

Seminario 3: Lenguaje de Simulación SPICE

SPICE : *Simulation Program with Integrated Circuit Emphasis*

- Se desarrolló en la Universidad de California (Berkeley), y su versión SPICE1 (Fortran) es de 1973
- La primera versión en C y entorno gráfico (X-Windows) es SPICE3 1989
- **Sintaxis básica común** entre todas las versiones
- Diferentes empresas de semiconductores lo adaptaron (codigo abierto)
- Las versiones comerciales más populares:
 - **PSPICE** (orig MicroSim, OrCAD, Cadence). Primera versión (1984) para PC, con versión de evaluación (*Student Version*). Contiene alteraciones del modelo original, siendo muy usado por las compañías de componentes electrónicos.
 - **HSPICE** (orig Meta, Synopsys). Pensado para funcionar en mainframes. Los fabricantes de silicio (foundries) lo emplean para ajustar elaborados modelos (BSIM).

- Se suelen distinguir tres bloques:

1. *Netlist* : Describe los elementos que componen un circuito, conexionado y modelos
2. *Analyses* : Describe las pruebas a realizar
3. *Representación* : Visualización y Medida de resultados

Aunque las dos primeras suelen ser muy compatibles entre versiones, la representación es dependiente del software empleado.

Bibliografía

- www.wikipedia.org
- www.gte.us.es/~chavez/doc/spice.pdf
- www.freedra.org/doc/SPICE/spice.pdf

Netlist

- Fichero de entrada en texto plano (*ASCII*)
- No se diferencia entre mayúsculas y minúsculas (*case insensitive*)
- Lenguaje orientado a línea (80 caracteres)
 - El primer carácter de la línea indica el componente descrito
 - **+** : indica continuación línea anterior (primer carácter)
 - ***** : indica comentario (primer línea)
 - La primera línea es el título y la última **.END**
 - **.INCLUDE "filename"** : permite insertar ficheros
- Descripción nodal
 - la conexión del elemento es mediante nodos
 - los nodos pueden ser números (consecutivos o no) y/o cadena de caracteres
 - El nodo **0** es la referencia de tensiones del circuito (siempre debe de existir)
 - la solución son las **tensiones nodales** y las **corrientes de las fuentes de tensión**

Las cantidades se pueden expresar mediante: enteros, flotantes, formato exponencial o exponencial literal:

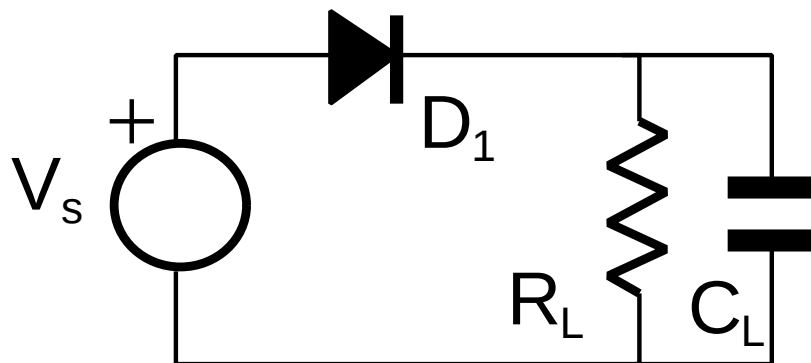
$$1500 = 1.5e3 = 15e2 = 1.5k$$

$k = 10^{+3}$	$m = 10^{-3}$
$meg = 10^{+6}$	$u = 10^{-6}$
$g = 10^{+9}$	$n = 10^{-9}$
$t = 10^{+12}$	$p = 10^{-12}$
	$f = 10^{-15}$

■ Una línea de elemento está compuesta por:

- **identificador del elemento** : cuya primera letra determina el tipo de elemento
- **nodos** entre los que se conecta
- **valor** de los parámetros

Ejemplo:



Circuito Rectificador

*BAS521, NXP Semiconductors

*Single high-voltage switching diode

*VRRM=300V, IFRM=1A @ tp=1ms trr=50ns

.MODEL BAS521 D IS=7.219E-9 N=2 BV=310

+ IBV=0.0001 RS=0.63 CJO=4.074E-13

+ VJ=0.6538 M=0.05409 FC=0.5 TT=2.1987E-8

+ EG=0.69 XTI=2

Vs 1 0 dc 12 sin(0, 12, 50)

D1 1 2 bas521

RL 2 0 120

CL 2 0 100u

.end

Veamos algunos componentes:

■ Resistencia

Rxxx n1 n2 valor

Ejemplo: R1 1 2 1kohm

■ Condensador

Cxxx n+ n- valor [IC=initial]

Ejemplo: C10 12 23 100pf ic=0v

■ Inductor

Lxxx n+ n- valor [IC=initial]

Ejemplo: L4 0 21 10uH ic=10mA

■ Diodo

Dxxx na nk d_model

■ Bipolar

Qxxx nc nb ne bjt_model

■ Mosfet

Mxxx nd ng ns nb mos_model

■ Fuente independiente

- de tensión *Vxxx n+ n- wave*
- de intensidad *Ixxx n+ n- wave*

Pudiendo ser la forma de onda:

• DC waveform

DC val

• PULSE waveform

PULSE v1 v2 td tr tf pw per

• SIN waveform

SIN vm va freq td theta

• EXP waveform

EXP v1 v2 td1 tau1 td2 tau2

• PWL waveform

PWL t1 v1 t2 v2 t3 v3 t4 v4 . . .

■ Fuente dependiente (lineal)

- VCCS *Gxxx n+ n- nc+ nc- gain_mho*
- VCVS *Exxx n+ n- nc+ nc- gain_av*
- CCCS *Fxxx n+ n- vmeas gain_ai*
- CCVS *Hxxx n+ n- vmeas gain_ohm*

Análisis

- **analisis OP**

Resolución estática (regimen permanente) del circuito.

OP

- **analisis DC**

Resolución estática de un circuito, para diferentes valores de una fuente de tensión/corriente (característica estática)

DC name ini last step

- **analisis AC**

Resolución en frecuencia, en pequeña señal del circuito (linealizado en el OP). Debe existir al menos una fuente independiente, cuya excitación senoidal de frecuencia variable se considera la entrada.

AC dec/lin nstep f_ini f_last

- **analisis TRAN**

Resolución en el tiempo (*transient*), del circuito (sistema de ecuaciones diferenciales). Pueden especificarse condiciones iniciales para simular transitorios.

TRAN tstep tstop

Representación de resultados

■ PRINT value/s

Permite imprimir un valor o un vector, una vez realizado un análisis.

- $V(n1)$ permite acceder a la tensión de un nodo
- $V(n1,n2)$ permite acceder a la tensión entre dos nodos
- $I(Vname)$ permite acceder a la corriente por una fuente

Se puede emplear como calculadora:

*PRINT 1m+sqrt(15*20u/8k)*

Dispone de una función para computar decibelios: $dB(XX)$

El operador redirección ($>$) permite el almacenamiento en fichero de texto.

■ SHOW device

Permite acceder a los detalles del modelo de los dispositivos

■ PLOT value/s

Permite realizar una gráfica¹ con los vectores de salida.

¹El eje horizontal es: variable en un DC, **tiempo** en un TRAN o **frecuencia** en un AC

Software WinSpice

Versión **WIN32** del Berkeley Spice3f4

- admite librerías *PSpice*
- admite modelos *BSIM3* y *BSIM4*
- permite generar ventanas de representación
- versión “Shareware”

