

Se debe entregar el listado del programa (fichero *.cir) y los resultados que se piden (sólo los que se piden). Estos resultados pueden ser numéricos, que se obtienen del fichero *.out, o bien las gráficas generadas por Probe.exe. Los listados, los resultados, todo, se debe entregar en papel, nada en disquete.

INSTALACION Y USO

Descomprimir los ficheros, y copiarlos en un directorio. PSPICE para DOS consta de dos programas:

PSPICE1.EXE El pspice propiamente dicho. Su entrada es un fichero acabado en ".CIR" y su salida es un fichero con el mismo nombre, pero acabado en ".OUT". El fichero ????.CIR se crea con cualquier editor de texto, contiene el programa y la descripción del circuito. El fichero ????.OUT contiene los resultados de la simulación, o las explicaciones y posición de los errores cometidos.

PROBE.EXE Permite ver gráficamente los resultados, siempre que en el fichero ????.CIR se pidiera esta opción con la orden ".PROBE". Su fichero de entrada es PROBE.DAT (lo genera PSPICE), y su fichero de configuración es PROBE.DEV.

PSPICE.BAT Es una macro que corre primero PSPICE1 y luego PROBE.

"pspice ej4ac" o "pspice ej4ac.cir" simula el circuito en ej4ac.cir. El fichero de salida con los resultados sería ej4ac.out. Tras la simulación se arranca probe.exe inmediatamente. Para no llamar a probe.exe, y mandar los resultados a un fichero salida.out, ejecutar "pspice1 ej4ac.cir salida.out"

CONFIGURACION

En el fichero PROBE.DEV está la configuración de pantalla e impresora para PROBE.EXE:

DISPLAY = IBM

HARD-COPY = LPT1, EPSONMX

La opción mas útil es imprimir las gráficas en un fichero en formato plotter (extensión PLT, HPG o HGL), que puede ser importado por WordPerfect, Word, y la mayoría de programas de dibujo. Esto se consigue mandando los datos a un "FILE" y utilizando como impresora "HP":

HARD-COPY = FILE, HP

Otra opción, es correr el programa Probe en una ventana DOS dentro de Windows, y hacer un volcado de pantalla hacia un fichero. Resultará un gráfico binario, cuya resolución dependerá de la que tenga la pantalla del ordenador. Además será conveniente invertir los colores para no imprimir el fondo negro.

Valores posibles de pantalla:

TEXT	IBM	IBMClr	IBMEGA	GenericEGA	IBMEGAMono	AT&T	Hercules
Tecmar8	Tecmar4	Tecmar3	Tecmar1	FutureNet			

Valores posibles de puerto de impresora:

PRN	LPT1	LPT2	LPT3	COM1	COM2	FILE
-----	------	------	------	------	------	------

Valores posibles de impresoras:

TEXT	TEXT132	Okidata	Okidata132	Epson	Epson132	EpsonMX
EpsonMX132	IBMClr	IBMClr132	IBMClrSlw	IBMClrSlw132	Citoh	Citoh132
CitohSlw	CitohSlw132	Citoh-300	Citoh-300-132	Printronix	Printronix132	DECLA50
DECLA100	DECLA100-8	TOSHIBA	TOSHIBA132	HI	HP	HP6
HP8	HPLJ	HPLJ100	HPLJ150	HPLJ300		

1. INTRODUCCION

El programa SPICE resuelve un circuito utilizando las ecuaciones de corriente de Kirchhoff. A cada nudo se le asigna un número único como identificación; el nudo cero siempre debe corresponder a tierra. Todos los nodos deben tener un camino de continua hacia tierra.

El programa SPICE siempre empieza con un título y acaba con ".END". Entre estas dos sentencias debe estar la descripción del circuito, la definición de dispositivos (.MODEL, .LIB), el tipo de análisis (.AC, .DC, .TRAN) y las variables que se desean calcular o mostrar (.PRINT, .PLOT, .PROBE).

