Nutmeg Help

Robert Zeff
2413 Kirkwood Ave.
Modesto California, 95350
Phone:(209) 577-4268, ext. 101
FAX: (209) 577-8548, 522-5866
CompuServe: 70323,1251

R0107932119

Tasks

Zoom
Plot Functions

Zoom

To zoom in, place the mouse cursor on a corner of the desired zoom area. Press and hold the left mouse button, drag the mouse to the adjacent corner of the desired zoom area and release the mouse button.

To zoom out, double click the right mouse button anywhere in the plot area. A single click of the right mouse button sometimes causes an infinite zoom, garbling the display area. A single click of the left mouse button will return the plot to normal zoom. Also see the menu command, Axis

Registration

I am an electronics engineer who couldn't pay the price (>$8,000) of a commercial Windows 16 bit Spice program, so I re-wrote UC Berkeley's Spice for Windows NT(tm). Due to the 32 bit platform, this Spice is faster and more powerful than other Windows Spice programs. So... if you use Spice and appreciate my work, please send $275.00 to the address below. I'll send you my latest revision, whatever device models I've accumulated, and the original Berkeley source code and documentation (>10 mb!). A more complete help system will also be included.

Educational institutions may use the program for free.

Others who do not wish to pay must discontinue use after a 30 day trial period.

You may not, in any case, sell this program to others.

Note:

The shareware message that appears on the display unfortunately causes an increase in plot times. This message also increases the size of the print image and consequently slows the printing process. Registered programs will not have this nuisance.

Address:

Robert Zeff
2413 Kirkwood Ave.
Modesto, California 95350

Phone: (209) 577-4268 ext. 101 (days)
(209)-521-1448 (Evenings)
Fax:: (209) 577-8548, (209) 522-5866
Compuserve: 70323,1251

Note: I am *not* selling Berkeley software, I am charging for my enhancements and support, per Berkeley's licensing agreement.

About Spice32

Spice32 is derived from UC Berkeley's Spice, version 3e2. Spice32 is the Windows NT port of Berkeley's Spice3e2. This 32 bit program has a circuit file as the input and a binary database as the output.

Nutmeg32 is derived from the Berkeley Nutmeg program, which is an interactive graphics program used to display the database created by Spice32.

Spice32 / Nutmeg32 is a general-purpose circuit simulation program for nonlinear dc, nonlinear transient, and ac analyses. Circuits may contain resistors, capacitors, inductors, mutual inductors, independent voltage and current sources, four types of dependent sources, lossless and lossy transmission lines, two switches, uniform distributed RC lines, and the five most common semiconductor devices: diodes, BJTs, JFETs, MESFETs, and MOSFETs.

The Berkeley SPICE3 version is based directly on SPICE 2G.6. While SPICE3 is being developed to include new features, it continues to support those capabilities and models which remain in extensive use in the SPICE3 program. SPICE3 has built-in models for the semiconductor devices, and the user need specify only the per model parameter values. The model for the BJT is based on the integral-charge model of Gummel-Poon; however, if the Gummel-Poon parameters are not specified, the model reduces to the simpler model. In either case, effects, ohmic resistance, and a output conductance may be included. The diode model can be used for either junction diodes or Schottky barrier diodes. The JFET model is based on the FET model of Shichman and Hodges. Six MOSFET models are implemented: MOS1 is described by a square-law I-V characteristic, MOS2[1] is an analytical model, MOS3[1] is a semi-empirical model; MOS6[2] is a simple analytic model, accurate in the short channel region; MOS4[3, 4] and MOS5[5] are the BSIM (Berkeley Short-channel IGFET) and BSIM2. MOS3, and MOS4 include second-order effects such as channel length modulation, subthreshold scattering-limited velocity saturation, small-size effects, and charge-controlled capacitances See Spice32

Spice32

Subcircuits

A subcircuit that consists of SPICE elements can be defined and in a fashion similar to device models. The subcircuit is defined in the input deck by a grouping of element lines; the program then automatically inserts the group of elements wherever the subcircuit is. There is no limit on the size or complexity of subcircuits, and subcircuits may contain other subcircuits.

1. .SUBCKT Line
2. .ENDS Line
3. Subcircuit Calls

3. Subcircuit Calls

General form:

XYYYYYYY N1 <N2 N3 ...> SUBNAM

Example:

```
X1  2  4  17  3  1  MULTI
```

Subcircuits are used in SPICE by specifying pseudo-elements beginning with the letter X, followed by the circuit nodes to be used in expanding the subcircuit.

2. .ENDS Line

General form:

.ENDS <SUBNAM>

Examples:

```
.ENDS OPAMP
```

This line must be the last one for any subcircuit definition. The subcircuit name, if included, indicates which subcircuit definition is being terminated; if omitted, all subcircuits being defined are terminated. The name is needed only when nested subcircuit definitions are being made.

1 .SUBCKT Line

General form:

.SUBCKT subnam N1 <N2 N3 ...>

Example:

```
.SUBCKT OPAMP 1 2 3 4
```

A circuit definition is begun with a .SUBCKT line. SUBNAM is the subcircuit name, and N1, N2, ... are the external nodes, which cannot be zero. The group of element lines which immediately follow the .SUBCKT line define the subcircuit. The last line in a subcircuit definition is the .ENDS line (see below). Control lines may not appear within a subcircuit definition; however, subcircuit definitions may contain anything else, including other subcircuit definitions, device models, and subcircuit calls (see below). Note that any device models or subcircuit definitions included as part of a subcircuit definition are strictly local (i.e., such models and definitions are not known outside the subcircuit definition). Also, any element nodes not included on the .SUBCKT line are strictly local, with the exception of 0 (ground) which is always global.

Control lines

1. .OPTIONS Line
2. .OP Line
3. .DC Line
4. .NODESET Line
5. .IC Line
6. .TF Line
7. .AC Line
8. .DISTO Line
9. .NOISE Line
10. .TRAN Line
11. .PZ Line

11. .PZ Line

General form:

.PZ NODE1 NODE2 NODE3 NODE4 CUR POL
.PZ NODE1 NODE2 NODE3 NODE4 CUR ZER
.PZ NODE1 NODE2 NODE3 NODE4 CUR PZ
.PZ NODE1 NODE2 NODE3 NODE4 VOL POL
.PZ NODE1 NODE2 NODE3 NODE4 VOL ZER
.PZ NODE1 NODE2 NODE3 NODE4 VOL PZ

Examples:

```
.PZ 1 0 3 0 CUR POL
.PZ 2 3 5 0 VOL ZER
.PZ 4 1 4 1 CUR PZ
```

CUR stands for a transfer function of the type (output voltage)/(input current) while VOL stands for a transfer function of the type (output voltage)/(input voltage). POL stands for pole analysis only, ZER for zero analysis only and PZ for both. This feature is provided mainly because if there is a nonconvergence in finding poles or zeros, then, at least the other can be found. Finally, NODE1 and NODE2 are the two input nodes and NODE3 and NODE4 are the two output nodes. Thus, there is complete freedom regarding the output and input ports and the type of transfer function. In interactive mode, the command syntax is the same except that the first field is PZ instead of .PZ. To print the results, one should use the command 'print all'.

10. .TRAN Line

General form:

.TRAN TSTEP TSTOP <TSTART <TMAX>>

Examples:

```
.TRAN 1NS 100NS
.TRAN 1NS 1000NS 500NS
.TRAN 10NS 1US
```

TSTEP is the printing or plotting increment for line-printer output. For use with the post-processor, TSTEP is the suggested computing increment. TSTOP is the final time, and TSTART is the initial time. If TSTART is omitted, it is assumed to be zero. The transient analysis always begins at time zero. In the interval <zero, TSTART>, the circuit is analyzed (to reach a steady state), but no outputs are stored. In the interval <TSTART, TSTOP>, the circuit is analyzed and outputs are stored. TMAX is the maximum step size that SPICE uses; for default, the program chooses either TSTEP or (TSTOP-TSTART)/50.0, whichever is smaller. TMAX is useful when one wishes to guarantee a computing interval which is smaller than the printer increment, TSTEP. UIC (use initial conditions) is an optional keyword which indicates that the user does not want SPICE to solve for the quiescent operating point before beginning the transient analysis. If this keyword is specified, SPICE uses the values specified using IC=... on the various elements as the initial transient condition and proceeds with the analysis. If the .IC control line has been specified, then the node voltages on the .IC line are used to compute the initial conditions for the devices. Look at the description on the .IC control line for its interpretation when UIC is not specified.

