷键 f 可以使电路图自动缩放到合适大小。

③编辑电路图过程中注意要及时保存，保存方法是菜单栏->Design->Save，也可以键入快捷键大写的 S(Shift+s)来保存。

3.2 添加器件

现在，我们要开始画一个标准的 CMOS 反相器。一个反相器包括 PMOS, NMOS, VDD, GND。

添加器件有三种方法，①菜单栏->Add->Instance，②键入快捷键 i，③工具栏 Instance，弹出的对话框如如图 3.3 所示。

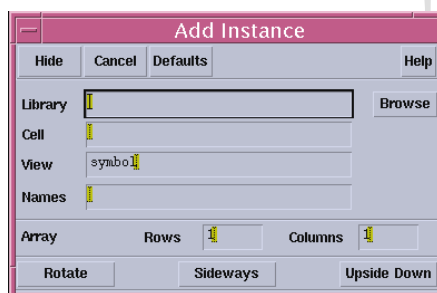


图 3.3 添加器件窗口

点击 Browse，弹出库浏览器，如图 3.4 所示。选中 Show Categories 可以分类显示器件，方便我们快速找到所要的器件。

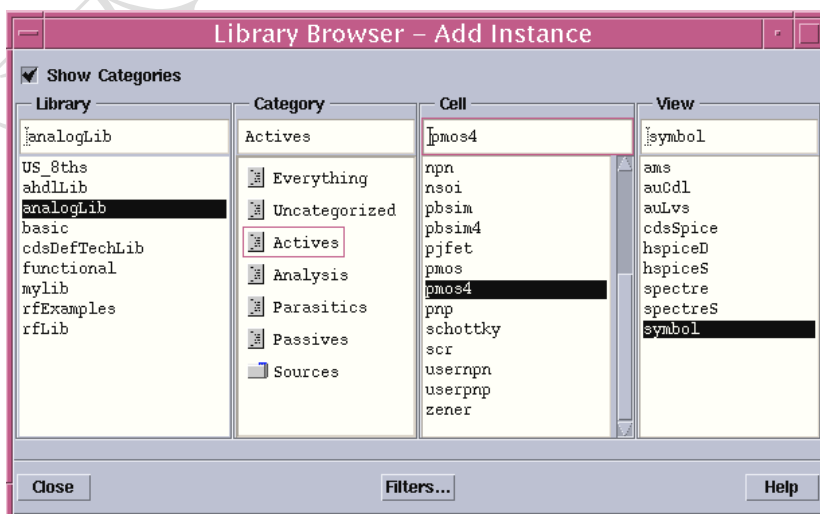


图 3.4 库浏览器

依次点击 analogLib->Actives->pmos4->symbol，再单击 close。刚才的添加器件窗口发生

变化,如图 3.5 所示。可以发现 Library, Cell, View 等都自动填上了相应的信息。同时多出了一些参数列表(拖动滚动条可以看到更多)。点击 Hide 隐藏当前窗口,此时鼠标对应有一个 PMOS 的 symbol,此时按 r 键,可以旋转 PMOS。移动 PMOS 到合适的位置点击鼠标左键将其放下。如果要放置更多的 PMOS,继续点击鼠标左键,否则按 ESC 取消当前的放置器件命令。

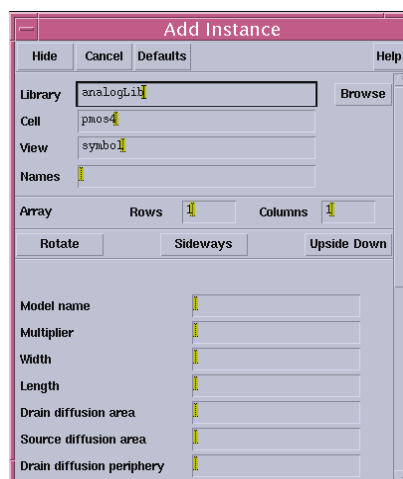


图 3.5 选择了元件后的添加器件窗口

继续放置 NMOS 晶体管、电源与地。对应的器件名称为 nmos4, vdd, gnd。放置完所有器件后的原理图如图 3.6 所示。注意, vdd 与 gnd 仅仅是全局电源与地标识,并不是独立电源器件, vdd 并不能提供电源。仿真时必须有 gnd, 否则仿真不收敛。

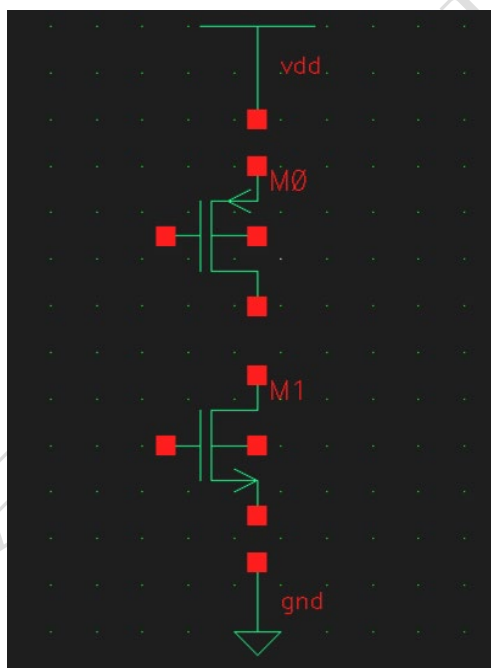


图 3.6 放置完电源与地

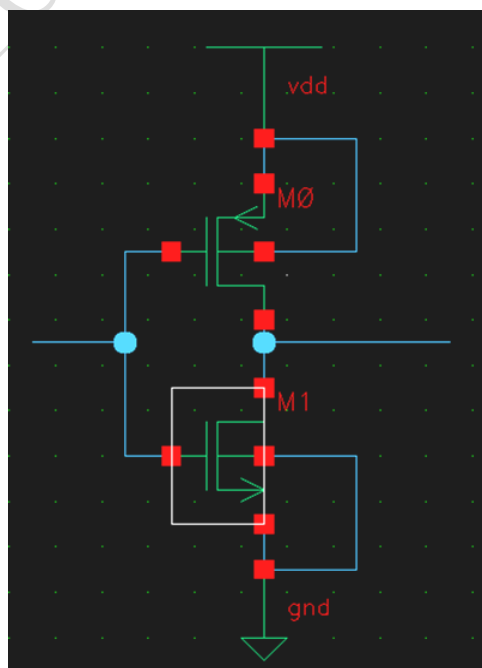


图 3.7 连线

3.3 连线

现在要用导线把器件连起来。画导线的方法有三种,①菜单栏->Add->wire(narrow),②

键入快捷键 w，③工具栏 wire(narrow)。注意区别 **wire(narrow)**与 **wire(wide)**，**wire(narrow)**表示普通连接导线，而 **wire(wide)**表示总线连接。总线连接的快捷键是大写的 W。

进入连线命令后，于起点单击左键，再于终点单击左键。画完一段导线后，此时并没有退出画线命令，可以继续画连接线，直到画完所有的连接线后，按 ESC 退出画线命令。连好线的电路图如图 3.7 所示。其中左右两条水平导线是后面连接端口用的。

还可以对画好的线进行命名，键入快捷键 l，在弹出的对话框中输入线名，比如 a，点击 Hide，然后将字母 a 移动到要命名的线附近点击左键放下，如果名字离线较远，则要求再单击所要命名的线。

3.4 设置元件参数

现在需要设置元件参数，有三种方法，①菜单栏-> Edit-> Properties-> Objects，再点击要修改参数的元件，②先选中器件，再键入快捷键 q，③选中器件，再点击工具栏 Property。

参数可以是以下三种形式的各种数学组合表达式，①变量，②常量，③skill 语言函数。变量作参数会在仿真时用到。常量和 skill 语言函数作参数，在下面就会用到。

例如，单击 PMOS 选中它，这样 PMOS 会被一个白色方框包围。然后键入快捷键 q，会弹出属性编辑对话框。这里我们需要填上 model name，以及 PMOS 的栅长和栅宽。栅长我们设为常量 0.18u(注意 u 是小写!)，而栅宽我们设为函数 pPar("wp")，注意大小写不能错。当然也可以设一个固定的尺寸，但这样就不能利用参数修改晶体管的栅宽了。pPar 函数就是把 wp 作为传递参数，在其它电路图中调用这个电路时对 wp 赋值，就相当于给这个 PMOS 的栅宽赋值，这样做的目的是为了层次化设计。在后面仿真时大家会更加明白这一点。Composer 会根据数值大小自动变换单位。如图 3.8 所示。

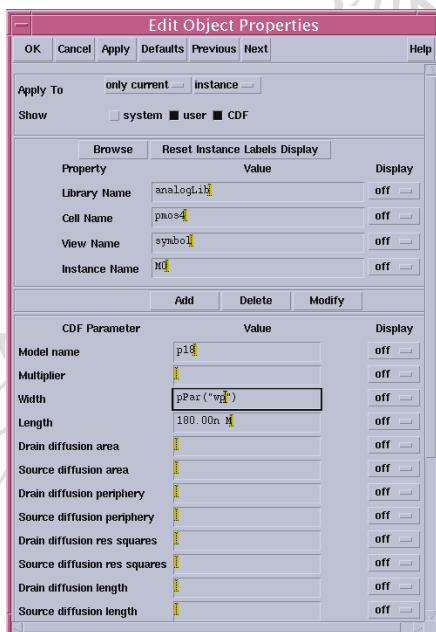


图 3.8 设置元件参数

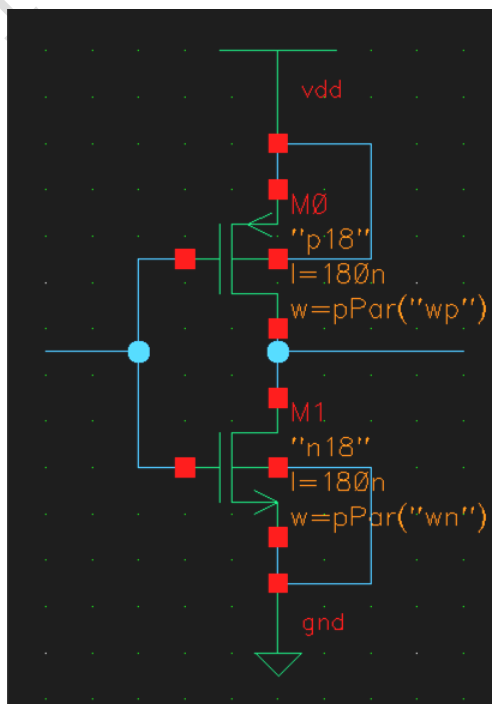


图 3.9 设置好元件参数的电路图

模型名我们填 p18，这是因为我们这里采用的 Spice 模型是由 SMIC 提供的，对应 PMOS 的模型有 p18 和 p33 两种，18 代表电源电压为 1.8V。在后边仿真的时候我们还要再设 SpiceModel 文件的具体路径。如果需要查看其它模型名以及具体的模型参数，可以参阅如下

文件，用任一文本编辑器打开即可。

/cad/smic018_tech/Process_technology/Mixed-Signal/SPICE_Model/ms018_v1p6_spe.mdl

同样的方法继续设置 NMOS 参数，只是模型名为 n18，栅长为固定值 0.18u，栅宽设为 pPar("wn")。设置好参数后的电路图如图 3.9 所示。

注意：①设置参数时不要自己输入单位，系统会自动加上。比如 0.18uM 是错误的写法。如果非要自己写单位，也要和数值之间留一个空格，否则系统会把 M 识别为变量。

②器件的参数也可以在放置时就设置好。

3.5 放置端口

完成以上工作后，还必须放置 I/O 端口以标明电路的输入输出。放置端口有三种方法，①菜单栏->Add->Pin，②键入快捷键 p，③点击工具栏 Pin。

执行放置端口命令后，会弹出如图 3.10 所示对话框。于 Pin Names 栏输入端口名，比如 Vin，于 Direction 栏选择 input，点击 Hide，然后将端口放到反相器的左边输入线上。同样的方法再放置输出端口，端口方向要改为 output，名称为 Vout，将其放在反相器右边的输入线上。最终的完成图如图 3.11 所示。

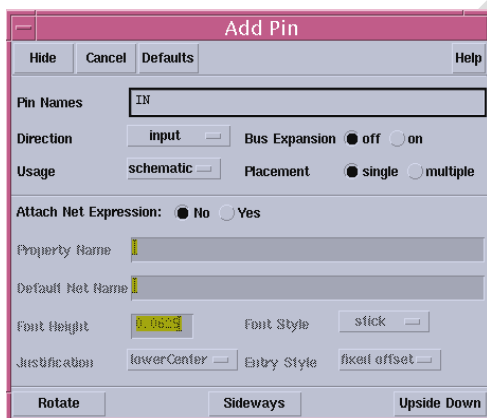


图 3.10 放置端口

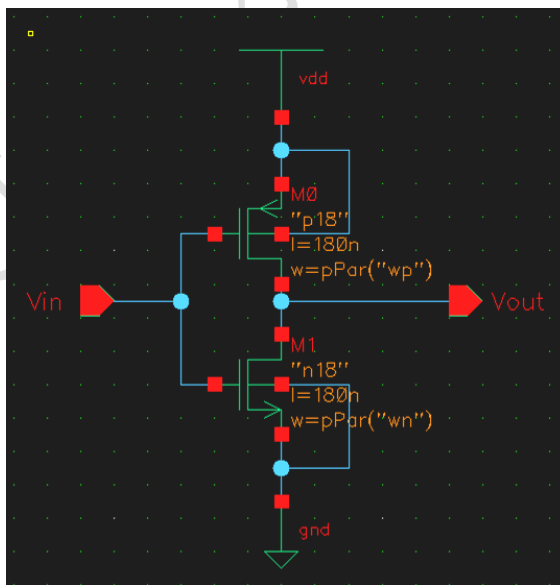


图 3.11 最终的电路图

3.6 检查并存储

设计完成的电路图需要经过检查方能进行仿真。单击菜单栏->Check and Save 或者键入快捷键大写的 X，可以对电路进行检查并存储。

检查后如果有错会在 CIW 窗口上显示错误或警告信息。如果没错，则如图 3.12 所示：

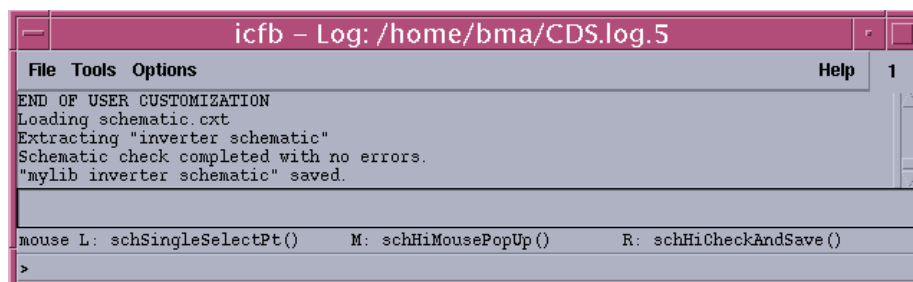


图 3.12 检查电路后 CIW 中的提示信息

检查无误后可以关闭 Composer 了。

四、创建 Symbol——Composer

现在我们要对上一章中画的反相器创建 Symbol，这样做的目的是为了在更大的电路用到我们前面所画的反相器时，可以用这个 Symbol 来代替。

4.1 打开 inverter 原理图

CIW 窗口菜单栏->Open，弹出打开对话框，选择自己的库，然后选择器件 inverter，再于 viewname 栏选 schematic，点击 OK 打开上一章画好的反相器原理图。

4.2 创建 Symbol

Composer 窗口菜单栏->Design->Create CellView->From Cellview，弹出 Cellview from Cellview 窗口，如图 4.1 所示。

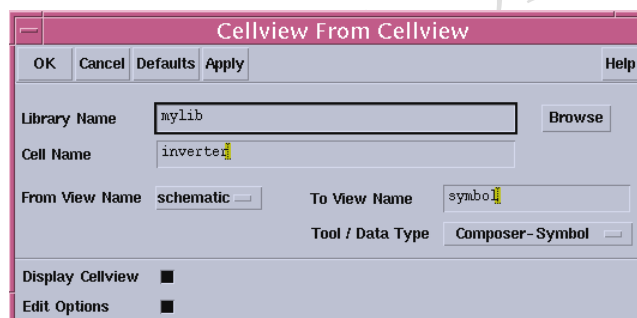


图 4.1 创建 Symbol

其中 Library Name、Cell Name 等栏已经自动填好，确认 To View Name 栏是 symbol，点击 OK。弹出创建 Symbol 选项窗口，如图 4.2 所示。

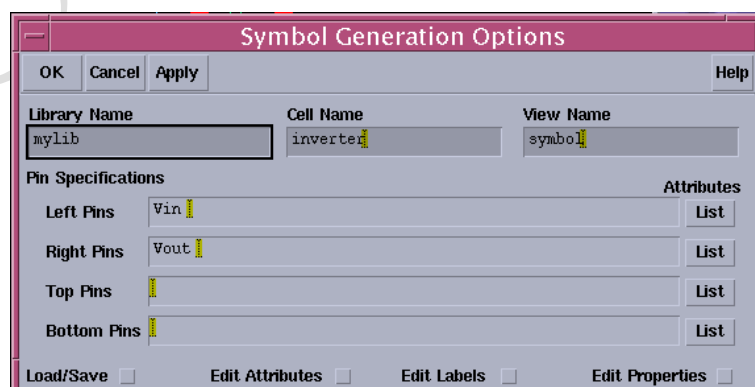


图 4.2 创建 Symbol 选项窗口

这里已经自动识别出电路原理图中的输入输出端口，默认输入在左，输出在右，点击 OK。显示出 Symbol 编辑窗口，如图 4.3 所示。

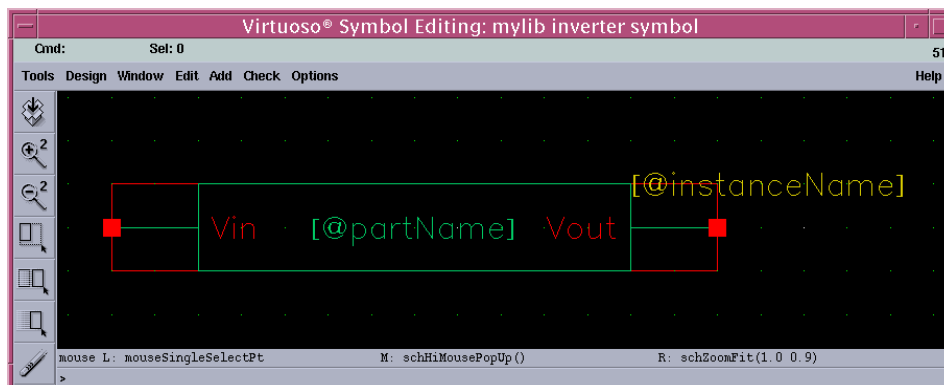


图 4.3 Symbol 编辑窗口

默认生成的反相器 Symbol 是一个绿色矩形框，引脚按刚才编辑好的方式左右排列。红色矩形框代表调用这个模块时点选的区域，也就是说鼠标点到此区域范围内才可以选中这个 Symbol。图中所有元素均可修改，但我们一般只改绿色矩形框。

4.3 编辑 Symbol 并保存

默认的 Symbol 是一个比较大的矩形。对于反相器，我们习惯用一个三角形再加小圆圈来表示。

选中绿色矩形框，delete 之，然后 Add->Shape>Polygon，在刚才矩形框的位置画一个三角形。鼠标在三个端点点击 3 次即可。注意在三角形右边留出画圆圈的位置。

再 Add->Shape>Circle，先于圆心位置单击左键，再移动鼠标，得到合适的圆的半径后左键确认。

再把图中的输入输出端口以及 partname 和 instanceName 移动到合适位置。其中，@instanceName 代表以后调用此反相器时的编号，@partName 代表对应的 schematic 的名字，一般不用改。最后再把红色框大小修改合适(框住三角形和端口)。

