

# Hspice 的使用

——《集成电路课程  
设计》

李翔宇 2001年10月



# Hspice是什么？——从设计流程说起

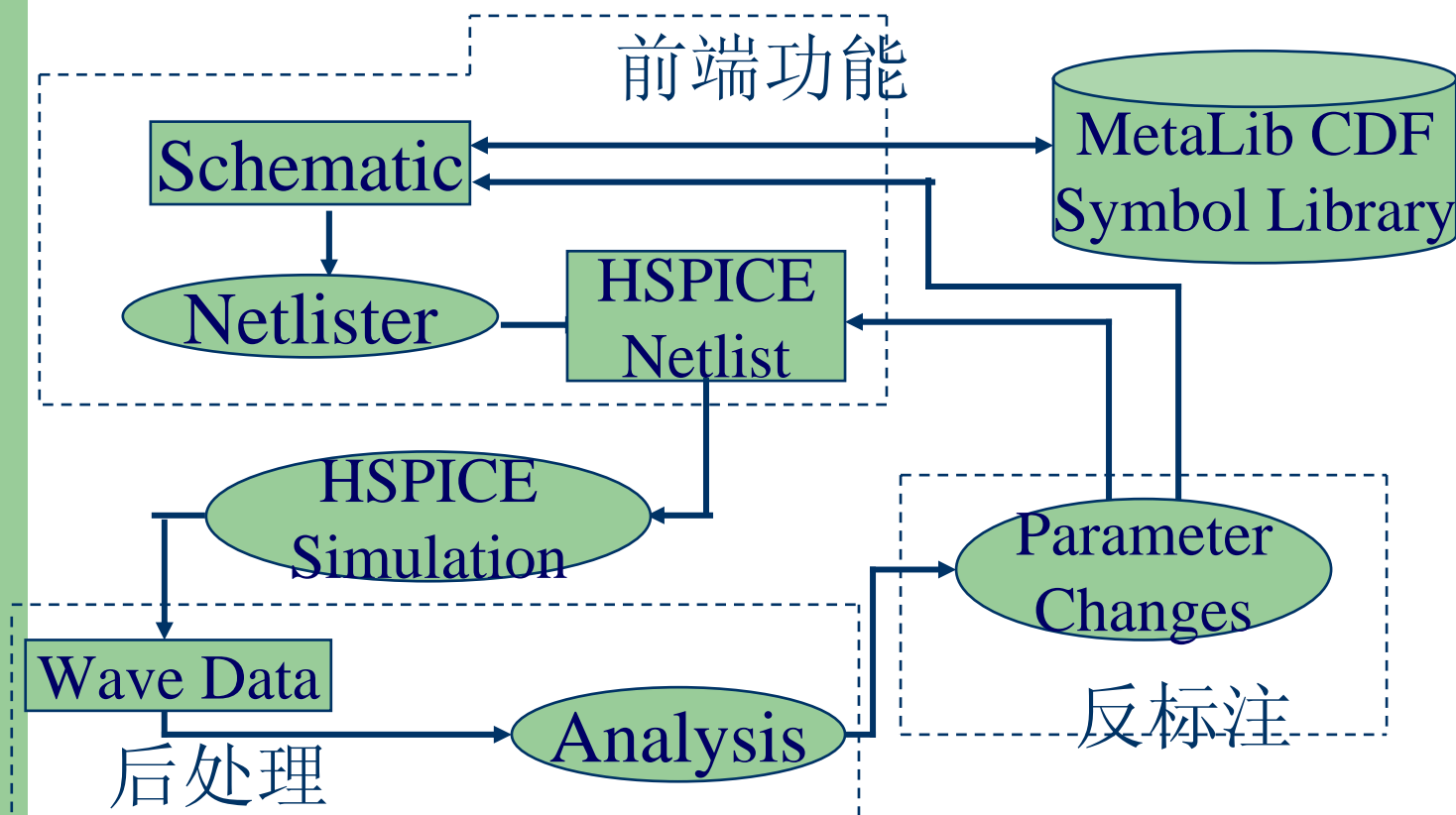
- 简单电路正向设计的典型流程
  1. 功能定义
  2. 行为设计
  3. 逻辑级电路设计——得到由基本逻辑单元组成的电路（数字电路）
  4. 逻辑级仿真（迭代）
  5. 选择合适的工艺库。把各基本功能单元映射至其上；或设计各单元晶体管级电路——得到电路级网表

# Hspice是什么？——从设计流程说起

6. 电路级仿真：验证各单元电路是否具有期望的功能，性能估计。（迭代）
7. 版图设计、DRC, LVS
8. 提取版图网表，进行后仿真：验证功能，估计性能。（迭代）

Hspice主要应用于电路级仿真、分析。可以辅助调整电路参数。得到功耗、延时等性能估计。

# Hspice的流程



# Hspice有哪些功能？

- § 电路级和行为级仿真
- § 直流特性分析、灵敏度分析
- § 交流特性分析
- § 瞬态分析
- § 电路优化（优化元件参数）
- § 温度特性分析
- § 噪声分析
- § 傅立叶分析
- § Monte Carlo, 最坏情况, 参数扫描, 数据表扫描
- § 功耗、各种电路参数（如H参数、T参数、s参数）等可扩展的性能分析

# Hspice的样子

- Hspice是一个在cmd shell窗口中运行的程序，无图形化界面；
- Hspice的输入网单文件是一个有特定格式的纯文本文件——可在任意的文本编辑工具中编辑；
- Hspice的输出也是一系列纯文本文件，根据不同分析要求，输出不同扩展名的文件。如：.lis .mea .dat .smt等。

# Hspice的样子

- HSPICE 的运行：在运行HSPICE之前，应该首先登录到SUN工作站上，并确保你的使用HSPICE的权限和环境变量已设好。
- 打开一个“终端”窗口，然后进入到你的工作目录下。输入行命令运行。
- hspice有两种工作模式：提示行模式和非提示行模式

# 两种工作模式——提示行模式

键入 **hspice**, 然后回车;

系统会提示你输入一些参数, 比如

**Enter input file name:**

此时输入你的**HSPICE**网表文件, 缺省的扩展名为**.sp**

**Enter output file name or directory: [<filename.lis>]**

缺省值为输入**HSPICE**网表文件名加上**.lis**扩展名。但**.sp**和**.lis**并不是必须。除此之外, 还有一些参数(这些参数的隐含值一般不需要更改), 直接回车即可。等你按照系统的提示确定所有的参数后, **HSPICE**就开始运行。



## 两种工作模式——非提示行模式

一般情况下的输入举例如下：

`hspice demo.sp` 或者

`hspice demo.sp > demo.lis`

# Hspice的输入——网单文件

文件结构:

.title	输入文件的标题
options	设置模拟的条件
Analysis statement	设置扫描变量、设置分析模式
.print/.plot/.graph/.probe	设置输出结果的显示方式
Sources (I or V)	设置输入激励
netlist	电路网表
.lib	元件库
.model libraries	元件模型描述
.end	结束语句

# Hspice的输入——网单文件

- 例 (The Star-Hspice netlist for the RC network circuit) :

```
.title A SIMPLE AC RUN
.OPTIONS LIST NODE POST
.OP
.AC DEC 10 1K 1MEG
.PRINT AC V(1) V(2) I(R2) I(C1)
V1 1 0 10 AC 1
R1 1 2 1K
R2 2 0 1K
C1 2 0 .001U
.END
```

# Hspice的输出

- 输出文件：一系列文本文件
  - \*.ic : initial conditions for the circuit
  - \*.lis : **text simulation output listing**
  - \*.mt0 : post-processor output for MEASURE statements
  - \*.pa0 : subcircuit path table
  - \*.st0 : run-time statistics
  - \*.tr0 ,\*.tr1....: post-processor output for transient analysis
  - \*.ac0,\*.ac1....: post-processor output for AC analysis
- MetaWave: 观察波形 (post-processor), 人机交互界面

# Hspice的输入——网单文件

- .TITLE 语句

.TITLE <string of up to 72 characters>

或者: <string of up to 72 characters>

如果是第二种形式, 字符串应该是输入文件的首行; 如果一个HSPICE语句出现在文件的首行, 则它将被认为是标题而不被执行。

- .END 语句

形式: .END <comment>

在 .END语句之后的文本将被当作注释而对模拟没有影响。

# Hspice的输入——网单文件

- 网表:

网表是描述电路元件和连接关系的部分，首先对电路的结点进行标记，不同结点起不同的名字。再说明各个元件的引脚连接到哪个结点及元件的类型和模型。一般格式为：

名称 器件的类型 器件所连接的节点 参数值

例: .....

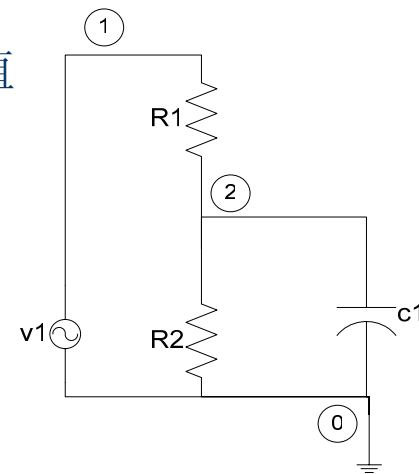
```
V1 1 0 10 AC 1
```

```
R1 1 2 1K
```

```
R2 2 0 1K
```

```
C1 2 0 .001U
```

.....



# Hspice的输入——网单文件

- 输入行格式
  - 输入网表文件不能是压缩格式；
  - 文件名、语句、等式的长度不能超过256字符；
  - 上标和下标将被忽略；
  - 用加号（+）表示续行，此时加号应该是新续之行的第一个非数字、非空格字符；
  - 星号（\*）和美元符号（\$）可以引出注释行，但\*必须是每行第一个字母，而\$一般跟在一个语句后，并与语句有至少一个空格。

# Hspice的输入——网表文件

- 分隔符
  - 包括: tab键, 空格, 逗号, 等号, 括号
  - 元件的属性由冒号分隔, 例如 M1:beta
  - 级别由句号指示, 例如 X1.A1.V 表示电路X1的子电路A1的节点V
- 常量:
  - M—毫, p—皮, n—纳, u—微, MEG—兆, 例如c1 1 2 10pF;
  - 单位可以省略, 例如c1 1 2 10p



# 电路网表

- § 元件名：
- 元件名以元件的关键字母开头：电阻—R，电容—C.....
  - 子电路的名字以“X”开头
  - 元件名不超过16个字符

- § 节点：
- 节点名长度不超过16个字符，可以包括句号和扩展名
  - 开始的零将被忽略：
  - 节点名可以用下列符号开始：# \_ ! %
  - 节点可以通过.GLOBAL语句定义成跨越所有子电路的

全局节点：.GLOBAL node1 node2 node3 ...

node1 node2 node3都是全局节点，例如电源和时钟名

- 节点0, GND, GND!, GROUND 都指全局的地电位节点

- § 元件语句： 名称 器件的类型 器件所连接的节点 参数值

# 电路网表

- 无源器件:

- 电阻:

`Rxxx n1 n2 <mname> <R=>resistance <AC=val>`

电阻值可以是表达式。例:

`Rterm input gnd R='sqrt(HERTZ)'`

`Rxxx 9 8 1 AC=1e10` 直流电阻1欧姆, 交流电阻为 $1e+10$ 欧姆

# 电路网表

- 无源器件:

- 电容:

一般形式:

`Cxxx n1 n2 <mname> <C=>capacitance`

例, `Cload driver output 1.0e-6`。

# 电路网表

- 无源器件：
  - 电感：
    - 一般形式：  
 $L_{xxx\ n1\ n2} \langle L \rangle \text{inductance}$

# 电路网表

- 有源器件:

- 二极管:

- `Dxxx nplus nminus mname /params`

- 模型中的寄生电阻串联在正极端。*

- 双极型晶体管:

- `Qxxx nc nb ne <ns> mname`

- JFET:

- `Jxxx nd ng ns <nb> mname`

# 电路网表

- 有源器件:

- MOSFET:

- Mxxx nd ng ns <nb> mname <params>

- Or

- Mxxx nd ng ns <nb> mname <width> <length> <other options...>

- 下面是一个CMOS反相器网表:

- .....

- Mn out in 0 0 NMOS W=1.2u L=1.2u

- Mp out in vdd vdd PMOS W=3u L=1.2u

- .....

