

TEMA 1: Lenguaje SPICE

1.1. Introducción.

SPICE es un acrónimo inglés que significa Simulation Program with Integrated Circuits Emphasis (Programa de simulación con énfasis en circuitos integrados). Dicho programa fue desarrollado por Donald Pederson, de la Universidad de California, Berkeley, en 1975.

SPICE es un simulador que mediante un lenguaje de descripción hardware, nos permite estudiar el comportamiento de circuitos electrónicos.

El simulador SPICE usa un lenguaje de descripción, del que dada su complejidad, explicaremos los aspectos más importantes necesarios para estudiar los circuitos electrónicos de esta asignatura.

Hay diversos programas, denominados genéricamente *capture*, que mediante un interfaz gráfico facilitan la creación del fichero de texto con el lenguaje SPICE (fichero *netlist*). En esta asignatura utilizaremos el simulador gratuito LTSpice, que contiene tres partes diferentes:

- Interfaz gráfico que permite introducir un esquemático, y generar la *netlist* del circuito.
- Simulador Spice propiamente dicho.
- Visualizador de resultados.

1.2. Normas generales del lenguaje SPICE.

Los circuitos pueden ser creados con cualquier editor de textos, pero el fichero de texto no debe tener caracteres de control (como códigos relativos a los márgenes, tipo de letra, interlineado, etc.).

Aquí tenemos una lista de reglas sobre los ficheros de texto que describen los circuitos:

- 1) La primera línea será siempre el título y/o comentario del circuito.
- 2) La última línea será la sentencia .END (de final), si bien en el editor de Control Shell no es necesaria incluirla.
- 3) Las líneas que sean un comentario deben empezar con un asterisco (*).
- 4) Las líneas que sean una continuación de la sentencia de la línea anterior deben empezar con un signo de suma (+).
- 5) El orden de las líneas que describen el circuito no importa, excepto para el título, definiciones de subcircuitos y la línea con la sentencia .END.
- 6) PSPICE no diferencia letras mayúsculas de minúsculas, por lo que podemos utilizar cualquiera de ellas.
- 7) Para separar los distintos parámetros de una sentencia, podemos utilizar espacios, tabuladores o comas, que son equivalentes y no importa cuántos se usen.

1.3. Nombres de los elementos y nudos de conexión

Los nombres de los elementos deben comenzar con una letra (que identifica el tipo de elemento al que pertenecen) seguidos del nombre del elemento en sí, pudiendo ser letras, números o los caracteres \$, _, *, /, %, y aunque pueden tener hasta 131 caracteres de longitud, es aconsejable no superar los 8.

ELEMENTO DEL CIRCUITO	SÍMBOLO	MODELO
Resistencias.	R	RES
Condensadores.	C	CAP
Bobinas.	L	IND
Acoplamientos magnéticos (transformadores).	K	CORE
Interruptores controlados por tensión.	S	VSWITCH
Interruptores controlados por intensidad.	W	ISWITCH
Diodos.	D	D
Transistores bipolares.	Q	NPN, PNP o LPNP
Transistores de efecto de campo JFET.	J	NJF o PJF
Transistores de efecto de campo MOSFET.	M	NMOS o PMOS
Transistores de efecto de campo GaAsFET.	B	GASFET
Fuentes de tensión independientes.	V	
Fuentes de intensidad independientes.	I	
Fuentes de tensión controladas por tensión.	E	
Fuentes de intensidad controladas por intensidad.	F	
Fuentes de intensidad controladas por tensión.	G	
Fuentes de tensión controladas por intensidad.	H	

Los nudos no han de ser obligatoriamente números enteros, pueden ser cualquier cadena alfanumérica, como los nombres. El nudo 0 (cero) está predefinido y es el correspondiente a tierra. Hay que remarcar que la numeración de los nudos no ha de seguir ningún orden especial.

1.4. Valores de los elementos.

Los valores de los componentes los escribiremos en notación de punto flotante estándar (ej. $1E-3 = 1 \times 10^{-3}$), y opcionalmente con sufijos multiplicadores y escala (unidos al valor sin dejar espacios intermedios). Estos son los sufijos multiplicadores reconocidos por PSPICE:

$F=10^{-15}$
 $P=10^{-12}$
 $N=10^{-9}$
 $U=10^{-6}$
 $MIL=25.4 \times 10^{-6}$
 $M=10^{-3}$
 $K=10^3$
 $MEG=10^6$
 $G=10^9$
 $T=10^{12}$

Estos sufijos multiplican el número que les precede inmediatamente. Cualquier otro sufijo añadido (como V, A, etc.) será ignorado por PSPICE, no teniendo efecto alguno. Los sufijos para las unidades normalmente utilizados son:

V = voltios
A = amperios
HZ = hertzios
OHM = ohmios
H = henrios
F = faradios
DEG = grados

Hay que hacer notar que SPICE admite como sufijo multiplicador la letra F (que multiplica el valor que le precede por un factor de 10^{-15}). Así, si al definir un condensador le damos un valor de 0.001F, éste para PSPICE no será de 0.001 faradios, sino de 0.001×10^{-15} faradios.

1.5. Parámetros.

En ciertas aplicaciones es conveniente usar parámetros globales (es decir, una variable), en lugar de valores numéricos; así, al darle un valor a un determinado parámetro, este valor aparecerá en todos los lugares donde esté ese parámetro.

Los parámetros pueden ser definidos usando la sentencia .PARAM (cuya sintaxis se estudiará más adelante), y serán globales a todo el circuito, es decir, se podrán utilizar en la definición de componentes en todo el circuito. Cuando un valor numérico es sustituido por un parámetro, este se escribirá entre llaves, {PARÁMETRO}.

Por ejemplo, si las resistencias R3 y R5 de un circuito dado tienen el mismo valor (digamos 1K), podemos definir las mediante las sentencias:

```
R3 4 6 1K; Resistencia colocada entre los nudos 4 y 6, de valor 1K.  
R5 3 8 1K; Resistencia colocada entre los nudos 3 y 8, de valor 1K.
```

o bien, utilizando un parámetro, podemos definir las mediante:

```
R3 4 6 {CARGA}  
R5 3 8 {CARGA}  
.PARAM CARGA=1K
```

1.6. Expresiones

Además de por parámetros, los valores de los elementos se pueden sustituir por expresiones aritméticas que pueden contener parámetros. Siguiendo el ejemplo del apartado anterior, si la resistencia R4 colocada entre el nudo 3 y 8, tiene un valor de 4K, se puede definir como:

```
R4 7 10 {4*CARGA}; Resistencia colocada entre los nudos 7 y 10, de valor 4K.
```

Las expresiones pueden ser usadas en lugar de los valores numéricos, en la descripción del circuito, en los siguientes casos:

- En los valores de todos los parámetros de los modelos.
- En los valores de todos los parámetros de los componentes, excepto los parámetros TC1 y TC2 (coeficientes de temperatura) de una resistencia si están definidos en la misma línea (con la sentencia R) que la resistencia y no con una sentencia .MODEL (usada para definir los modelos de los elementos).
- En los valores de todos los parámetros de las fuentes independientes de voltaje e intensidad, excepto para las
- fuentes definidas por tramos.
- En los valores de las sentencias .IC y .NODESET, usadas para establecer las condiciones iniciales del punto de
- trabajo del circuito.

Pero no podemos usar expresiones en los siguientes casos:

- En sustitución de los coeficientes de los polinomios que definen el valor de las fuentes de tensión e intensidad controladas no lineales.
- En lugar de los nombres/números de los nudos.
- En el lugar de los valores numéricos en las sentencias de realización de análisis del circuito.

Las expresiones deben tener una longitud máxima de una línea de texto y pueden tener contener las operaciones básicas de suma (+), resta (-), multiplicación (*) y división (/) además de los paréntesis necesarios y las funciones mostradas en la siguiente tabla:

FUNCIÓN	COMENTARIO
ABS(X)	Valor absoluto de X.
SQRT(X)	Raíz cuadrada de X.
EXP(X)	Función e^x .
LOG(X)	Logaritmo en base e de X, $\ln(X)$.
LOG10(X)	Logaritmo en base 10 de X, $\log(X)$.
PWR(X,Y)	Función $ X ^Y$.
SIN(X)	Función $\sin(X)$, expresado X en radianes.
COS(X)	Función $\cos(X)$, expresado X en radianes.
TAN(X)	Función $\tan(X)$, expresado X en radianes.
ATAN(X)	Función $\arctg(X)$, con resultado en radianes.
ARCTAN(X)	Función $\arctg(X)$, con resultado en radianes.

Además de estas funciones, podemos definir otras nuevas con la sentencia .FUNC (función) para incluirlas en las expresiones (lo cual nos permite aumentar la longitud de las mismas). Así, por ejemplo, para definir una función llamada MED(X,Y), encargada de calcular el valor medio (media aritmética) de los valores X e Y, la definiríamos mediante:

```
.FUNC MED(X,Y) (X+Y)/2
```

El número máximo de variables que puede contener una función es de 10. Las expresiones son evaluadas para conocer su valor antes de realizar los análisis y recalculadas con los nuevos valores de los parámetros si existen análisis en continua y/o análisis paramétrico.

1.7. Modelos

En PSPICE es muy frecuente el empleo de modelos, que se utilizan para definir el valor de los distintos parámetros de los elementos usados en el circuito. Si bien se pueden definir modelos para todos los elementos, en algunos casos su uso es opcional y en otros es obligado. Así, para los elementos pasivos, los modelos son opcionales, y para los elementos semiconductores son necesarios, aunque todos los parámetros de los modelos tienen asignado un valor por defecto, que será el usado por PSPICE en caso de que no le asignemos otro.

Para la descripción de los modelos de los diferentes componentes del circuito se utiliza la sentencia .MODEL, en la cual se define el valor de los parámetros de los distintos elementos disponibles en PSPICE. La sintaxis general de la sentencia es:

```
.MODEL (nombre) (tipo) (parámetro del modelo = valor (tolerancia)*)
```

El término con (*) es opcional, los valores entre doble paréntesis significan que hay que escribirlos entre paréntesis. ejemplo: EJ ((valor1) , (valor2)) se escribe como: EJ (valor1) , (valor2).

El término (nombre) es el que le asignamos al modelo en cuestión, y debe comenzar con una letra. El término (tipo) hace referencia a la clase de elemento a la que corresponde el modelo, y puede ser uno de los mostrados en la siguiente tabla:

En la descripción del circuito pueden existir varios modelos para el mismo tipo de elementos, por lo que deberán tener un nombre distinto. Cada tipo de modelo, correspondiente a una clase de elementos, tiene sus propios parámetros, los cuales tienen asignado un valor por defecto, por lo que podemos usar ese valor o bien darle uno diferente, si bien no es necesario cambiar el valor de todos los parámetros.

