

# Using the PSPICE Circuit Simulation Software Package

By Matt Valerio

Junior EE

Ohio Northern University

Fall Quarter 2002

## Overview

PSPICE (PC Simulation Program with Integrated Circuit Emphasis) is an exceptional circuit simulation software package that can be used to efficiently solve complicated circuit designs. The software allows you to construct a virtual circuit using a schematic, simulate it using various solving techniques, and analyze the results. It will prove to be a valuable tool in double-checking homework and verifying solutions. This tutorial will step you through constructing a circuit, simulating a circuit, and verifying the answers.

## Starting PSPICE

To begin this tutorial, first log into the computer. Once you are logged in, find the desktop icon labeled **Schematics**. Double click on it to start the schematic editor.

You should notice that three programs have started. First off, the Microsim Design Manager (Fig. 1) has been launched. This utility provides easy access to starting the other software packages in the PSPICE software suite. For simple circuit analysis, we will only use the Schematics program. Take a look around at the other programs such as PSpice A/D, Optimizer, and PCBoards.

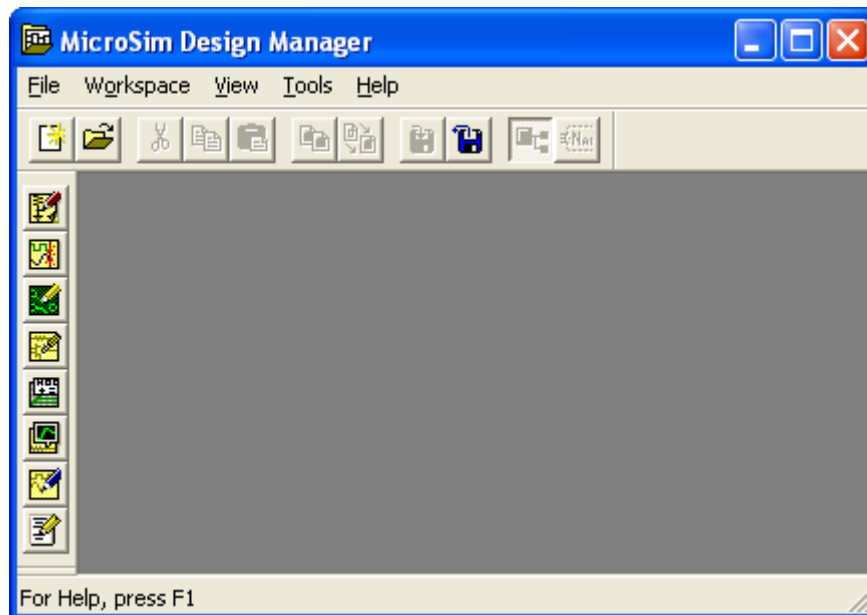


Fig. 1

You'll also see a program running called MicroSim Message Viewer (Fig. 2). When you simulate your circuit in Schematics, any errors will be posted in this window. Keep an eye on it if something isn't working correctly.

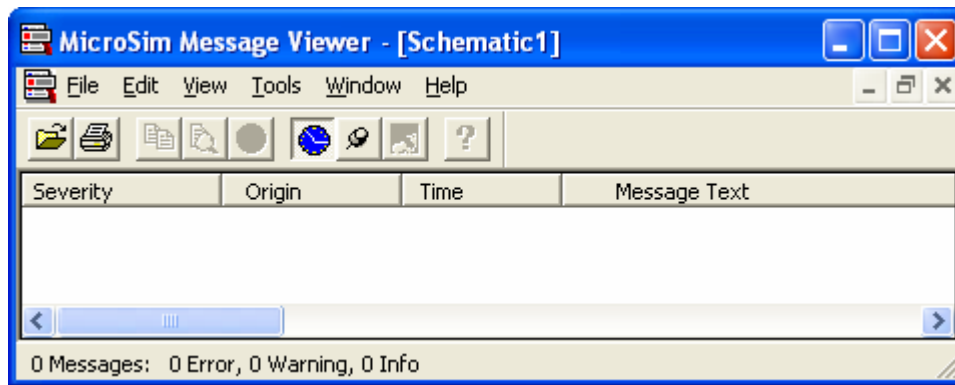


Fig. 2

Next, you'll see the MicroSim Schematics editor. This is the core of PSPICE, and you will spend the majority of your time using this program. Your screen should look something like Fig. 3.

