Telesis

USER'S MANUAL

VOLUME I OPERATOR'S MANUAL

TELESIS SYSTEMS CORPORATION, 21 ALPHA RD., CHELMSFORD, MASS. 01824 (617) 256-2300

THIS PACKAGE CONTAINS THE TELESIS USER'S MANUAL
SCHEMATIC/PRINTED CIRCUIT

- VOLUME 1: OPERATOR'S MANUAL

- PART NUMBER: 76-00004-001 REV I

2/87

CHANGE CONTROL SHEET FOR 8204 MANUALS: VOLUMES 1 & 2

1. UPDATED FOR 8301A SOFTWARE RELEASE
2. UPDATED FOR 8302A SOFTWARE RELEASE
3. UPDATED FOR 8304A SOFTWARE RELEASE
4. UPDATED FOR 8304D SOFTWARE RELEASE
5. UPDATED FOR 8304E SOFTWARE RELEASE
6. UPDATED FOR EDA-3000 REV 1 SOFTWARE RELEASE
7. UPDATED FOR EDA-3000 REV 2 SOFTWARE RELEASE
8. UPDATED FOR EDA-3000 REV 3 SOFTWARE RELEASE
9. UPDATED FOR EDA-3000 REV 4 SOFTWARE RELEASE

PREFACE

PURPOSE

The User's Manual teaches the beginning user to operate the Telesis system, and serves as a reference for experienced operators.

AUDIENCE

The user should be familiar with printed circuit board design. No previous experience with CAD/CAM (computer-aided design/computer-aided manufacturing) or computer operation is required.

CONTENTS

The User's Manual is in two volumes:

VOLUME 1 - OPERATOR'S MANUAL

VOLUME 2 - REFERENCE MANUAL

CONVENTIONS USED IN THIS MANUAL

Your input is shown in two ways. Menu box input is enclosed in a rectangle corresponding to the menu box:

|ADD SYMBOL [NAME]| |N|A|N|D|2| |ENTER|

Tablet input is shown with this symbol (the P stands for PICK):

P

TELESIS CUSTOMER SUPPORT CENTER

PROBLEM REPORTING PROCEDURE

(617) 256 - 2388

A problem that occurs during the normal operation of your system may be considered a BUG. To determine if you have uncovered a BUG, we recommend the following :

1. Check the documentation carefully and make certain that you are using the command correctly.

2. Determine if the problem is reproducible.

3. Determine if the command worked previously; store your current drawing away and reboot the system, then try the command again.

WARNING

Do not reboot the system if you experienced a system crash. Contact the Customer Support Center immediately or your work may be lost.

IF THE ABOVE STEPS FAIL TO SOLVE THE PROBLEM, CALL THE CUSTOMER SUPPORT CENTER AT (617) 256 - 2388.

A support center operator will take your call and will require the following information :

a. Company name
b. System number (located on the back of the workstation)
c. Modem phone number
d. Software release number
e. Application (PC or mechanical)
f. Menu set (Draw Symbol, Draw Schematic, Design Board, or Other)
g. Command that caused problem
h. Command sequence
i. Error message (if any)
j. Complete description of problem in detail.

You may be asked additional questions regarding the problem, and the System Specialist may utilize Telesis Remote Diagnostics to further investigate. If the problem cannot be diagnosed, you may be requested to send the drawing and / or project to the Customer Support Center for further investigation. Upon verification of the problem, a copy of the Software Performance Report (SPR) with the appropriate Customer Support Event Number will be sent to you. The event number should be referenced if you have additional problems or questions about that specific problem.

The Customer Support Center is open between 8:30 a.m. and 5:00 p.m. in your time zone, Monday through Friday, excluding Telesis holidays. While coverage is provided only during prime business hours, the phones will be answered 24 hours a day, 7 days a week, 365 days a year, including all holidays. If you have a problem after normal hours or on the weekend, call the center and leave a message. A Customer Support Representative will contact you at the beginning of the next business day.

VOLUME 1: OPERATOR'S MANUAL

TABLE OF CONTENTS

VOLUME 1: OPERATOR'S MANUAL

TABLE OF CONTENTS

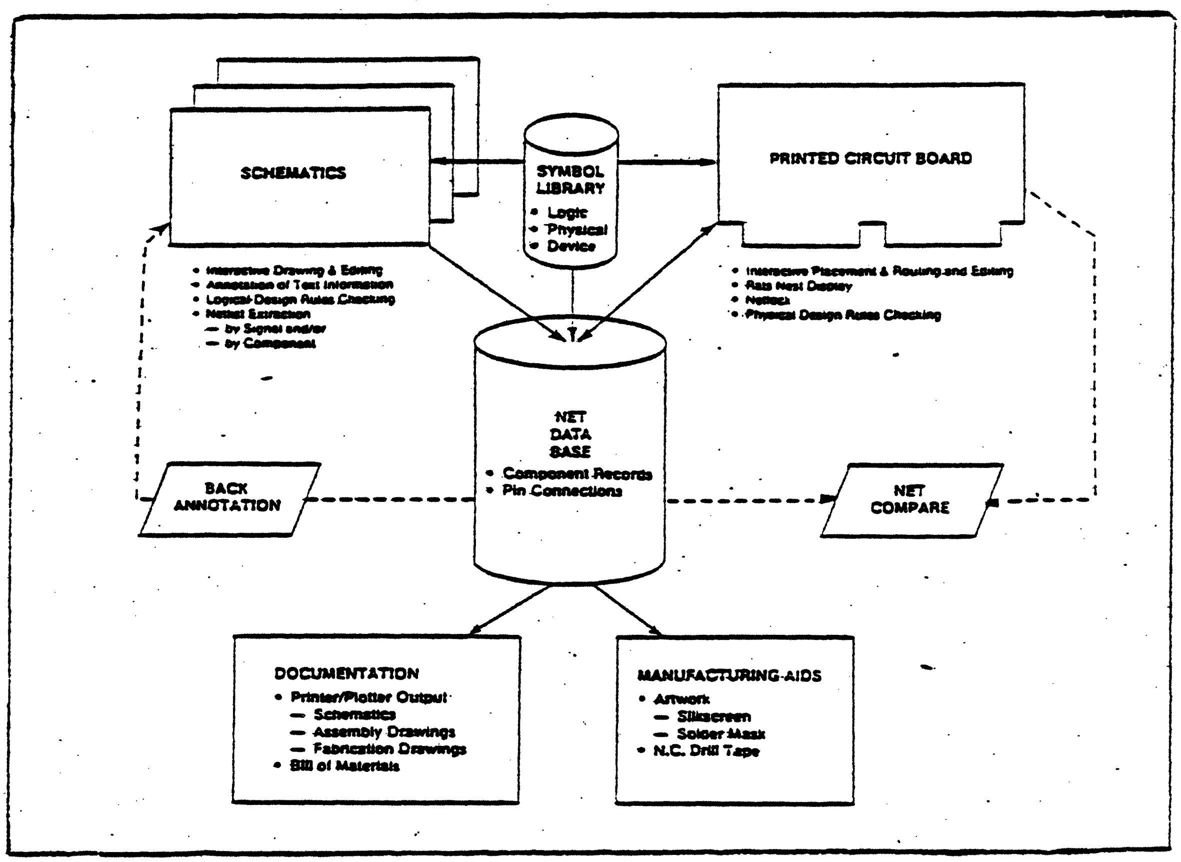
1. OVERVIEW OF THE TELESIS EDA-300 AND EDA-700 DESIGN SYSTEMS

The Telesis EDA-300 and EDA-700 are computer-aided design/computer-aided manufacturing (CAD/CAM) systems that allow you to design printed circuit boards without the time-consuming taping, checking, and calculating involved in manual design. Two software packages run on both these systems: the EDA-3000 printed circuit board (PCB) design package and the EDA-3100 schematic design package.

This introduction will first give a general overview of the printed circuit board design process using the Telesis system, and then it will describe the differences between the EDA-300 and the EDA-700.

2. OVERVIEW OF THE PRINTED CIRCUIT BOARD DESIGN PROCESS USING THE TELESIS SYSTEM

The diagram below gives an overview of the design process on the Telesis system:



You work interactively with the system in completing this process. The system collects and organizes information about the board design into a file called NET-DATA-BASE. This file is then used to automate the design process.

2.1 THE FIRST DESIGN - BASIC STEPS IN THE DESIGN PROCESS

Use the following steps to design the first printed circuit board on the system:

1. Create a library of logic symbols. (Optional)
2. Create a library of physical symbols. (Optional)
3. Create a library of device description files. (Optional)
4. Draw the schematic or input a text netlist file.
5. Create a net database file from the schematic or text netlist.
6. Run verification tasks (physical design rules checking and netcompare).
7. Place, improve, and route the board.
8. Generate fabrication documentation.
9. Generate an artwork tape.

During steps 1-8: Use file management to create archives during the design process.

(Steps 1-3 are optional because Telesis provides a library of symbols and device files. However, these symbols and device files may not meet all your needs. Therefore, you can create your own as well as using the ones that Telesis provides.)

2.2 LATER DESIGNS

After you have stored all logic symbols, component symbols, and device files that you created in a project file or SYSTEM-LIBRARY, execute the following steps to create a board design:

1. Draw the schematic or input a text netlist file.
2. Create a net database file from the schematic or text netlist.

3. Place, improve, and route the board.

4. Run verification tasks (physical design rules checking and netcompare).

5. Generate fabrication documentation.

6. Generate an artwork tape.

During steps 1-5: Use file management to create archives during the design process.

2.3 IMPORTANCE OF THE NET DATABASE IN THE DESIGN PROCESS

The NET-DATA-BASE is a system file containing electrical information about the design of a particular board. The system collects all net, pin, reference designator, logic function, and device type information to be used in the board design. Then it assembles it into the file called NET-DATA-BASE.

The NET-DATA-BASE is the foundation for the system's automated capabilities, which are as follows:

- Interactive editing with netlock
- Creating the NETLIST-REPORT and the COMPONENT-REPORT
- Creating the BILL-OF-MATERIALS report
- Logical Design Rules Checking
- Automatic and interactive placement
- Ratsnest
- Component, logic function, and pin swapping
- Automatic routing
- Net compare
- Back annotation
- Artwork generation
- Physical design rules checking.

With the Telesis system, there are two ways to create a NET-DATA-BASE:

1. Add, interconnect, and annotate symbols on the schematic. When the schematic is graphically complete, use the EXTRACT NETLIST command to create the file NET-DATA-BASE.

or

2. Input a text netlist. Use this method when you have the information for a board design and are not required to produce a schematic. When you complete the netlist, use the LOAD TEXT NETLIST command to create the file NET-DATA-BASE.

After the NET-DATA-BASE is created, you can continue your printed printed circuit board design by placing and interconnecting the board using the electrical and component information from the NET-DATA-BASE.

2.4 EASE OF USE OF THE TELESIS SYSTEM

The Telesis system is engineered for easy use and for comfort. The Telesis function screen is a high-resolution monitor which provides a dynamic interface to the system. It completely eliminates the need for the complex array of pre-printed menus, tablets, joysticks, and keyboard common to other systems.

All of the commands that you need to complete a printed circuit board design are presented in plain language on menu pages displayed on the function screen. There are also special keyboard, keypad, and tablet menus for explicit data input.

The Telesis graphics screen is a high-resolution color monitor that offers excellent glare protection and outstanding image clarity for schematic generation and board placement and interconnection.

3. THE TELESIS EDA-300 PRINTED CIRCUIT DESIGN SYSTEM

Telesis supports two computer-aided design/computer-aided manufacturing (CAD/CAM) systems: the EDA-300 and the EDA-700. The EDA-3000 printed circuit board (PCB) design software package and the EDA-3100 schematic design software package both run on the EDA-300. Figure INTRODUCTION-I on the next page shows the EDA-300.

3.1 MAJOR ELEMENTS OF THE EDA-300 SYSTEM

- GRAPHICS SCREEN - the 19-inch high-resolution color monitor that displays drawings and text files.
- FUNCTION SCREEN - displays menus and messages.
- LIGHT PEN - transmits menu selections to the system.
- DISK UNIT - stores the EDA-3000 software and records, and stores your input for the design database using a 40 megabyte Winchester disk.
- FLOPPY DISK DRIVE - reads files stored on a floppy diskette for transfer to the disk unit and allows archiving of files stored on the system.

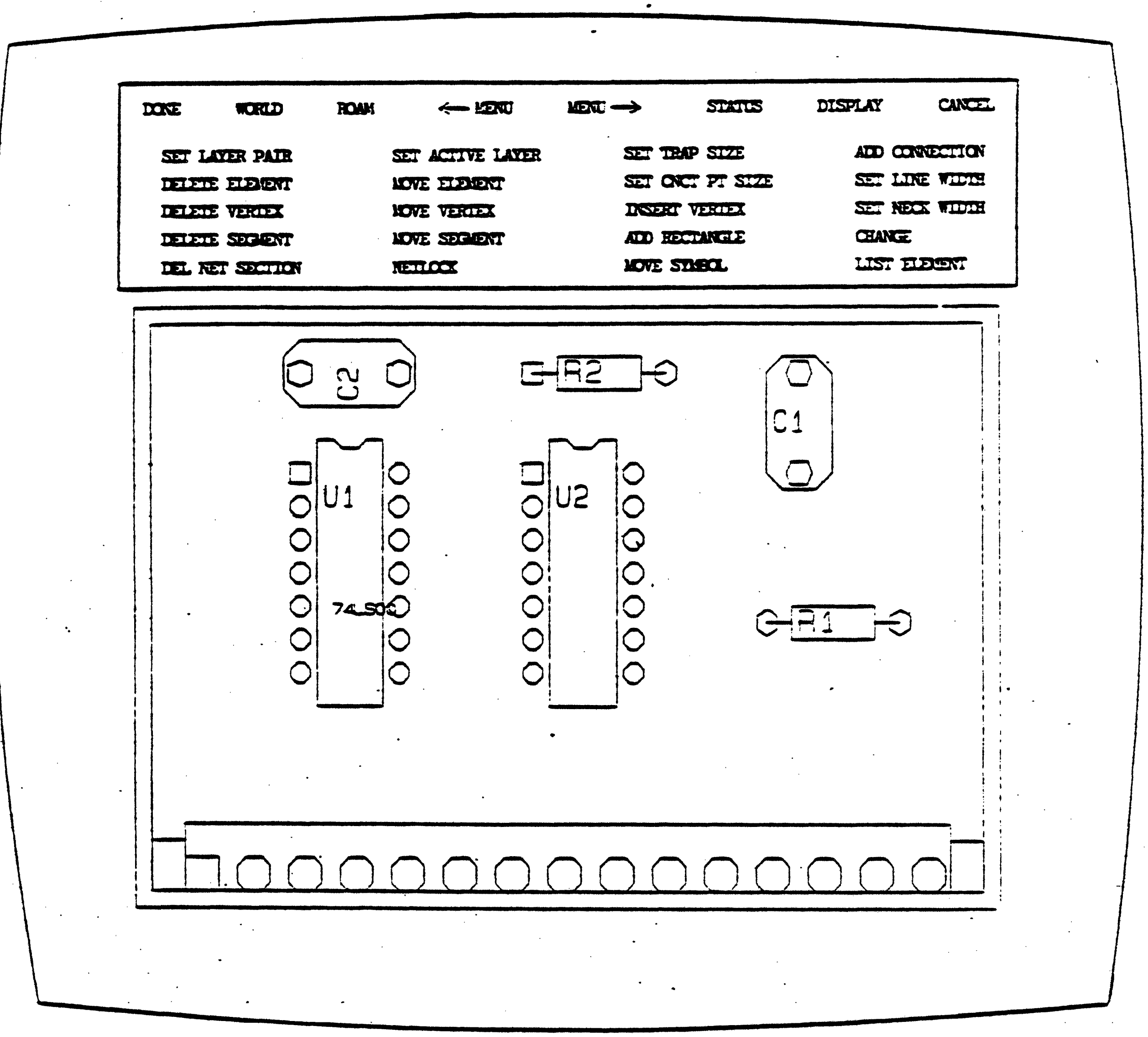


Figure INTRODUCTION-2: Screen of the EDA-700

In tandem with the pop-up menu, you use the alphanumeric keyboard and optional function keys. The EDA-3000 software package that runs on the EDA-700 is the same software that runs on the EDA-300. That is, the same PCB design commands that run on the EDA-300 also run on the EDA-700. Because the hardware of these two systems is different, you execute commands in a slightly different way on each. Execution of commands on the EDA-700 will be discussed in greater detail in Section 4.6 below.

4.2 Major Elements of the Telesis EDA-700 Design System

o GRAPHICS SCREEN	-	the 19-inch high-resolution color monitor that displays drawings, menus, and text files.
o PROCESSING UNIT	-	stores the major internal hardware modules and system components listed below:
* Central Processing Unit	-	controls and processes information stored on the system.
* Disk Unit	-	stores the design application software package(s), and records and stores operator input for the design data base using a 65 Mb Winchester disk. An optional 140 Mb Winchester is available.
* Floppy Disk Drive	-	located on the front of the Processing Unit, the floppy disk drive reads files stored on a floppy diskette for transfer to the disk unit, and allows archiving of files stored on the system.
* Graphics Processor	-	controls the display and viewing of drawings on the graphics screen.
o MOUSE	-	the cursor positioning device that transmits menu and tablet selections to the system's processing unit.
o KEYBOARD	-	allows you to define and input textual information to the system.

4.3 Options for the EDA-700

Following are the options for the printed circuit board design package running on the EDA-700:

- o Extended roam space
- o Magnetic tape drive
- o Pen plotter
- o Remote processing unit
- o Local area networking
- o Filled penplot software capacity
- o Numerical control tape punch
- o Matrix printer/plotter
- o Digitizer
- o Text editing terminal
- o Additional disk and memory

4.4 Hardware Differences between the EDA-300 and EDA-700 Workstations

The hardware differences between each workstation are listed below:

Telesis EDA-300	Telesis EDA-700
Dual screens (one screen for graphics and another screen for function menus)	Single screen (graphics and popup softkey function menus on one screen)
Keyboard on function screen	Alphanumeric keyboard and function keys
Light pen	Three-button mouse
Black background only	Options of black, brown, or light blue background color

The EDA-700 Design System has a single screen. You communicate to the system using an alphanumeric keyboard, function keys, and a three-button mouse. (A detailed description of the keyboard and the mouse functionality is provided in section 4.6 below.)

When the system boots up initially, the graphics background is black, the menu background is slate blue, and the menu lettering is white. You may select two other color combinations, depending on your preference. Each time you simultaneously press the <ALT> and <A> keys, the colors change to the next of the three color settings listed below. If you press <ALT> <A> when in the light blue background, the system returns to the black background.

Background color	Menu lettering	Menu background
Black	White	Slate blue
Brown	White	Slate blue
Light blue	Black	White

On the EDA-700, you have the choice of using the function keys (numbered F1 - F10) on the left-hand side of the keyboard.

4.5 Software Differences between the EDA-300 and EDA-700 Workstations

The same applications software package that operates on the Telesis EDA-300 workstation also operates on the EDA-700. Menus and commands are identical and operate in the same way. However, commands that appear on two lines on the EDA-300 are limited to single lines on the EDA-700.

The use of the EDA-700 editor is somewhat different from that of the EDA-300 editor. These changes are summarized in the following table:

EDA 300 Command	EDA 700 Key	Description
GO TO TEXT	F1	Move to text area
GO TO HEAD	F2	Move to header area
NEW FILE	F3	Create new file
EDIT FILE	F4	Edit old file
CLS FILE	F5	Close file
CANC CMD	F6	Cancel command
EXCT	F7	Execute command
REPLACE	F8	Replace a character
REP (lock)	F9	Replace a series of characters
EXIT	F10	Exit and close file
CANCEL	ESC	Exit without closing
DEL CHAR	DEL	Deletes current character
RUBOUT	Backspace	Deletes previous character
	Tab	Move to extremes
Right arrow (—>)	Right arrow	Cursor right 1 character
Left arrow (<—)	Left arrow	Cursor left 1 character
Up arrow (↑)	Up arrow	Cursor up 1 character
Down arrow (↓)	Down arrow	Cursor down 1 character
RETURN	RETURN and/or ENTER	In text editor: inputs a carriage return to signify a new line At systems level: enters a command

The new software that runs on the EDA-300 and EDA-700 no longer outputs task abort and dismount messages. Other error messages appear on the message line rather than at the bottom of the screen. They are output in the following format:

TASK "XXXXXX" TERMINATED. aa:bbbbbb ERR=cc IO=ddd

xxxxxx	the six character task name
aa	a two letter abort code
bbbbbb	user PC of the error
ccc	Pascal error code (if applicable)
ddd	I/O status code if Pascal I/O error

If the error message appears, please record the message before rebooting the system, and contact customer support for the appropriate action to be taken.

4.6 Using the Mouse and Keyboard on the EDA-700

On the EDA-700, you use the mouse to select commands from the command menus, and you use the keyboard to define and input explicit information to the system.

The mouse, which is illustrated in Figure INTRODUCTION-3 below, is a mechanical device that you can move about any flat surface.

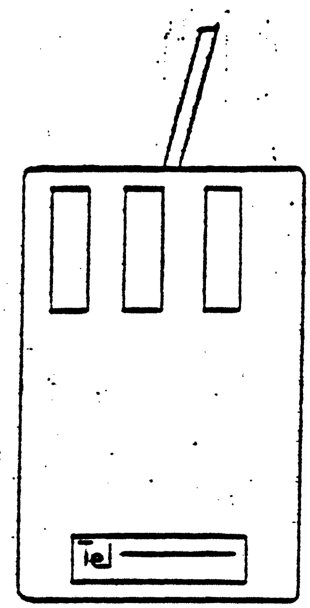


Figure INTRODUCTION-3: The Mouse

Moving the mouse allows you to position the menu cursor (reverse-video rectangle) on one of the commands on the currently displayed menu at the top of the graphics screen. When you position the menu cursor over the command, you may press the left button on the mouse to select the command. If the command requires explicit keyboard input (such as a filename), enter the information on the keyboard. As you enter information, the system displays it on the message line portion of the command menu. To execute the command, press the <Enter> key on the keyboard.

The mouse has three buttons, which are as follows:

- o Left Button (Select) - Use the left button on the mouse to select commands from the command menus and to execute interactive drawing tasks such as |ADD CONNECTION| or |ADD LINE|. When you select commands from a menu on the graphics screen, the system displays a menu cursor (reverse-video rectangle) that moves dynamically with the movement of the mouse. During an interactive command, such as |ADD CONNECTION|, the system displays the graphics screen cursor (cross-hair). Use the left button on the mouse to select the points on the drawing defining the connection.

  When you are digitizing with the left button on the mouse during a command such as |ADD CONNECTION|, you can see the explicit XY coordinates by pressing the middle button on the mouse. This displays the softkey and short message line. Then use the keypad menu to enter explicit XY coordinates in order to define a precise location.

- o Middle Button (Menu) - Use the middle button on the mouse to blank and redisplay the current menu or softkey line. You can do this during any interactive command.

- o Right Button - The right button on the mouse is comparable to the softkey command NEXT. This allows you to reexecute the last command you just executed without having to return to the menu. This command is to be used only during interactive commands, for example, |ADD RECTANGLE|.

The EDA 700 keyboard contains all the keys that the ordinary typewriter has. In addition, it contains a number of special function keys that the typewriter does not have. All used keys and their functions are listed below.

Key	Function
F1 - F10	Variable (see list below)
Esc	Cancel
↑	Shift
Alt	Alternate function of pressed key
<-	Backspace
<-	Return and enter

Note that there are several keys with arrows. Figure INTRODUCTION-4 below shows the differences between these various keys.

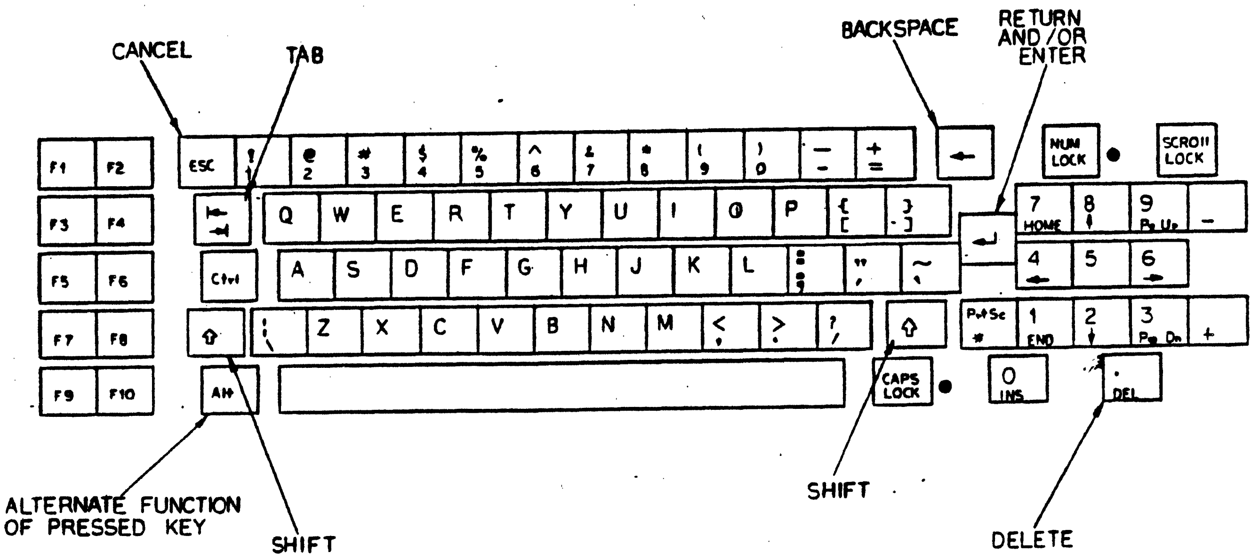


Figure INTRODUCTION-4: The EDA 700 Keyboard

Use the keyboard to define and enter explicit data. For example, the |SET ACTIVE LAYER| command requires you to input a database layer number. When you execute |SET ACTIVE LAYER|, the system issues the LAYER prompt on the message line portion of the command menu. When you input a layer number on the keyboard, the input is displayed immediately after the LAYER prompt. If you make an error during input, use the backspace key (<-) to delete the error. To execute and to complete the command, press the <Enter> key on the keyboard. To cancel the command, press the <Esc> key at the top left portion of the keyboard. The <Esc> key is equivalent to the CANCEL softkey command; you can use it when either the mouse or the keyboard is the active device.

When you are entering multi-line text into a drawing while on the EDA-300, you enter the carriage return on the keyboard to indicate a space between lines. However, on the EDA 700, the carriage return performs the function of entering the command. Therefore, to enter multi-line text into a drawing while on the EDA 700, hit the <ALT> key while simultaneously hitting the <ENTER> key.

You can also use the keyboard to execute softkey commands within a currently active command, or outside of a command with a menu blanked or displayed.

You can use the function keys located to the left of the keyboard to execute the softkeys. The entire set of function keys corresponds to whatever softkey menu is currently on the screen. That is, the function keys always correspond to the currently active softkey line. Thus, F1 corresponds to the first function on the left, F2 corresponds to the second function on the left, etc.

As on the EDA 300, the menus on the EDA 700 are arranged in a hierarchy. For the printed circuit board (PCB) design application (EDA-3000), the following function keys correspond to the family of four menu set.

F1 = DONE	F2 = WORLD
F3 = ROAM	F4 = <-MENU
F5 = MENU->	F6 = STATUS
F7 = DISPLAY	F8 = NOT USED
F9 = ZOOM	F10 = NOT USED

F7 (display) in the above menu brings up the following function keys:

F1 = DONE	F2 = WORLD
F3 = ROAM	F4 = BLANK
F5 = UNBLANK	F6 = ZOOM : 1
F7 = ZOOM	F8 = NOT USED
F9 = ZOOM	F10 = NOT USED

F6 (status) in the menu of menus brings up the following menu:

F1 - F7 = Return to menu page	
	F8 = NOT USED
F9 = ZOOM	F10 = NOT USED

The above three menus contain keys to be executed outside an interactive command. In contrast, the next two menus shown contain softkeys that are executed during a printed circuit design command. These also can be executed with a menu either blanked or displayed.

F1 = SWAP	F2 = NEXT
F3 = DRILL	F4 = KEYPAD
F5 = NECK	F6 = OOPS!
F7 = ENTER	F8 = NOT USED
F9 = ZOOM	F10 = NOT USED

F1 = NOT USED	F2 = NOT USED
F3 = NOT USED	F4 = NOT USED
F5 = PRINT	F6 = <- PAGE
F7 = PAGE ->	F8 = NOT USED
F9 = ZOOM	F10 = NOT USED

Note that the display monitor cannot display both the pop-up menu and the design graphics at the same time if the hardware zoom feature is set to 2:1 or 4:1. You can alternate between displaying the menu or design graphics, or you can reset the hardware zoom to 1:1.

4.7 EDA-700 Adjustment Controls

Following are the control knobs for the EDA-700:

- Power On/Off (red button) - The power on/off button is located at the front of the processing unit. To power the system on, press the red button. The button will light up, showing that the system is powered on. The system will then proceed with the boot-up process and will display the data/time menu.

  To power down the system, press the red button; the light will go out.

CAUTION: Be sure to close any open file before powering the system off. Save any open files and return to the |NEW PROJECT/OLD PROJECT| menu before turning the power off.

- o Graphics Screen - Power On/Off — The graphics screen power on/off knob (located on the back of the monitor) allows you to turn the monitor on and off. It is recommended that the monitor remain on at all times.

- o Graphics Screen - Brightness — Use the BRIGHT knob (located on the back of the monitor) to adjust the graphics screen brightness.

4.8 Rebooting the EDA-700

Rebooting is the procedure you use to bring the system back into operation if it has stopped running. For example, if a power failure occurs during system operation, you must reboot the system when power is restored. On the EDA-700, system boot-up messages will not be displayed on the graphics screen.

Use the RESET button to reboot the system; this knob is located on the front of the processing unit, to the right of the power on/off button. When you press the reset button, the system locks the currently active drawing under its current name with the revision label X.... You should not reopen this drawing, due to the possible loss of its integrity. If an open file exists, and if the system must be rebooted, contact the Customer Service Center.

Refer to the EDA-3000 and/or EDA-4000 manual for additional information on procedures for rebooting the system.

During boot-up, the system displays the boot-up menu shown below:

DONE				OOPS!	ENTER
			PLEASE ENTER DAY:		
-JAN-	-FEB-	7	8	9	
-MAR-	-APR-	4	5	6	
-MAY-	-JUN-	1	2	3	
-JUL-	-AUG-		:	0	
-SEP-	-OCT-				
-NOV-	-DEC-				
		AM		PM	

The first prompt displayed on the message line portion of the menu is:

PLEASE ENTER DAY

Select the day by moving the mouse so that the menu cursor (reverse video rectangle) is positioned over the appropriate day; then press the left button on the mouse. The day you selected will appear immediately after the prompt. Move the menu cursor to the "ENTER" command; then press the left button on the mouse.

The next prompt displayed on the menu is:

PLEASE ENTER MONTH:

Move the menu cursor to the appropriate month; then press the left button on the mouse. The month you selected will be displayed on the message line immediately following the "PLEASE ENTER MONTH" prompt. Move the menu cursor to the ENTER command and press the left button on the mouse.

The system prompts for the year:

PLEASE ENTER YEAR: 19

Select the year (entering only the last two numeric values) by pressing the left button on the mouse; then select "ENTER".

The system now prompts for the time:

PLEASE ENTER TIME: HR:MIN AM/PM):

Select the appropriate information from the menu using the left button on the mouse. After selecting all appropriate information, move the cursor to the "ENTER" command, and press the left button on the mouse.

The system then displays a message on the menu message line showing the date and time you selected from the boot-up menu:

"THE TIME IS: 08-APR-85 08:30 AM. HIT DONE OR OOPS!

If the information you entered is correct, move the menu cursor to the "DONE" command; then press the left button on the mouse. If the information is incorrect, select "OOPS!" and reenter the correct information.

After you have entered the correct information, the system displays the applications menu (if you have a system configured for both PCB and mechanical design applications). This allows you to select the appropriate applications software. The menu is shown below:

PLEASE PICK DESIRED APPLICATION	
PCB	MD

Move the menu cursor to the application you wish to select, and then press the left button on the mouse to boot the system to the selected design application software. The system then displays the following menu:

PLEASE PICK DESIRED APPLICATION	BOOT-UP CONTINUING

When the boot-up process is complete, the system displays the project menu shown below:

			STATUS
RPU ACCESS	NEW PROJECT	OLD PROJECT	FILE MGT/ARCHIV
TO MD SYSTEM		SYSTEM LIBRARY	ACCESS REMOTE
	USER MENUS		ENB FILE TRANS
SELF LOAD	EXECUTE MENU		FREE DISK SPACE
CURRENT INDEX			

For information on how to use the commands, see other sections of this manual. When giving instructions on how to execute commands, this manual refers to the light pen and function screen keyboard; on the EDA-700, the mouse and keyboard are used in their place. For example, when this manual tells you to input a pick using the light pen (to specify a location to the system), use the left button of the mouse.

4.9 Using the Floppy Disk Drive on the EDA-700

The floppy disk drive is located on the front of processing unit, directly above the POWER and RESET buttons. To open the floppy drive door, press the long rectangular button directly on the front of the drive.

When inserting floppy disks for archiving or transfer of files, be sure that the floppy is installed with the manufacturer's label up and the WRITE/ENABLE tab to the left. Use the "FILE MGT/ARCHIV" command to open the appropriate menus for archiving and transfer of files to the system. For further information on how to use the file management and archiving commands, see the File Management & Archives section of this manual.

B A S I C S

- O G E T T I N G   S T A R T E D
- O G R A P H I C S
- O T E X T   F I L E S
- O T E L E S I S   K E Y B O A R D   T E X T   E D I T O R

GETTING STARTED

1. TURNING THE SYSTEM ON AND OFF

2. USING THE LIGHT PEN AND MENUS

3. ENTERING THE DATE AND TIME

4. UNDERSTANDING WHAT A FILE IS

5. THE BOOT-UP MENU: OPENING A PROJECT FILE

6. THE MENU OF MENUS: CHOOSING THE WORK YOU WANT TO DO

1. TURNING THE SYSTEM ON AND OFF AND REBOOTING

LOCATION OF SYSTEM CONTROLS

Power, reset, and adjustment controls are all located below the graphics screen. Lift the hinged panel to gain access to them.

MAIN POWER SWITCH: TURNING THE SYSTEM ON AND OFF

CONTROL SWITCH	ACTION	RESULT
ON-1/OFF-0 SWITCH (white switch at far left)	Switch to ON-1	After about 2 minutes, the function screen will display the DATE AND TIME menu. The system is ready to operate.
	Switch to OFF-0	Power to the system is turned off. BE CAREFUL: Close any open file before you turn the power off. We recommend that you always return to the BOOT-UP menu before turning the power off.

THE RESET SWITCH: REBOOTING THE SYSTEM

If the system ceases to function, you can bring it back into operation by rebooting. Before rebooting, call the Telesis Customer Support Center if you want to try to save the drawing that was active at the time the system stopped functioning.

CONTROL SWITCH	ACTION	RESULT
RESET SWITCH	Press the RESET switch. Then proceed as you would when first turning on the system.	When you press RESET, the system closes your current file under its name and revision label****. You should not reopen a drawing file closed in this way because you cannot be sure of its integrity. Ordinarily you should delete the file and open a previous revision to continue your work unless you have contacted the Customer Support Center to determine whether the drawing might be salvageable.

NOTE: BEFORE REBOOTING, CHECK FOR A RESPONSE FROM THE SYSTEM:

Before rebooting, always check to be sure that the system has actually ceased to function.

Failure to respond to a command may simply mean that the system is engaged in a lengthy task such as NET COMPARE.

TO CHECK FOR A RESPONSE:

Pick any menu box (except |CANCEL|) with the light pen. You may even pick an empty menu box.

- If the system beeps, it is still functioning even if it does not respond to the command.
- If the system does not beep, it has either ceased to function or it is engaged in cancelling a command. Look at the command echo to see whether the last command was |CANCEL|. If it was not, go ahead and reboot. If the last command was |CANCEL|, wait several minutes and test again for a response before rebooting.

ADJUSTMENT AND OTHER CONTROLS

CONTROL SWITCHES AND KNOBS	PURPOSE
POWER: ON/OFF SWITCH (below graphics screen)	Always leave this switch in the ON position. It turns the function screen and graphics screen on and off.
CONTRAST KNOB (below graphics screen)	Use this to adjust graphics screen contrast.
BRIGHTNESS KNOB (below graphics screen)	Use this to adjust graphics screen brightness.
DEGAUSS BUTTON	Press this button occasionally to renew color purity on the graphics screen.
FUNCTION SCREEN: BRIGHTNESS KNOB	Use this to adjust function screen brightness.
KEYBOARD/REMOTE SWITCH	Always leave this switch at KEYBOARD position unless you are using remote diagnostics. (Contact Customer Support Center.)

2. USING THE LIGHT PEN AND MENUS

2.1. WHAT IS INPUT?

DEFINITION

You must communicate with the system to operate it. Communication from you to the system is called input.

TYPES OF INPUT

There are two types of input:

1. COMMANDS - You input commands to tell the system what you want it to do.

2. EXPLICIT DATA - You input explicit data to specify coordinate point values, file names, layer numbers, and other specific information.

EXAMPLE:

This is the input you would use to start a new project named ALPHA:

|NEW PROJECT| |A|L|P|H|A| |ENTER|

command — explicit data — command

|NEW PROJECT| tells the system to create a new project file.

|A|L|P|H|A| tells the system to name the file ALPHA.

|ENTER| tells the system: input is complete, go ahead and create the file.

ALL INPUT IS MADE WITH LIGHT PEN AND MENUS

All of your input is made by using the light pen to select commands or data from the menu pages displayed on the function screen.

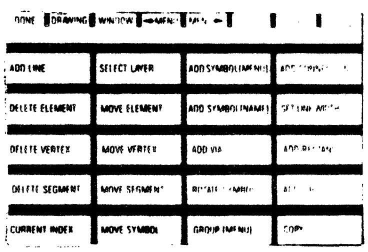
2.2. MENU FORMATS

TYPES OF MENUS

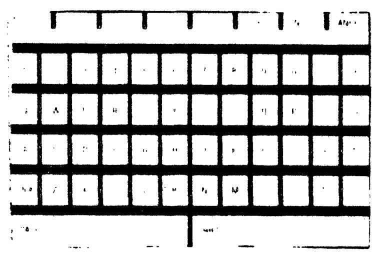
Commands and data are arranged on menus in these formats:

- COMMAND MENUS
- 2 KEYBOARD MENUS
- 1 TABLET MENU
- 1 KEYPAD MENU

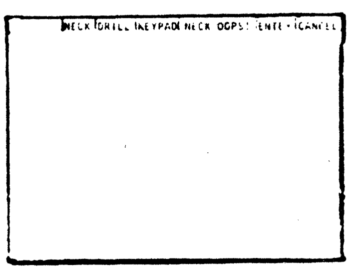
An example of each format is shown below:



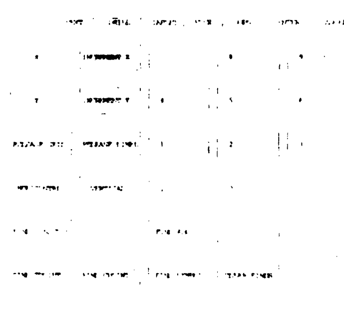
COMMAND MENU



KEYBOARD MENU



TABLET MENU



KEYPAD MENU

MENU BOXES

Except for the tablet, all commands and data are enclosed in menu boxes.

The most frequently used commands are arranged in the eight boxes across the top of each menu.

MESSAGE LINE

The message line is just below the top row of commands. On the left of the message line, the system displays an echo of the last command. On the right, it displays an echo of input, prompts, error messages and other short messages.

LONG MESSAGES

The system uses the entire function screen area to display long messages. To view the next page of a long message, pick the |->PAGE| command. To return to the previous function screen menu, pick |DONE| or |CANCEL|.

2.3. HOW TO USE THE LIGHT PEN

To input commands or data, point to your selection on the menu and press down on the face of the function screen with the tip of your light pen. This is called a PICK.

The system beeps to acknowledge receipt of your input, and beeps again when ready for further input. (Often these two beeps will be almost simultaneous.)

BE CAREFUL!: As you move the light pen over the surface of the function screen, be sure to apply pressure only when you want to input the command or data beneath the pen.

MENU BOX INPUT

You may pick any part of the menu box containing the command or data you want, but it's a good idea to aim for the center of the box.

TABLET INPUT

When you point the light pen at the blank part of the tablet menu, you will see a cursor appear on the graphics screen in the form of a large blue crosshair.

The location of the crosshair on the graphics screen corresponds to the location of the light pen on the tablet. It moves as you move the light pen, helping you to choose the precise location you want on your drawing.

When the crosshair is at the point you want, you pick as with any other menu input. Be careful to point straight down.

The terminal will beep, and the crosshair will disappear momentarily.

A small stationary crosshair will appear on the graphics screen marking the location you picked.

The large crosshair reappears when the system is ready for further input.

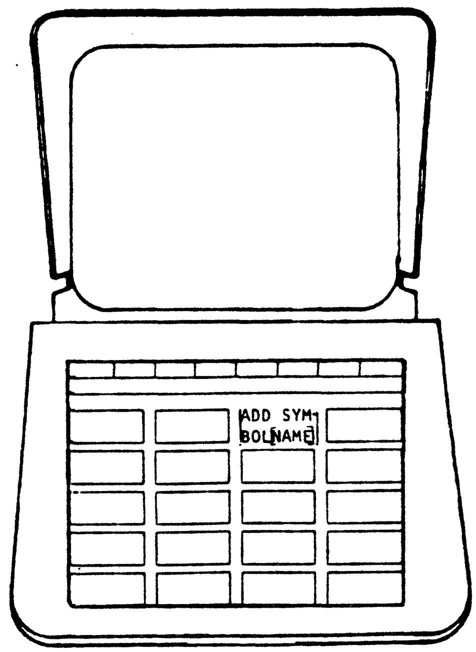
2.4. SYSTEM RESPONSE TO YOUR INPUT

The system responds to your input in several ways:

- Echoes your command input on the message line.
- Echoes your data input on the message line.
- Prompts you for further input when appropriate.
- Flips you to a new menu page when appropriate.
- Sends you an error message when appropriate.
- Performs your command.