TIPO DE MODELO	ELEMENTO CORRESPONDIENTE
CAP	Condensadores.
IND	Bobinas.
RES	Resistencias.
D	Diodos.
NPN	Transistores bipolares NPN.
PNP	Transistores bipolares PNP.
LPNP	Transistores bipolares de estructura lateral PNP.
NJF	Transistores de unión FET de canal N.
PJF	Transistores de unión FET de canal P.
NMOS	Transistores MOSFET de canal N.
PMOS	Transistores MOSFET de canal P.
GASFET	Transistores GaAsFET de canal N.
VSWITCH	Interruptores controlados por tensión.
ISWITCH	Interruptores controlados por intensidad.

EJEMPLOS:

Definir un modelo, llamado RMAX, correspondiente a una serie de resistencias, en el que los parámetros de las mismas tengan unos valores de $R=1.5$, $TC1=0.02$ y $TC2=0.005$:

```
.MODEL RMAX RES(R=1.5 TC1=0.002 TC2=.005)
```

Definir un modelo, llamado DNOM, correspondiente a un diodo, en el que se utilicen los valores por defecto de todos los parámetros:

```
.MODEL DNOM D
```

Definir un modelo, llamado CFIL, correspondiente a unos condensadores, en el que el coeficiente multiplicador del valor de los mismos, es decir el parámetro C, tenga un valor de 1 y una tolerancia del tipo DEV del 5%:

```
.MODEL CFIL CAP(C=1 DEV=5%)
```

1.8. Opciones para simulación.

Para configurar las distintas opciones en PSPICE, que nos permitirán un mayor control sobre la simulación de los circuitos, utilizaremos las sentencia:

```
.OPTIONS (opción) (opción = valor)
```

La sentencia .OPTIONS se usa para establecer todas las opciones, límites y parámetros de control para los distintos análisis, incluido el formato (anchura) del fichero de salida de resultados.

Las opciones se listan en cualquier orden. Hay dos tipos de opciones: las que tienen valor numérico y las que no lo tienen. Las opciones sin valor son banderas o indicadores de varios tipos que se activan con sólo mencionarlos.

Las sentencias .OPTIONS se acumulan; es decir, si hay varias, tienen el mismo efecto que una sola que fuese equivalente (donde estuvieran todas las opciones juntas). Si la misma opción aparece varias veces, sólo el último valor será usado por PSPICE.

A continuación veremos una lista de las opciones **sin valor numérico** disponibles. Su valor por defecto es estar desactivadas; para activarlas sólo hemos de nombrarlas en la sentencia .OPTIONS:

OPCIÓN	SIGNIFICADO
ACCT	Presenta un sumario e información al final de todos los análisis en el fichero de salida.
EXPAND	Lista los componentes incluidos en los subcircuitos.
LIBRARY	Lista las líneas usadas de las librerías.
LIST	Presenta un sumario de los elementos del circuito.
NOBIAS	Suprime la presentación de las tensiones de los nudos del punto de trabajo.
NODE	Presenta un sumario de las conexiones del circuito.
NOECHO	Suprime la descripción del circuito en el fichero de salida.
NOMOD	Suprime el listado de los parámetros de los modelos y valores actualizados de temperatura.
NOPAGE	Suprime los saltos de página para cada sección del fichero de salida.
OPTS	Lista el valor de todas las opciones.
WIDTH	Establece el ancho del fichero de salida a 80 o 132 columnas.

En caso de incluir en la descripción del circuito la sentencia `.OPTIONS` con la opción `NOECHO`, se colocará al principio del fichero, tras la línea de título, para que realice adecuadamente su función.

Pasamos ahora a las opciones que tienen **valores numéricos**. Todas ellas tienen asignado un valor por defecto en PSPICE, pero nosotros podemos cambiar ese valor si lo estimamos oportuno. Para ello, hemos de utilizar la sentencia `.OPTIONS` con el nombre de la opción en cuestión seguido del nuevo valor que le vamos a asignar. En la siguiente tabla se muestran estas opciones con su significado y valor por defecto:

OPCIÓN	SIGNIFICADO	VALOR POR DEFECTO
ABSTOL	Máxima precisión para las intensidades.	1 pA
CHGTOL	Máxima precisión para las cargas.	0.01 pC
CPTIME	Tiempo permitido a la CPU para esta simulación.	1E+6 Sg
DEFAD	Área del drenador para los MOSFET por defecto.	0 m ²
DEFAS	Área del surtidor para los MOSFET por defecto.	0 m ²
DEFL	Longitud para los MOSFET por defecto.	100 um
DEFW	Anchura para los MOSFET por defecto.	100 um
GMIN	Mínima conductancia usada por cualquier rama.	1E-12 Ω^{-1}
ITL1	Límite de iteraciones para el cálculo del punto de trabajo DC.	40
ITL2	Límite de iteraciones para el cálculo del punto de trabajo DC con aproximaciones iniciales.	20
ITL4	Límite de iteraciones para calcular un punto del análisis transitorio.	10
ITL5	Límite total de iteraciones para el análisis del transitorio.	5000
LIMPTS	Máximo valor de puntos para una tabla de valores o representación gráfica por puntos.	∞
NUMDGT	Número de dígitos en las tablas de valores.	4
PINREL	Magnitud relativa en la resolución de matrices.	1E-3
PIVTOL	Magnitud absoluta en la resolución de matrices.	1E-13
RELTOL	Precisión relativa para tensiones e intensidades.	.001
TNOM	Temperatura nominal por defecto en °C.	27 °C
TRTOL	Factor de error por truncamiento.	7
VNTOL	Máxima precisión para tensiones.	1 uV

Ejemplos:

Establecer para los diferentes análisis a realizar en el circuito las siguientes condiciones: suprimir la descripción del circuito en el fichero de salida, suprimir el listado de los parámetros de los modelos, asignar una longitud para los MOSFET de 12|mm y asignar una anchura para los MOSFET de 8|mm:

```
.OPTIONS NOECHO NOMOD DEFL=12U DEFW=8U
```

Establecer para la simulación del circuito las siguientes opciones: generar un listado en el fichero de salida de los componentes incluidos en los subcircuitos, establecer una precisión relativa para las tensiones e intensidades de un 1% y establecer un número de iteraciones para el análisis transitorio de 50000:

```
.OPTIONS EXPAND RELTOL=.01 ITL5=50000
```

1.9. Incluir un fichero en el circuito

La sentencia .INC se usa para insertar el contenido de otro fichero en la descripción de nuestro circuito de trabajo. Tiene la siguiente sintaxis:

.INC (nombre del fichero)

El término (nombre del fichero) es el nombre del fichero cuyo contenido queremos insertar. Este fichero puede contener sentencias de todo tipo, excepto línea de título (pues ésta ha de estar en el circuito principal) y la sentencia de final .END.

Incluir un fichero en la descripción del circuito de trabajo equivale a escribir el texto que contiene dicho fichero en nuestro circuito.

Ejemplo:

Incluir en la descripción del circuito el contenido del fichero de texto llamado FUNCION.TXT (en el que tenemos definidas una serie de funciones):

.INC FUNCION.TXT

1.10. Utilización de librerías.

Para hacer referencia a un fichero librería en la descripción del circuito, utilizaremos la sentencia .LIB, con la siguiente sintaxis:

.LIB (nombre de librería)

En una librería será donde están definidos los modelos de los componentes comerciales, como diodos, transistores, etc., y también subcircuitos que simulan elementos más complejos, como amplificadores operacionales, tiristores, etc.

Una librería sólo puede contener sentencias .MODEL, definiciones de subcircuitos, algún comentario y sentencias .LIB que, a su vez, hacen referencia a otras librerías. En lugar del término (nombre de librería) tenderemos que especificar el nombre completo de la librería (incluida la extensión del fichero). Si la librería no se encuentra en el directorio actual hay que especificar la ruta completa.

Entre las librerías de PSPICE hay que destacar la NOM.LIB, que no es una librería propiamente dicha (pues no contiene modelos de elementos ni subcircuitos), pero que hace referencia a todas las demás librerías suministradas por el programa. Si en un momento dado queremos utilizar un componente y no recordamos en qué librería se encuentra definido, podremos hacer referencia a la librería NOM.LIB, con lo que PSPICE irá buscando el componente por todas las librerías hasta encontrarlo.

Si se omite en la sentencia .LIB el nombre del fichero librería, entonces PSPICE buscará el fichero NOM.LIB.

1.11. Sentencias de descripción de elementos

Cada elemento está representado en la descripción del circuito con una sentencia en una o más líneas (son las que comienzan por una letra).

La primera letra indica el tipo de componente (resistencia, condensador, diodo, etc.) y el resto de la línea nos da la información de ese componente (nombre, nudos de conexión, valor, nombre del modelo si es necesario y demás parámetros para la correcta definición del elemento).

Algunos componentes permiten asignarles un modelo y para otros es obligatorio. Un modelo es una forma de especificar los valores de los parámetros de un determinado elemento, como pueden ser los coeficientes de temperatura, etc.

El orden de los componentes en la definición del circuito no es importante, aunque para una mejor comprensión podemos agruparlos según el tipo (resistencias, condensadores, transistores, etc.). Las conexiones entre los componentes vienen determinadas por los nudos a los que van conectados.

1.12. Elementos pasivos

Los elementos pasivos disponibles en PSPICE son: resistencias, condensadores, bobinas y transformadores. Estos componentes se definen con las letras **R**, **C**, **L** y **K**, respectivamente.

1.12.1. Resistencias

Para insertar una resistencia en la descripción del circuito utilizaremos la sentencia:

R(nombre) (nudo+) (nudo-) (modelo)* (valor) TC=(tc1),(tc2)*

Opcionalmente las resistencias pueden tener un modelo. En caso de incluirlo, para definir los parámetros de las mismas, utilizaremos la sentencia:

.MODEL (modelo) RES((parámetro del modelo = valor)*)

El asterisco (*) indica que este valor es opcional, este símbolo se utilizará en la nomenclatura de todos los componentes que restan.

En la siguiente tabla podemos ver los distintos parámetros del modelo de la resistencia, así como su significado y valor por defecto con unidades.

PARÁMETRO	SIGNIFICADO	VALOR
R	Coeficiente multiplicador.	1
TC1	Coeficiente lineal de temperatura.	0°C ⁻¹
TC2	Coeficiente cuadrático de temperatura.	0°C ⁻²
TCE	Coeficiente exponencial de temperatura.	0° % / °C

Los nudos (+) y (-) definen el sentido de la polaridad cuando la resistencia tiene una tensión determinada. La corriente positiva circula desde el nudo (+) al nudo (-) a través de la resistencia.

Los coeficientes de temperatura pueden definirse en la misma sentencia de la resistencia. Si la resistencia tiene definido un modelo, los coeficientes de temperatura se usarán para calcular el valor de la misma, teniendo prioridad sobre los definidos en la sentencia propia de la resistencia.

Si no hemos asignado ningún modelo a la resistencia, su valor en ohmios será el especificado por el término (valor), y ha de ser positivo, nunca nulo.

1.12.2. Condensadores

Para insertar un condensador en la descripción del circuito, utilizaremos la sentencia:

C(nombre) (nudo +) (nudo -) (modelo)* (valor) (IC = condiciones iniciales)*

Opcionalmente, los condensadores pueden tener un modelo. En caso de incluirlo, para definir los parámetros de los mismos, utilizaremos la sentencia:

.MODEL (modelo) CAP(parámetros del modelo = valor)*

En la siguiente tabla podemos ver los distintos parámetros del modelo del condensador, así como su significado y valor por defecto con unidades.

