

This chapter shows the basics of AC simulations, including small signal gain and noise. It also shows many detailed features of the system.

---

## Lab 3: AC Simulations

## OBJECTIVES

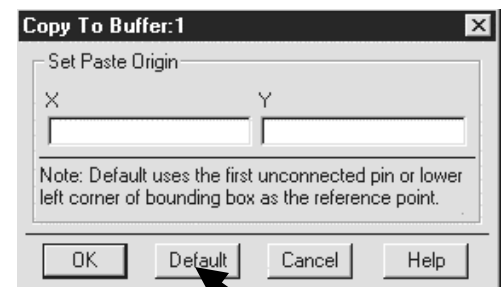
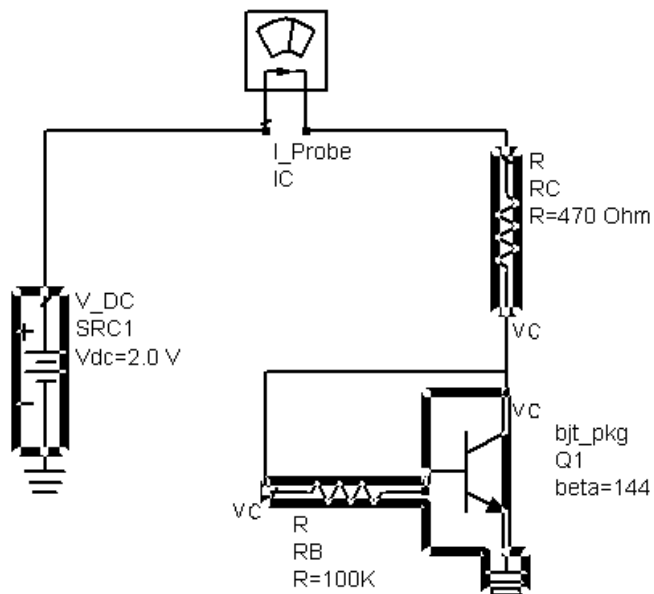
- Perform AC small-signal and noise simulations
- Sweep variables, tune parameters, write equations
- Control plots, traces, datasets, and AC sources

**About this lab:** This lab continues the mixer project and uses the same sub-circuit as the previous lab.

## PROCEDURE

### 1. Use copy/paste to create a design

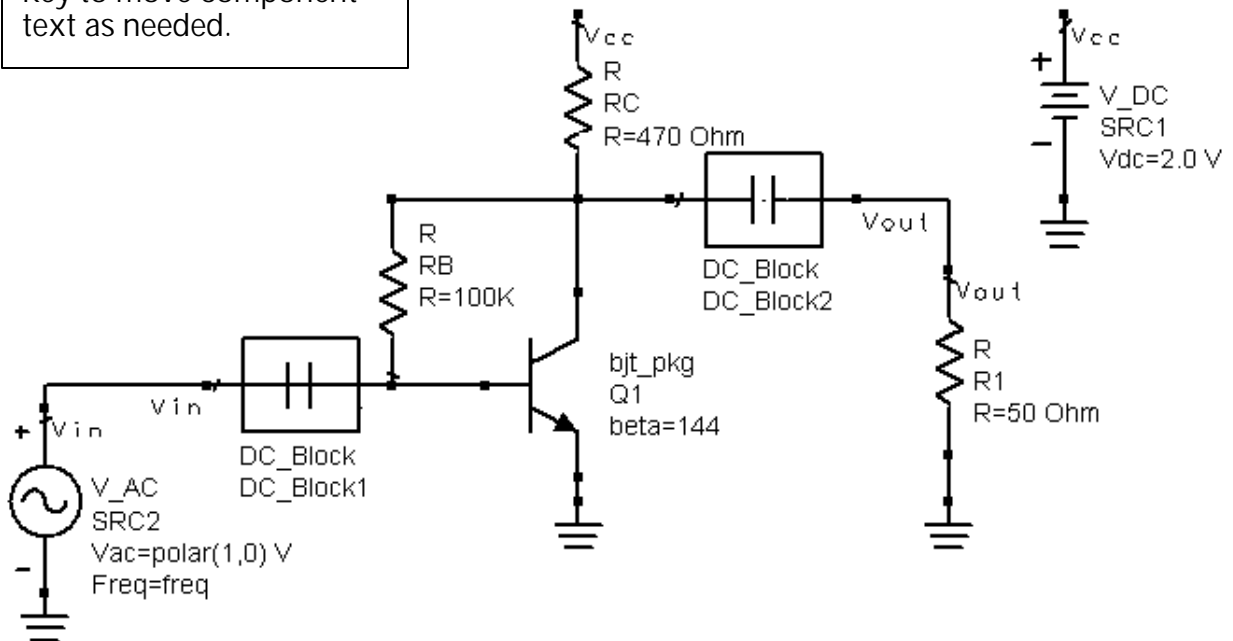
- Open the last design from lab 2 (**dc\_net**) and select (Shift click) the following items: **Vdc**, both bias **resistors**, the **bjt\_pkg**, and the **ground**. Then click: **Edit > Copy / Paste > Copy to buffer**. Select the **Default** origin and then close the window.



