The circuit that results when we set b_2 and b_0 to zero is shown in Figure 4.32d. The equivalent resistance of the resistors in both boxes is $2R$. We must be careful when we make this simplification because we lose direct information about V_0. However, by the voltage divider rule, we know that V_0 will be one-half the voltage across the equivalent $2R$ resistance. The simplified equivalent circuit for this case looks just like that in Figure 4.32c if we replace b_2 by b_1 and V_0 by $2V_0$. Therefore, another application of the voltage divider formula results in $V_0 = b_1/6$. The circuit that results when we set b_2 and b_1 to zero is shown in Figure 4.32e. The part of the circuit enclosed in the box is another $R-2R$ ladder. Again we lose direct information about V_0 when simplifying, but careful analysis will yield the expression $V_0 = b_0/12$.

The final result comes from summing up the three individual answers, $V_0 = (4b_2+2b_1+b_0)/12$, and gives us the proper conversion formula (with a proportionality constant). ■

4.7 An Introduction to Electric Circuit Simulation with MicroSim PSpice

The use of computer simulation tools for the analysis of electric and electronic circuits pervades the modern workplace. These codes are typically employed when the circuits contain a sufficient number of elements to render impractical the computation of circuit voltages and currents by hand. This is the case for many circuits of interest. While most of these circuits can be broken down into subcircuits that can be analyzed by hand, analysis of the performance of the complete circuit is often of paramount importance and can be done only via computer. These simulation tools are also useful for investigating circuits of any size which have nonlinear elements. Furthermore, they can also be used to evaluate the dependence of output quantities on the parameter tolerances or the nonideal properties of various circuit elements.

In this book we will make use of numerical simulations only to examine the relatively small, linear, electric circuits that we can also analyze by hand. The purpose of this is threefold. First, because of the prevalent use of simulators in industry, you should have a working knowledge of at least one circuit simulation package and have a general understanding of their capabilities. Second, once you have the ability to simulate circuits, you can use the codes to check your hand calculations and perhaps even to investigate deviations from ideal circuit performance. Finally, if you utilize a code which contains a good graphics package (like the one described here), you can readily obtain visual information about any circuit variable in either the time or the frequency domain.

It cannot be overemphasized that **a circuit simulator is not a substitute for the knowledge of basic circuit theory principles.** It will not design circuits from scratch for you. It will not troubleshoot a circuit for you if it is not performing according to expectations. Perhaps most important, the code will likely, without hesitation or complaint, give you incorrect or unreasonable answers if you give it incorrect or unreasonable input data. The only way you can trust the simulation results is to

have a solid understanding of Kirchhoff's laws, terminal relations, and the rest of the electric circuit theory concepts.

In this section we will give a brief introduction, via several examples, to the operation of the MicroSim PSpice circuit simulator and auxiliary routines. At the end of Chapters 6 through 10 you will find examples and homework problems which are based on this simulator. PSpice was chosen for two reasons. First, it is based on the SPICE2 simulator developed at University of California Berkeley in the 1970s, which is the basic engine (albeit often modified) for a number of simulators that are widely used in industry. Second, there is available an evaluation copy of MicroSim's family of products, which is comprised of a schematic generator, the PSpice simulator for analog and digital components, and the PROBE graphics package. This software is available free to professors and can be copied for students with the encouragement of the manufacturer, MicroSim Corporation.[2] This discussion is based on the IBM-PC compatible version 6.2 for Microsoft Windows. Furthermore, there are often site licenses for the regular SPICE simulator on the various workstations that are typically available to students. If one is familiar with the operating system and can locate the SPICE code, this introduction should be sufficient to enable one to simulate circuits in that environment as well.

This introduction is by no means a substitute for an extensive operating manual for PSpice. It contains only a bare-bones description designed to give the user the ability to perform the basic operations of circuit simulation. There are a number of excellent texts dedicated to PSpice simulation. Several of them are listed in Appendix C. The Reference guide from MicroSim Corporation is also indicated there. However, after mastering the basics, one can also learn more about the code's capabilities by roaming through the various drop-down menus and experimenting with the various options.

