

TINA PRO ADVANCED TOPICS

Introduction

TINA offers many useful and advanced features for designing, testing, and teaching electronics. These include S-parameter models, network analysis, Fourier series and Fourier spectrum analysis, noise analysis, symbolic analysis, post-processing of analysis results, phasor diagrams, Nyquist diagrams, a built-in interpreter, multi-parameter optimization, and much more.

Detailed descriptions of all these features can be found in this electronic manual.

Table of Contents

- Parameter Stepping
- DC Transfer Characteristic and Parameter Sweeping
- Phasor Diagram
- Nyquist Diagram
- Noise Analysis
- Network Analysis and S-parameters
- Symbolic Analysis
- Post-processing Analysis Results
- Optimization
- Fourier Series and Fourier Spectrum


We are continuously adding new topics and examples to this chapter, so please check back regularly to the Documentation section of our web site www.tina.com for the latest version.

Parameter Stepping

In Parameter stepping mode, TINA steps the associated component values a given number of times through a given range and runs the selected analysis for each value. This results in the calculation and plotting of a set of curves illustrating the sensitivity of the circuit to changes in the associated parameters.

Parameter Stepping is controlled by four parameters: Start value, End value, Number of cases, and Sweep type (Linear, Logarithmic or List).

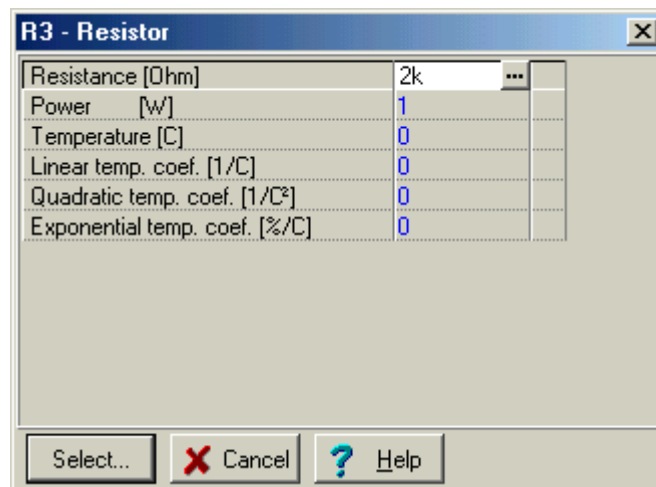
The stepped component value parameters can be any numerical parameters of the TINA components—e.g., the resistance of resistors, the setting of a potentiometer, or parameters on transistor data sheets.

To set up parameter stepping for a component, click on the Analysis>Control Object menu item, or click on the  Select Control Object speed button.

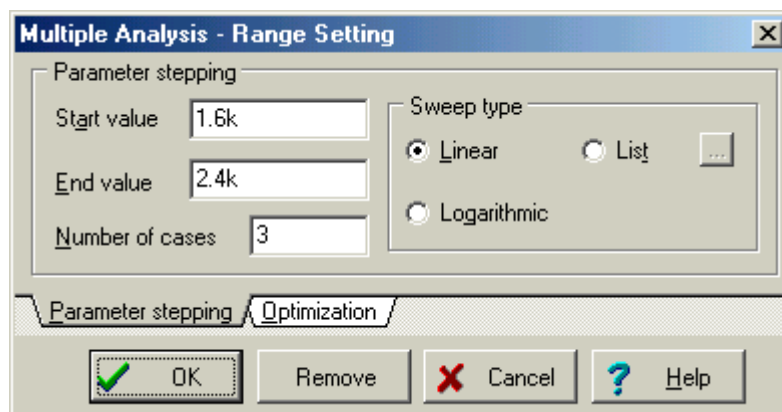
A cursor that looks like a resistor will appear in the schematic editor window, waiting for you to identify the component for which you wish to assign a stepped parameter. Point at the desired component, right click, and all of its associated parameters will be displayed in the dialog window.

Should the desired parameter belong to a catalog component or be an excitation attribute, you must first select the semiconductor or waveform type so that you can access the appropriate data sheet.

In the case of a resistor, for example, the following dialog window will appear:



Select the line containing the parameter to be stepped and then click on the Select button. The following dialog will appear:



Set the appropriate stepping parameters. The number of cases includes the Start and End values.

By setting the number of steps to 1, you can perform a single analysis at the value defined by the Start value of the parameter. This gives you a way of examining the response with only a temporary change of parameter (without modifying the original value).

If you press the List button and then press the unlabeled button to the right, the List Editor for parameter stepping will appear:



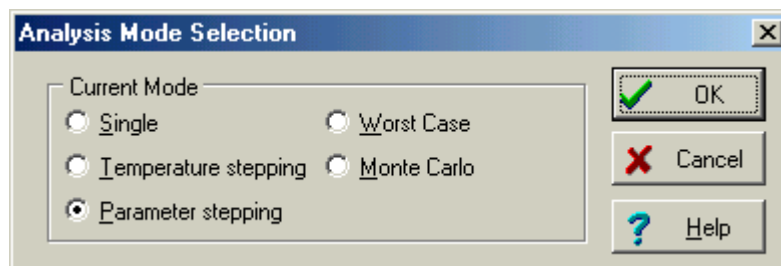
By default it will show three values calculated between the Start and End value, but you can enter any values you wish, remove values, or clear the list entirely.

Press OK to return to the previous dialog window which, in turn can be closed by its OK button. You can also turn off parameter stepping for this component using the Remove button.

You can repeat the above procedure for any number of parameters to be stepped.


Note that you do not have to use the same number of cases for each value being stepped. TINA will run an analysis for all possible combinations of values stepped. For example, if you ask to vary part A in three steps and part B in four steps, you'll get twelve curves.

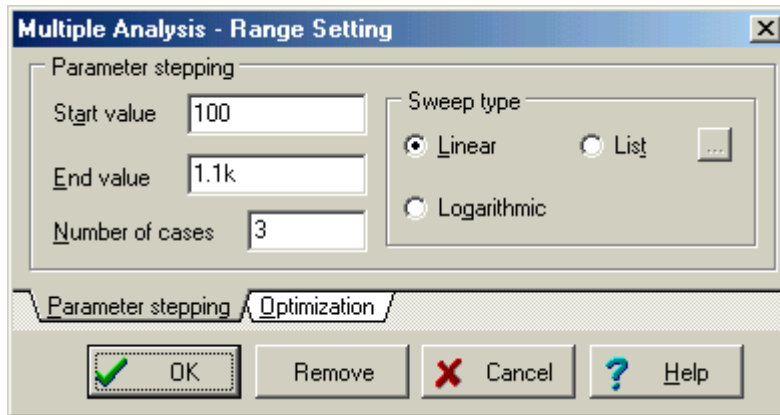
The program will go into parameter stepping mode automatically whenever a component is set as control object according to the procedure above. The mode status can be checked via the Analysis/Mode dialog window.



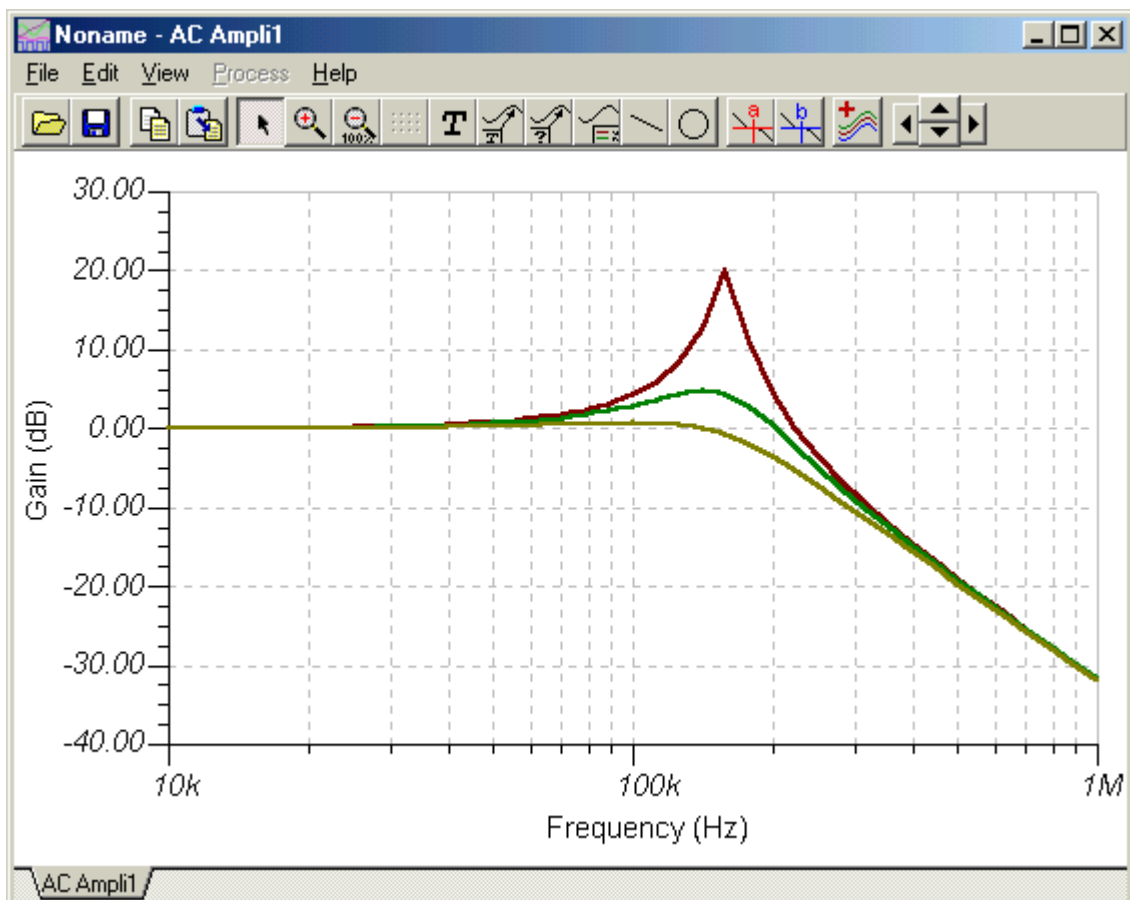
You can use this dialog window to select single analysis or parameter stepping analysis.

Using this dialog window, you can also select Temperature Stepping, Worst Case and Monte Carlo analysis.

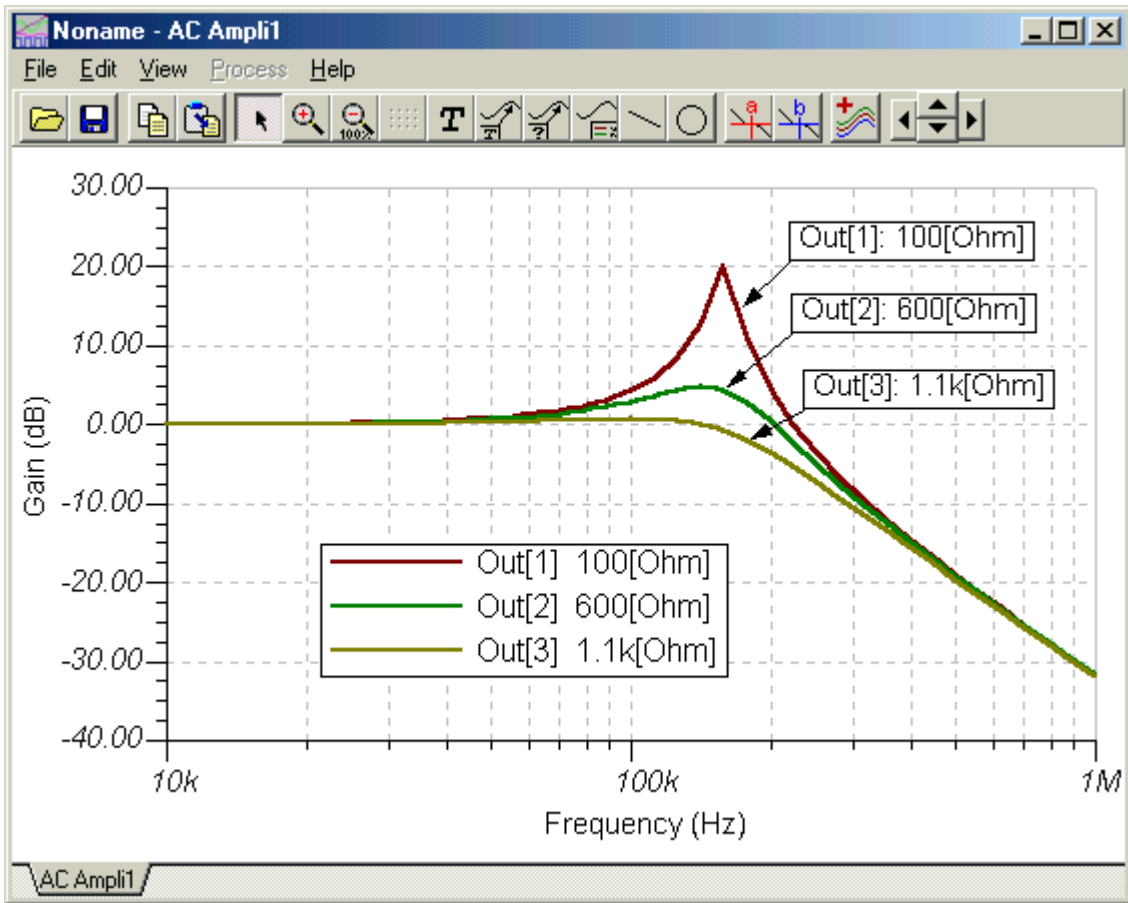
An example will make Parameter Stepping clearer. Load the file RLC_1.TSC from TINA's EXAMPLES folder. Note that this circuit is already in parameter stepping mode for the resistor R, which is changed between 100 Ω and 1.1k Ω in 3 steps. Click the  button, select the R resistor with the new cursor, and press Select. You should see the following dialog window:




After closing this dialog and running Analysis/AC Analysis/AC Transfer Characteristic, the following diagram will appear:

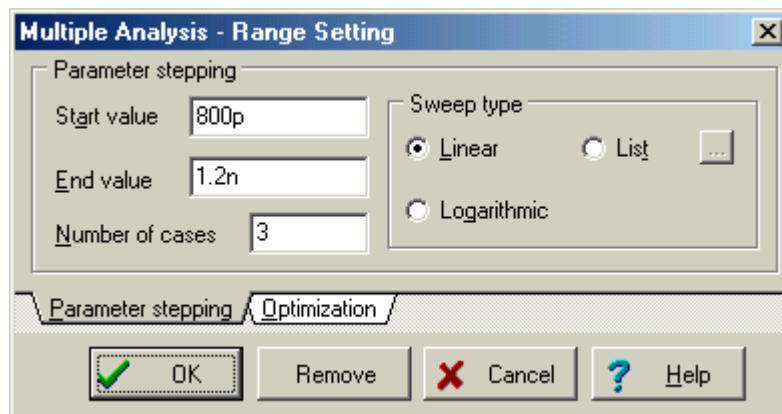


To determine which value of the stepped parameter is associated with a curve, move the cursor over the curve and read the corresponding value in the line at the bottom of the screen. You can also use the Legend and Auto value buttons in the diagram window to show the corresponding stepped values. Refer to the diagram below.



Now let's step the value of the capacitor using the List option. Let the values be 100pF and 1nF.

Click the  button, select the capacitor C, and press Select. You should see the following window:

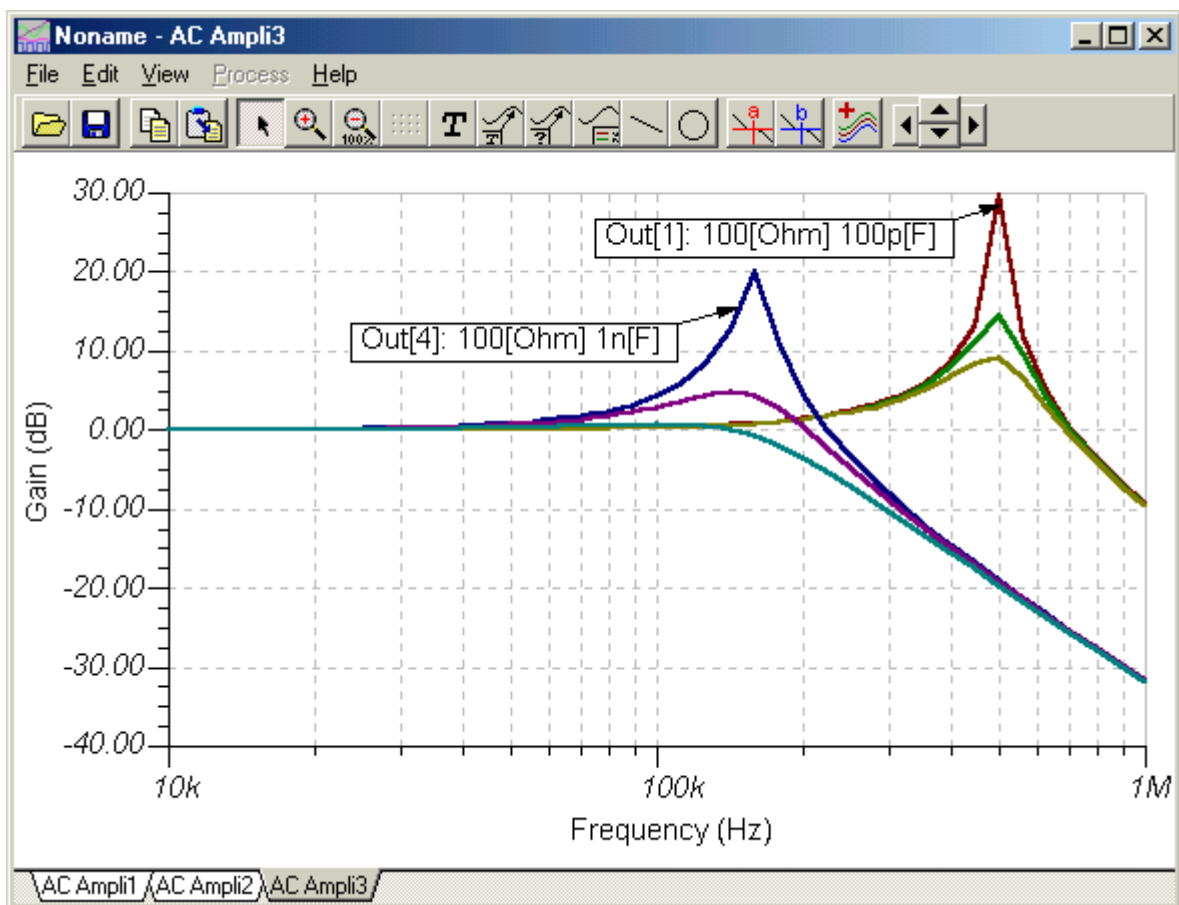


Select the List option and press the button to its right. The following dialog will appear:



Press the Remove Last button and set the first parameter to 100p.

Now close all the dialog windows by pressing their OK buttons and run Analysis/AC Analysis/AC Transfer Characteristic.



The new set of curves belonging to the cases where C=100pF can be clearly recognized on the right side of the diagram. We have also added two auto labels showing the typical values of the parameters.

DC Transfer Characteristic and parameter sweeping

To calculate a DC transfer characteristic, you should assign exactly **one** input (normally a voltage or current source or generator, or a resistor) and at least one output. Later, you can change the assigned input in the DC Transfer Characteristic dialog window. You can also add more outputs using the post-processor. Plotting a result as a function of a range of resistor values is called DC parameter sweeping.

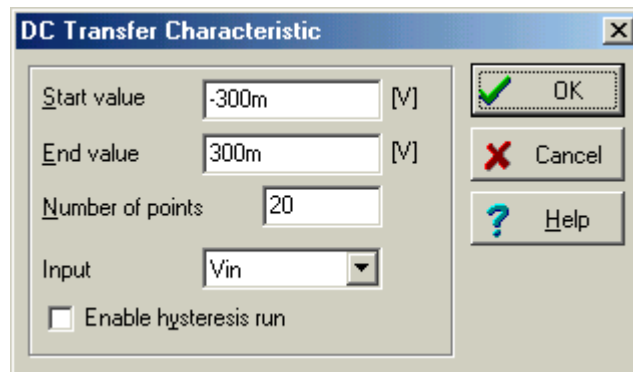
There are several ways to assign a component to serve as the input.

1. In TINA Version 6, the simplest way is to select the part from the Input list box of the DC Transfer Characteristic dialog window. Once selected, TINA will remember it and it will be saved as the selected input in your circuit file.
2. Use the I/O state field of generators and sources to assign them as input.
3. You can also use the Insert/Input command to assign an input to a source/generator or to a resistor. First select the Insert/Input command and see a special I^+ shaped cursor appear. Click the first node of the component to be selected and hold down the mouse button while you drag the input to the other node. The Insert/Input command is not the easiest way to select a DC Transfer input: method one or two is preferred.

To assign an output, use any appropriate meter from the Meters Toolbar (Voltage Pin, Voltage Meter, Ampere Meter, Power Meter, etc.) or use the Insert/Output command. In most cases, a meter is the best choice; however, if your output is on nodes far from each other, the Output command might be useful.

After you have assigned at least one output, you can invoke the Analysis/DC Analysis/DC Transfer Characteristic command.

The following dialog will appear:



Explanation of the parameters:

Number of points: The program calculates the transfer characteristic at the number of points defined. The default value is 20.

Start Value and End Value: The program evenly divides the specified input range so that the number of points includes the start value and the end value. For each input point, the program calculates the output value and uses it to build the transfer characteristic. The units indicated in the square bracket correspond to that of the input (e.g., V for a voltage generator). Normally, the input values are generated in increasing order. However, if you prefer calculation in a decreasing order, assign the greater limit of the input range to the Start value and the smaller limit to the End

value. The order of calculation only matters with the DC characteristics of circuits with hysteresis (e.g., Schmitt trigger).

Input: Here you can check and change, if desired, the input. To change the input, click the arrow in this field and the available input quantities (label names of sources and resistors in the circuit) will appear. If a component has no label, TINA will automatically label it in the form V_no_label_1. Click on the name to select it as the input.

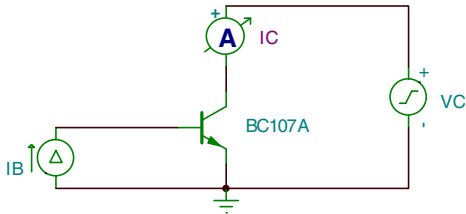
Enable hysteresis run: If this checkbox is checked, TINA will first run an analysis varying the Input quantity from the **Start value** to the **End value**, and then run an analysis from the **End value** to the **Start value**. You should use this option for components with hysteresis (where the characteristic depends on the direction of change).

Note that if you use a resistor as an input, TINA calculates the transfer characteristic multiple times, once for each step of the resistance (parameter sweeping).

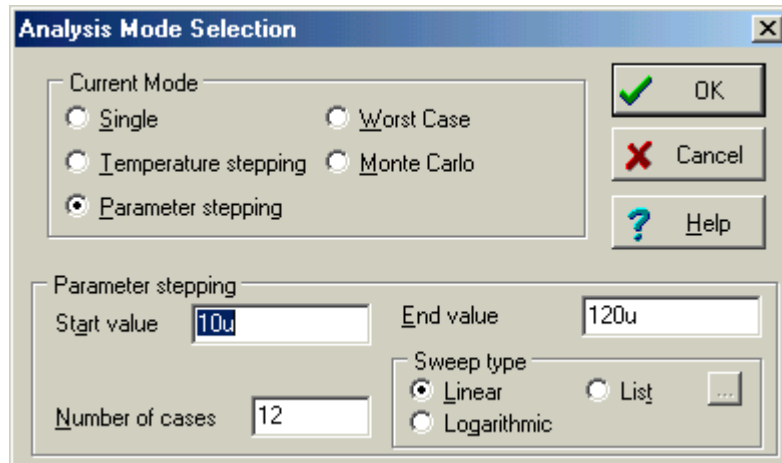
When running an analysis on a nonlinear network, the DC analysis might not converge at certain points. These points will be skipped when plotting the response. If they are at the start and/or end values, the displayed (input) range will be truncated accordingly.

Let's look at two examples.

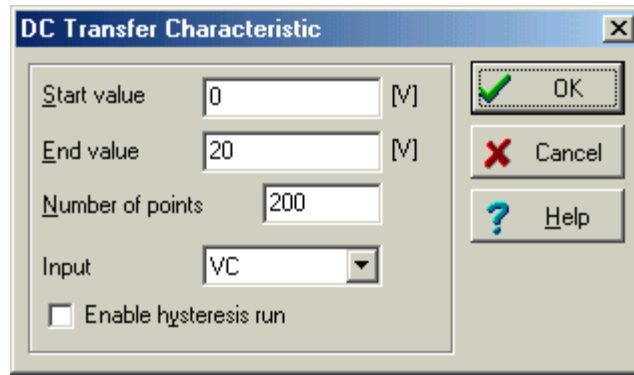
First, let's draw the characteristics of a bipolar transistor (CHARBIPO.TSC)



Select the current source and Analysis/Mode to see how the base current, IB, is stepped.

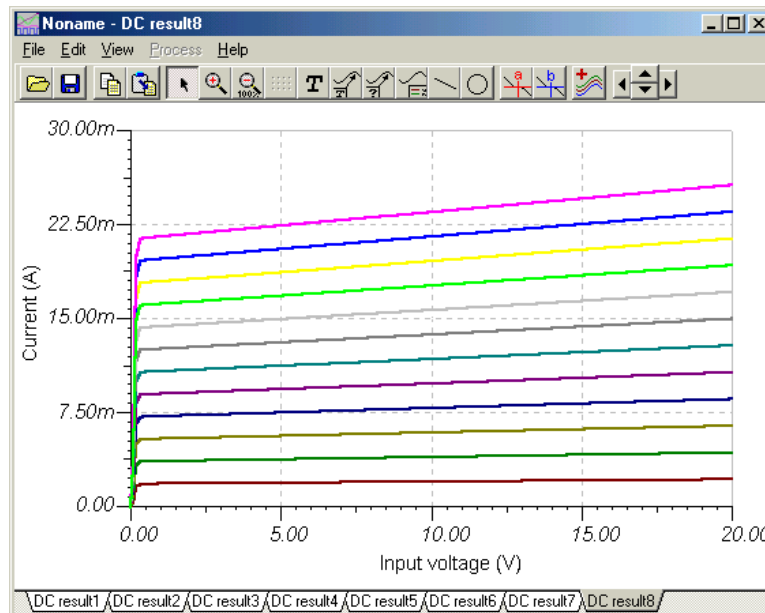


Then invoke the Analysis/DC Analysis/DC Transfer Characteristics command. See the following dialog:



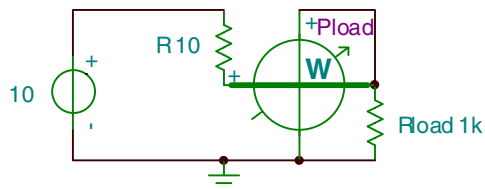
According to the dialog window, the input, VC, will be varied from the Start value (0V) to the End value (20V)

The result is familiar family of curves that constitute the common emitter characteristic of a transistor.



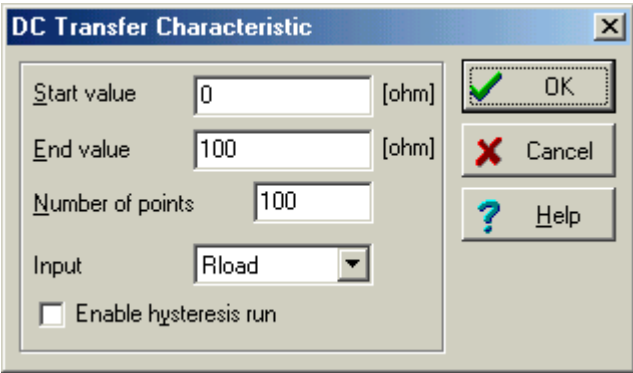
Now let's see an example of parameter sweeping (sweep.tsc)

The circuit:



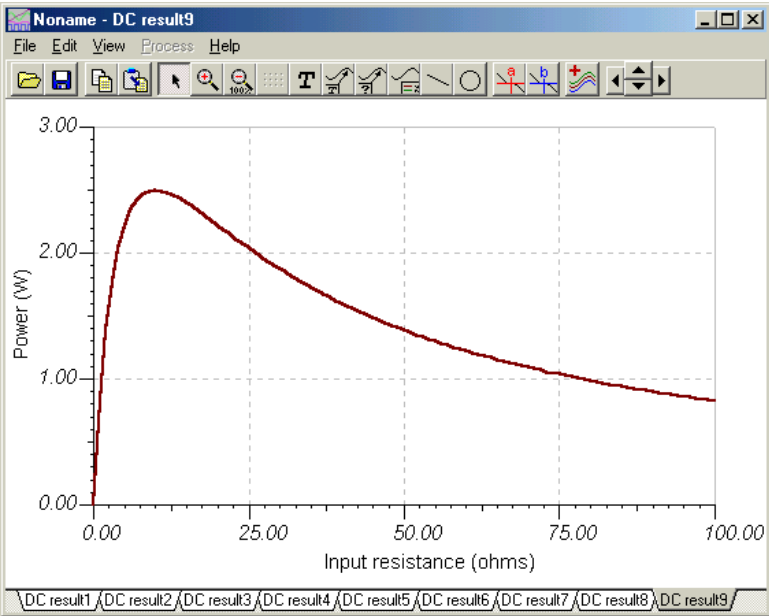
The output is the power meter, showing the dissipation on Rload.

If you invoke the Analysis/DC Analysis/DC Transfer Characteristics command, you'll be greeted by this dialog box:



Note that the resistor Rload is swept from 0 to 100 ohms.