Control Window

```

C:\tmp\sim\prac-eb\wspice3.exe
File Edit Settings Help
Note: can't find init file.
*****
WinSpice © Copyright 1996-2002 Mike Smith. All Rights Reserved.

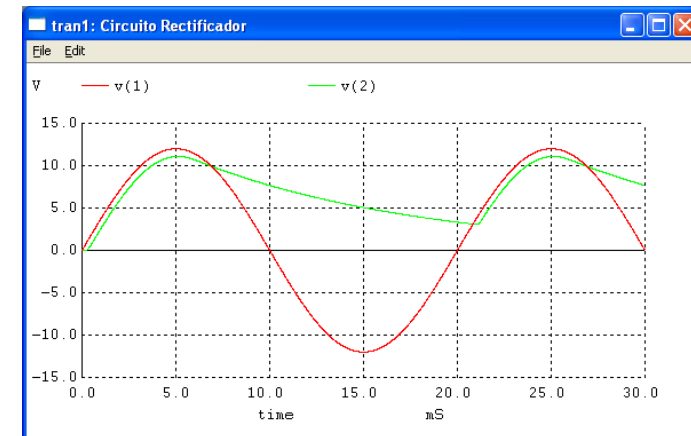
Version: 1.04.00
Built : Aug 3 2002 17:56:39

Shareware version of WinSpice. For non-commercial use only.
Please read the file 'license.txt' for conditions of use.
*****
WinSpice 1 -> source drec.cir
Reading .\drec.cir
NOTE: This is a Spice2 circuit file
Spice3 commands added:-
.control
destroy all
set units=degrees
.endc

Circuit: Circuito Rectificador

WinSpice 2 -> tran 10u 30m
Transient analysis ... 100%
WinSpice 3 -> plot v(1), v(2)
WinSpice 4 -> _
  
```

Plot Window



Disponible en el apartado **Documentos**

Ejemplos

Tengamos un fichero de texto: **drec.cir**

```
Circuito Rectificador
*BAS521, NXP Semiconductors
.MODEL BAS521 D IS=7.219E-9 N=2
BV=310
+ IBV=0.0001 RS=0.63 CJO=4.074E-13
+ VJ=0.6538 M=0.05409 FC=0.5
+ TT=2.1987E-8 EG=0.69 XTI=2
Vs 1 0 dc 12 sin(0, 12, 50)
D1 1 2 bas521
RL 2 0 120
CL 2 0 100u
.end
```

■ Análisis OP

```
WinSpice 2-> op
TEMP=27 deg C
DC Operating Point ... 100%
WinSpice 3-> print all
v(1) = 1.200000e+01
v(2) = 1.109467e+01
vs# branch = -9.24556e-02
WinSpice 4-> show d1
Diode: Junction Diode model
device d1
model bas521
vd 0.847
id 0.0934
...
```

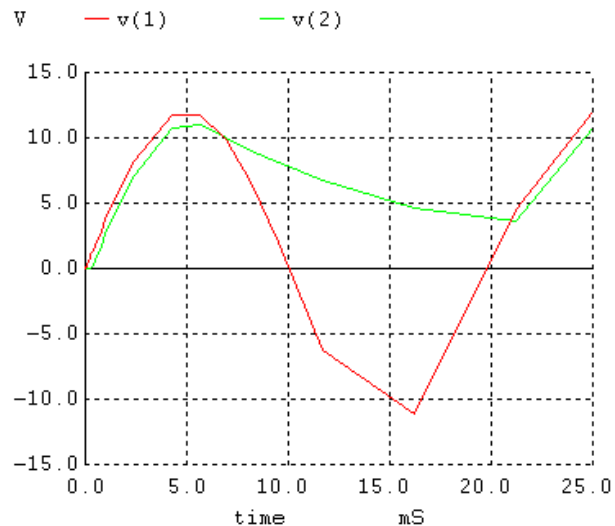
Comandos en la **Control Window**

■ Carga el fichero de netlist:

```
WinSpice 1-> drec.cir
Reading .\drec.cir
...
Circuit: Circuito Rectificador
WinSpice 2->
```