8. .DISTO Line

General form:

.DISTO DEC ND FSTART FSTOP <F2OVERF1>
.DISTO OCT NO FSTART FSTOP <F2OVERF1>
.DISTO LIN NP FSTART FSTOP <F2OVERF1>

Examples:

```
.DISTO DEC 10 1kHz 100Mhz
.DISTO DEC 10 1kHz 100Mhz 0.9
```

This line does a small-signal distortion analysis of the circuit. A multi-dimensional Volterra series analysis is done using multi-dimensional Taylor series to represent the nonlinearites at the operating point. Terms of upto third order are used in the series expansions. If the optional parameter F2OVERF1 is not specified, .DISTO does a harmonic analysis - i.e., it analyses distortion in the circuit using only a single input frequency F1, which is swept as specified by arguments of the .DISTO command exactly as in the .AC command. Inputs at this frequency may be present at more than one input source, and their magnitudes and phases are specified by the arguments of the DISTOF1 keyword in the input file lines for the input sources (see the description for independent sources). (The arguments of the DISTOF2 keyword are not relevant in this case). The analysis produces information about the A.C. values of all node voltages and branch currents at the harmonic frequencies 2F1 and 3F1, vs. the input frequency F1 as it is swept. (A value of 1 (as a complex distortion output) signifies cos(2 pi (2F1)) at 2F1 and cos(2 pi (3F1)) at 3F1, using the convention that 1 at the input fundamental frequency is equivalent to cos(2 pi F1)) The distortion component desired (2F1 or 3F1) can be selected using commands in nutmeg, and then printed or plotted. (Normally, one is interested primarily in the magnitude of the harmonic components, so the magnitude of the AC distortion value is looked at). It should be noted that these are the A.C. values of the actual harmonic components, and are not equal to HD2 and HD3. To obtain HD2 and HD3, one must divide by the corresponding A.C. values at F1, obtained from an .AC line. This division can be done using nutmeg commands. If the optional F2OVERF1 parameter is specified, it should be a real number between (and not equal to) 0.0 and 1.0; in this case, .DISTO does a spectral analysis. It considers the circuit with sinusoidal inputs at two different frequencies F1 and F2. F1 is swept according to the .DISTO control line options exactly as in the .AC control line. F2 is kept fixed at a single frequency as F1 sweeps the value at which it is kept fixed is equal to F2OVERF1 times FSTART. Each independent source in the circuit may potentially have two (superimposed) sinusoidal inputs for distortion, at the frequencies F1 and F2. The magnitude and phase of the F1 component are specified by the arguments of the DISTOF1 keyword in the source's input line (see the description of independent sources); the magnitude and phase of the F2 component are specified by the arguments of the DISTOF2 keyword. The analysis produces plots of all node voltages/branch currents at the intermodulation product frequencies F1 + F2, F1 - F2, and (2 F1) - F2, vs the swept frequency F1. The IM product of interest may be selected using the setplot command, and displayed with the print and plot commands. It is to be noted as in the harmonic analysis case, the results are the actual AC voltages and currents at the intermodulation frequencies, and need to be normalized with respect to .AC values to obtain the IM parameters. If the DISTOF1 or DISTOF2 keywords are missing from the description of an independent source, then that source is assumed to have no input at the corresponding frequency. The default values of the magnitude and phase are 1.0 and 0.0 respectively. The phase should be specified in degrees. It should be carefully noted that the number F2OVERF1 should ideally be an irrational number, and that since this is not possible in practice, efforts should be made to keep the denominator in its fractional representation as large as possible, certainly above 3, for accurate results (i.e., if F2OVERF1 is represented as a fraction A/B, where A and B are integers with no common factors, B should be as large as possible; note that A < B because F2OVERF1 is constrained to be < 1). To illustrate why, consider the cases where F2OVERF1 is 49/100 and 1/2. In a spectral analysis, the outputs produced are at F1 + F2, F1 - F2 and 2 F1 - F2. In the latter case, F1 - F2 = F2, so the result at the F1-F2 component is erroneous because there is the strong fundamental F2 component at the same frequency. Also, F1 + F2 = 2 F1 - F2 in the latter case, and each result is erroneous individually. This problem is not there in the case where F2OVERF1 = 49/100, because F1-F2 = 51/100 F1 < > 49/100 F1 = F2. In this case, there are two very closely spaced frequency components at F2 and F1 - F2. One of the advantages of the Volterra series technique is that it computes distortions at mix frequencies expressed symbolically (i.e. n F1 \ F2), therefore one is able to obtain the strengths of distortion components accurately even if the separation between them is very small, as opposed to transient analysis for example. The disadvantage is of course that if two of the mix frequencies coincide, the results are not merged

together and presented (though this could presumably be done as a postprocessing step). Currently, the interested user should keep track of the mix frequencies himself or herself and add the distortions at coinciding mix frequencies together should it be necessary.

9. .NOISE Line

General form:

.NOISE V(OUTPUT <,REF>) SRC (DEC | LIN | OCT) PTS FSTART FSTOP
+ <PTS_PER_SUMMARY>

Examples:

```
.NOISE V(5) VIN DEC 10 1kHZ 100Mhz
.NOISE V(5,3) V1 OCT 8 1.0 1.0e6 1
```

This line does a noise analysis of the circuit. OUTPUT is the node at which the total output noise is desired; if REF is specified, then the noise voltage V(OUTPUT) - V(REF) is calculated. By default, REF is assumed to be ground. SRC is the name of an independent source to which input noise is referred. PTS, FSTART and FSTOP are .AC type parameters that specify the frequency range over which plots are desired. PTS_PER_SUMMARY is an optional integer; if specified, the noise contributions of each noise generator is produced every PTS_PER_SUMMARY frequency points. The .NOISE control line produces two plots - one for the Noise Spectral Density curves and one for the total Integrated Noise over the specified frequency range. All noise voltages/currents are in squared units (V^2 and A^2 for spectral density, V^2 and A^2 for integrated noise).

6. .TF Line

General form:

.TF OUTVAR INSRC

Examples:

```
.TF V(5, 3) VIN
.TF I(VLOAD) VIN
```

This line defines the small-signal output and input for the dc small- signal analysis. OUTVAR is the small-signal output variable and INSRC is the small-signal input source. If this line is included, SPICE computes the dc small-signal value of the transfer function (output/input), input resistance, and output resistance. For the first example, SPICE would compute the ratio of V(5, 3) to VIN, the small-signal input resistance at VIN, and the small-signal output resistance measured across nodes 5 and 3.

5. .IC Line

General form:

.IC V(NODNUM)=VAL V(NODNUM)=VAL ...

Example:

```
.IC V(11)=5 V(4)=-5 V(2)=2.2
```

This line is for setting transient initial conditions. It has two different interpretations, depending on whether the UIC parameter is specified on the .TRAN control line. Also, one should not confuse this line with the .NODESET line. The .NODESET line is only to help dc convergence, and does not affect final bias solution (except for multi-stable circuits). The two interpretations of this line are as follows:

1. When the UIC parameter is specified on the .TRAN line, then the node voltages specified on the .IC control line are used to compute the capacitor, diode, BJT, JFET, and MOSFET initial conditions. This is equivalent to specifying the IC=... parameter on each device line, but is much more convenient. The IC=... parameter can still be specified and takes over the .IC values. Since no dc bias (initial transient) solution is computed before the transient analysis, one should take care to specify all dc source voltages on the .IC control line if they are to be used to compute device initial conditions.
2. When the UIC parameter is not specified on the .TRAN control line, the dc bias (initial transient) solution is computed before the transient analysis. In this case, the node voltages specified on the .IC control line is forced to the desired initial values during the bias solution. During transient analysis, the constraint on these node voltages is removed. This is the preferred method since it allows SPICE to compute a consistent dc solution.

4. .NODESET Line

General form:

.NODESET V(NODNUM)=VAL V(NODNUM)=VAL ...

Example

```
.NODESET V(12)=4.5 V(4)=2.23
```

This line helps the program find the dc or initial transient solution by making a preliminary pass with the specified nodes held to the given voltages. The restriction is then released and the iteration continues to the true solution. The .NODESET line may be necessary for convergence on bistable or astable circuits. In general, this line should not be necessary.

3. .DC Line

General form:

.DC SRCNAM VSTART VSTOP VINCR [SRC2 START2 STOP2 INCR2]

Examples:

```
.DC VIN 0.25 5.0 0.25
.DC VDS 0 10 .5 VGS 0 5 1
.DC VCE 0 10 .25 IB 0 10U 1U
```

This line defines the dc transfer curve source and sweep limits. SRCNAM is the name of an independent voltage or current source. VSTART, VSTOP, and VINCR are the starting, final, and incrementing values respectively. The first example causes the value of the voltage source VIN to be swept from 0.25 Volts to 5.0 Volts in increments of 0.25 Volts. A second source (SRC2) may optionally be specified with associated sweep parameters. In this case, the first source is swept over its range for each value of the second source. This option can be useful for obtaining semiconductor device output characteristics. See the second example circuit description

2. .OP Line

General form:

.OP

The inclusion of this line in an input deck directs SPICE to determine the dc operating point of the circuit with inductors shorted and capacitors opened. Note: a dc analysis is automatically performed prior to a transient analysis to determine the transient initial conditions, and prior to an ac small-signal analysis to determine the linearized, small-signal models for nonlinear devices.

1. .OPTIONS Line

General form:

.OPTIONS OPT1 OPT2 ... (or OPT=OPTVAL ...)

Example:

```
.OPTIONS RELTOL=.005 TRTOL=8
```

This line allows the user to reset program control and user options for specific simulation purposes. Additional options for Nutmeg may be specified as well and take effect when Nutmeg reads the input deck. Options specified to Nutmeg via the 'set' command are also passed on to SPICE3 as if specified on a .OPTIONS line. See the following section on the interactive command interpretor for the parameters which may be set with a .OPTIONS line and the format of the 'set' command. Any combination of the following options may be included, in any order. 'x' (below) represents some positive number.

option	effect
GMIN=x	Resets the value of GMIN, the minimum conductance allowed by the program. The default value is 1.0e-12.
RELTOL=x	Resets the relative error tolerance of the program. The default value is 0.001 (0.1 percent).
ABSTOL=x	Resets the absolute current error tolerance of the program. The default value is 1 picoamp.
VNTOL=x	Resets the absolute voltage error tolerance of the program. The default value is 1 microvolt.
TRTOL=x	Resets the transient error tolerance. The default value is 7.0 This parameter is an estimate of the factor by which SPICE overestimates the actual truncation error.
CHGTOL=x	Resets the charge tolerance of the program. The default value is 1.0e-14.
PIVTOL=x	Resets the absolute minimum value for a matrix entry to be accepted as a pivot. The default value is 1.0e-13.

option	effect
PIVREL=x	Resets the relative ratio between the largest column entry and an acceptable pivot value. The default value is 1.0e-3. In the numerical pivoting algorithm the allowed minimum pivot value is determined by: EPSREL=AMAX1(PIVREL*MAXVAL, PIVTOL) where MAXVAL is the maximum element in the column where a pivot is sought (partial pivoting)
TNOM=x	Resets the nominal temperature at which device parameters are measured. The default value is 27 deg C (300 deg K). TNOM can be overridden by a specification on any temperature dependent device model.
TEMP=x	Resets the operating temperature of the circuit. The default value is 27 deg C (300 deg K). TEMP can be overridden by a temperature specification on any temperature dependent instance.
ITL1=x	Resets the dc iteration limit. The default is 100.
ITL2=x	Resets the dc transfer curve iteration limit. The default is 50.
ITL3=x	Resets the lower transient analysis iteration limit. the default value is 4. (Note: not implemented in Spice3).
ITL4=x	Resets the transient analysis timepoint iteration limit. The default is 10.
ITL5=x	Resets the transient analysis total iteration limit. The default is 5000. Set ITL5=0 to omit this test. (Note: not implemented in Spice3).
DEFL=x	Resets the value for MOS channel length; the default is 100.0 micrometer
DEFW=x	Resets the value for MOS channel width; the default is 100.0 micrometer.
DEFAD=x	Resets the value for MOS drain diffusion area; the default is 0.0.