Las líneas de comentarios, deben empezar por "*", pero si se quieren añadir comentarios a una sentencia se utiliza ";". Si una línea es muy larga, la línea de continuación debe empezar por "+".

2. DESCRIPCION DE ELEMENTOS

Las líneas que describen los elementos empiezan por una letra fija (R para resistencias) y un máximo de 7 caracteres cualesquiera (xxxxxxx), luego vienen los números de nodos (n+ y n-) y al final el valor del elemento o el modelo de dispositivo.

Elementos:

Resistencia	Rxxxxxxx	n+	n-			valor (Ohmios)
Condensador	Cxxxxxxx	n+	n-			valor (Faradios)
Inducción	Lxxxxxxx	n+	n-			valor (Henrios)
Diodo	Dxxxxxxx	n+	n-			nombre de modelo
BJT	Qxxxxxxx	nc	nb	ne		nombre de modelo
JFET	Jxxxxxxx	nd	ng	ns		nombre de modelo
MOSFET	Mxxxxxxx	nd	nq	ns	nb	nombre de modelo

En el diodo, n+ y n- son los nodos del lado P y del lado N. En el BJT, nc, nb y ne, se refieren a los nodos de Colector, Base y Emisor. En los FETs, nd, ng, ns y nb, se refieren a los nodos de Drenador, puerta (Gate), fuente (Source) y substrato (Bulk o Body).

Fuentes independientes:

Tensión	Vxxxxxxx	n+	n-	[[DC] valorDC] [AC valorAC]
Intensidad	Ixxxxxxx	n+	n-	[[DC] valorDC] [AC valorAC]

Fuentes dependientes:

Tensión	cont. por tensión	Exxxxxxx	n+	n-	nc+	nc-	valor (V)
Intensidad	cont. por tensión	Gxxxxxxx	n+	n-	nc+	nc-	valor (A)
Tensión	cont. por intensidad	Hxxxxxxx	n+	n-	vnom		valor (V)
Intensidad	cont. por intensidad	Fxxxxxxx	n+	n-	vnom		valor (A)

En las fuentes dependientes de tensión (Exxx, Gxxx) nc+ y nc- son los nodos de los cuales depende la fuente. Las fuentes Hxxx y Fxxx, dependen de la intensidad que circula por la fuente de tensión vnom. En SPICE es usual insertar fuentes de tensión de valor 0V para utilizarlas como amperímetros, basta pedir la intensidad que circula por la fuente de tensión.

Para usar fuentes de alterna con forma sinusoidal ("SIN"), de onda cuadrada ("PULSE"), o lineales a tramos ("PWL") se utilizan:

SIN (offset amplitud frecuencia tiempo-retardo amortiguamiento)

PULSE (mínimo máximo tiempo-retardo tiempo-subida tiempo-bajada ancho-pulso período)

PWL (t₁ V₁ t₂ V₂ ... t_n V_n)

Ejemplos:

Vcc 10 0 DC 6 fuente de tensión continua entre los nodos 10 y tierra, de 6V.
Vac 2 1 AC 1 fuente de tensión alterna de amplitud 1V.
Vi 2 0 sin(0 2V 1kHz 0 0) fuente senoidal, con amplitud 2V y frecuencia 1 kHz

3. FORMATO NUMERICO

Los valores pueden ser enteros (2, -35), en coma flotante (3.16, -6.5) o en formato exponencial (-1.2E-4, 2.34E11). Recordar que los ingleses utilizan un punto en lugar de la coma decimal usada en el resto de países.