- Use the File> New command to create a new schematic window and name it: **ac\_sim**. Then click **Edit > Copy/Paste > Paste from buffer** and insert the ghost image on the schematic.
- Save the new file. You must save it or it will not be written to the disk drive.

- d. Continue building the circuit shown here using the following steps:

NOTE: After inserting a node name, use the F5 key to move component text as needed.

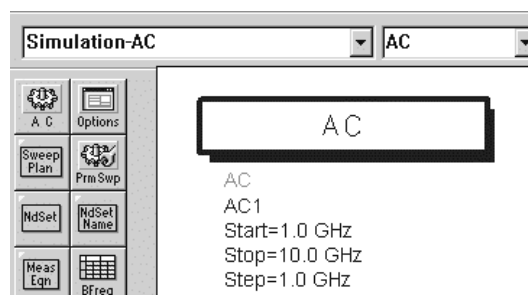


- e. Insert the remaining components: **AC Simulation controller**, **dc blocking capacitors**, and the **V\_AC voltage source**, **50 ohm load**, etc. Use the palettes to find the desired items.
- f. Add **Vcc** as a **Node Name** instead of using a wire.
- g. Add **Vin** and **Vout** as Node Names also.
- h. Select the **bjt\_pkg** and **push** into the sub-circuit (using the icon) to verify that it is your circuit, and then push out again.



## 2. Set up the AC Simulation

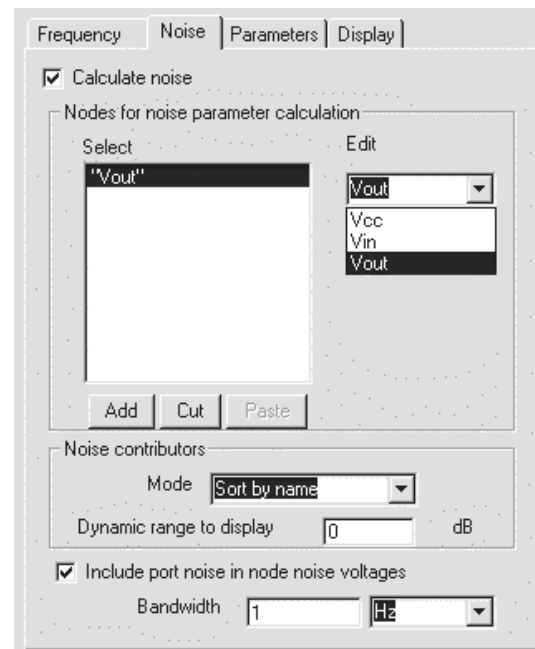
- a. Insert an AC Simulation controller.



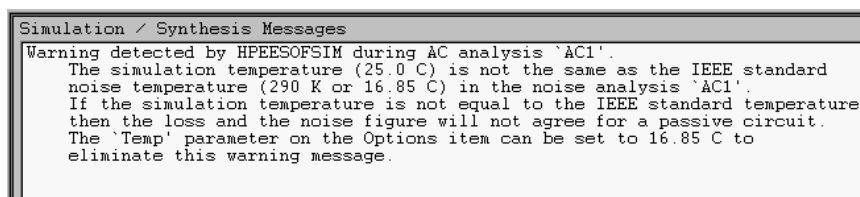
- b. Edit the AC controller start, stop, and step as shown here.
- c. Turn on the **Calculate noise** button and add the **Vout** node. Also, set the Mode to **Sort by Name**. You could sort by value to see the greatest contributors listed first and then list the name in order to locate them on the schematic (good for large circuits)
- d. Turn on the **Display** for each of the parameters.