最终的 symbol 如图 4.4 所示。

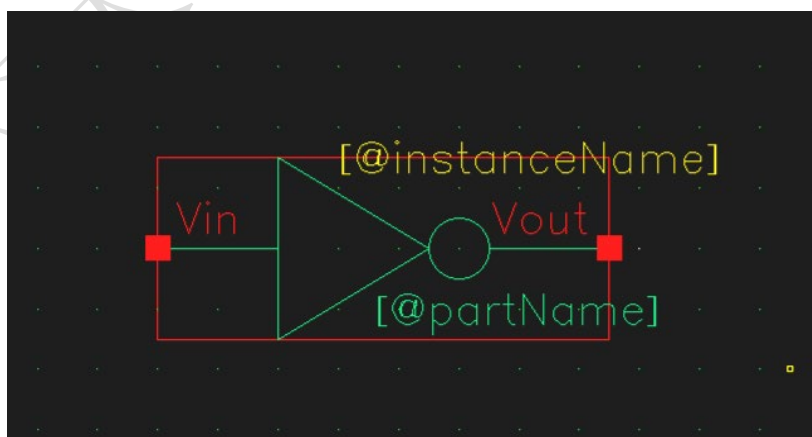


图 4.4 反相器最终的 Symbol

画好的 Symbol 需要检查保存。Design->check and save，检查结果显示在 CIW 窗口中。

五、电路仿真——ADE

现在我们用画好的反相器的 symbol 组成一个缓冲器(buffer)进行仿真，通过对 buffer 做瞬态分析、DC 分析、AC 分析，分别得到该 buffer 的延迟时间、输入输出特性以及小信号频率响应。

对电路进行仿真需要加激励信号，而加激励信号有两种方法，一种是在原理图中直接加入信号源元件，另一种是在仿真环境窗口(ADE)中对输入端口加激励。这里我们介绍的是第一种方法，第二种方法将在后续实验中介绍。

5.1 创建缓冲器原理图和 Symbol

缓冲器是由两个反相器组成，利用前边的方法，新建一个 cellview，画出缓冲器原理图，如图 5.1 所示。

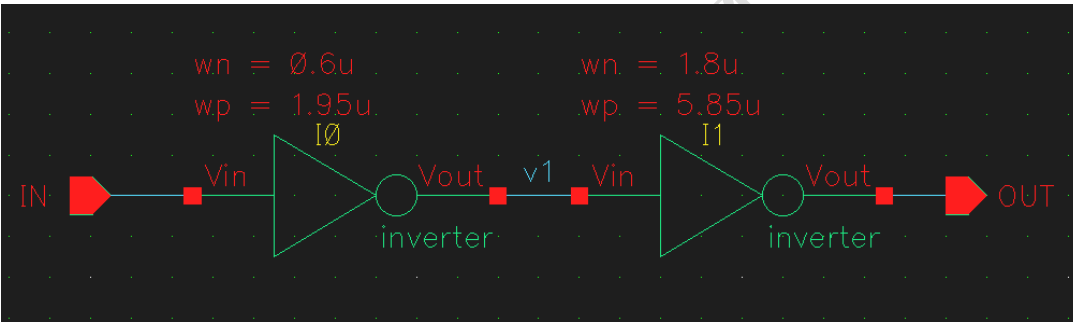


图 5.1 buffer 原理图

其中反相器我们直接调用了上一章中画好的 Symbol(注意是在自己的库中)。选中 inverter，键入 e，再点 OK，可以显示和 Symbol 对应具体的 schematic，但是这时只能看，而不能修改。Ctrl+e 退出该 Symbol。

另外我们把缓冲器的输入输出两条线命名为 IN 和 OUT，把第一级反相器的输出线命名为 V1。这样是为了在仿真时显示得更清楚。给连线命名的方法是：键入快捷键 l，弹出连线命名窗口，于 name 栏输入线名，然后点击 Hide，将名字移到要命名的线附近单击放下。

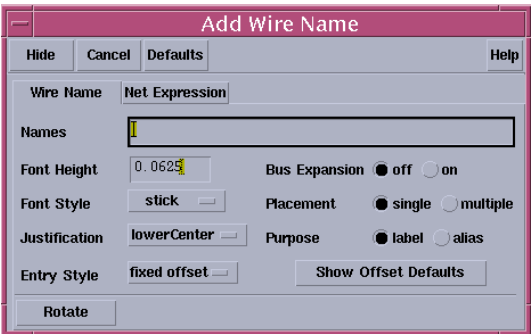


图 5.2 连线命名窗口

设置 inverter 参数。选中 inverter，键入 q，就会弹出反相器的属性，如图 5.3 所示。这里我们需要分别设 wn 和 wp 的值。回忆 3.4 中给两个 MOS 管的栅宽设置传递参数，其实我

们是给反相器的 NMOS 和 PMOS 的栅宽赋值。将 Display 设为 both，可以在原理图上显示出参数值，当然你也可以不让其显示。

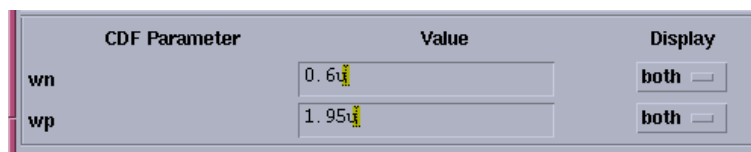


图 5.3 inverter 的参数传递

回忆第四章内容，我们由缓冲器的原理图再生成 Symbol。缓冲器的 Symbol 我们习惯用一个三角形来表示，如图 5.4 所示。

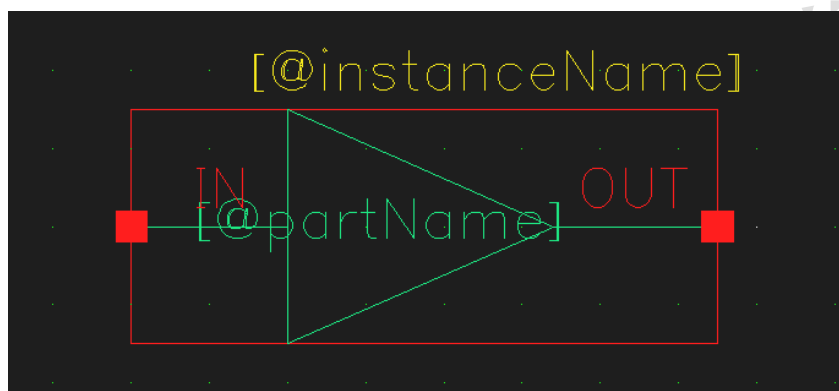


图 5.4 缓冲器 Symbol

5.2 创建仿真电路

现在新建一个电路图，名为 `buffer_test`，注意选为自己的库。画出测试电路，如图 5.5 所示。

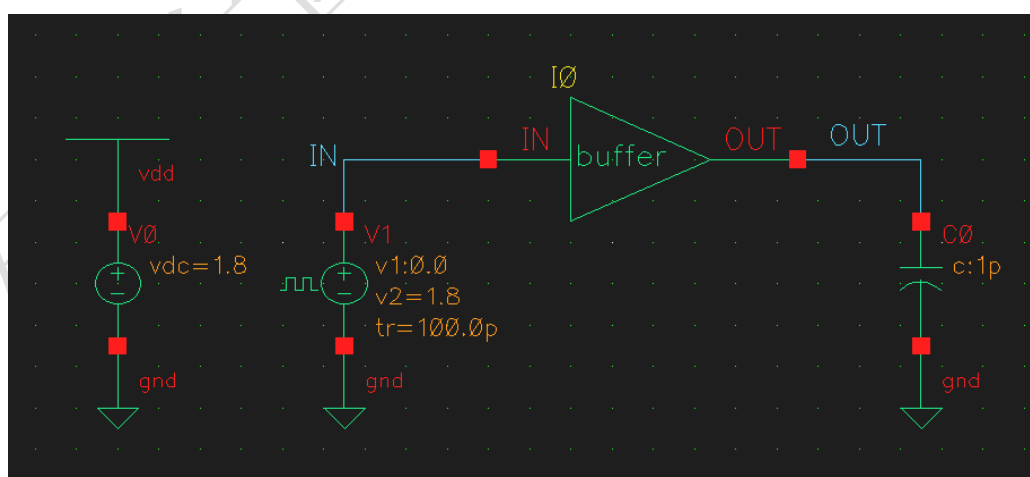


图 5.5 buffer 的仿真电路

①独立电源 `vdc` 也是在 `analoglib` 库中，将其属性中的 DC voltage 设为 1.8。

②另一个激励信号是方波源，对应器件名称为 `vpulse`，也位于 `analoglib` 库中。方波源的属性设置如图 5.6 所示。方波上升下降时间为 0.1n，周期为 10n，脉冲宽度为 4.9n，`voltage1`

设为 0, voltage2 设为 1.8。因为我们还要作 AC 分析, 交流电压幅值为 1(这样测得的输出直接就是小信号增益)。为了得到 inverter 的输入输出特性, 我们把直流电压设为变量 vin。

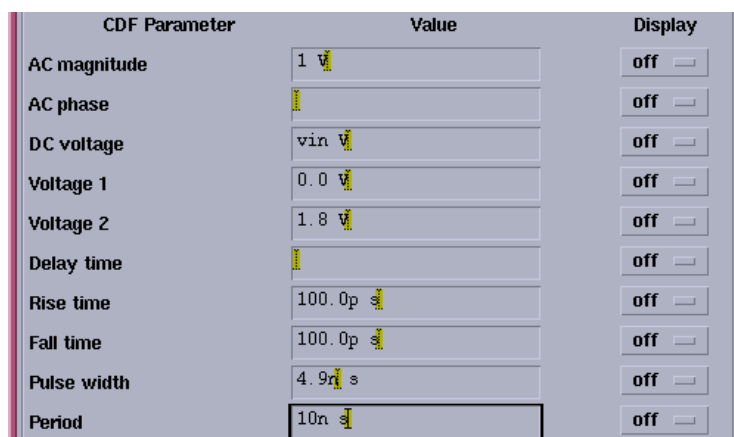


图 5.6 方波信号源的参数设置

对于初学者需要强调的一点, 这里设置的方波电压、直流电压、交流电压是相互独立的, 分别用于瞬态分析、dc 分析、ac 分析(但某些情况下是有联系的)。

5.3 打开仿真环境

Composer 菜单栏->Tools->Analog Environment, 打开仿真窗口(简称为 ADE 窗口), 如图 5.7 所示。其中比较重要的常用按钮已经标明。

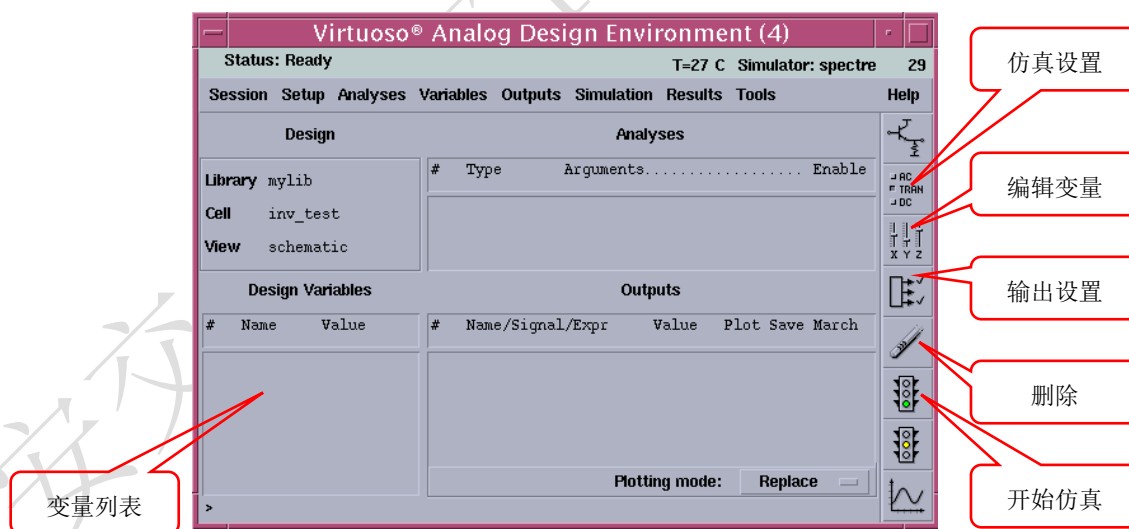


图 5.7 仿真环境窗口

仿真时需要进行一些诸如仿真库文件路径、结果存储路径、仿真器选择等的设置, 相关设置在 setup 中进行设置。这里我们只需要设置仿真库文件路径(仿真库文件记录着不同工艺角的参数, 并指明了各元件类型的 model 文件所在路径), 其它均为默认设置。

于 ADE 窗口, Setup->Model Libraries, 打开 Model 库设置窗口, 如图 5.8 所示。单击“Browser”, 打开浏览窗口, 如图 5.9 所示, 双击列表里的选项可以进入对应的目录, 双击../可以进入上一级目录。我们所使用的 model 文件是由 smic 提供, 其路径如下:

/cad/smic018_tech/Process_technology/Mixed-

Signal/SPICE_Model/ms018_v1p6_spe.lib

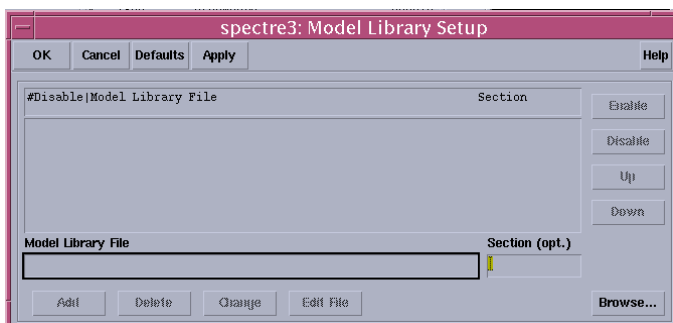


图 5.8 Model 库设置窗口



图 5.9 选择 Model 文件

选好后点击 OK 确认，然后再于 Model 库设置窗口的 Section 栏填写工艺角，这里我们填 tt(即 NMOS 和 PMOS 速度均为典型值)，然后再点 Add 按钮，将当前的仿真库文件添加进列表，点击 OK 退出。

5.4 设计变量

仿真环境菜单栏->Variables->Edit, 或者直接点击右侧的工具栏中的 Edit Variables 按钮，弹出如图 5.10 所示窗口。点击按钮“Copy From”，就会列出我们前面在测试电路中添加的变量 vin。选中该变量，我们设置一个初始值 0.9。之所以是 0.9，是因为当反相器的输入为 0.9 时两个 MOS 管都会导通，此时反相器实际起的是放大器的作用，而我们作 AC 分析就是要得到 MOS 管处于饱和区时它的频率响应。需要强调的是，AC 分析是建立在一定的工作点上的，而电路的工作点正是由电路图中所设置的电源 DC 值以及这里的变量初始值决定的。

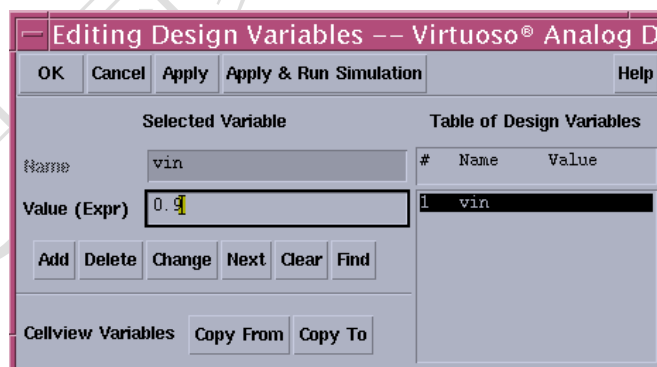


图 5.10 变量初始值设置

5.5 仿真设置

由于我们要做瞬态分析，dc 分析，ac 分析，所以我们一次弄好所有设置。

①Analyses->choose, 或者点击右侧工具栏的 choose analyses 按钮。弹出如图 5.11 所示窗口。于 Analysis 栏选择不同的仿真。对于瞬态分析，我们选 tran，然后于 stop time 栏输入仿真时间 10n。

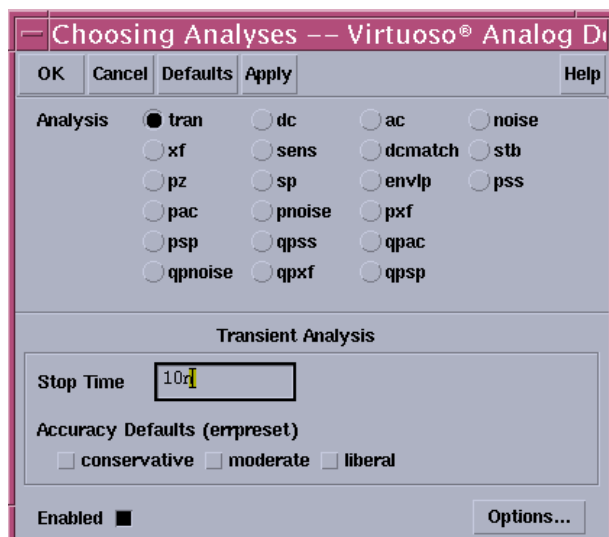


图 5.11 仿真设置窗口，瞬态分析的设置

②再于该窗口中选取 dc，窗口会有相应变化。于 sweep variable 栏选择 Design Variable，然后输入变量名 vin，并输入变量的扫描范围，从 0 到 1.8。并选择扫描类型为线性，扫描为 0.001。这样仿真时就会对 vin 从 0 到 1.8V 以 0.001 为间隔进行扫描，从而得到输入输出特性。

注意选中 DC Analysis 复选框，这样就可以将 AC 分析时所采用的的直流工作点保存下来，以便我们后面查看电路的直流工作点。如图 5.12 所示。

③再于该窗口中选取 ac，窗口会有相应变化。默认是频率扫描。我们只需要输入频率扫描范围。这里输入 10 到 1000M(注意是大写)。其它均为默认值。如图 5.13 所示。

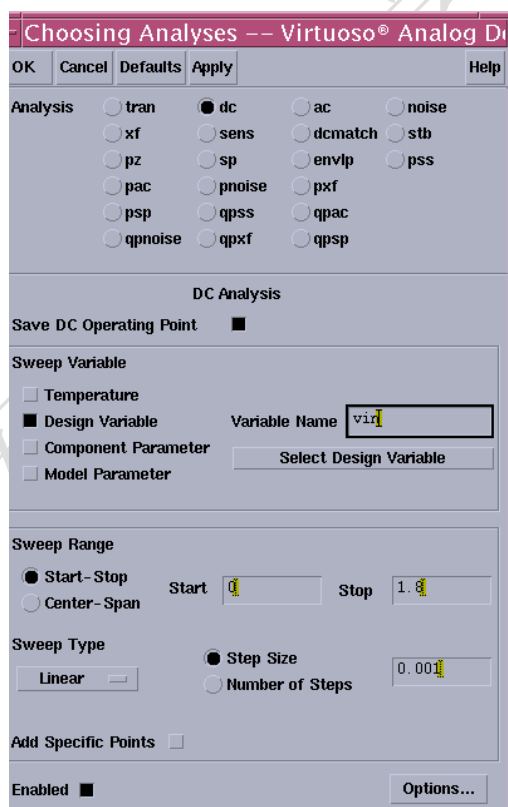


图 5.12 仿真设置窗口，dc 分析的设置

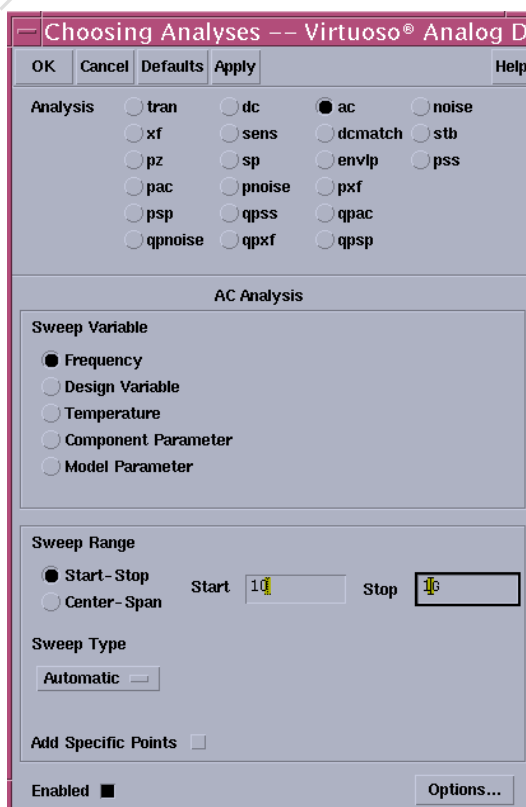


图 5.13 仿真设置窗口，ac 分析的设置

5.6 选择输出，并保存当前仿真设置

接下来要选择我们需要观察的对象，即我们要看哪个节点的电压，或者要看哪一条支路的电流。

于 ADE 窗口，Output->To Be Plotted->Select On Schematic，这样会弹出我们画的电路图。然后分别单击输入和输出两条线 IN 和 OUT，以及第一级反相器的输出 v1。注意选择 v1 时要先选中反相器，再按 e 进入下一层子电路。并注意一定要单击导线，而不是元件的 Pin 角！然后按 ESC 退出选择状态。此时在仿真窗口中已经有了 IN、OUT、v1 三项，如图 5.13 所示。

