



PC BOARD LAYOUT TOOLS

386
PLUS

U
S
E
R'
S
G
U
I
D
E

PC Board Layout

Tools 386+

User's Guide



Electronic Design Automation Tools

PC Board Layout

Tools 386+

User's Guide

Copyright © 1993 OrCAD, Inc. All rights reserved.

No part of this publication may be reproduced, translated into another language, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise without the prior written consent of OrCAD, Inc.

Every precaution has been taken in the preparation of this publication. OrCAD assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

OrCAD[®] is a registered trademark of OrCAD, Inc.

IBM[®] is a registered trademark of International Business Machines Corporation.

Microsoft[®] is a registered trademark of Microsoft Corporation.

Windows[™] is a trademark of Microsoft Corporation.

Phar Lap[®] is a registered trademark of Phar Lap software, Inc.

386 | DOS-Extender[™] is a trademark of Phar Lap software, Inc.

Portions of this document copyright Phar Lap software, Inc.

All other brand and product names mentioned herein are used for identification purposes only, and are trademarks or registered trademarks of their respective holders.

OrCAD[®] 

C O N T E N T S

Chapter 1: Welcome to OrCAD PC Board Layout Tools 386+	
Minimum configuration.....	3
Virtual memory recommendations.....	3
Configuring virtual memory.....	3
Finding the information you need.....	4
Project-oriented design environment.....	4
Beyond the basics.....	5
Working in the Design Environment.....	5
Tools.....	6
Editors.....	7
Processors.....	7
Librarians.....	7
Reporters.....	8
Transfers.....	8
Learning PC Board Layout Tools 386+.....	9
Chapter 2: Installing PC Board Layout Tools 386+.....	9
Chapter 3: Transferring from schematic to layout.....	9
Chapter 4: Introducing Edit Layout.....	9
Chapter 5: Creating board modules.....	9
Chapter 6: Placing the TUTOR board.....	9
Chapter 7: Routing the TUTOR board.....	10
Chapter 8: Autorouting the TUTOR board.....	10
Chapter 9: Printing and plotting the TUTOR board.....	10
Appendix A: Phar Lap technical information.....	10
Chapter 2: Installing PC Board Layout Tools 386+	
Upgrading from OrCAD/PCB II.....	11
Installing PC Board Layout Tools 386+.....	12
Before you install.....	12
Installing the software.....	12

Chapter 3: Transferring from schematic to layout

The files you need	15
Before you begin.....	16
Keys.....	16
<Enter>.....	16
<Ctrl>.....	16
Other keys.....	16
Mouse basics.....	16
Keyboard input.....	17
Operating system command prompt.....	17
Commands	17
Filenames.....	17
Designs.....	18
Running the ESP design environment.....	18
Changing to the TUTOR design.....	19
Changing the startup design.....	20
Configuring DRAFT and Schematic Design Tools.....	22
Configuring DRAFT.....	22
Configuring Schematic Design Tools.....	23
Displaying the schematic.....	24
Annotating the schematic.....	25
Viewing key fields.....	27
Creating a stuff file.....	28
Updating field contents.....	30
Checking design integrity.....	31
Configuring Cleanup Schematic.....	31
Configuring Cross Reference Parts	32
Configuring Check Electrical Rules.....	33
Running Check Design Integrity.....	34

Chapter 3: Transferring from schematic to layout (continued)

Creating the netlist.....	35
About INET.....	35
Configuring INET.....	35
About ILINK.....	36
Configuring ILINK.....	36
About IFORM.....	37
Configuring IFORM.....	37
Running To Layout.....	38
Viewing the netlist.....	38
Summary	39

Chapter 4: Introducing Edit Layout

Configuring PC Board Layout Tools.....	42
Configuring Edit Layout.....	46
Running Edit Layout.....	47
Moving around the screen	47
Edit Layout command basics	48
Displaying the main menu	48
Commands	48
The command interface.....	49
Menus	49
Command lines.....	49
Dialog boxes	50
Dialog box items	50
Button.....	50
List box.....	51
Droplist box.....	52
Check box	52
Radio button	53
Entry box.....	53
Scroll buttons	53
How command names are shown in this guide.....	55
Returning to the main menu level.....	55

Chapter 4: Introducing Edit Layout (continued)

Setting up Edit Layout conditions.....	56
The SET command	56
Layer.....	57
Current Settings	59
Selecting a layer.....	60
LAYER	60
/ OTHER	61
+ LAYER and - LAYER.....	61
* LAYER.....	61
Changing your view of the layout.....	62
Zoom	62
Setting a zoom scale	63
Selecting a zoom window.....	63
WINDOW ZOOM	63
Pointer movement resolution.....	64
Using bookmarks.....	65
Creating a bookmark.....	65
Jumping to a bookmark.....	66
Deleting a bookmark.....	67
Changing the origin	68
Setting grid options.....	69
Setting a grid size	69
Setting a grid divisor	70
Disabling the snap grid.....	70
Changing the grid color.....	70
Saving and backing up the board file.....	71
Updating the file.....	71
Saving configurations.....	71
Writing to another filename.....	72
Copying a file.....	73
Renaming a file.....	73
Deleting a file	74
Suspend to System.....	75

Chapter 4: Introducing Edit Layout (continued)

Macros.....	76
Creating the first macro	76
Creating the second macro	79
Running the macros.....	80
Saving all macros to a file	80
Exporting a macro to a file.....	83
Deleting a macro from the disk.....	84
Deleting a macro from Edit Layout.....	84
Deleting all macros from Edit Layout.....	84
Loading a macro from disk.....	85
Running a defined macro	85
Summary	85

Chapter 5: Creating board modules

About modules.....	87
About the library editor.....	88
Selecting the library editor.....	88
Working with module libraries	89
Copying the DEMO library	89
Copying and getting a module.....	89
Updating the library file.....	91
Renaming a module	92
Displaying module information.....	93
INQUIRE	93
VERBOSE INQUIRE.....	94
EDIT.....	95

Chapter 5: Creating board modules (continued)

Editing a module.....	96
Moving a module.....	96
Rotating a module to a specific angle.....	97
Rotating a module in preset steps.....	98
Mirroring a module along the X axis.....	99
Mirroring a module along the Y axis.....	100
Mirroring a module along both the X and Y axis.....	100
Flipping a module to the other side of the board.....	101
Moving a single module object.....	102
Moving selected objects within a group.....	103
Moving an off-grid object.....	104
Moving an off-grid object back on grid.....	104
Deleting and undeleting module objects.....	105
Deleting objects on any layer.....	105
Deleting objects on a specific layer.....	105
UNDELETE.....	106
SELECTIVE.....	106
Permanently deleting deleted objects.....	107
Exporting and importing modules.....	108
Exporting.....	108
Importing.....	109
Deleting exported modules.....	110
Creating a module.....	111
Starting a new module.....	111
Setting preferences.....	112
Changing the view of the display.....	113
Drawing methods.....	113
Drawing the outline with 90 degree corners.....	113
Adding arcs.....	114
Positioning the arcs.....	116
Deleting the 90° corners.....	117
Drawing the outline with arc corners.....	119
Adding holes.....	121

Chapter 5: Creating board modules (continued)	
Placing the pad array	122
Selecting a new pad symbol	124
Designing the pad array.....	125
Positioning the placeholders.....	127
Saving the module.....	128
Leaving the library editor.....	128
Summary	128
Chapter 6: Placing the TUTOR board	
About layout placement.....	129
Loading the Edit Layout template.....	130
Setting options.....	130
Drawing the board outline.....	131
Loading the netlist.....	132
Defining a netlist block and selecting TUTOR386.NET	132
About module placement.....	135
Placement aids.....	135
Ratsnest.....	135
Force vector.....	135
Displaying the ratsnest.....	136
Turning off the ratsnest.....	136
Displaying force vectors.....	137
Turning off force vectors.....	137
Placing the modules	138
Coordinate placement	138
Dynamic placement.....	141
Editing module placement	142
Hiding module text	142
Rotating module text.....	143
Flipping a module to the other side of the board	144

Chapter 6: Placing the TUTOR board (continued)

Placing other board objects.....	145
Layer marker.....	145
Board identification.....	146
Drawing the board name outline.....	146
Placing the board name.....	147
Placing a fill zone.....	148
Creating the fill zone.....	148
Understanding zone/pad isolation.....	149
Assigning a net to the fill zone.....	150
Viewing thermal relief.....	151
Placing alignment targets.....	152
Saving your work.....	153
Summary.....	153

Chapter 7: Routing the TUTOR board

About manual routing.....	155
Getting started.....	155
Zooming in on the routing area.....	155
Highlighting a net.....	156
Displaying a ratsnest for a single pad.....	157
Creating a new copper tool.....	158
Routing the board.....	159
Setting conditions.....	159
Routing the first track.....	159
Routing with vias.....	160
Routing with arc segments.....	161
Performing a DRC check.....	164
Running a block DRC check.....	164
Identifying DRC violations.....	165
Using INQUIRE.....	165
Using JUMP.....	166
Viewing the violated areas.....	166

Chapter 7: Routing the TUTOR board (continued)

Editing the routed board.....	168
Correcting DRC errors.....	168
Drawing a new track.....	168
Deleting the offending track.....	169
Running another DRC check.....	169
Deleting a stub.....	171
Deleting and undeleting a track.....	171
Changing a track path.....	172
Changing track width.....	173
Saving your work.....	174
Summary.....	174

Chapter 8: Autorouting the TUTOR board

About autorouting.....	175
Preparing for autorouting.....	175
Placing an autoroute zone.....	176
Locking an existing route.....	177
Setting routing conditions for a net.....	178
Specifying a copper tool.....	178
Excluding vias.....	178
Setting autorouter options.....	179
Setting an autoroute method.....	179
Setting a sweep routing direction.....	179
Autorouting the board.....	180
Autorouting a section of the board.....	180
Autorouting the whole board.....	182
Setting a sweep window.....	182
Begin autorouting.....	183
Via reduction.....	184
Additional processing.....	185
Erasing all routes.....	185

Chapter 8: Autorouting the TUTOR board (continued)

Finishing the layout	186
Moving reference designators.....	186
Placing an assembly outline.....	187
Placing dimensions.....	188
Placing the first dimension.....	188
Placing the second dimension.....	190
Saving your work	191
Summary	191

Chapter 9: Printing and plotting the TUTOR board

About printing and plotting	193
Getting started.....	193
Printing	194
Configuring printer options.....	195
Configuring pages.....	196
Building the TOP COPPER LAYER page.....	196
Building the BOTTOM COPPER LAYER page.....	198
Building the ASSEMBLY DRAWING page.....	199
Printing pages.....	200
Printing all pages.....	200
Printing selected pages.....	201
Saving a printer setup	203
Loading a printer setup.....	204
Plotting	205
Gerber (274-X).....	205
HP-GL/2.....	206
Postscript.....	206

Appendix A: Phar Lap technical information

About the CFIG386 utility..... 207

 How to run CFIG386..... 207

 Error messages..... 209

386 | DOS-Extender command line switches..... 211

 Conventional memory switches..... 212

 Systems Call Data Buffer Switches..... 213

 Mixed mode program switches..... 214

 Stack allocation switches..... 215

 Extended memory switches..... 216

 Weitek 1167 switch..... 217

 Interrupt relocation switches..... 218

 Interrupt mapping switches..... 219

 Paging disable switch 220

 Compaq built-in memory switch 221

 VDISK compatibility switch 221

 80386 step B0 switch..... 222

 EMS simulator switch 223

 Address line 20 switch..... 224

 PC and PC/XT detection switch 225

386 | VMM command line switches..... 226

 Virtual memory driver switches..... 227

 Swap file location switch..... 228

 Page replacement policy switches..... 228

 Swap file growing policy switches..... 230

Index..... 235



Welcome to OrCAD PC Board Layout Tools 386+

You now have **OrCAD PC Board Layout Tools 386+**, a powerful yet straightforward PC board layout tool set with the capability of an engineering workstation. **PC Board Layout Tools 386+** is designed with today's high density, multi-package board engineering environment in mind.

PC Board Layout Tools 386+ features extensive autorouting capability of up to 16 simultaneous layers, and gives the PC board designer sophisticated manual routing tools. Also featured is an extensive collection of surface mount and thru-hole modules libraries, and all the utilities needed to release a board to manufacturing.

You can import an existing board file into **PC Board Layout Tools 386+**, then use the board as a module. This capability reduces your design time by providing access to reusable board designs.

Complex pad array layouts are easy to create in the **PC Board Layout Tools 386+** module library editor. You use pad array generators to automatically lay out chip carriers, staggered pin connectors, pin grid arrays, and polar coordinate modules.

PC Board Layout Tools 386+ uses extended memory, up to the maximum available. It also uses virtual memory, so you can work with very large boards by having program and board file information temporarily stored on the hard disk that would normally be stored in memory. Board design and module complexity are limited only by the amount of available PC memory and virtual memory disk space.

PC Board Layout Tools 386+ is specifically written for personal computers using an 80386 or better microprocessor. The added capabilities of these microprocessors—chiefly larger memory capacity and greater speed—are leveraged by OrCAD's new 32-bit database and software.

PC Board Layout Tools 386+ is a completely new product, and supports most popular graphics boards, printers, and plotters. All of the **PC Board Layout Tools 386+** programs, libraries, and drivers work with and support the 32-bit database.

Minimum configuration

To use **PC Board Layout Tools 386+**, you must have an IBM PC or compatible with:

- ❖ An 80386 or faster microprocessor. A floating point coprocessor (80387, 80487) is highly recommended.
- ❖ A VGA or higher resolution display.
- ❖ Four megabytes of free RAM after DOS and all device drivers are loaded. Full autorouting capability requires eight megabytes of RAM, with sixteen megabytes recommended for increased performance.
- ❖ A hard disk with ten megabytes or more of free storage space for product installation. Additional contiguous hard disk space is needed for effective use of virtual memory. See **Virtual memory recommendations**.

Virtual memory recommendations

PC Board Layout Tools 386+ uses virtual memory to swap portions of program code and file data to your computer hard disk when all system RAM is used. The data swapped to disk is stored in a temporary file and read back when the data is needed. The following are recommendations for achieving optimum virtual memory performance:

- ❖ Use a disk defragmenting utility to maintain your free hard disk space as a single, contiguous area.
- ❖ Provide adequate disk space for swapping. A good rule of thumb is to multiply your system memory by 1.5 and have at least that much contiguous disk space available for the swap file.

Virtual memory for **PC Board Layout Tools 386+** is dynamically allocated. This means that when system RAM is filled, the swap file increases and decreases in size, according to program demands.

Configuring virtual memory

You use **CFIG386.EXE** to perform custom virtual memory configurations for **PC Board Layout Tools 386+**. For most systems, the default configuration is acceptable.

Refer to *Appendix A: Phar Lap technical information* for virtual memory configuration options.

Finding the information you need

These guides accompany **PC Board Layout Tools 386+**:

- ❖ *PC Board Layout Tools 386+ User's Guide*
- ❖ *PC Board Layout Tools 386+ Reference Guide*
- ❖ *Installation & Technical Support User's Guide*
- ❖ *ESP Design Environment User's Guide*
- ❖ *Stony Brook M2EDIT Text Editor User's Guide*
- ❖ *Fast Track*, a quick reference for PCB II users upgrading to PCB 386+

Project-oriented design environment

PC Board Layout Tools 386+ is one part of a fully integrated *Electronic Design Automation* (EDA) system. The design environment means you can focus on what's important: the design. Designs are organized on a project-by-project basis, with all the design files—schematics, netlists, parts lists, simulation results, and board layouts—stored together.

The *ESP Design Environment User's Guide* introduces the graphical environment under which **PC Board Layout Tools 386+** and the other OrCAD tool sets operate. In this environment, OrCAD tools and tool sets, such as **PC Board Layout Tools 386+**, are accessed via buttons. There are four OrCAD tool sets. They are:

- ❖ Schematic Design Tools
- ❖ Digital Simulation Tools
- ❖ Programmable Logic Design Tools
- ❖ PC Board Layout Tools

Buttons to access all four OrCAD tool sets display on the **Design Environment** screen, even if you only have one tool installed on your computer.

Beyond the basics

Once you have mastered the basics, refer to the *PC Board Layout Tools 386+ Reference Guide* for information that will help you plan and create your design. The reference guide explains how to tailor configurations to match your personal requirements, and provides detailed information about the commands and concepts of **PC Board Layout Tools 386+**. The *PC Board Layout Tools 386+ Reference Guide* is designed to be a continuing source of instruction and reference as you use **PC Board Layout Tools 386+**.

Working in the Design Environment

PC Board Layout Tools 386+ is one part of a fully integrated electronic design automation environment. The graphical design environment:

- ❖ Runs the tools within a tool set. The tools that make up **PC Board Layout Tools 386+** are listed in the next section.
- ❖ Moves between tool sets without switching directories or copying files.
- ❖ Configures tools. Each tool can be configured and the configuration stored. This eliminates the need to enter command line switches every time a tool is used.
- ❖ Organizes designs by project. All files associated with a design—schematics, netlists, reports, PLD source code, simulation results, and board layouts—are stored in one location. This location is actually a directory on your computer's hard disk. Each design has its own directory containing all of the files described above.

Tools

The tools in a tool set are organized by function:

- ❖ Editors
- ❖ Processors
- ❖ Librarians
- ❖ Reporters
- ❖ Transfers

Figure 1-1 shows how these tools are organized on the PC Board Layout Tools screen.

These functions are described briefly on the pages that follow. The explanations assume you are already familiar with common electronic design terms and concepts. If you are just learning about PC board design, some terms we use to describe the tools may not be familiar to you. Don't worry: basic, essential concepts and skills are thoroughly covered in chapters 3 through 9 of this guide. Advanced concepts are fully explained in the *PC Board Layout Tools 386+ Reference Guide*.

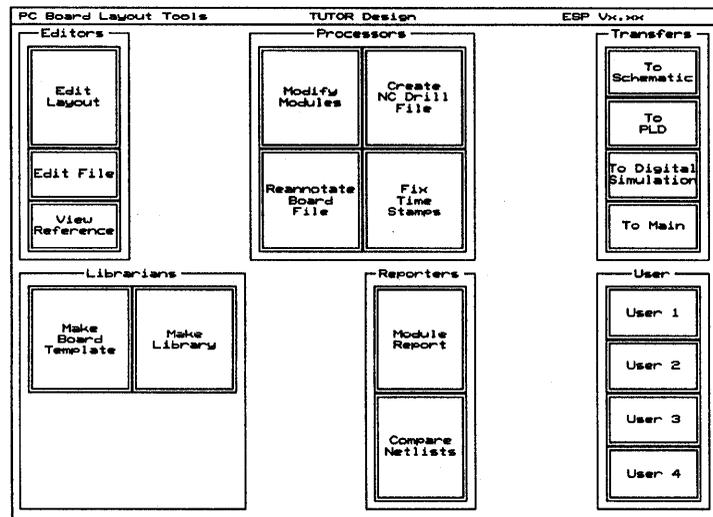


Figure 1-1. The PC Board Layout Tools screen.

- Editors** Editors modify or create design files. **PC Board Layout Tools 386+** contains three editors:
- ❖ **Edit Layout** routes the layout.
 - ❖ **Edit File** is used to create and edit text files.
 - ❖ **View Reference** is used to review reference material supplied with **PC Board Layout Tools 386+** using a text editor.
- Processors** Processors are tools that subject a design file to a specific process. **PC Board Layout Tools 386+** includes four processors:
- ❖ **Modify Modules** modifies pad shape, pad size, and drill size for modules either in a layout or in a module library.
 - ❖ **Create NC Drill File** creates a report of drilling information, including location and drill size, for a board file.
 - ❖ **Reannotate Board File** reannotates your board file so the modules are numbered sequentially. You can reannotate specific modules, or all modules in a board file.
 - ❖ **Fix Time Stamps** compares the netlist file with the board file and assigns the time stamps in the netlist to the modules in the board file, based upon reference designators.
- Librarians**
- ❖ **Make Board Template** creates a custom PC board template file from a board file.
 - ❖ **Make Library** creates a module library file from a board file.

Reporters Reporters are tools that produce human-readable reports, but do not modify design data in any way. **Reporters** include:

- ❖ **Module Report** reports module locations in a PC board layout file.
- ❖ **Compare Netlists** compares an EDIF netlist with a board file and reports differences between the two.

Transfers Transfer tools run utilities that create the files necessary for other tool sets to continue the design process. During the design process, the design database created in one tool set (such as **PC Board Layout Tools 386+**) is not useable by other tool sets (such as **Schematic Design Tools**) for much of the design process. This is because the design is not complete. The transfer is how the design database is updated so that the other tools may have access. The **Transfers** tools take care of intermediate steps so that you don't have to. The four transfer tools in **PC Board Layout Tools 386+** are:

- ❖ **To Schematic**
- ❖ **To PLD**
- ❖ **To Digital Simulation**
- ❖ **To Main**

For example, the **To Schematic** tool does this intermediate step:

- ❖ **Runs Back Annotate**, updating the reference designators in the schematic so they match the new reference designators in the board file.

Learning PC Board Layout Tools 386+

The remainder of the *PC Board Layout Tools 386+ User's Guide* shows how to use the tool to design a PC board by guiding you through various exercises. To create a board design, you use **Edit Layout**.

Each of the remaining chapters builds on the skills and concepts from the previous chapter.

The summary below describes the design concepts and skills you learn in each chapter.

Chapter 2: Installing PC Board Layout Tools 386+

In this chapter you learn how to install **PC Board Layout Tools 386+**.

Chapter 3: Transferring from schematic to layout

In this chapter you learn how to transfer a design from **Schematic Design Tools** to **PC Board Layout Tools 386+**. This chapter describes how to edit the TUTOR386 schematic so it contains all the information required by **PC Board Layout Tools 386+**. You learn how to configure **Schematic Design Tools** to produce a netlist, which **Edit Layout** uses to produce a board layout.

Chapter 4: Introducing Edit Layout

This chapter introduces **Edit Layout**. You learn how to change default configuration settings, change view and display options, and define and save macros.

Chapter 5: Creating board modules

Although **PC Board Layout Tools 386+** provides extensive libraries, you may occasionally need a module not in any library. This chapter describes how to edit modules from within **Edit Layout**. In this chapter you learn how to create a new module, save the new module in a library, and export and import modules.

Chapter 6: Placing the TUTOR board

In this chapter you create a PC board layout by loading the netlist you produced in *Chapter 3: Transferring from schematic to layout*. You learn the basic procedures required for placing modules. You also learn how to edit module placement.

**Chapter 7: Routing the
TUTOR board**

In this chapter you route the TUTOR board. You also learn how to edit routed tracks.

**Chapter 8: Autorouting
the TUTOR board**

In this chapter you use the autorouter to automatically route the TUTOR board. In this chapter you learn how to customize autorouting methods by setting routing options.

**Chapter 9: Printing and
plotting the TUTOR
board**

In this chapter you produce a print and a plot of the routed board on a printer and on a plotting device.

**Appendix A: Phar Lap
technical information**

This appendix describes command line options for the Phar Lap DOS memory extender.



Installing PC Board Layout Tools 386+

Upgrading from OrCAD/PCB II

The installation program makes it easy to upgrade your OrCAD/PCB II board files and custom modules to **PC Board Layout Tools 386+**. You can choose to have all your OrCAD/PCB II board files and custom modules automatically converted during the installation process, or you can choose to convert the files manually.

△ *NOTE: OrCAD/PCB II will not be accessible from the ESP design environment after PC Board Layout Tools 386+ is installed. See Fast Track for instructions on running OrCAD/PCB II from outside the ESP design environment.*



To make it easy to find the instructions that apply only to systems that are being upgraded from OrCAD/PCB II, they are marked with the symbol shown at left.

If you are installing **PC Board Layout Tools 386+** on a system that doesn't have any OrCAD software installed, you can ignore the instructions marked by this symbol.

▲ *CAUTION: If you are installing PC Board Layout Tools 386+ on a system that already contains OrCAD software, you must install new drivers. Be sure to follow the instructions given in the Installing the software section very carefully.*

Installing PC Board Layout Tools 386+

To install PC Board Layout Tools 386+ you must use the *new* INSTALL program to copy PC Board Layout Tools 386+ to your system. This is explained in *Installing the software*.

Before you install



Follow these steps if you are upgrading from OrCAD/PCB II to PC Board Layout Tools 386+:

1. Back up all of your custom PC board modules. This is especially important if you have modified any OrCAD-provided modules.
2. Back up all of your OrCAD/PCB II board designs.

Installing the software

Use the INSTALL program provided with PC Board Layout Tools 386+ to install the software. *Do not use an older version of INSTALL already on your hard disk.*



NOTE: You may have problems installing the software if you have several TSR's (terminate and stay resident programs) loaded, such as an anti-virus program. If your computer is connected to a network, running SHARE.EXE may cause installation problems. Also, problems may occur if you run the installation program as a DOS application under Microsoft® Windows™.

1. Insert the disk labeled "Install" into your computer's floppy disk drive.
2. At the DOS prompt, enter the name of the drive the disk is in. For example, if you placed the installation disk in drive A, type **A:** and press <Enter>.
3. Type **INSTALL** and press <Enter>.

INSTALL prompts you to enter the information it needs to install the software on your system. Answer the questions and insert disks into your computer's disk drive as requested.



If you already have any of OrCAD's Release IV or 386+ software installed on your system, be aware of these two important details:

- ❖ You *must* install the new INSTALL program on your hard disk, and you *must* install new display drivers. When INSTALL asks you if you want to install any of these, *be sure to always answer YES*.
- ❖ The INSTALL program asks whether or not to update the ORCADESP.DAT files in your design directories. If your system uses the directory structure recommended by OrCAD, answer YES to this question so that INSTALL will do this for you. If your directory structure does not match the directory structure recommended by OrCAD or if you answer NO to this question you will need to manually update these files using the MERGEDAT program. MERGEDAT is described in technical note #45: *Updating ORCADESP.DAT files with MERGEDAT*.

INSTALL asks you if you want all your OrCAD/PCB II board files and custom modules automatically converted for PC Board Layout Tools 386+.

If you select automatic conversion, the installation program uses the OrCAD environment variables listed in your AUTOEXEC.BAT file to locate and convert all OrCAD/PCB II board files and custom modules. If you select manual conversion, refer to *Fast Track* for information on file conversion commands.

You may need to manually convert files if you have OrCAD\PCB II board files or custom modules in directories that are not in any directory path defined by the OrCAD environment variables,

When the installation is complete, the DOS prompt displays.

4. Reboot your system so that any changes INSTALL made to your AUTOEXEC.BAT file can take effect.



Transferring from schematic to layout

This chapter describes the processes used to transfer a design from **Schematic Design Tools** to **PC Board Layout Tools 386+**. The schematic you will work with is TUTOR386.SCH, which is the schematic for the board layout you load in the next chapter. In this chapter, you:

- ❖ Configure the ESP design environment and **Schematic Design Tools**
- ❖ Configure **To Layout** to produce netlist TUTOR386.NET, which is used to create a circuit board in chapter 6.
- ❖ Transfer from **Schematic Design Tools** to **PC Board Layout Tools 386+**

The files you need

These files are installed on your computer when you install **PC Board Layout Tools 386+**, and are used in this chapter:

- ❖ TUTOR386.SCH – This is the schematic you edit to create the netlist TUTOR386.NET.
- ❖ TUTOR386.LIB – This is the schematic parts library for TUTOR386.SCH.
- ❖ TUTORORC.NET – This is an OrCAD-supplied netlist you can use if you do not want to create TUTOR386.NET.

If you do not want to perform the steps describing how to create a netlist, you can complete this chapter up to the **Configuring DRAFT and Schematic Design Tools** section, then skip to the next chapter.

When you begin *Chapter 6: Placing the TUTOR board*, you substitute TUTORORC.NET, the OrCAD-supplied netlist, for TUTOR386.NET.

Before you begin

Before you begin the exercises in this part of the user's guide, take a minute to review the conventions used in this guide and learn some operating system basics.



Keys

PC Board Layout Tools 386+ is designed to operate on a wide variety of 386 and 486 computer systems. Since many computers label their keyboard keys differently, OrCAD has adopted standards to name two of the most widely-used keys.

<Enter>

Whenever you see <Enter>, it means to press the <Enter> key on your keyboard. On your keyboard, the <Enter> key may be labeled Return.

Throughout the user's guide, you are instructed to enter text. For example, the instructions may read "Enter the filename." This means to type the name of the file and press <Enter>. If you are instructed to "Type the following characters," you should type the specified characters *without* pressing the <Enter> key.

<Ctrl>

Whenever you see <Ctrl> it means to hold down the <Ctrl> key and press another key. For example, if the instructions say "press <Ctrl><A>," you should hold down the <Ctrl> key and press the <A> key.

Other keys

Alphanumeric, function keys, and other special keys are shown in angle brackets.

Mouse basics



- ❖ Clicking the left mouse button is the same as pressing the <Enter> key. In this user's guide, when you are instructed to "press <Enter>," you can use either the keyboard or the mouse, whichever you prefer.
- ❖ Clicking the right mouse button is the same as pressing the <Esc> key. In this user's guide, when you are instructed to "press <Esc>," you can use either the keyboard or the mouse, whichever you prefer.

Keyboard input

Characters that you enter are shown in bold monospace font, such as “enter **tutor.bd1**”. This text can also be enclosed in a box:

```
tutor.bd1
```

In the example above, you enter only the characters shown in bold.

Operating system command prompt

In this user’s guide, the operating system command prompt is shown as:

```
C:>
```

Commands

Commands are shown in **bold type**. Main menu commands are shown in uppercase letters. Other commands are shown as they appear on the menu. When you are asked to select a command, usually both the main menu command and other command are specified.

Filenames

Filenames can be from one to eight characters long. A filename may also have a period and an extension consisting of up to three characters. You can use either uppercase or lowercase letters when entering a filename or extension, but the operating system converts all the letters to uppercase.

Filenames and extensions usually contain only letters and numbers. However, you can use additional characters supported by the operating system. For compatibility with OrCAD’s environment, use only letters (A–Z and a–z), numbers (0–9), underscores (_), number signs (#), and “at” signs (@).

Most OrCAD software works with any characters your operating system supports. Some applications used in conjunction with OrCAD software—including SPICE programs, some PCB layout programs, and some text editors—support a more limited character set. You should keep any such limitations in mind as you design and avoid using characters that are allowed by one piece of software but not another.

Designs In the OrCAD design environment, all files pertaining to a design are stored in one directory on your disk. Putting different designs in different directories organizes your files, much as you would organize a file cabinet.

Running the ESP design environment

To run an OrCAD tool, you must first display the ESP design environment main screen. To do this, enter the command shown in bold:

```
C:> ORCAD
```

In a moment, the design environment screen displays (figure 3-1).

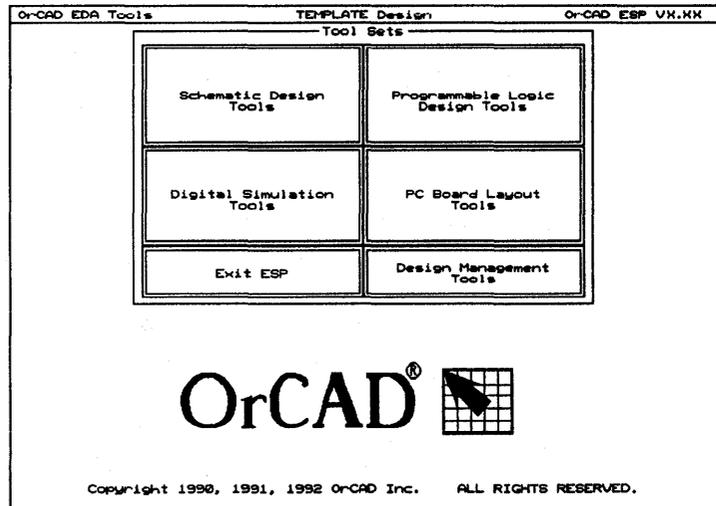


Figure 3-1. The ESP design environment main screen.

Changing to the TUTOR design

Before you work with any of the tools accessed from the main screen, you need to change to the TUTOR design. Remember, a design is a directory in which all the files related to a project are stored.

1. Place the pointer on **Design Management Tools** and click the left mouse button. The menu shown at right displays.

Design Management Tools

Execute
Local Configuration
Assign Hot Key
Configure ESP
Help

2. Select **Execute**. The dialog box shown in figure 3-2 displays.

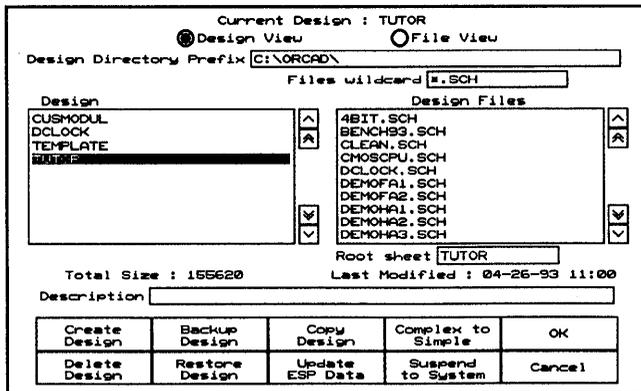


Figure 3-2. Design Management Tools screen, Design View.

3. Place the pointer on the design named TUTOR and click the left mouse button. This selects the TUTOR design.
4. Select **OK** to return to the main screen. Notice that the heading in the upper center of the screen has changed to **TUTOR Design**.

△ **NOTE:** See the ESP Design Environment User's Guide for instructions on how to use *Design Management Tools*.

Changing the startup design

The ESP design environment is configured to the TEMPLATE design each time you run OrCAD tools. Since you will be working in the TUTOR design throughout this tutorial, you need to change the startup design to TUTOR. Follow these steps:

1. **Select Design Management Tools.**
The menu shown at right displays.
2. **Select Configure ESP.**
The Configure ESP screen displays (figure 3-3).

Design Management Tools

- Execute
- Local Configuration
- Assign Hot Key
- Configure ESP
- Help

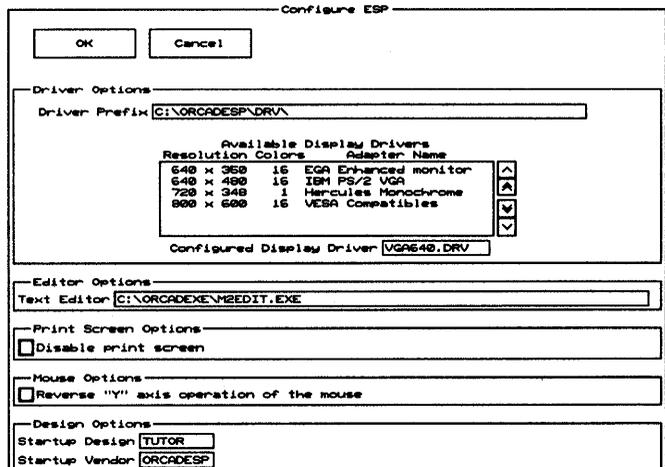


Figure 3-3. The Configure ESP screen.

3. Enter **TUTOR** in the Startup Design entry box.
4. Select **OK** to save the configuration changes. The main screen displays.



NOTE: See the ESP Design Environment User's Guide for detailed instructions on how to configure ESP.

△ **NOTE:** *If you do not want to create the netlist TUTOR386.NET, perform the two steps listed below and skip to Chapter 4: Introducing Edit Layout. When you load the netlist in Chapter 6: Placing the TUTOR board, select TUTORORC.NET as the netlist filename.*

*If you want to create the netlist, skip the two steps listed below and proceed to **Configuring DRAFT and Schematic Design Tools.***

1. Select **PC Board Layout Tools**. The menu shown at right displays.
2. Select **Execute**. The **PC Board Layout Tools** screen displays.

PC Board Layout Tools

Execute
Local Configuration
Assign Hot Key
Configure ESP
Help

Configuring DRAFT and Schematic Design Tools

Before you can edit TUTOR386.SCH you need to configure DRAFT to specify the source schematic. You also need to configure Schematic Design Tools to select TUTOR386.LIB as the configured library.

Configuring DRAFT

1. Select Schematic Design Tools, then select Execute. The Schematic Design Tools screen displays (figure 3-4).

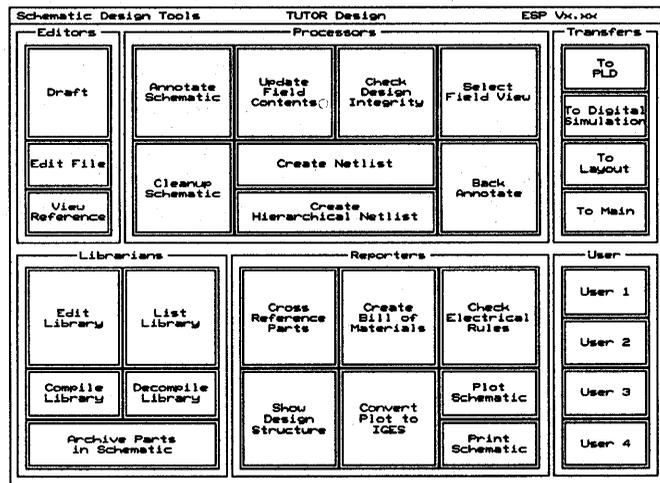


Figure 3-4. The Schematic Design Tools screen.

2. Select Draft, then select Local Configuration from the menu and select Configure DRAFT. The Configure DRAFT screen displays (figure 3-5).

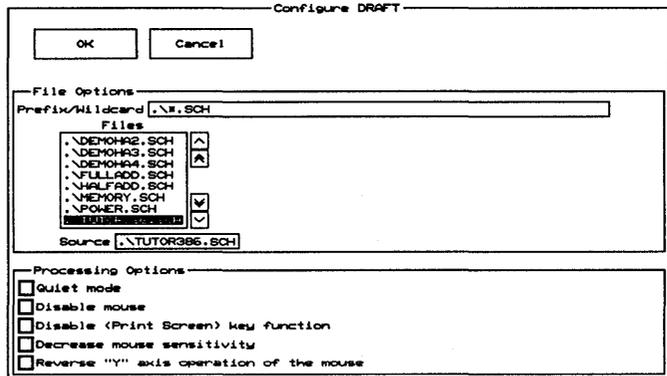


Figure 3-5. The Configure DRAFT screen.

3. Select `.\TUTOR386.SCH` from the Files list box.
4. Select OK. The Schematic Design Tools screen displays.

Configuring Schematic Design Tools

1. Select Draft, then select Configure Schematic Tools from the menu. The Configure Schematic Design Tools screen displays (figure 3-6).

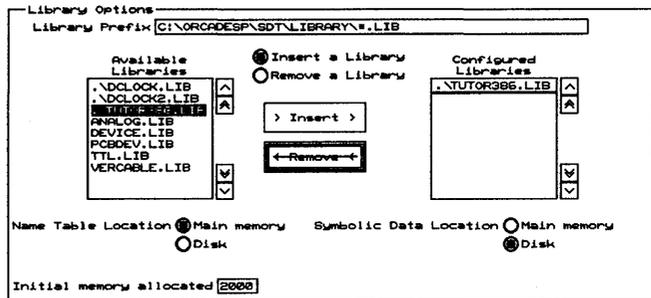


Figure 3-6. Partial view of the Configure Schematic Design Tools screen, Library Options section.

2. Scroll to the Library Options section.
3. Select `.\TUTOR386.LIB` from the Available Libraries list box, then select Insert. The Configured Libraries list box displays `.\TUTOR386.LIB`.
4. Select OK. The Schematic Design Tools screen displays.

Displaying the schematic

1. Select **Draft**, then select **Execute** from the menu. In a moment, the schematic displays (figure 3-7).

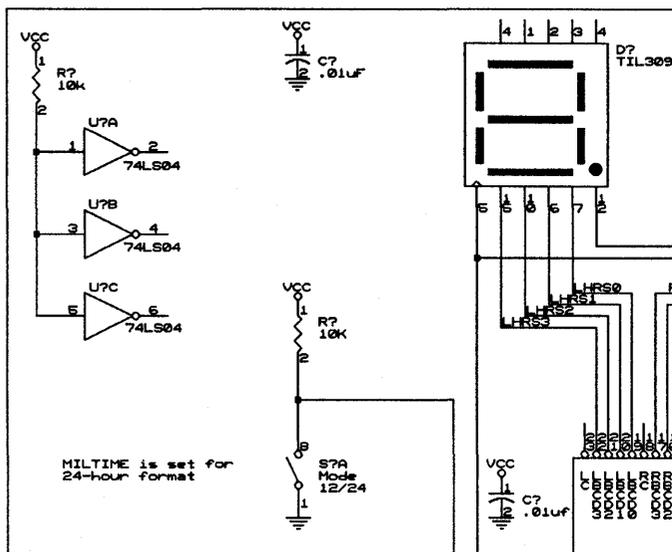


Figure 3-7. Partial view of the TUTOR386 schematic.

The reference designators are not annotated and no module values are assigned. These tasks need to be done before you can create a netlist.

You annotate reference designators by selecting **Annotate Schematic** from **Schematic Design Tools**. See **Annotating the schematic**.

You assign module values to the schematic by creating a stuff file, then you select **Update Field Contents** to select the stuff file and insert the module values into **Part Field 8** on the schematic. See **Creating a stuff file** and **Updating field contents**.

2. Select **QUIT Abandon Edits** to return to the **Schematic Design Tools** screen.

Annotating the schematic

Annotate Schematic scans a design and automatically updates the reference designators of all parts in the design.

Annotate Schematic updates reference designators in the order the parts are placed in the design. You may assign all parts a new reference designator, including any manually edited parts, when annotating the design. To selectively change reference designators and leave others unmodified, use **Draft's EDIT Reference Name** command.

See *Chapter 6: Annotate Schematic* in the *Schematic Design Tools Reference Guide* for more information about **Annotate Schematic**.

Follow these steps to annotate the reference designators on TUTOR386.SCH:

1. Select **Annotate Schematic**, then select **Local Configuration** from the menu and select **Configure ANNOTATE**. The **Configure Annotate Schematic** screen displays (figure 3-8).

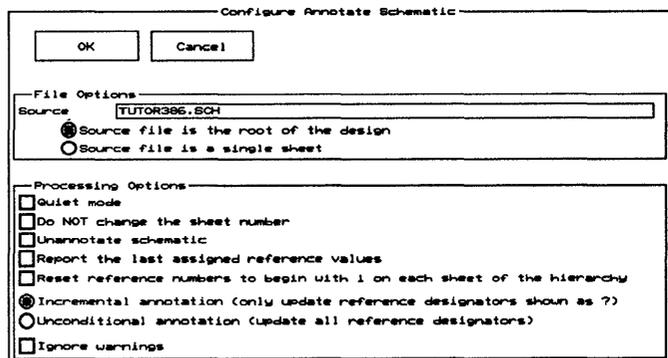


Figure 3-8. The *Configure Annotate Schematic* screen.