Press OK to obtain the following diagram:



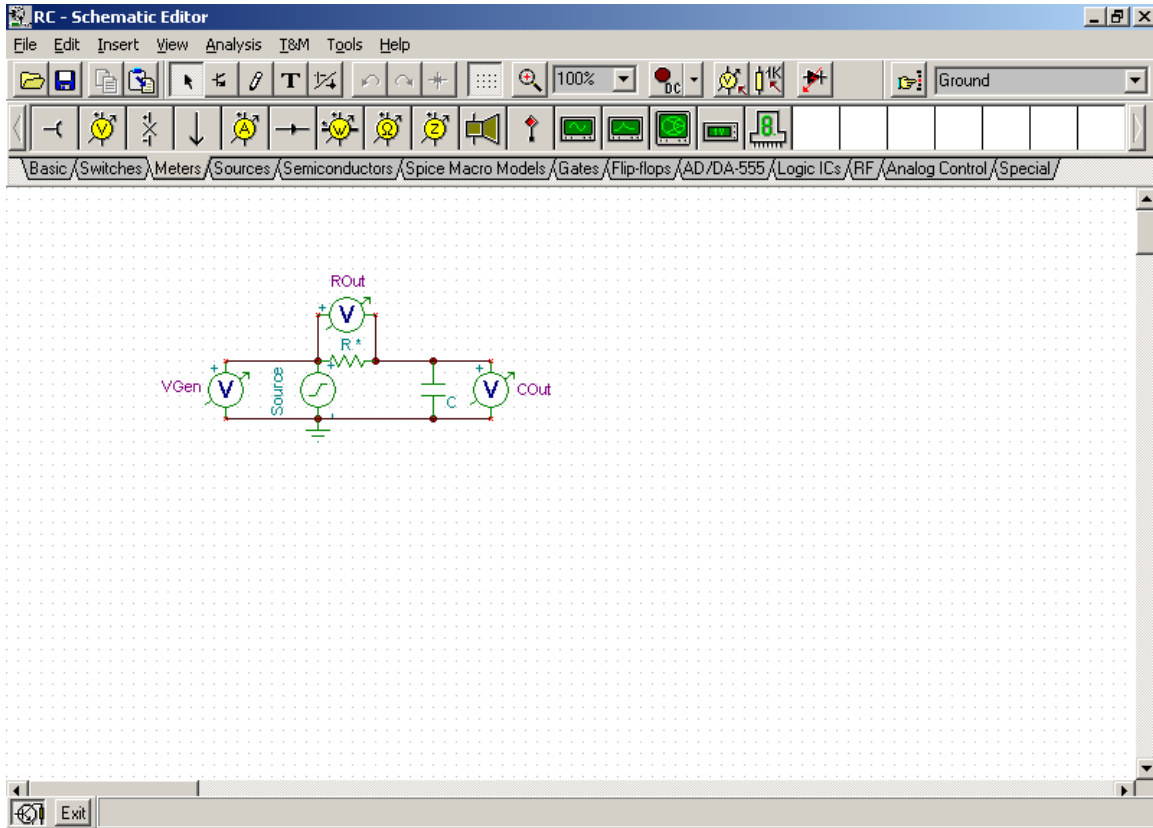
This is the well known curve of power dissipated on a load resistor as the load resistance is varied. Maximum power is transferred when the source resistance and load resistance are equal.

Phasor diagram

The phasor diagram is a great tool for demonstrating and studying circuit operation via complex phasors (vectors). Phasor diagrams can be used only with AC analysis with linear circuits, or with nonlinear circuits in working point linearization.

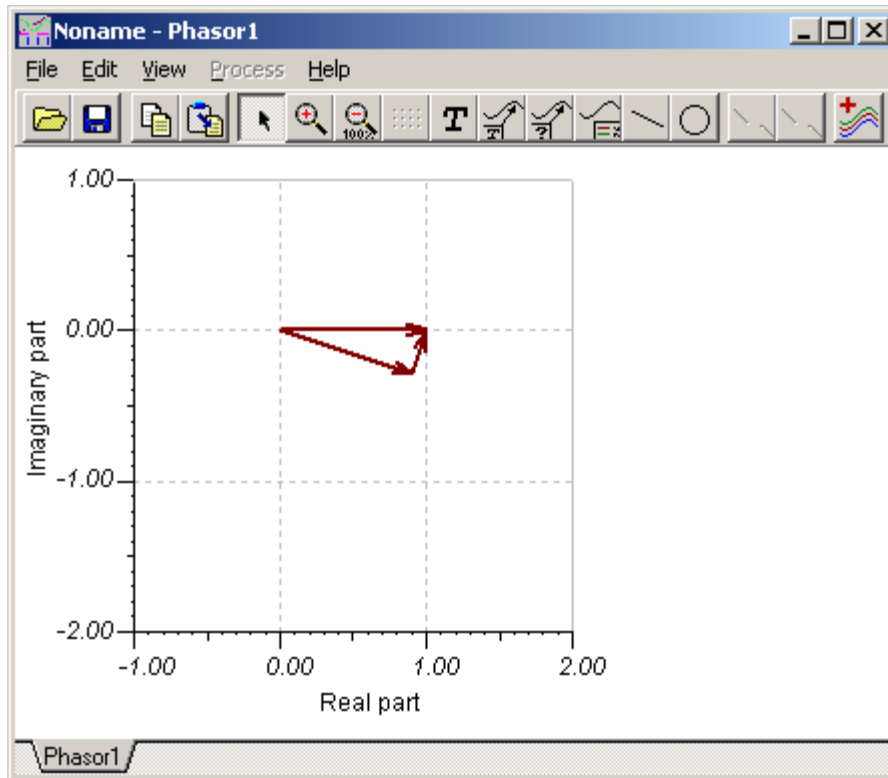
Let's use TINA to draw the phasor diagram of an RC circuit.

First open the circuit RC.tsc in the examples\phasor directory.



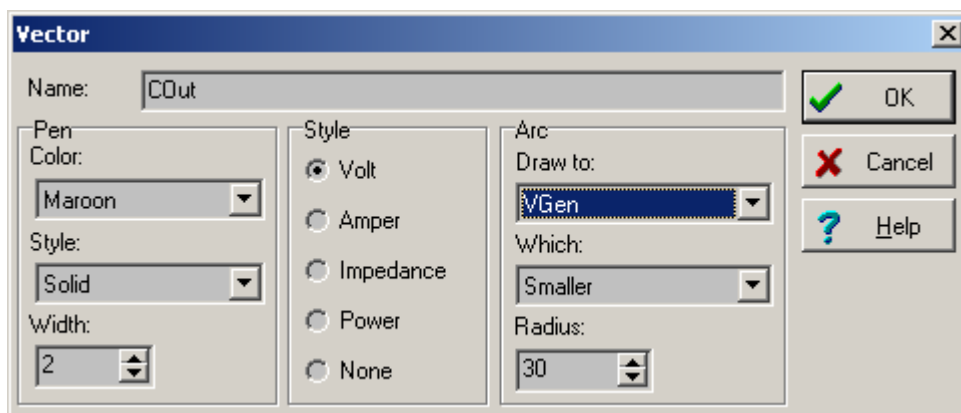
Run the *AC Analysis.Phasor diagram* command. TINA generates a new diagram containing three vectors in the diagram window. As you can see, *Vgen* is the vector sum of *Rout* and *Cout*, but to make this clearer you can change the position of the vectors.

By default all the vectors originate from the origin of the coordinate system; however, you can translate them by simply clicking and dragging them into another place. When you drag a vector close to another vector's starting or ending point, it will be automatically snapped to it. Try dragging the vector *Rout* to the end of *Cout* to form a triangle with *Vgen*. Notice that when you move the cursor above a vector, TINA presents its name, amplitude, and phase in the status-line.



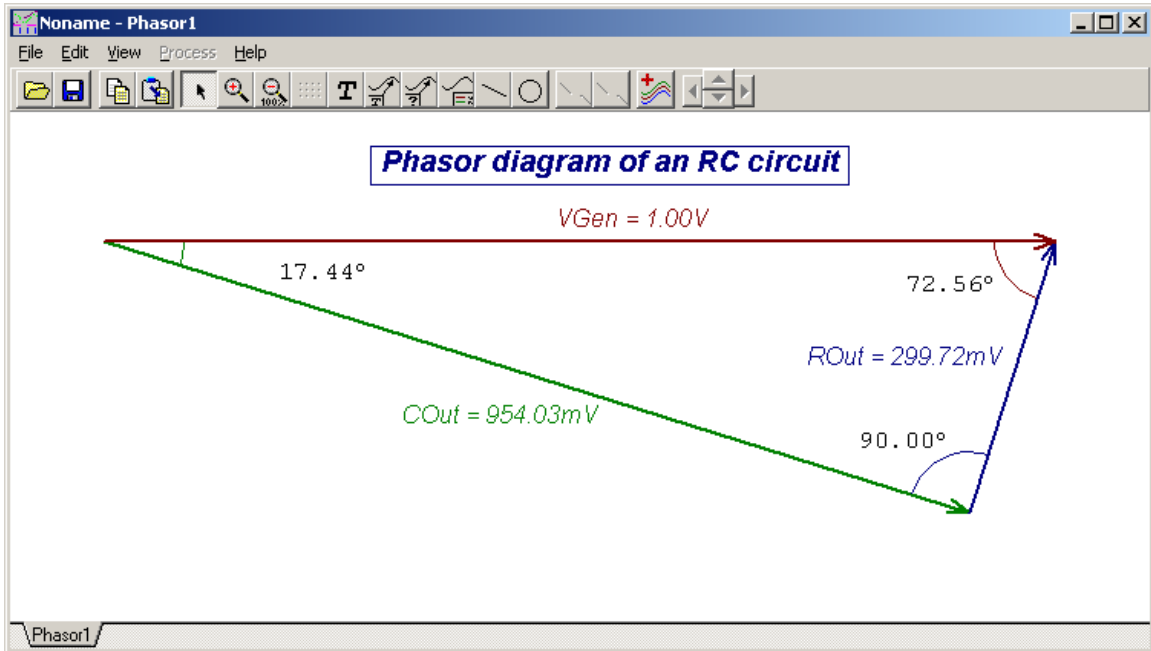
If you prefer a display without axes, turn the axes off using the *View.Display axes* menu.

If you double click on a vector (or right-click it and select Properties), the vector's configuration dialog box appears. In addition to the usual settings for the vector's color, width, and style, you can also set the vector head style.



Using the Arc settings, you can draw an arc between *COut* and *VGen* by setting *VGen* in the *Arc.Draw to* listbox of the *COut* vector's configuration dialog box. You can also set the radius of the arc in pixels and choose whether the smaller ($\leq 180^\circ$) or the bigger arc ($> 180^\circ$) should be displayed.

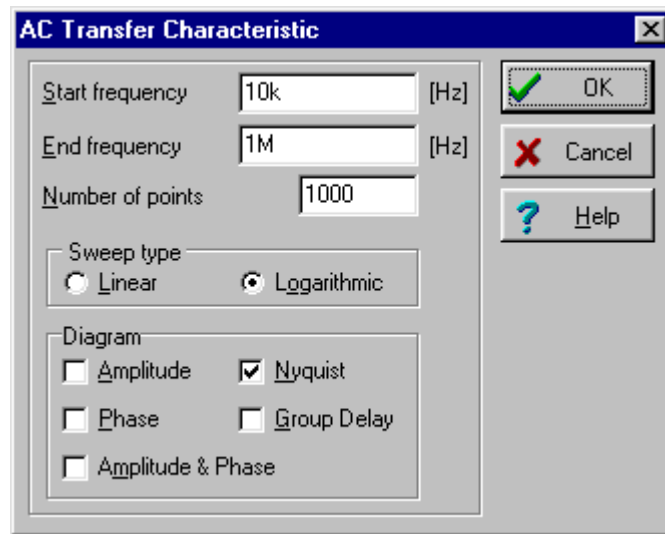
Next, you can add labels to all the vectors at once by pressing the legend button on the toolbar. You can also add legends individually: press the *Auto label* button, then select the vectors one-by-one. The actual label format depends on the selected *Vector label style* under the *View* menu. Finally, you can add lines, circles, or texts to finish your diagram.



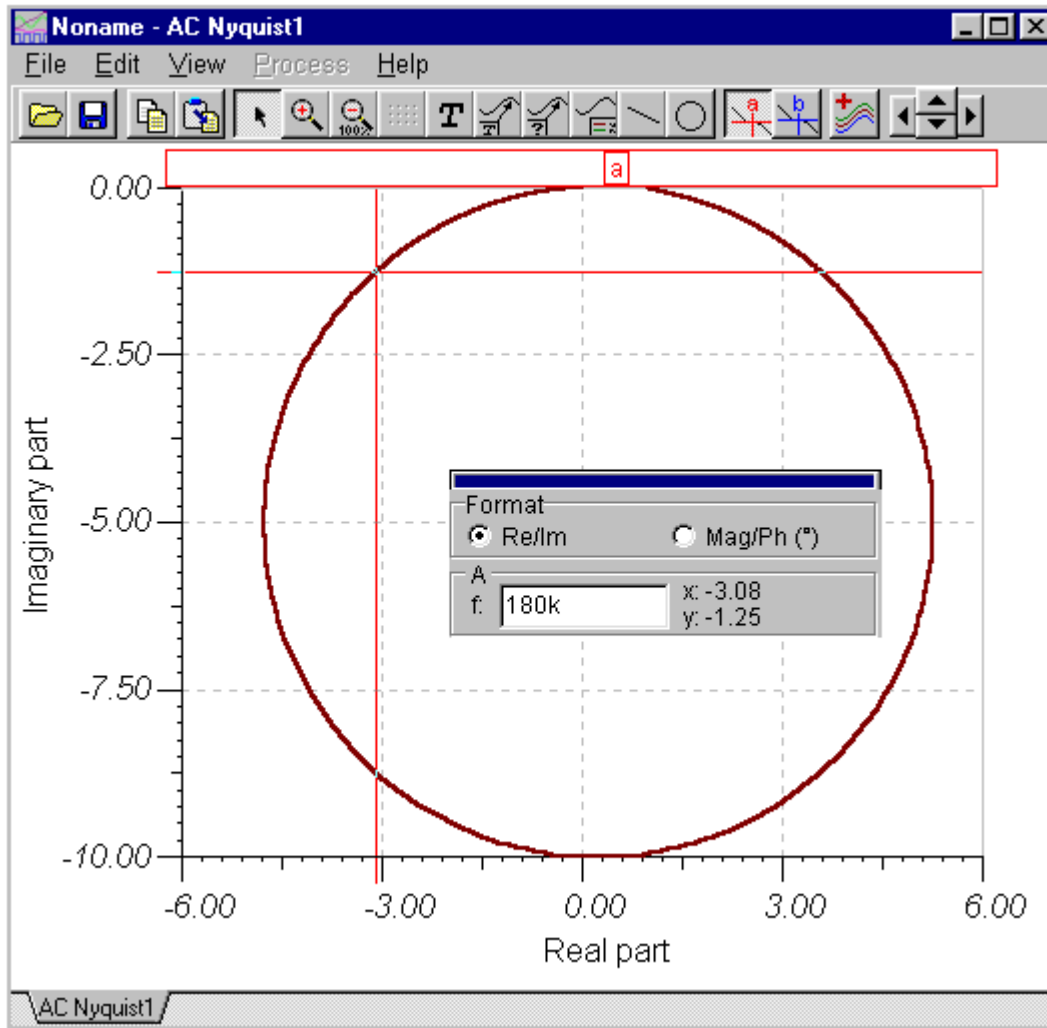
This diagram can then be copied to the clipboard and pasted into the schematic editor or any word processing program.

Nyquist Diagram

TINA's AC transfer analysis can plot not only the Amplitude, Phase and Group delay of the transfer function of analog circuits, but the Nyquist diagram as well. To see how this works, first load the examples\rlc_1.sch circuit. Next, select the Analysis.AC Analysis.AC Transfer Characteristic menu item. Now the AC Transfer Characteristic dialog box appears on screen, where you can select the Frequency ranges of the analysis, the sweep type, and the diagram you want to plot (Amplitude, Phase, Nyquist, Group Delay).



Set the frequency from 10k to 1M and set the diagram type to Nyquist. In order to get a smooth curve, increase the Number of point to 1000. Press the OK button and see the Nyquist diagram of the circuit calculated and plotted, as shown in the diagram window.



This diagram is different from a normal Transient or AC analysis diagram. A Nyquist diagram requires that horizontal and vertical scales always be proportional to each other. This keeps circular curves circular, even if you resize the window. Furthermore, a Nyquist curve presents the trace of a vector quantity as frequency is varied. When you activate a cursor and explore the trace, the cursor dialog shows the numerical values of the curves at a given frequency, in either real-imaginary or amplitude-phase format.

Noise analysis

The minimum signal level which a given circuit can measure or amplify is determined by spontaneous fluctuations occurring in the circuit component. These spontaneous fluctuations are called noise. Noise usually has energy in all parts of the frequency range.

TINA calculates the noise contributions of circuit components and then presents them referred to the output. The random noise fluctuations are analyzed statistically, considering the time domain average to be zero, while the variance and sum of the squares are not zero. Noise analysis results can be displayed and printed over a frequency range similar to the Bode diagram of AC analysis.

TINA takes into account the noise sources inherent in resistors and semiconductors (transistors, diodes). Resistors are modeled as an ideal noiseless resistor in parallel with an equivalent thermal noise current. This equivalent noise is modelled by white noise, i.e., noise containing all frequency components with the same amplitude density. The contributions of all the noise generators in the circuit are calculated to determine each one's magnitude at the output. The 'output noise' is formed from the square root of the sum of the squares (rms value) of these individual contributions.

To correctly account for the noise contribution of operational amplifiers, you should use models made by manufacturers by choosing devices from the Spice Macro Models toolbar. Open the macro and check the noise model. Some manufacturer models have special purpose noise models, or no noise modelling whatsoever. TINA's ideal, standard and linear opamp models (on the Semiconductor toolbar) *do not include* a noise model.

Since the transfer functions of the circuit are known, the equivalent input noise can be calculated from the output noise. Expressing all the circuit noise in terms of an equivalent input noise makes it possible to compare the levels of real signals presented to the input to the circuit noise referred to the input. A circuit expected to amplify small input signals, signals approaching the magnitude of the equivalent input noise, will not be very useful.

Resistors generate resistive (thermal) noise which is computed as the following source current:

$$i^2 = \frac{4 \cdot k \cdot T}{R}$$

where $k = 1.38e-23$ (Wsec/K) Boltzmann constant
T = temperature in K
R = resistance in ohms

Semiconductors generate not only thermal noise (similar to resistors), but flicker noise (1/f noise) as well. The spectral density of flicker noise decreases in inverse proportion to the frequency, hence the name 1/f noise.

TINA calculates the Input and Output noise and expresses the noise as the noise voltage per \sqrt{Hz} of bandwidth.

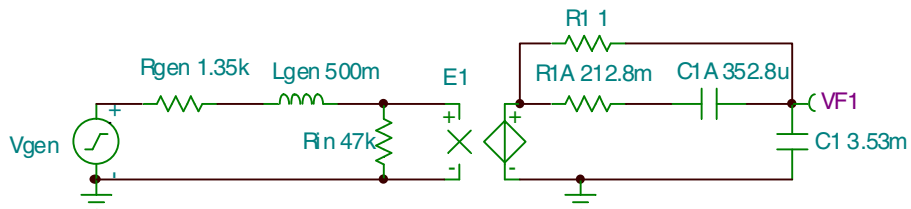
TINA can also present noise as a curve of Total noise, at the output, as a function of frequency. In computing this curve, TINA sums all the noise from the start frequency up to the specified maximum frequency value and presents it as total noise. Naturally, the curve rises monotonically. For example, if you set the start frequency to 20Hz and the end frequency to 20kHz, you will learn the total output noise over the typical audio bandwidth.

Another measure of noise, the "signal to noise ratio," takes the signal magnitude into account. This is the ratio of signal power to noise power, according to the following formula:

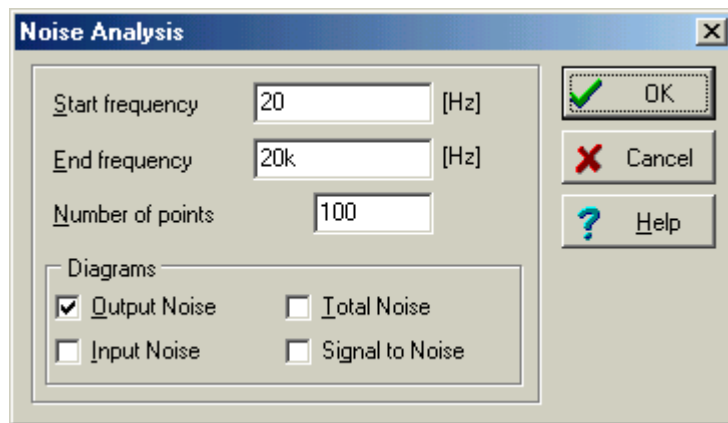
$$S / N = 20 * \log \frac{V_{Sig}}{V_{Tot}}$$

where V_{Tot} is the total input-referred noise voltage and V_{Sig} represents the expected signal amplitude. V_{Sig} is a user-defined value that you enter in the Noise Analysis dialog. The default value is 1 mV.

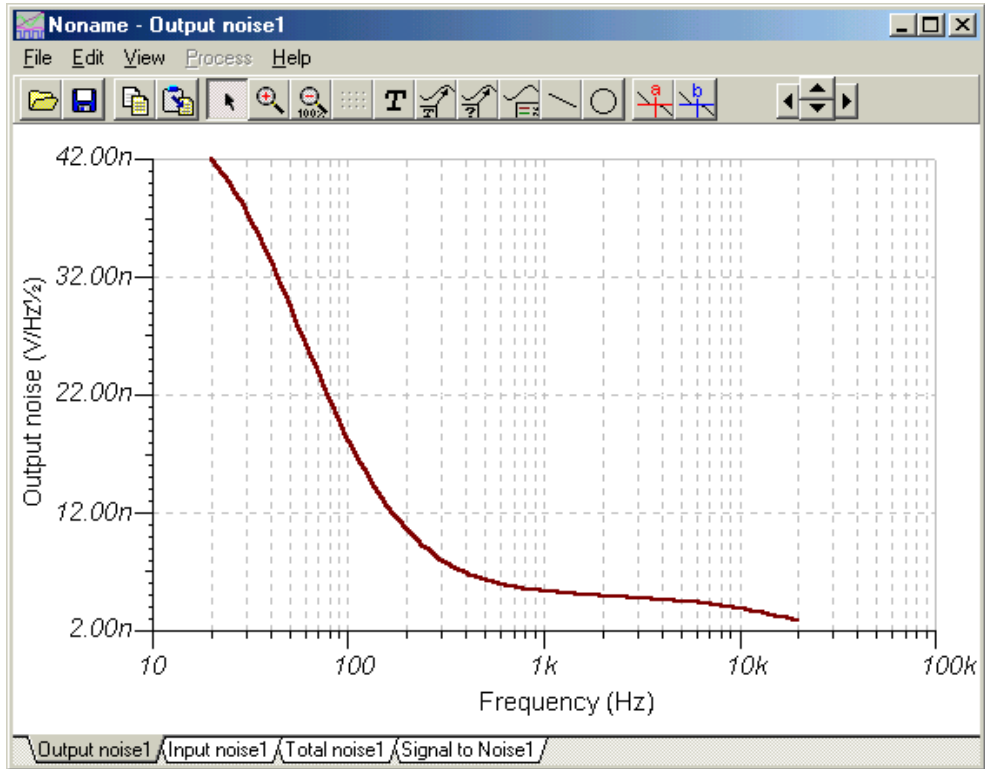
To get a better feeling for these noise concepts, let's study the following circuit, an equivalent circuit of a basic amplifier (noise.tsc).



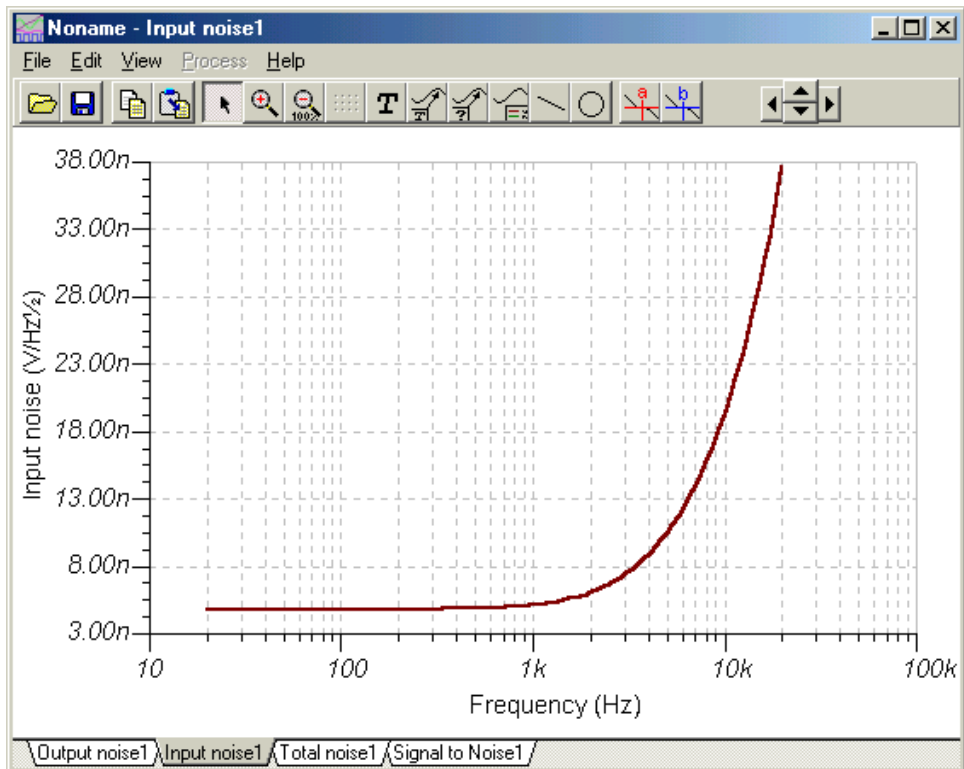
Select Analysis.Noise Analysis. The following dialog appears



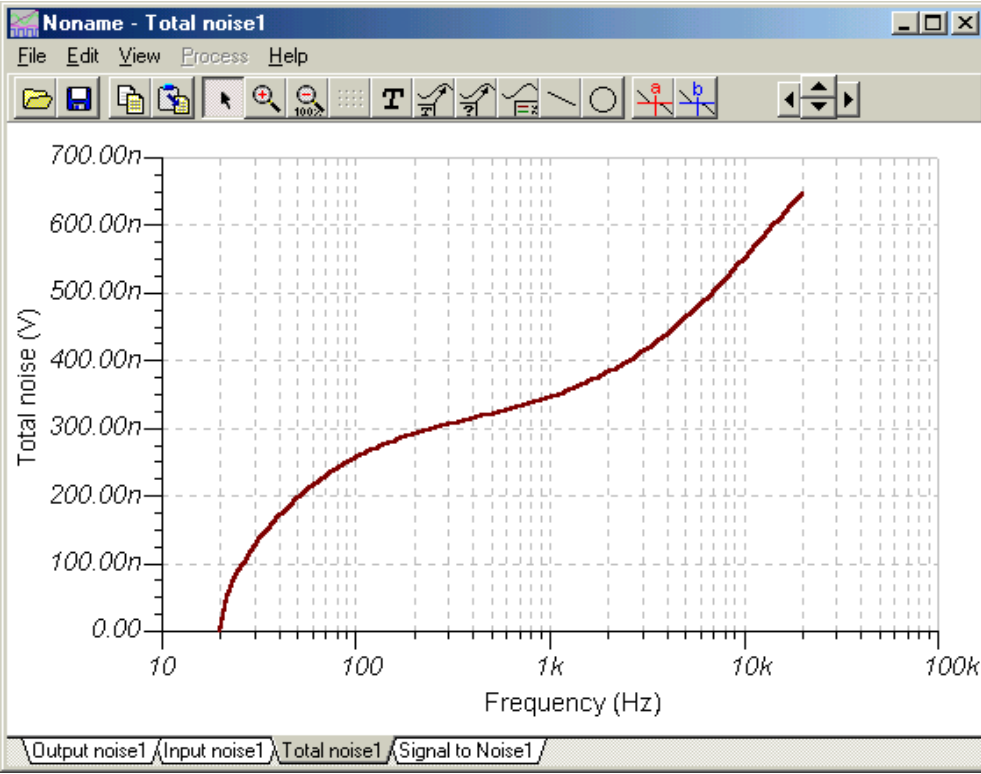
The dialog allows you to set the **Start frequency**, **End frequency**, the **Number of points** for which the noise is to be computed, and the **Signal amplitude**. You may also select the required analysis results—Output Noise, Total Noise, Input Noise, and Signal to Noise.



