Universidad
Politécnica
de Cartagena

www.upct.es

Universidad Politécnica de Cartagena

LTSPICE

Linear Technology Simulation Program with Integrated Circuits Emphasis (Programa de simulación con énfasis en circuitos integrados).

LTspice IV is a high performance SPICE simulator, schematic capture and waveform viewer with enhancements and models for easing the simulation of switching regulators. Our enhancements to SPICE have made simulating switching regulators extremely fast compared to normal SPICE simulators, allowing the user to view waveforms for most switching regulators in just a few minutes. Included in this download are LTspice IV, Macro Models for 80% of Linear Technology's switching regulators, over 200 op amp models, as well as resistors, transistors and MOSFET models.

¿Por qué/para qué un simulador de circuitos?

*Un **simulador de circuitos electrónicos** es una herramienta de software utilizada por profesionales en el campo de la electrónica y los estudiantes de las carreras de tecnologías de la información. Ayuda a crear algún circuito que se desee ensamblar, ayudando a entender mejor el mecanismo, y ubicar los fallos dentro del mismo de manera sencilla y eficiente. ©Wikipedia.*

VENTAJAS

*Utilizar un simulador de circuitos le permite al ingeniero electrónico **hacer pruebas sin correr el riesgo de dañar el circuito**, si eso llegase a ocurrir, implicaría mayor gasto de material semiconductor.*

Cuando un circuito trabaje correctamente en el simulador, será más fácil montarle en una protoboard, y se puede tener la seguridad de que el circuito funcionará correctamente.

*Con el simulador **se puede hallar de manera más fácil los errores y problemas** que surgen a la hora de ensamblar los circuitos eléctricos, con algunas herramientas que los programas ya cuentan como por ejemplo: multímetros, generadores de voltaje u osciloscopios.*

*Algunos programas **cuentan con diferentes vistas al circuito que se está montando**. Se puede observar como si se estuviese conectando en un protoboard, o como un diagrama de conexiones. También se puede ver como una placa de circuitos la cual se puede mandar a fabricar con alguna compañía y así obtendrá un trabajo final funcionando.*

INCONVENIENTES

*Algunos simuladores de circuitos no están lo suficientemente actualizados, y **no cuentan con todos los chips del mercado**, y eso seria un contratiempo para el diseñador, ya que deberá darse a la tarea de fabricar su propio semiconductor, y eso podría tomarse su tiempo.*

*Cuando no se sabe como manejar el programa de simulación, genera retrasos en los diseños, **se deben estudiar de manera completa todos los componentes y opciones** que tiene el programa, para poder realizar el trabajo de manera correcta.*

LTspice IV

http://www.linear.com/designtools/software/

LINEAR TECHNOLOGY

- **Download LTspice IV for Windows** (Updated January 20, 2015) >40Mb
- **Download LTspice IV for Mac OS X 10.7+**
- LTspice Information Flyer & Shortcuts
- Mac OS X Shortcuts
- LTspice Getting Started Guide
- LTspice Blog
- LTspice Demo Circuit Collection
- View Upcoming LTspice Seminars

Follow LTspice on Twitter!

View the LTspice Video Channel

twitter

www.twitter.com/ltspice

Follow @ LTspice on Twitter for up-to-date information on models, demo circuits, events and user tips.

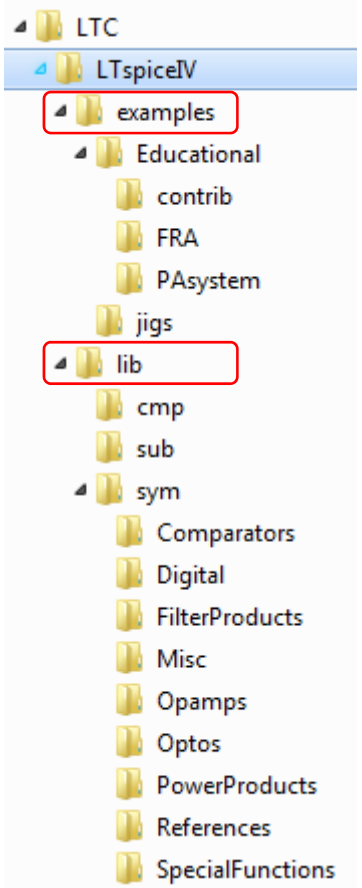
FREE LTspice® Download

www.linear.com/ltspice

Included in the download is a complete and fully functional SPICE program, help files, macro models for Linear Technology's power products, over 200 op amp models, as well as models for resistors, transistors and MOSFETs.

video.linear.com/ltspice

View Instructional Videos.



LTSpice se instala en: C:\Archivos de programa\LTC\LTspiceIV.

En el raíz de LtspiceIV aparecen diversos archivos, entre los que se encuentra el ejecutable de la aplicación *scad3.exe*, así como dos carpetas, *examples* y *lib*.

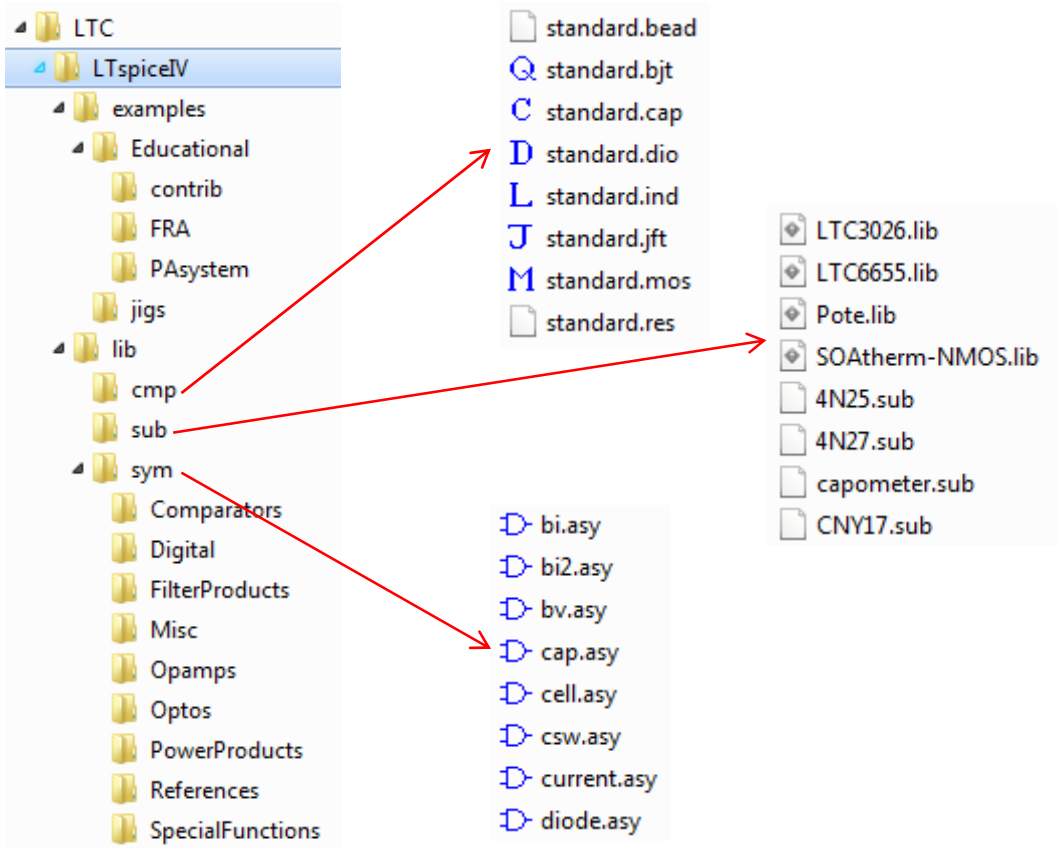
La carpeta *examples* se subdivide en otras subcarpetas que contienen diversos ejemplos de circuitos para simular.

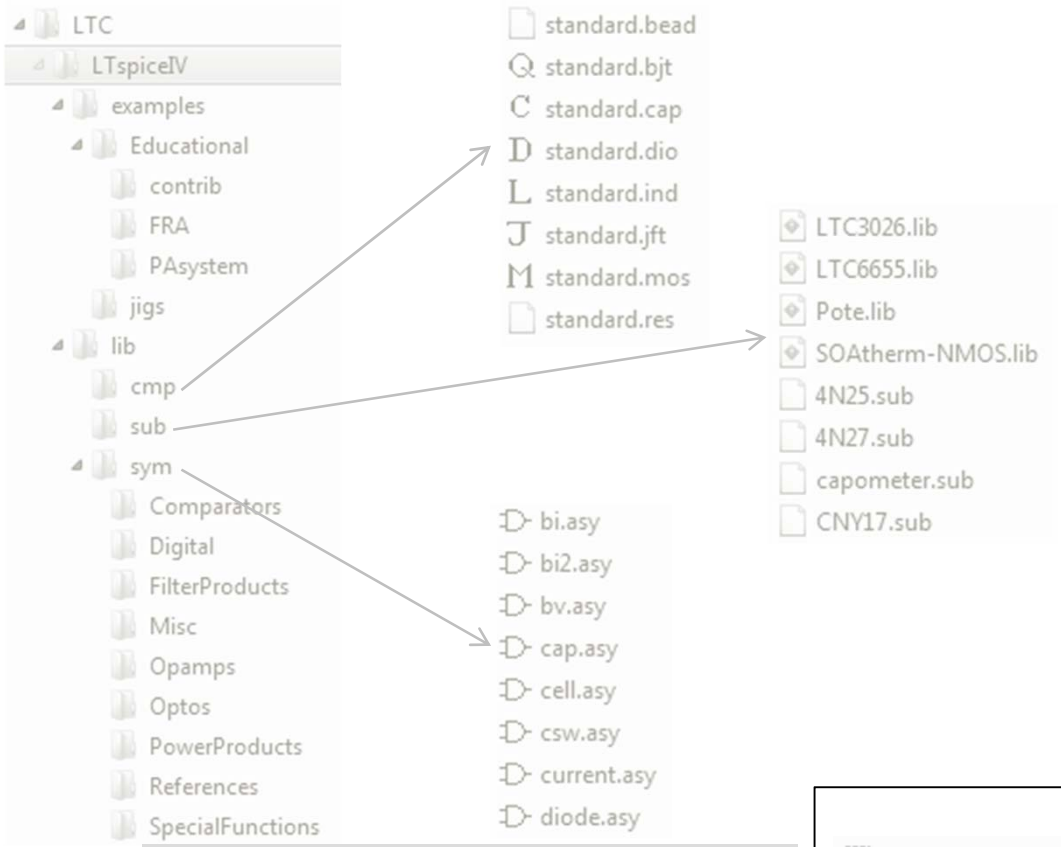
La carpeta *lib* contiene tres subcarpetas:

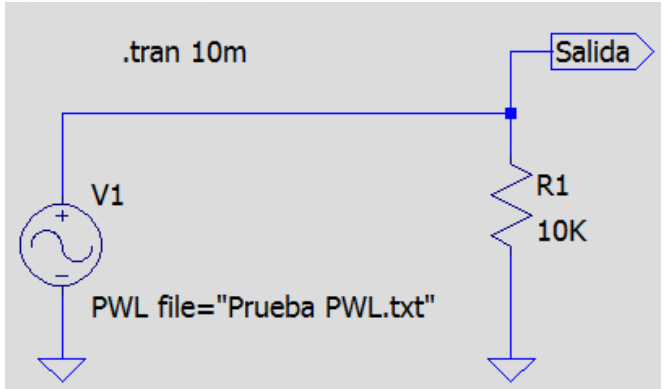
- **cmp** Contiene las librerías elementales de componentes Spice: Resistencia, Bobina, Condensador, Diodo, BJT, FET y MOS.
- **sub** Contiene librerías de componentes, modelos y subcircuitos.
- **sym** Contiene símbolos de componentes.

Todas las librerías y archivos de símbolos son archivos de texto, incluidos los archivos que contienen los circuitos de simulación.

Todos los archivos de texto han de comenzar por una línea de comentario, que por ser la primera no es necesario que se preceda del asterisco. Si hubiese más líneas de comentario, sí deberán de ser precedidas por un asterisco o un punto y coma.


LTC
LTspiceIV
examples
Educational
contrib
FRA
PAsystem
jigs
lib
cmp
sub
sym
Comparators
Digital
FilterProducts
Misc
Opamps
Optos
PowerProducts
References
SpecialFunctions
standard.bead
standard.bjt
standard.cap
standard.dio
standard.ind
standard.jft
standard.mos
standard.res
LTC3026.lib
LTC6655.lib
Pote.lib
SOAtherm-NMOS.lib
4N25.sub
4N27.sub
capometer.sub
CNY17.sub
bi.asy
bi2.asy
bv.asy
cap.asy
cell.asy
csw.asy
current.asy
diode.asy





Ejemplo de esquemático

Prueba PWL.txt	Definición de puntos de onda
Prueba.asc	Esquema del circuito
Prueba.log	Bitácora
Prueba.net	Fichero de Netlist
Prueba.op.raw	Visor de gráficas
Prueba.plt	Tipo de gráficas a mostrar
Prueba.raw	Visor de gráficas

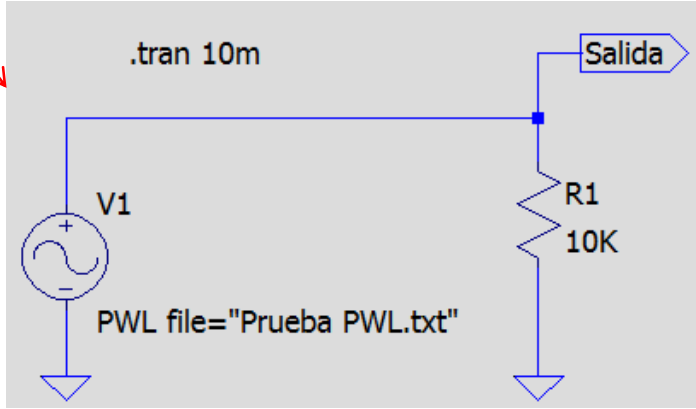
- Prueba PWL.txt
- Prueba.asc
- Prueba.log
- Prueba.net
- Prueba.op.raw
- Prueba.plt
- Prueba.raw

Definición de puntos de onda
Esquema del circuito
Bitácora
Fichero de Netlist
Visor de gráficas
Tipo de gráficas a mostrar
Visor de gráficas

Prueba PWL.txt: Bloc de notas

Archivo Edición Formato Ver Ayuda

0	0
1m	1
2m	0.5
2.4m	-1
3m	2.1
5m	-0.6



- Prueba PWL.txt
- Prueba.asc
- Prueba.log
- Prueba.net
- Prueba.op.raw
- Prueba.plt
- Prueba.raw

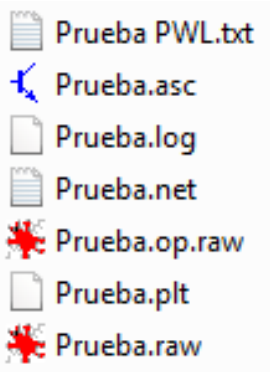
Definición de puntos de onda
Esquema del circuito
Bitácora
Fichero de Netlist
Visor de gráficas
Tipo de gráficas a mostrar
Visor de gráficas

C:\LTSpice practicas\Prueba.log - Notepad++

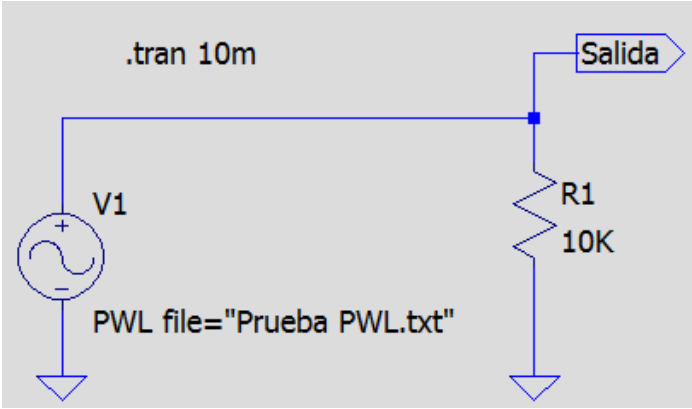
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Macro Ejecutar

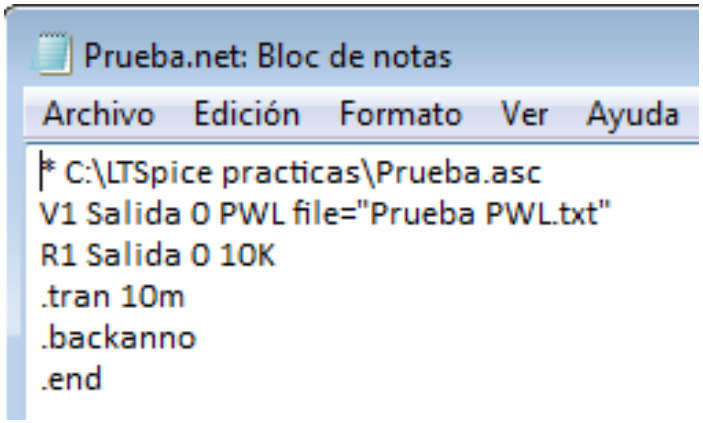
Prueba.log

```
Circuit: * C:\LTSpice practicas\Prueba.asc

.OP point found by inspection.

Date: Thu Feb 05 13:11:38 2015
Total elapsed time: 0.156 seconds.

tnom = 27
temp = 27
method = modified trap
totiter = 2112
traniter = 2112
tranpoints = 1057
accept = 1057
rejected = 0
matrix size = 2
fillins = 0
solver = Normal
Matrix Compiler1: 36 bytes object code size  0.1/0.2/[0.1]
Matrix Compiler2: 96 bytes object code size  0.1/0.3/[0.1]
```



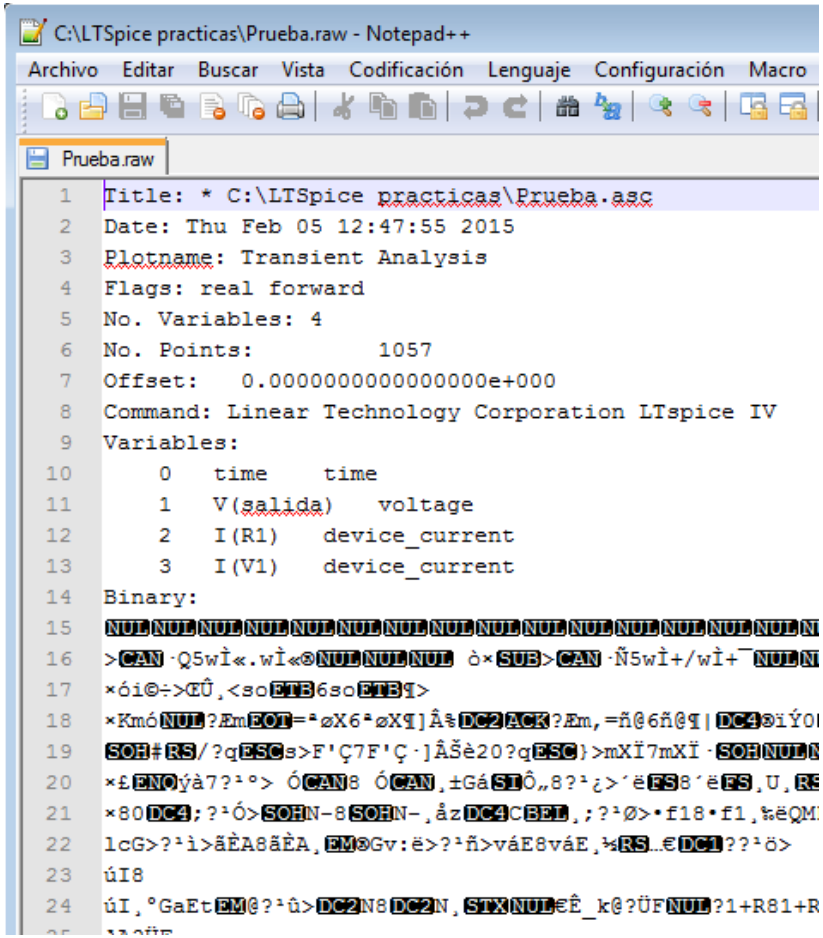
Definición de puntos de onda
Esquema del circuito
Bitácora
Fichero de Netlist
Visor de gráficas
Tipo de gráficas a mostrar
Visor de gráficas





- Prueba PWL.txt
- Prueba.asc
- Prueba.log
- Prueba.net
- Prueba.op.raw
- Prueba.plt
- Prueba.raw

Definición de puntos de onda
Esquema del circuito
Bitácora
Fichero de Netlist
Visor de gráficas
Tipo de gráficas a mostrar
Visor de gráficas



- Prueba PWL.txt
- Prueba.asc
- Prueba.log
- Prueba.net
- Prueba.op.raw
- Prueba.plt
- Prueba.raw

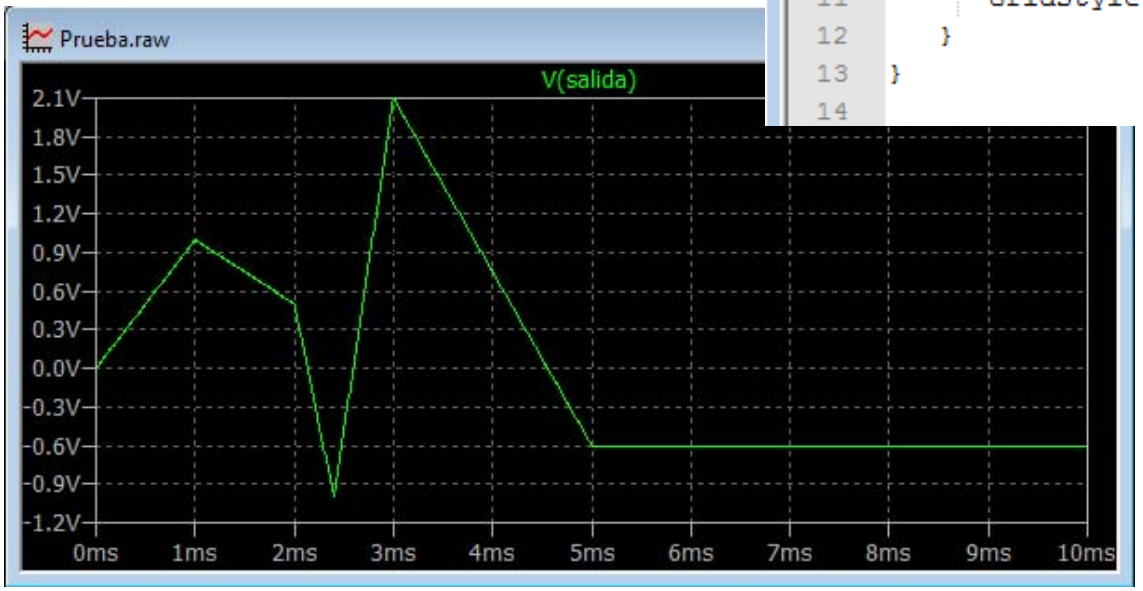
Definición de puntos de onda
Esquema del circuito
Bitácora
Fichero de Netlist
Visor de gráficas
Tipo de gráficas a mostrar
Visor de gráficas

C:\LTSpice practicas\Prueba.plt - Notepad++

Archivo Editar Buscar Vista Codificación Lenguaje Configura

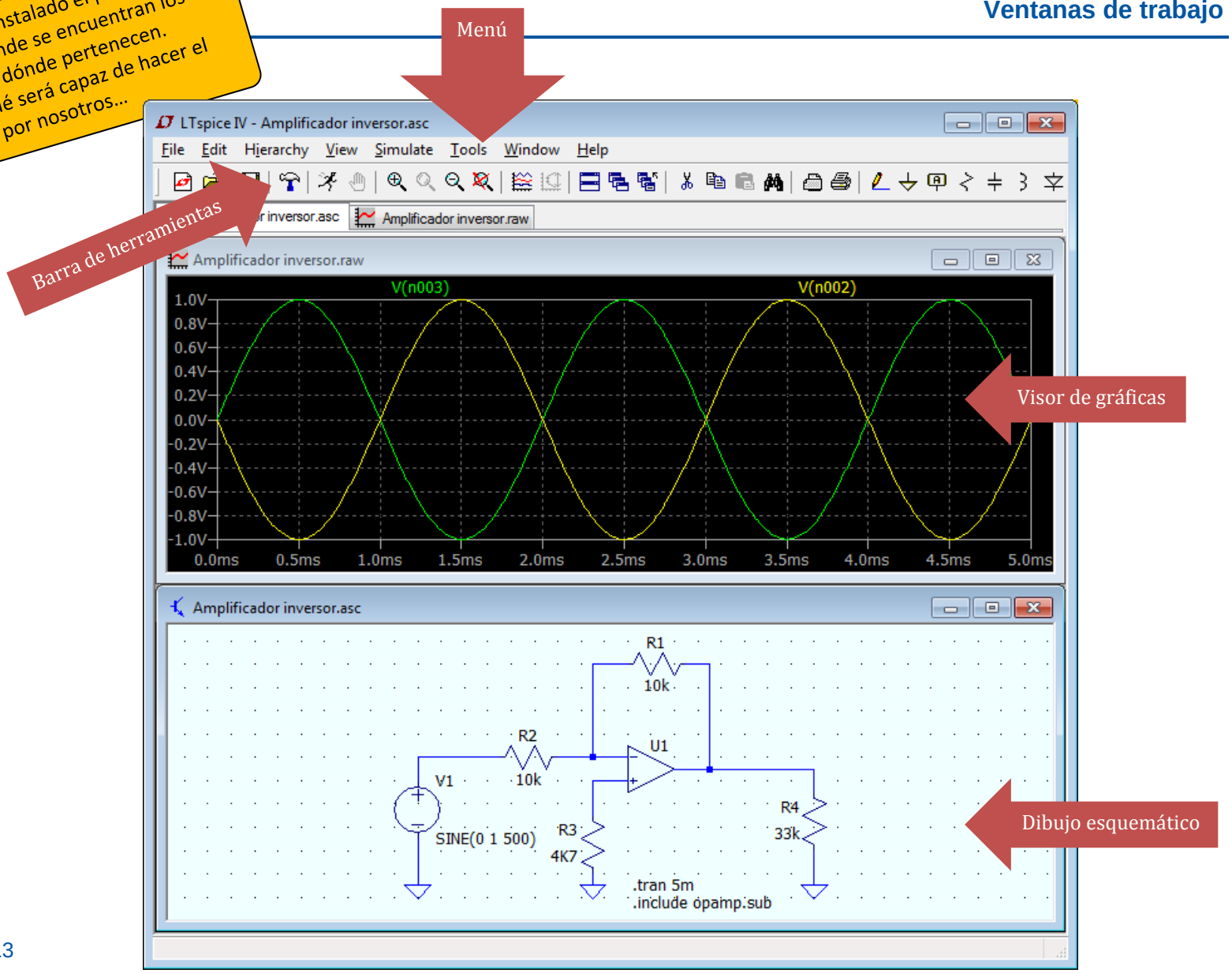
Prueba.plt

```
[Transient Analysis]
{
    Npanes: 1
    {
        traces: 1 {524290,0,"V(salida)"}
        X: ('m',0,0,0.001,0.01)
        Y[0]: (' ',1,-1.2,0.3,2.1)
        Y[1]: ('µ',0,1e+308,1e-005,-1e+308)
        Volts: (' ',0,0,1,-1.2,0.3,2.1)
        Log: 0 0 0
        GridStyle: 1
    }
}
```



Ventanas de trabajo

Ya tenemos instalado el programa, sabemos dónde se encuentran los ficheros y a dónde pertenecen. Veamos qué será capaz de hacer el programa por nosotros...



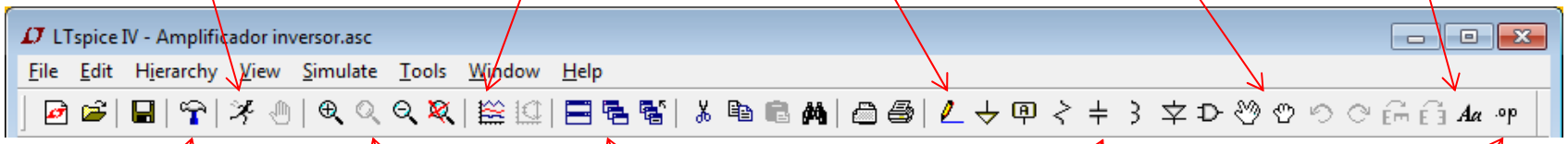
Ejecutar simulación

Visualizar gráficas

Dibujar cables

Mover/Arrastrar

Insertar texto



Control panel

Zoom

Organizar ventanas

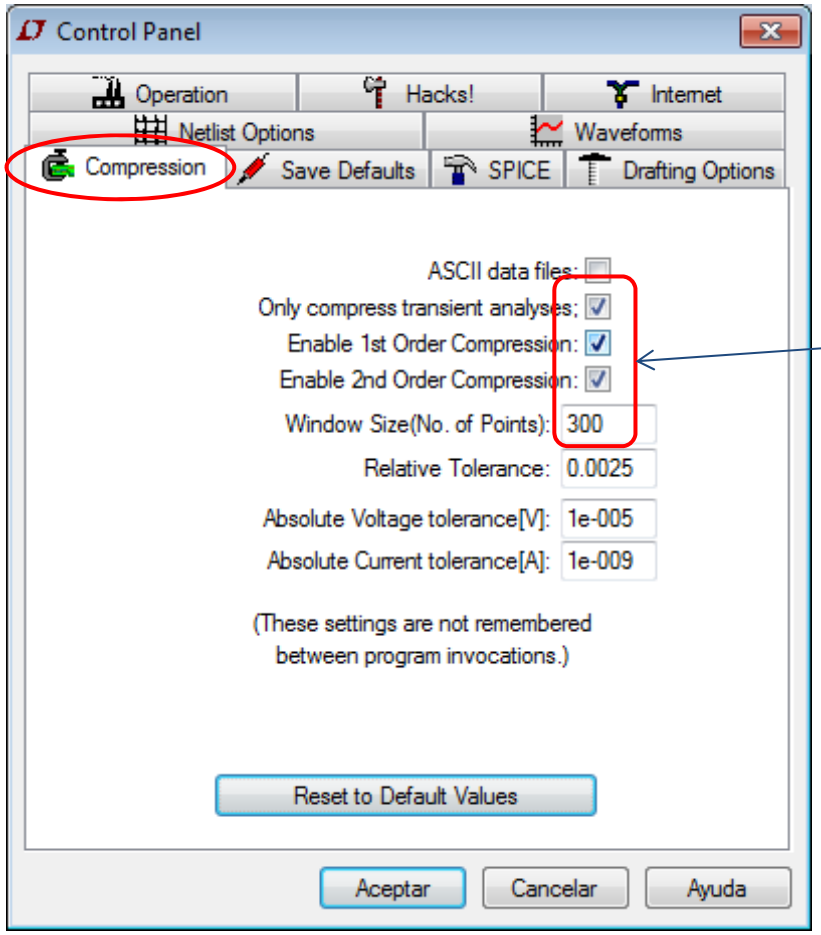
Insertar componentes

Directivas SPICE

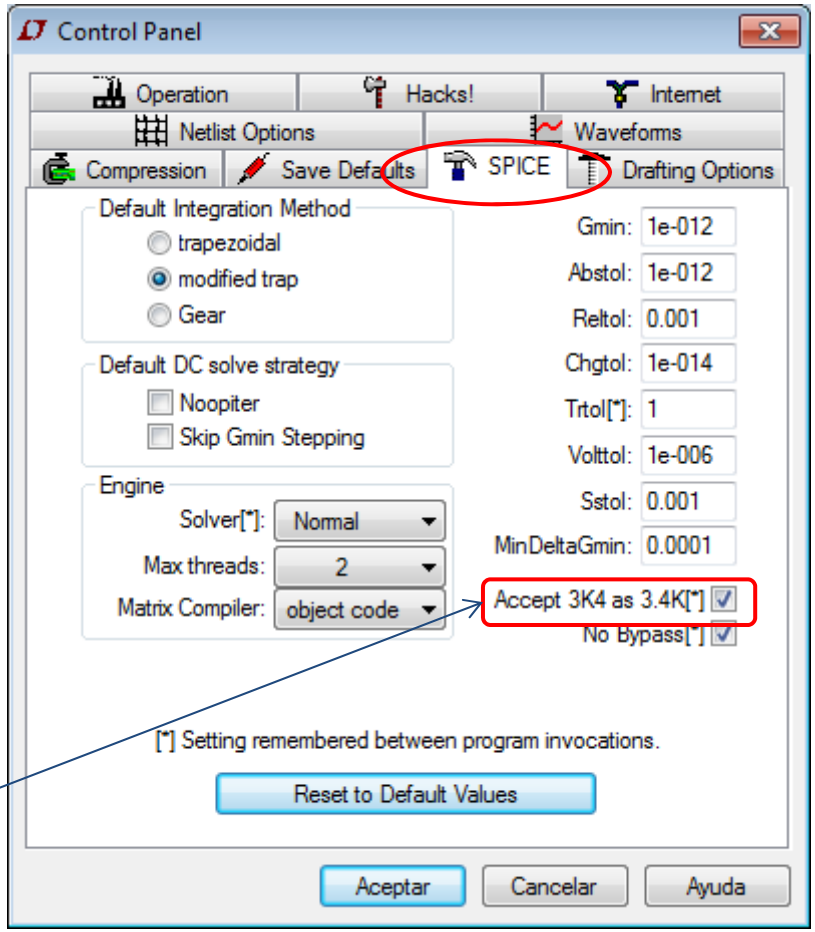
- Resistencias
- Condensadores
- Bobinas
- Diodos
- Etc.