# 电路网表

## § 子电路语句

▣ 子电路定义开始语句

```
.SUBCKT SUBNAM <node1 node2...>
```

其中，SUBNAM为子电路名，node1...为子电路外部节点号，不能为零。子电路中的节点号（除接地点），器件名，模型的说明均是局部量，可以和外部的相同。

例 `.SUBCKT OPAMP 1 2 3 4`

# 电路网表

## § 子电路语句

### 子电路终止语句

```
.ENDS <SUBNAM>
```

若后有子电路名，表示该子电路定义结束；若没有，表示所有子电路定义结束。

例 .ENDS OPAMP

### 子电路调用语句

```
X***** <node1 node2 ...> SUBNAM
```

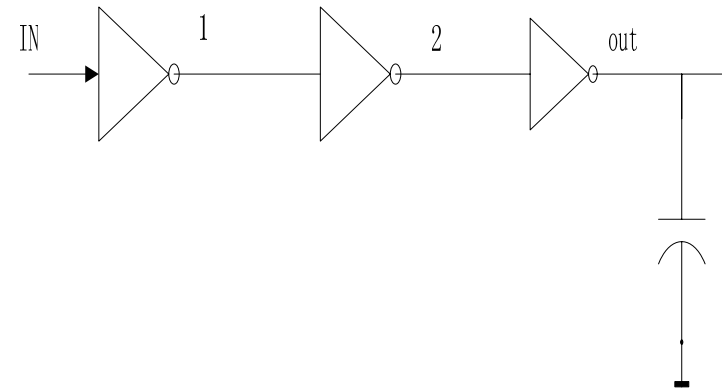
例 .X1 2 4 17 3 1 MULT1



# 子电路使用举例

下面是由前面举例的CMOS反相器组成的  
三级反相器链网表：

```
.....  
.global vdd  
.SUBCKT INV IN OUT wn=1.2u wp=1.2u  
Mn out in 0 0 NMOS W=wn L=1.2u  
Mp out in vdd vdd PMOS W=wp L=1.2u  
.ENDS  
X1 IN 1 INV WN=1.2U WP=3U  
X2 1 2 INV WN=1.2U WP=3U  
X3 2 OUT INV WN=1.2U WP=3U  
CL OUT 0 1PF  
VCC VDD 0 5V  
.....
```



# 电路网表

- 激励源:

- 独立源: 电压源—V, 电流源—I

Vxxx/Ixxx n+ n- <<DC=> dcval> <AC=acmag, <acphase>>  
+ <M=val>

例, V1 1 0 DC=5V                    或                    V1 1 0 5V

I1 1 0 DC=5mA                    或                    I1 1 0 5mA

交流模式: V1 1 0 AC=10V,90 幅度为10v, 相位为90度

交直流模式: V1 1 0 0.5v AC=10V,90 直流分量是0.5v

or

Vxxx/ Iyyy n+ n- <tranfun>

+ <M=val>

*tranfun: EXP, PULSE, PWL...*

<M=val>表示并联的电流源个数。

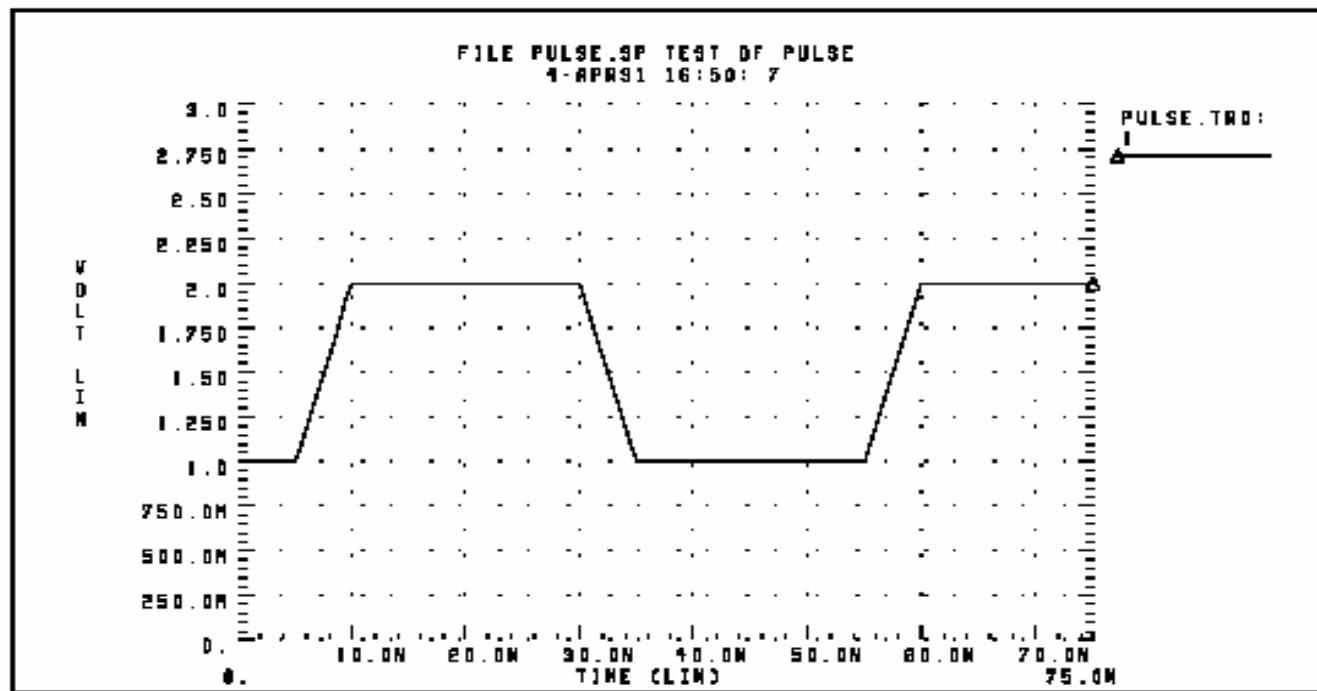
# 电路网表

- 激励源：
  - 独立源：
    - 脉冲形式：  $V_{xxx} n^+ n^- PU\langle LSE \rangle \langle (\rangle v1 v2 \langle td \langle tr \langle tf \langle pw \langle per \rangle \rangle \rangle \rangle \langle \rangle \rangle$

V1	值1
V2	值2
td	上升延迟时间
tr	上升时间
tf	下降时间
pw	脉冲宽度
per	周期

# 脉冲形式举例

例：VPU 3 0 PULSE(1 2 5N 5N 5N 20N 50N)



# 电路网表

- 激励源：

- 独立源：

- 正弦形式：Vxxx n+ n- SIN <( > vo va <freq <td <θ +<φ>>>> <)>

v0	失调值
va	幅度
freq	频率
td	延迟时间
θ	阻尼因子
φ	相位

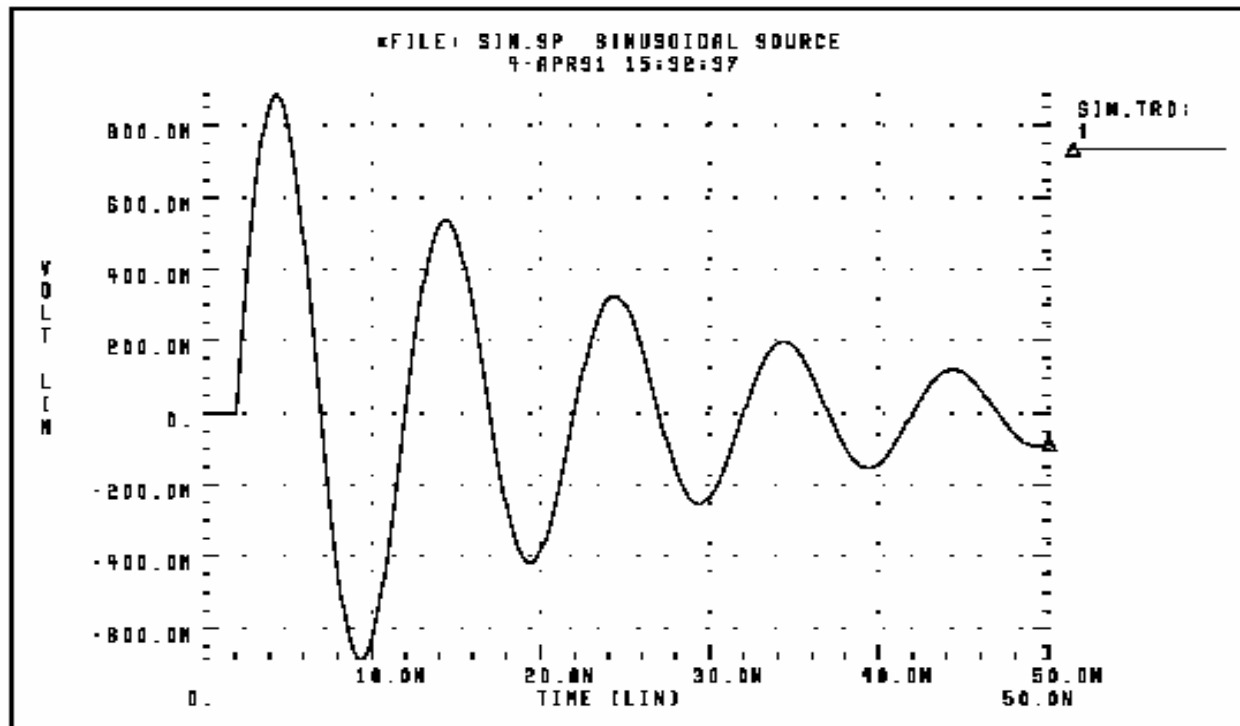
# 电路网表

得到的波形:

Time=0~td	$v_0 + v_a \cdot \sin(2\pi \cdot \phi / 360 + \text{Time})$
Time=td~瞬态分析的结束时间	$v_0 + v_a \cdot \text{Exp}[-(\text{Time} - \text{td}) \times \theta] \cdot \text{Sin}\{2\pi \cdot [\text{freq}(\text{Time} - \text{td}) + \phi / 360]\}$

# 正弦形式举例

例: `VIN 3 0 SIN (0 1 100MEG 1NS 1e10)`



# 电路网表

- 激励源：

- 独力源：

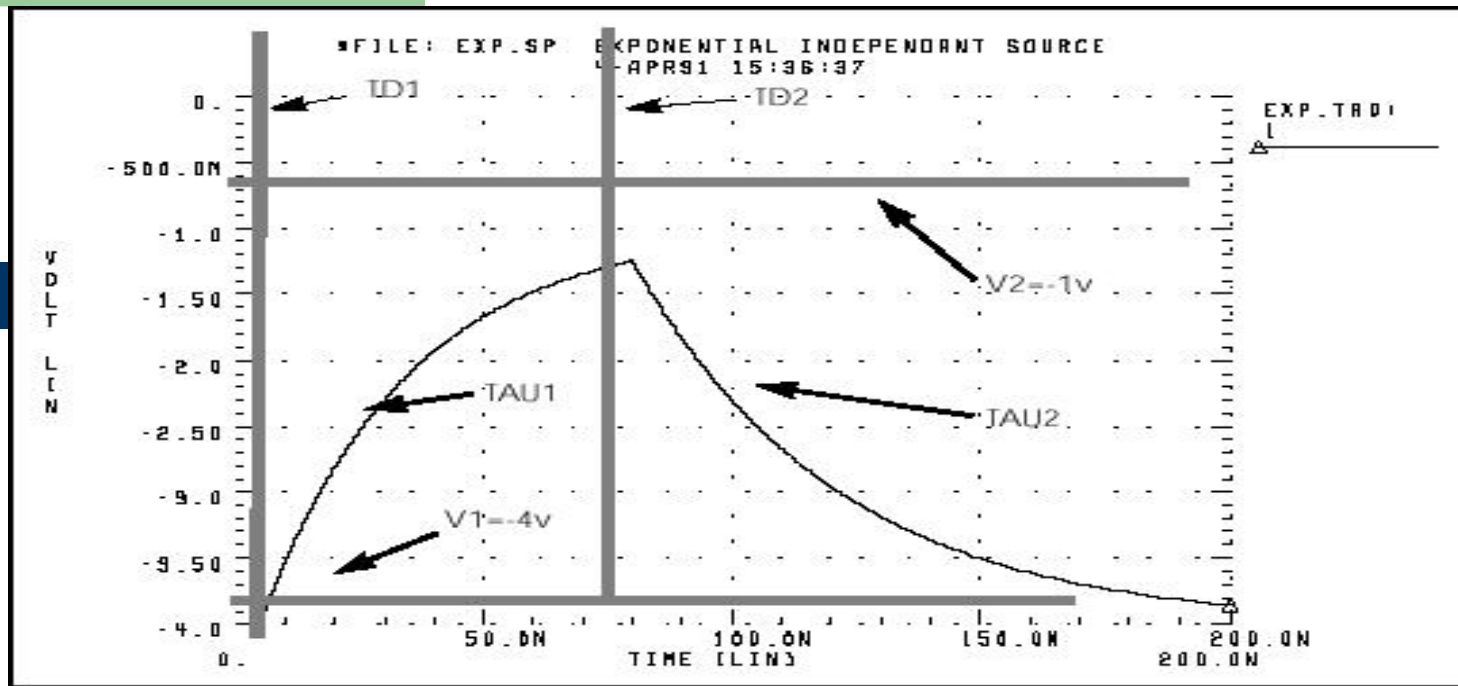
- 逐段线性形式： `pw1 <(> t1v1 <t2 v2 t3 v3... > <R  
<=repeat>>  
+ <TD=delay> <(>>`

*$v_i$ 是 $t_i$ 时刻的值，repeat 是开始重复的起始点；delay是延迟时间。*

- 指数形式： `EXP <(> v1 v2 <td1 <t1 <td2 <t2>>>> <(>>`

*V1是初始值，v2是峰值，td1是上升延迟时间，t1是上升时间常数，t2是下降时间常数。*





**Figure 5-3: Exponential Source Function**

```

*FILE: EXP.SP THE EXPONENTIAL WAVEFORM
.OPTIONS POST
.PARAM V1=-4 V2=-1 TD1=5N TAU1=30N TAU2=40N TD2=80N
V 1 0 EXP (V1 V2 TD1 TAU1 TD2 TAU2)
R 1 0 1
.TRAN .05N 200N
.END

```

# 完整的网表部分举例

前面反相器链的网表:

.....

```
.SUBCKT INV IN OUT wn=1.2u wp=1.2u
```

```
Mn out in 0 0 NMOS W=wn L=1.2u
```

```
Mp out in vdd vdd PMOS W=wp L=1.2u
```

```
.ENDS
```

```
X1 IN 1 INV WN=1.2U WP=3U
```

```
X2 1 2 INV WN=1.2U WP=3U
```

```
X3 2 OUT INV WN=1.2U WP=3U
```

```
CL OUT 0 1PF
```

```
VCC VDD 0 5V
```

```
VIN IN 0 PULSE(0 5V 10NS 1N 1N 50N 100N)
```

.....

# 模型卡

模型卡中列出了一系列元件的类型，并给出了各类型元器件的有关参数，对于不同类型的元件，参数的集合有不同的内容。一个模型对应于一类元件，不同的元件可以对应同一模型，其中各元件间的参数值可能不同，但参数集是一样的，一般值相同的参数的值在模型说明中给出。模型卡的语句是一条条.MODEL引导的模型说明语句。每个模型有一个名字。

# 模型卡

- 电阻模型 (wire RC) : .MODEL 模型名 R keyword=value

NOISE, RX: 热噪声参数,  $inr = \text{SQRT}(\text{NOISE} \cdot 4KT/R)$ ,  
噪声 =  $inr^2 \cdot RX^2$

- 电容模型: .MODEL 模型名 C parameter=value

## Capacitance Parameters

Name(Alias)	Units	Default	Description
CAP	F	0	Default capacitance value
CAPSW	F/m	0	Sidewall fringing capacitance
COX	F/m <sup>2</sup>	0	Bottomwall capacitance
DEL	m	0	Difference between drawn width and actual width or length DELeff = DEL · SCALM
DI		0	Relative dielectric constant
L	m	0	Default length of capacitor Lscaled = L · SHRINK · SCALM
SHRINK		1	Shrink factor
TC1	1/deg	0	First temperature coefficient for capacitance
TC2	1/deg <sup>2</sup>	0	Second temperature coefficient for capacitance
THICK	m	0	Insulator thickness
TREF	degC	TNOM	Reference temperature
W	m	0	Default width of capacitor Wscaled = W · SHRINK · SCALM

# 模型卡

有源器件的模型说明都有一个LEVEL参数，不同的LEVEL对应不同的模型参数集。

- 二极管模型：包括齐纳二极管、Schottky、扩散结

.MODEL 模型名 D <LEVEL = val> <keyword = val> ...