2. Enter **TUTOR386.SCH** in the **Source** entry box. Do not change any of the default settings.
3. Select **OK**.

4. Select **Annotate Schematic**, then select **Execute** from the menu. Processing status displays in a window at the bottom of the screen. The status window closes when **Annotate Schematic** is complete.
5. Select **Draft**, then select **Execute**. The reference designators are annotated for TUTOR386.SCH, as shown in figure 3-9.

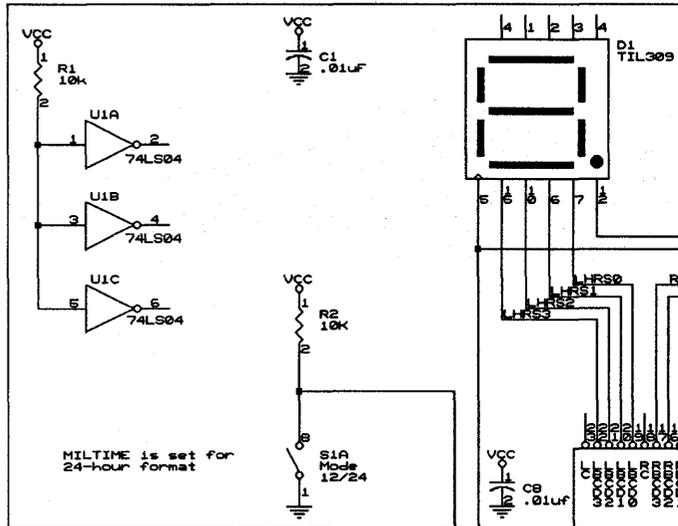


Figure 3-9. A partial view of the annotated schematic.

6. Select **QUIT Abandon Edits** to return to the **Schematic Design Tools** screen.

Viewing key fields

Key fields tell **Update Field Contents** where to look for data. A stuff file, or update file, tells **Update Field Contents** what module value to insert in the schematic when it finds a match.

Update Field Contents uses the preconfigured key field entry in the **Update Field Contents** entry box of the **Configure Schematic Design Tools** screen to construct a text string called a *match string*.

Key fields for **Schematic Design Tools** are already configured to use the schematic **Part Value** field and **8th Part Field**. The **Part Value** field is used as the match string that determines what module values go in the **8th Part Field** on the schematic.

Follow these steps to view the preset key fields for TUTOR386.SCH:

1. Select **Draft**, then select **Configure Schematic Tools**. The **Configure Schematic Design Tools** screen displays.
2. Scroll down to the **Key Fields** section.
3. A **V** displays in the **Combine for Field 8** entry box for **Update Field Contents**. The letter **V** represents the value in the schematic **Part Value** field, and specifies the **Part Value** field as the match string.
4. The number **8** displays in the **Module Value Combine** entry box for **Create Netlist**. This specifies the **8th Part Field** on the schematic as the field that **To Layout** checks for the module value when it creates a netlist.
5. Go to the top of the **Configure Schematic Design Tools** screen and select **Cancel**.

Creating a stuff file

Update Field Contents requires a stuff file. You create this text file using a text editor like M2EDIT, the text editor that comes with **Schematic Design Tools**. If you use a different text editor, use the comparable commands of your text editor.

△ **NOTE:** Be sure to save this file as text only. Any special formatting inserted by your text editor may cause **Update Field Contents** to fail.

Follow these steps to create the stuff file TUTOR386.STF:

1. Select **Edit File**, then select **Execute**. The **Edit File** screen displays (figure 3-10).

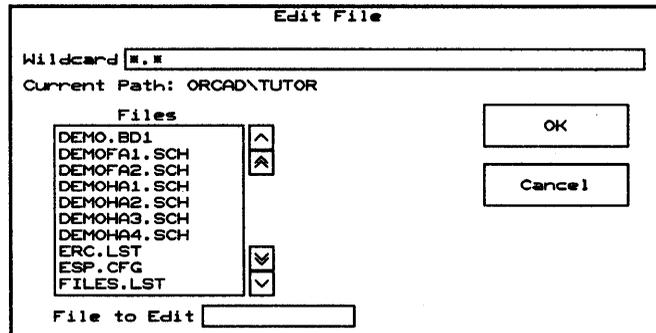


Figure 3-10. The **Edit File** screen.

2. Enter **TUTOR386.STF** in the **File to Edit** entry box, then select **OK**. The M2EDIT screen displays.

A stuff file is composed of a list of text strings which are delimited with single quotes. The strings are separated with any number of space, tab, or return characters. A string cannot contain a single quote.

Update Field Contents looks at the strings in the stuff file in pairs, so for readability and clarity each pair is placed on a separate line in the stuff file. The first string in each pair is the match string. The second string is what **Update Field Contents** places into the schematic part field when it finds a match.

3. Enter the text shown below, including single quotes and upper and lower case characters. Separate the pairs on each line with two tabs.

'10k'	'RC05'
'9.1k'	'RC05'
'.01uF'	'CK05'
'100uF'	'CK05'
'47uF'	'CK05'
'470uF'	'CK05'
'22uF'	'CK05'
'74LS04'	'14DIP300'
'TIL309'	'TIL309'
'22V10'	'24DIP600'
'LM7805'	'TO220'
'9V'	'BAT9V'
'Mode'	'8DIP300'
'Reset'	'PBTN2PIN'

4. Select **Output** to save the text file as TUTOR386.STF.
5. Select **Exit** to close M2EDIT. The **Edit File** screen displays.
6. Select **Cancel** to dismiss the **Edit File** screen and return to the **Schematic Design Tools** screen.

Updating field contents

You prepared the stuff file, now you are ready to update TUTOR386.SCH by inserting module values into the schematic 8th Part Field. Follow these steps:

1. Select **Update Field Contents**, then select **Local Configuration** and **Configure FLDSTUFF**. The **Configure Update Field Contents** screen displays (figure 3-11).

Figure 3-11. The **Configure Update Field Contents** screen.

2. Enter **TUTOR386.SCH** in the **Source** entry box.
3. Select **.\TUTOR386.STF** from the **Files** list box. The filename displays in the **Stuff File** entry box.
4. **Part Field 8** is already selected in the **Field to be updated** section of **Processing Options**. This specifies that the 8th Part Field of the schematic receives the module values recorded in the stuff file.
5. Select **Set the specified field to visible**. You can view the inserted module values on the schematic when this is selected.

6. Select **OK**. The **Schematic Design Tools** screen displays.
7. Select **Update Field Contents**, then select **Execute**. Program status displays in the status window at the bottom of the screen. When the process is complete, the status window closes.
8. Select **Draft**, then select **Execute** to load the updated schematic. Module values display for each schematic part. These values are inserted in the **8th Part Field**.
9. Select **QUIT Abandon Edits** to return to the **Schematic Design Tools** screen.

Checking design integrity

Before you create the netlist, you select **Check Design Integrity** to check for duplicate objects or overlapping wires in the schematic. **Check Design Integrity** also creates a report file, then scans the schematic for conformity to basic electrical rules, such as unused inputs and invalid connections. The three processors in **Check Design Integrity** are:

- ❖ **Cleanup Schematic**
- ❖ **Cross Reference Parts**
- ❖ **Check Electrical Rules**

Follow these steps to configure the three processors:

Configuring Cleanup Schematic

1. Select **Check Design Integrity** from the **Schematic Design Tools** screen, then select **Local Configuration**. The menu shown at right displays.
2. Select **Configure CLEANUP**. The **Configure Cleanup Schematic** screen displays (figure 3-12).

Configure CLEANUP
Configure CROSSREF
Configure ERC

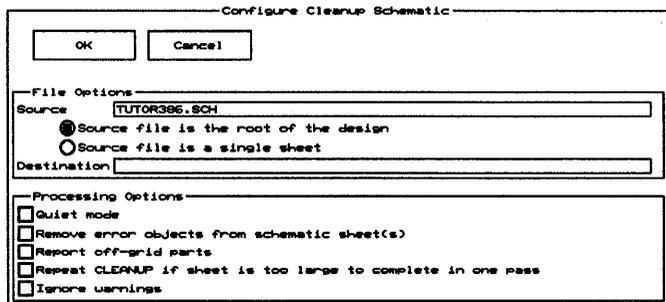


Figure 3-12. The Configure Cleanup Schematic screen.

3. Enter TUTOR386.SCH in the Source entry box, then select OK. The Schematic Design Tools screen displays.

Configuring Cross Reference Parts

1. Select Check Design Integrity from the Schematic Design Tools screen, then select Local Configuration and select Configure CROSSREF. The Configure Cross Reference Parts screen displays (figure 3-13).

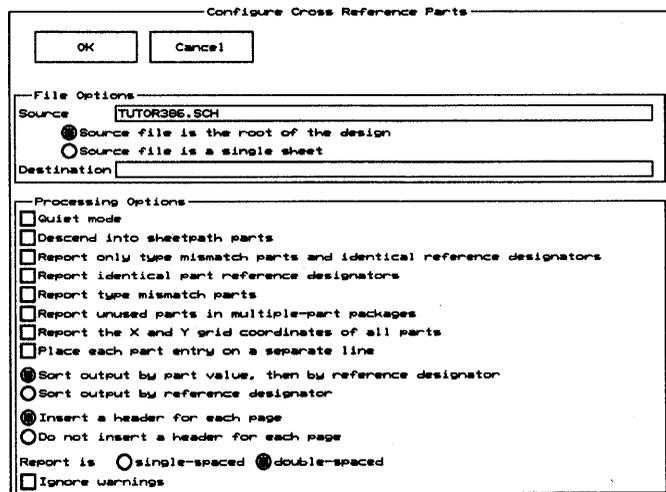


Figure 3-13. The Configure Cross Reference Parts screen.

2. Enter TUTOR386.SCH in the Source entry box.

3. Enter **TUTOR386.XRF** in the **Destination** entry box. This is the filename of the report generated by this processor. You can view this file using a text editor. If a filename is not specified in **Destination**, the report displays in the monitor box at the bottom of the screen and is recorded in the ESP design environment redirection file **#ESP_OUT.TXT**.
4. Select **OK**. The **Schematic Design Tools** screen displays.

Configuring Check Electrical Rules

1. Select **Check Design Integrity** from the **Schematic Design Tools** screen, then select **Local Configuration** and select **Configure ERC**. The **Configure Check Electrical Rules** screen displays (figure 3-14).

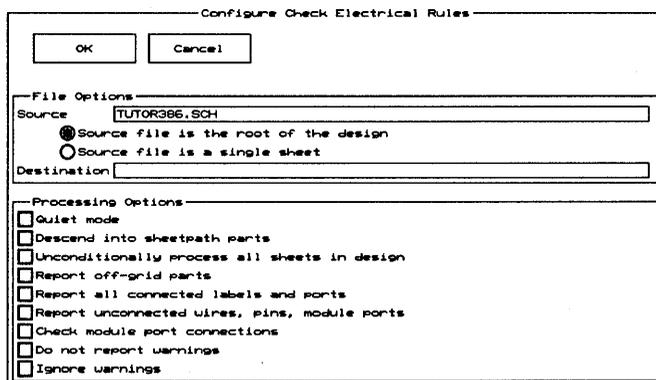


Figure 3-14. The **Configure Check Electrical Rules** screen.

2. Enter **TUTOR386.SCH** in the **Source** entry box, then select **OK**. The **Schematic Design Tools** screen displays.
3. Select **Ignore warnings**, then select **OK**. The **Schematic Design Tools** screen displays.



NOTE: You select **Ignore warnings** because **TUTOR386.SCH** has unconnected pins and **Check Electrical Rules** will not complete successfully unless this option is selected. The preferred solution is to place a no-connect object on each of the unconnected pins in **DRAFT**.

Running Check Design Integrity

1. Select **Check Design Integrity**, then select **Execute**. Processing status displays in the monitor box at the bottom of the screen. When processing is complete, the monitor box closes.
2. If errors or warnings are reported, refer to the appropriate chapters of the *Schematic Design Tools Reference Guide* for more information. The chapter for each of the processors is listed below:
 - ❖ **Cleanup Schematic**—Chapter 8
 - ❖ **Cross Reference Parts**—Chapter 23
 - ❖ **Check Electrical Rules**—Chapter 22

Creating the netlist

You use **To Layout in Schematic Design Tools** to run the processes needed to create the netlist and transfer your design to **PC Board Layout Tools 386+**.

To Layout runs three processes that update the connectivity database. These processes are **INET**, **ILINK**, and **IFORM**.

About INET

INET creates or updates the connectivity database for the design. See *Chapter 9: Creating a Netlist* in the *Schematic Design Tools Reference Guide* for more information about this processor.

Configuring INET

1. Select **To Layout**, then select **Local Configuration**. The menu at right displays. **INET**, **ILINK**, and **IFORM** are set to **on** by default.

Configure INET	
Configure ILINK	
Configure IFORM	
INET	on
ILINK	on
IFORM	on

2. Select **Configure INET**. The **Configure Incremental Netlist** screen displays (figure 3-15).

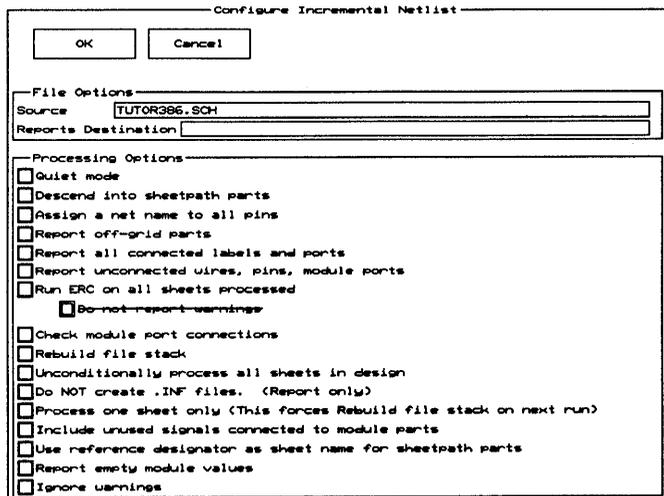


Figure 3-15. The *Configure Incremental Netlist* screen.

3. Enter **TUTOR386.SCH** in the **Source** entry box.
4. Select **OK**. The **Schematic Design Tools** screen displays.

About ILINK

ILINK creates the intermediate netlist structure and the linked connectivity database. This database contains information on connectivity, parts, fields, pin typing information, and layout directives.

See *Chapter 9: Creating a Netlist in the Schematic Design Tools Reference Guide* for more information about this processor.

Configuring ILINK

1. Select **To Layout**, then select **Local Configuration**. The menu at right displays.
2. Select **Configure ILINK**. The **Configure Netlist Linker** screen displays (figure 3-16).

Configure INET	
Configure ILINK	
Configure IFORM	
INET	on
ILINK	on
IFORM	on

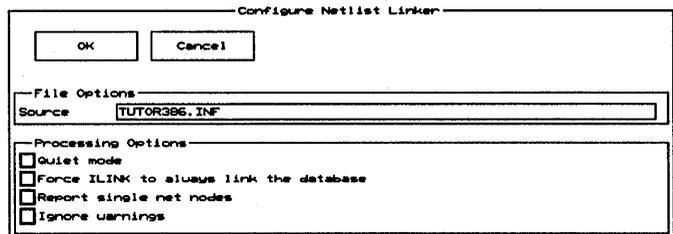


Figure 3-16. The *Configure Netlist Linker* screen.

3. Enter **TUTOR386.INF** in the **Source** entry box.
4. Select **OK**. The **Schematic Design Tools** screen displays.

About IFORM

IFORM uses the netlist format file and an intermediate netlist structure created by ILINK to create a netlist in the format you define. EDIF is the netlist format used in this tutorial.

Configuring IFORM

Follow these steps to configure IFORM so it produces a netlist in EDIF format:

1. Select To Layout, then select Local Configuration. The menu at right displays.
2. Select Configure IFORM. The Configure Netlist Format screen displays (figure 3-17).

Configure INET	
Configure ILINK	
Configure IFORM	
INET	on
ILINK	on
IFORM	on

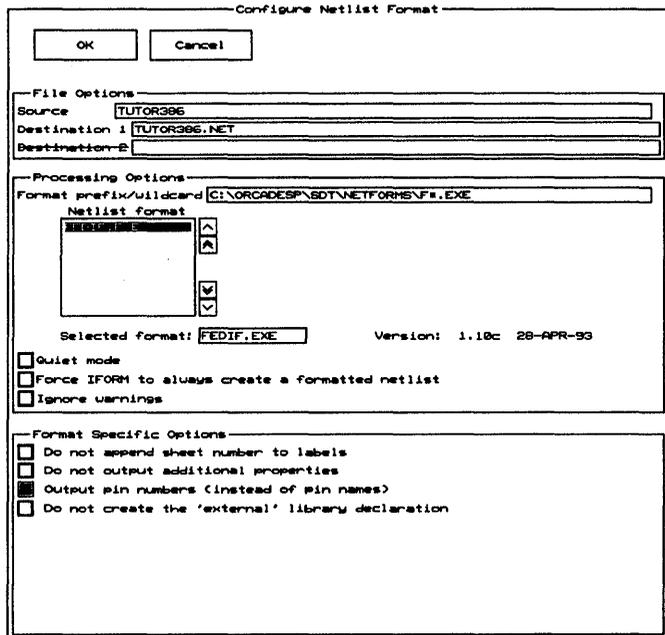


Figure 3-17. The Configure Netlist Format screen.

3. Enter **TUTOR386** in the **Source** entry box.
4. Enter **TUTOR386.NET** in the **Destination 1** entry box.
5. Select **FEDIF.EXE** in the **Netlist Format** list box. The filename displays in the **Selected Format** entry box.
6. Select **Output pin numbers (instead of pin names)** in the **Format Specific Options** section.
7. Select **OK**. The **Schematic Design Tools** screen displays.

Running To Layout

Now that you have configured all of the necessary **To Layout** processors, you are ready to run **To Layout** and create the **TUTOR386.NET** netlist.

1. Select **To Layout**.
2. Select **Execute** from the menu. **INET**, **ILINK**, and **IFORM** process sequentially. Processing status displays in the status window at the bottom of the screen. When **To Layout** is done, the netlist file is created and the **PC Board Layout Tools** screen displays.

Viewing the netlist

Follow these steps to use **M2EDIT** to display the netlist. If you have a different text editor configured, use the comparable commands of that text editor.

1. Select **Edit File** from the **PC Board Layout Tools** screen, then select **Execute**. The **Edit File** screen displays.
2. Select **.\TUTOR386.NET** from the **Files** list box. The filename displays in the **File to Edit** entry box.
3. Select **OK** to display **TUTOR386.NET**. The netlist looks like figure 3-18.

```

(edifVersion 2 0 0)
(edifLevel 0)
(keywordMap (keywordLevel 0))
(status
  (written
    (timeStamp 0 0 0 0 0 0)
    (program "IFORM.EXE")
    (comment "Original data from OrCAD/SDT schematic"))
  (comment "Digital clock schematic")
  (comment " April 23, 1993")
  (comment "")
  (comment "")
  (comment "")
  (comment "")
  (comment "")
  (comment "")
  (comment ""))
(external OrCAD_LIB
  (edifLevel 0)
  (technology
    (numberDefinition
      (scale 1 1 (unit distance))))
(cell &CAP
  (cellType generic)
  (comment "From OrCAD library TUTOR386.LIB")
  (view NetlistView
    (viewType netlist)
    (interface
      (port &1 (direction INOUT))
      (port &2 (direction INOUT))))))
(cell &R
  (cellType generic)
  (comment "From OrCAD library TUTOR386.LIB"))

```

Figure 3-18. Part of the TUTOR386.NET netlist.

4. Select **Exit** to close M2EDIT without making any changes to the netlist file. The **Edit File** screen displays.
5. Select **Cancel** to close the **Edit File** screen. The **PC Board Layout Tools** screen displays.

Summary

In this chapter you updated schematic reference designators using **Annotate Schematic**. You created a module value stuff file and inserted module values into schematic part fields using **Update Field Contents**. You also created a netlist using the processors in **To Layout**.

The next chapter introduces you to **Edit Layout**.



Introducing Edit Layout

PC Board Layout Tools 386+ uses the **Edit Layout** editor to create the board layout. As its name suggests, **Edit Layout** routes your layout both manually and automatically.

Edit Layout is designed to support the complete PC board layout *process* from netlist to high-resolution output. **Edit Layout** stores the information on the computer's disk as a data file.

Edit Layout saves the board file in the design in which you are working, or you can save it to another directory. The board file can have the design name and an extension of **.BD1**, or you may give it a different filename and extension.

In this chapter, you learn how to:

- ❖ Change default configuration settings
- ❖ Change view and display options
- ❖ Save, copy, and rename board files in **Edit Layout**
- ❖ Define and save macros

Configuring PC Board Layout Tools

You configure PC Board Layout Tools to define the working board file, current device drivers, available module libraries, file paths, and default file extensions.

After completing *Chapter 3: Transferring from schematic to layout*, the PC Board Layout Tools screen displays (figure 4-1).

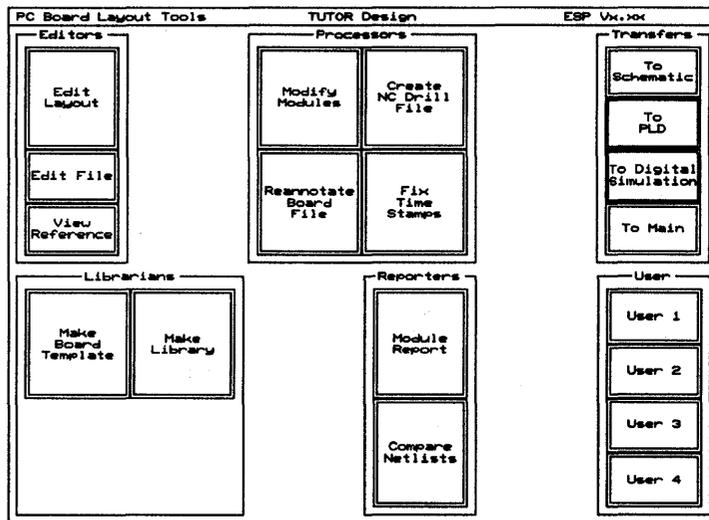


Figure 4-1. The PC Board Layout Tools screen.

Follow these steps to configure PC Board Layout Tools:

1. Select **Edit Layout**, then select **Configure Layout Tools**. The **Configure PC Board Layout** screen displays (figure 4-2).

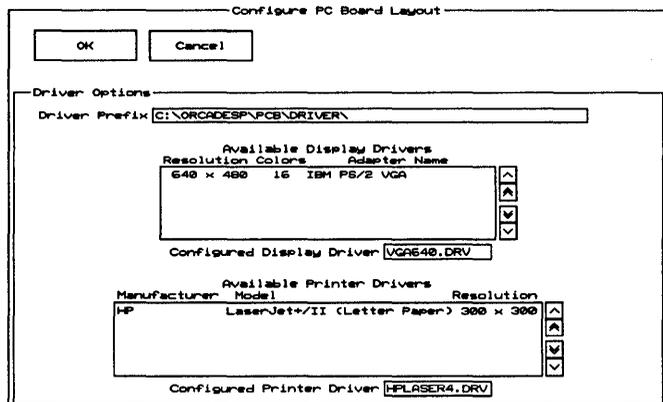


Figure 4-2. Top of the Configure PC Board Layout screen.

2. Select a display driver for Edit Layout from the **Available Display Drivers** list box. The driver filename displays in the **Configured Display Driver** entry box. If you want to use a custom driver that is not listed, enter its filename and extension in the **Configured Display Driver** entry box.
3. Select a printer driver for Edit Layout from the **Available Printer Drivers** list box. The driver filename displays in the **Configured Printer Driver** entry box. If you want to use a custom driver that is not listed, enter its filename and extension in the **Configured Printer Driver** entry box.

△ **NOTE:** You must select a printer driver if you want to print the TUTOR board in Chapter 9: Printing and plotting the TUTOR board.

4. Enter the path for your TUTOR design directory in the **Library Prefix** entry box. The module libraries you use in this tutorial are stored in TUTOR.

If you use the default directory structure, you enter **C:\ORCAD\TUTOR*.MLB** in the **Library Prefix** entry box.

If your TUTOR design directory is located in a different path you *must* enter the correct path, using the format shown in the example above.

△ **NOTE:** It is important that you enter the proper path so you can follow the steps in this tutorial.

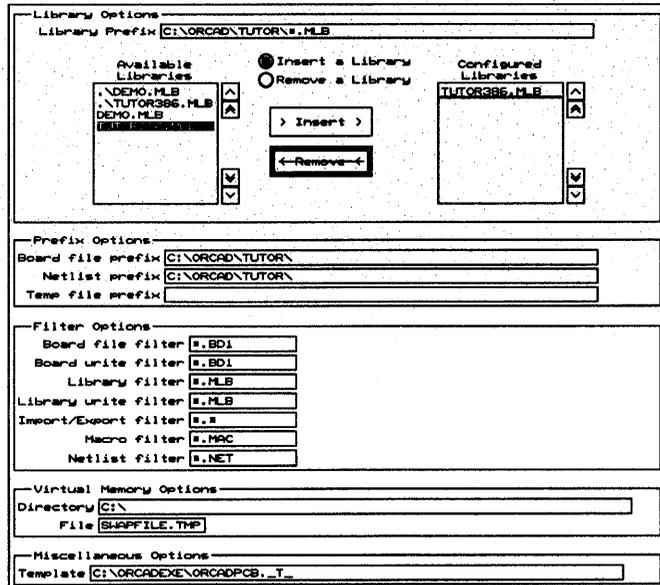


Figure 4-3. Bottom of the *Configure PC Board Layout* screen.

5. Select **TUTOR386.MLB** from the **Available Libraries** list box, then select **Insert**. The filename displays in the **Configured Libraries** list box. You need **TUTOR386.MLB** configured when you load a netlist in *Chapter 6: Placing the TUTOR board*.

△ *NOTE: If a module library is not configured in the **Configured Libraries** list box, then all module libraries in the **Available Libraries** list box are automatically configured.*

6. Enter the path for **Board file prefix** and **Netlist prefix** if they are different from the default directory paths.
7. Enter the path for **Temp file prefix** if you want to change its default path. This specifies the location of the temporary work file used by **Edit Layout**.
8. Change the file filters in the **Filter Options** area if you use different extensions for the listed file types.
9. If you want to change the directory path and filename of the virtual memory swap file, enter a new path and filename in the **Directory** and **File** entry boxes in the **Virtual Memory Options** area.

△ *NOTE: If you have partitioned drives, or multiple drives, you should place the virtual memory swap file on the partition or drive that has the largest amount of contiguous free disk space.*

10. Change the path and filename of the **Edit Layout** template file in the **Template** entry box if you want to specify a different template file. A template file provides default settings for a new **Edit Layout** work session. See the *PC Board Layout Tools 386+ Reference Guide* for additional information on template files.
11. Scroll to the top of the configuration screen (or press <Home>) and select **OK** to return to the **PC Board Layout Tools** screen.

Configuring Edit Layout

1. Select **Edit Layout**. The menu at right displays.
2. Select **Local Configuration**, then select **Configure PCB386**. The **Configure Edit Layout** screen displays (figure 4-4).

Edit Layout

Execute
Local Configuration
Assign Hot Key
Show Version
Configure Layout Tools
Help

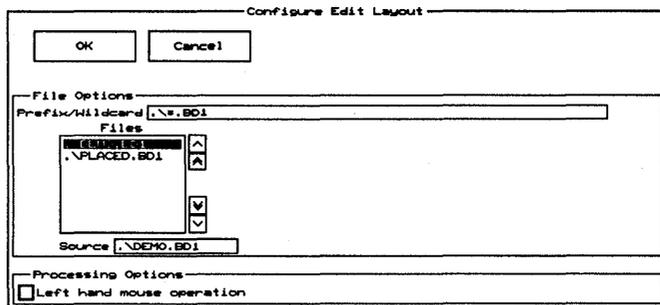


Figure 4-4. The *Configure Edit Layout* screen.

3. Select **.\DEMO.BD1** from the **Files** list box. This specifies which board file automatically loads when you run **Edit Layout**. Files displayed in the **Files** list are in the current design directory, and are selectively listed by entering a unique filename prefix or extension in the **Prefix/Wildcard** entry box.
4. If you want to reverse the function of the left and right mouse buttons in **Edit Layout**, select **Left hand mouse operation**.

△ **NOTE:** In this manual, all references to the left and right mouse buttons are based on the assumption that **Left hand mouse operation** is selected.

5. Select **OK** to save the changes and return to the **PC Board Layout Tools** screen.

Running Edit Layout

Now that you have selected the DEMO board, you are ready to begin learning about **Edit Layout**.

1. Select **Edit Layout**. The **Edit Layout** menu displays.
2. Select **Execute**.

Edit Layout is now running, and the DEMO board displays. The **Edit Layout** work area is larger than the layout, so only part of the work area is visible. You can think of the current screen as a window into the larger work area.

Moving around the screen

Use the mouse to move the pointer around the layout. When the pointer reaches the edge of the screen, the display automatically pans to expose adjacent areas of the layout.

Press the up, down, left, or right arrow keys (the keys on the main keyboard, not the arrow keys on the numeric keypad) to move the pointer one space at a time. The size of the space is determined by the use of a snap grid and the zoom factor.

Very precise routing and object placement is possible using keyboard keys and combinations of the **ZOOM** and **SET Grid Size** commands. See *Changing your view of the layout* later in this chapter for a description of **ZOOM** commands, and *Setting grid options* for a description of **SET Grid Size**.

Edit Layout command basics

Menus guide you from step to step in **Edit Layout**. **Edit Layout** organizes commands and program options using menus, command lines, and dialog boxes. You select a command or option by either clicking the mouse or pressing a key.

See the *PC Board Layout Tools 386+ Reference Guide* for complete command descriptions.

Displaying the main menu

Commands

Press <Enter> or click the left mouse button to see the main menu (shown at right). Press <Esc> or click the right mouse button to dismiss the main menu and return to the **Edit Layout** screen.

There are several ways to select and use a command. You can use the methods shown in table 4-1 in any combination. The method you use is a matter of personal preference.

Block
Cut
Delete
Edit
Find
Go To Function
Highlight
Inquire
Jump
Track Delete
Layer
Move
Origin
Place
Quit
Route
Set
Selective
Undelete
Verbose Inquire
Window Zoom
X show RatsNest
Zoom
= bookmark
+ layer
- layer
* layer
/ other
? conditions
% macro

	<i>Using the keyboard</i>	<i>Using the mouse</i>
To highlight a menu command	Press the up and down arrow keys to slide the highlighting over the command.	Move the mouse to slide the highlighting over the command.
To select a highlighted menu command	Press <Enter>.	Click the left mouse button.
To select any command	Press the highlighted letter in the command name.	

Table 4-1. Using the keyboard or mouse to select a menu command.

The command interface

Edit Layout responds to a command by either performing the command's function or displaying another menu, a command line, or a dialog box.

Menus

All menus look and work just like the main menu. Press <Esc> or click the right mouse button to return to the menu or command line that called the current menu. Follow these steps to familiarize yourself with these processes:

1. Press <Enter> to display the main menu.
2. Select **QUIT**. The menu shown at right displays.
3. Press <Esc> to dismiss the menu.

Update Board File
Write Board File
Initialize Board File
Erase All Routes
Flush Undelete Buffer
Cleanup Stubs
Suspend to System
Abandon Program

Command lines

Command lines are a series of command names listed across the top of the screen. When a command line displays, you can move the pointer around the work area and select a command by pressing the character on the keyboard that corresponds to the highlighted character in the command name.

If you prefer selecting a command with the mouse, rather than typing the highlighted character in the command name, press <Enter> or click the left mouse button to display a menu containing the same commands.

Press <Esc> or click the right mouse button to return to the menu or command line.

1. Press <Enter> to display the main menu.
2. Select **BLOCK**. A command line displays across the top of the screen. Part of the **BLOCK** command line is shown below.

BlockEnd	Set	Cut	Delete	Edit	Find	Go	To	Editor
----------	-----	-----	--------	------	------	----	----	--------

3. Press <Esc> to return to the main menu level.

Dialog boxes Dialog boxes are used to select options, add, revise, or delete selectable items, and enter keyboard data for program control. Dialog boxes give **Edit Layout** enormous power and flexibility. A dialog box can be displayed and used to change an item in the board file even while you are performing another task, such as moving a group of objects or manually routing.

Dialog boxes can be accessed from other dialog boxes, creating a flexible, hierarchical command structure. For example, the **Edit Pad** dialog box is displayed for a module pad that needs to be changed. The **Module Properties** dialog box can be accessed from there, which is used to change the pad's parent module properties, like text visibility or text size.

Dialog box items Dialog boxes in **Edit Layout** may contain these items:

Button A button performs a task or branches to another dialog box. Click the left mouse button on it to perform the button's function.

In **Edit Layout**, a button can be either active or inactive, depending on the action selected in the dialog box. Active buttons in the **Edit Layout** screen display as shadowed, three-dimensional rectangles. Inactive buttons display as gray-filled rectangles. Selecting an item in the dialog box may change an inactive button to active, or an active button to inactive.



Active button, as shown in Edit Layout.



Inactive button, as shown in Edit Layout.



Active button, as shown in this manual.



Inactive button, as shown in this manual.

This manual shows all buttons as black outlined rectangles with black labels. Refer to **Edit Layout** to identify inactive buttons.

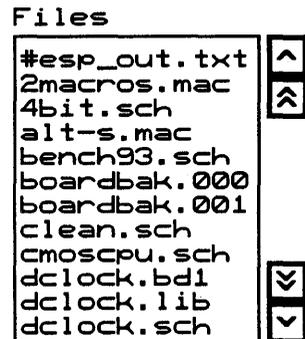
△ **NOTE:** *There are two buttons that are common to almost all dialog boxes: **OK** and **Cancel**. Select **OK** to close the dialog box and incorporate any changes. Select **Cancel** to close the dialog box without incorporating any changes.*

*If the change made in the dialog box involves the addition or deletion of items, the **Cancel** button becomes a **Close** button, signifying that changes made in the dialog box cannot be reversed or undone. This is different from the ESP design environment, where the **Cancel** button does not change to **Close**.*

List box

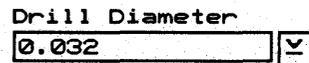
A list box contains a list of selectable items. Place the pointer on the item and click the left mouse button to highlight and select the item.

Scroll buttons accompany a list box when the list of items is longer than the area displaying them. See *Scroll buttons* in this section.

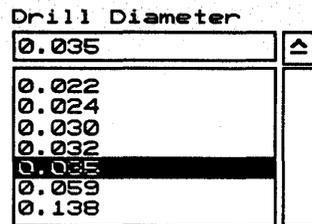


Droplist box

A droplist box contains a list of selectable items that are viewed by placing the pointer in the selected item window or on the droplist button and clicking the left mouse button. Place the pointer on a droplist item and click the left mouse button to highlight and select the item.



Closed droplist box.



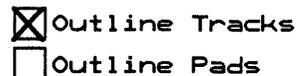
Open droplist box.

Scroll buttons accompany a droplist box when the list of items is longer than the area displaying them. See *Scroll buttons* in this section. When scroll buttons accompany a droplist box, you can use the <Page Up> and <Page Down> keys to scroll the list up or down one window-full at a time. You use the up and down arrow keys to scroll the list up or down one item at a time.

To close an opened droplist box without making a selection, select the droplist button or press <Esc>.

Check box

A check box is an option that can be enabled or disabled. Place the pointer on the check box and click the left mouse button to select it. Highlight the check box to enable the option. Select the check box again to remove the highlight and disable the option.



In this manual, an enabled check box is represented by a square outline with an 'X' in it, and a disabled check box is represented by an unfilled square outline. See the example shown above.

Radio button

Radio buttons are used in lists of mutually exclusive items: only one button can be active at a time. To activate a button, place the pointer on it and click the left mouse button.

- Component Copper
- Internal Copper 1
- Internal Copper 2
- Internal Copper 3
- Internal Copper 4

Entry box

An entry box is a field that accepts typed characters or a field with characters in it that can be changed.

Size X	0.040
Size Y	0.040
Offset X	0.000
Offset Y	0.000

Place the pointer in the entry box and click the left mouse button. A vertical bar appears at the pointer location. This is the text cursor, which you can move left or right within the character string by pressing the left and right arrow keys.

Press <Backspace> and <Delete> to delete the character to the left and right of the cursor, respectively.

Press <Ctrl><Backspace> and <Ctrl><Delete> to delete all characters to the left and right of the cursor, respectively. Press <Alt> <Backspace> to delete all characters in the entry box, regardless of cursor position.

Press <Enter> or the left mouse button to accept changes in the entry box. Press <Esc> or the right mouse button to undo any changes in the entry box.

Scroll buttons

You use scroll buttons to view a list of items that is longer than the window in which they are displayed.

Move the pointer to the scroll button and click the left mouse button to perform the button's task.

You can repeat scroll button tasks by holding down the left mouse button while selecting the scroll button.

Each scroll button is described on the next page:



Scrolls the list up by one item at a time.



Scrolls the list up by one "window-full" of items at a time.



Scrolls the list down by one "window-full" of items at a time.



Scrolls the list down by one item at a time.

How command names are shown in this guide

In this guide, main menu command names are shown in bold uppercase letters. Other command names are shown with just the first letter capitalized. When you are asked to select a command, usually both the main menu command name and other command names are specified.

For example, the statement "Select **PLACE Text**" means "Select **PLACE** from the main menu, and select **Text** from the **PLACE** menu."

Where the context is clear, though, the main menu command is not specified. For example, if the **PLACE** menu already displays, and you are asked to select the **Text** command, the instruction is simply "Select the **Text** command."

Returning to the main menu level

To return to the main menu level from any menu or command line in **Edit Layout**, press <Esc> as many times as necessary until no menu displays in the upper left corner of the screen, or until the main menu command line displays. At this point, the main menu displays if you press <Enter>.

To return to the main menu from a dialog box, select **OK**, **Cancel**, or **Close**, as needed, until no dialog box displays.

You can use these keyboard shortcut keys for the **OK** and **Cancel** buttons:

- ❖ Press <Home> and the pointer jumps to the **OK** button
- ❖ Press <Ctrl><Home> and the pointer jumps to the **Cancel** or **Close** button
- ❖ Press <Enter> or click the left mouse button to select the button under the pointer

Setting up Edit Layout conditions

Now that you understand how **Edit Layout's** menus, command lines, and dialog boxes operate, take some time to become familiar with some of the commands that govern the way **Edit Layout** displays and maintains layouts.

The SET command

1. Press <Enter> to display the main menu.
2. Select SET. The **Global Options** dialog box (figure 4-5) displays.

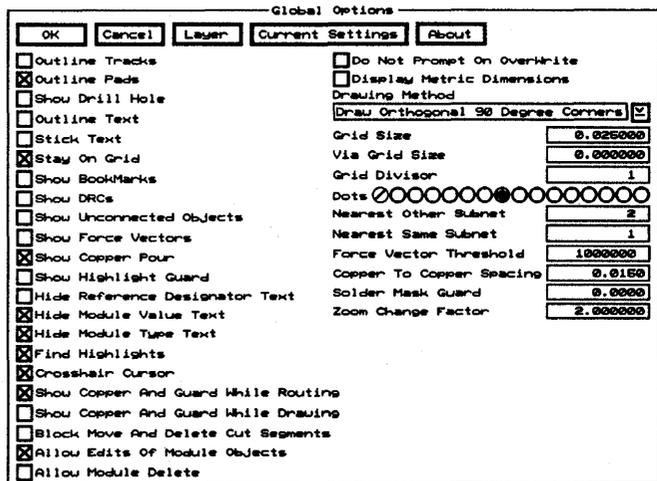


Figure 4-5. The **Global Options** dialog box.

Using the selections in the **Global Options** dialog box, you control features such as object appearance, selectability, cursor style, and grid size. Note the options that are enabled. These settings determine how the **DEMO** board displays.

You can set **Edit Layout's** global options to match your personal preferences, or tailor them to suit the requirements of the board design.

Most of the options in this dialog box are enabled or disabled through the use of check boxes. Other options are changed by selecting an item in a droplist box, or by editing a value in an entry box.

Layer and Current Settings display additional options for **Edit Layout**.

The **Global Options** dialog box is accessible from many other dialog boxes during an editing session, so you can alter global settings at any point in the design process.

Layer You select **Layer** to specify the number of layers that a board uses, which layer is the current active layer, and the display color for each layer.

1. Select **Layer** in **Global Options**. The **Layer** dialog box displays (figure 4-6).

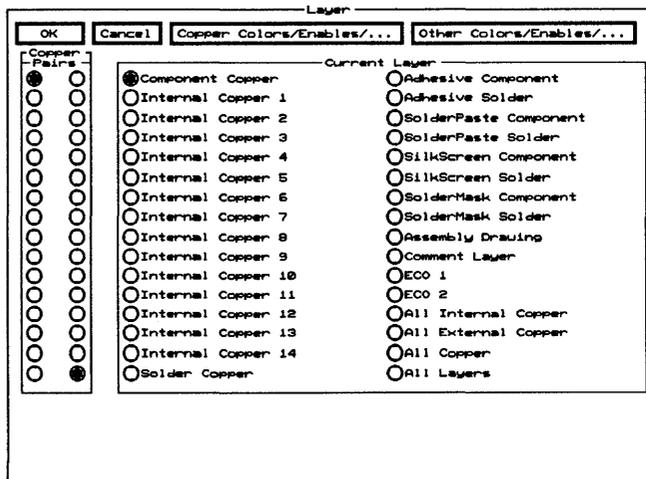


Figure 4-6. The **Layer** dialog box.

2. Select **Copper Colors/Enables/...** to display the **Copper Colors/Enables/...** dialog box (figure 4-7).

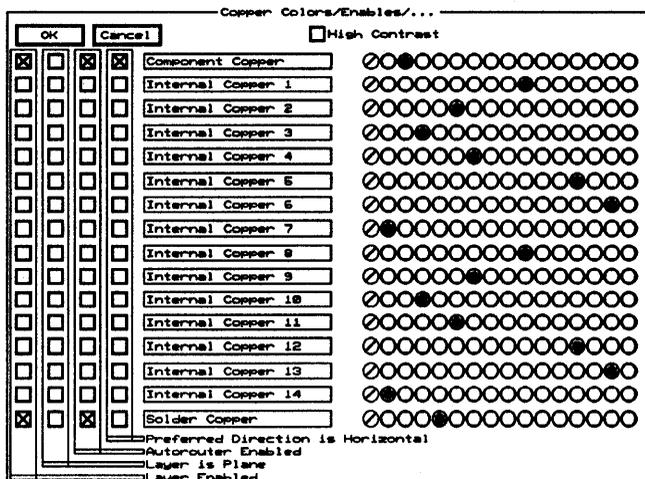


Figure 4-7. The Copper Colors/Enables/... dialog box.

