PC Board Layout

Tools 386+

User's Guide

OrCAD®

Electronic Design Automation Tools

PC Board Layout Tools 386+ User's Guide

Second Edition 8 Dec 1993

OrCAD®

9300 S.W. Nimbus
Beaverton, Oregon 97005
U.S.A.

Sales & Administration	(503) 671-9500
FAX	(503) 671-9501
Technical Support	(503) 671-9400
24-Hour Bulletin Board System	(503) 671-9401

CONTENTS

Chapter 3: Transferring from schematic to layout (continued)

Welcome to OrCAD PC Board Layout Tools 386+

You now have **OrCAD PC Board Layout Tools 386+** version 1.10, a powerful yet straightforward PC board layout tool set with the capability of an engineering workstation. **PC Board Layout Tools 386+** version 1.10 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 through-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*

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.*

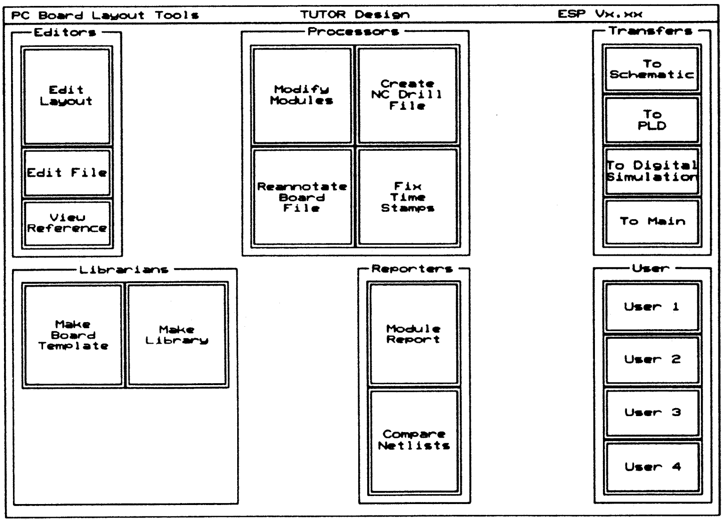


*Figure 1-1. The **PC Board Layout Tools** screen.*

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+** version 1.10.

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 a 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.*

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.

New display drivers

If you are installing **PC Board Layout Tools 386+** v1.10 on a system that does not contain PCB 386+ v1.00, you *must* install new display drivers. Be sure to follow the instructions in the *Installing PC Board Layout Tools 386+ v1.10* section.

Updating from PCB 386+ v1.00

If PCB 386+ v1.00 is already installed, you do not need to install new drivers when you install PCB 386+ v1.10. Local configurations are automatically created for updated PCB 386+ utilities.

You should back up any modules that you modified in the OrCAD supplied module libraries. Enhancements have been made to many of the modules, so it is recommended that you install the new libraries. Place your modified modules in a custom library.

Installing PC Board Layout Tools 386+ v1.10

To install **PC Board Layout Tools 386+** v1.10 you must use the *new* INSTALL program to copy **PC Board Layout Tools 386+** to your system. This is explained in *Running INSTALL.*

Running INSTALL

Use the INSTALL program provided with **PC Board Layout Tools 386+** v1.10 to install the software. *Do not use an older version of INSTALL already on your hard disk.* INSTALL is customized for each product installation.

△ *NOTE: You may have problems installing the software if your computer has several TSRs (terminate and stay resident programs) loaded, such as an antivirus 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™.*

To minimize problems with INSTALL:

- ❖ *Reboot without TSRs loading.*
- ❖ *Do not run SHARE if you are connected to a network.*
- ❖ *Install from the DOS command line, not from the DOS window 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.

Installing new display drivers

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

- You *must* install new display drivers if you are upgrading from OrCAD/PCB II or if you never before installed an OrCAD PCB product.
- You do not need to install new display drivers if you are updating from PCB 386+ v1.00.

Installing tutorial files

INSTALL asks if you have **Schematic Design Tools Release IV** or **Schematic Design Tools 386+** already installed. If you select SDT Release IV, compatible 16-bit PCB 386+ tutorial files are installed in the specified tutorial directory (C:\ORCAD\TUTOR if you use the default directory structure). The subdirectory ORCAD\TUTOR\SDT386 is created, and the corresponding 32-bit tutorial files are installed in this directory.

If you select SDT 386+, compatible 32-bit tutorial files are installed in the specified tutorial directory (C:\ORCAD\TUTOR), and the 16-bit tutorial files are installed in the subdirectory ORCAD\TUTOR\SDT4.

See *Chapter 3: Transferring from schematic to layout* for more information on these PCB 386+ tutorial files.

Converting PCB II boards and modules

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 are updating from PCB 386+ v1.00, and you already converted your OrCAD/PCB II files, then no conversion is necessary.

If you select automatic conversion, the installation program uses the paths you supply during installation 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 during installation.

Updating ORCADESP.DAT

INSTALL checks your system to see if the ESP design environment is installed. If ESP is present, then the ORCADESP.DAT files in your design directories are automatically updated. The ORCADESP.DAT files must be updated so the new ESP design environment configurations are available.

If problems occur during this part of the installation, you will need to manually update your ORCADESP.DAT files using the MERGEDAT program.

MERGEDAT is described in *Technical Note #45: Updating ORCADESP.DAT files with MERGEDAT.*

Finishing the installation

INSTALL asks if you want to automatically or manually update your AUTOEXEC.BAT file. INSTALL adds an entry to the PATH statement specifying the location of the ORCADEXE directory, and also adds four environment variables. The environment variables designate where OrCAD applications look for files. You can elect to manually add these entries after the installation is complete.

If you choose to have AUTOEXEC.BAT updated automatically, and you use the default directory structure, the following entries are added:

```
PATH=C:\;C:\DOS\;C:\ORCADEXE

SET ORCADEXE=C:\ORCADEXE\
SET ORCADESP=C:\ORCADESP\
SET ORCADPROJ=C:\ORCAD\
SET ORCADUSER=C:\ORCADESP\
```

If you already have Release IV or 386+ products installed, and your OrCAD directory structure is unchanged, AUTOEXEC.BAT does not need to be updated because the correct entries are already in the file.

When the installation is complete, the DOS prompt displays. Follow these steps to verify and establish your system configurations for OrCAD applications:

1. Check your AUTOEXEC.BAT file to verify that the PATH statement and OrCAD environment variable entries are included.
2. Reboot your system so that the changes 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+**. In this chapter, you:

- Configure the ESP design environment and **Schematic Design Tools**
- Annotate and add module values to a schematic so you can produce a netlist from the schematic
- Configure **To Layout** to produce netlist TUTOR386.NET, which is used to create a circuit board in *Chapter 6: Placing the TUTOR board*
- Transfer from **Schematic Design Tools** to **PC Board Layout Tools 386+**

The files you need

If you specified during the installation of PCB 386+ v1.10 that you have **Schematic Design Tools 386+** currently installed, INSTALL copies the following 32-bit tutorial files to your ORCAD\TUTOR directory:

- TUTOR386.SCH – A 32-bit schematic
- TUTOR386.LIB – A 32-bit library for TUTOR386.SCH

If you specified that you have **Schematic Design Tools Release IV** installed, INSTALL copies the following 16-bit tutorial files to your ORCAD\TUTOR directory:

- TUTOR4.SCH – A 16-bit schematic
- TUTOR4.LIB – A 16-bit library for TUTOR4.SCH

The filenames referenced in this chapter are TUTOR386.SCH and TUTOR386.LIB because the procedures are written for **Schematic Design Tools 386+** v1.10.

Documentation changes for 16-bit tutorial files

If you use **Schematic Design Tools Release IV**, you use the 16-bit tutorial files, TUTOR4.SCH and TUTOR4.LIB. Note the following documentation changes for this chapter:

- All references to TUTOR386.SCH should be changed to TUTOR4.SCH. Enter **`TUTOR4.SCH`** when you are told to enter **`TUTOR386.SCH`**.
- All references to TUTOR386.LIB should be changed to TUTOR4.LIB. Enter **`TUTOR4.LIB`** when you are told to enter **`TUTOR386.LIB`**.
- In the section *Creating the netlist*, item 5 of the *Configuring IFORM* subsection should read "Select **EDIF.CCF** in the **Netlist Format** list box." You must use EDIF.CCF because it is the 16-bit netlist formatter.

Performing this part of the tutorial

When you complete all the steps in this chapter you produce a netlist named TUTOR386.NET. An identical netlist, TUTORORC.NET, is placed in your ORCAD/TUTOR directory when you install **PC Board Layout Tools 386+**. 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.bdl.**" This text can also be enclosed in a box:

```
tutor.bdl
```

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).

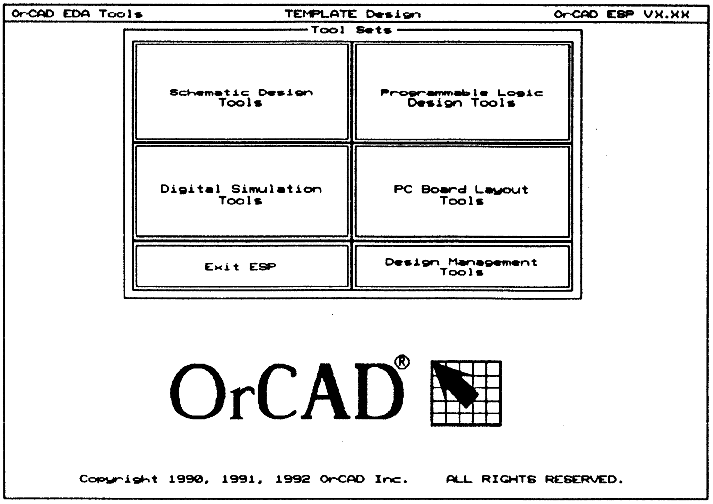


Figure 3-1. The ESP design environment main screen.

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.

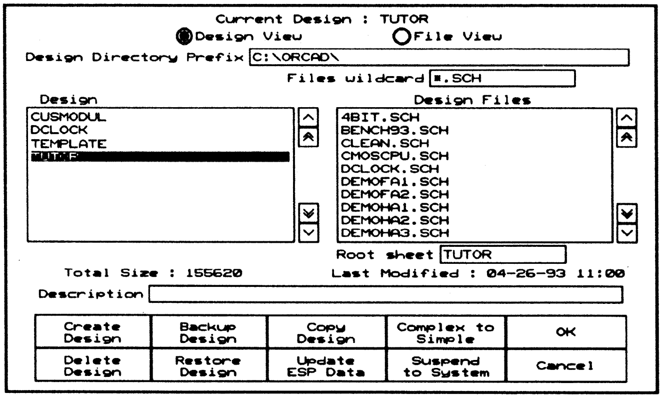


Figure 3-2. Design Management Tools screen, Design View.

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.

```
Design Management Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```

2. Select **Configure ESP.** The **Configure ESP** screen displays (figure 3-3).

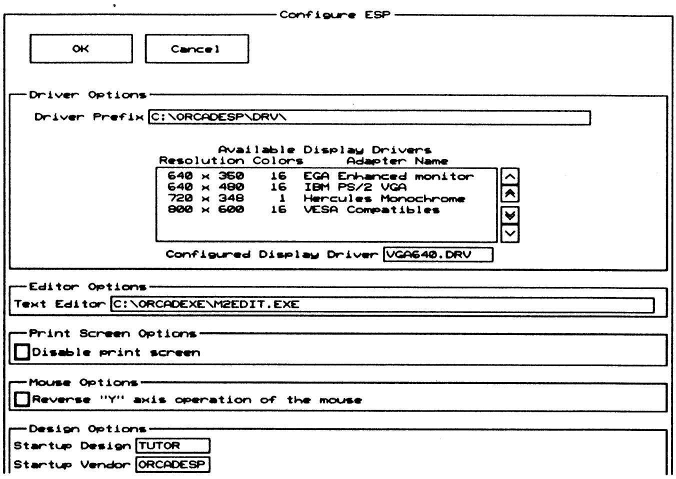


*Figure 3-3. The **Configure ESP** screen.*

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

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).

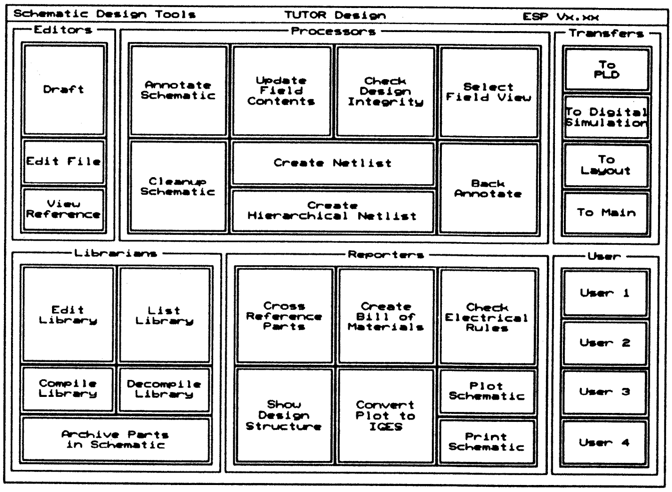


Figure 3-4. The Schematic Design Tools screen.

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

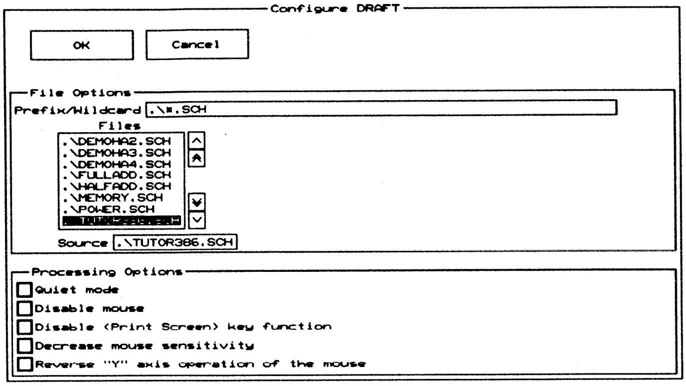


Figure 3-5. The ***Configure DRAFT*** *screen.*

3. Select **.\TUTOR386.SCH** from the **Files** list box. If you have **Schematic Design Tools Release IV,** select **.\TUTOR4.SCH.**

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).

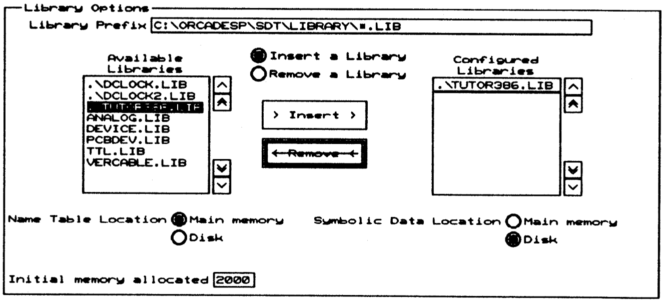


Figure 3-6. Partial view of the ***Configure Schematic Design Tools*** *screen,* ***Library Options*** *area.*

2. Scroll to the **Library Options** area.
3. Select **.\TUTOR386.LIB** from the **Available Libraries** list box, then select **Insert.** The **Configured Libraries** list box displays **.\TUTOR386.LIB.** If you have **Schematic Design Tools Release IV,** select **.\TUTOR4.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).

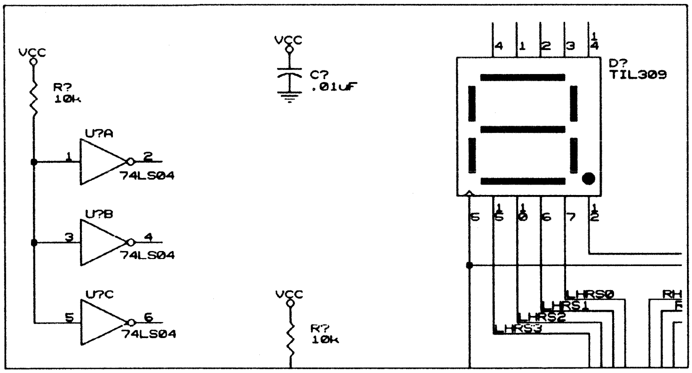


Figure 3-7. Partial view of the TUTOR386 schematic.

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 an update file, then you select **Update Field Contents** to select the update file and insert the module values into **Part Field 8** on the schematic. See **Creating an update 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).

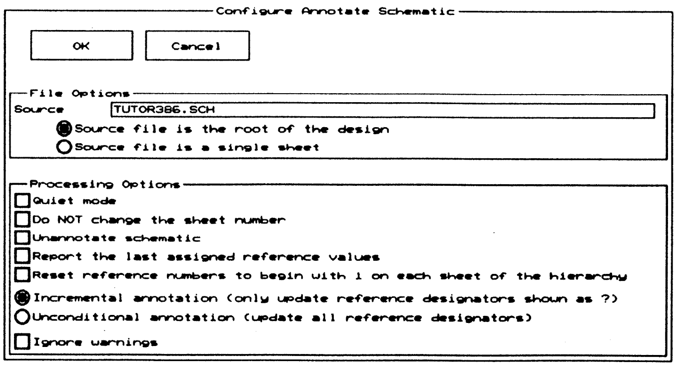


*Figure 3-8. The **Configure Annotate Schematic** screen.*

2. Enter **`TUTOR386.SCH`** in the **Source** entry box.
3. Select **Source file is a single sheet.**
4. Select **OK.**

5. 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.

6. Select **Draft,** then select **Execute.** The reference designators are annotated for TUTOR386.SCH, as shown in figure 3-9.

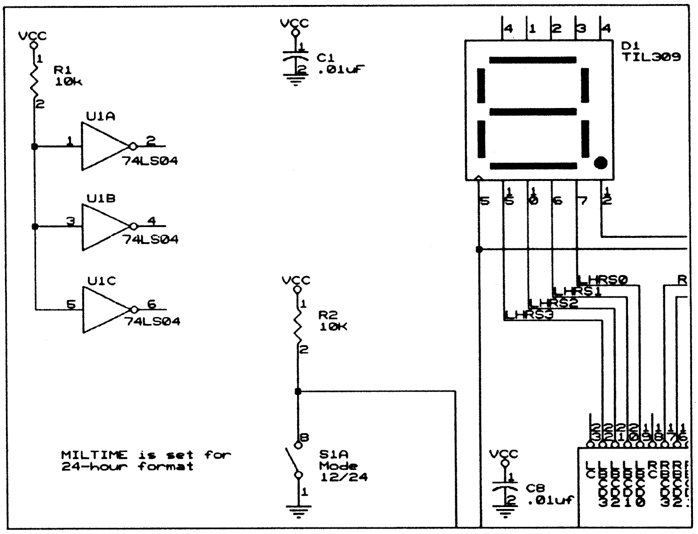


Figure 3-9. A partial view of the annotated schematic.

7. 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. An update file, or stuff file, tells **Update Field Contents** what module value to insert in the schematic when it finds a match.

△ *NOTE: The following sections refer to preset configurations for entries in* ***Configure Update Field Contents*** *and* ***Configure Schematic Design Tools****. The configurations are preset if you do not have a previous version of* ***Schematic Design Tools*** *installed when you install* ***Schematic Design Tools 386+*** *v1.10.*

If you have a version of ***Schematic Design Tools*** *already installed, your previously set configurations are used.*

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 **Configure 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:

1. Select **Draft**, then select **Configure Schematic Tools.** The **Configure Schematic Design Tools** screen displays.
2. Scroll down to the **Key Fields** area.
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 an update file

Update Field Contents requires an update 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 update file TUTOR386.STF:

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

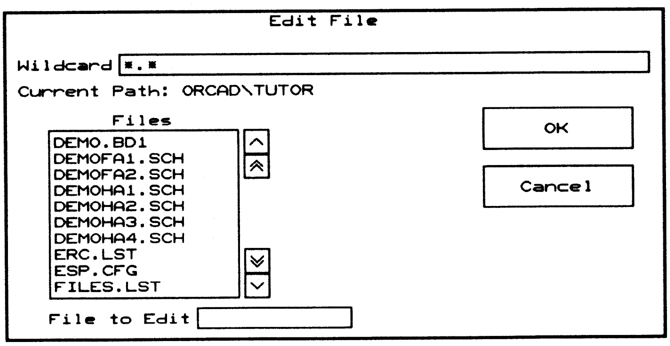


Figure 3-10. The ***Edit File*** *screen.*

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

   An update 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 update file in pairs, so for readability and clarity each pair is placed on a separate line in the update 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 update 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).

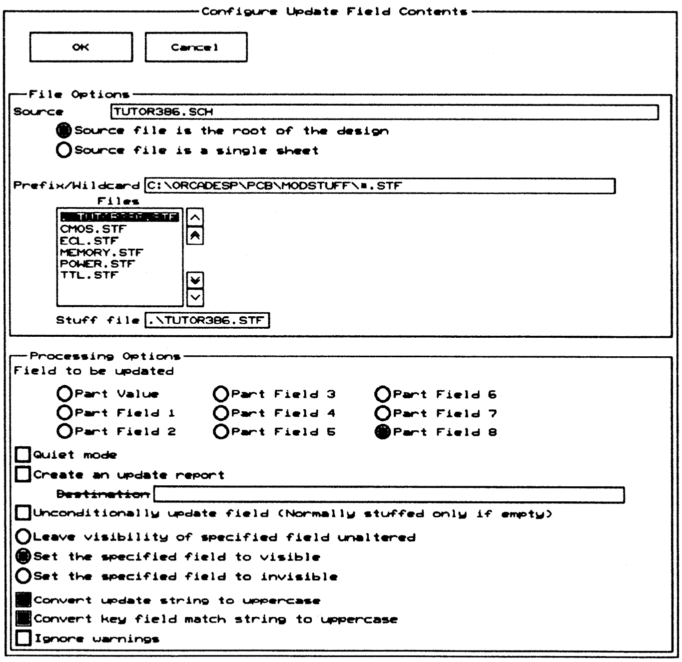


Figure 3-11. The ***Configure Update Field Contents*** *screen.*

2. Enter **`TUTOR386.SCH`** in the **Source** entry box.
3. Select **Source file is a single sheet.**
4. Select **.\TUTOR386.STF** from the **Files** list box. The filename displays in the **Stuff File** entry box.
5. **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 update file.

6. Select **Set the specified field to visible.** You can view the inserted module values on the schematic when this is selected.
7. Select **OK**. The **Schematic Design Tools** screen displays.
8. 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.
9. 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.**
10. 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

```
Configure CLEANUP
Configure CROSSREF
Configure ERC
```

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).

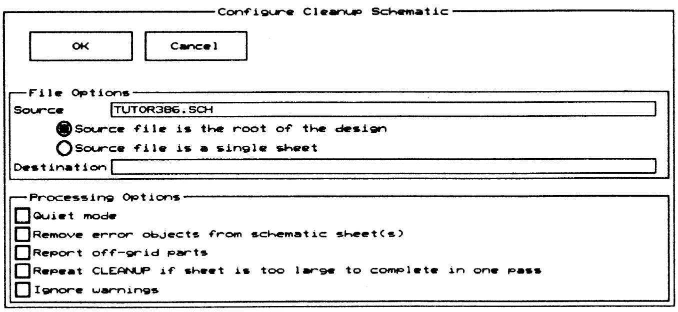


Figure 3-12. The Configure Cleanup Schematic screen.

3. Enter `TUTOR386.SCH` in the **Source** entry box.
4. Select **Source file is a single sheet.**
5. 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).

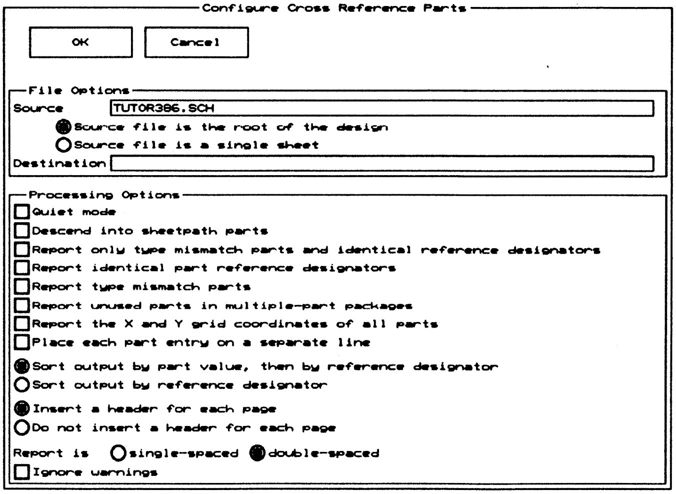


*Figure 3-13. The **Configure Cross Reference Parts** screen.*

2. Enter `TUTOR386.SCH` in the **Source** entry box.
3. Select **Source file is a single sheet.**
4. 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.
5. 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).

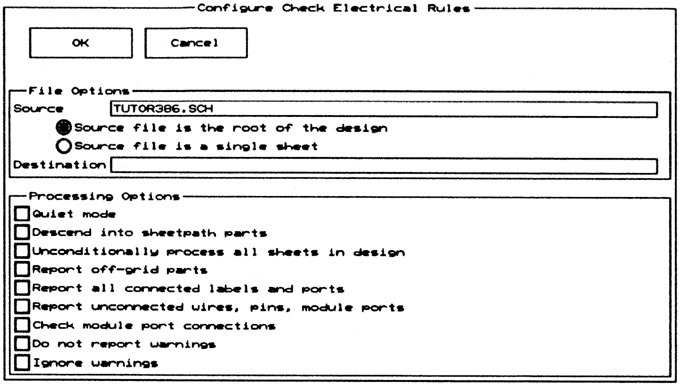


*Figure 3-14. The **Configure Check Electrical Rules** screen.*

2. Enter `TUTOR386.SCH` in the **Source** entry box.
3. Select **Source file is a single sheet.**
4. 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).

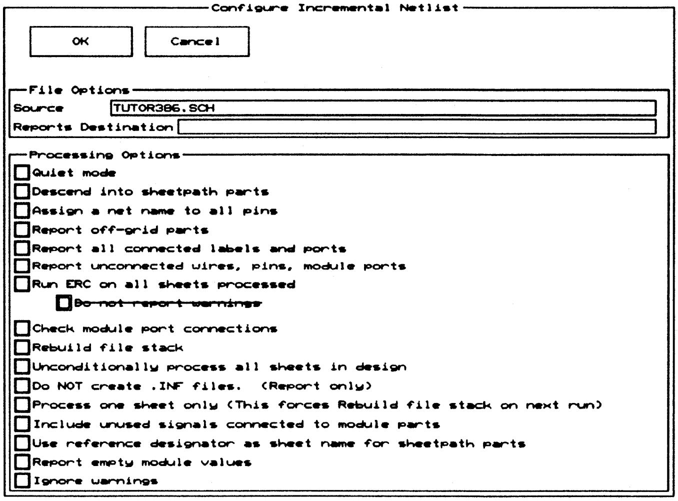


Figure 3-15. The ***Configure Incremental Netlist*** *screen.*

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.

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

2. Select **Configure ILINK**. The **Configure Netlist Linker** screen displays (figure 3-16).

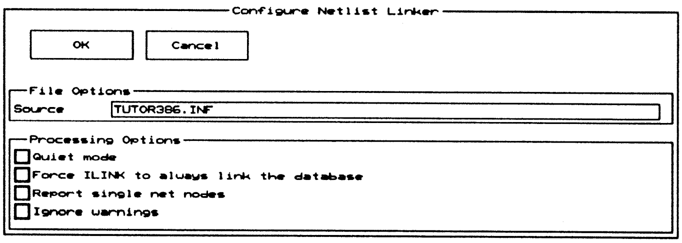


*Figure 3-16. The **Configure Netlist Linker** screen.*

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:

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

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).

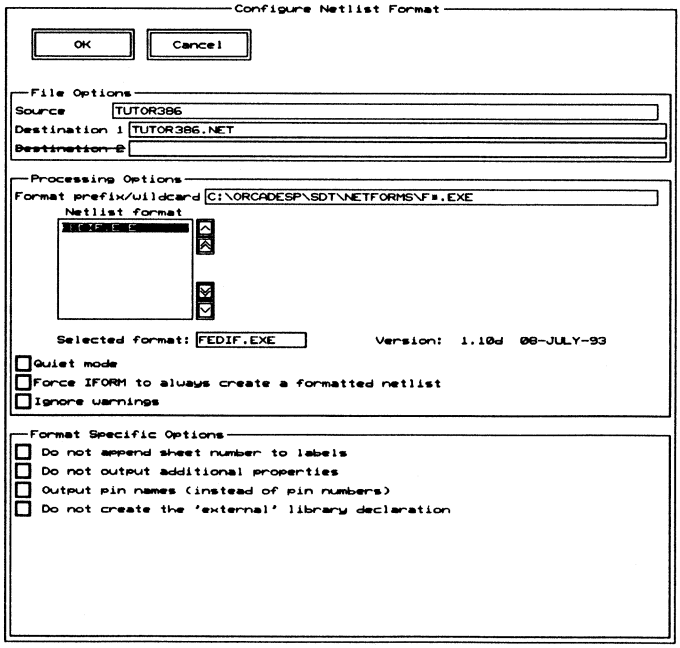


*Figure 3-17. The **Configure Netlist Format** screen.*

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.

   If you are using **Schematic Design Tools Release IV,** change the file extension value from .CF to .CCF in the **Format prefix/wildcard** entry box, then select **EDIF.CCF** in the **Netlist Format** list box.
6. 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 update 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**.

CHAPTER 4

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).

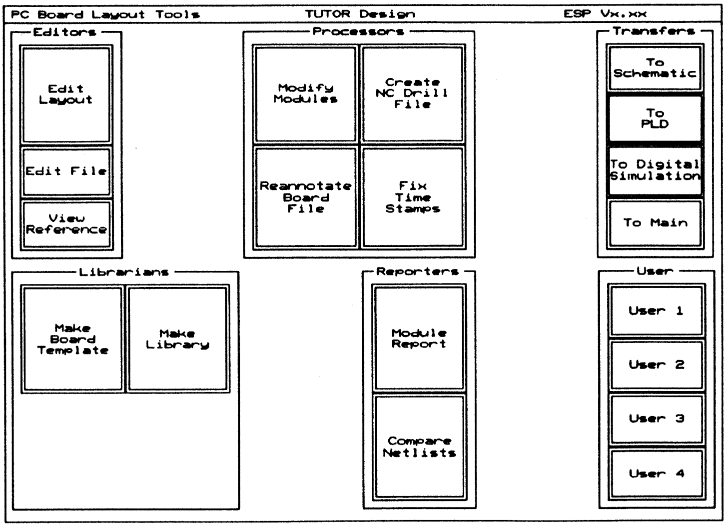


*Figure 4-1. The **PC Board Layout Tools** screen.*

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).

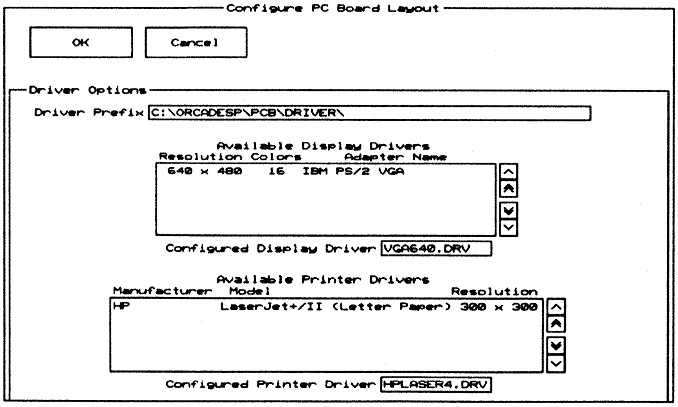


*Figure 4-2. Top of the **Configure PC Board Layout** screen.*

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 and configure a printer driver from the **Available Printer Drivers** list box to print the TUTOR board in* Chapter 9: Printing and plotting the TUTOR board. *However, to plot the board you configure a plotter driver from within **Edit Layout**.*

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 the you enter the proper path to your TUTOR directory so you can follow the steps in this tutorial.*

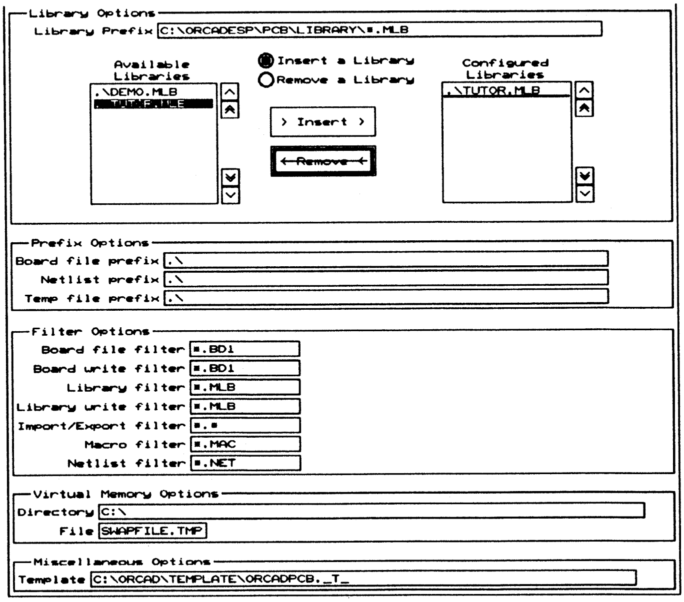


Figure 4-3. Bottom of the ***Configure PC Board Layout*** *screen.*

5. Select **TUTOR.MLB** from the **Available Libraries** list box, then select **Insert**. The filename displays in the **Configured Libraries** list box. You need TUTOR.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. Change the file filters in the **Filter Options** area if you use different extensions for the listed file types.

7. 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.*

8. 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.

9. 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

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

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).

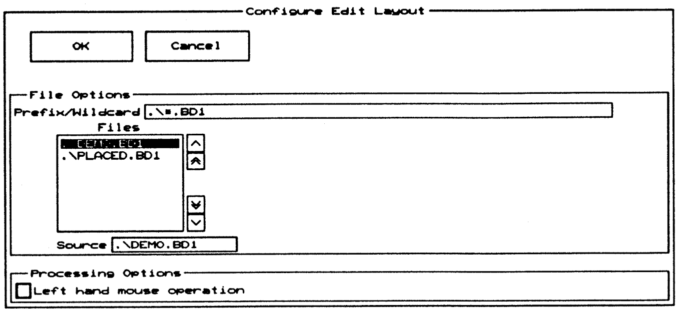


Figure 4-4. The ***Configure Edit Layout*** *screen.*

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**, enable **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 not enabled.*

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

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.

```
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
```

Commands

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.

	Using the keyboard	*Using the mouse*
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 Highlight
```

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.

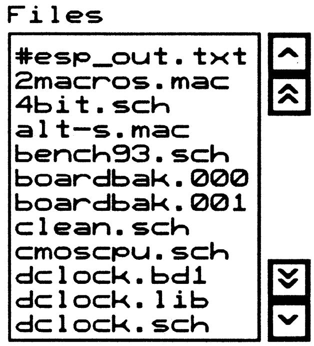
△ *NOTE: There are two buttons that are common to almost all dialog boxes:* ***OK*** *and* ***Cancel.*** *Select* ***OK*** *to close the dialog box and incorporate any changes. Select* ***Cancel*** *to close the dialog box without incorporating any changes.*

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

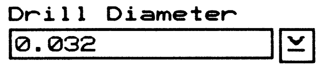
A list box contains a list of selectable items. Place the pointer on the item and click the left mouse button to highlight and select the item.

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

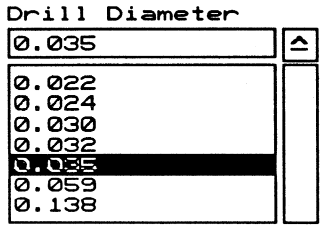


Droplist box

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.



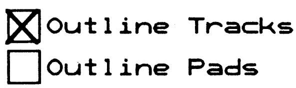
Open droplist box.

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

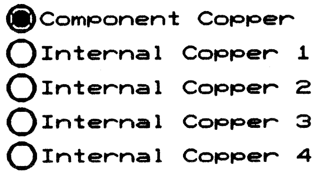
A check box is an option that can be enabled or disabled. Place the pointer on the check box and click the left mouse button to select it. Highlight the check box to enable the option. Select the check box again to remove the highlight and disable the option.



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.

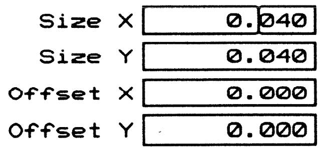
Radio button

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.



Entry box

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



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.

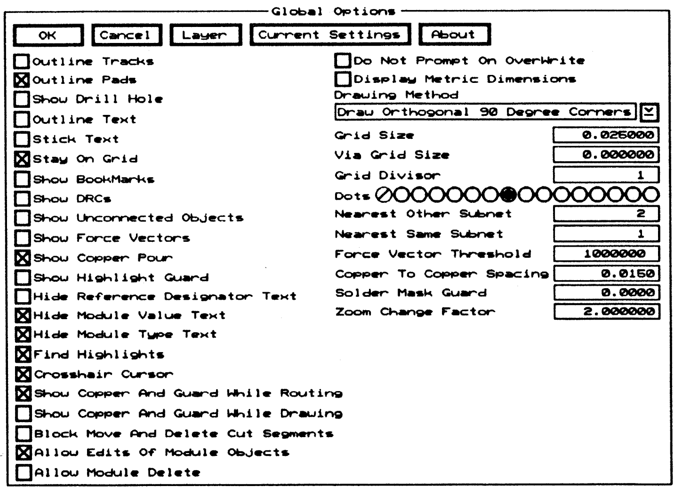


*Figure 4-5. The **Global Options** dialog box.*

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).

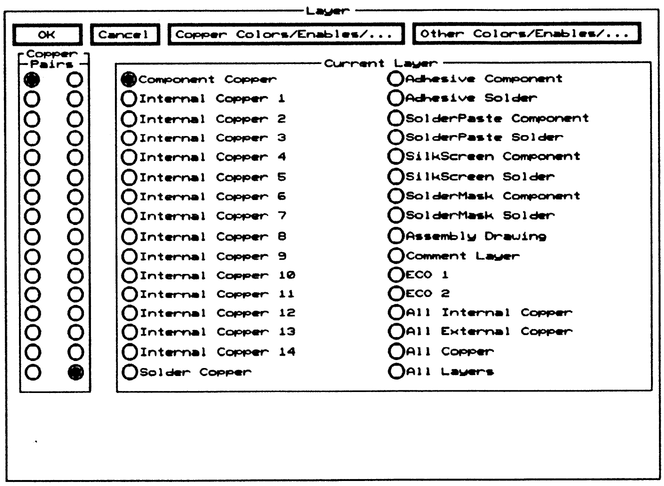


Figure 4-6. The Layer dialog box.

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

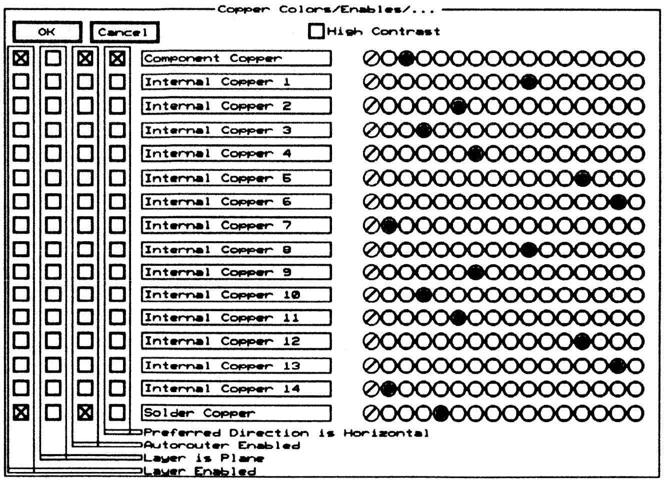


Figure 4-7. The ***Copper Colors/Enables/...*** *dialog box.*

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).

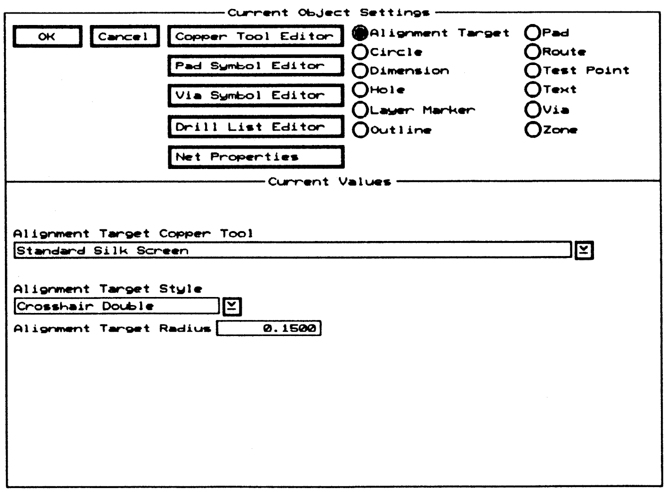


*Figure 4-8. The **Current Object Settings** dialog box.*

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.

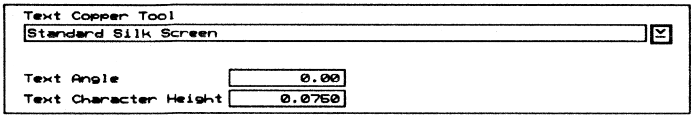


Figure 4-9. Current values for text, as shown in ***Current Object Settings****.*

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 the layer selection commands:

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 the 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.

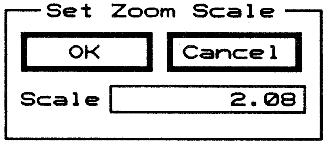


*Figure 4-10. The **Set Zoom Scale** dialog box.*

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.

```
Window Zoom
Jump
Origin
= bookmark
```

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.

Zoom level	*Arrow key resolution*	*<Ctrl> Arrow key resolution*
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).

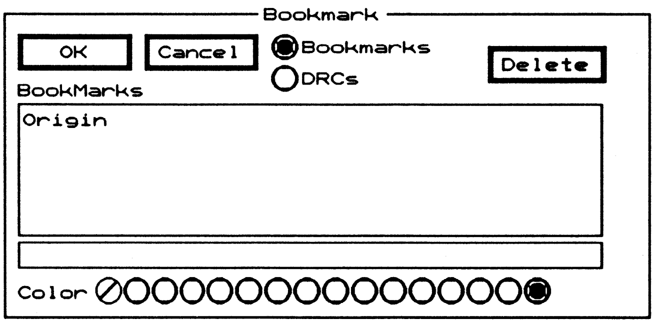


Figure 4-11. The ***Bookmark*** *dialog box.*

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.

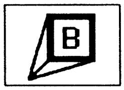
The **DRCs** button works the same way. A DRC is a special type of bookmark (*DRC* stands for *D*esign *R*ule *C*heck) that **Edit Layout** places on the board design where it detects a design error. See *Chapter 7: Routing the TUTOR board* for more information on DRC. You select **DRCs** to display 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).

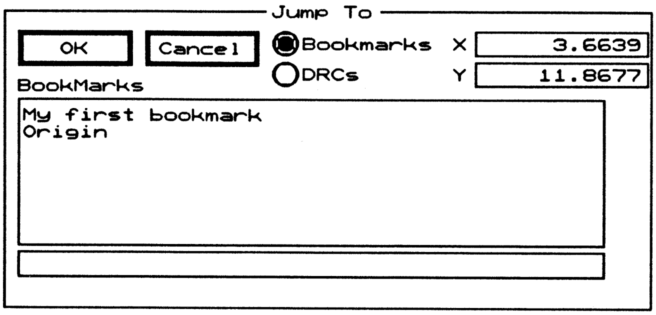


Figure 4-12. The Jump To dialog box.

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 directory.

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).

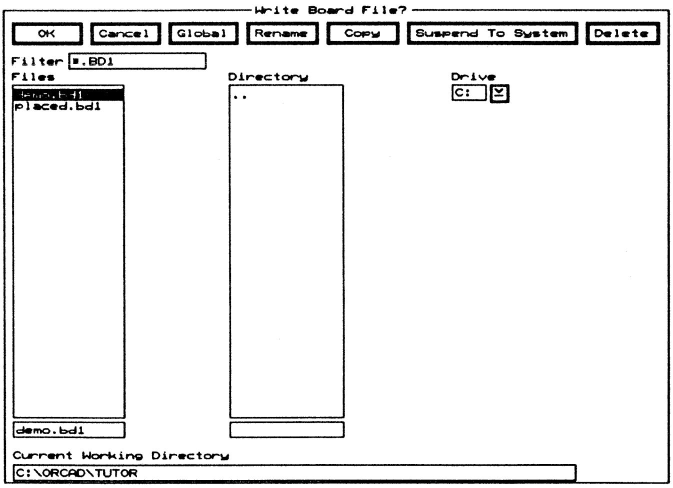
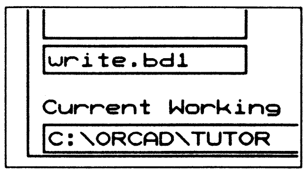


Figure 4-13. The ***Write Board File*** *dialog box.*

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.

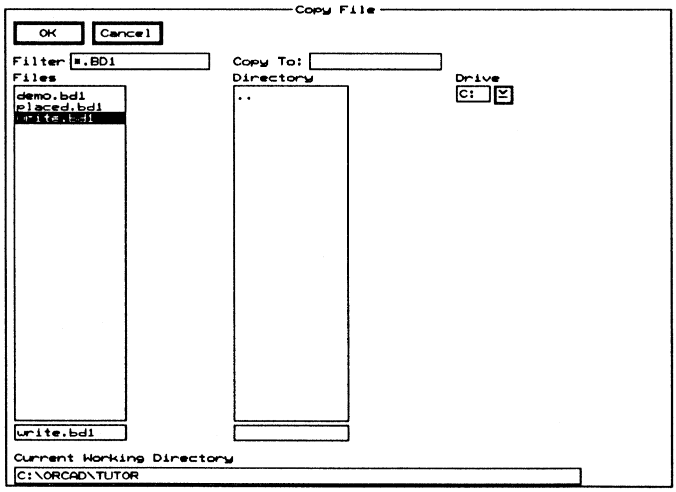


Figure 4-14. The ***Copy File*** *dialog box.*

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).

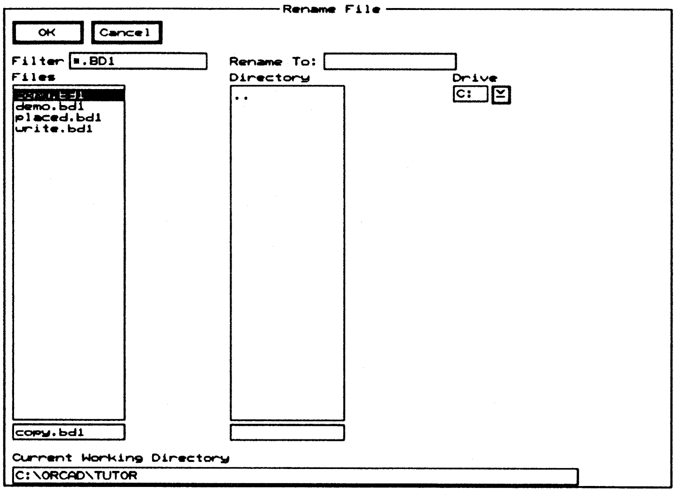


*Figure 4-15. The **Rename File** dialog box.*

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****.*

Suspending 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.

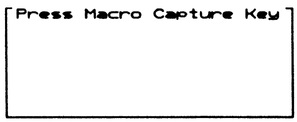
Macros in **Edit Layout** are relative-event data captures. This means that a macro executes its commands relative to the current pointer location, rather than from the original location when the macro was created.

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).

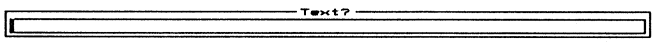


Figure 4-16. The ***Text*** *entry box.*

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).

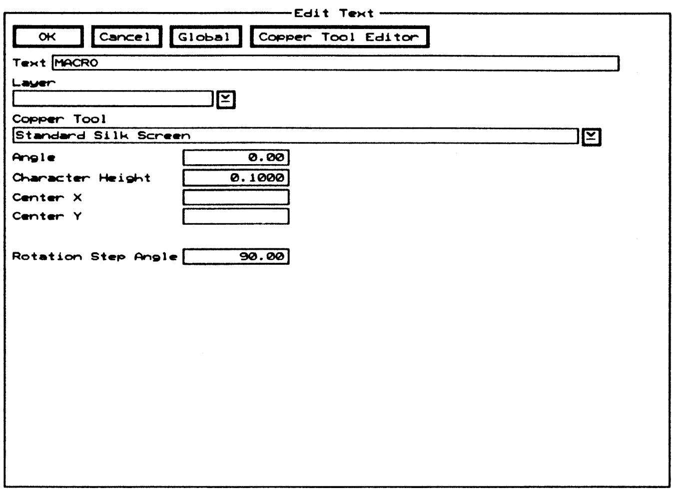


Figure 4-17. The ***Edit Text*** *dialog box.*

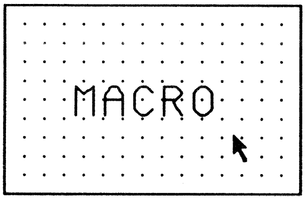
19. Place the pointer in the **Character Height** entry box and click the left mouse button. The text cursor displays.

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
Autorouter
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.

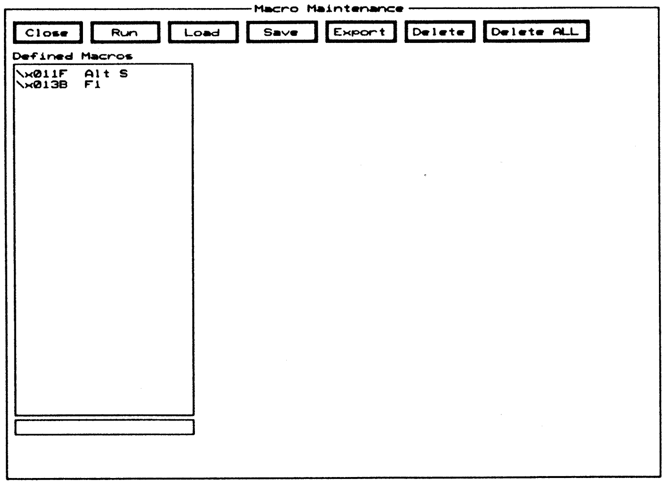


Figure 4-18. The ***Macro Maintenance*** *dialog box.*

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

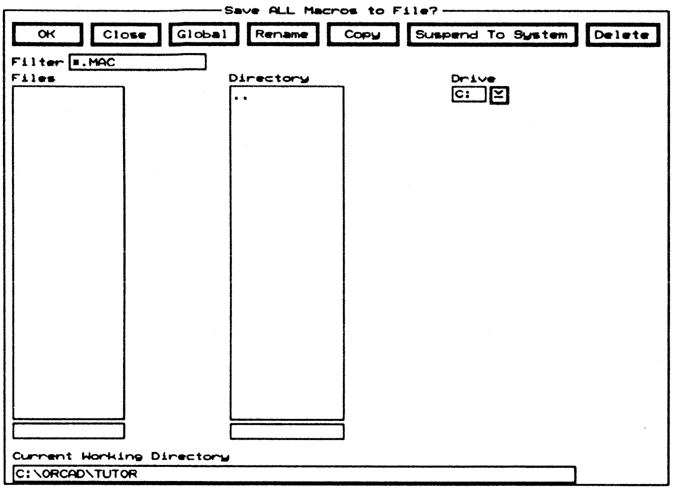


Figure 4-19. The ***Save ALL Macros to File*** *dialog box.*

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).

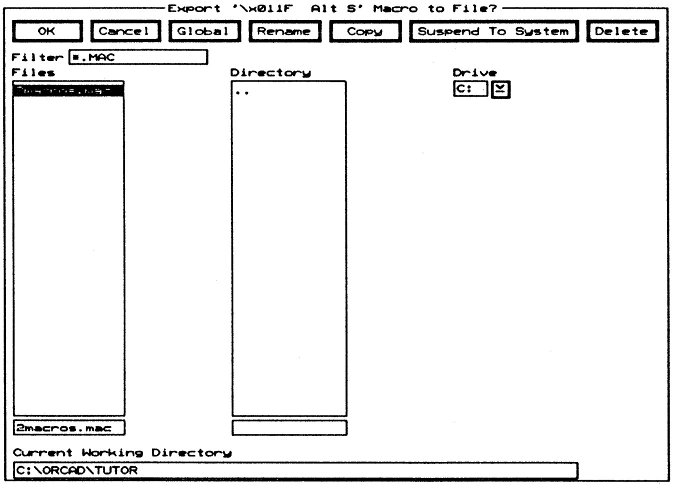


*Figure 4-20. The **Export '\x011F Alt S' Macro to File** dialog box.*

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.

4. Select **OK** to export the macro to ALT-S.MAC. The **Export '\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).

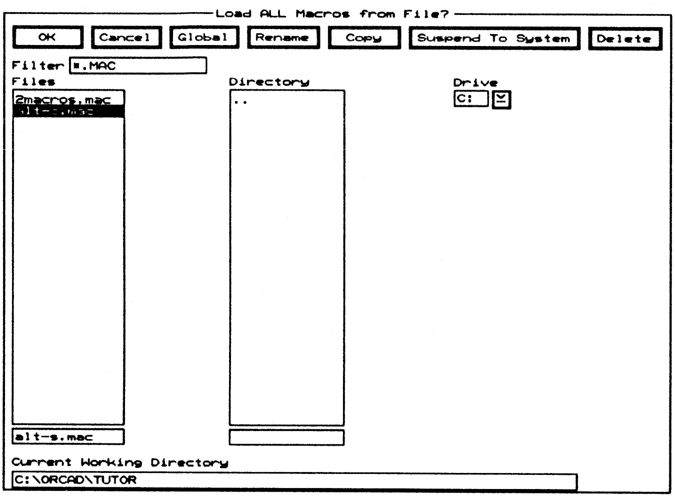


*Figure 4-21. The **Load ALL Macros From File** dialog box.*

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 **\x013B 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
```

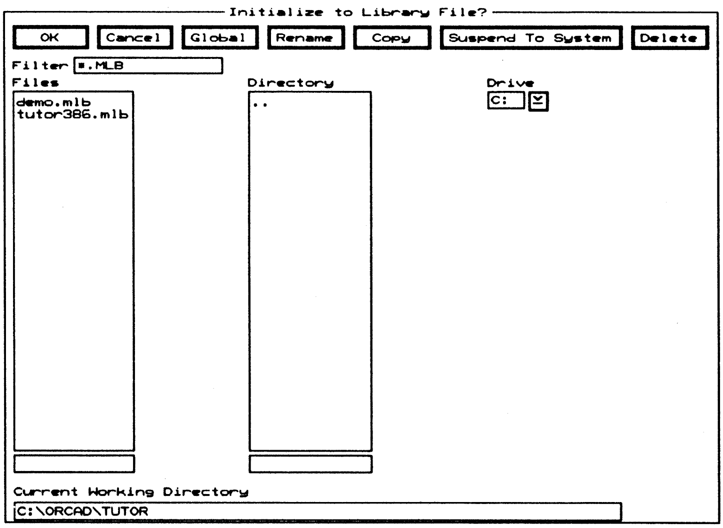


*Figure 5-1. The **Initialize to Library File** dialog box.*

△ *NOTE: If you use the default OrCAD directory structure, C:\ORCAD\TUTOR should display in **Current Working Directory**. You defined this path when you configured PC Board Layout Tools at the beginning of* Chapter 4: Introducing Edit Layout.

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).

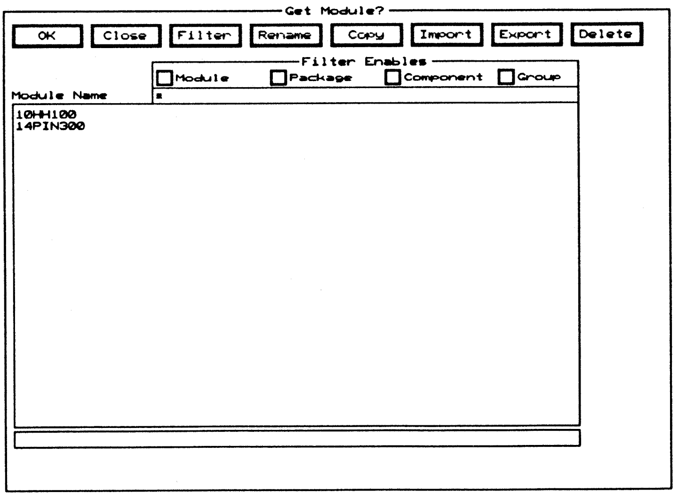


Figure 5-2. The ***Get Module*** *dialog box.*

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.

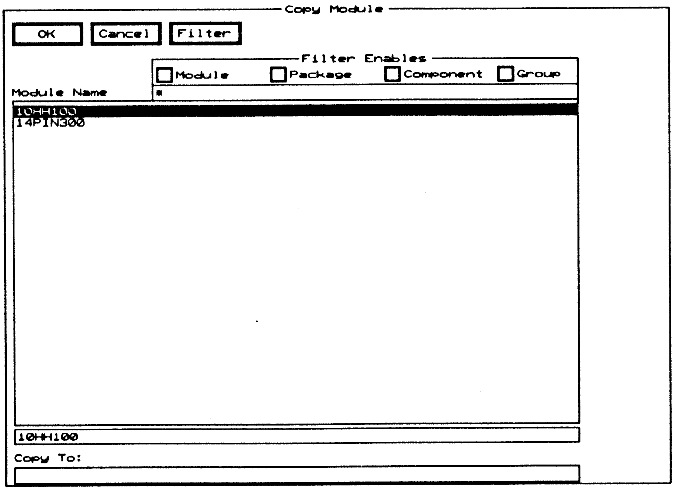


Figure 5-3. The ***Copy Module*** *dialog box.*

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.

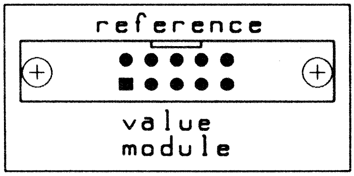


Figure 5-4. The HEADER module.

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.

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

2. Select **Update Library File**. This saves the module HEADER to DEMO.MLB.

Renaming a module

1. Select **GO TO FUNCTION.** The menu shown at right displays.

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

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).

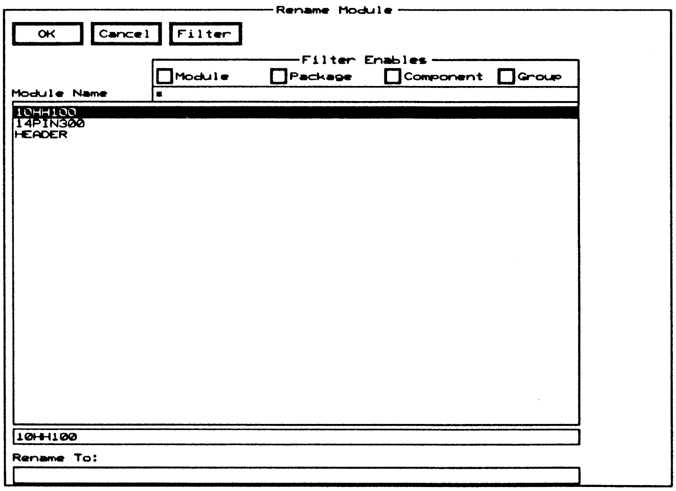


Figure 5-5. The ***Rename Module*** *dialog box.*

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. Additional information may display, depending on your configured video driver.

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).

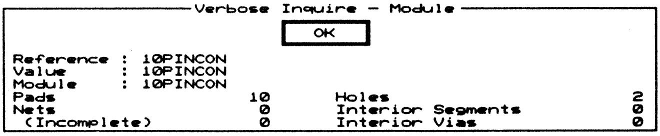


Figure 5-6. The ***Verbose Inquire - Module*** *dialog box.*

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).

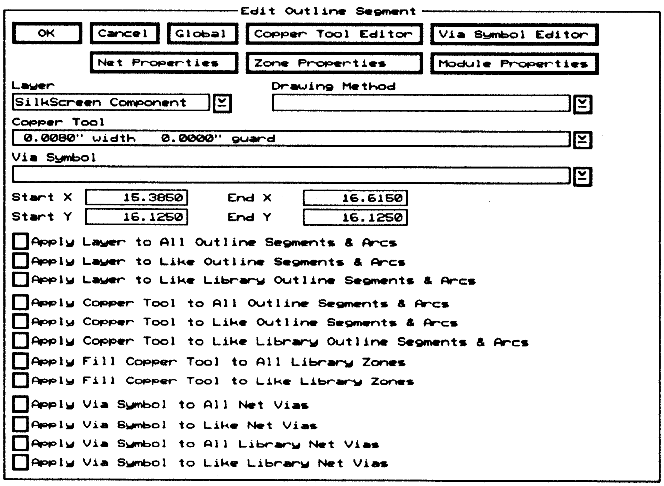


*Figure 5-7. The **Edit Outline Segment** dialog box.*

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 selecting 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.

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).

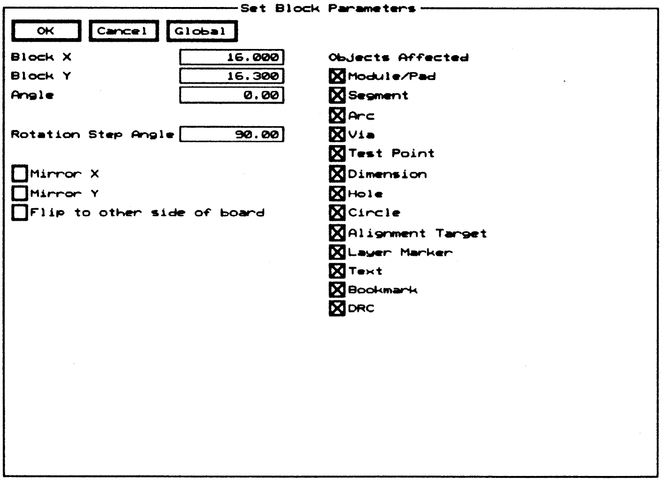
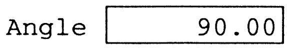


Figure 5-8. The ***Set Block Parameters*** *dialog box.*

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.

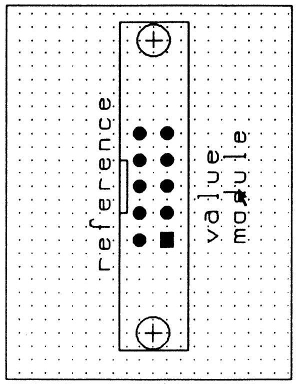


Figure 5-9. The module rotated 90°.

5. Select **Place** to place the module.

   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 [     15.00]
```

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.
2. Select **BLOCK**.

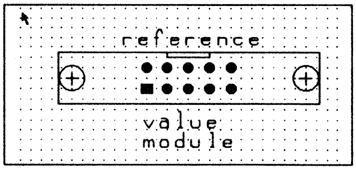


*Figure 5-10. Pointer position for start of **BLOCK** command.*

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.

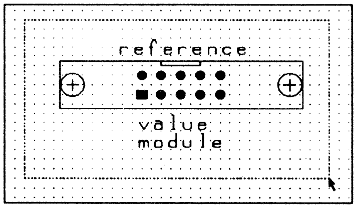
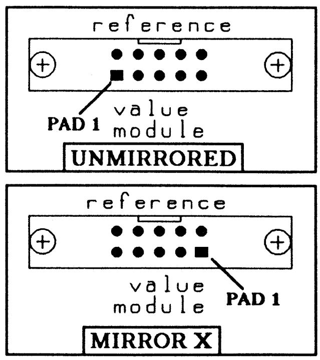


*Figure 5-11. Pointer position for end of **BLOCK** command.*

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

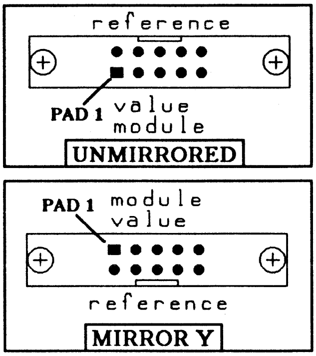
```
Delete Block
Move Block
Drag Block
RatsNest Block
Module Snap Block
```

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

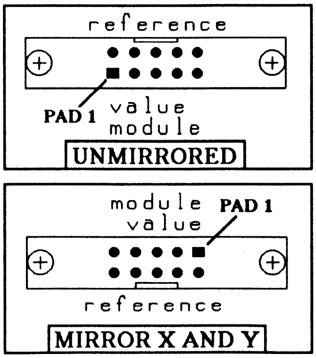
1. Use the **BLOCK** command to select the module, as described in the previous procedure.
2. Select **Move Block,** then select **Set.** The **Set Block Parameters** dialog box displays.
3. Enable **Mirror Y,** then select **OK.** The module mirrors along the Y axis, as shown at right.



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.

   Objects Affected
   - [x] Module/Pad
   - [x] Segment
   - [x] Arc
   - [x] Via
   - [x] Test Point
   - [x] Dimension
   - [x] Hole
   - [x] Circle
   - [x] Alignment Target
   - [x] Layer Marker
   - [x] Text
   - [x] Bookmark
   - [x] DRC

   Enabled objects within the block boundaries are selected if the layer containing the object 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.

   ```
   Delete Block
   Move Block
   Drag Block
   RatsNest Block
   Module Snap Block
   ```
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 other objects that 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. 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

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.

Exporting

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).

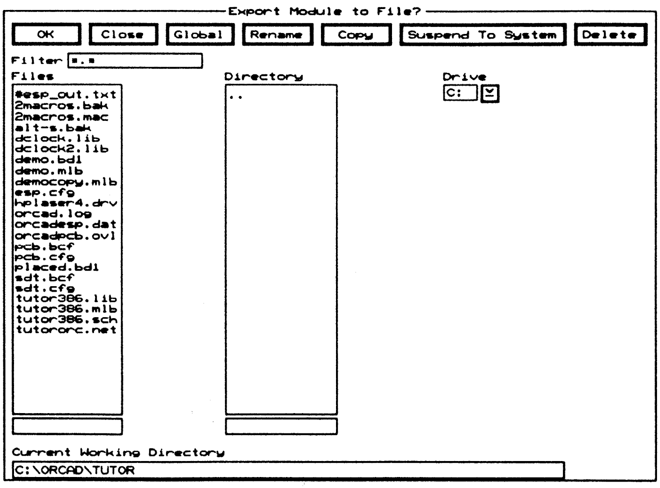


Figure 5-12. The ***Export Module to File*** *dialog box.*

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.

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).

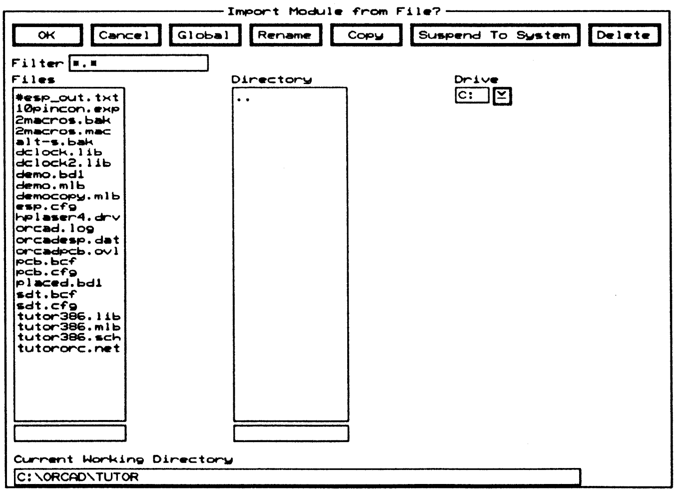


Figure 5-13. The ***Import Module from File*** *dialog box.*

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.

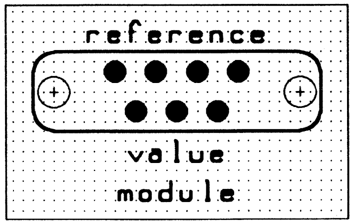


Figure 5-14. The finished 7PINDEMO module.

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.

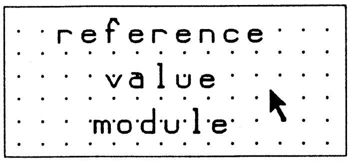


Figure 5-15. Module property placeholders.

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 the preset library editor preferences. Preferences are saved in the module library when you save the file. When you create a new module it inherits the saved preferences.

1. Select **SET**. The **Global Options** dialog box displays. The following options are automatically enabled whenever you create a new module in this library:
   - ❖ **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.

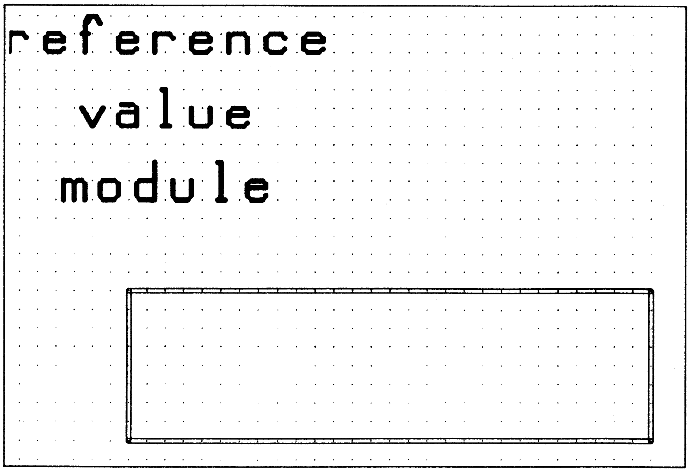


Figure 5-16. The module outline.

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.

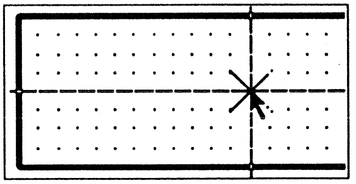


Figure 5-17. The ***PLACE Circle*** *cursor.*

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

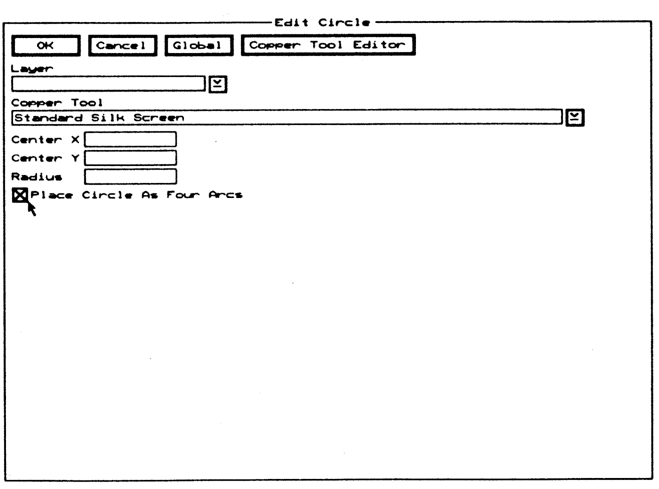


Figure 5-18. The ***Edit Circle*** *dialog box.*

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.

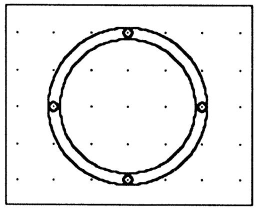


Figure 5-19. Finished circle, showing arc endpoints.

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).

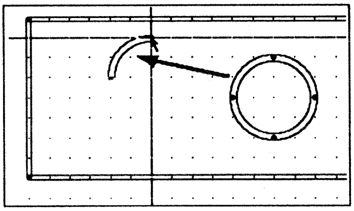


Figure 5-20. Moving the upper left arc.

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).

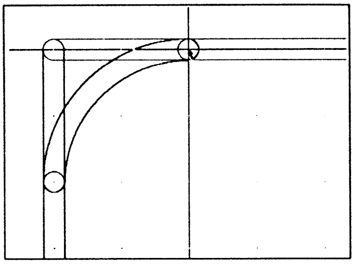


Figure 5-21. Magnified view of arc placement.

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.

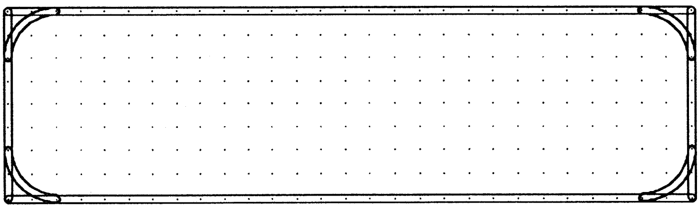


Figure 5-22. The module outline with an arc in each corner.

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).
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**.

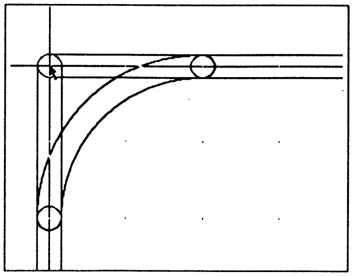


Figure 5-23. Pointer position to delete cut segments.

△ *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.

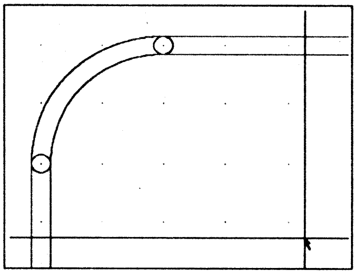


Figure 5-24. Enlarged view of corner after deleting 90° segments.

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

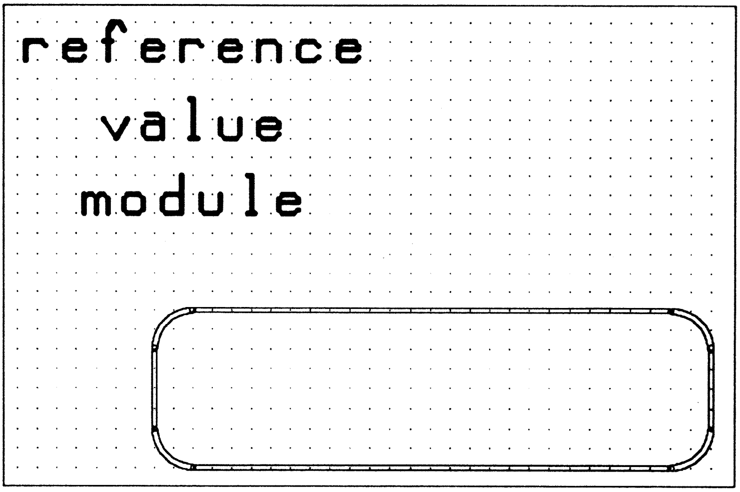


Figure 5-25. The completed module outline.

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.

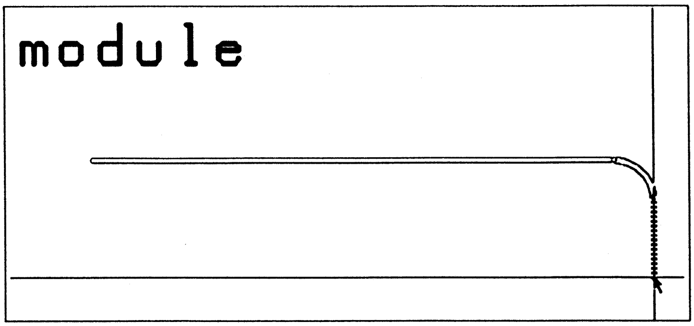


Figure 5-26. Drawing the outline with arc corners.

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 left 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).

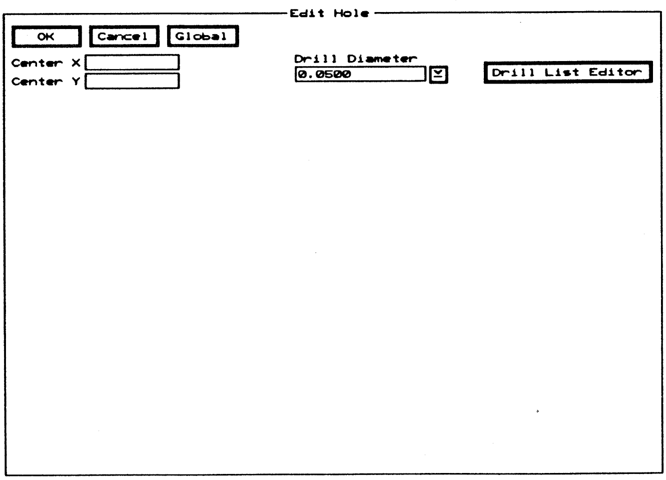
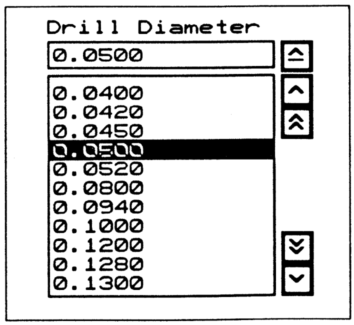


Figure 5-27. The ***Edit Hole*** *dialog box.*

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.

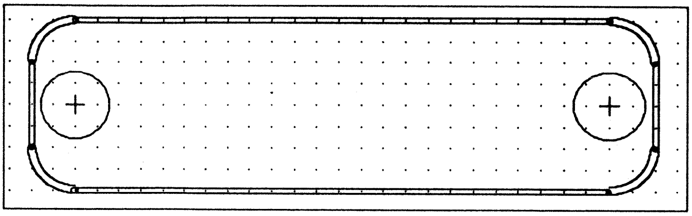
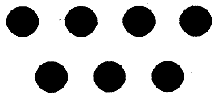


Figure 5-28. The module, with mounting holes added.

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).

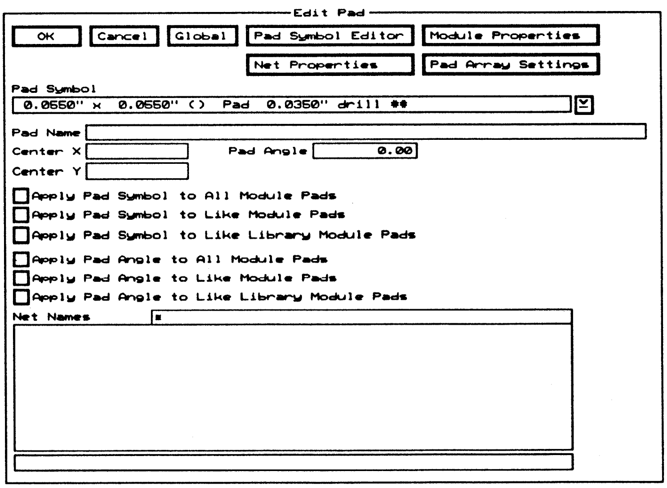


Figure 5-29. The ***Edit Pad*** *dialog box.*

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.

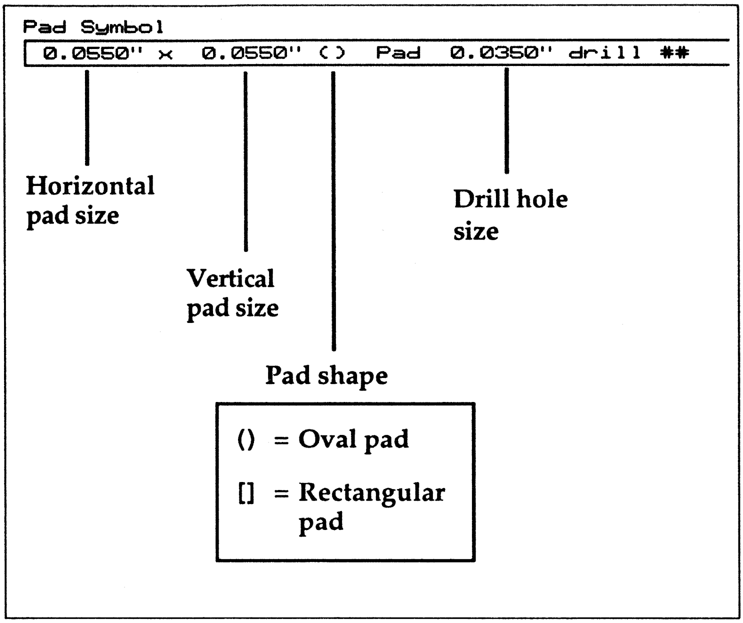


Figure 5-30. Description of pad symbol values.

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).

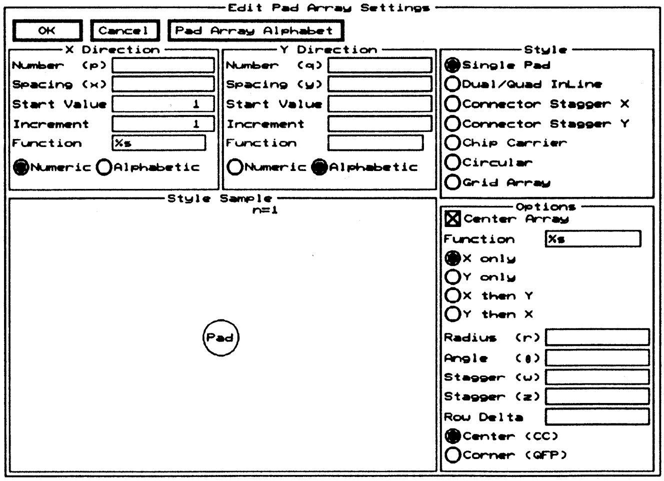


Figure 5-31. The ***Edit Pad Array Settings*** *dialog box.*

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.

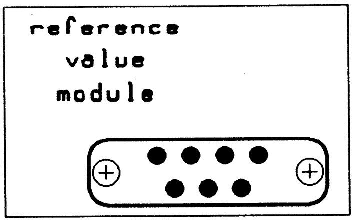


Figure 5-32. The pad array.

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.

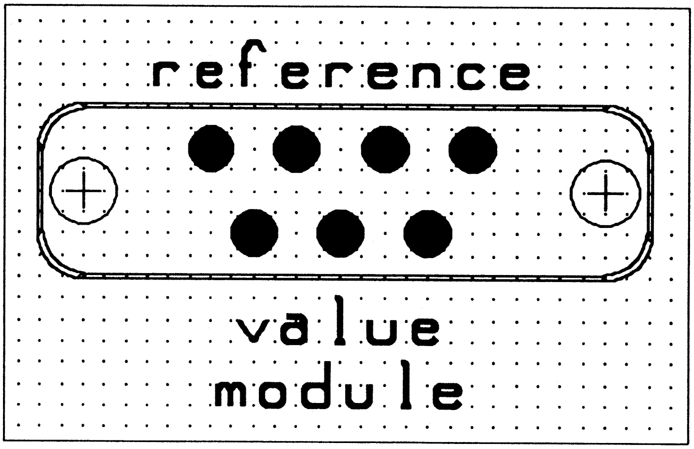


Figure 5-33. The positioned module placeholders.

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.

CHAPTER 6

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 to 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. Note that **Stay On Grid** is already enabled.
2. Enable **Outline Pads.**
3. Enable **Crosshair Cursor**. The full screen cursor makes it easier to draw and align objects.
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 TUTOR.MLB as the configured module library. If you did not configure TUTOR.MLB in the* ***Configuring PC Board Layout Tools*** *section of chapter 4, save this file as TUTOR.BD1, quit* ***Edit Layout****, and configure TUTOR.MLB using the instructions in chapter 4.*

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

```
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
```

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**, 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).

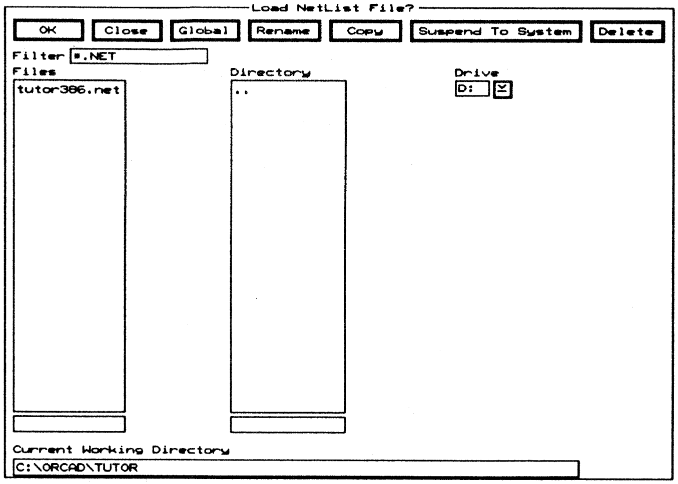


Figure 6-1. The ***Load Netlist File*** *dialog box.*

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

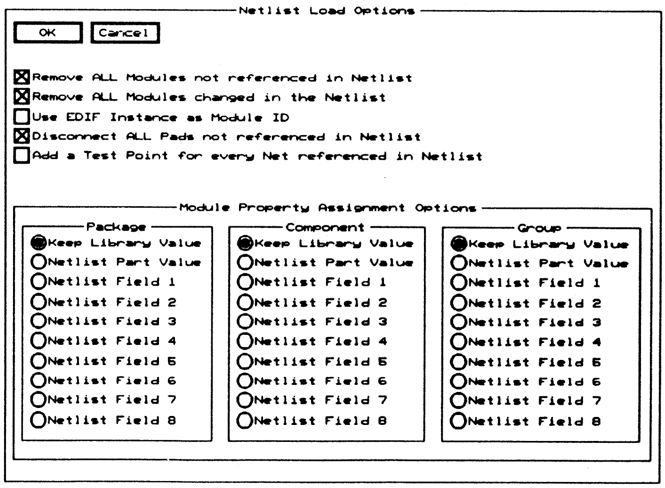


Figure 6-2. The ***Netlist Load Options*** *dialog box.*

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.

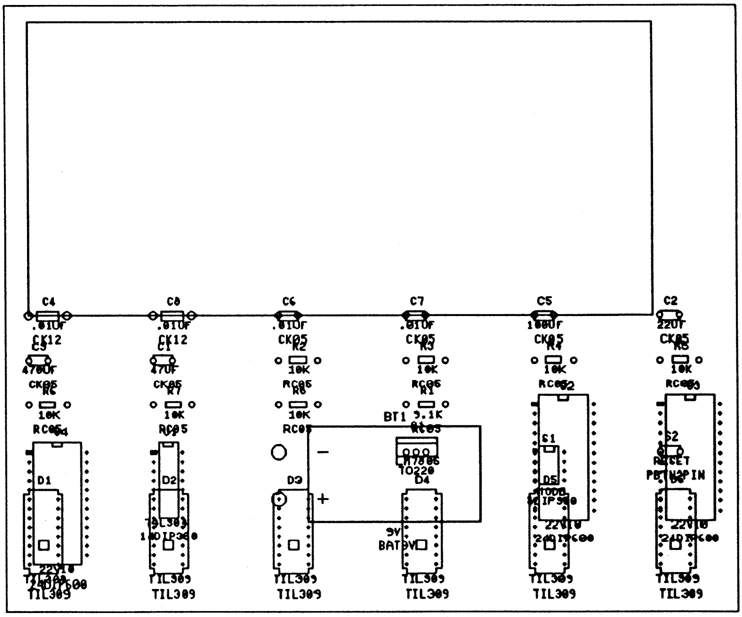


Figure 6-3. The TUTOR board, after loading TUTOR386.NET.

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.

```
Delete Block
Move Block
Drag Block
RatsNest Block
Module Snap Block
```

3. Select **RatsNest Block**. All modules in the block display their ratsnest connections (figure 6-4).

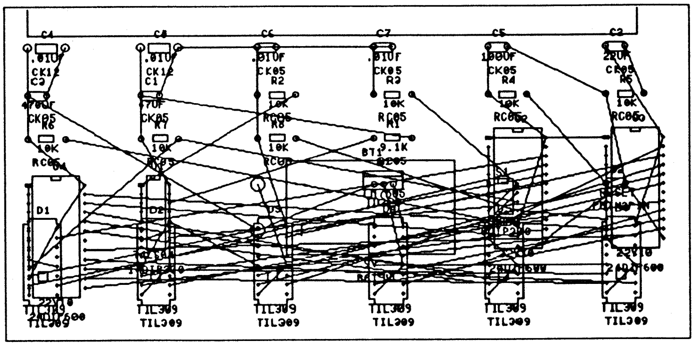


Figure 6-4. Ratsnest of the TUTOR layout.

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).

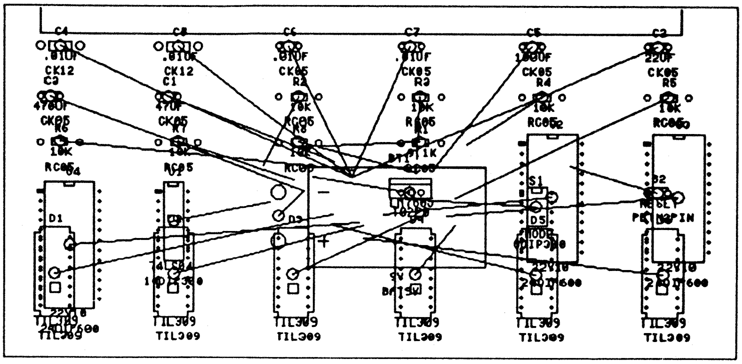


Figure 6-5. Force vectors for the TUTOR layout.

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).

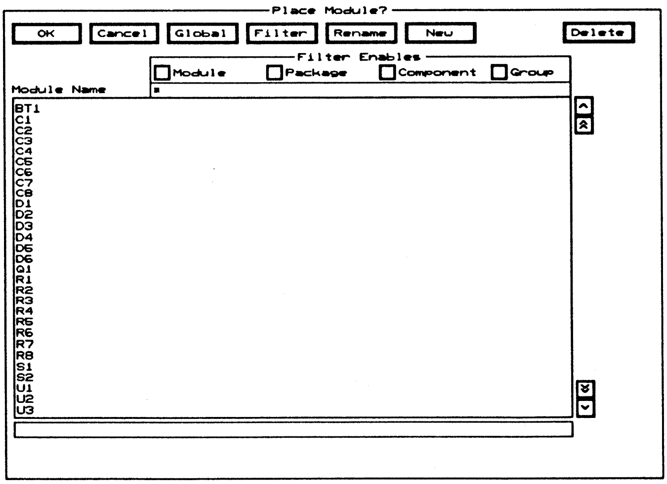


*Figure 6-6. The **Place Module** dialog box.*

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.