AC1  
Start=100 MHz  
Stop=2 GHz  
Step=100 MHz  
CalcNoise=yes  
NoiseNode[1]="Vout"



- e. **Simulate** the circuit: press the F7 key (default dataset name is the same as the schematic: ac\_sim). Look at the status window. It should give a warning message like the one here because the default simulation-temperature is room temperature (25° C) and not at the IEEE standard for noise measurements (290° Kelvin).



### 3. Set the *Options* card and Simulate

From the simulation palette, insert the **Options** card. This is a global used for temperature. Set **Temp** to **16.85**. **Simulate again** and there should be **no warning message**. The Options card also sets the tolerance for DC solutions.

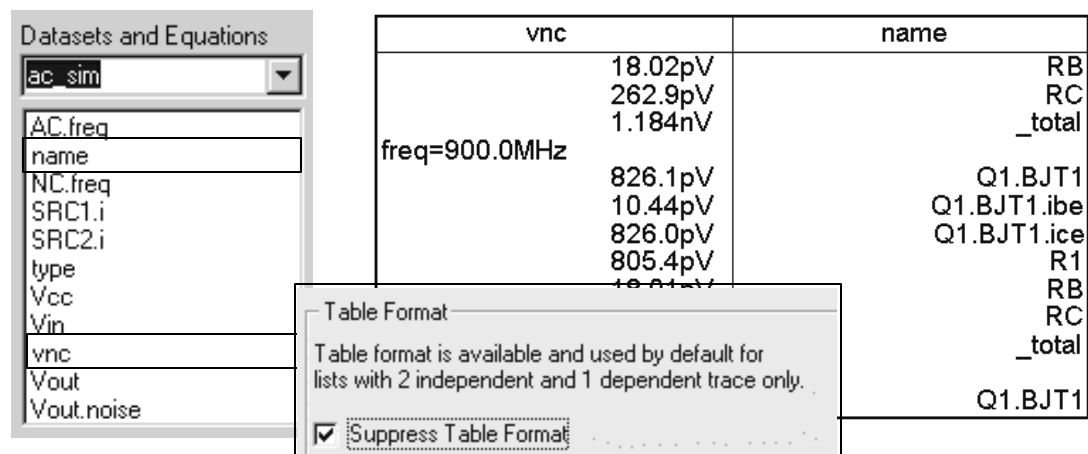


OPTIONS

Options  
Options1  
Temp=16.85  
TopologyCheck=yes  
V\_RelTol=1e-6  
I\_RelTol=1e-6  
GiveAllWarnings=yes  
MaxWarnings=10

#### 4. Display the noise data

- Open a new data display and save it as **ac\_data**.
- Insert a list of **name** and **vnc** (voltage noise contributors) and click **Plot Options** and **Suppress Table Format**. As shown here, Q1.BJT1 is the total noise voltage for the device. It is composed of two pieces: Q1.BJT1.ibe and Q1.BJT1.ice. This means that the total BJT noise comes from both the base-emitter current (ibe) and collector-emitter current (ice).  
However, these are two uncorrelated noise voltages that have been added as noise powers:  $(V_{\text{total}})^2 = (V_{\text{ibe}})^2 + (V_{\text{ice}})^2$ .  
Also, note that the total vnc is the same as Vout noise. If you have time, insert a separate list of Vout.noise and verify this.



vnc	name
18.02pV	RB
262.9pV	RC
1.184nV	_total
freq=900.0MHz	
826.1pV	Q1.BJT1
10.44pV	Q1.BJT1.ibe
826.0pV	Q1.BJT1.ice
805.4pV	R1
18.02pV	RB
	RC
	_total
	Q1.BJT1

Table Format  
Table format is available and used by default for lists with 2 independent and 1 dependent trace only.  
☒ Suppress Table Format

- Save the data display.

#### 5. Write a Measurement Equation to calculate gain

- Insert a **MeasEqn** from the AC simulation palette. Or, you can type in **MeasEqn** in the component history list.

```
MeasEqn
meas1
your_measurement_equation_here
```

- Now, edit the equation so it looks like the one shown. It computes the gain in dB using voltages at name nodes Vin and Vout:

```
MeasEqn
meas2
Gain_dB=20*log(mag(Vout)/mag(Vin))
```

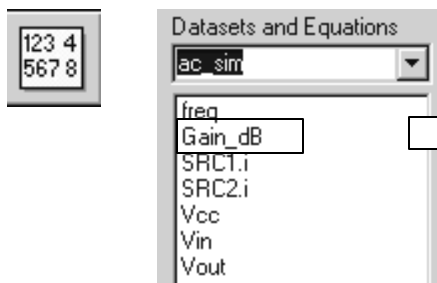
## 6. Simulate without noise and display the results

- a. In the schematic, **turn off the noise** calculation by editing the simulation controller setting on-screen. Turning off the noise calculation will save simulation time and data, especially for large circuits. Of course, this will make the list you inserted (name and vnc) become invalid.

