

Introduction to OrCAD Capture and PSpice

Notes for demonstrators

Professor John H. Davies

2010 April 06

Objectives

This handout explains how to get started with Cadence OrCAD version 16.3 to draw a circuit (schematic capture) and simulate it using PSpice. It includes examples of all four types of standard simulation and a selection of different plots.

1 Introduction

This document introduces you to a suite of computer programs that are used to design electronic circuits. Cadence OrCAD PCB Designer with PSpice comprises three main applications.

- **Capture** – used to draw a circuit on the screen, known formally as *schematic capture*. It offers great flexibility compared with a traditional pencil and paper drawing, as design changes can be incorporated and errors corrected quickly and easily. (On the other hand, it is much faster to develop the outline of a circuit using pencil and paper.)
- **PSpice** – simulates the captured circuit. You can analyse its behaviour in many ways and confirm that it performs as specified.
- **PCB Editor** – for the design of printed circuit boards. The output is a set of files that can be sent to a manufacturer or the electronics workshop in the Rankine Building. I do not cover PCBs in this handout.

The programs communicate using files called *netlists* but you should never need to look at them – everything happens transparently.

OrCAD is available on networked Windows PCs in the department, with up to 40 users at any one time. Unfortunately the licensing arrangements do not permit access from out-with the Rankine Building. A CD with Hambley's textbook contains a free version and we have a demo CD that may be copied. In principle you can download the demo version from www.cadence.com/orcad but the file is enormous and the download is unreliable in my experience. Little has changed in the basic functionality of Capture and PSpice since version 9.0 but the appearance was updated in version 16.2 to bring it into line with PCB Editor. The type of project can be changed in Version 16.3, which is a great help in the laboratory.

All these programs provide extensive online help. Please try this before asking for help. It is part of the learning process. Most 'professional' software is so complicated that even experts make regular use of the help system.

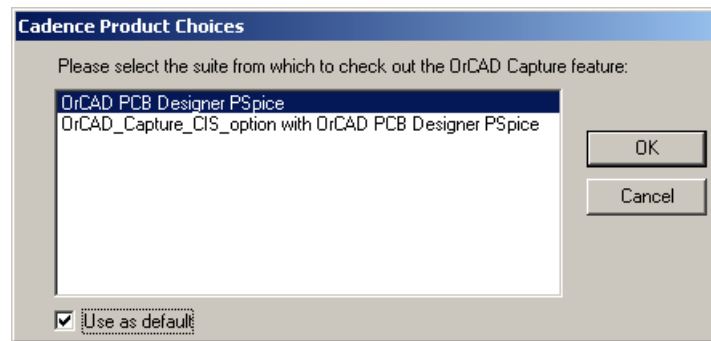


Figure 1. Dialogue box to choose product when Capture starts. Select OrCAD PCB Designer PSpice.

The aim of this laboratory is to simulate the behaviour of a simple electrical circuit. The program used is a version of SPICE (Simulation Program for Integrated Circuit Engineering), which you will use throughout your degree. SPICE was developed at the University of California at Berkeley in the 1970s and for many years has been the most widely used circuit simulator in the electronics industry. You will be using a version called PSpice A/D. There are three steps to using this software.

1. Draw an electronic circuit on the computer using **Capture**.
2. Simulate it with **PSpice** using specific models for your devices.
3. Analyse its behaviour with **Probe**, which can produce a range of plots. Historically this was a separate application but it is now integrated with PSpice.

People often refer to the whole suite as ‘Spice’. The material in this sheet is a basic start, but please feel free to experiment.

2 Starting OrCAD

Select Capture with Start > Programs > Cadence > Release 16.3 > OrCAD Capture (the version number may be different if the software has been updated). I use ‘>’ throughout this document to show the levels of a hierarchical menu. There will be a short delay while the software is loaded and the licence server is accessed. You may be offered a choice of product as in figure 1; select OrCAD PCB Designer PSpice. Wait until the red splash screen disappears. The screen will then show the OrCAD Capture main window with a menu bar and a dimmed tool bar. The untitled sub-window at the bottom is the Session Log.

OrCAD creates a large number of files as it runs, which are organised into a *project*. These files should be stored in your workspace on the University’s central system, accessed via the network as the H drive. Do not store files in any other place on the network or on a computer because they will be erased.

- **Always** create a new folder whenever you start work on a new project with OrCAD. Terrible things go wrong if you attempt to store more than one project in the same directory!

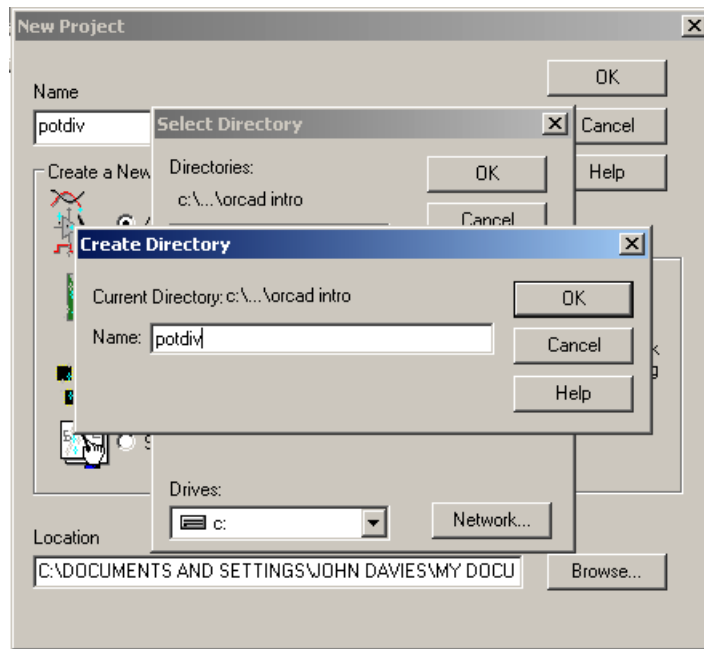


Figure 2. Dialogue boxes to create a directory for a new project.

- **Save** your work frequently and take regular backups of important circuits. It is a good idea to save these on a removable storage device.

Your first action must always be to create a new project when you are starting a design or to open an existing project if you are returning to a previous design. Follow these steps to create a new project.

1. Select **File > New > Project...** from the menu bar. This brings up the New Project dialogue box.
2. Select the Analog or Mixed A/D button. This is essential or you will not be able to use PSpice. (The type of project can be changed later with version 16.3; there is no easy remedy if you get this wrong in earlier versions, other than copying your circuit into a new project.)
3. Create a directory for the new project if you have not done this already. See figure 2.
 - Click on the **Browse...** key, which brings up the **Select Directory** box.
 - Select drive H: and navigate to a suitable location.
 - Click the **Create Dir...** button, which brings up the **Create Directory** box.
 - Enter a suitable name for a subdirectory, such as potdiv and click OK.
 - Double-click on this new directory to select it. In principle the path and new directory now show in the **Location** box but in practice the box is too short.
4. Enter a name for the project, such as potdiv (I used the same name as the directory but you don't have to do this).

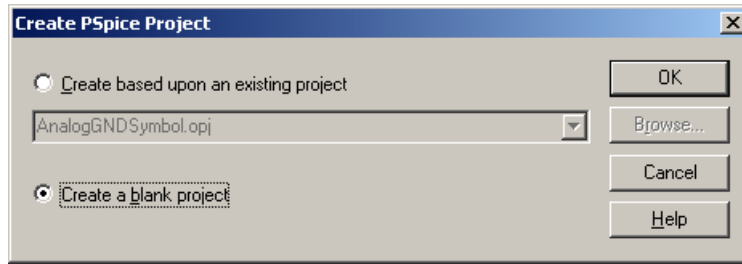


Figure 3. Final dialogue box to create a PSPice project. Select Create a blank project.

5. Click on OK in the New Project dialogue box to create the project.
6. Select the Create a blank project button in the small dialogue box that appears, shown in figure 3, and click OK.

This should open a Schematic window as shown in figure 4 on the next page but without the circuit, which we'll draw later. The layout may be slightly different if you have a larger monitor. If you don't see a schematic window, expand the tree in the Project window (top left) and double-click PAGE1.

2.1 Capture windows

The screenshot of capture shown in figure 4 on the following page shows three windows.

- **Project manager** – usually in the top left corner of the screen. Here is what it contains.
 - Its title is that of the project, except that it's far too long to display because it shows the whole path.
 - Under the title is the type of project, which should be Analog or Mixed A/D if you want to use PSpice.
 - The body of the window shows the various files that are used by your project, although you don't have to bother with most of them. I have expanded the tree for `./potdiv.dsn`, which is the current *design*, by clicking on the + signs. Each design can contain several *schematic* drawings, each of which can comprise several pages, but usually there is only one of each. If you close the schematic window by mistake, expand the tree and double-click on PAGE1.
- **Schematic Page** – where you draw your circuit. The Draw toolbar down the right-hand edge of the screen appears when this window is active.
- **Session Log** – along the bottom, title not shown. Most of the material written to this window can be ignored but you should study it carefully if an error occurs because this is where you find the explanation.

The row of tabs can be used to switch between the windows of your design.

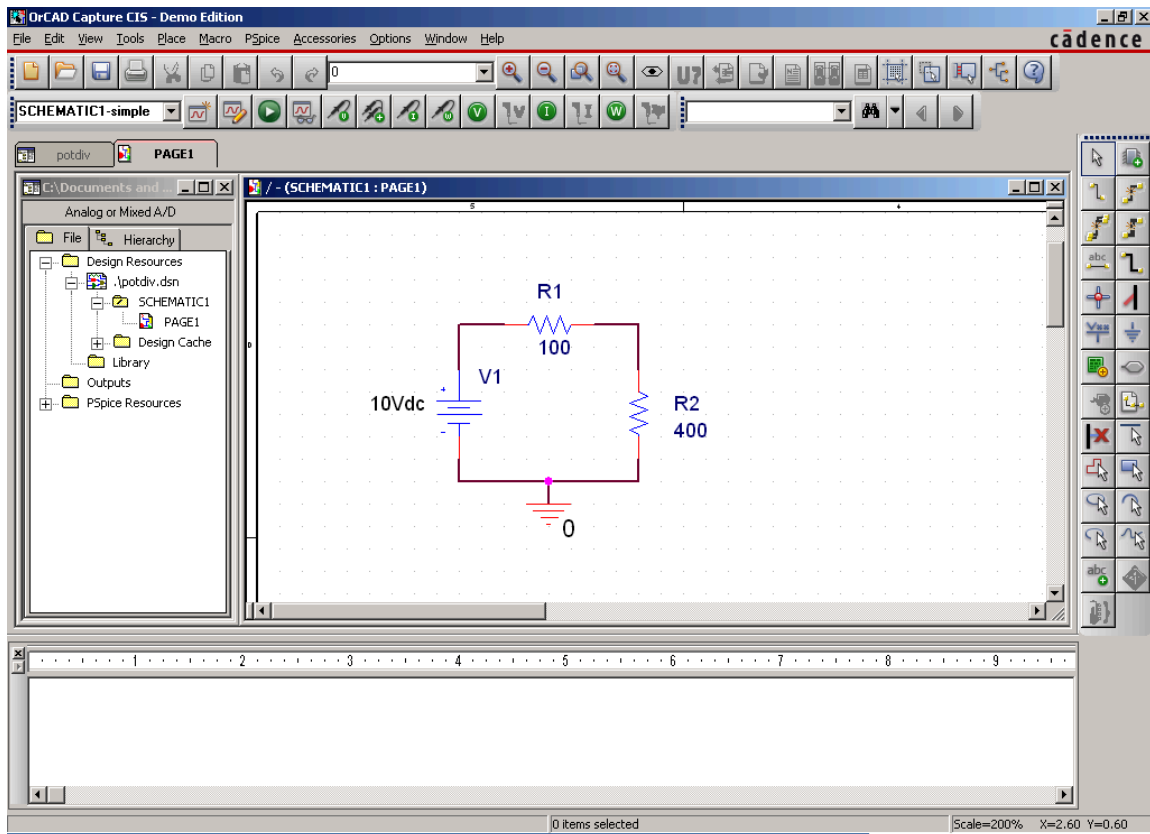


Figure 4. Screenshot of Capture, showing the simple potential divider used to illustrate the first tutorial. The Schematic window is active and the Draw toolbar on the right is therefore visible.