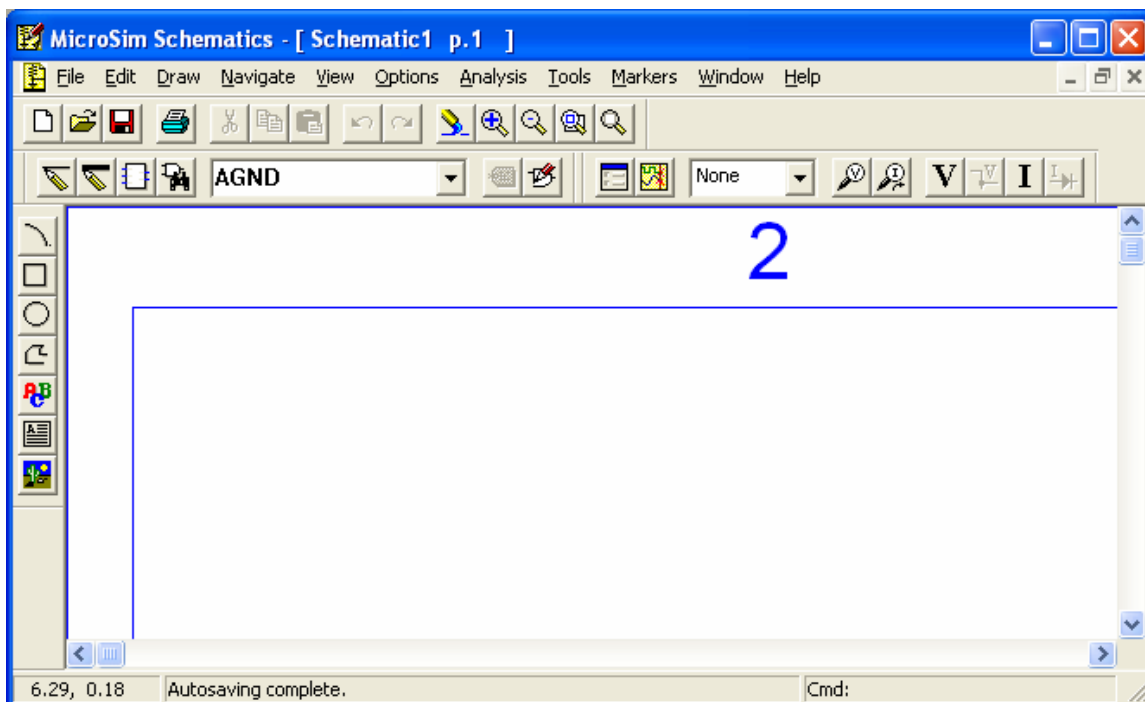


Fig. 3

The process of simulating a circuit has five major steps. First, a circuit is drawn by inserting parts, arranging the parts, and connecting them with wires. Next, the names, values, and other attributes of the parts are modified. The schematic file is then saved, and the circuit is simulated. The output results are then viewed and analyzed. The process can be summed up in Fig. 4.

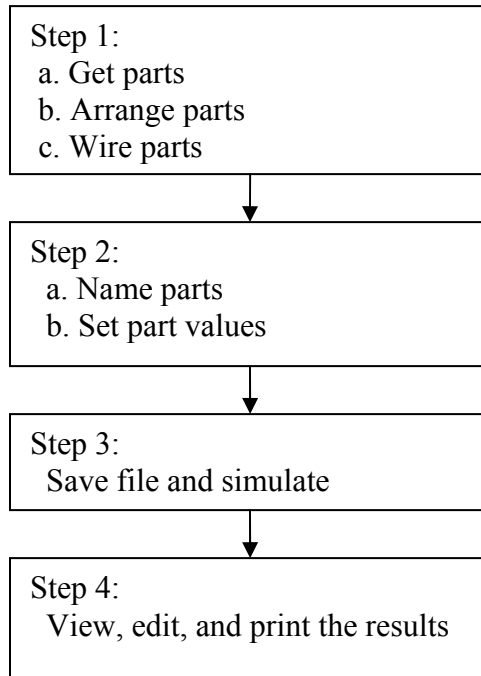


Fig. 4

As an example, we will solve the simple circuit shown in Fig. 5.