DEFAS=x Resets the value for MOS source diffusion area; the default is 0.0.

TRYTOCOMPACT Applicable only to the LTRA model. When specified, the simulator tries to condense LTRA transmission lines' past history of input voltages and currents.

METHOD=name sets the numerical integration method used by SPICE. Possible names are "Gear" or "trapezoidal" (or just "trap"). The default is trapezoidal.

7. .AC Line

General form:

.AC DEC ND FSTART FSTOP
.AC OCT NO FSTART FSTOP
.AC LIN NP FSTART FSTOP

Examples:

```
.AC DEC 10 1 10K
.AC DEC 10 1K 100MEG
.AC LIN 100 1 100HZ
```

DEC stands for decade variation, and ND is the number of points per decade. OCT stands for octave variation, and NO is the number of points per octave. LIN stands for linear variation, and NP is the number of points. FSTART is the starting frequency, and FSTOP is the final frequency. If this line is included in the deck, SPICE performs an AC analysis of the circuit over the specified frequency range. Note that in order for this analysis to be meaningful, at least one independent source must have been specified with an ac value.

9. Device Models

1. Resistor Model
2. Capacitor Model
3. Uniform Distributed RC Model
4. Lossy Transmission Line (LTRA) Model
5. Switch Model
6. Diode Model
7. BJT Models (both NPN and PNP)
8. JFET Models (both N and P channel)
9. MOSFET Models (both N and P channel)
10. MESFET Models (both N and P channel)

8. JFET Models (both N and P Channel)

The JFET model is derived from the FET model of Shichman and Hodges. The dc characteristics are defined by the parameters VTO and BETA, which determine the variation of drain current with gate voltage, LAMBDA, which determines the output conductance, and IS, the saturation current of the two gate junctions. Two ohmic resistances, RD and RS, are included. Charge storage is modeled by nonlinear depletion layer capacitances for both gate junctions which vary as the -1/2 power of junction voltage and are defined by the parameters CGS, CGD, and PB.

name	parameter	units	default	example	area
VTO	threshold voltage (VTO)	V	-2.0	-2.0	
BETA	transconductance parameter (b)	A/V 2	1.0e-4	1.0e-3	*
LAMBDA	channel length modulation parameter	1/V	0	1.0e-4	
RD	drain ohmic resistance			0	100 *
RS	source ohmic resistance		0	100	*
CGS	zero-bias G-S junction capacitance	F	0	5pF	*
CGD	zero-bias G-D junction capacitance	F	0	1pF	*
PB	gate junction potential	V	1	0.6	
IS	gate junction saturation current	A	1e-14	1e-14	
KF	flicker noise coefficient	-	0		
AF	flicker noise exponent	-	1		
FC	coefficient for forward-bias depletion capacitance formula	-	0.5		
TNOM	parameter measurement temperature	deg C	27	50	

6. Diode Model

The dc characteristics of the diode are determined by the parameters IS and N. An ohmic resistance, RS, is included. Charge storage effects are modeled by a transit time, TT, and a nonlinear depletion layer capacitance which is determined by the parameters CJO, VJ, and M. The temperature dependence of the saturation current is defined by the parameters EG, the energy and XTI, the saturation current temperature exponent. The nominal temperature at which these parameters were measured is TNOM, which defaults to the circuit-wide value specified on the .OPTIONS control line. Reverse breakdown is modeled by an exponential increase in the reverse diode current and is determined by the parameters BV and IBV (both of which are positive numbers)

name	parameter	units		default	example	
IS	saturation current	A		1.0e-14	1.0e-14	
RS	ohmic resistance				0	10
N	emission coefficient	-	1	1.0		
TT	transit-time	sec	0	0.1ns		
CJO	zero-bias junction capacitance		F	0	2pF	
VJ	junction potential	V	1	0.6		
M	grading coefficient	-	0.5	0.5		
EG	activation energy	eV	1.11	1.11 Si		
			0.69 Sbd			
			0.67 Ge			
XTI	saturation-current temp. exp		-	3.0	3.0 jn	
			2.0 Sbd			
KF	flicker noise coefficient	-	0			
AF	flicker noise exponent	-	1			
FC	coefficient for forward-bias depletion capacitance formula		-	0.5		
BV	reverse breakdown voltage	V	infinite	40.0		
IBV	current at breakdown voltage		A	1.0e-3		
TNOM	parameter measurement temperature			deg. C	27	50

7. BJT Models (both NPN and PNP)

The bipolar junction transistor model in SPICE is an adaptation of the integral charge control model of Gummel and Poon. This modified Gummel-Poon model extends the original model to include several effects at high bias levels. The model automatically simplifies to the simpler Ebers-Moll model when certain parameters are not specified. The parameter names used in the modified Gummel-Poon model have been chosen to be more easily understood by the program user, and to reflect better both physical and circuit design thinking

The dc model is defined by the parameters IS, BF, NF, ISE, IKF, and NE which determine the forward current gain characteristics, IS, BR, NR, ISC, IKR, and NC which determine the reverse current gain .characteristics, and VAF and VAR which determine the output conductance for forward and reverse regions. Three ohmic resistances RB, RC, and RE are included, where RB can be high current dependent. Base charge storage is modeled by forward and reverse transit times, TF and TR, the forward transit time TF being bias dependent if desired, and nonlinear depletion layer capacitances which are determined by CJE, VJE, and MJE for the B-E junction , CJC, VJC, and MJC for the B-C junction and CJS, VJS, and MJS for the C-S (Collector-Substrate) junction. The temperature dependence of the saturation current, IS, is determined by the energy-gap, EG, and the saturation current temperature exponent, XTI. Additionally base current temperature dependence is modeled by the beta temperature exponent XTB in the new model. The values specified are assumed to have been measured at the temperature TNOM, which can be specified on the .OPTIONS control line or overridden by a specification on the .MODEL line.

The BJT parameters used in the modified Gummel-Poon model are listed below. The parameter names used in earlier versions of SPICE2 are still accepted.

Modified Gummel-Poon BJT Parameters.

name	parameter	units	default	example	area
IS	transport saturation current	A	1.0e-16	1.0e-15	*
BF	ideal maximum forward beta	-	100	100	
NF	forward current emission coefficient	-	1.0	1	
VAF	forward Early voltage	V	infinite	200	
IKF	corner for forward beta high current roll-off	A	infinite	0.01	*
ISE	B-E leakage saturation current	A	0	1.0e-13	*
NE	B-E leakage emission coefficient	-	1.5	2	
BR	ideal maximum reverse beta	-	1	0.1	
NR	reverse current emission coefficient	-	1 1		
VAR	reverse Early voltage	V	infinite	200	
IKR	corner for reverse beta high current roll-off	A	infinite	0.01	*
ISC	B-C leakage saturation current	A	0	1.0e-13	*
NC	B-C leakage emission coefficient	-	2	1.5	
RB	zero bias base resistance			0	100 *
IRB	current where base resistance falls halfway to its min value	A	infinite	0.1	*
RBM	minimum base resistance			RB	10 *

	at high currents					
RE	emitter resistance		0	1	*	
RC	collector resistance			0	10	*
CJE	B-E zero-bias depletion capacitance		F	0	2pF	*
VJE	B-E built-in potential	V	0.75	0.6		
MJE	B-E junction exponential factor	-	0.33	0.33		
TF	ideal forward transit time	sec	0	0.1ns		
XTF	coefficient for bias dependence of	TF	-	0		
VTF	voltage describing VBC dependence of TF	V	infinite			
ITF	high-current parameter for effect on TF	A	0	*		
PTF	excess phase at freq=1.0/(TF*2PI)	Hz	deg	0		
CJC	B-C zero-bias depletion capacitance	F		0	2pF	*
VJC	B-C built-in potential	V		0.75	0.5	
MJC	B-C junction exponential factor	-		0.33	0.5	
XCJC	fraction of B-C depletion capacitance connected to internal base node	-		1		
TR	ideal reverse transit time	sec		0	10ns	
CJS	zero-bias collector-substrate capacitance	F		0	2pF	*
VJS	substrate junction built-in potential	V		0.75		
MJS	substrate junction exponential factor	-		0	0.5	
XTB	forward and reverse beta temperature exponent	-		0		
EG	energy gap for temperature effect on IS	eV		1.11		
XTI	temperature exponent for effect on IS	-		3		
KF	flicker-noise coefficient	-		0		
AF	flicker-noise exponent	-		1		
FC	coefficient for forward-bias depletion capacitance formula	-		0.5		
TNOM	Parameter measurement temperature	deg C		27	50	

5. Switch Model

The switch model allows an almost ideal switch to be described in SPICE. The switch is not quite ideal, in that the resistance can not change from 0 to infinity, but must always have a finite positive value. By proper selection of the on and off resistances, they can be effectively zero and infinity in comparison to other circuit elements. The parameters available are:

name	parameter	units	default	switch
VT	threshold voltage	Volts	0.0	S
IT	threshold current	Amps	0.0	W
VH	hysteresis voltage	Volts	0.0	S
IH	hysteresis current	Amps	0.0	W
RON	on resistance		1.0	both
ROFF	off resistance		1/GMIN*	both

*(See the .OPTIONS control line for a description of GMIN, its default value results is a off resistance of 1.0e+12 ohms.)