Configuración

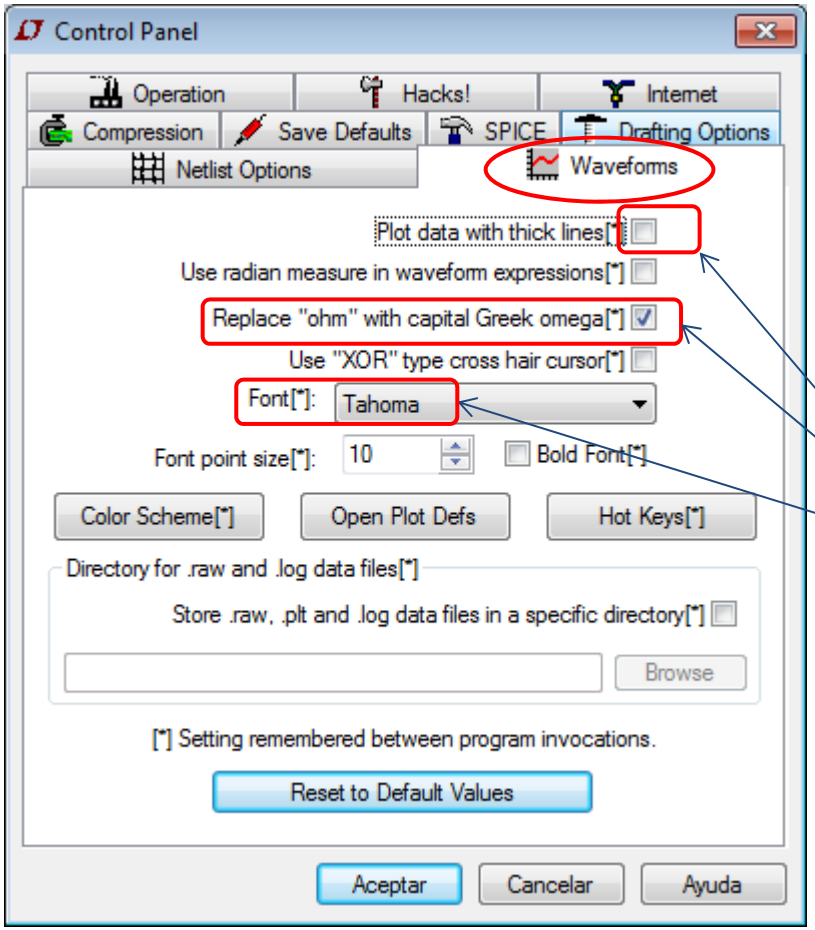
Control Panel



Eliminar la compresión de datos: Mejora los puntos por gráfica y la visualización de las mismas.

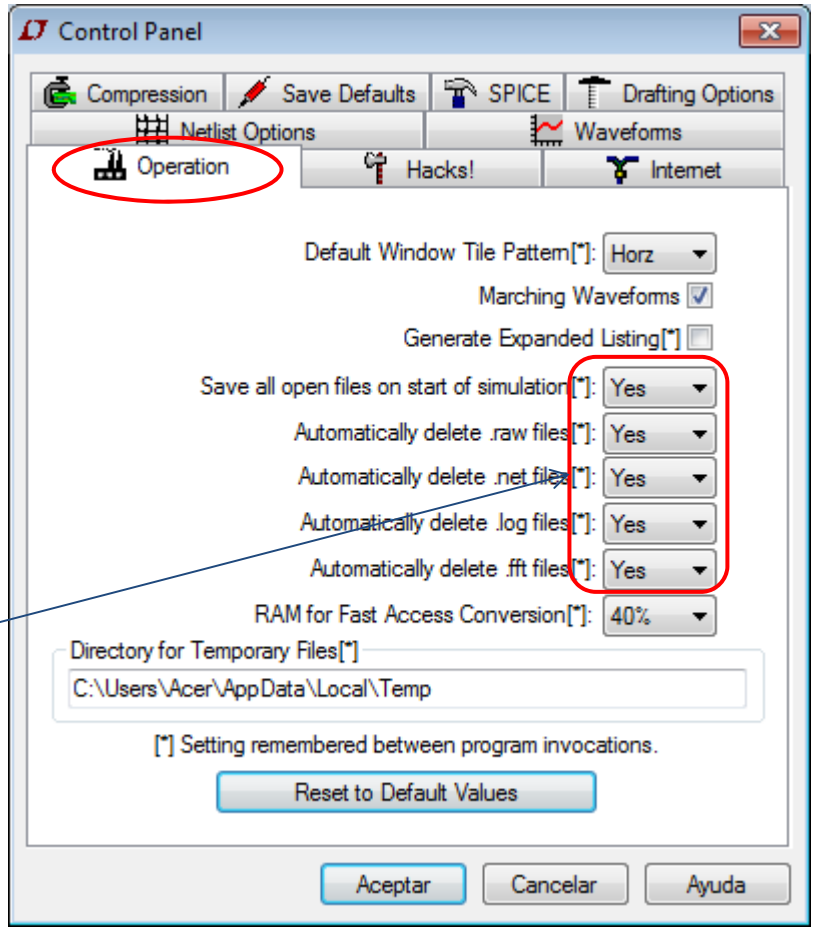


Permite escribir valores de forma más “natural”.

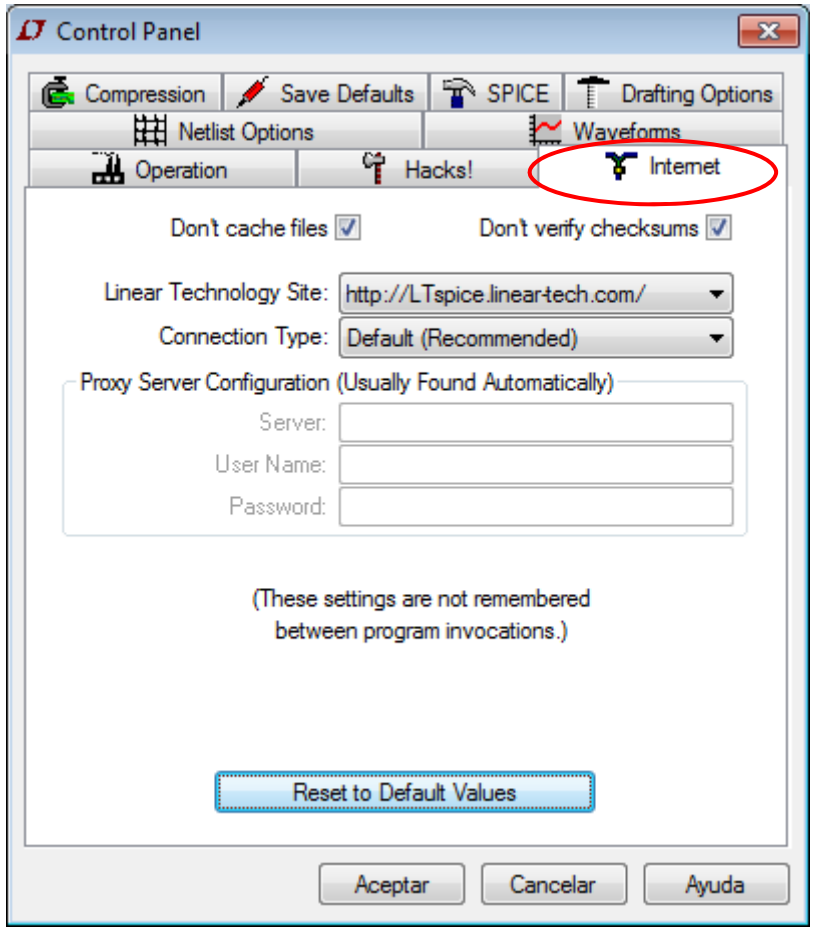


- Dibuja más gruesas las líneas de las gráficas de resultados.
- Permite utilizar la “Omega” para los valores de resistencia.
- Fuente de pantalla para los resultados.

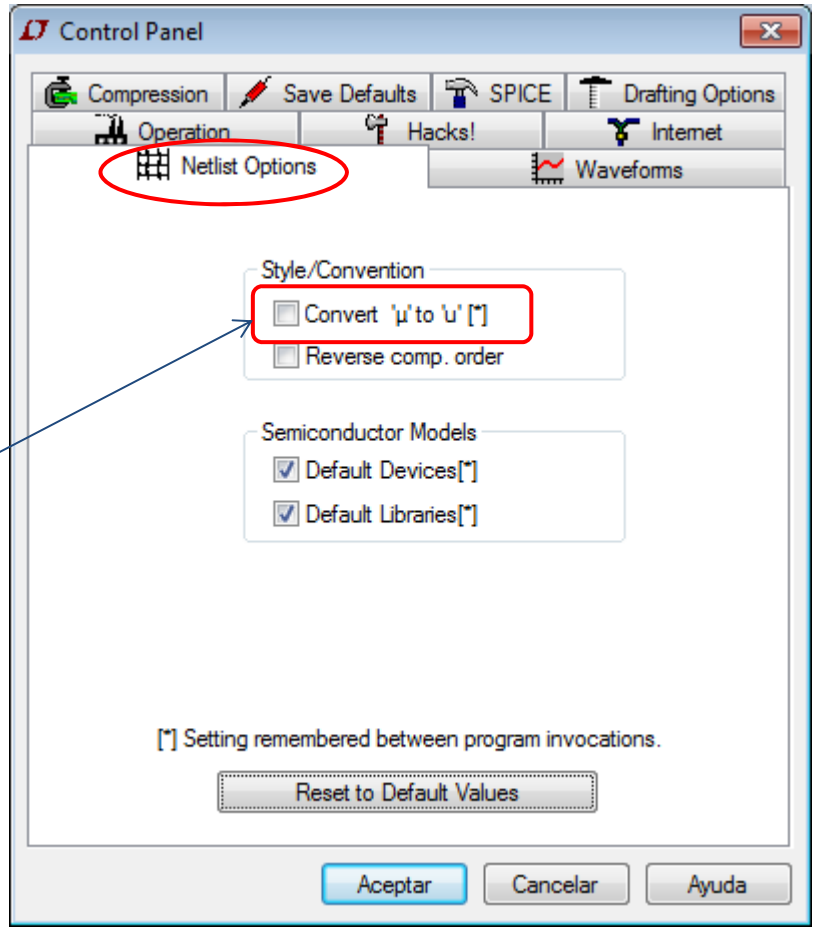
Elimina los ficheros temporales de análisis de los circuitos (Raw, Net, Log y Fourier).



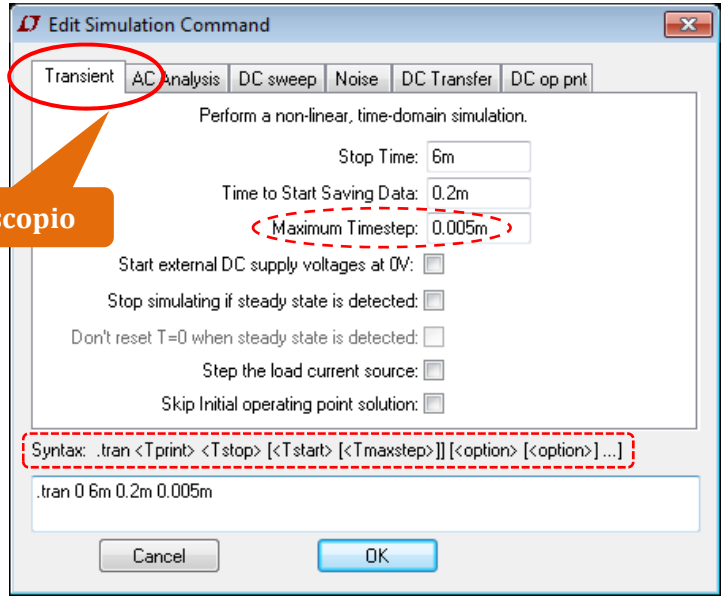
Control Panel



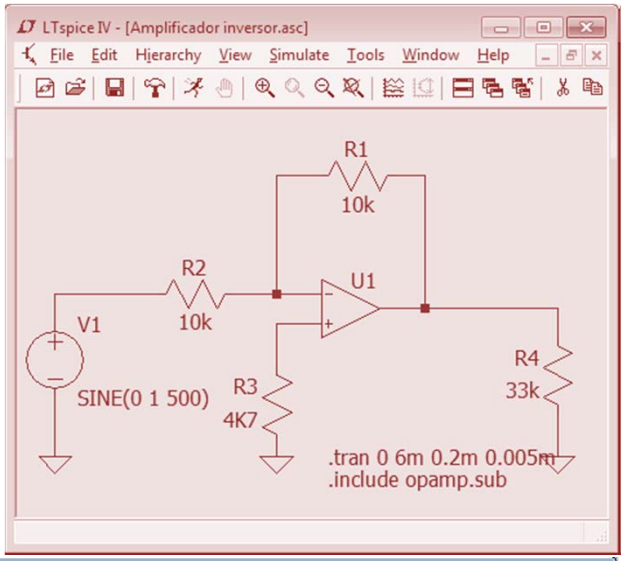
Mejora la escritura de valores de componentes