EXAMPLE OF SYSTEM RESPONSE TO YOUR INPUT

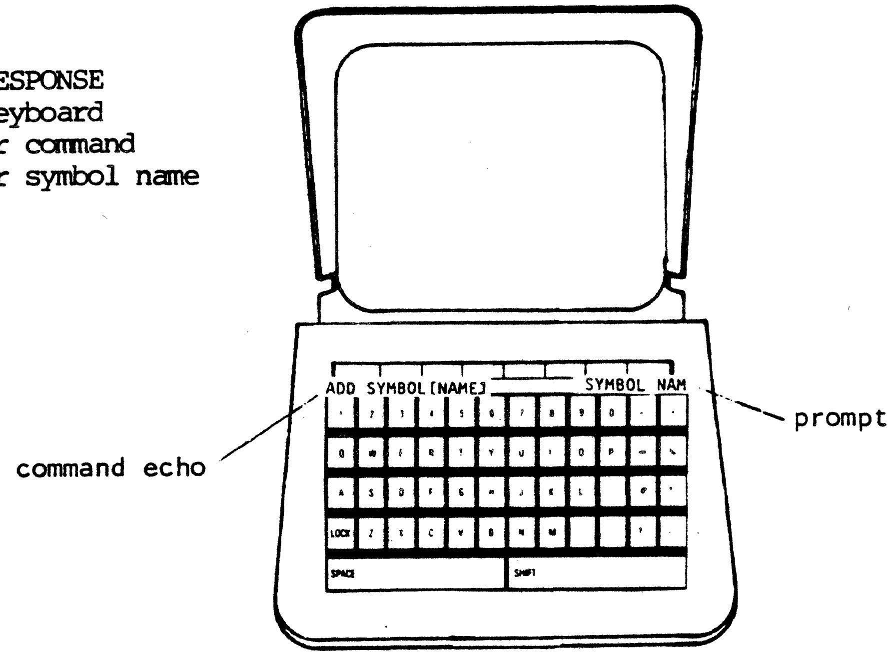
BEGINNING ON ONE OF THE SCHEMATIC MENUS



YOUR INPUT | ADD SYMBOL [NAME] |

FUNCTION SCREEN RESPONSE

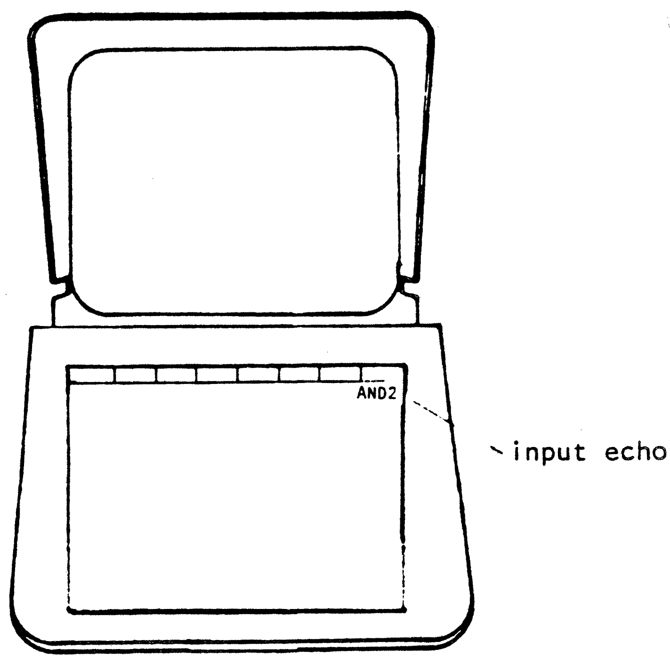
- displays keyboard
- echoes your command
- prompts for symbol name



YOUR INPUT |A|N|D|2| |ENTER|

FUNCTION SCREEN RESPONSE

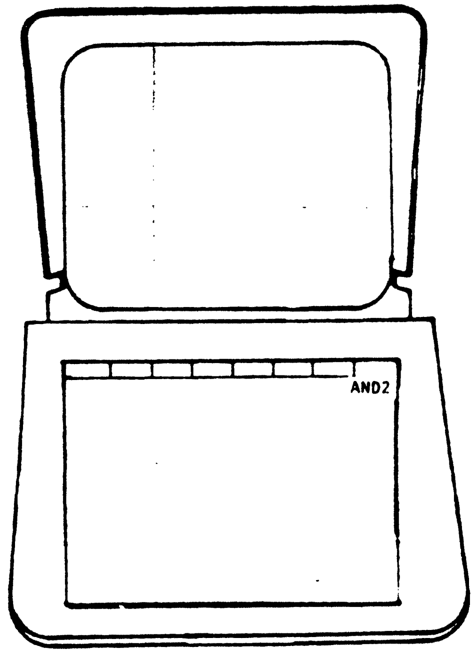
- displays tablet
- echoes your input



YOUR INPUT (Point the light pen at the tablet)

GRAPHICS SCREEN RESPONSE

- large crosshair appears



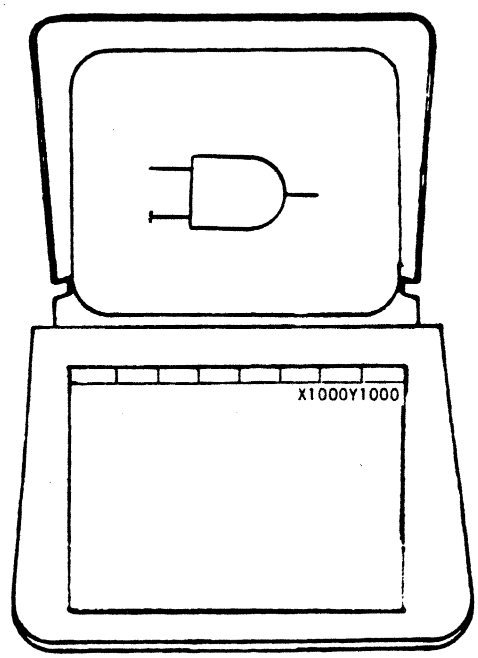
YOUR INPUT (P) (When the large crosshair is at the desired position, press down with the light pen)

FUNCTION SCREEN RESPONSE

- echoes your input

GRAPHICS SCREEN RESPONSE

- small crosshair appears at picked location
- symbol appears in highlight color (red)



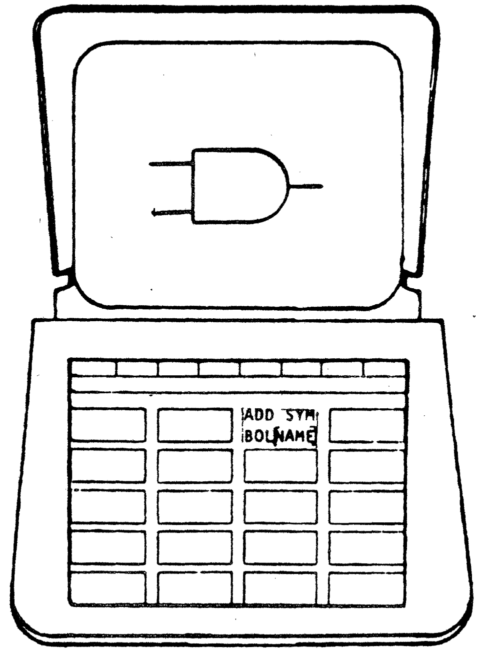
YOUR INPUT |ENTER|

GRAPHICS SCREEN RESPONSE

- displays symbol in color of active layer

FUNCTION SCREEN RESPONSE

- returns to original command menu



2.5. COMMANDS THAT HELP YOU WITH YOUR INPUT

These commands are used only with other commands to help you with your input or to cancel it. They are found in the top row of menus where they are needed:

COMMAND	WHEN TO USE IT.	EXAMPLES AND COMMENTS
|CANCEL|	To cancel a command when you have decided not to complete it.	- Pick |ADD LINE| - Function screen displays tablet - Pick |CANCEL| - Function screen returns to command menu and system ignores the |ADD LINE| command.
|ENTER|	To indicate that your input is complete. Used with commands that must be followed by further input.	- Pick |ADD LINE| - Function screen displays tablet - Pick Ⓟ Ⓟ Ⓟ Ⓟ |ENTER| - Graphics screen displays a line with 3 segments
|NEXT|	To indicate that your input is complete, but you want to continue using the same command.	- Pick |ADD LINE| - Function screen displays tablet - Pick Ⓟ Ⓟ Ⓟ Ⓟ |NEXT| Ⓟ Ⓟ Ⓟ Ⓟ |ENTER| - Graphics screen displays two lines with three segments each
|OOPS!|	To cancel your last single, data input.	- Pick |ADD LINE| - Function screen displays tablet - Ⓟ Ⓟ Ⓟ Ⓟ |OOPS!| |OOPS!| |ENTER| - Graphics screen displays a line with one segment, and ignores your last 2 Ⓟ s. When you pick |OOPS!|, the system will echo, backspace and delete your last input. EX: INPUT: |S| |Y| |S| |T| |O| |M| The word "system" is misspelled. PICK: |OOPS!| |OOPS!| The inputs |O| and |M| are deleted from the message line. INPUT: |E| |M| |ENTER| RESULT: SYSTEM

2.6. HOW TO GET FROM ONE MENU TO ANOTHER

MENUS FLIP AUTOMATICALLY WHEN APPROPRIATE

The system automatically flips (changes the display on the function screen) from one menu to another when you pick a command that requires a new menu. For example, when you pick |DRAW SCHEMATIC|, the system flips to one of the menus used for drawing a schematic.

COMMANDS TO USE FOR GETTING FROM ONE MENU TO ANOTHER

When you do not pick a command that requires a new menu, you use these commands to flip menu pages:

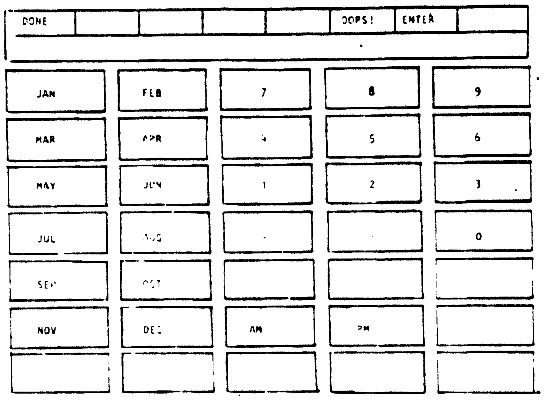
<table>
<tr><th>COMMAND</th><th>WHEN TO USE IT</th><th>COMMENTS</th></tr>
<tr><td>|DONE|</td><td>When you have finished with a menu page.</td><td>When you pick |DONE| the function screen will flip to the menu you are most likely to need after the one just used.</td></tr>
<tr><td>|MENU->|
and
|<-MENU|</td><td>To display an adjacent menu in a menu set.</td><td>Some closely related menus are looped together so that you can easily flip from one to another. For example, there are 3 menus containing logic symbol names.

|MENU->| :
SYMBOL MENU 1
↙ ↖
SYMBOL MENU 2 → SYMBOL MENU 3

|<-MENU| :
SYMBOL MENU 1
↗ ↘
SYMBOL MENU 2 ← SYMBOL MENU 3</td></tr>
</table>

3. ENTERING THE DATE AND TIME

When you turn on the system, the first menu displayed on the function screen is the DATE & TIME MENU.



The system prompts you on the message line for each item of information you must input:

<table>
<tr><th>MESSAGE LINE</th><th>EXAMPLE OF INPUT</th></tr>
<tr><td>1. PLEASE ENTER DAY</td><td>|1|4| |ENTER|</td></tr>
<tr><td>2. PLEASE ENTER MONTH</td><td>|MAY| |ENTER|</td></tr>
<tr><td>3. PLEASE ENTER YEAR:19</td><td>|8|2| |ENTER| (only enter the last two digits of the year)</td></tr>
<tr><td>4. PLEASE ENTER TIME (HR: MIN AM/PM)</td><td>|2|:|1|5|PM| |ENTER| (follow the exact sequence shown in the prompt)</td></tr>
<tr><td>5. THE TIME IS 14-MAY-82 2:15 PM PICK OOPS! OR DONE</td><td>Pick |DONE| if the date and time are correct.
Pick |OOPS!| to start over if they are not correct.</td></tr>
</table>

After you pick |DONE|, there will be an interval of 5-7 minutes before the system is ready to begin working. When it is ready, the system will beep and display the BOOT-UP menu on the function screen.

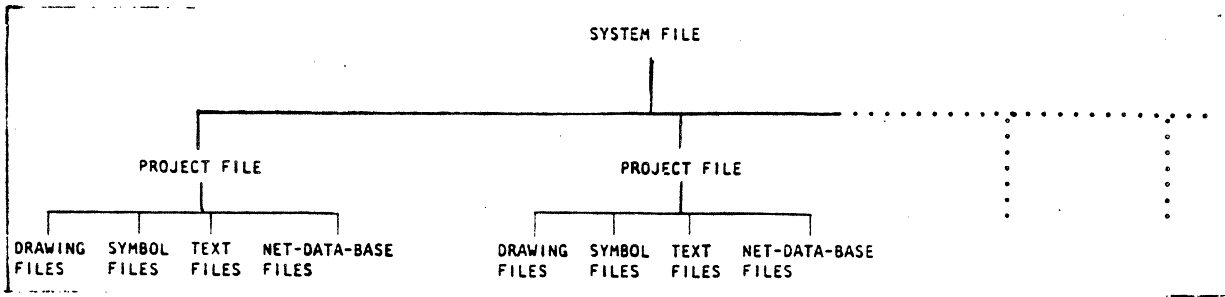
4. UNDERSTANDING WHAT A FILE IS

You use the BOOT-UP MENU to open a project file by naming it.

Before you can do this, you need to know how the Telesis file structure works and how to name files.

4.1 FILE STRUCTURE

All information on the system is organized into a hierarchy of files:



4.2. DEFINITIONS OF TYPES OF FILES

SYSTEM FILE	This is the master file. It contains all project files.
PROJECT FILE	A project file typically contains the drawing files, text files, and net data base files relating to the design of one printed circuit board. SYSTEM-LIBRARY is the name of a project file that is used as a library. It typically contains symbol files, pin record files and other files that are likely to be used by more than one project.
DRAWING FILE	Each drawing file contains a single drawing, typically a board drawing or a schematic sheet.

SYMBOL FILE	Each symbol file contains the graphics for a single symbol, typically a logic symbol or a component symbol.
TEXT FILE	A text file contains data in text format, for example, a pin record file for a component symbol, a net comparison report, a list of design rules, a list of design rules violations.
NET-DATA-BASE FILE	This file contains the net data base for a single board design.

4.3. MOST FILES ARE CREATED BY YOU

On a new Telesis system, the system file is empty.* The other types of files are created by you as you work.

4.4 FILE NAMES AND REVISION LABELS

Each time you create a new file, you assign a name to it. (Except for the net data base file; the system automatically names this NET-DATA-BASE.) You then use that name whenever you want to open that file again.

You may also assign a revision label to a file. If you do not, the system will label your file: revision 1.

* The only exception is that a SYSTEM-LIBRARY project file containing Telesis-prepared symbol files may be provided on your system.

The Command Description Section of the manual tells you how to input names and revision labels for each command (like |NEW PROJECT|) that opens a file.

The table below gives you the general rules to follow.

RULES FOR NAMES AND REVISION LABELS

- PROJECTS -- You must give a unique name or revision label to each project currently on the system.

  ALLOWED:
  project file ALPHA revision 1
  project file ALPHA revision 2
  project file BETA revision 1

  NOT ALLOWED:
  project file ALPHA revision 1
  project file ALPHA revision 1

- DRAWING FILES, TEXT FILES, SYMBOL FILES -- You must give a unique name or revision label to each file of the same type within a single project file.

  ALLOWED:
  drawing file TESTER revision A
  text file TESTER revision A

  NOT ALLOWED:
  drawing file TESTER revision A
  drawing file TESTER revision A

- FILE NAMES -- may be up to 18 alphanumeric characters with no blank spaces allowed.

- FILE REVISION LABELS -- may be up to 4 alphanumeric characters with no blank spaces allowed. To input a revision label, follow the file name with a single blank space; then input the revision label: |T|E|S|T|E|R|_|A|

4.5 FILE MANAGEMENT AND ARCHIVES

FILE MANAGEMENT

The system's file management capability allows you to copy files, delete files, and to change file names and revision labels. (Refer to the FILE MANAGEMENT Section of the Manual for instructions on how to use these capabilities.)

ARCHIVES

The system's archival capability allows you to copy files from the system to floppy disks or magnetic tape (optional), and to copy files back to the system from disk or tape.

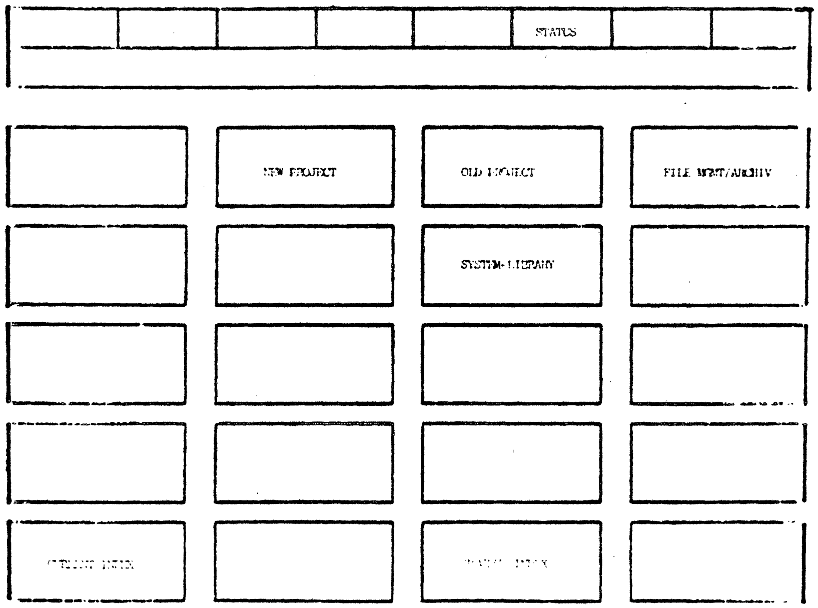
You need this capability to:

1. Free space on the system when space is limited.

2. Keep back-up copies of your work.

(Refer to the FLOPPY DISK and MAGNETIC TAPE Sections of this Manual for instructions.)

5. THE BOOT-UP MENU: OPENING A PROJECT FILE



The first menu you see after entering the date and time is the BOOT-UP MENU.

You are at the system file level whenever you are on the BOOT-UP MENU whether you: a) just turned on the system, or b) flipped back to this menu after working on the system.

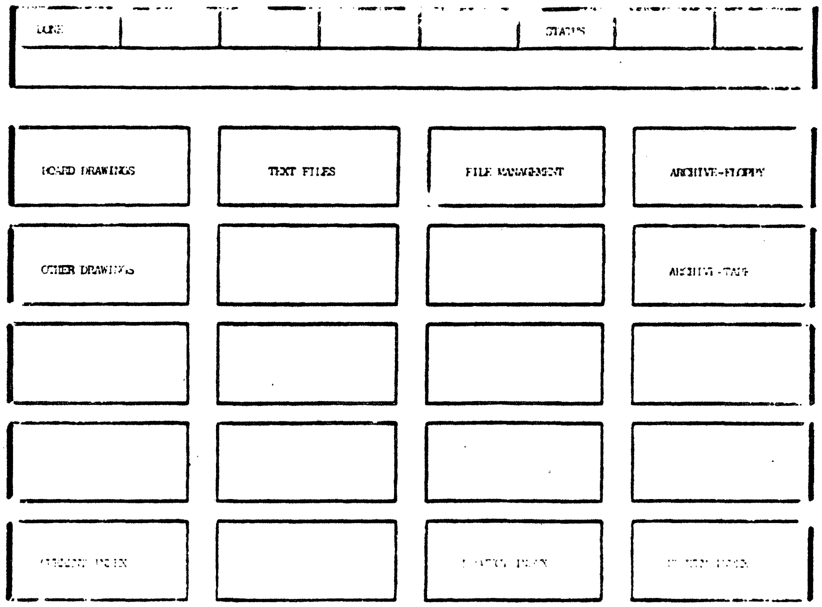
The four major commands on the BOOT-UP MENU allow you to choose whether to open a project file or to stay at the system level. The other commands provide you with information.

The COMMAND DESCRIPTION Section of the manual tells you how to use each of these commands. The table below tells you when to use them.

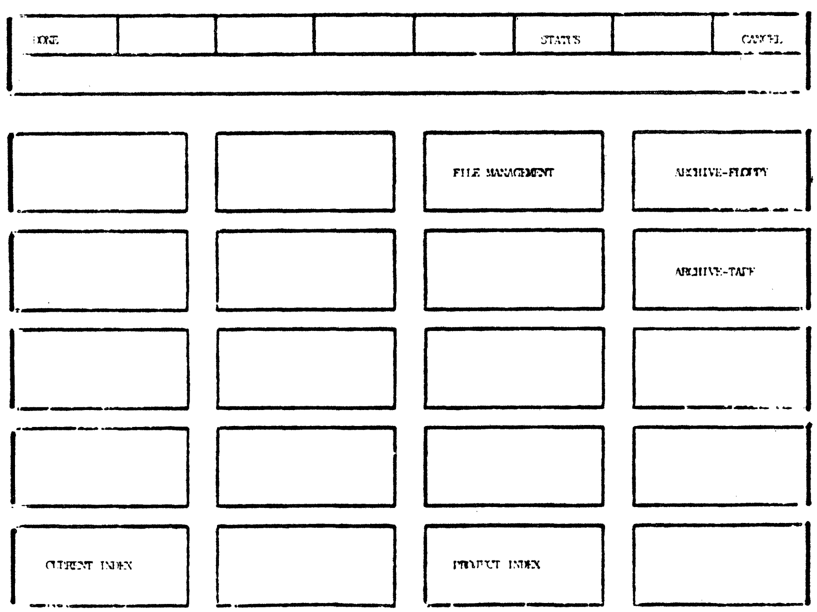
COMMAND	WHEN TO USE IT
|NEW PROJECT| (followed by project name)	To create and name a new project file so that you can create new drawings, text files, and/or symbols.
|OLD PROJECT| (followed by project name)	To open an existing project file so that you can use the existing files in it, or add new files to it.
|FILE MGMT/ARCHIV|	To use the system's file management and archival capabilities without opening any project file. (You remain at the system file level.)
|SYSTEM-LIBRARY|	To open the existing project file named SYSTEM-LIBRARY. Before you can use this command, you must already have used the |NEW PROJECT| command to create and name the SYSTEM-LIBRARY file (unless a Telesis-prepared SYSTEM-LIBRARY was provided with your system.)
|STATUS|	You usually use this command with the drawing menus. It displays on the function screen the name of the current project file, drawing file, and drawing parameters. When you use it on the BOOT-UP MENU, it will inform you that you are in the system file, and it will display the default drawing parameters.
|CURRENT INDEX|	To display on the function screen the names and revision labels of all projects currently on the system.
|PROJECT INDEX| (followed by project name)	To display on the function screen the names and revision labels of all drawing files, text files, net data base files, and symbol files currently in the specified project file.

6. THE MENU OF MENUS: CHOOSING THE WORK YOU WANT TO DO

When you pick |OLD PROJECT|, |NEW PROJECT|, or |SYSTEM-LIBRARY| on the BOOT-UP MENU, the function screen flips to a PROJECT MENU OF MENUS:



When you pick |FILE MGMT/ARCHIV| on the BOOT-UP MENU, the function screen flips to a SYSTEM LEVEL MENU OF MENUS:



You use the major commands on these menu pages to tell the system what kind of work you want to do:

COMMAND	WHEN TO USE IT
BOARD DRAWINGS	To work on a board drawing
OTHER DRAWINGS	To work on a symbol or schematic drawing
TEXT FILES	To work on a text file
FILE MANAGEMENT	To use the file management capability
ARCHIVE-FLOPPY	To use the floppy disk archive capability
ARCHIVE-TAPE	To use the magnetic tape archive capability

When you pick one of these commands, the function screen flips to the menu pages needed to do the work you have chosen.

THE INDEX COMMANDS

- On the PROJECT MENU OF MENUS:

  CURRENT INDEX displays the contents of the currently open project.

  SYSTEM INDEX displays the contents of the system file.

- On the SYSTEM LEVEL MENU OF MENUS:

  CURRENT INDEX displays the contents of the system file.

GRAPHICS

1. GENERAL INFORMATION

You can create three types of drawings with the Telesis system:

- SYMBOL DRAWINGS
- SCHEMATIC DRAWINGS
- BOARD DRAWINGS

You use a different menu set for each drawing type (SYMBOL MENUS, SCHEMATIC MENUS, BOARD MENUS), but most of the commands on these menus sets are the same basic drawing commands.

This section of the Manual tells you how to use these basic commands.

The sections of the manual covering symbol creation, schematics and boards, instruct you in the commands and procedures used exclusively for each drawing type.

2. OPENING A DRAWING FILE

STEP 1: |BOARD DRAWINGS| or |OTHER DRAWINGS|

After you have opened a project file on the BOOT-UP MENU, select |BOARD DRAWINGS| or |OTHER DRAWINGS| from the MENU OF MENUS. (|OTHER DRAWINGS| is used for opening schematic or symbol drawings.)

EXAMPLE:
- Pick |OTHER DRAWINGS|
- Function screen display
  - NEW/OLD DRAWING MENU

STEP 2: |NEW DRAWING| or |OLD DRAWING|

Pick |NEW DRAWING| or |OLD DRAWING| depending on whether you want to create a new drawing file, or work on an existing one. |OLD DRAWING| must be followed by the name of the drawing file.

EXAMPLE:
- Pick |NEW DRAWING|
- Function screen display
  - DRAWING SIZE MENU

EXAMPLE:
- Pick |OLD DRAWING|
- Function screen display
  - KEYBOARD
- Pick |D|I|O|D|E| |ENTER|
- Function screen display
  - DRAWING TASK MENU

STEP 3: SETTING DRAWING SIZE (for new drawings only)

The DRAWING SIZE MENU gives you 5 choices:

|A SIZE| - 8500 by 11000 mils

|B SIZE| - 17000 by 11000 mils

|C SIZE| - 22000 by 17000 mils

|D SIZE| - 32000 by 22000 mils

|OTHER SIZE| - you specify the drawing size

NOTE

Maximum drawing extents are + 32000 and - 32000.

When you pick A, B, C, or D SIZE, the 0/0 point of your drawing is at the lower left, and all coordinate point values in your drawing are positive. After you pick, the function screen flips to the NEW DRAWING NAME MENU.

When you pick |OTHER SIZE|, the function screen flips to the KEYBOARD and prompts you for your drawing extents. You may input negative values depending on where you want to locate the 0/0 point of your drawing.

EXAMPLE:

- Pick |OTHER SIZE|
- Function screen display
  - KEYBOARD & prompt: LX (lower x value)
- Pick |-|1|0|0|0|0| |ENTER|
- Function screen display
  - KEYBOARD & prompt: LY (lower y value)
- Pick |-|1|0|0|0|0| |ENTER|
- Function screen display:
  - KEYBOARD & prompt: UX (upper x value)
- Pick |1|0|0|0|0| |ENTER|
- Function screen display
  - KEYBOARD & prompt: UY (upper y value)
- Pick |1|0|0|0|0| |ENTER|
- Function screen display
  - NEW DRAWING NAME MENU

In this case, your drawing size is 20,000 by 20,000 mils with the 0/0 point at the center.

STEP 4: |NEW DRAWING NAME| (for new drawings only)

Use this command to name your new drawing file.

EXAMPLE:

- Pick |NEW DRAWING NAME|
- Function screen display
  - KEYBOARD
- Pick |D|I|O|D|E| |ENTER|
- Function screen display
  - DRAWING TASK MENU

STEP 5: |DESIGN BOARD|, |DRAW SCHEMATIC|, |DRAW SYMBOL|

If you started in STEP 1 with the |BOARD DRAWINGS| command, you will now be on the BOARD DRAWING TASK MENU. Pick |DESIGN BOARD| to flip to the first menu in the BOARD MENU SET.

If you started in STEP 1 with the |OTHER DRAWINGS| command, you will now be on the OTHER DRAWINGS TASK MENU. Pick |DRAW SCHEMATIC| or |DRAW SYMBOL| to flip the first menu in the SCHEMATIC MENU SET or the SYMBOL MENU SET.

3. PARAMETERS: SETTING THEM AND USING THEM

Parameter commands control the contents of the display on the graphics screen. They are found on the first menu in each drawing menu set, although they are sometimes repeated on other menus in the set.

These are the drawing parameter commands:

LAYER COMMANDS:

|SET ACTIVE LAYER|
|BLANK LAYER|
|DISPLAY LAYER|
|ASSIGN RED|
|ASSIGN BLUE|
|ASSIGN GREEN|
|ASSIGN VIOLET|
|ASSIGN YELLOW|

GRID COMMANDS:

|SET GRID SIZE|
|BLANK GRID|
|DISPLAY GRID|

TRAP COMMAND:

|SET TRAP SIZE|

LINE LOCK COMMANDS:

|LINE LOCK 45/ON|
|LINE LOCK 90/ON|
|LINE LOCK OFF|

LINE & POINT SIZE COMMANDS:

|SET LINE WIDTH|
|SET CNCT PNT SIZE|

You use these commands to set the parameters of your drawing before you begin to draw, and to change parameters as needed while you work.

You are not required to set all drawing parameters. Each parameter has a default value. But you will find that different drawing types require different parameter settings. These requirements will be discussed in the sections of this manual covering symbol, schematic, and board drawings. You will also find that you need to change some settings, such as GRID SIZE, to help you as you work on different areas of your drawing.

THE |STATUS| COMMAND

The |STATUS| command is very useful to you in using the parameter commands. When you pick |STATUS|, the function screen displays all the current parameter settings--default settings, as well as ones input by you.

3.1 SETTING LAYER PARAMETERS

LAYERS

Every drawing you create has up to 256 layers. They are numbered 0-255.

To operate the system properly, you must place certain elements of your drawing on distinctive layers. For example, we recommend that you place reference designators on layer 54, and connect lines on layer 1. Recommended layer standards are shown in Table GRAPHICS-I.

The system reserves some layers for its own use. For example, it places ratsnest lines on layer 101. Reserved layers are shown with an asterisk (*) in Table GRAPHICS-1.

It is very important to adhere to layer standards when placing elements. For example, when you create a pen-plot or artwork tape, you use layer numbers to tell the system what elements of your drawing you want to include in the plot or artwork. (See Plotter and Artwork Sections of this manual.) For this to work properly, you must have consistently placed elements on the required layers.

You specify the layer on which an element is placed by using the |SET ACTIVE LAYER| command before adding an element. For example,

- Pick |SET ACTIVE LAYER|
- Function screen display
  - KEYBOARD
- Pick |5|4| |ENTER|
- Function screen display
  - PARAMETER MENU

All elements you add will now be placed on layer 54 until you change the active layer by using the |SET ACTIVE LAYER| command again.

You may only add elements to the active layer, but you may edit the elements on any visible layer.

All layers of your drawing are visible on the graphics screen unless you choose to blank a layer (or layers) from the screen by using the |BLANK LAYER| command.

The drawing you see on the graphics screen looks as if it is all in one plane. You can visibly distinguish one layer from another by assigning different colors to different layers. There are five colors available to you: red, blue and green, violet and yellow.

06-JUNE-83 * = HARD CODED (ELEMENTS MUST EXIST ON DESIGNATED LAYERS)

TELESIS LAYER STANDARDS FOR SCHEMATIC SYMBOLS AND DRAWING

LAYER	DESIGN REFERENCE
0 . . . . *	CONNECT POINTS FOR SYMBOLS
1 . . . . *	CONNECT LINES
36	DRAWING FORMAT, TITLE BLOCKS & NOTES
52	LOGIC SYMBOL GRAPHICS (LINES, ARCS ETC)
53	PIN NUMBERS (SHOWN ON SCHEMATIC)
93	PIN NUMBERS (NOT SHOWN ON SCHEMATICS)
54	REFERENCE DESIGNATORS
55	DEVICE TYPE (SHOWN ON SCHEMATIC)
87	DEVICE TYPE (NOT SHOWN ON SCHEMATICS)
86	SIGNAL NAMES (NOT SHOWN ON SCHEMATICS)
64	SIGNAL NAMES (SHOWN ON SCHEMATICS)
148	PIN USE CODE
149	FUNCTION TYPE
150	FUNCTION DESIGNATOR
151	LOGICAL PIN NAME (SHOWN ON SCHEMATICS)
85	LOGICAL PIN NAME (NOT SHOWN ON SCHEMATICS)
152	COMPONENT VALUES

TELESIS LAYER STANDARDS FOR PACKAGE SYMBOLS AND DESIGN DRAWINGS

LAYER	DESIGN REFERENCE
0 . . . *	CONNECT POINTS FOR SYMBOLS (THRU HOLES)
1 . . . *	COMPONENT SIDE CONNECT LINES
2 THRU 14 . . . * .	SOLDER SIDE CONNECT LINES, IMBEDDED PWR & SIGNAL LAYERS
17 THRU 32	RESERVED FOR FUTURE TRACK LAYERS
15	COMPONENT SIDE SOLDER MASK
16	SOLDER SIDE SOLDER MASK
33	CARD OUTLINE
34	PLATING BAR
35	BOARD DIMENSIONS (ASSEMBLY DWG)
88	BOARD DIMENSIONS (DRILL DWG)
36	DRAWING FORMAT, TITLE BLOCK & NOTES
40 . . . *	COMPONENT KEEPOUT AREA (COMPONENT SIDE)
70 . . . *	TRACK KEEPOUT FOR ALL TRACK LAYERS
71 THRU 84. . . * .	TRACK KEEPOUT FOR BRD. LAYER (ROUTE OUTSIDE ONLY)
44 . . . *	VIA KEEPOUT AREAS
51	PACKAGE SYMBOL GRAPHICS
52	PAD GRAPHICS
53	COMPONENT PIN NUMBERS
54	REFERENCE DESIGNATORS
55	DEVICE TYPE
89	SILKSCREEN
90	TOOLING CORNERS
91	EDGE CONNECTOR SIGNAL NAMES
101 . . . *	RATSNEST CONNECT LINES
102 . . . *	RATSNEST HISTOGRAM
98	N/C DRILL LEGEND HEADER
99 . . . *	N/C DRILL FIGURES & LEGEND
103	N/C DRILL PATHS
104	N/C INSERT PATHS
131 THRU 144. . . .	PHYSICAL DESIGN RULE CHECK *
146 . . . *	OUTLINE FOR TRACK ROUTING (ROUTE INSIDE ONLY)
147 . . . *	OUTLINE FOR AUTO PLACEMENT (PLACE INSIDE ONLY)
154	COMPONENT OUTLINES (AUTO PLACEMENT)
200 THRU 255. . . .	RESERVED BY TELESIS

Table GRAPHICS-I: LAYER STANDARDS

These standards are recommendations. You may establish your own layer standards if you wish, as long as you adhere to them consistently. For example, you could choose to place reference designators on layer 52 and symbol graphics on layer 54.

The only layer standards you are required to use exactly as they are shown in this table are those marked with an asterisk (*).

BE CAREFUL!: The system does not prevent you from using these * layers for other purposes. But some system functions will not operate properly if you do.

WHEN TO USE THE LAYER COMMANDS

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|SET ACTIVE LAYER|	Use this, before adding elements, to specify the layer on which they are to be placed.
|BLANK LAYER|	Use this to make a layer or layers of your drawing invisible on the graphics screen. For example, pin numbers are on layer 53; if you don't care about seeing them while you are designing, blank layer 53. Use it also to eliminate unwanted layers when outputting a matrix plot of your drawing. BE CAREFUL:! Make sure you do not blank the active layer, or activate a blank layer. You cannot place elements on a blank layer.
|DISPLAY LAYER|	Use this to make a blank layer visible again.
|ASSIGN RED| |ASSIGN BLUE| |ASSIGN GREEN| |ASSIGN VIOLET| |ASSIGN YELLOW|	Use these commands to assign colors to layers. You are free to assign any color to any layer in order to improve the visibility of your drawing. The default layer color is green, so unless you assign red or blue to a layer, the elements on it will be drawn in green. You may use these commands to change a layer color even after you have placed elements on that layer. When you do, both existing elements and future elements will be displayed in the new drawing color.

NOTE: When assigning new colors to displayed layers on Telesis 2.1 systems use the |ZOOM RATIO| command at 1:1 before continuing. Otherwise, a double image of updated elements will occur if |ZOOM RATIO| is not used.

3.2 SETTING GRID PARAMETERS

THE GRID

The grid is a matrix of points that the system superimposes on your drawing to serve 2 purposes:

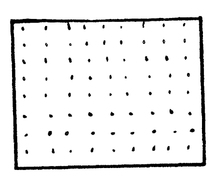
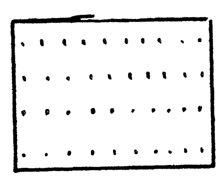
1. Visual aid — The grid assists you in judging distances and angles in the same way that graph paper assists you in manual drawing.

2. Construction tool — When you pick (P)s from the tablet, the element you are drawing snaps to the grid, that is, the system places your (P) on the nearest grid point. The snap grid capability allows you to draw straight lines and precise angles quite easily with the light pen and tablet.

The system's default grid size is x:100, y:100. This setting places a grid point at every 100th mil in both x and y axes.

WHEN TO USE THE GRID COMMANDS

The Command Description Section of the manual tells you how to use the grid commands. The table below tells you when to use them.

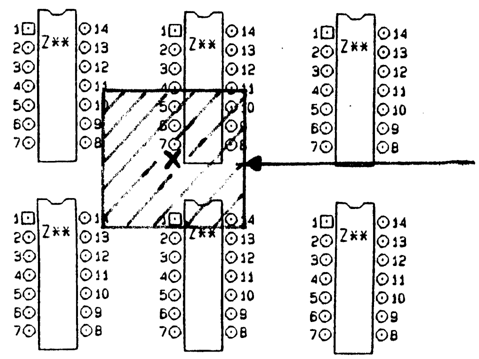
COMMAND	WHEN TO USE IT
SET GRID SIZE	Use this command to set the grid size. The system will prompt you for x and y values. You may set x and y to different values: - Grid setting x:100, y:100  - Grid setting x:100, y:200 

	Recommended grid sizes for symbols, schematics and boards are discussed in the manual sections covering each drawing type.
BLANK GRID	Use this command to turn off the grid display. You may want to do this to get an uncluttered view of your drawing, or to output a matrix plot of your drawing. When you blank the visible grid, it continues to act as a snap grid. NOTE: If you have zoomed into a small area of your drawing and are working with a small grid size, be sure to blank the grid before zooming out to view the whole drawing. Otherwise, the grid display will be so dense that it will obscure your view.
DISPLAY GRID	Use this to make a blanked grid visible again.

3.3 SETTING THE TRAP PARAMETER

The trap, like the grid, is a construction tool.

There are several commands that require you to point to an element using the light pen and tablet (i.e., DELETE ELEMENT). The system does not require your pick to be exactly on the element. Instead, the system searches for your element in an area around your pick. This area is called the trap. The following figure illustrates the trap.



TRAP AREA

OPERATOR PICK ON GRAPHICS SCREEN SHOWN AS X

The trap is a square (invisible to you) with your pick at the center. The default trap size is 150 mils by 150 mils.

The system uses the trap every time you input a command that requires the system to find an element in your drawing. The following is a partial list of some of the commands that use the trap :

DELETE ELEMENT
COPY ELEMENT
MOVE ELEMENT
ADD CONNECTION
ATTACH SIGNAL NAME
(and any command that requires a pick on the graphics screen)

WHEN TO USE THE TRAP PARAMETER COMMAND

The Command Description section of this manual details the procedures to use the SET TRAP SIZE command. The table below tells you when to use it.

COMMAND	WHEN TO USE IT
SET TRAP SIZE	We recommend that you begin by setting the trap size at one and one half of your current grid size. Reset the trap size every time you reset the grid size.

After you become accustomed to the system, you may wish to deviate from this recommendation.

BE CAREFUL!: When you are working with connect lines with width, use of the trap can be tricky. We recommend these trap settings:

1. If your line width is less than your grid size, set the trap size to be equal to the grid size.

EXAMPLE: LINE WIDTH = 12 mils
GRID SIZE = 25 by 25 mils
TRAP SIZE = 25 mils

2. If your line width is greater than your grid size, set the trap size to be equal to the line width + 2.

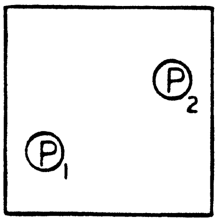
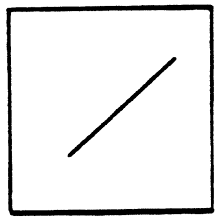
EXAMPLE: LINE WIDTH = 17 mils
GRID SIZE = 15 by 15 mils
TRAP SIZE = 19 mils

3.4 THE LINE LOCK PARAMETER

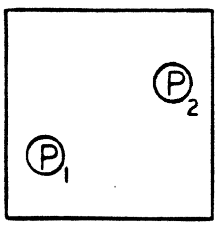
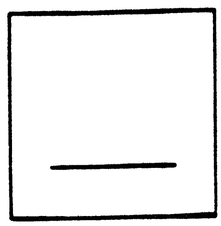
LINE LOCKING

Line locking is another construction tool. It requires the system to draw lines that are vertical, horizontal, or at a 45° angle, even when your P is not in vertical, horizontal, or 45° angle alignment with your last P. For example:

WITH LINE LOCK OFF:

ACTION	RESULT
	

WITH LINE LOCK 90/ON:

ACTION	RESULT
	

WHEN TO USE THE LINE LOCK PARAMETER COMMANDS

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them. These commands affect all subsequent input until you reset the parameter.

COMMAND	WHEN TO USE IT
|LINE LOCK 90/ON|	Use this when you want the system to draw only horizontal and vertical lines, for example--when you are using the |ADD CONNECTION| command.
|LINE LOCK 45/ON|	Use this when you want the system to draw only horizontal, vertical and 45° angle lines.
|LINE LOCK OFF|	Use this to turn the line lock parameter off.

3.5. LINE & POINT SIZE PARAMETERS

These commands allow you to specify the width of lines and the size of connect points to be used in your drawing.

The line width parameter affects the lines of circles and arcs as well as plain lines and connect lines.

The default line width is 0.

The default connect point size is 50 mils.

WHEN TO USE THE LINE & POINT SIZE PARAMETER COMMANDS

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

These commands affect all subsequent input until you reset the parameter.

COMMAND	WHEN TO USE IT
SET LINE WIDTH	SYMBOL DRAWINGS -- Use 0 line width. BOARD DRAWINGS -- Set and change the line width as needed for connect lines with width. SCHEMATIC DRAWINGS -- Usually you will use 0 line width for connect lines, but you may use lines with width if you wish to.
SET CNCT PT SIZE	The default connect point size (50 mils) is large and clearly visible in most drawings. Use this command to make them smaller (or larger) if you prefer. (The system will recognize a connect point even if you set its size to 0.)

4. ELEMENTS OF TELESIS DRAWINGS

DEFINITION OF AN ELEMENT

You create a drawing by adding and editing graphic elements.

An element is a piece of graphics that the system treats as a single unit. For example, even though a line may have many segments, the |DELETE ELEMENT| command deletes all segments of the line. And even though a symbol may have arcs, lines, and connect points in it, the |ROTATE SYMBOL| command rotates the whole symbol, not just the line or arc you hit with your (P).

TYPES OF ELEMENTS

These are the types of elements you may add and edit:

LINES
ARCS
CIRCLES
CONNECT POINTS
SYMBOLS
CONNECT LINES
TEXT LABELS & NOTES

Each type of element is described below.

(Section 5. discusses the ADD and EDIT commands for LINES, ARCS, CIRCLES, CONNECT POINTS, SYMBOLS, and CONNECT LINES.)

(Section 6. discusses the use of TEXT LABELS and NOTES.)

4.1 LINES

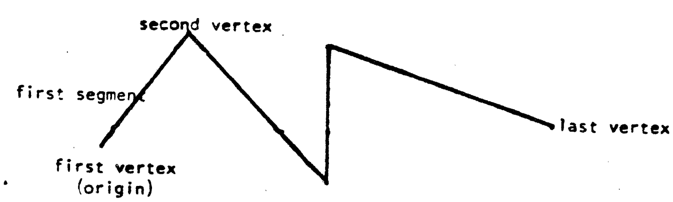
A line is a set of line segments linked together. We call each "corner" point, including the endpoints of the line, a "VERTEX." A line in the Telesis system may have up to 125 vertices.

A LINE:

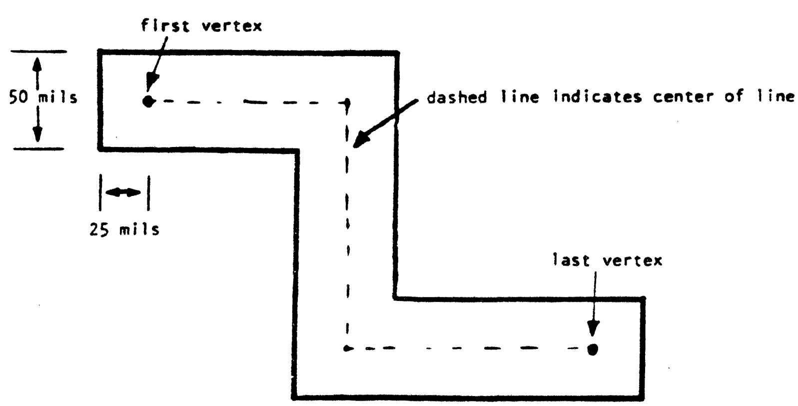
- Is at the layer that was the active layer when the line was added to the drawing.
- Has a width that was the set width when the line was added to the drawing.
- Has an origin, which is the first vertex picked when the line was originally added to the drawing. Note that the origin of the line will change relative to the other vertices if you move the first vertex.
- Cannot be used to make a logical connection.

NOTE ON LINE WIDTH: If a line or connect line has non-zero width, it is displayed as a boundary around the center of each line segment at the set line width. At the two endpoints of the line, you will find that the boundary has been extended by one-half the line width. This is to show more accurately what the line will actually look like when it is photoplotted.

EXAMPLE - LINE WIDTH SET TO 0 (ZERO) MILS:



EXAMPLE - LINE WIDTH SET TO 50 MILS:



4.2 ARCS AND CIRCLES

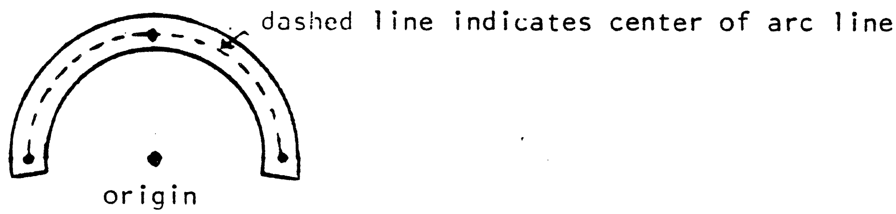
ARCS and CIRCLES

- Are at the layer that was the active layer when the arc or circle was added to the drawing.