This dialog box sets the number of enabled layers on a board, and specifies how the layers are considered during autorouting.

The **Component Copper** and **Solder Copper** layers are enabled in the **Layer Enabled** column, which specifies that the DEMO board is a two layer board and does not use any internal copper layers.

3. Select **Cancel** to close the **Copper Colors/Enables/...** dialog box and return to the **Layer** dialog box.

You specify the two routing layers in **Copper Pairs**. When you place a via while routing on one of the selected layers in **Copper Pairs**, you continue routing on the other selected layer.

The selections in **Copper Pairs** correspond to the positions of the copper layers listed in the left column of **Current Layer**.

The **Component Copper** and **Solder Copper** layers are the only layers enabled in **Copper Colors/Enables/...**, so they are the only valid selections in **Copper Pairs**.

Note that **Component Copper** is selected in **Current Layer**, which sets it as the current working layer.

4. Select **CLOSE** to close the **Layer** dialog box and return to **Global Options**.

Current Settings

When you create a new object, **Edit Layout** applies default options and properties to it. You set these default options and properties through the use of the **Current Object Settings** dialog box.

1. Select **Current Settings** in **Global Options**. The **Current Object Settings** dialog box displays (figure 4-8).

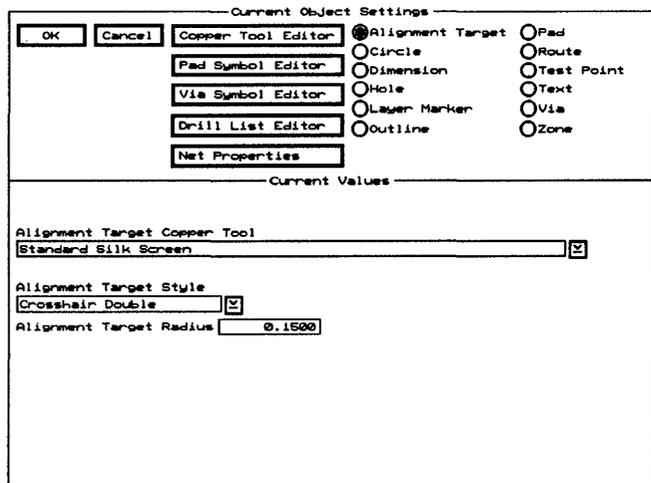


Figure 4-8. The *Current Object Settings* dialog box.

This dialog box establishes default settings for all new objects placed in **Edit Layout**. You select an editor button to display the editor dialog box. You select a radio button to display in **Current Values** the current settings for the selected object.

2. Select **Text**. The current text values display in **Current Values**, as shown in figure 4-9.

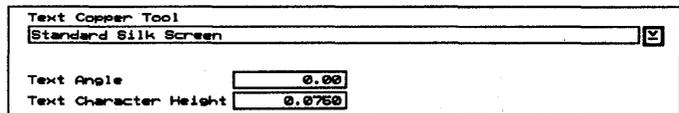


Figure 4-9. Current values for text, as shown in *Current Object Settings*.

3. Place the pointer inside the **Text Character Height** entry box and click the mouse. The text cursor displays.
4. Delete the current entry, then enter **0.1000** to change the text height. All text placed in **Edit Layout** using the **PLACE Text** command after making this change is 0.1000 inch high. Any text placed before making this change is unaffected.
5. Select **OK** to accept the change and return to **Global Options**.
6. Select **OK** to close the **Global Options** dialog box.

Selecting a layer

Edit Layout has many commands that select a particular layer, or that toggle through specified groups of layers. Follow these steps to learn about the ways you can select layers:

- LAYER**
1. Display the main menu and select **LAYER**. The **Layer** dialog box displays.
 2. Select **SilkScreen Component** in **Current Layer**, then select **OK**. **SilkScreen Component** displays at the bottom of the screen, indicating it is the current layer.

/ OTHER Selecting **/ OTHER** activates one of the two copper layers selected in **Copper Pairs** in the **Layer** dialog box. Follow these steps to toggle between the Component Copper and Solder copper layers:

1. Select **/ OTHER** from the main menu. The current layer changes from SilkScreen Component to Component Copper.
2. Select **/ OTHER**. The current layer changes to Solder Copper.
3. Select **/ OTHER** again to toggle the current layer back to Component Copper.

+ LAYER and – LAYER You select **+ LAYER** to incrementally select each enabled copper layer, from the lowest copper layer (Component Copper) to the highest copper layer (Solder Copper).

You select **– LAYER** to incrementally select each enabled copper layer, from the highest copper layer (Solder Copper) to the lowest copper layer (Component Copper).

Follow these procedures to learn about **+ LAYER** and **– LAYER**:

1. With Component Copper as the current layer, select **+ LAYER** to set a higher enabled copper layer (Solder Copper) as the current layer.
2. Select **– LAYER** to set a lower enabled copper layer (Component Copper) as the current layer.

The **+ LAYER** and **– LAYER** commands are useful when you are manually routing between external and internal copper layers on a multilayer board.

*** LAYER** Select *** LAYER** to set All Layers as the current layer. With All Layers selected, you can select any object on any enabled layer.

*** LAYER** is useful when you need to edit many objects on a board and the objects are on different layers.

Changing your view of the layout

Edit Layout provides many options for displaying board layouts and specific objects on the board.

ZOOM

Edit Layout can display layouts at many different magnification scales. You change the view size using the **ZOOM** command. The layout can be zoomed in or out to magnify or reduce its visible image.

When **Edit Layout** is zoomed out, you can see a large portion of the layout. Zooming in enlarges a small portion of the layout and displays more details. You can zoom in to draw intricate areas of your layout with exacting detail and then zoom out to look at the finished layout.

To zoom out and see more of the work area on the screen at one time, follow these steps:

1. Select **ZOOM** from the main menu. The menu shown at right displays.
2. Select **Out**. A reduced view of the layout displays.
3. Experiment with the scale using **In**, **Out**, and the numeric zoom scales.

The numeric values in the **ZOOM** menu represent the number of mils per displayed pixel. A zoom scale of 1 is then 1 pixel=1 mil (.001 inch). A scale of 5 is then 1 pixel=5 mils (.005 inch). A scale of .01 is 1 pixel=.01 mil (.00001 inch), or 100 pixels=1 mil.

The zoom scale range in **Edit Layout** is from .01 (maximum magnification) to 100 (minimum magnification).

Center	
In	
Out	
Previous	
Refresh	
Set Scale	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
20	T
50	F
100	H
Window	

The current zoom scale is displayed at the bottom of the screen, to the right of the pointer coordinates.

Setting a zoom scale

You can set the zoom scale to any value from 0.01 to 100. Follow these steps to display to entire DEMO board:

1. Select **ZOOM Set Scale**. The **Set Zoom Scale** dialog box displays (figure 4-10).
2. Enter **13** in the **Scale** entry box, then select **OK**. Pan the display, if necessary, until the entire board is visible.

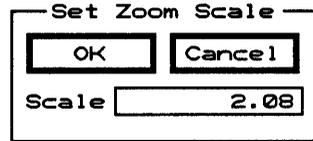
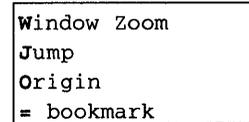


Figure 4-10. The **Set Zoom Scale** dialog box.

Selecting a zoom window

Follow these steps to zoom in on a selected area of the TUTOR board:

1. Select **ZOOM Window**.
2. Place the pointer at (1.8000", 1.5000"), then click the mouse or press <Enter>. The menu at right displays.
3. Select **Window Zoom**. A bounding box displays as you move the pointer. This box represents the zoom window.
4. Move the pointer to (4.5000", 3.5000") and select **Window Zoom End**. The display magnifies, filling the screen with the selected area.

**WINDOW ZOOM**

The same window zoom capability is available from the main menu when you select **WINDOW ZOOM**.

1. Place the pointer at (2.0000", 1.7000") and select **WINDOW ZOOM**.
2. Move the pointer to (3.2000", 2.5000") and select **Window Zoom End**. The display changes to show the new zoom window.
3. Select **ZOOM Set Scale**, then enter **13** in the **Scale** entry box.
4. Select **OK**. The entire DEMO board displays.

Pointer movement resolution

The maximum pointer movement resolution with **Stay On Grid** disabled is one ten-thousandth of an inch (0.0001 inch), or 0.1 mil., at the zoom levels shown in table 4-2.

Hold down the <Ctrl> key and press an arrow key to move the pointer five grid spaces at a time. Hold down the <Alt> key and press an arrow key to move the pointer in that direction to the edge of the current window.

The following table lists pointer movement resolution for the zoom levels in the **ZOOM** menu. The first entry in the table (Zoom level 0.01 to 0.1) is not in the **ZOOM** menu, but is included in the table to show the zoom range for maximum pointer movement resolution. The resolution is shown in decimal inches, and is achieved using the arrow keys and <Ctrl> arrow keys with **Stay On Grid** disabled.

<i>Zoom level</i>	<i>Arrow key resolution</i>	<i><Ctrl> Arrow key resolution</i>
0.01 to 0.1	0.1 (.0001 inch)	0.5 (.0005 inch)
1	1 (.001 inch)	5 (.005 inch)
2	2 (.002 inch)	10 (.010 inch)
3	3 (.003 inch)	15 (.015 inch)
4	4 (.004 inch)	20 (.020 inch)
5	5 (.005 inch)	25 (.025 inch)
6	6 (.006 inch)	30 (.030 inch)
7	7 (.007 inch)	35 (.035 inch)
8	8 (.008 inch)	40 (.040 inch)
9	9 (.009 inch)	45 (.045 inch)
10	10 (.010 inch)	50 (.050 inch)
20 T	20 (.020 inch)	100 (.100 inch)
50 F	50 (.050 inch)	250 (.250 inch)
100 H	100 (.100 inch)	500 (.500 inch)

Table 4-2. Pointer movement, by arrow key, for each zoom level.

Using bookmarks

A bookmark is a reference point that Edit Layout places on the board layout. You assign a unique name to the bookmark, then use the JUMP command to move the pointer to the selected bookmark.

A bookmark name can be up to 40 characters long, and can contain spaces and special ASCII characters. Assigning a descriptive name to a bookmark makes it easy to recognize.

Creating a bookmark

1. Select **SET**. The **Global Options** dialog box displays.
2. Enable **Show BookMarks**, then select **OK**.
3. Move the pointer to where you want the bookmark placed.
4. Select = **BOOKMARK**. The **Bookmark** dialog box displays (figure 4-11).

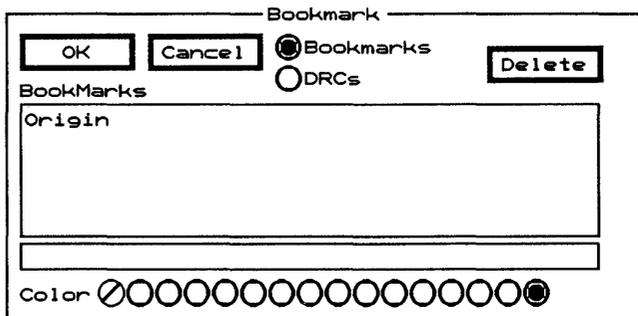


Figure 4-11. The **Bookmark** dialog box.

Note that the **BookMarks** list box already contains a bookmark named **Origin**. This bookmark is automatically created when you start a design and it cannot be deleted.

Also, note the two radio buttons labeled **BookMarks** and **DRCs**. These selections regulate what types of bookmarks display in the list box. Select **BookMarks** to display all bookmarks in the list box.

The DRCs button works the same way. A DRC is a special type of bookmark (DRC stands for *Design Rule Check*) that **Edit Layout** places on the board design where it detects a design error. See *Chapter 7: Routing the TUTOR board* for more information on DRC. You select DRCs to display all DRC markers in the list box.

5. Enter the following name in the **BookMarks** entry box:
My first bookmark
6. Select a bookmark color from the color list. Organizing your bookmarks by color helps you locate them on the display.
7. Select **OK** to accept the entries and close the dialog box. The symbol shown at right displays at the pointer location.



The letter *B* in the symbol indicates it is a bookmark. The converging lines point to the location on the board that is referenced by the bookmark.

△ **NOTE:** *Bookmarks always display at the same size in Edit Layout, regardless of the current zoom scale.*

Jumping to a bookmark

You use **JUMP** to select a bookmark name from a list in a dialog box. The pointer jumps to the selected bookmark location when you close the **Jump To** dialog box.

Use the following steps to jump to the bookmark you just placed:

1. Move the pointer away from the bookmark.
2. Select **JUMP**. The **Jump To** dialog box displays (figure 4-12).

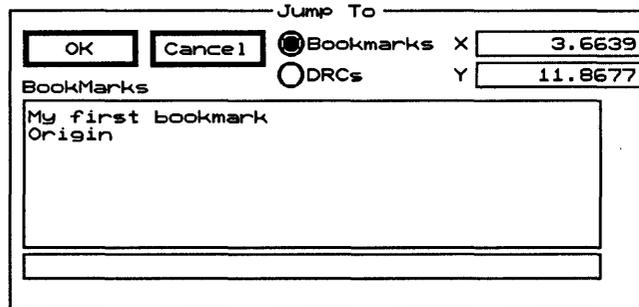


Figure 4-12. The *Jump To* dialog box.

3. Select **My first bookmark** from the **BookMarks** list box, or enter the name in the entry box below the list box. The name entered in the entry box must exactly match the bookmark name, including capitalization and spaces. The name highlights in the list box, and the bookmark's coordinates appear in the **X** and **Y** entry boxes.
4. Select **OK**. The pointer moves to the bookmark location.

You can also jump to a known set of coordinates by entering them in the **X** and **Y** entry boxes and selecting **OK**.

Deleting a bookmark

You may want to delete a bookmark at some stage of the design process. The reference may not be required any longer, and you want to delete the bookmark to reduce the number of displayed objects.

Follow these steps to delete a bookmark by selecting its name:

1. Select = **BOOKMARK**. The **Bookmark** dialog box displays.
2. Select **My first bookmark** in the **BookMarks** list box, or enter the name in the entry box. The name highlights in the list box.
3. Select **Delete**. The name disappears from the list box and the bookmark is deleted from the board.
4. Select **Close** to close the dialog box.

- △ *NOTE: You can also delete any bookmark by placing the pointer on the bookmark symbol and selecting **DELETE** from the main menu. Using this method, you can undelete the bookmark by selecting **UNDELETE**.*

Changing the origin

The **ORIGIN** command resets the current pointer position as (0.0000", 0.0000"). Edit Layout reports locations in relation to the new origin until you change it again.

Follow these steps to locate the origin and reposition it.

1. Select **JUMP** from the main menu. The **Jump To** dialog box displays.
2. Select **Origin** in the **BookMarks** list box. The selection highlights and displays in the entry box, and the coordinates for the bookmark display in both the **X** and **Y** entry boxes.
3. Select **OK**. The pointer jumps to the origin.
4. Move the pointer to (0.7000", 0.5000") and select **ORIGIN**. The origin bookmark, and position (0.0000", 0.0000), are at the new pointer location.
5. Move the pointer to (-0.7000", -0.5000"), the far upper left corner of the display, and select **ORIGIN** to reposition the origin.

- △ *NOTE: The **ORIGIN** command is accessible from many menus and command lines in **Edit Layout**. You can set the origin even while you perform tasks such as moving objects or manually routing.*

Setting grid options

While working on a layout, it is important to properly space and align objects on the board. This is done by using a grid.

You use the grid options in the **Global Options** dialog box to set up a grid. These options set grid spacing and appearance, and whether or not to stay on grid.

Setting a grid size

Grid Size defines the space between points on a snap grid. **Grid Size** also sets the routing grid for autorouting. The allowed range for the **Grid Size** entry box is from one tenth of a mil (0.0001 inch) to 33 inches.

1. Select **SET** to display the **Global Options** dialog box.
2. Enter **0.050000** in the **Grid Size** entry box to specify a 50 mil grid, then select **OK**.
3. Move the pointer to the origin and use the arrow keys to move the pointer one grid space at a time. Notice that the coordinates in the lower left change in 50 mil increments.



***NOTE:** All entry boxes in **Edit Layout** have predefined allowable ranges for values. If you enter a value that is beyond the allowed range for the entry box, the entry is not accepted and the following message appears in the upper left part of the screen:*

Allowed Range: <min value> to <max value>

Setting a grid divisor

Grid Divisor establishes minor grid divisions for both the **Grid Size** and **Via Grid Size** options.

1. Select **SET** to display the **Global Options** dialog box.
2. Enter **3** in the **Grid Divisor** entry box. **Grid Divisor** establishes minor grid divisions for both **Grid Size** and **Via Grid Size**.
3. Select **OK**.
4. Move the pointer to the origin, then use the arrow keys to move the pointer. Note that the 50 mil grid is divided into three 16.7 mil grid spaces.
5. Select **SET** to display **Global Options**.
6. Enter **0.025000** in **Grid Size** and a **1** in **Grid Divisor** to return the grid to its original setting.

Disabling the snap grid

1. Disable **Stay On Grid**, then select **OK**.
2. Use the arrow keys to move the pointer around the display. The pointer movement is not constrained to the grid.
3. Select **SET** to display **Global Options**.
4. Enable **Stay On Grid**, then select **OK**. The pointer moves on grid when you press the arrow keys.



*NOTE: Leave **Stay On Grid** enabled unless you have a compelling reason to work off grid.*

Changing the grid color

1. Select **SET** to display **Global Options**.
2. Select a grid dot color in **Dots**, then select **OK**. Zoom in until you see the grid dots displayed with the selected color.



*NOTE: If you want the grid dots to be invisible, select **black** in **Dots**.*

Saving and backing up the board file

When you work on a layout for a long time, it is important to save your work on disk periodically as a precaution against power failures and other unexpected events.

These procedures show you how to save, copy, and rename board files.

Updating the file

To update the board file, follow these steps:

1. Select **QUIT**. The **QUIT** menu at right displays.
2. Select **Update Board File**. **Edit Layout** saves the file to the same file name in the current design.

Update Board File
Write Board File
Initialize Board File
Erase All Routes
Flush Undelete Buffer
Cleanup Stubs
Suspend to System
Abandon Program

Update Board File also creates a backup file, which is the last saved version of the board. The backup file has a .BAK extension, and is also in the current design.

Saving configurations

Configurations are automatically saved with the board file when you select **QUIT Update Board File** or **QUIT Write Board File**. When you reload the file into **Edit Layout**, all previously defined configurations for the board are used.

Writing to another filename

To save the board to a different filename, follow these steps:

1. Select **QUIT Write Board File**. The **Write Board File** dialog box displays (figure 4-13).

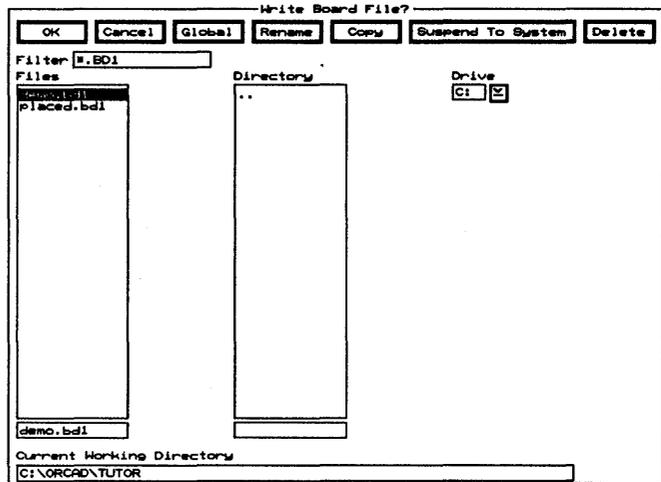
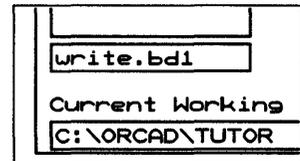


Figure 4-13. The **Write Board File** dialog box.

DEMO.BD1 is highlighted in the **Files** list box, and displays in the list entry box. The current drive, working directory, and any subdirectories of the current working directory also display in the dialog box.

2. Enter **WRITE.BD1** in the entry box beneath the **Files** list box, then select **OK**. **WRITE.BD1** is written to the current working directory. The **Edit Layout** screen displays.



△ **NOTE:** You can use **QUIT Write Board File** to create incremental backups of your board file. Use the procedures described above and change the filename each time you save.

Copying a file

Follow these steps to copy the file you just saved, WRITE.BD1, to another filename:

1. Select **QUIT Write Board File**. The **Write Board File** dialog box displays and WRITE.BD1 is selected in the **Files** list box.
2. Select **Copy**. The **Copy File** dialog box displays (figure 4-14) and WRITE.BD1 is selected in the list box.

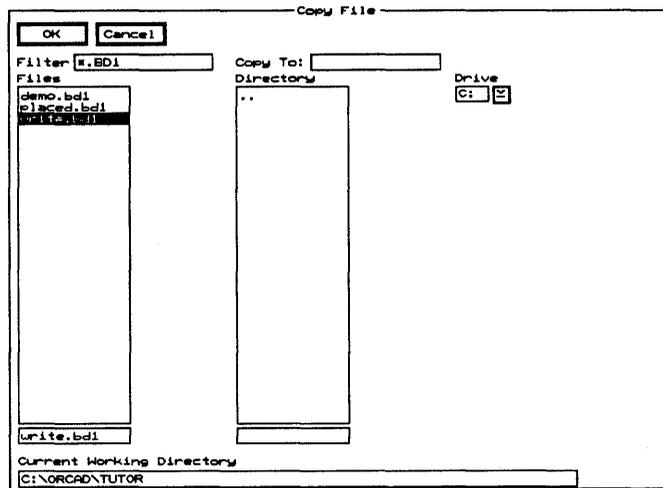


Figure 4-14. The *Copy File* dialog box.

3. Enter **COPY.BD1** in the **Copy To:** entry box, then select **OK**. The file is copied and the **Write Board File** dialog box displays. **COPY.BD1** displays in the **Files** list box.
4. Select **Close** to dismiss the **Write Board File** dialog box.

Renaming a file

Follow these steps to rename COPY.BD1:

1. Select **QUIT Write Board File**. The **Write Board File** dialog box displays.
2. Select **COPY.BD1** in the **Files** list box, then select **Rename**. The **Rename File** dialog box displays (figure 4-15).

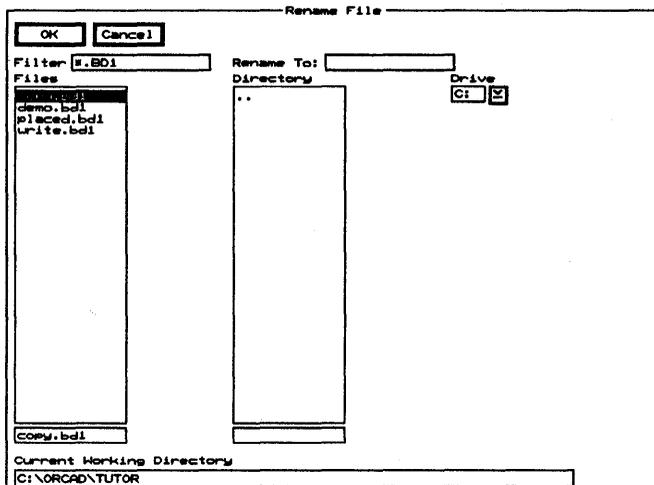


Figure 4-15. The Rename File dialog box.

3. Enter **RENAME.BD1** in the **Rename To:** entry box and select **OK**. The **Write Board File** dialog box displays, and **RENAME.BD1** displays in the **Files** list box.
4. Select **Close** to dismiss the dialog box.

Deleting a file

After a design is complete, you may wish to delete some of the older versions of the board file. Follow these steps to delete **WRITE.BD1** and **RENAME.BD1**:

1. Select **QUIT Write Board File**. The **Write Board File** dialog box displays.
2. Select **RENAME.BD1** in the **Files** list box, then select **Delete**. The file is deleted from the disk and removed from the **Files** list box.
3. Select **WRITE.BD1** and select **Delete**. The file is deleted and removed from the **Files** list box.
4. Select **Close** to dismiss the dialog box.



NOTE: You can also copy, rename, and delete a file by selecting **QUIT Initialize Board File**, then selecting the file in the **Initialize to Board File** dialog box and selecting **Copy**, **Rename**, or **Delete**.

Suspend to System

Most of the file maintenance tasks previously described, as well as running other programs, can be executed from the DOS command line while **Edit Layout** is suspended in the background.

Follow these steps to access the DOS command line from **Edit Layout**:

1. Select **QUIT Suspend to System**. The **Edit Layout** screen disappears and the DOS prompt displays.

A right arrow bracket (>) is added to the end of the DOS prompt, indicating that **Edit Layout** is suspended in the background.

2. Enter **EXIT** to close the DOS editing session and return to **Edit Layout**.

Suspend to System is also available from many of the dialog boxes in **Edit Layout**.

△ *NOTE: Suspend to System is operational only on computers that have a floating point coprocessor, such as an 80387, or built-in coprocessing functions.*

Macros

Macros record virtually anything you do in **Edit Layout**, so you can automate many repetitive tasks and speed up your work. You assign a macro to a key or combination of keys, then press the key to execute the recorded macro. See the *PC Board Layout Tools 386+ Reference Guide* for a list of valid keys.

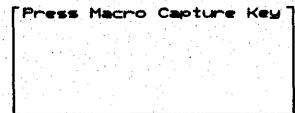
Macros in **Edit Layout** are relative-event data captures. This means that a macro executes its commands relative to the current pointer location, rather than from the original location when the macro was created.

You can record macros using either menu selections or keystrokes. A macro created using keystrokes runs faster because there are fewer events to process when the macro is played back.

The following steps describe how to create two macros. The first macro is assigned to the <Alt><S> keys, and records the commands to draw an outline and place a text string inside the outline. The second macro is assigned to function key <F1>, and places a line of text on the screen.

Creating the first macro

1. Select **SET** to display the **Global Options** dialog box.
2. Enter 0.100000 in **Grid Size** to change the grid size to one tenth of an inch. Be sure **Stay On Grid** is enabled, then select **OK**.
3. Move the pointer to (0.0000", 3.5000").
4. Select **ZOOM**, then select scale 5 to magnify the display.
5. Select **ORIGIN** to set the current pointer location to (0.0000", 0.0000").
6. Select **% MACRO**. The **Press Macro Capture Key** box at right displays.



7. Press <Alt><S>. **Alt S** displays in the **Press Macro Capture Key** prompt box, as shown at right. The macro internal code `\x011F` also displays.

```

Press Macro Capture Key
      Alt S
      \x011F

```

△ **NOTE:** If you press an invalid key combination, it does not display in the **Press Macro Capture Key** prompt box. Select a valid key or combination.

8. Press <Enter>. The highlighted prompt “Macro Capture” displays in the lower right corner of the screen, reminding you that you are defining a macro. Any commands you execute while “Macro Capture” displays, including pointer movements and selections, are added to the list of commands stored in the macro.
9. Type **P O B** for **PLACE Outline Begin**.
Use the arrow keys to move the pointer (not the arrow keys on the numeric keypad). The pointer moves one grid dot at a time, drawing the outline segment as it moves. You could use the mouse to draw the outline, but using the arrow keys makes it easier to follow the coordinates.
10. Use the right arrow key to move the pointer to (1.5000", 0.0000"). Refer to the coordinates in the lower left corner of the screen. The outline segment appears as a dotted line.
11. Type **B** (for **Begin**). A new outline segment begins at that point. The previous outline segment changes to a solid line.
12. Use the down arrow key to move the pointer to (1.5000", 1.0000").
13. Type **B**. Use the left arrow key to move the pointer to (0.0000", 1.0000").
14. Type **B**. Use the up arrow key to move the pointer to (0.0000", 0.0000"), completing the outline.

15. Type **E** (for End). The outline is complete, and all segments display as solid lines.
16. Type **P T** (for **PLACE Text**). The Text entry box displays (figure 4-16).

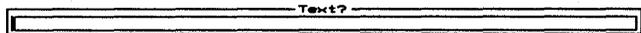


Figure 4-16. The Text entry box.

17. Enter **MACRO**. The text displays in the Text entry box.
18. Use the right and down arrow keys to move the text to (0.7000", 0.5000"), then type **S** (for Set). The Edit Text dialog box displays (figure 4-17).

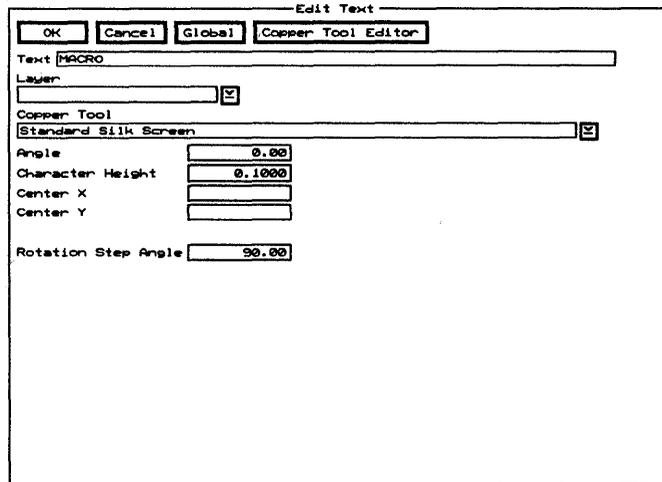
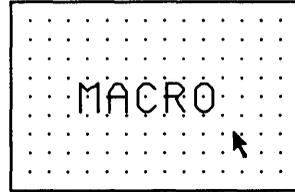


Figure 4-17. The Edit Text dialog box.

19. Place the pointer in the **Character Height** entry box and click the left mouse button. The text cursor displays.
20. Press the <Alt><Backspace> keys to delete the entry, then enter 0.1500.
21. Press <Home> to move the pointer to the **OK** button, then press <Enter> to close the **Edit Text** dialog box. The text is now 0.1500 inches high.

22. Type **P** (for **Place**) to place the text. The **Text** entry box displays. Press **<Esc>** to dismiss the **Text** entry box.
23. Type **%** (for **% MACRO**) to end the macro recording session. The outline and text should look like the example at right. The message "*xxx* events captured" displays. The value "*xxx*" reflects the number of mouse movements, keystrokes, and mouse button clicks you performed while recording the macro.



Creating the second macro

1. Use the cursor keys to move the pointer to (0.7000", 1.2000").
2. Select **% MACRO** again. The **Press Macro Capture Key** box displays.
3. Press **<F1>**. **F1** displays in the **Press Macro Capture Key** box, and the macro internal code also displays.
4. Press **<Enter>**. The message "**Macro Capture**" displays in the lower right corner of the screen, reminding you that you are defining a macro.
5. Type **P T** (for **PLACE Text**). The **Text** entry box displays.
6. Enter **F1 KEY**.
7. Type **P (Place)** to place the text at (0.7000", 1.2000"). Press **<Esc>** to dismiss the **Text** entry box.
8. Type **%** to end this second macro recording session. The message "*xxx* events captured" displays.
9. Before proceeding with the next sections, reset the origin to the far upper left corner of the work area by moving the pointer to (0.0000", -3.5000") and selecting **ORIGIN**.

Running the macros

The completed macros are now stored in memory.

1. Move the pointer to the right of the outline and text you just placed and press <Alt><S> to run the first macro.
2. Move the pointer again and press <F1> to run the second macro.

△ *NOTE: If you need to terminate a macro while it is running, press <Ctrl><Break>.*

The outline segments and text draw at the new pointer position, rather than the original macro starting point of (0.0000", 0.0000"). Remember that macros capture relative events. This means that objects placed on the board relative to the pointer's starting position while recording the macro are placed on the board the same relative distance from the *new* pointer position when the macro is played back.

Saving all macros to a file

In many cases, you will want to use your macros every time you run **Edit Layout**. To do so, you must first save them to a file.

Edit Layout gives you the ability to save all macros defined in one editing session in a single file, or export a single macro to a file. Use the following steps to save the two macros you just created in a single file:

1. Select **GO TO FUNCTION**. The menu shown at right displays. Some of the menu items may display as gray on your screen. A menu item displayed in gray is unavailable and cannot be selected.

Pad Symbol Editor
Via Symbol Editor
Copper Tool Editor
Drill List Editor
Net Property Editor
Library Editor
Autrouter
Netlist Loader
Printing and Plotting
Macro Maintenance

2. **Select Macro Maintenance.** The Macro Maintenance dialog box displays (figure 4-18). The two macros you created display in the **Defined Macros** list box. These macros are now stored only in memory.

Note that some of the buttons at the top of the dialog box display as gray. A gray button is presently inactive and cannot be selected; however, selecting another item in the dialog box may change an inactive button to active.

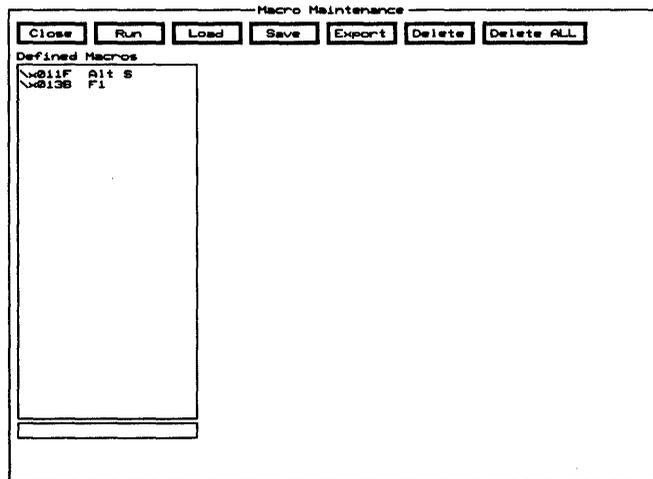


Figure 4-18. The Macro Maintenance dialog box.

3. **Select Save.** The Save ALL Macros to File dialog box displays (figure 4-19).

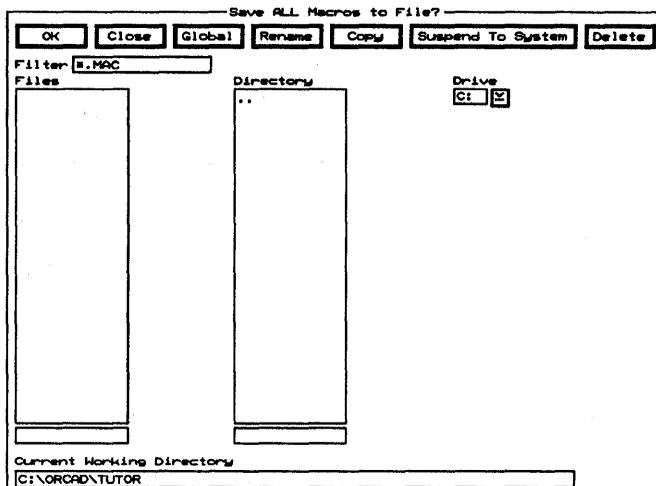


Figure 4-19. The Save ALL Macros to File dialog box.

4. Place the pointer in the entry box below the Files list box and enter the filename **2MACROS.MAC**.

You can use the **Directory** and **Drive** list boxes to change the directory where 2MACROS.MAC will be saved. The destination directory is shown in the **Current Working Directory** entry box. For this example, leave the current working directory at its present setting.

5. Select **OK** to save 2MACROS.MAC in the current working directory. The **Save ALL Macros to File** dialog box closes and the **Macro Maintenance** dialog box displays.

Exporting a macro to a file

To save a single macro from a group, follow these steps:

1. With the **Macro Maintenance** dialog box displayed, select `\x011F Alt S` from the **Defined Macros** list box.
2. Select **Export**. The **Export '\x011F Alt S' Macro to File** dialog box displays (figure 4-20).

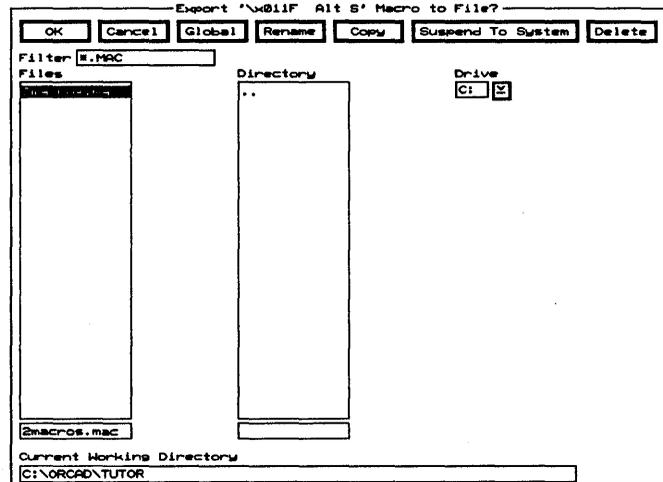


Figure 4-20. The **Export '\x011F Alt S' Macro to File** dialog box.

3. Enter `ALT-S.MAC` in the entry box below the **Files** list box.

You can use the **Directory** and **Drive** list boxes to change the directory where `ALT-S.MAC` is saved. The destination directory is shown in the **Current Working Directory** entry box. For this example, leave the current working directory at its present setting.

4. Select **OK** to export the macro to `ALT-S.MAC`. The **Save '\x011F Alt S' Macro to File** dialog box closes and the **Macro Maintenance** dialog box displays.

△ **NOTE:** If you save or export a macro to a file that already exists, *Edit Layout* asks if you want to overwrite the existing file. If you select **OK**, the new file is written and the old file receives a `.BAK` extension.

Deleting a macro from the disk

1. With the **Macro Maintenance** dialog box displayed, select **Load**. The **Load ALL Macros from File** dialog box displays and **ALT-S.MAC** is selected (figure 4-21).

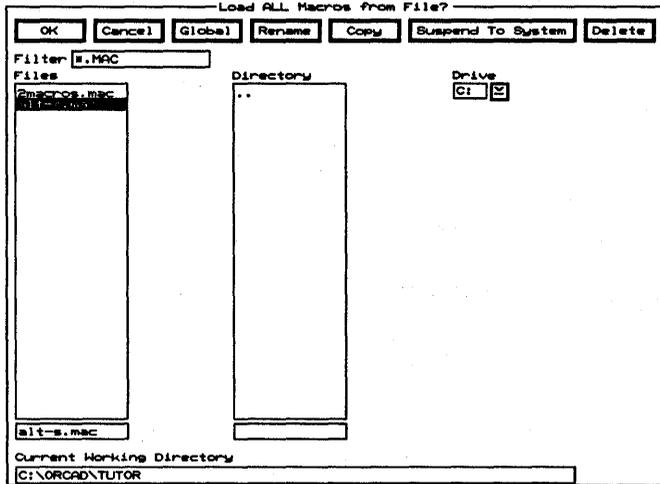


Figure 4-21. The **Load ALL Macros From File** dialog box.

2. Select **Delete**. The file is deleted from the disk and its filename does not display in the list box. The **Delete** button becomes inactive.
3. Select **Close** to exit the dialog box and return to the **Macro Maintenance** dialog box.

Deleting a macro from Edit Layout

Follow these steps to delete individual macros that are in **Edit Layout** memory:

1. With the **Macro Maintenance** dialog box displayed, select **Alt+F1** in the **Defined Macros** list box. The selection highlights and displays in the entry box.
2. Select **Delete**. The macro is deleted from memory and its name disappears from the list box.

Deleting all macros from Edit Layout

With the **Macro Maintenance** dialog box displayed, select **Delete ALL**. All macros in **Edit Layout** memory are deleted and no macros display in the list box.

**Loading a macro
from disk**

Follow these steps to load a macro file that is saved on the disk into **Edit Layout**:

1. With the **Macro Maintenance** dialog box displayed, select **Load**. The **Load ALL Macros From File** dialog box displays.
2. Select **2MACROS.MAC** from the **Files** list box. The filename displays in the entry box.
3. Select **OK**. The **Macro Maintenance** dialog box displays and the two macros saved in **2MACROS.MAC** display in the **Defined Macros** entry box. The listed macros are now loaded in **Edit Layout** memory.

**Running a defined
macro**

To run a macro listed in the **Macro Maintenance** dialog box, follow these steps:

1. Select a macro in the **Defined Macros** list box. The selection highlights and displays in the entry box.
2. Select **Run**. The **Macro Maintenance** dialog box closes and the macro executes, starting at the current pointer location.

Summary

In this chapter you learned how to run **Edit Layout** and examine and modify work preferences. You also learned how to create and save macros.

The next chapter gives you detailed instructions for creating circuit board modules using the library editor in **Edit Layout**.



Creating board modules

About modules

A module can be defined as a representation of the physical shape, size, and required pad layout of a component that is mounted to a circuit board.

Although **PC Board Layout Tools 386+** provides extensive libraries containing over 1000 modules, you may occasionally need a module not found in any library. You can modify an existing module or create an entirely new module from within **Edit Layout** using the library editor.

In this chapter you will:

- ❖ Learn library editor commands
- ❖ Edit an existing module
- ❖ Create a module library
- ❖ Import and export module files
- ❖ Create a module
- ❖ Save the new module in a library

About the library editor

The library editor in **Edit Layout** gives you the ability to create complex PC board modules, even while a board file is loaded in **Edit Layout**. Modules can be routed while they are in the library editor, and an entire routed board file can be imported, edited, and then exported as a single module that can be placed in another board file.

Selecting the library editor

1. Select **GO TO FUNCTION**. The menu at right displays.
2. Select **Library Editor**. The **Initialize to Library File** dialog box displays (figure 5-1).

- Pad Symbol Editor
- Via Symbol Editor
- Copper Tool Editor
- Drill List Editor
- Net Property Editor
- Library Editor
- Autorouter
- Netlist Loader
- Printing and Plotting
- Macro Maintenance

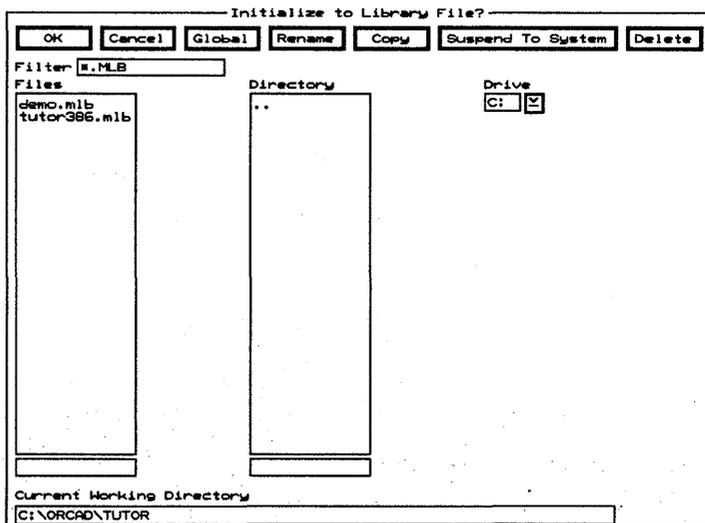


Figure 5-1. The **Initialize to Library File** dialog box.