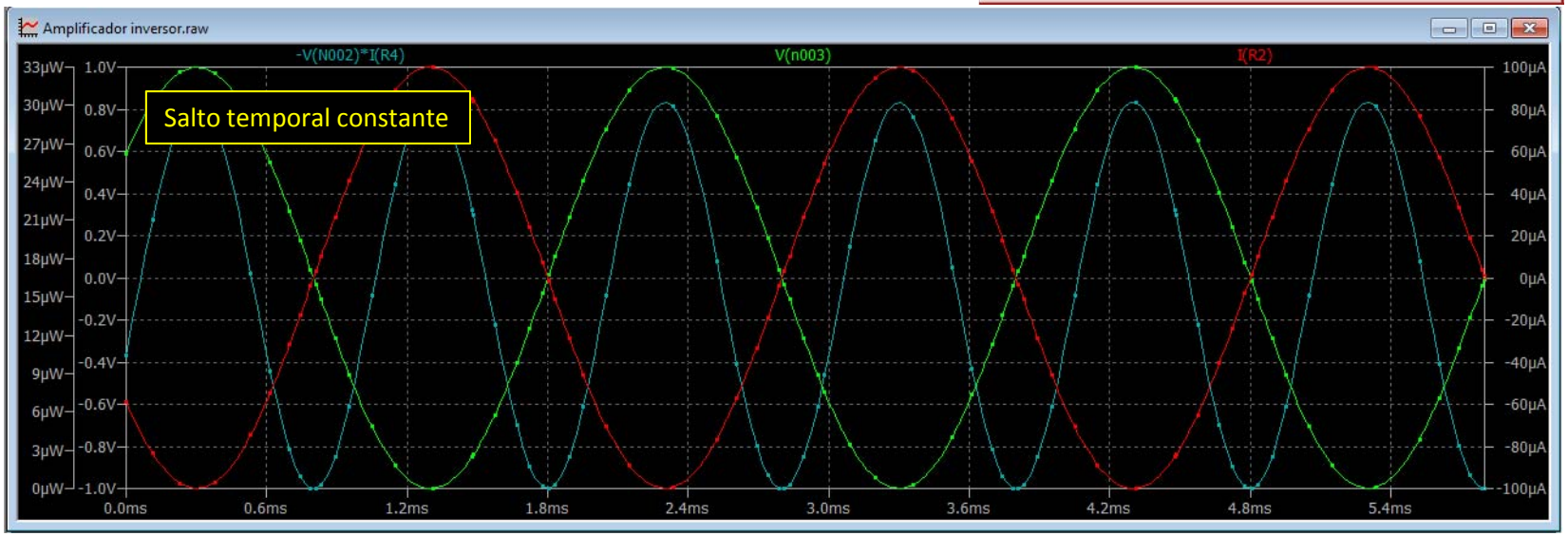


Tipos de simulación

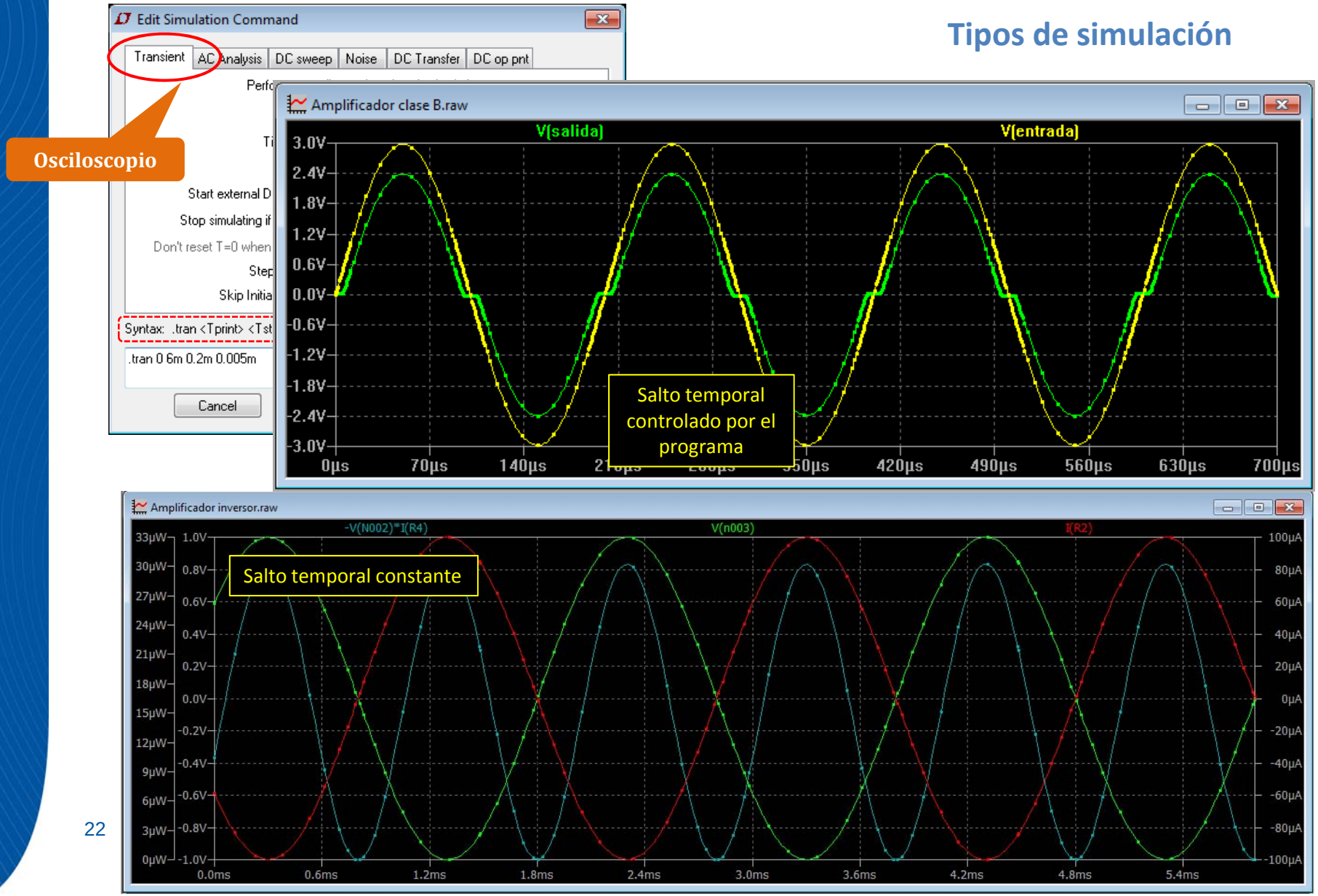


Osciloscopio

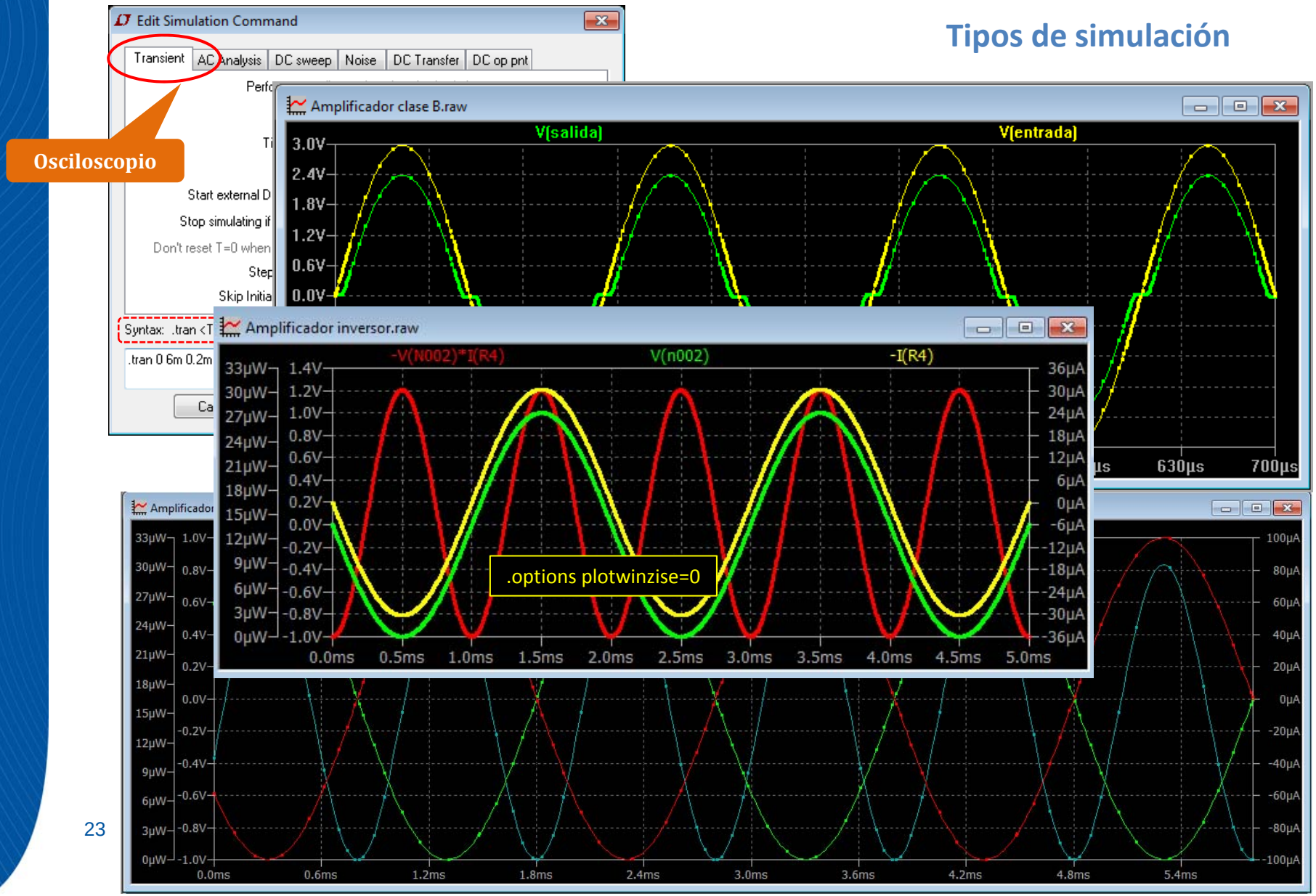




Tipos de simulación

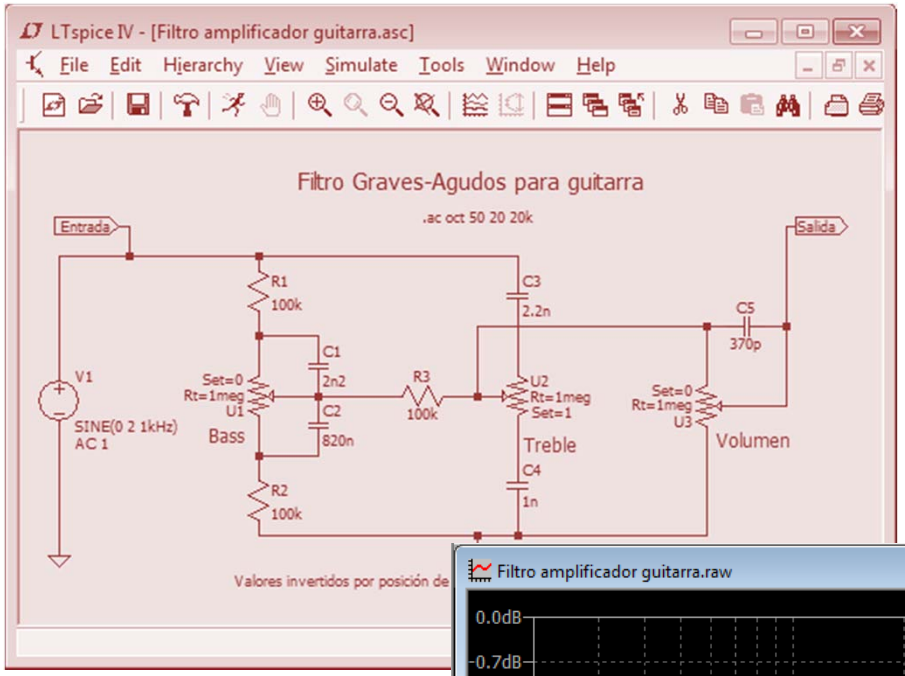


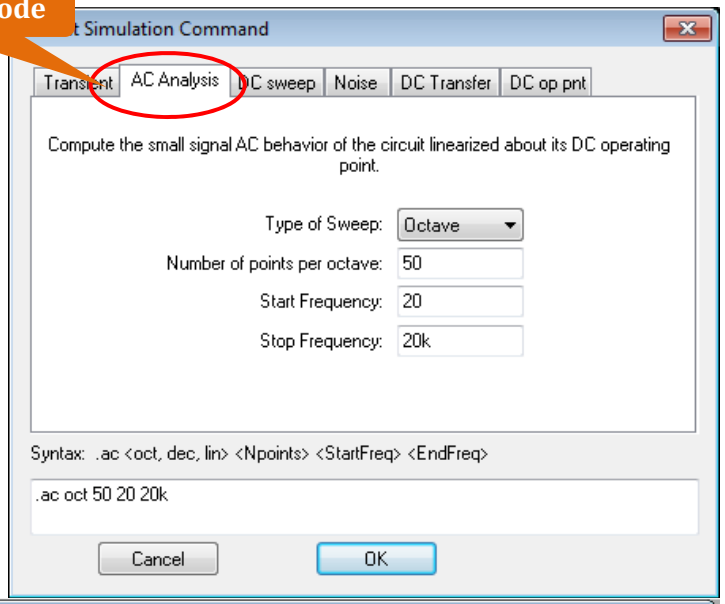
Tipos de simulación

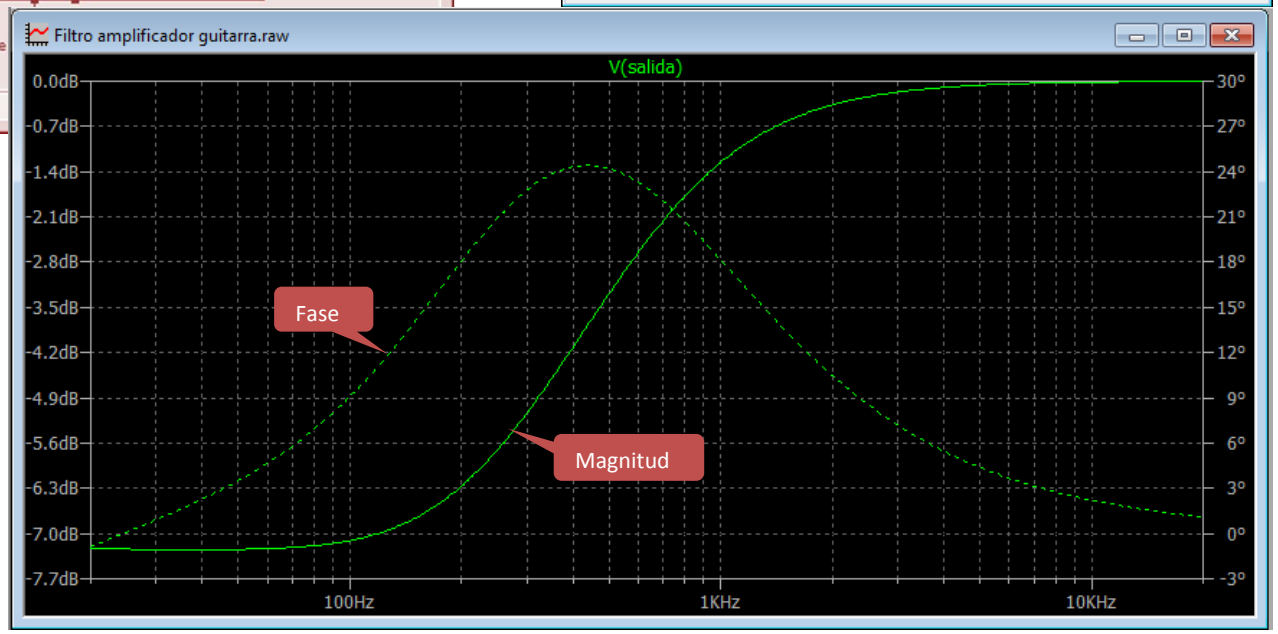


Ejecutar la simulación

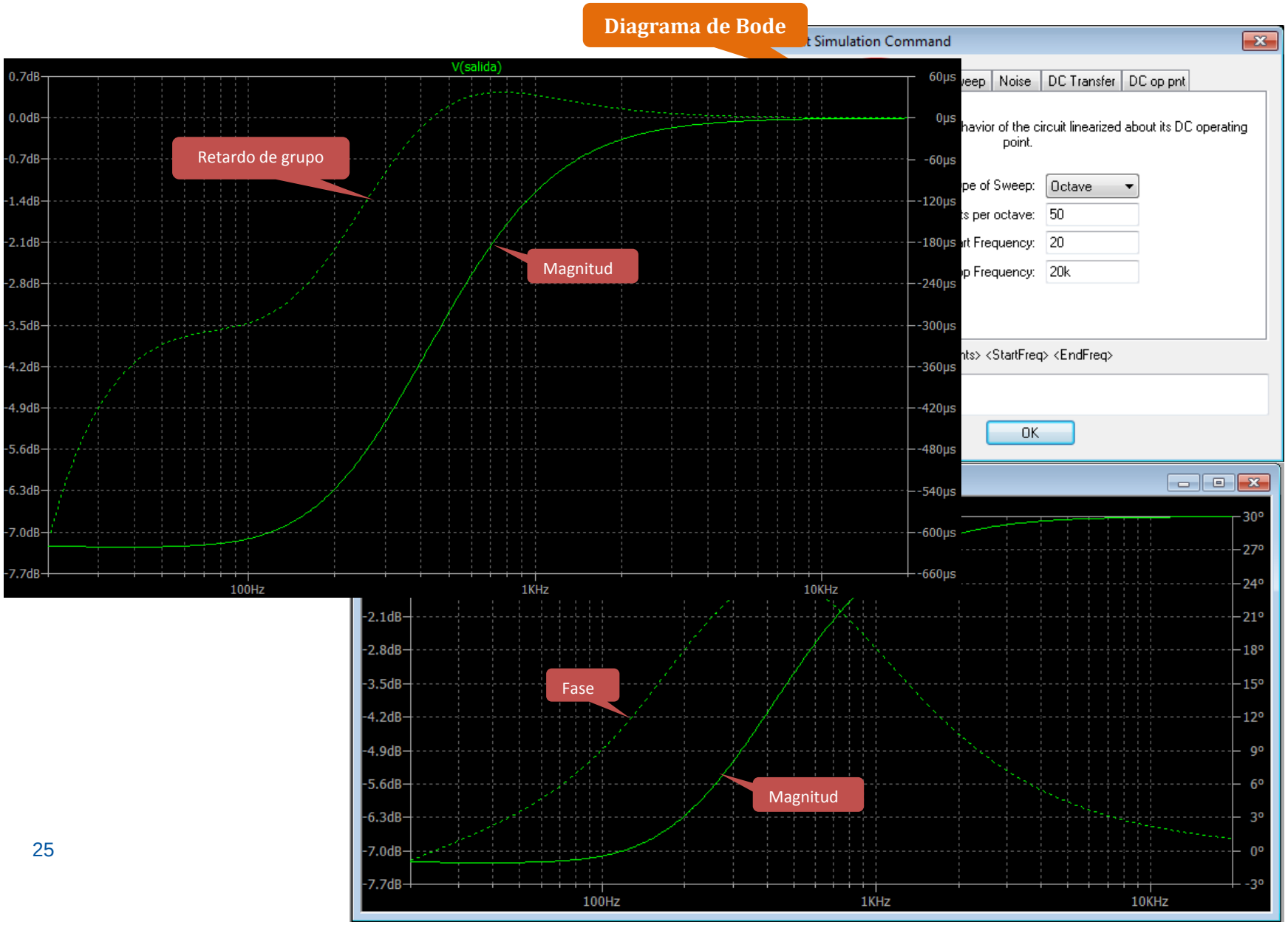
Diagrama de Bode



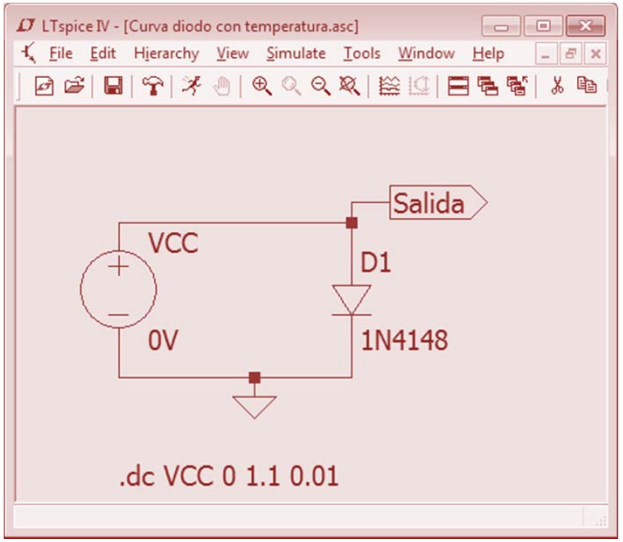




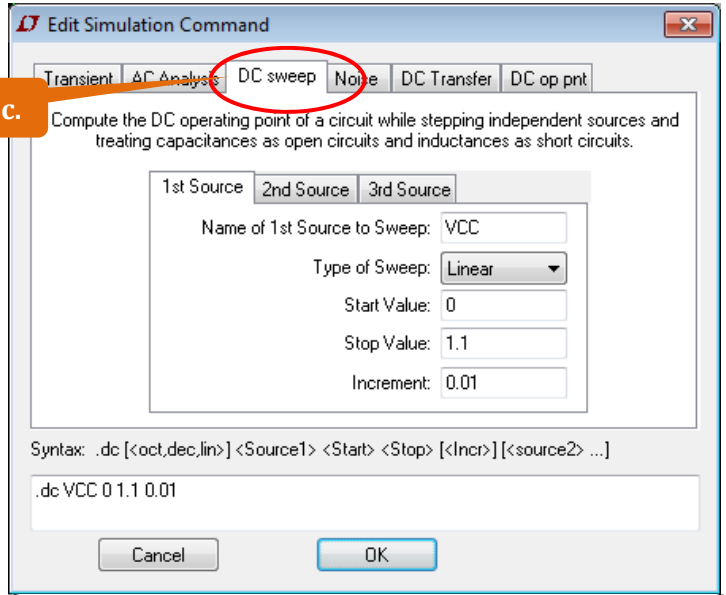
Ejecutar la simulación

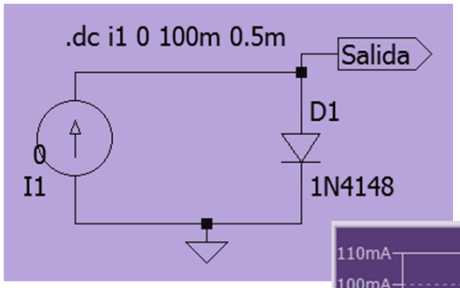


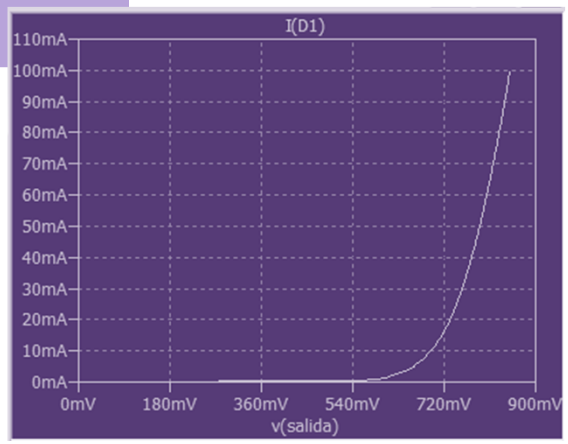
Ejecutar la simulación

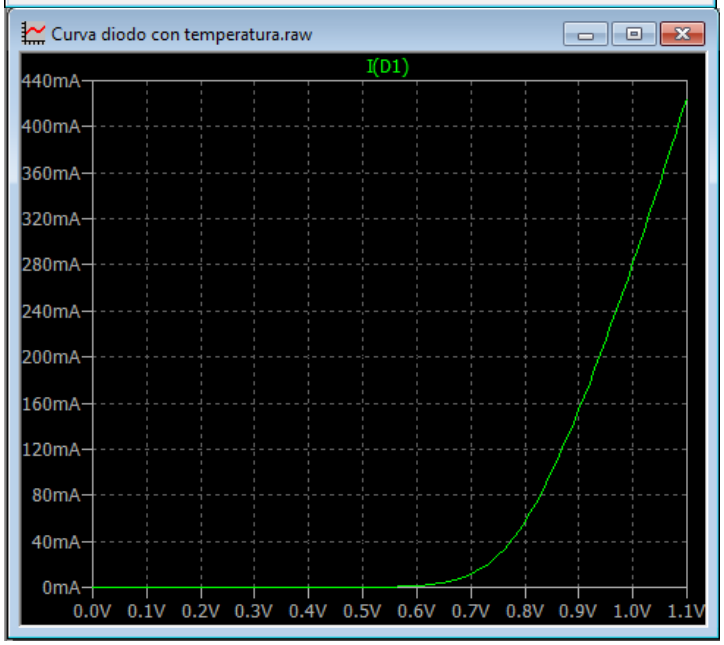


Barrido de c.c.

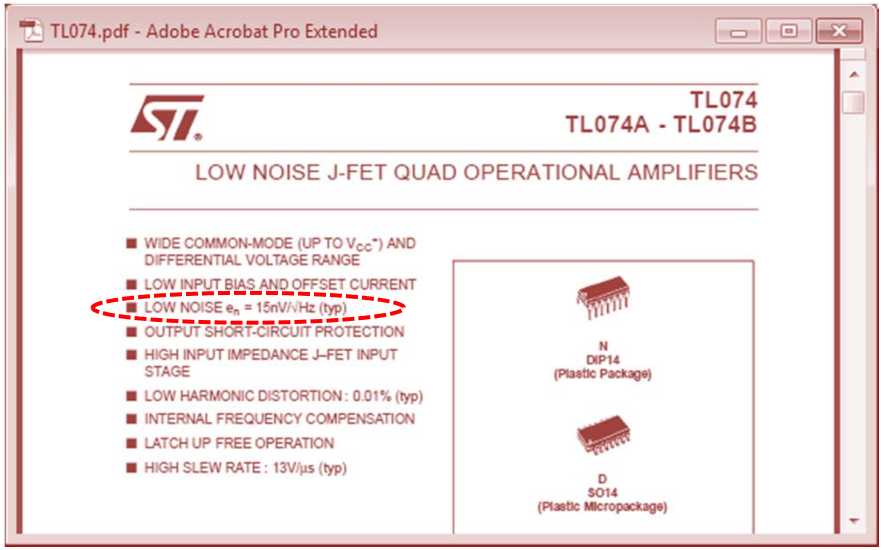


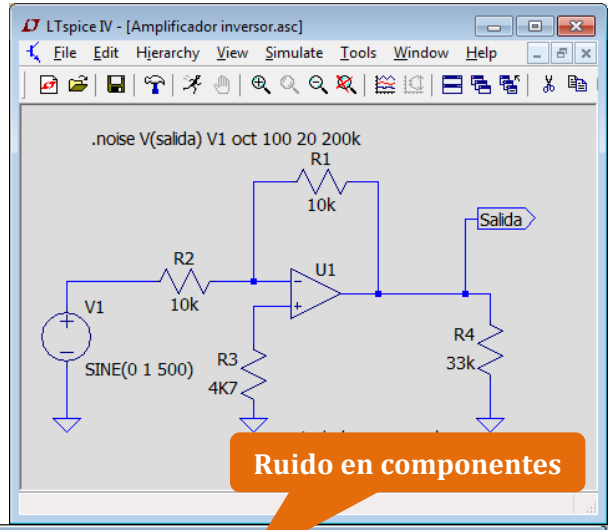




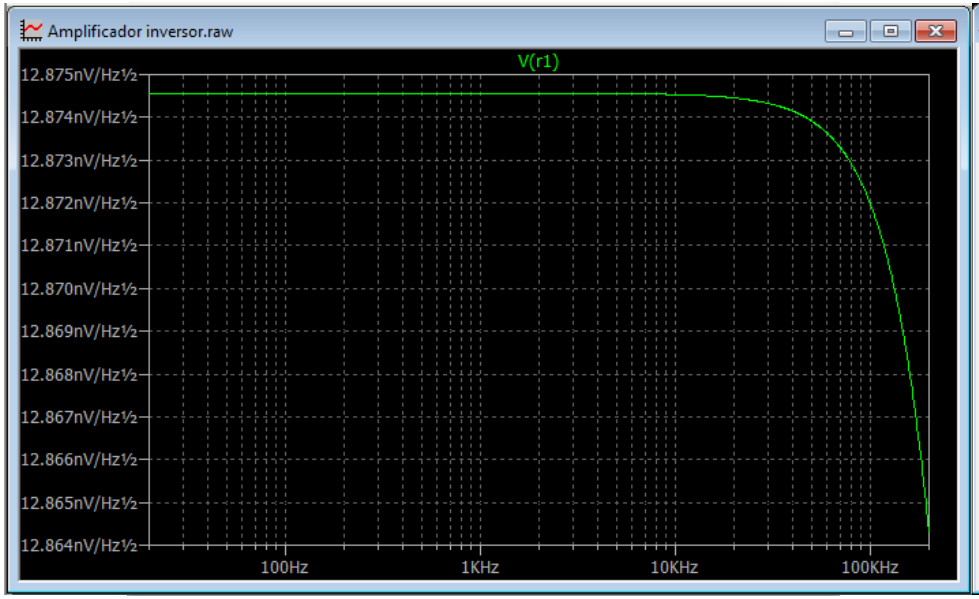


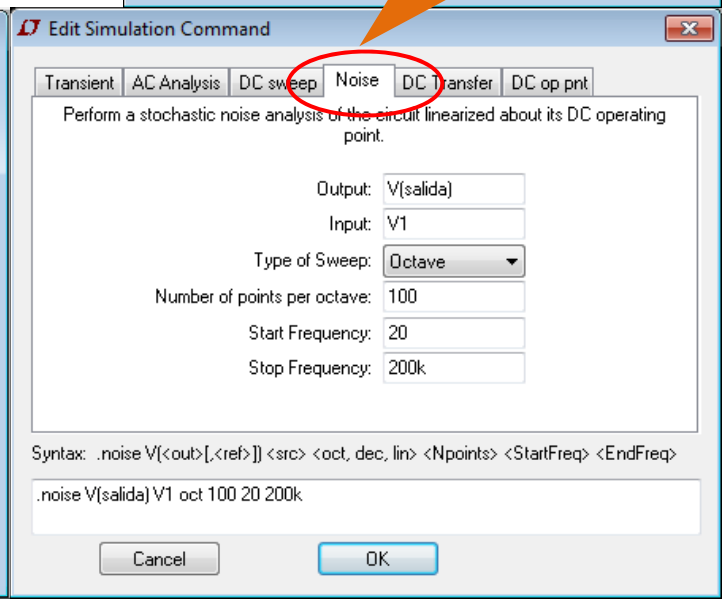
Ejecutar la simulación



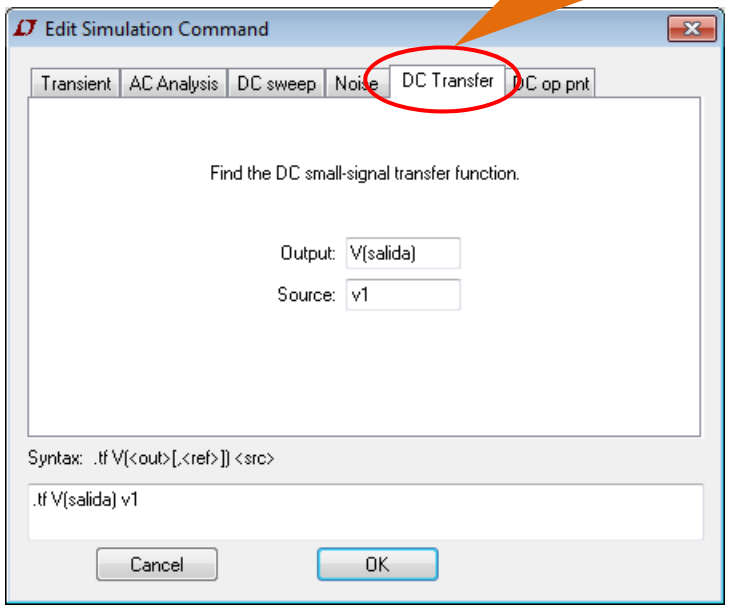


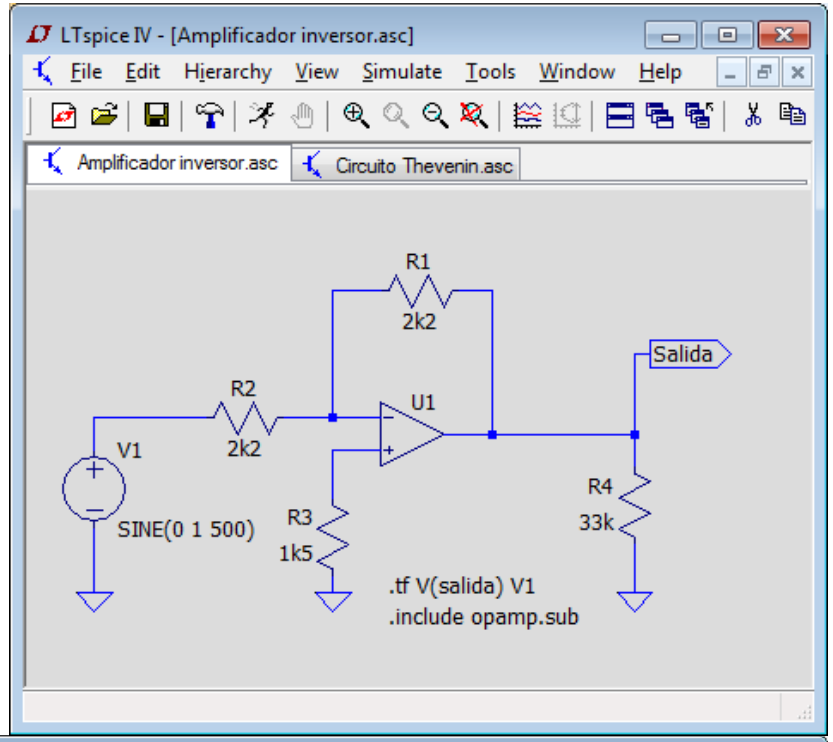
Ruido en componentes

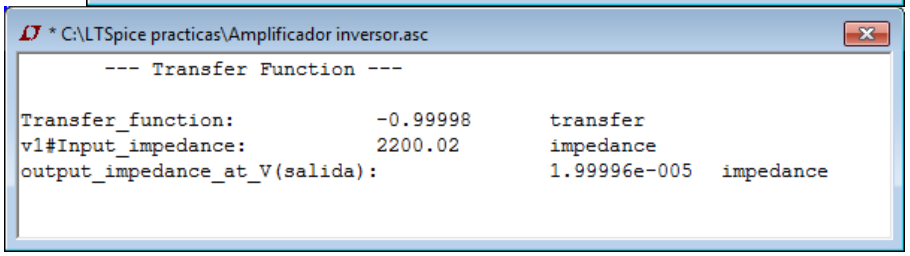




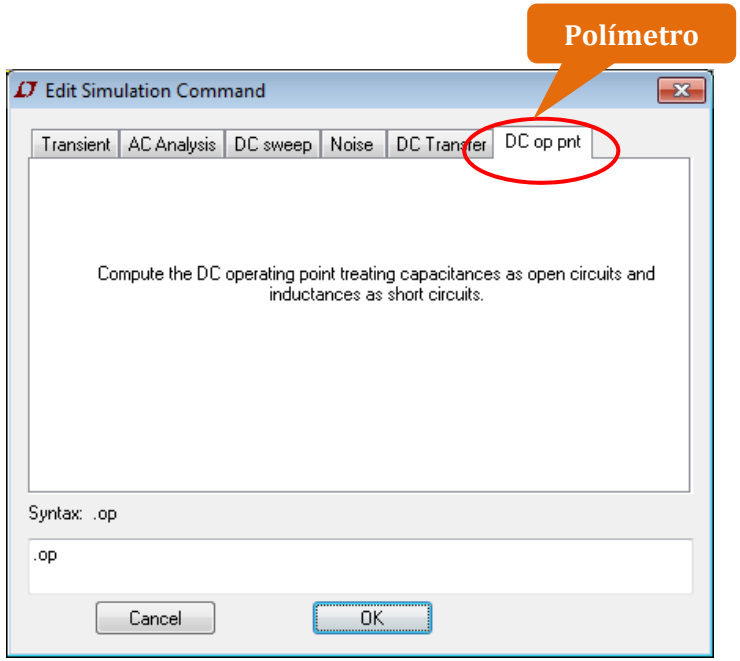
Función de transferencia

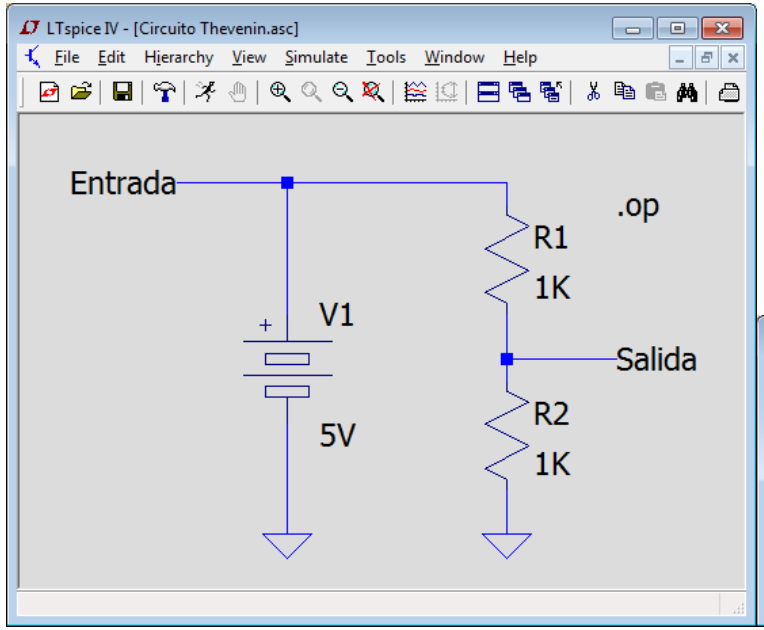


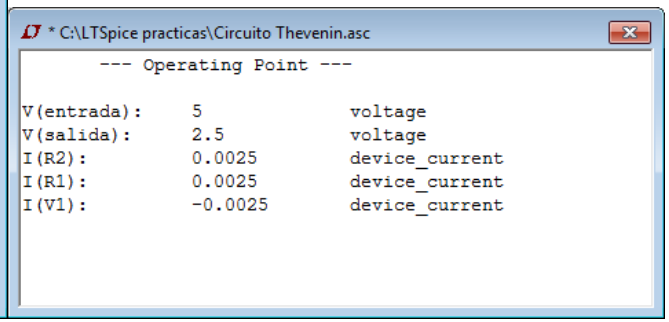




Ejecutar la simulación

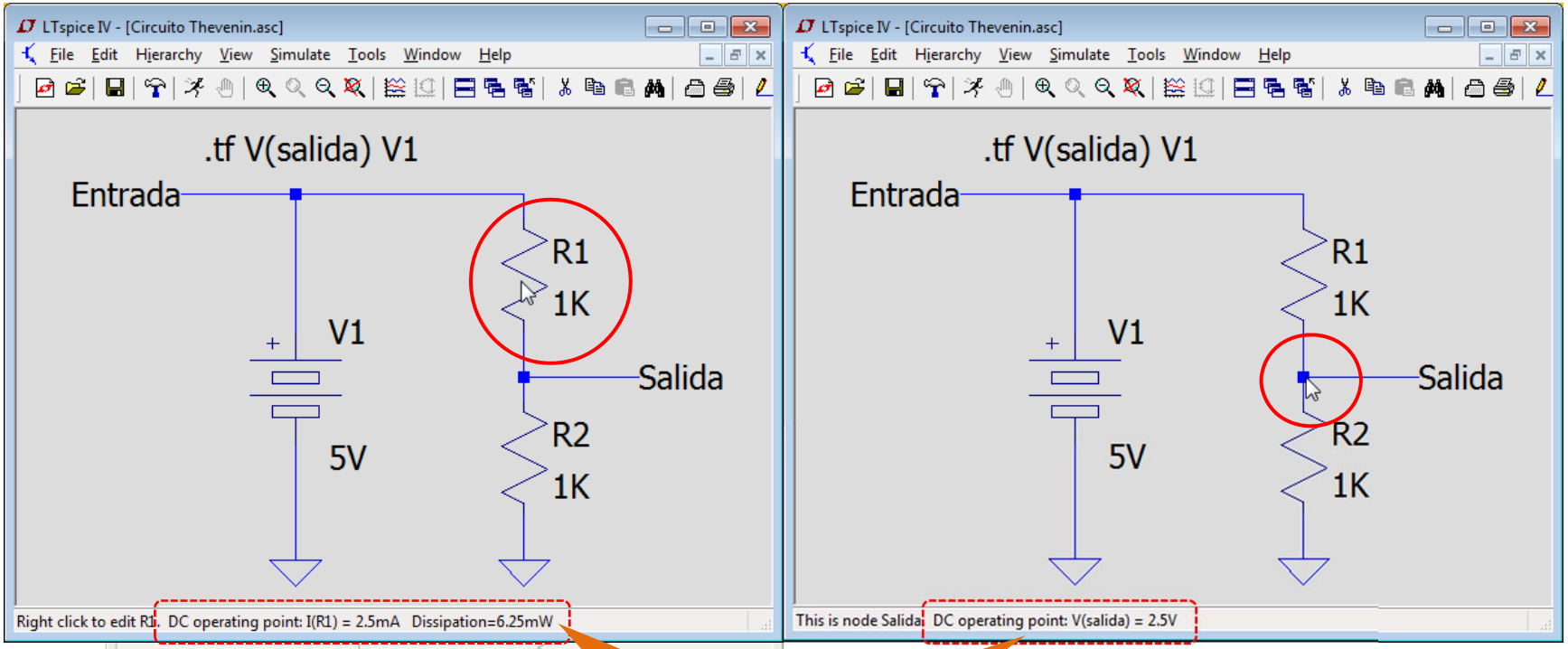






Marcadores V, I, P, ...

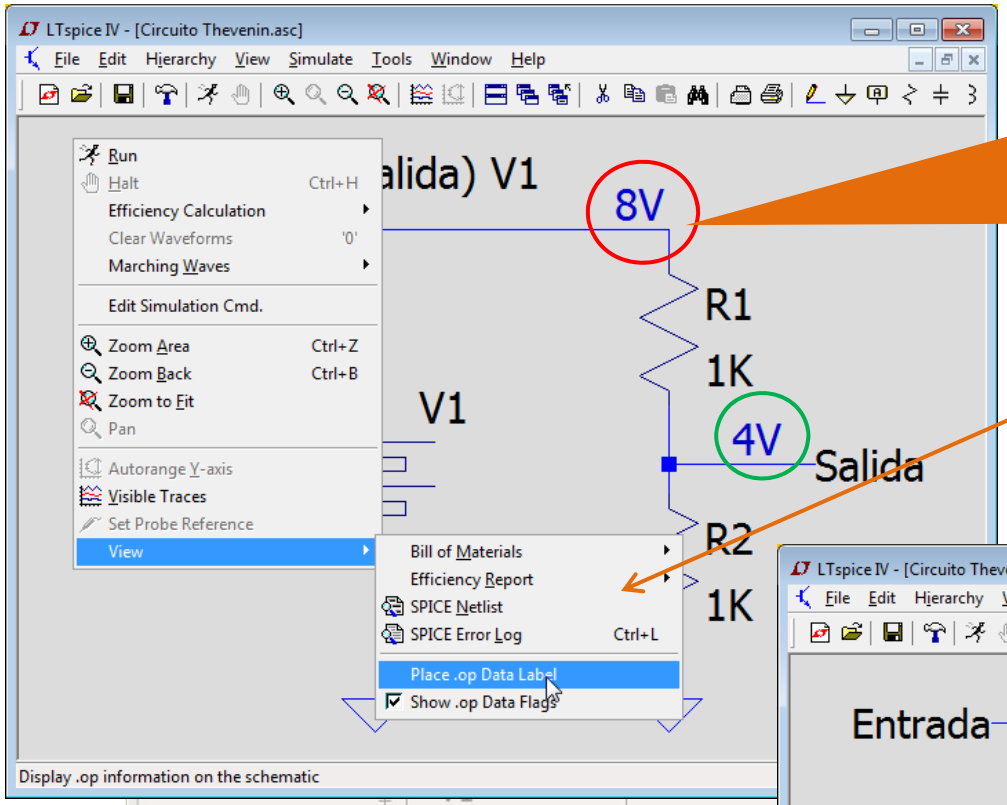
Ejecutar la simulación



- Moviendo el ratón encima de los componentes, se indican la corriente y la potencia en los mismos.
- Si se mueve sobre los conductores o los nudos del circuito, se muestran las tensiones.

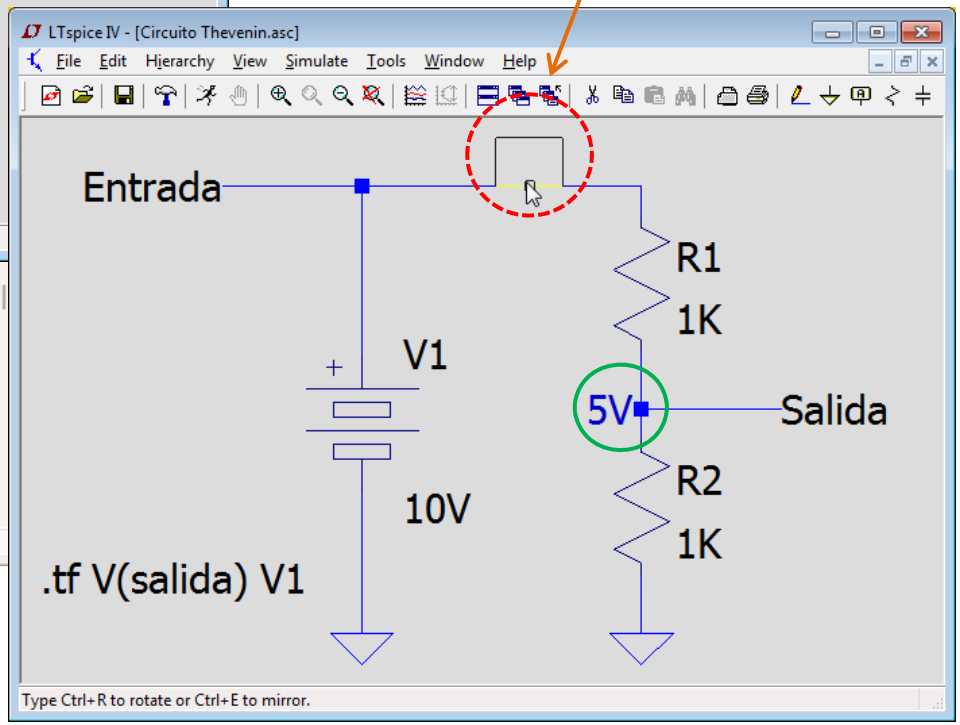
Sólo disponible para simulación .tf o .op

Ejecutar la simulación

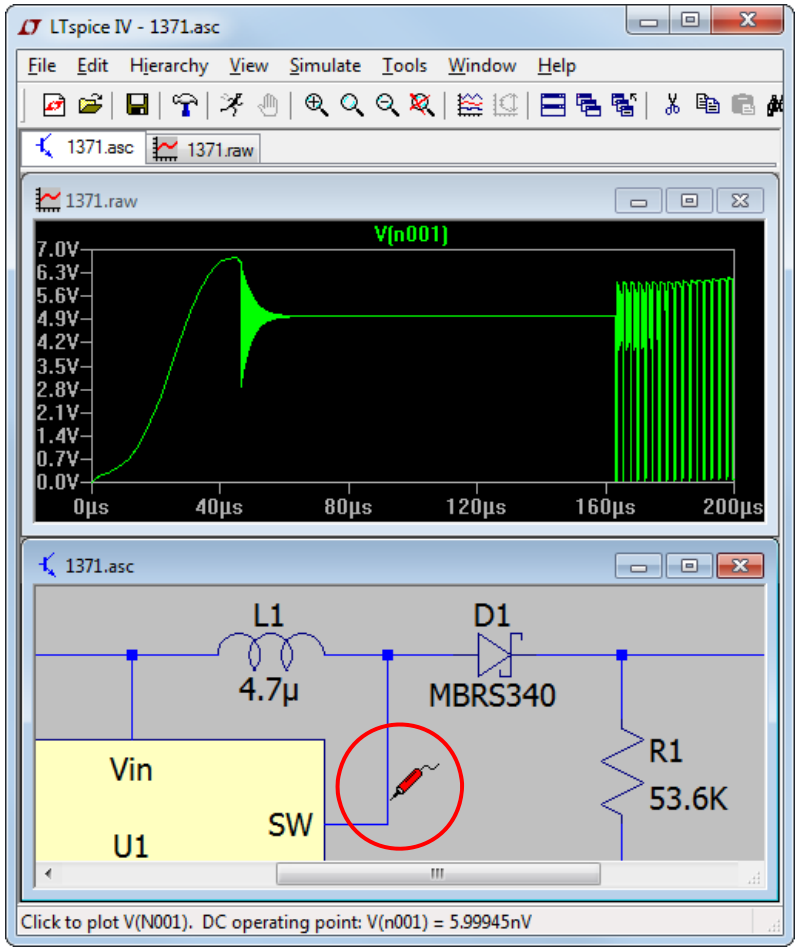


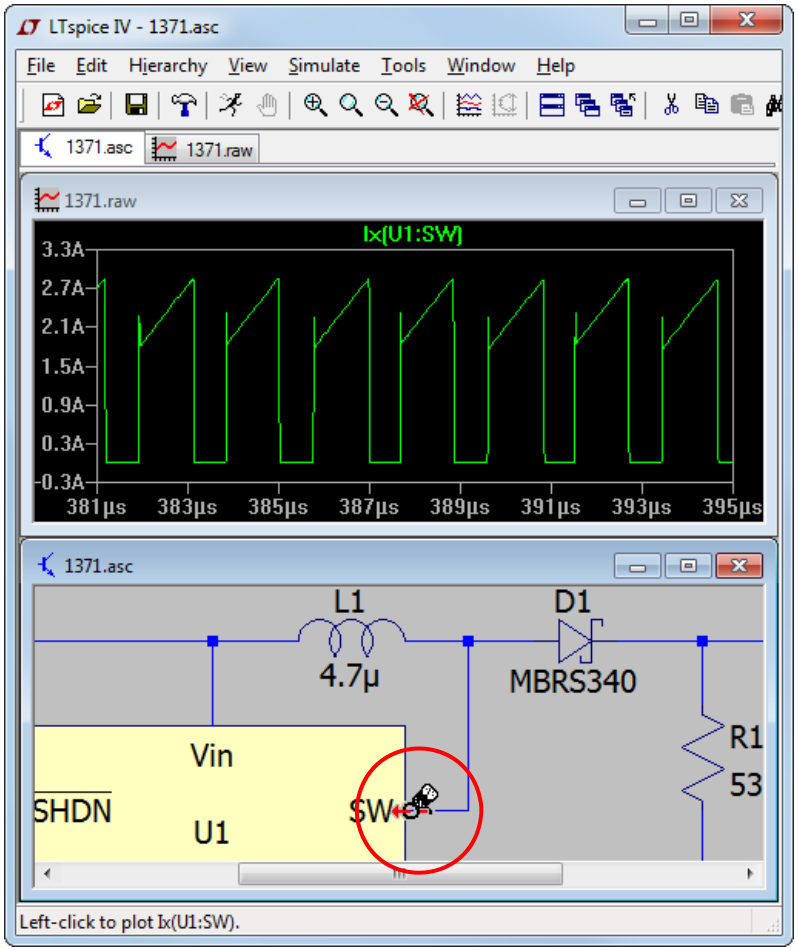
- Se puede mostrar permanentemente la tensión en los conductores/nudos del circuito. Aparece un cuadrado vacío donde se desea colocar la tensión. (Vía menú contextual).
- **Más fácil**: Si se pincha encima del conductor, también aparece este dato.