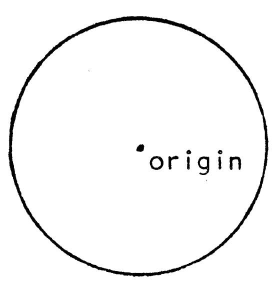
- Have a width that was the set line width when the arc or circle line was added to the drawing.

- Has an origin, which is the first center about which the arc or circle is drawn.

EXAMPLE - ARC WITH LINE WIDTH SET TO 50 MILS:



EXAMPLE - CIRCLE WITH LINE WIDTH SET TO 0 (ZERO) MILS:



4.3 CONNECT POINTS

A connect point is a point defined on a drawing where a connect line can be attached. You normally add connect points only into a drawing that will be used to create a symbol. These connect points then represent the pins of the corresponding physical device. You can also create symbols with connect points that will be used simply to terminate a connect line to an on-page or off-page net.

The abbreviation for connect point is CPOINT.

Every connect line must connect on both ends to a connect point before the Telesis EXTRACT NET, NET COMPARE, and DESIGN RULES CHECK operations can work correctly. Therefore, whenever you command the system to start or end a connect line on another connect line, the system automatically breaks the existing connect line into two, adds a connect point to the drawing, and joins all three connect line ends to the new connect point. (See example.)

CONNECT POINT SIZE

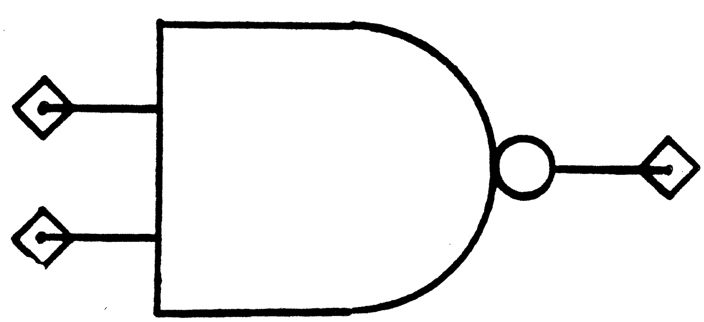
Since the Telesis system cannot actually display a dimensionless point, it displays a vertical daimond at each point where a connect point has been added to a drawing. You can control the size of the connect point diamond by using the SET CONNECT POINT SIZE command. You state the size in mils, and this is the distance from the bottom to the top of the diamond. NOTE that this diamond is only visible on the graphic display and dot-matrix plots. It is not actually part of the data base, and does not pen plot nor photoplot.

NOMENCLATURE

In this manual, we usually refer to the connect points on a symbol as PINS. To be more exact, a pin is a connect point with a pin number label attached to it. We use pin numbers to declare an exact correspondence between a connect point in a schematic diagram and the connect point on a physical package symbol on the PC board drawing.

EXAMPLE - CONNECT POINTS ON A LOGIC SYMBOL:

diamonds show
connect points



4.4 SYMBOLS

Symbols are like pre-set appliques that you "paste-up" onto a schematic or printed circuit drawing on the Telesis system. Each symbol must have been defined once by creating a drawing on the Telesis system the same size as the required symbol, and with all the elements in it required to be in the symbol. And the |CREATE SYMBOL| command must have been used to change the drawing file to a symbol file.

Symbols allow you to draw an unlimited number of instances of the same logic function or physical package on your drawing, each instance taking only a few seconds to add. Each symbol is recognized by the Telesis system as representing a certain device, and having a certain name (ref des), once you have annotated those labels on the symbol.

SYMBOLS:

- Are created from a drawing by the |CREATE SYMBOL| command. The created symbol is the same size as the drawing it was created from, and has all the same elements as the original drawing.

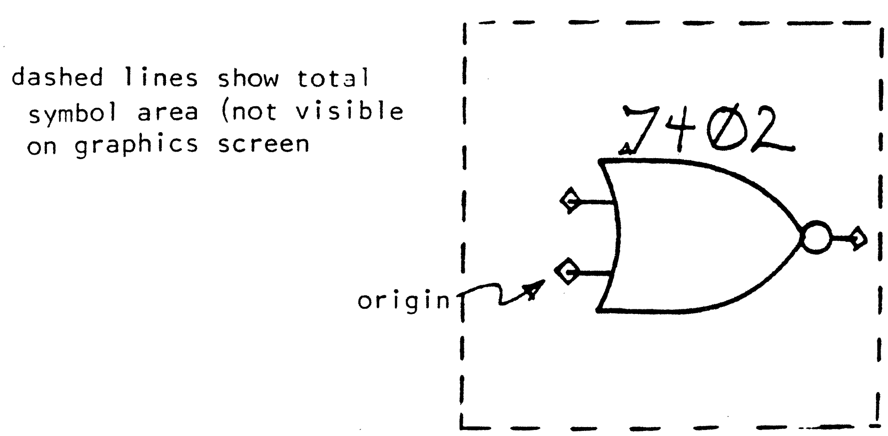
- Are not on any one layer. However, each line, arc, connect point, and label making up a symbol is at the layer it was put on in the original symbol drawing.

- Have a scale which is one-to-one with the original drawing, when they are first added to the drawing. You can later change the scale to make the symbol larger or smaller in the drawing.

- Have a rotation, which is zero degrees when they are first added to a drawing. You can later rotate the symbol around its origin in the drawing in increments of 90 degrees.

- Contains other elements: lines, arcs, circles, connect points, and/or text labels. Even after a symbol is placed in a drawing, these elements can be individually edited with EDIT commands such as |DELETE ELEMENT|, |MOVE ELEMENT|.

EXAMPLE



4.5 CONNECT LINES

A connect line is a set of line segments linked together, normally with both ends of the line connected to connect points. As with a line each corner of a connect line is called a vertex. A connect line may have up to 125 vertices.

The abbreviation for connect line is CLINE.

A CONNECT LINE:

- Is at a layer that was the active layer when the line was added to the drawing.
- Has a width that was the set width when the line was added to the drawing.
- Has an origin, which is the present position of the first position of the first vertex picked when the line was first created.

NOTE ON LINE WIDTH: If a line or connect line has non-zero width, it is displayed as a boundary around the center of each line segment at the set line width. At the two endpoints of the line, you will find that the boundary has been extended by one-half the line width. This is to show more accurately what the line will actually look like when it is photoplotted.

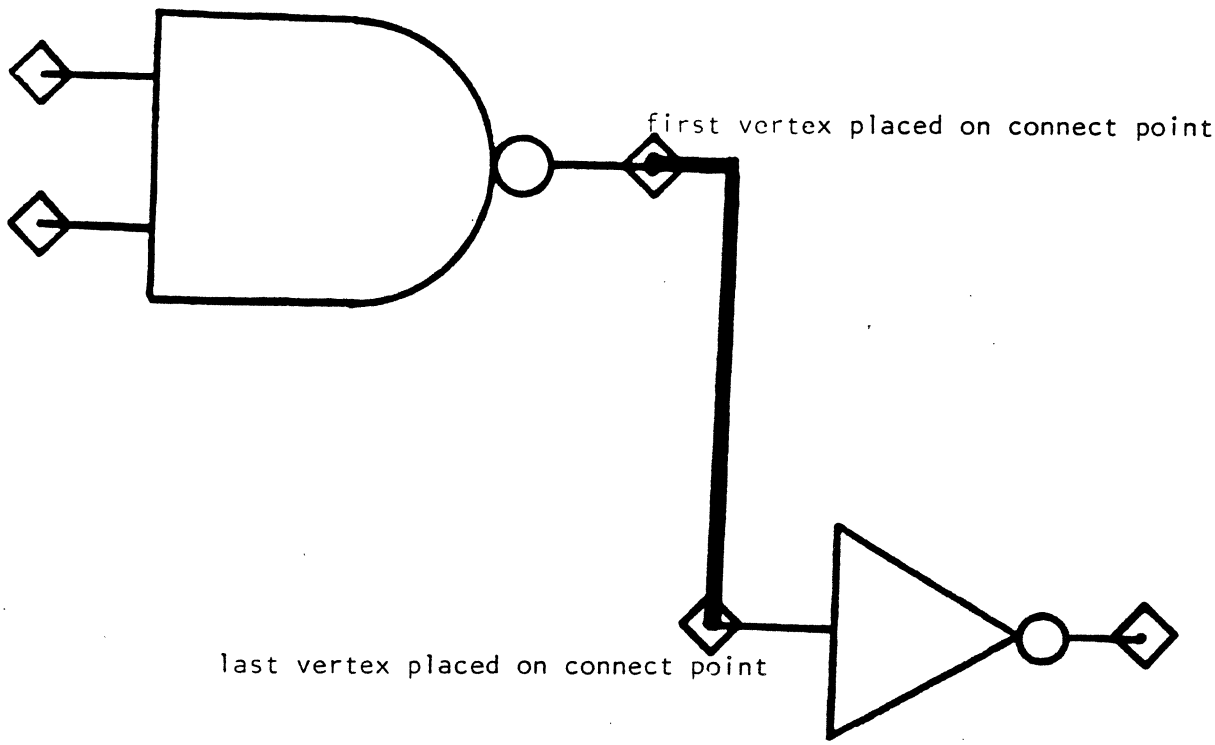
UNCONNECTED CONNECT LINES

The Telesis system allows you to make partial connections amongst symbols, and leave some connections to be completed at a later time in the creation of a drawing. This means you may end a connect line in a blank area of your drawing, without connecting the line to another connect line or connect point. These connect lines are considered incomplete (or floating end) and the |EXTRACT NET| and |NET COMPARE| commands will issue error messages to their log text files unless all such connections have been extended to a connect point as part of the completion of the drawing.

RELATION TO CONNECT POINTS

When you are connecting a connect line to a connect point or another connect line, place the end point of the line directly on the point to which you want to connect. Do not cross the point with a segment or intermediate vertex of the connect line.

EXAMPLE



4.6 TEXT LABELS AND NOTES

TEXT POINTS

When you place any text label or note in a drawing, the system places a text point at the lower left corner of the text. This is the reference point for the text. All text parameters are associated with this point, so that whenever you replace the text, the new text will have the same parameters as the old text.

The abbreviation for text point is TPOINT.

TEXT PARAMETERS

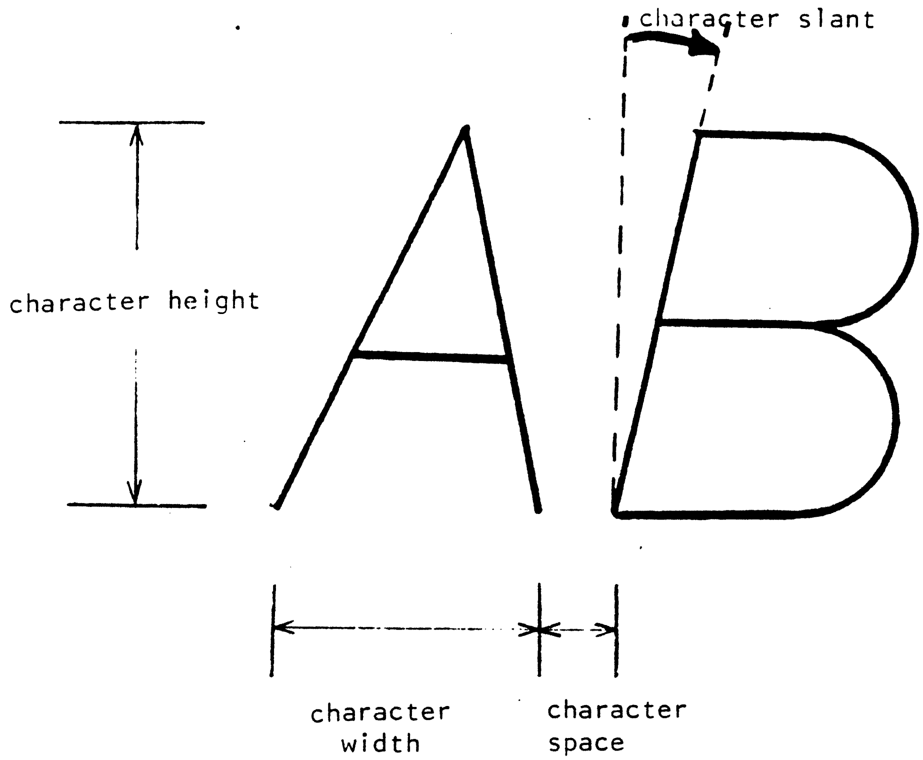
Each text point has a set of text parameter values associated with it. These parameters are:

- Height, width, and spacing of the characters on the text point.
- Slant of each character in degress.
- Angle of the line of characters in degrees.
- Justification of the text about the text point origin (left, center, or right justified).

TEXT POINT SIZE

Since the Telesis system cannot actually display a dimensionless point, it displays a triangle at each point where a text point has been added to a drawing. You can control the size of the displayed text point by using the SET TEXT POINT SIZE command. You state the size in mils, and this is the distance from the bottom to the top of the triangle. NOTE that this triangle is only visible on the graphic display and dot-matrix plots. It is not actually part of the data base, and does not pen plot nor photoplot. (The text itself does plot, of course.)

EXAMPLES



RIGHT JUSTIFIED: $_{\Delta}$ABC

CENTER JUSTIFIED: A$\underset{\Delta}{\text{B}}$C

LEFT JUSTIFIED: ABC$_{\Delta}$

5. ADDING AND EDITING ELEMENTS

Once you have set the parameters for your drawing, you are ready to begin placing graphics in it using the ADD and EDIT commands.

ADD/EDIT COMMANDS

Any element you add to a drawing can be edited as needed. Below are the ADD commands and the corresponding commands used to edit the element added.

ADD COMMANDS	CORRESPONDING EDIT COMMANDS
|ADD LINE|	|DELETE ELEMENT|, |DELETE SEGMENT|, |DELETE VERTEX| |MOVE ELEMENT|, |MOVE SEGMENT|, |MOVE VERTEX| |COPY ELEMENT|, |INSERT VERTEX|, |CHANGE LAYER|, |CHANGE LAYER AND WIDTH|, |CHANGE WIDTH|
|ADD ARC|	|DELETE ELEMENT|, |MOVE ELEMENT|, |COPY ELEMENT|
|ADD CIRCLE|	|DELETE ELEMENT|, |MOVE ELEMENT|, |COPY ELEMENT|
|ADD CONNECT PNT|	|DELETE ELEMENT|, |MOVE ELEMENT|, |COPY ELEMENT|
|ADD CONNECTION|	|DELETE ELEMENT|, |DELETE SEGMENT|, |DELETE VERTEX| |MOVE ELEMENT|, |MOVE SEGMENT|, |MOVE VERTEX| |COPY ELEMENT|, |CHANGE SEGMENT LAYER|, |INSERT VERTEX|, |CHANGE LAYER AND WIDTH|, |CHANGE WIDTH|, |MOVE SECTION|, |CHANGE LAYER|
|ADD SYMBOL [NAME]| |ADD SYMBOL [MENU]|	|DELETE SYMBOL|, |MOVE SYMBOL|, |SCALE SYMBOL|, |ROTATE SYMBOL|

Each of these commands is discussed in Sections 5.2 and 5.3 below.

5.1. USING THE TABLET AND THE KEYPAD

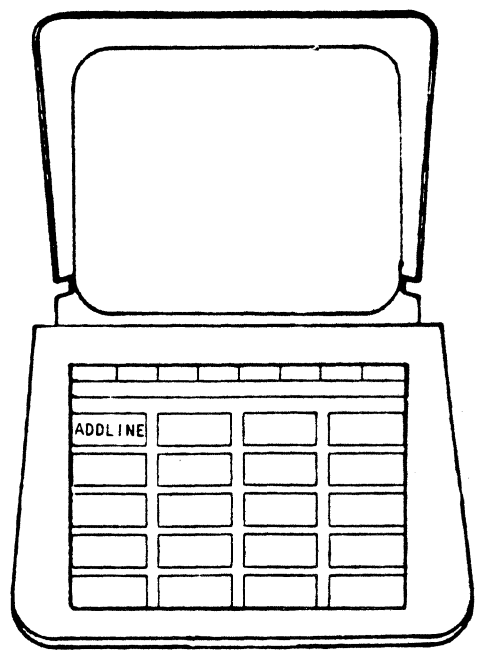
You draw an element on the graphics screen by picking one of the ADD commands and then placing the element in your drawing using either the tablet or the keypad.

With the tablet you specify a location by pointing to it. With the keypad you specify a location by inputting coordinate values.

With edit commands, you also have the choice of making a tablet pick or using the keypad.

EXAMPLE OF TABLET METHOD

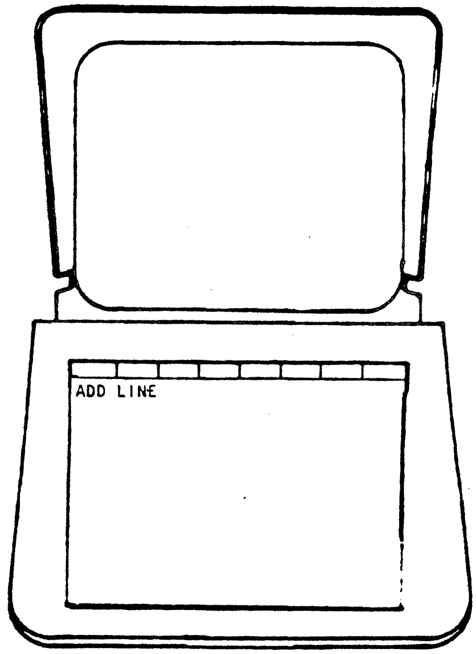
SYSTEM DISPLAY



YOUR INPUT [ADD LINE]

FUNCTION SCREEN RESPONSE

- displays tablet
- echoes your command



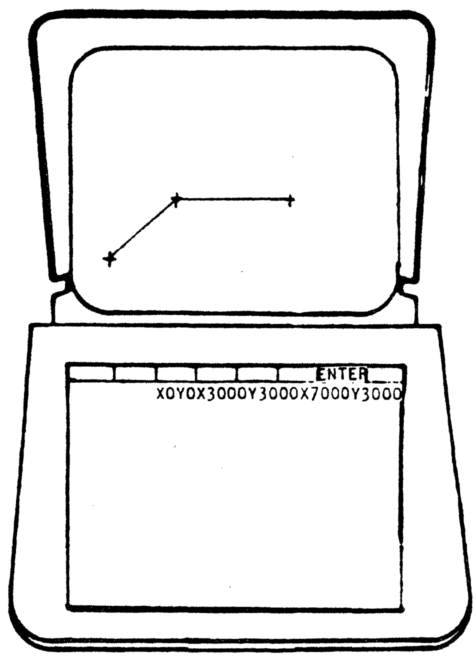
YOUR INPUT (P) (P) (P)

GRAPHICS SCREEN RESPONSE

- displays small crosshairs at picked locations
- draws line segments in red highlight color

FUNCTION SCREEN RESPONSE

- echoes your data input



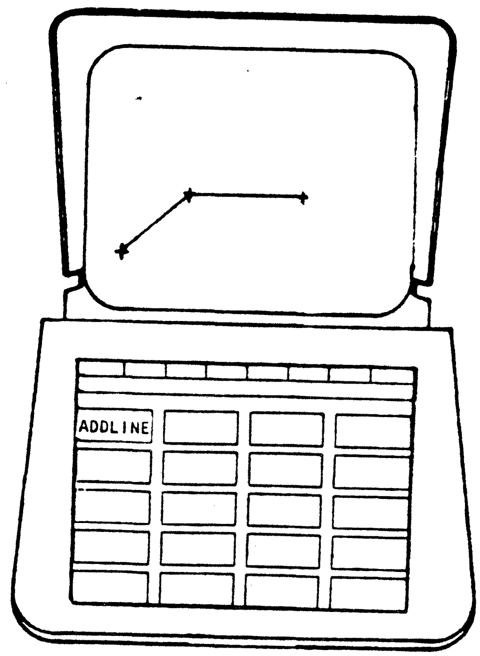
YOUR INPUT | ENTER |

GRAPHICS SCREEN RESPONSE

- line is displayed in the color of the active layer

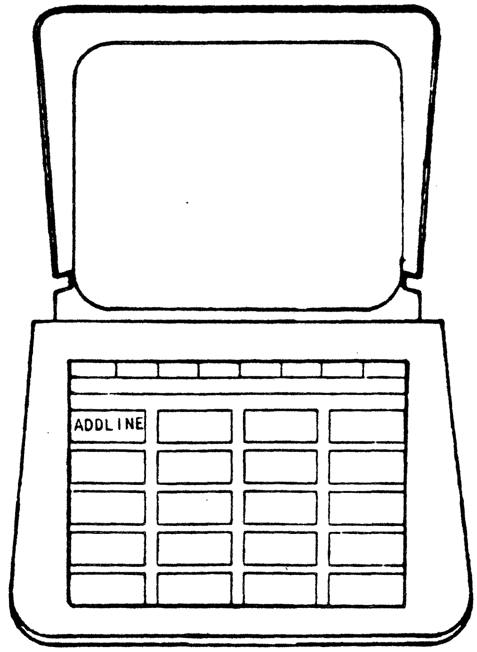
FUNCTION SCREEN RESPONSE

- displays command menu



EXAMPLE OF KEYPAD METHOD

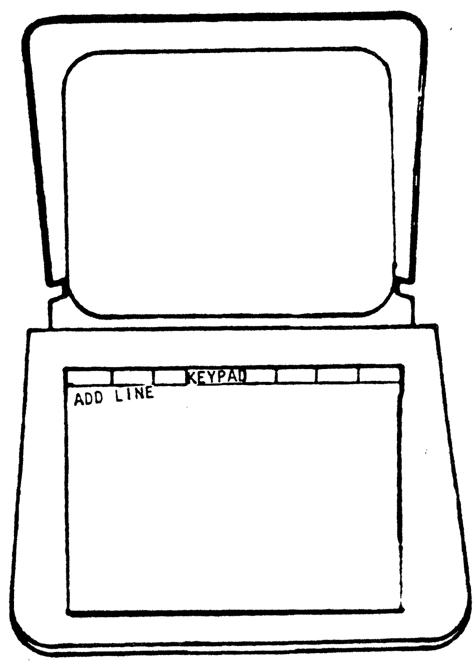
SYSTEM DISPLAY



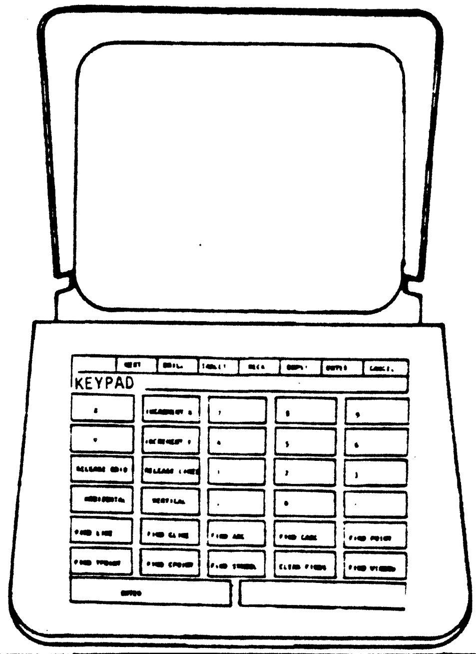
YOUR INPUT |ADD LINE|

FUNCTION SCREEN RESPONSE

- displays tablet
- echoes your command



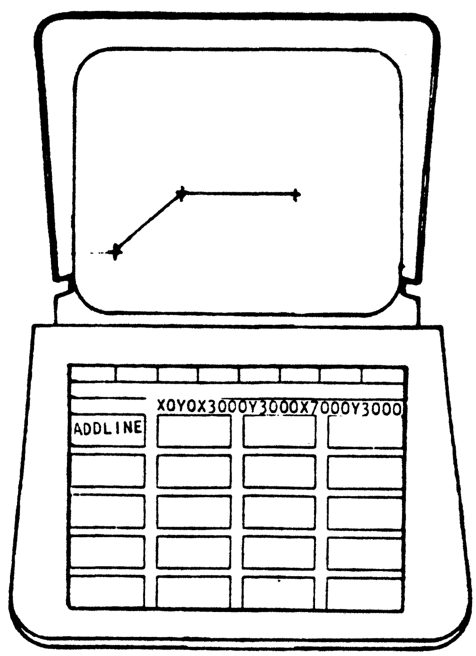
YOUR INPUT |KEYPAD|



FUNCTION SCREEN RESPONSE

- displays keypad

YOUR INPUT |X|0|Y|0|,|X|3|0|0|0|Y|3|0|0|0|,|
|X|7|0|0|0|Y|3|0|0|0| |ENTER|
(use a comma as punctuation between pairs of coordinate values.)



GRAPHICS SCREEN RESPONSE

- displays your line (in red before your |ENTER|; in color of active layer after your |ENTER|

FUNCTION SCREEN RESPONSE

displays command menu
echoes your data input

If you find it more convenient, you may pick |INCREMENT X| or |INCREMENT Y| from the keypad instead of |X| or |Y|:

YOUR INPUT

|X|0|Y|0|,| |INCREMENT X| |3|0|0|0| |INCREMENT Y| |3|0|0|0|,| |INCREMENT X| |4|0|0|0| |ENTER|

When picking the coordinate values of points, you do not need to repeat an x or y value if it has not changed from the last point. For example:

|X|0|Y|0|,|X|3|0|0|0|Y|3|0|0|0|,|X|7|0|0|0|

|Y|3|0|0|0| |ENTER|

is the same as:

|X|0|Y|0|,|X|3|0|0|0|Y|3|0|0|0|,|X|7|0|0|0| |ENTER|

KEYPAD & TABLET: DIFFERENCES

TABLET	KEYPAD
The system places your (P) on the nearest grid point.	The system uses the coordinate point you input, whether it is on a grid point or not.
This is true except when the command you are using tells the system to find an element in the trap. For example, if you pick |ADD CONNECTION|(P), your (P) will be placed on the nearest connect point within the trap, whether the connect point is on a grid point or not.	This is true except when the command you are using tells the system to find an element in the trap. For example, if you pick |ADD CONNECTION| |KEYPAD|X|3|4|5|0| |Y|5|1|2|0|, your connect line will be placed on the nearest connect point to this coordinate point.
You need not be aware of precise coordinate values as you work.	You must know the coordinate values of the area of your drawing in which you are working.

5.2 THE ADD COMMANDS

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

BE CAREFUL!: Before you use the ADD commands, be sure that you have set the active layer to the proper layer number (refer to the layer table in Table GRAPHICS-I).

NOTE: Do not add arcs, circles, or symbols so that they come in contact with the outer extents of your drawing. Doing so will cause them to be distorted.

COMMAND	WHEN TO USE IT
|ADD LINE|	Add lines when drawing symbols and border formats, and when drawing straight lines, that are not connect lines, on your schematic and board drawings.
|ADD ARC|	Add arcs when drawing symbols and border formats, and when drawing arcs, that are not connect arcs, on your schematic and board drawings.
|ADD CIRCLE|	Add circles when drawing symbols and border formats.
|ADD CONNECT PT|	Add connect points when drawing symbols. On board and schematic drawings, you do not add connect points because: 1) they are added with the symbol when you add a symbol. 2) the system adds them automatically when you make a T point by ending or starting one connect line on a segment of another line. 3) the system adds them automatically when you join the end points of two connect lines having different widths. Connect points should be placed on layer 0, so they can be connected to elements on any layer. If you place a connect point on any layer other than 0, you can only connect elements on that layer to it.
|ADD CONNECTION|	Use this to draw connect lines on schematic and board drawings. BE CAREFUL!: On the graphics screen, connect lines look like ordinary straight lines, but the system recognizes the difference. Do not use connect lines for other purposes, for example--in making your drawing format.

COMMAND	WHEN TO USE IT
ADD SYMBOL [NAME] ADD SYMBOL [MENU]	You may place a symbol in your schematic or board drawing in two ways: 1) Pick ADD SYMBOL[NAME], then use the keyboard to input the symbol name. For example, - Pick ADD SYMBOL [NAME] - Function screen display • KEYBOARD - Pick D I O D E ENTER 2) Pick ADD SYMBOL [MENU], then use the symbol menus to input the symbol name. For example, - Pick ADD SYMBOL [MENU] - Function screen display • SYMBOL MENU - Pick DIODE The symbol is the same regardless of the method you choose. SYMBOL FILES When you input the symbol name, the system looks for a symbol file by that name in two places: 1) the current project file 2) the project file named SYSTEM-LIBRARY If the symbol file is not in one of these two places, the system will put an error message on the function screen. (Refer to the Symbol Libraries Section of this manual for further information on symbol files.) LOCATION AND SIZE OF SYMBOLS The system places the symbol's reference point on your P. The reference point is the point that was at 0/0 when the symbol was first drawn. On Telesis-prepared symbols, this is always the lowest left connect point of the symbol. When placed in your schematic or board drawing, the symbol is the same size in mils that it was in the original symbol drawing.

5.3. THE EDIT COMMANDS

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
COPY ELEMENT	To copy any element (except a symbol) in a drawing.
CHANGE LAYER	To reassign elements to new layers of a drawing.
CHANGE LAYER AND WIDTH	To change the layer and width of entire lines and connect-lines in a board drawing.
CHANGE SEGMENT LAYER	To assign connect-line segments to other board layers.
CHANGE WIDTH	To change the width of entire lines and connect-lines in a drawing.
INSERT VERTEX	To insert vertices on lines and connect-lines in in a drawing.
DELETE ELEMENT	To delete any element (except a symbol) from a drawing.
DELETE SEGMENT	To delete a segment of a line or connect line.
DELETE VERTEX	To delete a vertex of a line or connect line.
DELETE SYMBOL	To delete a symbol from a board or schematic drawing.
DEL NET SECTION	To delete a section of a net from a board or schematic drawing.
MOVE ELEMENT	To move any element (except a symbol) in a drawing.
MOVE SECTION	To move a connect-line section in a board drawing.
MOVE SEGMENT	To move a segment of a line or connect line.
MOVE VERTEX	To move a vertex of a line or connect line.
MOVE SYMBOL	To move a symbol in a board or schematic drawing.
SCALE SYMBOL	To reduce or enlarge the size of a symbol in a drawing.
ROTATE SYMBOL	To rotate a symbol in a drawing.

6. ADDING TEXT TO YOUR DRAWING

Symbol, schematic and board drawings all require text labels, such as reference designators, in addition to graphic elements. The sections of the manual on symbol, schematic and board drawings define the labels the system requires. The current section explains the basics of adding and editing labels.

MULTI-LINE TEXT IN A DRAWING

When using the ADD NOTES command, you may use the keyboard RETURN key to create multi-line entries.

EXAMPLE: |ADD NOTES|

INPUT: ALL RESISTORS |RETURN| ARE 1 WATT |RETURN| UNLESS SPECIFIED |ENTER|

RESULT:

△ALL RESISTORS
ARE 1 WATT
UNLESS SPECIFIED

YOU MAY USE THE |SET LINE SPACING| COMMAND TO SPECIFY THE DISTANCE (in mils) BETWEEN LINES OF TEXT.

CAUTION: If you use the multi-line text capability with commands such as |ADD REF DES|, your multi-line entries will apear as one line during |BACK ANNOTATION|. Therefore, it is advisable that you create multi-line entries with the |ADD NOTES| command only!

6.1 TEXT PARAMETERS

You set text parameters in the same way that you set other graphic parameters, by using parameter commands that affect all subsequent input.

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|SET CHAR HT|	To specify the height in mils of text characters.
|SET CHAR WIDTH|	To specify the width in mils of text characters.
|SET CHAR SPC|	To specify the distance in mils between text characters.
|SET CHAR SLANT|	To specify the slant angle of text characters. Ordinarily, you input vertical characters. However, your company drawing format may require slanted letters.
|SET LINE SPACING|	To specify the distance in mils between multi-line text entries when you use the |ADD NOTES| command.
|SET TXT LINE ANG|	To specify the angle of lines of text. Ordinarily, you input horizontal lines of text. However, when you rotate a symbol (a resistor, for example) to a vertical orientation, this command allows you to input a label with the same orientation.
|SET TXT PNT SIZE|	The text point is your label's reference point. It is a Δ displayed at the lower left of your text. You may make it smaller or larger depending on how visible you want it to be on the graphics screen.

6.2 ADDING AND ATTACHING TEXT

These are the commands used for placing text in your drawing.*

ADD DEVICE TYPE	ATTCH DVICE TYPE
ADD REF DESIG	ATTCH REF DESIG
ADD VALUE	ATTCH VALUE
ADD NOTES	ATTCH PIN NUMBER
	ATTCH PIN NAME
	ATTCH SIGNL NAME

THE DIFFERENCE BETWEEN ADDING TEXT AND ATTACHING TEXT

- The ADD commands have this input sequence:

  |ADD REF DESIG| - keyboard input - |ENTER| P_1 |ENTER|

- The ATTCH commands have this input sequence:

  |ATTCH REF DESIG| - keyboard input - |ENTER| P_1 P_2 |ENTER|

- Note that there is a second P in the ATTACH sequence.

- This is what your input means:

  keyboard input -- defines the text of the label.

  P_1 defines the location of the text point.

  P_2 defines the element to which the label is to be attached.

You use the ATTACH commands when the system would not otherwise know what element you were labelling. For example, when you create a symbol drawing, you use the |ADD REF DESIG| command because there is no doubt that you are labelling the symbol itself. But, when you create a schematic drawing, you use the |ATTCH REF DESIG| command because the system would not otherwise know which symbol in the schematic you were labelling.

---

*|ASSIGN REF DESIG| is not included on this list because it is a special command with special requirements used only on board drawings. It is discussed fully in the Board Section of the manual.

WHEN TO USE THE ADD AND ATTACH TEXT COMMANDS

The Command Description Section of the Manual tells you how to use these commands. The table below tells you when to use them.

BE CAREFUL!: Before you add or attach labels, be sure that you have set the active layer to the proper layer number (refer to the layer standards in Table GRAPHICS-I).

COMMAND	WHEN TO USE IT
|ADD REF DESIG|	To place a reference designator label on a symbol drawing.
|ATTCH REF DESIG|	To place a reference designator label on a symbol in a board or schematic drawing. BE CAREFUL not to use the |ATTCH REF DESIG| command on a board drawing when the |ASSIGN REF DES| command is appropriate. (See PCB Section of the manual.)
|ADD DEVICE TYPE|	To place a device type label on a symbol drawing.
|ATTCH DVICE TYPE|	To place a device type label on a symbol in a board or schematic drawing.
|ADD VALUE|	To place a value label on a symbol drawing.
|ATTCH VALUE|	To place a value label on a symbol in a board or schematic drawing.
|ADD NOTES|	To place notes in a drawing.
|ATTCH PIN NUMBER|	To place a number label on a pin in a drawing.
|ATTCH PIN NAME|	To place a name label on a pin in a drawing.
|ATTCH SIGNL NAME|	To place a name label on a signal net in a board or schematic drawing.

6.3. EDITING TEXT IN YOUR DRAWING

You edit text in much the same way that you edit graphic elements.

The Command Description Section of the manual tells you how to use the commands for editing text in a drawing. The table below tells you when to use them.*

COMMAND	WHEN TO USE IT
|UPDATE TEXT|	Use this to change text that was placed in your drawing with the ADD or ATTACH commands. This command changes the text only. The type of label remains the same. For example, you cannot use this command to change a reference designator label to a device type label. When you use |UPDATE TEXT|, the old text will be replaced by the new text that you input. BE CAREFUL! Do not use |UPDATE TEXT| to change text that was placed on a board drawing with the |ASSIGN REF DES| or |CHANGE REF DES| commands.
|MOVE TEXT|	Use this to move text from one location in a drawing to another, just as you use the |MOVE ELEMENT| command. Usually you move text to make your drawing neater, or to conform to a standard. BE CAREFUL! Moving text does not alter its relationship to the element it labelled. For example, do not move a reference designator from one symbol to another. If you do, the system will still regard it as the label for the first symbol regardless of what the graphics screen shows.
|DELETE TEXT|	Use this command to delete text that was placed in a drawing with the ADD or ATTACH commands. BE CAREFUL! Do not use the |DELETE TEXT| command to delete text that was placed on a board drawing with the |ASSIGN REF DES| or |CHANGE REF DES| commands.

* The |CHANGE REF DES| and |DEASSIGN REF DES| commands are discussed in the PCB Section of the manual.

7. AIDS IN THE GRAPHIC PROCESS

There are four groups of commands whose function is to aid you as you add and edit the elements and text in your drawing:

1. COMMANDS THAT ALTER YOUR VIEW OF A DRAWING

   |ZOOM RATIO|
   |WINDOW|
   |DRAWING|
   |CENTER|

2. COMMANDS THAT PROVIDE YOU WITH INFORMATION

   |LIST ELEMENT|
   |STATUS|

3. COMMANDS THAT LET YOU SET OR RELEASE A PARAMETER FOR ONE P ONLY

   |HORIZONTAL|
   |VERTICAL|
   |RELEASE GRIP|
   |RELEASE LINEL|

4. COMMANDS THAT HELP YOU PICK PRECISELY THE ELEMENT YOU WANT

   |FIND LIN/CLIN|
   |FIND ARC|
   |FIND TPOINT|
   |FIND CPOINT|
   |FIND SYMBOL|
   |CLEAR FINDS|

7.1 COMMANDS THAT ALTER YOUR VIEW OF A DRAWING

These commands are found on the ADD/EDIT, TEXT, and other appropriate menus. Most of them are in the top row for convenience because you will use them frequently.

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|ZOOM RATIO|	To zoom into a small part of your drawing, or to zoom out. Use this, or |WINDOW|, when you need to see small details on a part of your drawing, for example--when you are using the |ADD CONNECTION| command. You should reset the grid size of your drawing whenever you zoom in or out.
|WINDOW|	This is a short-cut way of zooming into a small part of your drawing.
|DRAWING|	This is a short-cut way of zooming all the way out to view your entire drawing.
|CENTER|	Use this command to move the center of your drawing laterally on the graphics screen. Usually, you use this when you are zoomed in, and want to see an area of the drawing adjacent to the area currently shown on the screen.

7.2 COMMANDS THAT PROVIDE YOU WITH INFORMATION

As you create a drawing, you frequently need information that is not shown on the graphics screen, such as:

- Current parameter settings
- Active layer
- Which layers have been blanked
- Coordinate point values of an element
- Layer number of an element

You use the |STATUS| and |LIST ELEMENT| commands to display this kind of information on the function screen.

The |STATUS| command is found in the top row of most menus. The |LIST ELEMENT| command is found on the PARAMETER MENUS and on other appropriate menus.

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|STATUS|	When you pick |STATUS|, the function screen displays a list of the current parameter settings, whether you set them yourself or are using default parameters. It also displays the name of your current drawing and project file.
|LIST ELEMENT|	Pick |LIST ELEMENT| when you need information about an element in your drawing. For example--the layer of the element and the coordinate point values of its location.

When LIST ELEMENT is selected, the following information about the specified element is listed on the screen :

- Type of element
- Layer that the element is located on
- The X and Y coordinates of the element
- Net numbers attached to element (if applicable)
- ID Label of the element
- The angle the element is rotated (if any).

ID Labels are assigned to text labels within an element to specify the type of text label that they represent. ID Labels and their corresponding text label are listed below.

TEXT LABEL	ID LABEL
SIGNAL NAME	1
REF DESIGNATOR	2
PIN NUMBER	4
DEVICE TYPE	5
PIN NAME	14
PIN USE CODE	15
VALUE	18
FUNCTION TYPE	22

The following examples illustrate the information listed on the screen when LIST ELEMENT is selected.

```
                    Element Type - Symbol

    Symbol Name  DIP14                     Rev
    Location  3200 5400                    Rotate Angle  0
    Symbol Scale  1.00

       Text Points

    Label        Value

      4            P
      5           7400
      2           Z**
```

```
                    Element Type - Arc

    Center  3100 3800                      Layer  1
    Radius  400                            Width  0
                                      Width Justification  Center
    Short Angle  0.000                     Dashpattern  1
    End Angle  0.000                       Tag  1
```

7.3 COMMANDS THAT LET YOU SET OR RELEASE A PARAMETER FOR ONE (P) ONLY

When you set a parameter on the PARAMETER MENU, it remains at that setting throughout all subsequent commands until you reset it.

The |HORIZONTAL|, |VERTICAL|, |RELEASE GRID|, and |RELEASE LINEL| commands allow you to set or release a parameter for one pick only. These commands are found on the KEYPAD.

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|HORIZONTAL|	Use this when you have not set the |LINE LOCK 90/ON| parameter, but want to be sure that one segment of your line is placed horizontally.
|VERTICAL|	Use this when you have not set the |LINE LOCK 90/ON| parameter, but want to be sure that one segment of your line is placed vertically.
|RELEASE GRID|	Ordinarily, your tablet (P) s are snapped to the nearest grid point. Use |RELEASE GRID| when you want to (P) one point that is not on a grid point.
|RELEASE LINEL|	RELEASE LINE LOCK: Use this to override the line lock parameter for one segment of a line.

7.4 COMMANDS THAT HELP YOU PICK PRECISELY THE ELEMENT YOU WANT

When you pick commands (like |ADD CONNECTION| or |DELETE ELEMENT|) that require the system to find an element in your drawing, the system searches for the element in a trap area around your tablet or keypad P. (See Parameter Section for more information on the trap.)

On a detailed drawing, there will frequently be more than one element within the trap area. The FIND commands allow you to specify the type of element you want the system to find in the trap area.

These commands are located on the KEYPAD.

You may use more than one FIND command at a time (up to 4). For example, if you wanted to delete both connect lines and symbols (but nothing else), you would pick:

|DELETE ELEMENT| |KEYPAD| |FIND LIN/CLIN| |FIND SYMBOL| P P .... |ENTER|

The FIND commands remain in force until you end the original command with |ENTER|, or until you use the |CLEAR FINDS| command.

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

These commands are only used after picking a command (like |DELETE ELEMENT|) that require the system to find an element.

COMMAND	WHEN TO USE IT
|FIND LIN/CLIN|	To instruct the system to find only lines or connect lines.
|FIND ARC|	To instruct the system to find only arcs.

COMMAND	WHEN TO USE IT
|FIND TPOINT|	To instruct the system to find only text points.
|FIND CPOINT|	To instruct the system to find only connect points.
|FIND SYMBOL|	To instruct the system to find only symbols.
|CLEAR FINDS|	Use this to retract all FIND commands in force. After picking |CLEAR FINDS|, you may pick new FIND commands, for example: |DELETE ELEMENT| |FIND CLINE| (P) (P) P |CLEAR FINDS| |FIND SYMBOL| (P) (P) (P) |ENTER|

8. CLOSING A DRAWING FILE

WHEN TO CLOSE A DRAWING FILE

Close a drawing file when:

1. You have completed the drawing.

2. You have finished a work session on the drawing.

3. You have completed a significant part of a drawing and want to be sure it is saved intact.

4. You want to leave the drawing to work on something else, for example--a text file, or file management.

or 5. You want to delete the active drawing.

COMMANDS USED TO CLOSE A DRAWING

There are three commands used for closing a drawing file:

|SAVE DRW SAME RV|

|SAVE DRW NEW REV|

|CANCEL ACTV DRW|

These are located on the TASK MENU of each drawing menu set. You get to the TASK MENU, by picking |DONE| on any one of the menus in the menu set.

The Command Description Section of the manual tells you how to use these commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|SAVE DRW SAME RV| |SAVE DRW NEW RV|	IF YOU OPENED WITH THE |OLD DRAWING| COMMAND: If you opened an existing drawing file, you either named the revision you wanted, or (by default) opened the most recently used revision of the drawing. In either case, the system placed a copy of this revision on the graphics screen. Your additions and edits were made on this copy. The system also retained the original of this revision. The original was unaltered by your work on the copy. When you close your drawing, you have two choices: 1) You may save both the copy with your new work on it, and the unaltered original, by using |SAVE DRW NEW REV| followed by a new revision label. Do this when you believe you might want to refer to the original version again. 2) You may save the copy with your new work on it, and delete the original by using |SAVE DRW SAME RV|. In this case, the copy replaces the original and uses the same revision label. Do this when you are sure that you have no further use for the original version. IF YOU OPENED WITH THE |NEW DRAWING| COMMAND: Use |SAME DRW SAME RV| to close the drawing under the name and revision label you specified when you opened it. Use |SAME DRW NEW RV| to close it if, for some reason, you want to change the revision label you specified when you opened it.

