The following text material is Appendix C of the Fourth Edition of your text – Sedra and Smith, *Microelectronics Circuits*. It is a basic description of the syntax of SPICE input. It should be read in conjunction with example files that are available on the CD accompanying the current edition. The baseline file for the SPICE assignment is also an example that should become clearer with the information given here.

One point about this material: the section on SPICE output is not really applicable to our version of SPICE. Although our version will respond to the commands for .PRINT, .PLOT, etc., that is not the way one would handle the problem of output specification. Omit all output commands, but be sure that you include the line ".PROBE/CSDF" in your input file. This will cause all node voltages and branch currents to be saved. Also, when you need small signal data, use the Analysis/Run Batchprint menu command because that will cause the small signal data to be written in great detail to the output window where you can save it or copy it to a file. One then uses the Display/Vectors menu sequence to open a list of saved data from which you can pick out what you want to view, plot, print, or save to file.

APPENDIX C

An Introduction to SPICE

INTRODUCTION

In this Appendix, we provide a brief introduction to SPICE; specifically, we show how to describe a circuit to SPICE. This material is intended to help the reader to get started on the use of SPICE, but it is *not* an adequate substitute for consulting SPICE manuals and books, some of which are listed in the bibliography section.

Examples of the use of SPICE in the analysis and design of a wide variety of electronic circuits have been presented in the last section of each of Chapters 3–14. There, we also provided some guidance on the appropriate use of SPICE in the circuit-design process. Also, the models employed by SPICE for the three basic semiconductor devices have been described in the SPICE sections of Chapters 3–5.

C.1 WHAT IS SPICE?

SPICE is a computer program that can be used to analyze the operation of an electronic circuit containing a variety of components, e.g., transistors, diodes, resistors, capacitors, etc. Using user-specified device-model parameters, SPICE can perform a dc analysis to determine the dc bias or operating point of each device, can use these to compute the small-signal model parameters of the devices, and use these in turn to compute voltage gain, frequency response, etc. It can also perform a nonlinear transient response of a circuit, e.g., a logic-gate circuit, and a variety of other analysis types.

In order for SPICE to perform the simulation of a given circuit, the user must provide it with the following:

(*a*) *Circuit description:* a complete description of the circuit to be analyzed; its elements and the dc- and signal-sources present, and how they are connected together. Also, the parameter values of the models of the electronic devices utilized.

(*b*) *Analysis requests:* the types of analysis, e.g., dc, small-signal, transient, etc., that the user wishes SPICE to perform.

(*c*) *Output requests:* the type of output required, e.g., a table of dc bias currents and voltages, a plot of the VTC of a logic gate, etc.

In the following we describe how this information is to be conveyed to SPICE. Before we do this, however, we should note that all this information is presented to SPICE in a sequence of lines entered via a computer terminal into a computer file commonly referred to as the **SPICE input deck** or file. The first line in the SPICE input file must be a **title** to identify the circuit being analyzed, and the last line must be an .End statement, to indicate the end of the SPICE input file. The sequence of the remaining lines is arbitrary. It is advisable, however, to add comments throughout the file, to help both the potential reader as well as the creator of the original file. Comment statements are identified by inserting an asterisk as the first character of the comment line. For examples of SPICE input files, the reader is referred to Appendix D which includes a listing of all such files corresponding to the SPICE examples presented in the text. (These are also available on the CD-ROM and can be downloaded from the Web site.)

C.2 CIRCUIT DESCRIPTION

Each element in the circuit is specified by an **element statement** containing the element name, the circuit nodes to which it is connected, and the electrical parameter value(s). The first letter of an element name denotes element type (for example, R for resistor), and the name can be from one to eight alphanumeric characters. Nodes are specified by nonnegative integers but need not be numbered sequentially. The datum node (ground) must be numbered 0. Every node must have at least two connections, except for MOSFET substrate nodes and unterminated transmission lines. Basic (but incomplete) specification formats are given in Table C.1, where:

Table C.1 SYNTAX OF ELEMENT STATEMENTS

Component	Name	Nodes and value		
Resistor	Rxxxxxxx	N+	N−	VALUE
Capacitor	Cxxxxxxx	N+	N−	VALUE
Inductor	Lxxxxxxx	N+	N−	VALUE
VCCS	Gxxxxxxx	N+	N−	NC + NC − VALUE
VCVS	Exxxxxxx	N+	N−	NC + NC − VALUE
CCCS	Fxxxxxxx	N+	N−	VNAM VALUE
CCVS	Hxxxxxxx	N+	N−	VNAM VALUE
Voltage source	Vxxxxxxx	N+	N−	QUAL
Current source	Ixxxxxxx	N+	N−	QUAL

1. The component name begins with a particular letter as indicated, and is from one to eight alphanumeric characters long.
2. N+ and N− indicate the nodes connected, the first being positive (if that matters). Note in particular that the current of a current source flows from N+ to N−.
3. VALUE is in units of ohms, farads, henries, A/V, V/V, A/A, V/A, respectively, for the first seven components above. For convenience, unit prefixes can be used, as indicated in Table C.2.