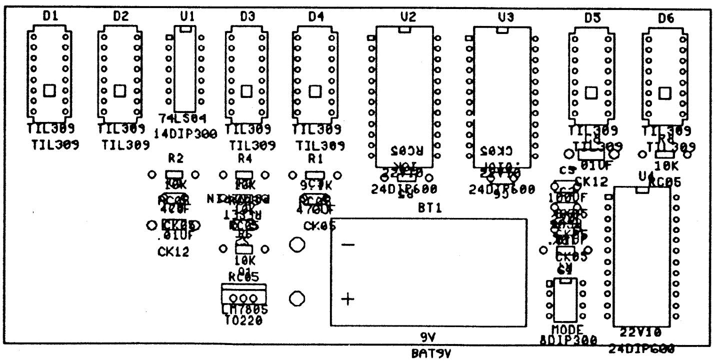


Figure 6-7. The placed TUTOR board.

Reference designator	*Block X value*	*Block Y value*	*Angle*
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.

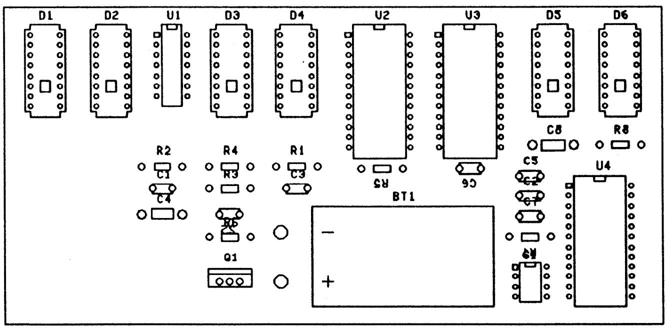


Figure 6-8. The TUTOR board, with hidden module text.

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

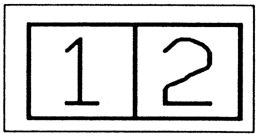
Now that the modules are properly placed on the board, you can identify an area of the board to place these additional objects:

- 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).

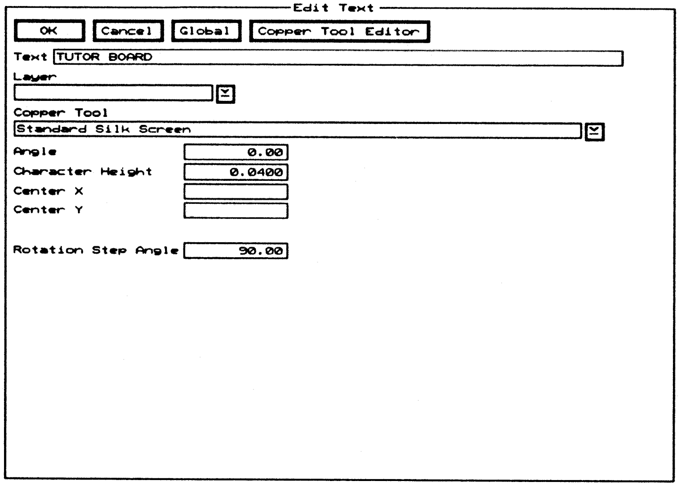
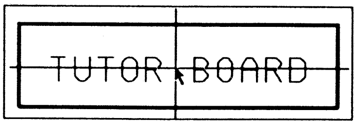


*Figure 6-9. The **Edit Text** dialog box.*

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.**

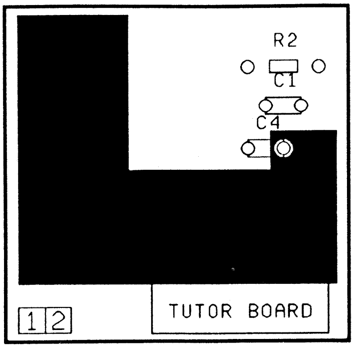
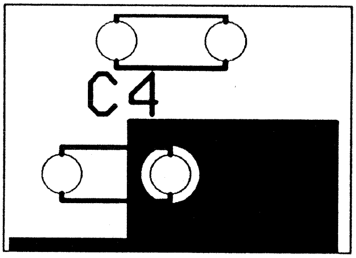


Figure 6-10. The completed fill zone.

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 (2.3000", 2.2750"), the center of the pad.
2. Select **INQUIRE**. The following message displays at the bottom of the screen:

```
C4 2 gnd Pad: 0.0750" x 0.0750"
```

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).

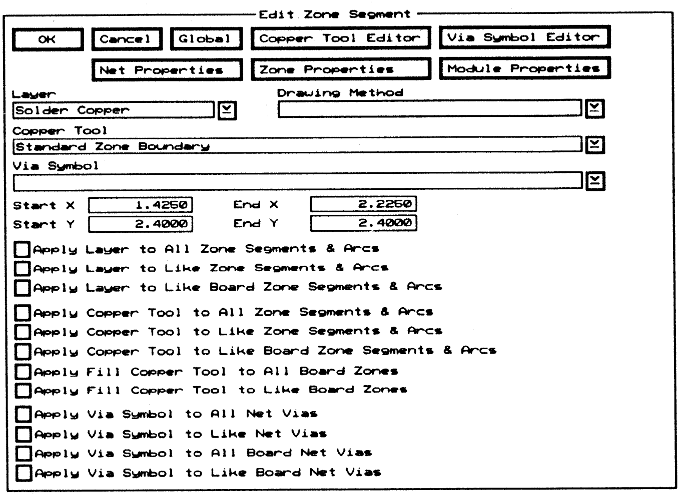


*Figure 6-11. The **Edit Zone Segment** dialog box.*

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

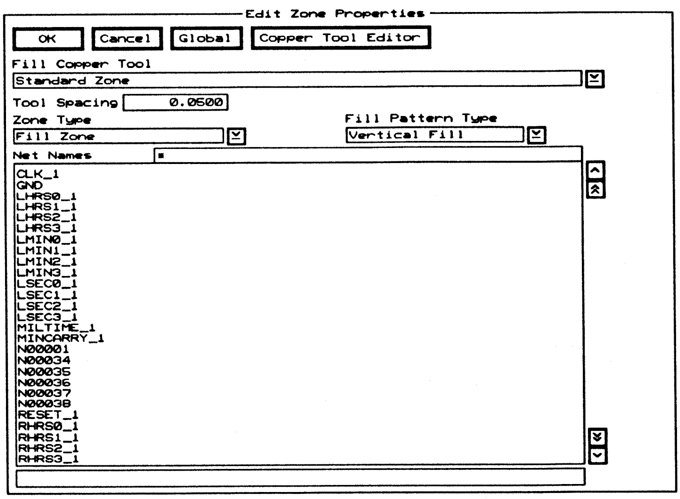
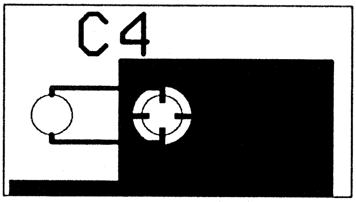


*Figure 6-12. The **Edit Zone Properties** dialog box.*

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.

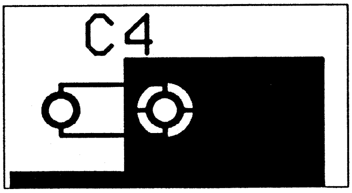


Viewing thermal relief

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.

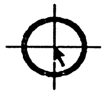
Refer to the *PC Board Layout Tools 386+ Reference Guide* for a description of thermal relief.

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).

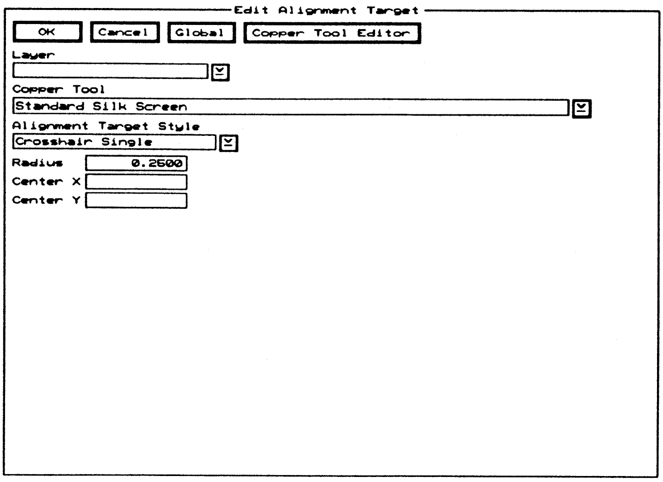
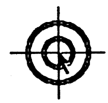


*Figure 6-13. The **Edit Alignment Target** dialog box.*

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).

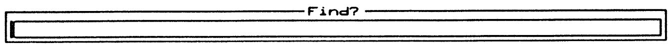


Figure 7-1. The ***Find*** *entry box.*

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

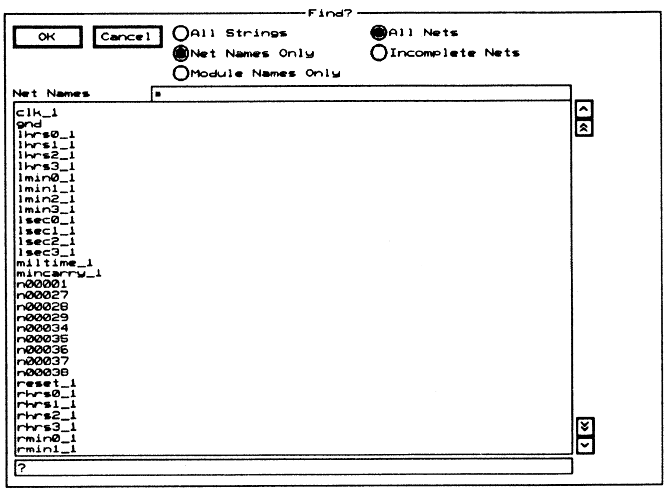


Figure 7-2 The ***Find*** *dialog box.*

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.

```
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
```

2. Select **Copper Tool Editor**. The **Edit Copper Tool** dialog box displays (figure 7-3).

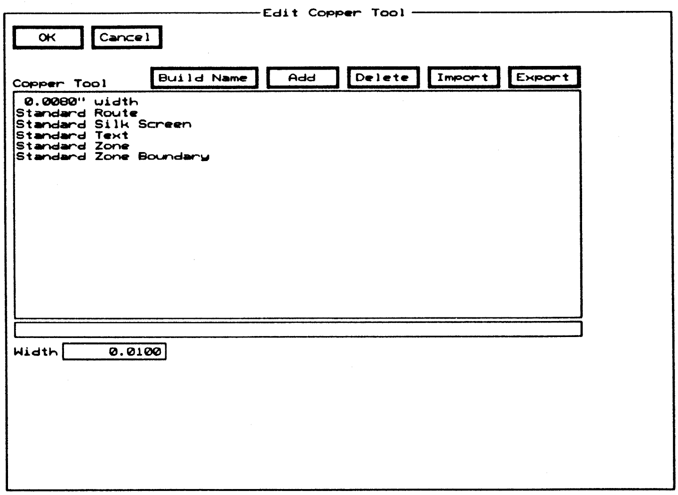
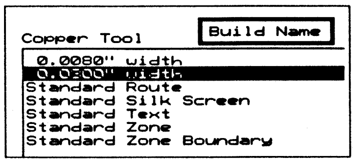


*Figure 7-3. The **Edit Copper Tool** dialog box.*

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**.

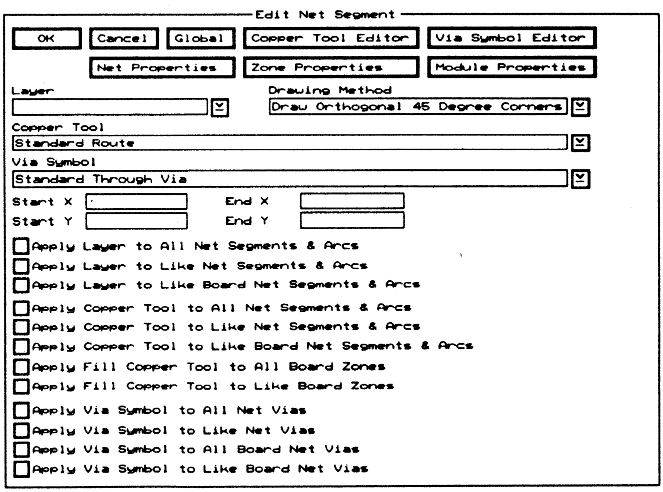
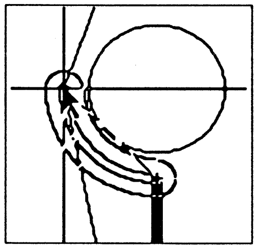
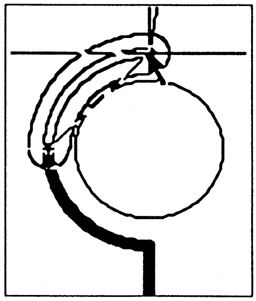


Figure 7-4. The Edit Net Segment dialog box.

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.

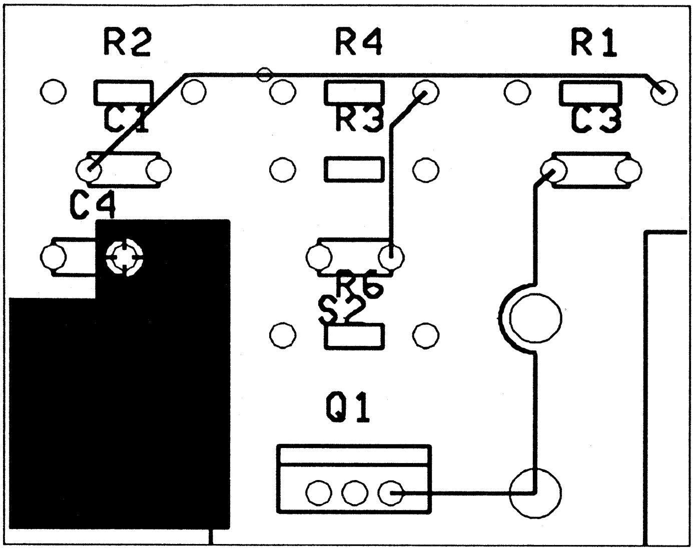


Figure 7-5. The three completed tracks.

Performing a DRC check

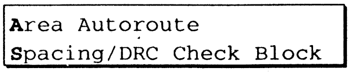
One of the tracks you routed is too close to several pads. The allowed clearance between tracks, pads, and vias is specified in **Copper To Copper Spacing** in **Global Options.** The default value is 0.0150 inches (15 mils).

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

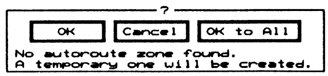
You perform a DRC check two ways in **Edit Layout:** a whole board check and a block check, which checks a selected area. Follow these steps to perform a block DRC check:

Running a block DRC check

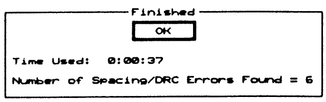
1. Move the pointer to (1.9000", 1.6000") and select **GO TO FUNCTION Autorouter.**
2. Select **Block**, then move the pointer and stretch the bounding box to (3.9000", 2.3250"), enclosing most of the routed area inside the block.
3. Select **Block End**. The menu shown at right displays.



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.

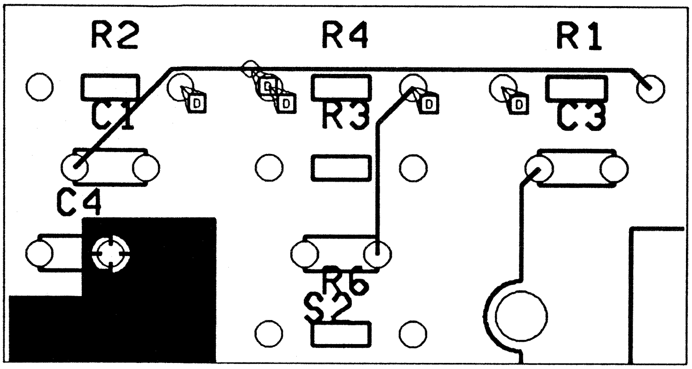
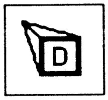


Figure 7-6. DRC markers indicating violations.

Identifying DRC violations

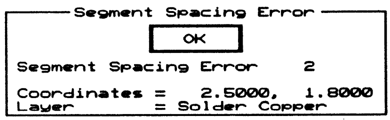
A DRC marker looks like the example shown at right. The converging lines point to the center of the violated object.



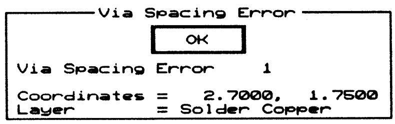
You identify DRC violations using two commands: **INQUIRE** and **JUMP**. Both methods are described in the following procedures.

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 at right displays.

   The dialog box identifies the type of violation, the coordinates for the violated object, and the layer where the violation occurred.
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 on the Solder Copper layer.

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**.

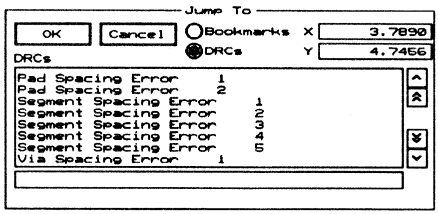


*Figure 7-7. The **Jump To** dialog box, showing design errors.*

   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.

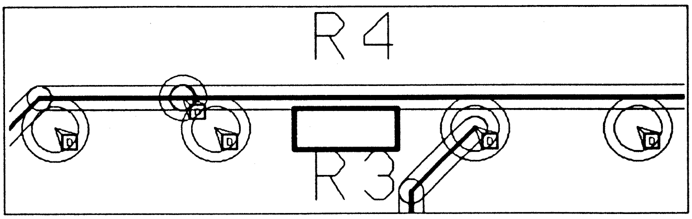


Figure 7-8. Copper to copper spacing of violated objects.

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 violations.

Editing the routed board

After manually routing a board, you may need to edit your work to correct DRC violations and make design modifications.

Correcting DRC violations

Follow these procedures to correct the DRC violations 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.

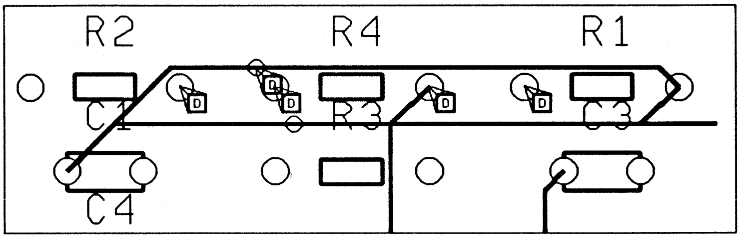


Figure 7-9. The old and new tracks for net N00037.

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. Move the pointer to (1.9000", 1.6000") and select **GO TO FUNCTION Autorouter.**
2. Select **Block**, then move the pointer to (3.9000", 2.3250"), enclosing the routed area. The menu at right displays.

```
Area Autoroute
Spacing/DRC Check Block
```

3. Select **Spacing/DRC Check Block**. In a moment, the dialog box shown at right displays. There are no DRC errors.

```
Finished
OK
Time Used:  0:00:12
Number of Spacing/DRC Errors Found = 0
```

4. Select **OK** to close the dialog box. Pan the display to 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** , then move the pointer to (4.0000", 3.1000") and select **Window Zoom End**. The display looks similar to figure 7-10.

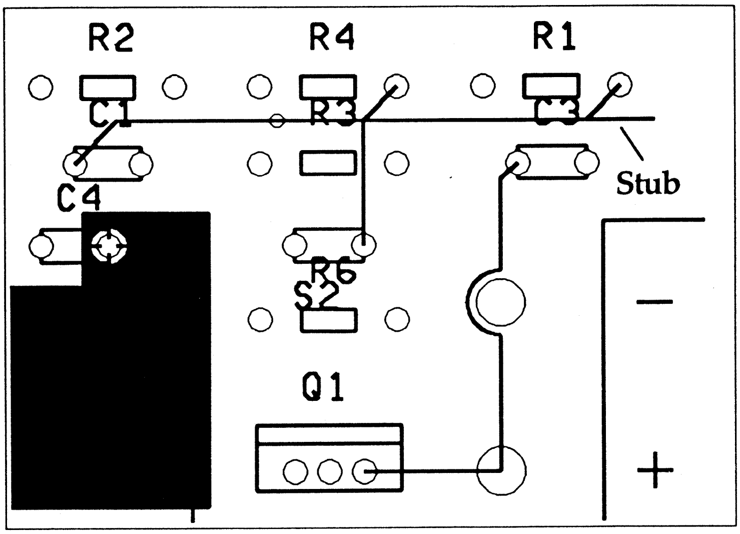


Figure 7-10. The edited routing area.

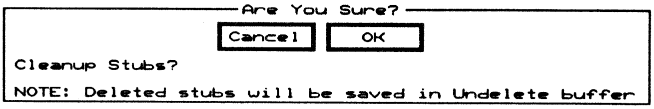
Deleting a stub

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:

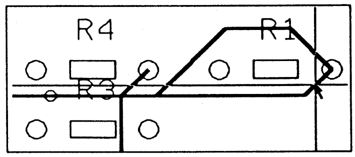
1. Place the pointer at (3.3000", 1.9000"), which positions it on a Component Copper segment of the track connecting C1 and R1.
2. Select **TRACK DELETE**. The entire track, including the via, is deleted.
3. Select **UNDELETE**. The entire track is recovered.

△ *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.*

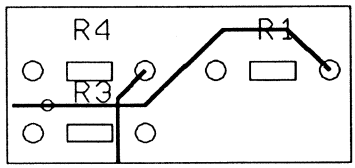
Changing a track path

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).

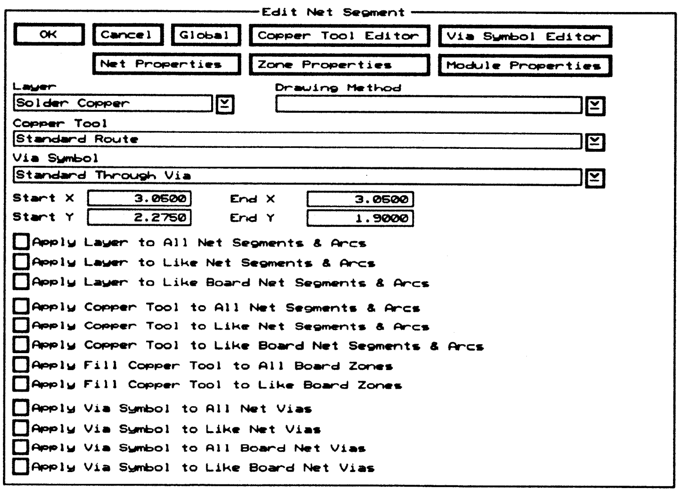


*Figure 7-11. The **Edit Net Segment** dialog box.*

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 you begin 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.

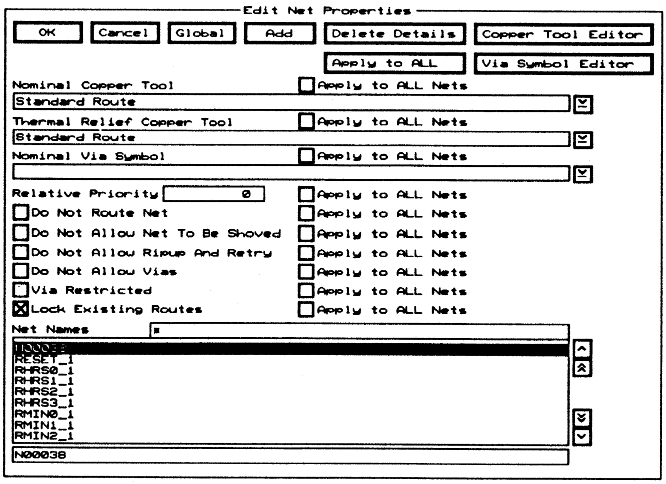


Figure 8-1. The ***Edit Net Properties*** *dialog box.*

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.

```
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
```

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.

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).

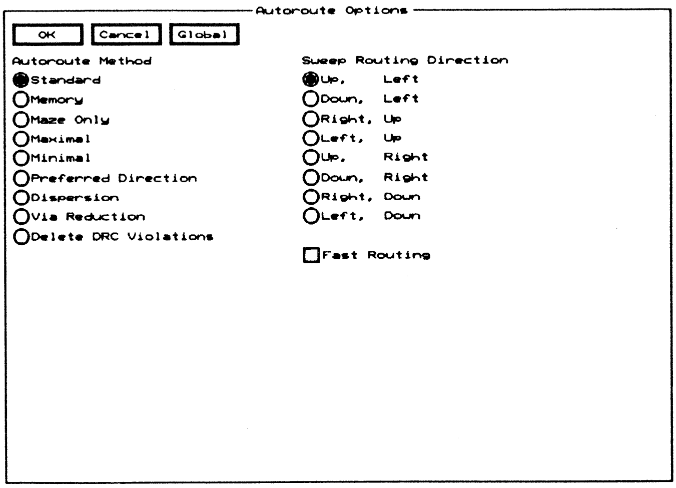


*Figure 8-2. The **Autoroute Options** dialog box.*

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).

```
Finished
[ OK ]
Time Used:  0:00:46
Total Number of Connections = 134
Number of Active Connections = 134
Number of Completed Connections = 12
Number of Partial Routes = 15
```

*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.

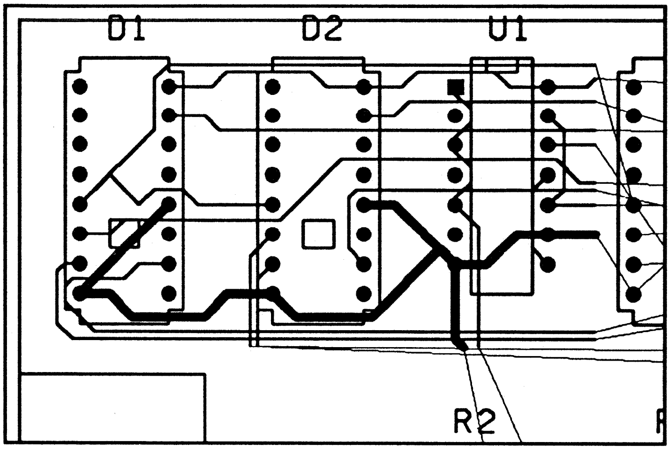


Figure 8-4. The finished autorouted area.

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.

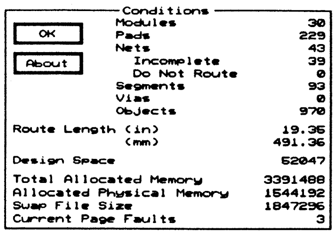


Figure 8-5. The ***Conditions*** *dialog box.*

7. Select **OK** to close the 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.

```
Autoroute Whole Board
Spacing/DRC Check Whole Board
Set Sweep Window
```

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.

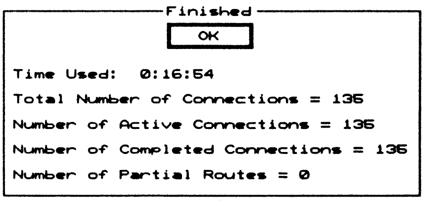
Begin autorouting

1. With the menu still displayed, select **Autoroute Whole Board** to begin autorouting.

   The display zooms 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.

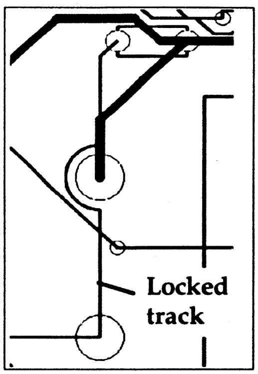


Figure 8-6. The locked track.

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.

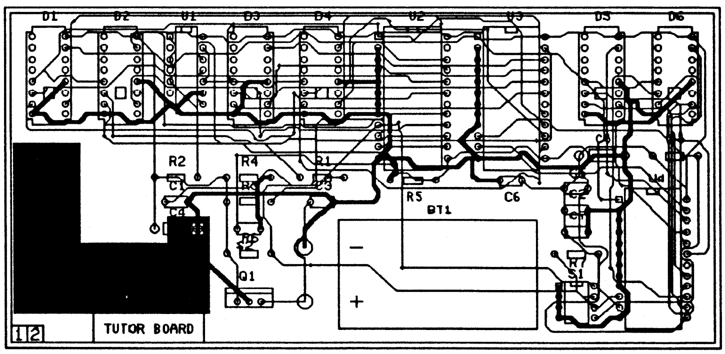


Figure 8-7. The autorouted TUTOR board, after via reduction.

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.

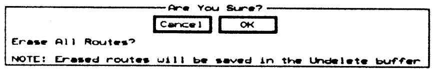
Follow these steps to perform another via reduction:

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.

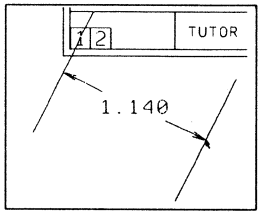
Placing dimensions

Edit Layout has many options for placing dimensions. Follow these steps to place dimensions on the board:

Placing the first dimension

1. Make sure **Assembly Drawing** is 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.

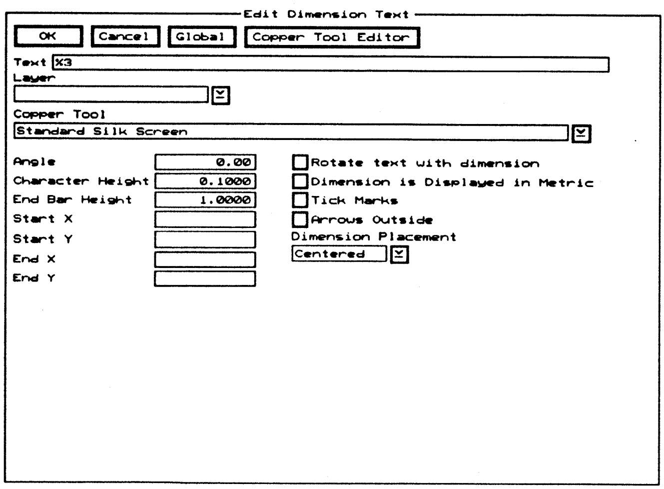
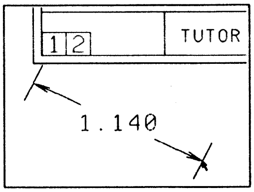


Figure 8-8. The **Edit Dimension Text** *dialog box.*



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.

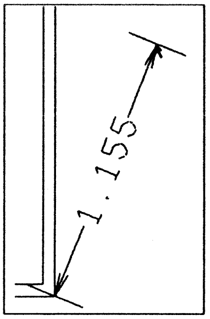
10. Press <Esc> to dismiss the additional dimension text.

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.

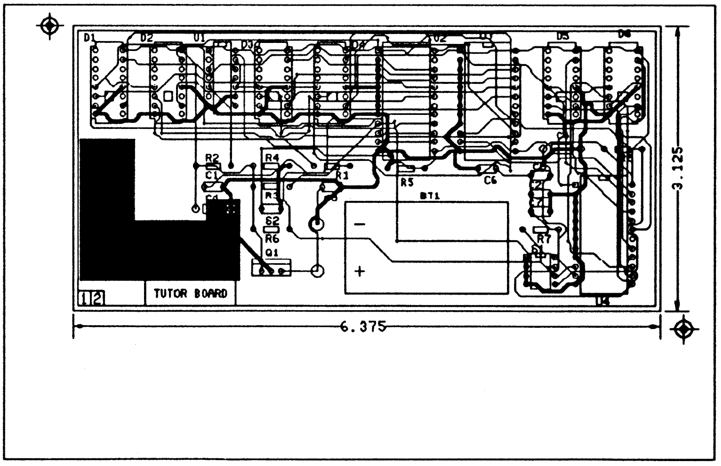


Figure 8-9. The completed TUTOR board.

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
```

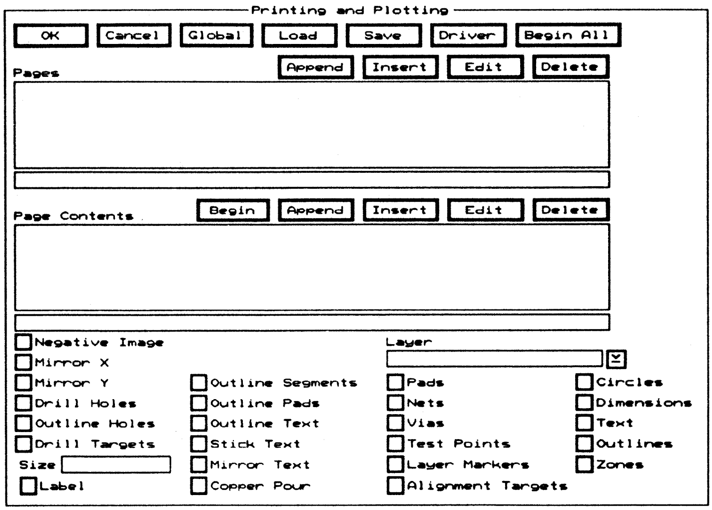


*Figure 9-1. The **Printing and Plotting** dialog box.*

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).

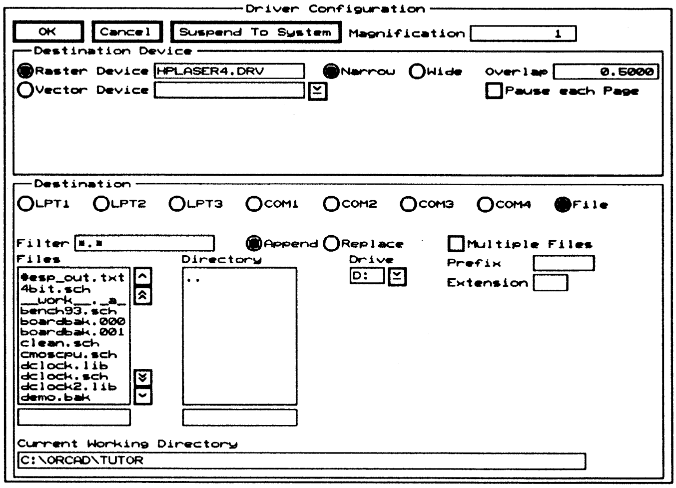


Figure 9-2. The ***Driver Configuration*** *dialog box.*

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 the **Destination Device** area.

- ❖ **File** is selected in the **Destination** area, and the contents of the current working directory display in the **Files** list box.
- ❖ **Replace** is selected in the **Destination** area. When this option is selected, the previous printer file that was written to the disk is replaced by the new printer file with the same filename.

2. Select **LPT1** in the **Destination** area 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. Enter **TOP COPPER LAYER** in the entry box below the **Pages** list box.

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.

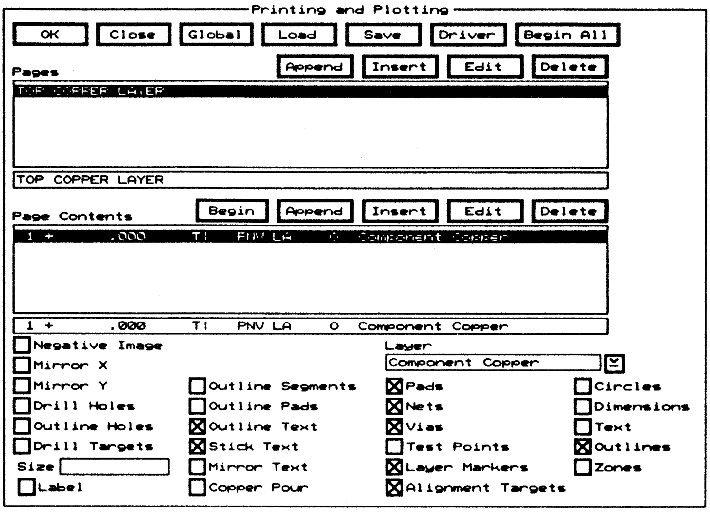


Figure 9-3. The configured TOP COPPER LAYER page.

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.

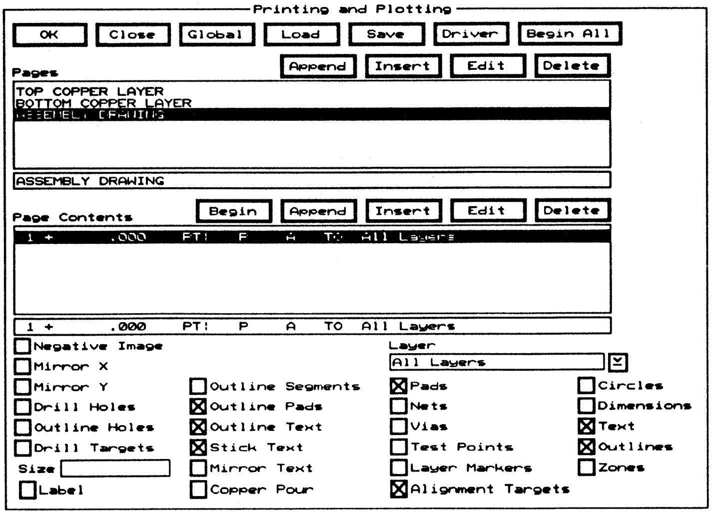


Figure 9-4. The configured printer pages.

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 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 TOP COPPER LAYER. 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 printed pages look similar to figures 9-5 through 9-7.

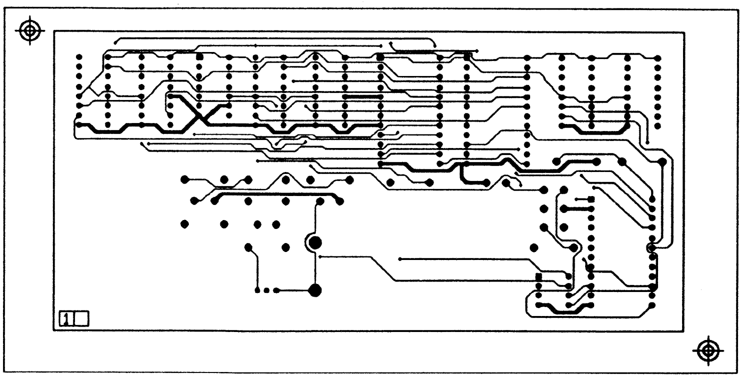


Figure 9-5. The TOP COPPER LAYER page.

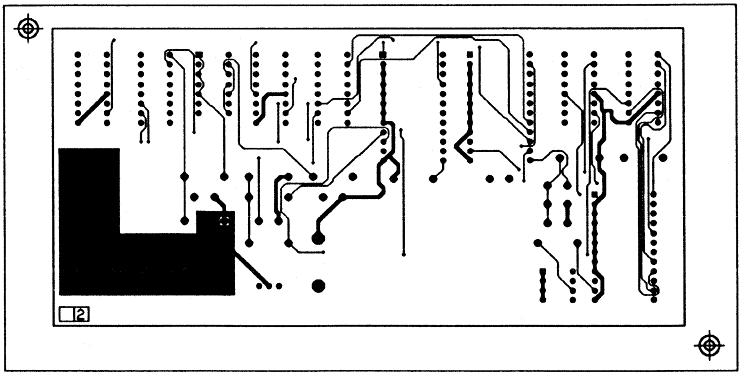


Figure 9-6. The BOTTOM COPPER LAYER page

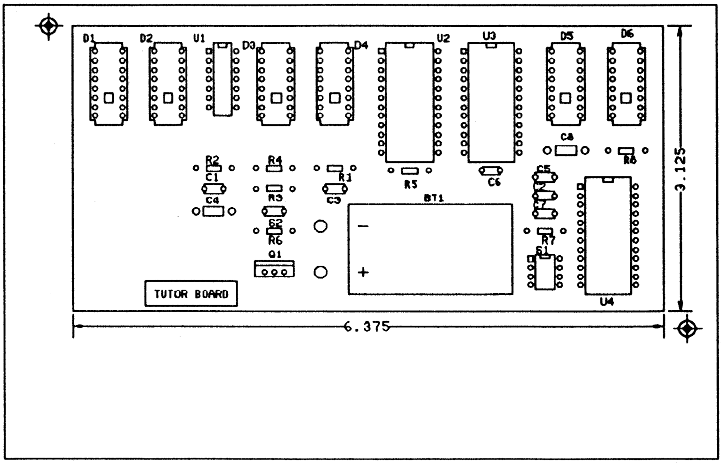


Figure 9-7. The ASSEMBLY DRAWING page.

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).

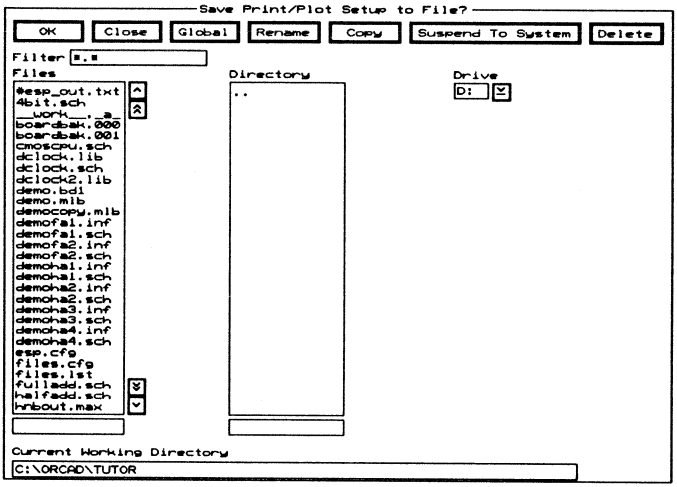


Figure 9-8. The ***Save Print/Plot Setup to File*** *dialog box.*

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).

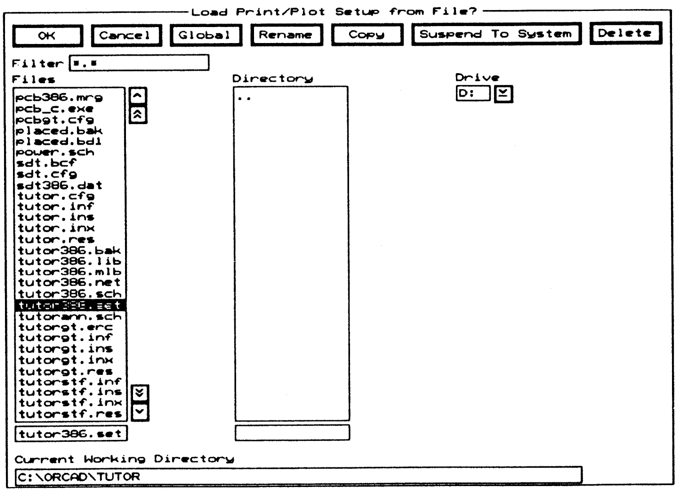


*Figure 9-9. The **Load Print/Plot Setup from File** dialog box.*

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:

- Gerber (274-X)
- Fire 9xxx
- Gerber (274-D)
- HP-GL/2
- HP-GL (outline mode only)
- Postscript

Plotting all pages to Gerber (274-X) and Fire 9xxx

The Gerber (274-X) and Fire 9xxx drivers embed tool list codes into the Gerber file. These drivers produce Gerber files compatible with contemporary photoplotters and laser imaging equipment.

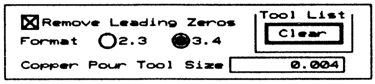
Produce Gerber (274-X) or Fire 9xxx Gerber files for all pages using the following procedures.

Configuring pages

1. From the main menu, select **GO TO FUNCTION Printing and Plotting** to display the **Printing and Plotting** dialog box.
2. Configure your pages using the procedures described in the **Configuring pages** section earlier in this chapter, or load the saved printer setup file TUTOR386.SET.

Configuring Gerber (274-X) and Fire 9xxx drivers

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)** or **Fire 9xxx**. The options shown at right display below the **Vector Device** droplist box. The **Clear** button in the **Tool List** area is inactive.



4. Leave **Remove Leading Zeros** enabled. This removes leading zeros from coordinates in the Gerber file.
5. Select a format for the photoplotter. The selections are **2.3** or **3.4**. These determine the number of digits to the left and right of the assumed decimal point for coordinates in the Gerber file. The default selection is **3.4**.
6. Note the default value of **0.004** in the **Copper Pour Tool Size** entry box. This specifies the width, in inches, for a copper tool that is used to draw the interiors of fill zones. Enter another value if you want fill zone interiors drawn with a different width.

△ *NOTE: Specifying a width smaller than 0.004 increases the size of the Gerber file, but also increases the resolution for pad isolation and thermal relief within the fill zone. Specifying a width larger than 0.004 decreases the size of the Gerber file, but also decreases the resolution within the zone. The recommended maximum copper pour tool size is 0.006 inches (6 mils).*

Specifying the Gerber filenames

1. Leave **File** selected in the **Destination** area. Plotter files are written to your current design directory when **File** is selected.
2. Enable **Multiple Files**. The **Prefix** and **Extension** entry boxes become active.

   The **Prefix** entry box contains the part of the filename that is shared by the multiple Gerber files. You can enter up to six characters. When the Gerber files are created, sequential digits from 00 to 99 are appended to each of the filenames.

   The **Extension** entry box contains the file extension for the Gerber files. You can enter up to three characters.
3. Enter TUTOR in the **Prefix** entry box.
4. Enter GBR in the **Extension** entry box.
5. Select **OK** to display the **Printing and Plotting** dialog box.

Producing the Gerber files

Select **Begin All** to produce the three Gerber files. When the files are done, the **Printing and Plotting** dialog box displays. If you use the recommended OrCAD directory structure, the following Gerber files are written to your \ORCAD\TUTOR directory:

```
TUTOR00.GBR
TUTOR01.GBR
TUTOR02.GBR
```

Clearing the tool list

All tool list information required by the Gerber (274-X) and Fire 9xxx driver is read from the board file, stored in memory, and embedded into the Gerber files. Follow these steps to delete the tool list from memory:

1. Select **Driver** to display the **Driver Configuration** dialog box. The **Clear** button in the **Tool List** area is now active.
2. Select **Clear** to clear the tool list from memory. You can retain the tool list if you need to produce additional Gerber files.
3. Select **OK** to display the **Printing and Plotting** dialog box.

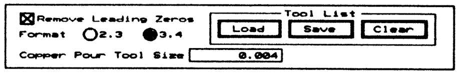
Plotting all pages to Gerber (274-D)

The Gerber (274-D) driver produces Gerber files without embedded tool codes. The tool list is created separately, saved as a text file, and supplied with the Gerber files. The Gerber (274-D) driver serves a class of mechanical photoplotters that require operator-entered tool lists to specify aperture settings on the photoplotter aperture wheel.

The remaining procedures in the *Plotting* section assume you have all pages configured. Produce Gerber (274-D) Gerber files for all pages using the following procedures.

Configuring the Gerber (274-D) driver

1. From the **Printing and Plotting** dialog box, 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–D)**. The options shown at right display below the **Vector Device** droplist box. The **Load** button is the only active selection in the **Tool List** area. You select **Load** when you want to load a Gerber (274-D) tool file into memory.



4. Perform the same driver configuration and Gerber filename procedures that are described in the *Plotting all pages to Gerber (274-X) and Fire 9xxx* section earlier in this chapter.

Producing the Gerber files and saving the tool list

1. Select **Begin All** from the **Printing and Plotting** dialog box. All copper tools and objects in the board file are scanned, and a Gerber (274-D) tool list is created in memory. The following Gerber files are produced:

   ```
   TUTOR00.GBR
   TUTOR01.GBR
   TUTOR02.GBR
   ```

   The tool list in memory lists aperture sizes and shapes that the Gerber (274-D) plotter needs to plot tracks, pads, and other objects in the board file. You need to save the tool list to a file so it can accompany the Gerber files.

2. Select **Driver** to display the **Driver Configuration** dialog box. Now the **Save** and **Clear** buttons in the **Tool List** area are also active.

3. Select **Save**. The **Save Tool List to File** dialog box displays (figure 9-10).

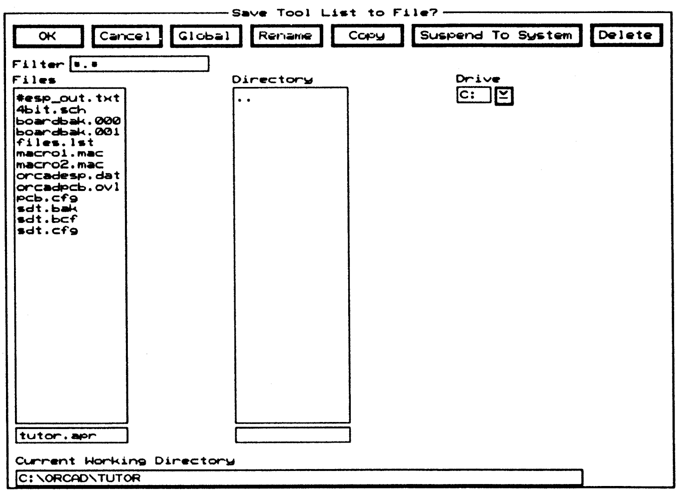


*Figure 9-10. The **Save Tool List to File** dialog box.*

4. Enter **`TUTOR.APR`** in the entry box beneath the **Files** list box, then select **OK** to save the tool list in your current working directory. The **Driver Configuration** dialog box displays.

Clearing the tool list in memory

You can clear the tool list in memory, or you can retain it and load another board file. When you produce Gerber (274-D) plot files for the new board file, the new tool list is added to the tool list already in memory. The tool list in memory is automatically cleared when you quit **Edit Layout**.

To clear the tool list, select **Clear** from the **Driver Configuration** dialog box. The tool list is removed from memory and the **Clear** and **Save** buttons become inactive.

Displaying the tool list

If your system has a 80387 math coprocessor, or if it has an 80486 processor with coprocessing functions, you can display the tool list using the following steps:

1. Select **Suspend To System** from the **Driver Configuration** dialog box. **Edit Layout** is suspended, the DOS prompt displays, and your current directory is C:\ORCAD\TUTOR if you follow the default OrCAD directory structure.
2. Enter `TYPE  TUTOR.APR` to display the tool list (figure 9-11).

```
# Format, Version
%IMPERIAL, V3.1
#
# Author:    OrCAD PCB 386+      Version   V1.10g   02-Dec-93
#            (C) Copyright 1985-1993  OrCAD, Inc.  ALL RIGHTS RESERVED.
#            OrCAD is a Registered Trademark of OrCAD, Inc.
#
# Written:   Mon Dec  6 06:33:47 1993
#
# D   Shape      Width  Height    Type  Tool    Tool Size    Legend    R90
#
D10   Round      0.0080 0.0080     TH    0      0.0000       D10       0
D11   Round      0.0750 0.0750     TH    0      0.0000       D11       0
D12   Square     0.0550 0.0550     TH    0      0.0000       D12       0
D13   Round      0.0550 0.0550     TH    0      0.0000       D13       0
D14   Round      0.1450 0.1450     TH    0      0.0000       D14       0
D15   Round      0.0700 0.0700     TH    0      0.0000       D15       0
D16   Round      0.0100 0.0100     TH    0      0.0000       D16       0
D17   Round      0.0450 0.0450     TH    0      0.0000       D17       0
D18   Round      0.0300 0.0300     TH    0      0.0000       D18       0
D19   Round      0.0020 0.0020     TH    0      0.0000       D19       0
D70   Round      0.0040 0.0040     TH    0      0.0000       D70       0
```

Figure 9-11. The TUTOR.APR tool list.

3. Enter `EXIT` to end the DOS session and return to the **Driver Configuration** dialog box.

If your system does not have a coprocessor or coprocessing functions, follow these steps.

1. Select **Cancel** from the **Driver Configuration** dialog box, then select **Cancel** from the **Printing and Plotting** dialog box.
2. Select **QUIT Abandon Program.** The **PC Board Layout Tools** screen displays.
3. Select an unoccupied area of the **PC Board Layout Tools** screen. The menu at right displays.

```
Design Management Tools
Suspend to System
Vendor Selection
Show Hot Key Assignments
Exit
```

4. Select **Suspend to System**. The DOS prompt displays, and you can display the tool list by entering `TYPE TUTOR.APR`.
5. Enter `EXIT` to return to the **PC Board Layout Tools** screen.
6. Select **Edit Layout**, then select **Execute** to reload the board file.
7. Select **GO TO FUNCTION Printing and Plotting** to display the **Printing and Plotting** dialog box.
8. Load TUTOR386.SET, the printing setup file for the TUTOR board you saved earlier in this chapter, by selecting **Load.** The **Load Print/Plot Setup from File** dialog box displays.
9. Select **TUTOR386.SET** from the **Files** list box and select **OK**. The saved pages and page contents display in the **Printing and Plotting** dialog box.

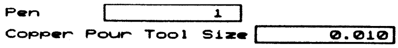
Plotting all pages to an HP-GL/2 or HP-GL plotter

The HP-GL/2 and HP-GL plotter drivers produce vector output on pen plotters that support the HP-GL/2 or HP-GL plotting language. Pen plotters are used to produce high resolution plots for reviewing the final board design, as well as master artwork. The HP-GL driver plots in outline mode only.

The following procedures also assume you have all pages configured. Produce Gerber HP-GL/2 or HP-GL plotter files for all pages using the following procedures:

Configuring HP-GL/2 and HP-GL

1. From the **Printing and Plotting** dialog box, select **Driver**. The **Driver Configuration** dialog box displays.
2. Select **Vector Device**, then select the droplist button to display the droplist box.
3. Select **HPGL2** or **HPGL** from the droplist box, depending on the type of plotter you have. The options shown at right display below the **Vector Device** droplist box.



4. Enter a value in the **Pen** entry box that corresponds to the pen slot in the plotter carrousel containing the pen you will use to plot the pages. The default number is 1.
5. Enter a value in the **Copper Pour Tool Size** entry box. The value specifies the line width used to plot the interior of a fill zone. The default value is 0.010 inches.

Plotting directly to the plotter

1. Select the configured printer port in the **Destination** area. If your plotter is connected to printer port LPT1, select **LPT1**.
2. Select **OK** to display the **Printing and Plotting** dialog box.
3. Select **Begin All** to plot the three pages directly to the plotter.

Plotting to files

1. With the **Driver Configuration** dialog box displayed, select **File** in the **Destination** area. Plotter files are written to your current design directory when **File** is selected.
2. Enable **Multiple Files**. The **Prefix** and **Extension** entry boxes become active.
3. Enter **TUTOR** in the **Prefix** entry box.
4. Enter **HPG** in the **Extension** entry box.
5. Select **OK** to display the **Printing and Plotting** dialog box.
6. Select **Begin All** to produce the three HPGL/2 or HPGL files. When the files are done, the **Printing and Plotting** dialog box displays. If you use the recommended OrCAD directory structure, the following files are written to your C:\ORCAD\TUTOR directory:

```
TUTOR00.HPG
TUTOR01.HPG
TUTOR02.HPG
```

Copying the files to the plotter

You can use the COPY command from DOS to send your plot files to the plotter. Follow these steps if your system has an 80387 math coprocessor, or if it has an 80486 processor with coprocessing functions:

1. Select **Driver** from the **Printing and Plotting** dialog box. The **Driver Configuration** dialog box displays.
2. Select **Suspend To System. Edit Layout** is suspended, and you open a new DOS editing session.The DOS prompt displays, and you are in the current working directory (C:\ORCAD\TUTOR).
3. If your plotter is connected to LPT1, enter the following command:

```
C:\ORCAD\TUTOR>>COPY TUTOR00.HPG LPT1:
```

4. When the plotter completes TUTOR00.HPG, plot TUTOR01.HPG and TUTOR02.HPG using the same command syntax.
5. When all plots are complete, return to the **Driver Configuration** dialog box by entering **EXIT** at the DOS prompt.
6. Select **OK** to display the **Printing and Plotting** dialog box.

If your system does not have an 80387 coprocessor or built-in coprocessing functions, follow these steps to use the DOS COPY command to plot the files:

1. Select **Cancel** from the **Printing and Plotting** dialog box. The board file displays.
2. Select **QUIT Abandon Program.** The **PC Board Layout Tools** screen displays.
3. Select an unoccupied area of the **PC Board Layout Tools** screen. The menu at right displays.

```
Design Management Tools
Suspend to System
Vendor Selection
Show Hot Key Assignments
Exit
```

4. Select **Suspend to System.** The DOS prompt displays, and you can copy the files to the plotter using the procedures in the *Copying the files to the plotter* section.
5. Enter `EXIT` to return to the **PC Board Layout Tools** screen.
6. Select **Edit Layout**, then select **Execute** to reload the board file.
7. Select **GO TO FUNCTION Printing and Plotting** to display the **Printing and Plotting** dialog box.
8. Load TUTOR386.SET, the printing setup file for the TUTOR board you saved earlier in this chapter, by selecting **Load**. The **Load Print/Plot Setup from File** dialog box displays.
9. Select **TUTOR386.SET** from the **Files** list box and select **OK**. The saved pages and page contents display in the **Printing and Plotting** dialog box.

If you run **Edit Layout** as a full screen DOS application under Microsoft Windows, follow these steps to use the DOS COPY command to plot the files:

1. Press <Alt><Tab> until the Windows Program Manager displays.
2. Select the **MS-DOS Prompt** icon. The DOS prompt displays.
3. If your Windows and ORCAD directories are on the same drive, change to the \ORCAD\TUTOR directory by entering `CD \ORCAD\TUTOR`.
4. Copy the files to the plotter using the procedures in the *Copying the files to the plotter* section.
5. Enter `EXIT` to return to the Windows Program Manager.
6. Press <Alt><Tab> to switch back to **Edit Layout**. The **Printing and Plotting** dialog box displays.

Plotting all pages to a Postscript printer

1. Select **Driver** from the **Printing and Plotting** dialog box to display the **Driver Configuration** dialog box. A postscript printer should be connected to your computer.
2. Select **Vector Device**, then select the droplist button and select **PostScript** from the droplist box. The **Copper Pour Tool Size** entry box displays below the selected vector device and 0.002 inches is the default value.
3. Select the configured printer port in **Destination.** If your printer is connected to LPT1, select **LPT1.**
4. Select **OK** to display the **Printing and Plotting** dialog box.
5. Select **Begin All** to print the three pages.

Plotting to postscript files

1. With the **Driver Configuration** dialog box displayed, select **File** in the **Destination** area.
2. Enable **Multiple Files**. The **Prefix** and **Extension** entry boxes become active.
3. Enter TUTOR in the **Prefix** entry box.
4. Enter PS in the **Extension** entry box.
5. Select **OK** to display the **Printing and Plotting** dialog box.
6. Select **Begin All** to produce the three postscript files. When the files are done, the **Printing and Plotting** dialog box displays.

Copying the files to a postscript printer

To copy the files to a postscript printer, follow the procedures described in the *Copying the files to the plotter* section, substituting TUTOR00.PS, TUTOR01.PS, and TUTOR02.PS for the filenames.

Plotting a selected page

The following procedures describe how to select only the BOTTOM COPPER LAYER page, configure the destination, and plot the single page to a plotter or to a file. The procedures apply to all vector devices.

Plotting a page to a plotter

1. Select **BOTTOM COPPER LAYER** in the **Pages** list box of the **Printing and Plotting** dialog box.
2. Select **Edit** above **Pages.** The item in **Page Contents** changes to the page contents of the BOTTOM COPPER LAYER page. The selected layer and enabled options change to reflect the settings of the new item in **Page Contents.**
3. Select **Driver** to display the **Driver Configuration** dialog box.
4. Select a vector device and configure a plotter port connection. If your plotter is connected to LPT1, select **LPT1** in the **Destination** area.
5. Select **OK.** The **Printing and Plotting** dialog box displays.
6. Select **Begin** to plot the BOTTOM COPPER LAYER page. When the plot is completed, the **Printing and Plotting** dialog box displays.

Plotting a page to a file

1. Perform steps 1 through 3 listed in the *Plotting a page to a plotter* section.
2. Select a vector device, then select **File** in the **Destination** area.
3. Enter a filename and extension in the entry box beneath the **Files** list box.
4. Select **OK.** The **Printing and Plotting** dialog box displays.
5. Select **Begin** to plot the BOTTOM COPPER LAYER page to the specified filename. When the file is completed, the **Printing and Plotting** dialog box displays.

Phar Lap technical information

This appendix provides detailed information about the Phar Lap memory extender.

About the CFIG386 utility

All Phar Lap programs support the use of command line switches to override the default operation of the program. The CFIG386 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 CFIG386 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 CFIG386

The command line format for CFIG386 is the following:

CFIG386 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 CFIG386 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 CFIG386 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 -HWIVEC and -PRIVEC 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 -HWIVEC 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 -PRIVEC 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

```
-HWIVEC vector
-PRIVEC vector
```

Short form

```
-HWI vector
-PRI vector
```

Example

```
run386 -hwivec 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's 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 388 | 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 vmmdrv hello
minibug -swd e:\tmp -vm vmmdrv 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

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Schematic Design Tools

User's Guide

OrCAD®

Electronic Design Automation Tools

Schematic Design Tools

User's Guide

McBoole is a public domain process developed by Michel Dagenais of McGill University.

Fourth Edition 17 Mar 1992

OrCAD®

3175 NW Aloclek Drive
Hillsboro, Oregon 97124-7135
U.S.A.

Sales & Administration	(503) 690-9881
Technical Support	(503) 690-9722
24-Hour Bulletin Board System	(503) 690-9791
FAX	(503) 690-9891

Contents

Welcome to OrCAD Schematic Design Tools

Welcome to practical electronic engineering. You now own **OrCAD Schematic Design Tools**, a design automation tool set with the power of an engineering workstation. You can complete complex design tasks using **Schematic Design Tools** in a fraction of the time it takes by hand.

Developed specifically to run on personal computers, **Schematic Design Tools** supports most popular graphics boards, printers, and plotters.

Finding the information you need

These five manuals accompany **Schematic Design Tools**:

- *Installation & Technical Support Guide*
- *OrCAD/ESP Design Environment User's Guide*
- *Stony Brook M2EDIT Text Editor User's Guide*
- *Schematic Design Tools User's Guide*
- *Schematic Design Tools Reference Guide*

Installation

Before you begin to explore **Schematic Design Tools**, take a few minutes to install the tool set and register for technical support. Just follow the instructions in the *Installation & Technical Support Guide.*

Project-oriented design environment

Schematic Design Tools is one part of a fully integrated *Electronic Design Automation* (EDA) system. The design environment is structured so 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, part lists, simulation results, and board layouts—stored together.

The *OrCAD/ESP Design Environment User's Guide* introduces the graphical environment under which **Schematic Design Tools** and the other OrCAD tool sets operate. In this environment, OrCAD tools and tool sets, such as **Schematic Design Tools**, 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 main screen, even if you only have one tool set installed on your computer.

Learning Schematic Design Tools

This *User's Guide* introduces **Schematic Design Tools.** The best way to get to know **Schematic Design Tools** is to start with *Chapter 2: Introducing Schematic Design Tools,* then proceed chapter-by-chapter through this book. You will be guided through several practice sessions that show you the basics about using **Schematic Design Tools.**

Beyond the basics

Once you have mastered the basics, refer to the *Schematic Design Tools Reference Guide* for information that will help you plan and create your design. The *Reference Guide* explains how to tailor the configuration of the software to match your personal requirements, provides detailed information about commands and concepts, and tells how to transfer a design between OrCAD tool sets. It is designed to be a continuing source of instruction and reference as you use **Schematic Design Tools.**

The design environment

Schematic Design Tools is one part of a fully integrated electronic design automation environment. Using this design environment you can:

- Run the tools within a tool set. (The tools that make up **Schematic Design Tools** are listed in the next section.)
- Move between tool sets without switching directories or copying files.
- Configure 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.
- Organize designs by project. All files associated with a design—schematics, netlists, reports, PLD source code, simulation results, and 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 **Schematic Design Tools** screen.

These functions are described briefly on the pages that follow. The explanations assume that you are already familiar with common electronic design terms and concepts. If you are just learning about schematic 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 2 through 8 of this guide. Advanced concepts are fully explained in the *Schematic Design Tools Reference Guide.*

You can run all OrCAD tools on a single worksheet or on a multiple-sheet design. Multiple-sheet designs can be either flat designs or hierarchical designs. To learn about these different types of files, see the *Design Structures* section later in this chapter.

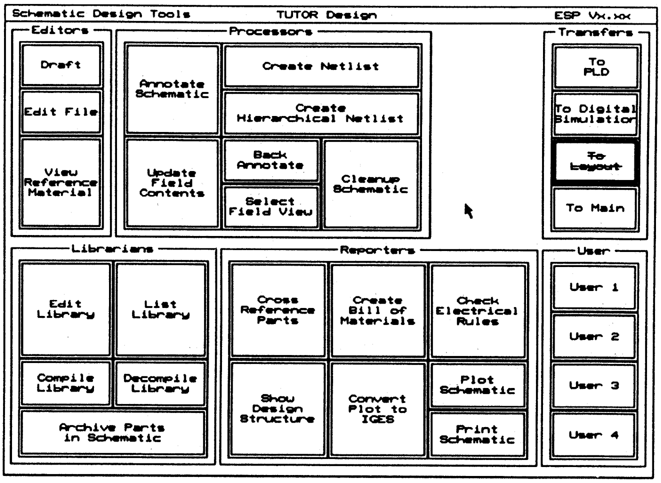


Figure 1-1. ***Schematic Design Tools*** *screen.*

Editors

Editors are used to create or modify design files. The three editor tools in **Schematic Design Tools** are:

- **Draft**. The heart of **Schematic Design Tools** is the schematic editor, **Draft. Draft** is used to create schematics, which are part of the design database.
- **Edit File**. This text editor is used to create and edit text files.
- **View Reference.** This tool runs a text editor in a reference material directory provided by OrCAD. This directory contains supplemental "read me" files about drivers, libraries, netlist formats, and other topics of interest.

Processors

Processors are tools that subject a design file to a specific process. The seven processor tools in **Schematic Design Tools** are:

- **Annotate Schematic**. This tool automatically updates part reference designators (such as U?, R?). It also updates the pin numbers associated with the reference designators in multiple-element parts. **Annotate Schematic** can handle very large, complex, multiple-sheet designs. It can update incrementally (leaving previously assigned reference designators alone) or unconditionally.

- **Create Netlist.** This tool creates a netlist that is a text file listing the logical interconnections between signals and pins. When the design becomes a real circuit board, the netlist turns into patterns of physical connections called tracks and nets. **Create Netlist** creates a netlist in one of over 30 different formats. Refer to *Appendix B: Netlist formats* in the *Schematic Design Tools Reference Guide* for a list of available formats.

  You can also create your own netlist formats. See *Appendix D: Creating a custom netlist format* in the *Schematic Design Tools Reference Guide* for instructions.

  Create Netlist also creates the connectivity database. The connectivity database is used when transferring to OrCAD's **Programmable Logic Design Tools** and **Digital Simulation Tools.**

- **Create Hierarchical Netlist.** This tool operates similarly to the **Create Netlist** tool, only it is used on hierarchical designs. Hierarchical designs are discussed later in this chapter.

- **Update Field Contents.** This tool updates part value and part fields. Every part has ten fields that are used to hold text or data associated with the part. One data field holds reference designator values, such as "U1A" or "Q1." Another holds the part's name, such as "74LS04" or values relevant for the part, such as ohm (Ω) values for resistors. The other eight data fields can store any information you might find useful: part tolerance, vendor name, part number, and so on.

  Update Field Contents can change information in all but the reference designator field. It changes fields based on the contents of an update file. You create the update file using **Edit File.**

- **Back Annotate.** This tool updates part reference designators by using a list of old and new reference designators called a Was/Is file. You create the Was/Is file using **Edit File.**

- **Cleanup Schematic.** This tool checks to see if any wires, buses, junctions, labels, module ports, or other objects have been placed on top of one another.

- **Select Field View**. This tool makes the contents of a data field either visible or invisible on the schematic.

Librarians

Schematic Design Tools includes libraries containing more than 20,000 parts. These parts represent TTL, IEEE, CMOS, memory, ECL, discrete, analog, microprocessor, and peripheral devices.

Librarians are tools for managing and creating library parts. The five librarian tools in **Schematic Design Tools** are:

- **List Library**. This tool lists all the parts in a library.
- **Archive Parts in Schematic**. This tool scans a single schematic, or an entire design and collects all the parts used. It then creates a library file containing those parts.
- **Edit Library**. This tool is a graphical editor for creating or modifying library parts. You can save an edited part in a new or existing library.

Parts can also be created or modified using a text editor, such as **Edit File**. If you prefer to create or modify parts in this manner, you will find the following tools very useful:

- **Compile Library**. This tool converts a library source file into a compiled library object file. The compiled library object file can be used by the other **Schematic Design Tools**.
- **Decompile Library**. The inverse of the **Compile Library** tool, this tool converts a compiled library object file into a library source file. You can edit the library source file using **Edit File**.

Reporters

The seven reporter tools in **Schematic Design Tools** are:

- **Cross Reference Parts.** This tool scans specified schematic files, gathers information for all the parts used in the schematic files, and creates a report that lists each part's location in the design.
- **Create Bill of Materials.** This tool creates a summary list, sorted by reference designator, of all the parts used. You can also merge additional information into the summary list by using an include file.
- **Check Electrical Rules.** This tool checks for conformity to basic electrical rules. It checks for shorts, inputs with no driving source, unconnected pins, bus contention, and other common electrical hookup problems.
- **Show Design Structure.** This tool scans a hierarchical design and displays the design's root worksheet filename and all the associated sheet names. The filename of each sheet is also listed.
- **Convert Plot to IGES.** This tool translates a schematic plot file (created by **Plot Schematic**) into IGES (Initial Graphics Exchange Specification) text format. This common data format allows schematic plot files to be stored on a mainframe computer or used with other applications that accept IGES input (such as VersaCAD®.)
- **Plot Schematic.** This tool plots a single schematic or all the schematics in a design. It produces high-resolution plots of your designs.

  Devices that accept *vector* commands are considered to be plotters. A vector is a series of points with a specifically defined function.

- **Print Schematic**. This tool prints a single schematic or all the schematics in a design. It produces draft-quality printouts of your designs.

  Devices that accept *raster* commands are considered to be printers. A raster is an array of dots.

△ *NOTE: Reporters do not modify design data in any way.*

Transfers

Three of the transfer tools perform the steps needed to prepare a design for use by another OrCAD tool set. The **To Main** transfer tool simply displays the main screen. The four transfer tools in **Schematic Design Tools** are:

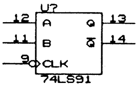
- **To PLD.** This tool updates part value and part fields, annotates the reference designators, extracts the **PLD** information, and displays the **Programmable Logic Tools** screen.
- **To Digital Simulation.** This tool annotates the reference designators, builds a connectivity database, creates trace and stimulus files, and displays the **Digital Simulation Tools** screen.
- **To Layout.** This tool updates part value and part fields, annotates the reference designators, builds a connectivity database, and displays the **PC Board Layout Tools** screen.
- **To Main.** This tool displays the main screen.

Graphic objects

Schematics are made up of a variety of graphic objects. You can include any of the following graphic objects in your schematic designs:

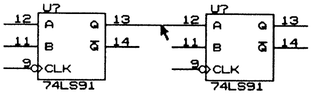
- Parts
- Wires
- Buses
- Junctions
- Power objects
- Module ports
- Sheet symbols
- Labels
- Text
- Title block
- Stimulus objects
- Vector objects
- Trace objects
- Layout objects

Parts



Parts are graphic objects you place on the schematic worksheet to represent the electronic parts in your design.

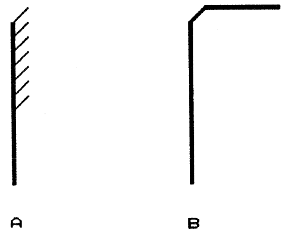
Wires



Wires are graphic objects you place on the worksheet to represent connections between objects, such as pins of parts and power objects.

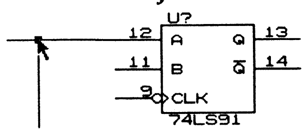
Buses

Buses are graphic objects that represent an array of signals as a single unit on your worksheet.



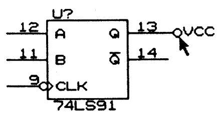
Junctions

Junctions are graphic objects that indicate a physical connection between wires, buses, and nodes. Junctions look like small square boxes.



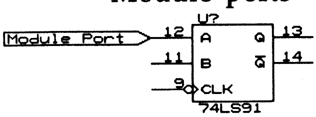
Power objects

Power objects are graphic objects that indicate a connection to a power source.



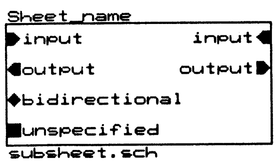
Module ports

Module ports are graphic objects that indicate where signals are conducted between worksheets.

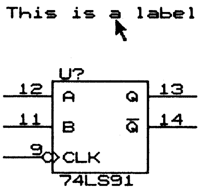


Sheet symbols

Sheet symbols are block-shaped symbols representing other worksheets. Each sheet symbol represents a subsheet. Sheet symbols are only used in hierarchical designs.

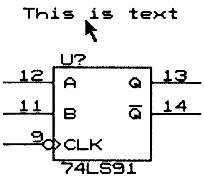


Labels



Labels identify the locations of signal connections without actually showing the physical connections on the worksheet.

Text



You can also place *text* in your worksheet. Text is used to leave notes or descriptive text (that isn't required by the circuit) on a schematic diagram. Such text helps you and others understand the functions being performed or documents some aspect of circuit operation.

Title block

The *title block* is used to label your worksheets so that you can tell them apart. It contains information such as company name and address, schematic title, number, size, and revision.

Stimulus objects

OrCAD's **Digital Simulation Tools** uses *stimulus objects* to determine where a stimulus is to be applied to a circuit.

Vector objects

OrCAD's **Digital Simulation Tools** uses *vector objects* to determine where sets of stimuli are to be applied to a circuit.

Trace objects

OrCAD's **Digital Simulation Tools** uses *trace objects* to determine which signals to trace.

Layout objects

OrCAD's **PC Board Layout Tools** uses *layout objects* to get information about particular signals such as track width, via size, routing layer, and so forth.

The design process

As its name suggests, **Draft** is designed to be analogous to the schematic design tools with which you are already familiar: drafting board, pencil, sheets of paper, standard logic symbols and symbol templates, and so on.

In addition, **Draft** is designed to support the complete design process from the general concept of a design to the final sets of detailed schematic diagrams.

The computer screen represents the drafting table. The pointer does what a pencil does, and more. Drawing and erasing are done using **Draft** commands.

Draft calls the sheets of drafting paper on which the schematics are drawn *worksheets*. Worksheets appear on the computer screen as a rectangular area in which you can place parts and wires.

When you save the work you have done on a worksheet, **Draft** stores the information on the computer's disk as a data file. The name of the worksheet is the name of the file in which it is saved. Worksheets are stored inside designs. A design is a directory that contains all of the files (including the worksheet) that are part of the design process. All designs are contained in the \ORCAD directory.

Draft saves the worksheet in the design in which you are working. The worksheet can have the design name and an extension of .SCH, or you may give it different name.

For example, if you have a design called TUTOR, the pathname for the TUTOR schematic is \ORCAD\TUTOR\TUTOR.SCH.

Design structures

Some designs are small enough to be represented entirely on a single schematic worksheet. **Draft's** standard page sizes correspond to the five standard sheet sizes for plotters and printers (A through E for English, and A4 through A0 for metric). You can also create custom page sizes up to 65 inches by 65 inches.

However, some designs may be too large for even the biggest sheet, and even if a very complex design could fit on one sheet, there are good reasons for dividing it up:

- To partition a design so that several people can work on it at once.
- To develop the design using a top-down approach. That is, you may want to begin with a block diagram in which each block represents a major function, and then construct more detailed diagrams for each of the blocks.
- To organize your design by functional parts.
- To maximize the performance of your tools.

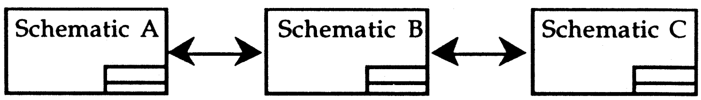
Draft offers two ways of handling multiple-sheet designs:

- Flat designs
- Hierarchical designs

Flat designs

Best suited for small designs with no more than five to ten sheets, flat designs laterally connect the output signals from one schematic to the input signals of another.

All schematics in a flat design are of equal importance, as shown below.



Since you must manage all of the interconnections between the sheets of a flat design by the names assigned to the module ports, it is best to keep a flat design relatively small.

Module ports

Module ports that have identical names on both schematics are considered to be electrically connected.

Figure 1-2 shows an example of connections between schematics in a simple two-sheet flat design.

The module ports in figure 1-2 that connect between the schematics are named **COUNT, CLEAR, LOAD,** and **RCO**. The module ports named **Hi[0..3]** and **Lo[0..3]** don't connect to each other.

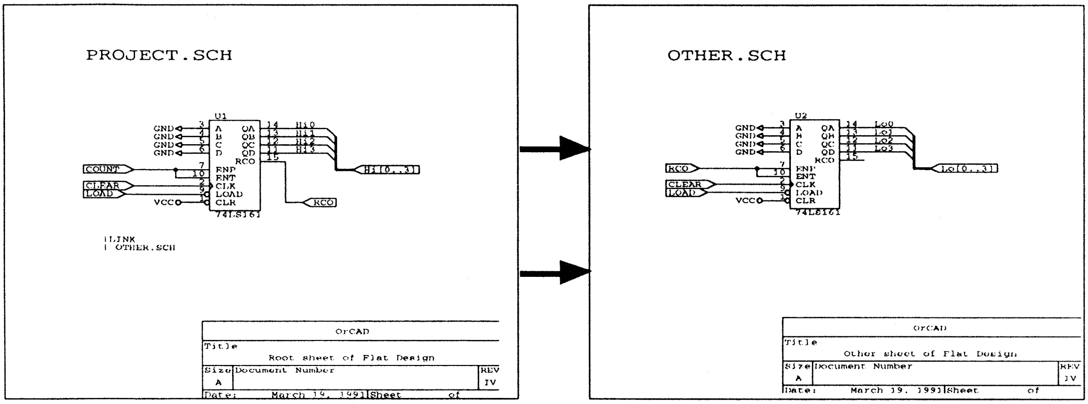
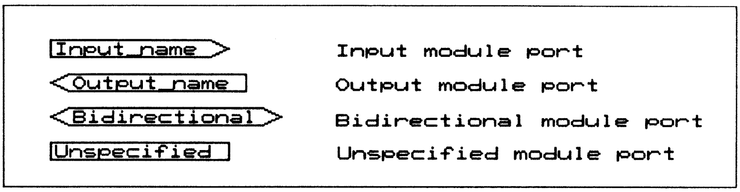


Figure 1-2. Module ports used to link one schematic to another.

Figure 1-2 shows only input and output module ports, and connections between single wires. **Draft** has two other types of module ports: bidirectional and unspecified. You can use module ports to connect buses, as well as single wires. There are four types of module ports:



You place module ports on a schematic using **Draft's PLACE Module Port** commands.

|LINK command

Module ports indicate the names of the signals to be connected but do not specify which schematics are to be included in the design. Therefore, flat designs must have one other component: a list of the worksheets in the schematic.

This list appears on the root schematic and consists of the "pipe" character (the vertical bar on your keyboard) followed by the keyword "LINK," followed by subsequent lines consisting of the pipe character and the filenames of the worksheets to be linked to the root schematic. The root schematic of the design usually has the name of the design and a .SCH extension.

You place this list on the root schematic using **Draft**'s **PLACE Text** commands.

The example at right shows text as it would appear on a root schematic that has module ports that link to schematics called SCHEM1.SCH, SCHEM2.SCH, and SCHEM3.SCH. This text can appear anywhere on the worksheet.

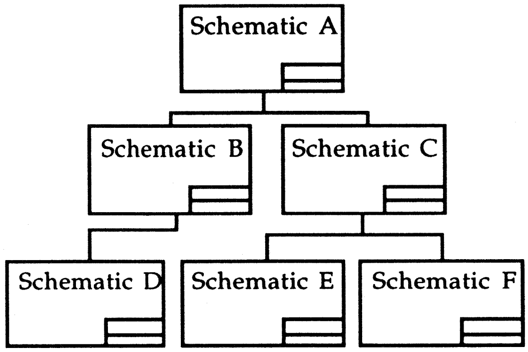
```
|LINK
|SCHEM1.SCH
|SCHEM2.SCH
|SCHEM3.SCH
```

Notice the |LINK command on the PROJECT.SCH worksheet in figure 1-2.

△ *NOTE: For details about module ports, see the* ***PLACE Module Port*** *command in the* Schematic Design Tools Reference Guide. *For details about placing text on a worksheet, see the* ***PLACE Text*** *command in the* Schematic Design Tools Reference Guide.

Hierarchical designs

Instead of using a flat design, you can create schematics that contain symbols representing other schematics. These symbols are called *sheet symbols*. The layered arrangement created by placing schematics inside other schematics is called a *hierarchy*. Any hierarchy—whether it is a corporate organizational chart or a schematic design—has "higher" and "lower" levels.



Any schematic can contain sheet symbols that reference other schematics, and this nesting structure can be made many levels deep. The schematic at the top of a hierarchy, which directly or indirectly references all other schematics in the design, is called the *root worksheet*.

You place sheet symbols on a schematic using **Draft's PLACE Sheet** commands.

How signals enter and leave sheet symbols

Just as module ports indicate where signals connect between schematics, *sheet nets* indicate where signals connect between a sheet symbol and its associated schematic. In figure 1-3, sheet nets are the small black objects shown on the borders of the sheet symbols. The sheet nets on a sheet symbol correspond to module ports on the schematic named in the sheet symbol.

You place sheet nets using **Draft's Add-NET** command, which becomes available when you select the **PLACE Sheet** command. To associate a particular sheet net with a particular module port, assign them the same name.

The bracketed notation shown on the module ports and sheet nets [m..n] designates the number of signals carried by a bus. So [0..3] indicates four signals, 0 through 3.

In real designs, buses must have a label or a module port with similar bracket notation to indicate the number of signals they carry, and wires connected to buses must have labels identifying the signal they carry. These details are shown in figure 1-3.

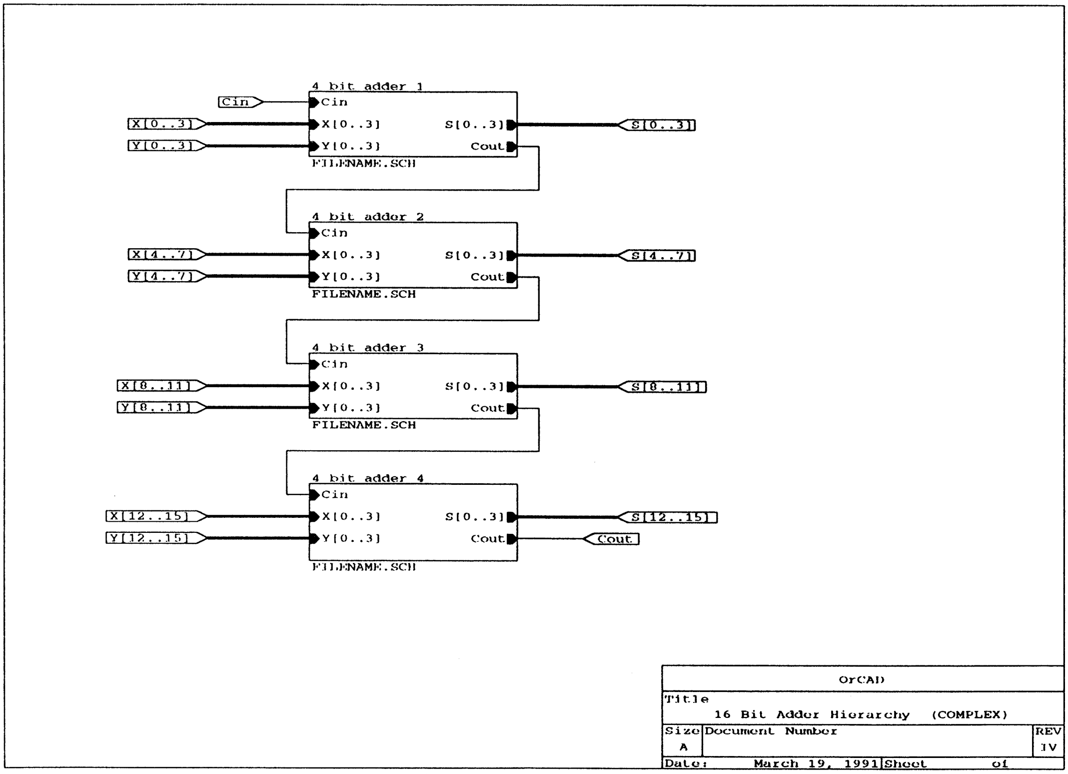


Figure 1-3. Simple hierarchical structure.

Hierarchies can access the same logic repetitively

The diagram shown in figure 1-3 shows a one-to-one correspondence between sheet symbols and the schematic diagrams they reference. This structure is called a *simple*

Schematic Design Tools can also reference a single schematic from more than one sheet symbol. To do this, you mark all the sheet symbols with the filename of the schematic to reference. This structure is called a *complex* hierarchy.

How sheet symbols reference schematic logic

To get a sheet symbol to access the logic of a particular schematic, you mark the sheet symbol with that schematic's filename. The filename displays at the bottom of the sheet symbol, as shown in figure 1-3.

You mark the sheet symbols using the **Filename** command, which becomes available when you select the **PLACE Sheet** command.

In addition to their filename markers, sheet symbols can also have names. In figure 1-3, the sheet symbol names are shown just above each sheet symbol.

You name the sheet symbols using the **Name** command, which becomes available when you select the **PLACE Sheet** command.

Moving between levels in a hierarchy

Draft makes it easy to move up and down in a hierarchy, from sheet symbol to associated schematic and back again.

To go from a sheet symbol to its associated schematic, put the pointer on the sheet symbol and select **QUIT Enter Sheet**. To go from the schematic back to the schematic in which it is referenced by a sheet symbol, select the **QUIT Leave Sheet** command.

More about hierarchical design structures

The schematic represented by a sheet symbol can itself have a sheet symbol within it. This means you can create hierarchies that are many levels deep can be created, each level containing progressively more detail.

This is particularly useful for very complex designs. It encourages a logical, function-oriented approach to partitioning designs, and makes them easier to manage.

Another advantage offered by hierarchical structure is the ability to use sheet symbols to repeatedly reference "stock" schematics containing common circuit functions. This is often used in gate array and FPGA designs.

△ *NOTE: Designing a deep hierarchy is much more efficient than designing a wide hierarchy. A wide hierarchy, while not a flat design, has many of the limitations in organization, presentation, and structure that flat designs have. A deep hierarchy lets the functional nature of the design be represented more clearly.*

For more information, study the hierarchy examples in *Chapter 8: Structuring your design.*

Learning Schematic Design Tools

The remainder of the *Schematic Design Tools User's Guide* shows how to design schematics by guiding you through the process of creating the schematic diagrams for a digital clock. To do this, you use the schematic editor called **Draft** to create the schematic of the clock circuitry. Within the schematic are three smaller circuits:

- A clock oscillator circuit
- A power regulator circuit
- A logic and display circuit

Each chapter builds on the skills and concepts from the previous chapter. As you complete each chapter, you create a series of working files.

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

Chapter 2: Introducing Schematic Design Tools

This chapter introduces **Draft**, the **Schematic Design Tools** schematic editor. You learn how to run **ESP**, run **Design Management Tools**, set up work conditions for **Draft**, run **Draft**, capture and save an initial macro, and save your work.

Chapter 3: Capturing the clock oscillator schematic

In this chapter you create (or *capture*) a small schematic and learn the basic procedures required for schematic capture. You learn how to get and place library parts, how to place wires, how to place junctions, and how to place labels and text.

Chapter 4: Capturing the power regulator schematic

In this chapter you capture a schematic that is slightly more complex than the previous schematic. You learn how to move a group of parts, delete a part, undo a delete operation, rotate a part, place a power symbol, set a tag, jump to a tag or a reference, and print a hard copy of the schematic.

Chapter 5: Creating a custom part

In this chapter you use **Edit Library** to define a custom part (a seven-segment display). You learn how to draw the part body, draw special shapes, use shading, add pins to the part body, add pin names, and save the new part in a library.

Chapter 6: Capturing the logic and display circuit schematic

In this chapter you capture the final portion of the digital clock schematic. You learn how to draw a repeatable portion of the schematic, make and place multiple copies of it, write and use a macro, and use repeat parameters to place wires and labels.

Chapter 7: Using other Schematic Design Tools

This chapter introduces you to some of the other tools included in **Schematic Design Tools**. You learn to use **Annotate Schematic, Check Electrical Rules , Create Netlist, Back Annotate, Create Bill of Materials,** and **Plot Schematic.**

Chapter 8: Structuring your design

This chapter describes and reviews a complex hierarchy and shows how to convert a complex hierarchy to a simple hierarchy. Flat designs and how sheets are linked together are also reviewed.

Chapter 9: Tips and techniques

This chapter provides a collection of tips and techniques that you can use to enhance your ability to use **Schematic Design Tools** productively. This chapter does not follow the tutorial style of the other chapters.

Introducing Draft

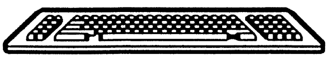
In this chapter, you establish **Draft's** work conditions. You learn how to:

- Run **ESP**
- Run **Design Management Tools**
- Set up work conditions for **Draft**, the schematic editor
- Run **Draft**
- Capture and save an initial macro
- Save your work

Before you begin

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

Keys



Schematic Design Tools is designed to operate on a wide variety of 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 Enter, New Line, Next, Return, or Send.