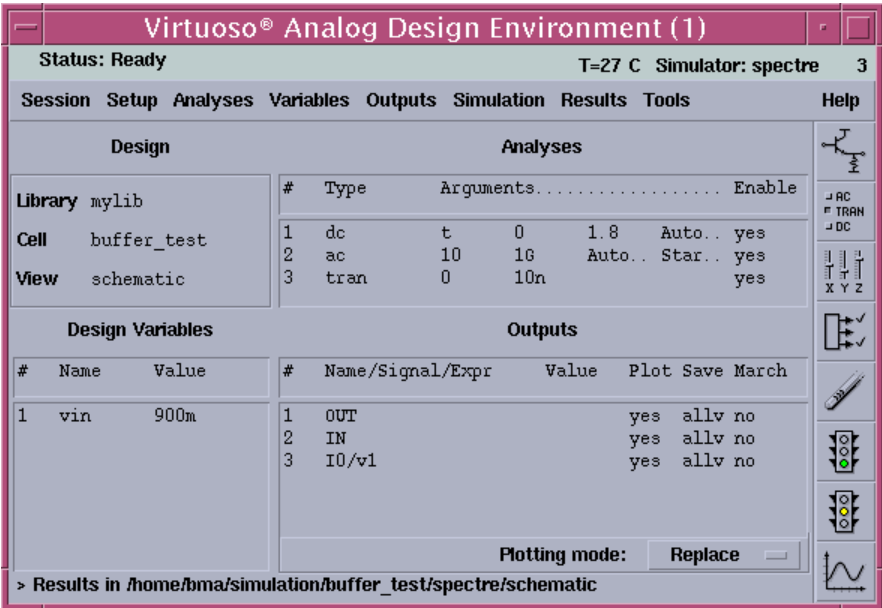


图 5.13 全部设置完成后的仿真环境窗口

保存当前的仿真设置。Session->Save State，弹出保存对话框，填好名称，点击 OK 确定。这样下次再仿真时，可以直接调用该仿真设置，而不用每次都进行同样的设置。

5.7 进行仿真并查看波形

现在可以进行仿真了。于 ADE 窗口，Simulation->Run，或者点工具栏 Netlist and run 按钮，再点 OK，就可以显示出前边所设置的 output 中指定的信号波形。如图 5.14 所示。

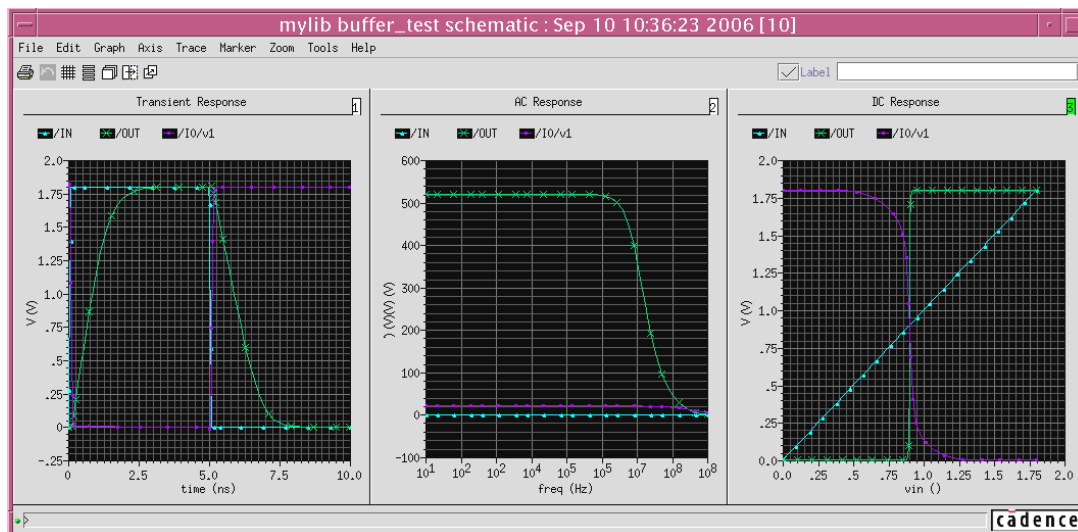







图 5.14 波形显示窗口

默认是将所有仿真波形显示在同一个大窗口中，在上图中，从左到右分别是瞬态分析波形窗口、AC 分析波形窗口、DC 分析波形窗口。


图中每一个小窗口称为子窗口。子窗口右上角的数字图标若为绿色，表示当前子窗口处于被选中状态。对于选中的窗口，我们可以通过工具栏的按钮改变其显示方式，如将一个子窗口中的多条曲线分行显示等。工具栏各按钮作用如下：

-  将选中子窗口中的栅格打开或隐藏
-  将选中子窗口中的曲线分行显示。
-  将子窗口重叠排列，这样当前波形窗口内每次就只显示一个子窗口。
-  新建子窗口
-  新建波形窗口

我们也可以对波形进行操作。单击窗口中的曲线或者子窗口上方的信号名称，可以选中相应的波形。对于选中的波形，可以对其进行诸如删除、各种数学计算等操作，还可以对波形做任意的拖动操作，详细功能将在后面介绍。

5.8 仿真结果分析

通过对仿真结果的分析，要学会波形测量、查看电路工作点状态、查看电路节点电压、查找器件参数、计算器的使用以及对波形窗口坐标等各种操作。

①瞬态结果分析。选中瞬态分析子窗口，点击工具栏按钮 ，将瞬态波形放大到当前波形窗口中。为了看清缓冲器延时，需要放大波形的特定区域。在波形窗口中按住鼠标右键不放，拖出一个矩形框，框住瞬态波形中输出信号的上升部分，松开右键后，矩形框中的波形就会被放大到整个窗口。如图 5.15 所示。

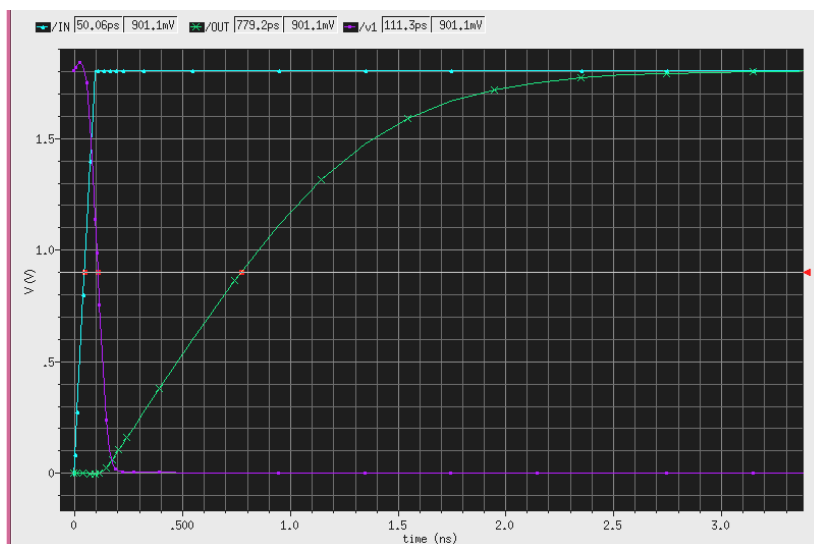




图 5.15 瞬态分析波形中的上升下降延时测量

于波形窗口点击菜单栏 **Trace**，然后将 **Horiz Cursor** 的对勾打上，此时会在当前窗口出现一条水平白线。拖动白线右边的红色头，可以上下移动白线，同时窗口上方会显示出水平白线与图中三条曲线交点的坐标值。我们拖动白线到纵坐标为 0.9V 的位置，然后就可以读出信号 **VIN**、**v1**、**VOUT** 对应的时间。比如这里分别为 50ps、111ps、779ps，则第一级反相器的输出延时为 61ps，总的缓冲器上升下降延时为 729ps。同理我们也可以测出下降上升延时 t_{PLH} 。

②AC 结果分析。点击波形窗口右上角图标  中的数字，可以使相应窗口弹到最前面。点击数字 2，则 AC 波形就到最前面。从图中我们可以大致读出缓冲器作放大器时小信号增益约为 500，单位增益带宽接近 1G。为了更精确的测量单位增益带宽以及 3dB 带宽，我们还要利用计算器得到输出信号的波特图及其相位频率响应。

由于带宽比较大，我们把 AC 分析的范围改大到 10G，重新仿真一次。然后于波形窗口点击按钮  新建一个子窗口，再于菜单栏，**Tools->Calculator**，打开计算器，如图 5.16 所示。

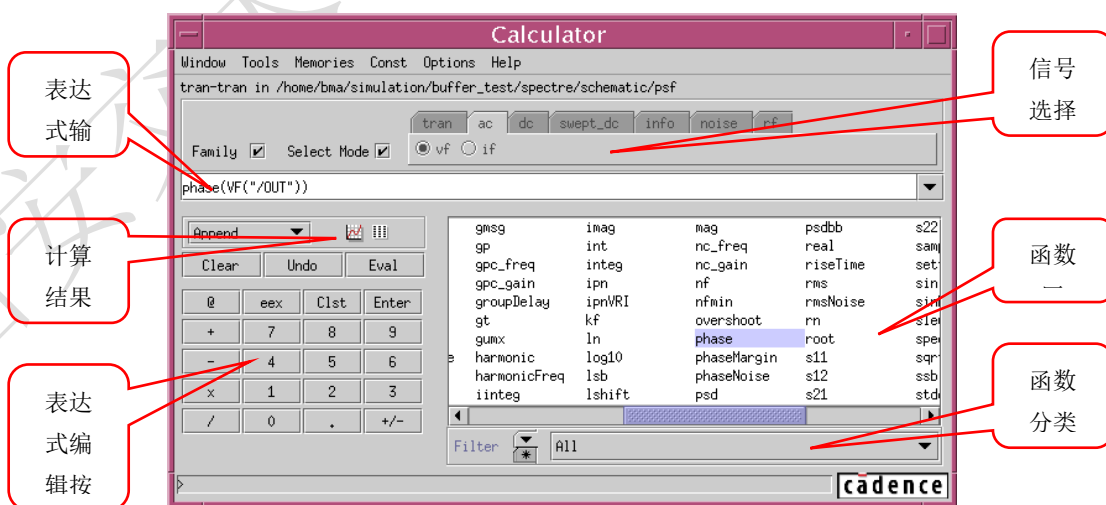



图 5.16 计算器

先点击 **Clear** 清除当前选中的信号。然后于信号选择区选中 **ac** 中的 **vf**，这时会弹出原

理图，从原理图中点击 OUT 线，此时会在表达式输入区出现 VF(“/OUT”)，然后再于函数区单击 dB20，输入区变为 dB20(VF(“/OUT”))。再点击 plot 按钮，就会在当前选中的波形窗口中显示出输出信号的波特图。此时波形窗口如图 5.17 所示。

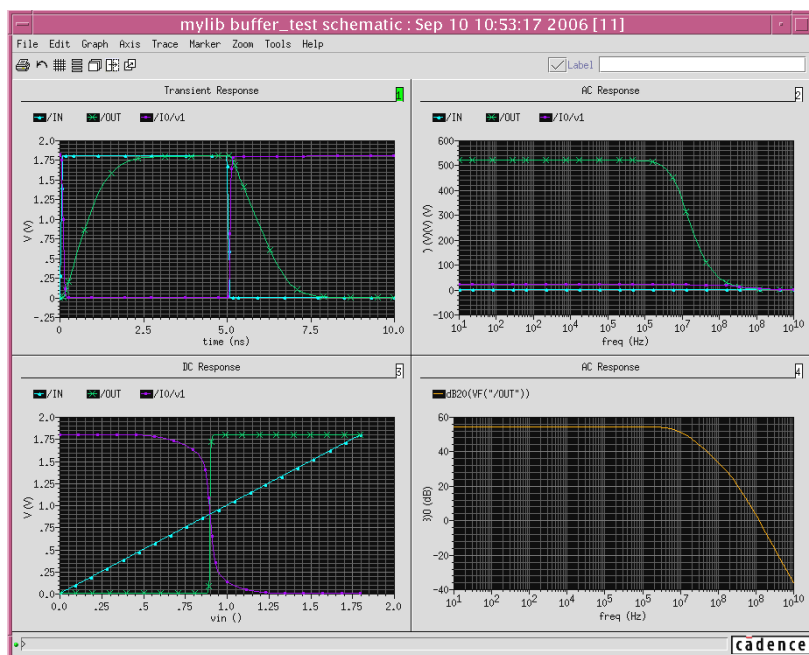


图 5.17 新添加子窗口

同样的方法再添加 $\text{phase}(\text{VF}("/\text{Vout}"))$ ，然后将该子窗口放大到波形窗口中，并点击分行显示按钮，如图 5.18 所示。利用 Trace 工具，我们可以精确地量出小信号单位增益带宽为 1.2GHz。

另外，双击窗口坐标，弹出一个坐标属性窗口，可以调整坐标和栅格的范围、刻度等。例如对于相位频率响应的坐标，我们可以通过调整让其显示出 90° 、 180° 等我们希望的刻度。这里我们不再详述，同学们自己试验。

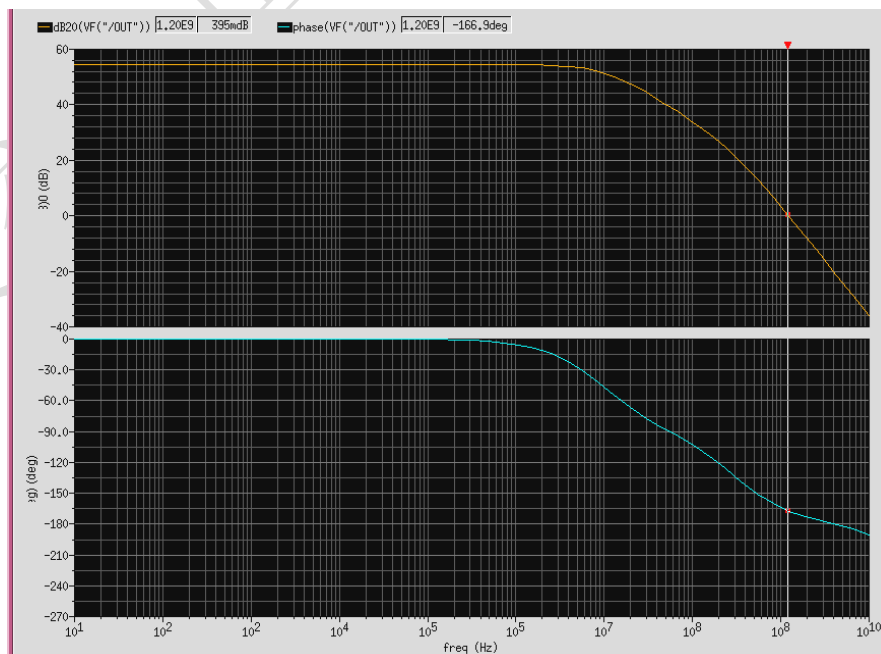


图 5.18 输出信号波特图与相位频率响应

③DC 分析结果。将 DC 波形放大到当前窗口中，放大 0.9V 附近的波形。可以从中测出开关阈值电压 V_M 。如图 5.19 所示。

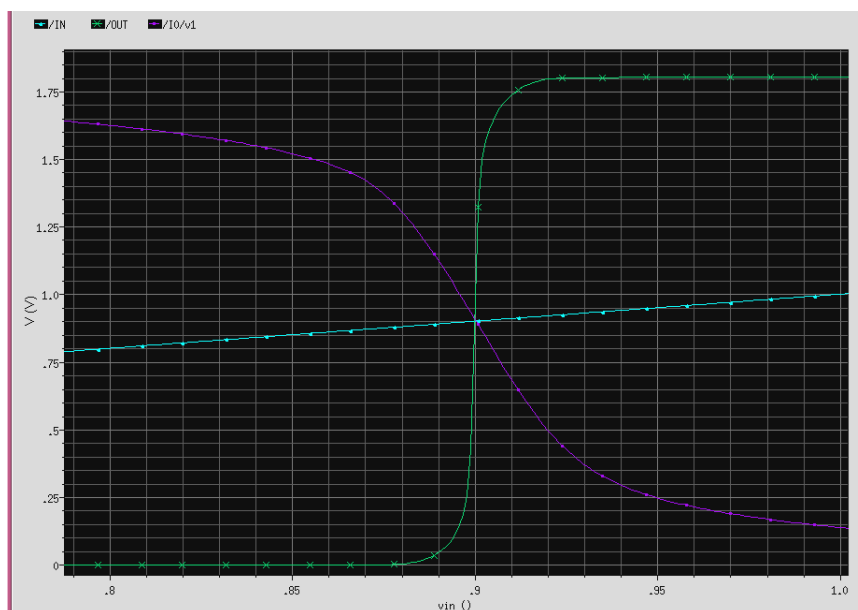


图 5.19 缓冲器的输入输出特性

从 DC 分析结果中也可以看出，缓冲器比单级反相器具有更大的噪声容限。

此外，我们还可以得到 AC 分析中电路的工作点状态。于 ADE 窗口点击 Results->Print，下拉菜单中有 DC Node Voltages, DC operating Points, Model Parameters, Transient Node Voltages, Transient Operating Points 等选项，可以分别打印出电路的节点电压、工作点状态、模型参数、瞬态节点电压、瞬态工作点状态，如图 5.20 所示。

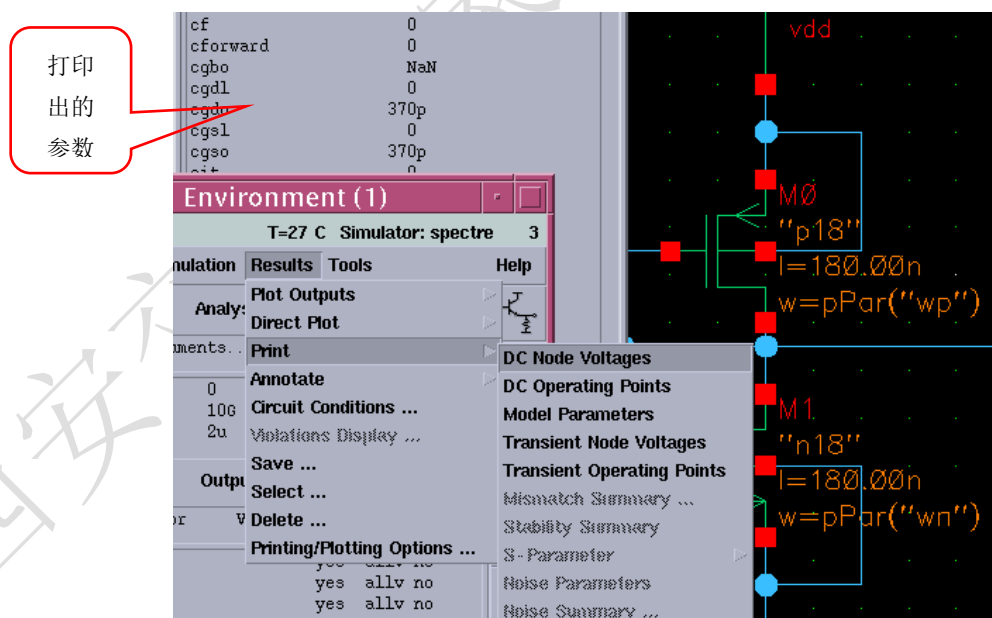


图 5.20 打印节点电压、工作点状态以及器件参数

同学们可以测试各种宽长比下电路的瞬态、AC、DC 波形，分析第一级反相器和第二级反相器分别对电路性能有什么影响，并思考为什么。另外考虑该缓冲器既然有放大作用，能否用来作放大器，为什么。