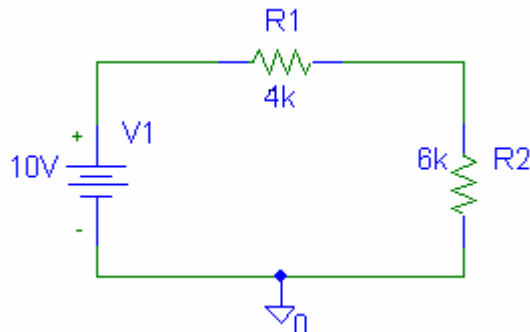


Fig. 5

### *Drawing the Circuit*

Now we will enter the circuit into the schematic editor.

On the menu bar, click on **Draw** and then **Get New Part**. This brings up the Parts Browser. You should see a dialog box like Fig. 6. We will insert the 10V DC source, 2 resistors, and a ground terminal.

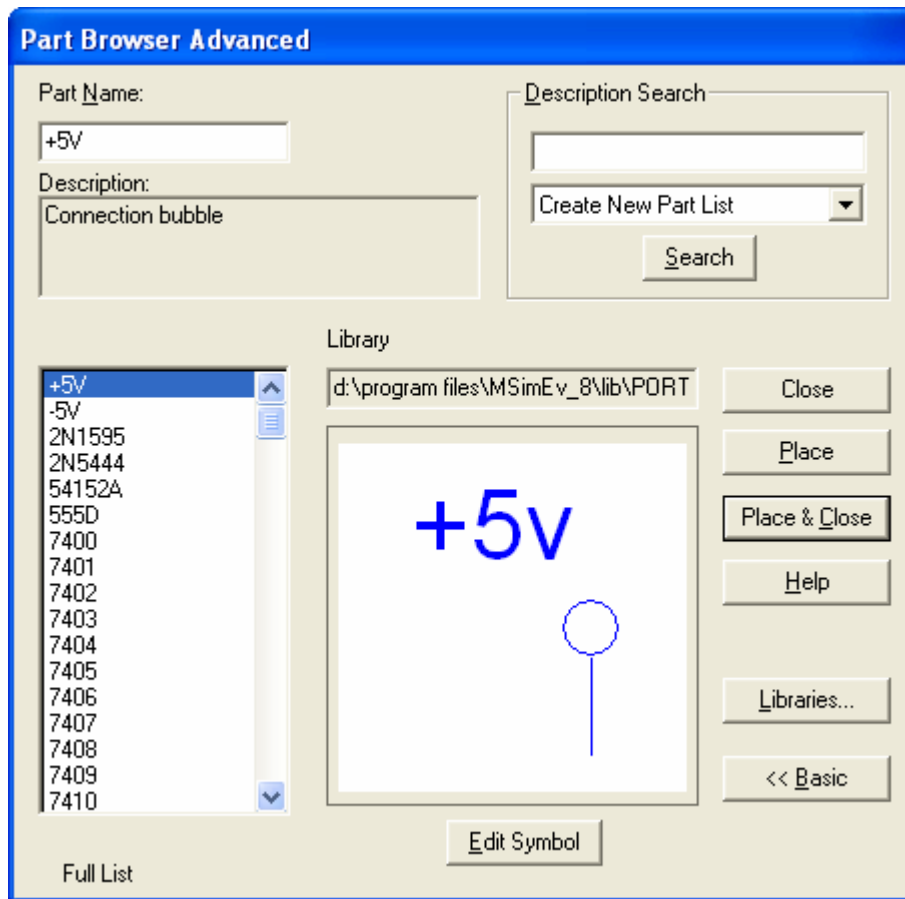


Fig. 6

To get the part for a DC voltage source, click in the Part Name field and type in “VDC”. You can also select this part by scrolling down to the part name and clicking on it. Click **Place & Close** to dismiss the Part Browser dialog box and place your part. Move your mouse to an area of the blank schematic and click once to drop the part.