Before proceeding, the evaluation copy must be loaded onto an IBM-PC. Version 6.2 is usually supplied on five 3 1/2 inch 1.44 MB floppy disks or a CD-ROM. Installation follows the usual procedure for any Windows 3.1 software.[3] Disk 1 is loaded into the floppy drive and the setup.exe program on that disk is executed either by double-clicking on the "setup.exe" listing in the File Manager or by using the "Run" command on the "File" drop-down menu in the Program Manager. (Windows 95 users can use the Windows Explorer or the control panel to load the program.) Follow the directions that appear on the screen to load the software, generate a Program Group entitled "MicroSim Eval 6.2," and make any necessary changes to your autoexec.bat file. The files will nominally be saved in a directory called "MSIMPR62." The pro-

[2] Submit a request on educational letterhead to: Product Marketing Department, MicroSim Corporation, 20 Fairbanks, Irvine, CA 92718. Their phone number is (714) 770-3022.

[3] Users of older versions of Windows may have to also install "Win32s" in order to run PSpice. This code is also supplied on the PSpice CD-ROM and can be transferred to two floppy disks for distribution. Alternatively, one can use an older version of MicroSim's products. Version 6.0, for example, has most of the Version 6.2 features and runs without "Win32s."

gram group should contain at least five items: 1) MicroSim Schematics, 2) MicroSim PSpice, 3) MicroSim Probe, 4) MicroSim Stimulus Editor, and 5) MicroSim Parts. There will also be a "README" file which has some information about the latest versions of the codes. We will always begin our examples by double-clicking on the Schematics icon; all of the other programs can be invoked at the proper time from the Schematics window. (Unless otherwise stated, "clicking" always refers to pressing the left button on a two-button mouse.)

The first circuit we will analyze is the current divider of Example 4.5. The first step will be to draw the schematic of the circuit. Then we must select the types of analyses that we would like PSpice to perform. In this example we claim that this circuit functions as a low-pass (frequency) circuit, so we will plot the current through the resistor as a function of frequency. Third, we must run the PSpice simulator. Finally, we will use Probe to draw the resistor current over the specified frequency range.

We begin by opening the "MicroSim Eval" Program Group in Windows and double-clicking on the Schematics icon. (Windows 95 (or later) users must use the start button and move through the various menus to arrive at the "schematics" label.) From the drop-down menus at the top of the window, click on "Draw" and then on "Get New Part..." A window entitled "Add Part" will appear and in the dialog box you should enter "C" and then press the "OK" button (by clicking on it). The box will disappear and the cursor arrow will have a capacitor attached to it. Move the capacitor to a desirable location and place it by clicking the left mouse button. You can continue to add capacitors by repeating the last two operations. Click the right mouse button to stop adding capacitors. If you have placed any parts unintentionally, you can delete them by clicking on them (they change color to show that they have been selected) and then hitting the delete key. Next click the "Edit" drop-down menu and select "Rotate" to get the capacitor vertical on the page.

Default values for the capacitor's name and capacitance value are indicated in the schematic. The capacitor's name is typically *Cn*, where *n* is an ascending integer. The name can be changed by double-clicking on it so that the "Edit Reference Designator" dialog box appears. Type in a new name and hit "OK." Double-click on the capacitance (whose default is typically 1 nF) to invoke the "Set Attribute Value" dialog box. For this example set the capacitance to 1 F by entering a "1" (farad is assumed — if you type 1 F it will assume that you meant one femtofarad). The multiplying factors that can be entered to scale the units are essentially the same as those listed in Table 1.1. Two exceptions are that "u" is used instead of "μ" to mean 10^{-6} and "MEG" must be used to represent 10^6 because "m" and "M" are both interpreted by PSpice to mean 10^{-3}.

Add a resistor to the circuit by repeating the steps above for the capacitor, except that in the "Add Part" dialog box you should enter "R" for a resistor. Change the resistance to "1" (ohms are assumed) and move the part parallel to the capacitor by "dragging" it with the mouse (holding the left mouse button down while moving the mouse). Connect the two parts in parallel using the "Draw" drop-down menu and selecting "Wire." Click once after moving the pointer to the top of the capacitor and once again at the top of the resistor. Note that the wires are only drawn orthogonally and that they "snap" to the grid points. You can change this in the drop-down

"Options" menu by selecting "Display Options," but the settings are adequate for our purposes. You can select "Repeat" from the "Draw" menu to place the wire between the two lower ends of the elements.

