SPICE (A007)

國研院國家晶片系統設計中心

July. 2005

王朝琴　Email: hotline@cic.org.tw

Course Objectives

- Know Basic elements for circuit simulation
- Learn the basic usage of standalone spice simulators
- Know the concept of device models
- Learn the usage of waveform tools
- Advanced features of spice simulator

Further Reading

- **HSPICE 使用手冊**
  PDF格式檔：/usr/meta/cur/docs/pdf/2001.4_hspice.pdf
- **HSPICE版本新增功能說明**
  PDF格式檔：/usr/meta/cur/docs/pdf/2001.4_release_notes.pdf
- **HSPICE - awaves使用手冊**
  PDF格式檔：/usr/meta/cur/docs/pdf/2001.4_avanwaves.pdf
- **SBTSPICE使用手冊**
  PDF格式檔：/usr/sbt/man/manu.pdf
  PS格式檔：/usr/sbt/man/manu.ps
- **SBTPLOT 使用手冊**
  PDF格式檔：/usr/sbt/man/plot.pdf
  PS格式檔：/usr/sbt/man/plot.ps
- **問題諮詢**
  http://www.cic.org.tw

Contents

Class Schedule

Day1

- SPICE Overview
- Graphic Tools
- Lab1 awaves
- Simulation Input and Controls
- Element Syntax
- Lab2 LPF & Schmitt trigger..
- Sources and Stimuli
- Lab3 Diode_sin & MOS_pwl..
- DC Types Analysis

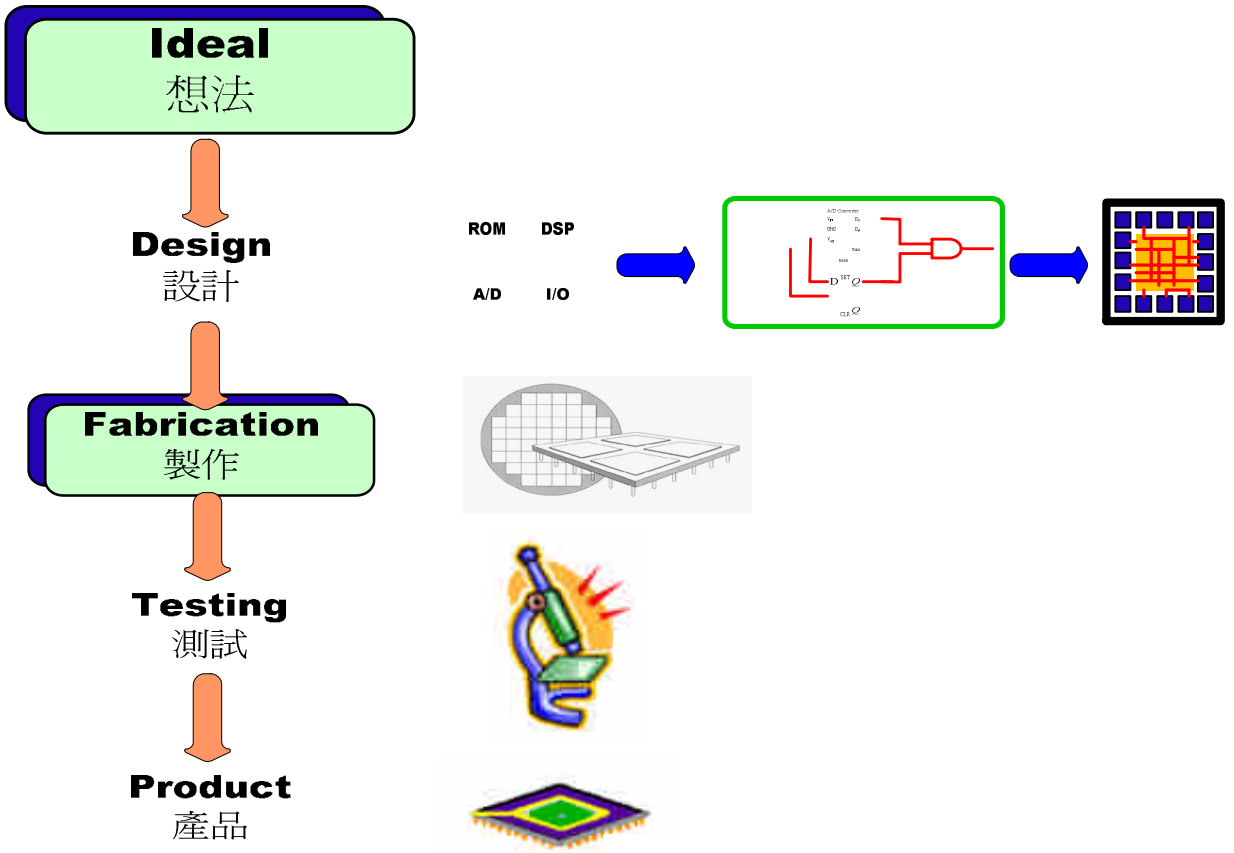
Day2

- TRAN & AC Types Analysis
- Lab 4 Differential Amplifier
- Simulation Output and Controls
- Lab 5 High Q Bandpass Filters …
- Elements and Device Models
- Optimization & Statistical Analysis
- Control Options & Convergence
- Lab 6 RLC circuit test
- Applications Demonstration

Chapter 1

SPICE Overview

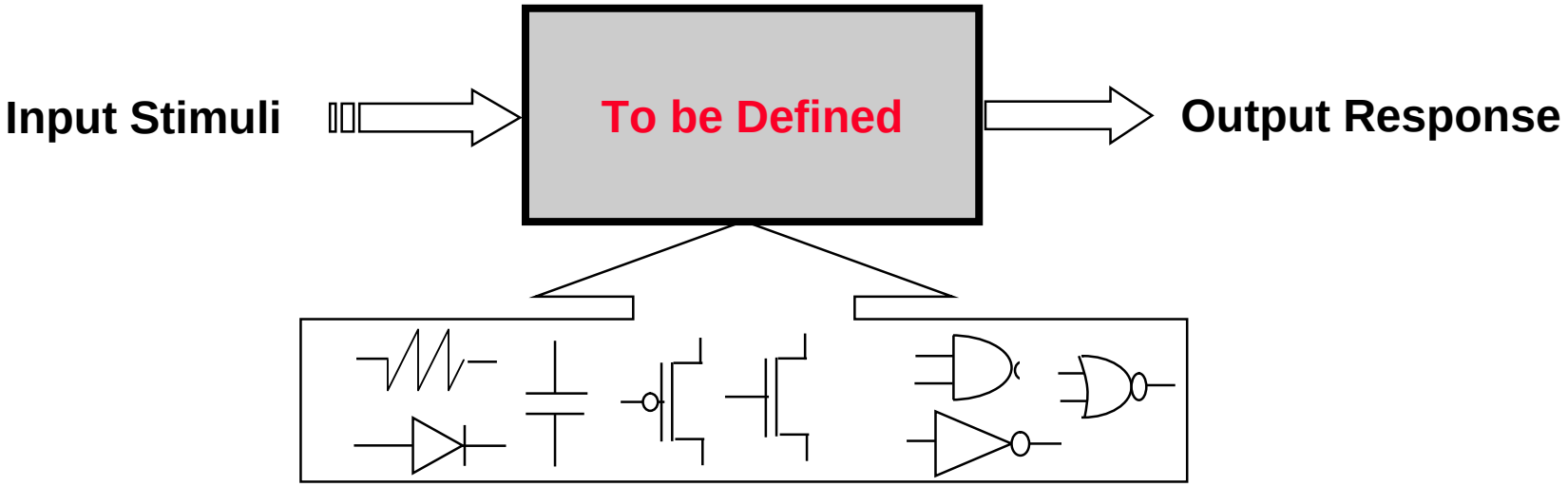
Product Development Process



Circuit Design Background

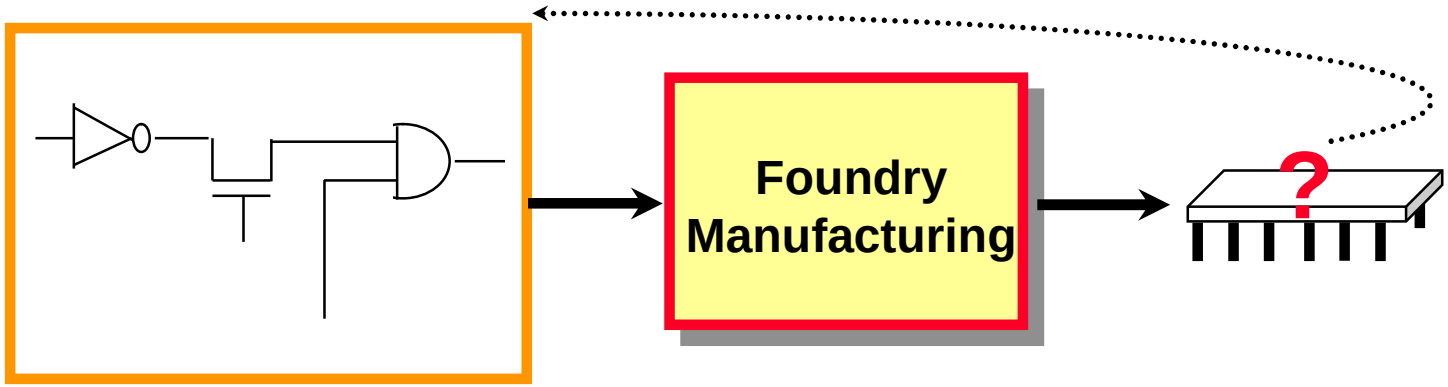
Circuit/System Design :

A procedure to construct a physical structure which is based on a set of basic component, and the constructed structure will provide a desired function at specified time/ time interval under a given working condition.



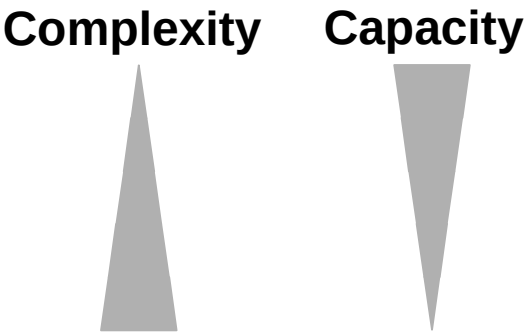
What is Simulation

Simulation : To predict the Circuit/System Characteristic after manufacturing

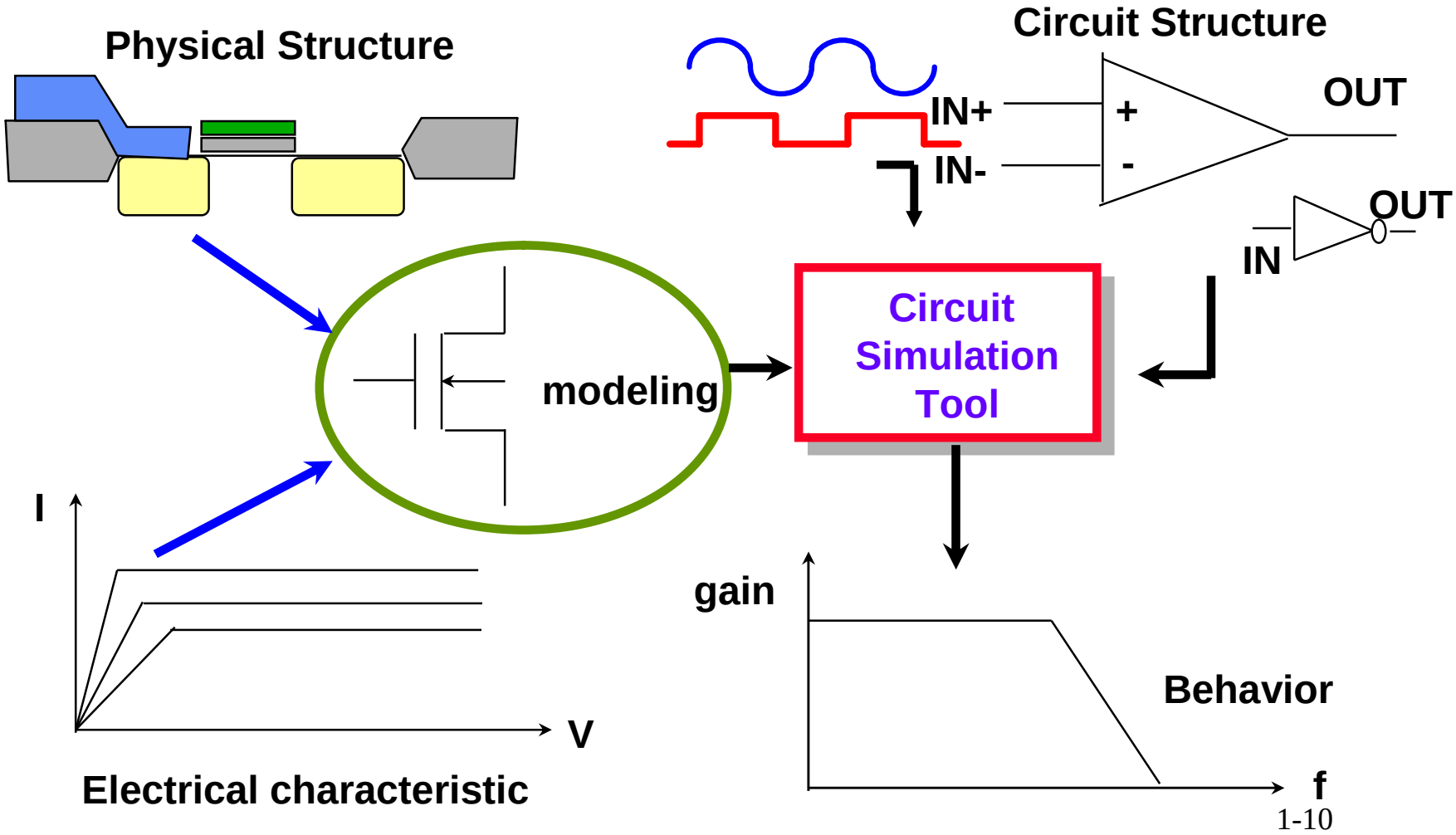


Depends on the component behavior, simulation categories include :

- **Functional simulation**
- **Logic/Gate Level Simulation**
- **Switch/Transistor Level Simulation**
- **Circuit Simulation**
- **Device Simulation**



Circuit Simulation Background



SPICE Background

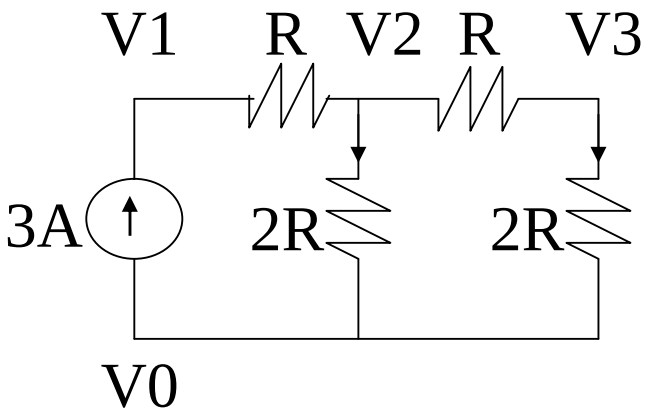
- **SPICE : Simulation Program with Integrated Circuit Emphasis**
  - **Developed by University of California/Berkeley (UCB)**
  - **Successor to Earlier Effort *CANCER***
  - **Widely Adopted, Become *De Facto* Standard**
- **Numerical Approach to Circuit Simulation**
  - **Circuit Node/Connections Define a Matrix**
- **Must Rely on Sub-Models for Behavior of Various Circuit Elements**
  - **Simple (e.g. Resistor)**
  - **Complex (e.g. MOSFET)**

Source : IEEE 1997 CICC Educational Sessions, E3.3

SPICE Introduction

- SPICE generally is a Circuit Analysis tool for Simulation of Electrical Circuits in Steady-State, Transient, and Frequency Domains.
- There are lots of SPICE tools available over the market, SBTSPICE, HSPICE, Spectre, TSPICE, Pspice, Smartspice, ISpice...
- Most of the SPICE tools are originated from Berkeley's SPICE program, therefore support common original SPICE syntax
- Basic algorithm scheme of SPICE tools are similar, however the control of time step, equation solver and convergence control might be different.

Solution for Linear Network



R=5ohm

$$\begin{pmatrix} 0.2 & 0 & -0.1 & -0.1 \\ 0 & 0.2 & -0.2 & 0 \\ -0.1 & -0.2 & 0.5 & -0.2 \\ 0 & 0 & -0.2 & 0.2 \end{pmatrix} \begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} = \begin{pmatrix} -3 \\ 3 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 0.2 & -0.2 & 0 \\ -0.2 & 0.5 & -0.2 \\ 0 & -0.2 & 0.2 \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \\ 0 \end{pmatrix}$$

V_0 ground

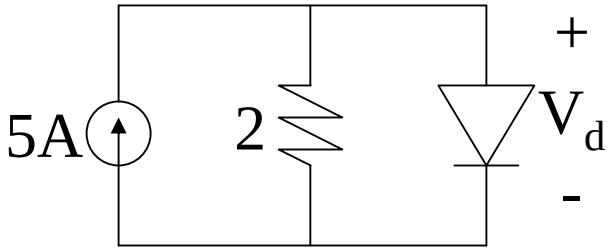
With Gaussian elimination

$$\begin{pmatrix} 0.2 & -0.2 & 0 \\ 0 & 0.3 & -0.2 \\ 0 & 0 & 0.25 \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 3 \\ 3 \end{pmatrix}$$

Results : $V_3 = 12V$, $V_2 = 18V$, $V_1 = 33V$

Iteration and approximation - How solution is obtained

Newton-Raphson formula



$$I_d = 1pA*[\exp(40*V_d)-1]$$

$$5 = V_d/2 + I_d$$

$$5 = V_d/2 + 1pA*[\exp(40*V_d)-1]$$

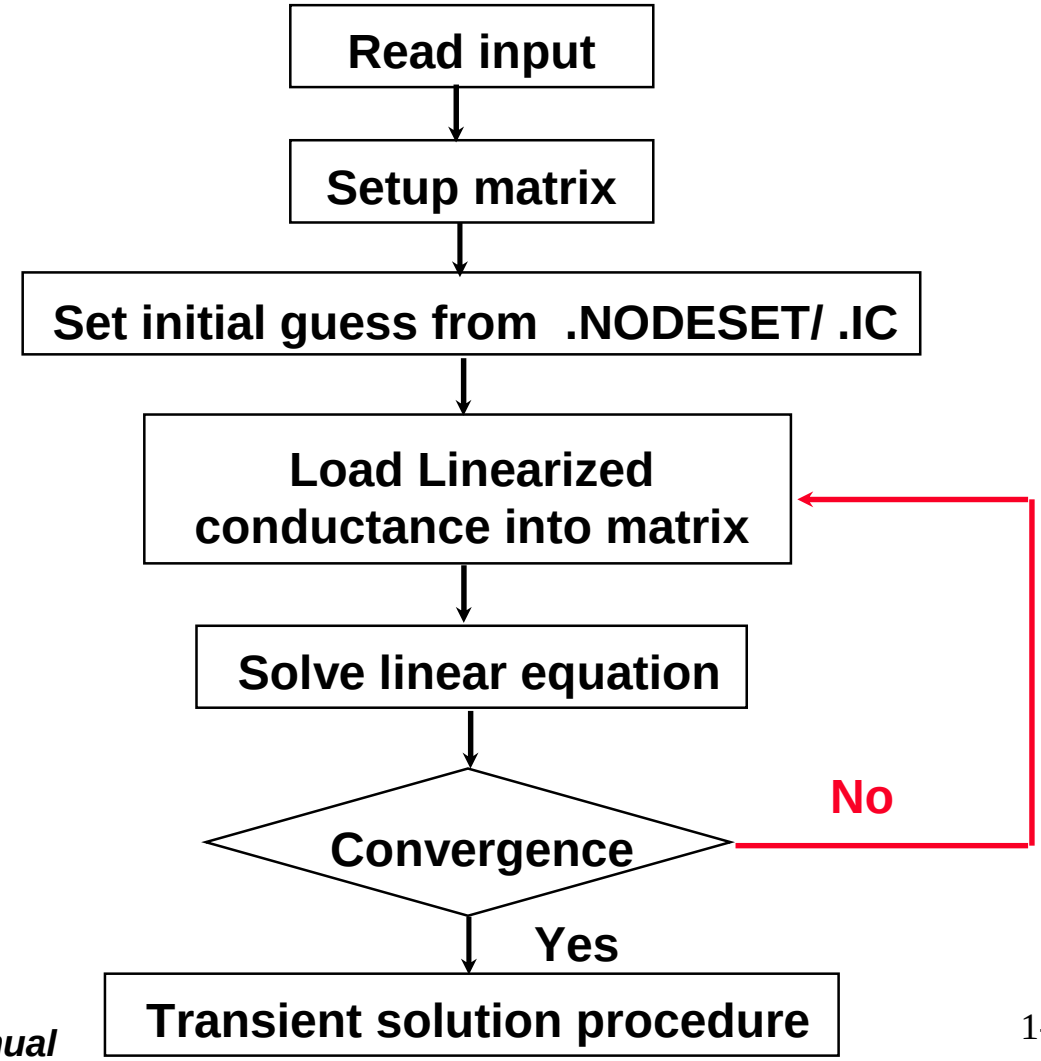
$$F(V_d) = 0 = -5 + \frac{V_d}{2} + 1pA*[\exp(40*v_d) - 1]$$

$$V_{d+1} = V_d - F(V_d)/F'(V_d)$$

	V_d	V_{d+1}	Delta V
1	1	0.975001	0.02499
2	0.975001	0.950002	0.02499
3	0.950002	0.925005	0.02499
4	0.925005	0.900015	0.02499
5	0.900015	0.875041	0.02497
6	0.875041	0.850113	0.02493
7	0.850117	0.825309	0.02481
8	0.825309	0.800838	0.02447
9	0.800838	0.777250	0.02359
10	0.777250	0.755885	0.02136
11	0.755885	0.739445	0.01644
12	0.739447	0.730983	0.00846
13	0.730983	0.729186	0.00179
14	0.729186	**0.729119**	**0.00007**

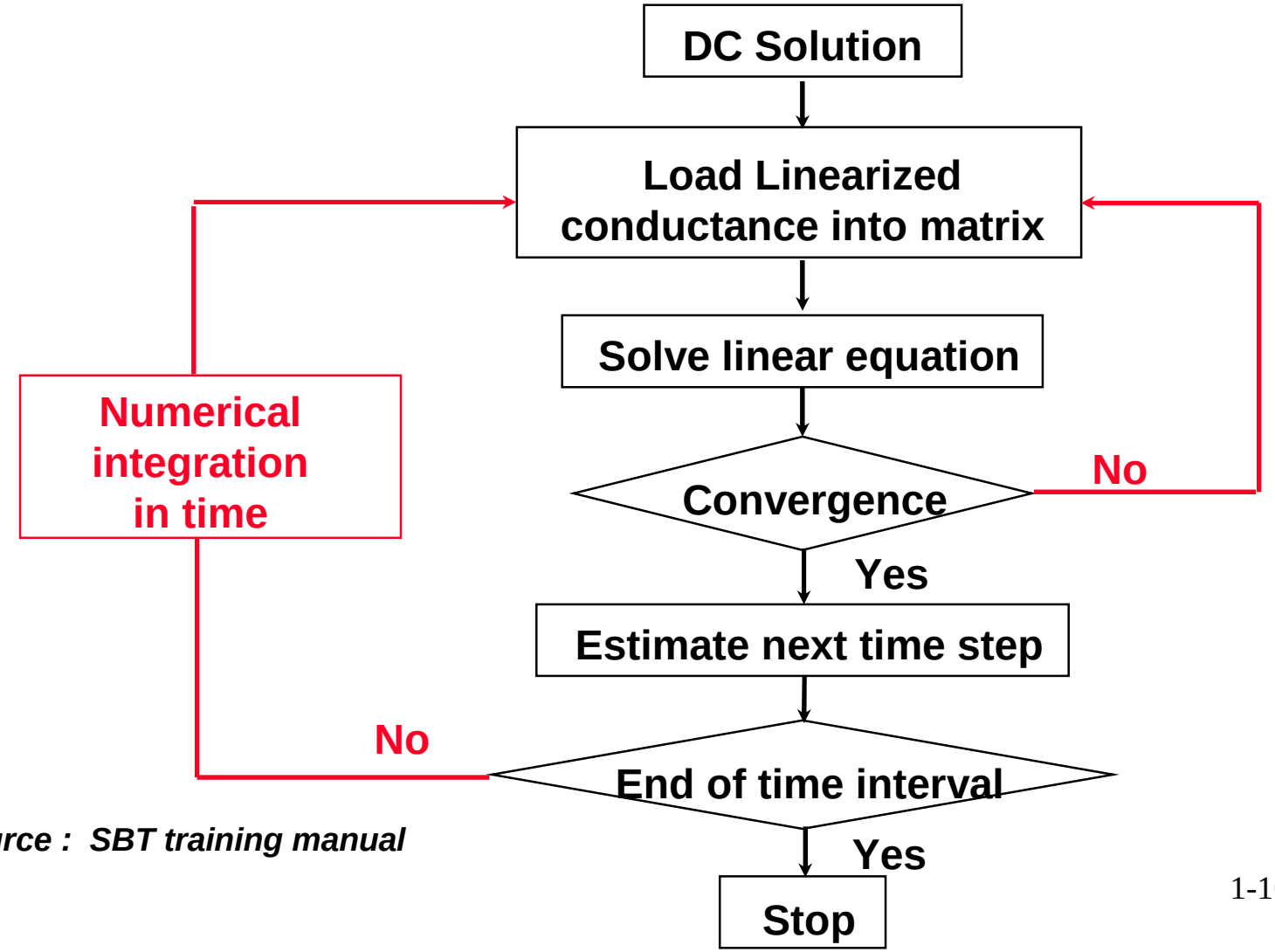
Convergence criteria : Delta V(V_{d+1}-V_d) <0.001

SPICE Simulation Algorithm - DC



Source : SBT training manual

SPICE Simulation Algorithm – Transient



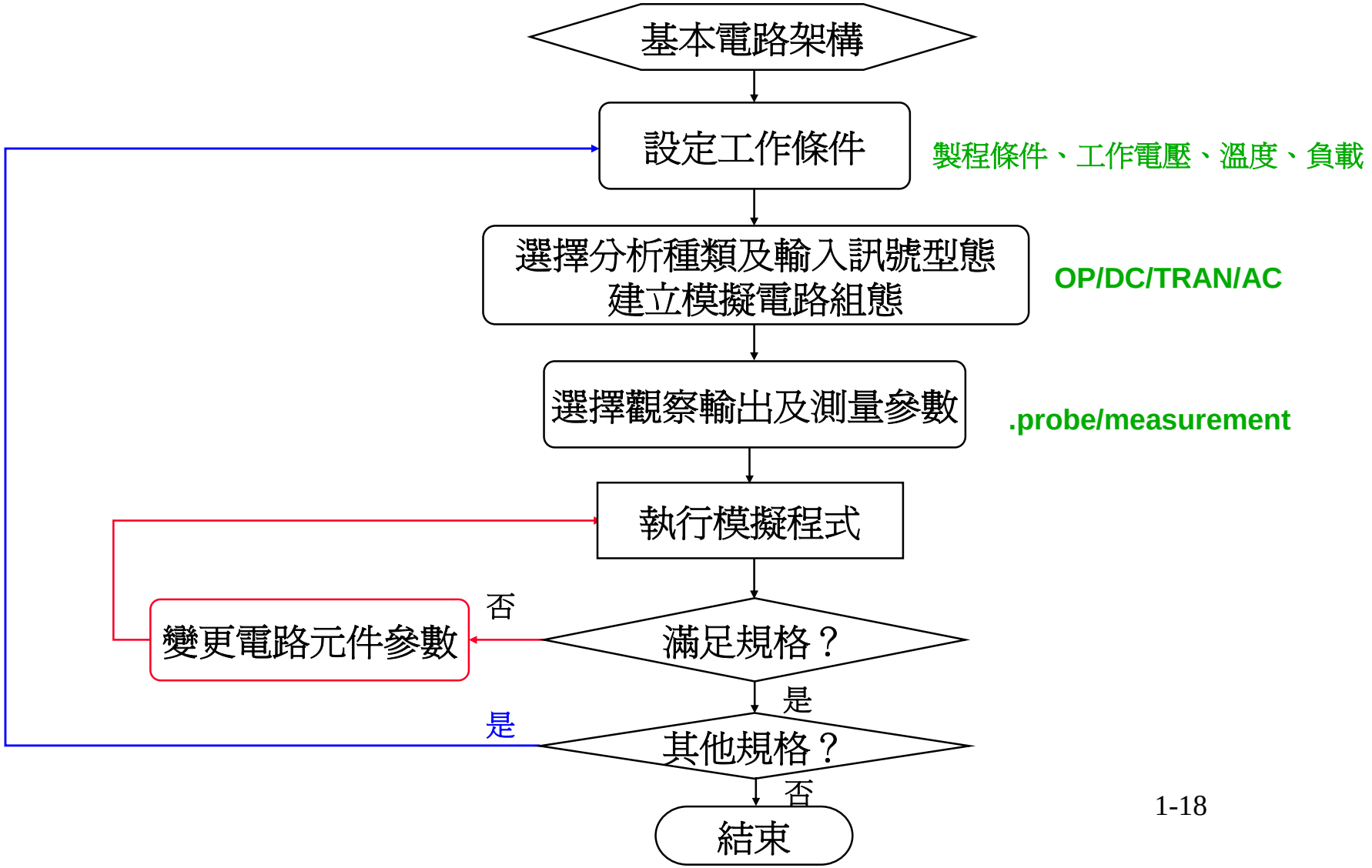
Source : SBT training manual

Basics for Using SPICE Tools

SPICE 之外所需的基本概念

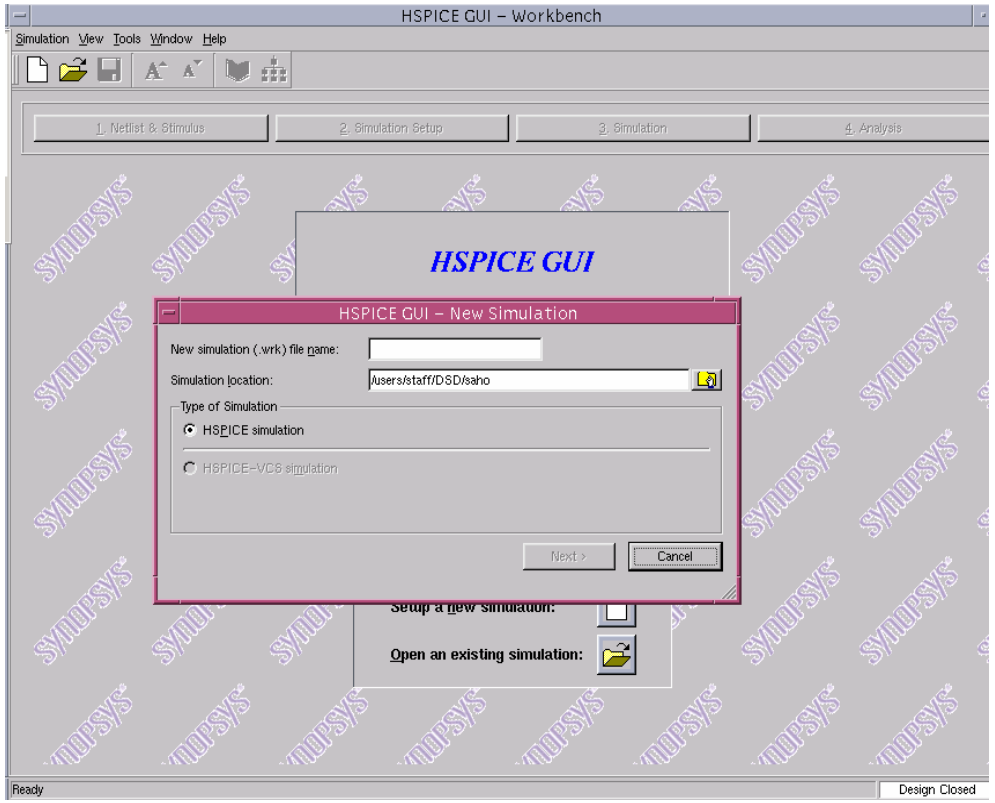
- 了解元件的基本特性
- 熟悉所設計電路的功能
- 了解需要驗證的電路規格及對應的模擬種類及電路組態
- 了解電路的輸入信號特性
- 了解電路各項規格的相依性及優先程度
- 了解電路元件參數與架構對各項電路特性的相關性，以利模擬結果的改進

