TINA v10

The Complete Electronics Lab for Windows

USERS MANUAL

DesignSoft

www.designsoftware.com

COPYRIGHTS

LIMITED LIABILITY

TRADEMARKS

* English version

TABLE
OF CONTENTS

DesignSoft

Contents

DesignSoft

DesignSoft

DesignSoft

DesignSoft

INTRODUCTION

1.1   What is TINA and TINA Design Suite?

TINA Design Suite is a powerful yet affordable software package for analyzing, designing and real time testing of circuits with analog, digital & microcontroller comoonents and components defined in various HDLs (Hardware Description Languages) and for designing their PCB layouts. You can also analyze RF, communication, optoelectronic circuits and mechatronics applications with 3D interface.

Every year, electronic circuits become faster and more complex, and therefore require more and more computational power to analyze their operation. To meet this requirement DesignSoft engineers have included the ability to utilize the increasingly popular scalable multi-thread CPUs. Computers that incorporate multi-core CPUs can deliver up to 20 times faster execution time for TINA's analysis engine–while maintaining the accuracy of prior single-threading computers.

There is now an online version of TINA called TINACloud. If you have licenses for both products, you can store your designs on the web and run anytime on any platforms without installation, including PCs, Macs, thin clients, tablets, smart phones, smart TVs and e-book readers. The program will run on our powerful web server with the same high speed whether you use a laptop, tablet or just a smartphone. You can then smoothly download your design from the web to your PC. Should you change something while you are on the road, continue the development off-line and upload your design again.

In TINA 10 and later versions you can create and simulate multidisciplinary designs. Disciplines now include electronics, 3D mechanics and control engineering. This opens a rich new field of applications in automotive engineering, sensing and control, automation, robotics and more.

A unique feature of TINA permits you to bring your circuit to life with the optional USB controlled TINALab II and LabXplorer hardware turn your computer into a powerful, multifunction T&M instrument. With LabExplorer, you can carry out remote measurement which is great for distance education.

TINA is distributed in two major versions – TINA Standard and TINA Design Suite. TINA Standard includes circuit simulation only, while TINA Design Suite also includes the advanced PCB designer. This fully integrated layout module has all the features you need for advanced PCB design, including Multilayer flexible PCB's with split power planes, powerful autoplacement & autorouting, rip-up and reroute, manual "follow-me" trace placement, DRC, forward and back annotation, pin and gate swapping, keep-in and keep-out areas, thermal relief, fanout, plane layers, Gerber file output and much more. TINA and TINA Design Suite also have different editions tailored to customer needs. Some HDL languages and the Mechatronics extension are optional.

TINA can also be used in the training environment. It includes unique tools for testing students' knowledge, monitoring progress and introducing troubleshooting techniques. With optional hardware it can be used to test real circuits for comparison with the results obtained from simulation. Of great importance to educators, the package includes all the tools needed to prepare educational materials.

Schematic Capture. Circuit diagrams are entered using an easy to use schematic editor. Component symbols chosen from the Component bar are positioned, moved, rotated and/or mirrored on the screen by the mouse. TINA's semiconductor catalog allows the user to select components from a user-extendible library. An advanced "rubber wire" tool is provided allowing easy modification of the schematic diagrams. You can open any number of circuit files or subcircuits, cut, copy and paste circuit segments from one circuit into another, and, of course analyze any of the currently open circuits. TINA gives you tools to enhance your schematic by adding graphics elements such as lines, arcs, arrows, frames around the schematic, and title blocks. You can also draw non-orthogonal (diagonal) components such as bridges and 3-phase networks.

Live 3D Breadboard Tool. You can take your design for a solder-less breadboard (sometimes called a "whiteboard") and automatically build a life-like 3D picture of the breadboard. Now when you run TINA in interactive mode, virtual components such as switches, LEDs, instru- ments, etc. become "live" and will function with satisfying realism. Students will use the Live 3D Breadboard tool to prepare and docu- ment eye-catching lab experiments.

PCB Design. TINA Standard includes only circuit simulation, while TINA Design Suite includes TINA's advanced PCB designer. This fully integrated layout module has all the features you need for advanced PCB design, including Multilayer PCB's with split power planes, powerful autoplacement & autorouting, rip-up and reroute, manual and "follow-me" trace placement, DRC, forward and back annota- tion, pin and gate swapping, keep-in and keep-out areas, thermal relief, fanout, plane layers, 3D view from any angle, and much more. With TINA Design Suite you can prepare a PCB in at least two ways: using the G-Code control files to make in-house prototypes with milling machines using the G-Code control files provided by TINA; or sending Gerber files to PCB manufacturers.

Electrical Rules Check (ERC) will examine the circuit for questionable connections between components and display the results in the Electrical Rules Check window. ERC is invoked automatically, so missing connections will be brought to your attention before analysis begins.

Schematic Symbol Editor. In TINA, you can simplify a schematic by turning portions of it into a subcircuit. In addition, you can create new TINA components from any Spice subcircuit, whether created by yourself, downloaded from the Internet, or obtained from a manufacturer's CD. TINA automatically represents these subcircuits as a rectangular block on your schematic, but you can create any shape you like with TINA's Schematic Symbol Editor.

Library Manager. TINA has large libraries containing Spice- and S-parameter models provided by semiconductor manufacturers such as Analog Devices, Texas Instruments, National Semiconductor, and others. You can add more models to these libraries or create your own Spice- and S-parameter library using TINA's Library Manager (LM).

Parameter Extractor. Using TINA's Parameter Extractor you can also create component models that more closely represent actual real world devices by converting measurement or catalog data into model parameters.

Text and Equation Editor. TINA includes a Text and Equation Editor for annotating schematics, calculations, includes graphic output, and measurement results. It is an invaluable aid to teachers preparing problems and examples.

The circuit diagrams and the calculated or measured results can be printed or saved to files in standard Windows BMP, JPG and WMF format. These output files can be processed by a number of well known software packages (Microsoft Word, Corel Draw etc.). Netlists can be exported and imported in Pspice format and also to drive popular PCB packages such as ORCAD, TANGO, PCAD, PROTEL, REDAC and other programs.

DC analysis calculates the DC operating point and the transfer characteristic of analog circuits. The user can display the calculated and/ or measured nodal voltages at any node by selecting the node with the cursor. For digital circuits, the program solves the logic state equation and displays the results at each node step-by-step.

Transient analysis. In the transient and mixed mode of TINA you can calculate the circuit response to the input waveforms that can be selected from several options (pulse, unit step, sinusoidal, triangular wave, square wave, general trapezoidal waveform, and user-defined excitation) and parameterized as required. For digital circuits, programmable clocks and digital signal generators are available.

Fourier analysis. In addition to the calculation and display of the response, the coefficients of the Fourier series, the harmonic distortion for periodic signals, and the Fourier spectrum of non-periodic signals can also be calculated.

Digital Simulation. TINA also includes a very fast and powerful simulator for digital circuits. You can trace circuit operation step-by-step, forward and backward, or view the complete time diagram in a special logic analyzer window. In addition to logic gates, there are ICs and other digital parts from TINA's large component library.

HDL simulation. TINA now includes all major analog, digital and mixed Hardware Description Languages: VHDL, Verilog,

Verilog-A and Verilog AMS to verify designs in analog, digital and mixed-signal analog-digital environments. Your circuits can contain editable HDL blocks from the libraries of TINA and Xilinx or other HDL components created by yourself or downloaded from the Internet. TINA compiles HDL into highly efficient machine code for speed optimization. You can freely combine HDL and Spice macros and the schematic components of TINA. Also you can edit the HDL source of any HDL components then simulate and see the result instantly. With the built in HDL debugger you can execute the HDL code step-by-step, add breakpoints, watchpoints, display variable information, etc.

Microcontroller (MCU) simulation. TINA includes a wide range of microcontrollers (PIC, AVR, 8051, HCS, ARM) which you can test, debug and run interactively. The built in MCU assembler allows you to modify your assembler code and see the result promptly. You can also program and debug MCUs in C, using external C compilers.

Flowchart Editor and Debugger. Writing MCU assembly code is often a hard and tedious task. You can simplify software development and gain more time to design the electronics hardware if, instead of manual coding, you use TINA's Flowchart editor and debugger to generate and debug the MCU code. This easy-to-use tool works with symbols and flow control lines with which you can represent the algorithm you want.

AC analysis calculates, complex voltage, current, impedance, and power can be calculated. In addition, Nyquist and Bode diagrams of the amplitude, phase and group delay characteristics of analog circuits can be plotted. You can also draw the complex phasor diagram. For non-linear networks, the operating point linearization is done automatically.

Network analysis determines 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 subcircuits (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.

Noise analysis determines the noise spectrum with respect to either the input or the output. The noise power and the signal-to-noise ratio (SNR) can also be calculated.

Symbolic analysis produces the transfer function and the closed form expression of the response of analog linear networks in DC, AC, and transient modes. The exact solution, calculated through the symbolic analysis, can also be plotted and compared to the numerically calculated or measured results. The built-in interpreter can evaluate and plot arbitrary functions.

Monte-Carlo and Worst-case analysis. Tolerances can be assigned to the circuit elements for use in Monte-Carlo and/or worst-case analyses. The results can be obtained statistically, and their expected means, standard deviations and yields can also be calculated.

Design Tool This powerful tool works with the design equations of your circuit to ensure that the specified inputs result in the specified output response. The tool offers you a solution engine that you can use to solve repetitively and accurately for various scenarios. The calculated component values are automatically set in place in the companion TINA schematic and you can check the result by simulation. This feature is also very useful for semiconductor and other electronics component manufacturers to provide application circuits along with the design procedure.

Optimization. TINA's enhanced optimization tool can tweak one or more unknown circuit parameters to achieve a predefined target response. The target circuit response (voltage, current, impedance, or power) must be "monitored" by meters. For example, you can specify several working point DC voltages or AC transfer function parameters and have TINA determine the values of the selected components.

Post-processor. Another great new tool of TIN A is its post-processor. With the post-processor, you can add new curves of virtually any node and component voltage or current to existing

diagrams. In addition, you can post-process existing curves by adding or subtracting curves, or by applying mathematical functions to them. You can also draw trajectories; i.e., draw any voltage or current as a function of another voltage or current.

Presentation. With TINA you can make quality documents incorporating Bode plots, Nyquist, Phasor, Polar and Smith diagrams, transient responses, digital waveforms and other data using linear or logarithmic scales. Customize presentations easily using TINA's advanced drawing tools-you can print your plots directly from TINA, cut and paste them into your favorite word processing package, or export them to popular standard formats. Customization includes complete control over texts, axes, and plot style; e.g., setting line width and color, fonts in all sizes and color, and automatic or manual scaling for each axis.

Interactive mode. When everything is in order, the ultimate test of your circuit is to try it in a "real life" situation using its interactive controls (such as keypads and switches) and watching its displays or other indicators. You can carry out such a test using TINA's interactive mode. You can not only play with the controls, but you can also change component values while the analysis is in progress. In addition, you can assign hotkeys to component values and switches to change them simply by pressing a key. You will immediately see the effect of the change. You can also test MCU applications in TINA's interactive mode. You can not only run and test them using the several lifelike interactive controls e.g., keyboards, but you can also debug them while the MCU executes ASM code step by step, And displays the register contents and TINA's outputs in each step. If necessary you can modify the ASM code on the fly and test your circuit again without using any other tool.

Virtual instruments. In addition to standard analysis presentations such as Bode and Nyquist plots, TINA can present its simulation results on a wide range of high-tech virtual instruments. For example, you can simulate the time response of your circuit using a virtual square wave generator and a virtual oscilloscope. Using TINA's virtual instruments is a good way to prepare for the use of real test and measurement equipment. Of course it is important to remember that the "measurement results" obtained with virtual instruments are still simulated.

Real-time Test & Measurements. TINA can go beyond simulation when supplementary hardware is installed on the host computer. With this hardware, TINA's powerful tools can make real-time measurements on real circuits and display the results on its virtual instruments.

Training and Examination. TINA has special operating modes for training and for examination. In these modes, under TINA's control, the students solve problems assigned by the teacher. The solution format depends on the types of problems: they can be selected from a list, calculated numerically, or given in symbolic form. The interpreter - providing a number of solution tools - can also be used for problem solving. If the student cannot solve the problem, he/she can turn to the multilevel Advisor. The package includes all the tools needed to produce educational materials. A collection of examples and problems worked out by teachers is also part of the package. Another special educational function of TINA is the software or hardware simulation of circuit faults to practice troubleshooting. Using TINA, you can transform existing PC classrooms into contemporary electronics training labs at low cost.

Mechatronics Extension. With this optional add-on package you can create and simulate multidisciplinary designs currently including electronics, 3D mechanics and control engineering. You can place light sources, light sensors, motors and actuators in TINA's mechanical window and connect with their counterparts in the analog, digital mixed electronic circuits. You can control the mechanics from the electronics part of TINA even with complex software written in C or assembly language, then compile and execute the code in the MCUs while running the electronic and 3D mechanical simulation simultaneously.

1.2   Available Program Versions

Different program versions, tailored to meet various needs, are available.
TINA is distributed in to major versions TINA and TINA Design Suite. TINA includes simulation only while TINA Design Suite includes our new advanced PCB designer too.

Both versions are available with the following features:

- **Industrial version:** Includes all of TINA's features and utilities.

- **Network version:** TINA can be used under most well known networks including Microsoft, Linux, Novell, Citrix and more. This feature is especially recommended for corporate and educational use.

- **Educational version:** It has most features of the Industrial version but parameter stepping and optimizations are allowed for one parameter only, Stress Analysis and the Steady State Solver are not included.

- **Classic Edition:** It has the same features as the Educational version above, except that Network Analysis is not allowed, TINA's large S-parameter component library and the Parameter Extractor, Stress Analysis and the Steady State Solver are not included.

- **Student Version:** Has the same features as Classic Edition version except that the circuit size is limited to 100 nodes including internal Spice macro nodes. The number of pads on the PCB layout is also limited to 100. Global Parameters and HDL extensions are not allowed.

- **Basic version:** Has the same features as Classic Edition except that the circuit size is limited to 200 nodes including internal Spice macro nodes. The number of pads on the PCB layout is also limited to 200. Global Parameters and HDL extensions are not allowed.

- **Basic Plus version:** Has the same features as Classic Edition except that the circuit size is limited to 800 nodes including internal Spice macro nodes. The number of pads on the PCB layout is also limited to 800. Global Parameters and HDL extensions are not allowed.

OPTIONS:

- **HDL Extension**: Extends the default VHDL hardware description language in TINA with Verilog, Verilog A and Verilog AMS

- **Mechatronics Extension** add-on package: Create and simulate multidisciplinary designs simultaneously including electronics, 3D mechanics and control engineering.

1.3   Optional supplementary hardware

1.3.1   TINALab II High Speed Multifunction PC Instrument

With TINALab II you can turn your laptop or desktop computer into a powerful, multifunction test and measurement instrument. Whichever instrument you need; multimeter, oscilloscope, spectrum analyzer, logic analyzer, arbitrary waveform generator, or digital signal generator it is at your fingertips with a click of the mouse. In addition TINALab II can be used with the TINA circuit simulation program for comparison of simulation and measurements as a unique tool for circuit development, troubleshooting, and the study of analog and digital electronics.

TINALab II includes a DC to 50MHz bandwidth, 10/12 bit resolution, dual-channel **Digital Storage Oscilloscope**. Due to its advanced equivalent-time sampling technology, TINALab can acquire any repetitive signal with up to **4GS/s equivalent sampling rate**, while in single shot mode the sampling rate is 20 MS/s. The full scale input range is ±80V, with 5mV to 20V/div ranges.

The synthesized **Function Generator** provides sine, square, ramp, triangle and arbitrary waveforms from DC to 4MHz, with logarithmic and linear sweep, and modulation up to 10V peak to peak. Arbitrary waveforms can be programmed via the high level, easy to use language of TINA's Interpreter. Working automatically in conjunction with the Function Generator, the **Signal Analyzer** measures and displays Bode amplitude and phase diagrams, Nyquist diagrams, and also works as **Spectrum Analyzer.**

Digital I/O for the high-tech **Digital Signal Generator** and Logic Analyzer instruments allow fast 16-channel digital testing up to 40MHz.

The optional **Multimeter** for TINALab II allows DC/AC measurements in ranges from 1mV to 100V and 100 mA to 1A. It can also measure DC resistance in ranges from 1Ω to 1MΩ.

You can also plug **Experimenter Modules** into the slot on the front of TINALab II, allowing you to simulate, measure, and troubleshoot virtually the whole range of analog and digital electronics.

Using TINALab II with TINA gives you the unique capability to have circuit simulation and real time measurements in the same integrated environment. This provides an invaluable tool for troubleshooting and brings your designs to life by comparing simulated and measured results.

1.3.2 LabXplorer: Multifunction Instrument for Education and Training with Local and Remote Measurement capabilities

LabXplorer turns your desktop, laptop, tablet or smart phone into a powerful, multifunction test and measurement instrument for a wide range of applications. Instruments, whatever you need, are at your fingertips. LabXplorer provides multimeter, oscilloscope, spectrum analyzer, logic analyzer, programmable analog and digital signal generator, impedance analyzer and also measures characteristics of passive electronic components and semiconductor devices.

LabXplorer can be used with its virtual instruments both stand-alone or remotely through the Internet or LAN.

It also supports the TINA circuit simulation program and its cloud based version TINACloud for comparison of simulation and measurements as a unique tool for circuit development, troubleshooting, and the study of analog and digital electronics.

In remote mode Labexplorer's virtual instruments run on most OSs and computers, including PCs, Macs, thin clients, tablets--even on many smart phones, smart TVs and e-book readers. You can use LabXplorer remotely in the classroom, computer lab, at home, and, in fact, anywhere in the world that has internet access. LabXplorer comes with various, remotely programmable, plug-in analog, digital and mixed circuit experiment boards.

CHAPTER 2

NEW FEATURES IN TINA

This chapter describes the new features and changes in the latest TINA v10 and also in the previous 9.x, 8.0 and 7.0 versions. Many of the new features were suggested by TINA users, while others were created by DesignSoft's team . We are sure you will share our excitement about these new features.

2.1   List of New features in TINA v10

- Windows 8 compatibility

- Open project files with preview of schematics and mechatronics

- Edif import

- Global Spice variables

- Integration with TINACloud, files upload and download

- Optional Mechatronics extension

- KLU - a faster solver on large-size circuits

- Enhanced and accelerated VHDL and Verilog simulation 10x times faster than in v9

- Xilinx simprim simulation in digital and in mixed mode

- Verilog A

- Verilog AMS

- MCU C compiling on 8051, AVR, PIC16, PIC18, PIC32, ARM

- Linux simulation on ARM MCU

- PSpice compatible AD-DA interface

DesignSoft

- Oscilloscope settings are stored

- Oscilloscope works together with the interactive mode

- Frequency and waveform parameters are shown with cursors

- Hotkeys to AC and Transient Analysis

- Cursor Max and Min functions added to the Process menu

2.2   List of New features in TINA v9

- Full Vista and Windows 7 compatibility

- Multi core support for dual, quad, i7 and higher multi-core processors

- Enhanced analysis speed up to 10 times higher speed on 1 core, 15x on dual core, 20x on quad core

- Improved convergence performance based on the latest state of the art algorithms

- Advanced powerful Spice-VHDL mixed mode analysis

- Extended catalog with ARM 7, ARM 9 and HCS08 microcontrollers, more SMPS ICs models, realistic ADC and DAC models, LCD display, Bi-color LEDs

- Extended MCU simulation, USB and other modules

- Programmable Design Tool where users can implement design procedures for calculating and setting circuit parameters so that the circuits can produce predefined target output values.

- Extended flowchart tool: PIC, AVR, 8051, code box, USART, external interrupt handling

- Open and save TINA designs, models and libraries directly from the Web

- Import Spice .CIR and .LIB files directly from the Web

- Import Libraries, Examples and Designs from any earlier versions of TINA v7 and later.

- Show analysis results on diagrams during analysis

- Fast diagram drawing and processing speed, even for very large circuits

PCB Design

- G-Code export of PCB designs for creating control files for milling machines
- Revised and largely extended PCB footprint catalog

2.3 List of New features in TINA v8 and TINA Design Suite v8

- Vista style installation and folder scheme
- Controlled sources
- Powerful Spice-VHDL co-simulation including MCUs
- Finite State Machine (FSM) editor with VHDL generation
- Flowchart editor an debubgger for controlling MCUs
- Any number of MCUs in one circuit
- Extended MCU catalog including PIC18, CAN and more
- Execution time measurement and statistics for Transient Analysis
- Hyperlinks can be added to schematics and to the diagram window
- Extended semiconductor catalog
- Application examples from Texas Instruments
- Labview based virtual instruments
- Interface to build LabVIEW based virtual instruments
- Wave (.wav) files can be used as input
- New Open Examples command in file menu to open built in examples
- Autosave. Save your current schematic or PCB design at adjustable time intervals.
- Parameter adding possibility to Spice subcircuits
- On line update possibility for libraries, program or both
- Post-processing formulas are stored with schematics, editable later
- Live 3D Breadboard (displaying and animating circuits with 3D parts on a virtual 3D breadboard)

DesignSoft

- 3D virtual instruments to prepare and document lab experiments
- Integrated electronic design textbook with "live" circuits (optional)
- SMPS design templates from Christophe Basso (optional)
- Detection of components or nodes linked with convergence or irregular circuit problems.

PCB Design (Only in TINA Design Suite v8 Edition, additional to TINA v8)

- Creating "flex" PCBs including 3D display
- Creating PCBs of any shape including round edges
- Buried and blind vias
- Extended catalog
- Improved optimizing autorouter
- Display of complete 3D circuits including parts connected externally to the PCB

2.4 List of New features in TINA v7 and TINA Design Suite v7

- Much faster analog solver algorithm with improved convergent properties
- Integrated VHDL support
- User defined VHDL components with VHDL source code
- VHDL components containing VHDL source code can edited and executed instantly
- MCU support including wide range of PIC processors and more
- Built in debugger and assembler compiler for MCUs
- Assembler code of MCUs can be edited and executed instantly
- External simulator and debugger for VHDL
- 3D component view in the schematic editor to review if the design is PCB ready
- Passive and active filter design
- SMPS (Switching Mode Power Supply) design support (Steady State Solver)

- Control of interactive mode from the new Interactive menu
- Stress Analysis
- Advanced integrated PCB design

  Multi-layer PCBs

  Autoplacement

  Autorouting

  Rip-up and reroute

  Follow-me trace placement

  DRC

  Forward and back annotation

  Pin/Gate swapping

  Keep-in/out areas

  Thermal relief,

  Fanout

  Gerber file output

  Copper pour

  Split planes

  Footprint editor with multi-pin footprint wizard

  3D view of PCB boards

- Support to design multi-pin schematic symbols
- More advanced Logic Design (simplification) tool
- IF statement allowed in Spice netlists
- More advanced file export (EMF, BMP, JPG) (File/Export)
- More advanced file import (EMF, WMF, BMP, JPG) (Insert/Graphics)
- Copy and Paste of any Windows dialogs (captured by Alt Prt Scr) into the Schematic Editor.
- Extended virtual instrument for real-time XY-recording, with average value, RMS calculation and recording vs. time

CHAPTER 3

INSTALLATION AND

START-UP

3.1 Installation Procedure

3.1.1 Minimum hardware and software requirements

- Intel Pentium or equivalent processor

- 1 GB of RAM

- 300 MB of available hard disk space

- CD-ROM (in case of CD ROM installation)

- Mouse or touchpad

- VGA adapter card and monitor

- Microsoft Windows 9x / ME / NT / 2000/ XP / Vista / Windows 7 / Windows 8

- Supported Networks (for Network versions): MS Windows 2000/ 2003/2008/2012 Server or later, Linux Server, Novell Netware version 3.12 or later.

    If the program is copy protected by a hardware key, the minimum hardware configuration includes also a USB port.

DesignSoft

3.1.2   Installation from CD-ROM or from the WEB

3.1.2.1 Installation from CD-ROM

To begin the installation simply insert the CD into your CD-ROM drive. The Setup Program will start automatically if the Auto-Run function of your CD-ROM has been enabled (Windows-Default). If not, Select Start/Run and type:

D:SETUP (Enter) (where D represents your CD-ROM drive).

The setup program will start.

3.1.2.2 Installation from the WEB

If you have purchased a downloadable version of TINA you need to download it using the download link sent by email after your successful purchase.

Windows usually saves downloaded files in the Downloads folder which is located under your user name in the **users** folder on the drive where Windows is installed (for example C:\users**your name**\downloads). Double-click the file to start the setup. If you purchased the Mechatronics add-on package, you should install it separately after installing TINA.

NOTE:

This software may come with copy protection. For further details see the Copy Protection and the Network Installation sections.
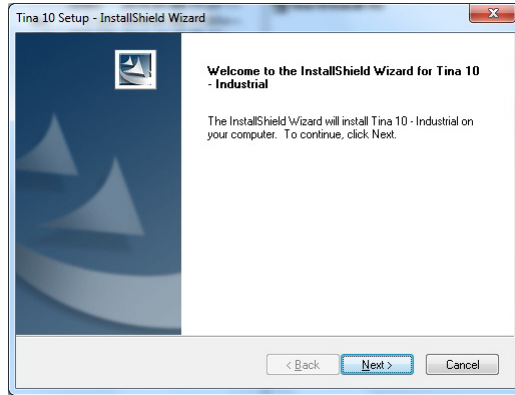
3.1.3 Following the Installation Steps

TINA's Setup Procedure follows the steps standard with most Windows Programs. There are several screen pages where you can enter or change important installation choices, such as Type of Installation, Destination Directory, etc. To continue installation, click on [Next >]. You can always step back, using the [< Back]-Button. If you do not want to continue installation for any reason, click on [Cancel]. If you elect to cancel installation, the program will ask you if you really want to exit. At this point you can either resume or exit Setup.

3.1.4 Welcome and Software License Agreement

To begin the Procedure click on Next on the Welcome Page. The first step is the Software License Agreement.
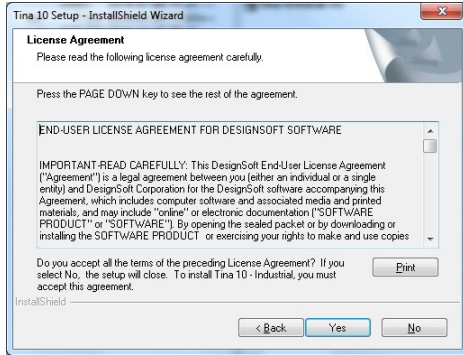
DesignSoft

NOTE:

By clicking on "Yes" you are agreeing fully with DesignSoft's Terms and Conditions for using this software.

3.1.5   Entering User Information

This data is used to personalize your copy of the software. By default, the installation program picks up the data entered when you set up Windows. You accept these names as defaults by clicking on Next or you can change them.

Depending on your program version you might also need to enter a Serial Number located on your CD-ROM package or on your Quick Start Manual.
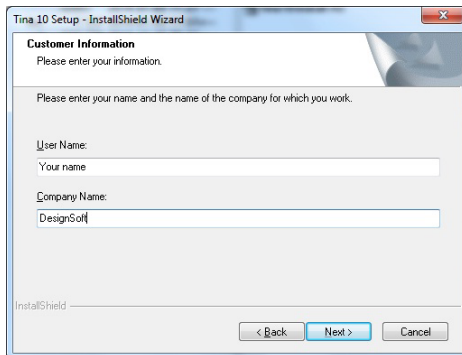
3.1.6 Networking options

3.1.6.1 Single user license (installed on a local PC)

Select this option if you have a single user license and want to use a single copy of TINA on a local PC.

However if you purchased a network license and want to use TINA in a network environment you can choose between 2 options.
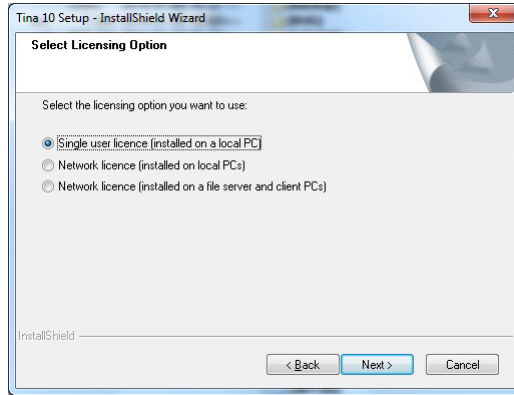
3.1.6.2 Network license installed on local PCs

Select this option if you want to use the server as a license server. In this case the server stores the license information and the software is installed on the workstations. After you've installed the package on the first workstation start TINA and select the location of the license information file on the file server then authorize the package. Finally install the software on every other workstation (client) where you want to use TINA. On these stations when you start TINA for the first time you have to locate the license file on the server.

No other authorization is required.

### 3.1.6.3 Network license installed on file server

Select this option if you want to use the server as a file server. In this case the server store the files of the software and the license

information as well. You have to select a network share in the destination dialog of the setup and install the package there. Next you have to install the package on the first workstation and authorize it. Select the Run commands from the Windows Start menu, enter the command U:\Tina\NWSetup\setup, where U: represents your network drive, and follow the instructions. Note that TINA is the main program directory holding TINA on the server. Start TINA on the first workstation.

Note that when you authorize the first workstation full access is needed on the network share where TINA is located. Authorize TINA. Finally you must run the setup program on every other workstation (client) where you want to use TINA. No authorization is required on these stations.

3.1.7   Choose Destination Location

Here you can select an Installation Directory other than the one suggested as a default. The default is the Windows Standard Directory for Programs. To change the directory, click on Browse and select a different drive and/or directory from the Choose Folder Dialog.

DesignSoft
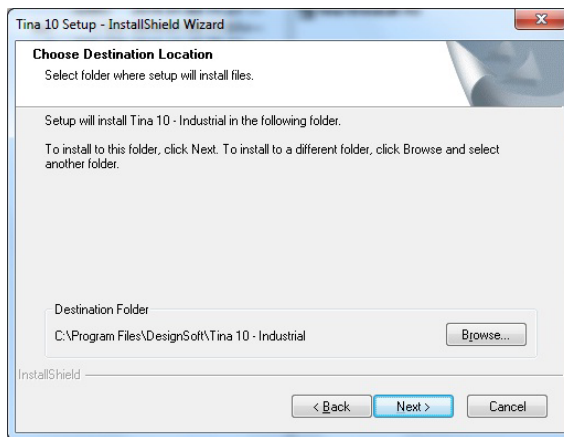
IMPORTANT NOTE:

> *If you are installing TINA for Windows to a hard disk that already has an earlier version of TINA, you must be sure to use a new directory name for TINA for Windows, such as the suggested directory, C:\Program Files\DesignSoft\Tina, or the working files you have already created will be overwritten and lost. If uncertain, exit setup, copy your TINA files safely to another hard disk directory or to floppy disks, then resume setup.*

3.1.8   Selecting a Setup Type

TINA offers you three different types of Setup. You can either run a Typical Setup (Default) a Compact Setup or a Custom Setup.
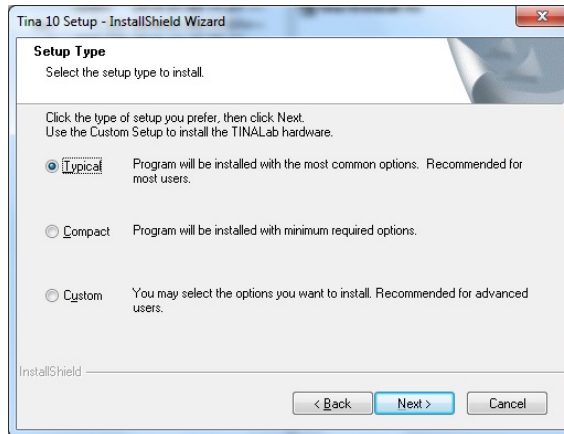
NOTE:

> *The detailed settings for the Compact installation are made after you select Compact and click on Next.*

DesignSoft

3.1.8.1 Typical

The commonly used components are installed. This includes Program Files, Samples and Utilities (i.e. Exam Manager, Spice Library Manager).

3.1.8.2 Compact

Only the most important components are installed. This results in a usable TINA installation, but one without certain program components, such as Exam Manager.

3.1.8.3 Custom

You will be able to decide which components are to be installed. The default settings are similar to those of the Typical installation. Deselect the unwanted components or select the missing ones.
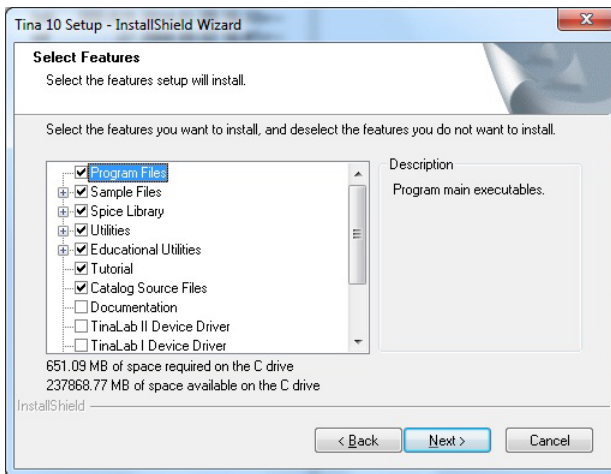
NOTE:

If you wish to install the TINALab Card, TINALab II or other third party supplementary hardware, you must select the Custom installation option at the time of installation and check the appropriate device driver on the list.

DesignSoft

3.1.9   Selecting the Program Folder

Here you can choose where in the Programs Section of your Windows Start Menu the Program Icons will appear. The default is a new sub menu called for example TINA 10 – Industrial. You can change this name or select an existing Program Folder from the list.
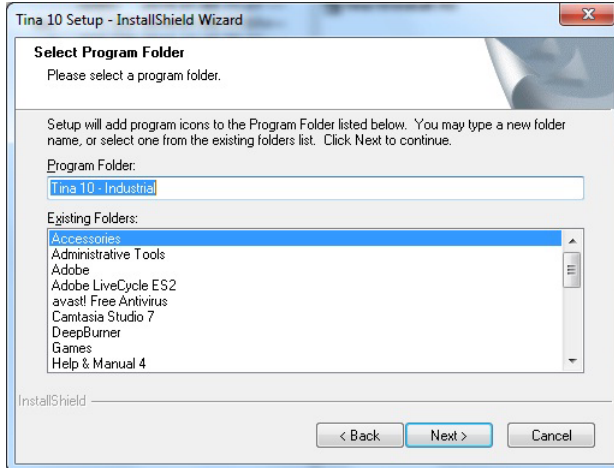
3.1.10   Select Environment Options

In TINA you can set up the settings, private/shared catalog and temporary folder. The Settings folder stores your personal settings. The private catalog folder will store your catalog files, while the shared catalog folder can be used to share catalog files with other users of the same PC or with other users in the network. The temporary folder stores the temporary files of the software. By default these folders are set to common Windows folders however you may change the folders by pressing the browse button.

DesignSoft
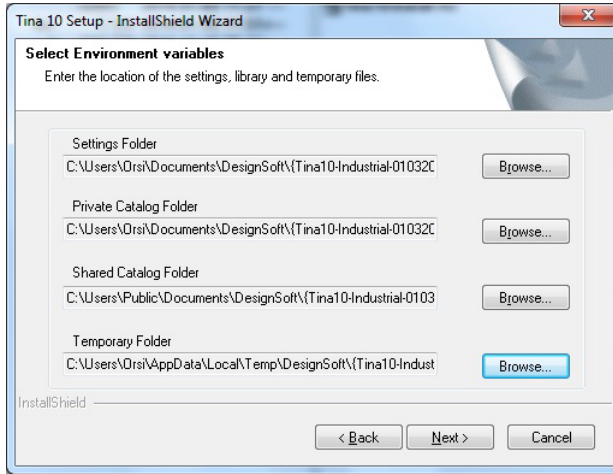
3.1.11   Selecting the Symbol Set

TINA can display its component schematic symbols according to the US (ANSI) or the European (DIN) conventions. Select the one appropriate for you.
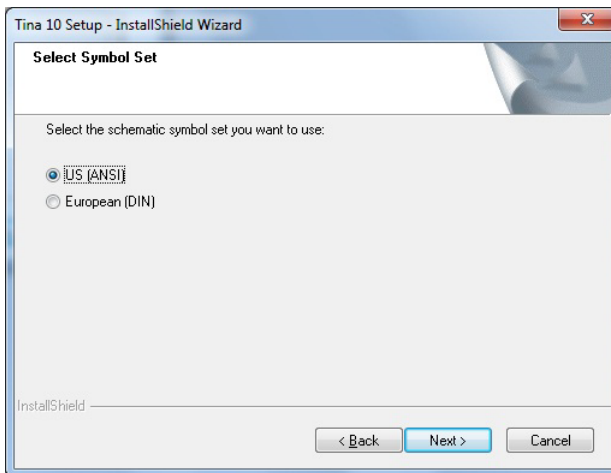
3.1.12 Final check and copying the files

This page lists the settings you have made, giving you an opportunity to check these settings and alter them and step back if changes are necessary. After you click on Next, the Setup Program starts copying the files automatically.
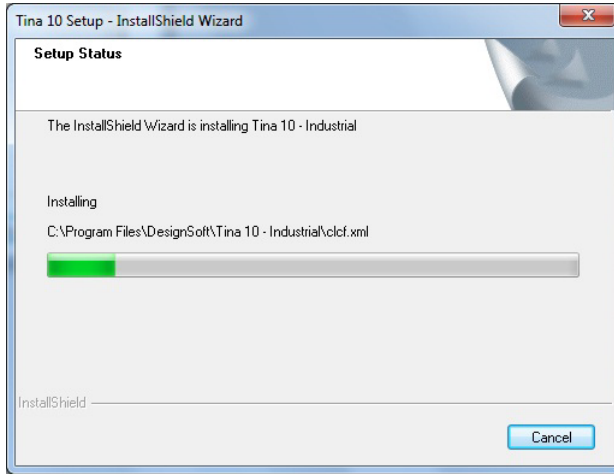
3.1.13 Completing the Setup

After all the selected files have been copied and the Start Menu entries created, you are asked if you want to place a Shortcut to the TINA program file on your Desktop. The last page indicates a successful installation and invites you to open and read a file with the latest information about TINA. We urge you to take a moment and review that file. Click on finish when you're ready.
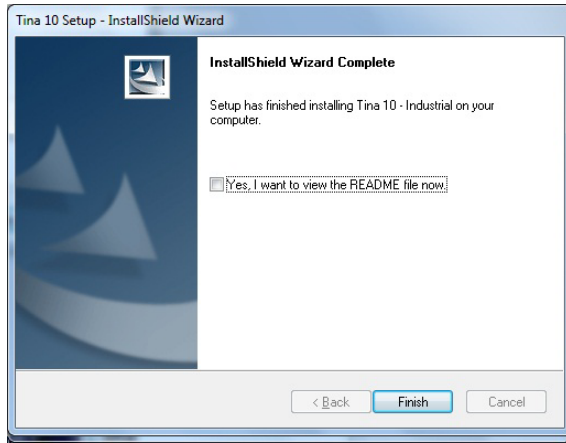
NOTE:

> *You can read the latest information in the file again at any time by selecting Read Me from the Tina Start Menu Entries. You can also get the latest information about changes or new features by visiting our Web Site, www.tina.com.*

3.2   Uninstalling TINA

You can uninstall TINA at any time. Note that this will not delete files you have created.

1. To begin Uninstallation locate Uninstall Tina on the Start or Apps screen of Windows 8 or in the Start Menu of earlier Windows versions.

2. Click on Uninstall Tina.

3. In the next dialog click on Yes if you are positive you want to uninstall TINA .

3.3   Maintaining or Repairing an Installation

You can modify or repair an existing installation of TINA as follows: Select Settings and Control panel from the Start menu of Windows. Click on the Add or Remove programs icon. Locate your TINA installation on the list and press the Change button (Press the Remove button if you want to uninstall the software). The installer of TINA will start and you can either modify the existing installation by adding or removig components or repair the current installation or uninstall the pacakage.
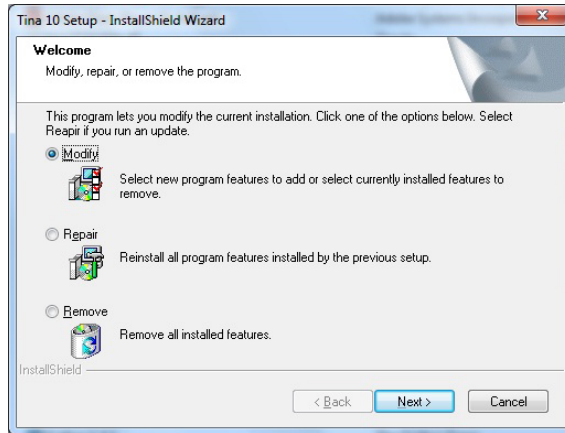
3.4   Network Installation

To install the Network version of TINA, you must have administrative privileges on the server machine and you have to share a disk volume or directory on your network, the place where TINA will be installed. With file sharing users can access TINA files. The share must be writeable by the administrator user during the installation process.

Therefore, carry out the following steps to make all files in the specified directory shared:

DesignSoft

Novell Netware 3.x: Logon to the server and perform the following commands:

```
FLAG *.* S SUB
```

Novell Netware 4.x and later versions:

```
FLAG *.* +SH /S
```

Linux server: You need Samba, a free software suite to provide file services for Windows clients. Logon as root and create a Samba share directory on your Linux system, adding the following section to your /etc/samba/smb.conf file, then restart Samba service. Example:

[TINA]

comment = TINA install folder

path = /TINA

writeable = yes

admin users = administrator root

valid users = TINAUserGroup

read list = TINAUserGroup

store dos attributes = yes

Later, you might have problem mapping Linux Samba shares to Windows Vista/7 client. Then check Vista LAN Manager authentication level: open the Run command and type "secpol.msc" and click OK. Go to Local Policies, Security Options and navigate "Network Security: LAN Manager authentication level" and open it. Change the settings from "Send NTLMv2 response only" to "Send LM & NTLM - use NTLMv2 session security if negotiated". Once you have done this, Windows Vista will be able to view network drives based on Samba servers.

Windows Server: Logon as Administrator and use the NET SHARE command, for example:

NET SHARE TINAFolder="C:\Program Files\DesignSoft\TINA"

Or, you can use the Windows Explorer:

1. Right-click the drive or folder, and then click Sharing and Security.

2. Select the "Share this folder" option and type a Share name.

3. Click the Permissions button and ensure that the administrator have Full control permission and click OK twice.

Windows Client:

Next make sure that the clients have a mapped drive set to the network drive containing the TINA program folder.

To assign (map) a drive letter to a network computer or folder do the following:

1. Open Windows Explorer

2. On the Tools menu, click Map Network Drive.

3. In Drive, select a drive letter, e.g.: G:

4. In Path (Win9x/Me) or Folder (NT/2000/XP/Vista/7), select from the drop-down list or type in the network drive (server and share name: \\MyServer\Volume1) or folder name to which you want to assign (map) a Drive letter (\\MyServer\Volume1\Program Files\DesignSoft\TINA) . Note, that share name refers to a shared folder on the server. On Windows NT/2000/XP/Vista/7 you can use Browse to find the network computer, drive and folder.

5. Set the Reconnect at Logon checkbox, then press OK.

Then execute the installation procedure directed by section 3.1.1 on the mapped disk volume, that is accessible from the network.

After you have set everything up on the network disk according to the instructions above, you must run the setup program on each client where you want to run TINA. Start setup.exe (in some versions nsetup.exe) from the TINA\NWSETUP directory.

When you run setup.exe you must specify the working directory which should be located on a local drive of the workstation.

The working directory can be on the network; however in this case the path of this directory must be different on every work station. After you've specified the working directory, you may install the optional measurement hardware for TINA (e.g TINALab). After running setup.exe, you will be able to run TINA simultaneously on any number of workstations, just as though each workstation had a single user version.

Network versions are copy protected and need authorization. For the details regarding the special procedures required for authorizing networked computers, refer to section 3.6.

3.5   Copy Protection

Executing the authorization procedure described in this section, you will be able to run TINA. If you use a network version of the program, you will be able to do it simultaneously on the number of workstations the program licensed, just as though each workstation had a single user version.

3.5.1   Copy protection by Software

If your version of TINA is copy-protected by software you need to authorize it.

You have to be in Administrator mode when you authorize the program.

NOTE FOR VISTA AND WINDOWS 7 AND 8:

Even though you are a user configured as an "Administrator", Vista and Windows 7 treats you as a standard user so the authorization may not succeed unless you make sure that User Account Control (UAC) is enabled in Vista. This is enabled by default, so if you didn't turn it off manually it should be OK. (You can find this setting in the Control Panel when you enter 'UAC' in the search field in the upper right corner of the Window.)

Authorize the program with the following steps:

1. Run Authorization & Trial from the Start or Apps screens of Windows 8 or from the TINA group of the Start menu.

2. The Authorization Manager dialog will appear.
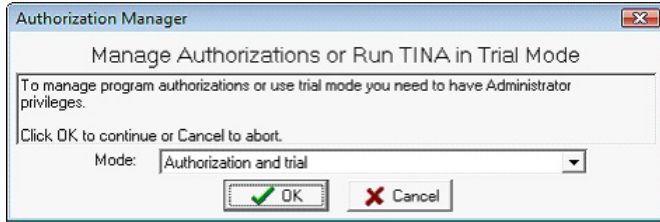
In most cases you should just press OK to continue. The License Status dialog will appear showing your initial authorization status.

If the program starts but the License Status dialog does not appear, select the Authorization/ Authorize from the Help menu of the Schematic Editor. You will normally have 31 trial sessions to provide you enough time obtain the authorization.
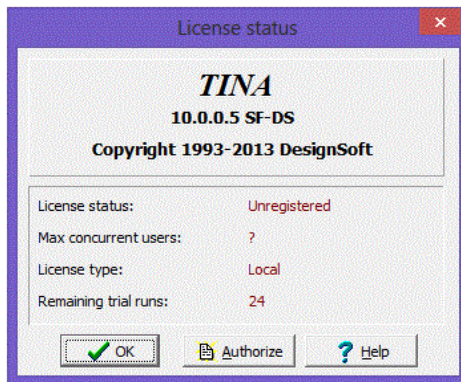
\-   Press the Authorize button on the License Status dialog which is displayed at program start or select Authorization/Authorize from the Help menu of TINA.