It is important to note that PSpice will automatically number the nodes in a circuit, but that the user must supply the ground node in order for the simulation to run. You do this by selecting "AGND" in the "Add Part" dialog box. Connect this ground to the base of the capacitor. The final element of this circuit is the current source. The notation for this element in the "Add Part" dialog box is "ISRC." Rotate it twice to get the direction arrow pointing up and then add two wires to place it in parallel with the other elements. To set the parameters of the current source, double-click on the arrow to bring up the window which is titled "I1 Part name: ISRC." There are many options that can be adjusted. Because we want to perform an ac analysis, click on the line that says "AC=." There are two boxes at the top of the window. The left one is entitled "Name" and "AC" should appear in that box. Click somewhere in the box on the right, enter a "1" for one ampere, and press the "Save Attr" button. Finally, press the "OK" button. The circuit has now been completely drawn and should look like the circuit in Figure 4.33.[4] Note that you can use the "Text" option of the "Draw" drop-down menu if you want to add comments to your circuit.

You are now ready to save this schematic on disk. This is accomplished by using the "File" drop-down menu and clicking on the "Save" option. The usual file extension for Schematics drawings is *.SCH; we will call our schematic example1.sch. The schematic can be printed (if a printer is connected to your computer) by selecting "File" and "Print." After selecting any options you want in the window that appears, press the "OK" button to initiate the hardcopy generation.

Once the file is saved, go to the "Analysis" drop-down menu and click on "Create Netlist." This will convert the schematic to the tabular form that is required by PSpice. Also in the "Analysis" menu you can click on "Examine Netlist" to view the data in this form. This will launch the Windows Notepad editor to allow you to examine

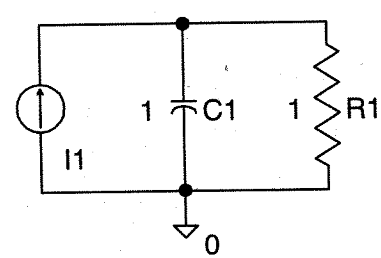

Figure 4.33: The PSpice schematic for Example 4.5.

[4] In this edition, it was not possible to incorporate the PSpice screen images directly into the text. Instead, graphics packages were used to imitate the PSpice output. Consequently, there will be slight differences occasionally between the figures and your computer screen images.

Table 4.1: The circuit Netlist for Example 4.5.

Schematics Netlist			
C_C1	0	N_0001	1
R_R1	0	N_0001	1
I_I	0	$N_0001 AC	1

the Netlist, which has been saved on disk as example1.net. The result is shown in Table 4.1. For the capacitor and the resistor (the second and third rows in the Netlist, respectfully), the first column is the name of the element, the next two entries are the nodes to which the elements are connected, and the fourth and final entry is the value (capacitance and resistance, respectfully). The first node entered is always considered the "plus" node in terms of the assumed reference voltage polarity. *Note that PSpice uses the passive sign convention for all elements, including sources!* The plus node in this example is always the ground node and is indicated by a zero. The only other node in this example is given the name $N_0001. The final row in the Netlist represents the current source. The columns are similar to those of the passive elements except that an "AC" precedes the current value. We could have also assigned currents for dc analysis, transient analysis, etc. in the appropriate dialog box and they would have been indicated on this line as well.

Before PSpice can be executed we must indicate the type of analyses to be run. Go to the "Analysis" drop-down menu and select "Setup." In the "Analysis Setup" dialog box that appears, normally only "dc Bias Point Detail" is selected, as indicated by the "x" in the box to the left of that button. Calculation of the dc bias point is always the first step of a PSpice run. Enable the ac analysis by clicking on the box to the left of "AC Sweep" so that an "x" appears. Next, push the ac sweep button. A window will appear entitled "AC Sweep and Noise Analysis" that will allow us to modify the parameters of the frequency sweep. First, change the "AC Sweep Type" to "Decade" by clicking in the appropriate circle. Next, in the "Sweep Parameter Box" set the "Pts/Decade" to 21, the "Start Freq." to 0.01 (Hz), and the "End Freq." to 100. Press the "OK" button at the bottom of that window. Finally, press the "Close" button in the "Analysis Setup" window to return to the schematic. We are now ready to run PSpice.

From the "Analysis" drop-down menu select "Simulate." First a small window entitled "Schematics" will appear announcing that the Netlist is being generated (unless this has already been done). Next, a "PSpice" window will appear that will update you on the status of the simulation. First it will say that the program is reading and checking the circuit. If any errors are encountered, an error message will be printed in the window and the simulation will stop. The user must then determine the error and correct it before attempting another simulation. If there are no errors, PSpice will proceed with the bias point calculation and then move on to the AC analysis.