2.2 Capture menus and toolbars

The usual menu bar runs along the top and provides all the commands. Please note the Help menu on the right. Go here first if you need assistance. It provides the usual index and search. The manuals are also installed as pdf files.

Some commands are used so often that they have buttons and keyboard shortcuts as well. Many of the shortcuts use unmodified letters, such as P for Place Part, and do not need the Ctrl or Alt key. Here is the function of the button bars along the top, shown in figure 5.

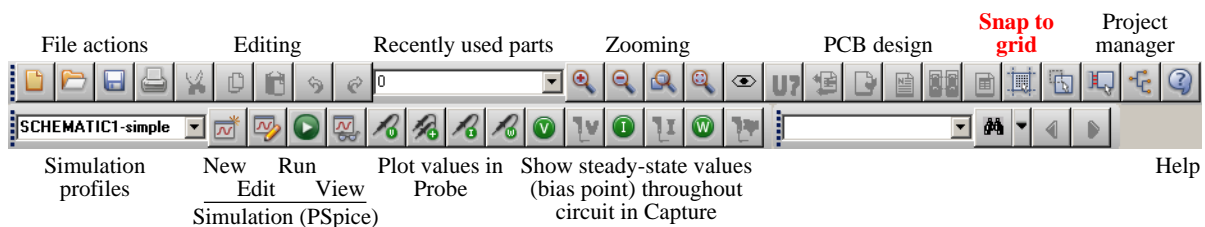


Figure 5. Toolbars along the top of the screen in Capture. Not all buttons are active at the same time.

1. The top bar is for Capture commands and falls into several groups.
 - First are the usual buttons for dealing with files and editing, as in many Windows applications.
 - The dropdown list in the middle contains recently used components so that you can select them quickly when drawing your circuit. The list is initially empty.
 - The next set of four buttons are for zooming in and out so that the circuit is drawn on the screen at a convenient scale.
 - We won't use the next set in this tutorial; they are more useful for larger circuits and laying out PCBs.
 - The next button is important because it controls whether components are placed on a grid or can be moved freely. *Always use the grid* (which is the default) or you will have terrible trouble joining the wires to them.
 - A button selects the Project Manager, which can also be activated from the Window menu.
 - The final button is for Help.
2. The second toolbar along the top is for PSpice commands. You are in trouble if this is missing! Check the type of project if this happens; it must be Analog or Mixed A/D. I'll explain the functions of the buttons later.
3. The Draw toolbar, down the right-hand side of the screen, is for drawing circuits and is shown only when a Schematic window is active. The most important buttons are labelled in figure 6 on the following page.

Let's put this into action and simulate a simple circuit: a potential divider.

3 Capture and simulation of a potential divider

We'll first draw and simulate a very simple circuit: a potential divider supplied by a battery. Of course you can solve this circuit mentally in a second or two but the aim is to gain familiarity with the software. In fact you should solve it yourself first as a check. To start the simulation in OrCAD, place the components.

3.1 Place the components

Make the Schematic window active. We next need to find the components: two resistors to form the potential divider, a DC voltage source (battery) and a ground. Their symbols are shown in figure 7 on the next page.

Choose Place > Part... from the menu bar, click the Place Part button on the right-hand toolbar or type P. This brings up the dialogue box with a list of parts shown in figure 8 on page 8, from which you choose the desired component.

Note: the Place Part dialogue box comes in two varieties, either the 'traditional' or (since version 16.2) the 'docking' form. Both are shown in figure 8. The docking form can slide in and out or it can be pinned open. Change between the two forms by choosing Options >

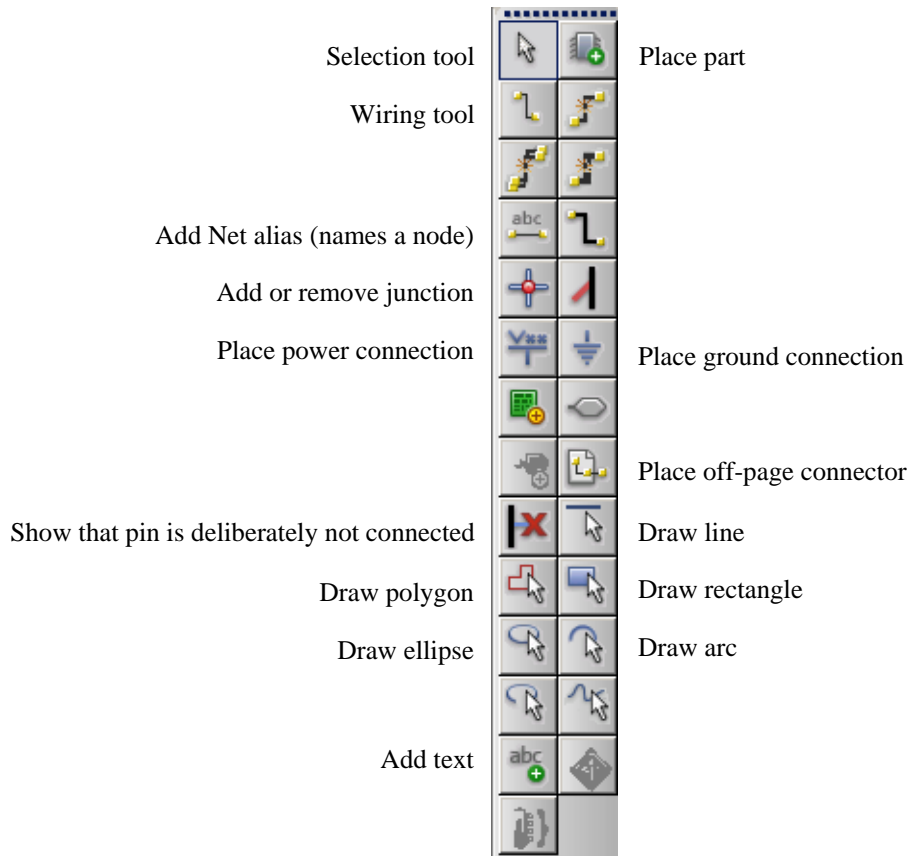


Figure 6. Draw toolbar for Schematic window in Capture with labels for the most important buttons.

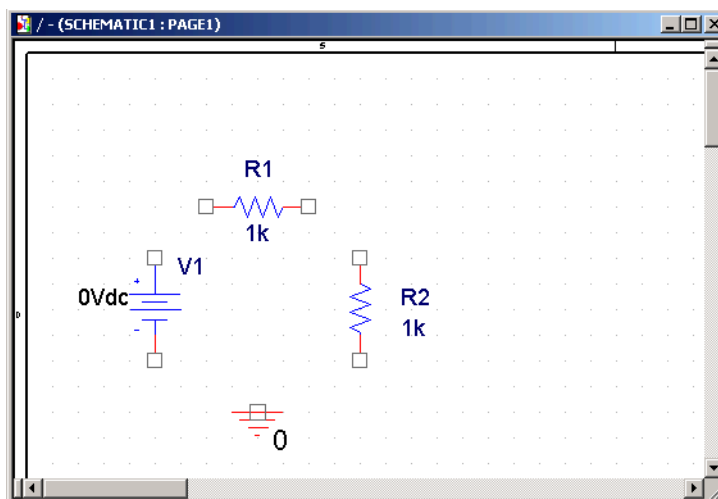


Figure 7. Components for potential divider before connecting them.

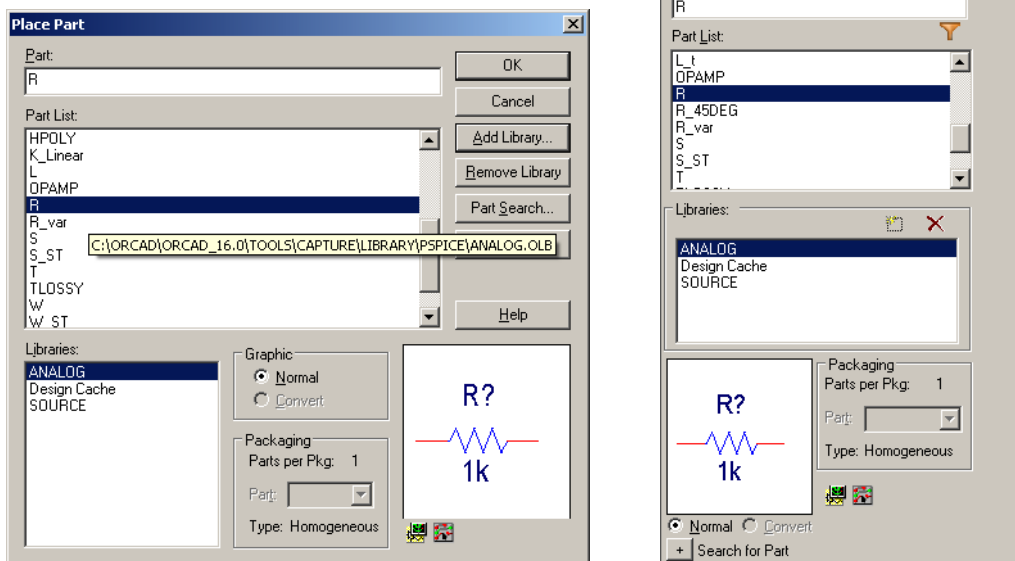


Figure 8. Traditional and docking forms of the Place Part dialog box with a resistor selected.

Preferences from the menu bar, select the Miscellaneous pane and use the Docking Place Part option.

Capture includes a database with a vast number of components, which are organised into *libraries* for convenience. We must first select the desired libraries, shown in the bottom left list.

- If there are libraries included already, select them all and click the Remove Library button. Capture will warn you that it cannot remove the Design Cache – dismiss the box.
- Click Add Library... (the dotted box with a star in the docking form of the dialogue), which brings up the dialogue box shown in figure 9 on the next page. Check that the libraries are in the pspice folder and navigate to it if necessary. Select analog.olb and click OK.

This library contains basic analogue components such as resistors and capacitors.