El valor se actualiza en cada nueva simulación.



Sólo disponible para simulación .tf o .op



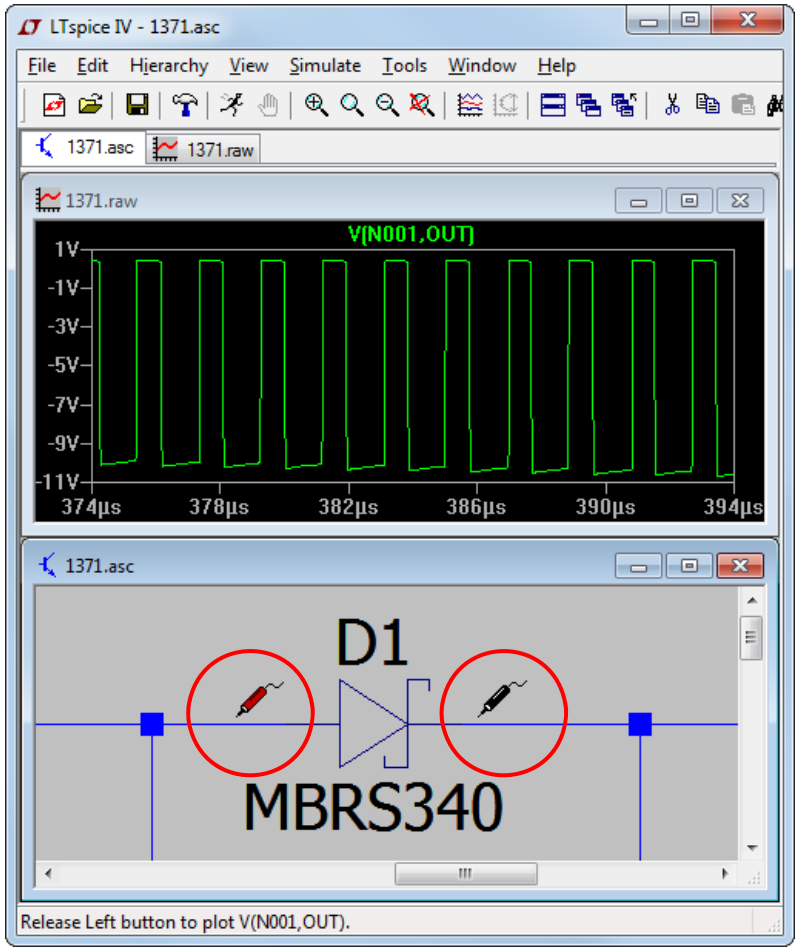


Tensión referida a masa

Haciendo left click sobre el cable deseado.

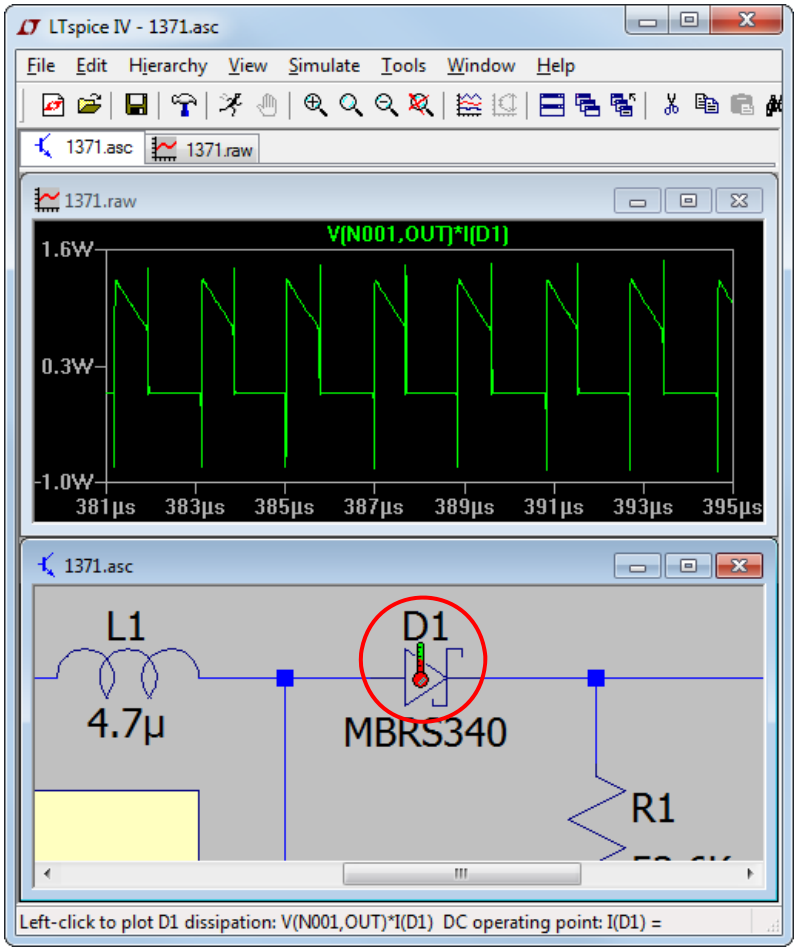
Corriente que atraviesa un componente

Left click sobre el componente. Situando el cursor encima del componente se indica el sentido convencional de la corriente.



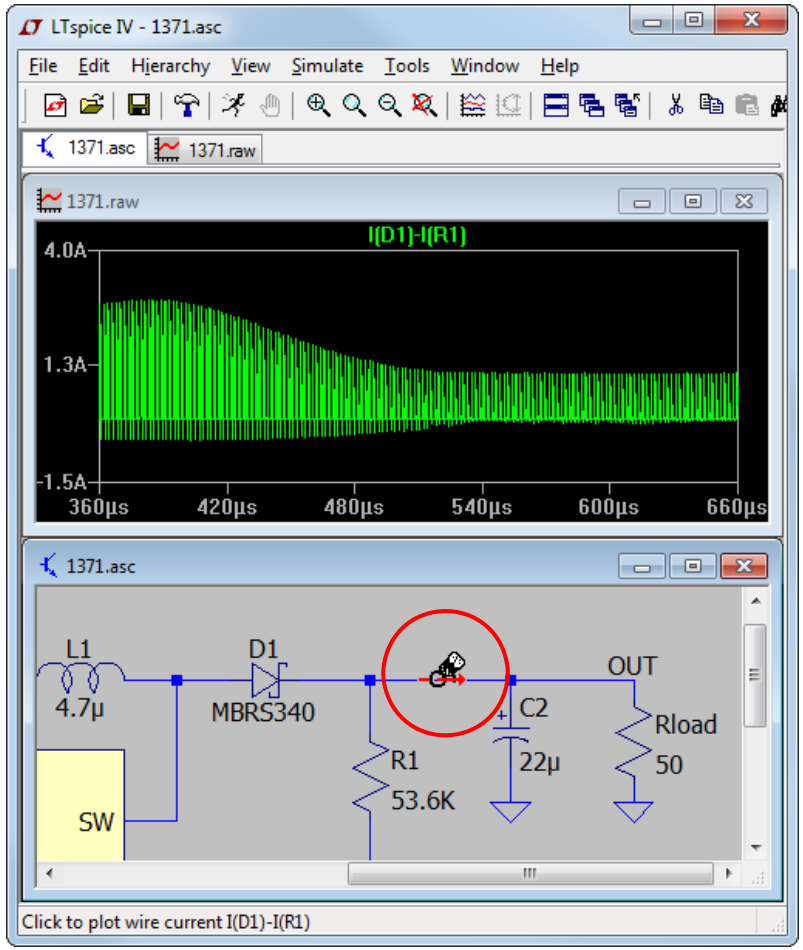
Tensión diferencial

Haciendo left click sobre el cable deseado y, manteniendo pulsado el botón, arrastrarlo al segundo punto.



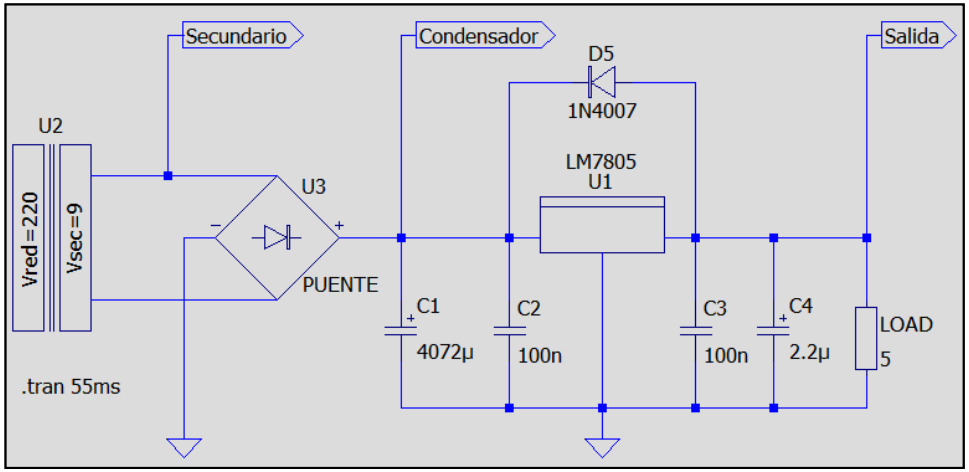
Potencia instantánea disipada

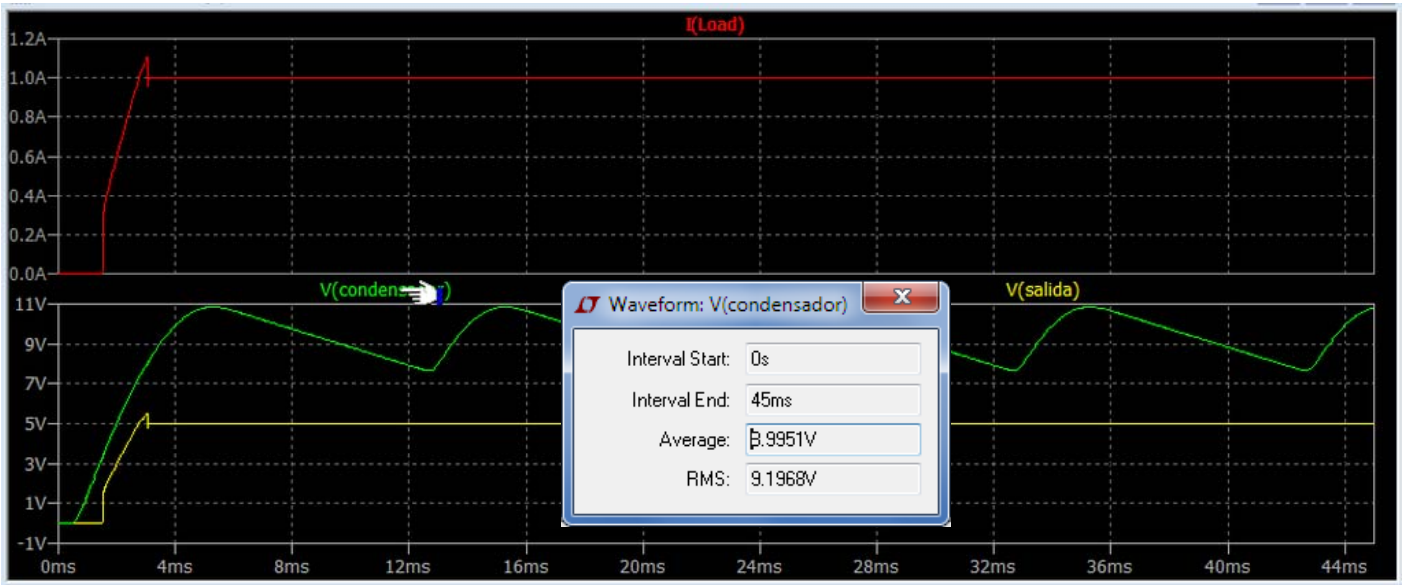
Haciendo Alt+click sobre el componente deseado.



Corriente por un conductor

Pulsando Alt+click sobre el conductor deseado.



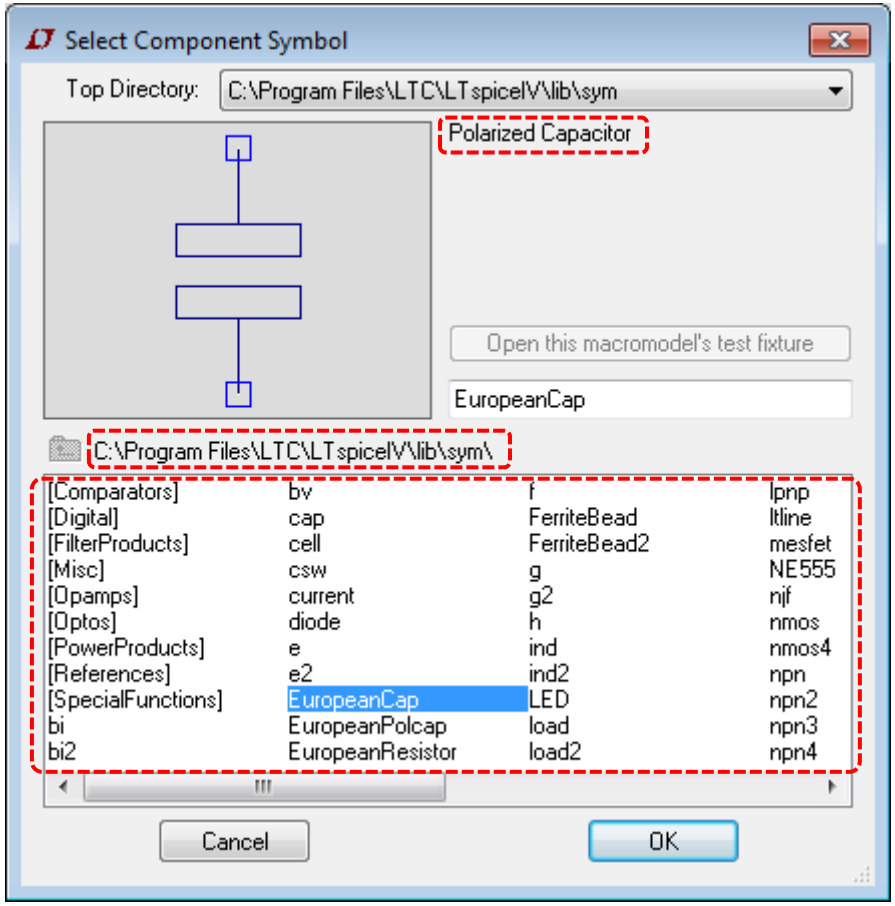


Valores medio (Average) y eficaz (RMS)

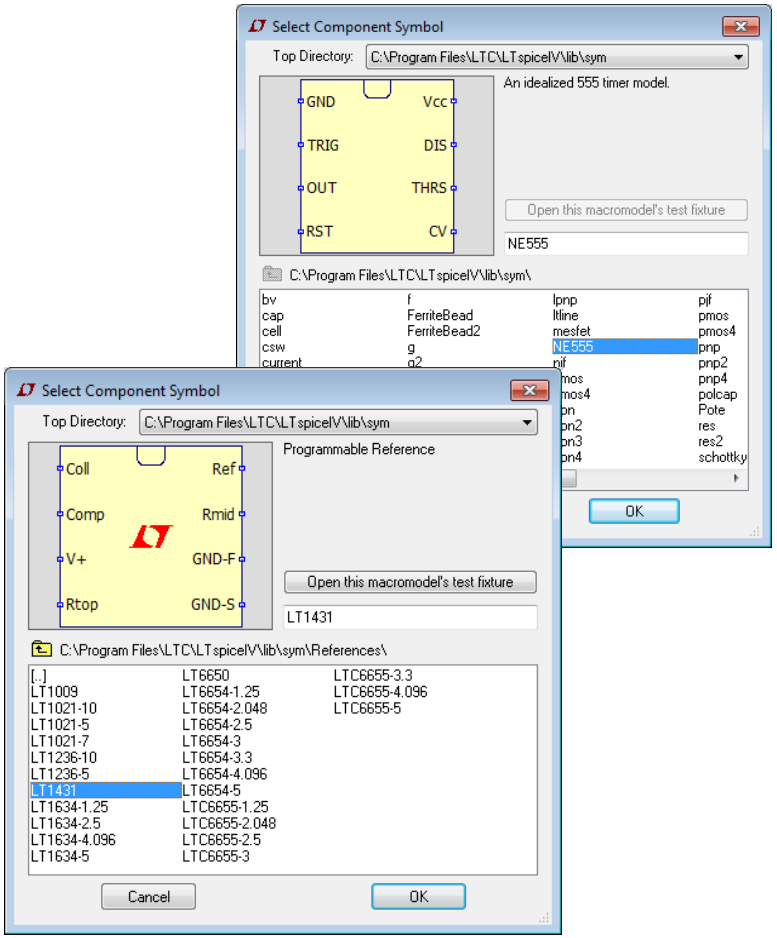
Pulsando Ctrl+click sobre la etiqueta deseada "V(condensador)".

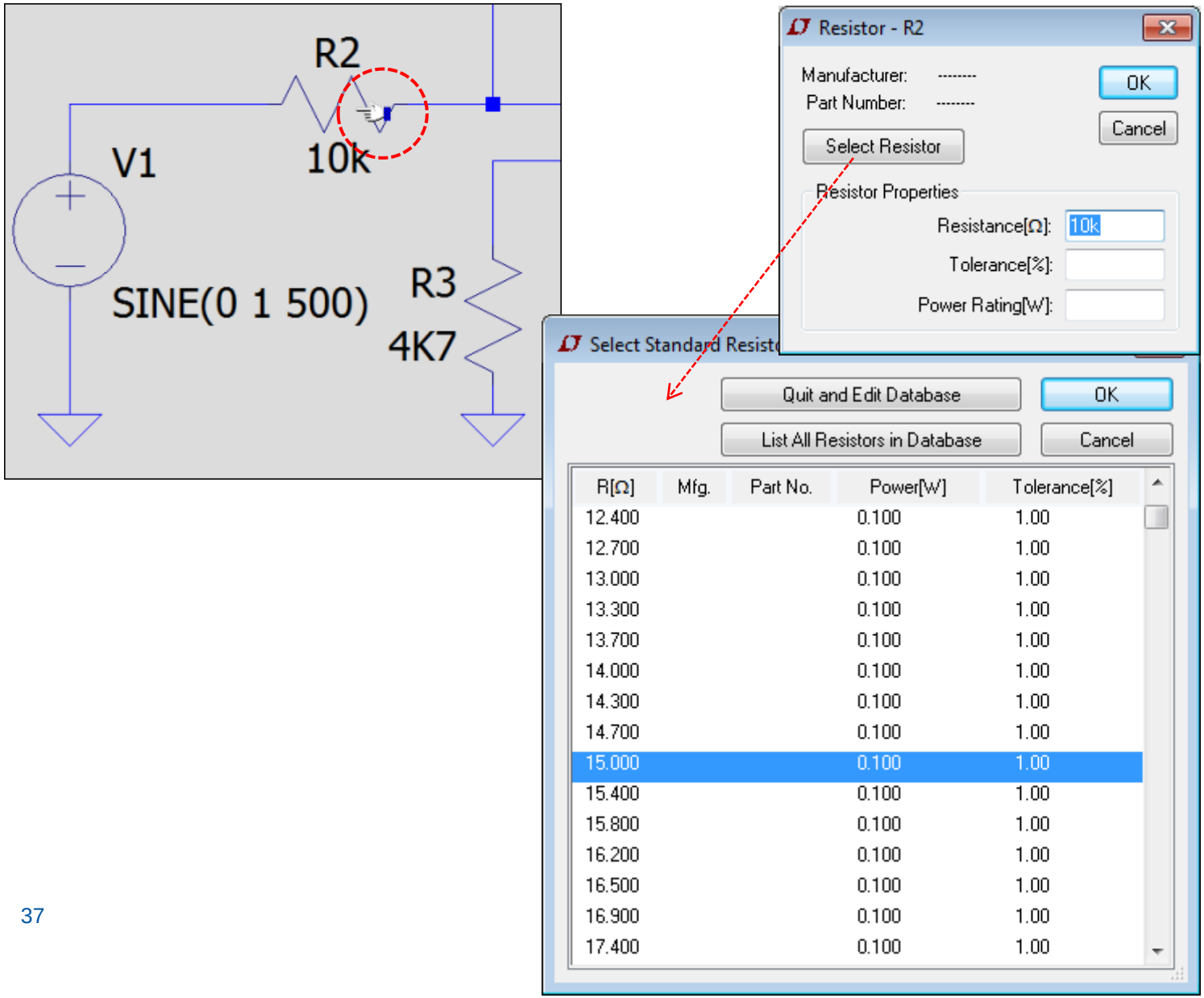
Selección de componentes

Insertar componentes

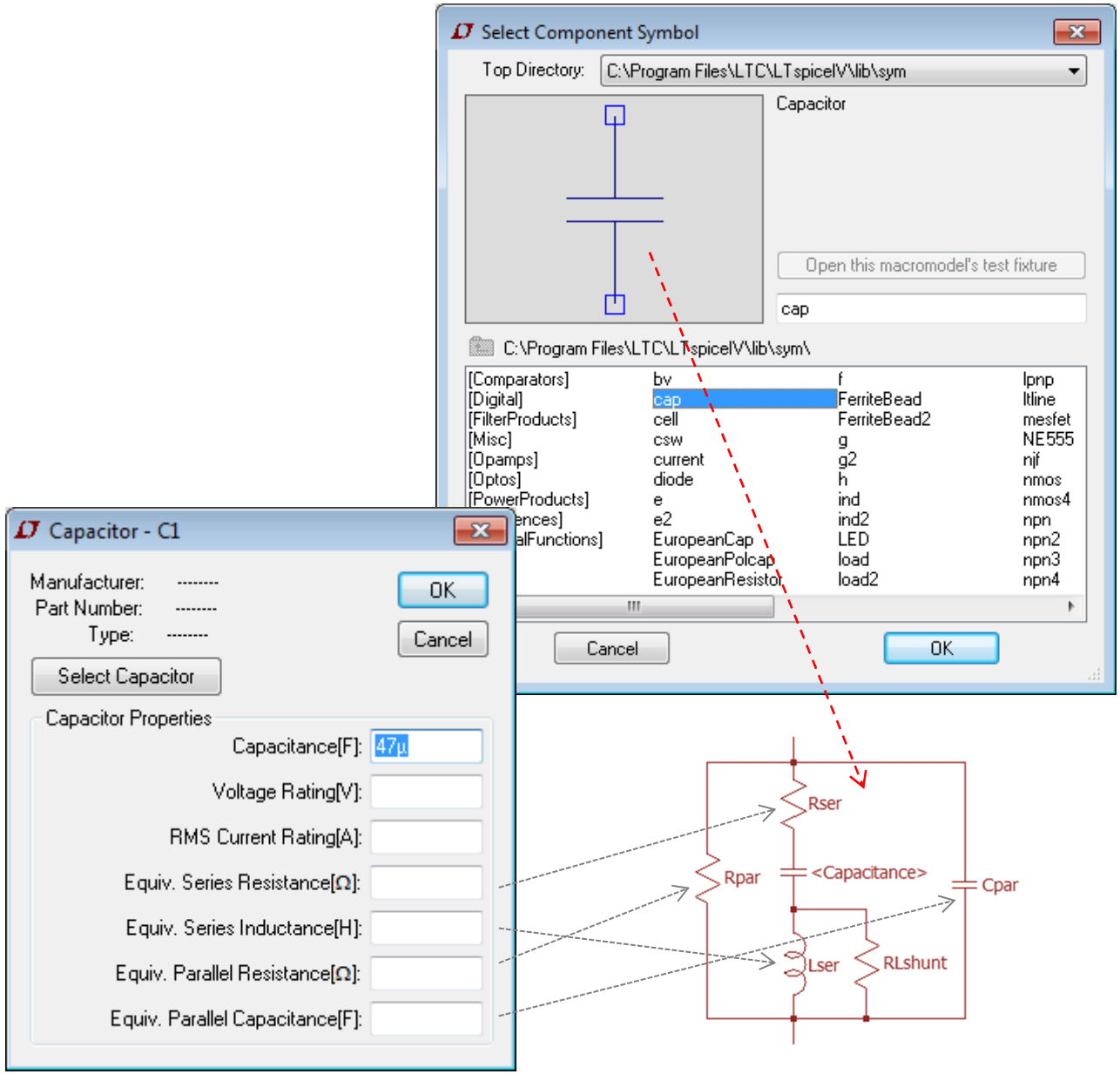


- Muestra todos los componentes
- Dibujo del componente
- Subdirectorio de trabajo
- Funcionalidad eléctrica/electrónica

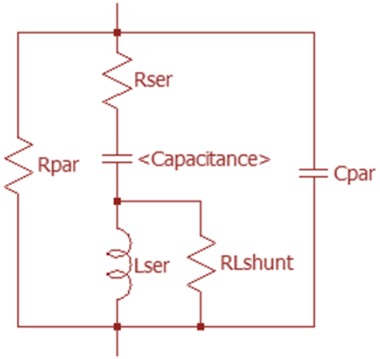



R2
10k
V1
SINE(0 1 500)
R3
4K7
Resistor - R2
Manufacturer: --------
Part Number: --------
OK
Cancel
Select Resistor
Resistor Properties
Resistance[Ω]: 10k
Tolerance[%]:
Power Rating[W]:
Select Standard Resist
Quit and Edit Database
OK
List All Resistors in Database
Cancel
R[Ω] Mfg. Part No. Power[W] Tolerance[%]
12.400 0.100 1.00
12.700 0.100 1.00
13.000 0.100 1.00
13.300 0.100 1.00
13.700 0.100 1.00
14.000 0.100 1.00
14.300 0.100 1.00
14.700 0.100 1.00
15.000 0.100 1.00
15.400 0.100 1.00
15.800 0.100 1.00
16.200 0.100 1.00
16.500 0.100 1.00
16.900 0.100 1.00
17.400 0.100 1.00

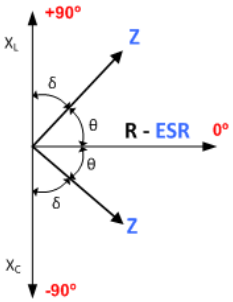
Insertar componentes



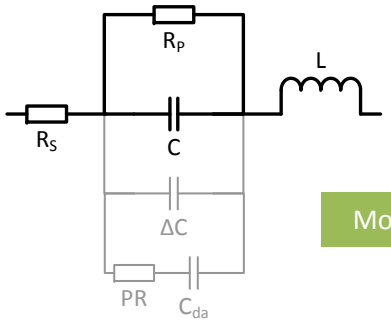
Insertar componentes (C)



Modelo LTSpice IV



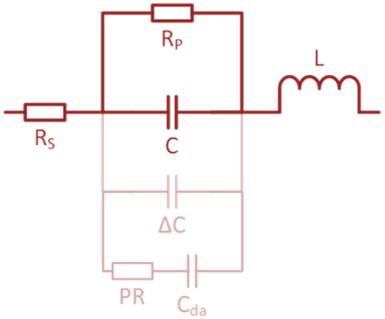
Precision LCR Meter: Agilent 4284A



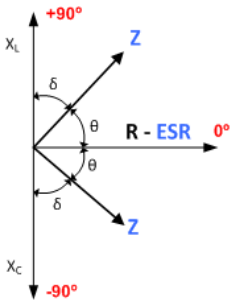
Modelo Agilent

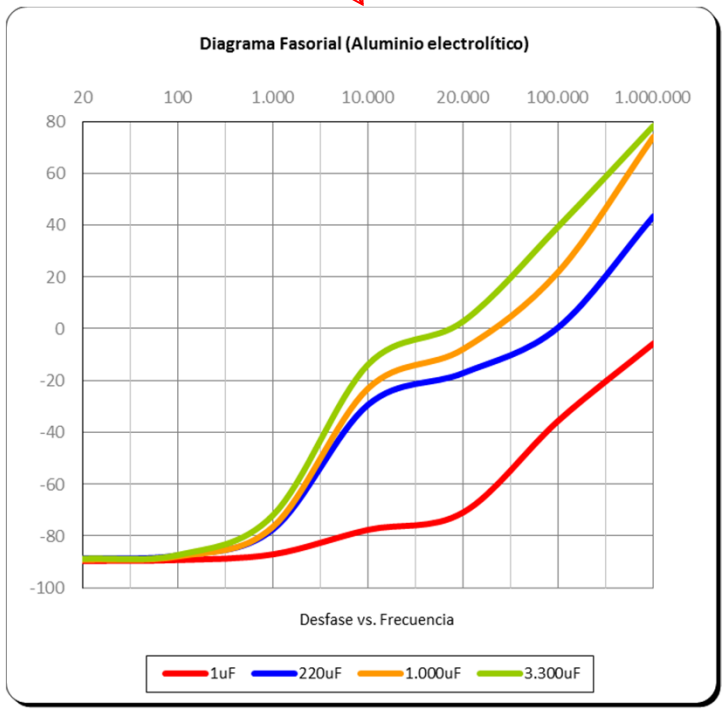
Rs/ESR: Resistencia equivalente serie (Debida a la longitud de las placas del condensador, patillas de conexión, etc.).
Rp/IR: Resistencia paralelo/Resistencia del aislamiento/Pérdidas-fugas internas del condensador (Debidas a la no perfección del dieléctrico entre placas).
C: Capacidad nominal del condensador.
L: Inductancia (Debida al arrollamiento de las placas del condensador). Más importante en condensadores de Aluminio, despreciable en condensadores de Tántalo.
ΔC: Variación de la capacidad nominal del condensador (Debida a efectos de temperatura, frecuencia de trabajo, etc.).
PR: Resistencia de polarización del dieléctrico.
Cda: Variación de la capacidad debida a la absorción del dieléctrico (dielectric absorption).