After the successful completion of the simulation, the "PROBE" program will be initiated automatically. There are two points to note. First, even though "PROBE" usually comes up "full screen," the schematics generator, PSpice, and PROBE are all running at the same time in different windows and one can move back and forth between them as necessary. Second, PROBE need not be launched automatically after PSpice is run; this option is selected in the "Analysis" drop-down menu under "Probe Setup. . . ." When the PROBE window appears, the frequency axis is drawn but no curves are drawn. Notice that the independent axis has a log scale as requested in an earlier dialog box. Select the "Trace" drop-down menu and click on "Add." In the window that appears, double-click on "I(R1)." The dependence of the resistor current on frequency is then displayed, along with a dependent axis scale and a legend. The current is nearly equal to the source current (of 1 A) at low frequency and goes to zero as the frequency is increased. This is the principal characteristic of the low-pass filter. Note that we can delete any trace if necessary by clicking on the appropriate name in the legend (its color will then change) and hitting the delete key. There are many modifications that we could make to this plot, but we will just modify the labels and leave other changes for later examples. Click on the "Edit" drop-down menu and select "Modify Title. . . ." In the resulting dialog box enter "Example 1" and press "OK." From the "Plot" drop-down menu select "Y Axis Settings," type "Resistor Current" in the "Axis Title" box, and then hit "OK." Selecting "File" and then "Print" will bring up a window that can be used to generate a hard copy. This graph is shown in Figure 4.34.

We conclude this rather long example with a brief discussion of the files that are generated during a typical circuit simulation. The files example1.sch and exam-

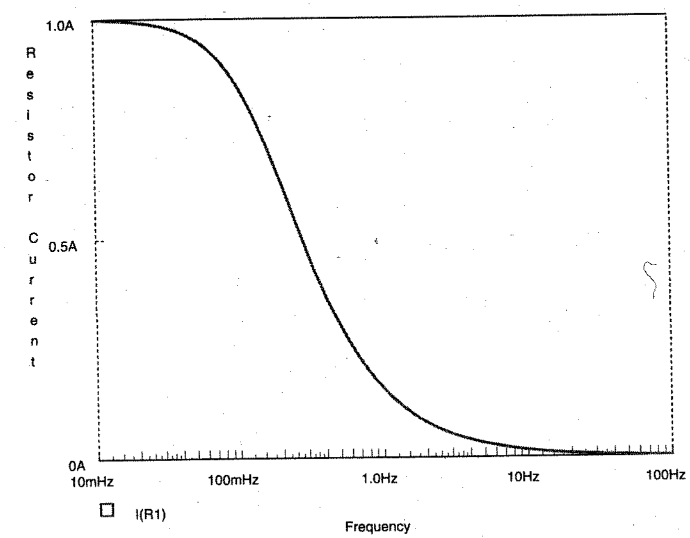

Figure 4.34: The probe output for Example 4.5.

Table 4.2: The PSpice input file for Example 4.5.

```
* D:\ CIRCUITS\ EXAMPLE1.SCH
* Schematics Version 6.2 - April 1995
* Sun Jun 25 15:31:14 1995
** Analysis setup **
.ac DEC 21 .01 100
.OP
* From [SCHEMATICS NETLIST] section of msim.ini:
.lib nom.lib
.INC "EXAMPLE1.net"
.INC "EXAMPLE1.als"
.probe
.END
```

le1.net have already been described. The input to PSpice is actually read from the file xample1.cir, a listing of which is given in Table 4.2. The first line is automatically a omment line and gives the name of the schematic file. Additional comment lines are enerated by placing an asterisk in the first position. Comments can also be placed at ie end of input lines by separating them from the data by a semicolon. The first two oncomment lines tell PSpice what types of analyses to run. The first line initiates ie ac analysis and the trailing parameters indicate the method of determining the equency points. They come directly from our earlier dialog box entries. The next ne tells PSpice to find the dc operating point. The line which begins with ".lib" idicates the name of a library that will be used to specify the parameters of various components. This is discussed in detail at the end of Chapter 7 after diodes are itroduced. The two lines beginning with ".inc" tell PSpice to read in the data from ie two files indicated. We have already seen the Netlist file. The other file contains iases, i.e., a list of alternate ways to identify nodes. We will examine this file in the ext example. The second to last line invokes the PROBE program at the end of the mulation and the final line identifies the end of the PSpice data.