- Repeat this for the source.olb library. This contains ‘sources’ such as batteries. Almost every analogue circuit needs these two libraries.

After adding the libraries, click on the ANALOG library in the Place Part dialog box, scroll down the Part List and choose R as in figure 8. This is a resistor of course. The corresponding symbol is shown at the bottom right; it is a zigzag rather than the blank rectangle favoured by the IET. Click OK, which takes you back to the Schematic window.

The resistor moves around the workarea with the mouse until you left-click on the mouse to secure it in place. Once positioned in the work area, the first resistor assumes the name R1 and snaps to the grid, which is just visible as a faint array of dots. You can place successive resistors by repeatedly left-clicking. Here it would be useful to have the second resistor vertical rather than horizontal so right-click and choose Rotate before left-clicking to place the second

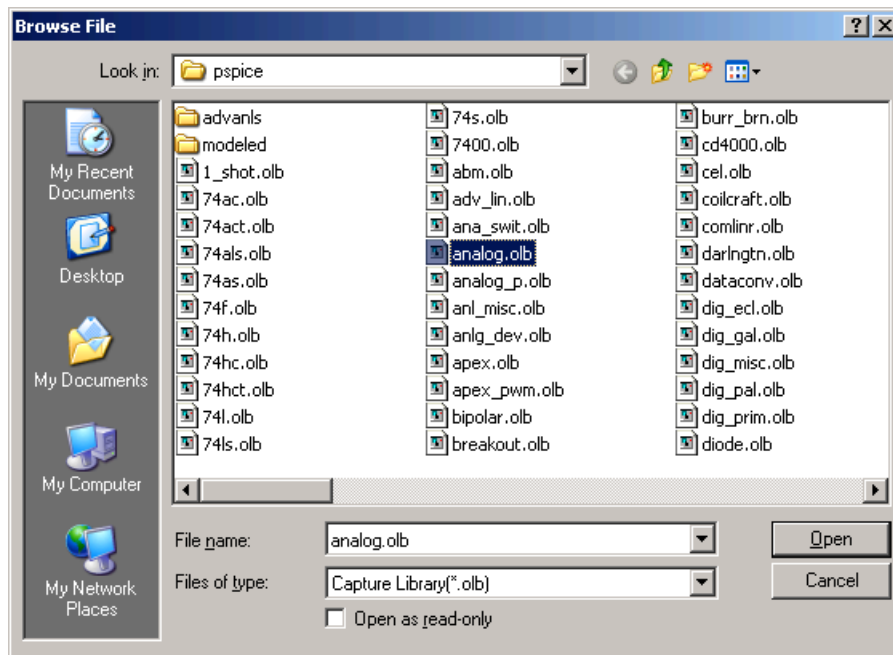


Figure 9. Dialogue box for adding a library to Capture.

resistor. When you have placed both resistors, right-click and choose End Mode. Alternatively, hit the escape (Esc) key.

Each part must have a unique *reference designator* (*reference* or *refdes* for short) to identify it, such as R1, R2 and so on for resistors. Capture creates this automatically. It is a fatal error to have two components with the same refdes so leave them alone. You can move and rotate components after they have been placed. Select a component by left-clicking near its centre, which turns the component and labels to magenta. You can then move, mirror or rotate it with the contextual menu.

Now add the battery. This is called VDC and is in the SOURCE library. Finally, add a ground connection. This is a bit different.

- Choose Place > Ground... from the menu bar or click the Place ground connection button on the toolbar, which brings up the dialogue box shown in figure 10 on the next page.
- Choose 0/CAPSYM and place it in the usual way.

A circuit must *always* have a ground (earth) connection called 0 (zero) if you wish to simulate it in PSpice. You will get error messages about floating nodes if you forget this, which is very easy! The reason is that all voltages in PSpice are measured from a particular node, numbered zero, and this must be defined as ground.

Your circuit should now look something like figure 7 on page 7 although the components need not be in exactly the same positions. Save your work!

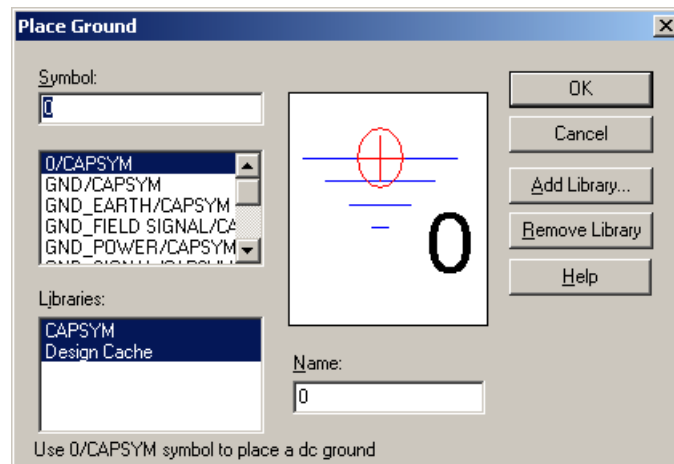


Figure 10. Dialogue box for adding a ground connection.

3.2 Set the values of the components

Capture gives each component a default value when it is placed but this must usually be changed. All resistors are initially 1 k Ω , for instance. The unit is not shown – ohms are assumed by default so the display is just 1k. Double-click on a value to change it. This brings up the dialogue box shown in figure 11 on the next page for the battery (VDC). If you see something different, you have probably double-clicked in the wrong place. Close the box and try again.

Change the values of the resistances and voltage to those shown in figure 12 on the following page. I'll explain how to wire the components next.

3.3 Wire the components

The final step is to connect the components using the wiring tool. Choose Place > Wire from the menu bar, the button on the Draw toolbar or type W.

- Place the cursor where you wish to start connecting the circuit and left-click to start a wire.
- Move parallel to the grid, horizontally or vertically.
- Left-click again to create a corner if necessary.
- Left-click again to end the wire, usually on another component or an existing wire.
- Repeated until the wiring is complete. Exit the wire drawing mode as usual with Esc or right-clicking and choosing End Mode. Unwanted wires can be highlighted and deleted.

Your circuit should now resemble figure 12 on the next page. Here are some tips for wiring.

- Capture adds blobs to junctions where wires are joined. You can see one for the ground connection in figure 12. There should not be blobs anywhere else – on the ends of components, for instance. If there are, remove them with the Junction tool from the toolbar or Place > Junction.

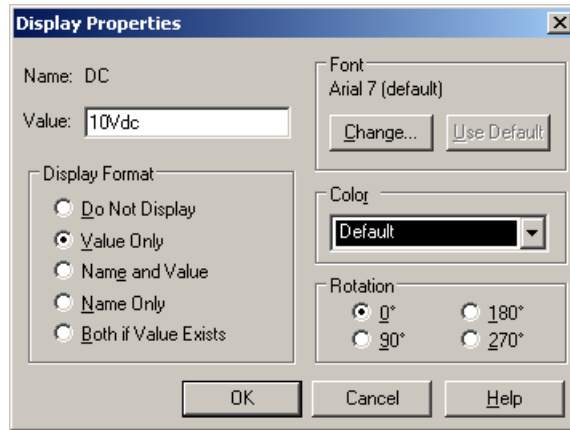


Figure 11. Dialogue box for changing the value of a component, VDC here.

- Join wires in threes at T-junctions, never with four wires in a cross. The problem is that one of the four wires may not be connected but you can't tell. This is standard practice for circuit drawings.
- Always join components with wires, not by placing them next to each other so that their pins overlap. Again you can't tell whether the connection has been made correctly.

Now the circuit is drawn and it is time to simulate it.

3.4 Simulation of a single situation – bias point

The first step in a simulation is to create a *simulation profile*, which contains the instructions to PSpice. Choose PSpice > New Simulation Profile or use the button, which brings up the dialogue box in figure 13 on the following page. Each profile needs a name, which is used to identify the plots. Leave Inherit From as none – this is used if you want to base a new simulation profile on an existing one. Click Create and you will see the Simulation Settings dialogue box in figure 14 on page 13. This is where you tell PSpice what to do. It has several tabs but you rarely need to look at anything except Analysis.

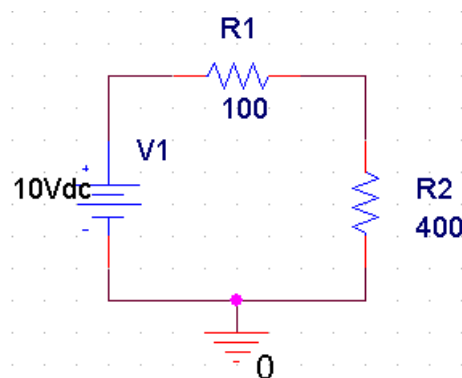


Figure 12. Final circuit of potential divider, fully connected, with the correct values.

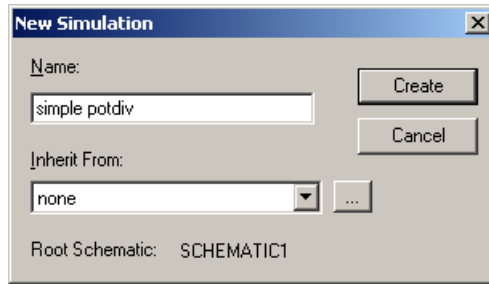


Figure 13. Dialogue box to create a new simulation profile.

The most important choice is the Analysis type. We want only a single simulation for the potential divider because it is so simple and this is called Bias Point. Choose this, as in figure 14 on the following page, and click OK.

Hint for demonstrators. If the PSpice menu and toolbar are missing, the student probably forgot to choose ‘Analog or Mixed A/D’ when the project was created. With versions before 16.3, the only remedy is to copy and paste the circuit into a new project with the correct type. In version 16.3 you can change it: right-click on the design (with extension .dsn) in the project manager and choose Change Project Type. 🐛

Now you can run the simulation with PSpice > Run or the arrow on the toolbar. A dialogue box should flash by and a new window opens for PSpice A/D. The lower left pane of this contains a log, which should conclude with Simulation complete. Most of the window is intended for a graph but there is nothing to plot here, so close the application and return to Capture.

Click on the **V** button in the toolbar (it may be active already), which displays the voltage at each node of the circuit. Clicking on **I** shows the currents in the same way. The result for the potential divider is shown in figure 15 on page 14. You should be able to confirm that PSpice has applied Ohm’s law correctly!

3.5 Slightly more complicated simulation – DC sweep

The first simulation was unusual because it was only a single situation. Usually we examine the behaviour of a circuit while something varies. The variable might be time (transient analysis), the frequency of a signal (AC sweep) or the magnitude of a voltage (DC sweep). We’ll look at a DC sweep now.

Create a new simulation profile as in figure 16 on page 15. This time choose a DC Sweep for the type of analysis. The Sweep variable should be a voltage source and we have to tell PSpice the name of the source that we wish to vary. Copy the name from your schematic drawing; it’s V1 here. (It does not produce a list of available voltage sources automatically.) The Sweep type should be Linear and I chose a range from –10 to 10 with an increment of 1 (these are all voltages but the units are not needed). Click OK and run the simulation as usual.