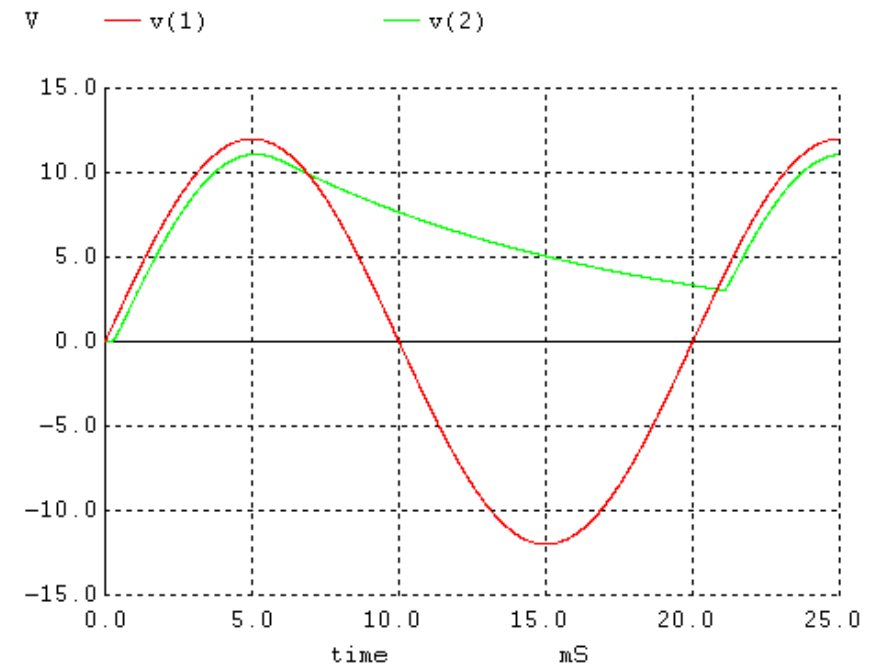
■ Análisis TRAN

```
WinSpice 5-> tran 5m 25m
Transient analysis ... 100%
WinSpice 6-> print v(1), v(2)
Circuito Rectificador
Transient Analysis
-----
Index time v(1) v(2)
-----
0 0.000000e+00 0.000000e+00 -5.084690e-23
1 5.000000e-06 1.884955e-02 2.361428e-10
2 5.429122e-06 2.046730e-02 2.578441e-10
3 6.287367e-06 2.370280e-02 3.042247e-10
...
WinSpice 7-> plot v(1), v(2)
```

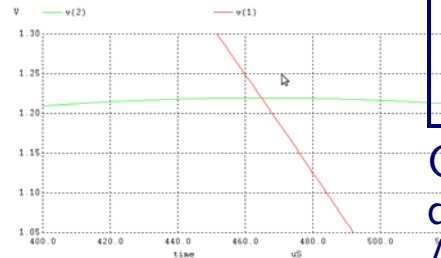
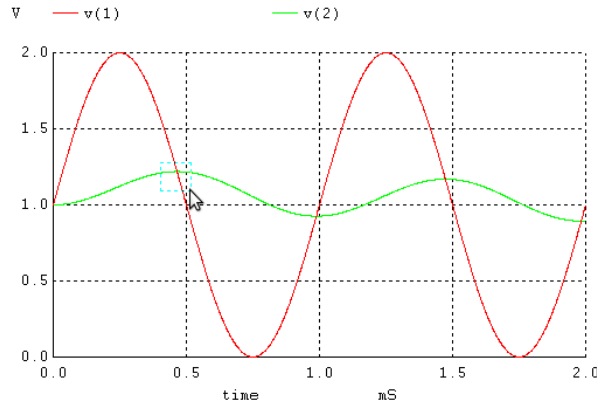


Aunque el ajuste del tiempo de integración es dinámico, es conveniente acotarlo para evitar anomalías que impidan la correcta medida

```
WinSpice 8-> tran 0.2m 25m
WinSpice 9-> plot v(1), v(2)
```