The final file that we will point out is the example.out file. This file contains e output from the PSpice run and can be loaded into Notepad from the schematics indow via the drop-down "Analysis" window by clicking on "Examine Output." can also be examined from the PSpice window from the "File" drop-down menu / clicking on "Examine Output." Scrolling through it, one can find a listing of the put information as well as some details of the simulation. Error messages that are enerated if the input file is not correct will be given in this file. One curious point is at next to total power dissipation the output shows 0.00 W. This is because PSpice lculates only the power dissipated by voltage sources. Another thing that you may otice is that nowhere are the frequency dependences of the circuit voltages and irrents to be found in this output file. These numbers are stored in another file that typically readable by PROBE, but they are not in a simple text format. If we want generate a table of the resistor current as a function of frequency in the output file,

for example, we must insert the line: ".PRINT AC I(R_R1)" into the example.cir file somewhere before the ".END" statement and rerun PSpice. Notepad, or any other editor that is at your disposal, can be readily used to accomplish the required change.

Next, we will use PSpice to analyze the circuit from Example 4.11. We will perform a transient analysis on this circuit and compare the simulations with the analytic results. The circuit redrawn by the schematic editor is shown in Figure 4.35. The resistors, the capacitor, and ground are added and their values are modified as described in the previous example. The inductors are added by selecting "L" in the "Add Part" dialog box. Their values (in henries) are modified by double-clicking on the default values in the schematic.

The name for the voltage sources in the "Add Part" dialog box is "VSIN." By double-clicking on the left voltage source one brings up a window entitled "V1 Part Name: VSIN." Click on the "Voff =" line and then move to the upper right box and type a "0" and push the "Save Attr" button. This sets the source's dc offset to zero. Next, click on the "Vampl =" line, type a "1" in the box, and hit "Save Attr" to set the sine wave amplitude to 1 V. Repeat the steps outlined above to set the frequency to Freq = 0.31831 (2 rad/s) and the phase to 90°. Note that zero degrees gives a sine function for the source, so 90 is required to achieve a cosine dependence. When adding the right voltage source, be sure to rotate it twice to get the positive terminal facing downward. For this element we need to enter Voff = 0, Vampl = 1.414213 ($\sqrt{2}$), Freq = 0.31831, and Phase = 45°. The Netlist for this problem is shown in Table 4.3. There is a total of six nodes (including ground) because PSpice must label even the trivial ones.

To select the analysis type we click on "Setup" in the "Analysis" drop-down menu. We enable the Transient analysis by clicking on the box to the left of the "Transient. . . " button. Then, by depressing this button a window appears that allows us to define the transient analysis. Click on "Skip initial transient solution." Because the period is about 3.14 seconds, set the "Final Time" to 10 (seconds). That will allow over three full periods to be simulated. For a total of about 200 printed points, select the "Print Step" to be 50m. Then click the "OK" button in that window and the "Close" button in the "Analysis Setup" window. Save this circuit as example2.sch. We are now ready to run the second simulation.

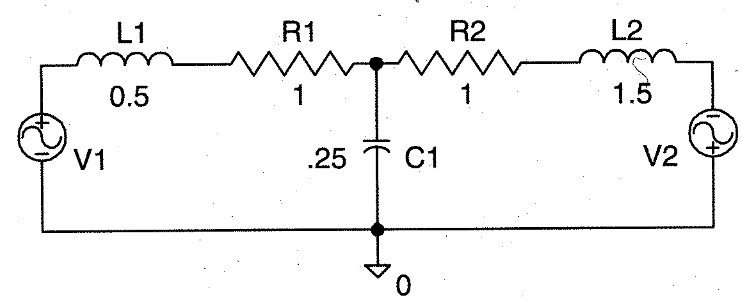

Figure 4.35: The PSpice schematic for Example 4.11.

Table 4.3: The circuit Netlist for Example 4.11.

R_R1	N_0002	N_0001	1			
R_R2	N_0001	N_0003	1			
L_L1	N_0004	N_0002	0.5			
L_L2	N_0003	N_0005	1.5			
C_C1	0	N_0001	0.25			
V_V1	N_0004	0				
+SIN	0	1	0.31831	0	0	90
V_V2	0	N_0005				
+SIN	0	1.414213	0.31831	0	0	45

From the "Analysis" drop-down menu select "Simulate." The usual sequence of windows will come and go and, if there are no errors, the "PROBE" program will be initiated automatically. The input file for PSpice is displayed in Table 4.4, and the alias list is given in Table 4.5. The only new input in the example2.cir file is the directive for the transient analysis, which includes the print step, the final time, and an instruction to skip the initial conditions. The alias file just gives a list of alternate expressions for the nodes in terms of the circuit elements. For example, the first line after the .ALIASES line means that R1:1 should be interpreted as $N_0002 and R1:2 is the same as $N_0001. In the next line and in the seventh line, R2:1 and C1:2 are also made equivalent to $N_0001, respectively, since they are also connected to that node.