Working with module libraries

Note that *.MLB displays in the **Filter** entry box, and the files that display in the **Files** list box all have a .MLB extension. These are libraries that contain the modules you place on a board in **Edit Layout**.

Each library contains module files organized by component type and function. The libraries are named according to the types of modules stored in them. You can rename a library, and you can import and export module files. These functions are described later in this chapter.

In this chapter you work with the DEMO library. To acquaint yourself with library editor commands, you make a copy of the DEMO library, then make a copy of a module in the DEMO library and edit it. Next, you export a copy of the edited module from the DEMO library and import it into the copied library.

Copying the DEMO library

1. Select **DEMO.MLB** from the **Files** list box. The library name highlights.
2. Select **Copy**. The **Copy File** dialog box displays.
3. Enter **DEMOCOPY.MLB** in the **Copy To:** entry box, then select **OK**. The new library is created and the **Initialize to Library File** dialog box displays with **DEMOCOPY.MLB** listed in the **Files** list box.

Copying and getting a module

It is a good idea to make a copy of a module you want to edit and edit the copy. Also, when you are creating modules that are very similar, it is easier and faster to modify a copy than to build a new module from scratch.

Follow these steps to copy 10HH100 in DEMO.MLB to another name and load the copy into the library editor:

1. Check that **DEMO.MLB** is the selected library in the **Files** list box.
2. Select **OK**. The **Get Module** dialog box displays (figure 5-2).

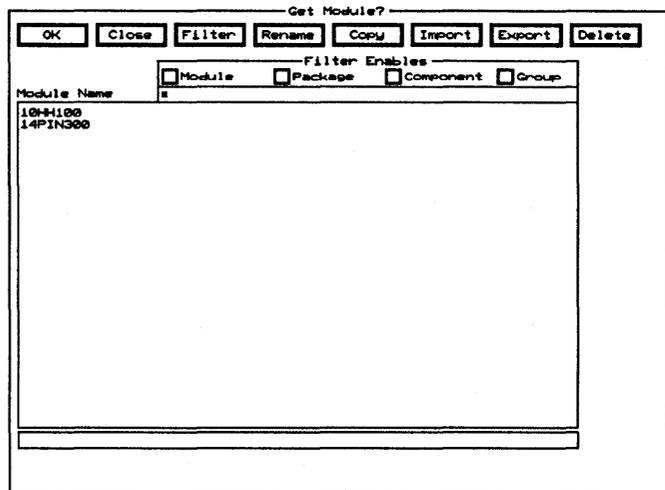


Figure 5-2. The Get Module dialog box.

3. Select 10HH100 in the Module Name list box.
4. Select Copy. The Copy Module dialog box displays (figure 5-3) and 10HH100 is highlighted.

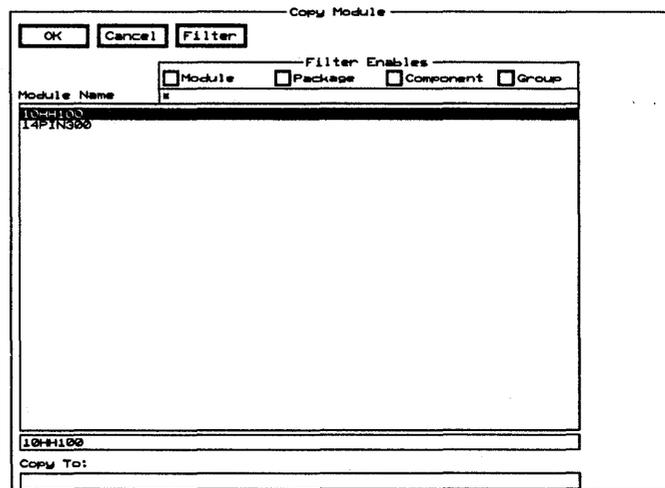


Figure 5-3. The Copy Module dialog box.

5. Enter HEADER in the Copy To: entry box, then select OK. The name displays in the Module Name list box.

6. Select **HEADER** in the **Module Name** list box, then select **OK**. The module displays in the library editor, as shown in figure 5-4.

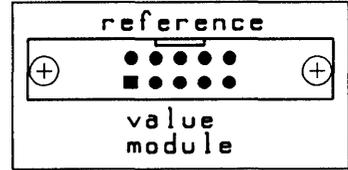


Figure 5-4. The **HEADER** module.

Note the three labels, “reference,” “value,” and “module,” that are placed by the module. These are placeholders for values that are associated with the module when it is loaded from a netlist and placed in **Edit Layout**.

The “reference” placeholder receives the reference designator that is assigned to the 10-pin header schematic part in **Draft**, such as “JP1.” The “value” placeholder receives the schematic part value, such as “10K” for a resistor. The “module” placeholder receives the module name for this part, which is “HEADER.”

Updating the library file

Making a copy of a module and loading it into the library editor loads it in memory only. The copy is not actually stored on the disk until you save the file. Follow these steps:

1. Select **QUIT**. The menu at right displays.
2. Select **Update Library File**. This saves the module **HEADER** to **DEMO.MLB**.

```
Update Library File
Write Library File
Initialize to Library
Flush Undelete Buffer
Suspend to System
Leave Library Editor
```

Renaming a module

1. Select **GO TO FUNCTION**.
The menu shown at right displays.
2. Select **Module Selection**.
The **Get Module** dialog box displays.
3. Select **10HH100** in the **Module Name** list box, then select **Rename**. The **Rename Module** dialog box displays (figure 5-5).

Pad Symbol Editor
Via Symbol Editor
Copper Tool Editor
Drill List Editor
Board Editor
Module Selection
Macro Maintenance

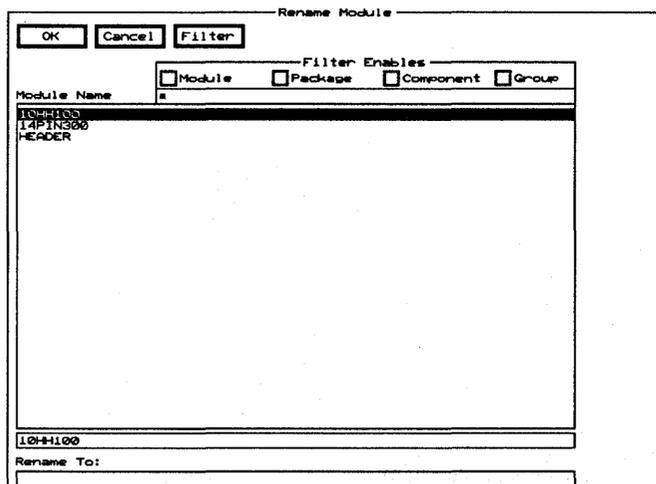


Figure 5-5. The *Rename Module* dialog box.

4. Enter **10PINCON** in the **Rename To:** entry box, then select **OK**. Module **10HH100** is renamed to **10PINCON**.

△ **NOTE:** *Renaming a module renames it in memory only. The new name is not saved in the library until you select **QUIT Update Library File**.*

5. Select **10PINCON**, then select **OK**. The module displays in the library editor.
6. Select **QUIT Update Library File**. The renamed file is saved in the library.

Displaying module information

Modules are constructed with many graphic objects, such as outline segments, pads, text, holes, and zones. You display information about these objects using the **INQUIRE**, **VERBOSE INQUIRE**, and **EDIT** commands.

INQUIRE

Follow these steps to familiarize yourself with **INQUIRE**. You will use 10PINCON, the module you just updated in the library editor, as an example.

1. Press <Enter> or click the left mouse button to display the main menu. The menu shown at right displays. The library editor main menu is very similar to the **Edit Layout** main menu.
2. Select * **LAYER**. This sets All Layers as the current working layer, so you can select any object on any layer. The layer color and name display in the lower part of the screen.
3. Place the pointer in the center of the square module pad. The object under the tip of the pointer is the one selected for inquiry.
4. Select **INQUIRE**. A description of the pad displays in the lower right part of the screen, as shown in the example below.

```
Block
Cut
Delete
Edit
Find
Go To Function
Highlight
Inquire
Jump
Track Delete
Layer
Move
Origin
Place
Quit
Route
Set
Selective
Undelete
Verbose Inquire
Window Zoom
X show RatsNest
Zoom
= bookmark
+ layer
- layer
* layer
/ other
? conditions
% macro
```

```
1 Pad: 0.0550" x 0.0550" []
```

The description tells you that it is pad number 1, and that the pad size is 0.0550" square.

5. Place the pointer on each of the pads and select **INQUIRE**. Note the pad number sequence from pad 1 to pad 10.
6. Place the pointer on one of the outline segments and select **INQUIRE**. The following description displays:

Outline Segment: 0.0080" wide

VERBOSE INQUIRE

VERBOSE INQUIRE provides information about the entire module, as well as information about the selected module object. Follow these steps to familiarize yourself with **VERBOSE INQUIRE**:

1. Place the pointer on the square pad and select **VERBOSE INQUIRE**. The same pad information provided with the **INQUIRE** command displays in the lower right part of the screen, and the **Verbose Inquire - Module** dialog box displays (figure 5-6).



Figure 5-6. The *Verbose Inquire - Module* dialog box.

This dialog box contains additional information about the module, such as the module name, number of pads, and internal routing information.

2. Select **OK** to close the dialog box.

EDIT The **EDIT** command is one of the most powerful tools in the library editor. You use **EDIT** to change properties, such as line width, size, and layer, of any module object.

EDIT is also context-sensitive. The editing options that display are determined by the type of object you select.

Procedures for editing properties are described later in this chapter. In this example, you use **EDIT** to acquire information about a module object. Follow these steps to determine what layer an outline segment is on:

1. Make sure that All Layers is the current editing layer.
2. Place the tip of the pointer on one of the outline segments that defines the rectangular shape of the module and select **EDIT**. The Edit Outline Segment dialog box displays (figure 5-7).

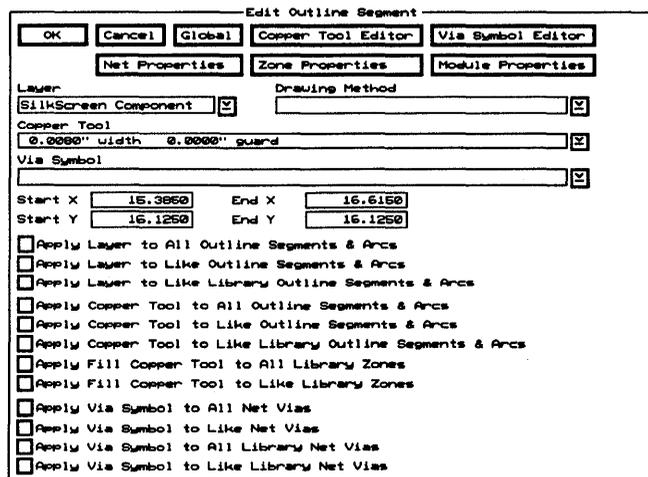


Figure 5-7 shows the 'Edit Outline Segment' dialog box. The title bar reads 'Edit Outline Segment'. At the top, there are buttons for 'OK', 'Cancel', 'Global', 'Copper Tool Editor', 'Via Symbol Editor', 'Net Properties', 'Zone Properties', and 'Module Properties'. The 'Layer' dropdown menu is set to 'Silkscreen Component'. Below it, there are fields for 'Copper Tool' (0.0080" width, 0.0080" guard) and 'Via Symbol'. The 'Start X' and 'End X' fields are 15.3850 and 16.6150, respectively. The 'Start Y' and 'End Y' fields are 16.1250 and 16.1250. There are several checkboxes for applying properties to various elements, all of which are currently unchecked.

Figure 5-7. The Edit Outline Segment dialog box.

3. Note that **Silkscreen Component** displays in the **Layer** droplist box. This is the current layer for the selected outline segment. **Silkscreen Component** is a graphic layer for the top, or component, side of the board.
4. Select **Cancel** to close the dialog box without making any changes.

Editing a module

The following procedures describe how to edit a module, using 10PINCON as an example:

Moving a module

1. Check that All Layers is the current layer. You can move a module by select any of its objects, such as the outline or labels, if All Layers is selected. If the Component Copper or Solder Copper layer is selected, you move a module by selecting one of its pads.
2. Select SET. The Global Options dialog box displays.
3. Disable Allow Edits Of Module Objects. With this option disabled, module objects (such as outlines, labels, and pads) cannot be moved individually.

Leave Stay On Grid enabled. This constrains object movement to a snap grid.

4. Select OK to close the Global Options dialog box.
5. Place the pointer in the center of a module label, as shown at right.
6. Select MOVE. The entire module moves as you move the pointer around the screen.
7. Select PLACE to place the module in a new location, or press the <Esc> key or the right mouse button to cancel the move.



module

Rotating a module to a specific angle

1. Place the pointer on a module object and select **MOVE**.
2. Select **Set**. The **Set Block Parameters** dialog box displays (figure 5-8).

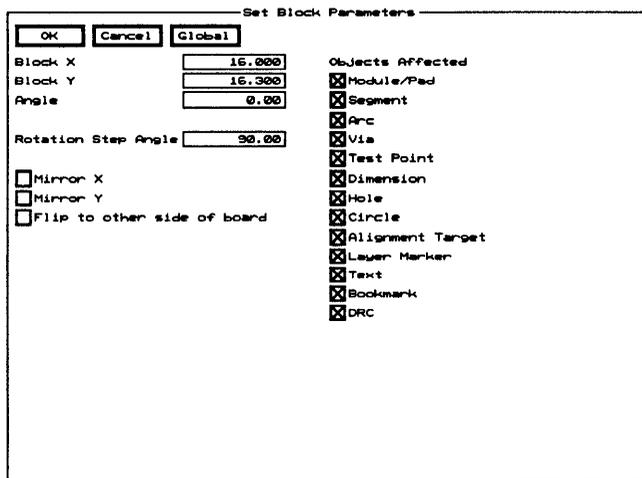


Figure 5-8. The **Set Block Parameters** dialog box.

3. Enter **90.00** in the **Angle** entry box, as shown at right.
4. Select **OK**. The module displays rotated 90° counterclockwise, as shown in figure 5-9. The module rotates using the pointer position as its pivot point.
5. Select **Place** to place the module.

Angle

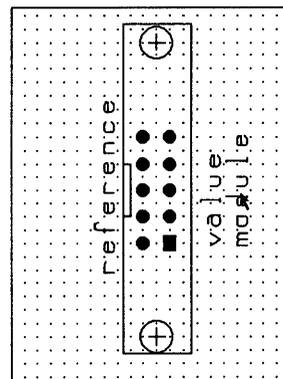


Figure 5-9. The module rotated 90° .

You can rotate a module in one-hundredth of a degree increments (0.01°), and you can rotate it either counterclockwise or clockwise.

Enter a minus sign before the rotation value to rotate the module clockwise.

If you enter an invalid rotation value in the **Angle** entry box, the value is not accepted and the following error message displays in the upper left part of the screen:

Allowed Range: -359.99 to 359.99

6. Rotate the module back to its original orientation by repeating steps 1 through 5, entering **-90.00** in the **Angle** entry box.

Rotating a module in preset steps

You can set a rotation step angle so a module rotates by a specified number of degrees clockwise or counterclockwise. Follow these steps to set a rotation step angle of 15°:

1. Place the pointer on a module object and select **MOVE**.
2. Select **Set**. The **Set Block Parameters** dialog box displays.
3. Enter **15** in the **Rotation Step Angle** entry box, as shown below, then select **OK**.

Rotation Step Angle

4. Select **> Rotate Clockwise**. The module rotates 15° clockwise. Each time you select **> Rotate Clockwise** the module rotates 15° clockwise.
5. Select **< Rotate Counter Clockwise**. The module rotates 15° counterclockwise.
6. Rotate the module back to its original orientation, then select **Place**.

Mirroring a module along the X axis

Mirroring a module makes it easy to change the pad orientation. To mirror a module along the X axis, follow these steps:

1. Place the pointer above and to the left of the module, as shown in figure 5-10.

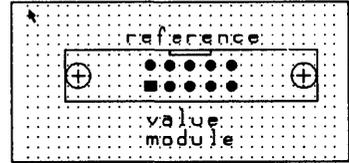


Figure 5-10. Pointer position for start of BLOCK command.

2. Select **BLOCK**.
3. Move the pointer down and to the right. A dotted rectangle stretches to follow it. Place the pointer below and to the right of all module objects, as shown in figure 5-11.

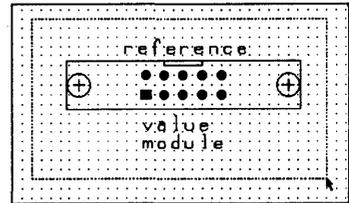
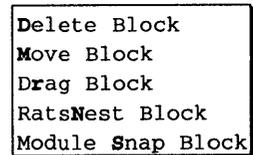


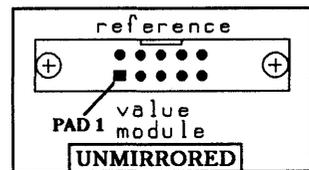
Figure 5-11. Pointer position for end of BLOCK command.

4. Select **Block End**. The menu shown at right displays.
5. Select **Move Block**.

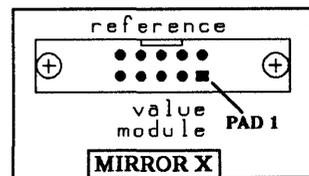


6. Select **Set**. The **Set Block Parameters** dialog box displays.
7. Enable **Mirror X**, then select **OK**.

8. Select **Place**. The module and its pads mirror along the X axis, as shown at right. Note the new position of pad 1. Use **INQUIRE** to check the new pad layout.

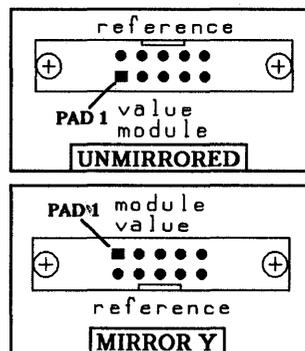


9. Repeat steps 1 through 8 to return the module to its original orientation.



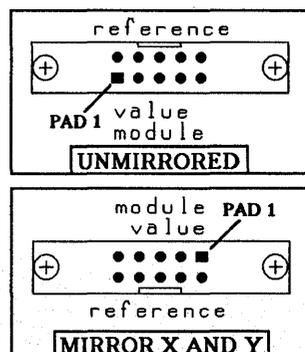
Mirroring a module along the Y axis

1. Use the **BLOCK** command to select the module, as described in the previous procedure.
2. Select **Move Block**, then select **Set**. The **Set Block Parameters** dialog box displays.
3. Enable **Mirror Y**, then select **OK**. The module mirrors along the Y axis, as shown at right.
4. Return the module to its original orientation.



Mirroring a module along both the X and Y axis

1. Use the **BLOCK** command to select the module.
2. Select **Move Block**, then select **Set**. The **Set Block Parameters** dialog box displays.
3. Enable both **Mirror X** and **Mirror Y**, then select **OK**. The module mirrors along both the X and Y axis, as shown at right.
4. Return the module to its original orientation.



Flipping a module to the other side of the board

You can create or modify a module in the library editor so it represents a component mounted on the bottom, or solder, side of the board. Follow these steps to change 10PINCON to a bottom-mounted module.

1. Make sure **Allow Edits Of Module Objects** in **Global Options** is not enabled and **Stay On Grid** is enabled.
2. Make sure that **All Layers** is the current layer.
3. Place the pointer on any module object and select **MOVE**.
4. Select **Set**. The **Set Block Parameters** dialog box displays.
5. Enable **Flip to other side of board**, then select **OK**.
6. Select **Place**. The module displays as if it is mirrored on the X axis.
7. Move the pointer to an outline segment and select **EDIT**. Note that the **Layer** droplist box displays **Silkscreen Solder** as the current layer. **Silkscreen Solder** is a graphic layer on the solder, or bottom, side of the board.

△ **NOTE:** *The label placeholders flip to the other side of the board, but display normally. When you print or plot the module you specify if you want the text printed or plotted as flipped text.*

8. Repeat the previous procedure to flip the module back to the **Silkscreen Component** side of the board.

**Moving a single
module object**

1. Select **SET**. The **Global Options** dialog box displays.
2. Enable **Allow Edits Of Module Objects**, then select **OK**. This allows you to edit individual parts of a module.
3. Check that **All Layers** is the current layer.
4. Place the pointer near the center of one of the module labels, as shown at right.
5. Select **MOVE**. The label moves with the pointer, but the rest of the module does not move.
6. Select **Place** to place the label in a new location, or press <Esc> to cancel the move and leave the label in its original position.
7. Place the pointer on a pad and select **MOVE**. The selected pad moves with the pointer.
8. Move the pad to a new location and select **Place**, or press <Esc> to cancel the move and leave the pad in its original position.

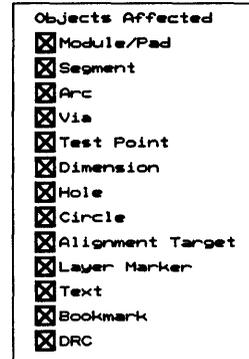


Moving selected objects within a group

Follow these steps to move the module mounting holes and all outline segments, but not move the pads or text:

1. Place the pointer above and to the left of all the module objects and select **BLOCK**.
2. Move the pointer down and to the right until all module objects are surrounded by the stretching box.

3. Select **Set**. The **Set Block Parameters** dialog box displays. Note the list of check boxes beneath **Objects Affected**, and that all check boxes are enabled.

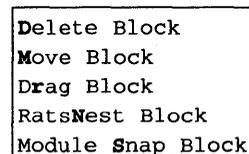


Enabled objects within the block boundaries are selected if the layer the object is on is active. Disabled objects, and enabled objects on inactive layers, are not selected.

4. Disable **Module/Pad** and **Text** in the **Objects Affected** menu, then select **OK**.

You do not need to disable any of the other objects in the list because the area defined by the block does not contain any of those objects.

5. Select **Block End**. The **BLOCK** menu at right displays.



6. Select **Move Block**. Move the pointer and note that the two holes and the outline segments move, but the disabled objects in the block (the module outline, pads, and text) do not move. Press <Esc> or click the right mouse button to cancel the **Move Block** command.

Moving an off-grid object

Some parts of a module, such as an outline segment, may be placed off-grid because the outline dimensions do not coincide with the grid setting. To move an object that is off grid, follow these steps:

1. Select **SET** to display the **Global Options** dialog box.
2. Disable **Stay On Grid**, then select **OK**.

△ *NOTE: When **Stay On Grid** is enabled, you may not be able to select an object that is off-grid if no part of the object is on an on-grid point.*

3. Make sure **All Layers** is the current layer.
4. Place the pointer on one of the module outline segments and select **MOVE**. The segment moves with the pointer.
5. Select **Place** to place the segment off-grid.

Moving an off-grid object back on grid

1. Place the pointer on the off-grid segment.
2. Select **MOVE**, then select **SET**. The **Set Block Parameters** dialog box displays.
3. Select **Global**. The **Global Options** dialog box displays.
4. Enable **Stay On Grid**, then select **OK**.
5. Select **OK** or **Cancel** to close the **Set Block Parameters** dialog box. Notice that the outline segment movement is constrained to the snap grid.
6. Select **Place** to place the segment on the grid.

△ *NOTE: You cannot place an off-grid pad back on grid using these procedures.*

Deleting and undeleting module objects

These procedures acquaint you with the various methods used to delete and undelete module objects.



NOTE: The module placeholders "reference," "module," and "value" cannot be deleted. These items are fields that receive values from the netlist when the module is placed in Edit Layout.

Deleting objects on any layer

Follow these steps to set up the library editor so you can delete any module object on any layer.

1. Make sure **All Layers** is the current layer.
2. Select **SET** to display the **Global Options** dialog box.
3. Enable **Allow Edits Of Module Objects**, then select **OK**.
4. Place the pointer on the square pad and select **DELETE**. The square pad is deleted.
5. Place the pointer on the top outline segment and select **DELETE**, then place the pointer on the left mounting hole and select **DELETE**.
6. Select **ZOOM Refresh** to redraw the display.

Deleting objects on a specific layer

If a module has overlapping objects on different layers, you specify only the layer containing the object you want to delete so you do not delete the wrong object.

1. Select **/ OTHER** to specify the Component Copper layer as the current layer. The **/OTHER** command toggles between the copper layer pairs that are currently set, which are the Component Copper and Solder Copper layers.
2. Place the pointer on one of the round pads and select **DELETE**. The pad is deleted.
3. Place the pointer on an outline segment and select **DELETE**. The message "Nothing to delete" displays in the lower right corner of the display.

The outline segment is not deleted because it is on the Silkscreen Component layer, which is not selected.

4. Select **LAYER**. The Layer dialog box displays.
5. Select **Silkscreen Component** from the **Current Layer** menu, then select **OK**. The Silkscreen Component layer becomes the current layer.
6. Place the pointer on the same outline segment and select **DELETE**. The segment is deleted.

UNDELETE

After you delete an object, it is stored in an undelete buffer. You can restore the object from the undelete buffer by selecting the **UNDELETE** command. Follow these steps:

1. Select **UNDELETE**. The last deleted object reappears.
2. Continue selecting **UNDELETE**. All deleted objects are recovered in the reverse order they were deleted. When all objects are undeleted, the message "Nothing to Undelete" appears in the lower right part of the screen

Up to 254 levels of undelete can be performed. This means that all objects deleted during the last 254 delete commands in the same editing session can be undeleted.



***NOTE:** The contents of the undelete buffer are not saved when you save the board file.*

SELECTIVE

You use **SELECTIVE** to select which objects you want to undelete from the undelete buffer.

1. Select * **LAYER** to set **All Layers** as the current layer.
2. Delete the square pad, the top outline segment, and the left mounting hole.
3. Select **SELECTIVE**. Notice that the deleted objects reappear in their original display colors, but the other module objects now display in dark gray. This provides visual contrast between deleted objects and the other objects on the screen, which cannot be selected while in this mode.

4. Place the pointer on the square pad and select **Undelete**. The color of the square pad changes to dark gray, indicating that it is removed from the undelete buffer.
5. Place the pointer on the outline segment and select **Undelete**.
6. Select **Quit Selective Undelete**. The module objects display with their normal layer colors and the previously deleted objects are restored. Note that the left mounting hole does not display because it is still in the undelete buffer.

**Permanently deleting
deleted objects**

As you build and revise a module, you may want to permanently delete old objects from the undelete buffer. Follow these steps to permanently delete the mounting hole:

1. Choose **SELECTIVE**. The left mounting hole in the undelete buffer displays in its normal color, while the rest of the module displays in dark gray.
2. Place the pointer on the mounting hole and select **Permanently Delete**. This deletes the mounting hole from the undelete buffer and the display. The mounting hole cannot be restored.
3. Select **Quit Selective Undelete** to return to the library editor.



NOTE: These changes are stored in memory only, until you save the file.

4. Select **QUIT Update Library File**. This saves the revised 10PINCON module to the DEMO library.

Exporting and importing modules

Exporting

As you create, revise, and organize modules in a library you may need to transfer them from one library to another. You transfer a module to another library by exporting it to a file, then importing the file into the destination library.

Follow these steps to export 10PINCON from the DEMO library to a file, and import the file into the DEMOCOPY library you created earlier in this chapter.

1. Select **GO TO FUNCTION Module Selection**. The **Get Module** dialog box displays and 10PINCON is selected in the **Module Name** list box.
2. Select **Export**. The **Export Module to File** dialog box displays (figure 5-12).

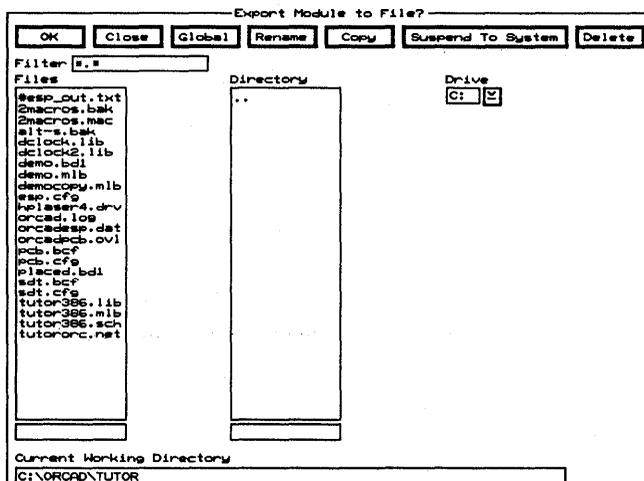


Figure 5-12. The **Export Module to File** dialog box.

3. Enter **10PINCON.EXP** in the entry box below the **Files** list box. The **.EXP** extension is not required, but it is helpful when you want to identify or find the file.
4. Select **OK**. **Edit Layout** writes the file in the directory specified in the **Current Working Directory** entry box. The **Get Module** dialog box displays and 10PINCON is selected.
5. Select **OK**. The 10PINCON module displays in the library editor.

- Importing**
1. Select **QUIT Initialize to Library**. The **Initialize to Library File** dialog box displays.
 2. Select **DEMOCOPY.MLB** from the **Files** list box and select **OK**. The **Get Module** dialog box displays, and the modules in **DEMOCOPY.MLB** display in the **Module Name** list box.
 3. Enter **NEWITEM** in the entry box below the **Module Name** list box.
 4. Select **Import**. The **Import Module from File** dialog box displays (figure 5-13).

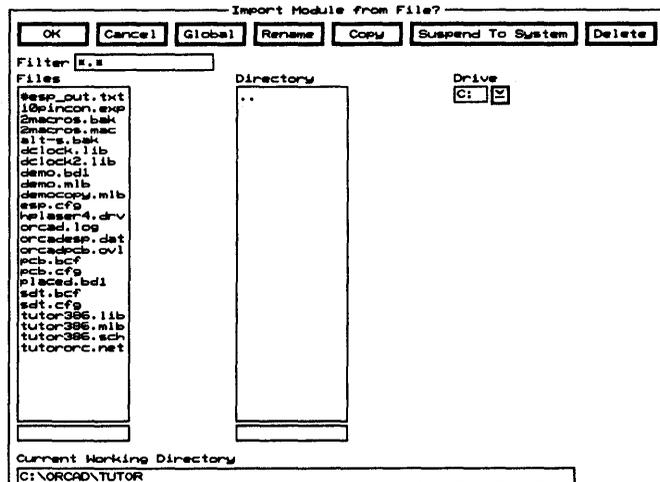


Figure 5-13. The **Import Module from File** dialog box.

5. Enter ***.EXP** in the **Filter** entry box. Only files with a **.EXP** extension display in the **Files** list box.
6. Select **10PINCON.EXP** in the **Files** list box, then select **OK**. The exported module is imported into **DEMOCOPY.MLB** and receives the name **NEWITEM**. The **Get Module** dialog box displays and **NEWITEM** is selected.

△ **NOTE:** *NEWITEM* is stored in memory only. You must select **QUIT Update Library File** to save *NEWITEM* in *DEMOCOPY.MLB*.

7. Select **OK**. *NEWITEM* displays in the library editor.
8. Select **QUIT Update Library File** to save the module.
9. Verify that *NEWITEM* is in *DEMOCOPY.MLB*. Select **GO TO FUNCTION Module Selection**. The **Get Module** dialog box displays and *NEWITEM* is selected in the **Module Name** list box.
10. Select **OK**. The module displays in the library editor.

Deleting exported modules

After you import a module into a library and save it, you can delete the exported module file. Follow these steps to delete *10PINCON.EXP*:

1. Select **QUIT Initialize to Library**. The **Initialize to Library File** dialog box displays.
2. Enter ***.EXP** in the **Filter** entry box. Only files with a **.EXP** extension display in the **Files** list box.
3. Select *10PINCON.EXP* in the **Files** list box and select **Delete**. The file is deleted from the disk and removed from the list box.
4. Select **Close** to return to the library editor.

Creating a module

As your design needs increase, you may require a module for your board layout that is not available in any of the supplied OrCAD module libraries. You use the library editor in **Edit Layout** to automatically create complex pad arrays and edit all objects in a module.

In this section you learn how to design a module by creating 7PINDEMO, a seven pin connector with two mounting holes. When you are done with the exercises, the finished module looks like figure 5-14.

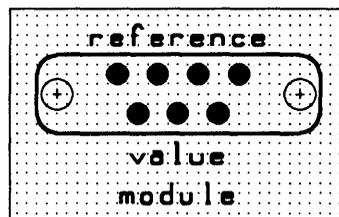


Figure 5-14. The finished 7PINDEMO module.

Starting a new module

1. Select **QUIT Initialize to Library**. The **Initialize to Library File** dialog box displays.
2. Enter *.MLB in the **Filter** entry box. A list of module libraries displays in the **Files** list box.
3. Select DEMO.MLB from the list and select **OK**. The **Get Module** dialog box displays and the modules in DEMO.MLB display in the **Module Name** list box.
4. Enter 7PINDEMO in the entry box below the **Module Name** list box and select **OK**.

This assigns the filename 7PINDEMO to the current editing session. The library editor displays, containing the three text strings shown in figure 5-15.

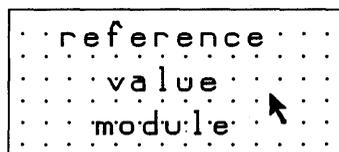


Figure 5-15. Module property placeholders.

These three text strings are module properties associated with every module, and they automatically display in the library editor when you create a new module. These properties are placeholders for values that are associated with the module when it is loaded from a netlist and placed in **Edit Layout**.

The "reference" placeholder receives the reference designator that is assigned to the schematic symbol for the module in **Draft**. The "value" placeholder receives the schematic part value, such as "10K" for a resistor. The "module" placeholder receives the module filename, which is 7PINDEMO.

Setting preferences

Before you begin laying out your new module, note some preset library editor preferences. Preferences are saved with the module when you save the file.

1. Select **SET**. The **Global Options** dialog box displays. The following options are automatically enabled whenever you start a new module:
 - ❖ **Outline tracks**. This displays segments as outlined objects, which makes them easier to edit.
 - ❖ **Stay On Grid**. You should always use a grid to lay out a module. Manually routing and autorouting tracks between module pads in **Edit Layout** is more efficient when module pads are placed on a grid.
 - ❖ **Crosshair Cursor**. This displays a pointer with a full-screen crosshair, which makes it easier to align objects.
 - ❖ **Allow Edits of Module Objects**. When this is enabled you can select individual module objects.
 - ❖ **Grid Size** is set at 0.025000" (25 mils).
2. Select a different grid dot color in **Dots** if you want to make the grid dots more visible. Select black to make the grid dots invisible.
3. Select **Layer**. The **Layer** dialog box displays. Note that **SilkScreen Component** is selected as the current layer.
4. To display the Silkscreen Component layer in another color, select **Other Colors/Enables/...** to display the **Other Colors/Enables/...** dialog box.
5. Select another display color in **SilkScreen Component**, then select **OK** to return to the **Layer** dialog box.
6. Select **OK** to close the **Layer** dialog box, then select **OK** to close **Global Options**.

You place non-copper module graphic objects, like its outline, on the SilkScreen Component layer when you want these graphic objects silkscreened to the Component Copper side of the board.

Changing the view of the display

1. Move the pointer to (0.2250", 0.0250") and select **WINDOW ZOOM**.
2. Move the pointer and stretch the zoom window to (1.1500", 0.6500"), then select **Window Zoom End**. The display zooms to the selected area.

Drawing methods

This tutorial describes two methods for drawing the module outline. You can perform either one of these procedures, or both if you wish:

- ❖ Drawing the outline with 90 degree corners, then placing four arcs and modifying the rectangular outline.
- ❖ Drawing the outline with orthogonal arc corners.

Use the arrow keys to move the pointer for these procedures—it makes it easier to follow the coordinates.

Drawing the outline with 90 degree corners

1. Move the pointer to (0.4000", 0.4000") and select **ORIGIN**. Setting the origin where you begin the module outline makes it easier to see that it is drawn to scale.
2. From (0.0000", 0.0000"), select **PLACE Outline Begin**.
3. Move the pointer to the right to (0.7000", 0.0000"). The outline segment stretches with the pointer.
4. Select **Begin**, then draw the segment down to (0.7000", 0.2000").
5. Select **Begin**, then draw the segment to the left to (0.0000", 0.2000").
6. Select **Begin**, then draw the segment up to (0.0000", 0.0000").
7. Select **End** to complete the rectangular outline. Press <Esc> to dismiss the **PLACE** menu. The outline looks like figure 5-16.

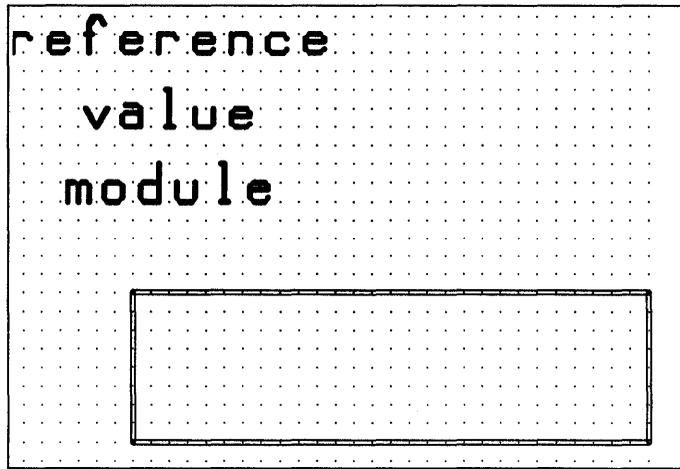


Figure 5-16. The module outline.

Adding arcs

The four corners of this module outline need to be changed to a 0.050" radius. Follow these steps to add a circle constructed of four separate arcs:

1. Move the pointer to (0.3000", 0.1000").
2. Select **PLACE Circle**.
An X-shaped cursor displays in the center of the pointer (figure 5-17). The 'X' represents the center of the circle.

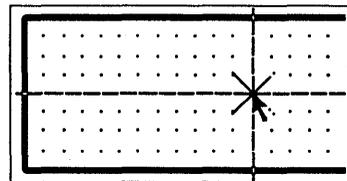


Figure 5-17. The **PLACE Circle** cursor.

3. Select **Set**. The **Edit Circle** dialog box displays (figure 5-18).

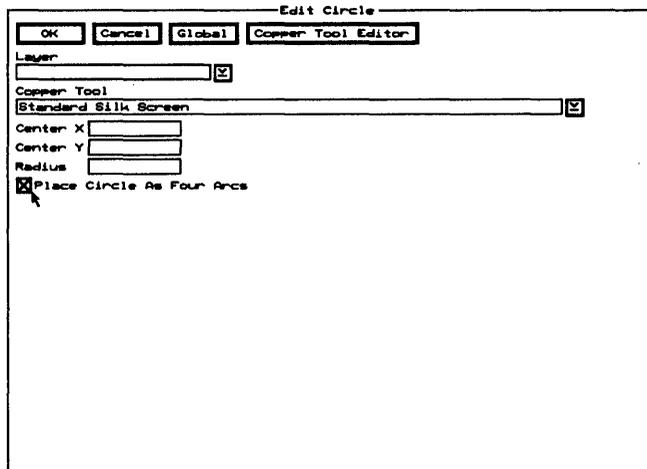


Figure 5-18. The *Edit Circle* dialog box.

4. Enable **Place Circle As Four Arcs**, then select **OK**. Circles created with this option enabled are built with four 90° arc segments, which can be individually moved.
5. Select **Begin**. Move the pointer to the right to (0.3500", 0.1000"). The circle enlarges as the pointer moves.
6. Select **End** to complete the circle. Notice that another 'X' cursor appears in the center of the pointer, indicating you can place another circle.
7. Press <Esc> twice to dismiss the **PLACE Circle** menu. The circle looks like figure 5-19.

When objects display with outlined segments it is easier to see the end points of each segment. The four small circles inside the outlined circle are the overlapping end points for the four arcs.

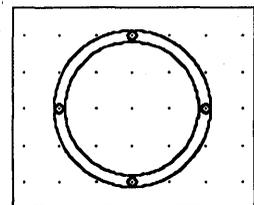


Figure 5-19. Finished circle, showing arc endpoints.

Positioning the arcs

1. Place the pointer at (0.3000", 0.0500") and select **MOVE**. The upper left arc moves with the pointer (figure 5-20).
2. Move the arc to (0.0500", 0.0000"). The endpoints of the arc align with the top and left outline segments (figure 5-21).
3. Select **Place** to position the arc.
4. Use the same procedures to move the three remaining arcs to their respective outline corners, aligning the endpoints with the outline segments. The module looks like figure 5-22.

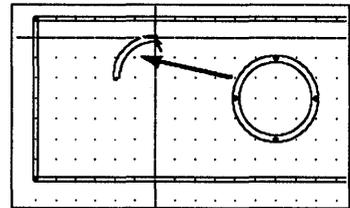


Figure 5-20. Moving the upper left arc.

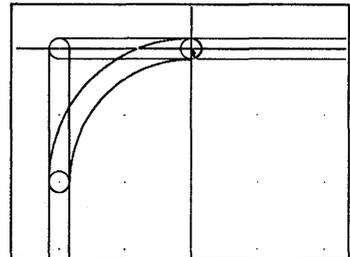


Figure 5-21. Magnified view of arc placement.

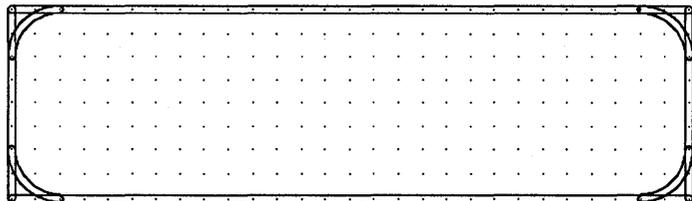


Figure 5-22. The module outline with an arc in each corner.

Deleting the 90° corners

Now that the arcs are in each corner, you need to delete the 90° corners from the outline. Perform these steps:

1. Move the pointer to (0.0000", 0.0500"). This places the pointer on the lower endpoint of the upper left arc.
2. Select **CUT**. This cuts the left straight outline segment into two segments, and "Cut" displays at the bottom of the screen.
3. Move the pointer to (0.0500", 0.0000"), the other endpoint for the arc, and select **CUT**. The top outline segment is cut into two segments at the pointer location.

4. Move the pointer to (0.0000", 0.0000"). The pointer is on the intersection of the left and top cut outline segments (figure 5-23).

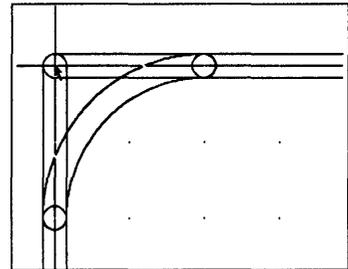


Figure 5-23. Pointer position to delete cut segments.

5. Select **DELETE**. One of the cut outline segments is deleted.
6. Select **DELETE** again to delete the other cut outline segment. You delete a segment by placing the pointer anywhere along its length and selecting **DELETE**.



NOTE: If you have objects intersecting at one point but you do not want to delete all of them, do not try to delete an object at the intersection—you will probably delete the wrong one. If you delete the wrong object, select **UNDELETE** to recover it.

