

A8 Apéndice a los capítulos T6, T7, T8 y T9

Simuladores eléctricos: SPICE

Simuladores eléctricos

Un simulador es una herramienta informática que permite reproducir sobre el computador el funcionamiento de los circuitos eléctricos y electrónicos, de forma que:

- pueda compararse tal funcionamiento con el deseado hasta comprobar que el diseño funciona correctamente, que cumple las especificaciones que se desean alcanzar;
- se detecte cualquier defecto o anomalía en el funcionamiento del circuito, para corregirlo o, en su caso, verificar que es admisible;
- se conozca cualquier transitorio o cualquier particularidad funcional del circuito para evaluar sus efectos y valorar la necesidad o no de evitarlo.

Los simuladores eléctricos reproducen el comportamiento físico real de los circuitos, es decir, su comportamiento eléctrico, a través de los valores de las tensiones e intensidades en los diversos nudos y ramas del circuito a la largo del tiempo; lo cual permite conocer la respuesta real del circuito frente a las señales de entrada que interesan y, en concreto, los transitorios reales, los tiempos de propagación efectivos, los posibles espurios (*glitches*) u oscilaciones, la violación, en su caso, de los tiempos de anticipación o de mantenimiento de las entradas en las partes secuenciales, etc.

Los simuladores eléctricos actúan a nivel eléctrico: leyes de Kirckoff para el análisis de nudos y de mallas y ecuaciones $I = f(V)$ para describir el comportamiento de los componentes circuitales; han de resolver sistemas de un gran número de ecuaciones, con relaciones I-V no lineales, teniendo que efectuar análisis transitorios con cálculos punto a punto a lo largo del eje de tiempos, para lo cual requieren prolongados tiempos de cálculo (tiempos de CPU) y amplias capacidades de memoria (memoria RAM).

El análisis en términos eléctricos del comportamiento real de los circuitos:

- resulta imprescindible en la simulación de cada una de las celdas o módulos básicos de un circuito integrado (o de un diseño con componentes discretos) para poder calcular su comportamiento temporal (tiempos de propagación, anticipación, mantenimiento, etc., en el caso digital, respuesta en frecuencia, *slew rate*, etc., en el caso lineal) y para estudiar los efectos de carga, fan-out, transitorios, oscilaciones,...
- es necesario para analizar el comportamiento de temporizadores, osciladores, multivibradores,... y, en general, cualquier etapa con realimentación;
- es muy útil en la simulación de pequeños circuitos multietapa;
- es, en la práctica, la única herramienta efectiva para circuitos analógicos;
- pero presenta dificultades (por razones de tiempo y de capacidad de memoria) cuando se trata de circuitos grandes con alto número de componentes.

SPICE *Simulation Program with Integrated Circuit Emphasis* es el simulador eléctrico más difundido y utilizado; fue desarrollado en los años 70 por la Universidad de California (Berkeley) y ha sido mejorado constantemente, existiendo múltiples versiones comerciales, continuamente actualizadas por los suministradores de aplicaciones informáticas.

Descripción circuital: «net list» SPICE

La descripción del circuito se efectúa mediante su **NET LIST**, lista circuital formada por **nudos** numerados (siempre debe existir un nudo de referencia global del circuito, que se numera con 0), **componentes** conectados a los nudos, **valores** de los componentes y **modelos** de los mismos.

La primera línea de la *net list* debe contener el nombre de la misma y es ignorada (aun cuando sea, por equivocación, una línea de descripción de un componente); el final de la «lista circuital» se indica mediante la instrucción **.END**.

Ejemplos de componentes:

	id	= identificador, generalmente un número			
resistencia	Rid	nudo	nudo	valor	
condensador	Cid	nudo	nudo	valor	
autoinducción	Lid	nudo	nudo	valor	
diodo	Did	nudo (P)	nudo (N)	modelo	
transistor	Qid	nudo (C)	nudo (B)	nudo (E)	modelo
MOS	Mid	nudo (D)	nudo (G)	nudo (S)	nudo (subst) modelo
alimentación	Vid	nudo (+)	nudo (-)	valor	
señal	Vid	nudo	nudo	descripción de la señal de tensión	
intensidad	Iid	nudo	nudo	descripción del generador de intens.	
...					