对于电路仿真我们就只进行一个简单的介绍，更多的功能将在后续实验中介绍。

西安交大国家IC人才培养基地

六、版图设计——Virtuoso Layout Editing

现在开始根据我们前边的设计尺寸，来画缓冲器的版图。为了显示层次化设计，我们先分别画出两个不同尺寸的反相器版图，再将两个反相器组成缓冲器的版图。

通常的版图设计步骤大概如下：①NMOS：画出 N 有源区->栅->做接触孔->覆盖金属->N 注入区；②PMOS：画出 P 有源区->栅->做接触孔->覆盖金属-> N 注入区；③连接晶体管：放置 PMOS 和 NMOS->连接输出->连接输入->金属连接->电源线；④PAD。

由于反相器较为简单，我们直接开始画它的版图。

6.1 新建 layout

类似新建原理图，CIW 窗口菜单栏->File->New->CellView，选择自己的库，然后输入版图名字 inverter，注意于 tool 栏选择 Virtuoso，ViewName 栏会自动变为 layout。点击 OK 后，会弹出 Virtuoso 主界面，以及 LSW 窗口。如图 6.1 所示。

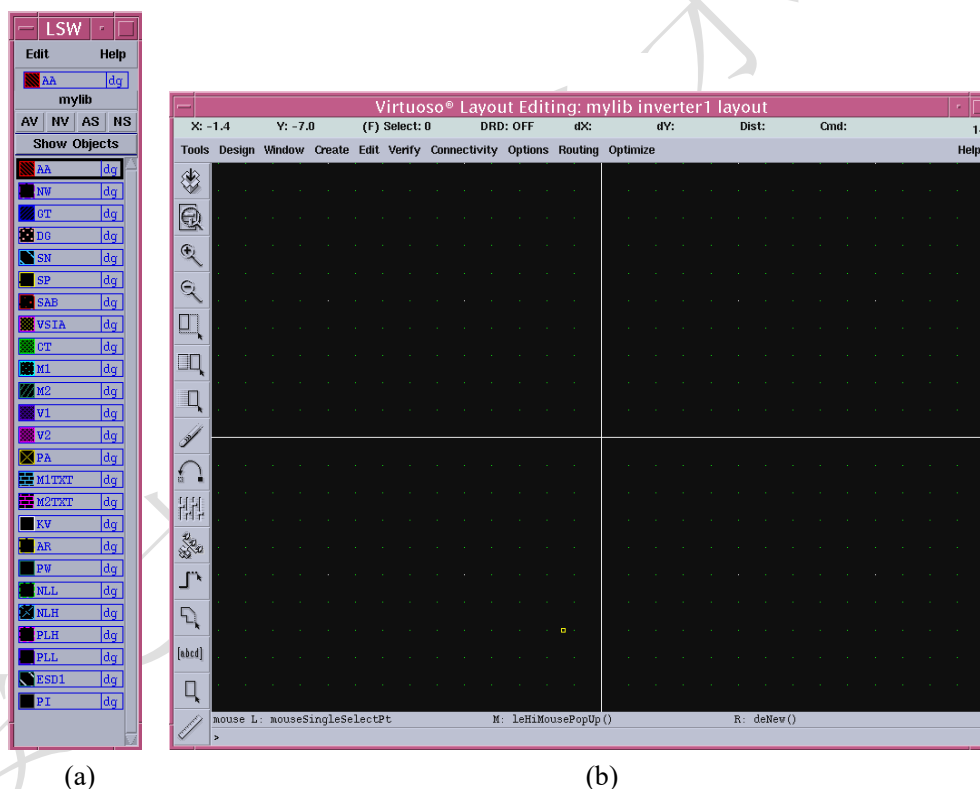


图 6.1 (a)LSW 窗口 (b)版图编辑主界面 Virtuoso

图 6.1(a)中的窗口是用来选择掩膜层的，我们可以在设计时随时改变层，在 LSW 中选相应层，那么在 Virtuoso 中就使用此层画图。(b)就是我们画版图的主窗口，结构类似 Composer。

Virtuoso 主界面包括：标题栏，菜单栏，工具栏，状态栏(第二行)，提示区(就是最底下那行)以及最大的那个工作区。状态栏会提示当前的命令以及所选择的物体个数，提示区会告诉你当前应该做什么事。

注意：①类似 Composer，Virtuoso 中的多数命令会一直保持，直到你调用其它命令替代

它或者按 Esc 取消，尤其是在执行 delete 命令时，忽视这一点很可能会误删除！

②点击工具栏的 zoomin 和 zoomout 按钮可以放大缩小版图。键入快捷键 f 可以使版图自动缩放到合适大小。

③编辑版图过程中注意要及时保存，保存方法是菜单栏->Design->Save，也可以键入快捷键 S(大写)来保存。

6.2 参阅设计规则手册

版图有非常多的物理规则约束，这是由 Foundry 的工艺决定的。我们所使用的 SMIC 的 0.18um 工艺的设计手册在如下路径：

/cad/smic018_tech/Process_technology/Mixed-Signal/Design_Rule>TD-MM18-DR-2001v6P.pdf

手册中每一条规则都给出了编号，比如 AA.5，SN.01 等。同学们在设计版图的过程中要时刻注意设计规则手册。在后面做设计规则验证(DRC)遇到错误时还要不断的回来查阅该设计手册。

6.3 设置 LSW 窗口可见层

SMIC 的工艺库默认有非常多的掩膜层。我们画反相器只需要其中一小部分层，为了方便我们操作，我们需要对 LSW 进行设置。LSW 窗口->Edit->Set Valid Layers，会弹出如图 6.2 所示窗口。表 6-1 给出了 smic 工艺中的关键层，并标出了我们在画反相器时所需要的层。请根据表 6-1 设置可见层。点击 OK 确定后，LSW 窗口->Edit->Save，弹出保存对话框，点击 OK 保存当前设置，这样下次再进入 Virtuoso 时就会使用该设置。



表 6-1 SMIC 关键掩膜层

序号	层名字	含义	含义	本实验所需要的
1	AA	Active Area	有源区	√
2	AR	Reverse Active Area	无源区	
3	KV	Alignment mark clear-out	对准标记	
4	NW	N-Well	N阱	√
5	PW	P-Well	P阱	
6	DG	Dual Gate(thick oxide)	双栅(厚氧)	
7	GT	Poly gate	多晶硅	√
8	PLH	PMOS LDD implant for 3.3V	3.3V下PMOS轻掺杂注入	
9	NLH	NMOS LDD implant for 3.3V	3.3V下NMOS轻掺杂注入	
10	PLL	PMOS LDD implant for 1.8V	1.8V下PMOS轻掺杂注入	
11	NLL	NMOS LDD implant for 1.8V	1.8V下NMOS轻掺杂注入	
12	SP	P+implant	P注入	√
13	SN	N+ implant	N注入	√
14	ESD1	ESD implant for Boron(B)	静电放电注入	
15	SAB	Salicide block area	硅化物阻挡区	
16	CT	Contact	有效接触孔	√
17	M1	Metal-1	第一层金属	√
18	V1	VIA-1	通孔1	√
19	M2	Metal-2	第二层金属	√
20	V2	VIA-2	通孔2	
21	M3	Metal-3	第三层金属	
22	V3	VIA-3	通孔3	
23	M4	Metal-4	第四层金属	
24	V4	VIA-4	通孔4	
25	M5	Metal-5	第五层金属	
26	V5	VIA-5	通孔5	
27	M6	Metal-6	第六层金属	
28	PA	Passivation/Pad	焊盘	
29	PI	Polyimide		

注意除了表 6.1 中我们标出的需要显示的图层外，还需要 M1TXT 和 M2TXT 两层，用来标记输入输出。

6.3 画第一个反相器

6.3.1 画第一个晶体管——NMOS

一个标准 CMOS 反相器有两个晶体管，现在我们来画第一个晶体管 NMOS，其栅长为 0.18 μm ，栅宽为 0.6 μm 。

①由于我们采用的是 $0.18\mu\text{m}$ 工艺，而在 Virtuoso 中默认最小单位为 $0.1\mu\text{m}$ ，因此我们需要改变该设置。Virtuoso 菜单栏->Options->Display，会弹出显示选项窗口，将其中的 X snap Spacing 和 Y snap Spacing 两栏设为 0.01，点击 Save to 保存，然后点击 OK 确定。

②画有源区。先选择标尺工具(Window->Create Ruler 或者直接键入快捷键 k)进行定位。点击左键开始，再点一次左键结束画标尺。用标尺定位一个 $1.42\mu\text{m} \times 0.6\mu\text{m}$ 的矩形。点击 ESC 退出标尺命令。然后于 LSW 窗口中选择 AA 层。再于 Virtuoso 窗口->Create->Rectangle 或者直接键入快捷键 r，沿刚才的标尺，画一个 $1.42\mu\text{m} \times 0.6\mu\text{m}$ 的矩形有源区。如图 6.3 所示。

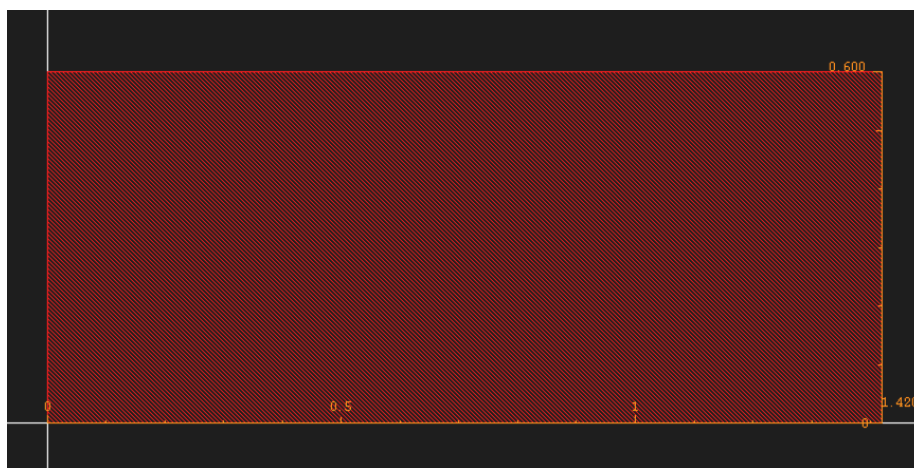


图 6.3 用标尺定位有源区

③画多晶硅栅。键入快捷键大写的 K 清除当前所有标尺。然后利用标尺，于有源区中间定位好栅的位置。再于 LSW 窗口中选择 GT 层，然后再在 Virtuoso 中沿标尺画栅。如图 6.4 所示，注意由设计规则 GT.5 知多晶硅栅相对有源区的最小延伸为 $0.22\mu\text{m}$ 。

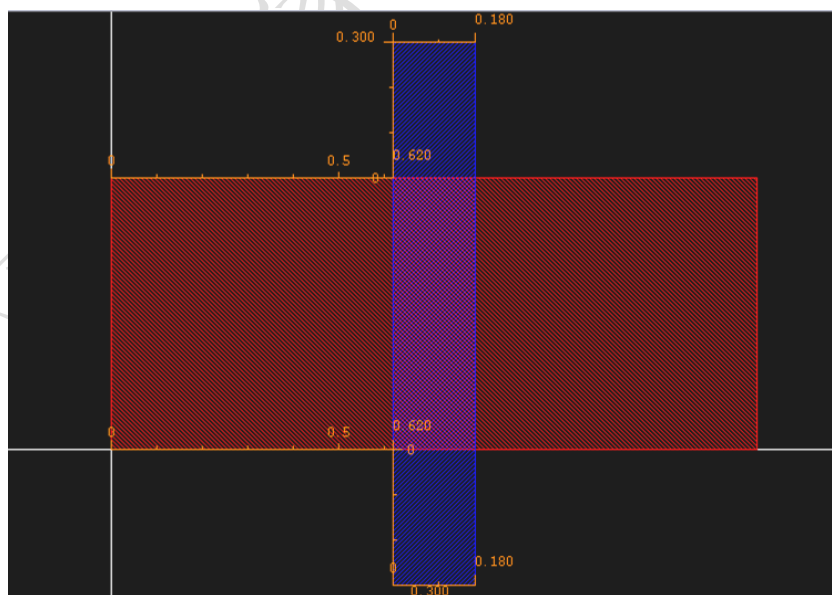


图 6.4 画栅

④画接触孔。同上面的方法，于 MOS 管的漏区和源区各画一个有效接触。有效接触孔的图层名为 CT。注意，画好一个有效接触后，另一个可以直接复制过去。方法为：键入快捷键 c，然后选中要复制的对象，再移动对象到合适位置，点击左键放下。在放下对象之前

按 F3 可以进一步选择细节, 比如复制的物体以何种方式移动(水平、垂直、45 度角等)。画好 CT 后如图 6.5 所示。注意由设计规则 CT.1 知接触孔的大小必须为 $0.22\mu\text{m}$ 。

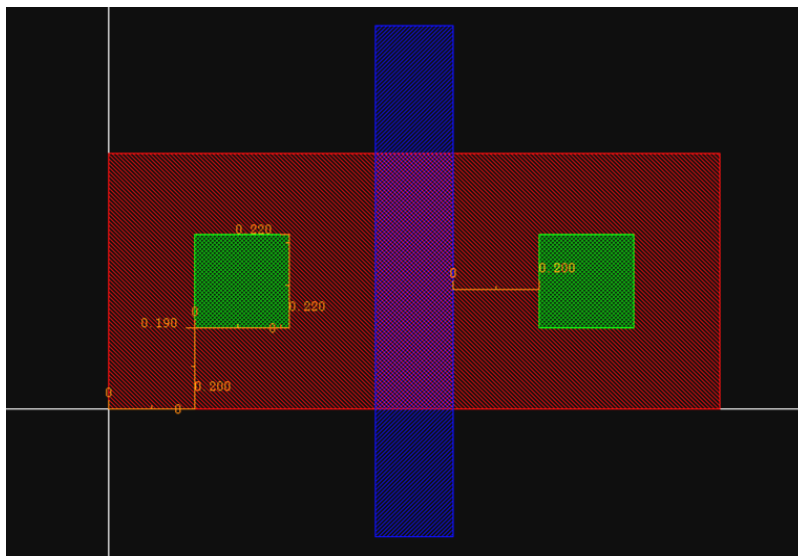


图 6.5 画有效接触

⑤画金属层。有效接触只是在氧化层上打了个孔, 实际的连接得靠金属层。在 LSW 中选中 M1 层, 然后以 CT 为基准外延 $0.16\mu\text{m}$ 画矩形。另一个 M1 矩形可以直接复制过去。

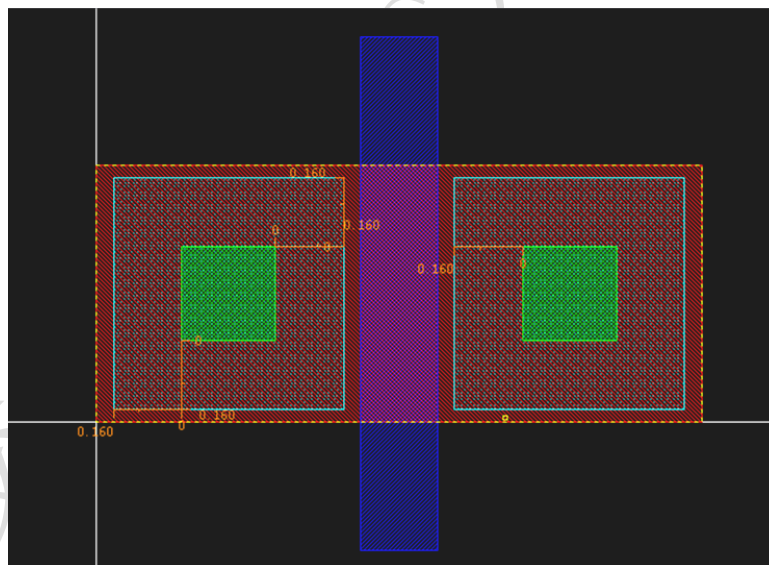


图 6.6 用金属包围有效接触

⑥画扩散区。有源区仅仅是把芯片分为有源区和场氧区两部分。具体的有源区究竟是用来做 PMOS 还是做 NMOS 就是由扩散区类型来决定的。在 LSW 中选中 SN, 然后以有源区为基准, 水平方向外延 $0.2\mu\text{m}$, 垂直方向外延 $0.35\mu\text{m}$ 画矩形。

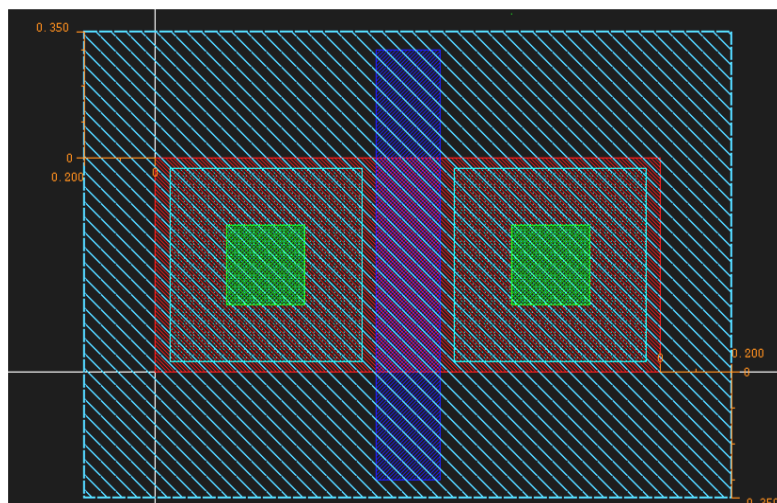


图 6.7 画 N 扩散区

至此，第一个晶体管 NMOS 我们就画好了。

6.3.2 画第二个晶体管——PMOS

现在我们画第二个晶体管——PMOS。根据前边设计要求，L 为 $0.18\mu\text{m}$ ，W 为 $1.95\mu\text{m}$ 。我们在 NMOS 旁边的空白处开始画 PMOS。前几步和 NMOS 差不多，只是在画接触孔时，由于 P 管较长，因此需要打多个孔，这样才能使得电流均匀地流过沟道，注意孔间距要大于 $0.25\mu\text{m}$ 。注意要用金属层 M1 分别包住源区和漏区各自的三个接触孔。

此外，我们的工艺是 P 衬底，对于 PMOS 我们还需要画 N 阱。以 PMOS 的有源区 AA 外延 $0.5\mu\text{m}$ 画 N 阱(图层名为 NW)。画好的 PMOS 见图 6.8。