Basic Flow for SPICE



HSPICE GUI

- **% hspicegui**



Chapter 2

Graphic Tools

HSPICE 模擬波形觀測介面-awaves簡介

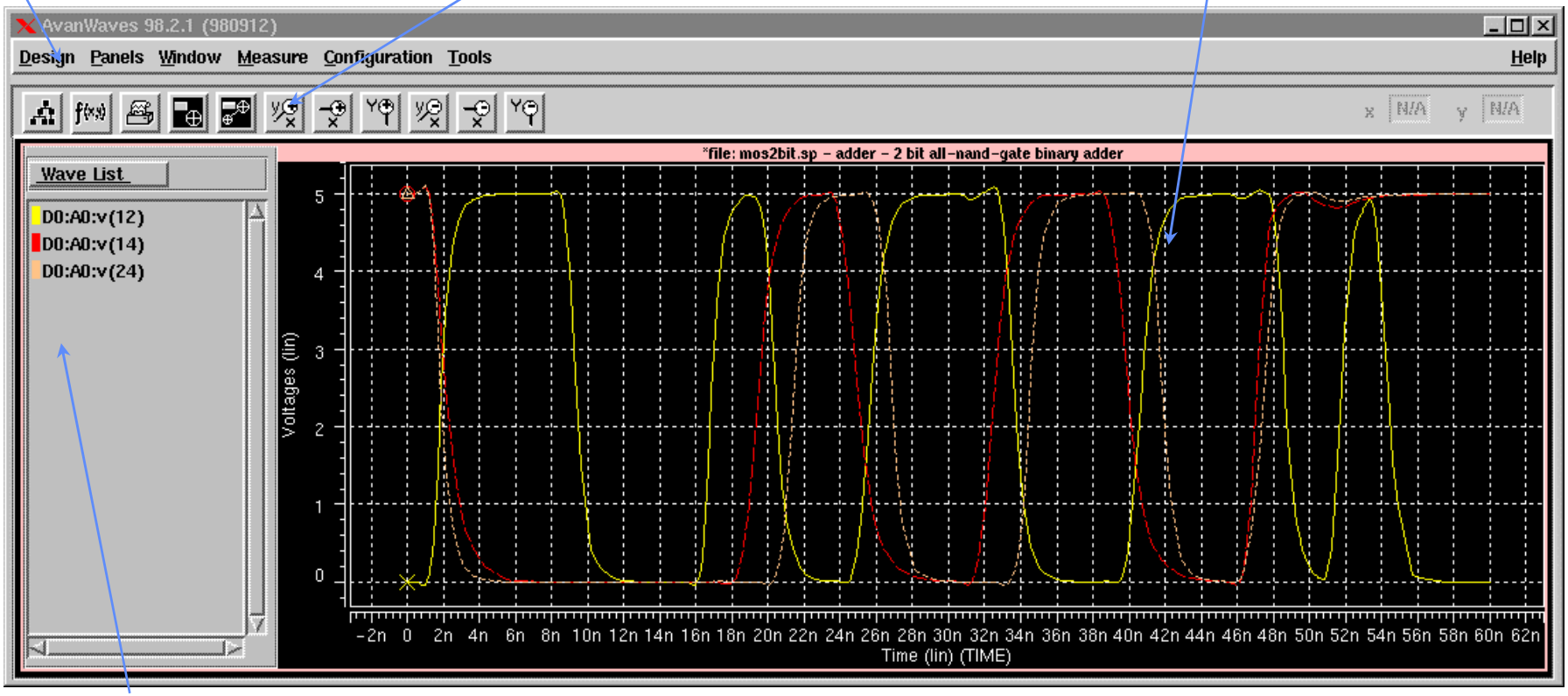
- awaves係視窗程式，需在Xwindow環境下使用。
- 在輸入netlist檔中需指定graphic輸出，hspice才會儲存waveform資料，即加入 .option post 的指令項
- 執行awaves 之指令 /usr/meta/cur/bin/awaves
- awaves可以顯示.sw# ，.ac# ，.tr# ，.mt# ，.ma#, .ms# ，.ft# 等模擬結果

Awaves - Awaves Window

menu

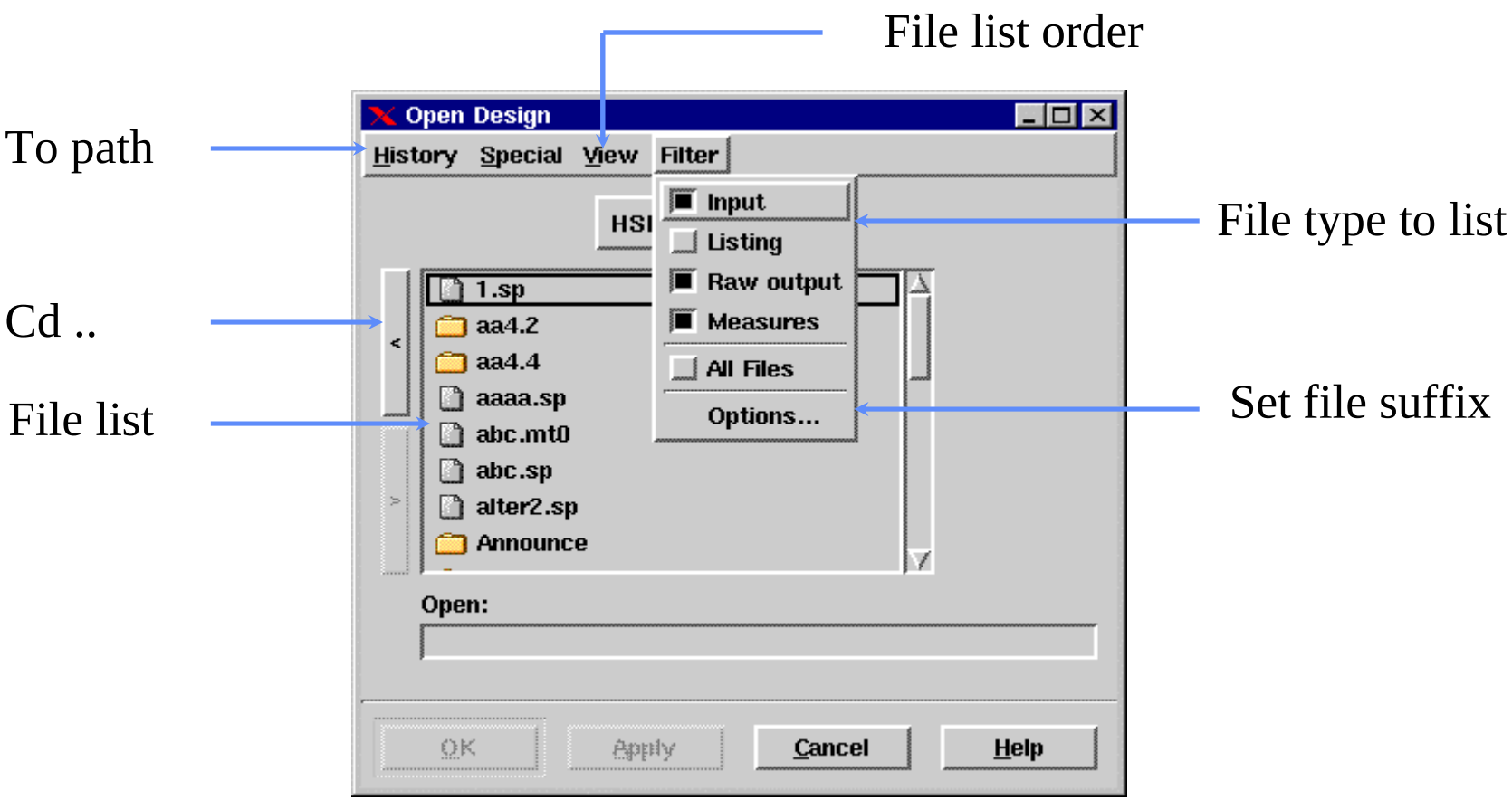
Tool button

waveform

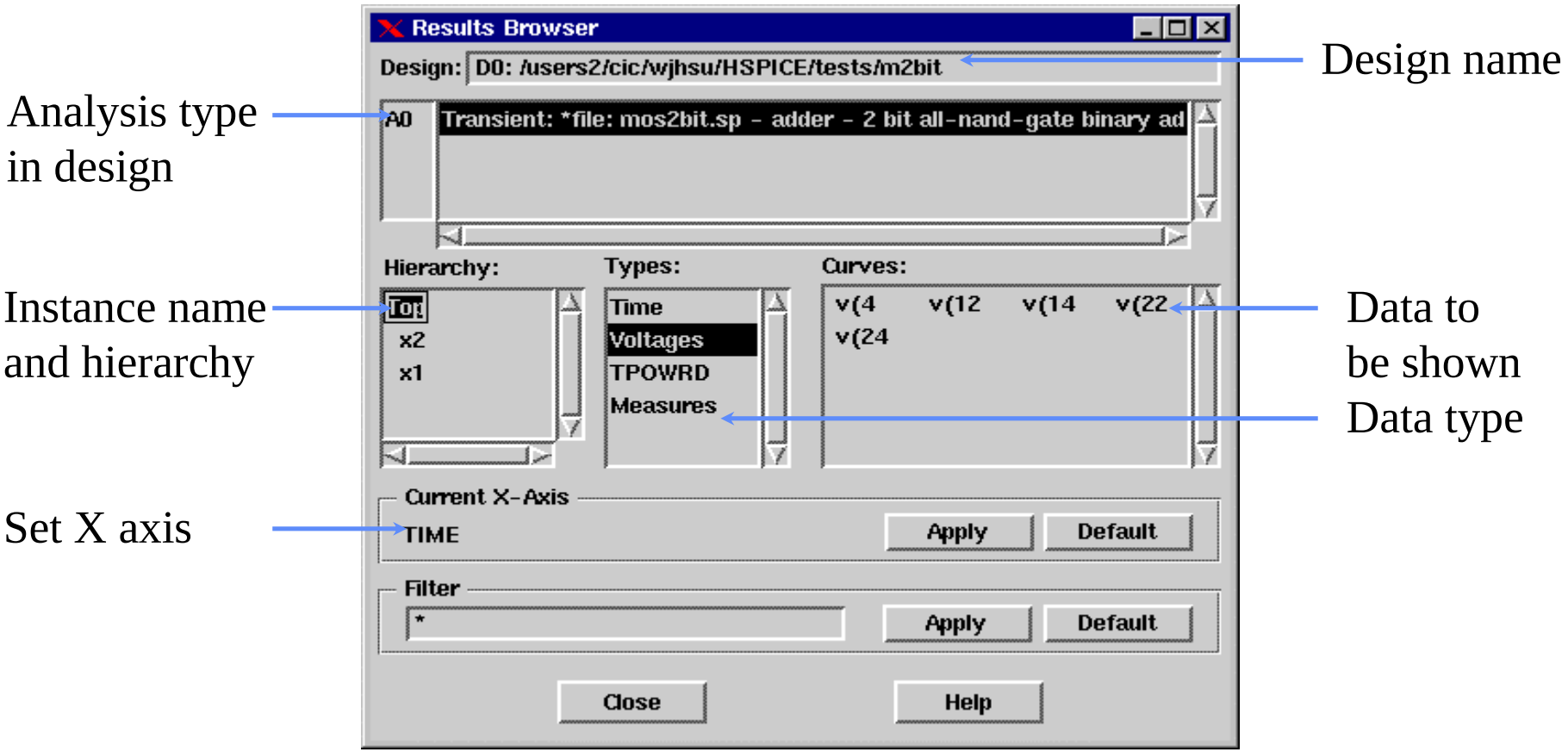


Data names

Awaves- Open Design

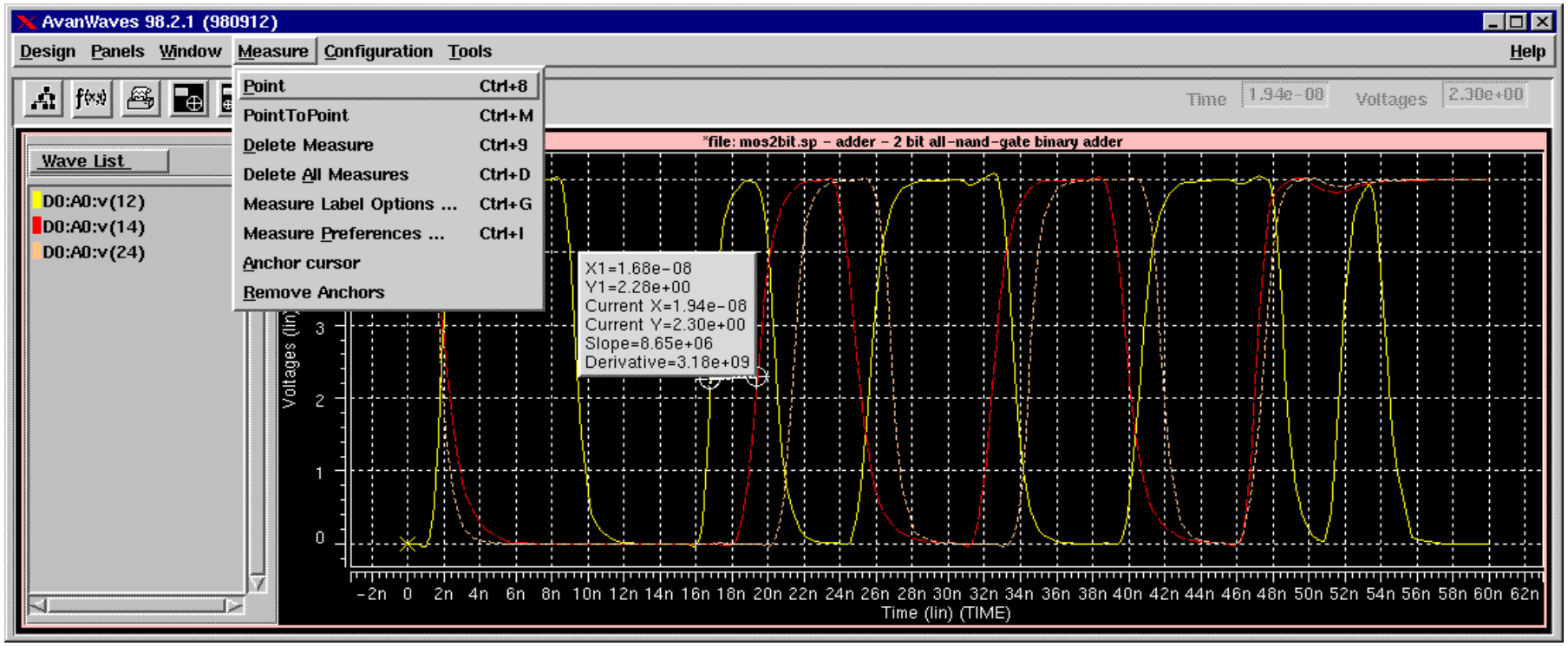


Awaves- Result Browser

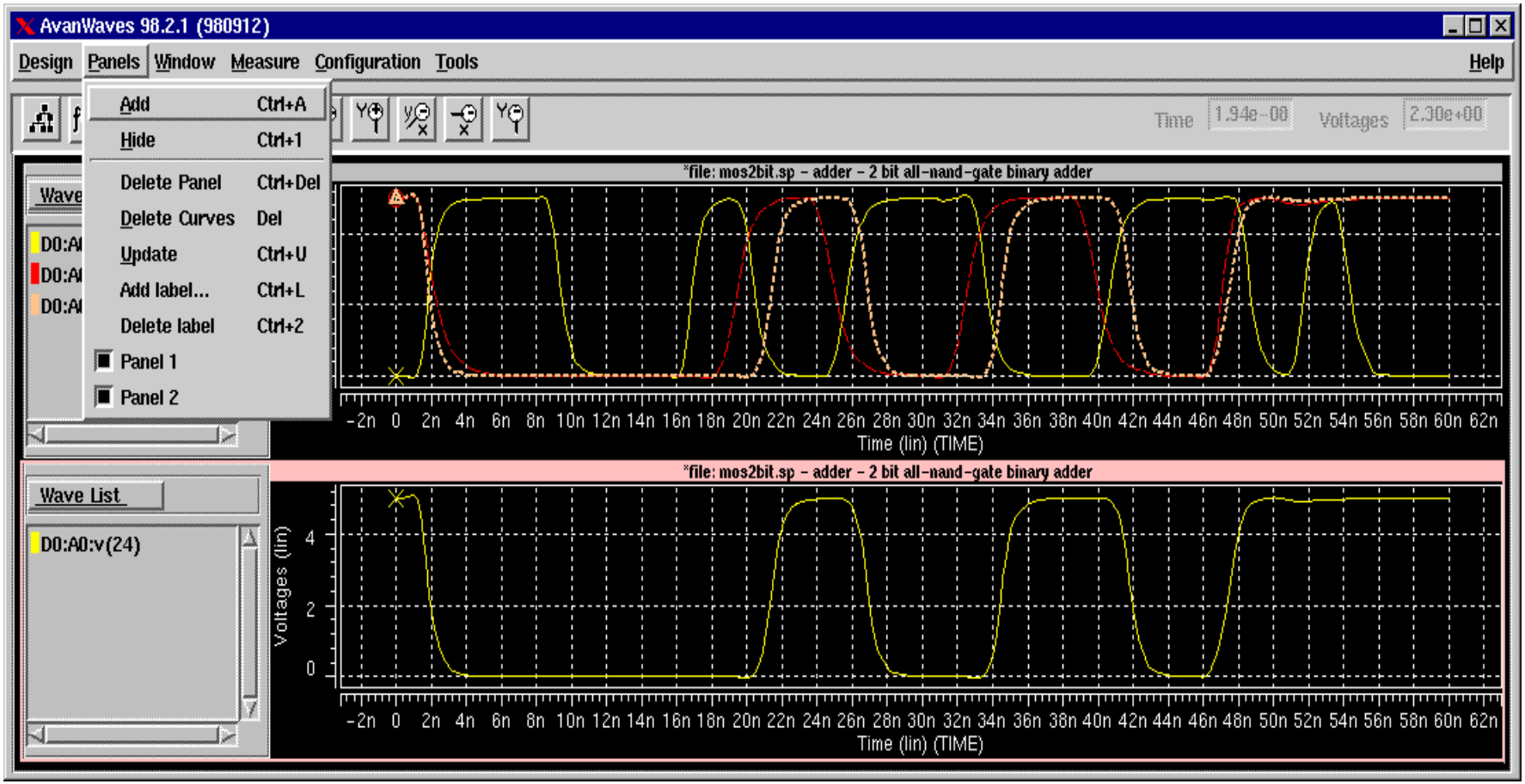


Awaves - Measurement

On window measurement function

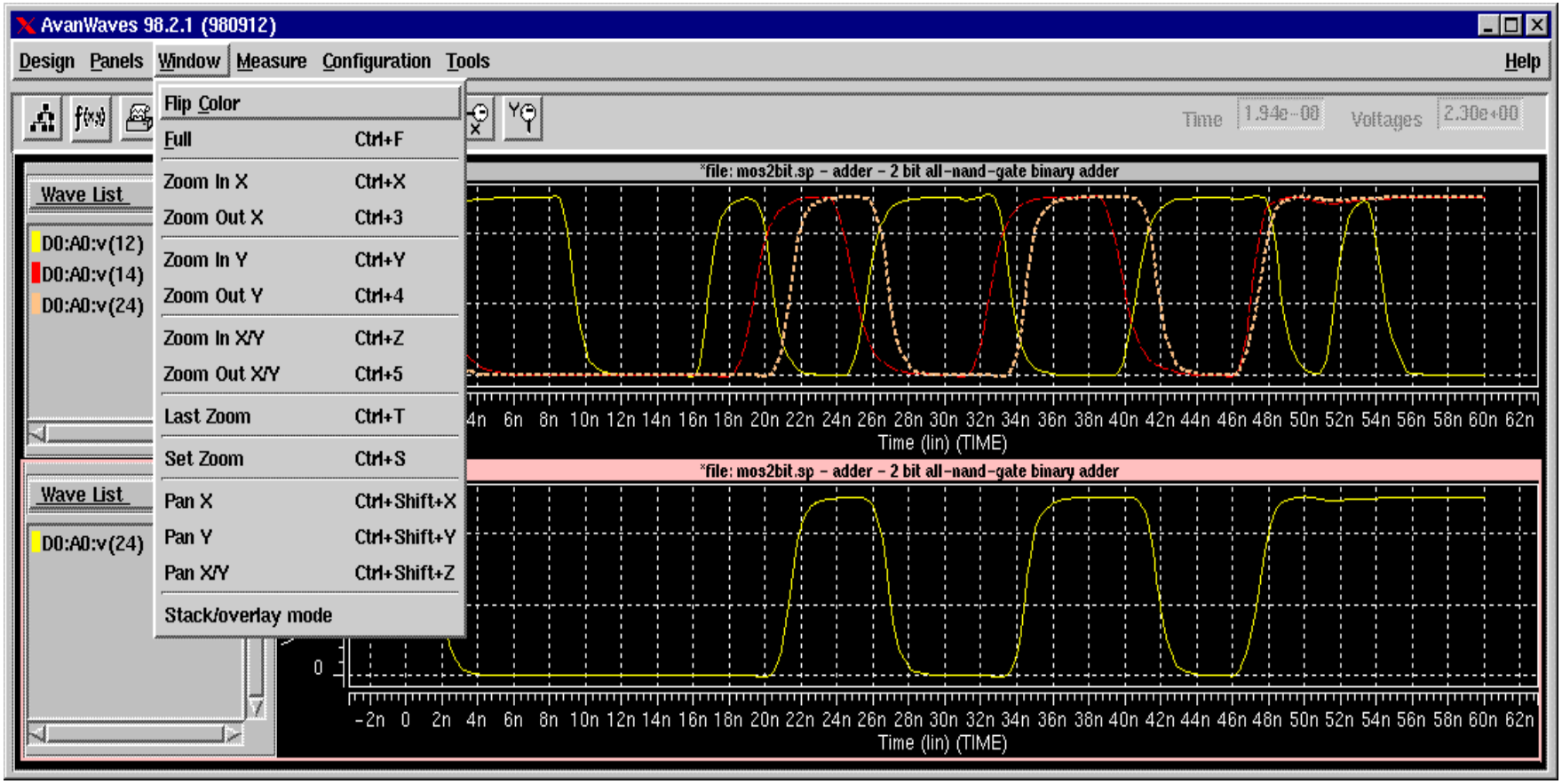


Awaves - Multiple Panels



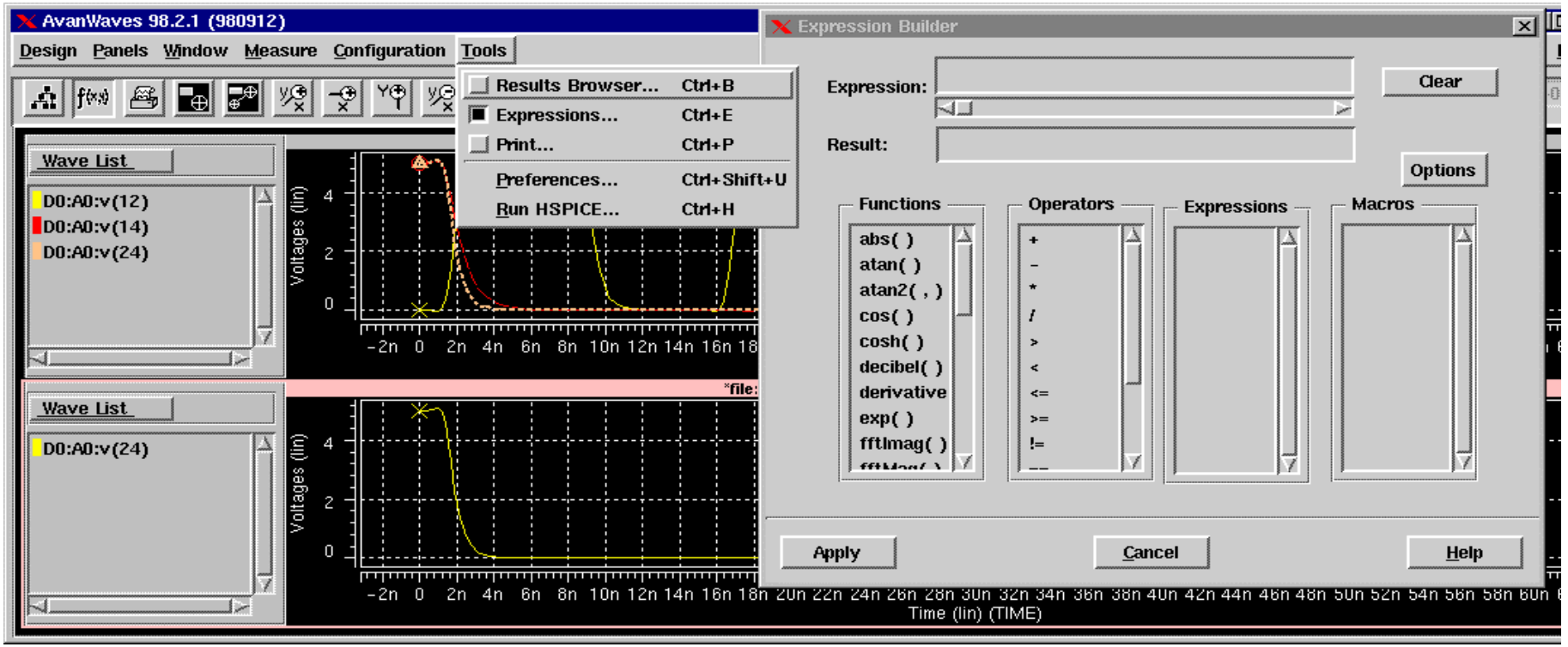
Awaves - View Port Management

Zoom size is independent between panels



Awaves - Expressions

Expressions provide capability for data calculation
Can use drag&drop to type expression



Lab1_Awaves

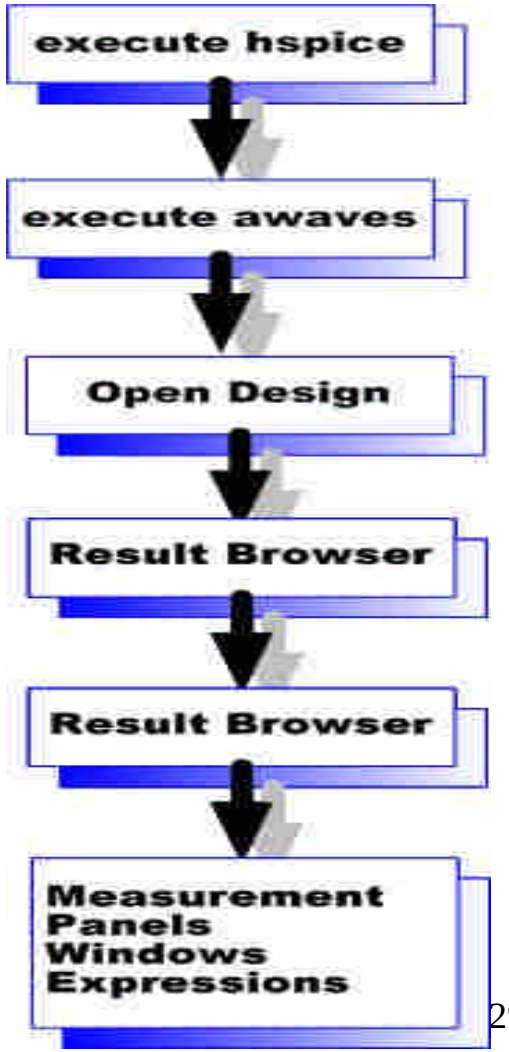
View simulation results with awaves.

Open inv.readme

more inv.readme

info: *** hspice job concluded**
HSPICE job inv.sp completed

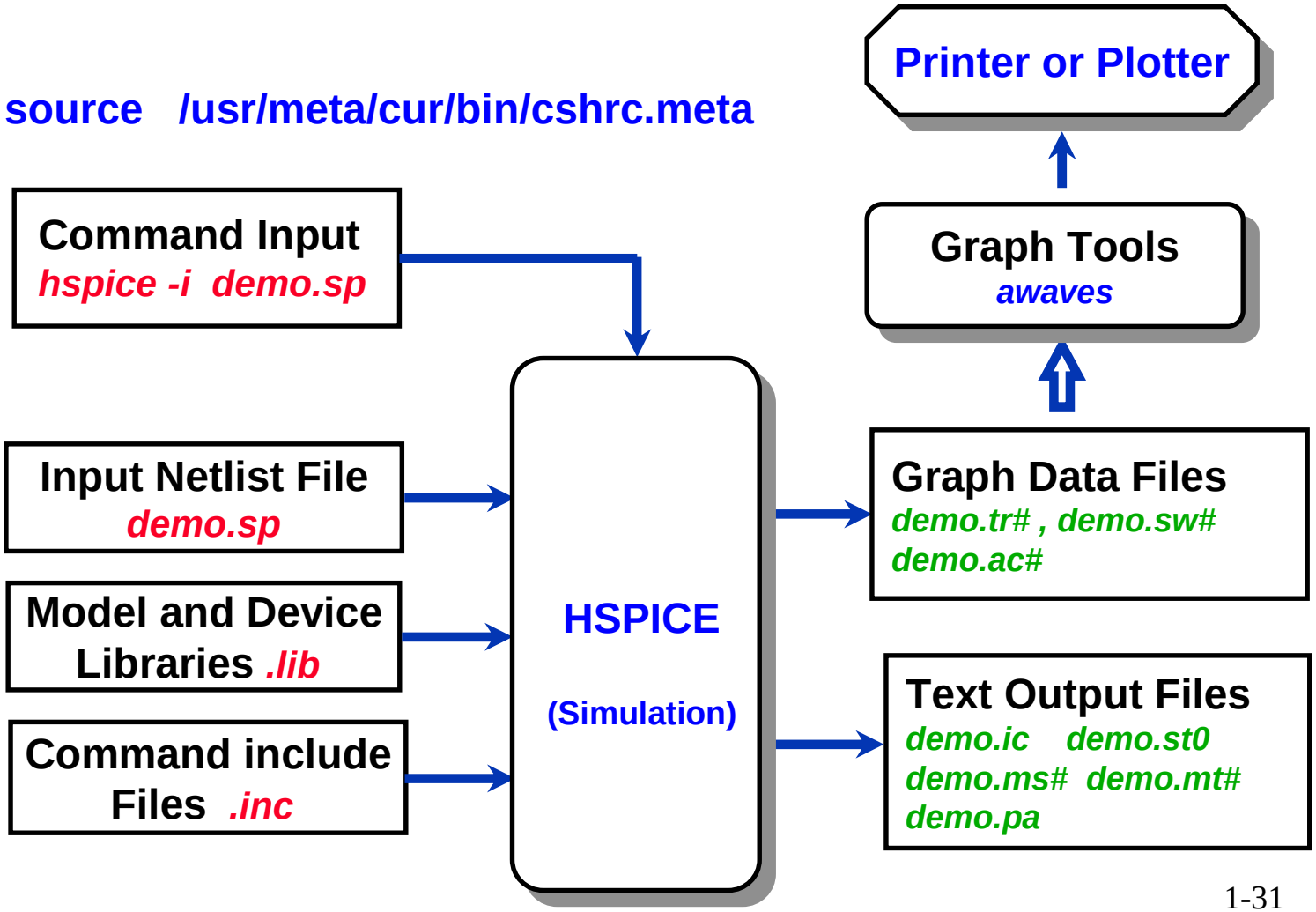
cic13> ls
inv.ic0 inv.pa0 inv.sp inv.st0 inv.tr0



Chapter 3

Simulation Input and Controls

HSPICE Data Flow



Netlist Statements and Elements

TITLE	First line is Input Netlist File Title
*** or $**	Commands to Describe Circuit
.OPTIONS	Set Conditions for Simulation
Analysis(AC,DC,TRAN..) & .TEMP	Statements to Set Sweep Variables
.PRINT/.PLOT/.PROBE/.GRAPH	Set Print, Plot, and Graph Variables
.IC or **.NODESET**	Sets Initial State
.VEC `digital_vector_file`	Sets Input Stimuli File
Sources (I or V)	Sets Input Stimuli
Schematic Netlist	Circuit Description
+	In first Column ,+, is Continuation Char.
.SUBCKT/.ENDS	Sets/Ends Subcircuit Description
.MEASURE (Optimization Optional)	Provides Scope-like Measurement Capability
.LIB or .INCLUDE	Call Library or General Include Files
.MODEL Library	Element Model Descriptions
.DATA or **.PARAM**	Specify parameters or Parametric Variations
.ALTER	Sequence for In-line Case Analysis
.DELETE LIB	Remove Previous Library Selection
.END	Required Statement to Terminate Simulation

Netlist Structure (SPICE Preferred)

Title	Title Statement - Ignored during simulation
	* Components
Components	c2 2 0 2pf
	r1 1 0 1k
	m1 1 2 3 4 mod L=10u W=30u
	x3 2 3 INV
	*Model & Subcircuit
Models & Subckts	.model... or .LIB or .Subckt
	* voltage sources
Sources	v3 3 0 dc 0 ac 0 0 pulse 0 1 0 0.1 0.1 4 8
	vin in 0 sin(0 2 10k 0.5 0)
Controls	.option nomod nopage
	.tran 1 10
	.print v(5) i(r1)
	.plot v(3) v(in)
End file	.end

Element and Node Naming Conventions

- **Node and Element Identification:**
  - Either Names or Numbers (e.g. data1, n3, 11, ...)
  - 0 (zero) is Always Ground
  - Trailing Alphabetic Character are ignored in Node Number, (e.g. 5A=5B=5)
  - Ground may be 0, GND, !GND
  - All nodes are assumed to be local
  - Node Names can be may Across all Subcircuits by a .GLOBAL Statement (e.g. .GLOBAL VDD VSS)

Element and Node Naming Conventions (Cont.)