Now the upper left corner of the outline looks like figure 5-24.

7. Cut and delete the 90° outline segments from the three remaining corners, using the procedures previously described.

After making these revisions, the completed module outline looks like figure 5-25.

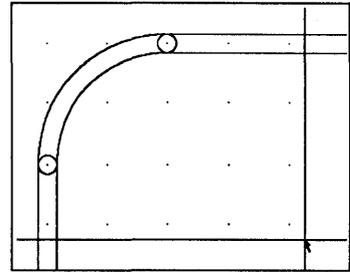


Figure 5-24. Enlarged view of corner after deleting 90° segments.

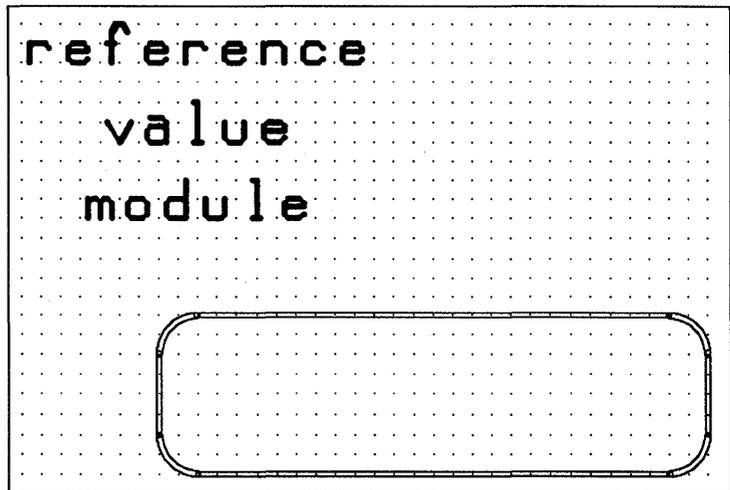


Figure 5-25. The completed module outline.

**Drawing the outline
with arc corners**

1. If you already drew the outline using the previous procedures, delete the outline and start with step 3 (you do not need to set the origin again). If you skipped to this section to draw the outline, start with step 2.
2. Move the pointer to (0.4000", 0.4000") and select **ORIGIN**. Setting the origin where you begin the module outline makes it easier to see that it is drawn to scale.
3. Move the pointer to (0.0500", 0.0000") and select **PLACE Outline**.
4. Select **Set. The Edit Outline Segment** dialog box displays.
5. Select the **Drawing Method** droplist button to display the selections in the droplist box.
6. Select **Draw Orthogonal Arc Corners**, then select **OK**.
7. Select **Begin** and draw the segment to the right, then down to (0.7000", 0.0500"). The segment automatically draws an arc as you move the pointer down.
8. Select **Begin**, then draw the segment down to (0.7000", 0.1500"). The outline looks like figure 5-26.

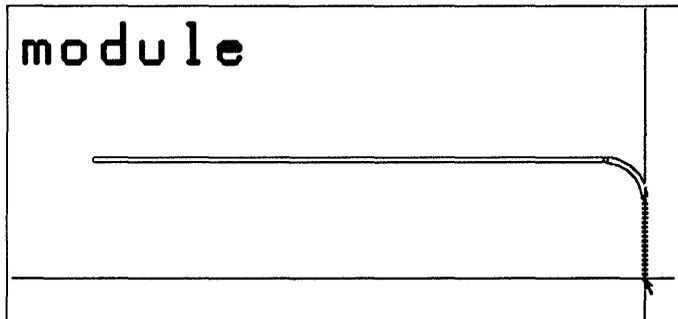


Figure 5-26. Drawing the outline with arc corners.

9. Select **Begin** and move the pointer down, then to the left to (0.6500", 0.2000") to draw the second arc segment.

10. Select **Begin** and move the pointer to the right to (0.0500", 0.2000").
11. Select **Begin** and move the pointer left, then up to (0.0000", 0.1500") to complete the third arc segment.
12. Select **Begin** and move the pointer up to (0.0000", 0.0500").
13. Select **Begin** and move the pointer up, then to the right to (0.0500", 0.0000").
14. Select **End** to complete the module outline. Press <Esc> to dismiss the **PLACE** menu. The outline looks like figure 5-25.

Adding holes Follow these steps to add the two mounting holes:

1. Select **PLACE Hole** to display a hole.
2. Select **Set** to display the **Edit Hole** dialog box (figure 5-27).

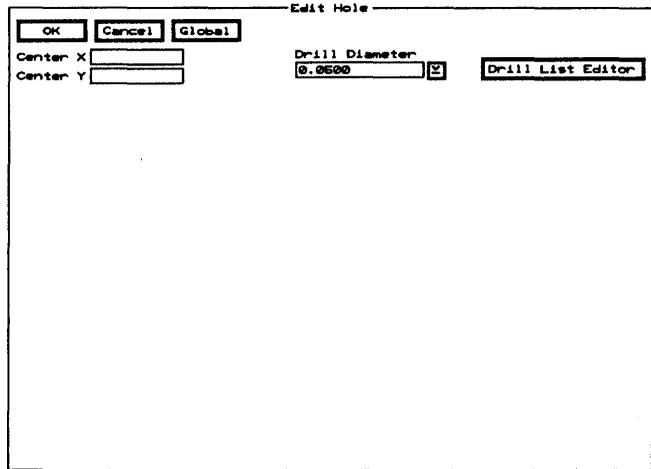
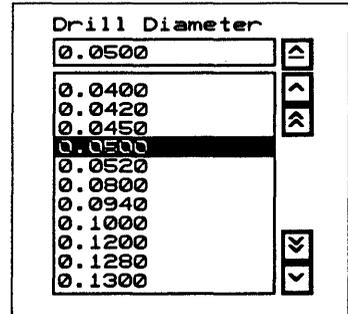


Figure 5-27. The *Edit Hole* dialog box.

3. Select the **Drill Diameter** droplist button to display its droplist box, as shown at right.
4. Select **0.0800**. The droplist box closes and **0.0800** displays as the new drill diameter.
5. Select **OK**. The hole is now larger.
6. Move the pointer to (0.0500", 0.1000") and select **Place**. Another hole displays.



7. Move the second hole to (0.6500", 0.1000") and select **Place**.
8. Press <Esc> twice to dismiss the **PLACE** menu. The module looks like figure 5-28.

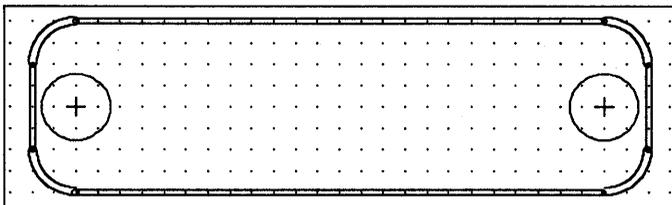
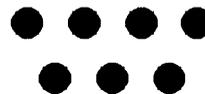


Figure 5-28. The module, with mounting holes added.

Placing the pad array

The library editor has a powerful pad array generator that makes the task of laying out complex sets of module pads an easy task.

The module needs seven pads placed in two rows, with the three pads in the bottom row offset so they are centered beneath the pads in the top row. See the illustration at right. Follow these



steps to create this seven pin pad array:

1. Select **PLACE Pad** to display a pad.
2. Select **Set**. The **Edit Pad** dialog box displays (figure 5-29).

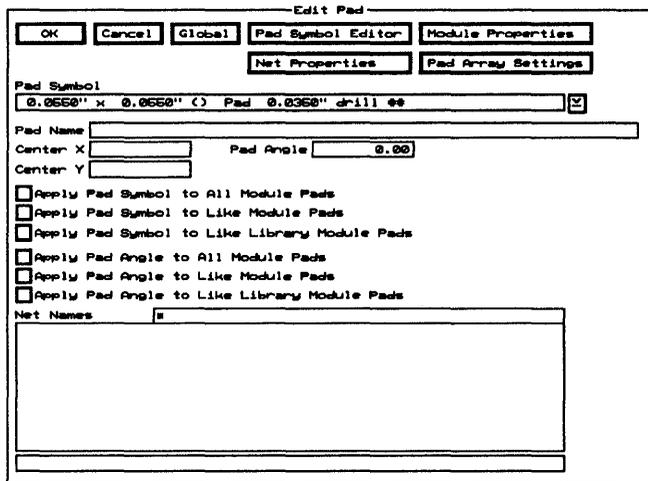


Figure 5-29. The Edit Pad dialog box.

Note the values in the Pad Symbol droplist box. The values describe the size, shape, and drill hole size of the selected pad. Refer to figure 5-30 for more information on these pad symbol values.

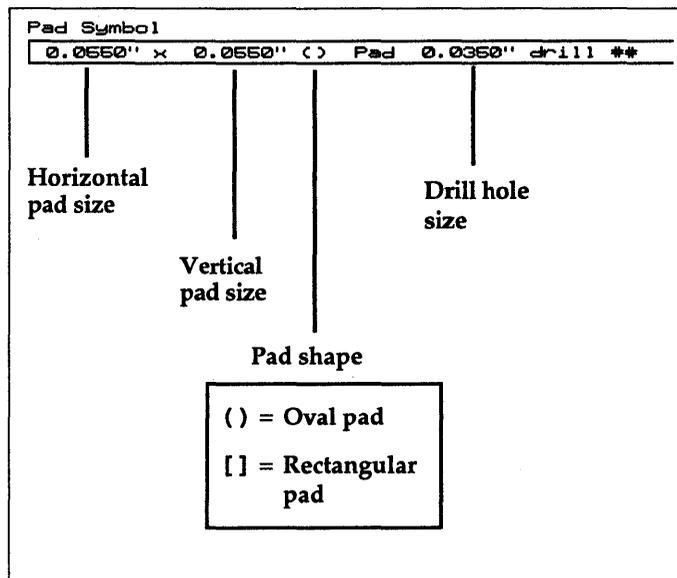


Figure 5-30. Description of pad symbol values.

*Selecting a new pad
symbol*

The pads for 7PINDEMO require a 0.0450" drill hole, but the current pad symbol has a 0.0350" drill hole. Follow these steps to select a new pad symbol:

1. Select the **Pad Symbol** droplist button to display its droplist box.
2. Select **0.0550" x 0.0550" () Pad 0.0450" drill ##**. The new pad displays in the **Pad Symbol** entry box.

Designing the pad array

1. Select Pad Array Settings. The Edit Pad Array Settings dialog box displays (figure 5-31).

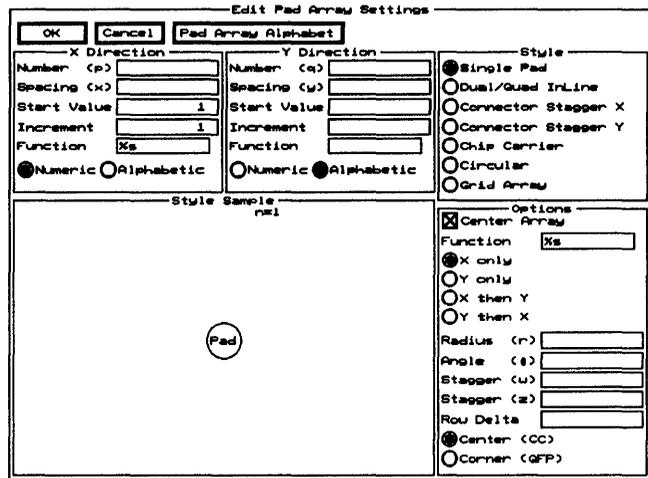


Figure 5-31. The Edit Pad Array Settings dialog box.

You choose the type of pad array you want from the selections in the **Style** menu. Note that **Single Pad** is selected because you have only one pad placed in the library editor.

Style Sample displays an example of the pad array chosen in **Style**, and provides visual aids for configuring entry boxes in the menus. A single pad displays in **Style Sample** because **Single Pad** is selected in the **Style** menu.

Select the other options in the **Style** menu and see how the sample layout changes in **Style Sample**.

2. Select **Connector Stagger X**. The layout in **Style Sample** displays what you want—you just need to define the number of rows, the number of pads in each row, and the pad spacing.

3. The number of pads in the X direction is 4, so enter **4** in the **Number (p)** entry box in **X Direction**.
4. The number of pads in the Y direction is 2, so enter **2** in the **Number (q)** entry box in **Y Direction**.
5. The spacing between the pads should be one tenth of an inch (0.1000 inch) in both the X and Y direction, so leave **Spacing (x)** in **X Direction** and **Spacing (y)** in **Y Direction** at their default values.
6. The first pad in the array should be pad number 1, and the other pad numbers should increment by 1. Leave **Start Value** and **Increment in X Direction** set at their default values.
7. Leave **Numeric** as the default selection in **X Direction**. This assigns a numeric value to the pads. **Alphabetic** assigns alphabetic values to the pads, which increment according to the value in **Increment**.
8. Leave **Center Array** enabled in **Options**. This places the pointer in the center of the pad array, which makes it easier to align the pad array with other objects.
9. Enter **0.0500** in **Stagger (w)**. This staggers the bottom row of pads so they are offset 50 mils from the pads in the top row. Refer to the sample displayed in **Style Sample**.
10. The pad array needs seven pads, but there are eight pads currently defined (two rows of four pads). Enter **-1** in **Row Delta**. This deletes the last pad (pad 8) from the bottom row of the pad array.
11. Select **OK** to close the **Edit Pad Array Settings** dialog box, then select **OK** to close the **Edit Pad** dialog box.
12. The pad array displays in the library editor. Move the pad array to (0.3500", 0.1000"), the center of the module.
13. Select **Place**. Another pad array displays. Press <Esc> or click the right mouse button twice to dismiss the additional pad array and the **PLACE** menu.

Now the module looks like figure 5-32.

You are almost finished designing 7PINDEMO. All that is left to do is position the three module value placeholders.

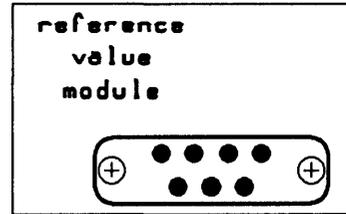


Figure 5-32. The pad array.

Positioning the placeholders

It is recommended that you position the module value placeholders close to the module outline. If you place modules close together on a board, it is easier to select the correct module value if you keep the placeholders close to the module outline.

Follow these steps to position the module placeholders:

1. Make sure Silkscreen Component is the current layer.
2. Place the pointer in the center of "value" and select MOVE.
3. Move the placeholder to the position shown in figure 5-32 and select Place.
4. Perform the same steps for "module" and "reference," placing them as shown in figure 5-33.

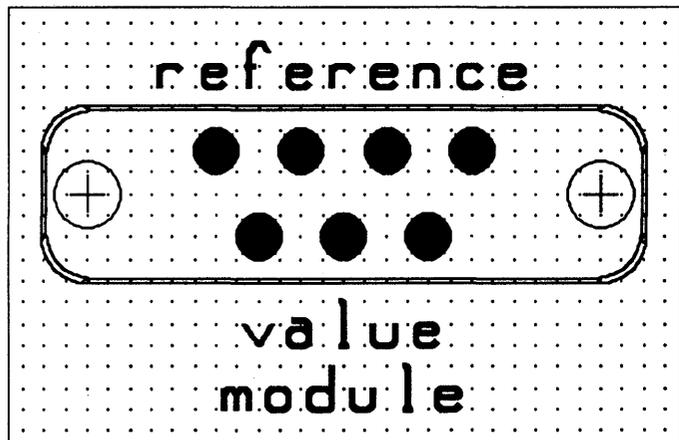


Figure 5-33. The positioned module placeholders.

Saving the module

You are now ready to save the finished module. Select **QUIT Update Library File** to save 7PINDEMO in the DEMO library.

Leaving the library editor

Select **QUIT Leave Library Editor** to return to the board editor in **Edit Layout**. The module you just designed remains in the library editor while you work in **Edit Layout**, even if you load another board file.



*NOTE: You can also return to the **Edit Layout** board editor from the library editor by selecting **GO TO FUNCTION**, then selecting **Board Editor** from the menu.*

Summary

In this chapter you learned how to load, edit, and create a module using the library editor.

The next chapter describes how to begin a board layout by loading a netlist and placing modules on the TUTOR board.



Placing the TUTOR board

About layout placement

The most critical part of the PC board design process is placing the modules on the board. During module placement, some of the important considerations are:

- ❖ Placing modules so tracks between pads on a net are as short as possible
- ❖ Placing modules so their physical size and location on the board does not interfere with other PC boards or components in the assembled instrument
- ❖ Isolating modules that create electrical interference or generate excessive heat

This chapter introduces you to the processes used to create the TUTOR board layout. In this chapter you learn how to:

- ❖ Load the **Edit Layout** template file
- ❖ Draw a board outline
- ❖ Load a netlist
- ❖ Display **Edit Layout** placement aids
- ❖ Place modules
- ❖ Edit module placement
- ❖ Place other board objects

If you want to do this part of the tutorial at another time and proceed to *Chapter 7: Routing the TUTOR board*, select **QUIT Initialize Board File**, then select **PLACED.BD1** and proceed to chapter 8.

PLACED.BD1 represents how the TUTOR board appears when you complete this chapter.

Loading the Edit Layout template

To begin this part of the tutorial you load the **Edit Layout** template, which is an empty file with predefined configurations.

Follow these steps to load the **Edit Layout** template file:

1. From **Edit Layout**, select **QUIT Initialize Board File**. The **Initialize Board File** dialog box displays.
2. Delete the filename from the entry box beneath the **Files** list box, then press <Enter>.
3. Select **OK**. The template loads into **Edit Layout**.

Setting options

Follow these steps to set some options for **Edit Layout**:

1. Select **SET** to display the **Global Options** dialog box.
2. Enable **Outline Pads**.
3. Enter **0.025000** in the **Grid Size** entry box to set a 25 mil grid.
4. Select **OK**.
5. Select **ZOOM Set Scale**. The **Set Zoom Scale** dialog box displays.
6. Enter **14** in the **Scale** entry box, then select **OK**. The new zoom scale displays at the bottom of the screen, next to the pointer coordinates.

Drawing the board outline

You define the shape and size of the board by drawing an outline. Follow these steps to draw the TUTOR board outline:

△ *NOTE: Use a combination of the <Ctrl> key and the arrow keys to draw the board outline. The <Ctrl> key causes the pointer to move five grid spaces at a time. The arrow keys make it easier to follow the coordinates. You must release the <Ctrl> key before you select **Begin**.*

1. Select * **LAYER** to set **All Layers** as the current layer. This places the outline on all board layers.
2. Use the arrow keys to move the pointer to (0.7500", 0.2500").
3. Select **PLACE Outline Begin**.
4. Hold down the <Ctrl> and right arrow keys to draw the first outline segment to (7.1250", 0.2500").
5. Select **Begin**, then use the down arrow key to draw a segment to (7.1250", 3.3750").
6. Select **Begin**, then use the left arrow key to draw a segment to (0.7500", 3.3750").
7. Select **Begin**, then use the up arrow key to draw a segment to (0.7500", 0.2500").
8. Select **End** to complete the board outline.

Tip

You can draw two sides of a rectangular board outline at the same time. Select **PLACE Outline Begin** where you want the upper left corner of the outline, then move the pointer down and to the right—the outline segment automatically bends to follow the pointer.

Select **New** where you want the lower right corner of the board outline, then select **Begin** to start a new segment. Move the pointer up and to the left to the beginning of the board outline—the outline segment automatically bends to follow the pointer.

Select **End** to complete the board outline.

Loading the netlist

Now that you completed the TUTOR board outline, you are ready to load the netlist.



NOTE: You must have TUTOR386.MLB as the configured module library. If you did not configure TUTOR386.MLB in the *Configuring PC Board Layout Tools* section of chapter 5, save this file as TUTOR.BD1, quit Edit Layout, and configure TUTOR.MLB using the instructions in chapter 5.

When a netlist is loaded, the modules assigned to the parts on the schematic are loaded into the board layout and nets are assigned to the appropriate pads on the modules. A net is a common signal name shared by two or more module pads on a board.

Before you load the netlist, you define a block to load the modules into and specify the netlist filename.

Defining a netlist block and selecting TUTOR386.NET

1. Select **GO TO FUNCTION**. The menu at right displays.
2. Select **Netlist Loader**.
3. Move the pointer to (0.7500", 3.3750"), the lower left corner of the board outline.
4. Select **Block Begin**, then move the pointer to (8.5000", 6.2500"). A bounding box stretches to follow the pointer. The modules loaded from the netlist are placed inside the block area.
5. Select **Block End**. The Load Netlist File dialog box displays (figure 6-1).

Pad Symbol Editor
Via Symbol Editor
Copper Tool Editor
Drill List Editor
Net Property Editor
Library Editor
Autorouter
Netlist Loader
Printing and Plotting
Macro Maintenance

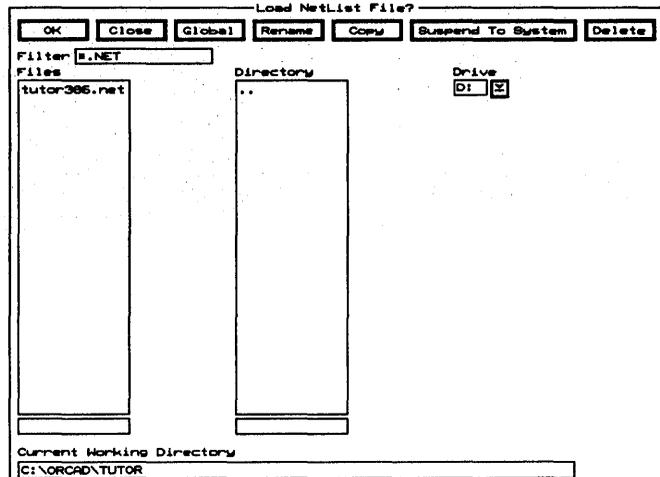


Figure 6-1. The Load Netlist File dialog box.

6. Select TUTOR386.NET from the Files list box, then select OK. The Netlist Load Options dialog box (figure 6-2) displays.

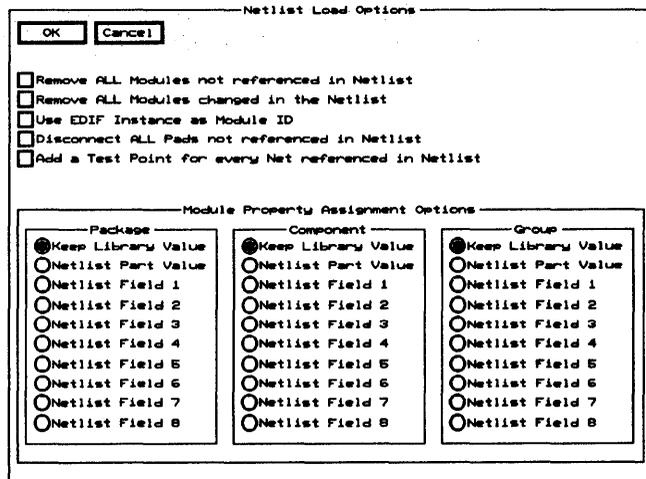


Figure 6-2. The Netlist Load Options dialog box.

7. Select OK to accept the default configurations in the Netlist Load Options dialog box. The netlist loads and modules display in the area defined by the block.

The netlist loader spaces the modules evenly inside the block, and uses the size of the block to calculate the amount of space between the modules. If the defined block is small, some of the modules may overlap. Making a large netlist block minimizes the number of overlapping modules.

8. Pan the display, if necessary, until the board outline and all modules display.

Figure 6-3 shows how your board layout appears after loading TUTOR386.NET.

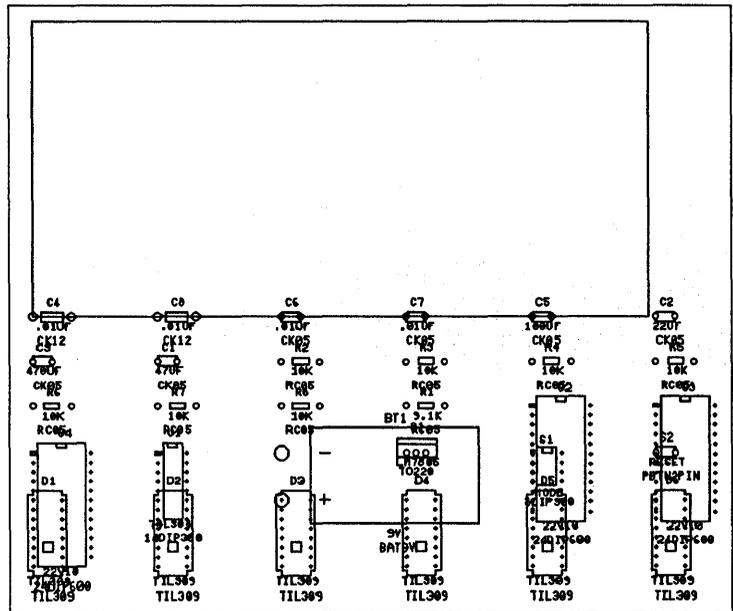


Figure 6-3. The TUTOR board, after loading TUTOR386.NET.

About module placement

There are thirty modules to place on the TUTOR board: eight capacitors, eight resistors, six display chips, four integrated circuits, two switches, one transistor, and one battery.

When you place modules on a board, you must consider the electrical relationships among the modules. Ideally, all modules sharing common nets are placed together so tracks between the modules are as short as possible. Complex circuits make this difficult to accomplish, so you usually achieve optimum placement by using visual placement aids. Placement aids display circuit connectivity, which describes the electrical paths that the nets make through the modules on the board.

Placement aids

Edit Layout provides two visual aids to help you determine optimum module placement:

- ❖ Ratsnest
- ❖ Force vector

Ratsnest

A ratsnest is a straight line visual connection between two or more pads in a layout that are electrically, but not physically, connected.

Force vector

A force vector is a single vector representing the mathematical sum of all the ratsnest vectors for a module. The length of the vector indicates the length of the routes and how close to optimum a module's position is on the board. The goal is to place the module so the vector is as short as possible.

Displaying the ratsnest

1. Place the pointer above and to the left of the modules, then select **BLOCK**.
2. Enclose all the modules in a block by moving the pointer below and to the right of the group and selecting **Block End**. The menu at right displays.
3. Select **RatsNest Block**. All modules in the block display their ratsnest connections (figure 6-4).

Delete Block
Move Block
Drag Block
RatsNest Block
Module Snap Block

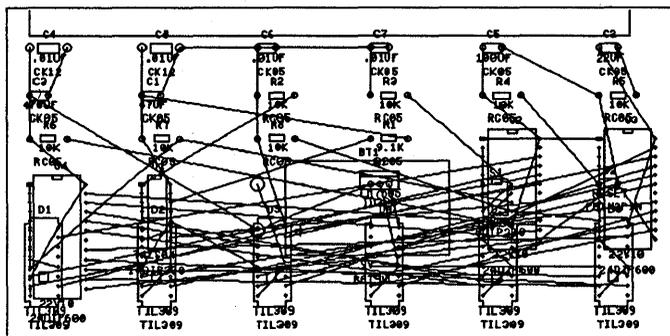


Figure 6-4. Ratsnest of the TUTOR layout.

Turning off the ratsnest

1. Position the pointer so it is not on any of the modules.
2. Select **X SHOW RATSNEST**. The ratsnest is removed from the display and the message "Show RatsNest Cleared" displays in the lower right corner.

Displaying force vectors

1. Select **SET** to display the **Global Options** dialog box.
2. Enable **Show Force Vectors**, then select **OK**. All modules display their force vectors (figure 6-5).

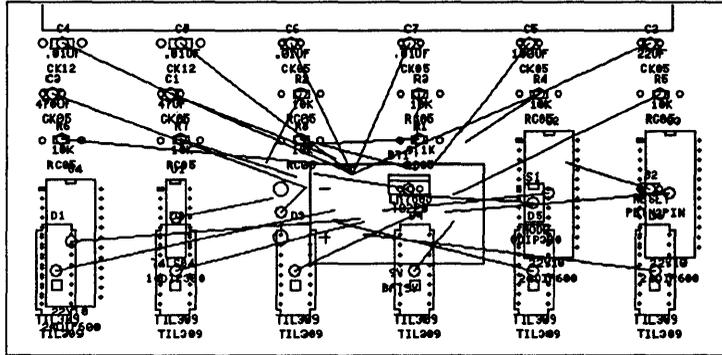


Figure 6-5. Force vectors for the TUTOR layout.

Turning off force vectors

1. Select **SET** to display the **Global Options** dialog box.
2. Disable **Show Force Vectors**, then select **OK**. The force vectors do not display.

Placing the modules

This tutorial describes two methods for placing the modules on the TUTOR board. The method you select is a matter of personal preference. Refer to the section describing the method you wish to use. The two methods are described below.

Coordinate placement—You select the module reference designator from the **Place Module** dialog box, then enter its Block X, Block Y, and Angle values shown in table 6-1 into the **Set Block Parameters** dialog box.

Dynamic placement—You select the module reference designator from the **Place Module** dialog box, then rotate the module, if specified, and use the mouse and arrow keys to move it to the coordinates shown in table 6-1.

Coordinate placement

Follow these steps to place the modules by entering the values from table 6-1:

1. Select **PLACE Module**. The **Place Module** dialog box displays (figure 6-6).

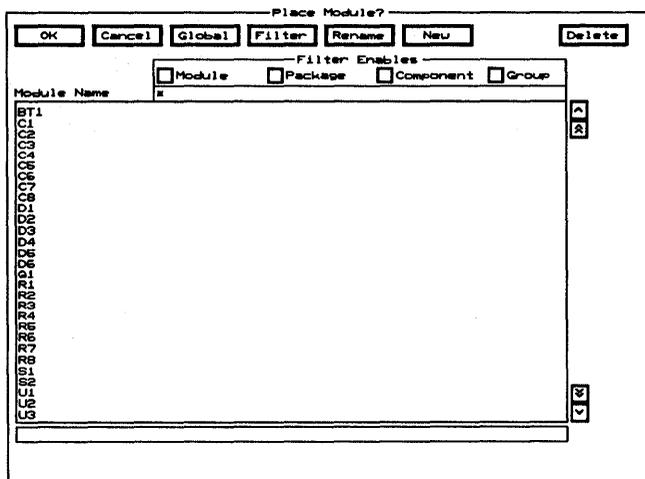


Figure 6-6. The **Place Module** dialog box.

2. Select **BT1**, the first item in the **Module Name** list box, then select **OK**. The pointer jumps to the module and the module is ready to move.

3. Select **Set**. The **Set Block Parameters** dialog box displays.
4. Enter the value in the **Block X** column of table 6-1 into the **Block X** entry box.
5. Enter the value in the **Block Y** column of table 6-1 into the **Block Y** entry box.
6. Enter the value in the **Angle** column of table 6-1 into the **Angle** entry box if the value in table 6-1 is other than zero.
7. Select **OK**. The module moves to the specified coordinates. Check that the coordinates at the bottom of the screen match the entered coordinates.
8. Select **Place**. The module is placed and you return to the **Place Module** dialog box.
9. Repeat steps 2 through 8 for the other modules in the list box.
10. Select **Cancel** after you place U4, the last module.

After you place all modules, the layout looks like figure 6-7.

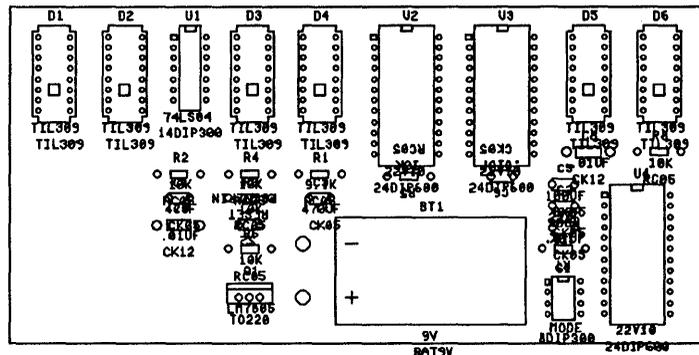


Figure 6-7. The placed TUTOR board.

<i>Reference designator</i>	<i>Block X value</i>	<i>Block Y value</i>	<i>Angle</i>
BT1	3.450	2.450	0.00
C1	2.200	2.025	0.00
C2	5.750	2.100	0.00
C3	3.500	2.025	0.00
C4	2.100	2.275	0.00
C5	5.750	1.900	0.00
C6	5.375	1.825	180.0
C7	5.750	2.300	0.00
C8	5.875	1.600	0.00
D1	1.000	0.525	0.00
D2	1.650	0.525	0.00
D3	2.825	0.525	0.00
D4	3.450	0.525	0.00
D5	5.925	0.525	0.00
D6	6.575	0.525	0.00
Q1	2.850	2.950	0.00
R1	3.400	1.800	0.00
R2	2.100	1.800	0.00
R3	2.750	2.025	0.00
R4	2.750	1.800	0.00
R5	4.600	1.825	180.0
R6	2.750	2.500	0.00
R7	6.050	2.500	180.0
R8	6.525	1.600	0.00
S1	5.700	2.800	0.00
S2	3.050	2.275	180.0
U1	2.250	0.525	0.00
U2	4.100	0.525	0.00
U3	4.975	0.525	0.00
U4	6.225	2.000	0.00

Table 6-1. Placement coordinates and rotation angles for TUTOR board modules.

Dynamic placement

Follow these steps to move the modules to their locations using the mouse and arrow keys:

1. Select **PLACE Module**. The **Place Module** dialog box displays (figure 6-6).
2. Select the first module in the list, then select **OK**. The pointer jumps to the module and the module is ready to move.
3. Select **> Rotate Clockwise** or **< Rotate Counter Clockwise** to rotate the module to the angle shown in table 6-1. The module does not need to be rotated if the listed angle is 0.00. As you rotate the module, the rotation angle displays in the lower right corner of the screen.
4. Using both the mouse and arrow keys, move the module to the coordinates listed in table 6-1. Use the coordinates that display in the lower left part of the screen as a reference.
5. Select **Place**. The module is placed and you return to the **Place Module** dialog box.
6. Repeat steps 2 through 4 for the other modules in the list box.
7. Select **Cancel** after you place the last module in the list.

After you place all modules, the layout looks like figure 6-7.

Editing module placement

You placed the modules in their proper locations, but there is additional work to be done before the module layout is complete.

Hiding module text

The modules display text that is not required for the finished layout. Follow these steps to hide specific module text for all modules on the board:

1. Enlarge the view of the layout. Move the pointer to (0.7000", 0.2000") and select **WINDOW ZOOM**.
2. Enclose the board in a zoom window by moving the pointer to (7.2000", 3.6000").
3. Select **Window Zoom End** to enlarge the layout view.
4. Select **SET**. The **Global Options** dialog box displays.
5. Enable **Hide Module Value Text**, then select **OK**. Note that the module values do not display.
6. Select **SET** to display **Global Options** again.
7. Enable **Hide Module Type Text**, then select **OK**. No module type text displays. Refer to figure 6-8.

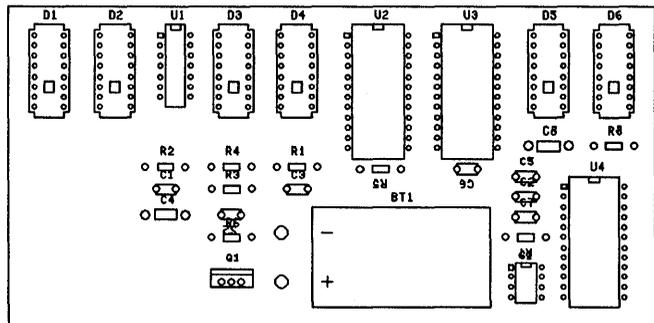


Figure 6-8. The TUTOR board, with hidden module text.

Rotating module text

When you placed the modules, these four reference designators were rotated when you rotated the modules:

- ❖ C6
- ❖ R5
- ❖ R7
- ❖ S2

You need to rotate these reference designators to the same orientation as the rest of the reference designators on the board. Follow this procedure:

1. Place the pointer at (2.7000", 1.7000") and select **WINDOW ZOOM**.
2. Move the pointer to (6.1000", 3.2000") and select **Window Zoom End** to zoom in to the area of the board that has the rotated reference designators.
3. Select **SET** to display the **Global Options** dialog box.
4. Enable **Allow Edits of Module Objects**, then select **OK**.
5. Place the pointer in the center of rotated reference designator **S2** and select **MOVE**.
6. Select **> Rotate Clockwise** to rotate the text 90 degrees clockwise.
7. Select **> Rotate Clockwise** to rotate the text another 90 degrees clockwise. This rotates **S2** so it is oriented properly.
8. Select **Place**.
9. Move the pointer to reference designator **R5** and repeat steps 5 through 8 for **R5**.
10. Repeat steps 5 through 8 for reference designators **C6** and **R7**. Move **R7** up slightly so it is not overlapping **S1**.

Flipping a module to the other side of the board

Design requirements or changes may require you to place modules on the other side of the board. Follow these steps to flip the transistor, Q1, to the other side of the board:

1. Select **SET** to display the **Global Options** dialog box.
2. Disable **Allow Edits Of Module Objects**, then select **OK**. The entire module is selected when this option is disabled.
3. Place the pointer at (2.9500", 2.9500"), the center of the middle pad for Q1, and select **MOVE**.
4. Select **Set** to display the **Set Block Parameters** dialog box.
5. Enable **Flip to other side of board**, then select **OK**.
6. Select **Place** to place the module on the other side of the board.

The module reference designator and silkscreen outline change color to reflect their new layer—the **SilkScreen Solder** layer.

△ **NOTE:** When a module flips to the other side of the board the module text continues to display in its normal orientation, rather than as mirrored text. However, the text prints as mirrored text when you enable **Mirror Text** in the **Printing and Plotting** dialog box. See Chapter 9: **Printing and Plotting the TUTOR board for more information.**

Placing other board objects

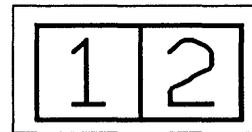
Now that the modules are properly placed on the board, you can identify an area of the board to place these additional objects:

- ❖ Layer marker
- ❖ Board identification
- ❖ Fill zone
- ❖ Alignment targets

Layer marker

A layer marker indicates the number of enabled board layers. A rectangular outline containing a layer number is automatically placed on each of the enabled layers.

1. Select **PLACE Layer Marker**. A layer marker displays, as shown at right.



2. Move the layer marker to the left and pan the display until the pointer is at (0.9500", 3.2500"), then select **Place** to position the layer marker. Another layer marker is attached to the pointer.
3. Press <Esc> or click the right mouse button to dismiss the additional layer marker.

Board identification

Most board layouts have some kind of identification that is either copper etched or silkscreened on the outer layers. The identification can be a board name or a part number.

*Drawing the board
name outline*

You need to draw a rectangular outline, then place the board name inside the rectangle. Follow these steps:

1. Select **LAYER** to display the **Layer** dialog box.
2. Select **SilkScreen Component**, then select **OK**.
3. Move the pointer to (1.5500", 3.0500") and select **PLACE Outline Begin**.
4. Draw the outline segment to the right to (2.5500", 3.0500") and select **Begin**.
5. Draw the segment down to (2.5500", 3.3250") and select **Begin**.
6. Draw the segment to the left to (1.5500", 3.3250") and select **Begin**.
7. Draw the segment up to (1.5500", 3.0500") and select **End** to complete the rectangular outline.

- Placing the board name*
1. Select **PLACE Text**. The **Text** entry box displays.
 2. Enter **TUTOR BOARD**. The text string displays.
 3. Select **Set**. The **Edit Text** dialog box displays (figure 6-9).

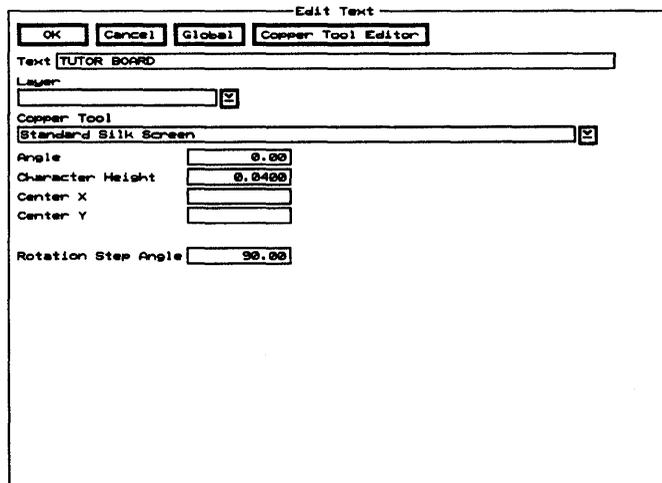
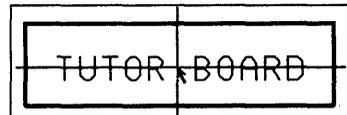


Figure 6-9. The *Edit Text* dialog box.

4. Enter **0.0750** in the **Character Height** entry box, then select **OK**. The text displays at its new size.
5. Move the text to (2.0500", 3.2000") and select **Place**. The **Text** entry box displays.
6. Press **<Esc>** or click the right mouse button to dismiss the **Text** entry box. The completed board identification looks like the illustration at right.



Placing a fill zone

A fill zone is a copper-filled area of the board that can be assigned a net name. Fill zones are commonly used to provide additional circuit grounding on the board.

You need to place a fill zone on the Solder Copper layer and assign it to GND, the ground net. Follow these procedures:

Creating the fill zone

1. Select / **OTHER** until **Solder Copper** displays as the current layer.
2. Select **PLACE Fill Zone**, then move the pointer to (2.6000", 3.0500") and select **Begin**.
3. Draw the zone segment to the left to (0.8000", 3.0500") and select **Begin**.
4. Draw up to (0.8000", 1.5000") and select **Begin**.
5. Draw right to (1.4250", 1.5000") and select **Begin**.
6. Draw down to (1.4250", 2.4000") and select **Begin**.
7. Draw right to (2.2250", 2.4000") and select **Begin**.
8. Draw up to (2.2250", 2.1750") and select **Begin**.
9. Draw right to (2.6000", 2.1750") and select **Begin**.
10. Draw down to (2.6000", 3.0500") and select **End** to complete the fill zone outline.
11. Select **SET** to display the **Global Options** dialog box.
12. Enable **Show Copper Pour**, then select **OK**. The fill zone displays as a solid, filled polygon, as shown in figure 6-10.
13. Place the pointer at (1.9000", 2.0000") and select **WINDOW ZOOM**.

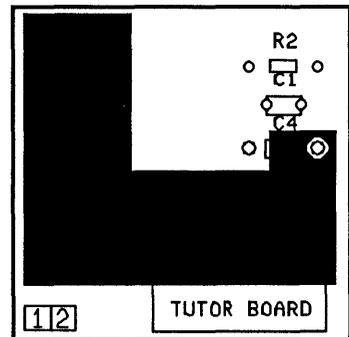
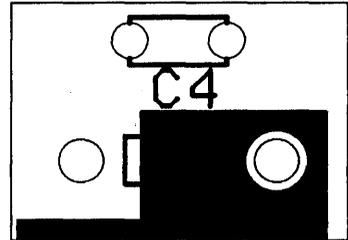


Figure 6-10.
The completed fill zone.