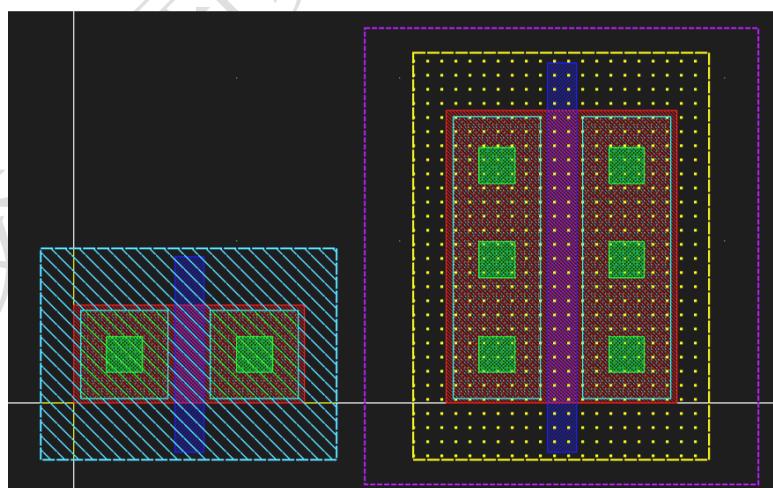


图 6.8 NMOS 与 PMOS

6.3.4 信号线的连接

现在我们要把 NMOS 和 PMOS 连接起来，使其组成 CMOS 反相器。

①把 PMOS 移动到合适位置。用鼠标拖一个框，选中整个 PMOS，键入快捷键 m，然后点击 PMOS，按 F3 改变 snam mode 到 anyangle，点击 hide，这时移动 PMOS 到 NMOS 上方，使 P 管和 N 管的栅在同一直线上。点击左键放下。如图 6.9 所示。

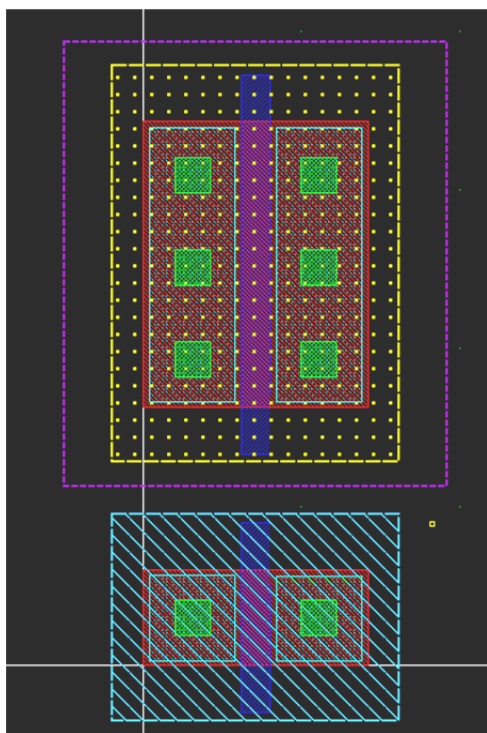


图 6.9 移动 PMOS 到合适位置

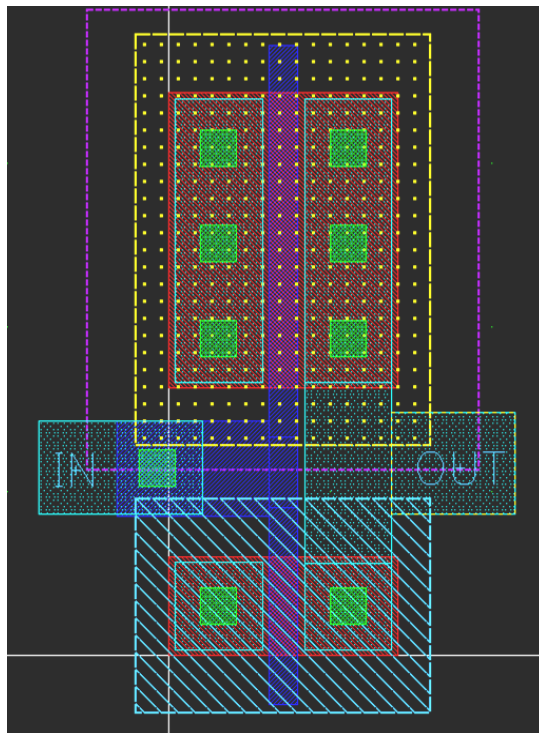


图 6.10 连接输入输出

②连接输出。在 NMOS 和 PMOS 的漏极之间画一个金属矩形，把 NMOS 和 PMOS 漏区的两块金属连接起来。然后再加上一块水平的金属。

③给输出加上标记。于 LSW 窗口选择 M1TXT 层，利用 Create->Label(或者直接键入快捷键 l)，于 Label 栏输入 Vout，把 Height 改变为 0.2，点击 Hide，将标记放在水平的金属上。

④连接输入。在 NMOS 和 PMOS 多晶硅栅之间再画一个 GT 层矩形，把 NMOS 和 PMOS 的栅极连接起来。然后再加上一块水平的 GT。栅必须还得和金属层连起来才能够连到外部输入。用 CT 连接一层金属到刚才的水平 GT 上。

⑤给输入加上标记。同上面的方法，给输入金属加上标记 Vin。

到这一步的版图如图 6.10 所示。

6.3.4 连接电源与地

现在来画电源线和地线。电源线和地线使用金属 2。

①画电源线。于 LSW 窗口选择 M2，然后在 PMOS 上方画水平矩形。

②连接电源线与 PMOS 源区。在 PMOS 源区上方用 M1 画矩形，使其一端连接 PMOS 源区金属，另一端和刚才画的电源线交叠。现在要把这两层交叠的金属连起来，金属 1 与金属 2 的连接是利用通孔 1(图层名为 V1)。于 LSW 窗口选择 V1，在 M1 与 M2 交叠的地方画一矩形。由设计规则 V1.1 知通孔边长等于 0.26um。

③给电源线加上标记。于 LSW 窗口选择 M2TXT，然后给电源线加上标记 vdd!。注意加上叹号的意思是该节点为全局节点。

④同样的方法用 M2 画出地线，并用 V1 将其和 NMOS 源区金属相连，然后用 M2TXT

加上标记 gnd!。

到这一步的版图如图 6.11 所示。

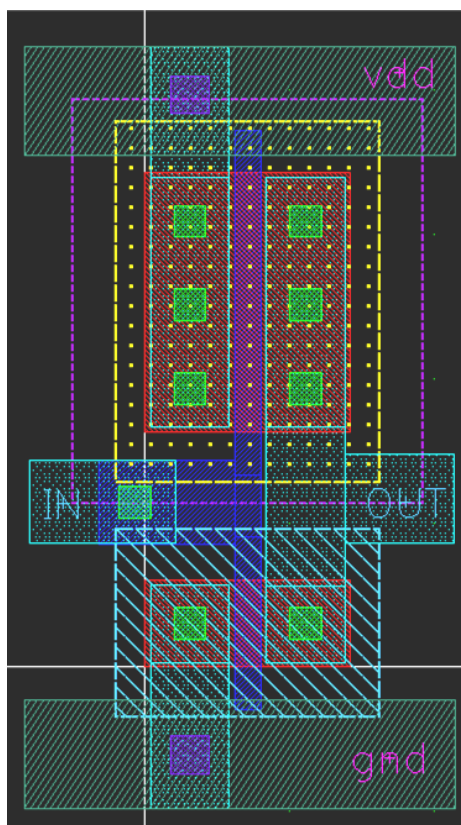


图 6.11 加上电源线与地线

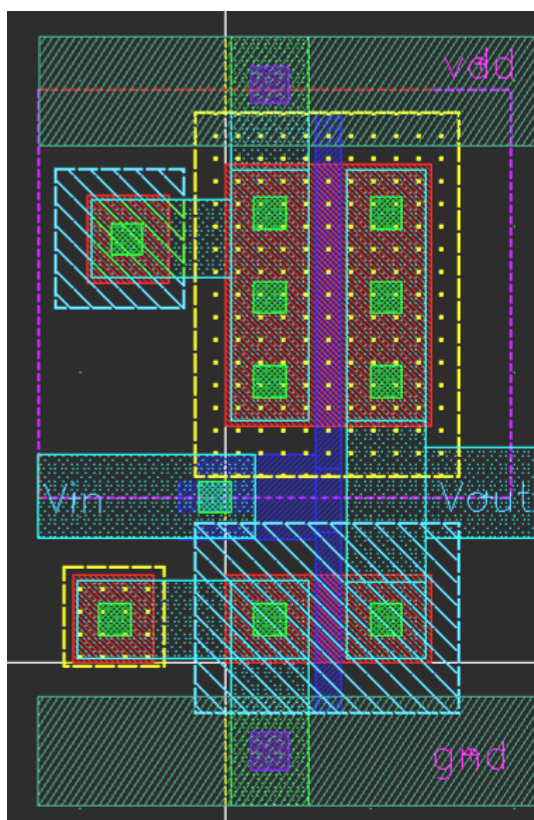


图 6.12 加上衬底接触

6.3.5 做衬底接触

晶体管所在的衬底要有正确的偏置，NMOS 的衬底必须接地，而 PMOS 的衬底必须接电路中的最高电位，即电源。给衬底加偏置的方法是作衬底接触。在 P 衬底工艺下，对于 NMOS，就直接在 P 衬底上加 P 注入形成欧姆接触，并与地相连，对于 PMOS，是在 N 阱上加 N 注入形成欧姆接触，并与电源相连。

①做 P 衬底接触。在 NMOS 旁边画 AA 矩形，然后用 SP 将其包围，再用接触孔 CT 和金属 M1 将该衬底接触与地相连。注意留够 SP 与 AA 间距。

②做 N 阱接触。注意先将 N 阱扩大，以便能放下欧姆接触。Edit->Stretch 或者键入快捷键 s，然后将鼠标移动到 N 阱的边线上，点击左键，这时移动鼠标就可改变 N 阱大小，再次点击左键确定。然后类似上面的方法做 N 阱接触，只是将 SP 换成 SN，然后再用 CT 和 M1 将接触与电源相连。注意由设计规则 SN.11，在靠近 N 阱边缘的地方，SN 与 AA 的间距要大于 0.18um。

图 6.12 即为第一个反相器最后画完的版图。注意这还不是最终的版图，只有通过了 DRC 和 LVS 的版图才能算是最终的版图。

6.4 画第二个反相器

同样的方法，我们新建一个 cellview，画第二个反相器的版图。注意根据前边的设计要求，第二个反相器的 NMOS 栅宽为 $1.8\mu\text{m}$ ，PMOS 的栅宽为 $5.85\mu\text{m}$ 。为了使两个反相器在最后版图拼接时易于连线，同时也为了使 MOS 管的栅不要过长，我们利用叉指结构来做。图 6.13 显示了如何用叉指结构来形成栅长为 $5.85\mu\text{m}$ 的 PMOS，清楚其间，我们没有画出 P 扩散区。

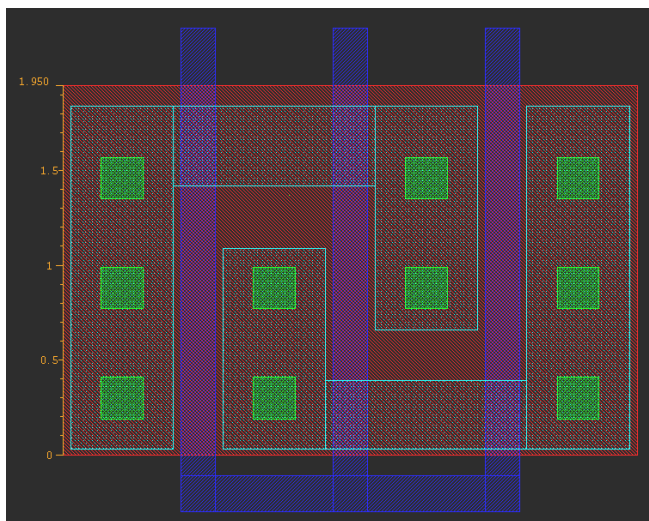


图 6.13 叉指结构

实际上，叉指结构就相当于几个 MOS 管并联。图 6.13 中就相当于 3 个 PMOS 管的 G、S、D 分别相连。完整的第二个反相器的版图如图 6.14 所示。

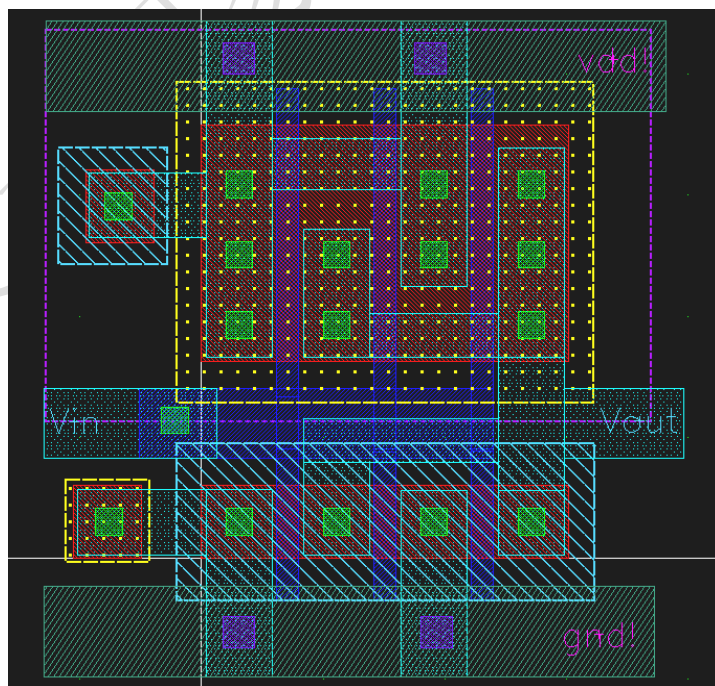


图 6.14 第二个反相器版图

6.5 画缓冲器版图

现在用前边画的两个不同尺寸的反相器来组成缓冲器。新建一个 layout 的 cellview，名字取为 buffer。

在 Virtuoso 界面中，键入快捷键 i 添加器件，点击 browse，选择自己库中的反相器的 layout，就会出现一个大的方框，代表 inverter 的版图，单击鼠标放在合适位置。同理再放置另一个反相器的版图。如图 6.15 所示。按 Shift+f 可以显示出下一级器件的版图细节。按 Ctrl+f 可以隐藏下一级器件的版图细节。

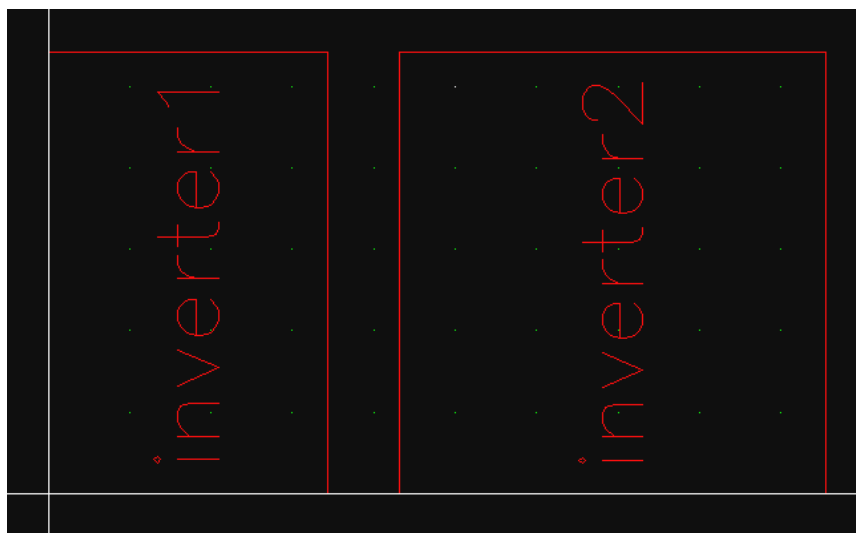


图 6.15 用反相器组成缓冲器

按 Shift+f 显示出版图的内部结构，然后分别用 M1 和 M2 将两个反相器的输入输出以及电源和地连起来，然后用 M1 画输入与输出线。最后用 M1TXT 层给输入与输出加上标记 IN 和 OUT，用 M2TXT 层给电源与地加上标记 vdd! 和 gnd!。最终完整的缓冲器版图如图 6.16 所示。

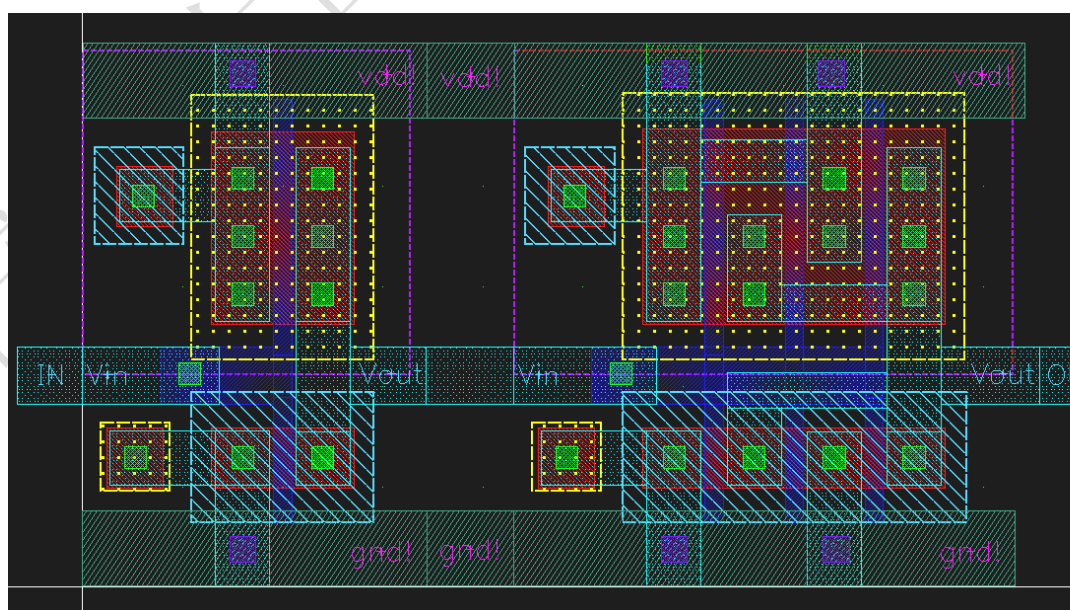


图 6.16 完整的缓冲器版图

西安交大国家IC人才培养基地

七、设计规则检查(DRC)——Assura DRC

Cadence 用来做版图验证的工具具有 Diva, Dracula, Assura。Diva 和 Dracula 已经处于淘汰的地位。因此这里我们简单介绍如何用 Assura 进行 DRC、LVS 以及寄生参数提取。

Assura 具有完全的图形界面，并整合到 Virtuoso 的主界面中，是性能全面的版图验证工具，支持交互式 and 批处理操作，使用层次化的处理，能够快速、高效地识别和改正设计规则错误。

7.1 运行 Assura DRC

①首先我们在工作目录下新建一个目录 DRC，用于存放 DRC 输出文件。

②打开上一章中我们画好的缓冲器版图，从 Virtuoso 菜单栏->Assuro->Run DRC 可以打开 Run Assura DRC 界面，如图 7.1 所示。