PARÁMETRO	SIGNIFICADO	VALOR
C	Multiplicador de la capacidad.	1
VC1	Coeficiente lineal de tensión.	0 V ⁻¹
VC2	Coeficiente cuadrático de tensión.	0 V ⁻²
TC1	Coeficiente lineal de temperatura.	0 °C ⁻¹
TC2	Coeficiente cuadrático de temperatura.	0 °C ⁻²

Los nudos (+) y (-) definen el sentido de la polaridad cuando el condensador tiene una tensión almacenada. La corriente positiva circula desde el nudo (+) hasta el nudo (-) a través del condensador.

Si omitimos el (modelo) al describir el condensador, su capacidad en faradios será la especificada en el término (valor), que ha de ser una cantidad positiva, nunca nula.

1.12.3. Bobinas

Para insertar una bobina en la descripción del circuito, utilizaremos la sentencia:

L(nombre) (nudo +) (nudo -) (modelo)* (valor) (IC = condiciones iniciales)*

Opcionalmente, las bobinas pueden tener un modelo. En caso de incluirlo, para definir los parámetros de las mismas utilizaremos la sentencia:

.MODEL (modelo) IND(parámetro del modelo = valor)*

En la siguiente tabla podemos ver los distintos parámetros del modelo de la bobina, así como su significado y valor por defecto con unidades.

PARÁMETRO	SIGNIFICADO	VALOR
L	Factor multiplicador de la bobina.	1
IL1	Coeficiente lineal de corriente	0 A ⁻¹
IL2	Coeficiente cuadrático de corriente.	0 A ⁻²
TC1	Coeficiente lineal de temperatura.	0 °C ⁻¹
TC2	Coeficiente cuadrático de temperatura.	0 °C ⁻²

Los nudos (+) y (-) definen el sentido de la polaridad cuando la bobina tiene una tensión positiva. La corriente es positiva cuando circula desde el nudo (+) al nudo (-) a través de la bobina.

Si no se especifica el (modelo), entonces el valor de la bobina en henrios será el especificado en el término (valor), que ha de ser una cantidad positiva, nunca nula.

1.13. Elementos activos.

En este apartado estudiaremos las sentencias que se utilizan para introducir en la descripción del circuito los elementos activos del mismo, tales como diodos, transistores bipolares y de unión MOSFET.

1.13.1. Diodos

Para introducir un diodo en la descripción del circuito, usaremos la sentencia:

D(nombre) (ánodo) (cátodo) (modelo) (área)*

Una vez introducido el diodo, para su correcta descripción hemos de definir su modelo correspondiente o bien usar uno de la librería. La sintaxis del modelo es:

.MODEL (modelo) D(parámetro del modelo = valor)*

El término (área) es un coeficiente multiplicador que permite definir con los mismos parámetros aquellos diodos que están fabricados con el mismo proceso tecnológico, pero que difieren en el área de la unión p-n efectiva de cada diodo.

Todos los parámetros del modelo son características físicas del diodo, que dependen de la propia fabricación del mismo. En las librerías de PSPICE se encuentran descritos los modelos para una gran cantidad de diodos comerciales que podemos utilizar en nuestro circuito.

Ejemplos:

Describir un diodo estándar, al que hemos llamado con el número 3, que está colocado entre los nudos 5 y 4 para el ánodo y el cátodo, respectivamente:

```
D3 5 4 EST  
.MODEL EST D
```

Describir un diodo llamado RECT, colocado entre los nudos 1 y 2, cuyo modelo corresponde al diodo 1N4148 de la librería DIODE.LIB:

```
DRECT 1 2 D1N4148  
.LIB DIODE.LIB
```

1.13.2. Transistores Bipolares.

Para introducir un transistor en la descripción del circuito, usaremos la sentencia:

Q(nombre) (colector) (base) (emisor) (substrato)* (modelo) (área)*

Una vez introducido el transistor, para su correcta descripción hemos de definir su modelo correspondiente o bien usar uno de la librería. La sintaxis del modelo es:

```
.MODELO (modelo) NPN(parámetro del modelo = valor)*  
.MODELO (modelo) PNP(parámetro del modelo = valor)*
```

según sea el transistor NPN o PNP.

El nudo de conexión del substrato es opcional, y si no se especifica se conectará por defecto a tierra.

El término (área) es un factor multiplicador utilizado para establecer el número de transistores equivalentes en paralelo (para un modelo en concreto).

Ejemplos:

Describir un transistor PNP estándar, al que hemos llamado con el número 1, que está colocado entre los nudos 14, 2 y 13 para el colector, base y emisor, respectivamente:

```
Q1 14 2 13 PNPEST .MODEL PNPEST PNP
```

Describir un transistor NPN estándar, al que hemos llamado PASO, que está colocado entre los nudos 12, 5 y masa para el colector, base y emisor, respectivamente, y cuyo valor del área es 1.5:

```
QPASO 12 5 0 NPNEST 1.5  
.MODEL NPNEST NPN
```

Describir el transistor al que hemos llamado con el número 3, correspondiente al transistor Q2N2222 de la librería BIPOLAR.LIB, que está colocado entre los nudos 1, 2 y 5 para el colector, base y emisor, respectivamente:

```
Q3 1 2 5 Q2N2222  
.LIB BIPOLAR.LIB
```

1.13.3. Transistores MOSFET

Para introducir un transistor MOSFET en la descripción del circuito, usaremos la sentencia:

M(nombre) (drenador) (puerta) (surtidor) (substrato) (modelo) (L = valor)* (W = valor)* + (AD = valor)*
(AS = valor)* (PD = valor)* (PS = valor)* (NRD = valor)* (NRS = valor)* + (NRG = valor)* (NRB = valor)*
(M = valor)*

Una vez definido el transistor, para su correcta descripción hemos de definir su modelo correspondiente o bien usar uno de la librería. La sintaxis del modelo es:

.MODEL (modelo) NMOS(parámetro del modelo = valor)* .MODEL (modelo) PMOS(parámetro del modelo = valor)*

según sea un transistor MOSFET de canal N o de canal P, respectivamente.

Los parámetros L y W son la longitud y anchura del canal, los cuales pueden especificarse en la sentencia del elemento, en el modelo o en la sentencia .OPTIONS. El valor en la sentencia del elemento reemplaza al valor dado en el modelo y éste reemplaza al valor dado en la sentencia .OPTIONS.

AD y AS son las áreas de difusión del drenador y el surtidor. PD y PS son los perímetros de difusión del drenador y el surtidor. NRD, NRS, NRG y NRB son las resistividades relativas del drenador, surtidor, puerta y sustrato, respectivamente.

Por último, M es un multiplicador del elemento con valor por defecto 1, que simula el efecto de varios elementos en paralelo. La anchura efectiva, capacidades de unión y corrientes de unión del MOSFET se multiplicarán por M. El valor de las resistencias parásitas se dividirá por M.

Ejemplos:

Describir un transistor MOSFET de canal N, al que hemos llamado con el número 20, colocado entre los nudos 14, 2, 13 y masa para el drenador, puerta, surtidor y sustrato respectivamente, y con unos parámetros L=25u. y W=12u.:

```
M20 14 2 13 0 NNOM L=25u W=12u
.MODEL NNOM NMOS
```

Describir un transistor MOSFET, al que hemos llamado AMPL, colocado entre los nudos 4, 5, 7 y 3 para el drenador, puerta, surtidor y sustrato, respectivamente, y que corresponde el transistor M2N6792 de la librería PWRMOS.LIB:

```
MAMPL 4 5 7 3 M2N6792
.LIB PWRMOS.LIB
```

1.14. Fuentes Independientes

Aquí veremos la sintaxis de las sentencias que nos permiten describir las fuentes independientes de tensión o intensidad contenidas en el circuito.

1.14.1. Fuentes de tensión independientes

Para introducir en la descripción del circuito una fuente de tensión independiente, utilizaremos la sentencia:

V(nombre) (nudo+) (nudo-) (DC (valor)) (AC (amplitud) (fase)) (especificaciones transitorias)

La fuente puede tener una tensión continua (que se especifica con el término DC seguido del valor en voltios), una tensión sinusoidal de frecuencia variable (que se especifica con el término AC seguido de la amplitud en voltios y el desfase en grados) o bien una tensión variable en el tiempo, como puede ser una señal exponencial, pulsante, sinusoidal, sinusoidal modulada en frecuencia o definida por el usuario en tramos. Los valores DC, AC y especificaciones transitorias por defecto son nulos. Podemos especificar para una fuente valores DC, AC y transitorios independientemente, o bien sólo algunos de ellos. Resaltar que el valor de (fase) para la fuente AC será especificado en grados.

Hay que destacar que la corriente, en PSPICE, se considera positiva cuando entra en la fuente por el nudo positivo (+).

Las especificaciones transitorias pueden ser:

EXP(parámetro = valor)* para formas de onda exponenciales.

PULSE(parámetro = valor)* para formas de onda pulsantes.

PWL(parámetro = valor)* para formas de onda definidas por tramos.

SFFM(parámetro = valor)* para formas de onda sinusoidales moduladas en frecuencia.

SIN(parámetro = valor)* para formas de onda sinusoidales.

Las variables (paso pres) y (tiempo final), usadas por defecto para algunos parámetros de las formas de onda, están definidas en la sentencia .TRAN correspondiente al análisis transitorio. (Paso pres) es el paso entre valores a presentar y (tiempo final) es el valor de tiempo final para el análisis transitorio.

Ejemplos:

Describir una fuente de tensión continua, llamada ALIM, de valor 15V, colocada entre los nudos 1 y masa:

VALIM 1 0 DC 15

Describir una fuente de tensión, llamada con el número 4, colocada entre los nudos 5 y 6, con un valor de tensión continua de 5V y un valor de tensión sinusoidal de frecuencia variable, de 10V de amplitud y un desfase de 30 grados:

V4 5 6 DC 5 AC 10 30

1.14.2. Fuentes de intensidad independientes

Para introducir en la descripción del circuito una fuente de intensidad independiente, utilizaremos la sentencia:

I(nombre) (nudo+) (nudo-) (DC (valor)) (AC (amplitud) (fase)) (especificaciones transitorias)

La fuente puede tener una intensidad continua (que se especifica con el término DC seguido del valor en amperios), una intensidad sinusoidal de frecuencia variable (que se especifica con el término AC seguido de la amplitud en amperios y el desfase en grados) o bien una intensidad variable en el tiempo, como puede ser una señal exponencial, pulsante, sinusoidal, sinusoidal modulada en frecuencia o definida por el usuario por tramos. Los valores DC, AC y especificaciones transitorias por defecto son nulos. Podemos especificar para una fuente valores DC, AC y transitorios independientemente, o bien sólo algunos de ellos. Hay que resaltar que el valor de (fase) para la fuente AC será especificado en grados.