- **Instance and Element Names:**

C	Capacitor
D	Diode
E,F,G,H	Dependent Current and Voltage Controlled Sources
I	Current
J	JFET or MESFET
K	Mutual Inductor
L	Inductor
M	MOSFET
Q	BJT
R	Resistor
O,T,U	Transmission Line
V	Voltage Source
X	Subcircuit Call

- **Path Names of Subcircuits Nodes:** e.g. V(X1.net1), I(X1.X4.net3),I(X2.Mn0)

Units and Scale Factors

- **Units:**

R	Ohm	(e.g. R1 n1 n2 1K)
C	Farad	(e.g. C2 n3 n4 1e-12)
L	Henry	(e.g. L3 n5 n6 1e-9)

- **Scale Factors :**

F	1e-15
P	1e-12
N	1e-9
U	1e-6
M	1e-3

K	1e3
Meg	1e6
G	1e9
T	1e12
DB	$20\log_{10}$

Examples:

1pF
1nH
10Meg Hz
vdb(v3)

Warning: in SBTSPICE 1.e-15F , will be interpreted as 1e-15 fento Farad

- **Technology Scaling :** All Length and Widths are in **Meters**

Using .options scale=1e-6 ➡ L=2 W=100

Library Input Statement

- **.INCLUDE Statement** Copy the content of file into netlist

```
.INCLUDE  '$installdir/parts/ad'
```

- **.LIB Definition and Call Statement** File reference and Corner selection

```
.LIB TT   ← Corner name
.MODEL nmos_tt nmos (level=49  Vt0=0.7
+TNOM=27 ..........................)
.ENDL  TT
```

```
.LIB '~users/model/tsmc/logic06.mod'   TT  ← Corner name
```

- **.PROTECT** ← Prevent the listing of included contents

```
.LIB "~users/model/tsmc/logic06.mod" TT
.UNPROTECT
```

Hierarchical Circuits, Parameters, and Models

- **.SUBCKT Statement : Description**
  - **.SUBCKT Syntax**

.SUBCKT subname n1 <n2 n3...> <param=val...>

n1 ... Node Number for External Reference; Cannot be Ground node (0)
Any Element Nodes Appearing in Subckt but not Included in this list are Strictly LOCAL, with these Exceptions :

(1) Ground Node (0)

(2) Nodes Assigned using .GLOBAL Statement

(3) Nodes Assigned using BULK=node in MOSFET or BJT Models

param Used ONLY in Subcircuit, Overridden by Assignment in Subckt Call or by values set in .PARAM Statement

.ENDS [subname]

Hierarchical Circuits, Parameters, and Models (Cont.)

- .SUBCKT Statement : Examples

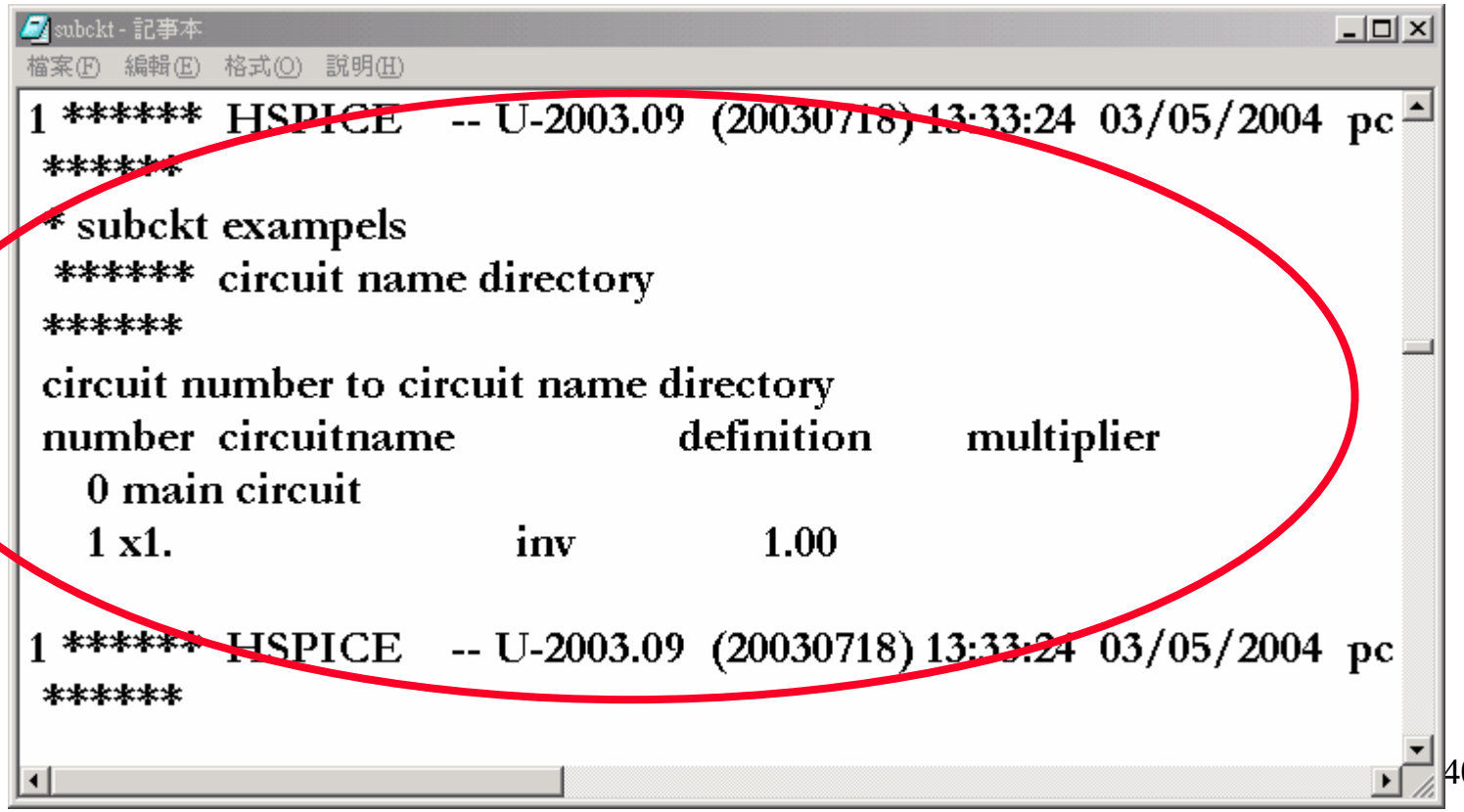
```
* subckt exampels
.PARAM VALUE=5V WN=2u WP=8u
*set up invert's sub-circuit
.SUBCKT  INV   IN  OUT   WN=2u  WP=8u
M1   OUT  IN  VDD  VDD  PCH  L=0.5u  W=WP
M2   OUT  IN   0      0     NCH  L=0.5u  W=WN
R1   OUT    4     1K
R2       4     5    10K
.ENDS    INV
* call sub-circuit INV
X1   1     2     INV    WN=5u    WP=20u
*.LIB Definition and Call Statement
.lib 'mm0355v.l' tt
* input signal
vin in 0 pulse (0 3.3 0 1n 1n 49n 100n)
*.tran type  analysis
.tran 50n 1u
.end
```

Hierarchical Circuits, Parameters, and Models (Cont.)

- **Subcircuit Calls (X Element Syntax)**

Xyyyy n1 <n2 n3...> subname <param=val...> <M=val>

XNOR3 ina inb inc out NOR WN=3u LN=0.5u M=2



Input Control Statements: .TEMP

- .TEMP Statement: Description
  - When TNOM is not Specified, it will Default to 25 °C for HSPICE
    When TNOM is not Specified, it will Default to 27 °C for SBTSPICE
  - Example 1:

```
.TEMP   30   $ Ckt simulated at 30 °C
```

  - Example 2:

```
.OPTION  TEMP =  30   $ Ckt simulated at 30 °C
```

  - Example 3:

```
.TEMP   100
D1  n1  n2  DMOD   DTEMP=30     $ D1  simulated at 130 °C
D2  n3  n4  DMOD                $ D2  simulated at 100 °C
R1  n5  n6     1K
```

HSPICE : DTEMP
SBTSPICE : TEMP

Input Control Statements : .ALTER

- **.ALTER Statement : Description**
  - **Rerun a Simulation Several Times with Different**

Circuit Topology
Models
Elements Statement
Parameter Values
Options
Analysis Variables, etc.

  - **1st Run : Reads Input Netlist File up to the first .ALTER**
  - **Subsequent : Input Netlists to next .ALTER, etc.**

Input Control Statements : .ALTER (Cont.)

- .ALTER Statement : Example

```
*file2: alter2.sp   alter examples      $ Title Statement
.lib 'mos.lib' normal
.param wval=50u Vdd=5V
r4   4   3    100
.
.
.alter
.del lib 'mos.lib' normal               $ remove normal model lib
. lib 'mos.lib' fast                    $ get fast model lib
.alter
.temp -50 0 50                          $ run with different temperature
r4   4   3    1K                         $ change resistor value
c3   3   0    10p                        $add the new element
.param wval=100u Vdd=5.5V               $ change parameters
.end
```

Input Control Statements : .ALTER (Cont.)

- **.ALTER Statement : Limitations**
  - **CAN Include:**
    - Element Statement (Include Source Elements)
    - .DATA, .LIB, .INCLUDE, .MODEL Statements
    - .IC, .NODESET Statement
    - .OP, .PARAM, .TEMP, .TF, .TRAN, .AC, .DC Statements
  - **CANNOT Include:**
    - .PRINT, .PLOT, .GRAPH, or any I/O Statements

Input Control Statements: .DATA

- **.DATA Statement:**

Operating point: .DC DATA=dataname

DC sweep: .DC vin 1 5 .25 SWEEP DATA=dataname

***AC sweep*:** .AC dec 10 100 10meg SWEEP DATA=dataname

***TRAN sweep*:** .TRAN 1n 10n SWEEP DATA=dataname

```
.DATA dataname pnam1 <pnam2 pnam3 ... pnamxxx >
+ pval1<pval2 pval3 ... pvalxxx>
+ pval1' <pval2' pval3' ... pvalxxx'>
.ENDDATA
```

Input Control Statements: .DATA (cont.)

- .DATA Statement: Inline or Multiline .DATA Example

Inline .DATA Example

```
.TRAN 1n 100n SWEEP DATA=devinf
.AC DEC 10 Hz 100khz SWEEP DATA=devinf
.DC TEMP -55 125 10 SWEEP DATA=devinf
*
.DATA devinf  Width   Length   Vth   Cap
+               10u     100u     2v    5p
+               50u     600u    10v   10p
+              100u     200u     5v   20p
......
.ENDDATA
```

Multiline .DATA Example

```
.PARAM Vds=0  Vbs=0  L=1.0u
.DC DATA=vdot
.DATA vdot
Vbs   Vds     L
 0     0.1    1.0u
 0     0.1    1.5u
-1     0.1    1.0u
 0     0.5    1.0u
.....
.ENDDATA
```

Input Control Statements: .OPTION

- **.OPTION Statement : Description**
  - **.Option Controls for**

    Listing Formats
    Simulation Convergence
    Simulation Speed
    Model Resolution
    Algorithm
    Accuracy

  - **.Option Syntax and Example**

    .OPTION opt1 <opt2> .... <opt=x>

    .OPTION LVLTIM=2 POST PROBE SCALE=1

Input Control Statements: .OPTION(Cont.)

- **.OPTION Keywords Summary :**
  - **General Control Options**

    Input, Output
    CPU
    Interfaces
    Analysis
    Error
    Version

  - **Model Analysis**

    General
    MOSFETs
    Inductors
    BJTs
    Diodes

  - **DC Operating Point and DC Sweep Analysis**

    Accuracy
    Matrix
    Input,Output
    Convergence
    Pole/Zero

  - **Transient and AC Small Signal Analysis**

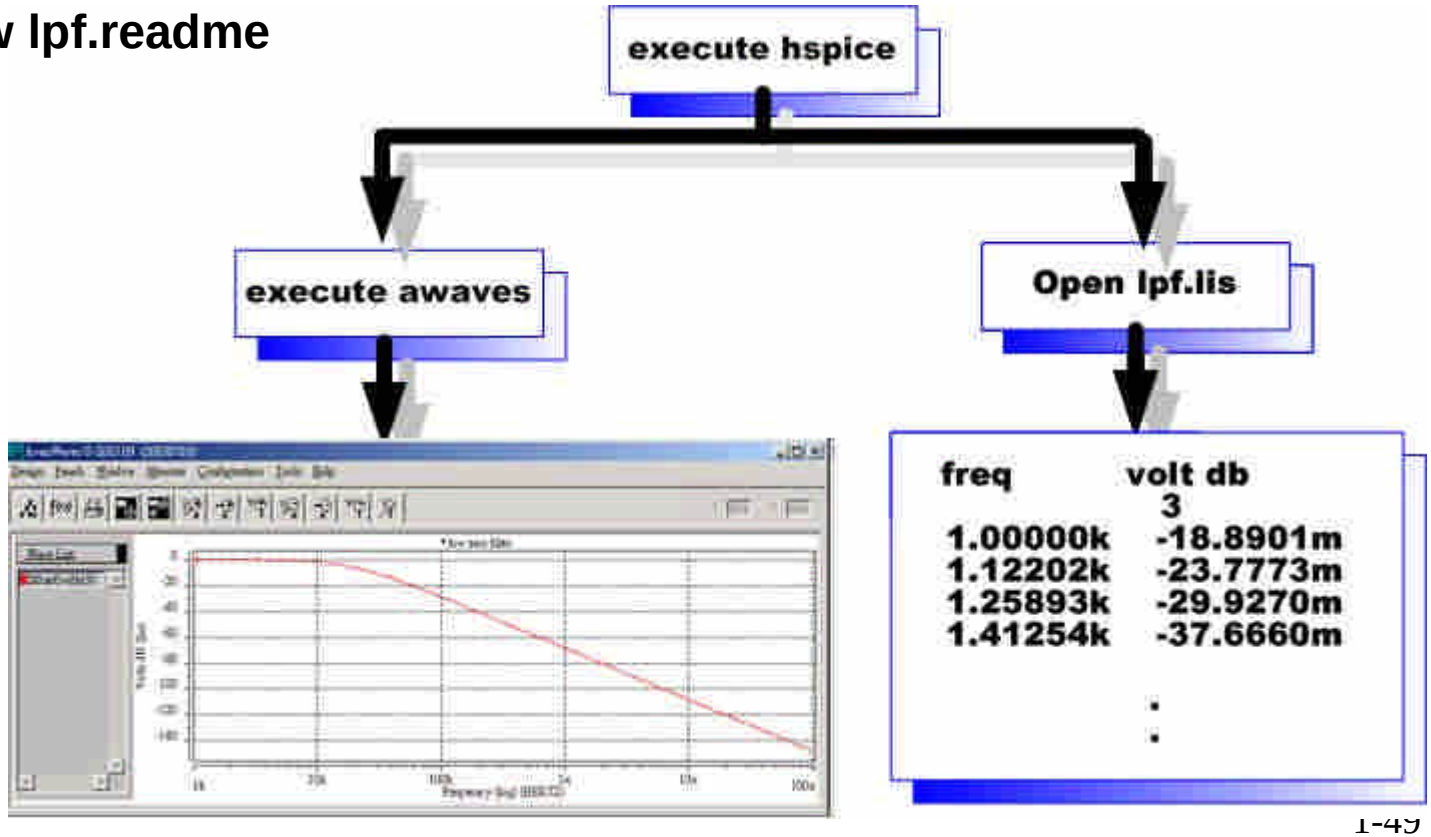
    Accuracy
    Speed
    Timestep
    Algorithm
    Input, Output

Lab2_ low-pass filter

Exercise 1 low-pass filter

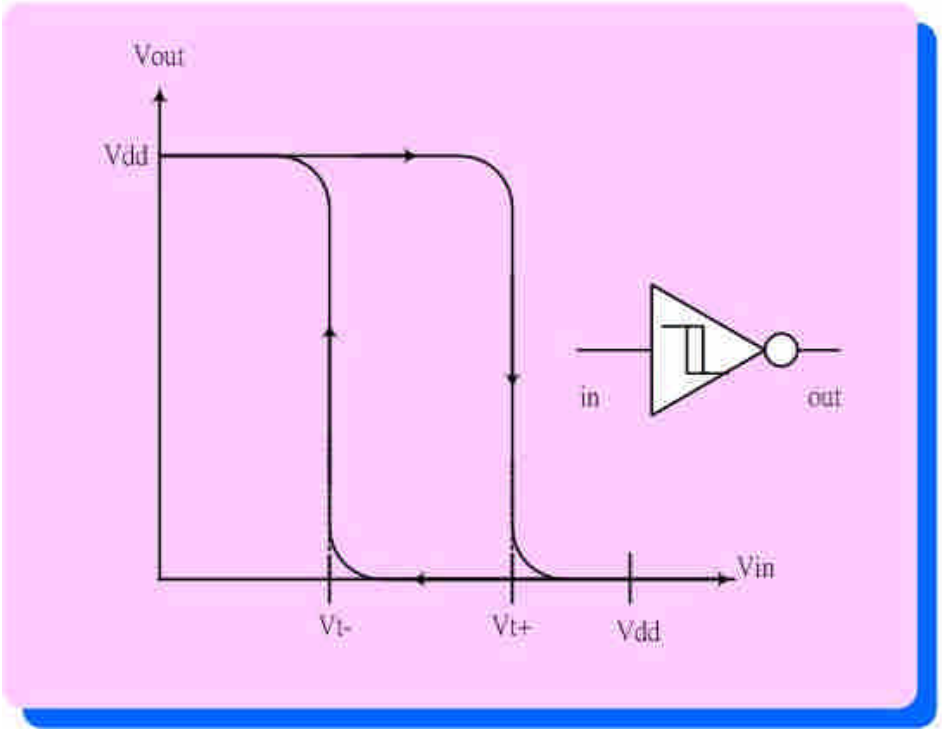
View simulation results with awaves and lpf.lis file

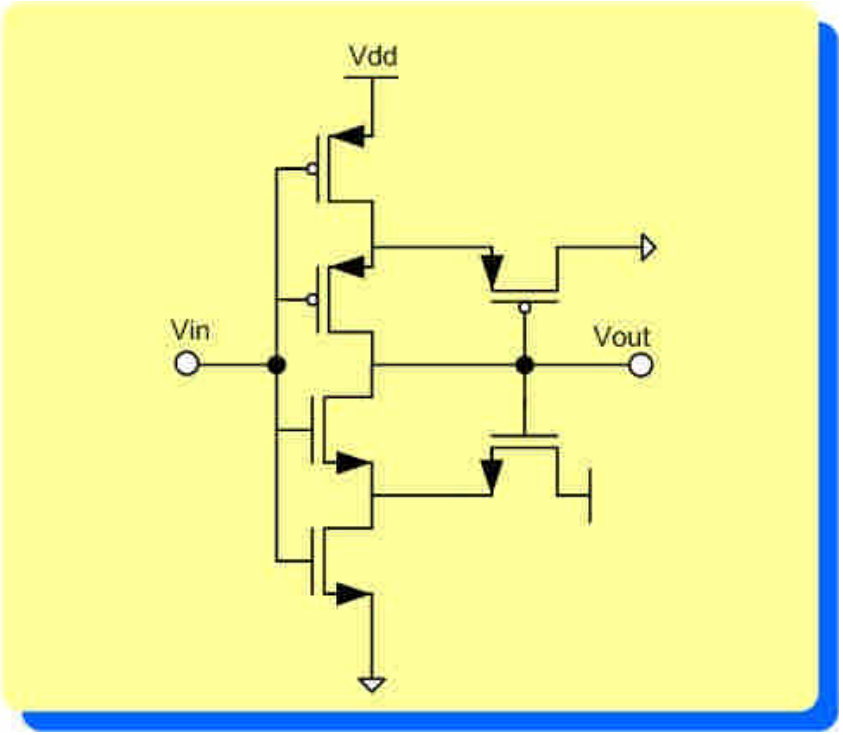
Follow lpf.readme



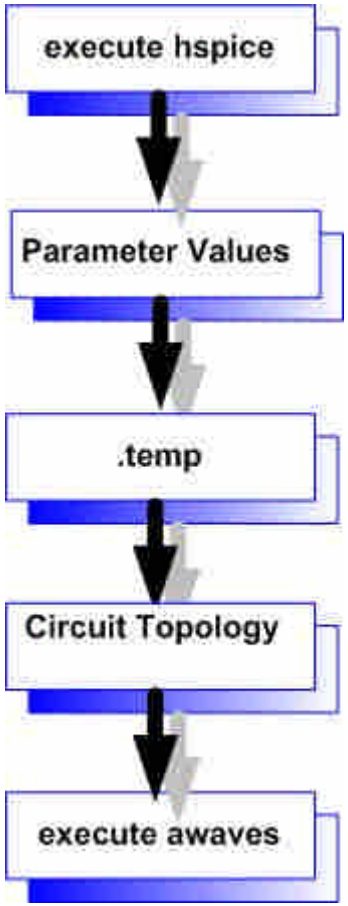
Lab2_ Schmitt trigger

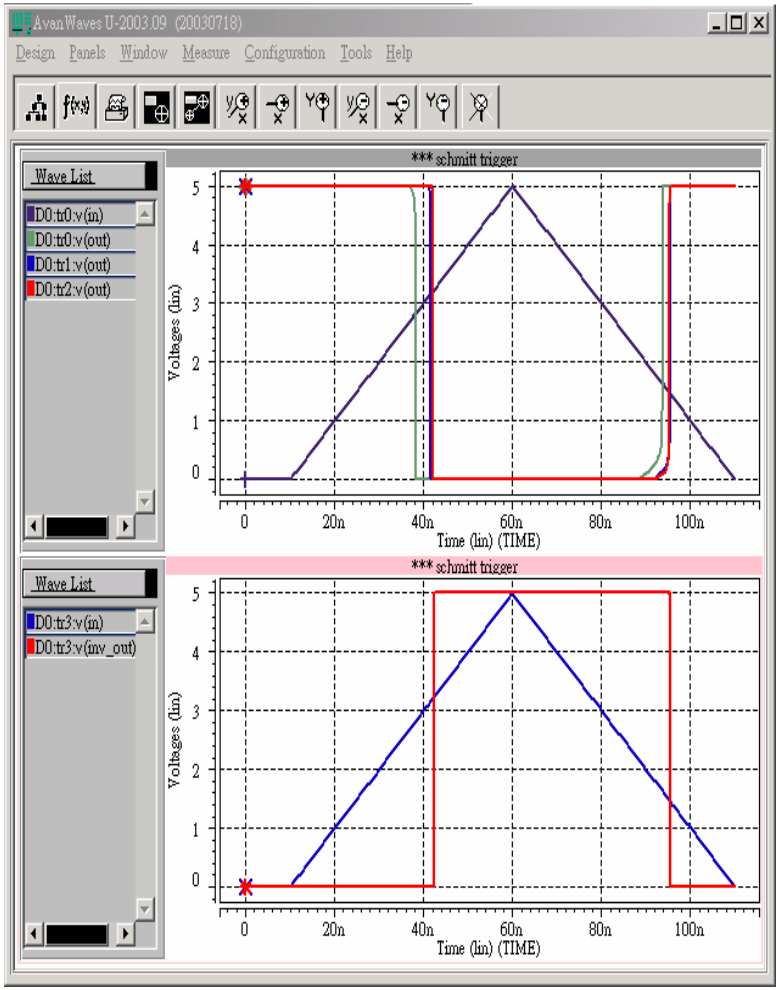
Exercise2 Schmitt trigger

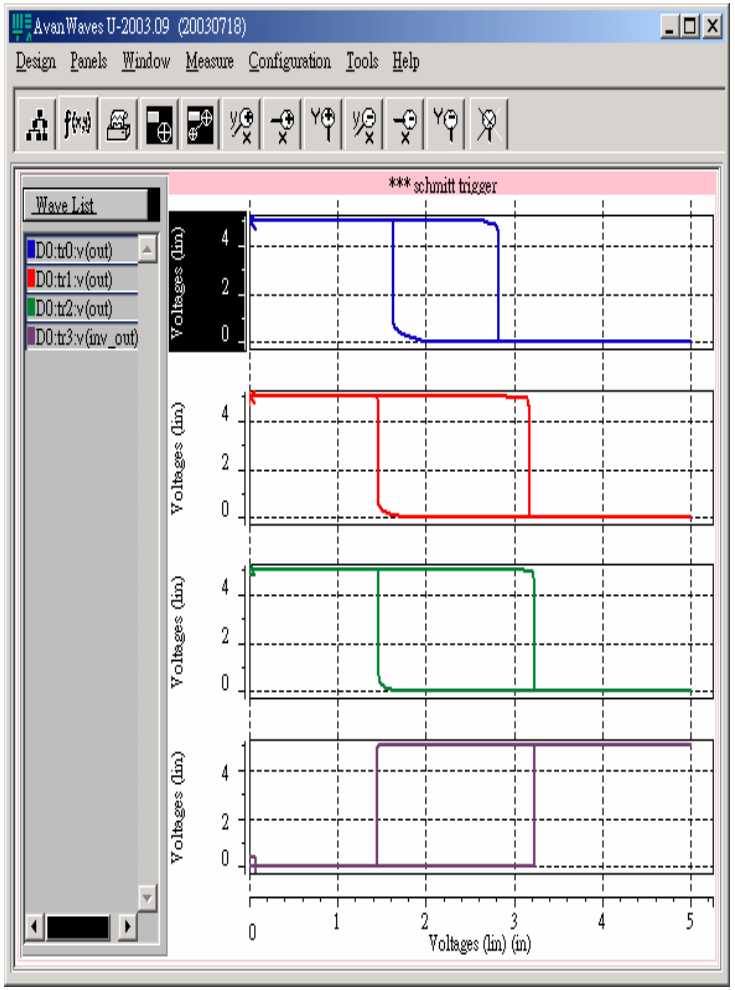




Lab2_ Schmitt trigger







Follow Schmitt_trigger.readme

Chapter 4

Sources and Stimuli

Source / Stimuli ： 提供電路驅動來源

1. 固定值獨立電源

   提供固定偏壓或固定驅動電流

2. 時變/頻變 獨立電源

   提供變動的電壓或電流輸入，一般供輸入信號用

3. 時變/頻變 壓控/源控 相依電源

   提供可控制的電壓或電流源，一般供建立模型用

   壓控電壓源(VCVS) -- *E Elements*

   壓控電流源(VCCS) -- *G Elements*

   流控電壓源(CCVS) -- *H Elements*

   流控電流源(CCCS) -- *F Elements*

AC, DC Sources

- **Source Element Statement :**
  - **Syntax :**

Vxxx n+ n- < <DC=>dcval> <tranfun> <AC=acmag, <acphase>>

Iyyy n+ n- < <DC=>dcval> <tranfun> <AC=acmag, <acphase> <M=val>

  - **Examples of DC & AC Sources :**

```
V1   1   0   DC=5V
V2   2   0   5V
I3   3   0   5mA
```

```
V4   4   0   AC=10V, 90
V5   5   0   AC  1.0   180
```

***AC or Freq. Response Provide Impulse Response**

  - **Examples of Mixed Sources :**

```
V6   6   0   5V   AC=1V, 90
V7   7   0   0.5V  AC  1.0 SIN (0 1 1Meg)
```

Transient Sources

- **Types of Independent Source Functions :**

Pulse (PULSE Function)

Sinusoidal (SIN Function)

Piecewise Linear (PWL Function)

Exponential (EXP Function)

Single-Frequency FM (SFFM Function)

Single-Frequency AM (AM Function)

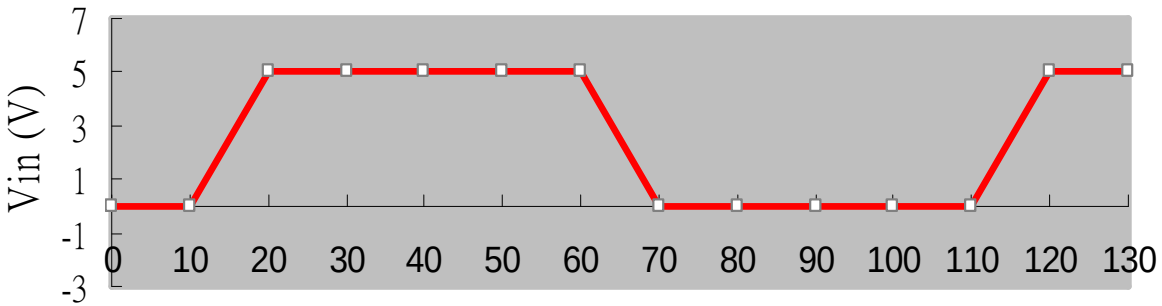
Pulse Source Function

- **Syntax :**

PULSE (V1 V2 < Tdelay Trise Tfall Pwidth Period >)

- **Example :**

Vin 1 0 PULSE (0V 5V 10ns 10ns 10ns 40ns 100ns)



Sinusoidal Source Function

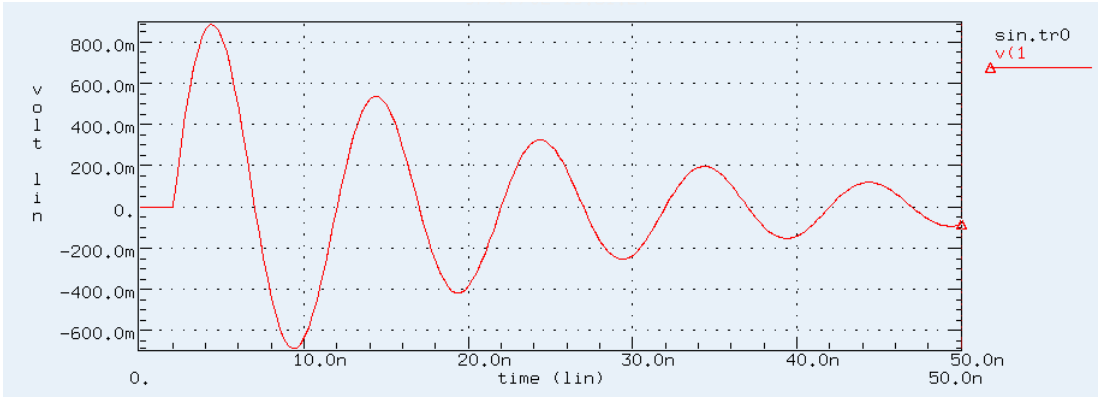
- **Syntax :**

SIN (Voffset Vacmag < Freq Tdelay Dfactor >)

Voffset + Vacmag* $e^{-(t-TD)*Dfactor}$ * sin(2π Freq(t-TD))

- **Example :**

Vin 3 0 SIN (0V 1V 100Meg 2ns 5e7)



Piecewise Linear Source Function

- **Syntax :**

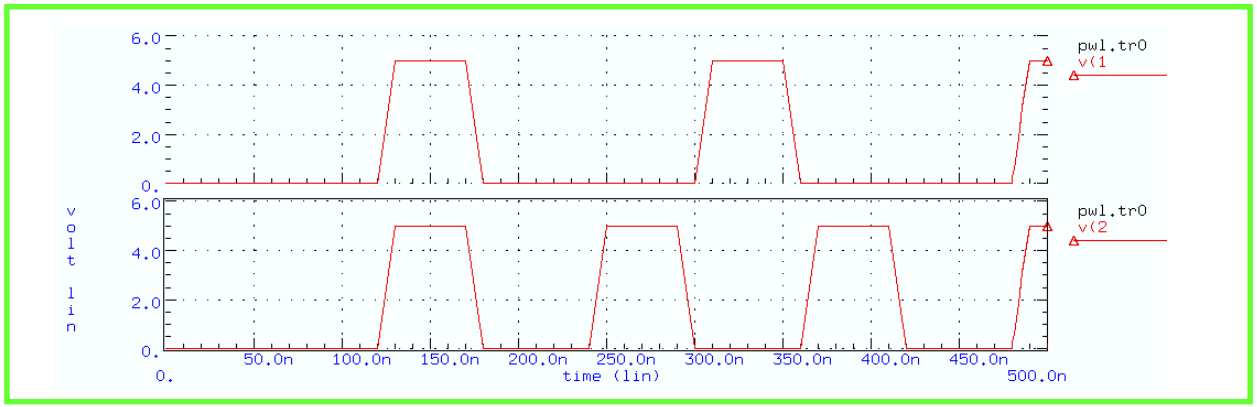
PWL ***(<t1 v1 t2 v2 .....> <R<=repeat>> <Tdelay=delay>)***

