CMPE 306

Lab II: Introduction to (LT) Spice:

Part 1: DC Applications

Created by: E.F.C. LaBerge based on presentation material from Y. Levin-Schwartz and T. Kuester

July 2013

Updated January 2017

E.F.C. LaBerge and Aksel Thomas

1. Purpose and Introduction

The purpose of this second lab exercise for CMPE306 is to introduce the student to the functionality of a particular version of the Simulation Program with Integrated Circuit Emphasis (SPICE). The particular version used in this lab will be the product LTSPICE, which is available for download from http://www.linear.com/designtools/software/. LTSPICE is available on the lab computers in ITE242. A version is available for Apple OS X 10.7+ There are other tools but this one is pretty easy to use and free, which a tough combination to beat!

LTSPICE is one of a multitude of circuit simulation programs that are based on the Simulation Program with Integrated Circuit Emphasis (SPICE) core originally developed by Laurence Nagel and Professor Donald Pedersen at University of California, Berkeley in the late 1960s and early 1970s (!).

LTSPICE is available on the lab computers in ITE242. You should save your work to a memory stick and *remember to take the stick with you!*

This week we will learn to use LTSPICE to simulate simple DC circuits. Students are encouraged to experiment with the program, as you can't really hurt anything, and to investigate its features and controls. The pre-lab exercises for future labs in this sequence will expect that all students have rudimentary skill in using LTSPICE (or some similar product) to simulate the circuits they will be building and testing during the lab sessions. This pre-lab activity serves to increase student awareness of the expected results so that they become more adept in identifying and troubleshooting errors in their circuit construction.

LTSPICE may be used to check or verify the solution to any homework problem assigned in CMPE306. LTSPICE may *not* be used to replace an analytical (i.e., by hand or MATLAB) solution.

By the end of this lab exercise, you should be able to do the following:

1. Start LTSPICE and create a new circuit.
2. Use standard models provided within the program to select appropriate circuit elements.
3. Wire the circuit elements together, creating a virtual breadboard of your circuit.
4. Label your circuit so that you can obtain unambiguous information from the simulation output.
5. Configure the simulation parameters to do a basic DC operating point and simple DC sweep.
6. Display sweep results on a plot.
7. Display simulation results on your basic circuit diagram.
8. Output and/or save the circuit diagram for future use and reference.

This is an attendance-based lab. No lab report is required.

2. Pre-lab assignment

The pre-lab assignment for this lab has two parts.

1. Students should consider downloading the free LTSPICE IV from http://www.linear.com/designtools/software/#LTspice. Obviously, you should download the correct version to correspond with your operating system!
2. Students should watch the LTSPICE tutorial available at http://www.youtube.com/watch?v=AsdwDpgpsj4. There are many LTSPICE tutorials, I chose this one because it does not introduce components or concepts that have not been covered (yet) in CMPE306 lectures. Feel free to explore, but recognize that the lectures have not yet covered op amps, capacitors, inductors or frequency response. Concentrate on what the videos are teaching about the tool, not the underlying math and physics. Lectures will get to all of that before the end of the semester.

   For Apple users, try https://www.youtube.com/watch?v=6AA4YBtqhwE . Same ideas, but a little different.

3. Equipment

The only equipment for this particular lab is a machine that can execute LTSPICE. This machine *may* be the lab computer in ITE242, or it may be your own laptop or iPad.

4. Procedure for today

We will develop and simulate six different circuits today:

1. A simple voltage source with resistor.
2. A simple two resistor voltage divider.
3. A simple current source with resistor.
4. A simple two resistor current divider.
5. A circuit with a voltage dependent current source.
6. A circuit with a current dependent voltage source.

Do your work carefully and save it on your laptop or on your memory stick. Some of the circuits will be very similar to pre-lab exercises for future lab sessions, so saving your work will give you a head start on those pre-lab assignments.

4.1. Simple voltage source and resistor

If you completed your pre-lab assignment, this procedure should be straightforward. I'll use LTSPICE as an example: any other tool should be similar. You should be able to figure it out.

1. Start LTSPICE by finding and clicking on the LTSPICE icon. If you can't find LTSPICE, go to Computer. Then click on Local Disk C: -> Program File (x86) -> LTC -> LTSpiceIV -> scad3.exe. When it opens, select "No" when asked to update.
2. From the menu bar at the top click on the New Circuit icon shown in Figure 1, or select File New Schematic. Your background should change to a light grey color. We're now ready to create a circuit.

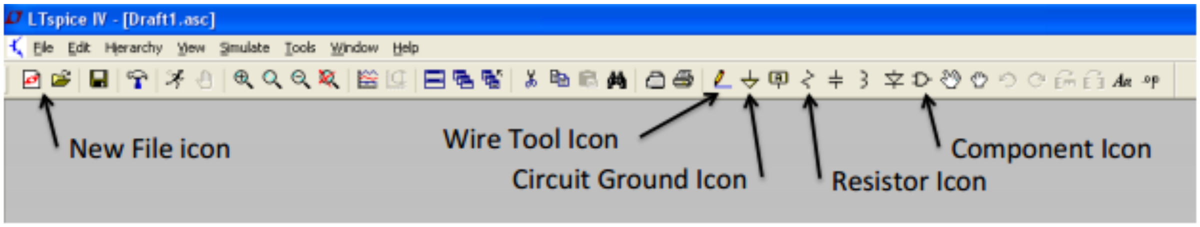

Figure 1 Icons needed for Simple Voltage Circuit

3. Click on the Component icon, as shown in Figure 1. This will open a component window with a wide range of common components and folders containing additional types of components. For now, we just want a voltage source. The components are arranged alphabetically, so scroll to the far right column and select Voltage, as shown in Figure 2.[1]

[1] For those of you using iCircuit, the symbol independent voltage source is the standard symbol for a battery, not the circle we will generally use. Sorry, that's just the way it is.