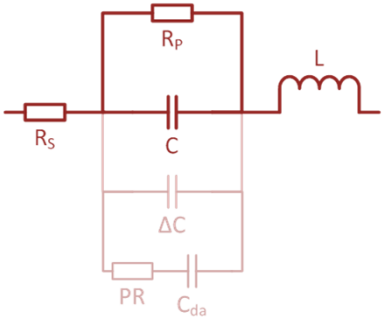
Insertar componentes (C 2)



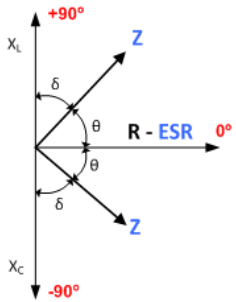
	Frecuencia (Hz)	Z (Ω)	θ (Grados)	Cp (µF)	Rp (Ω)	D	Cs (µF)	Rs (Ω)	
220µ	20	40,91	-88,75	194,39	1.867,12	0,0211	194,47	0,89	85°
220µ	100	8,40	-87,54	189,21	195,32	0,0413	189,56	0,36	85°
220µ	1.000	0,88	-77,46	174,93	4,06	0,2061	183,69	0,19	85°
220µ	10.000	0,19	-29,33	40,26	0,22	1,6089	169,29	0,16	85°
220µ	20.000	0,15	-17,15	15,12	0,16	3,0235	171,20	0,14	85°
220µ	100.000	0,15	0,49	-0,00080	0,16	2,9000	-1.562,00	0,14	85°
220µ	1.000.000	0,21	43,36	-0,000516	0,28	2,5213	-1.224,00	0,13	85°

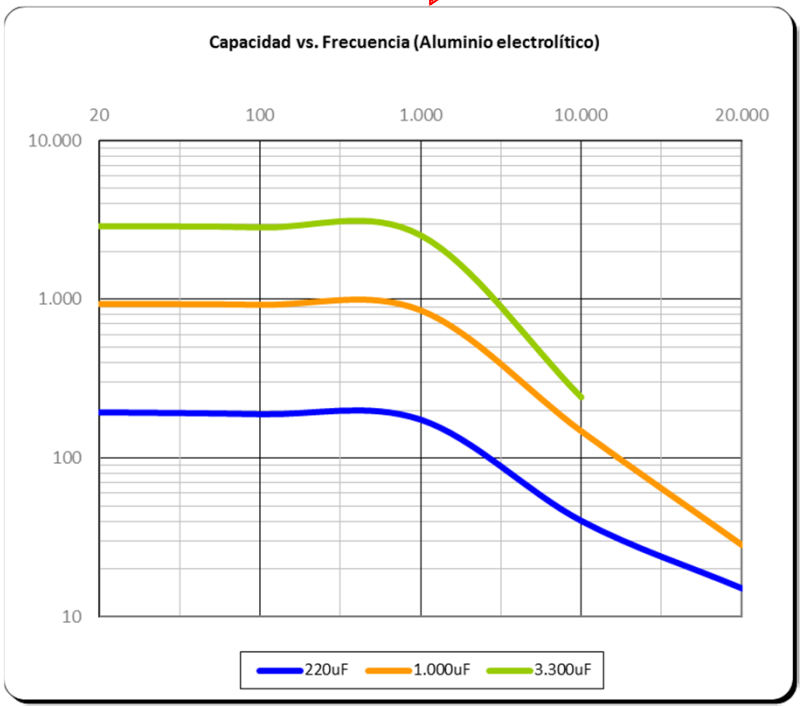


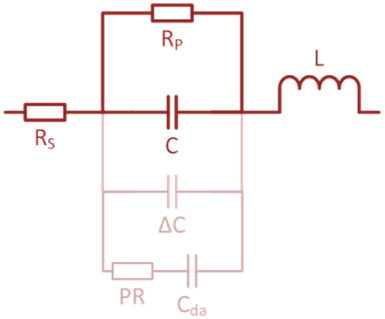




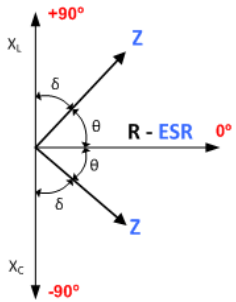
	Frecuencia (Hz)	Z (Ω)	θ (Grados)	Cp (µF)	Rp (Ω)	D	Cs (µF)	Rs (Ω)	
220µ	20	40,91	-88,75	194,39	1.867,12	0,0211	194,47	0,89	85°
220µ	100	8,40	-87,54	189,21	195,32	0,0413	189,56	0,36	85°
220µ	1.000	0,88	-77,46	174,93	4,06	0,2061	183,69	0,19	85°
220µ	10.000	0,19	-29,33	40,26	0,22	1,6089	169,29	0,16	85°
220µ	20.000	0,15	-17,15	15,12	0,16	3,0235	171,20	0,14	85°
220µ	100.000	0,15	0,49	-0,00080	0,16	2,9000	-1.562,00	0,14	85°
220µ	1.000.000	0,21	43,36	-0,000516	0,28	2,5213	-1.224,00	0,13	85°

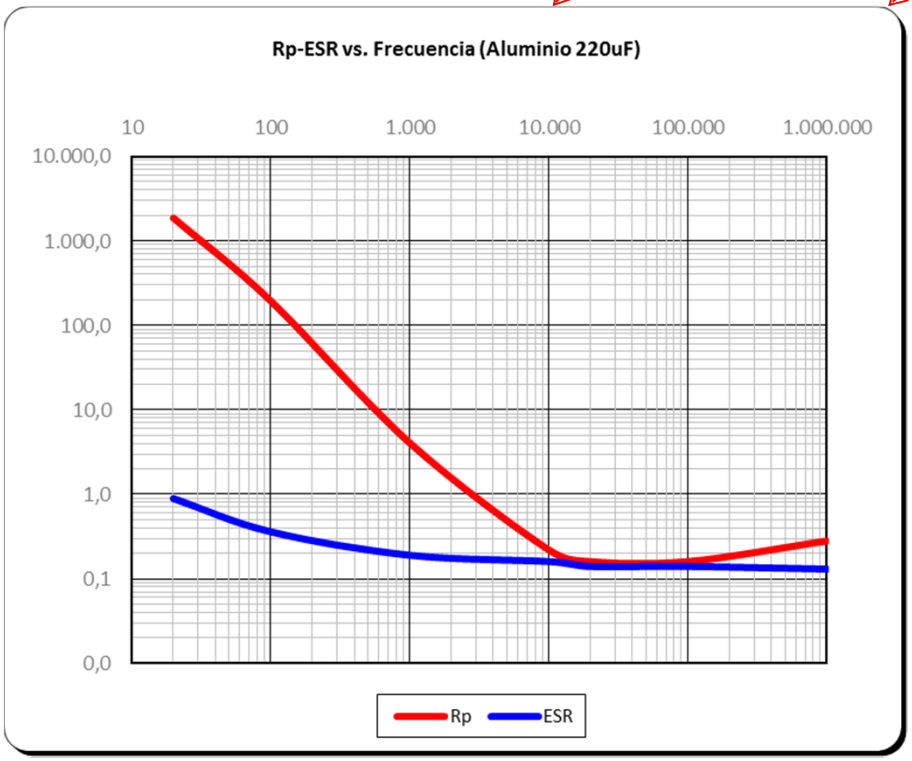




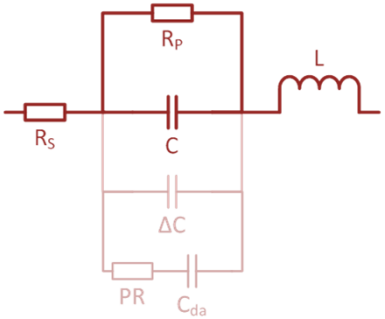


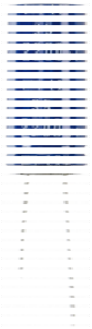
	Frecuencia (Hz)	Z (Ω)	θ (Grados)	Cp (µF)	Rp (Ω)	D	Cs (µF)	Rs (Ω)	
220µ	20	40,91	-88,75	194,39	1.867,12	0,0211	194,47	0,89	85°
220µ	100	8,40	-87,54	189,21	195,32	0,0413	189,56	0,36	85°
220µ	1.000	0,88	-77,46	174,93	4,06	0,2061	183,69	0,19	85°
220µ	10.000	0,19	-29,33	40,26	0,22	1,6089	169,29	0,16	85°
220µ	20.000	0,15	-17,15	15,12	0,16	3,0235	171,20	0,14	85°
220µ	100.000	0,15	0,49	-0,00080	0,16	2,9000	-1.562,00	0,14	85°
220µ	1.000.000	0,21	43,36	-0,000516	0,28	2,5213	-1.224,00	0,13	85°

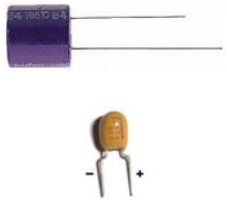


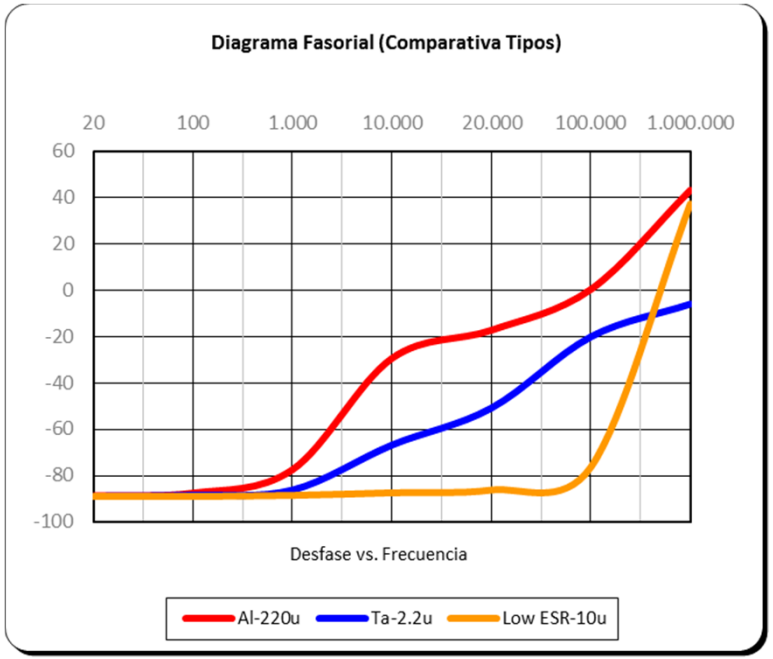


Insertar componentes (C y 5)

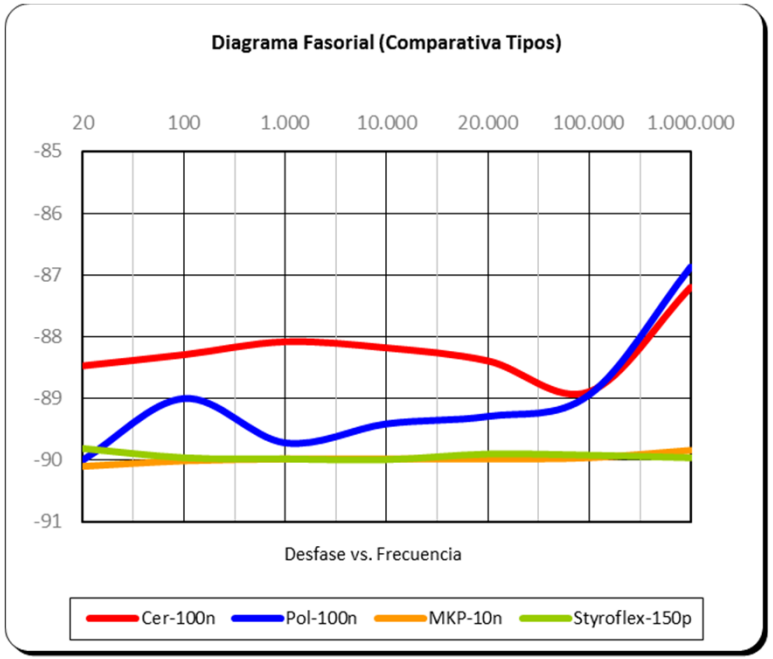




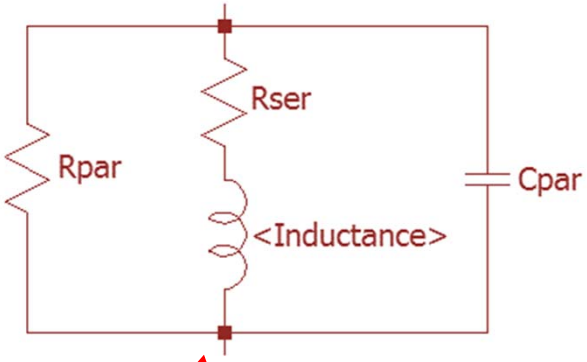


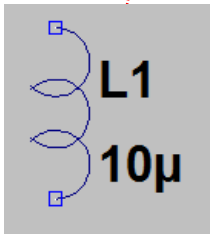


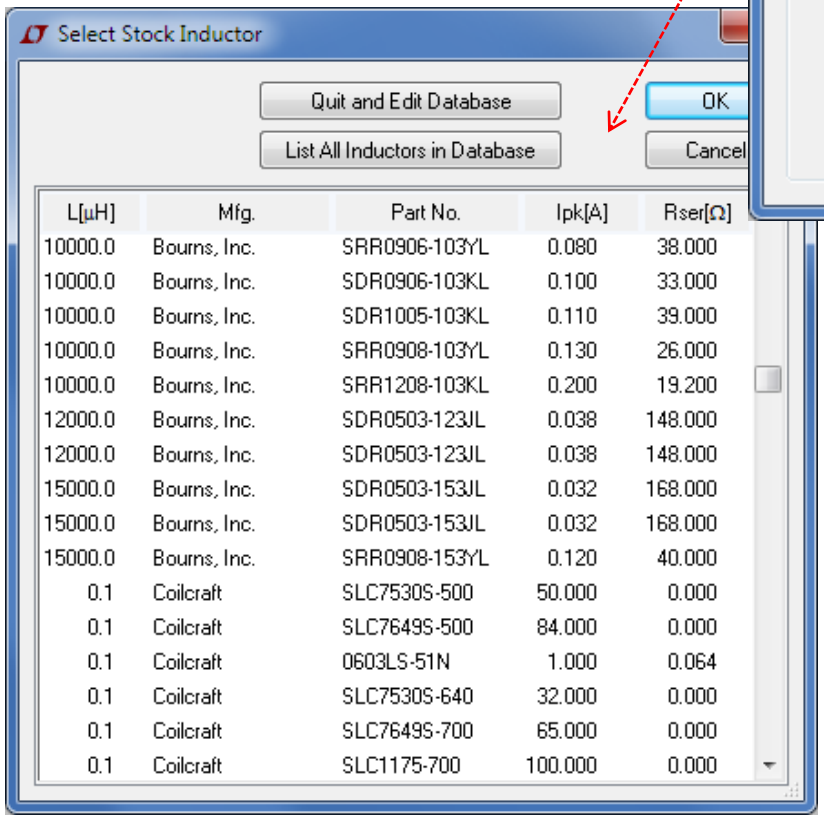
Polarizados



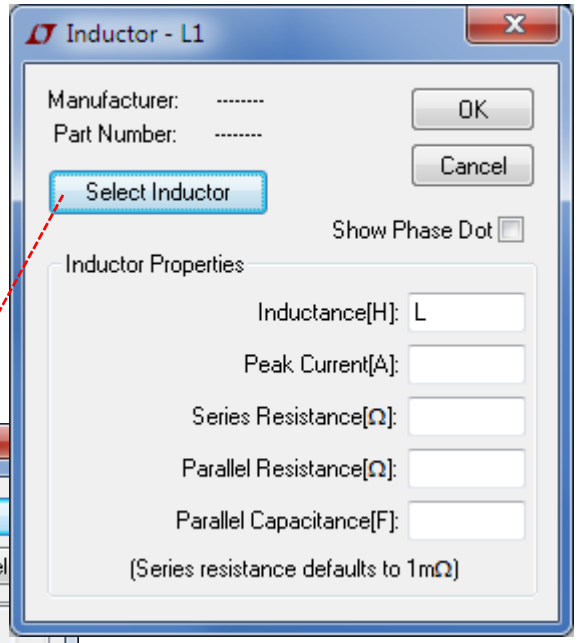
No polarizados



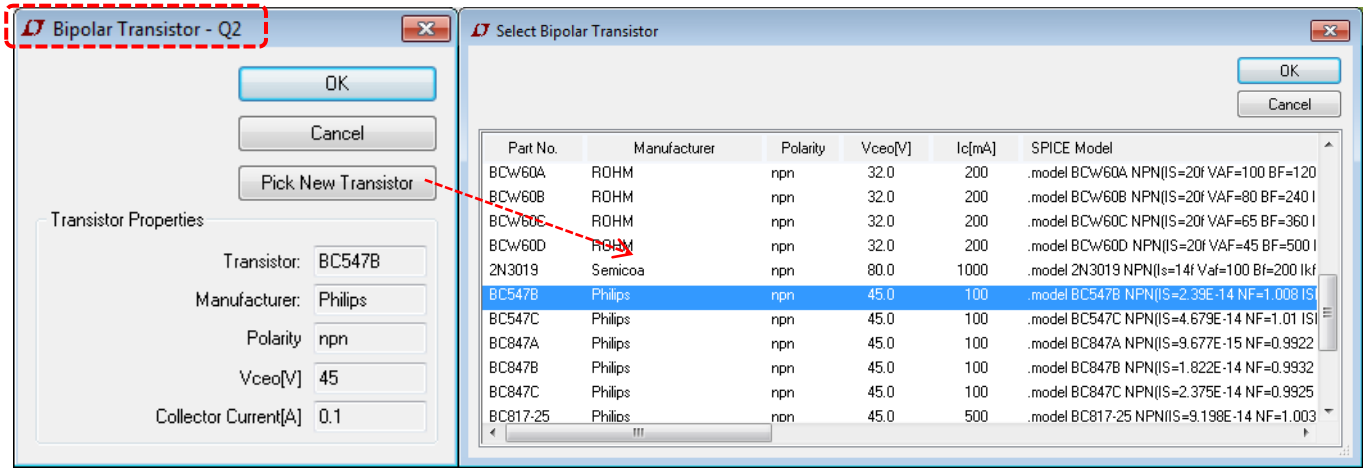




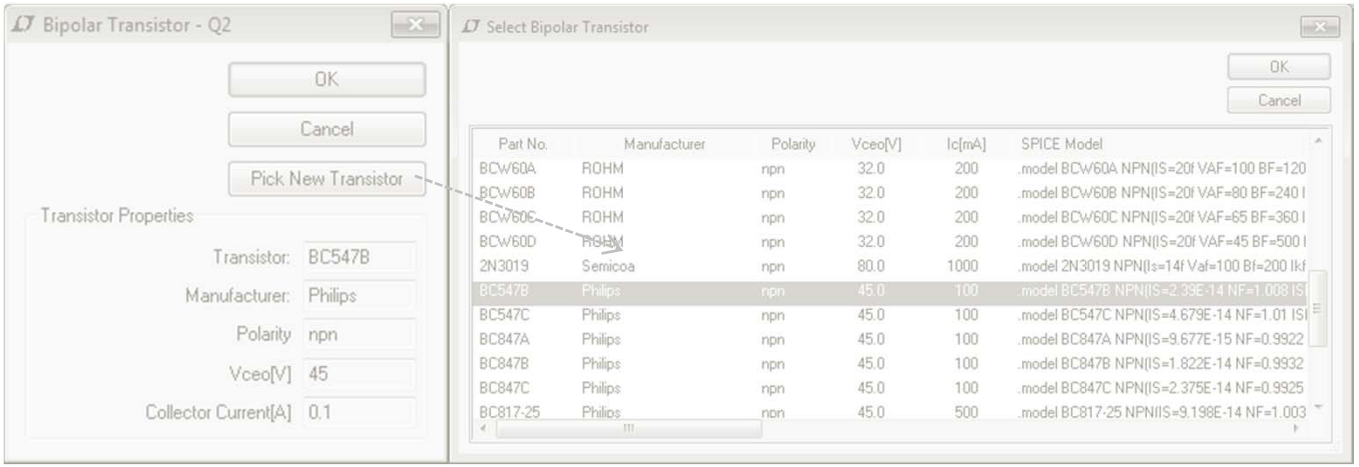
L[µH]	Mfg.	Part No.	Ipk[A]	Rser[Ω]
10000.0	Bourns, Inc.	SRR0906-103YL	0.080	38.000
10000.0	Bourns, Inc.	SDR0906-103KL	0.100	33.000
10000.0	Bourns, Inc.	SDR1005-103KL	0.110	39.000
10000.0	Bourns, Inc.	SRR0908-103YL	0.130	26.000
10000.0	Bourns, Inc.	SRR1208-103KL	0.200	19.200
12000.0	Bourns, Inc.	SDR0503-123JL	0.038	148.000
12000.0	Bourns, Inc.	SDR0503-123JL	0.038	148.000
15000.0	Bourns, Inc.	SDR0503-153JL	0.032	168.000
15000.0	Bourns, Inc.	SDR0503-153JL	0.032	168.000
15000.0	Bourns, Inc.	SRR0908-153YL	0.120	40.000
0.1	Coilcraft	SLC7530S-500	50.000	0.000
0.1	Coilcraft	SLC7649S-500	84.000	0.000
0.1	Coilcraft	0603LS-51N	1.000	0.064
0.1	Coilcraft	SLC7530S-640	32.000	0.000
0.1	Coilcraft	SLC7649S-700	65.000	0.000
0.1	Coilcraft	SLC1175-700	100.000	0.000

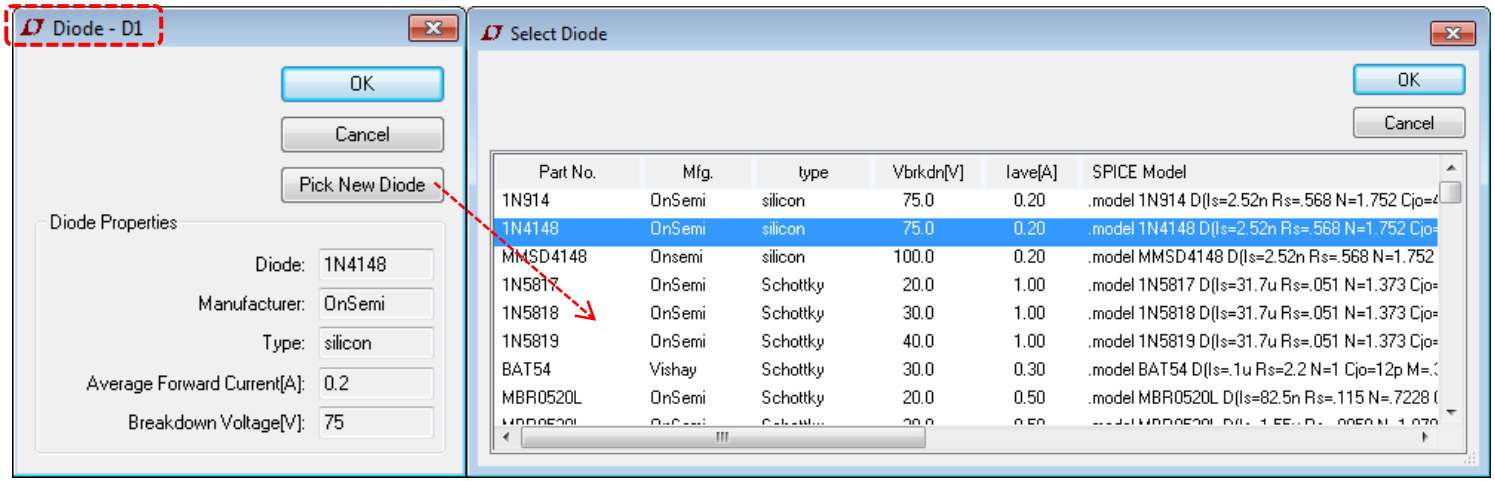


Insertar componentes

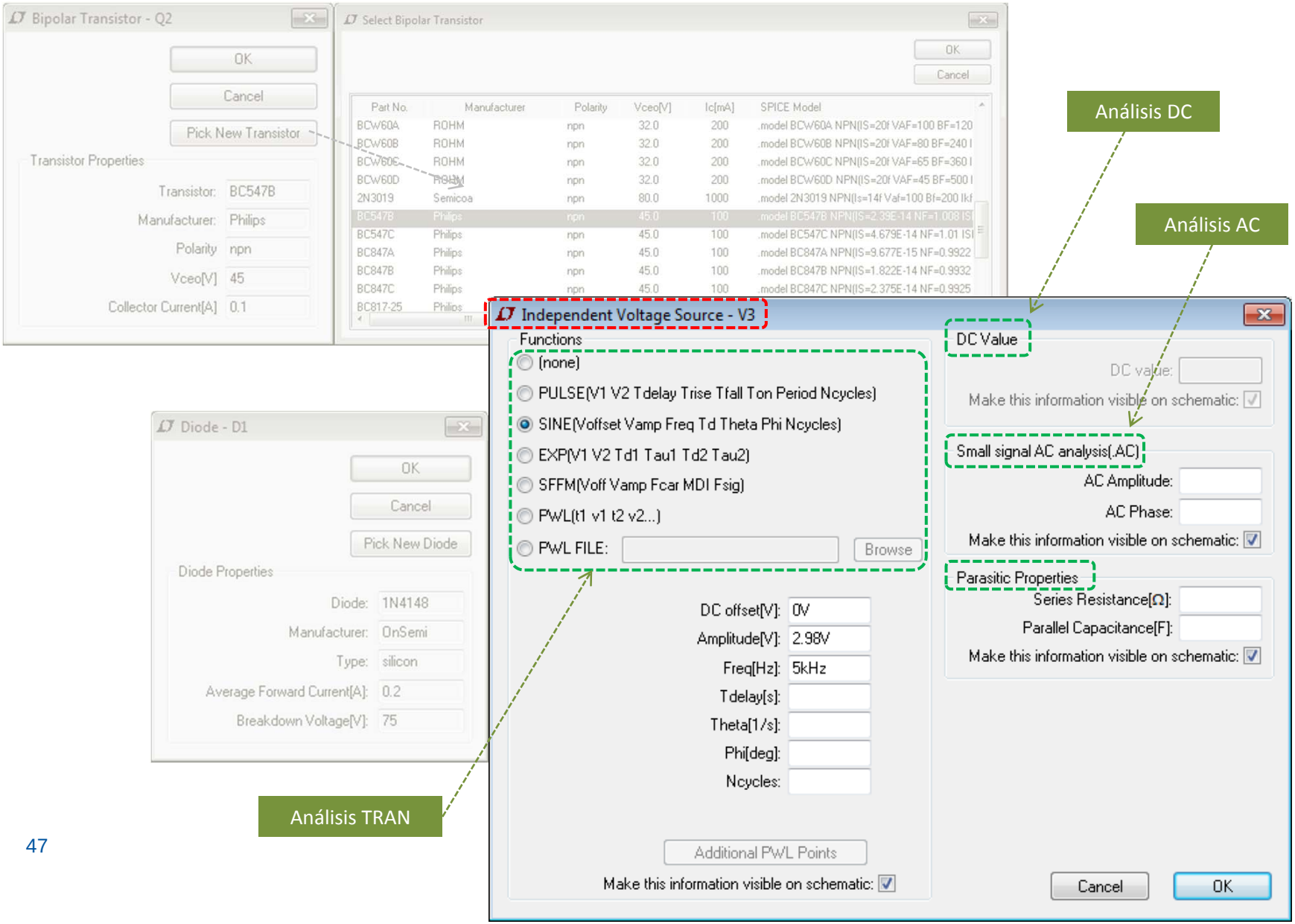


Insertar componentes





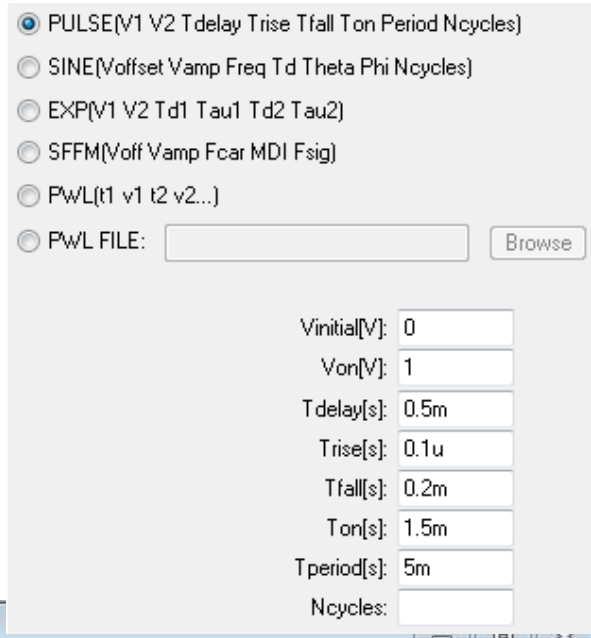
Insertar componentes (V)

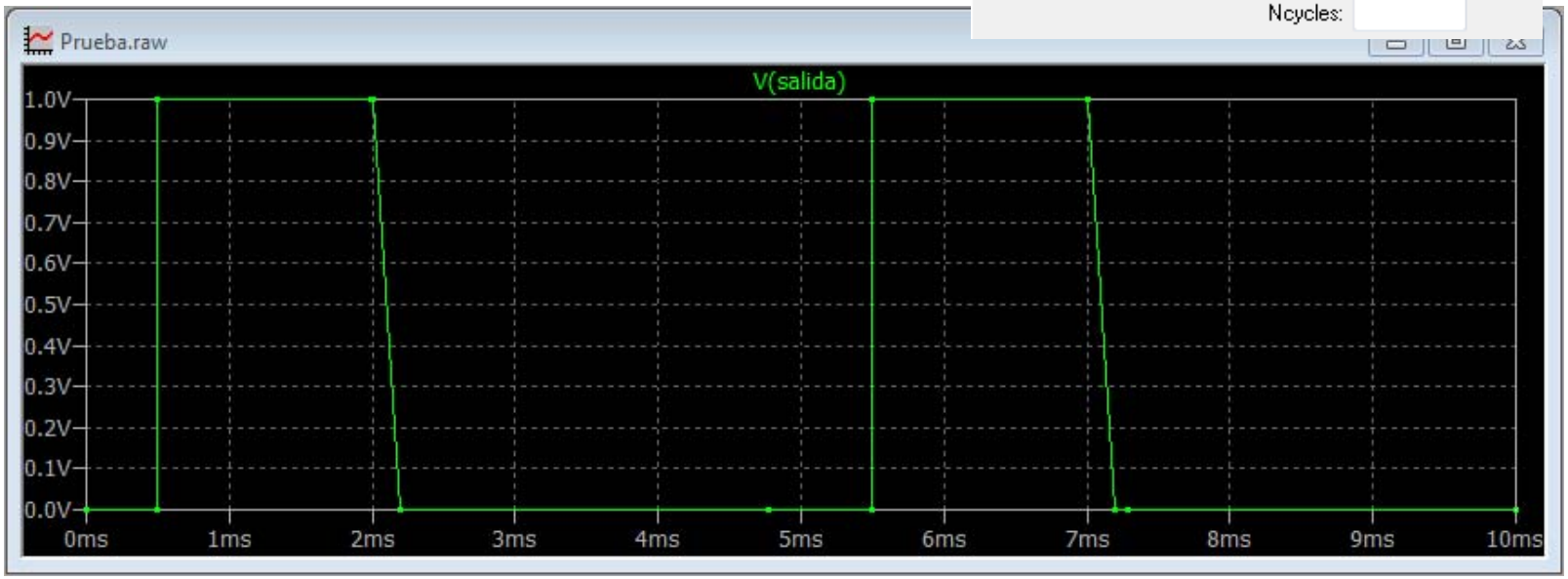


Formas de onda

Insertar componentes (V 2)

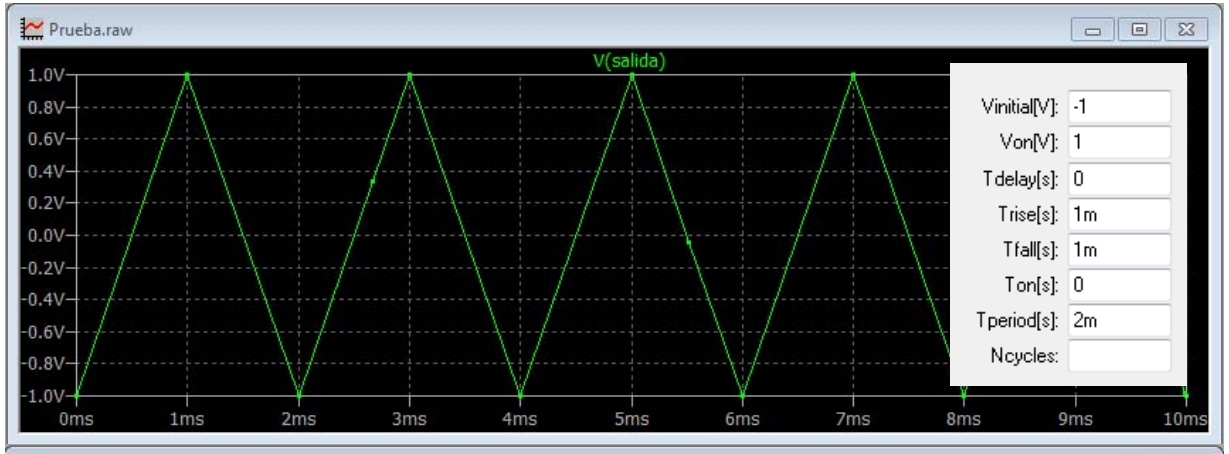
Onda PULSOS

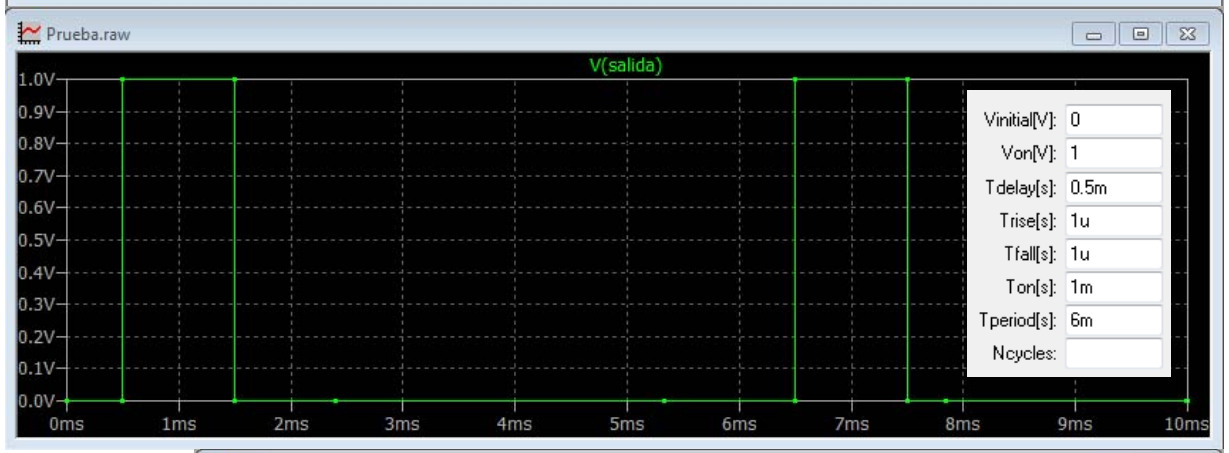


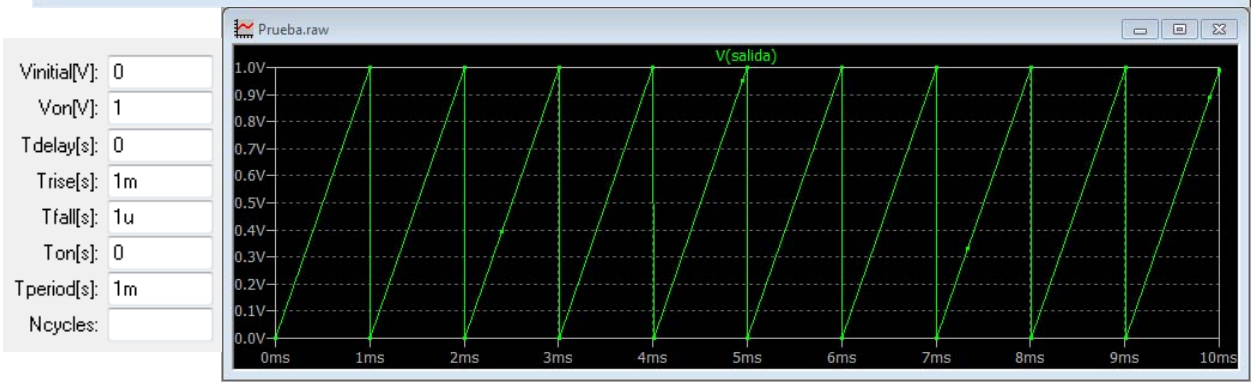


Onda PULSOS

TON = 0 → Triangular
TR, TF ≠ 0 → Trapezoidal
TR = 0, TF ≠ 0 → Diente Sierra
TR, TF = 0 → Cuadrada
TON = ∞ → Escalón unitario
PER = ∞ → Pulso único