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, special, and combinational 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

Text that you enter is shown in two ways:

- As bold text in typewriter font. For example, "enter **`tutor.sch`**."
- As bold text in typewriter font enclosed in a box. For example,

```
tutor.sch
```

or

```
load file? tutor.sch
```

In the examples above, you only enter the characters shown in bold. The non-bold characters show what displays on the screen.

Operating system command prompt

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

```
C:>
```

Callouts

In the later chapters of this guide, callouts—such as: ①②③—appear on schematic diagrams. These callouts refer to the corresponding step numbers in the instructions.

Filenames

Filenames can be from one to eight characters long. A filename may also have an extension consisting of a period and one 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, only use letters (A-Z and a-z), numbers (0-9), the underscore (_), pound sign (#), and at sign (@).

Most OrCAD software works with any characters your operating system supports. However, when some applications are used in conjunction with OrCAD software, a more limited character set is supported. These applications include SPICE programs, some PCB layout programs, and some text editors.

Designs

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

By following the steps in this tutorial, you will work on a design named "TUTOR." All of the files for this design are contained in the directory named "TUTOR." Files in the directory have the filename "TUTOR" and an extension that indicates the type of file. For example, the TUTOR schematic worksheet that you create in chapters 2 through 6 is named TUTOR.SCH.

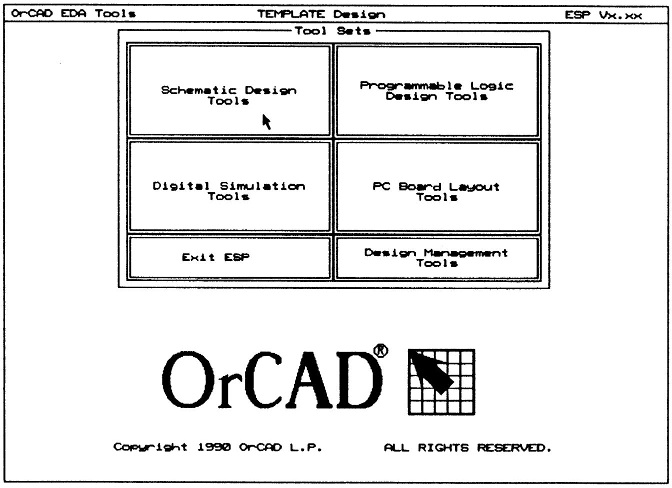
Running ESP

To run an OrCAD tool, you must first display the main screen. Follow these steps:

1. Be sure that your computer is turned on.
2. At the operating system prompt, enter the command shown in bold:

```
C:> orcad
```

In a moment, the main screen displays:



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.

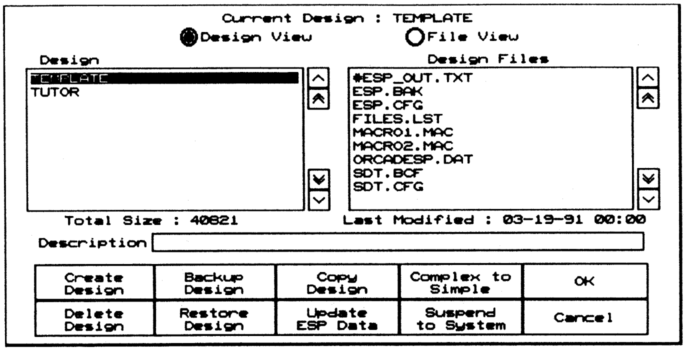
Run Design Management Tools

Follow these steps to run **Design Management Tools** and change to the TUTOR design:

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

```
Design Management Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```

2. Click the left mouse button to select the **Execute** command. The screen shown below displays.



Design Management Tools screen.

3. Place the pointer on the design named TUTOR and click the left mouse button. This selects the TUTOR design.

4. Click **OK** to return to the main screen. Notice that the heading in the upper center of the screen has changed to **TUTOR Design**.

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

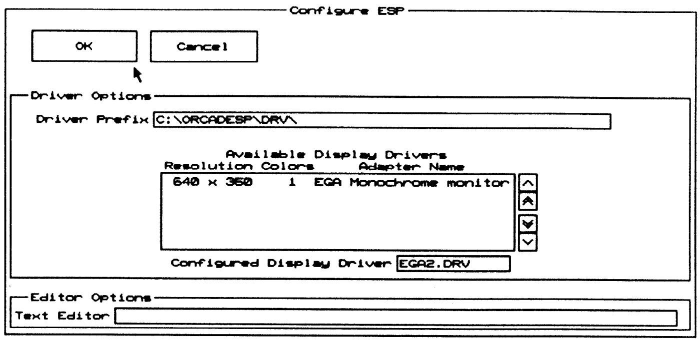
Change the start-up design

The design environment is configured to start in the TEMPLATE design each time you run the design environment. Since you will be working in the TUTOR design throughout this guide, you need to change the start-up design to TUTOR. Follow these steps:

1. Click the **Design Management Tools** button. The menu at right displays.

```
Design Management Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```

2. Select **Configure ESP**. The screen below displays:



*Top portion of the **Configure ESP** screen.*

3. Move the pointer to the bottom of the screen. The display *pans* to show more of the **Configure ESP** screen. Continue panning until you reach the **Design Options** area.

```
Design Options
Startup Design   TEMPLATE
```

4. Place the pointer in the **Startup Design** entry box and click the left mouse button. The pointer becomes a cursor in the entry box. Use the <Backspace> key to delete `TEMPLATE`. Enter **TUTOR** as the start-up design.

```
Design Options
Startup Design   TUTOR
```

5. Move the pointer to the top of the screen and click the **OK** button. (An easy way to move to the **OK** button is to press the <Home> key.)

   The changes you made to the **Configure ESP** screen are saved and the main screen displays.

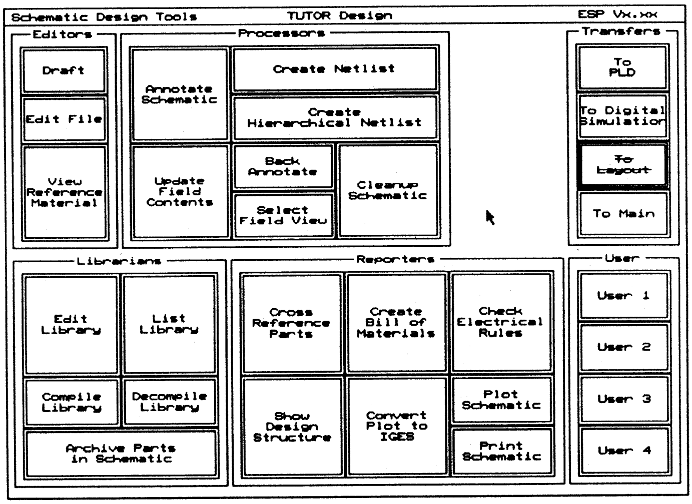
△ *NOTE: Refer to the* OrCAD/ESP Design Environment User's Guide *for detailed instructions on how to configure ESP.*

Running Schematic Design Tools

Follow these steps to display the **Schematic Design Tools** screen.

1. Click the **Schematic Design Tools** button. The menu at right displays.
2. Select the **Execute** command. The **Schematic Design Tools** screen displays:

```
Schematic Design Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```



Schematic Design Tools screen.

Defining title block information

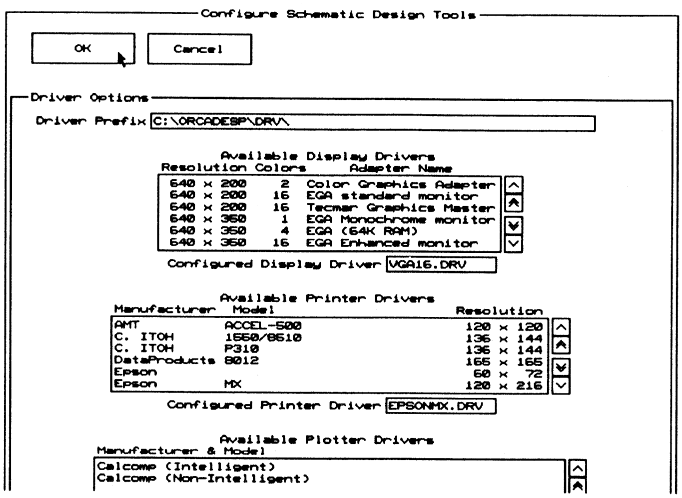
Before you run the schematic editor **Draft**, take a few minutes to configure the information to appear in the title block of the worksheet you will create. To do this, you must display the **Configure Schematic Design Tools** screen.

View the configuration for Schematic Design Tools

1. Click the **Draft** button. The menu at right displays.
2. Select the **Configure Schematic Tools** command.

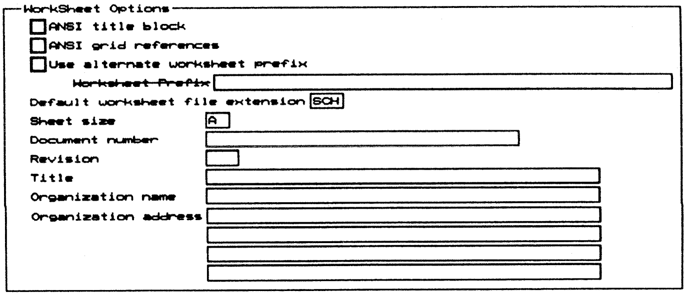
```
Draft
Execute
Local Configuration
Assign Hot Key
Show Version
Configure Schematic Tools
Help
```

The following figure shows the top portion of the **Configure Schematic Design Tools** screen. The parameters you see may vary, because some of the configuration information depends on your system hardware. For more information about the **Configure Schematic Design Tools** screen, see *Chapter 1: Configure Schematic Tools* in the *Schematic Design Tools Reference Guide.*



*Top portion of the **Configure Schematic Design Tools** screen.*

3. Pan to the **Worksheet Options** area (shown below).



Worksheet Options area of the Configure Schematic Design Tools screen.

Notice the **Document number, Revision, Title, Organization name,** and **Organization address** entry boxes. Any information entered in these fields becomes a part of your worksheet's title block. For this tutorial, you enter information in the **Title, Organization name,** and **Organization address** entry boxes.

4. Position the pointer within the **Title** entry box and press <Enter>. Enter the title: `Digital clock schematic.`

5. Press <Tab> to move to the next entry box, in this case, **Organization name,** and press <Enter>. Enter the name of your organization.

6. Press <Tab> to move to the first entry box of **Organization address,** and press <Enter>. Enter the street address of your organization.

7. Press <Tab> to move to the second entry box of **Organization address,** and press <Enter>. Enter the city and state of your organization.

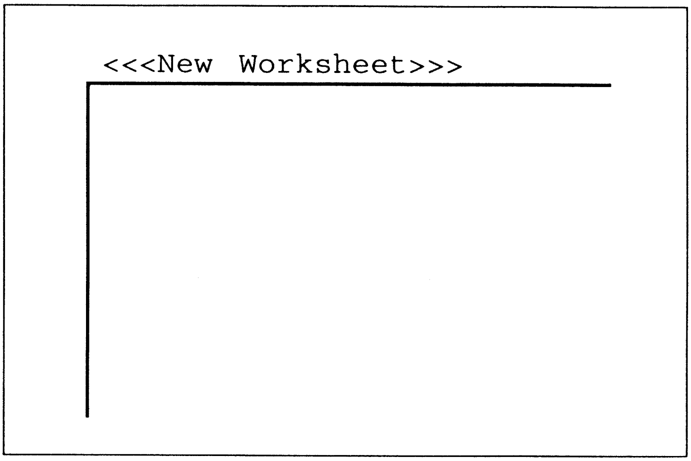
8. Move the pointer to the top of the screen and click the **OK** button. This updates the configuration and displays the **Schematic Design Tools** screen.

Running Draft

Now that you have changed the start-up design to TUTOR and set up your title block information, you are ready to begin learning about the schematic editor **Draft**. Follow these steps:

1. Click the **Draft** button. The **Draft** menu displays.
2. Select the **Execute** command.

   Draft is now running. The top and left edges of the sheet display. Because the screen is smaller than the worksheet, the right and bottom edges of the worksheet are not visible. You can think of the screen as a window into the larger worksheet area.



*New worksheet in **Draft**.*

Learning OrCAD basics

Pop-up menus guide you step-by-step through OrCAD software. **Draft** organizes commands using menus and command lines. You can select a command by either clicking the mouse or pressing a key. (For complete command descriptions, refer to the *Schematic Design Tools Reference Guide*.)

Main menu

Follow these steps to display and remove the main menu (shown at right).

1. Press <Enter> to see the main menu.
2. Press <Esc> to remove the main menu from the screen.
3. Click the left mouse button to see the main menu.
4. Click the right mouse button to remove the main menu from the screen.

```
Again
Block
Conditions
Delete
Edit
Find
Get
Hardcopy
Inquire
Jump
Library
Macro
Place
Quit
Repeat
Set
Tag
Zoom
```

To return to the main menu—no matter where you are in **Draft**—press <Esc> or the right mouse button and then press <Enter> or the left mouse button.

Commands

There are several ways to select and use a command. You can use the methods shown in table 2-1 in any combination. The method you use is a matter of personal preference.

	Using the keyboard	*Using the mouse*
To select a command	Use the arrow keys on the keyboard to place the highlight over the command.	With the mouse, slide the highlight over the command.
To use a command	Press <Enter>.	Click the left mouse button.
To select and use a command	Press the first capitalized letter in the command.	

Table 2-1. Using the keyboard or mouse to select a command.

Draft responds to a command by either performing the command's function or by displaying another menu or a command line.

Menus

All menus look and work just like the main menu. **Draft** displays the menu name on the top line of the screen. Press <Esc> or 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 the **BLOCK** command. The menu at right displays.
3. Press <Esc> to return to the main menu.

```
Block
Move
Drag
Fixup
Get
Save
Import
Export
ASCII Import
Text Import
```

Command lines

Command lines are a series of commands listed across the top of the screen. When a command line displays, you can move the pointer around the working area or select a command. Press <Esc> or the right mouse button to return to the menu or command line that called the command line. Follow these steps to familiarize yourself with these processes:

1. Press <Enter> to display the main menu.
2. Select the **EDIT** command. Notice a command line displays across the top of the screen. The **EDIT** command line is shown below.

```
Edit  Find  Jump  Zoom
```

3. Press <Esc> and then press <Enter> to return to the main menu.

Returning to the main menu

To return to the main menu—no matter where **Draft** is—press <Esc> as many times as necessary until no menu or command line displays in the upper left corner of the screen. At this point, the main menu displays if you press <Enter>.

How commands are shown in this guide

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

For example, the statement, "Select the **PLACE Wire** command" means, "Select the **PLACE** command from the main menu, and then from the resulting **PLACE** menu, select the **Wire** command."

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

Setting up Draft's work conditions

Now that you understand how **Draft**'s commands, menus, and command lines operate, you will use the **SET** command to change the default work conditions that govern the way **Draft** displays and maintains schematics.

Display work conditions settings

Follow these steps to display the **SET** menu:

1. Press <Enter> to see the main menu.
2. Select **SET** from the main menu. The **SET** menu displays, as shown below.

```
Set
Auto Pan             YES
Backup File          YES
Drag Buses            NO
Error Bell           YES
Left Button           NO
Macro Prompts        YES
Orthogonal           YES
Show Pins            YES
Title Block          YES
Worksheet Size         A
X,Y Display           NO
Grid Parameters
Repeat Parameters
Visible Lettering
```

Using the commands in the **SET** menu, you can control features such as automatic backup of schematic files, the angles at which you can place wires, and whether or not pin numbers display on part symbols. For more information about **Draft**'s work conditions, see the **SET** command description in the *Schematic Design Tools Reference Guide.*

The next few sections describe a few of **Draft**'s work conditions and the commands controlling them.

Pan across the schematic

Auto Pan is the first command in the **SET** menu. When you start work on a new worksheet, **Auto Pan** is set to **Yes.**

When **Auto Pan** is set to **Yes**, the worksheet follows the movement of the pointer. If part of a worksheet is off the screen and you move the pointer beyond the edge of the display, the hidden part of the worksheet pans into view.

If you set **Auto Pan** to **No**, the screen does not pan. In this case, you must use the **JUMP** and **ZOOM** commands to see different parts of the worksheet.

Follow these steps to pan across the schematic:

1. Press <Esc> to remove the **SET** menu from the screen. **Auto Pan** remains set to **Yes.**

2. Move the pointer to the lower right corner until the title block displays. The screen pans to keep up with the pointer. Notice that the title block information you entered earlier in this chapter displays.
3. Move the pointer toward the upper left-hand corner until the upper left corner of the worksheet displays.

Redisplay the SET menu

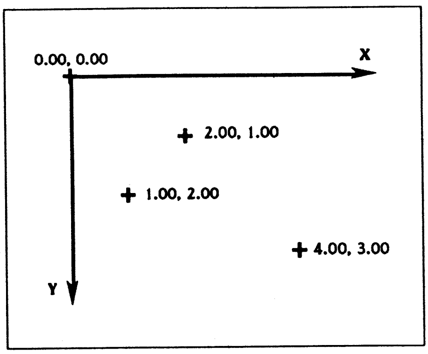
Follow these steps to display the **SET** menu again:

1. Press <Enter> to recall the main menu. When the menu displays, you'll see the highlight bar is on the **AGAIN** command.
2. Press <Enter> to select **AGAIN**. This selects the main menu command you chose last—in this case, **SET**.

Display X,Y coordinates

Draft uses a coordinate system to locate points on the worksheet, as shown in the illustration below.

An X coordinate specifies horizontal location and a Y coordinate specifies vertical location. Thus any point on the worksheet can be indicated by an X and Y coordinate pair in the form (X,Y). The (0.00, 0.00) point is always at the upper left of the worksheet.



If **X,Y Display** is set to **Yes**, the X and Y coordinates of the pointer's position display in the upper right corner of the screen. The default setting is **No**.

Follow these steps to display X, Y coordinates:

1. Select **X,Y Display**. "Display X,Y Coordinates of Pointer" and a short menu display.
2. Select **Yes**. The prompt and the menu disappear.
3. Move the pointer in any direction and watch the X,Y coordinates in the upper right-hand corner of the screen.

The units shown in the X,Y display represent inches on the printed schematic. The upper left corner is (.00, .00) and the lower right corner is (9.50, 7.00). On a sheet 8.5 inches by 11 inches, the actual drawing area is 7 inches by 9.5 inches. This allows for borders around the drawings.

Select worksheet size

The **Worksheet Size** command selects one of five sizes for your schematic. Follow these steps to change the worksheet size:

1. Press <Enter> to display the main menu, then select **AGAIN**. The **SET** menu displays.
2. Select **Worksheet Size.** A menu lists the five options available for the size of a worksheet, as shown at right.
3. Select **C** size.
4. Move the pointer to the edges and corners of the worksheet to explore the size of the editable region of a C-size sheet. The dimensions shown in the **Worksheet Size** menu are the worksheet's borders. On a C-size sheet 22 inches by 17 inches, the actual drawing area is 20 inches by 15 inches.

```
Set Worksheet Size
(Area inside borders)
A    9.50 x  7.00
B   15.00 x  9.50
C   20.00 x 15.00
D   32.00 x 20.00
E   42.00 x 32.00
```

*NOTE: If **Schematic Design Tools** is configured to use metric dimensions, the **Set Worksheet size** menu displays the International Standards Organization paper sizes: A4 through A0. In addition, the X, Y display is given in millimeters. For information about configuring **Schematic Design Tools** to use metric dimensions, refer to* Chapter 1: Configure Schematic Tools *in the* Schematic Design Tools Reference Guide.

Changing your view of the worksheet

Draft can display worksheets at five different scales. You change the view using the **ZOOM** command. The worksheet can be zoomed in or out to magnify or reduce its visible image.

When **Draft** is zoomed out, you can see a large portion of the worksheet. Zooming in enlarges a small portion of the worksheet and displays more details. You can zoom in to draw intricate portions of your worksheet with exacting detail and then zoom out to look at the whole schematic.

ZOOM in and out

Follow these steps to zoom out and see more of the worksheet on the screen at one time:

1. Move the pointer to lower right corner until the title block displays.
2. Select **ZOOM** from the main menu. The menu at right displays.

```
Zoom (present
scale=1)
Center      ( 1)
In          ( 1)
Out         ( 2)
Select
```

3. Select **Out**. A view of the worksheet at one-half the original scale displays.
4. Experiment with the scale using **In, Out**, and **Select**. If you use **Select** you can choose the scale at which to view the worksheet, as shown in the following figure.

   ```
   Zoom - Select Scale
   (present scale=1)
    1
    2
    5
    10
    20
   ```

   If you choose **1**, you view the worksheet at full size. This shows the most detail ("zooms in" the farthest). If you choose **2**, you view the worksheet at one-half the original scale. If you choose **20**, you view the worksheet at one-twentieth the original scale—you see the maximum working area and the least detail.
5. When you are done experimenting with zooming, return to full size view (scale level 1).

Set grid parameters

While working on a large worksheet, it is useful to have visual cues that tell you approximately where you are on the sheet.

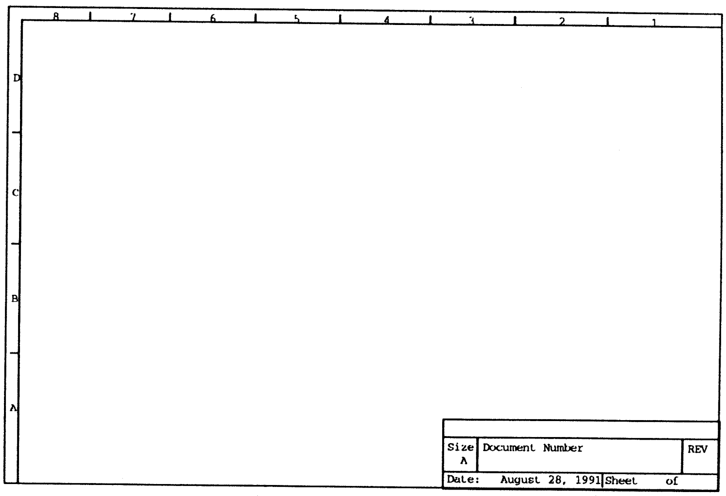
The **Grid Parameters** command on the **SET** menu sets up some of these visual cues. The **Set Grid Parameters** menu is shown at right.

Set Grid Parameters

Grid References	NO
Stay On Grid	YES
Visible Grid Dots	NO

Display grid references

Grid References turns grid reference guides along the top and left edges of the display on and off. The guides divide the worksheet into blocks. Horizontally, the guides divide the worksheet from 8 to 1. Vertically, they divide the worksheet from D to A. For example, the title block (lower right corner) is located at A-1, as illustrated in the following figure.



Grid references.

Use the **JUMP Reference** command on the main menu to move to specific locations using these map-like coordinates.

Follow these steps to display grid references:

1. Select **SET** from the main menu.
2. Select **Grid Parameters.**
3. Select **Grid References.**
4. Select **Yes**. The grid reference guides display at the top and left edges of the screen.

△ *NOTE:* ***Schematic Design Tools*** *can be set up to use ANSI Y14.1 drawing standards. Refer to the* Schematic Design Tools Reference Guide *for details.*

Stay on grid

Stay on Grid determines whether or not pointer movement is restricted to grid intersections. **Stay On Grid** is set to **YES**. Do not make any changes here.

△ *NOTE: Keep* ***Stay on Grid*** *set to* ***Yes*** *unless you have a compelling reason to be off-grid. Anything placed off-grid—such as text and labels—may be hard to select and edit later.*

Make the grid visible

Visible Grid Dots turns the dots representing intersections on and off. The space between the dots represents 0.1 inch on the printed worksheet. Follow these steps to make the grid dots visible:

1. Select **SET** and **Grid Parameters** again.
2. Select **Visible Grid Dots,** then select **Yes**. Grid dots display on the worksheet. You may want to adjust the intensity on your monitor to make the grid dots brighter or dimmer.

Updating the worksheet

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

Update the file

Follow these steps to save the worksheet without changing its filename:

1. Select **QUIT** from the main menu. **Draft** displays the filename and the **Quit** menu, as shown at right.
2. Select **Update File. Draft** saves the file.
3. Press <Esc> to dismiss the **Quit** menu.

```
Quit TUTOR.SCH
Enter Sheet
Leave Sheet
Update File
Write to File
Initialize
Suspend to System
Abandon Edits
Run User Commands
```

Creating a macro

Macros can record virtually anything you do in a program—so you can automate many repetitive tasks and speed up your work.

Capture a macro

Earlier in this chapter, you used the **SET** command to change work conditions. Follow the steps below to capture commands for setting work conditions in a macro.

NOTE:This macro only works when you are at the main menu level of ***Schematic Design Tools.***

```
Macro
Capture
Delete
Initialize
List
Read
Write
```

1. Select **MACRO** from the main menu. The **MACRO** menu at right displays.
2. Select **Capture.** The prompt "Capture macro?" displays.

   Macros can be run by a single key or a combination of keys. Single keys that can run macros are the function keys (<F1> – <F10>) and special keys in the numeric keypad (such as <Home>, <Page Up>, and <Page Down>).

   Key combinations that can run macros include:

   - <Ctrl> + function keys
   - <Ctrl> + alpha keys (except C, H, and M)
   - <Alt> + function keys
   - <Alt> + alpha keys
   - <Shift> + function keys

   If you choose a prohibited key combination, **Draft** displays "Key cannot be defined as macro" and displays the "Capture macro?" prompt again.

3. Press <Ctrl><A> to assign a keystroke to this macro. **^A** displays at the "Capture macro?" prompt.
4. Press <Enter>. The message "<macro>" displays to remind you that you are defining a macro. Any commands you select while "<macro>" displays are added to the list of commands stored in the macro.

5. Type the commands in the left column below. **Schematic Design Tools** will perform each command, but the typed commands will not display.

<Enter>	*Displays the main menu*
SXY	Captures the SET X,Y Display YES commands
SGGY	Captures the SET Grid Parameters Grid References YES commands
SGVY	Captures the SET Grid Parameters Visible Grid Dots YES commands
ZS1	Captures the ZOOM Select 1 commands

6. Press <Ctrl><End> to end the macro definition. **Draft** confirms the macro definition is complete by displaying:

```
<<<MACRO END>>>
```

The macro is now stored in the computer's memory. You can run it when you are at the main menu level of **Schematic Design Tools** by pressing the key combination you specified, <Ctrl><A>.

△ *NOTE: Some keyboards have two keys labeled <End>. If you press <Ctrl><End> and <<<MACRO END>>> fails to display, repeat the entry using the <End> on the numeric keypad.*

Save the macro

Follow these steps to save the macro in a file:

1. Select **MACRO Write. Draft** displays:

```
Write all macros to?
```

2. Enter **`tutor.mac`**.

   Draft writes the macro to the TUTOR.MAC file in the TUTOR design directory.

3. To tell **Draft** to read macros from the TUTOR.MAC file, select **MACRO Read. Draft** displays:

```
Read all macros from?
```

4. Enter **`tutor.mac`**.

5. If you wish to test the macro you just saved, change some of the work conditions and press <Ctrl><A> to restore the work conditions.

Exiting Draft

You are nearly done with this chapter. Follow these steps to exit **Draft**:

1. Select **QUIT** from the main menu. **Draft** displays the filename and the **QUIT** menu shown at right.

2. Select **Update File. Draft** saves the file.

3. Leave **Draft** by selecting **Abandon Edits. Draft** exits to the **Schematic Design Tools** screen.

```
Quit TUTOR.SCH
Enter Sheet
Leave Sheet
Update File
Write to File
Initialize
Suspend to System
Abandon Edits
Run User Commands
```

Setting up automatically

In addition to using **SET** to control **Draft's** work conditions, you can automate the process of defining **Draft** work conditions by configuring **Schematic Design Tools** so that the macro you just created runs every time you run **Draft.** A macro that runs when the tool starts is called an *initial macro.*

View the configuration

Follow these steps to display the **Schematic Design Tools** configuration screen:

1. Select **Draft** from the **Schematic Design Tools** screen.
2. Select **Configure Schematic Tools** from the menu that displays. The **Configure Schematic Design Tools** screen displays.
3. Pan to the **Macro Options** area of the **Configure Schematic Design Tools** screen.
4. Position the pointer within the **Draft Macro File** entry box and press <Enter>. This entry box defines the name of the macro file you created earlier in this chapter. The pointer becomes a cursor inside the entry box.
5. Enter the macro path and filename:

   ```
   Draft Macro File \X(\orcad\tutor\tutor.mac
   )
   ```

   Notice that the **Draft Initial Macro** entry box became highlighted once you made an entry in the **Draft Macro File** entry box.
6. Position the pointer within the **Draft Initial Macro** entry box and press <Enter>. This entry box defines an **Initial Macro** that automatically runs when you run **Draft.**
7. Enter the keystrokes used to run the initial macro: <Ctrl><A>. However, instead of pressing the <Ctrl> key, simultaneously press <Shift> and <6> to enter the "caret" symbol and press <A>. The caret symbol (^) represents the <Ctrl> key.

   ```
   Draft Initial Macro ^A
   ```

8. Move the pointer to the top of the screen and click **OK**. This updates the configuration and displays the **Schematic Design Tools** screen.

NOTE: Once you configure ^A in the ***Draft Initial Macro*** *entry box on the* ***Configure Schematic Design Tools*** *screen, the macro runs automatically each time you run* ***Draft****. You can also run it when you are at the main menu level of* ***Schematic Design Tools*** *by pressing <Ctrl> <A>.*

Summary

In this chapter you learned how to run **Draft,** and examine and modify work conditions. You also learned how to capture an initial macro and have it automatically set up work conditions each time you run **Draft**.

The next chapter gives you instructions for capturing the schematic for the clock oscillator circuit. In later chapters, you build on the knowledge you gain while learning more about **Schematic Design Tools**.

Capturing the clock oscillator schematic

This chapter shows you the processes used to create a basic schematic. In this chapter, you learn how to:

- Get and place library parts
- Place wires
- Place junctions
- Place labels and text

Running Draft

Figure 3-1 shows the schematic diagram of the clock oscillator circuit you create in this chapter. Refer to this figure for placement and orientation information while capturing the clock oscillator schematic.

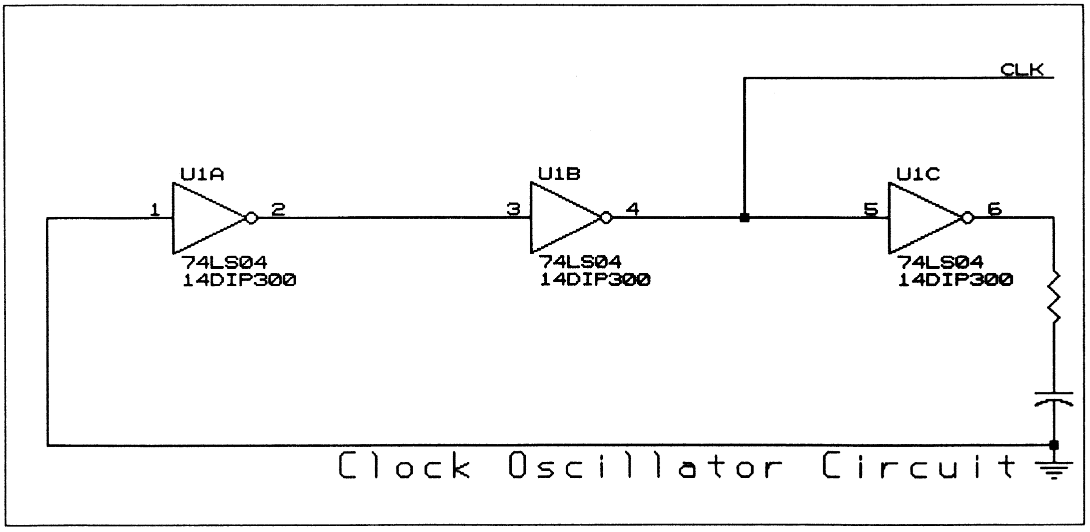


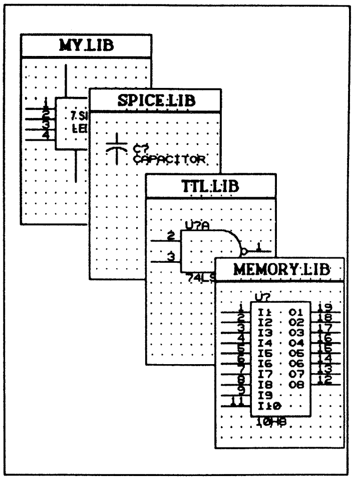
Figure 3-1. Clock oscillator circuit schematic.

About symbols

The first step in building a schematic diagram with **Draft** is to place symbols for the parts on the worksheet. The symbols can represent basic logic functions (such as AND gates), individual parts (such as capacitors), or blocks of circuitry to be designed later. The symbols can represent parts that use different technologies, such as TTL or CMOS.

About libraries

Symbols representing parts are stored in libraries. For **Draft** to get a symbol and place it on a schematic, the library containing it must be configured on the **Configure Schematic Design Tools** screen.



Parts libraries.

As shown in the illustration at right, library filenames typically end with the extension .LIB.

To build the clock oscillator, you need the following symbols:

- ❖ Three inverters
- ❖ One resistor
- ❖ One capacitor

The examples in this tutorial use TTL technology for the inverters.

Where to start

If you are continuing from chapter 2, the **Schematic Design Tools** screen is displayed. Follow these steps if it is not displayed:

1. If the operating system prompt is displayed, enter **ORCAD**.

△ *NOTE: In chapter 1, you set the start-up design to be TUTOR. Check to be sure that "TUTOR Design" is displayed in the middle of the top line of the screen. If it is not, go into **Design Management Tools** and change to the TUTOR design. This process is described in detail in chapter 1.*

2. On the main screen, click the **Schematic Design Tools** button and then select **Execute**.

Check library files

Follow these steps to check which libraries are configured:

1. On the **Schematic Design Tools** screen, click the **Draft** button.
2. Select **Configure Schematic Tools**. The **Configure Schematic Design Tools** screen displays.
3. Pan down until you can see the **Library Options** area.

   As shown in figure 3-2, **Library Options** shows **Available Libraries** on the left, and **Configured Libraries** on the right.

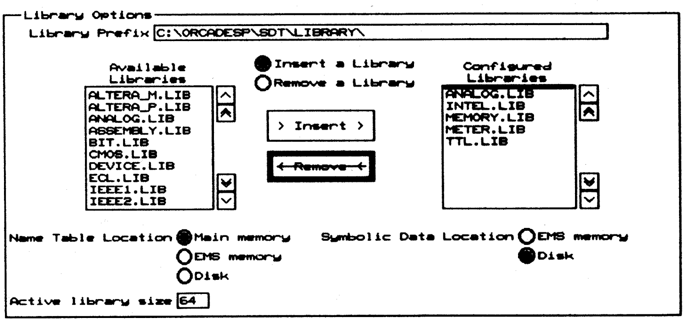


*Figure 3-2. **Library Options** area of **Configure Schematic Design Tools** screen.*

Draft loads and maintains libraries in your computer's memory in the order in which they are listed in the **Configured Libraries** box. This is important when retrieving parts while creating schematics. When you tell **Draft** to get a certain part name, it searches the libraries in the order listed during configuration and gets the *first* part it finds with a matching name.

4. For this chapter, **Draft** needs the library files PCBDEV.LIB and .\DCLOCK.LIB. PCBDEV.LIB must be the first library listed in the **Configured Libraries** box.

   Scroll the **Available Libraries** list box up and down by clicking the up and down arrow keys to the right of the list box. The double-arrow keys scroll the list a full box at a time.

5. Locate .\DCLOCK.LIB in the **Available Libraries** list box and select it. Click the **>Insert>** button to place .\DCLOCK.LIB in the **Configured Libraries** list box.

6. Repeat step 5, but this time select PCBDEV.LIB.

*NOTE: If any libraries other than PCBDEV.LIB and .\DCLOCK.LIB are listed in the **Configured Libraries** list box, remove them. To do this, click the **Remove a Library** button, select the library to remove, and then click the **<Remove<** button.*

7. Pan to the top of the configuration screen (or press the <Home> key) and click **OK** to return to the **Schematic Design Tools** screen.

8. Click the **Draft** button, and then select **Execute.**

   Draft runs the initial macro you captured in chapter 2. This macro sets the viewing scale to full size and causes the X,Y coordinates, grid references, and visible grid dots to be displayed. When the macro is done running, a blank worksheet displays.

Placing parts

Follow these steps to get symbols from part libraries:

1. Select the **GET** command from the main menu. The "Get?" prompt displays.

2. Press <Enter> to display the **Which Library**? menu. The menu at right shows the libraries configured for the TUTOR design.

```
Which Library?
PCBDEV.LIB
.\DCLOCK.LIB
```

3. Move the highlight to .\DCLOCK.LIB and press <Enter>. A list of the parts stored in the .\DCLOCK.LIB file displays, as shown at right.

```
Get?
22V10
4SW SPST
74LS04
BATTERY
CAP
GND
LM7805
R
SW PUSHBUTTON
```

4. Select the 74LS04 inverter. An image of the part displays on the worksheet and a command line displays across the top of the screen. When you move the part, the image simplifies temporarily—only the object's outline displays. When you stop moving the part, details redisplay.

5. Move the part to its general location on the worksheet. Refer to the grid reference bars at the left and top edges of the display and use the mouse to move the image to region A-3.

6. To move the part to its precise location, refer to the X,Y grid display at the upper right of the screen and move the image until the display shows that it is at location (12.80, 11.80). You can use the arrow keys to position the part. The part's upper left corner is its reference point for positioning.

7. Press <Enter> and select **Place** from the menu that displays. **Draft** places the part on the worksheet and creates another movable image of the part.
8. Since you need two more copies of the inverter, place copies of the part at locations (14.80, 11.80) and (16.80, 11.80).
9. When you have placed all three parts, press <Esc> to end the operation.

Shortcuts for getting parts

If you know the full name of a part you want to get from a library, you don't have to work your way through the menus. Simply type the complete part name at the "Get?" prompt. For example, if you enter **R** in response to the "Get?" prompt, **Draft** searches through the libraries and displays a resistor.

Place the remaining parts

Follow these steps to add a resistor, a capacitor, and a ground symbol to the clock oscillator circuit:

1. Press <G> <R> to select the resistor part from the PCBDEV.LIB library. An image of the resistor displays on the worksheet.
2. Place the resistor at location (17.60, 12.30).
3. Press <Esc>.
4. Press <G> to display the "Get?" prompt. Enter **CAP**. An image of the capacitor displays on the worksheet.
5. Place the capacitor at location (17.60, 13.00).
6. Press <Esc>.
7. Press <G> to display the "Get?" prompt. Enter **GND**. An image of the ground symbol displays on the worksheet.
8. Place the ground symbol two grid spaces below the capacitor symbol, and then press <Esc>.

You have now placed all the parts and symbols for the clock oscillator circuit on the worksheet. The next step is to place the wires.

Placing wires

Compare your worksheet with figure 3-1. Your worksheet should contain the parts shown in figure 3-1, but not the wires.

Most of the remaining tasks in this chapter establish signal connections between the parts you placed on the worksheet.

Place wires

Follow these steps to place wires on your schematic:

1. Select **PLACE** from the main menu. The **PLACE** menu displays.
2. Select **Wire**. The **PLACE Wire** command line displays.
3. Move the pointer until it rests at the free end of the output pin of the left-most inverter. This is location (13.50, 12.00).
4. Select **Begin**, then move the pointer right to the input pin of the next inverter.
5. Select **End**. The wire segment is completed.
6. To complete the wiring, place wires between the remaining parts as shown in figure 3-1.

   You can speed up wire placement two ways:

   - Select **New** instead of **End** for each wire except the last one.
   - Press <P> <W> <B> <N> and <E> to select the required menu commands.

△ *NOTE: When placing wires, be sure to begin and end each wire segment at the end of a part pin, not within the body of the pin. Also be sure that the end of a wire does not overlap a pin. If you accidentally overlap wires on pins or part bodies, error messages result.*

Placing junctions at intersections

Wires that cross one another do not represent a connection. To tell **Draft** that the crossing wires are connected, you must define the intersection as a wire junction. You do this by placing a junction at the intersection.

However, if two wires (or a wire and a part pin) are connected end-to-end, a junction is not necessary. The connection between the resistor-capacitor junction and the input of the left-most inverter requires a junction. This is not necessary for the connection between the capacitor and the ground wire since they connect end-to-end.

A junction is also required where the wire connects between the middle and right-hand inverters and ends at a point above and to the right of the rightmost inverter and is labeled CLK in figure 3-1.

Place junctions

Follow these steps to place a junction:

1. Select **PLACE**, then **Junction**.
2. Put the pointer on one of the wire intersections and select **Place**. A junction displays.
3. Place a junction at the other intersection by putting the pointer on it and selecting **Place**.
4. Press <Esc> to dismiss the **Place** command line.

You aren't finished with this circuit yet. You still have to assign values to the resistor and capacitor, add a signal label, and assign reference designators to all the parts. These steps are described in the next sections.

Editing part fields

Each part in **Schematic Design Tools** has ten reserved data areas called *part fields* for holding and displaying additional information. For example, you might want to record part numbers on the schematic to make it easier to track and order parts from manufacturers. Or you may want to specify the multiple-element part to which a particular sub-part belongs.

Two of the ten part fields are reserved for particular types of data:

- The **Reference** field is reserved for holding reference designator values, such as "U1A" or "Q1."
- The **Part Value** field is reserved for holding part names, such as "74LS04" or values relevant for the part, such as ohm (Ω) values for resistors.

The other eight fields are named **1st Part Field** through **8th Part Field.**

To be processed correctly by **Schematic Design Tools,** every part *must* have data in the **Reference** field and in the **Part Value** field.

The data in a part field can be up to 128 characters long. You can edit the contents of these fields and make them visible or invisible on the schematic using the **EDIT** command.

In this chapter, you learn how to edit part fields one at a time. Alternatively, you can automate part field editing using the **Update Field Contents** tool. You will learn how in chapter 7.

Edit part fields

Follow these steps to specify the multiple-element part for the inverters:

1. Select **EDIT** from the main menu.
2. Put the pointer on the part you want to edit, in this case the leftmost inverter.
3. Select **Edit**. The **Edit part** menu at right displays.

```
Edit part
Reference
Part Value
1st Part Field
2nd Part Field
3rd Part Field
4th Part Field
5th Part Field
6th Part Field
7th Part Field
8th Part Field
SheetPart Name
Orientation
Which Device
```

4. Select **1st Part Field**. The menu at right displays.

```
1st Part Field
Name
Location
Visible
```

5. Select **Name. Draft** displays:

```
1st Part Field?
```

6. Enter **`14DIP300`**. After you do, this information displays below the inverter symbol.
7. Select **Which Device** from the **Edit Part** menu.

   The prompt "Which device from package?" and a list of suffix letters (A through F) displays. A through F represent the six 74LS04 inverters in the 14DIP300 multiple-element part.
8. Select **A** from the list and press <Esc>.
9. Repeat steps 2 through 8 for the other inverters you placed. Since they are from the same multiple-element part, enter **14DIP300** for each, and assign suffix letters **B** and **C** to them.
10. Press <Esc> twice to remove the menus from the screen.

About reference designator assignments

Notice that the suffix letters of the reference designators change to U?B and U?C, respectively. U?A is the first part in the multiple-element part, U?B is the second part, and U?C is the third part. When you run the **Annotate Schematic** tool on this schematic, it changes all of the question marks for this multiple-element part to a common number, such as 4. The parts will then be labeled U4A, U4B, and U4C. The **Annotate Schematic** tool is described in *Chapter 7: Using Schematic Design Tools.*

You also can edit the reference designator and part values displayed for a part, but doing so prevents **Annotate Schematic** from performing this task. **Annotate Schematic** automatically updates part reference designators and pin numbers associated with the reference designators in multiple-element parts.

Edit part fields for the remaining parts

Follow these steps to edit the part fields for the resistor and capacitor:

1. Select **EDIT** from the main menu.
2. Put the pointer on the resistor.
3. Select **Edit**. The **Edit Part** menu displays.
4. Select **Part Value** and **Name**. **Draft** displays:

```
Value? R
```

5. To change the value, backspace over the present value and enter **91K**.
6. Press <Esc> to dismiss the "Value?" prompt.
7. Put the pointer on the capacitor.
8. Repeat steps 3 through 6 to change the part value of the capacitor, measured in microFarads (**uF**), to `47uF`.

You are nearly finished with the schematic for the clock oscillator circuit. In the next section, you learn to connect a wire in your circuit using a label. The label allows another circuit on the worksheet to behave as though it is directly connected to the output of this circuit.

Specifying connections with labels

Sometimes you may want to connect wires far apart on the worksheet. To keep the worksheet from looking cluttered, you'd like to do so without having to place a line representing the wire connecting them. You can do this by assigning a label with the same name to both wires.

Add a label

Follow these steps to add a label to a wire:

1. Select **PLACE** from the main menu.
2. Select **Label.** At the "Label?" prompt, enter **CLK**. The label displays.
3. Position the label image so the pointer rests on the unconnected output wire of the clock oscillator circuit. Labels must be placed with the leftmost point of the label name next to the bus or wire.
4. Select **Place**. The "Label?" prompt redisplays.
5. Press <Esc> to dismiss the "Label?" prompt.

Schematic Design Tools treats all wires on this sheet labeled "CLK" as connected, just as if you had placed the wire from the clock oscillator circuit directly to the other area of the schematic that is using it. You will reference this wire label in a later chapter of this guide.

Placing comment text

You may often want to leave notes or descriptive text (that isn't required by the circuit) on a schematic diagram. Such text helps you and others understand the functions being performed or documents some aspect of circuit operation.

Add a title

Follow these steps to add a descriptive title to this circuit:

1. Select **PLACE** from the main menu, and then select **Text**.
2. The prompt "Text?" displays. Enter `Clock Oscillator  Circuit`.
3. To use the next larger type size for the text, select **Larger**. The image of the text becomes larger.
4. Move the text image so it is centered immediately below the circuit diagram and select **Place**. The "Text?" prompt redisplays.
5. Press <Esc> to dismiss the "Text?" prompt.

△ *NOTE: You may wish to use the **ZOOM Center** commands to center the circuit before placing this text.*

Updating the file

This circuit is now complete. To save your work and exit **Draft**, follow the same steps you took earlier. Select **QUIT**, then **Update file**, then **Abandon Edits**. **Draft** exits and the **Schematic Design Tools** screen displays. When you select **Update file**, the file is saved in TUTOR.SCH.

Summary

You just completed the schematic diagram for the clock oscillator circuit of the digital clock. In the next chapter, you capture the schematic of the power regulator circuit.

Capturing the power regulator schematic

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

- Move a group of parts
- Delete a part
- Undo a delete
- Rotate a part
- Place a power symbol
- Capture and use a macro
- Set a tag
- Jump to a tag or reference location
- Print the worksheet

Figure 4-1 shows the schematic diagram of the power regulator circuit you create in this chapter. Refer to this figure for placement and orientation information while performing the exercise.

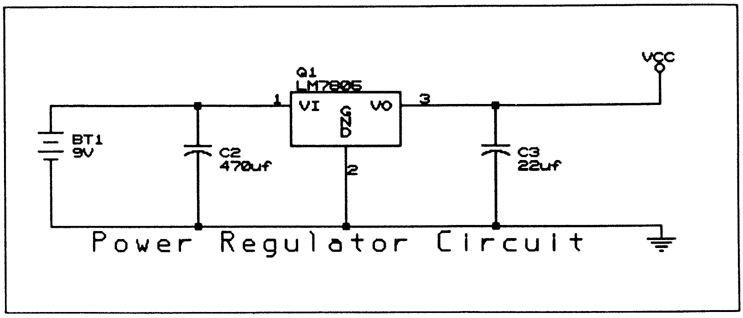


Figure 4-1. Power regulator circuit schematic.

Continuing schematic capture

If you did not abandon edits at the end of chapter 3, you can skip to the next section. Otherwise, follow these steps:

1. From the **Schematic Design Tools** screen, select **Draft**.
2. Select **Execute**. The worksheet view of the TUTOR.SCH schematic that was last active displays.

Moving a group of objects

Although you could just move your viewpoint over to another area of the worksheet to begin working on the power regulator schematic, now is a good time to learn about **BLOCK Move.**

Move the clock oscillator circuit to another place on the worksheet

Before beginning a **BLOCK Move,** zoom out so you can see all of the objects you are moving, as well as the beginning and ending points of the move. Follow these steps to move the clock oscillator circuit:

1. Change the scale from one to five, by selecting **ZOOM Out** twice, or **ZOOM Select 5**. The entire worksheet displays.
2. Select **BLOCK** and then select **Move.**
3. Place the pointer above and to the left of the clock oscillator circuit, and select **Begin**.
4. Move the pointer below and to the right of the circuit. As you move the pointer, a rectangle expands and contracts.
5. When the rectangle encloses the entire circuit, select **End**. The rectangle locks onto the circuit.
6. Move the outline of the circuit until it is centered in the B-2 region of the worksheet.
7. Select **Place** to move the clock oscillator circuit. The circuit moves to the new location.
8. Use **ZOOM** to return to a one-to-one scale. Place the pointer in the A-2 area of the worksheet and select **ZOOM Center. Draft** moves the view of the worksheet so that the pointer displays in the center of the screen. You are now ready to capture the schematic for the power regulator circuit.

Building the power regulator circuit

To build the power regulator circuit, you need the following parts:

- An LM7805 IC regulator
- Two capacitors
- A nine-volt battery
- Power (V_{CC}) and ground (GND) symbols

As in chapter 3, the digital clock parts library (.\DCLOCK.LIB) contains the parts you need to construct the power regulator circuit.

Get library parts and place them on the worksheet

Follow these steps to get the necessary parts for the power regulator circuit:

1. Select **GET** from the main menu. The "Get?" prompt displays.
2. Press <Enter> then select **.\DCLOCK.LIB**.
3. The parts menu displays. Select an LM7805 (an IC regulator) and place it at location (15.00, 12.50), as shown in figure 4-1.
4. By now you should be experienced at placing parts. Get the capacitor and place one on each side of the regulator, as shown in figure 4-1.
5. Now get the ground symbol (**GND**) and place it in the location shown in figure 4-1.

Deleting parts from the worksheet

If you place a part and then decide you don't need it after all, **Draft**'s **DELETE** command lets you remove any object placed on the worksheet.

If you delete an object by mistake, you can **Undo** your action.

Delete an object

To familiarize yourself with the delete process, delete the capacitor on the right side of the IC regulator. Follow these steps:

1. Select **DELETE** from the main menu. The **DELETE** menu displays, as shown below.
2. Select **Object**. The **DELETE Object** command line displays.
3. Put the pointer on the rightmost capacitor.
4. Select **Delete. Draft** deletes the capacitor from the worksheet.

   Because of the way **Draft** deletes things, some dots may remain on the screen where the deleted object was.They are not really on the worksheet.
5. Press <Esc>. **Draft** redraws the screen.The extra dots disappear.

```
Delete
Object
Block
Undo
```

Recover a deleted object

Follow these steps to recover the deleted capacitor:

1. Select **DELETE** again.
2. Select **Undo**.The capacitor redisplays.

Rotating parts before they are placed

Now you are going to try something a little different. A battery symbol exhibits polarity, so even though you know that the negative terminal goes to ground, the symbol may end up backwards on the schematic if you are not careful. You may have to rotate the part to get the polarity correct. Follow these steps to familiarize yourself with this process:

1. Get the battery part (**BATTERY**) from the .\DCLOCK.LIB library. Once the part is selected, the **Get Part** command line displays:

   ```
   Place Rotate Normal Up Over Down Mirror Find
   ```

2. Select **Rotate** twice and see the effect this has on the battery symbol.

   Before placing the part, experiment with the other **Place** commands to see their effect on the part orientation.

3. If you look closely at the part, you'll notice that the pin 1 end of the part has a long heavy line as part of the border. The pin 2 end has a shorter heavy line as part of the border. The long heavy line indicates the positive terminal of the battery. As shown in figure 4-1, you want the positive terminal up, so rotate the symbol to this orientation (**Down**), and place it on the worksheet.

You have now placed all the parts and symbols, except for the V_{CC} power symbol associated with the power regulator circuit on the worksheet. Next you place the wires for the power regulator circuit.

Placing wires

Compare your worksheet with figure 4-1. Notice that several wires are missing from your worksheet.

In this section, you learn how to place multi-segment wires in one operation. A multi-segment wire is a single wire that changes direction several times.

Place a multi-segment wire

Follow these steps to place a multi-segment wire:

1. Select **PLACE Wire**. The **PLACE Wire** command line displays.
2. Move the pointer to the negative terminal of the battery, and select **Begin**.
3. Move the pointer down approximately three grid spaces.
4. Select **Begin** again and move the pointer to the right until it is directly under the first capacitor.
5. Select **Begin** again and move the pointer to the end of the capacitor pin.
6. Select **End** or **New**. When you place multi-segment wires, remember to start and turn corners with **Begin** and cut the wire with **End** or **New**.
7. Now, connect wire segments between the remaining parts as shown in figure 4-1. Be sure to **Begin** and **End** each wire segment at the end of a part pin, not within the body of the part.
8. Using the **PLACE Junction** command, place junctions in the circuit at the five locations shown in figure 4-1.

△ *NOTE: If you cut a wire with **New**, the **PLACE Wire** command line remains displayed.You don't need to select **PLACE Wire Begin** to start a new wire. You only need to select **Begin**.*

More macros

You could continue placing wires using keyboard or menu commands, but it's a repetitious process. Every time you begin placing a wire, you must enter three commands in sequence, **PLACE**, **Wire**, and **Begin**. You can do this by pressing the first letters of each command, <P><W><B>.

Or, you can use **Draft**'s macro feature to make it even easier by combining these three keystrokes into one keystroke. You were introduced to macros when you developed the initial macro that sets up the work conditions each time **Draft** runs.

The following is a simple example of how to capture a macro. You can extend the principle to create complex macros, automating long command sequences.

Capture a macro to begin wires

Follow these steps to capture a macro to begin wires:

1. Select **MACRO**. The **MACRO** menu displays.
2. Select **Capture**. The "Capture macro?" prompt displays.
3. Press <F1> to assign a keystroke to this macro. "F1" displays at the "Capture macro?" prompt.
4. Press <Enter>.The message "<macro>" displays to remind you that you are capturing a macro and that any commands you select are added to the list of commands being stored in the macro.
5. Type the commands required to begin a wire by pressing <P> <W> <B>.
6. Press the key combination <Ctrl><End> to end the macro definition. The message "<<<MACRO END>>>" displays.

The macro you captured is now stored in the computer's memory and can be run by simply pressing the key you specified, in this case, <F1>. However, should you turn your computer off, the macro will be lost. You must save the macro to a file.

Save the macro

Follow these steps to save the macro:

1. Select **MACRO Write.** The "Write all macros to?" prompt displays.
2. Enter the following filename for this macro:

```
Write all macros to? tutor.mac
```

3. **Draft** displays, "WARNING: File: tutor.mac exists. Write over it?" and a short menu. Since all the macros in TUTOR.MAC were read into the macro buffer when you ran **Draft**, select **Yes.**

You just saved this macro—and the macros that were already in TUTOR.MAC—to the macro file that automatically loads each time you start **Draft**. You can add more macros to this file as you capture them.

Placing the power symbol

Follow these steps to place the power symbol in the power regulator circuit:

1. Select **PLACE Power.** An image of the power symbol displays, with the value V_{CC} above it. The **PLACE Power** command line displays:

```
Place Orientation Value Type Find Jump Zoom
```

   In this example, the power symbol is connected at the top of the wire. However, there may also be cases in which you need to turn the power symbol around.

2. Select **Orientation** to change the power symbol's orientation. The **Orientation of Power Value** menu shown below displays. The image of the power symbol disappears until you make a selection from this menu.

3. Practice changing the orientation of the power symbol. When you finish, select **Top** orientation.

```
Orientation of Power Value
Top
Bottom
Left
Right
```

   See the *Schematic Design Tools Reference Guide* for detailed information about the display options available for the power symbol.

4. Now move the image of the power symbol until it rests on the end of the wire, as shown in figure 4-1, and select **Place.**

5. Press <Esc> to dismiss the **PLACE Power** command line.

Dragging wires

You may often want to move parts without having to replace the wires connected to the parts. Use **BLOCK Drag** to do this. Follow these steps to familiarize yourself with this process:

1. Select **BLOCK Drag. Draft** displays:

```
Begin  Find  Jump  Zoom
```

2. Move the pointer above and to the left of the power regulator circuit and select **Begin.**
3. Move the pointer so the rectangle encloses all of the power regulator circuit, except the ground symbol and the bottom wire of the circuit.
4. Select **End**. The circuit changes color.
5. Move the selected circuitry up approximately two grid spaces.
6. Select **Place.** Notice that the lower wires grow and remain connected to the ground wire.

Editing part fields

For the parts in the power regulator circuit, you need only specify the correct part values for the capacitors and battery. After all of the schematic is captured, you use the **Annotate Schematic** tool to update the other fields.

Edit part values for the capacitors and battery

Follow these steps to edit part values for the capacitors and battery:

1. Place the pointer on the leftmost capacitor and select **EDIT Edit Part Value Name**. The "Value?" prompt displays. Change the part value to **`470uF`**.
2. Place the pointer on the rightmost capacitor and change the part value to **`22uF`**.
3. Place the pointer on the battery and change the part value to **9V**.

Placing comment text

A title isn't necessary for a circuit, but it is helpful when someone new needs to understand what a portion of circuitry does.

Add a title

Follow these steps to add a descriptive title to this circuit:

1. Select **PLACE Text** command and enter **`Power Regulator Circuit`**.
2. Select **Larger** to use the type size that is one step larger than the part labels.
3. Center the text immediately below the schematic diagram and select **Place**.

Changing viewpoints

You have now captured two separate schematics on the same worksheet. At times, you may want to quickly change your viewpoint from one area of the worksheet to another. You use the **JUMP** command to do this.

Jump to new coordinates

Follow these steps to move around the worksheet:

1. Select **JUMP**. The **JUMP** menu displays, as shown below. You can move around the worksheet three ways:

   ```
   Jump
   A tag
   B tag
   C tag
   D tag
   E tag
   F tag
   G tag
   H tag
   Reference
   X location
   Y location
   ```

   - Using **X location** and **Y location**, specify the number of grid steps to add or subtract from the current pointer coordinates.
   - Using **Reference**, specify a new pointer location using grid reference regions, such as "A3."
   - Using the tags, move to a pointer location you defined earlier using the **TAG** command.

2. Select **X location**. The prompt "Jump X" displays. Note the current pointer coordinates.

3. At the prompt, enter `+5`. The pointer moves five grid spaces to the right (in the positive direction) and the X reference coordinate reflects a change of 0.50 inches (since each grid space is 0.10 inches).

   - To move left, enter a negative X value.
   - To move to an exact X reference, enter a value without a positive or negative sign. For example, to move to X reference .5, enter **5**; or to move to X reference 5.0, enter `50`.
   - To move up and down, use the **Y location** command.

4. Experiment for a moment with these commands and positive, negative, and unsigned **JUMP** values.

Tag and jump to specific locations

The **TAG** and **JUMP Tag** commands are useful when you need to return again and again to a particular location on the worksheet. The **TAG** command assigns a tag to a location on the worksheet. Then the **JUMP Tag** command moves the pointer to that location. Follow these steps to practice assigning tags and jumping to them:

1. Place the pointer on the power regulator circuit.
2. Select **TAG** from the main menu. The **Tag set** menu displays, listing eight tag names you can use.
3. Select **A tag**.
4. Move to the clock oscillator circuit. Put the pointer in the middle of the center inverter, and assign it **B tag**.
5. Select **JUMP**, and from the **JUMP** menu, select **A tag**. The pointer jumps to the middle of the power regulator circuit, where you assigned the **A tag**.
6. Now jump to the **B tag**.

Making a draft-quality print

The last thing to do before ending this chapter is to print out a copy of the worksheet. While **Schematic Design Tools** includes the **Print Schematic** and **Plot Schematic** tools for making copies of entire designs, **Draft** also has a quick way to get a draft-quality print: the **HARDCOPY** command.

To do this, your computer must be connected to a printer. **HARDCOPY** does not work for plotters. The correct printer driver program must be installed along with your other **Schematic Design Tools** software.

Update the file

Follow these steps to save your work before you print the schematic:

1. Select **QUIT** and **Update file. Draft** updates the file TUTOR.SCH to reflect the current state of the worksheet.
2. Press <Esc> to dismiss the **QUIT** menu.

Make a hardcopy of the worksheet

Follow these steps to make a hardcopy of the worksheet:

1. Make sure the printer is connected to your computer, powered on, and online.
2. Select **HARDCOPY** from the main menu. The **HARDCOPY** menu displays.
3. Select **Width of Paper.** Choose the correct paper width for your printer. Select **Narrow** for paper 8.5 inches wide; select **Wide** for paper 13 inches wide.

   After you specify width, **Draft** redisplays the **HARDCOPY** menu.
4. Select **Make Hardcopy. Draft** sends the worksheet to the printer.

△ *NOTE: The size of the printed image depends on the printer driver* ***Draft*** *uses. With* ***HARDCOPY*** *(and the* ***Print Schematic*** *tool),* ***Draft*** *always produces an image at a resolution of 100 dpi (dots per inch). If the printer driver used prints at some other resolution, the image printed is changed by a fixed scale factor (100 dpi divided by the printer driver resolution).*

- *If the printer driver resolution is greater than 100 dpi, the printed image is smaller.*
- *If the driver resolution is less than 100 dpi, the printed image is larger.*

For example, if the printer driver you are using prints at a resolution of 300 DPI, the image printed on the paper is reduced in size by a factor of 100\300, or 1\3X. If the driver prints at 75 DPI, the image printed is enlarged by a factor of 100\75, or 1.33X.

For more information on sending designs to printers and plotters, see the **HARDCOPY** command and the **Print Schematic** tool and the **Plot Schematic** tools in the *Schematic Design Tools Reference Guide.*

Ending a Draft work session

After you save your design and make a hardcopy, you are done with chapter 4. You need to exit **Draft** to perform steps in the next chapter.

Since you already saved your work, just select **QUIT** and then **Abandon Edits**. **Draft** exits and the **Schematic Design Tools** screen displays.

Summary

You have completed the schematic diagram for the power regulator circuit of the digital clock. In the next chapter, you use **Edit Library** to create a custom part to use in the display area of the digital clock schematic.

CHAPTER 5

Creating a custom part

Although **Schematic Design Tools** provides extensive libraries containing over 20,000 parts, you may occasionally need a part or symbol not in any library. You use the **Edit Library** tool to modify an existing part or create an entirely new part.

In this chapter, you learn how to:

- Run **Edit Library**
- Reconfigure **Edit Library**'s work conditions
- Draw a part body
- Draw special shapes
- Use shading and fills
- Add pins to the part body
- Add pin names
- Save the new part in a library

Running Edit Library

Edit Library performs a variety of tasks for creating and modifying custom parts and libraries. In this chapter, you create a completely new part to add to an existing library file. For detailed discussions of **Edit Library** commands, see the *Schematic Design Tools Reference Guide.*

Configure Edit Library

Before running **Edit Library,** you must configure it to open the library file called .\DCLOCK.LIB. Follow these steps:

1. Select **Edit Library** from the **Schematic Design Tools** screen.
2. Select **Local Configuration** from the menu that displays and then select **Configure LIBEDIT. Edit Library's** configuration screen displays.
3. Look for the .\DCLOCK.LIB file in the **Files** list box. Select .\DCLOCK.LIB. The name .\DCLOCK.LIB displays in the **Source** entry box:

   `Source` | `.\DCLOCK.LIB` |

4. Click the **OK** button to save the configuration.

Run Edit Library

From the **Schematic Design Tools** screen, select **Edit Library** and **Execute.**

The **Edit Library** screen displays. Initially it is blank, except for pointer coordinates displayed at the upper right of the screen.

Setting up the work conditions

As with **Draft**, you set up certain work conditions in **Edit Library**. You adjust two features: one governs visibility of the outline of the part body. The other governs visibility of the grid dots in the work area.

Make part body border and grid dots visible

Follow these steps to set up the work conditions:

1. Press <Enter> to display **Edit Library's** main menu.
2. Select **SET** from the main menu. The menu shown below displays.
3. Select **Show Body Outline**. "Show Bitmap Body Outline?" displays.
4. Select **Yes**.
5. Select **SET Visible Grid Dots Yes**. Grid dots display in the work area.

Set

AutoPan	YES
Backup file	YES
Error Bell	YES
Left Button	NO
Macro Prompts	YES
Power Pins Visible	NO
Show Body Outline	NO
Visible Grid Dots	NO

Beginning a new part

To modify or create a part, you use the **GET Part** command. When you create a new part, choosing **GET Part** initiates a sequence of queries about the type of part you want to create. You will create a seven-segment LED named TIL309.

Open a part editing pad

Follow these steps to begin a new part:

1. Press <Enter> to display the main menu.
2. Select **GET PART**. The prompt "Get?" displays.
3. Enter `TIL309`, the name of the part you plan to create. **Edit Library** displays "TIL309 - New Part?" and a short menu.
4. Select **Yes**. The prompt "Sheet Path?" displays. This is relevant when you create a hierarchical design and want the part to reference another schematic worksheet.
5. Press <Enter> since the TIL309 part does not reference a schematic. The **Kind of Part?** menu displays. You use **Block** for simple rectangular parts, **Graphic** for more complex shapes, and **IEEE** for IEEE\ANSI drawing standard parts.
6. Select **Graphic** since the LED display is complex. The prompt "Number of Parts per Package" and a menu display.
7. Select **1** since the seven-segment LED display is a single-element part. The prompt "Does Graphic Part have CONVERT?" displays. This tells **Edit Library** whether you will also create a DeMorgan equivalent of the part you are creating.
8. Select **No**. The part editing pad displays, bordered by a solid line. Within the pad, a dotted border displays with the name you assigned the new part, **TIL309**. The pointer displays at the bottom right corner of the dotted border. The command line displays **Place** and **escape**.

   The dotted border defines the size and shape of the region within which you create the part body. Pins you attach to the part are created outside this region, with their connection points on the part body border.

You can adjust the size and shape of the dotted border by moving the pointer. Try it.

9. Move the pointer to location (+12.0, +12.0). This changes the part body border to a square shape. Figure 5-1 shows the part editing pad when the pointer is at location (+12.0, +12.0).

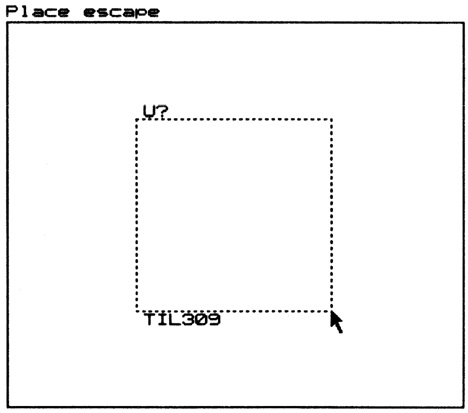


Figure 5-1. Part editing pad.

NOTE: Depending on your computer's monitor, the border may not look square due to the proportions of the screen display.

10. Select **Place** to set the size of the editing pad.The **BODY <Graphic>**menu displays.

Drawing the body outline

Follow these steps to draw the body outline:

1. Select **Line**. The **BODY Line** command line displays.
2. Move the pointer to the upper left corner of the body (location +.0, +.0) and select **Begin** .
3. Move the pointer to the next corner (location +12.0, +0) and select **Begin** again.
4. Move the pointer to the next corner (location +12.0, +12.0) and select **Begin** again.
5. Move the pointer to the next corner (location +.0, +12.0) and select **Begin** one last time.
6. Move the pointer to the first corner (location +.0, +.0) and select **End**. The **BODY <Graphic>** menu displays.
7. Press <Esc> to dismiss the **BODY <Graphic>** menu.

Changing the reference designator

Edit Library automatically puts a placeholder reference designator at the upper left of the part body border. The default class letter is the letter U and the default number is a question mark. The "?" serves as a placeholder for the values to be supplied when you use the part in a schematic and run **Annotate Schematic.** Because U conventionally designates IC parts, you need to change the class letter to D.

Change reference designator class letter to 'D'

Follow these steps to edit the reference designator:

1. Select **REFERENCE** from the main menu. The prompt "Initial Reference Designator? U" displays. U is the current value.
2. Backspace over the U and enter D. The reference designator reflects the change immediately.

Creating a part body

Now you are ready to create the part itself, in this case, a seven-segment LED display. The first step is to create the part body. It consists of seven rectangular objects arranged in the shape of a numeric display, and a circle for the decimal point, as shown in figure 5-2.

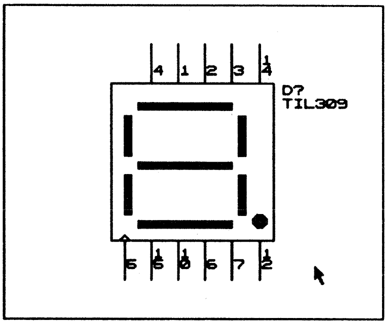


Figure 5-2. The part body you will create.

Zoom in to gain finer pointer control

Like **Draft, Edit Library** can display the part you are working on at several levels of detail.At the lowest level, level 1,the pointer snaps to grid points.

At either of the two higher magnification levels, you can move the pointer to any of 10 intermediate locations between the grid points. You need this fine control to draw the thin LED segments. Follow these steps:

1. Select **ZOOM In**. The image doubles in size.
2. Move the pointer between the grid points. Notice that the pointer no longer snaps to grid points.

Draw a rectangle to represent an LED

Follow these steps to draw the first LED segment:

1. Select **BODY** from the main menu. The **BODY <Graphic>** menu displays, as shown below.
2. Select **Line**. The **BODY Line** command line displays.
3. Place the pointer at location (+2.0, +1.5).
4. Select **Begin**. **Edit Library** is now in line-drawing mode.
5. Move the pointer to location (+9.0, +1.5). A line stretches behind the pointer to show the line segment you are creating.

```
Body <Graphic>
Line
Circle
Arc
Text
IEEE Symbol
Fill
Delete
Erase Body
Size of Body
Kind of Part
```

6. Select **Begin**. The line you drew changes color, showing that it is completed.
7. Move the pointer to location (+9.0, +2.0). This forms the right side of the rectangular shape. A line stretches from the first line to the pointer.
8. Select **Begin** again to complete this segment and begin another.
9. Move the pointer to location (+2.0, +2.0). This forms the bottom segment of the rectangle.
10. Select **Begin**.
11. Move the pointer to location (+2.0, +1.5), the starting point, to complete the rectangle.
12. Select **End** or **New** to end the last segment and complete the rectangle. The **BODY <Graphic>** menu displays.

Draw six more segments

Now, repeat this process for the remaining six rectangles that represent the LED segments. To ensure your LED comes out right, use the coordinates shown in table 5-1 to draw the remaining six rectangles. Each line of coord-inates defines one rectangle, starting with the leftmost coordinate and then drawing each segment to the next coordinate shown.

You can capture the commands for one rectangle as a macro, and then run it for each rectangle of the same size and orientation you want to draw, only in different locations.

To do this, move the pointer to the coordinates for the top left portion of a rectangle. Then select **MACRO Capture**. When the "Capture macro?" prompt displays, enter the key to be used to start the macro (such as <F2>). Now go ahead and draw the rectangle as described above. When the rectangle is complete, press <Ctrl><End> to end the macro. To draw the next rectangle, move the pointer to the coordinates for the top left portion of the new rectangle and press the key you assigned to the macro (such as <F2>).

	Top left	*Top right*	*Bottom right*	*Bottom left*
Segment 2	(2.0, 6.0)	(9.0, 6.0)	(9.0, 6.5)	(2.0, 6.5)
Segment 3	(2.0, 10.5)	(9.0, 10.5)	(9.0, 11.0)	(2.0, 11.0)
Segment 4	(1.0, 2.5)	(1.5, 2.5)	(1.5, 5.5)	(1.0, 5.5)
Segment 5	(9.5, 2.5)	(10.0, 2.5)	(10.0, 5.5)	(9.5, 5.5)
Segment 6	(9.5, 7.0)	(10.0, 7.0)	(10.0, 10.0)	(9.5, 10.0)
Segment 7	(1.0, 7.0)	(1.5, 7.0)	(1.5, 10.0)	(1.0, 10.0)

Table 5-1. Coordinates for rectangular LED segments. All coordinates are positive (+) values.

Add the decimal point

In addition to the seven rectangular LED segments, the display unit also has a circular LED at the lower right to represent a decimal point. Follow these steps to draw the decimal point:

1. Select **BODY Circle** to draw the circle.
2. Place the pointer at the location where you want the center of the circle, in this case, location (+11.0, +10.5).
3. Select **Center**. More commands display, one of which is **Edge**. Edge means the edge of the circle being defined. When you move the pointer, a circle expands and contracts.
4. Move the pointer to any location five pointer steps from the center point. For example, put the pointer at location (+11.5, +10.5).
5. Select **Edge. Edit Library** places the circle.
6. Press <Esc> to return to the **BODY <Graphic>** menu.

Shading closed shapes

When you create a part, you may want to shade certain objects to make them stand out. To do this you can use the **BODY Fill** command. Follow these steps to shade the LED shapes:

1. Select **Fill** from the **BODY <Graphic>** menu. The **Fill** command line displays.
2. Put the pointer within one of the LED shapes.
3. Select **Fill. Edit Library** fills in the shape.
4. Repeat steps 2 and 3 for all the LED shapes.
5. Press <Esc> twice to dismiss the **Fill** command line and the **BODY <Graphic>** menu.

After drawing the LEDs, you are ready to add pins so the part can be electrically connected when you place it in a schematic. Because this is a representation of an existing part, you want to add the pins corresponding to the standard version of the part.

Adding pins to a part

Edit Library's **PIN** command is used to add pins. Pins must terminate on the border of the part body. The dotted line around the part is the part's border. If the edge of a part body coincides with this border, pins can terminate directly on the part body. But if the part body is inside this border, you must make a connection between the part body and the border using the **BODY Line** command.

Add a clock pin

Follow these steps to add a clock pin:

1. Select **PIN** from the main menu. The **PIN** command line displays.
2. Move the pointer around. You'll find it is restricted to the part body border.
3. Put the pointer at a location on the border where you want to place the first pin. For this example, put it at coordinates (+1.0, +12.0).
4. Select **Add**. "Pin Name?" displays. The pin name is an identifier that **Draft** uses to identify particular pins. The pin name does not display on the graphic representation of a part.
5. Enter the name **STROBE**. The **Edit Library** tool assigns the name. The prompt "Pin Number?" displays.
6. Enter **5**. The **Pin Type** menu displays. The STROBE pin conducts a clock signal to the internal logic of the part. It should be defined as an input pin type.
7. Select **Input**. The **Pin Shape** menu displays.
8. Select **Clock** . **Edit Library** places the pin and displays the pin number you entered.

Add a reset pin

Follow these steps to add a reset pin:

1. Place the pointer at the coordinates (+11.00, +12.00).
2. Select **Add**. The prompt "Pin Name?" displays.
3. Enter the name `DPIN`. The **Edit Library** tool assigns the name. The prompt "Pin Number?" displays.
4. Enter `12`. The **Pin Type** menu displays. The DPIN pin conducts a reset signal to the internal logic of the part. It should be defined as an input type pin.
5. Select **Input**. The **Pin Shape** menu displays.
6. Select **Line. Edit Library** places the pin and displays the pin number you entered.

Add the remaining pins

Follow these steps to add the rest of the pins:

1. Put the pointer at a location where you want to place a pin. For this example, put it at coordinates (+3.00, +12.00).
2. Select **Add**. The prompt "Pin Name?" displays.
3. Enter the name `QAIN`. The **Edit Library** tool assigns the name. The prompt "Pin Number?" displays.
4. Enter `15`. The **Pin Type** menu displays. The QAIN pin conducts a signal to an LED segment. It should be defined as a passive type pin.
5. Select **Passive** by putting the highlight bar on this menu item, not by pressing <P>. This is because another menu item begins with "P" (**Power**) and displays in the menu before **Passive**; pressing <P> selects **Power,** not **Passive.**

   When you have specified the pin type, the **Pin Shape** menu displays.
6. Select **Line**.

7. Repeat these steps for the pins connected to the other LED segments. Table 5-2 lists the coordinates, names, pin numbers, pin type and pin shape to use for the other pins. You already defined the first three pins. Go ahead and start with the fourth pin.

Coordinates	*Pin Name*	*Pin No.*	*Pin Type*	*Pin Shape*
(1.0, 12.0)	STROBE	5	Input	Clock
(11.0, 12.0)	DPIN	12	Input	Line
(3.0, 12.0)	QAIN	15	Passive	Line
(5.0, 12.0)	QBIN	10	Passive	Line
(7.0, 12.0)	QCIN	6	Passive	Line
(9.0, 12.0)	QDIN	7	Passive	Line
(3.0, 0.0)	QAOUT	4	Passive	Line
(5.0, 0.0)	QBOUT	1	Passive	Line
(7.0, 0.0)	QCOUT	2	Passive	Line
(9.0, 0.0)	QDOUT	3	Passive	Line
(11.0, 0.0)	DPOUT	14	Passive	Line

Table 5-2. Pins for the TIL309 library part. All coordinates are positive (+) values.

8. Press <Esc>.

When you are finished, you should have 11 pins on the LED. The next step is to add the part to the library.

Saving a new part

Saving a new part involves two operations:

- Copy the part displayed on the screen to the part library currently loaded in the computer's internal memory. This is done by selecting **LIBRARY Update Current.**
- Write the modified library file in the computer's internal memory to disk. Select either **QUIT Update file** or **QUIT Write to file.**

Save the new part

Follow these steps to save the new part to the current library:

1. Select **LIBRARY.**
2. Select **Update Current.** The part currently displayed is written to the library now loaded in memory.

Write the library in memory to a file on disk

Follow these steps to write the current library to a file and confirm the operation:

1. Select **QUIT Update file. Edit Library** updates the library with the edits you performed during this session and then redisplays the **QUIT** menu.
2. To confirm that the part TIL309 has been stored in a library named .\DCLOCK.LIB, select **Initialize.** The prompt "Read Library?"displays.
3. Enter `.\DCLOCK.LIB`.
4. Select **LIBRARY List Directory Screen.** TIL309 should be in the list of parts in .\DCLOCK.LIB.
5. To dismiss the directory, press any key.

Get the new part

Follow these steps to retrieve the part you created:

1. Select **GET PART**. When the prompt "Get?" displays, press <Enter>. The **Get** menu displays containing the name of the part you created, TIL309.
2. Select the TIL309 part. It displays in the edit pad.
3. Select **QUIT Abandon Edits** to leave **Edit Library** and return to the **Schematic Design Tools** screen.

Summary

Using the **Edit Library** tool, you created a new part and saved it on disk in an existing library. In *Chapter 2: Introducing Schematic Design Tools,* you configured **Draft** to load the .\DCLOCK.LIB parts library. By adding the TIL309 part to this library, you made the new part available in **Draft** for use while capturing schematics.

Capturing the logic and display circuit schematic

This final schematic diagram for the digital clock circuit contains the logic and display circuit. This circuitry is more complex than the smaller schematics that you captured in the earlier chapters. The tasks you complete in this chapter are a natural progression from the processes that were introduced in the earlier chapters.

In this chapter you learn how to:

- Draw a repeatable portion of the schematic
- Make and place multiple copies of a schematic block
- Use repeat parameters to place wires and labels

Figure 6-1 shows the portion of the schematic you capture in this chapter.

Choosing parts

To build the rest of the digital clock schematic, you need the following parts:

- 22V10s
- TIL309 LED displays
- Resistors
- Capacitors
- Two switch types (SPST and pushbutton)
- Power (V_{CC}) and ground (GND) symbols

About TIL309 LED display chips

TIL309 display chips were selected in order to keep the chip count for the design down. These displays are capable of accepting binary-coded decimal input. Using TIL309s eliminates the need for decoder circuits. Six TIL309s are required: two each for seconds, minutes, and hours.

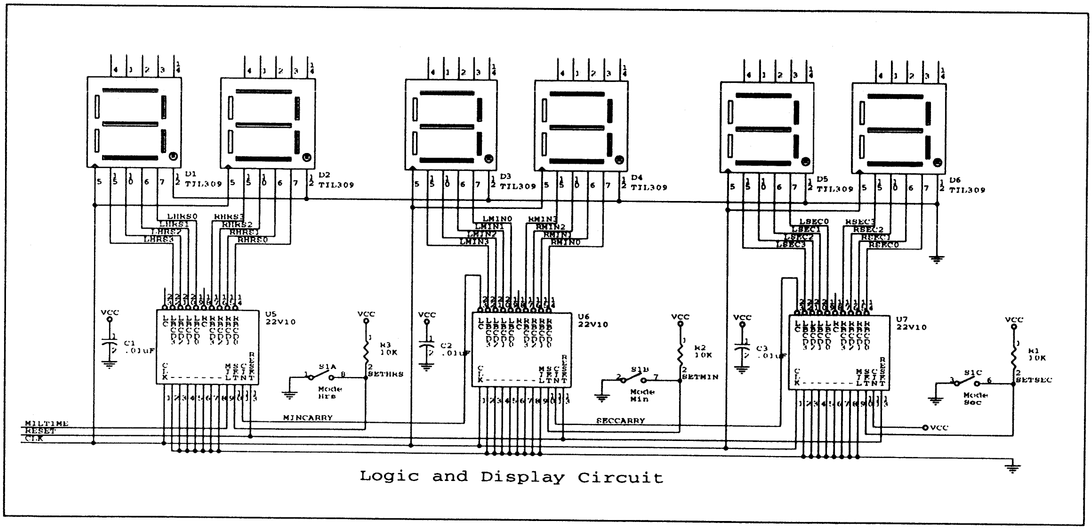


Figure 6-1. The logic and display circuitry.

The schematic requires enough pins to drive the six TIL309 display chips plus transfer carry signals. Once again, to reduce the total chip count for the design, 22V10s were chosen to drive the TIL309s rather than individual parts. Since the TIL309s are divided into pairs for seconds, minutes, and hours, you use one 22V10 per pair, three 22V10s altogether.

When deciding to use the 22V10s, the following factors were considered: number of inputs and outputs needed, complexity of the logic that the part needs to handle, cost, and availability. The 22V10s were chosen because they have enough inputs and outputs to accommodate fairly complex logic, are readily available from several manufacturers, and are not extremely expensive.

As in the previous chapters, the clock parts library (.\DCLOCK.LIB) contains the parts you need to construct this circuit. In chapter 5, you added the seven segment display part (TIL309) to the parts library.

Re-running Draft

Select **Draft** from the **Schematic Design Tools** screen. The worksheet view of the TUTOR.SCH schematic that was last active displays.

Drawing a portion of the schematic

As you look at the schematic of the logic and display circuitry in figure 6-1, it becomes apparent that three regions are nearly identical—seconds, minutes, and hours. Take advantage of this duplication by creating the schematic for the minutes area (figure 6-2), and copying it to the other areas.

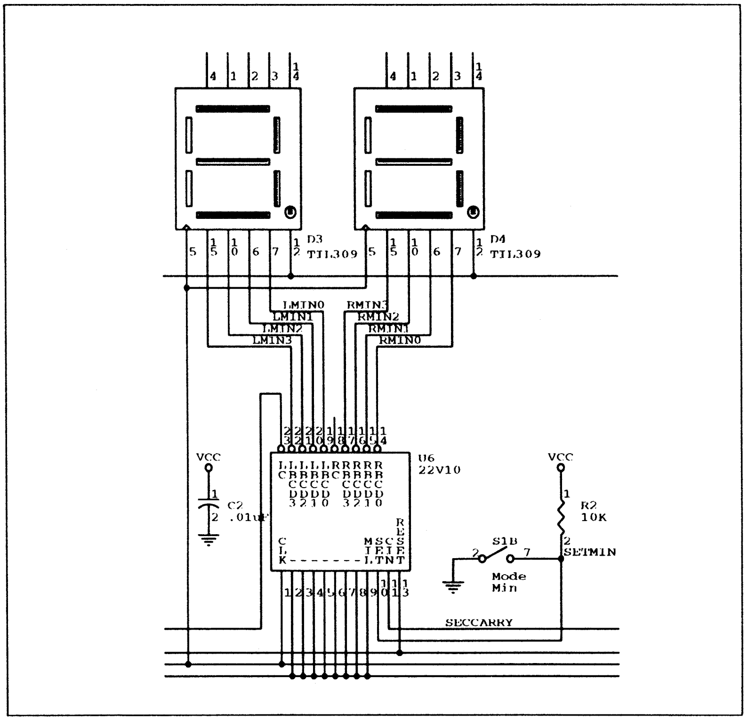


Figure 6-2. The minutes circuit.

Change viewpoint to a clear area

Follow these steps to move to an area of the worksheet with enough room to add the display and logic circuitry:

1. Select **ZOOM Select 2** to change the scale to two-to-one. This scale will work better for the tasks outlined in the next steps.
2. Select **JUMP Reference C 4** to move the display screen near the center of the worksheet.

The clock oscillator and power regulator circuits you captured earlier are in the lower right area (grid references A-2 and B-2) of the worksheet. The entire upper half of the worksheet is still vacant, so you can use it for this portion of the schematic.

The display and logic circuit shown in figure 6-1 contains so much detail that your immediate task of capturing the minutes area may seem more difficult than it actually is. Figure 6-2 shows only the parts and wires associated with the minutes area of the schematic.

By comparing figure 6-2 with figure 6-1, you see the similarities in each of the areas. The steps in the next section describe how to build the minutes circuit.

Place the parts

Follow these steps to place parts for the minutes circuit on the worksheet:

1. Select **GET**. The prompt "Get?" displays.
2. Press <Enter>. A list of the part libraries specified in **Schematic Design Tools'** configuration displays.
3. Select **.\DCLOCK.LIB**. A list of the parts in .\DCLOCK.LIB displays.
4. Select **22V10**. The part and a command line display. The part's orientation is not correct for this schematic, so you will have to rotate the part.
5. Select **Rotate** to change the part orientation to match the part orientation shown in figure 6-2.
6. Position the part at coordinates (10.50, 6.00) and select **Place.**
7. Get a **TIL309** and place copies at (9.60, 2.50) and (11.50, 2.50).
8. Get a resistor, **R**, and place it at (13.70, 6.30).
9. Get a capacitor, **CAP**, and place it at (9.90, 6.30).
10. Finally, get a switch, **4SW SPST**, and place it at (13.00, 6.80).

At this point you have placed all parts and only need to place wires, nets, and the power and ground symbols.

Place the wires

Follow these steps to place the wires:

1. Select **ZOOM In**. You need a close-up view of the schematic to perform the next steps.
2. Referring to figure 6-2, move the pointer to the bottom of the resistor symbol and select **Place Wire Begin** to start placing a wire.
3. Place the wire so that it is three grid spaces below the lower pins of the 22V10 part at (13.80, 7.60).
4. Select **Begin**.
5. Place the wire to (11.50, 7.60), and select **Begin**.
6. Continue the wire so that it connects to pin 10 on the 22V10 part (11.50, 7.30).
7. Select **End** to end the wire.

Run the macro to place wires

The <F1> macro you captured earlier to start placing a wire should still be active. Follow these steps to use the macro to place the rest of the wires:

1. Referring to figure 6-2, place the wires between the 22V10 part and the right-hand seven segment display as shown. To begin a new wire, instead of issuing the <P><W><B> commands, just press the <F1> key, and then proceed as usual.
2. Continue using the macro and place the wires between the 22V10 part and the left-hand seven segment display as shown in figure 6-2.

The <F1> macro lets you save some time, but there are other things you can do to save even more time. One time saver is the **REPEAT** command.

Define REPEAT parameters

REPEAT duplicates the last entered object, label, or text string and places it on the worksheet. Follow these steps to define the **REPEAT** parameters:

1. Select **SET Repeat Parameters.**
2. Select **X Repeat Step**. The prompt "X Repeat Step?" displays. Enter **1.**
3. Repeat step 1 and select **Y Repeat Step**. The prompt "Y Repeat Step?" displays. Enter **0.**

REPEAT is now set to place a new object exactly one grid space to the right of the pointer when you select **REPEAT**.

Change viewpoint to speed wire placement

The wire placements in the next steps work better if you center the display. Follow these steps:

1. Move the pointer to the end of pin 2 at the bottom of the 22V10.
2. Select **ZOOM Center** to change your viewpoint to center pin 2 on the worksheet.

Use REPEAT to speed up wire placement

Follow these steps to quickly place more wires:

1. Place a wire seven grid spaces long extending down from pin 2 of the 22V10 PAL. Press <F1> to begin the wire, and press <E> to end it.
2. Select **REPEAT** from the main menu and observe the wire that **Draft** places on pin 3 of the 22V10 PAL. If you usually use the mouse to select commands, try pressing <R> when you select the **REPEAT** command.
3. Select **REPEAT** six more times to place the remaining wires of this length shown in figure 6-2.
4. Place a single horizontal wire along the bottom of these wires as shown in figure 6-2.
5. Select **PLACE Junction**, and then **Place** to put a junction at the leftmost intersection of the wires placed in the prior steps. Press <Esc>.
6. Press <R> seven times to place wire junctions at each of the other wire intersections.

It takes longer to describe how to use the **REPEAT** command than to use it. It's a good idea to plan your schematics to take advantage of **REPEAT**.

Place the remaining parts of the minutes circuit

You have some more wires and junctions, and the power and ground symbols to place before you are done with this portion of the circuit. Because we intend to copy this circuit, it doesn't make sense to edit part labels or comment text yet. Follow these steps to finish placing objects:

1. Select **PLACE Power Place** to put power symbols above the resistor and capacitor symbols, as shown in figure 6-2.
2. Select **GET**, then enter **GND**.
3. Place ground symbols below the capacitor, and below and to the left of the switch symbol.
4. Place the remaining wires shown in figure 6-2.
5. Place junctions at the remaining locations shown in figure 6-2.

6. Examine your worksheet and carefully compare it with the schematic in figure 6-2. The exact position of objects is not as important as the presence or absence of these objects.
7. Correct any problems you find before going to the next exercise.

Copying a block

So far in this chapter, we have been careful to capture only the portions of the schematic that are repeated in several areas. Because three portions of the schematic share common areas, there should be approximately a three-to-one time saving when you copy the circuit.

Save a schematic block

Follow these steps to save a block:

1. Select **ZOOM Out** twice or **ZOOM Select 5** to change to the five-to-one scale. This way you can see all of the objects you are working with.
2. Select **BLOCK Save. Draft** displays this command line:

```
Begin Find Jump Zoom
```

3. Move the pointer above and left of the minutes circuit, and select **Begin**.
4. Move the pointer so the rectangle encloses the minutes circuit.
5. Select **End. Draft** saves the enclosed area in memory and returns to the main menu level.

Copy a circuit

Follow these steps to retrieve and place a copy of the minutes circuit:

1. Select **BLOCK Get**. An outline of the minutes circuit and the **Place** command line displays:

```
Place Find Jump Zoom
```

2. Look at the Y coordinate on the screen. Carefully move the copy to the right of the original, keeping the copy at the same Y coordinate. When the block is positioned correctly, select **Place**.

△ ***NOTE:*** *Be sure the copy is horizontally aligned with the original and that there is enough space between the two to allow more wires to be placed.*

3. After you place the copy of the circuit, the outline displays again so you can continue placing copies.
4. Next, place a copy of the circuit to the left of the original. Again, be sure that the copy is at the same Y coordinate as the original.

It's been a while since you had a look at the schematic you're duplicating. Figure 6-3 is another copy of the logic and display circuit schematic. Your worksheet should show parts placed similarly to the figure 6-3.

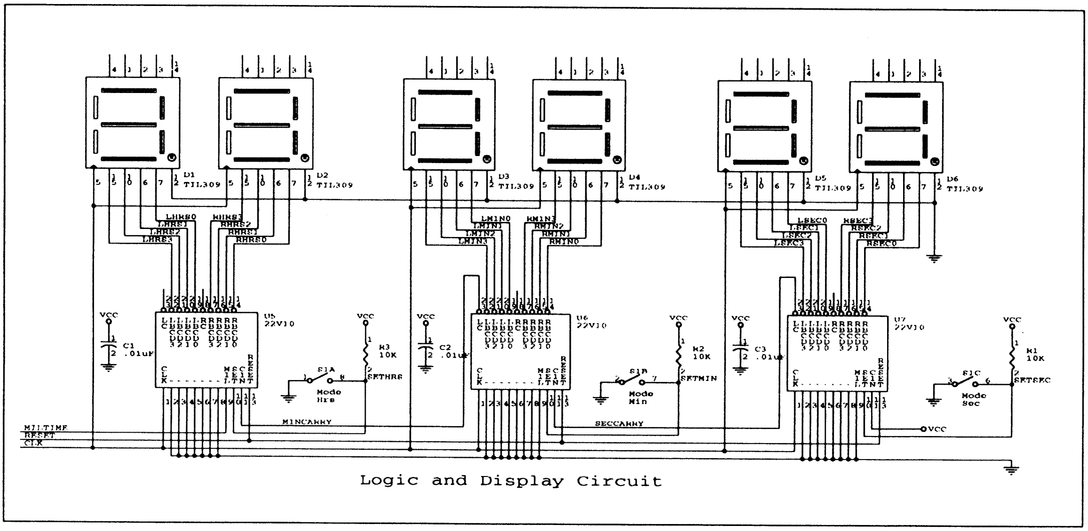


Figure 6-3. The logic and display circuitry.

Finishing the wiring

Figure 6-3 on the next page shows how the clock logic will look once you place the wires to connect the seconds, minutes, and hours circuits together. The following sections describe how to do this. As you follow the steps in each of these sections, refer to the callouts—such as these: ①②③—in each figure. These callouts correspond to the numbered steps in each section.

Wire the seconds circuit

The following steps correspond to the callouts in figure 6-4. Before you begin, move the pointer to the rightmost 22V10, and select **ZOOM Select 1.**

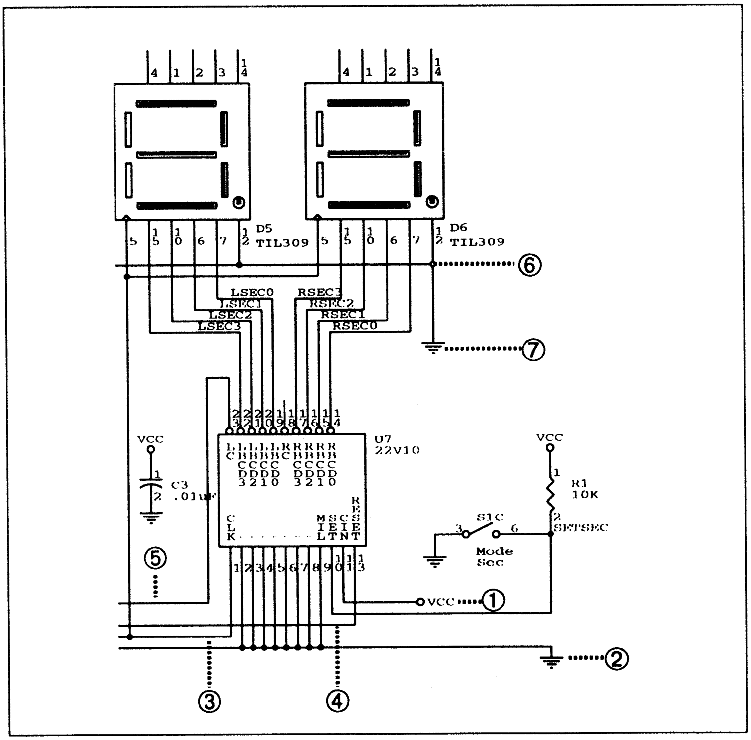


Figure 6-4. Seconds area of the clock logic. The callouts refer to the step numbers in this section.

1. The horizontal wire from the 22V10's pin 11 should be shortened. Referring to the ① in figure 6-4, delete the wire and redraw it so that it is only six or seven grid spaces long.

   Once the wire is the correct length, select **PLACE Power** to get a power symbol. This symbol must be turned before it is placed on the schematic, so select **Orientation Right** and then place it at the end of the wire you just drew.

2. The bottom horizontal wire must have a **GND** symbol added to it.

   Place a wire that extends two grid spaces down from the end of this wire. Get a **GND** symbol from the library .\DCLOCK.LIB and place it at the end of this wire.

   Place a wire at the left end of this wire to connect it to the minutes circuit.

3. The second-from-bottom horizontal wire needs to be shortened so that it doesn't run as far to the right. Delete and redraw this wire so that it starts at the end of the wire connecting to the 22V10's pin 1 and goes left to connect to the minutes circuit. The junction at the end of the pin 1 wire is no longer needed. Delete it.

4. The third-from-bottom horizontal wire needs to be shortened so that it stops at the wire that connects to the 22V10's pin 13. Delete this wire and place it so that it starts at the end of the wire connecting to the 22V10's pin 13 and goes left to connect to the minutes circuit. Since the junction at the end of the pin 13 wire is no longer needed, delete it.

5. Connect the wire that comes from the 22V10's pin 23 to the minutes logic.

6. Delete the horizontal wire that is immediately below the seconds display and place it so that it starts at the wire that comes from the rightmost TIL309's pin 12 and goes left to connect to the minutes circuit.

7. Extend the wire that comes from the rightmost TIL309's pin 12. Get a **GND** symbol from the .\DCLOCK.LIB library and place it at the end of this wire.

The seconds circuit is now complete and connected to the minutes circuit. Next you complete the minutes circuit.

Wire the minutes circuit

Before working on the minutes circuit, move the pointer to the middle 22V10 and select **ZOOM Center**. Then follow the steps below to complete the wiring for the minutes circuit (figure 6-5).

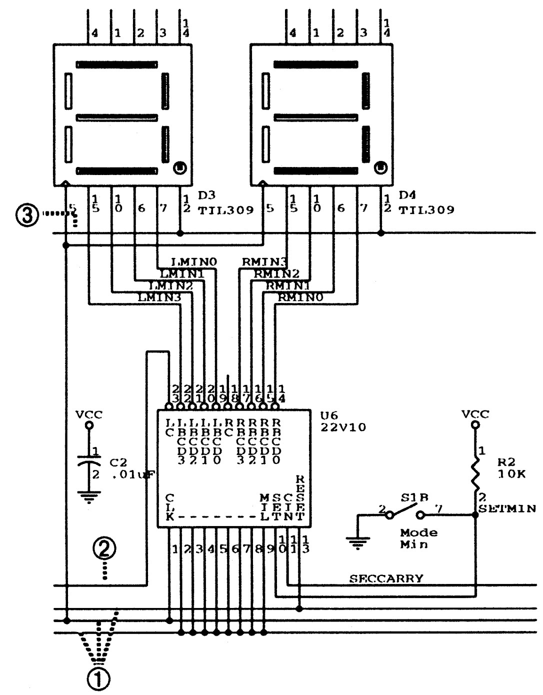


Figure 6-5. Minutes area of the clock logic. The callouts refer to the step numbers in this section.

1. Connect the bottom three horizontal wires to the hours logic.
2. Connect the wire from the 22V10's pin 23 to the hours logic.
3. Connect the horizontal wire that runs just below the minutes display to the hours logic.

The minutes circuit is now complete and connected to the hours circuit. Next you complete the hours circuit.

Wire the hours circuit

Before working on the hours circuit, move the pointer to the leftmost 22V10 and select **ZOOM Center**. Then follow the steps below to complete the wiring for the hours circuit (figure 6-6).

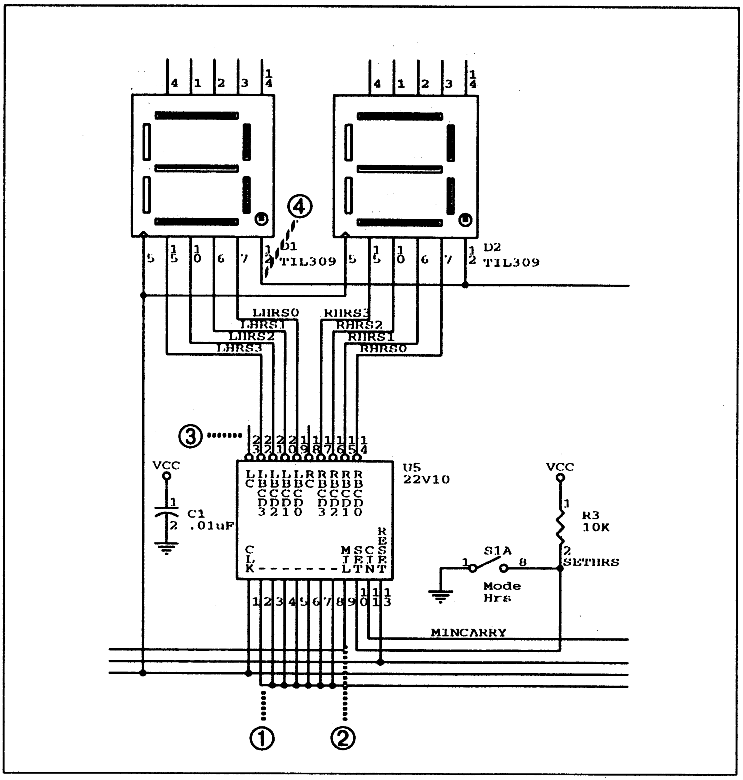


Figure 6-6. Hours area of the clock logic. The callouts refer to the step numbers in this section.

1. The bottom horizontal wire should end at the wire that extends from pin 2 of the 22V10.

   Delete this wire and place it so that it ends at the wire from pin 2 of the 22V10. Delete the junction at the end of the pin 2 wire also.

2. The vertical wire from pin 9 of the 22V10 should change so that it doesn't connect to the bottom horizontal wire. Delete this wire and its junction.

   Place the wire again so it comes down from pin 9, turns left, and goes as far to the left as the other wires.

3. Delete the wire that comes from pin 23 of the 22V10. Be sure to delete all segments of this wire.

4. The horizontal wire that runs just below the hours display should stop at the wire that extends from pin 12 of the leftmost numeric display. Remove the portion of this wire that is to the left of pin 12. Delete the junction at the end of the pin 12 wire also.

View clock logic

You have now connected all of the wires in the logic and display portion of the schematic. Select **ZOOM Select 5** to view the entire schematic. It should look like figure 6-3. Note that the clock oscillator circuit and the power regulator circuit on your schematic do not show in figure 6-3.

Figure 6-7 on the next page shows how the schematic will look when you complete the remaining steps in this chapter.

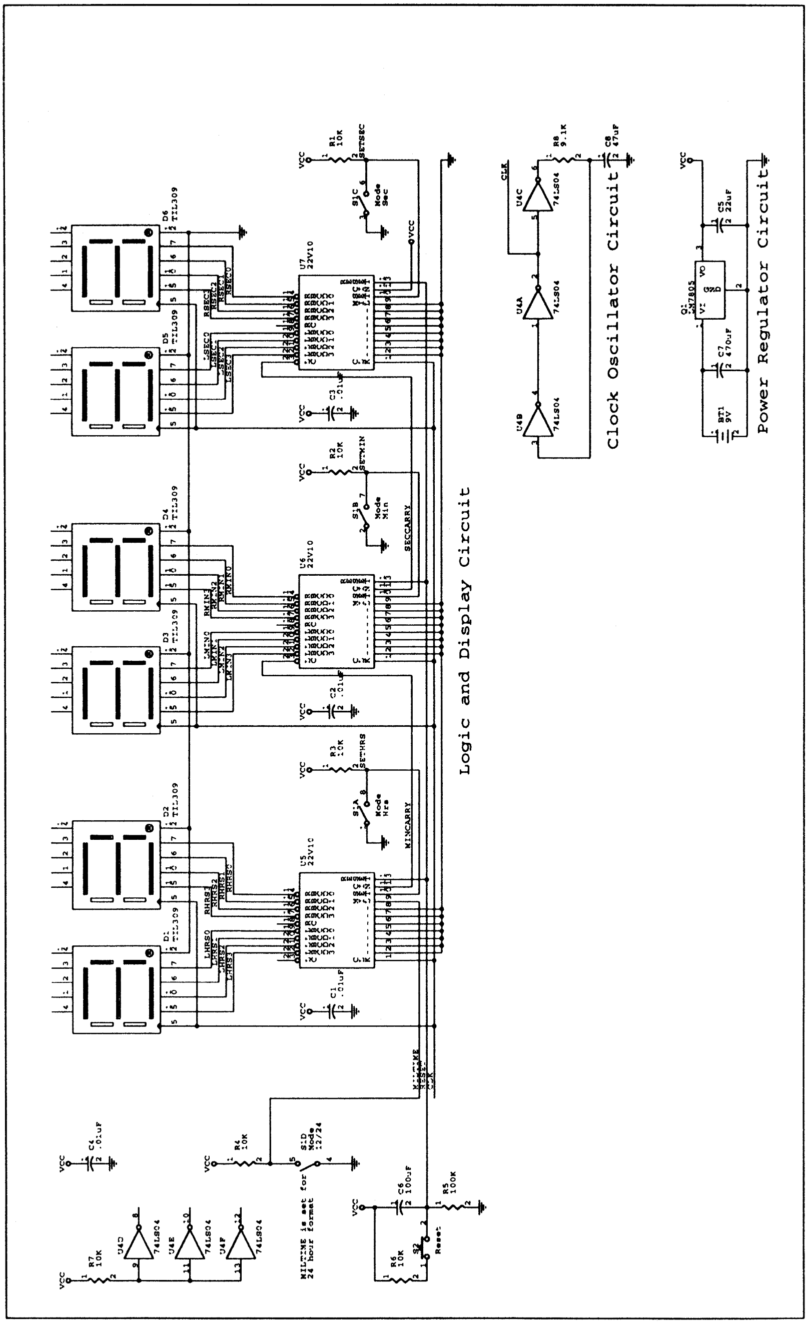


Figure 6-7. Completed TUTOR.SCH schematic.

Finishing the clock schematic

Compare the schematic in figure 6-7 with the schematic you have captured so far. Notice that you only need to add a few parts and place a few more wires to have a functional circuit. You also need to edit the labels and other text in the schematic. The following sections describe how.

Place the remaining schematic parts

There are four resistors, three inverters, two capacitors, two switches, and several power and ground symbols needed to complete the logic and display circuit schematic. Figure 6-8 shows the location of these parts.

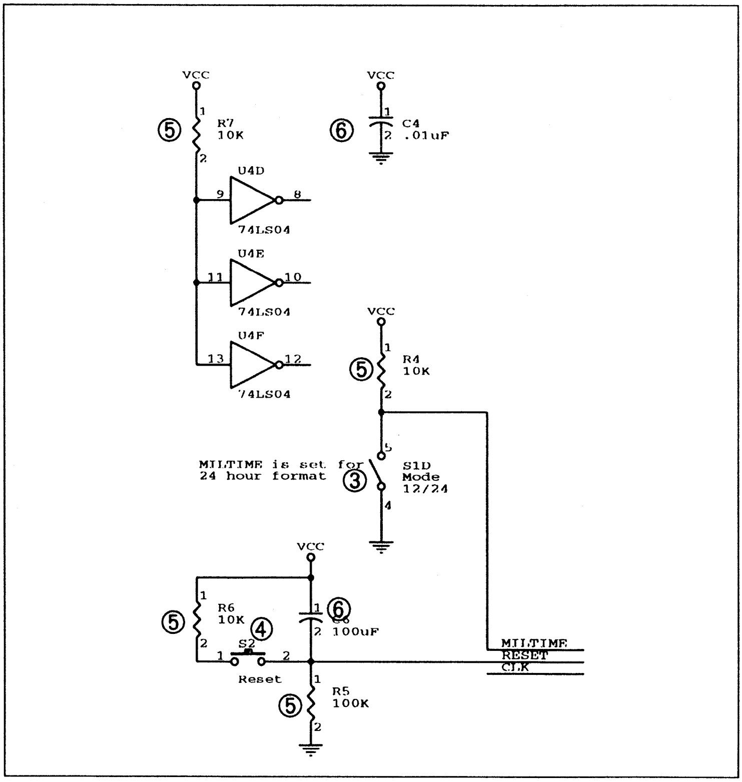


Figure 6-8. Switches, resistors, and capacitors to be placed. The callouts refer to the step numbers in this section.

Follow these steps to place the last parts on the schematic:

1. Select **ZOOM In,** or **Zoom Select 2** to change the scale to two-to-one.

   Change your view of the worksheet so grid reference C-7 is visible.

2. Get the **4SW SPST** switch from .\DCLOCK.LIB.

   The orientation of the 4SW SPST is not correct for this schematic.

3. With the part selected and the outline showing on the screen, select **Rotate** to turn the part so the orientation matches that shown in figure 6-8. Move the part to location (2.70, 6.00) and place it.

4. Now get the **SW PUSHBUTTON** part from .\DCLOCK.LIB and place it at location (1.20, 7.70).

5. Next get a resistor **(R)** and place four copies at locations (.50, 2.70); (2.70, 4.80); (.50, 7.20); and (2.20, 8.20).

6. Finally, select a capacitor and place it in locations (2.70, 2.60) and (2.20, 7.20).

Place the extra parts

There are also some leftover parts (from multiple-element parts) to be placed on the schematic. Figure 6-9 shows the leftover parts for this design.

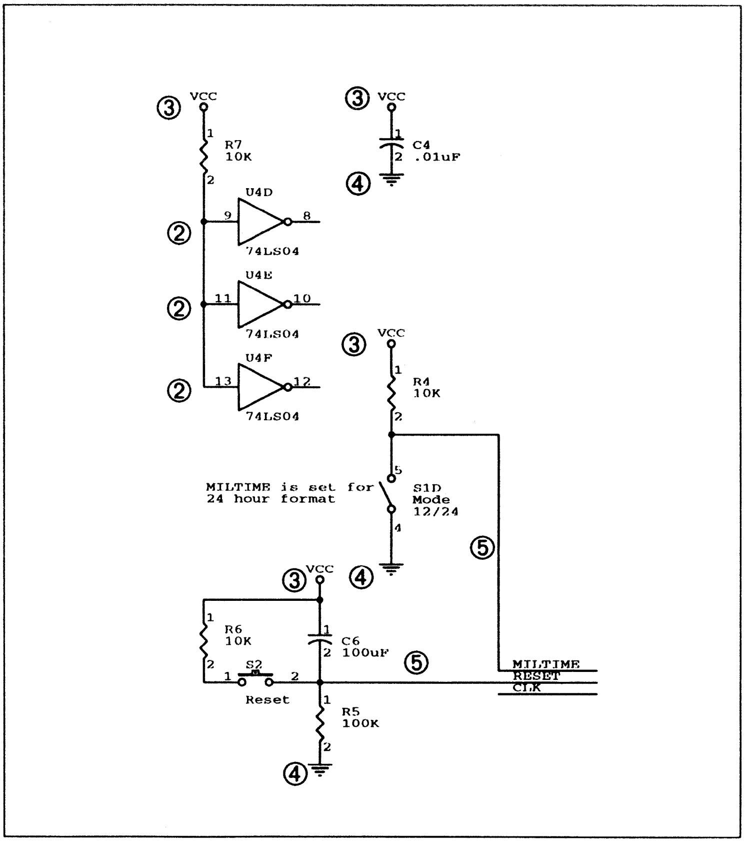


Figure 6-9. Inverters, power symbols, ground symbols, and wires to be placed. The callouts refer to the step numbers in this section.

Follow these steps to place the extra parts:

1. Use **ZOOM** to change the scale to one-to-one. Move to reference grid D-8.

2. Get the **74LS04** inverter from .\DCLOCK.LIB and place three copies at locations (1.00, 3.20); (1.00, 3.90); and (1.00, 4.60), as shown in figure 6-9.

3. Place four power symbols at locations (0.60, 2.60); (2.80, 2.50); (2.80, 4.70); and (2.30, 6.90).
4. Get a ground symbol from .\DCLOCK.LIB and place three copies at locations (2.70, 2.80); (2.70, 6.50); and (2.20, 8.60).
5. Place wires to connect the remaining parts, as shown in figure 6-9. Be sure to connect the wires to the logic and display circuit at the two places shown in figure 6-9.
6. Inspect the wire intersections and use the **Place Junction** command to add junctions where required, as shown in figure 6-8.

Editing remaining text

Now you assign and edit labels, edit part values, and add comment text to complete the digital clock schematic.

Edit the part values

Follow these steps to edit the part values:

1. Put the pointer on the capacitor located at (2.20, 7.20).
2. Select **EDIT Edit**. The **Edit Part** menu displays.
3. Select **Part Value Name**, and change the default value **CAP** to **100uF**.
4. Using the same procedure as in steps 2 and 3, assign the values shown in figure 6-7 to the all parts on the schematic. Table 6-1 gives a list of the new values to edit (you have already edited the first item in this table). Notice that some of the parts require that you enter information into **1st Part field**.

Part	*Approximate Location*	*Old Part Value Name*	*New Part Value Name*	*New 1st Part field*
Capacitor	2.20, 7.20	CAP	100uf	
Capacitor	2.80, 2.60	CAP	.01uf	
Resistor	.60, 2.80	R	10K	
Resistor	.60, 7.30	R	10K	
Resistor	2.30, 8.30	R	100K	
Resistor	2.80, 4.80	R	10K	
Switch	2.80, 6.10	4SW SPST	Mode	12\24
Switch	1.40, 7.80	SW PUSH-BUTTON	Reset	
Capacitor	4.90, 6.40	CAP	.01uf	
Capacitor	9.90, 6.40	CAP	.01uf	
Capacitor	15.30, 6.40	CAP	.01uf	
Resistor	8.70, 6.40	R	10K	
Resistor	13.80, 6.40	R	10K	
Resistor	19.20, 6.40	R	10K	
Switch	7.40, 6.90	4SW SPST	Mode	Hrs
Switch	13.10, 6.90	4SW SPST	Mode	Min
Switch	18.30, 6.90	4SW SPST	Mode	Sec

Table 6-1. Part value fields to edit.

Add labels to the wires

Follow these steps to add labels to the wires:

1. Select **PLACE Label** from the main menu. "Label?" displays.
2. Enter **CLK**. This label corresponds to the CLK label you assigned to a wire in the clock oscillator circuit schematic in chapter 3.
3. Move the pointer to the end of the unconnected wire at the left side of the logic and display circuit, and place the CLK label. Remember, when placing a label on a wire, the leftmost point of the label name must be placed next to the wire.

   The clock signal from the clock oscillator circuit is now logically connected to the wire to which you attached the CLK label. (see figure 6-7).
4. The "Label?" prompt returns each time you place a label. Label the following wires: MILTIME, RESET, SETHRS, MINCARRY, SETMIN, SECCARRY, and SETSEC. Refer to figure 6-7 for the location of these wires. Press <Esc> to dismiss the "Label?" prompt.

You still need to add labels to the wires between the 22V10s and the seven segment display parts. You could continue placing labels as with the previous steps, but **Draft** allows you to take a shortcut when labeling repeated text.

Set repeat text parameters

Follow these steps to set the parameters for quickly adding labels to the wires between the 22V10s and the LEDs:

1. Move the pointer to grid reference C-6. You want to look at the area where the labels will be placed.
2. Select **SET Repeat Parameters.** The menu shown below displays.
3. Set **X Repeat Step** to **+2.**
4. Set **Y Repeat Step** to **-1** (equal to the wire spacing).
5. Set **Label Repeat Delta** to **-1.**

```
Set Repeat Parameters
X Repeat Step           +0
Y Repeat Step           +1
Label Repeat Delta      +1
Auto Increment Place    NO
```

NOTE: Depending on the spacing between wires, you may have to adjust the X and Y values. Try it and see what works for your worksheet.

These **Repeat Parameters** cause labels to be placed two grid spaces to the right and one space up, and cause the number in the text to be decremented by one count each time you run the **REPEAT** command.

Placing labels with repeat text

Follow these steps to quickly add labels to the wires between the 22V10s and the LEDs:

1. Select **PLACE Label** from the main menu. The prompt "Label?" displays.
2. Enter `LHRS3`.
3. Move the pointer to the bottom wire directly below the leftmost clock segment, and place the label as shown in figure 6-7.
4. Press <Esc> to dismiss the **Place** command line.
5. Select **REPEAT** three times.

   The labels **LHRS2, LHRS1,** and **LHRS0** should be placed in the proper relative locations on the worksheet.

6. If the labels are not in the proper location, **DELETE** the out-of-position labels, adjust the **Repeat Parameters** to correct the problem, and repeat steps 1 through 5 again.
7. See figure 6-7 and place labels for the remaining left displays (**LMIN***n* and **LSEC***n*) by repeating steps 1 through 5.

Place the remaining repeat labels

The labels for wires going to the right displays slant in a different direction than those of the left displays, but otherwise the placement procedure is unchanged. Follow these steps:

1. Select **SET Repeat Parameters.**
2. Set the **X Repeat Step** to **-2**, the **Y Repeat Step** to **-1** (again, these values may vary depending on your wire spacing), and the **Label Repeat Delta** to **+1**.
3. Select **PLACE Label**. The prompt "Label?"displays.
4. Enter `RHRS0`.
5. Move the pointer to the top wire for the right hours display, and place the label as shown in figure 6-7.
6. Press <Esc> to dismiss the **Place** command line.
7. Select **REPEAT** three times.
8. See figure 6-7 and place labels for the remaining right displays (**RMIN***n* and **RSEC***n*) by repeating steps 3 through 6.

Add comment text

Follow these steps to add comment text to the schematic:

1. Select **PLACE Text**. The "Text?" prompt displays.
2. Enter **Logic and Display Circuit.**
3. Select **Larger** from the **PLACE Text** menu to use a larger type size for the text. The image of the text becomes larger.
4. Center the text below the schematic diagram (at approximately 9.20, 8.40). Press <P> to place the text.
5. The "Text?" prompt redisplays. Enter **MILTIME is set for.**
6. Select **Smaller** from the **PLACE Text** menu until the text size is the same size as the part and wire labels.
7. See figure 6-7 and place the text to the left of the **12\24** switch, at approximately (.80, 6.10).
8. The "Text?" prompt redisplays Enter **24-hour format.**
9. See figure 6-7 and place the text under the text you placed in step 7.

Editing the title block

The title block is located in the lower-right corner of the worksheet. You use the title block to provide standard types of information on the schematic, such as a title for the sheet, date, and reference number.

Jump to the title block

You can use the mouse to move the pointer to the title block region of the worksheet, or move there quickly by using the **JUMP** command. Follow these steps:

1. Select **JUMP Reference**. The **JUMP to Reference** menu displays.
2. Select **A**, and then **1**. The pointer jumps to region A-1 of the worksheet, and the title block is in view.

   Notice that the title block contains the information entered in chapter 2.

Edit the title block

Follow these steps to add or change information in the title block:

1. Select **EDIT**. The **EDIT** command line displays.
2. Put the pointer somewhere within the title block. Select **Edit**. The **Edit title block** menu displays.
3. Select one of the types of information listed in the menu. For example, Select **Organization name**. The "Organization name?" prompt displays.

```
Edit title block
Revision code
Title of sheet
Document number
Sheet number
Number of sheets
Organization name
1st Address line
2nd Address line
3rd Address line
4th Address line
```

4. Since you already entered the name of your organization in chapter 2, you can either leave it as it is, or you can delete the name and enter a new name.

   Once you press <Enter>, **Draft** stores the information and redisplays the **Edit title block** menu so you can specify other types of information.

△ *NOTE: Once you change a field in the title block, the information entered in the* ***Worksheet Options*** *area of the* ***Configure Schematic Tools*** *screen is no longer used for the changed fields.*

5. Following the procedures in steps 3 and 4, fill in or change other title block information. Filling in the boxes is optional for this tutorial.
6. When you are done, press <Esc>. The title block displays the information you entered.

Updating the file

The digital clock design schematic is now complete. Save your work and exit by selecting **QUIT**, then **Update file**, then **Abandon Edits**. The **Schematic Design Tools** screen displays.

Summary

In the past five chapters, you learned several ways to quickly create circuits using **Draft**. In the next chapter, you learn to use some of the other tools.

CHAPTER 7

Using other Schematic Design Tools

In this chapter you learn how to use some of the other **Schematic Design Tools**. These tools are normally used after the schematic is complete. The tools covered in this chapter are:

Annotate Schematic	Automatically updates part reference designators. It also updates the pin numbers associated with the reference designators in multiple-element parts.
Create Netlist	Creates a netlist and general wire list in one of over thirty standard formats.
Back Annotate	Updates part reference designators by using a list of old and new reference designators.
Create Bill of Materials	Creates a summary list, sorted by reference designator, of all the parts used.
Check Electrical Rules	Checks for conformity to basic electrical rules.
Plot Schematic	Plots a single schematic or all the schematics in a design. This tool supports scaling.

Housekeeping

Before proceeding with the tutorial, you should perform a few housekeeping tasks. You have completed quite a bit of work up to this point, so it's a good idea to back up your design files.

In case you were unable to successfully complete the exercises in chapters 1-6, OrCAD has provided copies of the TUTOR schematic and library files. These files are called TUTOR2.SCH and .\DCLOCK2.LIB. By using these files you can perform the remaining exercises with predictable results. Once you back up your design, you copy these two files to TUTOR.SCH and .\DCLOCK.LIB.

Backup Design

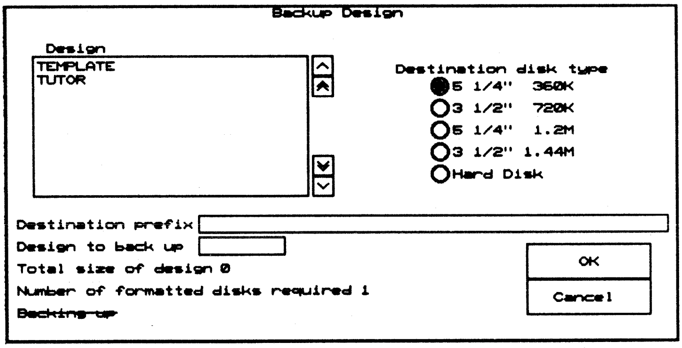
Use the **Backup Design** tool to back up onto floppy disks or to another part of your hard disk all the files belonging to a design . To conserve disk space, back-up files are stored in a condensed format. To restore the files to their normal format, use the **Restore Design** tool, which is described in the *OrCAD/ESP Design Environment User's Guide.*

Follow these steps to back up a design:

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button. The **Design Management Tools** menu displays.

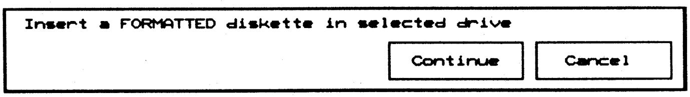
```
Design Management Tools
Design Management Tools
Suspend to System
Show Hot Key Assignments
Exit
```

2. Select **Design Management Tools**. The **Design Management Tools** screen displays.
3. Click the **Backup Design** button. The screen shown on the next page displays.



Backup Design screen.

4. Select the TUTOR design in the **Design** list box.
5. Move the pointer to the **Destination prefix** entry box and press <Enter>.
6. Enter the path to use for the backup. To back up the design on a floppy disk, enter the destination prefix `A:`. The message shown below displays.



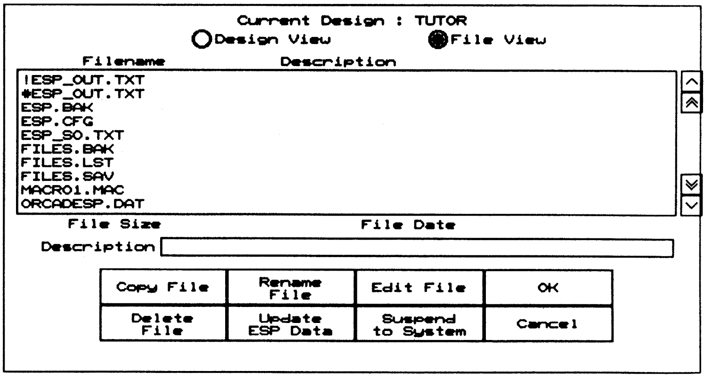
7. Insert a properly formatted disk in drive A and select **Continue**. Select **Cancel** if you want to cancel the backup for the time being.
8. Click the **OK** button. The design environment makes a back-up copy of the selected design onto the disk or into the directory specified.

   Once the design is backed up, the message "Backup successfully completed" displays along with an **OK** button.
9. Click this **OK** button and then click the **Cancel** button. The **Design Management Tools** screen redisplays.

Rename files

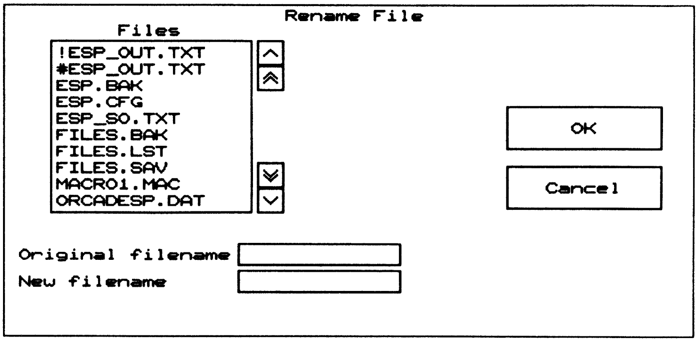
Follow these steps to rename the TUTOR2.SCH and .\DCLOCK2.LIB files to TUTOR.SCH and .\DCLOCK.LIB:

1. The **Design Management Tools** screen should still be displayed. Click the **File View** button at the top of this screen. The screen shown below displays.



File View screen.

2. Click the **Rename File** button. The screen shown below displays.



Rename File screen.

3. Select the TUTOR2.SCH file from the **Files** list box. You will have to scroll the list to see this filename.

4. Move the pointer to the **New filename** entry box and press <Enter>.
5. Enter the new name for the file, `TUTOR.SCH`.
6. Click the **OK** button to rename the file.
7. Select the DCLOCK2.LIB file from the **Files** list box.
8. Move the pointer to the **New filename** entry box and press <Enter>.
9. Enter the new name for the file, `.\DCLOCK.LIB`, and then click the **OK** button.
10. Click the **Cancel** button to return to the **File View** screen. Click the **Cancel** button again to return to the **Schematic Design Tools** screen.

Now that your files are backed up and you have renamed TUTOR2.SCH and .\DCLOCK2.LIB to TUTOR.SCH and .\DCLOCK.LIB, you are ready to continue learning about **Schematic Design Tools**.

Running Annotate Schematic

Annotate Schematic scans schematic designs and automatically updates part reference designators. It also updates the pin numbers associated with the reference designators in multiple-element parts. **Annotate Schematic** updates reference designators in the order in which they were placed on the worksheet.

When you first place a part, a default reference designator value displays above the part, such as **U?** or **U?A**. **Annotate Schematic** changes the default values to unique values for each part in a specified design.

For example, suppose the specified design contains three occurrences of the same part, and this particular part is a two-element part. **Annotate Schematic** assigns unique values such as U1A, U1B and U2A. When the parts are placed, parts U1A and U1B are referenced from the two-element part identified as "U1." The "A" and "B," the suffix letters, designate the unique identity of each part and its "slot" in the two-element part. The U2A part is referenced from a second two-element part identified as "U2."

You should run **Annotate Schematic** before running the other tools. Other tools report information about the worksheet file, and if you run **Annotate Schematic** first, you ensure that reported information includes the updated reference designators.

Annotate Schematic modifies the worksheet file; but it also creates a back-up file containing the original worksheet file.

You can also assign values of your choice using **Draft's EDIT** command, but assigning values using **Annotate Schematic** guarantees unique values. Unique reference designator values are necessary for some other processes, such as producing a netlist.

Run Annotate Schematic on TUTOR.SCH

Follow these steps to configure and run **Annotate Schematic**:

1. Select **Annotate Schematic** from the **Schematic Design Tools** screen. The menu at right displays.

```
Annotate Schematic
Execute
Local Configuration
Assign Hot Key
Show Version
Configure Schematic Tools
Help
```

2. Select **Local Configuration** and then select **Configure ANNOTATE**. The **Configure Annotate Schematic** screen displays (figure 7-1).

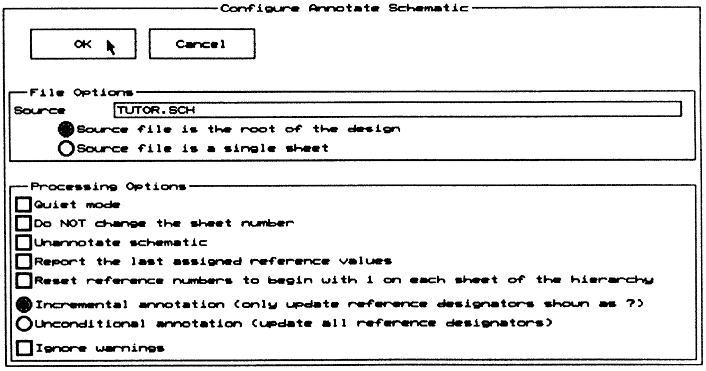


*Figure 7-1. **Configure Annotate Schematic** screen.*

Notice that the **Source** entry box contains the filename TUTOR.SCH. The design environment automatically sets the source to the design name and the default worksheet file extension found in the Worksheet Options area of the **Configure Schematic Tools** screen.

3. Now, click the **Source file is a single sheet** button.
4. Click the **OK** button.
5. Select **Annotate Schematic** and then **Execute**.

As it processes, **Annotate Schematic** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

```
Annotation"    V4.00 01-DEC-90"
(C) Copyright 1985,1986,1987,1988,1989,1990 OrCAD L.P      ALL RIGHTS RESERVED.
Loading  "\ORCADESP\SDT\LIBRARY\DCLOCK.LIB"
```

Status messages display at the bottom of the ***Schematic Design Tools*** *screen.*

When **Annotate Schematic** is finished, the monitor box disappears and the full **Schematic Design Tools** screen displays.

△ *NOTE: When you run* ***Annotate Schematic*** *on a multiple-sheet design, click the* ***Source file is the root of the design*** *button.*

6. Run **Draft** and examine the TUTOR.SCH worksheet. Note the reference designators and pin numbers. Your reference designators may be slightly different than those shown in this tutorial. This is because **Annotate Schematic** assigns reference designators to parts in the order in which you placed them on the worksheet.

Notice the updated reference designators on the 74LS04 inverters. The U?A changed to U2A, U2B, and U2C. The inverters are all from the same multiple-element part. Also notice that the inverters' pin numbers changed.

Running Check Electrical Rules

Check Electrical Rules performs a general electrical rules check. It issues warnings for unused inputs on parts, unlabeled wires connected to a bus, and invalid connections.

△ *NOTE: Always use* ***Check Electrical Rules*** *on your designs before going on to* ***Digital Simulation Tools*** *or* ***PC Board Layout Tools****. If any errors are reported, correct them before trying to simulate the design or the simulation results will be inaccurate.*

Follow these steps to configure and run **Check Electrical Rules**:

1. Select **Check Electrical Rules Local Configuration Configure ERC**. The **Configure Check Electrical Rules** screen displays.

   Notice that the **Source** entry box contains the filename TUTOR.SCH.

   Notice that the **Destination** entry box contains the filename TUTOR.ERC. **Check Electrical Rules** stores the report it creates in a text file with this filename.

   You can also specify a path to another directory and another file, but for the purposes of this exercise, you really should place the report in TUTOR.ERC.

2. Click the OK button.
3. Select **Check Electrical Rules** and then **Execute**.

   As it processes, **Check Electrical Rules** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

4. After **Check Electrical Rules** is finished running, a message box displays. You may click the **View Output** button to display the report, or click the **OK** button to dismiss the message box. Click the **OK** button.
5. Examine the file TUTOR.ERC with **Edit File**. The contents of the file should be similar to figure 7-2.

```
Time Stamp - 14-FEB-1990    9:43:18

"SHEET\TUTOR.SCH"

LABEL REPORT
 (power) VCC
 (power) GND
     .
    .
    .
 MILTIME
 RESET
 CLK

UNCONNECTED REPORT
 X=  1.90,Y=  1.20 Output          U1D,O
 X=  1.90,Y=  1.80 Output          U1E,O
    .
    .
    .
 X= 11.10,Y=  5.70 I/O             U3,RC
 X= 16.30,Y=  5.70 I/O             U4,RC

Check Electrical Rules Report

Digital Clock Schematic
Revised: February 14, 1990
Revision:
```

Figure 7-2. The TUTOR.ERC file.

The **UNCONNECTED REPORT** portion shows some pins are unconnected. For example, consider the statement below:

```
X= 3.40 Y= 2.00 Output  U1,RCO
```

This means there is an unconnected signal at location (3.40, 2.00). It is further identified by the reference designator of the part on which it is found, U1, and by its pin name, RCO. Since the reported pins were intentionally not connected, you can ignore this information. If desired, you can examine the schematic and locate these pins.

View errors

Now use **Draft** to view the schematic. Notice a circle at each location where an error is reported by **Check Electrical Rules**. These are *error markers*. To display the associated error message, place the pointer in the center of an error marker and select **INQUIRE** from the main menu. The error message displays at the top of the screen. Repeatedly selecting **INQUIRE** cycles through all of the error messages for a particular error marker.

Select **QUIT Abandon Edits** when you are done looking at the schematic.

△ *NOTE: If you save the schematic file by selecting* ***QUIT Update File****, the error markers are erased.*

Running the Create Netlist tool

The **Create Netlist** tool creates a connectivity database, or netlist, in a number of possible formats.

To create a proper netlist, you must carefully deal with labels, module ports, and power objects. The general guidelines are:

- Place labels in the correct format on all buses.
- Place labels in the correct format on all signals connecting to a bus.
- Place module ports in the correct format on all signals going off the worksheet.
- Don't put blank spaces in labels or between prefixes and suffixes in bus and module port names.
- Do not overlap wires or buses with other wires, buses, or object pins.

For a more detailed discussion of these guidelines, see *Chapter 3: Guidelines for creating designs* in the *Schematic Design Tools Reference Guide.*

Create a netlist in WIRELIST format

Follow these steps to create a netlist in WIRELIST format:

1. Select **Create Netlist** and then **Local Configuration** to configure **Create Netlist**.

   The menu shown at right displays. This menu has three processes to configure: INET, ILINK, and IFORM. INET is the compiler. ILINK is the connectivity linker, and IFORM is the netlist formatter. Each of these processes must be turned on to create a netlist. For more information on each of these processes, see the *Schematic Design Tools Reference Guide.*

   ```
   Select Configuration
   Configure INET
   Configure ILINK
   Configure IFORM
   INET on
   ILINK on
   IFORM on
   ```

2. Select **Configure INET**. The **Configure Incremental Netlist** screen displays (figure 7-3).

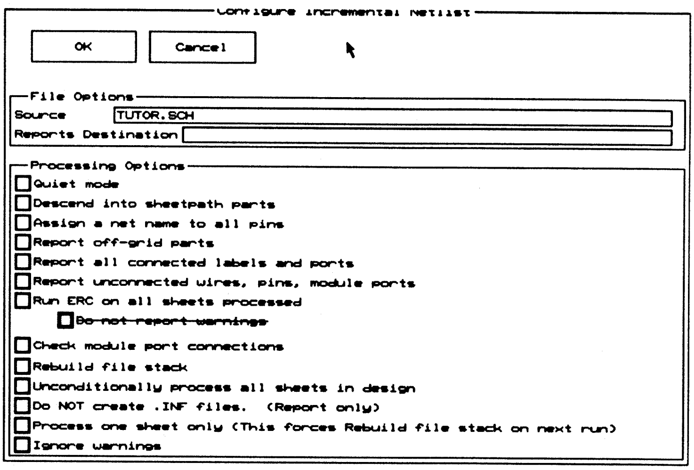


Figure 7-3. ***Configure Incremental Netlist screen.***

3. In the **File Options** area of the screen, check to be sure that the **Source** entry box contains the filename `TUTOR.SCH`. The design environment automatically sets the source to the design name and the default worksheet file extension found in the Worksheet Options area of the **Configure Schematic Tools** screen.
4. Click the **Cancel** button to leave the configuration screen without making any changes. The **Schematic Design Tools** screen displays.
5. Now display **ILINK's** local configuration screen **Configure Netlist Linker.** Notice that the **Source** entry box contains the filename TUTOR.INF. Click the **Cancel** button.
6. Now display **IFORM's** local configuration screen **Configure Netlist Format. IFORM** is the netlist formatter that converts the connectivity database that has been linked by **ILINK** into the format specified in this configuration.

   The **Source** entry box should contain the name TUTOR, showing that you will format the TUTOR files created by **ILINK.**

   The **Destination 1** entry box should contain the filename TUTOR.NET.

7. The **Format prefix/wildcard** should contain the following pathname and filter:

   ```
   Format prefix/wildcard
   C:\ORCADESP\SDT\NETFORMS\*.CCF
   ```

   The **Netlist format** list box contains a number of files. Edit the **Format prefix/wildcard** entry box by inserting a "W" before the *, so that it becomes:

   ```
   Format Prefix/Wildcard
   C:\ORCADESP\SDT\NETFORMS\W*.CCF
   ```

   The list box now contains far fewer filenames. Select WIRELIST.CCF. The selected netlist format filename displays in the **Selected format** entry box:

   ```
   Selected format: WIRELIST.CCF
   ```

8. Click the **OK** button to save all of the configurations.
9. Click the **Create Netlist** button and then select **Execute** to run **Create Netlist**.

   As it processes, **Create Netlist** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

10. Using **Edit File,** look at the file generated by **Create Netlist** TUTOR.OUT. It should look like the wirelist-format netlist file shown Figure 7-4.

```
Wire List

Digital Clock Schematic                        Revised:  November 1, 1990
                                               Revision:

<<< Component List >>>
.01UF                              C5          .01UF
.01UF                              C6          .01UF
.01UF                              C7          .01UF
.01UF                              C8          .01UF
100K                               R4          100K
100UF                              C4          100UF
10K                                R2          10K
10K                                R3          10K
10K                                R5          10K
10K                                R6          10K
10K                                R7          10K
10K                                R8          10K
22UF                               C3          22UF
470UF                              C2          470UF
47UF                               C1          47UF
74LS04                             U1          74LS04
9.1K                               R1          9.1K
9V                                 BT1         9V
HRS                                U2          HRS
LM7805                             Q1          LM7805
MINSEC                             U3          MINSEC
MINSEC                             U4          MINSEC
MODE                               S1          MODE
RESET                              S2          RESET
TIL309                             D1          TIL309
TIL309                             D2          TIL309
TIL309                             D3          TIL309
TIL309                             D4          TIL309
TIL309                             D5          TIL309
TIL309                             D6          TIL309

<<< Wire List >>>

  NODE   REFERENCE     PIN #   PIN NAME       PIN TYPE         PART VALUE
[00001]  N00001
         R8              2        2           Passive          10K
         U1              9        I_D         Input            74LS04
         U1             11        I_E         Input            74LS04
         U1             13        I_F         Input            74LS04
[00002]  LHRS3
         D6             15        QAIN        Input            TIL309
         U2             22        LBCD3       BiDirectional    HRS
[00003]  LHRS2
         D6             10        QBIN        Input            TIL309
         U2             21        LBCD2       BiDirectional    HRS
[00004]  LHRS1
         D6              6        QCIN        Input            TIL309
         U2             20        LBCD1       BiDirectional    HRS
[00035]  CLK
         U1              4        O_B         Output           74LS04
         U1              5        I_C         Input            74LS04
         D5              5        STROBE      Input            TIL309
         D6              5        STROBE      Input            TIL309
         U2              1        CLK         Input            HRS
```

Figure 7-4. Wirelist-format netlist (continued on next page).

	D4	5	STROBE	Input	TIL309
	U3	1	CLK	Input	MINSEC
	D1	5	STROBE	Input	TIL309
	D2	5	STROBE	Input	TIL309
	U4	1	CLK	Input	MINSEC
[00040]	GND				
	C2	2	2	Passive	470UF
	BT1	2	2	Passive	9V
	Q1	2	GND	Input	LM7805
	C3	2	2	Passive	22UF
	C1	2	2	Passive	47UF
	U1	7	GND	Power	74LS04
	R4	2	2	Passive	100K
	U2	3	-	Input	HRS
	U3	2	-	Input	MINSEC
	U3	3	-	Input	MINSEC
	U3	4	-	Input	MINSEC
	U3	5	-	Input	MINSEC
	U3	6	-	Input	MINSEC
	U3	7	-	Input	MINSEC
	U3	8	-	Input	MINSEC
	U3	9	MIL	Input	MINSEC
	U4	2	-	Input	MINSEC
	U4	3	-	Input	MINSEC
	U4	4	-	Input	MINSEC
	U4	5	-	Input	MINSEC
	U4	6	-	Input	MINSEC
	U4	7	-	Input	MINSEC
	U4	8	-	Input	MINSEC
	U4	9	MIL	Input	MINSEC
	S1	4	1_D	Passive	MODE
	S1	3	1_C	Passive	MODE
	S1	2	1_B	Passive	MODE
	U4	12	GND	Power	MINSEC
	U3	12	GND	Power	MINSEC
	U2	12	GND	Power	HRS
	C7	2	2	Passive	.01UF
	C6	2	2	Passive	.01UF
	C5	2	2	Passive	.01UF
	S1	1	1_A	Passive	MODE
	D5	12	DPIN	Input	TIL309
	D6	12	DPIN	Input	TIL309
	D4	12	DPIN	Input	TIL309
	D3	12	DPIN	Input	TIL309
	D2	12	DPIN	Input	TIL309
	D1	12	DPIN	Input	TIL309
	D1	8	GND	Power	TIL309
	D2	8	GND	Power	TIL309
	D3	8	GND	Power	TIL309
	D4	8	GND	Power	TIL309
	D5	8	GND	Power	TIL309
	D6	8	GND	Power	TIL309
	C8	2	2	Passive	.01UF

Figure 7-4. Wirelist-format netlist (continued from previous page).

Running Back Annotate

If you want to change the reference designator values assigned by **Annotate Schematic** (or the values you manually assigned), you need not reopen the worksheet and edit the reference designators one by one.

Back Annotate lets you change as many reference designators as you want in a single operation. You create a text file containing the current and new values (called a WAS/IS file) and then run **Back Annotate**, specifying the worksheet name and the WAS/IS filename.

You can run **Back Annotate** on a single worksheet or on an entire design.

For example, consider the TUTOR.SCH worksheet. Currently, the six LED parts in TUTOR.SCH have reference designators of D1, D2, D3, and so on. Suppose you decide you want the values to be A1, A2, A3, and so on. In this example, you will run **Back Annotate** on the schematic, TUTOR.SCH.

Change reference designator values

Follow these steps to change the reference designator values.

1. Create a text file using **Edit File**. Click the **Edit File** button and then select **Execute.** The **Edit File** screen displays. Enter a filename in the **File to Edit** entry box. For the purposes of this exercise, name the file **TUTREF.**

   See the *ESP Design Environment User's Guide* for more information about the text editor that comes with ESP, or to learn how to configure ESP to use another text editor.

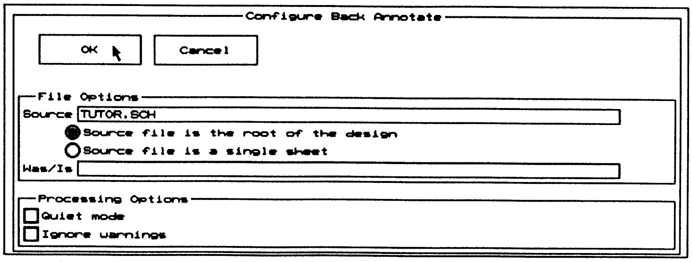
2. Include the information shown at right in the text file. Use <Tab> or blank spaces to separate the paired items.

```
D1  A1
D2  A2
D3  A3
D4  A4
D5  A5
D6  A6
```

3. Save the text file.

△ *NOTE: Be sure to save this file as text only. Any special formatting inserted by your text editor causes the* **Back Annotate** *tool to fail. In addition, some text editors may attach an extension to the TUTREF file. If it does, be sure to enter the extension when running* **Back Annotate***.*

4. Return to the **Schematic Design Tools** screen, click the **Back Annotate** button, and select **Local Configuration Configure BACKANNO.**



Configure Back Annotate screen.

5. Enter the name of the file where **Back Annotate** gets the back annotation information—in this case **TUTREF**—in the **Was/Is** entry box in the **File Options** area of the **Configure Back Annotate** screen.
6. Click the **OK** button. The **Schematic Design Tools** screen displays.
7. Run **Back Annotate**. As it processes, **Back Annotate** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   Back Annotate modifies the schematic file, TUTOR.SCH to reflect the new reference designator values found in the WAS/IS file, NEWREF.
8. Run **Draft** on TUTOR.SCH to confirm that **Back Annotate** modified the reference designators on the schematic.

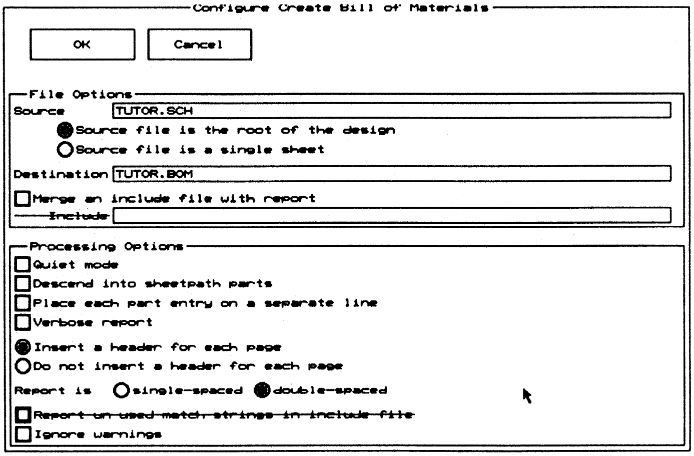
Running Create Bill of Materials

Create Bill of Materials creates a text file listing all parts in a single worksheet or an entire design.

Make a parts list

Follow these steps to make a parts list:

1. Click the **Create Bill of Materials** button, and then select **Local Configuration Configure PARTLIST.** The **Configure Create Bill of Materials** screen displays.



Configure Create Bill of Materials screen.

2. In the **File Options** area of the **Configure Create Bill of Materials** screen, there are two filenames:

   - In the **Source** entry box, the name of the worksheet from which the Bill of Materials is produced: `TUTOR.SCH`.

     This entry box tells **Create Bill of Materials** to use the worksheet file TUTOR.SCH to get the correct reference designator values.

   - In the **Destination** entry box, the name of the file where **Create Bill of Materials** stores the report: `TUTOR.BOM`.

3. Click the **OK** button to save all of the configurations. The **Schematic Design Tools** screen displays.

4. Click the **Create Bill of Materials** button and then select **Execute.**

   As it processes, **Create Bill of Materials** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   The contents of TUTOR.BOM are shown in the TUTOR design Bill of Materials figure below. Use **Edit File** to look at this file.

```
Digital Clock Schematic                              Revised:   November 1, 1990
                                                     Revision:
Bill Of Materials    November 1, 1990    16:17:16          Page  1

 Item  Quantity  Reference                 Part
______________________________________________________________________

    1       1     BT1                       9V
    2       1     C1                        47uF
    3       1     C2                        470uF
    4       1     C3                        22uF
    5       1     C4                        100uf
    6       4     C5,C6,C7,C8               01uf
    7       6     A1,A2,A3,A4,A5,A6         TIL309
    8       1     Q1                        LM7805
    9       1     R1                        9.1K
    10      6     R2,R3,R4,R5,R6,R7         10k
    11      1     S1                        Mode
    12      1     S2                        Reset
    13      1     U1                        74LS04
    14      3     U2,U3,U4                  Minsec
```

TUTOR design Bill of Materials.

Running Plot Schematic

The last task in this chapter of *Learning Schematic Design Tools* is to plot the design you have created so far.

Plot Schematic is used to send designs to a plotter, or optionally, to a printer using the **Send output to printer** button.

△ *NOTE: This section focuses on running the* ***Plot Schematic*** *tool and assumes you have configured* ***Schematic Design Tools*** *and connected your printer or plotter correctly. There are many variables affecting plotting. As with other mechanical processes, make sure your equipment, paper, pens, and so on, are in good working order and set up properly.*

Follow these steps to run **Plot Schematic** :

1. Click the **Plot Schematic** button and then select **Local Configuration Configure PLOTALL.** The **Configure Plot Schematic** screen displays.

   If you are using a printer instead of a plotter, click the **Send output to printer** button.

△ *NOTE: When plotting a multiple-sheet design,* ***Plot Schematic*** *plots every worksheet in the design.*

2. If the plot produced is too large or too small, you can change the scale. Display the **Configure Plot Schematic** screen. Click the **Automatically scale and set X, Y offsets for specified sheet size** button and select the **Set scale factor** option. The **Set Scale factor** entry box becomes highlighted.

   Enter the scale factor, expressed in the form n.nnn. For example, if the plot is larger than the paper, you might run **Plot Schematic** at half scale by entering the number: `0.500`.

3. Click the **OK** button.
4. Click the **Plot Schematic** button and then select **Execute** to run **Plot Schematic.**

Structuring your design

In this chapter you look at three types of design structures: a flat design, a simple hierarchy, and a complex hierarchy.

A flat design

A flat design is one in which all of the worksheets are linked together at the same level. Use a flat design for relatively small designs with no more than five to ten worksheets. Since you must manage all of the interconnections between the worksheets of a flat design by the names assigned to the module ports, large designs consisting of many worksheets, or repetitive logic, are more easily managed using hierarchical structures.

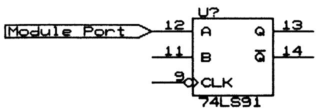
Flat designs are linked together using module ports and the | LINK command.

| LINK command

```
|LINK
|OTHER
```

You use the | LINK command on the root worksheet to inform the various schematic tools which worksheets are in a flat design. In figure 8-1, the | LINK command links the root worksheet, PROJECT.SCH, to OTHER.SCH. The filename of the root worksheet consists of the name of the design and a .SCH extension.

Module ports



Module ports are graphic objects that indicate where signals are conducted between worksheets. Module ports that have identical names are considered to be electrically connected. In figures 8-1 and 8-2, CLEAR, LOAD, and RCO are connected; Hi[0..3] and Lo[0..3] are not connected.

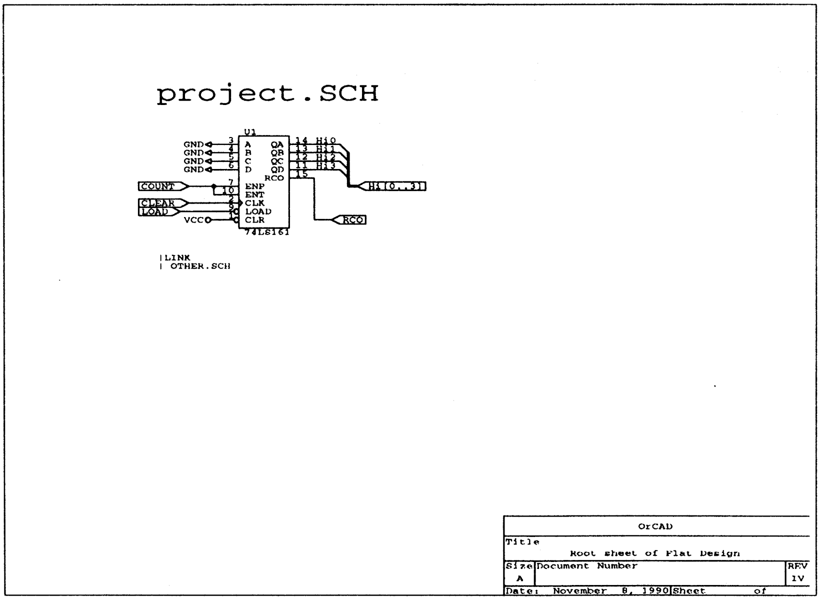


Figure 8-1. Root worksheet of flat design.

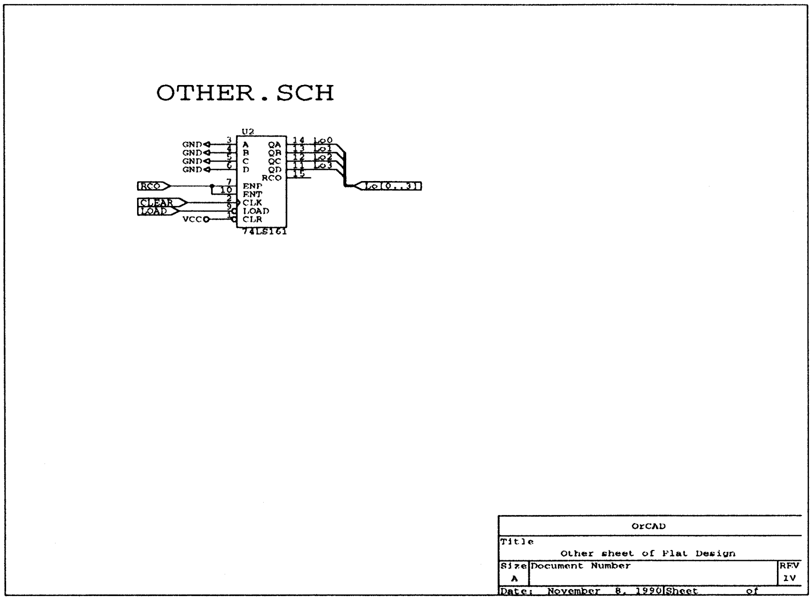


Figure 8-2. Other worksheet of flat design.

Creating new designs

The remainder of this chapter covers simple and complex hierarchies. Example files are included in the TUTOR design. Before you proceed through the rest of this chapter, you need to create two new design directories: CMOSCPU and 4BIT. You will copy the example files from the TUTOR design into these new design directories. Follow these steps:

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button. The **Design Management Tools** menu displays.
2. Select **Design Management Tools**. The **Design View** portion of the **Design Management Tools** screen displays.
3. Click the **Create Design** button. The **Create Design** screen displays. Make sure the **Copy all files** button is selected.
4. Enter **CMOSCPU** in the **New design name** entry box. Click the **OK** button. The prompt " Working . . ." and several messages display at the top left corner of the screen. After the new design name is created, the **Create Design** screen is dismissed and the design name, CMOSCPU, appears in the **Design** list box.
5. Repeat steps 3 and 4, but this time create a new design named 4BIT.
6. Select **TUTOR** from the **Design** list box. The files you need to copy are in the TUTOR design.
7. Click the **File View** button to see the TUTOR files.
8. Click the **Copy File** button. The **Copy File** screen displays.
9. Using the information in table 8-1, select the source file from the **Files** list box, enter the destination in the **Destination** entry box, and then click the **OK** button. Repeat this procedure for each file in the table.

Source file name	*Destination*
CMOSCPU.SCH	..\CMOSCPU\CMOSCPU.SCH
MEMORY.SCH	..\CMOSCPU\MEMORY.SCH
POWER.SCH	..\CMOSCPU\POWER.SCH
4BIT.SCH	..\4BIT\4BIT.SCH
FULLADD.SCH	..\4BIT\FULLADD.SCH
HALFADD.SCH	..\4BIT\HALFADD.SCH

Table 8-1. Files to be copied and their destinations.

10. When you have copied all the files to their appropriate destinations, click the **CANCEL** button. The **Copy File** screen is dismissed.
11. Click the **Design View** button. Reset the current design to CMOSCPU. Click the **OK** button. The main screen displays.

A simple hierarchical design

The layered arrangement created by placing worksheets inside other worksheets is called a hierarchical design. This section describes the structure of a simple, three worksheet hierarchical design.

Sheet symbols represent other worksheets in a hierarchical design. Each sheet symbol represents a subsheet. Sheet symbols may be placed at any level of the hierarchy.

The following example is a simple hierarchy because the two sheet symbols reference different worksheets. In complex hierarchies, any number of sheet symbols can reference the *same* worksheet.

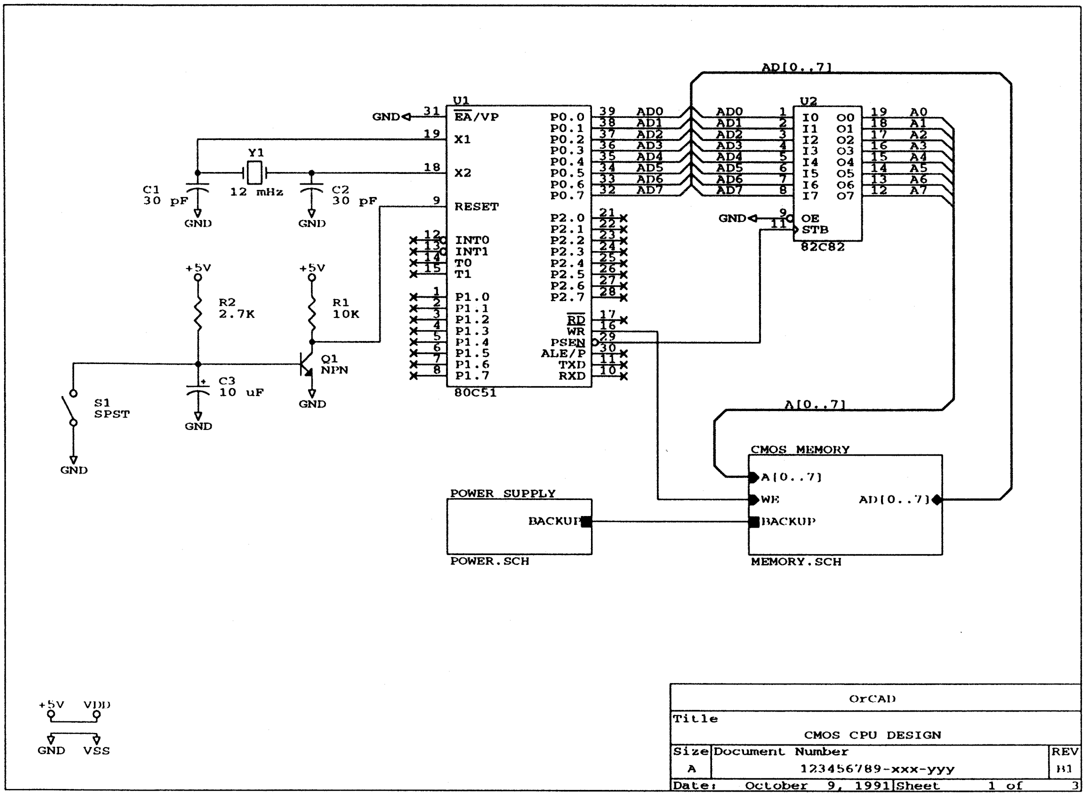


Figure 8-3. The root worksheet of the CMOS CPU design.

Libraries

The CMOS CPU design uses the following libraries:

- ANALOG2.LIB
- DEVICE.LIB
- INTEL.LIB
- MEMORY.LIB
- TTL.LIB

All of these libraries must be configured. Check the **Configured Libraries** list box in the **Library Options** area of the **Configure Schematic Design Tools** screen to determine their status. Follow these steps to configure a library:

1. Click the **Draft** button and then select **Configure Schematic Tools.** The **Configure Schematic Design Tools** screen displays.
2. Pan down to the **Library Options** area.
3. Select the desired library from the **Available Libraries** list box, and then click the **>Insert>** button.

If any of the above mentioned libraries do not appear in the **Configured Libraries** list box, configure them now.

4. Click the **OK** button to save the configurations.

The root worksheet CMOSCPU.SCH

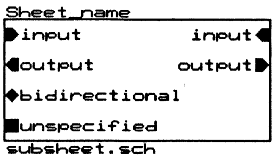
Figure 8-3 shows the root worksheet of this simple hierarchy. The design is called CMOS CPU. The root worksheet filename is CMOSCPU.SCH.

The root worksheet contains a descriptive name in the title block. A title helps identify a worksheet, but it is not required. The title is independent of the filename. A worksheet is identified by **Draft** and the operating system by the worksheet's filename.

The root worksheet of the CMOS CPU design contains:

- An 80C51 and an 82C82 part
- Discrete analog parts: a transistor, capacitors, resistors, and so on
- Two sheet symbols: POWER SUPPLY and CMOS MEMORY
- Power and ground symbols
- Wires and buses connecting the parts

Sheet symbols



Each sheet symbol has a *name* and a *filename*. The name and filename are separate and distinct. The sheet symbol name displays above the sheet symbol; the filename displays below the sheet symbol. The filename has to be identical to the filename of the subsheet that the sheet symbol references. **Draft** uses the sheet symbol filename to nest the referenced worksheet.

The two sheet symbols in figure 8-3 were placed on the root worksheet using the **PLACE Sheet** command. When you place a sheet symbol, **Draft** automatically assigns it a unique filename generated from the date and time of day on your computer. You can accept this unique (but not very descriptive) filename or change it to a filename of your choice. To change the filename, place the pointer inside the sheet symbol and select **EDIT Edit Filename.** The prompt "Filename?" followed by the sheet symbol's current filename displays. Enter the new filename.

In this example, the CMOS MEMORY sheet symbol was assigned the filename MEMORY.SCH, and the POWER SUPPLY sheet symbol was assigned the filename POWER.SCH. MEMORY.SCH references the worksheet on which the circuit's memory is located. POWER.SCH references the worksheet on which the system's power supply is located.

Sheet nets

The CMOS MEMORY sheet symbol contains four sheet nets: A[0..7], WE, BACKUP, and AD[0..7]. These sheet nets were placed into the sheet symbol using the **PLACE Sheet Add Net** commands. Sheet nets are *not* module ports.

The A[0..7] sheet net is connected to a bus with eight members. The bus members are labeled A0 through A7.

While sheet nets do not need to have the same *prefixes* as the bus labels to which they connect ("A" in this example), the sheet nets *must* specify the same *ranges* as the bus labels to which they connect ("[0..7]").

For sheet nets and module ports, there should be no space between the prefix and suffix portion of the names.

Power objects

Power objects represent connections from the outside world to the pins in a part. Unless otherwise specified, power objects are global in scope; they connect to all other signals of the same name.

On the root worksheet a power object named +5V connects to a power object named V_{DD}. This connects the +5V supply to the V_{DD} pins of the 80C51 and the 82C82 parts. Likewise, a power object named GND connects to a power object named V_{SS}. This connects power ground to the V_{SS} pins of the 80C51 and the 82C82 parts.

Nested schematic worksheets

Once the sheet symbols for the nested logic are completed, you then create the worksheets these sheet symbols reference.

△ *NOTE: You don't have to create the root worksheet of the hierarchy before creating the nested worksheets. However, a top-down design methodology is recommended.*

You will not have to create the nested worksheet CMOS MEMORY. It has been provided.

Follow these steps to display the CMOS MEMORY worksheet.

1. Select **QUIT Enter Sheet.**
2. Place the pointer inside the CMOS MEMORY sheet symbol and select **Enter. Draft** displays the CMOS MEMORY worksheet.

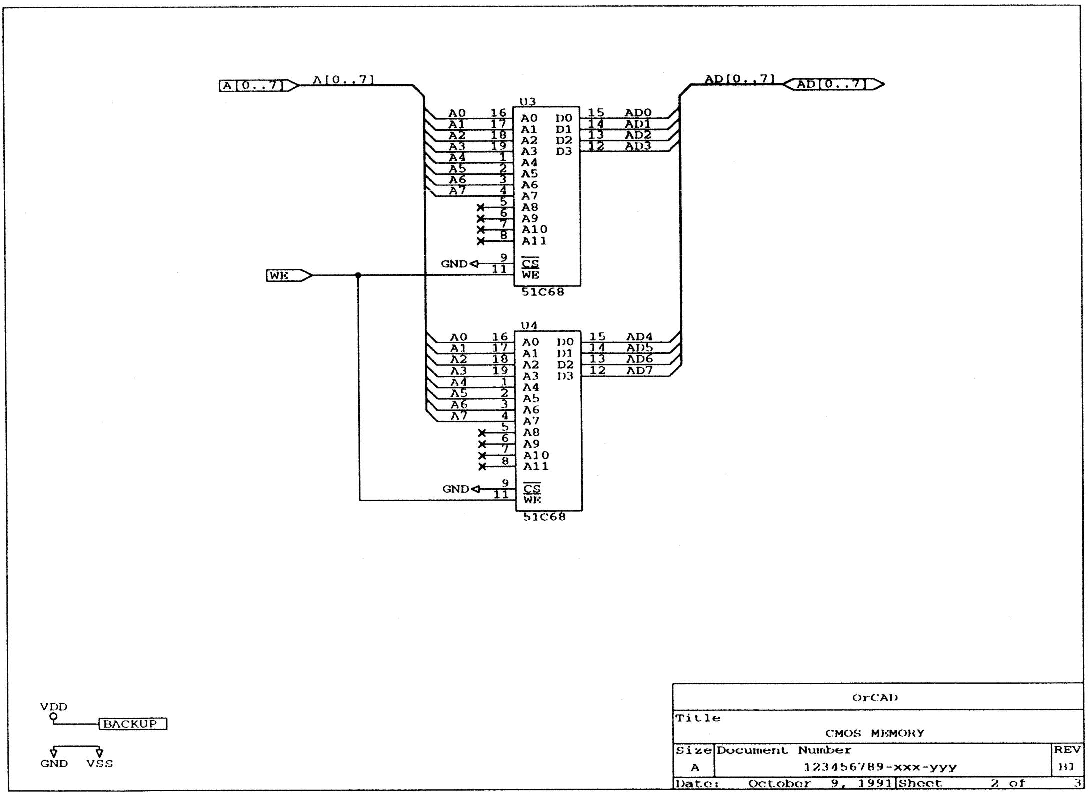


Figure 8-4. CMOS MEMORY worksheet.

The CMOS MEMORY worksheet is the schematic for the circuit's memory.

Notice the four module ports: A[0..7], AD[0..7], WE, and BACKUP. They connect to identically named sheet nets located in the CMOS MEMORY sheet symbol on the root worksheet.

Power is supplied to the CMOS MEMORY worksheet through the module port named BACKUP.

Notice the buses. Buses are automatically connected to module ports with labels having the same name and range—module port A[0..7] connects to a bus labeled A[0..7].

Finally, notice the power objects.

The V_{DD} power object connects to the module port named BACKUP. This isolates the power in the worksheet.

The GND power object connects to the V_{SS} power object. This shorts the GND power net and the V_{SS} power net together. This allows any pin with a GND symbol to be connected to the V_{SS} power net.

When you are done reviewing the CMOS MEMORY worksheet, return to the root worksheet by selecting the **QUIT Leave Sheet** commands.

You will not have to create the nested worksheet POWER SUPPLY. It has been provided.

Follow these steps to display the POWER SUPPLY worksheet.

1. Select **QUIT Enter Sheet.**
2. Place the pointer inside the POWER SUPPLY sheet symbol and select **Enter. Draft** displays the POWER SUPPLY worksheet.

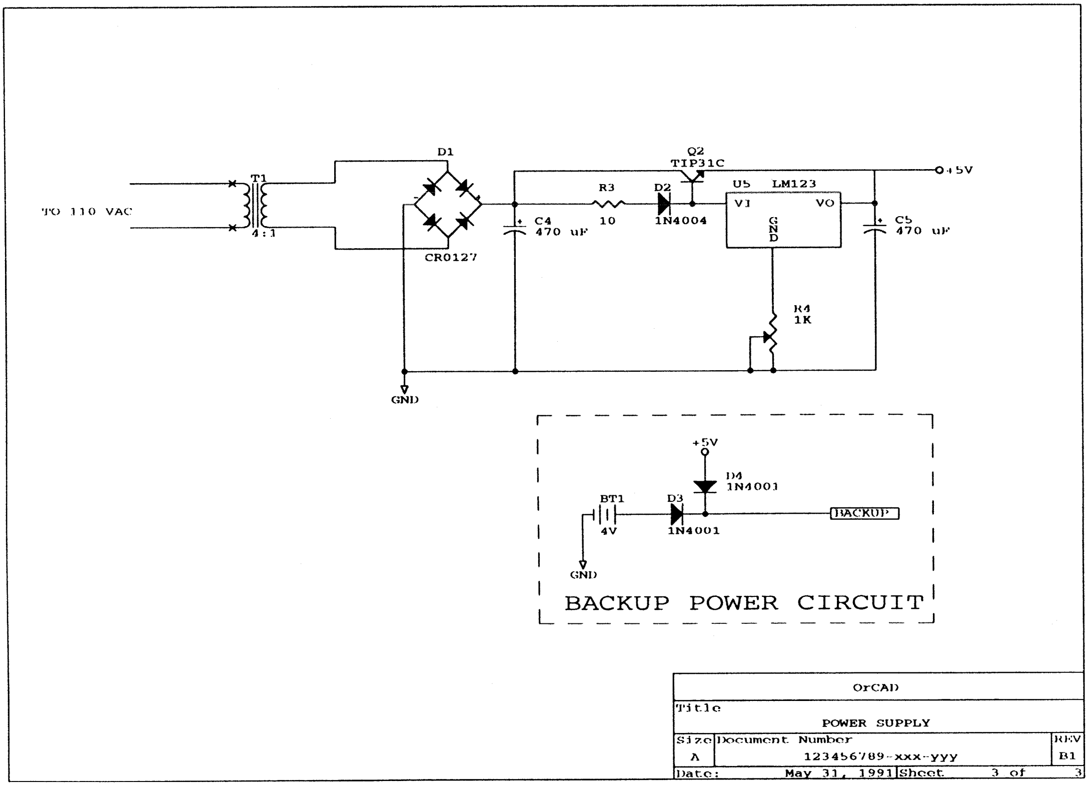


Figure 8-5. POWER SUPPLY worksheet.

The POWER SUPPLY worksheet is the schematic for the power supply circuitry.

Notice the module port named BACKUP. The BACKUP module port makes the *logical* connection to the sheet net named BACKUP. The BACKUP sheet net is in the POWER SUPPLY sheet symbol on the root worksheet, CMOS CPU DESIGN (see figure 8-1).

The BACKUP module port makes the *electrical* connection to the BACKUP module port on the CMOS MEMORY worksheet (see figure 8-4).

When you are done reviewing the POWER SUPPLY worksheet, return to the root worksheet by selecting the **QUIT Leave Sheet** commands.

The next sections in this chapter discuss the following tools: Annotate Schematic, Check Electrical Rules, Show Schematic Structure, and Create Bill of Materials .

Using Annotate Schematic on a simple hierarchy

Once a design is complete, you run **Annotate Schematic** to assign unique values to the part reference designators.

Follow these steps to annotate the simple hierarchy represented by the worksheet, CMOSCPU.SCH:

1. Click the **Annotate Schematic** button on the **Schematic Design Tools** screen. The menu at right displays.

```
Annotate Schematic
Execute
Local Configuration
Assign Hot Key
Show Version
Configure Schematic Tools
Help
```

2. Select **Local Configuration Configure ANNOTATE.** The **Configure Annotate Schematic** screen displays.
3. Check to make sure that the **Source** entry box contains the filename CMOSCPU.SCH. If not, enter **CMOSCPU.SCH** in the **Source** entry box.
4. Click the **OK** button to save the configurations.
5. Click the **Annotate Schematic** button and then select **Execute.**

   As it processes, **Annotate Schematic** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

When **Annotate Schematic** is done, the reference designators for each part in the worksheet have new, unique values. You may want to use **Draft** to view the new reference designator values on the CMOS CPU worksheet.

Using Check Electrical Rules on a simple hierarchy

After the worksheets are annotated, the design should be checked for electrical rule violations. **Check Electrical Rules** checks for several problems associated with a design, including: open input pins, shorts, and bus contention.

Run **Check Electrical Rules** to check for any electrical rules violations in the design. For instructions, refer to the earlier discussion of how to run **Check Electrical Rules.** After you run **Check Electrical Rules,** you may review the error report with **Edit File.**

Figure 8-6 shows how the error report looks for the CMOS CPU.SCH design.

Warnings

Check Electrical Rules flags certain conditions possibly overlooked when your design was created. These *WARNINGS* are not critical errors. In this example, most warnings inform you of inputs with no driving source. This is perfectly acceptable, if these pins are intentionally left open in the design. The connected power supply warnings are also acceptable, since they were intentionally connected in the design.

Errors

Normally, if **Check Electrical Rules** reports *ERRORS* in a design, you should correct them before running other tools.

However, in this example, all warnings are acceptable and other tools may be run.

```
"cmoscpu.sch"
UNCONNECTED REPORT
 X=  4.50   Y=  1.90 I/O                U1,P2.0
 X=  4.50   Y=  2.00 I/O                U1,P2.1
 X=  2.60   Y=  2.10 I/O                U1,INT0
 X=  4.50   Y=  2.10 I/O                U1,P2.2
 X=  2.60   Y=  2.20 I/O                U1,INT1
 X=  4.50   Y=  2.20 I/O                U1,P2.3
 X=  2.60   Y=  2.30 I/O                U1,T0
 X=  4.50   Y=  2.30 I/O                U1,P2.4
 X=  2.60   Y=  2.40 I/O                U1,T1
 X=  4.50   Y=  2.40 I/O                U1,P2.5
 X=  4.50   Y=  2.50 I/O                U1,P2.6
 X=  2.60   Y=  2.60 I/O                U1,P1.0
 X=  4.50   Y=  2.60 I/O                U1,P2.7
 X=  2.60   Y=  2.70 I/O                U1,P1.1
 X=  2.60   Y=  2.80 I/O                U1,P1.2
 X=  4.50   Y=  2.80 I/O                U1,RD
 X=  2.60   Y=  2.90 I/O                U1,P1.3
 X=  4.50   Y=  2.90 I/O                U1,WR
 X=  2.60   Y=  3.00 I/O                U1,P1.4
 X=  2.60   Y=  3.10 I/O                U1,P1.5
 X=  2.60   Y=  3.20 I/O                U1,P1.6
 X=  4.50   Y=  3.20 I/O                U1,TXD
 X=  2.60   Y=  3.30 I/O                U1,P1.7
 X=  4.50   Y=  3.30 I/O                U1,RXD

WARNING: POWER Supplies are CONNECTED GND <-> VSS
WARNING: POWER Supplies are CONNECTED VDD <-> +5V

"POWER.SCH"
UNCONNECTED REPORT
 X=  1.10   Y=  1.10 Passive            T1,AA
 X=  1.90   Y=  1.30 Passive            T1,BCT
 X=  1.10   Y=  1.50 Passive            T1,AB

"MEMORY.SCH"
UNCONNECTED REPORT
 X=  4.30   Y=  2.40 Input              U4,A8
 X=  4.30   Y=  2.50 Input              U4,A9
 X=  4.30   Y=  2.60 Input              U4,A10
 X=  4.30   Y=  2.70 Input              U4,A11
 X=  4.30   Y=  4.50 Input              U5,A8
 X=  4.30   Y=  4.60 Input              U5,A9
 X=  4.30   Y=  4.70 Input              U5,A10
 X=  4.30   Y=  4.80 Input              U5,A11

WARNING: INPUT has NO Driving Source U4,A8
WARNING: INPUT has NO Driving Source U4,A9
WARNING: INPUT has NO Driving Source U4,A10
WARNING: INPUT has NO Driving Source U4,A11
WARNING: INPUT has NO Driving Source U5,A8
WARNING: INPUT has NO Driving Source U5,A9
WARNING: INPUT has NO Driving Source U5,A10
WARNING: INPUT has NO Driving Source U5,A11
WARNING: POWER Supplies are CONNECTED VSS <-> GND
```

Figure 8-6. The error report produced by ***Check Electrical Rules*** *for CMOSCPU.SCH.*

Using Show Design Structure on a simple hierarchy

Use **Show Design Structure** to obtain a text file listing the worksheets in a hierarchy. This tool is helpful for organizing a hierarchy containing many worksheets. Follow these steps to run **Show Design Structure**:

1. Click the **Show Design Structure** button on the **Schematic Design Tools** screen, and then select **Local Configuration Configure TREELIST**. The **Configure Show Design Structure** screen displays.
2. Check to make sure that **the Source** entry box contains the filename CMOSCPU.SCH.
3. Check to make sure that the `Destination` entry box contains the filename CMOSCPU.TWG

   Show Design Structure is now configured to run a schematic structure list for CMOSCPU.SCH and save the results in a text file, CMOSCPU.TWG.
4. Click the **OK** button to save your configurations.
5. Click the **Show Design Structure** button and then select **Execute.**

   As it processes, **Show Design Structure** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   Use **Edit File** to examine the CMOSCPU.TWG text file. The figure below shows the information stored in CMOSCPU.TWG.