$ R=repeat_from_what_time TD=time_delay_before_PWL_start

- **Example :**

V1 1 0 PWL 60n 0v, 120n 0v, 130n 5v, 170n 5v, 180n 0v, R 0

V2 2 0 PL 0v 60n, 0v 120n, 5v 130n, 5v 170n , 0v 180n , R 60n



Specifying a Digital Vector File

- **Syntax :**

.VEC 'digital_vector_file' → **In spice Netlist**

The digital vector file consists of three parts:

- **Vector Pattern Definition**
- **Waveform Characteristics**
- **Tabular Data**

- **Digital Vector File Example (ref. adder2.sp):**

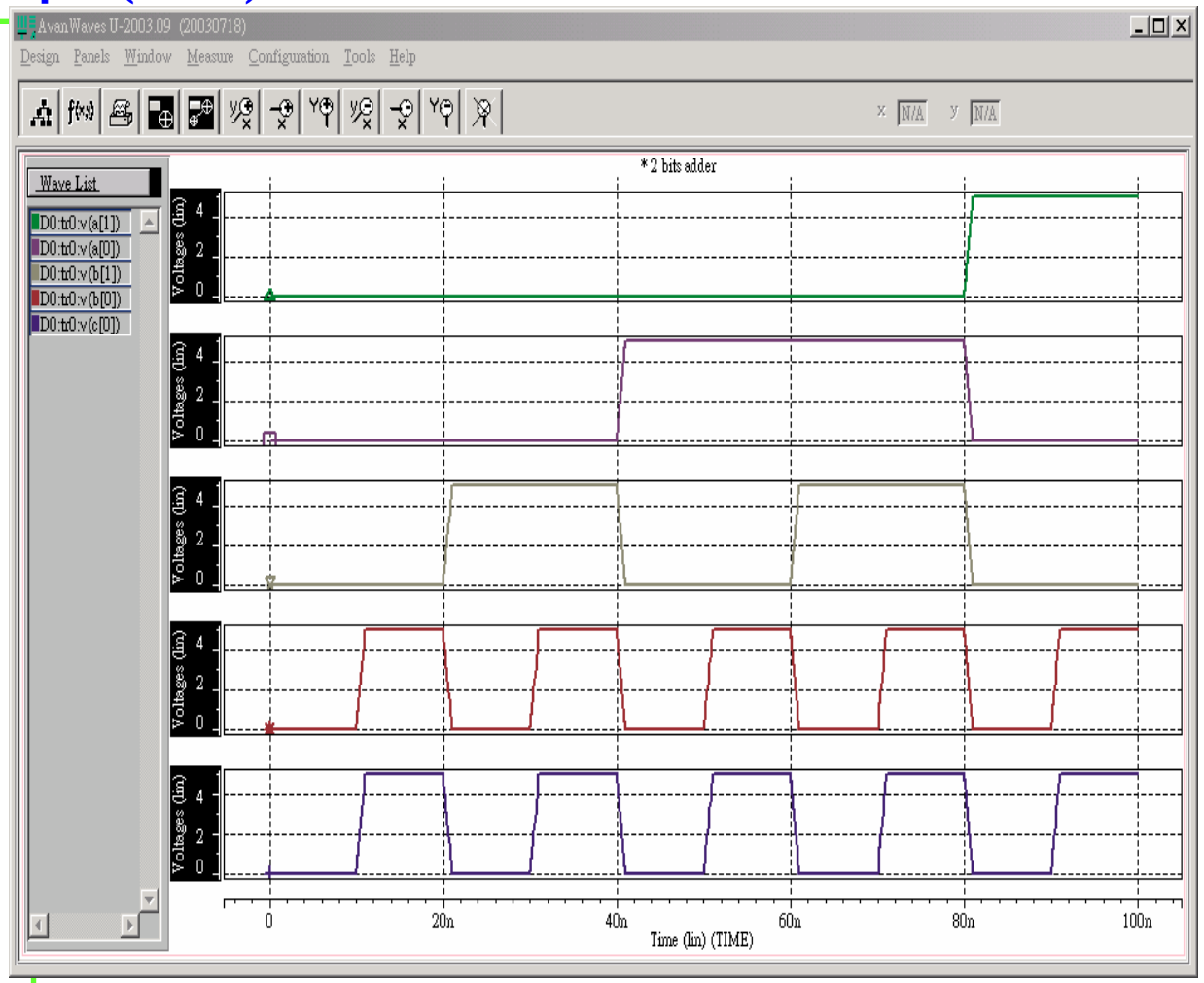
```
; Vector Pattern
RADIX 2 2 1
VNAME A[[1:0]] B[[1:0]] C[0]
io  i i  i
tunit  ns
```

Specifying a Digital Vector File*(Cont.)*

- **Digital Vector File Example (Cont.) :**

```
; Waveform Characteristics
slope 1
vil 0
vih 5
tunit ns
; Tabular Data
period 10
0 0 0   ;t=0
0 1 1   ;t=10
0 2 0   ;t=20
0 3 1
1 0 0
1 1 1
1 2 0
1 3 1
2 0 0
2 1 1
```

```
00  00  0
00  01  1
00  10  0
00  11  1
01  00  0
01  01  1
01  10  0
..  ..  .
```



VCVS --- E Elements

- **Syntax :**

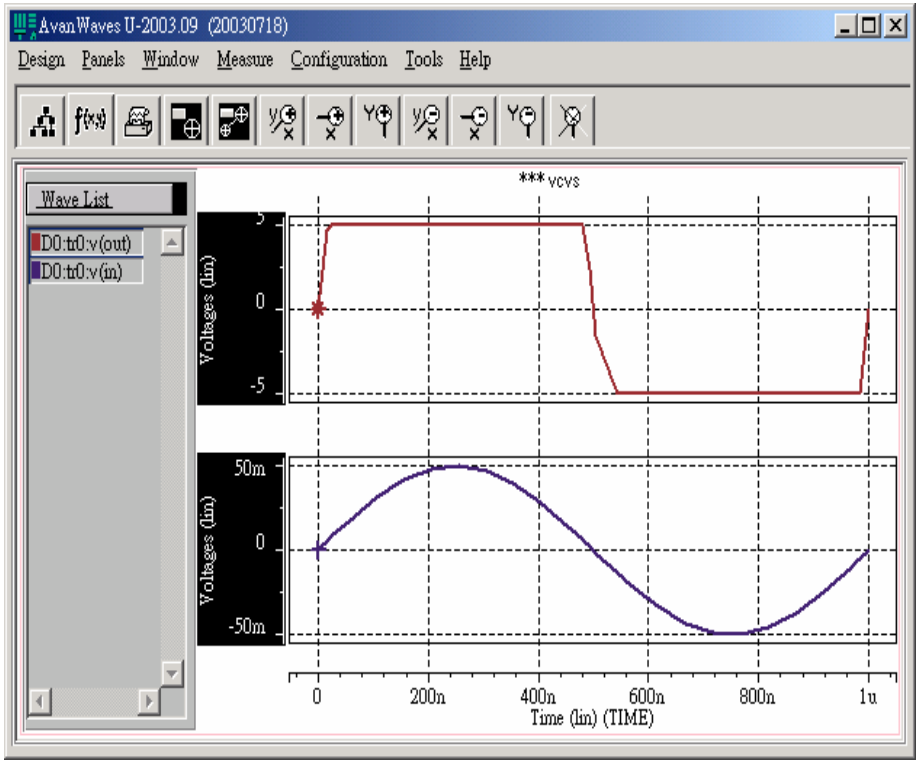
Linear form

Exxx n+ n- in+ in- gain <MAX=val> <MIN=val> <IC=val>….

- **Example :**

```
*** ideal op-amp example
vin     in     0  sin(0 50m 1meg)
EOPA out   0  in  0 max=+5 min=-5 1e3
.tran 10n 1u
.end
```

Ebuf 2 0 1 0 1.0



- ***Polynomial Controlled Sources***
  POLY(1) ,POLY(2), POLY(3)

VCCS --- G Elements

- **Syntax :**

Voltage Controlled Resistor (VCR)

Gxxx n+ n- VCR NPWL(1) in+ in- <TC1=val>x1,y1 x2,y2 ... <IC=val>

*** NPWLModels the symmetrical bidirectional switch or transfer
*** gate, NMOS

- **Example :**

*** Switch-Level MOSFET ***

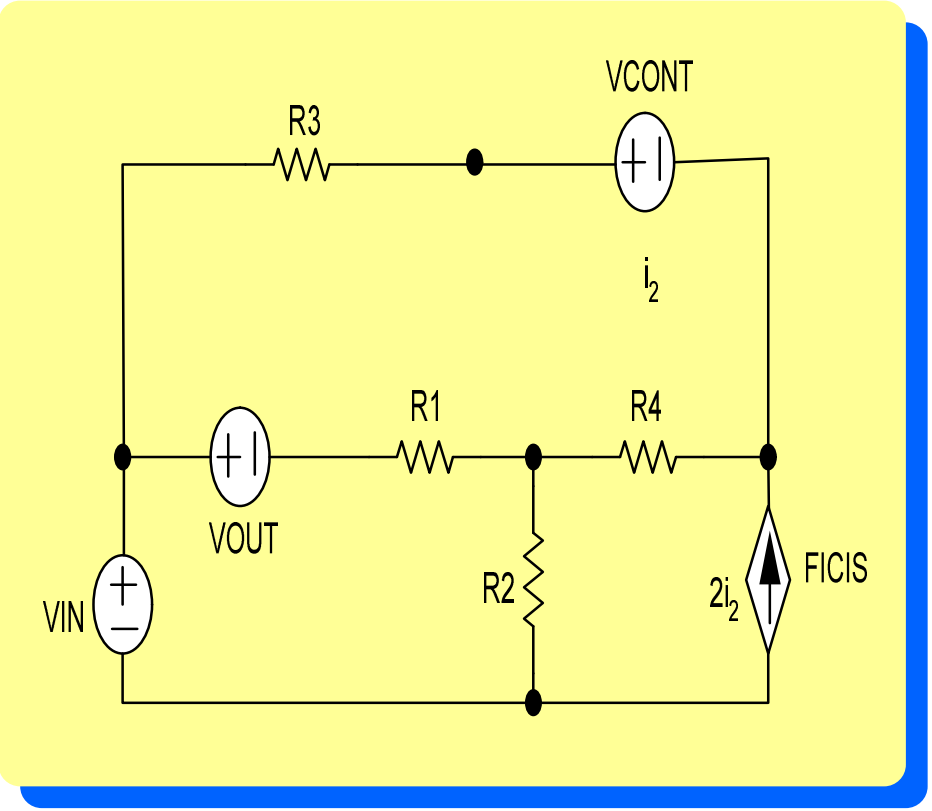
Gnmos d s VCR NPWL(1) g s LEVEL=1 0.4v,150g 1v,10meg 2v,50k
+ 3v,4k 5v,2k

*** Switch ***

Gswitch 2 0 VCR PWL(1) 1 0 0v,10meg 1v,1m

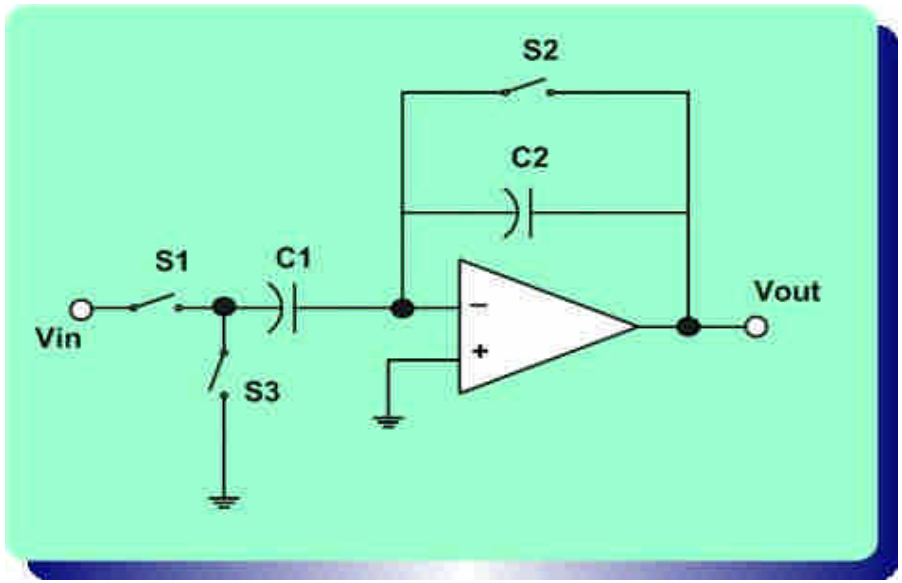
- **Voltage Controlled Capacitor (VCCAP)**

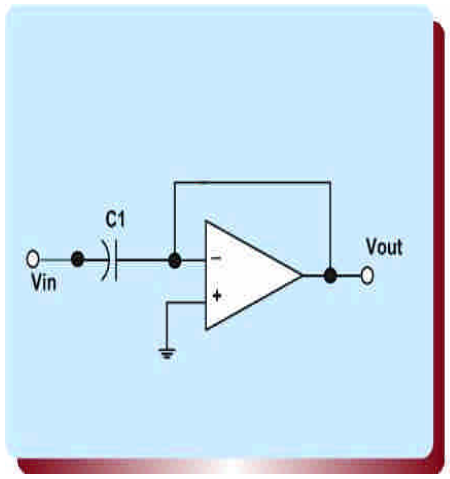
ICIS --- F Elements

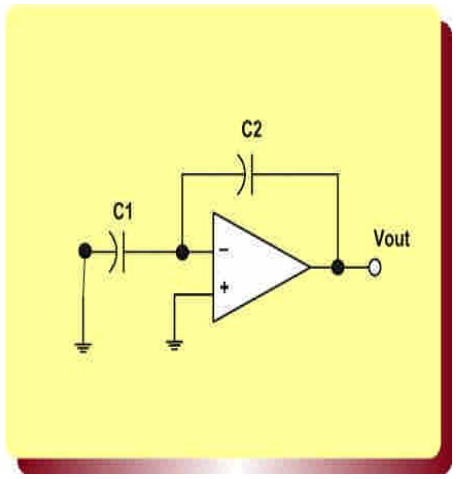


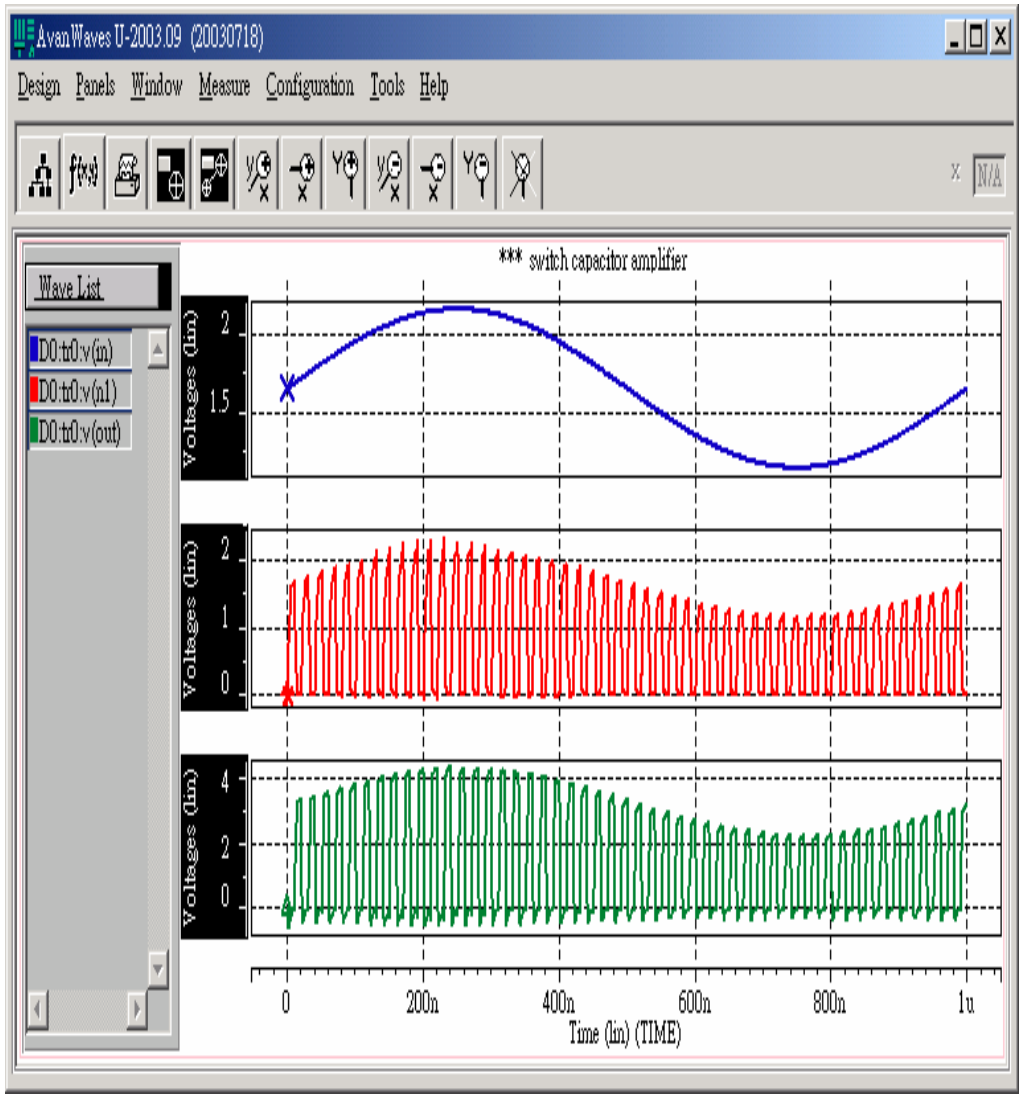

```
*** icis example
vin  1  0   dc 3
r1   5  2    1
r2   2  0    2
r3   1  4    3
r4   2  3    4
vcont  4  3  0
ficis  0  3  vcont   2
vout  1  5  0
.op
.end
```

Switch Capacitor Amplifier









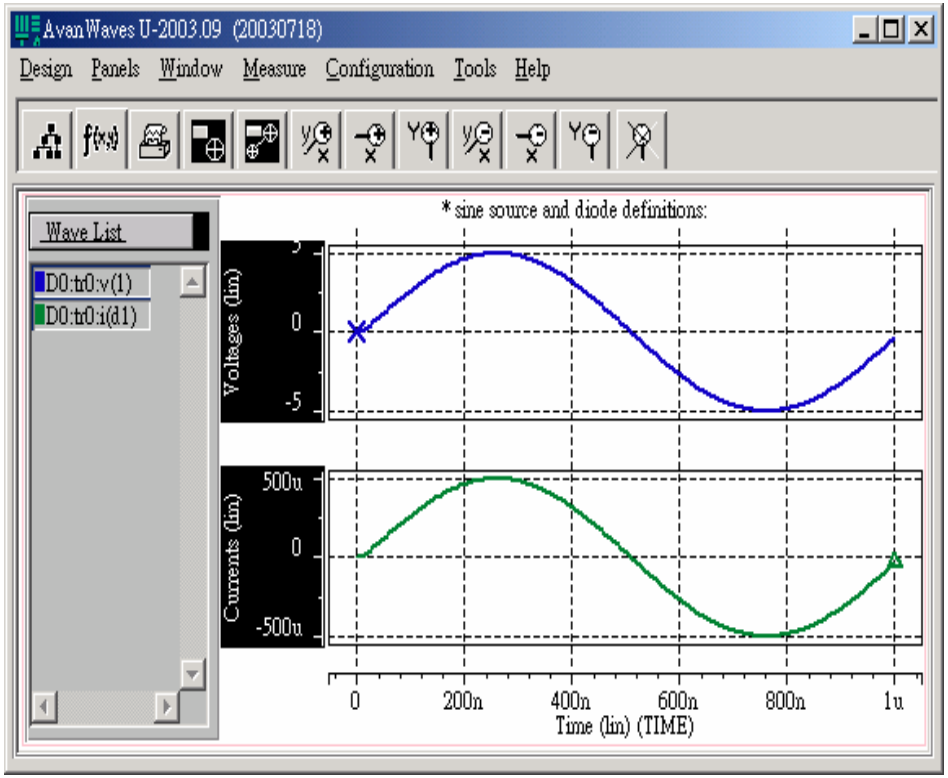
Switch Capacitor Amplifier

```
***  switch capacitor amplifier
eopamp out  0  inb  ina  1e4  * opa element
c1   n1   ina    2p
c2   ina  out    1p
gsw1 in    n1   VCR NPWL(1)  phi1 in    LEVEL=1 0.1,1e+28 .2,4k 1,2.8k 3,841  5,495
gsw2 ina  out  VCR NPWL(1)  phi1 ina  LEVEL=1 0.1,1e+28 .2,4k 1,2.8k 3,841  5,495
gsw3 n1   0     VCR NPWL(1)  phi2 n1   LEVEL=1 0.1,1e+28 .2,4k 1,2.8k 3,841  5,495
*** gsw1, gsw2,gsw3 Switch-Level MOSFET
*** voltage sources, clock signals
vin  in   0   sin(1.65 0.5 1meg)
vinb inb  0  0
vphi1 phi1 0  pulse(0 5 1n 1n 1n 9n 20n)
vphi2 phi2 0  pulse(5 0 1n 1n 1n 9n 20n)
*** transient type simulation
.tran 10n 1u
*** set environment
.option post
.end
```

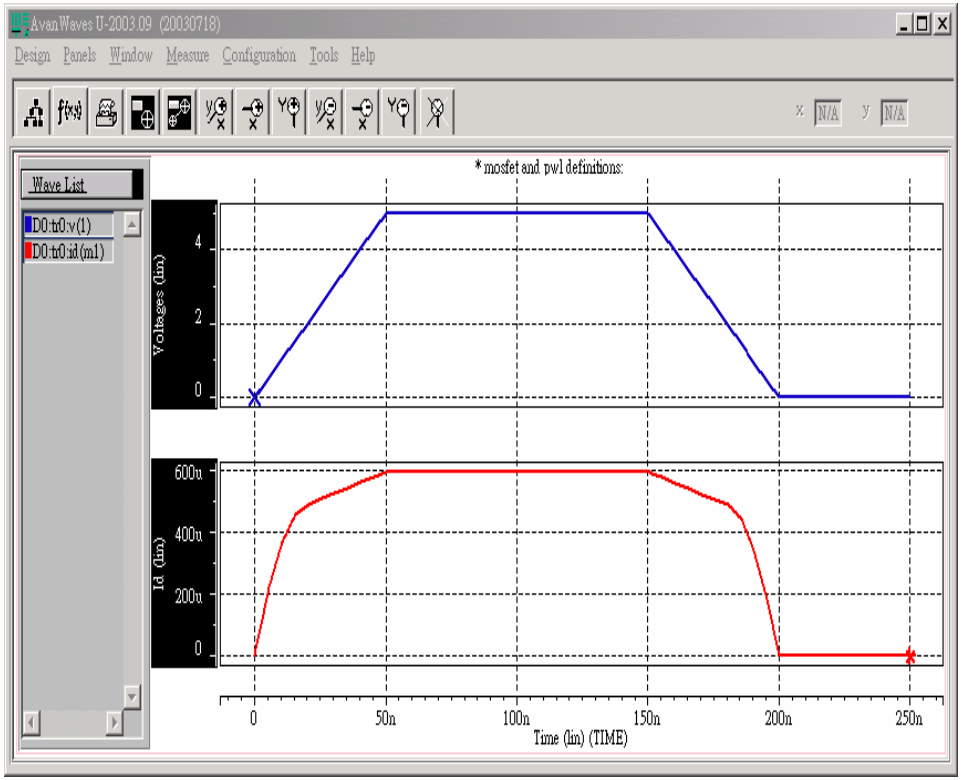
Lab3 Diode_sin & MOS_pwl

View simulation results with awaves.

diode_sin.sp



mos_pwl.sp



Chapter 5

Analysis Types

Analysis Types & Orders

- **Types & Order of Execution :**
  - **DC Operating Point : *First Calculated for ALL Analysis Types***

    .OP .IC .NODESET
  - **DC Sweep & DC Small Signal Analysis :**

    .DC .TF .PZ .SENS
  - **AC Sweep & Small Signal Analysis :**

    .AC .NOISE .DISTO .SAMPLE .NET
  - **Transient Analysis:**

    .TRAN .FOUR (UIC) .FFT
- **Other Advanced Modifiers :**
  - **Temperature Analysis, Optimization**

DC Operating Point Analysis

- **Initialization and Analysis:**
  - First Thing to Set the DC Operating Point Values for All Nodes and Sources : Set Capacitors OPEN & Inductors SHORT
  - Using .IC or .NODESET to set the Initialized Calculation
  - If UIC Included in .TRAN ==> Transient Analysis Started Directly by Using Node Voltages Specified in .IC Statement
  - .NODESET Often Used to Correct Convergence Problems in .DC Analysis
  - .IC force DC solutions, however .NODESET set the initial guess
- **.OP Statement :**
  - .OP Print out :(1). Node Voltages; (2). Source Currents; (3). Power Dissipation; (4). Semiconductors Device Currents, Conductance, Capacitance

DC Sweep & DC Small Signal Analysis

- **DC Analysis Statements :**
  - **.DC : Sweep for Power Supply, Temp., Param., & Transfer Curves**
  - **.OP : Specify Time(s) at which Operating Point is to be Calculated**
  - **.TF : Calculate DC Small-Signal Transfer Function (.OP is not Required)**
  - **.PZ : Performs Pole/Zero Analysis (.OP is not Required)**
  - **.SENS : Performs DC Small-Signal Sensitivities. (.OP is not Required)**
- **.DC Statement Sweep :**
  - **Any Source Value**
  - **Temperature Value**
  - **DC Circuit Optimization**
  - **Any Parameter Value**
  - **DC Model Characterization**
  - **DC Monte Carlo(Random Sweep)**

Sweep over model parameter is not allowed
Monte Carlo sweep is not supported in SBTSPICE

DC Sweep & DC Small Signal Analysis *(Cont.)*

- **.DC Analysis : Syntax**

```
.DC var1 start1 stop1 incr1 < var2 start2 stop2 incr2 > )
.DC var1 start1 stop1 incr1 < SWEEP var2 DEC/OCT/LIN/POI np start2 stop2 > )
```

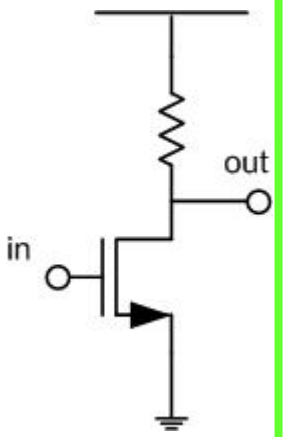
- **Examples :**

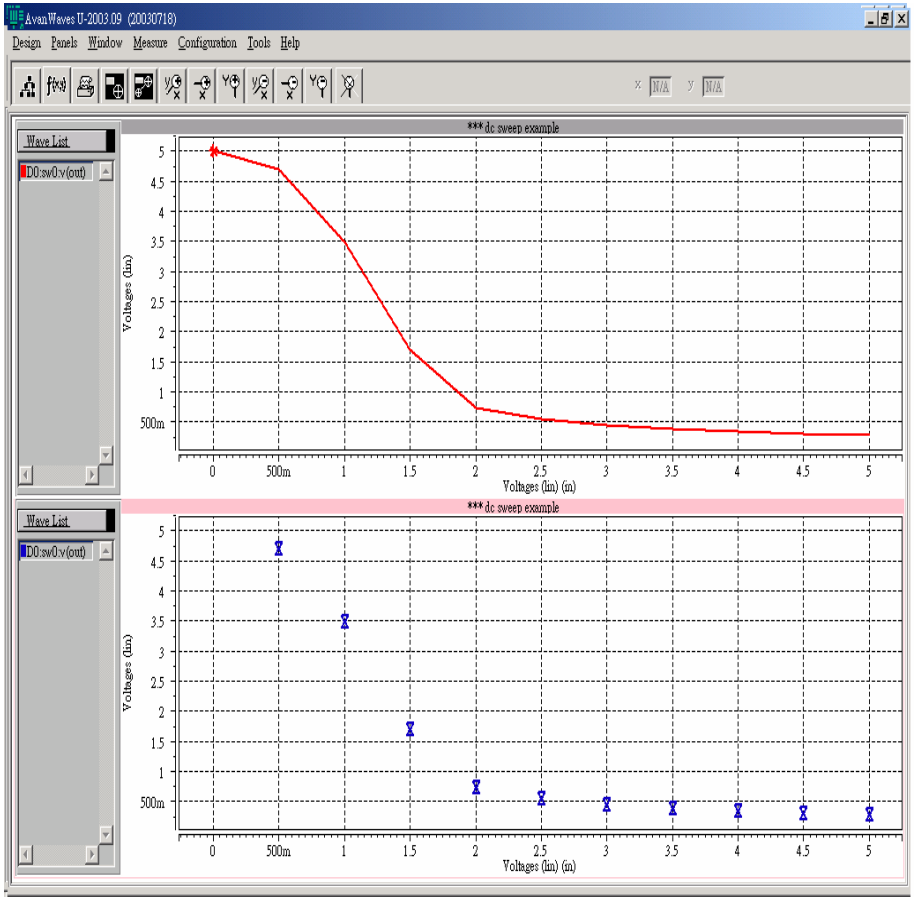
```
.DC  VIN  0.25  5.0  0.25
.DC  VDS  0  10  0.5  VGS  0  5  1
.DC  TEMP  -55  125  10
.DC  TEMP  POI  5    0  30  50  100  125
.DC  xval  1k  10k  0.5k  SWEEP  TEMP  LIN   5  25  125
.DC  DATA=datanm  SWEEP  par1  DEC  10   1k  100k
.DC par1 DEC  10  1k  100k  SWEEP  DATA=datanm
```

DC Sweep & DC Small Signal Analysis *(Cont.)*

- **Examples :**

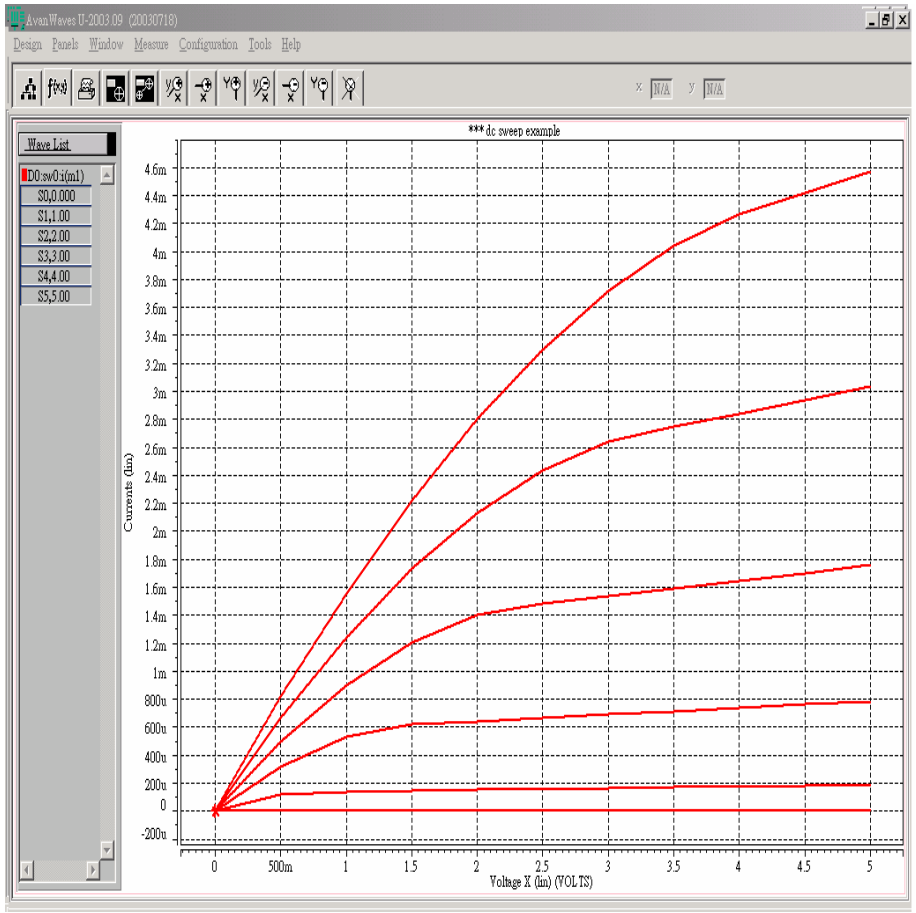
```
*** DC sweep example
m1 out in 0 0 nch l=1u w=1u
r1 vdd out 100k
vdd vdd 0 5
vin in  0
.model nch nmos level=2

.DC vin 0 5 0.5

.option post
.end
```





