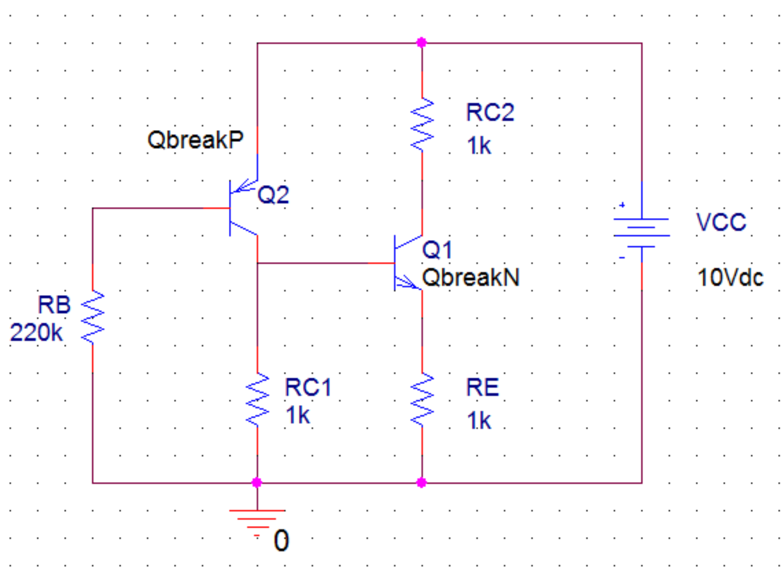
PSPICE tutorial: BJT circuits at DC

In this tutorial, we will examine the use of BJTs in PSPICE. We will use PSPICE to simulate a simple DC circuit that has *npn* and *pnp* transistors. We will simulate the circuit twice, using different values of β_F in the two runs.

This tutorial is written with the assumption that you know how to do all of the basic things in PSPICE: starting a project, adding parts to a circuit, wiring a circuit together, using probes, and setting up an using a simulation profile.

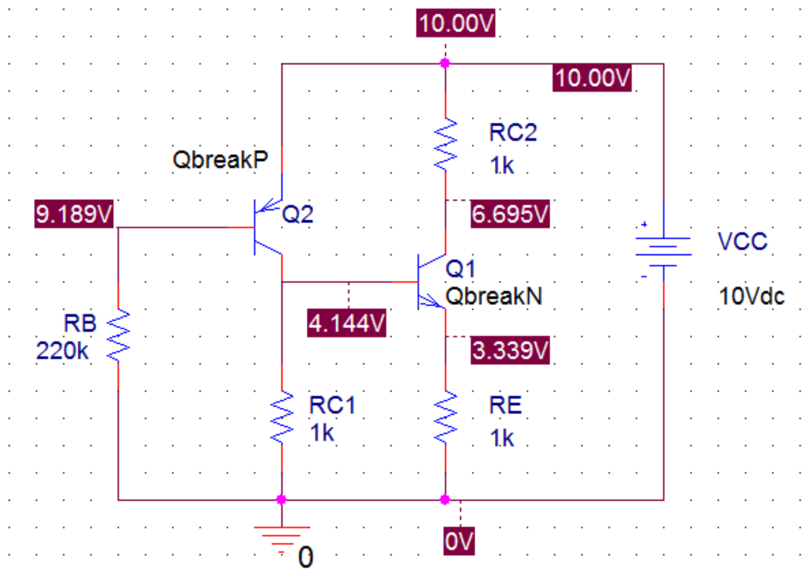
Simulation with default values

Build the circuit shown below. For the BJTs. use the “QbreakN” and “QbreakP” devices from the Breakout library. QbreakN is a generic *npn* transitor with a default $\beta_F = 100$. QbreakP is a generic *pnp* model, also with default $\beta_F = 100$. In the first simulation run, we will use these default values. As always, the resistors come from the analog library and the DC source is from the source library. Note that we could use specific BJT components from the EVAL library, but we don’t necessarily know that model descriptions being used for those. The breakout models allow us to model the components in exactly the way we want.