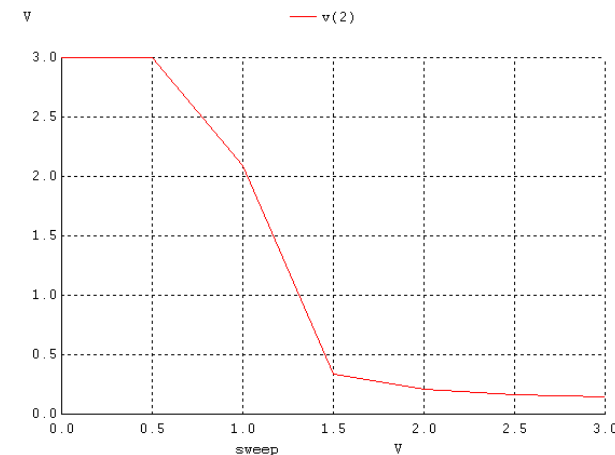


- **Zoom** Podemos hacer un zoom con el LMB, que hace aparecer una nueva ventana de zoom, que facilita la medida



```
WinSpice 1-> inv.cir
Reading .\inv.cir
...
Circuit: *** Inversor nMOS
WinSpice 2-> dc vs 0 3 0.5
DC analysis ... 100%
WinSpice 3-> plot v(2)
```

Con el análisis del comando 2, hemos variado V_s desde 0 a 3 con incrementos de $0,5v$. Al reproducir la tensión de drenador, vemos que sólo tenemos 4 puntos. . .

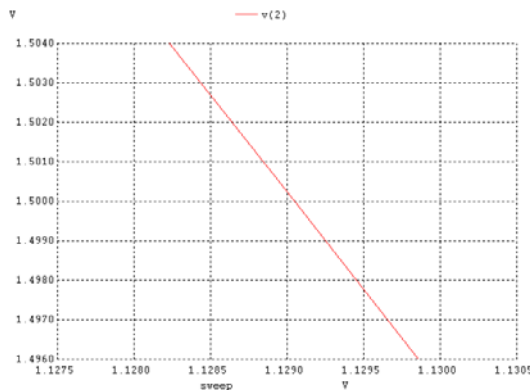
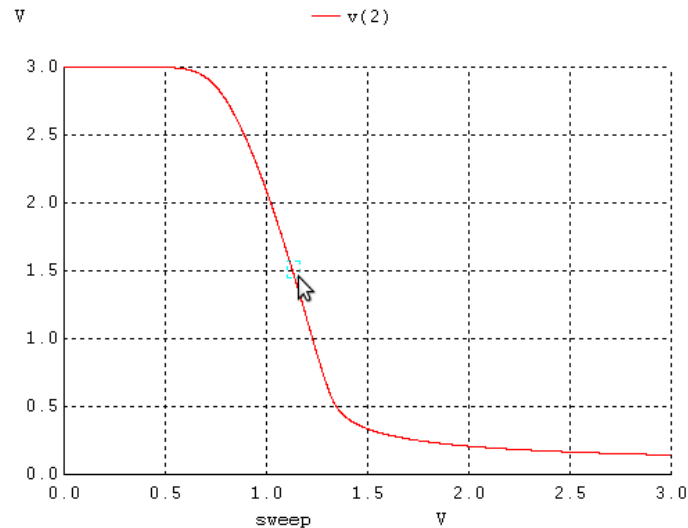


```
Inversor mos
*** Inversor nMOS
.include "mosisv1.mod"
Vdd 3 0 dc 3
Vs 1 0 dc 1
m1 2 1 0 0 cmosn w=10u l=1u
R1 3 2 10k
.end
```

- **Análisis DC**
Tengamos el siguiente circuito: **inv.cir**, y queremos obtener la tensión V_s para que la tensión de drenador sea $1,5v$

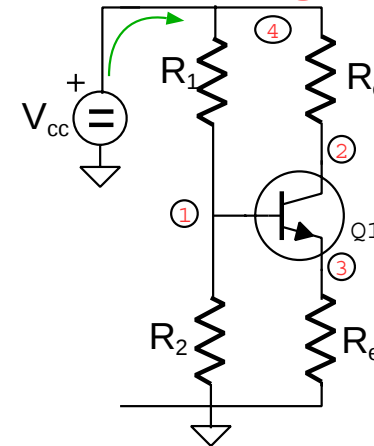
WinSpice 4-> `dc vs 0 3 1m`
 DC analysis ... 100%
 WinSpice 5-> `plot v(2)`

Hemos repetido el análisis pero con incrementos de 1mV , ahora la curva tiene suficiente resolución para realizar un zoom y una medida