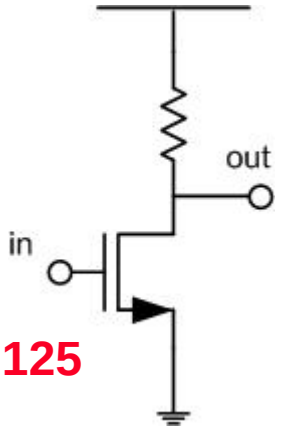
DC Sweep & DC Small Signal Analysis *(Cont.)*

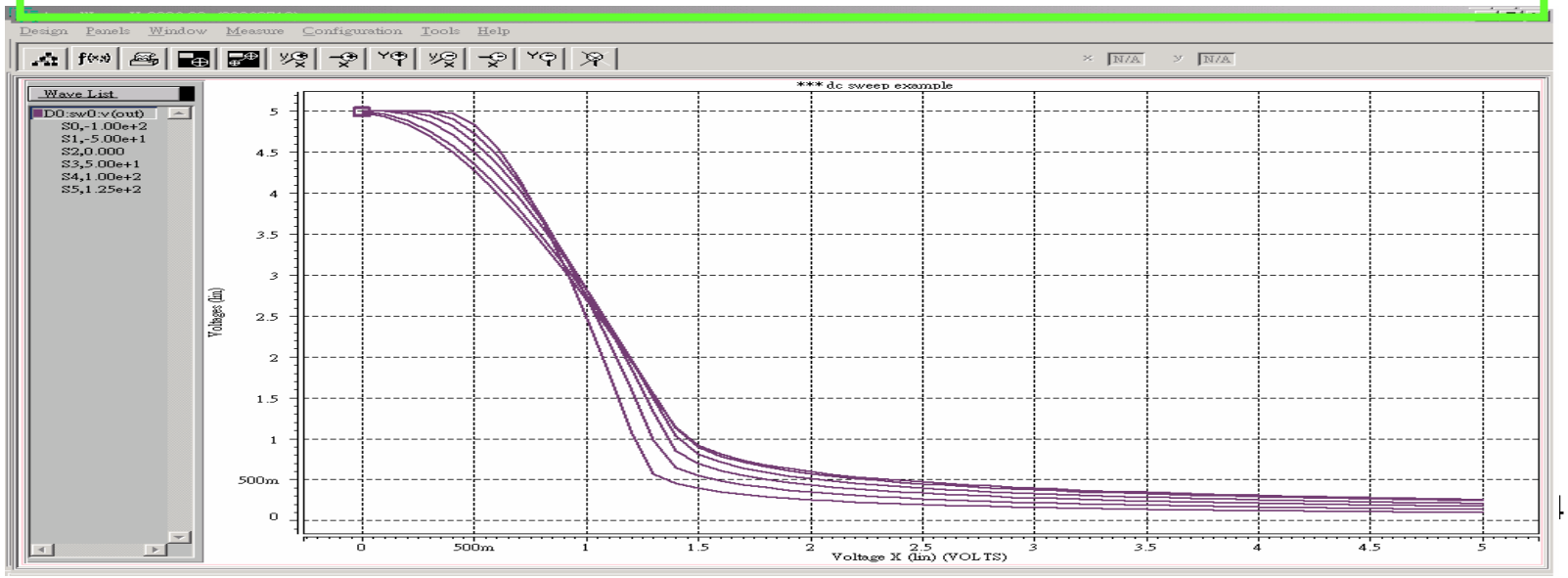
```
*** DC sweep example
m1 vd vg vs 0 nch l=1u w=10u
*** mos model
.model nch nmos level=2
*** voltage
vds vd vs
vgs vg vs

.DC VDS  0  5  0.5  VGS  0  5  1

.probe dc i(m1)
.end
```



DC Sweep & DC Small Signal Analysis *(Cont.)*

```
*** DC sweep example
m1 out in 0 0 nch l=1u w=1u
r1 vdd out 150k
vdd vdd 0 5
vin in  0
.model nch nmos level=2
.DC vin 0 5 0.1 SWEEP  TEMP POI  6 -100 -50 0 50  100 125
.option post
.end
```





DC Sweep & DC Small Signal Analysis *(Cont.)*

- **Transfer function statement**

```
.TF  V(OUT)  VIN      **output-variable, input-source
```

**NOTE: Compute Transfer Function v(out)/vin , Rin ,Rout

- **Example:**

```
*** tf example
.subckt inv in out
mn1 out in vss! vss! nch l=1u w=10u
mp1 out in vdd! vdd! pch l=1u w=20u
.ends
.model nch nmos level=1
.model pch pmos level=1
```

```
x1 in out inv
.global vdd! vss!
vdd! vdd! 0 5
vss! vss! 0 0
vin in 0
.tf v(out) vin
.end
```

```
****  small-signal transfer characteristics
    v(out)/vin                    =   -2.4582n
     input resistance at vin      =  1.000e+20
     output resistance at v(out)  =    1.1902k
```

DC Sweep & DC Small Signal Analysis *(Cont.)*

- **Pole/Zero statement**

```
.PZ  V(OUT)  VIN      **output-variable, input-source
```

**NOTE: Complete information about pole/zero analysis

```
******   pole/zero analysis  tnom=  25.000 temp=  25.000
....
     poles (rad/sec)              poles ( hertz)
*****************************************************************************
   real          imag          real          imag
  -1.0393x        0.         -165.4096k        0.
 -12.7888g        0.           -2.0354g        0.
.....
zeros (rad/sec)               zeros ( hertz)
*****************************************************************************
   real          imag          real          imag
 -12.7834g        0.           -2.0345g        0.
 -45.4697g      20.0122g       -7.2367g        3.1850g
```

DC Sweep & DC Small Signal Analysis *(Cont.)*

- **Pole/Zero Example:**

```
** differential amp
.subckt dopa va vb out
M1 net1 va   net2 0    nch w=25u l=1u
M2 out  vb   net2 0    nch w=25u l=1u
M3 net1 net1 vdd! vdd! pch w=8u   l=1u
M4 out  net1 vdd! vdd! pch w=8u   l=1u
M5 net2 net3 0    0    nch w=123u  l=1u
M6 net3 net3  vdd! vdd! pch w=1u  l=1u
M7 net3 net3  0    0    nch w=100u  l=1u
Cl out 0 5p
.ends
.model nch nmos level=2
.model pch pmos level=2
```

```
** supply voltage
.GLOBAL vdd! vss!
vdd vdd! 0 3.3
vss vss! 0  0
xopa1 ina inb out dopa
vr1   inb 0    dc 1.65
vin  ina inb ac 1.65 ** input signal
.option post
.pz v(out) vin
.end
```

DC Sweep & DC Small Signal Analysis *(Cont.)*

- **Sensitivity statement**

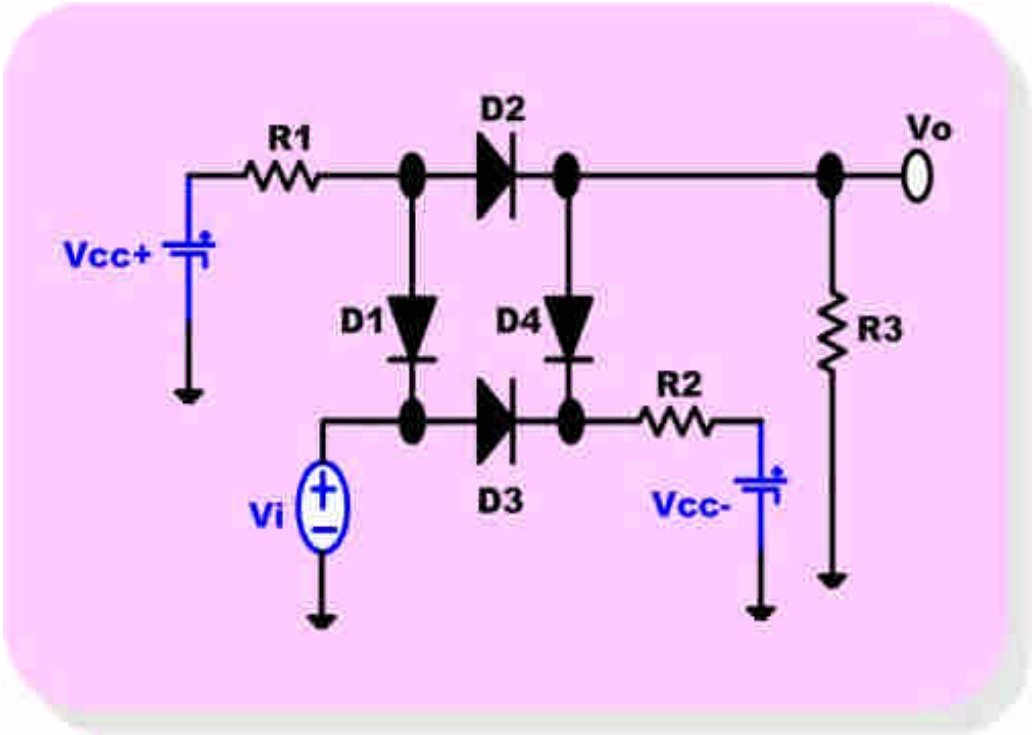
```
.SENS  V(OUT)  I(VCC)  **branch currents or nodal voltage for DC component
                       **sensitivity analysis
**NOTE: Generate very large amounts of output for large circuits
```

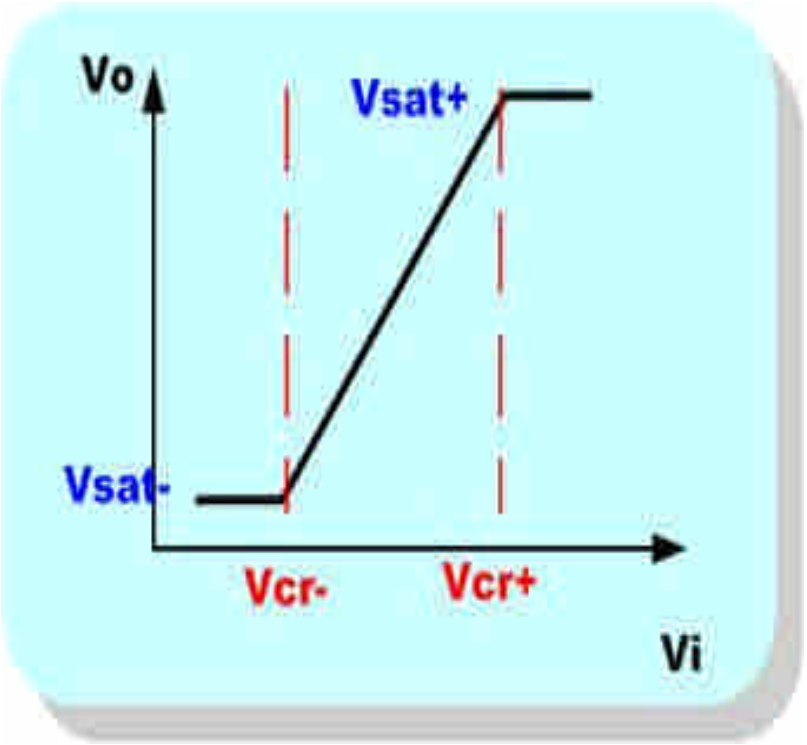
```
* .sens example
r1 1 2 1
r2 2 0 2
r3 2 3 3
r4 3 0 4
vin 1 0 10
.sens v(3)
.end
```

```
*obtains the dc small-signal sensitivities of …..
******  dc sensitivity analysis  ….
 dc sensitivities of output v(3)
      element        element       element        normalized
       name          value      sensitivity      sensitivity
                                (volts/unit)  (volts/percent)
      0:r1          1.0000      -1.3611         -13.6106m
      0:r2          2.0000     529.3006m         10.5860m
      0:r3          3.0000    -453.6862m        -13.6106m
      0:r4          4.0000     415.8790m         16.6352m
      0:vin        10.0000     347.8261m         34.7826m
      0:r4:i1       0.         478.2609m         0.
```

Floating – Diode Limiter

***error* only 1 connection at node 0:net5 defined in subckt 0**





AC Sweep & Small Signal Analysis

- **AC Analysis Statements :**
  - **.AC : Calculate Frequency-Domain Response**
  - **.NOISE : Noise Analysis**
  - **.DISTO : Distortion Analysis**
  - **.SAMPLE : Sampling Noise Analysis**
  - **.NET : Network Analysis**

- **.AC Statement Sweep :**
  - **Frequency**
  - **Temperature**
  - **Optimization**
  - **Element**
  - **.param Parameter**
  - **Monte Carlo(Random Distribution)**

AC Sweep & Small Signal Analysis(Cont.)

- .AC Analysis : Syntax

```
.AC  DEC/OCT/LIN/POI  np  fstart  fstop
.AC DEC/OCT/LIN/POI  np  fstart  fstop < SWEEP var start stop incr > )
```

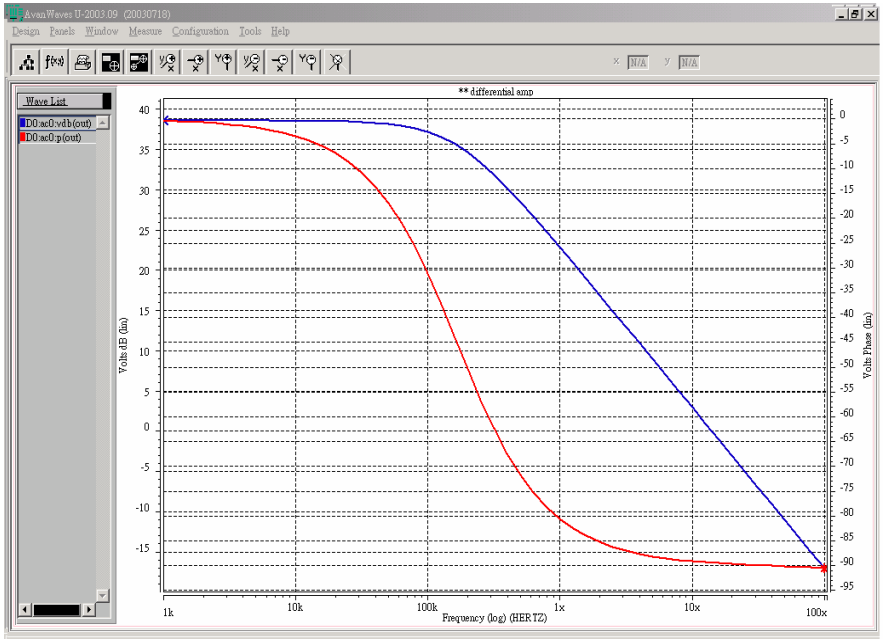
- Examples :

```
.AC  DEC  10  1K  100MEG
.AC  LIN  100  1  100Hz
.AC  DEC  10  1  10K  SWEEP  Cload  LIN  20  1pf  10pf
.AC  DEC  10  1  10K  SWEEP  Rx  POI  2  5K  15K
.AC  DEC   10  1  10K  SWEEP  DATA=datanm
```

AC Sweep & Small Signal Analysis(Cont.)

```
** differential amp
.subckt dopa va vb out
M1 net1 va   net2 0    nch w=25u l=1u
M2 out  vb   net2 0    nch w=25u l=1u
M3 net1 net1 vdd! vdd! pch w=8u   l=1u
M4 out  net1 vdd! vdd! pch w=8u   l=1u
M5 net2 net3 0    0    nch w=123u  l=1u
M6 net3 net3  vdd! vdd! pch w=1u  l=1u
M7 net3 net3  0    0    nch w=100u  l=1u
Cl out 0 5p
.ends
.model nch nmos level=2
.model pch pmos level=2
.GLOBAL vdd! vss!
vdd vdd! 0 3.3
vss vss! 0  0
xopa1 ina inb out dopa
vr1   inb 0    dc 1.65
```

```
vin  ina inb ac 1.65 ** input signal
.ac dec 10 1k 100meg
.probe vdb(out)
.option post
.end
```



AC Sweep & Small Signal Analysis(Cont.)

- **Other AC Analysis Statements:**
  - **.NOISE Statement :** **Only one noise analysis per simulation**

```
.NOISE  v(5)  VIN  10          $  output-variable, noise-input reference, interval
```

V(5) <- node output at which the noise output is summed
VIN <- noise input reference node
10 <- interval at which noise analysis summary is to be printed

Transient Analysis

- **Transient Analysis Statements :**
  - **.TRAN : Calculate Time-Domain Response**
  - **.FOUR : Fourier Analysis**
  - **.FFT : Fast Fourier Transform**
- **.TRAN Statement Sweep :**
  - **Temperature**
  - **Optimization**
  - **.Param Parameter**
  - **Monte Carlo (Random Distribution)**

Transient Analysis (Cont.)

- .TRAN Analysis : Syntax

```
.TRAN  tincr1  tstop1 < tincr2  tstop2 ..... > < START=val>
.TRAN  tincr1  tstop1 < tincr2  tstop2 ..... > < START=val> UIC <SWEEP..>
```

- Examples :

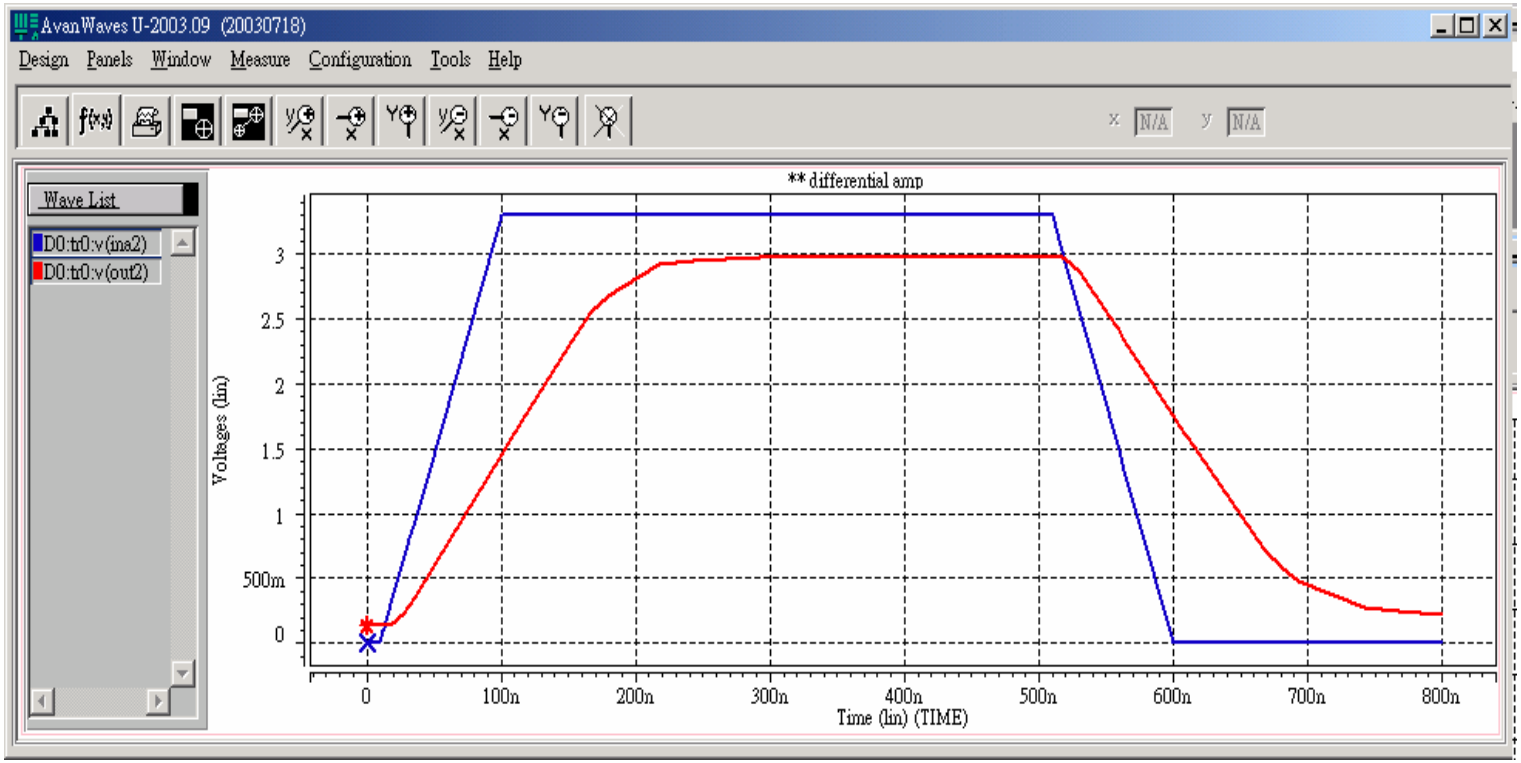
```
.TRAN   1NS  100NS
.TRAN  10NS  1US  UIC
.TRAN  10NS  1US  UIC  SWEEP  TEMP  -55  75  10  $ step=10
.TRAN  10NS  1US  SWEEP  load POI  3 1pf  5pf  10pf
.TRAN DATA=datanm
```

Transient Analysis (Cont.)

```
** differential amp
.subckt dopa va vb out
M1 net1 va   net2 0    nch w=25u l=1u
M2 out  vb   net2 0    nch w=25u l=1u
M3 net1 net1 vdd! vdd! pch w=8u    l=1u
M4 out  net1 vdd! vdd! pch w=8u    l=1u
M5 net2 net3 0    0    nch w=123u  l=1u
M6 net3 net3  vdd! vdd! pch w=1u  l=1u
M7 net3 net3  0    0    nch w=100u  l=1u
Cl out 0 5p
.ends
.model nch nmos level=2
.model pch pmos level=2
.GLOBAL vdd! vss!
vdd vdd! 0 3.3
vss vss! 0  0
xopa2 ina2 out2 out2 dopa
vin2  ina2 0 pwl (0 0 10n 0 100n  3.3 510n  3.3 600n 0)
```

Transient Analysis (Cont.)

```
.tran 10n 800n
.option post
.end
```



Transient Analysis (Cont.)

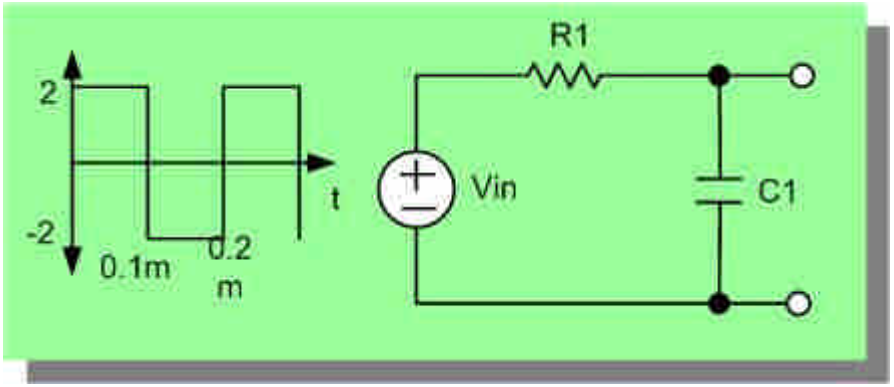
- **Other Transient Analysis Statements:**
  - **.FOUR Statement :**

.FOUR freq ov1 …. $ fundamental-freq , output-variable1,2,.....

Note1: As a part of Transient Analysis

Note2: Determines DC and first Nine AC Harmonics & Reports THD (%)

```
*** fourier series example
vin   in   0   pulse (-2  2
+ 0 10n 10n  0.1m  0.2m )
r1    in   out  100k
c1   out   0   1n
.tran  1u .2m
.four  5k v(out)
.end
```



$$THD = \frac{1}{R1} \cdot \left(\sum_{m=2}^{9} R_m^2 \right)^{1/2} \cdot 100\%$$

Transient Analysis (Cont.)

fourier components of transient response v(out)

dc component = -4.658D-01

harmonic no	frequency (hz)	fourier component	normalized component	phase (deg)	normalized phase (deg)
1	5.0000k	824.3853m	1.0000	-92.4135	0.
2	10.0000k	146.4523m	177.6503m	-170.2336	-77.8200
3	15.0000k	134.3362m	162.9531m	-131.0504	-38.6368
4	20.0000k	73.9205m	89.6674m	-174.0062	-81.5927
5	25.0000k	68.4249m	83.0011m	-146.3467	-53.9332
6	30.0000k	49.3735m	59.8912m	-174.7982	-82.3847
7	35.0000k	46.1717m	56.0074m	-153.9257	-61.5122
8	40.0000k	37.0599m	44.9546m	-174.8353	-82.4217
9	45.0000k	35.0115m	42.4698m	-158.1674	-65.7538

total harmonic distortion = 28.9121 percent

Transient Analysis (Cont.)

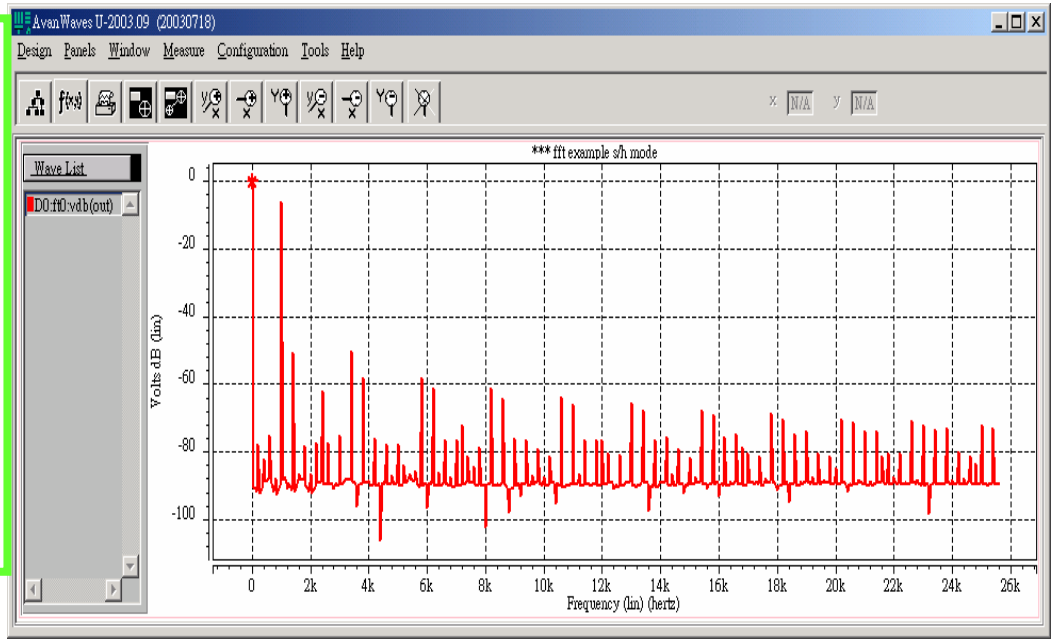
- **.FFT Statement :**

.FFT v(1,2) np=1024 start=0.3m stop=0.5m freq=5K window=Hamm alfa=2.5

Note1: Window Types : RECT, BLACK, HAMM, GAUSS, KAISER, HINN....

Note2: Determines DC and first Ten AC Harmonics & Reports THD (%)

```
*** fft example s/h mode
m1 out clk in vss! nch l=1u w=5u
c1 out 0 1p
vin   in 0 sin(1 1 1k)
vclk clk 0 pulse(0 5 0 1n 1n 5u 10u)
.model nch nmos level=2
.fft v(out) 1k  1024
.tran 10u 20m
.option post
.end
```



Transient Analysis (Cont.)

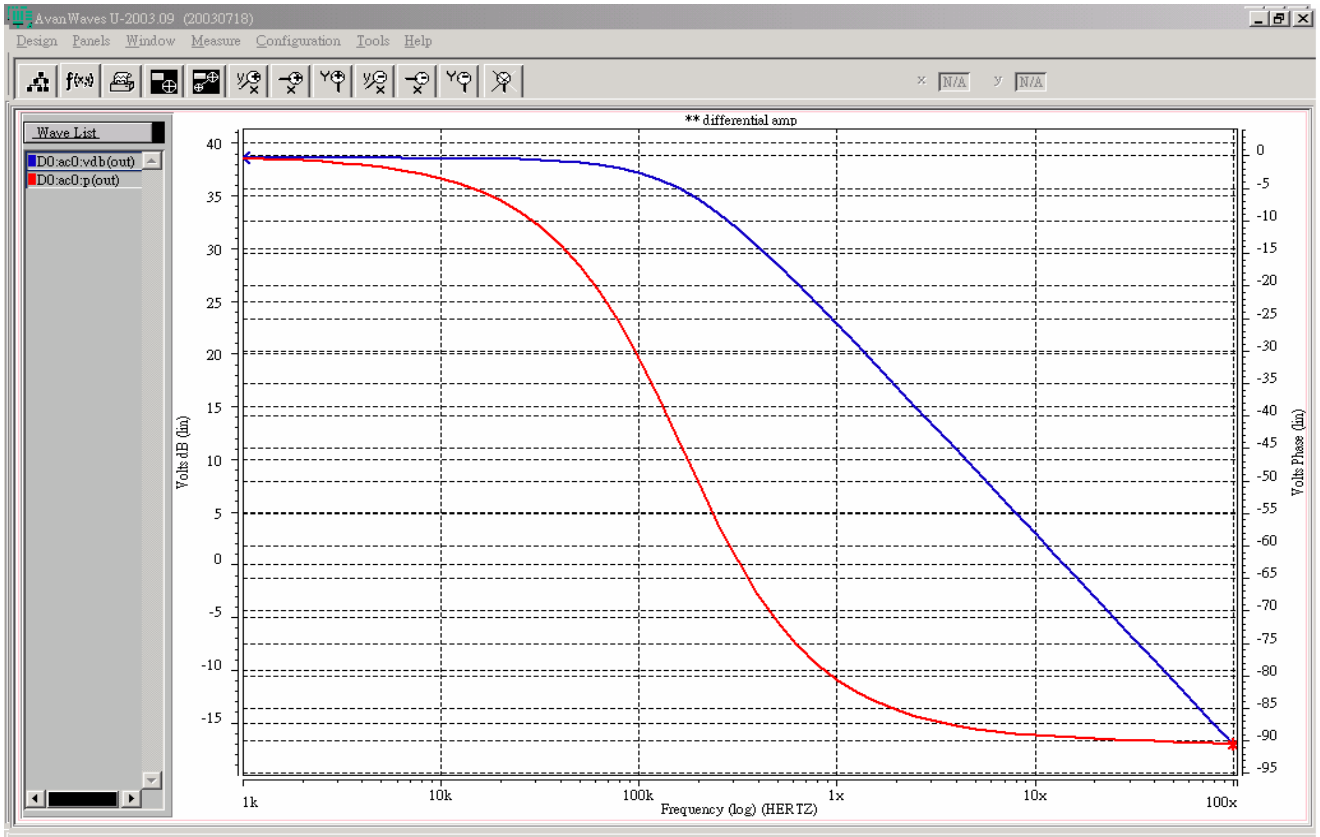
fft components of transient response v(out)
Window: Rectangular
DC: ON
First Harmonic: 1.0000k
Start Freq: 1.0000k
Stop Freq: 25.6000k
dc component: mag(db)= 0.000D+00 mag= 1.000D+00 phase= 0.000D+00

frequency index	frequency (hz)	fft_mag (db)	fft_mag	fft_phase (deg)
20	1.0000k	-6.0077	500.7442m	-90.4639
40	2.0000k	-86.5629	46.9737u	-37.4189
60	3.0000k	-75.0164	177.4933u	119.2220
80	4.0000k	-88.6074	37.1217u	15.4612
100	5.0000k	-77.5812	132.1107u	132.7042
120	6.0000k	-96.4409	15.0644u	-1.6789
……				
480	24.0000k	-88.2746	38.5718u	8.0802
500	25.0000k	-72.0603	249.4506u	110.3186

total harmonic distortion = 187.3653m percent

Lab4 Differential Amplifier

- **Differential Amplifier Circuit - .ac type analysis**
- **Observe AV , GB , PM**



Chapter 6

Simulation Output and Controls

Output Files Summary (HSPICE):

Output File Type	Extensi
Output Lis	.lis
DC Analysis Results	.sw#
DC Analysis Measurement Results	.ms#
AC Analysis Results	.ac#
AC Analysis Measurement Results	.ma#
Transient Analysis Results	.tr#
Transient Analysis Measurement Results	.mt#
Subcircuit Cross-Listing	.pa#
Operating Point Node Voltages (Initial Condition)	.ic

Output Statements

- **Output Commands :**
  - **.PRINT Statement : Print Numeric Analysis Results**
  - **.PLOT Statement : Generates Low Resolution Plot in .lis file**
  - **.PROBE Statement : Allows Save Output Variables Only into the Graph Date Files**
  - **.MEASURE Statement : Print Numeric Results of Measured Specifications**
- **Output Variables:**
  - **DC and Transient Analysis : Displays Individual Voltage, Current, & Power**
  - **AC Analysis : Display Real & Imag. Components of Voltage & Current.....**
  - **Element Template Analysis : Display Element-Specific Voltage, Current.....**
  - **.MEASURE : Display User-Defined Variables Defined in .MEAS Statement**

Output Variable Examples

- **DC & Transient Analysis :**
  - Nodal Voltage Output : V(1), V(3,4), V(X3.5)
  - Current Output (Voltage Source) : I(VIN), I(X1.VSRC)
  - Current Output (Element Branches) : I2(R1), I1(M1), I4(X1.M3)
- **AC Analysis :**
  - AC : V(2), VI(3), VM(5,7), VDB(OUT), IP(9), IP4(M4)
- **Element Template :**
  - mn1[vth] → LV9(mn1)
  - mn1[gds] → LX8(mn1)
  - mn1[gm] → LX7(mn1)
  - mn1 gate caps (cgs+cgd+cgb) → LX18(mn1)

R : Real
I : Imaginary
M : Magnitude
P : Phase
DB : Decibels

Parametric Statements

- **Algebraic Expressions for Output Statements:**
  - .PRINT DC V(IN) V(OUT) PAR('V(OUT)/V(IN)')
  - .PROBE AC Gain=PAR('VDB(5)-VDB(2)') Phase=PAR('VP(5)-VP(2)')
- **Other Algebraic Expressions :**
  - Parameterization : .PARAM WN=5u LN=10u VDD=5.0V
  - Algebra : .PARAM X='Y+5'
  - Functions : .PARAM Gain(IN, OUT)='V(OUT)/V(IN)'
  - Algebra in Element : R1 1 0 r='ABS(V(1)/I(M1))+10'
- **Built-In Functions :**

sin(x) cos(x) tan(x) asin(x) acos(x) atan(x) sinh(x) tanh(x) abs(x)
sqrt(x) log(x) log10(x) exp(x) db(x) min(x,y) max(x,y) power(x,y)...