You should see the PSpice A/D window as before. This time there should be a plot in the window with a horizontal scale to match the variable in the simulation, V1 here. We must now tell PSpice what to plot on the vertical axis. This can be done from within PSpice by choosing

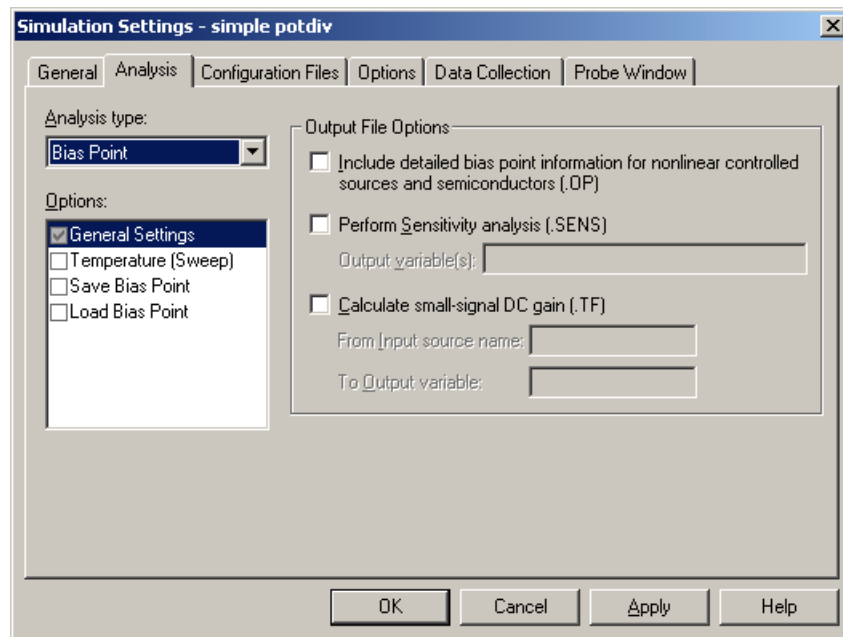


Figure 14. Simulation settings for the simple potential divider.

Trace > Add Trace... from the menu bar but it's easier to return to the Capture window, leaving PSpice open. Turn off the **V** button if the original values are still displayed – they get in the way. Choose the Voltage Probe tool (something like an oscilloscope probe, or select PSpice > Markers > Voltage Level from the menu bar) and click on the wire that joins the two resistors as in figure 17 on page 15. The probe will change colour. Return to PSpice and you will find a line whose colour matches that of the probe.

Make the PSpice window active again and you should see a line on the plot, as in figure 18 on page 16. This shows the voltage at the selected node as a function of V1.

There are many options to control the plot, selected from the menu or button bars. You can also edit the simulation profile and repeat the simulation from PSpice, which is convenient if you find that the range is wrong.

Cursors are useful if you want exact readings from a plot. Choose Trace > Cursor > Display from the menu or click the button. This will generate two sets of cross-wires which can be moved across the graph using the mouse or the ← and → keys. A dialogue box shows the numerical values at the intersection of the cross-wires. Experiment with the left- and right-click buttons on the mouse to see how the two sets of cross-wires can be manipulated back and forth. The right-hand part of the lower toolbar becomes active for the cursors. It helps you to locate maxima, minima or other features. The rightmost button generates a label to show the values at the selected point.

You can annotate the graph with your own text labels. Choose Plot > Label > Text or use the button. Type the required label and Close. Move the text box to the desired location and left-click the mouse to place it.

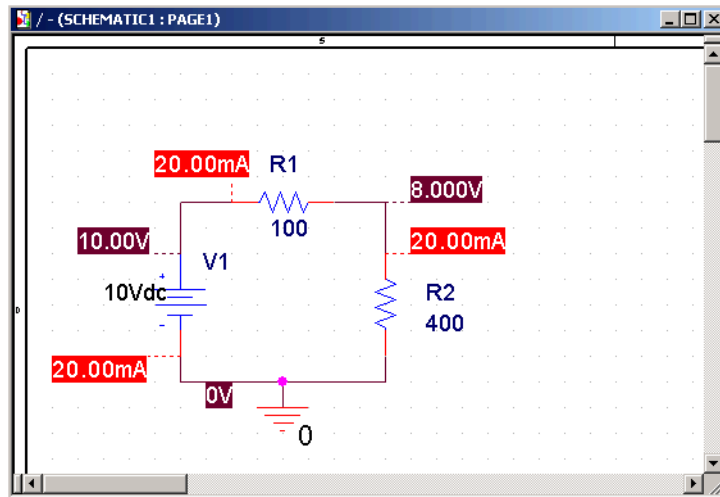


Figure 15. Simulated voltages and currents for the simple potential divider.

3.6 Vary the load resistor – Parametric sweep

You very often want to investigate how the performance of a circuit depends on the value of a particular component. You therefore want to run the simulation several times for different values of that component so that you can compare them. This is called a *parametric sweep* and is clumsy in OrCAD. However, it is used so often that you need to learn how to do it.

Stick with the potential divider. Suppose that V1 and R1 act as a Thévenin voltage source and that R2 is a load. Let's investigate how the voltage that we plotted in figure 18 on page 16 depends on the value of the load resistor.

There are three steps for using a parameter: create the parameter, which is stored in a PARAM block, use it in the circuit, and vary it in the simulation.

1. First, create a parameter for the resistance of the load, which I've called Rload.
 - Place a PARAM component on your schematic from the special library. This isn't a real component – just a place to hold the parameters that we create.
 - Double-click the PARAM block or right-click it and choose the **Edit Properties...** contextual menu item. This brings up the Property Editor or spreadsheet for the component, shown in figure 19 on page 16. Sometimes it comes up as a row rather than a column, If this happens, right-click in the empty cell at the top left (to the left of A) and choose **Pivot**.
 - Choose **New row...**, enter the name of the parameter (Rload) and default value (use its previous, fixed value of 400). Click **OK** to get rid of the dialogue box. The row will now appear in the spreadsheet.
 - The parameter does not appear on the schematic by default (sigh...) so you must select the newly added row/column in the spreadsheet, click the **Display...** button, select **Name and Value** and finally click **OK**.
 - Close the spreadsheet by clicking the **Close** box, which saves the new values.

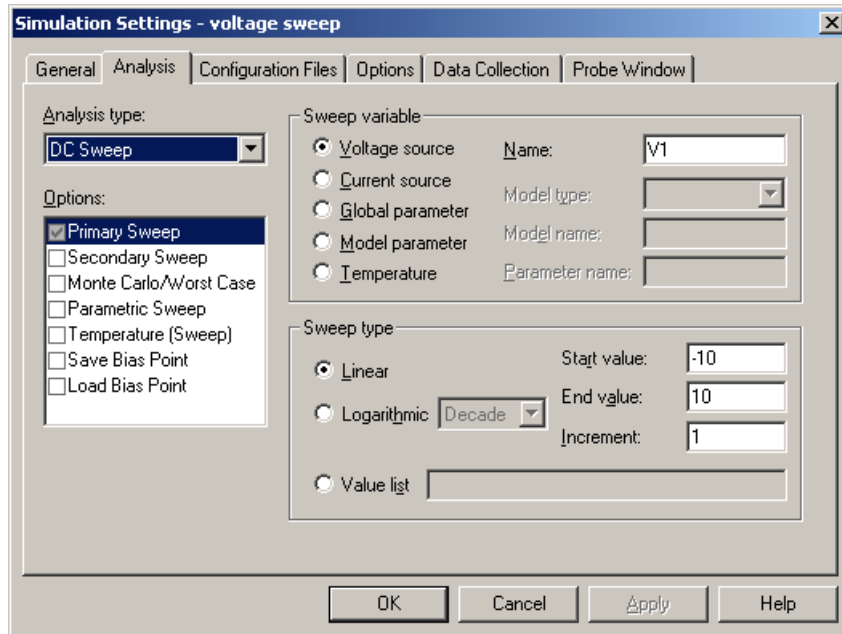


Figure 16. Simulation profile for a DC sweep.

2. The next step is to use this parameter for the load resistance.
 - Double-click on the value of R2, which brings up a box like that in figure 11 on page 11.
 - Change the value from 400 to {Rload}. This is the name of the parameter *in curly brackets*. Click OK. The schematic drawing should now resemble figure 20 on page 17.
3. Finally, we must tell Spice to vary this parameter in the simulation.
 - Create a new simulation profile. This time it is a good idea to base it on the previous profile, because we just want to add the parametric sweep to it, so select voltage sweep in the Inherit From list as in figure 21 on page 17. Click OK, which brings up the usual Simulation Settings box shown in figure 22 on page 17.

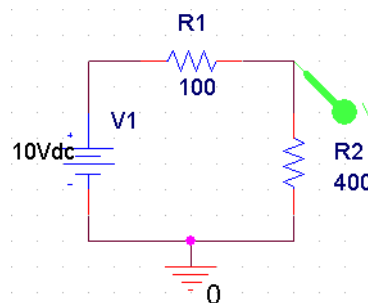


Figure 17. Potential divider with a voltage probe.

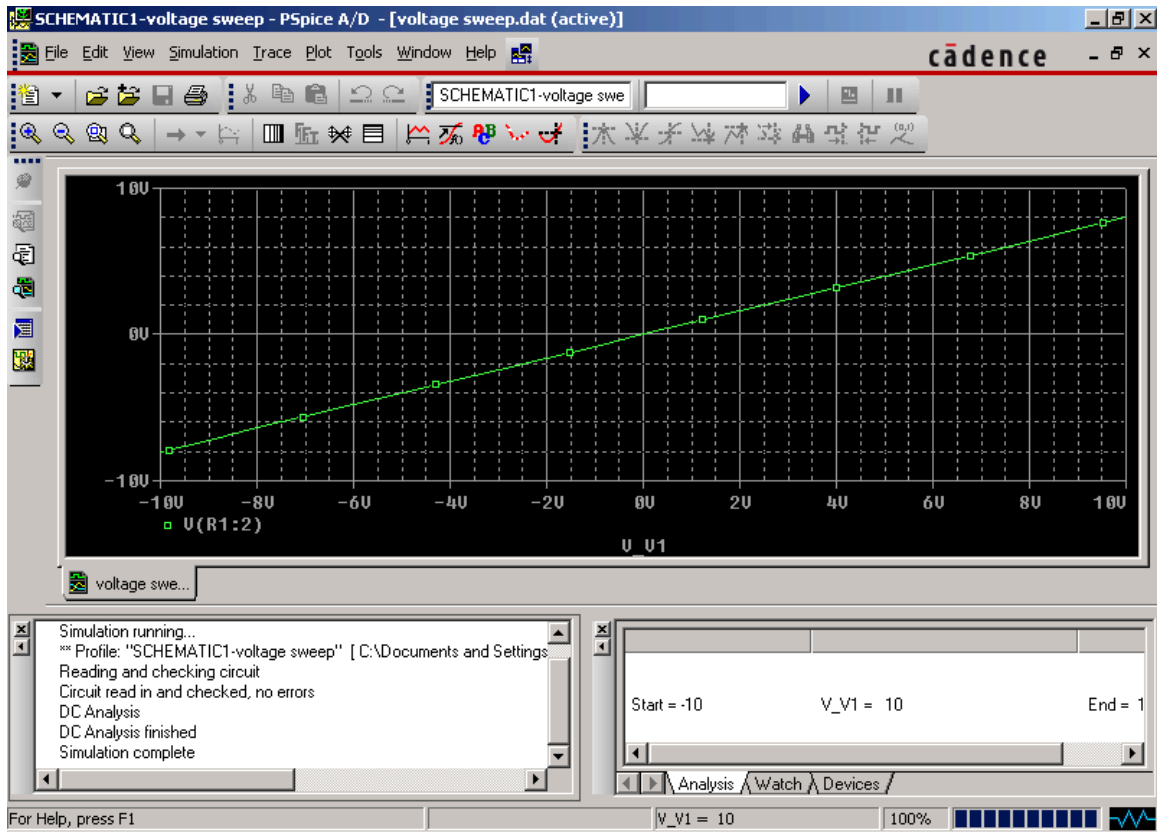


Figure 18. PSpice A/D with a plot of the voltage in the potential divider.