AC

AC  
AC1  
Start=100 MHz  
Stop=2 GHz  
Step=100 MHz  
CalcNoise=no  
NoiseNode[1]="Vout"

- b. Save the schematic. **Simulate** again. When the simulation is finished, insert the **Gain\_db** equation in a list.



freq	Gain_db
100.0MHz	5.572
200.0MHz	5.572
300.0MHz	5.571
400.0MHz	5.571
500.0MHz	5.570
600.0MHz	5.569
700.0MHz	5.567
800.0MHz	5.566
900.0MHz	5.564
1.000GHz	5.563

- c. Now, in the data display, **insert an equation** to calculate the same gain. However, this time give it a different name, such as **dB\_Gain**:

$$\text{Eqn dB\_Gain} = 20 * \log(\text{mag}(\text{Vout}) / \text{mag}(\text{Vin}))$$

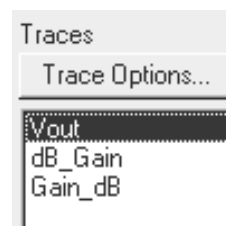
- d. **Edit the list** and add the data display equation: **dB\_Gain**. Now you have two results (they are the same) from two equations – one written before simulation and one after simulation.

freq	Gain_db	dB_Gain
100.0MHz	5.572	5.572
200.0MHz	5.572	5.572
300.0MHz	5.571	5.571
400.0MHz	5.571	5.571
500.0MHz	5.570	5.570
600.0MHz	5.569	5.569
700.0MHz	5.567	5.567
800.0MHz	5.566	5.566
900.0MHz	5.564	5.564
1.000GHz	5.563	5.563

Equation from schematic

Equation from data display

- e. **Edit the list** one more time and add **Vout**. With the Vout data selected, click the **Trace Options** button.
- f. In the **Trace Expression** field, change Vout to read: **dB(Vout)** as shown and click **OK**. You are using the built-in dB function on the Vout data. Because the AC signal at Vin is 1 volt, the dB value of Vout will have the same value as the dB gain equations you wrote.

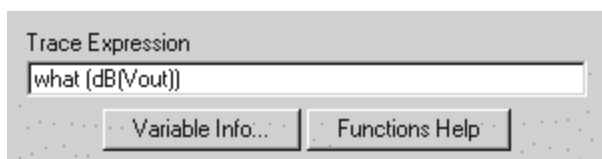


freq	dB_Gain	Gain_dB	dB(Vout)
100.0MHz	5.572	5.572	5.572
200.0MHz	5.572	5.572	5.572
300.0MHz	5.571	5.571	5.571
400.0MHz	5.571	5.571	5.571

**NOTE on equations:** The point of these last steps was to show the similarity and difference between equations you write in schematic and those you write in the data display. In addition, you should remember that variable equations in schematic (VarEqn) are primarily used to initialize (declare) variables sweeping, scaling, etc.

## 7. Use the *what* function on the Vout data

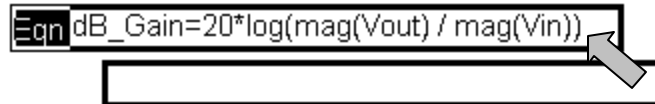
- a. Insert a new **list** (dataset is still **ac\_sim**). Add the **Vout** data again, select it, and click on the **Trace Options** button.
- b. When the dialog box appears, insert the cursor in front of the trace expression and type the **what** function in front of the dB of Vout as shown here, using parentheses on each side. Click **OK** and you get the similar information as clicking *Variable Info* but you get it for the explicit expression: dB(Vout). Of course, the dependency is the same for dB(Vout) and Vout: freq. Try clicking **Variable Info** and see. Later on, you will use this function to determine how to index into dataset values, especially S-parameters and harmonic balance simulations where there is mixing.



what (dB(Vout))	
Dependency :	[freq]
Num. Points :	[20 ]
Matrix Size :	scalar
Type :	Real

## 8. Copy the data display equation using Ctrl C Ctrl V

- Select the **dB\_Gain equation** and then press: **Ctrl C** and **Ctrl V**. Move the cursor and click nearby. The highlighted copy of the equation will appear with "1" appended to the equation name (dB\_Gain1).