Ex: .print tran Gain=PAR('log (V(out)/V(in)) ')

.PRINT & .PLOT

- **Syntax :**

.PRINT anatype ov1 <ov2 ov2...>

Note : .PLOT with same Syntax as .PRINT, Except Adding <pol1, phi1>
to set plot limit

- **Examples :**

.PRINT TRAN V(4) V(X3.3) P(M1) P(VIN) POWER PAR('V(OUT)/V(IN)')

.PRINT AC VM(4,2) VP(6) VDB(3)

.PRINT AC INOISE ONOISE VM(OUT) HD3

.PRINT DISTO HD3 HD3(R) SIM2

.PLOT DC V(2) I(VSRC) V(37,29) I1(M7) BETA=PAR('I1(Q1)/I2(Q1)')

.PLOT AC ZIN YOUT(P) S11(DB) S12(M) Z11(R)

.PLOT TRAN V(5,3) (2,5) V(8) I(VIN)

.PROBE & .GRAPH

- **.PROBE Statement :**

.PROBE Syntax : .PROBE anatype ov1 <ov2 ov2...>

Note 1 : .PROBE Statement Saves Output Variables into the Interface & Graph Data Files

Note 2 : Set .OPTION PROBE to Save Output Variables Only, Otherwise HSPICE Usually Save All Voltages & Supply Currents in Addition to Output Variables

Note 3 : Set .OPTION POST to save data into the Graph Data Files

MEASURE Statement

- **General Descriptions :**
  - **.MEASURE Statement Prints User-Defined Electrical Specifications of a Circuit and is Used Extensively in Optimization**
  - **.MEASURE Statement Provides Oscilloscope-Like Measurement Capability for either AC , DC, or Transient Analysis**
  - **Using .OPTION AUTOSTOP to Save Simulation Time when TRIG-TARG or FIND-WHEN Measure Functions are Calculated**
- **Fundamental Measurement Modes :**
  - **Rise, Fall, and Delay (TRIG-TARG)**
  - **AVG, RMS, MIN, MAX, & Peak-to-Peak (FROM-TO)**
  - **FIND-WHEN**

MEASURE : Rise, Fall, and Delay

畫個時序圖(pulse waveform), 來說明measure capability

- **Syntax :**

.MEASURE DC|AC|TRAN result_var TRIG ... TARG ... <Optimization Option>

- ***result_var* : Name Given the Measured Value in HSPICE Output**
- ***TRIG ...* : TRIG trig_var VAL=trig_value <TD=time_delay> <CROSS=n>**
  + <RISE=r_n> <FALL=f_n|LAST>
- ***TRIG ...* : TRIG AT=value**
- ***TARG ...* : TARG targ_var VAL=targ_value <TD=time_delay>**
  + <CROSS=n|LAST> <RISE=r_n|LAST> <FALL=f_n|LAST>
- ***<Optimization Option>* : <GOAL=val> <MINVAL=val> <WEIGHT=val>**

- **Example:**

```
.meas TRAN tprop trig v(in) val=2.5 rise=1 targ v(out) val=2.5 fall=1
```

MEASURE : AVG, RMS, MIN, MAX, & P-P

- **Syntax :**

```
.MEASURE  DC|AC|TRAN  result  FUNC  out_var  <FROM=val1>  <TO=val2>
+                                  <Optimization Option>
```

- *result_var* : Name Given the Measured Value in HSPICE Output
- *FUNC* : *AVG* ----- Average *MAX* ----- Maximun *PP* ---- Peak-to-Peak
  MIN ------ Minimum *RMS* ----- Root Mean Square
- *out_var* : Name of the Output Variable to be Measured
- *<Optimization Option>*: <GOAL=val> <MINVAL=val> <WEIGHT=val>

- **Example:**

```
.meas  TRAN  minval      MIN   v(1,2)     from=25ns   to=50ns
.meas  TRAN  tot_power   AVG   power      from=25ns   to=50ns
.meas  TRAN  rms_power   RMS   power
```

MEASURE : *Find & When Function*

- **Syntax :**

```
.measure  DC|AC|TRAN  result  WHEN  ... <Optimization Option>
.measure  DC|AC|TRAN  result  FIND out_var1 WHEN ...<Optimization Option>
.measure  DC|AC|TRAN  result_var  FIND out_var1 AT=val <Optimization Option>
```

- ***result* : Name Given the Measured Value in HSPICE Output**
- ***WHEN ...* : WHEN out_var2=val|out_var3 <TD=time_delay>**
  + <CROSS=n|LAST> <RISE=r_n|LAST> <FALL=f_n|LAST>
- ***<Optimization Option>* : <GOAL=val> <MINVAL=val> <WEIGHT=val>**

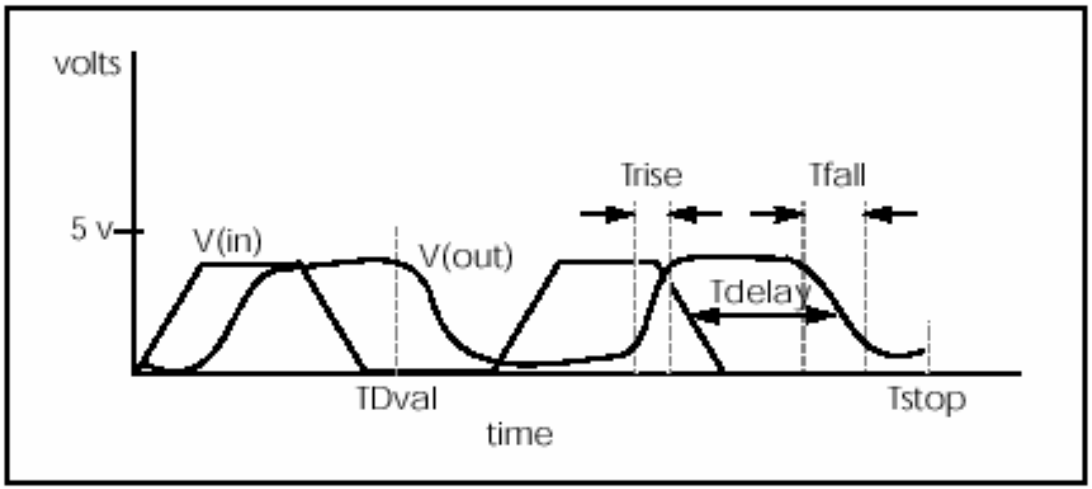
- **Example:**

```
.meas  TRAN  fifth    WHEN v(osc_out)=2.5V rise=5
.meas  TRAN  result  FIND   v(out)  WHEN v(in)=2.5V rise=1
.meas  TRAN  vmin   FIND   v(out)  AT=30ns
```

MEASURE : Application Examples

- **Rise, Fall, and Delay Calculations :**

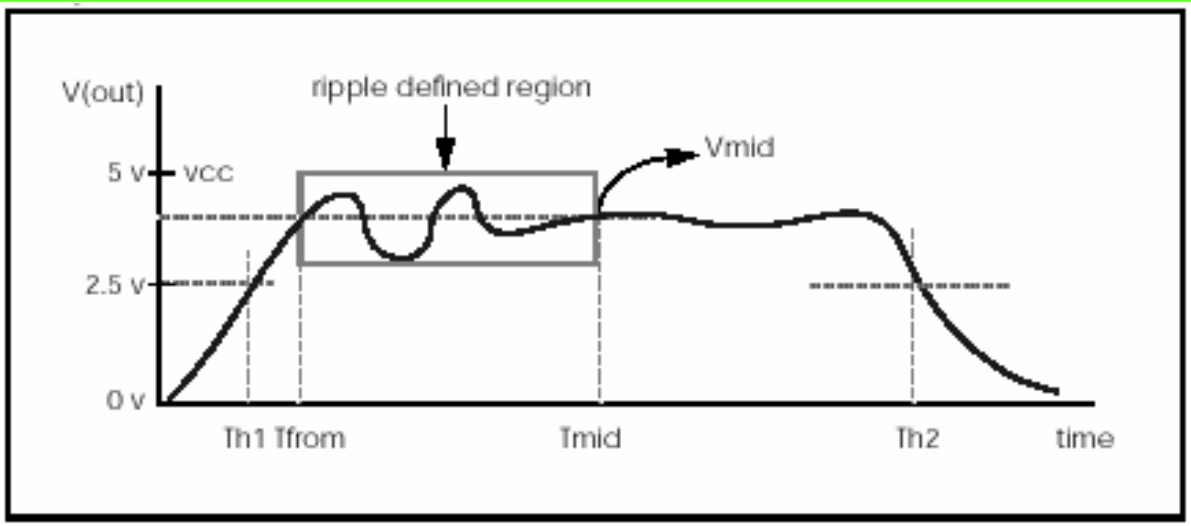
```
.meas TRAN Vmax MAX    v(out)  FROM=TDval              TO=Tstop
.meas TRAN Vmin MIN    v(out)  FROM =TDval             TO =Tstop
.meas TRAN Trise TRIG  v(out)  VAL='Vmin+0.1*Vmax'  TD=TDval   RISE=1
+                TARG  v(out)  VAL='0.9*Vmax'                   RISE=1
.meas TRAN Tfall TRIG  v(out)  VAL='0.9*Vmax'       TD=TDval   FALL=2
+                TARG  v(out)  VAL='Vmin+0.1*Vmax'              FALL=2
.meas TRAN Tdelay TRIG v(in)   VAL=2.5              TD=TDval   FALL=2
+                TARG  v(out)  VAL=2.5                         FALL=2
```



MEASURE : Application Examples(Cont.)

- **Ripple Calculation :**

```
.meas TRAN  Th1     WHEN    v(out)='0.5*v(Vdd)'   CROSS=1
.meas TRAN  Th2     WHEN    v(out)='0.5*v(Vdd)'   CROSS=2
.meas TRAN  Tmid    PARAM='(Th1+Th2)/2'
.meas TRAN  Vmid    FIND    v(out)                AT='Tmid'
.meas TRAN  Tfrom   WHEN    v(out)='Vmid'         RISE=1
.meas TRAN  Ripple  PP      v(out)                FROM='Tfrom'   TO='Tmid'
```



MEASURE : Application Examples(Cont.)

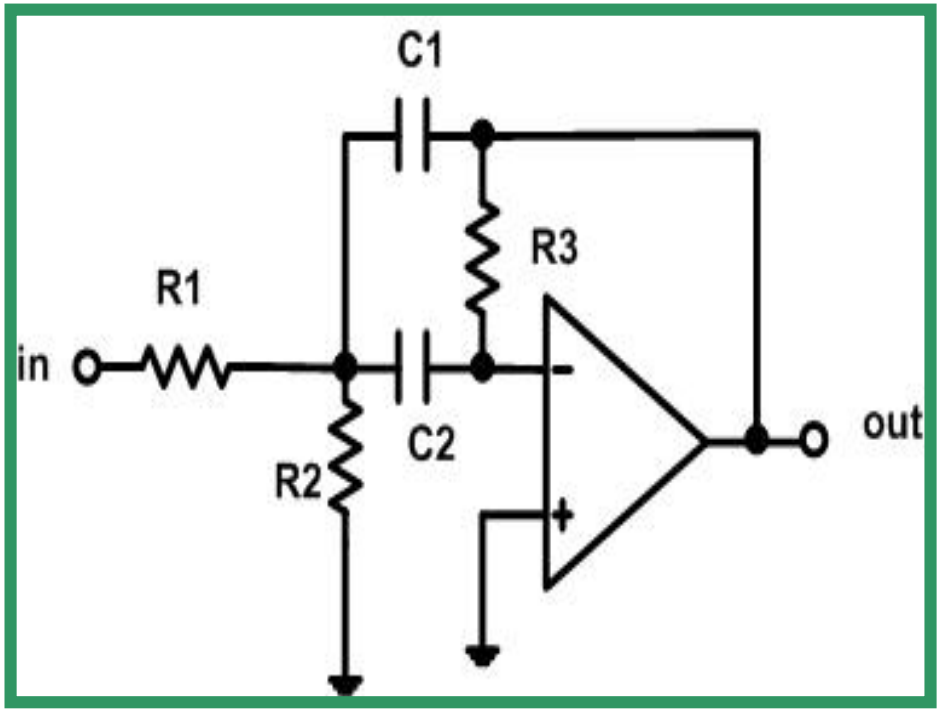
- **Unity-gain Freq, Phase margin, & DC gain(db/M):**

```
.meas  AC  unitfreq      WHEN  vdb(out)=0    FALL=1
.meas  AC  phase         FIND    vp(out)       WHEN vdb(out)=0
.meas  AC  'gain(db)'     MAX     vdb(out)
.meas  AC  'gain(mag)'    MAX     vm(out)
```

- **Bandwidth & Quality Factor (Q):**

```
.meas  AC  gainmax       MAX     vdb(out)
.meas  AC  fmax          WHEN    vdb(out)='gainmax'
.meas  AC  band          TRIG    vdb(out)    VAL='gainmax-3.0'     RISE=1
+                        TARG    vdb(out)    VAL='gainmax-3.0'     FALL=1
.meas  AC  Q_factor      PARAM='fmax/band'
```

Lab5 Buffer & High Q Bandpass Filters



Chapter 7

Elements and Device Models

Types of Elements

- **Passive Devices :**
  - ***R ---- Resistor***
  - ***C ---- Capacitor***
  - ***L ---- Inductor***
  - ***K ---- Mutual Inductor***
- **Active Devices :**
  - ***D ---- Diode***
  - ***Q ---- BJT***
  - ***J ---- JFET and MESFET***
  - ***M ---- MOSFET***
- **Other Devices :**
  - ***Subcircuit (X)***
  - ***Behavioral (E,G,H,F,B)***
  - ***Transmission Lines (T,U,O)***

R, C, L, and K Elements

- **Passive Devices Parameters :**

	Resistor	Capacitor	Inductor	Mutual Inductor
Netlist	Rxxx, n1,n2, mname, rval	Cxxx, n1,n2, mname, cval	Lxxx, n1,n2, mname, lval	Kxxx, Lyyy, Lzzz, kval
Temperature	DTEMP, TC1, TC2	DTEMP, TC1, TC2	DTEMP, TC1, TC2	
Geometric	L, M, W, SCALE	L, M, W, SCALE	M, SCALE	
Parasitics	C		R	
Initialization		IC(v)	IC(i)	

- **Examples :**

```
R1   12   17   1K   TC1=1.3e-3  TC2=-3.1e-7
C2    7    8   0.6pf   IC=5V
LSHUNT  23   51   10UH 0.01  1   IC=15.7mA
K4  Laa Lbb  0.9999
```

Diodes

TYPE	Parameters
Netlist	Dxxx, n+, n-, mname
Temperature	DTEMP
Geometric	AREA, L, W, PJ, M
Initialzation	IC, OFF

- **Syntax Examples :**

  D1 n1 n2 diode AREA=1.5 IC=0.6

- **Model Syntax:**

  .MODEL mname D <param=val>……

- **Models:**

  Level 1 nongeometric junction model – no SCALM

  Level 2 Fowler-Nordheim model – M (Insulator)S device

  Level 3 geometry juncion model ** Foundry supplied

BJT Element

- **BJT Element Parameters :**

TYPE	Parameters
Netlist	Qxxx, nc, nb, ne, ns, mname
Temperature	DTEMP
Geometric	AREA, AREAB, AREAC, M
Initialization	IC(VBE, VCE), OFF

- **BJT Syntax Examples :**

Q100 NC NB NE QPNP AREA=1.5 AREAB=2.5 AREAC=3.0 IC= 0.6, 5.0

- **BJT Model Syntax :**

.MODEL mname NPN (PNP) <param=val> .....

- **BJT Models in SBTSPICE:** ***Gummel-Poon Model***

MOSFET Element

- **MOSFET Element Syntax :**

```
Mxxx nd  ng  ns  <nb>  mname <L=val> <W=val> <AD=val> <AS=val>
+                            <PD=val> <PS=val> <NRD=val>
+                            <NRS=val>
+                            <OFF> <IC=vds,vgs,vbs> <M=val>
+                            <TEMP=val> <GEO=val> <DELVTO=val>
```

- **MOSFET Element Statement Examples:**

```
M1   24   2   0   20   MODN   L=5u    W=100u    M=4
M2    1   2   3    4   MODN        5u        100u
M3    4   5   6    8   N   L=2u W=10u AS=100P AD=100p PS=40u PD=40u
```

```
.OPTIONS   SCALE=1e-6
M1   24   2   0   20   MODN   L=5    W=100     M=4
```

MOSFET Introduction

- **MOSFET Model Overview :**
  - MOSFET Defined by : (1). MOSFET Model & Element Parameters
    (2). Two Submodel : CAPOP & ACM
  - CAPOP : Specifies MOSFET Gate Capacitance
  - ACM : Modeling of MOSFET Bulk_Source & Bulk_Drain Diodes
- **MOSFET Model Levels :**
  - Available : All the public domain spice model
  - Level = 4 or 13 : BSIM1
  - Modified BSIM1
  - Level = 5 or 39 : BSIM2
  - Level = 49/53 : BSIM3V3 ,53 for Berkerley BSIM3 code
  - Level = 8 : SBT MOS8
  - Level = 54 BSIM4

Model Statement

- **MOSFET Model Syntax :**

```
.MODEL mname  NMOS  <LEVEL=val> <name1=val1> <name2=val2>.....
.MODEL mname  PMOS  <LEVEL=val> <name1=val1> <name2=val2>.....
```

- **MOSFET Model Statement Examples:**

```
.MODEL  MODP  PMOS  LEVEL=2  VTO=-0.7  GAMMA=1.0.....
.MODEL  NCH    NMOS  LEVEL=39  TOX=2e-2  UO=600.....
```

- **Corner_LIB of Models:**

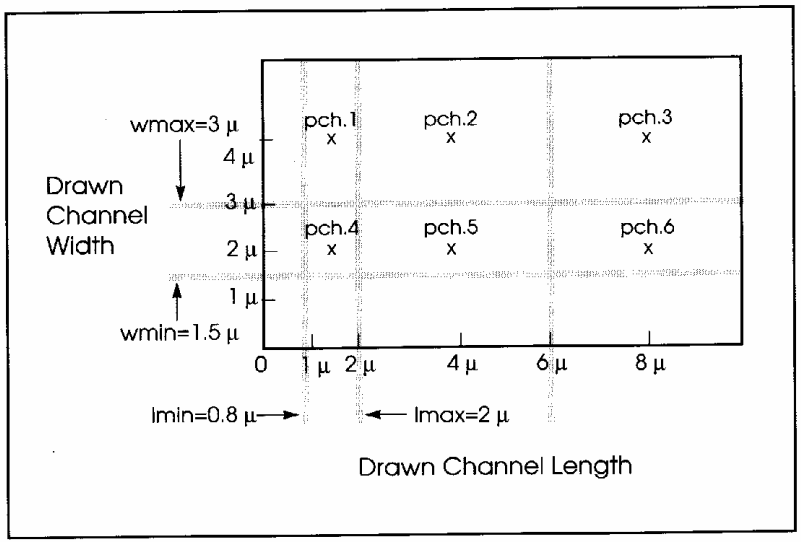
```
.LIB TT or (FF|SS|FS|SF)
 .param  toxn=0.0141 toxp=0.0148.....
.lib '~/simulation/model/cmos.l'  MOS
.ENDL TT or (FF|SS|FS|SF)
```

```
.LIB MOS
 .MODEL NMOD NMOS (LEVEL=49
+  TOXM=toxn  LD=3.4e-8 , ......)
.ENDL MOS
```

Automatic Model Selection

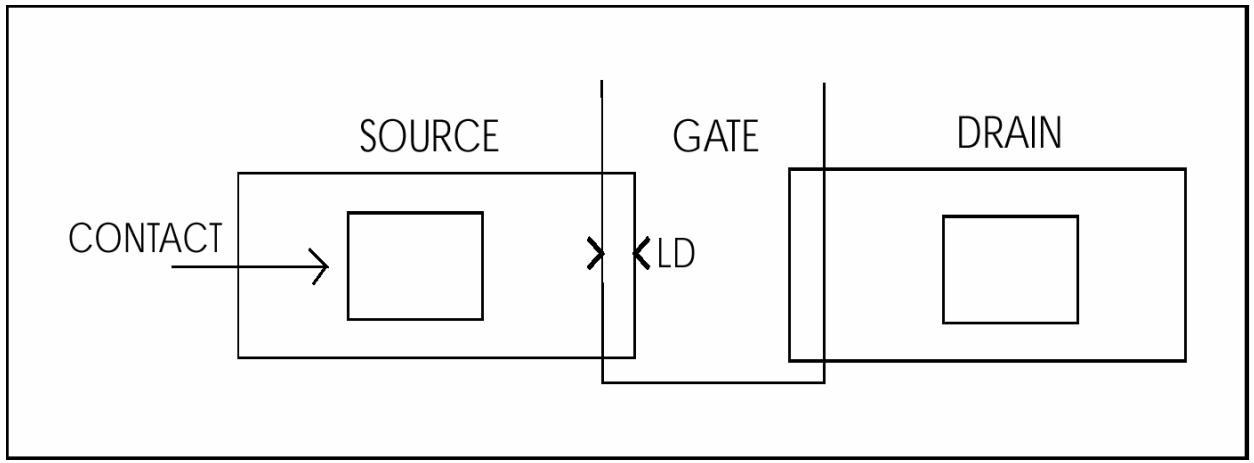
- **Automatic Model Selection :**
  - **HSPICE can Automatically Find the Proper Model for Each Transistor Size by Using Parameters, LMIN,LMAX,WMIN, & WMAX in MOSFET Models**

```
.MODEL pch.4  PMOS  WMIN=1.5u WMAX=3u LMIN=0.8u LMAX=2.0u
.MODEL pch.5  PMOS  WMIN=1.5u WMAX=3u LMIN=2.0u LMAX=6.0u
M1   1   2   3   4  pch  W=2u  L=4u  $ Automatically Select pch.5 Model
```



MOSFET Diode Model

- **MOSFET Diode Model : ACM**
  - **Area calculation Method (ACM) Parameter Allows for the Precise Control of Modeling Bulk-Source & Bulk_Drain Diodes within MOSFET Models**
- **ACM=0 MOSFET Diode: (Conventional MOSFET Structure)**

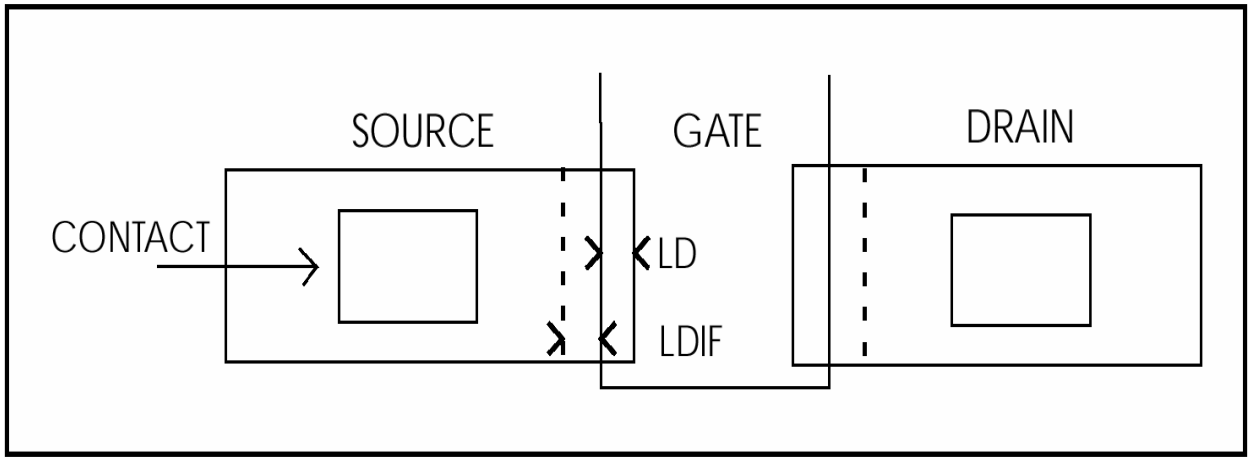


  - **ACM=0 : PN Bulk Junction of MOSFET are Modeled in the SPICE-style.**
  - **ACM=0 : Not Permit Specifications of HDIF & LDIF.**

$$ADeff = M \bullet AD \bullet WMLT^2 \bullet SCALE^2$$

MOSFET Diode Model (Cont.)

- **ACM=1 MOSFET Diode: (Not Popular)**



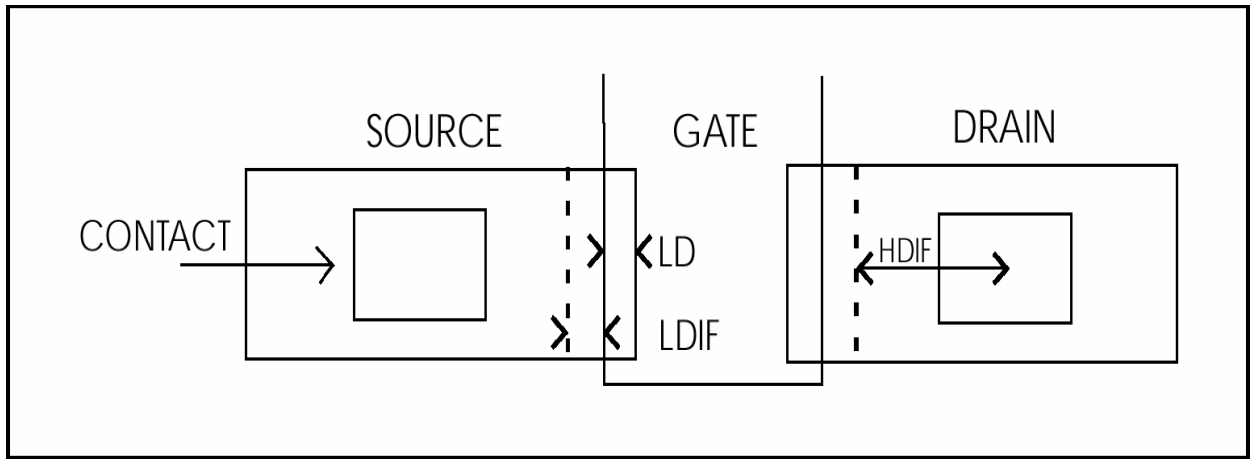
- **ACM=1 : ASPEC-style Diode Model.**
- **ACM=1 : Parameter Function of Element Width.**
- **ACM=1 : AS, AD, PS, & PD are not Used.**
- **ACM=1 : JS and CJ Differ from the SPICE_style Diode (ACM=0)**
  Ignore AS,AD, PS,PD parameters

ADeff = Weff • WMLT

Pdeff = Weff

MOSFET Diode Model (Cont.)

- **ACM=2 MOSFET Diode: (MOSFET LDD Structure)**



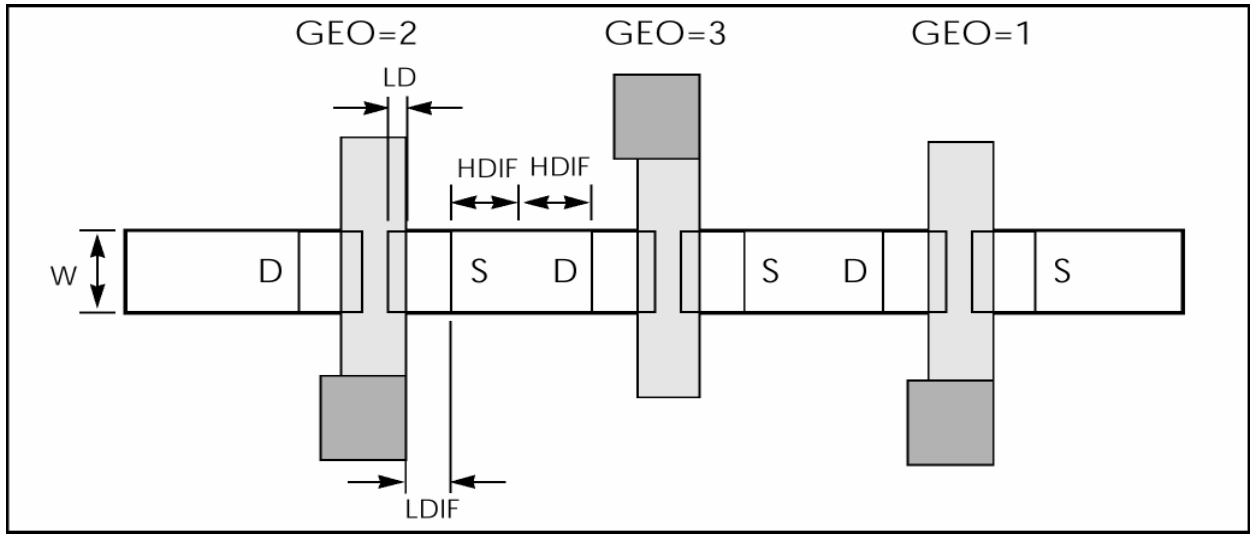
- **ACM=2 : HSPICE_Style Diode Model, Combination of ACM=0 & 1.**
- **ACM=2 : Supports both Lightly & Heavily Doped Diffusions by Settling LD, LDIF, and HDIF Parameters.**
- **ACM=2 : Effective Areas and Peripheries can be Calculations by LDIF & HDIF (i.e. AS, AD, PS, & PD can be Omitted in MOS Element Statement)**

$$ADeff = 2 \cdot HDIF \cdot Weff$$

$$PDeff = 4 \cdot HDIF + 2 \cdot Weff$$

MOSFET Diode Model (Cont.)

- **ACM=3 MOSFET Diode : (Stacked MOSFET Diode Model)**



- **ACM=3 : Extension of ACM=2 Model that Deals with Stacked Devices.**
- **ACM=3 : AS, AD, PS, & PD Calculations Depend on the Layout of the Device, which is Determined by the Value of Elememt Parameter GEO.**
- **ACM=3 : GEO=0 (Default) Indicates Drain & source are not Shared by other Devices**

Gate Capacitance Models

- **MOSFET Gate Capacitance Models:**
  - Capacitance Model Parameters can be Used with all MOSFET Model Statement.
  - Model Charge Storage Using Fixed and Nonlinear Gate Capacitance and Junction Capacitance.
  - Fixed Gate Capacitance : Gate-to-Drain, Gate-to-Source, and Gate-to-Bulk Overlap Capacitances are Represented by CGSO, CGDO, & CGBO.
  - Nonlinear Gate Capacitance : Voltage-Dependent MOS Gate Capacitance Depends on the Value of Model Parameter CAPOP.
- **MOSFET Gate Capacitance Selection :**
  - Available CAPOP Values = 0, 1, 2(General Default), 4

Higher-Order Effects

- **Geometry and Doping Effects on Vth :**
  - Short Channel Effect (Small L)
  - Narrow Channel Effect (Small W)
  - Non-Uniform Doping Effect
- **Physical Effects on Output Resistance :**
  - Channel Length Modulation (CLM)
  - Drain Induced Barrier Lowering (DIBL)
  - Substrate Current Induced Body Effects (SCBE)
- **Other Physical Effects :**
  - Channel Mobility Degradation
  - Carrier Drift Velocity
  - Bulk Charge Effect
  - Parasitic Resistance
  - Subthreshold Current

Source : HP Eesof Device Modeling Seminar, March 1996

Historical Evolution

- Can Define Three Clear Model "Generations"
- First Generation :
  - "Physical" Analytical Models
  - Geometry Coded into the Model Equations
  - Level 1, Level 2, & Level 3
- Second Generation :
  - Shift in Emphasis to Circuit Simulation
  - Extensive Mathematical Conditioning
  - Individual Device Parameters & Separate Geometry Parameter
  - Shift "Action" to Parameter Extraction (Quality of Final Model is Heavily Dependent on Parameter Extraction)
  - BSIM1, Modified BSIM1, BSIM2

Source : IEEE 1997 CICC Educational Sessions E3.3

Historical Evolution (Cont.)