Output noise



Input Noise



Total Noise

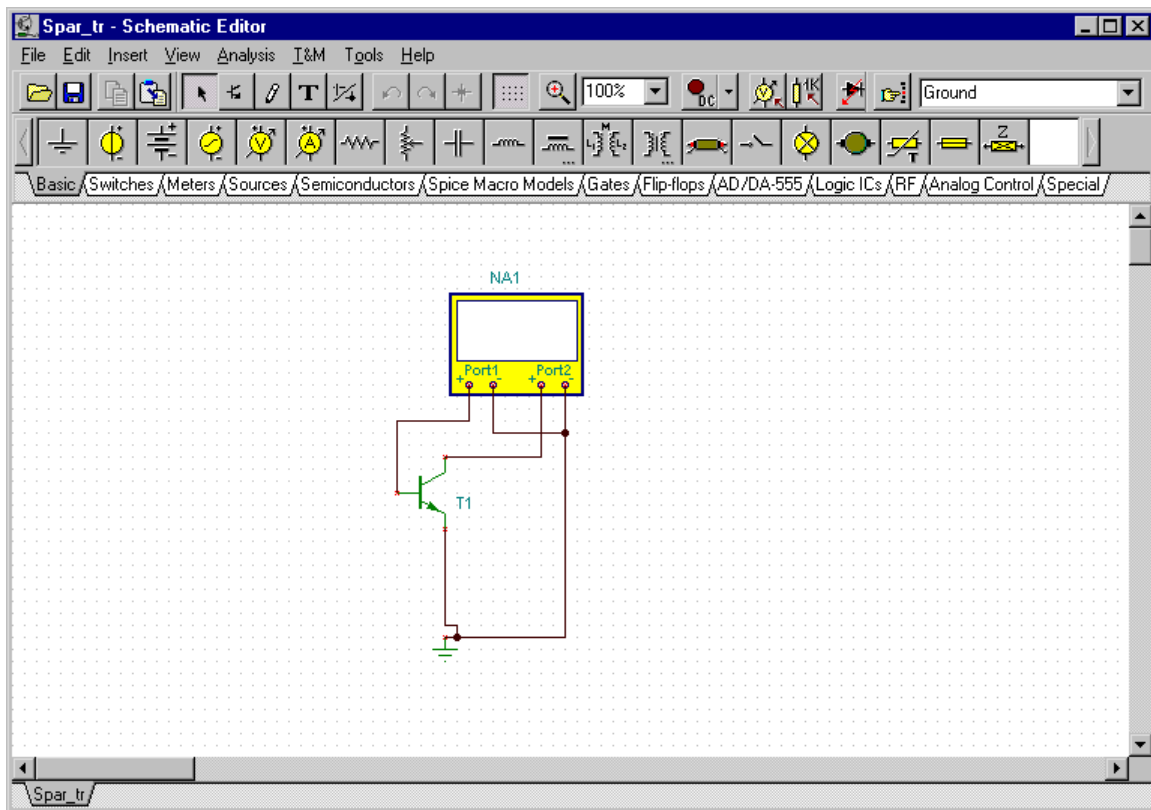
Network Analysis and S-parameters

Network Analysis

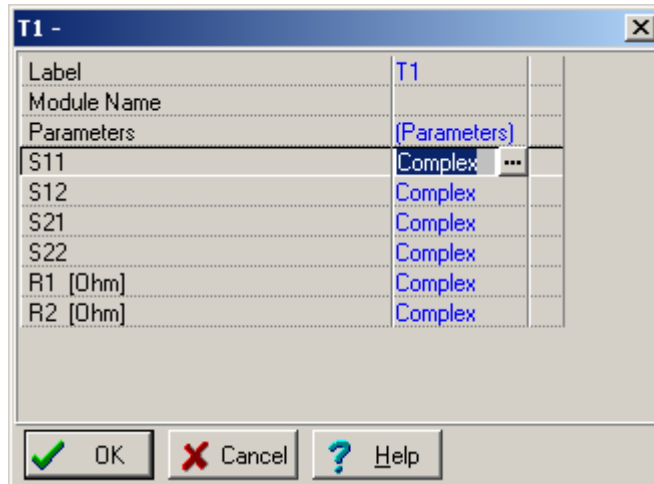
TINA helps you perform network analysis and determine the two-port parameters of networks (S, Z, Y, H). This is especially useful if you work with RF circuits. Results can be displayed in Smith, Polar, or other diagrams.

The network analysis is carried out with the help of TINA's network analyzer. The RF models of the circuit elements can be defined as SPICE sub-circuits (SPICE macros) which contain parasitic components (inductors, capacitors) or as an S-parameter model defined by its S (frequency) function. S functions are normally provided by the component manufacturers (based on their measurements) and can be downloaded from the Internet and inserted into TINA either manually or by using TINA's library manager.

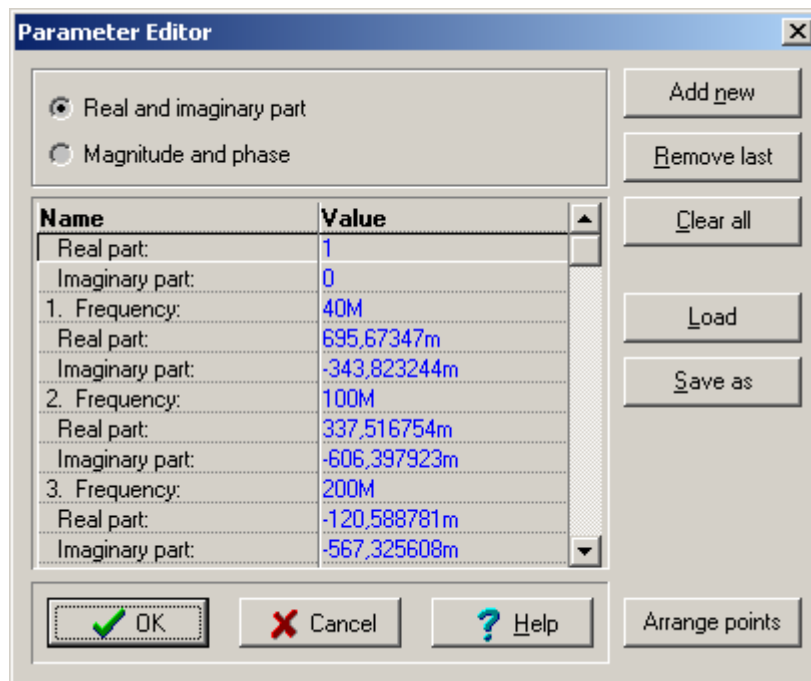
Let's get started with an example. Open the circuit examples\rf\spar_tr.sch.



This circuit consists of a network analyzer connected to an RF transistor. The first port of the network analyzer measures the base-emitter characteristic while the second port measures the collector-emitter characteristic of the transistor. If you double-click on the transistor, you will see that the actual characteristics of this transistor are defined by its S11, S12, S21, S22 parameters, and R1 and R2 input/output impedances.



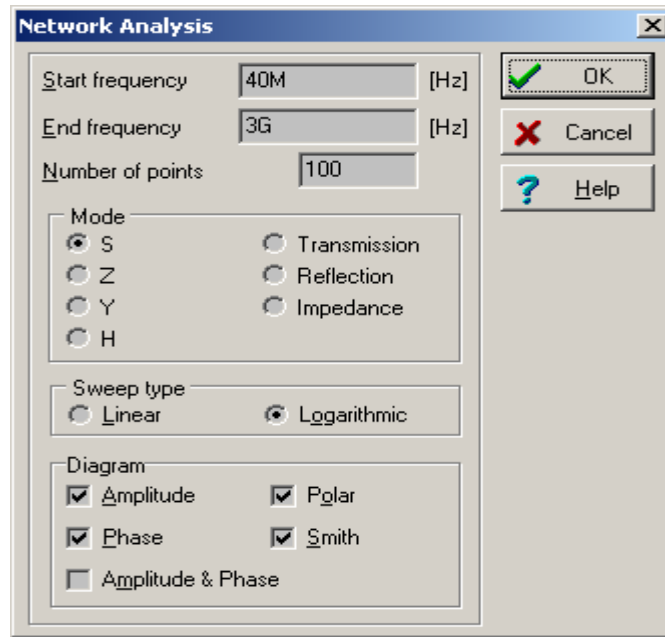
Each function is defined as a table of the S parameter and frequency. If you click on the ... button beside the Complex label, you can see this table.



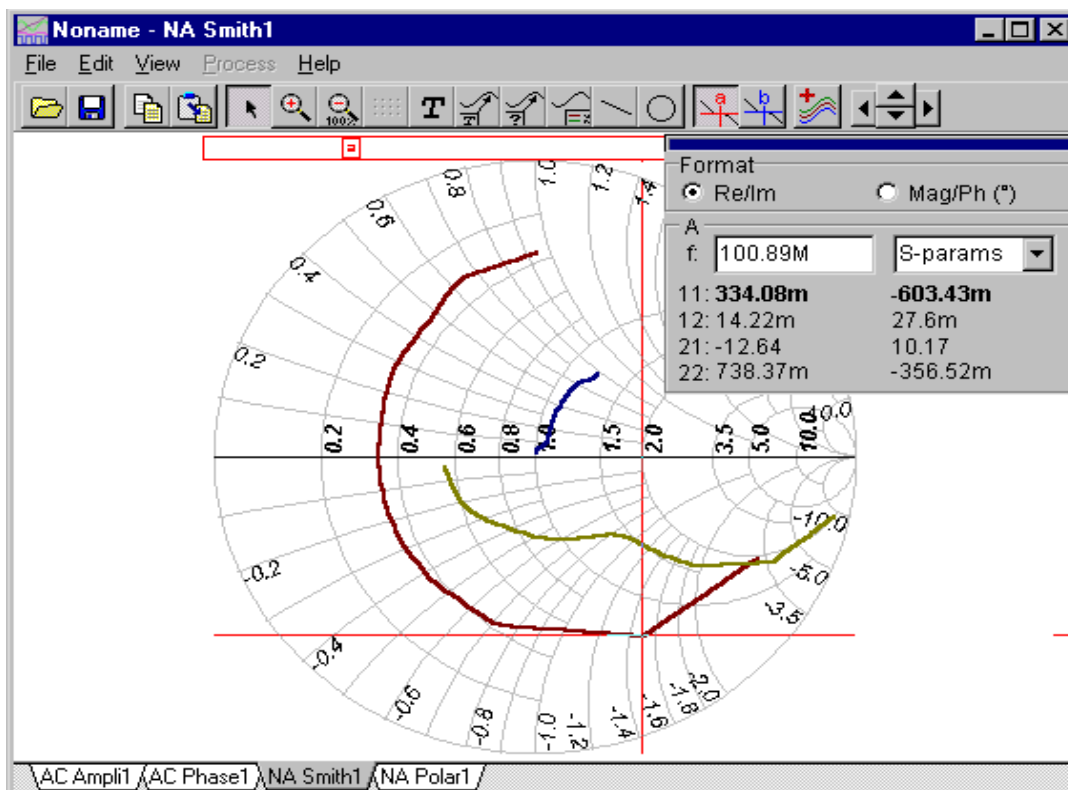
You can easily add more points to the function with the *Add new* button or load an entirely new function with the *Load* button. Similarly, the *Remove last* button removes the last point, and the *Clear all* removes all the points from the list. The *Arrange points* button sorts all the points in ascending order of frequency.

Note that TINA comes with many other models defined by S-parameters (under the RF toolbar).

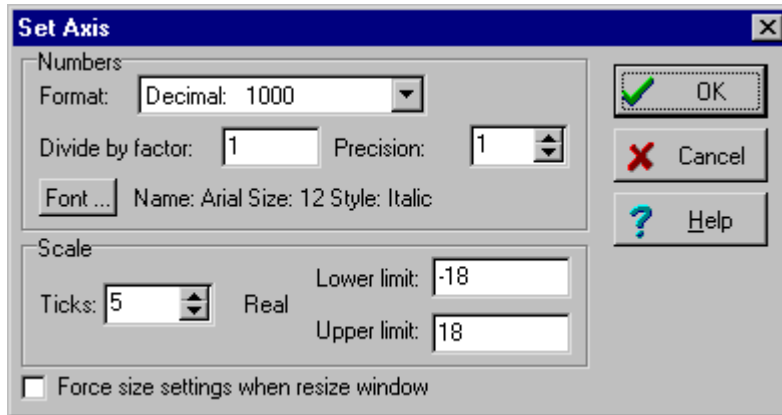
Close all the dialog boxes and run Analysis.AC analysis.Network analysis. Among many other parameters in the dialog, you can specify the frequency range and the mode (i.e., what to draw) of the analysis.



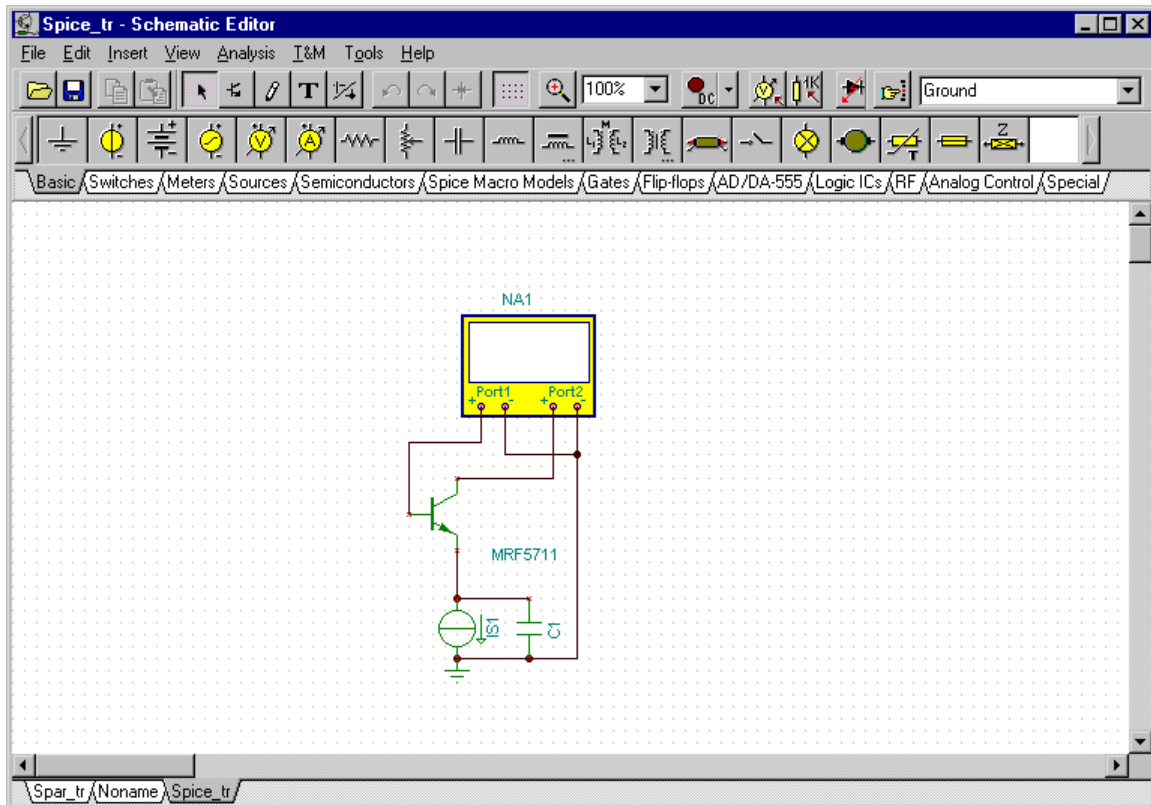
The diagrams available depend on the selected analysis mode. In S Analysis, the Amplitude, Phase, Polar, or Smith diagram can be drawn. Let's select Analysis mode S, the Amplitude, Phase, Polar and Smith diagram types, and press OK. Four new pages are inserted into the diagram window. Each new page contains four curves, the S11, S12, S21, and S22 curves, displayed in the appropriate diagram style. You can obtain specific data values by placing a cursor on the desired points. When you place the cursor on a Smith curve, a special cursor diagram comes up, which can display the curve's values in the format set with the *Format* and *params* (S-params, Z-params, etc.) fields.



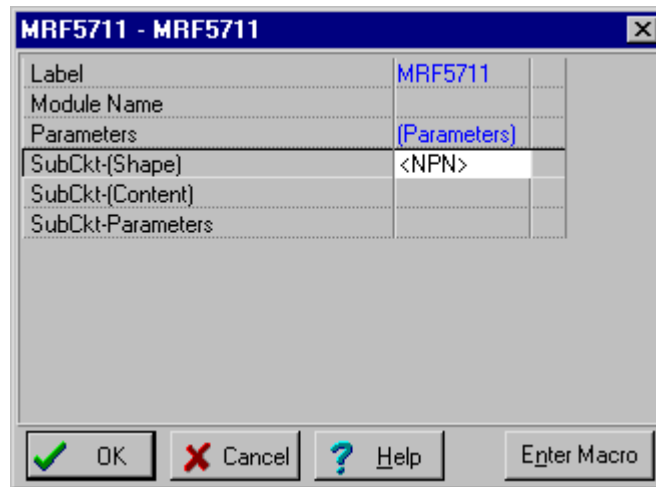
You can annotate these diagrams, as you can any other diagram types, by adding texts, circles, and lines. You can explore data values of particular points on the curves using the cursors, and zoom in and out. But for Smith and Polar diagrams, axis settings are handled differently. Even though these diagrams look like polar diagrams, the coordinate ranges are defined in the real-imaginary domain. The real part (i.e. real axis) can be set by double clicking on the horizontally aligned numbers (on the Polar diagram the amplitude values), and the imaginary part by double clicking on the circularly aligned numbers (on the Polar diagram the Phase values). You can set the format of the numbers and the horizontal or vertical range of the diagram using the dialog that appears.



Models for Network analysis can also be defined using SPICE subcircuits. To demonstrate this, open the circuit examples\rf\spice_tr.sch in the schematic editor. This example contains a circuit similar to the one we've just looked at: a network analyzer connected to a transistor, and a current source and a capacitor.



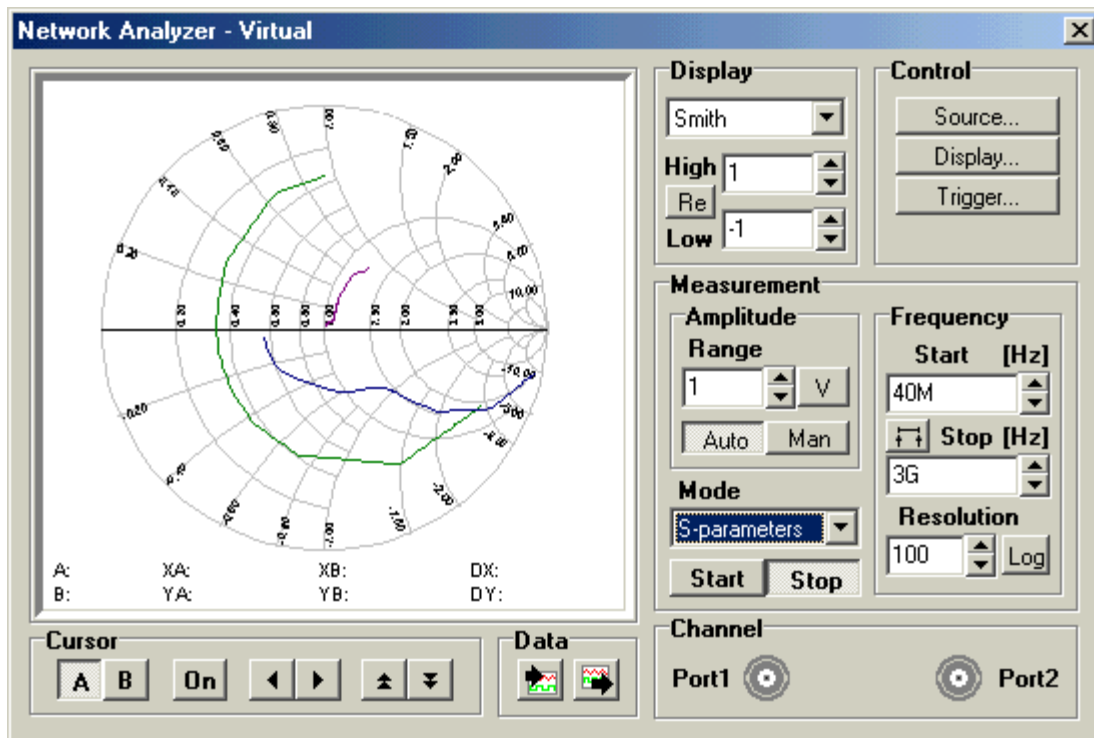
In this circuit, however, the transistor contains a SPICE subcircuit. Its netlist can be viewed by clicking *Enter macro* in its property dialog.



TINA contains many other RF models that can be found under the Spice Macro Models toolbar.

Now run Analysis.AC Analysis.Network Analysis again and note that the results are similar to those obtained with the S-parameter models.

Network analysis curves can also be generated using TINA's Network Analyzer under the T&M menu. To select the diagram you need, first set the Network Analyzer Mode, then the display type. After you press *Start*, the selected curves will be displayed on the instrument.

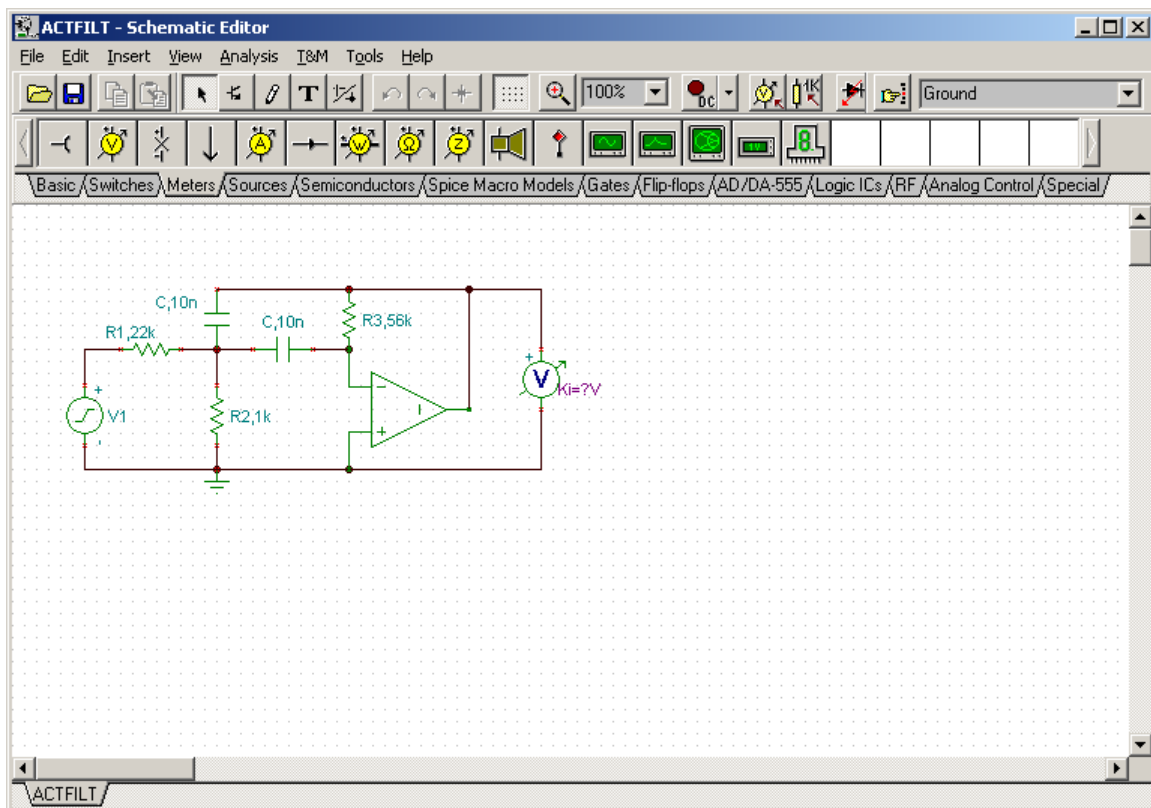


Symbolic Analysis

Symbolic Analysis

TINA's symbolic analysis produces closed form expressions of the transfer function, equivalent resistance or impedance, as well as the DC, AC, and transient response of linear circuits. These equations can then be inserted into the schematic editor or the diagram window. With the help of the built-in interpreter, they can be even further manipulated and displayed as a diagram in the diagram window. The formulas provided by symbolic analysis allow a unique in-depth understanding of circuit operation, invaluable both for design and education.

Let's see a few examples. First load the circuit examples\symbolic\actfilt.sch. It is an active filter with an ideal operational amplifier.



Now run Analysis.Symbolic Analysis.AC Transfer. The transfer function is generated and displayed in the Equation Editor.

Equation Editor

File Edit Settings Help

Transfer function:

$$W(s) = \frac{R_2 \cdot C \cdot R_3 \cdot s}{-R_1 - R_2 - 2 \cdot R_2 \cdot R_1 \cdot C \cdot s - R_2 \cdot R_1 \cdot C^2 \cdot R_3 \cdot s^2}$$

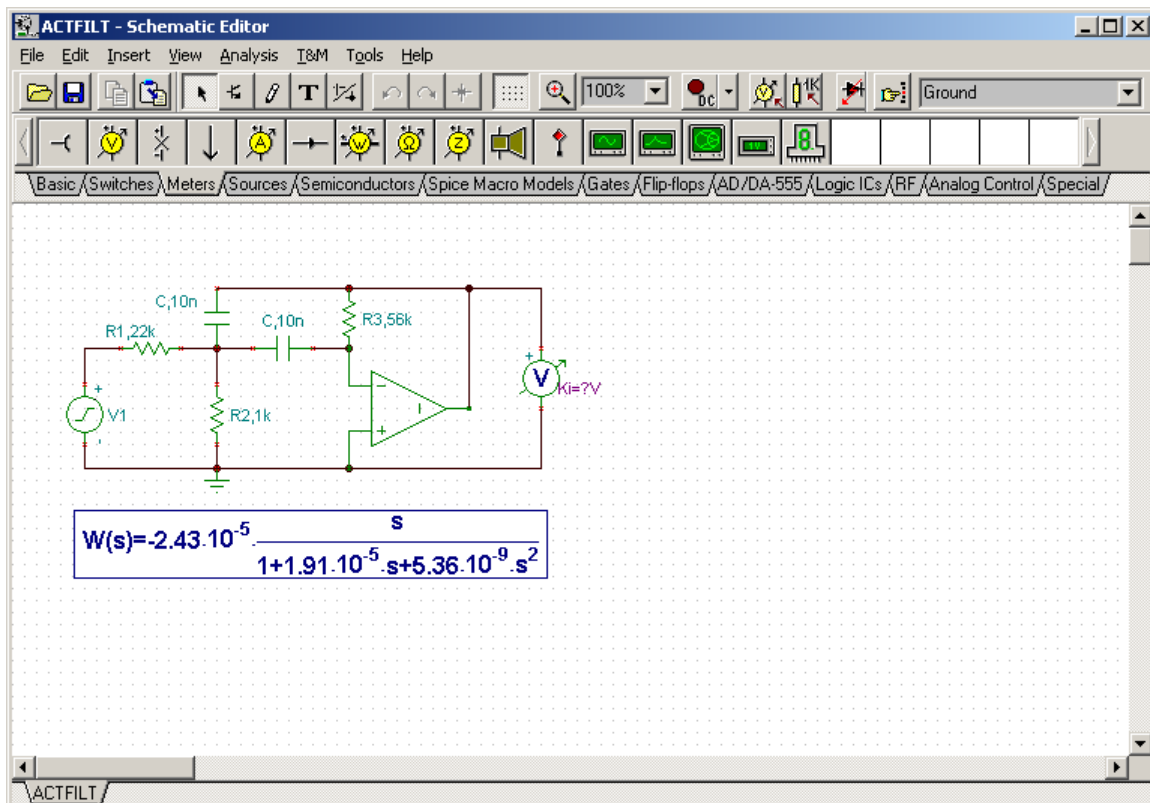
All the variables (values of the capacitors, resistors, etc...) were treated as (unknown) symbols, so the expression contains their symbolic names. However, in case you want to calculate the numerical form of the transfer function using the circuit element's actual values, you need to click on Analysis.Symbolic Analysis.Semi-symbolic AC Transfer, and the following expression will appear.

Equation Editor

File Edit Settings Help

$$W(s) = -2.43 \cdot 10^{-5} \cdot \frac{s}{1 + 1.91 \cdot 10^{-5} \cdot s + 5.36 \cdot 10^{-9} \cdot s^2}$$

This expression can then be copied to the clipboard (by pressing the copy button on the toolbar) and pasted to the schematic editor or to any of the diagram pages.

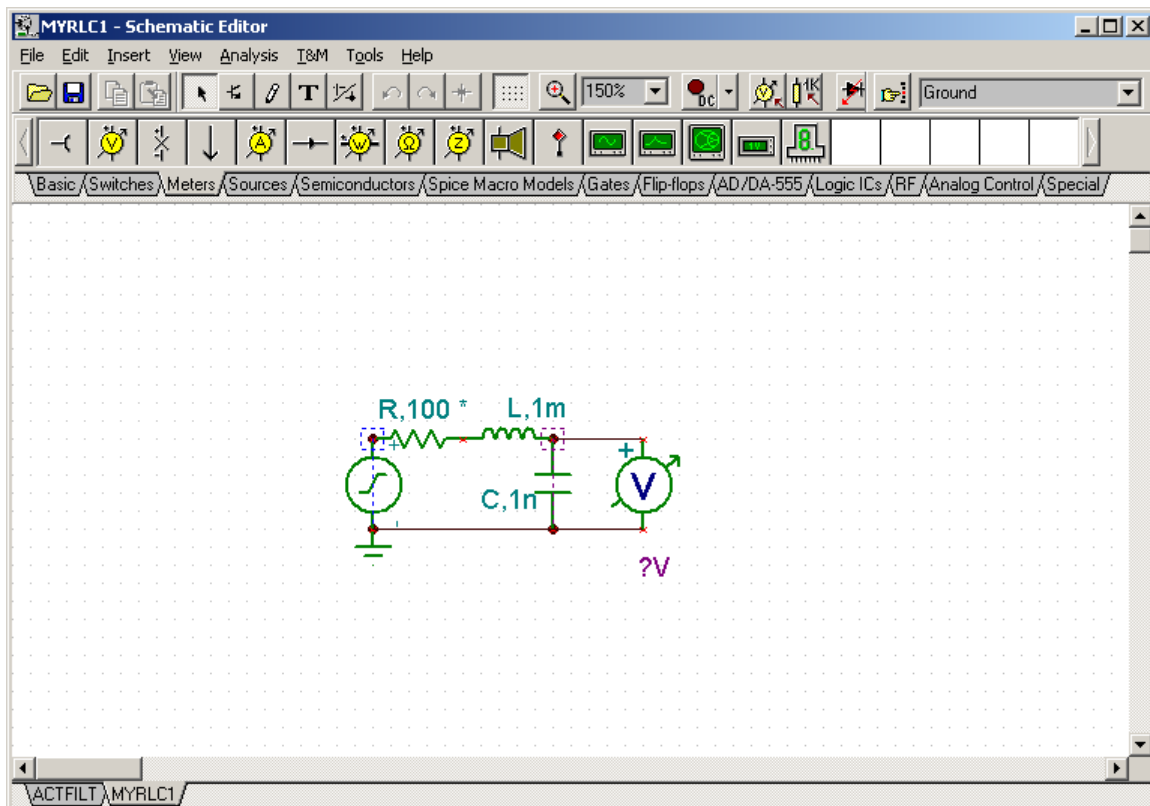


You can also generate a symbolic result with only a few symbolic variables, while the numerical values of the rest of the variables have been substituted. Symbolic analysis takes the symbols of the variable from the circuit element's *labels* in the schematic editor. When the label is empty or has been determined to not be a valid symbol name, symbolic analysis uses the part's numerical *value*. For example, if you change the label of the resistor R1 to "22k" or to "R1=22k", then the value of the part (22k) will be used in place of the symbol R1.