PSPICE automatically assumes that you want to place multiple instances of the same part, and will let you place a part for every left click of the mouse. **Right click** the mouse or **hit Escape** to let PSPICE know that you are done placing VDCs.

To get the two resistors, repeat clicking **Draw/Get New Part**. This time, choose the part called “R”. Click **Place & Close** and left click twice in two different spots on the schematic to place two resistors, R1 and R2. Right click to end the command.

Finally, we must give PSPICE a ground reference for our circuit. We accomplish this by clicking **Draw/Get New Part** and choosing the part called “AGND”. This is an Analog Ground part, and is *absolutely necessary* for every circuit that you simulate. If you do not have one of these parts connected in your circuit properly, you will get errors when you attempt to run the simulation. Click **Place & Close** and place the ground part towards the bottom of the other components. Your screen should look something like Fig. 7.

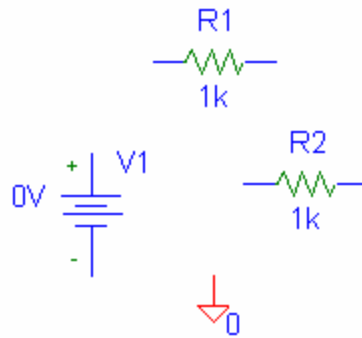


Fig. 7

By now you can see that R2 looks a little disoriented. We can fix it by selecting R2 and then rotating it. Click on the squiggly line section of R2 so that it turns red, indicating it's selected. Click **Edit/Rotate** (or type Ctrl-R) so that the resistor rotates 90 degrees. Then click it again to make sure it's selected, and drag it back into position.

Now we are ready to turn our pile of pieces into a functioning circuit. Click **Draw/Wire** (or click the toolbar button that looks like a pencil) to bring up the wire connection tool. Click the end of a component to specify the beginning of the wire, and click again on the end or another component to specify the ending of the wire. You should have 4 wires to connect:

- From V1 to R1
- From R1 to R2
- From R2 to ground
- From ground to V1

Right click or hit escape when you are done to turn off the wire connection tool.

After all of the components are successfully connected, your circuit should look like Fig. 8.

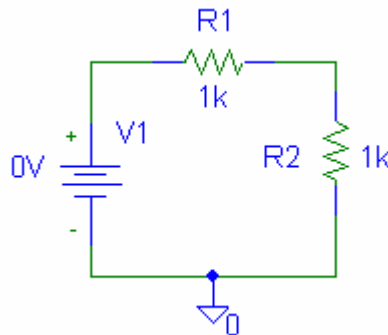


Fig. 8

Take a couple seconds here and try to get used to how PSPICE will let you manipulate your circuit. You can select a component (say, R2,) and drag it around. The wires stay connected! PSPICE will move them around so they are connected. If PSPICE draws a wire someplace where you don't want it, you can select a wire (it will turn red if it's highlighted) and then move it as you please. Also, the text around components may

be selected and moved around as you see fit. Knowing how to move items around can help your circuit look nice and neat.

Now the only thing that needs changed is the value of the voltage source and the values of the resistors. Simply double click on a value to bring up a dialog box where you can change it. Double clicking on the resistor value brings up a box that looks like Fig. 9.

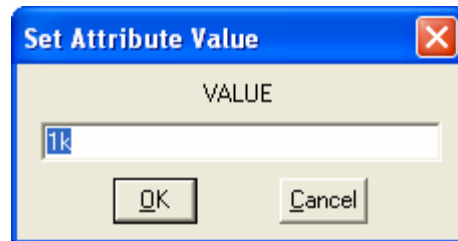


Fig. 9

Enter the values of “4k” and “6k” for the resistors, and “10V” for the voltage source. PSPICE will automatically know that if you enter “4k” for the resistance value that you mean 4,000Ω. Similarly, it understands that “10V” means 10 Volts. Some convenient scale factors that PSPICE recognizes are:

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

These letters must follow directly after the number you want to scale, and are not case sensitive. No spaces are allowed. For example, a 1,000,000Ω resistor should be denoted as “1meg”, not as “1 meg”.