- **Third Generation :**
  - **"Original Intent" was a Return to Simplicity**
  - **Scalable MOSFET model**
  - **1-st derivative is continuous**
  - **Attempt to Re-Introduce a Physical Basis While Maintaining "Mathematical Fitness"**
  - **BSIM3, MOS-8, Other ???**

Source : IEEE 1997 CICC Educational Sessions E3.3

Overview of Most Popular MOSFET Models

- **UCB Level 1 : (Level = 1)**
  - Shichman-Hodges Model (1968)
  - Simple Physical Model, Applicable to L> 10um with Uniform Doping
  - Not Precise Enough for Accurate Simulation
  - Use only for Quick, Approximate Analysis of Circuit Performance
- **UCB Level 2 : (Level = 2)**
  - Physical/Semi-Empirical Model
  - Applicable to Long Channel Device (~ 10 um)
  - Advanced Version of Level 1 which Includes Additional Physical Effects
  - Can Use either Electrical or Process Related Parameters

SPICE : Simulation Program with Integrated Circuit Emphasis

UCB : University of California at Berkeley

Overview of Most Popular MOSFET Models(Cont.)

- **UCB Level 3 : (Level = 3)**
  - Semi-Empirical Model Model (1979)
  - Applicable to Long Channel Device (~ 2um)
  - Includes Some New Physical Effects (DIBL, Mobility Degradation by Lateral Field)
  - Very successful Model for Digital Design (Simple & Relatively Efficient)
- **BSIM : (Level = 13)**
  - First of the "Second Generation" Model (1985)
  - Applicable to Short Channel Device with L~ 1.0um
  - Emphasis on Mathematical Conditioning of Circuit Simulation
  - Empirical Approach to Small Geometry Effects

BSIM : Berkeley Short-Channel IGFET Model

Overview of Most Popular MOSFET Models (Cont.)

- **Modified BSIM1 LEVEL 28 :**
  - Enhanced Version of BSIM 1, But Addressed most of the Noted Shortcomings
  - Empirical Model Structure --> Heavy Reliance on Parameter Extraction for Final Model Quality
  - Applicable to Deep Submicron Devices (~ 0.3 - 0.5um)
  - Suitable for Analog Circuit Design
- **BSIM 2 : (HSPICE Level = 39)**
  - "Upgraded" Version of BISM 1 (1990)
  - Applicable to Devices with (L~ 0.2um)
  - Drain Current Model has Better Accuracy and Better Convergence Behavior
  - Covers the Device Physics of BSIM 1 and Adds Further Effects on Short Channel Devices

Overview of Most Popular MOSFET Models (Cont.)

- **BSIM 3v3 : (Level = 49)**
  - Newly Advanced Submicron MOSFET Model (Third Generation)
  - Latest Physics-Based, Deep-Submicron Model (BSIM3v3.1)
  - Use a Single Equation for a Device Property for All Regions of Device Operation, But Slow & Inefficient Behavior During Circuit Simulation
  - Use of Smoothing Functions Greatly Improves Final Results
  - Attempt to Impose "Global" Capability Creates a Number of Problems

  In HSPICE, level=53 is BSIM 3V3 Berkeley compliance
  , level=49 is Avanti implemented BSIM3V3

- **MOS8 Model : (SBTSPICE Level = 8)**
  - Developed by SBT
  - 4-Terminals NQS charge-conservation model
  - Physical-oriented scalable model

Overview of Most Popular MOSFET Models (Cont.)

- **EKV Model :**
  - Developed at Swiss Federal Institute of Technology in Lausanne (EPFL)
  - A Newly "Candidate" Model for Future Use
  - Description of Small Geometry Effects is Currently Being Improved
  - Developed for Low Power Analog Circuit Design
  - Fresh Approach to FET Modeling
    - Use Substrate (not Source) as Reference
    - Simpler to Model FET as a Bi-Directional Element
    - Can Treat Pinch-Off and Weak Inversion as the same Physical Phenomenon
  - First "Re-Thinking" of Analytical FET Modeling Since Early 1960s.

Source : IEEE 1997 CICC Educational Sessions E3.3

MOSFET Model Comparison

- **Model Equation Evaluation Criteria :** **(Ref: HSPICE User Manual 1996, Vol._II)**
  - Potential for Good Fit to Data
  - Ease of Fitting to Data
  - Robustness and Convergence Properties
  - Behavior Follows Actual Devices in All Circuit Conditions
  - Ability to Simulate Process Variation
  - Gate Capacitance Modeling
- **General Comments :**
  - Level 3 for Large Digital Design
  - HSPICE Level 28 for Detailed Analog/Low Power Digital
  - BSIM 3v3 & MOS Model 9 for Deep Submicron Devices
  - All While Keeping up with New Models

Source : IEEE 1997 CICC Educational Sessions E3.3

Chapter 8

Control Options
&
Convergence

Control Options : Output Format

- **Output Format : General (LIST, NODE, ACCT, OPTS, NOMOD)**
  - **.OPTION LIST : Produces an Element Summary Listing of the Data to be Printed. (Useful in Diagnosing Topology Related NonConvergence Problems)**
  - **.OPTION NODE : Prints a Node Connection Table. (Useful in Diagnosing Topology Related NonConvergence Problems)**
  - **.OPTION ACCT : Reports Job Accounting and Run-Time Statistics at the End of Output Listing. (Useful in Observing Simulation Efficient)**
  - **.OPTION OPTS : Prints the Current Settings of All Control Options.**
  - **.OPTION NOMOD : Suppress the printout of Model Parameters (Useful in Decreasing Size of Simulation Listing Files)**

Simulation Controls & Convergence

- **Definition of "Convergence" :**
  - The Ability to Obtain a Solution to a Set of Circuit Equations Within a Given Tolerance Criteria & Specified Iteration Loop Limitations.
  - The Designer Specifies a Relative & Absolute Accuracy for the Circuits Solution and the Simulator Iteration Algorithm Attempts to Converge onto a Solution that is within these Set Tolerance
- **Error Messages for "NonConvergence" :**
  - " No Convergence in Operating Point (or DC Sweep)"
  - " Internal TimeStep is Too Small in Transient Analysis"
- **Possible Causes of "NonConvergence" :**
  - Circuit Reasons : (1). Incomplete Netlist; (2). Feedback; (3). Parasitics
  - Model Problems : (1). Negative Conductance (2). Model Discontinuity
  - Simulation Options : (1). Tolerances; (2). Iteration Algorithm

Accuracy Tolerances

- HSPICE uses Accuracy Tolerances specifications to help assure convergence by determining whether or not to exit the convergence loop.
- Absolute and relative accuracy tolerances:

Type	Option	Default
Nodel Voltage Tolerance	ABSV	50uV
	RELVDC	0.001(0.1%)
Current Element Tolerances	ABSI	1nA
	RELI	0.01(%)
	ABSMOS	1uA
	MELMOS	0.05(5%)

- ABSVDC&ABSI set the floor value below which values are ignored.
- Values above the floor use the relative tolerances of RELVDC & RELI.
- ABSMOS & RELMOS are the tolerances for MOSFET Drain Current.
- Number of iterations required is directly affected by accuracy setting.

AutoConvergence for DC Operating Point Analysis

- **AutoConvergence Process:**
  - If Convergence is not Achieved in the Number of Iteration set by ITL1, HSPICE Initiates an AutoConvergence Process, in which it Manipulates DCON, GRAMP, and GMINDC, as well as CONVERGENCE in some Cases.
  - ITL1= *x* : Set the Maximum DC Iteration Limit, Default=100. Increasing Values as High as 400 Have Resulted in Convergence for Certain Large Circuits with Feedback, such as OP Amp & Sense Amplifiers.
  - GMINDC= *x* : A Conductance that is Placed in Parallel with All PN Junction and All MOSFET Nodes for DC Analysis. It is Important in Stabilizing the Circuit During DC Operating Point Analysis.
  - CONVERGE=X : INVOKES Different Methods for Solving NonConvergence Problems.
  - .OPTIONs OFF =: turn off active device for easing DC calculation.

Steps for Solving DC Operating Point NonConvergence

- **(1). Check Topology :**
  - Set .OPTIONS NODE to Get a Nodes Cross Reference Listing if You are in Doubt
  - Check if All PMOS Substrates Connected to VDD or Positive Supplies ?
  - Check if All NMOS Substrates Connected to GND or Negative Supplies ?
  - Check if All Vertical NPN Substrates Connected to GND or Negative Supplies ?
  - Check if All Lateral PNP Substrates Connected to Negative Supplies ?
  - Check if All Latches Have Either an OFF Transistor or a .NODESET or an .IC on one side ?
  - Check if All Series Capacitors Have a Parallel Resistance, or is .OPTION DCSTEP Set ?

Steps for Solving DC Operating Point NonConvergence (Cont.)

- **(2). Check Your .MODEL Statements :**
  - Be Sure to Check your Model Parameter Units.
  - Check if Your MOS Models had Subshreshold Parameter Set ? (NFS= 1e11 for HSPICE Level 1,2 &3 and N0=1 for HSPICE BSIM1,2,3 Models & Level 28)
  - Use MOS ACM=1, ACM=2, or ACM=3 Source and Drain Diode Calculation to Automatically Generate Parasitics.
- **(3). General Remarks :**
  - Circuits that Converge Individually and Fail when Combined are Almost Guaranteed to have a Modeling Problem.
  - Schmitt Triggers are Unpredictable for DC Sweep and Sometimes for Operating Point for the same Reasons Oscillators and Flip-Flops are. Use Slow Transient.

Steps for Solving DC Operating Point NonConvergence (Cont.)

- **(3). General Remarks (Cont.):**
  - **Open Loop OP Amp have High Gain, which can Lead to Difficulties in Converging. ==> Start OP Amp in Unity-Gain Configuration and Open them Up in Transient Analysis with a Voltage-Variable Resistor or a Resistor with a Large AC Value for AC Analysis.**
- **(4). Check Your Options :**
  - **Remove All Convergence-Related Options and Try First with No Special Options Setting.**
  - **Check NonConvergence Diagnostic Table for NonConvergence Nodes. Look up NonConvergence Nodes in the Circuit Schematic. They are Generally Latches, Schmitt Triggers or Oscillating Nodes.**
  - **SCALE and SCALM Scaling Options Have a Significant Effect on the Element and Model Parameters Values. Be Careful with Units.**

Solutions for Some *Typical* NonConvergence *Circuits*

- **Poor Initial Conditions :**
  - **Multistable Circuits Need State Information to Guide the DC Solution. For Example, You must Initialize Ring Oscillator or Flip-Flops Circuits Using the .IC Statement.**
- **Inappropriate Model Parameters :**
  - **It is Possible to Create a Discontinuous Ids or Capacitance Model by Imposing Nonphysical Model Parameters**
  - **Discontinuities Most Exits at the Intersection of the Linear & Saturation Regions**
- **PN Junctions (Diodes, MOSFETs, BJTs) :**
  - **PN Junctions Found in Diodes, BJTs, and MOSFET Models can Exhibit NonConvergence Behavior in Both DC and Transient Analysis.==> Options GMINDC and GMIN Automatically Parallel Every PN Junction with a Conductance.**

Solutions for Some Typical NonConvergence Circuits (Cont.)

- **Rapid Transitions in DC Sweep :**
  - Increase the ITL2 , which is the Number for Iteration Allowed at Each Steps in DC Sweep Analysis, Value to 100, 200 or more. Default=50. (ITL1 is for the Initial Operating Point)
- **DC Options for High Power Circuits :**
  - For High Power Bipolar Transistors, Set Options ABSI up from Default 1 nA to Perhaps 10 mA. Also, Minimum Voltage Tolerance Should be Raised from 50 uV to 10mV.
  - If the Accuracy Given by Tight Tolerance is Necessary, Be Sure to Increase Option ILT1 to 300~~400.
- **DC Options for Op Amp and Comparator :**
  - High Gain Op Amp and Comparator can Cause Convergence Problems, as well as Possible Violations of KCL Law. It is Sometimes Necessary to Tighten the Options RELI to 0.005 and ABSI to 1 pA

Speed, Accuracy and Convergence for Tran ananlysis

- **simulation Speed:**
  - **.option fast : set additional options to increase simulation speed with little loss of accuracy.**
  - **.option autostop: terminates the simulation when all .measure statements have completed.**
- **Simulation Accuracy:**
  - **The control options and their default settings to Maximize accuracy are:**

DVDT=4 LVLTIM=1 RMAX=5 SLOPETOL=0.75

FT=FS=0.25 BYPASS=1 BYTOL=MBYPASS*VNTOL=0.10M

Speed, Accuracy and Convergence for Tran analysis (cont.)

- **Timestep control for accuracy:**
  - **DVDT control option selects the timestep control algorithm.**
  - **DELMAX specifies the maximum allowed timestep size. For circuits contain oscillators or ideal delay elements, DELMAX should be set to one-hundredth of the period or less.**
  - **ACCURATE control option tightens the simulation options to give the most accurate set of simulation algorithms and tolerances.**
- **Models and accuracy:**
  - **MOS Model parameter ACM that calculate defaults for source & drain junction parasitics.**
  - **MOS Model parameter CAPOP=4 that models the most advanced charge conservation.non-reciprocal gate capacitance.**

Guidelines for choosing accuracy options

- Use the ACCURATE option for :
  - Analog or mixed signal circuits
  - Circuits with long time constants, such as rc networks
  - Circuits with ground bounce
- Use the default options (DVDT=4) for:
  - Digital circuits
  - Digital cell characterization
  - Circuits with fast moving edges (short rise and fall times)
- use ideal delay elements, use one of the following:
  - ACCURATE
  - DVDT=3
  - DVDT=4 and set DELMAX to a value small than the minimum pulse width if the minimum pulse width of any signal is less than the Mim. Ideal Delay

Numerical Integration Algorithm Controls

- **Types of Numerical Integration Methods :**
  - Trapezoidal Algorithm (Default in HSPICE) — .OPTION METHOD = TRAP
  - GEAR Algorithm — .OPTION METHOD = GEAR
- **Trapezoidal Algorithm :**
  - Highest Accuracy
  - Lowest Simulation Time
  - Best for CMOS Digital Circuits
- **GEAR Algorithm :**
  - Most Stable
  - Highly Analog, Fast Moving Edges
- **One Limitation of Trapezoidal Algorithm :**
  - It can Results in Unexpected Computational Oscillation. (Also Produces an Usually Long Simulation Time)
  - For Circuits are Inductive in Nature, such as Switching Regulator, Use GEAR Algorithm. (Circuit NonConvergent with TRAP will often Converge with GEAR)

Timestep control algorithm

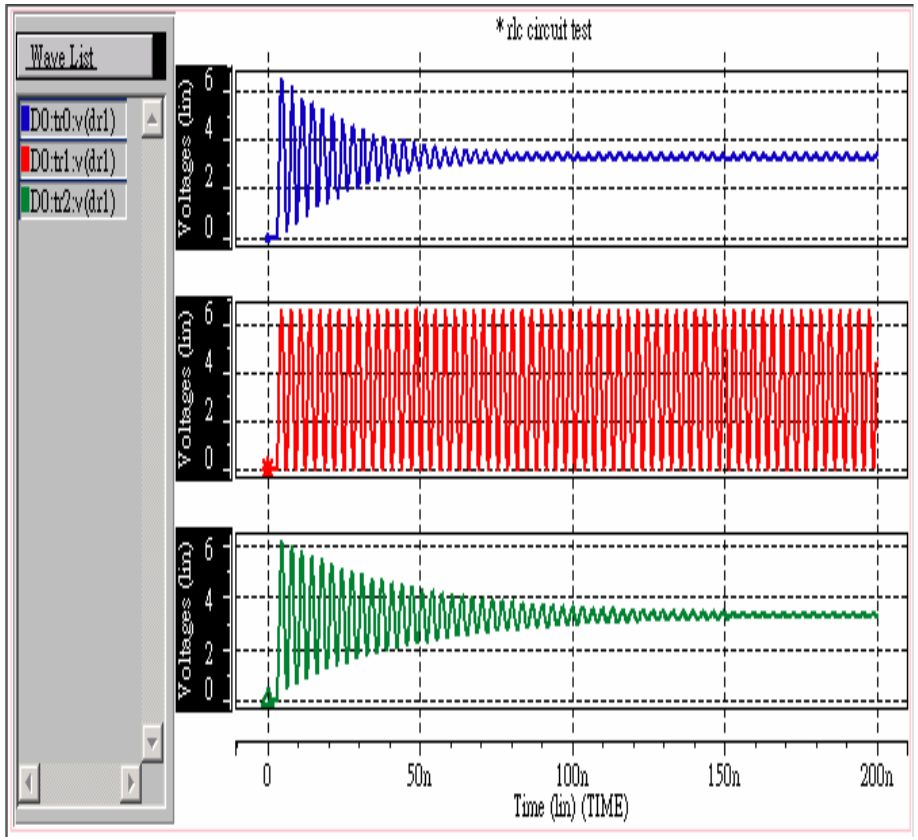
- **Dynamic timestep control algorithm:**
  - Dynamically changing timestep increase the accuracy of simulation and reduced the simulation time by varying the value of the timestep over the transient ananlysis sweep depending upon the stability of the output.
- **Operation principles of Dynamic Timestep Control Algorithm:**
  - decrease the Timestep value when nodal voltage are changing quickly.
  - Increase the Timestep value when internal nodal voltage are Stable.
- **Selection of Timestep Control algorithm:**
  - LVLTIM= 1 (default) uses the DVDT timestep algorithm.
  - LVLTIM=2 LVLTIM = 2 uses the timestep algorithm for local truncation error.
  - LVLTIM=3 uses the DVDT timestep algorithm with timestep reversal.
  - To use the GEAR method of numerical integration and linearization, select LVLTIM = 2.
  - To use the TRAP linearization algorithm, select LVLTIM = 1 or 3. Using LVLTIM = 1 (DVDT option) is the default, and helps avoid internal timestep too small non-convergence.

Timestep Control Algorithms

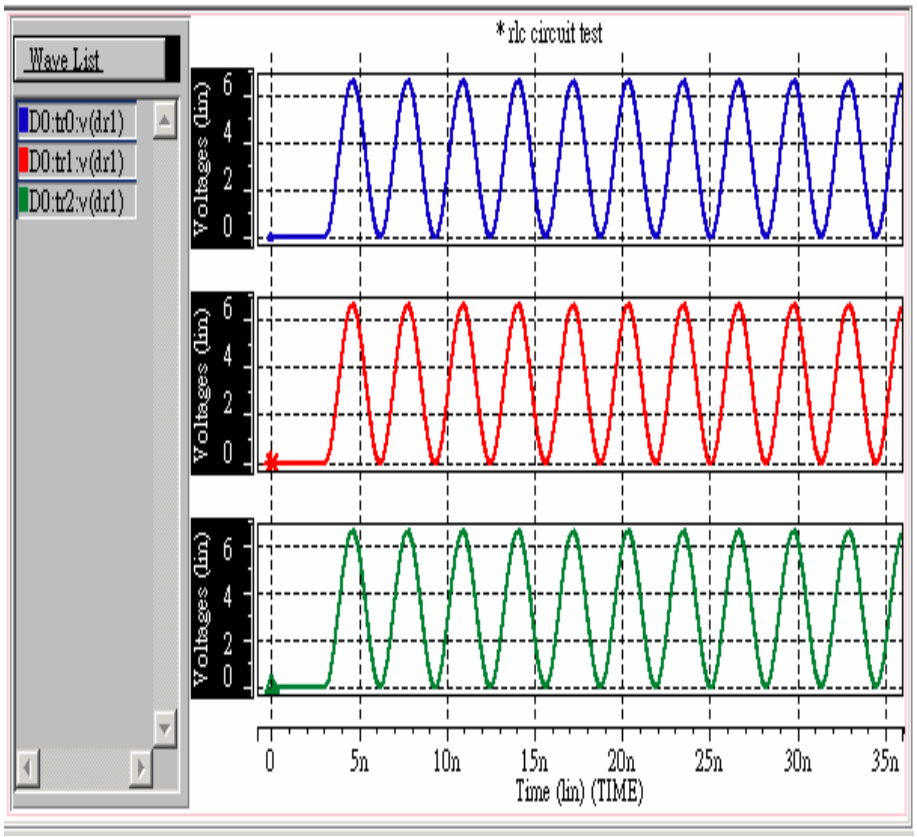
- **Types of Dynamic Timestep Control Algorithm :**
  - **Iteration Count (Simplest):**
    - If Iterations Required to Converge > MAX, Decrease the Timestep
    - If Iterations Required to Converge < MIN, Increase the Timestep
  - **Local Truncation Error (LTE) :**
    - Use a Taylor Series Approximation to Calculate Next Timestep
    - Timestep is Reduced if Actual Error is > Predicted Error
  - **DVDT (default):**
    - Based on the rate of change of nodal voltages
- **Timestep Control Algorithm vs. Numerical Integration Algorithm :**
  - For GEAR is Selected => Defaults to Truncation Timestep Algorithm;
    For TRAP is Selected => Defaults to DVDT Algorithm

Lab6 RLC circuit test

.option accuracy



Reduce maximum timestep

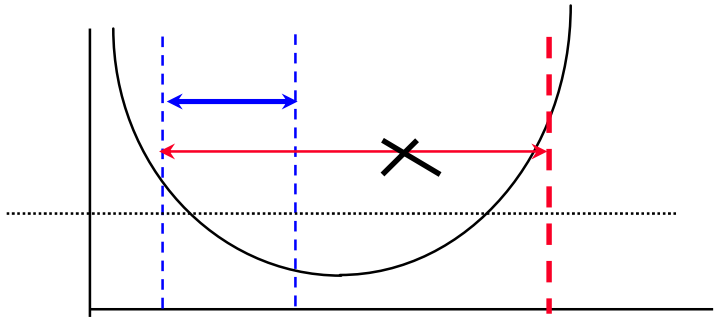


CHAPTER 9

Optimization
&
Statistical Analysis

Optimization

- **Circuit Level Goal Optimization:**
  - A procedure for automatic searching instance parameters to meet design goal
  - Can be applied for both .DC , .AC and .TRAN analysis
  - Optimization implemented in SBTSPICE can optimize one goal
  - Optimization implemented in HSPICE can optimize multi-goal circuit parameter/device model parameter
  - The parameter searching range must differentiate the optimization goal



Optimization Preliminaries

- Circuit Topology Including Elements and Models
- List of Element to be Optimized
  - Initial Guess, Minimum, Maximum
- .Measure Statements for Evaluating Results
  - Circuit Performance Goals
  - Selection of Independent or Dependent Variables Measurement Region
- Specify Optimizer Model

Optimization Syntax : General Form

- **Variable Parameters and Components :**

```
.PARAM  parameter = OPTxxx (init, min, max)
```

- **Optimizer Model Statement :**

```
.MODEL  method_namd   OPT   <Parameter = val .....>
```

- **Analysis Statement Syntax :**

```
.DC|AC|TRAN .....<DATA=filement >  SWEEP   OPTIMIZE = OPTxxx
+                          Results = meas_name    MODEL = method_name
```

- **Measure Statement Syntax :**

```
.MEASURE meas_name .....<GOAL=val>  <MINVAL=val>
```

Example1

```
* rc network optimization
.ic 1 1
r1 1 0 rx
c1 1 0 cx
.model opt1 opt
.option post
.param rx=optrc(.5,1e-2,1e+2)
+       cx=optrc(.5,1e-2,1e+2)

.tran .1 2 sweep optimize = optrc results = trc,prc model =opt1
.measure tran trc trig at=0 targ v(1) val=.3679 fall=1 goal =1s
.measure tran prc rms p(r1) goal=50mw
.end
```

Specify parameter range

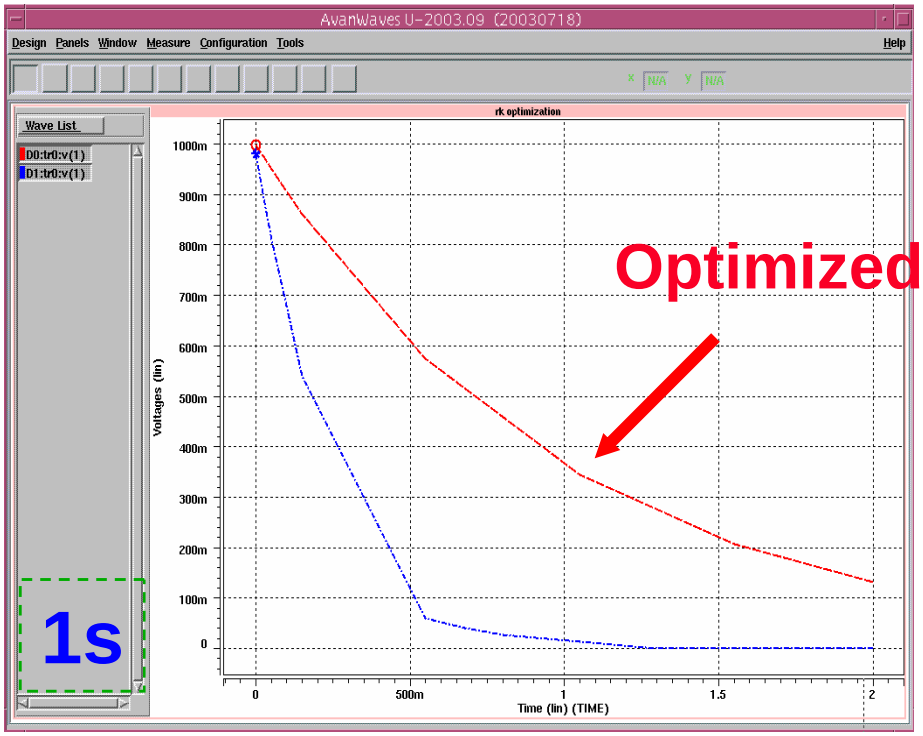
Analysis type and optimization algorithm

Optimization goal by measure command

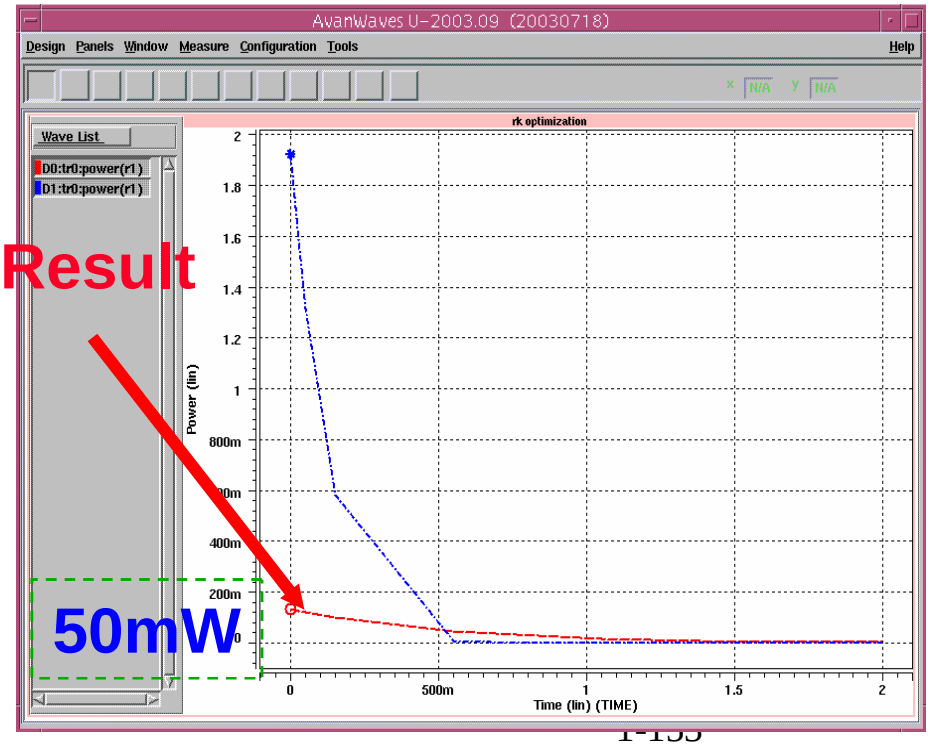
Example1 (cont.)