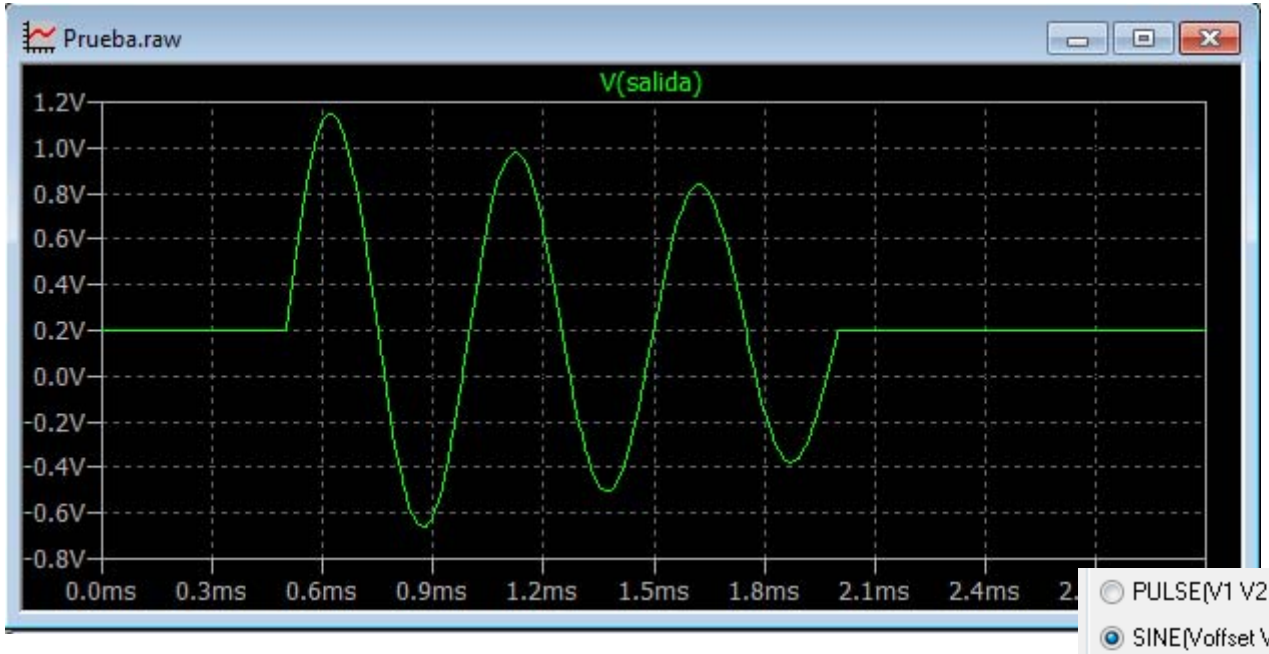


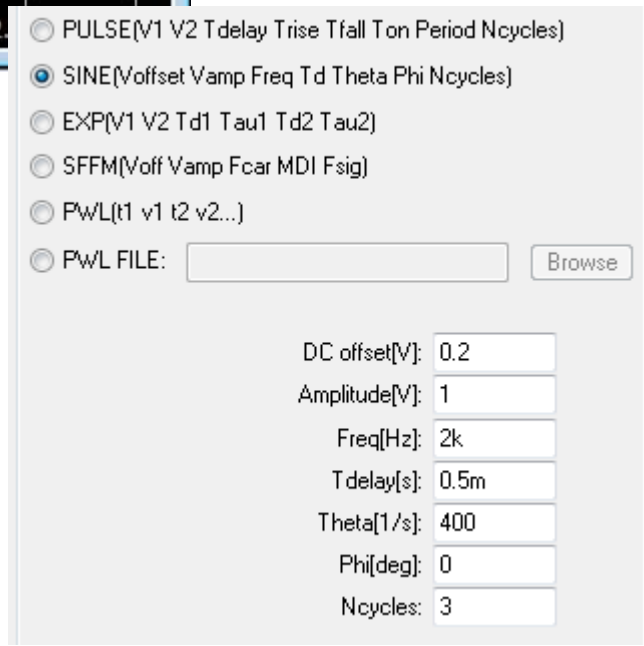


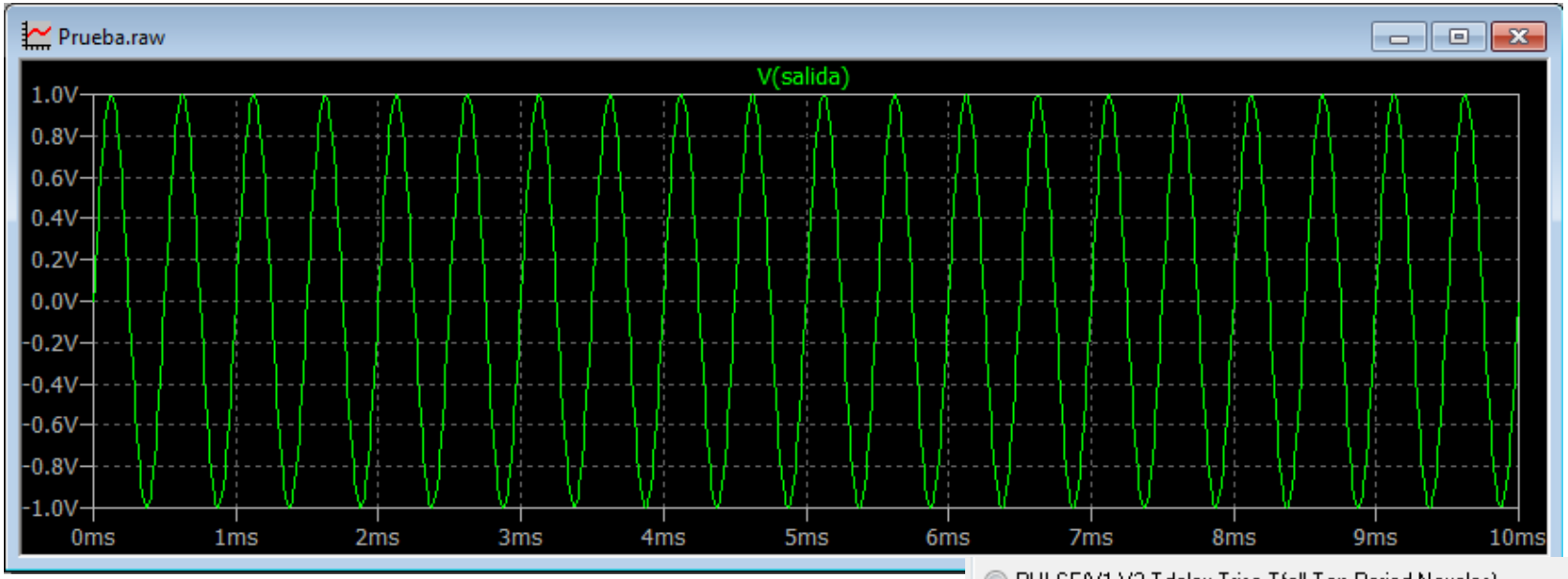
Onda PULSOS

TON = 0 → Triangular
TR, TF ≠ 0 → Trapezoidal
TR = 0, TF ≠ 0 → Diente Sierra
TR, TF = 0 → Cuadrada
TON = ∞ → Escalón unitario
PER = ∞ → Pulso único

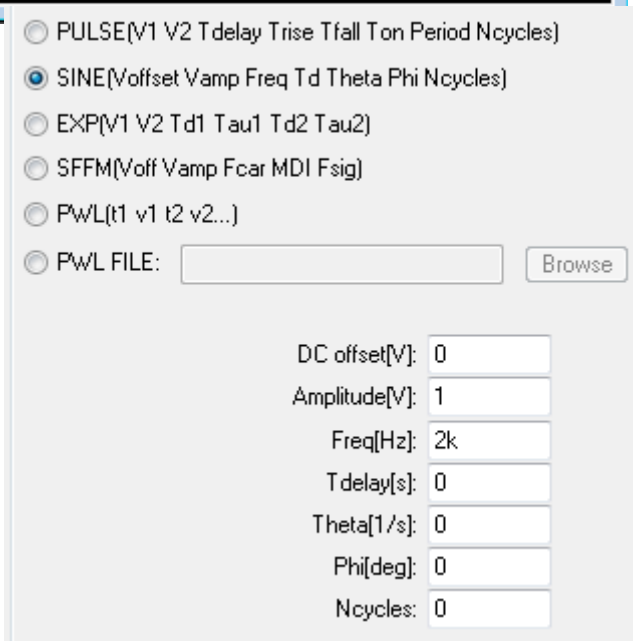
Onda SENOIDAL



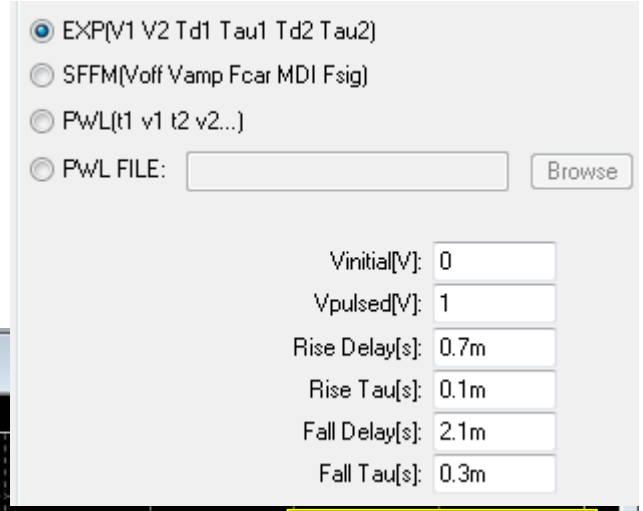




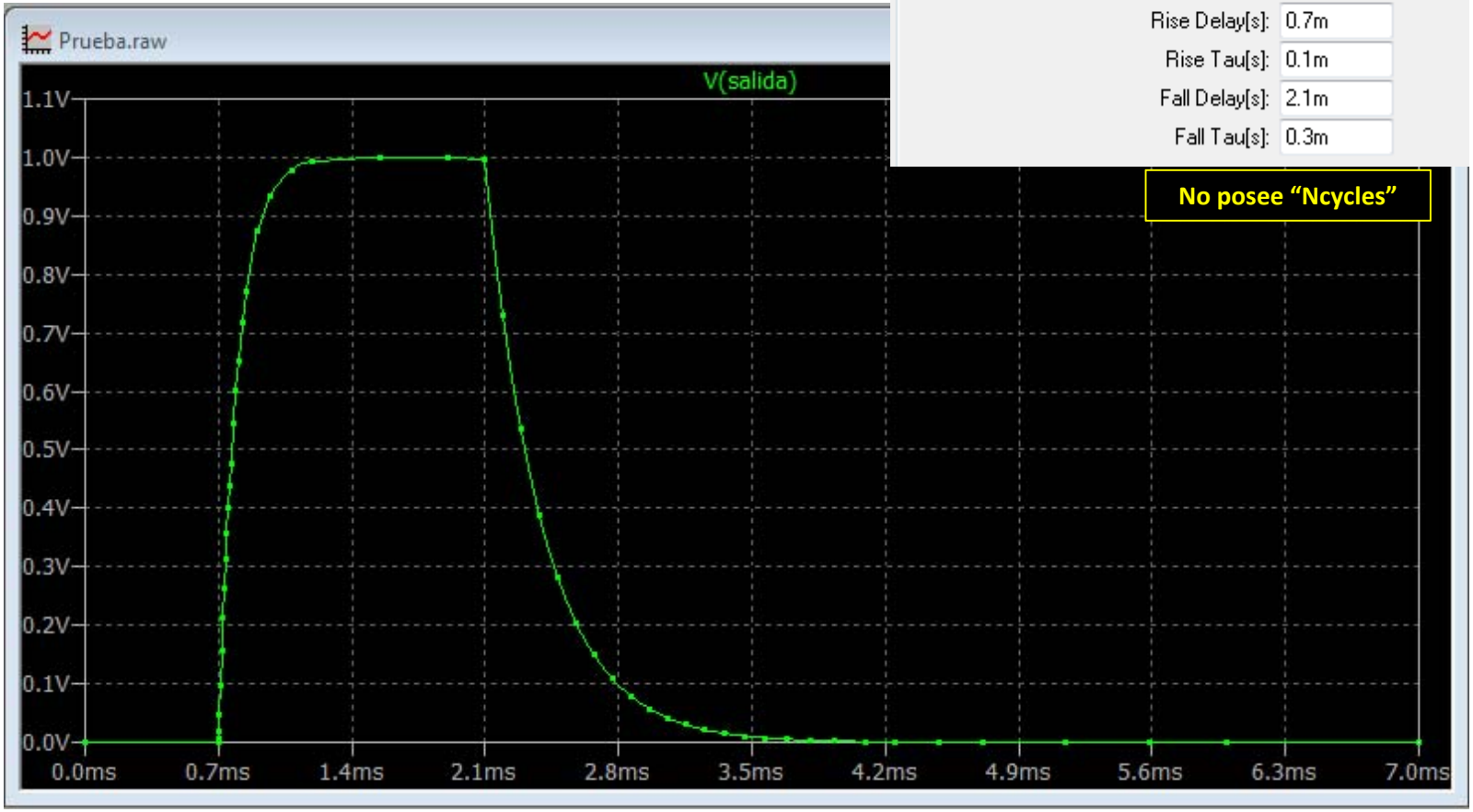
Onda SENOIDAL



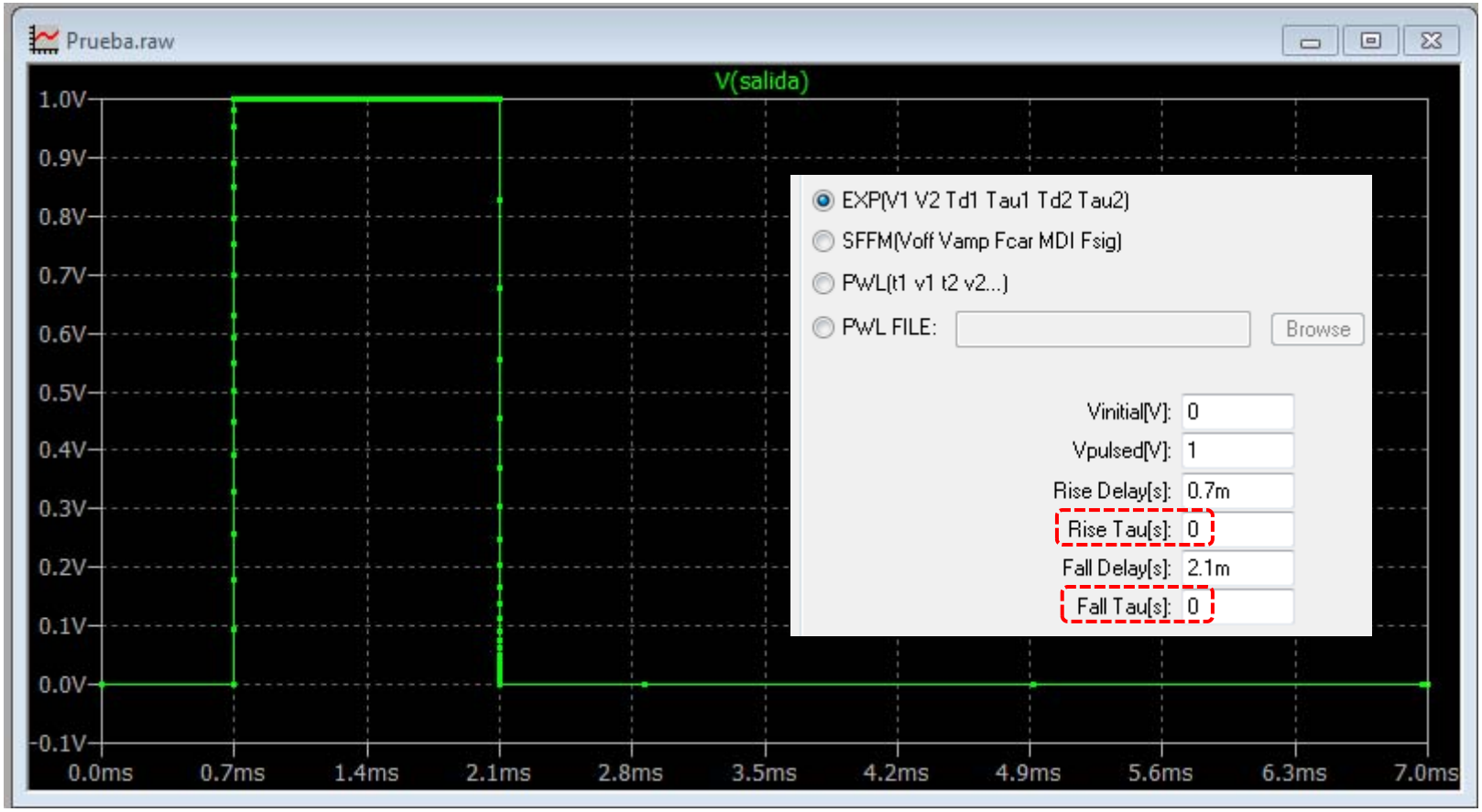
Onda EXPONENCIAL



No posee "Ncycles"

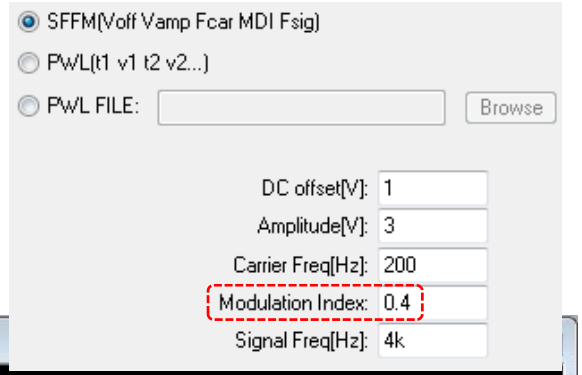


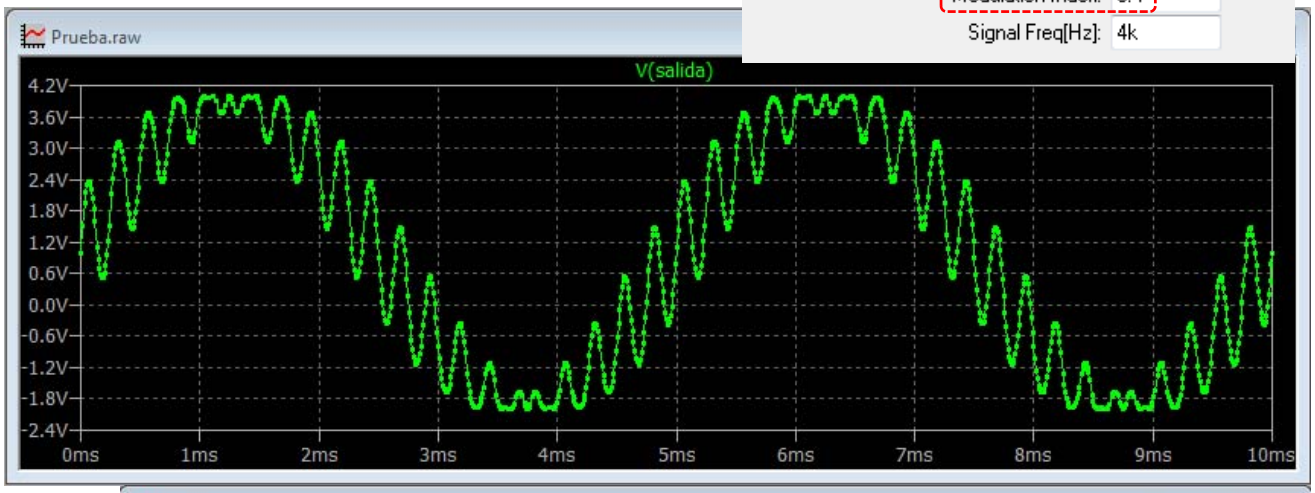
Onda EXPONENCIAL

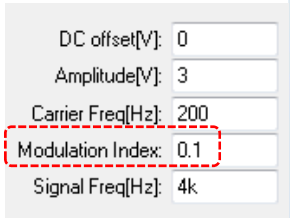


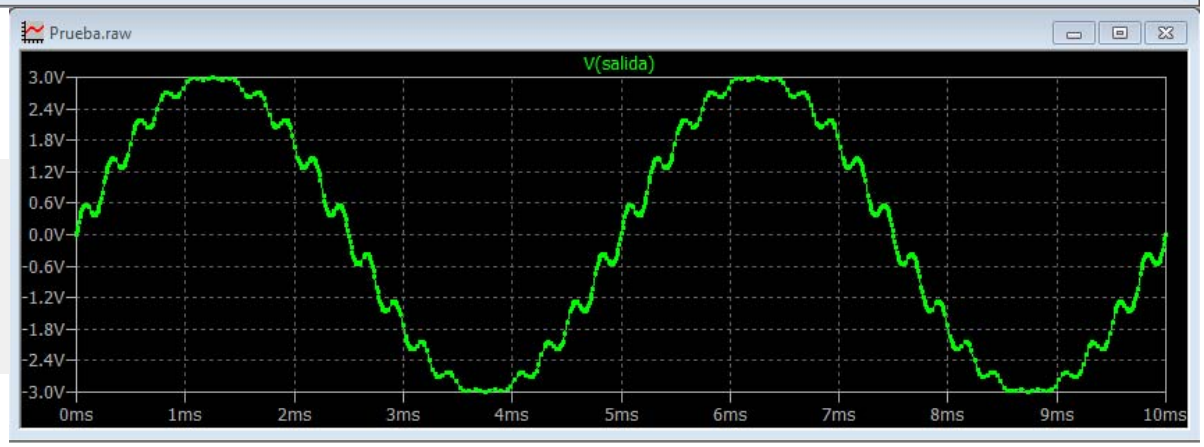
Onda simple de FM

Single Frecuency FM (SFFM)

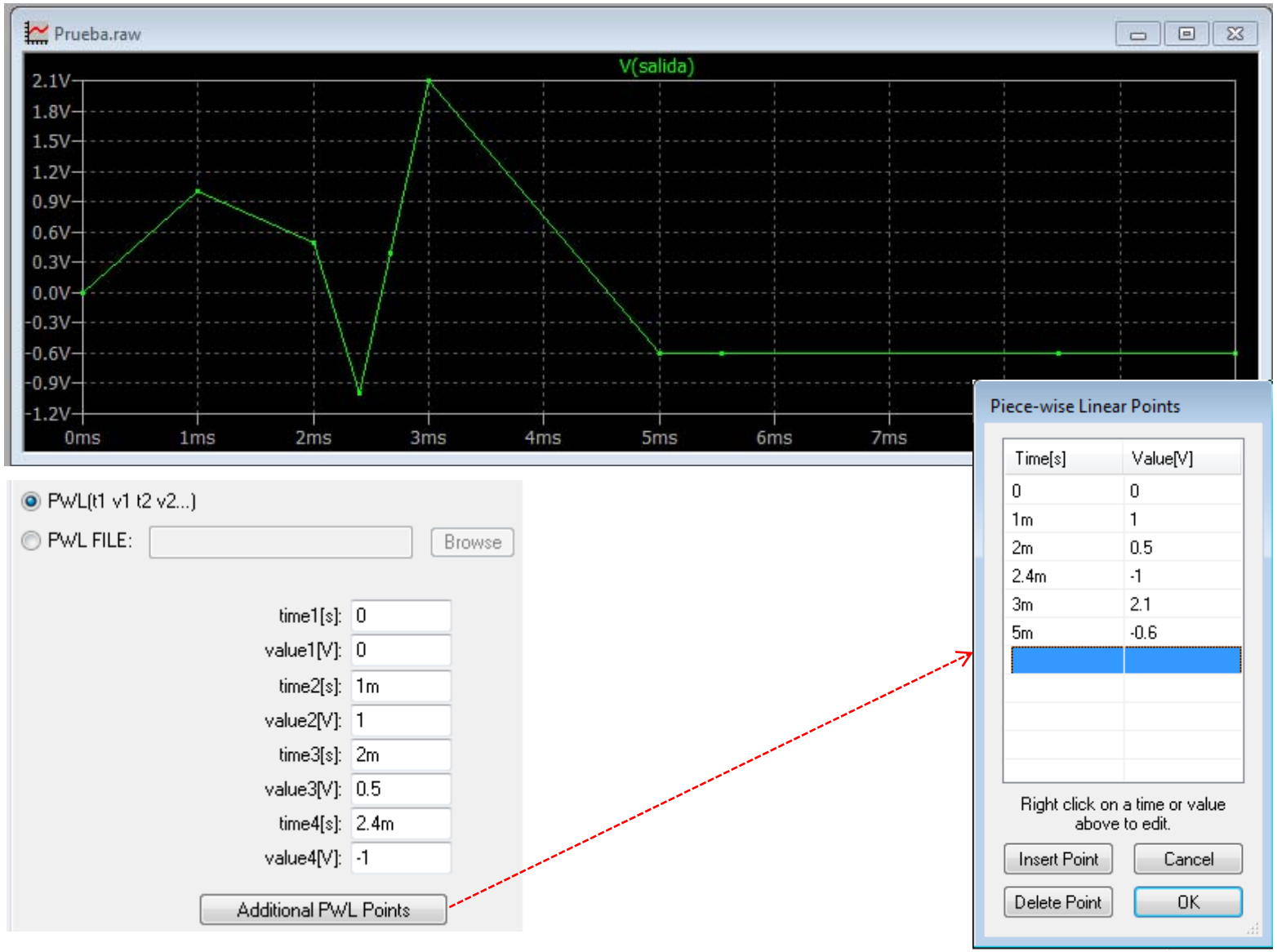




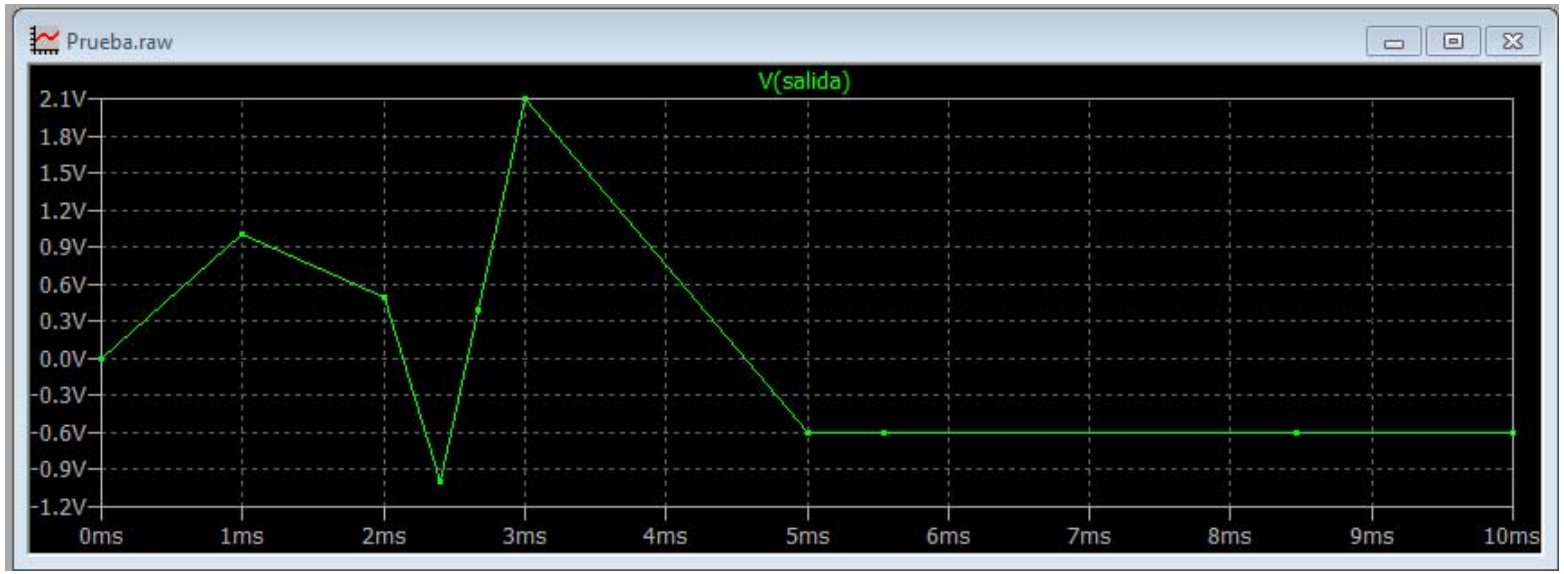


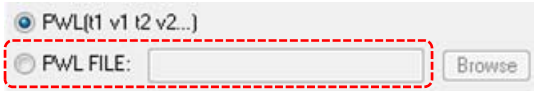


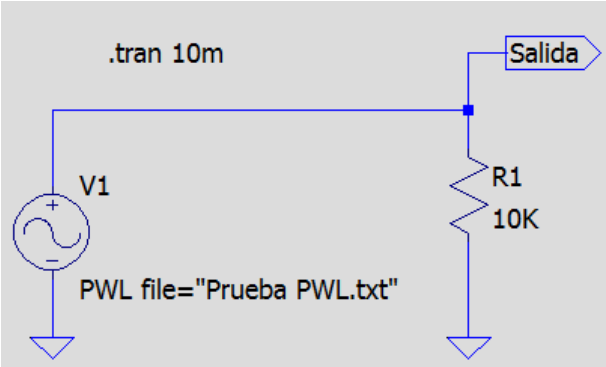
Onda PWL (Piece Wise Linear) | Lineal por tramos