```
<<<root>>>
[CMOSCPU.SCH]     November 8, 1990
   CMOS MEMORY
   [MEMORY.SCH]   November 5, 1990
   POWER SUPPLY
   [POWER.SCH]    November 5, 1990
```

All worksheet filenames are enclosed within brackets: [*filename*]. Next to the filename is the date the worksheet was last modified. **Show Design Structure** lists sheet symbol names above the filenames of the worksheets they reference.

In this example, the root worksheet filename is CMOSCPU.SCH. Below the root worksheet filename are sheet symbol names and the filenames of the worksheets they reference. The sheet symbol named CMOS MEMORY references the worksheet with the filename, MEMORY.SCH. The sheet symbol named POWER SUPPLY references the worksheet with the filename, POWER.SCH.

Using Create Bill of Materials on a simple hierarchy

Create Bill of Materials creates a list of parts for all types of design structures . In this example, **Create Bill of Materials** is used on the simple hierarchy, CMOSCPU.SCH. Follow these steps to run **Create Bill of Materials:**

1. Click the **Create Bill of Materials** button on the **Schematic Design Tools** screen and then select **Local Configuration Configure PARTLIST**. The **Configure Create Bill of Materials** screen displays.
2. Check to make sure that the **Source** entry box contains the filename CMOSCPU.SCH.
3. Check to make sure than the **Source file is the root of the design** button is selected.
4. Check to make sure that the `Destination` entry box contains the filename CMOSCPU.BOM.
5. Click the **OK** button to save your configurations.
6. Click the **Create Bill of Materials** button and then select **Execute.**

   As it processes, **Create Bill of Materials** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   To examine the **Create Bill of Materials** text file, use **Edit File**. Figure 8-7 shows the information stored in the text file CMOSCPU.BOM.

CMOS CPU DESIGN
Revised: November 8, 1990
123456789-xxx-yyy
Revision: B1
Bill Of Materials November 8, 1990
11:40:57 Page 1

Item	Quantity	Reference	Part
1	1	BT1	4V
2	2	C1,C2	30 PF
3	1	C3	10 UF
4	2	C4,C5	470 UF
5	1	D1	1N4004
6	2	D2,D3	1N4001
7	1	D4	CR0127
8	1	Q1	NPN
9	1	Q2	TIP31C
10	1	R2	10K
11	1	R2	2.7K
12	1	R3	10
13	1	R4	1K
14	1	S1	SPST
15	1	T1	4:1
16	1	U1	80C51
17	1	U2	82C82
18	1	U3	LM123
19	2		51C68

Figure 8-7. Bill of Materials for CMOSCPU.SCH.

A complex hierarchical design

This section describes a three sheet *complex hierarchy.*

Complex hierarchies are very useful when designing common logic blocks that are repeated. Figure 8-8 shows the root worksheet for the 4BIT ADDER design.

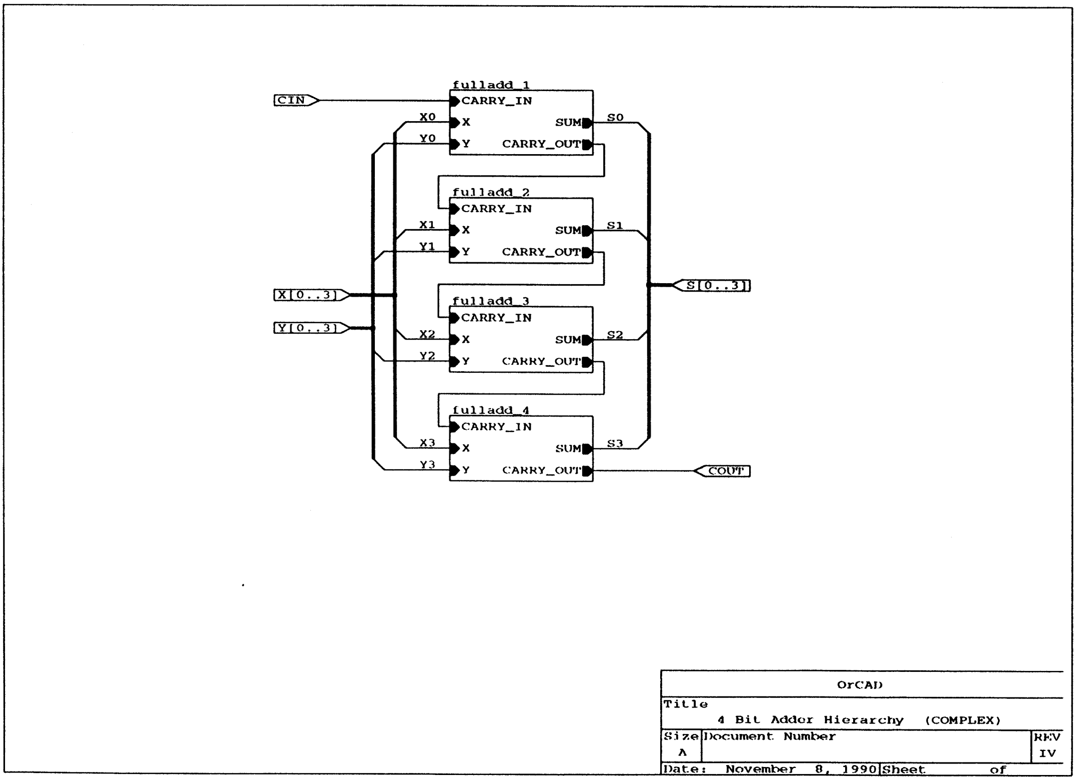


Figure 8-8. The root worksheet of the 4BIT ADDER design.

The 4 BIT ADDER root worksheet

Follow these steps to display the 4 BIT ADDER root worksheet:

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button. The **Design View** portion of the **Design Management Tools** screen displays.
2. Select 4BIT from the **Design** list box.
3. Click the **OK** button. The **Design View** screen is dismissed and the main screen displays.
4. Click the **Schematic Design Tools** button and then select **Execute.** The **Schematic Design Tools** screen displays.
5. Click the **Draft** button and then select **Execute.** The 4BIT ADDER root worksheet displays (see figure 8-9).

The 4BIT ADDER design is a three sheet complex hierarchy. The root worksheet contains four identical sheet symbols, named fulladd_1, fulladd_2, fulladd_3, and fulladd_4.

These sheet symbols reference four full adders that are identical in their design. Because they are identical, it is not necessary to create a separate worksheet for each one. Instead, create just one full adder worksheet and assign the full adder worksheet's filename to all four sheet symbols. Notice that all of the sheet symbols on the root worksheet have the filename FULLADD.SCH.

The full adder worksheet FULLADD.SCH

Follow these steps to display the full adder worksheet.

1. Select **QUIT Enter Sheet.**
2. Place the pointer inside one of the sheet symbols and select **Enter. Draft** displays the worksheet referenced by the selected sheet symbol. Figure 8-9 shows the full adder worksheet.

The full adder worksheet contains two sheet symbols, named HALFADD_A AND HALFADD_B. These sheet symbols reference two half adders that are identical in their design. Because they are identical, it is not necessary to create a separate worksheet for each one. Just as the four full adder sheet symbols in the root worksheet reference full adder worksheet for their logic, the two half adder sheet symbols reference a single half adder worksheet for *their* logic.

The half adder worksheet HALFADD.SCH

Follow these steps to display the half adder worksheet.

1. Select **QUIT Enter Sheet.**
2. Place the pointer inside one of the sheet symbols and select **Enter. Draft** displays the worksheet referenced by the selected sheet symbol. Figure 8-10 shows the half adder circuit.

Each module port in the HALFADD.SCH worksheet is named to connect to the sheet nets in the FULLADD.SCH worksheet, one level up in the hierarchy.

Each module port in the FULLADD.SCH worksheet is named to connect to the sheet nets in the 4BIT.SCH worksheet, one level up in the hierarchy.

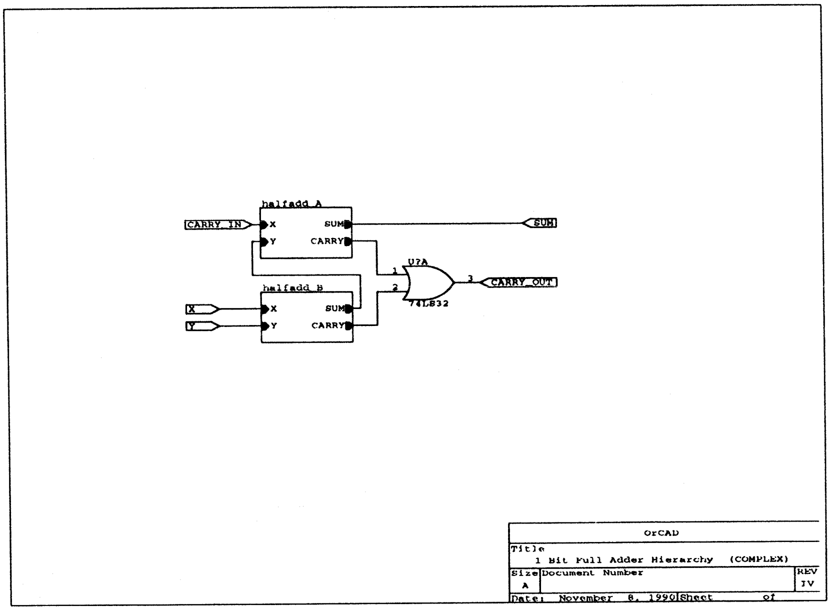


Figure 8-9. FULL ADDER worksheet.

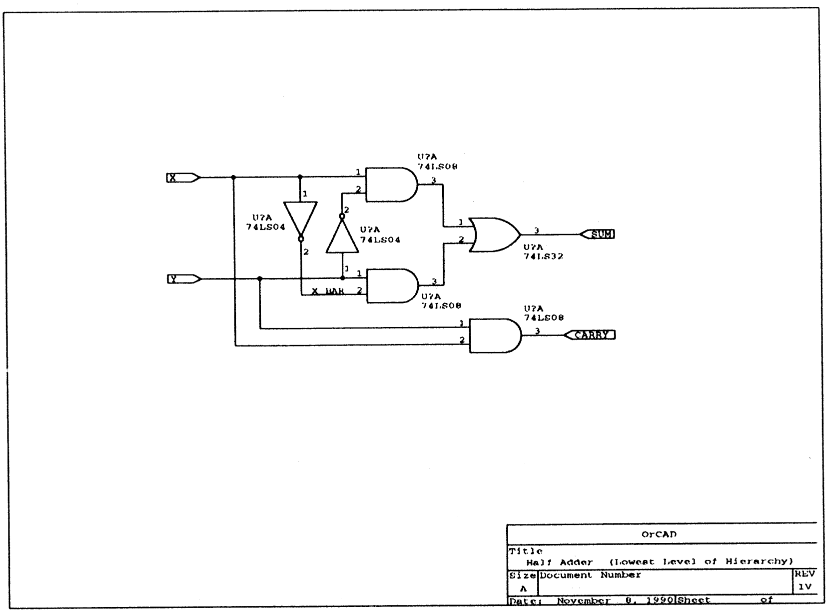


Figure 8-10. HALF ADDER worksheet.

Using Show Design Structure on a complex hierarchy

Use **Show Design Structure** to obtain a text file listing the worksheets in a hierarchy. This tool is helpful for organizing a hierarchy containing many worksheets.

Follow the steps given in *Using Show Design Structure on a simple hierarchy* earlier in this chapter, but substitute the filename `4BIT` for `CMOSCPU`.

Use **Edit File** to examine the 4BIT.TWG text file. The figure below shows the information stored in 4BIT.TWG:

```
<<<root>>>
[4BIT.SCH]    November  8, 1990
  fulladd_1
  [fulladd.sch]    November  8, 1990
    halfadd_A
    [halfadd.sch]    November  8, 1990
    halfadd_B
    [halfadd.sch]    November  8, 1990
  fulladd_2
  [fulladd.sch]    November  8, 1990
    halfadd_A
    [halfadd.sch]    November  8, 1990
    halfadd_B
    [halfadd.sch]    November  8, 1990
  fulladd_3
  [fulladd.sch]    November  8, 1990
    halfadd_A
    [halfadd.sch]    November  8, 1990
    halfadd_B
    [halfadd.sch]    November  8, 1990
  fulladd_4
  [fulladd.sch]    November  8, 1990
    halfadd_A
    [halfadd.sch]    November  8, 1990
    halfadd_B
    [halfadd.sch]    November  8, 1990
```

Notice in the above report that there are a number of references to fulladd.sch and halfadd.sch, and that there are thirteen file references. The 4BIT ADDER design, a complex hierarchy of only three worksheets, expands to thirteen referenced worksheets. Again, the advantage of complex hierarchical design structures is that during the design phase, all of the repeated logic needs to be only drawn once.

Converting a complex hierarchy to a simple hierarchy

While a complex hierarchy is very useful in the design phase, it is not practical for some aspects of the design cycle. This is especially true when a design is to be turned into a printed circuit board. All of the design must then be simplified—converted to a simple hierarchy. This is necessary because all of the parts in the design must be assigned unique reference designators. It would be quite difficult to have a number of parts labeled U17 on the board and have to figure out which was which from the complex hierarchy schematic.

Design Management Tools includes the tool **Complex to Simple**. This tool creates a new design and builds a new version of the complex hierarchy, a version in which each sheet symbol refers to a unique filename.

Follow these steps to run **Complex to Simple**:

1. Click the **Design Management Tools** button on the main screen and then select **Execute.** The **Design View** portion of the **Design Management Tools** screen displays.
2. Click the **Complex to Simple** button. The **Complex to Simple** screen displays.
3. Select 4BIT from the **Designs** list box. The **Source design** entry box contains the name 4BIT.
4. Enter `S4BIT` in the **Destination design** entry box.
5. Click the **OK** button. **Design Management Tools** builds the new design directory and converts the 4BIT design to S4BIT. As it processes, **Design Management Tools** displays the prompt " Working . ." and several messages display at the top left corner of the screen. Then, **Design Management Tools** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.
6. Select **CANCEL** when the process is complete.
7. Select S4BIT from the **Design** list box and click the **OK** button. The main screen displays.

Viewing the S4BIT design

Follow these steps to view the S4BIT design:

1. Click the **Schematic Design Tools** button and then select **Execute.** The **Schematic Design Tools** screen displays.
2. Click the **Draft** button and then select **Execute.**

Notice that 4BIT.SCH is now S4BIT.SCH, FULLADD.SCH is FULLADDA.SCH, FULLADDB.SCH, and FULLADDC.SCH. Also notice that HALFADD.SCH now has eight files: HALFADDA.SCH through HALFADDH.SCH. If you use **Draft** to examine the design, you will see the filenames of the sheet symbols are now all unique.

Running Annotate Schematic on the S4BIT design

As with any new design, you should run **Annotate Schematic** before running the other tools. Other tools report information about the worksheet file, and if you run **Annotate Schematic** first, you ensure that reported information includes the updated reference designators.

The following figures show the simplified design after it is annotated using **Annotate Schematic.**

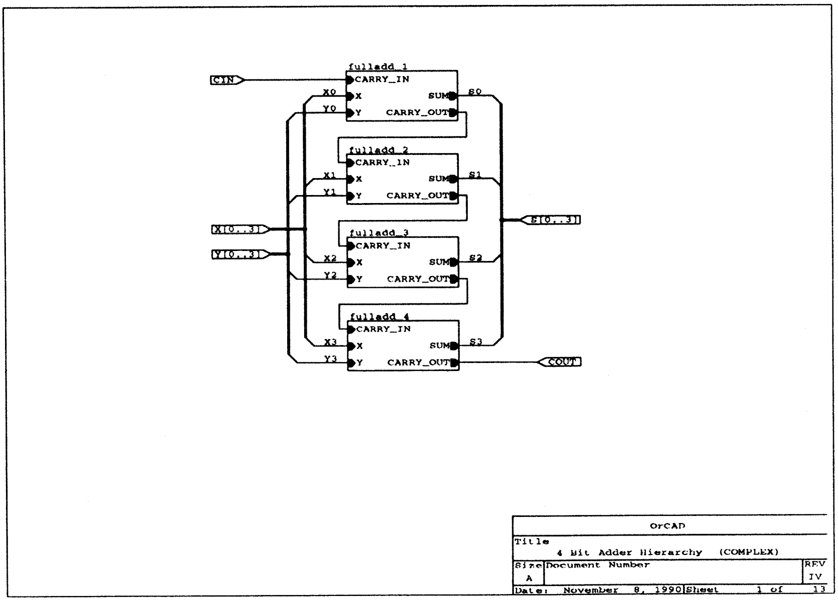


Figure 8-11. 4BIT.SCH schematic.

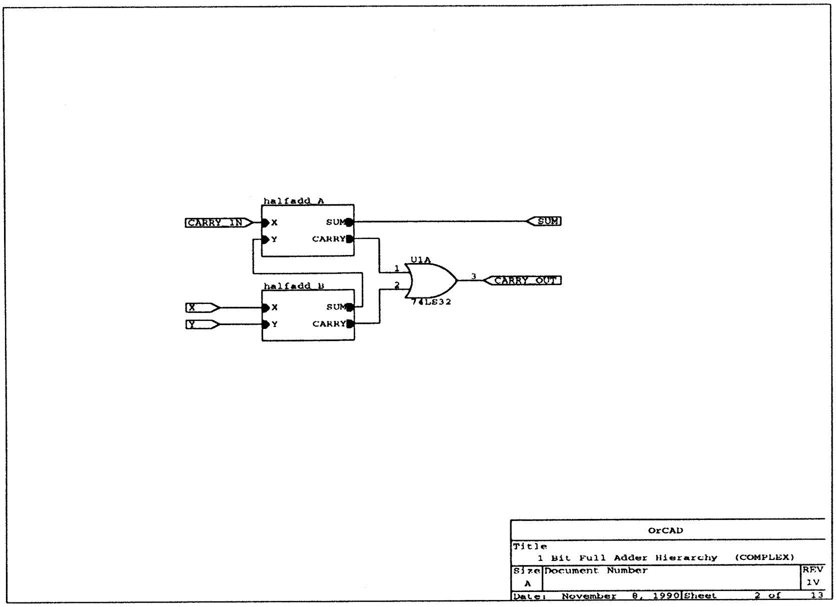


Figure 8-12. FULLADDA.SCH schematic.

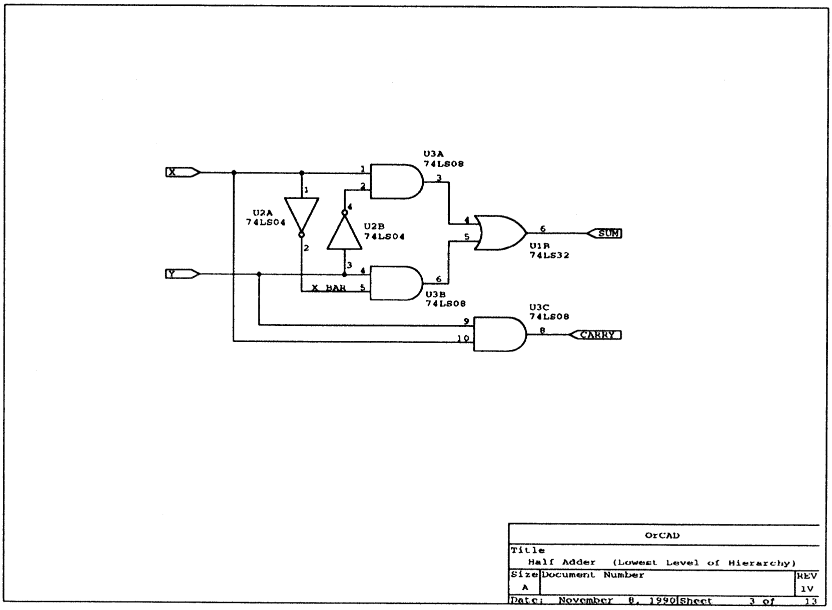


Figure 8-13. HALFADDA.SCH schematic.

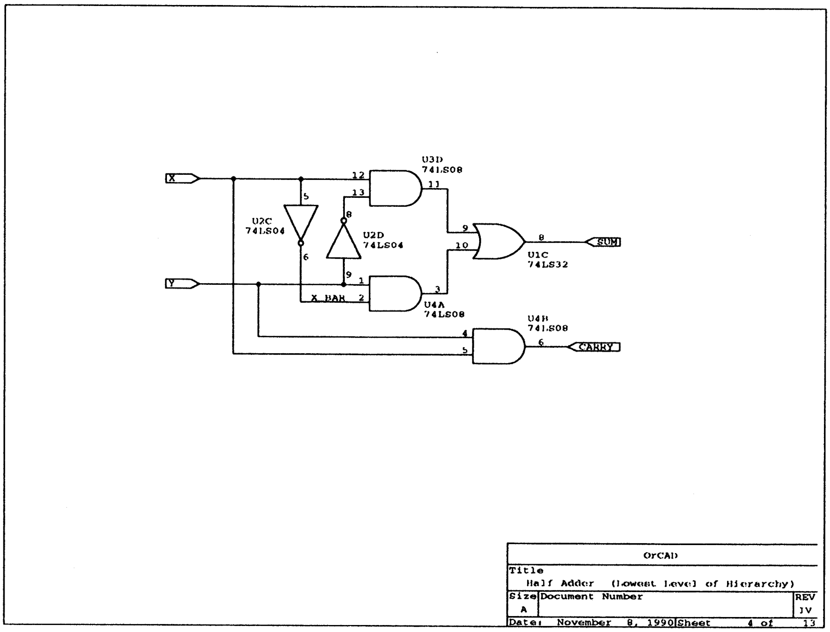


Figure 8-14. HALFADDB.SCH schematic.

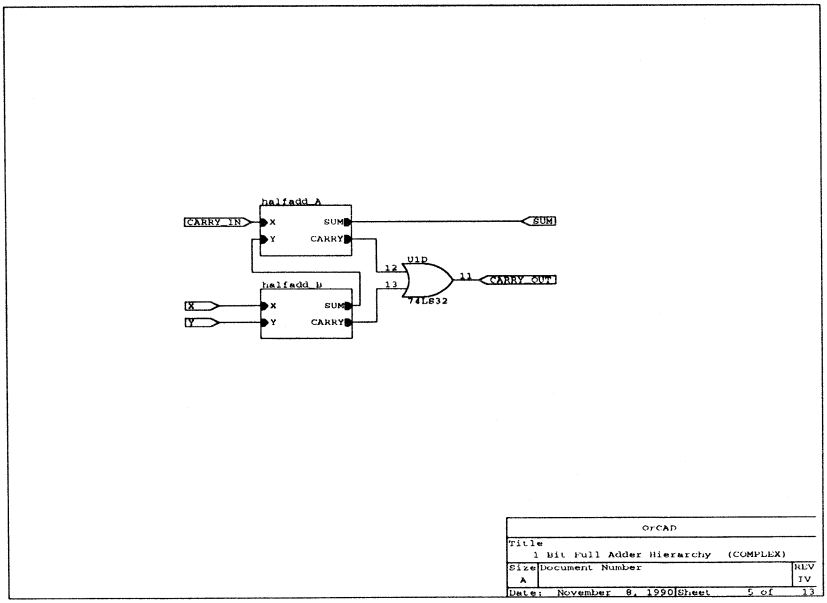


Figure 8-15. FULLADDB.SCH schematic.

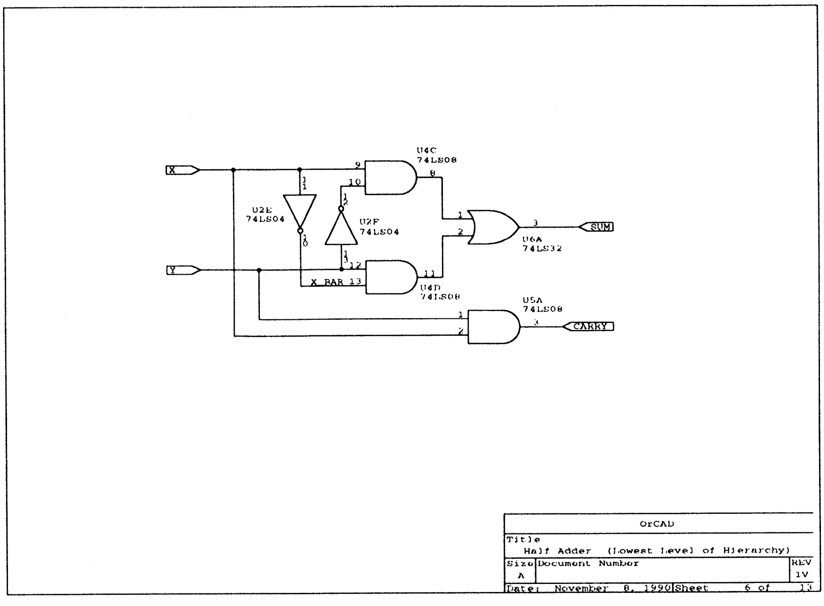


Figure 8-16. HALFADDC.SCH schematic.

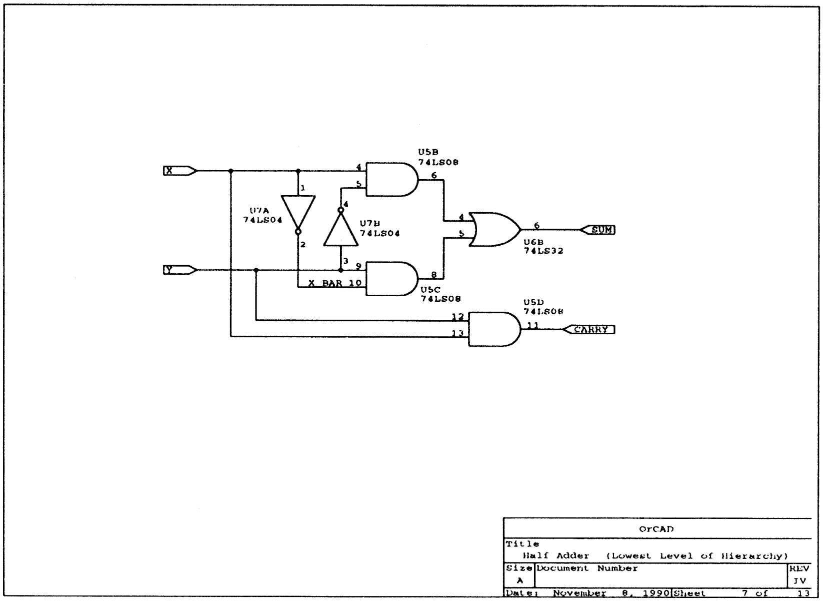


Figure 8-17. HALFADDD.SCH schematic.

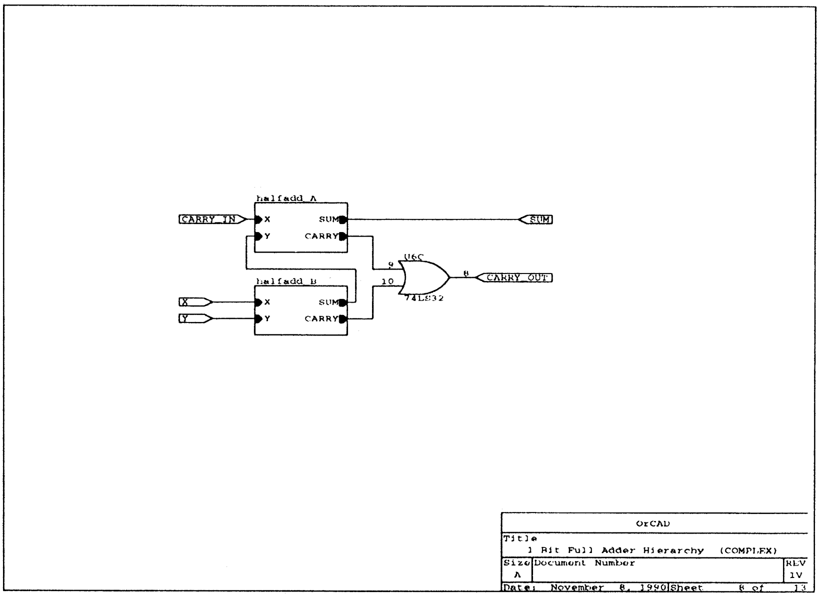


Figure 8-18. FULLADDC.SCH schematic.

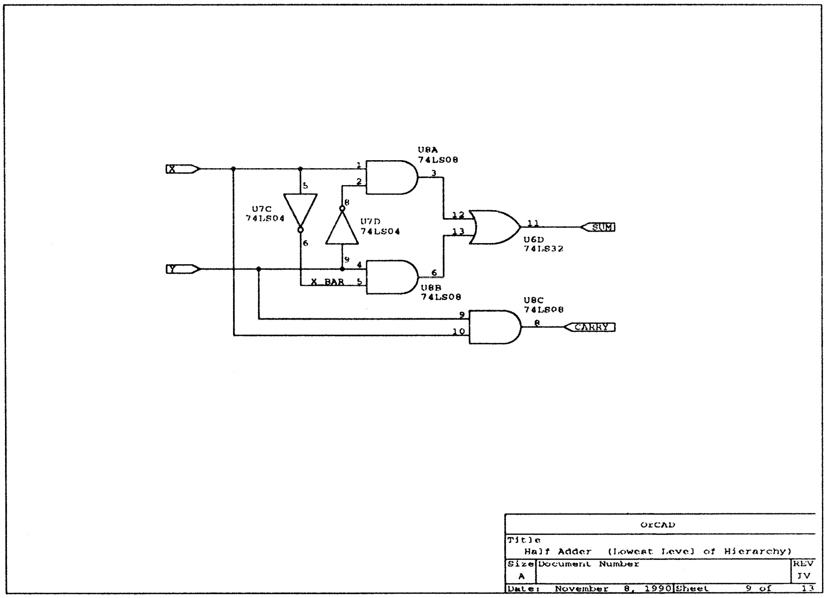


Figure 8-19. HALFADDE.SCH schematic.

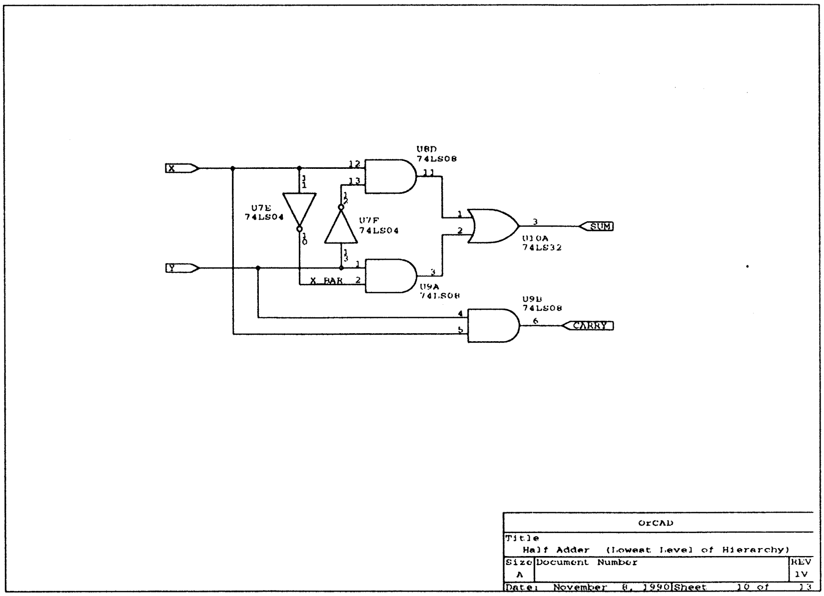


Figure 8-20. HALFADDF.SCH schematic.

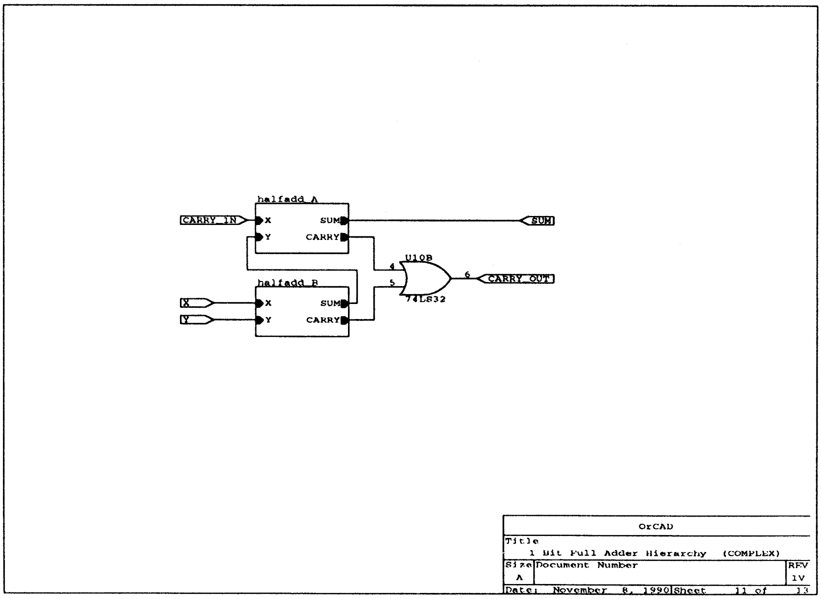


Figure 8-21. FULLADDD.SCH schematic.

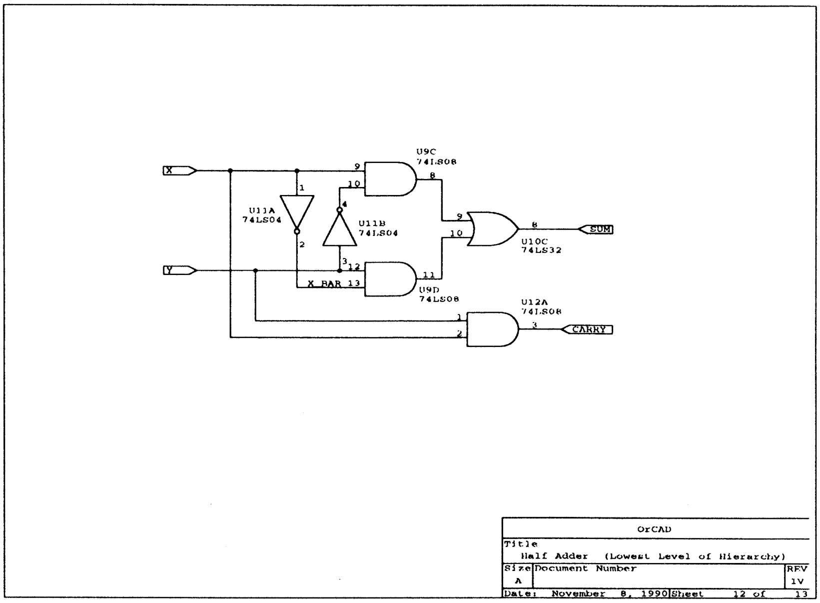


Figure 8-22. HALFADDG.SCH schematic.

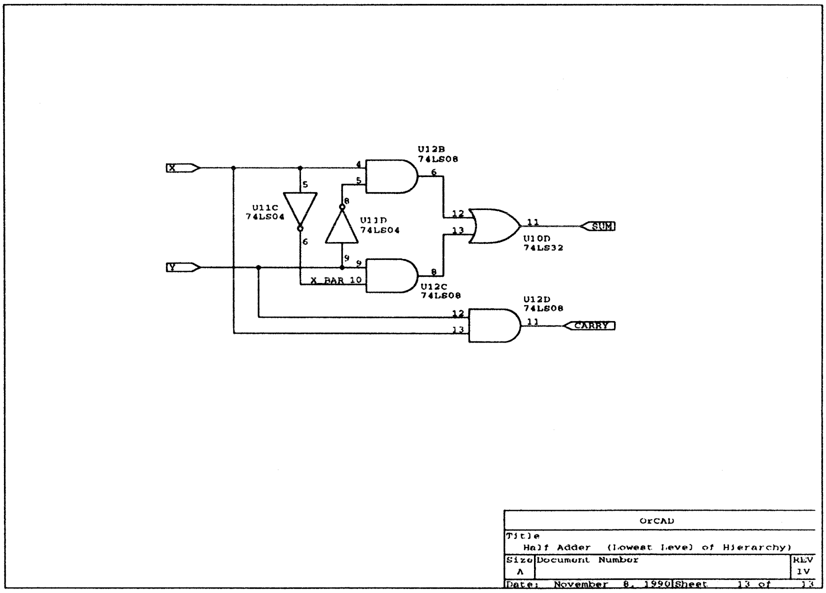


Figure 8-23. HALFADDH.SCH schematic.

Tips and techniques

Overview

This chapter is a collection of tips and techniques to help you better use **Schematic Design Tools**. It includes information about the title block, non-connective objects, EMS, and other topics.

Unlike the other chapters in this guide, this chapter is not tutorial in nature. For additional information about any of the commands, menus, or options described in this chapter, see the chapter for the corresponding tool in the *Schematic Design Tools Reference Guide.*

Converting complex hierarchies

When you use **Complex to Simple** to convert a complex hierarchy that contains any sheetpath parts, be sure not to use the same filename for schematics in the design directory and schematics in the library directory. If you do, **Complex to Simple** uses the schematic in the design directory instead of the schematic in the library directory.

When **Complex to Simple** is copying and naming schematics used more than once in the design, it checks the design directory for duplicate filenames before naming new files. For example, if a complex design includes schematics named INPUT.SCH and INPUTA.SCH, each of which is referenced twice, the simplified design contains these four files:

- INPUT.SCH
- INPUTA.SCH
- INPUTAC.SCH
- INPUTB.SCH

Title block tips

Schematic Design Tools provides a great deal of flexibility with the title block on a worksheet. You can use **Draft**'s standard title block, an ANSI title block, or you can create your own title block. If you have paper that has your title block pre-printed on it, you can set up **Draft** so that the title block is not plotted, and in its place, text is plotted in locations to line up with your pre-printed title block. This section discusses the many different ways that title blocks can be manipulated.

OrCAD's title block

The **Schematic Design Tools** schematic editor, **Draft,** creates a title block that looks like this:

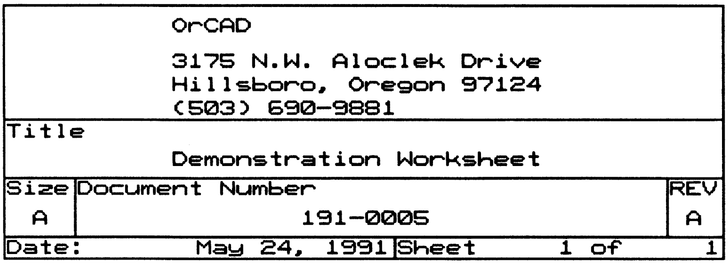


Figure 9-1. Sample OrCAD title block.

You define all of the information on the title block except the date and size. **Draft** automatically enters the size and modification date of the worksheet.

ANSI title block

You can configure **Schematic Design Tools** so that **Draft** creates an ANSI title block:

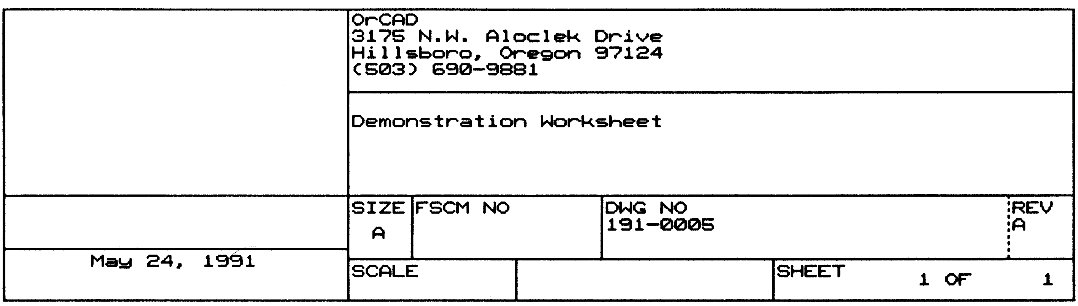


Figure 9-2. ANSI title block.

The ANSI title block conforms to the guidelines given in ANSI Standard Y14.1-1980. As you can see in figure 9-2, the ANSI title block is larger than the default OrCAD title block. On an A-size drawing, it takes up a large amount of the drawing area.

See *On the Configure Schematic Design Tools screen* in this section for instructions on how to create an ANSI title block.

△ *NOTE: If you use an ANSI title block, you may want your worksheet to have ANSI standard dimensions. These dimensions are given in table 1-3 in this guide. Since most, if not all, PC-compatible printers and plotters cannot print as close to the edge of the page as specified in the ANSI standard, OrCAD's default worksheet dimensions are reduced. These dimensions are given in tables 1-5 and 1-6. If your printer or plotter can print closer to the edge of the paper, adjust the worksheet size in the* ***Template Table*** *area of the* ***Configure Schematic Design Tools*** *screen.*

Defining title block information

You define title block information in one of two places: in **Draft** or on the **Configure Schematic Design Tools** screen. Both are described below.

In Draft

To define title block information in **Draft**, place the pointer in the title block and select **EDIT**.

The menu shown at right displays.

```
Edit title block
Revision code
Title of sheet
Document number
Sheet number
Number of sheets
Organization name
1st Address Line
2nd Address Line
3rd Address Line
4th Address Line
```

Select the field to edit and answer the prompts that display.

For more information, see the **EDIT** command description in *Chapter 2: Draft*.

△ *NOTE: If you are using an ANSI title block (see figure 9-2), you must use the* ***PLACE Text*** *command to place text in the* ***FSCM NO*** *and* ***SCALE*** *boxes. The* ***EDIT*** *command does not contain menu items to edit this information.*

On the Configure Schematic Design Tools screen

To define title block information on the **Configure Schematic Design Tools** screen, display the **Configure Schematic Tools** screen and pan to the **Worksheet Options** area (shown below).

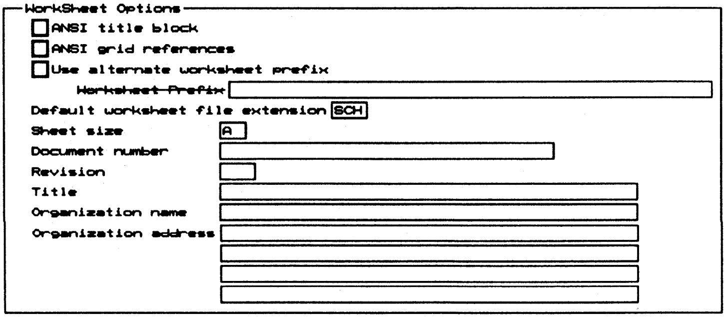


*Figure 9-3. **Worksheet Options** area of the **Configure Schematic Design Tools** screen.*

Information entered here automatically displays in the title block of each schematic created after the information is defined.

To use an ANSI title block (pictured in figure 9-2), select the **ANSI title block** option on this screen.

See *Worksheet Options* in chapter 1 for more information.

➤ *Helpful hint . . .*

Consider defining title block information such as organization name and address on the **Configure Schematic Design Tools** screen in your TEMPLATE directory. That way, each new design you create will be set up with your company name and address.

Suppressing the title block

You can suppress the title block's lines, text, or both. Methods for doing this are described here.

Suppressing title block lines

To suppress title block lines and leave title block text on the worksheet, display the **Configure Schematic Design Tools** screen. Pan down to the **Color and Pen Plotter Table** area.

Click in the **Pen** entry box to the right of **Title Block**. Enter **99** to tell **Plot Schematic** not to plot the title block.

When you open a worksheet in **Draft**, notice that the title block lines still display. However, they do not appear on the plot, as shown in the figure below.

```
                 OrCAD

                 3175 N.W. Aloclek Drive
                 Hillsboro, Oregon 97124
                 (503) 690-9881
Title
                 Demonstration Worksheet

Size Document Number                                REV
  A                   191-0005                       A
Date:         May 24, 1991 Sheet      1 of          1
```

Figure 4. Plot of a title block with lines suppressed.

△ *NOTE: This also turns off the border around the drawing area during printing.*

Suppressing title block text

To suppress title block text and leave title block lines on the worksheet, display the **Configure Schematic Design Tool** screen. Pan down to the **Color and Pen Plotter Table** area.

Click in the **Pen** box to the right of **Title Text**. Enter **99** to tell **Plot Schematic** not to plot the title block's text.

When you open a worksheet in **Draft**, notice that the title block text still displays. However, it does not appear on the plot, as shown in the figure below.

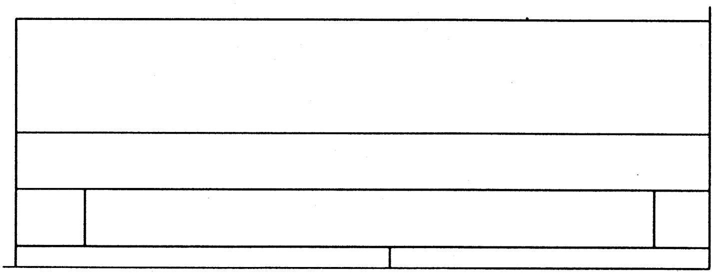


Figure 5. Plot of a title block with text suppressed.

Suppressing title block lines and *text*

There are two ways to suppress both title block lines and text:

- Use **Draft's SET Title Block No** command. If you use this method, the title block and its text do not display on the screen or appear on a print or a plot.
- Set both the title block and title text to a pen of 99 on the **Configure Schematic Design Tools** screen. If you use this method, the title block and its text display on the screen, but do not appear on a plot.

➤ *Helpful hint . . .*

If you are plotting on paper that has your title block pre-printed on it, suppress the title block and its text as described previously. Use **Draft's PLACE Text** command to place text in the correct position so that when you plot your schematic the text prints in the correct place in your title block.

Creating a custom title block

You can create a custom title block using a library part made to look like a title block, or by using wires without labels. Both of these methods are described below.

Using a library part

To create a library part that looks like a title block, use **Edit Library** to make a library part that looks like a title block. Be sure that the part is non-connective. This means it must be a zero-element part, and cannot have any pins. For more information, see *Non-connective objects* in this appendix.

In **Draft**, suppress the title block using the **SET Title Block No** command, as described previously.

Place the library part on each new schematic.

Using wires

To use wires to create a title block, suppress the title block using **Draft's SET Title Block No** command, as described previously.

Draw the title block using the **PLACE Wire** command.

If text is required, be sure to place text , not labels. The netlist tools ignore the wires if they have no labels or no pins attached.

➤ *Helpful hints . . .*

If you would like wider lines in your title block, draw buses rather than wires.

You may want to use a combination of both of the methods described above. For example, you can create a logo as a library part , then draw the title block with wires, and place the logo in your title block.

Using your custom title block in each design

Once you create a custom title block, you can easily duplicate it on each new design using one of the methods described here.

Create a template schematic

Use one of the methods described in *Creating a custom title block* to create a worksheet that contains only the title block. Keep this worksheet in the TEMPLATE directory. It will be copied into each new design. If it has a name of TEMPLATE.SCH, it will always be given the name of the new design with an extension of .SCH.

Create a macro

Create a macro that draws the title block with wires. This macro can also include any text that must be part of the title block.

Place the macro in a macro file in your TEMPLATE directory. It will become a part of each new design. Run the macro in each new design.

Archiving parts

To achieve more efficient use of memory, run **Archive Parts in Schematic** on each design, turning both LIBARCH and COMPOSER on. This creates a library containing only the parts used in your design.

Doing this protects the design from changes in standard libraries and results in more efficient memory use because you only have to configure one library instead of several.

Non-connective objects

Three **Schematic Design Tools** libraries contain objects that have no electrical connectivity. These objects are not included in the netlist and can therefore be used to custom-ize your schematic worksheet as explained in this section.

About non-connective objects

You may want a design to contain objects that have no electrical connectivity and are not processed by **Create Netlist** or **Create Hierarchical Netlist**. These objects can be:

- Mounting holes.
- Mechanical hardware such as screws and washers.
- A physical representation of the device you are designing.
- Pins that are floating or not connected that can connect to an option such as unused pins on a serial port.
- Flowchart symbols.

OrCAD/SDT III

In OrCAD/SDT version 3.22, you could place a part on a schematic for illustration purposes by deleting the part's value and reference fields. The part would not appear in the netlist. NETLIST viewed these unconnected devices as errors and reported:

```
<<<ERROR>>> X=  .80,Y=  1.00 Part has no REFERENCE
Part has no VALUE -  Part will be ignored.
```

However, it still produced a netlist.

Schematic Design Tools Release IV

Because of the enhanced error processing capabilities of Release IV software, **Create Netlist** and **Create Hierarchical Netlist** interpret these unconnected parts as errors and terminate the netlist process without producing a netlist or connectivity database. It halts at the error and goes no further. Since this is an "error" and not a "warning," it is not effective to use INET's **Ignore Warnings** option.

Release IV solution

Release IV includes special libraries that contain non-connective objects that can be placed on the schematic and still take advantage of the powerful error checking available in the Release IV netlist tools.

These non-connective objects have no reference designators, values, or pins. They are found in the libraries listed in the table below.

Library	*Contents*
ASSEMBLY.LIB	Part outlines for assembly drawings to specify position of devices for board placement.
FLOWCHT.LIB	Programming flowchart symbols.
SHAPES.LIB	Generic library containing circles, squares, 90° arcs, and diamonds.

Libraries that contain non-connective parts.

See technical note #30: *Non-connective objects in Schematic Design Tools* for illustrations of some of the parts found in these libraries.

Placing non-connective objects on your schematic

Follow the steps below to place non-connective objects on your schematic:

1. Use **Edit Library** to create an object that looks like an actual device but has no electrical connectivity. You can use objects from any of the libraries listed in the table above. Use the objects as they are, or use them to create a new object.

   To be non-connective, an object must be a zero-element part and cannot have any pins. Since IEEE parts are always single-element parts, they cannot be used. Only block or graphic parts can be non-connective.

2. Place the part in a custom library. You can use it whenever you need it.

Converting SDT III schematics to Schematic Design Tools Release IV

You may have schematics developed prior to Release IV that contain parts with deleted reference designators and part values. If this is the case, your schematic will not pass the rigorous checking of the Release IV netlist tools.

To correct this, use **Edit Library** to change the objects in question so that they have no pins and are zero-element parts. When you run **Create Netlist** or **Create Hierarchical Netlist**, the new parts will not produce errors.

△ *NOTE: If you run **Create Netlist** or **Create Hierarchical Netlist**, and then change parts in one of the libraries, you must select the **Unconditionally process all sheets in design** option when you run **Create Netlist** or **Create Hierarchical Netlist** again. Since the schematic itself has not changed, INET will not detect any changed time stamps and will not find anything to process.*

Uppercase letters in key fields

Lowercase letters entered in key fields are handled as literals. To use the special characters "V" for **Value** and "R" for **Reference** and have them interpreted correctly, enter them in uppercase.

Duplicate sheet names

Sheet symbols in a design must have unique names. If your design has two sheet symbols with the same name, you receive this message when you run **Create Netlist**:

```
Duplicate Sheet Names
```

For example, two sheet symbols named `SHEET` cause this message to display, but two sheet symbols `SHEET1` and `SHEET2` do not.

Changing netlist formats

Create Netlist and **Create Hierarchical Netlist** create netlists incrementally. When you run a netlist, only the items that *changed* are included. A new file date or time denotes change. If you configure your netlist for one for-mat, run a netlist, then configure it for a different format and run a netlist again, the netlist does not change.

To produce a netlist with a new format from an unchanged design, run IFORM with the **Force IFORM to always create a formatted netlist** option selected.

About EMS

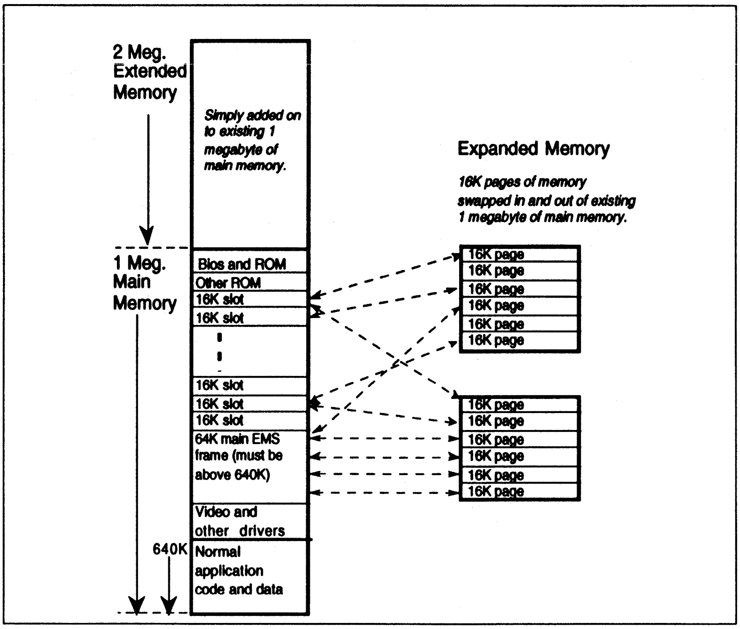
This section describes how the ESP design environment and **Schematic Design Tools** use EMS (Expanded Memory Specification).

What is EMS?

EMS is an acronym for Expanded Memory Specification. The full acronym is LIM-EMS, for Lotus-Intel-Microsoft Expanded Memory Specification.

EMS specifies how software works with special hardware to swap 16K *pages* of memory into and out of the one megabyte of main memory typically available in IBM PCs and compatibles.

Using memory management software, an application can read and write data on one page, swap in another page, and then swap back to the first page, with all data intact. Using this method, an application can use a small number of 16K main memory *slots* to access much more than one megabyte of memory. The figure below shows the main memory slots and the 16K pages that are swapped into and out of the slots.



Expanded memory (EMS) and extended memory.

△ *NOTE: Expanded memory is sometimes confused with extended memory. Extended memory is memory above the one-megabyte of main memory, while expanded memory is memory that is swapped into and out of the one megabyte of main memory. Some software can make extended memory act like expanded memory. While Release IV software may work with this type of software, it does not use extended memory directly.*

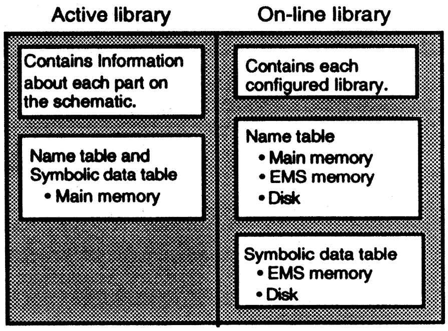
EMS in the ESP design environment

When the design environment is run, it checks to see if at least two 16K pages of EMS are available. If they are, the design environment loads its display driver into EMS. If the design environment and an OrCAD tool set use the same display driver, it only needs to be loaded once. This removes the display driver from the lower 640K of memory, providing more lower memory for the OrCAD tool set.

When you exit ESP and return to DOS, all EMS used by the design environment and OrCAD tool sets is released. The EMS is then available for other applications.

EMS in Schematic Design Tools

Schematic Design Tools uses two types of libraries: the active library and the on-line library. Both of these libraries contain a *name table* and a *symbolic data table*. The name table is a list of parts. The symbolic data table contains symbolic information about each part. The figure above right shows the active library, the symbolic library, and what each contains.



Schematic Design Tools' active library and on-line library.

Active library

The active library contains information about each part *on the schematic*. It always resides in main memory and can be configured to be 64–152K (this is done on the **Configure Schematic Design Tools** screen).

- The name table contains a list of the parts found *on the schematic*.
- The symbolic data table contains all of the symbol information for each part *on the schematic*.

On-line Library

The on-line library contains information about *each configured library*.

- The name table contains a list of all the parts in *each configured library*. It can be stored in main memory, EMS memory, or on disk.
- The symbolic data table contains all of the symbol information for each part in *each configured library*. It can be stored in EMS memory or on disk.

Configuring Schematic Design Tools to use EMS

Follow these steps to configure **Schematic Design Tools** to use EMS:

1. Select **Draft** and **Configure Schematic Tools** from the **Schematic Design Tools** screen. The **Configure Schematic Design Tools** screen displays.
2. Move to the **Library Options** area.

   Notice the headings **Name Table Location** and **Symbolic Data Location** at the bottom of the **Library Options** area.
3. Select the desired location for each of these tables. The next section discusses the performance impacts of the different configurations.

Performance impacts

Depending on the location of the on-line library's name and symbolic data tables, you can expect the performance impacts listed below. This list is given in order of efficiency. The most efficient configuration is given first, the least efficient is given last.

- **Name table in main memory and symbolic data table in EMS.** **Draft's GET** and **LIBRARY Browse** commands execute fastest under this configuration.
- **Name table in EMS memory and symbolic data table in EMS.** The **GET** and **LIBRARY Browse** commands may be slightly slower than in the first configuration. However, you can add additional EMS memory to get as many parts on line as possible.
- **Name table in main memory and symbolic data table on disk.** The **GET** and **LIBRARY Browse** commands are even slower, but are still tolerable. This is the best option for PCs without EMS.

- **Name table in EMS memory and symbolic data table on disk.** This configuration should only be used for the following special cases:
  - Very large designs, such as E size drawings with a large number of parts.
  - PCs with a small amount of EMS memory.
  - PCs with a small amount of available main memory. This can be caused by running multi-tasking software or a large network driver.

  Performance in this configuration is degraded compared to the above configurations, but is still acceptable.
- **Name table on disk and symbolic data table on disk.** This is the slowest configuration. It should only be used with portable computers that have 512K main memory. It is tolerable for long use only if the hard disk is fast.

Viewing EMS memory allocation in Draft

The **CONDITIONS** command in **Draft** displays the amount of EMS and main memory used by the active library and the on-line library. To view this data, simply select **CONDITIONS** from **Draft's** main menu. When you are done viewing this information, press any key to return to **Draft**'s main menu.

△ *NOTE: The* ***CONDITIONS*** *command displays information about the active library and the "reference" library. The reference library is the same as the on-line library.*

Reporting unused match strings

When printing a bill of materials report using **Create Bill of Materials** and an include file, you may want to find out which strings in your include file do not have a corresponding match string in the design.

To do this, make the following settings on the local configuration screen for **Create Bill of Materials:**

- Specify your include file by selecting the **Merge an include file with report** option and entering the include file name in the **Include** entry box.
- Select the **Report un-used match strings in include file** option.

Copying parts from one library to another

Follow these steps to copy parts from one library to another:

1. Use **Edit Library**'s **EXPORT** command to copy a part from the original library to a temporary file.
2. Run **Edit Library** on the target library and **IMPORT** the temporary file.
3. Edit the part if necessary.
4. Select **LIBRARY Update Current**.
5. Select **QUIT Update File**.

Encapsulated PostScript

This section describes how to take an EPS file from an OrCAD application into two widely-used Microsoft programs, Word for Windows (WINWORD) version 1.1a and Word 4.0 for the Macintosh.

Creating EPS

Follow the instructions here to create an EPS file that can be incorporated into any application that accepts EPS .

Configure the tool set

Display the **Configure Schematic Design Tools** screen. Select one of the four EPS drivers. (To install the drivers on your system, run the INSTALL program from the Install disk.) Most illustrations in OrCAD manuals use EPS1.DRV.

Locally configure the Plot Schematic tool

Display **Plot Schematic**'s local configuration screen. Select the option **Send output to a file,** and the **Create a plot file** option below it. Some applications, such as WINWORD, require EPS filenames end with .EPS, so enter **EPS** in the **File Extension** entry box. Then, select **Automatically scale and set X,Y offsets for specified sheet size** and the **A** size option below.

Plot the schematic to disk

Once the tools are configured, you make an EPS file just by clicking the **Plot Schematic** button.

You can now bring the EPS file into other applications.

Placing EPS in WINWORD

In WINWORD version 1.1a, EPS support is built into NORMAL.DOT. Make sure your EPS filename ends with .EPS (required by WINWORD's EPS filter). Then, open NORMAL.DOT, select **Insert**, then **Picture**, and enter the name of the EPS file. In a moment, the placeholder for the EPS image appears.

That is all there is to it. The schematic does not display on your screen, but it will print all right on a PostScript printer. Earlier versions of WINWORD worked differently, so be sure to read version 1.1a's README.DOC.

Placing EPS into Word 4.0

On the Macintosh, you can see the EPS graphic, manipulate it, and place it into many applications.

Transfer the plot file from the PC to the Macintosh

Use one of many possible techniques to transfer the plot file from the PC to the Macintosh. For example, use MacLink Plus™ V5.01 (from DataViz Inc.) with file translation set to "EPS to EPS."

Create a screen image of the file

Create a screen image of the EPS file. The desk accessory SmartArt™ V2.0 (from Adobe) does this very simply. The Macintosh must be connected to a PostScript printer. Open SmartArt, select "EPSF and Text," open the plot file, and select "Reimage."

The reimage option sends the EPS file to the PostScript printer for processing. The printer maps the EPS instructions to pixels and sends the image back. Save the image.

Select the Copy command to copy the image to the clipboard or scrapbook. Close SmartArt.

Open a Word document

Paste the image in place in a Word document. If necessary, size or crop the image to fit the space available.

Error objects

If you run **Check Electrical Rules** and it flags errors on your schematic, remove them using **Draft's QUIT Update File** command.

Using sheets and parts to point to another worksheet

To create a hierarchy, you place an object on your worksheet that points to another worksheet file. This object can be a sheet symbol, a sheet part, or a sheet path part. Their similarities and differences are discussed in detail in this section.

About sheets and parts

Before discussing sheet symbols, sheet parts, and sheet path parts, it is important that you understand the difference between a *sheet* and a *part*. In trying to understand these differences, think of the function of the object rather than the object's physical appearance.

Parts are graphic objects that you place on the worksheet to represent the electronic devices in your design. Among other characteristics, they have part fields, are annotated, and may have power pins. If you print a bill of materials report, all parts on your worksheet are listed.

Sheets are objects that you place on a worksheet that point to another worksheet file. Sheets do not have part fields, are not annotated, and cannot have power pins. Sheets are not listed in a bill of materials report.

Table 9-1 compares sheets and parts.

Characteristic	*Sheet*	*Part*
Part fields	doesn't have	has
Annotated	no	yes
Bill of materials report	does not appear	appears
Appears in the incremental connectivity database as	an instance of a child	a part
Power pins	cannot have	may have

Table 9-1. Characteristics of sheets and parts.

Sheet symbol

A sheet symbol is an ANSI standard rectangle that you place on a worksheet using **Draft's PLACE Sheet** command. **Draft** automatically assigns it a random filename, such as 91G8F06#.SCH. You can tell it to point to the file of your choice by selecting **PLACE Sheet Filename.**

A sheet symbol functions as a sheet, and has all the characteristics given in column 2 of table 9-1.

The worksheet file that a sheet points to should be stored in the current design directory unless a path is specified with the **Filename.**

Sheet part

A sheet part is a library part that you change to a sheet. You place a library part on a worksheet using **Draft's GET** command. To change it to a sheet part, you use **Draft's EDIT SheetPart Name** command to tell it to point to a worksheet file.

Once you give a library part a sheet part name, it no longer functions as a part. It functions as a sheet, and has all the characteristics given in column 2 of table 9-1.

The worksheet file that a sheet part points to should be stored in the current design directory unless a path is specified with the **SheetPart Name.**

Sheet path part

A sheet path part is similar to a library part in that it is stored in a library. It may function as a sheet, *or* as a library part. When you get a sheetpath part from a library, it *already* points to a worksheet file.

If you select the **Descend into sheetpath parts** option on a tool's local configuration screen, a sheet path part functions as a sheet, and has all the characteristic given in column 2 of table 9-1. If you don't select this option, it functions as a part and has all of the characteristics given in column 3 of table 9-1.

OrCAD libraries don't contain any sheetpath parts. You must create them and add them to a library. The worksheet file that a sheetpath part points to should be stored in the directory specified in the **Library Prefix** entry box in the **Library Options** area of the **Configure Schematic Design Tools** screen.

Conclusion

As described previously in this section in *Sheets and parts,* be sure to think of the function of an object rather than it's physical appearance. Table 9-2 summarizes the different type of objects, and tells whether they function as a part or a sheet.

Type of object	*Functions as a sheet*	*Functions as a part*
Library part	no	yes
Sheet symbol	yes	no
Sheet part	yes	no
Sheet path part	if **Descend into sheetpath parts** option is selected	if **Descend into sheetpath parts** option is *not* selected.

Table 9-2. Functionality of objects.

A

analog ▪ Circuitry where both voltage and frequency output vary continuously as a function of the input.

annotation ▪ Assigning reference designators to parts in a schematic.

area ▪ A section of a screen containing related buttons or configuration options. Most areas are bordered and named. For example, the **Editors** area on tool set screens and the **File Options** area on local configuration screens.

ASCII ▪ An acronym for *American Standard Code for Information Interchange;* a seven-bit code used to represent letters of the alphabet, the ten decimal digits, and other instructions used to edit text on a computer, such as Backspace, Carriage Return, Line Feed, and so on.

B

bulletin board system ▪ A computer dedicated to maintaining messages and software and making them available over telephone lines. People *upload* (contribute) and *download* (gather) messages by calling the bulletin board from their own computers. Abbreviated BBS.

button ▪ A pushbutton-like image that you click to start an *action*. The *action* runs a single tool or a series of processes.

byte ▪ A piece of computer data composed of eight contiguous bits that are grouped together as a single unit.

C

CAE ▪ An acronym for *computer aided engineering*.

check box ▪ A small square button: ❑. Check boxes are used in lists of options when more than one option can be selected at a time.

complex hierarchy ▪ A design in which two or more sheet symbols reference a single worksheet. Compare with *simple hierarchy*.

configuration ▪ The information a button or tool set uses to operate. Configurations can be tailored to your needs. The configuration for a tool set applies to all tools in the set. Local configuration applies to the tool (or processes) the button runs.

connectivity database ▪ The *connectivity database* consists of the incremental connectivity database (created by INET) and the linked connectivity database (created by ILINK). It describes the connectivity of a design, and is used to transfer a design to **Digital Simulation Tools** or **PC Board Layout Tools.** See *incremental connectivity database* and *linked connectivity database*.

cursor ▪ A marker inside an entry box showing where characters typed on the keyboard will display. The cursor for insert mode is a heavy underline, and the cursor for overtype mode is a square. See *pointer*.

D

default ▪ A preselected parameter.

design ▪ *noun:* A set of plans for electronic circuitry.

design cycle ▪ The process of conceiving, developing, testing, and producing a circuit.

digital ▪ Circuitry where data in the form of digits are produced by binary on and off or positive and negative electronic signals.

E

EDA ▪ An acronym for *electronic design automation.*

editor ▪ A tool used to create or modify a design file.

entry box ▪ A box indicating that something (text or numbers) should be entered using the keyboard: $\boxed{}$

F

flat design ▪ A schematic structure in which output lines of one sheet connect laterally to input lines of another sheet through graphical objects called *module ports.* Flat designs are practical for small designs of three or fewer sheets. See *module port, schematic, hierarchical structure.*

H

hierarchical design ▪ A schematic structure in which sheets are interconnected in a tree-like pattern vertically and laterally. At least one sheet, the root sheet, contains symbols representing other sheets, called subsheets.

I

incremental connectivity database ▪ INET produces the *incremental connectivity database.* It consists of an incremental connectivity database file (.INF) for each sheet in the design and an .INX file. The .INF file is a description of connectivity on each sheet. The .INX file lists each sheet referenced in the design. ILINK uses the *incremental connectivity data base* to create an incremental netlist. See *connectivity database* and *incremental netlisting.*

incremental netlisting ▪ A method of creating a netlist in which only changed worksheets are processed each time **Create Netlist** or **Create Hierarchical Netlist** is run.

initial macro ■ A macro that runs automatically whenever you run **Draft** or **Edit Library**. For the initial macro to work, you must configure **Schematic Design Tools** to load a macro file containing the desired macro definition.

intermediate netlist structure ■ ILINK produces the *incremental netlist structure*. This consists of the .INS (instance) file, the .RES (resolved) file, and the .PIP file (contains pipe link commands). These files are used by IFORM to create a netlist in one of over 30 formats.

K

K ■ A unit of measurement. 1K byte is equal to 1024 bytes. The "K" is taken from the metric system, where is stands for "kilo," or 1000. 1024 is 2^{10} and is close to 1000.

key field ■ To tell **Draft** and other tools which fields you want to combine and compare, *key fields* are used. A key field lists the part fields to combine and compare. Key fields are defined on the **Configure Schematic Design Tools** screen.

L

library ■ A collection of standard, often-used part symbols stored as templates to speed up design.

librarian ■ A tool used to manage or create library parts.

linked connectivity database ■ ILINK can optionally be configured to create the *linked connectivity database*. This ASCII file has an extension of .LNF and is used to transfer to **PC Board Layout Tools**.

list box ■ A box on local configuration screens and in windows that lists files in specific designs or directories. You move through the list using scroll buttons next to the list box. On local configuration screens, you can specify a wildcard so the list box contains files that match the criteria you specify.

local configuration ■ Configuration settings for a particular button. If the button runs several processes, each process can be configured locally. One tool can have different configurations in different buttons. For example, **Annotate** is configured differently under the **To Layout** button and under the **To Digital Simulation** button.

M

MB ■ An abbreviation for *megabyte*. See *megabyte*.

macro ■ Series of commands you can execute automatically at the touch of a single key. Macros dramatically reduce the number of keystrokes required to perform complex or repetitive actions.

megabyte ■ Slightly more than one million bytes; 10 megabytes equals 10 million bytes. A megabyte is equal to 2^{20} bytes (1,048,576). "Mega" is taken from the metric system, where it is a prefix meaning one million.

module port ▪ Graphical objects that conduct signals between schematic worksheets. See *flat structure*.

N

net ▪ Just as signals are conducted between schematic worksheets through module ports, they are conducted into and out of sheet symbols through graphical objects called *nets*.

netlist ▪ An ASCII file that lists the interconnections of a schematic diagram by the names of the signals, modules, and pins connected together on a PCB. The nodes in a circuit. See *incremental netlisting*.

P

pan ▪ To change the portion of the worksheet being viewed by dragging the pointer from one location on the worksheet to another location. As you drag the pointer, the worksheet *pans* across the screen.

part ▪ A schematic symbol that represents an object. The object can be either a part or another worksheet.

part field ▪ A slot for holding text or data to be associated with a part. Each part has two part fields reserved for part value and part reference. It has eight other part fields that can be used to store other useful information. See *key fields*.

PCB ▪ An acronym for *printed circuit board*.

PLD ▪ An acronym for *programmable logic device*. See *programmable logic device*.

pointer ▪ An arrow on the screen that moves as you move the mouse: ↖ See *cursor*.

processor ▪ A tool that subjects a design file to a specific process.

programmable logic device ▪ A type of integrated circuit that contains fuses that can be blown, eliminating certain logical operations in the device and leaving others intact, giving the device one of many possible logical architectures or logical configurations.

prompt ▪ A query from a program asking you to enter specific information.

R

radio button ▪ A small round button: O. Radio buttons are used in lists of mutually exclusive options: only one button can be active at a time.

reporter ▪ A tool that creates a report, but does not modify design data.

root directory ▪ The main directory on your computer; the directory that the computer boots from.

root sheet ▪ The worksheet at the top of a flat or hierarchical design. A design has only one root worksheet.

S

schematic ▪ A graphical representation of a circuit using a standard set of electronics symbols. See *flat design*, *hierarchical design*, and *root sheet*.

scroll buttons ▪ Buttons used to move a directory in its window so that a different part is visible. The four scroll buttons are:

- Page Up
- Line Up
- Page Down
- Line Down

sheet symbol ▪ Block-shaped symbols representing other worksheets. Signals are conducted into and out of sheet symbols by graphical objects called nets. See *nets*.

simple hierarchy▪ A one-to-one correspondence between sheet symbols and the schematic diagrams they reference. Each sheet symbol represents a unique subsheet. See *hierarchical design*.

syntax ▪ The formal structure of a language. Syntax includes the rules for making statements in the language, but excludes the meanings of the statements.

T

tag ▪ A marked or saved location on a schematic or layout. You can use the **JUMP** command to go to a tag.

template ▪ A set of patterns used to create new designs. The template is *not* a design itself.

text export ▪ The process of copying text from a schematic worksheet to a text file.

text import ▪ The process of copying text from a text file to a schematic worksheet.

TTL ▪ An acronym for *transistor transistor logic*.

tool ▪ A tool is a computer program you can use to do some useful task. Tools are grouped into five categories: editors, processors, reporters, librarians, transfers.

tool set ▪ A collection of tools designed to perform a set of electronic design automation tasks. OrCAD tool sets include: **Schematic Design Tools, Programmable Logic Design Tools, Digital Simulation Tools,** and **PC Board Layout Tools.**

transfer ▪ A tool that transfers design information from one tool set to another tool set. Also runs whatever processes are necessary to go from one tool set to another.

U

upload ▪ The process of sending a file to another computer.

user button ■ A button that you can program to perform whatever combination of functions you find useful (such as executables or batch files). User button programs are saved with the design files, so you can create design-specific buttons and not worry about overwriting user button programs for other designs.

W

wildcard ■ A series of characters you specify in a **Wildcard** entry box to filter the files that display in a list box. For example, *.* filters nothing from a list of files.

worksheet ■ **Draft** calls the sheets of drafting paper on which the schematics are drawn *worksheets.* Worksheets appear on the computer screen as a rectangular area in which you can place parts and draw wires.

Z

zoom ■ The ability to change the view on the screen by making the objects appear larger or smaller.

A

B

C

F

G

H

I

O

P

Schematic Design Tools

User's Guide

OrCAD®

Electronic Design Automation Tools

Schematic Design Tools

User's Guide

McBoole is a public domain process developed by Michel Dagenais of McGill University.

Sixth Edition 15 Jul 92

OrCAD®

CONTENTS

Welcome to OrCAD Schematic Design Tools

Welcome to practical electronic engineering. You now own **OrCAD Schematic Design Tools,** a design automation tool set with the power of an engineering workstation. You can complete complex design tasks using **Schematic Design Tools** in a fraction of the time it takes by hand.

Developed specifically to run on personal computers, **Schematic Design Tools** supports most popular graphics boards, printers, and plotters.

Finding the information you need

These five manuals accompany **Schematic Design Tools:**

- *Installation & Technical Support Guide*
- *OrCAD/ESP Design Environment User's Guide*
- *Stony Brook M2EDIT Text Editor User's Guide*
- *Schematic Design Tools User's Guide*
- *Schematic Design Tools Reference Guide*

Installation

Before you begin to explore **Schematic Design Tools,** take a few minutes to install the tool set and register for technical support. Just follow the instructions in the *Installation & Technical Support Guide.*

Project-oriented design environment

Schematic Design Tools is one part of a fully integrated electronic design automation (EDA) system. The design environment is structured so you can focus on what's important: the design. Designs are organized by project, with all the design files—schematics, netlists, part lists, simulation results, and board layouts—stored together.

The *OrCAD/ESP Design Environment User's Guide* introduces the graphical environment under which **Schematic Design Tools** and the other OrCAD tool sets operate. In this environment, OrCAD tools and tool sets, such as **Schematic Design Tools**, 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 main screen, even if you only have one tool set installed on your computer.

Learning Schematic Design Tools

This user's guide introduces **Schematic Design Tools**. The best way to get to know **Schematic Design Tools** is to start with *Chapter 2: Introducing Draft*, then proceed chapter by chapter through this book. You will be guided through several practice sessions that show you the basics about using **Schematic Design Tools**.

Beyond the basics

Once you master the basics, refer to the *Schematic Design Tools Reference Guide* for information that will help you plan and create your design. The reference guide explains how to tailor the configuration of the software to match your personal requirements, provides detailed information about commands and concepts, and tells how to transfer a design between OrCAD tool sets. It is designed to be a continuing source of instruction and reference as you use **Schematic Design Tools**.

The design environment

Schematic Design Tools is one part of a fully integrated electronic design automation environment. Using this design environment you can:

- Run the tools within a tool set. The tools that make up **Schematic Design Tools** are listed in the next section.
- Move between tool sets without switching directories or copying files.
- Configure 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.
- Organize designs by project. All files associated with a design—schematics, netlists, reports, PLD source code, simulation results, and 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 **Schematic Design Tools** screen.

These functions are described briefly on the pages that follow. The explanations assume that you are already familiar with common electronic design terms and concepts. If you are just learning about schematic 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 2 through 8 of this guide. Advanced concepts are fully explained in the *Schematic Design Tools Reference Guide*.

You can run all OrCAD tools on a single worksheet or on a multiple-sheet design. Multiple-sheet designs can be either flat designs or hierarchical designs. To learn about these different types of files, see the *Design structures* section later in this chapter.

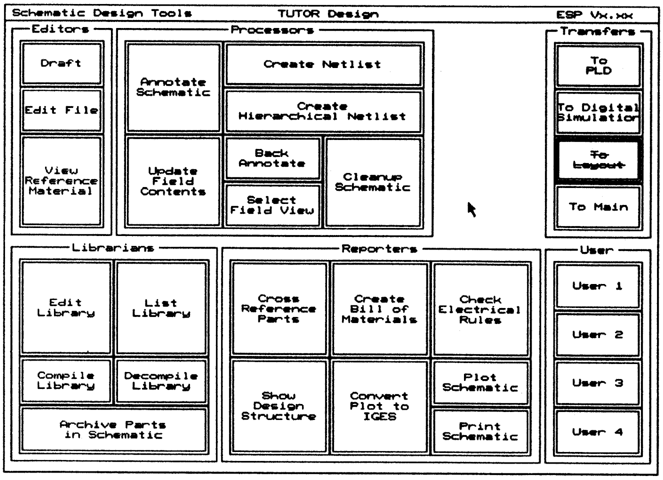


Figure 1-1. ***Schematic Design Tools*** *screen.*

Editors

Editors create or modify design files. The three editor tools in **Schematic Design Tools** are:

- ❖ **Draft.** The heart of **Schematic Design Tools** is the schematic editor, **Draft. Draft** is used to create schematics, which are part of the design database.
- ❖ **Edit File.** This text editor is used to create and edit text files.
- ❖ **View Reference.** This tool runs a text editor in a reference-material directory provided by OrCAD. You can view files about drivers, libraries, netlist formats, and other topics of interest.

Processors

Processors are tools that subject a design file to a specific process. The seven processors in **Schematic Design Tools** are:

- **Annotate Schematic**. This tool automatically updates part reference designators (such as U?, R?). It also updates the pin numbers associated with the reference designators in multiple-element parts. **Annotate Schematic** can handle very large, complex, multiple-sheet designs. It can update incrementally (leaving previously assigned reference designators alone) or unconditionally.
- **Create Netlist**. This tool creates a text file listing the logical interconnections between signals and pins. When the design becomes a real circuit board, the netlist turns into patterns of physical connections called tracks and nets. **Create Netlist** creates a netlist in one of over 30 different formats. See *Appendix B: Netlist formats* in the *Schematic Design Tools Reference Guide* for a list of available formats.

  You can also create your own netlist formats. See *Appendix D: Creating a custom netlist format* in the *Schematic Design Tools Reference Guide* for instructions.

  Create Netlist also creates the connectivity database. The connectivity database is used when transferring to OrCAD's **Programmable Logic Design Tools** and **Digital Simulation Tools**.
- **Create Hierarchical Netlist**. This tool operates similarly to the **Create Netlist** tool, only it is used on hierarchical designs. Hierarchical designs are discussed later in this chapter.

- **Update Field Contents**. This tool updates part value and part fields. Every part has ten fields that are used to hold text or data associated with the part. One data field holds reference designator values, such as "U1A" and "Q1." Another holds the part's name, such as "74LS04" or values relevant to the part, such as ohm (Ω) values for resistors. The other eight data fields can store any information you might find useful: part tolerance, vendor name, part number, and so on.

  Update Field Contents can change information in all but the reference designator field. It changes fields based on the contents of an update file. You create the update file using **Edit File**.

- **Back Annotate**. This tool updates part reference designators by using a list of old and new reference designators called a Was/Is file. You create the Was/Is file using **Edit File**.

- **Cleanup Schematic**. This tool checks to see if any wires, buses, junctions, labels, module ports, or other objects have been placed on top of one another.

- **Select Field View**. This tool makes the contents of a data field either visible or invisible on the schematic.

Librarians

Schematic Design Tools includes libraries containing more than 20,000 parts. These parts represent TTL, IEEE, CMOS, memory, ECL, discrete, analog, microprocessor, and peripheral devices.

Schematic Design Tools includes five librarians, which are tools for managing and creating library parts. Three of these tools work directly on libraries:

- **List Library**. This tool lists all the parts in a library.
- **Archive Parts in Schematic**. This tool scans a single worksheet or an entire design and collects all the parts used. It then creates a library file containing those parts.
- **Edit Library**. This tool is a graphical editor for creating or modifying library parts. You can save an edited part in a new or existing library.

You can also create or modify library parts with a text editor, such as **Edit File**. These two tools convert libraries from source form (text) to compiled form and vice versa:

- **Compile Library**. This tool converts a library source file into a compiled library object file. The compiled library object file can be used by the other **Schematic Design Tools**.
- **Decompile Library**. The inverse of the **Compile Library** tool, this tool converts a compiled library object file into a library source file. You can edit the library source file using **Edit File**.

Reporters

Reporters produce reports but do not modify design data. The seven reporter tools in **Schematic Design Tools** are:

- **Cross Reference Parts**. This tool scans specified schematic files, gathers information for all the parts used in the schematic files, and creates a report that lists each part's location in the design.
- **Create Bill of Materials**. This tool creates a summary list, sorted by reference designator, of all the parts used. You can also merge additional information into the summary list by using an include file.
- **Check Electrical Rules**. This tool checks for conformity to basic electrical rules. It checks for shorts, inputs with no driving source, unconnected pins, bus contention, and other common electrical hookup problems.
- **Show Design Structure**. This tool scans a hierarchical design and displays the design's root worksheet filename and all the associated sheet names. The filename of each sheet is also listed.
- **Convert Plot to IGES**. This tool translates a schematic plot file (created by **Plot Schematic**) into IGES (Initial Graphics Exchange Specification) text format. This common data format allows schematic plot files to be stored on a mainframe computer or used with other applications that accept IGES input (such as VersaCAD®.)
- **Plot Schematic**. This tool plots a single schematic or all the schematics in a design. It produces high-resolution plots of your designs.

  Devices that accept *vector* commands are considered to be plotters. A vector is a series of points with a specifically defined function.
- **Print Schematic**. This tool prints a single schematic or all the schematics in a design. It produces draft-quality printouts of your designs.

  Devices that accept *raster* commands are considered to be printers. A raster is an array of dots.

Transfers

Three of the transfer tools perform the steps needed to prepare a design for use by another OrCAD tool set. The **To Main** transfer tool simply displays the main screen. The four transfer tools in **Schematic Design Tools** are:

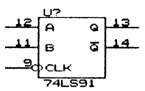
- ❖ **To PLD**. This tool updates part value and part fields, annotates the reference designators, extracts the **PLD** information, and displays the **Programmable Logic Tools** screen.
- ❖ **To Digital Simulation**. This tool annotates the reference designators, builds a connectivity database, creates trace and stimulus files, and displays the **Digital Simulation Tools** screen.
- ❖ **To Layout**. This tool updates part value and part fields, annotates the reference designators, builds a connectivity database, and displays the **PC Board Layout Tools** screen.
- ❖ **To Main**. This tool displays the main screen.

Graphic objects

Schematics are made up of a variety of graphic objects. You can include any of the following graphic objects in your schematic designs:

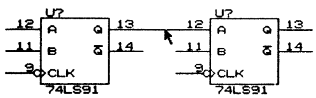
- Parts
- Wires
- Buses
- Junctions
- Power objects
- Module ports
- Sheet symbols
- Labels
- Text
- Title block
- Stimulus objects
- Vector objects
- Trace objects
- Layout objects

Parts



Parts are graphic objects you place on the schematic worksheet to represent the electronic parts in your design.

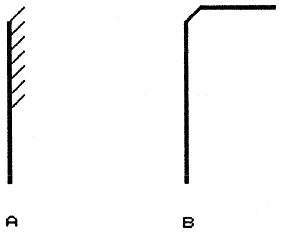
Wires



Wires are graphic objects you place on the worksheet to represent connections between objects, such as pins of parts and power objects.

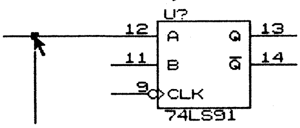
Buses

Buses are graphic objects that represent arrays of signals as single units on your worksheet.



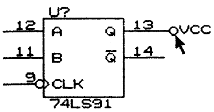
Junctions

Junctions are graphic objects that represent physical connections between wires, buses, and nodes. Junctions look like small squares.



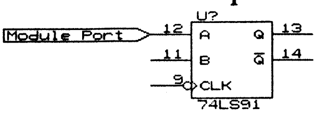
Power objects

Power objects are graphic objects that represent connections to a power source.



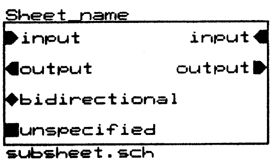
Module ports

Module ports are graphic objects that indicate where signals are conducted between worksheets.

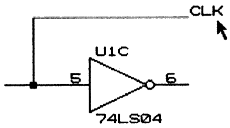


Sheet symbols

Sheet symbols are block-shaped symbols representing other worksheets. Each sheet symbol represents a subsheet. Sheet symbols are only used in hierarchical designs.

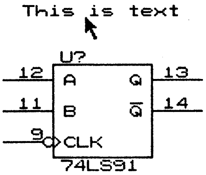


Labels



Labels identify the locations of signal connections without actually showing the physical connections on the worksheet.

Text



You can also place *text* in your worksheet. Text is used to leave notes or descriptive text (that isn't required by the circuit) on a schematic diagram. Such text helps you and others understand the functions being performed or documents some aspect of circuit operation.

Title block

The *title block* is used to label your worksheets so that you can tell them apart. It contains information such as company name and address, schematic title, number, size, and revision.

Stimulus objects

OrCAD's **Digital Simulation Tools** uses *stimulus objects* to determine where a stimulus is to be applied to a circuit.

Vector objects

OrCAD's **Digital Simulation Tools** uses *vector objects* to determine where sets of stimuli are to be applied to a circuit.

Trace objects

OrCAD's **Digital Simulation Tools** uses *trace objects* to determine which signals to trace.

Layout objects

OrCAD's **PC Board Layout Tools** uses *layout objects* to get information about particular signals such as track width, via size, routing layer, and so forth.

Editing schematic diagrams

As its name suggests, **Draft** is designed to be analogous to the schematic design tools with which you are already familiar: drafting board, pencil, sheets of paper, standard logic symbols and symbol templates, and so on.

Draft is designed to support the complete design process from the conception of a design to the final sets of detailed schematic diagrams.

The computer screen represents the drafting table. The pointer does what a pencil does, and more. Drawing and erasing are done using **Draft** commands.

In **Schematic Design Tools**, the sheets of drafting paper on which the schematics are drawn are called *worksheets*. Worksheets appear on the computer screen as a rectangular area in which you can place parts and draw wires.

When you save the work you have done on a worksheet, **Schematic Design Tools** stores the information on the computer's disk as a data file. The name of the worksheet is the name of the file in which it is saved. Worksheets are stored inside designs. A *design* is a directory that contains all of the files (including the worksheet) that are part of the design process. All designs are contained in the \ORCAD directory.

Schematic Design Tools saves the worksheet in the design in which you are working. The worksheet can have the design name and an extension of .SCH, or you may give it different name.

For example, if you have a design called TUTOR, the default pathname for the TUTOR schematic is \ORCAD\TUTOR\TUTOR.SCH.

Design structures

Some designs are small enough to be represented entirely on a single schematic worksheet. **Schematic Design Tools**'s standard page sizes correspond to the five standard sheet sizes for plotters and printers (A through E for English, and A4 through A0 for metric). You can also create custom page sizes up to 65 inches square.

However, some designs are too large for even the biggest sheet and, even if a very complex design could fit on one sheet, there are good reasons for dividing it up:

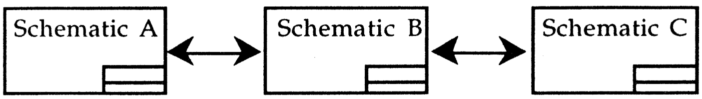
- To partition a design so that several people can work on it at once.
- To develop the design using a top-down approach. That is, you may want to begin with a block diagram in which each block represents a major function, and then construct more detailed diagrams for each of the blocks.
- To organize your design by functional parts.
- To maximize the performance of your tools.

Schematic Design Tools offers two ways of handling multiple-sheet designs: flat designs and hierarchical designs.

Flat designs

Best suited for small designs with no more than ten sheets, flat designs laterally connect the output signals from one schematic to the input signals of another.

All schematics in a flat design are on a single level, as shown below.



Since you must manage all of the interconnections between the sheets of a flat design by the names assigned to the module ports, it is best to keep a flat design relatively small.

Module ports

Module ports that have identical names on both schematics are considered to be electrically connected.

Figure 1-2 shows an example of connections between schematics in a simple two-sheet flat design.

The module ports in figure 1-2 that connect between the schematics are named CLEAR, LOAD, and RCO. The module ports named Hi[0..3] and Lo[0..3] are not connected.

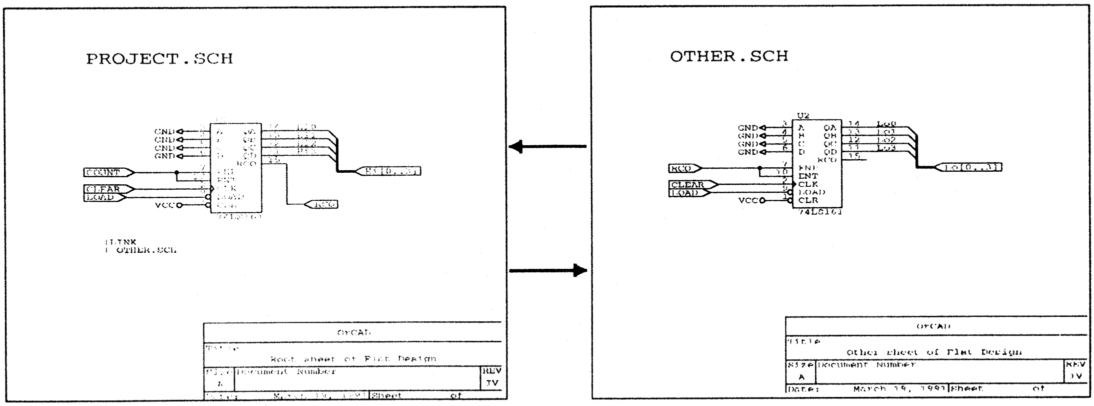
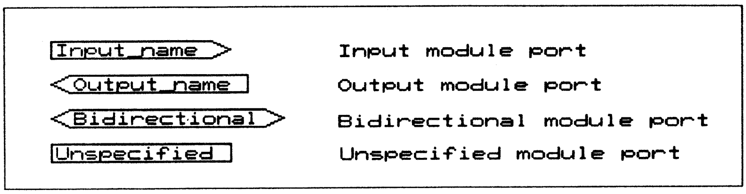


Figure 1-2. Schematics linked by module ports.

Figure 1-2 shows only input and output module ports, and connections between single wires. **Draft** has two other types of module ports: bidirectional and unspecified. You can use module ports to connect buses, as well as single wires. The four types of module ports are represented as follows:



You place module ports on a schematic using **Draft's PLACE Module Port** command.

|LINK command

Module ports indicate the names of the signals to be connected but do not specify which worksheets are to be included in the design. Therefore, flat designs must have one other component: a list of the worksheets in the design.

This list of worksheets appears on the root worksheet and consists of the "pipe" character (the vertical bar on your keyboard) followed by the keyword "LINK," followed by subsequent lines consisting of the pipe character and the filenames of the worksheets to be linked to the root worksheet.