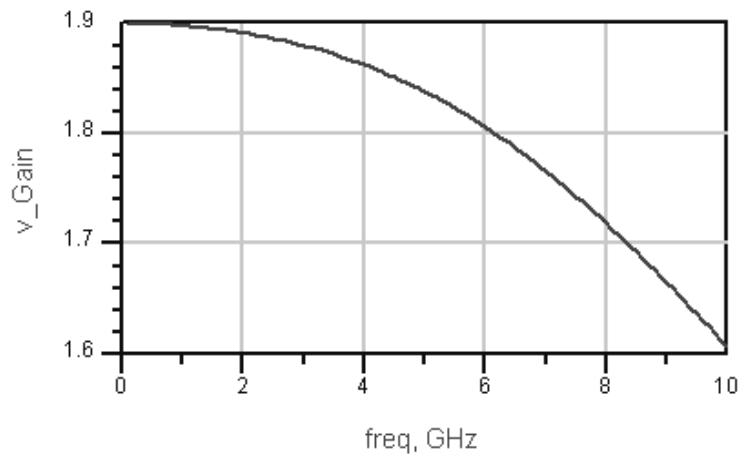
Eqn dB\_Gain=20\*log(mag(Vout) / mag(Vin))

- Edit the copied equation to become a voltage gain equation:



v\_Gain= (mag(Vout) / mag(Vin))

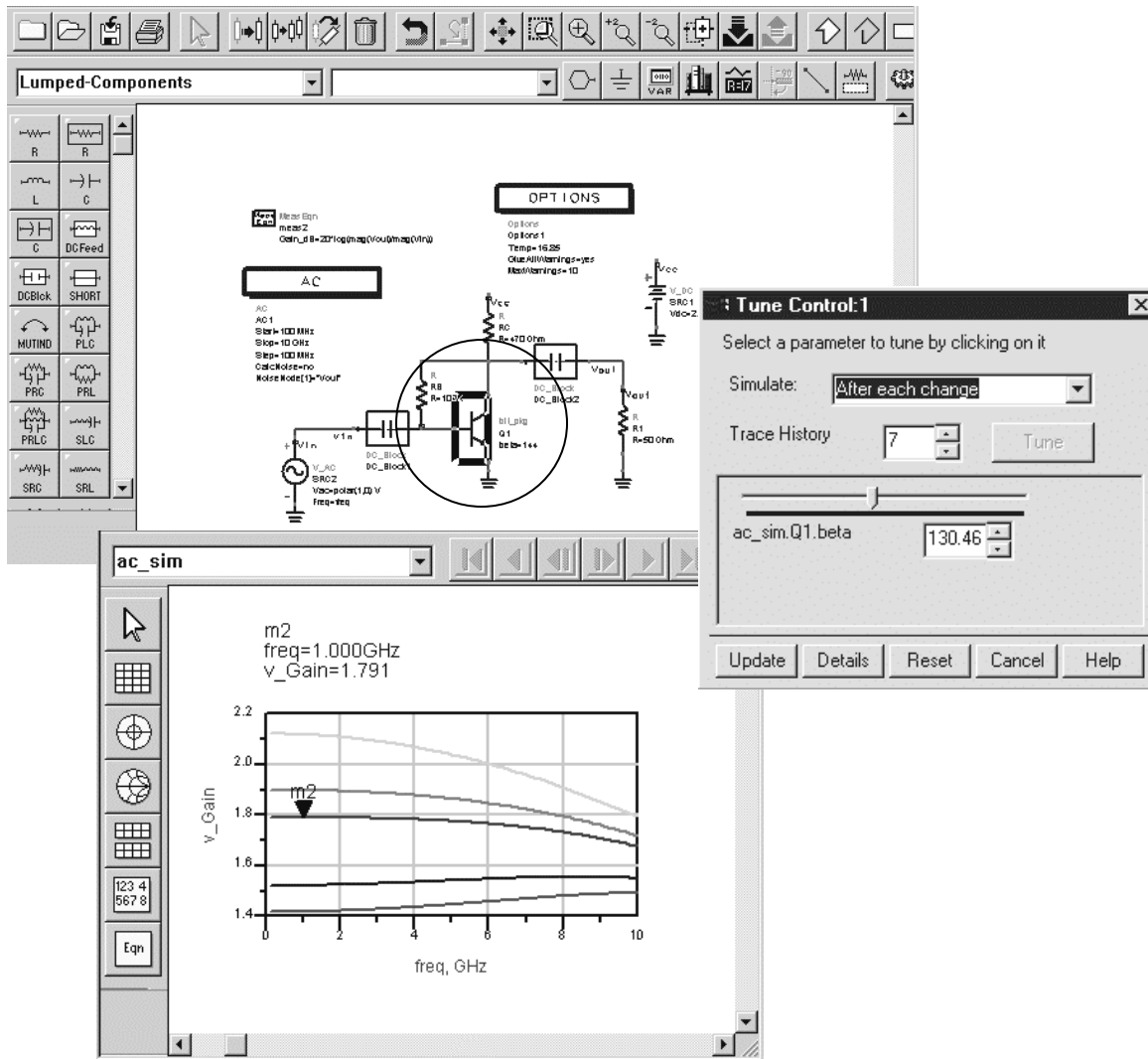
- Return to the schematic, change the simulation stop frequency to **10 GHz**, and simulate.
- In the data display, insert a plot and add the **v\_Gain** equation. You should see a plot similar to the one here showing the voltage gain.