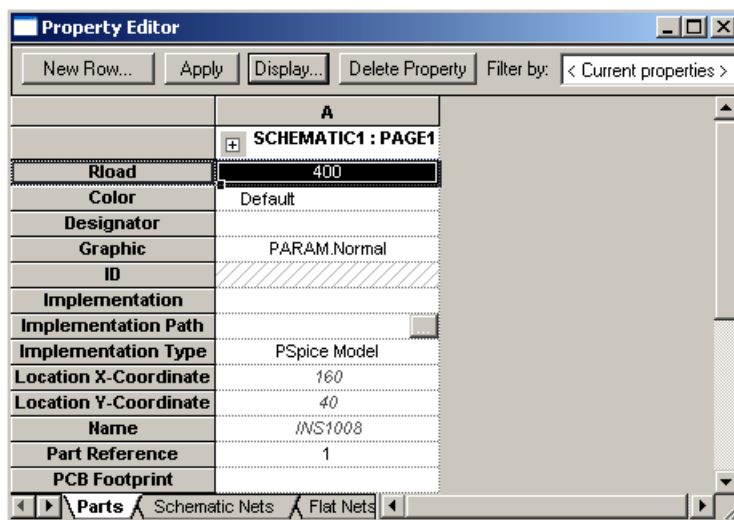


Figure 19. Property editor for a parameter block. A new parameter Rload has been added with a default value of 400.

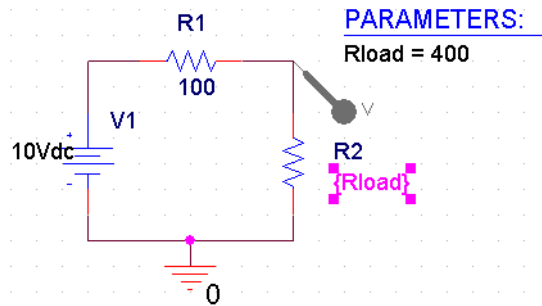


Figure 20. Potential divider with the value of R2 given by the parameter Rload. Don't forget the curly brackets!

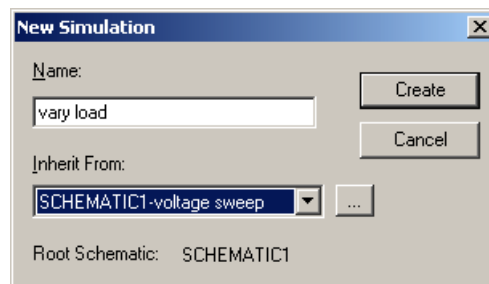


Figure 21. Dialogue box to create a new simulation profile, vary load, based on the previous profile voltage sweep.

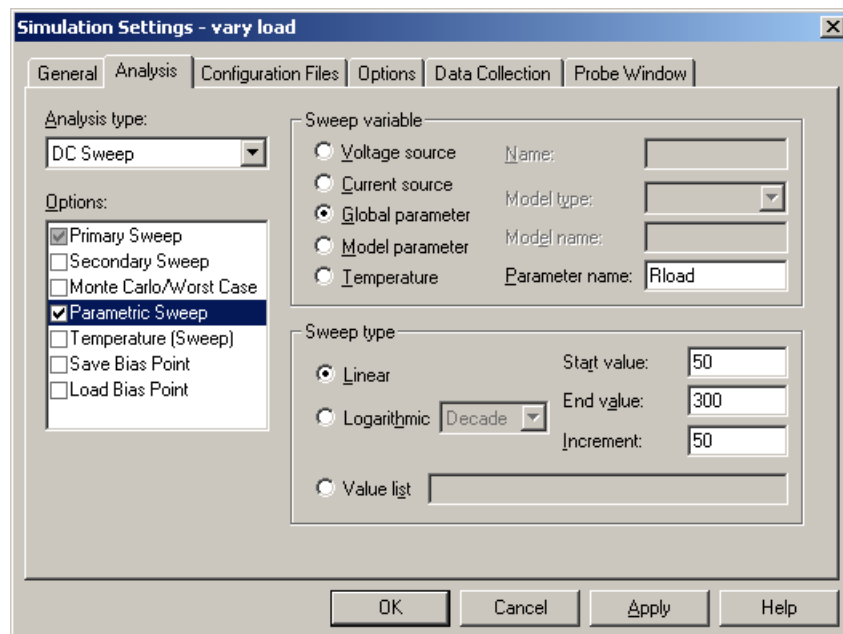


Figure 22. Simulation settings for adding a parametric sweep to the previous DC sweep.

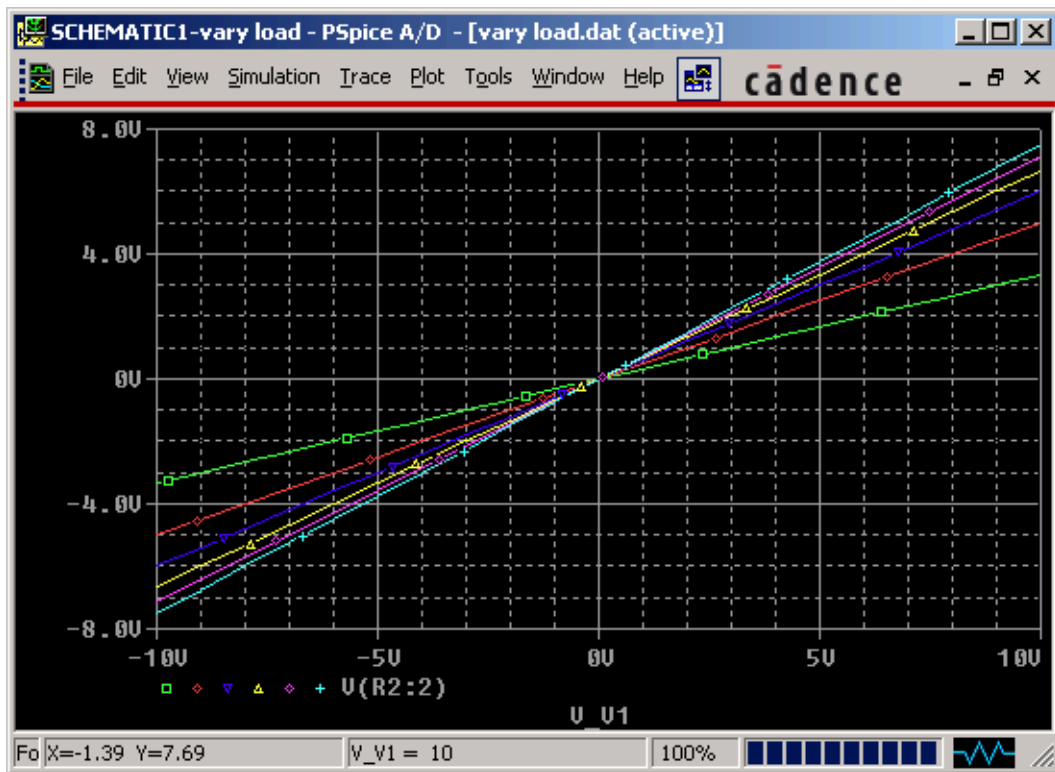


Figure 23. PSpice A/D with a plot of the voltage in the potential divider for each value of the load resistance. This is the Alternate Display, selected by clicking on the last menu item.

- Leave the existing DC Sweep as it is and click on the Options box for Parametric Sweep. Select Global Parameter for the Sweep variable and enter its name, Rload. *Type the name in exactly the same way whenever you use it – don't insert spaces or change letters between UPPER and lower case. Curly brackets are not needed here.*
- Choose a Linear sweep with a Start value of 50, End value of 300 and increment of 50. This means values of 50, 100, 150, ..., 300. These will override the default value of 400 in the parameter block. Finally, click OK.

Run the simulation. Spice will perform a DC sweep for each value of the parameter Rload. It presents you with a dialogue box called Available Sections when the simulation has finished so that you can select which curves you want to plot. You want all of them so click OK. The plot then shows a family of lines as in figure 23 instead of a single one. I hid all the text boxes in the window by clicking on the last menu item, which has only an icon rather than a name, and gives the Alternate Display shown here.

An obvious question is: Which curve goes with which parameter? There seems to be no way of displaying a legend. The simplest way is to double-click on one of the symbols below the plot. This brings up a box with information about the simulation, including the value of the parameter.

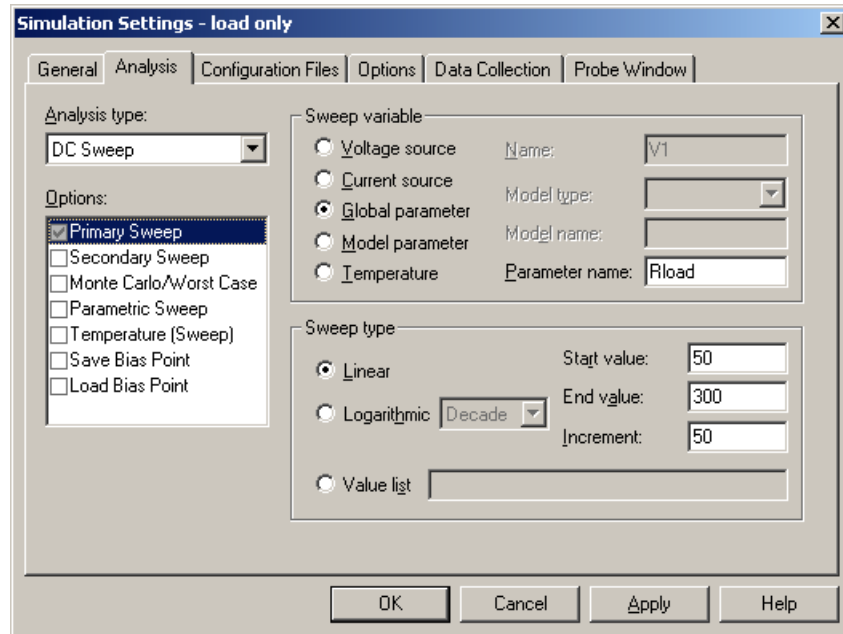


Figure 24. Simulation settings for a parametric sweep alone. The Analysis type is DC sweep.