14. Move the pointer to (2.7000", 2.5000) and select **Window Zoom End** to zoom in to the area around the fill zone and module C4.

*Understanding
zone/pad isolation*

Even though the zone is a filled area, **Edit Layout** automatically isolates the C4 pad from the zone, as shown in the illustration at right. **Edit Layout** maintains isolation between these copper objects because they are on the same layer and they are not assigned to the same net.



The amount of isolation is specified in **Copper To Copper Spacing** in the **Global Options** dialog box. The default value is 0.0150.

Follow these steps to determine the assigned net for the pad surrounded by the fill zone:

1. Move the pointer to (2.5000", 2.2750"), the center of the pad.
2. Select **INQUIRE**. The following message displays at the bottom of the screen:

C4 2 gnd Pad: 0.0940" x 0.0940"

The pad is part of the GND net.

Assigning a net to the fill zone

1. Place the pointer at (2.6000", 2.1750"), the upper right corner of the fill zone, and select EDIT. The Edit Zone Segment dialog box displays (figure 6-11).

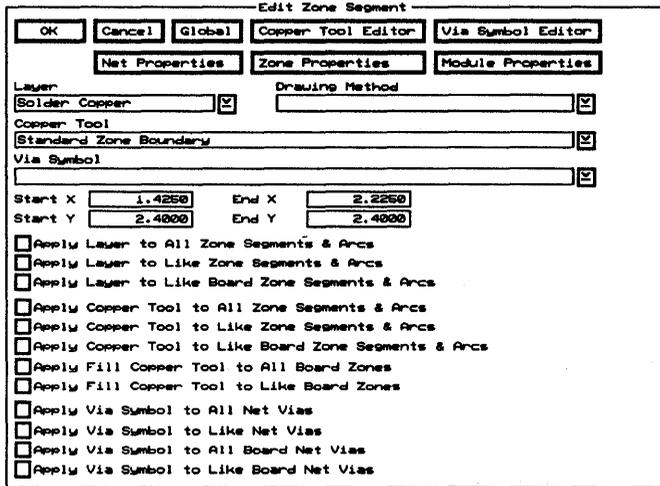


Figure 6-11. The Edit Zone Segment dialog box.

2. Select Zone Properties. The Edit Zone Properties dialog box displays (figure 6-12).

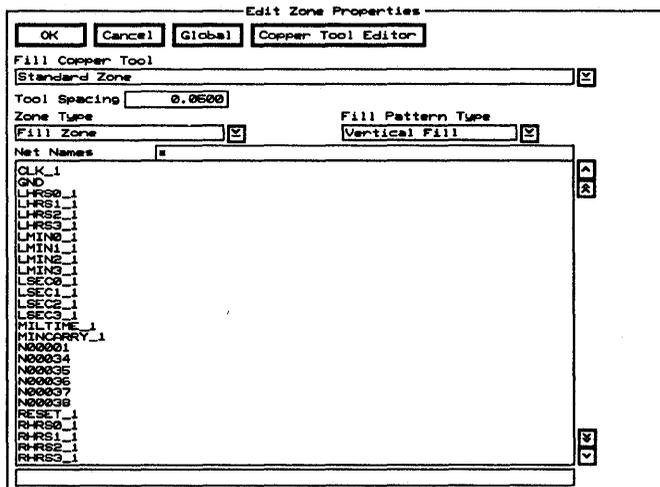
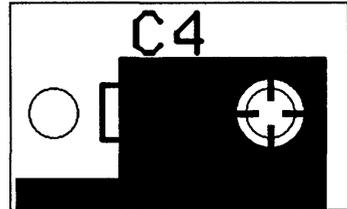


Figure 6-12. The Edit Zone Properties dialog box.

3. Select **GND** from the **Net Names** list box and select **OK**. This assigns the **GND** net to the fill zone and closes the **Edit Zone Properties** dialog box.
4. Select **OK** to close the **Edit Zone Segment** dialog box.

The pad and fill zone are now assigned to the same net and are automatically connected, with thermal relief created between the pad and the zone. See the illustration at right.



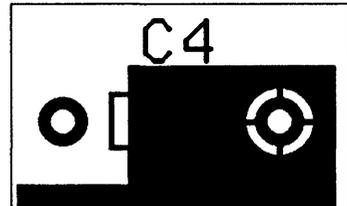
Viewing thermal relief

Follow these steps to change how thermal relief displays between the pad and zone:

1. Select **SET**. The **Global Options** dialog box displays.
2. **Disable Outline Pads.**
3. **Enable Show Drill Hole**, then select **OK**.

The thermal relief area looks like the illustration at right.

4. Select **SET** to display the **Global Options** dialog box.



5. **Enable Outline Pads.**
6. **Disable Show Drill Hole**, then select **OK** to restore the previous display settings.

Refer to the *PC Board Layout Tools 386+ Reference Guide* for a description of thermal relief.

Placing alignment targets

You use alignment targets as visual aids for aligning separately plotted board layers when you check the board design or when the board is fabricated.

Use this procedure to change the zoom scale so you can view the entire board, then place two alignment targets outside the board outline:

1. Select **ZOOM Set Scale**.
2. Enter 13 in the **Scale** entry box and select **OK**. The entire board displays.
3. Select * **LAYER** to set **All Layers** as the current layer.
4. Select **PLACE Alignment Target**. A default alignment target displays, as shown at right.
5. Select **Set**. The **Edit Alignment Target** dialog box displays (figure 6-13).

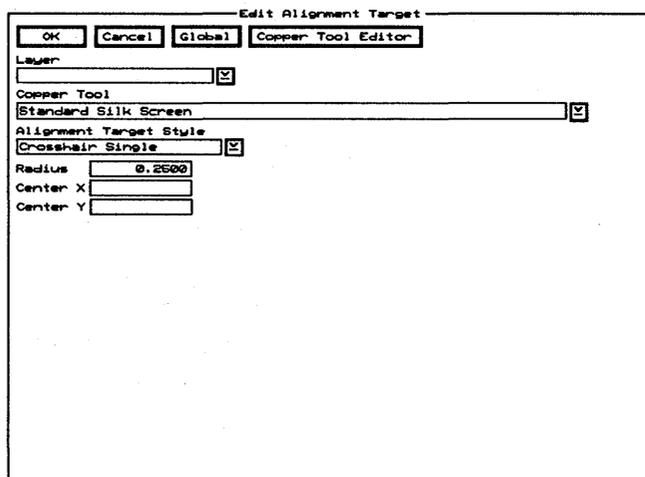


Figure 6-13. The *Edit Alignment Target* dialog box.

6. Enter 0.1500 in the **Radius** entry box.
7. Select the **Alignment Target Style** droplist button to display its droplist box.

8. Select **Crosshair Double** from the droplist box, then select **OK**. The alignment target is smaller, with two concentric circles, as shown at right.
9. Move the pointer to (0.5000", 0.2500") and select **Place**. Another alignment target is attached to the pointer.
10. Move the pointer to (7.3750", 3.5750") and select **Place** to position the second alignment target.
11. Press <Esc> or click the right mouse button to dismiss the additional alignment target.



Saving your work

You are finished placing the TUTOR board, but before you proceed to the next chapter you need to save your work. Follow these steps:

1. Select **QUIT Update Board File**. The **Write Board File** dialog box displays. This dialog box displays because you loaded the **Edit Layout** template file at the beginning of this chapter, and **Edit Layout** requires that you save your work to a new filename.
2. Enter **TUTOR.BD1** in the entry box below the **Files** list box, then select **OK**. **TUTOR.BD1** is saved in the current working directory.

Summary

In this chapter you learned how to create a board outline, load a netlist into **Edit Layout**, and display ratsnest and force vectors as module placement aids. You also learned how to place and edit modules and other objects, and how to assign a net to a fill zone.

The next chapter gives you detailed instructions for manually routing part of the TUTOR board.



Routing the TUTOR board

About manual routing

Some boards may be designed so that connections between pads on a net must have tracks of a specific length or shape. You must place these tracks using manual routing techniques.

In this chapter you use the processes you have already learned, and also learn how to:

- ❖ Highlight a net to locate routing targets
- ❖ Manually route part of the TUTOR board
- ❖ Edit routed tracks

Getting started

In these exercises you manually route only a portion of the TUTOR board. Follow these steps to zoom in on the area where you will route the board.

Zooming in on the routing area

1. Place the pointer at (1.7500", 1.3000") and select **WINDOW ZOOM**.
2. Move the pointer to (4.5000", 3.1000") and select **Window Zoom End**. The enclosed area magnifies to fill the screen.

Highlighting a net

Before you start routing the TUTOR board, you need to identify module pads that share the same net. Follow these steps to highlight net N00034, which has two pads:

1. Select SET to display the Global Options dialog box.
2. Enable Find Highlights, then click OK. Find Highlights highlights an object that is located with the FIND command. If the object is a pad attached to a net, then the entire net is highlighted.
3. Select FIND. The Find entry box displays (figure 7-1).

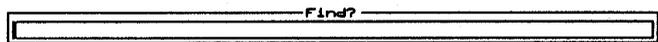


Figure 7-1. The Find entry box.

4. Enter ? in the entry box. The Find dialog box displays (figure 7-2).

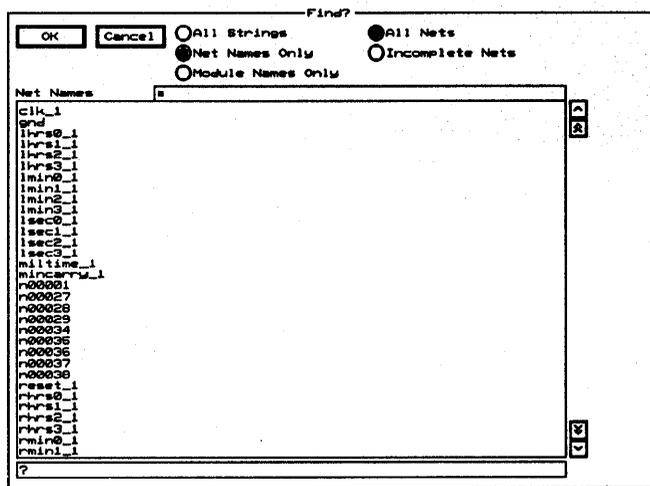


Figure 7-2 The Find dialog box.

5. Select N00034 from the Net Names list box, then select OK. Two pads highlight and the pointer jumps to one of the pads.

Tip

You can enhance the display of highlighted objects by following this procedure:

1. Select **SET** to display **Global Options**.
2. Note **Copper To Copper Spacing** to see if it is set at 0.0150, the default setting. This determines the allowable spacing between tracks, pads, vias, and testpoints.
3. Enable **Show Highlight Guard**. This displays the guard zone established with **Copper To Copper Spacing**.
4. Select **Layer**, then select **Other Colors/Enables/...** from the **Layer** dialog box.
5. Change the color for **SolderMask Solder** to a bright color, such as yellow, then select **OK**.
6. Select **OK** to close the **Layer** dialog box, then select **OK** to close the **Global Options** dialog box.

The highlighted pads now display an additional yellow band, which is the soldermask guard.

**Displaying a ratsnest
for a single pad**

You can also locate routing targets by displaying a single ratsnest between a selected pad and its nearest pad on the same net. Follow these steps:

1. Place the pointer on the highlighted pad for R4.
2. Select **X SHOW RATSNEST**. A single ratsnest vector displays between the two highlighted pads.
3. Move the pointer away from any module pad and select **X SHOW RATSNEST** to clear the ratsnest from the display.

Creating a new copper tool

You need to create a new copper tool for one of the nets that you route later in this chapter. Follow these steps to create a copper tool that produces 30 mil wide tracks:

1. Select **GO TO FUNCTION**. The menu shown at right displays.
2. Select **Copper Tool Editor**. The **Edit Copper Tool** dialog box displays (figure 7-3).

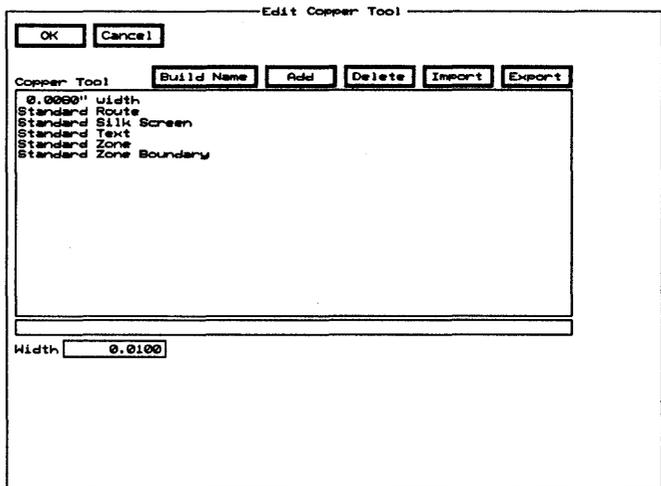
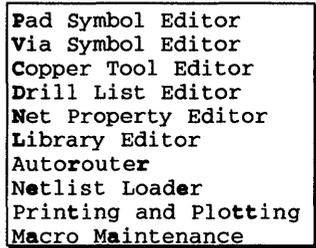
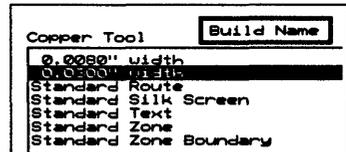


Figure 7-3. The *Edit Copper Tool* dialog box.

3. Enter 0.0300 in the Width entry box.
4. Select **Build Name**. The entry box below the Copper Tool list box displays 0.0300" width.
5. Select **Add**. The new copper tool is added to the Copper Tool list box, as shown in the example at right.
6. Select **OK**.



Routing the board

In these exercises you manually route three nets, using a variety of routing techniques.

Setting conditions

1. Select **SET** to display **Global Options**.
2. Enable **Show Copper And Guard While Routing**. This displays the copper to copper guard zone while you are routing the track.
3. Select the **Drawing Method** droplist button to display the droplist box.
4. Select **Draw Orthogonal 45 Degree Corners**, then select **OK** to close the **Global Options** dialog box.
5. Select **/ OTHER** until **Solder Copper** displays as the current routing layer.

Routing the first track

1. Place the pointer in the center of the highlighted pad for R4.
2. Select **ROUTE Begin**. A ratsnest vector displays, indicating the nearest target pad.
3. Move the pointer to the left to (3.0500", 1.8000"). A track segment, displaying its copper to copper spacing guard, follows the pointer.
4. Move the pointer straight down to (3.0500", 2.2750"), the center of the other highlighted pad (module S2). Notice that the segment automatically bends at a 45° angle.
5. Select **End**. The highlight guard disappears and the message "N00034: Complete" displays at the bottom of the screen, indicating that the net is completely routed. You just routed a track.
6. Move the pointer to an empty part of the display and select **HIGHLIGHT**. This removes the highlight from the pads.

Routing with vias This time you will route net N00037 and place a via.

1. Select **FIND**. The previously routed net name, **N00034**, displays in the entry box.
2. Enter **N00037**. The two net pads are highlighted and the pointer jumps to one of the pads.
3. With the pointer in the center of the highlighted pad for C1, select **ROUTE Begin**. The ratsnest for the net displays.
4. Move the pointer straight up from the pad, then move it to the right to (2.7000", 1.7500").
5. Select / **OTHER**. A via is placed at the pointer location and the current routing layer switches from Solder Copper to Component Copper.

Another ratsnest vector displays, connecting the highlighted pad for C1 and the via. This indicates that the connection up to the via is a subnet of the entire net.

6. Draw the new segment straight to the right, then down to (3.8000", 1.8000"), the other highlighted pad, and select **End**. The message "N00037: Complete" displays and the track is completed.

Routing with arc segments

You can route with arc segments in **Edit Layout**. Arc segments can be used to tightly route around and through pads or other objects. Follow these steps to begin a track, then change to an arc segment and route around a pad:

1. Move the pointer to an empty part of the display and select **HIGHLIGHT** to remove the highlight from the previously routed pads.
2. Select **FIND**, then erase the net name in the entry box and enter ? to display the **Find?** dialog box.
3. Select **N00038** from the **Net Names** list box and select **OK**. Three pads highlight.
4. Move the pointer to (3.0500", 2.9500"), the center of the highlighted pad for Q1.
5. Select **ROUTE Begin** and draw the segment to the right to (3.4500", 2.9500"), the highlighted pad for BT1.
6. Select **Begin**, then draw a segment straight up to (3.4500", 2.5500").
7. Select **Begin**, then select **Set**. The **Edit Net Segment** dialog box displays (figure 7-4). Note that the current selection for **Drawing Method** is **Draw Orthogonal 45 Degree Corners**.

Edit Net Segment

Layer: Drawing Method:

Copper Tool:

Via Symbol:

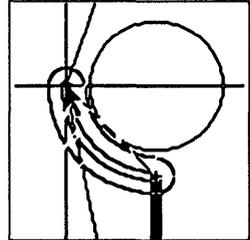
Start X: End X:
 Start Y: End Y:

Apply Layer to All Net Segments & Arcs
 Apply Layer to Like Net Segments & Arcs
 Apply Layer to Like Board Net Segments & Arcs
 Apply Copper Tool to All Net Segments & Arcs
 Apply Copper Tool to Like Net Segments & Arcs
 Apply Copper Tool to Like Board Net Segments & Arcs
 Apply Fill Copper Tool to All Board Zones
 Apply Fill Copper Tool to Like Board Zones
 Apply Via Symbol to All Net Vias
 Apply Via Symbol to Like Net Vias
 Apply Via Symbol to All Board Net Vias
 Apply Via Symbol to Like Board Net Vias

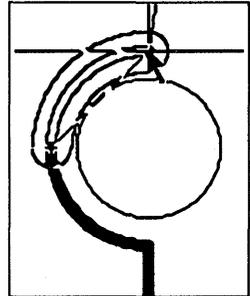
Figure 7-4. The **Edit Net Segment** dialog box.

8. Select the **Drawing Method** droplist button to display the options in the droplist box.
9. Select **Draw Orthogonal Arc Corners**. The droplist box closes and the new selection displays.
10. Select **OK** to close the **Edit Net Segment** dialog box.

11. Move the pointer to the left, then up to (3.3500", 2.4500"). Notice that the segment following the pointer is now an arc. See the illustration at right.



12. Select **Begin**. Move the pointer up, then to the right to (3.4500", 2.3500"). The two arc segments follow the contour of the large pad. See the illustration at right.



13. Select **Begin**, then select **Set** to display the **Edit Net Segment** dialog box.
14. Select the **Drawing Method** droplist button and select **Draw Orthogonal 45 Degree Corners** from the droplist box. Select **OK**.
15. Move the pointer up, then right to (3.5000", 2.0250"), the center of the third highlighted pad. Notice that the segment now draws at a 45 degree angle.
16. Select **End** to complete this complex track.
17. Move the pointer away from the pad and select **HIGHLIGHT**. The three finished tracks look like figure 7-5.

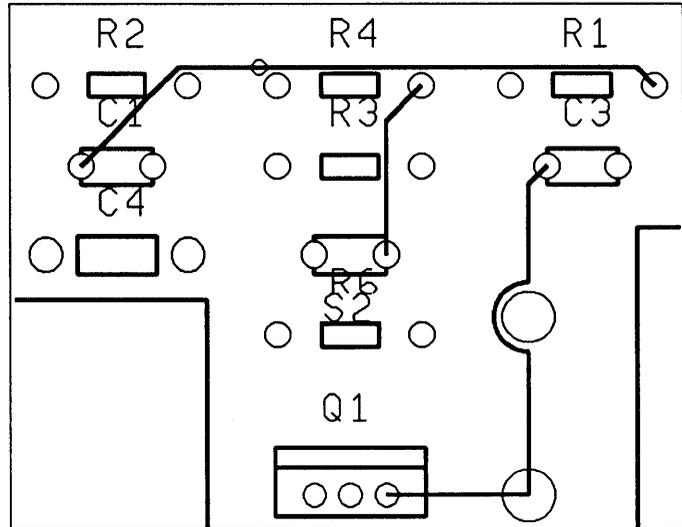


Figure 7-5. The three completed tracks.

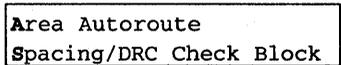
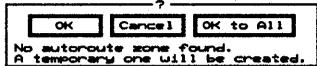
Performing a DRC check

One of the tracks you routed is too close to several pads. The allowed clearance between tracks, pads, and vias is specified in **Copper To Copper Spacing** in **Global Options**. The default value is 0.0150 inches (15 mils).

You locate routing and clearance violations by performing a DRC check. DRC stand for *Design Rule Check*.

You perform a DRC check two ways in **Edit Layout**: a whole board check and a block check, which checks a selected area. Follow these steps to perform a block DRC check:

Running a block DRC check

1. Move the pointer to (1.9000", 1.6000") and select **GO TO FUNCTION** Autorouter.
2. Select **Block**, then move the pointer and stretch the bounding box to (3.9000", 3.0000"), enclosing the routed area inside the block.
3. Select **Block End**. The menu shown at right displays.
4. Select **Spacing/DRC Check Block**. The dialog box shown at right displays.
5. Select **OK to ALL**. The display zooms in to the block area. In a moment the dialog box shown at right displays, showing the number of spacing/DRC errors found within the block.
6. Select **OK**. DRC markers display, pointing to the violated objects. See figure 7-6.
7. Press <Esc> to dismiss the autorouter menu.

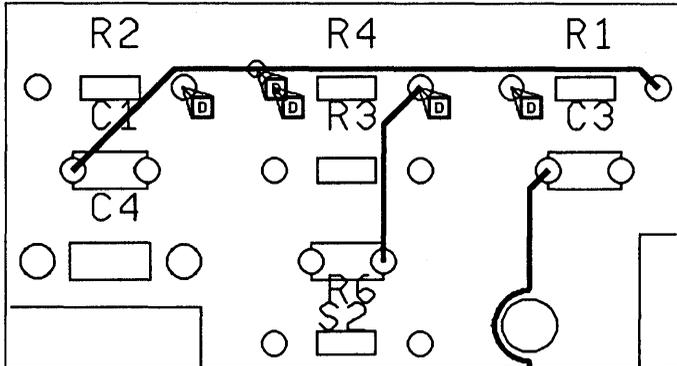


Figure 7-6. DRC markers indicating violations.

Identifying DRC violations

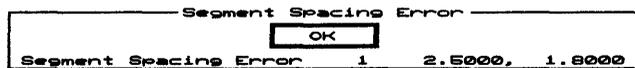
A DRC marker looks like the example shown at right. The converging lines point to the center of the violated object.



You identify DRC violations using two commands: **INQUIRE** and **JUMP**. Both methods are described in the following procedures.

Using *INQUIRE*

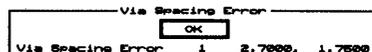
1. Select **LAYER** to display the Layer dialog box.
2. Select **Comment Layer**, then select **OK**. When Comment Layer is the current layer you can easily select DRC markers because all other objects assigned to layers are not selectable.
3. Move the pointer to (2.5500", 1.8500") and select **INQUIRE**. The dialog box shown below displays.



The dialog box identifies the type of violation, and the coordinates for the violated object.

4. Select **OK**.

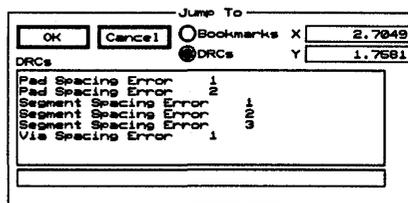
5. Move the pointer to (2.7000", 1.7500"), the center of the via, and select **INQUIRE**. The dialog box shown at right displays. This dialog box tells that the copper to copper spacing zone of the via is violated by another object (a via spacing error).
6. Select **OK**.



Using JUMP

1. Select **JUMP** to display the **Jump To** dialog box.

2. Select **DRCs** to display all design errors in the **DRCs** list box, as shown in figure 7-7.



3. Select **Segment Spacing Error 3**, then select **OK**.

Figure 7-7. The Jump To dialog box, showing design errors.

The pointer jumps to that DRC marker. Select **INQUIRE** to verify that it is **Segment Spacing Error 3**, then select **OK**.

4. Select **JUMP** and select the other items in the **DRCs** list box to see how you can locate a specific violation.

Viewing the violated areas

You can change the way objects on copper layers display so you can view the areas where the design violations occurred. Follow these steps:

1. Zoom in on the area containing the violations by placing the pointer at (2.4000", 1.6500") and selecting **WINDOW ZOOM**.
2. Move the pointer to (3.5000", 1.9500") and select **Window Zoom End**. The display magnifies to the selected area.
3. Select **SET** to display the **Global Options** dialog box.

4. Enable **Show Copper And Guard While Drawing**, then select **OK**. All copper objects display their copper to copper spacing boundaries, as shown in figure 7-8.

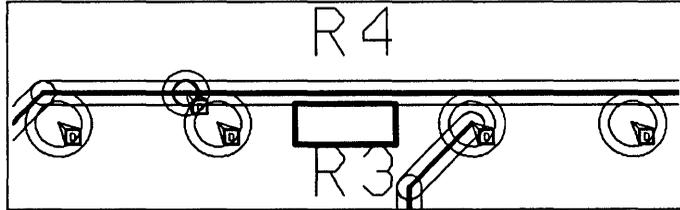


Figure 7-8. Copper to copper spacing of violated objects.

The guard band for the track extends inside the guard bands for the four marked pads. The guard band for the marked via extends inside the adjacent module pad.

5. Select **SET** to display **Global Options**.
6. Disable **Show Copper And Guard While Drawing**, then select **OK**. The copper to copper spacing does not display.

The next section describes how to correct DRC errors.

Editing the routed board

After manually routing a board, you may need to edit your work to correct DRC errors and make design modifications.

Correcting DRC errors

Follow these procedures to correct the DRC errors described earlier in **Performing a DRC check**:

Drawing a new track

1. Select **ZOOM Set Scale**, then enter **3.25** in the **Scale** entry box and select **OK**. The display zooms out so you can view the entire track that is violating the pads.
2. Select **/ OTHER** until **Solder Copper** is the current layer.
3. Move the pointer to (2.2000", 2.0250"), a routed pad.
4. Select **ROUTE Begin**. A ratsnest vector displays, between the pad and the via, indicating the nearest subnet routing connection. See the *PC Board Layout Tools 386+ Reference Guide* for a description of subnets.
5. Draw the segment straight up, then to the right to (2.8000", 1.9000") and select **/ OTHER** to place a via and switch the routing layer to **Component Copper**.
6. Draw the segment to the right to (3.9000", 1.9000"), then select **New** to complete the segment.
7. Move the pointer to (3.8000", 1.8000"), the routed pad for R1, and select **Begin** to start another segment.
8. Draw the segment to (3.7000", 1.9000") and select **End**. The message "N00037: Complete" displays and the new track is complete, as shown in figure 7-9.

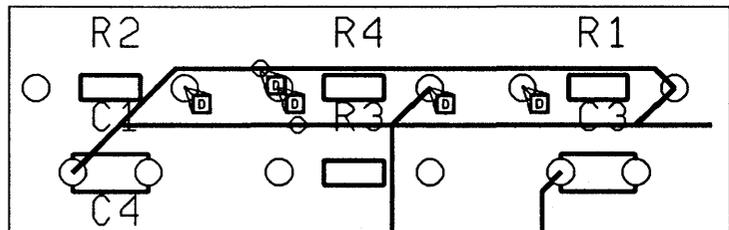


Figure 7-9. The old and new tracks for net N00037.

Deleting the offending track

1. Place the pointer at (3.7500", 1.7500"), the junction where the horizontal segment for the old track angles into the pad for R1.
2. Select **DELETE** to delete one of the segments that intersects at the pointer location.
3. Select **DELETE** to delete the other intersecting segment.
4. Select **/ OTHER** to select Solder Copper as the current layer.
5. Move the pointer to (2.4750", 1.7500"), the junction where the horizontal segment for the old track angles into the pad for C1.
6. Select **DELETE** twice to delete the two segments.
7. Place the pointer at (2.7000", 1.7500"), the center of the via with a DRC marker, and select **DELETE**.
8. Select **ZOOM Refresh** to redraw the display. The old track is deleted, but the DRC markers remain.

Running another DRC check

Now you can check your work to see if the new track has enough clearance from the pads. Follow these steps:

1. Select **GO TO FUNCTION Autorouter**.

2. Select **Whole Board**. The menu shown at right displays.

```
Autoroute Whole Board
Spacing/DRC Check Whole Board
Set Sweep Window
```

3. Select **Spacing/DRC Check Whole Board**. The entire board displays, and in a moment the dialog box shown at right displays. The number of DRC errors is zero.

```
Finished
OK
Time Used: 0:00:12
Number of Spacing/DRC Errors Found = 0
```

△ **NOTE:** You can also define a block around the modified area and select **Spacing/DRC Check Block**.

4. Select **OK** to close the dialog box. Pan the display so you can view the routed area. The DRC markers are removed.
5. Press **<Esc>** to dismiss the autorouter menu.
6. Place the pointer at (1.9500", 1.5000") and select **WINDOW ZOOM**.
7. Move the pointer to (4.0000", 3.1000") and select **Window Zoom End** so the display looks similar to figure 7-10.

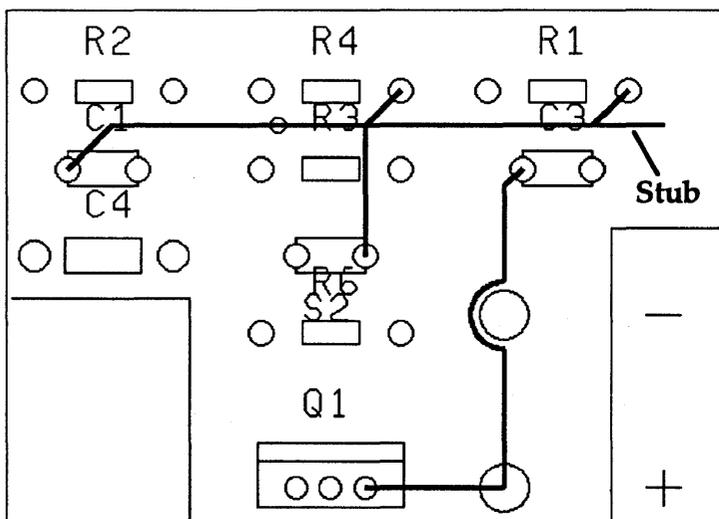


Figure 7-10. The edited routing area.

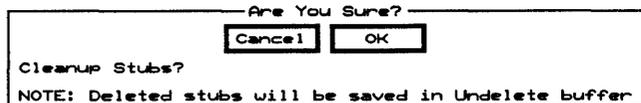
Deleting a stub

The track you just completed has an unneeded extra segment, called a "stub" (see figure 7-10). Delete the stub using one of the two methods described below:

1. Select / OTHER to select Component Copper as the current layer.
2. Place the pointer at (3.9000", 1.9000"), the end of the stub, and select DELETE. Remember that deleted objects are retained in the undelete buffer.

Or, use this method:

1. Select QUIT Cleanup Stubs. The dialog box shown below displays.



2. Select OK. The stub is deleted from the board, but it is retained in the undelete buffer.

Deleting and undeleting a track

It is easy to delete and undelete an entire track, including vias, that connects two pads. Follow this procedure:

1. Place the pointer at (3.3000", 1.9000"), which positions it on a Component Copper segment of the track connecting C1 and R1.
2. Select TRACK DELETE. The entire track, including the via, is deleted.
3. Select UNDELETE. The entire track is recovered.



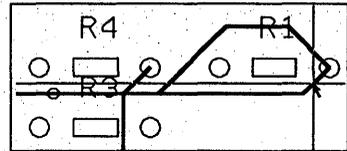
NOTE: You can also use TRACK DELETE to delete sections of a track that have been replaced by rerouted segments. See Changing a track path for an example.

Changing a track path

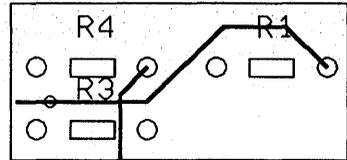
As you route your board, you usually have tracks that need to be rerouted to improve the board layout.

1. Place the pointer at (3.8000", 1.8000"), the center of the routed pad for R1, and select **ROUTE Begin**.
2. Draw the segment up, then left to (3.4500", 1.6500") and select **Begin**.
3. Draw the segment to (3.2000", 1.9000") and select **End**. The track looks like the illustration below.

4. Place the pointer at (3.7250", 1.8750"), the first segment of the original track connected to R1. See the illustration at right.



5. Select **TRACK DELETE**. The track segments delete from the pad to the point where the new track intersects the original segments.



6. Select **ZOOM Refresh** to redraw the display.

Changing track width

Some of the nets in the design require wide tracks because of the types of signals they carry. Wide power and ground nets are common because those signals usually require larger copper areas.

Follow these steps to change the track width for a routed net to 0.0300" (30 mils):

1. Move the pointer to (3.0500", 2.1000"), which positions it on a track. Note that the track color indicates it is on the Solder Copper layer.
2. Select / OTHER to change the current layer to Solder Copper.
3. Select EDIT. The Edit Net Segment dialog box displays (figure 7-11).

Edit Net Segment

OK Cancel Global Copper Tool Editor Via Symbol Editor

Net Properties Zone Properties Module Properties

Layer: Solder Copper [v] Drawing Method: [v]

Copper Tool: Standard Route [v]

Via Symbol: Standard Through Via [v]

Start X: 3.0500 End X: 3.0500
Start Y: 2.2760 End Y: 1.9000

Apply Layer to All Net Segments & Arcs
 Apply Layer to Like Net Segments & Arcs
 Apply Layer to Like Board Net Segments & Arcs
 Apply Copper Tool to All Net Segments & Arcs
 Apply Copper Tool to Like Net Segments & Arcs
 Apply Copper Tool to Like Board Net Segments & Arcs
 Apply Fill Copper Tool to All Board Zones
 Apply Fill Copper Tool to Like Board Zones
 Apply Via Symbol to All Net Vias
 Apply Via Symbol to Like Net Vias
 Apply Via Symbol to All Board Net Vias
 Apply Via Symbol to Like Board Net Vias

Figure 7-11. The Edit Net Segment dialog box.

4. Select the Copper Tool droplist button to display the droplist box. Standard Route is highlighted.
5. Select 0.0300" width. The droplist box closes and the new copper tool displays in the dialog box.
6. Enable Apply Copper Tool to All Net Segments & Arcs, then select OK. All segments for the routed net are redrawn using the selected 30 mil copper tool.

Saving your work

It is time to save your work again before you go to the next chapter. Select **QUIT Update Board File** to save the file as **TUTOR.BD1**.

Summary

In this chapter you learned how to identify routing targets and manually route part of the board. You also learned how to perform a DRC check and edit tracks.

The next chapter tells you how to automatically route the TUTOR board using the **Edit Layout** autorouter.



Autorouting the TUTOR board

About autorouting

Autorouting a board can save you many hours of manual routing work by automatically routing connections.

The autorouter in **Edit Layout** has many routing options that you can select from to successfully route your board and enhance the routing results.

In this chapter you learn how to:

- ❖ Place an autoroute zone
- ❖ Lock an existing route
- ❖ Set routing conditions for a net
- ❖ Specify autoroute options
- ❖ Autoroute a board
- ❖ Optimize an autorouted board

Preparing for autorouting

Before autorouting the board, you need to do these tasks:

- ❖ Place an autoroute zone to specify the area of the board to autoroute
- ❖ Lock a manually routed net so it will not be rerouted
- ❖ Specify a different copper tool for autorouting the ground net
- ❖ Set autorouting options

Placing an autoroute zone

The autorouter requires an autoroute zone to define the area of the board where the autorouter places tracks on the enabled layers. If you do not define a zone, a zone is created by the autorouter before routing begins.

In this tutorial you create an autoroute zone that leaves a 50 mil border around the edge of the board. The area outside the autoroute zone will not contain any copper tracks because all tracks stay inside the autoroute zone.

Follow these steps to create an autoroute zone:

1. Select **ZOOM Set Scale**, then enter **13** in the **Scale** entry box and select **OK**. The entire board displays.
2. Select *** LAYER** to set **All Layers** as the current layer. The autoroute zone is placed on all board layers when **All Layers** is selected.
3. Select **PLACE Autoroute Zone**.
4. Select **Set**. The **Edit Zone Segment** dialog box displays.
5. Select the **Drawing Method** droplist button to display the droplist box. **Draw Orthogonal 45 Degree Corners** is the current selection.
6. Select **Draw Orthogonal 90 Degree Corners**, then select **OK**.
7. Move the pointer to (0.8000", 0.3000"), then select **Begin**.
8. Draw the zone segment to the right to (7.0750", 0.3000") and select **Begin**.
9. Draw the segment down to (7.0750", 3.3250") and select **Begin**.
10. Draw the segment left to (0.8000", 3.3250") and select **Begin**.
11. Draw the segment up to (0.8000", 0.3000") and select **End** to complete the autoroute zone.

Locking an existing route

To prevent the autorouter from changing a critical route that you routed manually, you lock the route using the following procedure:

1. Move the pointer to (0.7000", 0.2000"), the upper left corner of the board, and select **WINDOW ZOOM**.
2. Move the pointer to (3.9000", 3.4000") and select **Window Zoom End**, magnifying the left half of the board.
3. With All Layers selected, place the pointer at (3.3000", 2.9500"). This places the pointer on the segment connecting module Q1 and the battery, BT1.
4. Select **EDIT**. The Edit Net Segment dialog box displays.
5. Select **Net Properties**. The Edit Net Properties dialog box displays (figure 8-1). N00038, the net name for the selected track, is selected in the Net Names list box.

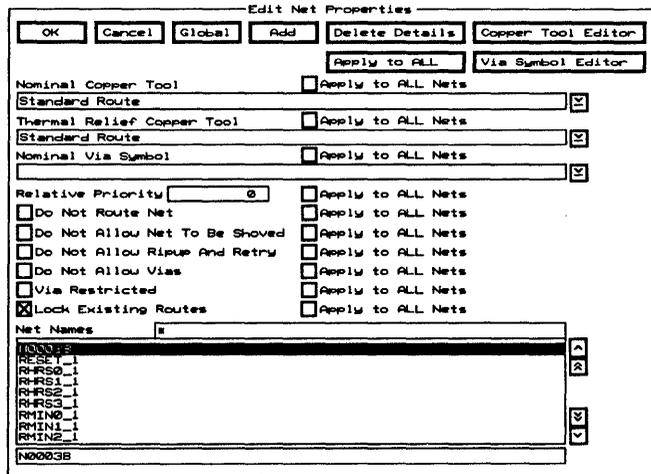


Figure 8-1. The Edit Net Properties dialog box.

6. Enable **Lock Existing Routes**, then select **OK**. The autorouter will not reroute the selected net.
7. Select **OK** to close the Edit Net Segment dialog box.

Setting routing conditions for a net

You can give the autorouter specific instructions on how to route a net. The following procedures set routing conditions for the GND net.

Specifying a copper tool

The GND net should have wide copper tracks so proper grounding is achieved. Follow these steps to assign a 30 mil copper tool to the net:

1. Select **GO TO FUNCTION**. The menu shown at right displays.
2. Select **Net Property Editor**. The **Edit Net Properties** dialog box displays.
3. Select **GND** from the **Net Names** list box. This is the net name for the ground net.
4. Select the **Nominal Copper Tool** droplist button to display the droplist box.
5. Select **0.0300" width** from the droplist box. The selection displays in the **Nominal Copper Tool** list box.

- | |
|--|
| Pad Symbol Editor
Via Symbol Editor
Copper Tool Editor
Drill List Editor
Net Property Editor
Library Editor
Autorouter
Netlist Loader
Printing and Plotting
Macro Maintenance |
|--|

Excluding vias

Follow these steps to specify that the GND net should be routed without using vias:

1. With the **Edit Net Properties** dialog box still displayed, enable **Do Not Allow Vias**.
2. Select **OK** to accept the selections for the GND net and close the **Edit Net Properties** dialog box.

Setting autorouter options

You select autorouter options in **Edit Layout** to specify the type of routing pattern and the direction that the routing sweep window moves. The options you select depend on the way the board is layed out, and the types of modules placed on the board.

Procedures are described in the following sections for setting autorouting options for the TUTOR board.

Setting an autoroute method

1. Select **GO TO FUNCTION Autorouter**, then select **Set**. The **Autoroute Options** dialog box displays (figure 8-2).

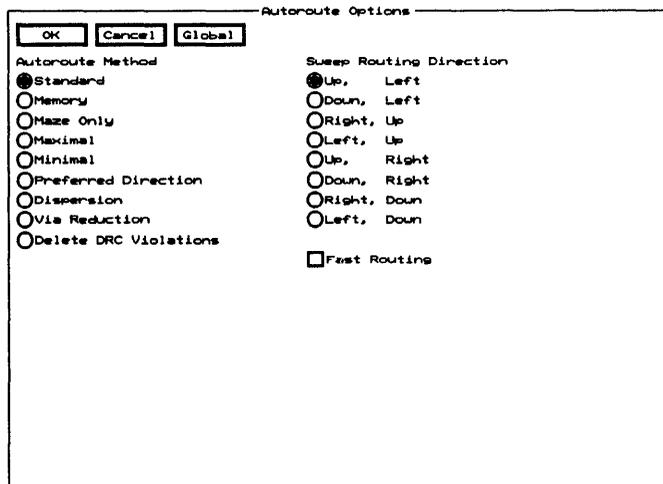


Figure 8-2. The *Autoroute Options* dialog box.

2. Leave **Standard** selected in **Autoroute Method**.

Setting a sweep routing direction

You specify the direction of travel for the sweep window based on the density of the surrounding circuitry. Ideally, the sweep window should work its way from the most dense area of the board to the least dense.

1. Select **Down, Right** in **Sweep Routing Direction**.
2. Select **OK** to accept the autoroute options and close the dialog box.

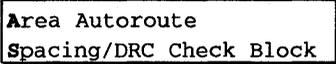
Autorouting the board

You can autoroute an entire board, or you can autoroute a selected area and manually route the rest of the board. The methods you use depend on the routing requirements of the design.

Autorouting a section of the board

1. Place the pointer at (0.8000", 0.3000"), the upper left corner of the autoroute zone, and select **Block**.

2. Move the pointer to (2.7000", 1.4000") and select **Block End**. The menu shown at right displays.



```
Area Autoroute
Spacing/DRC Check
Block
```

3. Select **Area Autoroute**.

The selected autoroute area magnifies to fill the screen and the autorouter begins routing tracks between pads. Tracks that connect to pads outside the block route to the edge of the block.

During autorouting, routing statistics display at the bottom of the screen, as shown below.