When the PROBE window appears, the time axis is drawn but no curves are present. Select the "Trace" drop-down menu and click on "Add." In the window that appears, one can select time, five voltages that correspond to each of the nonzero node potentials via aliases, a current through any of the passive or active elements, or the ground potential. One can also select combinations of the variables. For example, the voltage across the resistor R1 can be plotted as follows. Click on the "Trace Command" line and then enter "V(R1:1)-V(R1:2)" and press "OK." We can

Table 4.4: The PSpice input file for Example 4.11.

```
* D:\ CIRCUITS\EXAMPLE2.SCH
* Schematics Version 6.2 - April 1995
* Mon Jun 26 02:29:11 1995
** Analysis setup **
.tran 50m 10 SKIPBP
.OP
* From [SCHEMATICS NETLIST] section of msim.ini:
.lib nom.lib
.INC "EXAMPLE2.net"
.INC "EXAMPLE2.als"
.probe
.END
```


Table 4.5: The PSpice alias file for Example 4.11.

```
* Schematics Aliases *
.ALIASES
R_R1          R1(1=$N_0002 2=$N_0001 )
R_R2          R2(1=$N_0001 2=$N_0003 )
L_L1          L1(1=$N_0004 2=$N_0002 )
L_L2          L2(1=$N_0003 2=$N_0005 )
C_C1            C1(1=0 2=$N_0001 )
V_V1            V1(+=$N_0004 -=0 )
V_V2            V2(+=0 -=$N_0005 )
.ENDALIASES
```

plot the analytic result for comparison as follows. Select the "Trace" drop-down menu again and click on "Add." Click on the "Trace Command" line and then enter "0.2357 * cos(2 * time − 2.35619)" and then press "OK." [The analytic result is $v_{R_1}(t) = (\sqrt{2}/6)\cos(2t - 135°)$.]

Probe can be used to make multiple plots. Click on "Add Plot" in the "Plot" drop-down menu. Repeat the above sequences, but enter "V(R2:1)-V(R2:2)" to plot the voltage across R2. The analytic result for R2 is $v_{R_2}(t) = (2/3)\sin(2t)$. The resulting plots are shown in Figure 4.36. Note that the curves do not agree well at early times but are nearly indistinguishable at later times. The difference between the curves is called the transient response and is the subject of Chapter 7.

Our final example will be that of the three-stage digital-to-analog converter that was analyzed in Example 4.14. We will perform only a detailed bias point analysis, but we will use the "Parametric" analysis to change the voltage output of some of the sources. The circuit as it is redrawn in PSpice is shown in Figure 4.37. The resistors and the ground are drawn and the resistances are modified as in the previous examples. The name for the voltage sources in the "Add Part" dialog box is "VSRC." By double-clicking on the left voltage source one brings up a window entitled "V1 Part Name: VSRC." Click on the "DC =" line then move to the upper right box and type "{Vt}" and push the "Save Attr" button and then "OK." Vt will be the name of the parameter that we will assign two values to: 0 V for "off" and 5 V for "on." Symbolic values will not work properly without the braces {}. Repeat the above procedure for the middle voltage source. Set the "DC =" voltage on the rightmost voltage source to 5 V. Don't use braces around the 5, of course, since it is a numeric value.

From the "Analysis" drop-down menu select "Setup." Click the box to the left of the "Parametric" button to place an "x" and activate this sweep. Press the "Parametric" button and a window will appear that will allow us to define Vt. Set the "Swept Var. Type" to "Global Parameter" and set the "Sweep Type" to "Value List" by clicking on the circle to the left of each name (or on the name itself). Click on the "Name" box in the upper right corner and enter "Vt." Click on the "Value" box in the lower right corner and enter "0, 5." Finally, hit the "OK" and "Close" buttons and save the