Factores de escala:

K = 1E3 (kilo) MEG = 1E6 (Mega) G = 1E9 (Giga) T = 1E12 (Tera)
M = 1E-3 (mili) U = 1E-6 (micro) N = 1E-9 (nano) P = 1E-12 (pico)
F = 1E-15 (femto) p.ej. 2.3K equivale a 2300 , 3E+10F equivale a 3E-5

4. FORMATO DE MODELO

".MODEL" se usa para definir las características de un dispositivo concreto. El nombre que se dará al nuevo dispositivo es *nombre_de_modelo*, y *tipo_de_modelo* es uno de los tipos predefinidos. Cada *tipo_de_modelo* tiene unos parámetros, a los que hay que dar valores concretos. La sintaxis es:

.MODEL *nombre_de_modelo* *tipo_de_modelo* (pnom1=pval1 pnom2=pval2 ...)

Tipo_de_modelo:

D para diodo
NPN para BJT de tipo npn
PNP para BJT de tipo pnp
NJF para JFET de canal n
PJF para JFET de canal p
NMOS para MOSFET de canal n
PMOS para MOSFET de canal p

En la siguiente tabla se listan algunos parámetros, sus abreviaturas (pnom) y los valores por omisión:

D. Corriente de saturación	IS	1E-14 A
D. Coeficiente de emisión (η)	N	1
D. Tensión umbral	VJ	1 V
D. Resistencia óhmica	RS	0 Ohm
D. Tensión de ruptura inversa	BV	infinito
BJT. Beta directa máxima	BF	100
BJT. Tensión Early	VA	infinito
BJT. Tensión base-emisor	VJE	0.75 V
BJT. Capacid. colector-base	CJC	0 F
BJT. Capacid. base-emisor	CJE	0 F
BJT. Tiempo transito directo	TF	0 s
JFET Tensión umbral	VTO	-2 V
JFET Transconductancia	BETA	1E-4 A/V ²
JFET Capacid. de unión G-S	CGS	0 F
JFET Capacid. de unión G-D	CGD	0 F
MOSFET Tensión umbral	VTO	0 V
MOSFET Transconductancia	KP	2E-5 A/V ²
MOSFET Capacid. de sustrato-d	CBD	0 F
MOSFET Capacid. de sustrato-s	CBS	0 F

5. SUBCIRCUITOS

Es posible descomponer circuitos complejos en asociaciones de circuitos más simples o subcircuitos.

Llamada a un subcircuito: Xnombre n1 n2 n3 modelo_subcircuito

Definición de un subcircuito: .SUBCKT modelo_subcircuito n1 n2 n3
 (definición)
 .ENDS

En la librería OPNOM.lib se puede ver la definición del 741 como un subcircuito, y la llamada al subcircuito UA741 se puede ver en EJ4AC.cir de los programas de ejemplo en la sección 9.

6. LAS LIBRERIAS

Las librerías son ficheros que contienen definiciones de subcircuitos y dispositivos (diodos, transistores, etc.). Estas definiciones se pueden utilizar en los circuitos, siempre que el programa (*.CIR) incluya una llamada a la librería que contiene las definiciones.

La sentencia ".LIB NOM.LIB" es equivalente a ".LIB", y llama a la librería NOM.LIB que vuelve a llamar a otras (DNOM.LIB, QNOM.LIB, MNOM.LIB, OPNOM.LIB, CPNOM.LIB y BJT.LIB). También se puede llamar por separado a una sola librería, usando el nombre y la ruta: ".LIB c:\directorio\BJT.lib".

Contenido de las librerías:

DNOM.LIB Contiene las definiciones de los diodos D1N914, D1N916 y diodos zener (tensión zener indicada en voltios) D1N752 (5,2V), D1N754 (6,4V), D1N759 (11,4V).

QNOM.LIB Contiene las definiciones de los transistores bipolares 2N2222 (NPN, $\beta=217$) y 2N2907 (PNP, $\beta=197$).

BJT.LIB Con los modelos de los transistores BC107B (NPN, $\beta=300$), BD557 (PNP, $\beta=30$), LM394 (NPN, $\beta=600$), BC547B (NPN, $\beta=300$), BC557B (PNP, $\beta=344$), BD237 (NPN, $\beta=108$) y alguno más.