Standard-Window Completed 15 Failed 0 Remaining 11

These statistics list the current autorouting options, and the number of completed, failed, and remaining connections. The statistics update as routing connections are made.

△ **NOTE:** You can terminate an autorouting session at any time by pressing <Esc>.

When the block autoroute is finished, the **Finished** dialog box displays (figure 8-3).

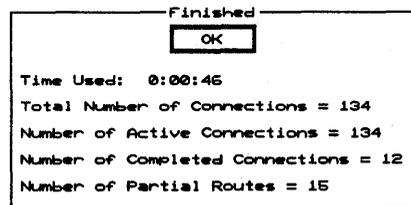


Figure 8-3. The *Finished* dialog box.

The **Finished** dialog box displays total routing time and connection statistics.

4. Select **OK** to close the dialog box. The ratsnest for the whole board displays, and the ratsnest for the incomplete routed tracks displays at their unconnected endpoints. See figure 8-4.

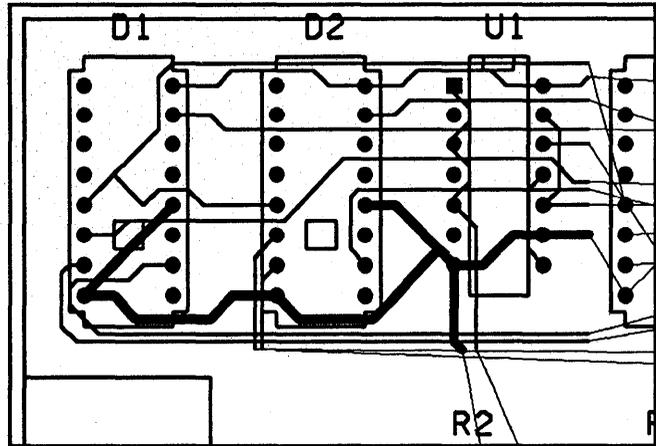


Figure 8-4. The finished autorouted area.

Note the wider tracks. These are the 30 mil GND net tracks that you specified earlier in this chapter in **Setting routing conditions for a net**.

5. Press <Esc> to exit the **Autorouter** menu.
6. Select ? **CONDITIONS** to display the **Conditions** dialog box (figure 8-5). **Conditions** displays additional routing information, such as the number of incomplete nets, the number of vias placed during autorouting, and the total length of all routed tracks.
7. Select **OK** to close the dialog box.

Conditions	
Modules	30
Pads	229
Nets	43
Incomplete	39
Do Not Route	0
Segments	93
Vias	0
Objects	978
Route Length (in)	19.36
(mm)	491.36
Design Space	62047
Total Allocated Memory	3391488
Allocated Physical Memory	1644192
Swap File Size	1647296
Current Page Faults	3

Figure 8-5. The **Conditions** dialog box.

Autorouting the whole board

Before you autoroute a complex board you should review the layout to determine where you want the autorouter to start. The autorouter may complete all connections in one routing pass if you direct its movement properly.

If there are incomplete connections after the first rigorous routing pass, you select a more liberal routing option for the second pass to complete the remaining connections.

For example, you could select **Standard** option for the first pass, then select the more liberal **Maximal** for the second routing pass. After all connections are complete, you could select **Via Reduction** to reduce the number of vias.

The TUTOR board is not a complex layout, but you will use routing methods that are used on complex boards so you can become familiar with the options.

Setting a sweep window

You set a sweep window to define the routing origin and window size. You usually set a sweep window to initially route in the densest area of the board, then follow a sweep pattern to the least dense area. Follow these steps to set a sweep window:

1. Select **ZOOM Set Scale**, then enter **13** in the **Scale** entry box and select **OK**. The entire board displays, showing the ratsnest for all pads.
2. Select **GO TO FUNCTION Autorouter**.
3. Select **Whole Board**. The menu shown at right displays.
4. Select **Set Sweep Window**.
5. Place the pointer at (7.0750", 0.3000"), the upper right corner of the autoroute zone, then select **Sweep Window Begin**.
6. Move the pointer left, then down to (5.0500", 1.9000") and select **Sweep Window End**. The menu shown above displays again.

Autoroute Whole Board Spacing/DRC Check Whole Board Set Sweep Window
--

Begin autorouting

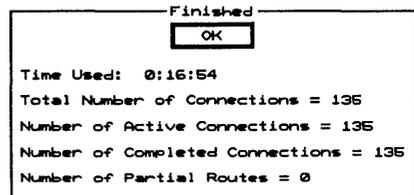
1. With the menu still displayed, select **Autoroute Whole Board** to begin autorouting.

The display zooms in to the initial sweep window and routes connections. The display pans as the autorouter moves to the next sweep window and routes connections.

Statistics describing the routing status of the current sweep window display at the bottom of the screen, as shown in the example below.

Standard-Maze: Completed 15 Failed 0 Remaining 8

When the autorouter is finished, the **Finished** dialog box displays. Note that there are no incomplete connections or partial routes.



2. Select **OK** to close the dialog box.
3. Select **? CONDITIONS** to display the **Conditions** dialog box. Note that the number of incomplete nets is zero. Also note the number of vias used on the board and the total route length.
4. Select **OK** to close the dialog box.
5. Verify that the manually routed track that you specified as a locked track remains unchanged after the autorouting process.

Move the pointer and pan the display until the pointer is at (3.4500", 2.9500"), the lower large pad for battery BT1. The track including the arc segments around the other BT1 pad is unchanged. Refer to figure 8-6.

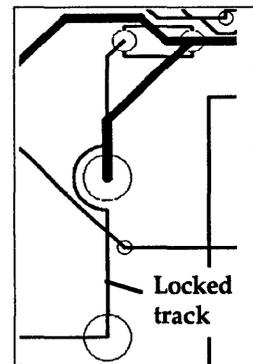


Figure 8-6. The locked track.

Via reduction After the autorouter completes all connections you may want to reroute the board to eliminate some of the vias. Follow these steps to automatically remove vias from the board:

1. Select **Set** to display the **Autoroute Options** dialog box.
2. Select **Via Reduction** in **Autoroute Method**. **Up, Left** is automatically selected in **Sweep Routing Direction**.
3. Select **OK**.
4. Select **Whole Board**, then select **Autoroute Whole Board** to begin the via reduction process.

The display pans when the via reduction process in the initial sweep window is complete, and the sweep window moves to the next section of the board.

Statistics describing the routing status of the current sweep window display at the bottom of the screen.

When the via reduction process is complete the **Finished** dialog box displays.

5. Select **OK** to close the dialog box.
6. Select **ZOOM Set Scale**, then enter **13** in the **Scale** entry box and select **OK**. Pan the display to view the entire board, as shown in figure 8-7.

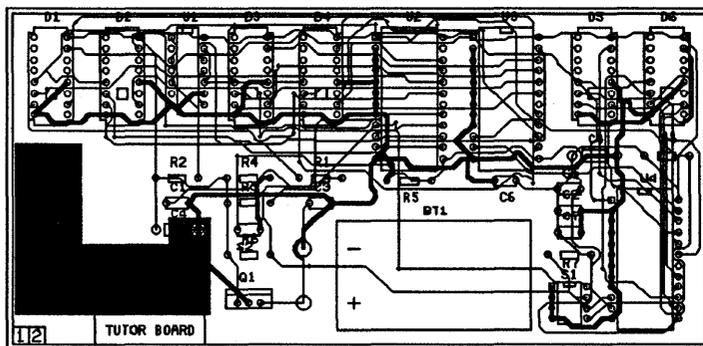


Figure 8-7. The autorouted TUTOR board, after via reduction.

Additional processing

Performing another via reduction may remove more vias, but the additional rerouting of tracks could increase the total track length on some boards.

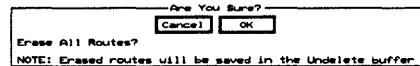
Follow these steps to perform another via reduction:

1. Select **Whole Board**, then select **Autoroute Whole Board** from the menu. The autorouter reroutes tracks and removes additional vias. When the process is complete, the **Finished** dialog box displays.
2. Press <Esc> to exit the **Autorouter** menu.
3. Select **? CONDITIONS** to display the **Conditions** dialog box. Note the number of vias.
4. Select **OK** to close the dialog box.

Erasing all routes

You can completely erase all tracks and vias if you want to autoroute the board using different options. This part of the tutorial is optional. Follow these steps:

1. Select **QUIT Erase All Routes**. The dialog box shown at right displays.



2. Select **OK**. All tracks and vias are deleted, but saved in the undelete buffer.

If you do not want to retain the objects in the undelete buffer, you can flush the undelete buffer by selecting **QUIT Flush Undelete Buffer** and selecting **OK** in the displayed dialog box.

3. Repeat the steps in this chapter, starting with **Autorouting the whole board**. Select different autorouting and sweep window options.

Finishing the layout

Now that the board is completely routed, you need to move the silkscreened reference designators so they are not positioned over any tracks on the Component Copper layer.

You also need to place an assembly outline around all objects in the board file and add dimensions. The assembly outline defines the printing and plotting area when you print and plot the board in *Chapter 9: Printing and plotting the TUTOR board*.

Moving reference designators

1. Select **SET** to display the **Global Options** dialog box.
2. Enable **Allow Edits Of Module Objects** so you can move the reference designators, but not move the rest of the module objects.
3. Select **Layer** to display the **Layer** dialog box.
4. Select **SilkScreen Component**, then select **OK** to set **SilkScreen Component** as the current layer.
5. Select **OK** to accept the changes and close the **Global Options** dialog box.
6. Place the pointer on a reference designator and select **MOVE**.
7. Move the reference designator so it is close to the module, but not positioned over a pad, via, or track on the Component Copper layer.

If you need to change the grid size to achieve better placement, select **Set** to display the **Set Block Parameters** dialog box. Select **Global** to display the **Global Options** dialog box, then enter the new grid size in **Grid Size**. Select **OK** to close **Global Options**, then select **OK** to close **Set Block Parameters**. The grid size is changed.

8. Select **Place**.
9. Pan to adjacent parts of the board and move all reference designators that are on the **SilkScreen Component** layer.

10. To move the reference designator for Q1, which is on the SilkScreen Solder layer, select **LAYER** to display the Layer dialog box. Select **SilkScreen Solder**, then select **OK** and move the reference designator.
11. Select **ZOOM Set Scale**, then enter 13 in the Scale entry box and select **OK**. Pan the display to view the entire TUTOR board.

Placing an assembly outline

1. Select **LAYER**. The Layer dialog box displays.
2. Select **Assembly Drawing**, then select **OK** to set the Assembly Drawing layer as the current layer.
3. Move the pointer to (0.0000", 0.0000") and select **PLACE Outline Begin**.
4. Draw the outline segment to the right to (7.7000", 0.0000") and select **Begin**.
5. Draw another segment down to (7.7000", 5.0000") and select **Begin**.
6. Draw another segment left to (0.0000", 5.0000") and select **Begin**.
7. Draw another segment up to (0.0000", 0.0000") and select **End** to complete the outline.

You can include additional objects on the Assembly Drawing layer, such as documentation, a title block, and even a company logo if the logo is drawn in the library editor and placed as a module.

Placing dimensions

Placing the first dimension

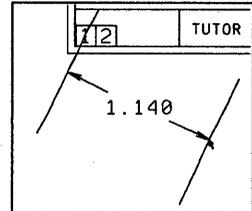
Edit Layout has many options for placing dimensions. Follow these steps to place dimensions on the board:

1. Make sure **Assembly Drawing** displays as the current layer.
2. Place the pointer at (0.7500", 3.5000"). The pointer is below the bottom left corner of the board outline.
3. Select **PLACE Dimension**. The object shown at right displays at the pointer location.



4. Select **Begin** and move the pointer to the right. The dimension value changes as you move the pointer, indicating the distance from where you selected **Begin**. Two end bars and dimension arrows display.

5. Move the pointer down. The dimension pivots on its beginning point but the text remains horizontal, as shown in the example at right.



6. Select **Set**. The **Edit Dimension Text** dialog box displays.

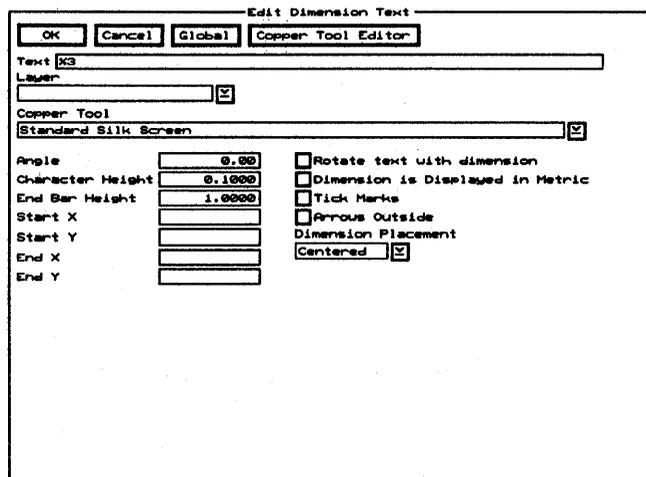
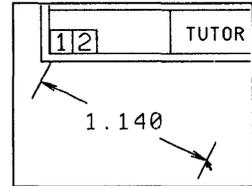


Figure 8-8. The *Edit Dimension Text* dialog box.

7. Enter **0.2500** in the **End Bar Height** entry box and select **OK**. The end bars are shortened from one inch high to a quarter of an inch, as shown at right.



8. Move the pointer up, then to the right. As you move the pointer the current dimension length, its rotation angle, and an alignment aid display in the lower right part of the screen, as shown in the example below.

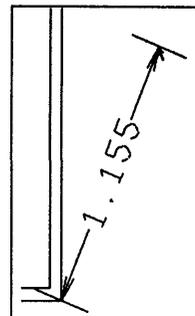
2.980 351.81° X

An "X" displays when the dimension is at an angle other than 0°, 90°, 180°, 270°, or 360°. A "+" displays when the dimension is rotated to one of those five angles.

9. Place the pointer at (7.1250", 3.5000") and select **End** to complete the dimension.
10. Press <Esc> to dismiss the additional dimension text.
11. With the pointer still at (7.1250", 3.5000"), select **MOVE**.
12. Move the dimension text down to (7.1250", 3.5500") and select **Place**. This moves the end bars away from the board outline. The first dimension is complete.

Placing the second dimension

1. Move the pointer to (7.1250", 3.3750") and select **PLACE Dimension Begin**.
2. Select **Set** to display the **Edit Dimension Text** dialog box.
3. Enable **Rotate text with dimension**, then select **OK**. As you move the pointer, the text rotates to follow the angle of the dimension lines. See the example at right.
4. Move the pointer to (7.1250", 0.2500") and select **End**.
5. Press <Esc> to dismiss the additional dimension text.
6. Move the pointer to (7.2500", 0.2500") and select **MOVE**.
7. Move the dimension to the right to (7.4250", 0.2500") and select **Place**.



The completed TUTOR board looks like figure 8-9.

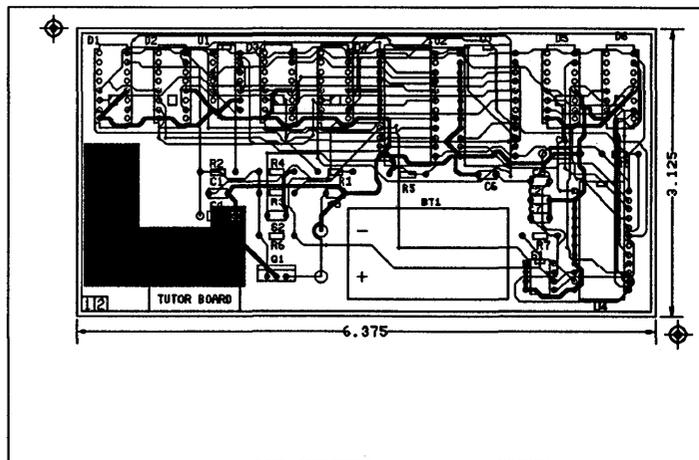


Figure 8-9. The completed TUTOR board.

Saving your work

It is time to save your work again before you go to the next chapter. Select **QUIT Update Board File** to save the file as TUTOR.BD1.



NOTE: After you save your work you can compare your board with DEMO.BD1, which was prepared using the procedures in this tutorial.

Summary

In this chapter you learned how to set routing conditions for individual nets. You learned how to define a sweep window, then applied these settings when you autorouted the TUTOR board. You also learned how to optimize your autorouted design.

The next chapter gives you instructions for printing and plotting the TUTOR board.



Printing and plotting the TUTOR board

About printing and plotting

There are two basic types of output devices you can use with **Edit Layout**: printers and plotters. These devices are categorized by the type of input they require.

If a device accepts *raster* commands, it is a printer. A raster is an array of dots. When you draw a line to a raster device, you must specify each and every dot.

If a device accepts *vector* commands, it is considered a plotter. A vector is a series of points with a specific function defined. For example, a line has a beginning point and an ending point. A circle has a center and a radius.

The device needs to know what the vector information is, but does not need every point along the vector.

You can print working copies of your designs on dot matrix printers and laser printers. The output quality depends on the maximum printer resolution.

You can plot final artwork for your designs on pen plotters, laser photoplotters, and Gerber photoplotters.

Getting started

All plotting functions are contained in **Edit Layout**. You configure a printer driver when you configure **Edit Layout** in the ESP design environment. You must have a printer driver configured before you can print in **Edit Layout**.

See *Chapter 4: Introducing Edit Layout* for information on configuring your printer driver in the ESP design environment.

You produce a print or plot in **Edit Layout** by selecting layers and objects, then assigning those layers and objects to a "page". You can specify how objects will print on each page, and you can print one page at a time or print all pages.

You print the currently loaded board file with these settings. If you want to print another board file using the same settings, load the file into **Edit Layout**. The print settings remain intact for the new file.

Printing

Follow these steps to display the **Printing and Plotting** dialog box:

1. Select **GO TO FUNCTION**. The menu at right displays.
2. Select **Printing and Plotting**. The **Printing and Plotting** dialog box displays (figure 9-1).

Pad Symbol Editor
Via Symbol Editor
Copper Tool Editor
Drill List Editor
Net Property Editor
Library Editor
Autorouter
Netlist Loader
Printing and Plotting
Macro Maintenance

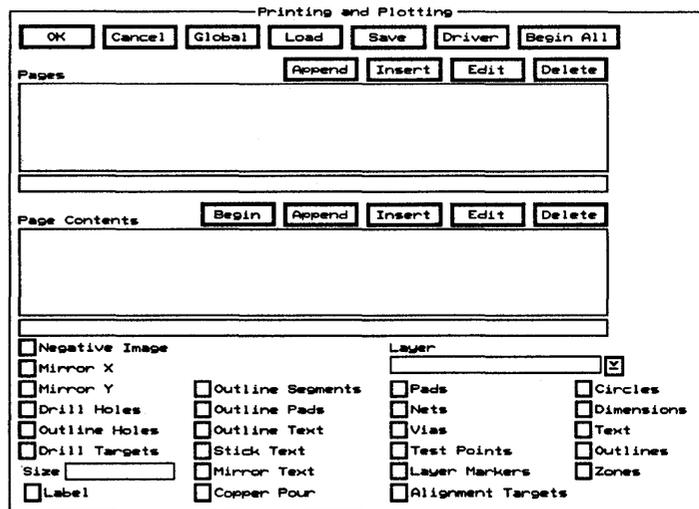


Figure 9-1. The **Printing and Plotting** dialog box.

Configuring printer options

Before you can print your board file you need to configure printer options so **Edit Layout** can find the device. You also need to configure the page settings. Follow these steps:

1. Select **Driver**. The **Driver Configuration** dialog box displays (figure 9-2).

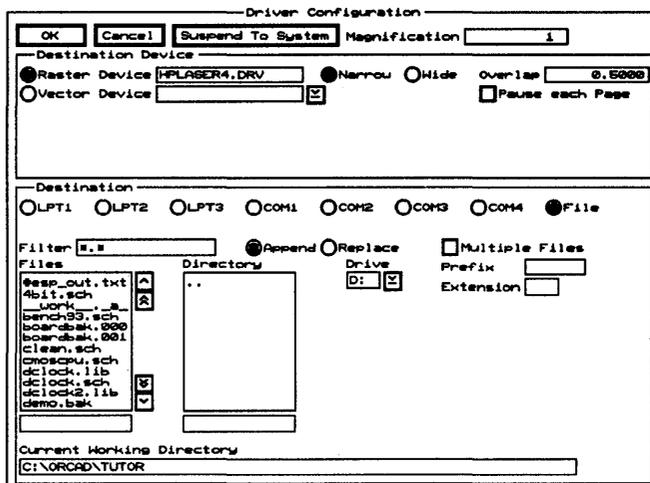


Figure 9-2. The **Driver Configuration** dialog box.

This tutorial assumes you have a laser printer that is assigned to printer port LPT1, printing in portrait mode, and the configured printer driver is HPLASER4.DRV.

The following settings are already configured in the **Driver Configuration** dialog box:

- ❖ **HPLASER4.DRV** displays in the **Raster Device** list box, and is the default selected printing device.
- ❖ **Narrow** is selected as the print orientation in **Destination Device**.
- ❖ **File** is selected in **Destination**, and the contents of the current working directory display in the **Files** list box.
- ❖ **Append** is selected in **Destination**. Multiple print jobs are appended to a single printer file when this option is selected.

2. Select **LPT1** in **Destination** to specify you are printing directly to the printer through LPT1. The **File** options do not display.
3. Leave **Magnification** and **Overlap** at their default settings.
4. Select **OK**. The **Printing and Plotting** dialog box displays.

Configuring pages

When you print a board file in **Edit Layout** you need to consider these factors:

- ❖ The board layers that you want to print
- ❖ The layers and objects that you want to include on each printed page
- ❖ How you want the objects to appear on the page

You define three pages in this chapter: a **TOP COPPER LAYER** page, a **BOTTOM COPPER LAYER** page, and an **ASSEMBLY DRAWING** page.

Building the TOP COPPER LAYER page

Follow these steps to select the layers and objects needed to create the **TOP COPPER LAYER** page:

1. Select the **Layer** droplist button to display the droplist box.
2. Select **Component Copper**.
3. Select the **Component Copper** objects that will appear on the page by enabling these options:
 - ❖ **Pads**
 - ❖ **Nets**
 - ❖ **Vias**
 - ❖ **Layer Markers**
 - ❖ **Alignment Targets**
 - ❖ **Outlines**

4. Select the way the layer marker text appears on the print by enabling these options:
 - ❖ Outline Text
 - ❖ Stick Text

These options print the layer marker text as single line segments, rather than as wide, filled lines.
5. Select **Insert for Page Contents**. The selected options display in the **Page Contents** list box.
6. Place the pointer in the entry box below the **Pages** list box and click the left mouse button. Enter **TOP COPPER LAYER**.
7. Select **Insert for Pages**. **TOP COPPER LAYER** displays in the **Pages** list box. The page name **TOP COPPER LAYER** is associated with the items listed in **Page Contents**. The dialog box looks like figure 9-3.

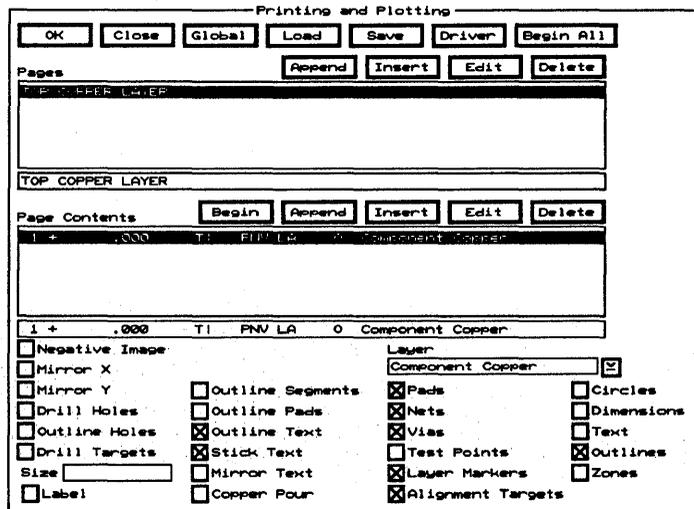


Figure 9-3. The configured TOP COPPER LAYER page.

*Building the BOTTOM
COPPER LAYER page*

The TOP COPPER LAYER page is built, now follow these steps to build the BOTTOM COPPER LAYER page:

1. Select **Delete for Page Contents** to clear the entry in the **Page Contents** list box. The enabled options and selected layer remain unchanged.
2. Select the **Layer** droplist button to display the droplist box.
3. Select **Solder Copper** from the droplist box.
4. Do not disable any of the enabled options. The selections also apply to objects on the Solder Copper layer.
5. Enable **Zones** to print the fill zone that you placed on the Solder Copper layer.
6. Enable **Copper Pour** to print the fill zone as a solid, filled area.
7. Select **Insert for Page Contents**. The selected options display in the **Page Contents** list box.
8. Place the pointer in the entry box below the **Pages** list box and click the left mouse button. **TOP COPPER LAYER** displays in the entry box.
9. Press <Alt><Backspace> to clear the entry box, then enter **BOTTOM COPPER LAYER**.
10. Select **Append for Pages** to add **BOTTOM COPPER LAYER** below **TOP COPPER LAYER** in the **Pages** list box. The items in the **Page Contents** list box are associated with **BOTTOM COPPER LAYER**.

*Building the
ASSEMBLY
DRAWING page*

The BOTTOM COPPER LAYER page is complete, now you need to build the ASSEMBLY DRAWING page, which contains the board outline, module outlines, pads, reference designators, and the outline on the Assembly Layer.

Follow these steps to build the ASSEMBLY DRAWING page:

1. Select **Delete for Page Contents** to delete the entry in the **Page Contents** list box.
2. Select **All Layers** from the **Layer** droplist box.
3. Disable these options:
 - ❖ **Copper Pour**
 - ❖ **Nets**
 - ❖ **Vias**
 - ❖ **Layer Markers**
 - ❖ **Zones**
4. Enable these options:
 - ❖ **Outline Pads**
 - ❖ **Text**
5. Select **Insert for Page Contents** to add the selections to the **Page Contents** list box.
6. Enter **ASSEMBLY DRAWING** in the entry box below the **Pages** list box.
7. Select **Append for Pages** to add ASSEMBLY DRAWING to the bottom of the list of items in the **Pages** list box. The three pages are configured, as shown in figure 9-4.

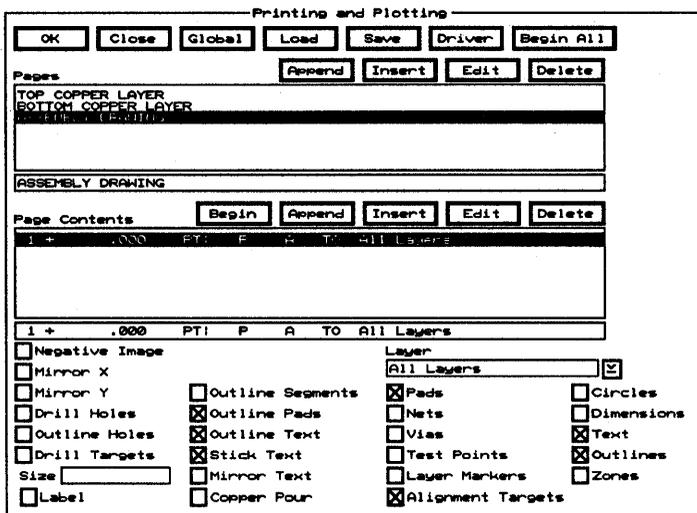


Figure 9-4. The configured printer pages.

Printing pages

Now that you have configured the three pages, you are ready to print them. You can print all pages, or you can print selected pages. Follow the procedures listed below.

Printing all pages

Select **Begin All** to print all pages listed in the **Pages** list box. The pages print in the sequence they display in the list box.

The **Printing and Plotting** dialog box is replaced by a screen displaying the page being printed and the size of the print window. When the first page is done, the next page displays. When all pages are sent to the printer, the **Printing and Plotting** dialog box displays.

Printing selected pages

1. Select **TOP COPPER LAYER** in the **Pages** list box.
The item in the **Page Contents** list box is the page contents for the **ASSEMBLY DRAWING** layer, which was the last page added to **Pages**. You need to replace the item in **Page Contents** with the page contents of **TOP COPPER LAYER**.
2. Select **Edit for Pages**. The item in **Page Contents** changes to the page contents of the **TOP COPPER LAYER** page. The selected layer and enabled options change to reflect the settings of the new item in **Page Contents**.
3. Select **Begin**. Only the page contents of **TOP COPPER LAYER** print.

The three printed pages look like figures 9-5 through 9-7.

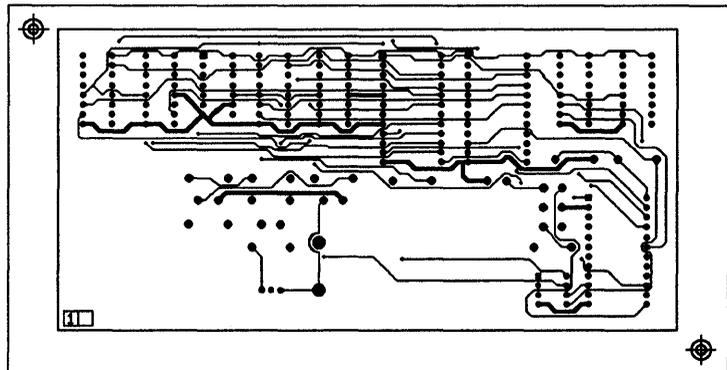


Figure 9-5. The **TOP COPPER LAYER** page.

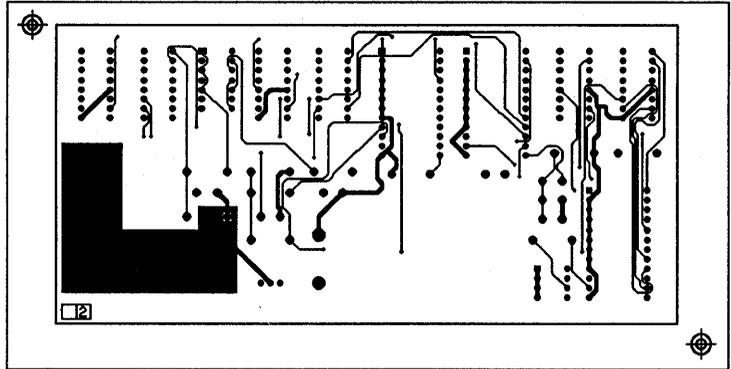


Figure 9-6. The *BOTTOM COPPER LAYER* page

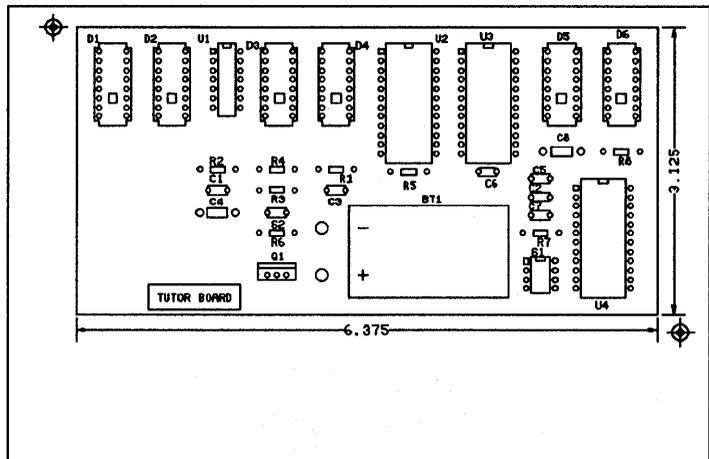


Figure 9-7. The *ASSEMBLY DRAWING* page.

Saving a printer setup

You can save your page configurations to a file, then load the configurations when you want to print the pages. Follow these steps:

1. With the three pages listed in the Pages list box, select **Save**. The **Save Print/Plot Setup to File** dialog box displays (figure 9-8).

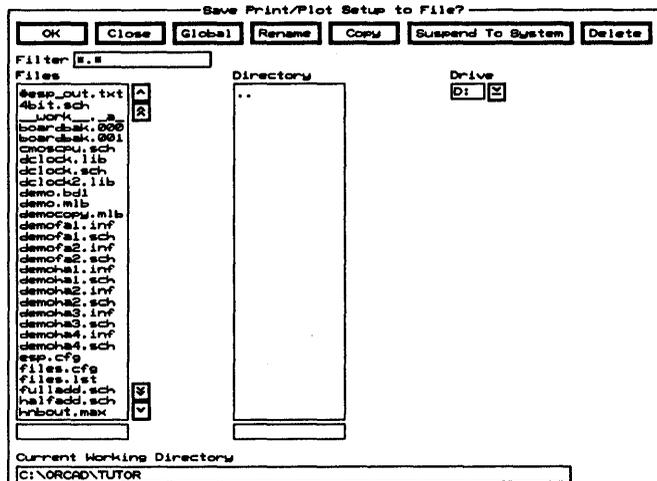


Figure 9-8. The **Save Print/Plot Setup to File** dialog box.

2. Enter **TUTOR386.SET** in the entry box below the **Files** list box.
3. Select **OK**. The page configurations are saved to the file and the **Printing and Plotting** dialog box displays.

Loading a printer setup

Follow these steps to load the previously saved setup file into the **Printing and Plotting** dialog box:

1. With **TOP COPPER LAYER** already highlighted in **Pages**, select **Delete** for **Pages**. The entry is deleted.
2. Select **BOTTOM COPPER LAYER** and select **Delete** for **Pages**, then select **ASSEMBLY DRAWING** and select **Delete** for **Pages**. All entries are deleted from the **Pages** list box.
3. Select **Delete** for **Page Contents** to delete the highlighted entry. The **Pages** and **Page Contents** list boxes are now empty.
4. Select **Load**. The **Load Print/Plot Setup from File** dialog box displays, and **TUTOR386.SET** is selected (figure 9-9).

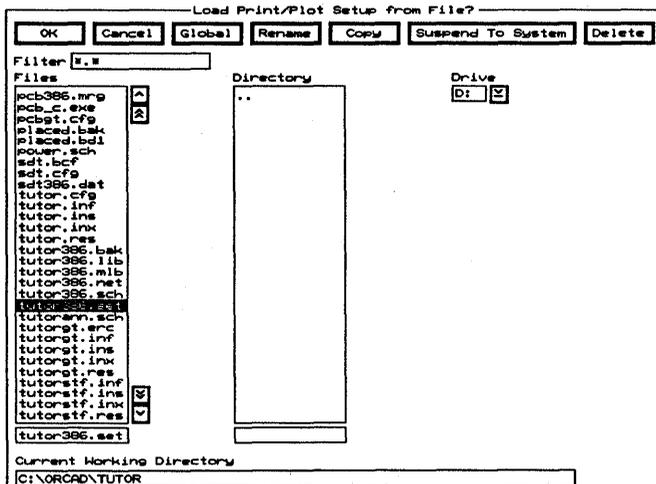


Figure 9-9. The **Load Print/Plot Setup from File** dialog box.

5. Select **OK**. The configured pages load and display in the **Pages** list box.

Plotting

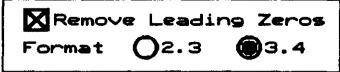
You create high resolution images of your board file pages by sending them to a plotting device, or plotter.

Edit Layout supports the following vector plotting devices:

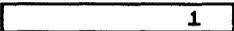
- ❖ Fire 9xxx
- ❖ Gerber (274-D)
- ❖ Gerber (274-X)
- ❖ HP-GL/2
- ❖ Postscript

Refer to the procedures below to plot the three pages to a specific vector device.

Gerber (274-X)

1. Select **Driver**. The **Driver Configuration** dialog box displays.
2. Select **Vector Device**, then select the droplist button to display the droplist box.
3. Select **Gerber (274-X)**. The options shown at right display below the **Vector Device** droplist box.
4. Select a format for the photoplotter.
5. Select the configured printer port in **Destination**, then select **OK** to display the **Printing and Plotting** dialog box.
6. Select **Begin All** to photoplot the three pages.

All aperture information required by the Gerber (274-X) photoplotter is embedded in the board file.

- HP-GL/2**
1. Select **Driver**. The **Driver Configuration** dialog box displays.
 2. Select **Vector Device**, then select **HPGL2** from the droplist box. The **Pen** entry box shown at right displays below the **Vector Device** droplist box. 
 3. Enter a number that corresponds to the pen slot in the carrousel containing the pen you will use to plot the layout.
 4. Select the configured printer port in **Destination**, then select **OK** to display the **Printing and Plotting** dialog box.
 5. Select **Begin All** to plot the three pages.

- Postscript**
1. Select **Driver** to display the **Driver Configuration** dialog box. A postscript printer should be connected to your computer.
 2. Select **Vector Device**, then select **PostScript** from the droplist box.
 3. Select the configured printer port in **Destination**, then select **OK** to display the **Printing and Plotting** dialog box.
 4. Select **Begin All** to print the three pages.



Phar Lap technical information

This appendix provides detailed information about the Phar Lap memory extender.

About the CFG386 utility

All Phar Lap programs support the use of command line switches to override the default operation of the program. The CFG386 utility allows you to customize a Phar Lap program by specifying command line switches to be automatically processed every time the program is run. Judicious use of the CFG386 utility will allow you to avoid having to type commonly used switches every time you run a program.

When a Phar Lap program is run, the program defaults are first set up. Then, any switches configured into the program are processed from left to right. If any conflicting switches are given, the last switch processed takes precedence. Thus, switches configured into a program can be overridden with command line switches.

How to run CFG386

The command line format for CFG386 is the following:

```
CFG386 PCB386.EXE [switches]
```

The first command operand is the name of the .EXE file that you want to configure, in this case, PCB386.EXE. The file must be a Phar Lap program. The operand must follow the standard DOS filename conventions. For example:

```
\MYLI99B\386ASM
```

```
Run386
```

```
A:386LINK.EXE
```

If a filename extension is not specified, then ".EXE" is assumed.

Following the filename, you list one or more command switches. The switches are given in the same format they are given on the command line for the program being configured. CFIG386 determines which Par Lap program is being configured from a program signature in the program's configuration block. The specified switches are added to the configuration block of the program, after any switches that are already there from any previous configurations.

For example:

```
CFIG386 386ASM -NOLIST -8086
CFIG386 RUN386 -MINREAL 100h -MAXREAL 400h
CFIG386 386DEBUG -BO
```

CFIG386 recognizes one switch that alters its own processing. This is the -CLEAR switch. The -CLEAR switch causes CFIG386 to erase the current contents of the program's configuration block. Any switches specified after the clear switch are added to the just-cleared configuration block. For example:

```
CFIG386 386LINK -CLEAR
CFIG386 MINIBUG -CLEAR -CEMM
```

If CFIG386 is run with no switches on the command line, it will display the current contents of the program's configuration block. For example:

```
CFIG386 386ASM
CFIG386: 1.1s--Copyright (c) 1986,1987 PharLap
Software, Inc.
```

```
Phar Lap program type: 386|ASM Version 1.1s
Configured switch values:
  -INCLUDE \INCLUDES\
  -TWOCASE
  -386p
```

△ *NOTE: CFIG386 does NOT check the values of any of the switches or switch parameters which it stores in the program's configuration block. Thus, it is possible to configure invalid switch values into a Phar Lap program. You should ALWAYS run the program after configuring it to make sure that the configured switch values have the desired effect.*

Error messages

Several errors can be reported by the CFIG386 utility. The error messages and their causes are listed below.

Filename too long: filename

The filename specified is too long for an internal CFIG386 buffer.

System Error

An internal error in the CFIG386 utility has been detected. Save a copy of the program which causes the error and contact Par Lap.

Unable to open: filename: reason

An error of some sort prevented filename from being opened. The reason gives more detail as to why the error occurred.

Unable to read configuration for: filename

The .EXE file being configured has a valid Par Lap signature but an invalid switch block of some sort. This error should never occur with a Phar Lap program. Save a copy of the program and contact Par Lap.

Out of configured space on switch: switchname

The internal switch buffer in the program being configured has overflowed. Switch switchname and all following switches will not be added to the default switch table.

Unable to save new configured switch values in executable file

The program file is either write-locked or has been corrupted between the time the switch block was read from the file and when it was written back.

Unable to read config block from: filename

The .EXE file being configured has a valid Phar Lap signature but an invalid switch block of some sort. This error should never occur with a Phar Lap program. Save a copy of the program and contact Par Lap.

Not a Phar Lap program: filename

The specified .EXE file does not have one of the Phar Lap program signatures that the CFG386 utility expects to find in the program's configuration block.

386 | DOS-Extender command line switches

Command line switches are used to change the default operation of 386 | DOS-Extender. By default, 386 | DOS-Extender:

- ❖ Uses all memory below 640K for your application and leaves no free real mode memory for other applications.
- ❖ Allocates four kilobytes for the data buffer used on DOS and BIOS function calls.
- ❖ Allocates four buffers of one kilobyte each to be used for stack memory when switching from protected mode to real mode.

Command line switches begin with a minus sign ("-") character, followed by the name of the switch. There are two forms of each switch name: a long form and a short form. Any argument to the switch must immediately follow the switch name, with a space as a separator. If conflicting switches are given on a command line, the last (right-most) switch takes precedence.

Some of the command line switches take a number as an argument. By default, the number is considered to be a decimal (base 10) number.

Hexadecimal (base 16) numbers may be specified by appending the character "h" or "H" to the number. The following two examples both give the same number as an argument to the switch -MAXREAL:

```
run386 -maxreal 200h hello
run386 -maxreal 512 hello
```

386 | DOS-Extender switches may be specified in three different ways: (1) some switches may be given when the program is linked, (2) switches may be configured into the 386 | DOS-Extender task image (RUN386.EXE) using the CFG386 utility, and (3) switches may be entered on the command line, when the program is actually run. The link time switch settings are processed first, then configured in switches, and, last, the command line switches. If conflicting switch settings are given, the last switch processed takes precedence.

Conventional memory switches

The -MINREAL and -MAXREAL switches are used to control how much conventional memory (memory below 640K) is left free by 386 | DOS-Extender. By default, 386 | DOS-Extender allocates all the available conventional memory for use by the application program.

If the application program you are executing ever makes a system call to execute another program, you must make sure that 386 | DOS-Extender leave sufficient conventional memory free for the second program's needs. The -MINREAL switch specifies the minimum amount of conventional memory to leave free; 386 | DOS-Extender refuses to run the program if it cannot leave at least this amount of memory free. The -MAXREAL switch specifies the maximum amount of conventional memory to leave free. 386 | DOS-Extender guarantees that at least MINREAL memory is left free and that as much as possible, up to MAXREAL memory, is left free.

The -MINREAL and -MAXREAL switches both take a number as an argument. The number specifies memory size in units of 16-byte paragraphs (the standard unit of memory allocation under MS-DOS).

The number must be less than or equal to 65535 (FFFFh). By default, 386 | DOS-Extender sets both -MINREAL and -MAXREAL to zero. These switches may also be specified at program link time.