```
|LINK
|SCHEM1.SCH
|SCHEM2.SCH
|SCHEM3.SCH
```

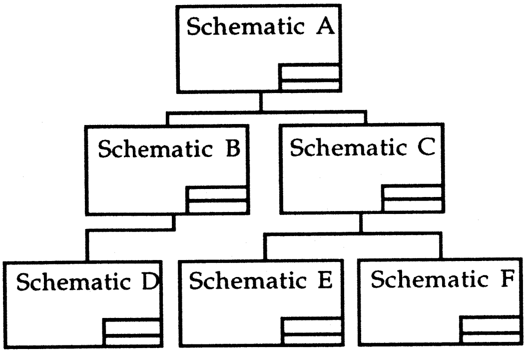
You place the worksheet list on the root worksheet using **Draft**'s **PLACE Text** command. The example at right shows text as it appears on a root worksheet that has module ports that link to worksheets called SCHEM1.SCH, SCHEM2.SCH, and SCHEM3.SCH. This text can appear anywhere on the worksheet.

Notice the |LINK (read as "pipe-link") command on the PROJECT.SCH worksheet in figure 1-2.

△ *NOTE: For details about module ports, see the* ***PLACE Module Port*** *command in the* Schematic Design Tools Reference Guide. *For details about placing text on a worksheet, see the* ***PLACE Text*** *command in the* Schematic Design Tools Reference Guide.

Hierarchical designs

As an alternative to the flat design, you can create schematics that contain symbols representing other schematics. These symbols are called *sheet symbols*. The layered arrangement created by placing schematics inside other schematics is called a *hierarchy*. Any hierarchy—whether it is a schematic design or a corporate organizational chart—has "higher" and "lower" levels.



Any schematic can contain sheet symbols that refer to other schematics, and this nesting structure can be made many levels deep. The schematic at the top of a hierarchy, which directly or indirectly refers to all other schematics in the design, is called the *root worksheet*.

You place sheet symbols on a schematic using **Draft**'s **PLACE Sheet** command.

How signals enter and leave sheet symbols

Just as module ports indicate which signals connect between schematics, *sheet nets* indicate which signals connect between a sheet symbol and its associated schematic. In figure 1-3, sheet nets are the small black objects shown on the borders of the sheet symbols. The sheet nets on a sheet symbol correspond to module ports on the schematic named in the sheet symbol.

You place sheet nets using **Draft**'s **Add-NET** command, which becomes available when you select the **PLACE Sheet** command. To associate a particular sheet net with a particular module port, assign the same name to both.

The bracketed notation (*A*[*m*..*n*]) shown on the module ports and sheet nets designates the number of signals carried by a bus. For example, X[0..3] indicates four signals: X0, X1, X2, and X3.

Buses must have a label or a module port with similar bracket notation to indicate the number of signals they carry, and wires connected to buses must have labels identifying the signal they carry. These details are shown in figure 1-3.

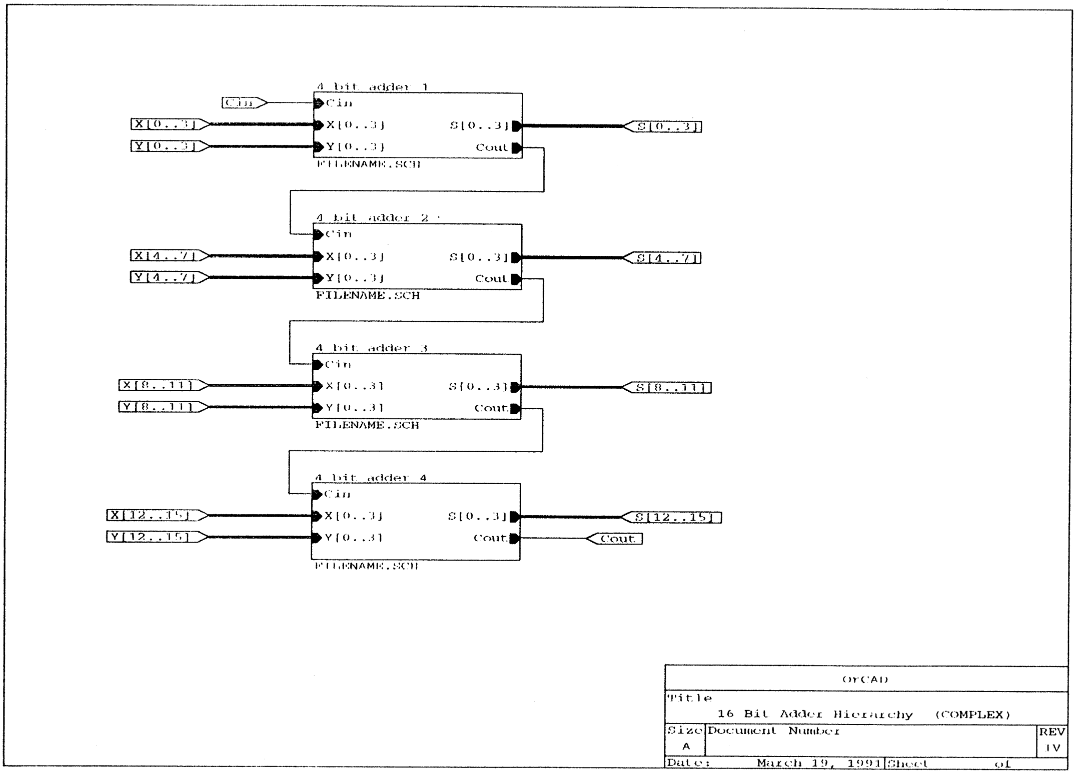


Figure 1-3. Hierarchical design.

Simple and complex hierarchies

A one-to-one correspondence between sheet symbols and the worksheets they refer to defines a structure called a *simple hierarchy.*

Several sheet symbols can refer to a single worksheet, resulting in a structure called a *complex hierarchy.* For example, figure 1-3 shows a complex hierarchy in which there are four references to a single worksheet called FILENAME.SCH.

How sheet symbols refer to schematic logic

To give a sheet symbol access to the logic of a particular worksheet, you assign the sheet symbol a worksheet's filename. The filename displays at the bottom of the sheet symbol, as shown in figure 1-3.

You assign a sheet symbol its corresponding worksheet's filename using the **Filename** command, which becomes available when you select the **PLACE Sheet** command.

In addition to their filename markers, sheet symbols can also have names. In figure 1-3, the sheet symbol names are shown just above each sheet symbol.

You name the sheet symbols using the **Name** command, which becomes available when you select the **PLACE Sheet** command.

Moving between levels in a hierarchy

Draft makes it easy to move up and down in a hierarchy, from sheet symbol to associated worksheet and back again.

To go from a sheet symbol to its associated worksheet, put the pointer on the sheet symbol and select **QUIT Enter Sheet**. To go from the schematic back to the schematic in which reference to it is made by a sheet symbol, select the **QUIT Leave Sheet** command.

More about hierarchical design structures

The worksheet represented by a sheet symbol can itself contain sheet symbols. This means you can create hierarchies that are many levels deep, each level containing greater detail.

This is particularly useful for very complex designs. It encourages a logical, function-oriented approach to partitioning designs and makes them easier to manage.

Another advantage offered by hierarchical structure is the ability to use sheet symbols to refer repeatedly to "stock" worksheets containing common circuit functions. This is often used in gate array and field-programmable gate array (FPGA) designs.

△ *NOTE: A deep hierarchy is much more efficient than a wide hierarchy. A wide hierarchy, while not a flat design, has many of the same limitations in organization, presentation, and structure. A deep hierarchy more clearly represents the functional nature of the design.*

For more information, study the hierarchy examples in *Chapter 8: Structuring your design.*

Learning Schematic Design Tools

The remainder of the *Schematic Design Tools User's Guide* shows how to design schematics by guiding you through the process of creating the schematic diagrams for a digital clock. To do this, you use the schematic editor called **Draft** to create the schematic of the clock circuitry. Within the schematic are three smaller circuits:

- A clock oscillator circuit
- A power regulator circuit
- A logic and display circuit

Each chapter builds on the skills and concepts from the previous chapter. As you complete each chapter, you create a series of working files.

The summaries that follow describe the design concepts and skills you learn in each chapter.

Chapter 2: Introducing Draft

This chapter introduces **Draft**, the **Schematic Design Tools** schematic editor. You learn how to run the design environment, run **Design Management Tools**, set up work conditions for **Draft**, run **Draft**, capture and save an initial macro, and save your work.

Chapter 3: Capturing the clock oscillator schematic

In this chapter you create (or *capture*) a small schematic and learn the basic procedures required for schematic capture. You learn how to get and place library parts, how to place wires, how to place junctions, and how to place labels and text.

Chapter 4: Capturing the power regulator schematic

In this chapter you capture a schematic that is slightly more complex than the previous schematic. You learn how to move a group of parts, delete a part, undo a delete operation, rotate a part, place a power symbol, set a tag, jump to a tag or a reference, and print a hard copy of the schematic.

Chapter 5: Creating a custom part

In this chapter you use **Edit Library** to define a custom part (a seven-segment display). You learn how to draw the part body, draw special shapes, use shading, add pins to the part body, add pin names, and save the new part in a library.

Chapter 6: Capturing the logic and display circuit schematic

In this chapter you capture the final portion of the digital clock schematic. You learn how to draw a repeatable portion of the schematic, make and place multiple copies of it, write and use a macro, and use repeat parameters to place wires and labels.

Chapter 7: Using other Schematic Design Tools

This chapter introduces you to some of the other tools included in **Schematic Design Tools**. You learn to use **Annotate Schematic, Check Electrical Rules, Create Netlist, Back Annotate, Create Bill of Materials,** and **Plot Schematic.**

Chapter 8: Structuring your design

This chapter describes and reviews a complex hierarchy and shows how to convert a complex hierarchy to a simple hierarchy. Flat designs and how to link sheets are also reviewed.

Chapter 9: Tips and techniques

This chapter provides a collection of tips and techniques that you can use to enhance your ability to use **Schematic Design Tools** productively. This chapter does not follow the tutorial style of the other chapters.

Introducing Draft

In this chapter, you establish **Draft**'s work conditions. You learn how to:

- Run **ESP**
- Run **Design Management Tools**
- Set up work conditions for **Draft**, the schematic editor
- Run **Draft**
- Capture and save an initial macro
- Save your work

Before you begin

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

Keys

Schematic Design Tools is designed to operate on a wide variety of 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 Enter, New Line, Next, Return, or Send.

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, and 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.

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.

Keyboard input

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

```
tutor.sch
```

```
load file? tutor.sch
```

In the last example, you enter only the characters shown in bold. The nonbold characters show what the computer displays.

Operating system prompt

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

```
C:>
```

Callouts

In the later chapters of this guide, callouts such as ①, ②, and ③ appear on schematic diagrams. These callouts refer to the corresponding step numbers in the instructions.

Commands

In this guide, 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 kept in one directory on your disk. Putting different designs in different directories lets you organize your files, much as you would organize a file cabinet.

By following the steps in this tutorial, you will work on a design named "TUTOR." All of the files for this design are contained in the directory named "TUTOR." Files in the directory have the filename "TUTOR" and an extension that indicates the type of file. For example, the TUTOR schematic worksheet that you create in chapters 2 through 6 is named TUTOR.SCH.

Running the design environment

To run an OrCAD tool, you must first display the main screen.

1. Be sure that your computer is turned on.
2. At the operating system prompt, enter the command shown in bold:

```
C:> orcad
```

In a moment, the main screen displays (figure 2-1).

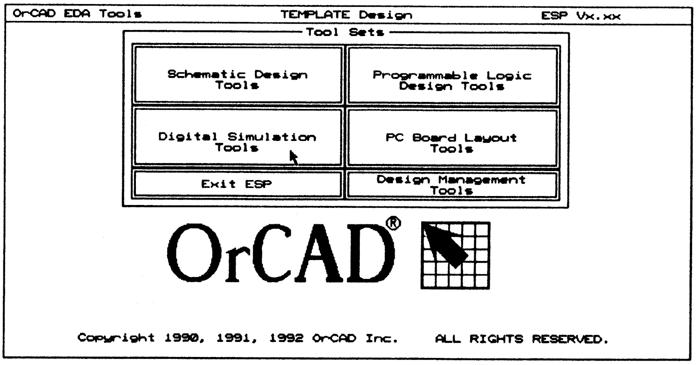


Figure 2-1. Main screen.

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.

Run Design Management Tools

Follow these steps to run **Design Management Tools** and change to the TUTOR design:

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

```
Design Management Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```

2. Click the left mouse button to select the **Execute** command. The screen shown in figure 2-2 displays.

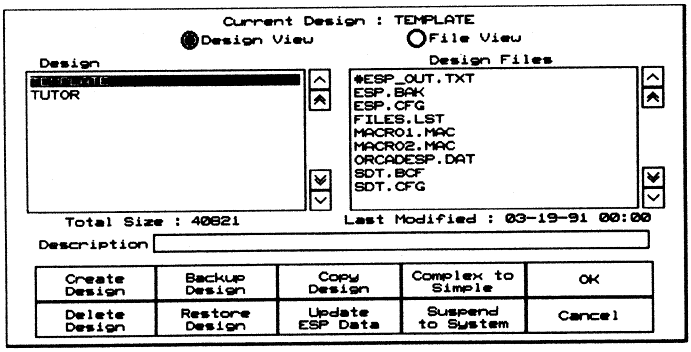


*Figure 2-2. **Design Management Tools** screen, Design View.*

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: Refer to the* OrCAD/ESP Design Environment User's Guide *for instructions on how to use **Design Management Tools**.*

Change the start-up design

The design environment is configured to start in the TEMPLATE design each time you run the design environment. Since you will be working in the TUTOR design throughout this guide, you need to change the start-up design to TUTOR. Follow these steps:

1. Select **Design Management Tools**. The menu at right displays.

```
Design Management Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```

2. Select **Configure ESP**. The screen in figure 2-3 displays.

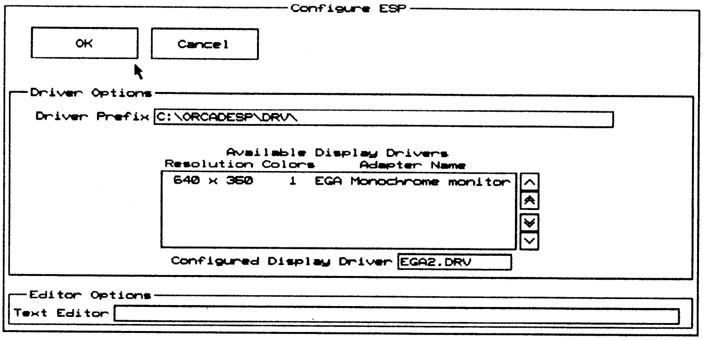


*Figure 2-3. Top portion of the **Configure ESP** screen.*

3. Move the pointer to the bottom of the screen. The display pans to show more of the **Configure ESP** screen. Continue panning until you reach the **Design Options** area.

```
Design Options
Startup Design  TEMPLATE
```

4. Place the pointer in the **Startup Design** entry box and click the left mouse button. The pointer becomes a cursor in the entry box. Delete `TEMPLATE`, and enter **TUTOR** as the start-up design.

```
Design Options
Startup Design  TUTOR
```

5. Move the pointer to the top of the screen and select **OK**. (An easy way to move to the **OK** button is to press the <Home> key.)

   The changes you made to the **Configure ESP** screen are saved and the main screen displays.

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

Running Schematic Design Tools

Follow these steps to display the **Schematic Design Tools** screen.

1. Select **Schematic Design Tools**. The menu at right displays.
2. Select the **Execute** command. The **Schematic Design Tools** screen displays (figure 2-4).

```
Schematic Design Tools
Execute
Local Configuration
Assign Hot Key
Configure ESP
Help
```

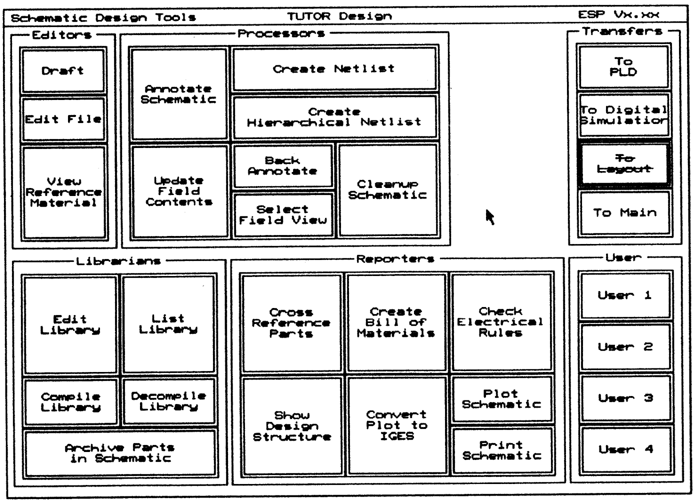


*Figure 2-4. **Schematic Design Tools** screen.*

Defining title block information

Before you run the schematic editor **Draft**, take a few minutes to configure the information to appear in the title block of the worksheet you will create. To do this, you must display the **Configure Schematic Design Tools** screen.

View the configuration for Schematic Design Tools

1. Select **Draft**. The menu at right displays.
2. Select the **Configure Schematic Tools** command.

```
Draft
Execute
Local Configuration
Assign Hot Key
Show Version
Configure Schematic Tools
Help
```

Figure 2-5 shows the top portion of the **Configure Schematic Design Tools** screen. The parameters you see may differ from those in the figure, because some of the configuration information depends on your system hardware. For more information about the **Configure Schematic Design Tools** screen, see *Chapter 1: Configure Schematic Tools* in the *Schematic Design Tools Reference Guide.*

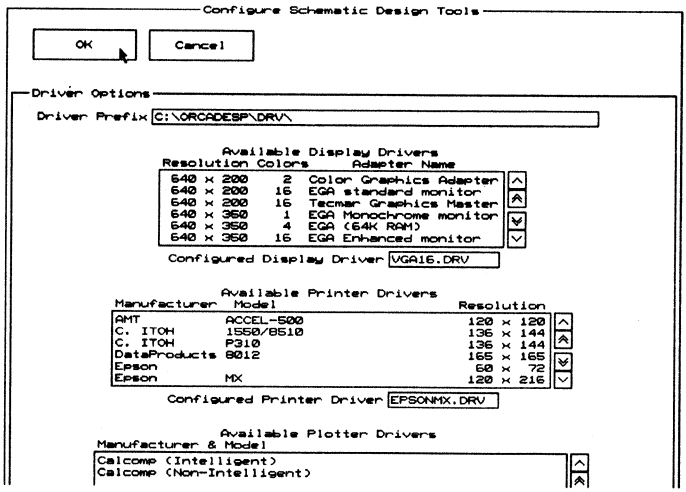


Figure 2-5. Top portion of the ***Configure Schematic Design Tools*** *screen.*

3. Scroll to the **Worksheet Options** area (figure 2-6).

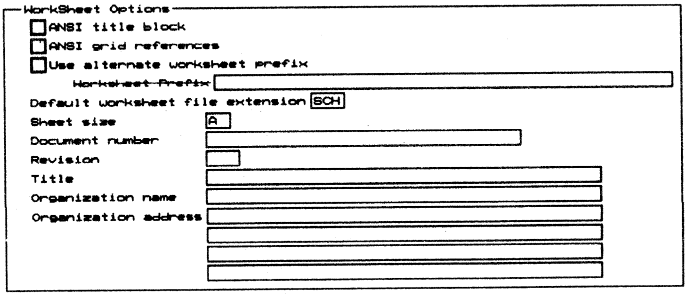


Figure 2-6. ***Worksheet Options*** *area of the* ***Configure Schematic Design Tools*** *screen.*

Notice the **Document number, Revision, Title, Organization name,** and **Organization address** entry boxes. Any information entered in these fields becomes a part of your worksheet's title block. For this tutorial, you enter information in the **Title, Organization name,** and **Organization address** entry boxes.

4. Position the pointer within the **Title** entry box and press <Enter>. Enter the title, `Digital clock schematic`.

5. Press <Tab> to move to the next entry box—in this case, **Organization name**—and press <Enter>. Enter the name of your organization.

6. Press <Tab> to move to the first entry box of **Organization address,** and press <Enter>. Enter the street address of your organization.

7. Press <Tab> to move to the second entry box of **Organization address,** and press <Enter>. Enter the city and state of your organization.

8. Move the pointer to the top of the screen and select **OK**. This updates the configuration and displays the **Schematic Design Tools** screen.

Running Draft

Now that you have changed the start-up design to TUTOR and set up your title block information, you are ready to begin learning about the schematic editor **Draft.**

1. Select **Draft.** The **Draft** menu displays.
2. Select the **Execute** command to run **Draft.**

   The top and left edges of the new worksheet display, as shown in figure 2-7. Because the screen is smaller than the worksheet, the right and bottom edges of the worksheet are not visible. You can think of the screen as a window into the larger worksheet area.

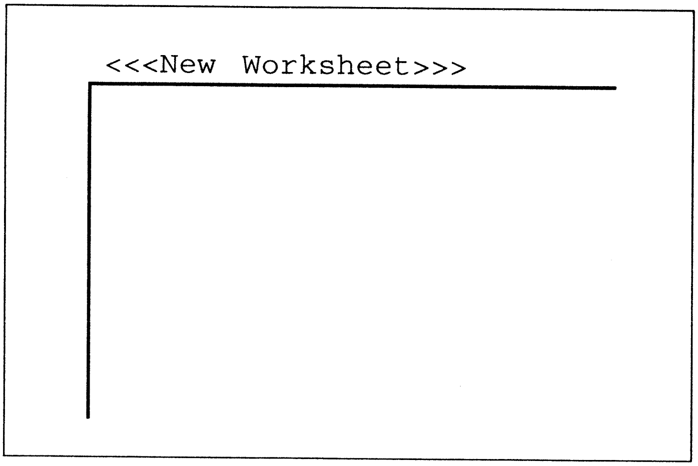


*Figure 2-7. New worksheet in **Draft**.*

Learning OrCAD basics

Pop-up menus guide you step by step through OrCAD software. **Draft** organizes commands using menus and command lines. You can select a command by either clicking the mouse or pressing a key. (For complete command descriptions, see the *Schematic Design Tools Reference Guide*.)

Main menu

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 remove the main menu from the screen.

To return to the main menu—no matter where you are in **Draft**—press <Esc> or click the right mouse button as many times as necessary until no menu or command line displays in the upper left corner of the screen, and then press <Enter> or click the left mouse button.

```
Again
Block
Conditions
Delete
Edit
Find
Get
Hardcopy
Inquire
Jump
Library
Macro
Place
Quit
Repeat
Set
Tag
Zoom
```

Commands

There are several ways to select a command. You can use the methods shown in table 2-1 in any combination. The method you use is a matter of personal preference.

	Using the keyboard	*Using the mouse*
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 first capital letter in the command name.	

Table 2-1. Selecting commands.

Draft responds to a command either by performing the command's function or by displaying another menu or a command line.

Menus

All menus look and work just like the main menu. **Draft** displays the menu name on the top line of the screen. Press <Esc> or 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 the **BLOCK** command. The menu at right displays.
3. Press <Esc> to dismiss the **BLOCK** menu.

```
Block
Move
Drag
Fixup
Get
Save
Import
Export
ASCII Import
Text Import
```

Command lines

Command lines are a series of commands listed across the top of the screen. When a command line displays, you can move the pointer around the working area or select a command (by pressing the first capitalized letter in its name). Press <Esc> or the right mouse button to return to the menu or command line that called the command line. Follow these steps to familiarize yourself with these processes:

1. Press <Enter> to display the main menu.
2. Select the **EDIT** command. The **EDIT** command line appears at the top of the screen:

```
Edit  Find  Jump  Zoom
```

3. Press <Esc> to dismiss the **EDIT** command line.

Returning to the main menu

To return to the main menu—no matter where you are in **Draft**—press <Esc> as many times as necessary until no menu or command line displays in the upper left corner of the screen, and press <Enter>.

How commands are shown in this guide

As described earlier in this chapter, commands are shown in **bold** type. Main menu commands are shown in uppercase letters. For example, the statement "Select the **PLACE Wire** command" means "Select the **PLACE** command from the main menu, and select the **Wire** command from the **PLACE** menu."

When you are asked to select a command, usually both the main menu command and other command are specified. 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 **Wire** command, the instruction is simply "Select the **Wire** command."

Setting up Draft's work conditions

Now that you understand how **Draft**'s commands, menus, and command lines operate, you will use the **SET** command to change the default work conditions that govern the way **Draft** displays and maintains schematics.

Display work conditions settings

Follow these steps to display the **SET** menu:

1. Press <Enter> to see the main menu.
2. Select **SET** from the main menu. The **SET** menu displays, as shown below.

```
Set
Auto Pan            YES
Backup File         YES
Drag Buses           NO
Error Bell          YES
Left Button          NO
Macro Prompts       YES
Orthogonal          YES
Show Pins           YES
Title Block         YES
Worksheet Size        A
X,Y Display          NO
Grid Parameters
Repeat Parameters
Visible Lettering
```

Using the commands in the **SET** menu, you can control features such as automatic backup of schematic files, the angles at which you can place wires, and whether or not pin numbers display on part symbols. For more information about **Draft**'s work conditions, see the **SET** command description in the *Schematic Design Tools Reference Guide*.

The next few sections describe a few of **Draft**'s work conditions and the commands controlling them.

Pan across the schematic

Auto Pan is the first command in the **SET** menu. When you start work on a new worksheet, **Auto Pan** is set to **Yes.**

When **Auto Pan** is set to **Yes**, the worksheet follows the movement of the pointer. If part of a worksheet is off the screen and you move the pointer beyond the edge of the display, the hidden part of the worksheet pans into view.

If you set **Auto Pan** to **No**, the screen does not pan. In this case, you must use the **JUMP** and **ZOOM** commands to see different parts of the worksheet.

Follow these steps to pan across the schematic:

1. Press <Esc> to dismiss the **SET** menu. **Auto Pan** remains set to **Yes**.

2. Move the pointer to the lower right corner until the title block displays. The screen pans to keep up with the pointer. Notice that the title block information you entered earlier in this chapter displays.

3. Move the pointer toward the upper left corner until the upper left corner of the worksheet displays.

Redisplay the SET menu

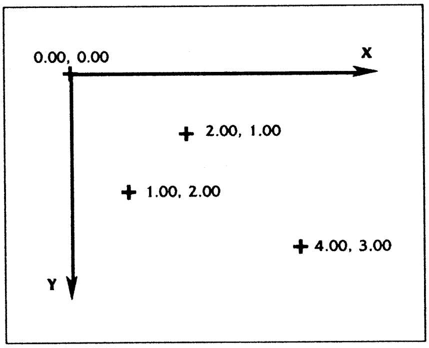
1. Press <Enter> to recall the main menu. When the menu displays, you'll see the highlight bar is on the **AGAIN** command.

2. Press <Enter> to select **AGAIN**. This selects the main menu command you chose last—in this case, **SET**.

Display X,Y coordinates

Draft uses a coordinate system to locate points on the worksheet, as shown in the illustration below.

An X coordinate specifies horizontal location and a Y coordinate specifies vertical location. Thus any point on the worksheet can be indicated by an X and Y coordinate pair in the form (X,Y). The (0.00, 0.00) point is always at the upper left of the worksheet.



If **X,Y Display** is set to **Yes**, the X and Y coordinates of the pointer's position display in the upper right corner of the screen. The default setting is **No**.

Follow these steps to display X, Y coordinates:

1. Select **X,Y Display**. "Display X,Y Coordinates of Pointer" and a short menu display.
2. Select **Yes**. The prompt and the menu disappear.
3. Move the pointer in any direction and watch the X,Y coordinates in the upper right corner of the screen.

The units shown in the X,Y display represent inches on the printed schematic. The upper left corner is (.00, .00) and the lower right corner is (9.50, 7.00). On a sheet 8.5 inches by 11 inches, the actual drawing area is 7 inches by 9.5 inches. This allows for borders around the drawings.

Select worksheet size

The **Worksheet Size** command selects one of five sizes for your schematic. Follow these steps to change the worksheet size:

1. Press <Enter> to display the main menu, then select **AGAIN**. The **SET** menu displays.
2. Select **Worksheet Size.** A menu lists the five options available for the size of a worksheet, as shown at right.
3. Select **C** size.
4. Move the pointer to the edges and corners of the worksheet to explore the size of the editable region of a C-size sheet. The dimensions shown in the **Worksheet Size** menu are the worksheet's borders. On a C-size sheet 22 inches by 17 inches, the actual drawing area is 20 inches by 15 inches.

```
Set Worksheet Size
(Area inside borders)
A   9.50 x  7.00
B  15.00 x  9.50
C  20.00 x 15.00
D  32.00 x 20.00
E  42.00 x 32.00
```

△ *NOTE: If* ***Schematic Design Tools*** *is configured to use metric dimensions, the* ***Set Worksheet size*** *menu displays the International Organization for Standardization (ISO) paper sizes: A4 through A0. In addition, the X, Y display is given in millimeters. For information about configuring* ***Schematic Design Tools*** *to use metric dimensions, see* Chapter 1: Configure Schematic Tools *in the* Schematic Design Tools Reference Guide.

Changing your view of the worksheet

Draft can display worksheets at five different scales. You change the view using the **ZOOM** command. The worksheet can be zoomed in or out to magnify or reduce its visible image.

When **Draft** is zoomed out, you can see a large portion of the worksheet. Zooming in enlarges a small portion of the worksheet and displays more details. You can zoom in to draw intricate portions of your worksheet with exacting detail and then zoom out to look at the whole schematic.

ZOOM in and out

Follow these steps to zoom out and see more of the worksheet on the screen at one time:

1. Move the pointer to lower right corner until the title block displays.

2. Select **ZOOM** from the main menu. The menu at right displays.

```
Zoom (present
scale=1)
Center      ( 1)
In          ( 1)
Out         ( 2)
Select
```

3. Select **Out**. A view of the worksheet at one-half the original scale displays.

4. Experiment with the scale using **In, Out,** and **Select.** If you use **Select** you can choose the scale at which to view the worksheet, as shown in the following figure.

```
Zoom - Select Scale
(present scale=1)
1
2
5
10
20
```

   If you choose **1,** you view the worksheet at full size. This shows the most detail ("zooms in" the closest). If you choose **2,** you view the worksheet at one-half the original scale. If you choose **20,** you view the worksheet at one-twentieth the original scale—you see the maximum working area and the least detail.

5. When you finish experimenting with zooming, return to full-size view (scale level 1).

Set grid parameters

While working on a large worksheet, it is useful to have visual cues that tell you approximately where you are on the sheet.

The **Grid Parameters** command on the **SET** menu sets up some of these visual cues. The **Set Grid Parameters** menu is shown at right.

```
Set Grid Parameters
Grid References     NO
Stay On Grid        YES
Visible Grid Dots   NO
```

Display grid references

Grid References turns grid reference guides along the top and left edges of the display on and off. The guides divide the worksheet into blocks. Horizontally, the guides divide the worksheet from 8 to 1. Vertically, they divide the worksheet from D to A. For example, the title block (lower right corner) is located at A-1, as illustrated in figure 2-8.

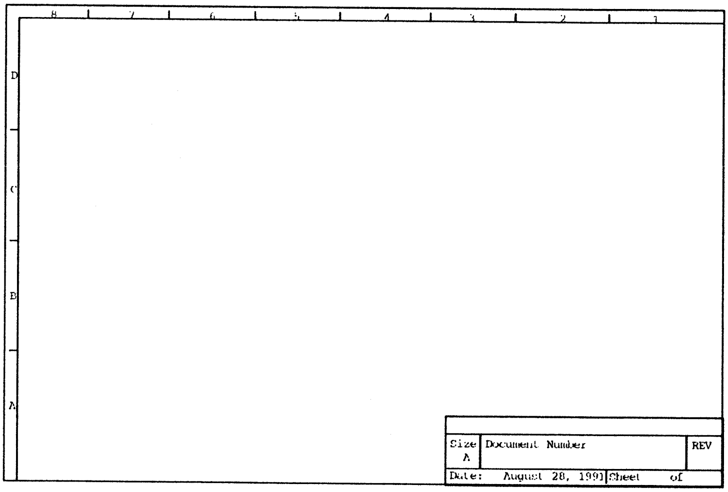


Figure 2-8. Grid references.

Use the **JUMP Reference** command on the main menu to move to specific locations using these map-like coordinates.

Follow these steps to display grid references:

1. Select **SET** from the main menu.
2. Select **Grid Parameters.**
3. Select **Grid References.**
4. Select **Yes.** The grid reference guides display at the top and left edges of the screen.

△ *NOTE: **Schematic Design Tools** can be set up to use ANSI Y14.1 drawing standards. Refer to the* Schematic Design Tools Reference Guide *for details.*

Stay on grid

Stay on Grid determines whether or not pointer movement is restricted to grid intersections. **Stay On Grid** is set to **YES.** Do not make any changes here.

△ *NOTE: Keep **Stay on Grid** set to **Yes** unless you have a compelling reason to be off-grid. Anything placed off-grid—such as text and labels—may be hard to select and edit later.*

Make the grid visible

Visible Grid Dots turns the dots representing intersections on and off. The space between the dots represents 0.1 inch on the printed worksheet. Follow these steps to make the grid dots visible:

1. Select **SET** and **Grid Parameters** again.
2. Select **Visible Grid Dots,** then select **Yes.** Grid dots display on the worksheet. You may want to adjust the intensity on your monitor to make the grid dots brighter or dimmer.

Updating the worksheet

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

Update the file

Follow these steps to save the worksheet without changing its filename:

1. Select **QUIT** from the main menu. **Draft** displays the filename and the **Quit** menu, as shown at right.
2. Select **Update File. Draft** saves the file.
3. Press <Esc> to dismiss the **Quit** menu.

```
Quit TUTOR.SCH
Enter Sheet
Leave Sheet
Update File
Write to File
Initialize
Suspend to System
Abandon Edits
Run User Commands
```

Creating a macro

Macros can record virtually anything you do in a program—so you can automate many repetitive tasks and speed up your work.

Capture a macro

Earlier in this chapter, you used the **SET** command to change work conditions. Follow the steps below to capture commands for setting work conditions in a macro.

△ *NOTE: This macro only works when you are at the main menu level of* ***Schematic Design Tools.***

1. Select **MACRO** from the main menu. The **MACRO** menu at right displays.
2. Select **Capture.** The prompt "Capture macro?" displays.

   Macros can be run by a single key or a combination of keys.

```
Macro
Capture
Delete
Initialize
List
Read
Write
```

Single keys that can run macros are the function keys (<F1> through <F10>) and special keys in the numeric keypad (such as <Home>, <PgUp>, and <PgDn>).

Key combinations that can run macros include:

- <Ctrl> with function keys
- <Ctrl> with alpha keys (except C, H, and M)
- <Alt> with function keys
- <Alt> with alpha keys
- <Shift> with function keys

If you choose a prohibited key combination, **Draft** displays "Key cannot be defined as macro" and displays the "Capture macro?" prompt again.

3. Press <Ctrl><A> to assign a keystroke to this macro. "^A" displays at the "Capture macro?" prompt.

4. Press <Enter>. The message "<macro>" displays to remind you that you are defining a macro. Any commands you select while "<macro>" displays are added to the list of commands stored in the macro.

5. Type the keys shown in the left column below. **Draft** performs the commands as it captures them.

Key sequence	*Command sequence*	*Effect*
<Enter>	———	Display main menu
SXY	**SET X,Y Display Yes**	Display coordinates
SGGY	**SET Grid Parameters Grid References Yes**	Display grid references
SGVY	**SET Grid Parameters Visible Grid Dots Yes**	Display grid dots
ZS1	**ZOOM Select 1**	Display schematic at full size

6. Press <Ctrl><End> to end the macro definition. **Draft** displays "<<<MACRO END>>>" to confirm that the macro definition is complete.

   The macro is now stored in the computer's memory. You can run it when you are at the main menu level of **Draft** by pressing the key combination you specified, <Ctrl><A>.

△ *NOTE: Some keyboards have two keys labeled <End>. If you press <Ctrl><End> and "<<<MACRO END>>>" does not appear, try using the other <End> key.*

Save the macro

1. Select **MACRO Write**. The prompt "Write all macros to?" displays.
2. Enter `tutor.mac`. **Draft** writes the macro to the TUTOR.MAC file in the TUTOR design directory.
3. To tell **Draft** to read macros from the TUTOR.MAC file, select **MACRO Read**. The prompt "Read all macros from?" displays.
4. Enter `tutor.mac`.
5. If you wish to test the macro you just saved, change some of the work conditions and press <Ctrl><A> to restore them.

Exiting Draft

You are nearly finished with this chapter. Follow these steps to exit **Draft**:

1. Select **QUIT** from the main menu. **Draft** displays the filename and the **QUIT** menu shown at right.
2. Select **Update File. Draft** saves the file.
3. Leave **Draft** by selecting **Abandon Edits. Draft** exits to the **Schematic Design Tools** screen.

```
Quit TUTOR.SCH
Enter Sheet
Leave Sheet
Update File
Write to File
Initialize
Suspend to System
Abandon Edits
Run User Commands
```

Setting up automatically

In addition to using **SET** to control **Draft**'s work conditions, you can automate the process of defining **Draft** work conditions by configuring **Schematic Design Tools** so that the macro you just created runs every time you run **Draft.** A macro that runs when the tool starts is called an *initial macro.*

1. Select **Draft** from the **Schematic Design Tools** screen.
2. Select **Configure Schematic Tools** from the menu that displays. The **Configure Schematic Design Tools** screen displays.
3. Scroll to the **Macro Options** area of the **Configure Schematic Design Tools** screen.
4. Position the pointer within the **Draft Macro File** entry box and press <Enter>. This entry box defines the name of the macro file you created earlier in this chapter.
5. Enter the macro path and filename:

   ```
   Draft Macro File  \orcad\tutor\tutor.mac
   ```

   Notice that the **Draft Initial Macro** entry box becomes accessible (not dimmed) when you enter text in the **Draft Macro File** entry box.
6. Position the pointer within the **Draft Initial Macro** entry box and press <Enter>. This entry box defines a macro that automatically runs when you run **Draft.**

   <Ctrl><A> is the keystroke that runs the macro. In the **Draft Initial Macro** entry box, however, you use a caret symbol (^) to represent the <Ctrl> key.
7. In the **Draft Initial Macro** entry box, simultaneously press <Shift> and <6> to enter the caret symbol (^), and then enter **A**. The entry box should look like this:

   ```
   Draft Initial Macro  ^A
   ```
8. Move the pointer to the top of the screen and select **OK.** This updates the configuration and displays the **Schematic Design Tools** screen.

△ *NOTE: Once you configure ^A in the* ***Draft Initial Macro*** *entry box on the* ***Configure Schematic Design Tools*** *screen, the macro runs automatically each time you run* ***Draft****. You can also run it when you are at the main menu level of* ***Draft*** *by pressing <Ctrl><A>.*

Summary

In this chapter you learned how to run **Draft** and examine and modify work conditions. You also learned how to capture an initial macro and have it automatically set the work conditions each time you run **Draft**.

The next chapter gives you instructions for capturing the schematic for the clock oscillator circuit. In later chapters, you build on the knowledge you gain while learning more about **Schematic Design Tools**.

Capturing the clock oscillator schematic

This chapter shows you the processes used to create a basic schematic. In this chapter, you learn how to:

- ❖ Get and place library parts
- ❖ Draw wires
- ❖ Place junctions
- ❖ Place labels and text

Running Draft

Figure 3-1 shows the schematic diagram of the clock oscillator circuit you create in this chapter. Refer to this figure for placement and orientation information while capturing the clock oscillator schematic.

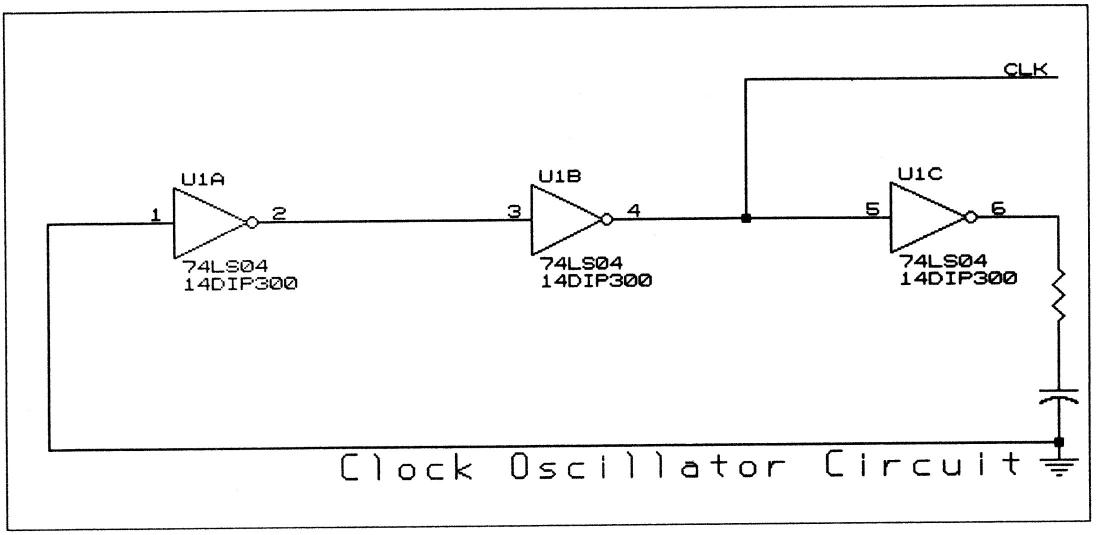


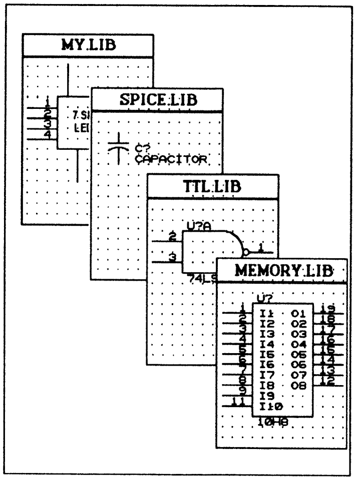
Figure 3-1. Clock oscillator circuit schematic.

About symbols

The first step in building a schematic diagram with **Draft** is to place symbols for the parts on the worksheet. The symbols can represent basic logic functions (such as AND gates), individual parts (such as capacitors), or blocks of circuitry to be designed later. The symbols can represent parts that use different technologies, such as TTL or CMOS.

About libraries

Symbols representing parts are stored in libraries. For **Draft** to get a symbol and place it on a schematic, the library containing it must be configured on the **Configure Schematic Design Tools** screen.



Parts libraries.

As shown in the illustration at right, library filenames typically end with the extension .LIB.

To build the clock oscillator, you need the following symbols:

- ❖ Three inverters
- ❖ One resistor
- ❖ One capacitor

The examples in this tutorial use TTL technology for the inverters.

Where to start

If you are continuing from chapter 2, the **Schematic Design Tools** screen is displayed. Follow these steps if it is not displayed:

1. If the operating system prompt is displayed, enter **ORCAD.**

△ *NOTE: In chapter 1, you set the start-up design to be TUTOR. Check to be sure that "TUTOR Design" is displayed in the middle of the top line of the screen. If it is not, go into* ***Design Management Tools*** *and change to the TUTOR design. This process is described in detail in chapter 1.*

2. On the main screen, select **Schematic Design Tools** and then select **Execute.**

Check library files

Follow these steps to check which libraries are configured:

1. On the **Schematic Design Tools** screen, select **Draft.**
2. Select **Configure Schematic Tools.** The **Configure Schematic Design Tools** screen displays.
3. Scroll down until you can see the **Library Options** area.

   As shown in figure 3-2, **Library Options** shows **Available Libraries** on the left, and **Configured Libraries** on the right.

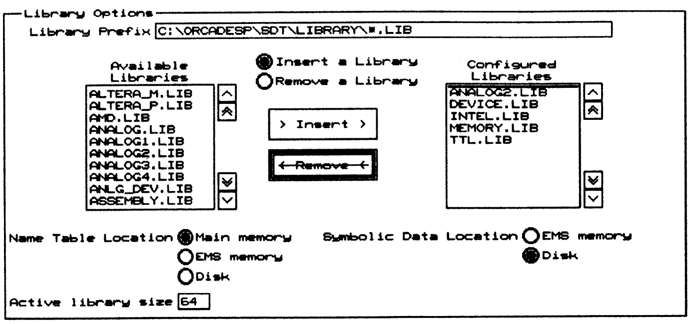


Figure 3-2. ***Library Options*** *area of* ***Configure Schematic Design Tools*** *screen.*

Draft loads and maintains libraries in your computer's memory in the order in which they are listed in the **Configured Libraries** box. This is important when retrieving parts while creating schematics. When you tell **Draft** to get a certain part name, it searches the libraries in the order listed during configuration and gets the *first* part it finds with a matching name.

4. For this chapter, **Draft** needs the library files .\DCLOCK.LIB and PCBDEV.LIB. The library .\DCLOCK.LIB must be listed first in the **Configured Libraries** box.

   Scroll the **Available Libraries** list box up and down by clicking the up and down arrow keys to the right of the list box. The double-arrow keys scroll the list a full box at a time.

5. Locate .\DCLOCK.LIB in the **Available Libraries** list box and select it. Select **Insert** to place it in the **Configured Libraries** list box.

6. Repeat step 5, but this time select PCBDEV.LIB.

*NOTE: If any libraries other than .\DCLOCK.LIB and PCBDEV.LIB are listed in the **Configured Libraries** list box, remove them. To do this, select **Remove a Library**, select the library to remove, and then select **Remove**.*

7. Scroll to the top of the configuration screen (or press the <Home> key) and click **OK** to return to the **Schematic Design Tools** screen.

8. Select **Draft**, and then select **Execute**.

   Draft runs the initial macro you captured in chapter 2. This macro sets the viewing scale to full size and causes the X,Y coordinates, grid references, and visible grid dots to be displayed. When the macro finishes running, a blank worksheet displays.

Placing parts

Follow these steps to get symbols from part libraries:

1. Select the **GET** command from the main menu. The "Get?" prompt displays.
2. Press <Enter> to display the **Which Library?** menu. The menu at right shows the libraries configured for the TUTOR design.

```
Which Library?
PCBDEV.LIB
.\DCLOCK.LIB
```

3. Move the highlight to .\DCLOCK.LIB and press <Enter>. A list of the parts stored in the .\DCLOCK.LIB file displays, as shown at right.

```
Get?
22V10
4SW SPST
74LS04
BATTERY
CAP
GND
LM7805
R
SW PUSHBUTTON
```

4. Select the 74LS04 inverter. An image of the part displays on the worksheet and a command line displays across the top of the screen. When you move the part, the image simplifies temporarily—only the object's outline displays. When you stop moving the part, details redisplay.
5. Move the part to its general location on the worksheet. Refer to the grid reference bars at the left and top edges of the display and use the mouse to move the image to region A-3.
6. To move the part to its precise location, refer to the X,Y grid display at the upper right of the screen and move the image until the display shows that it is at location (12.80, 11.80). You can use the arrow keys to position the part. The part's upper left corner is its reference point for positioning.

7. Press <Enter> and select **Place** from the menu that displays. **Draft** places the part on the worksheet and creates another movable image of the part.
8. Since you need two more copies of the inverter, place copies of the part at locations (14.80, 11.80) and (16.80, 11.80).
9. When you have placed all three parts, press <Esc> to end the operation.

Shortcuts for getting parts

If you know the full name of a part you want to get from a library, you don't have to work your way through the menus. Simply type the complete part name at the "Get?" prompt. For example, if you enter **R** in response to the "Get?" prompt, **Draft** searches through the libraries and displays a resistor.

Place the remaining parts

Follow these steps to add a resistor, a capacitor, and a ground symbol to the clock oscillator circuit:

1. Press <G>, <R>, and <Enter> to select the resistor part from the .\DCLOCK.LIB library. An image of the resistor displays on the worksheet.
2. Move the image to location (17.60, 12.30) and press <P> (or press <Enter> and select **Place**) to place the resistor.
3. Press <Esc>.
4. Press <G> to display the "Get?" prompt. Enter **CAP**. An image of the capacitor displays on the worksheet.
5. Place the capacitor at location (17.60, 13.00).
6. Press <Esc>.
7. Press <G> to display the "Get?" prompt. Enter **GND**. An image of the ground symbol displays on the worksheet.
8. Place the ground symbol two grid spaces below the bottom of the capacitor symbol, and then press <Esc>.

You have now placed all the parts and symbols for the clock oscillator circuit on the worksheet. The next step is to place the wires.

Drawing wires

Compare your worksheet with figure 3-1. Your worksheet should contain the parts shown in figure 3-1, but not the wires. Most of the remaining tasks in this chapter establish signal connections between the parts you placed on the worksheet.

1. Select **PLACE** from the main menu. The **PLACE** menu displays.
2. Select **Wire**. The **PLACE Wire** command line displays.
3. Move the pointer until it rests at the free end of the output pin of the leftmost inverter. This is location (13.50, 12.00).
4. Select **Begin**, then move the pointer right to the input pin of the next inverter.
5. Select **End**. The wire segment is completed.
6. To complete the wiring, draw wires between the remaining parts as shown in figure 3-1.

   You can speed up wire drawing two ways:

   - Select **New** instead of **End** for each wire except the last one.
   - Press <P>, <W>, <B>, <N>, and <E> to select the required menu commands.

△ *NOTE: When drawing wires, be sure to begin and end each wire segment at the end of a part pin, not within the body of the pin. Also be sure that the end of a wire does not overlap a pin. If you accidentally overlap wires on pins or part bodies, error messages result.*

Placing junctions at intersections

Wires that cross one another do not represent a connection. To tell **Draft** that the crossing wires are connected, you must define the intersection as a wire junction. You do this by placing a junction at the intersection.

If two wires (or a wire and a part pin) are connected end to end, however, a junction is not necessary. The connection between the capacitor and the input of the leftmost inverter requires a junction. No junction is necessary for the connection between the resistor and the capacitor because they connect end to end.

A junction is also required where the wire labeled CLK (figure 3-1) connects, between the middle and right-hand inverters.

Place junctions

1. Select **PLACE Junction**.
2. Put the pointer on one of the wire intersections and select **Place**. A junction displays.
3. Place a junction at the other intersection by putting the pointer on it and selecting **Place**.
4. Press <Esc> to dismiss the **Place** command line.

You aren't finished with this circuit yet. You still have to assign values to the resistor and capacitor, add a signal label, and assign reference designators to all the parts. These steps are described in the next sections.

Editing part fields

Each part in **Schematic Design Tools** has ten reserved data areas called *part fields* for holding and displaying additional information. For example, you might use part fields to record part numbers on the schematic—making it easier to track and order parts—or to specify the multiple-element part to which a particular part belongs.

Two of the ten part fields are reserved for particular types of data:

❖ The **Reference** field is reserved for holding reference designator values, such as "U1A" and "Q1."

❖ The **Part Value** field is reserved for holding part names, such as "74LS04," or values relevant to the part, such as ohm (Ω) values for resistors.

The other eight fields are named **1st Part Field** through **8th Part Field.**

To be processed correctly by **Schematic Design Tools,** every part *must* have data in the **Reference** field and in the **Part Value** field.

The data in a part field can be up to 128 characters long. You can edit the contents of these fields and make them visible or invisible on the schematic using the **EDIT** command.

In this chapter, you learn how to edit part fields one at a time. You can also automate part field editing using the **Update Field Contents** tool, as described in *Chapter 7: Using other Schematic Design Tools.*

For the inverters

Follow these steps to specify the multiple-element part for the inverters:

1. Select **EDIT** from the main menu.
2. Put the pointer on the part you want to edit, in this case the leftmost inverter.
3. Select **Edit**. The **Edit part** menu at right displays.

```
Edit part
Reference
Part Value
1st Part Field
2nd Part Field
3rd Part Field
4th Part Field
5th Part Field
6th Part Field
7th Part Field
8th Part Field
SheetPart Name
Orientation
Which Device
```

4. Select **1st Part Field**. The menu at right displays.

```
1st Part Field
Name
Location
Visible
```

5. Select **Name. Draft** displays:

```
1st Part Field?
```

6. Enter **`14DIP300`**. The information displays below the inverter symbol.
7. Select **Which Device** from the **Edit Part** menu.

   The prompt "Which device from package?" and a list of suffix letters (A through F) displays. A through F represent the six 74LS04 inverters in the 14DIP300 multiple-element part.
8. Select **A** from the list and press <Esc>.
9. Repeat steps 2 through 8 for the other inverters you placed. Since they are from the same multiple-element part, enter **14DIP300** for each, and assign suffix letters **B** and **C** to them.
10. Press <Esc> again to remove the **EDIT** command line from the screen.

About reference designator assignments

Notice that the suffix letters of the second and third reference designators you just modified changed to U?B and U?C, respectively. U?A is the first part in the multiple-element part, U?B is the second part, and U?C is the third part. When you run the **Annotate Schematic** tool on this schematic, it changes all of the question marks for this multiple-element part to a common number, such as 4. The parts will then be labeled U4A, U4B, and U4C. The **Annotate Schematic** tool is described in *Chapter 7: Using other Schematic Design Tools.*

You also can edit the reference designator and part values displayed for a part, but doing so prevents **Annotate Schematic** from performing this task. **Annotate Schematic** automatically updates part reference designators and pin numbers associated with the reference designators in multiple-element parts.

For the resistor and capacitor

Follow these steps to edit the part fields for the resistor and capacitor:

1. Select **EDIT** from the main menu.
2. Put the pointer on the resistor.
3. Select **Edit**. The **Edit Part** menu displays.
4. Select **Part Value** and **Name**. **Draft** displays:

```
Value? R
```

5. To change the value, backspace over the present value and enter **`91K`**.
6. Press <Esc> to dismiss the **Edit Part** menu.
7. Put the pointer on the capacitor.
8. Repeat steps 3 through 6, entering **`47uF`** for the part value of the capacitor, measured in microFarads (uF).
9. Press <Esc> again to dismiss the **EDIT** command line.

You are nearly finished with the schematic for the clock oscillator circuit. In the next section, you learn to connect a wire in your circuit using a label. The label allows another circuit on the worksheet to behave as though it is directly connected to the output of this circuit.

Specifying connections with labels

Sometimes you may want to connect wires far apart on the worksheet. To keep the worksheet from looking cluttered, you'd like to do so without having to place a line representing the wire connecting them. You can do this by assigning a label with the same name to both wires.

1. Select **PLACE** from the main menu.
2. Select **Label**. At the "Label?" prompt, enter **CLK**. The label displays.
3. Position the label image so the pointer rests on the unconnected output wire of the clock oscillator circuit. Labels must be placed with the leftmost point of the label name next to the bus or wire.
4. Select **Place**. The "Label?" prompt redisplays.
5. Press <Esc> to dismiss the "Label?" prompt.

Schematic Design Tools treats all wires on this sheet labeled "CLK" as connected, just as if you had drawn the wire from the clock oscillator circuit directly to the other area of the schematic that uses it. You will refer to this wire label in a later chapter of this guide.

Placing comment text

You may often want to leave notes or descriptive text on a schematic diagram. Such text helps you and others understand the functions being performed or documents some aspect of circuit operation. Follow these steps to add a title:

1. Select **PLACE** from the main menu, and then select **Text.**
2. The prompt "Text?" displays. Enter `Clock Oscillator Circuit`.
3. To use the next larger type size for the text, select **Larger**. The image of the text becomes larger.
4. Move the text image so it is centered immediately below the circuit diagram and select **Place**. The "Text?" prompt redisplays.
5. Press <Esc> to dismiss the "Text?" prompt.

△ *NOTE: You may wish to use the* ***ZOOM Center*** *command to center the circuit before placing this text.*

Updating the file

This circuit is now complete. To save your work and exit **Draft**, follow the same steps you took earlier. Select **QUIT**, then **Update file**, to save the worksheet in TUTOR.SCH. Select **Abandon Edits** to exit from **Draft** and return to the **Schematic Design Tools** screen.

Summary

You just completed the schematic diagram for the clock oscillator circuit of the digital clock. In the next chapter, you capture the schematic of the power regulator circuit.

Capturing the power regulator schematic

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

- ❖ Move a group of parts
- ❖ Delete a part
- ❖ Undo a delete
- ❖ Rotate a part
- ❖ Place a power symbol
- ❖ Capture and use a macro
- ❖ Set a tag
- ❖ Jump to a tag or reference location
- ❖ Print the worksheet

Figure 4-1 shows the schematic diagram of the power regulator circuit you create in this chapter. Refer to this figure for placement and orientation information while performing the exercise.

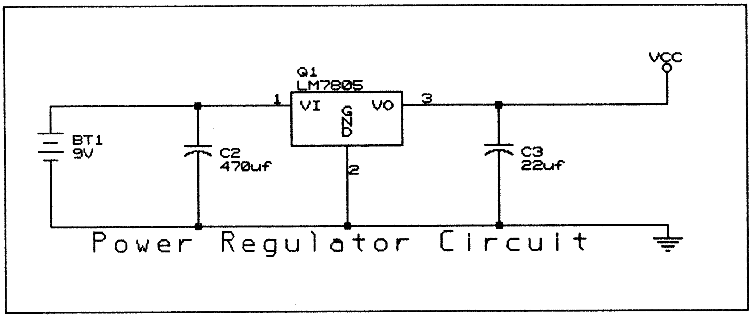


Figure 4-1. Power regulator circuit schematic.

Continuing schematic capture

If you did not abandon edits at the end of chapter 3, you can skip to the next section. Otherwise, follow these steps:

1. From the **Schematic Design Tools** screen, select **Draft.**
2. Select **Execute**. The last active view of the TUTOR.SCH schematic displays.

Moving a group of objects

Although you could just move your viewpoint over to another area of the worksheet to begin working on the power regulator schematic, now is a good time to learn about **BLOCK Move.**

Move the clock oscillator circuit to another place on the worksheet

Before beginning a **BLOCK Move,** zoom out so you can see all of the objects you are moving, as well as the beginning and ending points of the move. Follow these steps to move the clock oscillator circuit:

1. Change the scale from one to five, by selecting **ZOOM Out** twice, or **ZOOM Select 5**. The entire worksheet displays.
2. Select **BLOCK** and then select **Move.**
3. Place the pointer above and to the left of the clock oscillator circuit, and select **Begin**.
4. Move the pointer below and to the right of the circuit. As you move the pointer, a rectangle expands and contracts.
5. When the rectangle encloses the entire circuit, select **End**. The rectangle locks onto the circuit.
6. Move the outline of the circuit until it is centered in the B-2 region of the worksheet.
7. Select **Place** to move the clock oscillator circuit. The circuit moves to the new location.
8. Use **ZOOM** to return to a one-to-one scale. Place the pointer in the A-2 area of the worksheet and select **ZOOM Center. Draft** moves the view of the worksheet so that the pointer displays in the center of the screen. You are now ready to capture the schematic for the power regulator circuit.

Building the power regulator circuit

To build the power regulator circuit, you need the following parts:

- ❖ An LM7805 IC regulator
- ❖ Two capacitors
- ❖ A nine-volt battery
- ❖ Power (V_{CC}) and ground (GND) symbols

As in chapter 3, the digital clock parts library (.\DCLOCK.LIB) contains the parts you need to construct the power regulator circuit.

Get library parts and place them on the worksheet

Follow these steps to get the necessary parts for the power regulator circuit:

1. Select **GET** from the main menu. The "Get?" prompt displays.
2. Press <Enter> then select **.\DCLOCK.LIB**.
3. The parts menu displays. Select an LM7805 (an IC regulator) and place it at location (15.00, 12.50).
4. By now you should be experienced at placing parts. Get the capacitor and place one on each side of the regulator, as shown in figure 4-1.
5. Now get the ground symbol (**GND**) and place it in the location shown in figure 4-1.

Deleting parts from the worksheet

If you place a part and then decide you don't need it after all, **Draft's DELETE** command lets you remove any object placed on the worksheet.

If you delete an object by mistake, you can undo your action.

Delete an object

To familiarize yourself with the delete process, delete the capacitor on the right side of the IC regulator.

1. Select **DELETE** from the main menu. The **DELETE** menu displays, as shown below.
2. Select **Object**. The **DELETE Object** command line displays.
3. Put the pointer on the rightmost capacitor.

```
Delete
Object
Block
Undo
```

4. Select **Delete. Draft** deletes the capacitor from the worksheet.

   Because of the way **Draft** deletes things, some dots may remain on the screen where the deleted object was. They are not really on the worksheet.

5. Press <Esc>. **Draft** redraws the screen. Any extra dots disappear.

Recover a deleted object

Follow these steps to recover the deleted capacitor:

1. Select **DELETE** again.
2. Select **Undo**. The capacitor reappears.

Rotating parts before they are placed

Now you are going to try something a little different. A battery symbol exhibits polarity, so even though you know that the negative terminal goes to ground, the symbol may end up backwards on the schematic if you are not careful. You may have to rotate the part to get the polarity correct. Follow these steps to familiarize yourself with this process:

1. Get the battery part (**BATTERY**) from the .\DCLOCK.LIB library. Once the part is selected, the **Get Part** command line displays:

   ```
   Place Rotate Normal Up Over Down Mirror Find
   ```

2. Select **Rotate** twice and see the effect this has on the battery symbol.

   Before placing the part, experiment with the other **Place** commands to see their effect on the part orientation.

3. If you look closely at the part, you'll notice that the pin 1 end of the part has a long heavy line as part of the border. The pin 2 end has a shorter heavy line as part of the border. The long heavy line indicates the positive terminal of the battery. As shown in figure 4-1, you want the positive terminal up, so rotate the symbol to this orientation (**Down**), and place it on the worksheet.

You have now placed all the parts and symbols, except for the V_{CC} power symbol associated with the power regulator circuit on the worksheet. Next you draw the wires for the power regulator circuit.

Drawing multisegment wires

Compare your worksheet with figure 4-1. Notice that several wires are missing from your worksheet.

In this section, you learn how to draw multisegment wires in one operation. A multisegment wire is a single wire that changes direction several times.

1. Select **PLACE Wire**. The **PLACE Wire** command line displays.
2. Move the pointer to the negative terminal of the battery, and select **Begin**.
3. Move the pointer down approximately three grid spaces.
4. Select **Begin** again and move the pointer to the right until it is directly under the first capacitor.
5. Select **Begin** again and move the pointer to the end of the capacitor pin.
6. Select **End** or **New**. When you draw multisegment wires, remember to start and turn corners with **Begin** and cut the wire with **End** or **New**.
7. Now, connect wire segments between the remaining parts as shown in figure 4-1. Be sure to **Begin** and **End** each wire segment at the end of a part pin, not within the body of the part.
8. Using the **PLACE Junction** command, place junctions in the circuit at the five locations shown in figure 4-1.

△ *NOTE: If you cut a wire with **New**, the **PLACE Wire** command line remains displayed. You don't need to select **PLACE Wire Begin** to start a new wire. You only need to select **Begin**.*

More macros

You could continue drawing wires using keyboard or menu commands, but it's a repetitious process. Every time you begin drawing a wire, you must enter three commands in sequence, **PLACE**, **Wire**, and **Begin**.

You can do this by pressing the first letters of each command, <P><W><B>. You can also use **Draft**'s macro feature to make it even easier by combining these three keystrokes into one keystroke. You were introduced to macros when you developed the initial macro that sets up the work conditions each time **Draft** runs.

The following is a simple example of how to capture a macro. You can extend the principle to create complex macros, automating long command sequences.

Capture a macro to begin a wire

1. Select **MACRO**. The **MACRO** menu displays.
2. Select **Capture**. The "Capture macro?" prompt displays.
3. Press <F1> to assign a keystroke to this macro. "F1" displays at the "Capture macro?" prompt.
4. Press <Enter>. The message "<macro>" displays to remind you that you are capturing a macro and that any commands you select are added to the list of commands being stored in the macro.
5. Type the commands required to begin a wire by pressing <P> <W> <B>.
6. Press the key combination <Ctrl><End> to end the macro definition. The message "<<<MACRO END>>>" displays.

The macro you captured is now stored in the computer's memory and can be run by simply pressing the key you specified—in this case, <F1>. If you turn your computer off, however, the macro will be lost. You must save the macro to a file.

Save the macro

1. Select **MACRO Write.** The "Write all macros to?" prompt displays.
2. Enter the following filename for this macro:

```
Write all macros to? tutor.mac
```

3. **Draft** displays "WARNING: File: tutor.mac exists. Write over it?" and a short menu. Since all the macros in TUTOR.MAC were read into the macro buffer when you ran **Draft,** select **Yes.**

You just saved this macro—and the macros that were already in TUTOR.MAC—to the macro file that automatically loads each time you start **Draft**. You can add more macros to this file as you capture them.

Placing the power symbol

Follow these steps to place the power symbol in the power regulator circuit:

1. Select **PLACE Power**. An image of the power symbol displays, with the value V_{CC} above it. The **PLACE Power** command line displays:

   ```
   Place Orientation Value Type Find Jump Zoom
   ```

   In this example, the power symbol is connected at the top of the wire. However, there may also be cases in which you need to turn the power symbol around.

2. Select **Orientation** to change the power symbol's orientation. The **Orientation of Power Value** menu shown below displays. The image of the power symbol disappears until you make a selection from this menu.

3. Practice changing the orientation of the power symbol. When you finish, select **Top** orientation.

   ```
   Orientation of Power Value
   Top
   Bottom
   Left
   Right
   ```

   See the *Schematic Design Tools Reference Guide* for detailed information about the display options available for the power symbol.

4. Now move the image of the power symbol until it rests on the end of the wire, as shown in figure 4-1, and select **Place**.

5. Press <Esc> to dismiss the **PLACE Power** command line.

Dragging wires

You may often want to move parts without having to replace the wires connected to the parts. Use **BLOCK Drag** to do this. Follow these steps to familiarize yourself with this process:

1. Select **BLOCK Drag. Draft** displays:

```
Begin  Find  Jump  Zoom
```

2. Move the pointer above and to the left of the power regulator circuit and select **Begin.**
3. Move the pointer so the rectangle encloses all of the power regulator circuit, except the ground symbol and the bottom wire of the circuit.
4. Select **End**. The circuit changes color.
5. Move the selected circuitry up approximately two grid spaces.
6. Select **Place.** Notice that the lower wires grow and remain connected to the ground wire.

Editing part fields

For the parts in the power regulator circuit, you need only specify the correct part values for the capacitors and battery. After all of the schematic is captured, you use the **Annotate Schematic** tool to update the other fields.

1. Place the pointer on the left capacitor and select **EDIT Edit Part Value Name**. The "Value?" prompt displays. Change the part value to 470uF.
2. Place the pointer on the right capacitor and change the part value to 22uF.
3. Place the pointer on the battery and change the part value to 9V.

Placing comment text

A title isn't necessary for a circuit, but it is helpful when someone new needs to understand what a portion of circuitry does.

1. Select **PLACE Text** command and enter `Power Regulator Circuit`.
2. Select **Larger** to use the type size that is one step larger than the part labels.
3. Center the text immediately below the schematic diagram and select **Place**.

Changing viewpoints

You have now captured two separate schematics on the same worksheet. At times, you may want to quickly change your viewpoint from one area of the worksheet to another. You use the **JUMP** command to do this.

Jump to new coordinates

Follow these steps to move around the worksheet:

1. Select **JUMP**. The **JUMP** menu displays, as shown below. You can move around the worksheet three ways:

```
Jump
A tag
B tag
C tag
D tag
E tag
F tag
G tag
H tag
Reference
X location
Y location
```

   - ❖ Using **X location** and **Y location**, specify the number of grid steps to add or subtract from the current pointer coordinates.
   - ❖ Using **Reference**, specify a new pointer location using grid reference regions, such as "A3."
   - ❖ Using the tags, move to a pointer location you defined earlier using the **TAG** command.

2. Select **X location**. The prompt "Jump X" displays. Note the current pointer coordinates.

3. At the prompt, enter `+5`. The pointer moves five grid spaces to the right (in the positive direction) and the X reference coordinate reflects a change of 0.50 inches (since each grid space is 0.10 inches).

   - ❖ To move left, enter a negative X value.
   - ❖ To move to an exact X reference, enter a value without a positive or negative sign. For example, to move to X reference .5, enter `5`; or to move to X reference 5.0, enter `50`.
   - ❖ To move up and down, use the **Y location** command.

4. Experiment for a moment with these commands and positive, negative, and unsigned **JUMP** values.

Tag and jump to specific locations

The **TAG** and **JUMP Tag** commands are useful when you need to return again and again to a particular location on the worksheet. The **TAG** command assigns a tag to a location on the worksheet. The **JUMP Tag** command moves the pointer to a tagged location. Follow these steps to practice assigning tags and jumping to them:

1. Place the pointer on the power regulator circuit.
2. Select **TAG** from the main menu. The **Tag set** menu displays, listing eight tag names you can use.
3. Select **A tag**.
4. Move to the clock oscillator circuit. Put the pointer in the middle of the center inverter, and repeat step 2.
5. Select **B tag**.
6. Select **JUMP**, and then select **A tag**. The pointer jumps to the middle of the power regulator circuit, where you assigned the A tag.
7. Now jump to the B tag.

Making a draft-quality print

The last thing to do before ending this chapter is to print out a copy of the worksheet. While **Schematic Design Tools** includes the **Print Schematic** and **Plot Schematic** tools for making copies of entire designs, **Draft** also has a quick way to get a draft-quality print: the **HARDCOPY** command.

To do this, your computer must be connected to a printer. **HARDCOPY** does not work for plotters. The correct printer driver program must be installed along with your other **Schematic Design Tools** software.

Update the file

Follow these steps to save your work before you print the schematic:

1. Select **QUIT** and **Update file. Draft** updates the file TUTOR.SCH to reflect the current state of the worksheet.
2. Press <Esc> to dismiss the **QUIT** menu.

Make a hardcopy of the worksheet

1. Make sure the printer is connected to your computer, powered on, and on line.
2. Select **HARDCOPY** from the main menu. The **HARDCOPY** menu displays.
3. Select **Width of Paper**. Choose the correct paper width for your printer. Select **Narrow** for paper 8.5 inches wide; select **Wide** for paper 13 inches wide.

   After you specify width, **Draft** redisplays the **HARDCOPY** menu.
4. Select **Make Hardcopy. Draft** sends the worksheet to the printer.

△ *NOTE: The size of the printed image depends on the printer driver **Draft** uses. With **HARDCOPY** (and the **Print Schematic** tool), **Draft** always produces an image at a resolution of 100 DPI (dots per inch). If the printer driver used prints at some other resolution, the image printed is changed by a fixed scale factor (100 DPI divided by the printer driver resolution).*

- *If the printer driver resolution is greater than 100 DPI, the printed image is smaller.*
- *If the driver resolution is less than 100 DPI, the printed image is larger.*

For example, if the printer driver you are using prints at a resolution of 300 DPI, the image is printed at one-third (100 ÷ 300) its original size. If the driver prints at 75 DPI, the image is printed at 1⅓ (100 ÷ 75) its original size.

For more information on sending designs to printers and plotters, see the **HARDCOPY** command and the **Print Schematic** tool and the **Plot Schematic** tools in the *Schematic Design Tools Reference Guide.*

Ending a Draft work session

After you save your design and make a hardcopy, you are finished with chapter 4. You need to exit **Draft** to perform steps in the next chapter.

Since you already saved your work, just select **QUIT** and then **Abandon Edits. Draft** exits and the **Schematic Design Tools** screen displays.

Summary

You have completed the schematic diagram for the power regulator circuit of the digital clock. In the next chapter, you use **Edit Library** to create a custom part to use in the display area of the digital clock schematic.

Creating a custom part

Although **Schematic Design Tools** provides extensive libraries containing over 20,000 parts, you may occasionally need a part or symbol not in any library. You use the **Edit Library** tool to modify an existing part or create an entirely new part.

In this chapter, you learn how to:

- Run **Edit Library**
- Reconfigure **Edit Library**'s work conditions
- Draw a part body
- Draw special shapes
- Use shading and fills
- Add pins to the part body
- Add pin names
- Save the new part in a library

Running Edit Library

Edit Library performs a variety of tasks for creating and modifying custom parts and libraries. In this chapter, you create a completely new part to add to an existing library file. For detailed discussions of **Edit Library** commands, see the *Schematic Design Tools Reference Guide.*

Configure Edit Library

Before running **Edit Library,** you must configure it to open the library file called .\DCLOCK.LIB.

1. Select **Edit Library** from the **Schematic Design Tools** screen.
2. Select **Local Configuration** from the menu that displays and then select **Configure LIBEDIT. Edit Library's** configuration screen displays.
3. Look for the .\DCLOCK.LIB file in the **Files** list box. Select .\DCLOCK.LIB. The name .\DCLOCK.LIB displays in the **Source** entry box:

   ```
   Source  .\DCLOCK.LIB
   ```

4. Select **OK** to save the configuration.

Run Edit Library

From the **Schematic Design Tools** screen, select **Edit Library** and **Execute.**

The **Edit Library** screen displays. Initially it is blank, except for pointer coordinates displayed at the upper right of the screen.

Setting up the work conditions

As with **Draft**, you set up certain work conditions in **Edit Library**. You adjust two features: one governs visibility of the outline of the part body. The other governs visibility of the grid dots in the work area.

Make part body border and grid dots visible

1. Press <Enter> to display **Edit Library**'s main menu.
2. Select **SET** from the main menu. The menu shown below displays.
3. Select **Show Body Outline.** "Show Bitmap Body Outline?" displays.
4. Select **Yes.**
5. Select **SET Visible Grid Dots Yes.** Grid dots display in the work area.

Set

AutoPan	YES
Backup file	YES
Error Bell	YES
Left Button	NO
Macro Prompts	YES
Power Pins Visible	NO
Show Body Outline	NO
Visible Grid Dots	NO

Beginning a new part

To modify or create a part, you use the **GET PART**command. When you create a new part, choosing **GET PART** initiates a sequence of queries about the type of part you want to create. You will create a seven-segment LED named TIL309.

Open a part editing pad

Follow these steps to begin a new part:

1. Press <Enter> to display the main menu.
2. Select **GET PART**. The prompt "Get?" displays.
3. Enter `TIL309`, the name of the part you plan to create. The prompt "TIL309 – New Part?" displays.
4. Select **Yes**. The prompt "Sheet Path?" displays. This is relevant when you create a hierarchical design and want the part to refer to another schematic worksheet.
5. Press <Enter> because the TIL309 part does not refer to a schematic. The **Kind of Part?** menu displays. You use **Block** for simple rectangular parts, **Graphic** for more complex shapes, and **IEEE** for IEEE/ANSI drawing standard parts.
6. Select **Graphic** because the LED display is complex. **Edit Library** displays a menu with the numbers 0 through 16 and prompts you for the number of elements in the part.
7. Select **1** because the seven-segment LED display is a single-element part. **Edit Library** asks whether the part has a convert (that is, whether you will also create a DeMorgan equivalent of the part you are creating).
8. Select **No**. The part editing pad displays, bordered by a solid line. Within the pad, a dotted border displays with the name you assigned the new part, **TIL309**. The pointer displays at the bottom right corner of the dotted border. The command line displays **Place** and **escape**.

   The dotted border defines the size and shape of the region within which you create the part body. Pins you attach to the part are created outside this region, with their connection points on the part body border.

You can adjust the size and shape of the dotted border by moving the pointer. Try it.

9. Move the pointer to location (+12.0, +12.0). This changes the part body border to a square shape. Figure 5-1 shows the part editing pad when the pointer is at location (+12.0, +12.0).

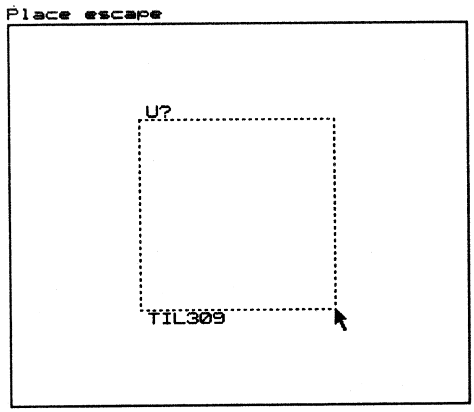


Figure 5-1. Part editing pad.

△ *NOTE: Depending on your computer's monitor, the border may not look square due to the proportions of the screen display.*

10. Select **Place** to set the size of the editing pad. The **BODY <Graphic>** menu displays.

Drawing the body outline

1. Select **Line.** The **BODY Line** command line displays.
2. Move the pointer to the upper left corner of the body, at (+.0, +.0), and select **Begin.**
3. Move the pointer to the next corner, at (+12.0, +0), and select **Begin** again.
4. Move the pointer to the next corner, at (+12.0, +12.0), and select **Begin** again.
5. Move the pointer to the next corner, at (+.0, +12.0), and select **Begin** one last time.
6. Move the pointer to the first corner, at (+.0, +.0), and select **End**. The **BODY <Graphic>** menu displays.
7. Press <Esc> to dismiss the **BODY <Graphic>** menu.

Changing the reference designator

Edit Library automatically puts a placeholder reference designator at the upper left of the part body border. The default class letter is the letter U and the default number is a question mark. The "?" serves as a placeholder for the values to be supplied when you use the part in a schematic and run **Annotate Schematic.** Because U conventionally designates IC parts, you need to change the class letter to D.

1. Select **REFERENCE** from the main menu. The prompt "Initial Reference Designator? U" displays, indicating that U is the current value.
2. Backspace over the U and enter **D**.

Creating a part body

Now you are ready to create the part itself, in this case, a seven-segment LED display. The first step is to create the part body. It consists of seven rectangular objects arranged in the shape of a numeric display, and a circle for the decimal point, as shown in figure 5-2.

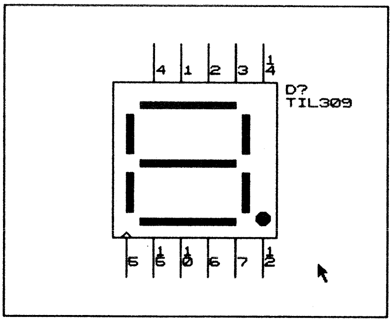


Figure 5-2. The part body you will create.

Zoom in to gain finer pointer control

Like **Draft, Edit Library** can display the part you are working on at several levels of detail. At the lowest level, level 1, the pointer snaps to grid points.

At either of the two higher magnification levels, you can move the pointer to any of 10 intermediate locations between the grid points. You need this fine control to draw the thin LED segments.

1. Select **ZOOM In**. The image doubles in size.
2. Move the pointer between the grid points. Notice that the pointer no longer snaps to grid points.

Draw a rectangle to represent an LED

Follow these steps to draw the first LED segment:

1. Select **BODY** from the main menu. The **BODY <Graphic>** menu displays, as shown below.