The use of an ideal element that is highly non-linear such as a switch can cause large discontinuities to occur in the circuit node voltages. A rapid change such as that associated with a switch changing state can cause numerical roundoff or tolerance problems leading to erroneous results or timestep difficulties. The user of switches can improve the situation by taking the following steps: First of all it is wise to set ideal switch impedances only high and low enough to be negligible with respect to other circuit elements. Using switch impedances that are close to "ideal" in all cases aggravates the problem of discontinuities mentioned above. Of course, when modeling real devices such as MOSFETS, the on resistance should be adjusted to a realistic level depending on the size of the device being modeled. If a wide range of ON to OFF resistance must be used in the switches (ROFF/RON >1e+12), then the tolerance on errors allowed during transient analysis should be decreased by using the .OPTIONS control line and specifying TRTOL to be less than the default value of 7.0. When switches are placed around capacitors, then the option CHGTOL should also be reduced. Suggested values for these two options are 1.0 and 1e-16 respectively. These changes inform SPICE3 to be more careful around the switch points so that no errors are made due to the rapid change in the circuit.

3. Uniform Distributed RC Model

The URC model is derived from a model proposed by L. Gertzberrg in 1974. The model is accomplished by a subcircuit type expansion of the URC line into a network of lumped RC segments with internally generated nodes. The RC segments are in a geometric progression, increasing toward the middle of the URC line, with K as a proportionality constant. The number of lumped segments used, if not specified for the URC line device, is determined by the following formula: (not available yet)

The URC line is made up strictly of resistor and capacitor segments unless the ISPERL parameter is given a non-zero value, in which case the capacitors are replaced with reverse biased diodes with a zero bias junction capacitance equivalent to the capacitance replaced, and with a saturation current of ISPERL amps per meter of transmission line and an optional series resistance equivalent to RSPERL ohms per meter.

	name	parameter	units	default	example
1	K	Propagation Constant	-	2.0	1.2
2	FMAX	Maximum Frequency of interest	Hz	1.0G	6.5Meg
3	RPERL	Resistance per unit length	/m	1000	10
4	CPERL	Capacitance per unit length	F/m	1.0e-15 1pF	-
5	ISPERL	Saturation Current per unit length	A/m	0	-
6	RSPERL	Diode Resistance per unit length	/m	0	-

.4. Lossy Transmission Line Model

The uniform RLC/RC/LC/RG transmission line model to as the LTRA model henceforth) models a uniform constant-parameter distributed transmission line. The RC and LC cases may also be modelled using the URC and TRA models; however, the newer LTRA model is usually faster and more accurate than the others. The operation of the LTRA model is based on the convolution of the transmission line's impulse responses with its inputs see [9].

The LTRA model takes a number of parameters, some of which must be given and some of which are optional.

name	parameter	units/type	default	example
R	resistance/length		/unit	0.0 0.2
L	inductance/length	henrys/unit	0.0	9.13e-9
G	conductance/length	mhos/unit	0.0	0.0
C	capacitance/length	farads/unit	0.0	3.65e-12
LEN	length of line	no default		1.0
REL	breakpoint control	arbitrary unit	1	0.5
ABS	breakpoint control	arbitrary unit	1	5
NOSTEPLIMIT	don't limit timestep to less than line delay			
NOCONTROL	don't do complex timestep control	flag	not set	set
LININTERP	use linear interpolation	flag	not set	set
MIXEDINTERP	use linear when quadratic seems bad	flag	not set	set
COMPACTREL	special reltol for history compaction	flag	RELTOL	1.0e-3
COMPACTABS	special abstol for history compaction		ABSTOL	1.0e-9
TRUNCNR	use Newton-Raphson method for timestep control	flag	not set	set
TRUNCDONTCUT	don't limit timestep to keep impulse-response errors low	flag	not set	set

The following types of lines have been implemented so far: RLC (uniform transmission line with series loss only), RC (uniform RC line), LC (lossless transmission line), and RG (distributed series resistance and parallel conductance only). Any other combination will yield erroneous results and should not be tried. The length LEN of the line must be specified. NOSTEPLIMIT is a flag that will remove the default restriction of limiting time-steps to less than the line delay in the RLC case. NOCONTROL is a flag that prevents the default limiting of the time-step based on convolution error criteria in the RLC and RC cases. This speeds up simulation but may in some cases reduce the accuracy of results. LININTERP is a flag that, when specified, will use linear interpolation instead of the default quadratic interpolation for calculating delayed signals. MIXEDINTERP is a flag that, when specified, uses a metric for judging whether quadratic interpolation is not applicable and if so uses linear interpolation; otherwise it uses the default quadratic interpolation. TRUNCDONTCUT is a flag that removes the default cutting of the time-step to limit errors in the actual calculation of impulse-response related quantities. COMPACTREL and COMPACTABS are quantities that control the compaction of the past history of values stored for convolution. Larger values of these lower accuracy but usually increase simulation speed These are to be used with the TRYTOCOMPACT option, described in the OPTIONS section. TRUNCNR is a flag that turns on the use of Newton-Raphson iterations to determine an appropriate timestep in the timestep control routines. The default is a trial and error procedure by cutting the previous timestep in half. REL and ABS are quantities that control the setting of breakpoints.

The option most worth experimenting with for increasing the speed of simulation is REL. The default value of 1 is usually safe from the point of view of accuracy but occa- sionally increases computation time. A value of greater than 2 eliminates all breakpoints and may be worth trying depending on the nature of the rest of the circuit, keeping in mind that it might not be safe from the viewpoint of accu racy. Breakpoints may usually be entirely eliminated if it is expected the circuit will not display sharp discontinuities. Values between 0 and 1 are usually not required but may be used for setting many breakpoints.

COMPACTREL may also be experimented with when the option TRYTOCOMPACT is specified in a .OPTIONS card. The legal range is between 0 and 1. Larger values usually decrease the accuracy of the simulation but in some cases improve speed. If TRYTOCOMPACT is not specified on a .OPTIONS card, history compaction is not attempted and accuracy is high. NOCONTROL, TRUNCDONTCUT and NOSTEPLIMIT also tend to increase speed at the expense of accuracy.

2. Capacitor Model

The capacitor model contains process information that may be used to compute the capacitance from strictly geometric information.

name	parameter	units	default	example
CJ	junction bottom capacitance	F/meters^2	-	5e-5
CJSW	junction sidewall capacitance	F/meters	-	2e-11
DEFW	default device width	meters	1e-6	2e-6
NARROW	narrowing due to side etching	meters	0.0	1e-7

The capacitor has a capacitance computed as:

CAP = CJ (LENGTH - NARROW) (WIDTH - NARROW) + 2 CJSW (LENGTH + WIDTH - 2 NARROW)

1. Resistor Model

The resistor model consists of process-related device data that allow the resistance to be calculated from geometric information and to be corrected for temperature. The parameters available are:

name	parameter	units	default	example
TC1	first order temperature coeff.	/C	0.0	-
TC2	second order temperature coeff.	/C2	0.0	-
RSH	sheet resistance	/sq	-	50
DEFW	default width	meters	1e-6	2e-6
NARROW	narrowing due to side etching	meters	0.0	1e-7
TNOM	parameter measurement temperature	C	27	50

The sheet resistance is used with the narrowing parameter and L and W from the resistor device to determine the nominal resistance by the formula:

$$R = RSH (L - NARROW / W-NARROW$$

DEFW is used to supply a default value for W if one is not specified for the device. If either RSH or L is not specified, then the standard default resistance value of 1k is used. TNOM is used to override the circuit-wide value given on the .OPTIONS control line where the parameters of this model have been measured at a different temperature. After the nominal resistance is calculated, it is adjusted for temperature by the formula:

$$R(T) = T(T_0)[1+TC1(T-T_0)+TC2(T-T_0)_2]$$

Semiconductor Devices

1. Semiconductor Resistor
2. Semiconductor Capacitor
3. Uniform Distributed RC Lines (Lossy)
4. Junction Diodes
5. MESFETS

8. MESFETs

General form:

ZXXXXXXX ND NG NS MNAME <AREA> <OFF> <IC=VDS, VGS>

Example:

```
Z1 7 2 3 Zm1 Off
```

ND, NG, and NS are the drain, gate, and source nodes, respectively. MNAME is the model name, AREA is the area factor, and OFF indicates an (optional) initial condition on the device for dc analysis. If the area factor is omitted, a value of 1.0 is assumed. The (optional) initial condition specification, using IC=VDS, VGS is intended for use with the UIC option on the .TRAN control line, when a transient analysis is desired starting from other than the quiescent operating point. See the .IC control line for a better way to set initial conditions.

4. Junction Diodes

General form:

DXXXXXXX N+ N- MNAME <AREA> <OFF> <IC=VD> <TEMP=T>

Examples:

```
DBRIDGE 2 10 DIODE1
DCLMP 3 7 DMOD 3.0 IC=0.2
```

N+ and N- are the positive and negative nodes, respectively. MNAME is the model name, AREA is the area factor, and OFF indicates an (optional) starting condition on the device for dc analysis. If the area factor is omitted, a value of 1.0 is assumed. The (optional) initial condition specification using IC=VD is intended for use with the UIC option on the .TRAN control line, when a transient analysis is desired starting from other than the quiescent operating point. The (optional) TEMP value is the temperature at which this device is to operate, and overrides the temperature specification on the .OPTION control line.