## 9. Tune the beta parameter:

- Position the schematic window and the data display so you can see them both. The select the bjt\_pkg and start the tune mode (Simulate > Tuning). Put a marker on the trace – as you tune the parameter, the marker will move to the most recent trace.

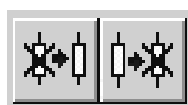


- Try clicking **Update** to see the updated value of beta on the schematic. Note that the Reset button only resets the initial value on the Tune Control dialog. Be sure that beta is 144 when you Cancel the tuning or simply edit beta on the schematic to be 144.

## 10. Use another source for the analysis

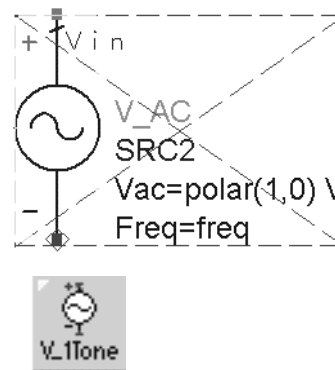
This step shows how sources are related to simulation controllers. By substituting a different source in the design, you will see the relationship. The V\_AC, I\_AC and P\_AC sources are specifically designed for use with the AC Simulation controller. However, almost any frequency domain source can be used for an AC simulation if it has the Vac, Iac or Pac variable.

- In the circuit schematic, select the V\_AC source and move it (**Edit > Move > Move and Disconnect**) to the side of the schematic and deactivate it.

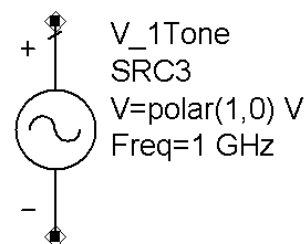
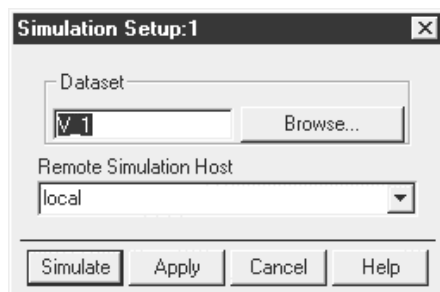


Activate and Deactivate icons

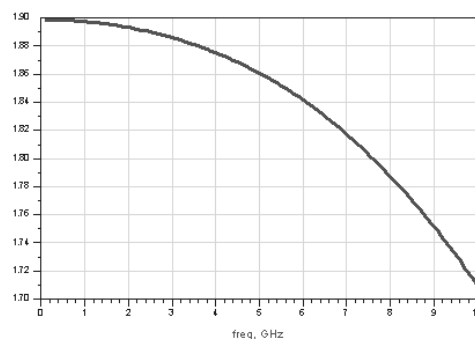
- Insert a **V\_1Tone** source (Frequency Domain sources). This source is designed to be used with the harmonic balance simulator but can be used here also. Note the difference in the default for **freq** (= freq or 1 GHz).