- nongeometric junction diode: 孤立元件 (LEVEL=1)  
电阻、电容、电流参数

例: .MODEL D D (CO=2PF, RS=1, IS=1P)

.MODEL DFOWLER D (LEVEL=2, TOX=100, JF=1E-10, EF=1E8)

.MODEL DGEO D (LEVEL=3, JS=1E-4, JSW=1E-8)

- geometric junction diode: 芯片中的二极管 (LEVEL=3)  
金属、多晶层的几何参数

模型说明中涉及的参数需与. OPTIONS的设置相配合:

# 模型卡

- MOS模型:

- .MODEL 模型名 PMOS <LEVEL=val> <parameters>

- .MODEL 模型名 NMOS <LEVEL=val> <parameters>

- LEVEL=1 常用于数字电路, 精度低、速度快

- LEVEL=2 耗尽型MOSFET

- LEVEL=13, 39, 49模拟电路, 精度高、速度慢

# MOSFET模型说明举例

例：1.2um CMOS工艺MOS管SPICE模型：

```
.MODEL NMOS NMOS LEVEL=2 LD=0.15U TOX=200.0E-10 VTO=0.74 KP=8.0E-05  
+NSUB=5.37E+15 GAMMA=0.54 PHI=0.6 U0=656 UEXP=0.157 UCRIT=31444  
+DELTA=2.34 VMAX=55261 XJ=0.25U LAMBDA=0.037 NFS=1E+12 NEFF=1.001  
+NSS=1E+11 TPG=1.0 RSH=70.00 PB=0.58  
+CGDO=4.3E-10 CGSO=4.3E-10 CJ=0.0003 MJ=0.66 CJSW=8.0E-10 MJSW=0.24
```

```
.MODEL PMOS PMOS LEVEL=2 LD=0.15U TOX=200.0E-10 VTO=-0.74 KP=2.7E-05  
+NSUB=4.33E+15 GAMMA=0.58 PHI=0.6 U0=262 UEXP=0.324 UCRIT=65720  
+DELTA=1.79 VMAX=25694 XJ=0.25U LAMBDA=0.061 NFS=1E+12 NEFF=1.001  
+NSS=1E+11 TPG=-1.0 RSH=121.00 PB=0.64  
+CGDO=4.3E-10 CGSO=4.3E-10 CJ=0.0005 MJ=0.51 CJSW=1.35E-10 MJSW=0.24
```



# 模型卡

- BJT模型:

.MODEL mname NPN <( < <pname1 = val1> ... < ) >

or

.MODEL mname PNP <pname1 = val1> ...

模型参数中一般包括LEVEL, 说明哪种模型, 不同级的模型有不同的模型参数集。

## BJT模型说明举例:

```
.MODEL NPN NPN BF=100 BR=1 IS=1.E-17 VAF=50  
+TF=10E-12 TR=5E-9 IKF=2E-2 IKR=0.5  
+RE=0 RC=75 RB=120  
+CJE=20E-15 VJE=0.8 MJE=0.5 CJC=22E-15 VJC=0.7  
+MJC=0.33 CJS=47E-15 VJS=0.7 MJS=0.33
```

# 控制卡

控制卡是hspice输入文件的命令部分，告诉hspice要进行哪些操作和运算，并给出相关的参数——如分析方式、输出的变量等。其内容主要包括选项语句（.OPTIONS）、分析命令语句、输出控制语句几类。这些语句格式的共同特点是都由保留字引导，后面更随相应的参数，在保留字前要加“.”

# 控制卡

- LIB 语句:

```
.lib '<filepath>filename' entryname
```

该语句根据文件路径和文件名来调用一个库文件，一般该文件包含器件模型。

例 `.lib 'MODELS' cmos1`

MODELS文件:

...

```
.MODEL CMOS1 nmos ...
```

- .INCLUDE语句: 引用一个文件，被引用的文件置于引用文件前。

例:

LNA

```
.include "me98xxxx/model.sp"
```

...

# 直流分析仿真流程

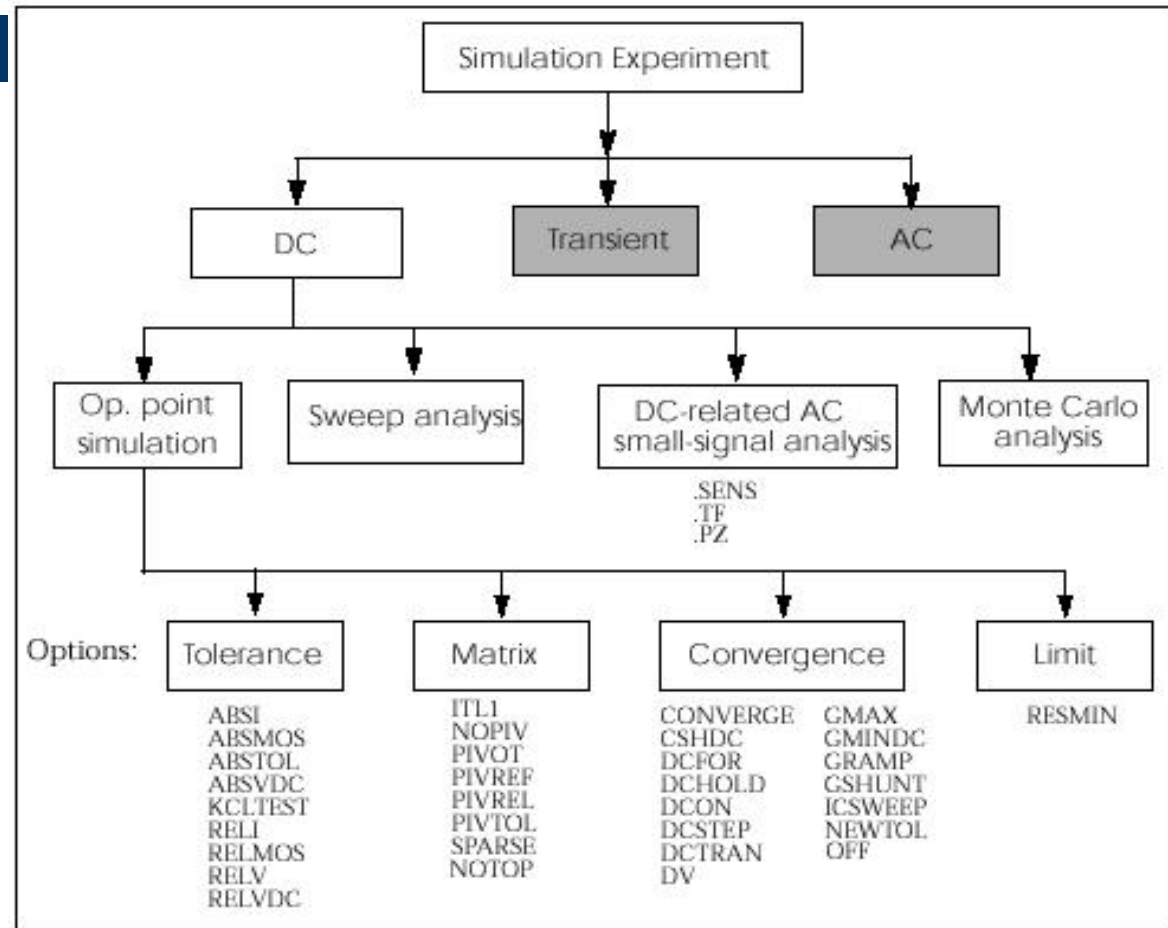


Figure 9-1: DC Initialization and Operating Point Analysis Simulation Flow

# 控制卡——直流分析

- .OP: 直流工作点分析

会在输出文件中列出一些直流参数和各结点的工作点电压与支路电流、静态功耗。

例：对前面反相器链电路的直流工作点分析。

# 控制卡——直流分析

- .dc:

- 扫描: .DC 变量1扫描 <变量2扫描>...

扫描: var1 START STOP STEP/<SWEEP var2 type np start2 stop2>, type-DEC(十进位) /OCT(倍频) /LIN(线性) /DATA= datanm/POI(列表)

Np—单位范围内的点数(依type而定)。SWEEP后的变量可是电压、电流或温度等变量。

例: .DC xval 1k 10k .5k SWEEP TEMP LIN 5 25 125

.DC TEMP POI 5 0 30 50 100 125

对前面反相器链的直流特性扫描:

...

VIN IN 0

.DC VIN 0 5V 0.1V (从0v到5v, 步长0.1v)

...

# 直流分析举例

例：分析反相器链的直流传输特性和工作点

.....

```
.global vdd
```

```
.SUBCKT INV IN OUT wn=1.2u wp=1.2u
```

.....

```
.ENDS
```

```
X1      IN      1      INV      WN=1.2U WP=3U
```

```
X2      1      2      INV      WN=1.2U WP=3U
```

```
X3      2      OUT    INV      WN=1.2U WP=3U
```

```
CL      OUT     0      1PF
```

```
VCC     VDD 0 5V
```

```
VIN     IN      0
```

```
.DC VIN 0 5V 0.1V
```

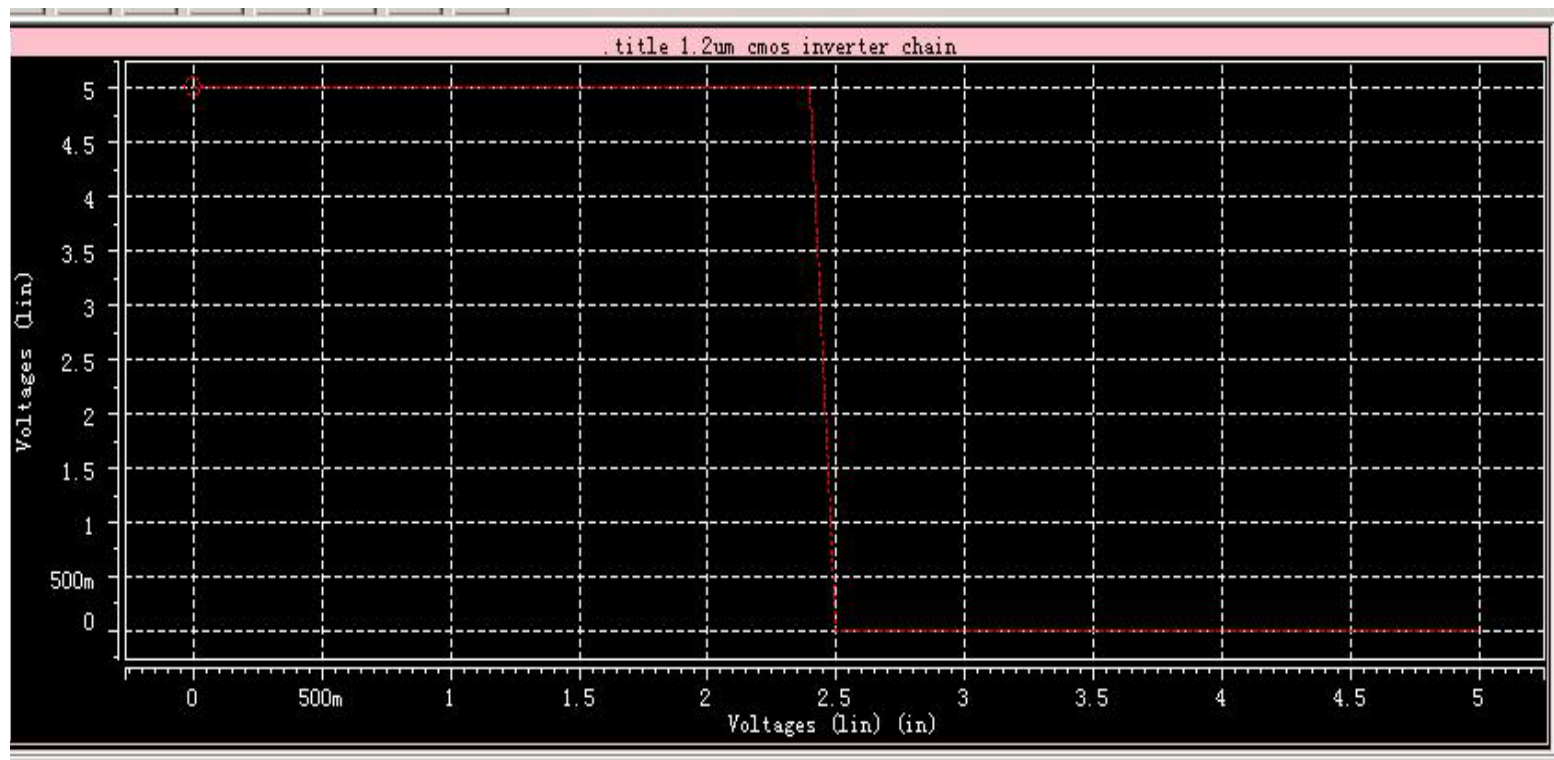
```
.OP
```

.....

```
.END
```



# 在Metawave中输出的直流传输特性曲线



# 控制卡——直流分析

- 小信号灵敏度分析: `.SENS ov1 <ov2 ...>`  
Ov1, ov2是做灵敏度分析的支路电流或节点电压。  
计算给出输出变量对于每个电路参数的偏导，并做归一化。同一输出变量对所有电路参数的灵敏度和为100%  
%
- 小信号转移函数: `.TF ov srcnam`  
ov是输出变量，srcnam是输入源。  
\*  
`.TF V(5,3) VIN` 计算 $V(5,3)/V_{IN}$