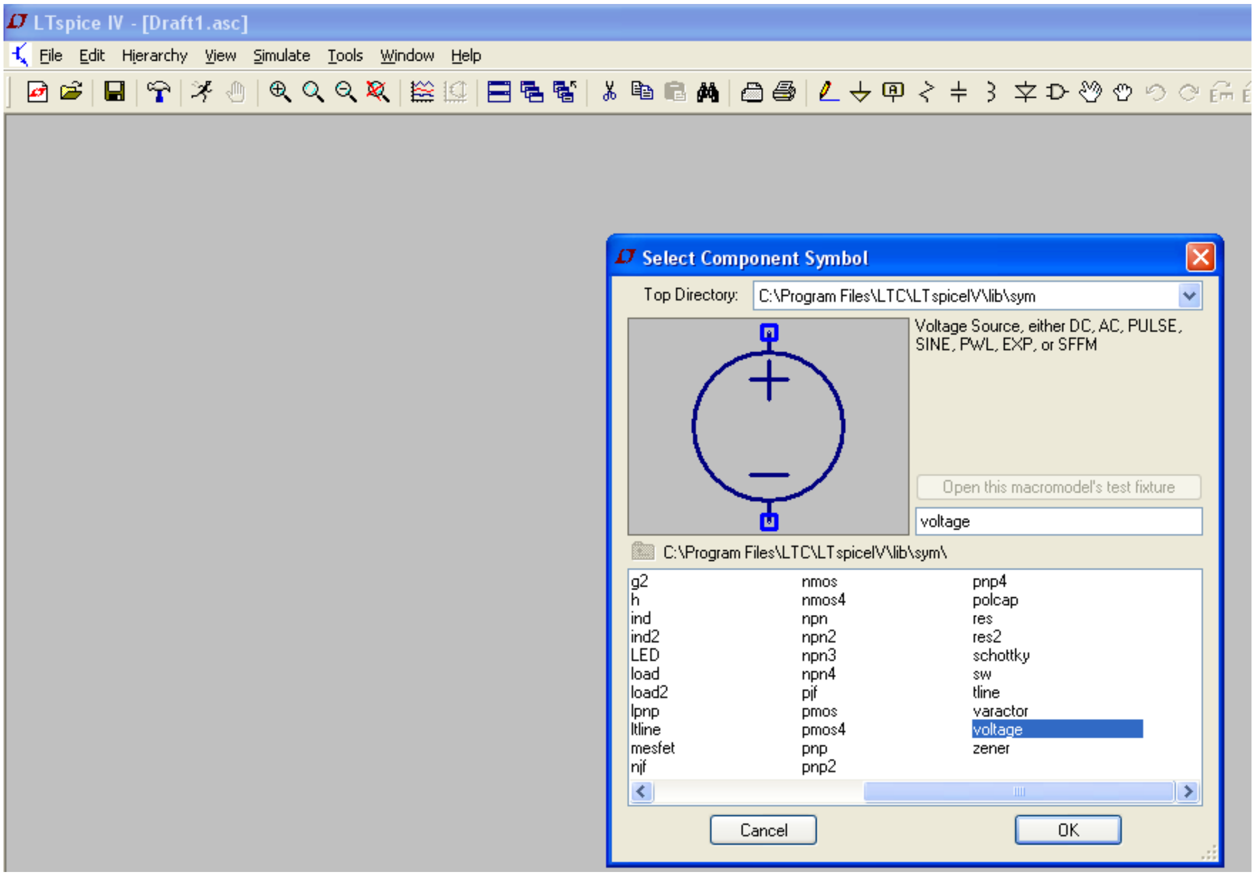

Figure 2 Selecting an independent voltage source

4. Drag the voltage source symbol to where you want it using your mouse. Click to drop it (you can move it later if you need to). Press <ESC> to exit the voltage source selection. You can, of course, drop more voltage sources *before* you escape, but this exercise only needs one.

5. From the icons at the top of the page, select a resistor and drag it onto your schematic. You can use <CTRL> R to rotate the resistor in $90°$ increments, if necessary. For this problem, it doesn't matter if you rotate it or not. Press <ESC> to exit the resistor selection.

6. From the icons at the top of the page, select the ground or reference node icon. Every LTSPICE circuit must have a ground node before it can be simulated. Place it in an appropriate spot (usually near the bottom) of your circuit schematic. Press <ESC> to exit the ground node selection.

7. Now select the Wire Tool icon (see Figure 1) and click on the upper node of the voltage source. Move the cursor vertically to a convenient spot and click. The wire tool draws a wire to that point. Then move horizontally and click. Your circuit should look like Figure 3.

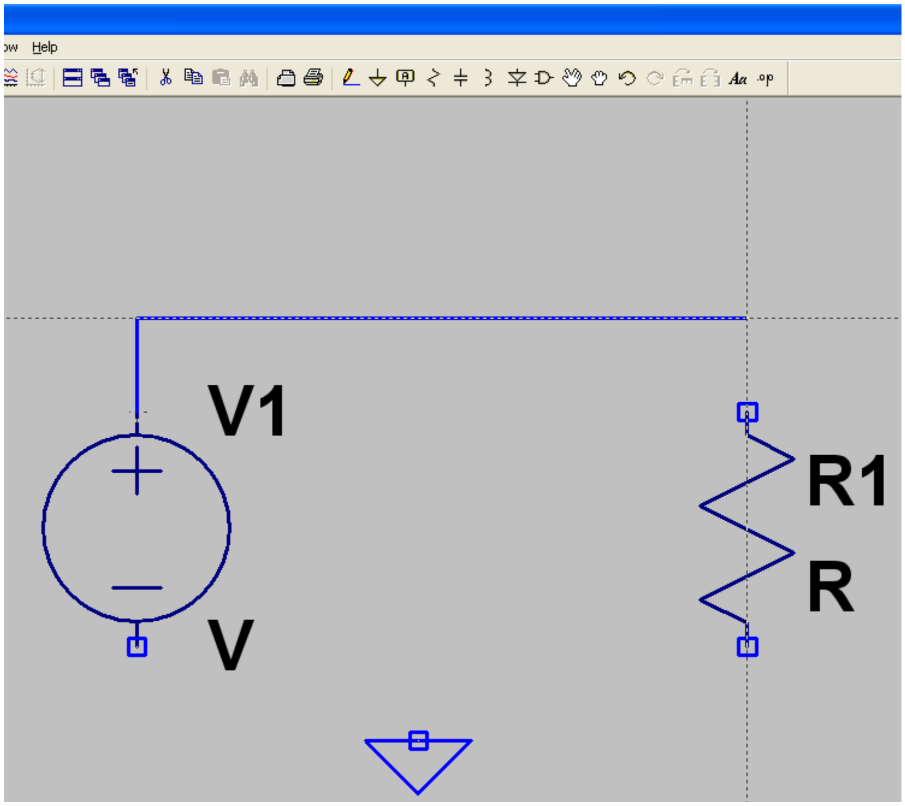

Figure 3 Partially wired simple circuit schematic

8. Use the Wire Tool to finish connecting the circuit. Don't forget that the ground node must be connected.
9. We've drawn the first circuit. Save your work with a representative name so you can find it again. I called mine SimpleCircuit1.
10. LTSPICE cannot simulate the circuit as we currently have it. We need to assign values to the components, which in this case, are just the voltage source and the resistor. The voltage source *label* is "V1" and its *value* is currently "V" volts. Place the cursor on the value and left-click the mouse to open the value window, shown in Figure 4. The default units for a voltage source are volts, so you can just enter the numeral 5 to set the value to 5 volts. LTSPICE also knows prefixes as shown in Table 1. In addition, you *may* choose to abbreviate the units. So to set the voltage source, you may use "5" or "5V". LTSPICE will ignore anything that is not a known prefix.