3. Uniform Distributed RC Lines (Lossy)

General form:

UXXXXXXX N1 N2 N3 MNAME L=LEN <N=LUMPS>

Examples:

```
U1 1 2 0 URCMOD L=50U
URC2 1 12 2 UMODL l=1MIL N=6
```

N1 and N2 are the two element nodes the RC line connects, while N3 is the node to which the capacitances are connected. MNAME is the model name, LEN is the length of the RC line in meters. LUMPS, if specified, is the number of lumped segments to use in modeling the RC line (see the model description for the action taken if this parameter is omitted).

2. Semiconductor Capacitors

General form:

CXXXXXXX N1 N2 <VALUE> <MNAME> <L=LENGTH> <W=WIDTH> <IC=VAL>

Examples:

```
CLOAD 2 10 10P
CMOD 3 7 CMODEL L=10u W=1u
```

This is the more general form of the Capacitor presented in section 6.2, and allows for the calculation of the actual capacitance value from strictly geometric information and the specifications of the process. If VALUE is specified, it defines the capacitance. If MNAME is specified, then the capacitance is calculated from the process information in the model MNAME and the given LENGTH and WIDTH. If VALUE is not specified, then MNAME and LENGTH must be specified. If WIDTH is not specified, then it is taken from the default width given in the model. Either VALUE or MNAME, LENGTH, and WIDTH may be specified, but not both sets.

1. **Semiconductor Resistors**

General form:

RXXXXXXX N1 N2 <VALUE> <MNAME> <L=LENGTH> <W=WIDTH> <TEMP=T>

Examples:

```
RLOAD 2 10 10K
RMOD 3 7 RMODEL L=10u W=1u
```

This is the more general form of the resistor presented in section 6.1, and allows the modeling of temperature effects and for the calculation of the actual resistance value from strictly geometric information and the specifications of the process. If VALUE is specified, it overrides the geometric information and defines the resistance. If MNAME is specified, then the resistance may be calculated from the process information in the model MNAME and the given LENGTH and WIDTH. If VALUE is not specified, then MNAME and LENGTH must be specified. If WIDTH is not specified, then it is taken from the default width given in the model. The (optional) TEMP value is the temperature at which this device is to operate, and overrides the temperature specification on the .OPTION control line.

Example Circuits

Differential Pair
MOSFET Characterization
RTL Inverter
Four-Bit Binary Adder
Transmission-Line Inverter
Electronic Crossover

14. BIBLIOGRAPHY

[1] A. Vladimirescu and S. Liu, *The Simulation of MOS Integrated Circuits Using SPICE2*
ERL Memo No. ERL M80/7, Electronics Research Laboratory
University of California, Berkeley, October 1980

[2] T. Sakurai and A. R. Newton, *A Simple MOSFET Model for Circuit Analysis and its application to CMOS gate delay analysis and series-connected MOSFET Structure*
ERL Memo No. ERL M90/19, Electronics Research Laboratory,
University of California, Berkeley, March 1990

[3] B. J. Sheu, D. L. Scharfetter, and P. K. Ko, *SPICE2 Implementation of BSIM*
ERL Memo No. ERL M85/42, Electronics Research Laboratory
University of California, Berkeley, May 1985

[4] J. R. Pierret, *A MOS Parameter Extraction Program for the BSIM Model*
ERL Memo Nos. ERL M84/99 and M84/100, Electronics Research Laboratory
University of California, Berkeley, November 1984

[5] Min-Chie Jeng, *Design and Modeling of Deep-Submicrometer MOSFETS*
ERL Memo Nos. ERL M90/90, Electronics Research Laboratory
University of California, Berkeley, October 1990

[6] H.Statz et al., *GaAs FET Device and Circuit Simulation in SPICE*,
IEEE Transactions on Electron Devices, V34, Number 2, February, 1987 pp160-169.

[7] Soyeon Park, *Analysis and SPICE implementation of High Temperature Effects on MOSFET*,
Master's thesis, University of California, Berkeley, December 1986.

[8] Clement Szeto, *Simulator of Temperature Effects in MOSFETs (STEIM)*,
Master's thesis, University of California, Berkeley, May 1988.

[9] J.S. Roychowdhury and D.O. Pederson, *Efficient Transient Simulation of Lossy Interconnect*,
to appear in Proceedings of the 28th ACM/IEEE Design Automation Conference, June 17-21 1991, San Francisco

7. ELEMENT LINES

1. Resistors
2. Capacitors and Inductors
3. Coupled Inductors
4. Transmission Lines
5. Lossy Transmission Lines
6. Linear Dependent Sources
7. Non-linear Dependent Sources
8. Independent Sources
9. Switches

9. Switches

General form:

```
SXXXXXXX N+ N- NC+ NC- MODEL <ON><OFF>
 WYYYYYYY N+ N- VNAM MODEL <ON><OFF>
```

Examples:

s1 1 2 3 4 switch1 ON
s2 5 6 3 0 sm2 off
Switch1 1 2 10 0 smodel1
w1 1 2 vclock switchmod1
W2 3 0 vramp sm1 ON
wreset 5 6 vclck lossyswitch OFF

Nodes 1 and 2 are the nodes between which the switch terminals are connected. The model name is mandatory while the initial conditions are optional. For the voltage controlled switch, nodes 3 and 4 are the positive and negative controlling nodes respectively. For the current controlled switch, the controlling current is that through the specified voltage source. The direction of positive controlling current flow is from the positive node, through the source, to the negative node.

6. TITLE LINE, COMMENT LINES AND .END LINE

1. Title Line
2. .END Line
3. Comment Line

6.1. Title Line

Examples:

```
POWER AMPLIFIER CIRCUIT
TEST OF CAM CELL
```

This line must be the first in the input deck. Its contents are printed verbatim as the heading for each section of output

6.2. .END Line

Example:

```
.END
```

This line must always be the last in the input deck. Note that the period is an integral part of the name

6.3. Comment Line

General Form:

* <any comment>

Examples:

```
* RF=1K GAIN SHOULD BE 100
* MAY THE FORCE BE WITH MY CIRCUIT
```

The asterisk in the first column indicates that this line is a comment line. Comment lines may be placed anywhere in the circuit description. Note that SPICE3 also considers any line with leading white space to be a comment.

7.7. Non-linear Dependent Sources

General form:

BXXXXXXX N+ N- <I=EXPR> <V=EXPR>

Examples:

```
B1 0 1 I=cos(v(1))+sin(v(2))
B1 0 1 V=ln(cos(log(v(1,2)^2)))-v(3)^4+v(2)^(v1)
B1 3 4 I=17
```

N+ is the positive node, and N- is the negative node. The values of the **V** and **I** parameters determine the voltages and currents across and through the device, respectively. If **I** is given then the device is a current source, and if **V** is given the device is a voltage source. One and only one of these parameters must be given.

The small-signal AC behavior of the non-linear source is a linear dependant source (or sources) with a proportionality constant equal to the derivitive (or derivitives) of the source at the DC operati.

The expressions given for **V a**nd **I** may be any function of voltages and currents through voltage sources in the system. The following functions of real variables are defined:

abs	asinh	cosh	sin
acos	atan	exp	sinh
acosh	tanh	ln	sqrt
asin	cos	log	tan

The following operations are defined:

+ - * / ^ unary -

If the argument of log, ln, or sqrt becomes less than zero, the absolute value of the argument is used. If a divisor becomes zero or the argument of log or ln becomes zero, an error will result. Other problems may occur when the argument for a function in a partial derivitive enters a region where that function is undefined.

To get time into the expression you can integrate the current from a constant current source with a capacitor and use the resulting voltage (don't forget to set the initial voltage across the capacitor). Non- linear capacitors, resistors, and inductors may be synthisized with the non-linear dependant source. Non- linear resistors are obvious. Non-linear capacitors and inductors are implemented with their linear counter- parts by a change of variables implemented with the non-linear dependant source. The following subcircuit will implement a non-linear capacitor:

```
* Bx: calculate f(input voltage)
Bx 1 0 v = f(v(pos,neg))
* Cx: linear capacitance
Cx 2 0 1
* Vx: Ammeter to measure current into the capacitor
Vx 2 1 DC 0Volts
* Drive the current through Cx back into the circuit
Fx pos neg Vx 1
```

Non-linear inductors are similar.

7.8.2. Sinusoidal

General form:

SIN(VO VA FREQ TD THETA)

Example:

```
VIN 3 0 SIN(0 1 100MEG 1NS 1E10)
```

Parameter	Default value	Units
VO (offset)		Volts or Amps
VA (amplitude)		Volts or Amps
FREQ (frequency)	1/TSTOP	Hz
TD (delay)	0.0	seconds
THETA (damping factor)	0.0	1/seconds

The shape of the waveform is described by the following table:

Time	Value
0 to TD	VO
TD to TSTOP	$VO + VA\, e^{-(t - TD)THETA} \sin(2 \quad FREQ\ (t + TD))$

7.8.1. Pulse

General form:

PULSE(V1 V2 TD TR TF PW PER)

Example:

```
VIN 3 0 PULSE(-1 1 2NS 2NS 2NS 50NS 100NS)
```

Parameter	Default value	Units
V1 (initial value)		Volts or Amps
V2 (pulsed value)		Volts or Amps
TD (delay time)	0.0	Seconds
TR (rise time)	TSTEP	Seconds
TF (fall time)	TSTEP	Seconds
PW (pulse width)	TSTOP	Seconds
PER (period)	TSTOP	Seconds

A single pulse so specified is described by the following table:

Time	Value
0	V1
TD	V1
TD+TR	V2
TD+TR+PW	V2
TD+TR+PW+TF	V1
TSTOP	V1

Intermediate points are determined by linear interpolation.