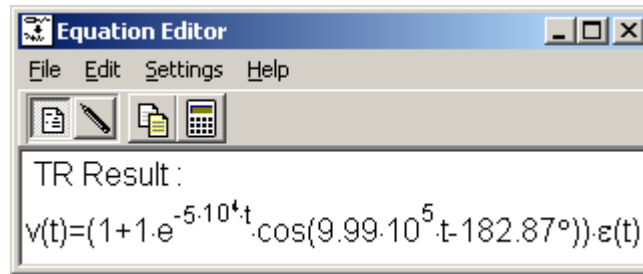
The Equation Editor window displays the following transfer function:

$$W(s) = \frac{R_2 \cdot C \cdot R_3 \cdot s}{-2.2 \cdot 10^4 - R_2 - 4.4 \cdot 10^4 \cdot R_2 \cdot C \cdot s - 2.2 \cdot 10^4 \cdot R_2 \cdot C^2 \cdot R_3 \cdot s^2}$$

TINA goes beyond simply presenting the function as an equation. Any of these functions can also be interpreted and drawn in the diagram window. To demonstrate this, let's first generate the transient response of an RLC series resonant circuit. Load the examples\symbolic\myrlc1.sch circuit into the schematic editor.



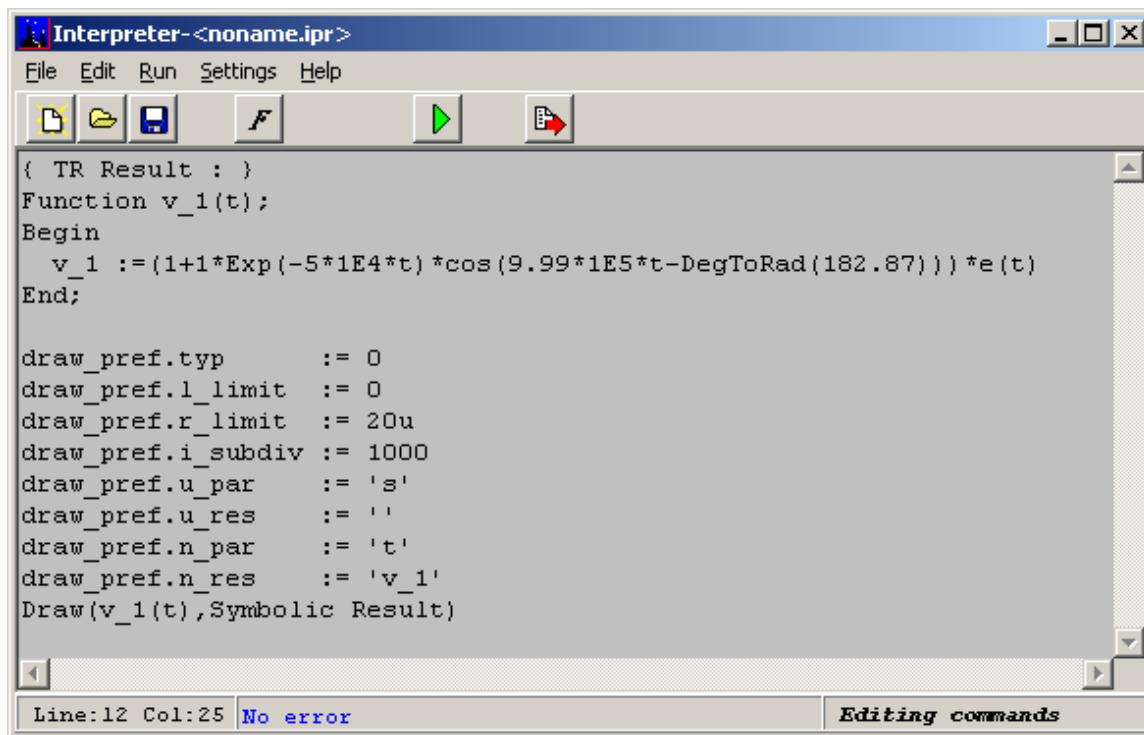
Next, run Analysis.Symbolic Analysis.Semi-symbolic Transient. The transient response of this circuit to a voltage generator with Unit step waveform is then the following:



The Equation Editor window displays the following equation:

$$v(t) = (1 + 1 \cdot e^{-5 \cdot 10^4 \cdot t} \cdot \cos(9.99 \cdot 10^5 \cdot t - 182.87^\circ)) \cdot \varepsilon(t)$$


Now press the small calculator (interpreter) button on the toolbar. The interpreter window appears, and our transient function is already defined as a v_1 function.

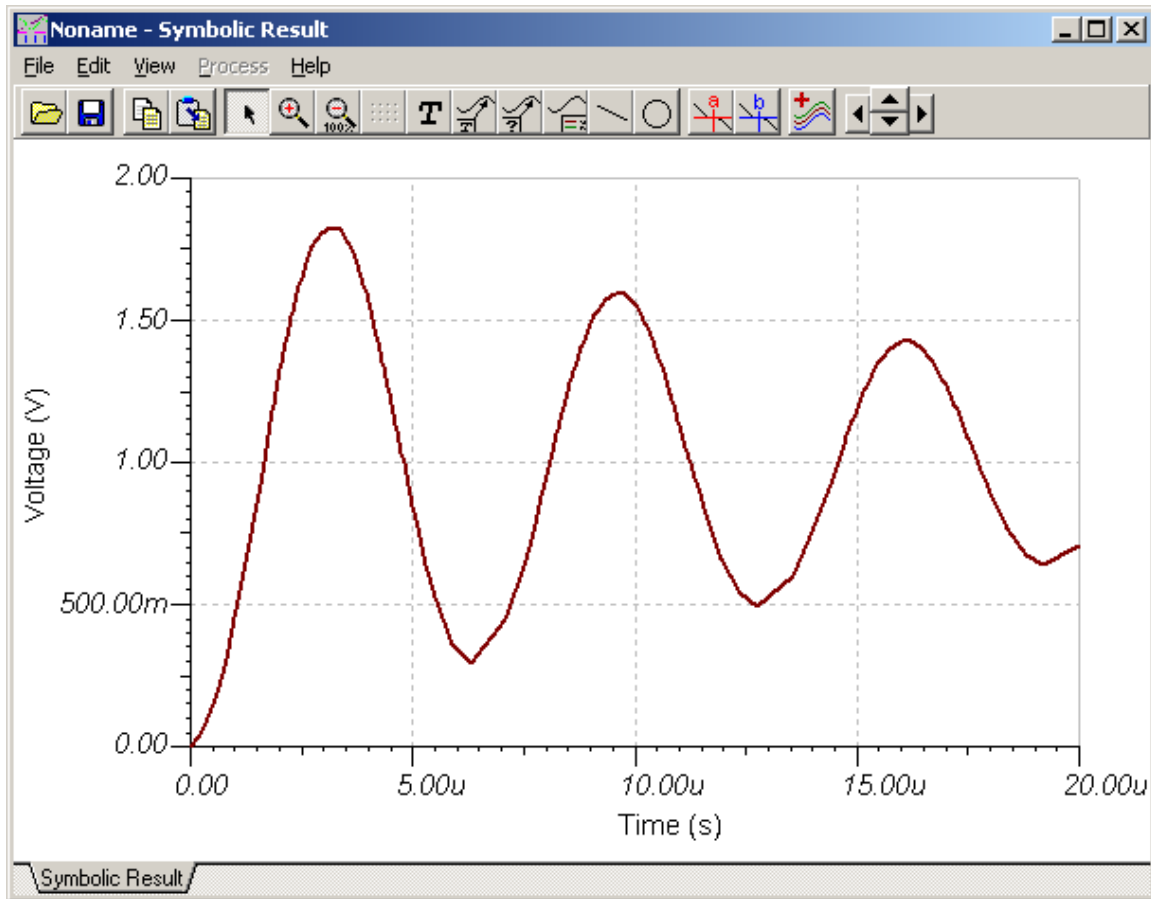


```
{ TR Result : }
Function v_1(t);
Begin
  v_1 := (1+1*Exp(-5*1E4*t) *cos(9.99*1E5*t-DegToRad(182.87))) *e(t)
End;

draw_pref.typ      := 0
draw_pref.l_limit  := 0
draw_pref.r_limit  := 20u
draw_pref.i_subdiv := 1000
draw_pref.u_par    := 's'
draw_pref.u_res    := ''
draw_pref.n_par    := 't'
draw_pref.n_res    := 'v_1'
Draw(v_1(t),Symbolic Result)
```

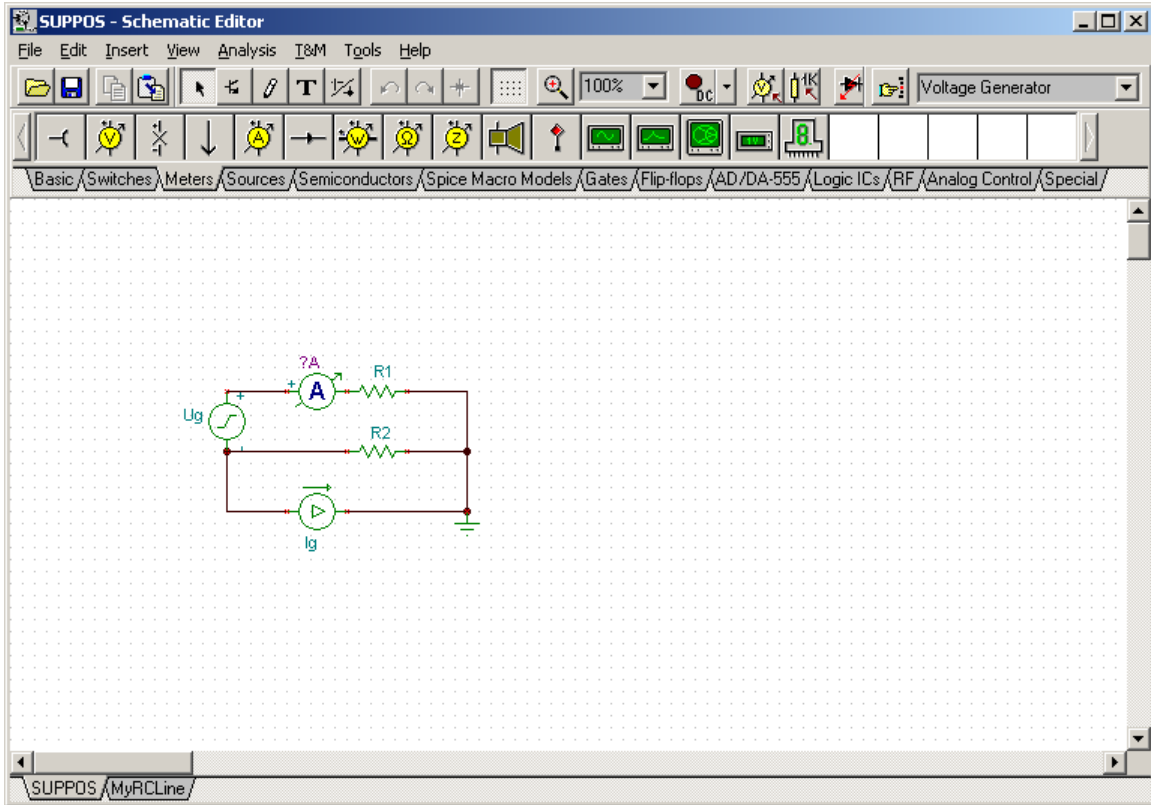
Line:12 Col:25 No error Editing commands

As you can see, the function definition is followed by a few drawing preference settings, where you can define the ranges of the function and the number of points at which the function will be calculated. Finally, the last command draws this function in the diagram window. The second parameter of the draw command defines the name of our newly generated function. After pressing the  Run button, the function appears in the diagram window.



Symbolic analysis can also calculate the DC and AC results (response) of a linear network containing one or more sources. Just as with symbolic analysis of AC transfer functions, these results can be generated both in symbolic and semi-symbolic form. If we choose the symbolic form, all variables in the circuit that have symbolic identifiers (labels) will be treated as pure symbols. If we choose the semi-symbolic form, every parameter of the circuit will be handled as a number. When the circuit contains more than one generator (source), the result due to each of these generators will be calculated and added together (using the superposition law).

Let's use a simple parallel resistor circuit to get to know symbolic analysis of circuit results. Open the circuit examples\symbolic\suppos.sch.



This circuit contains two generators: a voltage and a current generator. Now press Analysis.Symbolic Analysis.DC Result.

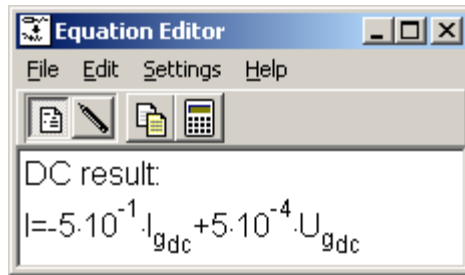
The screenshot shows the Equation Editor window. The title bar reads 'Equation Editor'. The menu bar includes 'File', 'Edit', 'Settings', and 'Help'. Below the menu bar is a toolbar with icons for a document, a pencil, a calculator, and a numeric keypad. The main text area displays the following equation:

$$\text{DC result:}$$

$$I = \frac{R_2}{-R_2 - R_1} I_{gdc} + \frac{1}{R_2 + R_1} U_{gdc}$$

As you can see the result is a fully symbolic combination of the circuit's response to each of the generators.

Now press Analysis.Symbolic Analysis.Semi-symbolic DC Result.

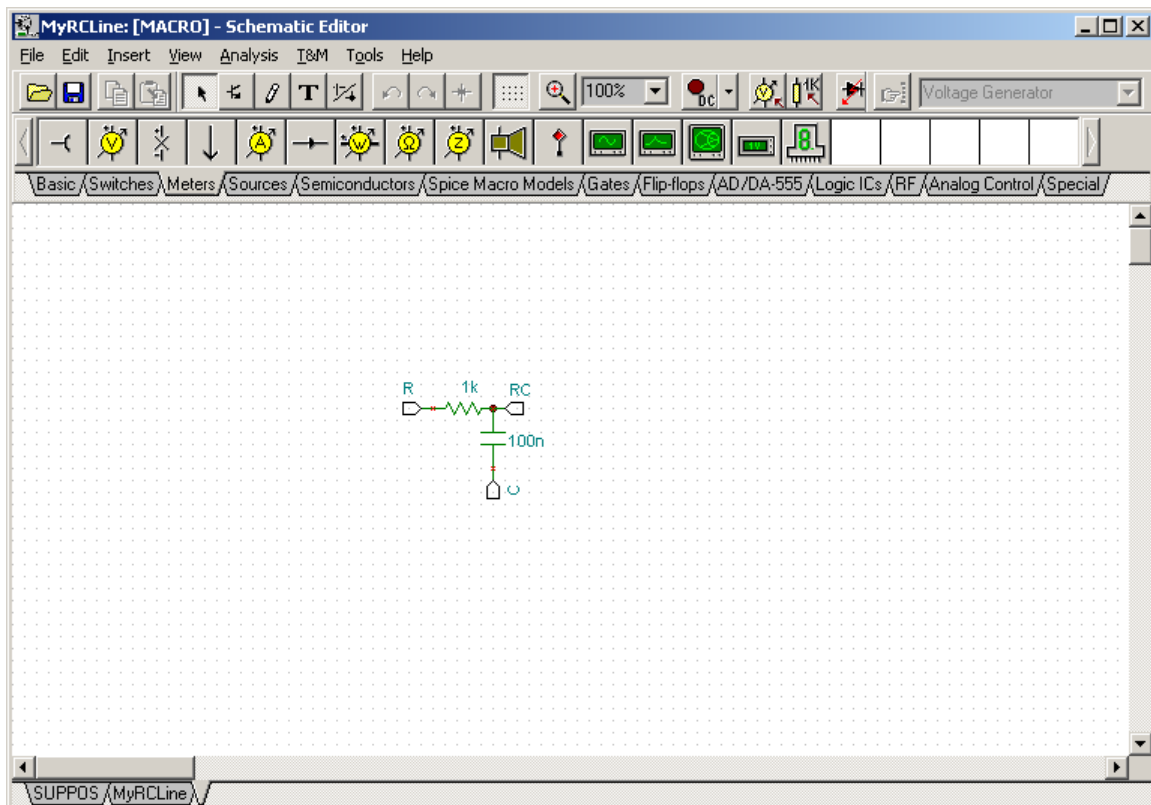


In this Semi-symbolic DC Result, the coefficients of I_{gdc} and U_{gdc} are calculated numerically. Of course, this requires that R1 and R2 be assigned both a label and a numerical value.

Hierarchical Semi-symbolic Analysis

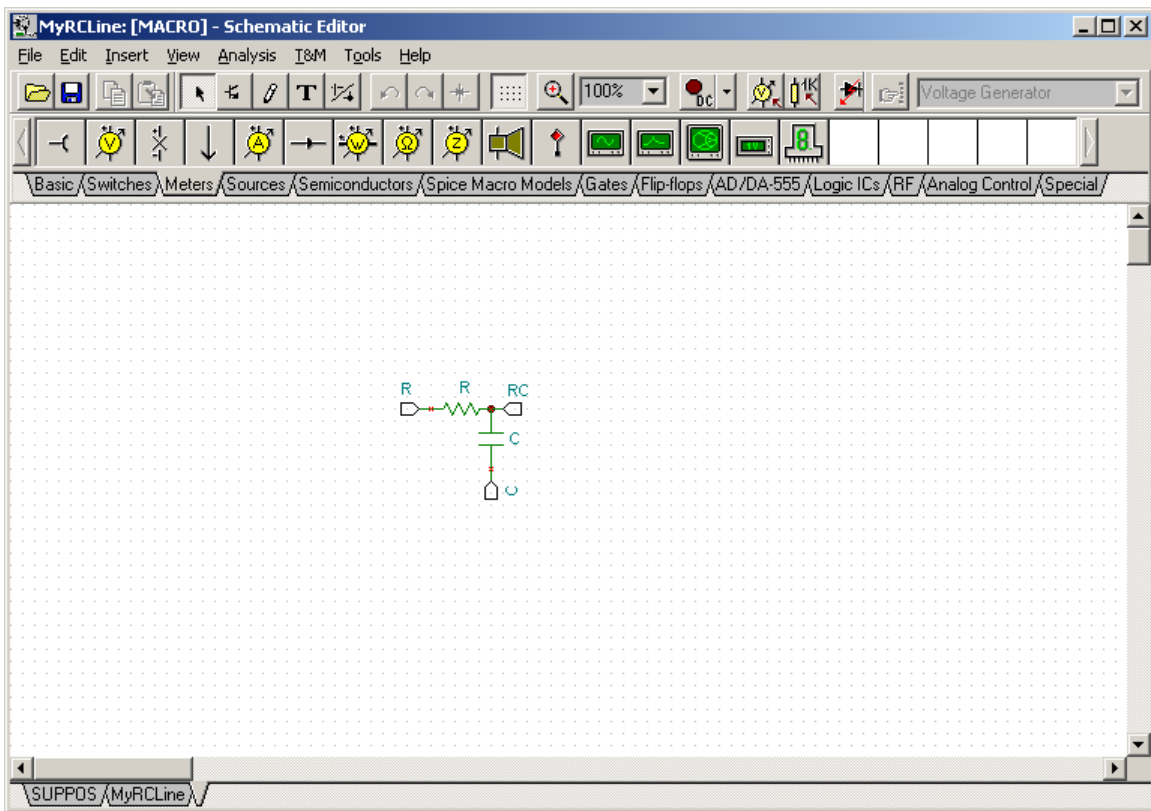
Depending on the circuit size and topology, the symbolic result of an average circuit can easily become a huge, complicated expression. This is especially true for circuits that contain components whose models contain further symbolic components, such as transistors.

The symbolic expressions can be simplified by identifying parts of the circuit that are less important, or whose effect are already better understood, and setting only numeric values (no entry in the label field) for these components. For other components that are more important, be sure to enter a label so that they will be treated symbolically in the analysis. For visual convenience you can use TINA's advanced macro feature to create two different macros for each of the component you would like to examine in detail: one with a symbolic model and another with a purely numeric model. To demonstrate this feature, let's create an RC transmission line where some of the RC sections are numeric and others are symbolic. The numeric macro will look this.

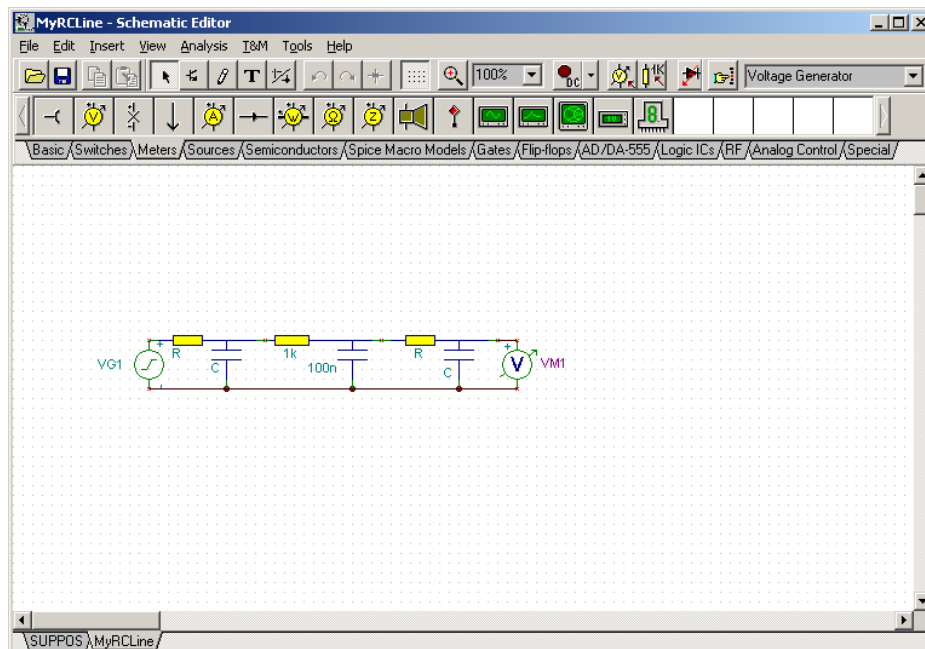


The resistor is 1k Ohm, and the capacitor is 100nF.

The symbolic model (below) has the same topology, but has symbolic R and C parameters.



Now let's consider an RC transmission line circuit with three RC sections. Only the middle section is numeric.



Pressing Analysis.Symbolic Analysis. AC Transfer, TINA calculates the result:

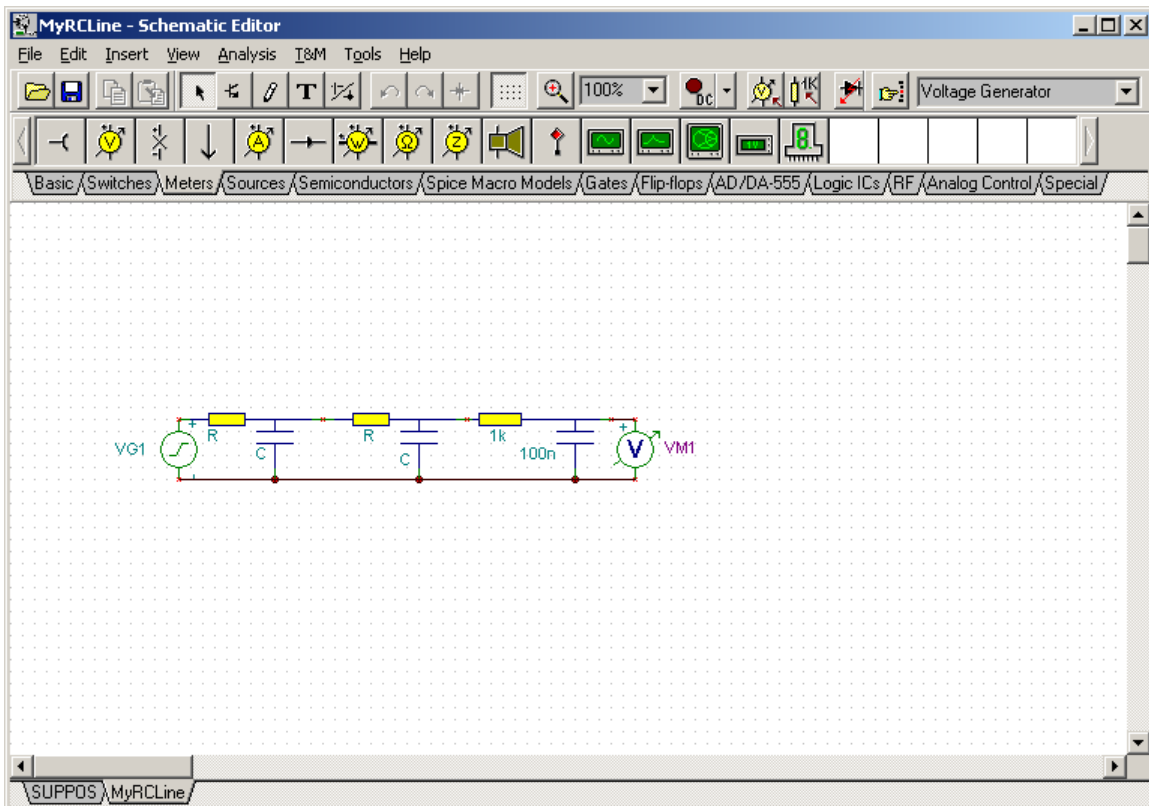
Equation Editor

File Edit Settings Help

Transfer function:

$$W(s) = \frac{10 \cdot 10^6}{10 \cdot 10^6 + (R + 3 \cdot 10^7 \cdot C \cdot R + 1000 + 10 \cdot 10^9 \cdot C) \cdot s + (R + 10 \cdot 10^6 \cdot C \cdot R + 2000 + 10 \cdot 10^9 \cdot C) \cdot C \cdot R \cdot s^2 + 1000 \cdot C^2 \cdot R^2 \cdot s^3}$$

If we change the position of the numeric section of our circuit (by simply dragging the whole macro to its new position), we see:



And the result changes as well:

Equation Editor

File Edit Settings Help

Transfer function:

$$W(s) = \frac{10 \cdot 10^6}{10 \cdot 10^6 + (3 \cdot 10^7 \cdot C \cdot R + 2 \cdot R + 1000) \cdot s + (3000 + R + 10 \cdot 10^6 \cdot C \cdot R) \cdot C \cdot R \cdot s^2 + 1000 \cdot C^2 \cdot R^2 \cdot s^3}$$


Note that TINA's built in transistor models are purely symbolic: i.e., the bipolar transistor contains a base resistor and a current controlled voltage source, while the MOS and JFET contain a drain-source resistance and a voltage controlled current source. If you want to calculate the DC or AC symbolic transfer or result function of a circuit with several transistors, you can simplify the expressions by creating your own transistor models using only numeric components.

Post-processing Analysis Results

Adding new curves to diagrams and post-processing analysis curves

TINA has a great tool to add new curves of virtually any node and component voltage or currents. In addition, you can post process existing curves by adding or subtracting curves, or by applying mathematical functions. You can also draw trajectories; i.e., draw any voltage or current as a function of another voltage or current.

If you check “Save all analysis results” in the Analysis.Options dialog, TINA will calculate and store all the results that are reasonable in the current circuit. However, only those results that have been set as output are inserted automatically into the diagram window.

To add these curves, or to add new, user-defined curves to a diagram, press the *Add more curves* button  on the toolbar or call the *Add more curves* menu in the Edit menu. The post-processing dialog appears on the screen.

This dialog box consists of two parts: in the upper part you can select and insert/remove existing curves into/from the current diagram, while in the lower part you can define your own curves. The lower part of the dialog box is hidden by default. Press the “More >>” button if you want to create new curves.

Upper part:

When this dialog first opens, the *Available curves* list box contains all the curves that have already been calculated. The status bar shows the circuit file name and the analysis which was run. You can select and add any of the curves into the *Curves to insert* list box by selecting them in the *Available curves* list and pressing “Add>>”. Similarly, you can move back any curves from the *Curves to insert* listbox into the *Available curves* list box by selecting them and pressing <<Remove. To make selection easier, you can filter out some of the available curves by unchecking any of the *Show* types. When you press OK, all the curves in the *Curves to insert* list box will be inserted into the active diagram page.