\-   Enter your 16 digit Order number into the Order number field of the Authorize dialog appearing and press OK. For successful operation you must be connected to the Internet and your firewall should allow communication with our server.

-      If the above is not possible for any reasons select the Other tab on the Authorize dialog.

-      Email your Site code to DesignSoft using the link in the Authorize dialog or contact your dealer.

-      We will email back a Site key, which should be copied into the Site key field of the Authorize dialog.

-      Press OK to finish the authorization.

NOTE FOR NETWORK VERSION:

If you use a network version of Tina then it will run on workstations where the setup with the NWSETUP\SETUP.EXE program has already been done. It is sufficient to authorize the software at one workstation. This will allow to run the software simultaneously, as long as the total number of simultaneous users does not exceed the licensed number. In some cases the program comes with a Serial number, which needs to be entered during installation. When entering the serial number you do not need an active internet connection. For more information refer to the program's Authorization Help by pressing the Help button.

Authorization in secure environment

If you started the program with the "Authorization and Trial" command but you still get the "Please log on with administrator privileges!" message it is a sign that you have a secure environment and need a special installation. This may be the case in systems at larger companies.

In this case select Mode: "Authorization in secure environment" in the Authorization Manager dialog. However you need to consult with DesignSoft or with your dealer and ask for a special Order number before you continue.

3.5.2   Copy Protection by Hardware (dongle)

3.5.2.1 Single user version

Make sure you are in Administrator mode.

If you have a USB dongle-protected version of TINA, install TINA first before connecting the dongle.

Next, connect the dongle to the USB port. The dongle driver installation will begin. If Windows looks for the dongle driver, select the recommended option, which is your hard disk.

If the dongle is not connected or not installed correctly the following error message will appear:

```
     Hardware protection key is not present
                    (USB).
```

3.5.2.2 Site license with multi-user dongle (DSPROTKEY)

To avoid loss or damage of dongles you may have just one dongle for a whole site. In this case you need the dongle only at the first start of the program at each workstation.

1) Install the software on each computer as described in the manual for single computers, according to the number of licenses purchased for your site.

2) Start the program with the dongle plugged in, then close the program and remove the dongle. After you authorized all workstations this way keep the dongle at safe place. If on some workstations the license is lost for any reasons (e.g. disk crash), you can reinstall the program and authorize again with the same technique. Please do not use this possibility to install the software on more workstations than licensed, because this might eliminate the possibility to recover lost licenses.

3.5.2.3 Authorization with network dongle

If you have a network dongle do the following to authorize TINA on the server.

1.   After setting up the workstations (see section 3.4), login on one workstation as Administrator (with writing right to the volume where TINA resides).

DesignSoft

2.   Connect the dongle with the above workstation. The system must recognize the dongle and the LED on the dongle should light.

3.   Start TINA. Based on the information in the dongle TINA will be authorized for the number of users licensed, and a dialog box should appear to confirm this.

4.   Remove the dongle and keep it at a safe place, as you may need it to recover the license in case of a system crash.

5.   Now TINA should run on all workstations without the dongle.

3.6   Starting Up

After successfully installing TINA, you can start the program by simply double-clicking the TINA icon on your Desktop or by choosing Tina from the TINA Start Menu Entries.

3.7   Experimenting with Example Circuits, avoiding common problems

Start the program and click the *File* menu item in the top line of the screen to drop down the *File* menu. Select the *Open* command and the standard open file dialog box appears with **.TSC*, indicating that a file name with *.TSC* extension is sought. Select the *EXAMPLES* folder, and a list of files with *.TSC* extensions will appear. After selecting a file, the circuit schematic will appear.

Now you can execute an analysis, modify or expand the circuit, and evaluate the results. Keep in mind that every command may be aborted by pressing the *[Esc]* key or clicking on the *Cancel* button.

We recommend that you load the following circuits and follow the instructions on the screen for the circuit types listed below.

This will avoid some common problems.

Oscillator circuit	EXAMPLES*colpitts.tsc*
555 Oscillator	EXAMPLES*555_AST.tsc*
Rectifier circuit	EXAMPLES*Bridge Rectifier1.tsc*

GETTING STARTED

In this chapter, we present TINA's screen format and menu structure. A step by step introduction is given using examples.

4.1   Schematic Editing Using the Mouse

Here are some basic mouse techniques to help you edit schematics:

4.1.1   Using the right mouse button

If you press the right button of the mouse at any time, a popup menu appears. Using this menu you can:

- **Cancel Mode:** Exit from the last operation (e.g. moving a component, drawing a wire).

- **Last Component:** Return to the last component and reposition it.

- **Wire:** Switch to wire-drawing mode. In this mode, the cursor turns into a pen and you can draw a wire. For more details, see the **Wire** paragraph below.

- **Delete:** Delete selected component(s).

- **Rotate Left, Rotate Right, Mirror:** Rotate or mirror the component currently selected or being moved. You can also rotate a selected component by pressing the Ctrl-L or Ctrl-R keys.

- **Properties:** Use this command to edit the properties (value, label) of the component currently selected or being moved. From the Properties menu, you can set all parameters of a component (before it is placed). This lets you place multiple copies of the component, all with the properties just entered. While you are in the component property editor, the right mouse button has another function. When you are editing the field of any component parameter other than the label field, you can copy that field to the label field by pressing the right mouse button and then selecting the *Copy to Label* command. You can accomplish the same thing by pressing *[F9]*.

4.1.2   Using the left mouse button

In the descriptions below, 'clicking' always refers to the left mouse button.

- **Selection:** Clicking on an object will select the desired object and deselect all other objects.

- **Multiple selection:** Clicking while holding down the [Ctrl] key will add the object under the cursor to the group of currently selected objects. If the object under the cursor is already in the currently selected group, clicking will remove it from the group.

- **Block selection:** To select a block of objects all at once, first make sure there is no object under the cursor. Then press and hold down the left button while moving the mouse (dragging). This will create a rectangular block, and all objects within the block will be selected.

- **Selection of all objects:** Press Ctrl+A to select all objects.

- **Moving objects:** A single object can be moved by dragging it (Position the cursor on the object, press and hold the left button, and move the mouse.) Multiple objects can be moved by first selecting them (see above), then clicking the left button while the cursor is over one of the selected objects, holding the left button down, and dragging.

- **Parameter modification:** Double-clicking on an object will bring up its parameter menu so that you can modify its parameters (if it has any).

- **Crossing wires:** The crossing of two wires does not result in a connection at the crossing unless you deliberately choose there to be one. Use Edit.Hide/Reconnect to place or remove a connecting dot. However, it is better drafting practice to never make a connection at a wire crossing, as this avoids ambiguity about the presence or absence of a dot.

- **Block or symbol copying:** After a block or symbol has been selected, you may copy it by pressing Crtl+C. Then click outside the block or symbol to release it, and press Crtl+V. You will see a copy of the block which you can place as you wish. If the schematic window doesn't show enough room for the copy, press Alt - to zoom out. Once you've located the block, click the left mouse button once to anchor it and a second time to deselect the moved block.

4.2   Measurement Units

When setting parameters for electronic components or specifying numerical values, you may use standard electronic abbreviations. For example, you can enter 1k (ohm) for 1000 (ohm). The multiplier abbreviations should follow the numeric value, e.g., 2.7k, 3.0M, 1u, etc.

The following characters indicate multiplier factors:

$p = pico = 10^{-12}$	$T = tera = 10^{12}$
$n = nano = 10^{-9}$	$G = giga = 10^{9}$
$u = micro = 10^{-6}$	$M = mega = 10^{6}$
$m = milli = 10^{-3}$	$k = kilo = 10^{3}$

NOTE:

Upper and lower cases must be carefully distinguished (e.g., M = m), and the selected letter must follow the numeric characters without a space (e.g., 1k or 5.1G), or TINA will indicate an error.

GETTING STARTED

DesignSoft

4.3 The Basic Screen Format
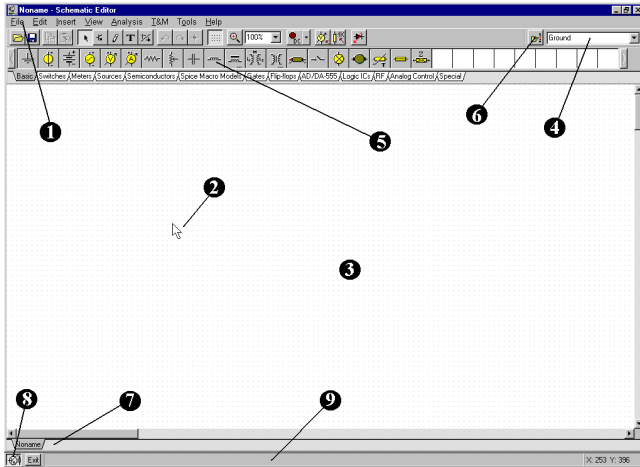
After start-up, the following screen appears on your monitor:

❶ **The Menu bar**

❷ **The Cursor or pointer:** This is used to select commands and to edit schematics. You can move the cursor only with the mouse.

Depending on the mode of operation, the cursor assumes one of the following forms:

An **arrow**, when a command selection is required in the edit window.

A **component symbol** (accompanied by an arrow and small box), when inserting that component onto the circuit in the schematic window. Until the position of the component on the schematic is chosen, its movement is controlled by the mouse.

A **pen**, when defining the endpoint of a wire.

An **elastic line**, when defining the endpoint of a wire or the second node of an input or output.

An **elastic box**, when defining a block after fixing its first corner.

A **dashed line box**, when positioning a component label or a text block.

A **magnifying glass**, when defining a zoom window.

❸ **The Schematic window:** This shows the circuit schematic currently being edited or analyzed. The schematic window is actually a window onto a larger drawing area. You can move the screen window over the full drawing area using the scroll bars at the right and bottom of the screen. When selecting the New command on the File menu, the system automatically aligns the origin of the editor window with the center of the entire editor drawing area. The same is true when an existing circuit file is loaded, as this is the default window position.

You can think of *TINA's* schematic as existing on several "layers." In addition to the primary layer that holds components, wires, and text, there are two other drawing layers, which you can turn on or off individually. It is generally convenient to have these two layers on.

View|Pin Markers On/Off: Displays/hides component pin ends.

View|Grid On/Off: Displays/hides the grid.

A grid of closely spaced dots covering the entire drawing area may be made visible or invisible in the schematic window, depending on the current state of the *grid button* of the *Grid On/Off* switch on the *View* menu. At some schematic zoom levels, you will not see the dots of the grid; nevertheless, all component pins and connecting wires will be on the grid. These dots represent the only available interconnecting points. Component symbols are positioned on the drawing area horizontally and vertically. These symbols are rigid patterns with predefined pin positions and are handled as single units. This permits the software to unambiguously recognize the network nodes.

❹ **The Tool Bar:** You can select most of the editor commands (e.g., select, zoom, wire etc.,) from this tool bar. Let's summarize the most important commands on the Toolbar. You can find more detailed information in TINA's Help system. Note that most commands on the toolbar can also be found under the drop down menus, and can often be activated by Hotkeys. We show the menu name separated from the command name by a dot (Menu name.Command name).

File Edit Insert View Analysis Interactive T&M Tools Help
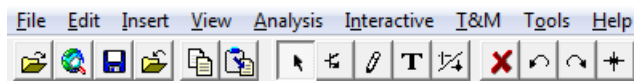
(File.Open) Opens a schematic circuit file (.TSC or .SCH), TINA macro files (.TSM) or Spice netlist files (.CIR).

.TSC extension is the current schematic file extension used in TINA v6 and above. The .SCH extension was also used for schematics in TINA v4 and v5.

.TSM is the extension of TINA macros, which can contain asubcircuit either as a TINA schematic, Spice netlist, or VHDL code.

.CIR files must be circuit files or subcircuits in Spice netlist format. The files will appear in the Netlist Editor, where you can execute most TINA analyses, and edit or complete the netlist.

(File.Open from the web) This command starts the built in web browser that allows you to navigate to any website and then directly open TINA files with TSC, SCH or CIR extension by simply clicking a link. It will also save TSM, LIB, and TLD files into their proper place in the user area. Finally, TINA will recognize the.ZIP file extension and will help you select, copy, and extract files.

By default the built in web browser in TINA opens the TINA circuits on Web page on the www.tina.com website (at the time of writing this manual www.tina.com/English/tina/ circuits) where you will find interesting electronic circuit files which you can download or open directly from the web and then simulate with TINA.

(File.Save) Saves the actual circuit or subcircuit into its original storage location. It is wise to frequently save the circuit that you are working on to avoid loss of data in case of a computer crash.

(File.Close) Closes the actual circuit or an open subcircuit on the screen. It is very useful for closing an open subcircuit.

(Edit.Copy) Copy a selected part of the circuit or text to the clipboard.

(Edit.Paste) Paste Clipboard contents into the schematic editor. Note that the content may come from the schematic

editor itself, TINA's diagram window, or any other Windows program.

▶ *Selection mode.* If this button is pressed you can select and drag components with the cursor. To select a component (part), wire or text, just click on it with the cursor. You can also select several objects by holding down the Ctrl key and clicking on the objects one-by-one; or by clicking at one corner of the area, holding down the left mouse button, moving to the opposite corner and then releasing the mouse button. Selected objects will turn red. You can drag the selected objects by dragging one of them. Click and hold the left mouse button when the cursor is over one of the selected objects and move them with the mouse. You can unselect all selected object(s) by clicking on an empty area. One or more selected objects can be deleted while leaving the others still selected by holding the Ctrl key down and left-clicking the mouse.

You can select all the wire segments, connected by the same ID on Jumpers, by holding down the Shift Key and clicking one of the wire segments.

⊢ *(Insert.Last component)* Retrieves the last component inserted, for a new insertion of another copy, with the same parameters as the previous insertion.

𝓁 *(Insert.Wire)* Use this icon for inserting (adding) wires to the schematic design.

T *(Insert.Text)* Add comments into schematics and analysis results.

↳ *(Edit.Hide/Reconnect)* Use Hide/Reconnect to place or remove a connecting dot between crossing wire or wire-component connection.

↶ ↷ *(Edit.Rotate Left (Ctrl L), (Edit.Rotate Right (Ctrl R),* Rotates the selected component.
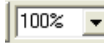
✦ *(Edit.Mirror)* Mirrors the selected component.

Hotkeys Ctrl L, Ctrl H

Switches On/Off the grid. I.e.makes the grid visible or invisible.

Explicitly zoom in on a selected portion of the current view. It will also zoom out a portion if you just click with the icon into the middle of the area you want to zoom out.

100% *Select zoom ratio* from a list from 10% to 200%. You can also select Zoom All which will zoom the effective drawing to full screen.

Interactive modes, see also on the Interactive menu:

DC mode

AC mode

Continuous transient mode

Single shot transient mode, the time is the same as set at Analysis Transient

Digital mode

VHDL mode

With this listbox you can select the analysis and the Interactive Mode Options dialog.

(Analysis.Optimization Target) Select Optimization Target to set up Optimization mode or to change settings.

(Analysis.Control Object) Select Control Object for Parameter Stepping or Optimization.

⊬ *(Analysis.Faults enabled)* If this button is pressed it enables component Faults, set by Fault property of components. You can set component faults with the Property Editor by double-clicking on components.

⊘ *(View.3D view/2D view)* Hotkey F6. 2D/3D view. If this button is pressed, TINA's Schematic Editor displays circuit components as 3D pictures of the real component assigned to the schematic symbol. This is a simple but useful check before starting the PCB design.

⊣ *(Tools.PCB Design)* Invokes the dialog which initiates the PCB design module of TINA, if available.

☞ Voltage Pin ▼ **☞** *(Tools.Find component)* Find Component tool. Invokes a component searching and placement tool. This tool helps you find by name any component in the TINA catalog. The search string that you enter will be found wherever it occurs at the beginning, at the end, or anywhere within the component's name. This tool is useful when you don't know where a particular component is located, or if you want a list of all the components that match particular search criteria. A component found in a search can be placed immediately into the schematic by selecting it and pressing the Insert button of this tool.

Resistor ▼ *Component list*. With this tool you can select components from a list.

❺ **The Component bar:** Components are arranged in groups, named by the tabs on the Component bar. Once you have selected a group, the available component symbols appear above the tabs. When you click on the desired component (and release the button), the cursor changes to that component symbol and you can move it anywhere in the drawing area. You can also rotate the component by pressing the + or - keys (on your computer's numeric keypad) or mirror it by pressing the asterisk (*) key (also on your computer's numeric keypad). Once you have selected the part's position and orientation, press the left button of the mouse to lock the symbol in place.

❻   **Find component tool:** This tool helps you find by name any component in the TINA catalog. See more details above at the toolbar description.

❼   **Open files tab:** You can have several different circuit files or different parts (macros) of a circuit open in the schematic editor at the same time. Clicking on a tab brings that circuit page up in the editor.

❽   **The *TINA* Task bar:** *TINA's* Task bar appears at the bottom of the screen and provides speed buttons for the various tools or T&M instruments currently in use. Each tool or instrument operates in its own window and can be made active by clicking on its speed button (icon of the tool). Once the cursor is over the speed button, a brief hint appears. Note that the first button (furthest to the left), the Lock schematic button, has a special function. When the Lock schematic button is pressed, the schematic window is locked in place as a background behind other windows, so that it can never cover a diagram or virtual instruments. When the schematic window is not locked and it is currently selected, you will always see the entire schematic window with any other windows hidden behind.

❾   **The Help line:** The Help line, at the bottom of the screen, provides short explanations of items pointed to by the cursor.

4.4   Placing the Circuit Components

Components are selected from the Component bar and their symbols are moved by the mouse to the required position. When you click the left mouse button, the program locks the pins of the component symbol to the nearest grid dots.

Components can be positioned vertically or horizontally and rotated by 90-degree steps in a clockwise direction by pressing the [+] or [Ctrl-R] keys, or in a counterclockwise direction with the [-] or [Ctrl-L] key. In addition, some components (like transistors) can also be mirrored around their vertical axis by using the *[*]* key on the numeric keypad. You can also use the ⟳⟲✱ buttons or the popup menu (right mouse button) to position components.

After a component symbol has been selected and positioned, you may double click on it to enable a dialog window where you can enter parameter values and a label. When entering numeric values,

abbreviations of integral powers from 10^{-12} to 10^{12} can be used. For example, 1k is understood as 1,000.

NOTE:

> *Upper and lower cases must be carefully distinguished (e.g., M = m), and the selected letter must follow the numeric characters without a space (e.g., 1k or 5.1G), or TINA will indicate an error.*

TINA will automatically assign a label for each component you place on the schematic. It will also display the numerical value of the main component parameter (for example: R4 10k). Note that the value is shown only if the Values option of the View menu is checked. For files from the older versions of *TINA*, the Values option is turned off by default. The first part of the label, e.g., R4, is required for symbolic analysis modes. You can also display the units of the capacitors and inductors (for example: C1 3nF) if both the Values and the Units options of the View menu are checked.

4.4.1   Wire

A wire establishes a simple short (zero ohm connection) between two component pins.

To place a wire, move the cursor to the component terminal point where you want to begin. The cursor will change into a drawing pen. You can draw a wire in two different ways:

1) Select the starting point of the wire with a left mouse click, then move the pen with the mouse while TINA draws the wire along the path. While drawing the wire, you can move in any direction and the wire follows. At the end point of the wire, click the left button of the mouse again.

2) Hold down the left mouse button while positioning the pen; release it at the end point.

While drawing a wire, you can delete previous sections by moving backwards on the same track. By pressing the Ctrl key while drawing you can move the last horizontal or vertical section.

DesignSoft

You can easily modify existing wires by selecting and dragging sections or edges.

For short wire sections, you may need to hold down the shift key while drawing.

You can also invoke the Wire-drawing tool by the Insert|Wire command (hotkey: [Space]). You can start drawing the wire at any place by clicking the left button of the mouse. When you have completed wiring, use the popup menu, press the right mouse button, or press the Esc key to terminate the wiring mode.

Be sure not to leave any component nodes unconnected. If there are unconnected components or terminals, TINA's Electric Rule Check tool (ERC) will issue a warning (unless you have disabled it).

Wire segments made by the Wire tool are always vertical or horizontal. However, you can add angled wire segments using the components made for bridges, Y and D circuits under the Special component toolbar.

4.4.2   Input and Output

Certain types of analysis (DC Transfer characteristic, Bode diagram, Nyquist diagram, Group delay, Transfer function) cannot be executed until both input and output have been selected. These establish where the excitation is applied and where the circuit response is taken. The output(s) chosen also determine which curve(s) will be displayed in the chosen analysis mode. Sources and generators can be configured as inputs, while meters can be configured as outputs. However, meters can also serve to determine the location of the input quantity that will be used when computing AC Transfer curves and functions. For even greater flexibility, inputs or outputs can be established at nearly any location by using the **Insert|Input** and **Insert|Output** commands. Note that you can define the input parameter for parameter sweeping only through the **Insert|Input** command.

To insert an input or output, select the Input or Output command from the Insert menu and move the input (**I+)** or output (**O+**) symbol attached to the cursor over the first schematic node that it should define. Click on that node, release the mouse button, and move the symbol to the second node, and click on that node. The program will draw a dashed rubber line between the two nodes while

drawing, and will also place this line on the schematic when you click on the second node.

Since an input reference can be established in so many ways, it is important to remember that only one input at a time can be defined within a circuit.

Similarly, in some of *TINA's* analysis methods (e.g., Symbolic Analysis) only one output can be defined within a circuit.

4.5   Exercises

These exercises will help you build upon and integrate what you've learned from the manual so far.

4.5.1   Editing an RLC Circuit Schematic

Create the circuit diagram of a series RLC network as shown in the following figure.

DesignSoft
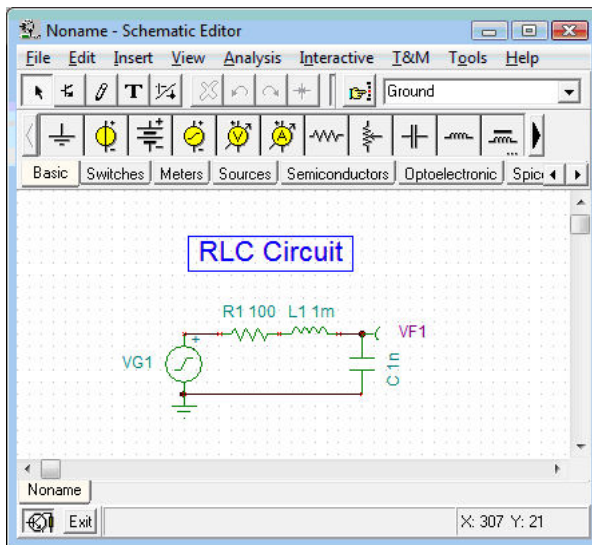
First clear the schematic window with the **File | New** command. The file name in the top line is set to Noname, indicating that a new circuit file is being edited.

Now start adding components. Click on the voltage generator icon, then release the mouse button. The cursor will change into the generator symbol. Position it using the mouse (or by pressing the [+]/[Ctrl-R] or [-]/[Ctrl-L] key for rotation] or the *[*]* key for mirroring) somewhere in the middle of the screen, then press the right mouse button; the schematic editor's popup menu will come up. Select *Properties*. The following dialog box will appear:
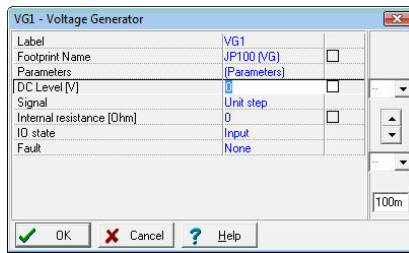
Leave the *DC level* and the *IO state* parameters unchanged. Note that by accepting *Input* for the *IO state* parameter you have selected the output of this generator to be the input for the Bode diagram.

Select the *Signal* menu line and then press the ▣ button, a new dialog box with the graphics icons of available voltage generator signals will appear. When you select one of them (in this case, click on the Cosinusoidal button), the associated curve comes up with some default parameters. In the case of the Cosine signal, these are:
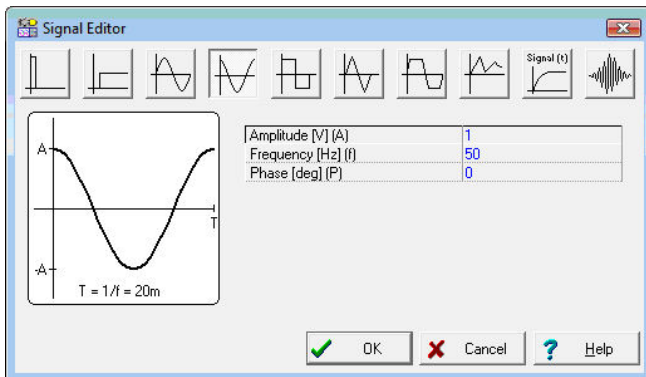
Change the frequency to 200k (200kHz). Click on *OK* and return to the previous dialog box and click on *OK* again. The program will automatically place the label near the component and you will be able to position and place the component and the label together. If the default label position is not satisfactory, you'll be able to drag the label to the desired position later on. When the component is where you want it, press the left mouse button to drop it. This completes the placement of the generator.

Now click on the *Basic* tab on the Component bar and choose the **Resistor** icon (your cursor will automatically change when you are over the tabs or the icons). After the symbol of a resistor has appeared in the schematic window, press the right button of the mouse and then select *Properties* from the popup menu.

When the dialog box appears, change the Resistance to 100.

After setting all parameters, click *OK.* Your cursor will turn into the resistor with the frame of the label. Position it as required and press the left mouse button to drop it.

Continue circuit entry with the **L** and **C** components as indicated in the figure above. Set the parameters to *L 1*= 1 m and *C1* = 1 n. Note the default values of the parallel resistive losses for the capacitor and the series resistive losses of the capacitor. Add the Voltage Pin (chosen from the Meters component group) on the upper pin of the capacitor (or you can add a volt- meter in parallel with the capacitor). Note that even though all the computed voltages, currents and signals are available after running an analysis (see below in this chapter and also in the *Post-processing analysis results* section), you still need to define at least one output. Place a ground below the generator and connect the generator and capacitor as shown in the figure. To do this, move the cursor over the appropriate pin node until the small drawing pen appears. When the pen appears, click the left button of the mouse, draw the wire, and left click again at its end point.

Finally, add the title to the schematic. Click the T button and the text editor will appear. Type in: RLC Circuit. Click on the icon and set size 14. The editor also lets you choose another font, style, color etc. Now click on ✔, then position and drop the text on the schematic editor Window.

Before going on, save the circuit with the *File|Save As* command. Name the circuit as **RLC_NEW.TSC** (the *.TSC* extension is added automatically).

Autosave. Note that TINA can automatically save your current schematic at adjustable time intervals. You can set the time interval at the Options dialog (under the View menu):

Autosave interval (0 - no autosave)   5   [min]

The default value is 5 minutes. To disable Autosave, set the interval to 0. Note that you must give the circuit file a unique name–certainly not the default name Noname.TSC–in order that the autosave command will be completed.

If you wish, you can still change the circuit in many ways:

• Add new components.

• Delete, copy, or move selected objects using the Edit|Cut, Copy, Paste, and Delete commands.

• Move, rotate, or mirror groups of components. Select the components one by one, holding down the shift key as you click on them. You may also use window selection to identify the group. When you've selected the last component, release the left mouse button, then move the cursor over one of the selected components, press and hold down the left mouse button, and drag the selected parts with the mouse. While dragging, you can use the [+]/[Ctrl-L]/ [Ctrl-R], [-] and [*] keys to rotate and mirror components.

• Move any component label separately by clicking on it and dragging.

• Modify component parameter value(s) and labels of the component by double-clicking on it.

Of course, if you want to keep these changes you must save the circuit again.

4.6   Analyses

TINA has a variety of analysis modes and options:

The analysis method is analog when a circuit contains only analog components; then the components are modeled with their analog models.

The analysis method is mixed when a circuit contains both analog and digital components. TINA will analyze the analog parts in analog, the digital parts in digital, and will automatically create the interfaces among the components. This ensures synchronization and fast convergence.

The analysis method is digital when a circuit contains only digital components; then the components are modeled with their fast digital models.

Analysis Options

This dialog, with which you can set analysis parameters, is on the Analysis menu. Examine the screenshot that follows to see the parameters that you can adjust.

In this manual we will present only the most important options, those which you might want to change. The complete explanation of all the options is included in the on-line help, which you can display pressing the Help button on the dialog.

Note: If you change any settings in this dialog and close the dialog by pressing the OK button, the program will remember the changed settings.

Performance: TINA v9 and later support multi-core processors and run analysis in parallel threads. This results in a significant speed increase. By default the number of threads is equal to the number of cores in your CPU. However you can control this with the Number of threads parameter.

Number of threads: This parameter is set to Max by default and results in one thread per core. You can, however, set it to any number between 1 and Max, or to Dynamic. If you have a quad (4) core machine, it would be good to set this parameter to 3, to leave some computing capacity for other programs and processes. You can also use the Dynamic parameter which controls the number of threads automatically, depending on other processes running on your computer.

Matrix compilation: By default this parameter is enabled and results in compilation of an extremely fast code for some matrix operations. The only reasons to disable this parameter are to perform speed comparisons or to debug software. For maximum performance, it should normally be enabled.

Enable run-time statistics: If this option is set, Tina displays the simulation time of the transient analysis in the status bar and creates a file which contains detailed information on the last transient run. You can load this file by selecting the View.Transient statistics menu item.

Enable instant diagram drawing: If this option is selected TINA will draw the diagram during transient analysis, refreshing the diagram every 1-2 seconds. This is very useful for observing the progress of long calculations.

Save all analysis results: Check this box if you want to save the result of all nodal voltages, resistor, capacitor and inductor voltages and currents, to make the later post processing more convenient. However this option may slow down the analysis by 30-50%.

Disable warning for large site analysis results If you run a transient analysis, a warning will appear if the number of points is greater than 1,000,000 (1M). Select this checkbox if you want to disable this warning.

4.6.1 Analyzing an RLC circuit (DC, AC, Transient and Fourier analysis)

Now execute AC and transient analyses on the RLC circuit you have just entered.

First perform an AC nodal analysis. Select **Analysis|AC analysis|Calculate nodal voltages.** Your cursor will turn into a test probe which you can connect to any node. In a separate window the nodal voltages will be displayed. If you have placed any meters on the schematic, clicking on them with the probe will present detailed information from that instrument. Note that you can acquire DC nodal voltages in a similar fashion through DC Analysis.

Now select **AC Analysis|AC Transfer Characteristic...** from the main menu. The following dialog box appears:
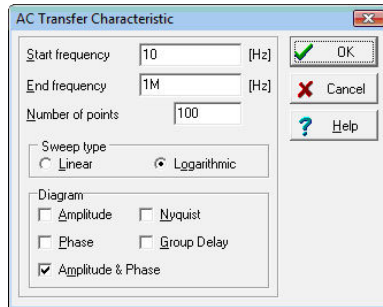
By default Amplitude & Phase will be calculated. Select **Amplitude** and **Nyquist** in addition. Modify the Start frequency to 10k and then press *OK*. A progress bar will appear while the program is calculating. After the calculations are finished, the Bode amplitude characteristic will appear in the Diagram Window. You can easily switch to Nyquist or Amplitude & Phase diagrams by using the Tabs at the bottom of the Diagram Window.

You can read exact input/output values by enabling one or more of the cursors. Note that in any representation you can get and place the formula of the transfer function using Symbolic Analysis and selecting AC Transfer or Semi Symbolic AC Transfer. The formula will appear in the Equation Editor Window and you can place it either on the Diagram or the Schematic window as described above.

GETTING STARTED
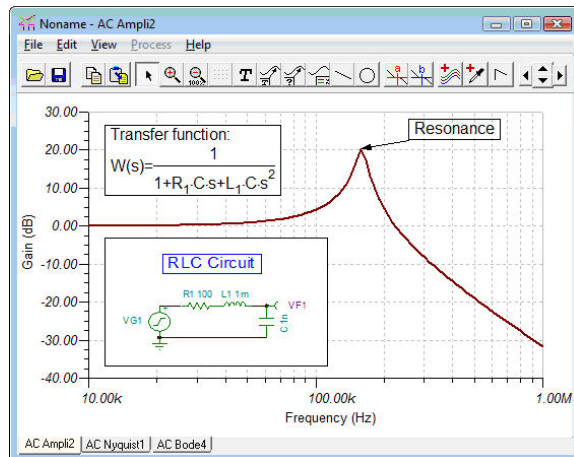
Using *TINA's* graphic facilities, you can add more useful information to your diagram. As an example, let's add markers, a special annotation, and the circuit schematic itself to the diagram.

To add markers to a curve move the cursor over the curve, find a position when the cursor changes into a + shape and click on the curve at this position. When the curve becomes selected, it turns red. Now you can either double-click on it or press the right button of the mouse and select Properties on the popup menu. A dialog box will appear and you can set the curve's parameters: Color, Line width, Marker. Select Marker Type: Square and click on *OK*.

To add some text, click on the T text icon. When the Text editor appears, type in **Resonance**. Note that using the F font icon of the editor you can select any font, style, size and color. Click on OK and place the text in the neighborhood of the resonance peak. Now click on the pointer icon, then on the text, and finally on the peak of the curve. Note that the cursor turns into a + when you are at the right position. You have just entered a line and arrow that will always point from the text to the curve, even if you drag the text into another position or make other changes.

Now place the schematic itself on your diagram. Click the schematic editor Window and select *Edit|Select All*. Copy this selection into the clipboard by selecting *Edit|Copy* or clicking on the Copy icon or using the Ctrl C hotkey. Click on the Diagram Window and use *Edit|Paste*, or click on the Paste icon, or use the Ctrl V hotkey. The frame of the circuit diagram will appear. Position and drop it at the left corner of your diagram. Now you can still modify this picture by dragging or double-clicking on it and changing its size, frame or background.

Now perform a transient analysis. First, make sure your cursor is the selection arrow, then double-click on the voltage generator and change the waveform to the default unit step. After selecting **Analysis | Transient Analysis**, the following dialog box appears:
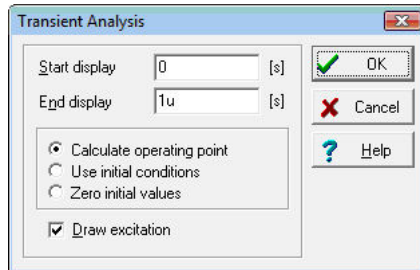
Change the *End Display* parameter to 30 u then press OK. In a separate window the transient response will appear.

As expected, the RLC circuit exhibits a response of damped oscillation. Exact input/output data pairs can be read by enabling the a and/or b graphic cursors.

Now select **Analysis|Symbolic** or **Analysis|Semi-symbolic Transient** from the menu. The closed form expression of the circuit response appears in the Equation Editor window. Click on the Copy icon of the Equation Editor Window, then switch to the Schematic window and select the Paste icon. The frame of the formula will appear. Move the frame to the desired location and press the left mouse button to place the formula. Note that you can reposition it by dragging to any position and you can edit it by double-clicking on it.

Now go back to the equation editor and click on the Interpreter icon (small calculator) on the toolbar. The expression shown in the equation editor is then transferred to the Interpreter window. The actual definition of the time function is at the top of the window, followed by drawing preferences and the 'Draw' command. Press the run button to draw the function in the diagram window on a new page. This curve can then be copied and pasted into the same transient function diagram, where all of the results can be seen simultaneously.

TR Result :
$$v(t)=(1+1 \cdot e^{-5 \cdot 10^4 \cdot t} \cdot \cos(9.99 \cdot 10^5 \cdot t - 182.87°)) \cdot \varepsilon(t)$$
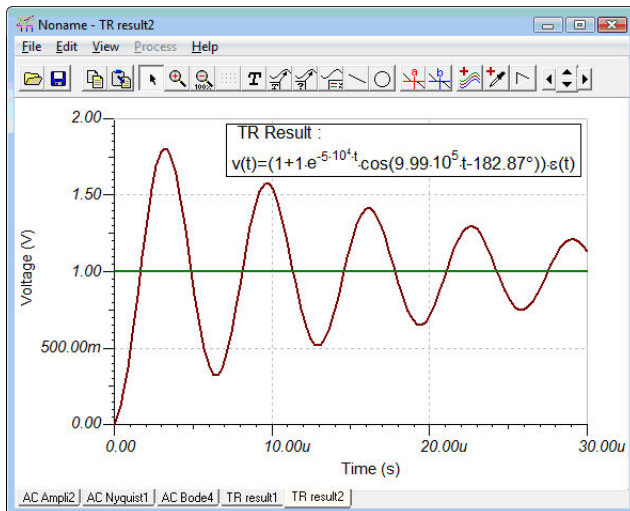
To make it easier to work with curves, you can switch them off or on individually. Use the Show/Hide curves… command under the View menu of the Diagram Window. Before demonstrating this feature, set the checkbox in the Analysis/Analysis Options dialog as shown in the partial screenshot below:
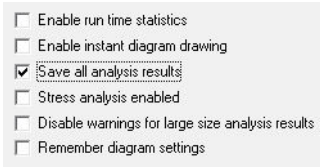
If the check box was not yet set, run the Transient Analysis again. Select the Show/Hide curves… command from the View menu of the Diagram Window. The following dialog box appears:
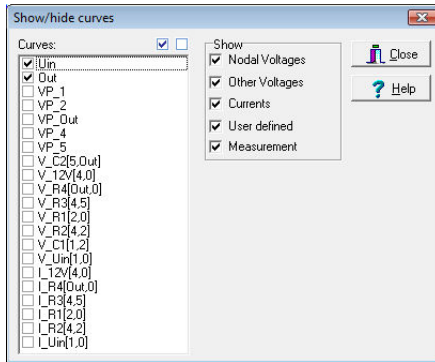
Now you can show or hide any voltages and currents in the circuit using the checkboxes. VP_1, VP_2, VP_3 are nodal voltages of the nodes denoted by numbered yellow boxes **1**, **2**, **3** on the schematic diagram in this working mode of the program.

DesignSoft
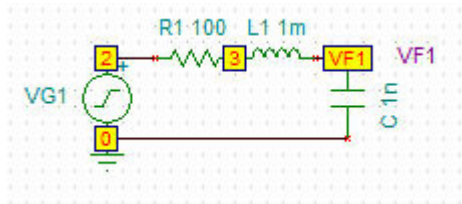
You can also select/deselect component or nodal voltages by moving the cursor and clicking above the selected component or node. Note that at least one of the original outputs (Out and Source) must remain switched on.

Another method to add more curves or process analysis results is to use TINA's Post Processor.

Note; in order to use this feature, you must check "Save all analysis results" in the Analysis.Options dialog.
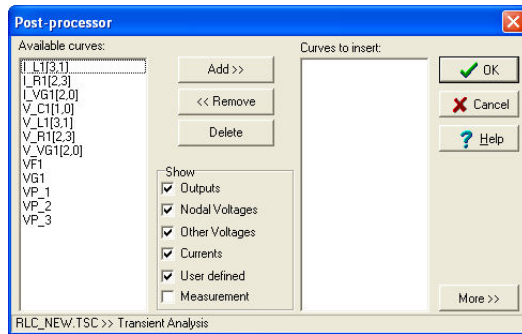
The available curves listbox on the left side shows all the curves that have already been calculated.

The symbols V_label[i,j] and I_label[i,j] denote the voltage and current respectively of the labeled components between the nodes *i* and *j*. The symbol VP_n denotes the nodal voltage of node *n*.

To add the voltage of the coil to the list of curves to insert, select V_L1[3,1] and press the Add>> button. Pressing OK inserts this curve into the current diagram page.
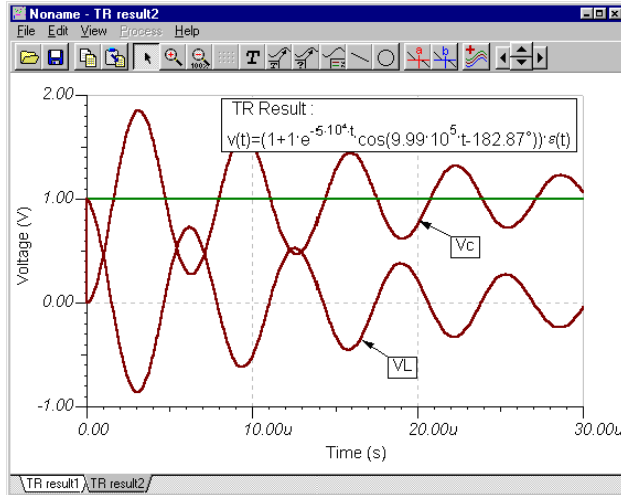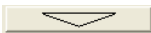
Although you could add a new curve to the diagram, if you run the Transient analysis again the voltage of the previously added coil would not be shown in the diagram. Next add another voltage to the diagram, in a way that the program retains that setting for future calculations and also see how to modify or delete these settings. As an example add the voltage of the RL-part of the circuit to the diagram.

First run the Transient analysis and then open the Post-processor by pressing the 🔺 Post-processor button on the toolbar. The Post-processor window appears. Press the More >> button on the right-bottom side of this window, the extended dialog box of the Post-processor appears.
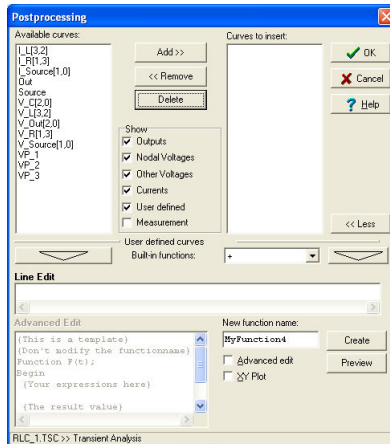
The best way to express the voltage of the RL part is, to express it as a difference of the generator voltage VG1 and the VF1 output voltage: `VG1-VF1`

To create the required voltage by the Post processor click the VG1 voltage on the Available curves list and then press the [⌄] „Add to edit area" button. VG1(t) appears in the Edit line of the post processor. Now enter the – operand and click the VF1 voltage on the curve list and press the [⌄] button again.

Note that although you could do the same by using the VP_1, VP_2, VP_3 these may change if you edit your circuit. Therefore it is better to use node outputs, voltage pins etc. as these are more independent of circuit changes.

The extended part of the postprocessor dialog looks like this.
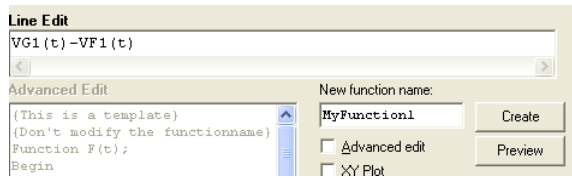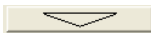
Change MyFunction1 to VRL then press the **Create** button. The VRL identifier will appear on the Curves to Insert list.
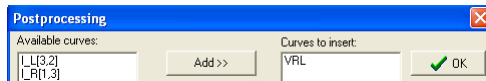
Press OK and the VRL voltage appears in the diagram Window.

GETTING STARTED

DesignSoft
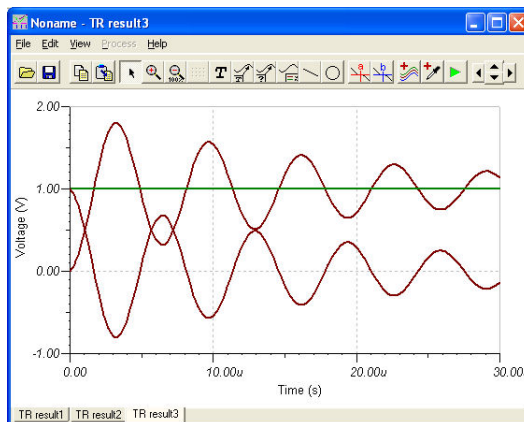
Now let's see how to modify the additional new curves. Press the ![icon] button again, the Post-processor window will appear, with the new VRL curve at the end of the 'Available curves" list. Click on VRL and press the ⌐⌐⌐⌐⌐⌐⌐ button, the definition of the curve will appear in the Line Edit field. However the New function name will be different, for example Myfunction3. Let's change this back to VRL. and change the VG1(t)-VF1(t) expression to something else, e.g. 2*(VG1(t)-VF1(t)). Press Create and OK and the new curve will appear in the diagram.

If you want to delete the new curve open the post processor again click on VRL and press the Delete button.

Note that if you save and reload the circuits with new curves added by the Post-processor, the new curve(s) will automatically be generated as long as they are present on the Post processor's "Available curves" list, and you can either edit these expressions or delete them in the Post-processor.
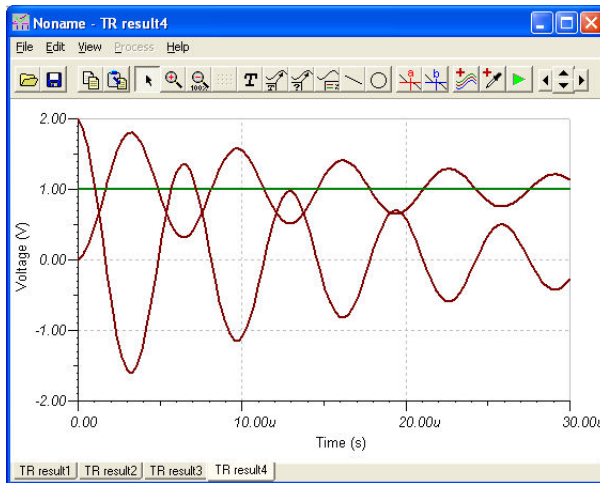
You can do a lot more with TINA's post-processing tool. For example, you can create curves of new functions created by adding or subtracting curves, or by applying mathematical functions on them. For a more detailed description, refer to the Post-processing analysis results section in the Advanced Topics chapter.