COMMAND	WHEN TO USE IT
CANCEL ACTV DRW	Use this if you decide not to save any of the work you've done since you opened the drawing. IF YOU OPENED WITH THE OLD DRAWING COMMAND: CANCEL ACTV DRW does not delete the file you opened, only the copy on which you were working. IF YOU OPENED WITH THE NEW DRAWING COMMAND: CANCEL ACTV DRW deletes the drawing file altogether.

CREATING AND EDITING TEXT FILES

1. GENERAL INFORMATION
2. THE KEYBOARD AND GRAPHICS SCREEN
3. OPENING A TEXT FILE
4. INPUTTING TEXT
5. MOVING THE CURSOR
6. EDITING TEXT
7. CLOSING A TEXT FILE

1. GENERAL INFORMATION

When you use the Telesis system to design a printed circuit board, you must create some text files in addition to the graphics for your symbols, schematic and board. For example, when you create a library of component symbols, you will input pin description text files (specifying pad sizes and layers) as well as symbol graphics. You use the text file keyboard to create these files.

In addition to the text files you input, the system itself writes text files when you pick certain commands. For example, when you pick |CREATE BOM|, the system uses information in the net data base to create a text file called BOM-REPORT containing a bill of materials.

File management for text files is the same as it is for drawing files. Each text file is part of a larger project file. Text files may be copied, deleted, or stored on disks or tape. (See File Management and Archives Section of the manual.) Text files may also be printed. (See Printer Section of the manual.) You can display a listing of the text files in any project file by using the INDEX commands.

The requirements for the text files you must input, and the contents of the text files created by the system, are discussed in the sections of the manual covering symbol libraries, schematic and board drawings, and artwork. This section instructs you in the basic skills needed to use the text file keyboard.

2. THE KEYBOARD AND GRAPHICS SCREEN

With text files as with drawings, you use the function screen to input information, and the graphics screen to view it.

GETTING TO THE TEXT FILE KEYBOARD

After you open a project file, the MENU OF MENUS is displayed. To get to the text file keyboard:

- Pick |TEXT FILES|
- Function screen display
  - TEXT FILE TASK MENU
- Pick |CREATE/EDIT TEXT|
- Function screen display
  - TEXT FILE KEYBOARD

  Graphics screen display
  - TEXT FILE FORMAT

THE GRAPHICS SCREEN FORMAT

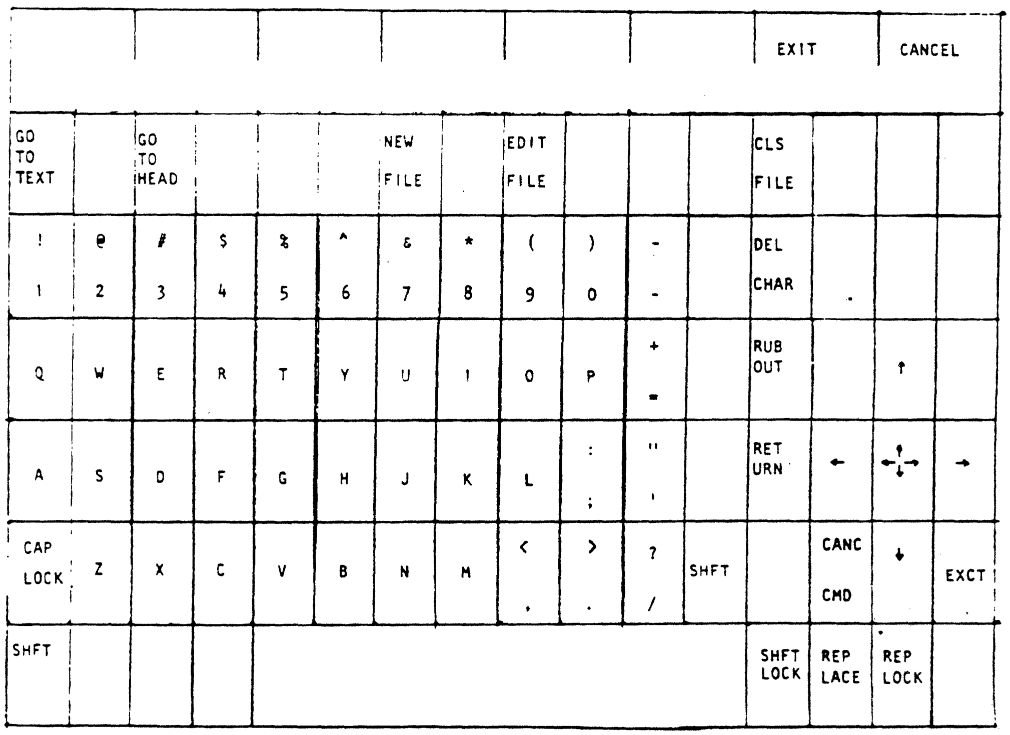
When you first get to the text file keyboard, the format on the graphics screen looks like this:

```
                                                              CURSOR
CMND: |_|
 FILE STATUS:  CLOSED                    REV:                 HEADER
 ECHO:                                                        AREA


++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++


                                                              TEXT
                                                              AREA
```

The HEADER AREA is used for opening and closing files, and always remains displayed at the top of the screen.

The TEXT AREA is where your text input is displayed.

THE KEYBOARD



TO LEAVE THE TEXT FILE KEYBOARD

Use the |EXIT| or |CANCEL| command in the top row of the keyboard to flip back to the TEXT FILE TASK MENU. These commands are described in the table below.

COMMAND	DESCRIPTION
|EXIT|	Use this command to flip back to the TEXT FILE TASK MENU. If you have an open text file, the system closes it when you use this command. |EXIT| must be followed by the |EXCT| command: |EXIT| |EXCT|
|CANCEL|	Use this command to flip back to the TEXT FILE TASK MENU. BE CAREFUL!: Do not use |CANCEL| when you have an open text file.

COMMANDS USED TO INPUT OR CANCEL OTHER COMMANDS

COMMAND	DESCRIPTION
|EXCT|	EXECUTE: This command is similar to the |ENTER| command on the drawing menus. It indicates that you have finished your input. For example, to open a new text file named DIODE.PIN, your input would be: |NEW FILE| |D|I|O|D|E|.|P|I|N| |EXCT|
|CANC CMD|	CANCEL COMMAND: This command is similar to the |CANCEL| command on the drawing menus. It cancels the last input command (if you have not yet picked |EXCT|).

3. OPENING A TEXT FILE

These are the commands used to open a text file:

|NEW FILE| - to create a new file

|EDIT FILE| - to open an existing file

After you pick |NEW FILE| or |EDIT FILE|, you input the name of the file using the keyboard. Each time you input a character, the letter is displayed at the cursor location on the graphics screen, and the cursor moves one space to the right.

COMMAND	DESCRIPTION
|NEW FILE|	PURPOSE: To create and name a new text file. INPUT SEQUENCE: |NEW FILE| - keyboard input - |EXCT| PARAMETERS: ● KEYBOARD INPUT defines the text file name and revision label. ● Revision label is optional. If you do not input one, the system will assign revision label: 1. ● Text file names may be up to 18 alphanumeric characters. Do not use blank spaces in the name. ● Revision label may be up to 4 alphanumeric characters. Do not use blank spaces in the label. ● Revision label follows the name, and must be preceded by a blank space. EXAMPLE: |NEW FILE| |D|I|O|D|E|.|P|I|N|_|A| |EXCT| BE CAREFUL!: When you name a new text file, DO NOT USE any non-alpha characters that do not appear on the graphics keyboard. Otherwise, you may create a file that you cannot refer to using the graphics keyboard.

COMMAND	DESCRIPTION
|EDIT FILE|	PURPOSE: To open an existing text file. INPUT SEQUENCE: |EDIT FILE| - keyboard input - |EXCT| PARAMETERS: ● KEYBOARD INPUT defines the text file name and revision label. It must be the exact file name. ● You are not required to input the revision label. If you do not, the system displays the most recently used revision. EXAMPLE: |EDIT FILE| |B|O|M|-|R|E|P|O|R|T| |EXCT| NOTE: The system displays the text file on the graphics screen. You may add to it and edit it. You may use the |EDIT FILE| command to display a system-created text file in the same way that you use it to display one created by you. But keep in mind that if you edit a system-created file, you do not change the net data base on which the text file was based.

INFORMATION IN THE HEADER AREA

As you input a text file name, it is displayed on the "CMND:" line of the header format. Go slowly when inputting characters in the header area, and check the name displayed to see if it is correct.

When you pick |EXCT|, there is an interval of several seconds. After this, the name and revision label of your file are displayed on the "FILE STATUS:" line. At the same time, the system moves the cursor from the header to the upper left corner of the text area.

If the system cannot open your file for some reason, an error message is displayed on the "ECHO:" line. For example, the system will display: "FILE NOT FOUND", if you pick |EDIT FILE| followed by the name of a text file not in the current project file.

If the system will not open your file, check to see that you have used the correct file name. You can check the names of the text files in the current project file by picking |EXIT| |EXCT| to flip to the TEXT FILES TASK MENU, and then picking the |CURRENT INDEX| command.

If you make a mistake while inputting a file name (before you pick |EXCT|), you may use the editing commands described below to correct it. You may also use the |SHFT| |SHFT LOCK| and |CAP LOCK| commands in the header area. (Some keyboard commands may only be used in the text area — see below.)

4. INPUTTING TEXT

You use the keyboard to input characters in the same way that you would use a typewriter or word processor: there are keyboard boxes for characters, spaces, shift and return.

KEYBOARD BOXES	DESCRIPTION
characters	The characters available to you are the ones shown on the keyboard. They are displayed on the graphics screen in upper case unless you use the |SHFT|, |SHFT LOCK|, or |CAP LOCK| boxes.
|CAP LOCK|	CAPITAL LOCK: Use this to switch from lower case to upper case and vice versa. It affects letters only.
|SHFT LOCK|	SHIFT LOCK: This is the same as |CAP LOCK| except that it affects the numeral/symbol keys as well as letters.
|SHFT|	SHIFT: This is the same as |SHFT LOCK| except that it is for one character only.
|RETURN|	Use this to end a line, and to move the cursor to the leftmost position of the line below. The symbol ^{C}r will be displayed at the end of the upper line. |RETURN| cannot be used in the header.
space bar	This is the large, empty box in the bottom row of the keyboard. Use it to input blank spaces.

The cursor moves to the right as you input characters or spaces. If it reaches the right extreme of the graphics screen, it automatically moves down to the next line even if you do not use |RETURN|.

When the cursor reaches the lower extreme of the graphics screen, you may use |RETURN| and continue to input lines of text. As you input a new line, the system moves the display of your text file up one line. Your new line will be visible at the bottom of the text area, but your top line will no longer be visible. (Use the |↑| box -- described below -- to return the top line to the screen.

5. MOVING THE CURSOR

Before you can edit text, you must move the cursor to the character you want to edit.

You may also move the cursor from the text area to the header and vice versa.

Use these keyboard boxes to move the cursor:

KEYBOARD BOXES	DESCRIPTION
[→]	This moves the cursor one character to the right.
[←]	This moves the cursor one character to the left.
[↑]	This moves the cursor one character up. [↑] cannot be used in the header.
[↓]	This moves the cursor down one character. [↓] cannot be used in the header.
[↕↔]	Use this followed by [->] or [<-] to move the cursor to the extreme right or left of a line. Use this followed by [↑] or [↓] to move the cursor to the upper or lower extreme of the file. If the upper or lower extreme is not visible, picking these will display it. This cannot be used in the header.

KEYBOARD BOXES	DESCRIPTION
|GO TO HEAD|	Use this command to move the cursor to the header area from the text area. The cursor will move to the right of the last input character on the "CMND:" line. Ordinarily, you pick |GO TO HEAD| only when you want to close your file.
|GO TO TEXT|	Use this command to move the cursor to the text area from the header area. The cursor will move to the right of the last input character in the text area. Ordinarily, you pick |GO TO TEXT| only when you have picked |GO TO HEAD|, and have changed your mind about closing the file. (When you open a file, the |GO TO TEXT| command is not needed because the system automatically moves the cursor from the header to the text area.)

6. EDITING TEXT

To correct errors or insert text, move the cursor to the position you want to edit. The cursor can only be moved over characters, blank spaces, or ^{C}r's already input. It cannot be moved into areas of the graphics screen where no text has been input.

TO INSERT TEXT

Position the cursor one character to the right of the point where you want to insert text. Then, type in text as usual. The character under the cursor and the characters to the right will all be shifted to the right to accommodate your insertion.

TO CHANGE EXISTING TEXT

Use these keyboard boxes to change existing text:

KEYBOARD BOXES	DESCRIPTION
|DEL CHAR|	To delete the character, space, or ^{C}r under the cursor.
|RUBOUT|	To delete the character, space, or ^{C}r to the left of the cursor.
|REPLACE|	To replace the character under the cursor with a new one. EXAMPLE: - Pick |REPLACE| |L| (cursor) Before: H E L |K|O After: H E L L |O| |REPLACE| cannot be used in the header.
|REP LOCK|	REPLACE LOCK: To replace a series of characters with new ones. Pick |REPLACE| to undo the |REP LOCK| setting. |REP LOCK| cannot be used in the header.

7. CLOSING A TEXT FILE

When you have finished working on a text file, you may close it in one of 3 ways:

1. By picking |EXIT| |EXCT|
2. By picking |CLS FILE| |EXCT|
3. By using the |CLS FILE| command followed by the name of the file.

1. |EXIT| |EXCT|

WITH NEW TEXT FILES: If you opened your file with the |NEW FILE| command, picking |EXIT| |EXCT| will close the file with the name and revision label you specified when you opened it.

WITH OLD TEXT FILES: If you opened your file with the |EDIT FILE| command, picking |EXIT| |EXCT| will close the file with the name and revision label you specified when you opened it. The original version of the text file with this name and revision label will be deleted.

BE CAREFUL!: If you made changes to a text file, and you want to keep both the changed version and the original version, you must use the |CLS FILE| command followed by a file name.

2. |CLS FILE| |EXCT|

Picking |CLS FILE| |EXCT| has the same effect as |EXIT| |EXCT|.

3. |CLS FILE| FOLLOWED BY A FILE NAME

If you want to save both the original version of the text file you opened and the version that you have changed, you must change the name and/or revision label of the file when closing it.

To do this, you pick |GO TO HEAD| |CLS FILE| (the cursor will move to the right of the old file name), then input a name and/or revision label different from the old one.

EXAMPLE: If you opened the text file DIODE.PIN revision 1, and you wish to save both the original version and the one now displayed on the graphics screen, pick:

|GO TO HEAD| |CLS FILE| |D|I|O|D|E|.|P|I|N|_|2|EXCT|

You will now have these two text files in your current project file: DIODE.PIN revision 1
DIODE.PIN revision 2

If you wish to change only the revision label, you may use the |<-| and |->| commands instead of repeating the file name.

EXAMPLE: If you opened a file named DIODE.PIN revision 1, and want to close the file as DIODE.PIN revision 2:

- Pick |GO TO HEAD|

- Graphics screen display (cursor)
  - CMND: DIODE.PIN |_|
    FILE STATUS: DIODE.PIN REV: 1
    ECHO:

- Pick |<-| |<-| |<-| |<-| |<-| |<-| |<-| |<-| |<-|

- Graphics screen display (cursor)
  - CMND: |D|IODE.PIN
    FILE STATUS: DIODE.PIN REV: 1
    ECHO:

- Pick |CLS FILE| |->| |->| |->| |->| |->| |->| |->| |->| |->| |_| |2| |EXCT|

- Graphics screen display (cursor)
  - CMND: DIODE.PIN 2|_|
    FILE STATUS: CLOSED REV:
    ECHO:

TELESIS KEYBOARD TEXT EDITOR

1. THE TELESIS KEYBOARD TEXT EDITOR

GENERAL INFORMATION

The Telesis Keyboard Editor allows you to create and edit text files with the DIGITAL VT101* video terminal. This in an optional input and output device that communicates with the Telesis system. You may use the VT101 keyboard to input text file entries and to display and edit files previously created with the Telesis function screen editor.

The Telesis Keyboard Editor is an ideal tool that can increase your speed when working with long text files. A variety of commands allow you to instantly transfer, change or delete any line or section of your file. You may also instantly flip from a character editing mode to a line editing mode.

This section of the manual explains how to use the Telesis Keyboard Editor when creating or editing text files.

CREATING TEXT FILES WITH THE TELESIS KEYBOARD EDITOR

If you are creating a text file with the Telesis Keyboard Editor, you must enter a design project on the Telesis system where the file is to exist. If you wish to edit an existing file, simply enter the design project to locate the file.

The Telesis Keyboard Editor allows you to save your new or edited file in your particular design project. You may recall the file at a later time on the VT101 terminal, or you may view and edit the file on the Telesis graphics screen with the function screen editor.

The following section will show you how to edit an existing file with Telesis Keyboard Editor.

DIGITAL and VT101 are registered trademarks of Digital Equipment Corporation.

2. STARTING YOUR EDITING SESSION

After entering a design project to create a text file, or to locate a file you wish to edit, you may use the following steps to display the new or existing file on the VT101 terminal.

FROM THE TELESIS SYSTEM

STEP 1 Pick |KEY EDIT TEXT|

The message line on the function screen issues the prompt:

TYPE IN "HEL KEYEDIT/TEXT" AT THE TERMINAL

FROM THE TELESIS KEYBOARD EDITOR

STEP 2 From the Telesis Keyboard Editor, input the identical line that was displayed on the function screen:

Video display-->

```
>HEL KEYEDIT/TEXT
```

The video terminal displays the following message:

```
 HEL KEYEDIT/TEXT
"enter name of text file to be edited:"
```

STEP 3 Input the text file name.

EXAMPLE:

```
 HEL KEYEDIT/TEXT
"enter name of text file to be edited:" LAYERSTD
```

STEP 4   The video terminal displays the following information:

```
 HEL  KEYEDIT/TEXT
"enter name of text file to be edited:" LAYERSTD
"you will be editing an existing text file named LAYERSTD rev 3"
"enter RETURN to continue:"
```

When you input the file name LAYERSTD, the system locates the latest version of the file. In this example, rev 3 was the latest version.

If you do not wish to edit the latest version of the file, you must input the file name with the revision label.

EXAMPLE: LAYERSTD 2

If the file name that you input does not exist in the current index, the system will respond with:

TEXT FILE (NAME OF THE FILE) DOES NOT EXIST.
DO YOU WANT TO CREATE THIS FILE (Y OR N):

If you respond with a "Y" the system will create a new file in the current index.

If you respond with a "N" the system will not create a new file and will respond with:

DO YOU WANT TO EDIT ANOTHER FILE (Y OR N):

STEP 5 |RETURN|

Depress the |RETURN| key on the keyboard to display the top lines of the LAYERSTD file.

```
 HEL  KEYEDIT/TEXT
"enter name of text file to be edited:" LAYERSTD
"you will be editing an existing text file named LAYERSTD   rev 3"
"enter RETURN to continue:"

               LAYERSTD  3

(LAYERSTD)
(TELESIS STANDARD LIBRARY)
(PHYSICAL LAYER COMPONENT SIDE)
DBLAYER 1 COMPONENT-SIDE
(PHYSICAL LAYER SOLDER SIDE)
DBLAYER 2 SOLDER-SIDE
```

The file is displayed on the video terminal in the "character-edit" mode. The following section describes character editing with the VT101 keyboard and keypad.

3. THE CHARACTER EDITING DISPLAY

Character mode editing allows you to view, change, delete and add text to your file. In this mode, you may use the keyboard and keypad functions to create and edit text. You have the capability to position the cursor anywhere in the file during editing.

EXAMPLE: CHARACTER EDIT MODE

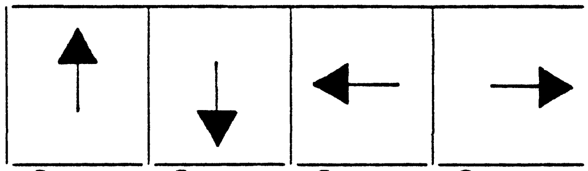
```
Flashing
Cursor
 ------>   |_|
                          LAYERSTD    3

           (LAYERSTD)
           (TELESIS STANDARD LIBRARY)
           (PHYSICAL LAYER COMPONENT SIDE)
           DBLAYER 1 COMPONENT-SIDE
           (PHYSICAL LAYER SOLDER SIDE)
           DBLAYER 2 SOLDER-SIDE
           (PHYSICAL LAYER INTERNAL SIGNAL)
           DBLAYER 3 INTERNAL-SIGNAL
           (PHYSICAL LAYER INTERNAL SIGNAL)
```

In the example, the beginning lines of the file are displayed over the entire area of the screen. Because the file is longer than the video display area, you may move the cursor down to scroll the file up. The remaining lines of the file begin to move up to the display area, while the top lines scroll up and out of view.

4. POSITIONING THE CURSOR DURING CHARACTER EDITING

During character editing, use the arrow keys to position the cursor to the line you wish to edit. The arrow keys are located at the top of the keyboard.

ARROW KEYS



Cursor Up	Cursor Down	Cursor Left	Cursor Right

Cursor Up - Moves the cursor up one line. If the line contains no characters, the cursor will be positioned at the start of the line.

Cursor Down - Moves the cursor down one line. If the line contains no characters, the cursor will be positioned at the start of the line.

Cursor Left - Moves the cursor to the left. When the cursor is moved beyond the left margin, it will back-up to the last character on the previous line.

Cursor Right - Moves the cursor to the right. If there are no additional characters on that line, the cursor will advance to the first character on the next line.

The following keys will move the cursor to a new position on a line:

BACKSPACE - Positions the cursor to the beginning of the current line regardless of its position on the line. The cursor stays in the left margin and backs-up one line each time you depress BACKSPACE.

DELETE - Deletes the character to the left of the cursor and repositions the cursor one space to the left.

LINE FEED - Deletes the characters to the left of the cursor to the beginning of a word. The character positioned at the cursor is not deleted.

After moving the cursor to the position on your file where you wish to edit, you may use the standard keyboard characters to input or update lines of text.

However, you may need to open up areas of your file to insert text between existing lines. Use the arrow keys to position the cursor to the line below the position of the insertion. When you depress the RETURN key, a new (blank) line will open up. The top lines scroll up, with lines positioned at the cursor scrolling down.

EXAMPLE: Using the RETURN key to open lines within a file.

```
                              LAYERSTD    3

                (LAYERSTD)
Flashing        (TELESIS STANDARD LIBRARY)
Cursor          (PHYSICAL LAYER COMPONENT SIDE)
-------->     [_]DBLAYER 1 COMPONENT-SIDE
                (PHYSICAL LAYER SOLDER SIDE)
                DBLAYER 2 SOLDER-SIDE
                (PHYSICAL LAYER INTERNAL SIGNAL)
                DBLAYER 3 INTERNAL-SIGNAL
                (PHYSICAL LAYER INTERNAL SIGNAL)
```

Depress the RETURN key (two times).

RESULT: Two blank lines are opened on the file.

```
                              LAYERSTD    3

                (LAYERSTD)
                (TELESIS STANDARD LIBRARY)
                (PHYSICAL LAYER COMPONENT SIDE)
Flashing
Cursor
------->
                DBLAYER 1 COMPONENT-SIDE
                (PHYSICAL LAYER SOLDER SIDE)
                DBLAYER 2 SOLDER-SIDE
                (PHYSICAL LAYER INTERNAL SIGNAL)
                DBLAYER 3 INTERNAL-SIGNAL
                (PHYSICAL LAYER INTERNAL SIGNAL)
```

5. KEYPAD EDITING IN THE CHARACTER EDIT MODE

The keypad is located to the right of the keyboard. You may use the keypad functions to increase your editing speed when working on a file.

The keypad can assist you in the following ways:

* Moves the cursor to the top or bottom of the file.
* Deletes entire lines, single words and characters.
* Undeletes lines, characters and words on the file.
* Opens up lines within the file.
* Allows you to "cut and paste" sections of the file.
* Allows you to specify the direction of cursor movement with the keypad functions.

KEYPAD

<table>
<tr><td>PF1</td><td>PF2</td><td>PF3</td><td>PF4</td></tr>
<tr><td>7</td><td>8</td><td>9</td><td>—</td></tr>
<tr><td>4</td><td>5</td><td>6</td><td>,</td></tr>
<tr><td>1</td><td>2</td><td>3</td><td rowspan="2">ENTER</td></tr>
<tr><td colspan="2">0</td><td>.</td></tr>
</table>

The following section defines the keypad functions that you may use when editing a text file.

5.1 STANDARD KEYPAD EDITING

When editing in the character mode, use the arrow keys to position the cursor to the lines or characters you wish to change in your text file. During keypad editing, you see the changes that occur in the file as you make them.

KEYPAD FUNCTION	RESULT
KEY #1	Moves the cursor to the next word on the line. The cursor will be positioned at the first character.
KEY #2	Moves the cursor to the end of a line.
KEY #3	Moves the cursor to the next character.
KEY #4	Specifies the direction for cursor movement with the keypad functions. This is the ADVANCE key.
KEY #5	Specifies a BACKUP direction for cursor movement with the keypad functions.
KEY #6	Allows you to delete and store a string of characters. This key is used in conjunction with other keypad functions. (See ALTERNATE KEYPAD EDITING)
KEY #7	Moves the cursor to the top or bottom of your file. KEY #4 and KEY #5 define cursor direction.
KEY #8	Moves the cursor up or down one 16-line section. KEY #4 and KEY #5 define cursor direction.
KEY #9	(See HELP, BASICS-102)
KEY #0	Functions as a carriage return.

ADDITIONAL KEYPAD EDITING

You may use additional keypad functions to increase your editing versatility. For example, you may use the PF1 key in conjunction with other keys to perform alternate functions.

KEYPAD FUNCTION	RESULT
,	Deletes the character positioned at the cursor.
—	Deletes the word positioned at the cursor.
PF1	Use this key in conjunction with other functions. (See ALTERNATE KEYPAD EDITING)
PF2	This is a HELP command file for the Telesis Keyboard Editor. (See BASICS-102)
PF3	Allows you to find specific words or strings of words in the file. (See ALTERNATE KEYPAD EDITING)
PF4	Deletes the current line of text to the right of the cursor.
.	Allows you to select a string of characters to delete, store, and move to other text file locations (See ALTERNATE KEYPAD EDITING)

5.2 ALTERNATE KEYPAD EDITING

Alternate keypad editing allows you to use the PF1 key in conjunction with other keys on the keypad.

The following table describes each editing function when you use the PF1 key.

PF1 ---> PF3

Allows you to search for a string of characters in your file. The system prompts you for input of the string name. You may use KEYS 4 and 5 to define the direction of your search.

Use the PF3 key to continue the search for additional strings of the same name.

PF1 ---> PF4

Allows you to undelete the line previously deleted with the PF4 key. You may delete a line, reposition the cursor to another location on the file, then undelete the line.

PF1 ---> KEY 1

Allows you to invert the case of alphabetic characters. The character at the cursor position is changed from upper to lower case, or the reverse.

PF1 ---> KEY 2

Allows you to delete an entire or portion of a line from the current cursor position to the end of the line. Contents of the file below the deleted line will not fill the open file.

PF1 ---> KEY 4

Allows you to advance from the current position on the file to the end.

PF1 ---> KEY 5

Allows you to back-up from the current position on the file to the top.

PF1 ---> KEY 7

Allows you to quit or exit from the text file. When the system prompts you for a "command", input EXIT or QUIT.

EXIT - Saves the text file in the current project under the same revision label that was named when the file was opened.

QUIT - Cancels the current file displayed on the video terminal without cancelling the original version that was named when the file was opened.

Use the ENTER key to execute the EXIT or QUIT commands.

PF1 ---> KEY 0

Allows you to open up lines on the text file. This keypad function operates identically to the keyboard RETURN key.

PF1 ---> ,

Allows you to reinsert the last character that was deleted with the |,| key. This function allows you to replace a character that was accidentally deleted.

PF1 ---> —

Allows you to reinsert the last word deleted from the file with the |-| . You may delete a word, move the cursor to another location on your file, then undelete the word.

5.3 SELECTING AND MOVING TEXT WITH ALTERNATE KEYPAD EDITING

The following input sequence allows you to select a line or section of text that may be deleted, stored, and moved, to another location on the file.

```
 _____
|  .  |       Select text                     Move cursor
|_____| --->  to be moved  ---> KEY #6 --->  to new location ---> PF1 ---> KEY #6
              with arrow                      in file.
              keys.
```

When you select text with the keyboard arrow keys, the terminal highlights your selected line or section in reverse video. When you depress KEY #6, the highlighted area disappears.

You may move the cursor to another location on the file to relocate the line or section of text. Depress the PF1 key, then KEY 6 to reinsert the selected text to the new location on the file.

Use the following sequence to cancel and reset before selecting new text to be moved.

```
              _____
PF1 --->     |  .  |
             |_____|
```

This function does not cancel an operation once it is executed. You may also use this function to reset the "search" function (PF1--->PF3).

6. THE LINE EDIT MODE

The line edit mode allows you to type-in specific editing commands on the VT101 keyboard. When a command is executed, the text file is changed according to the command you specified.

Depress CNTRL/Z on the terminal keyboard to instantly flip from character editing to the line edit mode. An asterisk appears at the bottom of your video display. If you depress the RETURN key, line numbers appear at the left margin. The first number identifies the line position below the cursor from the previous character edit display.

If you scroll the file to the last line with the keyboard return key, the [EOB] symbol ([EOB] = END OF BUFFER), signals the end line of the text file buffer.

EXAMPLE: LINE EDIT MODE

Video Display

```
                 LAYERSTD 3

1
*
2      (LAYERSTD)
*
3      (TELESIS STANDARD LIBRARY)
*
```

In the line edit mode, an asterisk appears between each line number. The asterisk also appears at the last line of your current video display. You may type in specific commands after the asterisk display to change, delete, or transfer lines and large sections of the text file.

Use the line number display to identify the lines of your file that you wish to edit.

6.1 LINE EDITING COMMANDS

Since you must type-in the line editing commands on the keyboard, you may abbreviate each command after the asterisk before executing it. In the examples below, the command abbreviation is the underlined portion of the command name.

*CHANGE

Allows you to instantly flip from the line edit mode to the character edit mode.

EX. *C (RETURN)

----

*COPY

Allows you to copy a range of lines from one position to another on the text file without deleting the original text.

EX. *CO 10 THRU 15 to 100 (RETURN)

Lines 10 through 15 are copied to the line just ahead of line 100.

----

*DELETE

Allows you to delete a line or range of lines from the text file.

EX. *D 10 THRU 15 (RETURN)

Lines 10 thru 15 are deleted from the text file.

----

*EXIT

Allows you to exit and save the contents of the file under the same revision label that was named when the file was opened.

EX. *EXIT (RETURN)

----

*FIND

Allows you to find a line in the file. This command locates the line and displays the asterisk (*) prompt.

EX. *FIND 25 (RETURN)

Line 25 of the text file is located.

---

*MOVE

Allows you to move a line or section to another area of the file. The MOVE command deletes the original text at its original position.

EX. *MOVE 25 THRU 40 to 80 (RETURN)

Lines 25 through 40 are moved to the line just ahead of line 80.

---

*QUIT

Allows you to cancel the current file displayed on the video terminal without cancelling the original version that was named when the file was opened.

EX. *QUIT (RETURN)

---

*REPLACE

Allows you to delete a line or range of lines for insertion of new text. When the lines are deleted, you may begin to insert text at the first line of your deletion.

EX. *REPLACE 25 THRU 50 (RETURN)
26 lines deleted

Lines 25 through 50 are deleted from the file. The REPLACE command allows insertion of text while in the line edit mode. Use CNTRL/Z to exit from the REPLACE command.

EX. *REPLACE 75; This is a text file. (RETURN)

Line 75 of the text file is deleted. The contents "This is a text file." is inserted at line 75.

*TYPE

Allows you to display a range of lines on the file.

EX. *TYPE (RETURN)

Displays the line at the current cursor position.

EX. *T BEGIN (RETURN)

Displays line 1 of the text file.

EX. *T END (RETURN)

Displays the last line of the text file buffer.
The [EOB] prompt is displayed.

EX. *TYPE 40 (RETURN)

Displays line 40 of your text file.

EX. *TYPE WHOLE (RETURN)

Displays and scrolls the entire text file on the video terminal.

----

7. USING THE HELP COMMAND FILE ON THE KEYBOARD EDITOR

You may obtain more information about character and line editing on the VT101 terminal. The HELP command file provides information about additional keypad and line edit commands not discussed in this manual.

NOTE: The keypad and line edit commands outlined in this manual are commonly used in the creation and editing of Telesis text files. However, the additional commands described in HELP may be useful if they suit your specific needs.

There are two ways to obtain HELP on the Telesis Keyboard Editor when creating or editing a text file:

1. In the character edit mode, depress the PF2 key. The terminal displays the keyboard arrow keys and the keypad. You may then depress a key on the keypad to obtain HELP information about that particular function.

2. In the line edit mode, type-in HELP. The terminal displays instructions for obtaining HELP on specific line edit commands.

8. LOGGING-ON AND OFF AFTER COMPLETION

When you EXIT or QUIT from a text file, the system will respond with:

DO YOU WANT TO EDIT ANOTHER FILE (Y OR N):

If you respond with a "Y" the system will request another file name to be edited.

If you wish to stop the editing session, respond with a "N."

If you wish to invoke the editor and the terminal is still logged on, type @TEXT on the keyboard. Then, type in the name of the new or existing text file to be edited.

If you respond with "N" in response to ´DO YOU WANT TO EDIT ANOTHER FILE´, you should log the terminal "off" if you do not intend to continue working on the VT101. A logged-on terminal WILL AFFECT PERFORMANCE of the Telesis system even if the terminal is not being used. After you EXIT or QUIT, type RUN SBYE at the terminal keyboard to log the VT101 off. You must pick the |KEYEDIT/TEXT| command at the Telesis system to log the VT101 back on.

If you are operating the VT101 while the Telesis system is being used by another operator, and if that operator changes projects during your editing session, the text file is saved in the original project when you EXIT.

9. LINES OF TEXT LONGER THAN 65 CHARACTERS

WARNING: If you create a text file on the VT101, then recall and save the file on the function screen editor, the function screen editor will reformat lines of text to a 65-character limit.

This may result in a text file that is improperly formatted when the file is used by the system. (Ex. TEXT NETLIST).

1. THE TELESIS GRAPHICS PROCESSOR
    1.1 GRAPHICS PROCESSOR COMMANDS
    1.2 COMMAND DEFINITIONS
2. COMMANDS LOCATED ON THE WORLD MENU PAGE
3. COMMANDS THAT WILL REPAINT YOUR DRAWING
4. HOW TO USE EACH COMMAND DURING THE CREATION AND EDITING OF YOUR DRAWING
5. THE BLANK/UNBLANK MENU

1. TELESIS GRAPHICS DISPLAY PROCESSOR (OPTIONAL)

The Telesis Graphics Display Processor allows you to perform a variety of new command functions on schematics, PC boards, and symbol drawings displayed on the graphics screen. It is a significant product enhancement that allows you to instantaneously alter the view of your drawing.

With this option, for example, you can |ROAM| within a selected view of your drawing while performing an interactive task (i.e. ADD CONNECTION). You can also instantaneously switch from a |WORLD| view of the entire drawing to a working |WINDOW| and back again.

This section of the manual explains how to use each Graphics Processor command. With these command options, you can increase your speed and viewing versatility when creating, adding and editing connections and symbols.

1.1 GRAPHICS PROCESSOR COMMANDS

When you begin a design project, |ROAM|, |ZOOM|, and |WORLD| commands are displayed at the top of the command menu page. |ROAM| and |ZOOM| are "instant" commands because they perform instantly when picked on the menu screen. The |WORLD| command displays the current view of the drawing on the graphics screen. It also displays the |WORLD| menu containing five command options:

|NEW WORLD| |MOVE WINDOW|

|NEW WINDOW|

|DRAWING|

|REPAINT WORLD|

You may use these commands to alter the view of your drawing while creating and editing your work displayed on the graphics screen.

NOTE: Systems that are updated with Telesis Graphics Processor are called Telesis 2.1 systems. This is hardware designation that differentiates systems with the Telesis Graphics Processor from Telesis 1.0 and 2.0 systems. The commands described in this section are found on Telesis 2.1 systems only.

If you do not have the Graphics Processor option on your Telesis system, refer to BASIC-63 of the manual for the current display commands already on your system.

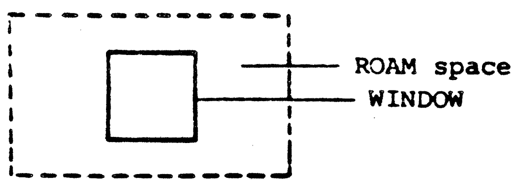
1.2 COMMAND DEFINITIONS

The following table outlines each new command in the Graphics Processor option. These commands are located at the top of the command menu page when you begin a new or existing design project.

<table>
<tr><th>COMMAND</th><th>DEFINITION</th></tr>
<tr><td>|ROAM|</td><td>The |ROAM| command allows you to shift the WINDOW portion of your display within a roam space. |ROAM| allows you to move and locate an area of the drawing not currently in your WINDOW display. You may use the |ROAM| command function instantaneously during interactive tasks such as |ADD CONNECTION| and |ADD SYMBOL|.

ROAM space - This is the area where you have the |ROAM| capability. It is represented by a "dotted" rectangle in the |WORLD| view.

WINDOW - This is the "zoomed-in" area of the drawing where you can perform interactive tasks. The WINDOW is always represented by the solid box that is displayed within the ROAM space in the |WORLD| view.


</td></tr>
<tr><td>|ZOOM|</td><td>The |ZOOM| command instantly zooms-in and zooms-back the WINDOW display at ratios of 2:1, 4:1 and 1:1 with each respective pick of the |ZOOM| command. These ratios are set by the system.

However, you have the option of defining a zoom ratio with the |ZOOM RATIO| command.</td></tr>
<tr><td>|WORLD|</td><td>The |WORLD| command displays the entire drawing or a "zoomed-in" area of the drawing. |WORLD| displays the portion of the drawing where you can alter the WINDOW and ROAM space before performing interactive tasks.</td></tr>
</table>

(Refer to COMMANDS section of this manual for additional information on these commands.)

2. COMMANDS LOCATED ON THE |WORLD| MENU PAGE

When you pick |WORLD|, the following command options appear on the function screen:

|NEW WORLD| |MOVE WINDOW|

|NEW WINDOW|

|DRAWING|

|REPAINT WORLD|

The table below defines each command used to alter the view of your drawing.

COMMAND	DEFINITION
|NEW WORLD|	The |NEW WORLD| command allows you to select a portion of the drawing (or the entire drawing) that you wish to see when you pick the |WORLD| command. It automatically assigns a new ROAM space and WINDOW to the drawing. This is your new viewing, placement and editing portion of the drawing.
|NEW WINDOW|	The |NEW WINDOW| command allows you to select a "zoomed-in" working area within the |WORLD| view. |NEW WINDOW| automatically defines a new ROAM space ("dotted" rectangle). The |WORLD| command instantly displays the new relationship between the |NEW WINDOW| and the ROAM space.
|MOVE WINDOW|	This command allows you to move the WINDOW portion of the drawing within the ROAM space while in the |WORLD| view.
|DRAWING|	This command allows you to fit the entire drawing into the |WORLD| view. It assigns a new ROAM space and WINDOW to the drawing.
|REPAINT WORLD|	This command repaints the |WORLD| view of the drawing. It is used to "refresh" the |WORLD| view after you have performed interactive active tasks (i.e. ADD, DELETE, MOVE, etc.).

(Refer to the COMMANDS section of this manual for additional information on these commands.)

3. COMMANDS THAT WILL REPAINT YOUR DRAWING

The following commands repaint the drawing on the graphics screen. Graphics screen repainting allows you to "refresh" the view of your drawing during placement and editing.

|NEW WORLD|	Repaints the area of the drawing that you specified as the |NEW WORLD| view.
|NEW WINDOW|	Repaints the area on the graphics screen that you defined as the |NEW WINDOW|.
|DRAWING|	Repaints and fits the entire drawing within the ROAM space and the |WORLD| view.
|REPAINT WORLD|	Repaints the area of the drawing that was specified as the |WORLD|.

NOTE: You may use the |DONE| command to flip the current display to the |WINDOW| portion of the drawing. All layer colors assigned to the drawing are displayed.

4. HOW TO USE EACH COMMAND DURING THE CREATION AND EDITING OF YOUR DRAWING

If you are creating or editing a drawing, you may use each Graphics Processor command to increase your speed in selecting the areas of your drawing where you wish to work.

The following section describes how to use each command while editing an active drawing displayed on the graphics screen. Each example shows precisely what happens when you perform each command.

BEGINNING YOUR EDITING SESSION ON AN ACTIVE DRAWING

This section describes each command as you begin editing your drawing. Starting with the view of the entire drawing displayed on the graphics screen, you may select the portion of the drawing where you wish to begin your work.

USING THE |WORLD| COMMAND

The |WORLD| command displays the current portion of the drawing where you are viewing and performing editing tasks. The |WORLD| display may be the entire drawing or only a portion. Your |WORLD| view is the maximum area of the drawing that is allowed for the WINDOW and ROAM space.

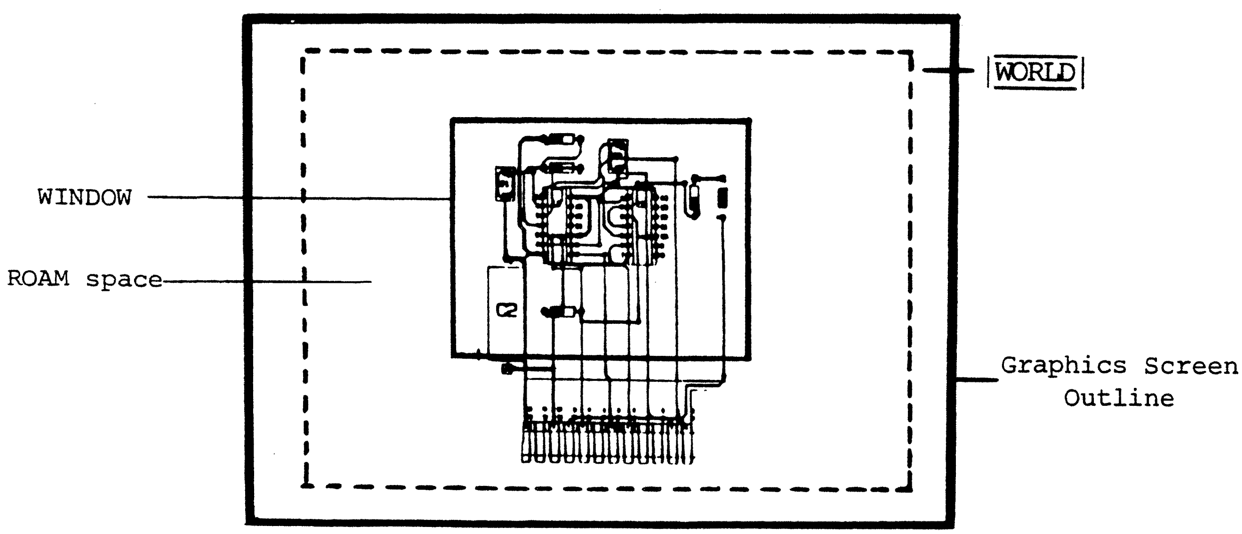
The solid box displayed on the graphics screen represents the WINDOW. The "dotted" rectangle represents the area where you have the capability to perform the |ROAM| function. The |WORLD| command always displays the WINDOW and the ROAM space.

The following diagram represents the |WORLD| view of an entire drawing displayed on the graphics screen. However, you may define a different |WORLD| view by using the |NEW WORLD| command.

EXAMPLE: Displaying the |WORLD| view of an entire drawing

PICK: |WORLD|

RESULT:



When you pick the |WORLD| command to view the current |WORLD| display, the system displays the |WORLD| menu page on the function screen.

EXAMPLE: |WORLD| menu page

DONE							
WORLD							

	NEW WORLD	NEW WINDOW	
	DRAWING	MOVE WINDOW	
	REPAINT WORLD		

The |WORLD| menu contains five optional commands.