Table C.2 SCALE-FACTOR ABBREVIATIONS RECOGNIZED BY SPICE

Power-of-Ten Suffix Letter	Metric Prefix	Multiplying Factor
T	tera	10^{+12}
G	giga	10^{+9}
Meg	mega	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}

4. NC+ and NC− are nodes across which the controlling voltage appears.
5. VNAM is the voltage source through which the controlling current flows.
6. QUAL is a set of qualifiers of the source, whether DC or transient (including pulse, sinusoid, exponential or piecewise-linear) with amplitudes and other qualifiers, or AC with magnitude and phase.
7. A voltage source of zero volts is used as a means to indicate the location of a current measurement.

As an example, a 6.8-kΩ resistor with the name R_{B2}, connected between nodes 4 and 5 can be described to SPICE with the element statement:

RB2 4 5 6.8K

As another example, a 1.0-μF capacitor (C_{C1}) connected in a circuit between nodes 3 and 4, and having an initial dc voltage of 5 V can be described by the statement:

CC1 3 4 1.OU IC = 5

where IC denotes "initial condition." As a final example, a voltage-controlled voltage source representing the differential gain of an op amp can be described by

EOUT 3 0 2 1 100K

where EOUT is used to denote the VCVS whose output terminal is at node 3 and is referenced to ground (node 0), whose input terminals are at nodes 1 and 2, and whose control ratio is 10^5 V/V.

Describing a semiconductor device to SPICE requires both an element statement and a **model statement.** Table C.3 displays the syntax of the element statements for the diode, the BJT and the MOSFET. The following comments are in order:

Table C.3 ELEMENT STATEMENTS FOR SEMICONDUCTOR DEVICES

Device	Name	Nodes and models				
Diode	Dxxxxxxx	N+	N−			MNAME AREA
BJT	Qxxxxxxx	NC	NB	NE	NS	MNAME AREA
MOSFET	Mxxxxxxx	ND	NG	NS	NB	MNAME L W

1. The statement begins with the name of the device, with the first letter of the name indicating the type of the device.
2. For a diode, N+ is the node to which the anode is connected, and N− is that to which the cathode is connected.
3. For a BJT, NC, NB, NE and NS are the circuit nodes to which the collector, base, emitter, and (for IC devices) substrate are connected.
4. For a MOSFET, ND, NG, NS, and NB are the circuit nodes to which the drain, gate, source and body (substrate) are connected.
5. MNAME refers to the name of the model for this particular device (e.g., Q2N2222A for a BJT of the 2N2222A type). The parameter values of the model are to be specified in a separate model statement; see below.
6. AREA is an (optional) area scaling factor; it is the number of diodes or BJTs of this type that are to be connected in parallel to form this particular device.
7. L and W are the MOSFET channel length and width, in meters.

Finally, we show in Table C.4 the syntax of the model statements for diodes, BJTs and MOSFETs.

Table C.4 SYNTAX OF MODEL STATEMENTS

Device	Model Statement
Diodel	.Model MNAME D(IS = . . . n = . . . , etc.)
BJT	.Model MNAME NPN (or PNP)(IS = . . . βF = . . . , etc.)
MOSFET	.Model MNAME NMOS (or PMOS)(kP = . . . Vt0 = . . . , etc.)

Here again, MNAME refers to the model name. Obviously, for every device type used in the circuit there must be a model statement, specifying the parameter values of the model to be used for this particular device. A number of devices (say, BJTs) of the same type need only one model statement. Following a device identifier (D for diode, NPN for an *npn* BJT, PNP for a *pnp* BJT, etc.) a listing is provided of the values of the model parameters. For further information on the SPICE models, the reader is referred to the SPICE section in Chapters 3, 4, and 5.

C.3 ANALYSIS REQUESTS

Once a circuit is described to SPICE via an input file, the user must then specify the analyses required in the simulation. There are three main choices: DC operating point, AC frequency response, and transient response. Table C.5 shows their syntax plus that of the DC sweep command. Notice that each of these commands begins with a dot (.), which tells SPICE that the line is a command line requesting action, and not part of the circuit description.

Table C.5 MAIN-ANALYSIS COMMANDS

Analysis Requests	Spice Command
Operating-point	.OP
DC sweep	.DC *source_name start_value stop_value step_value*
AC frequency response	.AC DEC *points_per_decade freq_start freq_stop*
	.AC OCT *points_per_octave freq_start freq_stop*
	.AC LIN *total_points freq_start freq_stop*
Transient response	.TRAN *time_step time_stop* [*no_print_time max_step_size*] [UIC]

The DC operating-point command, .OP, results in all the DC node voltages and branch currents and the power dissipation of all DC sources. The .OP command automatically prints the calculation results in the output file.

Although the DC transfer characteristic can be determined by running repeated .OP commands for various values of an input dc source, SPICE provides an alternative; namely, a DC sweep command (.DC) that performs this operation automatically. The syntax of this command includes the name of the DC source to be varied (*source_name*) beginning at the value marked by *start_value* and increased or decreased in steps of *step_value* until the final *stop_value* is reached.