To demonstrate a more advanced feature of TINA, examine the **Fourier Spectrum** of the non-periodic transient response just obtained. First, in order to get a finer curve, select Analysis | Set Analysis Parameters… and change the "TR maximum time step" parameter to 10n. Next run the Transient Analysis for a longer 1ms time by changing the End display parameter in the Transient Analysis dialog to 1m.
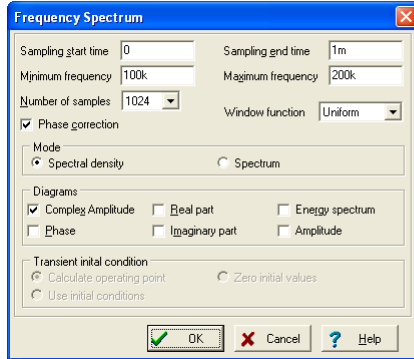
Select the damping output signal by moving the cursor over the curve and pressing the left mouse button when the cursor assumes a + form. The selected curve will change to red. Now press the right mouse button and select Fourier Spectrum from the popup menu. The Frequency Spectrum dialog box will appear. Set the Minimum frequency to 100k, the Maximum frequency to 200k and press OK.

The Fourier spectrum of the transient response will appear. The result is a continuous frequency spectrum shown in the figure below. As expected, the frequency where the Fourier Spectrum shows a maximum is the resonant frequency of the circuit.

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.
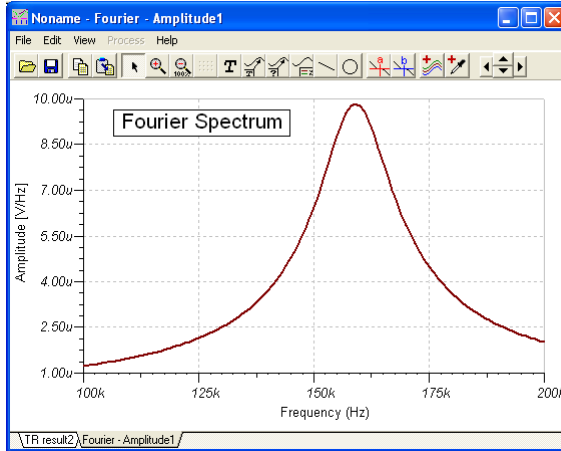
You might be surprised that the unit of the frequency spectrum is in V/Hz. That is because the continuous Fourier spectrum is a density function versus frequency. If you want to know the approximate amplitude in a narrow frequency band, you should multiply the average amplitude (given in V/Hz or Vs) with the bandwidth (given in Hz or 1/s).

You can also find the Amplitude in V directly, if you select Spectrum in the Mode field of the Fourier Analysis dialog. In this case the applied bandwidth is 1/DT, where DT is the length of the Transient analysis (End display - Start display). This feature is especially useful if your signal contains both non-periodic and periodic components. If your signal contains periodic components, you can display them in the diagram more accurately if you select a suitable *Window function* in the Frequency Spectrum dialog. For reading the amplitude from the diagram it is the best to use the Flattop window function.

Fortunately, Fourier analysis is not so complicated for clearly periodic signals. Periodic signals can be represented by Fourier Series or in other words as a sum of cosine and sine waves at the base (fundamental) frequency and integer multiples of the base frequency. To try out this kind of Fourier analysis in *TINA*, load **AMPLI.TSC** from the *EXAMPLES* folder.
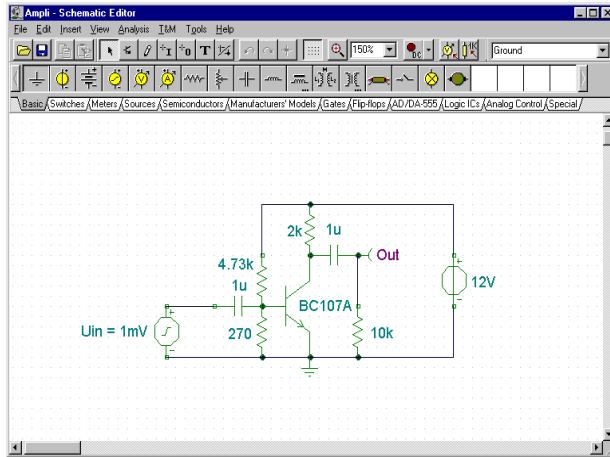
Run a transient analysis and then select the output curve with the largest amplitude. Press the mouse right button and select **Fourier Series** from the popup menu: the dialog box of the Fourier series will appear. Note that you can access this dialog directly from the Analysis. Fourier Analysis menu. Set Sampling start time to 1ms and the Number of samples to 2048. 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*. The list of Fourier components will appear.
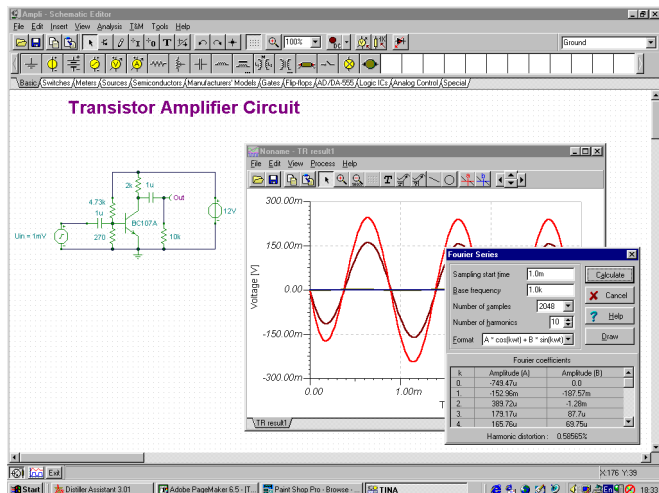
If you press *Draw* you can also draw a diagram showing the amplitudes in V (volts) at integer multiples of the base frequency.

4.6.2   Creating and analyzing an Op-Amp circuit

Create the circuit diagram using an OPA121E operational amplifier from Texas Instruments as shown in the following figure:
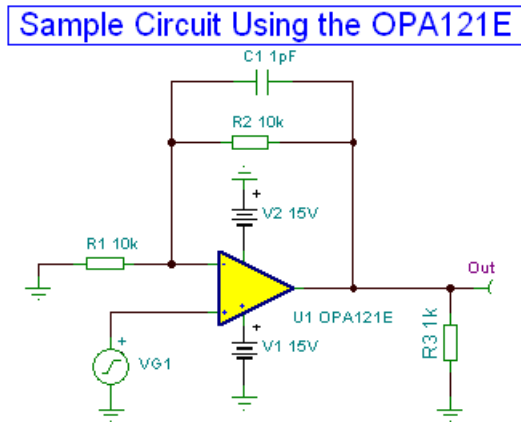
If you just opened TINA and wish to create your own circuit, you can start adding components right away. The circuit file name in the top line is set to Noname by default, indicating that a new circuit file Noname.TSC is being edited. If you already have a circuit loaded in the editor, for example, our previous RLC circuit, you can start a new one with the File|New command. You can switch between multiple circuits by clicking the Tabs at the bottom of the screen.

Now start adding components. Left click on the voltage generator icon then release the mouse button. The cursor will change into the generator symbol. Position it using the mouse (or by pressing the [+]/[Ctrl-R] or [-]/[Ctrl-L] key for rotation] or the [*] key for mirroring) somewhere in the middle of the screen, then press the left mouse button to drop the component into the schematic. We still need to set the properties of this generator. Double-click on the generator and the following dialog box will appear:
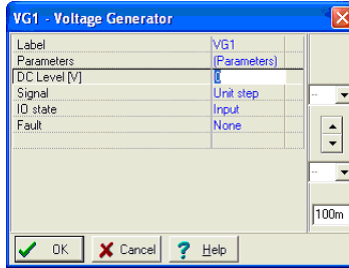
Leave the DC level, and the IO state parameters unchanged. Note that by accepting Input for the IO state parameter you have selected the output of this generator to be the input for this analysis (a Bode diagram in this example). Click on the Signal menu line. The dialog box will change as shown below:
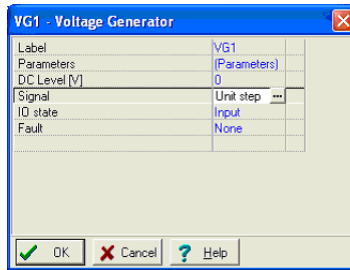
Press the button. A new dialog box with the graphic icons of available voltage generator signals will appear. When you select one of them (in our case, click on the Square Wave button), the associated curve comes up with some default parameters. In the case of the Square Wave signal, these are:

DesignSoft
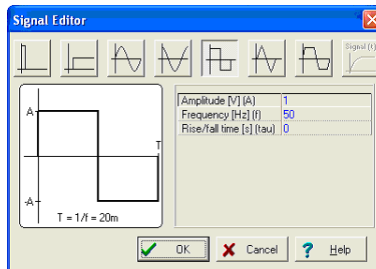
Change the Amplitude to 500m (this represents 500mV peak), the frequency to 100k (100kHz), and the Rise/Fall time to 1p (1ps). Click on OK and return to the previous dialog box and click on OK again. The program will automatically place the label (VG1) near the component and you will be able to position and place the component and the label together. If the default label position is not satisfactory, you'll be able to drag the label to the desired position later on.

Now click on the Spice Macros tab and press the left Operational Amplifiers button. The following dialog box will appear:
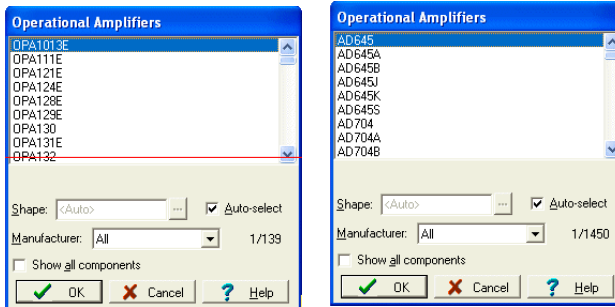
To find the IC we want, scroll down the list until you find OPA121E. You can narrow the list if you select the manufacturer (Texas Instrument in our example) from the Manufacturer listbox. You can also simply type OPA121E and the list will automatically jump to the IC (press the Delete button on the keyboard and try again if you make a typing error). Click on the line (OPA121E) and press the OK button. (Alternatively you can double-click on that line). The schematic symbol of this opamp will appear and be attached to the cursor. By moving the mouse, position the opamp as shown on the schematic at the beginning of the section and then press the left mouse button to place the opamp into your schematic.

You can also select a part using the 🖘 Find Component tool at the top-right corner of the Schematic Editor. If you type the part number into the "Component to find" field and press the Search button, the list of available component(s) will appear. (You can enter just part of the name if you are not sure of the entire name).

Press the Insert button to place the component. With the List Component button you can create the list of all available components in a textfile.
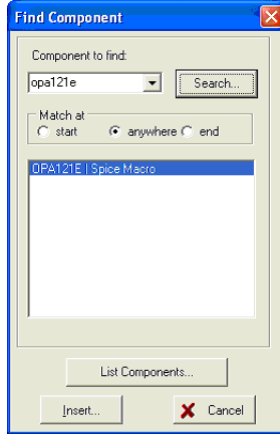
Note that other types of ICs are available under the buttons next to the Operational Amplifiers: Difference Amplifiers, Fully-Differential Amplifiers, Comparators, Voltage Regulators, Buffers, Current Shunt Monitors, and Other Components). You can bring all of these various components into the dialog box for any of the buttons if you set the Show All Components checkbox. In addition to selecting an IC on the list, you can also find it by clicking on any item on the list and then typing in the name of the IC.

Now click on the Basic tab on the Component bar and click the Resistor icon. The resistor symbol will be attached to the cursor. Move the resistor to the position of the R1 resistor on the sample schematic diagram at the beginning of this section and press the left mouse button to place this resistor into the schematic. Double-click on the resistor and the following dialog box will appear:

DesignSoft
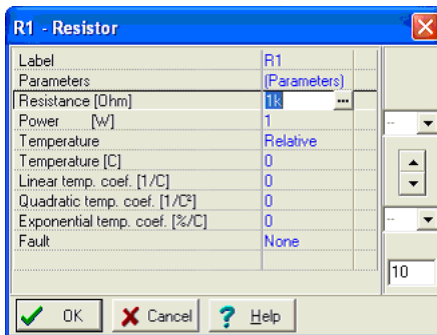
Change the value in the Resistance field to 10k, and press OK.

Note that you can set a component value before placement while you are moving it. To do this, press the right button of the mouse and select Properties on the popup menu. The dialog shown above will appear and you can set the properties of that component. After pressing OK, you can return to component placement.

Now let's place R2 at the top of the circuit. Click on the resistor symbol on the component toolbar, move and place the resistor. When you drop the resistor you will see that its value is already 10k since the program remembers the previous value.

Now let's place R3, which needs to be turned by 90 degrees. Click on the resistor symbol on the toolbar and turn the component by 90ş by clicking the ↶ or ↷ buttons or pushing the Ctrl L or Ctrl R keys. (+ and – on the numeric keyboard have the same effect). Place the component on the right side of the screen and set its value to 1k.

Continue circuit entry with the Capacitor, Battery, and Ground components as indicated in the figure above. Set the parameters to C =1 p, V1=15, and V2=15. Place a Voltage Pin Out (chosen from the Meters component group) at the right side of the new schematic. Pay attention to battery polarities and rotate the symbols if necessary.

NOTE:

If you have several opamps you may want to simplify their connection to the power supply. This can be done using the ⊣ Jumper component, which you can find at the first place on the Special toolbar.

All jumpers with the same label are considered as electrically connected components in TINA. Therefore, if you connect a jumper called VCC with the positive power supply of the opamp, it is enough to connect jumpers with the same VCC label to the positive power supply pins of the opamps. As an example, you can load and study OPAMP2.TSC from TINA's EXAMPLES\PCB folder, also shown below.
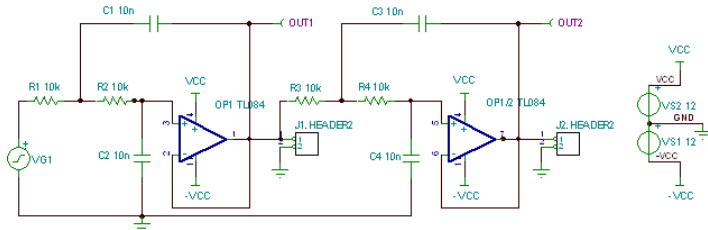
Note that even though all the computed voltages, currents and signals are available after running an analysis (see below in this chapter and also in the Post-processing analysis results section), you still need to define at least one output. We have placed the parts into the schematic but they are still unconnected. To connect devices, move the cursor over an appropriate pin node until a small drawing pen icon appears. When this pen appears, click the left button of the mouse, draw the wire, and left click again at its endpoint.

Finally, add the title to the schematic. using the $\mathbf{T}$ icon on the toolbar.

Let's check the circuit we have just built and run ERC from the Analysis menu. If everything is OK, the following dialog will appear:
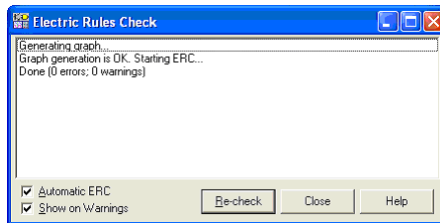
If there is a problem with the circuit, a list of warnings or error messages will appear in the dialog box. If you click on a warning or error message, the related part or wire will be highlighted in the circuit diagram.

4.6.2.1 Calculating DC Transfer characteristic

We have already seen several of TINA's analysis modes. But so far we have not used the DC analysis mode to calculate the DC transfer characteristic of this circuit. Select DC Analysis|DC Transfer Characteristic… from the Analysis menu. The following dialog box will appear:

GETTING STARTED
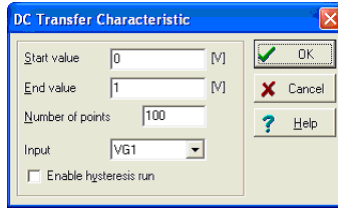
Set the Start value to –7.5, the End value to 7.5, and then press OK. After a short running time, a Diagram Window will appear as shown below. This displays the circuit's transfer curve-output voltage vs. input voltage.
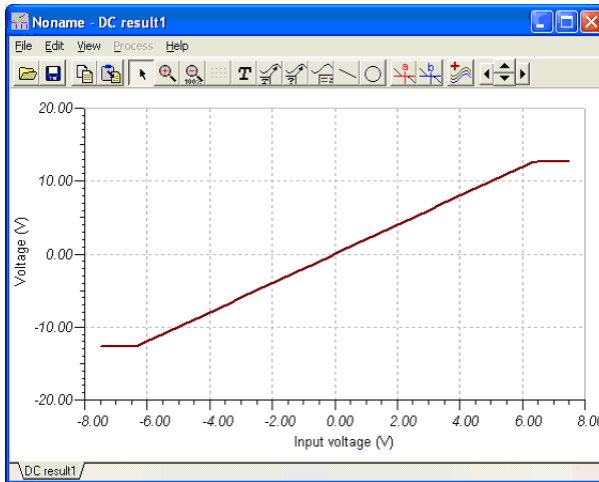
4.6.3   Analysis of SMPS circuits

SMPS or Switching-Mode Power Supply circuits are an important part of modern electronics. The heavy transient analysis needed to simulate such a circuit may take a lot of time and computer storage. In order to support the analysis of such circuits TINA provides powerful tools and analysis modes. In this Chapter we will demonstrate these through examples.

Using the Steady State Solver

The most time consuming part of an analysis of an SMPS circuit is to reach its steady state, when the DC level of the output voltage does not change and the output waveform has only a small periodic ripple.

To find this state automatically, TINA has a Steady State Solver under the Analysis menu.

To demonstrate this tool, let's load the Startup Transient TPS61000.TSC Boost Converter circuit file from the EXAMPLES\SMPS\QS Manual Circuits folder.
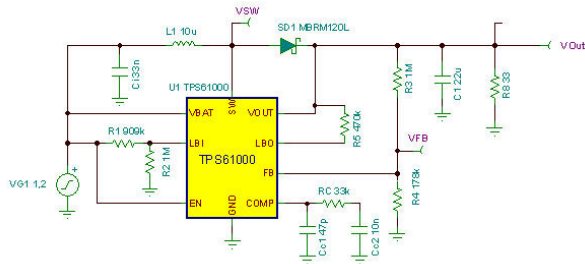
Select the Steady State Solver from the Analysis menu. The following dialog will appear:

DesignSoft
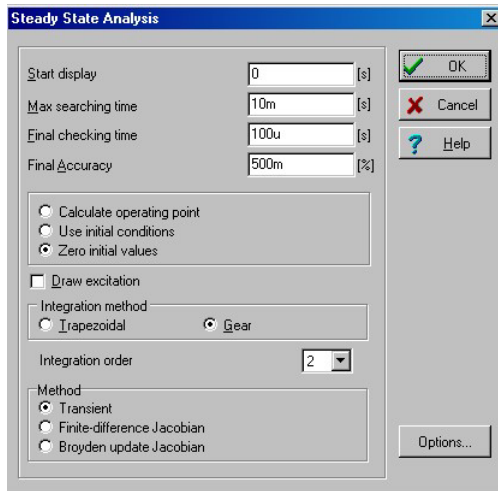
The new parameters compared to the Transient Analysis dialog box areas follows -

Max searching time: The solver will try to find the steady state solution for max 10ms. After this, the analysis will discontinued whether or not a solution was found..

Final checking time: After the steady state search is done, there is a final check for the length specified here. You should have a stationary waveform for this time interval.

Final accuracy: the maximum DC level change allowed. When the change is below this, the analysis will end. Note that the 500m in the example above means 0.1%

Method: You can select the method used for finding the steady state:

Transient: The steady state is searched by using transient analysis. *Finite-Difference Jacobian, Broyden update Jacobian:* The steady state is searched by the methods described in the paper, *Automated steady-state analysis of switching power converters* by Dragan Maksimovic.

Note that these last two methods may get to the steady state faster, but they do not go through the normal transient states, so the resulting waveform between the initial and final state does not reflect the real process (but rather the mathematical path of the methods to get there).

Now lets run the Solver. After a few minutes of running (approximately 2 minutes on a 2GHz Pentium computer) we'll get the following resulting waveform:
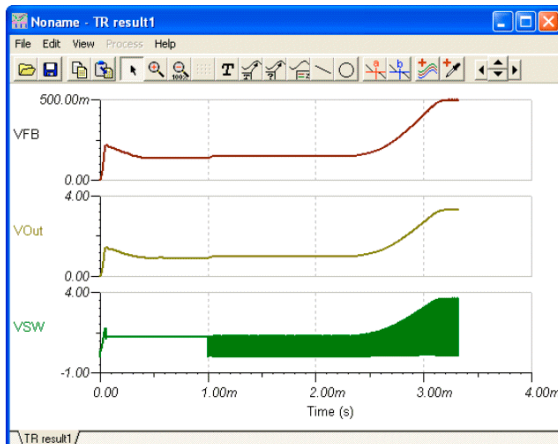
These waveforms show the detailed transient from switching on until reaching the steady output voltage. If you zoom out on the waveform, you can see that the period of the switching is around 500kHz and the time needed to arrive at the steady state is 4 milliseconds. Therefore, we need to calculate at least hundreds or sometimes thousands of periods if we want to see the whole transient waveform. This is why finding the steady state is such a time consuming process. The reason for this problem is the long start-up time of SMPS circuit compared with their switching frequency. The start-up time is basically determined by the filter capacitors on the output. The larger these capacitors are, the longer the start-up time.

NOTE:

> *In some cases you can accelerate the Steady State Search using the Finite-Difference Jacobian and Broyden update Jacobian methods, however they do not always converge and the intermediate waveforms provided by these methods do not reflect the real waveforms of the transient process.*

Trigger

Use this to determine the starting and ending times of the switching period.

You can find this component on the Meters toolbar of TINA. You should connect it to the oscillator frequency control pin of the SMPS/PWM controller IC, but any node where the oscillator waveform of the IC is present will do.

If you double-click on the Trigger component you can set its parameters.
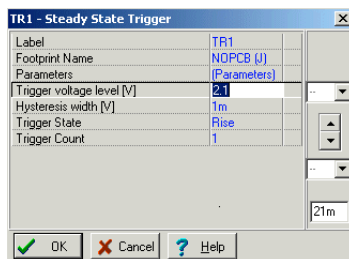
Trigger voltage level: the threshold voltage for the trigger event

Hysteresis width: hysteresis value for the trigger event. This value defines a region within which the trigger voltage is allowed to

oscillate without generating a trigger event.

Trigger State: Rise/Fall The direction of the voltage change required for a trigger event

Trigger Count: you can take several periods for the waveform analysis.

Once you have checked the initial transient and the steady state waveform and SMPS circuit the next thing you normally want to know is how it behaves when the input voltage or the load changes. This is realized by the **Input step** and the **Load step** analyses**.**

Sensor

The purpose of this component is to set the target voltage(s) to be watched during the steady state search. You can add more than one sensor to a circuit. By adding sensors, you can significantly accelerate the steady state search. You can make the search even faster if you can give the final voltage at a certain node.

Using the "Max. no. of saved TR. points" parameter in the Analysis/ Analysis parameters dialog, you can limit the maximum number of points placed in the Diagram. This is useful for large analyses to accelerate diagram drawing. By increasing this parameter, you can refine the diagrams but the drawing time will be greater.
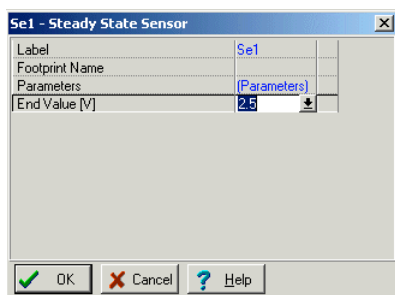
The only electric parameter of the Sensor component is -

End Value: Voltage|Not Used

Accelerating SMPS simulation using initial values

As we mentioned in the previous section, the long analysis time needed for reaching the steady state of SMPS circuits is mostly used for charging the output filter and some capacitors. If we start the analysis using initial values for larger capacitors and inductors, the analysis time can be significantly reduced. In TINA, the Steady State Solver will automatically place initial values into the model of larger capacitors and inductors and so the following Transient Analysis can be run significantly faster (assuming that we do not make changes which will need significantly different initial values). For example, if you want to study the effect of changing the output filter capacitor, it will not significantly change the output voltage DC level. Therefore, starting the new analysis with an initial value calculated by the steady state solver for another output capacitor, will result in a much faster analysis. You can accelerate the analysis of input and load changes in the same way.

To demonstrate this feature, let's run a transient analysis for our example. Selecting the Transient command from the Analysis menu, the following dialog box appears.
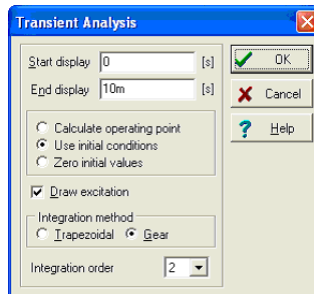
Note that Use initial conditions is set in the dialog. Press OK to start Transient analysis. You should see that analysis runs very fast compared to the previous steady state analysis. The output waveform is shown in the picture below.

DesignSoft
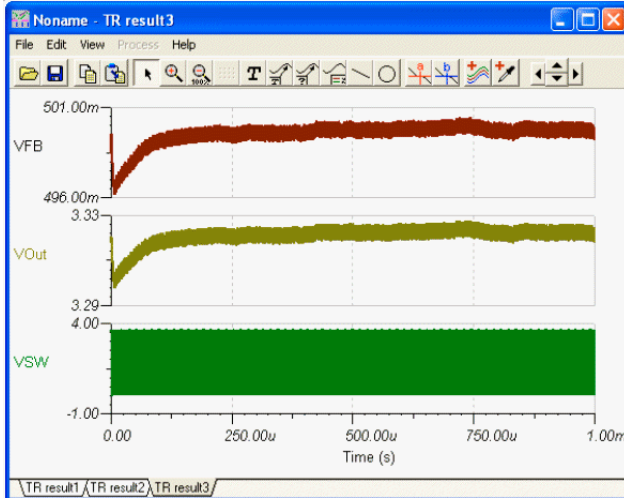
Why did the analysis run faster? Transient analysis was already preceded by the Steady State Analysis and the main capacitor's initial values (called Initial DC voltage in the Capacitor property box) was already set to the final DC voltage. For example , if you double-click on the C1 capacitor, you will see that the Initial DC voltage is already set to 3.31 V. Similarly, all the larger capacitor's initial values are set.

Once you have checked the initial transient, the steady state waveform, and SMPS circuit, the next thing you normally want to know is how it behaves when the input voltage or the load changes. This is realized by the **Input step** and the **Load step** analyses**.**

DesignSoft
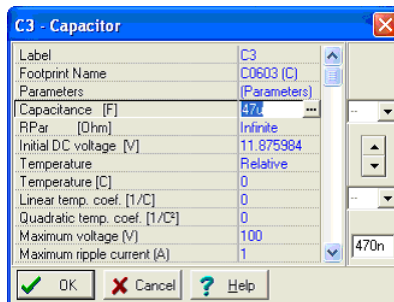
NOTE:

> *Using the "Max. no. of saved TR. points" parameter in the Analysis/ Analysis parameters dialog you can limit the maximum number of point placed in the Diagram. This is useful for large analyses to accelerate diagram drawing. By increasing this parameter you can refine the diagrams but the drawing time will be slower.*

Input step analysis

One of the standard analyses for SMPS circuits is the calculation of the response to an input change to test the capability of the SMPS design to regulate the output with step changes in the input line. This can be accomplished by adding a pulse to the input voltage and checking the output and other voltages. Since the input change is relative to the steady state, we can start it from the steady state initial values calculated by TINA's steady state solver.

Load the Input Step Transient TPS61000.TSC Boost Converter circuit file from the EXAMPLES\SMPS\QS Manual Circuits folder. The schematic design is the same as above.

To see the input step waveform, double-click on the VG1 voltagegenerator on the left. The following dialog box will appear:
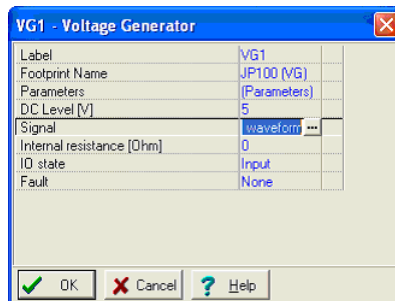
According this, the input voltage is 1.2V. This is converted by the SMPS circuit to 3.3V.

Now click on the Signal line of the above dialog and then the [...] button. The following signal in the Signal Editor will appear:

GETTING STARTED

DesignSoft
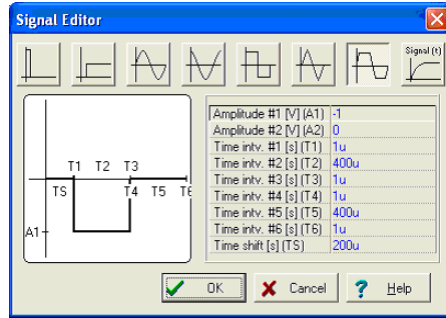
According to the waveforms, the input voltage will decrease from 1.2V to 0.8V for a T2=1ms time; and the starting edge (T1) and the ending edge (T3) of the pulse are 10us.

To see the response of the circuit, let's invoke and run the Transient analysis from the Analysis menu.
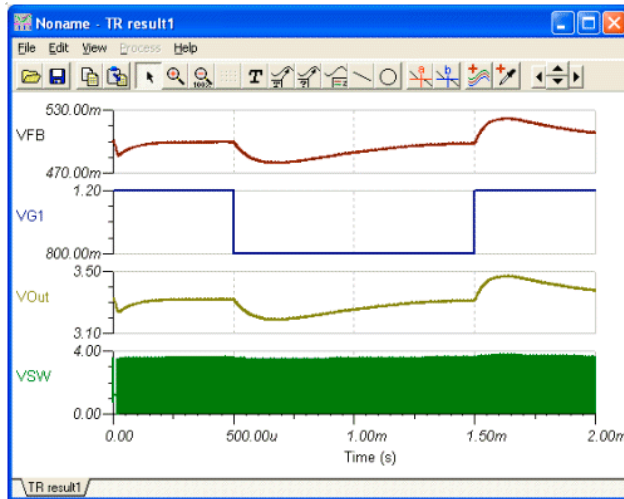
Load step analysis

Another standard analysis is to determine the SMPS response to a fast load change. Using simulation, the response to load changes is obtained by adding a current pulse to the load and analyzing the output and other voltages. Since the load change is relative to the steady state we can start it from the steady state initial values calculated by TINA's steady state solver.

Now load the example Load Step Transient TPS61000.TSC. The schematic design is the same as above , except for the ILoad Current generator on the output.

If you double-click on the ILoad generator and check its waveform, you will see that the DC part is
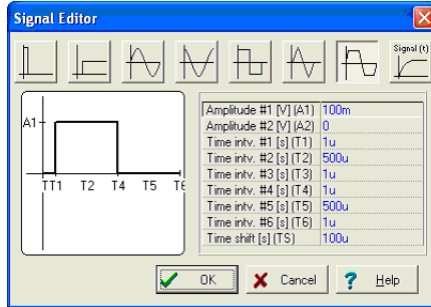
5mA while the pulse is 45mA in amplitude and 500us in width. Accordingly, the 5mA load current will rise to 50mA and then decrease to 5mA again.

Let's run Transient from the Analysis menu and see the result.
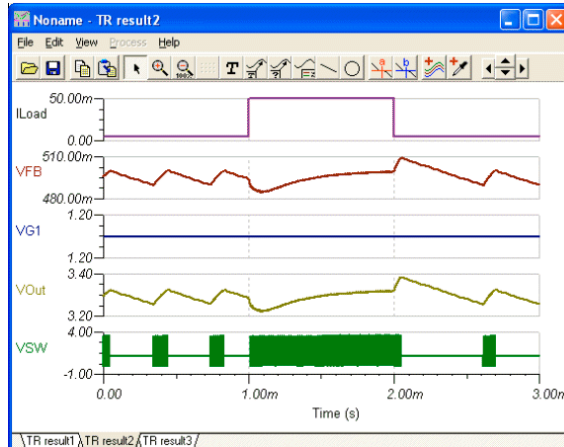
AC analysis

For AC analysis and stability analyses you can use the so called Average models provided in TINA. The average models represent a method, based on averaging the effects during the switching process. The resulting equations are linear therefore the method is extremely fast in order to draw Bode and Nyquist plots needed for stability analysis. Note that for using the AC analysis function of TINA you need an average model, the transient models are not applicable and will give improper results.

To demonstrate this tool, let's load the Average model TPS61000.TSC circuit file from the EXAMPLES\SMPS\QS Manual Circuits folder.
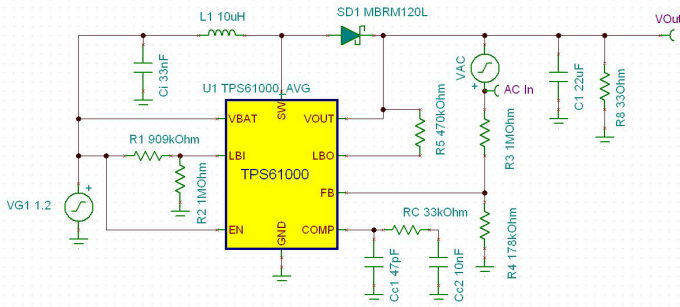
Note the VAC generator which is providing signal for the AC analysis., and the AC In Voltage pin which is the Input of the AC analysis (its IO state parameter is set to Input).

Let's run AC Analysis/AC Transfer Characteristic… from the Analysis menu and see the result.

DesignSoft
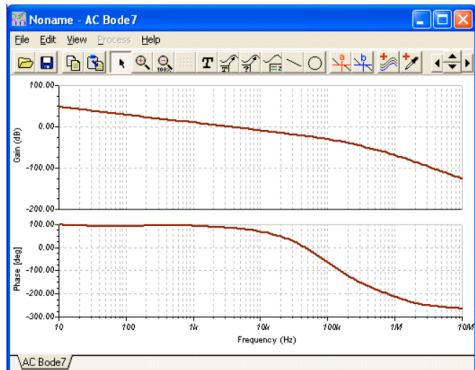
4.6.4   Stress Analysis

Stress Analysis can check parts for stress conditions such as maximum power dissipation and maximum voltage and current limits. You can set these parameters in the property window of the parts or in the catalog. This kind of analysis is also called Smoke analysis, because overloaded parts often emit smoke.

You can enable Stress Analysis by setting the Stress Analysis Enabled checkbox in the Analysis | Option dialog or on the Analysis menu.

When running DC or Transient Analysis from the Analysis menu, a list of components will appear, along with the parameters exceeding maximum limits.

If you click a component in the list, the corresponding component on the schematic diagram will be selected and turned red.

The maximum values of the components can be set in the component property dialogs or in the component catalog parameter dialogs. Both can be entered by double-clicking on the components. Before running an analysis, check and set the maximum values of the components in your circuit.

As an example of Stress Analysis, open the file Stress Analysis.TSC from TINA's EXAMPLES folder and run DC.Calculate Nodal Voltages and Transient Analysis from the Analysis menu or the corresponding interactive modes. In the following figure, you can see the result of Stress Analysis in DV interactive mode.
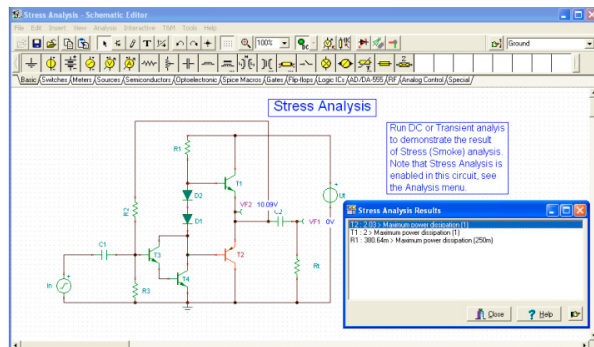
Apparently the power dissipation of T1, T2 and R1 exceed maximum limits allowed for these parts.

GETTING STARTED

DesignSoft

4.6.5   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. You can assign the two ports needed for Network Analysis with the Network Analyzer component to be found on the Meters component toolbar. As an example open the circuit EXAMPLES\RF\SPAR_TR.TSC.
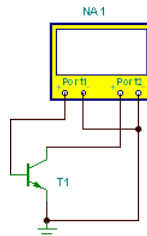
To analyze this circuit run Analysis/AC Analysis/Network Analysis. The amplitude diagram is as follows:
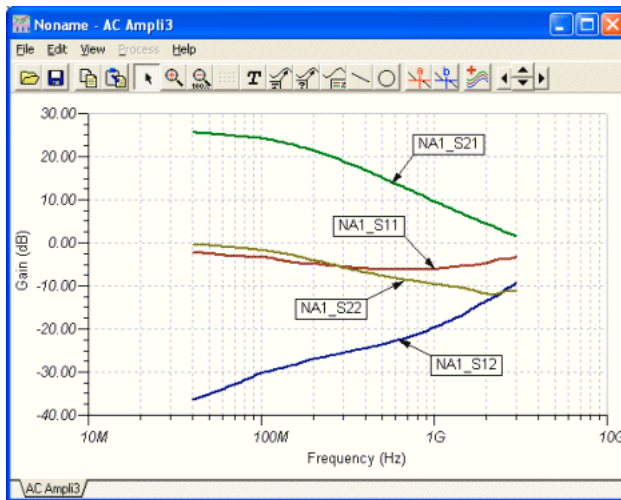
Note that we have added the labels to the curves using the ✍ Auto label tool of the diagram window. For more details on the Network Analysis see the "Network Analysisi and S-parameters" chapter of the advanced topics manual.

4.6.6 Analyzing a Digital Circuit with TINA's Digital Engine

Let's test a digital circuit. Open the file **HALF_ADD.TSC** from the EXAMPLES folder. Start the *Analysis|Digital Step-by-Step* command. A control panel will appear and you can examine the behavior of the circuit step-by-step by pressing the Step Forward button. Press the Play button for free-running mode. At each node a small box will indicate the logic level (Red for High, Blue for Low, Green for High Z, Black for undefined) as the circuit is clocked.

The picture below shows an intermediate state of the display.

Now let's examine the transient behavior of the circuit. Selecting the *Analysis|Digital Timing Analysis* command, brings up this menu:
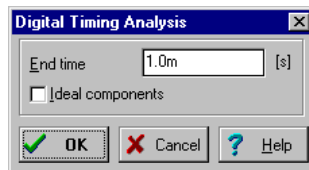
The result is shown on the timing diagram following.
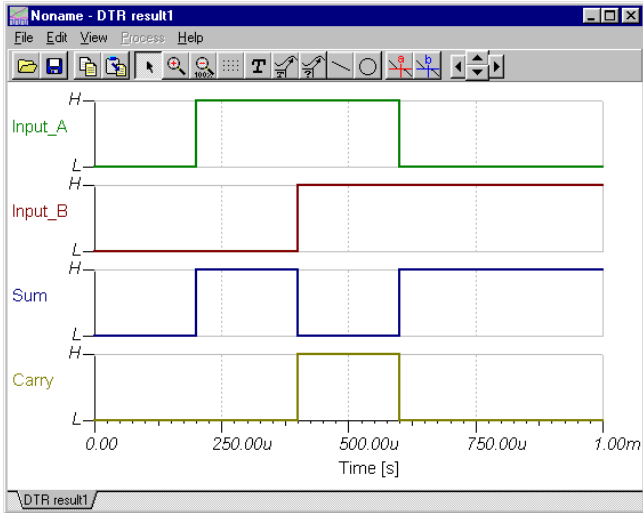
You could also select *Transient...* instead of *Digital Timing Analysis*, in which case the program would carry out an analog analysis, giving the detailed continuous waveforms and voltages instead of idealized logic levels. Note that circuits which contain only digital components can be analyzed by both digital and analog methods.

NOTE:

You can set the order of the curves by simply appending a colon (:) character and a number to the output name. This is particularly important when presenting the results of digital analysis, where each output is displayed as a separate diagram. For example, if you have outputs named OutA, OutB, Carry, and Sum, you can ensure that they will be displayed in the order given by using the labels OutA:1, OutB:2, Carry:3, and Sum:4.

The results of a purely analog analysis normally appear in one diagram: however, you can force TINA to display the results as separate diagrams, in the order you desire, by using the labeling method described above. You must use the View | Separate Curves command in the Diagrams window to separate the curves. If you don't use this labeling method, TINA presents the curves in alphabetical order.

4.6.7 Analyzing a Digital Circuits using Digital HDL Simulation Models

Hardware Description Languages (HDL) are standard text-based modeling languages used by electronic designers to describe and simulate their chips and systems prior to fabrication.

TINA now includes the four most widely used Hardware Description Languages defined by IEEE standards: VHDL, Verilog, Verilog-A and Verilog-AMS.

VHDL and Verilog are used for modeling digital circuits. The two languages are comparable in modeling digital hardware. However the behavioral capabilities of VHDL are more powerful, while Verilog is easier to learn and understand. In TINA you can use and mix models of both languages.

Verilog-A is an easy to read high-level behavioral language for modeling analog electronic circuits and devices (e.g., bipolar and MOS transistors).

Verilog-AMS is an extension of Verilog for modeling analog and mixed signal circuits allowing both Verilog and Verilog-A instructions, connect modules, and rules.

A full presentation of HDLs in TINA is beyond the scope of this manual. We refer the interested reader to the detailed standards, manuals and information on the Internet: **www.vhdl.org** and **www.verilog.org** .

In the following sections we will demonstrate the use of these languages through examples.

4.6.7.1 Analyzing a Digital Circuit Using Digital VHDL Simulation

TINA includes a powerful digital VHDL simulation engine. Any digital circuit in TINA can be automatically converted into VHDL code and analyzed as a VHDL design. In addition you can analyze a wide range of hardware available in VHDL and define your own digital components and hardware in VHDL. The great advantage of VHDL is not only that it is an IEEE standard hardware description language, but also that it can be realized automatically in programmable logic devices such as FPGAs and CPLDs.

GETTING STARTED

DesignSoft

TINA can generate synthesizable VHDL code along with the corresponding UCF file (User Constraints File for pin assignment within the FPGA), if the Generate synthesizable code checkbox is set in the Analysis/Options menu. You can save the created VHD and UCF files with the "Create VHD & UCF File" command in the T&M menu. You can read the files with the free Xilinx Webpack and generate the bit stream file describing the implementation of the design and then upload it to Xilinx FPGA chips.
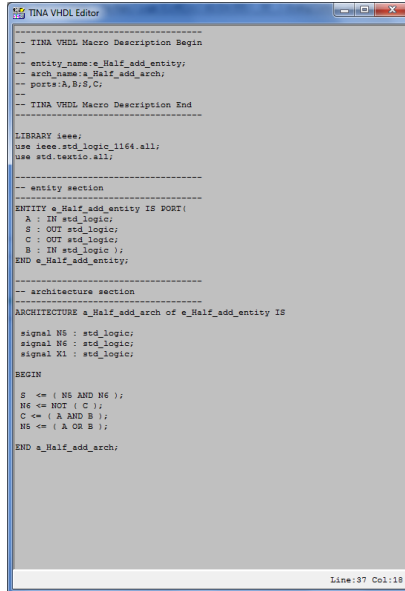
Before realizing a VHDL design, either with discrete components or with FPGA, verify it with simulation using TINA's Analysis|Digital Timing Analysis command. Let's examine some aspects of the VHDL simulation.

To do our first VHDL analysis, Open the FULL_ADD.TSC circuit from the EXAMPLES\VHDL folder. The following circuit will appear:
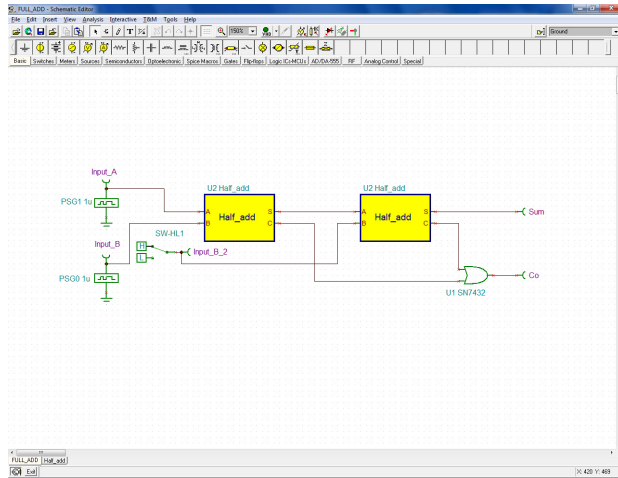
This circuit is a combination of two VHDL half adder blocks (macros) and a discrete OR gate.

If you double click on either of the HALF adder blocks and then press the Enter Macro button, the following window will appear:

DesignSoft

```
TINA VHDL Editor
--------------------------------------
-- TINA VHDL Macro Description Begin
--
-- entity_name:e_Half_add_entity;
-- arch_name:a_Half_add_arch;
-- ports:A,B;S,C;
--
-- TINA VHDL Macro Description End
--------------------------------------

LIBRARY ieee;
use ieee.std_logic_1164.all;
use std.textio.all;

--------------------------------------
-- entity section
--------------------------------------
ENTITY e_Half_add_entity IS PORT(
   A : IN std_logic;
   S : OUT std_logic;
   C : OUT std_logic;
   B : IN std_logic );
END e_Half_add_entity;

--------------------------------------
-- architecture section
--------------------------------------
ARCHITECTURE a_Half_add_arch of e_Half_add_entity IS

  signal N5 : std_logic;
  signal N6 : std_logic;
  signal X1 : std_logic;

BEGIN

  S  <= ( N5 AND N6 );
  N6 <= NOT ( C );
  C <= ( A AND B );
  N5 <= ( A OR B );

END a_Half_add_arch;

                                    Line:37 Col:18
```

Note that the essential VHDL code of the half adder is at the bottom and it is only

```
S  <= ( N5

AND N6 );

N6 <= NOT (

C );

C  <= ( A

AND B );

N5 <= ( A

OR B );
```

At first glance, the code may look a bit strange, but it in fact is a machine translation of our half adder, assembled from gates in 4.6.1. Introducing the node names N5 and N6 as shown on the figure below, it is clear that

DesignSoft

Half Adder

```
C  <=  ( A
AND B );
N6 <= NOT
( C );
N5 <= ( A OR B );
```
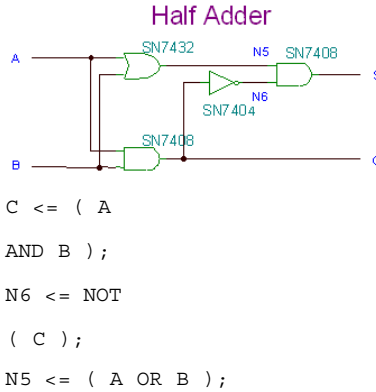
and therefore

```
S  <=  ( N5 AND N6 );
```

You might find it odd that in the VHDL code in the box, S appears to be calculated from N5 and N6 even before N5 and N6 have been calculated. This is valid, however, because VHDL is a concurrent language, and the order of the lines does not mean the order of execution.