# 瞬态分析仿真流程

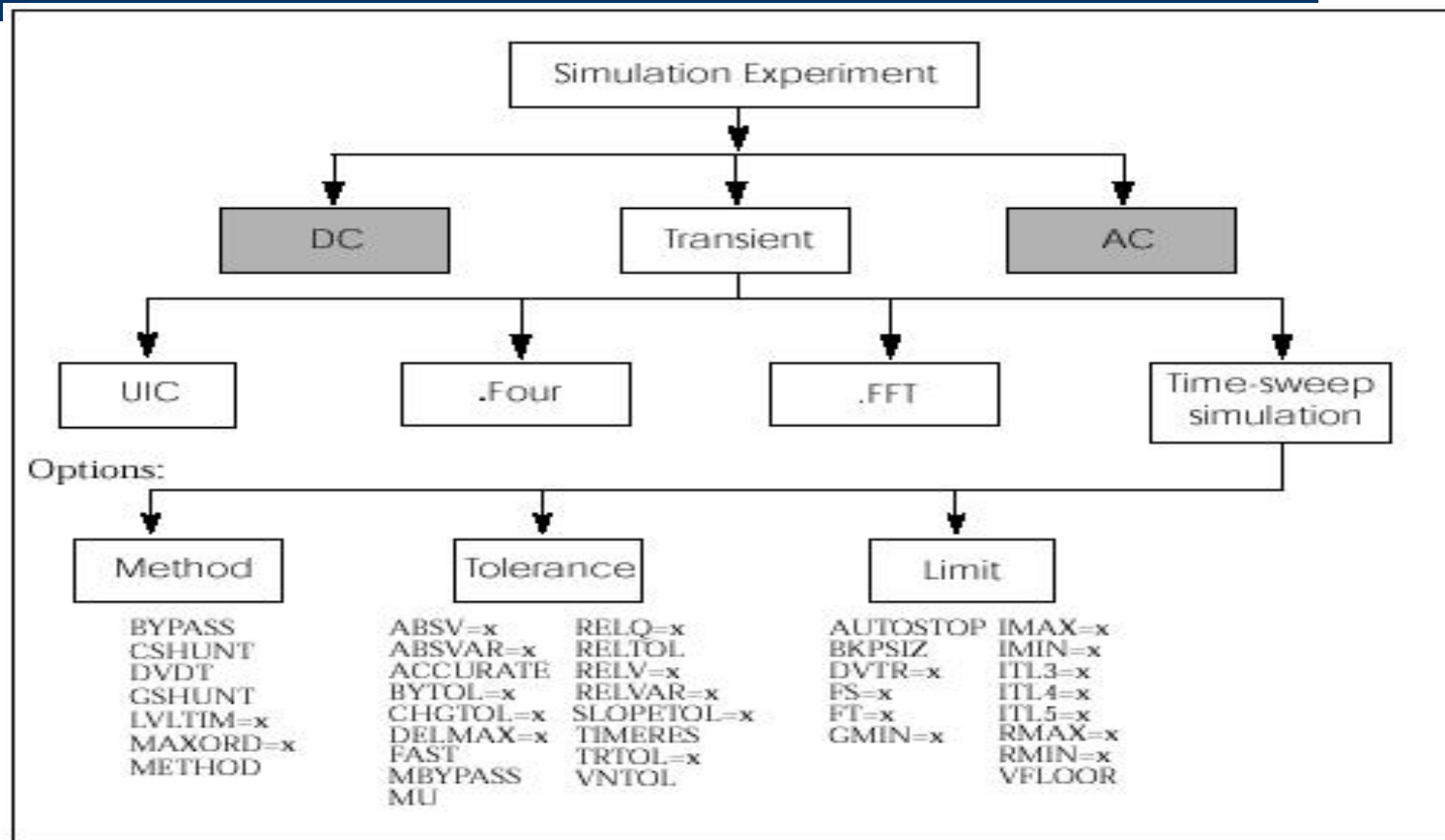


Figure 10-1: Transient Analysis Simulation Flow

# 控制卡——瞬态分析

- 一般分析:

```
.TRAN var1 START=start1 STOP=stop1 STEP=incr1
```

or

```
.TRAN tincr1 tstop1 <tincr2 tstop2 ... tincrN tstopN>
```

```
+ <START=val> <UIC>
```

**起始时刻和步长都指的是输出打印的时刻点，计算的时间步长由 hspice 自己决定。UIC 参数表示使用 .IC 语句指定的节点初始值。**

例: `.TRAN .1NS 25NS 1NS 40NS START=10NS`

0—25ns, 步长0.1ns, 25ns—40ns, 步长1ns; 从10ns开始输出结果。

```
.TRAN 1NS 100NS $以0.1ns的步长输出到100ns
```

# 控制卡——瞬态分析

- *Fourier*分析:

```
.FOUR freq ov1 <ov2 ov3 ...>
```

Freq—基频, ov1、ov2...—输出变量

```
CMOS INVERTER
```

```
M1 2 1 0 0 NMOS W=20U L=5U
```

```
M2 2 1 3 3 PMOS W=40U L=5U
```

```
VDD 3 0 5
```

```
VIN 1 0 SIN 2.5 2.5 20MEG
```

```
.MODEL NMOS NMOS LEVEL=3 CGDO=.2N CGSO=.2N CGBO=2N
```

```
.MODEL PMOS PMOS LEVEL=3 CGDO=.2N CGSO=.2N CGBO=2N
```

```
.OP
```

```
.TRAN 1N 100N
```

```
.FOUR 20MEG V(2)
```

```
.PRINT TRAN V(2) V(1)
```

```
.END
```

## 输出文件傅立叶分析结果部分:

fourier components of transient response v(2)

dc component = 2.430D+00

harmonic frequency fourier

no	(hz)	component	normalized component	phase (deg)	normalized phase(deg)
1	20.0000x	3.0462	1.0000	176.5386	0.
2	40.0000x	115.7006m	37.9817m	-106.2672	-282.8057
3	60.0000x	753.0446m	247.2061m	170.7288	-5.8098
4	80.0000x	77.8910m	25.5697m	-125.9511	-302.4897
5	100.0000x	296.5549m	97.3517m	164.5430	-11.9956
6	120.0000x	50.0994m	16.4464m	-148.1115	324.6501
7	140.0000x	125.2127m	41.1043m	157.7399	-18.7987
8	160.0000x	25.6916m	8.4339m	172.9579	-3.5807
9	180.0000x	47.7347m	15.6701m	154.1858	-22.3528

total harmonic distortion = 27.3791 percent

# 控制卡——交流分析

- .AC:

- 一般频域扫描:

- .AC type np fstart fstop <SWEEP var start stop incr>

- or

- .AC type np fstart fstop <SWEEP var type np start stop>

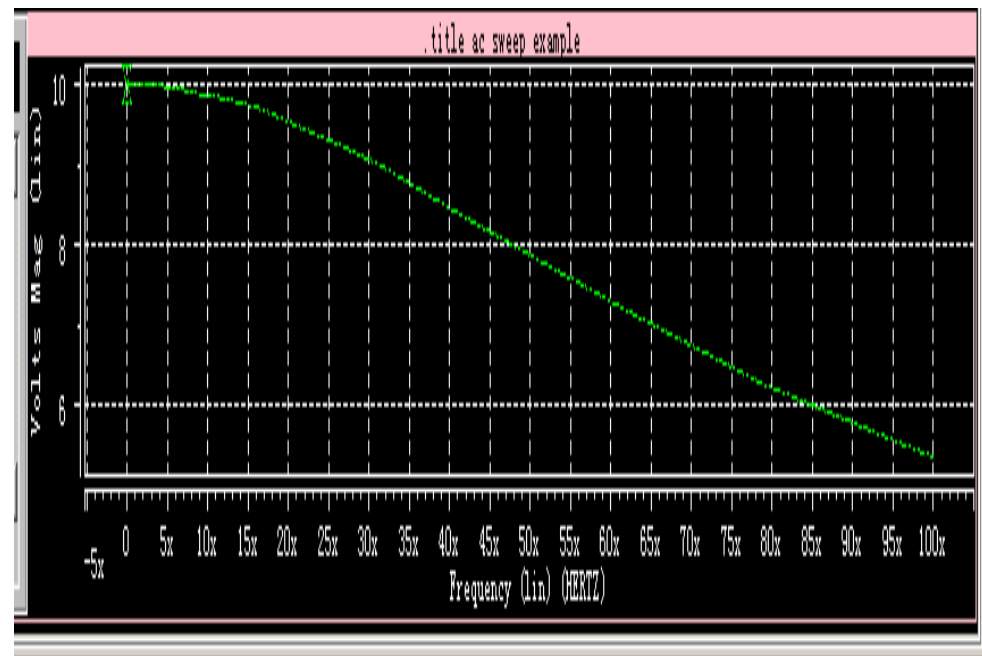
- or

- .AC var1 START = start1 STOP = stop1 STEP = incr1

- 例: .AC DEC 10 1K 100MEG 1kHz-100MHz,每10倍频10个采样点。

# 低通滤波器频率响应举例

```
.title ac sweep example  
.OPTIONS POST  
R1 in 1 5  
C1 1 0 500pF  
V1 IN 0 0 AC=10V,37  
.AC OCT 10 1 100MEG  
.PRINT ac V(1)  
.END
```





# 控制卡——交流分析

- 噪声分析：

用来计算各个器件的噪声对输出节点的影响并给出其均方根并输出，可完成.AC语句规定的各频率的计算，应在.AC分析之后。

```
.NOISE ovv srcnam inter
```

Ovv—输出变量，srcnam—输入源，inter—频率间隔

例：

```
.title ac sweep example
```

```
.OPTIONS POST
```

```
R1 in 1 5
```

```
C1 1 0 500pf
```

```
V1 IN 0 0 AC=10V,37
```

```
.AC OCT 10 1 100MEG
```

```
.noise v(1) v1 20——分析1点电压的噪声情况，噪声源为V1端口
```

```
.END
```

## Lis文件中输出的噪声分析结果

```
1 ***** Star-HSPICE -- 1999.4 (19991220) 22:12:12 04/16/2002 pcnt
*****
.title ac sweep example
***** noise analysis          tnom= 25.000 temp= 25.000
*****

frequency = 1.0000 hz
**** resistor squared noise voltages (sq v/hz)
element 0:r1
total 8.233e-20
rx 5.0000
**** total output noise voltage      = 8.233e-20 sq v/hz
                                     = 286.9260p v/rt hz

transfer function value:
v(1)/v1          = 1.0000
equivalent input noise at v1      = 286.9260p /rt hz
**** the results of the sqrt of integral (v**2 / freq)
from fstart upto 1.0000 hz. using more freq points
results in more accurate total noise values.
**** total output noise voltage = 0. volts
**** total equivalent input noise = 0.
.....
```

# 参数扫描

1. 要定义扫描的参数；
2. 在电路中引用参数；
3. 给出参数取值列表（DATA）；
4. 在分析语句中加入DATA=datanam，指定参数对应的数值表。

## 参数扫描举例

```
.title ac sweep example  
.OPTIONS POST  
.param cv=500pf
```

```
R1 in 1 5  
C1 1 0 cv
```

```
V1 IN 0 0 AC=10V,37
```

```
.data cv_table
```

```
cv
```

```
300p
```

```
500p
```

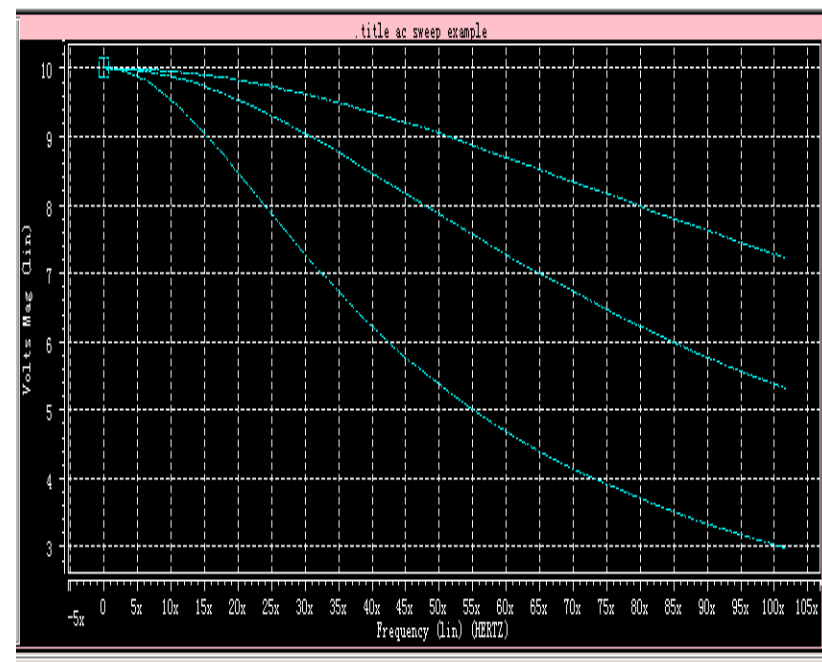
```
1n
```

```
.enddata
```

```
.AC OCT 10 1 100MEG sweep data=cv_table
```

```
*.PRINT ac V(1)
```

```
.END
```



# 控制卡

- 温度分析：
  - 与直流或瞬态分析等命令结合使用：例如对反相器链瞬态特性的温度扫描：

.....

```
VIN IN 0 PULSE(0 5V 10NS 1N 1N  
50N 100N)
```

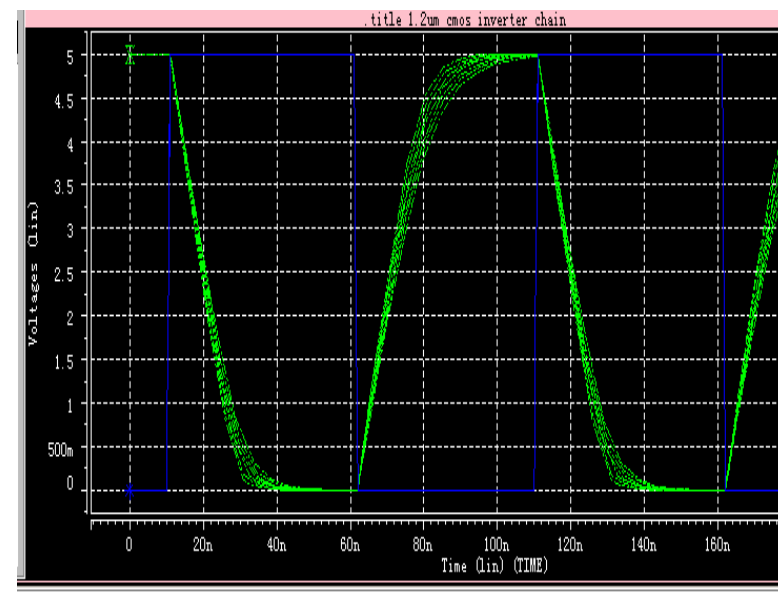
```
.TRAN 1N 200N sweep temp 0 125  
20
```

```
.PRINT V(OUT)
```

```
.END
```