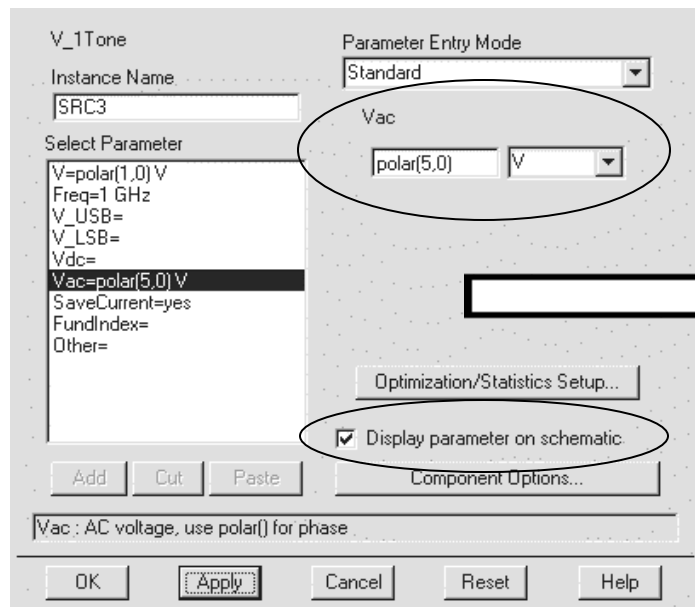
- Simulate** with the **dataset name = V\_1**.



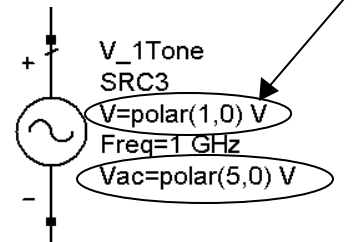
- In the data display, insert a plot of the magnitude of **Vout**.
- Go back and change the voltage from 1 V to 100 volts: **V=polar (100,0) V**. Now, set the dataset name as: **V\_100**. **Simulate** and add the trace to the plot. You will see the exact same value. The next step will explain this...



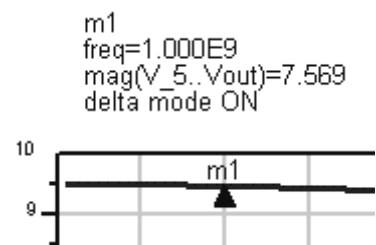
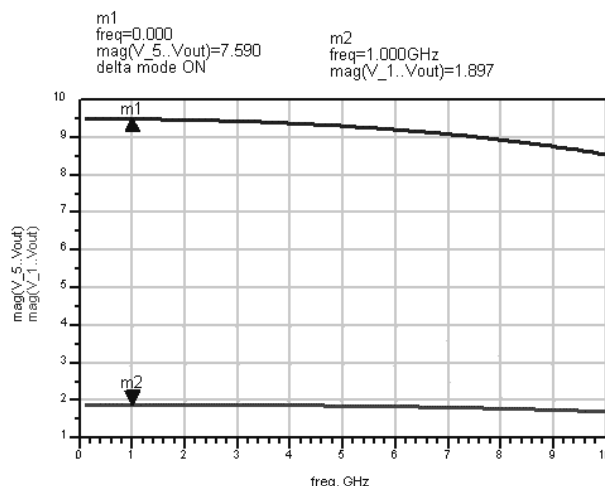
- f. The AC simulation controller only reads the Vac setting. Because the source can be used with different simulation controllers, the setting of V is necessary. Therefore, edit the source so that the **Vac=5** volts and that the **Vac setting is displayed**.



V is not used by the AC simulator. But Vac is used.



- g. Simulate again with the same **dataset name: V\_5**. When the simulation is finished, edit the plot and add the magnitude of V\_5 to the plot and you should now see the two traces.
- h. **Delta Marker Mode**: Insert a marker on each trace at 1 GHz. Then select the two markers (shift click) and then click: **Marker > Delta Mode On** and choose M2 as the reference. The text will show the difference between the two markers on the Y axis. Move m1 to 2 GHz and see the change in the displayed marker text.



## 11. Sweep Vcc (as if the battery voltage dropped below 2 volts)

This step will require you to use the skills you already learned in this lab and in lab 2. You will set up a parameter sweep for **Vdc** from **1.8** to **2** volts in **0.05** volt steps.

- Replace the V\_1Tone source with a **V\_AC** source and deactivate the V\_1Tone – use the command **Edit > Move > Move and Disconnect and Edit > Component > Deactivate**.