```
Body <Graphic>
Line
Circle
Arc
Text
IEEE Symbol
Fill
Delete
Erase Body
Size of Body
Kind of Part
```

2. Select **Line**. The **BODY Line** command line displays.
3. Place the pointer at location (+2.0, +1.5).
4. Select **Begin. Edit Library** is now in line-drawing mode.
5. Move the pointer to location (+9.0, +1.5). A line stretches behind the pointer to show the line segment you are creating.
6. Select **Begin**. The line you drew changes color, showing that it is completed.
7. Move the pointer to location (+9.0, +2.0). This forms the right side of the rectangular shape. A line stretches from the first line to the pointer.
8. Select **Begin** again to complete this segment and begin another.
9. Move the pointer to location (+2.0, +2.0). This forms the bottom segment of the rectangle.
10. Select **Begin**.
11. Move the pointer to location (+2.0, +1.5), the starting point, to complete the rectangle.
12. Select **End** or **New** to end the last segment and complete the rectangle. Press <Esc> to dismiss the **BODY <Graphic>** menu.

Draw six more segments

You can repeat this process for each of the remaining six rectangles, which represent the other LED segments. Each row of table 5-1 shows the coordinates for one rectangle.

To save some time, capture the commands to draw one of the horizontal rectangles as a macro, and run the macro for each of the remaining horizontal segments. Then capture the commands for one of the vertical rectangles, and run that macro for each of the remaining vertical segments. Be sure to use the same corner as the starting point each time you run the macro.

	Top left	*Top right*	*Bottom right*	*Bottom left*
Segment 2	(2.0, 6.0)	(9.0, 6.0)	(9.0, 6.5)	(2.0, 6.5)
Segment 3	(2.0, 10.5)	(9.0, 10.5)	(9.0, 11.0)	(2.0, 11.0)
Segment 4	(1.0, 2.5)	(1.5, 2.5)	(1.5, 5.5)	(1.0, 5.5)
Segment 5	(9.5, 2.5)	(10.0, 2.5)	(10.0, 5.5)	(9.5, 5.5)
Segment 6	(9.5, 7.0)	(10.0, 7.0)	(10.0, 10.0)	(9.5, 10.0)
Segment 7	(1.0, 7.0)	(1.5, 7.0)	(1.5, 10.0)	(1.0, 10.0)

Table 5-1. Coordinates for rectangular LED segments. All coordinates are positive (+) values.

Add the decimal point

In addition to the seven rectangular LED segments, the display unit also has a circular LED at the lower right to represent a decimal point.

1. Select **BODY Circle** to draw the circle.
2. Place the pointer at the location where you want the center of the circle, in this case, location (+11.0, +10.5).
3. Select **Center**. More commands display, one of which is **Edge**. Edge means the edge of the circle being defined. When you move the pointer, a circle expands and contracts.
4. Move the pointer to any location five pointer steps from the center point. For example, put the pointer at location (+11.5, +10.5).
5. Select **Edge**. **Edit Library** places the circle.
6. Press <Esc> to return to the **BODY <Graphic>** menu.

Shading closed shapes

When you create a part, you may want to shade certain objects to make them stand out. To do this you can use the **BODY Fill** command.

1. Select **Fill** from the **BODY <Graphic>** menu. The **Fill** command line displays.
2. Put the pointer within one of the LED shapes.
3. Select **Fill**. **Edit Library** fills in the shape.
4. Repeat steps 2 and 3 for all the LED shapes.
5. Press <Esc> twice to dismiss the **Fill** command line and the **BODY <Graphic>** menu.

After drawing the LEDs, you are ready to add pins so the part can be electrically connected when you place it in a schematic. Because this is a representation of an existing part, you want to add the pins corresponding to the standard version of the part.

Adding pins to a part

Edit Library's **PIN** command is used to add pins. Pins must terminate on the border of the part body. The dotted line around the part is the part's border. If the edge of a part body coincides with this border, pins can terminate directly on the part body. But if the part body is inside this border, you must make a connection between the part body and the border using the **BODY Line** command.

Add a clock pin

1. Select **PIN** from the main menu. The **PIN** command line displays.
2. Move the pointer around. You'll find it is restricted to the part body border.
3. Put the pointer at a location on the border where you want to place the first pin. For this example, put it at coordinates (+1.0, +12.0).
4. Select **Add**. "Pin Name?" displays. The pin name is an identifier that **Draft** uses to identify particular pins. The pin name does not display on the graphic representation of a part.
5. Enter the name **STROBE**. The **Edit Library** tool assigns the name. The prompt "Pin Number?" displays.
6. Enter **5**. The **Pin Type** menu displays. The STROBE pin conducts a clock signal to the internal logic of the part. It should be defined as an input pin type.
7. Select **Input**. The **Pin Shape** menu displays.
8. Select **Clock. Edit Library** places the pin and displays the pin number you entered.

Add a reset pin

1. Place the pointer at the coordinates (+11.00, +12.00).
2. Select **Add**. The prompt "Pin Name?" displays.
3. Enter the name `DPIN`. The **Edit Library** tool assigns the name. The prompt "Pin Number?" displays.
4. Enter `12`. The **Pin Type** menu displays. The DPIN pin conducts a reset signal to the internal logic of the part. It should be defined as an input type pin.
5. Select **Input**. The **Pin Shape** menu displays.
6. Select **Line**. **Edit Library** places the pin and displays the pin number you entered.

Add the remaining pins

1. Put the pointer at a location where you want to place a pin. For this example, put it at coordinates (+3.00, +12.00).
2. Select **Add**. The prompt "Pin Name?" displays.
3. Enter the name `QAIN`. The **Edit Library** tool assigns the name. The prompt "Pin Number?" displays.
4. Enter `15`. The **Pin Type** menu displays. The QAIN pin conducts a signal to an LED segment. It should be defined as a passive type pin.
5. Select **Passive** by putting the highlight bar on this menu item, not by pressing <P>. This is because another menu item begins with "P" (**Power**) and displays in the menu before **Passive**; pressing <P> selects **Power**, not **Passive**.

   When you have specified the pin type, the **Pin Shape** menu displays.
6. Select **Line**.

7. Repeat these steps for the pins connected to the other LED segments. Table 5-2 lists the coordinates, names, numbers, types, and shapes to use for all the pins on this part. You already defined the first three pins, so start with the fourth pin.

Coordinates	*Name*	*No.*	*Type*	*Shape*
(1.0, 12.0)	STROBE	5	Input	Clock
(11.0, 12.0)	DPIN	12	Input	Line
(3.0, 12.0)	QAIN	15	Passive	Line
(5.0, 12.0)	QBIN	10	Passive	Line
(7.0, 12.0)	QCIN	6	Passive	Line
(9.0, 12.0)	QDIN	7	Passive	Line
(3.0, 0.0)	QAOUT	4	Passive	Line
(5.0, 0.0)	QBOUT	1	Passive	Line
(7.0, 0.0)	QCOUT	2	Passive	Line
(9.0, 0.0)	QDOUT	3	Passive	Line
(11.0, 0.0)	DPOUT	14	Passive	Line

Table 5-2. Pins for the TIL309 library part. All coordinates are positive (+) values.

8. Press <Esc>.

When you are finished, you should have 11 pins on the LED. The next step is to add the part to the library.

Saving a new part

Saving a new part involves two operations:

- Copying the part displayed on the screen to the part library currently loaded in the computer's internal memory. This is done using **LIBRARY Update Current.**
- Writing the modified library file in the computer's internal memory to disk. This is done using **QUIT Update file** or **QUIT Write to file.**

Save the part to the current library

1. Select **LIBRARY**.
2. Select **Update Current**. The part currently displayed is written to the library now loaded in memory.

Write the current library to a disk file

1. Select **QUIT Update file. Edit Library** updates the library with the edits you performed during this session and then redisplays the **QUIT** menu.
2. To confirm that the part TIL309 has been stored in a library named .\DCLOCK.LIB, select **Initialize**. The prompt "Read Library?"displays.
3. Enter `.\DCLOCK.LIB`.
4. Select **LIBRARY List Directory Screen.** TIL309 should be in the list of parts in .\DCLOCK.LIB.
5. To dismiss the directory, press any key.

Get the new part

1. Select **GET PART**. When the prompt "Get?" displays, press <Enter>. The **Get** menu displays containing the name of the part you created, TIL309.
2. Select the TIL309 part. It displays in the edit pad.
3. Select **QUIT Abandon Edits** to leave **Edit Library** and return to the **Schematic Design Tools** screen.

Summary

Using the **Edit Library** tool, you created a new part and saved it on disk in an existing library. In *Chapter 2: Introducing Schematic Design Tools,* you configured **Draft** to load the .\DCLOCK.LIB parts library. By adding the TIL309 part to this library, you made the new part available in **Draft** for use while capturing schematics.

Capturing the logic and display circuit schematic

This final schematic diagram for the digital clock circuit contains the logic and display circuit. This circuitry is more complex than the smaller schematics that you captured in the earlier chapters. The tasks you complete in this chapter progress naturally from the processes that were introduced in earlier chapters.

In this chapter you learn how to:

- ❖ Draw a repeatable portion of the schematic
- ❖ Make and place multiple copies of a schematic block
- ❖ Use repeat parameters to place wires and labels

Figure 6-1 shows the portion of the schematic you capture in this chapter.

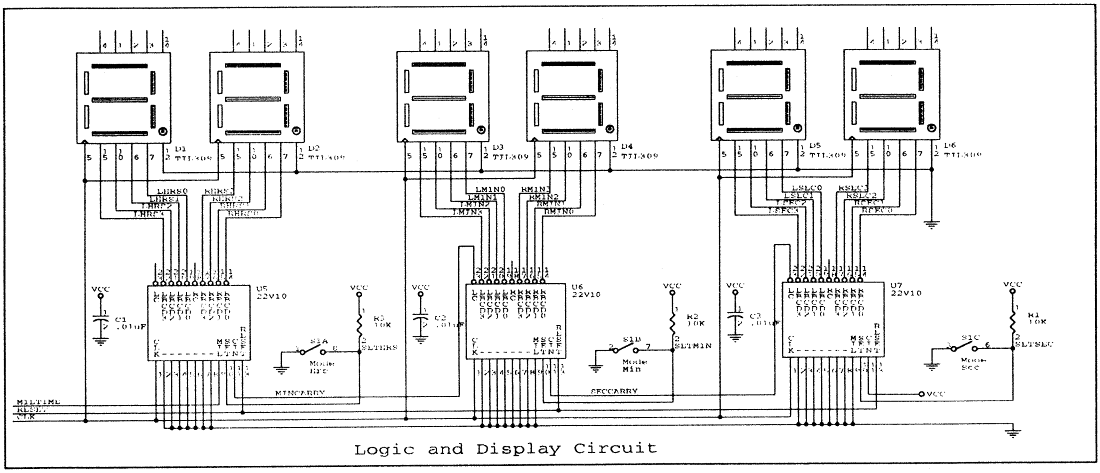


Figure 6-1. The logic and display circuitry.

Choosing parts

To build the rest of the digital clock schematic, you need the following parts:

- 22V10s
- TIL309 LED displays
- Resistors
- Capacitors
- Two switch types (SPST and pushbutton)
- Power (V_{CC}) and ground (GND) symbols

About TIL309 LED display chips

TIL309 display chips were selected in order to keep the chip count for the design down. These displays are capable of accepting binary-coded decimal input. Using TIL309s eliminates the need for decoder circuits. Six TIL309s are required: two each for seconds, minutes, and hours.

About 22V10 PALs

The schematic requires enough pins to drive the six TIL309 display chips and to transfer the "carry" signals. Once again, to reduce the total chip count for the design, 22V10s were chosen to drive the TIL309s rather than individual parts. Since the TIL309s are divided into pairs for seconds, minutes, and hours, you use one 22V10 per pair: three 22V10s altogether.

When deciding to use the 22V10s, the following factors were considered: number of inputs and outputs needed, complexity of the logic that the part needs to handle, cost, and availability. The 22V10s were chosen because they have enough inputs and outputs to accommodate fairly complex logic, are readily available from several manufacturers, and are not extremely expensive.

As in the previous chapters, the clock parts library (.\DCLOCK.LIB) contains the parts you need to construct this circuit. In chapter 5, you added the seven-segment display part (TIL309) to the parts library.

Running Draft again

Select **Draft** from the **Schematic Design Tools** screen. **Draft** displays the last active view of the TUTOR.SCH schematic.

Drawing a portion of the schematic

As you look at the schematic of the logic and display circuitry in figure 6-1, it becomes apparent that three regions are nearly identical—seconds, minutes, and hours. Take advantage of this duplication by creating the schematic for the minutes area (figure 6-2), and copying it to the other areas.

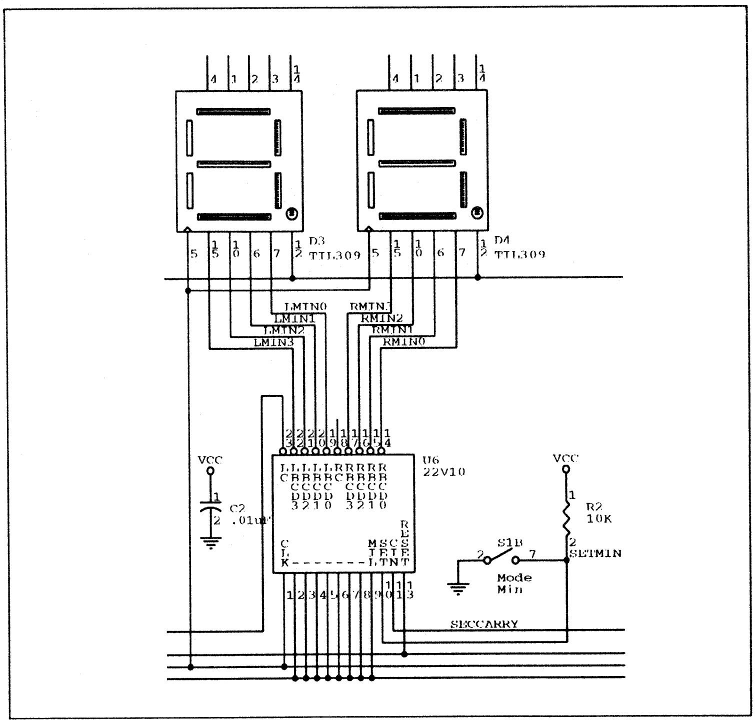


Figure 6-2. The minutes circuit.

Change viewpoint to a clear area

Follow these steps to move to an area of the worksheet with enough room to add the display and logic circuitry:

1. Select **ZOOM Select 2** to change the scale to two-to-one. This scale will work better for the tasks outlined in the next steps.
2. Select **JUMP Reference C 4** to bring the center of the worksheet into view.

The clock oscillator and power regulator circuits you captured earlier are in the lower right area (grid references A-2 and B-2) of the worksheet. The entire upper half of the worksheet is still vacant, so you can use it for this portion of the schematic.

The display and logic circuit shown in figure 6-1 contains so much detail that your immediate task of capturing the minutes area may seem more difficult than it actually is. Figure 6-2 shows only the parts and wires associated with the minutes area of the schematic.

By comparing figure 6-2 with figure 6-1, you see the similarities in each of the areas. The steps in the next section describe how to build the minutes circuit.

Place the parts

Follow these steps to place parts for the minutes circuit on the worksheet:

1. Select **GET**. The prompt "Get?" displays.
2. Press <Enter>. A list of the part libraries specified in the **Schematic Design Tools** configuration displays.
3. Select **.\DCLOCK.LIB**. A list of the parts in .\DCLOCK.LIB displays.
4. Select **22V10**. The part and a command line display. The part's orientation is not correct for this schematic, so you will have to rotate the part.
5. Select **Rotate** to change the part orientation to match the part orientation shown in figure 6-2.
6. Position the part at coordinates (10.50, 6.00) and select **Place.**
7. Get a **TIL309** and place copies at (9.60, 2.50) and (11.50, 2.50).
8. Get a resistor, **R**, and place it at (13.70, 6.30).
9. Get a capacitor, **CAP**, and place it at (9.90, 6.30).
10. Finally, get a switch, **4SW SPST**, and place it at (13.00, 6.80).

At this point you have placed all parts and only need to place wires, nets, and the power and ground symbols.

Draw the first wire

1. Select **ZOOM In**. You need a close-up view of the schematic to perform the next steps.
2. Referring to figure 6-2, move the pointer to the bottom of the resistor symbol and select **Place Wire Begin** to start drawing a wire.
3. Draw the wire so that it is three grid spaces below the lower pins of the 22V10 part at (13.80, 7.60).
4. Select **Begin**.
5. Draw the wire to (11.50, 7.60), and select **Begin**.
6. Continue the wire so that it connects to pin 10 on the 22V10 part (11.50, 7.30).
7. Select **End** to end the wire.

Run the macro to draw the other wires

The <F1> macro you captured earlier to start placing a wire should still be active. Follow these steps to use the macro to draw the rest of the wires:

1. Referring to figure 6-2, draw the wires between the 22V10 part and the right-hand seven-segment display, as shown. Instead of pressing the <P>, <W>, and <B> keys, just press <F1>, and then proceed as usual.
2. Continue using the macro to draw the wires between the 22V10 part and the left-hand seven-segment display, as shown in figure 6-2.

The <F1> macro saves some time, but there are other things you can do to save even more time. One timesaver is the **REPEAT** command.

Define REPEAT parameters

REPEAT duplicates the last entered object, label, or text string and places it on the worksheet. Follow these steps to define the **REPEAT** parameters:

1. Select **SET Repeat Parameters.**
2. Select **X Repeat Step**. The prompt "X Repeat Step?" displays. Enter `1`.
3. Repeat step 1 and select **Y Repeat Step**. The prompt "Y Repeat Step?" displays. Enter `0`.

REPEAT is now set to place a new object exactly one grid space to the right of the pointer when you select **REPEAT**.

Change viewpoint to speed wire placement

The wire placements in the next steps work better if you center the display first:

1. Move the pointer to the end of pin 2 at the bottom of the 22V10.
2. Select **ZOOM Center** to change your viewpoint to center pin 2 on the worksheet.

Use REPEAT to speed wire placement

Follow these steps to quickly draw more wires:

1. Draw a wire seven grid spaces long extending down from pin 2 of the 22V10 PAL. Press <F1> to begin the wire, and press <E> to end it.
2. Select **REPEAT** from the main menu and observe the wire that **Draft** places on pin 3 of the 22V10 PAL. If you usually use the mouse to select commands, try pressing <R> when you select the **REPEAT** command.
3. Select **REPEAT** six more times to place the remaining wires of this length shown in figure 6-2.
4. Place a single horizontal wire along the bottom of these wires as shown in figure 6-2.
5. Select **PLACE Junction**, and then **Place** to put a junction at the leftmost intersection of the wires placed in the prior steps. Press <Esc>.
6. Press <R> seven times to place wire junctions at each of the other wire intersections.

It takes longer to describe how to use the **REPEAT** command than to use it. It's a good idea to plan your schematics to take advantage of **REPEAT**.

Place the remaining parts of the minutes circuit

You have some more wires and junctions and the power and ground symbols to place before you are finished with this portion of the circuit. Because we intend to copy this circuit, it doesn't make sense to edit part labels or comment text yet. Follow these steps to finish placing objects:

1. Select **PLACE Power Place** to put power symbols above the resistor and capacitor symbols, as shown in figure 6-2.
2. Select **GET**, then enter **GND**.
3. Place ground symbols below the capacitor, and below and to the left of the switch symbol.
4. Place the remaining wires shown in figure 6-2.
5. Place junctions at the remaining locations shown in figure 6-2.
6. Examine your worksheet and carefully compare it with the schematic in figure 6-2. The exact position of objects is not as important as the presence or absence of these objects.
7. Correct any problems you find before going to the next exercise.

Copying a block

So far in this chapter, we have been careful to capture only the portions of the schematic that are repeated in several areas. Because three portions of the schematic are nearly identical, you should be able to finish this portion of the schematic in about a third of the time by copying the circuit you just drew.

Save a schematic block

1. Select **ZOOM Out** twice or **ZOOM Select 5** to change to the five-to-one scale. This way you can see all of the objects you are working with.
2. Select **BLOCK Save. Draft** displays this command line:

```
Begin Find Jump Zoom
```

3. Move the pointer above and left of the minutes circuit, and select **Begin.**
4. Move the pointer so the rectangle encloses the minutes circuit.
5. Select **End. Draft** saves the enclosed area in memory and returns to the main menu level.

Copy a circuit

Follow these steps to retrieve and place a copy of the minutes circuit:

1. Select **BLOCK Get.** An outline of the minutes circuit and the **Place** command line displays:

```
Place Find Jump Zoom
```

2. Look at the Y coordinate on the screen. Carefully move the copy to the right of the original, keeping the copy at the same Y coordinate. When the block is positioned correctly, select **Place.**

△ *NOTE: Be sure the copy is horizontally aligned with the original and that there is enough space between the two to allow more wires to be drawn.*

3. After you place the copy of the circuit, the outline displays again so you can continue placing copies.
4. Next, place a copy of the circuit to the left of the original. Again, be sure that the copy is at the same Y coordinate as the original.

It's been a while since you had a look at the schematic you're duplicating. Figure 6-3 is another copy of the logic and display circuit schematic. Your worksheet should look similar to the worksheet shown in figure 6-3.

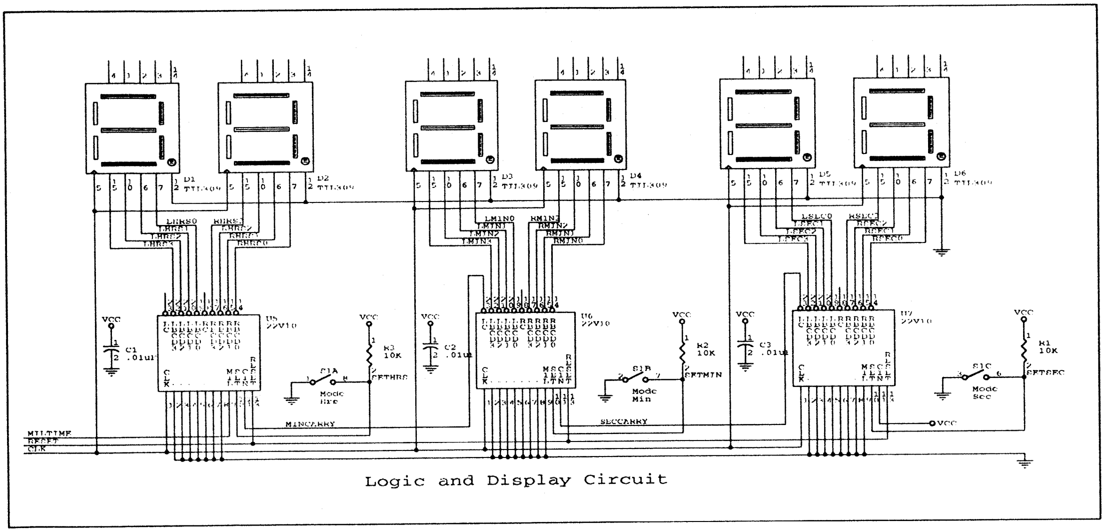


Figure 6-3. The logic and display circuitry.

Finishing the wiring

Figure 6-3 shows how the clock logic will look once you place the wires to connect the seconds, minutes, and hours circuits. The following sections describe how to do this. As you follow the steps in each of these sections, refer to the callouts—①, ②, ③, and so on—in each figure. These callouts correspond to the numbered steps that follow the figure.

Wire the seconds circuit

The following steps correspond to the callouts in figure 6-4. Before you begin, move the pointer to the rightmost 22V10, and select **ZOOM Select 1.**

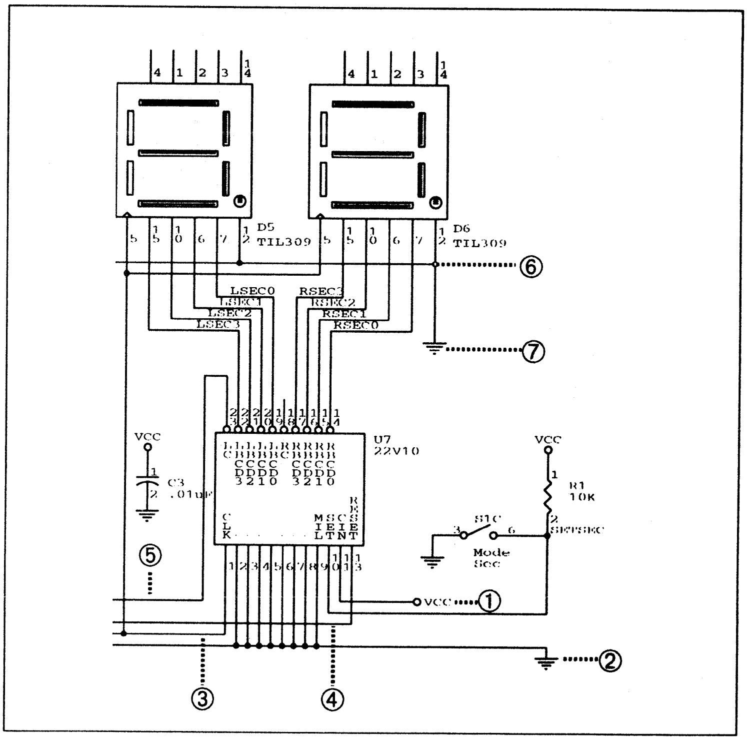


Figure 6-4. Seconds area of the clock logic. The callouts refer to the step numbers in this section.

1. The horizontal wire from the 22V10's pin 11 should be shortened. Referring to the ① in figure 6-4, delete the wire and redraw it so that it is only six or seven grid spaces long.

   Once the wire is the correct length, select **PLACE Power** to get a power symbol. This symbol must be turned before it is placed on the schematic, so select **Orientation Right** and then place it at the end of the wire you just drew.

2. The bottom horizontal wire must have a **GND** symbol added to it.

   Draw a wire that extends two grid spaces down from the end of this wire. Get a **GND** symbol from the library .\DCLOCK.LIB and place it at the end of this wire.

   Draw a wire at the left end of this wire to connect it to the minutes circuit.

3. The second-from-bottom horizontal wire needs to be shortened so that it doesn't run as far to the right. Delete and redraw this wire so that it starts at the end of the wire connecting to the 22V10's pin 1 and goes left to connect to the minutes circuit. The junction at the end of the pin 1 wire is no longer needed. Delete it.

4. The third-from-bottom horizontal wire needs to be shortened so that it stops at the wire that connects to the 22V10's pin 13. Delete this wire and redraw it so that it starts at the end of the wire connecting to the 22V10's pin 13 and goes left to connect to the minutes circuit. Since the junction at the end of the pin 13 wire is no longer needed, delete it.

5. Connect the wire that comes from the 22V10's pin 23 to the minutes logic.

6. Delete the horizontal wire that is immediately below the seconds display and redraw it so that it starts at the wire that comes from the rightmost TIL309's pin 12 and goes left to connect to the minutes circuit.

7. Extend the wire that comes from the rightmost TIL309's pin 12. Get a **GND** symbol from the .\DCLOCK.LIB library and place it at the end of this wire.

The seconds circuit is now complete and connected to the minutes circuit. Next you complete the minutes circuit.

Wire the minutes circuit

Before working on the minutes circuit, move the pointer to the middle 22V10 and select **ZOOM Center**. Then follow the steps below to complete the wiring for the minutes circuit (figure 6-5).

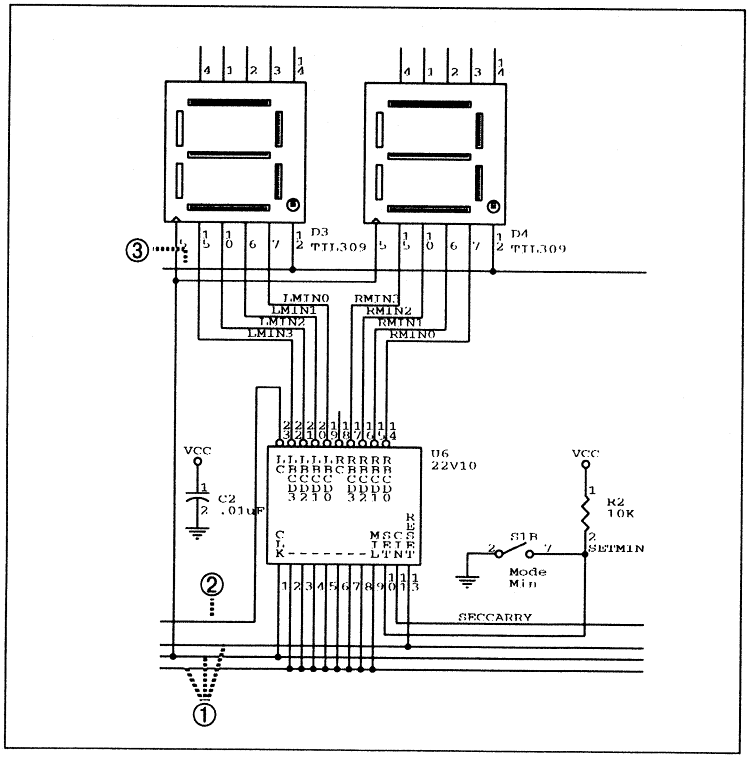


Figure 6-5. Minutes area of the clock logic. The callouts refer to the step numbers in this section.

1. Connect the bottom three horizontal wires to the hours logic.
2. Connect the wire from the 22V10's pin 23 to the hours logic.
3. Connect the horizontal wire that runs just below the minutes display to the hours logic.

The minutes circuit is now complete and connected to the hours circuit. Next you complete the hours circuit.

Wire the hours circuit

Before working on the hours circuit, move the pointer to the leftmost 22V10 and select **ZOOM Center**. Then follow the steps below to complete the wiring for the hours circuit (figure 6-6).

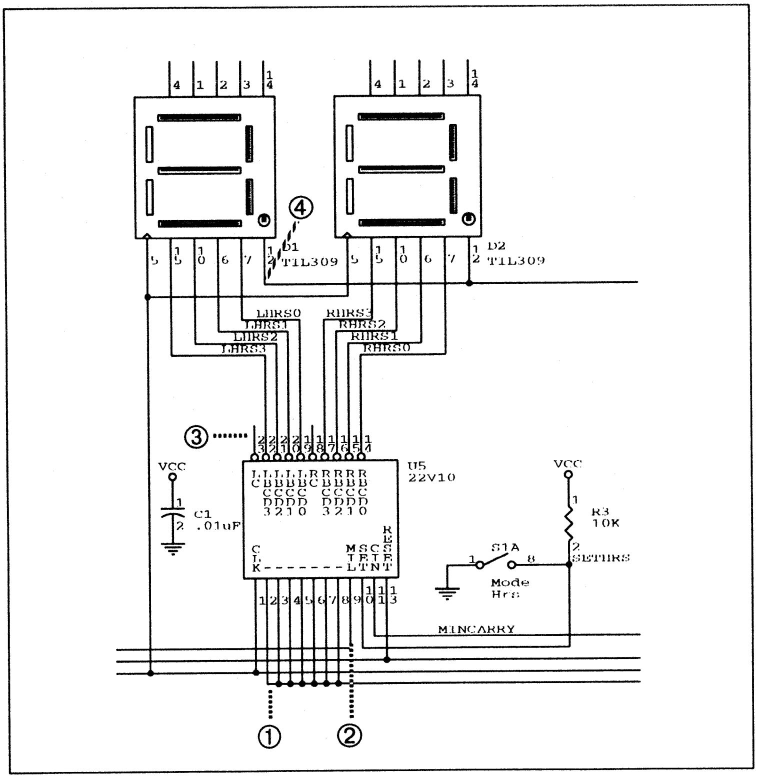


Figure 6-6. Hours area of the clock logic. The callouts refer to the step numbers in this section.

1. The bottom horizontal wire should end at the wire that extends from pin 2 of the 22V10.

   Delete this wire and redraw it so that it ends at the wire from pin 2 of the 22V10. Delete the junction at the end of the pin 2 wire also.

2. The vertical wire from pin 9 of the 22V10 should change so that it doesn't connect to the bottom horizontal wire. Delete this wire and its junction.

   Draw the wire again so it comes down from pin 9, turns left, and goes as far to the left as the other wires.

3. Delete the wire that comes from pin 23 of the 22V10. Be sure to delete all segments of this wire.

4. The horizontal wire that runs just below the hours display should stop at the wire that extends from pin 12 of the leftmost numeric display. Delete this wire and redraw the portion to the right of pin 12. Delete the junction at the end of the pin 12 wire.

View clock logic

You have now connected all of the wires in the logic and display portion of the schematic. Select **ZOOM Select 5** to view the entire schematic. It should look like figure 6-3. Note that the clock oscillator circuit and the power regulator circuit on your schematic do not show in figure 6-3.

Figure 6-7 shows how the schematic will look when you complete the remaining steps in this chapter.

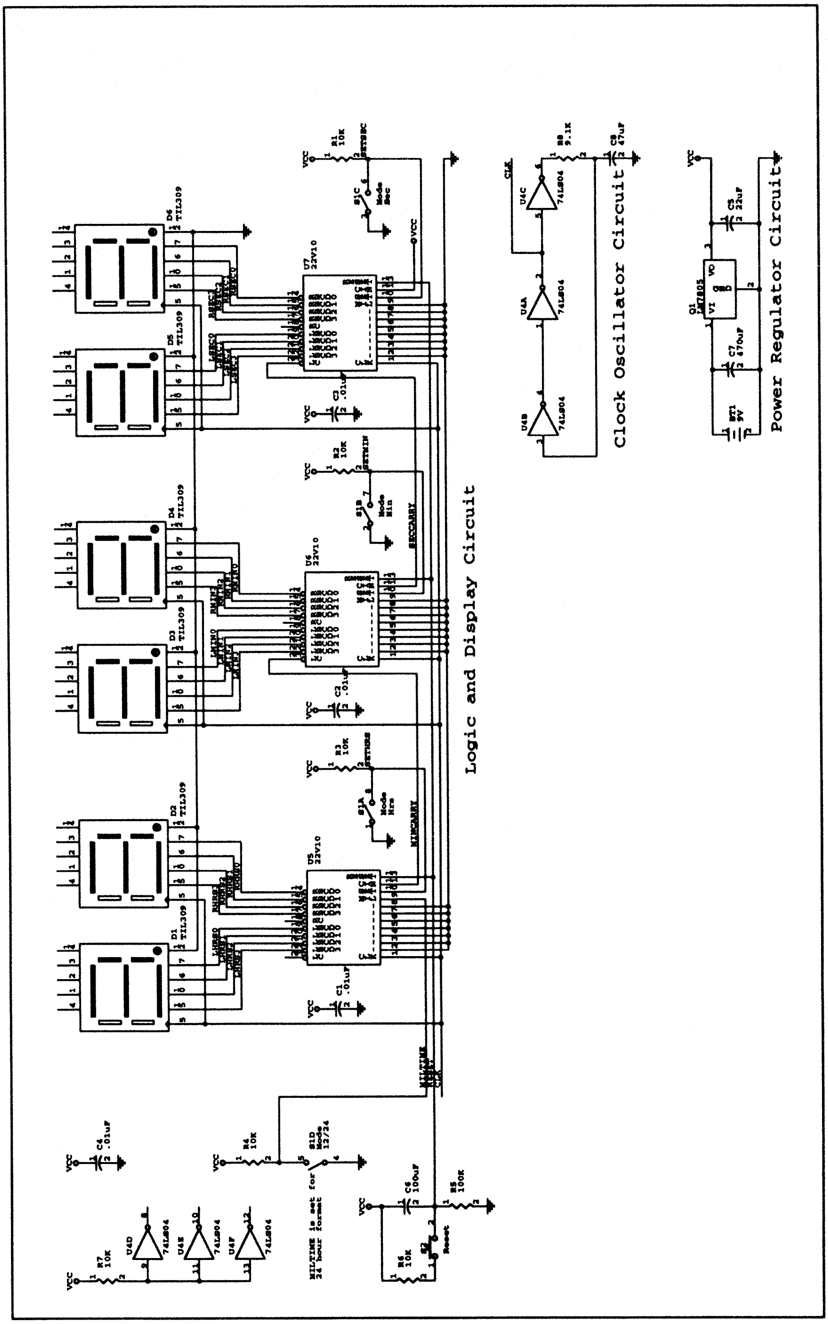


Figure 6-7. Completed TUTOR.SCH schematic.

Finishing the clock schematic

Compare the schematic in figure 6-7 with the schematic you have captured so far. Notice that you need to add a few parts and draw a few more wires to have a functional circuit. You also need to edit the labels and other text in the schematic. The following sections describe how.

Four resistors, three inverters, two capacitors, two switches, and several power and ground symbols are needed to complete the logic and display circuit schematic. Figure 6-8 shows the arrangement of these parts.

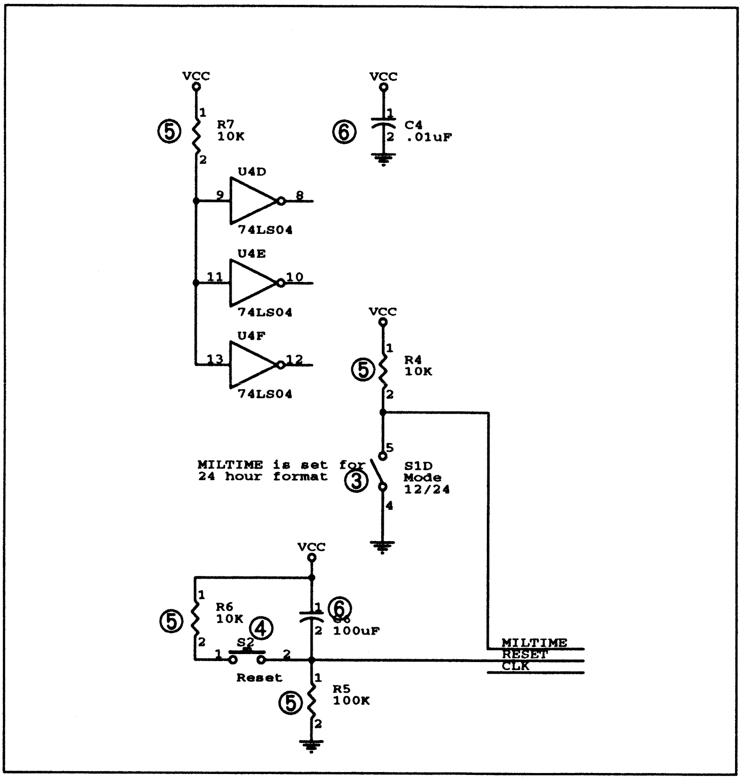


Figure 6-8. Switches, resistors, and other parts to be placed. The callouts refer to the step numbers in this section.

Refer to figure 6-8 as you follow these steps:

1. Select **ZOOM In,** or **Zoom Select 2** to change the scale to two-to-one.

   Change your view of the worksheet so grid reference C-7 is visible.

2. Get the **4SW SPST** switch from .\DCLOCK.LIB.

   The orientation of the 4SW SPST is not correct for this schematic.

3. With the part selected and the outline showing on the screen, select **Rotate** to turn the part so the orientation matches that shown in figure 6-8. Move the part to location (2.70, 6.00) and place it.

4. Get the **SW PUSHBUTTON** part from .\DCLOCK.LIB and place it at location (1.20, 7.60).

5. Get a resistor (**R**) and place four copies at locations (.50, 2.70), (2.70, 4.80), (.50, 7.10), and (2.20, 8.10).

6. Select a capacitor and place it in locations (2.70, 2.60) and (2.20, 7.10).

Place the extra parts

There are also some leftover parts (from multiple-element parts) to be placed on the schematic. Figure 6-9 shows the arrangement of these parts.

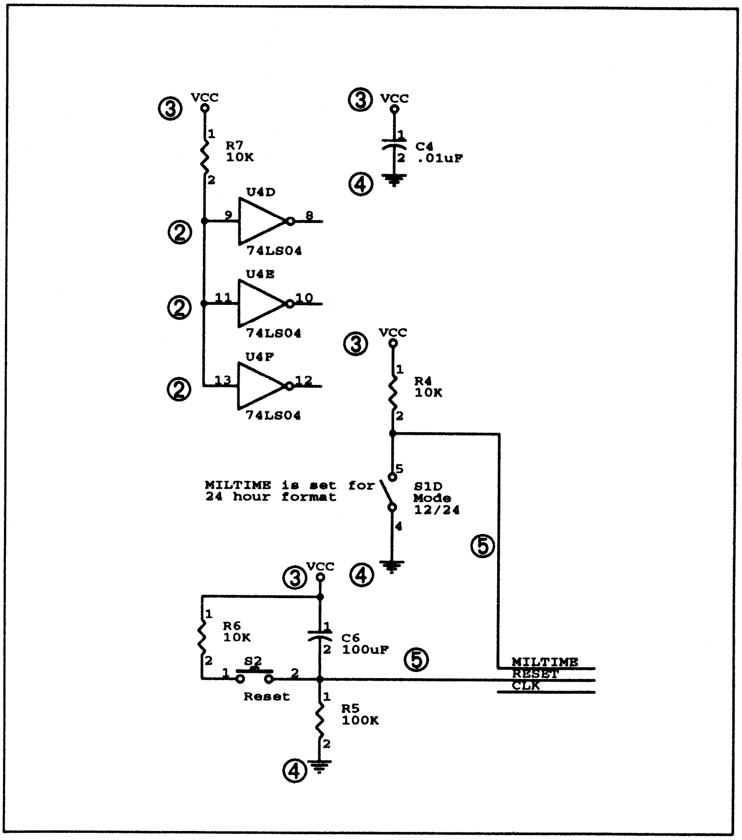


Figure 6-9. Inverters, power symbols, ground symbols, and wires to be placed. The callouts refer to the step numbers in this section.

Refer to figure 6-9 as you follow these steps:

1. Use **ZOOM** to change the scale to one-to-one. Move to reference grid D-8.
2. Get the **74LS04** inverter from .\DCLOCK.LIB and place three copies at locations (1.00, 3.20), (1.00, 3.90), and (1.00, 4.60), as shown in figure 6-9.

3. Place four power symbols at locations (0.60, 2.60), (2.80, 2.50), (2.80, 4.70), and (2.30, 6.80).
4. Get a ground symbol from .\DCLOCK.LIB and place three copies at locations (2.70, 2.80), (2.70, 6.50), and (2.20, 8.50).
5. Place wires to connect the remaining parts, as shown in figure 6-9. Be sure to connect the wires to the logic and display circuit at the two places shown in figure 6-9.
6. Inspect the wire intersections and use the **Place Junction** command to add junctions where required, as shown in figure 6-9.

Edit the part values

1. Put the pointer on the capacitor located at (2.20, 7.20).
2. Select **EDIT Edit**. The **Edit Part** menu displays.
3. Select **Part Value Name**, and change the default value CAP to 100uF.
4. Using the same procedure as in steps 2 and 3, assign the values shown in figure 6-7 to the all parts on the schematic. Table 6-1 gives a list of the values to edit (you have already edited the first item in this table). Notice that some of the parts require that you enter information into the **1st Part Field.**

Part	*Approximate Location*	*Old Part Value*	*New Part Value*	*New 1st Part Field*
Capacitor	(2.20, 7.20)	CAP	100uF	
Capacitor	(2.80, 2.60)	CAP	.01uF	
Resistor	(0.60, 2.80)	R	10K	
Resistor	(0.60, 7.30)	R	10K	
Resistor	(2.30, 8.30)	R	100K	
Resistor	(2.80, 4.80)	R	10K	
Switch	(2.80, 6.10)	4SW SPST	Mode	12/24
Switch	(1.40, 7.80)	SW PUSH-BUTTON	Reset	
Capacitor	(4.90, 6.40)	CAP	.01uF	
Capacitor	(9.90, 6.40)	CAP	.01uF	
Capacitor	(15.30, 6.40)	CAP	.01uF	
Resistor	(8.70, 6.40)	R	10K	
Resistor	(13.80, 6.40)	R	10K	
Resistor	(19.20, 6.40)	R	10K	
Switch	(7.40, 6.90)	4SW SPST	Mode	Hrs
Switch	(13.10, 6.90)	4SW SPST	Mode	Min
Switch	(18.30, 6.90)	4SW SPST	Mode	Sec

Table 6-1. Part value fields to edit.

Add labels to the wires

1. Select **PLACE Label** from the main menu. The "Label?" prompt displays.
2. Enter CLK. This label corresponds to the CLK label you assigned to a wire in the clock oscillator circuit schematic in chapter 3.
3. Move the pointer to the end of the unconnected wire at the left side of the logic and display circuit, and place the CLK label. Remember, when placing a label on a wire, the leftmost point of the label name must be placed next to the wire.

   The clock signal from the clock oscillator circuit is now logically connected to the wire to which you attached the CLK label (figure 6-7).

4. The "Label?" prompt returns each time you place a label. Label the following wires: MILTIME, RESET, SETHRS, MINCARRY, SETMIN, SECCARRY, and SETSEC. Refer to figure 6-7 for the location of these wires. Press <Esc> to dismiss the "Label?" prompt.

You still need to add labels to the wires between the 22V10s and the seven-segment display parts. You could continue placing labels as with the previous steps, but **Draft** allows you to take a shortcut when labeling with repeated text.

Set repeat text parameters

Follow these steps to set the parameters for quickly adding labels to the wires between the 22V10s and the LEDs:

1. Move the pointer to grid reference C-6. You want to look at the area where the labels will be placed.
2. Select **SET Repeat Parameters.** The menu shown below displays.
3. Set **X Repeat Step** to **+1.**
4. Set **Y Repeat Step** to **–1** (equal to the wire spacing).
5. Set **Label Repeat Delta** to **–1.**

```
Set Repeat Parameters
X Repeat Step          +0
Y Repeat Step          +1
Label Repeat Delta     +1
Auto Increment Place NO
```

NOTE: Depending on the spacing between wires, you may have to adjust the X and Y values. Try it and see what works for your worksheet.

These **Repeat Parameters** cause labels to be placed one grid space to the right and one space up, and cause the number in the text to be decremented by one count each time you run the **REPEAT** command.

Place labels with repeat text

Follow these steps to quickly add labels to the wires between the 22V10s and the LEDs:

1. Select **PLACE Label** from the main menu. The prompt "Label?" displays.
2. Enter `LHRS3`.
3. Move the pointer to the bottom wire directly below the leftmost clock segment, and place the label as shown in figure 6-7.
4. Press <Esc> to dismiss the "Label?" prompt.
5. Select **REPEAT** three times.

   The labels LHRS2, LHRS1, and LHRS0 should be placed in the proper relative locations on the worksheet.
6. If the labels are not in the proper location, delete the out-of-position labels, adjust the **Repeat Parameters** to correct the problem, and repeat steps 1 through 5.
7. See figure 6-7 and place labels for the remaining left displays (LMIN*n* and LSEC*n*) by repeating steps 1 through 5.

Place the remaining repeat labels

The labels for wires going to the right displays slant in a different direction than those of the left displays, but otherwise the placement procedure is unchanged.

1. Select **SET Repeat Parameters.**
2. Set the **X Repeat Step** to **–1**, the **Y Repeat Step** to **–1** (again, these values may vary depending on your wire spacing), and the **Label Repeat Delta** to **+1**.
3. Select **PLACE Label**. The prompt "Label?"displays.
4. Enter `RHRS0`.
5. Move the pointer to the bottom wire for the right hours display, and place the label as shown in figure 6-7.
6. Press <Esc> to dismiss the "Label?" prompt.

7. Select **REPEAT** three times.
8. See figure 6-7 and place labels for the remaining right displays (RMIN*n* and RSEC*n*) by repeating steps 3 through 7.

Add comment text

1. Select **PLACE Text**. The "Text?" prompt displays.
2. Enter `Logic and Display Circuit`.
3. Select **Larger** from the **PLACE Text** menu to use a larger type size for the text. The image of the text becomes larger.
4. Center the text below the schematic diagram (at approximately 9.20, 8.40). Press <P> to place the text.
5. The "Text?" prompt redisplays. Enter `MILTIME is set for`.
6. Select **Smaller** from the **PLACE Text** menu until the text size is the same size as the part and wire labels.
7. See figure 6-7 and place the text to the left of the 12/24 switch, at approximately (0.80, 6.10).
8. The "Text?" prompt redisplays. Enter `24-hour format`.
9. See figure 6-7 and place the text under the text you placed in step 7.

Editing the title block

The title block is located in the lower-right corner of the worksheet. You use the title block to provide standard types of information on the schematic, such as a title for the sheet, date, and reference number.

Jump to the title block

You can use the mouse to move the pointer to the title block region of the worksheet, or move there quickly by using the **JUMP** command.

1. Select **JUMP Reference.** The **JUMP to Reference** menu displays.
2. Select **A**, and then **1**. The pointer jumps to region A-1 of the worksheet, and the title block is in view.

   Notice that the title block contains the information entered in chapter 2.

Edit the title block

1. Select **EDIT**. The **EDIT** command line displays.
2. Put the pointer somewhere within the title block. Select **Edit.** The **Edit title block** menu displays.
3. Select one of the types of information listed in the menu. For example, Select **Organization name.** The "Organization name?" prompt displays.

```
Edit title block
Revision code
Title of sheet
Document number
Sheet number
Number of sheets
Organization name
1st Address line
2nd Address line
3rd Address line
4th Address line
```

4. Since you already entered the name of your organization in chapter 2, you can either leave it as it is or delete it and enter a new name.

   Once you press <Enter>, **Draft** stores the information and redisplays the **Edit title block** menu so you can specify other types of information.

△ *NOTE: Once you change a field in the title block, the information entered in the* ***Worksheet Options*** *area of the* ***Configure Schematic Tools*** *screen is no longer used for the changed fields.*

5. Following the procedures in steps 3 and 4, fill in or change other title block information. Filling in the boxes is optional for this tutorial.
6. When you are finished, press <Esc> twice to dismiss the **Edit title block** menu and the **EDIT** command line.

Updating the file

The digital clock design schematic is now complete. Save your work and exit by selecting **QUIT**, then **Update file**, then **Abandon Edits.** The **Schematic Design Tools** screen displays.

Summary

In the past five chapters, you learned several ways to quickly create circuits using **Draft**. In the next chapter, you learn to use some of the other tools.

Using other Schematic Design Tools

In this chapter you learn how to use some of the other **Schematic Design Tools.** These tools are normally used after the schematic is complete. The tools covered in this chapter are:

Annotate Schematic	Automatically updates part reference designators (and the associated pin numbers, in multiple-element parts).
Update Field Contents	Loads information into the **Part Value** field and the **1st Part Field** through **8th Part Field.**
Check Electrical Rules	Checks for conformity to basic electrical rules.
Create Netlist	Creates a netlist and general wire list in one of over thirty standard formats.
Back Annotate	Updates part reference designators by using a list of old and new reference designators.
Create Bill of Materials	Creates a summary list, sorted by reference designator, of all the parts used.
Plot Schematic	Plots a single schematic or all schematics in a design; supports scaling.

Housekeeping

Before proceeding with the tutorial, you should perform a few housekeeping tasks. You have completed quite a bit of work up to this point, so it's a good idea to back up your design files.

In case you were unable to complete the exercises in chapters 1–6, OrCAD has provided copies of the TUTOR schematic and library files. These files are called TUTOR2.SCH and .\DCLOCK2.LIB. By using these files you can perform the remaining exercises with predictable results. Once you back up your design, you copy these two files to TUTOR.SCH and .\DCLOCK.LIB.

Backup Design

Use the **Backup Design** tool to back up onto floppy disks or to another part of your hard disk all the files belonging to a design. To conserve disk space, back-up files are stored in a condensed format. To restore the files to their normal format, use the **Restore Design** tool, which is described in the *OrCAD/ESP Design Environment User's Guide.*

Follow these steps to back up a design:

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button. The **Design Management Tools** menu displays.

```
Design Management Tools
Design Management Tools
Suspend to System
Show Hot Key Assignments
Exit
```

2. Select **Design Management Tools.** The **Design Management Tools** screen displays.
3. Select **Backup Design.** The screen shown in figure 7-1 displays.

△ *NOTE: The **Backup Design** tool does not copy any files that reside outside the specified design directory. See* Chapter 9: Tips and techniques *for a list of the files to include when you transfer a design to another user or to OrCAD Technical Support.*

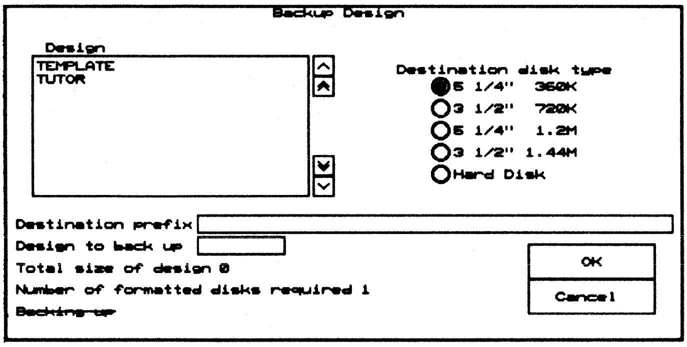
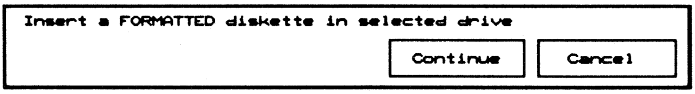


Figure 7-1. ***Backup Design*** *screen.*

4. Select the TUTOR design in the **Design** list box.
5. Move the pointer to the **Destination prefix** entry box and press <Enter>.
6. Enter the path to use for the backup. To back up the design on a floppy disk, enter the destination prefix **A:**. The message shown below displays.



7. Insert a properly formatted disk in drive A and select **Continue.** Select **Cancel** if you want to cancel the backup for the time being.
8. Select **OK.** The design environment makes a back-up copy of the selected design onto the disk or into the directory specified.

   Once the design is backed up, the message "Backup successfully completed" displays along with an **OK** button.
9. Select **OK** and then **Cancel.** The **Design Management Tools** screen redisplays.

Copy File

Follow these steps to copy the TUTOR2.SCH and .\DCLOCK2.LIB files to TUTOR.SCH and .\DCLOCK.LIB, respectively:

1. The **Design Management Tools** screen should still be displayed. Select **File View** at the top of this screen. The screen shown in figure 7-2 displays.

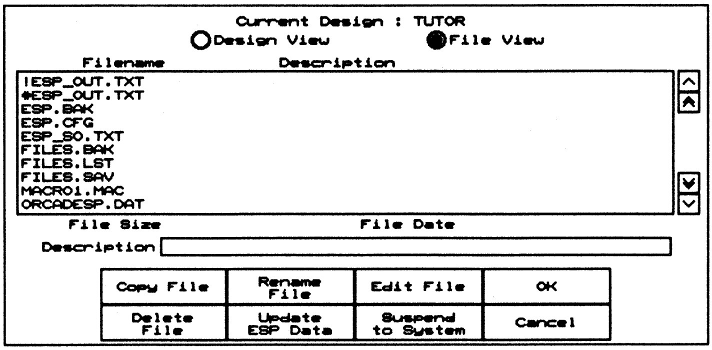


Figure 7-2. File View screen.

2. Select **Copy File**. The screen shown in figure 7-3 displays.

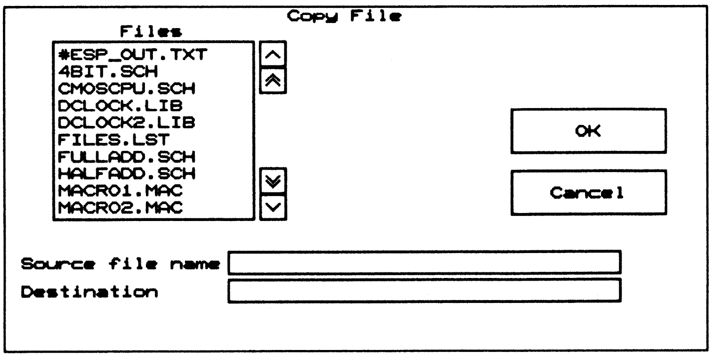


Figure 7-3. Copy File screen.

3. Select the DCLOCK2.LIB file from the **Files** list box.

4. Move the pointer to the **Destination** entry box and press <Enter>.

5. Enter the new name for the file, `.\DCLOCK.LIB`, and then select **OK**.
6. Because DCLOCK.LIB exists, a message box displays. Select **OK** to overwrite DCLOCK.LIB.
7. Select the TUTOR.SCH file from the **Files** list box. You will have to scroll the list to see this filename.
8. Move the pointer to the **Destination** entry box and press <Enter>.
9. Enter the new name for the file, `TUTOR.SCH`, and then select **OK**.
10. Because TUTOR.SCH exists, a message box displays. Select **OK** to overwrite TUTOR.SCH.
11. Select **Cancel** to return to the **File View** screen. Select **Cancel** again to return to the **Schematic Design Tools** screen.

Now that your files are backed up and you have copied TUTOR2.SCH and .\DCLOCK2.LIB to TUTOR.SCH and .\DCLOCK.LIB, you are ready to continue learning about **Schematic Design Tools.**

Running Annotate Schematic

Annotate Schematic scans schematic designs and automatically updates part reference designators. It also updates the pin numbers associated with the reference designators in multiple-element parts. **Annotate Schematic** updates reference designators in the order in which they were placed on the worksheet.

When you first place a part, a default reference designator—such as **U?** for a single-element part and **U?A** for a part of a multiple-element part—displays above the part. **Annotate Schematic** changes the default values to unique values for each part in a specified design.

For example, suppose the specified design contains three occurrences of a particular two-element part with the reference prefix "U." When each of the six parts is placed, **Draft** assigns it the default reference designator, **U?A**. **Annotate Schematic** updates these designators to **U1A, U1B, U2A, U2B, U3A,** and **U3B**. For each of these six parts, the number identifies the two-element part of which the part is an element, and the "A" and "B" suffix letter designates the part's "slot" in that two-element part.

You should run **Annotate Schematic** before running the other tools. Other tools report information about the worksheet file, and if you run **Annotate Schematic** first, you ensure that reported information includes the updated reference designators.

Annotate Schematic modifies the worksheet file, but it also creates a back-up file containing the original worksheet file.

You can also assign values of your choice using **Draft's EDIT** command, but assigning values using **Annotate Schematic** guarantees unique values. Unique reference designator values are necessary for some other processes, such as producing a netlist.

Run Annotate Schematic on TUTOR.SCH

Follow these steps to configure and run **Annotate Schematic:**

1. Select **Annotate Schematic** from the **Schematic Design Tools** screen. The menu at right displays.

```
Annotate Schematic
Execute
Local Configuration
Assign Hot Key
Show Version
Configure Schematic Tools
Help
```

2. Select **Local Configuration** and then select **Configure ANNOTATE.** The **Configure Annotate Schematic** screen displays (figure 7-4).

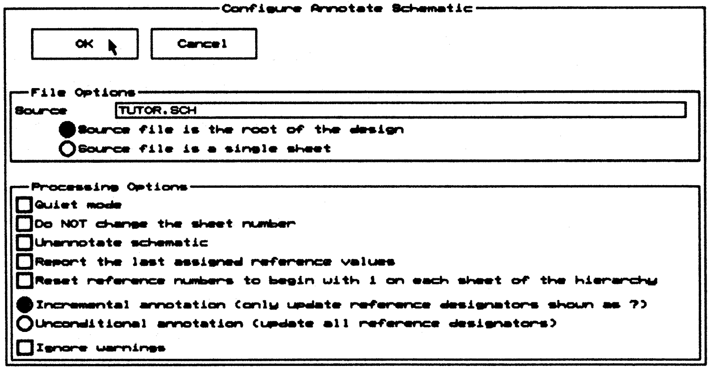


Figure 7-4. The ***Configure Annotate Schematic*** *screen.*

Notice that the **Source** entry box contains the filename TUTOR.SCH. The design environment automatically sets the source to the design name and the default worksheet file extension found in the **Worksheet Options** area of the **Configure Schematic Tools** screen.

3. Select **OK.**

4. Select **Annotate Schematic** and then **Execute.**

As it processes, **Annotate Schematic** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen (figure 7-5).

```
Annotation"    V4.00 01-DEC-90"
(C) Copyright 1985,1986,1987,1988,1989,1990 OrCAD L.P     ALL RIGHTS RESERVED.
Loading "\ORCADESP\SDT\LIBRARY\DCLOCK.LIB"
```

*Figure 7-5. Status messages display at the bottom of the **Schematic Design Tools** screen.*

When **Annotate Schematic** is finished, the monitor box disappears and the full **Schematic Design Tools** screen displays.

△ *NOTE: When you run **Annotate Schematic** on a multiple-sheet design, select **Source file is the root of the design**.*

5. Run **Draft** and examine the TUTOR.SCH worksheet. Note the reference designators and pin numbers. Your reference designators may be slightly different than those shown in this tutorial. This is because **Annotate Schematic** assigns reference designators to parts in the order in which you placed them on the worksheet.

Notice the updated reference designators on the 74LS04 inverters near the upper-left corner of the worksheet. The U?A changed to U1D, U1E, and U1F. All six inverters are from the same multiple-element part. Also notice that the inverters' pin numbers changed.

Running Update Field Contents

You can easily change the contents of the **Part Value** field and the **1st Part Field** through the **8th Part Field** with **Update Field Contents.** (The **Reference** field is protected and cannot be updated with **Update Field Contents.** You can change any of the ten part fields, including **Reference,** using the **EDIT** command in **Draft.**)

Using **Update Field Contents** involves several steps. For each part field you want to update, you must:

- Tell **Update Field Contents** which part field to update. When the value in the corresponding *key field* is the same as one of the *match strings* in the *update file,* **Update Field Contents** loads the part field with the *update string* associated with that match string.
- Define the corresponding key field, on the **Configure Schematic Design Tools** screen. The key field tells **Update Field Contents** what to compare with the match strings in an update file.
- Create the update file that contains the match strings and associated update strings.
- Run **Update Field Contents.**

Update Field Contents compares each match string in the update file with the key field defined for the specified part field. If it finds a match, **Update Field Contents** replaces the contents of the part's specified field with the update string associated with that match string.

In the following sections, you update the **3rd Part Field** of parts on the TUTOR.SCH worksheet. These values in the **3rd Part Field** will be used as module names when you create a netlist later in this chapter.

△ *NOTE: See* Chapter 1: Configure Schematic Tools *and* Chapter 13: Update Field Contents *in the* Schematic Design Tools Reference Guide *for more information about key fields and* ***Update Field Contents*** *and for a description of the update file format.*

Configure Update Field Contents

1. Select **Update Field Contents** on the **Schematic Design Tools** screen, and then select **Local Configuration** and **Configure FLDSTUFF.** The **Configure Update Field Contents** screen displays (figure 7-6).

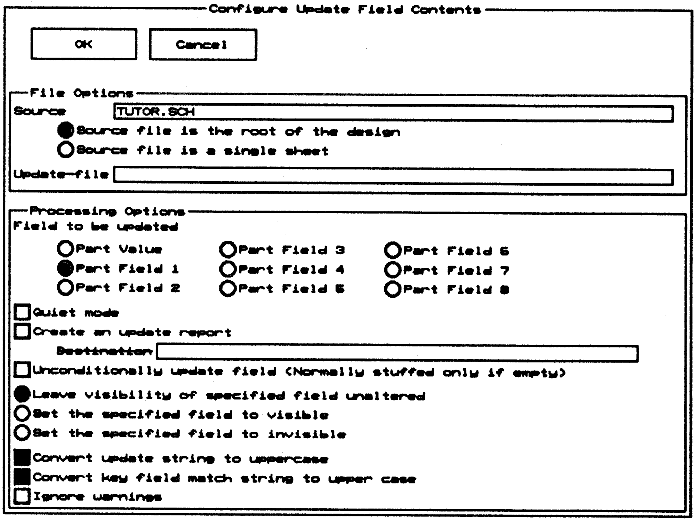


Figure 7-6. The ***Configure Update Field Contents*** *screen.*

2. Make sure the **Source** entry box contains the name of the worksheet, TUTOR.SCH.
3. Enter **`TUTOR.UPD`**, the name of the update file you just created, in the **Update-file** entry box.
4. Under **Field to be updated,** select **Part Field 3.**
5. For this tutorial, select **Unconditionally update field** and **Set the specified field to visible.** (You can make the field invisible later to keep it from cluttering your worksheet.)
6. Select **OK.** The **Schematic Design Tools** screen displays.

Define the key field

1. Select **Update Field Contents** on the **Schematic Design Tools** screen, and then select **Configure Schematic Tools.**

2. Scroll to the **Key Fields** area of the **Configure Schematic Design Tools** screen (figure 7-7).

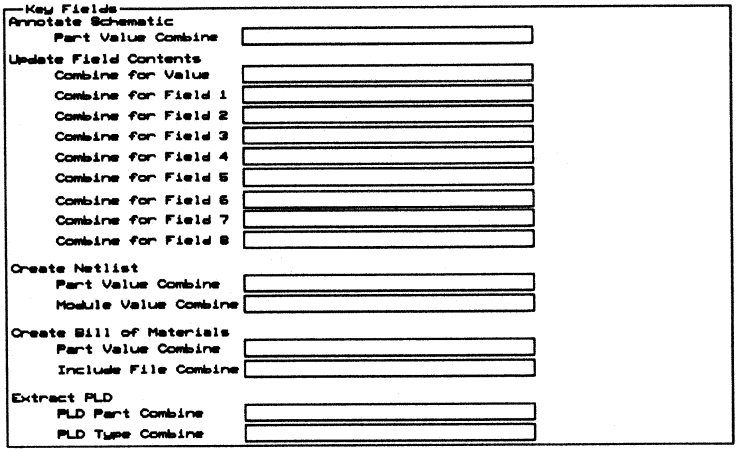


Figure 7-7. Key Fields area of the Configure Schematic Design Tools screen.

3. Enter **V** (it *must* be an uppercase "V") in the **Combine for Field 3** entry box. This tells **Update Field Contents** to compare the contents of the **Part Value** field with the match strings in the update file.

4. Select **OK.** The **Schematic Design Tools** screen displays.

Create the update file

The update file contains pairs of strings: a match string followed by an associated update string. Each string is enclosed in single quotation marks. For readability, place one pair on each line of the file.

1. Create a text file using **Edit File**. For this exercise, call the file TUTOR.UPD.

   See the *ORCAD/ESP Design Environment User's Guide* for more information about the text editor that comes with the design environment, or to learn how to configure the design environment to use another text editor.

2. Include the information shown at right in the text file. Press <Tab> or use spaces to separate paired items.

3. Save the text file.

```
'9V'        '9VBAT'
'.01uF'     'CK06'
'22uF'      'CK15'
'100uF'     'CK15'
'470uF'     'CK15'
'47uF'      'CK06'
'LM7805'    'TO220'
'91K'       'RC06'
'10K'       'RC06'
'100K'      'RC06'
'Mode'      '8DIP300'
'Reset'     'TJACK200'
'22V10'     '24DIP300'
'74LS04'    '14DIP300'
```

△ *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.*

Update Field Contents compares each of these match strings with the **Part Value** field of each part. If it finds a match, **Update Field Contents** loads the corresponding update string into the part's **3rd Part Field.**

Update the fields

1. Select **Update Field Contents** on the **Schematic Design Tools** screen, and then select **Execute.**

   As it processes, **Update Field Contents** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

2. Run **Draft** on TUTOR.SCH to see the updated fields on the worksheet.

Each part with a **Part Value** field that matched one of the match strings in TUTOR.UPD now displays a **3rd Part Field.** Each **3rd Part Field** contains the update string associated with the match string that matched the part's **Part Value** field.

Hide the fields

To minimize clutter on your worksheet, use **Select Field View** to hide the **3rd Part Field** for all the parts.

1. Select **Select Field View** on the **Schematic Design Tools** screen, and then select **Local Configuration** and **Configure FLDATTRB.** The **Configure Select Field View** screen displays (figure 7-8).

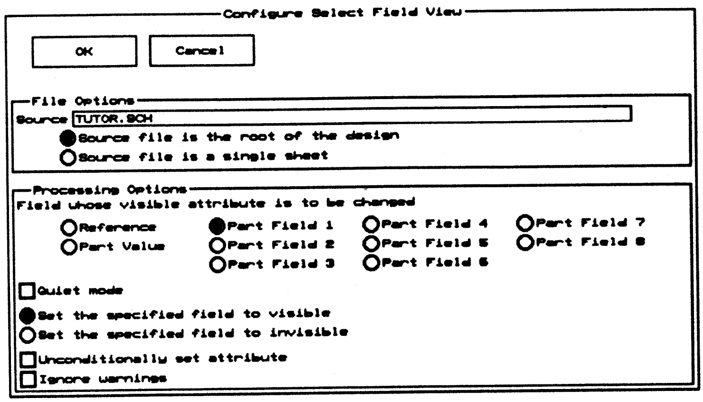


Figure 7-8. The Configure Select Field View screen.

2. Make sure the **Source** entry box contains the name of the worksheet, TUTOR.SCH.

3. In the **Processing Options** area, select **Part Field 3** and **Set the specified field to invisible.**

4. Select **OK**. The **Schematic Design Tools** screen displays.
5. Select **Select Field View** on the **Schematic Design Tools** screen, and then select **Execute.**

   As it processes, **Select Field View** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.
6. Look at TUTOR.SCH in **Draft**, and notice that the **3rd Part Field** on all the parts is not displayed.

△ *NOTE: You can also control the visibility of individual part fields with the* ***EDIT*** *command in* ***Draft****.*

Running Check Electrical Rules

Check Electrical Rules performs a general electrical rules check. It issues warnings for unused inputs on parts, unlabeled wires connected to a bus, and invalid connections.

△ *NOTE: Always use* ***Check Electrical Rules*** *on your designs before going on to* ***Digital Simulation Tools*** *or* ***PC Board Layout Tools****. If any errors are reported, correct them before trying to simulate the design or the simulation results will be inaccurate.*

Follow these steps to configure and run **Check Electrical Rules:**

1. Select **Check Electrical Rules, Local Configuration,** and then **Configure ERC.** The **Configure Check Electrical Rules** screen displays.

   Notice that the **Source** entry box contains the filename TUTOR.SCH.

   Notice that the **Destination** entry box contains the filename TUTOR.ERC. **Check Electrical Rules** stores the report it creates in a text file with the worksheet name and the .ERC extension.

   You can also specify a path to another directory and another file, but for the purposes of this exercise, you should place the report in TUTOR.ERC.
2. Select **OK.**
3. Select **Check Electrical Rules** and then **Execute.**

   As it processes, **Check Electrical Rules** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.
4. After **Check Electrical Rules** is finished running, a message box displays. You may select **View Output** to display the report, or select **OK** to dismiss the message box. Select **OK.**
5. Examine the file TUTOR.ERC with **Edit File.** The contents of the file should be similar to figure 7-9.

```
TUTOR.SCH

Electrical Rules Check Report
Digital clock schematic  Revised:      June 11, 1992
          Revision: B
OrCAD, Inc.
3175 NW Aloclek Drive
Hillsboro, Oregon  97124-7135



WARNING: Single node net. Net at:  X :=   4.90,Y :=   2.20
WARNING: Single node net. Net at:  X :=   5.10,Y :=   2.20
   .
   .
   .

WARNING: Single node net. Net at:  X :=  11.10,Y :=   5.70
WARNING: Single node net. Net at:  X :=  16.10,Y :=   5.70
```

Figure 7-9. The TUTOR.ERC file.

The report lists "single node nets," which are unconnected signals at each of the locations cited. Since the reported pins were intentionally not connected, you can ignore these warnings. If desired, you can examine the schematic and locate these pins.

View errors

Now use **Draft** to view the schematic. Notice a circle at each location where an error is reported by **Check Electrical Rules**. These are *error markers*. To display the associated error message, place the pointer in the center of an error marker and select **INQUIRE** from the main menu. The error message displays at the top of the screen. Repeatedly selecting **INQUIRE** cycles through all of the error messages for a particular error marker.

Select **QUIT Abandon Edits** when you are done looking at the schematic.

△ *NOTE: If you save the schematic file by selecting* **QUIT** ***Update File****, the error markers are erased.*

Running Create Netlist

Create Netlist creates a connectivity database, or netlist, in a number of possible formats. To create a proper netlist, you must carefully deal with labels, module ports, and power objects. The general guidelines are:

- Place labels in the correct format on all buses.
- Place labels in the correct format on all signals connecting to a bus.
- Place module ports in the correct format on all signals going off the worksheet.
- Don't put blank spaces in labels or between prefixes and suffixes in bus and module port names.
- Do not overlap wires or buses with other wires, buses, or object pins.

For a more detailed discussion of these guidelines, see *Chapter 3: Guidelines for creating designs* in the *Schematic Design Tools Reference Guide.*

Specify where to get the module value

1. Select **Create Netlist,** and then select **Configure Schematic Tools.**

2. Scroll to the **Key Fields** area of the **Configure Schematic Design Tools** screen (figure 7-10).

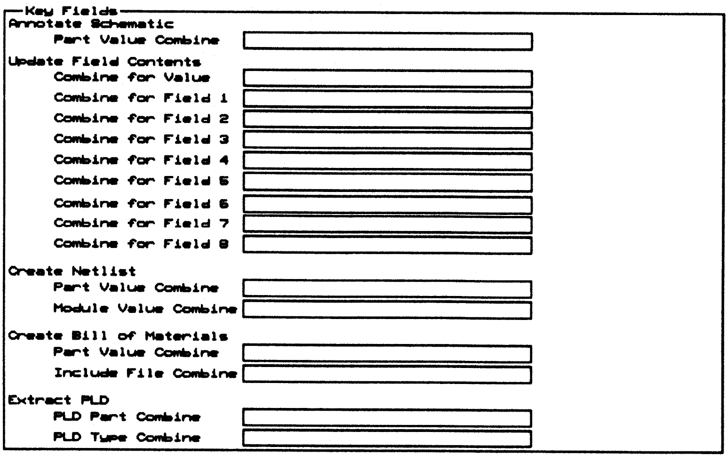


*Figure 7-10. **Key Fields** area of the **Configure Schematic Design Tools** screen.*

3. Enter **3** in the **Module Value Combine** entry box to tell **Create Netlist** to use the contents of the **3rd Part Field** as the module name. (You loaded the module names into the **3rd Part Field** when you ran **Update Field Contents** earlier in this chapter.)

4. Select **OK.** The **Schematic Design Tools** screen displays.

△ *NOTE: For any part that does not have a value in the field you specify, **Create Netlist** will substitute the contents of the **Part Value** field. In the section* Running Update Field Contents, *for example, you did not specify a **3rd Part Field** for the TIL309 parts, because the module name is the same as the **Part Value** for these parts.*

Create a netlist in WIRELIST format

1. Select **Create Netlist** and then **Local Configuration** to configure **Create Netlist**.

   The menu shown at right displays. This menu has three processes to configure: INET, ILINK, and IFORM. INET is the compiler. ILINK is the connectivity linker, and IFORM is the netlist formatter. Each of these processes must be turned on to create a netlist. For more information on each of these processes, see the *Schematic Design Tools Reference Guide.*

   ```
   Select Configuration
   Configure INET
   Configure ILINK
   Configure IFORM
   INET on
   ILINK on
   IFORM on
   ```

2. Select **Configure INET**. The **Configure Incremental Netlist** screen displays (figure 7-11).

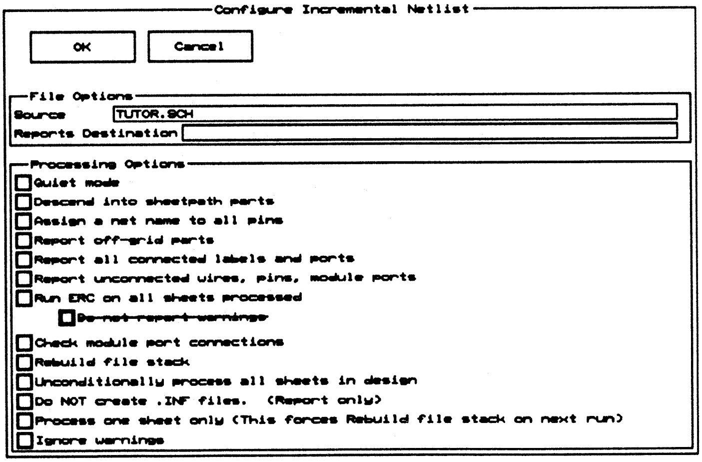


Figure 7-11. The ***Configure Incremental Netlist*** *screen.*

3. In the **File Options** area of the screen, check to be sure that the **Source** entry box contains the filename TUTOR.SCH. The design environment automatically sets the **Source** entry box to the design name and the default worksheet file extension found in the **Worksheet Options** area of the **Configure Schematic Tools** screen.

4. Select **Cancel** to leave the configuration screen without making any changes. The **Schematic Design Tools** screen displays.
5. Now display **ILINK's** local configuration screen. Notice that the **Source** entry box contains the filename TUTOR.INF. Select **Cancel.**
6. Now display **IFORM's** local configuration screen. **IFORM** is the netlist formatter that converts the connectivity database that has been linked by **ILINK** into the format specified in this configuration.

   The **Source** entry box should contain the name TUTOR, showing that you will format the TUTOR files created by **ILINK.**

   The **Destination 1** entry box should contain the filename TUTOR.NET.
7. The **Format prefix/wildcard** should contain the following pathname and filter:

   ```
   Format prefix/wildcard
   C:\ORCADESP\SDT\NETFORMS\*.CCF
   ```

   The **Netlist format** list box contains a number of files. Edit the **Format prefix/wildcard** entry box by inserting a "W" before the *, so that it becomes:

   ```
   Format Prefix/Wildcard
   C:\ORCADESP\SDT\NETFORMS\W*.CCF
   ```

   The list box now contains far fewer filenames. Select WIRELIST.CCF. The selected netlist format filename displays in the **Selected format** entry box:

   ```
   Selected format: WIRELIST.CCF
   ```
8. Select **OK** to save all of the configuration changes.

9. Select **Create Netlist** and then select **Execute** to run **Create Netlist.**

   As it processes, **Create Netlist** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

10. Using **Edit File,** look at the file generated by **Create Netlist,** TUTOR.NET. It should look like the wirelist-format netlist file shown in figure 7-12.

```
Wire List

Digital clock schematic                                   Revised:      June 11, 1992
                                                          Revision: B
OrCAD, Inc.
3175 NW Aloclek Drive
Hillsboro, Oregon  97124-7135


<<< Component List >>>
.01UF                         C5         CK06
.01UF                         C6         CK06
.01UF                         C7         CK06
.01UF                         C8         CK06
100K                          R4         RC06
100UF                         C4         CK15
10K                           R2         RC06
10K                           R3         RC06
10K                           R5         RC06
10K                           R6         RC06
10K                           R7         RC06
10K                           R8         RC06
22UF                          C3         CK15
22V10                         U2         24DIP300
22V10                         U3         24DIP300
22V10                         U4         24DIP300
470UF                         C2         CK15
47UF                          C1         CK06
74LS04                        U1         14DIP300
91K                           R1         RC06
9V                            BT1        9VBAT
LM7805                        Q1         TO220
MODE                          S1         8DIP300
RESET                         S2         TJACK200
TIL309                        D1         TIL309
TIL309                        D2         TIL309
TIL309                        D3         TIL309
TIL309                        D4         TIL309
TIL309                        D5         TIL309
TIL309                        D6         TIL309

<<< Wire List >>>

  NODE  REFERENCE     PIN #   PIN NAME          PIN TYPE       PART VALUE

[00001] N00001
        R2              2       2               Passive        10K
        U1              9       I_D             Input          74LS04
        U1              11      I_E             Input          74LS04
        U1              13      I_F             Input          74LS04

[00002] CLK_1 (local to sheet 1)
        U1              4       O_B             Output         74LS04
        U1              5       I_C             Input          74LS04
        D6              5       STROBE          Input          TIL309
        D5              5       STROBE          Input          TIL309
        U4              1       CLK             Input          22V10
        D4              5       STROBE          Input          TIL309
        D3              5       STROBE          Input          TIL309
```

Figure 7-12. The wirelist-format netlist (continued on next page).

```
         D2             5          STROBE       Input       TIL309
         D1             5          STROBE       Input       TIL309
         U2             1          CLK          Input       22V10
         U3             1          CLK          Input       22V10

         .
         .
         .

[00040]  GND
         C2             2          2            Passive     470UF
         BT1            2          -            Passive     9V
         Q1             2          GND          Input       LM7805
         C3             2          2            Passive     22UF
         C1             2          2            Passive     47UF
         U1             1          I_A          Input       74LS04
         R4             2          2            Passive     100K
         U4             3          -            Input       22V10
         U4             2          -            Input       22V10
         U4             4          -            Input       22V10
         U4             5          -            Input       22V10
         U4             6          -            Input       22V10
         U4             7          -            Input       22V10
         U4             8          -            Input       22V10
         U2             2          -            Input       22V10
         U2             3          -            Input       22V10
         U2             4          -            Input       22V10
         U2             5          -            Input       22V10
         U2             6          -            Input       22V10
         U2             7          -            Input       22V10
         U2             8          -            Input       22V10
         U2             9          MIL          Input       22V10
         U3             2          -            Input       22V10
         U3             3          -            Input       22V10
         U3             4          -            Input       22V10
         U3             5          -            Input       22V10
         U3             6          -            Input       22V10
         U3             7          -            Input       22V10
         U3             8          -            Input       22V10
         U3             9          MIL          Input       22V10
         S1             4          1_D          Passive     MODE
         U3             12         GND          Power       22V10
         S1             3          1_C          Passive     MODE
         U2             12         GND          Power       22V10
         S1             2          1_B          Passive     MODE
         U4             12         GND          Power       22V10
         C8             2          2            Passive     .01UF
         C7             2          2            Passive     .01UF
         C6             2          2            Passive     .01UF
         S1             1          1_A          Passive     MODE
         D6             12         DPIN         Input       TIL309
         D5             12         DPIN         Input       TIL309
         D1             12         DPIN         Input       TIL309
         D2             12         DPIN         Input       TIL309
         D3             12         DPIN         Input       TIL309
         D4             12         DPIN         Input       TIL309
         U1             7          GND          Power       74LS04
         C5             2          2            Passive     .01UF
```

Figure 7-12. The wirelist-format netlist (continued from previous page).

Running Back Annotate

If you want to change the reference designator values assigned by **Annotate Schematic** (or the values you manually assigned), you need not reopen the worksheet and edit the reference designators one by one.

Back Annotate lets you change as many reference designators as you want in a single operation. You create a text file containing the current and new values (called a WAS/IS file) and then run **Back Annotate**, specifying the worksheet name and the WAS/IS filename.

You can run **Back Annotate** on a single worksheet or on an entire design.

For example, consider the TUTOR.SCH worksheet. Currently, the six LED parts in TUTOR.SCH have reference designators of D1, D2, D3, and so on. Suppose you decide you want the values to be A1, A2, A3, and so on. In this example, you will run **Back Annotate** on the schematic, TUTOR.SCH.

Change reference designator values

1. Create a text file using **Edit File**. Enter a filename in the **File to Edit** entry box. For the purposes of this exercise, enter `TUTREF`.

   See the *OrCAD/ESP Design Environment User's Guide* for more information about the text editor that comes with ESP, or to learn how to configure ESP to use another text editor.

2. Include the information shown at right in the text file. Use <Tab> or blank spaces to separate the paired items.

```
D1  A1
D2  A2
D3  A3
D4  A4
D5  A5
D6  A6
```

3. Save the text file.

△ *NOTE: Be sure to save this file as text only. Any special formatting inserted by your text editor causes* ***Back Annotate*** *to fail. In addition, some text editors may attach an extension to the TUTREF file. If it does, be sure to enter the extension when running* ***Back Annotate****.*

4. Return to the **Schematic Design Tools** screen and select **Back Annotate**. Select **Local Configuration** and then **Configure BACKANNO**. The screen shown in figure 7-13 displays.

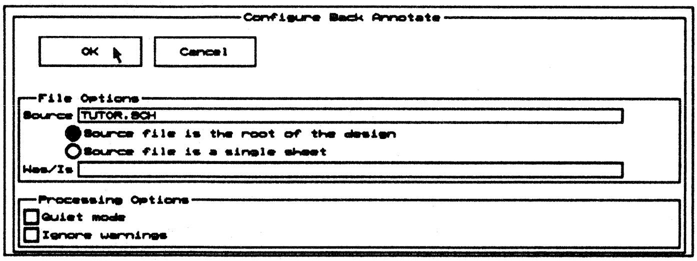


Figure 7-13. The ***Configure Back Annotate*** *screen.*

5. Enter the name of the file where **Back Annotate** gets the back annotation information—in this case **`TUTREF`**—in the **Was/Is** entry box.
6. Select **OK**. The **Schematic Design Tools** screen displays.
7. Run **Back Annotate**. As it processes, **Back Annotate** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   Back Annotate modifies the schematic file, TUTOR.SCH, to reflect the new reference designator values found in the WAS/IS file, TUTREF.
8. Run **Draft** on TUTOR.SCH to confirm that **Back Annotate** modified the reference designators on the schematic.

Running Create Bill of Materials

Create Bill of Materials creates a text file listing all parts in a single worksheet or an entire design.

Make a parts list

1. Select **Create Bill of Materials, Local Configuration,** and then **Configure PARTLIST.** The **Configure Create Bill of Materials** screen displays (figure 7-14).

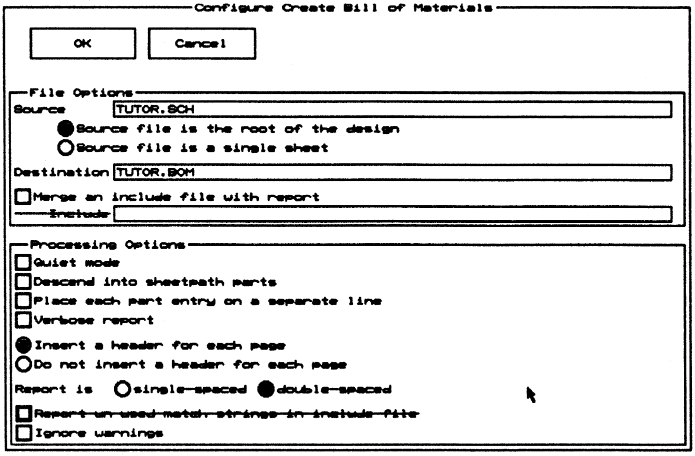


Figure 7-14. ***Configure Create Bill of Materials screen.***

2. In the **File Options** area of the **Configure Create Bill of Materials** screen, there are two filenames:

   ❖ In the **Source** entry box, the name of the worksheet from which the bill of materials is produced: `TUTOR.SCH`.

   This entry box tells **Create Bill of Materials** to use the worksheet file TUTOR.SCH to get the correct reference designator values.

   ❖ In the **Destination** entry box, the name of the file where **Create Bill of Materials** stores the report: `TUTOR.BOM`.

3. Select **Cancel.** The **Schematic Design Tools** screen displays.

4. Select **Create Bill of Materials** and then select **Execute.**

   As it processes, **Create Bill of Materials** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   The contents of TUTOR.BOM are shown in figure 7-15. Use **Edit File** to look at this file.

```
Digital clock schematic                      Revised:      June 11, 1992
                                             Revision: B
Bill Of Materials              June   11, 1992       22:16:45      Page    1

 Item  Quantity  Reference                   Part
_______________________________________________

    1       6    A1,A2,A3,A4,A5,A6           TIL309

    2       1    BT1                         9V

    3       1    C1                          47uF

    4       1    C2                          470uF

    5       1    C3                          22uF

    6       1    C4                          100uF

    7       4    C5,C6,C7,C8                 .01uF

    8       1    Q1                          LM7805

    9       1    R1                          91K

   10       6    R2,R3,R5,R6,R7,R8           10K

   11       1    R4                          100K

   12       1    S1                          Mode

   13       1    S2                          Reset

   14       1    U1                          74LS04

   15       3    U2,U3,U4                    22V10
```

Figure 7-15. The TUTOR design bill of materials.

Running Plot Schematic

The last task in this chapter is to plot the design you have created so far.

Use **Plot Schematic** to send designs to a plotter or, optionally, to a printer using the **Send output to printer** button.

△ *NOTE: This section focuses on running* ***Plot Schematic*** *and assumes you have configured* ***Schematic Design Tools*** *and connected your printer or plotter correctly. Many variables affect plotting. As with other mechanical processes, make sure your equipment—paper, pens, and so on—is in good working order and is set up properly.*

Follow these steps to configure and run **Plot Schematic:**

1. Select **Plot Schematic, Local Configuration,** and then **Configure PLOTALL.** The **Configure Plot Schematic** screen displays.

   If you are using a printer instead of a plotter, select **Send output to printer.**

△ *NOTE: When plotting a multiple-sheet design,* ***Plot Schematic*** *plots every worksheet in the design.*

2. If your plots are too large or too small, you can change the scale. Select **Manually set scale factor and/or X, Y offsets,** and the **Set Scale factor** entry box becomes accessible.

   Enter the scale factor, expressed in the form *n.nnn.* For example, if the plot is larger than the paper, you might run **Plot Schematic** at half scale by entering the number `0.500`.

3. Select **OK.**
4. Select **Plot Schematic** and then select **Execute** to run **Plot Schematic.**

Summary

In this chapter you learned how to use **Annotate Schematic, Update Field Contents, Select Field View, Check Electrical Rules, Create Netlist, Back Annotate, Create Bill of Materials,** and **Plot Schematic.** The next chapter describes three design structures and shows how to create and use them.

Structuring your design

In this chapter you look at three types of design structures: a flat design, a simple hierarchy, and a complex hierarchy.

A flat design

A flat design is one in which all of the worksheets are linked at the same level. Use a flat design for relatively small designs with no more than ten worksheets. Since you must manage all of the interconnections between the worksheets of a flat design by the names assigned to the module ports, large designs consisting of many worksheets or repetitive logic are more easily managed using hierarchical structures.

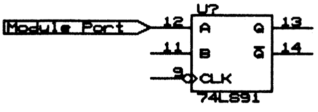
Flat designs are linked using module ports and the |LINK (read as "pipe-link") command.

The |LINK command

```
|LINK
|OTHER
```

You use the |LINK command on the root worksheet to inform the various schematic tools which worksheets are in a flat design. In figure 8-1, the |LINK command links the root worksheet, PROJECT.SCH, to OTHER.SCH. The filename of the root worksheet consists of the name of the design and a .SCH extension.

Module ports



Module ports are graphic objects that indicate where signals are conducted between worksheets. Module ports that have identical names are considered to be electrically connected. In figures 8-1 and 8-2, CLEAR, LOAD, and RCO are connected; Hi[0..3] and Lo[0..3] are not connected.

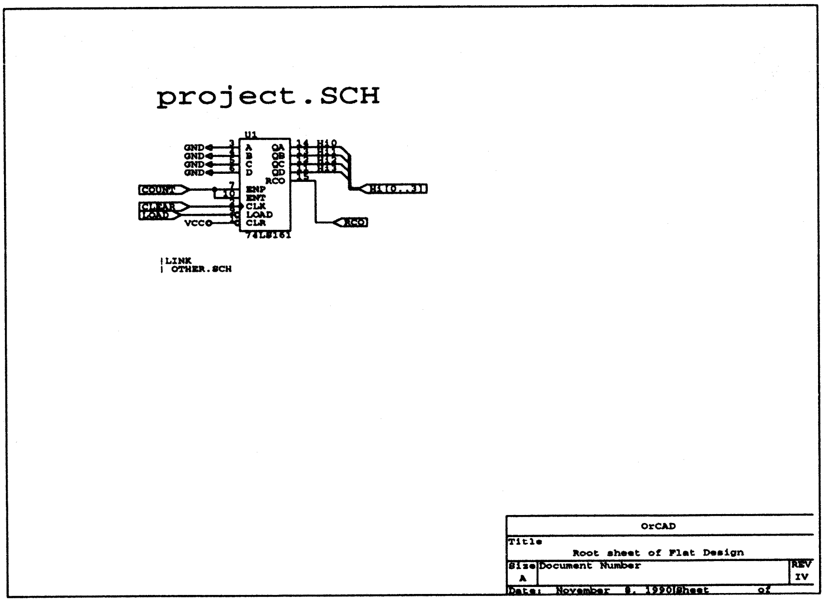


Figure 8-1. Root worksheet of flat design.

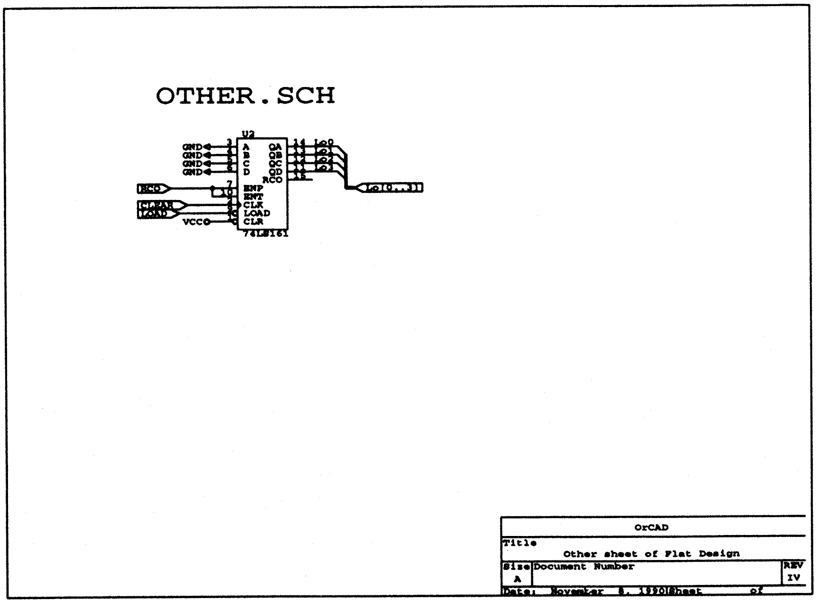


Figure 8-2. Other worksheet of flat design.

Creating new designs

The remainder of this chapter covers simple and complex hierarchies. Example files are included in the TUTOR design. Before you proceed through the rest of this chapter, you need to create two new design directories: CMOSCPU and 4BIT. You will copy the example files from the TUTOR design into these new design directories.

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button. The **Design Management Tools** menu displays.
2. Select **Design Management Tools.** The **Design View** portion of the **Design Management Tools** screen displays.
3. Select **Create Design.** The **Create Design** screen displays. Make sure that **Copy all files** is selected.
4. Enter **CMOSCPU** in the **New design name** entry box. Select **OK.** The prompt "Working . . ." and several messages display at the top left corner of the screen. After the new design name is created, the **Create Design** screen is dismissed and the design name, CMOSCPU, appears in the **Design** list box.
5. Repeat steps 3 and 4, but this time create a new design named 4BIT.
6. Select **TUTOR** from the **Design** list box. The files you need to copy are in the TUTOR design.
7. Select **File View** to see the TUTOR files.
8. Select **Copy File.** The **Copy File** screen displays.
9. Using the information in table 8-1, select the source file from the **Files** list box, enter the destination in the **Destination** entry box, and then select **OK.** Repeat this procedure for each file in the table.

Source file name	*Destination*
CMOSCPU.SCH	..\CMOSCPU\CMOSCPU.SCH
MEMORY.SCH	..\CMOSCPU\MEMORY.SCH
POWER.SCH	..\CMOSCPU\POWER.SCH
4BIT.SCH	..\4BIT\4BIT.SCH
FULLADD.SCH	..\4BIT\FULLADD.SCH
HALFADD.SCH	..\4BIT\HALFADD.SCH

Table 8-1. Files to be copied and their destinations.

10. When you have copied all the files to their appropriate destinations, select **CANCEL**. The **Copy File** screen is dismissed.

11. Select **Design View**. Reset the current design to CMOSCPU and then select **OK**. The main screen displays.

A simple hierarchical design

The layered arrangement created by placing worksheets inside other worksheets is called a hierarchical design. This section describes the structure of a simple, three-worksheet hierarchical design.

Sheet symbols represent other worksheets in a hierarchical design. Each sheet symbol represents a subsheet. Sheet symbols may be placed at any level of the hierarchy.

The following example is a simple hierarchy because the two sheet symbols refer to different worksheets. In complex hierarchies, any number of sheet symbols can refer to the *same* worksheet.

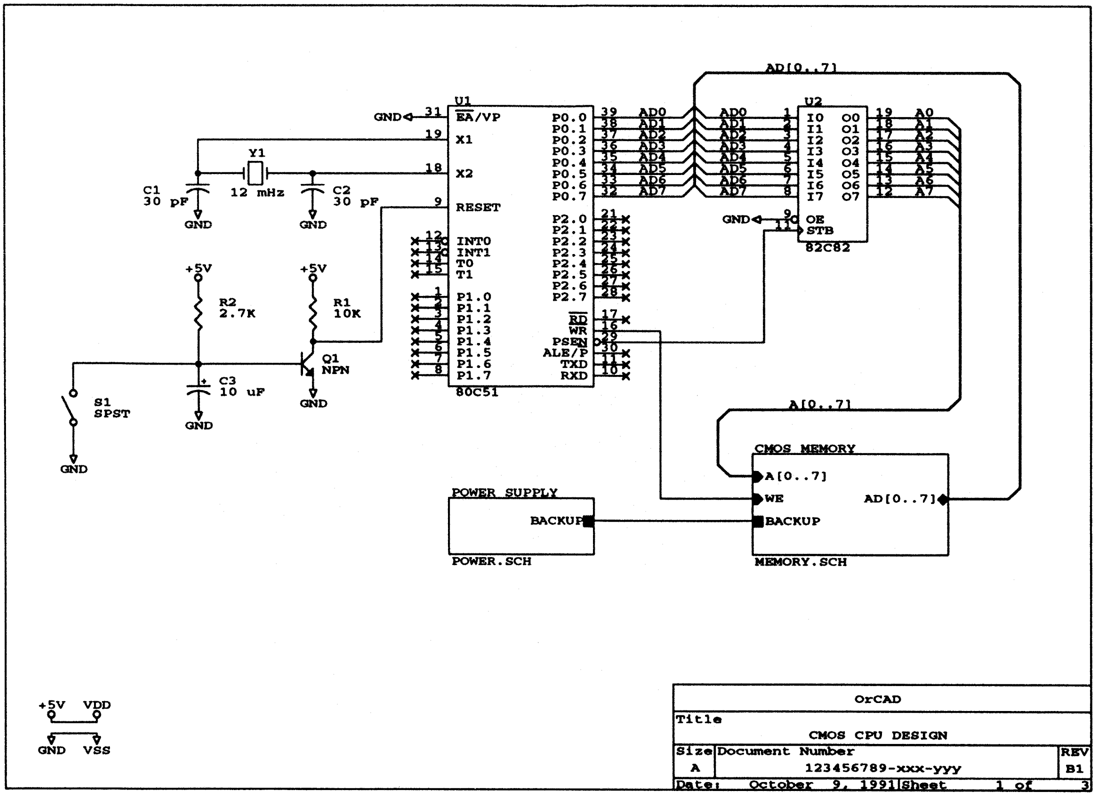


Figure 8-3. The root worksheet of the CMOS CPU design.

Libraries

The CMOS CPU design uses the following libraries:

- ANALOG2.LIB
- DEVICE.LIB
- INTEL.LIB
- MEMORY.LIB
- TTL.LIB

All of these libraries must be configured. Check the **Configured Libraries** list box in the **Library Options** area of the **Configure Schematic Design Tools** screen to determine their status. Follow these steps to configure a library:

1. From the **Schematic Design Tools** screen, select **Draft** and select **Configure Schematic Tools.** The **Configure Schematic Design Tools** screen displays.
2. Pan down to the **Library Options** area.
3. If any of the above mentioned libraries do not appear in the **Configured Libraries** list box, configure them now. Select the desired library from the **Available Libraries** list box, and then select **>Insert>**.
4. Select **OK** to save any configuration changes.

The root worksheet CMOSCPU.SCH

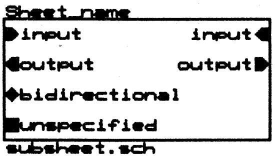
Figure 8-3 shows the root worksheet of this simple hierarchy. The design is called CMOS CPU. The root worksheet filename is CMOSCPU.SCH.

The root worksheet contains a descriptive name in the title block. A title helps identify a worksheet, but it is not required. The title is independent of the filename. A worksheet is identified by **Draft** and the operating system by the worksheet's filename.

The root worksheet of the CMOS CPU design contains:

- An 80C51 and an 82C82 part
- Discrete analog parts: a transistor, capacitors, resistors, and so on
- Two sheet symbols: POWER SUPPLY and CMOS MEMORY
- Power and ground symbols
- Wires and buses connecting the parts

Sheet symbols



Each sheet symbol has a *name* and a *filename*. The name and filename are separate. The sheet symbol name displays above the sheet symbol; the filename displays below the sheet symbol. The sheet symbol filename has to be identical to the name of the file containing the associated worksheet.

The two sheet symbols in figure 8-3 were placed on the root worksheet using the **PLACE Sheet** command. When you place a sheet symbol, **Draft** automatically assigns it a unique filename generated from the date and time of day on your computer. You can accept this unique (but not very descriptive) filename or change it to a filename of your choice. To change the filename, place the pointer inside the sheet symbol and select **EDIT Edit Filename.** The prompt "Filename?" followed by the sheet symbol's current filename displays. Enter the new filename.

In this example, the CMOS MEMORY sheet symbol was assigned the filename MEMORY.SCH, and the POWER SUPPLY sheet symbol was assigned the filename POWER.SCH. MEMORY.SCH refers to the worksheet on which the circuit's memory is located. POWER.SCH refers to the worksheet on which the system's power supply is located.

Sheet nets

The CMOS MEMORY sheet symbol contains four sheet nets: A[0..7], WE, BACKUP, and AD[0..7]. These sheet nets were placed into the sheet symbol using the **PLACE Sheet Add-NET** command. (Sheet nets are *not* module ports. See *Using sheets and parts to point to another worksheet* in *Chapter 9:Tips and Techniques* for more information.)

The A[0..7] sheet net is connected to a bus with eight members. The bus members are labeled A0 through A7.

While sheet nets do not need to have the same *prefixes* as the bus labels to which they connect ("A" in this example), the sheet nets *must* specify the same *ranges* as the bus labels to which they connect ("[0..7]").

For sheet nets and module ports, there should be no space between the prefix and suffix portions of the names.

Power objects

Power objects represent connections from the outside world to the pins in a part. Unless otherwise specified, power objects are global in scope; they connect to all other signals of the same name.

On the root worksheet a power object named +5V connects to a power object named V_{DD}. This connects the +5V supply to the V_{DD} pins of the 80C51 and the 82C82 parts. Likewise, a power object named GND connects to a power object named V_{SS}. This connects power ground to the V_{SS} pins of the 80C51 and the 82C82 parts.

Nested schematic worksheets

Once the sheet symbols for the nested logic are completed, you then create the worksheets to which these sheet symbols refer.

△ *NOTE: You don't have to create the root worksheet of the hierarchy before creating the nested worksheets; however, a top-down design methodology is recommended.*

Display the CMOS MEMORY worksheet

1. Run **Draft** and select **QUIT Enter Sheet.**
2. Place the pointer inside the CMOS MEMORY sheet symbol and select **Enter. Draft** displays the CMOS MEMORY worksheet (figure 8-4).

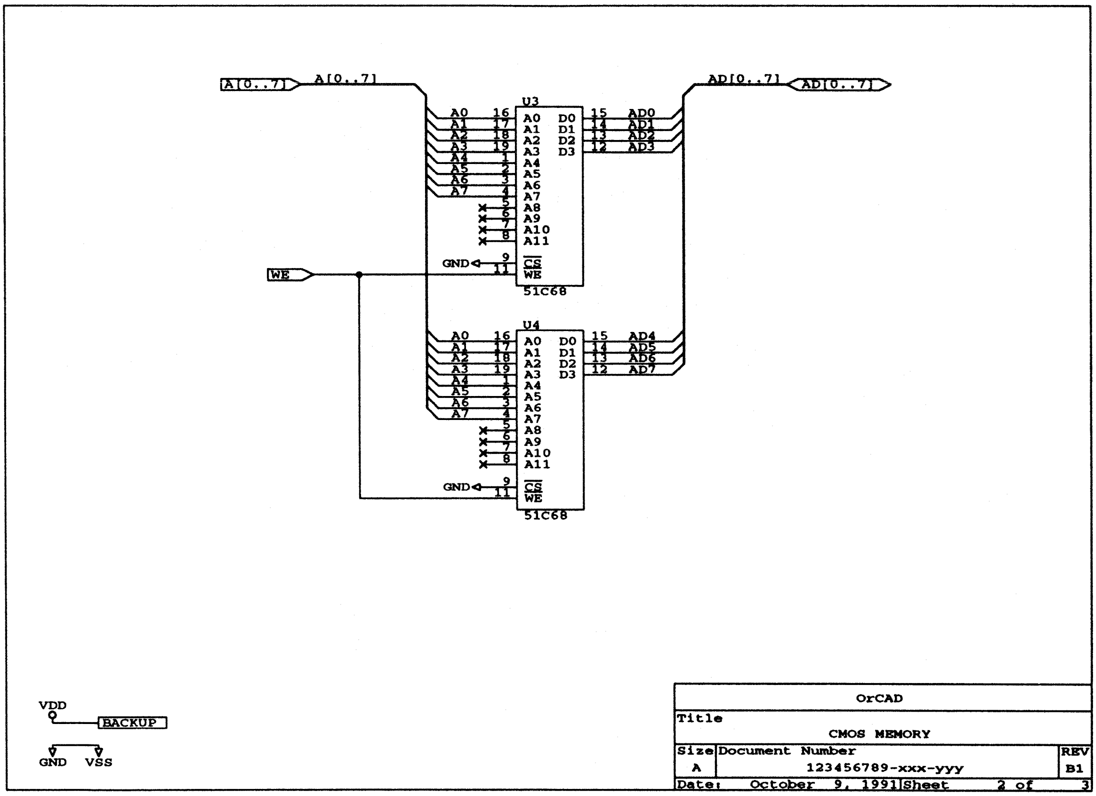


Figure 8-4. CMOS MEMORY worksheet.

The CMOS MEMORY worksheet is the schematic for the circuit's memory.

Notice the four module ports: A[0..7], AD[0..7], WE, and BACKUP. They connect to identically named sheet nets located in the CMOS MEMORY sheet symbol on the root worksheet.

Notice the buses. Buses are automatically connected to module ports with labels having the same name and range—module port A[0..7] connects to a bus labeled A[0..7].

Finally, notice the power objects. The V_{DD} power object connects to the module port named BACKUP. This isolates the power in the worksheet from any other V_{DD} power objects in the design.

The GND power object connects to the V_{SS} power object. This shorts the GND power net and the V_{SS} power net together. This allows any pin with a GND symbol to be connected to the V_{SS} power net.

When you are finished reviewing the CMOS MEMORY worksheet, return to the root worksheet by selecting **Leave** from the **QUIT Enter Sheet** command line (if it is displayed at the top of the screen) or by selecting the **QUIT Leave Sheet** command.

Display the POWER SUPPLY worksheet

1. If the **QUIT Enter Sheet** command line is not displayed at the top of the screen, select the **QUIT Enter Sheet** command.
2. Place the pointer inside the POWER SUPPLY sheet symbol and select **Enter**. Figure 8-5 shows the POWER SUPPLY worksheet.

The POWER SUPPLY worksheet is the schematic for the power supply circuitry.

Notice the module port named BACKUP. The BACKUP module port makes the *logical* connection to the sheet net named BACKUP. The BACKUP sheet net is in the POWER SUPPLY sheet symbol on the root worksheet of the CMOS CPU design (see figure 8-3).

The BACKUP module port makes the *electrical* connection to the BACKUP module port on the CMOS MEMORY worksheet (see figure 8-4).

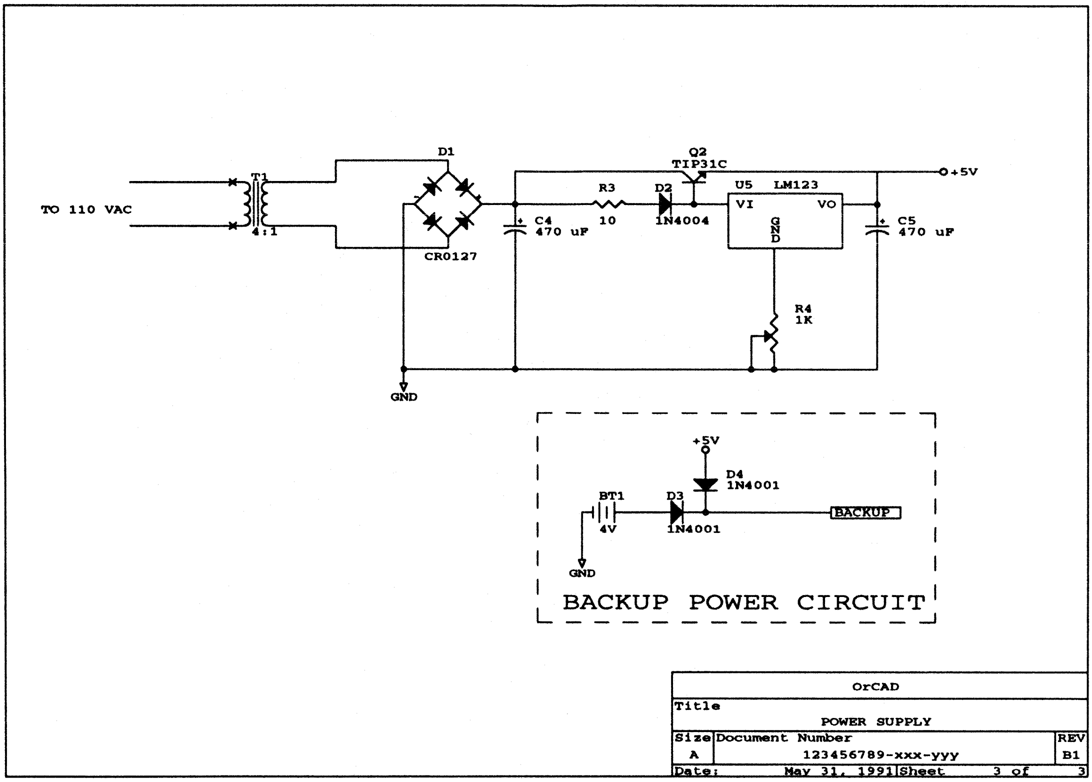


Figure 8-5. POWER SUPPLY worksheet.

When you are finished reviewing the POWER SUPPLY worksheet, return to the root worksheet by selecting **Leave** from the **QUIT Enter Sheet** command line, and then pressing <Esc> to dismiss the command line. (If the **QUIT Enter Sheet** command line is not displayed at the top of the screen, select the **QUIT Leave Sheet** command.) Select **QUIT Abandon Edits** to return to the **Schematic Design Tools** screen.

The next sections in this chapter discuss the following tools: **Annotate Schematic, Check Electrical Rules, Show Design Structure,** and **Create Bill of Materials.**

Using Annotate Schematic on a simple hierarchy

Once a design is complete, you run **Annotate Schematic** to assign unique values to the part reference designators.

Follow these steps to annotate the simple hierarchy represented by the worksheet, CMOSCPU.SCH:

```
Annotate Schematic
Execute
Local Configuration
Assign Hot Key
Show Version
Configure Schematic Tools
Help
```

1. Select **Annotate Schematic** on the **Schematic Design Tools** screen. The menu at right displays.
2. Select **Local Configuration** and then **Configure ANNOTATE.** The **Configure Annotate Schematic** screen displays.
3. Check to make sure that the **Source** entry box contains the filename CMOSCPU.SCH. If not, enter **CMOSCPU.SCH** in the **Source** entry box.
4. Check to make sure that **Source file is the root of the design** is selected.
5. Select **OK** to save the configuration.
6. Select **Annotate Schematic** and then select **Execute.**

   As it processes, **Annotate Schematic** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

When **Annotate Schematic** is done, the reference designators for each part in the worksheet have new, unique values. You may want to use **Draft** to view the new reference designator values on the CMOS CPU worksheet.

Using Check Electrical Rules on a simple hierarchy

After the worksheets are annotated, the design should be checked for electrical rule violations. **Check Electrical Rules** checks for several problems associated with a design, including open input pins, shorts, and bus contention.

Run **Check Electrical Rules** to check for any electrical rules violations in the design. For instructions, refer to the earlier discussion of how to run **Check Electrical Rules.** After you run **Check Electrical Rules,** you may review the error report with **Edit File.**

Figure 8-6 shows how the error report looks for the CMOSCPU.SCH design.

Warnings

Check Electrical Rules flags certain conditions possibly overlooked when your design was created. These *WARNINGS* are not critical errors. The warnings in this example are acceptable, because the power supplies were intentionally connected in the design.

Errors

Normally, if **Check Electrical Rules** reports *ERRORS* in a design, you should correct them before running other tools. In this example, however, all warnings are acceptable and other tools may be run.