曲线如右：

- `.TEMP t1 <t2 <t3 ... >>`：会产生一系列的瞬态分析文件：`tr0, tr1...`，在metawave中对应不同的分析。



# 控制卡

- 初始化: `.IC var1=val1 <var2=VAL2>...`  
进行含有双稳态电路的模拟时往往要用`.IC`语句。

# 控制卡

- .OPTIONS:

该语句允许用户重新设置程序的参数或控制程序的功能。常用的一些如下:

node: 列出个节点的元件端点, 便于查错;

post: 使输出数据可以使用 MetaWaves 浏览(即将数据输出到 post processor;

list: 列出元件列表;

MEASDGT: .MEASURE语句输出的有效数字位数

例: .option post probe \$MetaWaves只观察.probe语句输出的变量。

# 输出控制

- 输出语句:

- **.PRINT**: 在输出的list文件中打印数字的分析结果, 如果.OPTIONS中有POST则同时输出到post-processor中。
- **.PLOT**: 在输出的list文件中打印低分辨率的曲线 (由ASCII字符组成), 如果.OPTIONS中有POST则同时输出到post-processor中。
- **.GRAPH**: 生成用于打印机或PostScript格式的高分辨率曲线。
- **.PROBE**: 把数据输出到post-processor, 而不输出到list文件。
- **.MEASURE**: 输出用户定义的分析结果到mt0文件, 如果.OPTIONS中有POST则同时输出到post-processor中。
- **.OP, .TF, .NOISE, .SENS和.FOUR**都提供直接输出功能。



# 输出控制

- `.PRINT: .PRINT antype ov1 <ov2 ... ov32>`

Antype – AC/DC/TRAN;

Ovi: 输出变量, 可以有以下形式:

V (1) 节点1的电平, v (1, 2) 1、2间的电压, V (R1) 电阻R1的电压;

VM (1) v1的幅值, VR (1) v1的实部, VI (1) v1的虚部,  
VP (1) v1的相位, VDB (1) v1的分贝值;

(电流与以上类似) ;

INOISE,ONOISE;

# 输出控制

- `.PLOT: .PLOT antype ov1 <(plo1, phi1)> ... <ov32>`  
`+ <(plo32, phi32)>`

`(plo1, phi1) - ov1` 绘图的上下限。

- `.PROBE: .PROBE antype ov1 ... <ov32>`

\*元件电流引用: BJT:  $I1(Qx) - I_c$ ,  $I2(Qx) - I_b$ ,  
 $I3(Qx) - I_e$ ,  $I4(Qx) -$ 衬底电流;

MOS:  $I1(Mx) - I_d$ ,  $I2(Mx) - I_g$ ,  $I3(Mx) - I_s$ ,  
 $I4(Mx) -$ 衬底电流。

## 几个输出语句例子

- `.PRINT ac V(1)`
- `.TRAN 1N 200N`  
`. PROBE V(OUT)`
- `.NOISE v(out) vin 10`  
`.print noise onoise inoise`
- `.NET V(8) VIN RIN=50 ROUT=50`（二端口网络定义）  
`.PLOT AC ZIN(R) ZIN(P) zout(r) zout(i)`  
`.print im(rd)`  
`.PRINT AC S11(DB) S21(m) S22(DB)`

# Hspice的输出

- `.MEASURE`:
  - 包括以下测量模式：
    - Rise, fall, and delay
    - Find-when
    - Equation evaluation
    - Average, RMS, min, max, and peak-to-peak
    - Integral evaluation
    - Derivative evaluation
    - Relative error
  - `.MEASURE <DC|AC|TRAN> result TRIG ... TARG ...`

# 输出控制

- .MEASURE:

- Rise, Fall, Delay模式:

- .MEASURE <DC|AC|TRAN> result TRIG ... TARG ...

- Result—测量结果的名字， TRIG ... TARG 一起始...中止  
(依分析内容不同可是时刻、频率...)

# 输出控制

- . MEASURE :

- TRIG和TARG的格式

```
TRIG trig_var VAL=trig_val <TD=time_delay>  
<CROSS=c>
```

```
+ <RISE=r> <FALL=f> 或 TRIG AT=val
```

```
TARG targ_var VAL=targ_val <TD=time_delay>
```

```
+ <CROSS=c|LAST> <RISE=r|LAST> <FALL=f|LAST>
```

trig\_var和targ\_var指定引发变量；val指出上升、下降、或反转的临界点；time\_delay指出开始测量时跳过的时间量；CROSS, RISE, FALL分别指出开始触发的次数；LAST说明到最后一次；

```
例 .meas tran tdlay trig v(1) val=2.5 td=10n rise=2  
+           targ v(2) val=2.5 fall=2
```

# 计算反相器链电路的延迟时间

```
.TITLE 1.2UM CMOS INVERTER CHAIN
.INCLUDE "models.sp"

.....
X1 IN 1 INV WN=1.2U WP=3U
X2 1 2 INV WN=1.2U WP=3U
X3 2 OUT INV WN=1.2U WP=3U
CL OUT 0 1PF
VCC VDD 0 5V
VIN IN 0 PULSE(0 5V 10NS 1N 1N 50N 100N)
.TRAN 1N 200N
.measure tran tdelay trig v(in) val=2.5 td=8ns rise=1
+ targ v(out) val=2.5 td=9n fall=1
.END
```

# 测量结果

输出的Chain.mt0文件:

```
$DATA1 SOURCE='HSPICE'
```

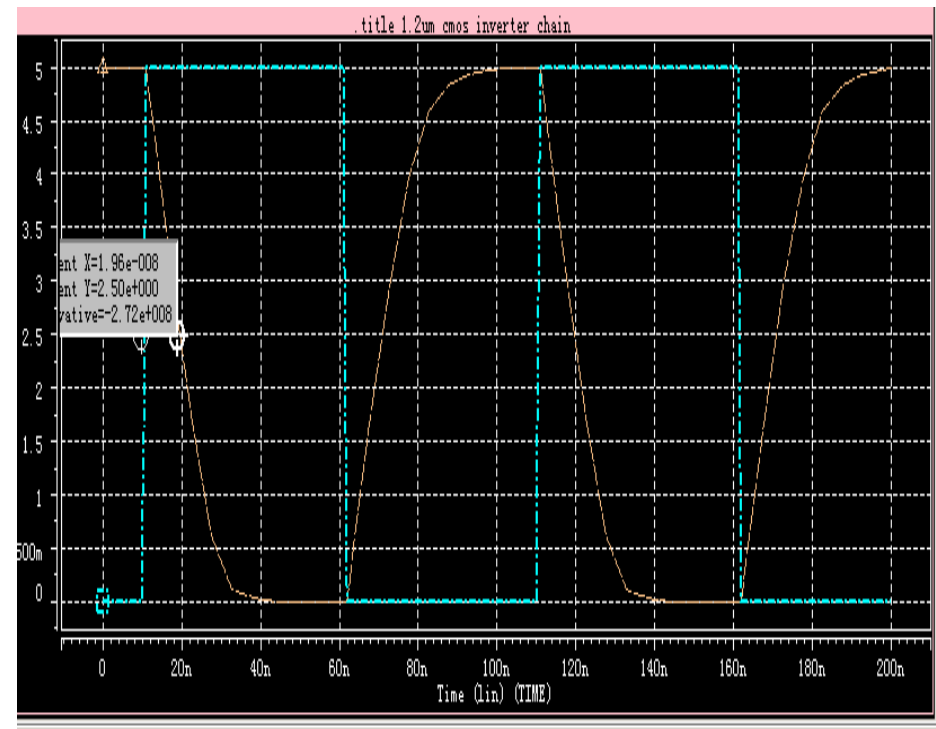
```
VERSION='1999.4'
```

```
.TITLE '.title 1.2um cmos  
inverter chain'
```

```
tdelay temper alter#
```

```
9.121e-09 25.0000 1.0000
```

延迟**9.121ns**





# Hspice的输出

- .MEASURE:

- Find-When模式:

```
.MEASURE <DC|TRAN| AC> result 条件 < RISE=r | LAST >  
+< FALL=f | LAST>< CROSS=c | LAST >
```

条件: WHEN out\_var = val或WHEN out\_var1=out\_var2或 FIND  
out\_var1 WHEN out\_var2=val或FIND out\_var1 WHEN out\_var2 =  
out\_var3

## .measure语句例二

若在控制卡中加如下一句：

.measure tran ttrans when v(out)=4.5v求输出电压降到4.5v的时刻，则输出文件chain.mt0如下：

```
$DATA1 SOURCE='HSPICE' VERSION='1999.4'  
.TITLE '.title 1.2um cmos inverter chain'  
tdelay ttrans temper alter#  
9.121e-09 1.262e-08 25.0000 1.0000
```

## 求元件功耗:

- 一般形式 `.print/plot <dc或tran> P(element) power`  
其中, `power` 关键词用来计算整个电路的功率。  
例 `.print tran P(M1) P(Vin) P(Cload) Power`  
\*这里只计算瞬态分析或直流分析中的瞬时功耗或静态功耗。
- 使用`measure`语句: `.measure tran p_ AVG POWER`  
`from=0n to=100ns`  
.....  
`tdelay p_ temper alter#`  
9.121e-09 2.653e-04 25.0000 1.0000