Syntax

-MINREAL nparagraphs
-MAXREAL nparagraphs

Short form

-MINR nparagraphs
-MAXR nparagraphs

Examples

```
run386 -minreal 100h hello  
run386 -minr 128 -maxr 512 hello
```

**Systems Call Data
Buffer Switches**

The -MINIBUF and -MAXIBUF switches are used to control how much memory is allocated to the data buffer used for DOS and BIOS function calls. This buffer is most important for file I/O; if your program reads or writes large amounts of data at a time, you should allocate a large buffer for efficiency.

The -MINIBUF and -MAXIBUF switches both take a number as an argument. The number specifies the buffer size in units of one kilobyte and must be between one and 64, inclusive. By default, 386|DOS-Extender sets MINIBUF to one kilobyte, and MAXIBUF to four. If 386|DOS-Extender cannot allocate at least MINIBUF kilobytes for the interrupt buffer, it refuses to run the program. If possible, MAXIBUG kilobytes are allocated. If there is not enough memory available to satisfy both the MAXREAL and MAXIBUF parameters, MAXIBUF takes precedence. These switches may also be specified at program link time.

Syntax

-MINIBUF nkilobytes

-MAXIBUF nkilobytes

Short form

-MINI nkilobytes

-MAXI nkilobytes

Examples

```
run386 -maxibuf 2 hello
```

```
run386 -mini 64 filecopy
```

Mixed mode program switches

The `-REALBREAK` and `-CALLBUFS` switches are used to control program loading of programs that contain both real mode and protected mode code.

The `-REALBREAK` switch controls how much of the program must be loaded into conventional memory, so that it can be accessed and/or executed in real mode. It takes an argument specifying the number of bytes at the beginning of the program, which must be loaded in conventional memory. This switch may also be specified at link time. It is usually more convenient to specify this switch at link time, when the argument can be the name of a public symbol appearing at the end of the real mode code and data. If this switch is used at run time, the argument must be an absolute number which has been calculated from the information in the link map.

The `-CALLBUFS` switch controls the size of the intermode call buffer, which is allocated in conventional memory for use by the application program as a data buffer on intermode procedure calls. The buffer address is obtained at run time with a 386 | DOS-Extender system call. The argument is the size of the buffer in kilobytes and must be less than or equal to 64. The default buffer size is zero. This switch may also be specified at link time.

Syntax

`-REALBREAK nbytes`
`-CALLBUFS nkilobytes`

Short form

`-REALB nbytes`
`-CALLB nkilobytes`

Examples

```
run386 -realbreak 200h -callb 2 switch  
386link switch -realb END_REAL -callbufs 2
```

Stack allocation switches

The `-NISTACK` and `-ISTKSIZE` switches are used to control how much memory is allocated to the buffers used to provide stack space when switching the 80386 from protected mode to real mode. For the vast majority of application programs, the default settings of these parameters are sufficient.

Both switches take a number as an argument. The `-NISTACK` switch specifies the number of stack buffers to allocate and must be four or greater. The `-ISTKSIZE` switch specifies the size of each stack buffer in kilobytes and must be between one and 64, inclusive. By default, 386 | DOS-Extender allocates four stack buffers of one kilobyte each. These switches may also be specified at program link time.

Syntax

```
-NISTACK nbuffers  
-ISTKSIZE nkilobytes
```

Short form

```
-NI nbuffers  
-ISTK nkilobytes
```

Example

```
run386 -ni 6 -istk 2 switch.exp
```

Extended memory switches

The -EXTLOW and -EXTHIGH switches are used to limit the amount of extended memory (memory above one megabyte) that 386|DOS-Extender allows the application program to use. By default, all extended memory that is not allocated to other programs is available for use by the application. Other programs which may have allocated extended memory include RAM disk programs, disk cache programs, and EMS simulators.

Both the -EXTLOW and -EXTHIGH switches take a number as an argument. The number specifies a physical memory address in extended memory. By default, 386|DOS-Extender sets EXTLOW to 100000h (one megabyte) and EXTHIGH to FFFFFFFFh (four gigabytes).

386|DOS-Extender uses only extended memory above the address specified with the -EXTLOW switch, or memory used by other programs, whichever is higher. Similarly, it uses only extended memory below the address specified with the -EXTHIGH switch, or memory used by other programs, whichever is lower.

Normally, it is not necessary to use the -EXTLOW or -EXTGHIGH switches. However, if your system has a program installed that uses extended memory and does not use either (1) the VDISK or RAMDRIVE standards for allocating memory from one megabyte up, or (2) the INT 15th function 88h BIOS call for allocating extended memory from the top of memory down, it may be necessary to use one or both of these switches to prevent 386|DOS-Extender from allocating extended memory used by the installed program.

Syntax

-EXTLOW address
-EXTHIGH address

Short form

-EXTL address
-EXTH address

Examples

```
run386 -extlow 200000h hello  
run386 -extl 180000h -exth 400000h hello
```

Weitek 1167 switch The -1167 switch is used to select how detection of the Weitek 1167 floating point coprocessor is performed. If 386 | DOS-Extender detects the presence of the 1167, segment selector 003Ch is initialized to map the memory space used by the 1167 and segment register FS is initialized to contain selector 003Ch. A program can, therefore, test for the presence of the 1167 at run time by examining the contents of the FS register.

The -1167 switch has three settings. The "-1167 AUTO" setting instructs 386 | DOS-Extender to use the Weitek-approved BIOS presence detection call. This may not work correctly on all machines, if the appropriate BIOS is not installed. The setting "-1167 ON" instructs 386 | DOS-Extender to assume the 1167 is present, and setting "-1167 OFF" assumes the 1167 is not present. The default setting is "-1167 AUTO."

Syntax -1167 AUTO
 -1167 ON
 -1167 OFF

Short form -1167 AUTO
 -1167 ON
 -1167 OFF

Example run386 -1167 on float.exp

Interrupt relocation switches

The -HWIVVEC and -PRIVVEC switches are used to select the interrupt vectors to be used for interrupts, which 386/DOS-Extender must relocate due to compatibility problems between the 80386 processor exceptions and PC/AT-compatible interrupts.

The -HWIVVEC switch selects a block of eight interrupt vectors to use for hardware interrupts IRQ0 through IRQ7. These interrupts must be relocated, because they are normally vectored through interrupts 08h-0Fh, which are also used for processor exceptions. The switch argument is the interrupt vector number to use for hardware interrupt IRQ0. The default, if no command line switch is used, is interrupt vector 78h. This switch is not available when executing under the DESQview 386 environment, since DESQ view also relocates hardware interrupts.

The -PRIVVEC switch selects the interrupt vector to use for the BIOS print screen function call. This interrupt must be relocated, because it normally uses vector 05h, which is also used for the processor bounds exception. The default setting for this switch is interrupt vector 80h.

Syntax -HWIVVEC vector
 -PRIVVEC vector

Short form -HWI vector
 -PRI vector

Example run386 -hwivvec 50h -pri 78h hello

Interrupt mapping switches

The -INTMAP and -PRIMAP switches prevent 386 | DOS-Extender from relocating any interrupt vectors, and to specify where the relevant interrupt vectors are already mapped. The primary purpose of these switches is to allow protected mode memory-resident (TSR) programs to be installed. A secondary purpose is compatibility with other programs which relocate interrupts.

The -INTMAP switch disables 386 | DOS-Extenders' remapping of hardware interrupts, and specifies the block of eight interrupt vectors to which hardware interrupts IRQ0 through IRQ7 are already mapped. When running under DOS, these interrupts are normally mapped to vectors 08h - 0Fh, but it is possible for other programs to relocate them. If the -INTMAP 8 switch is used, 386 | DOS-Extender (and a debugger, if one is being used) does not take over processor exceptions 08h - 0Fh, because it assumes these interrupt vectors are being used for hardware interrupts. The -INTMAP 8 switch should ONLY be used with TSR (or other) programs that have already been debugged, since a processor exception in a buggy program run with this switch would be interpreted as a hardware interrupt, causing the machine to crash.

The -PRIMAP switch disables 386 | DOS-Extender's remapping of the BIOS print screen function call, and specifies the interrupt vector to which this call is already mapped. The print screen function is normally invoked through INT 5. If the -PRIMAP 5 switch is used, 386 | DOS-Extender (and a debugger, if one is being used) does not take over processor exception 5, the BOUND exception. If a buggy program which causes a BOUND exception is run with the -PRIMAP 5 switch, a print screen will occur instead of an abnormal program termination.

Syntax -INTMAP vector
 -PRIMAP vector

Short form -INTM vector
 -PRIM vector

Examples run386 -intm 8 -primap 5 kbdtrap
 run386 -intmap 78h -prim 80h program2

Paging disable switch

The -NOPAGE switch is used to prevent 386 | DOS-Extender from using the 80386's hardware paging functionality to perform memory management. The ONLY reason to use this switch is to avoid a bug in early versions of the 80386 chip. Chip steps B1 and earlier have the bug. Chip step D0, which does not have the problem, was released by Intel in the second quarter of 1988. The problem only appears in programs which use the 80387 numeric coprocessor, and manifests itself as the machine halting, with not even the DOS system reboot command, CTRL-ALT-DEL, available. This problem does not occur on all 80386 machines (e.g., the Compaq DESKPRO 386/20), because it can be affected by the design of the system motherboard.

When this problem occurs in a program which uses floating point arithmetic, the only workarounds available are: (1) to simulate floating point operations in software to avoid using the 80387 coprocessor, (2) to install a step D0 or later 80386 chip in the machine, or (3) to disable paging, which removes one of the hardware conditions necessary for the problem to occur. The -NOPAGE switch disables paging.

There are disadvantages to using the -NOPAGE switch. Programs run with this switch cannot be linked with the -OFFSET switch. Programs run with -NOPAGE are loaded entirely in extended memory, with conventional DOS memory (below 640 K) not available. In addition, the dynamic memory allocation system services provided by 386 | DOS-Extender are disabled when this switch is used. This means that the program must specify at link time (using the -MINDATA and -MAXDATA linker switches) how much memory it needs at run time. Heap memory allocation performed by compiler run-time libraries is normally done out of memory allocated under the control of the linker -MAXDATA switch, and is not affected by the use of -NOPAGE. The system memory allocation calls which are disabled by the use of this switch are INT 21h functions 48h, 49h, 4Ah, and 250Ah.

Syntax -NOPAGE

Short form -NOP

Example run386 -nopage numcrunch

**Compaq built-in
memory switch**

The -NOBIM switch is used to disable the automatic use of Compaq built-in memory. By default, 386 | DOS-Extender attempts to use built-in memory mapped above 14 megabytes on Compaq 386 machines, if the memory is not allocated to another program. This switch instructs 386 | DOS-Extender not to check for Compaq built-in memory. Normally, it is not necessary to use this switch.

Syntax -NOBIM

Short form -NOB

Example run386 -nobim hello

**VDISK compatibility
switch**

The -VDISK switch is a workaround for compatibility problems with other programs which do not correctly follow the VDISK standard for allocating extended memory.

If 386 | DOS-Extender refuses to run an application program because of inconsistent VDISK allocation signatures, this switch can be used to force 386 | DOS-Extender to run the program. The larger of the two allocation marks present will be used. Before using this switch, you should check the allocation sizes printed out with the error message when 386 | DOS-Extender refuses to run the program. If the larger of the two numbers printed out does not seem reasonable, it will be necessary to calculate how much extended memory is in use by other programs and to use the -EXTLOW switch to inform 386 | DOS-Extender of the correct value.

Syntax -VDISK

Short form -VDISK

Example run386 -vdisk hello

80386 step B0 switch

The -B0 switch is used to enable operation of 386|DOS-Extender on a system that has a step B0 80386 chip. 386|DOS-Extender is able to run only on 80386 chips that are step B0 or later because of a bug in earlier chips that did not permit the processor to be switched from protected mode back to real mode.

386|DOS-Extender can only check at run time whether the 80386 chip is step B1 or later. By default, it will refuse to run the application program if the chip is earlier than step B1. The -B0 switch can be used to force 386|DOS-Extender to run. Note that if the -B0 switch is used with a chip that is earlier than step B0, the system will crash.

Syntax -B0

Short form -B0

Example `run386 -B0 hello.exp`

EMS simulator switch The -CEMM switch is used to turn off the COMPAQ CEMM or compatible EMS (Lotus/Intel/Microsoft Expanded Memory Specification) simulator programs. These EMS simulators operate in the 80386's Virtual 8086 Mode.

386 | DOS-Extender normally cannot run in Virtual 8086 Mode and, by default, will refuse to run the application program if the 80386 is in virtual mode and if the VCPI interface provided by the Quarterdeck QEMM and some other programs is not present. The -CEMM switch can be used to have 386 | DOS-Extender automatically disable the EMS simulator program and switch the 8086 back to real mode. Note that the same thing can be accomplished by manually disabling the EMS simulator before running 386 | DOS-Extender. If the -CEMM switch is used when some program other than a Compaq-compatible EMS simulator has switched the 80386 into virtual mode, 386 | DOS-Extender prints an error message and refuses to run the program.

Syntax -CEMM

Short form -CEMM

Example run386 -CEMM hello

Address line 20 switch

The -A20 switch is used to control how address line 20 is enabled or disabled. 80386 systems that conform to the IBM PC/AT standard have hardware either to allow full 32-bit addressing ("enable A20") or to truncate addresses to 20 bits ("disable A20"). When executing in real mode, A20 is normally disabled for compatibility with programs that take advantage of the address space wrap-around occurring at one megabyte on 8088/8086 systems. Very few programs rely on this behavior; the most common example is copy protection programs.

By default, 386 | DOS-Extender enables A20 before starting the application running, and restores the original A20 setting when the program terminates. The -A20 switch can be used to force 386 | DOS-Extender to disable A20 each time the 80386 is switched to real mode, and to re-enable A20 each time the 80386 is switched to protected mode. This can be important if, for example, a software diver, which can gain control at any time via a hardware interrupt, and which relies on one-megabyte addressing wrap-around, is installed on your machine.

There is a penalty associated with the -A20 switch. Depending on the hardware in your system, it can take several milliseconds to enable or disable A20. Thus, using the -A20 switch slows down the switch to 80386 real mode, then back to protected mode, that occurs whenever there is a hardware interrupt or DOS or BIOS function call.

Syntax -A20

Short form -A20

Example run386 -A20 hello

**PC and PC/XT
detection switch**

The -XT switch is used to inform 386|DOS-Extender that it is executing on a IBM-Compatible PC or PC/XT with a 386 board, such as the Intel Inboard/PC, installed. 386|DOS-Extender normally detects such configurations automatically, but it may not be able to detect systems which do not have the IBM standard system ID byte in the BIOS. If 386|DOS Extender does not correctly detect a PC environment, this switch can be used to allow the program to execute successfully.

Syntax -XT

Short form -XT

Example run386 -xt hello

386 | VMM command line switches

Command line switches are used to change the default operation of 386 | DOS-Extender, and of 386 | VMM. This section documents the 386 | DOS-Extender command line switches that apply specifically to operation with virtual memory. These switches can be used with 386 | DOS-Extender even if 386 | VMM is not used. Except where noted, the 386 | VMM-specific switches are ignored by 386 | DOS-Extender when used without virtual memory. By default, 386 | VMM:

- ❖ Places the swap file used for paging in the root directory of the device from which the application program was loaded.
- ❖ Always increases the swap file size when the program allocates additional virtual memory.
- ❖ Uses a least-frequently-used algorithm for selecting the page to be replaced (written to the swap file), when bringing another page into memory.
- ❖ Updates the virtual page aging information (used by the page replacement algorithm) every four seconds.

Command line switches begin with a minus sign ("-") character, followed by the name of the switch (e.g., -LFU). There are two forms of each switch name: a long form and a short form. Any argument to the switch must immediately follow the switch name, with a space as a separator (e.g., -VSCAN 4000). If conflicting switches are given on a command line, the last (right-most) switch takes precedence.

Some of the command line switches take a number as an argument. By default, the number is considered to be a decimal (base 10) number. Hexadecimal (base 16) numbers may be specified by appending the character "h" or "H" to the number. The following two examples both give the same number as an argument to the switch "-VSCAN":

```
run386 -vscan 2048 -vmfile vmmdrv bigsort
run386 -vscan 800h -vmfile vmmdrv bigsort
```

Virtual memory driver switches

The -VMFILE switch is used to specify the name and location of the development version of 386|VMM to be loaded by 386|DOS-Extender during initialization. Using this switch causes the application program to run in a virtual memory environment.

The -VMFILE switch is normally not used with bound applications (programs which have the redistribution versions of 386|DOS-Extender and 386|VMM bound to them in a single .EXE file). If the -VMFILE switch *is* used with a bound application, the virtual memory driver specified with the switch will be loaded, instead of the virtual memory driver that is bound to the application. This can be useful for testing a bound application with a later release of 386|VMM.

The -NOVM switch instructs 386|DOS-Extender not to load a virtual memory driver, regardless of whether the -VMFILE switch was used, or whether 386|VMM is bound to the application program. It causes the program to run in a non-virtual environment.

Syntax -VMFILE filename
 -NOVM

Short form -VM filename
 -NOVM

Examples run386 -vmfile vmmdrv hello
 386debug -vm \pharlap\vmmdrv.exp hello
 cfig386 hello.exe -novm

Swap file location switch The **-SWAPDIR** switch specifies the device and directory in which to place the page swap file. The default location for the swap file is the root directory of the device from which the application program was loaded. This switch is useful for placing the swap file on a device which has sufficient free space to allow the swap file to grow as needed.

The directory name specified with this switch must not end with a "\" character, as 386|VMM appends a "\" before adding the name of the swap file. The swap file is created with a unique filename (using the DOS Create Temporary File system call).

Syntax **-SWAPDIR** filename
 -SWD dirname

Short form **-VM** filename
 -NOVM

Examples run386 -swapdir d: -vm vmndrv hello
 minibug -swd e:\tmp -vm vmndrv hello

Page replacement policy switches 386|VMM supports two switch-selectable page replacement policies. The page replacement policy defines the algorithm used to select a page to be swapped to disk when a page already on disk needs to be brought into memory. The performance of a program in a virtual memory environment depends to some extent on whether the system usually replaces pages that are not needed for a long time; ideally, the page selected for replacement is the page not referenced by the program for the longest time into the future. Depending on the memory referencing patterns of an application, one of the page replacement algorithms supported by 386|VMM may yield better performance than the other.

The `-LFU` switch selects the Least-Frequently-Used replacement policy. A reference frequency count is kept with each page. Periodically, the page tables are scanned, and the count is either incremented or decremented, depending whether the page was referenced since the last scan. The page with the lowest count (the least-frequently-used page) is the page selected for replacement. This is the default page replacement policy if no switches are used.

The `-NUR` switch selects the Not-Used-Recently replacement policy. This algorithm chooses a page for replacement based on whether the page has been accessed by the program, and whether it is dirty (its contents have been modified). Periodically, the page tables are scanned to mark all pages not accessed. The page accessed information thus identifies pages which have been referenced recently (since the last page table scan).

The `-VSCAN` switch selects how frequently the page tables are scanned in order to update the page aging information used by the page replacement policy. Changing the scan period affects which pages are selected for replacement, and therefore, affects program performance. The `-VSCAN` switch takes as an argument a time expressed in milliseconds (ms). The minimum value which may be given is 1000 ms (1 second). The default scan period is 4000 ms. (Note that 386 | VMM assumes the timer tick interrupt occurs 18.2 times per second; application programs which change this standard timer operation must adjust the value specified with the `-VSCAN` switch appropriately).

Syntax

```
-LFU  
-NUR  
-VSCAN nmilliseconds
```

Short form

```
-LFU  
-NUR  
-VS nmilliseconds
```

Examples

```
run386 -lfu -vm vmmdrv hello  
minibug -nur -vm vmmdrv hello  
386debug -nur -vscan 2000 -vm vmmdrv hello
```

**Swap file growing
policy switches**

The page swap file can potentially grow very large, if the virtual address space required by the program is large. Under these circumstances, it is possible to run out of disk space. The tradeoff is using up more disk space than is actually needed versus taking the risk of running out of swap space during a page fault, in which case 386 | VMM is forced to abort the program.

The `-SWAPCHK` switch is used to select when the size of the swap file is increased by 386 | VMM. The `-SWAPCHK MAX` setting causes the swap file to grow whenever the virtual address space of the program is increased. The size of the swap file is always set to the size of the program's virtual address space, which is the largest size that could possibly be needed. If the swap file cannot be grown when a memory allocate system call is made, the memory allocate call returns failure, so the program can deal with the condition gracefully. This is the safest setting, because it guarantees that 386 | VMM will always have swap space available when a page fault occurs. It does, however, result in the largest swap file. (Remember that disk space problems can sometimes be solved by placing the swap file on a different disk drive with the `-SWAPDIR` switch.)

The default setting is `-SWAPCHK FORCE`, which still causes the swap file to grow whenever additional virtual memory is allocated. However, the size to which it is increased is smaller than the virtual address space for the program, while still large enough to guarantee that sufficient swap space is available when a page fault occurs. This setting is a good compromise. It results in a smaller swap file, but ensures that no unexpected program aborts will occur. However, if this setting is used and too large a value is specified by the `-CODESIZE` switch, it can result in an out of swap space condition during a page fault.

The `-SWAPCHK ON` setting does not grow the swap file when virtual memory is allocated; instead, the swap file size is increased by the page fault handler as it needs new swap space. When additional virtual memory is allocated, the amount of free space on the disk is checked to make sure there is sufficient free space, using the same swap space requirements as those imposed by the `-SWAPCHK FORCE` setting. However, the swap file is not actually grown until the space is needed. This setting minimizes the size of the swap file; but if the program uses up disk space for another purpose between the time the memory allocate is performed and the swap file needs to be grown, a fatal out-of-swap-space error may occur in the page fault handler. In addition, there will be some performance degradation, because it is more expensive to grow the swap file one page at a time than to grow it in large chunks when additional virtual memory is allocated. For these reasons, it is normally better to use the `-SWAPCHK FORCE` setting.

The `-SWAPCHK OFF` setting disables all swap space checking when virtual memory is allocated. The swap file is grown as needed when a page fault occurs. As with `-SWAPCHK ON`, this minimizes swap file size, but leaves the program vulnerable to out-of-swap-space fatal errors when a page fault occurs. If this setting is used, the program should install an out-of-swap-space handler that attempts either to create more swap space, or clean up and exit, when this condition occurs.

The `-CODESIZE` switch specifies the number of bytes of code which can be paged to disk without seriously affecting program performance. It is equal to the total size of the program's code, in bytes, minus the program's code "working set," that is, the amount of code that needs to be in memory at any given time to avoid excessive paging. This information is used with the `-SWAPCHK FORCE` and `-SWAPCHK ON` settings to calculate the minimum swap space required. Increasing the value reduces the swap file size. Specifying too large a value with this switch may result in unacceptable program performance, or even in fatal out-of-swap-space errors.

The `-SWFGROW1ST` and `-NOSWFGROW1ST` switches specify what the page fault handler should do when it needs a page in the swap file and one is not available. `-SWFGROW1ST` is the default, and causes the page fault handler, first, to attempt to grow the swap file, and then if that fails, to attempt to take a swap file page away from a virtual page currently in memory (this can be done because a page in memory does not need space in the swap file). The `-NOSWFGROW1ST` setting reverses the order; it causes a swap page to be taken away from an in-memory page first, and the swap file to be grown only if no in-memory page owns a page in the swap file. The tradeoff is performance versus disk space. Disk space requirements are reduced if the `-NOSWFGROW1ST` switch is used, but program performance suffers.

The `-MAXSWFIZE` switch is used to limit the maximum disk space that is allocated to the swap file. It specifies a size, in bytes, beyond which the swap file is never increased. If this switch is not used, the only upper bound on swap file size is the amount of free space available on the disk.

Syntax

```
-SWAPCHK OFF
-SWAPCHK ON
-SWAPCHK FORCE
-SWAPCHK MAX
-SWFGROW1ST
-NOSWFGROW1ST
-CODESIZE nbytes
-MAXSWFSIZE nbytes
```

Short form

-SWC OFF
-SWC ON
-SWC FORCE
-SWC MAX
-SWFG
-NOSWFG
-CODES nbytes
-MAXS nbytes

Examples

run386 -swapchk force -vm vmmdrv hello
run386 -swc off -noswfg -vm vmmdrv hello
minibug -swc on -codes 900 -vm vmmdrv hello
run386 -maxs A00000h -vm vmmdrv hello

Special characters

<Ctrl> key 16
 = BOOKMARK command 65, 168
 ? CONDITIONS command 180, 183, 185
 - LAYER command 61
 % MACRO command 76, 79
 * LAYER command 61, 93
 + LAYER command 61
 / OTHER command 61
 1167 switch
 weitek 217
 386 | DOS-Extender
 command line switches 211
 386 | VMM
 command line switches 226
 80386
 step B0 switch 222
 90-degree corners
 drawing 113, 117

A

about
 autorouting 175
 CFG386 utility 207
 IFORM 37
 ILINK 36
 INET 35
 layout placement 129
 library editor 88
 manual routing 155
 module placement 135
 modules 87
 printing and plotting 193
 adding
 arcs 114
 holes 121
 additional processing 185
 address
 line 20 switch 224
 aids, placement 135
 alignment targets, placing 152

all macros, deleting 84
 all pages, printing 200
 allocation, stack, switches 215
 Annotate Schematic 24
 any layer, deleting objects 105
 arc segments, routing 161
 arcs
 adding 114
 positioning 116
 ASSEMBLY DRAWING page, building 199
 assembly outline, placing 187
 assigning a net 150
 autoroute method, setting 179
 Autoroute Options dialog box 179
 autoroute zone, placing 176
 autorouter options, setting 179
 autorouting
 about 175
 begin 183
 board 180
 preparing 175
 section of board 180
 whole board 182
 Available Display Drivers 43
 Available Printer Drivers 43

B

- basics, mouse 16
- BLOCK command 49, 99, 100
- board
 - autorouting 180
 - file prefix 45
 - identification 146
 - modules, creating 9
 - name, placing 147
 - objects, placing 145
 - outline, drawing 131
 - routing 159
 - saving and backing up 71
- bookmark
 - creating 65
 - deleting 67
 - jumping 66
 - using 65
- BOOKMARK command 67
- Bookmark dialog box 65
- BOTTOM COPPER LAYER page, building 198
- boxes
 - check 52
 - dialog 50
 - droplist 52
 - entry 53
 - list 51
- building, page
 - ASSEMBLY DRAWING page 199
 - BOTTOM COPPER LAYER 198
 - TOP COPPER LAYER 196
- built-in memory switch, compag 221
- buttons 50

C

- CFIG386 utility, about 207
- changing
 - grid color 70
 - origin 68
 - startup design 20
 - track path 172
 - track width 173
 - TUTOR design 19
 - view of display 113
 - view of the layout 62
- check box 52
- Check Design Integrity, running 34
- Check Electrical Rules 31
 - configuring 33
- checking design integrity 31
- Cleanup Schematic 31
 - configuring 31
- command line switches
 - 386 | DOS-Extender 211
 - 386 | VMM 226
- command prompt, operating system 17
- commands
 - % MACRO 76, 79
 - * LAYER 61, 93
 - + LAYER 61
 - / OTHER 61
 - = BOOKMARK 65, 168
 - ? CONDITIONS 180, 183, 185
 - BLOCK 49, 99, 100
 - BOOKMARK 67
 - CUT 117
 - DELETE 105, 117, 169, 171
 - EDIT 95, 150, 173
 - EXIT 75
 - FIND 156, 160, 161
 - GO TO FUNCTION 80, 88, 92, 179
 - HIGHLIGHT 159, 161
 - INQUIRE 93, 99, 165
 - JUMP 65, 66, 68, 165
 - LAYER 60, 165
 - MOVE 96, 97, 98

- ORIGIN 68, 76, 79, 113, 119
- PLACE 77, 78, 79, 96
- QUIT 24, 31, 71, 72, 73, 74, 75, 91
- ROUTE 155, 159, 160, 161, 172
- SELECTIVE 106, 107
- SET 47, 56, 69, 70, 76, 96, 130, 148
- TRACK DELETE 171, 172
- UNDELETE 106, 171
- VERBOSE INQUIRE 94
- WINDOW ZOOM 63, 148
- X SHOW RATSNEST 136, 157
- ZOOM 47, 62, 63, 64, 76
 - LAYER 61
- compag built-in memory switch 221
- Compare Netlist 8
- compatibility switch, vdisk 221
- conditions, setting 159
- configuration
 - minimum 3
 - saving 71
- Configured Display Drivers 43
- Configured Printer Drivers 43
- configuring
 - DRAFT 22
 - Edit Layout 46
 - IFORM 37
 - ILINK 36
 - INET 35
 - key fields 24
 - pages 196
 - PC Board Layout Tools 42
 - printer options 195
 - Schematic Design Tools 22, 23
 - virtual memory 3
- conventional memory switches 212
- coordinate placement 138
- Copper Colors/Enables/... dialog box 58
- copper tool
 - creating 158
 - specifying 178
- Copy File dialog box 73
- copying
 - DEMO library 89
 - file 73
 - module 89
- correcting DRC errors 168
- Create NC Drill File 7
- creating
 - board module 9
 - bookmark 65
 - copper tool 158
 - fill zone 148
 - first macro 76
 - module 111
 - netlist 35
 - second macro 79
 - stuff file 28
- Cross Reference Parts 31
 - configuring 32
- Current Object Settings dialog box 59
- current settings 59
- CUT command 117

D

- data buffer switches, systems calls 213
- defined macro, running 85
- defining a netlist block 132
- DELETE command 105, 117, 169, 171
- deleted objects, permanently deleting 107
- deleting
 - 90° corner 117
 - bookmark 67
 - exported modules 110
 - file 74
 - macro file 84
 - module objects 105
 - objects 105
 - offending track 169
 - stub 171
 - track 171
- DEMO library, copy 89
- design
 - environment 18
 - integrity, checking 31
 - size 2
- designing a pad array 125

dialog boxes

- Autoroute Options 179
- Bookmark 65
- Copper Colors/Enables/... 58
- Copy File 73
- Current Object Settings 59
- Driver Configuration 195
- Edit Alignment Target 152
- Edit Circle 114
- Edit Copper Tool 158
- Edit Hole 121
- Edit Net Properties 177
- Edit Net Segment 161, 173
- Edit Outline Segment 95
- Edit Pad 122
- Edit Pad Array Settings 125
- Edit Text 147
- Edit Zone Properties 150
- Edit Zone Segment 150
- Export Module to File 108
- Find 156
- Finished 183
- Get Module 89
- Global Options 56
- Import Module from File 109
- Initialize to Library File 88
- Jump To 66, 166
- Layer 57
- Load ALL Macros from File 84
- Load Print/Plot Setup from File 204
- Macro Maintenance 81
- Netlist Load Options 133
- Place Module 141
- Printing and Plotting 194
- Rename File 73
- Rename Module 92
- Save '\x011F Alt S' Macro to File 83
- Save ALL Macros to File 82
- Save Print/Plot Setup to File 203
- Set Block Parameters 97, 139
- Verbose Inquire 94
- Write Board File 72

- dimensions, placing 188
 - disable switch, paging 220
 - disabling snap grid 70
 - displaying
 - force vectors 137
 - main menu 48
 - module information 93
 - ratsnest 136, 157
 - schematic 24
 - DRAFT, configuring 22
 - drawing
 - board outline 131
 - methods 113
 - new track 168
 - outline 113, 119, 146
 - DRC
 - check 164
 - DRC errors, correcting 168
 - DRC violations, identifying 165
 - Driver Configuration dialog box 195
 - driver switches, virtual memory 227
 - drivers 11
 - droplist box 52
 - dynamic, placement 141
- E**
- Edit Alignment Target dialog box 152
 - Edit Circle dialog box 114
 - EDIT command 95, 150, 173
 - Edit Copper Tool dialog box 158
 - Edit File editors 7
 - Edit Hole dialog box 121
 - Edit Layout
 - command basics 48
 - configuring 46
 - deleting macro 84
 - editors 7
 - introducing 9, 41
 - running 47
 - template, loading 130
 - Edit Net Properties dialog box 177
 - Edit Net Segment dialog box 161, 173
 - Edit Outline Segment dialog box 95
 - Edit Pad Array Settings dialog box 125
 - Edit Pad dialog box 122
 - Edit Text dialog box 147
 - Edit Zone Properties dialog box 150
 - Edit Zone Segment dialog box 150
 - editing
 - module 96, 142
 - routed board 168
 - editors
 - Edit File 7
 - Edit Layout 7
 - View Reference 7
 - ems, simulator switch 223
 - entry box 53
 - erasing
 - routes 185
 - error messages 209
 - ESP design environment
 - changing designs 19
 - running 18
 - startup designs 20
 - excluding vias 178
 - existing route, locking 177
 - EXIT command 75
 - Export Module to File dialog box 108
 - exported modules, deleting 110
 - exporting
 - macro file 83
 - modules 108
 - extended memory 2, 3
 - switches 216

F

field contents, updating 30
file
 copying 73
 deleting 74
 names 17
 renaming 73
 updating 71
 writing 72
fill zone
 creating 148
 placing 148
FIND command 156, 160, 161
Find dialog box 156
finding information 4
Finished dialog box 183
finishing, layout 186
first dimension, placing 188
first macro, creating 76
first track, routing 159
Fix Time Stamps 7
flipping 101, 144
force vector 135

G

Gerber 205
Get Module dialog box 89
getting started 155, 193
Global Options 130
 Find Highlights 130
 Grid Size 130
 Outline Pads 130
Global Options dialog box 56
GO TO FUNCTION command 80, 88, 92,
 179
grid
 color, changing 70
 divisor, setting 70
 options, setting 69
 size, setting 69

H

hiding module text 142
HIGHLIGHT command 159, 161
highlighting a net 156
holes, adding 121
HP-GL/2 206

I

identification, board 146
identifying DRC violations 165
IFORM
 about 37
 configuring 37
ILINK
 about 36
 configuring 36
Import Module from File dialog box 109
INET
 about 35
 configuring 35
information, finding 4
Initialize to Library File dialog box 88
input, keyboard 17
INQUIRE
 command 93, 99, 165
 using 165
installing PC Board Layout Tools 386+ 9, 12
interface, command 49
interrupt
 mapping switches 219
 relocation switches 218
introducing
 Edit Layout 9, 41
isolating, modules 129

J

JUMP command 65, 66, 68, 165
Jump To dialog box 66, 166
jumping to bookmark 66

K

key fields, viewing 27
keyboard input 16, 17

L

layer 57
 marker 145
 selecting 60
LAYER command 60, 165
Layer dialog box 57
layout
 finishing 186
 placement 129
learning PC Board Layout Tools 386+ 9
leaving the library editor 128
librarians
 Make Board Template 7
 Make Library 7
library
 editor
 leaving 128
 selecting 88
 file, updating 91
 prefix 44
line 20, address, switch 224
list, boxes 51
Load ALL Macros from File dialog box 84
Load Netlist File dialog box 133
Load Print/Plot Setup from File dialog box 204
loading
 macro 85
 netlist 132
 printer setup 204
 template file 130
location switch, swap file 228
locking existing route 177

M

macro file

- deleting 84
- exporting 83
- loading from 85
- saving 80

Macro Maintenance dialog box 81

macros 76, 80

main menu

- displaying 48
- returning 55

Make Board Template 7

Make Library 7

manual routing 155

mapping, interrupt, switches 219

marker, layer 145

memory

- extended 2, 3
- switches
 - conventional 212
 - extended 216

menus 49

messages, error 209

minimum configuration 3

mirroring

- modules 99, 100

mixed mode program switches 214

Modify Modules 7

Module Report 8

modules 12

- copying 89
 - creating 111
 - flipping 144
 - importing 108
 - information, displaying 93
 - isolating 129
 - libraries 89
 - moving 96
 - placing 129, 135, 138
 - renaming 92
 - saving 128
 - size 2
 - text
 - hiding 142
 - rotating 143
- ### mouse basics 16
- ### MOVE command 96, 97, 98
- ### movement, pointer, resolution 64
- ### moving
- around screen 47
 - back onto grid 104
 - module 96
 - off-grid object 104
 - reference designators 186
 - selected objects within a group 103
 - single module object 102

N

net

- assigning 150
- highlighting 156

netlist

- creating 35
- loading 132
- prefix 45
- viewing 38

netlist block, defining 132

Netlist Load Options dialog box 133

Netlist Loader 132

new module, starting 111

new pad symbol, selecting 124

new track, drawing 168

O

off-grid object, moving 104

offending track, deleting 169

operating system command prompt 17

options, setting 130

OrCAD/PCB II, upgrading 11

ORIGIN command 68, 76, 79, 113, 119

origin, changing 68

outlines, drawing 119, 146

P

pad array

- designing 125
- placing 122

page replacement policy switches 228

pages

- configuring 196
- printing 200

paging disable switch 220

pc and pc/xt detection switch 225

PC Board Layout Tools

- configuring 42
- installation 12
- introduction 2
- learning 9

performing DRC check 164

Phar Lap, technical information 10

PLACE command 77, 78, 79, 96

Place Module dialog box 141

placeholders, positioning 127

placement

- aids 135
- coordinate 138
- dynamic 141

placing

- assembly outline 187
- autoroute zone 176
- board name 147
- board objects 145
- dimensions 188
- fill zone 148
- first dimension 188
- modules 129, 138
- pad array 122
- second dimension 190
- TUTOR board 9

- plotting 205
- pointer movement resolution 64
- policy switches
 - page replacement 228
 - swap file 230
- positioning
 - arcs 116
 - placeholders 127
- PostScript 206
- preferences, setting 112
- prefix
 - board file 45
 - library 44
 - netlist 45
 - temp file 45
- preset steps, rotating in 98
- printer
 - configuring 195
 - setup 203
- printing and plotting
 - all pages 200
 - pages 200
 - selected pages 201
 - TUTOR board 10
- Printing and Plotting dialog box 194
- processors
 - Create NC Drill File 7
 - Fix Time Stamps 7
 - Modify Modules 7
 - Reannotate Board File 7
- program switches, mixed mode 214
- project, organizing by 4

Q

- QUIT command 24, 31, 71, 72, 73, 74, 75, 91

R

- radio button 53
- ratsnest 135, 136
- Reannotate Board File 7
- recommendations 3
- reduction, via 184
- reference designators, moving 186
- relocation, interrupt, switches 218
- Rename File dialog box 73
- Rename Module dialog box 92
- renaming
 - file 73
 - module 92
- reporters
 - Compare Netlist 8
 - Module Report 8
- returning to main menu level 55
- rotating
 - module text 143
 - preset steps 98
 - specific module angle 97
- ROUTE command 155, 159, 160, 161, 172
- routed board, editing 168
- routes, erasing 185
- routing
 - arc segments 161
 - area, zooming 155
 - board 159
 - conditions net, setting 178
 - first track 159
 - manually 155
 - TUTOR board 10
 - vias 160
- running
 - CFIG386 207
 - Check Design Integrity 34
 - defined macro 85
 - DRC check 164, 169
 - Edit Layout 47
 - ESP design environment 18
 - macros 80
 - To Layout 38

S

- Save `\x011F Alt S'` Macro to File dialog box 83
- Save ALL Macros to File dialog box 82
- Save Print/Plot Setup to File dialog box 203
- saving
 - board file 71
 - configuration 71
 - macro 80
 - module 128
 - printer setup 203
 - work 153, 174, 191
- schematic
 - annotating 25
 - displaying 24
 - transferring to layout 9
- Schematic Design Tools, configuring 22, 23
- scroll buttons 53
- second dimension, placing 190
- second macro, creating 79
- section of board, autorouting 180
- selected
 - objects within a group, moving 103
 - pages, printing 201
- selecting
 - layer 60
 - library editor 88
 - new pad symbol 124
 - TUTOR386.NET 132
 - zoom window 63
- SELECTIVE command 106, 107
- Set Block Parameters dialog box 97, 139
- SET command 47, 56, 69, 70, 76, 96, 130, 148
- setting
 - autoroute method 179
 - autorouter options 179
 - conditions 159
 - grid divisor 70
 - grid options 69
 - grid size 69
 - preferences 112
 - routing conditions net 178
 - sweep routing direction 179
 - sweep window 182
 - zoom scale 63
 - work conditions in Edit Layout 56
- settings, current 59
- simulator switch, ems 223
- snap grid, disabling 70
- specific layer, deleting objects on 105
- specifying a copper tool 178
- stack allocation switches 215
- starting
 - new module 111
- startup design, changing 20
- step B0 switch, 80386 222
- stub, deleting 171
- stuff file, creating 28
- summary 39, 85, 128, 153, 174, 191
- suspend to system 75
- swap file
 - location switch 228
 - policy switches 230
- sweep
 - routing direction 179
 - window 182
- system
 - call data buffer switches 213
 - requirements 2, 3
 - suspend to 75

T

- targets, alignment, placing 152
- technical information, Phar Lap 10
- temp file prefix 45
- thermal relief, viewing 151
- To Digital Simulation 8
- To Layout 38
- To Main 8
- To PLD 8
- To Schematic 8
- tools
 - editors 6
 - librarians 6
 - processors 6
 - reporters 6
 - transfers 6
- TOP COPPER LAYER page, building 196
- track, deleting and undeleting 171
- track
 - path 172
 - width 173
- TRACK DELETE command 171, 172
- transferring from schematic to layout 9
- transfers
 - To Digital Simulation 8
 - To Main 8
 - To PLD 8
 - To Schematic 8
- turning off
 - force vectors 137
 - ratsnest 136
- TUTOR board
 - autorouting 10
 - placing 9
 - printing and plotting 10
 - routing 10
- TUTOR design, changing 19
- TUTOR386.NET, selecting 132

U

- UNDELETE command 106, 171
- undeleting a track 171
- Update Field Contents 24
- updating
 - field contents 30
 - file 91, 71
- upgrading from OrCAD/PCB II 11, 12
- using
 - bookmarks 65
 - INQUIRE 165
 - JUMP 166

V

- vdisk compatibility switch 221
- vector, force 135
- VERBOSE INQUIRE command 94
- Verbose Inquire dialog box 94
- via reduction 184
- vias
 - excluding 178
 - routing 160
- view of display, changing 113
- view of layout, changing 62
- View Reference 7
- viewing
 - key fields 27
 - netlist 38
 - thermal relief 151
 - violated areas 166
- violated areas, viewing 166
- virtual memory 2
 - configuring 3
 - disk space 2, 3
 - driver switches 227
 - recommendations 3

W

weitek 1167 switch 217
whole board autorouting 182
WINDOW ZOOM command 63, 148
work, saving 153, 174, 191
working
 environment 5
 libraries 89
Write Board File dialog box 72

X

X SHOW RATSNEST command 136, 157

Z

zone/pad isolation 149
ZOOM command 47, 62, 63, 64, 76
zoom
 scale, setting 63
 window
 selecting 63
 routing area 155