The delays are taken from the given discrete values. But if the target hardware is an FPGA the synthesizer program will use the delay values of the FPGA data sheet.

Now select Digital VHDL Simulation from the Analysis menu and press OK. The following diagram will appear:

GETTING STARTED
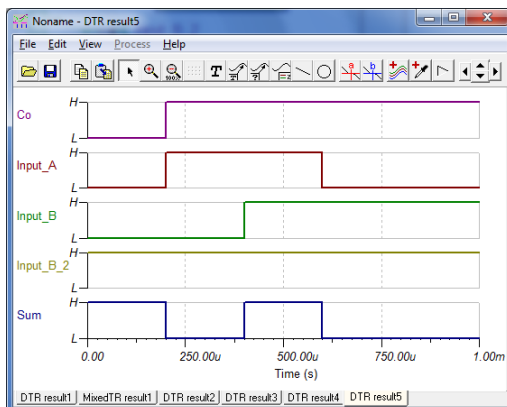
A great feature of TINA's VHDL is that you can not only view the VHDL code of each component, but you can edit and run them immediately. Let us replace the 4 line VHDL code -
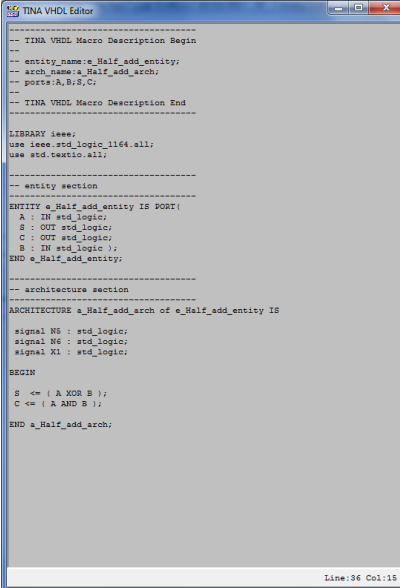
```
S <= ( N5 AND N6 ); N6 <= NOT ( C );

C <= ( A AND B ); N5 <= ( A OR B );
```

with this simpler 2 line code

```
S <= (A xor B); C <= (A and B);
```
This is easier to understand. In fact, if one of the A or B inputs is true, S is True. (A and B). We recognize this as an Xor function.

After editing the content of the VHDL blocks, they should look like this:

Now close the edit window by pressing on the Schematic Editor toolbar, select Digital VHDL Simulation from the Analysis menu, and press OK. The diagram that is drawn will be practically identical to the previous diagram.

NOTE:

> *In TINA of course you can make your own VHDL macros. This is decribed in chapter 5 under Making a VHDL macro from a .vhd file.*

4.6.7.2 The HDL Debugger: Debugging VHDL and Verilog codes

Debugging HDL programs is especially hard because of the concurrent processes in these languages.

A great feature in TINA is that the HDL debugger is now integrated. You can:

- Execute VHDL and Verilog codes statement-by-statement (Step)
- Execute subprograms as a single statement (Step Over)
- Add breakpoints (Toggle Breakpoint), running continuously (Start) and stopping at the breakpoints.
- Place variables, signals and other objects under the Watches tab and see their value during debugging.
- View all breakpoints and objects under the Breakpoints and Locals tabs at the bottom of the HDL debugger window.

To practice the use of the HDL debugger in TINA, open the vhdl_counter.TSC file from the EXAMPLES\VHDL folder with the Open command of the File menu. Next, click the Analysis menu and enable the debugger by clicking the Enable HDL Debugger line. Finally press, the DIG button on the toolbar at the top of the screen or click Start on the Interactive menu. The HDL Debugger will appear. Click the counter.vhd tab at the bottom of the code.

You should see the following screen:

DesignSoft
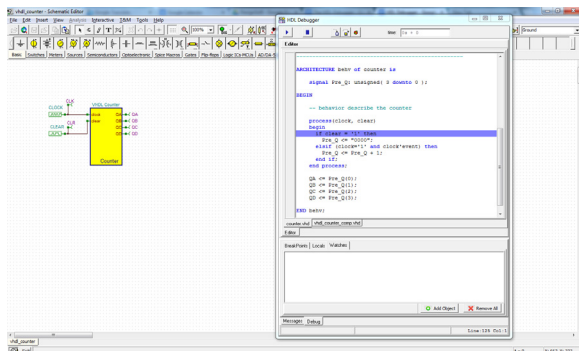
You will see two modules under the Tabs *counter.vhd* and *vhdl_counter_comp.vhd*. The first is the contents of the *VHDL Counter* macro, while the second file is the VHDL conversion of the whole circuit including the igitl sources.

This macro implements a counter. The counter entity consist of five processes, and all processes run in parallel. The first process is sensitive to the clock and the clear signal. So when one of these signals is changing, this process is triggering and executing. The other processes are sensitve to the Pre_Q signal. When Pre_Q(i) changes, the *i+1th* process is triggering and executing.

To follow some important steps in the program, let's add four breakpoints by clicking the lines of interest and pressng the Toggle Breakpoint button. The lines with the breakpoint will be marked by a red background:
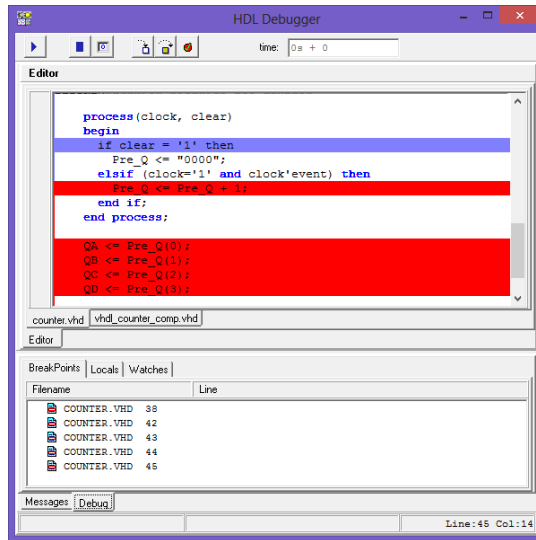
Note that under the Breakpoints tab you can see all the breakpoints and remove any of them with the Toggle Breakpoint button.

Now click the Watches tab, press the Add Object button and one-by-one add the Pre_Q signal and the QA, QB, QC, QD ports. These will be connected with the 4 outputs of the macro.

You will see the following in the HDL Debugger window:

DesignSoft
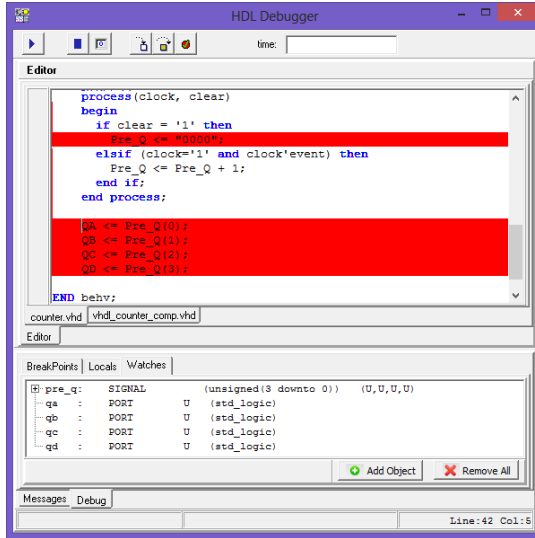
Now let's start debugging by pressing the Run button.

In the VHDL simulation each process runs once at the start of the simulation. Click the Run button several times until the debugger shows *500ns+1* in the *time* field. It means the simulator reaches 500ns and 1 delta cycle (Delta cycle is a special VHDL time period of infinitesimal duration). At this time clock='1', Pre_Q was initialized with '0'-s at *clear*. The line Pre_Q <= Pre_Q+1 will schedule a transaction on signal Pre_Q with value '0001' for time 500ns+2.

Press Run again and the debugger will stop at breakpoint QA <= Pre_Q(0). At this point time=500ns+2, because the nearest event was the previously described scheduled event. Now the simulator will schedule an event which will assign the value '1' for time=500ns+3 for the QA port.

Press Run again, now time=1.5us+1 and QA='1'. Note, the last three processes are now not triggered because there were no changes in their sensitivity list.

You can study updating the other ports in a similar way. There is a similar example in Verilog called verilog_counter.tsc in the Examples\Verilog folder.

4.6.7.3 **Analyzing a Digital Circuit Using Digital Verilog Simulation**

TINA also includes a powerful digital Verilog simulation engine. The advantage of Verilog compared to VHDL that it is easier to learn and understand, however there are more features in VHDL.

Verliog -similarly to VHDL- can also be realized automatically in programmable logic devices such as FPGAs and CPLDs.

TINA translates the Verilog models and the other digital components to synthesizable VHDL code along with the corresponding UCF file (User Constraints Fiile) for pin assignment within the FPGA), if the Generate synthesizable code checkbox is set in the Analysis/Options menu. You can save the created VHD and UCF files with the "Create VHD & UCF File" command in the T&M menu and, using the free Xilinx's Webpack software, generate the bit stream file describing the implementation of the design and then upload it to Xilinx FPGA chips.

Before realizing a Verilog or any other digital HDL design, either with discrete components or FPGA, you need to verify it with simulation with TINA's Analysis|Digital Timing Analysis command.

Let's run the previous VHDL circuit along with its Verilog model. the Full adder Verilog and VHDL comparison time diagram.TSC circuit from the EXAMPLES\Verilog folder. The following circuit will appear:

DesignSoft
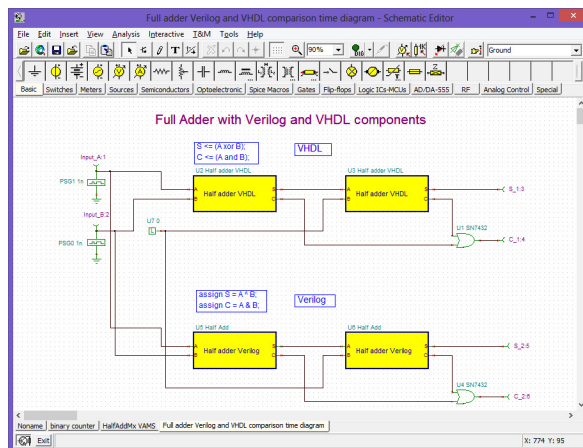
You can see realization of the half adder function in both languages, they are very similar. You can double click the VHDL or the Verilog macros and press Enter Macro to see all the details.

Now run the Digital Timing Analysis from the Analysis menu. The following diagram will appear:
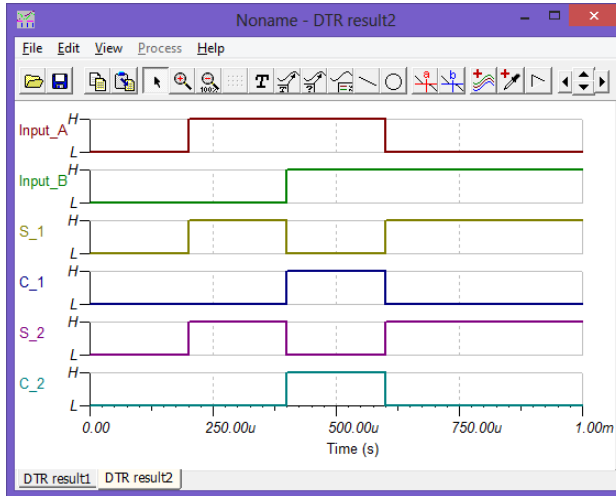
You can see that the output signals from both models are exactly the same.

4.6.7.4 Analyzing Circuits Using Verilog-A models
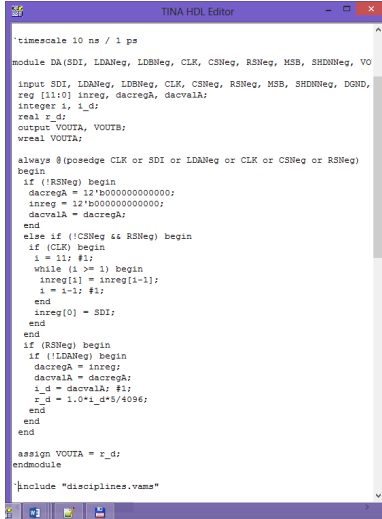
Today the most widely used language to describe electronics circuits and device models is the Spice netlist format (1973). However the Spice netlists are often hard to read and understand, and they lack a lot of the functionalities of programming languages which engineers would need while creating models and simulation.

The relatively new Verilog-A language (1995) provides an alternative method with an easy to read programming language style C like syntax. Thus Verilog-A is a suitable successor of the SPICE netlists for describing circuit topologies.

Most of the device libraries of TINA are in Spice netlist format. However you can already create and import models and place TINA macros in Verilog A format. You can find several language examples, device models, and circuits in the Examples\Verilog A folder of

TINA. Creating macros in Verilog A is described in Chapter 5 of this manual.

For a demonstration of Verilog-A in TINA, load the examples in the EXAMPLES\Verilog-A\Device Models folder.

We suggest that you start with the Opamp Model Comparison.TSC file where a simple opamp model is realized in three different ways: Verilog-A, Spice, and the schematic diagram.

You can also study nonlinear device models in Verilog-A and their characteristics in the other examples: diode.TSC, JFET.TSC etc.

4.6.7.5 **Analyzing Circuits Using Verilog-AMS models**

An even more sophisticated method of describing electronics circuit, containing both analog and digital components is the Verilog-AMS language. As we observed earlier, Verilog-AMS is a derivative of the purely digital Verilog extended with the purely analog Verilog A and an interface for the connection of the analog and digital parts.

In TINA you can also create or import Verilog AMS macros for modelling mixed signal devices.

Let's see the structure of such a model. Open the DAC VAMS.TSC circuit from the EXAMPLES\Verilog AMS folder. The following circuit will appear.

DesignSoft

This circuit contains a Digital Analog Converter (DAC) macro with Serial Peripheral Interface (SPI) and a test bench macro, generating the digital SPI signal.The DAC model is defined in Verilog AMS. Interestingly, test bench on the left side is written in VHDL which is an example of mixing different HDLs but here we will concentrate on the Verilog AMS macro on the right.

To see the Verilog AMS code of the model double-click the DAC macro and press the Enter Macro button. The following window will appear.

We will not go into a detailed analysis of the code. We just want to show that in the first part shown above, the DA Verilog module converts the serial signal into an analog signal (VOUTA).

At the end of the macro shown below (you can scroll down there), the DA module is called and the signal is smoothed by a simple opamp and an RC filter using Verilog A instructions. You can also see the definition of the capacitor in the code fragment below.

```
module cap(p,n);
 inout p,n;
 electrical p,n;
 parameter real c=1e-9 from [0:inf];

 analog
    I(p,n) <+ ddt(c*V(p,n));
endmodule

module opamp(vout, vin_p, vin_n);
      inout vin_p, vin_n;
      inout vout;
      electrical vout, vin_p, vin_n;
      parameter real gain = 1;

      analog
            V(vout) <+ gain*V(vin_p, vin_n);
endmodule

connectrules MyRules;
 connect a2d_TTL input electrical, output logic;
 connect d2a_TTL input logic, output electrical;
endconnectrules

module macro1(SDI, LDANeg, LDBNeg, CLK, CSNeg, RSNeg, MSB, SHDNN
 input SDI, LDANeg, LDBNeg, CLK, CSNeg, RSNeg, MSB, SHDNNeg, DGN
 inout VOUTA, VOUTB;
 wire i1;
 electrical VOUTA, i2, i3;
 ground gnd;

 DA U1( .SDI(SDI), .LDANeg(LDANeg), .LDBNeg(LDBNeg), .CLK(CLK),
 res #(.r(1k)) R1 (i1, i3);
 cap #(.c(1n)) C1 (i3, gnd);
 v_dc# (.dc(0)) V1(.vp(i2), .vn(gnd));
 opamp #(.gain(1)) U2 (VOUTA, i3, i2);
endmodule
```

Line:109 Col:10

GETTING STARTED

4.6.8   Mixed Mode Simulation (Spice - VHDL - MCU co-simulation)

TINA version 8 and above include a new powerful mixed mode simulation engine. It is based on the XSPICE mixed mode algorithm, extended with MCU and VHDL components. In your circuits you may freely mix any analog or digital components of TINA, including microcontrollers (MCUs) and macros with Spice or VHDL content.

You can modify these components on the fly along with the code in the MCUs. TINA will analyze the analog parts in analog, the digital parts in digital, and will automatically create the interfaces among the components. This ensures synchronization and fast convergence.

Let's explore some of the uses of this mode through a few examples.

DesignSoft

4.6.8.1 **Waveform generation with a VHDL and Spice subcircuits**

The following circuit (EXAMPLES\VHDL\Mixed\Wave generator.TSC) generates an analog sine or sawtooth signal depending on the status of the left SW-MODE switch.

Digital wave generator
VHDL subcircuit

DA converter

Low pass filter
SPICE subcircuit
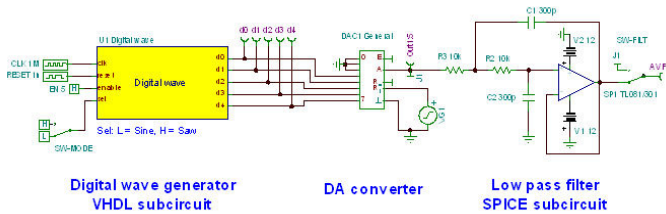
The Digital Wave box on the left of the circuit includes a VHDL code with a lookup table *Sine_LUT* for the sine wave and a counter for the sawtooth signal. The essential part of the VHDL code is:

```
process(Reset, Clk)
begin
      if (Reset = '1') then
      Wave <= (others => '0');
      LUT_index <= 0;
   elsif rising_edge(Clk) then
      if (Enable = '0') then
      Wave <= (others => '0');
   elsif (Sel = '0') then
      Wave <= Sine_LUT(LUT_index);
  else
      Wave <= conv_std_logic_vector(LUT_index,5);
  end if;
               if (LUT_index = LUT_index_max) then
               LUT_index <= 0;
      else
               LUT_index <= LUT_index + 1;
      end if;
      end if;
end process;
  d0 <= Wave(0);
  d1 <= Wave(1);
  d2 <= Wave(2);
  d3 <= Wave(3);
  d4 <= Wave(4);
```

You can see all the details of the code and modify it if necessary by double-clicking the Digital Wave box and pressing the Enter Macro button on its property dialog.
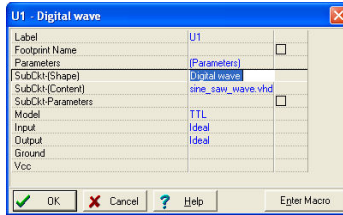
Note that the model is set to TTL in this dialog, but you may select from various other models (CMOS, LS, HC, HCT etc.).

The digital output of the counter is converted into an analog signal in the 5 bit DA converter of TINA shown in the middle of the circuit.

The DAC sine wave output needs to be cleaned up with a low pass filter. We will use a Spice opamp model of the TL081 in a Sallen and Key low pass filter configuration. Press the Enter Macro button on the property dialog and TINA will open the macro. You can review and, if necessary, modify the Spice code inside the macro.

The sawtooth signal from the counter output (on pin J1) does not need to be filtered, so we will connect it directly to one terminal of switch SW_FILT. The sine wave developed at the DAC output (DAEX) does, in fact, require filtering, so we will pass it through the low pass filter and connect the filtered Aout analog output to the other terminal of SW_FILT. A jumper (J1) connects the DAEX output to the switch. Although it's not obvious in the schematic, the switches SW_FILT and SW_MODE are synchronized as though they were a DPDT switch. We cause them to be synchronized by assigning both switches to be controlled by the Hotkey A. See the property dialog for SW_FILT where the Hotkey has been assigned to A:

GETTING STARTED

DesignSoft
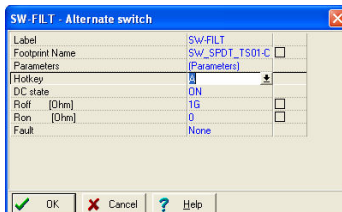
Here are the final waveforms of the full circuit, including the five counter output waveforms. SW_MODE is in the High state, selecting the sawtooth signal.
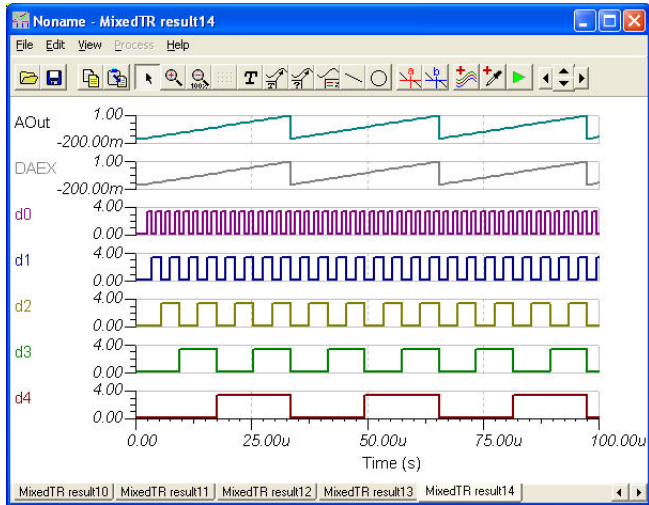
If we change the SW-MODE switch to Low and run Transient analysis again, the waveforms are:

DesignSoft
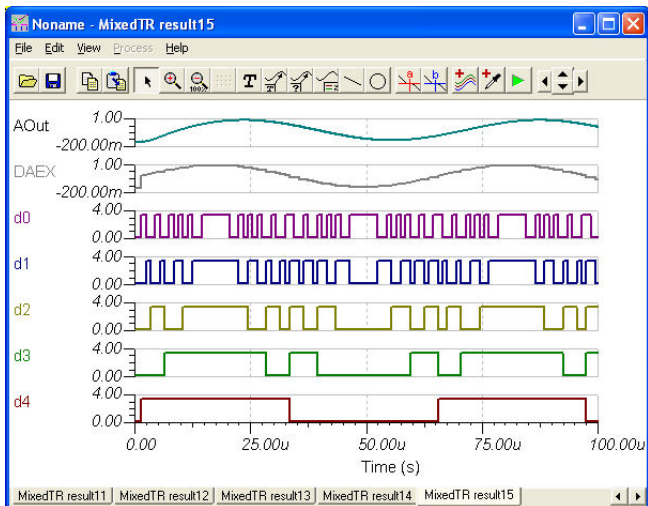
To see the effect of the analog filter, delete curves d0 to d4 from the diagram by clicking the curves and pressing the Del key. Alternatively, you can delete outputs d0 to d4 temporarily and run Transient Analysis again.
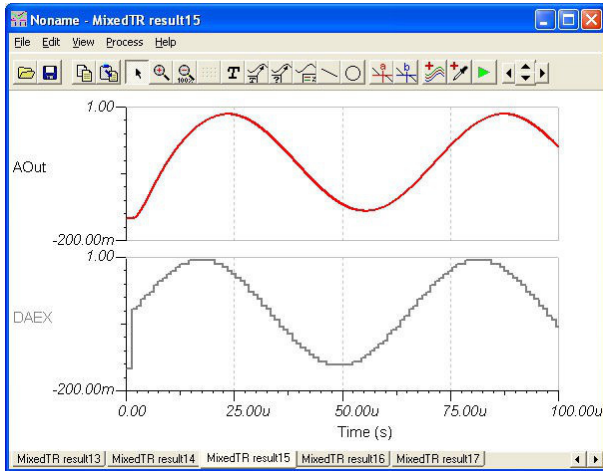
To demonstrate the flexibility of TINA's VHDL features, we'll modify the VHDL code to generate a square wave instead of the sawtooth waveform. Simply set Wave(0) to Wave(3) to zero in the VHDL code.

Double-click the Digital Wave macro and press the Enter Macro button. Locate the Wave <= conv_std_logic_vector(LUT_index,5) line and insert the following statements:

```
Wave(0)  <=  '0';
Wave(1)  <=  '0';
Wave(2)  <=  '0';
Wave(3)  <=  '0';
```

You can update the macro by simply closing the editor (click the x button in the top-right corner of the window). The following message will appear:

DesignSoft
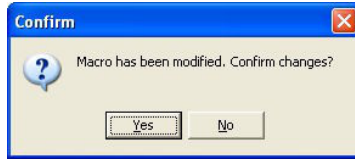
Press the Yes button to approve the changes.

Now you can run Transient or Mixed VHDL Simulation from the Analysis menu to get the following waveforms.
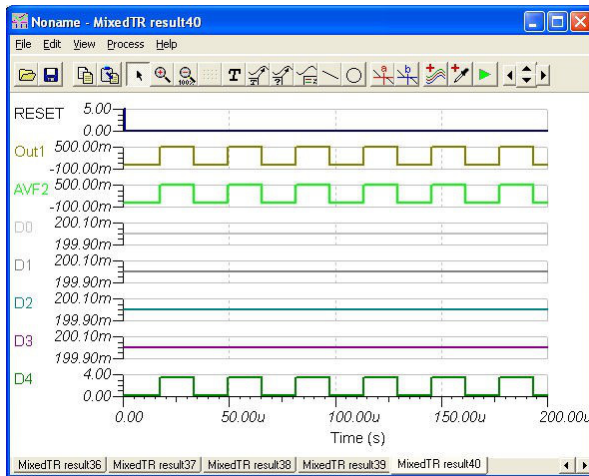
You can check out a more complex version of this circuit under EXAMPLES\VHDL\Mixed\Wave generator dipsw.TSC. There you can select all the three waveforms we discussed using a dip switch.

Note that you can download the VHDL portion of the code into an FPGA and use hardware form.

4.6.8.2 MCU controlled SMPS circuit

The mixed mode simulator of TINA not only allows MCUs, but also any linear or nonlinear parts in TINA's libraries. As an example, let's study the following circuit, which realizes a DC-DC converter, converting 5V DC to 13V DC, and operating in boost mode. You can find this circuit in TINA under EXAMPLES\Microcontrollers\Pic\ Boost_converter.TSC .
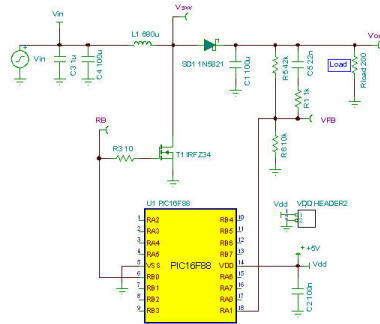
The PIC MCU in the circuit produces a PWM output at pin RB0 that controls the switching FET. The interrupt routine of the code in the PIC compares the feedback voltage at VFB (connected to pin RA1 of the PIC), with a built in threshold voltage. If the voltage is lower than the threshold defined in the code, the duty cycle of the PWM output waveform is increased. You can study the ASM code in the PIC by double-clicking on the PIC, clicking on the MCU line, pressing the [...] button (the little button with three dots in a row), and finally pressing the Edit ASM button. You can see and debug the code on the fly here. Click on the Enable MCU code debugger line under the Analysis menu, press the TR interactive transient analysis button (or select it on the Interactive menu), and, finally, click on Start.

The waveforms below demonstrate how the analog parts and the MCU interact in TINA.
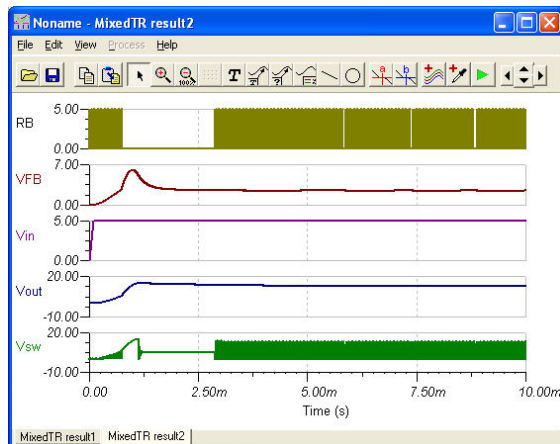
4.6.9   Testing your Circuit in Interactive mode

When everything is in order, the ultimate test of your circuit is to try it in a "real life" situation using its interactive controls (such as keypads and switches) and watching its displays or other indicators. You can carry out such a test using *TINA's* interactive mode. Not only can you play with the controls, but you can also change component values and even add or delete components while the analysis is in progress. The interactive mode is also very useful for educational and demonstration purposes, for tuning circuits interactively and for interactive circuits which you cannot test otherwise, e.g., circuits with switches, relays, or microcontrollers. First select the interactive mode required (DC, AC, TR, DIG or VHDL) with the ⏷ button, then press the 🔘 button. XX can be DC, AC, TR, VHD etc. depending on the mode, set by the ⏷ button. You can also select the required interactive mode with the DC, AC, Transient, …VHDL commands of the Interactive menu. You can start the interactive simulation with the Start command of the Interactive menu and stop it with the Stop command (The Start command will turn into Stop when the interactive simulation is started). Now the displays and indicators in your schematic will reflect whatever you do with the controls. In addition to displays, *TINA* has special multimedia components (light bulb, motor, LED, switch, etc.) which respond with light, motion and sound. Let's see a few examples.
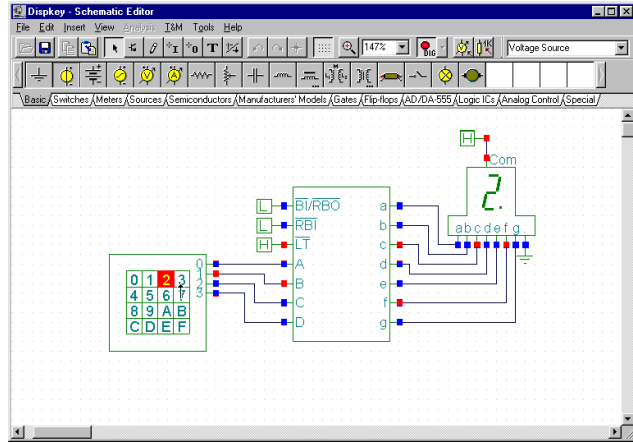
4.6.9.1 Digital Circuit with a Keypad

To try out the interactive mode, load the **DISPKEY.TSC** circuit from the *EXAMPLES\Multimedia* folder. The circuit is shown below. Select the *Digital* mode using the button, and then press the DIG button (the button will turn light green).

NOTE:

> *1) You can also select the Digital interactive mode with the Digital command of the Interactive menu. You can start the interactive simulation with the Start command of the Interactive menu and stop it with the Stop command.*

> *2) TINA can store the last Interactive mode in circuit files, so most likely the DIG mode is already set.*

Now you can play with the keypad and watch as the 7 segment display shows the setting of the keypad. If you have a soundcard on your PC, you will even hear the key clicks of the pad.

DesignSoft

GETTING STARTED

4.6.9.2 Light Switch with Thyristor

Open the Thyristor switch example, TSC circuit from the EXAMPLES folder and press the button button. You will see the following screen:
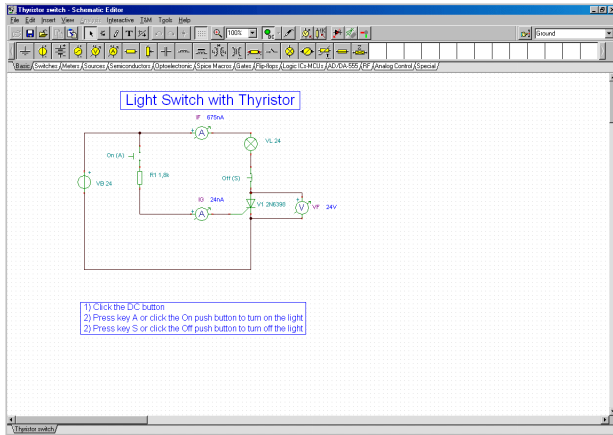
Press key A or click the On push button (Wait until the cursor turns into a vertical arrow) to turn on the light. The Thyristor will turn on and remain on even after the push button is released. So will the light. You can turn off both the Thyristor and the light bulb by pressing the key   on the keyboard or clicking on the push button . In both states of the circuit, you will see the currents shown by the two ammeters.

4.6.9.3 Ladder Logic networks

Another version of a self holding circuit, this one based on ladder logic, can be found in the LADDERL.TSC circuit file in the EXAMPLES/Multimedia folder.

Initially the red LED will light. If you click on the START button (click when the cursor changes into a vertical arrow), OCR1 will close and stay closed (since the current flowing through OCR1 will keep magnetizing the relay coil CR). Now the green LED will light, OCR2 will open, and the red LED will turn off. If you now click on the STOP button, you will break the self holding circuit and the relay

CR will release, the red LED will light again, and the green LED will turn off.

You can make it easier to operate switches if you "assign" them to "hotkeys" on the keyboard (your PC's keyboard). Double-click on a switch when the cursor has turned into a hand symbol. To assign a hotkey, select a letter or number on the list at the Hotkey field of the property dialog of the switch.
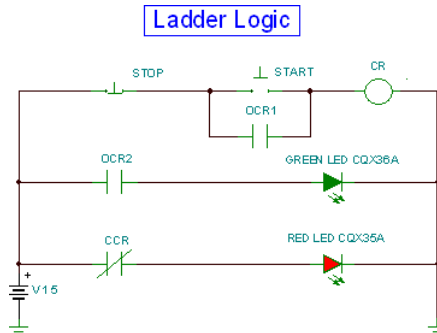
Ladder logic: Initial state or after clicking the STOP button.
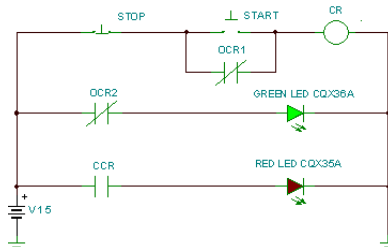
State after clicking the START button

4.6.9.4 HDL Circuits

A great feature of TINA is that you can not only test but also modify HDL circuits on the fly including, the HDL code itself. Let's see this with the example Calculator_ex_8.TSC in TINA's Examples\VHDL\Interactive folder.

This is a special calculator circuit controlled by the Opcode keypad.
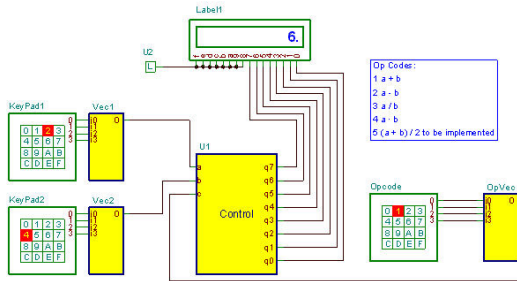
For the Operation codes 1, 2 , 3 and 4, it realizes a basic four func- tion calculator, complete with +, -, /, and * basic arithmetic opera- tions. Further operations can be added through modifying the VHDL code inside the Control unit. First press the [VHD] button; as the Opcode is 1, you should see 4+2=6 on the LCD display. Try the other Opcodes with different settings on KeyPad1 and KeyPad2.

Now let's implement the operation to be assigned to Opcode 5. Double-click on the Control box and press Enter Macro. The VHDL code of the component will appear.
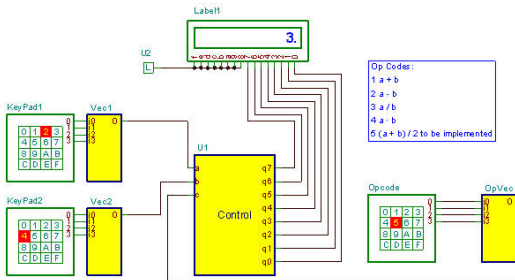
4.6.9.5 Microcontroller (MCU) Circuit

To test circuits with programmable devices requires special development software that permits a high degree of interactivity. This calls for debugging software that can test the code running in the device step-by-step.

You can see, modify, and debug the program running in any of the supported processors, and, of course, you can make and run your own code.

There are 4 ways of providing the program for microcontrollers in TINA.

You can:

1) use the binary code and debug file made by any standard compiler (e.g., MPLAB for PICs),

2) load your assembly code to run and debug directly in TINA using its built in assembler-debugger,

3) write your MCU code in C, install a C compiler which generates the code for the MCU you want to simulate, (TINA will automatically integrate it into its C code debugger),

4) or finally use the built in Flowchart editor in TINA to generate and debug the MCU code.

To load the code into the MCU, double-click on the schematic symbol.

One of the following dialogs will appear:
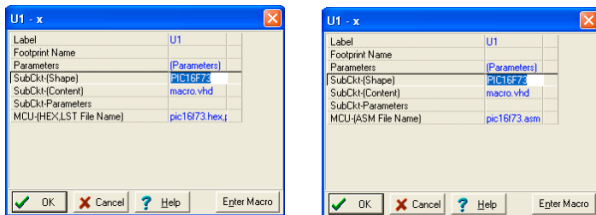
Click on the last MCU File name line and press the ▪▪▪ to proceed. The following dialog will appear:

DesignSoft
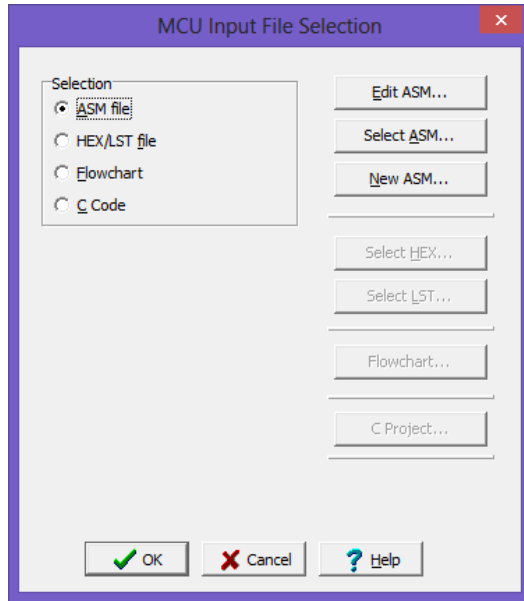
Here you can see and edit the ASM code in the MCU, select another ASM code file, or create a New ASM directly in the editor that will appear when you press the New ASM button.

If, however, you switch to the Use HEX/Lst file option, you can select the binary (HEX) file you want to run and the LST file to be used for debugging, as shown in the dialog below.

NOTE:

> *The HEX and LST files should be generated by an appropriate compiler (normally provided free by the MCU manufacturer. However, TINA has a built in compiler for all supported MCUs, so you can directly use your ASM source code.*
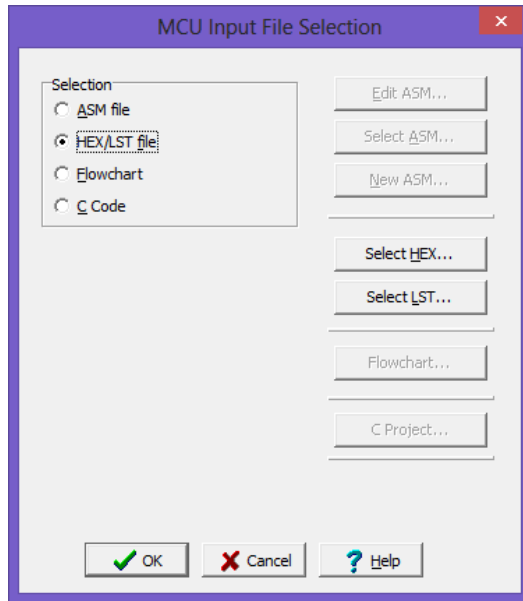
4.6.9.6   Using the ASM Debugger

Now let us run a microcontroller application and see how to test and modify its code. Load the PIC F lasher.TSC circuit from the Examples\Microcontrollers\PIC folder. The following schematic will appear with the 16F73 PIC microcontroller.
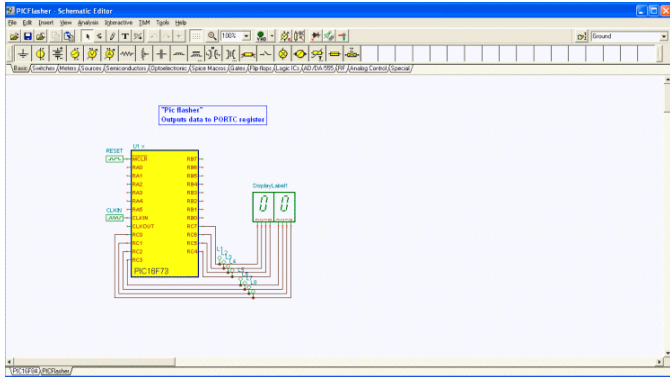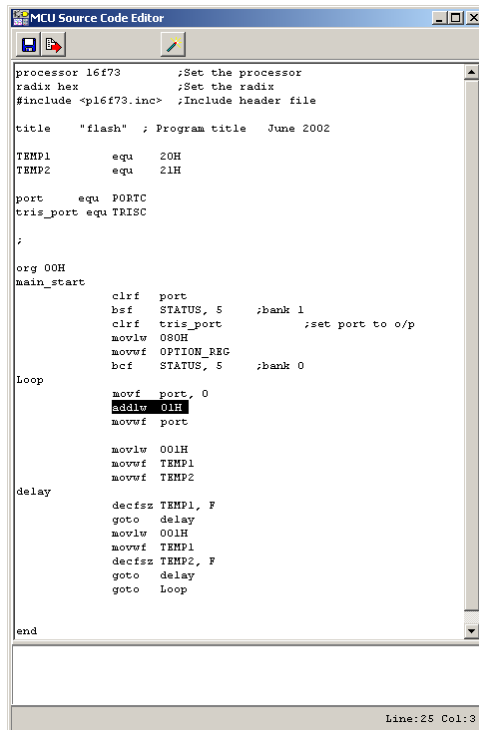
This circuit is simply counting forward one count at a time. Press the ⚙ button to see how it works. The display should step forward one-by one.
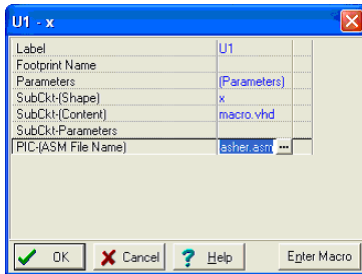
Now let's release the ⚙ button and modify the code to count by 2. Double-click the MCU, and click on the ··· button in the dialog below
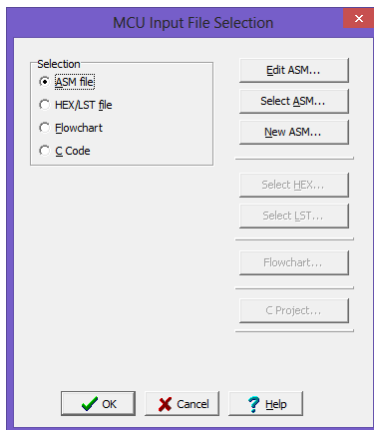
Press the Edit ASM button. The ASM code of the MCU will appear in the MCU Source code editor

Now lets make the following change in the code. Change the instruction (selected above) in line 25 (you can see the line number in the right bottom corner of the code editor window) from

```
        addlw 01H
```
to
```
        addlw 02H
```

Save the changed code by pressing the ⊟ icon and close the open MCU windows. If you press the 🟢 button, now the increment will be 2!

Note that the changed code will be automatically saved in the TINA.TSC file.

4.6.9.7 Example PIC Interrupt handling

Now let's see another application with some more interactivity. Load the PIC16F84_interrupt_rb4_rb7.TSC example from the Examples\Microcontrollers\PIC folder.
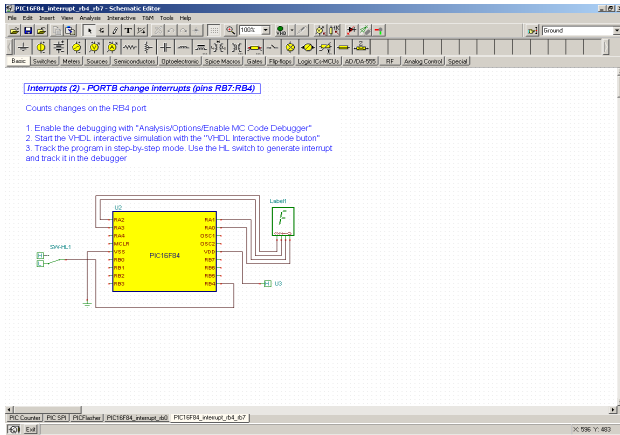
Press the [VHD] button. At first glance, it appears that nothing is happening.

However, if you click on the SW-HL1 switch, the display will step forward by 1 each time that the switch changes from Low to High. This is realized with the interrupt handling capability of the PIC16F84. Now let's see the operation in more detail using the interactive ASM debugger.

To activate the debugger, select Option on the Analysis menu. Set the "Enable MCU Code debugger checkbox," as shown below in the Analysis Options dialog box.

DesignSoft
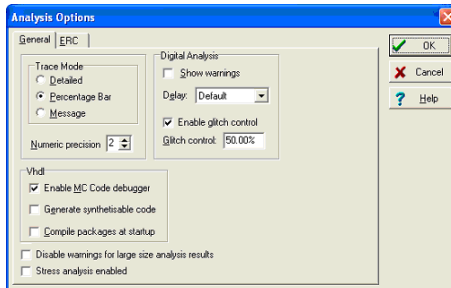
The MCU debugger will appear if you press the button:
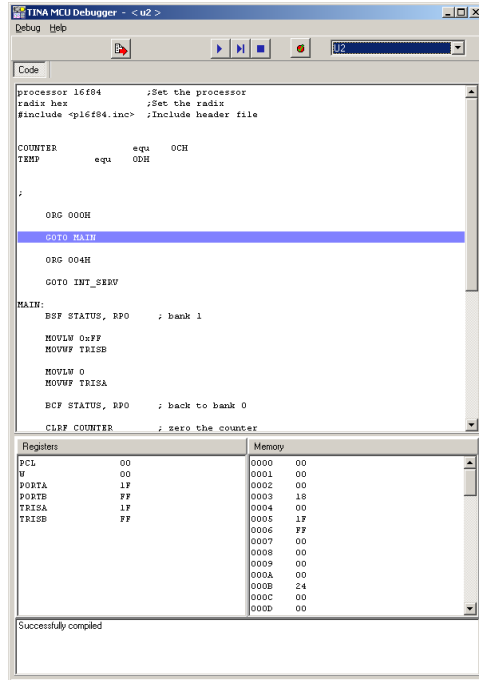
Here is a short description of the MCU debugger dialog.