MNOM.LIB Contiene la definición de dos MOSFET: IRF150 (canal N) y IRF9130 (canal P).

OPNOM.LIB En la librería se define el Amplificador Operacional uA741

CPNOM.LIB En la librería se define el Comparador LM111

7. TIPOS DE ANALISIS

De barrido en continua (DC): Se analiza el circuito mientras se hace variar el valor de continua de alguna fuente de tensión o de intensidad.

`.DC Vx1 valor_inicio valor_final dc_incremento`

La orden de la línea anterior, hace variar el valor de la fuente de tensión Vx1 desde *valor_inicio* hasta *valor_final*, en pasos de *dc_incremento*. Admite un doble bucle, por ejemplo, el siguiente comando, hace un barrido de la fuente V1 de 0V a 5V, para cada valor de la fuente Ib, que varía de 0 mA a 100 mA:

`.DC V1 0V 5V 1V Ib 0mA 100mA 20mA`

De barrido en el tiempo (TRANSitorio): Analiza la evolución temporal de magnitudes del circuito.

`.TRAN tiempo_paso tiempo_fin tiempo_inicio tiempo_incremento`

La orden anterior, realiza un análisis transitorio, calculando los resultados desde un tiempo igual a *tiempo_inicio* hasta *tiempo_fin*, con incremento igual a *tiempo_paso*. Si la resolución gráfica es pobre, se debe incluir *tiempo_incremento* ("Step_Ceiling") con valor menor de $(tiempo_final - tiempo_inicio)/50$.

De barrido en frecuencias (AC): Analiza la respuesta del circuito al variar la frecuencia de las fuentes de señal alterna (AC):

`.AC LIN npuntos frec_inicio frec_final`

El comando precedente, realiza un análisis en alterna, calculando los resultados pedidos en un barrido en total de *npuntos* frecuencias, de forma LINEal empezando por la frecuencia *frec_inicio* hasta *frec_final*. Si en la orden anterior se sustituyese "LIN" por "DEC", se realizaría un barrido logarítmico con *npuntos* por cada DECada. Con "OCT" sería un barrido con *npuntos* por cada OCTava.

De temperatura: Calcula los resultados pedidos a diferentes temperaturas.

`.TEMP 50 75` Realiza el análisis a temperaturas de 50°C y 75°C.

Del punto de operación (OP): Calcula el punto de operación de los transistores, da los voltajes en todos los nodos y corrientes en las fuentes de tensión. Es conveniente añadir esta orden cuando se hacen otros tipos de análisis.

8. SALIDA DE RESULTADOS

`.PRINT tipo_salida variables_salida` da el valor numérico de *variable_salida* en el fichero *.out

`.PLOT tipo_salida variables_salida` grafica las variables de salida, en formato caracter, en *.out

`.PROBE` Almacena todos los datos del circuito para ser vistos luego con Probe

Donde *tipo_salida* indica el tipo de datos que se han de mostrar, y puede ser:

DC se muestran los valores de continua de las variables de salida.

TRAN se muestran los valores de variables-salida en función del tiempo.

AC se muestran los valores de alterna de variables_salida.

Y *variables_salida* puede ser :

- V(n1,n2) caída de tensión entre los nodos n1 y n2
- V(n3) caída de tensión entre n3 y tierra (tierra siempre es nodo 0)
- I(Vs) intensidad que circula por la fuente de tensión Vs
- VDB(n1,n2) caída de tensión en decibelios (para obtener Bode en módulo)
- VP(n1,n2) fase de la caída de tensión (para diagrama de Bode en fase)

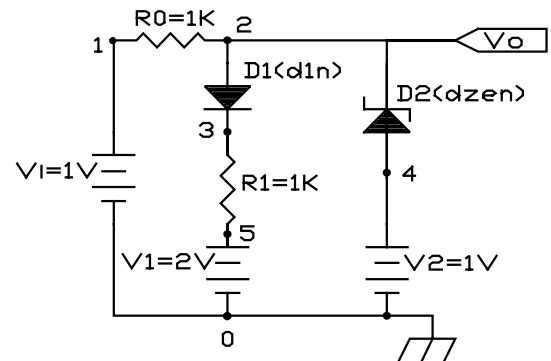
9. EJEMPLOS

Los ficheros mostrados se dan con la versión DOS de SPICE.