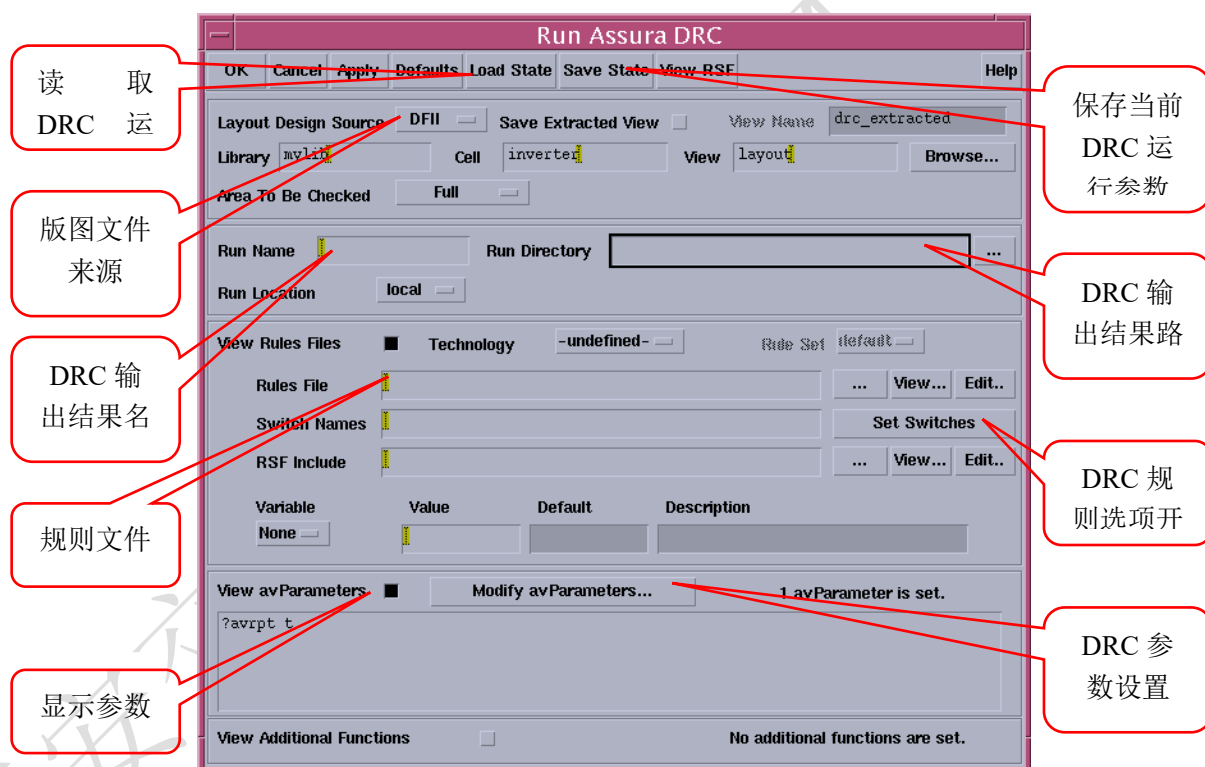


图 7.1 Run Assuro DRC 窗口

其中 Layout Design Source 设置版图文件来源，默认的是 DFII 格式版图，即标准的 Cadence Design Framework II 数据格式。你也可以选择 Stream，即由 GDS 格式的版图来进行 DRC。这里我们使用 DFII。

使用 DFII 格式，则 Library、Cell、View 就会自动填好。于 Run Name 栏输入 DRC 输出结果名字，比如 buf_drc，于 Run Directory 栏输入 DRC 输出结果路径，比如 ./DRC，则 DRC 输出结果将保存在当前工作目录下的 DRC 目录中。

③最重要的是规则文件的设置。我们所使用的 SMIC 的 .drc 规则文件位于如下路径：

/cad/smic018_tech/Design_Service_Technology/mixed-signal/DRC/SmicDRL2TM6P_asr018_mixlog_p2mx_mt.drc

可以直接将以上路径以及文件名输入，也可以点击“...”按钮来选择规则文件。

④由于反相器版图简单，只用到两层金属，而 SMIC 的设计规则默认要检查所有金属层以及多晶硅的覆盖率，简单起见，我们这里关掉覆盖率检查。在设置好规则文件后，就可以点击“Set Switches”按钮，弹出 Set Switches 窗口，选中 NoDensityChecking，点击 OK 确定。

其它选项我们暂时先不管。以上设置完毕，可以点击“Save State”将当前设置保存，然后点击 OK 就开始运行 DRC 了。

7.2 查找 DRC 错误并修改

点击 OK 运行 DRC 后，会先弹出一个 Progress 窗口，显示当前运行 DRC 的一些信息，当 DRC 结束后，该窗口就会消失，然后弹出是否查看当前 DRC 结果的对话框，点击 Yes。

实际上如果严格按照上一章中提到的规则画反相器版图，将不会有任何 DRC 错误。这里给出一个故意加了些设计错误的例子以说明如何查找 DRC 错误，如图 7.2 所示。

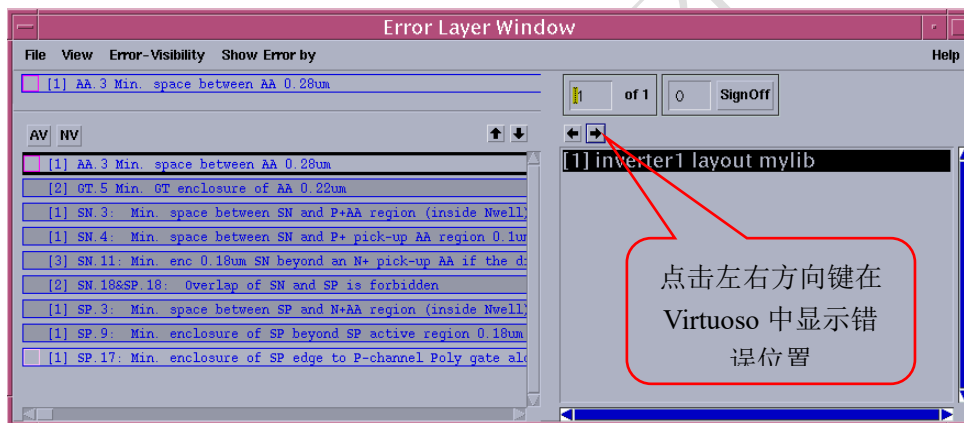


图 7.2 错误提示窗口

图 7.2 中的窗口简称为 ELW 窗口。ELW 窗口左边列出了当前版图中的 DRC 错误类型，每一错误前中括号中的数字是错误个数，后边是错误编号以及解释。比如

[3] SN.11: Min. enc 0.18um SN beyond an N+ pick-up AA if the distance between N+AA and N-Well < 0.43um

意思为编号为 SN.11 的 DRC 错误共有 3 个。该条设计规则的意思是：在距 N 阱边缘不到 0.43um 的情况下，SN 相对 AA 区的最小包围为 0.18um。根据编号我们可以查设计规则手册，以了解具体的错误信息。

ELW 窗口右边列出的是违反对应设计规则的元件。因为我们这里只有一个元件反相器，所以所有的错误对应的只有 inverter 一个元件。选中相应的错误，再点击右方向键，可以在 Virtuoso 窗口中依次显示出对应的错误位置，并用相应的颜色标出。

依照提示修改所有的错误，然后再次进行 DRC，直到没有 DRC 错误为止。注意每次重新进行 DRC 前要对修改的版图进行保存，并关掉当前运行的 DRC(菜单栏 Assura->Close Run)。

7.3 一些高级 DRC 功能

7.3.1 屏蔽器件

某些情况下在做 DRC 时，可能部分器件内部我们已经通过了 DRC，在上一层电路中并不需要对其进行 DRC 了，为了加快速度我们可以屏蔽该器件。比如我们要跳过对缓冲器中第一个反相器的 DRC。

①于 Virtuoso 窗口中，Assura-> Run DRC，弹出 Run Assura DRC 窗口。点击“Modify avParameters”按钮，弹出如图 7.3 所示的 DRC 参数设置窗口。

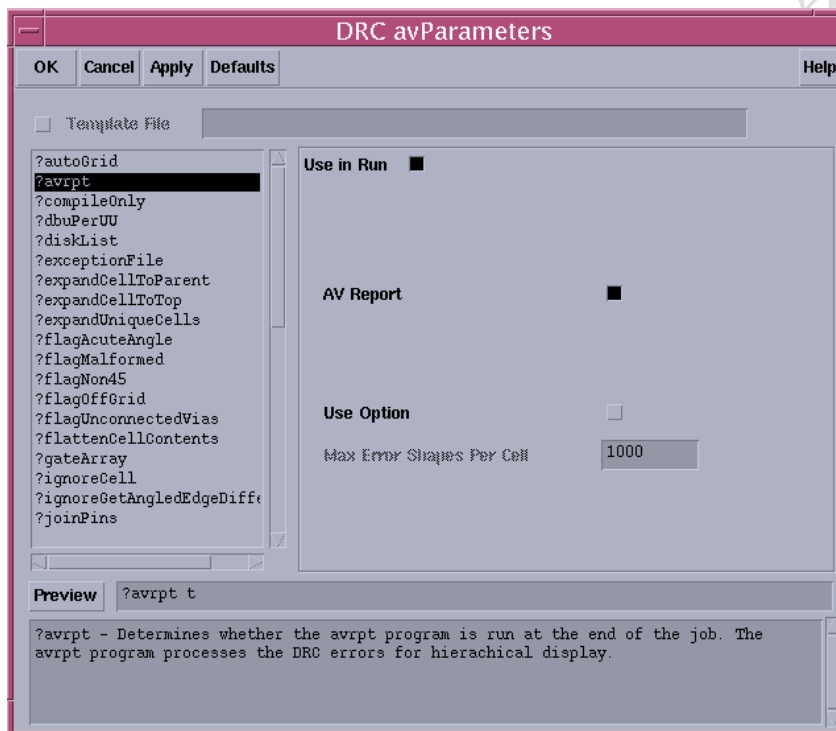


图 7.3 DRC 参数设置窗口

②选中 ?ignoreCell，再选中 Use in Run，然后于 Cells 栏中输入 inverter1，如图 7.4 所示。点击 OK 确定。

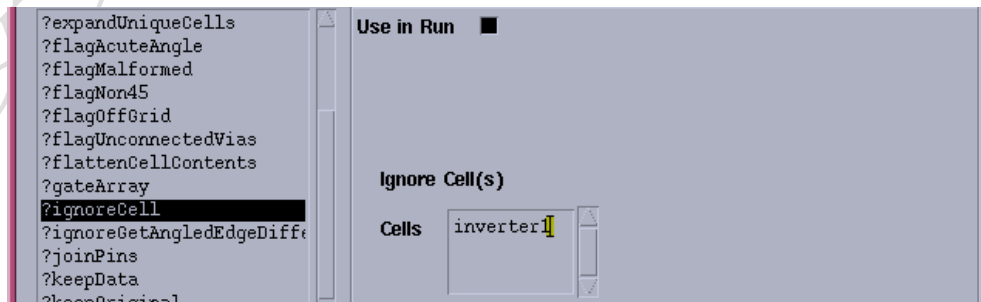


图 7.4 跳过某些器件的 DRC

③此时 Run Assura DRC 窗口底部的 View avParameters 一栏中多出一行，如图 7.5 所示。点击 OK 确定，开始运行 DRC。这样做 DRC 时就不会对 inverter1 的版图进行检查了。



图 7.5 DRC 参数

7.3.2 屏蔽错误

有时候检测出的一些 DRC 错误我们并不想马上更正(或者出于其它原因), 我们可以把这些错误隐藏掉。

①在跑完 DRC 后, 于 ELW 窗口中, File->Preferences, 会弹出 ELW Preferences 窗口, 选中其中的 Add Commands during SingOff 以及 SingOff Feature 然后点击 OK 确定。

②在 ELW 窗口中选中其中一项错误, 并单击右方向按钮, 会在 Virtuoso 中显示出当前的错误位置。

直接单击 SingOff 按钮, 将会只屏蔽一个错误, 在 Virtuoso 窗口中单击要屏蔽的错误位置, 然后会弹出一个注释输入对话框, 如图 7.6 所示。于 Comment 栏输入我们的注释(比如隐藏当前错误的原因), 然后点击 OK 确定。这样当前位置的指定类型错误就不再标出。

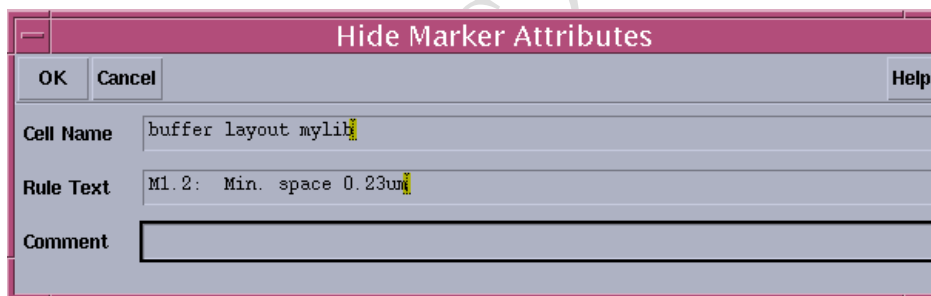


图 7.6 DRC 参数

我们也可以屏蔽选定区域内所有指定类型的错误。于 ELW 窗口中, Error Visibility->Sign Off Area, 会弹出一个对话框, 提示当前图层上被选中的错误标志将被隐藏, 点击 OK 确认, 然后于 Virtuoso 窗口中拖动鼠标画一个矩形, 围住我们想隐藏错误标志的区域, 又弹出一个注释输入对话框, 输入我们的注释, 然后确定。

这样我们选中的区域里指定类型的错误就不再被标注出来了。

于 ELW 窗口中, View->SingOff Errors, 可以查看当前隐藏掉的错误。

7.3.3 把屏蔽掉的错误作为 DRC 运行参数

在上次 DRC 中隐藏掉的错误, 若不作其它设置, 当再跑一次 DRC 时, 这些错误又会出现。如果我们想一直隐藏这些错误, 可以把隐藏掉的错误作为下次 DRC 时的参数。

①在隐藏掉一些错误后, 于 ELW 窗口, File->Save As, 会弹出保存对话框, 将其中的 SingOff Error File 中的文件名改为: buf_waived.evd, 然后点击 OK 保存。

②于 Virtuoso 窗口中, Assura->Close Run, 关掉当前运行的 DRC。

③于 Virtuoso 窗口中, Assura-> Run DRC, 弹出 Run Assura DRC 窗口。点击 “Modify avParameters” 按钮, 弹出如图 7.3 所示的 DRC 参数设置窗口。

④选中?exceptionFile, 再选中 Use in Run, 然后点击 “...” 按钮, 选择前边保存的 buf_waived.evd 文件(在 DRC 目录下)。点击 OK 确定。

⑤点击 OK 开始运行 DRC。

从运行结果中可以看到, 上一次隐藏掉的错误已经被忽略掉了, 不再显示在 ELW 窗口中了。

八、版图和原理图一致性检查—Assura LVS

DRC 完全没有错误以后，就可以进行版图和原理图一致性检测(LVS)了。

8.1 LVS 前的准备

①生成原理图对应的网表文件。于 CIW 窗口，Export->CDL，弹出 CDL Out 窗口，点击“Library Browser”，选择自己的库以及反相器的原理图。然后于 OutPut File 栏输入网表文件名，比如 inv_netlist.sp。再于 Run Directory 栏输入保存目录./LVS。注意一定要将 Scale 项选择为 micron，否则在 LVS 时将会产生参数错误。如图 8.1 所示。点击 OK 将在 LVS 目录下生成相应的网表文件。



图 8.1 生成 CDL 网表文件

②修改网表文件。用任一文本编辑器打开刚才生成的 buf_netlist.sp 文件，把 NMOS 和 PMOS 的模型名分别改为 N 与 P，即将以下两行

```
MM1 Vout Vin gnd! gnd! NM W=wn L=180.00n
```

```
MM0 Vout Vin vdd! vdd! PM W=wp L=180.00n
```

修改为

```
MM1 Vout Vin gnd! gnd! N W=wn L=180.00n
```

```
MM0 Vout Vin vdd! vdd! P W=wp L=180.00n
```

关于为什么要这样改，将在后面说明。

③在自己的工作目录下新建一个目录 LVS，用于存放 LVS 输出文件。

④用 Assura 作 LVS 需要四个文件：

extract.rul：版图提取规则文件

compare.rul：版图原理图比较规则文件

binding.rul：版图和原理图元件名字对应规则文件

smic18_lvs_include.rsf: LVS 选项设置文件

以上四个文件均在 SMIC 的库目录下的 smic18_lvs 文件夹内,我们把整个 smic18_lvs 目录都拷到新建的 LVS 目录中(注意目录名字 smic18_lvs 不能改), 命令如下:

```
cp -r /cad/smic018_tech/Design_Service_Technology/mixed-signal/LVS/smic18_lvs /home/bma/wavehorse/LVS
```

请将以上的/home/bma/wavehorse 换成自己的工作目录。

此外还有一个 assura_tech.lib 文件, 指定了以上 LVS 的库文件路径。把该文件也拷到刚才新建的 LVS 目录下。

```
cp /cad/smic018_tech/Design_Service_Technology/mixed-signal/LVS/assura_tech.lib /home/bma/wavehorse/LVS
```

8.2 设置并运行 Assura LVS

打开上一章中我们画好的缓冲器版图,先要指定 assura_tech.lib 文件的路径。于 Virtuoso 菜单栏, Assuro->Technology, 弹出 Technology Lib Select 窗口, 点击按钮 "...", 选择我们前边拷贝过来的 assura_tech.lib 所在目录, 或者直接输入如下的路径, 点击 OK 确定。

```
/home/bma/wavehorse/LVS/assura_tech.lib
```

于 Virtuoso 菜单栏, Assuro->RunLVS, 可以打开 Run Assura LVS 界面, 如图 8.2 所示。

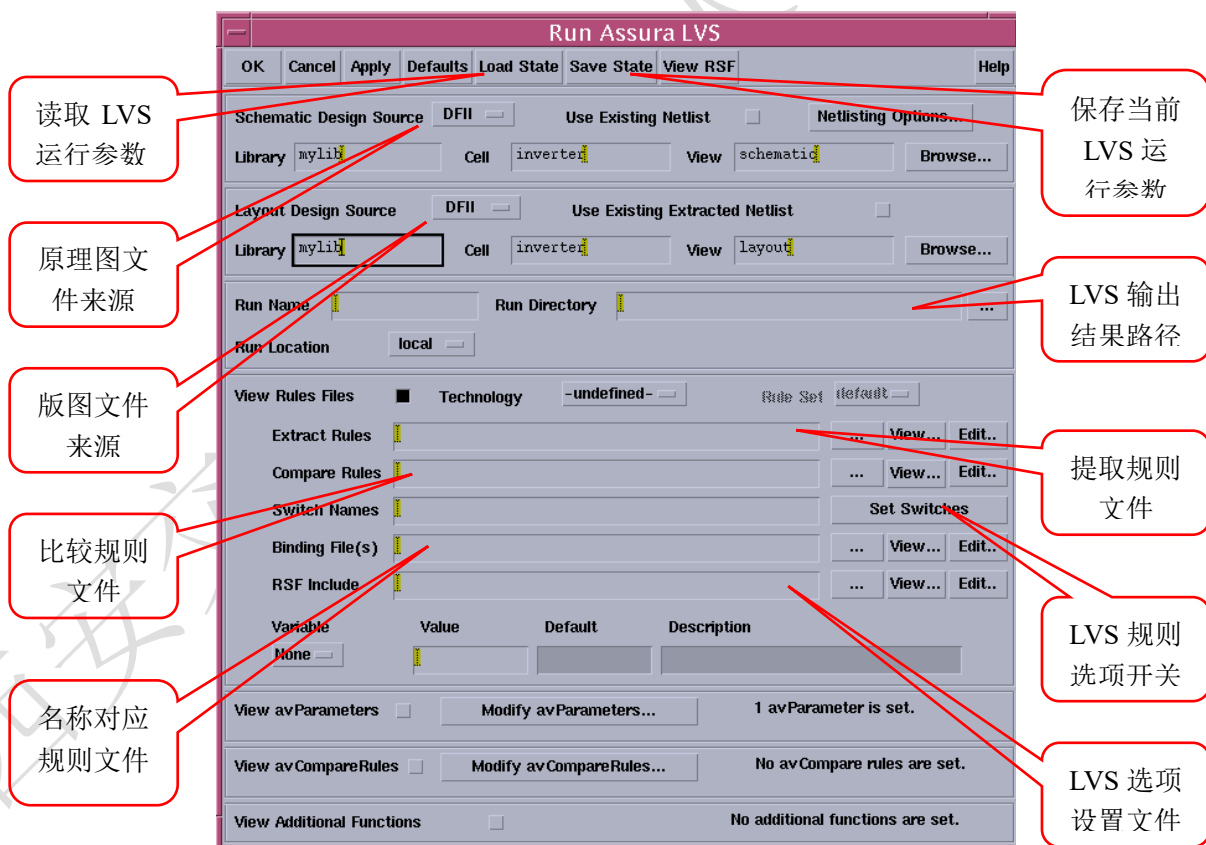


图 8.2 Run Assura LVS 窗口

①Schematic Design Source 设置原理图文件来源,默认的是 DFII 格式版图,即标准的 Cadence Design Framework II 数据格式。这里我们选择 netlist 格式,此时窗口会发生相应变化。点击“Add”按钮,选择我们前边生成的反相器的网表文件,并于 Netlist Type 栏选默认