■ Medida de Intensidades

SPICE computa la corriente que circula por las fuentes de tensión (se considera positiva si **entra por el signo +**).



```
** Montaje 4R bipolar
.include "mycomp.lib"
Vcc 4 0 dc 12
R1 4 1 10k
R2 1 0 15k
Rc 4 2 1k
Re 3 0 500
Q1 2 1 3 q2n2222a
.end
```

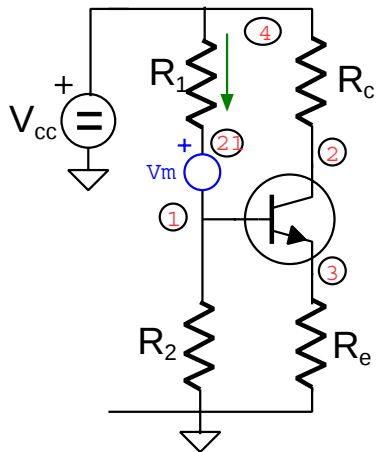
Para obtener la corriente de la fuente V_{cc}

WinSpice 10-> `op`
 WinSpice 11-> `print i(Vcc)`
`vcc#branch = -8.54969e-03`

El signo es negativo porque la fuente alimenta al circuito, y por tanto **sale por el +**

Si necesitamos la corriente por un elemento, se puede insertar una fuente de tensión de valor nulo, que no modifica el comportamiento y nos permite obtener su corriente.

En el siguiente ejemplo, queremos obtener la corriente por R1 para lo que insertamos la fuente **vm**, nótese que la corriente entra por el + de la fuente.



```
** Montaje 4R bipolar
.include "mycomp.lib"
Vcc 4 0 dc 12
R1 4 21 10k
Vm 21 1 dc 0
R2 1 0 15k
Rc 4 2 1k
Re 3 0 500
Q1 2 1 3 q2n2222a
.end
```

```
WinSpice 12-> op
WinSpice 13-> print i(vm)
vm#branch = 3.263027e-04
```

Alternativamente si queremos obtener la corriente DC de un transistor podemos emplear el comando **SHOW** o acceder individualmente:

```
WinSpice 14-> show q1
BJT: Bipolar Junction
Transistor
device q1
model q2n2222a
ic 0.00784
ib 0.000384
ie -0.00822
vbe 0.783
vbc 0.737
gm 0.185
gpi 0.00228
gmu 0.0106
...
WinSpice 15-> print @q1[ib]
@q1[ib] = 3.842436e-04
```

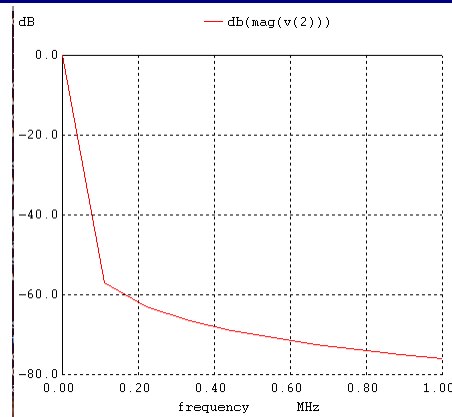
Análisis frecuencial

Tengamos el circuito: **rc.cir**, donde hemos indicado la señal de entrada (con amplitud unitaria) es $v(1)$.

```
*** Basic RC Lowpass
V1 1 0 dc 1.5 ac 1 sin(1.5,0.5,1k)
R1 1 2 1k
C1 2 0 1u
.end
```

Podemos realizar un barrido de 10pts desde $1Hz$: $1MHz$, representando la ganancia: $V(2)/V(1)$:

```
WinSpice 1->rc.cir
WinSpice 2-> ac lin 10 1 1meg
WinSpice 3-> plot vdb(2)
```