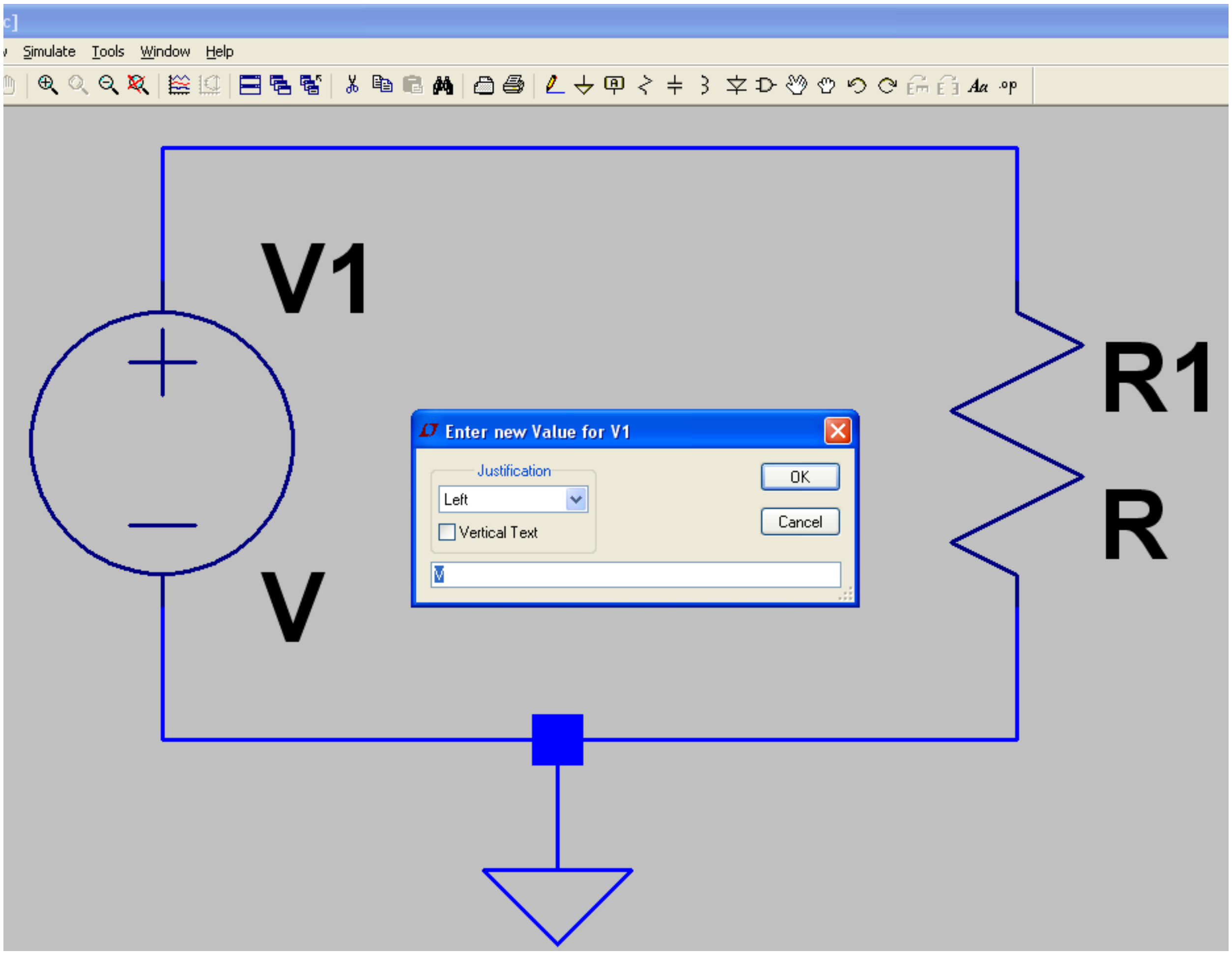

Figure 4 Setting the value of the independent voltage source.

Table 1 Commonly used LTSPICE Multiplier Prefixes.

LTSpice Prefix	Numerical Multiplier
F or f	10^{-15}
P or p	10^{-12}
N or n	10^{-9}
U or u	10^{-6}
m or M (!)	10^{-3}
K or k	10^{3}
mega (!)	10^{6}
G or g	10^{9}
T or t	10^{12}

11. Using the same technique, set the resistor value to 1000Ω. You can enter either "1000" or "1k"; both will work.
12. The circuit is now complete, and we need to set up the simulation parameters. LTSPICE has a rich array of simulations modes. For now, though, we only want the simplest mode, known as the "DC Operating point". From the LTSPICE menu select Simulate, and then the tab that says "DC op pnt". If you don't see the tabs after selecting Simulate, then click on Edit Simulation Cmd and the various simulation options will appear. When you click on "DC op pnt", the SPICE directive[2] .op will appear in a box on your schematic. Drag it around and drop it with a click at some place on your schematic. I generally put the directives at the bottom. Your circuit should look like Figure 5.

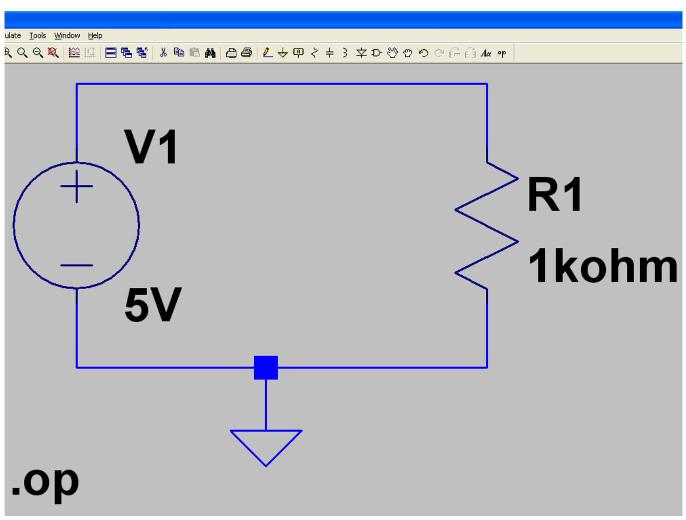

Figure 5 Completed simple circuit, ready for simulation

[2] A SPICE *directive* is an instruction to the simulation program itself. It's not really part of your circuit, but an instruction to the simulation on what to do with your circuit.

13. Now run the simulation. You can use the Simulation, Run menu pick, or just click the icon that looks like a runner. LTSPICE opens a new window showing the results of the simulation. For a simulation using the directive .op, the window looks like Figure 6.