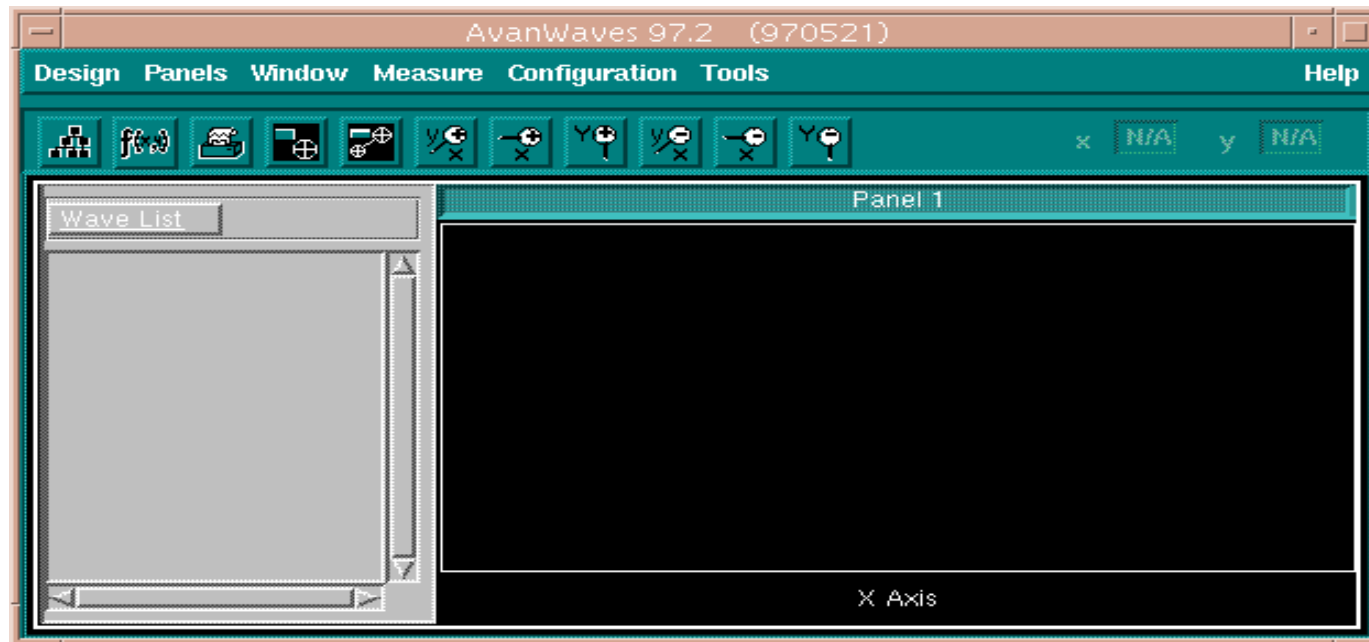
# 使用 MetaWaves 浏览波形

——HSPICE分析结果的  
浏览



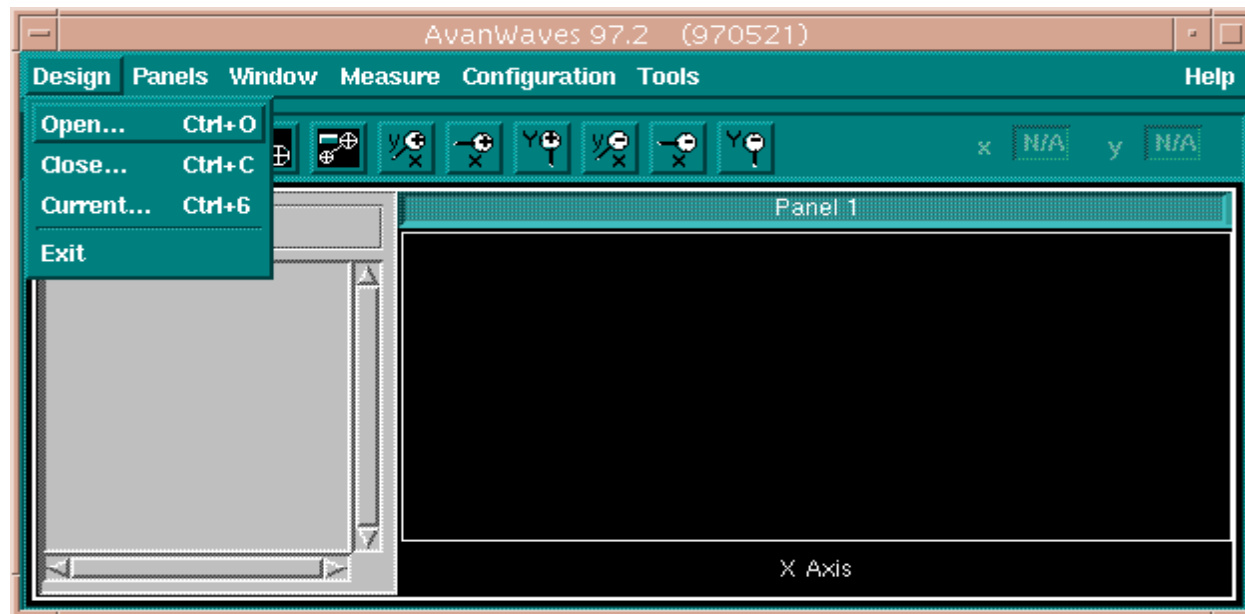
# 使用 MetaWaves 浏览波形

§ 启动MetaWaves: 在UNIX的 `cmdshell` 的提示符下, 键入 `awaves &`, 回车即进入MetaWaves的工作环境。



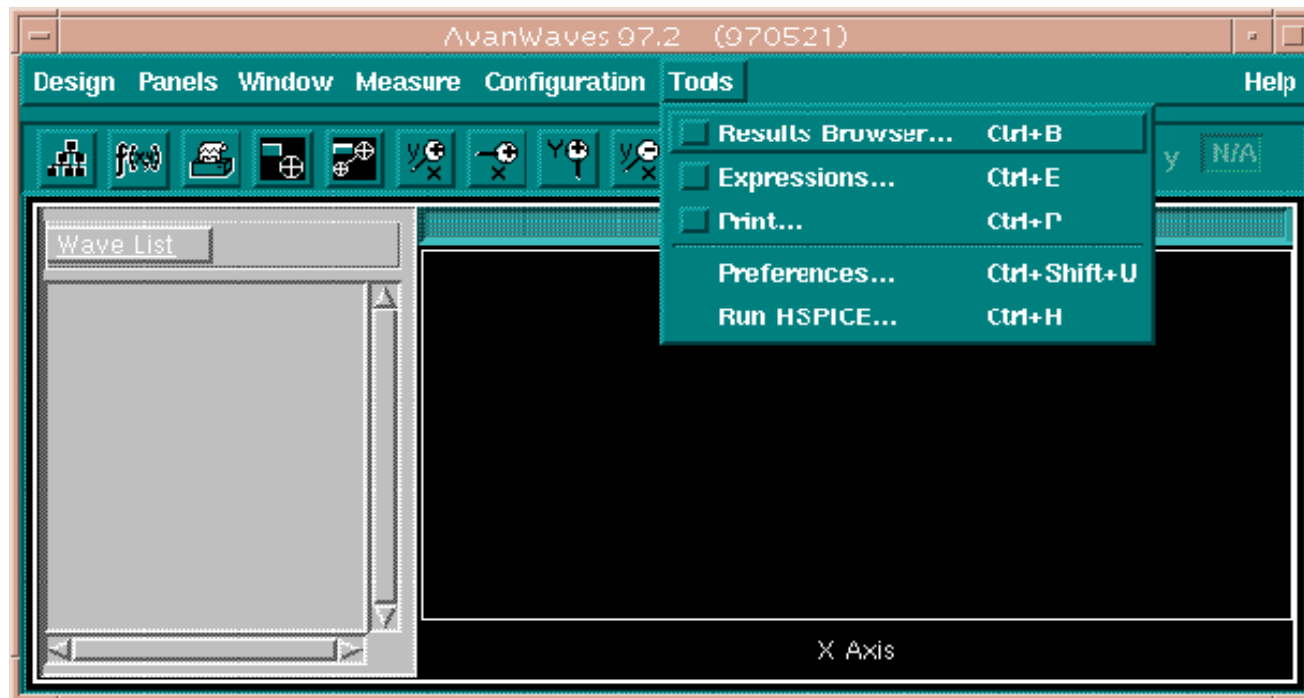
# 使用 MetaWaves 浏览波形

- § 模拟文件的打开与关闭：点开Design 菜单，其中的Open...和Close...命令分别用来打开和关闭待显示波形的模拟文件。



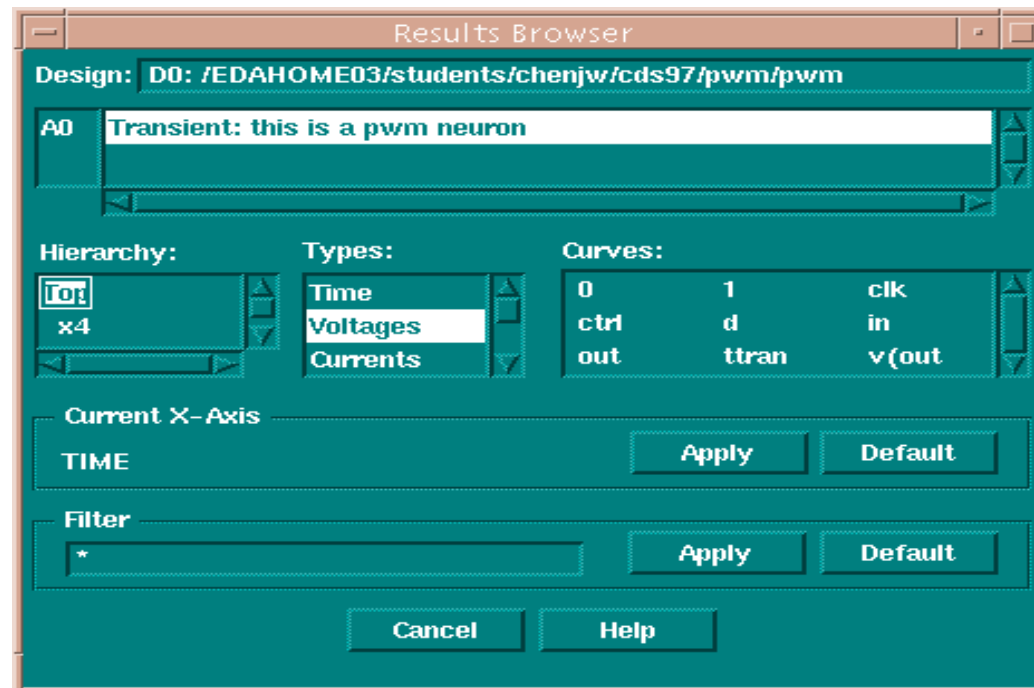
# 使用 MetaWaves 浏览波形

§ Tools窗口：提供了一些常用的工具，如下图所示：



# 使用 MetaWaves 浏览波形

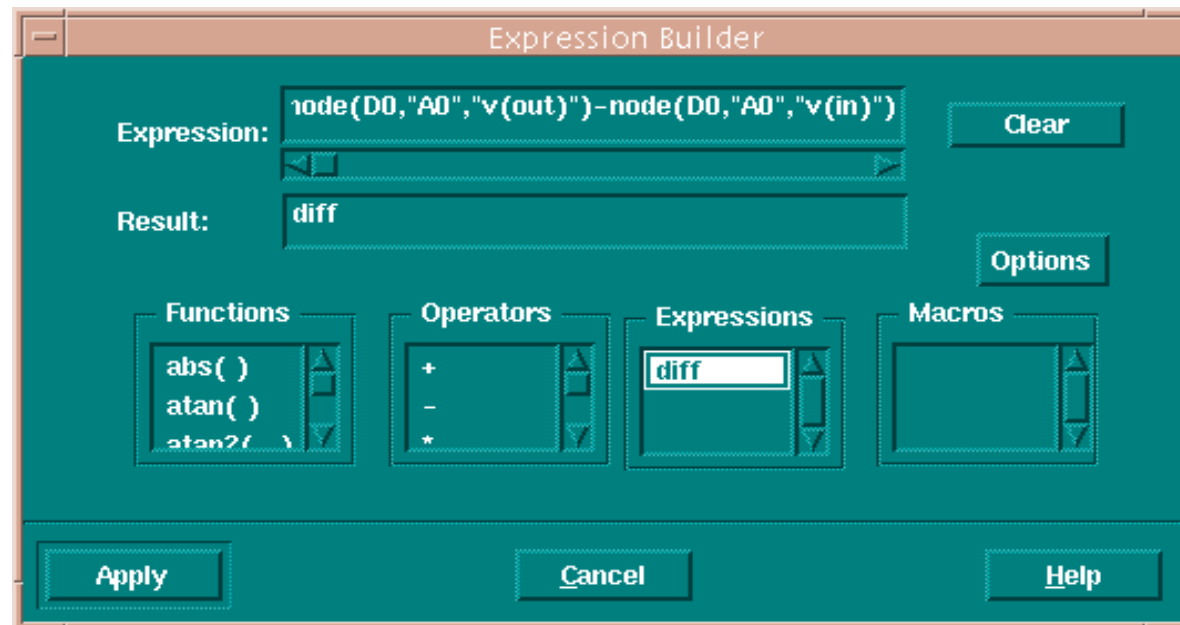
§ 结果编辑窗口：打开一个模拟文件后，就弹出Result Browser窗口用来编辑待显示的结果。可以直接显示的输出列在curves子窗口中。选定一个变量后，直接用鼠标中键拖进主菜单中的波形显示区，即panel中即可。





# 使用 MetaWaves 浏览波形

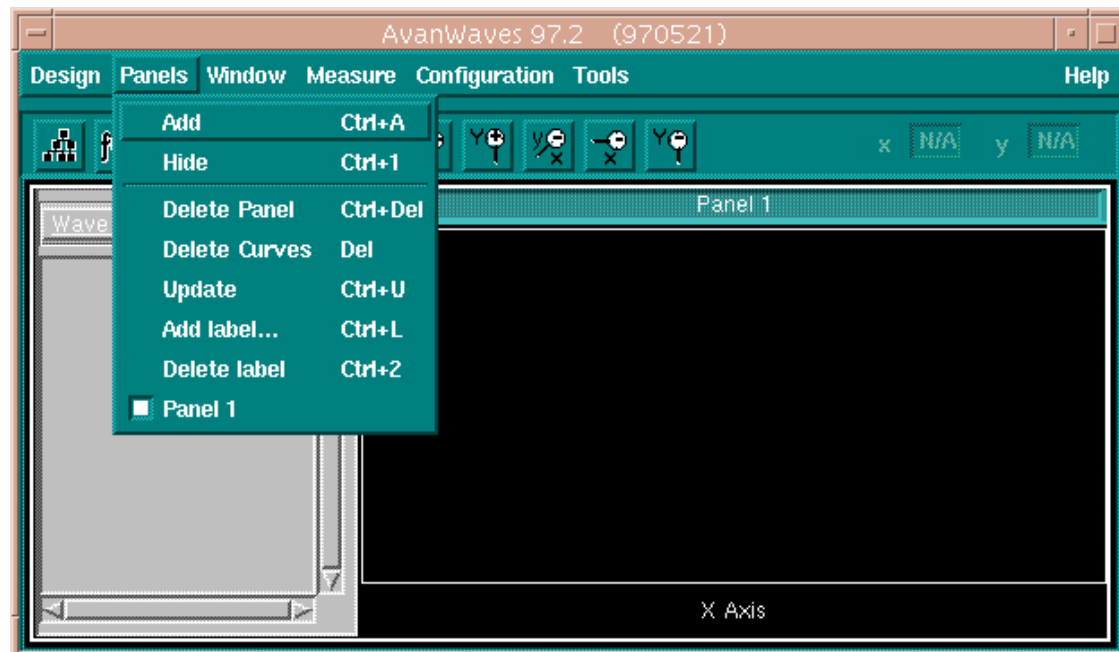
- § 表达式编辑窗口：点开Tools中的Expression Builder，出现该窗口。该窗口提供了常用的函数和运算符，能实现对已有输出波形的函数运算。它的结果也能被显示出来。



# 使用 MetaWaves 浏览波形

§ 波形浏览区编辑菜单Panels: 有时要显示的波形很多, 在一个窗口中放不下, 就需要开多个窗口。该菜提供了窗口的Add, Hide, Delete, Add Label, Delete Label 等操作;

以及单个窗口  
中Delete Curves  
的操作。



# 使用 MetaWaves 浏览波形

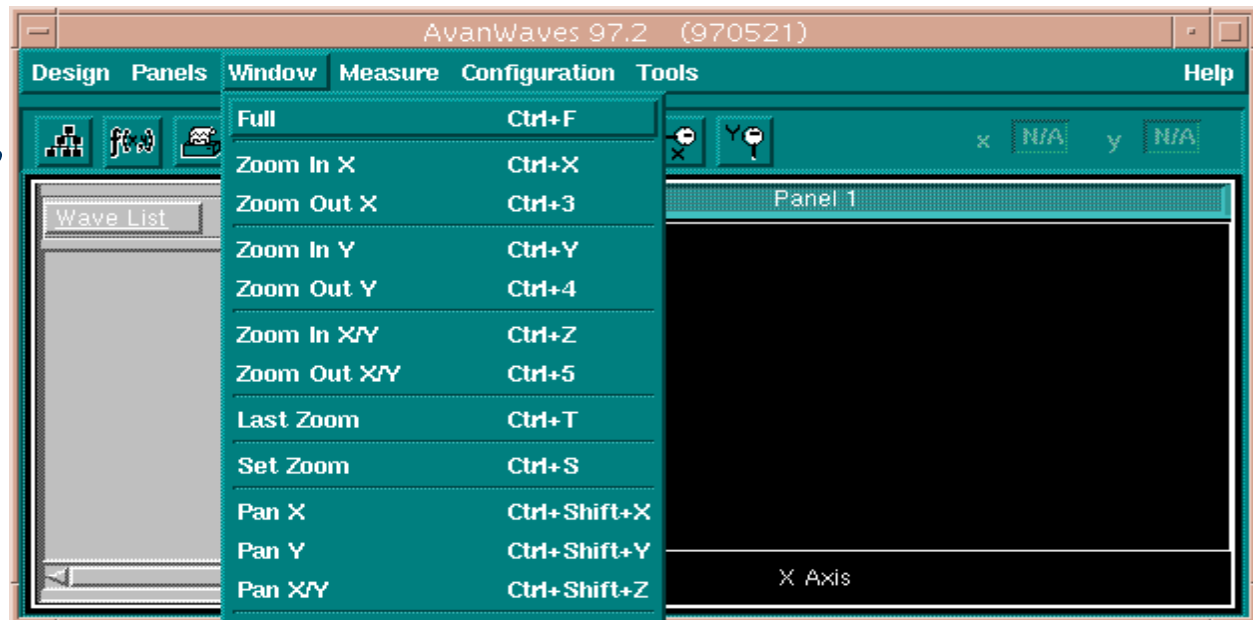
§ 波形显示编辑菜单Windows：对一个Panel中的波形的显示模式进行编辑，包括X轴，Y轴， X轴/Y轴的放大(Zoom In)、缩小(Zoom Out)，移动(Pan)，以及全图显示(Full)，