3.7 Parametric sweep only

In the previous simulation we performed a DC sweep for each value of the parameter. In fact the DC sweep is pointless because it just gives a straight line. We could instead plot the voltage as a function of R_{load} , keeping $V1$ constant at its original value of 10 V. In other words, we vary the parameter alone. This isn't done very often but is appropriate here. Figure 24 shows a suitable simulation profile. It is set up as a DC sweep although we do not sweep either a current or voltage. It looks rather like figure 22 on page 17 except that the parameter is used for the primary sweep, not the parametric sweep.

Figure 25 on the following page shows how the voltage depends on R_{load} . The curve would be smoother if more points were requested by reducing the increment in figure 24.

Modify the plot to show the *power* dissipated in the load resistor, rather than the voltage across it. This is most easily done with the probes in Capture. What value of R_{load} gives the maximum power dissipation? You may need to run the simulation with more points to get an accurate value. Does the result agree with theory?

Hint for demonstrators. ☺

4 Frequency response of an RC circuit

We'll now use Spice to analyse the behaviour of a simple circuit as a function of frequency. This is the RC filter shown in figure 27 on page 21. A *filter* is typically used to remove certain frequencies from a signal or pick out a frequency of interest. For example, a low-pass filter allows signals with low frequency to pass unaffected but attenuates (makes weaker) high-

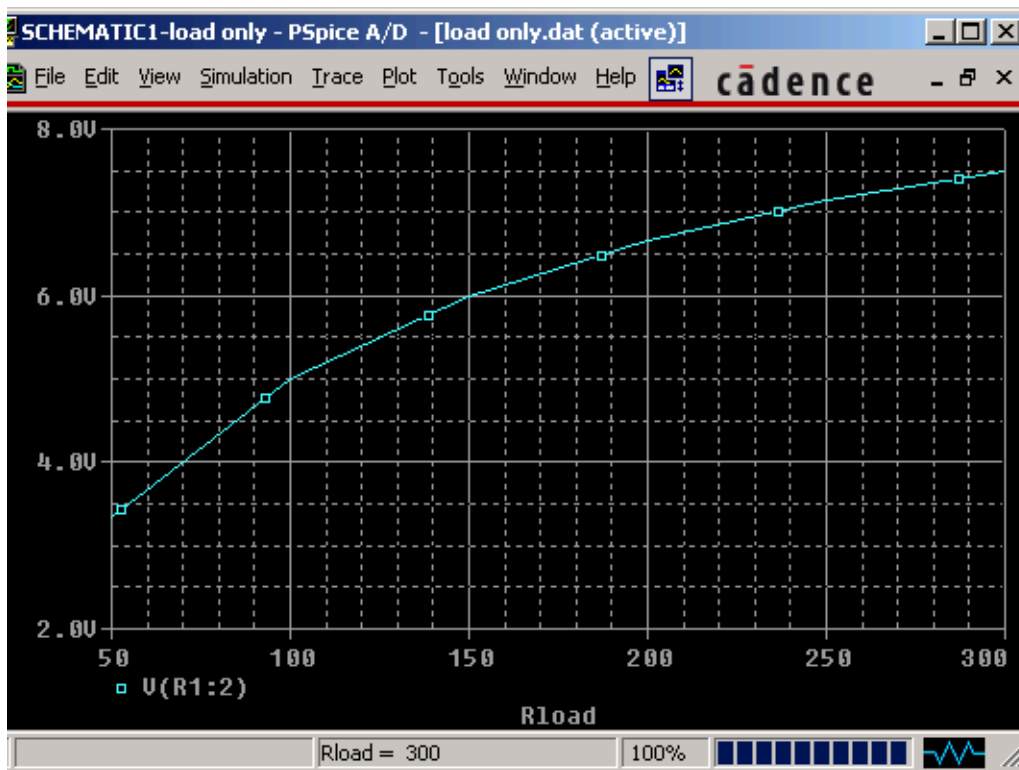


Figure 25. Plot of voltage as a function of Rload.

frequency signals. This might be used to remove high-frequency noise from a signal that is known to vary slowly – the temperature of a room, for instance.

First, analyse the behaviour of this circuit by hand (assuming that you have done this in the lectures). Follow these three steps.

1. What are the limiting forms of the impedance of the capacitor at high and low frequency?
2. From these, work out the behaviour of the filter at high and low frequency. Is it a high-pass or low-pass filter?
3. What is its *half-power*, -3 dB or *Bode* frequency, which separates ‘low’ and ‘high’ frequencies?

Hint for demonstrators.

1. Zero impedance at high frequency, infinite impedance at low frequency.
2. The infinite impedance of the capacitor blocks low frequencies. At high frequency it allows the signal through unaffected. This is a high-pass filter.
3. The *half-power* or *Bode* frequency is given by $\omega_B = 1/RC$ or $f_B = 1/2\pi RC$. Here $R = 10\text{ k}\Omega$ and $C = 10\text{ nF}$ so $f_B = 1.6\text{ kHz}$.

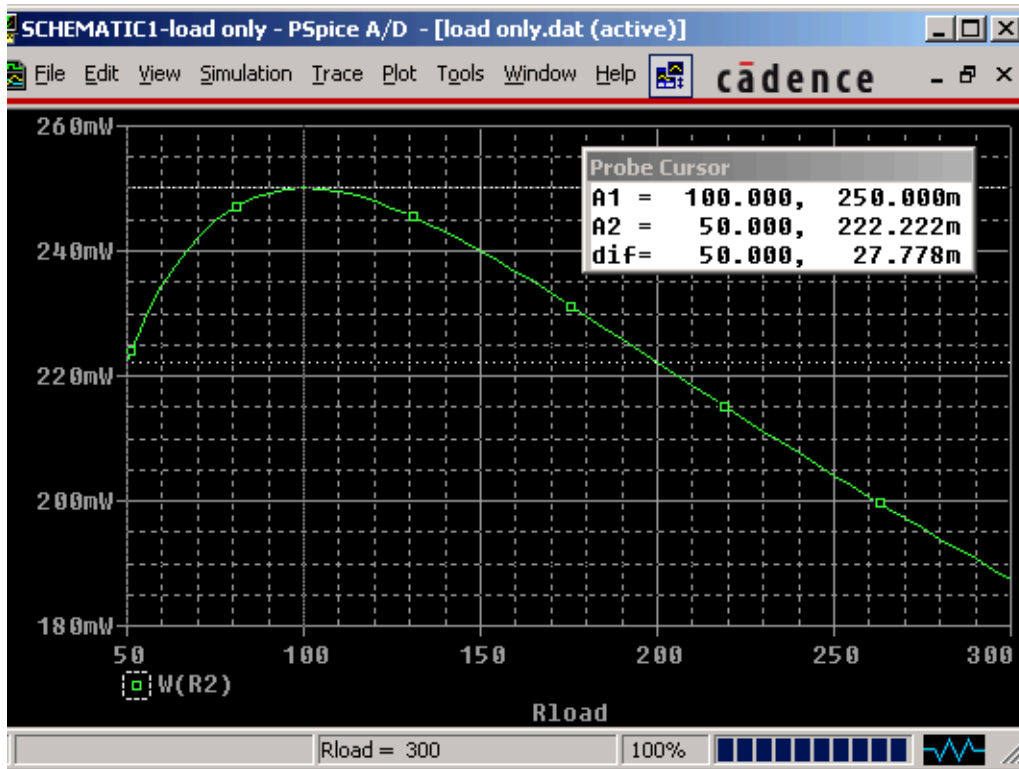


Figure 26. Plot of power dissipated in R2 as a function of its value, Rload. I activated the cursor and used the ‘maximum’ tool on the right-hand button bar to find the peak value.



Now simulate the filter. Create a new project in Capture (remember to make a new directory first) and draw this circuit.

- The source is a VAC and is in the source library. I changed its name from V1 to Vin, which is more descriptive. Leave the voltages at their default values.
- The symbol on the right is an *off-page connector*. This is really intended for carrying

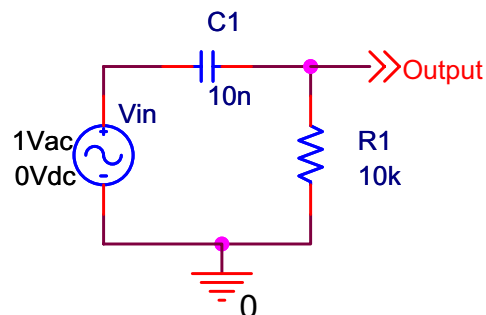


Figure 27. Filter formed by a resistor and capacitor.

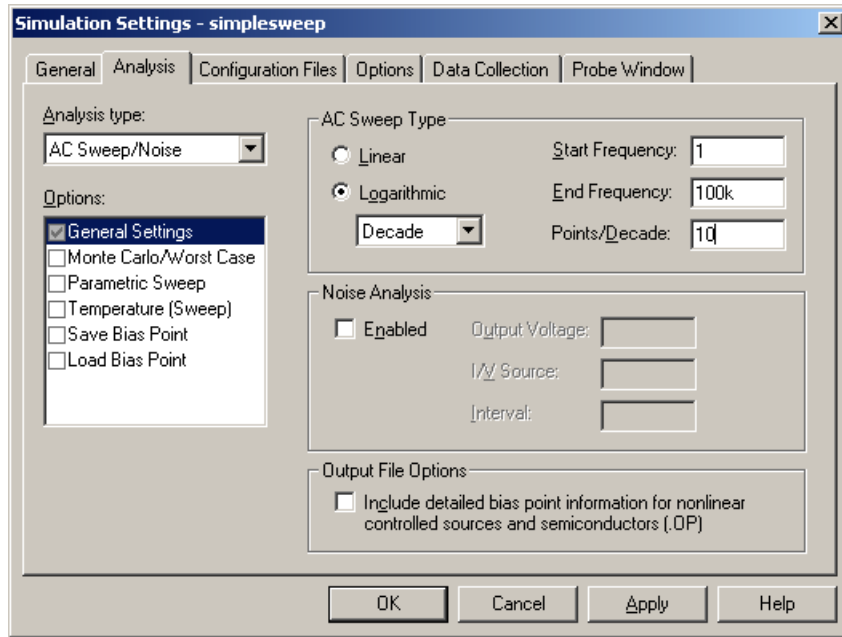


Figure 28. Simulation settings for an AC sweep.

signals from one page of a large drawing to another but is a clear way of naming signals on the circuit. You can find it on the **Place** menu or use the button.

- Don't forget the ground symbol for the zero node.

The simulation settings are shown in figure 28. The Analysis type is AC Sweep/Noise. Make sure that the AC Sweep Type is Logarithmic by Decade, which should be the default. Frequency sweeps are almost always logarithmic because this shows low and high frequencies equally well. Make sure that the range is suitable, based on the Bode frequency that you calculated above. My entries may not be correct!

Put a voltage marker on Output and run the simulation. You should see a plot of the voltage as a function of frequency, rising to 1 V at high frequency (yes, it's a high-pass filter). The frequency axis is logarithmic, which means that successive labels have the values 1, 10, 100. . . .

The behaviour of the circuit is clearer if the voltage axis is also logarithmic. There is a button for this or you can choose **Plot > Axis Settings. . .** from the menu bar. This makes the low-frequency part of the plot into a straight line, as in figure 29 on the following page, which shows that voltage \propto frequency here. I made the line thicker by selecting the trace, right-clicking and choosing **Properties. . .** Thicker lines (I usually choose the third) look ugly on the screen but give a much better printout.