Para los **valores** se puede utilizar notación exponencial o las siguientes letras:

$$\begin{aligned} \mathbf{K} &= \mathbf{E3} \equiv 10^3 & \mathbf{MEG} &= \mathbf{E6} \equiv 10^6 & \mathbf{G} &= \mathbf{E9} \equiv 10^9 & \mathbf{T} &= \mathbf{E12} \equiv 10^{12} \\ \mathbf{M} &= \mathbf{E-3} \equiv 10^{-3} & \mathbf{U} &= \mathbf{E-6} \equiv 10^{-6} & \mathbf{N} &= \mathbf{E-9} \equiv 10^{-9} & \mathbf{P} &= \mathbf{E-12} \equiv 10^{-12} & \mathbf{F} &\equiv 10^{-15} \end{aligned}$$

Los valores pueden incluir la correspondiente unidad (VOLTS, AMPS, HERZ) o abreviaturas de la misma (V, A, H), sin espacio de separación, pero tal indicación es ignorada por el simulador (por lo cual, puede ser errónea o sin sentido).

Los **modelos** pueden estar definidos en una librería: existen librerías típicas de diodos, transistores bipolares, FETs, MOSFETs, amplificadores operacionales, etc.... que incluyen grandes series de tales dispositivos con su nombre comercial; en este caso, es necesario indicar la librería o librerías en las cuales deben buscarse los modelos mediante la instrucción:

.LIB nombre de librería.

Un modelo puede definirse también expresando los valores de los parámetros físicos del dispositivo mediante la instrucción:

.MODEL nombre_del_modelo tipo_de_dispositivo parámetros

Ejemplos: **.MODEL** 1N4148 D IS=100F BV=500

.MODEL tipoN NMOS LEVEL=1 VTO=1 K=40U TOX=5E-8

+ L=0.2E-6 LAMBDA=0.01 L=6U W=2U

El signo + al comienzo de una línea quiere decir que dicha línea es continuación directa de la anterior, es decir, son datos relativos al componente o modelo a que se refiere la línea anterior.

Pueden añadirse líneas de comentarios, comenzando los comentarios con el signo *; al reconocer dicho símbolo, SPICE ignora lo que sigue hasta final de la línea.

Generadores de señal

Un generador de tensión continua o una tensión de alimentación se expresa en la forma:

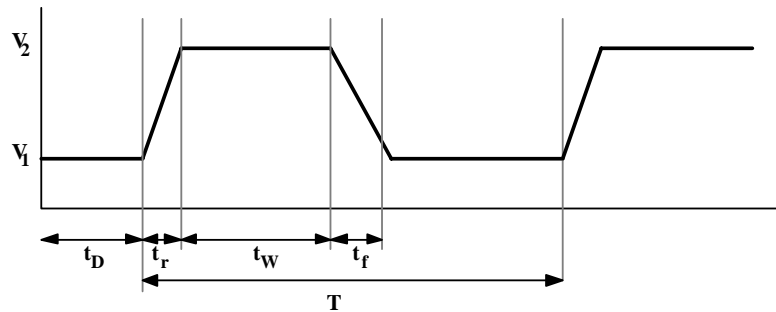
Vid nudo nudo valor_de_dicha_tensión

donde en valor_de_dicha_tensión se indicará la magnitud de la tensión en voltios.

Un generador o señal en onda cuadrada se describe en la forma siguiente:

Vid nudo nudo **PULSE**(V₁ V₂ t_D t_r t_f t_w T)

Este tipo de señal genera una onda rectangular, cuyos niveles son V₁ V₂, que se repite con período T; t_w es la anchura del pulso (en nivel V₂), t_D es el tiempo previo anterior al comienzo del primer pulso durante el cual la tensión es continua (en nivel V₁) y t_r y t_f son los tiempos de subida (paso de V₁ a V₂) y de bajada (paso de V₂ a V₁), respectivamente.



Un pulso o conjunto de pulsos también se puede definir por puntos, expresando las coordenadas (tiempo y tensión) de los puntos de conmutación:

Vid nudo nudo **PWL**(T₁ V₁ T₂ V₂ T₃ V₃ T₄ V₄ ...)

siendo T_i y V_i el tiempo y la tensión, respectivamente de cada punto i de conmutación; es una forma de descripción lineal a trozos, que corresponde a una señal compuesta exclusivamente por segmentos rectilíneos, de forma que en el intervalo que va de T_i a T_{i+1} la tensión varía linealmente de V_i a V_{i+1}.