恢复上一视图

(Last Zoom)，

及视图设置

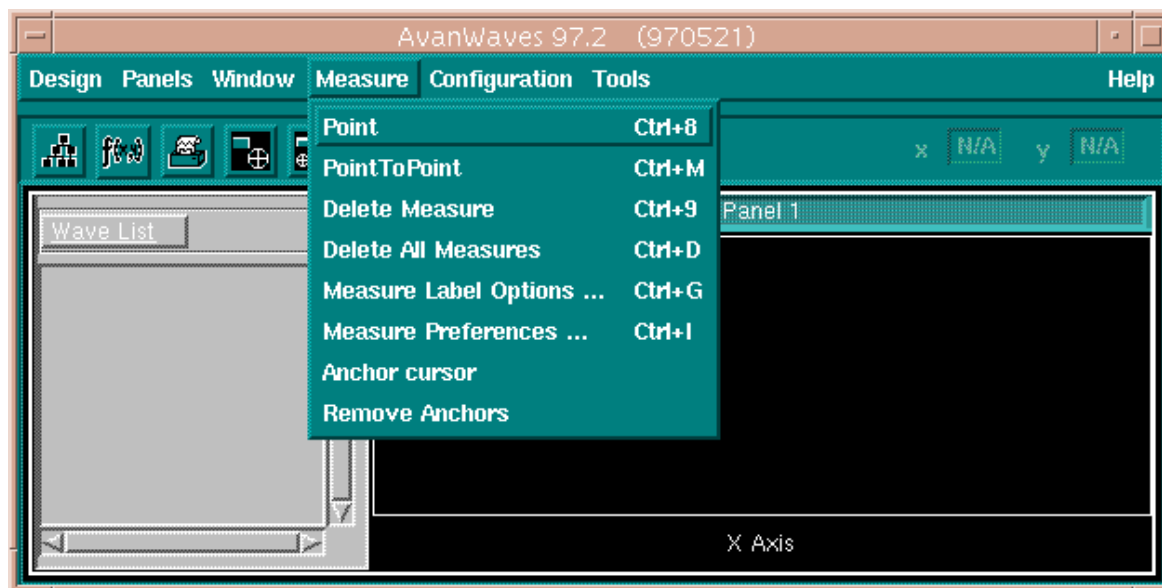
(Set Zoom)。



# 使用 MetaWaves 浏览波形

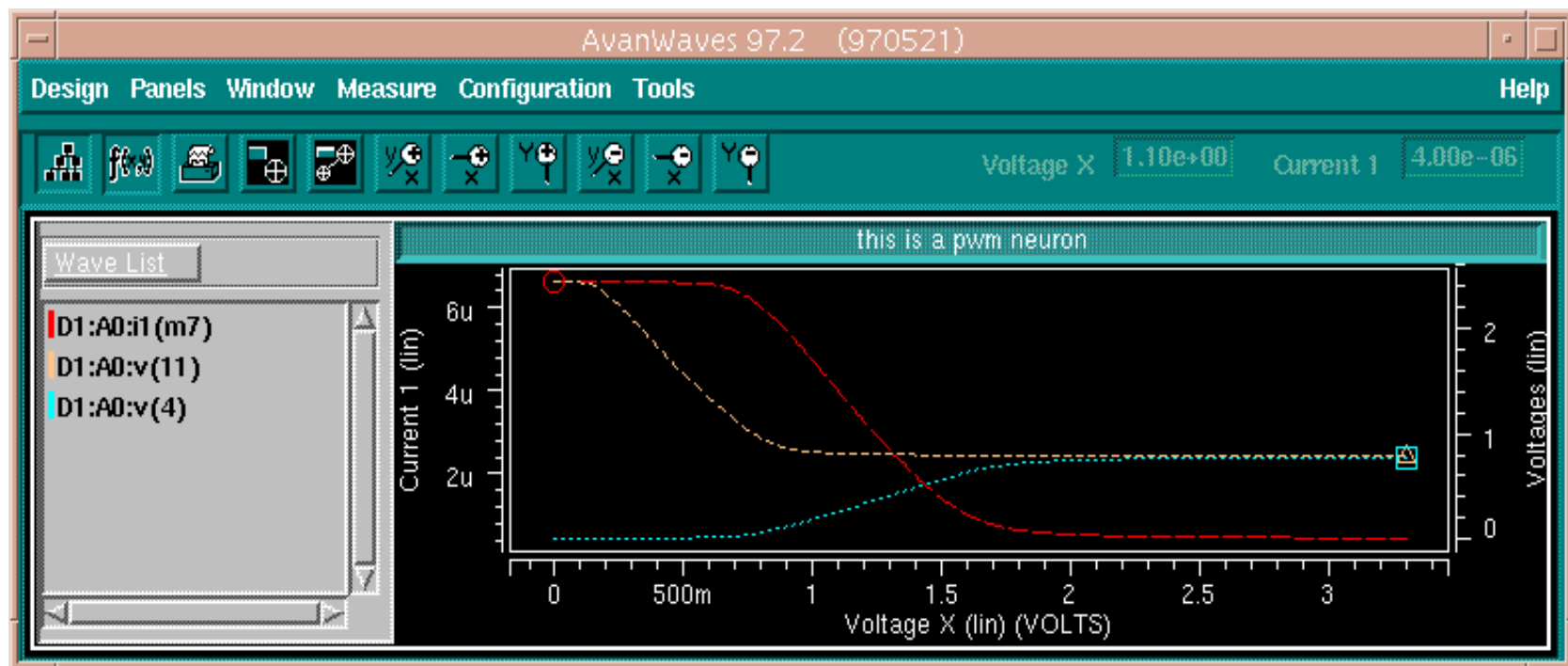
§ 测量菜单Measure: 它提供了对波形进行测量的一些基本工具, 用户可以选择点测量(Point)或点对点的测量(Point To Point), 测量完之后可以清除测量标记>Delete Measure 或>Delete All Measure)。

在Measure Label Options  
或Measure Preference 项  
中调整一些测量的  
参数, 比如  
精度等。



# 使用 MetaWaves 浏览波形

§ 例子:



# 一个完整例子：缓冲驱动器分析

——HSPICE分析举例



# 准备模型文件

- 选用1.2um CMOS工艺level II模型(Models.sp)

```
.MODEL NMOS NMOS LEVEL=2 LD=0.15U TOX=200.0E-10 VTO=0.74 KP=8.0E-05  
+NSUB=5.37E+15 GAMMA=0.54 PHI=0.6 U0=656 UEXP=0.157 UCRIT=31444  
+DELTA=2.34 VMAX=55261 XJ=0.25U LAMBDA=0.037 NFS=1E+12 NEFF=1.001  
+NSS=1E+11 TPG=1.0 RSH=70.00 PB=0.58  
+CGDO=4.3E-10 CGSO=4.3E-10 CJ=0.0003 MJ=0.66 CJSW=8.0E-10 MJSW=0.24
```

```
.MODEL PMOS PMOS LEVEL=2 LD=0.15U TOX=200.0E-10 VTO=-0.74 KP=2.70E-05  
+NSUB=4.33E+15 GAMMA=0.58 PHI=0.6 U0=262 UEXP=0.324 UCRIT=65720  
+DELTA=1.79 VMAX=25694 XJ=0.25U LAMBDA=0.061 NFS=1E+12 NEFF=1.001  
+NSS=1E+11 TPG=-1.0 RSH=121.00 PB=0.64  
+CGDO=4.3E-10 CGSO=4.3E-10 CJ=0.0005 MJ=0.51 CJSW=1.35E-10 MJSW=0.24
```

# 设计基本反相器单元

- 根据模型参数、设计要求设定管子尺寸
- 写出反相器网单：

```
.TITLE 1.2UM CMOS INVERTER CHAIN
```

```
.INCLUDE "models.sp"
```

```
.global vdd
```

```
Mn out in 0 0 NMOS W=1.2u L=1.2u
```

```
Mp out in vdd vdd PMOS W=3u L=1.2u
```

```
CL OUT 0 0.5PF
```

```
VCC VDD 0 5V
```

```
VIN IN 0 PULSE(0 5V 10NS 1N 1N 50N 100N)
```

```
.....
```



# 直流传输特性分析

```
.TITLE 1.2UM CMOS INVERTER CHAIN
.INCLUDE "models.sp"
.global vdd
.option probe
Mn out in 0 0 NMOS W=1.2u L=1.2u
Mp out in vdd vdd PMOS W=1.2u L=1.2u
CL OUT 0 0.5PF
VCC VDD 0 5V
VININ 0 PULSE(0 5V 10NS 1N 1N 50N 100N)

.DC VIN 0 5V 0.1V
.op
.probe v(out)
.end
```

- 在Cmdshell中输入“>hspice chain.sp&”，回车，

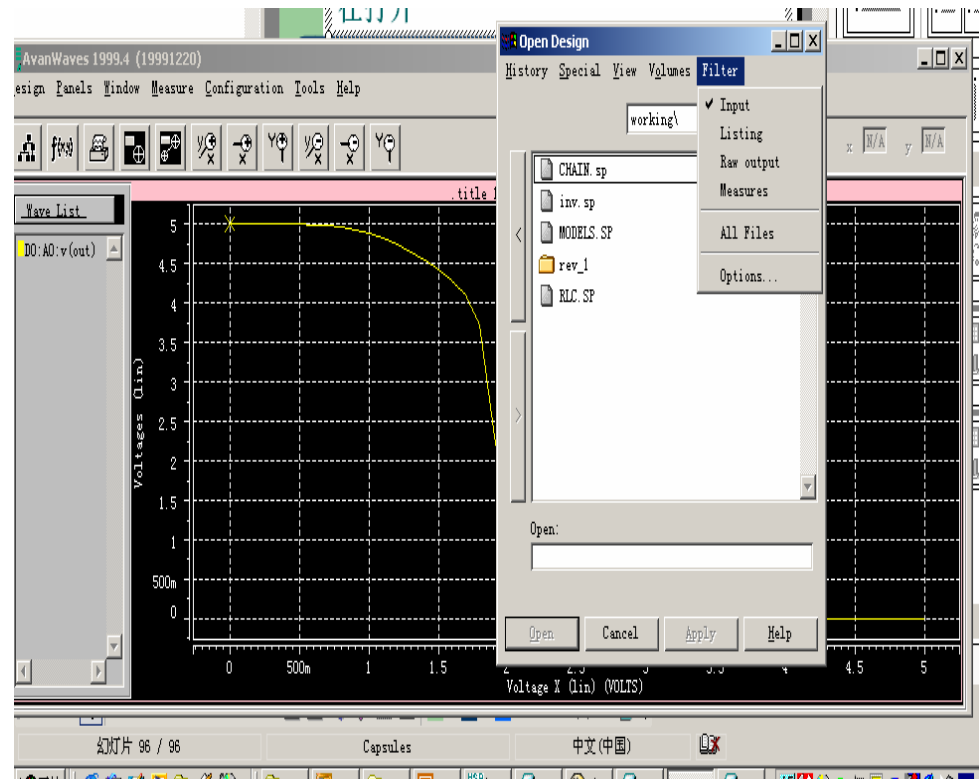
窗口中显示工作信息：lis文件内容，当显示：

```
“      ***** job concluded
1 ***** Star-HSPICE -- 1999.4 (19991220) 21:16:54
   04/21/2002 pcnt      *****
.title 1.2um cmos inverter chain
***** job statistics summary .....
```

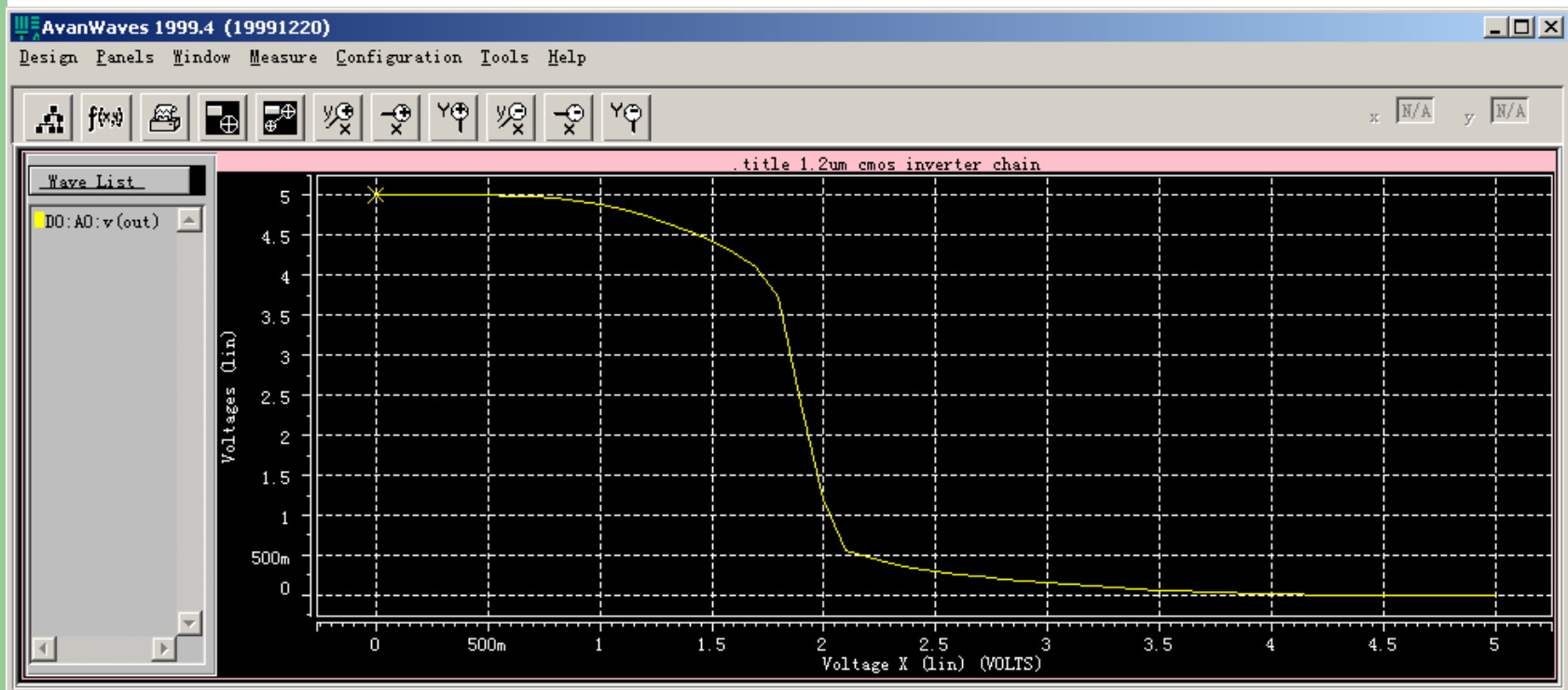
时工作结束且无错误（若是“... job aborted ...”则发生错误，可检查上面信息，会给出错误提示。）