Design target

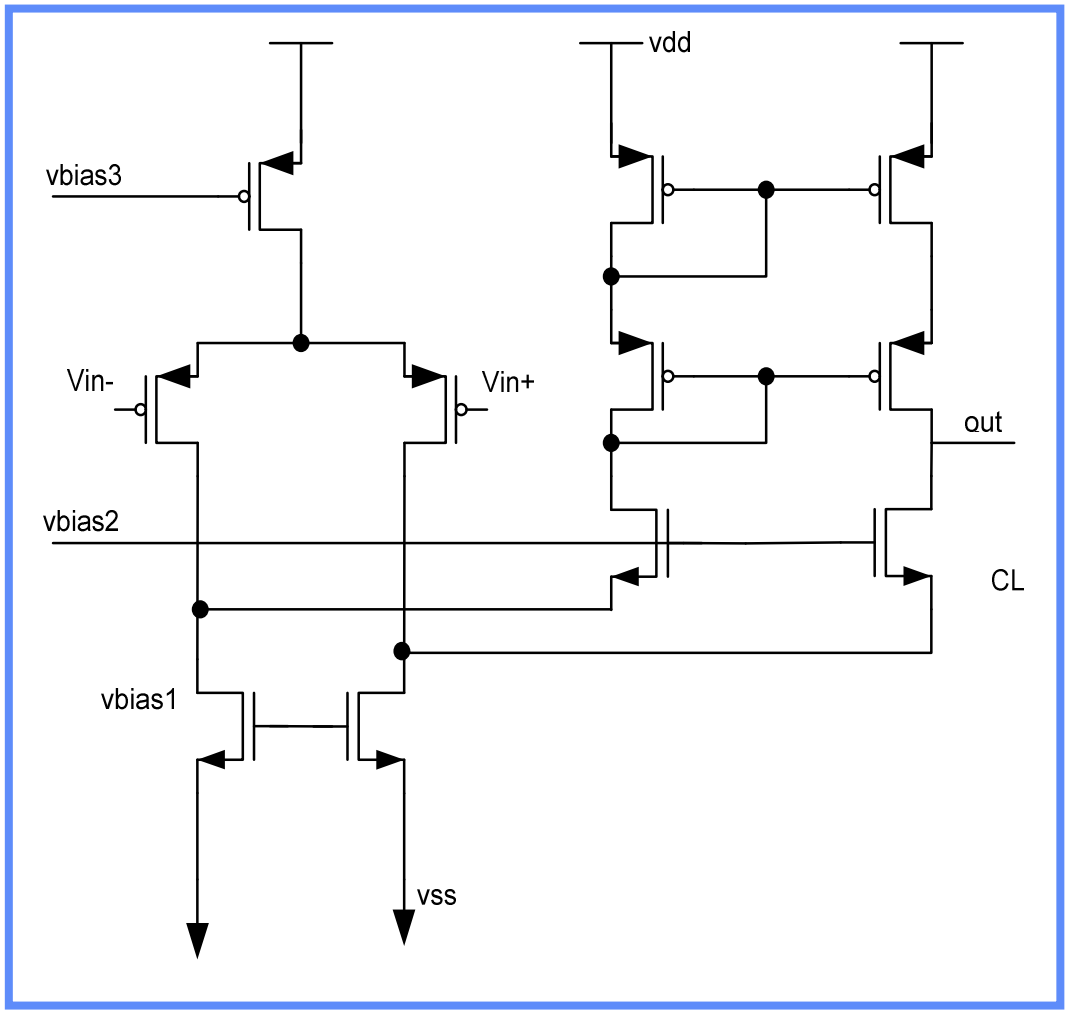
- 1s time constant
- 50mW rms power dissipation through the resistor



Optimized Result



Example 2
Folded cascode OPAMP



* Target specification :
* CL = 1pF, Av>70dB,
* GB=100MHz ,PM>60
* SR = 2 V/us , Pdiss < 10mW ,
* with level2 process

Components &Models & Subckts

```
** P COMOS folded-cascode opa
** COMPONENTS
  xopa ina inb out opa
* Models & Subckts
  .protect
 .lib 'mm0355v.l' TT
 .unprotect
** subcircuit operator amplifier
 .subckt opa va vb vout
  ** differential amp
  M9 net3 net0 vdd! vdd! pch W=w1 L=1u
  M1 net4 va net3 net3 pch W=w2 L=1u
  M2 net5 vb net3 net3 pch W=w2 L=1u
  M10 net4 net2 0 0 nch W=w3 L=1u
  M11 net5 net2 0 0 nch W=w3 L=1u
```

Subckts

```
** wide swing Cascode
  M3 net7 net6 vdd! vdd! pch W=w4 L=1u
  M4 net6 net6 vdd! vdd! pch W=w4 L=1u
  M5 net8 net8 net7 net7 pch W=w4 L=1u
  M6 vout net8 net6 net6 pch W=w4 L=1u
  M7 net8 net1 net5 0 nch W=w5 L=1u
  M8 vout net1 net4 0 nch W=w5 L=1u
  ** capacitator load
  Cl vout 0 1p
  vbias1  net0  0  bias1
  vbias2  net1  0  bias2
  vbias3  net2  0  bias3
 .ends
```

Sources & Controls

```
* Sources
  .GLOBAL vdd! vss! vbias1 vbias2 vbias3
  vdd vdd! 0 3.3
  vss vss! 0  0
  vr   ina 0 dc 1.65
  vin  inb ina ac 1
** Controls
  .op
  .option post relv=1e-3 relvar=0.1 nomod acct post=2
```

Optimizations 1

***** OPTIMIZATION**

****1 Variable Parameters and Components :***

```
.param bias1=opt1(2.4838,2.3,2.7)
+      bias2=opt1(1.543,1.3,1.8)
+      bias3=opt1(0.6852,0.5,1)
+      w1   =opt1(30u,5u,150u)
+      w2   =opt1(10u,2u,150u)
+      w3   =opt1(18u,1u,150u)
+      w4   =opt1(27u,1u,150u)
+      w5   =opt1(10u,1u,150u)
```

****2 Optimizer Model Statement :***

```
.model opt opt close=100 itropt=30
```

Optimizations 2

*3 Analysis Statement Syntax :

```
.ac dec 10 1k 1000Meg  sweep optimize=opt1 result =
 + Amx,unity_gain_freq,phase_margin  model=opt
```

*4 Measure Statement Syntax :

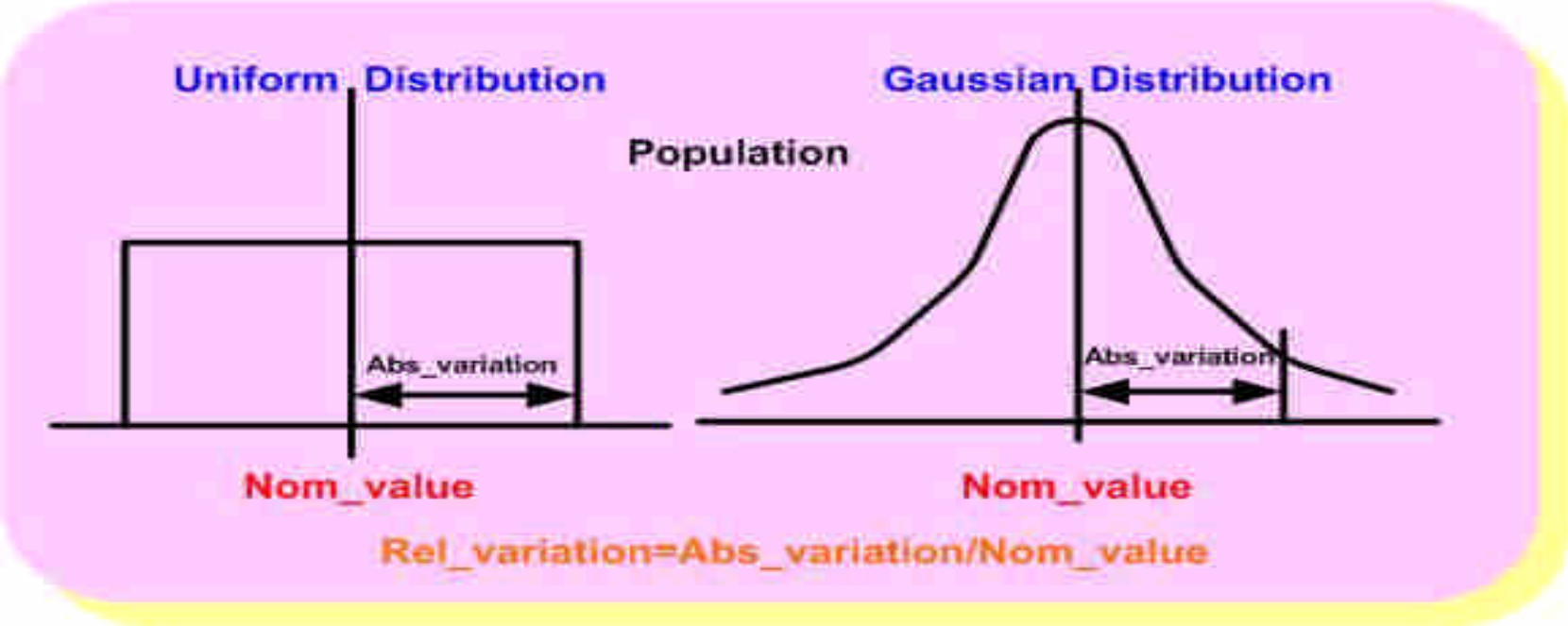
```
.probe vdb(out) vp(out)
.meas Amx max vdb(out) goal = 70
.meas unity_gain_freq when vdb(out)=0 goal=230Meg
.meas ac phase_margin find=par'180+vp(out)' when
+vdb(out)=0 goal=50
.end
```

Measure

```
$DATA1 SOURCE='HSPICE' VERSION='U-2003.09      '
.TITLE '** pmos folded-cascode opa'
 index          bias1          bias2          bias3
                w1             w2             w3
                w4             w5             amx
                unity_gain_freq  phase_margin     temper
                alter#
  28.0000         2.3200         1.8000         0.7785
                1.041e-04      9.773e-05      4.521e-05
                4.166e-05      5.313e-06       70.0928
                2.300e+08       50.0003        25.0000
                  1.0000
```

Monte Carlo Analysis

- a random number generator
- Gaussian 、 Uniform 、 Random Limit Parameter Distribution
- Relative 、Absolute 、 Bimodal 、min/max variation variation



Monte Carlo Distribution

Monte Carlo Setup

◆.PARAM statement

.PARAM xx=AUNIF(nominal_val, abs_variation <, multiplier>)

.PARAM xx=UNIF(nominal_val, rel_variation <, multiplier>)

.PARAM xx=GAUSS(nominal_val, rel_variation, sigma <, multiplier>)

.PARAM xx=AGAUSS(nominal_val, abs_variation, sigma <, multiplier>)

.PARAM xx=LIMIT(nominal_val, abs_variation)

where:

- ***LIMIT Random limit distribution function*** using absolute variation,+/- abs_variation is added to nominal_val based on whether the random outcome of a -1 to 1 distribution is greater or less than 0.
- ***multiplier*** If not specified, the default is 1. The calculation is repeated this many times and the largest deviation is saved.

Monte Carlo Setup(Cont.)

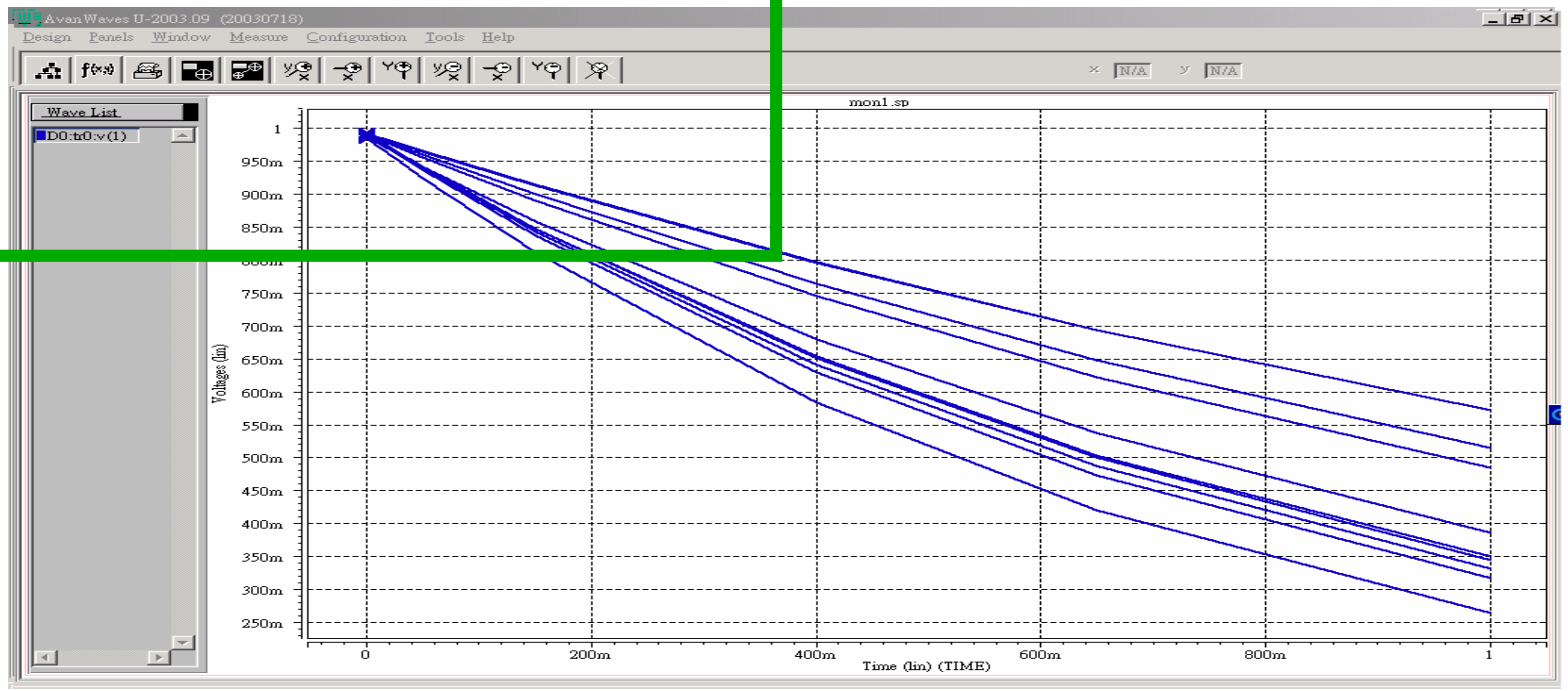
◆.DC, .AC, or .TRAN analysis

■Operating Point .DC MONTE=val
■DC Sweep .DC vin 1 5 .25 SWEEP MONTE=val
■AC Sweep .AC dec 10 100 10meg SWEEP MONTE=val
■TRAN Sweep .TRAN 1n 10n SWEEP MONTE=val

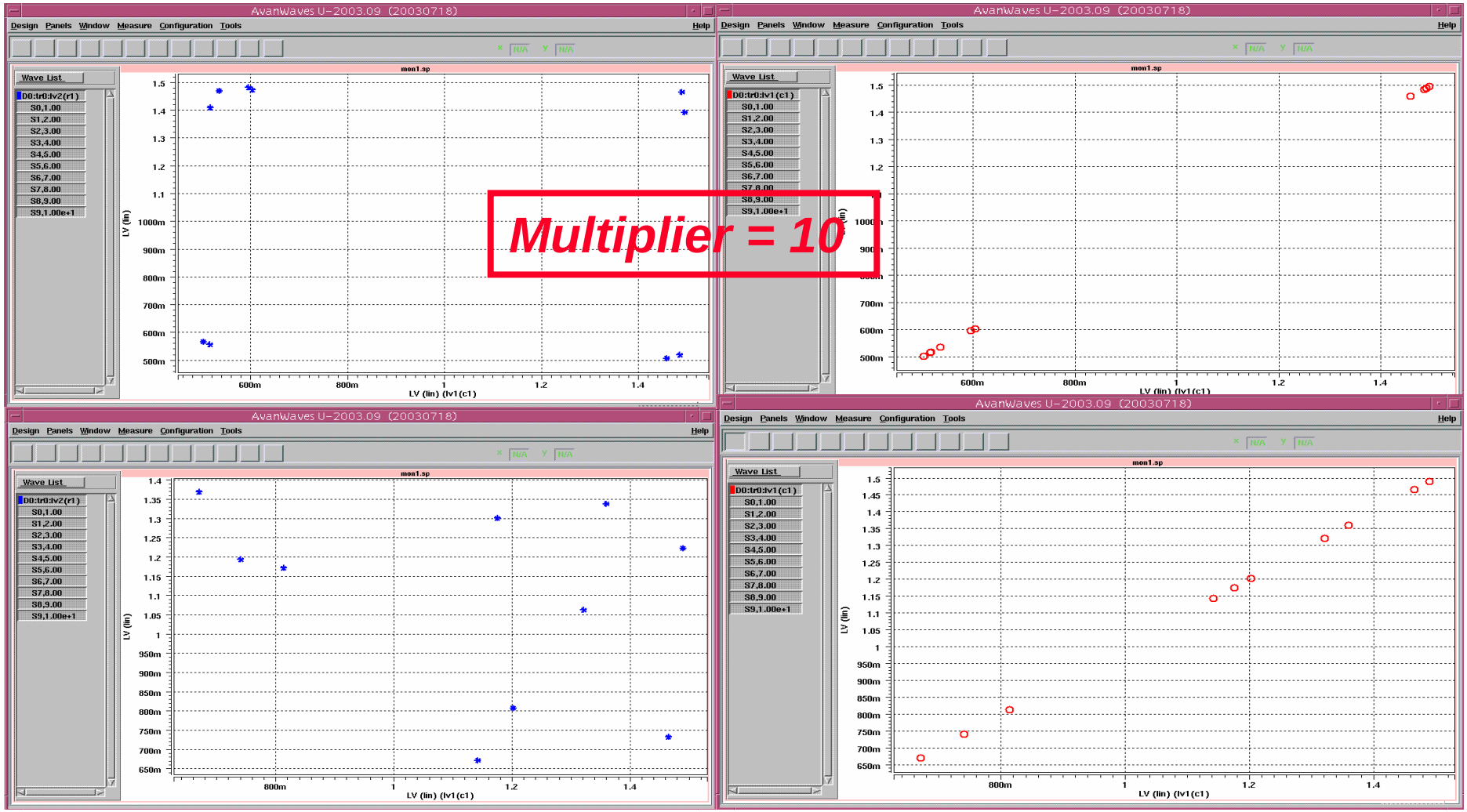
val --> Monte Carlo iterations number

Example 1

```
mon1.sp
.option list post=2
.param rx=unif(1,.5) cx=unif(1,.5)
*.param rx=unif(1,.5,10) cx=unif(1,.5,10)
.tran .1 1 sweep monte=10
.probe tran lv1(c1) lv2(r1)
.ic 1 1
r1 1 0 rx
c1 1 0 cx
.end
```



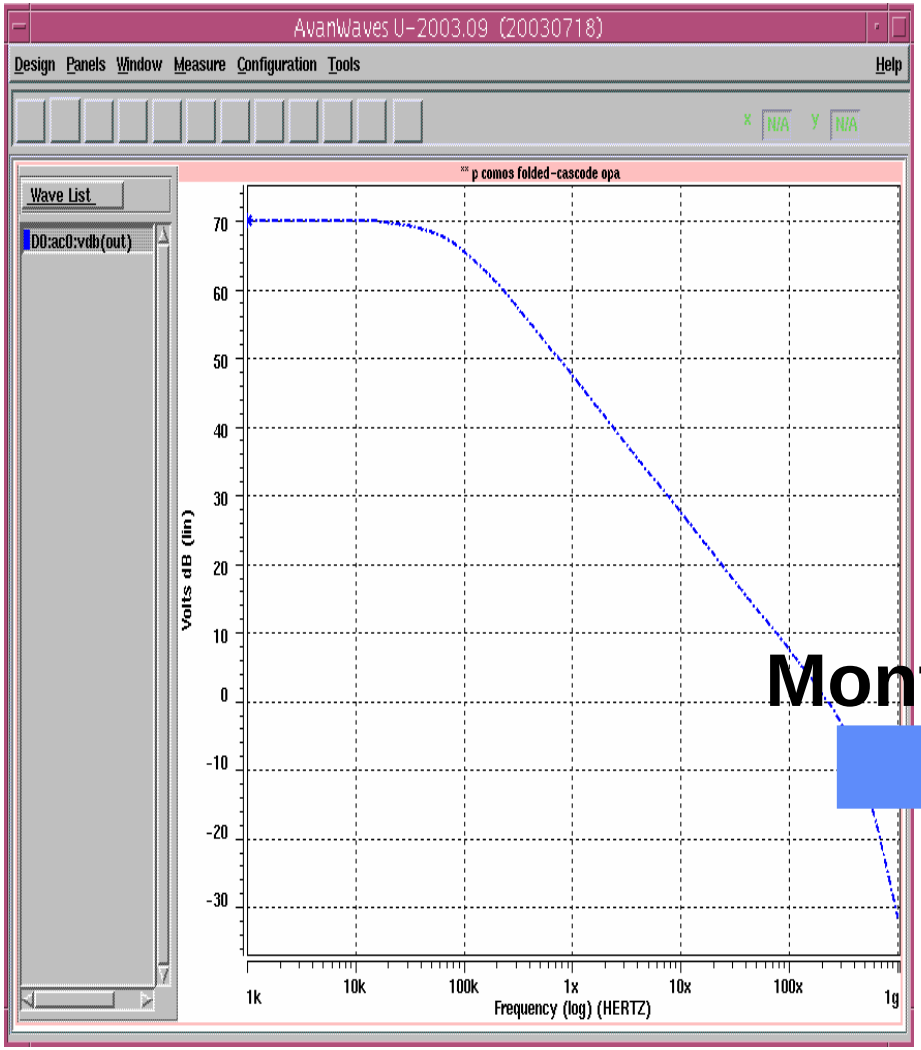
Example 1 (Cont.)



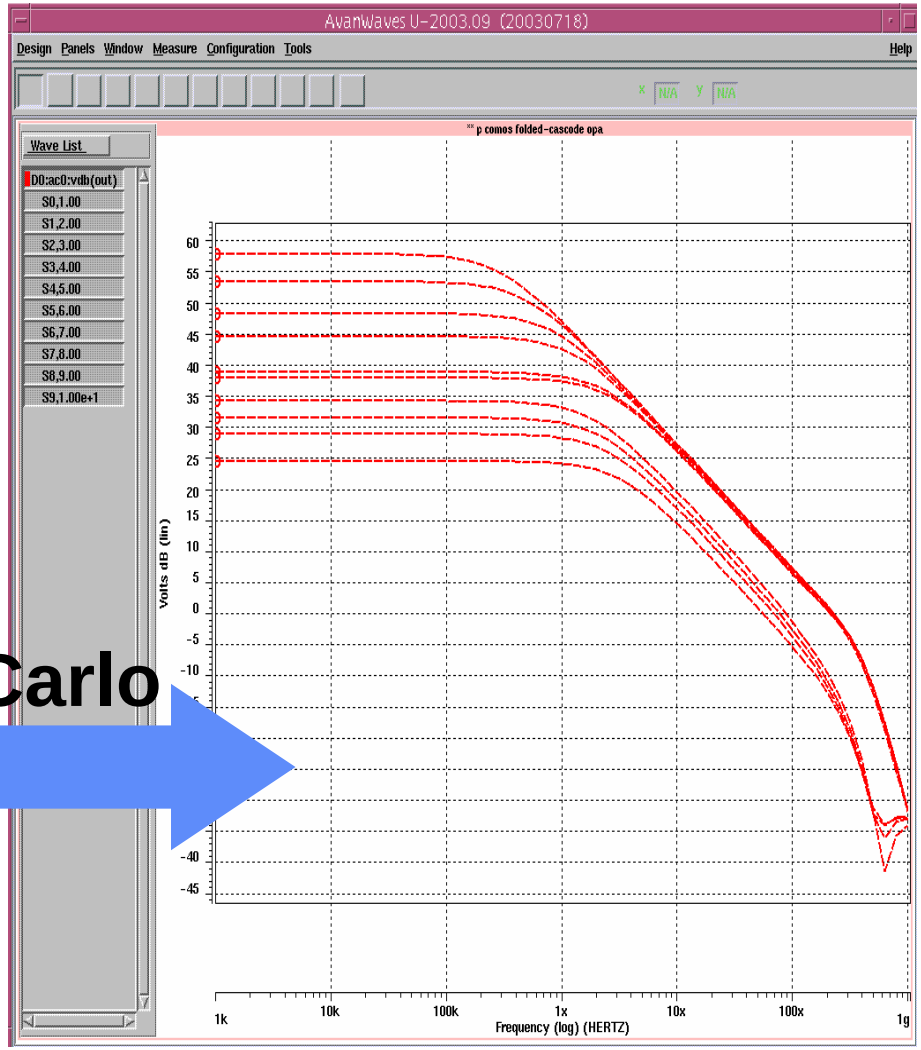
Example 2
Folded cascode OPAMP

```
vbias1  net0  0 2.32
vbias2  net1  0 1.8
vbias3  net2  0 0.778
* Monte-carlo's parameter
.param w1   =agauss(104.1u,1u,3)
+      w2   =agauss(97.73u,1u,3)
+      w3   =agauss(45.21u,1u,3)
+      w4   =agauss(41.67u,1u,3)
+      w5   =agauss(5.313u,1u,3)
* Monte-Carlo Analysis
.ac dec 10 1k 1000Meg  sweep  monte=10
* Monte-Carlo measure
.meas Amx max vdb(out)
.meas unity_gain_freq when vdb(out)=0
.meas ac phase_margin find=par'180+vp(out)' when vdb(out)=0
```

Example 2 (Cont.)



Monte-Carlo

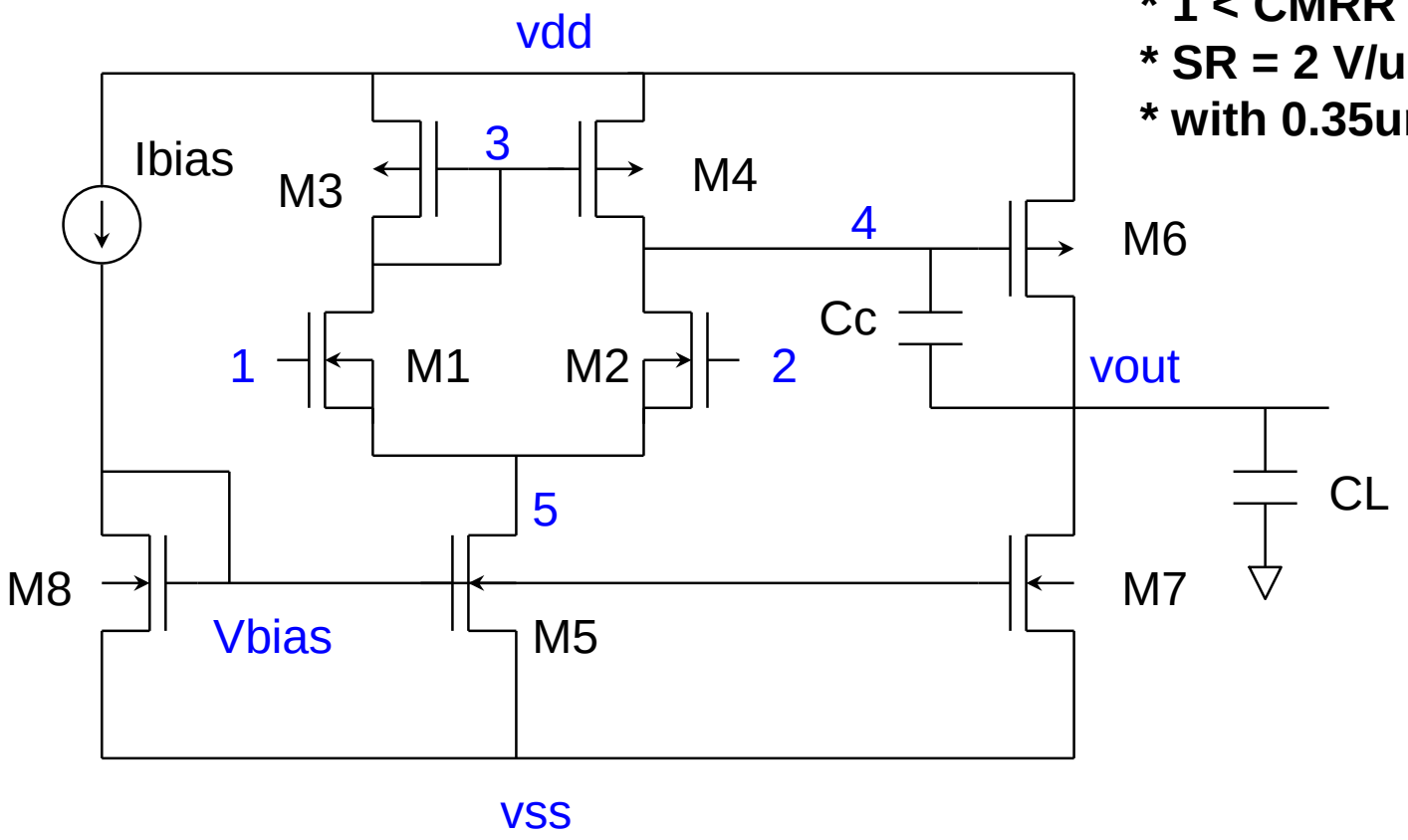


CHAPTER 10

Applications Demonstration

Two-stage OP AMP Design

*** Target specification :**
*** CL = 4pF, Av>4000,**
*** GB=2MHz**
*** 1 < CMRR < 4 , 0.8 < Vout < 4.2**
*** SR = 2 V/us , Pdiss < 10mW ,**
*** with 0.35um TSMC process**



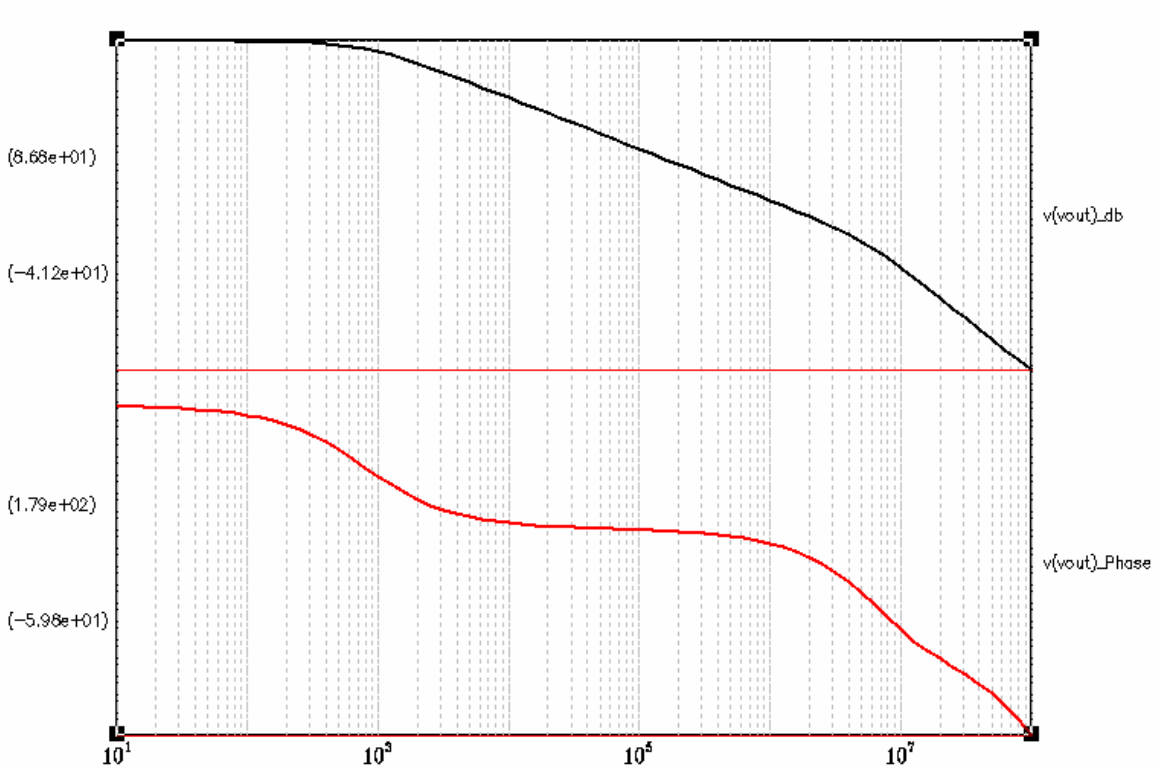
Netlist

```
*Two stage OP design
.lib “mm0355v.l” mos_tt
.option post nomod
* Netlist information
M1 3 1 5 0 nch L=2u W=8u AS=18p AD=18p
+                    PS=18u PD=18u
M2 4 2 5 0 nch L=2u W=8u AS=18p AD=18p
+                    PS=18u PD=18u
M3 3 3 vdd vdd pch L=10u W=10u AS=12p AD=12p PS=16u PD=16u
M4 4 3 vdd vdd pch L=10u W=10u AS=12p AD=12p PS=16u PD=16u
M5 5 vbias vss vss nch L=2u W=7u AS=49p AD=49p PS=26u PD=26u
M6 vout 4 vdd vdd pch L=2u W=70u AS=490p AD=490p PS=150u PD=150u
M7 vout vbias vss vss nch L=2u W=130u AS=930p AD=930p
+                                   PS=260u PD=260u
M8 vbias vbias vss vss nch L=2u W=7u AS=49p AD=49p PS=26u PD=26u
```

```
* Feedback CAP
Cc vout 4 0.44pF
Cl vout 0 4pF
Ibias vdd vbias 8.8u

* Voltage sourses
vdd vdd 0 5v
vss vss 0 0v
```

AC Frequency Analysis

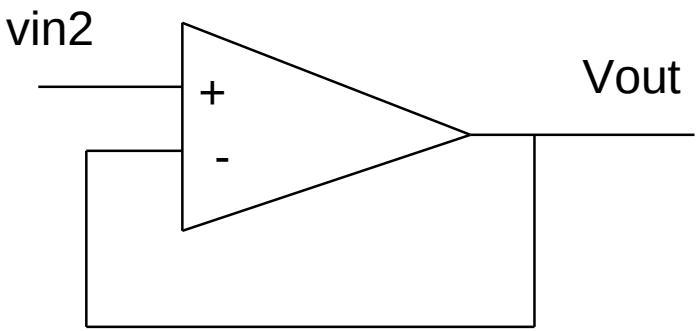
```
vin2 2 0 2.5v
vin1 1 0 DC 2.5v AC 1

*.OP

* AC Analysis function
.ac dec 10 10 100MEG
.probe ac vdb(vout)
+ vp(vout) vdb(4) vp(4)
.meas ac Unit_gain
+ when vdb(vout)=0
.meas ac phase_mar
+ FIND vp(vout) when vdb(vout)=0
```

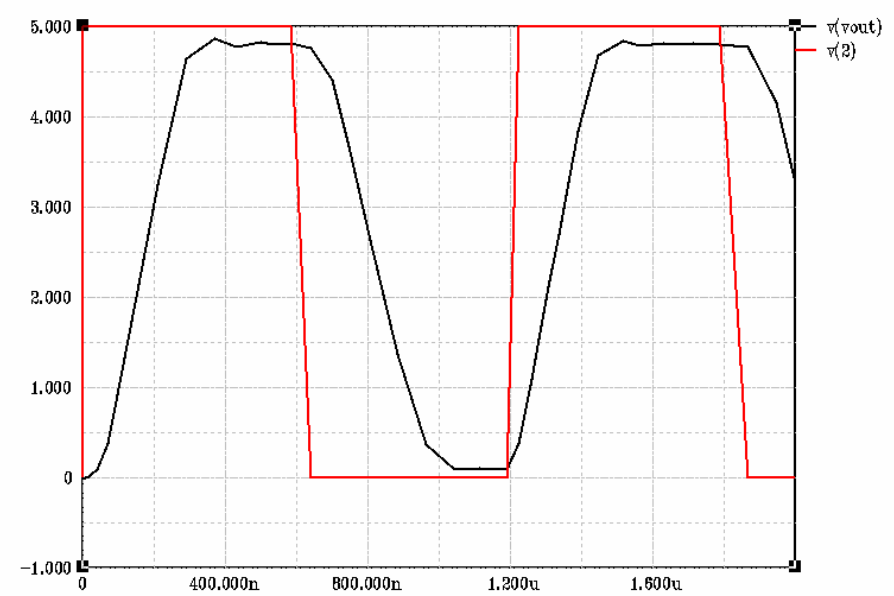


Transient Analysis : Slew Rate Analysis

*** Transient analysis section**

```
M1 3 vout 5 0 nch L=2u W=8u AS=18p AD=18p PS=18u PD=18u
vin2 2 0 pulse(0v 5v 1n 1p 1p 600n 1200n)
.tran 5n 2u
.probe tran
.save all
```





Future read

- %~synopsys/sold/cur/doc/online/hspice/home.pdf