EJ1DC.CIR (ver F.1):

*Ejemplo 1. Barrido en DC. Característica de transferencia

```
Vi 1 0 1V
R0 1 2 1Kohmio
D1 2 3 d1n
R1 3 5 1Kohmio
V1 5 0 2V
D2 4 2 dzen
V2 4 0 1V
.DC Vi -3 10 .1 ; Barre Vi desde -3V a +10V
.print DC v(1),v(2),i(d1),i(d2),i(v1),i(R1)
.probe
.lib d:ele2d.lib ; Librería con los diodos
.end
```



F.1

EJ1TRAN.CIR (ver F.1):

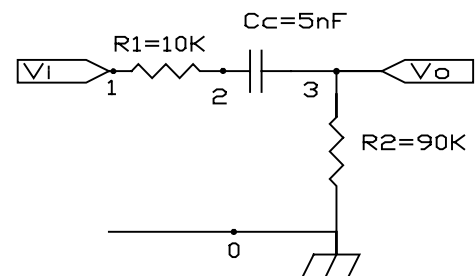
*Ejemplo 1. Transitorio. Cálculo de la salida en función del tiempo

```
Vi 1 0 sin(0 10V 1KHz 0 0 ) ; Amplitud de 10V con frecuencia de 1KHz
R0 1 2 1Kohmio
D1 2 3 d1n
R1 3 5 1Kohmio
V1 5 0 2V
D2 4 2 dzen
V2 4 0 1V
.TRAN 0.1ms 2ms 0ms ; Desde 0 segundos a 2 milisegundos paso=0.1 ms
.print TRAN v(1),v(2),i(d1),i(d2),i(v1),i(R1)
.probe
.lib d:ele2d.lib
.end
```

EJ2TRAN.CIR (ver F.2):

*Ejemplo 2. Transitorio. Respuesta de un circuito RC en función del tiempo
* pulse=onda cuadrada desde -1V a +2V de periodo 1ms, 0.6ms=ancho del pulso

```
Vi 1 0 pulse( -1V 2V 0 0 0 0.6ms 1ms )
R1 1 2 10Kohmio
Cc 2 3 5nF
R2 3 0 90Kohmio
.TRAN 0.02ms 2ms 0
.print TRAN v(1),v(3)
.probe
.end
```



F.2

EJ2AC.CIR (ver F.2):

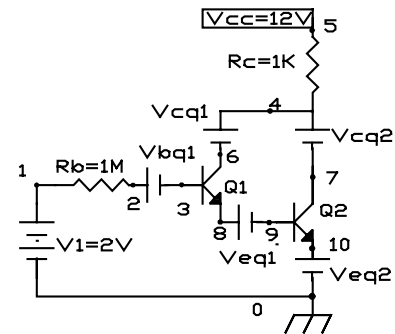
*Ejemplo 2. Análisis AC. Diagrama de Bode en módulo y fase de un circuito RC

```
Vi 1 0 AC 1 ; Vi es fuente alterna
R1 1 2 10Kohmio
Cc 2 3 5nF
R2 3 0 90Kohmio
.AC dec 14 10 10000 ; Se simula desde 10Hz a 10kHz con 14 puntos por década
.print AC v(1),v(2),vdb(3),vp(3)
.probe
.end
```

EJ3BJT.CIR (ver F.3):