- 再输入“>awaves&”启动metawave。

选择Design->open, 在open design窗口中从Filter菜单中选中input或all, 会在下面文件列表中显示出来, 任选一个设计文件。



# Metawave中输出的直流传输特性曲线



# 利用含参数的子电路组成反相器链

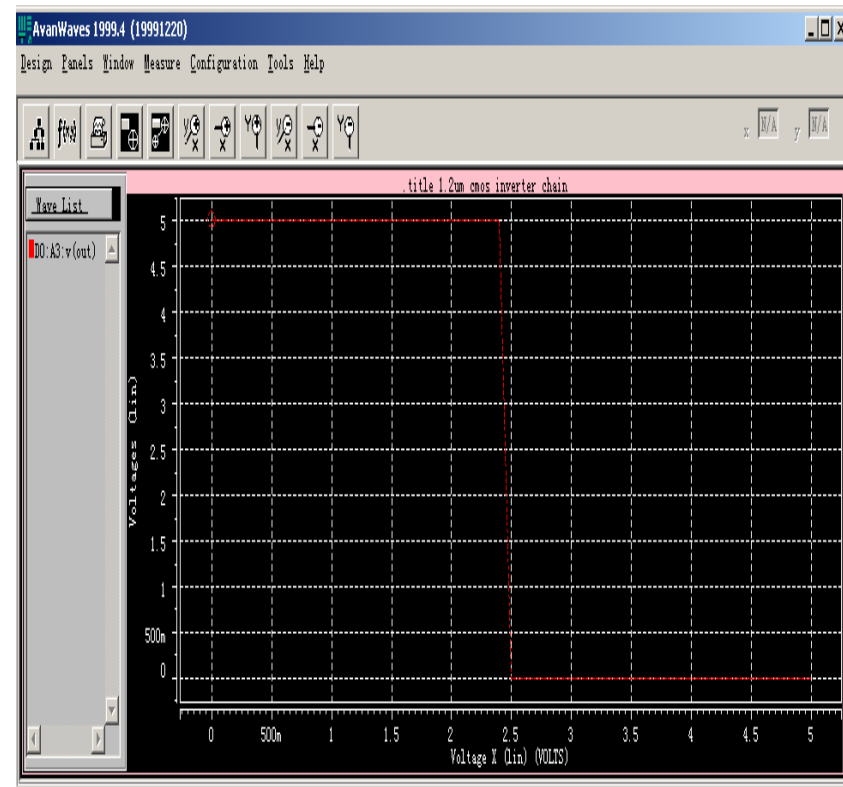
```
.TITLE 1.2UM CMOS INVERTER CHAIN
.INCLUDE "models.sp"
.global vdd
.SUBCKT INV IN OUT wn=1.2u wp=1.2u
Mn out in 0 0 NMOS W=wn L=1.2u
Mp out in vdd vdd PMOS W=wp L=1.2u
.ENDS
```

```
X1 IN 1 INV WN=1.2U WP=3U
X2 1 2 INV WN=1.2U WP=3U
X3 2 OUT INV WN=1.2U WP=3U
CL OUT 0 1PF
VCC VDD 0 5V
VININ 0
.....
```

# 直流特性分析

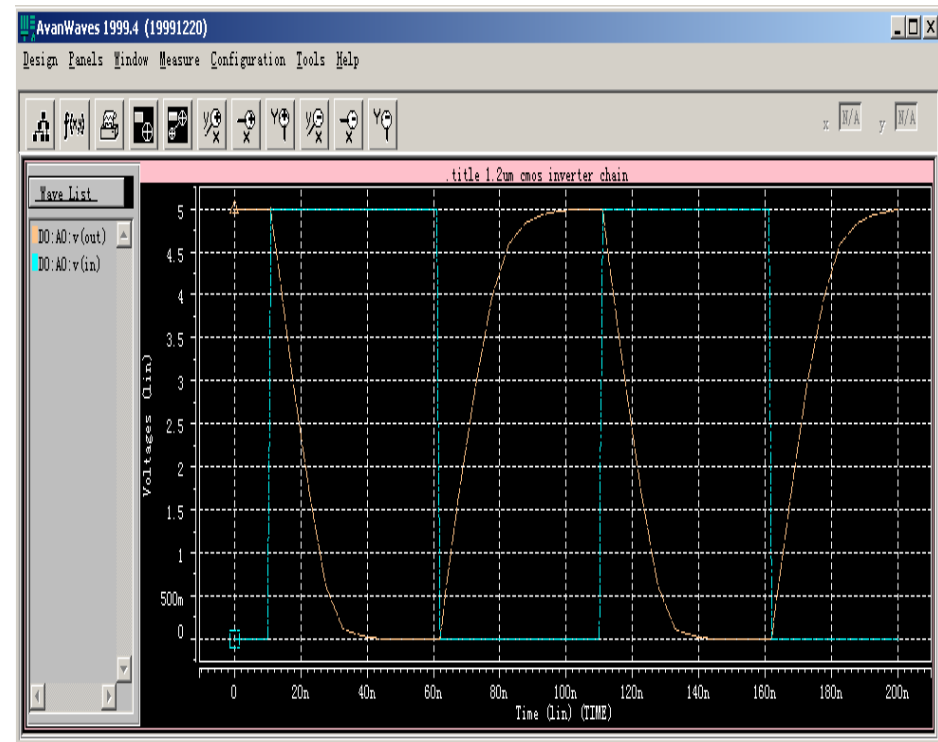
```
.....  
.DC VIN          0 5V 0.1V  
.measure DC tran ttrans when  
    v(out)=2.5v  
  
.....  
.END
```

```
.....  
ttrans temper alter#  
    2.4500  25.0000  1.0000
```



# 时序特性

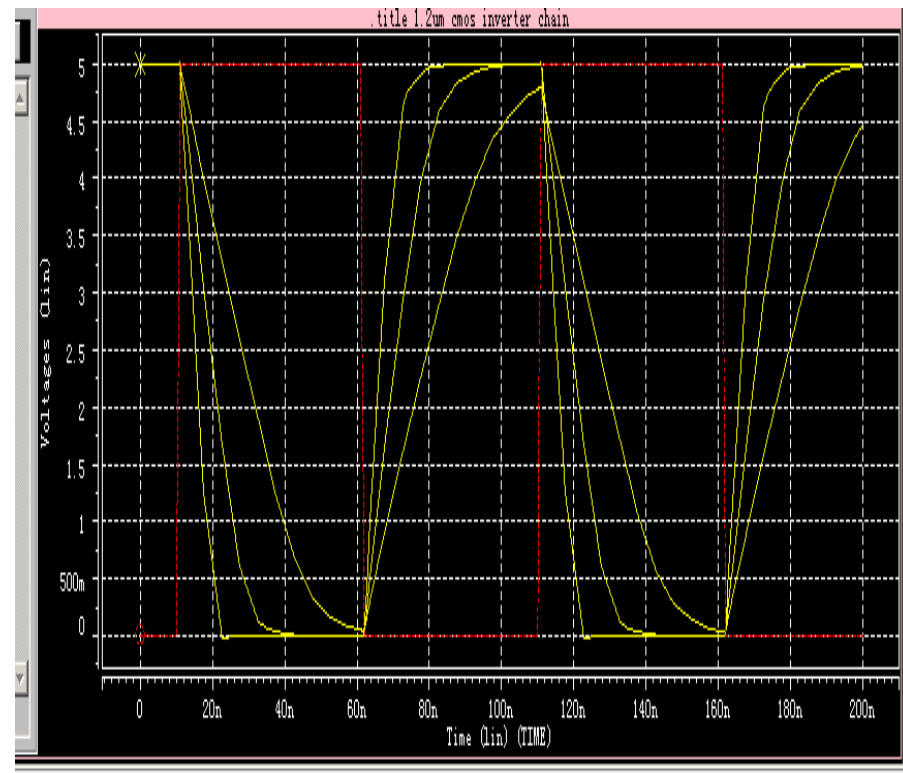
```
.....  
VIN IN 0 PULSE(0 5V 10NS 1N  
+ 1N 50N 100N)  
*.DC VIN 0 5V 0.1V  
.TRAN 1N 200N  
.measure tran tdelay trig v(in)  
val=2.5 td=8ns rise=1  
+ targ v(out) val=2.5 td=9n  
fall=1  
.PRINT V(OUT)  
.end
```



# 考察驱动能力

- 扫描负载电容，观察时序波形：

```
.....  
.param cload=1pf  
.....  
.data cv  
cload  
0.5p  
1p  
2p  
.enddata  
.....  
CL          OUT 0 cload  
VIN IN 0 PULSE(0 5V 10NS 1N  
1N  
+50N 100N)  
.TRAN 1N 200N sweep data=cv  
.....
```





# 固定负载，扫描管子尺寸

```
.....  
.param cload=1.2u  
.param wpt='2.5*cload'  
.....  
.data cv  
cload  
1.2u 2.4u 3u  
.enddata  
X1 IN 1 INV WN=cload WP=wpt  
X2 1 2 INV WN=cload WP=wpt  
X3 2 OUT INV WN=cload WP=wpt  
CL OUT 0 1pf  
.....  
.TRAN 1N 200N sweep data=cv  
.measure tran td trig v(in) val=2.5 td=8ns  
+rise=1 targ v(out) val=2.5 td=9n fall=1  
.END
```

```
$DATA1 SOURCE='HSPICE' VERSION='1999.4'  
.TITLE '.title 1.2um cmos inverter chain'  
index cload td temper alter#  
1.0000 1.200e-06 9.121e-09 25.0000 1.0000  
2.0000 2.400e-06 4.724e-09 25.0000 1.0000  
3.0000 3.000e-06 3.891e-09 25.0000 1.0000
```

# 优化的各级反相器尺寸

```
.....  
X1 IN 1 INV WN=1.2u WP=3u  
X2 1 2 INV WN=2.4u WP=6u  
X3 2 OUT INV WN=4.8u WP=12u  
CL OUT 0 1pf
```

```
.....  
.TRAN 1N 200N $sweep data=cv  
.measure tran tdelay trig v(in) val=2.5 td=8ns  
+rise=1 targ v(out) val=2.5 td=9n fall=1
```

.....  
面积和延迟（他的tdelay）都比前面3个wn全为3u的情况小。

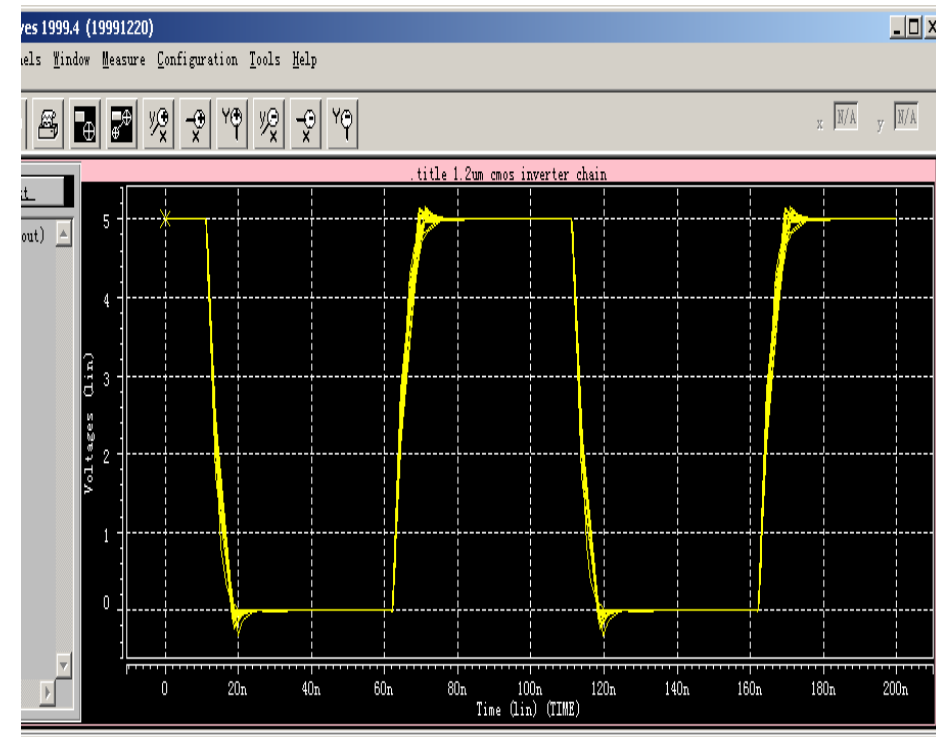
```
$DATA1 SOURCE='HSPICE'  
VERSION='1999.4'  
.TITLE '.title 1.2um cmos inverter  
chain'  
tdelay temper alter#  
3.011e-09 25.0000 1.0000
```

# 温度特性扫描

.....

.TRAN 1N 200N sweep temp -  
+10 125 10 从10 °C到125 °C

.....



# 计算功耗

.....

```
.TRAN 1N 200N  
.measure tran p_ AVG  
+POWER from=0n  
+to=100ns  
.PROBE POWER  
.END
```