8. Independent Sources

1. Pulse
2. Sinusoidal
3. Exponential
4. Piece-Wise Linear
5. Single-Frequency FM

8.5. Single-Frequency FM

General Form:

SFFM(VO VA FC MDI FS)

Example:

```
V1 12 0 SFFM(0 1M 20K 5 1K)
```

parameter	default value	units
VO (offset)		Volts or Amps
VA (amplitude)		Volts or Amps
FC (carrier frequency)	1/TSTOP	Hz
MDI (modulation index)		
FS (signal frequency)	1/TSTOP	Hz

8.4. Piece-Wise Linear

General Form:

PWL(T1 V1 <T2 V2 T3 V3 T4 V4 ...>)

Example:

```
VCLOCK 7 5 PWL(0 -7 10NS -7 11NS -3 17NS -3 18NS -7 50NS -7)
```

Each pair of values (Ti, Vi) specifies that the value of the source is Vi (in Volts or Amps) at time=Ti. The value of the source at intermediate values of time is determined by using linear interpolation on the input values.

8.3. Exponential

General Form:

EXP(V1 V2 TD1 TAU1 TD2 TAU2)

Example:

```
VIN 3 0 EXP(-4 -1 2NS 30NS 60NS 40NS)
```

parameter	default value	units
V1 (initial value)		Volts or Amps
V2 (pulsed value)		Volts or Amps
TD1 (rise delay time)	0.0 seconds	
TAU1 (rise time constant)	TSTEP seconds	
TD2 (fall delay time)	TD1+TSTEP seconds	
TAU2 (fall time constant)	TSTEP seconds	

7.4. Transmission Lines (Lossless)

General form:

TXXXXXXX N1 N2 N3 N4 Z0=VALUE <TD=VALUE> <F=FREQ <NL=NRMLEN>> <IC=V1, I1, V2, I2>

Example:

```
T1 1 0 2 0 Z0=50 TD=10NS
```

N1 and N2 are the nodes at port 1; N3 and N4 are the nodes at port 2. Z0 is the characteristic impedance. The length of the line may be expressed in either of two forms. The transmission delay, TD, be given, together with NL, the normalized electrical length of the transmission line with respect to the wavelength in the line at the frequency F. If a frequency is specified but NL is omitted, 0.25 is assumed (that is, the frequency is assumed to be the frequency). Note that although both forms for expressing the line length are indicated as optional, one of the two must be specified. Note that this element models only one propagating mode. If all four nodes are distinct in the actual circuit, then two modes may be excited. To simulate such a situation, two transmission-line elements are required. (see the example in Appendix A for further clarification.) The (optional) initial condition specification consists of the voltage and current at each of the transmission line ports. Note that the initial conditions (if any) apply 'only' if the UIC option is specified on the .TRAN control line. Note that a lossy transmission line (see below) with zero loss may be more accurate than than the lossless transmission line due to implementation details.

7.5. Lossy Transmission Lines

General form:

OXXXXXXX N1 N2 N3 N4 MNAME

Examples:

```
O23 1 0 2 0 LOSSYMOD
OCONNECT 10 5 20 5 INTERCONNECT
```

This is a two-port convolution model for single-conductor lossy transmission lines. N1 and N2 are the nodes at port 1; N3 and N4 are the nodes at port 2. Note that a lossy transmission line with zero loss may be more accurate than than the lossless transmission line due to implementation details. For further details, see the description of the LTRA type of the .MODEL card.

7.3. Coupled (Mutual) Inductors

General form:

KXXXXXXX LYYYYYYY LZZZZZZZ VALUE

Examples:

```
K43 LAA LBB 0.999
KXFRMR L1 L2 0.87
```

LYYYYYYY and LZZZZZZZ are the names of the two coupled inductors, and VALUE is the coefficient of coupling, K, which must be greater than 0 and less than or equal to 1. Using the 'dot' convention, place a 'dot' on the first node of each inductor.

5. CIRCUIT DESCRIPTION

The circuit to be analyzed is described to SPICE by a set of element lines, which define the circuit topology and element values, and a set of control lines, which define the model parameters and the run controls. The first line in the input file must be the title, and the last line must be ".END". The order of the remaining lines is arbitrary (except, of course, that continuation lines must immediately follow the line being continued). Each element in the circuit is specified by an element line that contains the element name, the circuit nodes to which the element is connected, and the values of the parameters that determine the electrical characteristics of the element. The first letter of the element name specifies the element type.

The format for the SPICE element types is given in what follows. The strings XXXXXXX, YYYYYYY, and ZZZZZZZ denote arbitrary alphanumeric strings. For example, a resistor name must begin with the letter R and can contain one or more characters. Hence, R, R1, RSE, ROUT, and R3AC2ZY are valid resistor names.

Data fields that are enclosed in less than and greater than signs '< >' are optional. All indicated punctuation (parentheses, equal signs, etc.) is optional and merely indicate the presence of any delimiter. A consistent style such as that shown here makes the input easier to understand. With respect to branch voltages and currents, SPICE uniformly uses the associated convention (current flows in the direction of voltage drop). Node names may be arbitrary character strings. The datum (ground) node must be named '0'.

Note the difference in SPICE3 where the nodes are treated as character strings and not evaluated as numbers, thus '0' and '00' are distinct nodes in SPICE3 but not in SPICE2. The circuit cannot contain a loop of voltage sources and/or inductors and cannot contain a cutset of current sources and/or capacitors. Each node in the circuit must have a dc path to ground. Every node must have at least two connections except for transmission line nodes (to permit unterminated transmission lines) and MOSFET substrate nodes (which have two internal connections anyway).

4. INPUT FORMAT

The input format for SPICE is of the free format type. Fields on a line are separated by one or more blanks, a comma, an equal ('=') sign, or a left or right parenthesis; extra spaces are ignored. A line may be continued by entering a '+' (plus) in column 1 of the following line; SPICE continues reading beginning with column 2.

A name field must begin with a letter (A through Z) and cannot contain any delimiters.

A number field may be an integer field (12, -44), a floating point field (3.14159), either an integer or floating point number followed by an integer exponent (1e-14, 2.65e3), or either an integer or a floating point number followed by one of the following scale factors:

T = 103	G = 109	Meg = 106	K = 103	mil = 25.4-6
m = 10-3	u(or) = 10-6	n = 0-9	p = 10-12	f = 10-15

Letters immediately following a number that are not scale factors are ignored, and letters immediately following a scale factor are ignored. Hence, 10, 10V, 10Volts, and 10Hz all represent the same number, and M, MA, MSec, and MMhos all represent the same scale factor. Note that 1000, 1000.0, 1000Hz, 1e3, 1.0e3, 1KHz, and 1K all represent the same number.

3. CONVERGENCE

Both dc and transient solutions are obtained by an iterative process which is terminated when both of the following conditions hold:

1) The nonlinear branch currents converge to within a tolerance of 0.1 percent or 1 picoamp (1.0e-12 Amp), whichever is larger.

2) The node voltages converge to within a tolerance of 0.1 percent or 1 microvolt (1.0e-6 Volt), whichever is larger.

Although the algorithm used in SPICE has been found to be very reliable, in some cases it fails to converge to a solution. When this failure occurs, the program terminates the job. Failure to converge in dc analysis is usually due to an error in specifying circuit connections, element values, or model parameter values. Regenerative switching circuits or circuits with positive feedback probably will not converge in the dc analysis unless the OFF option is used for some of the devices in the feedback path, or the .NODESET control line is used to force the circuit to converge to the desired state.

Types of Analysis

1. DC Analysis
2. AC Small-Signal Analysis
3. Transient Analysis
4. Pole-Zero Analysis
5. Small-Signal Distortion Analysis
6. Noise Analysis
7. Analysis at different temperatures

7. Analysis at different temperatures

All input data for SPICE is assumed to have been measured at a nominal temperature of 27 which can be changed by use the the TNOM parameter on the .OPTION control line. This value can further be overridden for any device which models temperature effects by specifying the TNOM parameter on the model itself. The circuit simulation is performed at a temperature of 27 C unless overridden by a TEMP parameter on the .OPTION control line. Individual instances may further override the circuit temperature through the specification of a TEMP parameter on the instance.

Temperature dependent support is provided for resistors, diodes, JFET's, BJT's, and level 1, 2, and 3 MOSFET's. BSIM (level 4) MOSFET's have an alternate temperature dependency scheme which adjusts all of the model parameters before input to SPICE.

Temperature appears explicitly in the exponential terms of the BJT and diode model equations. In addition, saturation currents have a built-in temperature dependence. The temperature dependence of the saturation current in the BJT models is determined by:

where k is Boltzmann's constant, q is the electronic charge, E G is the energy gap which is a model parameter, and XTI is the saturation current temperature exponent (also a model parameter, and usually equal to 3).

The temperature dependence of forward and reverse beta is according to the formula:

where T1 and T0 are in degrees Kelvin, and XTB is a user-supplied model parameter. Temperature effects on beta are carried out by appropriate adjustment to the values of b F , I SE, b R, and ISC (spice model parameters BF, ISE, BR, and ISC, respectively).

6. Noise Analysis

The noise analysis portion of SPICE does analysis noise for the given circuit. When provided with an input source and an output port, the analysis calculates the noise contributions of each device (and each noise generator within the device) to the output port voltage. It also calculates the input noise to the circuit, equivalent to the output noise referred to the specified input source. This is done for every frequency point in a specified range the calculated value of the noise corresponds to the spectral density of the circuit variable viewed as a stationary gaussian stochastic process.

After calculating the spectral densities, noise analysis integrates these values over the specified frequency range to arrive at the total noise voltage/current (over this frequency range). This calculated value corresponds to the variance of the circuit variable viewed as a stationary gaussian process.