TINA offers another way to select curves for inclusion in the diagram. With the Post processing dialog box open, move the cursor over the schematic. The cursor will change into a pencil shape. Move it over a node or circuit component in the schematic and click. The corresponding curve will be selected in the Available Curves list, and you can add it to the diagram by clicking the Add>> button.

Selected curves can also be deleted with the Delete button. Deleted curves will be removed both from the list boxes and from the dialog window pages!


Lower part:


This part is hidden by default. If you want to create new curves, press the “More >>” button.


Here you can define your own curves using arithmetic operators and mathematical functions. These custom curves must be defined either as a single-line mathematical expression or as a Pascal-like function.

Take special care if you transform the independent variable of curves; for example, writing $v(t+1)$ instead of $v(t)$. The transformed variable expects results from beyond the interval for which the calculation was run. TINA in this case would have no choice but to set the new curve to zero outside of the calculated interval.

While single-line expressions can be entered in the *Line Edit* field, more complex functions should be defined in the *Advanced Edit* field. The *Advanced Edit* checkbox enables/disables the *Advanced Edit* field.

Pressing the  “Add to edit area” button above the *Line Edit* field, you can insert the currently selected curve from the *Available curves* list into the editor field.

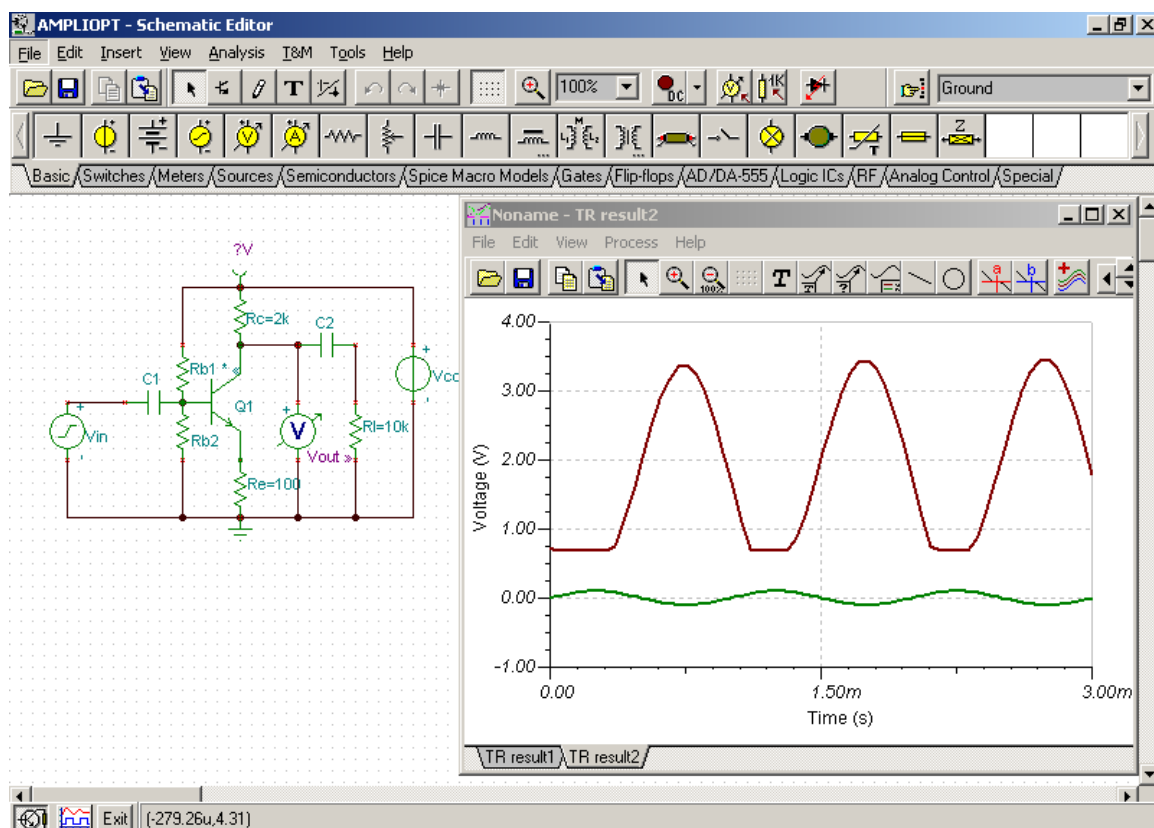
Pressing the  “Add to edit area” button to the right of the Built-in Functions drop down list, you can insert the currently selected function from the Built-in Functions list into the editor field.

You can also display a result as a function of another result (e.g., $v(t)$ as a function of $i(t)$) using the *XY Plot* checkbox. If you check this box, the *Line Edit* field is divided into an X-part and a Y-part. Bring the curve names into the two fields using the  “Add to Edit-area” button or type the curve names. You can use several curves, arithmetic operations, and functions—but you cannot use *Advanced Edit* in this mode.

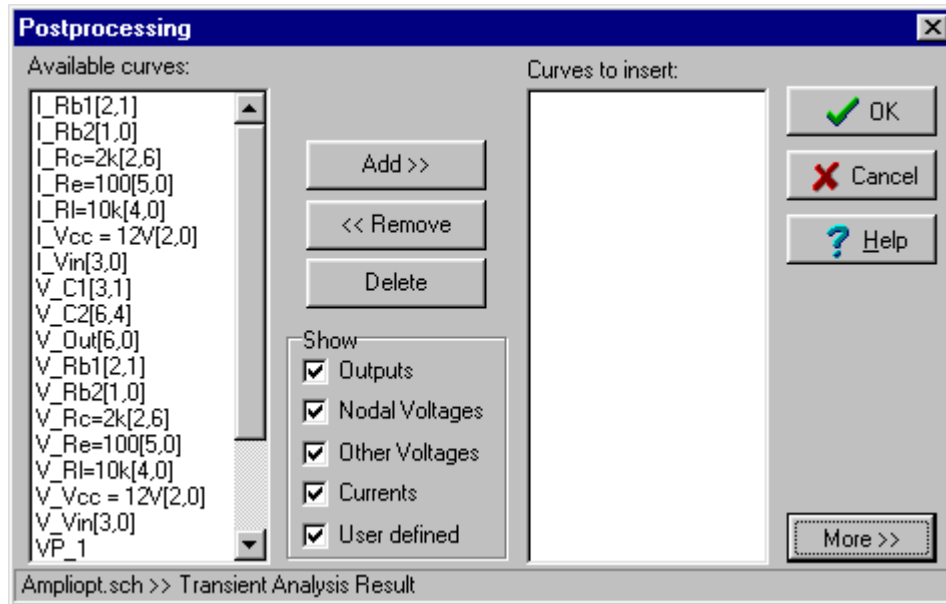
You can use the “*New function name*” edit field to enter your name for the new function (curve).

After you have completed the definition, you can either draw your new curve directly into the diagram window by pressing *Preview*, or you can first store the result by pressing *Create*, and then insert it with the OK button.

Now let’s run a transient analysis on the AmpliOpt circuit in the *EXAMPLES* directory. Two curves are drawn in the diagram window: the input, V_{in} , and the collector voltage, V_{out} .




Now press the *Add more curves* button on the toolbar. The post-processing dialog appears. The *available curves* listbox on the left side shows all the curves that have already been calculated.



Here, the symbols $V_{label[i,j]}$ and $I_{label[i,j]}$ denote the voltages and currents respectively of the *labeled* components between the node i and j . The symbol VP_n denotes the nodal voltage of the node n .

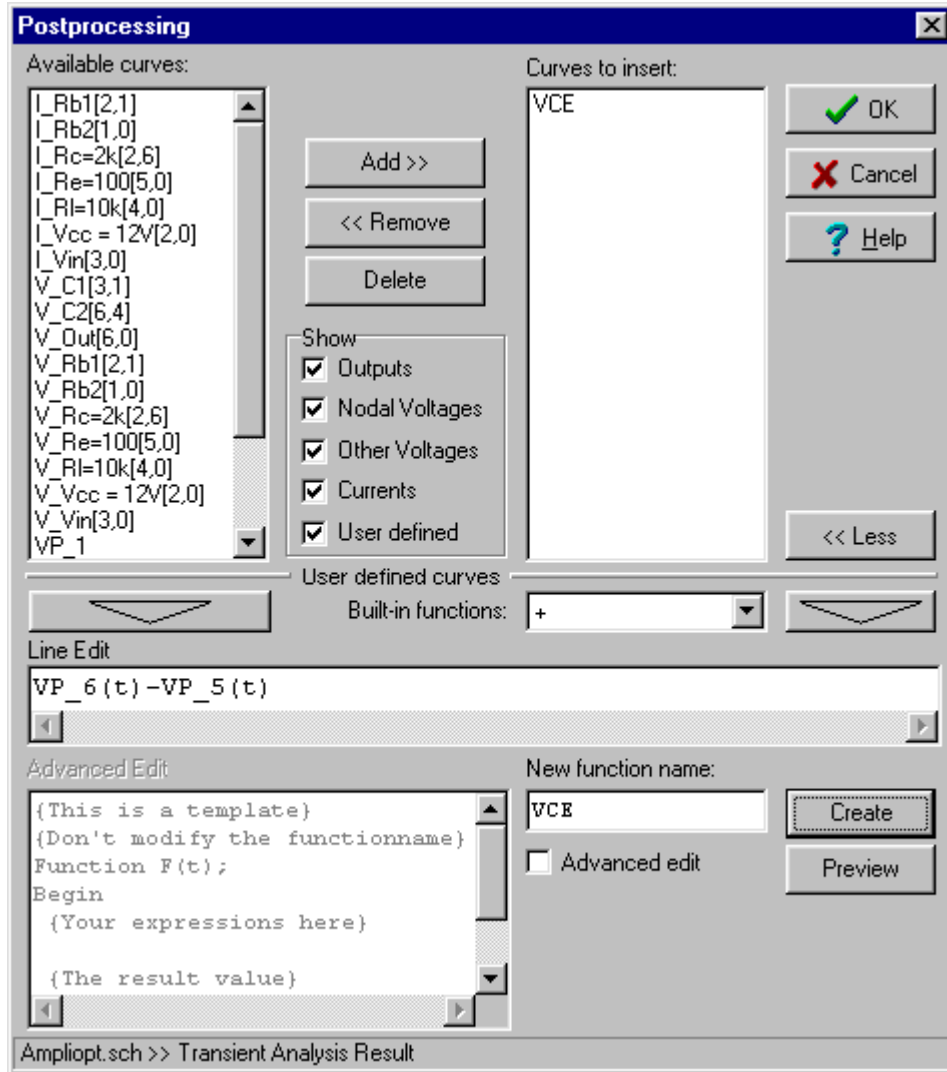
Now, select $V_{C2[6,4]}$ (i.e., the voltage of capacitor C2 connected between node 6 and 4) and press the *Add>>* button. Pressing *OK* inserts this curve into the current diagram page.

As the next step, let's draw the voltage across the transistor's collector and emitter. Press the *Add more curves* button again and then press the *More>>* button to reveal the lower part of the diagram. Type VP_6 - VP_5 in the *Line edit* field.

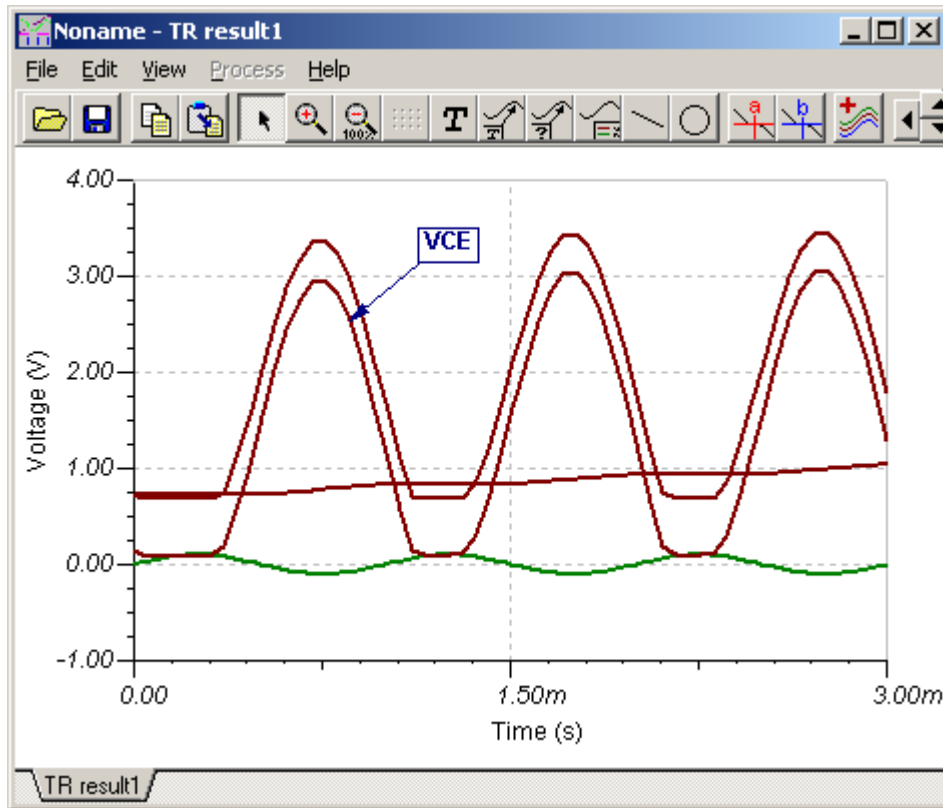
Note: another, perhaps easier way, to insert VP_6 or VP_5 into the *Edit* field is to move the cursor over Q1's collector or emitter, click, and press the  "Add to Edit-area" button below the listbox.

Note also that if you double click on any node in the circuit with the mouse, the corresponding nodal voltage symbol will be automatically inserted into the *Line edit* field at the current cursor position.

Now change the default name of our function to *VCE* and press *Create*.



A new function (VCE) is created and inserted automatically into the *Curves to insert* listbox. If you press OK, our new curve will immediately appear on the diagram.



Sometimes the definition of a new curve requires more editing space or other variables. In these cases we can use the post-processing dialog's *Advanced edit* field. To illustrate this, let's draw the output power dissipation of a differential amplifier.

Open the circuit called DiffAmp in the examples directory and run a transient analysis. Press *Add more curves* again and place a check in the *Advanced edit* field. Position the cursor in the *Advanced edit* field below the comment *{Your expressions here}*.

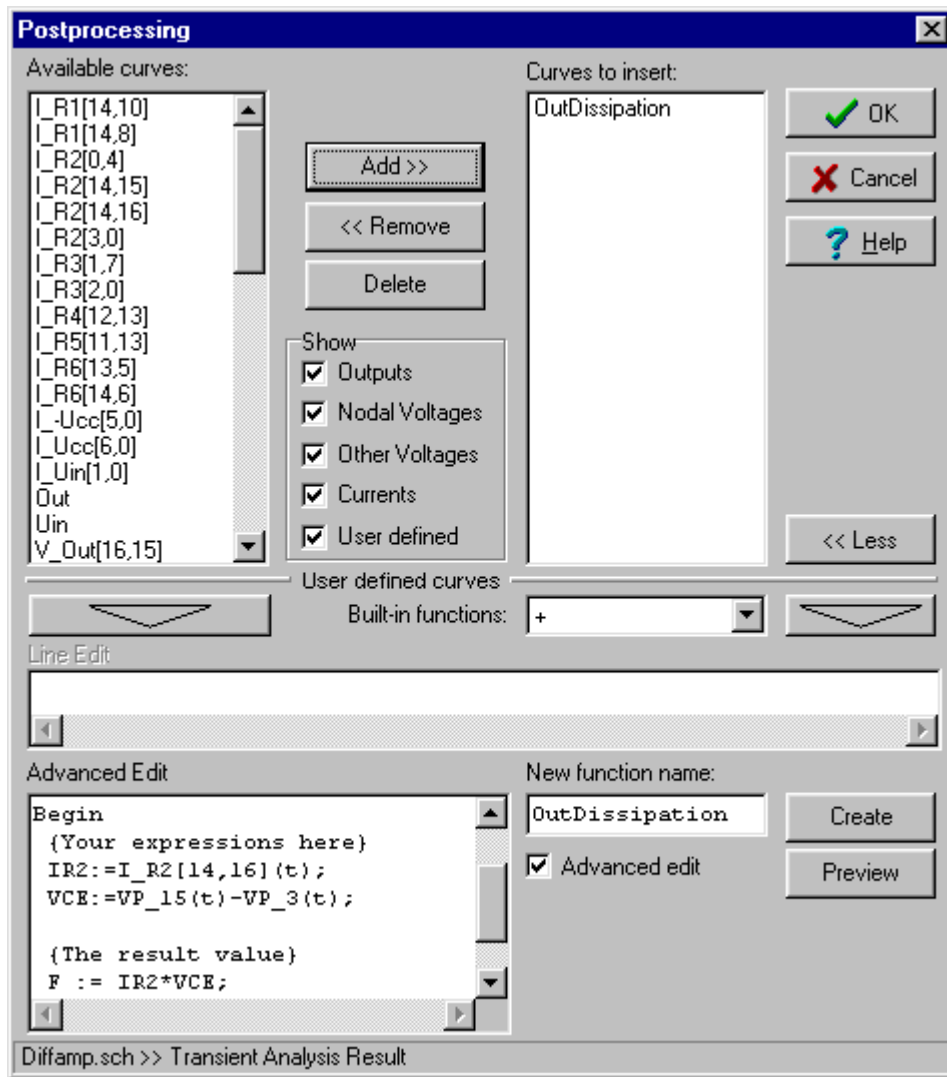
Type :

```
IR2 := I_R2[14,16](t);
VCE := VP_15(t)-VP_3(t);
```

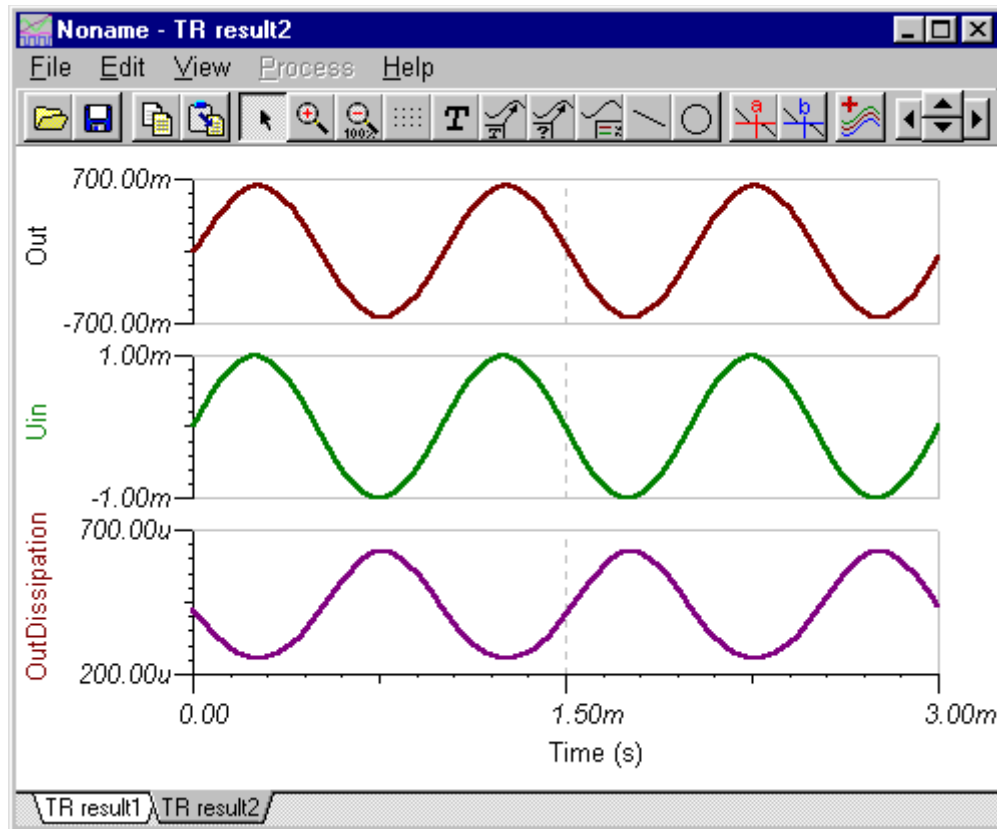
Change the line `F := 1;` *{The result value}* to:

```
F := IR2*VCE;
```

Enter the name of our new function, OutDissipation, and press Create.

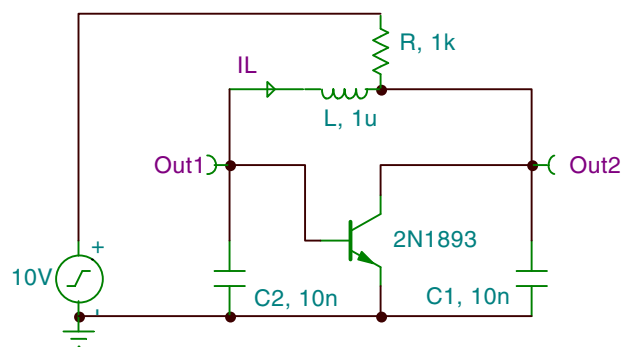


Finally, press OK to draw our new function in the diagram window.



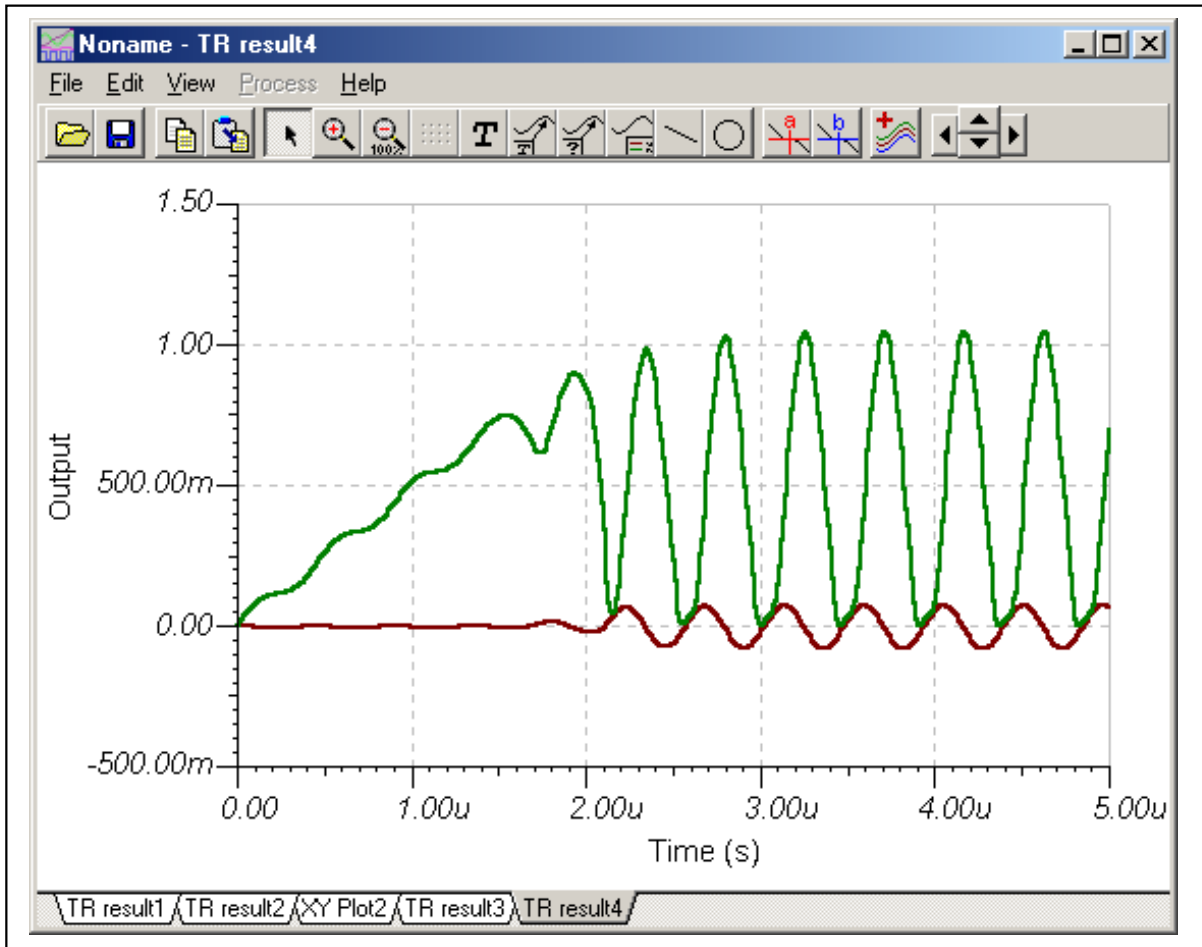
Drawing trajectories (xy-plots) in TINA



In the Post-processor you can also display a signal as a function of another signal, by drawing XY-plots.

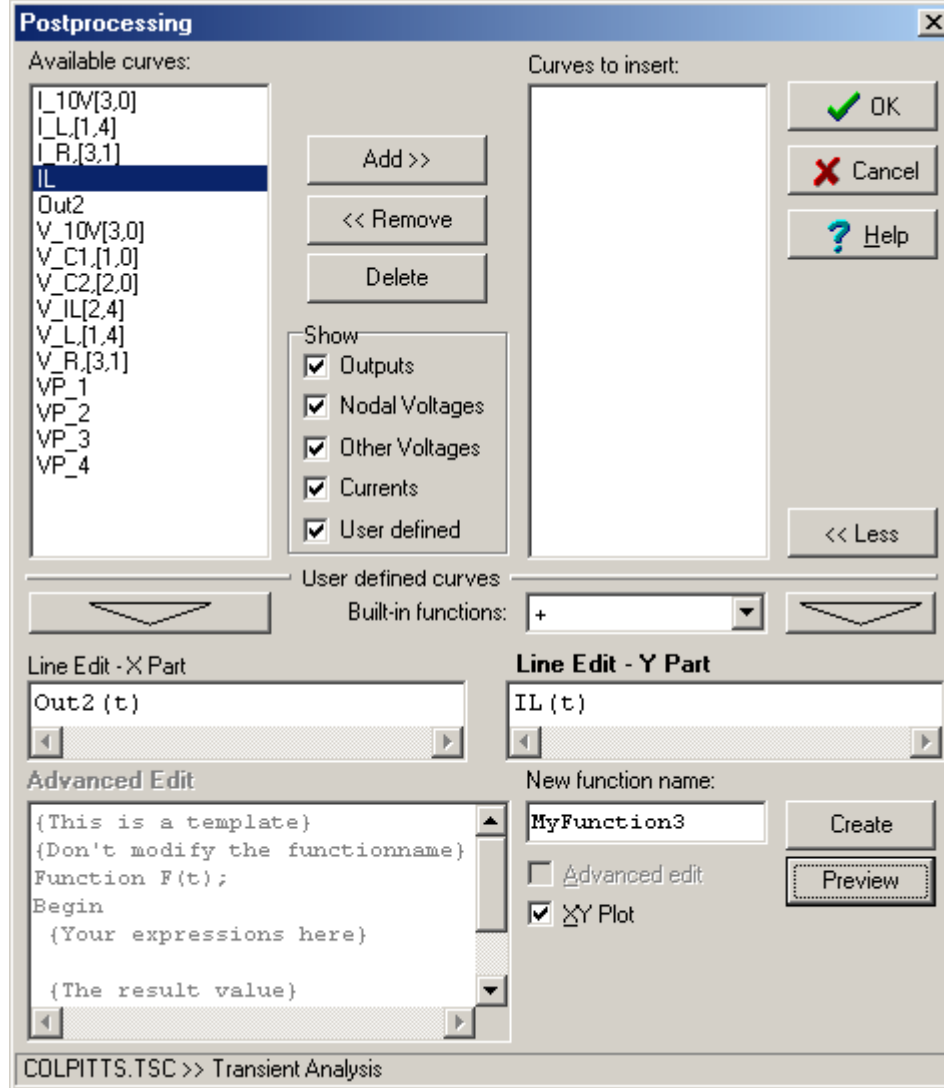


For example, let's draw the voltage of capacitor voltage Out2 as a function of the inductor current IL in the oscillator shown in the figure above.

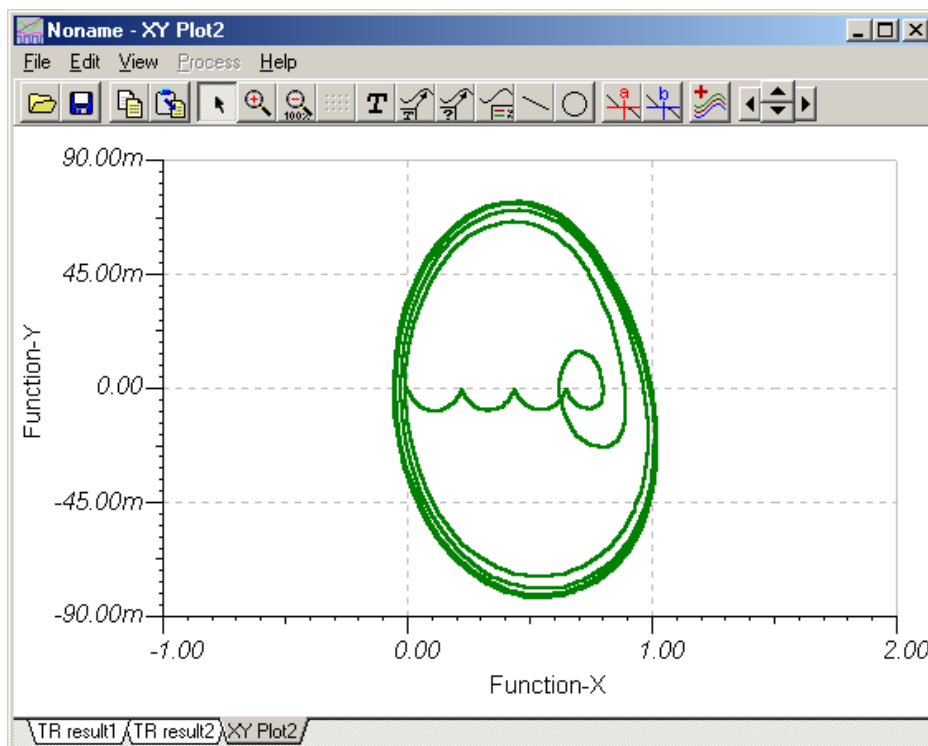
First load the circuit limitc.tsc from the *EXAMPLES* directory. Run the Analysis.Transient analysis. Now the diagram window shows two curves, the capacitor voltage, out1, and the inductor current, IL.