Hay que destacar que la corriente en PSPICE se considera positiva cuando entra en la fuente por el nudo (+).

Las especificaciones transitorias pueden ser:

EXP(parámetro = valor)* para formas de onda exponenciales.

PULSE(parámetro = valor)* para formas de onda pulsantes.

PWL(parámetro = valor)* para formas de onda definidas por tramos.

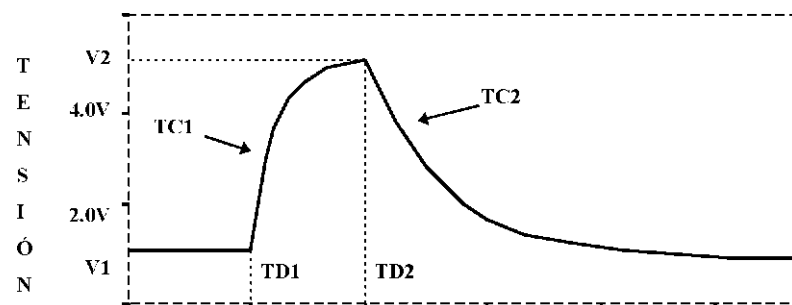
SFFM(parámetro = valor)* para formas de onda sinusoidales moduladas en frecuencia.

SIN(parámetro = valor)* para formas de onda sinusoidales.

Las variables (paso pres) y (tiempo final), usadas por defecto para algunos parámetros de las formas de onda, están definidas en la sentencia .TRAN correspondiente al análisis transitorio. (Paso pres) es el paso entre valores a presentar y (tiempo final) es el valor de tiempo final para el análisis transitorio.

1.14.3. Especificaciones transitorias

Señal exponencial.



Una fuente de tensión exponencial, como la mostrada en la figura anterior, se define con el término:
EXP((V1) (V2) (td1) (tc1) (td2) (tc2))

donde aparecen los términos mostrados en la siguiente tabla:

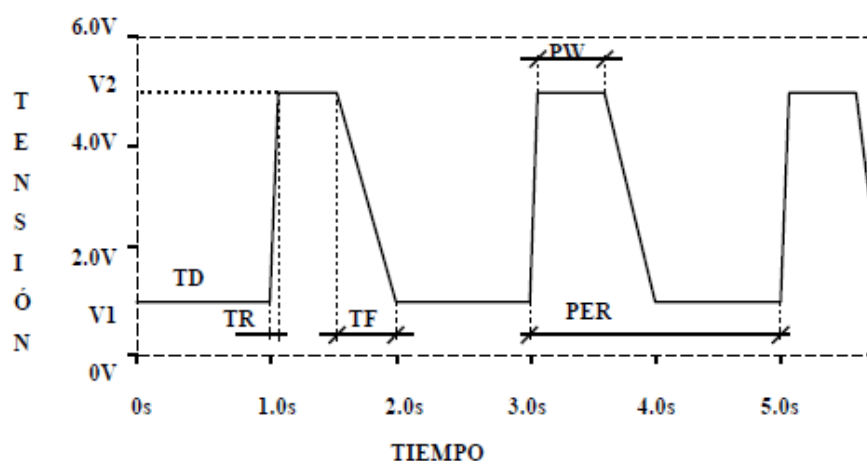
PARÁMETRO	SIGNIFICADO	VALOR POR DEFEC.
(V1)	Tensión inicial en voltios.	Ninguno.
(V2)	Tensión de pico en voltios.	Ninguno.
(td1)	Tiempo de retardo para la subida.	0 Sg
(tc1)	Constante de tiempo de subida.	PASO PRES Sg
(td2)	Tiempo de retardo para la bajada.	(td1)+PASO PRES Sg
(tc2)	Constante de tiempo de bajada.	PASO PRES Sg

Ejemplo:

Describir la fuente de tensión llamada EXP, colocada entre los nudos 2 y 0, que genere la forma de onda exponencial mostrada en la figura anterior:

VEXP 2 0 EXP(1V 5V 1 .2 2 .5)

Señal pulso.



Una fuente de tensión pulsante, como la mostrada en la figura anterior, se define con el término:

PULSE((V1) (V2) (td) (tr) (tf) (pw) (per))

donde aparecen los siguientes parámetros:

PARÁMETRO	SIGNIFICADO	VALOR POR DEFEC.
(V1)	Tensión inicial en voltios.	Ninguno.
(V2)	Tensión del pulso en voltios.	Ninguno.
(td)	Tiempo de retardo.	0 Sg
(tr)	Tiempo de subida.	PASO PRES Sg
(tf)	Tiempo de bajada.	PASO PRES Sg
(pw)	Duración del pulso (estado alto).	TIEMPO FINAL Sg
(per)	Periodo de la señal.	TIEMPO FINAL Sg

La forma PULSE genera una señal de tensión (V1) durante los (td) primeros segundos. Después, la señal crece linealmente desde (V1) hasta (V2) durante los próximos (tr) segundos. Entonces la tensión permanece constante al valor (V2) durante (pw) segundos. A continuación, la señal decrece linealmente, desde (V2) hasta (V1), durante los siguientes (tf) segundos. Se mantiene a una tensión (V1) durante (per)-(tr)-(pw)-(tf) segundos y después vuelve a comenzar el ciclo, exceptuando el tiempo inicial de retardo (td).

Ejemplo:

Definir la fuente de tensión, llamada PULSE, colocada entre el nudo 1 y masa, que genera una señal pulsante como la mostrada en la figura anterior:

```
VPULSE 1 0 PULSE(1V 5V 1S .1S .4S .5S 2S)
```

Señal definida por tramos

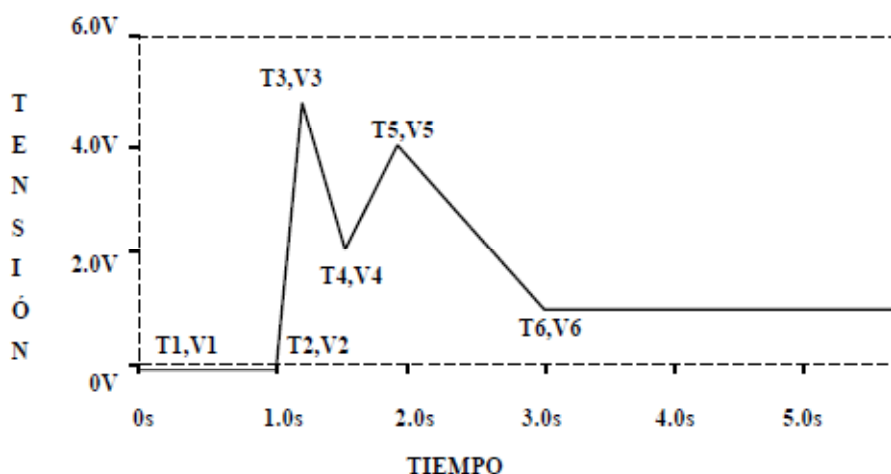
Una fuente de tensión definida por el usuario por tramos rectos, se describe mediante el término:

```
PWL((t1) (v1) (t2) (v2) ... (tn) (vn))
```

donde aparecen los parámetros:

PARÁMETRO	SIGNIFICADO
(tn)	Tiempo de un punto en segundos.
(vn)	Tensión de un punto en voltios.

La forma PWL describe una señal definida por tramos lineales. Cada par de valores tiempo-tensión especifica un punto de la señal. Podemos definir hasta 3995 pares de valores para generar la forma de onda. La tensión entre dos puntos definidos la calcula PSPICE por interpolación lineal.



Ejemplo:

Describir la fuente de tensión, llamada PWL, colocada entre el nudo 1 y masa, que genera una señal definida por tramos como la mostrada en la figura anterior:

```
VPWL 1 0 PWL(0 0V 1 0V 1.2 5V 1.4 2V 2 4V 3 1V)
```

Señal sinusoidal modulada en frecuencia

Una fuente de tensión sinusoidal modulada en frecuencia se define con el término:

SFFM((voff) (vAMPL) (fc) (mod) (fm))

Donde aparecen los parámetros:

PARÁMETRO	SIGNIFICADO	VALOR POR DEFEC.
(voff)	Tensión de offset en voltios.	Ninguno.
(vAMPL)	Tensión de pico en voltios.	Ninguno.
(fc)	Frecuencia de la portadora.	1/TIEMPO FINAL Hz
(mod)	índice de modulación.	0
(fm)	Frecuencia de modulación.	1/TIEMPO FINAL Hz

Ejemplo:

Definir la fuente de tensión, llamada SFFM, colocada entre los nudos 4 y 5, que genera una forma de onda sinusoidal modulada en frecuencia con tensión de pico de 1V, tensión de offset 2V, portadora de 8Hz, índice de modulación 4 y frecuencia de modulación 1Hz:

VSFFM 4 5 SFFM(2V 1V 8HZ 4 1HZ)

Señal sinusoidal

Una fuente de tensión sinusoidal se describe con el término:

SIN((voff) (vAMPL) (freq) (td) (df) (fase))

Donde aparecen los términos:

PARÁMETRO	SIGNIFICADO	VALOR POR DEFEC.
(voff)	Tensión de offset en voltios.	Ninguno.
(vAMPL)	Tensión de pico en voltios.	Ninguno.
(freq)	Frecuencia.	1/TIEMPO FINAL Hz
(td)	Tiempo de retardo.	0 Sg
(df)	Factor de amortiguamiento.	0 Sg ⁻¹
(fase)	Desfase de la señal.	0 grados

El tiempo de retardo es el tiempo en que empieza a atenuarse la señal. Desde 0 a (td) la señal tendrá un valor constante de voff+(1/2)(vAMPL).

Ejemplo:

Definir la fuente de tensión, llamada SIN, colocada entre el nudo 1 y masa, que genera una forma de onda sinusoidal con tensión de offset 2V, tensión de pico 2V, frecuencia 5Hz, tiempo de retardo 1s y desfase de 30 grados:

VSIN 1 0 SIN(2V 2V 5HZ 1S 1 30)

1.15. Fuentes Controladas

En este apartado estudiaremos las sentencias que definen a las diferentes fuentes controladas disponibles en PSPICE. Estas fuentes podrán ser lineales o no lineales. Aquí veremos cómo se describen las fuentes lineales.

1.15.1. Fuentes de tensión controladas por tensión

Para introducir una fuente de tensión controlada por tensión en la descripción del circuito, utilizaremos una de las siguientes sentencias:

E(nombre) (nudo+) (nudo-) (nudo control+) (nudo control-) (ganancia)

Los nudos (nudo+) y (nudo-) son los nudos de conexión de la fuente. La corriente circula desde el nudo (+) al nudo (-) a través de la fuente. Los nudos (nudo control +) y (nudo control -) son los que definen las tensiones de control. Un nudo en particular puede aparecer varias veces en la descripción, y los nudos de salida y de control no necesitan ser diferentes.