On the top line there are the following controlling icons:

🔴 Toggle breakpoint: Inserts or removes breakpoints in the selected line. Click on the line where you want to place or remove the breakpoint before clicking on the icon.

▶ Run the code in the debugger continuously. The lines being executed will be highlighted and the code is scrolled.

▶| Step forward. Step by step execution. Each time you press this button one command of the program is executed.

■ Stop Halts program execution.

The Code window (below the control icons) displays the ASM code. The next actual command is highlighted with blue.

The actual content of the registers and memory locations of the MCU are shown in the lower part of the screen.

Let's follow the program execution step-by-step by pressing the ▶| Step forward button. After around 14 clicks, we get to the PT1: label,

where the program seems to be in an infinite loop.

```
PT1: INCF TEMP, F
GOTO PT1
```
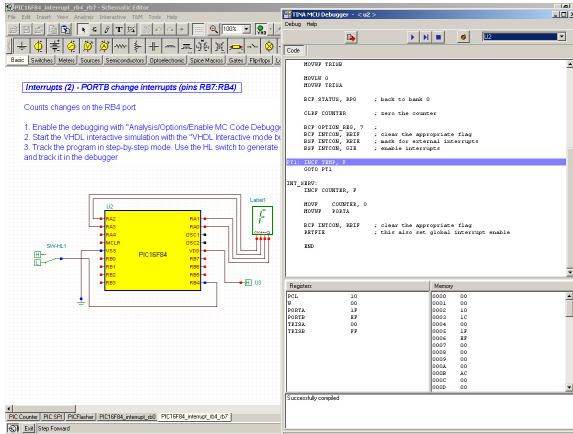

Now click on the SW-HL1 switch and change it to High. (You should click when the cursor changes into an upward pointing arrow Î). Return to the Debugger and click the ▶ Step forward button twice. The program will recognize the interrupt and jump into the INT_SERV: label.

```
INT_SERV:
INCF COUNTER, F
MOVF COUNTER, 0
MOVWF PORTA
```

increment the COUNTER, and copy it to PORT A. The output will now be 1. After this, the program will return to the "infinite loop" at PT1.

4.6.9.8   Editing the ASM Code in the Debugger

Now let's see how to make a small change in the program using the debugger. Duplicate the INCF COUNTER, F statement using Copy and Paste like so:

```
INT_SERV:
INCF COUNTER, F
INCF COUNTER, F
MOVF COUNTER, 0
MOVWF PORTA
```

Now if you press the ▶ the program will ask:

Confirm

? Source code modified. Compile now?

Yes   No
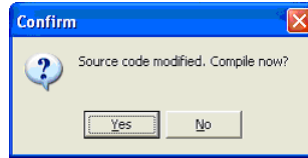
Press Yes and then press the ▶| button again. Now the increment will be 2 at each Low-High change of the switch.

You can also check the circuit in the Debugger's continuous Running mode by pressing the ▶ button. Even though the debugger will run fast, you can still see the "infinite cycle" and the jump to the Interrupt server routine (INT_SERV:) when you change the switch.

4.6.9.9 Making a Breakpoint in ASM

It is often essentially impossible to get to a certain place in the program since you'd have to single step a thousand times (if the program ever steps there in the first place). To get the program to run to a particular statement and halt there, you can tag the statement as a so called breakpoint. Now run the program in the Debugger's continuous mode using the ▶ Run command and the program will stop at the marked space before execution of the marked command. To demonstrate this, click on the increment statement in our interrupt service routine after the INT_SERV: label and press the

⚫ Toggle break button.

Now press the ▶ Run button. The program starts to run and falls into the "infinite loop."

Even though you have set a breakpoint, the code will not stop since it does not pass the breakpoint. However, when you change the switch from Low to High the program will stop at the

```
INT_SERV:
INCF COUNTER, F
```

statement. Now you can resume execution either step by step ▶| or

with the ▶ Run command again.

DesignSoft

4.6.9.10 Programming MCUs using C

Writing assembly code is not easy, and the programmers of desktop computers have turned to a high level programming language. The high level languages are increasing in popularity, and C is perhaps the most used and most useful language for MCU programming. There are many C compilers available on the market, many of them are free or have a free version. You have to install a C compiler which generates the code for the MCU you want to simulate, and then TINA will automatically integrate it into its C code debugger. Here are the C compilers compatible with TINA:

(1) **For PIC:** Install HI-TECH PIC compilers from http://www.htsoft.com/Install it in Lite mode. Select "Add to environment path".

(2) **For PIC18:** Install MPLAB C18 compiler from http://www.microchip.com.

(3) **For AVR:** Install WinAVR from http://winavr.sourceforge.net/

(4) **For 8051:** Install SDCC and GPUTILS

SDCC: http://sdcc.sourceforge.net/ , GPUTILS: http://gputils.sourceforge.net/

(5 **For ARM:** Install the yagarto-tools-*, yagarto-bu-* packages from http://www.yagarto.de

Do not install the AVR, ARM toolchain in a directory which contains spaces like "C:/Program Files/"

You may have to restart your computer after installing these tools.

You can also debug the C-code and execute step-by-step for most MCUs in TINA. However, this does not work for the 8051 and PIC processors older than PIC 18.

Now let's see how to load and run a C code in an MCU in TINA. We assume that the Winavr compiler has already been installed on your computer.

Create a new circuit file in TINA with File/New

Search for and place the ATTiny26 MCU with the Find component tool at the top right corner of the screen. You can also do the same using the Logic ICs MCUs tab of the component toolbar and selecting MCU, then AVR as Manufacturer.

Double Click the MCU, and the property window of the MCU will appear. Click the "C code" line and then the …button. "The MCU input file selection" dialog will appear. Switch on the C Code radio button. You will see the following dialog.

<div style="writing-mode:vertical"></div>

GETTING STARTED
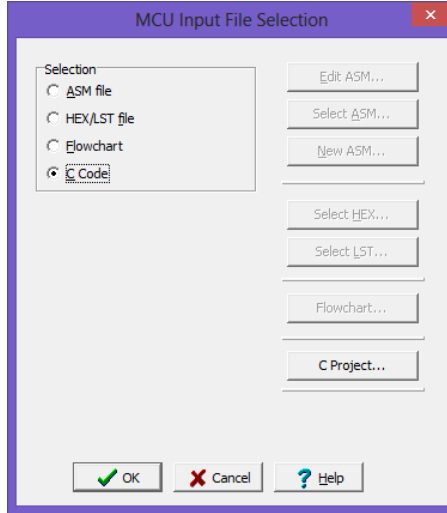
Press the C Project button. The following dialog will appear:
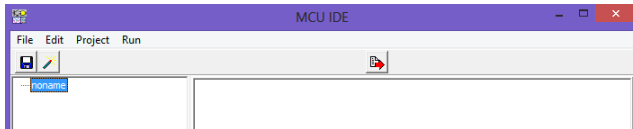
Right-click on the blue noname label on the top left corner of the screen.

A pop- up dialog will appear, click on "Add existing file."

An Open dialog will appear: navigate to the EXAMPLES\Microcontroller\C Compiler\AVR folder (or where your own file resides) and Open AVRflasher.C The following dialog will appear:

131

DesignSoft

Press the ![make] make project and then ![save] Save Project button, and exit by clicking the top right corner of the dialog.

Press OK in the MCU Input File Selection dialog and in the ATTINY26 property dialog.

Select the Digital on the Interactive menu or DIG with the narrow "half" button next to the Interactive mode On/Off button.

Now if you press the DIG button, your C code will start to run. You can see that from the changing red and blue logic states on the pins of the MCU. Note that the simulation of MCUs in TINA works even without a power supply, to simplify the schematics, but of course you should still connect the power supply and all necessary pins when you design a PCB.

Now add a power supply, voltage pin and an LED to the circuit as shown on the picture below.
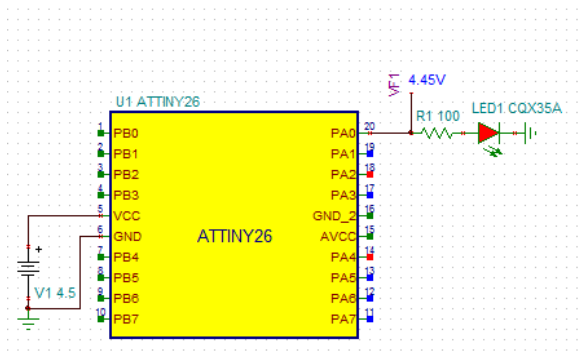
If you run the simulation, the LED will light when the logic level on PA0 is high and the voltage pin will show the analog voltage. Note that this voltage follows the voltage of the power supply. The other pins are handled with logic values according to the rules of Mixed Mode Simulation. This makes a big difference compared with external simulation of MCUs as in TINA. You can also see all the voltages and displays of your complete circuit during the simulation. You can do this even step-by-step with the help of the TINA C debugger

4.6.9.11 Debugging C code in MCUs

Just as with ASM and HEX code, you can follow the execution of a C program and even watch the values of the required variables.

To demonstrate this, let's open the "check_prime.TSC" file in the Microcontrollers\C compiler\AVR folder. The following circuit will appear:
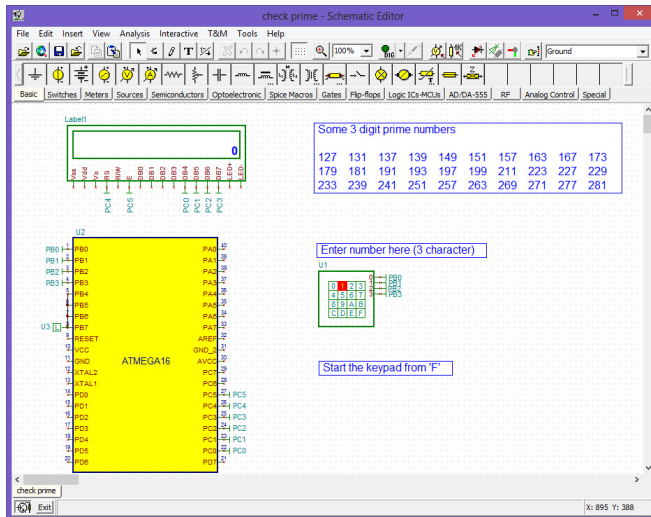
To test this circuit, press the DIG button and enter a 3 digit number (each digit must be different). The display will show "Prime number" or "Not prime."

Now to debug this C program, release the DIG button and then click the "Enable MCU Code debugger" in the Analysis menu and then press the DIG button again.

DesignSoft

The C code debugger window (MCU IDE) will appear.
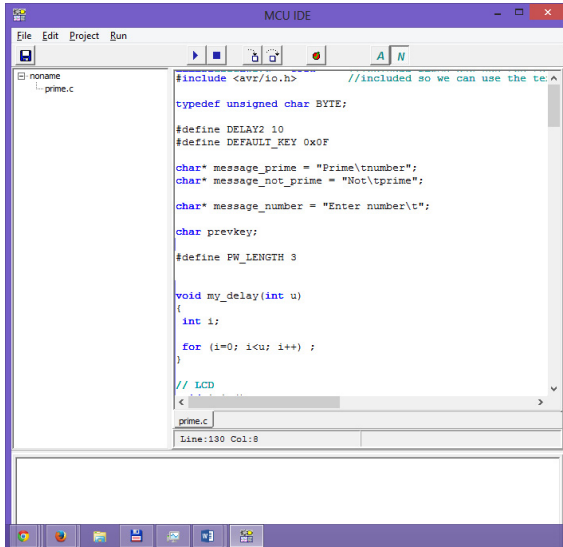
Scroll down the C code until the following function appears (line 128)

```
int check_prime(int a)
{
int c,mod;
for ( c = 2 ; c <= a - 1 ; c++ )
{
mod = a%c;
if (  mod == 0 )
return 0;
}
if ( c == a )
return 1;
}
```

Right click the variable "c" and click "Add Watch at cursor". Click the "Watch List" tab at the bottom of the MCU IDE window. The variable c should be on the list already. Add variables "a" and "mod" the same way.

Now click the if (mod == 0) line and press the ![icon] Toggle Breakpoint (Add Breakpoint) button at the top line of the MCU IDE window. The line with the breakpoint will turn red.
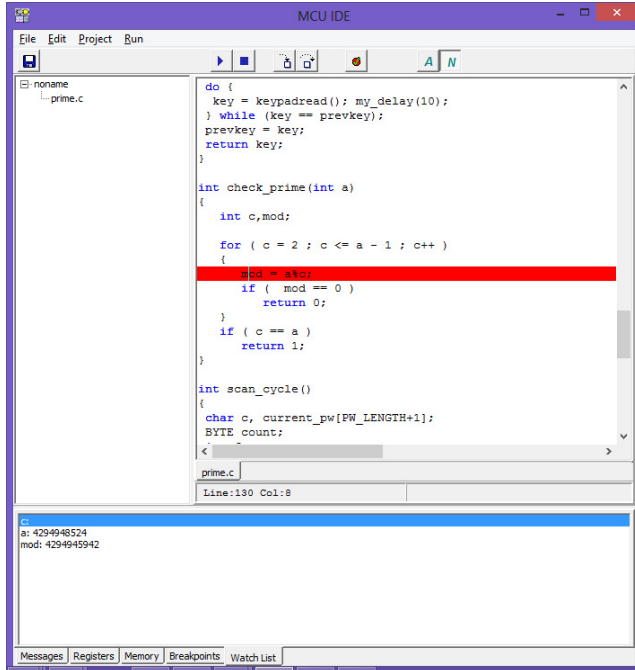
Press the ![icon] tart button. The progam will start to run. You will not see changes in the debugger window unless you also press the the „A" animate button at the top of the screen, which will show the active instructions and scroll the screen when necessary. However if you enter a 3 digit number on the small keyboard on the screen, the progam will stop at the if (mod == 0); line and you will see the a,c,mod variables at the breakpoint.

After the stop at the breakpoint, you can continue the execution step by step by pressing the ![icon] Step button for each step or run continuously by pressing the ![icon] Start button again.

4.6.10 Using the Flowchart Editor and Debugger in TINA

Writing MCU assembly code is often a hard and tedious task. You can simplify software development and gain more time to design the electronics hardware if, instead of manual coding, you use TINA's Flowchart editor and debugger to generate and debug the MCU code. This easy-to-use tool works with symbols and flow control lines with which you can represent the algorithm you want. The Flowchart editor is opened via a MCU device as described below. You can find detailed descriptions of the flowchart symbols and their parameters under the Help menu of the Flowchart editor.

4.6.10.1 Flowchart Editor

In the following example, we will create a flowchart to control a MCU embedded in a simple circuit. The flowchart adds two numbers that are read from two ports of the PIC16F73 microcontroller. (You can find the complete circuit under EXAMPLES\ Microcontrollers\PIC*PIC Adder.TSC* in TINA).
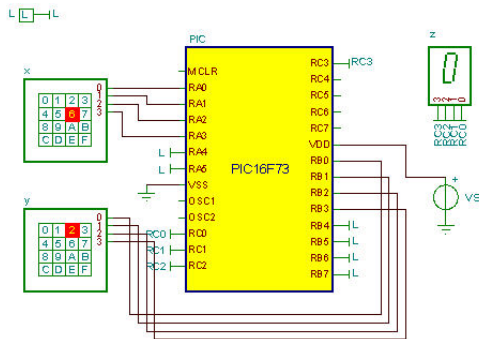
First, select the MCU from the component toolbar and insert it into the schematic editor. The MCU components are located under the *Logic IC-s-MCUs* tab. Click the MCU icon on the toolbar and select PIC16 in the Manufacturer line of the MCU dialog. The list of PIC16 MCUs will appear. Select PIC16F73 and click *OK*. The selected MCU will be placed in the schematic editor.
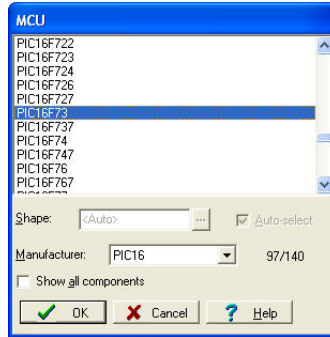
Now double-click on the PIC16 component in the editor, click the *MCU-(ASM File Name)* field, and then the ⚫⚫⚫ symbol. The *MCU Input File Selection* dialog appears.
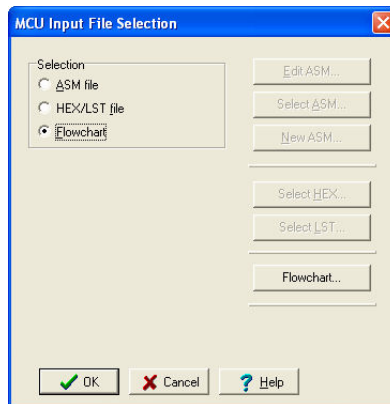
In this dialog, select the *Flowchart* mode under Selection on the left. Click on the *Flowchart* button on the right, and the TINA Flowchart Editor opens.
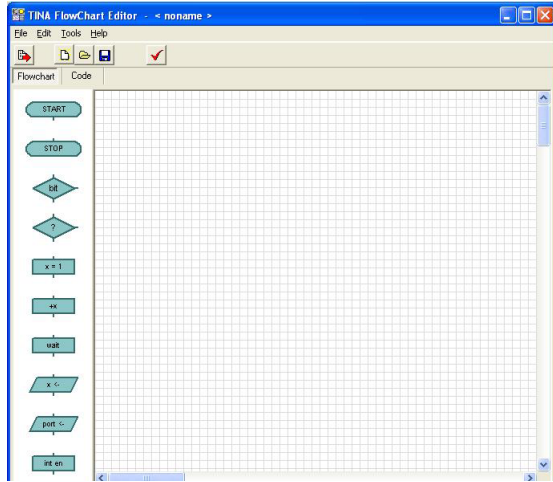
With the *Flowchart* tab selected, the editor window is divided into two areas. On the left there is a toolbar of the symbols that you can place in the editor area on the right.

Select the *START* symbol from the toolbar by clicking on it. The START symbol will be attached to the cursor. Move it to the middle of the editor space and click with the mouse to place it.

Next we will read PORTA and PORTB ports into the variables x and y, and add them, write the result into PORTC. Flowchart execution will loop back to Start and continue running until stopped.

To read in the contents of PORTA port, click the symbol (*Read Input*) on the toolbar and move it into the editor space. Double click on this component to select the port you want to use for reading (*Source port*) and to set a variable name (*Target variable*). Select *PORTA* for *Source port* and enter *x* for the *Target variable*.

Now we must read in the next data by reading from another port, PORTB. This is very similar to the previous step. Insert another Read Input symbol, set *y* for *Target variable* and *PORTB* for *Source port*.
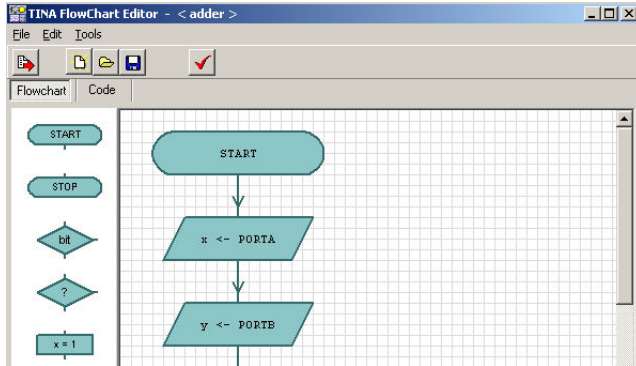
Next we will add the two numbers in *x* and *y* and write the result to PORTC. Here is how to do this.

First save the contents of *x* to a temporary variable, named *z*. To do this, select the [x = 1] *Set variable* component from the toolbar, set *Target variable* to *z*, and then *Value and variable* to *x*.

Next add *z* and *y*, by using the [+x] *Change variable* component Place the *Change variable* component below the previous *Set variable* component, set *Target variable* to *z*, *operator* to +, *Value or variable* to *y*.

To write the result to PORTC, add an *Output* component symbol, Select *Target port* to PORTC and *Value or variable* to *z*.

Finally, let's connect the flowchart symbols with flow lines, indicating the direction of the flow, by moving downwards until you reach the bottom PortC <-z *Output* symbol.

First, connect the START and Read Input symbols. To connect the two symbols, move the mouse above the end of the *connection line* out of the START symbol. The *connection point* is at the end of the connection line, and is marked by a small rectangle visible when the mouse is at the right position. When the small rectangle appears, press and hold the left mouse button, and drag the connection line until you reach the connection point of the other symbol.

Connect all other symbols in a similar manner.

When you reach the lowest symbol, connect the lower connection point of this symbol with the flow line between the START and x <. PORTA component.

If the symbol placement and connection are correct, the flowchart will look like this.

DesignSoft
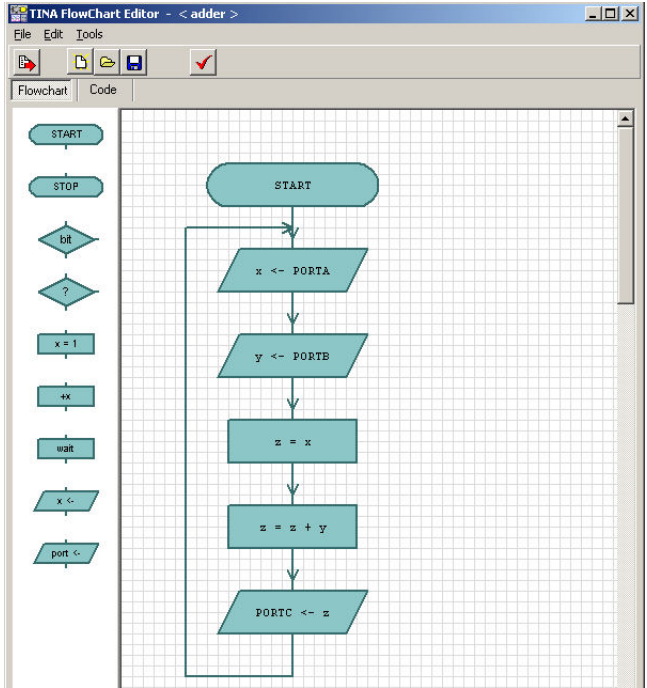
To make a formal check of the flowchart (e.g., see if all symbols are connected) press the ✓ button. To view the generated code, press the *Code* tab.

To save the flowchart in the MCU macro, press [icon] on the toolbar, then press *OK* twice (on the *MCU Input file selection* and on the MCU property box dialogs) to go back to the schematic editor.

You can also save flowcharts in .tfc files with the Save and Save As… commands of the Flowchart editor. You can then Open them and associate them with other MCUs.

4.6.10.2 Flowchart Debugger

TINA automatically produces the assembly code required for the simulation from the flowchart.

Let's test and debug the previously completed flowchart. (You can open the complete circuit from (EXAMPLES\Microcontrollers\ PIC*PIC Adder.TSC*)).

Set the *Analysis/Enable MCU code debugger* switch in the Analysis menu, then press the [icon] button on the TINA toolbar. The MCU debugger appears.

DesignSoft
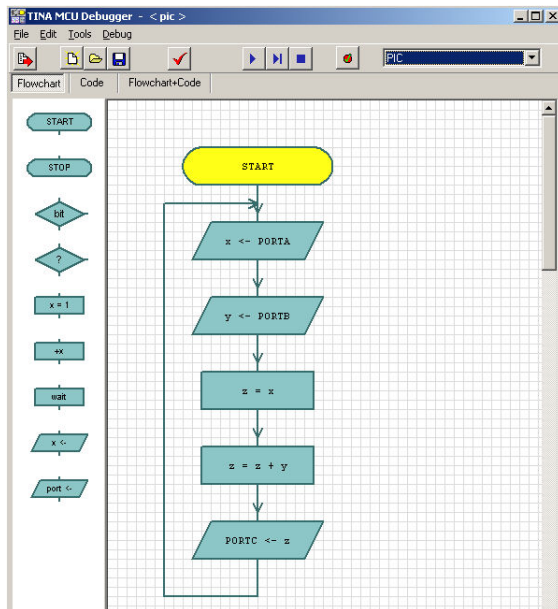
You can run the program continuously by pressing the ▶ Run button, step-by-step by pressing the ▶ǀ Step Forward button, or stop the program by pressing the ■ Stop button. The debugger will show the active flowchart component by setting its background color to yellow.

There are three important tabs at the top-left corner of the debugger that establish the view of the source. If you select the *Flowchart* tab, you can see and debug via the Flowchart. If you select the *Flowchart+Code* tab, TINA will display both the flowchart and the assembly code. You can, in this view, place breakpoints both in then flowchart and in the assembly code. If you select the last mode, *Code,* you can debug using traditional assembly language debugging. See section 4.6.9.9 in this manual.

Note that in order to synchronize the flowchart and the assembly code, and to make the code more readable, TINA inserts additional labels and comments into the code; for example:

Flowchart label2: ;x <- PORTA

These labels do not alter the performance or the operational logic of the code.

Breakpoints are used to halt code execution at user-specified points, permitting the examination of registers and parameters. TINA offers several ways to insert and remove breakpoints.

Insert breakpoints into the flowchart by clicking on a flowchart symbol and pressing the ⬤ Toggle Breakpoint button.

Or you can place breakpoints into the code directly via the code window. Select an instruction line and press the ⬤ button .You can remove the breakpoint with the same button when the breakpoint is selected.

When breakpoints are set, the program will stop at breakpoints *before* executing the instruction under the breakpoint. You can execute the instruction and continue the program by clicking the ▶ *Run* or the ▶ǀ *Step Forward* buttons.

4.6.11   Testing your Circuit with Simulated and Real Time Instruments

TINA lets you test and tune your circuit not only with the generators and analysis windows that you have used so far, but also with simulated and real-time measurements. Using the T&M menu you can place virtual instruments on screen which will automatically

replace the generator and analysis windows. You can control the settings of these instruments and immediately see the results, just as you would in a real lab. *TINA* normally simulates measurements with its analysis engine, but if you have *TINA's* supplementary hardware, you can simply switch to the Real Measurement mode (using Option of the T&M menu). Now you can work with the same on-screen instruments and settings and you will be making real measurements on a real circuit.
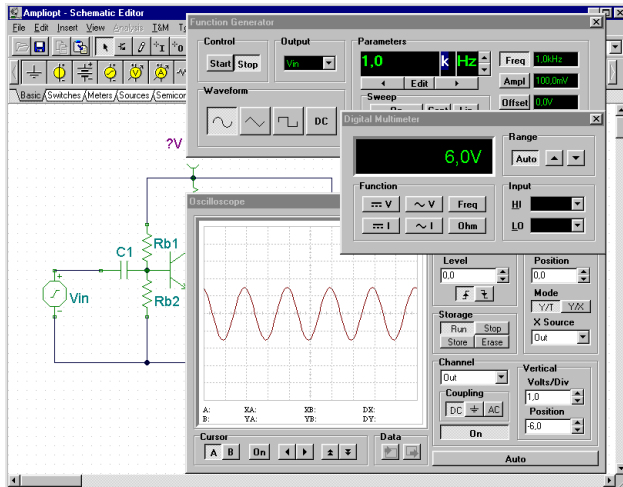
To start with virtual measurements, load the circuit **AMPLIOPT.TSC** from the examples directory. Select the T&M menu and place a Multimeter, Function Generator, and an Oscilloscope on screen. Press Run in the Oscilloscope panel. A distorted sine curve will appear on the oscilloscope screen. Click on the multimeter = button. The multimeter will show only about 0.7 V volts at the collector (Out) - this is the reason for the distortion. Now double-click on the Rb1 resistor. The dialog box of resistor properties will appear. Click on the Resistance field and then change the value with the arrows on the right hand side of the dialog box until the multimeter shows about 6V.

You can change the step size of the buttons by entering it into the edit box under the *down* arrow. You can also define a hotkey to the up and down arrow by selecting it from the upper and lower listboxes. Note that while the interactive mode is On, and once a hotkey is defined, the value of the resistor can be changed directly by pressing

the appropriate key without opening the property dialog. You can assign hotkeys to most component values in TINA, including switches. To avoid accidental changes, the hotkeys for component values will only work while TINA is in the interactive mode. Switch positions, however, can be changed before activating the interactive mode in order to set their initial position. Once the collector voltage reaches 6V, close the property editor dialog box (if it is still open) and press Run on the oscilloscope.

Set the vertical position to -6 V and use the horizontal and vertical settings to scale the curve for best appearance. The distortion will no longer be visible.

Press the *Ampl.*-button on the generator. The last amplitude value will appear in the large numeric display field of the generator. Use the vertical arrows beside the display to change the amplitude. You will see the sine wave become distorted again as you increase the amplitude, with the maximum input at about 500mV. Now change the waveform from sinusoidal to triangle and then to square wave. Vary the frequency of the function generator to explore the frequency domain over which the circuit performance is acceptable.

NOTE:

The virtual instruments under the T&M menu are not to be confused with the virtual instrument components on the Meters component toolbar. Some of the virtual instrument components are used in the interactive mode of the program, discussed in the next section. They are also used to assign outputs for the various analysis modes under the Analysis menu. The Oscilloscope and Signal Analyzer virtual instrument components have a small screen and their purpose main function is to be used with our 3D circuit analyzer program, EDISON.

4.6.12   Using the Design Tool in TINA

TINA's Design Tool works with the design equations of your circuit to ensure that the specified inputs result in the specified output response. The tool requires of you a statement of inputs and outputs and the relationships among the component values. The tool offers you a solution engine that you can use to solve repetitively and accurately for various scenarios. The calculated component values are automatically set in place in the companion TINA schematic and you can check the result by simulation.

As an example, this tool can calculate feedback or other resistor and capacitor values of an amplifier in order to achieve a certain gain and bandwidth, and it can calculate component parameters of power supply circuits to meet output voltage and ripple requirements.

The TINA Design Tool promotes good documentation by storing the design procedure together with the circuit.

It is also very useful for semiconductor and other electronics component manufacturers to provide application circuits along with the design procedure.

Let's demonstrate the use of this tool through a simple operational amplifier example.

Open the Invert Gain OPA350 Test Circuit Design.TSC circuit from the Examples\Design Tool folder of TINA. In the TINA Schematic

Editor the following circuit will appear:
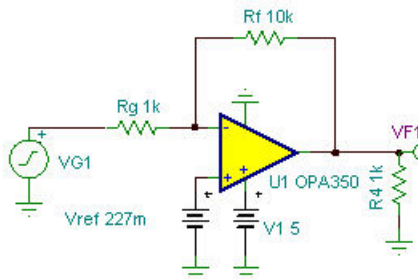
With the Design Tool we will set Rf and Vref to achieve the specified Gain and DC output voltage.

Now invoke the Design Tool from the Tools menu of TINA. The following dialog will appear.
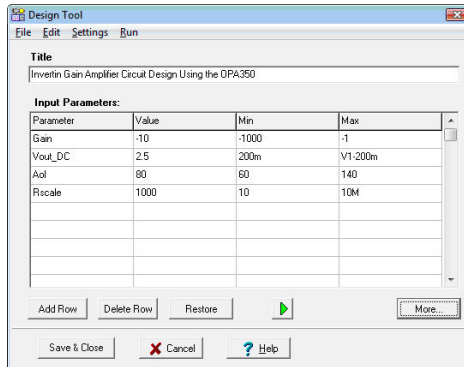
Here you can specify the Gain (Vout/Vin), the DC output voltage (Vout_DC) and some other parameters. The simple design procedure will calculate Rg and Vref. The allowed minimum (Min) and maximum (Max) parameters are also shown. To enable or disable modification of Min and Max select Options from the Settings menu of the Design Tool.

Note that in the Design Tool dialog you can also refer to component parameter names. For example in the Vout_DC line the maximum value is set as V1-200m, telling that the DC output voltage must be at least by 200mV less than the V1 supply voltage of the IC.

If you just want to Run the design procedure press the ▶ button or the F9 key or use the Run command in the menu of the tool. If you run TINA in interactive mode you can immediately see the effect of the changes made by the Design Tool.

To see the design procedure itself, press the More button in the dialog. The code of the design procedure, written in TINA's Interpreter, will appear.
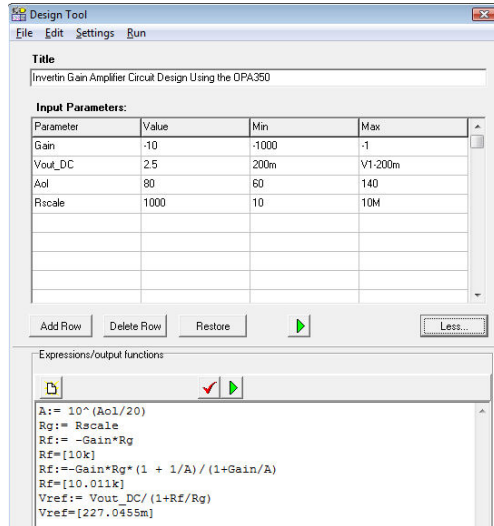
Note that the code part also shows the calculated parameter values according to the last stored calculation (Now Rf=[10.011k], Vref=[227.0455m]).

Now let's change the Gain input parameter to -1, Vout_DC to 3V and run the procedure by clicking Run in the menu or pressing the green ▶ button or F9 on the keyboard. In the code part we will see:

```
A:= 10^(Aol/20)
Rg:= Rscale
Rf:=-Gain*Rg*(1 + 1/A)/(1+Gain/A)
Rf=[1.0002k]
Vref:= Vout_DC/(1+Rf/Rg)
Vref=[1.4998]
```

and the new values will immediately appear in the schematic editor, drawn in brown color. Press the DC button to display the DC output voltage:

DesignSoft
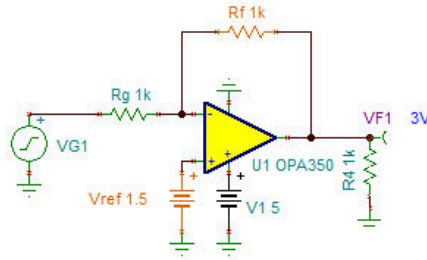
Now run an AC Transfer analysis, the Bode diagram will appear.
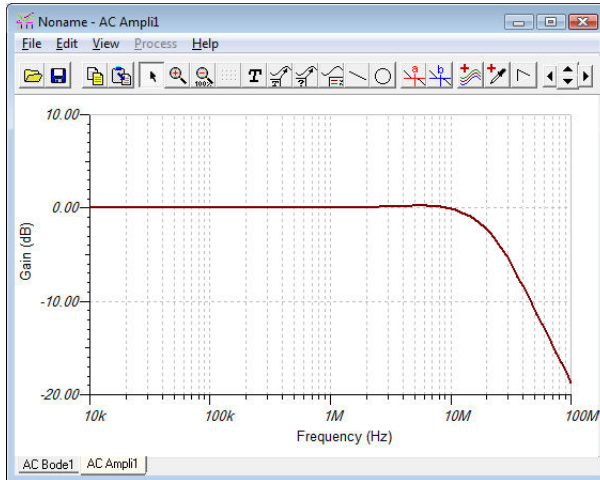
The small frequency Gain is 0dB which complies with the specified Vout/Vin=-1 value.

You can find more complex examples in the Design Tool folder of TINA.

You can make your own design procedure in any TINA circuits and save it together with the circuit itself.

For more information on the use and the controls see the on-line help in TINA by pressing the Help button on the tool and also the Advanced Topics Manual of TINA, available on-line at www.tina.com under Documentation.

4.6.13 Design Tool vs. Optimization in TINA

There are cases when writing a design procedure is not obvious or needs iteration or simply we do not have the time to implement it. In this case you may use the Optimization tool in TINA to determine the required parameter numerically in order to meet predefined circuit responses: voltage, current, power, gain, etc. You can learn more and find examples of Optimization in the Advanced Topics Manual of TINA available on-line at www.tina.com under Documentation.

NOTE:

> *Optimization for multiple parameters and target values is available in the Industrial version of TINA only.*

Generally speaking, although optimization is a very powerful tool, it is better to use a design procedure, if available because numerical optimization might need significant calculation time and it does not guaranty physically realistic results. But it is a very good tool to refine the results provided by a design procedure or tune already working circuits.

4.6.14 Live 3D Breadboard

Using the Live 3D Breadboard tool in TINA, you can *automatically* build a life-like 3D picture of a solderless breadboard (sometimes called a "whiteboard"). When you run TINA in interactive mode, components like switches, LEDs, instruments, etc. become "live" and will work on the virtual breadboard just as in reality. You can use this capability of TINA to prepare and document lab experiments. Beware, though, of the relatively high capacitance that exists between rows of adjacent pins. This comes about from the "parallel plates" that descend 3 to 5mm from the white board. For high frequency circuits (above 100kHz or so), this capacitance can lead to unexpected performance.

DesignSoft

You can either assemble the circuit step-by-step or by generating the whole circuit on the breadboard. Pick up and move parts on the breadboard using the mouse, and TINA will automatically rearrange the wiring while retaining connectivity. In the same fashion, you can select and move wires for clearer appearance. Note that you cannot change the endpoints of a wire this way–wiring integrity is preserved.

The breadboard tool is mostly intended for educational purposes to prepare laboratory experiments in a safe 3D environment. You can also use this breadboard to guide you in actually wiring a physical breadboard for lab verification.

Let us demonstrate the use of Live 3D Breadboard though a few examples.

Boot TINA and bring in an unpopulated breadboard. Starting at the View menu, select Live 3D Breadboard and New. The 3D Viewer window will appear with an empty breadboard, in TINA's main window. This window is always on top so you can see both the schematic and the breadboard at the same time.

place a Voltage Source, a Resistor, and a Voltmeter (found on the Basic toolbar), and an LED (from the Semiconductor toolbar), in the Schematic Editor window. The 3D version of these parts will be automatically appear in the 3D Viewer window.
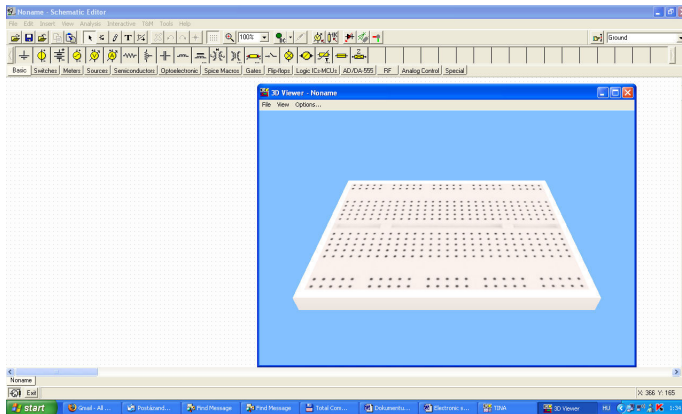
Connect the parts on the schematic with wires. TINA will place the corresponding wires on the breadboard automatically. Now save this circuit as LED Circuit.TSC using the File/Save command of the Schematic Editor.
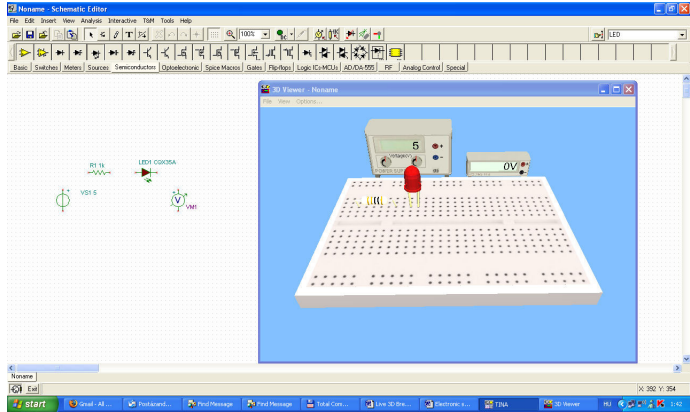
Press the DC Interactive analysis button or select Start from the Interactive menu. The LED on the breadboard will light and the Voltmeter will show the LED voltage. To get a better view, drag the Voltmeter to the foreground: click and hold down the left mouse button and move the mouse.
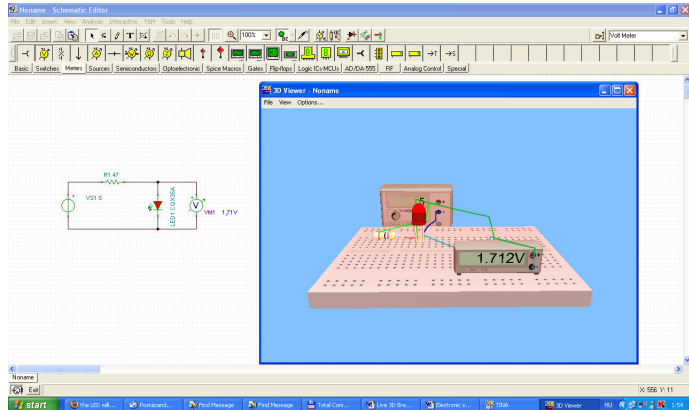
While in the interactive mode, you can modify several parameters and see the resulting changes on the virtual breadboard. For example, you can change the generator output voltage by moving the cursor above the knobs of the generator and turning the mouse scroll wheel. Or another technique is to hold down the left mouse button and move the cursor around the knobs. Yet another way to adjust a parameter is to move the cursor over the knob and right click with the mouse.

DesignSoft

A parameter dialog opens for the left knob of the generator, and you enter a value in the data box. The dialog box looks like this:

After you have entered a value, either press the ✔ button to confirm the change, or press the ✖ button to cancel.

When you are satisfied with the breadboard, save the 3D arrangement via the Save command from the File menu of the 3D View window. It will save the 3D view under the same name as the schematic, but with the .3DV extension.

Note: If you are dealing with a new circuit, we recommend that you name and save it right after choosing New file; otherwise the 3D view will be saved as Noname.3DV.

To open this example at a later time (with any changes you may have saved), load the file LED Circuit.TSC. Select 3D live breadboard from the View menu and press Continue.

You can find additional examples in the Breadboard folder EXAMPLES\3D Breadboard. Open the file 555_AST_BB.TSC and explore its more complex animation and interactivity. This screen will appear:
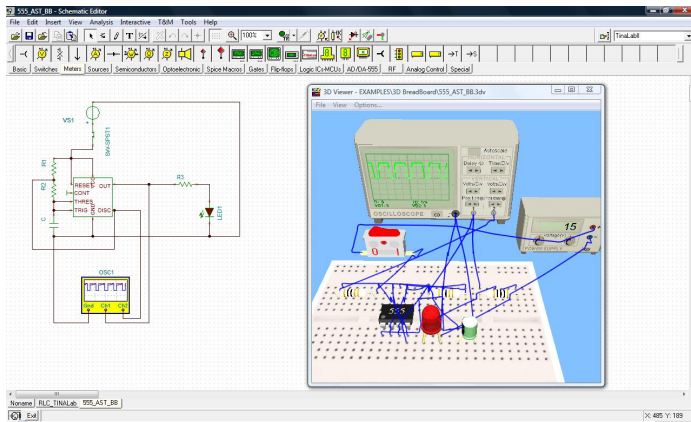
The file includes a screenshot of the circuit's 3D breadboard. To start the animation, open the working window, select Live 3D breadboard, and click on Continue (in the View menu).

Begin the live simulation (press the TR button on the toolbar or select Start from the Interactive menu). The waveform will appear on both the oscilloscope in the TINA schematic editor, and on the 3D oscilloscope in the 3D window. Once the 555 is active, the LED will start flashing. Note that, just as you could change a parameter of the generator in the previous example, here, too you can change the settings of the oscilloscope in the 3D window. Edit the settings using the mouse as described above.

To switch simulation on and off, click on either the switch on the 3D window or in the schematic editor.

Note: there is another way to create experiments in which virtual instruments are used; namely, you can connect virtual instruments to the 3D view of PCBs designed in TINA. Here's how you do this. With the same 555 timer file open, press the PCB Design button on the toolbar or select PCB Design from the Tools menu. The PCB design of the circuit will appear. Press the 3D View button on the PCB Designer's toolbar or select 3D View from the View menu. The 3D view of the experiment will appear. As with any breadboard, you can see the animation and control it with the switch.
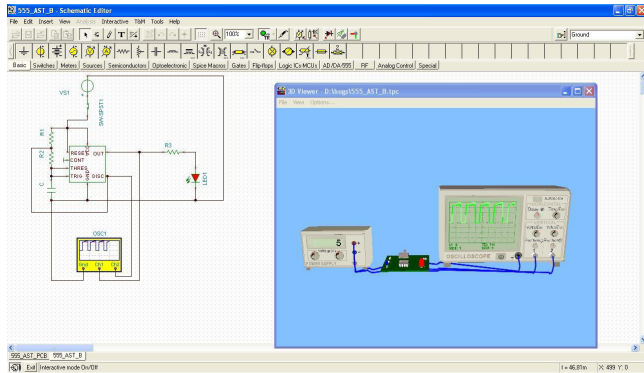
If you turn the PCB upside down (by holding down the left mouse button over an unused area and moving the mouse), you can see the connections of the wires from the PCB solder side to the oscilloscope.

DesignSoft
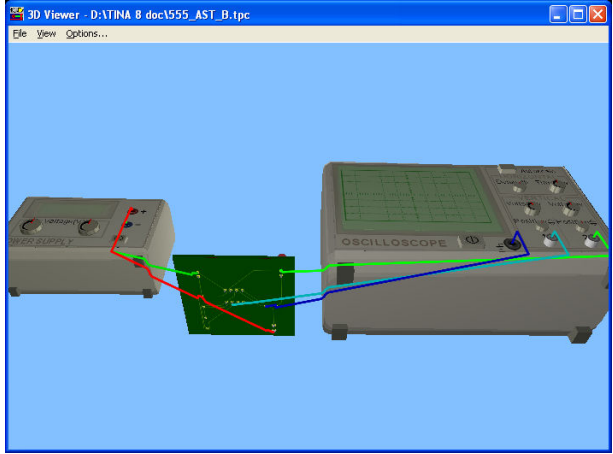
Note that you could use connectors instead of soldering wires directly to the PCB and connect the instruments through those. By adding Header2 and Header3 (they can be found under the Connectors button on the Switches component toolbar), you can make the interconnection shown on the picture below.
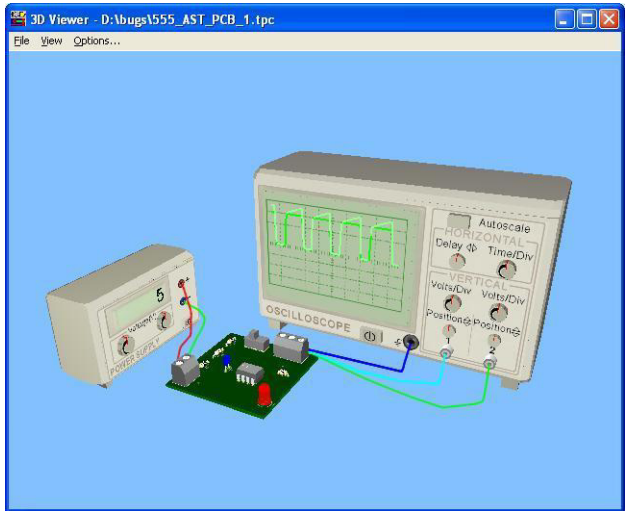
4.7   Creating a Printed Circuit Board (PCB)

Once you have completed your circuit diagram, you can design a printed circuit board to manufacture your design. This is easy to do in TINA 7 and later versions, since PCB design is an integral part of the program.