Press the  *Add more curves* button and check the XY-plot checkbox. As you can see in the next figure, the line-edit section is divided into an X-part and a Y-part. Type Out2(t) in the X part and IL(t) in the Y part and press Create and OK. Alternatively, you could have brought the names into the two fields using the  "Add to Edit-area" button.







The calculated curve is shown in the next figure. Place a cursor on any curve and move the cursor. As the curve starts from the (0,0) point and converges to a closed trajectory, we see a “limit cycle.” Limit cycles are very important in the theory of oscillators.




Optimization

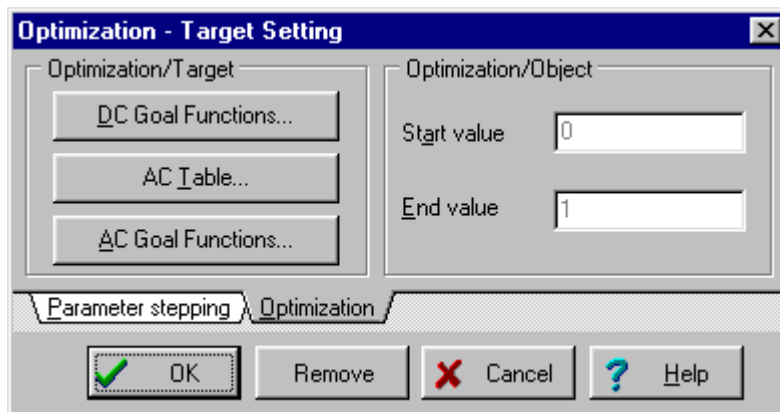
TINA's optimization tool will determine for you one or more unknown circuit parameters (Control Object) which satisfy a predefined circuit target response (Optimization Target). The target circuit response (voltage, current, impedance, or power) must be "monitored" by a meter—or meters—previously inserted at the desired location(s). The unknown circuit parameter(s) will be determined automatically so that the circuit will produce the target output. Two alternative iterative methods (simple and pattern search) are used to find the optimum.

Use the commands Analysis>Control Object, Analysis>Optimization Target, or the speed buttons Select Control Object , Select Optimization Target  to set up Optimization mode or to change their settings.

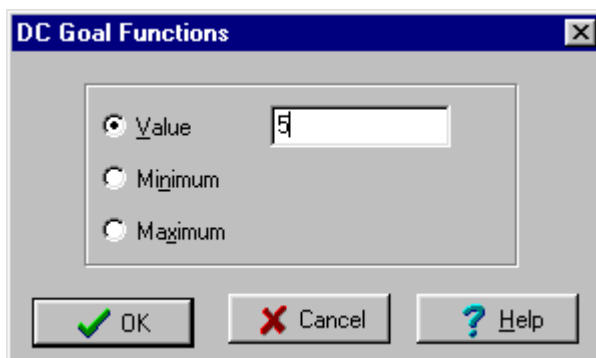
First, place the voltage, current, impedance or power meter(s) onto the schematic to represent the Optimization Targets. Click on the Select Optimization Target speed button  or select the Analysis>Optimization Target menu item. A special cursor  will appear in the schematic editor window, waiting for you to identify the meter. Should you need more meters to represent the optimization targets, repeat the procedure above for the other meters.

There are several target functions you can choose depending upon the analysis mode of the optimization (DC or AC).

For DC optimization problems, you'll need DC goal functions. When you click on a meter with the special target selector  cursor, a dialog box appears showing several goal functions:

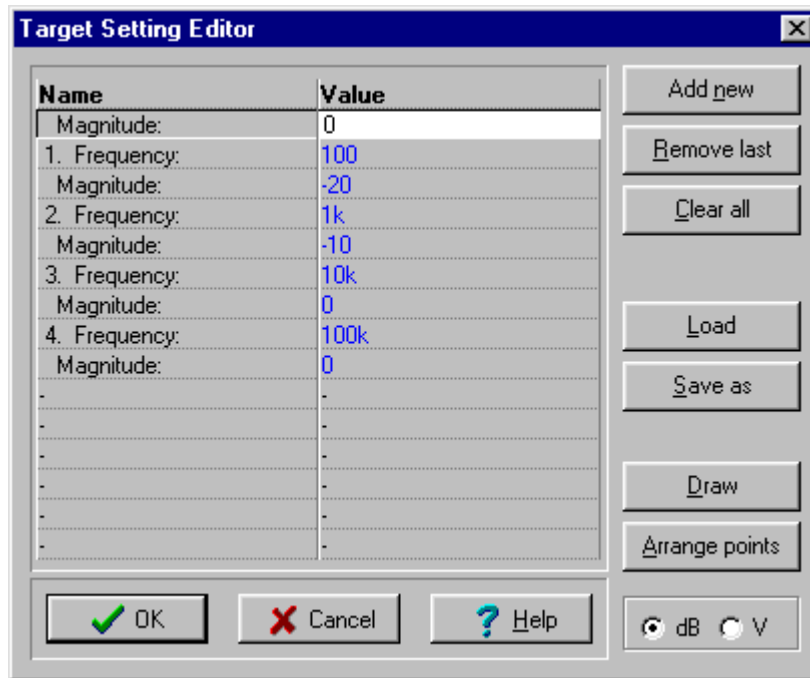


Click on the DC Goal Functions... button. Another dialog box appears which allows you to set a DC goal function:

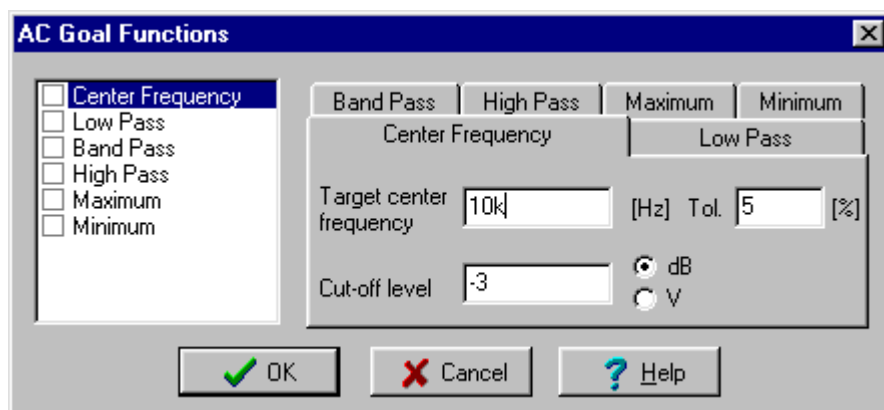


To optimize to a target Value, enter the desired value. To optimize to a Maximum or Minimum, select the appropriate option.



For AC optimization problems, you'll need AC goal functions. With a meter selected, click on the AC Table... or on the AC Goal Functions... button. The AC Table supports a special AC goal function defined by a series of frequency and amplitude values. Using these frequency-amplitude pairs, an AC transfer function can be defined.



You can access other AC goal functions by pressing the AC Goal Functions... button. Here you'll find Center Frequency, High Pass, Band Pass, and Low Pass target functions, as well as Minimum and Maximum target functions. You can combine these functions, effectively setting more than one criterion.

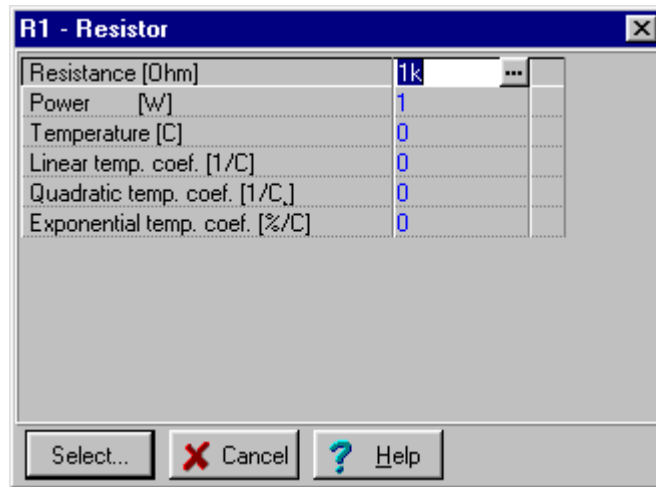


Now that you have established the target details, you must identify the control object(s) and their parameter(s) to be determined by optimization. To select a component, click on the Analysis.Control

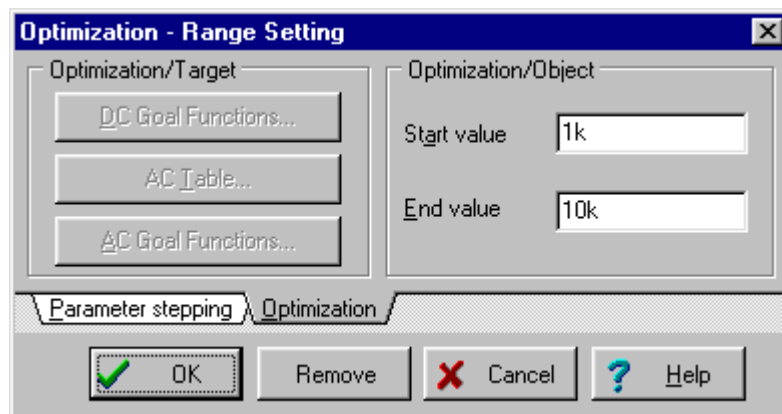
Object menu item or on the Select Control Object speed button . A special cursor  will appear in the schematic editor window, waiting for you to identify the component to which the

optimized parameter belongs. If you want TINA to optimize using more than one component, repeat the procedure for the other components.

When you point at the component and click, TINA will display all of its associated parameters in a dialog window.



Highlight the parameter to be determined by the optimization process, press the Select button, and enter the Start and End values for the variation of this component's value.



If the desired parameter belongs to a catalog component, or if it is an excitation attribute, you must first select the semiconductor or waveform type so that you can access the appropriate data sheet.

The selected parameter will be varied in the range between the Start and End values while TINA searches for the optimum result at the target. If no optimum is found, then the range will have to be enlarged. If there is more than one optimum case in the range, TINA will only find one of them. If you suspect that this is the case, try using multiple, narrower ranges, each of which has one or fewer optimums.



You can use the Remove button to deselect a previously assigned control object or optimization target.

Optimization is useful not only in the design of electronic circuits, but in teaching, where you can use it to construct examples and problems. Let us see some examples.

Two examples



Example 1: Differential Amplifier (DC Optimization)

This example illustrates how to use the DC Optimization to set the operating point of a differential amplifier.

- Specification:** First load the file named DIFFAMP.TSC from TINA's EXAMPLES/OPTI subdirectory and run Analysis/DC Analysis/Calculate Nodal Voltages to see the original VF1, VF2 collector voltages (3.78V). The goal of the optimization is to change the values of R1A, R1B in order to find those values which meet the specification—2.5V—for both collector voltages.
- Set up the optimization:** Before you start the optimization, you have to select those meters which will be monitored during the iteration and which represent the goal of the optimization. In this example, the meters are VF1 and VF2, and the goal establish 2.5V at the collector of the transistors. You also must select those parameters which will be determined by the optimization. In this example, they are the resistances of R1A and R1B.
 - Target:** Use the command Analysis.Optimization Target or the Select Optimization Target speed button  and click on the first voltage pin (VF1). A dialog box appears. When you press the DC Goal Functions... button another dialog appears which allows you to set the goal function for this meter. Select the Value option and set the value to 2.5. (Note that you mustn't use units here.) Then press OK twice. Repeat this procedure for the other voltage pin, VF2.
 - Control object:** Use the command Analysis.Control Object or the Select Control Object speed button  and click on the first control object (R1A). In the property dialog that appears, click on the first line (the resistance parameter). Press the Select button and another dialog appears which allows you to define the Start and End values for the variation of this parameter. Enter 1k and 100k respectively and press OK. Then repeat this procedure for the other resistor, R1B.
- Start the optimization:** From the Analysis menu choose Optimization, then DC Optimization... Accept the settings in the dialog that appears and press OK. Now TINA will find new R1A, R1B values to achieve the desired 2.5V collector voltages. After a short iteration the result—the new resistor values—will be shown in a dialog.
- Checking the result:** TINA will retain the optimized resistor values. Run Analysis/DC Analysis/Calculate Nodal Voltages to check the new collector voltages with these values. Note that they are now exactly 2.5V as we previously defined in the specification, verifying that the optimization was successful.

Example 2: Band pass Filter (AC Optimization)

This example illustrates how to use the AC Optimization to obtain a band pass filter which meets transfer function goals.

1. **Specification:** First load the file called BANDPASS.TSC from TINA's EXAMPLES/OPTI subdirectory and run Analysis/AC Analysis/AC Transfer Characteristic... to see the original transfer function. The diagram shows that the maximum of the curve is at about 10kHz, while the peak amplitude of the curve is -10dB. The goal of the optimization is to set the peak amplitude to -7dB and the frequency at which the curve reaches its maximum to 14kHz.
2. **Set up the optimization:** To do an optimization, you have to select those meters which will be monitored during the iteration and which represent the goal of the optimization. In this example, this is the meter "Out." You must also select those parameters which will be varied during the optimization. In this example, these parameters are the resistance of R1 and the capacitance of C1.
 - a) **Target:** Use the command Analysis.Optimization Target or the Select Optimization Target speed button  and click on the voltmeter (Out). A dialog box appears. When you press the AC Goal Functions... button another dialog appears which allows you to set the goal function for this meter. The specification contains two conditions. The first defines the new maximum point of the curve. To set this value, click on the Center Frequency tab, then enter 14k in the Target center frequency edit field. The other condition is the new maximum, -7dB. Select the Maximum tab and enter -7 in the Maximum edit field. The last step is to activate the current goal functions which are the combination of two functions. On the left side of the dialog, select both the Center Frequency and the Maximum function by clicking on the checkbox next to them. Finally press OK twice.
 - b) **Control object:** Use the command Analysis.Control Object or the Select Control Object speed button  and click on the first control object (R1). In the resistor property dialog, click on the first line, the resistance parameter. Press the Select button to reach another dialog where you can define the Start and End values for the variation of this parameter. Enter 1k and 10k respectively and press OK. Repeat this procedure for the capacitor (C1), and specify 1n and 10n for the Start and End values respectively.
3. **Start the optimization:** From the Analysis menu, choose Optimization then AC Optimization (Transfer)... Accept the settings in the dialog and press OK. Now TINA will find new R1 and C1 values that will satisfy the specification for the transfer function. After a short iteration the result—the new resistor and capacitor values—will be shown in a dialog box.
4. **Checking the result:** Run Analysis/AC Analysis/AC Transfer Characteristic... to check the new transfer function. The maximum now occurs at 14kHz, with a level at the maximum of -7dB. The optimization was successful

Fourier Series and Fourier Spectrum

Fourier analysis – Introduction

In Tina there are many advanced analysis features that provide powerful professional tools to study and evaluate electronic circuits both numerically and graphically. One of them is called spectral or Fourier analysis. In this booklet we will demonstrate Tina's implementation of Fourier analysis which covers two different methods: Fourier series and Fourier spectrum. We will especially emphasize applications that are not described in detail in the manual of Tina. In the following we will shortly summarize the theory behind Fourier analysis, and then present examples on the topic. All the examples are available on floppy disk, as well.

In Tina Fourier analysis must be always performed in conjunction with the transient analysis. This means, that first you always have to carry out a transient analysis on the circuit. When the diagram window appears you have to select the curve you want to process using the cursor and from the *Process* menu choose *Fourier series* or *Fourier spectrum*. In the first part of this booklet we will investigate Fourier series and later on Fourier spectrum.

Fourier series

One type of spectral analysis is called Fourier or harmonic decomposition. It's based on the theory, that any periodic function can be expressed as the sum, or series of sinusoidal functions. If a periodic function is expressed this way, each component in the series must be periodic over the same interval as the original function. The components are integer multiples or harmonics of the original function's fundamental (base) frequency. This decomposition is described by the following formulas:

$$f(t) = \sum_{k=0}^{\infty} (A_k \cos k\omega t + B_k \sin k\omega t), \text{ or in complex form } f(t) = \sum_{k=-\infty}^{\infty} C_k e^{jk\omega t}, \text{ where } A_k, B_k, \text{ and } C_k \text{ are the Fourier coefficients.}$$

Let us see this decomposition in case of some practical waveforms:

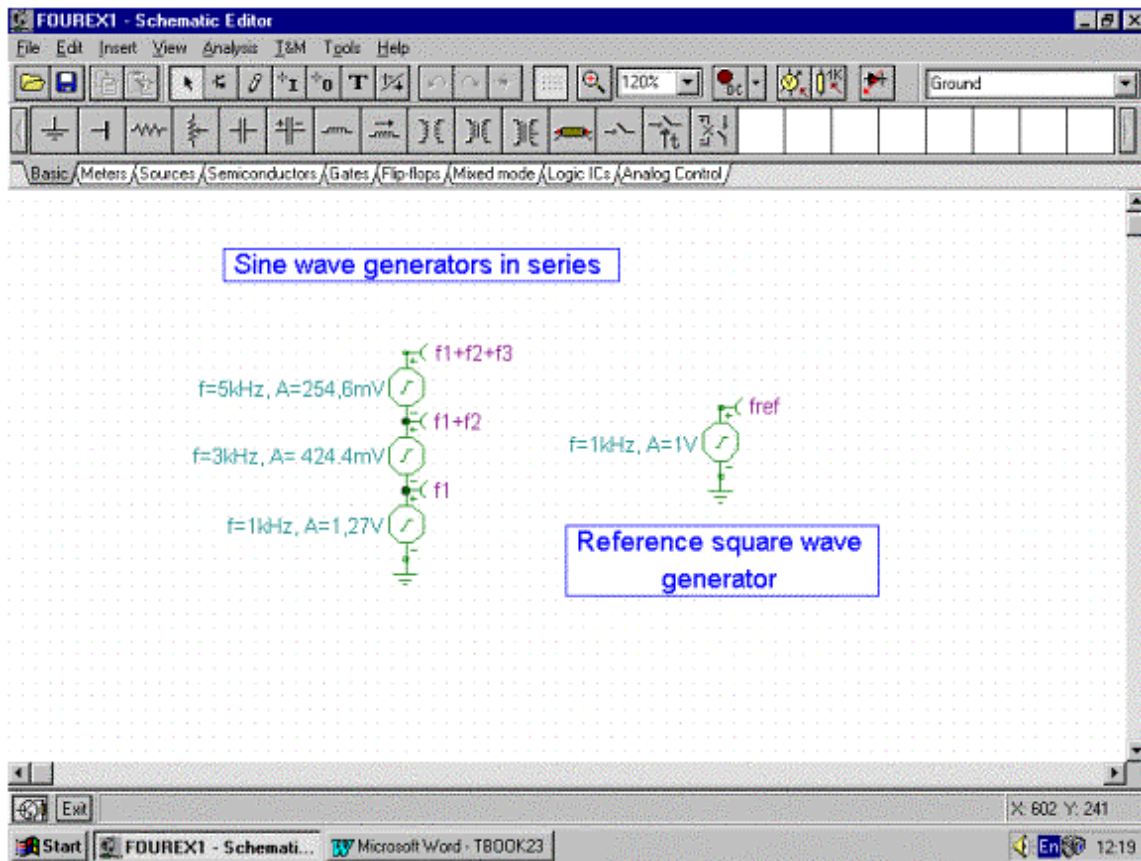
Triangle wave
Square wave
$f(t) = \frac{8}{\pi^2} \left[\sin \omega t - \frac{1}{3^2} \sin 3\omega t + \frac{1}{5^2} \sin 5\omega t - \dots \right]$
$f(t) = \frac{4}{\pi} \left[\sin \omega t + \frac{1}{3} \sin 3\omega t + \frac{1}{5} \sin 5\omega t + \dots \right]$

Let us demonstrate the above theorem with Tina in case of a square wave of $f=1\text{kHz}$ frequency and 1V amplitude up to three sine waves. The easiest way is to put together a simple circuit (fourx1.sch). The circuit consists of three sine wave generators connected in series and the reference square wave generator.

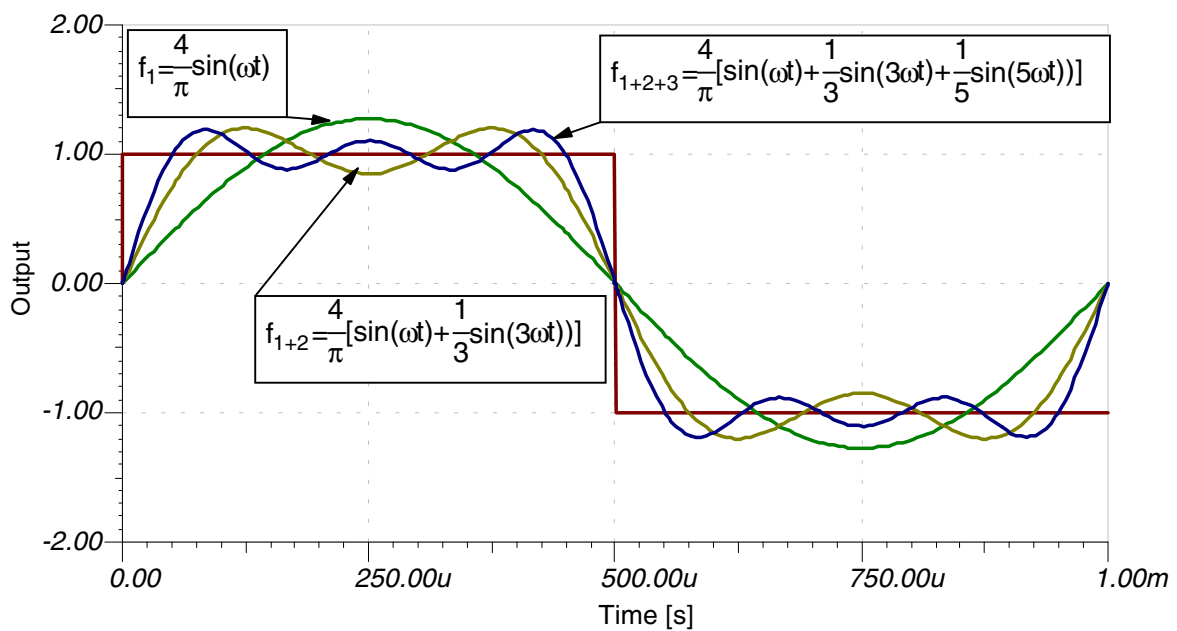
The frequencies of the sine wave generators are $f_1 = 1\text{kHz}$, $f_2 = 3f_1 = 3\text{kHz}$ and

$f_3 = 5f_1 = 5\text{kHz}$, however the amplitudes are $A_1 = \frac{4}{\pi} = 1.2732$, $A_2 = \frac{1}{3} A_1 = 0.4244$ and

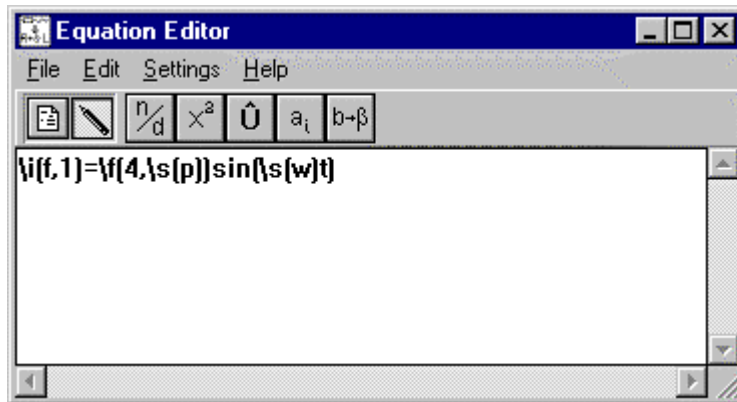
$A_3 = \frac{1}{5} A_1 = 0.2546$ according to the above formula.






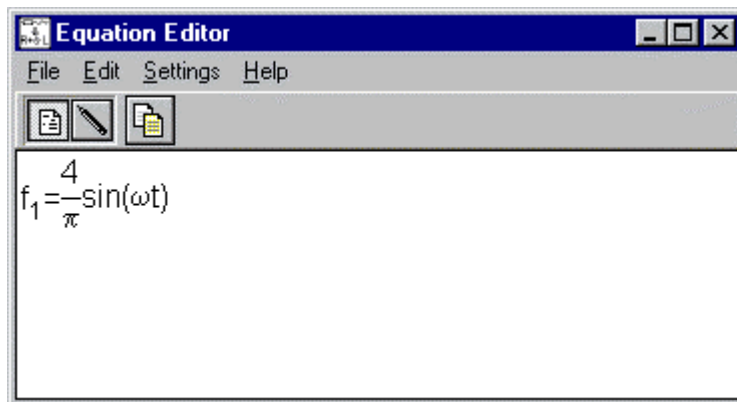
The result of the transient analysis can be seen on the following figure.



Note that the above figure was entirely made by Tina using its Equation Editor in the Diagram Window. For example the $f_1 = \frac{4}{\pi} \sin(\omega t)$ is described in the following way in the Equation Editor's text mode: $\backslash i(f,1)=\backslash f(4,\backslash s(p))\sin(\backslash s(w)t)$ as it can be seen on the next figure.



If you press the View button  you can preview the formula in the Equation Editor. Press the Edit button  to return into the text mode if you want to make further changes. Finally press the Copy button  to place the formula in the Diagram Window.




Obviously if you connected more than 3 generators in series, you would have achieved a better match. There is an easy way to demonstrate this statement. Using Tina's Interpreter you can easily write a function that calculates the sum of any number of sine waves (fourx1.ipr).

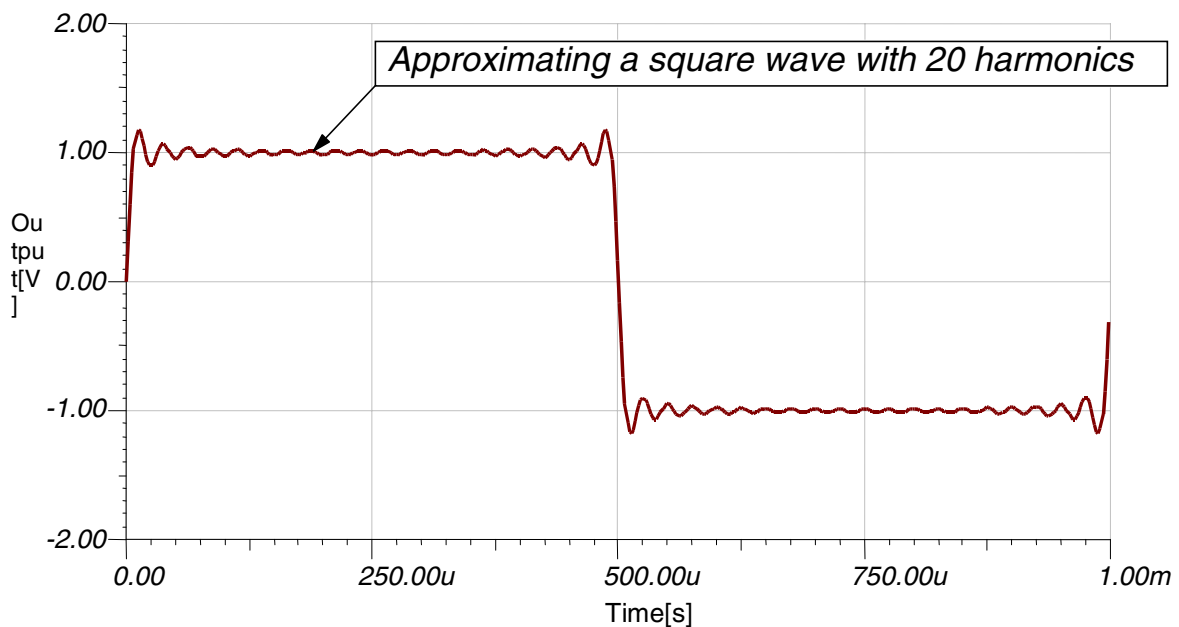
```

Function Square(t, n);
{Sum of n sine waves, n: number of sine waves, t: time}
Begin
  f := 1k;
  w := 2 * pi * f;
  x := 0;
  For i := 0 To n - 1 Do
    x := x + 1 / (2 * i + 1) * sin((2 * i + 1) * w * t);
  Square := 4 / pi * x;
End;

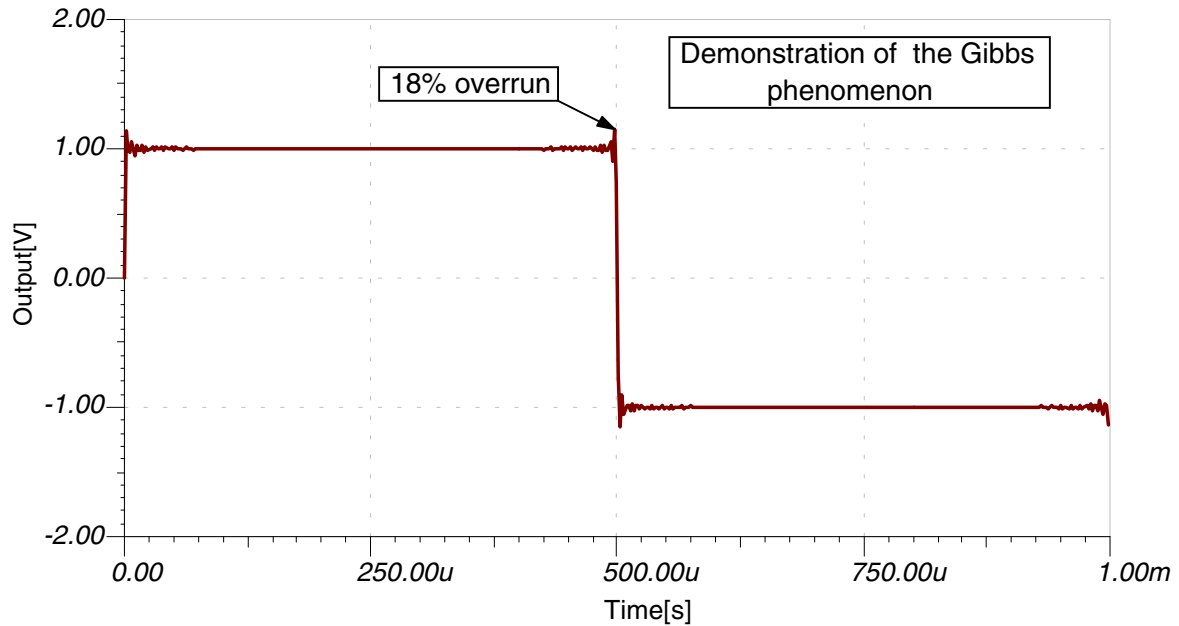
{Sum of 20 sine waves}
Draw(Square(Time, 20), Square);

```

To run the above example invoke the Interpreter from the Tools menu, type in the above text or read in fourx1.ipr by selecting the Open command from the File menu. If you press the Run  button after a short calculation the following function appears:



The more sine waves you would sum up the better match you would get. Note however the small peaks at the edges of the square wave. This is called Gibbs phenomenon. According to the mathematical theory the Fourier series can not approximate properly non-continuous functions in the point of discontinuity. In case of a square wave there will always be 18% overrun at the edges at any number of approximating sine waves. Let us demonstrate this with 100 harmonics; simple change 20 to 100 in the Draw command of the above example (fourx1.ipr).