Ejemplo:

Describir una fuente llamada SEN, conectada entre los nudos 5 y masa, siendo los nudos de control 7 y 10 y sabiendo que genera una tensión de valor 40 veces superior a la tensión de control:

ESEN 5 0 7 10 40

1.15.2. Fuentes de intensidad controladas por intensidad

Para introducir en la descripción del circuito una fuente de intensidad controlada por intensidad, utilizaremos una de las sentencias:

F(nombre) (nudo+) (nudo-) (fuente de control) (ganancia)

Los nudos (nudo+) y (nudo-) son los nudos de conexión de la fuente. La corriente circula del nudo (+) al nudo (-) a través de la fuente. La corriente de la fuente de tensión de control determina la corriente de salida. La fuente de control debe ser una fuente de tensión independiente, aunque no es necesario que tenga un valor nulo DC.

Ejemplo:

Describir una fuente de intensidad controlada por intensidad, llamada COMP, sabiendo que está conectada entre los nudos 1 y 2, generando una corriente de valor 10 veces superior a la de control (que será la intensidad que circula por la fuente VAUX):

FCOMP 1 2 VAUX 10

1.15.3. Fuentes de intensidad controladas por tensión

Para introducir una fuente de corriente controlada por tensión en la descripción del circuito, utilizaremos una de las siguientes sentencias:

G(nombre) (nudo+) (nudo-) (nudo control+) (nudo control-) (transconductancia)

Los nudos (nudo+) y (nudo-) son los nudos de conexión de la fuente. La corriente circula desde el nudo (+) al nudo (-) a través de la fuente. Los nudos (nudo control +) y (nudo control -) son los que definen las tensiones de control. Un nudo en particular puede aparecer varias veces en la descripción, y los nudos de salida y de control no necesitan ser diferentes.

La entrada de una fuente controlada tiene impedancia infinita, no fluye corriente. En otras palabras, la fuente controlada percibe la tensión entre los nudos de control, pero no está realmente conectada a esos nudos.

Ejemplo:

Describir una fuente de intensidad, llamada 2, posicionada entre los nudos 2 y 6, controlada por la tensión Vc situada entre los nudos 1 y 3, sabiendo que genera una corriente 7 veces superior al valor de la tensión de control Vc:

G2 2 6 1 3 7

1.15.4. Fuentes de tensión controladas por intensidad

Para introducir en la descripción del circuito una fuente de tensión controlada por intensidad, utilizaremos una de las sentencias:

H(nombre) (nudo+) (nudo-) (fuente de control) (transresistencia)

Los nudos (nudo+) y (nudo-) son los nudos de conexión de la fuente. La corriente circula del nudo (+) al nudo (-) a través de la fuente. La corriente de la fuente de tensión de control determina la corriente de salida. La fuente de control debe ser una fuente de tensión independiente, aunque no es necesario que tenga un valor nulo DC.

Ejemplo:

Describir una fuente de tensión, llamada con el número 6 y entre los nudos 3 y 7, controlada por la intensidad I_c que circula por la fuente V2, sabiendo que genera un voltaje 3 veces superior al valor de I_c :

H6 3 7 V2 3

1.16. Análisis

Para los circuitos, tenemos disponibles en SPICE ocho análisis diferentes, que son:

1. Cálculo del punto de trabajo del circuito. Se define con la sentencia .OP.
2. Análisis en continua. Se define con la sentencia .DC.
3. Cálculo de la función de transferencia para pequeña señal. Se define con la sentencia .TF.
4. Cálculo de la sensibilidad en continua. Se define con la sentencia .SENS.
5. Cálculo de la respuesta en frecuencia. Se define con la sentencia .AC.
6. Cálculo del ruido total e individual. Se define con la sentencia .NOISE.
7. Respuesta transitoria (comportamiento en el tiempo). Se define con la sentencia .TRAN.
8. Análisis de Fourier de la respuesta transitoria. Se define con la sentencia .FOUR.

Cada análisis que se desee efectuar al circuito habrá que incluirlo en la definición del mismo en una línea con su correspondiente sentencia (al final del fichero). Los análisis se realizarán en el orden en que aparezcan colocados en la descripción del circuito.

Nosotros sólo utilizaremos los análisis en continua (.DC), punto de trabajo del circuito (.OP) y respuesta transitoria (.TRAN)

1.16.1. Punto de trabajo en continua

Para introducir el cálculo del punto de trabajo en continua en el circuito, usaremos la sentencia:
.OP

Este análisis, calcula el punto de trabajo del circuito y muestra los valores de todas sus fuentes y distintos elementos no lineales; aparecen así tras apartados en el fichero de salida de resultados tras el análisis:

- Una lista de las tensiones de cada nudo.
- Las intensidades de todas las fuentes de tensión y la potencia entregada.
- Una lista de los parámetros de pequeña señal (linealizados) de todos los elementos no lineales.

Si se omite la sentencia .OP en la descripción de un circuito, el punto de trabajo será igualmente calculado, ya que es necesario para calcular los parámetros de pequeña señal de los elementos no lineales (como las fuentes controladas, diodos, transistores, etc.), necesarios para la realización de otros análisis, como el de continua, cálculo de la función de transferencia, de respuesta en frecuencia, etc. En este caso, a la salida sólo aparecerá una lista con las tensiones de cada nudo y no los dos restantes apartados especificados anteriormente.

1.16.2. Análisis en continua.

El análisis en continua, que se especifica con la sentencia .DC, permite hacer un estudio en continua del circuito para un determinado rango de valores de una fuente de tensión o intensidad, de un parámetro de un modelo o de una serie de temperaturas. Así, PSPICE calculará el punto de trabajo del circuito para cada valor de la variable que se está barriendo (es decir, variando de forma automática) y la ganancia de pequeña señal.

Este análisis se realiza en continua, y por lo tanto se consideran todos los condensadores como circuitos abiertos, y las bobinas como cortocircuitos. La información que proporciona el análisis .DC se podría resumir en cómo varía la variable de salida en función de la variable de entrada.

Para introducir este análisis en el circuito, usaremos una de las sentencias:

.DC (lin)* (variable) (valor inicial) (valor final) (incremento) (variable anidada)*
.DC (oct)* (dec)* (variable) (valor inicial) (valor final) (n° puntos) (variable anidada)*
.DC (variable) LIST (valor) (variable anidada)*

La primera forma realiza un barrido lineal de la (variable).

La segunda forma realiza un barrido logarítmico de la (variable).

La tercera forma se usa para darle a la (variable) una serie de valores concretos.

El (valor inicial) para la (variable) puede ser mayor o menor que el (valor final); así, el barrido de la (variable) puede hacerse de forma ascendente o descendente. Los valores de (incremento) y (n° de puntos) han de ser mayores que cero.

Se puede especificar una variable anidada. Esta segunda variable tendrá sus valores propios según el tipo de barrido que deseamos, valores inicial y final e incremento. En este caso, la primera (variable) será el bucle interno; así, para cada valor de la (variable anidada) se realizará un barrido de los valores de la primera.

Así, tenemos las posibilidades:

LIN Barrido lineal. Produce una variación lineal de la (variable), desde el (valor inicial) hasta el (valor final), con saltos según el (incremento) especificado.

OCT Barrido por octavas. Produce una variación logarítmica por octavas de la (variable). El número de puntos calculados por octava será el especificado en (n° puntos).

DEC Barrido por décadas. Produce una variación logarítmica por décadas de la (variable). El número de puntos calculados por década será el especificado en (n° puntos).

LIST En este caso se especifica una lista de valores. No hay valores de comienzo y final, sino una serie de valores precedidos por la palabra LIST que serán los asignados a la (variable) en los distintos análisis.

La (variable) puede ser una de las siguientes:

Una fuente. El nombre de una fuente independiente de tensión o intensidad. Durante el barrido, el valor de la fuente será el correspondiente a la (variable).

Un parámetro de un modelo. Podemos especificar un tipo de modelo, el nombre que le hemos asignado y el parámetro a variar escritos entre paréntesis. Durante el barrido, el valor del parámetro será el correspondiente a la (variable). Los siguientes parámetros no serán válidos: L y W de los transistores MOSFET y los parámetros de temperatura, como, por ejemplo, TC1 y TC2 para las resistencias, etc.

La temperatura. Para ello, usaremos la palabra TEMP en el lugar de (variable). La temperatura se irá ajustando a los distintos valores del barrido. Para cada valor del barrido, todos los componentes del circuito ajustarán sus parámetros a la temperatura en cuestión.

Un parámetro global (una variable). En este caso usaremos el término PARAM seguido del nombre del parámetro que queremos ir cambiando. Durante el barrido, el parámetro global se irá ajustando a los distintos valores especificados y todas las expresiones serán recalculadas.

Tras el análisis .DC, la (variable) volverá a tomar su valor inicial anterior a este análisis.

Ejemplos:

Definir un análisis de barrido en continua variando la fuente de tensión VIN desde un valor de -0.25V hasta 0.25V, con incrementos de 0.05V:

```
.DC VIN -.25 .25 .05
```

Definir un análisis de barrido en continua variando de forma lineal la fuente de intensidad Y2, desde un valor de 5mA hasta -2mA, con incrementos de 0.1mA:

```
.DC LIN Y2 5mA -2mA 0.1mA
```

Definir un análisis de barrido en continua variando la fuente de intensidad IB, desde un valor de 0mA hasta 1mA, con incrementos de 50uA, y realizando para cada valor un bucle interno en el que varíe el valor de la fuente de tensión VCC, desde un valor de 0V hasta 10V, con incrementos de 0.5V:

```
.DC VCC 0V 10V .5V IB 0mA 1mA 50uA
```

Definir un análisis de barrido en continua variando el parámetro R, correspondiente al modelo que hemos denominado RMOD, definido para una/s resistencia/s, desde un valor de 0.9 hasta 1.1, con incrementos de 0.001:

```
.DC RES RMOD(R) 0.9 1.1 .001
```

Definir una análisis de barrido en continua variando logarítmicamente el parámetro IS, correspondiente al modelo llamado QFAST, definido para un/os transistor/es NPN, desde una valor de 1x 10-18 hasta 1x 10-14, realizando el análisis en cinco puntos por década:

```
.DC DEC NPN QFAST(IS) 1E-18 1E-14 5
```

Definir un análisis de barrido en continua variando la temperatura de realización del análisis, con una serie de valores de 0°C, 20°C, 27°C, 50°C, 80°C, 100°C y -50°C:

```
.DC TEMP LIST 0 20 27 50 80 100 -50
```

Definir un análisis de barrido en continua variando el parámetro global (es decir, la variable que hemos definido en el circuito), al que hemos llamado SUPPLY, desde un valor de 7.5 hasta 15, con incremento de 0.5:

```
.DC PARAM SUPPLY 7.5 15 .5
```

1.16.3. Análisis transitorio (respuesta en el tiempo)

Para introducir un análisis transitorio en la descripción del circuito, usaremos la sentencia:

```
.TRAN(/OP)* (paso pres) (tiempo final) (tiempo inicial)* (paso calc)* (UIC)*
```

La sentencia .TRAN realiza un análisis transitorio del circuito. El análisis transitorio calcula el comportamiento del circuito a lo largo del tiempo, desde el instante de tiempo cero hasta el instante especificado en (tiempo final).