We'll learn about the PCB design process through a few examples.

The files from the different phases of the design examples have been saved in TINA's Examples/PCB directory using the following naming conventions:

* origin.tsc	original schematic file
*.tsc	schematic file backannotated (after pin/gate swapping and renumbering)
*placed.tpc	design parameters set, components placed pcb file
* routed.tpc	net properties set and routed pcb file
* finished.tpc	optionally pin/gate swapped and renumbered, routed, silkscreen adjusted, documentation layers finalized pcb file

4.7.1   Setting and checking footprint names

To see the first example, open the opamp2.tsc project from TINA's Examples\PCB folder. The following schematic will appear:
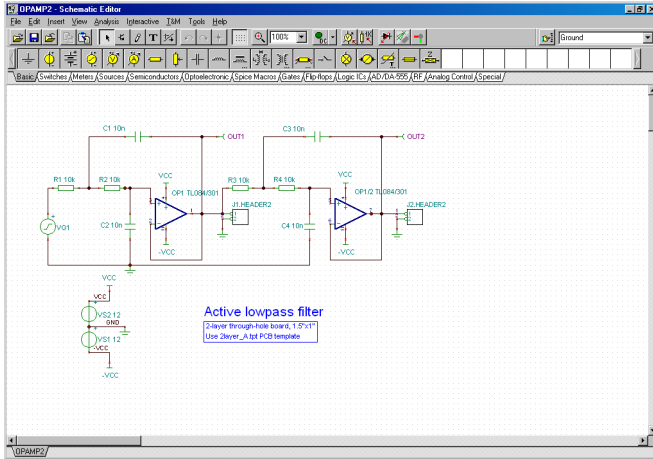
The most important thing in PCB design is that every part in your schematic must have a physical representation with exact physical size. This is accomplished through so called footprints–drawings showing the outline and the pins of the parts.

TINA's footprint naming uses as a starting point the IPC-SM-782A (Surface Mount Design and Land Pattern Standard) and the JEDEC standard JESD30C (Descriptive Designation System for Semiconduc- tor-Device Packages). (See http://www.jedec.org/download/search/ jesd30c.pdf.)

In TINA, we have already assigned default footprint names to all parts which represent real components.

NOTE:

Some parts used for theoretical investigations (for example, controlled sources) do not represent real physical parts so you cannot place them on a PCB. If your design contains such components, you should replace them with real physical parts.

Of course there is no guarantee that the default physical representatives of the parts are the same as those needed by your design There are two ways to check this.

1) You can use TINA's "Footprint name editor" which you can invoke from the Tools menu. In this dialog you see all of TINA's components and the corresponding footprint names.
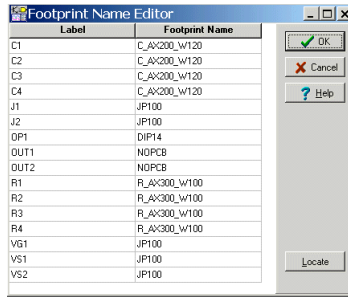
Clicking on the footprint name fields, you can select from the available footprint names. In the dialog, components that do not already have a footprint name association will be denoted by red characters and also by "???" in the footprint name field.
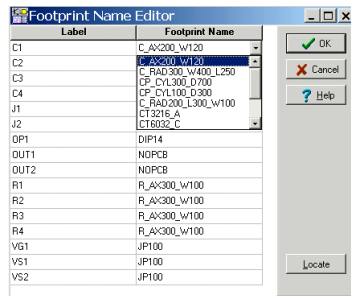
Note that although there are 2 opamps in the schematic diagram, there is only 1 opamp part in the footprint list above: a DIP14 package denoted by OP1. This is because the TL084 IC we use lcontains 4 opamps, so both of our opamps are in the same package. This is shown by the labels of the opamps in the schematic diagram: OP1 and OP1/2. If the parts are in the same package, the labels must also be the same. Note that the second, third, and fourth opamps have the extensions /2, /3, or /4 added to the label. You must not set the extensions /2, /3, or /4 manually, rather, use the "PCB information" dialog of the parts. An example: double-click on the parts to see the property dialog.
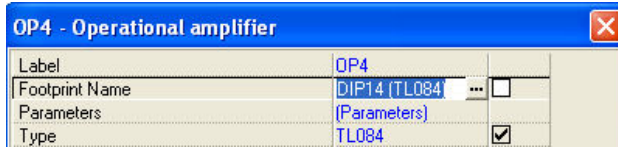
Now click the Footprint line and press the ⋯ button. The PCB information dialog appears and you can select the number of the part in the package at the Part field of the top-right corner of the dialog below.

2) Alternatively you can double-click on each part and check the Footprint Name of the component property dialog.

You can also click the ⋯ button in the Footprint Name line and see the "PCB information" dialog where you can select from the available footprint names. You can also see the 3D view of the different parts via the 3D package view field of the dialog.
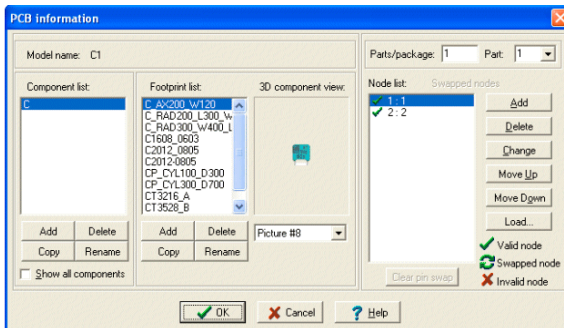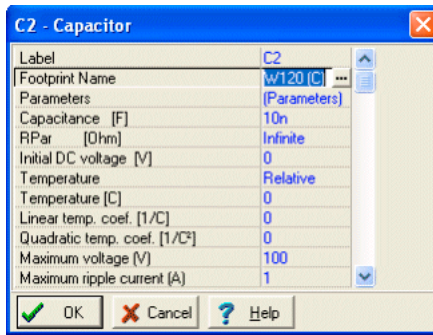
If you find the footprint name you want on the list, click on it and press OK: you will be returned to the component property dialog with the selected footprint name in the Footprint name line. To confirm the change, press OK on the component property dialog again.

If you do not find the footprint name you want, you can add a new footprint using the Add buttons of the "PCB information" dialog. Press the Help button for more information.

When everything looks good, you can make a final check by clicking the ![icon] 2D/3D view button . The 3D view of those components for which a physical representation has already been added will appear.
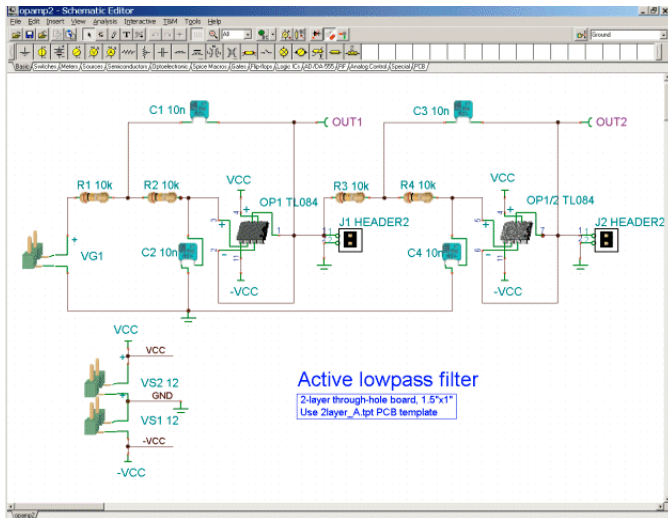
4.7.2   Invoking TINA PCB

Once each component has a satisfactory physical part association, we can proceed to PCB layout design. To do this, press the ![icon] button on TINA's toolbar (the last icon on the right) or select the "PCB Design" command on the Tools menu. Set the items in the PCB Design dialog as shown below.

GETTING STARTED

DesignSoft

Select "Start New Project", "Autoplacement" and "Use board template". With the Browse button find and select the 2layer_A.tpt template files from TINA's Template. The settings are appropriate for a double-sided PCB.
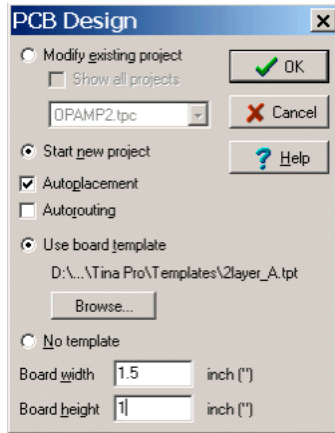
If you use a template, you should set the level of manufacturing complexity. The following three levels of manufacturing technology are defined by the IPC-2221 generic standard.

```
Level A : General Design Complexity
Level B : Moderate Design Complexity
Level C : High Design Complexity
```

The template file specifies the number of layers and their properties: system grid size, autorouter settings, spacing and track width. The following templates are included with PCB Designer:

	Level	Routing Layers	Plane Layers	Routing	Spacing	
1layer_A.tpt	A	1	-	25	12·1/2	Allows one track between standard
2layer_A.tpt	A	2	-	25	12·½	DIP IC pin
2layer_B.tpt	B	2	-	8·1/3	8·1/3	Use for SMT or mixed-technoly board
2layer_B_mm.tpt	B	2	-	0.1	0.2	
4layer_C_mm.tpt	C	2	2	0.1	0.15	For moderate and high density SMT boards

You can choose PCB template based on technology, density and package pitch.

Finally, you can set the size of the PCB board in inches or mm depending on the measurement unit settings in the View/ Options dialog of TINA.

When everything is set properly, press the OK button and the PCB layout design will appear with all the components automatically placed on the PCB board.
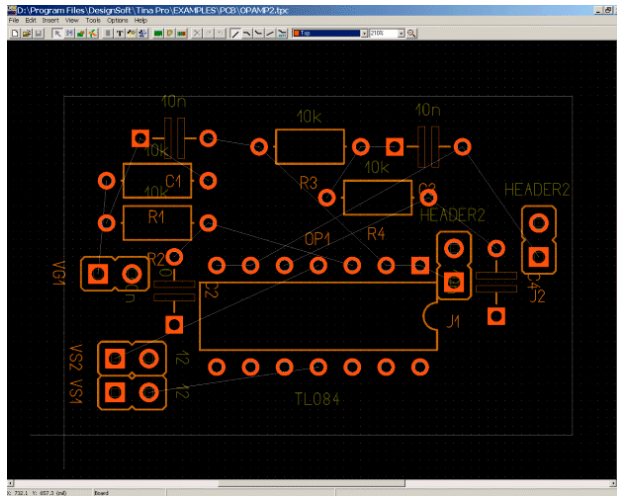
Now click and drag the parts to new positions as, shown on the figure below. (Find "opamp2 placed.tpc" to check your results.)

DesignSoft
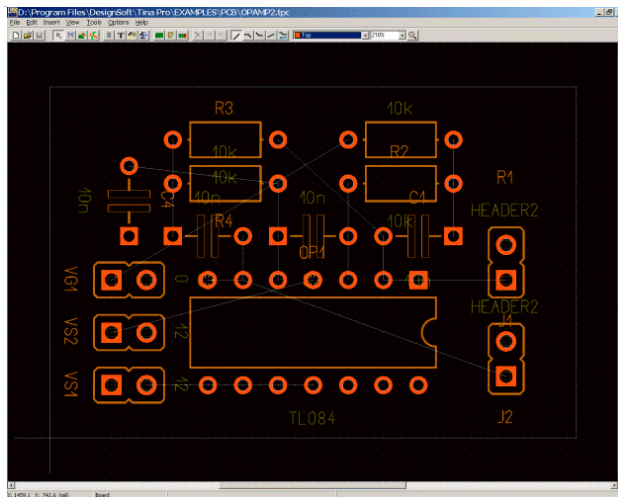
Press F4 to invoke the Net Editor and set nets routing width. First, click on "Modify all" and enter 12.5 into the "Track width" field. Then select power nets (Ground, VCC, -VCC) and set their widths to 25mil.
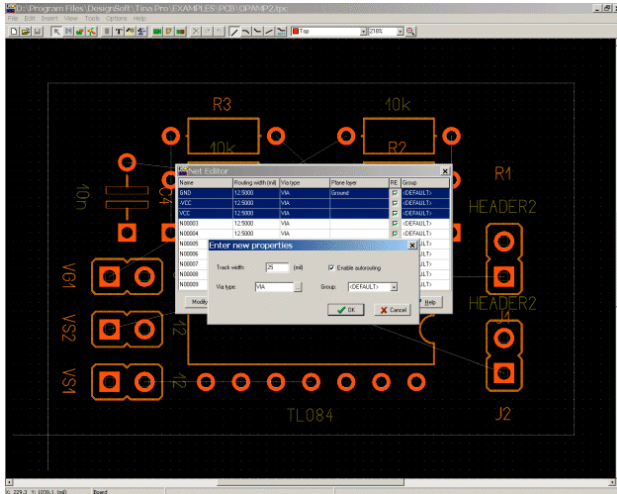
To automatically route the netlist, press the F5 button or select "Autoroute board" command from the Tools menu. The following screen will appear:

DesignSoft
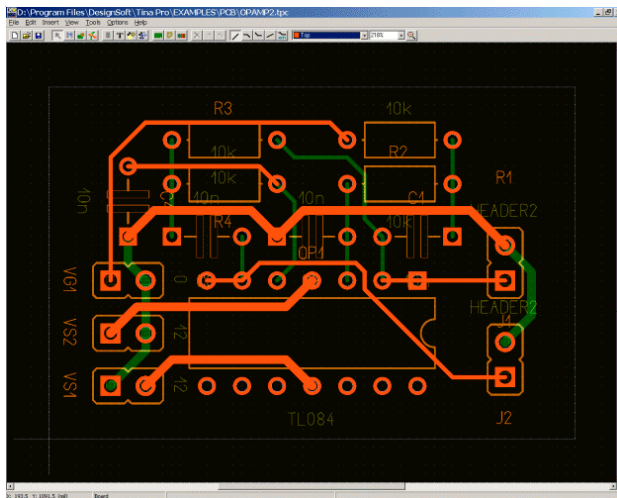
To see if everything is routed correctly, press F7 or select DRC (Design Rule Check) from the Tools menu. The following message will appear:

To finish our first simple design, let's add a text box to the silkscreen/ assembly layer. To do this, click the T button on the toolbar. The following message will appear:
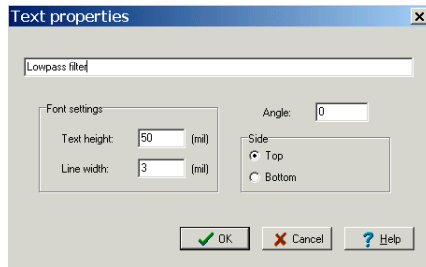
Enter the text into the empty upper field and press the OK button. The text will be attached to the cursor. Move it to the place shown on the picture below and press the left mouse button.
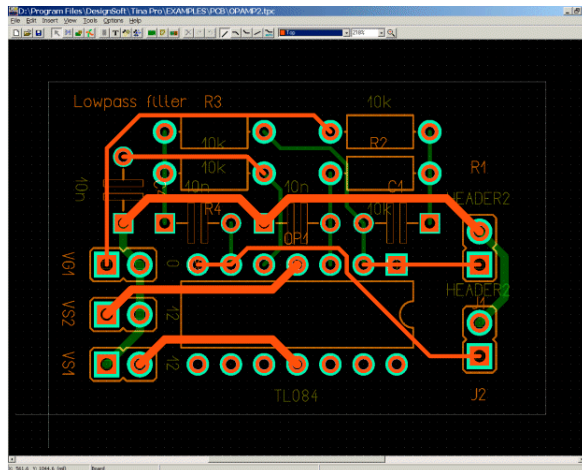
Finally, you can check your design in full 3D. To do this, press F3 or select 3D View from the View menu. After some calculation the following window will appear.
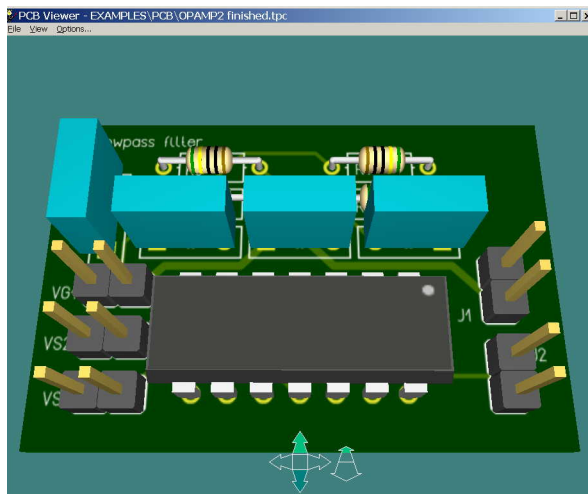
You can rotate the 3D model and zoom in and out with the control arrows at the bottom. You can display or hide these arrows through the Options menu with *"Use control arrows"* checkbox.

You can also rotate this 3D model by clicking with the mouse at any point, holding down the left button and moving the mouse. You can also move the camera forward or backward to see the whole design or just a part of it in more detail. To move the camera, hold down the right mouse button and move the mouse.

After this you can either print your design or create a Gerber file for a manufacturer.

To print use Print… from File menu.

To obtain Gerber files to direct a photoplotter, choose Export Gerber file from File menu. (Gerber option can be changed through Gerber output setting under Options menu.)

This example concludes the introduction to the use of TINA's PCB layout module. For more detailed information, a description of editing functions, the creation of multilayer PCBs, and more, see the TINA PCB Designer Manual.

We also suggest that you study the examples in TINA's EXAMPLES\PCB folder.

4.7.3 Multiple Logic Gates in the Same Package and their Power Supply

In order to simplify and reduce clutter on schematics, it is common practice to not show the power supply pins of Logic Gates and ICs on the schematic symbols. Likewise, several (two to eight, typically) gates or opamps are often manufactured in the same IC package. Although these simplifications do not have any effect on the result of circuit analysis, proper information is essential for the PCB design. In this chapter, we will show how to manage and solve these problems.

Let's consider the following logic circuit containing 6 NAND gates (EXAMPLES\PCB\Digital lock\Digilock.TSC):
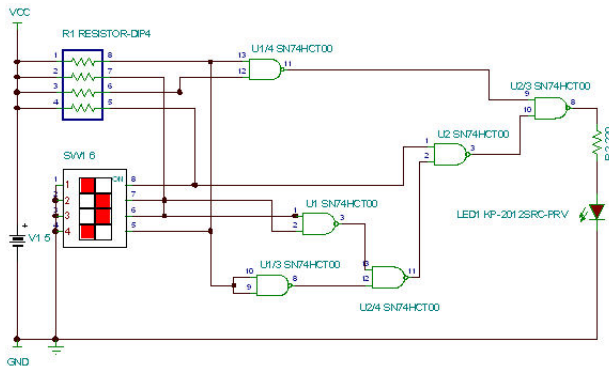
The SN74HCT00 package contains 4 NAND gates, but our design requires 6 gates. We will therefore use 2 packages labeled U1 and U2. In the U1 package we use all 4 of the gates, while in the U2 package we only will use 2 of them. The gates in the same package are labeled as U1, U1/2, U1/3, U1/4, respectively. The numbering of the gates is defined in the PCB information dialog of the parts. For example, let's see the assignment of the 3rd gate in the U1 package in the PCB

GETTING STARTED

Information dialog. Double-click on the U1/3 gate at the bottom of the schematic diagram. The property dialog of the component appears.
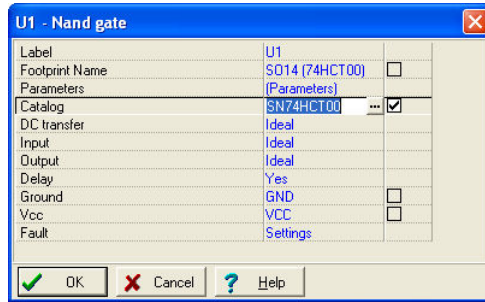
Click on the Footprint Name line and press the ••• button. The PCB Information dialog appears.
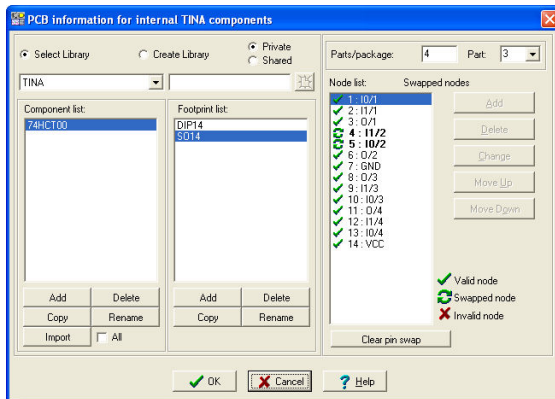
The Parts/Package assignment on the top-right part of the dialog shows that the package contains 4 gates and we are using the 3rd one.

Using the list box on the right, you can select any of the 4 instances of the NAND Gate. Be careful not to use the same gate more than once.

We will discuss in this section the assignment of power supply outputs to component power pins. In the quest for simpler, less cluttered schematics, the power pins are normally not shown, but can

DesignSoft

be seen when the part is opened up. The usual names are VCC for the power supply positive voltage, and GND for ground, the power supply negative voltage. And the usual VCC value is +5V. Recently, though, more and more ICs (like DSP chips, FPGAs, and processors) require 2 or even 3 different "VCCs." Study the chip data sheets or component properties and be sure to label the positive supply correctly. You may end up with a net for +5V VCC, another net for +3.3V VCC, and perhaps even a net for +1.8V. You could use the basic definition of VCC as +5V, a jumper labeled +3.3V, and another jumper labeled +1.8V. Jumpers in TINA are also useful for parts where the schematic symbols do not contain the power supply pins.

Let's study the Ground and VCC connections in the property dialog of the U1 package.

Ground	GND
Vcc	VCC

You will find (in the left side the schematic) the jumpers with the same GND and VCC names as in the property dialog above, establishing the connection with the package and the circuit nodes where the jumpers are connected.

Note that if the Ground and VCC fields of a component are left empty, TINA will connect them automatically to ground and +5V.

IMPORTANT: the Ground and VCC fields for all parts, even parts in the same package, must be filled out if their required voltage is different from the default (+5V).

Our completed PCB layout and its 3D view look like this:

GETTING STARTED

DesignSoft
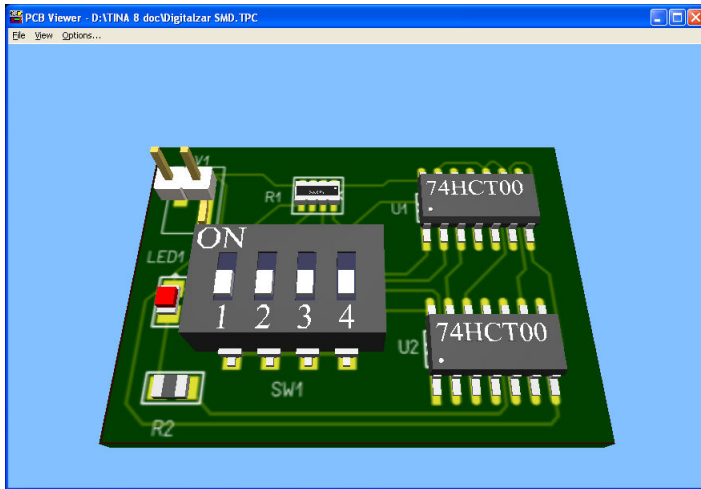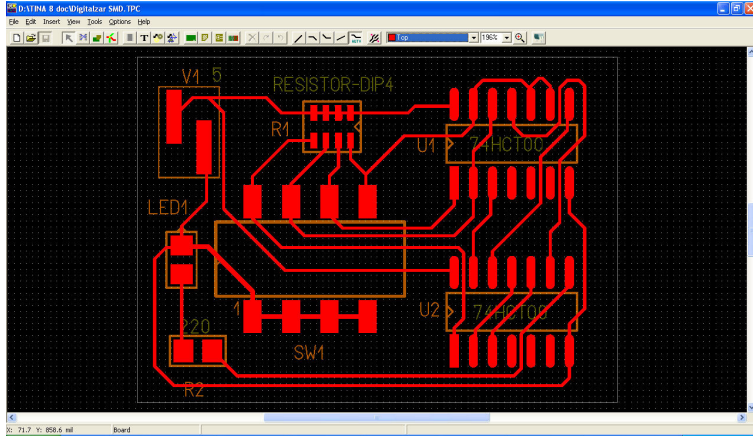
4.7.4 Creating a Flexible PCB Layout (Flex PCB)

Flex PCBs are PCBs whose electronic devices are mounted on flexible plastic substrates. They are widely used in modern electronics where space is a critical factor e.g., cameras, mobile phones, etc. TINA supports Flex PCB design, which we will introduce by way of an example. Our example will consist of a conventional rigid PCB with two flexible extensions.

DesignSoft

Let's start by loading the example file „PIC Flasher DIP4SW flex top.tpc" from the EXAMPLES\PCB\PIC Flasher flex folder.
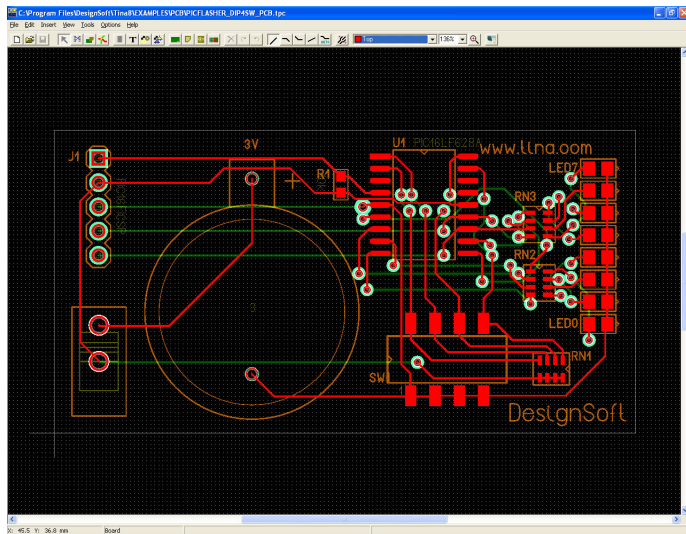
Our design requires two flexible extensions or "paddles." One paddle mounts the DIP switch and the second paddle accepts eight LEDs. (For reference, check our final result „PIC Flasher DIP4SW flex top.tpc" from the EXAMPLES\PCB\PIC Flasher flex folder.)

In our current example file, the LEDs and the switch are already routed. To do this tutorial, we'll remove their routing. You can do this manually by shift-clicking the appropriate tracks and pressing the DEL button. Alternatively, you can erase all tracks (Edit/Delete all tracks), then move the switch and the LEDs outside the board (so they will not be routed again) and run Tools/Autoroute board to re-route the other components.
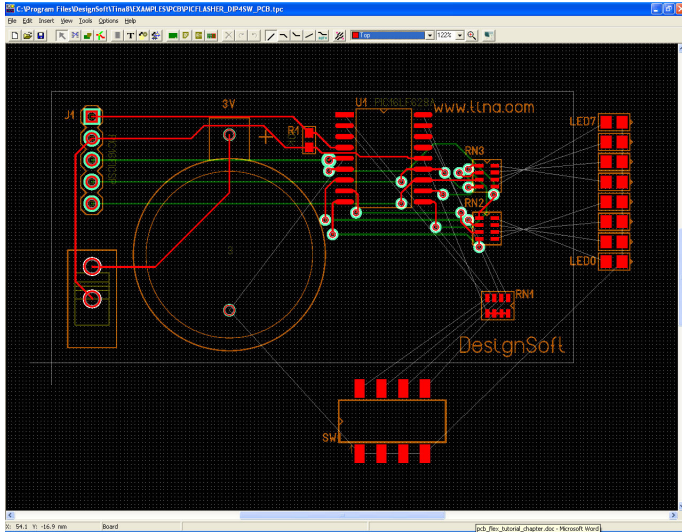
Now we are ready to set up the flexible board area. First, we have to decide whether to put the flexible part on the top or the bottom of the rigid PCB. In our case, all the SMD components are placed on the Top side, so let us also put the flexible parts of our design on the Top.

Note that we could just as well have placed the flex PCB on the Bottom layer of the rigid board. In this case, the rigid shape should be assigned to the Top layer and the flex shape to the Bottom. SMD components that go to the flex paddles should be left on the Bottom side. Routing on the flex areas is also done only on the Bottom side. Otherwise the two cases are similar. (For reference, check the final result „PIC Flasher DIPSW4 flex bottom finished.tpc".)

Before creating the flexible board shape, click on the Draw/Modify shapes button to enter Shape mode and double-click on the rigid board to edit its properties. At the bottom there is a list of layers, each of which can be assigned to the board. Make sure the Bottom layer is assigned to the rigid board but the Top is NOT (the rectangle next to it is empty), because we want to assign that layer to the flexible board shape. As for the other bottom layers (e.g., Solder Mask Bottom), they can be left on.
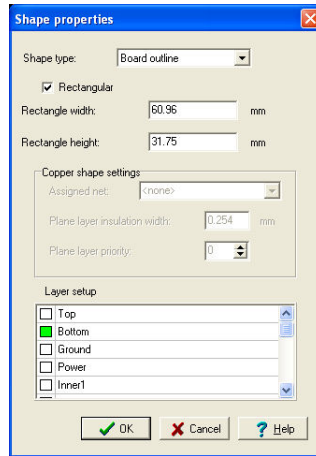
There is a more convenient way to draw the flexible board. Go to Options/System settings and set the system grid to 1 mm. (note: the rigid board was also drawn on a mm grid.)

Even though we want two paddles, we should create a shape as one piece, and that shape should overlap the rigid board because the layers will all be glued together in the manufacturing process. In our current example, the flexible shape should contain the whole rigid shape so we can use it for top side routing on the rigid part too.

To get started drawing a new board shape, select Insert/Board outline from the menu. Insert the first vertex by a single click. Make sure that you are not in the Rectangular shape drawing mode – right-click to see the pop-up menu and de-select the Rectangular mode. Then insert the other vertices of the shape.

We recommend that you give the flexible PCB bent edges. To create a 90 degree bend later, draw 45 degree cut-offs while drawing the shape.

Try to draw the paddles as in the image below. (This phase of the design is in the file "PIC Flasher DIP4SW flex top.TPC".)

DesignSoft
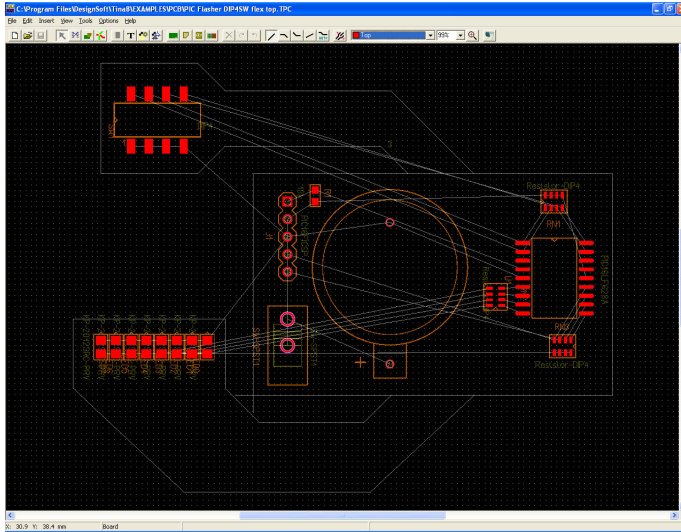
After inserting all the vertices, select Finish from the right-click pop-up menu. If you spot an error, press ESC and start again. However, you can also edit shapes later (see program Help).

After defining the flexible layer's shape, right-click on the 45 degree edges and select Bend shape edge. Bend it by moving the mouse, then left-click to finish. You could also directly enter numerical data to specify the bend. Double-click the edge and enter the bend angle in degrees.

After placing the shape, double-click on it and edit its properties – switch the Top layer ON and the Bottom layer OFF.

Now that the flexible PCB shape with two paddles is completed, we can place the components on it. In the original example, all SMD parts were placed on Top, and we should follow suit. Switch back to the first editor mode (Select/Move components/tracks), then double-click on the nine individual components to edit their properties. Move them to their final place.

Now you are ready for the final step, routing. But first, set a finer system grid (0.1 mm) in Options/System settings. Routing is done as in any other PCB design project. You can route manually (right-click on a pad, select Start/Continue track, then draw it section by section) or use the autorouter (Tools/Continue Autorouting, or press F5 button). The Auto track mode (right click while manually drawing a

track) can also be useful. Auto track mode allows you to draw the tracks one by one in the desired order and still take advantage of the autorouter. When routing manually, make sure that you only draw tracks on the appropriate side of the flex region (in our case, the Top). Don't route traces on the flex PCB's Bottom side. Only the rigid portion of the PCB can accept routing on the Bottom side.

PCB designers know that traces with right angle (90 degrees) bends can be subject to over-etching with degraded performance. It is good PCB design practice to break right angles into 45 degree angle segments. This applies to traces on flex PCBs as well.

You can bend a track section by right-clicking on it and selecting Bend track. Alternatively, you can select multiple tracks (preferably only those sections that are placed on the flex parts) and use Tools/ Bend track edges on them. This tool takes a given length of the selected tracks starting from the track vertices and replaces them with bent sections. The maximum bend length should be entered after selecting the tool. Its optimal value can be determined by experimenting. If you do not like the results, use Undo (Ctrl+Z) and try again with a different parameter or a different selection of track sections.

Your final result should look like this:

DesignSoft
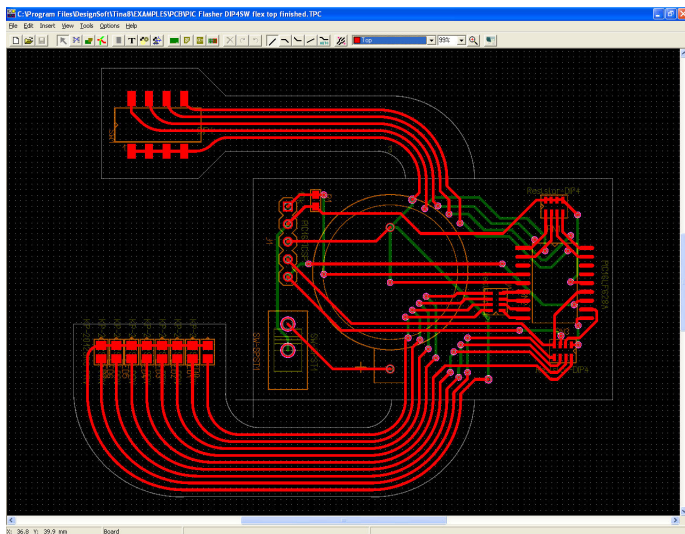
TINA can now present a 3D view of the circuit board. Press the rightmost button (3D View) in the TINA PCB Designer program see the PCB as presented in the next figure.
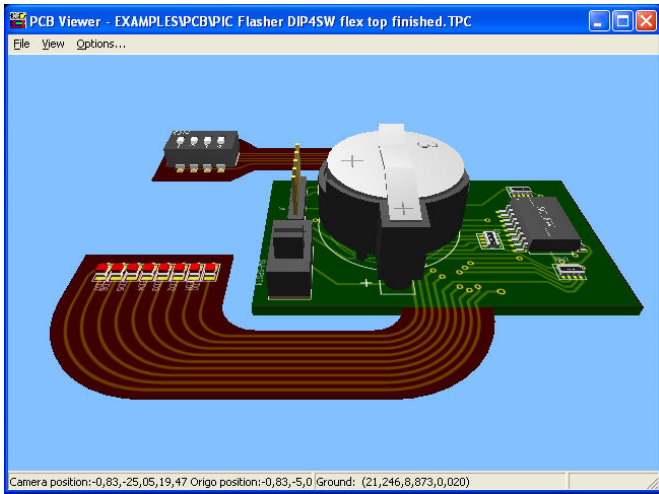
4.8   Mechatronics Extension

With the optional Mechatronics add-on package, you can create and simulate multidisciplinary designs, currently including electronics, 3D mechanics, and control engineering. You can place light sources, light sensors, motors and actuators in TINA's mechanical window and connect them to their counterparts in analog or analog-digital mixed electronic circuits. You can control the mechanics from the electronics part of TINA even with complex software written in C or assembly language. Then compile and execute the code in the MCUs while running the electronic and 3D mechanical simulations simultaneously.

In the following section, we will demonstrate the setup of the coupling between a circuit in the schematic editor and the mechanical window of TINA.

As an example, we will simulate a simple pendulum. Note that the mechanisms are created using the Newton software, which is now integrated into TINA (formerly it was only available as a standalone

DesignSoft

software package.) The detailed rules of using Newton are described in the Newton manual along with further examples.

We will build and simulate an electronically controlled "perpetual motion" pendulum (Novelty electric motor US Patent 3783550 A). The working principle described in the patent would allow us to construct many controlled devices; here, however, we will focus on the accelerated pendulum only.

A permanent magnet has been placed close beneath the pendulum. As the pendulum swings, it moves above the coil and induces an electric voltage in the coil. This voltage forces current into the bipolar transistor's base, which turns on the transistor and produces a much larger collector current. This current passes through the coil and acts as an electromagnet and accelerates the pendulum. As a result the amplitude of the pendulum's swing will not decrease (In fact, at first it will grow and then become stable), as long as energy is supplied to the controlling circuit

The controlling circuit and the pendulum are shown in the following screen.

Now let's create our first system.
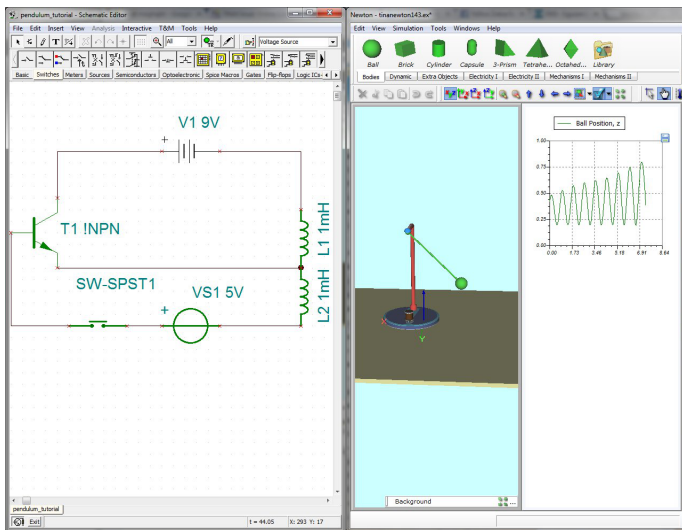
First we open a new mechatronics example.

Click on the *Analysis* menu then *Enable Mechatronics coupling*. Note that this menu item is available in TINA versions extended with the

GETTING STARTED

DesignSoft

Mechatronics add-on package only. The Newton window will appear. On the top of the Newton window there is the Object Toolbar. It contains the icons of the commonly used bodies, constraints, and accessory objects that can be used to build mechanisms. The objects are grouped logically according to function, and each group has its own tab.

Next, we create a simple pendulum.

Click on the Extra Objects tab on the object toolbar, than place a

stand onto the 3D scene by clicking on the stand icon ![Stand icon] . Select

the object - click on it - then click the Dynamic tools and add a hinge

![Hinge icon] to the stand.

Add a ball to the scene by clicking the appropriate icon ![Ball icon] in the

Bodies object toolbar. Move the ball under the hinge. Use the XY

planes ![XY icon] to lift the ball up by double-clicking on it and setting the

Location z to 0.2 m in its Properties Window.

Now link the ball to the hinge. If the hinge is selected, a stick joint will appear. Click and hold down the left mouse button on the stick, then move the cursor to the ball to be linked. When the left button of the mouse is released, the two objects are linked.

Now take a Coil from the *Electricity I* toolbar, and place it to under the ball.

Now let's create diagrams to represent the pendulum motion. First, switch the Description window to the Edit mode. Find the *Diagram*

icon ![Diagram icon] and click on it to bring up the Diagram properties

window. Select Ball – Position – z, then click the OK button.

Create the circuit below in TINA's Schematic Editor.

In the next step, we will define the coupling between the pendulum and the circuit.
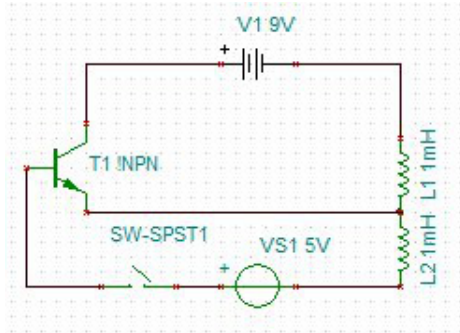
In the Newton window, select the *Tools* menu, then click *Couplings*. Click the *Add new coupling* button. From the drop-down menu select *Mechanical – Analog Converter*

Select the VS1 generator from the Tina components on the left side, then push the Editor button on the right. The Editor Window will appear. In this editor a Pascal-like language code can be added.

Enter the following code:

```
Begin
  result := 0.14/
dist(ball,coil)*sgn(ball.v[3]);
End.
```

This code will determine the voltage of the generator in TINA representing the voltage induced in the coil by the moving magnet.

The "dist" function calculates the distance between the center of masses of the ball and coil objects. This is a rough approximation for the sake of simplicity. If more precision is desired, one must use a more accurate formula.

Save the code under the name ma.cod, by pressing the Save button or using the Save command in the File menu. Close the Editor window.

Now you should assign the voltage calculated by the function to the VS1 generator in TINA. Scroll through the available codes using the control of the Code field and select ma.cod, then press the Save button. Close the Coupling Window.

Next we create a coupling between the coil object in the 3D window and the schematic coil symbol in TINA.

Press the Add new coupling button again. On the drop/down menu, click on COIL

From the Tina component list select L1, the coil where the Collector current of the transistor flows, and from the Newton component list select COIL

Now we need add a formula for calculating the "magnetic charge" (coil.charge) to be placed into the center of mass of the electromagnet (coil), which will be used for calculating the force between the electromagnet and the ball.

Press the *Editor* button and enter the following code:

```
var
N : integer; // number of turns
I : real; // current
A : real; // cross section area of core
L : real; // total length of magnetic field
C0 : real; // speed of light
m : real; // magnetic pole strength
Begin
N := 15000;
I := ABS(value);
A := 5.3E-4;
L := 0.2;
C0 := 299792458;
m := N*I*A/L;
coil.charge := m/C0;
End.
```

Save the code under coil.cod

NOTE:

> *In this simple approximation the electromagnet is substituted by a "magnetic charge" calculated from the current of the coil. This way the force between the electromagnet and the ball can be calculated simply by the Coulomb's law, included in Newton.*
> *(http://www.daviddarling.info/encyclopedia/C/Coulombs_law_for_magnets.html*
> *http://www.sciencedirect.com/science/article/pii/S1875389212000727)*

DesignSoft

Now you should assign the magnetic charge calculated by the function to the electromagnet in the 3D window. Scroll down the available codes using the control of the Code field and select coli.cod, then press the Save button. Close the Coupling Window.

Finally set some parameters.

In the TINA window, set the Simulation time unit to 1s using the *Options* command under the Interactive menu in the Schematic Editor.

In the Newton window, double click on the ball and in the Property window click the Material icon and set the Charge to 4E-4.

Drag the ball into an initial position of approx. 30° as shown on the screenshot at the beginning of this section.

Now start the simulation in interactive transient mode by pressing the TR button in the Schematic Editor. If the circuit power switch is off, the pendulum will swing as a damped harmonic oscillation. But if the switch is turned on, the amplitude of the pendulum swing increases at first, then settles down to constant value.

You can find the complete example at:

EXAMPLES\Mechatronics\pendulum\Pendulum_tutorial.TSC

You can also find further examples under EXAMPLES\Mechatronics.

GETTING STARTED

DesignSoft

USING SHEMATIC SUBCIRCUITS, SPICE AND HDL MACROS AND S-PARAMETER COMPONENTS

In *TINA*, you can simplify a schematic by turning portions of it into a subcircuit. In addition, you can create new *TINA* components from any Spice subcircuit hardware described by HDL or TouchStone format S-parameter file, whether created by yourself, downloaded from the Internet, or obtained from a manufacturer's CD. In this chapter, we show through text and examples how easy it is to do this in *TINA*.

5.1   Making a Macro from a schematic

Using *TINA*'s macro facility, you can simplify schematics and hide clutter by turning portions of the schematic into a subcircuit. *TINA* automatically represents these subcircuits as a rectangular block on your schematic, but you can create any shape you like with *TINA's Schematic Symbol Editor*.

You can convert any schematic diagram into a subcircuit - called a Macro in *TINA* - simply by adding the terminals and saving the new circuit in the special (*.tsm) format.