The response of filters is traditionally plotted in decibels (dB). These are defined by

$$\text{voltage gain in decibels} = 20 \log_{10} \left| \frac{\text{output voltage}}{\text{input voltage}} \right|. \quad (1)$$

The output of a filter often has a smaller magnitude than its input, in which case the 'gain' is less than unity and becomes negative in decibels.

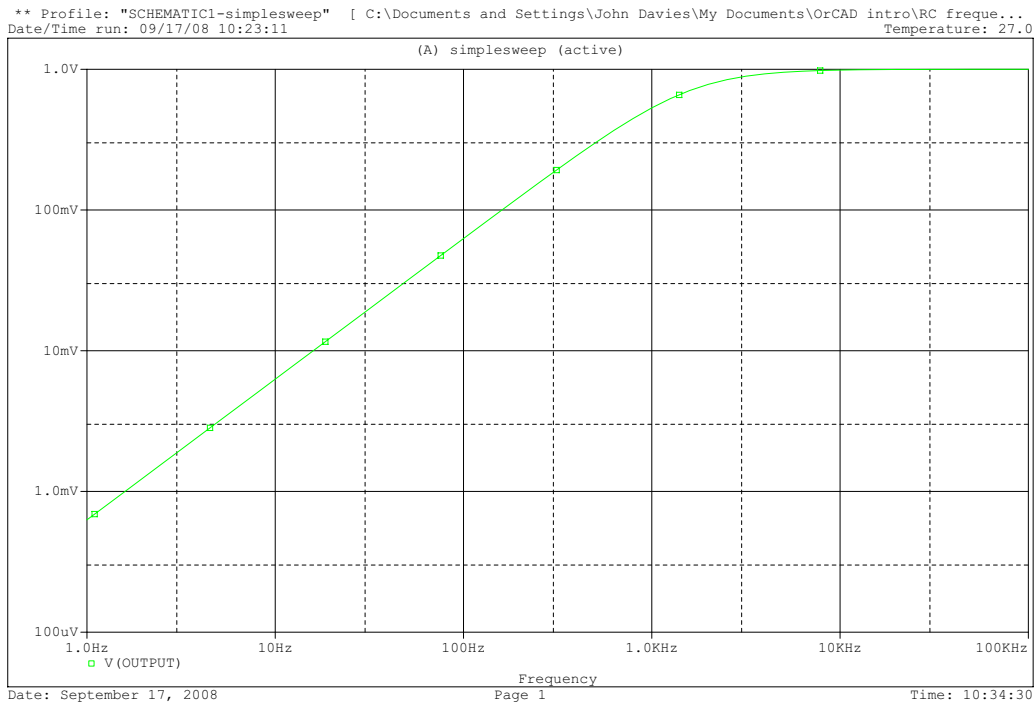


Figure 29. Amplitude of the output from a high-pass filter as a function of frequency. Both axes are logarithmic. This is from a printout rather than a screenshot, hence the white background.

Here is the easiest way of getting a plot in decibels. Return to Capture, remove the existing voltage marker, and choose PSpice > Markers > Advanced > dB Magnitude of Voltage from the menu bar. Place the marker on the output, run the simulation again and you will get a plot in decibels. (This works correctly only if the input voltage is 1 Vac.)

You should know that AC signals also have a *phase*. A filter changes the phase of a signal as well as its magnitude and it can be useful to plot this too. Again Capture has a built-in tool. Choose PSpice > Markers > Plot Window Templates... from the menu bar. This gives a list of plots that can be produced automatically. Try Bode Plot dB - separate and place the marker on Output as usual. Run the simulation and you will see a pair of plots for magnitude and phase as in figure 30 on the next page.

The scales of both graphs are poorly chosen by default. Make the plot clearer by changing them. Either double-click on the axis or choose Plot > Axis Settings... from the menu bar. Select User Defined for the Data Range and enter more appropriate numbers. You might like to change the grid as well.

At the half-power frequency, the filter should have a gain of -3 dB and a phase of 45° . (You should be able to derive these numbers.) Check this on the plot.

Add a parametric sweep to the plot for capacitors of 1, 10 and 100 nF. Explain how the behaviour changes.

Hint for demonstrators. The curves look the same but are shifted horizontally along the frequency axis according to the values of their half-power frequency. Increasing the value of the capacitor by a factor of 10 moves the curve down in frequency by the same factor. ☺

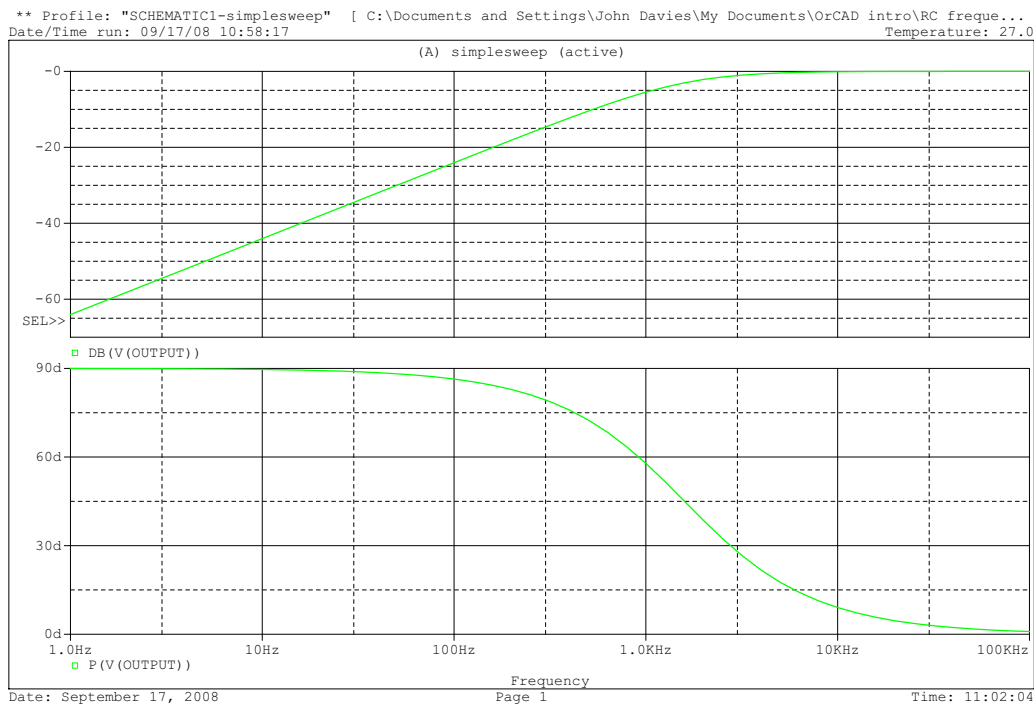


Figure 30. Amplitude in dB and phase (Bode plot) of the output from a high-pass filter as a function of frequency.

How does the circuit behave if the capacitor and resistor are interchanged?

Hint for demonstrators. It becomes a low-pass filter with the same half-power frequency. 🧠

5 Transient response of an RC circuit

The frequency response of a circuit is helpful to understand its behaviour when it is driven with simple sine waves, or signals that can be constructed from them. In other cases it is clearer to look at the behaviour in time. For example, how does the circuit respond to a sudden step or pulse on the input? Figure 32 on the following page shows the same RC circuit but with a different source. The idea is that we suddenly connect the circuit to the battery at $t = 0$, so its input voltage jumps from zero to V_1 , and observe the result at $V_{\text{Output}}(t)$.

For how long should the simulation be run? Typically there is a transient (short-lived) variation, after which the voltages reach a steady state. The duration of the transient behaviour is set by the *time-constant* of the circuit, $\tau = RC$. This is 0.1 ms for the values shown. It is closely related to the half-power frequency.

This time we use the final type of simulation, called Time Domain (Transient). The settings are shown in figure 33 on page 26. I've chosen to run for ten time-constants so that everything should have settled down to a steady value by the end of the simulation.

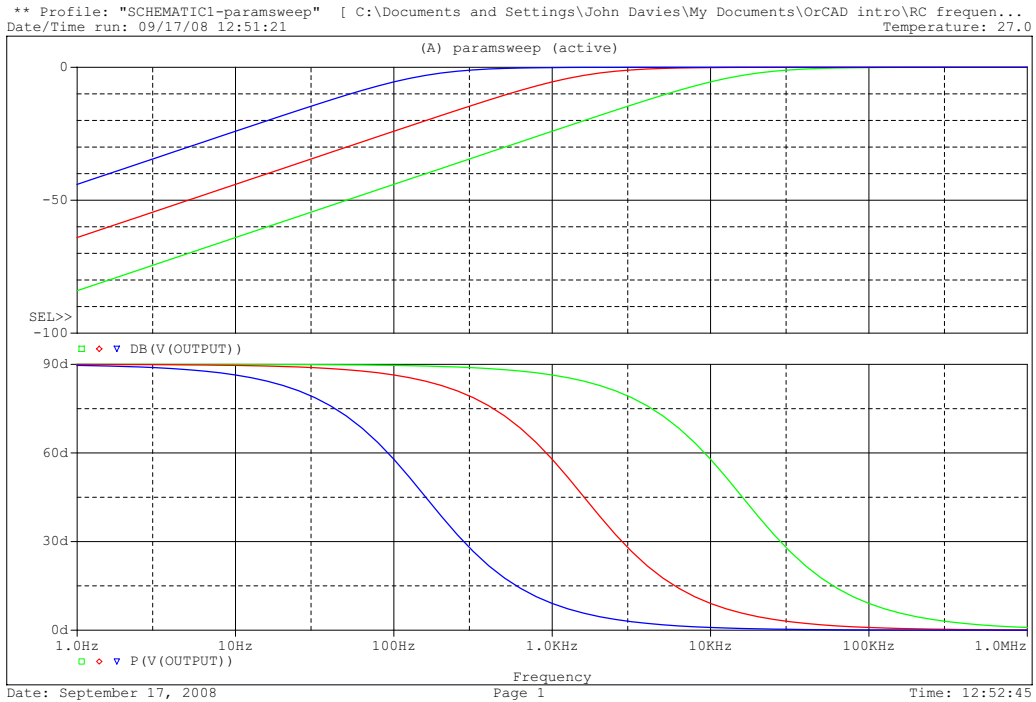


Figure 31. Amplitude in dB and phase (Bode plot) of the output from a high-pass filter as a function of frequency for $C = 1, 10$ and 100 nF.

There's one more step needed because we must specify the initial conditions for the capacitor: How much charge does it hold at the start of the simulation? Assume that we have left the circuit for a long time before connecting the battery, so that the capacitor has discharged. We must therefore set its initial charge to be zero. Select the capacitor and choose Edit > Properties... to open the Property Editor, shown in figure 34 on page 27. Enter zero for IC (initial condition) and close the window.