```
CMOSCPU.SCH

Electrical Rules Check Report
CMOS CPU DESIGN  Revised:   October  9, 1991
123456789-xxx-yyy          Revision: B1
OrCAD




WARNING: POWER Supplies are CONNECTED GND <-> VSS
WARNING: POWER Supplies are CONNECTED VDD <-> +5V

MEMORY.SCH

Electrical Rules Check Report
CMOS MEMORY  Revised:   October  9, 1991
123456789-xxx-yyy          Revision: B1
OrCAD




WARNING: POWER Supplies are CONNECTED VSS <-> GND

POWER.SCH

Electrical Rules Check Report
POWER SUPPLY  Revised:      May 31, 1991
123456789-xxx-yyy          Revision: B1
OrCAD
```

Figure 8-6. The error report produced by ***Check Electrical Rules*** *for CMOSCPU.SCH.*

Using Show Design Structure on a simple hierarchy

Use **Show Design Structure** to obtain a text file listing the worksheets in a hierarchy. This tool is helpful for organizing a hierarchy containing many worksheets. Follow these steps to run **Show Design Structure:**

1. Select **Show Design Structure** on the **Schematic Design Tools** screen, and then select **Local Configuration** and **Configure TREELIST.** The **Configure Show Design Structure** screen displays.
2. Check to make sure that the **Source** entry box contains the filename CMOSCPU.SCH.
3. Check to make sure that the **Destination** entry box contains the filename CMOSCPU.TWG.

   Show Design Structure is now configured to create a schematic structure list for CMOSCPU.SCH and save the results in a text file, CMOSCPU.TWG.
4. Select **OK** to save any changes to the configuration.
5. Select **Show Design Structure** and then select **Execute.**

   As it processes, **Show Design Structure** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   Use **Edit File** to examine the CMOSCPU.TWG text file. The figure below shows the information stored in CMOSCPU.TWG.

```
<<<ROOT>>>
[CMOSCPU.SCH]     October  9, 1991
  CMOS MEMORY
   [MEMORY.SCH]     October  9, 1991
  POWER SUPPLY
   [POWER.SCH]         May 31, 1991
```

All worksheet filenames are enclosed within brackets, as in [*filename*]. Next to the filename is the date the worksheet was last modified. **Show Design Structure** lists sheet symbol names above the filenames of the worksheets to which they refer.

In this example, the root worksheet filename is CMOSCPU.SCH. Below the root worksheet filename are sheet symbol names and the filenames of the worksheets to which they refer. The sheet symbol named CMOS MEMORY refers to the worksheet with the filename MEMORY.SCH. The sheet symbol named POWER SUPPLY refers to the worksheet with the filename POWER.SCH.

Using Create Bill of Materials on a simple hierarchy

Create Bill of Materials creates a list of parts for all types of design structures. In this example, **Create Bill of Materials** is used on the simple hierarchy, CMOSCPU.SCH. Follow these steps to run **Create Bill of Materials:**

1. Select **Create Bill of Materials** on the **Schematic Design Tools** screen and then select **Local Configuration** and **Configure PARTLIST**. The **Configure Create Bill of Materials** screen displays.
2. Check to make sure that the **Source** entry box contains the filename CMOSCPU.SCH.
3. Check to make sure that **Source file is the root of the design** is selected.
4. Check to make sure that the **Destination** entry box contains the filename CMOSCPU.BOM.
5. Select **OK** to save any changes to the configuration.
6. Select **Create Bill of Materials** and then select **Execute.**

   As it processes, **Create Bill of Materials** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.

   To examine the text file created by **Create Bill of Materials,** use **Edit File.** Figure 8-7 shows the information stored in the text file CMOSCPU.BOM.

```
CMOS CPU DESIGN                                    Revised:    October  9, 1991
123456789-xxx-yyy                                  Revision: B1
Bill Of Materials                      June   12, 1992     11:40:57    Page   1
```

Item	Quantity	Reference	Part
1	1	BT1	4V
2	2	C1,C2	30 pF
3	1	C3	10 uF
4	2	C4,C5	470 uF
5	1	D1	CR0127
6	1	D2	1N4004
7	2	D3,D4	1N4001
8	1	Q1	NPN
9	1	Q2	TIP31C
10	1	R1	10K
11	1	R2	2.7K
12	1	R3	10
13	1	R4	1K
14	1	S1	SPST
15	1	T1	4:1
16	1	U1	80C51
17	1	U2	82C82
18	2	U3,U4	51C68
19	1	U5	LM123
20	1	Y1	12 mHz

Figure 8-7. Bill of materials for CMOSCPU.SCH.

A complex hierarchical design

This section describes a three-sheet complex hierarchy. Complex hierarchies are very useful when designing common logic blocks that are repeated. Figure 8-8 shows the root worksheet for the 4-bit adder design.

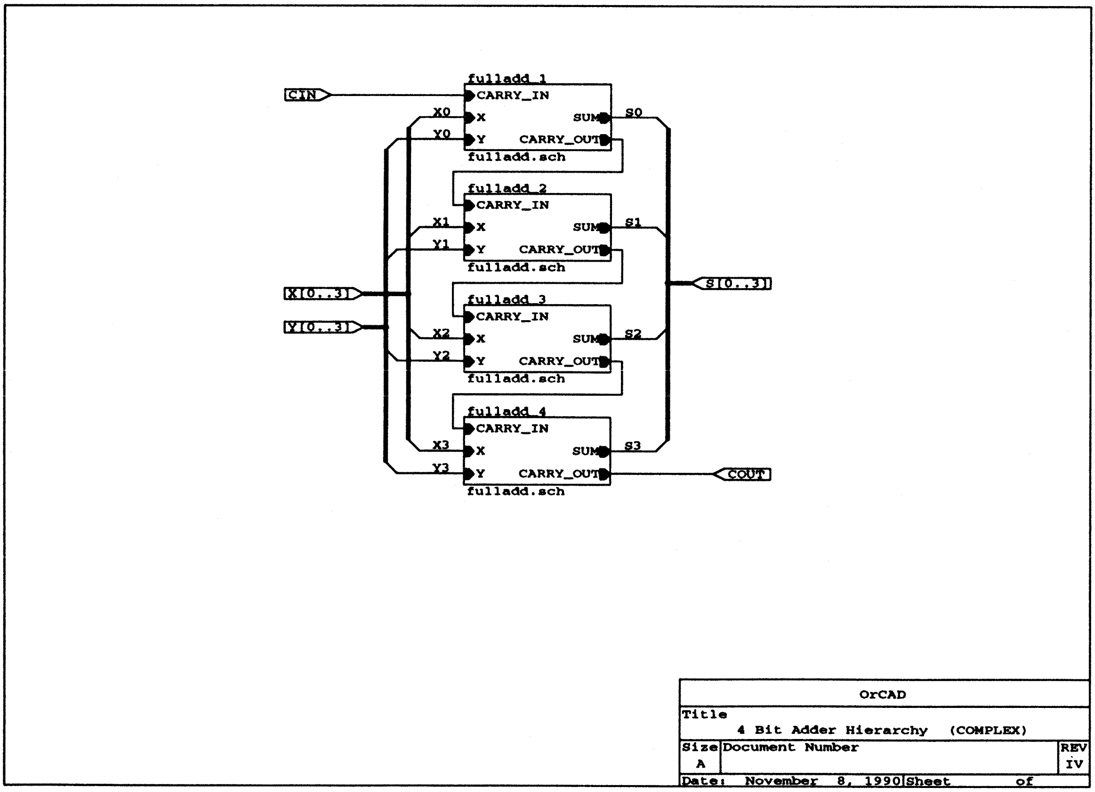


Figure 8-8. The root worksheet of the 4-bit adder design.

The 4-bit adder root worksheet

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button, and then select **Design Management Tools.** The **Design View** portion of the **Design Management Tools** screen displays.
2. Select **4BIT** from the **Design** list box.
3. Select **OK.** The **Design View** screen is dismissed and the main screen displays.
4. Select **Schematic Design Tools** and then select **Execute.** The **Schematic Design Tools** screen displays.
5. Select **Draft** and then select **Execute.** The 4BIT root worksheet displays (figure 8-8).

The 4-bit adder design is a three-sheet complex hierarchy. The root worksheet contains four identical sheet symbols: fulladd_1, fulladd_2, fulladd_3, and fulladd_4.

These sheet symbols refer to four identical full adders. Because they are identical, it is not necessary to create a separate worksheet for each one. Instead, create just one full-adder worksheet and assign the full-adder worksheet's filename to all four sheet symbols. Notice that all of the sheet symbols on the root worksheet have the filename FULLADD.SCH.

The full-adder worksheet

Follow these steps to display the full-adder worksheet.

1. Select **QUIT Enter Sheet.**
2. Place the pointer inside one of the sheet symbols and select **Enter. Draft** displays the worksheet referred to by the selected sheet symbol. Figure 8-9 shows the full-adder worksheet.

The full-adder worksheet contains two sheet symbols: halfadd_A and halfadd_B. These sheet symbols refer to two identical half adders. Because they are identical, it is not necessary to create a separate worksheet for each. Just as the four full-adder sheet symbols in the root worksheet refer to the full-adder worksheet for their logic, the two half-adder sheet symbols refer to a single half-adder worksheet for *their* logic.

Each module port in the FULLADD.SCH worksheet is named to connect to the sheet nets in the 4BIT.SCH worksheet, one level up in the hierarchy.

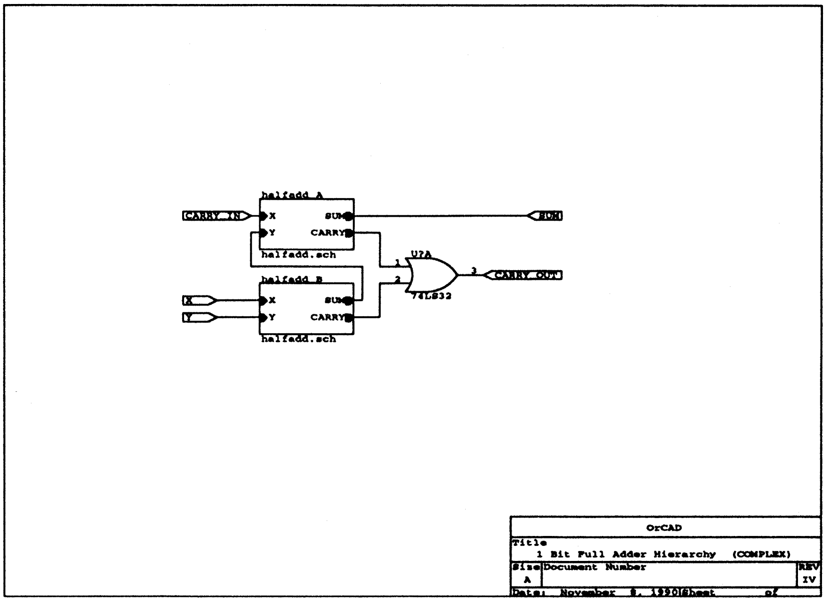


Figure 8-9. Full-adder worksheet.

The half-adder worksheet

Follow these steps to display the half-adder worksheet:

1. If the **QUIT Enter Sheet** command line is not displayed at the top of the screen, select the **QUIT Enter Sheet** command.

2. Place the pointer inside one of the sheet symbols and select **Enter. Draft** displays the worksheet referred to by the selected sheet symbol. Figure 8-10 shows the half-adder circuit.

Each module port in the HALFADD.SCH worksheet is named to connect to the sheet nets in the FULLADD.SCH worksheet, one level up in the hierarchy.

When you are finished reviewing the half-adder worksheet, return to the root worksheet by selecting **Leave** from the **QUIT Enter Sheet** command line, and then pressing <Esc> to dismiss the command line. (If the **QUIT Enter Sheet** command line is not displayed at the top of the screen, select the **QUIT Leave Sheet** command.) Select **QUIT Abandon Edits** to return to the **Schematic Design Tools** screen.

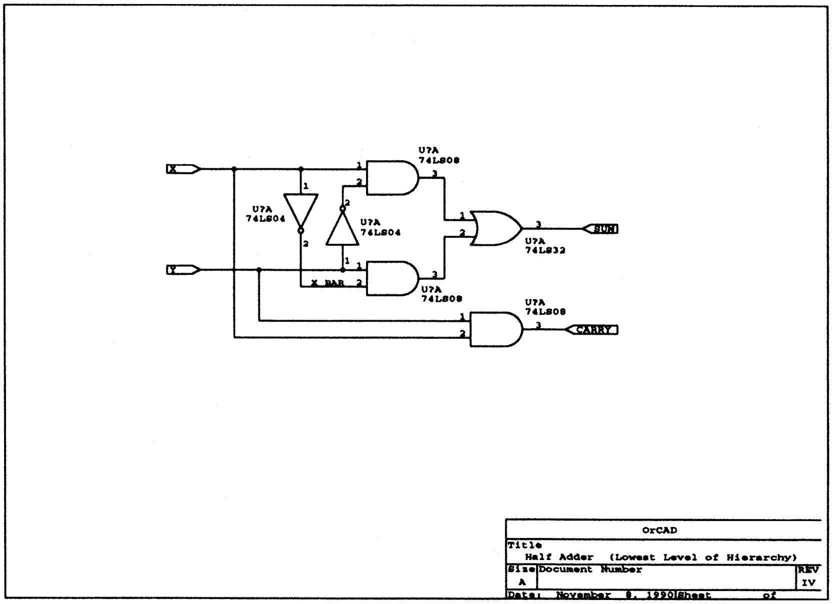


Figure 8-10. Half-adder worksheet.

Using Show Design Structure on a complex hierarchy

Use **Show Design Structure** to obtain a text file listing the worksheets in a hierarchy. This tool is helpful for organizing a hierarchy containing many worksheets.

Follow the steps given in *Using Show Design Structure on a simple hierarchy* earlier in this chapter, but substitute the filename 4BIT for CMOSCPU.

Use **Edit File** to examine the 4BIT.TWG text file. The figure below shows the information stored in 4BIT.TWG:

```
<<<ROOT>>>
[4BIT.SCH]   November  8, 1990
  fulladd_1
  [FULLADD.SCH]   November  8, 1990
    halfadd_A
     [HALFADD.SCH]   November  8, 1990
    halfadd_B
     [HALFADD.SCH]   November  8, 1990
  fulladd_2
  [FULLADD.SCH]   November  8, 1990
    halfadd_A
     [HALFADD.SCH]   November  8, 1990
    halfadd_B
     [HALFADD.SCH]   November  8, 1990
  fulladd_3
  [FULLADD.SCH]   November  8, 1990
    halfadd_A
     [HALFADD.SCH]   November  8, 1990
    halfadd_B
     [HALFADD.SCH]   November  8, 1990
  fulladd_4
  [FULLADD.SCH]   November  8, 1990
    halfadd_A
     [HALFADD.SCH]   November  8, 1990
    halfadd_B
     [HALFADD.SCH]   November  8, 1990
```

Notice that there are a number of references in this report to FULLADD.SCH and HALFADD.SCH. The 4-bit adder design, a complex hierarchy of only three worksheets, expands to thirteen worksheet references. Again, the advantage of complex hierarchical design structures is that, during the design phase, all of the repeated logic needs to be drawn only once.

Converting a complex hierarchy to a simple hierarchy

While a complex hierarchy is very useful in the design phase, it is not practical for some aspects of the design cycle. This is especially true when a design is to be turned into a printed circuit board. All of the design must then be simplified—converted to a simple hierarchy. This is necessary because all of the parts in the design must be assigned unique reference designators. It would be quite difficult to have a number of parts labeled U17 on the board and have to refer to the complex hierarchical schematic to determine which was which.

Design Management Tools includes the tool **Complex to Simple.** This tool creates a new design and builds a new version of the complex hierarchy, a version in which each sheet symbol refers to a unique file.

Follow these steps to run **Complex to Simple**:

1. On the **Schematic Design Tools** screen, click the title bar or any place that is not a button, and then select **Design Management Tools.** The **Design View** portion of the **Design Management Tools** screen displays.
2. Select **Complex to Simple.** The **Complex to Simple** screen displays.
3. Check to make sure that the **Source design** entry box contains the name 4BIT.
4. Enter `S4BIT` in the **Destination design** entry box.
5. Select **OK. Design Management Tools** builds the new design directory and converts the 4BIT design to S4BIT. As it processes, **Design Management Tools** displays "Working . ." and several messages at the top left corner of the screen. Then, **Design Management Tools** scrolls status messages three lines at a time in the monitor box at the bottom of the **Schematic Design Tools** screen.
6. Select **Cancel** when the process is complete.
7. Notice that S4BIT is now the current design. Select **OK.** The main screen displays.

Viewing the S4BIT design

Select **Draft** and then **Execute**. Notice that the four occurrences of the filename FULLADD.SCH are changed to FULLADD.SCH, FULLADDA.SCH, FULLADDB.SCH, and FULLADDC.SCH. Also notice that the eight occurrences of the filename HALFADD.SCH are changed to HALFADD.SCH and HALFADDA.SCH through HALFADDG.SCH.

Running Annotate Schematic on the S4BIT design

As with any new design, you should run **Annotate Schematic** before running the other tools. Other tools report information about the worksheet file, and if you run **Annotate Schematic** first, you ensure that reported information includes the updated reference designators.

The following figures show the simplified design after it is annotated using **Annotate Schematic**.

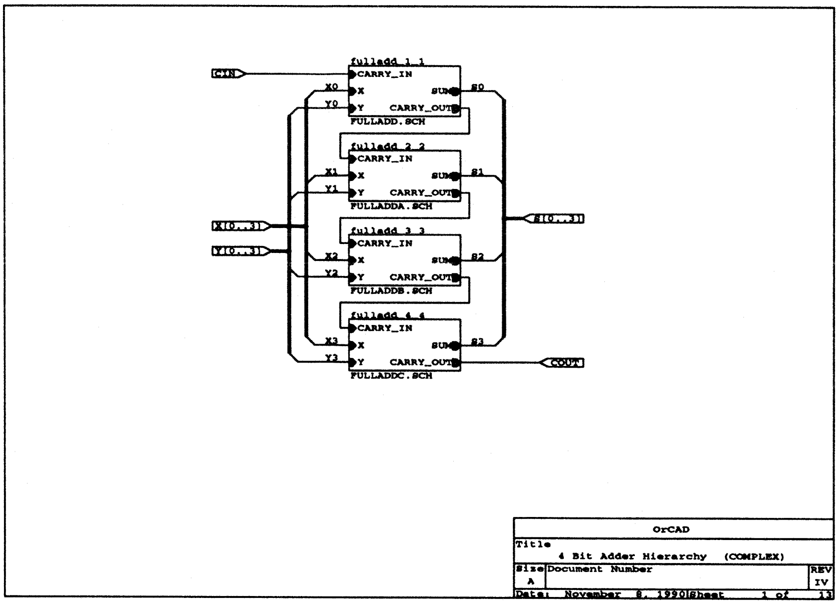


Figure 8-11. 4BIT.SCH schematic.

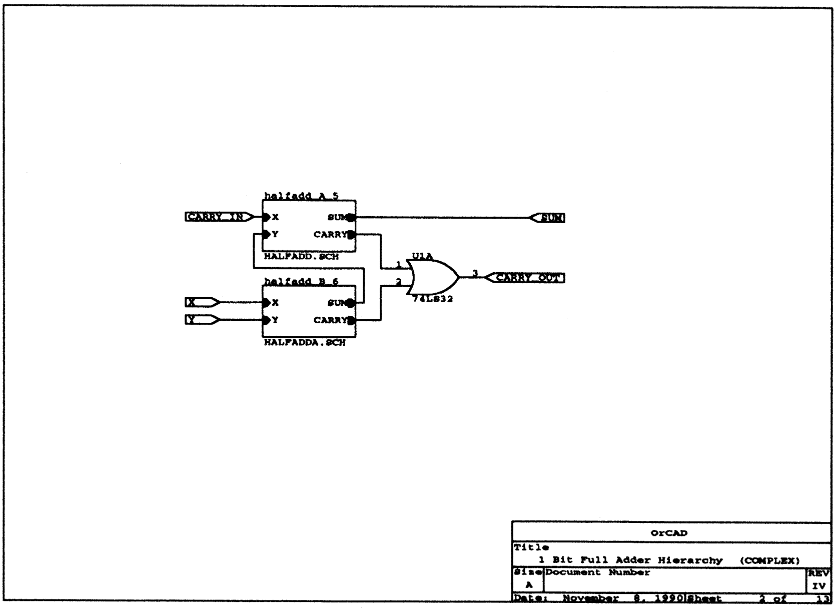


Figure 8-12. FULLADD.SCH schematic.

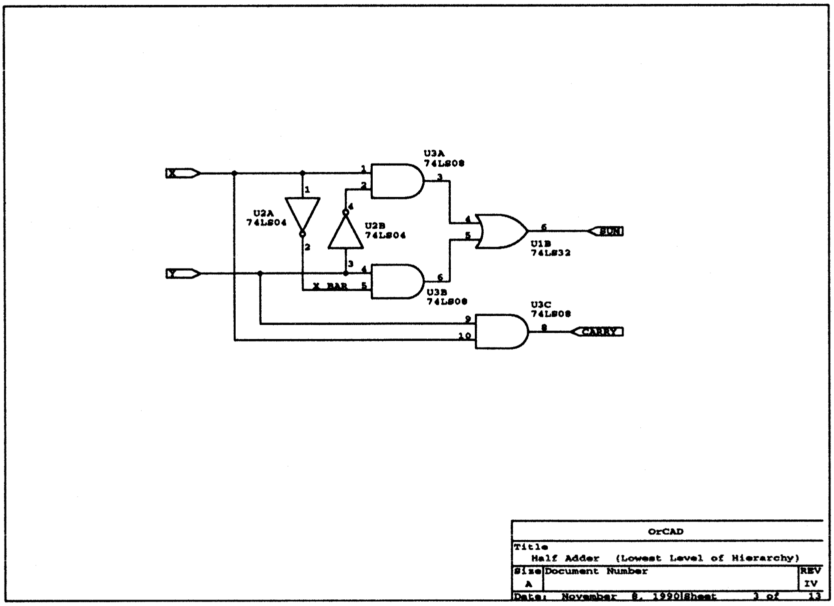


Figure 8-13. HALFADD.SCH schematic.

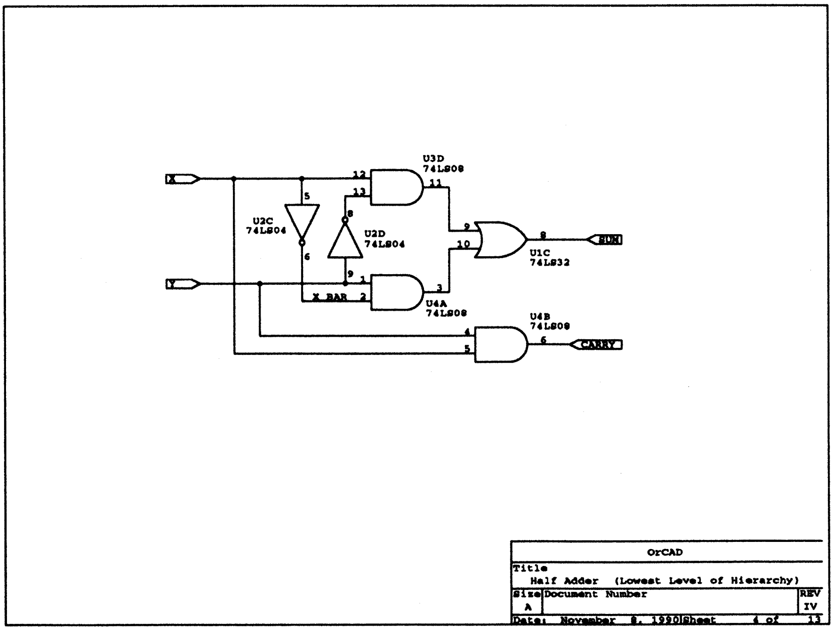


Figure 8-14. HALFADDA.SCH schematic.

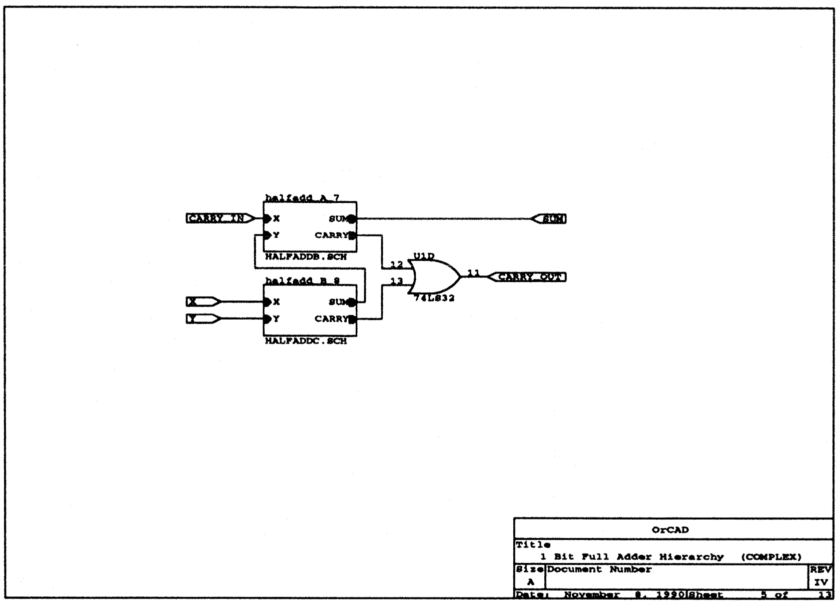


Figure 8-15. FULLADDA.SCH schematic.

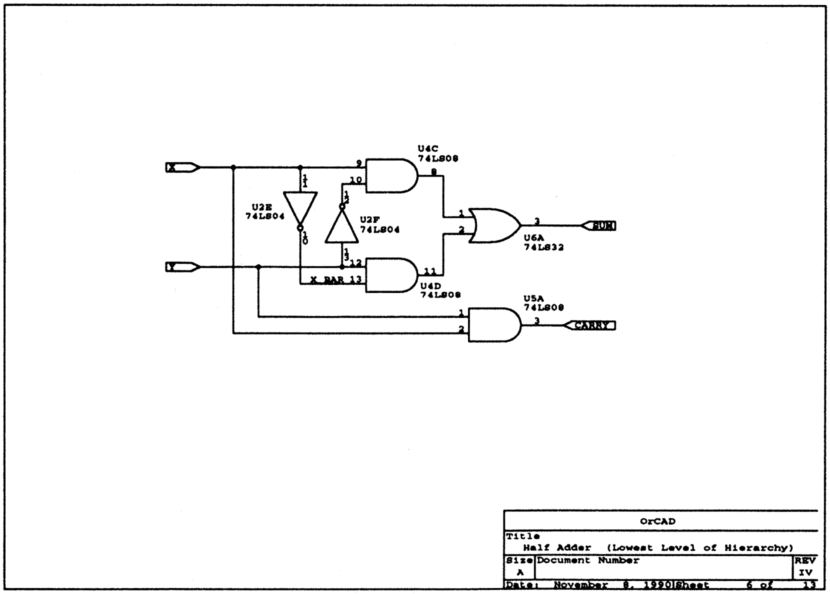


Figure 8-16. HALFADDB.SCH schematic.

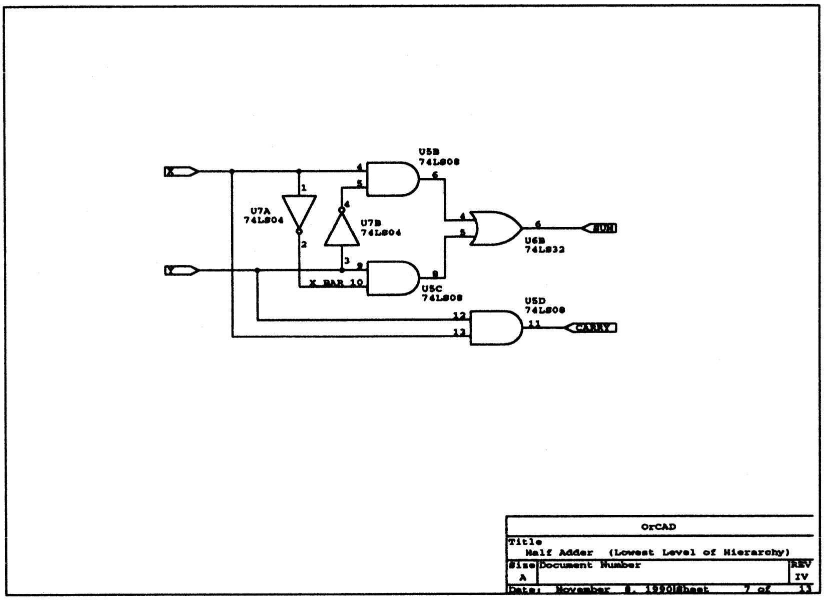


Figure 8-17. HALFADDC.SCH schematic.

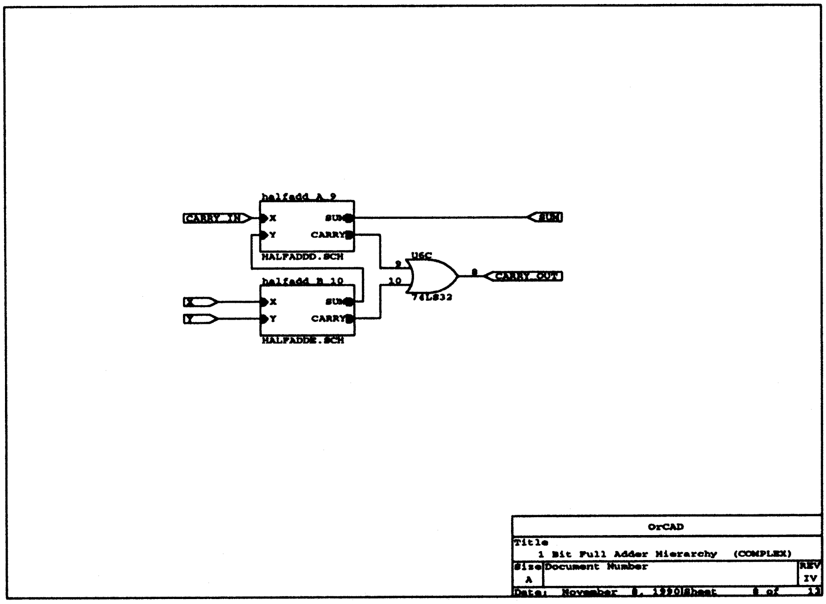


Figure 8-18. FULLADDB.SCH schematic.

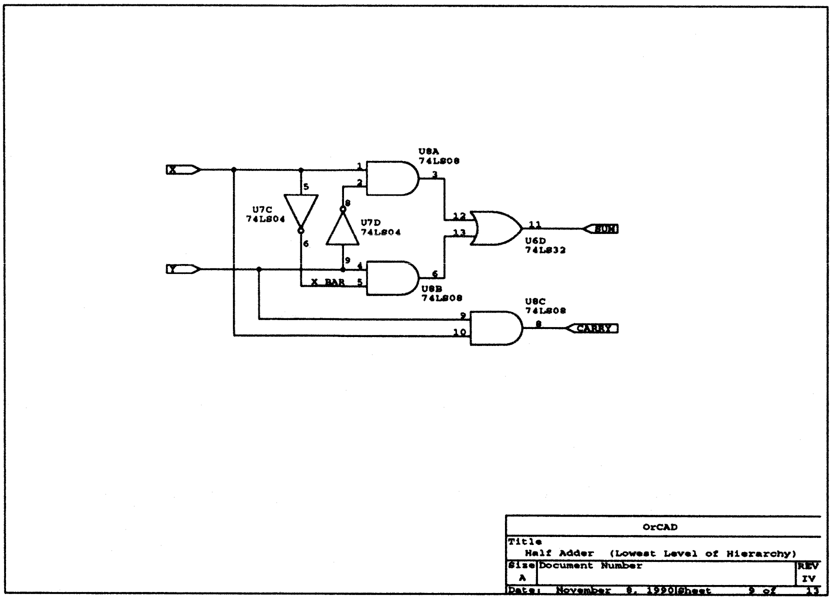


Figure 8-19. HALFADDD.SCH schematic.

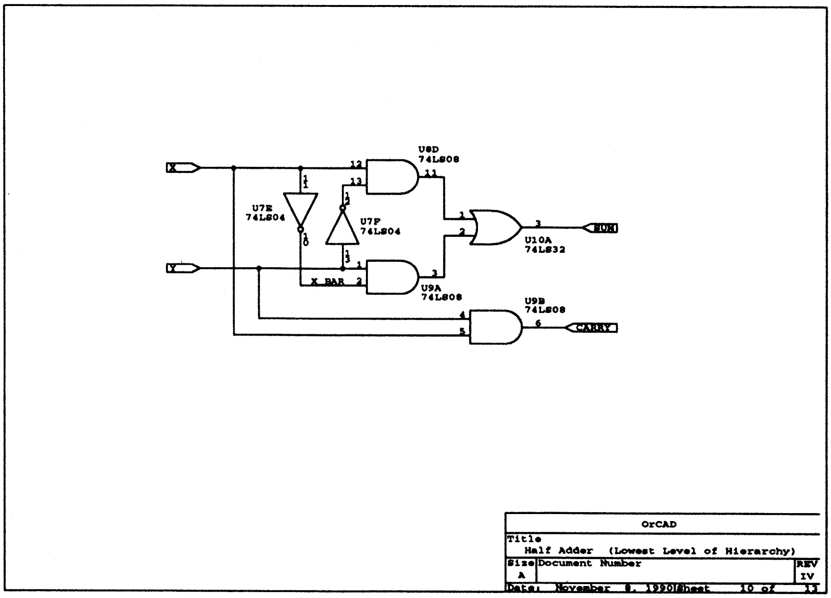


Figure 8-20. HALFADDE.SCH schematic.

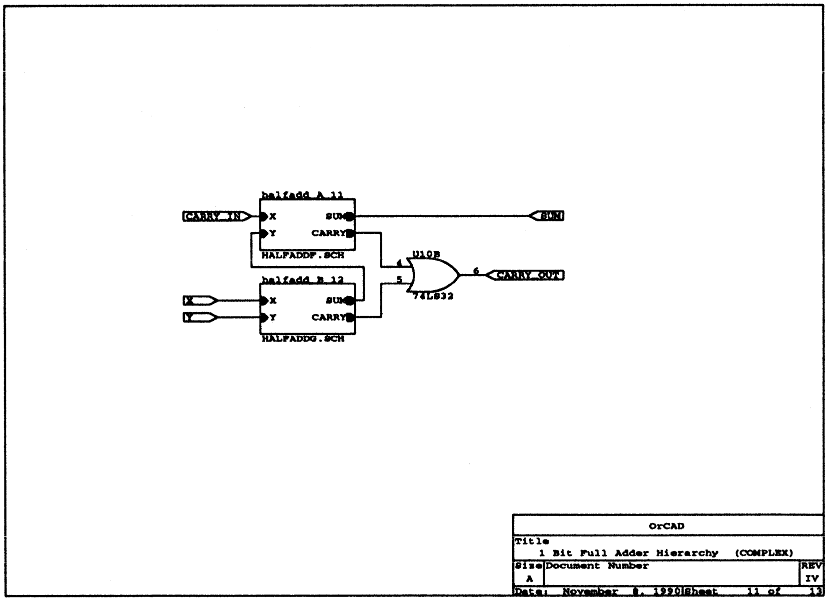


Figure 8-21. FULLADDC.SCH schematic.

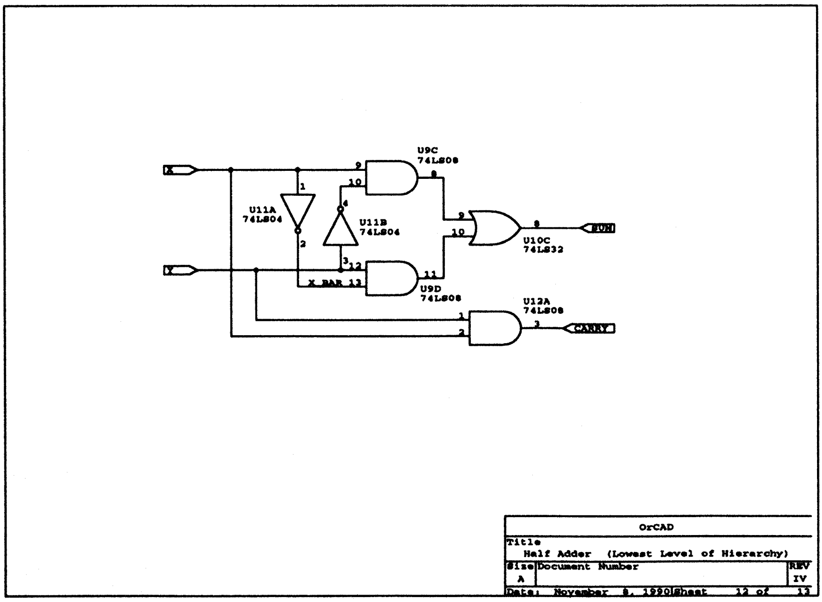


Figure 8-22. HALFADDF.SCH schematic.

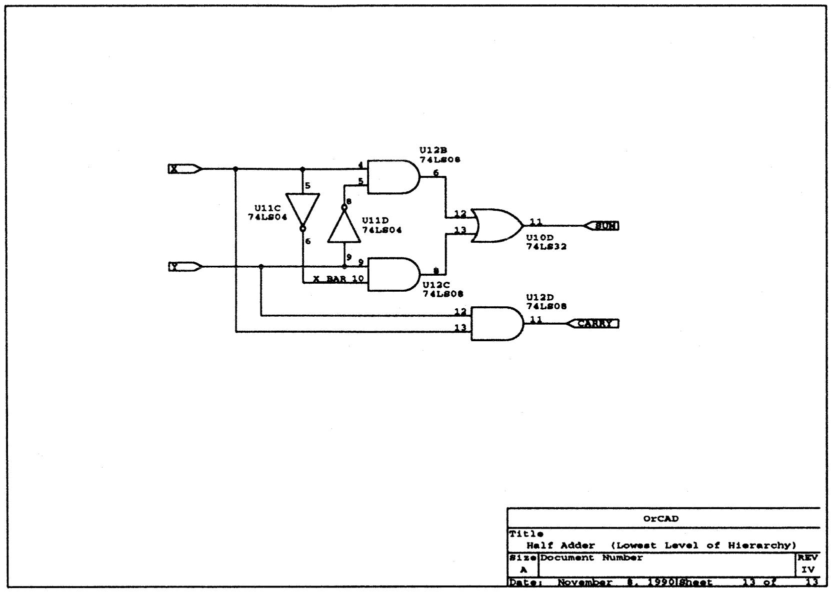


Figure 8-23. HALFADDG.SCH schematic.

Tips and techniques

This chapter is a collection of tips and techniques to help you better use **Schematic Design Tools.** It includes information about the title block, nonconnective objects, EMS, and other topics.

Unlike the other chapters in this guide, this chapter is not tutorial in nature. For additional information about any of the commands, menus, or options described in this chapter, see the chapter for the corresponding tool in the *Schematic Design Tools Reference Guide.*

Converting complex hierarchies

When you use **Complex to Simple** to convert a complex hierarchy that contains any sheetpath parts, be sure not to use the same filename for schematics in the design directory and schematics in the library directory. If you do, **Complex to Simple** uses the schematic in the design directory instead of the schematic in the library directory.

When **Complex to Simple** is copying and naming schematics used more than once in the design, it checks the design directory for duplicate filenames before naming new files. For example, if a complex design includes schematics named INPUT.SCH and INPUTA.SCH, each of which is referred to twice, the simplified design contains these four files:

- INPUT.SCH
- INPUTA.SCH
- INPUTAA.SCH
- INPUTAB.SCH

Title block tips

Schematic Design Tools provides a great deal of flexibility with the title block on a worksheet. You can use **Draft's** standard title block, an ANSI title block, or a custom title block you create. If you have paper that has your title block preprinted on it, you can set up **Draft** so that the title block is not plotted. In its place, text is plotted in locations to line up with your preprinted title block. This section describes the many different ways that title blocks can be manipulated.

OrCAD's title block

The **Schematic Design Tools** schematic editor, **Draft,** creates a title block that looks like the one shown in figure 9-1. You define all of the information on the title block except the date and size. **Draft** automatically enters the size and modification date of the worksheet.

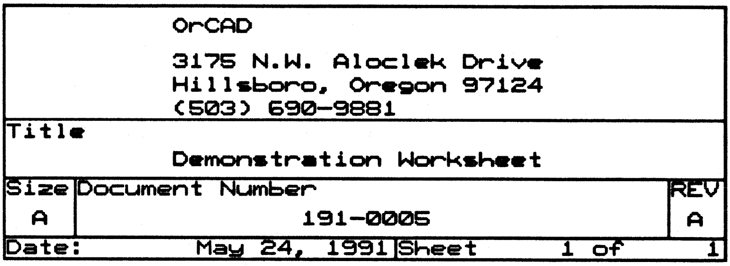


Figure 9-1. Sample OrCAD title block.

ANSI title block

You can configure **Schematic Design Tools** so that **Draft** creates an ANSI title block like the one shown in figure 9-2.

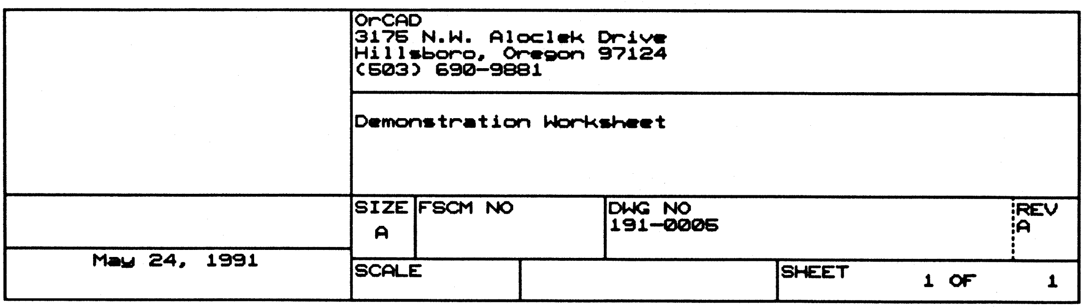


Figure 9-2. ANSI title block.

The ANSI title block conforms to the guidelines given in ANSI Standard Y14.1-1980. As you can see in figure 9-2, the ANSI title block is larger than the default OrCAD title block. On an A-size drawing, it takes up a large amount of the drawing area.

See *On the Configure Schematic Design Tools screen* in this section for instructions on how to create an ANSI title block.

△ ***NOTE:*** *If you use an ANSI title block, you may want your worksheet to have ANSI standard dimensions. These dimensions are given in tables 1-2 and 1-6 in the* Schematic Design Tools Reference Guide. *Since most, if not all, PC-compatible printers and plotters cannot print as close to the edge of the page as specified in the ANSI standard, OrCAD's default worksheet dimensions are reduced. These reduced dimensions are given in tables 1-4 and 1-5 in the* Schematic Design Tools Reference Guide. *If your printer or plotter can print closer to the edge of the paper, adjust the worksheet size in the* ***Template Table*** *area of the* ***Configure Schematic Design Tools*** *screen.*

Defining title block information

You can define title block information either in **Draft** or on the **Configure Schematic Design Tools** screen.

In Draft

To define title block information in **Draft**, place the pointer in the title block and select **EDIT**. The menu shown at right displays. Select the field to edit and answer the prompts that display.

```
Edit title block
Revision code
Title of sheet
Document number
Sheet number
Number of sheets
Organization name
1st Address Line
2nd Address Line
3rd Address Line
4th Address Line
```

For more information, see the **EDIT** command description in *Chapter 2: Draft*.

△ *NOTE: If you are using an ANSI title block (figure 9-2), you must use the **PLACE Text** command to place text in the **FSCM NO** and **SCALE** boxes. The **EDIT** command does not contain menu items to edit this information.*

On the Configure Schematic Design Tools screen

To define title block information, display the **Configure Schematic Tools** screen and pan to the **Worksheet Options** area (figure 9-3).

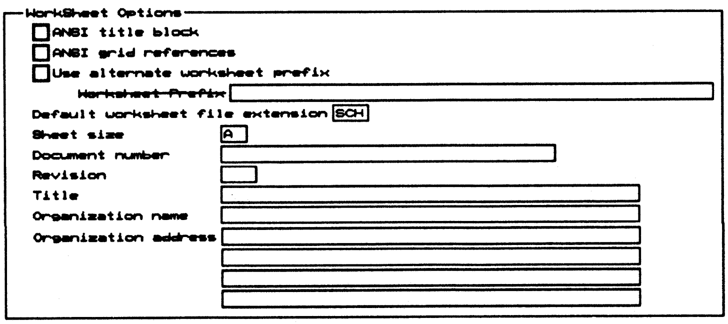


*Figure 9-3. **Worksheet Options** area of the **Configure Schematic Design Tools** screen.*

Information entered here automatically displays in the title block of each schematic created after the information is defined.

To use an ANSI title block (pictured in figure 9-2), select the **ANSI title block** option on this screen.

See *Worksheet Options* in chapter 1 for more information.

➤ *Helpful hint . . .*

Consider defining title block information such as organization name and address on the **Configure Schematic Design Tools** screen in your TEMPLATE directory. That way, each new design you create will be set up with your company name and address.

Suppressing title block elements

You can suppress the title block's lines, text, or both, as described here.

Lines

To suppress title block lines and leave title block text on the worksheet, display the **Configure Schematic Design Tools** screen. Pan down to the **Color and Pen Plotter Table** area.

Click in the **Pen** entry box to the right of **Title Block.** Enter `99` to tell **Plot Schematic** not to plot the title block.

When you open a worksheet in **Draft,** notice that the title block lines still display. However, they do not appear on the plot, as shown in figure 9-4.

```
                 OrCAD
                 3175 N.W. Aloclek Drive
                 Hillsboro, Oregon 97124
                 (503) 690-9881
Title
                 Demonstration Worksheet
Size Document Number                            REV
  A               191-0005                       A
Date:        May 24, 1991 Sheet      1 of        1
```

Figure 9-4. Plot of a title block with lines suppressed.

△ *NOTE: This also turns off the border around the drawing area during printing.*

Text

To suppress title block text and leave title block lines on the worksheet, display the **Configure Schematic Design Tools** screen. Pan down to the **Color and Pen Plotter Table** area.

Click in the **Pen** box to the right of **Title Text.** Enter **99** to tell **Plot Schematic** not to plot the title block's text.

When you open a worksheet in **Draft,** notice that the title block text still displays. However, it does not appear on the plot, as shown in figure 9-5.

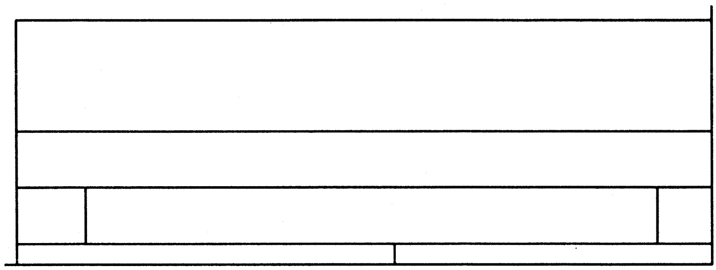


Figure 9-5. Plot of a title block with text suppressed.

Lines and text

There are two ways to suppress both title block lines and text:

- Use **Draft's SET Title Block No** command. If you use this method, the title block and its text do not display on the screen or appear on a print or a plot.
- Set both the title block and title text to a pen of 99 on the **Configure Schematic Design Tools** screen. If you use this method, the title block and its text display on the screen, but do not appear on a plot.

➤ *Helpful hint . . .*

If you are plotting on paper that has your title block preprinted on it, suppress the title block and its text as described previously. Use **Draft's PLACE Text** command to place text in the correct position so that when you plot your schematic the text prints in the correct place in your title block.

Creating a custom title block

You can create a custom title block using a library part made to look like a title block, or by using wires without labels.

Using a library part

To create a library part that looks like a title block, use **Edit Library** to make a library part that looks like a title block. Be sure that the part is nonconnective. This means it must be a zero-element part, and cannot have any pins. For more information, see *Nonconnective objects* in this chapter.

In **Draft**, suppress the title block using the **SET Title Block No** command, as described previously.

Place the library part on each new schematic.

Using wires

To use wires to create a title block, suppress the title block using **Draft's SET Title Block No** command, as described previously.

Draw the title block using the **PLACE Wire** command.

If text is required, be sure to place text, not labels. The netlist tools ignore the wires only if they have no labels and no pins attached.

➤ *Helpful hints . . .*

If you would like wider lines in your title block, draw buses rather than wires.

You may want to use a combination of both of the methods described above. For example, you can create a logo as a library part, then draw the title block with wires and place the logo in your title block.

Using your custom title block in each design

Once you create a custom title block, you can easily duplicate it on each new design using one of the methods described here.

Create a template schematic

Use one of the methods described in *Creating a custom title block* to create a worksheet that contains only the title block. Keep this worksheet in the TEMPLATE directory. It will be copied into each new design. If it has a name of TEMPLATE.SCH, it will always be given the name of the new design with an extension of .SCH.

Create a macro

Create a macro that draws the title block with wires. This macro can also include any text that must be part of the title block.

Place the macro in a macro file in your TEMPLATE directory. It will become a part of each new design. Run the macro in each new design.

Archiving parts

To achieve more efficient use of memory, run **Archive Parts in Schematic** on each design, turning both LIBARCH and COMPOSER on. This creates a library containing only the parts used in your design.

Doing this protects the design from changes in standard libraries and results in more efficient memory use because you have to configure only one library instead of several.

Nonconnective objects

Three **Schematic Design Tools** libraries contain objects that have no electrical connectivity. These objects are not included in the netlist and can therefore be used to customize your schematic worksheet as explained in this section.

About nonconnective objects

You may want a design to contain objects that have no electrical connectivity and are not processed by **Create Netlist** or **Create Hierarchical Netlist**. These objects can be:

- Mounting holes.
- Mechanical hardware such as screws and washers.
- A physical representation of the device you are designing.
- Floating or unconnected pins that can connect to an option, such as unused pins on a serial port.
- Flowchart symbols.

OrCAD/SDT III

In OrCAD/SDT version 3.22, you could place a part on a schematic for illustration purposes by deleting the part's value and reference fields. The part would not appear in the netlist. NETLIST viewed these unconnected devices as errors and reported:

```
<<<ERROR>>> X=  .80,Y=  1.00 Part has no REFERENCE
Part has no VALUE -  Part will be ignored.
```

However, NETLIST still produced a netlist.

Schematic Design Tools Release IV

Because of the enhanced error-processing capabilities of Release IV software, **Create Netlist** and **Create Hierarchical Netlist** interpret these unconnected parts as errors and terminate the netlist process without producing a netlist or connectivity database. It halts at the error and goes no further. Since this is an "error" and not a "warning," it is not effective to use INET's **Ignore Warnings** option.

Release IV solution

Release IV includes special libraries that contain nonconnective objects that can be placed on the schematic and still take advantage of the powerful error checking available in the Release IV netlist tools.

These nonconnective objects have no reference designators, values, or pins. They are found in the libraries listed in table 9-1.

Library	*Contents*
ASSEMBLY.LIB	Part outlines for assembly drawings to specify position of devices for board placement.
FLOWCHT.LIB	Programming flowchart symbols.
SHAPES.LIB	Generic library containing circles, squares, 90° arcs, and diamonds.

Table 9-1. Libraries that contain nonconnective parts.

See technical note #30: *Nonconnective objects in Schematic Design Tools* for illustrations of some of the parts found in these libraries.

Placing nonconnective objects on your schematic

Follow the steps below to place nonconnective objects on your worksheet:

1. Use **Edit Library** to create an object that looks like an actual device but has no electrical connectivity. You can use objects from any of the libraries listed in the table above. Use the objects as they are, or use them to create a new object.

   To be nonconnective, an object must be a zero-element part and cannot have any pins. Since IEEE parts are always single-element parts, they cannot be used. Only block and graphic parts can be nonconnective.

2. Place the part in a custom library. You can use it whenever you need it.

Converting OrCAD/SDT III schematics to Schematic Design Tools Release IV

You may have schematics developed prior to Release IV that contain parts with deleted reference designators and part values. If this is the case, your schematic will not pass the rigorous checking of the Release IV netlist tools.

To correct this, use **Edit Library** to change the objects in question so that they have no pins and are zero-element parts. When you run **Create Netlist** or **Create Hierarchical Netlist**, the new parts will not produce errors.

△ *NOTE: If you run **Create Netlist** or **Create Hierarchical Netlist**, and then change parts in one of the libraries, you must select the **Unconditionally process all sheets in design** option when you run **Create Netlist** or **Create Hierarchical Netlist** again. Since the schematic itself has not changed, INET will not detect any changed time stamps and will not find anything to process.*

Uppercase letters in key fields

Lowercase letters entered in key fields are handled as literals. To use the special characters "V" for **Value** and "R" for **Reference** and have them interpreted correctly, enter them in uppercase.

Duplicate sheet names

Sheet symbols in a design must have unique names. If your design has two sheet symbols with the same name, you receive this message when you run **Create Netlist**:

```
Duplicate Sheet Names
```

For example, two sheet symbols named `SHEET` cause this message to display, but two sheet symbols named `SHEET1` and `SHEET2` do not.

Changing netlist formats

Create Netlist and **Create Hierarchical Netlist** create netlists incrementally. When you run a netlist, only the items that have *changed* are included. A new file date or time denotes change. If you configure your netlist for one format, run a netlist, then configure it for a different format and run a netlist again, the netlist does not change.

To produce a netlist with a new format from an unchanged design, run IFORM with the **Force IFORM to always create a formatted netlist** option selected.

About EMS

This section describes how the design environment and **Schematic Design Tools** use expanded memory, commonly called EMS.

What is EMS?

EMS is an acronym for expanded memory specification. The full acronym is LIM-EMS, for Lotus-Intel-Microsoft Expanded Memory Specification.

EMS specifies how software works with special hardware to swap 16K *pages* of memory into and out of the one megabyte of main memory typically available in IBM PCs and compatibles.

Using memory management software, an application can read and write data on one page, swap in another page, and then swap back to the first page, with all data intact. Using this method, an application can use a small number of 16K main memory *slots* to access much more than one megabyte of memory. Figure 9-6 shows the main memory slots and the 16K pages that are swapped into and out of the slots.

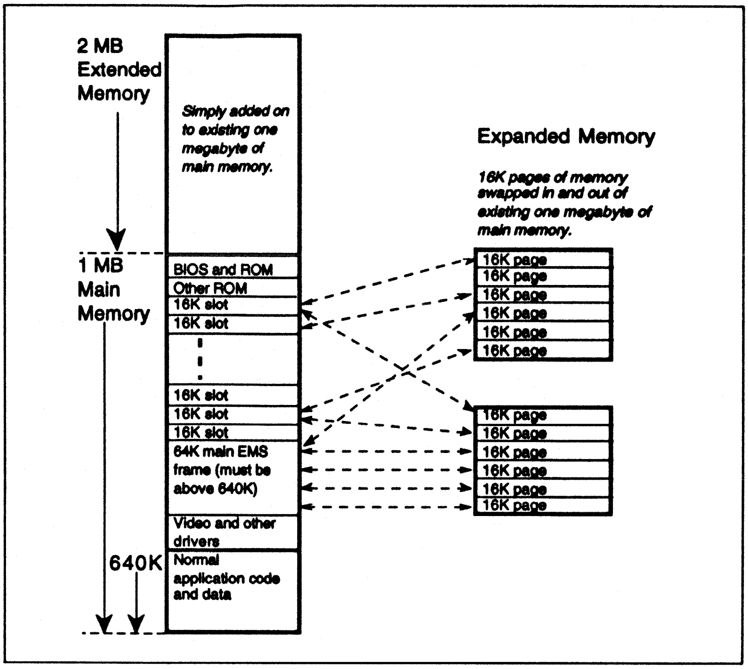


Figure 9-6. Extended memory and expanded memory (EMS).

△ *NOTE: Expanded memory is sometimes confused with extended memory. Extended memory is memory above the one megabyte of main memory, while expanded memory is memory that is swapped into and out of the one megabyte of main memory. Some software can make extended memory act like expanded memory. While Release IV software may work with this type of software, it does not use extended memory directly.*

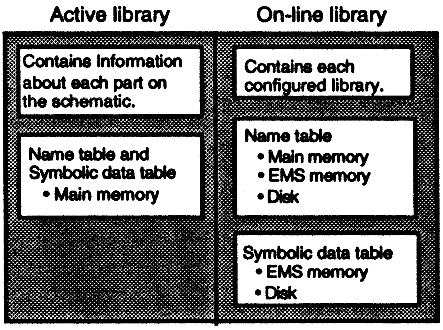
EMS in the ESP design environment

When the design environment is run, it checks to see if at least two 16K pages of EMS are available. If they are, the design environment loads its display driver into EMS. If the design environment and an OrCAD tool set use the same display driver, the driver needs to be loaded only once. This removes the display driver from the lower 640K of memory, providing more lower memory for the OrCAD tool set.

When you exit ESP and return to DOS, all EMS used by the design environment and OrCAD tool sets is released. The EMS is then available for other applications.

EMS in Schematic Design Tools

Schematic Design Tools uses two types of libraries: the active library and the on-line library. Both of these libraries contain a *name table* and a *symbolic data table.* The name table is a list of parts. The symbolic data table contains symbolic information about each part. The figure above right shows the active library, the on-line library, and what each contains.



Contents of the active library and on-line library.

Active library

The active library contains information about each part *on the schematic.* It always resides in main memory and can be configured to be 64–152K (this is done on the **Configure Schematic Design Tools** screen).

- The name table contains a list of the parts found *on the schematic.*
- The symbolic data table contains all of the symbol information for each part *on the schematic.*

On-line library

The on-line library contains information about *each configured library.*

- The name table contains a list of all the parts in *each configured library.* It can be stored in main memory, EMS memory, or on disk.
- The symbolic data table contains all of the symbol information for each part in *each configured library.* It can be stored in EMS memory or on disk.

Configuring Schematic Design Tools to use EMS

Follow these steps to configure **Schematic Design Tools** to use EMS.

1. Select **Draft** and **Configure Schematic Tools** from the **Schematic Design Tools** screen. The **Configure Schematic Design Tools** screen displays.
2. Move to the **Library Options** area.

   Notice the headings **Name Table Location** and **Symbolic Data Location** at the bottom of the **Library Options** area.
3. Select the desired location for each of these tables. The next section discusses the performance impacts of the different configurations.

Performance impact

Depending on the location of the on-line library's name and symbolic data tables, you can expect the performance impacts listed below. This list is given in order of efficiency, with the most efficient configuration given first.

- **Name table in main memory and symbolic data table in EMS.** Draft's **GET** and **LIBRARY Browse** commands execute fastest under this configuration.
- **Name table in EMS memory and symbolic data table in EMS.** The **GET** and **LIBRARY Browse** commands may be slightly slower than in the first configuration; however, you can add additional EMS memory to get as many parts on line as possible.

- **Name table in main memory and symbolic data table on disk.** The **GET** and **LIBRARY Browse** commands are even slower, but performance is still tolerable. This is the best option for PCs without EMS.
- **Name table in EMS memory and symbolic data table on disk.** This configuration should only be used for the following special cases:
  - Very large designs, such as E-size drawings with a large number of parts.
  - PCs with a small amount of EMS memory.
  - PCs with a small amount of available main memory. This can be caused by running multitasking software or a large network driver.

  Performance in this configuration is slower than in the preceding configurations, but is still acceptable.
- **Name table on disk and symbolic data table on disk.** This is the slowest configuration. It should only be used with portable computers that have 512K of main memory. It is tolerable for long use only if the hard disk is fast.

Viewing EMS memory allocation in Draft

The **CONDITIONS** command in **Draft** displays the amount of EMS and main memory used by the active library and the on-line library. To view this data, simply select **CONDITIONS** from **Draft's** main menu. When you are done viewing this information, press any key to return to **Draft's** main menu.

△ *NOTE: The* ***CONDITIONS*** *command displays information about the active library and the "reference" library. The reference library is the same as the on-line library.*

Reporting unused match strings

When printing a bill of materials report using **Create Bill of Materials** and an include file, you may want to find out which strings in your include file do not have a corresponding match string in the design.

To do this, make the following settings on the local configuration screen for **Create Bill of Materials:**

- Specify your include file by selecting the **Merge an include file with report** option and entering the include file name in the **Include** entry box.
- Select the **Report un-used match strings in include file** option.

Copying parts from one library to another

Follow these steps to copy parts from one library to another:

1. Use **Edit Library's EXPORT** command to copy a part from the original library to a temporary file.
2. Run **Edit Library** on the target library and **IMPORT** the temporary file.
3. Edit the part if necessary.
4. Select **LIBRARY Update Current.**
5. Select **QUIT Update File.**

Encapsulated PostScript

This section describes how to take an EPS file from an OrCAD application into two widely-used Microsoft programs, Word for Windows (WINWORD) version 1.1a and Word 4.0 for the Macintosh.

Creating EPS

Follow the instructions here to create an EPS file that can be incorporated into any application that accepts EPS.

Configure the tool set

Display the **Configure Schematic Design Tools** screen. Select one of the four EPS drivers. (To install the drivers on your system, run the INSTALL program from the Install disk.) Most illustrations in OrCAD manuals use EPS1.DRV.

Locally configure the Plot Schematic tool

Display **Plot Schematic**'s local configuration screen. Select the option **Send output to a file,** and the **Create a plot file** option below it. Some applications, such as WINWORD, require that EPS filenames end with .EPS, so enter **EPS** in the **File Extension** entry box. Then select **Automatically scale and set X,Y offsets for specified sheet size** and the **A** size option below.

Plot the schematic to disk

Once the tools are configured, you make an EPS file by selecting **Plot Schematic.** You can now bring the EPS file into other applications.

Placing EPS in WINWORD

In WINWORD version 1.1a, EPS support is built into NORMAL.DOT. Make sure your EPS filename ends with .EPS (required by WINWORD's EPS filter). Open NORMAL.DOT, select **Insert,** then **Picture,** and enter the name of the EPS file. In a moment, the placeholder for the EPS image appears.

That is all there is to it. The schematic does not display on your screen, but it will print all right on a PostScript printer. Earlier versions of WINWORD worked differently, so be sure to read the README.DOC file that accompanies version 1.1a.

Placing EPS into Word 4.0

On the Macintosh, as in Word for Windows, you can see the EPS graphic, manipulate it, and place it into many applications.

Transfer the plot file from the PC to the Macintosh

Use one of many possible techniques to transfer the plot file from the PC to the Macintosh. For example, use MacLink Plus™ V5.01 (from DataViz Inc.) with file translation set to "EPS to EPS."

Create a screen image of the file

Create a screen image of the EPS file. The desk accessory SmartArt™ V2.0 (from Adobe) does this very simply. The Macintosh must be connected to a PostScript printer. Open SmartArt, select "EPSF and Text," open the plot file, and select "Reimage."

The reimage option sends the EPS file to the PostScript printer for processing. The printer maps the EPS instructions to pixels and sends the image back. Save the returned image.

Select the **Copy** command to copy the image to the clipboard or scrapbook. Close SmartArt.

Open a Word document

Paste the image in place in a Word document. If necessary, size or crop the image to fit the space available.

Error objects

If you run **Check Electrical Rules** and it flags errors on your schematic, remove them using **Draft's QUIT Update File** command.

Using sheets and parts to point to another worksheet

To create a hierarchy, you place an object on your worksheet that points to another worksheet file. This object can be a sheet symbol, a sheet part, or a sheetpath part. Their similarities and differences are discussed in detail in this section.

About sheets and parts

Before discussing sheet symbols, sheet parts, and sheetpath parts, it is important that you understand the difference between a *sheet* and a *part*. In trying to understand these differences, think of the function of the object rather than the object's physical appearance.

Parts are graphic objects that you place on the worksheet to represent the electronic devices in your design. Among other characteristics, they have part fields, are annotated, and may have power pins. If you print a bill of materials report, all parts on your worksheet are listed.

Sheets are objects that you place on a worksheet that point to another worksheet file. Sheets do not have part fields, are not annotated, and cannot have power pins. Sheets are not listed in a bill of materials report.

Table 9-2 compares parts and sheets.

Part	*Sheet*
Has part fields	Does not have part fields
Is annotated	Is not annotated
Can have power pins	Cannot have power pins
Appears in BOM report	Does not appear in BOM report
Appears in incremental connectivity database as a part	Appears in incremental connectivity database as an instance of a child

Table 9-2. Characteristics of parts and sheets.

Sheet symbol

A sheet symbol is an ANSI-standard rectangle that you place on a worksheet using **Draft's PLACE Sheet** command. **Draft** automatically assigns it a random filename, such as 91G8F06#.SCH. You can tell it to point to the file of your choice by selecting **PLACE Sheet Filename.**

A sheet symbol functions as a sheet, and has all the sheet characteristics described in table 9-2.

The worksheet file that a sheet points to should be stored in the current design directory unless a path is specified with the **PLACE Sheet Filename** command.

Sheet part

A sheet part is a library part that you change to a sheet. You place a library part on a worksheet using **Draft's GET** command. To change it to a sheet part, you use **Draft's EDIT SheetPart Name** command to tell it to point to a worksheet file.

Once you give a library part a sheet part name, it no longer functions as a part. It functions as a sheet, and has all the sheet characteristics described in table 9-2.

The worksheet file that a sheet part points to should be stored in the current design directory unless a path is specified with the **SheetPart Name**.

Sheetpath part

A sheetpath part is similar to a library part in that it is stored in a library. It may function as a sheet *or* as a library part. When you get a sheetpath part from a library, it *already* points to a worksheet file.

If you select the **Descend into sheetpath parts** option on a tool's local configuration screen, a sheetpath part functions as a sheet and has all the sheet characteristics described in table 9-2. If you don't select this option, the sheetpath part functions as a part and has all of the part characteristics described in table 9-2.

OrCAD libraries don't contain any sheetpath parts. You must create them and add them to a library. The worksheet file that a sheetpath part points to should be stored in the directory specified in the **Library Prefix** entry box in the **Library Options** area of the **Configure Schematic Design Tools** screen.

Conclusion

As described previously in this section, be sure to think of the function of an object rather than its physical appearance. Table 9-3 summarizes the different type of objects, and tells whether they function as a part or a sheet.

Type of object	*Functions as a sheet*	*Functions as a part*
Library part	No	Yes
Sheet symbol	Yes	No
Sheet part	Yes	No
Sheetpath part	If **Descend into sheetpath parts** option is selected	If **Descend into sheetpath parts** option is *not* selected

Table 9-3. Functionality of objects.

Moving designs

When you want to give a design to another user or send it to OrCAD technical support, you must include the following files:

- ❖ All associated schematic files
- ❖ Any custom library files, which may be located in your library directory
- ❖ Any custom netlist format files, which may be located in your netlist directory
- ❖ The **Schematic Design Tools** configuration file for the design (SDT.CFG)

Because **Backup Design** copies only those files and libraries that reside in the specified design directory, be sure to include any files that are located in other directories.

Installing new drivers

To add new display, printer, or plotter drivers once you have installed **Schematic Design Tool**, run the INSTALL program from your installation disk. You don't have to reload all the software—just the drivers that are found on the installation or upgrade disk.

1. Insert the disk labeled "Install" into your computer's floppy 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>.
4. Follow the instructions on the screen, selecting the drivers you want to install. After you select drivers, INSTALL displays a menu of OrCAD tool sets. Don't select anything; just press <Enter>. This ends the installation.

Removing error objects

Check Electrical Rules puts temporary error markers, or *error objects*, on your worksheet to show you the locations of the errors. Select **QUIT Update** from **Draft**, or run **Cleanup Schematic** with the **Remove error objects from schematic sheet(s)** option selected to remove the error objects.

Converted part forms

A part's converted form *must* have the same pin numbers as the normal version of the part. Therefore, all pins must be declared in the normal version so they can be used in the converted form.

For example, if the normal version of a part has pins numbered 1 and 3, and the converted form has pins numbered 0 and 3, you will have errors when you create a netlist. To eliminate this problem, just be sure that the pin numbers used in the converted form of the part also appear in the normal version of the part. The normal part in the example should include a pin numbered 0.

Scaled printing

Plot Schematic can automatically scale your prints to the desired size. You do not have to do any calculations—**Plot Schematic** does it for you. **Plot Schematic** also automatically rotates the image for the best fit on the paper. Follow these steps:

1. Display **Plot Schematic**'s local configuration screen.
2. Enter the filename of the schematic you want printed in the **Source** entry box.
3. Select the **Send output to printer** option.
4. Select the **Automatically scale and set X, Y offsets for specified sheet size** option.
5. Select the size of paper to print your schematic on: **A, B, C, D,** or **E.** Or, if your software is set up to measure in **Millimeters** in the **Template Table** on the **Configure Schematic Design Tools** screen, select **A4, A3, A2, A1,** or **A0.**
6. Select **OK** to save the configuration and display the **Schematic Design Tools** screen.
7. Run **Plot Schematic.**

Creating global macros

If you have macros that you use with all your designs, try putting them all in a global directory. Follow these steps:

1. Create a directory called \ORCADESP\SDT\MACROS.
2. Copy the macro files you want to be global macro files to the \ORCADESP\SDT\MACROS directory and delete the files from the design you copied them from.
3. Configure the **Draft Macro File** and **Edit Library Macro File** in the **Macro Options** area of the **Configure Schematic Design Tools** screen with the correct path and filename of the macro file in the global macro directory.

A

analog ■ Circuitry in which both voltage and frequency output vary continuously as a function of the input.

annotation ■ Assigning reference designators to parts in a schematic.

area ■ A section of a screen containing related buttons or configuration options. Most areas are bordered and named. Examples include the **Editors** area on tool set screens and the **File Options** area on local configuration screens.

ASCII ■ An acronym for *American Standard Code for Information Interchange*; a seven-bit code used to represent letters of the alphabet, the ten decimal digits, and other simple printable characters, as well as certain basic printer instructions such as Backspace, Carriage Return, and Line Feed.

B

bulletin board system ■ A computer dedicated to maintaining messages and software and making them available over telephone lines. People *upload* (contribute) and *download* (gather) messages by calling the bulletin board from their own computers. Abbreviated BBS.

button ■ A pushbutton-like image that you click to start an *action*. The *action* runs a single tool or a series of processes.

byte ■ A piece of computer data composed of eight contiguous bits that are stored and typically interpreted as a single unit.

C

CAE ■ An acronym for *computer aided engineering*.

check box ■ A small square: ❑. Check boxes are used in lists of options when more than one option can be selected at a time. Compare *radio button*.

complex hierarchy ■ A design in which two or more sheet symbols refer to a single worksheet. Compare *simple hierarchy*. See also *hierarchical design*.

configuration ■ The information a button or tool set uses to operate. Configurations can be tailored to your needs. The configuration for a tool set applies to all tools in the set. See also *local configuration*.

connectivity database ■ The incremental connectivity database (created by INET) and the linked connectivity database (created by ILINK). The connectivity database describes the connectivity of a design, and is used to transfer a design to **Digital Simulation Tools or PC Board Layout Tools**. See *incremental connectivity database* and *linked connectivity database*.

cursor ■ A marker inside an entry box showing where characters typed on the keyboard will display. The cursor for insert mode is a heavy underline, and the cursor for overtype mode is a square. Compare *pointer*.

D

default ■ A preselected parameter.

design ■ A set of plans for electronic circuitry.

design cycle ■ The process of conceiving, developing, testing, and producing a circuit.

digital ■ Circuitry in which data in the form of digits are produced by binary (on-off or positive-negative) electronic signals.

download ■ The process of retrieving a file from another computer.

E

EDA ■ An acronym for *electronic design automation*.

editor ■ A tool used to create or modify a design file.

entry box ■ A box in which text or numbers can be entered using the keyboard: .

F

flat design ■ A schematic structure in which output lines of one sheet connect laterally to input lines of another sheet through graphical objects called *module ports*. Flat designs are practical for small designs of three or fewer sheets. See *module port, schematic,* and *hierarchical design*.

H

hierarchical design ■ A schematic structure in which sheets are interconnected vertically and laterally in a tree-like pattern. At least one sheet, the root sheet, contains symbols representing other sheets, called subsheets. See also *complex hierarchy, simple hierarchy, root sheet,* and *flat design*.

I

incremental connectivity database ■ Two or more files (.INX and .INF) produced by INET. The .INX file lists every sheet referred to in the design; a .INF file for each sheet describes connectivity for the sheet. ILINK uses the *incremental connectivity database* to create an incremental netlist. See also *connectivity database* and *incremental netlisting*.

incremental netlisting ■ A method of creating a netlist in which only changed worksheets are processed each time **Create Netlist** or **Create Hierarchical Netlist** is run. See also *netlist*.

initial macro ▪ A macro that runs automatically whenever you run **Draft** or **Edit Library**. For the initial macro to work, you must configure **Schematic Design Tools** to load a macro file containing the desired macro definition.

intermediate netlist structure ▪ Three files (.INS, .RES, and .PIP) produced by ILINK for a design. The .INS (instance) file, the .RES (resolved) file, and the .PIP file (pipe link commands) are used by IFORM to create a netlist in one of over 30 formats.

K

K ▪ An abbreviation for *kilobyte*. 1K is equal to 2^{10} (1024) bytes. The "K" is taken from the metric system, where it stands for "kilo," or 1000.

key field ▪ A list of the part fields to combine and compare. Key fields are defined on the **Configure Schematic Design Tools** screen.

L

library ▪ A collection of standard, often-used part symbols stored as templates to speed up the design process.

librarian ▪ A tool used to manage and create library parts.

linked connectivity database ▪ An optional file (.LNF) produced by ILINK. This ASCII file is used to transfer connectivity information to **PC Board Layout Tools.** See also *connectivity database.*

list box ▪ A box on local configuration screens and in windows that lists files in specific designs or directories. You move through the list using scroll buttons next to the list box. On local configuration screens, you can specify a wildcard so the list box contains files that match the criteria you specify.

local configuration ▪ Configuration settings for a particular button. If the button runs several processes, each process can be configured locally. One tool can have different configurations in different buttons. For example, **Annotate** is configured differently under the **To Layout** button and under the **To Digital Simulation** button.

M

MB ▪ An abbreviation for *megabyte.* See *megabyte.*

macro ▪ A series of commands you can execute automatically at the touch of a single key. Macros dramatically reduce the number of keystrokes required to perform complex or repetitive actions.

megabyte ▪ One megabyte is equal to 2^{20} (1,046,576) bytes. The prefix "mega" is taken from the metric system, where it stands for "one million." Abbreviated *MB.*

module port ▪ A graphical object that conducts a signal between schematic worksheets. See also *flat structure.*

N

net ■ A graphical object that conducts a signal into or out of a sheet symbol, much as a module port conducts signals between schematic worksheets.

netlist ■ An ASCII file that lists the interconnections of a schematic diagram by the names of the signals, modules, and pins connected on a PCB. The nodes in a circuit. See also *incremental netlisting*.

P

pan ■ To change the portion of the worksheet being viewed by dragging the pointer from one location on the worksheet to another location. As you drag the pointer, the worksheet *pans* across the screen.

part ■ A schematic symbol that represents an object. The object can be either a part or another worksheet.

part field ■ A "slot" for text or data to be associated with a part. Each part has two part fields reserved for part value and part reference. It has eight other part fields that can be used to store other useful information. See also *key field*.

PCB ■ An acronym for *printed circuit board*.

PLD ■ An acronym for *programmable logic device*. See *programmable logic device*.

pointer ■ An arrow on the screen that moves as you move the mouse: ↖ . Compare *cursor*.

processor ■ A tool that subjects a design file to a specific process.

programmable logic device ■ A type of integrated circuit that contains fuses that can be blown, eliminating certain logical operations in the device and leaving others intact, giving the device one of many possible logical architectures or logical configurations.

prompt ■ A query from a program asking you to enter specific information.

R

radio button ■ A small circle: ○. Radio buttons are used in lists of mutually exclusive options: only one button can be active at a time. Compare *check box*.

reporter ■ A tool that creates a report, but does not modify design data.

root directory ■ The main directory on your computer; the directory that the computer boots from.

root sheet ■ The worksheet from which references to other worksheets are made in a flat or hierarchical design. Each design has only one root worksheet. See also *flat design* and *hierarchical design*.

S

schematic ■ A graphical representation of a circuit using a standard set of electronics symbols. See *flat design*, *hierarchical design*, and *root sheet*.

scroll buttons ■ Buttons used to move a directory in its window so that a different part is visible. The four scroll buttons are:

- Page Up
- Line Up
- Page Down
- Line Down

sheet symbol ■ A block-shaped symbol representing another worksheet. Signals are conducted into and out of sheet symbols by nets. See *net*.

simple hierarchy ■ A one-to-one correspondence between sheet symbols and the schematic diagrams to which they refer. Each sheet symbol represents a unique subsheet. Compare *complex hierarchy*. See also *hierarchical design*.

syntax ■ The formal structure of a language. Syntax includes the rules for making statements in the language, but excludes the meanings of the statements.

T

tag ■ A marked or saved location on a schematic or layout. You can use the **JUMP** command to go to a tag.

template ■ A set of patterns used to create new designs. The template is *not* a design itself.

text export ■ The process of copying text from a schematic worksheet to a text file.

text import ■ The process of copying text from a text file to a schematic worksheet.

TTL ■ An acronym for *transistor-transistor logic*.

tool ■ A computer program that performs some useful task. OrCAD tools are grouped into five categories: editors, processors, reporters, librarians, transfers.

tool set ■ A collection of tools designed to perform a set of electronic design automation tasks. OrCAD tool sets include: **Schematic Design Tools, Programmable Logic Design Tools, Digital Simulation Tools,** and **PC Board Layout Tools.**

transfer ■ A tool that transfers design information from one tool set to another tool set. Also runs whatever processes are necessary to go from one tool set to another.

U

upload ■ The process of sending a file to another computer.

user button ■ A button that you can program to perform whatever combination of functions you find useful (such as executables or batch files). User button programs are saved with the design files, so you can create design-specific buttons and not worry about overwriting user button programs for other designs.

W

wildcard ■ A series of characters you specify in a **Wildcard** entry box to filter the files that display in a list box. For example, *.* allows all files to be displayed in a list of files.

worksheet ■ The sheets of drafting paper on which schematics are drawn. Worksheets appear on the computer screen as a rectangular area in which you can place parts and draw wires.

Z

zoom ■ To change the view on the screen by making the objects appear larger or smaller.

Q

R

S

W

X

Z