项 CDL。然后于 cell 栏输入 inverter。

②Layout Design Source 设置版图文件来源，我们仍然使用 DFII，则 Library、Cell、View 就会自动填好。

③于 Run Name 栏输入 LVS 输出结果名字，比如 inv_lvs，于 Run Directory 栏输入 LVS 输出结果路径. /LVS。

④于 Technology 栏选择 smic18_lvs (如果前边没有指定 assura_tech.lib 文件，则不会有这一选择项)，于 Rule Set 栏选择 Smic18_LVS_Bind，则下面四个规则文件会自动填好。如果选择 undefined，则需要手动设置四个规则文件的路径。

点击 Binding File 栏后面的“View”按钮，可以看到 Schematic 和 Layout 元件名的对应关系，其中原理图的 N 和 P 分别对应着版图的 n18 和 p18，由于我们的版图是采用 0.18um 工艺画的，所以 Assura 会把版图上的 NMOS 和 PMOS 提取为 n18 和 p18 的元件，然后再与原理图作比较，所以我们才会在前边需要对原理图的 netlist 文件进行修改。

设置好各个选项后如图 8.3 所示。

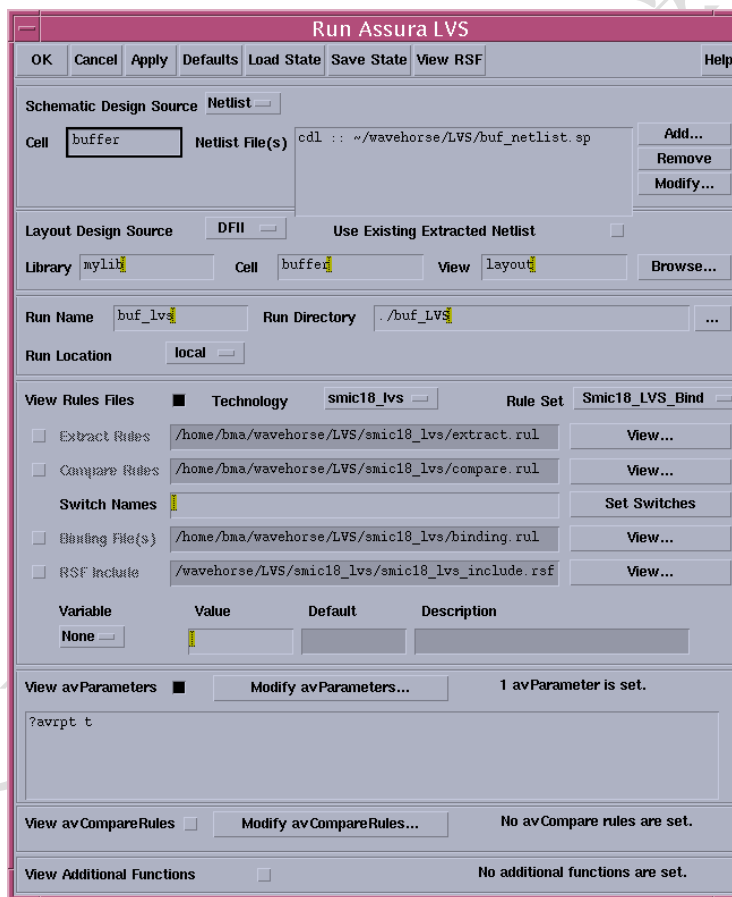


图 8.3 LVS 的设置

其它选项我们暂时先不管。以上设置完毕，可以点击“Save State”将当前设置保存，然后点击 OK 就开始运行 LVS 了。

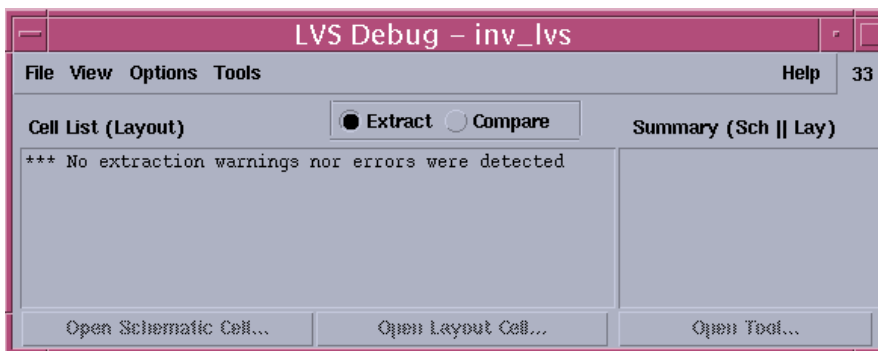
8.3 查找 LVS 错误并修改

点击 OK 运行 LVS 后，会先弹出一个 Progress 窗口，显示当前运行 LVS 的一些信息，当 LVS 结束后，该窗口就会消失，然后弹出是否查看当前 LVS 结果的对话框，点击 Yes。

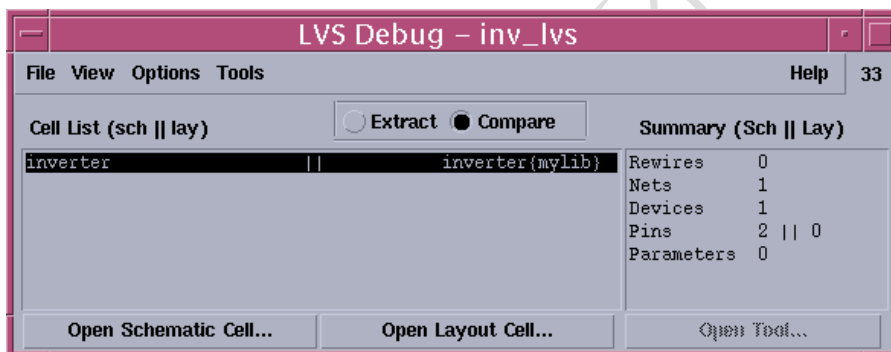
反相器较为简单，如果同学们一路顺利地做下来，到这里可能不会有任何 LVS 错误。这里给出一个故意加了些错误的例子以说明如何查找 LVS 错误，如图 8.4 所示。

LVS 错误分为两类，一类是提取错误，诸如版图上的短路、开路、非法器件等。另一类是原理图与版图差异的错误，诸如线网、器件、端口以及参数等的不匹配。在 LVS Debug 窗口中可以分别查看这两类错误。

图 8.4(a)中显示没有 Extract 错误，图(b)中显示出原理图与版图有 1 个线网、1 个器件、2 个端口不匹配。



(a)



(b)

图 8.4 LVS Debug 窗口 (a)Extract 错误 (b)Compare 错误

在图 8.4(b)的窗口右边，选中不匹配的类型，再点击“Open Tool”按钮，可以分别打开对应的不匹配查找工具。比如选中 Pins，再点“Open Tool”按钮，可以打开 Pins Mismatch Tool 窗口，如图 8.5 所示。



图 8.5 查找不匹配的端口

从图 8.5 中可以看出，原理图上的 Vin 和 Vout 端口，在版图上没有与之对应的端口。点击 Probe 可以定位端口位置。在 Assura Netlist 窗口中，相应的端口会以高亮显示。如果对应的原理图被打开，那么原理图上对应的端口也会被高亮标出。

九、寄生参数提取(PE)——Assura RCX

运行完 LVS 后，就可以进行寄生参数提取了。Assura 中进行寄生参数提取的工具称为 RCX(Parasitic Resistance/Capacitance eXtractor)。注意，运行 RCX 前必须先运行 LVS。

9.1 运行 RCX 前的准备

拷贝 RCX 所需文件到前边运行 LVS 时的库 smic18_lvs 目录下。命令如下

```
cp /cad/smic018_tech/Design_Service_Technology/mixed-signal/ParasiticExtraction/SmicSP7R_018_mixed_assura_RCX/capgen_2m>* /home/bma/wavehorse/LVS/smic18_lvs
```

9.2 设置并运行 RCX

运行完 LVS 后，于 Virtuoso 菜单栏，Assuro->Run RCX，可以打开 Assura Parasitic Extraction Run 界面，如图 9.1 所示。

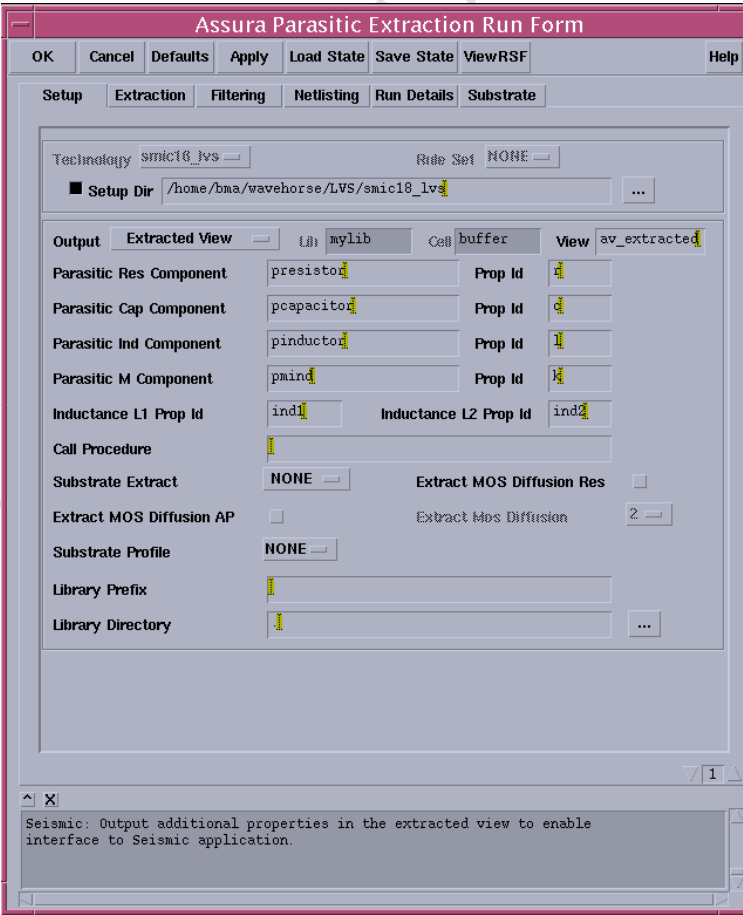


图 9.1 寄生参数提取设置窗口

该界面包含 6 个标签页，分别进行不同方面的设置。

Setup: 显示技术库名称和可用的 Rels Set，设置输出格式等。

Extraction: 设置需要提取的类型(如 R、C、RC、RCLK 等)，控制 R、C、L、K 提取典型值，选择提取范围(全芯片、指定的线网等)。

Filtering: 寄生 R、C、L、K 元件过滤器。

Netlisting: 输出类型中线网控制。

Run Details: 指定运行名称，设置临时文件以及 log 文件路径等。

本次实验中的大部分选项都采用默认设置，我们需要设置的几项如下。

①Setup 标签页。如图 9.2 所示。

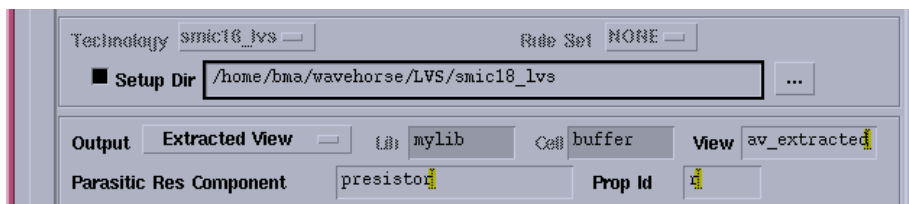


图 9.2 Setup 标签页

如果 LVS 运行正确，打开 RCX 后，Setup Dir 一栏会自动填写为 LVS 的库目录。Output 一栏选择寄生参数提取的输出格式，可以是 Spice 网表，Spectre 网表，DFII 格式文件等。这里我们选择 Extracted View，则会在当前的 buffer 单元下生成一个新的 View(默认名为 av_extracted)，对应的是一个从版图提取的包含了寄生元件的缓冲器电路。生成 av_extracted view 的好处是，可以直接通过 Cadence 的仿真环境对其进行仿真，并能直观的看到寄生元件等来源于版图的哪里。

②Extraction 标签页。如图 9.3 所示。

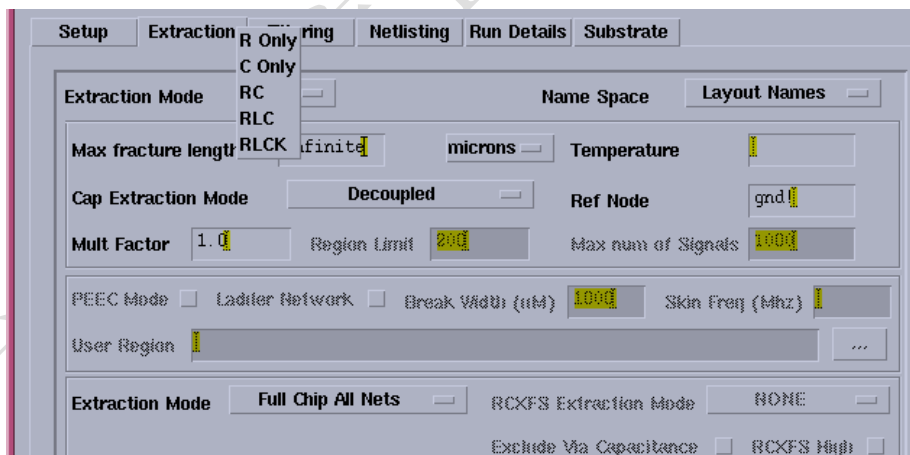


图 9.3 Extraction 标签页

于 Extraction Mode 栏选择 RC，即我们要提取的元件是电阻和电容。

Max fracture length 栏设置电阻提取方式。默认为 infinite，即每根连线被提取为一个电阻。如果输入具体数值，则每根连线被提取为若干电阻，电阻节点处提取一个等效电容。

Cap Extraction Mode 栏设置电容提取方式。选择 Decoupled 方式，则所有的寄生电容均被等效提取为相对指定节点的电容。如果选择 Coupled 方式，则所有在两根连线之间的电容均被提取，同时衬底电容被等耦合到相对指定节点的电容。

于 Ref Node 栏输入参考节点，这里我们就输入 gnd!。

设置完后注意保存当前设置，然后点击 OK 就开始运行 RCX 了。

运行过程中, 点击 Watch Log File, 可以查看运行过程中的 Log 文件。RCX 运行完后, 会有对话框提示 RCX 运行成功。在 Log 文件的底部, 显示出了 RCX 最终提取出的所有元件, 如图 9.4 所示。

```
instance count totals:
```

lib	cell	view	total
analogLib	pcapacitor	symbol	36
analogLib	presistor	symbol	48
pe	n18	ivpcell	4
pe	p18	ivpcell	4

图 9.4 RCX 提取出的元件

9.3 查看 RCX 结果

于 CIW 窗口, File->Open, 选择自己的库, Cell Name 为 buffer, View Name 为 av_extracted。点击 OK 按钮, 则会打开刚才提取生成的缓冲器的 av_extracted View, 如图 9.5 所示。

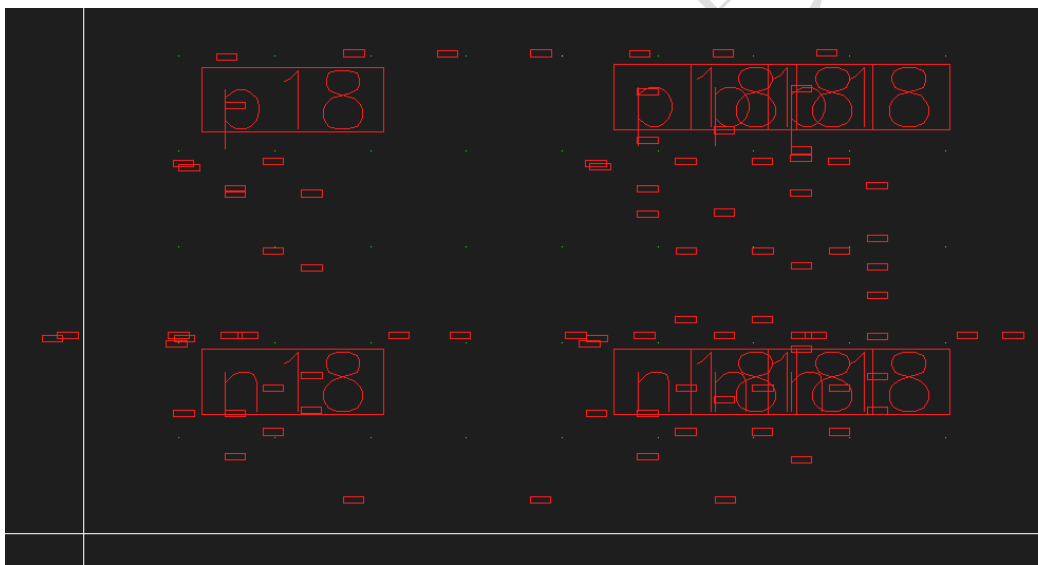


图 9.5 缓冲器的 av_extracted View

键入快捷键 **e**, 打开显示控制窗口, 选中 Use True BBox 和 Nets 确定, 并按 shift+f, 可以查看寄生元件的参数以及边线情况, 如图 9.7 和图 9.8 所示。

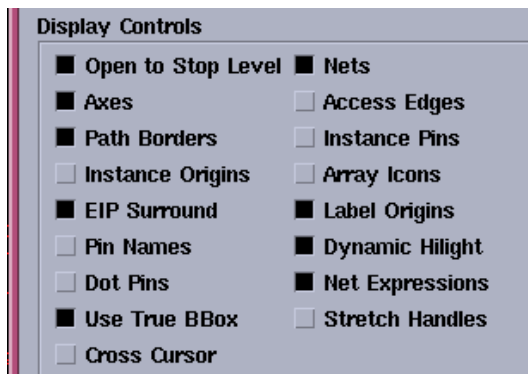


图 9.6 显示控制窗口

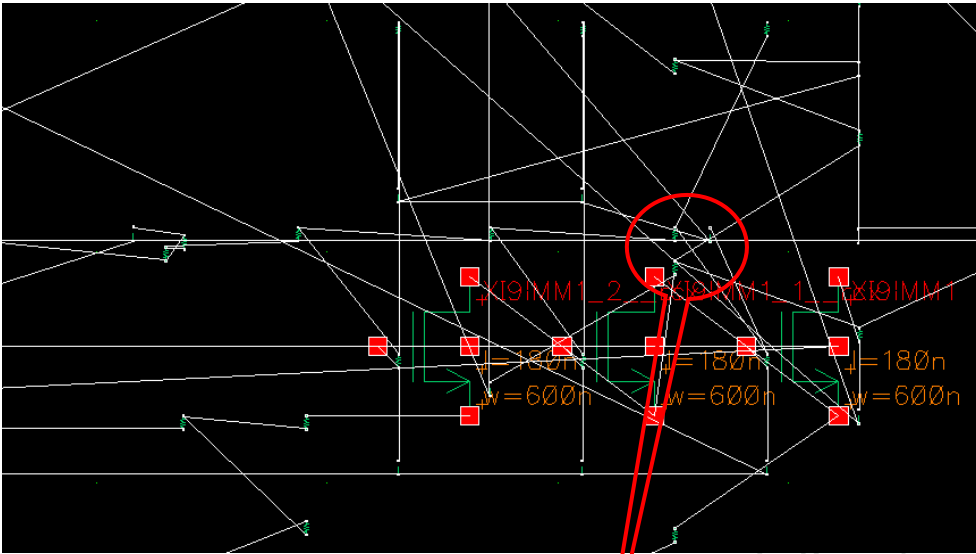


图 9.7 寄生元件的连线

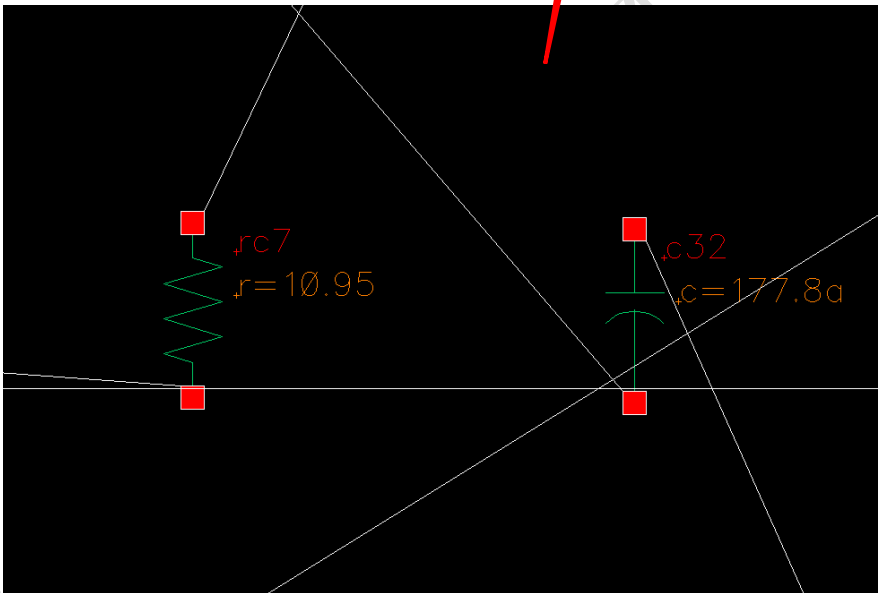


图 9.8 寄生元件的参数

十、后仿真(PS)