Now let's see how to create a macro in *TINA* through an example. Load the Half Adder example (*Half_add.tsc*) from the Examples folder of *TINA* and convert it into a macro.
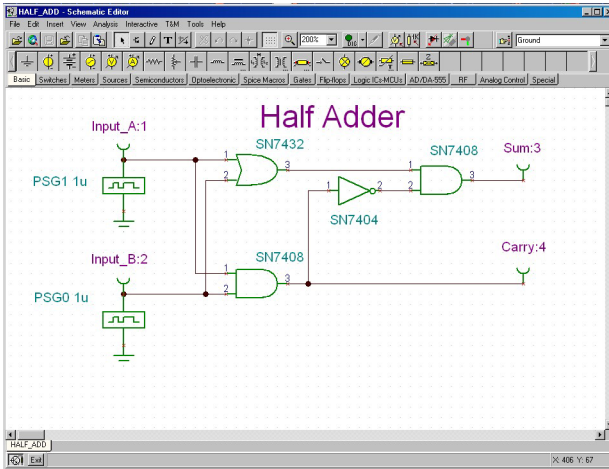
Delete the old terminals and replace them with subcircuit terminals, called Macro Pins in *TINA*. You can find and select the Macro Pins under the Special component toolbar.
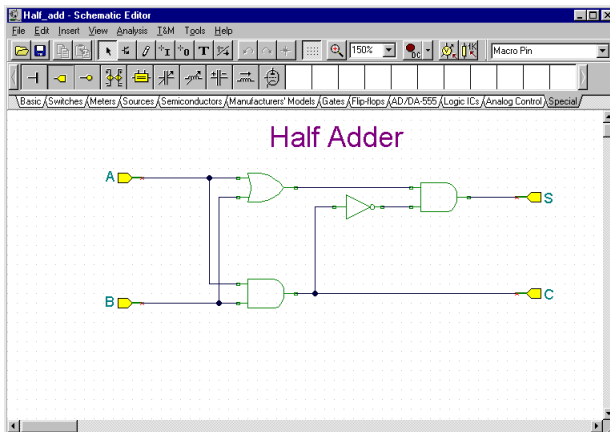
When you place Macro Pins, labels (such as Pin1, Pin2 etc.) are pre-filled in. Double click the Macro Pin and type in the new name in the label field. You can also drag the component with the mouse, or rotate it with the *[+]* and *[-]* keys or the ⟳⟲➕ buttons.

Next, create and save the new macro. Select the *New Macro Wizard* from the Tools menu. Set the Name to Half Adder (this will be displayed in the macro box that opens automatically), and set the Label to HA. This label will be displayed as the component label above the component. Note that you can leave this field blank if you don't want a component label.
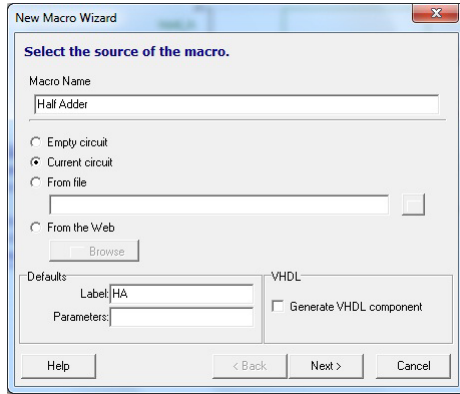
When done, press OK. A Save dialog box will appear. Set Half Adder as the File name and press Save. Note that there is already a macro with a similar name (Half_add.tsm). This has the same content as the one we just created, and is included for reference. You can also use it in the next section.
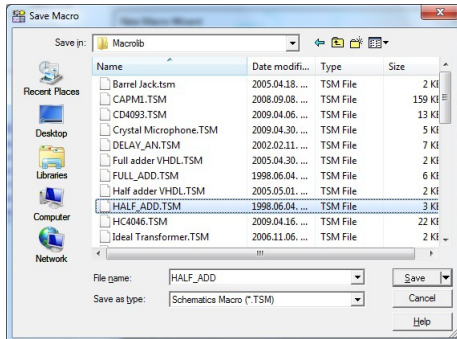
Now let's see how to insert a macro into a schematic and use it.

Clear the circuit with **File|New** or by restarting *TINA*. Select **Insert|Macro**, then our newly created *Half adder.tsm*, and click Open.
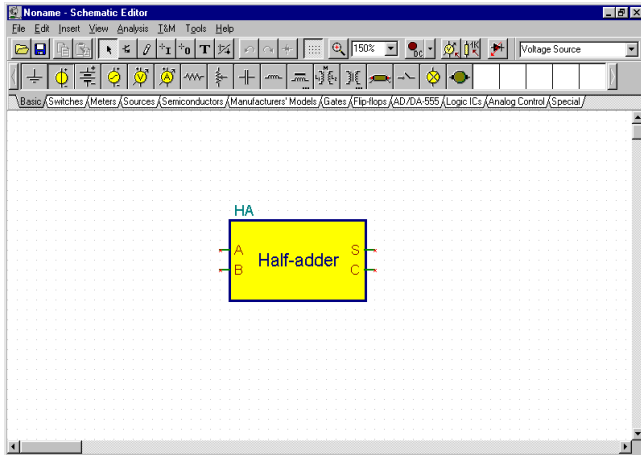
Our new macro will appear as a cursor. Move it to the center of the screen and click the left mouse button. The full symbol of the new macro will appear. Note that a rectangular schematic symbol has been automatically created, the macro name we specified is inside the rectangle, and the label name is above it.

Now you can add more components to the circuit, connecting them to the newly created macro, and start analysis as with any other circuit.

To check the content of the macro, double click on the symbol and *TINA* will display the model.

To return to the main circuit, select the Close command from the File menu.

TINA allows a hierarchical macro structure; that is, macros can contain other macros inside, and so on. Let us use our half adder macro to create a full adder macro containing two half adder macros.
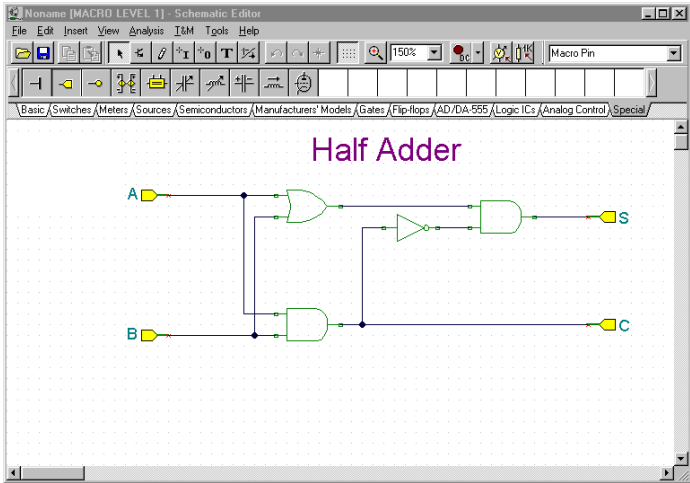
To do this, insert the newly created Half Adder twice into a new circuit and then add the additional components and wires as shown in the following picture.
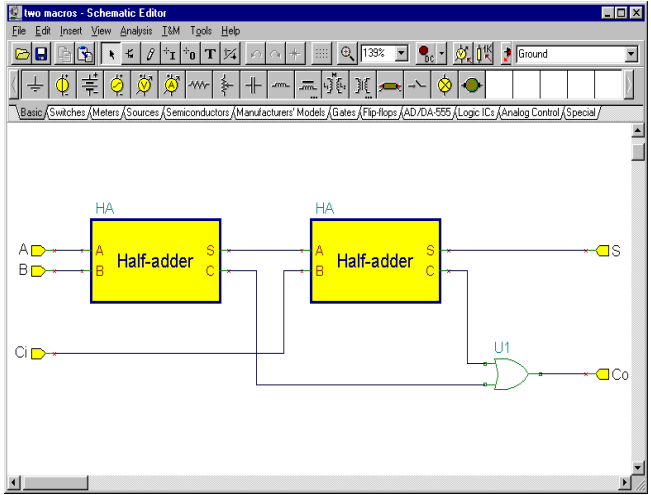
Now create and save the new macro with the *New Macro Wizard* from the *Tools* menu. At this point let us note that although the automatic symbol creation is very convenient, you can also create your own schematic symbols with *TINA*'s Schematic Symbol Editor and assign macros to them. Let us use this feature with an existing symbol. The creation of such a symbol will be described later in detail.

Set the Name to Full Adder and set the Label to FA (this will be displayed as the component label above the shape).
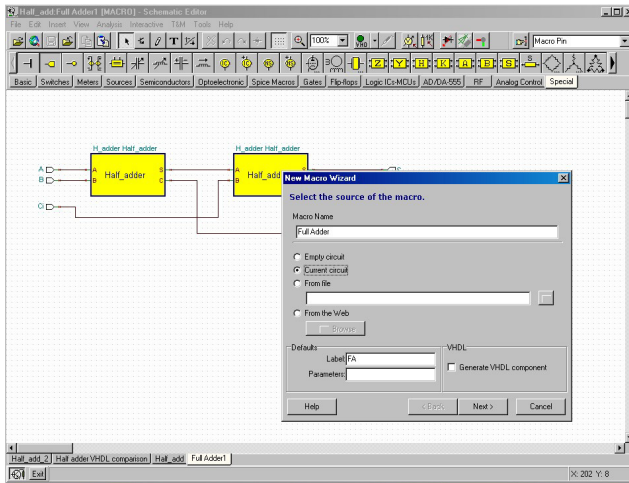
Press the Next button. The list of available symbols will appear in the Wizard as shown below.
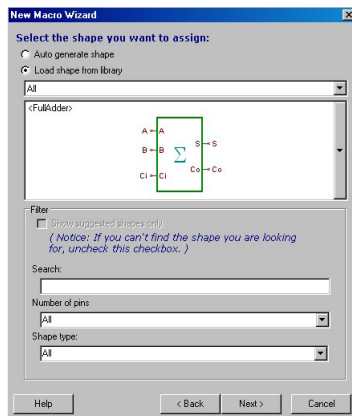
Note that in order to see the predefined symbols, the Macro Pin Label names must exactly match the names in the symbol. In our example, they must be (A, B, Ci, Co, S). If you do not see the symbol shown in the figure above, check the terminal names or try to recreate the symbol as shown later at "Making your own schematic symbols".
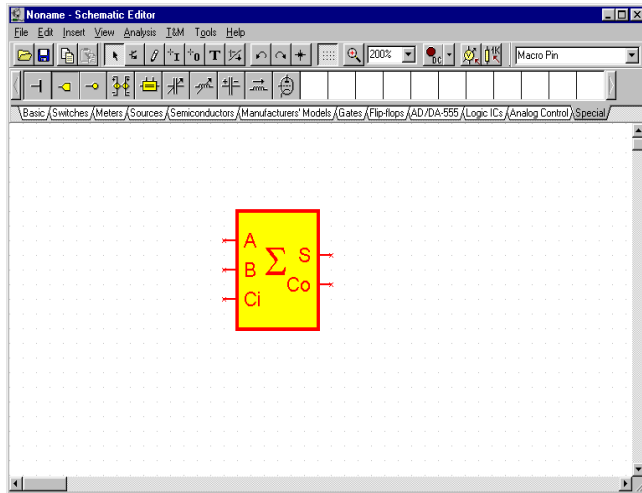
Click the schematic symbol with the large summation sign and press OK. The name of the schematic symbol will appear in the shape field of the New Macro Wizard dialog box. Finally, click OK and save the macro under the name of Full adder.tsm.

5.2   Making a Macro from a Spice subcircuit

5.2.1   Creating Spice Macros in TINA

5.2.1.1   Creating macros from downloaded files

In *TINA*, you can create your own components from any Spice subcircuit that you have made or downloaded from the Internet. Note that there are already many Spice component models in the large and extensible manufacturers' model library provided with *TINA*. The extension of those libraries is described later.

Let's create a UA741 operational amplifier using a Spice subcircuit.

To do this, select the New Macro Wizard from the Tool menu. The following dialog box will appear:
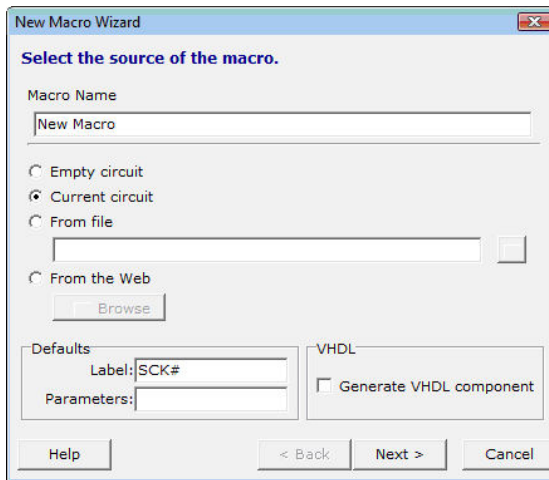
Change the settings from Current Circuit to From file and press the ☐ button. An Open dialog box will appear.
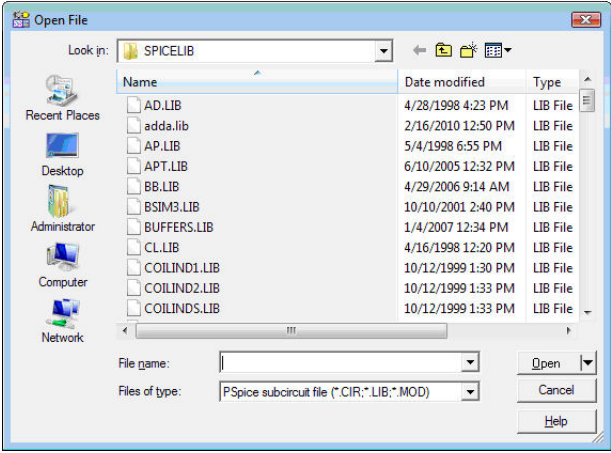
Now lets navigate to the EXAMPLES\SPICE folder of TINA using the small arrow next to the Open button.
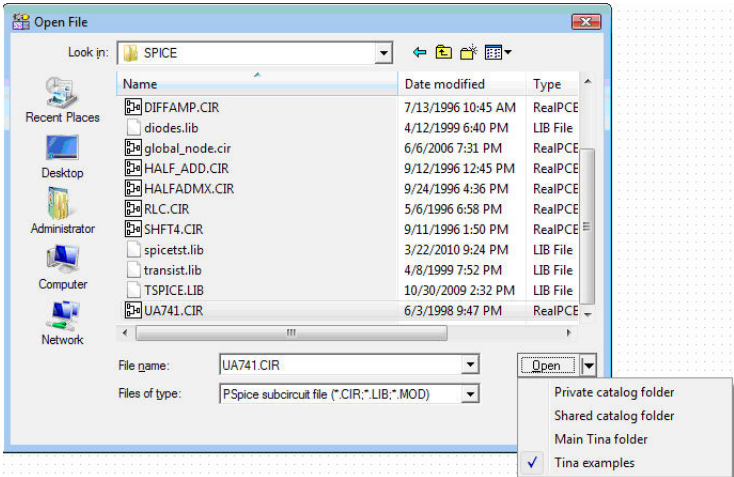
Select the UA741.CIR file and press the Open button. The New macro Wizard dialog will appar again with the path and name of the selected file. Now press the Next > button. The following dialog will appear:

DesignSoft
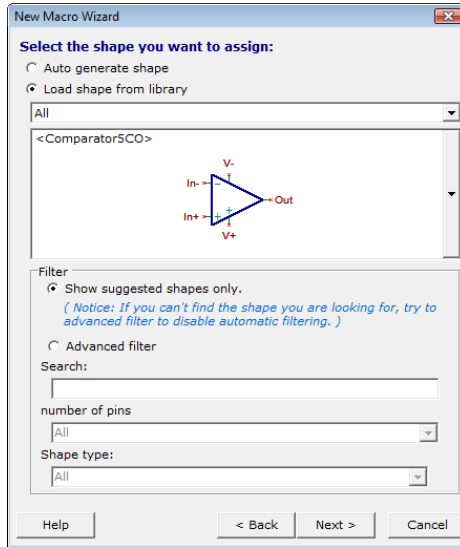
The wizard has already automatically selected the appropriate symbol. If you want something different, you can view and select a symbol by pressing the long vertical button on the right.

Press the Next button. The following dialog will appear:

DesignSoft
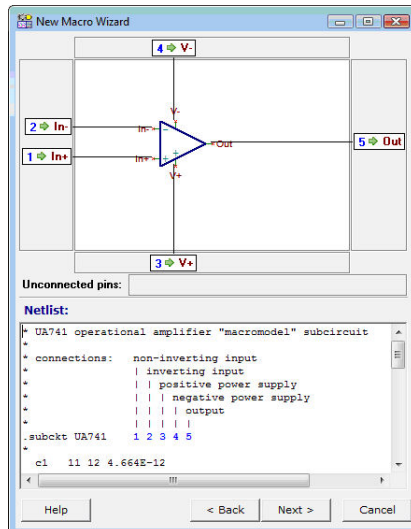
The dialog shows how the pin names on the graphic symbol are associated with the Spice node names in the macro. It also shows the text of the macro so you can check that the connections are properly made. If not, you can drag the macro node names to any terminal. However if all the pins are connected with a node number, then most likely the association is correct.

You've checked the connections; now press the Next button again. A Save dialog will appear and you can save the macro into the User macros area under **Documents\Designsoft\ TINA_Industrial_install date_id_number\Macrolib** or to the TINA macros area, under Program Files. For easier file selection, use the small arrow next to the Save button. Note that under Vista and Windows 7, you cannot normally write into the TINA program area. After saving your macro, you will see a dialog with which you can test the macro or close the wizard altogether.

Now we will see how to insert the new (or any other) subcircuit into a schematic and check its contents. Select the Macro command of the Insert menu.

NOTE:

> *You may need to select the User Macro or TINA Macro area using the selection arrow next to the Open button, or navigate there using the selection list on the top of the Open dialog.*

Click the UA741.TSM file and then press Open. Now the new macro will be attached to the cursor. Position it on the screen and drop it by clicking the left mouse button. Double click the symbol and press the Enter Macro button to see its content. The netlist editor will appear, showing the macro in detail. Note that you can modify this netlist, and the modified netlist will be saved with your circuit. However this will have no effect on the original macro; it remains unchanged.

DesignSoft

5.2.1.2 Creating macros on-the-fly by browsing the web

A more convenient way to add new models to TINA is to browse manufacturer websites and add the Spice models of interest from the web site. Of course it also possible first to download the models and use the technique described in the previous section. Note that even if you prefer the latter you may find useful hints in the following section.

Now let's select the From the web option in the wizard and press the **Browse button.**
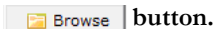
TINA's built-in internet browser will appear. Use this browser to find and select the THS4121 differential opamp from Texas Instruments. Enter www.ti.com and find the Spice macro on the TI web site using TI's Search option , or just enter following URL directly (use copy and paste if you like)

http://focus.ti.com/docs/prod/folders/print/ths4121.html (Note that the direct link above may change).

The THS4121 product page will appear. Scroll down the screen and find the link to the Spice model of this product shown below in red.
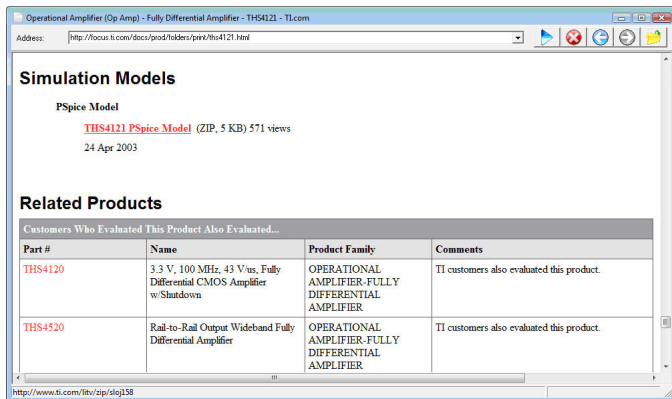
Click on the link. The following message appears in TINA:
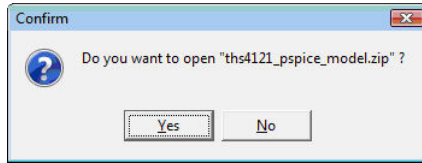
The file is compressed (or "zipped"), but TINA can open and download the files you need directly. Click the Yes button, and observe the next screen.
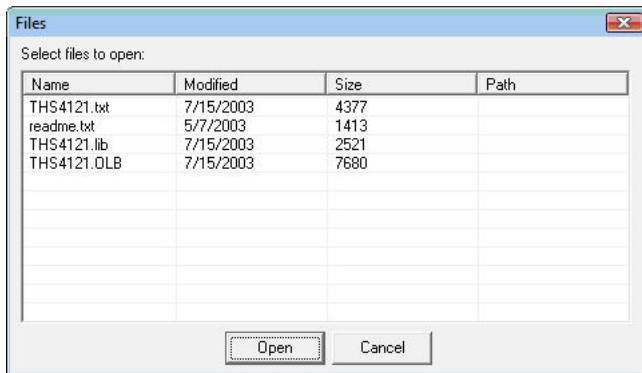
Click to select the THS4121.lib file
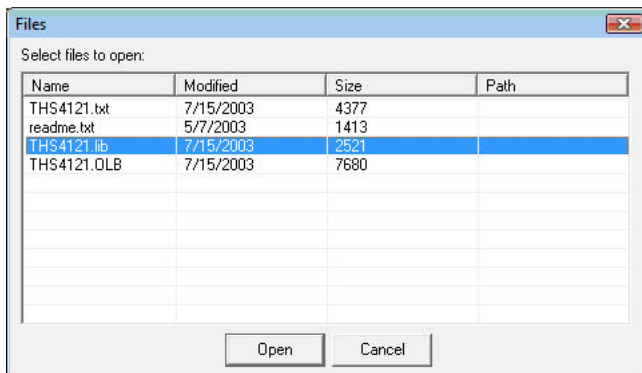
and press Open. The New Macro Wizard will appear again, with a (green) message (File downloaded press Next) confirming the successful download

DesignSoft
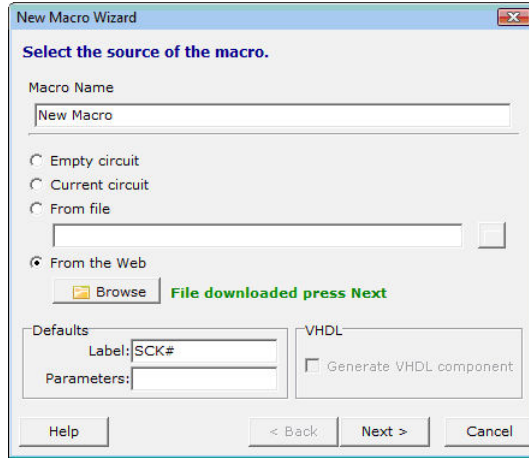
Press the Next button.
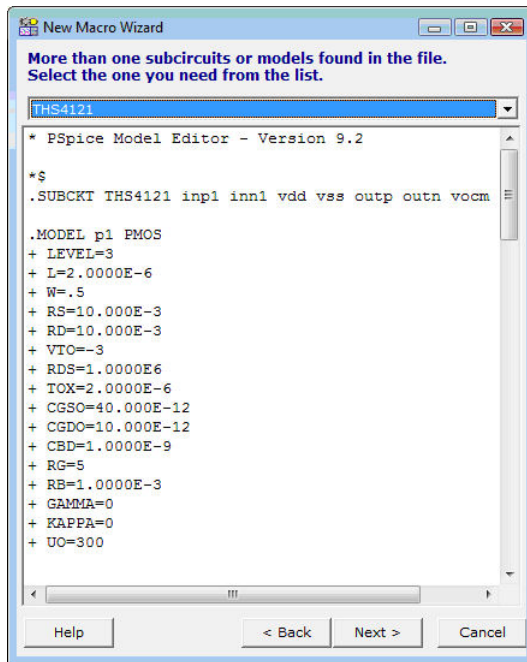
If there is more than one macro in the file, TINA presents them in a list.

NOTE:

Some manufacturers place several device models in the same file.

You can bring these device models into TINA macros with this tool by selecting them one-by-one.

If there are a lot of models, you might want to use the Library Manager tool which allows you to add all the models into the TINA catalog in one step. Learn about it in the next section.

Now press the Next button. The wizard will show the schematic symbol (shape) suggested:
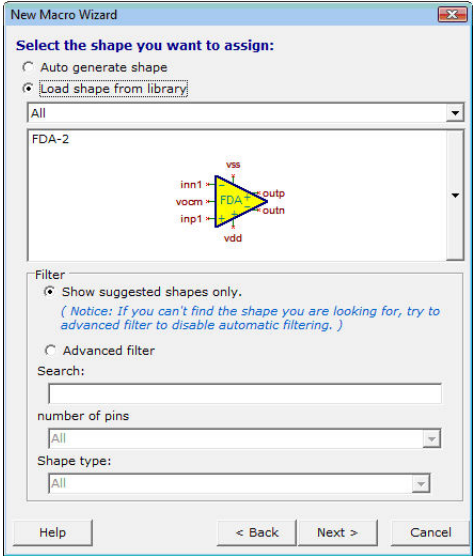
You can choose other shapes, if available, by pressing the long vertical button to the right of the symbol.

NOTE:

You may need to uncheck the "Show suggested shapes only" filter to see more shapes.

In this mode, you can also search by name, by the number of pins, and by function (op amps, comparators etc.) You must select the Shape type option before searching by function.

In our example, TINA appears to have automatically selected an appropriate shape, so you may now press the Next button. The selected shape, pin connections, and Spice macro text will appear.
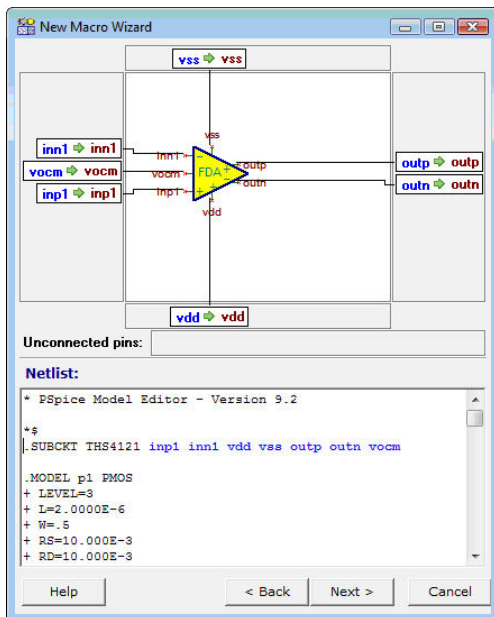
Take a moment to check the connections and if necessary correct them by dragging the connection labels.

If everything is correct, press the Next button again. The Save dialog will appear and you can save the macro into the User or TINA folder. Do so immediately, or do it later using the Insert menu.

Now let's insert a model from another manufacturer. Our choice is the LM111 comparator from National Semiconductors.

After invoking the internet browser in TINA find the Spice model page at http://www.national.com which is, at the time of writing this manual, http://www.national.com/analog/amplifiers/spice_models.

Scrolling down the page you will find the LM111 model under LM111.MOD name as shown below.

DesignSoft
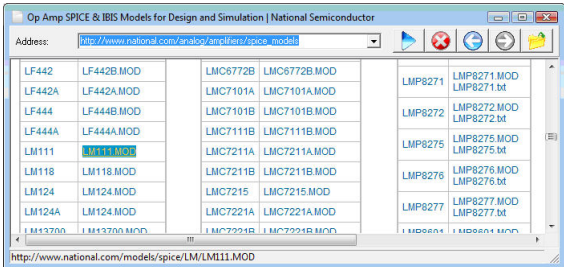
Click on the link and the text of the Spice model will appear in TINA.
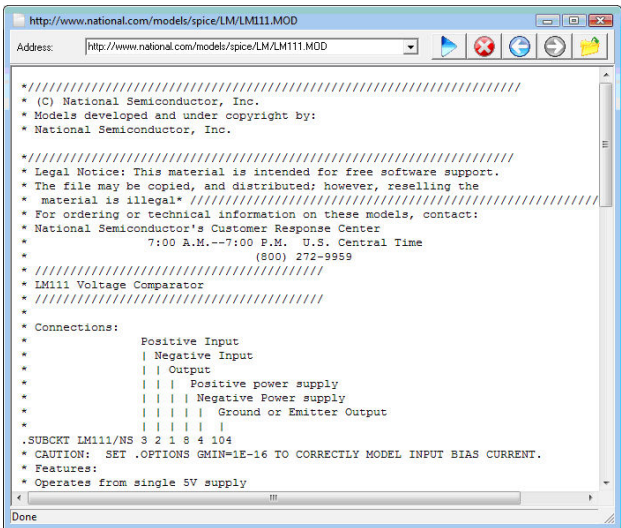
Now click on the Open icon at the top-right corner of the browser window, and the Macro Wizard will appear again confirming the successful load. Press Next. TINA will automatically present the selected symbol.

DesignSoft
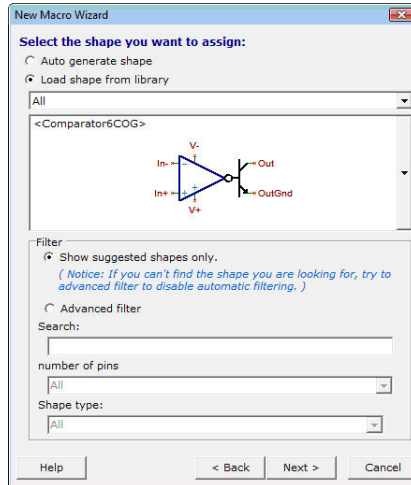
The remaining steps are the same as described in the previous section.

5.2.2 Adding Parameters to Spice Macros

TINA lets you add parameters to Spice subcircuits and set them from TINA. The parameters in the subcircuit are defined by the standard Spice syntax using the PARAMS keyword. For example look at the subcircuit in the circuit given in the MAC_RLC.TSC file in the EXAMPLES\SUBCIRC folder.
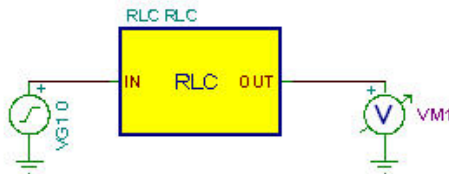
If you double-click on the RLC subcircuit and press the Enter Macro Button, the content of the subcircuit will appear:

```
.SUBCKT RLC In Out  PARAMS: R=100 L=1M C=1N
C1 Out 0  {C}
L1 1  Out {L}
R1 In  1  {R}
.ENDS
```

SPICE MACROS

DesignSoft

The parameters are R, L and C. You can set the parameters in the property dialog of the subcircuit in TINA created as described in this chapter. In our example
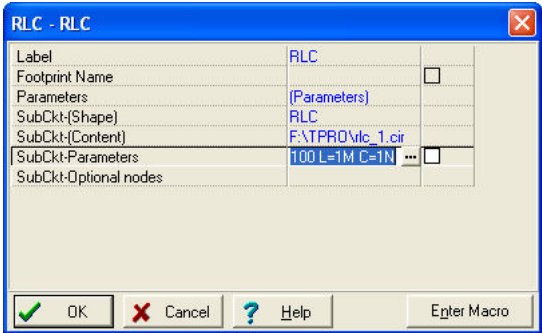
Edit the parameters either in the SubCkt-Parameters line or click the ⋯ button and the Macro Parameter Editor dialog will appear. Enter or edit the parameters you want to change and press the OK
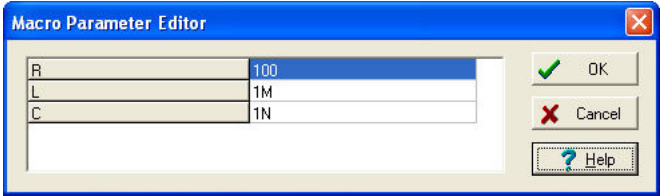
5.3 Using and extending Manufacturers' Spice model catalogs in TINA

In *TINA* you will find large catalogs of manufacturers' Spice models. In most cases, you will find that the required components are already in *TINA*'s manufacturers' model catalog. You can select components by function, manufacturer and part number.

You can also extend the library using *TINA*'s Library Manager program.

SPICE MACROS

DesignSoft

5.3.1   Using the Library Manager

TINA has large libraries containing Spice models provided by semiconductor manufacturers such as Analog Devices, Texas Instruments, National Semiconductor, and others. You can add more models to these libraries or create your own Spice Library using *TINA*'s Library Manager (LM).

Let's learn how to add a Spice model to *TINA*'s Spice libraries:

5.3.1.1   Introduction to Adding Spice macros to TINA Libraries

Start the Library Manager program. Use the Windows Start menu to locate the TINA folder and click on its icon. Select Collect subcircuits and models from the File menu. Find the EXAMPLES\SPICE folder (in the program folder where TINA is located) in the dialog box, click the SPICE folder where our example subcircuit -a ua741 amplifier model- has already been placed, and press Next.
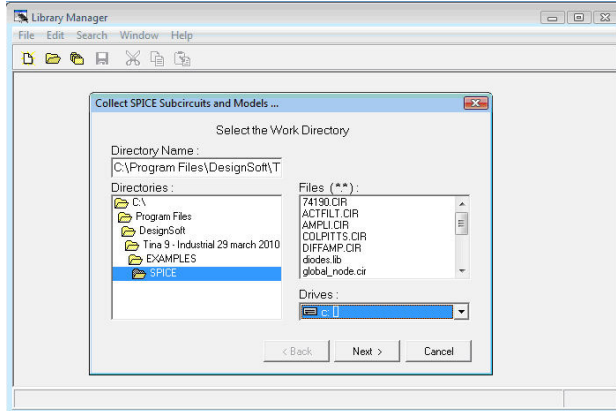
A new dialog box will appear with the list of available files on the left side. Note that the file you select must be a proper Spice subcircuit. Click UA741 and then press the > button. The UA741 model you have selected will appear on the list of selected files. In a similar way, you can select more files or even all the files by pressing the ⟩⟩ button.
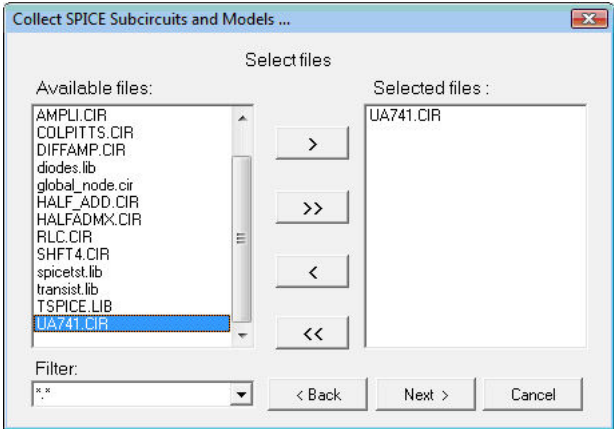
Press the Next button to continue. The following dialog box will appear.
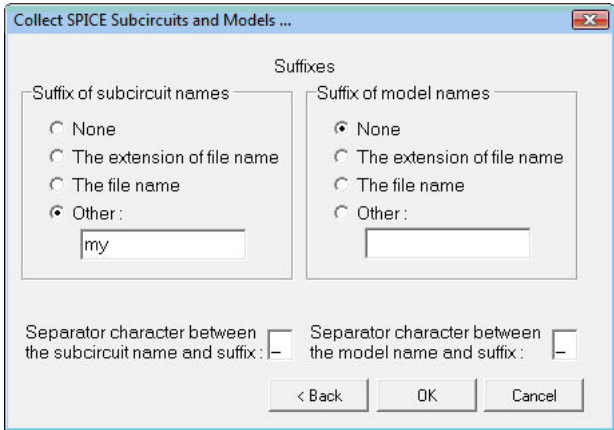
With this dialog box you can make changes in the subcircuit or model name. This might be necessary to avoid conflicts among different subcircuits or model versions with the same name. To differentiate the new model, you can add the file name as a suffix to the subcircuit name or add any text as a suffix to the name using the Other option. Let's add the suffix "my" to the subcircuit name and then press the Next button. The contents of the new library file will appear.
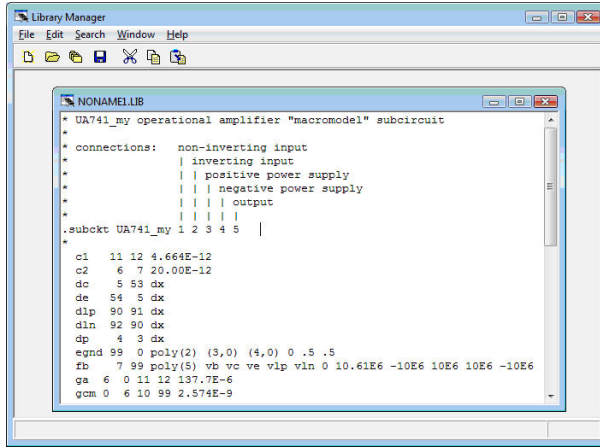
NOTE:

The new name of the subcircuit has the suffix "my": UA741_my.

Using File|Save As, save this library with the name my_741.lib in the Spicelib folder under the Windows Documents folder, for example: **Documents\Designsoft\TINA_Industrial_install date_id_number\Spicelib**

This folder is automatically created and set by TINA.

NOTE:

Under Vista and Windows 7 and later operating systems new libraries must be created in the user area under the Documents folder of Windows because the Program Files folder is normally write protected in these operating systems. This is also useful because the user data and the original libraries of TINA are separated this way.

DesignSoft

Now select *Create TINA Library Description* from the *File* menu and select select …for Spice models and subcircuits. The following dialog box will appear.
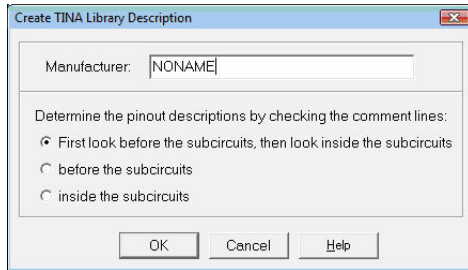
Here you give your new library a name– a name that will appear in the Manufacturer field of the Spice macro insertion tool. Let's change this to My Spice Library.

You can also specify some search options for determining the pinout description of the Spice model. The default setting is usually satisfactory. Press the Help button for more information.

The description (directory) of the new catalog will be displayed in a new window:

DesignSoft

SPICE MACROS
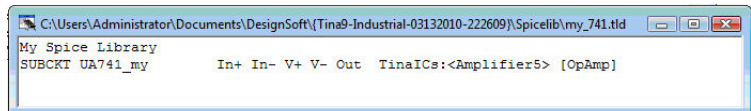
NOTE:

If everything went well, as it did above, you should not see any warnings (in red) such as "No processable statements" or "Autoshape" which would mean the Library Manager could not fully resolve the task fully automatically. If any of these warnings appear, read the next chapter for the resolution.

Finally, save the library directory as my_741.tld in the same Spicelib folder described above. Note that the Save As command applies to the active (selected) windows only.
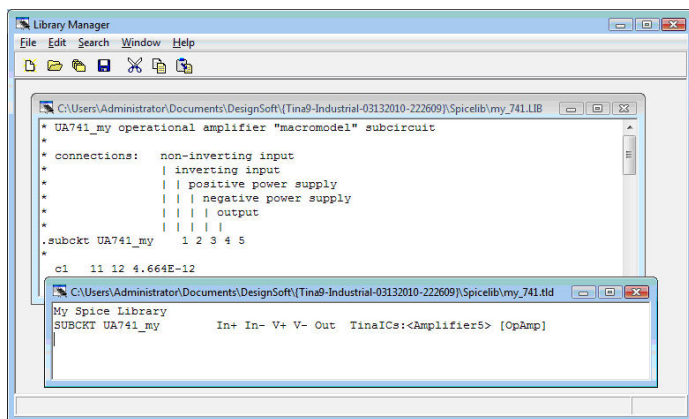
Next time you start *TINA*, this message will appear:
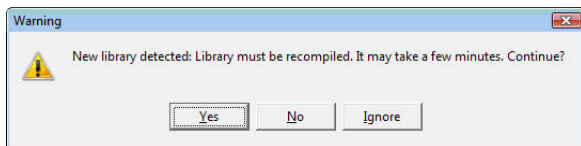
Press Yes to add your library to TINA's libraries.

NOTE:

*If this message does not appear for any reason (for example due to file date format differences), re-compile the libraries manually using the command **Re-compile Library** from the Tools menu. You can also recompile the library already in the Library Manager. Select Create TINA Library and Compile Library in the Library Manager File menu. In this case the warning will not appear.*

Now click on the Spice Macros tab and on Operational Amplifiers on the component toolbar. Your new component library should appear here in the list of Manufacturers . To access your new subcircuit, select either "My Spice Library" or All. If you have selected All, simply press U to jump directly get to the U's where it will be easy to find the UA741MY on the list. If, on the other hand, you had selected My Spice Library, the list would, of course, contain only your new opamp.
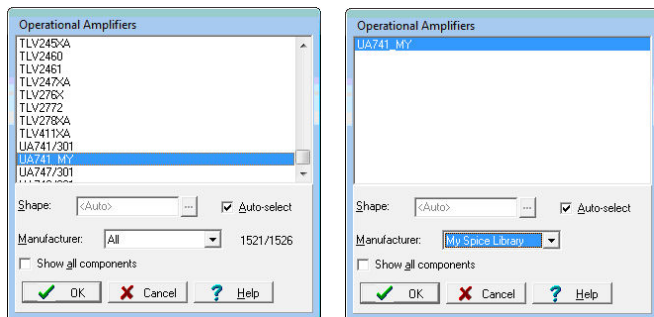
5.3.1.2 Problems and solutions while adding Spice macros to TINA

In many cases, adding models to TINA is as easy as described above, but in some cases it is impossible to find the connection between the Spice models and their graphic symbols automatically. Fortunately, TINA's latest Library Manager makes it easy to solve this problem.

Let's add the librar y "SPICETST.LIB" from the EXAMPLES\SPICE folder to TINA.

NOTE:

> *You can navigate among TINA's folders using the small arrow next to the Open button.*

DesignSoft

First, start the Library Manager as described above. Open the "SPICE TEST.LIB" file using the Open icon or the Open command from the File menu. Select the "Create TINA Library Description"command in the Spice Models and Subcircuits submenu and in the appearing Create TINA Library Description dialog window enter the name of your new library (spicetst) , the name that will appear in the Manufacturer field of the Spice macro insertion tool of TINA) as described above.

You will see the following window:

Looking at the lines of the "spice test.tld" file containing the library description, it seems that the first model, xMAX4200, was recognized automatically, since both the graphic symbol and the category were found.

For the second model, the xMAX4147, no graphic symbol was assigned and its category was not recognized; however, the program recognized its terminals.

For the third model, the xmoc223 Optocoupler, nothing was recognized. Note that even if the program does not recognize a model, it is placed in an automatically generated box (Autoshape) and still can be used.

NOTE:

There are also 4th and 5th models on the list. However, if you examine the Spice source in the spicetst.lib library, you can see that these are auxiliary subcircuits of the Optocoupler. We will address this issue later in this chapter.

SPICE MACROS

Now let's add appropriate graphic symbol the models.

From the Edit menu, select the "TLD Editor for Subcircuits" command. The following dialog box will appear:
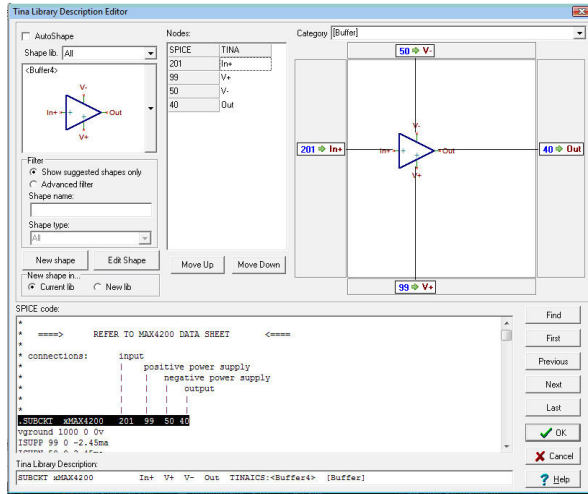
TINA presents the graphic symbol with the names of the terminals on the left. Next to the graphics symbol under Nodes: you can see the list of terminal nodes and the associated terminal names of the graphic symbol. You can move the graphic terminal names up or down by simply dragging them or by using the Move Up and Move Down buttons. You can also simply drag the terminal names to any terminals on the larger work area to right of the node list. If you drag a terminal name over another terminal and drop it by releasing the left mouse button, the two names will be interchanged. You can also shift the symbol and zoom in and out by holding down the left or right mouse button accordingly and moving the mouse. This is useful with larger and/or complicated symbols.

In the top right corner, TINA displays the category, which you can also change.

At the bottom of this dialog, the SPICE code of the selected component is shown, while below that, in the "TINA Library Description" field ,TINA displays the actual content of the line of the .TLD file. With the buttons First/Previous/Next/Last, you can move among the models in the library.

Check that all entries for the first model are correct.

Now jump to the second model by pressing the Next button.

The first things you should notice are the square Autoshape graphic symbol and the red TLD line at the bottom. These indicate that the Library Manager could not fully understand the model.

Uncheck the Autoshape checkbox at the top-left corner of the dialog. The dialog will change as follows.

Strange, but it now appears that the Library Manager has found a correct symbol. What's happening here? What was the problem before? The reason was that there were several symbols available with the same number of pins.

Press the vertical button at the right side of the Shape window and review the list of available symbols that appears. You can choose another symbol by clicking on it. The Library Manager seems to have selected the best option, but in general it is not guaranteed.

Check the list showing the connection between the Shape nodes and the Spice terminals. It should be correct since the Library Manager did not give an error message for the Spice comments. If there had been a discrepancy, you would have seen an error message in the TLD line: "No processable comments".

Change the Category at the top right corner to <Comparator >.

Now click the Next button again to bring in the last model in this library. The following window will appear:
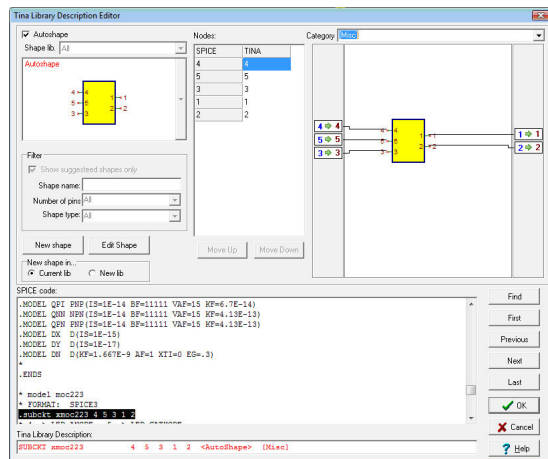
Note the "No processable comments" message in the bottom-line. This means that the Library Manager could not identify the Spice terminals from the comments in the Spice model. If you look at the comments in the Spice code of the component, you can see they are

DesignSoft

not common in the world of Spice macros (e.g. LED ANODE), this is why the Library Manager could not identify them. We will have to make the connection between the graphic shape and the Spice terminals manually.