Este análisis usa un tiempo de paso interno variable para los cálculos. Así, en intervalos de baja actividad el paso aumenta, y durante intervalos de gran actividad el paso disminuye. El término (paso pres) es el intervalo de tiempo usado para presentar los resultados del análisis transitorio. Los valores intermedios se obtienen por interpolación con un polinomio de segundo orden.

El análisis transitorio siempre comienza para el instante de tiempo cero. No obstante, podemos suprimir en la salida un intervalo de tiempo inicial, especificándolo con el valor de (tiempo inicial).

El paso interno para realizar los cálculos tiene un valor por defecto de (tiempo final)/50. Si queremos aumentar o disminuir este intervalo de tiempo entre cálculos, podemos especificar el nuevo valor con el término (paso calc).

Antes de realizar el análisis transitorio, PSPICE calcula el punto de trabajo del circuito según los valores de las fuentes independientes para este análisis. En el fichero de salida de datos aparecerán las tensiones de los nudos para este punto de trabajo. Si incluimos en la sentencia .TRAN la opción /OP, obtendremos en la salida una información detallada sobre este punto de trabajo.

Si utilizamos el término (UIC), no se calculará el punto de trabajo. Esto se usa cuando tenemos condiciones iniciales especificadas en los condensadores e inductancias y queremos usarlas.

Ejemplos:

Definir una análisis transitorio del circuito de forma que se presenten los resultados del mismo desde el instante inicial hasta el instante de tiempo $T=100\text{nSg}$, con un intervalo de presentación de resultados de 1nSg :

```
.TRAN 1NS 100NS
```

Definir una análisis transitorio al circuito de forma que aparezcan en la salida los resultados desde el instante de tiempo $T=20\text{nSg}$ hasta el instante $T=100\text{nSg}$, con un intervalo de presentación de 1nSg , así como un listado de la información del punto de trabajo. Para el análisis se deberán utilizar las condiciones iniciales de carga de los condensadores y bobinas:

```
.TRAN/OP 1NS 100NS 20NS UIC
```

Definir un análisis transitorio de forma que se presenten los resultados desde el instante de tiempo inicial hasta el instante $T=10\text{mSg}$, con un intervalo de presentación de datos de 1nSg , y asegurándonos que el intervalo de tiempo entre cálculos no sobrepasa los 0.1nSg :

```
.TRAN 1N 10U 0 .1N
```

TEMA 2: **Programa LTSpice**

2.1. Introducción

Como se comentó en el tema anterior, en la asignatura Fundamentos de Electrónica, usaremos como simulador Spice el programa gratuito LTSpice. Para explicar su uso, se realizará un ejemplo sencillo, en el que veremos todo el proceso que abarca una simulación, desde la introducción:

1º Introducción del esquemático.

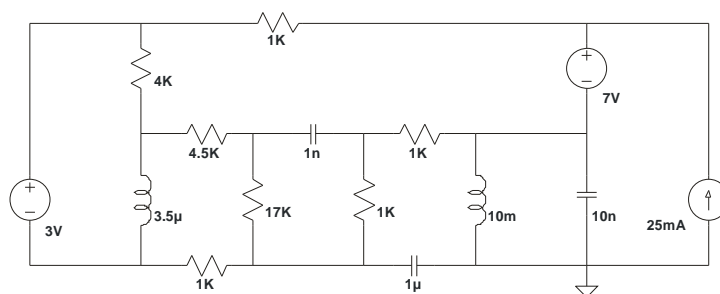
2º Creación de la netlist del circuito, generando un fichero de texto que contiene la descripción Spice del circuito.

3º Simulación y visualización de resultados.


Los ficheros que usaremos para una simulación son:

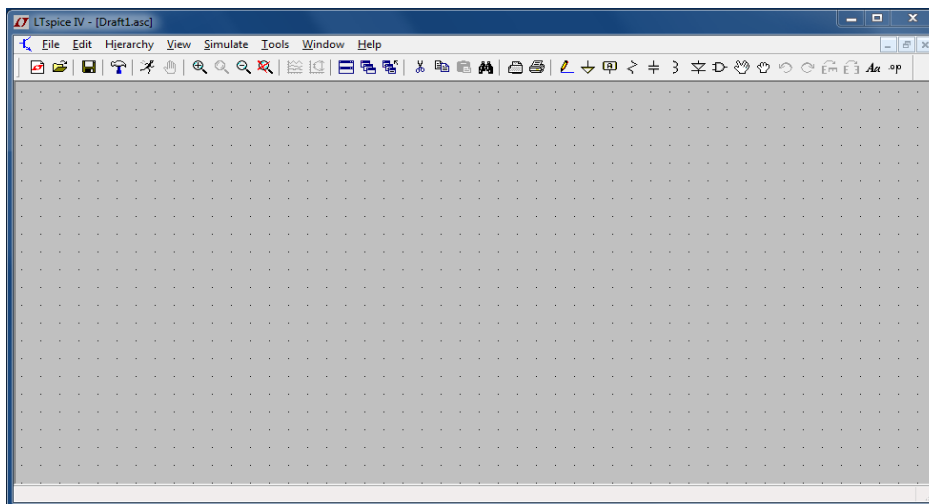
- .asc Contiene el esquemático y es el fichero principal. A partir de él podemos generar todos los demás.
- .net Contiene la netlist generada a partir del esquemático. Se crea al ejecutar la simulación (Run) o en cualquier momento, siempre que pedimos ver la netlist (View->SPICE Netlist).
- .log Contiene información de cómo ha ido la simulación. Se crea al ejecutar la simulación (Run).
- .raw Contiene todos los resultados de la simulación, que pueden ser visualizados sin necesidad de ejecutar de nuevo la simulación. Se crea al ejecutar la simulación (Run).

Utilizaremos el siguiente circuito como ejemplo práctico.








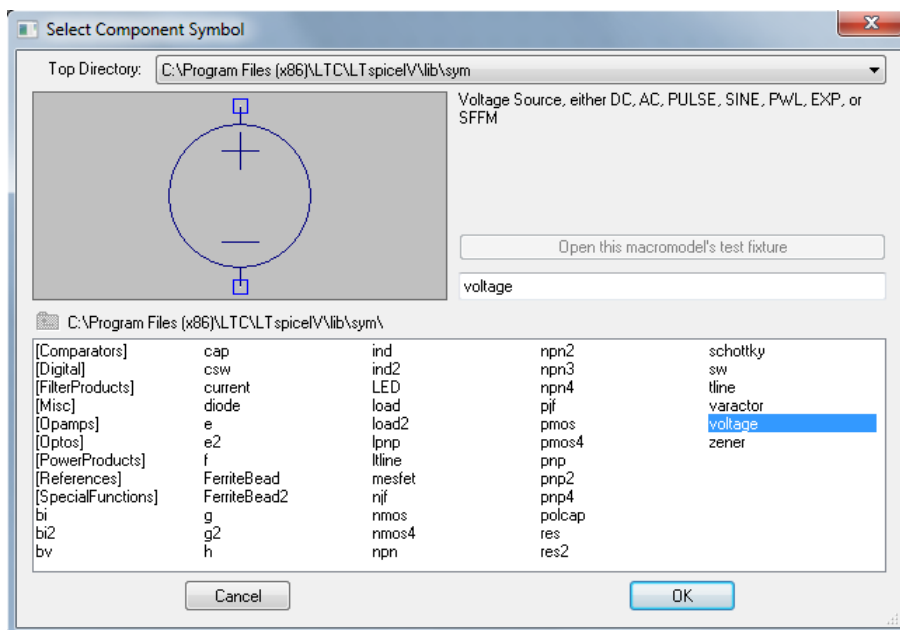
2.2. Creación del esquemático



Para crear el circuito iniciamos el programa LTSpice y creamos un nuevo esquemático desde el icono de la barra de tareas  o desde el menú *File->New Schematic*.



2.2.1. Introducción de elementos.

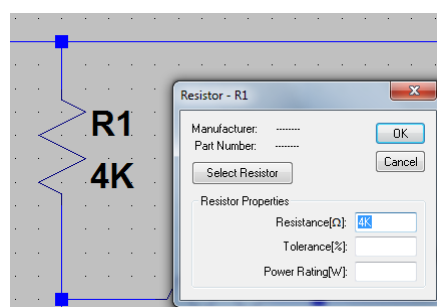
Comenzamos a introducir los distintos elementos que forman el circuito. Los elementos más comunes (resistencias, condensadores, bobinas y diodos), están disponibles desde la propia barra de tareas    , y el resto los seleccionamos de las librerías a través del icono  o desde el menú *Edit-Component*.



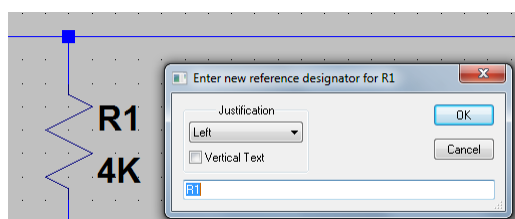
Una vez pulsado [OK] aparece el elemento la pantalla del esquemático donde tengamos el cursor. Antes de pulsar con el botón izquierdo del ratón y colocar el elemento, podemos seleccionar los iconos que nos permiten rotar  y reflejar  el componente.


Una vez introducido un elemento, debemos darle sus valores específicos: valor de resistencia si es una resistencia, valor de capacidad si es un condensador, valor de tensión si es una fuente de tensión, etc.


Para modificar los valores de un elemento debemos colocar el cursor sobre el elemento que queremos modificar, el cursor cambia de forma de cruz a forma de mano, y pulsamos el botón derecho del ratón. Nos saldrá un ventana diferente para cada tipo de elemento.




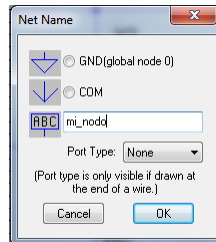
Para modificar los campos visibles de un elemento hay que colocar el cursor sobre el campo que queremos modificar, el cursor cambia de forma de cruz a raya vertical, y pulsamos el botón derecho del ratón. Nos saldrá un ventana diferente para cada tipo de referencia.