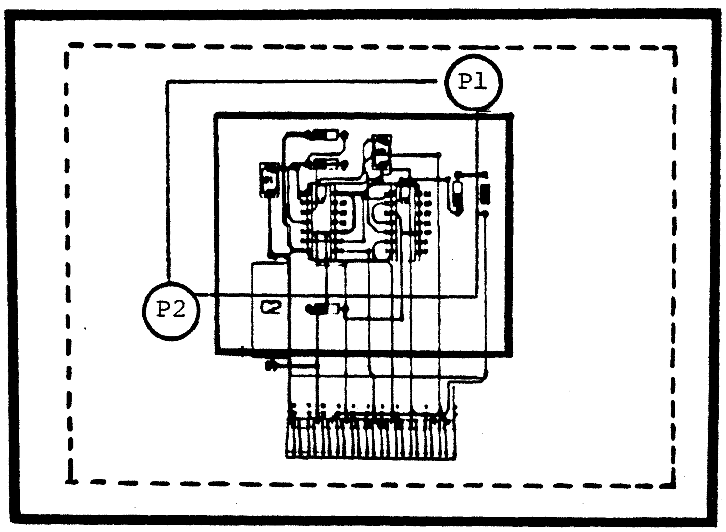
USING THE |NEW WORLD| COMMAND

The |NEW WORLD| command allows you to select a new world view of your drawing. It is a way of "scaling-up" a selected portion of the |WORLD| before performing interactive tasks within that selected portion.

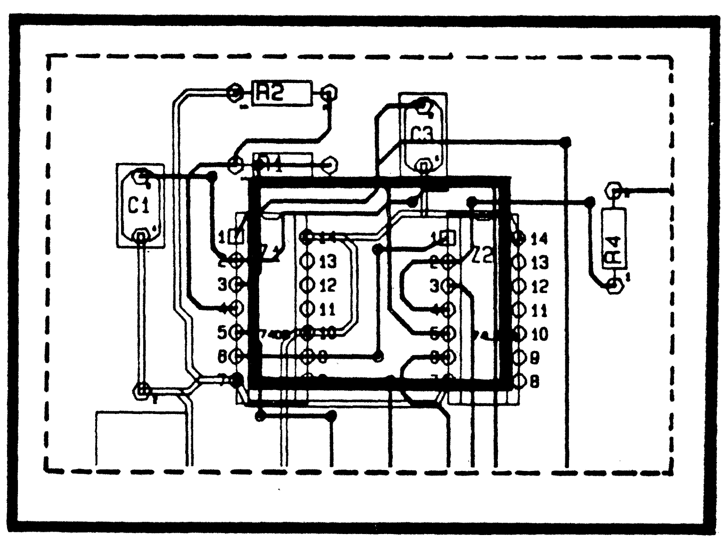
When you command |NEW WORLD| , the |WORLD| menu flips to the tablet menu on the function screen. You may now change the size of your WORLD view with two tablet picks of your choice. As you move the light pen across the tablet menu after your first pick, a "dynamic" rectangle appears. This rectangle pivots at the first pick point on the drawing and moves dynamically until you pick the second point on the drawing.

The area of the drawing selected as the |NEW WORLD| becomes the maximum allowable ROAM space. The "dotted" rectangle outlines the |NEW WORLD| view when your display repaints on the graphics screen.

EXAMPLE: PICK: |NEW WORLD| -----> tablet menu ----> P1 P2



RESULT: |NEW WORLD| display with ROAM space and WINDOW



USING THE |NEW WINDOW| COMMAND

You may use the |NEW WINDOW| command to change the size of the WINDOW and the ROAM space within the current WORLD view.

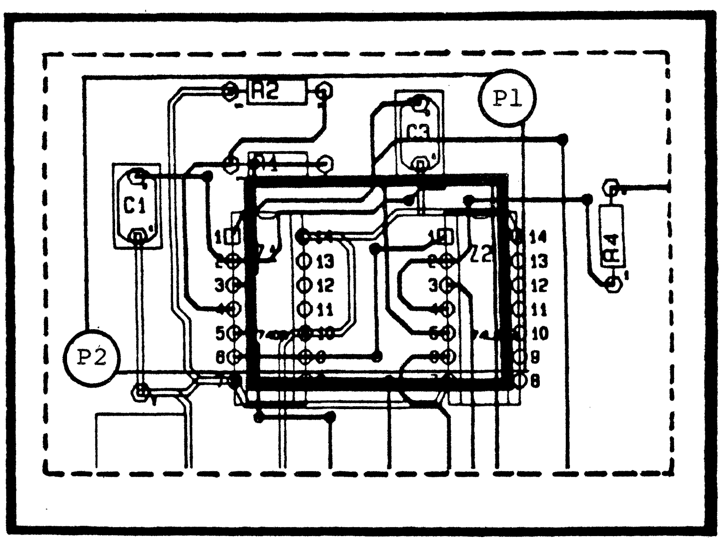
When you command |NEW WINDOW|, the function screen displays the tablet menu. You may enlarge or reduce the size of the WINDOW with two tablet picks. Two "dynamic" rectangles appear after the first pick. The "dotted" rectangle is the ROAM space; the square box within the ROAM space is the WINDOW. This feature allows viewing of the ROAM space and WINDOW displays while selecting the view of your drawing.

The rectangles pivot at the first pick point on the drawing; they move dynamically until you pick the second point that defines the |NEW WINDOW|.

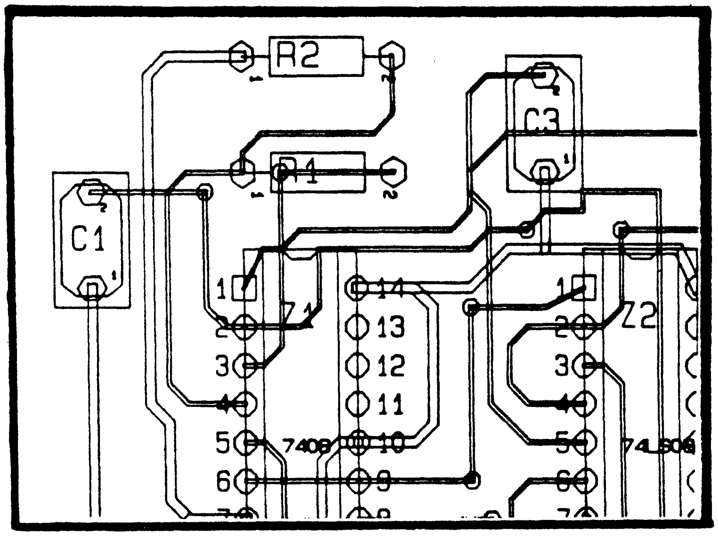
After defining the |NEW WINDOW|, the system assigns a ROAM space that is proportionate to the WINDOW. The |NEW WINDOW| and the ROAM space are always displayed inside the |WORLD| view of the drawing when you command |WORLD|.

EXAMPLE: Pick the |WORLD| command to display the current |WORLD|.

PICK: |NEW WINDOW| ---> Tablet Menu ---> P1 P2



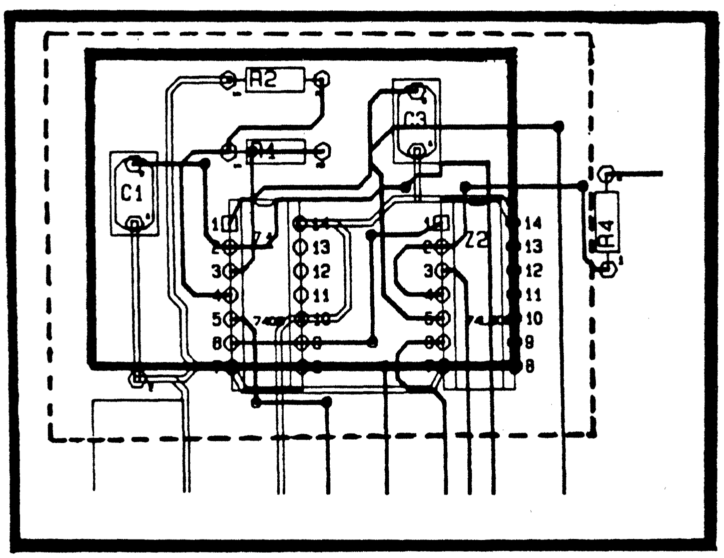
RESULT: The |NEW WINDOW| portion of the drawing is displayed on the graphics screen.



After selecting the |NEW WINDOW|, the WINDOW portion of the drawing is displayed over the entire area of the graphics screen. You may use the |WORLD| command to display the extent of your new ROAM space within the |WORLD| view.

EXAMPLE: Pick |WORLD|

RESULT: |NEW WINDOW| and ROAM space displayed within the |WORLD| view



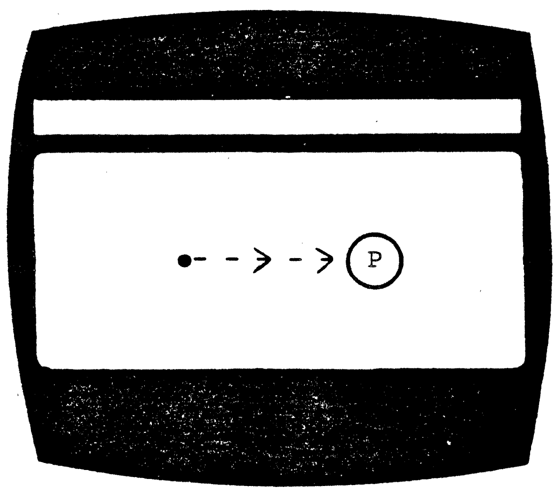
Pick |DONE| to return to the WINDOW display. You now have a selected portion of your drawing where you can perform interactive tasks. (|ADD CONNECTION|, |ADD SYMBOL|, etc.)

USING THE |MOVE WINDOW| COMMAND

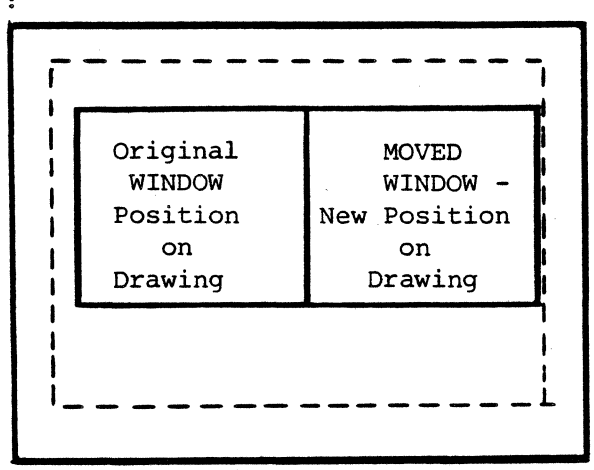
|MOVE WINDOW| allows you to move the WINDOW portion of your display within the ROAM space. This function is performed with your display in the |WORLD| view.

In the |WORLD| view, the current WINDOW and the ROAM space are displayed. You may control the direction and movement of the WINDOW with your light pen and tablet menu interaction. When the WINDOW is moved to your desired location, pick the tablet menu to "freeze" the WINDOW at that location within the ROAM space.

EXAMPLE: |MOVE WINDOW| --->Tablet ---> P
Menu



RESULT:



USING THE |ROAM| COMMAND FUNCTION

After selecting the portion of the active drawing where you wish to work , you may use the |ROAM| command function to move the WINDOW to a new area within the ROAM space. This feature operates when the WINDOW is displayed on the graphics screen.

When you pick |ROAM| , or an interactive command such as |ADD CONNECTION|, the tablet menu appears on the function screen. If you wish to add a connection to a symbol not currently in your WINDOW display, but is within your ROAM space, you may use the ROAM command function to move the WINDOW to that area as you make the connection.

The direction of WINDOW movement during "instant" |ROAM| is controlled by positioning the light pen to one of four edges of the tablet menu. When you move the light pen away from the function screen, |ROAM| stops.

PREREQUISITES:

1. You must have an active drawing displayed on the graphics screen.

2. You must display the WINDOW before you can perform |ROAM| function. If the display is in the |WORLD| view, use the |DONE| command to display the WINDOW.

EXAMPLE: Activating the |ROAM| function

To |ROAM| and locate an area of the drawing within the ROAM space:

```
|ROAM|---> Tablet ----> Pick selected edge of function screen
            menu              for ROAM movement
```

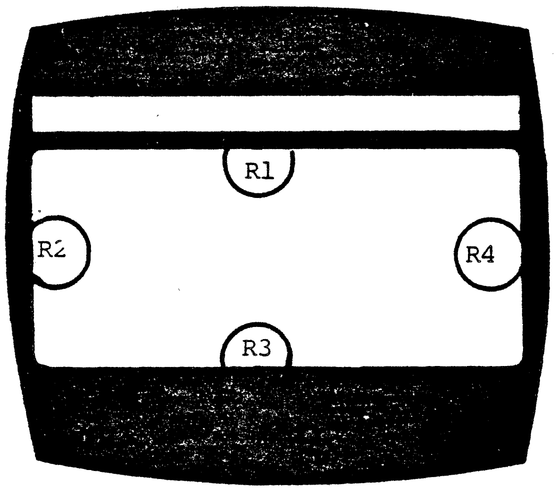
To |ROAM| and perform interactive tasks within the ROAM space:

```
e.g. |ADD CONNECTION|---> Tablet----> Pick selected edge of function
                           menu         screen for ROAM movement
```

INTERACTING WITH THE TABLET MENU TO PERFORM |ROAM|

The following diagram displays the edges on the tablet menu that activate the |ROAM| function. When you move the light pen to any point along the tablet menu edge, the display begins to move.

TABLET MENU



R1 - causes the WINDOW to shift up; the displayed drawing moves down.

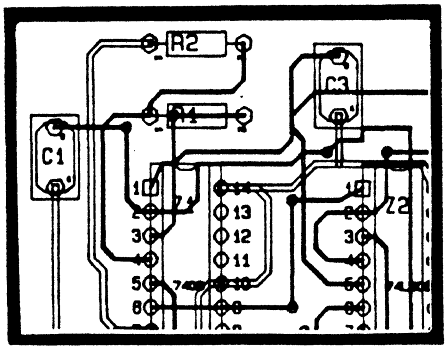
R2 - causes the WINDOW to shift to the left; the displayed drawing moves right.

R3 - causes the WINDOW to shift down; the displayed drawing moves up.

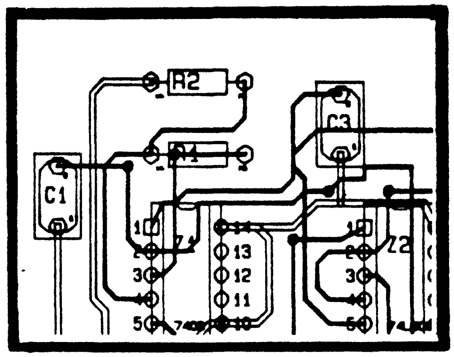
R4 - causes the WINDOW to shift to the right; the displayed drawing moves left.

EXAMPLE: Move the light pen to the Rl edge.

Current WINDOW



RESULT



After moving the WINDOW to the selected area of the drawing, you may pick the |WORLD| command to display the position of the WINDOW within the ROAM space. Use the |DONE| command to return to the current WINDOW display.

NOTE: If you move the WINDOW to the edge of the ROAM space ("dotted" rectangle), the system will "beep". This signals that the WINDOW is at the edge of the ROAM space and can no longer move in that direction.

You may use the following commands to redefine a new ROAM space if you cannot |ROAM| into a desired area of your drawing:

|DRAWING|

|NEW WORLD|

|NEW WINDOW|

|ZOOM RATIO|

USING THE ZOOM COMMANDS

Each |ZOOM| command allows you to zoom-in the current WINDOW display. The "instant" |ZOOM| command function is located at the top of the command menu page and the tablet menu. You may zoom-in and zoom-back the WINDOW at ratios of 2:1, 4:1 and 1:1 with each respective pick of the |ZOOM| command.

Ratios are set by the system. However, you have the option of defining a ratio with the |ZOOM RATIO| command on the command menu page.

With the Graphics Display Processor, there are two ways to zoom-in and zoom-back the WINDOW display:

"Instant" |ZOOM| - Instantly zooms at ratios of 2:1, 4:1 and 1:1 with each respective pick of the |ZOOM| command. It is a time-saving way to zoom-in the WINDOW display.

|ZOOM RATIO| - When you pick |ZOOM RATIO| you must define a ratio on the keyboard menu. |ZOOM RATIO| zooms and repaints the WINDOW display. Repainting provides improved screen resolution of the zoomed-in display when compared with the "instant" |ZOOM|.

NOTE: You must define a ratio on the keyboard menu to zoom-back to the original WINDOW. |ZOOM RATIO| also changes the ROAM space whenever you zoom-in or out of the WINDOW display.

After performing |ZOOM| or |ZOOM RATIO|, you may pick the |WORLD| command to view the size of the WINDOW in relation to the ROAM space. "Instant" |ZOOM| changes the size of the WINDOW only, and not the ROAM space. |ZOOM RATIO|, however, changes both.

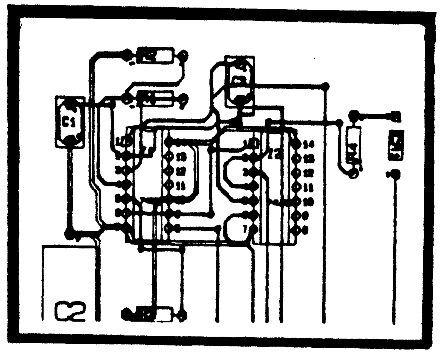
EXAMPLE: Using the "instant" |ZOOM| command

ZOOM ----> PICK 1 = WINDOW zoomed-in at 2:1 ratio

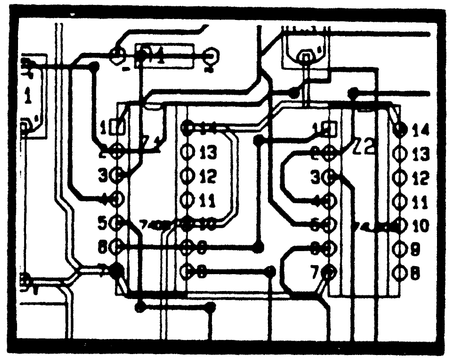
PICK 2 = WINDOW zoomed-in at 4:1 ratio

PICK 3 = WINDOW zoomed-back to 1:1 (original display)

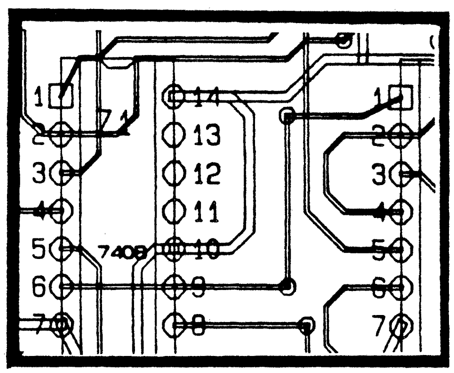
Current WINDOW display ----->



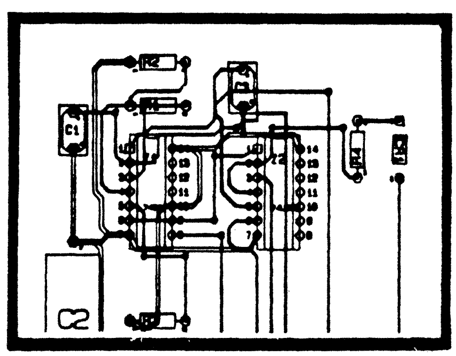
Pick |ZOOM| (2:1 ratio) ----->



Pick |ZOOM| (4:1 ratio) ----->



Pick |ZOOM| (1:1 ratio) ----->



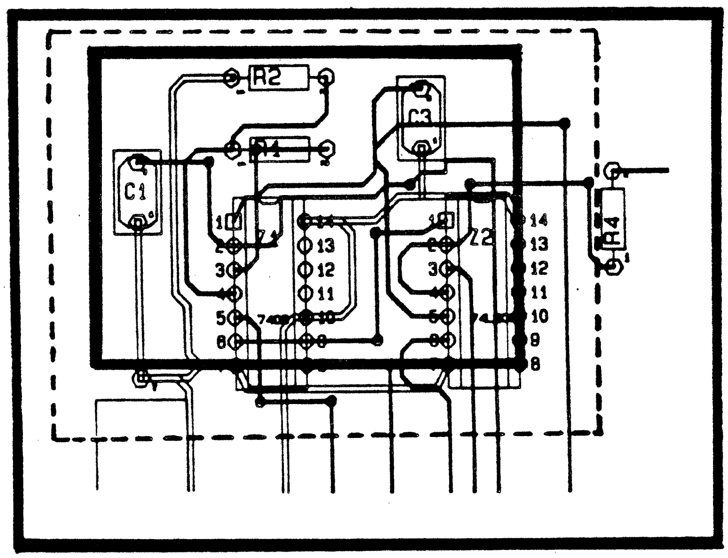
USING THE |DRAWING| COMMAND

|DRAWING| repaints and fits the entire drawing within the |WORLD| view and the ROAM space. After performing interactive tasks you may use the |DRAWING| command to establish an entire view of the drawing before selecting a |NEW WORLD| and |NEW WINDOW|.

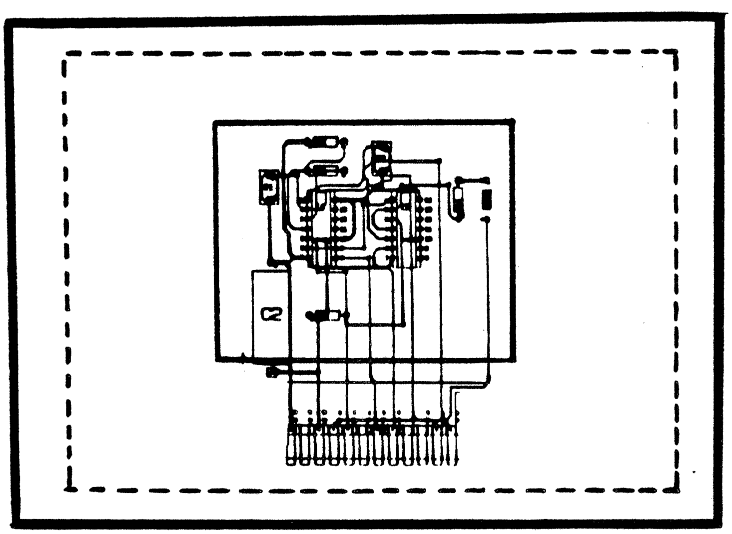
|DRAWING| is simply a method of zooming-back and cancelling the previous |WORLD| view, ROAM space and WINDOW display.

NOTE: Repainting time depends on the amount of graphics used in the drawing.

EXAMPLE: Current |WORLD|



Pick |DRAWING|



USING THE |REPAINT WORLD| COMMAND

You may use the |REPAINT WORLD| command to repaint the |WORLD| display after performing interactive tasks. It is simply a method of "refreshing" the |WORLD| view without repainting the entire drawing (if the |WORLD| was only a portion of the drawing).

Graphics screen repainting time varies with the amount of graphics displayed in the |WORLD| portion of the drawing.

5. USING THE |BLANK/UNBLANK| MENU (TELESIS 2.1 SYSTEM ONLY)

The |BLANK/UNBLANK| menu allows the operator to select and instantly blank and unblank colors assigned to displayed layers on the active drawing. This feature may be used to temporarily blank any number of colors and their assigned layers to allow clear visibility on a congested drawing during editing.

The |BLANK/UNBLANK| menu is located within the SCHEMATIC, BOARD, and SYMBOL menus.

When the operator picks the |BLANK/UNBLANK| command, the following menu is displayed on the function screen:

DONE	WORLD	ROAM				ZOOM	

BLANK ALL	BLANK RED	UNBLANK RED	UNBLANK ALL
	BLANK BLUE	UNBLANK BLUE	
	BLANK GREEN	UNBLANK GREEN	
	BLANK YELLOW	UNBLANK YELLOW	
	BLANK VIOLET	UNBLANK VIOLET	

For example, when the operator picks |BLANK RED|, all data base layers currently assigned the color RED will be blanked. When |UNBLANK RED| is picked, all layers currently assigned the color RED will instantly reappear on the graphics screen.

NOTE: If the operator picks the |BLANK RED| command, then proceeds to edit the drawing, elements displayed on the drawing will not highlight during interactive commands, such as |MOVE SYMBOL|. It is recommended that the operator use the |BLANK LAYER| command on specific layers so that the highlighting feature of the Telesis system continues to be visible.

REASSIGNING COLORS TO BLANKED LAYERS

If the operator assigns a new color to a specific layer, and if this layer was previously blanked (by color) with a command from the |BLANK/UNBLANK| menu, |ZOOM RATIO| must be used to re-display the blanked layer at the new color. This will only occur if the newly assigned color is currently visible (not blanked) on the graphics screen.

For example, LAYER 1 is assigned the color GREEN, then blanked with the |BLANK GREEN| command. If the operator uses the |ASSIGN RED| command on LAYER 1, |ZOOM RATIO| must be used to display the elements on LAYER 1 now assigned the color RED. |UNBLANK GREEN| will then re-display all remaining GREEN layers blanked with |BLANK GREEN|.

USING THE |BLANK ALL| and |UNBLANK ALL| COMMANDS

The |BLANK ALL| command blanks all colors displayed on the active drawing. The operator may use this command to blank all colors and their assigned layers, then UNBLANK only the desired colors. This is a fast method of displaying one or two colors on the drawing and their assigned layers.

The operator may use the |UNBLANK ALL| command to redisplay all colors previously blanked.

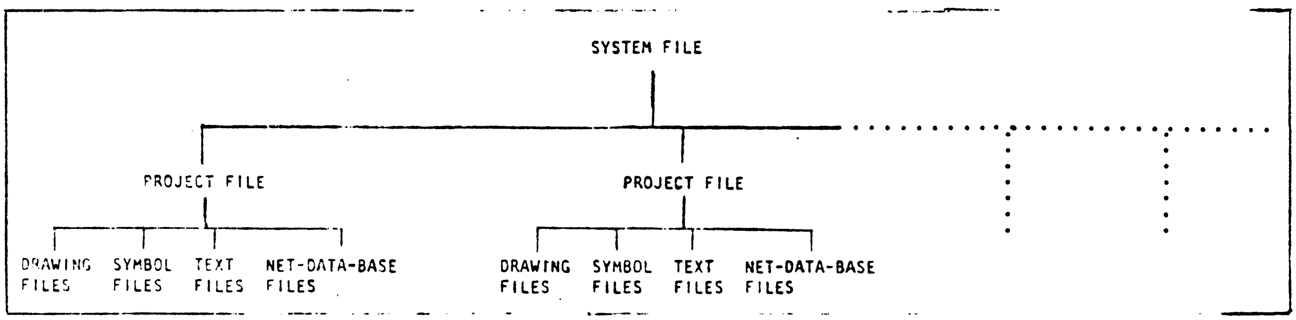
NOTE: |BLANK ALL| will blank the displayed grid on the graphics screen. If the operator chooses to unblank colors on the drawing one at a time with each UNBLANK command, the grid specification will not reappear on the graphics screen. |UNBLANK ALL| must be used to re-display the grid specification.

FILE MANAGEMENT AND ARCHIVES

FILE MANAGEMENT

1. TELESIS FILE STRUCTURE
2. DEFINITIONS OF TYPES OF FILES
3. FILE INDEXES
4. FILE MANAGEMENT CAPABILITIES
5. HOW TO GET TO THE FILE MANAGEMENT MENUS
6. COPYING FILES
7. DELETING FILES
8. CHANGING FILE NAMES
9. CHANGING FILE REVISION LABELS

1. TELESIS FILE STRUCTURE



The Telesis system organizes your data into the hierarchy of files illustrated in the diagram above.

Figure FILE MANAGEMENT-I shows how some typical data would fit into this structure.

SYSTEM FILE

project ABC rev 1

- drawing SHEET1 rev A
- drawing SHEET1 rev B
- drawing SHEET2 rev A
- drawing PCB rev 1
- text file EXTRACTION-LOG rev 1
- file NET-DATA-BASE rev 1

project XYZ rev 1

- drawing SCHEM rev 1
- drawing PCB rev 1
- text file BOM-REPORT rev 1
- file NET-DATA-BASE rev 1
- text file DESIGN-RULES1 rev 1
- text file VIOLATIONS1 rev 1
- file NET-DATA-BASE rev 2

project SYSTEM-LIBRARY rev 1

- symbol DIODE350 rev 1
- symbol DIP14 rev 1
- symbol DIP14 rev 2
- text file DIODE350.PIN rev 1
- text file DIP14.PIN rev 1
- text file DIP14.PIN rev 2
- drawing STANDARD-FORMAT rev A
- drawing STANDARD-FORMAT rev B

Figure FILE MANAGEMENT-I: EXAMPLE OF TELESIS FILE STRUCTURE

2. DEFINITIONS OF TYPES OF FILES

SYSTEM FILE	This is the master file. It contains all project files.
PROJECT FILE	A project file typically contains the drawing files, text files, and net data base files relating to the design of one printed circuit board. SYSTEM-LIBRARY is the name of a project file that is used as a library. It typically contains symbol files, pin record files and other files that are likely to be used by more than one project.
DRAWING FILE	Each drawing file contains a single drawing, typically a board drawing or a schematic sheet.
SYMBOL FILE	Each symbol file contains the graphics for a single symbol, typically a logic symbol or a component symbol.
TEXT FILE	A text file contains data in text format, for example, a pin record file for a component symbol, a net comparison report, a list of design rules, a list of design rules violations.
NET-DATA-BASE FILE	This file contains the net data base for a single board design.

3. FILE INDEXES

An index is a list of items. Index commands allow you to display the projects or files within the system or a specified project, respectively. Items in an index are listed in reverse chronological order, beginning with the item most recently created or worked on and ending with the item first created or worked on.

Index commands are listed below.

SYSTEM INDEX	This command lists all of the projects in the system.
PROJECT INDEX	This command lists all of the files that are within the specified project.
CURRENT INDEX	This command lists all of the projects or files within the system or active project, respectively.

4. FILE MANAGEMENT CAPABILITIES

- COPYING FILES
- DELETING FILES
- CHANGING FILE NAMES
- CHANGING FILE REVISION LABELS

As you work on the Telesis system, you will find that these file management capabilities can simplify your work in many ways, for instance:

EXAMPLE 1 COPYING PROJECT FILES -- You might want to change a nearly completed board design without losing your original design (because of an engineering change order, or just to try out a different way of designing the board). To do this you can use the |COPY PROJECT| command to make a copy of the original project file. You can then make changes to the copy without altering the original.

EXAMPLE 2 COPYING DRAWING FILES -- In creating your symbol library, you will be drawing many symbols that are quite similar to one another. You can avoid much of the repetitive work involved by using the |COPY DRAWING| command. After completing the drawing for a DIP16, for example, you could make a copy of it and then use the copy as the basis for a DIP20 drawing.

EXAMPLE 3 COPYING TEXT FILES -- As with symbol drawings, the pin record files you create are often quite similar to one another. You can save yourself time by using the |COPY TEXT FILE| command to use a copy of one pin record file as the basis for another.

EXAMPLE 4 DELETING FILES -- Usually you will use the DELETE commands to clean up your files. As you work on one drawing, it's a good habit to save the old revision of it, but eventually you may accumulate several old revisions that you are sure you no longer need. You can use the |DELETE DRAWING| command to eliminate these.

EXAMPLE 5 CHANGING FILE NAMES AND REVISION LABELS -- You can use these commands to correct errors, or to bring your file names and revision labels into conformance with a standard of your own.

5. HOW TO GET TO THE FILE MANAGEMENT MENUS

File management commands can be accessed from the system, project, or drawing level. The following list illustrates the procedures to access the File Management commands from each level.

1. SYSTEM LEVEL
   - Function screen displays:
     BOOT-UP MENU
   - Select the FILE MGMT & ARCHIV command
   - Function screen displays:
     TOP FILE MANAGEMENT OR ARCHIVE MENU
   - Select FILE MANAGEMENT
   - Function screen displays:
     COPY & DELETE MENU

2. PROJECT LEVEL
   - Function screen displays:
     TOP PROJECT LEVEL MENU
   - Select FILE MANAGEMENT
   - Function screen displays:
     COPY & DELETE MENU

3. DRAWING LEVEL
   - Function screen displays:
     TOP DRAWING LEVEL MENU
   - Select FILE MGMT/ARCHIV
   - Function screen displays:
     TOP FILE MANAGEMENT OR ARCHIVE MENU
   - Select FILE MANAGEMENT
   - Function screen displays:
     COPY MENU

All of the File Management commands are not available at every level. Only the file management commands that are available at a particular level are included in the File Management menu displayed in the level. (A project cannot be deleted from within a drawing; therefore, the DELETE PROJECT command is not available in the File Management menu displayed at the drawing level.)

6. COPYING FILES

These are the commands used for copying files:

COPY PROJECT
COPY CUR PROJECT
COPY DRAWING
COPY SYMBOL
COPY TEXT FILE
CPY NET-DB

Each command is described in detail in the Command Description Section of this manual.

Here are some general principles to keep in mind when you use these commands:

ORIGINAL IS UNALTERED

When you make a copy of a file, the original file remains unaltered.

NAME CHANGES

DRAWING FILES, SYMBOL FILES, TEXT FILES -- When you copy any of these items from one project to another, you may specify a new item name and/or revision label, or you may keep the original name and revision label. However, when your copy is in the same project file as the original, you must specify a new name and/or revision in order to distinguish the copy from the original.

NET-DATA-BASE FILES -- The net data base file is always named NET-DATA-BASE. When you copy a net data base from one project to another, you may give it a new revision label or you may keep the original revision label. However, when your copy is in the same project file as the original, you must specify a new revision label in order to distinguish the copy from the original.

PROJECT FILES -- When you copy a project file, the copy must have a different name and/or revision label in order to distinguish it from the original (except for archival copies which will be discussed in later sections of the manual.)

REVISION LABELS OF ORIGINALS

Whenever you copy a file, you may specify the revision label as well as the name of the original to be copied. If you do not, the system will copy the most recently used revision of the file you have named. Note that the "most recently used" revision is not always the same as the "most recently created" revision. For example, you might have created revisions 1, 2, 3 and 4 of drawing SCHEM in numerical order, but revision 3 may be the one that you looked at most recently. In this case, revision 3 is the "most recently used" revision.

To find out which revision of a file was the most recently used, look at the index where the file is listed. The revision listed closest to the bottom is the most recently used one.

REVISION LABELS OF COPIES

When you specify the name of the copy, you may specify a revision label too. If you do not, the system will assign a revision label identical with the original file's.

SHORT CUTS

Looking at the command descriptions, you will see that there are many instances in which the keyboard input called for by the command can be omitted.

OMIT CURRENT PROJECT NAME -- Whenever the command description calls for the keyboard input of a project name, you can omit it if the project you wish to specify is the currently active project.

OMIT NAMES OF TEXT FILES, DRAWING FILES, SYMBOL FILES -- Whenever the command description calls for keyboard input specifying the copy name for these files, you can omit it if you want the copy name to be the same as the original name.

OMIT NAME OF NET-DATA-BASE FILE -- You can always omit keyboard input of the name of the net data base file since the name is always NET-DATA-BASE. If you wish to specify a revision, you may input the revision label alone in place of the name.

7. DELETING FILES

These are the commands used for deleting files:

DELETE PROJECT
DELETE DRAWING
DELETE SYMBOL
DELETE TEXT FILE
DEL NET-DB

Each command is described in detail in the Command Description Section of this manual.

Here are some general principles to keep in mind when you use these commands.

BE CAREFUL!

Deleting a file is permanent. Don't delete a file unless you are sure you no longer need it.

REVISION LABELS

As with the COPY commands, you may specify a revision label as well as the name of the file to be deleted. If you do not the most recently used revision will be deleted. To find out which is the most recently used revision of a file, check the index in which the file is listed. The revision listed closest to the bottom is the most recently used one.

SHORT CUTS

OMIT CURRENT PROJECT NAME -- Whenever the command description calls for the keyboard input of a project name, you can omit it if the project you wish to specify is the currently active one.

8. CHANGING FILE NAMES

You get to the CHANGE NAME & REV MENU from the the COPY & DELETE MENU:

- Pick |CHANGE NAME/REV|
- Function screen display:
  - CHANGE NAME & REV MENU

These are the commands used for changing file names:

|CHG PROJECT NM|
|CHG CUR PROJ NM|
|CHG DRAWING NM|
|CHG SYMBOL NM|
|CHG TEXT FILE NM|

Each command is described in detail in the Command Description Section of this manual.

Here are some general principles to keep in mind when you use these commands:

ALL REVISIONS ARE CHANGED

When you change the name of a file, you change the name of all revisions of that file. You may not specify a particular revision of the file.

ALPHANUMERIC CHARACTERS

When you change a file name, you follow the same rule as when you name a new file. You may use up to 18 alphanumeric characters, and you may not use blank spaces in the name.

NET-DATA-BASE FILE NAME CANNOT BE CHANGED

There is no command for changing the name of the net data base file. It is always named NET-DATA-BASE. If you want to distinguish one net data base file from another, you can change its revision label.

SHORT CUTS

OMIT CURRENT PROJECT NAME -- Whenever the command description calls for the keyboard input of a project name, you can omit it if the project you wish to specify is the currently active one.

9. CHANGING FILE REVISION LABELS

These are the commands used for changing file revision labels:

CHG PROJECT RV
CHG CUR PROJ RV
CHG DRAWING RV
CHG SYMBOL RV
CHG TEXT FILE RV
CHG NET-DB RV

Each command is described in detail in the Command Description Section of this manual.

You find these commands on the same menu page as the name changing commands.

Here are some general principles to keep in mind when you use these commands:

OLD REVISION LABEL

When you input the file name, you may specify the revision label that you want to change. If you do not, the label of the most recently used revision wil be changed. To find out which is the most recently used revision of a file, check the index in which the file is listed. The revision listed closest to the bottom is the most recently used one.

ALPHANUMERIC CHARACTERS

When you change a revision label, you follow the same rules as when you give a revision label to a new file. You may use up to 4 alphanumeric characters, and you may not use spaces in the label.

SHORT CUTS

OMIT CURRENT PROJECT NAME -- Whenever the command description calls for the keyboard input of a project name, you can omit it if the project you wish to specify is the current active one.

OMIT NAME OF NET-DATA-BASE FILE -- You can always omit input of the name of the net data base file since the name is always NET-DATA-BASE. If you wish to specify the revision label you want to change, you may input the revision label alone in place of the name.

FLOPPY DISK ARCHIVES

1. GENERAL INFORMATION

On the Telesis system, as on all CAD systems, there are space limitations. The archival capability allows you to store projects and data that you are not using so that there will be adequate space on the system for your current work.

You may use floppy disks for archives or, if you have purchased the magnetic tape option, you may use tape. This section discusses the use of floppy disks.

2. INSERTING AND REMOVING A FLOPPY DISK

NOTE! Do not insert or remove the disks when you are on the ARCHIVE-FLOPPY menu pages.

INSERTING A FLOPPY DISK	
STEP 1	Hold the disk by its cover. Be careful not to touch exposed surfaces. Be careful not to bend the disk. NOTE: If the floppy disk is WRITE protected, ensure that a WRITE ENABLE tab is attached before archiving.
STEP 2	Insert the disk in the disk drive slot with the notched end first and the label facing up. Push gently until it clicks into place.
STEP 3	Push the door down until it clicks shut.

REMOVING A FLOPPY DISK	
STEP 1	Push up on the small bar beneath the door until the door clicks open.
STEP 2	The disk will be partially ejected. Remove it gently.

BE CAREFUL!: Remove each disk as soon as you are finished using it. Leaving them in wears them out more quickly. In particular, do not leave a disk in a partially ejected position.

3. ARCHIVE-FLOPPY CAPABILITIES AND COMMANDS

COPYING FILES TO AND FROM DISKS
DELETING FILES FROM DISKS
CHANGING FILE NAMES ON DISKS
CHANGING FILE REVISION LABELS ON DISKS

To exercise these capabilities you use the same commands that you use for file management. In fact, you will probably use the COPY commands more often for copying files to and from disks than for file management.

Read the File Management Section (page FILE MANAGEMENT-1) for a general discussion of these commands, and refer to the Command Description Section for a precise explanation of each command.

When using a copy command for archives rather than for file management, you input the characters |F| |:| before the name of any project file on the floppy disk. (You must input these characters even when you omit the project name.)

EXAMPLE 1 To copy a text file from the system to a disk--

|COPY TEXT FILE| |A|L|P|H|A|,|B|O|M|-|R|E|P|O|R|T| |ENTER|
|F|:|A|L|P|H|A|,|ENTER|

EXAMPLE 2 To copy a text file from the disk to the system--

|COPY TEXT FILE| |F|:|A|L|P|H|A|,|B|O|M|-|R|E|P|O|R|T| |ENTER|
|A|L|P|H|A|,|ENTER|

EXAMPLE 3 To copy a text file in the current project from the system to a disk--

|COPY TEXT FILE| |B|O|M|-|R|E|P|O|R|T| |ENTER| |F|:| |ENTER|

VERIFYING THE DISK COPY

Whenever you copy a file to a floppy disk, the system will display a prompt asking you if you want the copy verified. Pick |->PAGE| for "yes"; pick |CANCEL| for "no". If you pick |->PAGE|, the copy will be made and verified. If you pick |CANCEL|, the copy will be made but not verified.

We recommend that you always instruct the system to verify the disk copy. In verifying, the system makes the copy, and then reads through the copy comparing it with your original file(s).

DIFFERENCES BETWEEN THE COPY AND THE ORIGINAL

If the system finds differences between the copy and the original, it will report the file name and revision label where the difference occurs. When verification is finished, the system will report the total number of verification errors if there has been more than one file copied. The system displays these messages on the function screen.

DISK ERRORS

If the system finds a problem with the disk itself, it will stop the verification process and report a disk error on the function screen. Try using a new disk to copy your file(s). If you still receive a disk error message, have the disk drive checked.

MULTIPLE FLOPPY DISKS

If you are verifying copies made on multiple floppy disks, the system will prompt you to reinsert the disks after copying so that it can verify them.

5. INDEXES TO FILES ON FLOPPY DISKS

Use these index commands to display the contents of files on the disk:

|FLOPPY INDEX| - If you are at the project level, use this command to list all files on the disk that have the current project name.

- If you are at the system level, use this command to list all files on the disk.

|PROJECT INDEX|F|:|*| - If you are at the project level and wish to list all files on the disk, use this command and the characters following it.

6. HOW TO GET TO THE ARCHIVE-FLOPPY MENUS

As with the FILE MANAGEMENT menus, you must be at the system level or the project level to use the ARCHIVE-FLOPPY menus.

- Use the |NEW PROJECT|, |OLD PROJECT|, or |FILE MGMT/ARCHIV| commands to get to the MENU OF MENUS
- Function screen display:
  - o MENU OF MENUS
- Pick |ARCHIVE-FLOPPY|
- Function screen display:
  - o NEW/OLD FLOPPY MENU
- Pick |NEW FLOPPY| or |OLD FLOPPY| depending on whether the mounted disk is blank or has files on it
- Function screen display:
  - o ARCHIVE-FLOPPY COPY & DELETE MENU

7. RECOMMENDED USE OF FLOPPY DISKS

You use floppy disks for two purposes:

A. To free space on the system for your current work.
B. To back up your files.

A. Freeing Space on the System

We recommend that you store your unused projects and library files as archives, and that, as you work on a new project, you follow these overall steps:

STEP 1 Copy all the logic symbols you think you will need for your schematic from disk archives to the system. Copy them into a new project file and name it SYSTEM-LIBRARY.

STEP 2 Open your new design project and schematic drawing, and complete the schematic using the symbols in your SYSTEM-LIBRARY.

STEP 3 After you have used the |EXTRACT NETLIST| command and thoroughly checked your schematic drawing, you may delete the SYSTEM-LIBRARY project file.

STEP 4 Copy your schematic and net data base to a disk to be saved there.

STEP 5 Delete your schematic drawing from the system. (Do not delete the net data base because you will need it to design the board.)

STEP 6 Copy all the component symbols you think you will need for your board design from disk archives to the system. Again, copy them into a project file named SYSTEM-LIBRARY.

STEP 7 When you have completed the board design, you can delete the SYSTEM-LIBRARY project file again.

STEP 8 When you have completed your entire project, you may copy it to a disk to be saved there.

At any time during your work on a project, you may copy the project to a disk, and delete it from the system if you want to free the system for work on another project. You may then copy it back to the system when you want to resume work on it.

B. Backing Up Your Files

In addition to the recommended steps above, it is important to periodically back up your work by copying it to disks (or tape).

COMPLETED WORK — Always make a back-up copy when you have completed a drawing or project or series of library files. It's good practice when you have a completed board design, to keep 2 archival copies - one on-site and one off-site.

WORK-IN-PROGRESS — We recommend that you make a back-up copy of your work-in-progress on a daily basis.

