SPICE

NAME
spice circuit simulator

SYNOPSIS
spice [-n] [-t term] [-r rawfile] [-b] [-i] [input file ... **]**

DESCRIPTION
This manual page describes the commands available for interactive use of SPICE3. For details of circuit descriptions and the process of simulating a circuit, see the SPICE3 User's Manual. The commands available are a superset of those available for nutmeg only the additional commands available in SPICE3 are described here. You should be familiar with the manual page for nutmeg(1) before reading this manual page.

Arguments are:

-n (or -N)
Don't try to source the file ".spiceinit" upon startup. Normally SPICE3 tries to find the file in the current directory, and if it is not found then in the user's home directory.

-t term (or -T term)
The program is being run on a terminal with mfb name term.

-b (or -B)
Run in batch mode. SPICE3 will read the standard input or the specified input file and do the simulation. Note that if the standard input is not a terminal, SPICE3 will default to batch mode, unless the -i flag is given.

-s (or **-S**)
Run in server mode. This is like batch mode, except that a temporary rawfile is used and then written to the standard output, preceded by a line with a single "@", after the simulation is done. This mode is used by the spice daemon.

-i (or **-I**)
Run in interactive mode. This is useful if the standard input is not a terminal but interactive mode is desired. Command completion is not available unless the standard input is a terminal, however.

-r rawfile (or **-R rawfile**)
Use **rawfile** as the default file into which the results of the simulation are saved.

Further arguments are taken to be SPICE3 input decks, which are read and saved. (If batch mode is requested then they are run immediately.)

SPICE3 will accept any SPICE2 input decks, and output ascii plots, fourier analyses, and node printouts as specified in .plot, .four, and .print cards. If a out parameter is given on a .width card, the effect is the same as set width = ... Since SPICE3 ascii plots do not use multiple ranges, however, if vectors together on a .plot card have different ranges they will not provide as much information as they would in SPICE2. The output of SPICE3 is also much less verbose than SPICE2, in that the only data printed is that requested by the above cards.

Vector names are the same as in nutmeg, with this addition: a name such as @name[param], where name is either the name of a device instance or model, denotes the value of the param parameter of the device or model. See the SPICE3 User's Manual for details of what parameters are available. The value is a vector of length 1. This function is also available with the show command, and is available with variables for convenience for command scripts.

SPICE3 commands are as follows (these are only those commands not also available in nutmeg consult the nutmeg manual page for more commands):

setcirc [circuit name]
Change the current circuit. The current circuit is the one that is used for the simulation commands below. When a circuit is loaded with the *source* command (see below) it becomes the current circuit.

op [.op card args]
Do an operating point analysis.

tran [.tran card args]
Do a transient analysis.

ac [.ac card args]
Do an ac analysis.

dc [.dc card args]
Do a dc transfer curve analysis.

listing [logical] [physical] [deck] [expand]
Print a listing of the current circuit. If the **logical** argument is given, the listing is with all continuation lines collapsed into one line, and if the **physical** argument is given the lines are printed out as they were found in the file. The default is **logical**. A **deck** listing is just like the **physical** listing, except without the line numbers it recreates the input file verbatim (except that it does not preserve case). If the word **expand** is present, the circuit will be printed with all subcircuits expanded.

edit [file]
Print the current SPICE3 deck into a file, call up the editor on that file and allow the user to modify it, and then read it back in, replacing the origonal deck. If a filename is given, then edit that file and load it, making the circuit the current one.

resume
Resume a simulation after a stop.

show Show a device parameter.

alter Alter a device parameter.

state Print the state of the circuit. (This command is largely unimplemented.)

save [all] [output ...] or **.save [all] [output ...]**
Save a set of outputs, discarding the rest. If a node has been mentioned in a **save** command, it

will appear in the working plot after a run has completed, or in the rawfile if spice is run in batch mode. If a node is traced or plotted (see below) it will also be saved. For backward compatibility, if there are **no** save commands given, all outputs are saved.

stop [after n] [when something cond something] ...
Set a breakpoint. The argument **after n** means stop after **n** iteration number **n**, and the argument **when something cond something** means stop when the first **something** is in the given relation with the second **something**, the possible relations being **eq** or = (equal to), **ne** or <> (not equal to), **gt** or > (greater than), **lt** or < (less than), **ge** or >= (greater than or equal to), and **le** or <= (less than or equal to). IO redirection is disabled for the **stop** command, since the relational operations conflict with it (it doesn't produce any output anyway). The **something**s above may be node names in the running circuit, or real values. If more than one condition is given, e.g. **stop after 4 when v(1) > 4 when v(2) < 2**, the conjunction of the conditions is implied.

trace [node ...]
Trace nodes. Every iteration the value of the node is printed to the standard output.

iplot [node ...]
Incrementally plot the values of the nodes while SPICE3 runs.

step [number]
Iterate **number** times, or once, and then stop.

status
Display all of the traces and breakpoints currently in effect.