First uncheck the AutoShape checkbox. You will see the following window:
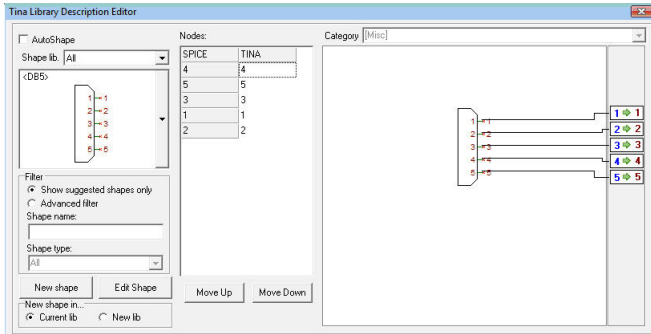
Obviously this is still not the right choice. To find the right symbol, uncheck the "Show suggested shapes only" filter and from the Shape type list select Optocouplers. Now the right symbol will appear:
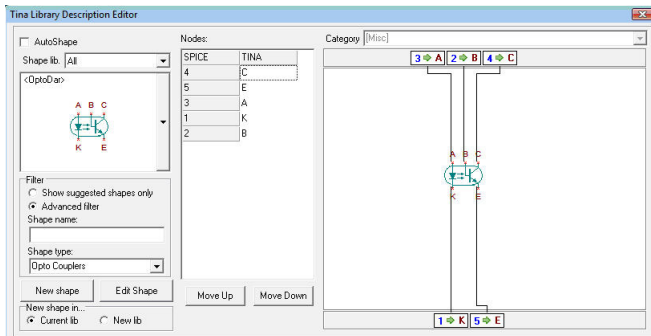
We're not done yet: the Spice terminal nodes and the Shape terminals are not properly mapped. For example the first node in the list, node 4, is associated with the Collector of the transistor, but according to the comments in the Spice code, it should be connected to A, the Anode of the input LED.

In the list under Nodes: click on A and drag A to the top of the list (to Node 4) , then drag K to Node 5, and finally drag E to node 3. Note that you have the option to make the same changes graphically by dragging the labels on the right side.

Check the remaining two nodes, which in this case must be correct. Change the Category at the top right corner to <Optocoupler >.

As mentioned earlier,, there are also 4th and 5th models on the list. However, if you examine the Spice source in the spicetst.lib library you will see that these are auxiliary subcircuits of the Optocoupler , so we do not need them separately in our schematics in TINA. To keep them from being listed in the TINA catalog, invoke them one after the other by pressing the Next button and setting their Category to Internal.

 This completes our editing of the various models. Press OK to close the TLD editor. The SPICE TST.TLD window should be updated and look like this:


```
C:\Program Files\DesignSoft\Tina 9 - Industrial 23 March 2010\EXAMPLES\SPICE\spicetst.tld
spicetst
SUBCKT xMAX4200      In+  V+  V-  Out  TINAICS:<Buffer4>  [Buffer]
SUBCKT xMAX4147      In+  In-  V+  V-  Out+  Q1=Sense+  Out-  Q2=Sense-  TINAICS:<Comparator8DO>  [Comparator]
SUBCKT xmoc223       C  E  A  K  B  TINAICS:<OptoDar>  [OptoCoupler]
SUBCKT delay         Vin1 Vin2 In+  <AutoShape>  [Internal]
SUBCKT moc223_dar    1  2  3  CONNECTORS:<AutoShape>  [Internal]
```

Using the File|Save As command, save both the SPICE TST.TLD and the SPICE TST.LIB files in the Spicelib folder under the *Windows* Documents folder, for example: Documents\Designsoft\TINA_Industrial_install date_id_number\Spicelib

This folder is automatically created by TINA at installation. You can navigate to this folder by using small arrow next to the Save button and selecting Private catalog folder.

NOTE:

> *You can also use the Shared catalog folder which you can setup at installation of TINA. Using the latter you can allow all users to use the new libraries in case they have access normally through local network to the Shared catalog folder.*

Finally, use the File|Create TINA Library command |Compile Library to register the changes for TINA. In this case when you restart TINA there will be no need to recompile the library.

Close the Library Manager.

When you restart TINA you can find these new models by looking in the appropriate category (Comparators, Buffers, and Optocouplers). These new parts will be at the end of the list, since the names of the new models start with X.
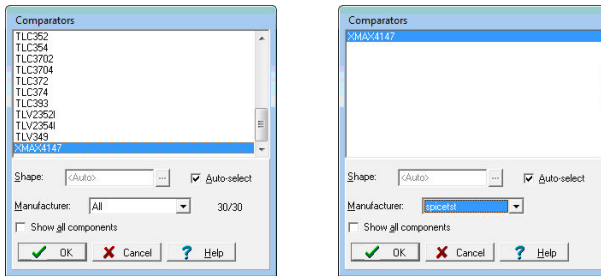

Of course, you can also set the Manufacturer in the appropriate category to "Spice test" to see only the newly added components.

5.3.1.3 Adding Spice models in .MODEL format to the library

In the previous example, you added a component described by a Spice subcircuit. You can also add diodes, transistors and other devices by simply using .MODEL instructions. These devices are normally placed in a file containing many .MODEL instructions. In TINA, there are two such sample libraries, called diodes.lib and transistors.lib

First, open the diodes.lib file from the EXAMPLES\SPICE folder using the File |Open File command or the corresponding icon on the toolbar. The following window listing the contents of the file appears:
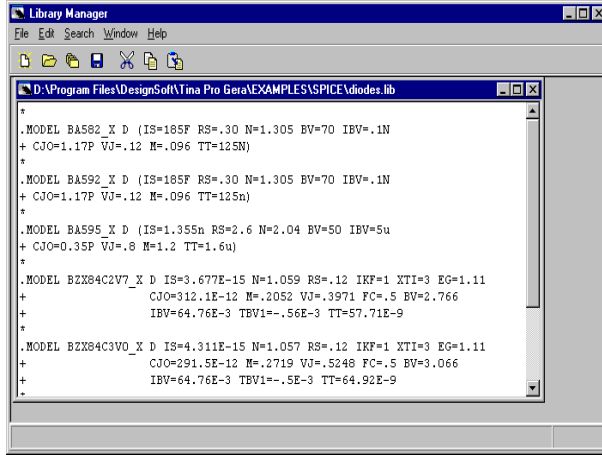
Execute the File/Create Tina Library Description/For SPICE models and libraries... command.
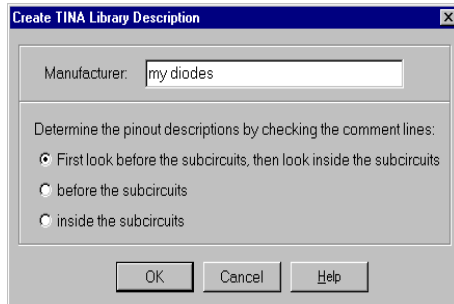
Do not change the other settings. Press OK. A description listing models in the new TINA library will appear:

DesignSoft
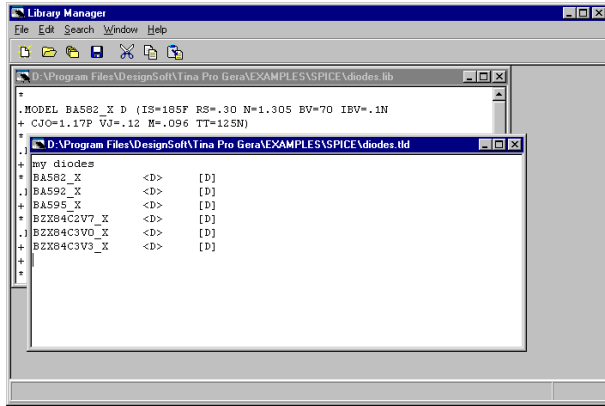
The file contains 3 normal and 3 Zener diodes. In the Spice language, there is no difference between normal, Zener, LED, Schottky, Varicap, and other diodes. However, in TINA, you can assign different schematic symbols to these types. To do this, select Categorize Components from the Edit menu. The following dialog box will appear:
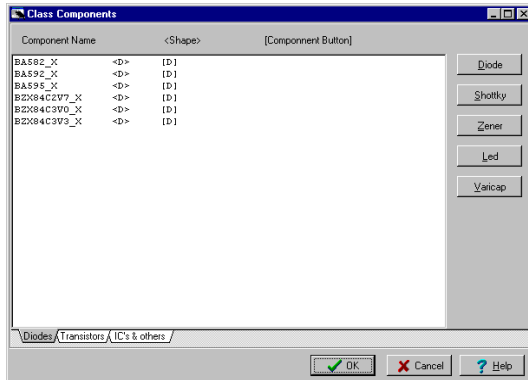
Select the Zener diodes (the last 3 items on the list) by clicking them one by one while holding the Ctrl key. Then press the Zener button.

<D> and [D] identifiers will change to <DZ> and [DZ] ensuring the use of the Zener diode symbols in TINA. Press OK and save both library files in the SPICELIB folder at

Documents\Designsoft\TINA_Industrial_install date_id_number\Spicelib

DesignSoft

To verify the new diodes, restart TINA, select diodes or Zener diodes from the toolbar, drop the diode onto the schematic, and double click on it. Press the ▣ button at the type line and select the "my diodes" library using the drop down menu of the Library field at the left top corner of the Catalog Editor dialog box.
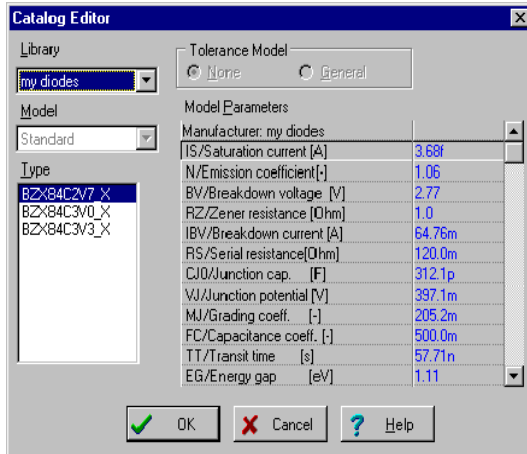
You will also find your new "normal" diodes under the Diodes category of the toolbar. Note that you can add new diodes to any existing manufacturer catalog if you select a library name already in the drop down menu (e.g., Motorola.).

In a similar way, you can try adding transistors given by .MODEL Spice commands to TINA using the transist.lib library. There is no need to categorize these components since the NPN and PNP transistors have different notations in Spice.

SPICE MACROS

DesignSoft

5.4   Adding S-parameter models

Let's learn how to add an S-parameter model to TINA's libraries.

Start the Library Manager program. Use the Windows Start menu to locate the TINA folder and double click on the Library Manager icon.

Next, select Collect S parameter files... from the File menu.

Note: You should not use the File|Open… command to collect S parameter files.

Find the folder EXAMPLES\RF in the dialog box and double-click on the SPICE folder. This is where our example, an S-parameter transistor called s_bfp405.s2p, has already been placed. Press the Next button. Note: the S-parameter files must have either S1P or S2P extension (the manufacturers use the same convention). If the extension is S1P it means that the device is a "1-port" (described with 1 parameter) otherwise a "2-port" device (described with 4 S-parameters).

A new dialog box will appear with the list of available files on the left. Note that the file you select must be a proper S-parameter file. S-Parameter data files are in the TouchStone format. This is a typical data segment of a two-port file:

S-Parameter file description

MHz S RI R 50

0.30 0.02 -0.05 -0.03 -0.02 -0.03 -0.02 0.02 -0.05

0.31 0.03 -0.06 -0.02 -0.01 -0.02 -0.01 0.03 -0.06

0.33 0.04 -0.07 -0.01 -0.03 -0.01 -0.03 0.04 -0.07

....

The first line is a header that gives the frequency units, parameter, measurement format, and characteristic impedance of the measurement (here, 50 Ohms).

The first column is the frequency in Hz. The next columns are, in order, S11 Real, S11 Imaginary, S21 Real, S21 Imaginary, S12 Real, S12 Imaginary, S22 Real, S22 Imaginary. One-port data files are similar to the two-port files, except that there are no columns for the S21, S12 and S22 parameters.

DesignSoft

Click on s_bfp405.s2p and then press the > button. The s_bfp405.s2p model you have selected will appear in the list of selected files. In a similar way, you can select more files or even all the files by pressing the >> button. In the next dialog box you can change the model name. This might be necessary to avoid conflicts among different modelversions with the same name. To differentiate the new model, you can create a model name from the file name or from one of the first 8 lines, or you can add a prefix or suffix to the model names. Let's just use the file name as a model name. Press the Next button and the contents of the new library file will appear.

Using File|Save As, save this library in the SPICELIB folder found at **Documents\Designsoft\TINA_Industrial_install date_id_number\Spicelib** using the name myslib.lib. Now select Create TINA Library Description...|...for S parameter models from the File menu.

In the following dialog you specify a name for your new library, e.g., My S Parameter Library. You could specify the name of the manufacturer as a library name, but note that if there already is a library in TINA with the same name (e.g., Siemens), then your new model will be added to this library. The library descriptor file of the new catalog will be displayed in a new window. However, in the case of S parameter files, you must always categorize the models (unless you want them to appear among the mixed components with a default shape). To do this, select Categorize Components from the Edit menu.

Press the IC's & other unrecognized components tab. Select one or more models from the list, then press a Move to page ... button (pick the button for the model type of the selected model). In our case, press Move to page Transistors, then click on the Transistors tab. Now select the appropriate category, which for this model is NPN.

Save the library descriptor file as myslib.tld in TINA's SPICELIB folder. (Both Spice and S parameter libraries are stored in this folder.) Note that the Save As command applies to the active (selected) windows only.

DesignSoft

Finally, use the File|Create TINA Library command to register the changes for TINA.

Next time you start TINA, select RF components, and then NPN RF Bipolar Transistors and you will find the new component library in the list of Manufacturers. Your S parameter model will appear on the list invoked either by selecting "My S Parameter Library" or All.

5.5   Making a HDL macro from a file

You can create a HDL macro from any .vhd / .v / .va / .vams file that contains an entity (interface to the outside world) with its architecture (description of the hardware). Files with .vhd extension are VHDL files, with .v extension are Verilog files, with .va extension are Verilog-A files and with .vams extension are Verilog-AMS files. The ports declared in the interface part will automatically appear in the macro symbol (shape). By default, the input ports of the interface will appear on the left side of the generated macro shape and the output ports of the interface will appear on the right side, but by editing the generated macro you can change this arrangement.

For example (VHDL):

```
ENTITY e_Half_add_entity IS PORT(
  A : IN std_logic;
  S : OUT std_logic;
  C : OUT std_logic;
  B : IN std_logic );
END e_Half_add_entity;
```

In this case the A,B ports will appear on the left side and the S,C ports will appear on the right side of the macro shape.

Lets see how to do a macro from the following VHDL code (a half adder):

```
LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
USE tina.primitives.all;
```

_____

S-PARAMETERS

```
— entity section
————————————————
ENTITY e_Half_add_entity IS PORT(
  A : IN std_logic;
  S : OUT std_logic;
  C : OUT std_logic;
  B : IN std_logic );
END e_Half_add_entity;
————————————————

— architecture section
————————————————
ARCHITECTURE a_Half_add_arch of e_Half_add_entity
constant delay : time := 20 ns;
BEGIN
 S <= (A xor B) after delay;
 C <= (A and B) after delay;
END a_Half_add_arch;
```

1. Select Tools/New Macro Wizard...
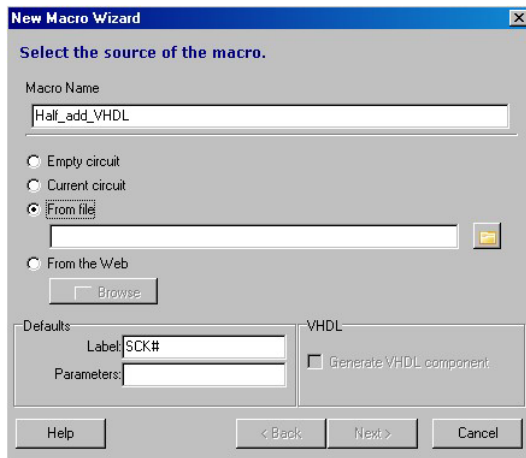
2. Type a name for the new macro.

3. Select the From file option then press the Open button and change the file type to VHDL, and navigate to EXAMPLES\VHDL in the TINA program folder. You should see the half_adder_VHDL.vhd file in the open dialog. Select this file and press Open.

S-PARAMETERS

DesignSoft

4. Press the Next button to save the macro, and save the macro into the default Macrolib folder.

5.5.1 Placing a HDL macro in the schematic editor

Now let's see how can we insert our previously saved VHDL macro into TINA's schematic editor.
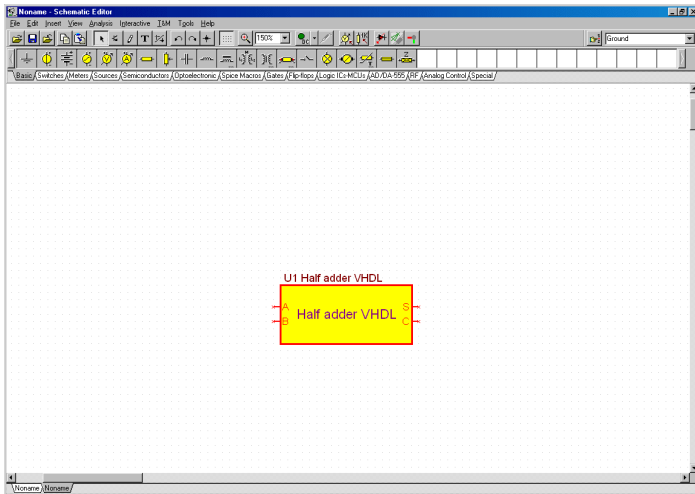
1. Select "Insert/Macro..." from the menu and select the previously saved macro half_add_VHDL.TSM from the MACROLIB folder of TINA's main program folder.

The screen will look like this:

To see the content of the macro double-click on it and press the Enter

Macro button on the property dialog that appears. The content of the macro will be displayed.

NOTE:

> *The command section starting with TINA VHDL Macro Description is generated automatically and should not be changed.*

5.5.2 Testing a HDL macro

Let's test our newly created macro in TINA's Digital interactive mode. To do this, place two High-Low digital switches from the Switches toolbar, one for each of the A,B inputs, and two logic indicators from

the Meters toolbar. Now select the DIG interactive mode with the ⏷

button or from the Interactive menu and press the 🟢 button. The

logic levels of the nodes will appear, Red for High. Blue for Low.

The logic indicators will also show the logic level of the outputs in a

Red square ![red] for High, and empty square ![empty] for Low.

5.5.3 Changing the pin arrangement of a VHDL macro

To change the pin arrangement, you should add a special header to your VHDL macro.

The easiest way to do this is to open the automatically generated macro and edit its header.

For example the header in the previous example is

```
—————————————————
— TINA VHDL Macro Description Begin
— entity_name:e_Half_add_entity;
— arch_name:a_Half_add_arch;
— ports:A,B;S,C;
— TINA VHDL Macro Description End
—————————————————
```

The pin arrangement is determined by the:

```
ports:A,B;S,C;
```

line, the ports before the first semicolon (;) are placed on the left while the rest are placed on the right side of the macro box.

For example, if you change the ports line to

```
ports:A,B,S;C;
```

and add the whole changed header to the original VHDL file (which had no header) we get the following file (you can also load it from the EXAMPLES\VHDL\half_adder_VHDL.vhd.)

```
—————————————————
— TINA VHDL Macro Description Begin
— entity_name:e_Half_add_entity;
— arch_name:a_Half_add_arch;
— ports:A,B,S;C;
— TINA VHDL Macro Description End
```

```
——————————————
LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
USE tina.primitives.all;
——————————————
— entity section
——————————————
ENTITY e_Half_add_entity IS PORT
  A : IN std_logic;
  S : OUT std_logic;
  C : OUT std_logic;
  B : IN std_logic );
END e_Half_add_entity;
——————————————
— architecture section
```
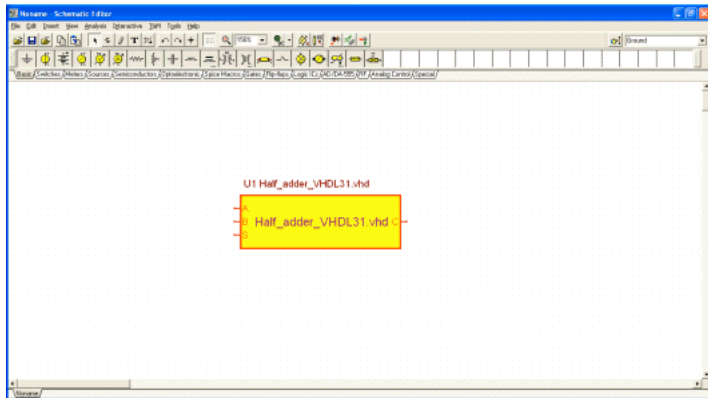


```
——————————————
ARCHITECTURE a_Half_add_arch of e_Half_add_entity
BEGIN
S <= (A xor B);
C<= (A and B);
END a_Half_add_arch;
```

Converting this into a new macro called Half_adder_VHDL.TSM
and then inserting it again we will see the revised pinout version.

MAKING YOUR OWN SCHEMATIC SYMBOLS AND FOOTPRINTS

6.1 Schematic Symbol Editor

Using *TINA*'s Schematic Symbol Editor, you can create new schematic symbols so that you can add your own circuit components to *TINA*.

To create new symbols, you place lines, arcs, rectangles, and arbitrary characters with any fonts, specifying line-width, color, and area color fills. After drawing the symbol, you add and define connections to it.

To start the Schematic Symbol Editor, use the Start or Apps screens of Windows 8 or the TINA group of the Start menu of Windows 7, Vista or XP.

To get acquainted with some features of the editor, read in the list of existing symbols. (Alternatively you may jump to paragraph "**Now let's create a new symbol"** below.)

Select *File|Open,* the little down arrow [▾] button on the right side of the Open button and select the Main TINA folder from the list. The folders and files in the main TINA folder will appear. Locate and then double click the *devices.ddb* file. On the right hand side of the

Editor Window, the list of current schematic symbols will appear.
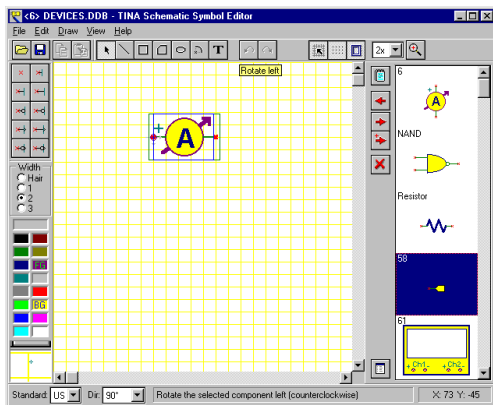
The first symbol on the list (an Ammeter) will appear in the editor window. Try the **Dir:** control at the bottom of the screen. Using this control, you can provide different shapes for symbols at each rotational orientation by designing each of them individually. Now click on the NAND symbol at the right side of the screen and press the ◀ button. The NAND gate symbol will appear in the editor window. Try the **Standard:** control to see the US and European versions of the part shape. You can design symbol versions for each standard, if necessary. If the symbols are identical in the two standards, you need create only one version.

Now let's create a new symbol for the full-adder circuit that was used above in our example of creating a half adder macro.

If this is your first own symbol press the File|New command to create a new library. If you already have your own symbol library open it with the Open command. First clear the editor window by pressing the ▣ New Device button on the left side of the Search field above the list of symbols on the right side of the screen. Now draw a rectangle as the body of the component. Press the ▣ button then click on any point in the drawing area, hold the mouse button, and move the mouse until the rectangle is properly sized.

YOUR OWN SCHEMATIC SYMBOLS
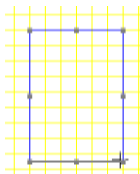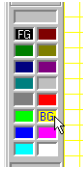
Fill the rectangle with a color by right-clicking on the palette at the lower left corner of the window. Note that a left-click will change the foreground color (FG), in our case the border of the rectangle.

Now add the terminals. Select the desired terminal type from the Terminal-Toolbar in the upper left corner of the window and move the cursor into the rectangle you just drew. Position it using the mouse or pressing the *[+]* or *[-]* key for rotation and click to locate the terminal. Be sure the small red x, indicating the pin end, is outside the body. Continue this process until every terminal is positioned.

Now add the terminals. Select the desired terminal type from the Terminal-Toolbar in the upper left corner of the window and move the cursor into the rectangle you just drew. Position it using the mouse or pressing the *[+]* or *[-]* key for rotation and click to locate the terminal. Be sure the small red x, indicating the pin end, is outside the body. Continue this process until every terminal is positioned.
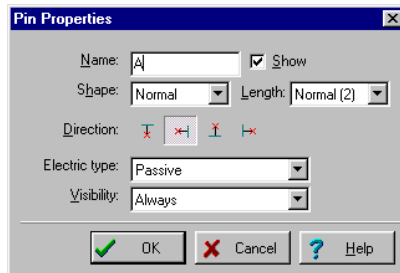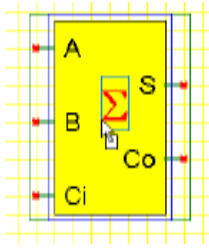
After you have positioned all the terminals, you can establish their properties by double clicking on each of them.

You should assign terminal names as shown in the picture below.

Next, add a large summation sign. Click on the 🔲 -Text-Editor button on the Toolbar, enter an S in the window, and select a font. To get the special Greek summation sign, select the Symbol-Font.
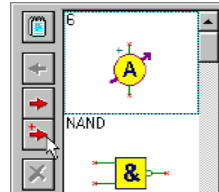
Press the 🔲 Device Properties button, set the Name of the symbol to Full Adder, and press OK.

Finally, copy the new symbol into the symbol library with the 🔲 button (it appears now at the end of the list), and use the File|Save or File|Save As… command to save the new or extended .ddb Library file.

Note, that if you create a new schematic symbol which must be placed in a new or existing library, it is strongly recommended to save it in a custom shape library (.ddb file) in your private catalog folder. To locate your **Private catalog folder** click on the little down arrow 🔲 button next to the Save button in the Save As... dialog.

Under Windows Vista, 7 and 8 or on a network you cannot save anything in the Main TINA folder, unless you have Administrator rights.

Another possibility to save your symbol(s) in the **Shared catalog folder**, which you can select from the same list, using the 🔲 button where the Main TINA folder and Private catalog folders are listed. The Shared catalog folder can be specified at Installation and located in an area which all TINA users on a computer or network can access.

If you have a new schematic symbol, it can be assigned to a new component (macro) using the New Macro Wizard, as it is described

DesignSoft

in chapters 5.1, 5.2 and 5.5. - or using the Library Manager as it is described in chapter 5.3.

6.2   IC Wizard in the Schematic Symbol Editor

When you need to create the shape of an IC with a lot of pins, the IC Wizard can assist you. The IC Wizard can be activated from the Draw menu by selecting the IC Wizard command. The following dialog box appears:
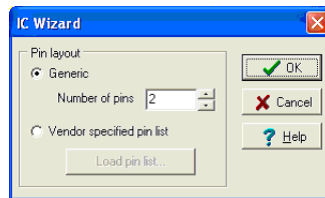
The wizard offers two options.

- **Generic** If you select this option, the Wizard creates a rectangular-shaped IC with a DIP-style pin layout. The total number of pins must be specified. For example, if you enter 14 pins in this field, you get the following pin layout:

DesignSoft
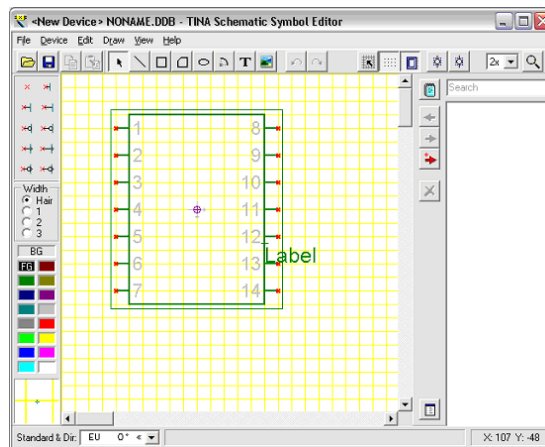
- **Vendor specified pin list** In this case the Wizard creates a shape based on a file where each line defines a terminal as **Pin number, Name, Electric type** separated by commas:

For example:.

> 1,RA2,INPUT
>
> 2,RA3,INPUT
>
> 3,RA4/T0CKI,INPUT
>
> 4,MCLR,INPUT
>
> 5,VSS,POWER
>
> etc.

The electrical type can be INPUT, OUTPUT, INOUT, BUFFER and POWER. For example, if you read in the PIC16F84A.CSV file from TINA's EXAMPLES\PCB folder, the Wizard generates the next IC:
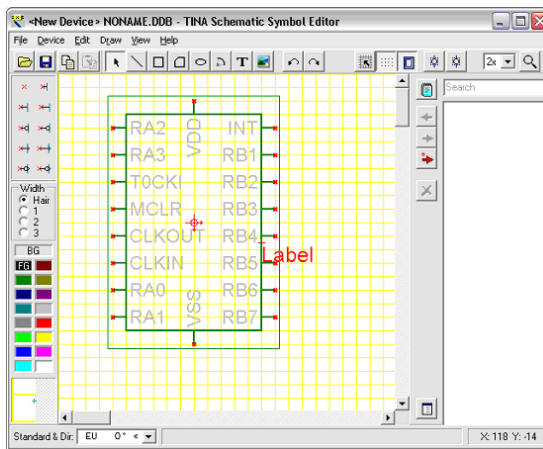
When the Wizard is finished, the shape can be further edited with the tools described above.

6.3   Footprint Editor

Using the Footprint Editor, you can create new footprint symbols that you can add to the footprint library. You can start the footprint editor from the Tools menu of TINA's PCB Designer by selecting the Footprint Editor command.

If you want to create a new footprint, you can build it by placing various primitive drawing elements and symbols, including lines, rectangles, arcs, text and pads. We'll recreate a simple resistor footprint already included in the system.

First clear the editor window by selecting the New Footprint command from the Footprint menu. Then set the position of the origin by double clicking on the cross symbol with the little arrows. Enter 1300, 1000 in the X and Y fields respectively. Check the Use Relative Coordinates checkbox, and press OK.
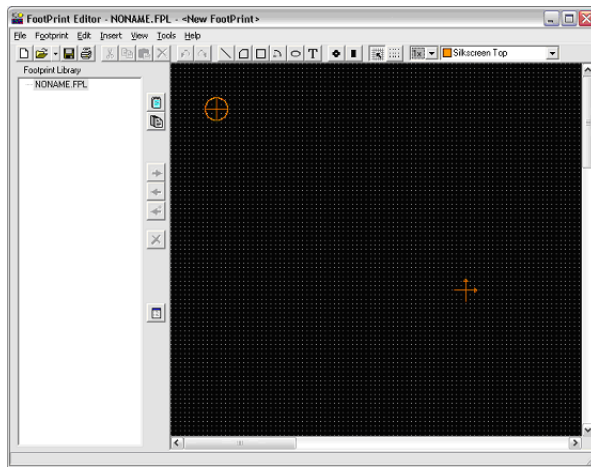
Now select the rectangle symbol on the toolbar and draw a rectangle around the origin. To do this, click on one corner, hold down the mouse left button, and drag the cursor to the opposite corner. Release the mouse button. If you create a footprint, you should be very careful with the dimensions. You must define the exact dimensions according to the manufacturer's data sheet, especially of the pads: otherwise the parts cannot fit on the board. To set the shape precisely, it is better to use coordinates rather than drawing with the mouse.

To set the size of our rectangle using coordinates, move the mouse over one of its edges and when the cursor changes into a hand symbol, double click at one edge of the rectangle. The Rectangle Property dialog will appear.

YOUR OWN SCHEMATIC SYMBOLS

DesignSoft
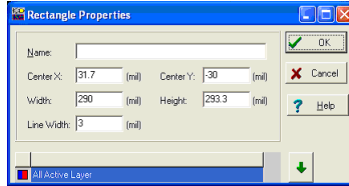
Now enter 0, 0 in the CenterX and CenterY fields; 840, 300 into Width and Height; and 5 into the Line Width fields.

In the Rectangle Property dialog of the shape you can change the layer settings, too. By default, a rectangle shape resides on the Silkscreen Top and Assembly Drawing Top layers.

Pressing the down arrow invokes the layer configuration editor. The layers can be turned on/off by double-clicking on the grey square next to the layer name. In our example, the default layer configuration is good, so don't change it. Close the property editor by pressing OK.
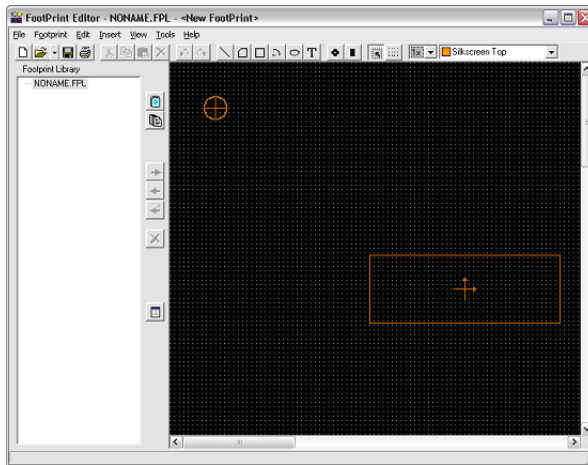
Now add 2 lines to our footprint. Select the line symbol and draw 2 horizontal lines next to the rectangle on both sides. Double click on the lines and modify the parameters as follows:

Line1: -460, 0, -420, 0, 5 (Point1 X, Point1 Y, Point2 X, Point2 Y and Line width)

Line2: 420, 0, 460, 0, 5 (Point1 X, Point1 Y, Point2 X, Point2 Y and Line width)

Finally, add two through hole pads to the footprint symbol. Select the pad symbol from the toolbar. Move the pad next to Line1. Now activate the property editor of the pad, by moving the mouse over it and double-clicking when the mouse changes into a hand symbol. Enter - 500, 0 in the Center X and Center Y fields. The Drill parameter is 37. Now click on the down arrow. By default the pad resides on the Top, Bottom, Power, Ground, Solder Mask Top, Solder Mask Bottom, Drill Drawing and Drill Tape layers.

The default layer configuration could have been changed in a way similar to what we've seen in the rectangle example. Though the default layer configuration is good, we have to change the dimensions of the pad. Double click on the size field and enter 58 in the Diameter field on the Top, Bottom, Solder Mask Top and Solder Mask Bottom layers, enter 78 on the Power and Ground layers and 37 on the Drill Drawing and Drill Tape layers. It's important to enter the package pin number into the name field.
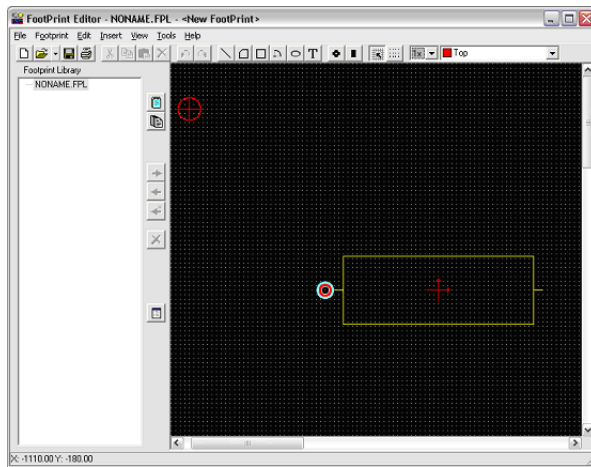
Now pick up the next pad and move to Line2. We have only one parameter to change, Center X, which should be 500.

DesignSoft
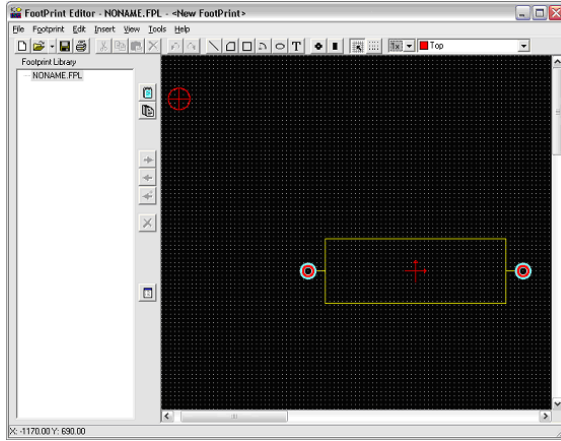
The footprint symbol is ready to save into a library. Open the package.fpl file, select the resistor group (or define a new group) and press the Add footprint button.
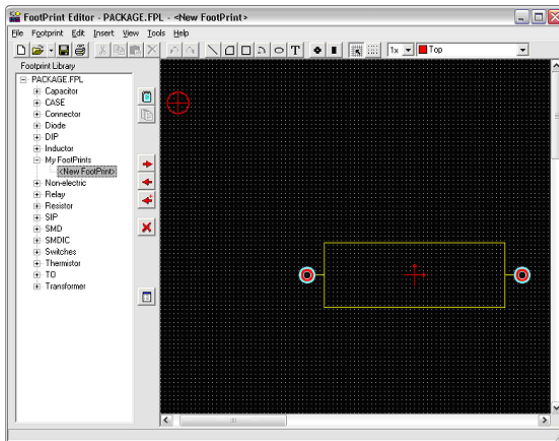
6.4 IC Wizard in the Footprint Editor

If you want to create the footprint of a more complex IC, e.g., an IC with a complex pin configuration, the IC Wizard can assist you. The IC Wizard can be activated from the Insert menu.

The wizard presents several properties of the IC which you can set.

YOUR OWN SCHEMATIC SYMBOLS

In the **Technology group,** you can set the mounting mode and the package type of the IC. The mounting mode can be through hole or surface mounted. Depending on the mounting mode the following packages are available: DIP (Dual in line package), PGA (Pin grid array package), SPGA (Staggered pin grid array package), SOP (Small outline package), LCC (Leaded chip carrier package), QFP (Quad flat package), BGA (Ball grid array package), SBGA (Staggered ball grid array package), SIP (Single in line package) and ZIP (Zigzag in line package) respectively.

In the **Package dimension** group, the dimensions (length, width, 3D height) of the package can be set. Depending on the selected package, the 4th parameter is either notch, corner cutoff, or ignored.

The **Pad dimension** defines the shape and dimensions (length, width) of the pad. If the mounting mode is through hole, the shape of the drilled pad can be round, square or octagon. Moreover, the shape and dimensions of the drill diameter can be defined. However, if the mounting mode is surface mounted, the shape of the pad can be circular, rectangular or rounded corner and the appropriate dimensions can be also set.

In the **Pad position,** the number of pins and the distances between them can be set according to the package type.

Finally, in the **Pad numbering** group, the type and direction for pad numbering can be entered, depending on the package type.

An example:

Technology: Through hole

Package type: DIP

Package dimension/Length: 400

Package dimension/Width:270

Pad dimension/Shape: Round

Pad dimension/Drill hole: 20

Pad dimension/Diameter 40

Pad position/Number of horz. pins: 14

Pad position/Between pins: 50

Pad position/Between rows: 160

YOUR OWN SCHEMATIC SYMBOLS

DesignSoft
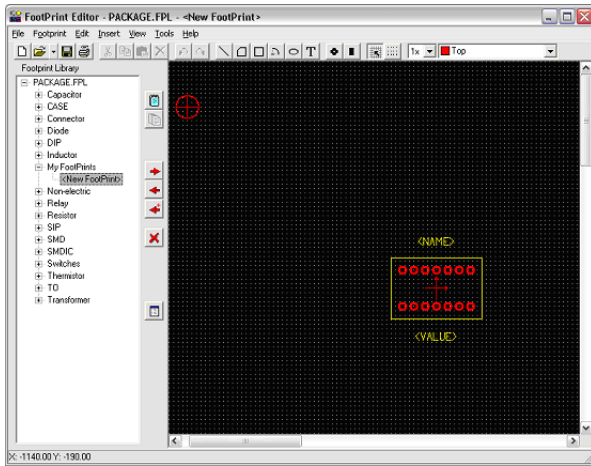
When the wizard is finished, the footprint can be further edited and saved in the library.

USING THE PARAMETER EXTRACTOR

Using *TINA*'s Parameter Extractor you can create component models that more closely represent actual real world devices by converting measurement or catalog data into model parameters.
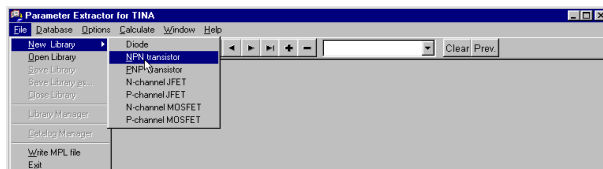
Use the Windows Start menu to locate the *TINA* folder. Start the Parameter Extractor by double-clicking its icon. To create a new transistor, which can be added to *TINA*'s transistor catalog later, select *File|New Library|NPN Transistor.*
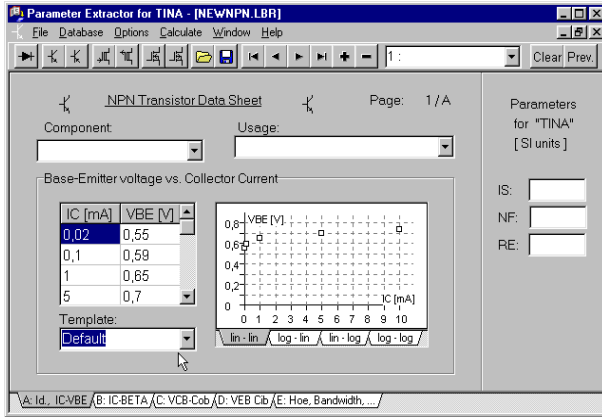
The dialog allows you to enter data from measurements, from manufacturers' catalog data, or from *TINA*'s default values (use the Template-ComboBox for this).

Click on each tab at the bottom of the screen and fill in all the transistor parameters. Select the default values or enter your own.
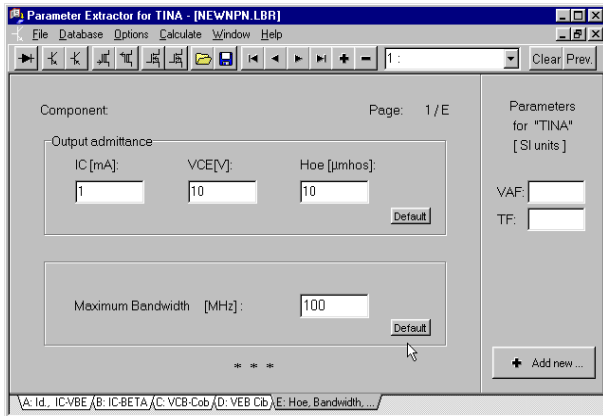
NOTE:

Be sure to fill in all data, since missing data may lead to incorrect results.

PARAMETER EXTRACTOR

Next select *Calculate|Current component*. To check how well *TINA*'s transistor model matches the input data, you can walk through the tabs to see the calculated graphs and numeric values for every parameter.

Finally, let's insert the new transistor into the source file for *TINA*'s transistor catalog by selecting *File|Catalog Manager*. To be able to use the new catalog, you must recompile the modified source files and link them together into the CLCC.CAT catalog file.

Locate and open a component-file compatible with your component (e.g., if adding a bipolar transistor, choose a bipolar catalog, bipol_x.crc). Click on the Browse-Button and select the file from the File Open Dialog. All component-files delivered with *TINA* are placed in the CLCC subdirectory of the *TINA* directory (by default *C:\Program Files\Designsoft\TINA*).
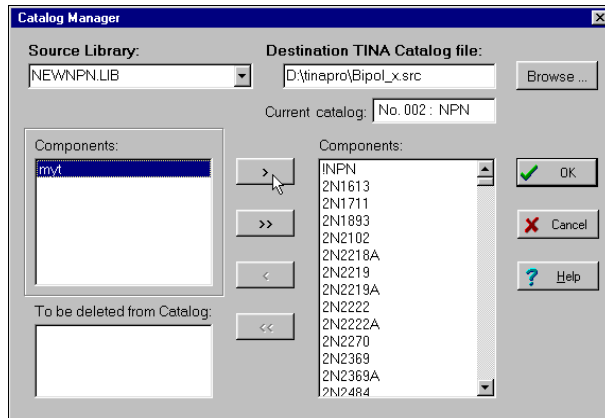
Move your component into the library by selecting it, clicking on the ➤ button and then the OK button.

After Pressing OK, TINA will prompt you and ask if you want to recompile the catalog source files and create a new updated catalog. If you answer "Yes," TINA will create the new catalog and you can use it after restarting TINA. You can also recompile the catalog using the "Compile TINA Catalog" command in the File menu. This may be necessary if a previous attempt at compiling failed, e.g., due to insufficient hard disk space.

In a similar fashion you can calculate magnetic core parameters. You should enter the upper (A) and lower (B) curve of the hysteresis and the geometric parameters of the core. Run an example with the default parameters (load Default from the Template listbox) to see typical values.

ADVANCED TOPICS

8.1   Introduction

In the previous chapters, we have introduced TINA's main features and how they can be accessed. Our demonstration, however, is far from complete, for TINA contains many other useful and advanced features for designing, testing and teaching electronics and electronic circuits. These include S-parameter models, network analysis, details of Fourier series and Fourier spectrum analysis, symbolic analysis, post-processing of analysis results, creation of phasor diagrams, Nyquist diagrams, the built-in interpreter, multiparameter optimization, creation of multilayer PCBs and other topics.

The detailed description of these topics is not included in the printed Quick Start manual; they are published only in electronic form. They can be found on the TINA install CD and on the web at www.tina.com in the Documentation section.

DesignSoft

8.2   Table of Contents of Advanced Topics

- 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
- Design Tool
- Optimization
- Design Tool vs. Optimization
- Fourier Series and Fourier Spectrum
- Interpreter manual
- TINA PCB Design manual

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

DesignSoft

INDEX