8. PROCEDURE IF YOUR FILE DOES NOT FIT ON ONE FLOPPY DISK

USING MULTIPLE FLOPPY DISKS

If the file or files you copy to the floppy disk exceed the capacity of the present floppy, the Telesis system will copy as much as it can on the present floppy, and then display the message:

```
The floppy is full so the copy must be continued on another
floppy.  Insert a new floppy.  This new floppy will be first
initialized which deletes any data that already exists on the
floppy.

Hit [->PAGE]  when the new floppy is ready
Hit [CANCEL]  to abort the command
```

Simply follow these directions, and the system will continue to copy out files, and ask for more floppies until it has copied all the requested data.

You may copy a maximum of 90 files to a multiple set of floppy disks.

NOTE:!! As you load each floppy disk into the drive, make sure you write clearly on it which number it is: #1 for the first floppy, #2 for the second, and so on.

When the system has finished copying to the last floppy, it will display the message:

```
Put in floppy number 1
Hit [->PAGE]  when floppy is ready
Hit [CANCEL]  to abort the command
```

At this point, you must re-insert the floppy disk on which the beginning part of the file(s) was copied so that the system can record the total number of disks required for saving the copied files.

RETURNING FILES TO THE SYSTEM THAT WERE RECORDED ON MULTIPLE FLOPPY DISKS

When you return files from a set of multiple floppy disks, you must load the floppies into the disk drive in the same order they were written originally: #1 first, #2 second, etc. The Telesis system will prompt you for each disk, after it has copied the requested data from the first disk. Simply follow the requests for disks, loading each disk as requested by the system, for example:

```
Put in floppy number 2
Hit |->PAGE| when floppy is ready
Hit |CANCEL| to abort the command
```

COPYING ADDITIONAL FILES TO AN EXISTING MULTIPLE FLOPPY SET

If your wish to add additional files to an existing set of multiple floppy disks, you MUST first copy all information from the floppy set back to the Telesis system.

You may then proceed to add the new files to the system. Use the methods previously described to re-copy the files to a multiple floppy set. Use the |NEW FLOPPY| command to initialize and delete the data stored on each floppy prior to copying back to the same multiple floppy set.

CAUTION: DO NOT add to or delete from a multiple floppy set.

MAGNETIC TAPE ARCHIVES

1. GENERAL INFORMATION
2. LOADING AND UNLOADING A TAPE
3. ARCHIVE-TAPE CAPABILITIES
4. HOW TO GET TO THE ARCHIVE-TAPE MENUS
5. COPYING TO TAPE
6. TAPE SECTIONS: ADVANCING, REWINDING, AND INDEXES
7. COPYING FROM TAPE
8. ERASING FILES FROM A TAPE
9. RECOMMENDED USE OF MAGNETIC TAPE

1. GENERAL INFORMATION

The magnetic tape option increases the archival capacity of your Telesis system.

The advantage of tape archives over floppy disk archives is that you can store a much greater amount of information on a single tape than on a single disk.

In practice, using both tape and disk archives gives you considerable flexibility. For example, you could store daily back-ups or single drawings on disks, and use tape for longer term or permanent storage of entire projects.

MAGNETIC TAPE POWER CONTROLS

The power control for the magnetic tape option is located at the upper right of the computer cabinet.

TAPE DRIVE ON:	Press the POWER button.	Button will light.
TAPE DRIVE OFF:	Press the POWER button.	Light will extinguish. BE CAREFUL!: Do not turn off the tape drive until you have indicated that you are finished with the ARCHIV-TAPE menus by picking |DONE|.

2. LOADING AND UNLOADING A TAPE

NOTE!: Do not load or unload a tape when you are on the ARCHIVE-TAPE menu pages.

LOADING A TAPE (see Figure MAGNETIC TAPE-I)	
STEP 1	Check to see that the POWER button is lit.
STEP 2	Install a write-enable ring on the tape you are loading if you intend to copy information to the tape. (See Figure MAGNETIC TAPE-II) BE CAREFUL!: If you accidentally pick the |NEW TAPE| command with the write-enable ring installed, you will destroy existing data on the tape. You can prevent this accident by removing the ring if you know that you will only be copying from, and not to, the tape you are loading.
STEP 3	Pull out reel-locking lever on the lower tape hub.
STEP 4	Place the tape reel on the hub so that the tape will unwind in a clockwise direction. Push in locking lever.
STEP 5	Install empty take-up reel on the upper hub in the same way.
STEP 6	Thread the tape following the arrow guide markers.
STEP 7	Wrap several turns of tape clockwise around the take-up reel.
STEP 8	Close front cover to protect the tape and mechanism from dust. NOTE!: It is important to keep the dust cover closed except when loading or unloading the tape. Otherwise, data reliability can be impaired by contaminants.
STEP 9	Push LOAD button. This tensions the tape.
STEP 10	Push LOAD button a second time. This advances the tape to its start point. LOAD button will light.
STEP 11	Check to see whether the WRT EN light is lit. If lit, a write-enable ring is in place on the loaded tape. If not lit, the write-enable ring is absent. (See Step 2 above.)
STEP 12	Press ON LINE button. It will light, indicating that the tape is ready for use with the Telesis system.

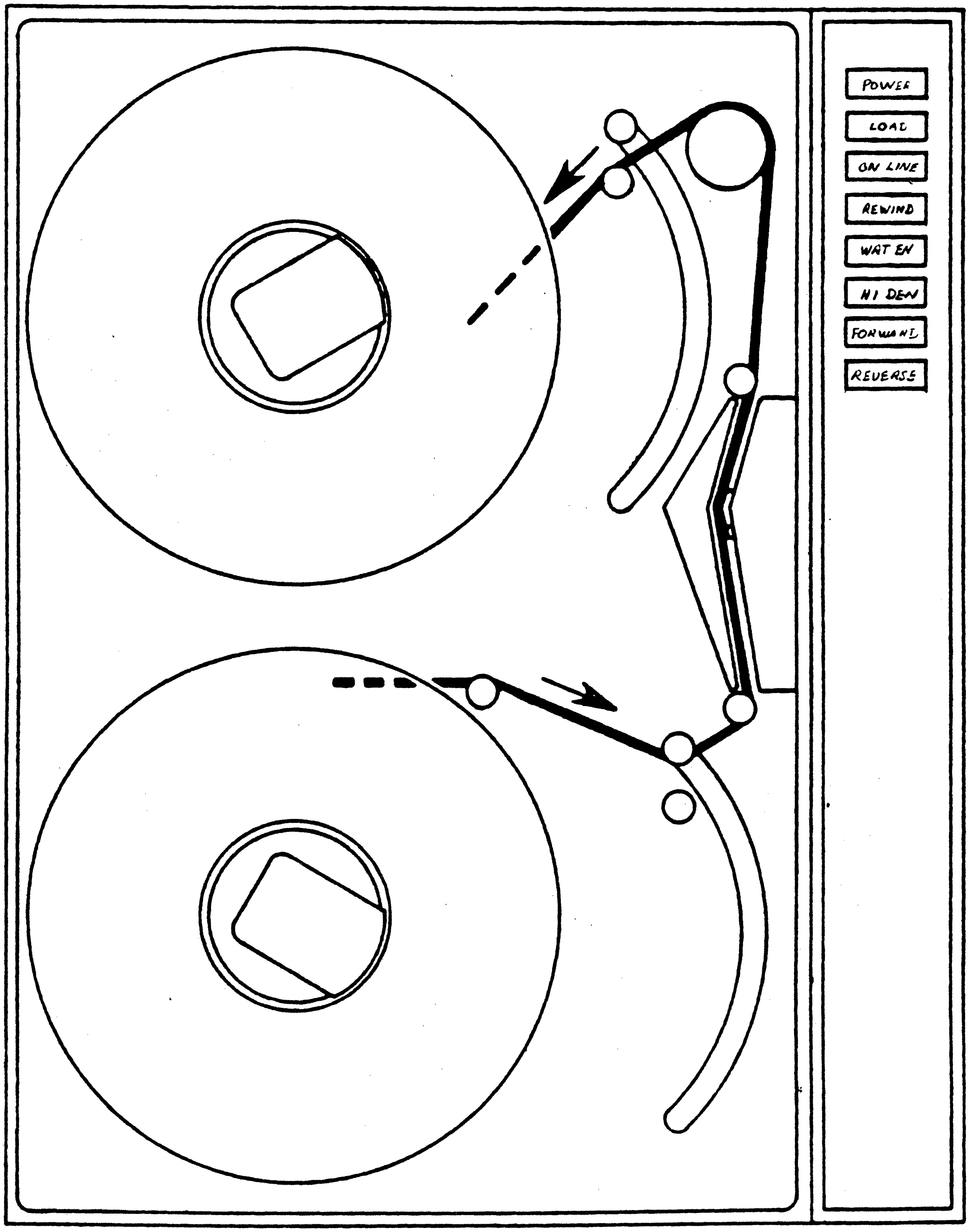
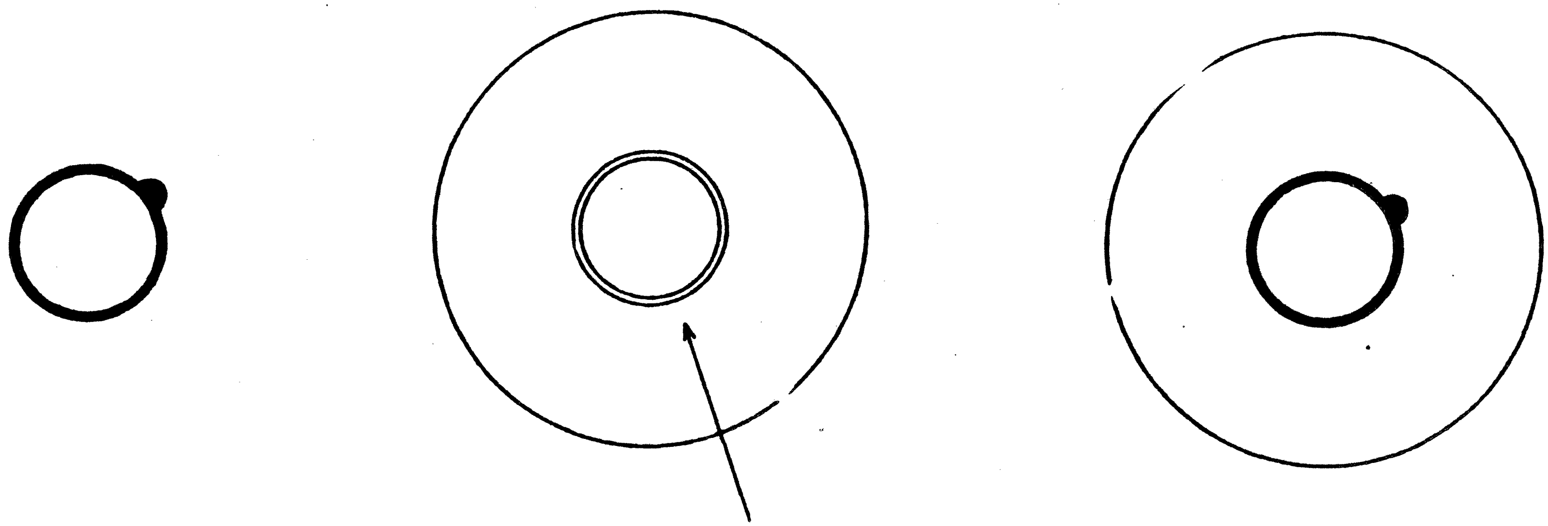


Figure MAGNETIC TAPE-I: TAPE CONTROLS AND THREADING PATH



WRITE-ENABLE RING

WRITE-ENABLE GROOVE
ON TAPE REEL

WRITE-ENABLE RING
INSTALLED IN GROOVE

<u>TO INSTALL A WRITE-ENABLE RING</u>: -Place ring over groove.
-Push down until ring is flush with reel's surface.

Figure MAGNETIC TAPE-II: INSTALLING A WRITE-ENABLE RING

UNLOADING A TAPE (see Figure MAGNETIC TAPE - I)	
STEP 1	Press the ON LINE button. ON LINE light will extinguish.
STEP 2	Push the REWIND button. Tape will be wound onto the lower reel.
STEP 3	Pull out reel-locking lever and remove tape.
STEP 4	Remove the write-enable ring (if present) to preserve the information on the tape.
STEP 5	Push in locking lever and close door.

OTHER TAPE CONTROLS AND INDICATORS (see Figure MAGNETIC TAPE - I)	
HI DEN	This is the HIGH DENSITY button. When lit, your data is recorded at 1600 BPI (bytes per inch). When not lit, your data is recorded at 800 BPI. Push the button to light it. When lit, push the button to extinguish it. NOTE!: When you make a tape for use with a Gerber photo-plotter, you must record at 800 BPI (HI DEN light must be off). When making other tapes, you will probably want to record with the HI DEN button lit to save space on your tape.
FORWARD and REVERSE	Do not use these controls when using tape with the Telesis system.

3. ARCHIVE-TAPE CAPABILITIES

COPYING FILES TO TAPE

COPYING FILES FROM TAPE

Any files on the system can be copied to magnetic tape. You may copy all system files at once, or you may copy selected project files, drawing files, symbol files, text files, and/or net data base files.

Any files copied to tape can later be copied back to the system.

4. HOW TO GET TO THE ARCHIVE TAPE MENUS

To get to the ARCHIVE TAPE menus, perform the following steps.

NOTE

> You can access the ARCHIVE TAPE menus from the system, project, or drawing level.

1. Select the FILE MGMT/ARCHIV command from the system or drawing level. (Proceed to step 2 if you are at the project level).

   The MENU OF MENUS is displayed

2. Select the ARCHIVE-TAPE command

   The ARCHIVE TAPE menu is displayed

CAUTION

> Ensure that you have loaded a magnetic tape in the tape drive before performing the next step or the system will be inoperative until a tape is loaded.

3. Select NEW TAPE or OLD TAPE

   The COPY TO/FROM TAPE MENU is displayed

NOTE

> If you select NEW TAPE the system displays a message saying that any information on the tape will be deleted if you continue.

4. Create a list of files to be copied by making selections from the menu and answering system prompts

5. Select ENTER from the command line of the menu to begin the copy.

5. COPYING TO TAPE

5.1 COPYING ALL PROJECTS TO TAPE

|ALL FILES TO TP|

Pick this command to make a magnetic tape copy of all files currently on the system.

5.2 COPYING SELECTED FILES TO TAPE

These are the commands used for copying files to tape; they are shown here in the sequence in which you must use them:

|CUR PROJ TO TAPE|	
|PROJECT TO TAPE|	
|DRAWING TO TAPE|	
|COPY TO TAPE|	|SYMBOL TO TAPE|	|ENTER|
|TEXT TO TAPE|	
|NET-DB TO TAPE|	

Each command is described in detail in the Command Description Section of this manual. The overall process of copying selected files to tape is described below:

STEP 1 Begin with the |COPY TO TAPE| command.

STEP 2 You may follow the |COPY TO TAPE| command with any combination of files, for example:

|COPY TO TAPE| |DRAWING TO TAPE| |A|L|P|H|A|,|S|C|H|E|M| |ENTER|

|NET-DB TO TAPE| |A|L|P|H|A|,|ENTER| |TEXT TO TAPE| |A|L|P|H|A|,|

|E|X|T|R|A|C|T|I|O|N|-|L|O|G| |ENTER| |DRAWING TO TAPE| |B|E|T|A|,|

|S|C|H|E|M| |ENTER| |ENTER|

STEP 3 Use the |SHOW COPY LIST| command if you want to display the list of file names you have input. You may use this command as often as you like before ending your list with the final |ENTER| command. The copy list is displayed on the function screen.

4. HOW TO GET TO THE ARCHIVE-TAPE MENUS

As with the FILE MANAGEMENT menus and the FLOPPY DISK menus, you must be at the system level or the project level to use the ARCHIVE-TAPE menus.

- Use the |NEW PROJECT|, |OLD PROJECT| or |FILE MGMT/ARCHIV| commands to get to the MENU OF MENUS

- Function screen display:
  - MENU OF MENUS

- Pick |ARCHIVE-TAPE|

- Function screen display:
  - NEW/OLD TAPE MENU

- Pick |NEW TAPE| or |OLD TAPE| depending on whether the loaded tape already has data on it or not. When you pick |NEW TAPE| the system prompts you to be sure that you want to use this command.

  BE CAREFUL!: You may record over old data by using the |NEW TAPE| command. But before doing so, be sure that you no longer need any of the data on the tape because the entire tape will be erased.

- Function screen display:
  - COPY TO/FROM TAPE MENU

5. COPYING TO TAPE

5.1 COPYING ALL PROJECTS TO TAPE

|ALL FILES TO TP|

Pick this command to make a magnetic tape copy of all files currently on the system.

5.2 COPYING SELECTED FILES TO TAPE

These are the commands used for copying files to tape; they are shown here in the sequence in which you must use them:

|COPY TO TAPE| |CUR PROJ TO TAPE| |PROJECT TO TAPE| |DRAWING TO TAPE| |SYMBOL TO TAPE| |TEXT TO TAPE| |NET-DB TO TAPE| |ENTER|

Each command is described in detail in the Command Description Section of this manual. The overall process of copying selected files to tape is described below:

STEP 1 Begin with the |COPY TO TAPE| command.

STEP 2 You may follow the |COPY TO TAPE| command with any combination of files, for example:

|COPY TO TAPE| |DRAWING TO TAPE| |A|L|P|H|A|,|S|C|H|E|M| |ENTER|

|NET-DB TO TAPE| |A|L|P|H|A|,|ENTER| |TEXT TO TAPE| |A|L|P|H|A|,|

|E|X|T|R|A|C|T|I|O|N|-|L|O|G| |ENTER| |DRAWING TO TAPE| |B|E|T|A|,|

|S|C|H|E|M| |ENTER| |ENTER|

STEP 3 Use the |SHOW COPY LIST| command if you want to display the list of file names you have input. You may use this command as often as you like before ending your list with the final |ENTER| command. The copy list is displayed on the function screen.

STEP 4 Pick a final |ENTER| command (at the top of the COPY TO TAPE MENU) after the |ENTER| that follows the last file name on your list. The system will display a prompt asking if you want the tape verified. Pick |->PAGE| for "yes"; pick |CANCEL| for "no". If you pick |->PAGE| the tape will be copied and verified. If you pick |CANCEL| the tape will be copied but not verified.

5.3 GENERAL PRINCIPLES TO KEEP IN MIND

ORIGINAL IS UNALTERED

When you make a copy of a file, the original file remains unaltered.

COPIES HAVE THE SAME NAME AS ORIGINALS

When you copy a file, then tape copy will have the same name as the original.

REVISION LABELS

As with file names, a tape copy will have the same revision label as the original.

Whenever you copy a file, you may specify the revision label as well as the name of the original to be copied. If you do not, the system will copy the most recently used revision of the file you have named. Note that the "most recently used" revision is not always the same as the "most recently created" revision. For example, you might have created revisions 1, 2, 3, and 4 of drawing SCHEM in numerical order, but revision 3 may be the one that you looked at most recently. In this case, revision 3 is the "most recently used" revision.

To find out which revision of a file was the most recently used, look at the index where the file is listed. The revision listed closest to the bottom is the most recently used one.

SHORT CUTS

OMIT CURRENT PROJECT NAME--Whenever a command description calls for keyboard input of a project name, you can omit it if the project you wish to specify is the currently active project.

OMIT NET-DATA-BASE FILE NAME--You can always omit keyboard input of the name of the net data base file since the name is not always NET-DATA-BASE. If you wish to specify a revision, you may input the revision label alone in place of the name.

VERIFYING THE TAPE COPY

We recommend that you always instruct the system to verify the tape. In verifying, the system copies your file to tape, rewinds the tape, and goes through the entire copy comparing it with your original files.

DIFFERENCES BETWEEN THE COPY AND THE ORIGINAL

If the system finds differences between the copy and the original, it will report the file name and revision label where the difference occurs. When verification is finished, the system will report the total number of verification errors. The system displays these messages on the function screen.

TAPE ERRORS

If the system finds a problem with the tape itself, it will stop the verification process and report a tape error on the function screen.

Try using a new tape to copy your files. If you still receive a tape error message, have the tape drive checked.

6. TAPE SECTIONS: ADVANCING, REWINDING, AND INDEXES

ADV 1 TAPE SECT
ADV N TAPE SECTS
REWIND TAPE
TAPE SECT INDEX

Each command is described in detail in the Command Description Section of this manual.

Here is a general overview of the use of these commands:

TAPE SECTIONS

Each time you use the |ALL FILES TO TP| command and each time you complete the COPY TO TAPE process, you create an individual section of recorded data on the tape. Depending on how much data you have copied, the section may be small or large.

When you use the |ALL FILES TO TP| command or the COPY TO TAPE process again, you create another tape section that immediately follows the previous one. You may create as many sections as you like until you run out of tape on the reel.

DISPLAYING A TAPE SECTION INDEX

The system automatically records an index at the beginning of each section. Use the |TAPE SECT INDEX| command to display this on the function screen.

ADVANCING THE TAPE

You do not need to advance the tape when you are copying files to the tape. This is done automatically. Use the ADVANCE commands when you want to find a file that is already on the tape.

|ADV 1 TAPE SECT|--Use this command to advance to the next complete section on the tape.

|ADV N TAPE SECTS|--To advance more than one section, use this command followed by keyboard input of the number of sections.

When you use either of these commands, the system advances to the beginning of a tape section and displays the section index on the function screen.

REWINDING THE TAPE

Use the |REWIND TAPE| command to rewind the tape to the beginning of the first tape section.

7. COPYING FROM TAPE

You may only copy from one tape section at a time. So use the ADVANCE and REWIND commands to find the section you want before starting the copying process.

7.1 COPYING ALL PROJECTS FROM A TAPE SECTION

|ALL FILES FRM TP|

Pick this command to copy everything on the current tape section back to the system.

7.2 COPYING SELECTED FILES FROM THE CURRENT TAPE SECTION

These are the commands used for copying files from tape; they are shown here in the sequence in which you must use them:

|COPY FROM TAPE|	|CPY TO CUR PROJ|*	|SECT FROM TAPE|	|ENTER|
	or	|PROJ FROM TAPE|	
	|CPY TO ORIG PROJ|	|DRWNG FROM TAPE|	
		|SYMBOL FROM TAPE|	
		|TEXT FROM TAPE|	
		|NET-DB FROM TAPE|	

Each command is described in detail in the Command Description Section of this manual. The overall process of copying selected files from tape is described below:

STEP 1 Begin with the |COPY FROM TAPE| command.

STEP 2 Use the |CPY TO CUR PROJ| command or the |CPY TO ORIG PROJ| command to specify whether files are to be copied to their original projects or to the current project. (When you are at the system level, there is no current project per se, so you must use the |CPY TO ORIG PROJ| command only.)

If a copied item's original project no longer exists on the system, the system will create a project file with that name and place the copy in it.

* This command does not apply when you are at the system level.

STEP 3 As when you copy to tape, you may copy any combination of files from tape, for example:

|COPY FROM TAPE| |DRWNG FROM TAPE| |A|L|P|H|A|,|S|C|H|E|M| |ENTER|

|NET-DB FROM TAPE| |A|L|P|H|A|,| |ENTER| |TEXT FROM TAPE| |A|L|P|H|A|,|

|E|X|T|R|A|C|T|I|O|N|-|L|O|G| |ENTER| |DRWNG FROM TAPE| |B|E|T|A|,|

|S|C|H|E|M| |ENTER| |ENTER|

Or you may use the |SECT FROM TAPE| command to copy the entire section at once:

|COPY FROM TAPE| |CPY TO CUR PROJ| |SECT FROM TAPE| |ENTER|

(The |SECT FROM TAPE| command accomplishes the same thing as the |ALL FILES FRM TP| command except that it allows you to copy the whole section into the current project. With the |ALL FILES FRM TP| command, all files on the section are copied into their original project files.)

STEP 4 Use the |SHOW COPY LIST| command if you want to display the list of file names you have input. You may use this command as often as you like before ending your list with the final |ENTER| command. The copy list is displayed on the function screen.

STEP 5 Pick a final |ENTER| command (at the top of the COPY FROM TAPE MENU) after the |ENTER| that follows the last file name on your list. The files on your list are not copied until this final |ENTER|.

7.3 GENERAL PRINCIPLES TO KEEP IN MIND

TAPE COPY REMAINS UNALTERED

When you copy a file from the tape to the system, the tape file remains unaltered.

FILES ALREADY ON THE SYSTEM WILL NOT BE COPIED

The system will not copy a file from tape to the system if a file with the same name and revision number already exists on the system.

NAME CHANGES

NAMES OF DRAWING FILES, TEXT FILES, SYMBOL FILES AND NET-DATA-BASE FILES--These names cannot be changed when the files are copied from the tape to the system.

PROJECT NAMES--When you copy a file using the |CPY TO ORIG PROJ| command, the project file name of the system copy will be the same as the tape copy.

When you use the |CPY TO CUR PROJ| command, the project file name will change to that of the current project file.

REVISION LABELS

As with file names, revision labels cannot be changed when you copy a file from tape to the system, except when you use the |CPY TO CUR PROJ| command in which case the project file name and revision label change to that of the current project file.

You may specify the revision label of the tape file you want to copy. If you do not, the revision listed last on the tape section index will be the one copied.

SHORT CUTS

OMIT CURRENT PROJECT NAME--Whenever a command description calls for keyboard input of a project name, you can omit it if the project name you wish to specify is the same the current project name.

OMIT PROJECT NAME--Whenever a command description calls for keyboard input of a project name followed by an item name (drawing file, text file, symbol file, or net data base file), you can omit the project name if there is no other project on the current tape section containing in item file of the same name.

OMIT NET-DATA-BASE FILE NAME--You can always omit keyboard input of the name of the net data base file since the name is always NET-DATA-BASE. If you wish to specify a revision, you may input the revision label alone in place of the name.

TAPE ADVANCES AUTOMATICALLY TO THE NEXT SECTION.

Whenever you copy from tape, the tape automatically advances to the beginning of the next tape section. This is true even if you have copied only one item. So bear in mind that your current section has changed if you intend to copy from tape again or if you use the ADVANCE or INDEX commands.

8. ERASING FILES FROM A TAPE

When you no longer need the files you have copied to a tape, you can re-use the tape for making new copies by writing over the old data.

To do this, use the |NEW TAPE| command instead of the |OLD TAPE| command after loading the tape.

BE CAREFUL!!--Once you pick |NEW TAPE|, the tape will be formatted to accept new data only, you will not be able to retrieve any of the information that was previously on the tape. Be sure you know exactly what is on the tape and that you no longer need it.

9. RECOMMENDED USE OF MAGNETIC TAPE

You may use magnetic tape for the same purposes that you use floppy disks (see p. FLOPPY DISK-5):

A. To free space on the system for your current work

B. To back up your files

A. FREEING SPACE ON THE SYSTEM

In the Floppy Disk Section we recommended a series of steps for maximizing the available space on your system. You may use tape instead of disks in following these steps. Depending on your own preferences and needs, you will probably use some combination of the two.

B. BACKING UP YOUR FILES

In the Floppy Disk Section we also stressed the importance of making back-up copies of your files. These recommendations are important enough to repeat here:

COMPLETED WORK	Always make a back up copy when you have completed a drawing or project or series of library files. It's good practice when you have a completed board design, to keep two archival copies--one on-site and one off-site.
WORK-IN-PROGRESS	We recommend that you make a back-up copy of your work-in-progress on a daily basis.

Again, your own needs and preferences will determine when you use disks for back-up and when you use tape.

C R E A T I N G   T H E   A S C I I   T A P E

1. CREATING THE ASCII CODED MAGNETIC TAPE

2. TAPE FORMAT CHARACTERISTICS

3. COPYING TEXT FILES TO AN ASCII FORMATTED TAPE

4. COPYING ASCII CODED TEXT FILES TO THE TELESIS SYSTEM

1. CREATING THE ASCII CODED MAGNETIC TAPE (ASCII IN/ASCII OUT)

If you have the magnetic tape option on your Telesis system, you may transfer and share any number of existing text files with other computer systems.

The Telesis system allows you to communicate text files in an ASCII* format. ASCII code is the communication link between computers of different manufacture. ASCII compatible systems allow quick information exchange from one system to another.

The |ASCII TO TAPE| and |ASCII FROM TAPE| commands allow you to transfer, and accept ASCII coded text files stored on magnetic tape.

For example, the |ASCII TO TAPE| command creates a magnetic tape from the Telesis system that can be read by another computer. Your file is then available in a readable format to other agencies involved in your production process.

The |ASCII FROM TAPE| command allows you to accept ASCII coded text files to the Telesis system that are created by non-Telesis systems. For example, an ASCII coded TEXT NETLIST created by a system of different manufacture can be read by the Telesis system via the magnetic tape.

The following section outlines the ASCII tape format that is used when text files are exchanged between the Telesis system and another computer.

(ASCII) AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE

2. TAPE FORMAT CHARACTERISTICS

When you are interchanging ASCII coded text files stored on magnetic tape between a Telesis system and a non-Telesis system, the following ASCII tape format allows intercommunication between the two systems.

1. The tape is 9 tracks, odd parity, with a NO VOLUME label at the beginning.

2. The tape is written in 80-byte records in ASCII code. Each record represents one line of the text file.

3. Each record is spaced-filled to 80 bytes.

4. Each text file on a tape ends with an end-of-file mark.

5. The tape has two file marks at the end of the last file on the tape.

The |ASCII TO TAPE| command automatically formats the Telesis magnetic tape (ASCII) to these characteristics. If you transfer the contents of this tape to a non-Telesis system, be certain that this system can be programmed to read this ASCII tape format.

An ASCII coded magnetic tape created by a non-Telesis system must follow this ASCII format if it is to be read by a Telesis system. Use the |ASCII FROM TAPE| command to transfer ASCII text files to the Telesis system.

3. COPYING TEXT FILES TO AN ASCII FORMATTED TAPE

PREREQUISITES

1. Text files to be copied must be in the current project or in the SYSTEM-LIBRARY.

2. The tape drive must be powered-on.

3. The tape drive must be loaded; the ON LINE and WRT EN lights on.

STEPS TO FOLLOW

1. Pick |ASCII TO TAPE|.

2. Pick |ASCII FILE TO TAPE|, then input the name of the first text file you wish to copy when the system displays the keyboard menu.

3. Pick |ASCII FILE TO TAPE| to input the name of the second text file to be copied to this tape.

   Repeat this procedure until you have input all the text file names needed for this tape.

4. Use the |SHOW COPY LIST| command to display all the file names you have input.

5. Pick |ENTER| to copy your selected files to the magnetic tape. The system also verifies each file as it is copied to tape. The function screen message line displays all verification messages.

6. Pick |DONE| before removing the tape from the tape drive.

NOTE: ASCII formatted tapes do not have indexes or file names. It is advisable that you attach a label to the tape reel to identify the ASCII files and their sequence on the tape. (See FILE/ARCH-45, ASCII FROM TAPE).

4. COPYING ASCII CODED TEXT FILES TO THE TELESIS SYSTEM

PURPOSE

You may copy ASCII coded text files stored on magnetic tape back to the Telesis system. ASCII text files created by non-Telesis systems can also be copied to a Telesis system.

If you are copying a Telesis created magnetic tape back to a Telesis system, use the |ASCII FROM TAPE| command. Because the tape was originally written in ASCII code, it must be copied back with the |ASCII FROM TAPE| command.

You may also use the |ASCII FROM TAPE| command to copy a non-Telesis tape to a Telesis system. The tape must be ASCII formatted and compatible with the Telesis system.

PREREQUISITES

1. The system must be in the project name where the files are to be copied.

2. The tape drive must be loaded, powered-on and on-line,

STEPS TO FOLLOW

1. Pick the |ASCII FROM TAPE|.

2. You may copy one file or all files; you may not copy selected files from the tape without repeating the |ASCII FROM TAPE| command.

   Input the number of the file to be copied to the current project, based on its sequence on the tape. Skip the keyboard input and pick |ENTER| to copy all of the text files on the tape to the system.

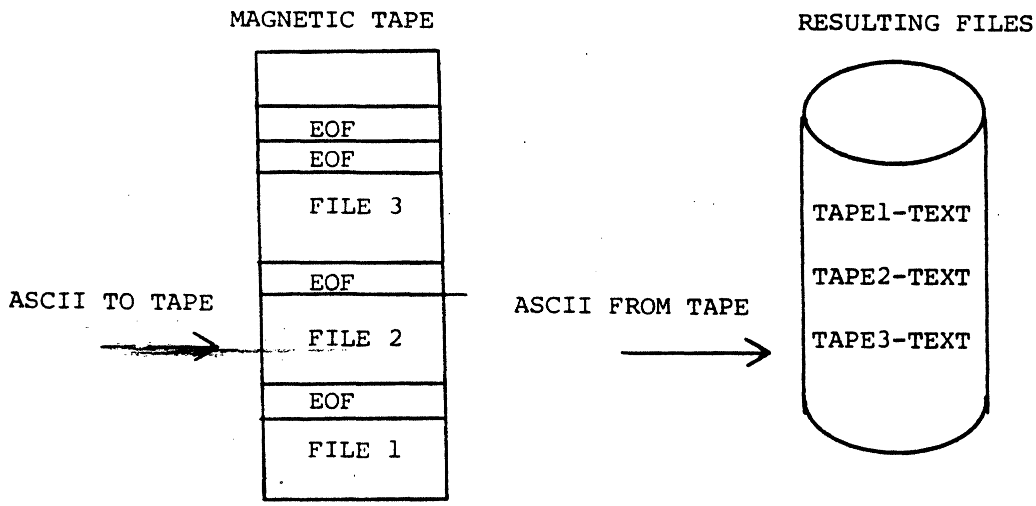
   ASCII formatted tapes do not have indexes, and the files do not have file names. Therefore, the tape must be labeled in order to determine the sequence of text files on tape.

   For example, if you are provided with a written record of all text files stored on the tape, you may select the files you wish to copy. The sequence of files on the tape determines the number you would input to select a particular file. If you input the number 3, the third file that is stored on the tape will be copied to the current project.

As the system copies ASCII coded text files to the current project, it gives each file a name based on its sequence on the tape. For example, the first file on the tape is named "TAPE1-TEXT". The second file on the tape is named "TAPE2-TEXT".

(Refer to the COMMANDS section of this manual for additional information about the |ASCII TO TAPE| and |ASCII FROM TAPE| commands.)

EXAMPLE: ASCII TO TAPE and ASCII FROM TAPE



L I B R A R Y

- HOW TO USE A LIBRARY
- SYMBOL FILES
- PIN FILES
- DRAWING FORMATS
- DEVICE FILES

HOW TO USE A LIBRARY

1. PURPOSE OF A LIBRARY

DEFINITION OF A LIBRARY

A library is a project file in which you keep symbol files, text files and any other files that you plan to use repeatedly in the creation of board designs.

Having a library eliminates repetitive work. For example, you can draw an AND symbol once, and then use the ADD SYMBOL commands to place hundreds of instances of the symbol in schematic drawings without ever having to draw the symbol again.

REQUIRED LIBRARY FILES

To create schematic and printed circuit board drawings using the full capabilities of the Telesis systems, you must have a library containing:

- LOGIC SYMBOLS
- BOARD SYMBOLS
- PIN DESCRIPTION TEXT FILES
- DEVICE DESCRIPTION FILES

OPTIONAL LIBRARY FILES

You may also create and keep library copies of other information that you expect to use repeatedly, for example:

- YOUR STANDARD COMPANY DRAWING FORMAT

YOU MAY USE THE TELESIS-PREPARED LIBRARY OR CREATE YOUR OWN

The Telesis-prepared library of symbols, pin files, and device files is shown in Volume 2 of this manual. You may use these symbols and pin files or you may create your own to conform to your company's standards.

2. THE SYSTEM-LIBRARY

DEFINITION OF THE SYSTEM-LIBRARY

SYSTEM-LIBRARY is the name of the project file in which you ordinarily keep library files. The system is programmed to look for the files it needs in a project file named SYSTEM-LIBRARY.

For example, when you pick the ADD SYMBOL commands, followed by a symbol name, the system searches for a symbol file by that name. The system can only search in two places: the currently open project file and a project file named SYSTEM-LIBRARY. It cannot search in an unopen project file by any other name.

The same is true of other operations. For example, when you use the physical design rules checking commands, the system searches for pin description files in only the current project file or in the SYSTEM-LIBRARY.

KEEP YOUR LIBRARY FILES IN THE SYSTEM-LIBRARY PROJECT FILE

When you are using library files, we recommend that you create a library project file named SYSTEM-LIBRARY and keep your library files in it, rather than keeping them in the current project file. We recommend that you keep only files unique to a single board design, such as the net data base and the board drawing, in the current project file. This practice will simplify the task of keeping track of your files.

FILE MANAGEMENT AND ARCHIVING OF THE SYSTEM-LIBRARY

For file management and archiving purposes, the SYSTEM-LIBRARY is like any other project file. It can be copied, deleted, and stored on tape or floppy disks; and the files in it can be copied, deleted, and stored in the usual way. (See the File Management and Archive Sections of the manual.)

Ordinarily, you will only be using a fraction of your library files at any one time. For example, when you are working on a board drawing, you have no need for logic symbols. You may copy the files you need back and forth from floppy disks or tape in order to keep only the ones you need on the system at any one time. You should do this so as to keep as much free disk space as possible on the system.

CREATING YOUR OWN SYMBOL FILES

1. GENERAL INFORMATION

If you choose to create your own symbol library, you should create a symbol file for every logic symbol and board symbol your company customarily uses in designing boards.

SUMMARY OF STEPS TO FOLLOW IN CREATING A SYMBOL FILE

These are the steps to follow in creating a symbol file:

STEP 1 Open a new drawing file.

STEP 2 Draw the symbol referring to the symbol requirements for the type of symbol you are drawing (P. LIBRARY-6-11).

STEP 3 Use the |CREATE SYMBOL| command to create a symbol file from the drawing file.

STEP 4 Print out the -LOG text file created by the |CREATE SYMBOL| command, and check the symbol for accuracy.

STEP 5 If the symbol was not created due to errors, or if you find mistakes in the -LOG File, correct the symbol drawing and repeat the |CREATE SYMBOL| command.

STEP 6 Repeat STEPS 4 and 5 until an acceptable symbol file has been created.

STEP 7 We recommend that you store the original drawing file on tape or floppy disk.

2. REQUIREMENTS FOR LOGIC SYMBOLS

There are two types of logic symbols having different requirements on the Telesis system:

1. GENERAL LOGIC SYMBOLS

2. SIGNAL-TIE SYMBOLS

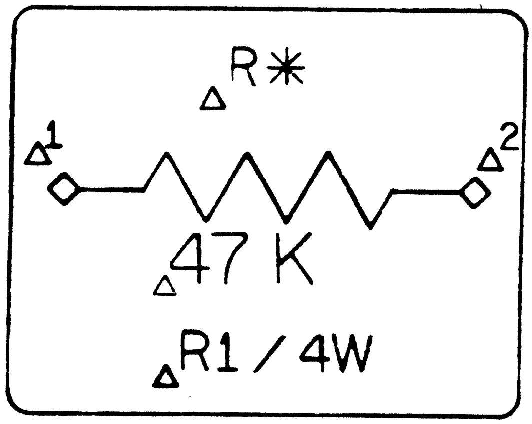
GENERAL LOGIC SYMBOLS

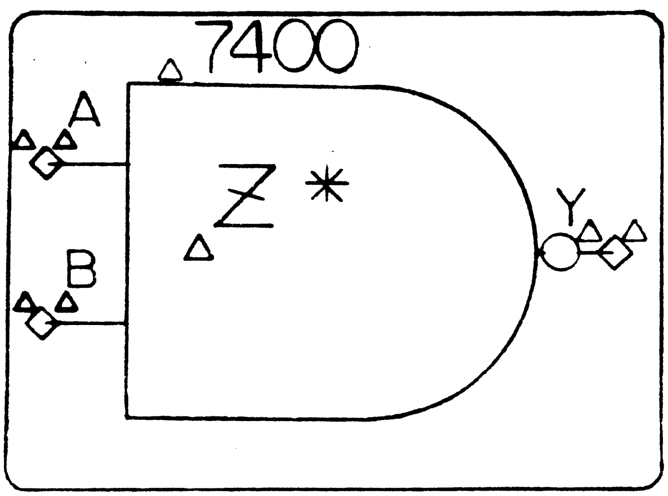
Most logic symbols are of this type. Examples are AND gate, NAND gate, transistors and resistors.

TABLE OF REQUIREMENTS FOR GENERAL LOGIC SYMBOLS

ELEMENT	REQUIRED	LOCATION	COMMAND	COMMENTS
Connect point	At least one	Layer 0 Pins on grid	ADD CONNECT PNT	Put a connect point at each position on the symbol where you will be making electrical connections. Connect points on logic symbol correspond to physical pins on your pc board. Each connect point with a pin number label is considered a pin by the system. The EXTRACT NETLIST command creates a pin entry in the net data base for that pin.
Pin number label	One for each connect point	Layer 53 to be shown Layer 93 not to be shown	ATTCH PIN NUMBER	
Pin name label	Optional	Layer 151 to be shown Layer 85 not to be shown	ATTCH PIN NAME	
Reference designator label	One	Layer 54	ADD REF DESIG	This is the name of the corresponding physical component on the pc board. You can put a dummy text on the label (such as R* for resistors); and later use the UPDATE TEXT command on the schematic drawing to set the actual reference.
Device type label	Optional	Layer 55	ADD DEVICE TYPE	If you do not input a device type label, the system will use the symbol file name as the device type name.
Value label	Optional	Layer 152	ADD VALUE	

EXAMPLES OF GENERAL LOGIC SYMBOLS





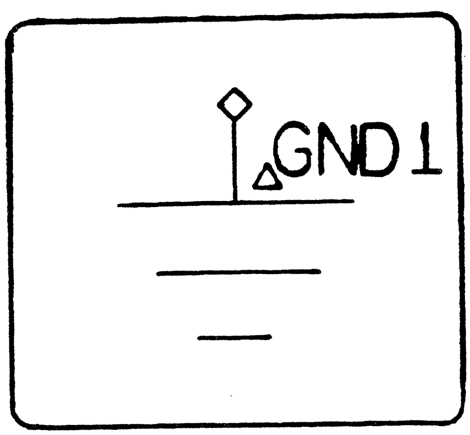
SIGNAL-TIE SYMBOLS

These are the symbols that you use simply to terminate a connect line, and, if you wish, to tie the symbol to a particular net name. Examples are a ground symbol, or an off-page or on-page connect bubble.