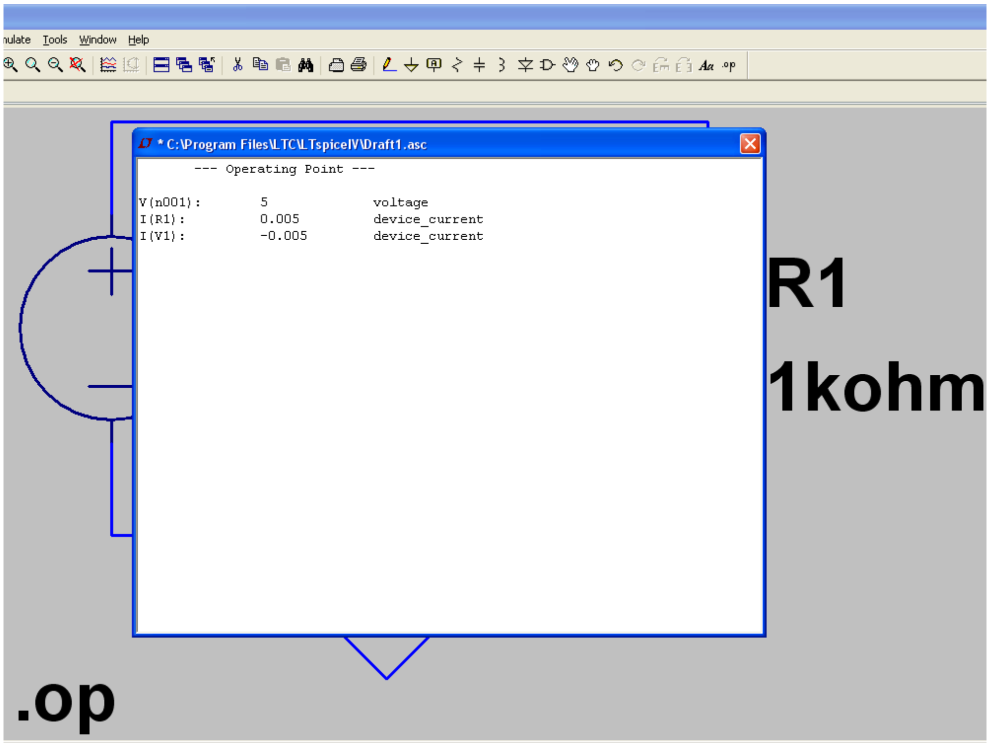

Figure 6 .op simulation of the simple circuit

Let's look at the output.

The first line says V(n001) 5 voltage. This tells us that the voltage at the generic node 1 (we did not specifically label any nodes, so LTSPICE does it for us) is 5 and the parameter is voltage. The current flowing through R1 is 0.005A or 5mA. The current flowing from the voltage supply is -0.005A or -5mA, where the minus sign tells us that the current is flowing *out* of the source.

A simple application of Ohm's Law shows that this is the correct answer, because $5\text{V} = \left(5\times10^{-3}\text{A}\right)\times1000\Omega$.

14. Save your circuit. We're done with the first element of this lab.

4.2. Simple two resistor voltage divider

We now move on to edit our simple circuit. LTSPICE is always in edit mode. To change a schematic, we just start adding new components and connect them appropriately.

1. Using your simple circuit schematic, select the drag icon (the closed hand, not the Vulcan death grip). Use it to grab the wire at the top of R1. Then press DELETE. A scissors icon will appear, indicating that the wire is about to be cut. Click to accept or ESC to exit. We want to accept.

The wire (or at least a portion of it) will disappear. Delete enough wire so that we have room to insert a second resistor.

2. Select the resistor tool, press <CTRL> R to rotate the resistor to a horizontal position, and place it appropriately.
3. Now wire V1 to the left end of new horizontal resistor, R2, and wire the right end of the new resistor to the top of R1.
4. Set the value of R2 to $1\text{k}\Omega$. Your circuit should look like Figure 7.

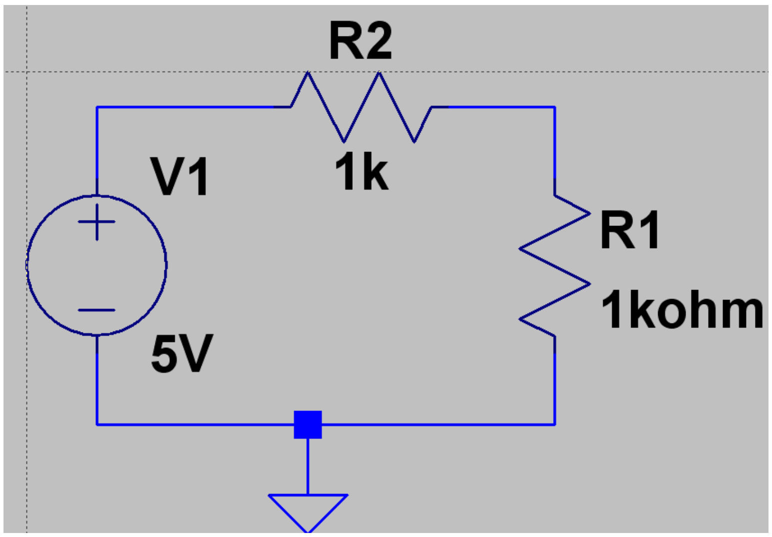

Figure 7 Simple 2 Resistor Circuit (Voltage Divider)

5. Save your circuit with a different name and simulate the circuit. The current is now 2.5mA. Why? The simulation now shows a voltage at n002 (generic node 2) as 2.5V. Why?
6. What happens if the value of either resistor changes? Change the value of R2 to $4\text{k}\Omega$. Predict the answer for the current through both series resistors and the voltage at n002. Simulate the circuit.
7. Now reverse the values of R1 and R2, restoring R2 to $1\text{k}\Omega$ and setting R1 to $4\text{k}\Omega$. What is your new prediction for the current through the resistors and the voltage at n002.
8. We will be doing things like this throughout the lab experience. Before we save the circuit as a graphic for insertion into a future lab report, let's label the nodes explicitly rather than relying on LTSPICE's effective, but not very elegant labeling. Select the Label Net tool, which has the icon of an A in a box. When you right click on the tool, a window opens and you can type in a label. *Spaces are not allowed in a label*. When you have the label, click OK and drag the label wherever you want it. You can drop it on a wire in the circuit (as in Node1 in Figure 8), or you can move the label *off* of the circuit and attach it with a wire (as in Node2 in Figure 8).

   When you have labeled your circuit with Node1 and Node2, redo the simulation. You will notice that the node voltages are now explicitly identified using your labels.