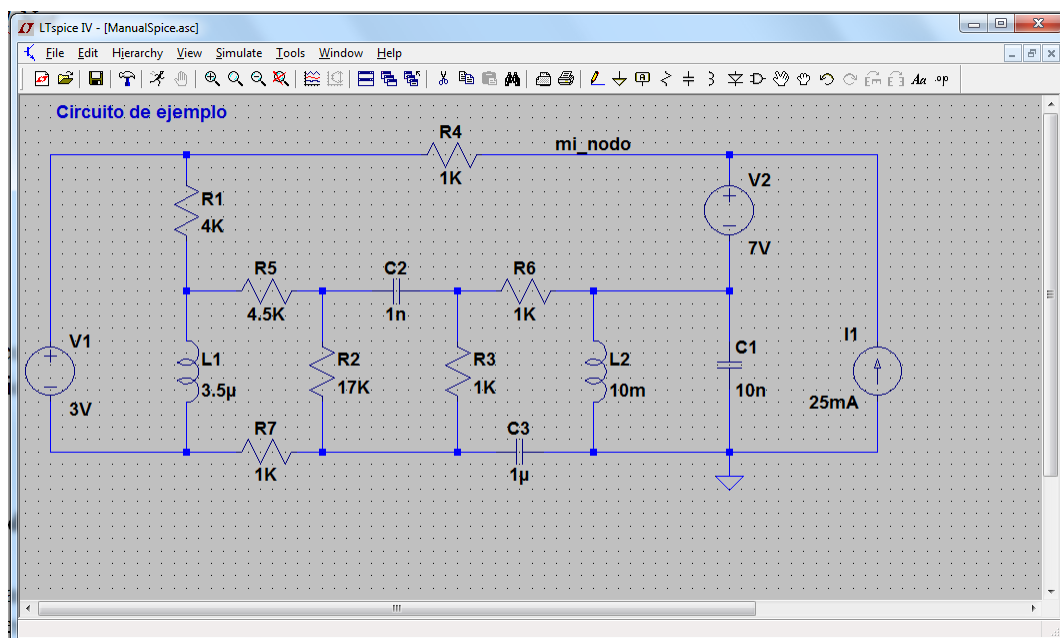
Cuando terminemos de introducir todos los elementos, hay que proceder a interconectarlos con cable mediante el icono  o desde el menú *Edit->Draw Wire*.

El último paso es indicar al simulador cual es el nodo conectado a tierra. Esto se hace colocando un componente especial  o desde el menú *Edit->Place GND* y conectándolo con cable al lugar deseado.

Si queremos identificar de forma fácil algún nodo, podemos darle nombres específicos. Esto lo hacemos con  o desde el menú *Edit->Label Net*.




Por último, con efecto meramente informativo, podemos introducir texto en el esquemático que aparecerán en la netlist como líneas de comentario. Para introducir texto lo hacemos con  o desde el menú *Edit->Text*.




2.2.2. Modificación de elementos.

Siguiendo las pautas anteriores (colocar el cursor sobre un elemento y pulsando el botón derecho del ratón) se puede modificar cuantas veces se desee elementos previamente introducidos.

Para borrar un elemento cualquiera, hay que pulsar la tecla [Supr] o desde el menú *Edit->Delete*, con lo que el cursor cambia de forma de cruz a forma de tijeras, y podemos borrar cualquier elemento sobre el que coloquemos las tijeras y pulsemos el botón izquierdo del ratón. Para salir del modo borrar basta pulsar la tecla [Esc].

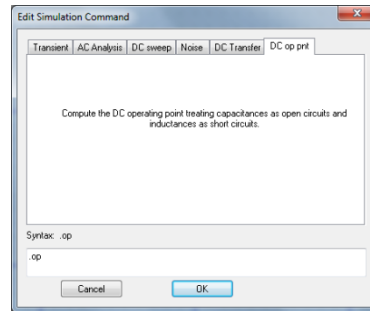
Para mover un elemento o conjunto de elementos sin arrastrar el cable que los conecta, debemos seleccionar  o desde el menú *Edit->Move*. El cursor cambia de forma de cruz a forma de manos. Ahora podemos pinchar con el botón izquierdo del ratón sobre un elemento o seleccionar un área que contenga varios elementos, y mover lo que tengamos seleccionado.

De forma análoga, para mover un elemento o conjunto de elementos arrastrando el cable que los conecta, debemos seleccionar  o desde el menú *Edit->Drag*. El cursor cambia de forma de cruz a forma de manos.

Ahora podemos pinchar con el botón izquierdo del ratón sobre un elemento o seleccionar un área que contenga varios elementos, y mover lo que tengamos seleccionado.

2.2.3. Introducción de directivas Spice

Para poder hacer una simulación Spice, debe estar indicado el tipo de análisis que deseamos realizar. Esto lo haremos desde el menú *Edit->SPICE Analysis*. En el esquemático aparecerá de forma textual el análisis que indiquemos.

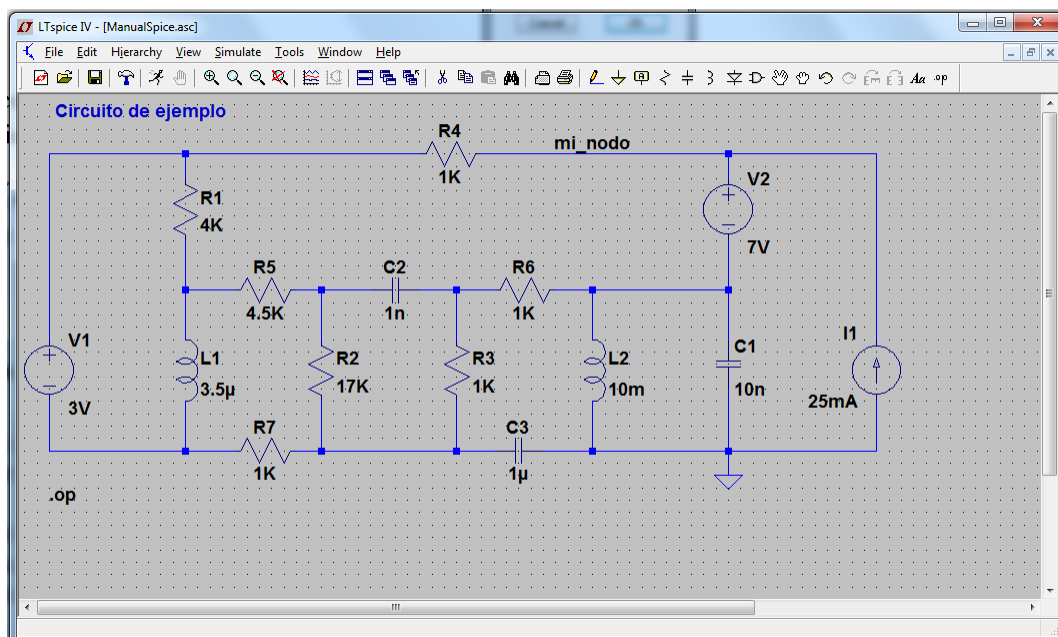


En este ejemplo sólo haremos una simulación para calcular el punto de operación (.op)

Si invocamos el simulador sin haber indicado el análisis a realizar, nos aparecerá la ventana en la que se nos pide el tipo de análisis, antes de lanzar la simulación.

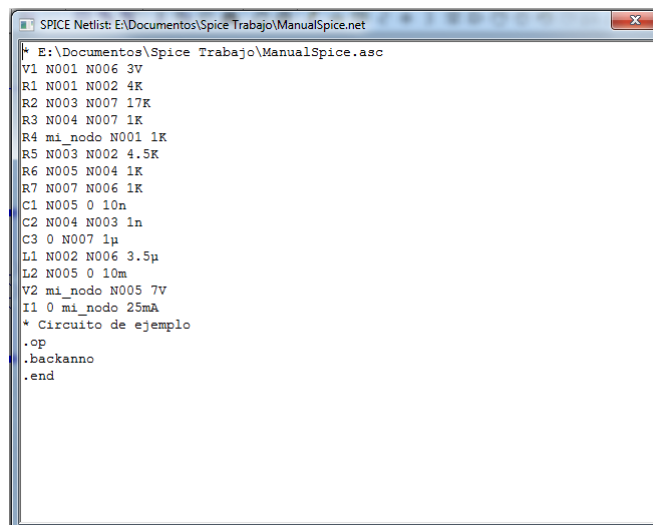
Opcionalmente podemos introducir cualquier sentencia Spice adicional, que aparecerán como líneas en la netlist resultante. Para esto pulsaremos  o desde el menú *Edit->SPICE Directive*.

Este es el esquemático completo.



2.3. Simulación y visualización de resultados.

Una vez terminado el circuito, podemos efectuar su simulación. Es conveniente, aunque no obligatorio, ver la netlist resultante. Esto lo hacemos desde el menú *View->SPICE Netlist*.




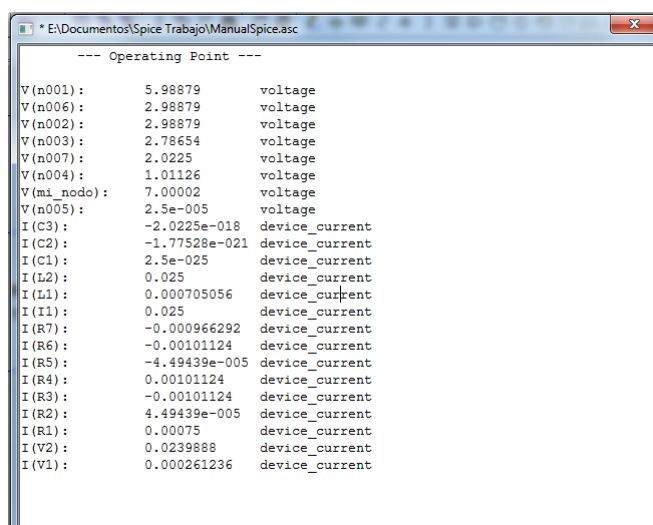
```

E:\Documentos\Spice Trabajo\ManualSpice.net
* E:\Documentos\Spice Trabajo\ManualSpice.asc
V1 N001 N006 3V
R1 N001 N002 4K
R2 N003 N007 17K
R3 N004 N007 1K
R4 mi_nodo N001 1K
R5 N003 N002 4.5K
R6 N005 N004 1K
R7 N007 N006 1K
C1 N005 0 10n
C2 N004 N003 1n
C3 0 N007 1p
L1 N002 N006 3.5p
L2 N005 0 10m
V2 mi_nodo N005 7V
I1 0 mi_nodo 25mA
* Circuito de ejemplo
.op
.backanno
.end
  
```

En la netlist podemos observar:

- La primera línea es el nombre del fichero que contiene el esquemático.
- Aparecen los elementos del esquemático con los nombres que les hayamos dado.
- Los nodos son nombrados de forma autónoma por el programa, salvo los que hayamos etiquetado. En el ejemplo se puso sólo una etiqueta: *mi_nodo*.
- El nodo donde colocamos el elemento de *GND*, el programa lo ha etiquetado como *0*.
- El texto que colocamos en el esquemático (*Circuito de ejemplo*) aparece como una línea de comentario.
- El tipo de análisis elegido (.op) aparece en su respectiva línea.

Para efectuar la simulación pulsaremos  o, desde el menú: *Simulate->Run*. Si sólo se ha pedido un análisis de punto de operación (.op) como en el ejemplo, el resultado de la simulación es un fichero de texto que contiene los valores de todos los elementos del circuito.