delete [debug number ...]
Delete the specified breakpoints and traces. The **debug numbers** are those shown by the **status** command. (Unless you do **status > file**, in which case the debug numbers aren't printed.)

reset Throw out any intermediate data in the circuit (e.g, after a breakpoint or after one or more analyses have been done already), and re-parse the deck. The circuit can then be re-run. (**Note**: this command used to be **end** in SPICE 3a5 and earlier versions -- **end** is now used for control structures.) The **run** command will take care of this automatically, so this command should not be necessary...

run [rawfile] Run the simulation as specified in the input file. If there were any of the control cards .ac, .op, .tran, or .dc, they are executed. The output is put in **rawfile** if it was given, in addition to being available interactively.

source file
Read the SPICE3 input file **file**. **Nutmeg** and SPICE3 commands may be included in the file, and must be enclosed between the lines *.control* and *.endc*. These commands are executed immediately after the circuit is loaded, so a control line of *ac* ... will work the same as the corresponding *.ac* card. The first line in any input file is considered a title line and not parsed but kept as the name of the circuit. The exception to this rule is the file *.spiceinit*. Thus, a SPICE3 command script must begin with a blank line and then with a *.control* line. Also, any line beginning with the characters *# is considered a control line. This makes it possible to imbed commands in SPICE3 input files that will be ignored by earlier versions of SPICE. *Note:* in spice3a7 and before, the *.control* and *.endc* lines were not needed, and any line beginning with the name of a front-end command would be executed.

linearize vec ...
Create a new plot with all of the vectors in the current plot, or only those mentioned if arguments are given. The new vectors will be interpolated onto a linear time scale, which is determined by the values of **tstep, tstart,** and **tstop** in the currently active transient analysis. The currently loaded deck must include a transient analysis (a **tran** command may be run interactively before the last **reset**, alternately), and the current plot must be from this transient analysis. This command is needed because SPICE3 doesn't output the results from a transient analysis in the same manner that SPICE2 did.

There are several **set** variables that SPICE3 uses but **nutmeg** does not. They are:
editor
The editor to use for the **edit** command.
modelcard
The name of the model card (normally **.model**).

noaskquit
Do not check to make sure that there are no circuits suspended and no plots unsaved. Normally SPICE3 will warn the user when he tries to quit if this is the case.

nobjthack
Assume that BJT's have 4 nodes.

noparse
Don't attempt to parse decks when they are read in (useful for debugging). Of course, they cannot be run if they are not parsed.

nosubckt
Don't expand subcircuits.

renumber
Renumber input lines when a deck has **.include**'s.

subend
The card to end subcircuits (normally **.ends**).

subinvoke
The prefix to invoke subcircuits (normally **x**).

substart
The card to begin subcircuits (normally **.subckt**).

There are a number of **rusage** parameters available, in addition to the ones available in **nutmeg**:

If there are subcircuits in the input file, SPICE3 expands instances of them. A subcircuit is delimited by the cards .subckt and .ends, or whatever the value of the variables substart and subend is, respectively. An instance of a subcircuit is created by specifying a device with type 'x' ' the device line is written

xname node1 node2 ... subcktname

where the nodes are the node names that replace the formal parameters on the .subckt line. All nodes that are not formal parameters are prepended with the name given to the instance and a ':', as are the names of the devices in the subcircuit. If there are several nested subcircuits, node and device names look like subckt1:subckt2:...:name. If the variable subinvoke is set, then it is used as the prefix that specifies instances of subcircuits, instead of 'x'.

SEE ALSO

nutmeg(1), sconvert(1), spice(1), mfb(3), writedata(3) SPICE3 User's Guide

AUTHORS

SPICE3: Tom Quarles (quarles@cad.berkeley.edu)

nutmeg / User interface: Wayne Christopher (faustus@cad.berkeley.edu)

BUGS

SPICE3 will recognise all the notations used in SPICE2 **.plot** cards, and will translate **vp(1)** into **ph(v(1))**, and so forth. However, if there are spaces in these names it won't work. Hence **v(1, 2)** and **(-.5, .5)** aren't recognised.

BJT's can have either 3 or 4 nodes, which makes it difficult for the subcircuit expansion routines to decide what to rename. If the fourth parameter has been declared as a model name, then it is assumed that there are 3 nodes, otherwise it is considered a node. To disable this kludge, you can set the variable "nobjthack", which will force BJT's to have 4 nodes (for the purposes of subcircuit expansion, at least).

The **@name[param]** notation might not work with **trace, iplot,** etc. yet.

The first line of a command file (except for the .spiceinit file) should be a comment. Otherwise SPICE may create an empty circuit structure.

CAVEATS

SPICE3 files specified on the command line are read in before the .spiceinit file is read. Thus if you define aliases there that you call in a SPICE3 source file mentioned on the command line, they won't be recognised.

VMS NOTES

The standard suffix for rawspice files in VMS is ".raw".

You may have to redefine the value EDITOR if you wish to use the **edit** command, since the default for VMS is "vi".

[Parent Directory]