TABLE OF REQUIREMENTS FOR SIGNAL-TIE SYMBOLS

ELEMENT	REQUIRED	LOCATION	COMMAND	COMMENTS
Connect point	At least one	Layer 0	ADD CONNECT PNT	Put a connect point at each position on the symbol where you will be making electrical connections. These connect points may not have pin number labels or pin name labels.
Signal name label	Optional	Layer 64 to be shown Layer 86 not to be shown	ATCH SIG NAME PN	You can make ground symbols and on/off page connectors by attaching a signal name to the connect point of a signal-tie only symbol.
Reference designator label	Prohibited			
Pin number label	Prohibited			
Pin name label	Prohibited			

EXAMPLE OF A SIGNAL-TIE SYMBOL



3. BOARD SYMBOLS

There are two types of board symbols having different requirements on the Telesis system:

1. COMPONENT SYMBOLS
2. VIA SYMBOLS

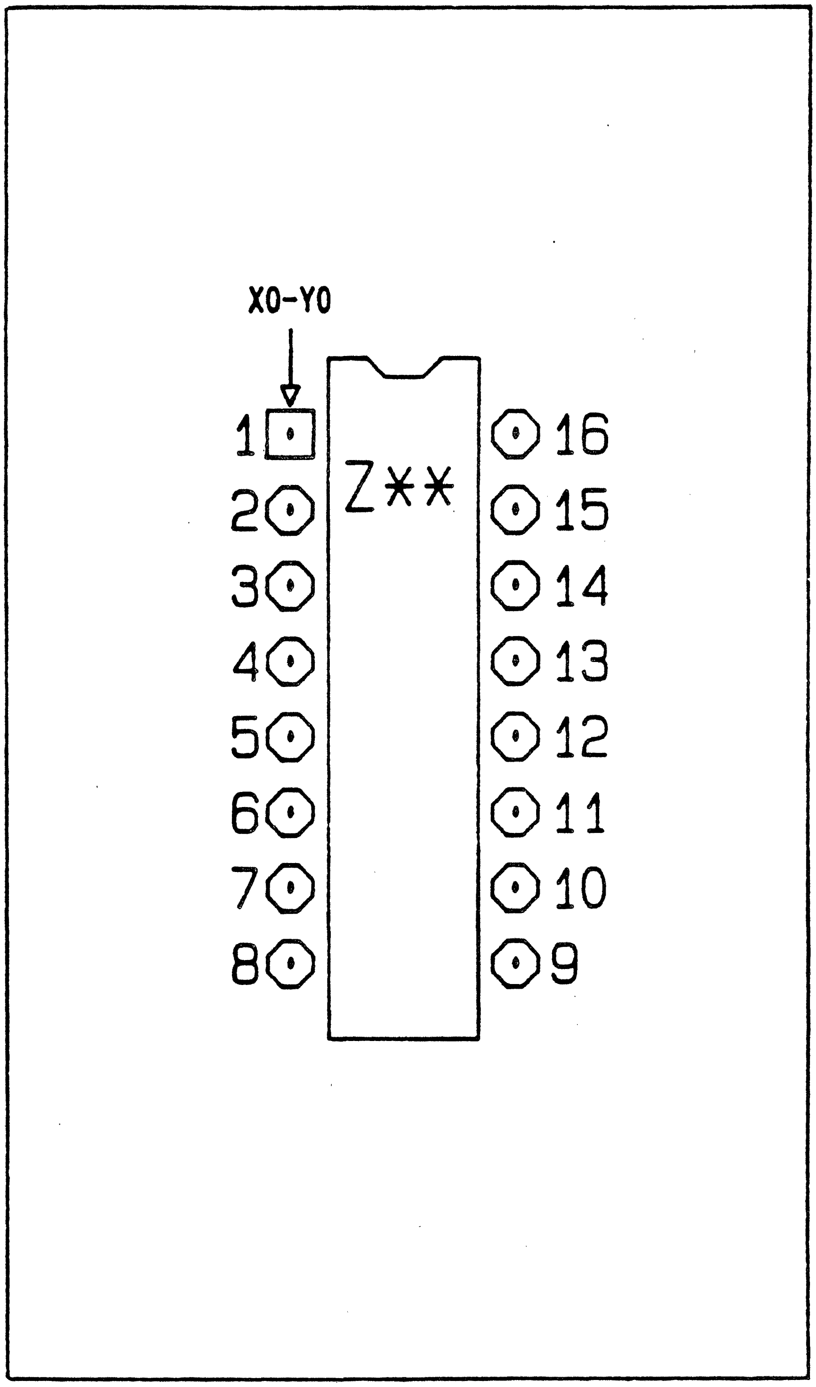
COMPONENT SYMBOLS

Component symbols represent physical packages on the board.

TABLE OF REQUIREMENTS FOR COMPONENT SYMBOLS

ELEMENT	REQUIRED	LOCATION	COMMAND	COMMENTS
Connect point	At least one	Layer 0 Pins on grid	ADD CONNECT PNT	Put a connect point at each position on the system where you will be making electrical connections. Each connect point with a pin number label is considered a pin by the system
Pin number label	One for each connect point	Layer 53 to be shown Layer 93 not to be shown	ATTCH PIN NUMBER	
Pin name label	Optional	Layer 151 to be shown Layer 85 not to be shown	ATTCH PIN NAME	
Reference designator label	One	Layer 54	ADD REF DESIG	You can put a dummy text on the label (such as R* for resistors). Later you will use the ASSIGN REF DES command on all board symbols.
Device type label	Optional	Layer 55	ADD DEVICE TYPE	If you do not input a device type label, the system will use the symbol file name as the device type name.
Value label	Optional	Layer 152	ADD VALUE	

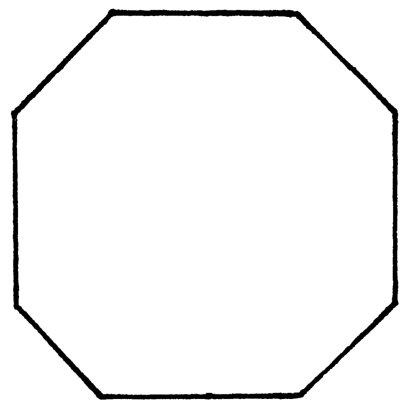
EXAMPLE OF COMPONENT SYMBOL



VIA SYMBOLS

Via symbols represent through holes on the board drawing. A through hole is used to route a connect line from one board layer to another. Via symbols are the only symbols that are not added to a board drawing using the ADD SYMBOL command. A via symbol is added to a board drawing by using the DRILL command during an ADD CONNECTION command. If a via symbol file does not exist in the current project or in the SYSTEM-LIBRARY, a DRILL command will not work.

The following figure illustrates a typical via symbol.



CREATING A VIA SYMBOL

To create a via symbol, perform the following steps.

Select : : SET ACTIVE LAYER from the SYMBOL DRAWING menu

The following system prompt is displayed below the command line of the menu :

Layer :

Enter : : Ø
Press or
Select : : ENTER

Select : : ADD CONNECT PNT

The system displays the CONNECT POINT menu :

Select : : A PAD=command to specify the shape of the via symbol (DO NOT use PAD=POINT or the via symbol may not be visible on the graphics screen) (Using PAD=CIRCLE will increase screen repaint time)

The following system prompt is displayed below the command line of the menu :

Size :

Enter : : A number to specify connect point size
Press or
Select : : ENTER

NOTE

Ensure that the connect point size specified is small enough to fit within the drawing extents (i.e., if drawing extents are : - 25 and + 25 in both the X and Y directions; the connect point size must be 50 or smaller).

The system flips to graphics mode :

Select : : The X = 0, Y = 0 coordinate of the drawing

The system places a highlighted connect point at X = 0, Y = 0 on the graphics screen :

Select : : ENTER from the command line of the menu

The system dehighlights the connect point and adds it to the symbol drawing. The system redisplays the CONNECT POINT menu :

Select : : DONE from the command line of the menu

The system displays the SYMBOL DRAWING menu :

Select : : CREATE SYMBOL

The following system prompt is displayed below the command line of the menu :

Symbol name :

Enter : : Symbol name
Press or
Select : : ENTER

The system creates the via symbol. If the system found errors in the symbol drawing, the symbol will not be created and an error message will be displayed on the screen.

4. DRAWING THE SYMBOL

MAKE A DRAFT COPY OF THE SYMBOL

Use a sheet like the ones shown in Figure SYMBOL FILES-I and Figure SYMBOL FILES-II to make a hand-drawn draft copy of your symbol. Keep the draft copy for reference.

DRAWING AND EXTENTS

Draw your symbol in the same standard size in mils that you want it to be in a schematic or board drawing. When you use the ADD SYMBOL commands, the system will place it in your schematic or board at this size. (You may then use the |SCALE SYMBOL| command if you want to make this instance of the symbol larger or smaller.

Use the |OTHER SIZE| command to set the drawing extents. The size of the drawing extents does not matter because the system will use only the area immediately enclosing the symbol graphics to create the symbol. However, the placement of the origin of your drawing does matter. We recommend that you always make the lower left connect point of the symbol the 0,0 point of your drawing. The 0,0 point of a symbol drawing will become the reference point of the symbol when it is added to a schematic or board. If you consistently place the lower left connect point on 0,0, you will always be aware of where its reference point is. This is important to know when you scale, move, rotate, or copy a symbol in a schematic or board drawing. Even if you do not place the lower left connect point on 0,0, do not place the symbol graphics far from 0,0 for your own convenience when adding the symbol to a drawing.

GRID

We recommend that you use a 200 by 200 mil grid so that your connect points will be at 200 mil intervals. This will simplify interconnection when you place the symbol in a schematic or board drawing where you are using a 200 or 100 mil grid, particularly if you scale any instances of the symbol to 1/2 size after adding them to a drawing.

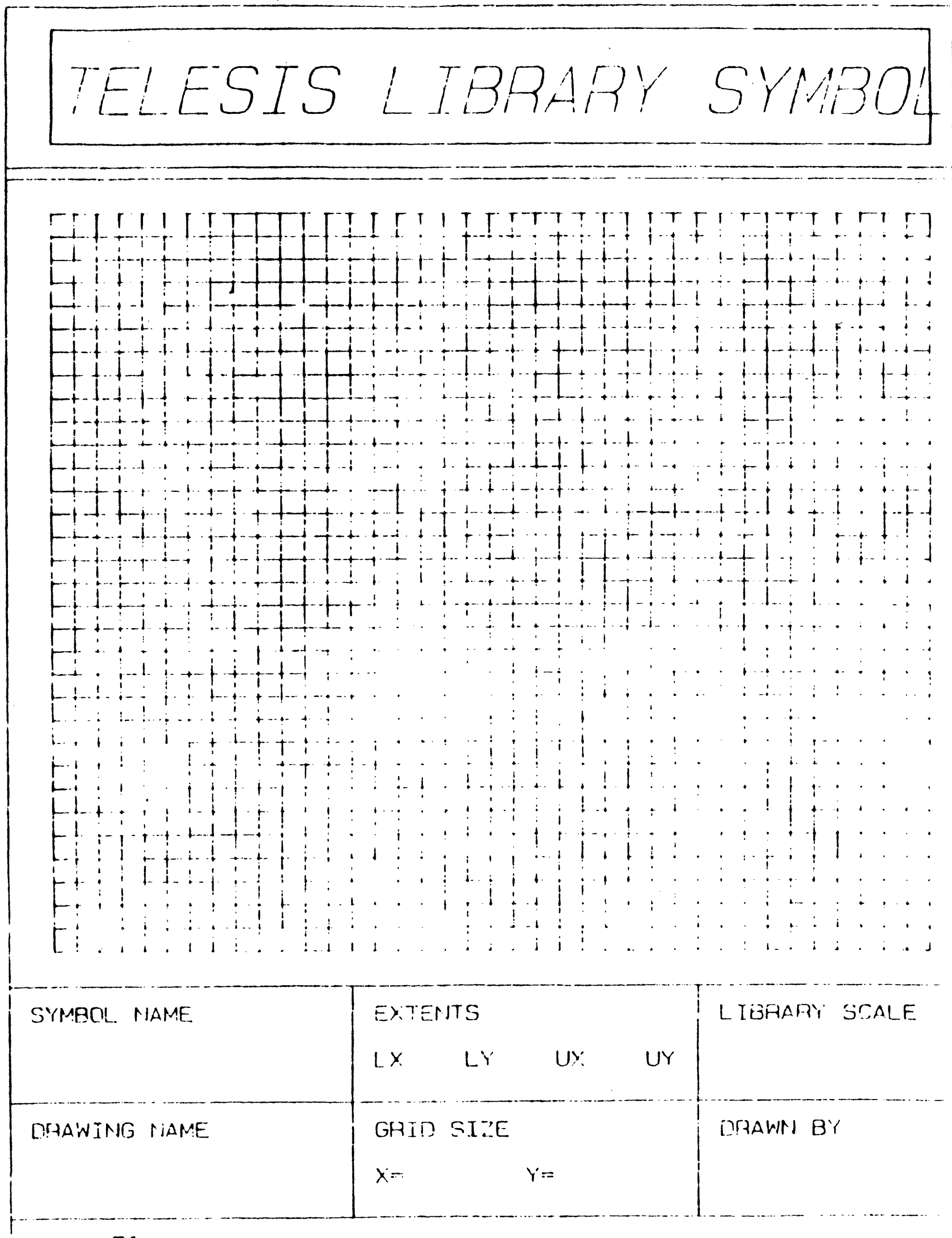


Figure SYMBOL FILES-I: Sample format for draft copy of a symbol

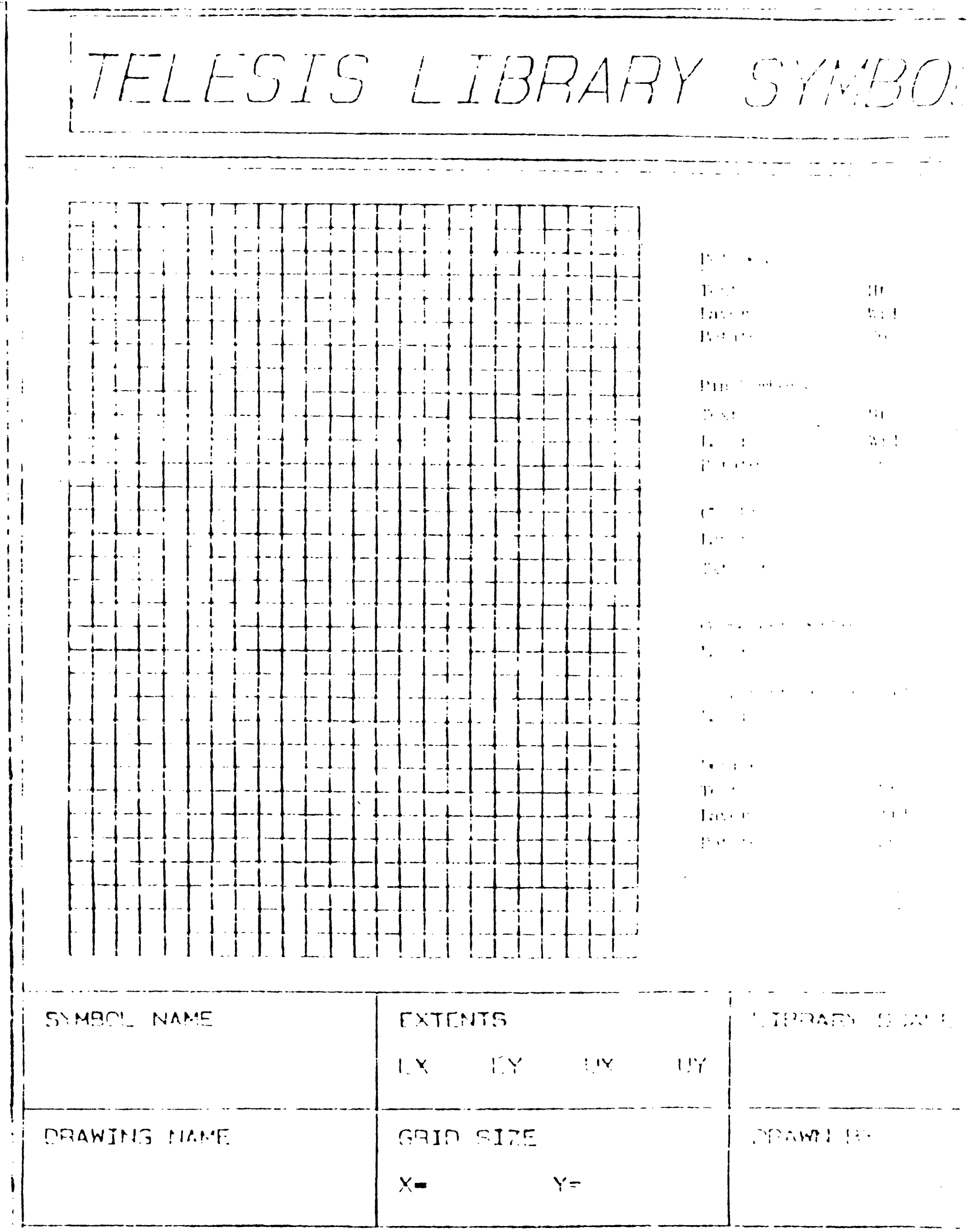


Figure SYMBOL FILES-II: Sample format for draft copy of a symbol

LAYERS

Refer to the Telesis Recommended Layer Standards Table (p. BASIC-30) when adding the elements of your symbol drawing. The layer placements you make here will be maintained when you place the symbol in a schematic or board drawing.

LABEL

Some labels are required on symbol drawings and some are not, depending on the type of symbol (See the Sections on Logic Symbols and Board Symbols.)

When you add a label, you may input the label text or you may leave the text blank (the text point only will show). If you input a blank label usin the label commands, you may add the text after placing the symbol in a schem or board drawing. Be aware, however, that you may only use that blank text point for the type of label originally specified when you added the text poi For example, if you used |ADD REF DESIG| to add a blank label, you may only use this text point later for reference designators.

ELEMENTS

Use lines, arcs, circles, and connect points for symbol graphics. Do not use connect lines or connect arcs.

5. USING THE |CREATE SYMBOL| COMMAND

When you have completed your symbol drawing, pick |CREATE SYMBOL|.

The Command Description Section of the manual gives you the exact input sequence to use.

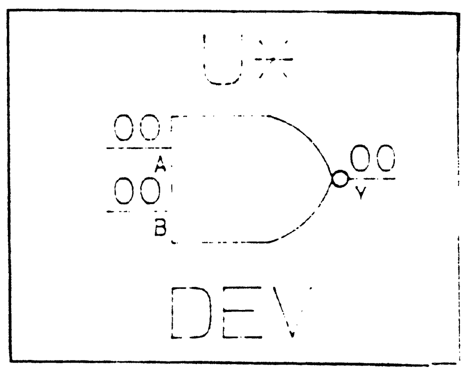
THE SYMBOL - LOG FILE

When you pick |CREATE SYMBOL|, the system creates a text file reporting on the lines, arcs, labels, connect points and other elements of the symbol you have drawn.

The system gives this text file the name of the symbol file with the suffix "-LOG" added. For example, if you have named the symbol file "NAND2", the system will name the corresponding report file "NAND2-LOG."

NOTE: The system creates a -LOG file even if errors have prevented the creation of a symbol file.

EXAMPLE OF A SYMBOL AND ITS -LOG FILE




```
SYMBOL CHECKED   NO DATA BASE ERRORS DETECTED



*******************************************************************************
*                    TELESIS  CREATE  SYMBOL  ELEMENT  REPORT                 *
*******************************************************************************

symbol name   NAND2               revision 1
==========================================
```


(example continued on next page)

(Example continued from previous page)

```
++++++++++++++++ CONNECT POINTS  (P) and attached LABELS  (T) ++++++++++++++

                      attached LABEL                     TEXT parameters
      X      Y    LAYER   # / type        LABEL-TEXT      hgt   wdt   spc
   ------ ----- -----  --  -----------  ----------      ----  ----  ----

P      0      0      0

T     25     25     53    4/  PIN-NUMBER   00       0 080  0 060  0 020
T    150    -75     85   14/  PIN-NAME     B        0 050  0 030  0 010
==========================================================================

P      0    200      0

T     25    225     53    4/  PIN-NUMBER   00       0 080  0 060  0 020
T    150    125     85   14/  PIN-NAME     A        0 050  0 030  0 010
==========================================================================

P    900    100      0

T    760    125     53    4/  PIN-NUMBER   00       0 080  0 060  0 020
T    775     25     85   14/  PIN-NAME     Y        0 050  0 030  0 010
==========================================================================

     TOTALS        3 CONNECT POINTS         6 LABELS




++++++++++++++++++++ LABELS attached to the SYMBOL ++++++++++++++++++++++

                                                   TEXT parameters
    X       Y     LAYER    # / type        LABEL-TEXT      hgt   wdt   spc
  ------  ------  -----   --  ----------  ----------     ----- ----  ----

   300     400     54       2/  REF-DES     U*          0 156  0 117  0 039
==========================================================================

   200    -400     55       5/  DEVICE-TYPE DEV         0 156  0 117  0 039
==========================================================================

     TOTALS       2 LABELS attached to symbol
==========================================================================
```

(Example continued on next page)

(Example continued from previous page)

```
++++++++++++++++++++++++++++++++  LINES  ++++++++++++++++++++++++++++++++

      X         Y      LAYER   WIDTH
   -------   -------   -----   -----

      0        200      52       0
    200        200
=========================================================================

      0          0      52       0
    200          0
=========================================================================

    500       -100      52       0
    200       -100
    400        300
    500        300
=========================================================================

    750        100      52       0
    900        100
=========================================================================

    TOTALS        4 LINES
=========================================================================


++++++++++++++++++++++++++++++++  ARCS  +++++++++++++++++++++++++++++++++

      CENTER                                          ANGLE (degrees
     X        Y      RADIUS   LAYER     WIDTH          start      end
   -------  -------  ------   -----     -----         -------   ------

    413      188      301      52         0            -[illegible]       -17
=========================================================================

    425       25      285      52         0             15         75
=========================================================================

    725      100       25      52         0              0          0
=========================================================================

    TOTALS        3 ARCS
=========================================================================



*************************************************************************
**********************  END OF SYMBOL ELEMENT REPORT  *******************
*************************************************************************
```

EXPLANATION OF TERMS USED IN THE -LOG FILE

As you can see from the example above, the -LOG file is largely self-explanatory. Below are the items that may need explanation:

P	The letter "P" in the left hand column means connect point.
T	The letter "T" in the left hand column means a text point associated with the connect point (P) listed above it.
Circles	Circles are not listed separately in the -LOG. They are shown as arcs with a "start" and "end" angle of 0.
Label #	Label # is listed to the left of label type. Label # is merely another way of referring to label type. For example, a reference designator is always label #2; a signal name is always label #1, etc.

ERROR MESSAGES

When you pick |CREATE SYMBOL|, the system checks your symbol for certain kinds of errors. Below is a list of error and warning messages the system wil report either on the function screen or in the -LOG file. (NOTE that the system checks VIA symbols much more thoroughly than other types of symbols.)

- SYMBOL LIMIT EXCEEDED, 512 ELEMENTS IS THE MAXIMUM ALLOWED

  The system has found more than the maximum number of allowed elements in your symbol. The symbol file will not be created.

- SYMBOL LIMIT EXCEEDED, 256 (textpoints) is the maximum allowed

  The system has found more than the maximum allowed number of textpoint in your symbol. The symbol file will not be created.

- ERROR: SYMBOL CONTAINS MORE THAN 200 CONNECT POINTS

  The system has found more than the maximum allowed number of connect points in your symbol. The symbol will not be created.

- WARNING: CONNECT LINES IN SYMBOL

  Ordinarily you should not have connect lines in a symbol. The system issues a warning, but it will create the symbol file.

- ERROR: CONNECT LINE HAS ZERO LENGTH. SYMBOL WILL NOT BE CREATED

  The system has found a connect line with no length. The symbol will not be created.

- CANNOT ALLOW SYMBOL WITHIN SYMBOL

  The system has found a symbol placed within your symbol drawing. It will not create the symbol file.

- SYMBOL ALREADY EXISTS. PICK -> PAGE TO OVERWRITE, ELSE CANCEL

  The name you have given this symbol file has already been used for an existing symbol file. Pick |-> PAGE| to input a new name, or |CANCEL| to cancel the |CREATE SYMBOL| command.

- ERROR DURING CREATE SYMBOL VIA: VIA MUST HAVE ONE AND ONLY ONE CONNECT POINT

  The system has found fewer than one, or more than one, connect point in a VIA symbol drawing. The symbol file will not be created.

- ERROR DURING CREATE SYMBOL VIA: VIA MUST HAVE CONNECT POINT ON LAYER ZERO

  The connect point in a VIA symbol drawing is not on layer zero. The system will not create the symbol file.

- ERROR DURING CREATE SYMBOL VIA: VIA MUST NOT HAVE CONNECT LINES

  The system has found a connect line in a VIA symbol drawing. It will not create the symbol file.

- ERROR DURING CREATE SYMBOL VIA: VIA MUST NOT HAVE TEXT POINTS ATTACHED TO CONNECT POINT

  The system has found one or more text points attached to the connect point in a VIA symbol drawing. The symbol file will not be created.

- ERROR DURING CREATE SYMBOL VIA: VIA CONNECT POINT MUST BE AT THE ORIGIN

  The system has found that the connect point in a VIA symbol drawing is not at the origin (0,0) of the drawing. The symbol file will not be created.

- <u>ERROR DURING CREATE SYMBOL VIA: VIA MUST NOT HAVE REFERENCE DESIGNATOR LABEL</u>

  The system has found a reference designator label in a VIA symbol drawing. The symbol file will not be created.

CHECK THE -LOG FILE FOR ACCURACY

Even if the -LOG file shows no errors, you must check it carefully to be sure that you have met the requirements for the type of symbol you are creating (especially if the symbol is not a VIA). Here are some questions to keep in mind as you review the -LOG file.

- Have you placed the required labels on your symbol?
- Have you placed all elements on the proper layers?
- Have you placed the origin (0,0) of your symbol properly?

RE-USING THE ORIGINAL SYMBOL DRAWING FILE

We recommend that you do not delete the original symbol drawing file after creating the symbol file with the |CREATE SYMBOL| command.

Symbol files cannot be edited, but drawing files can. If, for any reason, you wanted to alter a symbol, you could edit the original drawing and use the |CREATE SYMBOL| command again, instead of having to draw the symbol all over again.

You may also save some time by using the drawing file for one symbol as the basis for drawing another symbol. For example, you could draw a DIP14 symbol more quickly by editing a DIP16 drawing than by making an entirely new drawing.

P I N F I L E S

1. GENERAL INFORMATION

2. CREATING THE PIN DESCRIPTION TEXT FILE

3. EDITING THE PIN DESCRIPTION FILE

4. PIN DESCRIPTION FILES NOT CREATED WITH THE EDIT PIN FILE LEAD-THRU

1. GENERAL INFORMATION

PURPOSE OF PIN DESCRIPTION TEXT FILES

A pin description text file must exist for every symbol used on a PC board design. Pin files allow you to control pad sizes and shapes by pin. It also allows you to assign the physical layer assignment and the plating information for each pintype.

Pin files contain information that is required by the system for:

1. Physical Design Rules Checking (PDRC)
2. Artwork Generation
3. NCDrill
4. Router

When you use these system capabilities, the system searches for the needed pin files, and then checks for the required information. You will receive an error message if the pin files do not exist, or if they do not contain the needed information.

EXAMPLE: PIN DESCRIPTION FILE FORMAT

```
                 DIP14-PIN     1
(DIP14-PIN)
(PIN FILE FOR SYMBOL DIP14)
PINTYPE  A
DRILL .035-P
PAD CIRCLE=.040 DRILL-MASK
PAD SQUARE=.062 COMPONENT-SIDE
PAD SQUARE=.062 SOLDER-SIDE
PAD CIRCLE=.080 COMPONENT-SOLDER-MASK
PAD CIRCLE=.080 SOLDER-SOLDER-MASK
PAD CIRCLE=.040 INTERNAL-SIGNAL
THERMAL-RELIEF FLASH=AB12 IMBEDDED-PLANE
ANTI-PAD CIRCLE=.040 IMBEDDED-PLANE
PINTYPE B
DRILL .035-P
PAD CIRCLE=.040 DRILL-MASK
PAD CIRCLE=.062 COMPONENT-SIDE
PAD CIRCLE=.062 SOLDER-SIDE
PAD CIRCLE=.080 COMPONENT-SOLDER-MASK
PAD CIRCLE=.080 SOLDER-SOLDER-MASK
PAD CIRCLE=.040 INTERNAL-SIGNAL
THERMAL-RELIEF FLASH=AB12 IMBEDDED-PLANE
ANTI-PAD CIRCLE=.040 IMBEDDED-PLANE
PIN  1  A
PIN  2-14  B
END
```

NAME AND LOCATION OF PIN FILES

You must give the pin file exactly the same name as the board symbol file it describes, with the suffix "-PIN" added to the symbol name.

EXAMPLE: Symbol file name: DIP14

Corresponding pin file name: DIP14-PIN

Pin files may be placed in the current project file or in the SYSTEM-LIBRARY We recommend that you keep them in the SYSTEM-LIBRARY.

CONTENTS OF PIN FILES

Pin files contain the following information:

1. PINTYPE
2. DRILL SIZE AND PLATING INFORMATION FOR EACH PINTYPE
3. PAD SIZE, SHAPE AND LAYER FOR EACH PINTYPE
4. ANTI-PAD INFORMATION
5. THERMAL RELIEF INFORMATION
6. DEFINITION OF WHICH SYMBOL PINS BELONG TO EACH PINTYPE

GENERAL PRINCIPLES

1. All symbols on the PC board must have pin files.

2. Pin files should be created so that they can be used with all boards; and for boards with multiple internal power and signal layers. When using pin files, the system ignores any information it does not need during the creation of a board. You should create pin files to suit your maximum board designs.

3. All sizes specified in the pin file(s) require an entry in the APERTURE-TAB file when you generate artwork or perform NCDrill. If you use too many different sizes, you may have difficulty setting up an APERTURE-TAB file for a photoplotter aperture wheel. (Most aperture wheels can hold a maximum of 24 different aperture sizes.)

2. CREATING THE PIN DESCRIPTION TEXT FILE

The |EDIT PIN FILE| lead-thru allows you to create and edit pin files. It reduces the possibility of numerous format errors that could be generated when a pin file is created on a text editor keyboard.

When you create pin description files for your PC board designs, the Telesis system assists you in the following ways:

1. Automatically positions the cursor to the proper position.

2. Automatically inputs keywords required in the file.

3. Places parentheses () around comment lines.

4. Issues error messages if your input was improperly formatted.

   NOTE: When opening an existing pin description text file, error messages may appear on the function screen if your file was not created with the |EDIT PIN FILE| lead-thru. (See)

The following steps will lead you through the creation of the pin description text file:

STEP 1. |TEXT LEADTHRU|

Pick the |TEXT LEADTHRU| command to enter the selected menu page containing options for creating and editing text files.

STEP 2. |EDIT PIN FILE|

Pick the |EDIT PIN FILE| command to begin the creation of the new pin description file. The system will prompt you for a file name. The file name may contain up to 18 alphanumeric characters (no blank spaces) with the -PIN suffix included.

EXAMPLE: |EDIT PIN FILE| (FILE NAME) DIP14-PIN |ENTER|

The system displays the DIP14-PIN text file on the graphics screen as a new file. The display will have the information shown below.

```
            DIP14-PIN
  (DIP14-PIN)
  END
```

You may use the following commands to add information between (DIP14-PIN) and the END statement.

|ADD PINTYPE|
|ADD DRILL|
|ADD PAD|
|ADD THERM-RELIEF|
|ADD PIN LINE|
|ADD COMMENT|

STEP 3. |ADD PINTYPE|

In some cases, a board symbol has only one pintype. All pins assigned to that symbol have the same drill, pad, anti-pad and thermal relief specifications. In other cases, a board symbol may have more than one pintype. Different pins assigned to the symbol have different drill, pad, anti-pad and thermal relief specifications.

You must input a paragraph in the pin file describing each pintype. You must input the PINTYPE line before inputting the pin information (drill, shape, size, layer, etc.).

If the file has more than one pintype, you must input a name that distinguishes one pintype from another; for example: PINTYPE A, PINTYPE B, etc.

EXAMPLE: |ADD PINTYPE| (PINTYPE NAME) A |ENTER|

The line appears on the graphics screen as part of the file:

```
                             DIP14-PIN

New                      (DIP14-PIN)
Cursor ---> * |_|        PINTYPE  A
Position                 END
```

The pintype name may contain up to four alphanumeric characters.

STEP 4. SELECT UNITS

Pick either |UNITS:ENGLISH| or |UNITS:METRIC| as your input unit for drill size and pad size.

If you do not make a choice, the default is English units. Note that when you choose English or metric units, you are choosing an input value for this file only. **THIS INPUT VALUE DOES NOT AFFECT ANY OTHER FILES IN YOUR PROJECT.**

When you input a number for size, if a decimal point is part of the number, the system assumes either inches or millimeters (depending on the input you chose). If this number has no decimal point, the system assumes mils for English units input and board database units (.01mm) for metric input.

You may change the unit of measurement at any time. However, note that when you choose a new unit of measurement, **ONLY THOSE DRILL SIZES AND PAD SIZES FOLLOWING THAT UNIT OF MEASUREMENT ARE IN THAT UNIT.** Drill sizes and pad sizes following the previously chosen unit are in that previously chosen unit of measurement.

For example, in the file below, the drill size is in mils, while the pad circle width is in millimeters.

```
              DIP14-PIN
         (DIP14-PIN)
   *     PINTYPE A
         UNITS:ENGLISH
   *     DRILL .035-P
         UNITS:METRIC
   *     PAD CIRCLE=.040 DRILL-MASK
         END
```

STEP 5. ADD DRILL

Pick the |ADD DRILL| command to input the drill size in mils or millimeters (depending on which input unit you chose) and the type of plating for that hole: |PLATED|, |NOT PLATED|, or |OPTIONAL|. When you input a number for size, if a decimal point is part of the number, the system assumes either inches or millimeters (depending on the input unit you chose). If this number has no decimal point, the system assumes mils for English units input and board database units (.01mm) for metric input.

EXAMPLE: |ADD DRILL| (DRILL SIZE [in mils]) 35 |ENTER|

Pick plating type: |PLATED|
|NOT PLATED|
|OPTIONAL|

```
                 DIP14-PIN

          (DIP14-PIN)
  *       PINTYPE A
  * |_|   DRILL .035-P
          END
```

STEP 6. |ADD PAD|

Pad information contains the keyword, PAD, followed by a geometric shape, a size in mils or millimeters (depending on which input unit you chose), and a physical layer assignment.

Currently, the system provides the following shapes:

|CIRCLE|
|SQUARE|
|RECTANGLE|
|OBLONG|
|FLASH|

NOTE: The physical layer nomenclature must be EXACTLY the same as used in the LAYERSTD FILE. (Refer to Layer Standard File section of the manual.). The physical layer assignment may contain up to 24 alphanumeric characters with no blank spaces.

EXAMPLE:

|ADD PAD| |CIRCLE| (WIDTH) 40 |ENTER| (LAYER) DRILL-MASK |ENTER|

RESULT:

```
                       DIP14-PIN

                   (DIP14-PIN)
New            *    PINTYPE A
Cursor ---->   *    DRILL .035-P
Position       * |_| PAD CIRCLE=.040 DRILL-MASK
                    END
```

STEP 7. |ADD THERM-RELIEF|

The thermal relief line of the pin file contains the following information:

1. The keyword THERMAL-RELIEF

2. A photoplotter aperture type (EX. FLASH)

3. A name that distinguishes one thermal relief from another.

4. A physical layer assignment (up to 24 alphanumeric characters)

The thermal relief name may contain up to 10 alphanumeric characters with no blank spaces.

EXAMPLE:

|ADD THERMAL RELIEF| |FLASH| (NAME) AB12 |ENTER| (LAYER) IMBEDDED-PLANE |ENTER|

RESULT:

```
                     *     PAD SQUARE=.080   SOLDER-SIDE
New                  *     PAD CIRCLE=.080   COMPONENT-SOLDER-MASK
Cursor --->          *     PAD CIRCLE=.080   SOLDER-SOLDER-MASK
Position             *     PAD CIRCLE=.040   INTERNAL-SIGNAL
                     * |_| THERMAL RELIEF FLASH=AB12  IMBEDDED-PLANE
                           END
```

When the THERMAL-RELIEF line of the file is input, the system immediately prompts you for ANTI-PAD data. You must now select the shape of the ANTI-PAD from the menu displayed on the function screen.

EXAMPLE:

Pick: |CIRCLE| (WIDTH) .040 |ENTER|

The physical layer nomenclature must be exactly the same for the THERMAL-RELIEF line and the ANTI-PAD line. The system automatically inserts the same physical layer name from the THERMAL-RELIEF line to the ANTI-PAD line of the pin file.

RESULT:

```
*         PAD SQUARE=.080 SOLDER-SIDE
*         PAD CIRCLE=.080 COMPONENT-SOLDER-MASK
*         PAD CIRCLE=.080 SOLDER-SOLDER-MASK
*         PAD CIRCLE=.040 INTERNAL-SIGNAL
*         THERMAL-RELIEF FLASH=AB12 IMBEDDED-PLANE
* |_|     ANTI-PAD CIRCLE=.040 IMBEDDED-PLANE
          END
```

STEP 8. |ADD PIN LINE|

After inputting specifications for all pintypes assigned to the symbol, you must list the pins that belong to each pintype. The list contains one line for each pin or series of pins belonging to a pintype.

Each line begins with the keyword PIN, followed by pin number(s) and the name of the pintype for these pins.

EXAMPLE:
```
PIN  1  A
PIN  2-14 B
```

EXAMPLE:
```
PIN 1,3,5 A
PIN 2 4 6-14 B
```

If more than one pin is assigned to a pintype, you may enter a sequential series of pin numbers by using a hyphen. If the pin numbers are not sequential, you may use commas or spaces to separate a number of pins assigned to that pintype.

EXAMPLE: |ADD PIN LINE| (PIN LIST) 1 |ENTER| (PINTYPE NAME) A |ENTER|

|ADD PIN LINE| (PIN LIST) 2-14 |ENTER| (PINTYPE NAME) B |ENTER|

RESULT:

Cursor Position --->
```
      PAD CIRCLE=.080 SOLDER-SOLDER-MASK
      PAD CIRCLE=.040 INTERNAL-SIGNAL
      THERMAL-RELIEF FLASH=AB12 IMBEDDED-PLANE
      ANTI-PAD CIRCLE=.040 IMBEDDED-PLANE
      PIN 1 A
 |_|  PIN 2-14 B
      END
```

STEP 9. |ADD COMMENT|

Use the |ADD COMMENT| command to add comment lines to the pin file. The comment line is automatically placed within parentheses ( ). This identifies the line of text as a comment.

```
                    ┌──────────────────────────────────────────┐
                    │                DIP14-PIN                  │
Current             │                                           │
Cursor ----->       │  |_|  (DIP14-PIN)                         │
Position            │       PINTYPE A                           │
                    │       DRILL .035-P                        │
                    └──────────────────────────────────────────┘
```

EXAMPLE: |ADD COMMENT| ---> Keyboard Menu

INPUT: PIN FILE FOR SYMBOL DIP14 |ENTER|

RESULT:

```
                    ┌──────────────────────────────────────────┐
                    │                DIP14-PIN                  │
New                 │       (DIP14-PIN)                         │
Cursor ----->       │  |_|  (PINFILE FOR SYMBOL DIP14)          │
Position            │       PINTYPE A                           │
                    │       DRILL .035-P                        │
                    └──────────────────────────────────────────┘
```

Since the comment line is optional and is ignored by the system, youi may place the comment line above the section or line it is related to. The |ADD COMMENT| command allows you to input additional information for easy interpretation of the pin file.

NOTE: You may input a maximum of 59 alphanumeric and blank spaces on a comment line. If your input exceeds the 59 alphanumeric characters, the system interrupts your input and inserts the 59-character comment line on the file.

STEP 10. |DONE|

Pick |DONE| when you are finished with the creation of the pin description file. To close the file, pick one of the following commands:

|SAVE FILE| - causes the system to save the active version of the file. The file is saved under the same revision label that was named when the file was opened.

If you created a new file, the system assigns a revision label of "1" when you close the file.

EXAMPLE:

Active File: DIP14-PIN

|SAVE FILE|: DIP14-PIN 1

|SAVE FILE NEW RV| - allows you to input a new revision label for your active file. The new revision label may contain four alphanumeric characters.

The system saves the file that you opene4d under the new revision label WITHOUT deleting or changing the original and its revision.

EXAMPLE:

Original Version (old REV): DIP14-PIN

|SAVE FILE NEW RV|: (REV) 2 |ENTER|

RESULT: DIP14-PIN 2

|CANCEL ACTV FILE| - causes the system to delete the active version of the file. The original is not deleted; it keeps the same revision label.

|EDIT FILE| - allows you to "menu-back" and return to the active file for further editing.

3. EDITING THE PIN DESCRIPTION FILE

As you create a pin description file for a particular PC board symbol, you may wish to make additions or corrections. You may also edit the file at a later time to update any version of a file you previously created. When you reopen the file, you must input the name of the pin description file and the revision label. Simply pick ENTER after inputting the pin file name if you wish to edit the latest version.

You can use the following commands to edit the pin description file :

CHANGE PINTYPE	DELETE SECTION
CHANGE DRILL	UNDELETE SECTION
CHANGE PAD	DELETE LINE
CHANGE THERM - REL	UNDELETE LINE
CHANGE ANTI - PAD	CHANGE COMMENT
CHANGE PIN LINE	

POSITIONING THE CURSOR

To edit a file, you must always move the cursor to the proper position. The cursor remains in the left margin of the file, and must be positioned at or above the line to be edited.

For ADD commands used during editing :

Position the cursor on the line above the desired location of the new line. Lines positioned below the cursor shift down when the new line is inserted.

If the cursor is positioned at the END statement, the new line will be positioned above the END statement.

For CHANGE or DELETE commands :

Move the cursor to the line to be changed or deleted.

Use these lines to move the cursor :

SEC UP - Moves the cursor up one section.

SEC DWN - Moves the cursor down one section.

LINE UP - Moves the cursor up one line.

LINE DWN - Moves the cursor down one line.

THE DIFFERENCE BETWEEN A SECTION AND A LINE

The system groups the pin description file into sections of interrelated data. For example, the file may contain information about more than one pintype (PINTYPE A, PINTYPE B, etc.). Therefore, each pintype paragraph in the pin file is identified by the system as a section. When editing within a section, you may use the |LINE UP| and |LINE DWN| commands to position the cursor to a line you wish to CHANGE or DELETE.

While you are working within a section, asterisks (*) appear at the left margin of your file. The asterisk flags each line comprising a PINTYPE section, and labels the section were you may perform editing tasks. When you pick |DONE|, the system positions the cursor back to the PINTYPE line of that section and cancels the asterisk display.

When the cursor is positioned at a PINTYPE line, use the |SEC DWN| command to move the cursor to the next PINTYPE section. In some cases, a section may consist of only one line. For example, each PIN LINE is defined by the system as a section.

EXAMPLE: |LINE DWN| (Picked 4 times)

EXAMPLE:

```
                         DIP14-PIN      1
                  (DIP14-PIN)
Current           (PIN FILE FOR SYMBOL DIP14)
Cursor -->  * |_| PINTYPE  A
Position    *     DRILL .035-P
            *     PAD CIRCLE=.040 DRILL-MASK
New         *     PAD SQUARE=.062 COMPONENT-SIDE
Cursor -->  * |_| PAD SQUARE=.080 SOLDER-SIDE
Postion     *
```

EXAMPLE: |SEC DWN| (Picked twice)

```
Current           PAD CIRCLE=.040 INTERNAL SIGNAL
Cursor            THERMAL-RELIEF FLASH=AB12 IMBEDDED PLANE
Position          ANTI-PAD CIRCLE=.040 IMBEDDED-PLANE
    --->      |_| PIN 1  A
New               PIN 2-14 B
Cursor --->   |_| END
Position
```

IF THE FILE IS LONGER THAN SHOWN ON THE GRAPHICS SCREEN

If you add lines to the file after the graphics screen is full, the file scrolls up so that the bottom lines are visible and the top lines are not. Use the LINE and SEC commands to scroll your file to the lines or sections you wish to edit.

TO CHANGE INFORMATION IN THE FILE

Use the CHANGE commands to change or update the pin description file. You must move the cursor to the line you wish to edit.

When you wish to edit lines within a section of the pin file, use the |SEC UP| and |SEC DWN| commands to position the cursor to that section. The |CHANGE PINTYPE| command allows you to edit all lines assigned to that pintype section. When the system prompts you for a pintype name, simply pick |ENTER| to flip to the appropriate menu page for changing information within that pintype section. Each line assigned to that section is flagged with an asterisk at the left margin.

EXAMPLE: Cursor is positioned at the PINTYPE A section

Pick: |CHANGE PINTYPE| |ENTER|

RESULT: The asterisk display appears on the graphics screen.

```
                              DIP14-PIN      1
          (DIP14-PIN)
          (PIN FILE FOR SYMBOL DIP14)
*   |_|   PINTYPE  A
*         DRILL .035-P
*         PAD CIRCLE=.040 DRILL-MASK
*         PAD SQUARE=.062 COMPONENT-SIDE
*         PAD SQUARE=.080 SOLDER-SIDE
```

EXAMPLE: Use |LINE DWN| to position the cursor to the line to be changed.

PICK: |CHANGE PAD|

INPUT: |SQUARE| (WIDTH [in mils]) 85 |ENTER| (LAYER) SOLDER-SIDE |ENTER|

RESULT: The pad size .080 was changed to .085.

```
                              DIP14-PIN      1
                (DIP14-PIN)
                (PIN FILE FOR SYMBOL DIP14)
              *         PINTYPE  A
              *         DRILL .035-P
              *         PAD CIRCLE=.040 DRILL-MASK
Current       *         PAD SQUARE=.062 COMPONENT-SIDE
Cursor  --->  *   |_|   PAD SQUARE=.085 SOLDER-SIDE
Position
```

When you use the CHANGE commands, you may pick |ENTER| for any input that is unchanged. For example, if you use the |CHANGE PAD| command to update the pad size only, pick |ENTER| for those prompts that already have the correct information displayed on the graphics screen.