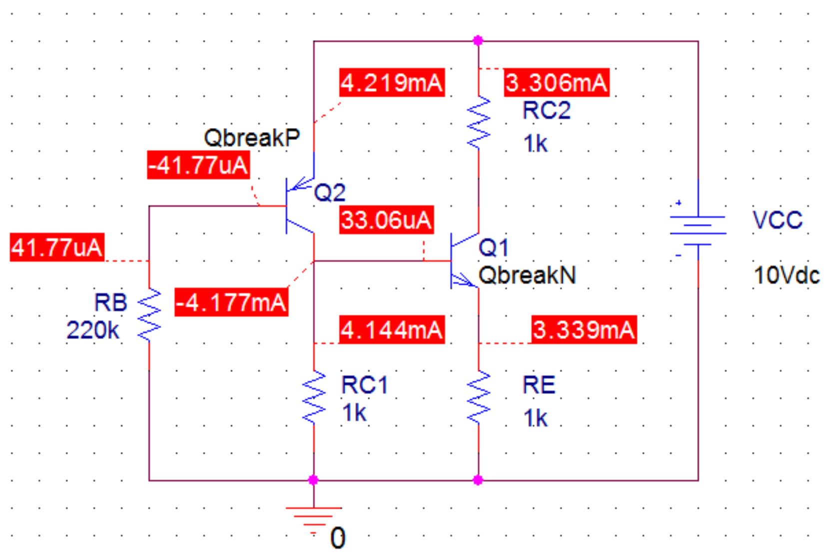


Create a new simulation profile and choose the “Bias point” simulation option from the drop down menu. A bias point simulation simply calculates the DC voltages and currents in the circuit.

Run the simulation. Then display the node voltages. (Click on the “V” icon in the toolbar.) The results will be displayed right on the circuit. If we are picky about how things look, we can move the value labels around to make the arrangement tidier.



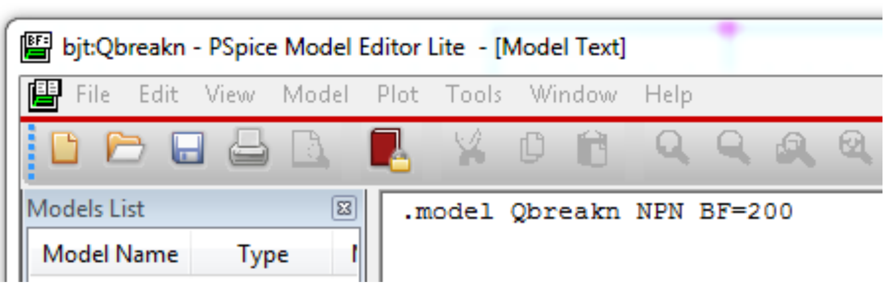
To see the currents, turn click on the "I" icon in the toolbar.



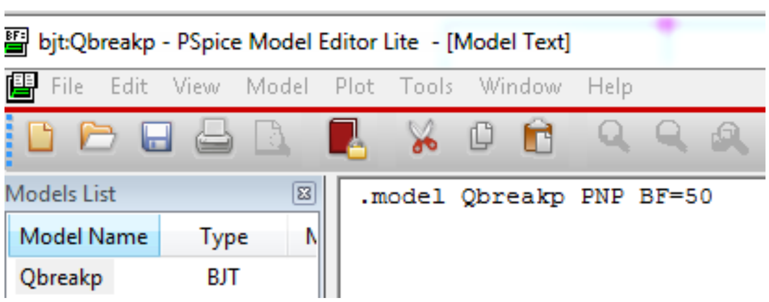
(Editorial notes: We can have currents and voltages displayed simultaneously, but the jumble of information becomes confusing. To keep things cleaner, view the voltages and currents separately. Also, since the currents at every terminal of every device, some information is redundant. For example, we don't need to display both the *npn* collector current and the R_{C2} current – we know that they are the same. In order to make things clearer, some of the superfluous currents labels have been deleted from the figure above, Finally, recall that PSPICE defines the current flowing into each terminal as being positive. So when currents are flowing out, as in the case of the *pnp* base and collector or the *npn* emitter, the value will be negative.)

Changing the PSPICE model

Let's run the simulation again, but with different values of β_F for each transistor. For the *npn*, we will change β_F to 200 and for the *pnp* we will use $\beta_F = 50$ To make these changes, we need to edit the model descriptions for the BJTs. Click on the *npn* symbol in the circuit to select it. Then right-click on the selected symbol to bring up a pop-up menu. From the menu, select "Edit PSPICE model". A dialog box opens that contains one line of text describing the properties of the BJT model. The name is "Qbreakn" – in SPICE "Q" means BJT. Of course, "NPN" defines the type of BJT. To change β_F, type in "BF = 200" after "NPN", as shown below. Save the changes (File -> Save) and close the window. Note that from this point on, every QbreakN transistor that we use in any PSPICE circuit will have $\beta_F = 200$. If we need a different value of β_F – for example, to match a value measured in lab – we will need to go through the same steps to edit the PSPICE model.



Now repeat the same steps to change the PSPICE model for the *pnp* transistor. Add BF = 50 to the model description, as shown below. Be sure to save the changes before re-running the simulation.



Running the simulation again gives the DC values shown below. We see that changing the βs can have significant effect on the DC values. (This shouldn't surprise us.)