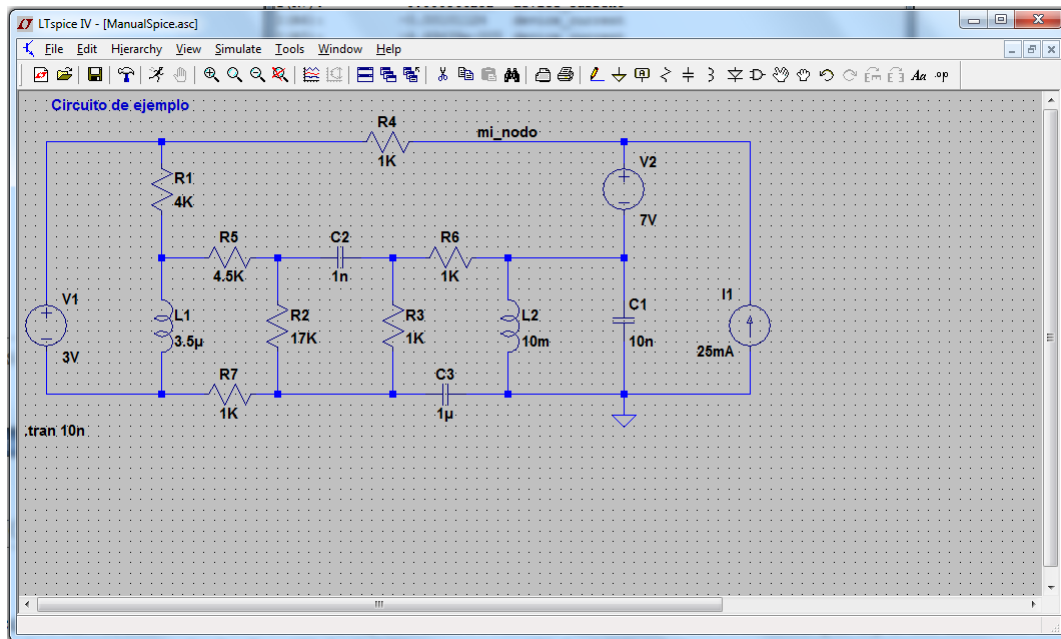


```

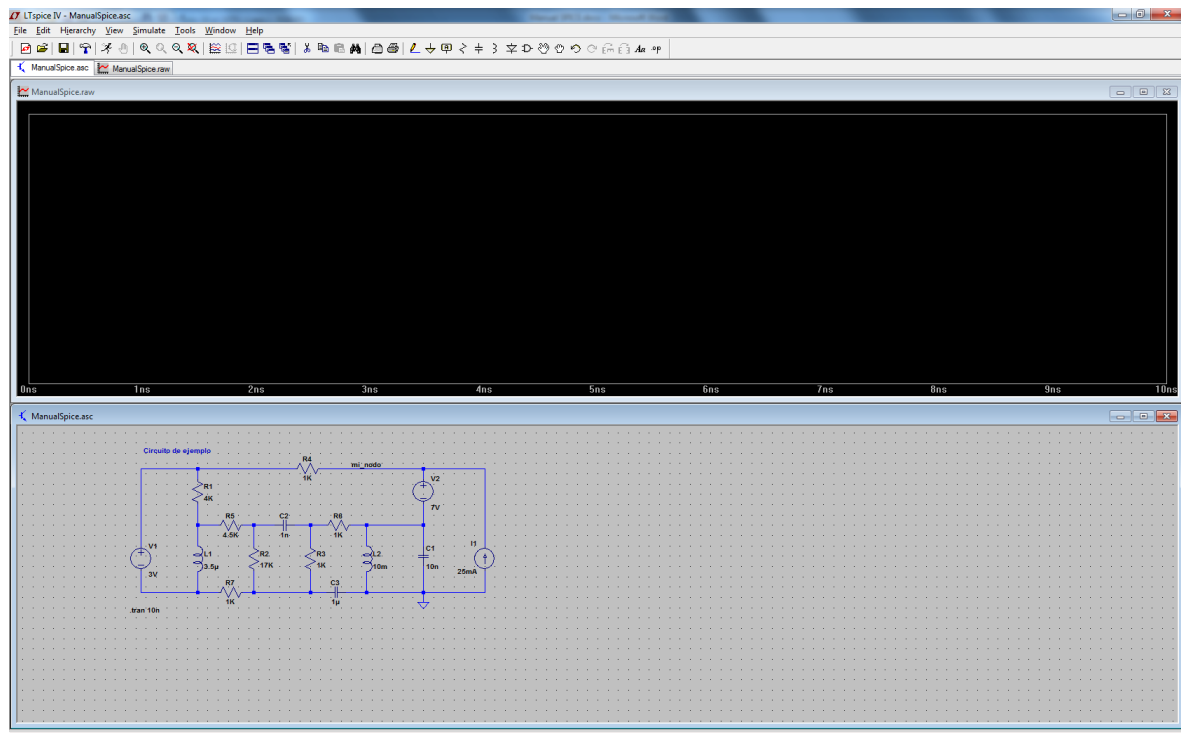
* E:\Documentos\Spice Trabajo\ManualSpice.asc
--- Operating Point ---
V(n001):      5.98879      voltage
V(n006):      2.98879      voltage
V(n002):      2.98879      voltage
V(n003):      2.78654      voltage
V(n007):      2.0225      voltage
V(n004):      1.01126      voltage
V(mi_nodo):   7.00002      voltage
V(n005):      2.5e-005      voltage
I(C3):        -2.0225e-018  device_current
I(C2):        -1.77528e-021  device_current
I(C1):        2.5e-025      device_current
I(L2):        0.025         device_current
I(L1):        0.000705056    device_current
I(I1):        0.025         device_current
I(R7):        -0.000966292    device_current
I(R6):        -0.00101124    device_current
I(R5):        -4.49439e-005  device_current
I(R4):        0.00101124    device_current
I(R3):        -0.00101124    device_current
I(R2):        4.49439e-005  device_current
I(R1):        0.00075        device_current
I(V2):        0.0239888      device_current
I(V1):        0.000261236    device_current
  
```

Para tener acceso al visualizador de resultados, debemos realizar un análisis en continua (.DC) o un análisis transitorio (.TRAN). En nuestro ejemplo vamos a sustituir el análisis (.OP) por un análisis (.TRAN) de 10 ns, aunque en este caso no tenga mucho sentido, debido a que los valores de continua van a permanecer constantes todo el tiempo.

Modificamos el análisis.



Cuando invocamos la simulación nos aparecen dos ventanas: una con el esquemático y otra con el visualizador de resultados.



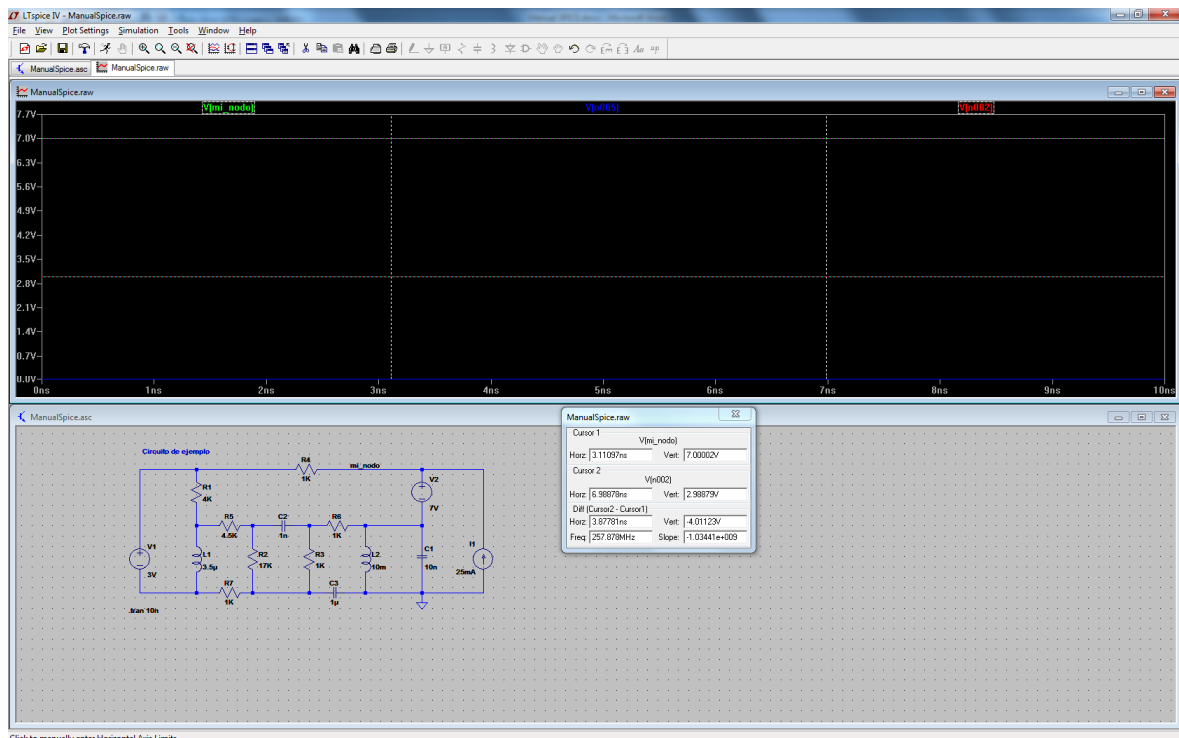
Cuando colocamos el cursor sobre una sección de cable (**un nodo**) en el circuito que tenemos en la pantalla del esquemático, nos cambia de forma de cruz a forma de sonda roja. Si pulsamos con el botón izquierdo del ratón, nos aparece el valor de la tensión en dicho punto en la ventana de visualización.

Cuando colocamos el cursor sobre **un elemento** en el circuito que tenemos en la pantalla del esquemático, nos cambia de forma de cruz a forma de amperímetro. Si pulsamos con el botón izquierdo del ratón, nos aparece el valor de la corriente que circula por dicho elemento en la ventana de visualización.

Para medir valores, el sistema dispone de dos cursores de forma que podemos medir los valores de cada uno de los cursores y la diferencia entre ellos.

En la parte superior de la ventana de visualización aparecen los nombres de las señales que hemos pinchado, con el color que aparecen representadas en la gráfica.

Al colocar el cursor sobre dichos nombres cambia de forma de cruz a forma de mano. Si pulsamos con el botón izquierdo del ratón nos aparece la ventana de cursores y se le asigna el *Cursor 1*. Si pulsamos el botón derecho del ratón podemos elegir si le asignamos a esa señal el *Cursor 1*, el *Cursor 2*, o *los dos cursores*.



En la anterior imagen vemos que el *Cursor 1* está asignado a la tensión del nodo *mi_nodo* y el *Cursor 2* está asignado a la tensión del nodo *n002*.

Otra posibilidad interesante es referenciar las tensiones respecto a la tensión en un nodo en lugar de referenciarlas a tierra, que es lo que se hace por defecto. Para ello debemos seleccionar el nodo de referencia desde el menú: *View->Set Probe Reference*. El cursor cambia de forma de cruz a forma de sonda negra y el punto del esquemático sobre el que pulsemos con el botón izquierdo del ratón, será la tensión de referencia.

Por último puede ser necesario estudiar valores que no son directamente los de una señal del circuito, sino que es función de una o varias de ellas. Para ello podemos añadir manualmente valores y funciones que queremos visualizar. Esto lo hacemos desde el menú: *Plot Settings->Add trace*. Nos aparece la ventana de la siguiente figura.



Dep. Electrónica