Una tensión senoidal se describe en la forma:

Vid nudo nudo **SIN**(V_{DC} amplitud frecuencia t_D)

siendo V_{DC} la tensión de continua de la senoide y t_D el tiempo previo anterior al comienzo de la senoide durante el cual la tensión es continua según el valor V_{DC}.

La señal senoidal admite también amortiguamiento exponencial

Vid nudo nudo **SIN**(V_{DC} amplitud frecuencia t_D α)

que corresponde a la siguiente descripción:

$$V_{id} = V_{DC} \quad \text{para } t < t_D$$

$$V_{id} = V_{DC} + \text{amplitud} \cdot e^{-\alpha t'} \cdot \text{sen}(2 \cdot \pi \cdot \text{frecuencia} \cdot t')$$

$$\text{para } t > t_D \quad (t' = t - t_D).$$

Asimismo, la senoide admite desfase:

Vid nudo nudo **SIN**(V_D amplitud frecuencia t_D α θ)

$$V_{id} = V_{DC} + \text{amplitud} \cdot e^{-\alpha t'} \cdot \text{sen}[2 \cdot \pi \cdot \text{frecuencia} \cdot t' - \theta].$$

Otra posibilidad es una entrada senoidal modulada en frecuencia:

Vid nudo nudo **SFFM**(V_{DC} amplitud frec_portadora mod frec_señal)

donde mod es el índice de modulación con el cual actúa la señal sobre la portadora:

$$V_{id} = V_{DC} + \text{amplitud} \cdot \text{sen}[2 \cdot \pi \cdot \text{frec_portadora} \cdot t' + \text{mod} \cdot \text{sen}(2 \cdot \pi \cdot \text{frec_señal} \cdot t')]$$

Obviamente pueden definirse, también, generadores de intensidad:

Iid nudo nudo parámetros_de_la_intensidad

con iguales tipos y la misma forma de descripción que los expresados anteriormente para las tensiones.

El circuito puede contener generadores de tensión o de intensidad dependientes:

Ei nudo nudo nudoA nudoB factor_de_ganancia
 generador de tensión controlado por una tensión (la existente entre los nudos A y B)

Gi nudo nudo nudoA nudoB factor_de_ganancia
 generador de intensidad controlado por tensión (la existente entre los nudos A y B)

Hi nudo nudo Vi factor_de_ganancia
 generador de tensión controlado por una corriente: Vi es un generador de tensión auxiliar (con valor de tensión nula) que se coloca en la rama por la que circula dicha corriente para detectarla.

Fi nudo nudo Vi factor_de_ganancia
 generador de intensidad controlado por una corriente (la que circula por Vi).

Tipos de análisis

Al efectuar la simulación de un circuito, SPICE puede realizar, entre otros, los siguientes tipos de análisis:

RESPUESTA TEMPORAL: Cálculo de tensiones e intensidades en el dominio del tiempo; para ello se utiliza la instrucción:

.TRAN intervalo tiempo_de_simulación

Con esta instrucción, SPICE realiza la simulación temporal del circuito durante el tiempo indicado en el segundo parámetro (tiempo_de_simulación) y almacena los resultados cada intervalo (primer parámetro). Ello no quiere decir que los cálculos los efectúe con $\Delta t = \text{intervalo}$; la variación del tiempo, a efectos del cálculo, es sumamente fina y el simulador ajusta automáticamente el paso de simulación Δt a los valores necesarios para seguir las variaciones de tensión e intensidad con suma precisión.

RESPUESTA FRECUENCIAL: Cálculo de tensiones e intensidades en el dominio de la frecuencia, mediante las instrucciones:

.AC LIN n°_de_puntos frec_inicial frec_final

.AC OCT n°_de_puntos frec_inicial frec_final

.AC DEC n°_de_puntos frec_inicial frec_final

La primera de ellas utiliza el eje de frecuencias en forma lineal y las otras dos lo usan en forma logarítmica: por octavas y por décadas, respectivamente: n°_de_puntos indica el número de puntos, con igual espaciado entre ellos, para los cuales se efectúa el análisis; en el primer caso (LIN) es el número total de puntos entre la frecuencia inicial y la final, mientras que en los otros dos casos se refiere a número de puntos por octava o por década.