提取完寄生参数，就可以作后仿真了。我们利用 Cadence 的层次化工具 Hierarchy 来管理我们的设计，方便我们对比寄生参数提取前后的仿真结果。

10.1 运行 Hierarchy

于 CIW 窗口，File->New->Cell View，新建一个 Cell View，选择自己的库，Tool 一栏选择 Hierarchy Editor，输入 cell 名 buffer_test，点击 OK 确定，则会弹出如图 10.1 所示的对话框。

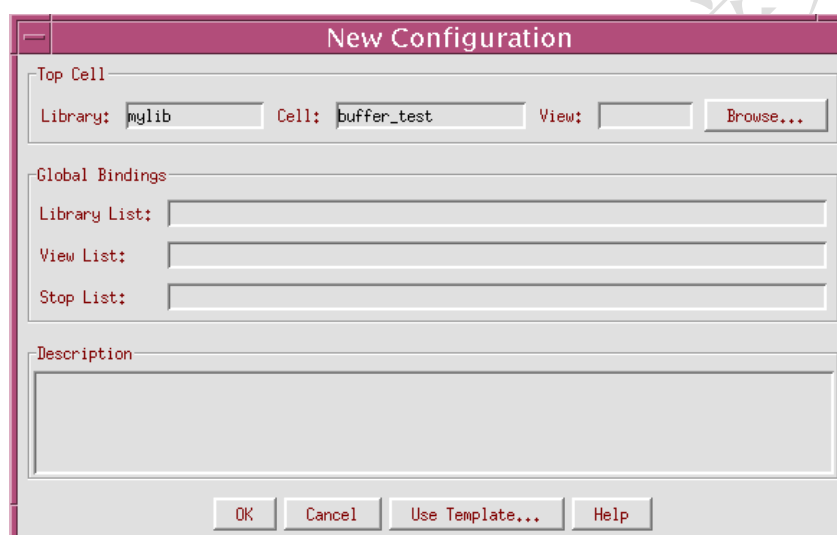


图 10.1 New Configuration

点击 Browse，弹出如图 10.2 所示的窗口，这里是要选择顶层 cell，我们就选择前面已经画好的缓冲器仿真电路的 schematic，点击 OK 确定。

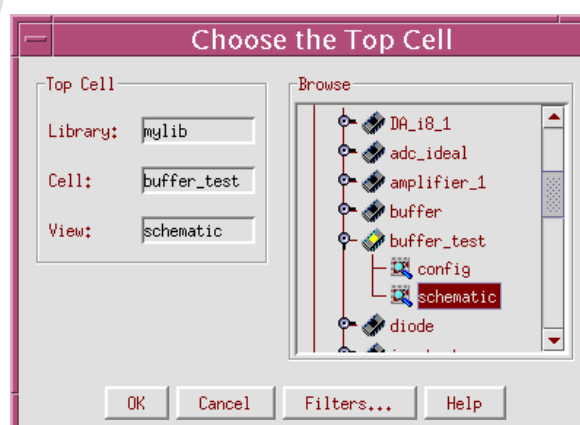


图 10.2 选择顶层元件

此时再于图 10.1 的 New Configuration 窗口中单击 “Use Template” 按钮，于新弹出的窗

口中选择 spectre, 点击 OK 确定, 则其它各项均自动填好。再点击 OK 确定。

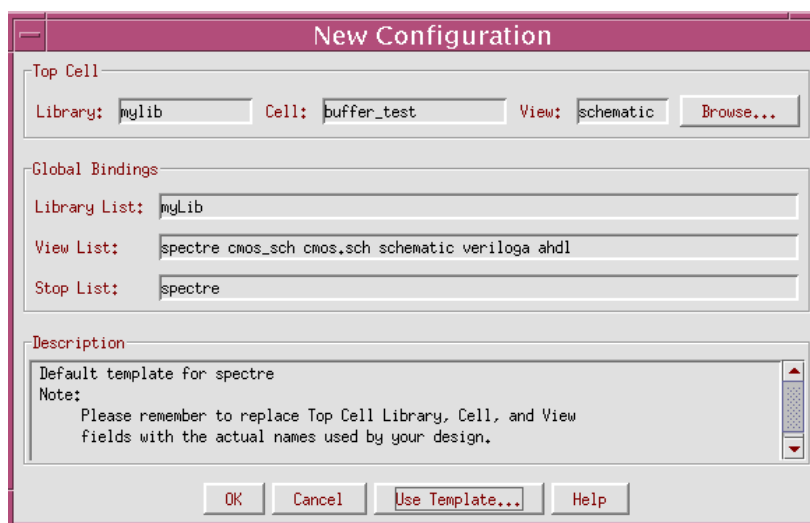


图 10.3 使用模板配置

此时就会弹出 Hierarchy Editor 的主窗口, 如图 10.4 所示。点击工具栏中的保存按钮保存当前设置。

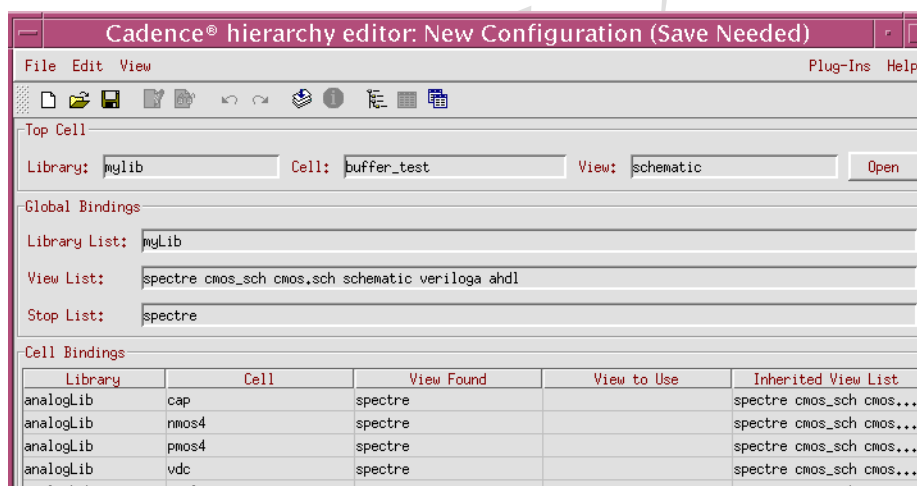



图 10.4 使用模板配置

Cell Bindings 栏中列出了当前顶层 cell 中 buffer_test 以及其所属 cell 中所有的元器件。

点击工具栏中的按钮 , 可以列出选中 cell 所属的所有器件, 如图 10.5 所示, 单元 buffer 中包含两个器件 I0 与 I1, 而这两个器件对应的 cell 是 inverter1。

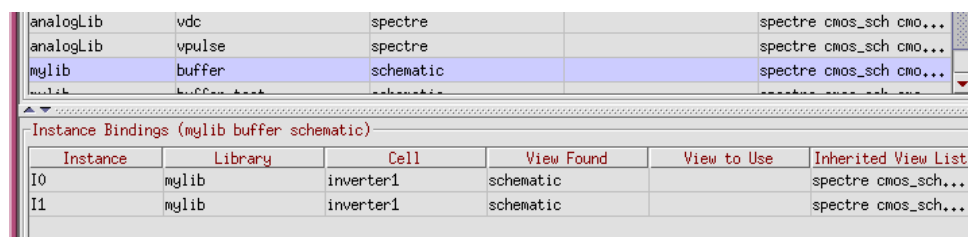



图 10.5 以表状结构显示器件

点击工具栏中的按钮，可以以树状结构显示器件。点击器件前面的箭头可以展开该器件下层的所有器件，如图 10.6 所示。

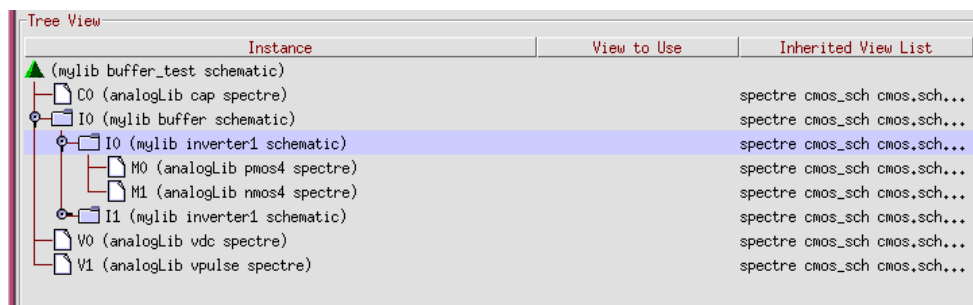



图 10.6 以树状结构显示器件

10.2 对不带寄生参数的原理图进行仿真

在上边打开的树状列表中，右键单击 I0(mylib buffer schematic)，Set Instance View->schematic，如图 10.7 所示。然后单击工具栏中的按钮进行更新，于弹出的对话框中单击 OK 确定。

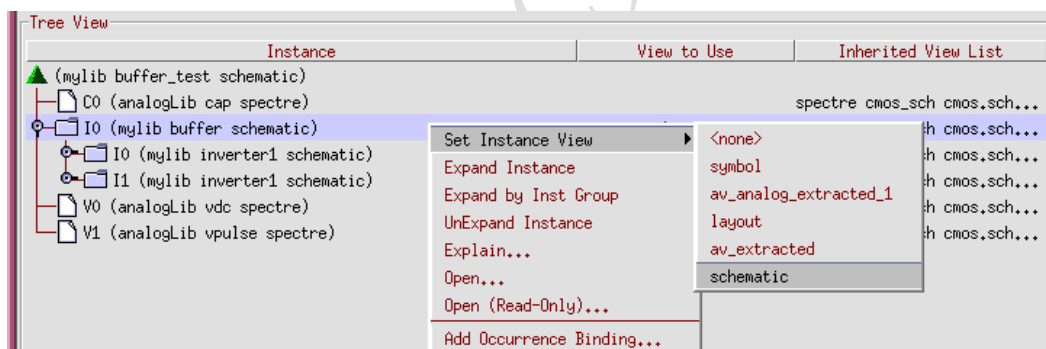



图 10.7 选择缓冲器的 View

再右键单击顶层 cell，即(mylib buffer_test schematic)，点击 Open，则会打开 buffer_test 的 schematic，接下来就是对该不带寄生参数的原理图进行仿真，方法和前边第五章完全一样，仿真波形参考第五章中的图。

仿真完毕后不必关闭波形窗口以及仿真环境。

10.3 对带寄生参数的原理图进行仿真

同上面的方法，打开的树状列表中，右键单击 I0(mylib buffer schematic)，Set Instance View->av_extracted，然后单击工具栏中的按钮进行更新，于弹出的对话框中单击 OK 确定。此时，buffer_test 中的元件缓冲器的 view 已经变成了带有寄生元件的 av_extracted view 了。

于刚才打开的仿真环境中，直接用前面设置好的仿真设置进行仿真，只是注意把仿真环境中 Plotting mode 选为 New Win，如图 10.8 所示，这样是为了方便对比寄生参数提取前后的仿真波形。

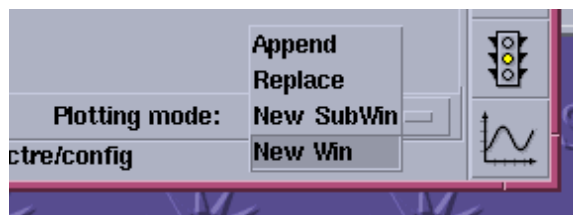


图 10.8 设置新窗口

图 10.9 为考虑了寄生参数后的仿真波形。与提取寄生参数前的仿真波形相比，似乎最大的区别是缓冲器作放大器时的低频增益变小。实际上低频增益并没有这么显著的变化，而是由于寄生参数的影响，缓冲器的共模电平范围发生微小变化，而该缓冲器的输入共模电平范围极窄，因此增益发生显著变化。我们把仿真变量 v_{in} 的初始值设为 0.898V，然后再仿真一次，就可以看到增益又达到 500 以上。这也正是该缓冲器并不能用来作放大器的原因之一。

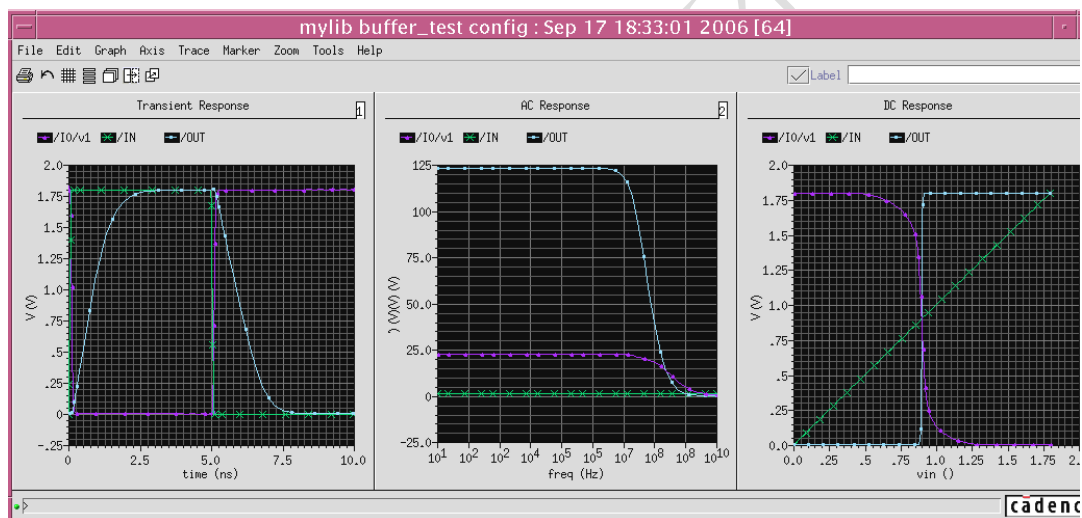


图 10.9 后仿真波形

对于缓冲器我们最关心的还是它的时序响应。在瞬态波形中，键入快捷键 **a**，选中起始点，再次键入快捷键 **a**，选中终点，系统会自动算出两点之间的 X、Y 坐标差值。利用这个我们可以很方便地测量时延。

寄生参数提取前后的电路仿真波形分别如图 10.10 和图 10.11 所示。可以看出，由于寄生参数的影响，缓冲器上升延时增加了 23ps。

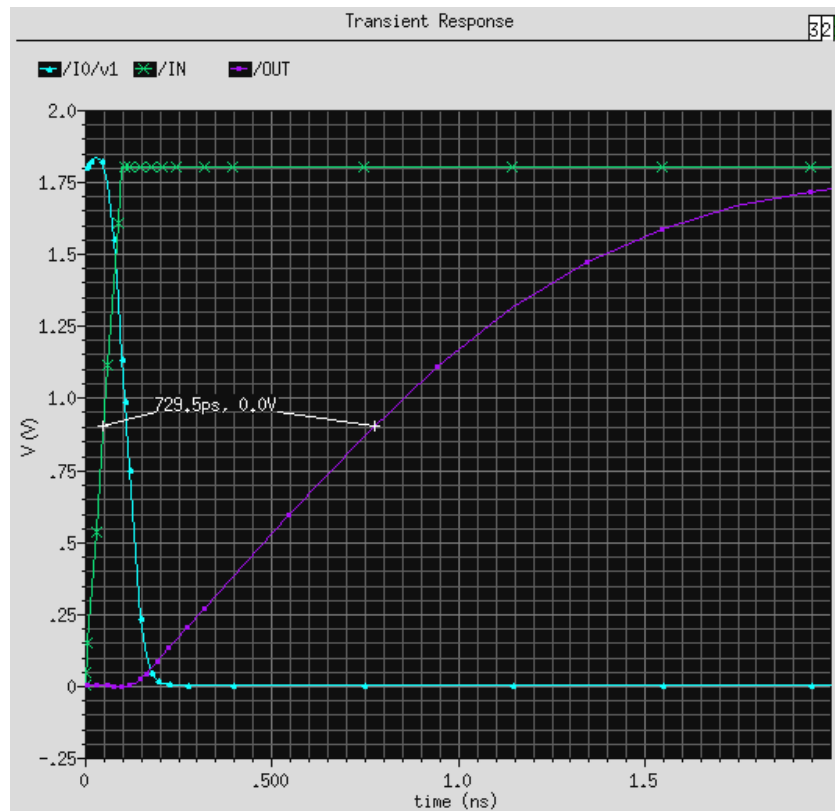


图 10.10 没有寄生参数时的延迟

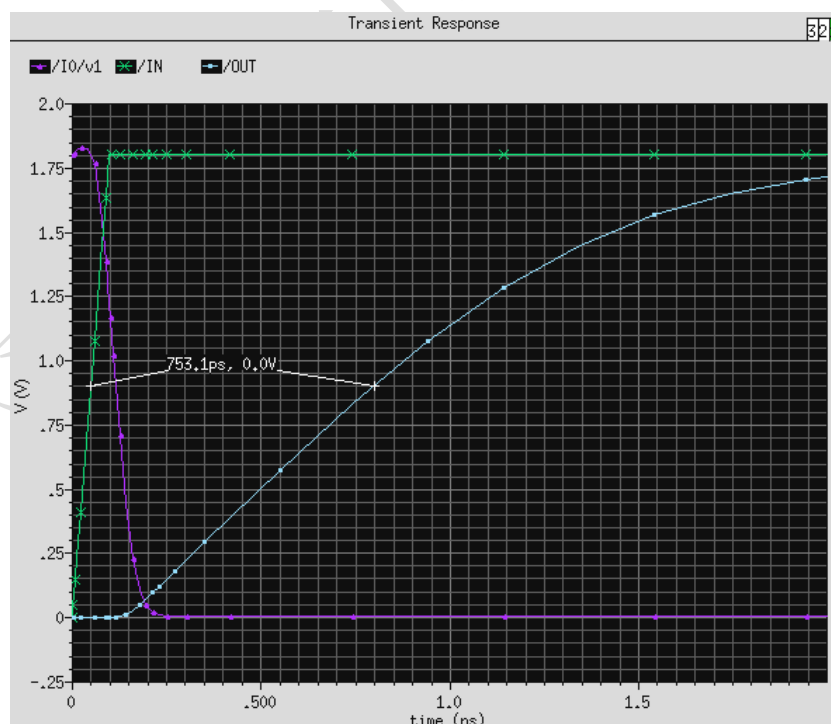


图 10.11 有寄生参数时的延迟