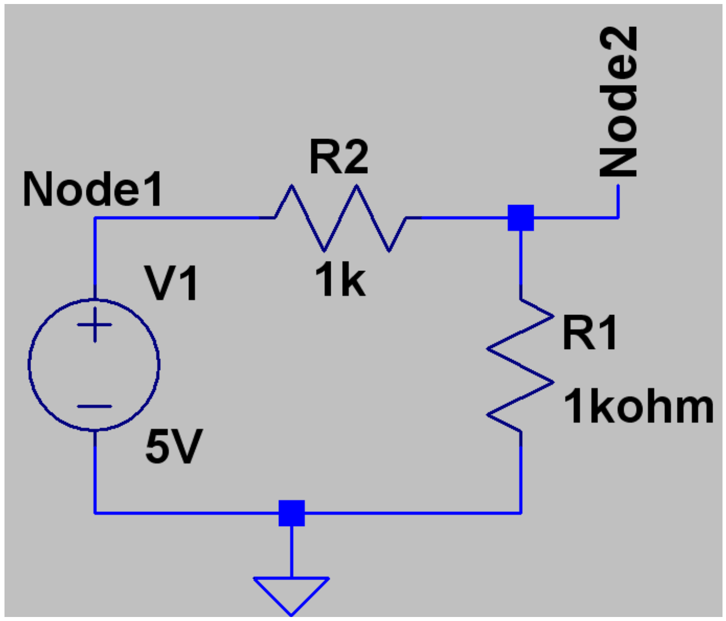

Figure 8 Labeling the circuit

9. With the labeling complete, you can now either copy the circuit directly to the Windows clipboard or save it to a .WMF file. Copying is fine if you're working on your own personal computer, but saving to a file is more useful if you need to share the information or if you're working on the lab computer. The options are available in the Tools menu.

10. Finally, we might want to have the results and the schematic all together in the same graphic. To do this, run the .op simulation. Select all of the text in the window using the usual Windows method of holding the right button and scrolling the mouse over the text to be highlighted. Copy this text to the Clipboard using <CNTRL> C. Close the results window with <ESC>, then select the Edit Text icon (*Aa*) and paste the Clipboard into the appropriate space. (Personally, I like to set the text size to 1.0, but that's up to you.) You can then drag and drop the text wherever you want, expanding or decreasing the display size to show what you're interested in. Figure 9 shows and example of the result of this process.

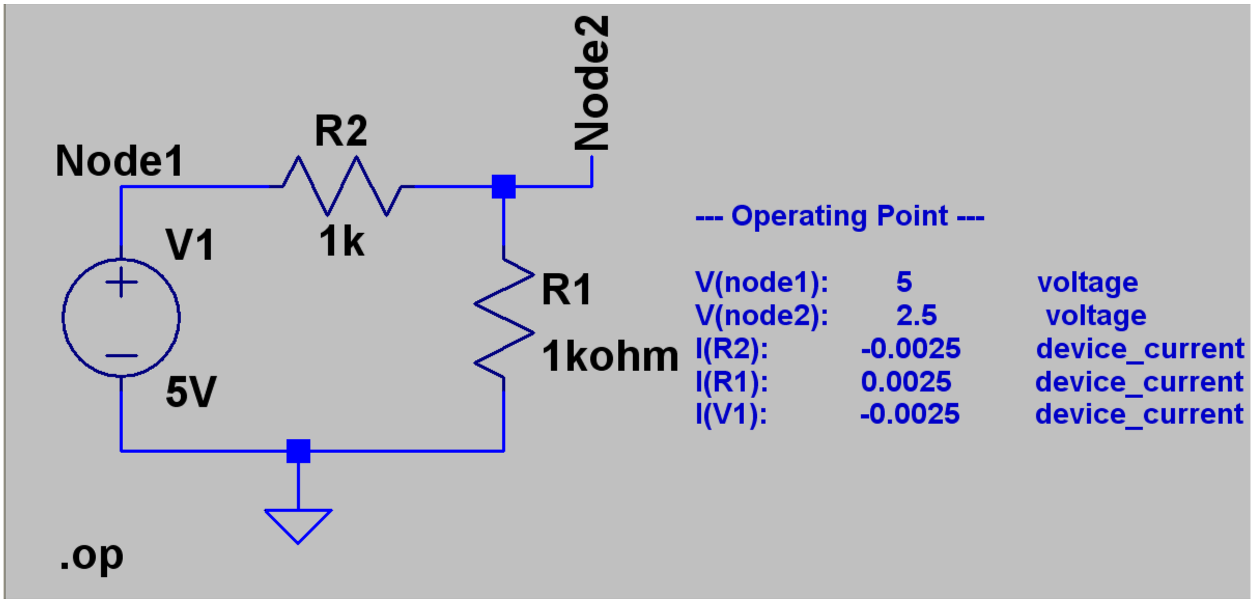

Figure 9 Annotated Circuit with Embedded .op Results

4.3. Simple current source and resistor

The same exact procedure is used to create a simple circuit with an independent current source and resistor. The circuit is shown in

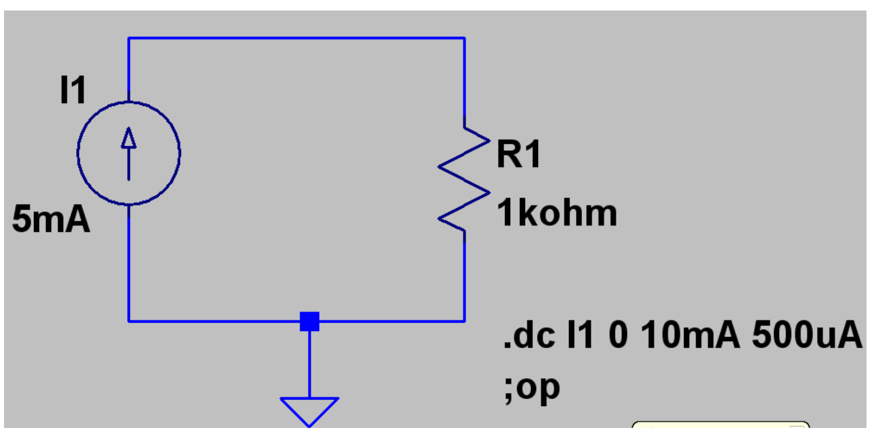

Figure 10 Simple Current Source with Resistor

1. Create a new schematic.
2. Use the components icon to enter the components menu and select a current device. Place the device on your circuit, then use <CTRL> R until the arrow points up.
3. Select a resistor and a ground.
4. Wire everything together.
5. Select the values of $1k\Omega$ for the resistor and $5mA$ for the current source.
6. Select a DC op pnt as the simulation mode.

7. Perform the simulation. What is voltage at the generic node n001? Does this make sense?
8. Now let's try a more advanced method of simulation. Say we want to look at the voltage at n001 as the current source is varied from 0-10mA. LTSPICE will do this for us. Select Simulate, and now Edit Simulation cmd. From the tabs, select DC Sweep, as we wish to sweep (i.e., vary) a DC parameter. We want to change the source I1, and we want the value to range from a minimum of 0 to a maximum of 10mA. LTSPICE will choose an appropriate step size for us, but let's set our step size to 500uA ($500\times10^{-6}\,\text{A} = 0.5\text{mA} = 500\mu\text{A}$). Notice that LTSPICE automatically creates the appropriate SPICE directive depending on your inputs. One reason to use LTSPICE is that you don't have to learn this "directive language" that underlies the simulation. Click OK, then position your new directive on your schematic. Click to release.