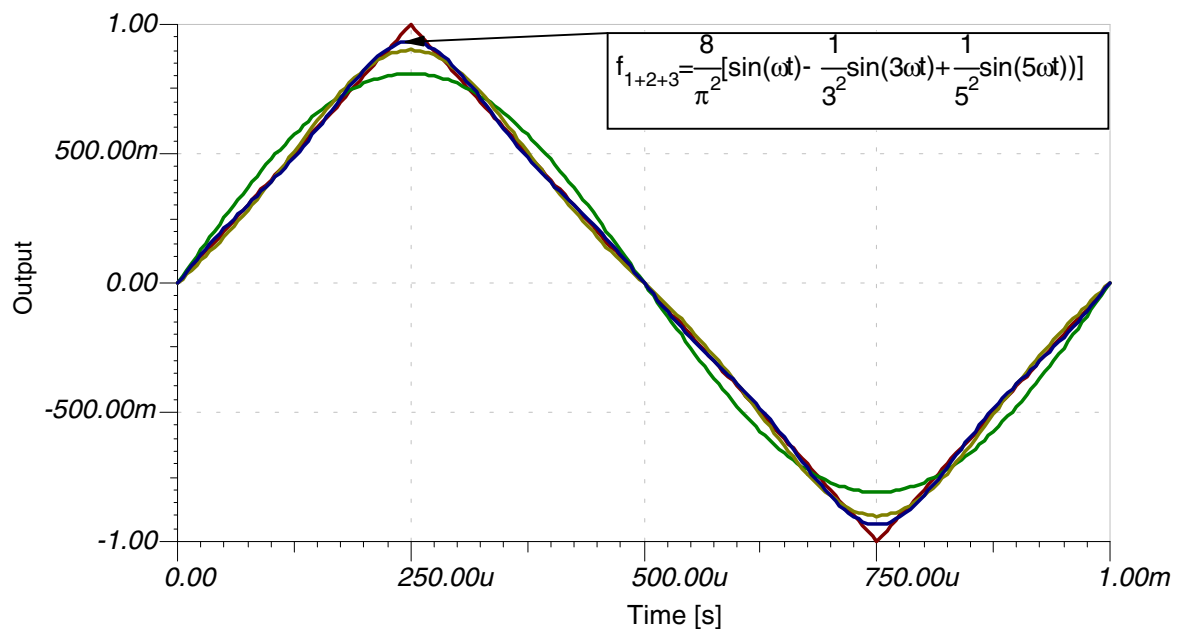


However there are methods which improve the convergence of the Fourier series near the discontinuity. The essence of these techniques is to modify the Fourier coefficients by multiplying each coefficient by a weight factor. This method is also called window function technique. Two Hungarian mathematicians - Fejér, and Lánzos - invented such window functions that are commonly used in practice.

In practice however there are no discontinuous waveforms though the rise/fall time could be very short in comparison to the period time. This means, that the Fourier series will always converge. Let us demonstrate this with the approximation of the triangular waveform with just 3 harmonics using the above circuit.

Change the amplitude of the sinusoidal generators to $A_1 = \frac{8}{\pi^2} = 0.8106$, $A_2 = -\frac{1}{3^2} A_1 = 0.0901$

and $A_3 = \frac{1}{5^2} A_1 = 0.0324$. Change the waveform of the reference generator to triangle wave (fourex12.sch).



In practice of course you use the Fourier series option of TINA in opposite direction. You have a periodic function and you want to know the Fourier components. To get them in Tina first you have to run a transient analysis and then in the appearing Diagram window select a curve. Then in the diagram window you have to select Fourier series from the Process menu. After selecting this menu item Tina performs a harmonic decomposition, calculating the Fourier coefficients for the sinusoidal components of any voltage or current. After you select Fourier series a dialog box appears, where you have to set the sampling start time, the base frequency, the number of samples, the number of harmonics, and the format.

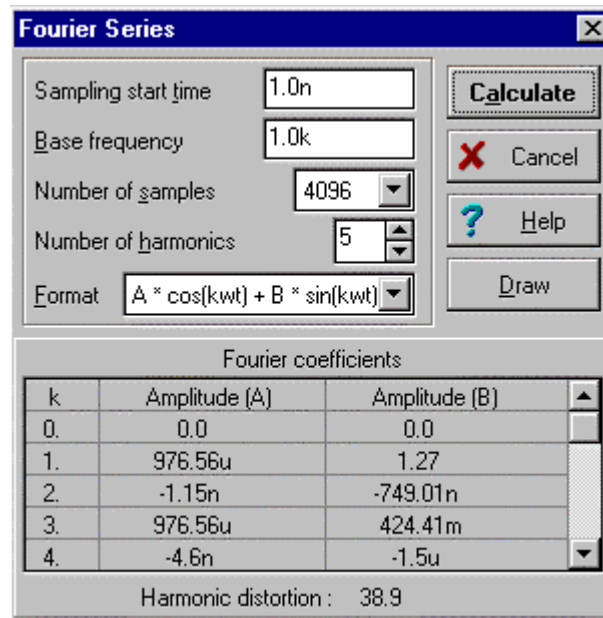
Example: Let us determine the Fourier decomposition of the previous square wave ($f=1\text{kHz}$, $A=1\text{V}$). We will compare the theoretical decomposition with the result calculated by TINA.

The Fourier series of a square wave function is $f(t) = \frac{4}{\pi} \left[\sin \omega t + \frac{1}{3} \sin 3\omega t + \frac{1}{5} \sin 5\omega t + \dots \right]$.

Because square wave function is an odd function $f(-x) = -f(x)$, all the coefficients $A_k = 0$, and $A_0 = 0$, too. Comparing the above function with the definition of the Fourier series it's quite clear that:

$B_1 = \frac{4}{\pi} = 1.27$, $B_2 = 0$, $B_3 = \frac{1}{3} B_1 = 0.42441$, $B_4 = 0$, $B_5 = \frac{1}{5} B_1 = 0.25464$. In Tina we get approximately the same result. However due to numerical inaccuracy we don't get exactly 0 for the theoretically 0 coefficients. If you examine these values more carefully you'll see, that they are not significant because they are $10^{-6} - 10^{-9}$ times less than the fundamental value.

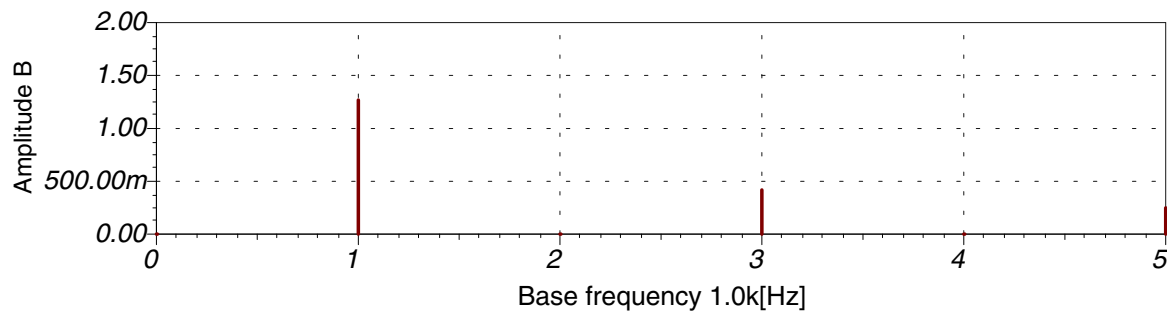
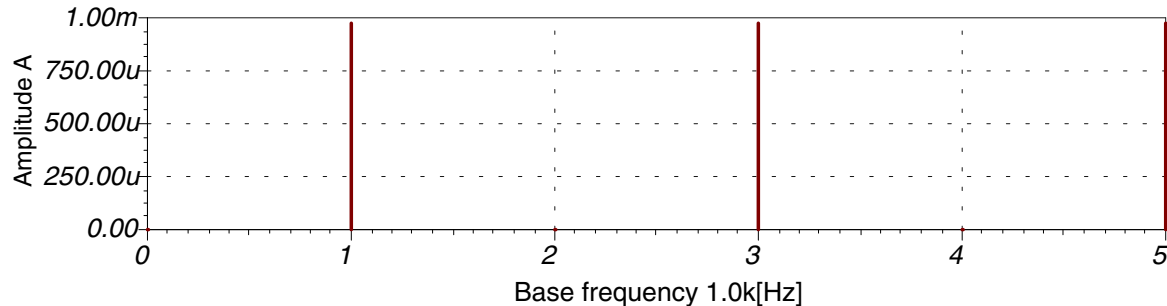
Load this example (fourex2.sch). Run transient analysis then select the appearing square wave and choose Fourier series from the *Process* menu. In the dialog box change the default format from complex representation $C * \exp(k\omega t + \phi)$ to real representation $A * \cos(k\omega t) + B * \sin(k\omega t)$.



If you have set all the parameters, press *Calculate*. At the bottom of the dialog box the Fourier coefficients appear. Tina allows you to use these values in other programs. Every time you press *Calculate* the coefficients are placed not only at the bottom of the dialog box but on the clipboard, as well. You can import this table of Fourier components in other programs, for example Word or Excel to process.

Harmonics	Amplitude (A)	Amplitude (B)
0.	0	0
1.	976.56u	1.27
2.	-1.15n	-749.01n
3.	976.56u	424.41m
4.	-4.6n	-1.5u
5.	976.56u	254.64m

It's also possible to evaluate the above result graphically. You only need to press *Draw*, then Tina generates you a diagram with the Fourier coefficients.



You can also get complex Fourier components with Tina. To do this select the complex representation in the Fourier series dialog box and press *Calculate* again. On the following three figures you'll see the harmonic decomposition of the same sine wave ($f=1\text{kHz}$, $A=1\text{V}$) but this time using the complex representation. On the first figure the dialog box can be seen, on the next the Fourier coefficients as they are placed on the clipboard, and on the last the graphical representation of the result.

Fourier Series ✕

Sampling start time: Calculate

Base frequency: ✖ Cancel

Number of samples: ? Help

Number of harmonics: Draw

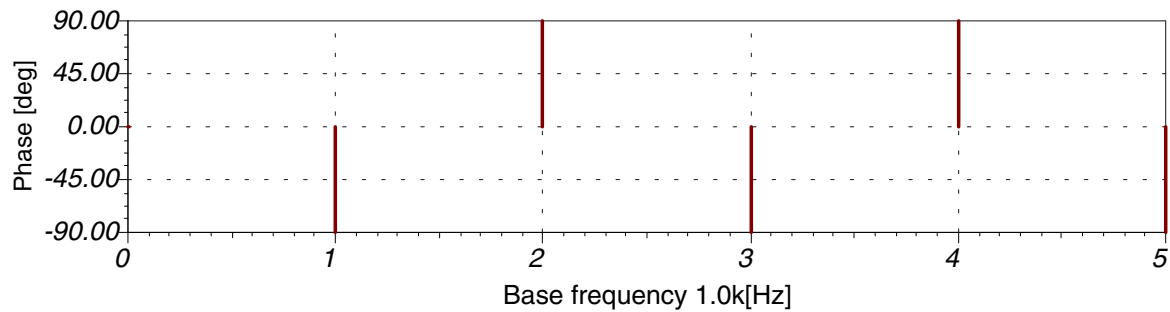
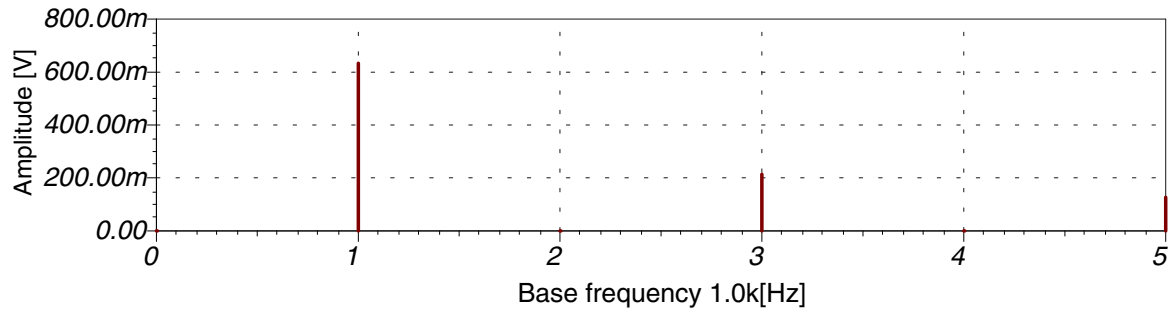
Format: ✕

Fourier coefficients

k	Amplitude (A)	Phase (ø)
0.	0.0	0.0
1.	636.62m	-89.96
2.	374.51n	90.09
3.	212.21m	-89.87
4.	749.01n	90.18

Harmonic distortion : 38.9

Harmonics	Amplitude (A)	Phase (θ)
0.	0	0
1.	636.62m	-89.96
2.	374.51n	90.09
3.	212.21m	-89.87
4.	749.01n	90.18
5.	127.32m	-89.78



To compare this result with the previous one we have to perform a simple mathematical transformation. Using the Euler-formula: $\frac{A_k + jB_k}{2} = C_{-k}$, and $\frac{A_k - jB_k}{2} = C_k$ or vice versa

$C_k + C_{-k} = A_k$, or $C_{-k} - C_k = B_k$. From this we get, that $C_1 = \frac{A_1}{2} = 0.63662$, $C_2 = 0$,

$C_3 = \frac{A_3}{2} = 0.21221$, $C_4 = 0$, $C_5 = \frac{A_5}{2} = 0.12732$. Again it is important to mention, that due to

numerical inaccuracy we don't get exactly 0 for the theoretically 0 coefficients. It's especially true in case of the phase values. As harmonic magnitudes become small it becomes difficult to determine phase. Again, this is a numerical problem only.

Fourier spectrum

Tina provides another type of spectral analysis called Fourier transform. Fourier transform can also be thought of as an extension of the Fourier series. This means the Fourier transform converts a function of time to a function of frequency, and vice versa. The physical interpretation of Fourier transform is the conversion of a time-domain signal to the AC steady-state frequency spectrum.

$$F(j\omega) \equiv \mathcal{F} f(t) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt$$

However the inverse Fourier transform converts the AC steady-state signal into the time-domain.

$$f(t) \equiv \mathcal{F}^{-1} F(j\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(j\omega)e^{j\omega t} d\omega$$

The Fourier transform in Tina is a discrete Fourier transform (DFT), where the Fourier integral has been replaced by a nearly equivalent summation formula applied to evenly spaced samples of the signal. To speed up the process the transform is accomplished by a special technique, called fast Fourier transform (FFT). The FFT gets its data from a set of samples with the appropriate number of data points.

Sometimes it's useful to define the real spectrums (amplitude-densities):

$$F(j\omega) = \frac{A(\omega) - jB(\omega)}{2}, \text{ where}$$

$$A(\omega) = 2 \operatorname{Re} F(j\omega) \text{ and } B(\omega) = -2 \operatorname{Im} F(j\omega)$$

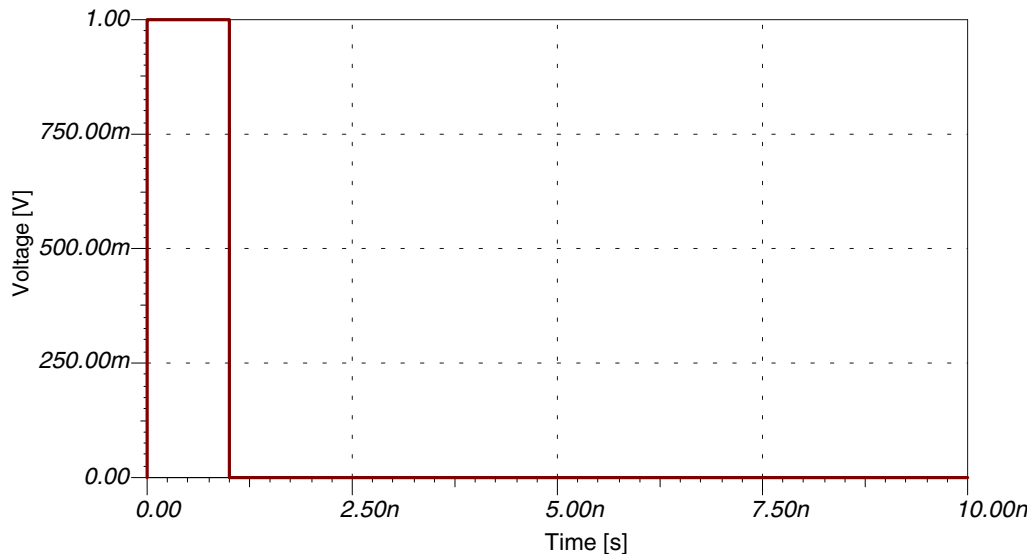
Note, that Tina calculates $F(j\omega)$ and displays in the $0 < \omega < \infty$ region only. For the $-\infty < \omega < 0$ region applies $F(j\omega) = F(j\omega)^*$, therefore the amplitude is the same and the phase has opposite sign.

$A(\omega)$, $B(\omega)$ can also be calculated directly from the time function.

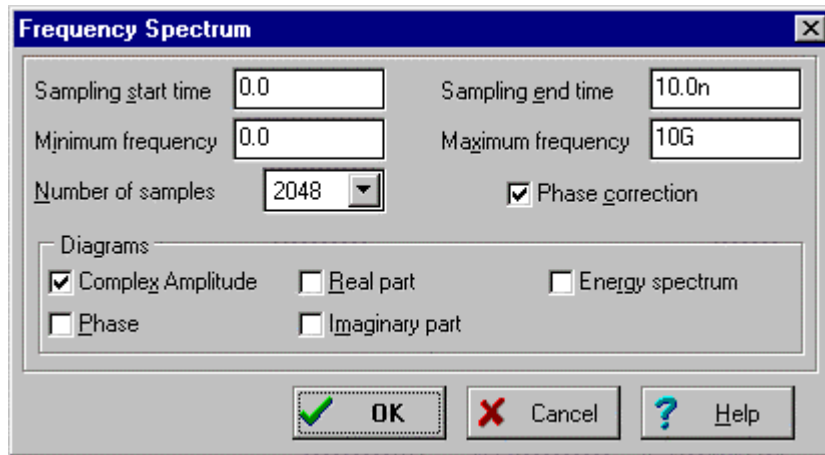
$$A(\omega) = 2 \int_{-\infty}^{\infty} f(t) \cos \omega t dt \text{ and } B(\omega) = 2 \int_{-\infty}^{\infty} f(t) \sin \omega t dt$$

$A(\omega) = A(-\omega)$, $B(-\omega) = -B(\omega)$ thus $A(\omega)$ is an even, $B(\omega)$ is an odd function.

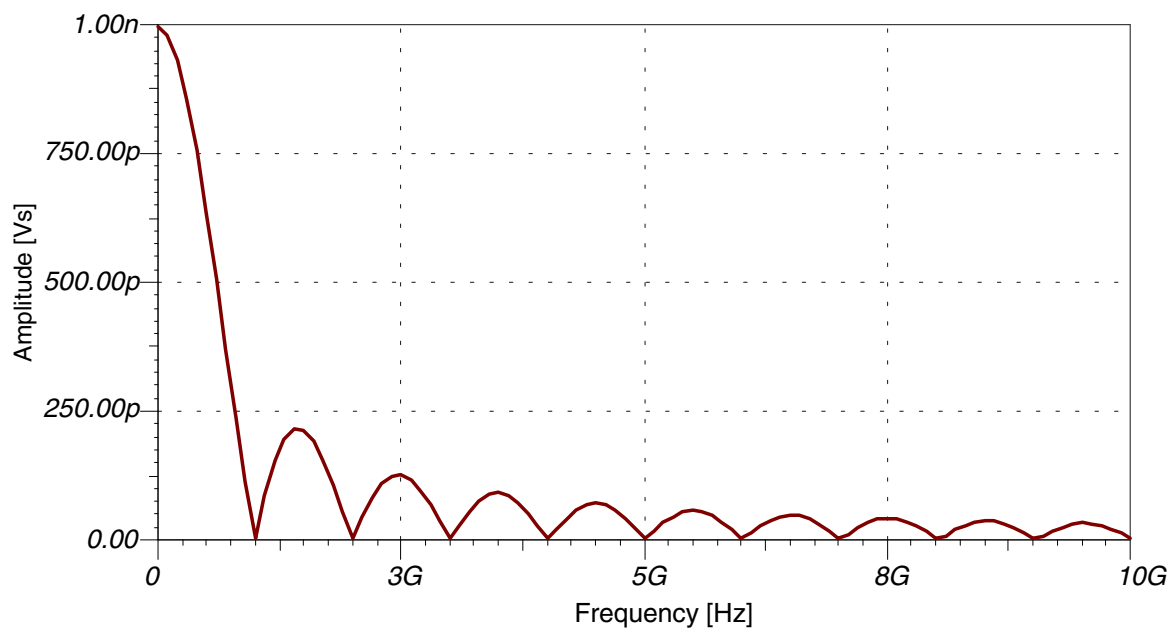
Example: Let's determine the complex spectrum of a pulse (amplitude 1, width $T=1n$; fourex3.sch) and compare it with the theoretical result. First carry out a transient analysis then select the pulse wave and choose Fourier spectrum from the *Process* menu.



After you select Fourier spectrum a dialog box appears, where you have to set the sampling start and end time, the minimum and maximum frequency and the number of samples. You can also choose the diagrams you want to display.



If you press OK Tina performs discrete Fourier transform. The complex amplitude spectrum can be seen on the following figure:



This function is described by the well-known formula:


$$F(j\omega) = \int_0^T 1 * e^{-j\omega t} dt = \frac{1 - e^{-j\omega T}}{j\omega} = e^{-j\omega T/2} \frac{e^{j\omega T/2} - e^{-j\omega T/2}}{j\omega} = \frac{2}{\omega} \sin \frac{\omega T}{2} e^{-j\omega T/2}, \text{ therefore the}$$

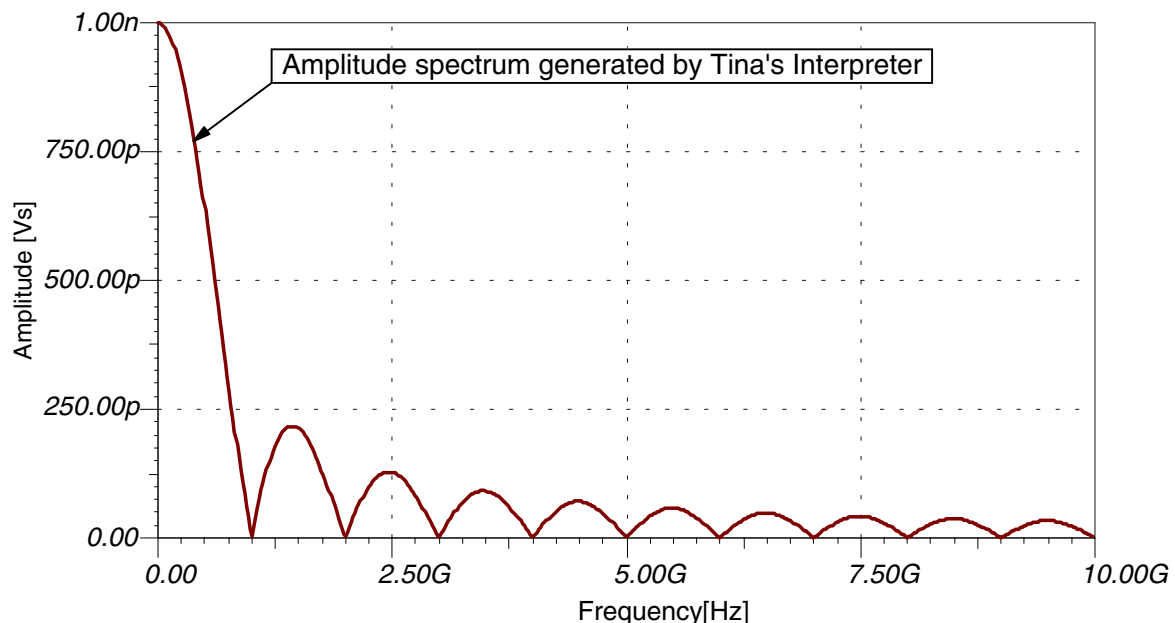
$$\text{amplitude spectrum is: } |F(j\omega)| = \frac{2}{\omega} \left| \sin \frac{\omega T}{2} \right|, \quad 0 < \omega < \infty$$

To check our previous result now we will recalculate the amplitude spectrum using the above formula in Tina's Interpreter (fourx2.ipr). The following function calculates the amplitude spectrum.

```
Function PulseAmplSpe(f);
Begin
  w := 2 * pi * f;
  T := 1n;
  PulseAmplSpe := 2 / w * Abs(Sin(w * T / 2));
End;

Draw(PulseAmplSpe(Frequency), AmplSpe);
```

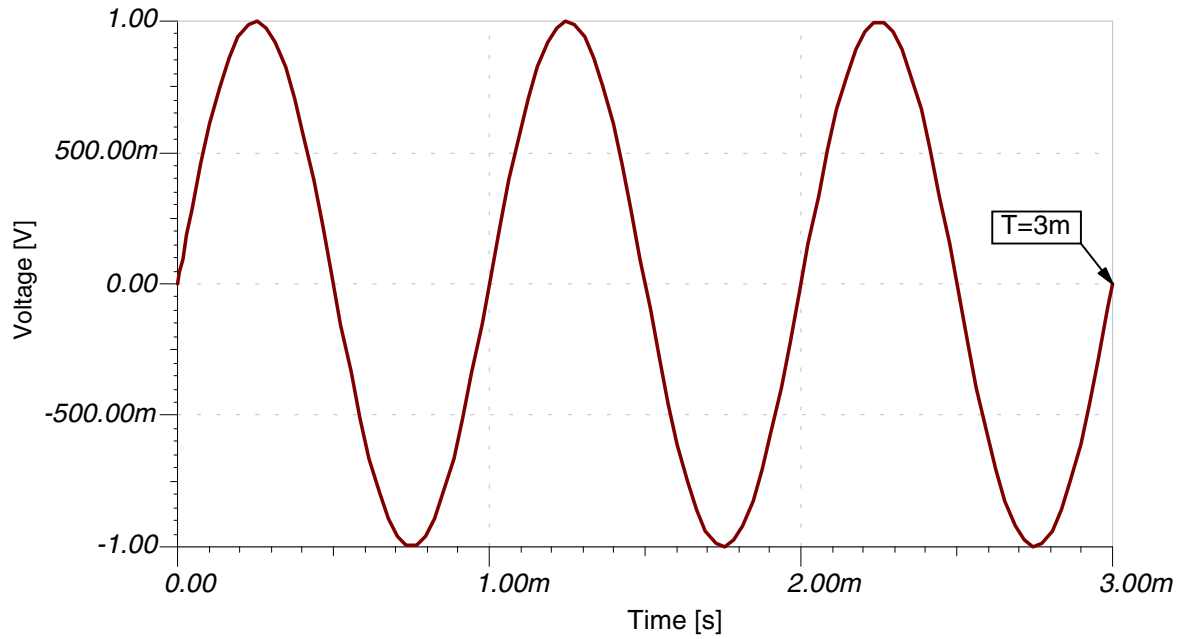
To try out the above example invoke the Interpreter from the Tools menu, type in the above text or read in fourx2.ipr by selecting the Open command from the File menu. If you press the Run  button after a short time the following function appears:



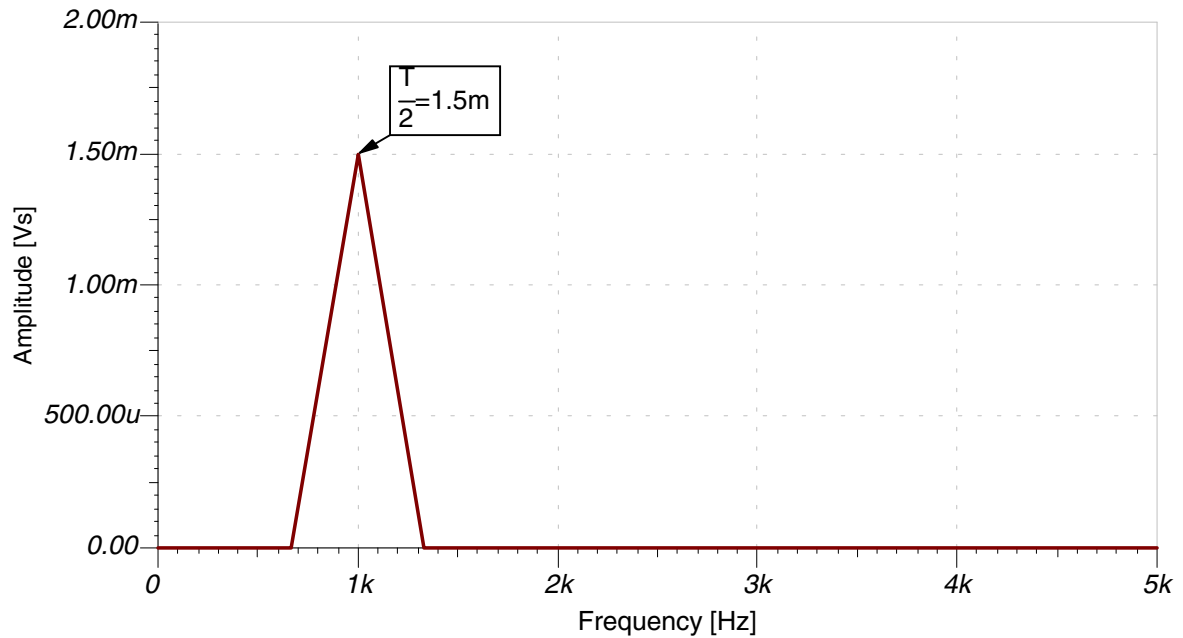
You can easily check, that the above spectrum provided by the closed formula and the spectrum provided by Tina's Fourier Spectrum command are exactly the same. It's rather easy to compare these curves using the clipboard. First select the curve which was generated by the Interpreter, then copy this curve onto the clipboard then in the diagram window go back to the previous page and paste the curve back. The curves will be identical, so we do not show here both.