El análisis frecuencial requiere una señal de entrada de tipo **AC**

Vi nudo nudo **AC** amplitud

en donde el tipo AC indica que se utiliza una señal senoidal de frecuencia variable, cuya amplitud viene fijada por el parámetro siguiente. Este tipo de análisis permite representar diagramas de Bode ($20 \cdot \log(V_o/V_i)$) y, también, la variación de la fase en relación con la señal de entrada (para lo cual se utiliza el indicativo **VP(nudo)**).

BARRIDO: Respuesta en continua frente a la variación de una variable:

.DC Va valor_inicial valor_final Δ valor

donde Va, señal que realiza el barrido, debe ser definida mediante asignación de nudos:

Va nudo(+) nudo(-)

y el barrido se realiza entre el valor_inicial y el valor_final con un paso de variación igual a Δ valor.

Por ejemplo, para representar la función de transferencia (V_o respecto a V_i):

Vi nudo(+) nudo(-)

.DC Vi valor_inicial valor_final Δ valor

Puede utilizarse, también, para obtener las curvas características de un componente; por ejemplo, las de un transistor MOS, I_D - V_{DS} para diversos valores de V_{GS} :

VDS nudo nudo

VGS nudo nudo

.DC VDS valor_inicial valor_final Δ valor

+ VGS valor_inicial valor_final Δ valor

Se representarán en la misma gráfica varias curvas, una para cada valor de V_{GS} .

Otras posibilidades de análisis se refieren a:

- análisis de Fourier **.FOUR** frecuencia
calcula los coeficientes de Fourier para la frecuencia indicada y ocho armónicos superiores
- cuadrupolos **.TF** Vvalida Ventrada
realiza el cálculo de los parámetros del circuito considerado como un cuadripolo
- análisis de sensibilidad **.SENS** variable
calcula el efecto porcentual que tiene sobre la variable indicada la variación del valor de cada uno de los componentes del circuito.
- análisis de ruido **.NOISE** V(nudo, nudo) fuente_de_ruido
-

El simulador SPICE realiza sus análisis a la temperatura ambiente típica, 27° C (300°K) pero puede indicarse que los realice a otra o a varias temperaturas:

.TEMP temp1 temp2 temp3 (en grados centígrados).

Asimismo se puede ordenar que repita un análisis para varios valores de una variable:

.STEP variable valor_inicial valor_final Δvalor

en este caso, los resultados para los diversos valores de la variable se representarán conjuntamente, es decir, se representarán varias curvas en la misma gráfica, una para cada valor de la variable.

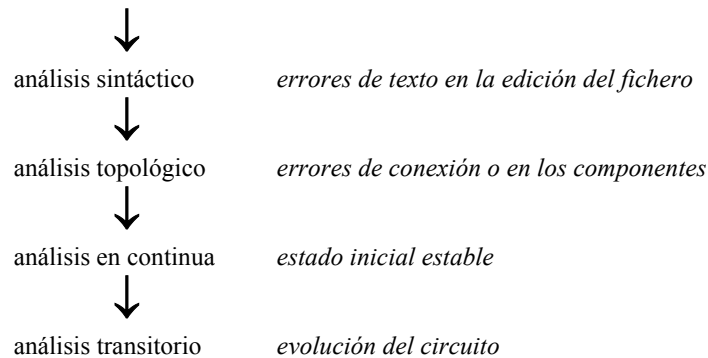
La instrucción **.NODESET** V(nudo)=valor permite fijar las tensiones iniciales (en continua) de los nudos; tales tensiones permanecen fijas en dicho valor en el cálculo del estado inicial, pero evolucionan adecuadamente al calcular la respuesta transitoria a lo largo del tiempo.

Una instrucción análoga **.IC** V(nudo)=valor permite ayudar en los cálculos del punto inicial, indicando un valor para uno o varios nudos, a partir del cual se efectúan los cálculos del estado inicial; a diferencia de **.NODESET**, **.IC** no fija la tensión de los nudos, tan sólo señala el primer valor a utilizar en el correspondiente cálculo iterativo de resolución de las ecuaciones (lo cual ayuda a resolver muchos casos de falta de convergencia).

La instrucción **.PROBE** determina el paso de los resultados de la simulación a un escenario gráfico interactivo que permite representar en pantalla cualquier combinación de las entradas y de las salidas y efectuar todo tipo de ampliaciones y medidas con cursores. Y las instrucciones **.PRINT** y **.PLOT** sirven para imprimir los resultados a través de impresora y de «plotter», respectivamente.