- Insert a **VAR** (variable equation) initializing **Vbias = 2 volts**.

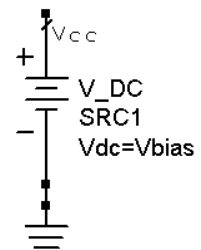


VAR  
Eqn  
VAR1  
Vbias=2.0 V

- Redefine Vcc: **Vdc = Vbias**.

- Insert a **Parameter Sweep**. Then set the **SweepVar** (sweep variable) to be **Vbias**, and be sure the Simulation Instance Name of the AC simulation controller is also set.

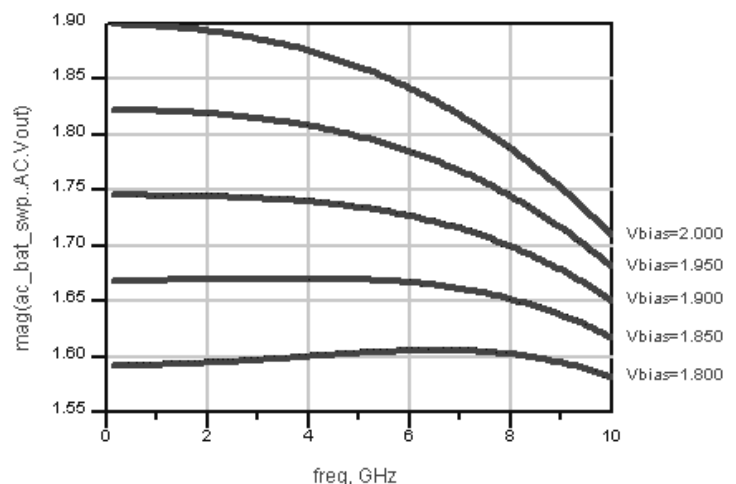
- Simulate** as **ac\_bat\_swp** (dataset name) and then display the magnitude of the **Vout** data as shown.



### PARAMETER SWEEP

```
ParamSweep
Sweep1
SweepVar="Vbias"
SimInstanceName[1]="AC1"
SimInstanceName[2]=
SimInstanceName[3]=
SimInstanceName[4]=
SimInstanceName[5]=
SimInstanceName[6]=
Start=1.8
Stop=2
Step=0.05
```

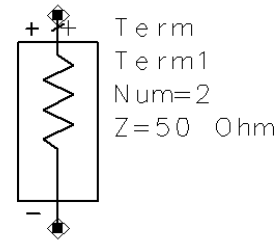
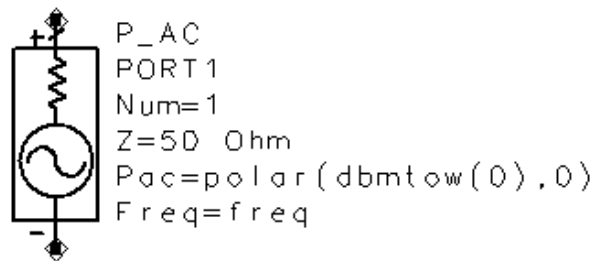
Simulated battery drain over broad frequency range. Also, Trace Options used to thicken the trace lines.



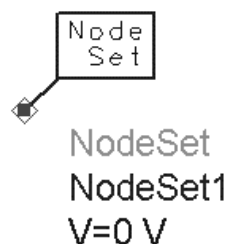
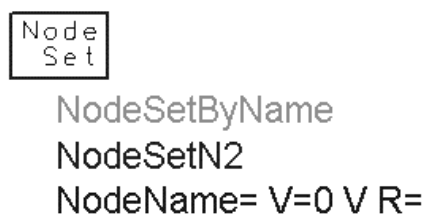
## 12. Save all your work and close the windows

**EXTRA EXERCISES:**

1. Simulate with port noise and ports. To do this, use a P\_AC source as the input port (Num=1) and place a Term on the output as port 2 (Num=2). These two components are shown here with the port numbers.



2. Insert the I\_AC constant current source and simulate. To do this, you need to put a large resistance in parallel with the source because the simulator needs to verify a dc path to ground and the current sources are open circuits.
3. Insert the P\_AC source and look at the power gain. Also, sweep another parameter and plot the results.
4. Try using the node settings in the AC simulation palette. You can set initial voltages at nodes using the Node Set or by referring to name nodes using the NodeSetByName component.



THIS PAGE LEFT INTENTIONALLY BLANK