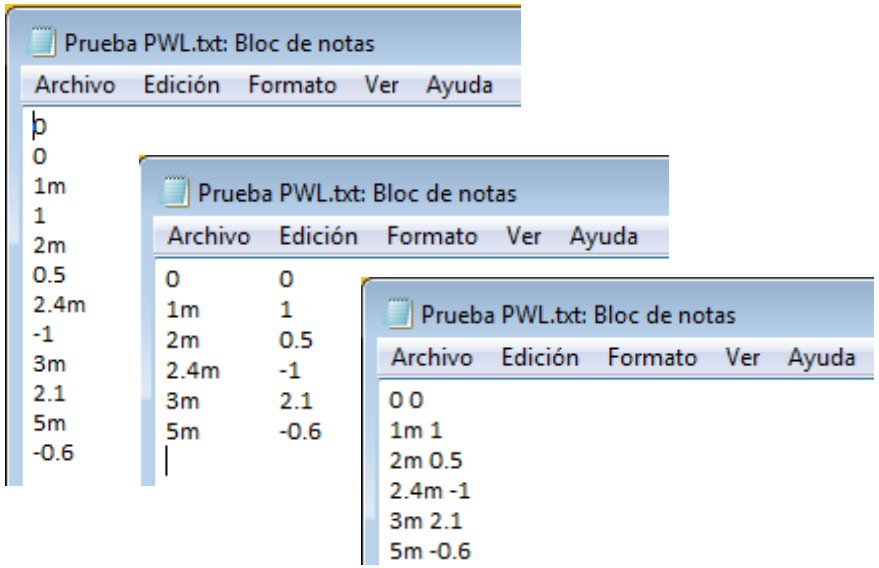


Onda PWL (Piece Wise Linear) | Lineal por tramos

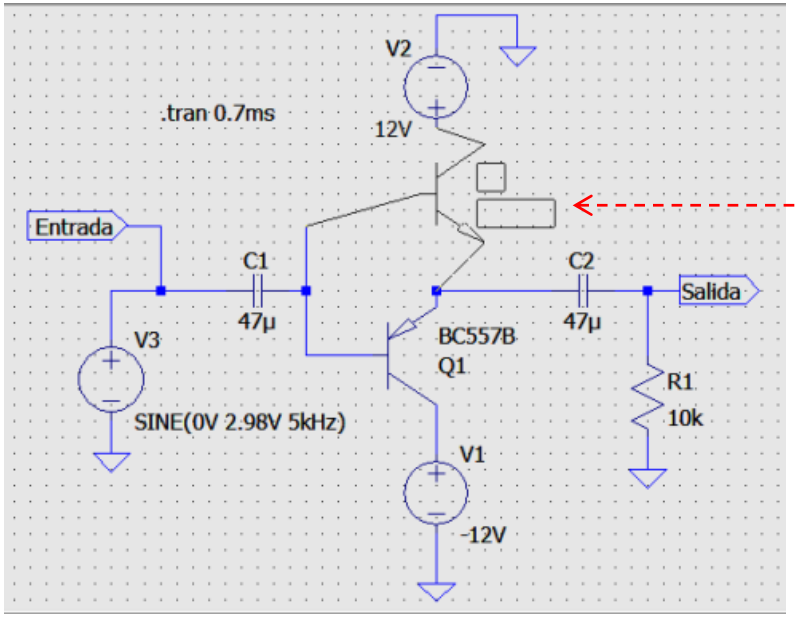




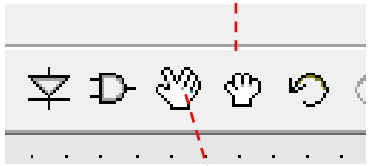




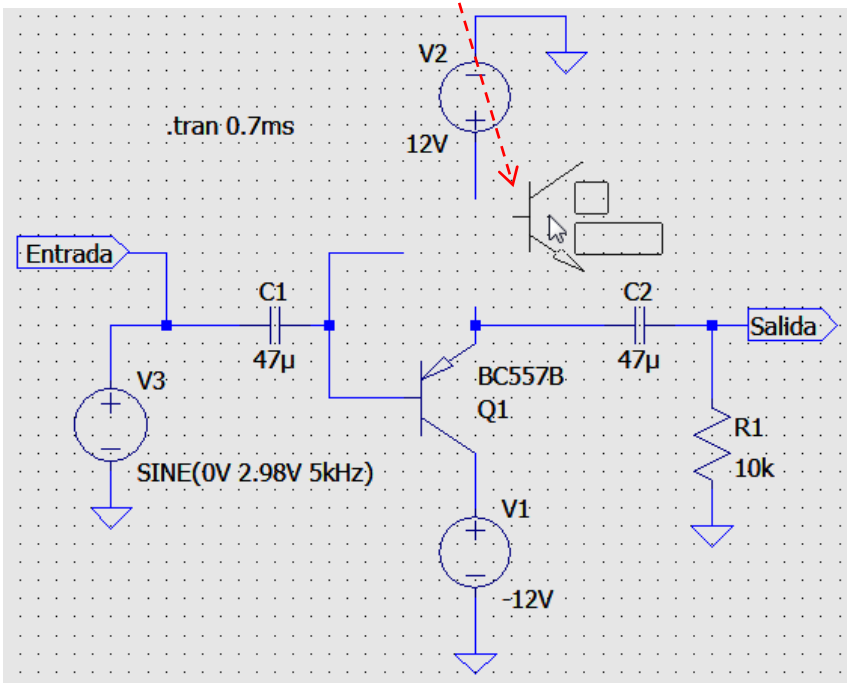
Mover/Arrastrar



Arrastrar



Mover



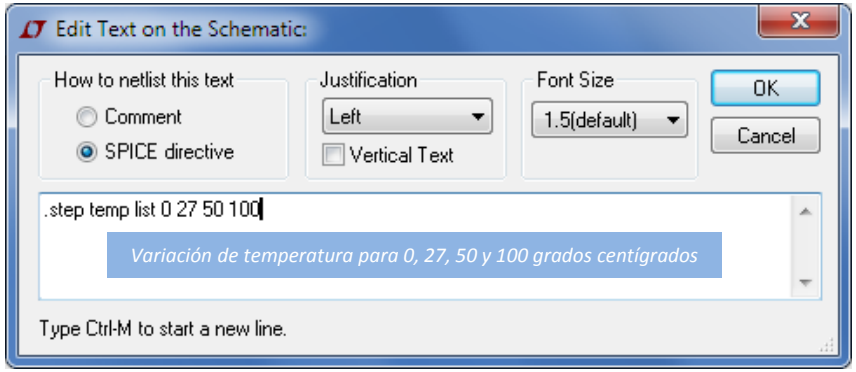
LTSpice puede realizar seis tipos de análisis:

1. AC respecto al punto de trabajo (AC)
2. Barrido en DC (DC Sweep)
3. Ruido (Noise)
4. Punto de trabajo en DC (OP)
5. Función de Transferencia con pequeña señal (TF)
6. Análisis transitorio (TRAN)

Estos análisis y otras características se incorporan al fichero de *Netlist* comenzando por un punto. Por eso se les denomina **comandos de punto**.

.step temp -55 125 10

Step the temperature from -55°C to 125°C in 10 degree step. Step sweeps may be nested up to three levels deep.



Variable Global

Examples:
.step oct V1 1 20 5
.step I1 10u 100u 10u
.step param RLOAD LIST 5 10 15
.step NPN BC547C(BF) LIST 100 350

Parámetro Interno

- *Variación Logarítmica (Octavas) de* V_1 *de 1V a 20V con 5 muestras entre octavas.*
- *Variación lineal de* I_1 *de 10uA a 100uA en saltos de 10uA.*
- *Modificación del parámetro (valor)* R_{LOAD}*: 5Ω, 10Ω y 15Ω. (Lista de valores)*
- *La ganancia de corriente (**Beta Forward**) del BJT BC547C será de 100 y de 350.*

LTSpice puede realizar seis tipos de análisis:

1. AC respecto al punto de trabajo (AC)
2. Barrido en DC (DC Sweep)
3. Ruido (Noise)
4. Punto de trabajo en DC (OP)
5. Función de Transferencia con pequeña señal (TF)
6. Análisis transitorio (TRAN)

Estos análisis y otras características se incorporan al fichero de *Netlist* comenzando por un punto. Por eso se les denomina **comandos de punto**.

Syntax: **.AC <oct, dec, lin> <Nsteps> <StartFreq> <EndFreq>**

ANALISIS Y FUNCIONES	COMANDOS
AC or frequency analysis	.AC
DC operating analysis	.OP
DC sweep	.DC
End of subcircuit	.ENDS
Fourier analysis	.FOUR
Frequency response transfer function	.FREQ
Function definition user	.FUNC
Global nodes	.GLOBAL
Graphical postprocessor	.PROBE
Include file	.INC - INCLUDE
Initial conditions	.IC
Library file	.LIB
Model definition	.MODEL
Node setting	.NODESET
Noise analysis	.NOISE
Options	.OPTIONS
Parameter definition	PARAM
Parameter variation	.PARAM
Parametric analysis	.STEP
Plot output	.PLOT
Print output	.PRINT
Sensitivity analysis	.SENS
Subcircuit definition	.SUBCKT
Table	TABLE
Temperature	.TEMP
Transfer function	.TF
Transient analysis	.TRAN
Value	VALUE
Width	.WIDTH

LTSpice puede realizar seis tipos de análisis:

1. AC respecto al punto de trabajo (AC)
2. Barrido en DC (DC Sweep)
3. Ruido (Noise)
4. Punto de trabajo en DC (OP)
5. Función de Transferencia con pequeña señal (TF)
6. Análisis transitorio (TRAN)

Estos análisis y otras características se incorporan al fichero de *Netlist* comenzando por un punto. Por eso se les denomina **comandos de punto**.

Syntax: **.AC <oct, dec, lin> <Nsteps> <StartFreq> <EndFreq>**

.BACKANNO	Annotate the Subcircuit Pin Names to the Port Currents
.END	End of Netlist
.FERRET	Download a File Given the URL
.LOADBIAS	Load a Previously Solved DC Solution
.MEASURE	Evaluate User-Defined Electrical Quantities
.NET	Compute Network Parameters in a .AC Analysis
.SAVE	Limit the Quantity of Saved Data
.SAVEBIAS	Save Operating Point to Disk
.WAVE	Write Selected Nodes to a .Wav File

ANALISIS Y FUNCIONES	COMANDOS
AC or frequency analysis	.AC
DC operating analysis	.OP
DC sweep	.DC
End of subcircuit	.ENDS
Fourier analysis	.FOUR
Frequency response transfer function	.FREQ
Function definition user	.FUNC
Global nodes	.GLOBAL
Graphical postprocessor	.PROBE
Include file	.INC - INCLUDE
Initial conditions	.IC
Library file	.LIB
Model definition	.MODEL
Node setting	.NODESET
Noise analysis	.NOISE
Options	.OPTIONS
Parameter definition	PARAM
Parameter variation	.PARAM
Parametric analysis	.STEP
Plot output	.PLOT
Print output	.PRINT
Sensitivity analysis	.SENS
Subcircuit definition	.SUBCKT
Table	TABLE
Temperature	.TEMP
Transfer function	.TF
Transient analysis	.TRAN
Value	VALUE
Width	.WIDTH

```
Syntax: Para un punto del eje de Abscisas
.MEAS[SURE] [AC|DC|OP|TRAN|TF|NOISE] <name>
+ [<FIND|DERIV|PARAM> <expr>]
+ [WHEN <expr> | AT=<expr>]]
+ [TD=<val1>] [<RISE|FALL|CROSS>=[<count1>|LAST]]
```

```
.MEAS TRAN res5 FIND V(out) WHEN V(x)=3*V(y) cross=3 TD=1m
```

*Indica el valor de V(out) la tercera ocasión que se encuentre la condición V(x)=3*V(y), pero no empiece a contar (demora) hasta pasado 1ms. A esto se le nombrará como* ***res5****.*

```
Syntax: Para un rango a lo largo del eje de Abscisas
.MEAS [AC|DC|OP|TRAN|TF|NOISE] <name>
+ [<AVG|MAX|MIN|PP|RMS|INTEG> <expr>]
+ [TRIG <lhs1> [[VAL]=]<rhs1>] [TD=<val1>]
+ [<RISE|FALL|CROSS>=<count1>]
+ [TARG <lhs2> [[VAL]=]<rhs2>] [TD=<val2>]
+ [<RISE|FALL|CROSS>=<count2>]

El rango sobre el eje de Abscisas esta definido por "TRIG" y "TARG". El valor TRIG, si se omite,
es, por defecto, el punto de comienzo de la simulación. De igual forma, TARG es el punto del
final de la simulación. Si se omiten estos datos, el rango de datos es el tiempo total de la
simulación. Los tipos de operaciones de medida que se pueden realizar en el intervalo son:
<AVG|MAX|MIN|PP|RMS|INTEG>
```

```
.MEAS TRAN res7 AVG V(NS01) TRIG V(NS05) VAL=1.5 TD=1.1u FALL=1 TARG V(NS03) VAL=1.5 TD=1.1u FALL=1
```

Indica el valor medio de V(NS01) desde la primera vez que descienda V(NS05) de 1.5V, después de 1.1us y la primera vez que descienda V(NS03) de 1.5V, después 1.1us. Se nombrará ***res7****.*

El resultado de un comando .MEAS se puede enlazar en otro comando .MEAS. En este ejemplo, se calcula el ancho de banda a -3dB:

```
.MEAS AC tmp max mag(V(out)); Encuentra el valor de pico y le llama "tmp"
.MEAS AC BW trig mag(V(out))=tmp/sqrt(2) rise=1 targ mag(V(out))=tmp/sqrt(2) fall=last
```

Syntax: **.MODEL** <modname> <**type**>[(<parameter list>)]

The parameter list depends on the type of model. Below is a list of model types:

Type	Associated Circuit Element
SW	Voltage Controlled Switch
CSW	Current Controlled Switch
URC	Uniform Distributed RC Line
LTRA	Lossy Transmission Line
D	Diode
NPN	NPN Bipolar Transistor
PNP	PNP Bipolar Transistor
NJF	N-channel JFET model
PJF	P-channel JFET model
NMOS	N-channel MOSFET
PMOS	P-channel MOSFET
NMF	N-channel MESFET
PMF	P-channel MESFET
VDMOS	Vertical Double Diffused Power MOSFET

Tipos de componentes

.PARAM – Parámetros definidos por el usuario

The .param directive allows the creation of user-defined variables. This is useful for associating a name with a value for the sake of clarity and parameterizing subcircuits so that abstract circuits can be saved in libraries.

The .param statement can be included inside a subcircuit definition to limit the scope the parameter value to that subcircuit.

Operaciones trigonométricas y matemáticas

abs(x)	Absolute value of x
acos(x)	Real part of the arc cosine of x, e.g., acos(-5) returns 3.14159, not 3.14159+2.29243i
arccos(x)	Synonym for acos()
acosh(x)	Real part of the arc hyperbolic cosine of x, e.g., acosh(.5) returns 0, not 1.0472i
asin(x)	Real part of the arc sine of x, e.g., asin(-5) returns -1.57080, not -1.57080+2.29243i
arcsin(x)	Synonym for asin()
asinh(x)	Arc hyperbolic sine
atan(x)	Arc tangent of x
arctan(x)	Synonym for atan()
atan2(y,x)	Four quadrant arc tangent of y/x
atanh(x)	Arc hyperbolic tangent
buf(x)	1 if x > .5, else 0
cbrt(x)	Cube root of (x)
ceil(x)	Integer equal or greater than x
cos(x)	Cosine of x
cosh(x)	Hyperbolic cosine of x
exp(x)	e to the x
fabs(x)	Same as abs(x)
flat(x)	Random number between -x and x with uniform distribution
floor(x)	Integer equal to or less than x
gauss(x)	Random number from Gaussian distribution with sigma of x.
hypot(x,y)	sqrt(x**2 + y**2)
if(x,y,z)	If x > .5, then y else z
int(x)	Convert x to integer
inv(x)	0. if x > .5, else 1.
limit(x,y,z)	Intermediate value of x, y, and z
ln(x)	Natural logarithm of x
log(x)	Alternate syntax for ln()
log10(x)	Base 10 logarithm
max(x,y)	The greater of x or y
mc(x,y)	A random number between x*(1+y) and x*(1-y) with uniform distribution.
min(x,y)	The smaller of x or y
pow(x,y)	Real part of x**y, e.g., pow(-.5,1.5) returns 0., not 0.353553i
pwr(x,y)	abs(x)**y
pwrs(x,y)	sgn(x)*abs(x)**y
rand(x)	Random number between 0 and 1 depending on the integer value of x.
random(x)	Similar to rand(), but smoothly transitions between values.
round(x)	Nearest integer to x
sgn(x)	Sign of x
sin(x)	Sine of x
sinh(x)	Hyperbolic sine of x
sqrt(x)	Real part of the square root of x, e.g., sqrt(-1) returns 0, not 0.707107i
table(x,a,b,c,d,...)	Interpolate a value for x based on a look up table given as a set of pairs of points.
tan(x)	Tangent of x.
tanh(x)	Hyperbolic tangent of x
u(x)	Unit step, i.e., 1 if x > 0., else 0.
uramp(x)	x if x > 0., else 0.

.PARAM – Parámetros definidos por el usuario

The .param directive allows the creation of user-defined variables. This is useful for associating a name with a value for the sake of clarity and parameterizing subcircuits so that abstract circuits can be saved in libraries.

The .param statement can be included inside a subcircuit definition to limit the scope the parameter value to that subcircuit.

&	Convert the expressions to either side to Boolean, then AND.
|	Convert the expressions to either side to Boolean, then OR.
^	Convert the expressions to either side to Boolean, then XOR.
>	True if expression on the left is greater than the expression on the right, otherwise false.
<	True if expression on the left is less than the expression on the right, otherwise false.
>=	True if expression on the left is greater than or equal the expression on the right, otherwise false.
<=	True if expression on the left is less than or equal the expression on the right, otherwise false.
+	Floating point addition
-	Floating point subtraction
*	Floating point multiplication
/	Floating point division
**	Raise left hand side to power of right hand side, only real part is returned, e.g., -2**1.5 returns zero, not 2.82843i

Booleanas y matemáticas

abs(x)	Absolute value of x
acos(x)	Real part of the arc cosine of x, e.g., acos(-5) returns 3.14159, not 3.14159+2.29243i
arccos(x)	Synonym for acos()
acosh(x)	Real part of the arc hyperbolic cosine of x, e.g., acosh(.5) returns 0, not 1.0472i
asin(x)	Real part of the arc sine of x, e.g., asin(-5) returns -1.57080, not -1.57080+2.29243i
arcsin(x)	Synonym for asin()
asinh(x)	Arc hyperbolic sine
atan(x)	Arc tangent of x
arctan(x)	Synonym for atan()
atan2(y,x)	Four quadrant arc tangent of y/x
atanh(x)	Arc hyperbolic tangent
buf(x)	1 if x > .5, else 0
cbrt(x)	Cube root of (x)
ceil(x)	Integer equal or greater than x
cos(x)	Cosine of x
cosh(x)	Hyperbolic cosine of x
exp(x)	e to the x
fabs(x)	Same as abs(x)
flat(x)	Random number between -x and x with uniform distribution
floor(x)	Integer equal to or less than x
gauss(x)	Random number from Gaussian distribution with sigma of x.
hypot(x,y)	sqrt(x**2 + y**2)
if(x,y,z)	If x > .5, then y else z
int(x)	Convert x to integer
inv(x)	0. if x > .5, else 1.
limit(x,y,z)	Intermediate value of x, y, and z
ln(x)	Natural logarithm of x
log(x)	Alternate syntax for ln()
log10(x)	Base 10 logarithm
max(x,y)	The greater of x or y
mc(x,y)	A random number between x*(1+y) and x*(1-y) with uniform distribution.
min(x,y)	The smaller of x or y
pow(x,y)	Real part of x**y, e.g., pow(-.5,1.5) returns 0., not 0.353553i
pwr(x,y)	abs(x)**y
pwrs(x,y)	sgn(x)*abs(x)**y
rand(x)	Random number between 0 and 1 depending on the integer value of x.
random(x)	Similar to rand(), but smoothly transitions between values.
round(x)	Nearest integer to x
sgn(x)	Sign of x
sin(x)	Sine of x
sinh(x)	Hyperbolic sine of x
sqrt(x)	Real part of the square root of x, e.g., sqrt(-1) returns 0, not 0.707107i
table(x,a,b,c,d,...)	Interpolate a value for x based on a look up table given as a set of pairs of points.
tan(x)	Tangent of x.
tanh(x)	Hyperbolic tangent of x
u(x)	Unit step, i.e., 1 if x > 0., else 0.
uramp(x)	x if x > 0., else 0.

.TRAN - Realiza análisis transitorio no lineal

This is the most direct simulation of a circuit. It basically computes what happens when the circuit is powered up. Test signals are often applied as independent sources.

```
Syntax: .TRAN <Tstep> <Tstop> [Tstart [dTmax]] [modifiers]
        .TRAN <Tstop> [modifiers]
```

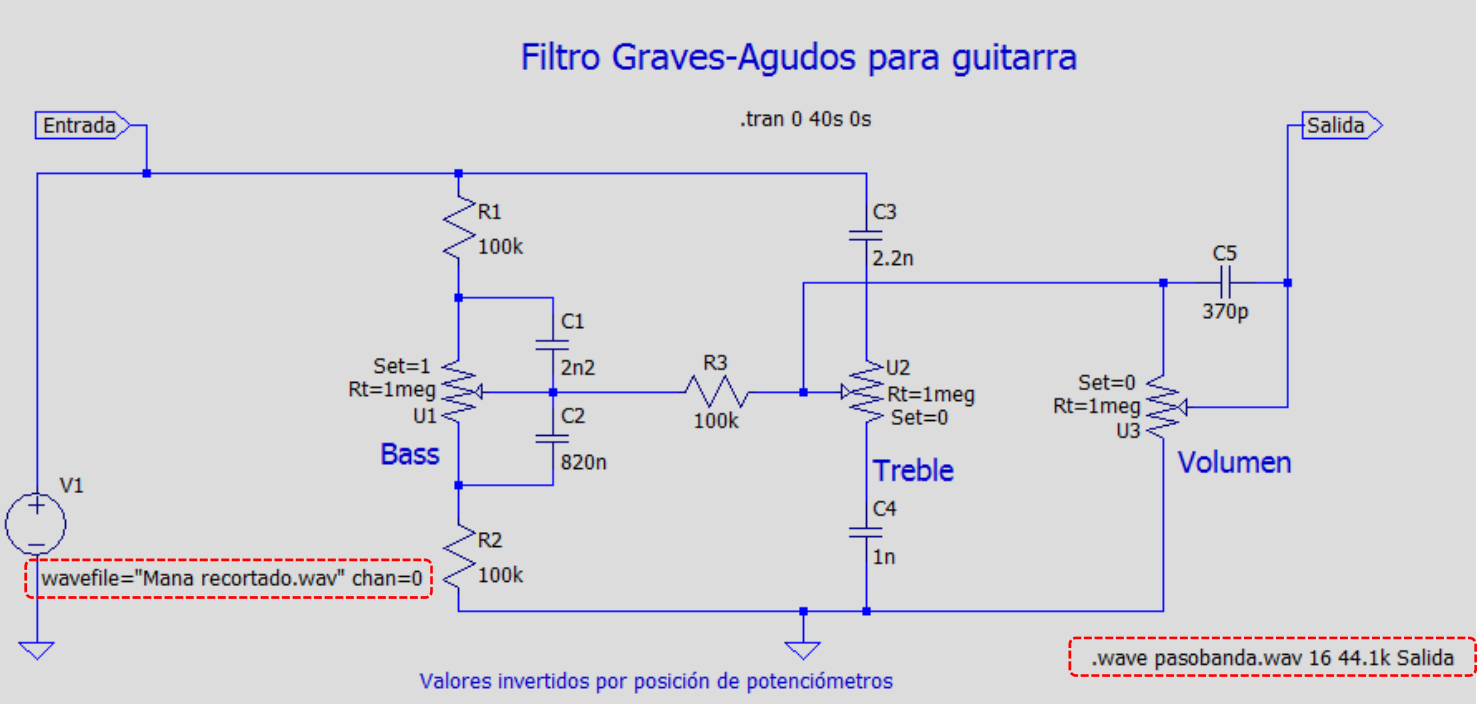
Tipos de análisis

.TRAN Modifiers

UIC: (Use Initial Conditions) Utiliza para el punto de operación en c.c. las condiciones iniciales fijadas por el usuario.
steady: Finaliza la simulación cuando se alcance el estado estable (*Finalice el período transistorio*).
nodiscard: No elimina el período transitorio simulado antes de que se alcanzase el estado estable.
startup: Resuelve el punto inicial de operación con las fuentes de tensión independientes y fuentes de corrientes desconectadas. Después activa el análisis transitorio y activa estas fuentes los primeros 20us de la simulación.
step: Calcula la respuesta del circuito para diferentes saltos (steps).

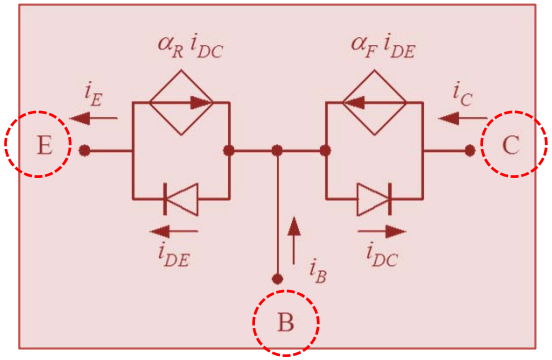
```
.WAVE -- Write Selected Nodes to a .Wav File

LTspice can write .wav audio files.  These files can then be listened to or be used as the input
of another simulation.

Syntax:  .wave <filename.wav> <Nbits>  <SampleRate> V(out) [V(out2) ...]
Example: .wave C:\output.wav 16 44.1K V(left) V(right)
```

La señal de entrada puede ser un fichero Wave



Elementos disponibles en un circuito

Component	Syntax
Special functions	Axx n1 n2 n3 n4 n5 n6 n7 n8 + <model> [extra parameters]
Arbitrary behavioral source	Bxx n+ n- <V=... or I=...>
Capacitor	Cxx n+ n- <capacitance> + [ic=<val.>] [Rser=<val.>] + [Lser=<val.>] [Rpar=<val.>] + [Cpar=<val.>] [m=<val.>]
Diode	Dxx A K <model> [area]
Voltage dependent voltage	Exx n+ n- nc+ nc- <gain>
Current dependent current	Fxx n+ n- <Vnam> <gain>
Voltage dependent current	Gxx n+ n- nc+ nc- <transcond.>
Current dependent voltage	Hxx n+ n- <Vnam> <transres.>
Independent current source	Ixx n+ n- <current>
JFET transistor	Jxx D G S <model> [area] [off] +[IC=<Vds,Vgs>] [temp=<T>]
Mutual inductance	Kxx L1 L2 L3... <coeff.>
Inductance	Lxx n+ n- <inductance> + [ic=<val.>] [Rser=<val.>] + [Rpar=<val.>] + [Cpar=<val.>] [m=<val.>]
MOSFET transistor	Mxx D G S B <model> [L=<len>] + [W=<width>] [AD=<area>] + [AS=<area>] [PD=<perim>] + [PS=<perim>] [NRD=<value>] + [NRS=<value>] [off] + [IC=<Vds, Vgs, Vbs> + [temp=<T>]
Lossy transmission line	Oxx L+ L- R+ R- <model>
Bipolar transistor	Qxx C B E [S] <model> [area] + [off] [IC=Vbe,Vce][temp=<T>]
Resistor	Rxx n1 n2 <value>
Voltage controlled switch	Sxx n1 n2 nc+ nc- <model> + [on,off]
Lossless transmission line	Txx L+ L- R+ R- ZO=<value> + TD=<value>
Uniform RC-line	Uxx n1 n2 ncommon <model> + L=<len> [N=<lumps>]
Independent voltage source	Vxx n+ n- <voltage>
Current controlled switch	Wxx n1 n2 <Vnam> <model> + [on,off]
Subcircuit	Xxx n1 n2 n3... <subckt name>
MESFET transistor	Zxx D G S model [area] [off] + [IC=<Vds,Vgs>]



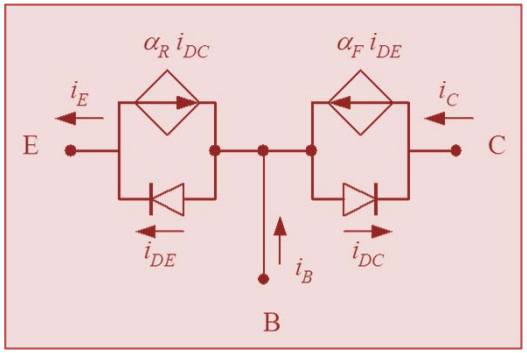
Modelo de Ebers-Moll

Jewell James Ebers y John Louis Moll

$$i_E = \underbrace{\frac{I_S}{\alpha_F}\left(e^{V_{BE}/V_T} - 1\right)}_{\text{Efecto transistor}} - \underbrace{I_S\left(e^{V_{BC}/V_T} - 1\right)}_{\text{Efecto diodo}}$$

$$i_C = I_S\left(e^{V_{BE}/V_T} - 1\right) - \frac{I_S}{\alpha_R}\left(e^{V_{BC}/V_T} - 1\right)$$

$$i_B = i_E - i_C$$



Modelo de Ebers-Moll

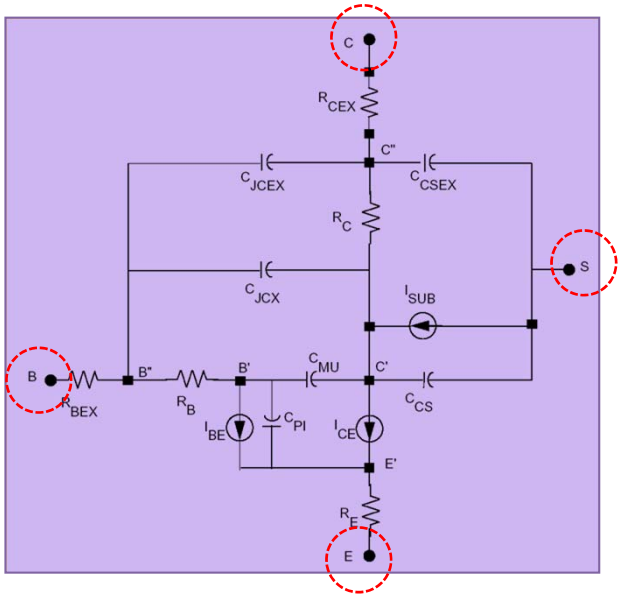
Jewell James Ebers y John Louis Moll

Efecto transistor | Efecto diodo

$$i_E = \frac{I_S}{\alpha_F}\left(e^{V_{BE}/V_T} - 1\right) - I_S\left(e^{V_{BC}/V_T} - 1\right)$$

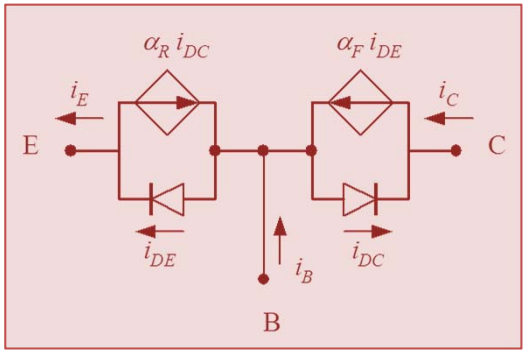
$$i_C = I_S\left(e^{V_{BE}/V_T} - 1\right) - \frac{I_S}{\alpha_R}\left(e^{V_{BC}/V_T} - 1\right)$$

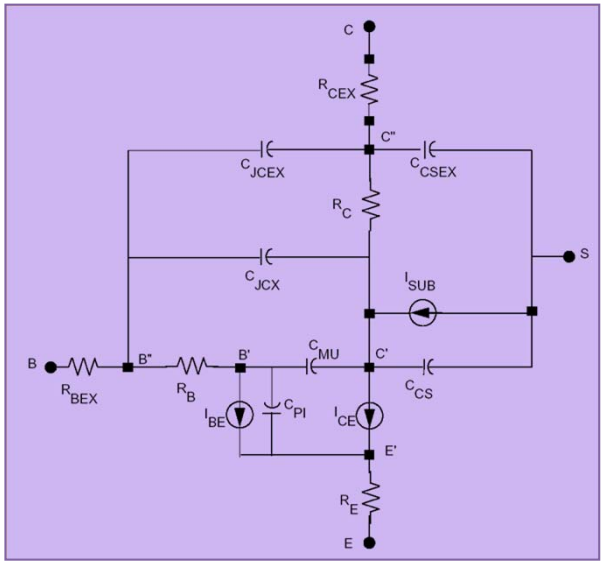
$$i_B = i_E - i_C$$



Modelo de Gummel-Poon

Hermann K. Gummel y Samuel H.C. Poon





Modelo de Gummel-Poon

Hermann K. Gummel y Samuel H.C. Poon

Modelo de Ebers-Moll

Jewell James Ebers y John Louis Moll

Efecto transistor | Efecto diodo

$$i_E = \frac{I_S}{\alpha_F}\left(e^{V_{BE}/V_T} - 1\right) - I_S\left(e^{V_{BC}/V_T} - 1\right)$$

$$i_C = I_S\left(e^{V_{BE}/V_T} - 1\right) - \frac{I_S}{\alpha_R}\left(e^{V_{BC}/V_T} - 1\right)$$

$$i_B = i_E - i_C$$

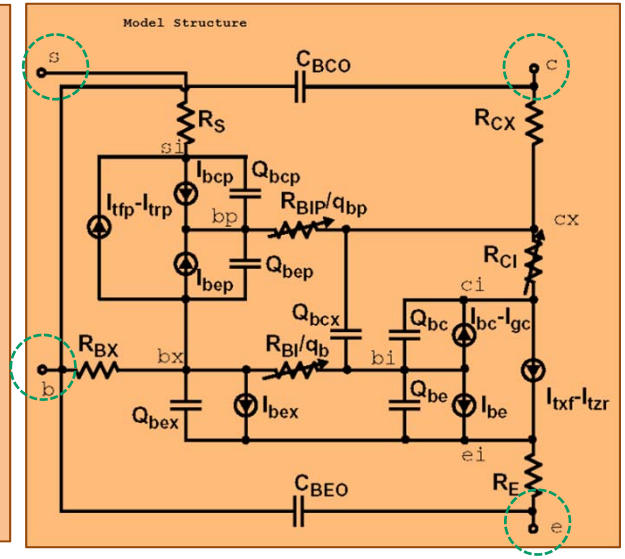
LTSpice utiliza 81 parámetros para definir el comportamiento de un BJT

VBIC - Vertical Bipolar Inter Company model is a generous contribution of source code from Dr.-Ing. **Dietmar Warning** of Danalyse GmbH, Berlin, Germany.