Forma de actuar

El simulador SPICE, en el caso de un análisis temporal (que es el más habitual), actúa de la siguiente forma:



El circuito queda organizado en **nudos** (puntos de conexión de dos o más componentes) y **ramas** (cada una de las conexiones existentes entre dos nudos: en cada rama hay un componente y sólo uno) y las ecuaciones circuitales se construyen según el siguiente proceso:

1. Ecuaciones de conservación de la corriente en los nudos: n ecuaciones
2. Sustitución en ellas de las corrientes en función de las tensiones de rama, a partir de las ecuaciones $I = f(V)$ de los componentes.
3. Sustitución de las tensiones de rama por diferencias de tensión de nudo.
4. Se añaden las ecuaciones de rama en aquellos casos en que las intensidades de rama no han podido ser sustituidas (por ejemplo, para un generador de tensión V_a no puede expresarse la intensidad en función de la tensión y su ecuación de rama es simplemente $V(n) = V_a$) a ecuaciones

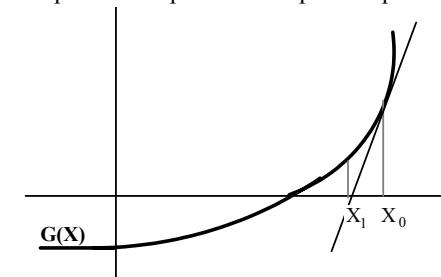
De esta forma resultan $n+a$ ecuaciones (n nudos + a ramas añadidas) con $n+a$ incógnitas, que se resuelven por técnicas de cálculo numérico.

Ahora bien, los dispositivos electrónicos son componentes no lineales; la relación $I = f(V)$ no es lineal y, por tanto, el sistema de ecuaciones tampoco lo es. Para resolverlo, se utilizan métodos numéricos de aproximación lineal mediante iteraciones sucesivas:

Fijado un punto V_0 se aproxima la función $I = f(V)$ por su primer término según el desarrollo en serie de Taylor:

$$I(V) = I(V_0) + \left. \frac{dI(V)}{dV} \right|_{V_0} \cdot (V - V_0) = a + b \cdot \Delta V$$

lo cual permite calcular un nuevo valor V_1 y volver a iterar hasta que V_{i+1} se diferencia mínimamente de V_i ; lo que en realidad se hace es sustituir la curva I-V por su tangente en un punto e ir aproximando punto a punto la solución de las ecuaciones.



Por ejemplo, sea la función $G(X) = 0$; para hallar una solución de la misma, se toma el punto X_0 , se sustituye la curva, propia de la función, por su tangente en dicho punto y se calcula un segundo punto X_1 , repitiéndose con este nuevo valor el proceso anterior y así sucesivamente ... hasta que X_i y X_{i+a} están muy próximos.

Este tipo de métodos de aproximación lineal de ecuaciones no lineales resulta muy eficaz, pero presenta problemas de convergencia por desbordamiento (valores muy altos) o por mínimos locales (en los cuales la iteración puede centrarse recursivamente sobre él y no salir del intervalo en torno al mínimo local).

En caso de no convergencia pueden intentarse las siguientes soluciones:

1. Respecto a valores altos de las impedancias:

- evitar que algún nudo quede eléctricamente aislado, incluyendo, cuando exista dicha posibilidad, una pequeña capacidad (por ejemplo 1 fF) entre dicho nudo y el nudo de referencia (nudo 0); también puede incluirse, en lugar de la capacidad, una resistencia de valor muy alto (pero, en circuitos digitales, da mejor resultado la inclusión de pequeños efectos capacitivos y esto se aproxima más a la situación real).
- limitar el valor máximo de la impedancia que puede presentar una rama, o lo que es lo mismo, limitar el valor mínimo de la admitancia de cualquier rama, para lo cual se emplea la instrucción **.OPTIONS** GMIN=valor (puede darse a la admitancia mínima el valor 1E-7 que corresponde a una impedancia máxima de 10 MΩ).

2. Respecto a la posibilidad de mínimos locales:

- mejorar los modelos de los componentes de forma que no presenten mínimos locales,
- definir valores iniciales para calcular la tensión de los nudos mediante la instrucción **.IC** V(nudo)=valor.

Resolviendo el sistema de ecuaciones, por iteraciones sucesivas, se obtiene un estado inicial estable: la solución estacionaria (de partida); a partir de ese punto ha de calcularse la evolución temporal V(t), I(t), es decir, el análisis transitorio.