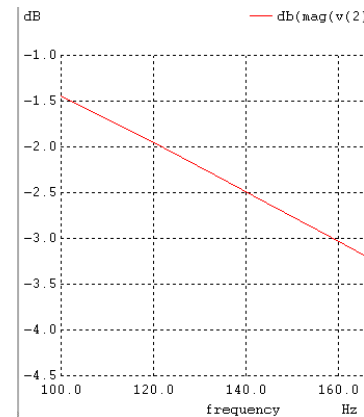
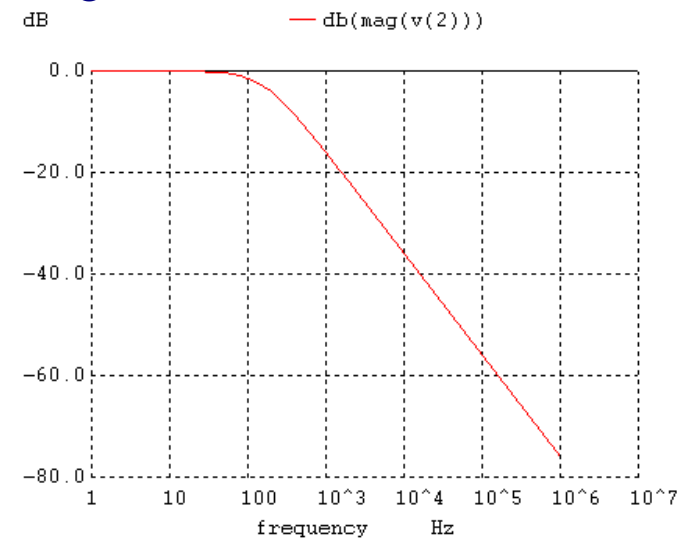


```
WinSpice 4-> ac dec 10 1 1meg
WinSpice 5-> plot vdb(2)
WinSpice 6-> ac lin 10 100 200
WinSpice 7-> plot vdb(2)
```

Con la línea 4 obtenemos un barrido entre $1Hz$: $1MHz$ con 10pts por decada (60pts total).

Con este análisis podemos ver el comportamiento LP, e incluso determinar la ganancia en banda ($0dB$ en este caso), pero no computar con precisión la frecuencia de $3dB$.

Para esto último emplearíamos la línea 6 y un rango reducido.

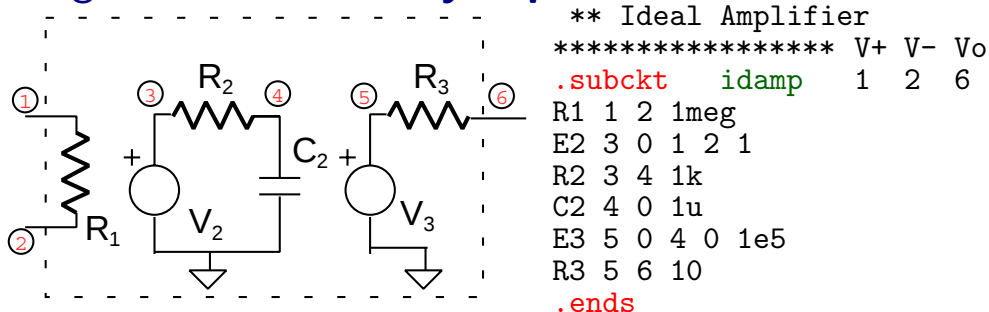


Subcircuito

Conjunto de elementos agrupados en bloque, dispone de terminales con los que se comunica con el exterior, pudiendo ser instanciado una o más veces.

Suele emplearse en librerías, ver instrucciones en comentarios

Tengamos la libreria **myamp.lib**



En **test_amp.cir** veremos cómo instanciar:

```

*** Two Stage Amplifier
.include "myamp.lib"
V1 10 0 dc 1.5 sin(1.5,0.5,1k)
R1 10 11 1k
R2 11 12 2k
X1 0 12 13 idamp
R3 12 13 4k
R4 13 14 5k
X2 0 13 14 idamp
.end

```

Aunque no es frecuente, podemos acceder a nodos internos mediante **subckt_id:int_node** :

```

WinSpice 1-> test_amp.cir
WinSpice 2-> tran 1u 3m
WinSpice 2-> plot v(10), v(x1:4)

```