5. Small-Signal Distortion Analysis

The distortion analysis portion of SPICE computes steady-state harmonic and intermodulation products for small input signal magnitudes. If signals of a single frequency are specified as the input to the circuit, the complex values of the second and third harmonics are determined at every point in the circuit. If there are signals of two frequencies input to the circuit, the analysis finds out the complex values of the circuit variables at the sum and difference of the input frequencies, and at the difference of the smaller frequency from the second harmonic of the larger frequency.

Distortion analysis is supported for the following nonlinear devices: DIO, BJT, JFET, MOSFETs (levels 1, 2, 3 and BSIM) and MESFETS. All linear devices are automatically supported by distortion analysis. If there are switches present in the circuit, the analysis continues to be accurate provided the switches do not change state under the small excitations used for distortion calculations.

4. Pole-Zero Analysis

The pole-zero analysis portion of SPICE computes the poles and/or zeros in the small-signal ac transfer function. The program first computes the dc operating point and then determines the linearized, small-signal models for all the nonlinear devices in the circuit. This circuit is then used to find the poles and zeros of the transfer function.

Two types of transfer functions are allowed : one of the form (output voltage)/(input voltage) and the other of the form (output voltage)/(input current). These two types of transfer functions cover all the cases and one can find the poles/zeros of functions like input/output impedance and voltage gain. The input and output ports are specified as two pairs of nodes.

The pole-zero analysis works with resistors, capacitors, inductors, linear-controlled sources, independent sources, BJTs, MOSFETs, JFETs and diodes. Transmission lines are not supported.

3. Transient Analysis

The transient analysis portion of SPICE computes the transient output variables as a function of time over a user-specified time interval. The initial conditions are automatically determined by a dc analysis. All sources which are not time dependent (for example, power supplies) are set to their dc value. The transient time interval is specified on a .TRAN control line.

2. AC Small-Signal Analysis

The ac small-signal portion of SPICE computes the ac output variables as a function of frequency. The program first computes the dc operating point of the circuit and determines linearized, small-signal models for all of the nonlinear devices in the circuit. The resultant linear circuit is then analyzed over a user-specified range of frequencies. The desired output of an ac small- signal analysis is usually a transfer function (voltage gain, transimpedance, etc). If the circuit has only one ac input, it is convenient to set that input to unity and zero phase, so that output variables have the same value as the transfer function of the output variable with respect to the input.

1. DC Analysis

The dc analysis portion of SPICE determines the dc operating point of the circuit with inductors shorted and capacitors opened. The dc analysis options are specified on the .DC, .TF, and .OP control lines. A dc analysis is automatically performed prior to a transient analysis to determine the transient initial conditions, and prior to an ac small-signal analysis to determine the linearized, small-signal models for nonlinear devices. If requested, the dc small-signal value of a transfer function (ratio of output variable to input source), input resistance, and output resistance is also computed as a part of the dc solution. The dc analysis can also be used to generate dc transfer curves: a specified independent voltage or current source is stepped over a user-specified range and the dc output variables are stored for each sequential source value.

1. Introduction

SPICE32 is a general-purpose circuit simulation program for nonlinear dc, nonlinear transient, and linear ac analyses. Circuits may contain resistors, capacitors, inductors, mutual inductors, independent voltage and current sources, four types of dependent sources, lossless and lossy transmission lines (two seperate implementations), switches, uniform distributed RC lines, and the five most common semiconductor devices: diodes, BJTs, JFETs, MESFETs, and MOSFETs.

The SPICE3 version is based directly on SPICE 2G.6. While SPICE3 is being developed to include new features, it continues to support those capabilities and models which remain in extensive use in the SPICE2 program.

SPICE has built-in models for the semiconductor devices, and the user need specify only the pertinent model parameter values. The model for the BJT is based on the integral-charge model of Gummel and Poon; however, if the Gummel-Poon parameters are not specified, the model reduces to the simpler Ebers-Moll model. In either case, effects, ohmic resistances, and a output conductance may be included. The diode model can be used for either junction diodes or Schottky barrier diodes. The JFET model is based on the FET model of Shichman and Hodges. Six MOSFET models are implemented: MOS1 is described by a square-law I-V characteristic, MOS2[1] is an analytical model, while MOS3[1] is a semi-empirical model; MOS6[2] is a simple analytic model accurate in the short channel region; MOS4[3, 4] and MOS5[5] are the BSIM (Berkeley Short-channel IGFET Model) and BSIM2. MOS2, MOS3, and MOS4 include second-order effects such as channel length modulation, subthreshold conduction, scattering-limited velocity saturation, small-size effects, and charge-controlled capacitances.

Linear Dependent Sources

Spice32 allows circuits to contain linear dependent sources characterized by any of the four equations:
i = gv v = ev i = fi v=hi
where g e f and h are constants representing transconductance, voltage gain, current gain, and transresistance, respectively.

See:

1. Linear Voltage-Controlled Current Sources
2. Linear Voltage-Controlled Voltage Sources
3. Linear Current-Controlled Current Sources
4. Linear Current-Controlled Voltage Sources

Linear Current-Controlled Voltage Sources

General form:

```
HXXXXXXX   N+   N-   VNAM-   VALUE
```

Example:

```
H1   13   5   VZ 0.5K
```

N+ and N- are the positive and negative nodes, respectively. VNAM is the name of a voltage source through which the controlling current flows. The direction of positive controlling current flow is from the positive node, through the source, to the negative node of VNAM. VALUE is the transresistance in ohms.

Linear Current-Controlled Current Sources

General form:

```
FXXXXXXX   N+   N-   VNAM-   VALUE
```

Example:

```
F1   13   5   VSENS 5
```

N+ and N- are the positive and negative nodes, respectively. Current flow is from the positive node, through the source, to the negative node. VNAM is the name of a voltage source through which the controlling current flows. The direction of positive controlling current flow is from the positive node, through the source, to the negative node of VNAM. VALUE is the current gain.

Linear Voltage Controlled Voltage Sources

General form:

```
EXXXXXXX  N+  N-  NC+  NC-  VALUE
```

Example:

```
E1  2  0  5  0  2.0
```

N+ and N- are the positive and negative nodes, respectively. NC+ and NC- are the positive and negative controlling nodes. VALUE is the voltage gain.

Linear Voltage-Controlled Current Sources

General form:

```
GXXXXXXX   N+   N-   NC+   NC-   VALUE
```

Example:

```
G1   2   0   5   0   0.1MHO
```

N+ and N- are the positive and negative nodes, respectively. Current flow is from the positive node, through the source, to the negative node. NC+ and NC- are the positive and negative controlling nodes. VALUE is the transconductance in mhos.

Capacitors and Inductors

General form:

```
CXXXXXXX N+ N- Value <IC=INCOND>
LYYYYYYY N+ N- Value <IC=INCOND>
```

Examples:

```
CBYP 13 0 1UF
COSC 17 23 10U IC=3V
LLINK 42 69 1UH
LSHUNT 23 51 10U IC=15.7MA
```

Resistors

General form:

```
RXXXXXXX N1 N2 Value
```

Examples:

```
R1 1 2 100
RC1 12 17 1k
Rbias 12 1 5e3
```

File Menu

- Load
- Reload & re-execute
- Create Spice database
- Save As
- Print
- Exit

Save As

This command allows you to save the current binary database in ascii form.

Create Spice database

This command runs spice on the selected circuit (*.cir) file. This command is equivalent to running "spice32 <filename.cir> -r <filename.bin>" from the dos command line. Running Spice from the command line allows viewing of diagnostic output.

Reload & re-execute

This command reloads the database and re-executes the last plot command.

File Load

Use this command to load an existing spice binary database file.

Dialog Box Options

File Name

Type or select the filename you want to open. This box lists files with the extension you select in the List Files Of Type box.

Note: To see a list of files with a particular extension, type an asterisk (*), a period, and the three-character extension. For example, if you want to see all files with a .cir extension in a directory, type *.cir

List Files Of Type

Select the type of file you want to open.

Spice binary files (*.bin)	Lists all files in the current directory that were saved with a .BIN extension.
Spice circuit files (*.cir)	Lists all files in the current directory that were saved with a .CIR extension.
Spice ascii files (*.raw)	Lists all files in the current directory that were saved with a .RAW extension.
All Files (*.*)	Lists all files in the current directory.

Drives

Select the drive which contains the file that you want to open.

Directories

Select the directory which contains the file that you want to open.

Plot Command

Enter the list of vectors (circuit nodes) that you want plotted. The command syntax is:
<vectors> [ylimit ylo yhi] [xlimit xlo xhi] [xindices xilo xihi] [xcompress comp] [xdelta xdel] ydelta ydel] [xlabel word] [ylabel word] [title word]

Example- plot the voltages at nodes 14, 2, & 5: (Transient analysis selected)

```
v(14) v(2) v(5)
```

Example- plot the magnitude and phase of the voltage at node 14: (AC analysis selected)

```
mag(v(14)) phase(v(14))
```

or-

```
mag(v(14)) vp(14)
```

Example- plot the magnitude in decibels and phase of the voltage at note 14: AC analysis selected)

```
db(v(14)) vp(14)
```

Also see Plot Functions

Menu Commands

Plot Vectors

Plot the given vector(s). Multiple vectors may be specified on the command line, These vectors will be real or imaginary, depending on the analysis.

File Exit Command

This command exits the program.

File Print Command

This command allows you to print the current plot, in the current zoom. Plots often contain a great deal of information, especially if the analysis specifies many time (or frequency) steps. Extra information is often necessary for detail in a very small zoom area. This extra detail often creates large print images. These images can be reduced by zooming in on the desired areas or reducing the amount of data requested in the circuit file.

Data

Show Coordinates

Show Coordinates

Use the show coordinates dialog box to view:

- The x and y values at the cursor location.
- The delta x and y between two mouse clicks.
- The slope of a straight line between two mouse clicks.
- The equivalent frequency of the time span between two mouse clicks.