By now your circuit should look like Fig. 5, and we are done with the schematic entry.

If we wanted to change the name of some of the components, we could double click on the part’s name and a dialog box like Fig. 10 would appear, allowing us to type in a new name for the part.

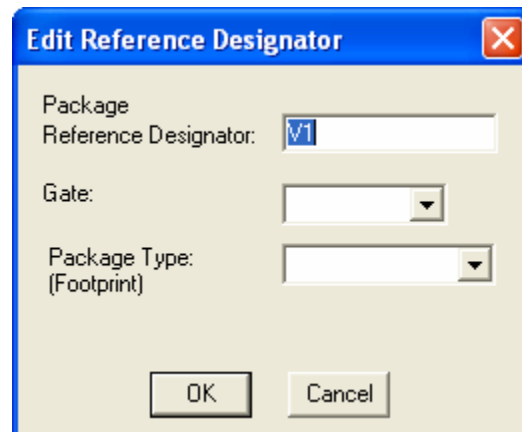


Fig. 10

Some of the most commonly used symbols are listed in Fig. 11 for your reference.

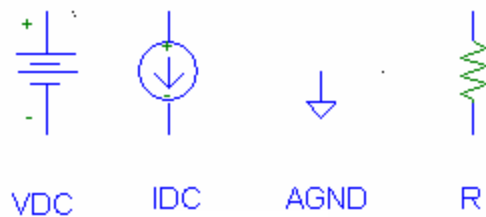


Fig. 11

### *Simulating the Circuit*

Since we have a correctly drawn circuit, we may now simulate it. Click **File/Save** and save the circuit somewhere. You must save the circuit before you will be allowed to simulate it.

Then click **Analysis/Setup** to bring up the Analysis Setup dialog box. The options in here control what type of simulation you want to run on your circuit. For finding the voltages at all the nodes and the current in all the loops, we need to use “Bias Point Detail”. Make sure the check box beside this is checked, and that no other simulations are checked, and hit **Close**.

It’s simulation time. Click **Analysis/Simulate** or hit F11. A window will pop up called “PSpiceAD” that runs your simulation. Your screen should look similar to Fig. 12.

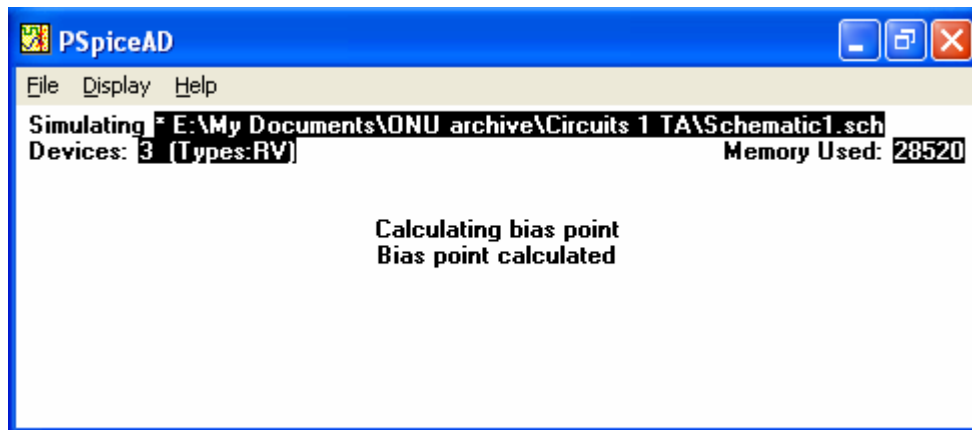


Fig. 12

This screen says that the bias points were successfully calculated. If your screen does not look like this, look at the MicroSim Message Viewer and determine the error that occurred.