El sistema se transforma en diferencial y dependiente del tiempo:

$$f_j \left(t, X, \frac{dX}{dt} \right) = 0 \text{ para } j = 1, 2, 3, \dots, n+a, \text{ y para resolverlo se define un paso o intervalo de tiempo } \Delta t, \text{ que permite sustituir las derivadas por incrementos:}$$

$$\frac{dX}{dt} = \frac{X - X_{\text{anterior}}}{\Delta t}.$$

De esta forma, las ecuaciones vuelven a ser de primer orden (aunque no lineales) y se resuelven por los mismos métodos numéricos de aproximación lineal, a través de iteraciones sucesivas.

De nuevo pueden producirse problemas de convergencia y, además, el tamaño del paso influye en tal posibilidad. Por otra parte, el tamaño del paso de simulación Δt introduce los correspondientes errores de aproximación, al sustituir una derivada puntual por un incremento que corresponde a un intervalo finito.

Para minimizar tales errores de aproximación y evitar, en lo posible, los problemas de convergencia, el simulador SPICE utiliza tamaños de paso variable (disminuyendo el tamaño en los puntos de fuerte variación de tensiones o intensidades y aumentándolo cuando su variación es lenta). Por ejemplo, en la simulación de un circuito digital, el tamaño del paso se hace sumamente pequeño en el entorno de los flancos de las señales de entrada y pasa a ser grande en las situaciones estables: estados booleanos **0** y **1**.

Subcircuitos

El simulador SPICE admite la utilización de subcircuitos como módulos para la construcción de circuitos más complejos, sin necesidad de detallar su contenido más que una sola vez. La declaración de subcircuito se realiza con las siguientes instrucciones (de cabecera y de final de subcircuito):

.SUBCKT nombre_del_subcircuito nudo nudo nudo nudo

descripción (net list) del subcircuito

.ENDS

Los nudos declarados en la cabecera serán los necesarios para la conexión externa del subcircuito en su utilización como tal y su numeración ha de corresponderse con de la «lista circuital» que describe el subcircuito. El nudo 0 es siempre un nudo global para todo el circuito, es decir, no puede utilizarse el 0 para numerar un nudo de un subcircuito que no vaya a coincidir con el nudo 0 de referencia del circuito global.

Para insertar un módulo o subcircuito en un circuito mayor se utiliza la instrucción:

Xi**** nudo nudo nudo nudo nombre_del_subcircuito

El orden de los nudos ha de corresponderse con el indicado en la cabecera de la declaración del subcircuito, pero su numeración (al utilizarlo como tal) será la propia del circuito global en el que se incluye.

Un mismo número de nudo (por ejemplo, el 2) puede ser utilizado dentro de la descripción de uno o de varios subcircuitos y, también, en el circuito global; pero, en cada caso, se refiere a un nudo diferente, es decir, los nudos dentro de los subcircuitos y los de fuera de ellos no quedan conectados por asignación del mismo número. La única excepción a esto es el nudo 0, que es común, con tal número a todos los subcircuito y al circuito global, pues es el nudo de referencia para todos ellos.

Un mismo módulo o subcircuito puede ser utilizado con valores distintos para alguno o algunos de los valores de sus componentes (por ejemplo, un inversor MOS puede ser utilizado con diferentes valores de la anchura de sus transistores). Es decir, la declaración de un subcircuito admite la parametrización (expresar en función de un parámetro) de alguna de sus características, para lo cual se utiliza la declaración:

.SUBCKT nombre_del_subcircuito nudo nudo
+ **PARAMS:** parámetro = valor_por_defecto

descripción (net list) del subcircuito y dentro de ella: variable = {parámetro}

.ENDS

«parámetro» representa el nombre del parámetro que se va a utilizar.

Al utilizar el subcircuito se indica el valor a dar al parámetro:

```
Xid      nudo nudo nudo ..... nombre_del_subcircuito
+          PARAMS: parámetro = valor_actual
```

Si al incluir un subcircuito se asigna un valor al parámetro utilizado (como sucede en la instrucción anterior) el subcircuito se construye dando tal valor a la variable parametrizada; en caso contrario (si no se asigna valor al parámetro), se construye con el valor por defecto expresado en la declaración del subcircuito.

Ejemplo: Cuatro inversores seguidos «en escalera» (con la anchura de sus transistores progresivamente creciente).