The VBIC model is a extended development of the **Standard Gummel-Poon** (SGP) model with the focus of integrated bipolar transistors in today's modern semiconductor technologies. With the implemented modified Quasi-Saturation model from **Kull** and **Nagel** it is also possible to model the special output characteristic of switching transistors. It is a widely used alternative to the SGP model for silicon, SiGe and III-V HBT devices.

VBIC Capabilities compared to SGP model:

- *Integrated Substrate transistor for parasitic devices in integrated processes.*
- *Weak avalanche and Base-emitter breakdown model.*
- *Improved Early Effect modeling.*
- *Physical separation of I_C and I_B.*
- *Improved Depletion capacitance model.*
- *Improved temperature modeling.*



Leading Character	Type of line
*	Comment
A	Special function device
B	Arbitrary behavioral source
C	Capacitor
D	Diode
E	Voltage dependent voltage source
F	Current dependent current source
G	Voltage dependent current source
H	Current dependent voltage source
I	Independent current source
J	JFET transistor
K	Mutual inductance
L	Inductor
M	MOSFET transistor
O	Lossy transmission line
Q	Bipolar transistor
R	Resistor
S	Voltage controlled switch
T	Lossless transmission line
U	Uniform RC-line
V	Independent voltage source
W	Current controlled switch
X	Subcircuit Invocation
Z	MESFET transistor
.	A simulation directive, For example: .options reltol=1e-4
+	A continuation of the previous line. The "+" is removed and the remainder of the line is considered part of the prior line.

The circuit to be analyzed is described by a text file called a netlist. The first line in the netlist is ignored, that is, it is assumed to be a comment. The last line of the netlist is usually simply the line ".END", but this can be omitted. Any lines after the line ".END" are ignored.

The order of the lines between the comment and end is irrelevant. Lines can be comments, circuit element declarations or simulation directives. Let's start with an example:

This first line is ignored
* The circuit below represents an RC circuit driven with a 1MHz square wave signal
R1 n1 n2 1K *; a 1KOhm resistor between nodes n1 and n2*
C1 n2 0 100p *; a 100pF capacitor between nodes n2 and ground*
V1 n1 0 PULSE(0 1 0 0 0 .5µ 1µ) *; a 1Mhz square wave*
.tran 3µ *; do a 3µs long transient analysis*
.end

The first two lines are comments. Any line starting with a "*" is a comment and is ignored. The line starting with "R1" declares that there is a 1K resistor connected between nodes n1 and n2. Note that the semicolon, ";", can be used to start a comment in the middle of a line. The line starting with "C1" declares that there is a 100pF capacitor between nodes n2 and ground. The node "0" is the global circuit common ground.

Leading Character	Type of line
*	Comment
A	Special function device
B	Arbitrary behavioral source
C	Capacitor
D	Diode
E	Voltage dependent voltage source
F	Current dependent current source
G	Voltage dependent current source
H	Current dependent voltage source
I	Independent current source
J	JFET transistor
K	Mutual inductance
L	Inductor
M	MOSFET transistor
O	Lossy transmission line
Q	Bipolar transistor
R	Resistor
S	Voltage controlled switch
T	Lossless transmission line
U	Uniform RC-line
V	Independent voltage source
W	Current controlled switch
X	Subcircuit Invocation
Z	MESFET transistor
.	A simulation directive, For example: .options reltol=1e-4
+	A continuation of the previous line. The "+" is removed and the remainder of the line is considered part of the prior line.

The circuit to be analyzed is described by a text file called a netlist. The first line in the netlist is ignored, that is, it is assumed to be a comment. The last line of the netlist is usually simply the line ".END", but this can be omitted. Any lines after the line ".END" are ignored.

The order of the lines between the comment and end is irrelevant. Lines can be comments, circuit element declarations or simulation directives. Let's start with an example:

This first line is ignored
* The circuit below represents an RC circuit driven with a 1MHz square wave signal
R1 n1 n2 1K *; a 1KOhm resistor between nodes n1 and n2*
C1 n2 0 100p *; a 100pF capacitor between nodes n2 and ground*
V1 n1 0 PULSE(0 1 0 0 0 .5µ 1µ) *; a 1Mhz square wave*
.tran 3µ *; do a 3µs long transient analysis*
.end

The first two lines are comments. Any line starting with a "*" is a comment and is ignored. The line starting with "R1" declares that there is a 1K resistor connected between nodes n1 and n2. Note that the semicolon, ";", can be used to start a comment in the middle of a line. The line starting with "C1" declares that there is a 100pF capacitor between nodes n2 and ground. The node "0" is the global circuit common ground.

Netlist

```
Q1 Salida N002 0 0 2N2222
.model NPN NPN
.lib C:\Program Files (x86)\LTC\LTspiceIV\lib\cmp\standard.bjt
```

Definición de funcionamiento

```
.model 2N2222 NPN(IS=1E-14 VAF=100
+  BF=200 IKF=0.3 XTB=1.5 BR=3
+  CJC=8E-12 CJE=25E-12 TR=100E-9 TF=400E-12
+  ITF=1 VTF=2 XTF=3 RB=10 RC=.3 RE=.2 Vceo=30 Icrating=800m  mfg=Philips)
```

Bipolar transistor	Qxx C B E [S] <model> [area] + [off] [IC=Vbe,Vce] [temp=<T>]

CONVERGENCE

A netlist that doesn't simulate isn't converging numerically. *Assuming the circuit contains no connection errors*, there are basically three parameters that can be adjusted to help convergence: **ABSTOL, VNTOL, and RELTOL**.

ABSTOL is the absolute current tolerance. Its default value is 1pA. This means that when a simulated circuit gets within 1pA of its "actual" value, SPICE assumes that the current has converged and moves onto the next time step or AC/DC value. **VNTOL** is the node voltage tolerance, default value of 1μV. RELTOL is the relative tolerance parameter, default value of 0.001 (0.1 percent). **RELTOL** is used to avoid problems with simulating large and small electrical values in the same circuit. For example, suppose the default value of RELTOL and VNTOL were used in a simulation where the actual node voltage is 1V. The RELTOL parameter would signify an end to the simulation when the node voltage was within 1mV of 1V (1V·RELTOL), while the VNTOL parameter signifies an end when the node voltage is within 1μV of 1V. SPICE uses the larger of the two, in this case the RELTOL parameter results, to signify that the node has converged.

Increasing the value of these three parameters helps speed up the simulation and assists with convergence problems at the price of reduced accuracy. To help with convergence, the following statement can be added to a SPICE netlist:
.OPTIONS ABSTOL=1uA VNTOL=1mV RELTOL=0.01

To (hopefully) force convergence, these values can be increased to
.OPTIONS ABSTOL=1mA VNTOL=100mV RELTOL=0.1

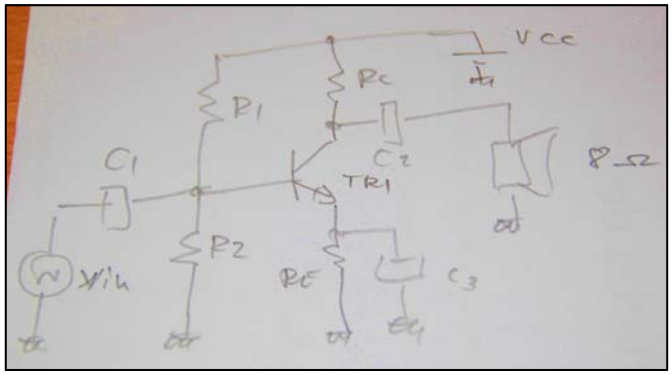
Note that in some high-gain circuits with feedback (like the op-amp's designed later in the book) decreasing these values can actually help convergence.

SOME COMMON MISTAKES AND HELPFUL TECHNIQUES

The following is a list helpful techniques for simulating circuits using SPICE.

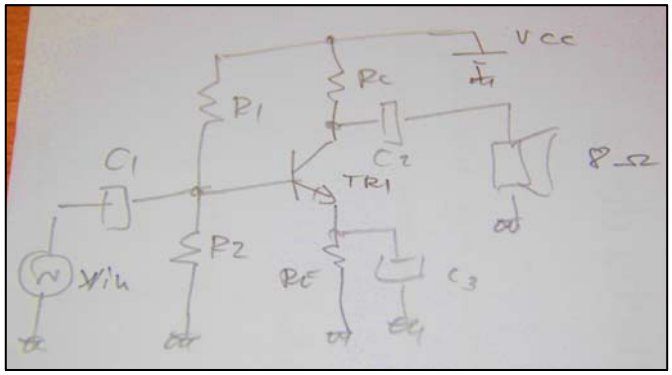
1. The first line in a SPICE netlist must be a comment line. SPICE ignores the first line in a netlist file.
2. One megaohm is specified using 1MEG, not 1M, 1m, or 1 MEG.
3. One farad is specified by 1, not 1f or 1F. 1F means one femto-farad or 10^{-15} farads.
4. Voltage source names should always be specified with a first letter of V. Current source names should always start with an I.
5. Transient simulations display time data; that is, the x-axis is time. A jagged plot such as a sinewave that looks like a triangle wave or is simply not smooth is the result of not specifying a maximum print step size.
6. Convergence with a transient simulation can usually be helped by adding a UIC (use initial conditions) to the end of a .TRAN statement.
7. A simulation using MOSFETs must include the scale factor in a .OPTIONS statement unless the widths and lengths are specified with the actual (final) sizes.
8. In general, the body connection of a PMOS device is connected to VDD , and the body connection of an n-channel MOSFET is connected to ground. This is easily checked in the SPICE netlist.
9. Convergence in a DC sweep can often be helped by avoiding the power supply boundaries. For example, sweeping a circuit from 0 to 1 V may not converge, but sweeping from 0.05 to 0.95 will.
10. In any simulation adding .OPTIONS RSHUNT=1E8 (or some other value of resistor) can be used to help convergence. This statement adds a resistor in parallel with every node in the circuit (see the WinSPICE manual for information concerning the GMIN parameter). Using a value too small affects the simulation results.

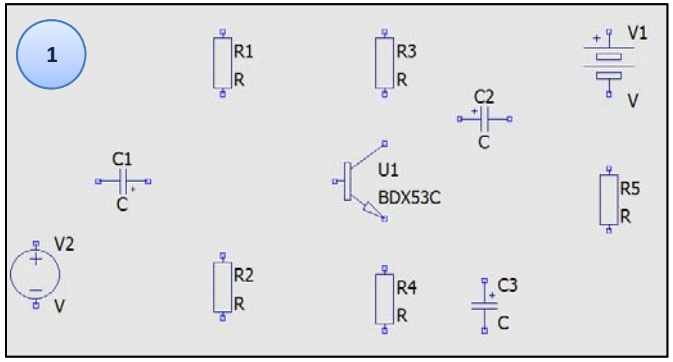
Pasos a seguir para realizar el primer esquemático



Partimos de la *idea feliz*

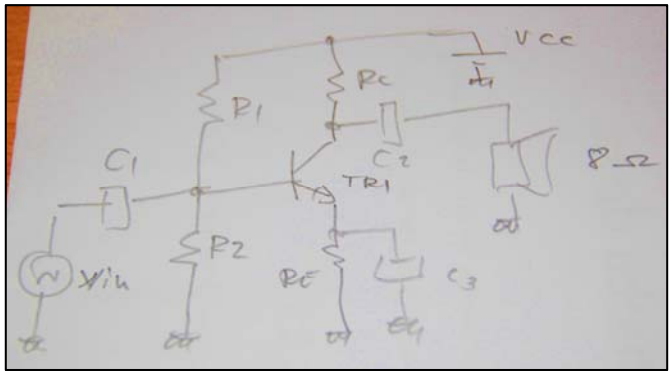
1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).





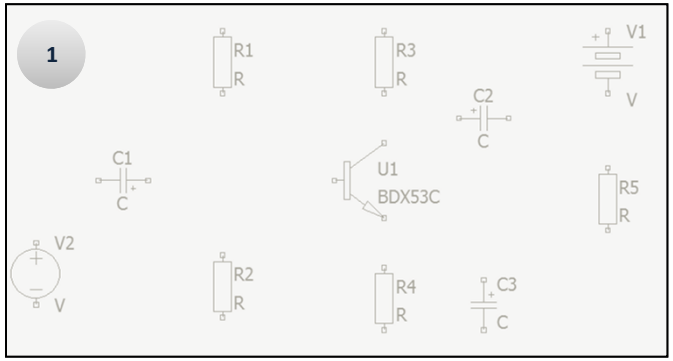
Partimos de la *idea feliz*

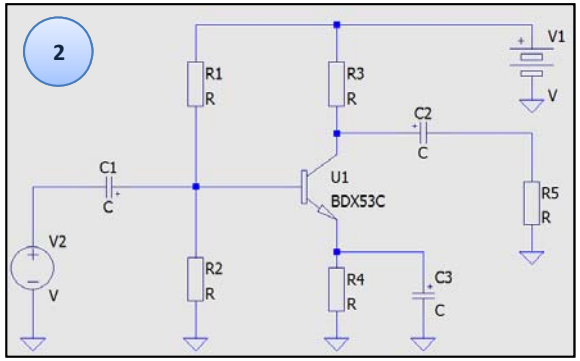
1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).

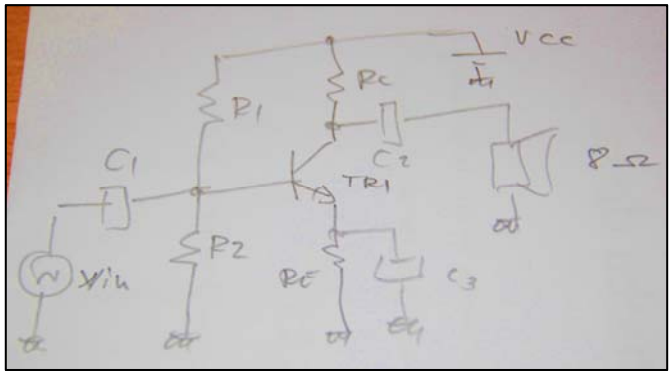


Partimos de la *idea feliz*

1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).

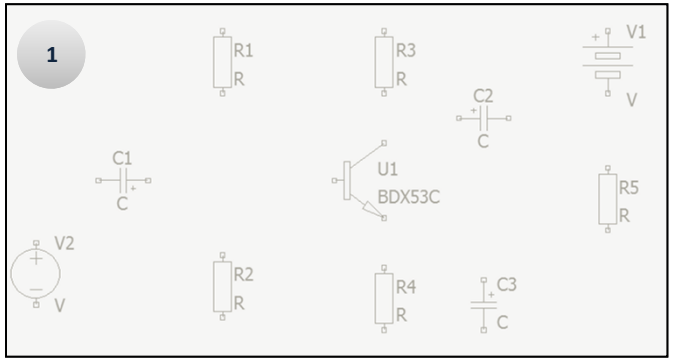


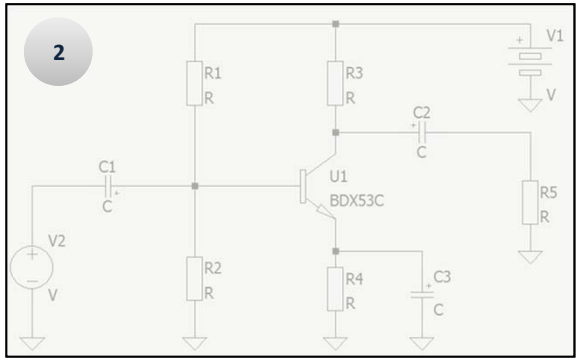


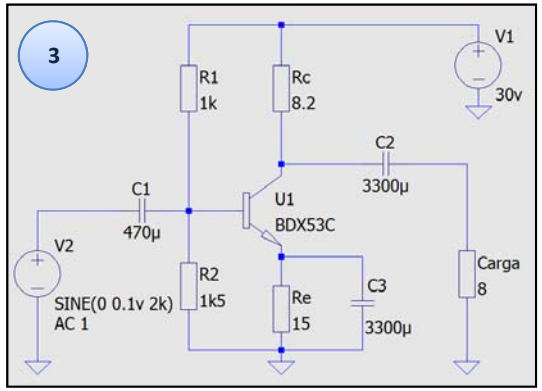


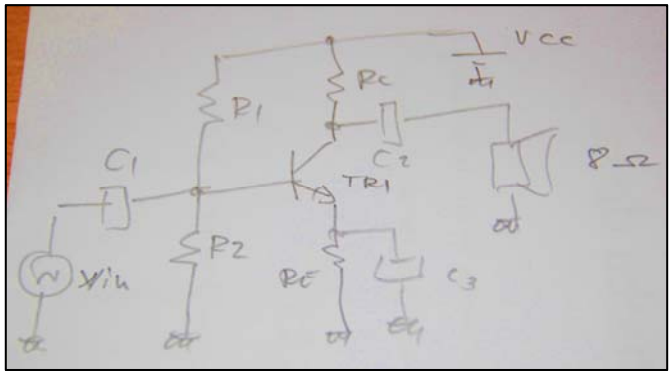
Partimos de la *idea feliz*

1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).



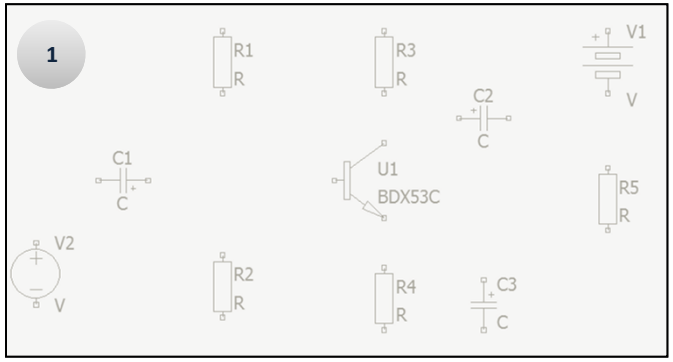


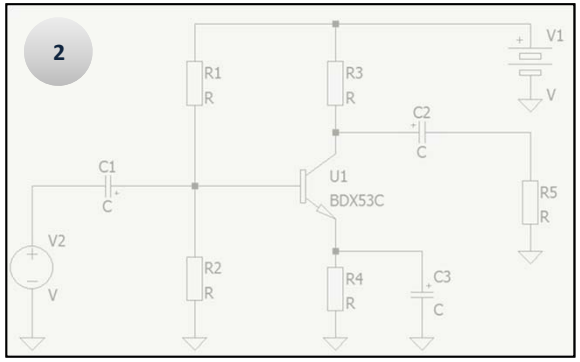


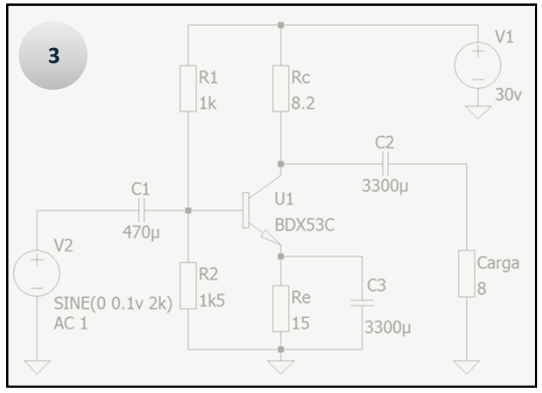


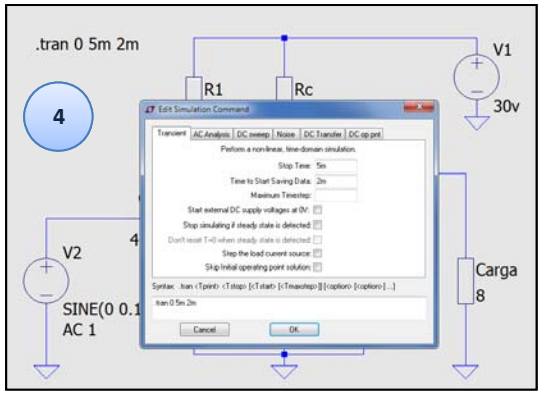
Partimos de la *idea feliz*

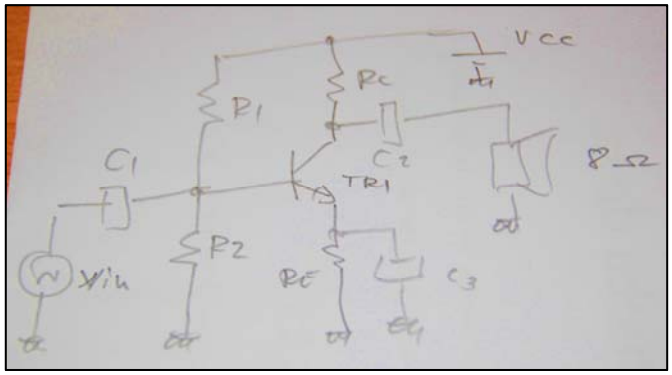
1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).





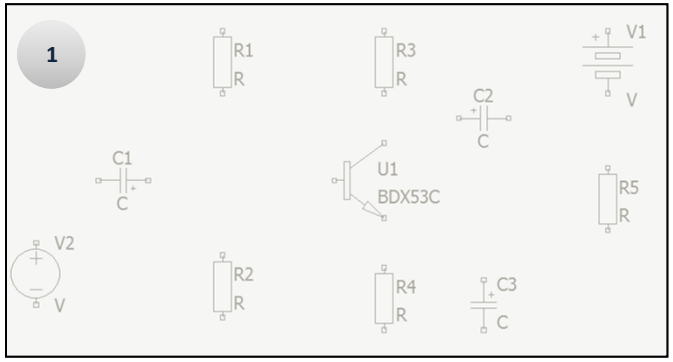


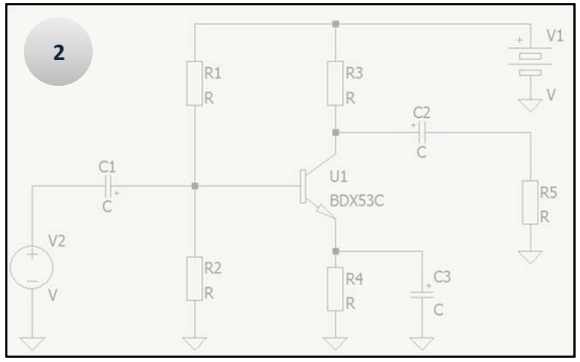


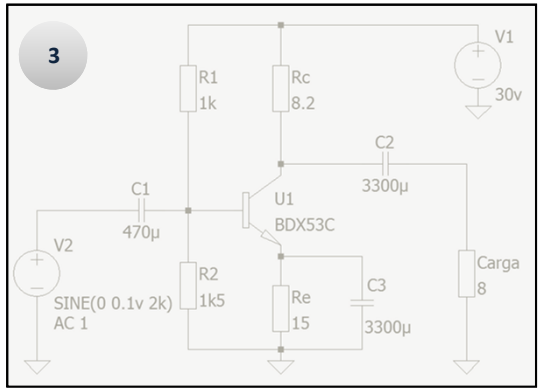


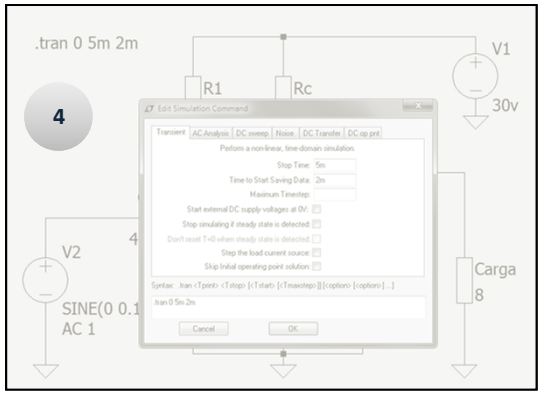
Partimos de la *idea feliz*

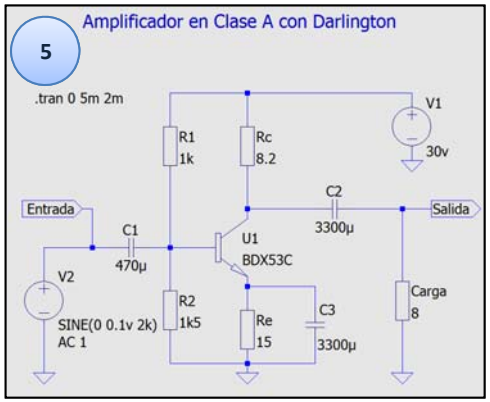
1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).



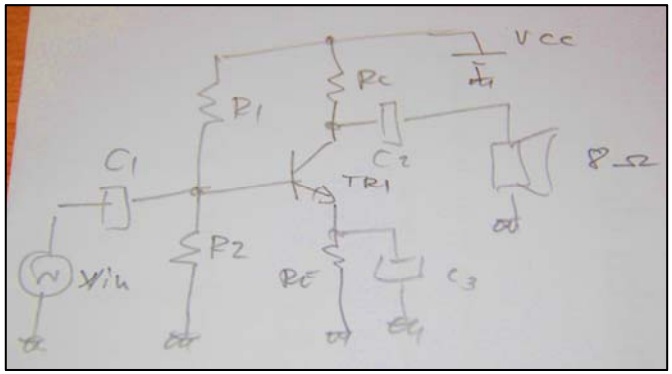






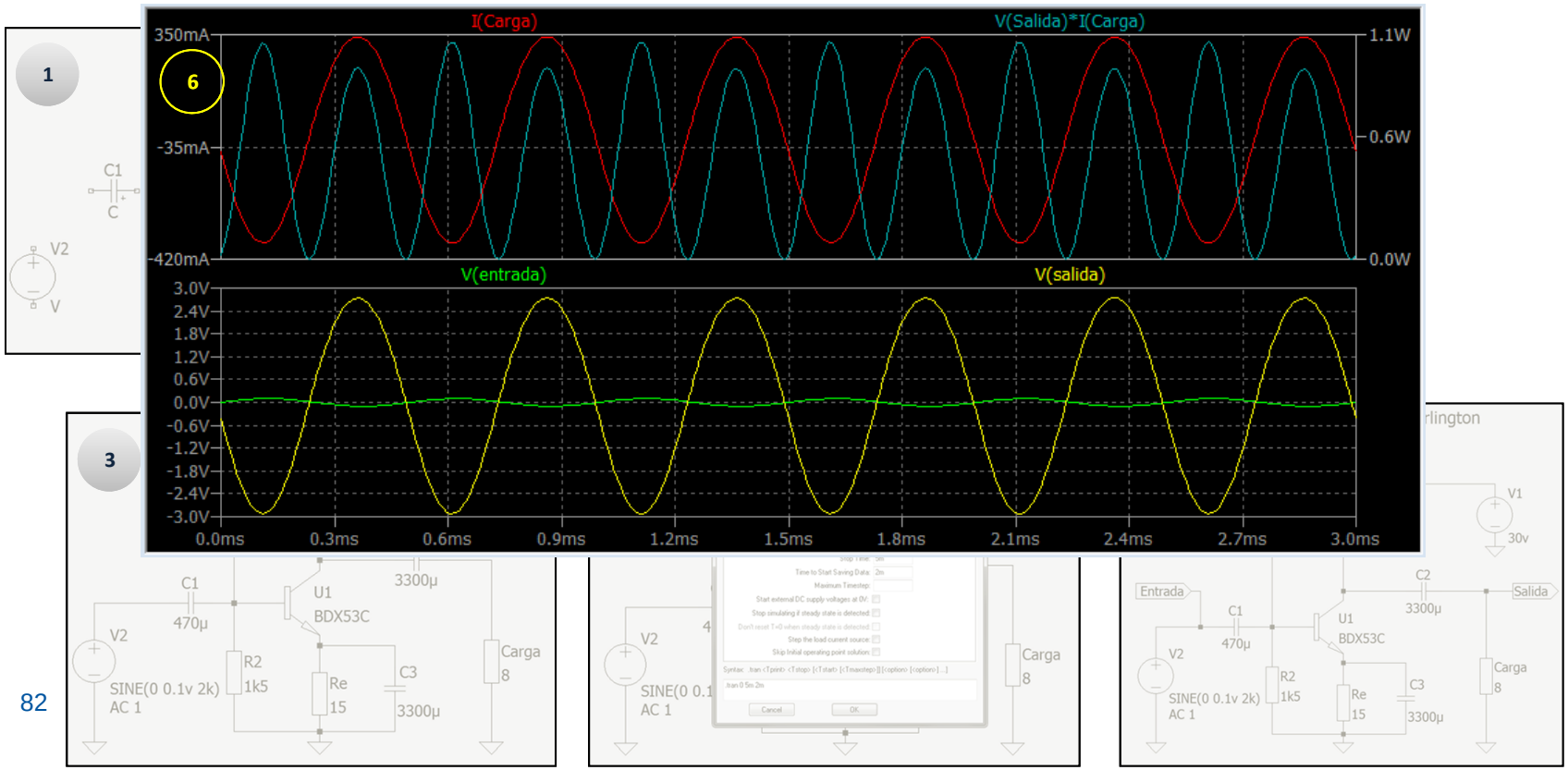


Pasos a seguir para realizar el primer esquemático



Partimos de la *idea feliz*

1. Colocamos los componentes.
2. Realizamos conexiones de los componentes.
3. Fijamos valores de los componentes.
4. Seleccionar tipo de simulación y características.
5. Añadimos etiquetas para mejorar la visualización.
6. Visualizar datos/curvas/gráficas de interés (V, I, P…).



T = tera = 10^{12}
G = giga = 10^{9}
MEG = meg = 10^{6}
K = kilo = 10^{3}
M = milli = 10^{-3}
U = micro = 10^{-6}
N = nano = 10^{-9}
P = pico = 10^{-12}
F = femto = 10^{-15}

LTspice leading characters

- A special functions device
- B arbitrary behavioral source
- C capacitor
- D diode
- E voltage dependent voltage source
- F current dependent current source
- G voltage dependent current source
- H current dependent voltage source
- I independent current source
- J JFET transistor
- K mutual inductance
- L inductor
- M MOSFET transistor
- O lossy transmission line
- Q bipolar transistor
- R resistor
- S voltage controlled switch
- T lossless transmission line
- U uniform RC-line
- V independent voltage source
- W current controlled switch
- X subcircuit invocation
- Z MESFET transistor
- * comment
- + continuation of prior line
- . simulation directive

Información relacionada

http://www.intusoft.com/models.htm#freemodels
http://www.penzar.com/links.htm
http://ee.cleversoul.com/simulation.html
http://cmosedu.com/
http://www.simonbramble.co.uk/lt_spice/ltspice_lt_spice.htm
http://www.onsemi.com/site/support/models
http://www.coilcraft.com/modelsswcad.cfm
http://tech.groups.yahoo.com/group/LTspice/
http://ltspice.linear.com/software/scad3.pdf

BIBLIOGRAFÍA

- LTspice IV (SwitcherCAD III) www.linear.com.
- Engelhardt, M., Using Transformers in LTspice/SwitcherCAD III. Linear Technology Magazine, September 2006.
- http://k6jca.blogspot.com.es/2012/07/monte-carlo-and-worst-case-circuit.html.
- http://ltwiki.org/index.php?title=Main_Page.
- John L. Moll (images.frompo.com). Jewell J. Ebers http://eds.ieee.org/jj-ebers-award.html.

Fin