Now run the simulation. You should get a plot in the usual way but with time along the horizontal axis as in figure 35 on page 27. The curve may be polygonal (obviously made of segments of straight lines) if there are too few points. In this case enter a suitable value in the Maximum step size box of the Simulation Settings (figure 33 on the following page)

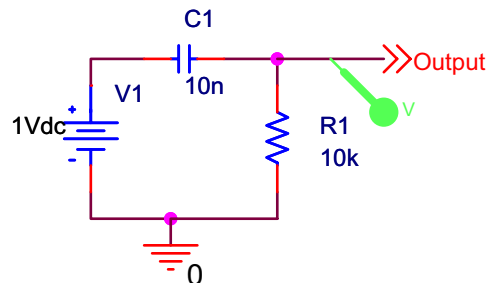


Figure 32. Filter formed by a resistor and capacitor, fed by a constant voltage.

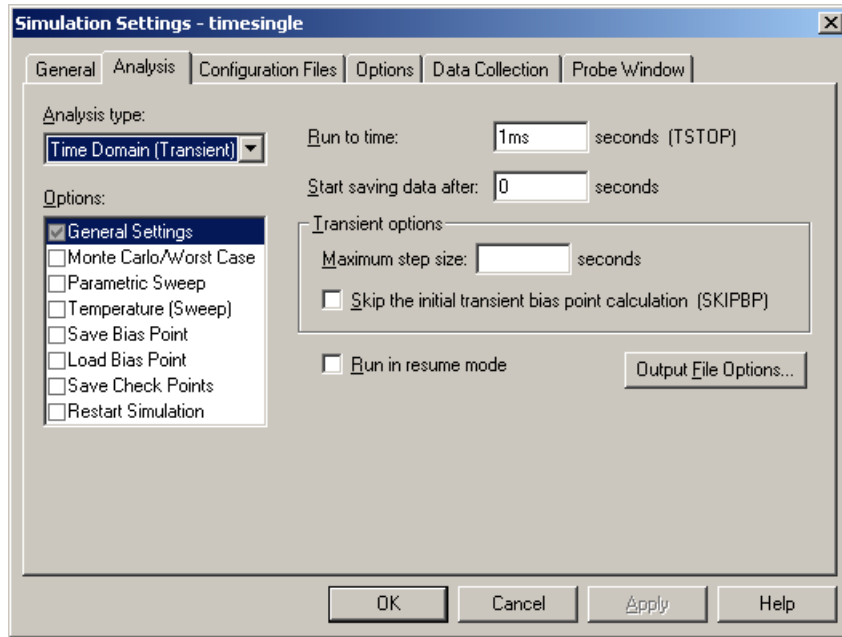


Figure 33. Settings for a transient simulation.

so that Spice calculate the voltage more often. Usually Spice chooses an suitable interval automatically.

One way of finding the time-constant from a graph like this is to extrapolate the initial decay linearly and find the point at which it cuts the time axis. This should give the time-constant directly. Check it for the plot.

The output is more interesting from a pulse rather than a single step. Spice offers the VPULSE source, whose parameters are listed in table 1. Change the source in your circuit to give a 0.2 ms pulse and plot both the input and output. You should find that the output voltage goes negative for a while, although the input is always positive or zero. How is this possible?

Hint for demonstrators. The capacitor tries to charge after the first edge of the pulse, which causes the output voltage to fall. The left-hand plate of the capacitor in the circuit has positive

Table 1. Parameters for the VPULSE source.

name	parameter	notes
V1	value of voltage outside pulse	usually 0 V
V2	value of voltage inside pulse	
TD	delay before pulse starts	
TR	rise-time of pulse	much shorter than width for a square pulse
PW	width of pulse	
TF	fall-time of pulse	much shorter than width for a square pulse
PER	period of train of pulses	leave empty for a single pulse

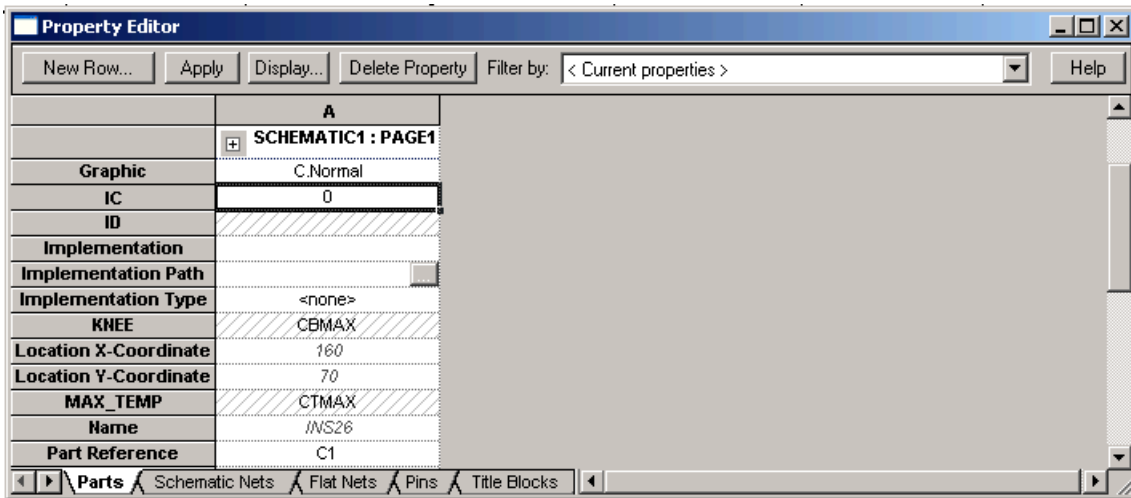


Figure 34. Property Editor for the capacitor with an initial condition (IC) set to 0.

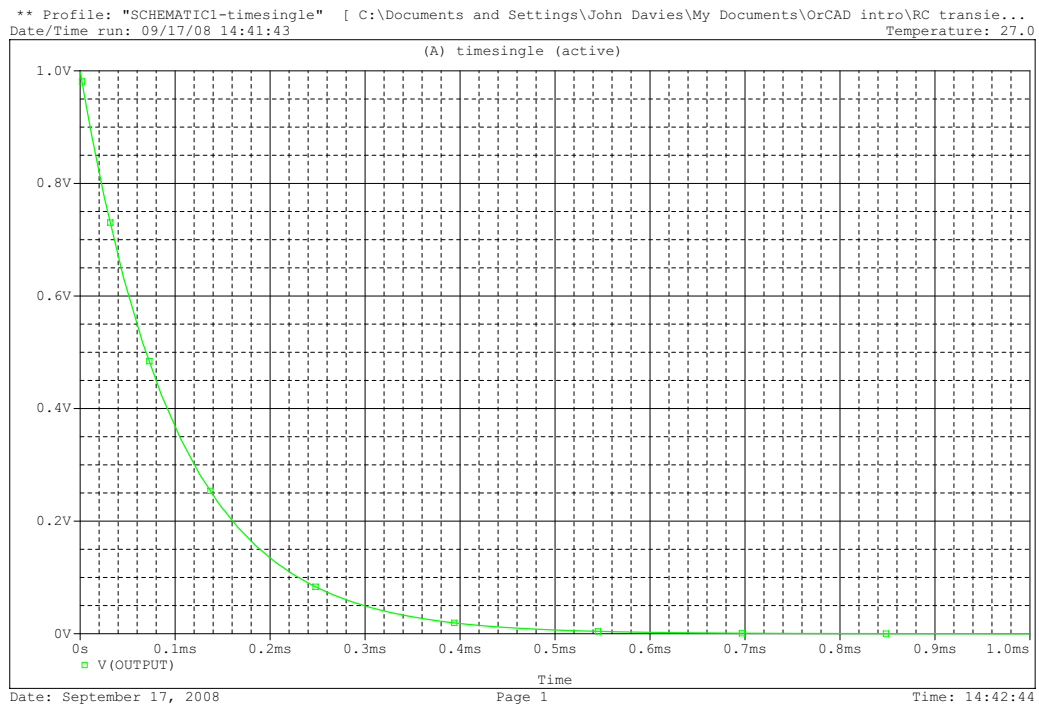


Figure 35. Output voltage from the RC high-pass filter with time-constant $\tau = 0.1$ ms.

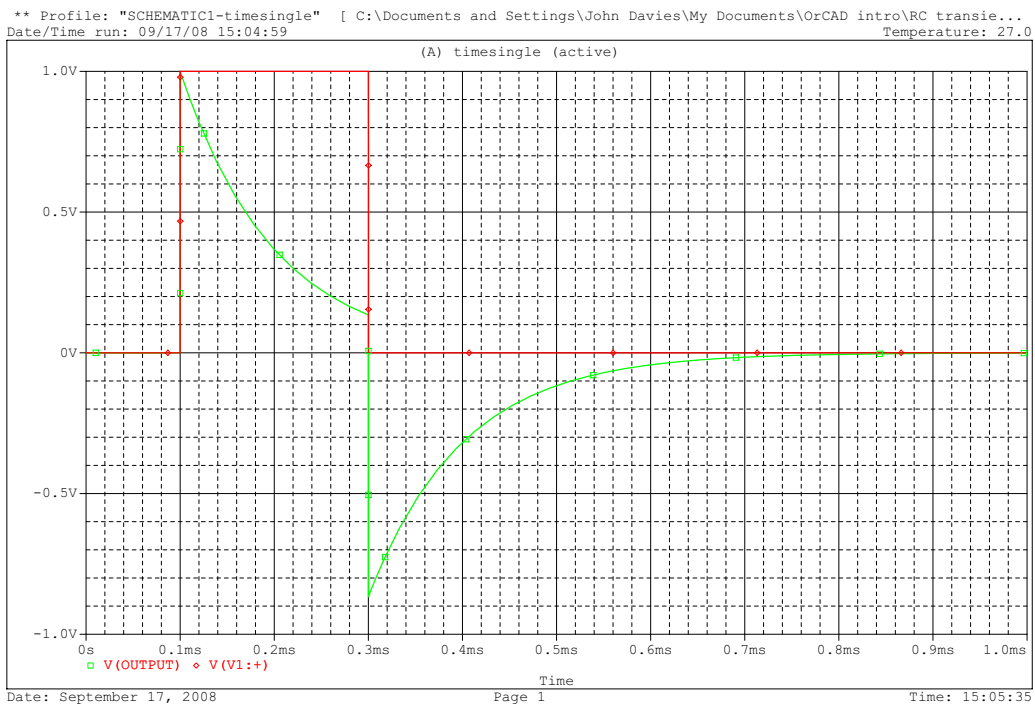


Figure 36. Input and output voltages from the *RC* high-pass filter with an input pulse of 0.2 ms.

charge, with equal negative charge on the right-hand plate. When the second edge of the pulse arrives, the charge on the capacitor cannot change instantaneously so the voltage across it remains the same. The left-hand plate is pulled down by 1 V so the right-hand plate and output are pulled down by the same. The output voltage must therefore become negative.

Another way of looking at the behaviour is to say that the circuit behaves rather like a differentiator. 🍷

Again, how does the circuit behave if the capacitor and resistor are interchanged?

Hint for demonstrators. This time the filter behaves rather like an integrator. 🍷

6 Conclusion

This handout has explained how to use the most common, basic features of Capture and PSpice. It can do far more than this, particularly for analysis. Look in the help files and tutorials to learn more. Another application is to design printed circuit boards, which is described in another handout.