Circuito de cuatro inversores en escalera

*Subcircuito inversor 1 entrada y 1 salida

```
.SUBCKT inver 1 2 PARAMS: a = 1.5U
```

```
M1 2 1 3 3 mipmos L=1U W={a}
```

```
M2 2 1 0 0 minmos L=1U W={a}
```

```
VCC 3 0 5
```

```
.model mipmos PMOS level=1 KP = 15U VTO = -1 TOX = 50E-9
```

```
.model minmos NMOS level=1 KP = 40U VTO = 1 TOX = 50E-9
```

```
.ENDS
```

* Conexión de los 4 inversores

```
X1 1 2 inver
```

```
X2 2 3 inver PARAMS: a = 15U
```

```
X3 3 4 inver PARAMS: a = 150U
```

```
X4 4 5 inver PARAMS: a = 1500U
```

* Carga capacitiva de la salida

```
C4 5 0 50P
```

* Tensión de entrada: onda cuadrada de 40 ns de período

```
Vi 1 0 PULSE(0 5 10N 0 0 20N 40N)
```

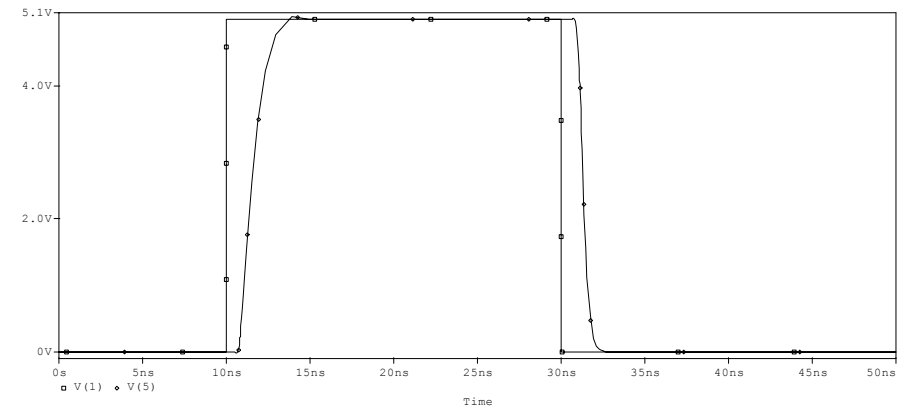
* Análisis temporal

```
.TRAN 0.01N 50N
```

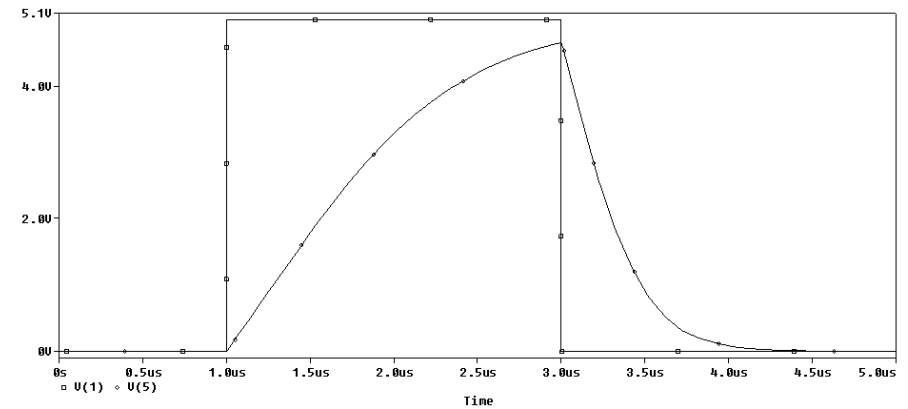
```
.PROBE
```

```
.END
```

En la página siguiente se muestra el resultado de la simulación de este circuito de 4 inversores en escalera (el primero de $W = 1,5 \mu\text{m}$ y creciendo W diez veces en cada uno de los siguientes) y, a efectos comparativos, se muestra el resultado de la simulación con los cuatro inversores iguales (todos ellos de $W = 1,5 \mu\text{m}$).



Respuesta del circuito de 4 inversores «en escalera» a un pulso de tensión (el eje X, eje de tiempos, se presenta con 5 ns/división)



Respuesta a un pulso de tensión de un circuito de 4 inversores iguales, de tamaño mínimo (téngase en cuenta que, en esta segunda gráfica, el eje de tiempos está en 500 ns/división, es decir, sus divisiones son 100 veces mayores, en tiempo, que en la gráfica anterior).