   Notice that when the directive is placed on the schematic, the old ".op" directive is changed to ";op". This essentially "comments out" the previous directive.
9. Now run the simulation. LTSPICE now opens a graph window above a copy of the schematic. But nothing is displayed! Go to the "Pick Visible Traces" icon – the one that looks like a series of Cartesian graphs – and click on it. You can now choose some circuit parameter to plot. Double click on V(n001). You see a linear plot from 0-10 mA on the x-axis, and 0-10V on the y-axis, as shown in Figure 11. If you take the slope of the plotted curve, you get $1\ \text{V/mA} = 1000\text{V/A} = 1000\Omega$ in agreement with Ohm's Law and the definition of resistance.
10. Save this as SimpleCircuit3, or something similar. You can save the graph file as before

Unless you like using one black ink cartridge per lab report, go into LT Spice's color preferences in the tools menu. This will let you get the nice black-and-white colors in a readable format. That method was used to create Figure 11

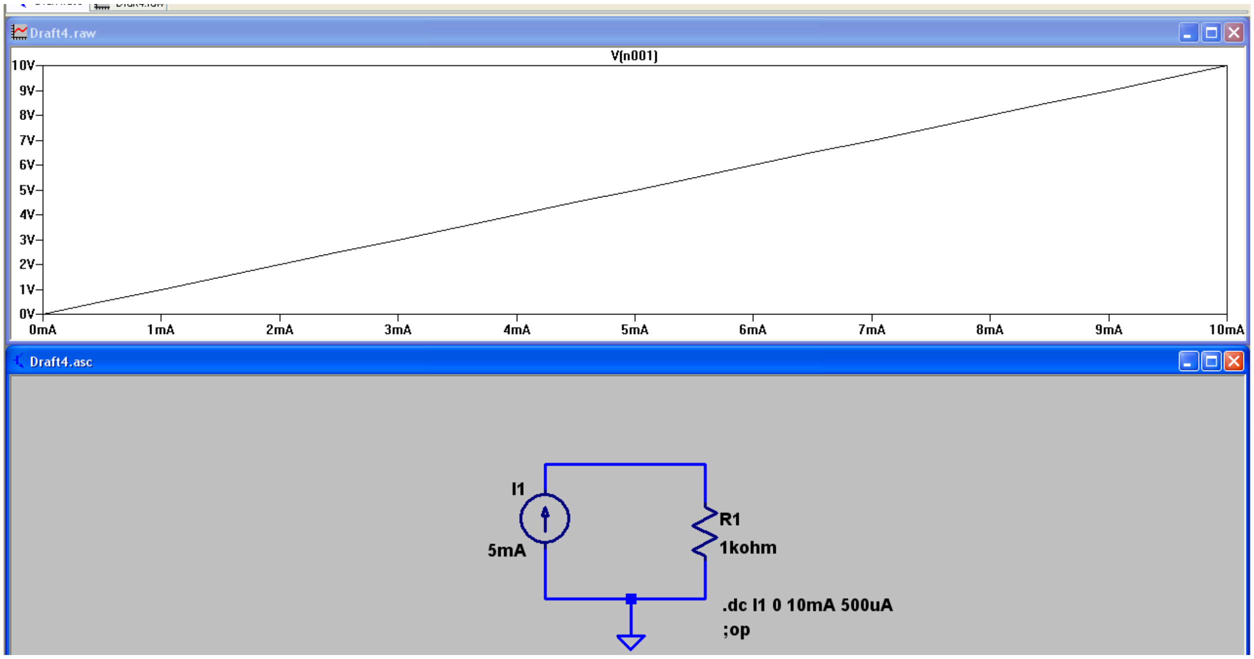

Figure 11 Simple Current Source with DC Sweep

4.4. Two resistor current divider

Just as resistors in *series* form a *voltage divider* (see 4.2, above), resistors in parallel form a *current divider*. Create and simulate a current divider in LTSPICE.

1. Using SimpleCircuit3, add a new resistor in parallel with R1. You don't need to erase any wires to do this (why not?). Set the value of the R2 to $1k\Omega$.
2. Delete the .dc directive, and set the directive back to .op.
3. Run the simulation. What is the ratio of the currents in R1 and R2? Does this make sense.
4. Change the value of R1 to $4k\Omega$ and re-run the simulation. What is the ratio of the currents in R1 and R2 to the total current provided by I1? Does this make sense? Which resistor has more current? Why?
5. Change the value of R2 to $6k\Omega$ and re-run the simulation. What is the ratio of the currents in R1 and R2 to the total current provided by I1. Does this make sense? Which resistor has more current? Why?

4.5. Circuit with a Voltage Controlled Voltage Source (VCVS)

Throughout this course, we consider four types of *dependent* source, without being specific about how such sources are implemented. The original versions of SPICE identified these as shown in Table 2. LTSPICE automates much of the complexity of using dependent sources.

Table 2 Dependent Sources in LTSPICE

SPICE Symbol	Source Type
e	Voltage Controlled Voltage Source (VCVS)
f	Current Controlled Current Source (CCCS)
g	Voltage Controlled Current Source (VCVS)
h	Current Controlled Voltage Source (CCVS)

To illustrate the process of installing a dependent source into your circuit, we will use a VCVS in this exercise and a CCCS in the following exercise. Implementation of a CCVS an VCCS is similar, except that the appropriate SPICE symbols are used. The important part of dependent sources is the *control parameter*.

For the VCVS, we'll simulate the circuit shown in Figure 12.