Example: Transform of a sine wave (fourex4.sch)

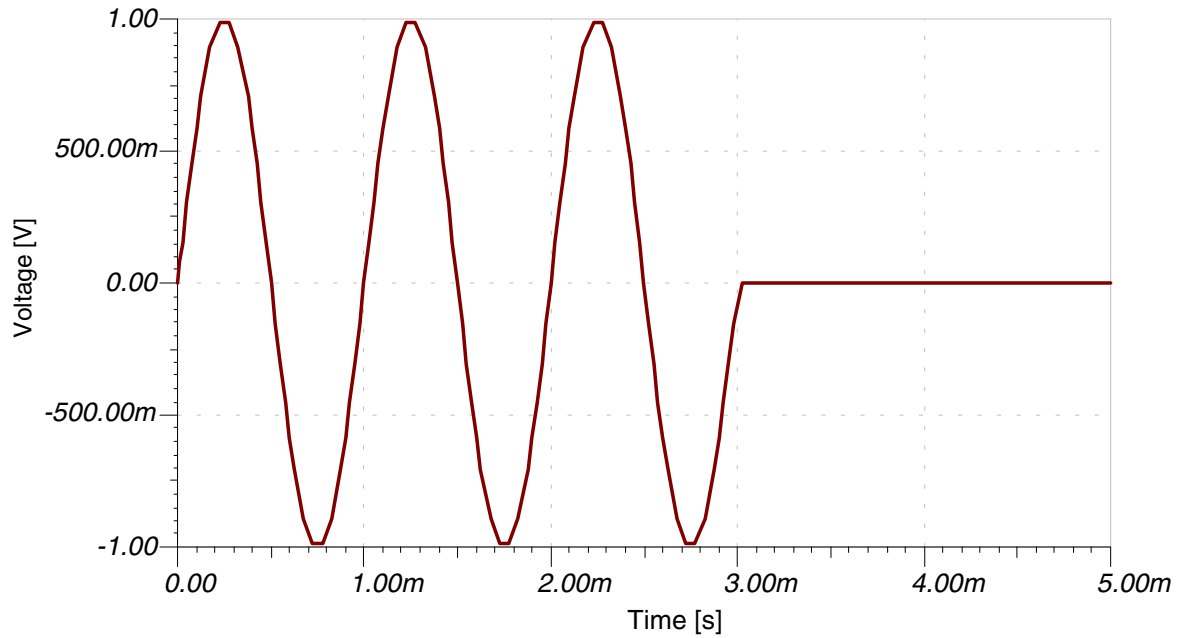
The Fourier transform of a sine wave results in a Dirac-delta function, that is an infinitely high spike of zero width at the frequency of a sine wave. We can not reproduce that with DFT. We can actually examine sine waves of finite length. The longer the sine wave is, the taller and narrower the spectrum becomes. Let us first examine a 3ms long sine wave with 1kHz frequency.



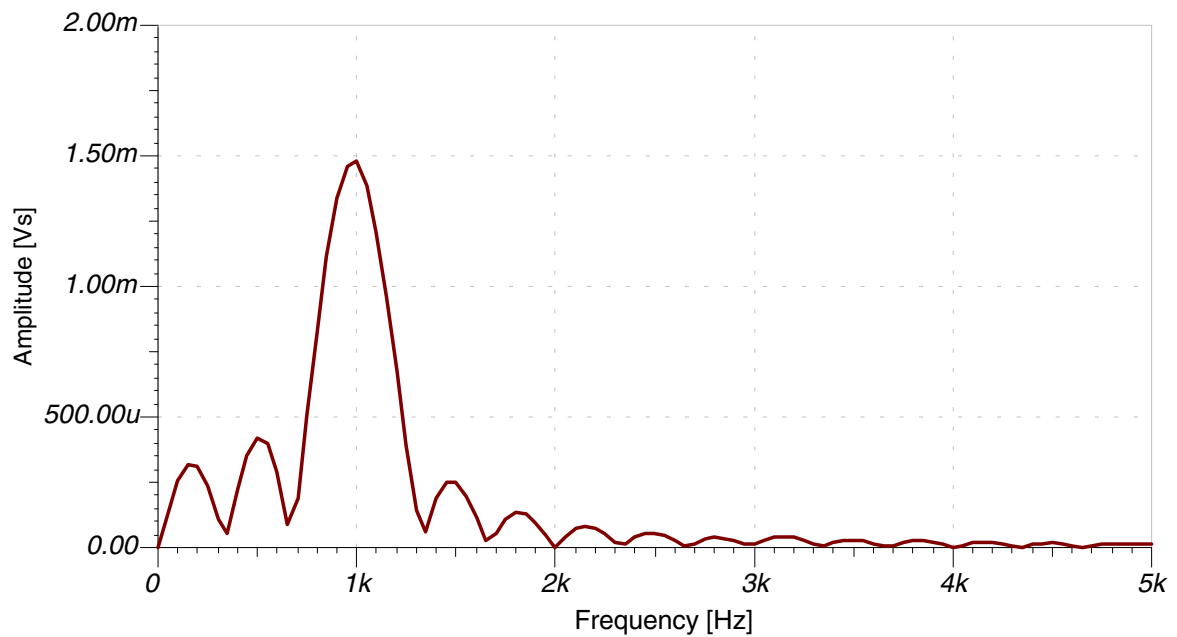
If you run Fourier spectrum on this curve you will get the following result.



Although the peak value of this spectrum is correct the shape of the spectrum is quite rough. To get a better spectrum change the simulation time to 21ms and carry out the analysis again. At FFT the frequency resolution is inversely proportional to the duration of the signal.



Note that the actual simulation time in this example was 21ms to get a better spectrum. (Not shown on the figure.)



Let us check this result using the theory of Fourier transform:


$$F\{d(t) \sin \omega_0 t\}, \text{ where } d(t) = \varepsilon(t) - \varepsilon(t-T) = \begin{cases} 0, & \text{if } t < 0 \text{ or } t > T \\ 1, & \text{if } 0 \leq t \leq T \end{cases}$$

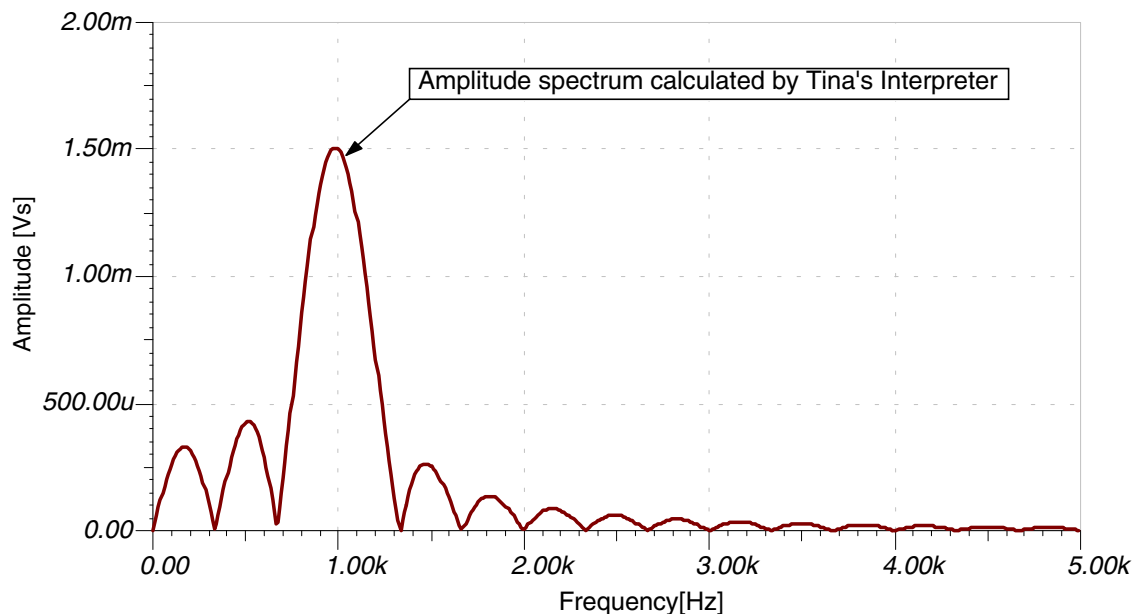
$$F(j\omega) = \int_{-\infty}^{\infty} d_T(t) \sin \omega_0 t e^{-j\omega t} dt = \int_{-\infty}^{\infty} d_T(t) \frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j} e^{-j\omega t} dt = \frac{1}{2j} \left\{ \int_0^T e^{-j(\omega-\omega_0)t} dt - \int_0^T e^{-j(\omega+\omega_0)t} dt \right\} =$$

$$e^{-j(\omega-\omega_0)\frac{T}{2}} \frac{\sin(\omega-\omega_0)\frac{T}{2}}{j(\omega-\omega_0)} - e^{-j(\omega+\omega_0)\frac{T}{2}} \frac{\sin(\omega+\omega_0)\frac{T}{2}}{j(\omega+\omega_0)}$$

Note, that the main amplitude of the spectrum at $\omega = \omega_0$ and at $\omega = -\omega_0$ is $T/2$. Let us draw this function with Tina's Interpreter (fourx3.ipr). In this example we will use a simplified expression, because this part is significant only regarding the amplitude of the spectrum. The following simplified function calculates the amplitude spectrum.

```
Function SinusAmplSpe(f);
Begin
  w := 2 * pi * f;
  w0 := 2 * pi * 1k;
  T := 3m;
  SinusAmplSpe := Abs(Sin((w - w0) * T / 2) / (w - w0) -
    Sin((w + w0) * T / 2) / (w + w0));
End;
Draw(SinusAmplSpe(Frequency), AmplSpe)
```

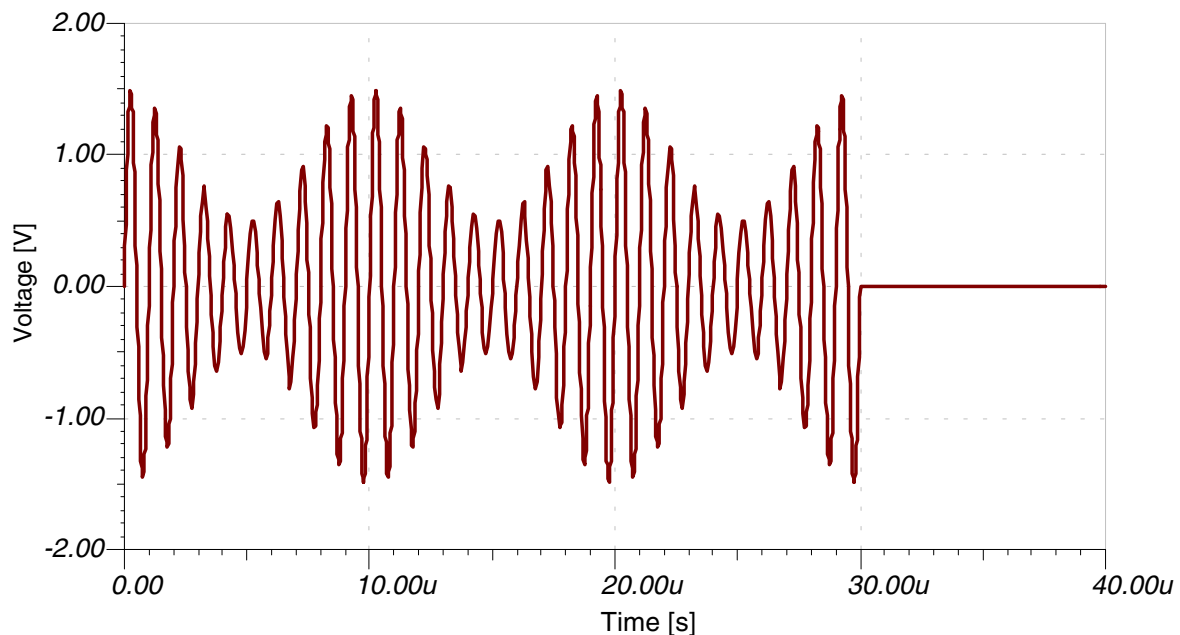
To try out the above example invoke the Interpreter from the Tools menu, type in the above text or read in fourx3.ipr by selecting the Open command from the File menu. If you press the Run  button after a short time the following function appears



You can easily check, that the above spectrum provided by the closed formula and the spectrum provided by Tina's Fourier Spectrum command are exactly the same. It's rather easy to compare these curves using the clipboard. First select the curve which was generated by the Interpreter, then copy this curve onto the clipboard then in the diagram window go back to the previous page and paste the curve back. The curves will be identical, so we do not show here both.

Example: Amplitude modulation (am.sch).

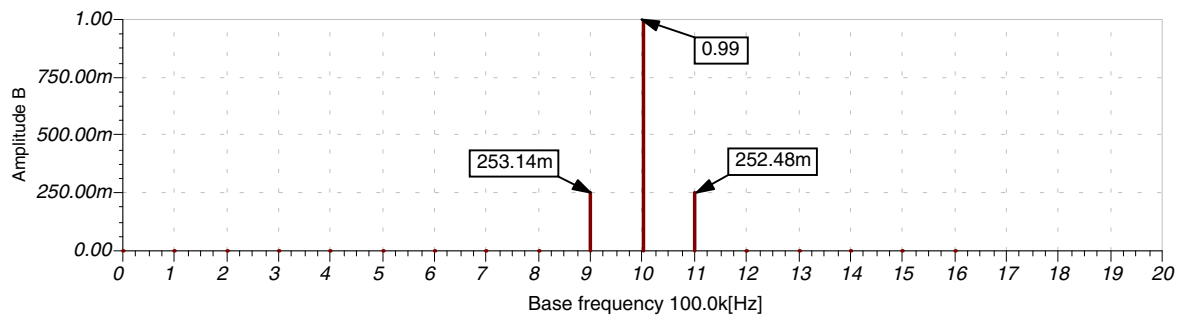
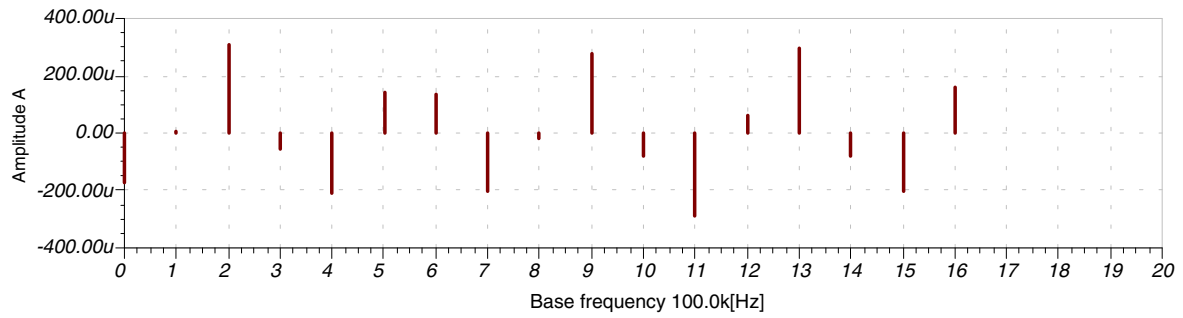
Let us see the spectrum derived by amplitude modulation. In our example the amplitude modulated signal is described by the following formula: $\sin(\omega_0 t) * (1 + m \cos(\omega_m t))$, where $\omega_0 = 1MHz$, $\omega_m = 100kHz$, $m=0.5$.



Let's calculate the Fourier spectrum of the modulated waveform:

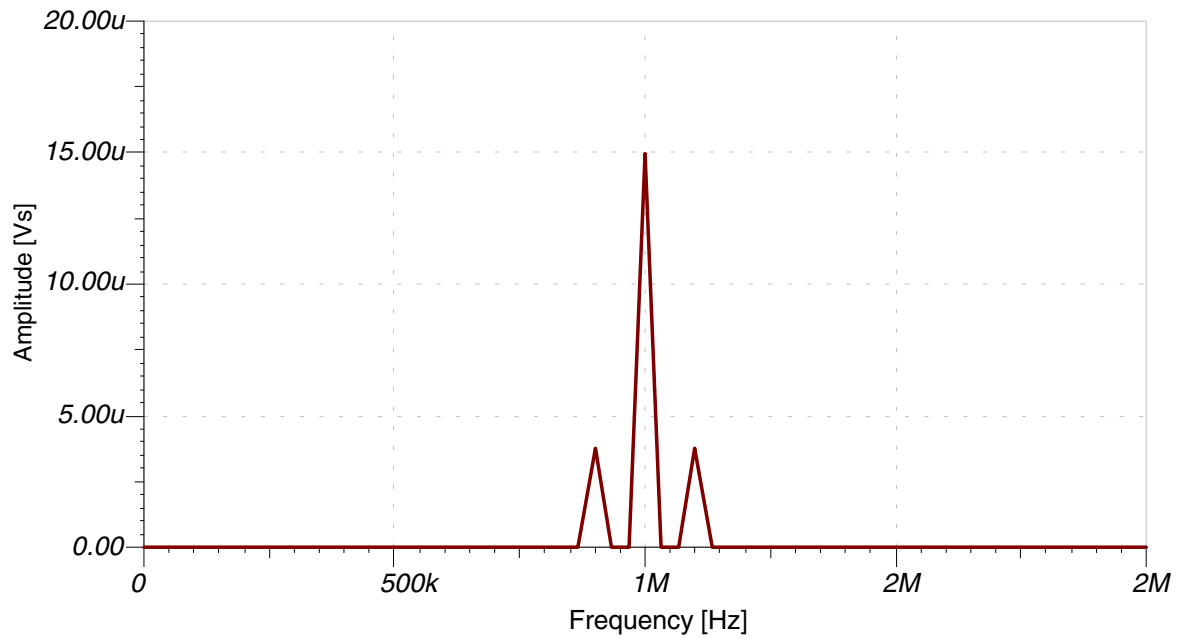
$$\sin(\omega_0 t) * (1 + m \cos(\omega_m t)) = \sin \omega_0 t + \frac{m}{2} \sin(\omega_0 + \omega_m)t + \frac{m}{2} \sin(\omega_0 - \omega_m)t . \text{ In our case:}$$

$\sin(1e6t) * (1 + 0.5 \cos(1e5t)) = \sin 1e6t + 0.25 \sin(1.1e6)t + 0.25 \sin(0.9e6)t$. From this formula it's clear, that the spectrum of the modulated waveform is the sum of the spectrum of three sinusoidal signals. First check our result using Fourier series. Select Fourier Series from the Process menu and set the Format field to $A \cos(k\omega t) + B \sin(k\omega t)$. Then press Calculate.



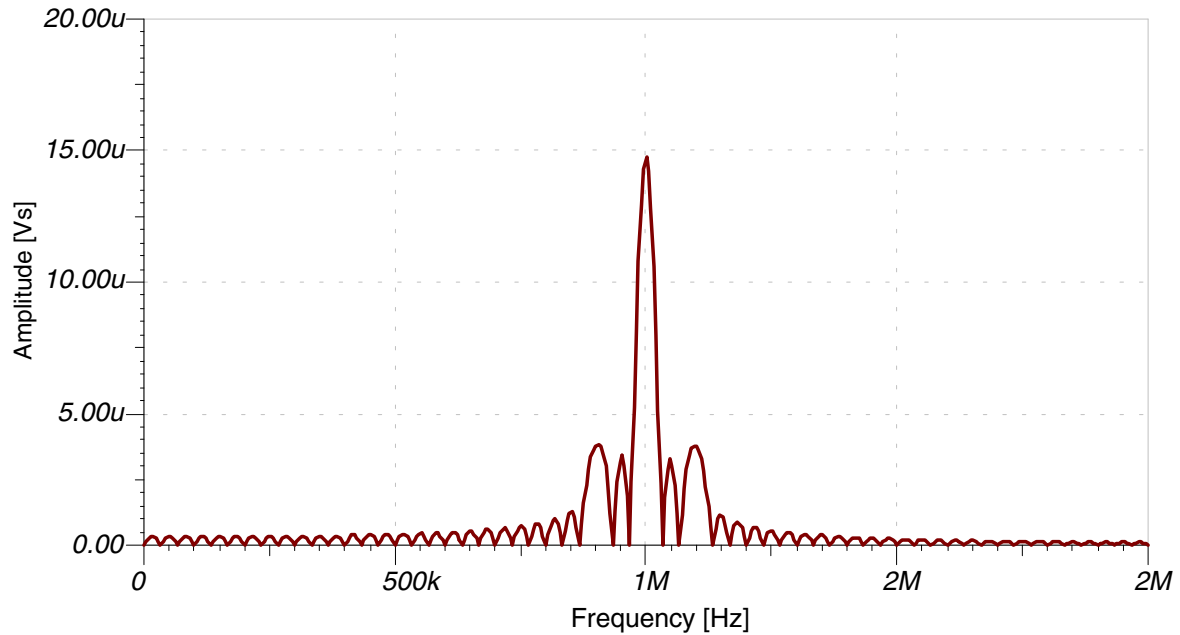
The cosinusoidal coefficients (A) are practically zero and for the sinusoidal coefficients (B) we get the theoretical result.

On the next figure the Fourier spectrum of the same signal can be seen.



You can see, that the amplitude of the smaller spikes is the quarter of the amplitude of the peak at the base frequency, which is $15\mu = T/2$, where $T=30\mu$ is the length of the modulated wave.

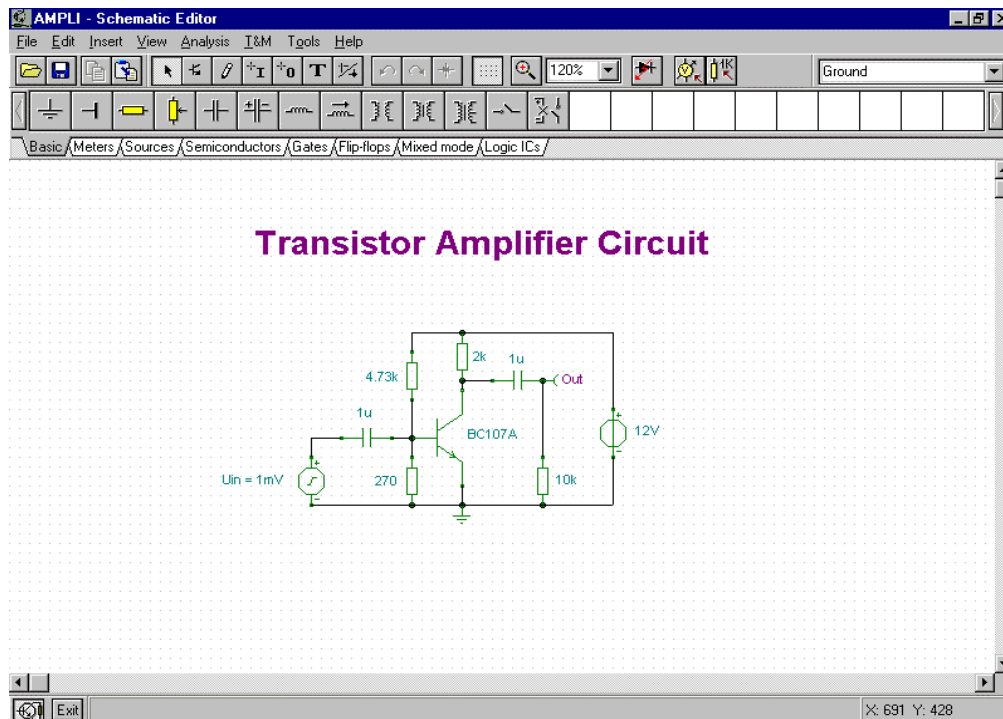
And finally on the last figure you can see the spectrum in case of 210 μ s simulation time.



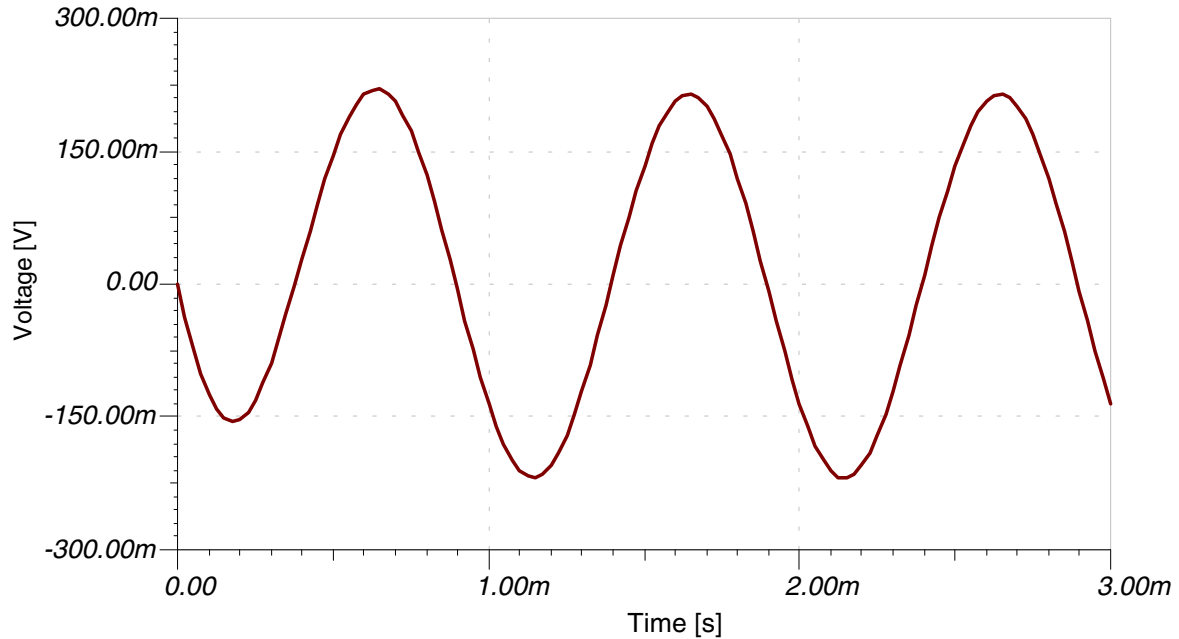
The Fourier spectrum and the Fourier series dialog box can also be obtained directly from the Analysis.Fourier Analysis menu. This way you do not need to calculate the transient function manually—TINA will automatically do it before generating the Fourier series or spectrum.

Harmonic distortion

In Tina you can also calculate the harmonic distortion.



To illustrate this, run the transient analysis with the above circuit (ampli.sch) then the following figure appears:



Select the output curve then press the right button of the mouse and select Fourier Series from the popup menu: the dialog box of the Fourier series will appear. Set **Sampling start time** to 1ms and the **Number of samples** to 4096. Note that for best accuracy, it is very important to set the starting time for the Fourier Series analysis to after the initial transient has died away. Now press **Calculate**. Tina calculates you the Fourier coefficients and the harmonic distortion, which is defined to be the ratio of the root-mean-square (RMS) sum of the magnitude of the harmonics to the magnitude of the fundamental, or as a percentage:

$$\% THD = 100 * \sqrt{\frac{V_2^2 + V_3^2 + \dots + V_n^2}{V_1^2}}, \text{ called percent total harmonic distortion (\%THD).}$$