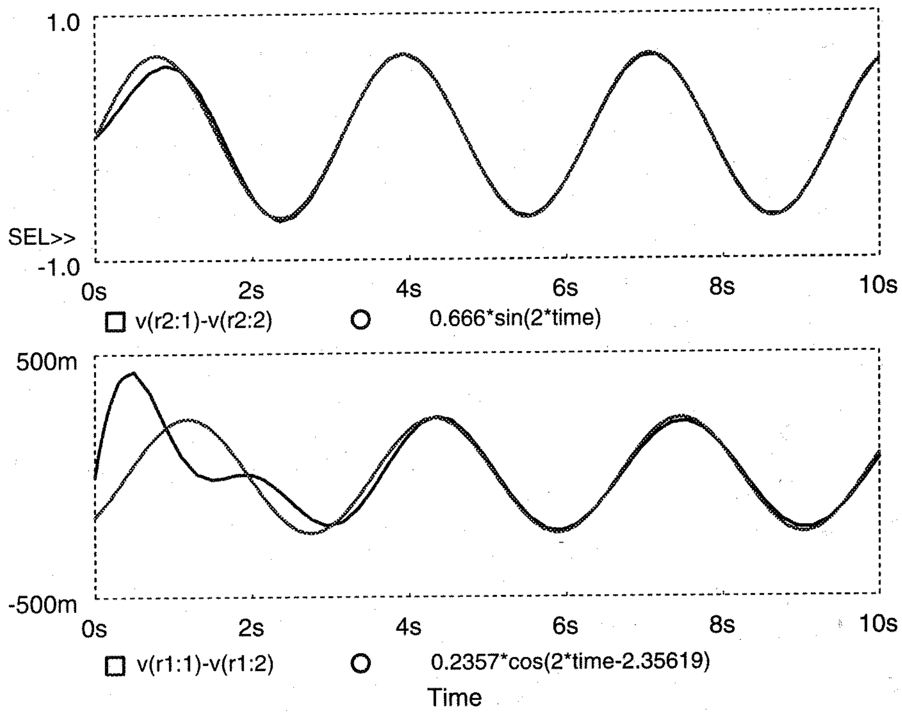

Figure 4.36: The PROBE output for Example 4.11.

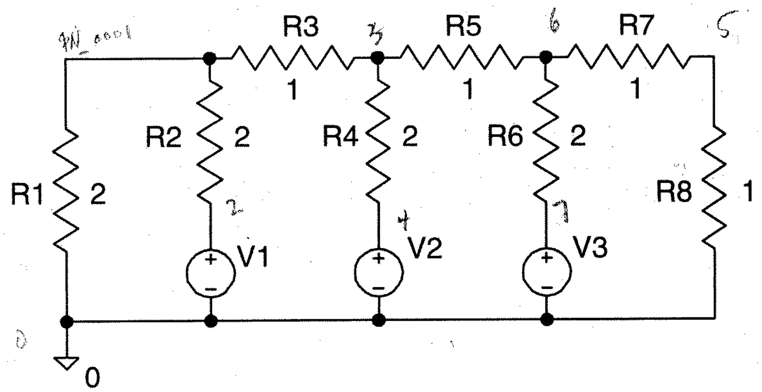

Figure 4.37: The PSpice schematic for Example 4.14.

Table 4.6: The analysis and Netlist sections of the output file for the PSpice Example 4.14.

```
** Analysis setup **
.STEP PARAM Vt LIST
+ 0,5
.OP
* Schematics Netlist *
R_R1        0 $N_0001            2
R_R2        $N_0002 $N_0001      2
R_R3        $N_0001 $N_0003      1
R_R4        $N_0004 $N_0003      2
R_R5        $N_0003 $N_0005      1
R_R6        $N_0006 $N_0005      2
R_R7        $N_0005 $N_0007      1
R_R8        0 $N_0007            1
V_V1        $N_0002              0  DC
+{Vt}
V_V2        $N_0004              0  DC
+{Vt}
V_V3        $N_0006              0  DC  5
```

Analysis

schematic as example3.net. (Note that you cannot run PSpice until the schematic has been saved.) We are now ready to perform the simulation.

Select "Simulate" from the "Analysis" drop-down menu to execute PSpice. When the analysis is complete, select "Examine Output" from the "File" drop-down menu in the PSpice window. The example3.out file will be loaded into the Notepad program for you to view. Selected lines from that file are listed in Table 4.6, and Table 4.7. The ".STEP" line in the Analysis setup section in Table 4.6 indicates the values of Vt at which the circuit variables will be computed. The lines for V_V1 and V_V2 in the Schematics Netlist section show the parameterized dc voltage.

Table 4.7 contains the results of the two DC bias point analyses for the different values of Vt. Note that the first case corresponds to the digital number 1 because V3 is the only non-zero voltage source. The output resistor is R1 and the expected output voltage for this case according to Example 4.14 is one-twelfth of the "on" voltage. We do in fact see $5/12 = 0.4167$ V for $N_0001 in the node voltage table. Note that the current flowing through V3 is given as -1.667 A. This source is supplying 8.33 W so clearly current is flowing out of the plus terminal of the source. The negative sign is a direct consequence of the passive sign convention. Current is flowing into the other two sources, but no power is consumed since their voltages are zero.