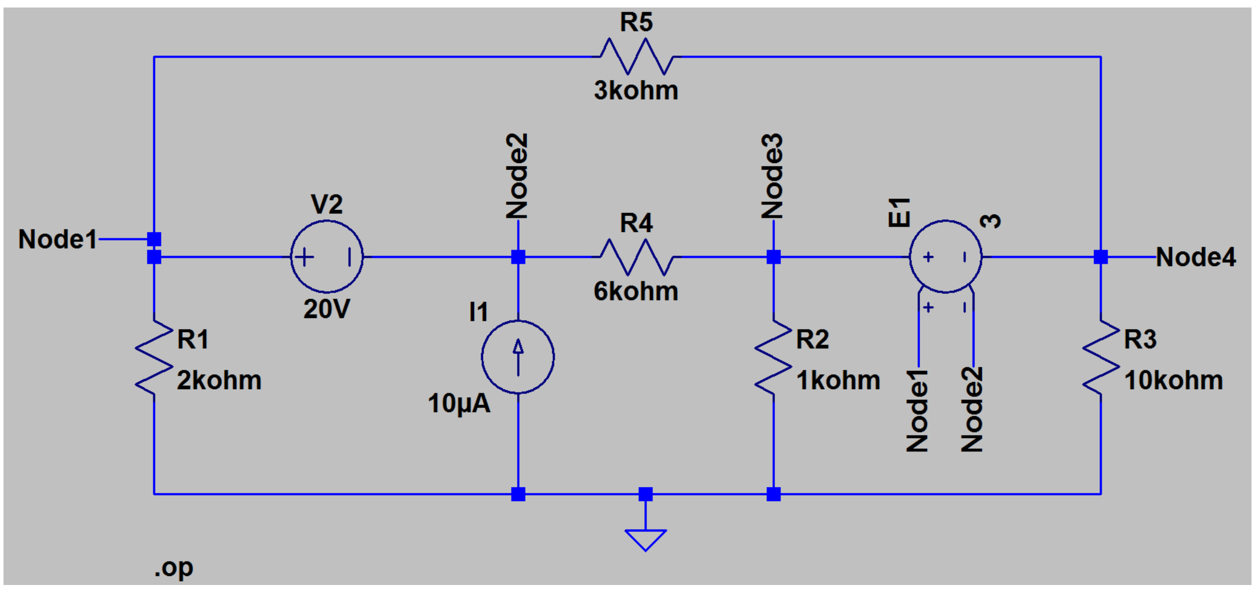

Figure 12 VCVS Example (modified from Alexander and Sadiku)

1. Open a new schematic and enter all of the elements *except* the VCVS. Connect the elements with wires. Label the nodes as shown in Figure 12.
2. Between Node3 and Node4, insert a VCVS. To do this, select the Component icon, and select the "e" (or "e2") component. Remember that (Table 2) "e" is the designation for VCVS. The only difference between "e" and "e2" is the orientation. Rotate the source to be horizontal with "+" on the left. Insert two new labels "Node1" and "Node2" and attach them to the control lines of the VCVS, as shown in Figure 12.
3. Now edit the "value" of Figure 12. For a dependent source, the "value" is the multiplier factor, so in this case, enter "3". The output voltage will be 3 times the voltage difference between Node1 and Node2.
4. Add the .op directive.

5. Run the simulation.
6. Look over the output. Which is higher, V(Node1) or V(Node4)? What is the difference, that is, what is the control voltage? What should the output voltage of the VCVS be? Is this, in fact, the voltage difference between V(Node3) and V(Node4)?

   The values I obtained are shown in Figure 13

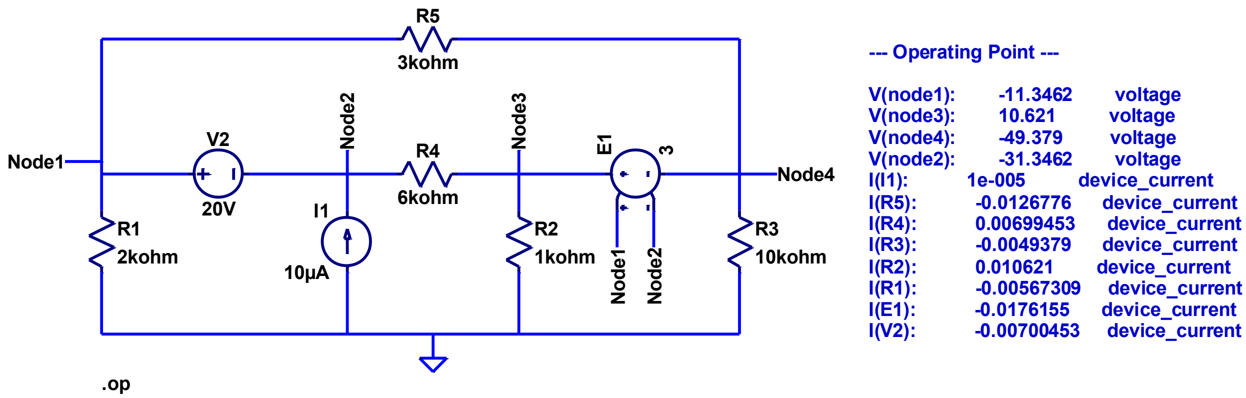

Figure 13 .op Values for VCVS Example[3]

4.6. Circuit with a Current Dependent Current Source (CCCS)

The last circuit for this lab uses a Current Dependent Current Source. Current controlled devices are the most cumbersome type of device to use in LTSPICE. But if you follow this procedure, things will work out fine. We're trying to simulate the circuit shown in Figure 14.

[3] While other figures in this writeup are screen shots taken from a Windows-based computer running LTSPICE, this figure is an inserted .wmf file, created using the process detailed in procedure 4.2, step 9.

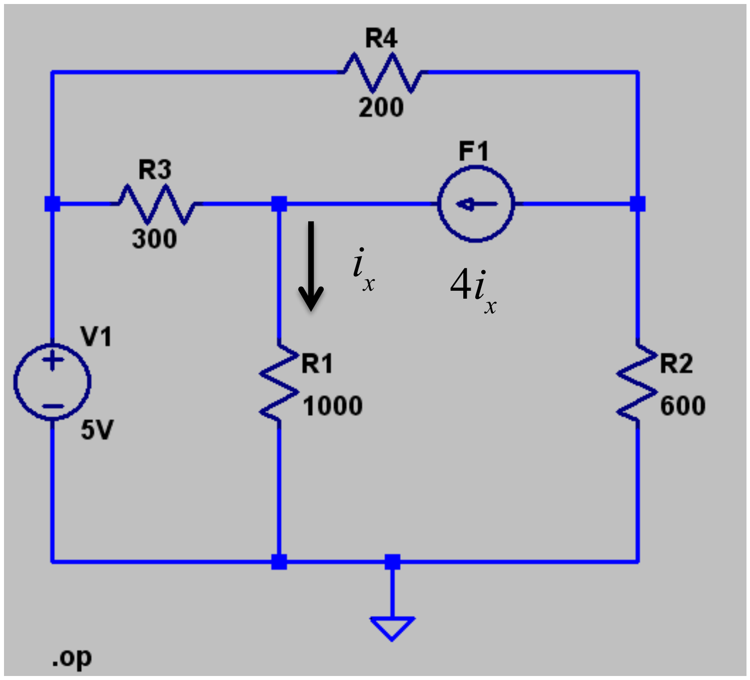

Figure 14 CCCS Example (modified from Alexander and Sadiku)

1. Create a new schematic, enter the circuit elements, and connect everything up. The CCCS should be a component **of type "f"**. Just wire it in as shown.
2. Add the .op.
3. Try and simulate. It won't work, because it doesn't have a gain or a reference current for element F1.
4. For legacy reasons reaching all the way back to the early implementation of SPICE, any current-controlled device (current or voltage outputs) needs to have the control current measured through an *independent voltage source*. If we want the control current to by i_x, the current flowing down across R1, we need to insert a *dummy voltage source* in series with R1. Using the hand tool (not the Vulcan mind grip) erase one of the wires in series with R1. The wire you choose can be above or below R1, it doesn't matter. The wire *must* be in series with R1.
5. In the gap you just created, place an independent voltage source. Label the source Vzero, and set the output voltage to 0V (yes, ZERO volts!). A voltage of 0V is equivalent to a wire! The current flowing through this voltage source will be our reference for the F1 device. Connect the wires, ensuring that this device is in *series* with R1.
6. Left click on device F1 to bring up the Component Attribute Editor as shown in Figure 15. This is the place where you can use older format SPICE directives to control the specific operation of the device. Double left-click on the Value column of the Value row (bad human interface, but it is what it is). A box will appear: enter "Vzero", a space, and "4". (No quotations!). This tells the model to use the current in the source Vzero and to multiply it by 4, which is what we want.