Edit Menu

Use this menu to copy the plot contents to the clipboard

Analysis Menu

Use this dialog box to select the analysis type:
Transient, AC, DC, Pole-Zero, Noise, and AC small-signal distortion analysis types are available.

Axis

Zoom on the plot area by setting the x and y axis limits. You can also zoom by holding down the left mouse button and dragging. Return to normal zoom by double clicking the right mouse button.

Plot Functions

mag(vector) - The magnitude of vector.

ph(vector) - The phase of vector.

j(vector) -i (sqrt(-1)) times vector.

real(vector) - The real component of vector.

imag(vector) - The imaginary part of vector.

db(vector) - 20 * log10(mag(vector)).

log(vector) - The logarithm (base 10) of the vector.

ln(vector) - The natural logarithm (base e) of vector.

exp(vector) - e to the vector power.

abs(vector) - The absolute value of vector.

sqrt(vector) - The square root of vector.

sin(vector) - The sin of vector.

cos(vector) - The cosine of vector.

tan(vector) - The tangent of vector.

atan(vector) - The inverse tangent of vector.

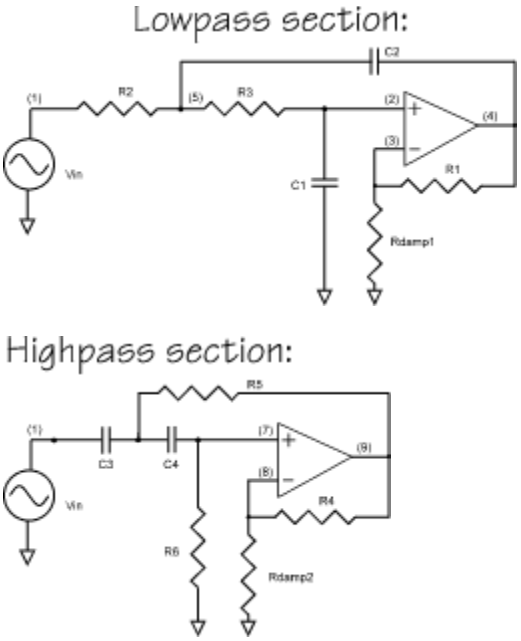
norm(vector) - The vector normalized to 1 (i.e, the largest magnitude of any component will be 1).

rnd(vector) - A vector with each component a random integer between 0 and the absolute value of the vectors's corresponding component.

mean(vector) - The result is a scalar (a length 1 vector) that is the mean of the elements of vector.

Electronic crossover

The following circuit is a low pass and high pass electronic crossover. The two operational amplifers have been implemented with voltage controlled voltage sources with a gain of 5000 for simplicity.




```
Electronic Crossover
.ac DEC 30 20 20k

*Lowpass section: (output on node 4)
vin 1 0 ac 1
*simulate an op-amp for fast converge:
e1      4 0 2 3 5000
*Change rdamp1 to change damping
rdamp1 3 0      4.65k
r1      3 4     2.8k
*Change r2 & r3 to change frequency (r2 = r3)
*F(-3dB) = 1/(2pi*r*c)
r2      1 5     10k
r3      5 2     10k
c1      2 0     .1uf
c2      5 4     .1uf

*Highpass section:  (output on node 9)
e2      9 0 7 8 5000
rdamp2 8 0      4.65k
r4      8 9     2.8k
r5      6 9     10k
r6      7 0     10k
c3      1 6     .1uf
c4      6 7     .1uf
.end
```

RTL Inverter

The following deck determines the dc transfer curve and the transient pulse response of a simple RTL inverter. The input is a pulse from 0 to 5 Volts with delay, rise, and fall times of 2ns and a pulse width of 30ns. The transient interval is 0 to 100ns, with printing to be done every nanosecond.

```
SIMPLE RTL INVERTER
.DC             VIN 0 5 0.1
.TRAN 1NS 100NS
VCC 4 0         5
VIN 1 0         PULSE 0 5 2NS 2NS 2NS 30NS
RB 1 2          10K
Q1 3 2 0        Q1
RC 3 4          1K
.MODEL          Q1 NPN BF 20 RB 100 TF .1NS CJC 2PF
.END
```

Four-Bit Binary Adder

The following deck simulates a four-bit binary adder, using several subcircuits to describe various pieces of the overall circuit.

```
ADDER - 4 BIT ALL-NAND-GATE BINARY ADDER
*** (FOR THOSE WITH MONEY (AND MEMORY) TO BURN)
.TRAN 1NS 6400NS

*** SUBCIRCUIT DEFINITIONS
.SUBCKT NAND 1 2 3 4
*    NODES: INPUT(2), OUTPUT, VCC
Q1 9 5 1        QMOD
D1CLAMP 0 1     DMOD
Q2 9 5 2        QMOD
D2CLAMP 0 2     DMOD
RB 4 5          4K
R1 4 6          1.6K
Q3 6 9 8        QMOD
R2 8 0          1K
RC 4 7          130
Q4 7 6 10       QMOD
DVBEDROP 10 3    DMOD
Q5 3 8 0        QMOD
.ENDS NAND

.SUBCKT ONEBIT 1 2 3 4 5 6
*    NODES: INPUT(2), CARRY-IN, OUTPUT, CARRY-OUT, VCC
X1 1 2 7 6      NAND
X2 1 7 8 6      NAND
X3 2 7 9 6      NAND
X4 8 9 10 6     NAND
X5 3 10 11 6    NAND
X6 3 11 12 6    NAND
X7 10 11 13 6   NAND
X8 12 13 4 6    NAND
X9 11 7 5 6     NAND
.ENDS ONEBIT

.SUBCKT TWOBIT 1 2 3 4 5 6 7 8 9
*     NODES: INPUT - BIT0(2) / BIT1(2), OUTPUT - BIT0 / BIT1,
*     CARRY-IN, CARRY-OUT, VCC
X1 1 2 7 5 10 9    ONEBIT
X2 3 4 10 6 8 9    ONEBIT
.ENDS TWOBIT

.SUBCKT FOURBIT 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
*     NODES: NPUT - BIT0(2) / BIT1(2) / BIT2(2) / BIT3(2),
*     OUTPUT - BIT0 / BIT1 / BIT2 / BIT3, CARRY-IN, CARRY-OUT, VCC
X1 1 2 3 4 9 10 13 16 15       TWOBIT
X2 5 6 7 8 11 12 16 14 15      TWOBIT
.ENDS FOURBIT

*** DEFINE NOMINAL CIRCUIT
.MODEL DMOD     D
.MODEL QMOD     NPN(BF=75 RB=100 CJE=1PF CJC=3PF)
VCC 99 0        DC 5V
```

```
VIN1A 1 0      PULSE(0 3 0 10NS 10NS 10NS 50NS)
VIN1B 2 0      PULSE(0 3 0 10NS 10NS 20NS 100NS)
VIN2A 3 0      PULSE(0 3 0 10NS 10NS 40NS 200NS)
VIN2B 4 0      PULSE(0 3 0 10NS 10NS 80NS 400NS)
VIN3A 5 0      PULSE(0 3 0 10NS 10NS 160NS 800NS)
VIN3B 6 0      PULSE(0 3 0 10NS 10NS 320NS 1600NS)
VIN4A 7 0      PULSE(0 3 0 10NS 10NS 640NS 3200NS)
VIN4B 8 0      PULSE(0 3 0 10NS 10NS 1280NS 6400NS)
X1 1 2 3 4 5 6 7 8 9 10 11 12 0 13 99 FOURBIT
RBIT0 9 0      1K
RBIT1 10 0     1K
RBIT2 11 0     1K
RBIT3 12 0     1K
RCOUT 13 0     1K
.END
```

Transmission-Line Inverter

The following deck simulates a transmission-line inverter. Two transmission-line elements are required since two propagation modes are excited. In the case of a coaxial line, the first line (T1) models the inner conductor with respect to the shield, and the second line (T2) models the shield with respect to the outside world.

```
TRANSMISSION-LINE INVERTER
.TRAN 0.1NS 20NS
V1 1 0          PULSE(0 1 0 0.1N)
R1 1 2          50
X1 2 0 0 4      TLINE
R2 4 0          50

.SUBCKT TLINE 1 2 3 4
T1 1 2 3 4 Z0=50 TD=1.5NS
T2 2 0 4 0 Z0=100 TD=1NS
.ENDS TLINE

.END
```

MOSFET Characterization

The following deck computes the output characteristics of a MOSFET device over the range 0-10V for VDS and 0-5V for VGS.

```
MOS OUTPUT CHARACTERISTICS
VDS             3 0
VGS             2 0
M1 1 2 0 0      MOD1 L=4U W=6U AD=10P AS=10P
*    VIDS MEASURES ID, WE COULD HAVE USED VDS, BUT ID WOULD BE NEGATIVE
VIDS 3 1
.MODEL MOD1 NMOS VTO=-2 NSUB=1.0E15 UO=550
.DC             VDS 0 10 .5 VGS 0 5 1
.END
```

Differential Pair

The following deck determines the dc operating point of a simple differential pair. In addition, the ac small-signal response is computed over the frequency range 1Hz to 100MEGHz.

```
SIMPLE DIFFERENTIAL PAIR
.AC DEC 10 1 100MEG
.tran 2ns 500ns 0 2ns

VCC 7 0           12
VEE 8 0           -12
VIN 1 0           pwl(0 0 .1us 0 .12us 1 .15us 1 .17us -1 .23us -1 .25us 0) AC
1
RS1 1 2           1K
RS2 6 0           1K
Q1 3 2 4          MOD1
Q2 5 6 4          MOD1
RC1 7 3           10K
RC2 7 5           10K
RE 4 8            10K
.MODEL MOD1 NPN BF=50 VAF=50 IS=1.E-12 RB=100 CJC=.5PF TF=.6NS

.END
```