* Ejemplo 3. Tr Bipolar. Calculo de la polarización de un par Darlington

```
V1 1 0 DC 2V
Rb 1 2 1MEGohmio
Vbq1 2 3 0V ; Fuentes de tensión de valor cero, sirven
Vcq1 4 6 0V ; para medir las intensidades que
Veq1 8 9 0V ; entran a base, colector y emisor
Vcq2 4 7 0V ; de Q1 y Q2
Veq2 10 0 0V
Q1 6 3 8 bc107 ; colector base emisor de Q1
Q2 7 9 10 bc107 ; El bc107 equivale al bc109
Rc 5 4 1Kohmios
Vcc 5 0 12V
.lib d:ele2d.lib
.print DC i(Rb) i(Rc)
.dc v1 1.8 2 .1 ; Para que Spice dé los datos
* hace falta hacer un barrido.
* muestra Vbe Vce Ib Ic Ie
.print DC v(3,8) v(6,8) i(vbq1) i(vcq1) i(v eq1)
.print DC v(9,10) v(7,10) i(v eq1) i(vcq2) i(v eq2)
.end
```



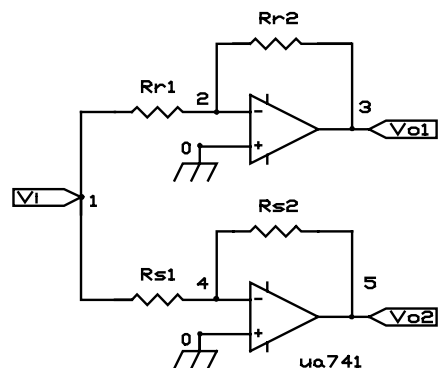
F.3

En Ej3bjt.cir se hace un barrido .DC sólo para que Spice calcule las variables pedidas (v(3,8), v(6,8) etc.). Se podría haber hecho también con un barrido .TRAN (más simple: .tran 1 1). Sin embargo, si lo que se necesitan son sólo las variables del punto de operación de un transistor (caso de Ej3bjt.cir), entonces lo mejor sería no hacer ningún barrido (ni .DC, ni .TRAN), sino que hubiese bastado con incluir la orden .OP (que calcula los puntos de operación, las tensiones en los nodos, y las corrientes en las fuentes de tensión).

EJ4AC.CIR (ver F.4):

* Ejemplo 4A. Bodes de 2 configuraciones de operacional

```
Vc1 8 0 +12V ; +Vcc
Vc2 9 0 -12V ; -Vcc
vv1 1 0 ac .2
rr1 1 2 10k
rr2 2 3 10k
xaor 0 2 0 8 9 3 UA741 ; +,-,gnd,+Vcc,-Vcc,output
rs1 1 4 2k
rs2 4 5 10k
xaos 0 4 0 8 9 5 UA741
.ac dec 9 10 1000000
.probe
.lib opnom.lib ; Librería que contiene al 741
.end
```



F.4

EJ4TRAN.CIR (ver F.4):

* Ejemplo 4B. Salida de los 2 Operacionales en función de t

```
Vc1 8 0 +12V ; +Vcc
Vc2 9 0 -12V ; -Vcc
vv1 1 0 sin(0 .3 1KHz 0 0 )
rr1 1 2 10k
rr2 2 3 10k
xaor 0 2 0 8 9 3 UA741 ; +,-,gnd,+Vcc,-Vcc,output
rs1 1 4 5k
rs2 4 5 10k
xaos 0 4 0 8 9 5 UA741
.tran .1ms 2ms 0
.probe
.lib opnom.lib
.end
```

10. PROBE

Los comandos de Probe están en inglés, y se explican perfectamente. Al añadir curvas, se pueden incluir las variables V(2,1), V(3) (diferencia de potencial entre los nodos 2 y 1, y los nodos 3 y 0), y expresiones como V(2,1)-2*V(3). Dispone además de funciones como seno (sin), coseno (cos), tangente (tan), raíz cuadrada (sqrt), derivada (d), integral (s), potencia (p(x,y)=|x|^y), mínimo (min), máximo (max), valor absoluto (abs), exponencial (exp), logaritmo natural y decimal (log y log10) y algunas otras más.