All the voltage sources are "on" in the second case. This corresponds to a digital 7 and the output voltage at node $N_0001 is seven times that of the first case. Note that all voltage source currents are now negative as they are all supplying power to the circuit.

ıble 4.7: The DC bias result sections of the output file for the PSpice Example 4.14.

```
**** 06/28/95 03:51:28 ******* Win32s Evaluation PSpice (April 1995) *********
* D:\CIRCUITS\EXAMPLE3.SCH
*** SMALL SIGNAL BIAS SOLUTION TEMPERATURE = 27.000 DEG C
*** CURRENT STEP PARAM VT = 0
****************************************************************************
NODE VOLTAGE NODE VOLTAGE
($N_0001) .4167 ($N_0002) 0.0000
($N_0003) .8333 ($N_0004) 0.0000
($N_0005) 1.6667 ($N_0006) 5.0000
($N_0007) .8333
VOLTAGE SOURCE CURRENTS
NAME CURRENT
V_V1 2.083E-01
V_V2 4.167E-01
V_V3 -1.667E+00
TOTAL POWER DISSIPATION 8.33E+00 WATTS
*** 06/28/95 03:51:28 ******* Win32s Evaluation PSpice (April 1995) *********
D:\CIRCUITS\EXAMPLE3.SCH
*** SMALL SIGNAL BIAS SOLUTION TEMPERATURE = 27.000 DEG C
*** CURRENT STEP PARAM VT = 5
****************************************************************************
NODE VOLTAGE NODE VOLTAGE
$N_0001) 2.9167 ($N_0002) 5.0000
$N_0003) 3.3333 ($N_0004) 5.0000
$N_0005) 2.9167 ($N_0006) 5.0000
$N_0007) 1.4583
VOLTAGE SOURCE CURRENTS
NAME CURRENT
V_V1 -1.042E+00
V_V2 -8.333E-01
V_V3 -1.042E+00
TOTAL POWER DISSIPATION 1.46E+01 WATTS
```

8 Summary

this chapter we introduced a number of important concepts and rules that can be d to simplify the analysis of electric circuits. Along with these concepts came many initions which we will continue to use throughout the text. The main concepts and initions are summarized below.

- Equivalent impedance is the impedance of a single element that can be used to replace a collection of elements at a pair of nodes without affecting any of the voltages and currents throughout the rest of the circuit.

- The principle of equivalent transformations is to preserve the terminal $\hat{V}$–$\hat{I}$ relationships under these transformations.
- Series connection—two (or more) elements are said to be connected in series if they always have identical currents flowing through them.
- The equivalent impedance of a series connection is equal to the sum of the individual impedances.
- Parallel connection—two (or more) elements are said to be connected in parallel if they always have the same voltage across them.
- The equivalent admittance of a parallel connection is equal to the sum of the individual admittances.
- The voltage divider formula gives the voltage across an individual element (say the jth one) which is in series with n elements in terms of the applied voltage and the impedances: $\hat{V}_j = \hat{V}_s Z_j / Z_{eq}$.
- The current divider formula gives the current through an individual element (say the jth one) which is in parallel with n elements in terms of the total current and the admittances: $\hat{I}_j = \hat{I}_s Y_j / Y_{eq}$.
- The input impedance of a circuit is the equivalent impedance with respect to the input terminals.
- The symmetry of some electric circuits can be exploited to simplify their analysis. The main idea in exploiting symmetry of electric circuits is to identify equipotential nodes in symmetric electric circuits and connect them together into one node.
- The superposition principle states that the voltages and currents in a complex circuit with multiple sources can be found by summing the voltages and currents found for several regimes of the original circuit when only some subsets of original sources are active while other sources are set to zero. These subsets of sources are formed as a result of subdivision of all original sources into several nonintersecting groups.
- When voltage sources are set to zero, they are replaced by short-circuit branches.
- When current sources are set to zero, they are replaced by open-circuit branches.

4.9 Problems

Problems 1–6. Find the equivalent impedances of the circuits shown in the figures indicated.

1. Figure P4-1.
2. Figure P4-2.
3. Figure P4-3. Assume $f = 2$ kHz.
4. Figure P4-4. Assume $f = 10$ kHz.
5. Figure P4-5.
6. Figure P4-6.