### *Analyzing the Circuit*

PSpiceAD just simulated our circuit, and has the necessary data saved so that each part has voltages associated with the nodes on either side of it, and has an associated

current through it. You can look at the details in PSpiceAD by clicking **File/Examine Output**. This is not in the File menu of Schematics, but rather in the File menu of PSpice AD. Usually it's just a bunch of gibberish that can be deciphered if needed, but thankfully Schematics has a better way of extracting the information we seek.

Close the MicroSim Text Editor and go back to Schematics. Look around on the toolbar for some buttons that look like Fig. 13.



Fig. 13

Press the “V” button to display the voltages at all the nodes, and press the “I” button to see the current in the branches. Pretty nifty, huh? You should see something like Fig. 14.

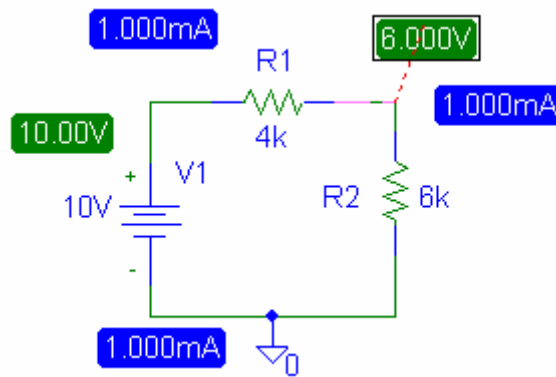


Fig. 14

Did PSPICE give us the right answer? We can do a quick voltage division calculation to see. The voltage across the 6kΩ resistor will be

$$V_R = \frac{6k}{6k + 4k} 10 = 6.0V$$

and this is precisely what PSPICE has shown us.

We can now print the schematic by selecting the entire circuit and clicking **File/Print**. Make sure that the “Only Print Selected Area” checkbox is checked, like in Fig. 15.

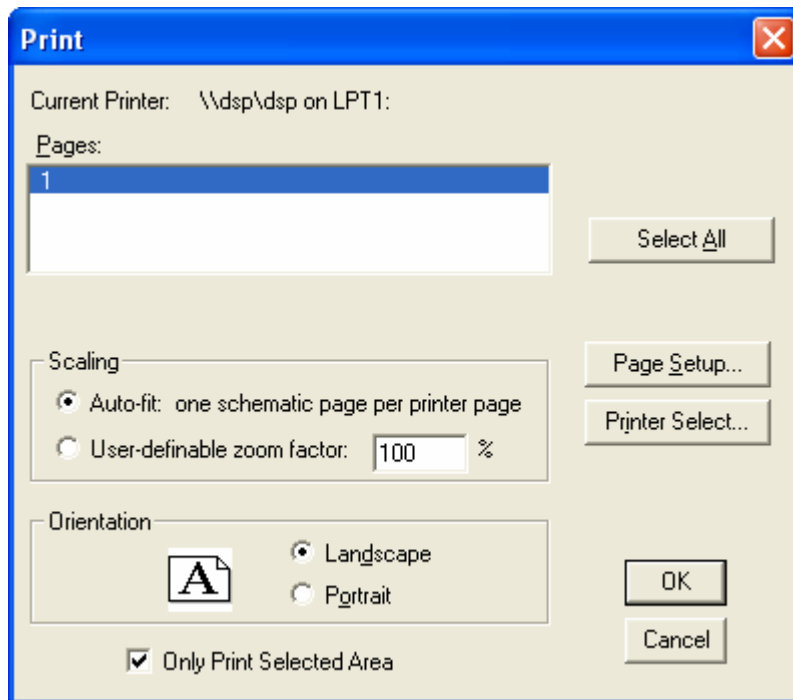


Fig. 15

And there you have it—a quick and dirty tour through PSPICE. With a little practice, PSICE will become an extremely useful tool in your experience with electronic circuits.

### *Practice Problem*

Draw the circuit shown below in Schematics and find the values of  $V_1$  and  $V_2$ . Check your results by solving the two node-voltage equations.