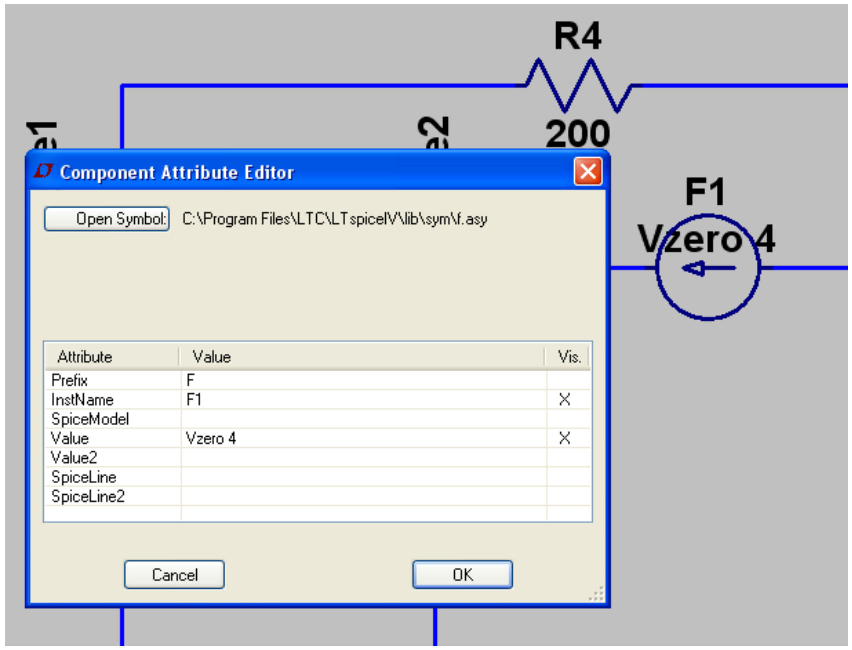

Figure 15 Component Attribute Menu for CCCS

7. Click OK.

8. Simulate the circuit. Does the answer make sense? What are nodes n001 through n004? It may not be clear.

9. Go back and label Node1, Node2, and Node3, as shown in Figure 16.

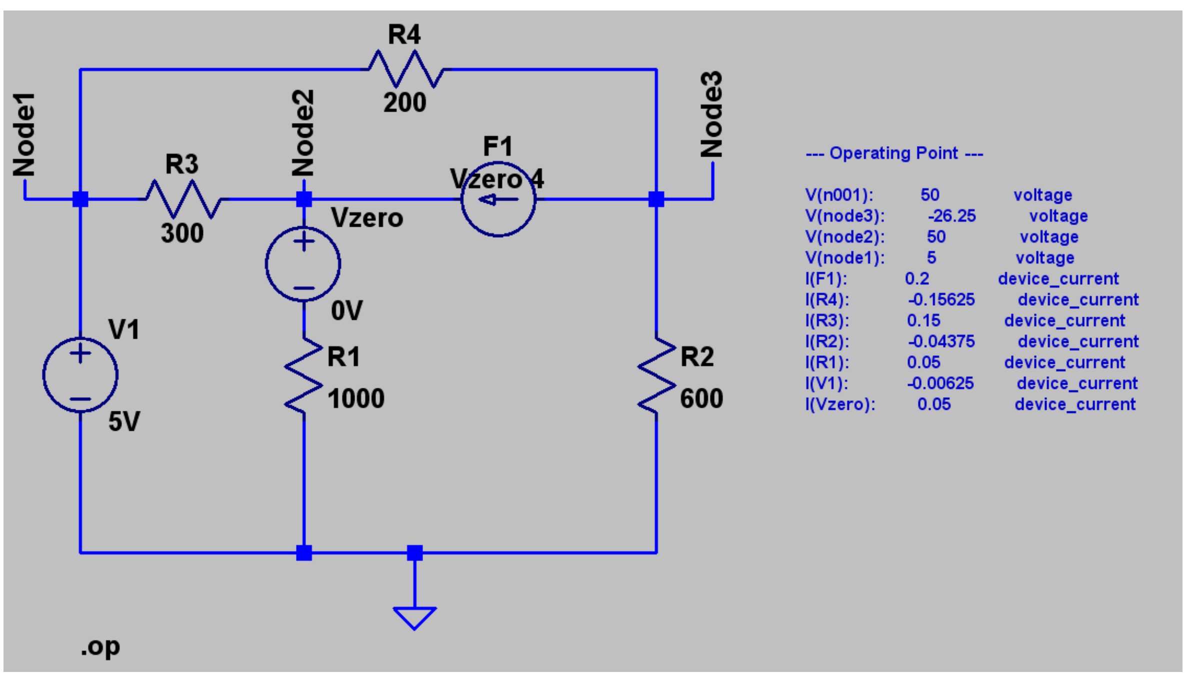

Figure 16 Labeled Circuit with .op Results for CCCS

10. Rerun the simulation. Now do the values make sense? Do they match Figure 16? Does the simulation satisfy KCL and KVL? Is the current through F1 four times the current through Vzero? What happens if you change the multiplier to 5?

4.7. Preparation for Next Lab

If you manage to complete everything with time to spare, you might take a look ahead at the next lab. Find the circuits that you will be building and simulate them in LTSPICE. It won't take long, as the circuits will be familiar to you already. In any case, you will have to complete this simulation activity as part of your pre-lab for next week.

This is an attendance based lab. No lab report is required. The next lab requires a pre-lab activity that is due at the start of lab period. A written lab report will be collected at the lab session *following* the actual lab. That is, you will turn in the lab report for Lab III when you are about to start Lab IV. For a schedule of lab activities, see the list in the Lab I writeup.

5. Tear Down and Clean Up

If you're using your own laptop, there's nothing to clean up.

If you're using the lab computer, save whatever work you want to your USB drive. Close LTSPICE. Eject your drive.

Be sure to clean up any paper or scraps. Always leave your lab area clean.