With the AC frequency response command (.AC), SPICE performs a linear small-signal frequency response analysis. It automatically calculates the DC operating point of the circuit, thereby establishing the small-signal equivalent circuit of all nonlinear elements. The linear small-signal equivalent circuit is then analyzed at frequencies beginning at *freq_start* and ending at *freq_stop*. Points in between are spaced logarithmically, either by decade (DEC) or octave (OCT). The number of points in a given frequency interval is specified by *points_per_decade* or *points_per_octave*. We can specify a linear frequency sweep (LIN) and the total number of points in it by *total_points*. We usually use a linear frequency sweep when the bandwidth of interest is narrow and a logarithmic sweep when the bandwidth is large.

Finally, with the transient response command (.TRAN), SPICE computes the circuit variables as a function of time over a specified time interval. The time interval begins at time $t = 0$ and proceeds in linear steps of *time_step* seconds until *time_stop* seconds is reached. Although all transient analysis must begin at $t = 0$, we have the option of delaying the printing or plotting of the output results by specifying the *no_print_time* in the third field enclosed by the square brackets. This is a convenient way of skipping over the transient response of a network and viewing only its steady-state response.

Before the start of any transient analysis, SPICE must determine the initial values of the circuit variables, usually from a DC analysis of the circuit. If the optional UIC (use initial conditions) parameter is specified on the .TRAN statement, SPICE will skip the DC bias calculation and instead use only the IC = information supplied on each capacitor or inductor statement. All elements without an IC = specification are assumed to have an initial condition of zero.

C.4 OUTPUT REQUESTS

Circuit simulation produces a lot of data, and it would be impractical to pass all of it on to the user. Instead, SPICE provides display features that enable us to specify which circuit variables we want to see and the best format for them. This is much like placing a measurement probe on the node of interest. Table C.6 lists the syntax of print and plot formats.

Table C.6 SPICE OUTPUT REQUESTS

Output Requests	Spice Command
Print data points	.PRINT DC *output_variables*
	.PRINT AC *output_variables*
	.PRINT TRAN *output_variables*
Plot data points	.PLOT DC *output_variables* [(*lower_plot_limit, upper_plot_limit*)]
	.PLOT AC *output_variables* [(*lower_plot_limit, upper_plot_limit*)]
	.PLOT TRAN *output_variables* [(*lower_plot_limit, upper_plot_limit*)]

Notes:
1. SPICE *output_variables* can be a voltage at any node V(*node*), the voltage difference between two nodes V($node_1$, $node_2$), or the current through a voltage source I(V*name*).
2. AC *output_variables* can also be
   Vr, Ir: real part
   Vi, Ii: imaginary part
   Vm, Im: magnitude
   Vp, Ip: phase
   Vdb, Idb: decibels
3. PSPICE provides a greater flexibility for specifying *output_variables*.

The .PRINT command prints out variables in tabular form as a function of the independent variable associated with the analysis. With it, we must also specify the analysis (i.e., DC, AC, or TRAN) for which the specified outputs are desired. Next, we specify a list of voltage or current variables (denoted as *output_variables*.) Generally, a voltage variable is specified as the voltage difference between two nodes, say $node_1$ and $node_2$, as V($node_1$, $node_2$). When one of the nodes is omitted, it is assumed to be the ground node (0).

SPICE allows only those currents flowing through independent voltage sources to be observed. Such a current would be specified by I(V*name*) where V*name* is the name of the independent voltage source through which the current is flowing. If we wish to observe a particular branch current without a voltage source, then we add a zero-valued voltage source in series with this branch and request that the current flowing through this source be printed or plotted.

For a DC analysis, the variables printed are the node voltages or branch currents computed as a function of a particular DC source in the network.

For an AC analysis, the output variables are sinusoidal or phasor quantities as a function of frequency and are represented by complex numbers. SPICE accesses these results in the form of real and imaginary numbers or in magnitude and phase form. Magnitude can also be expressed in terms of dBs when convenient. To access a specific variable type, Table C.6 shows how a suffix is appended to the letter V or I.

The results of a TRAN analysis are the node voltages or branch currents computed as a function of time.

SPICE's graphical feature generates a simple line plot from the list of output variables as a function of the independent variable. The syntax for the plot command is identical to that of the print command, and the .PRINT keyword is replaced by .PLOT.

C.5 A FINAL REMARK

To gain experience in writing SPICE programs, the reader is advised to follow the SPICE listings in Appendix D against the corresponding circuits in the Examples presented in the last sections of each of Chapters 3–14. The next step, of course, is to attempt SPICE simulations for other circuits in this book. The end-of-chapter problems present a wealth of circuits for this purpose.

BIBLIOGRAPHY

M. Rashid, *SPICE for Circuits and Electronics Using PSPICE,* Englewood Cliffs, N.J.: Prentice-Hall, 1990.

G.W. Roberts and A. Sedra, *SPICE,* New York: Oxford University Press, 1992 and 1997.

P. Tuinenga, *A Guide to Circuit Simulation and Analysis Using PSPICE,* 2nd ed., Englewood Cliffs, N.J.: Prentice-Hall, 1992.

A. Vladimirescu, *The SPICE Book,* New York: John Wiley, 1994.