EXAMPLE:

Current Line

```
PAD CIRCLE=.040 DRILL-MASK
```

PICK: |CHANGE PAD| |CIRCLE| (WIDTH [IN MILS]) 45 |ENTER| (LAYER) |ENTER|

RESULT: The pad size is changed to .045; DRILL-MASK remains the same.

```
PAD CIRCLE=.045 DRILL-MASK
```

DELETING INFORMATION FROM THE FILE

Use these commands to DELETE and reinsert information on the pin description file:

|DELETE LINE| - This command deletes one line from the file. You must move the cursor to the line you wish to delete within a section.

|UNDELETE LINE| - This command reinserts the last line deleted. You may use |UNDELETE LINE| to reposition the line to another area of the file within a section. The system restores the line below your new cursor position regardless of the line's original placement.

|DELETE SECTION| - This command deletes an entire section from your file. You must move the cursor to the beginning of the section you wish to delete.

|UNDELETE SECTION| - This command restores the last section you deleted. You may reposition an entire section to another location in the pin file.

NOTE: You cannot delete the symbol name at the top center position of your file, or the END statement. You can only delete lines that you input when you created the file.

USING THE |DELETE SECTION| AND |DELETE LINE| COMMANDS

Since the system may recognize a line as a section, you may use the |DELETE SECTION| command to eliminate a line from the pin file.

EXAMPLE: Deleting a comment line (section) from the pin file.

```
                    DIP14-PIN      1
      (DIP14-PIN)
|_|   (PIN FILE FOR SYMBOL DIP14)
      PINTYPE  A
      DRILL .035-P
      PAD CIRCLE=.040 DRILL-MASK
```

PICK: |DELETE SECTION|

When you pick |DELETE SECTION|, the comment line (PIN FILE FOR SYMBOL DIP14) is deleted from the file. The contents below your deleted line will shift up and close the open line.

RESULT:

```
                    DIP14-PIN      1
      (DIP14-PIN)
|_|   PINTYPE  A
      DRILL .035-P
      PAD CIRCLE=.040 DRILL-MASK
```

Use the |SEC UP| or |SEC DWN| commands to reposition the cursor to another section of the file. Pick the |UNDELETE SECTION| command to reinsert the line (PIN FILE FOR SYMBOL DIP14) to a new position below the cursor.

EXAMPLE: Deleting a line from the PINTYPE A section of the file.

```
Cursor  |                          DIP14-PIN      1
        |        (DIP14-PIN)
        |*       PINTYPE  A
---->   |*  |_|  DRILL .035-P
        |*       PAD CIRCLE=.040 DRILL-MASK
```

PICK: |DELETE LINE|

RESULT: The DRILL line of the file is deleted.

```
Cursor  |                       DIP14-PIN      1
        |     (DIP14-PIN)
        |*    PINTYPE  A
------> |* |_| PAD CIRCLE=.040 DRILL-MASK
```

You may now use the |UNDELETE LINE| command to reinsert the DRILL line back to the pin file.

Use the |LINE UP| command to position the cursor back to the PINTYPE A. Pick |UNDELETE LINE| to reinsert it to its original position in the file.

4. PIN DESCRIPTION FILES NOT CREATED WITH THE |EDIT PIN FILE| LEADTHRU

You may currently have pin description files on your system that were not created with the |EDIT PIN FILE| lead-thru. You may edit these files using the |EDIT PIN FILE| command.

When you open one of these files with the |EDIT PIN FILE| command, the system checks it for formatting errors. If errors are present, the system places a message on the function screen telling you how many faulty line exist in the file. It displays the file on the graphics screen with all faulty lines bracketed between parentheses ( ) and asterisks (*).

EXAMPLE: (*PAD CIRCLE =.040 DRILL MASK*)

The line is incorrect because there is a space between CIRCLE and the equals sign (=). A hyphen should also appear between the words DRILL and MASK.

The lines in parentheses are now comments. You may delete these lines with the |DELETE LINE| command. Use the ADD commands to input the correct information.

If you are using the optional VT101 terminal to create and edit the pin file; refer to the example at the beginning of this section for general formatting requirements. You may later use the |EDIT PIN FILE| lead-thru to display the file on the Telesis system and to check it for formatting errors.

NOTE: The spacing requirements, keyword spelling, individual line formats and the maximum character inputs are described within this section. However, refer to the COMMANDS section of this manual for additional information on each command used to create and edit the pin description file.

5. PIN FILES WITH SPECIAL REQUIREMENTS

VIAS

A via symbol has a connect point, but the connect point does not have a pin number. Use the |ADD PINTYPE| command to insert the keyword PINTYPE to the pin file. When the system prompts you for a pintype name, input a space, then |ENTER|. The keyword PINTYPE appears on the graphics screen WITHOUT a pintype name.

EXAMPLE:

```
                    VIA-PIN      1
(VIA -PIN)
PINTYPE
DRILL .035-P
PAD CIRCLE=.040 DRILL-MASK
PAD CIRCLE=.062 COMPONENT-SIDE
PAD CIRCLE=.062 SOLDER-SIDE
PAD CIRCLE=.080 COMPONENT-SOLDER-MASK
PAD CIRCLE=.080 SOLDER-SOLDER-MASK
PAD CIRCLE=.040 INTERNAL-SIGNAL
THERMAL-RELIEF FLASH=AB12 IMBEDDED-PLANE
ANTI-PAD CIRCLE=.040 IMBEDDED-PLANE
END
```

EDGE CONNECTORS

Edge connectors require a pin file even though they do not have drill or pad specifications. For this pin file, you need to input the PINTYPE line, and the PIN LIST.

EXAMPLE:

```
          ECON-PIN  1

(ECON-PIN)
PINTYPE  A
PIN 1-100 A
END
```

DRAWING FORMATS

1. GENERAL INFORMATION

2. TWO METHODS OF FILING A DRAWING FORMAT IN A LIBRARY

1. GENERAL INFORMATION

We recommend that you use the system to draw a copy of your standard drawing format(s) or outlines in each of the standard sizes: A, B, C and D SIZES. Then keep these formats in the SYSTEM-LIBRARY to be used each time you begin a new schematic or board drawing.

The system does not require you to keep a library copy of your drawing format, but it will save you the time involved in drawing the format on the system every time you create a new drawing.

ELEMENTS

Use lines, arcs, and/or circles in the format drawing. Do not use connect lines, connect arcs or connect points.

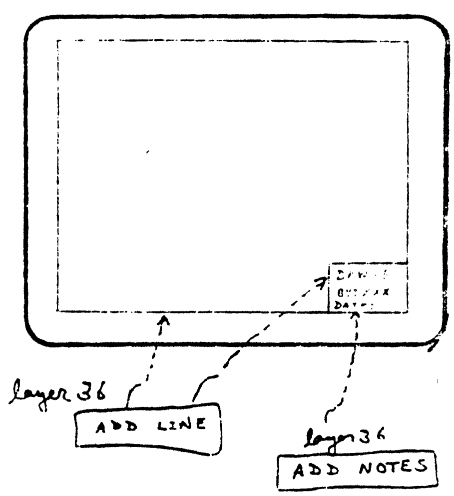
TEXT

Add the text in your format drawing with the |ADD NOTES| command only.

LAYERS

Refer to the Telesis Layer Standards Table (p. GRAPHICS-8) for the proper layer settings for the drawing format.

EXAMPLE



2. TWO METHODS OF FILING A DRAWING FORMAT IN A LIBRARY

You may keep the library copy of your drawing format as either a symbol file or a drawing file. Each method has its advantages and limitations.

SYMBOL FILE METHOD

To use this method, follow the same basic procedure as in creating any other symbol file:

STEP 1 Open a drawing file.

STEP 2 Draw the format according to your company's standards.

STEP 3 Use the |CREATE SYMBOL| command to create a symbol file from the drawing file.

Whenever you open a new board or schematic drawing, you may use the ADD SYMBOL commands to place your format in the drawing.

The only drawback to this method is that you must create a pin description text file to correspond with this symbol file. Even though your drawing format, of course, has no pins, the system cannot perform design rules checking or artwork unless it verifies that there is a pin file for every symbol on a board drawing.

The pin file you must create, if you use this method, should have the format shown in this example:

```
(PIN FILE FOR DRAWING FORMAT)
(FILE NAME FORMAT-PIN)
 PINTYPE
 END
```

DRAWING FILE METHOD

You may keep a library copy of your drawing format as a drawing file rather than as a symbol file if you wish.

With this method, each time you begin a schematic or board drawing, you use the |OLD DRAWING| command to open the format drawing file. Then change the name of the format drawing file to the name you wish to use for the schematic or board drawing.

If you use this method, you must draw the format twice:

1. You must have one copy of the format that you opened with the |BOARD DRAWING| command.

2. You must have another copy of the format that you opened with the |OTHER DRAWING| command.

The format drawing file that you opened with |BOARD DRAWING|, can only be used in board drawings; and the one you opened with |OTHER DRAWING|, can only be used in schematics or other drawings.

If you choose this method, be sure to name these two library drawing files in a way that indicates which is to be used with boards, and which is not, for example:

1st drawing file name: FORMAT-BRD-SIZE-A
2nd drawing file name: FORMAT-SCH-SIZE-A

D E V I C E   D E S C R I P T I O N   F I L E S

1. GENERAL INFORMATION
   - o EXAMPLE
   - o LOCATION OF DEVICE FILES
   - o TELESIS-PREPARED DEVICE LIBRARY

2. CONTENTS OF A DEVICE DESCRIPTION FILE

3. CREATING A DEVICE DESCRIPTION FILE WITH THE TEXT LEADTHRU

4. EDITING A DEVICE DESCRIPTION FILE WITH THE TEXT LEADTHRU
   - o POSITIONING THE CURSOR
   - o THE DIFFERENCE BETWEEN A SECTION AND A LINE
   - o CHANGING INFORMATION IN THE FILE
   - o DELETING INFORMATION FROM THE FILE
   - o CONTINUATION LINES
   - o USING THE MERGE AND BREAK LINE COMMANDS
   - o EDITING A DEVICE FILE NOT CREATED WITH THE EDIT DEVICE FILE TEXT LEADTHRU

DEVICE DESCRIPTION FILES

1. GENERAL INFORMATION

EXAMPLE

```
(DEVICE DESCRIPTION FILE: 7400)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER 7400 A B Y
PINUSE 7400 IN IN OUT
PINSWAP 7400 A B
FUNCTION G1 7400 1 2 3
FUNCTION G2 7400 4 5 6
FUNCTION G3 7400 9 10 8
FUNCTION G4 7400 12 13 11
POWER +5V; 14
GROUND GND; 7
END
```

Device description files provide information about each physical device, its logic functions, and all pin information. The system takes information from device files when the operator uses the |EXTRACT NETLIST|, or |LOAD TXT NETLIST| command to create the net data base file. Device files are essential for the following system capabilities requiring a net data base:

- -AUTOMATIC ASSIGNMENT-
- -AUTOMATIC PLACEMENT-
- -BACK ANNOTATION-
- -AUTOMATIC ROUTING-
- -INTERACTIVE PLACEMENT-
- -AUTOMATIC SWAPPING OF GATES AND PINS-
- -LOGICAL DESIGN RULES CHECKING-

LOCATION OF DEVICE FILES

When the operator uses the |EXTRACT NETLIST| or |LOAD TXT| NETLIST command, device files must be in the current project or in the SYSTEM-LIBRARY. It is recommended that devices files be kept in the SYSTEM-LIBRARY.

TELESIS-PREPARED DEVICE LIBRARY

The Telesis-prepared device description files are shown in Volume 2 of this manual. In addition, the operator may create device files for specific requirements.

Like all text files required by the system, device files have specific format requirements. To assist the operator in meeting these requirements, the system provides the |EDIT DEVICE FILE| text leadthru. The following sections describe the leadthru step-by-step.

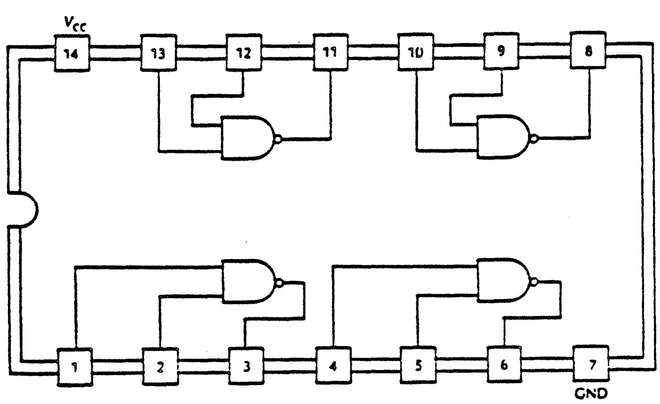
2. CONTENTS OF A DEVICE DESCRIPTION FILE

A device description file contains the following information:

- o The physical PACKAGE. (DIP14)
- o The CLASS associated with the device. (IC, IO or DISCRETE)
- o The number of pins contained on the device, or the PINCOUNT.
- o The types of logic FUNCTIONS contained within the device, and how their logical pins correspond to the pin numbers on the device.
- o The use of each pin contained on the device. (IN, OUT, etc.)
- o The swappable pin names contained on single logic function types within the device.
- o The connection of power and ground pins.

The example below illustrates the 7400 device shown in the TTL Data Book for Design Engineers and the Telesis device description file. The information for many device description files can be acquired from the TTL Data Book for Design Engineers which illustrates devices by "family" (i.e., TTL, ECL, etc.). For example, the 7400 device (TTL) pictured below, shows the four NAND2 gates and their appropriate function slots.

DEVICE 7400



DEVICE DESCRIPTION FILE

```
(DEVICE DESCRIPTION FILE: 7400)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER 7400 A B Y
PINUSE 7400 IN IN OUT
PINSWAP 7400 A B
FUNCTION G1 7400 1 2 3
FUNCTION G2 7400 4 5 6
FUNCTION G3 7400 9 10 8
FUNCTION G4 7400 12 13 11
POWER +5V; 14
GROUND GND;7
END
```

3. CREATING A DEVICE DESCRIPTION FILE WITH THE TEXT LEADTHRU

The text leadthru capability allows the operator to create a device file on the function screen, with the system maintaining the proper formatting requirements for each line entry. The operator simply uses the interactive commands on the |EDIT DEVICE FILE| leadthru to input and edit the information contained in the file. The leadthru reduces the possibility of numerous formatting errors that could result when the text file is created on a text editor keyboard.

When using the |EDIT DEVICE FILE| leadthru, the system assists the operator in the following ways:

- o Automatically positions the cursor when the operator inputs information.
- o Automatically inputs the keywords required in the file.
- o Places parentheses ( ) around comment lines.
- o Issues error messages if the operator-defined input is improperly formatted.

The operator may use the steps described below to create a device file using the leadthru capability:

STEP 1. |TEXT LEADTHRU|

Pick the |TEXT LEADTHRU| command to display the menu of leadthru options:

|EDIT DEVICE FILE|
|EDIT APERTURE-TAB|
|EDIT PIN FILE|
|EDIT LAYERSTD|
|EDIT PENPLOT-CON|

STEP 2. |EDIT DEVICE FILE|

Pick the |EDIT DEVICE FILE| command to open an existing device file, or to create a new file.

EXAMPLE: |EDIT DEVICE FILE| (FILENAME) 7400 |ENTER|

If creating a new file, the operator may input up to 18 alphanumeric characters with no blank spaces. A revision label with up to four alphanumeric characters may be specified. Separate the filename from the revision with a blank space.

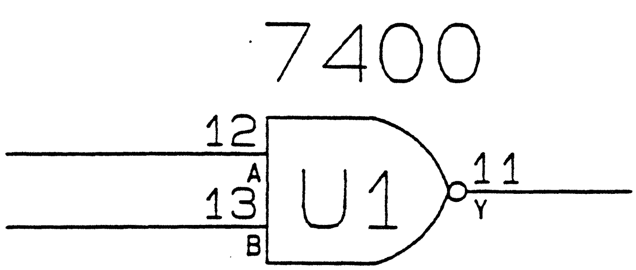
The device file name specified is the same name that will be used as the device type label on the logic symbols belonging to the device. For example, the device 7400 (Quad 2-input NAND) must have a device type label 7400 on each 7400 logic symbol (NAND2) on the schematic.

In cases where a symbol does not have a device type label, the system uses the symbol name as the device type. In these cases, the device description file must have the same name as the symbol:

EXAMPLE 1:	SYMBOL NAME:	NAND2
	SYMBOL DEVICE TYPE LABEL:	7400
	CORRESPONDING DEVICE FILE NAME:	7400
EXAMPLE 2:	SYMBOL NAME:	RPACK4
	SYMBOL DEVICE TYPE LABEL:	None
	CORRESPONDING DEVICE FILE NAME:	RPACK4

The example below illustrates the NAND2 gate (FUNCTION G4) with the 7400 device type label, and the 7400 device description file.

DEVICE TYPE ATTACHED




```
(7400)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER 7400 A B Y
PINUSE 7400 IN IN OUT
PINSWAP 7400 A B
FUNCTION G1 7400 1 2 3
FUNCTION G2 7400 4 5 6
FUNCTION G3 7400 9 10 8
FUNCTION G4 7400 12 13 11
POWER +5V; 14
GROUND GND; 7
END
```

When |ENTER| is picked, the system displays the following information on the graphics screen:

```
                    7400
  _  (7400)
     END
```

(The cursor is positioned at the start of the comment line (7400)).

The operator may now use the following commands to add information between (7400) and the END statement.

|ADD PACKAGE|
|ADD CLASS|
|ADD PINCOUNT|
|ADD COMMENT|
|ADD CONTINUATION LINE|
|ADD POWER|
|ADD GROUND|
|ADD NC|

|ADD PINORDER|
|ADD PINUSE|
|ADD PINUSE CONTINUATION|
|ADD PINSWAP|
|ADD FUNCTION|

STEP 3. |ADD PACKAGE|

Pick the |ADD PACKAGE| command to enter the package name.

EXAMPLE: |ADD PACKAGE| (ENTER PACKAGE NAME) DIP14 |ENTER|

The package name must be identical to the name of the component symbol used for this device on the board drawing.

A maximum of 14 characters are allowed, with no blank spaces.

RESULTING GRAPHICS SCREEN DISPLAY:

```
                    7400
     (7400)
  _  PACKAGE DIP14
     END
```

STEP 4. |ADD CLASS|

Pick the |ADD CLASS| command, then pick the device's class type from one of the three menu box choices provided:

|CLASS IO|
|CLASS IC|
|CLASS DISCRETE|

EXAMPLE: |ADD CLASS| |CLASS IC|

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

```
                 7400
      (7400)
      PACKAGE DIP14
      CLASS IC
  _   END
```

STEP 5. |ADD PINCOUNT|

Pick the |ADD PINCOUNT| command to enter the number of pins on the physical device.

EXAMPLE: |ADD PINCOUNT| (PINCOUNT) 14 |ENTER|

The PINCOUNT must be an integer.

NOTE: The PINORDER section may not be omitted if the PINCOUNT entered here is less than the largest pin number on the symbol.

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

```
                 7400
      (7400)
      PACKAGE DIP14
      CLASS IC
      PINCOUNT 14
  _   END
```

STEP 6. |MENU->| |ADD PINORDER|

Pick the |ADD PINORDER| command, then enter a logical function type followed by the pin names for that function.

EXAMPLE: |ADD PINORDER| (ENTER FUNCTION TYPE) 7400 ENTER

(ENTER LIST OF PIN NAMES) A B Y ENTER

The function type may contain up to 18 alphanumeric characters with no blank spaces. It must be the same name as the function type label on the logic symbol. If the logic symbol has no function type label, this entry must be the same as the device type label on the logic symbol. The system will add the function type label to each line entry contained in the PINORDER section of the file.

Pin names may contain up to eight alphanumeric characters with no blank spaces. Pin names must be separated with a blank space.

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

```
                 7400
     (7400)
     PACKAGE DIP14
     CLASS IC
     PINCOUNT 14
*  _ PINORDER 7400 A B Y
     END
```

When the operator uses the |ADD PINORDER| command, the system opens the PINORDER section of the file. The section is highlighted with an asterisk display preceding each line entry under PINORDER.

The following commands may then be used to build the PINORDER section of the device file:

|ADD PINUSE|
|ADD PINUSE CONTINUATION|
|ADD CONTINUATION LINE|
|ADD PINSWAP|
|ADD FUNCTION|

STEP 7. |ADD PINUSE|

Pick the |ADD PINUSE| command, then enter a pinuse code for each pin name entered on the PINORDER line of the file. There are nine pinuse codes available on the menu:

IN	OUT	OCA	BI	TRI	OCL
NC	POWER	GROUND			

EXAMPLE: |ADD PINUSE| |IN| |IN| |OUT| |ENTER|

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

```
                  7400
      (7400)
      PACKAGE DIP14
      CLASS IC
      PINCOUNT 14
*     PINORDER 7400 A B Y
*  _  PINUSE 7400 IN IN OUT
      END
```

ADD PINUSE CONTINUATION

Use the |ADD PINUSE CONTINUATION| command if the contents of the PINUSE line exceed the 59-character limit on the graphics screen. This command allows the operator to continue PINUSE information; the system simply adds a comma after the last entry on the current PINUSE line of the file, with the new information added directly below. The text file cursor must be positioned at the PINUSE line.

|ADD PINUSE CONTINUATION| may be used to add additional information to PINUSE without changing the line's current contents.

EXAMPLE: Existing PINUSE line in the device file:

```
* _ PINUSE 7400  IN IN
```

|ADD PINUSE CONTINUATION| |OUT| |ENTER|

Resulting PINUSE line with continuation:

```
PINUSE 7400  IN IN,
 OUT
```

The |ADD CONTINUATION LINE| command may be used for long line entries other than PINUSE. See CONTINUATION LINES.

STEP 8. |ADD PINSWAP|

Pick the |ADD PINSWAP| command to specify the swappable pins (by pin name) for each identical logic function type on the device. For example, the device 7400 contains four identical logic functions (NAND2). In the file shown on the previous page, the PINORDER line specifies the pin names "A" and "B", with PINUSE "IN "IN", respectively. Therefore, since the pin names "A" and B" are both specified as "IN", swappable pins exist for each instance of the logic function.

With a PINSWAP definition in the device file, the operator may create a net data base that allows interactive or automatic swapping of IC logic function pins on the active board drawing. Pin swapping improves the placement of ratsnest connect-lines for increased completion percentage during automatic routing.

EXAMPLE: |ADD PINSWAP| (ENTER PIN NAMES) A B |ENTER|

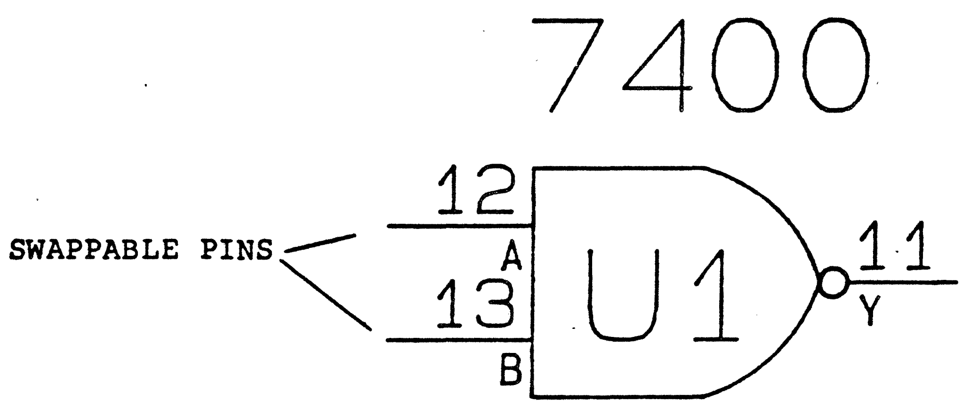
Separate each pin name with a blank space during input.

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

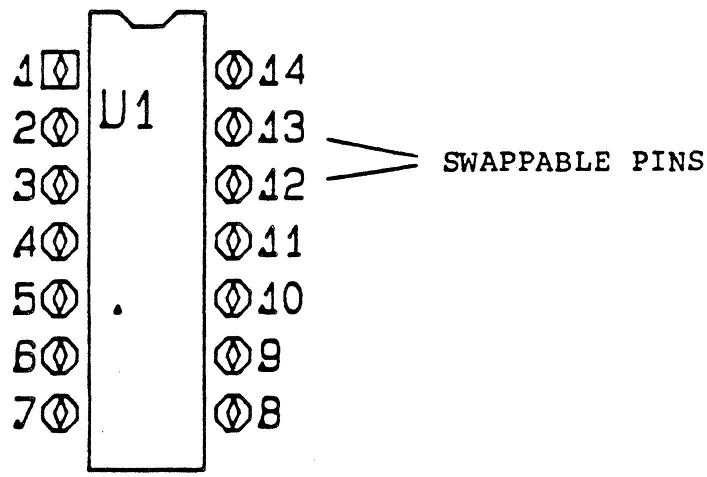
```
                 7400
     (7400)
     PACKAGE DIP14
     CLASS IC
     PINCOUNT 14
*    PINORDER 7400 A B Y
*    PINUSE 7400 IN IN OUT
*  _ PINSWAP 7400 A B
     END
```

The illustrations on the next page show the swappable pins for a single logic function (NAND2) on the 7400 device:

EXAMPLE: FUNCTION G4 (NAND2, DEVICE 7400) ON THE SCHEMATIC



PACKAGE DIP14 (DEVICE 7400) ON THE BOARD DRAWING



STEP 9. |ADD FUNCTION|

Pick the |ADD FUNCTION| command, then enter the slot name for this instance of the function, followed by the pin numbers used by this instance of the function.

|ADD FUNCTION| (ENTER SLOT NAME) G1 |ENTER| (PIN NUMBERS) 1 2 3 |ENTER|

The slot name may contain up to four alphanumeric characters with no blank spaces.

Pin numbers must be listed in order corresponding to the pin names on the PINORDER line. Pin numbers must be separated by blank spaces, or with commas (,). A pin number must be an integer from 1 to 32767.

Repeat the |ADD FUNCTION| command for each additional instance of this function type on the device.

Pick the |DONE| command after inputting each additional instance of the function type on the device.

Picking |DONE| completes the PINORDER section of the device file, with the asterisk display de-highlighting on the graphics screen.

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

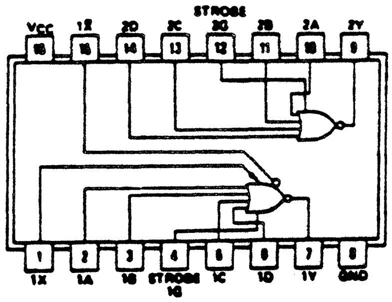
```
                    7400
     (7400)
     PACKAGE DIP14
     CLASS IC
     PINCOUNT 14
*    PINORDER 7400 A B Y
*    PINUSE 7400 IN IN OUT
*    PINSWAP 7400 A B
*    FUNCTION G1 7400 1 2 3
*    FUNCTION G2 7400 4 5 6
*    FUNCTION G3 7400 9 10 8
*    FUNCTION G4 7400 12 13 11
     END
```

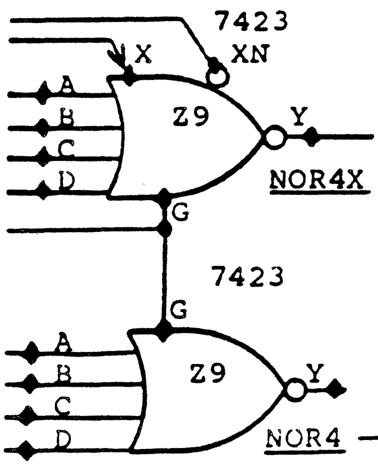
STEP 10. |ADD PINORDER| (if required)

Repeat the |ADD PINORDER| command (STEP 6.) if the device has more than one logical function. For example, the device 7423 contains the function type, NOR4, representing the 4-input NOR gate and the function NOR4X representing the extended 4-input NOR gate. The device file must contain two PINORDER sections, one for each function type on the device.

The example on the following page shows the 7423 device, its logic functions, and the appropriate entries in the device file for each function type.

EXAMPLE: DEVICE 7423






```
(DEVICE FILE: 7423)
PACKAGE DIP16
CLASS IC
PINCOUNT 16
PINORDER NOR4X A B C D G X XN Y
PINUSE NOR4X IN IN IN IN IN IN IN OUT
PINSWAP NOR4X A B C D
FUNCTION FS1 NOR4X 2 3 4 5 6 4 1 15 7
PINORDER NOR4 A B C D G Y
PINUSE NOR4 IN IN IN IN IN OUT
PINSWAP NOR4 A B C D
FUNCTION FS2 NOR4 10 11 13 14 12 9
POWER +5V; 16
GROUND GND; 8
END
```

STEP 11. |ADD POWER|

Pick the |ADD POWER| command, enter a signal name, then the pin number or numbers.

EXAMPLE: |ADD POWER| (ENTER POWER SIGNAL NAME) +5V |ENTER|

(PIN NUMBERS) 14 |ENTER|

The signal name may contain up to 18 alphanumeric characters with blank spaces allowed.

NOTE: If using the Telesis Keyboard or Function Screen text editor to create the device file, input a semi-colon immediately following the signal name. The |EDIT DEVICE FILE| leadthru automatically inserts a semi-colon when |ENTER| is picked.

The pin numbers must be separated by blank spaces or with commas.

A pin number must be an integer from 1 to 32767.

Power information may be omitted from device files for devices such a resistors that do not require this information.

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

```
                    7400
    (7400)
    PACKAGE DIP14
    CLASS IC
    PINCOUNT 14
    PINORDER 7400 A B Y
    PINUSE 7400 IN IN OUT
    PINSWAP 7400 A B
    FUNCTION G1 7400 1 2 3
    FUNCTION G2 7400 4 5 6
    FUNCTION G3 7400 9 10 8
    FUNCTION G4 7400 12 13 11
_   POWER +5V;  14
    END
```

STEP 12. |ADD GROUND|

Pick the |ADD GROUND| command, enter a ground signal name, then the pin number or numbers.

EXAMPLE: |ADD GROUND| (ENTER GROUND SIGNAL NAME) GND |ENTER|

(PIN NUMBERS) 7 |ENTER|

The |ADD GROUND| command has the same requirements as |ADD POWER|.

RESULTING DISPLAY ON THE GRAPHICS SCREEN:

```
                    7400
    (7400)
    PACKAGE DIP14
    CLASS IC
    PINCOUNT 14
    PINORDER 7400 A B Y
    PINUSE 7400 IN IN OUT
    PINSWAP 7400 A B
    FUNCTION G1 7400 1 2 3
    FUNCTION G2 7400 4 5 6
    FUNCTION G3 7400 9 10 8
    FUNCTION G4 7400 12 13 11
    POWER +5V;  14
    GROUND GND; 7
_   END
```

STEP 13. |ADD NC| (no connection)

Pick the |ADD NC| command to list any pins on the device that are not connected.

EXAMPLE: |ADD NC| (LIST OF PINS NOT CONNECTED) 3 11 |ENTER|

Pin numbers must be separated by blank spaces, or commas (,).

A pin number must be an integer from 1 to 32767.

The device file for the 7420 device is shown below with the NC entry:

```
        (7420)
        PACKAGE DIP14
        CLASS IC
        PINCOUNT 14
        PINORDER 7420 A B C D Y
        PINUSE 7420 IN IN IN IN OUT
        PINSWAP 7420 A B C D
        FUNCTION G1 7420 1 2 4 5 6
        FUNCTION G2 7420 9 10 12 13 8
        POWER +5V; 14
        GROUND GND; 7
---->   NC 3 11
        END
```

STEP 14. |ADD COMMENT| (OPTIONAL)

The operator may use the optional |ADD COMMENT| command to supply informational text into the file. A comment is always enclosed within parentheses ( ), and is ignored by the system when the system processes the device file. Comments may be conveniently placed above the related section or line entry, providing the additional information needed for easy interpretation of the file's contents.

NOTE: A maximum of 59 alphanumeric and/or blank spaces may be included in a comment entry. If the keyboard input exceeds 59 characters, the system interrupts, then inserts the first 59 characters into the file. The system automatically inserts parentheses around operator-defined text.

If using the Telesis Keyboard or Function Screen Editor, input parentheses around line entries to be treated as comments.

STEP 15. |DONE|

When finished creating the device file, pick |DONE|. If errors exist in the file, the system creates a DEVICE-LOG text file. This text file is displayed on the function screen, listing the error conditions.

EXAMPLE:

-ERROR IN PARSING DEVICE FILE 7400 REV...-

-PIN NUMBER 1 IS DEFINED IN BOTH A FUNCTION AND IN A POWER, GROUND OR NC STATEMENT. THIS IS ILLEGAL.-

-THIS DEVICE FILE WILL BE IGNORED.-

Proceed to use the |EDIT FILE| command to return to the active file for correction of any error condtionthat may exist. When errors have been cleared, pick |DONE|, then proceed to close the file with one of the following commands:

|SAVE FILE| - allows the operator to save the active version of the file, and to delete the previous version of the same revision. If the file is newly created, the system saves the file under the revision label specified when the file was opened, or with revision "1" (default).

|SAVE FILE NEW RV| - allows the operator to save the active file under a revision label other than the one originally specified when the file was opened. The system does not delete the old version.

If the active file is newly created without a revision label originally specified, the system saves the file with the revision label input with the |SAVE FILE NEW RV| command.

|CANCEL ACTV FILE| - allows the operator to cancel the actively displayed file on the graphics screen without deleting the original (if it exists). The original file maintains its original revision label.

When creating a device file, the operator may make additions or corrections. The file may also be re-opened at a later time to update any version of the file previously created. When re-opening the file, the operator must specify the revision label if the file is not the latest revision.

The following commands may be used to edit a device file using the leadthru capability:

CHANGE PACKAGE	CHANGE COMMENT
CHANGE CLASS	CHANGE POWER
CHANGE PINCOUNT	CHANGE GROUND
MERGE	CHANGE NC
BREAK LINE	DELETE
CHG CONTINUATION LINE	UNDELETE

CHANGE PINORDER
CHANGE PINUSE
CHANGE PINUSE CONTINUATION
CHANGE PINSWAP
CHANGE FUNCTION

Refer to COMMANDS section of the manual for the exact input sequence for each command used to CHANGE information in the file.

POSITIONING THE CURSOR

To edit the file, the operator must always position the cursor to the line to be edited. The cursor remains in the left margin of the file and must be properly positioned.

For ADD commands during editing:

Position the cursor to the line above the desired location of the addition. Lines positioned below the cursor will shift down when the new line is added. If the cursor is positioned at the END statement, the new line will be placed one line above END.

For CHANGE OR DELETE commands:

Move the cursor to the line to be changed or deleted.

Use the following commands to move the cursor:

|TOP| - moves the cursor to the top of the file, or to the PINORDER line if the section is highlighted with the asterisk (*) display.

|BOTTOM| - moves the cursor to the bottom of the file, or to the last line of the PINORDER section, if highlighted with the asterisk (*) display.

|UP| - moves the cursor up one line on the file, or up one section. If the PINORDER section is highlighted with the asterisk (*) display, the cursor can only be moved within the section.

|DWN| - moves the cursor down one line, or one section. If the PINORDER section is highlighted with the asterisk (*) display, the cursor can only be moved within the section.

|1ST LINE| - moves the cursor to the first line of the file.

THE DIFFERENCE BETWEEN A SECTION AND A LINE

The system groups the information on the device file into sections of interrelated data. In most cases, a section consists of just one line. The exception to this is the PINORDER section which consists of the PINORDER line, with the PINUSE, PINSWAP and FUNCTION lines beneath it.

TO CHANGE INFORMATION IN THE FILE

Use the CHANGE commands to replace information in the file with other information. For example, use the |CHANGE PACKAGE| command to replace the package name with a new package name.

The |CHANGE PINORDER| command may be used to open the PINORDER section for editing. Skipping the keyboard input, then picking |ENTER| after the (ENTER LIST OF PIN NAMES) prompt will highlight the PINORDER section with the asterisk (*) display. The cursor can then be moved to selected line entries for editing.

NOTE: If the PINORDER section is highlighted, the PINORDER line cannot be deleted. Picking the |DELETE| command with the cursor positioned at the PINORDER line will result in the following system message:

- CANNOT DELETE SECTION HEADER -

However, if the PINORDER section is de-highlighted with the cursor positioned at the PINORDER line, the |DELETE| command eliminates the entire PINORDER section. That is, PINORDER, PINUSE, PINSWAP, and all FUNCTION lines are deleted. The operator may use the UNDELETE command to restore the deleted PINORDER section, or use the |ADD PINORDER| command to rebuild the section.

Use the |CHANGE POWER| command to change the signal name and/or pin number(s). When prompted for a keyboard input, the operator may enter new information, or simply pick ENTER after the prompt to maintain the current entry.

Use the |CHANGE GROUND| command in the same way as the |CHANGE POWER| command.

DELETING INFORMATION FROM THE FILE

The operator may use the |DELETE| and |UNDELETE| commands when editing the file. The |DELETE| command can be used to delete single line entries, or a PINORDER section. However, the PINORDER line (section header) cannot be deleted when the section is highlighted with the asterisk (*) display. If the cursor is positioned at the PINORDER line, with the section de-highlighted, the system deletes the entire section.

Use the |UNDELETE| command to restore the last item deleted (line or section) into the file. The system restores the deleted entry directly below the new cursor position, regardless of the previous line or section location in the file.

CONTINUATION LINES

The operator may add information to existing line entries in the device file with the |ADD CONTINUATION LINE| command. Continuation lines allow additions to selected line entries without changing the existing contents of the line. When the |ADD CONTINUATION LINE| command sequence is complete, the system places a comma after the last entry on the line specified by the cursor, with the new information added to the line directly below.

|ADD CONTINUATION LINE| may also be used when inputting long lines of data exceeding the 59-character limit when creating the file.

EXAMPLE: Existing line in device file specified by the cursor:

```
* _ FUNCTION G1 7400 1 2
```

|ADD CONTINUATION LINE| (ENTER DATA) 3 |ENTER|

Resulting entry in the device file:

```
      FUNCTION G1 7400 1 2,
* _   3
```

USING THE |MERGE| AND THE |BREAK LINE| COMMANDS

|MERGE|

The operator may merge the contents of a continuation line with the line entry directly above it. The |MERGE| command may be used after a continuation line is added to the file. The result is a single line entry in the file, with the continuation line eliminated.

EXAMPLE: Existing line displayed on the graphics screen with a continuation:

```
* _ FUNCTION G1 7400 1 2,
     3
```

|MERGE|

Resulting line displayed on the graphics screen:

```
* _ FUNCTION G1 7400 1 2 3
```

If the cursor is positioned on a line that does not contain a continuation, and if |MERGE| is picked, the following message is displayed on the graphics screen:

- THIS LINE DOES NOT CONTAIN A CONTINUATION -

The system will not merge a continuation line that is too long to fit on the line above it.

|BREAK LINE|

This command allows the operator to break an existing line entry so that a continuation line is created. The system inserts a comma at the operator-defined position on the line, with information after the comma advancing to the next line position. Lines entries below the new continuation line shift down.

EXAMPLE: Existing line contained in the device file, displayed on the graphics screen:

```
* _  FUNCTION G4 7400 12 13 11

|BREAK LINE|  (AFTER WHICH PIN?)  2   |ENTER|

Result:  *     FUNCTION G4 7400 12 13,
         *  _    11
```

Position 2 specifies that the break is to occur after the second pin entry, pin 13. The system inserts a comma, with pin 11 advancing to the next line.

EDITING A DEVICE FILE NOT CREATED WITH THE |EDIT DEVICE FILE| LEADTHRU

Device files created with the Telesis Keyboard, or Function Screen Editor may be edited with the |EDIT DEVICE FILE| leadthru.

When using the |EDIT DEVICE FILE| command to open one of these files, the system first checks it for formatting errors. If errors exist, the system places a message on the function screen. This message states the number of existing faults. The file is then displayed on the graphics screen will all faulty lines bracketed between parentheses ( ) and asterisks (*).

EXAMPLE: (*PACKGE DIP14*)

The line is incorrect because the keyword is misspelled. Line entries in error are displayed as comments and must be deleted prior to using the ADD commands to specify the correct information.

C R E A T I N G   A   N E T   D A T A   B A S E

- THE NET DATA BASE
- CREATING A NET DATA BASE FROM A SCHEMATIC DRAWING
- CREATING A NET DATA BASE FROM A TEXT-INPUT NETLIST
- EDITING A NET DATA BASE WITH THE INCREMENTAL NET-LOAD TEXT FILE
- ADDITIONAL METHODS USED TO EDIT THE NET DATA BASE
- BACK ANNOTATION
- THE NETLIST-REPORT, COMPONENT-REPORT AND BILL-OF-MATERIALS REPORT
- LOGICAL DESIGN RULES CHECKING

THE NET DATA BASE

1. IMPORTANCE OF THE NET DATA BASE

   o DEFINITION

   o PURPOSE

   o ACCURACY OF THE NET DATA BASE

2. FORMAT AND FILE MANAGEMENT OF THE NET DATA BASE

   o FORMAT OF THE NET DATA BASE FILE

   o THE NET DATA BASE LINKED TO THE BOARD DRAWING

   o FILE MANAGEMENT OF THE NET DATA BASE

1. IMPORTANCE OF THE NET DATA BASE

DEFINITION

The net data base is a file created by the system when it has collected and organized all the information it has about nets, pins, reference designators, functions, and device types to be used in a board design.

Using the EDA-3000 software, you can create a net data base in one of two ways:

1. FROM A SCHEMATIC: The operator may annotate the symbols used in a schematic drawing with device type, pin number, pin name and reference designator labels. With device description files present, the operator may simply annotate the device type to each logic symbol.

   The |EXTRACT NETLIST| command is then used to create the net data base.

   Optionally, the |EXTRACT NETLIST| command will extract information from device files as well as the schematic. Device files are, however, required if the operator wishes to construct a net data base which allows component, logic function, and pin swapping on the board. This capability may be used to improve the placement of components, functions, and pins prior to starting the auto-router.

2. FROM A TEXT-INPUT NETLIST: The operator may create a text file containing net, pin, reference designator, and device type information to be used in the board design. Optionally, functions may be specified in this text file if the operator wishes to use the automatic assignment capability when creating the net data base.

   Once this text file has been created, the |LOAD TXT NETLIST| command may then be used to create the net data base. With device description files present, the resulting net data base will allow component, function, and pin swapping on the board.

When you pick |EXTRACT NETLIST| or |LOAD TXT NETLIST|, the system gathers all the essential information from the schematic or the text-input netlist. During the creation of the net data base, the system assigns a net number to each net.

If you have available an EDA-1000 system, you can run the netlist processor on that system to create a netlist. You can then send this netlist to the EDA-3000; the system will treat it in the same way as a netlist created on these systems.

PURPOSE

The net data base is the foundation for most of the system's automated capabilities:

CREATING THE NETLIST-REPORT AND THE COMPONENT-REPORT
CREATING THE BILL OF MATERIALS REPORT
LOGICAL DESIGN RULES CHECKING
AUTOMATIC AND INTERACTIVE PLACEMENT
RATSNEST
COMPONENT, LOGIC FUNCTION, AND PIN SWAPPING
AUTO-ROUTER
NET COMPARE
BACK ANNOTATION
ARTWORK GENERATION

Most of the board design work on the Telesis system is in two phases:

PHASE 1: Putting the information INTO the net data base through the schematic or text-input netlist.

PHASE 2: Informing the system to take information FROM the net data base to use in the automated processes mentioned above.

ACCURACY OF THE NET DATA BASE

It is important that the information used to create the net data base is accurate prior to using the system's automated capabilities.

The sections of the manual covering library files, schematics, and the text-input netlist, provide information about the essential requirements. These sections also explain how the system can assist the operator in checking these text files and drawings for accuracy.

2. FORMAT AND FILE MANAGEMENT OF THE NET DATA BASE

FORMAT OF THE NET DATA BASE FILE

The net data base file is in a format that can only be understood by the computer. For this reason, the operator cannot print the net data base or view it on the graphics screen. To print or view the information in the net data base, a NETLIST-REPORT or COMPONENT-REPORT must be created.

A NET DATA BASE IS LINKED TO A BOARD DRAWING BY AN IDENTICAL REVISION LABEL.

When the operator first uses the |EXTRACT NETLIST| or |LOAD TXT NETLIST| command, the system creates "file NET-DATA-BASE rev ****" in the current project file. Only the temporary revision label "****" is used because a corresponding board drawing does not yet exist.

When a new board drawing is opened in the same project, the system links the net data base to the board drawing, so that all information in the net data base can be applied to the board. Each time a board drawing is closed, the system assigns the board drawing revision label to the net data base file. If several revisions of a board drawing exist, a net data base will be linked to each revision. In such a case, the project index may look like this:

drawing PCB rev 1
file NET-DATA-BASE rev 1
drawing PCB rev 2
file NET-DATA-BASE rev 2
drawing PCB rev 3
file NET-DATA-BASE rev 3

If the operator updates the net database by repeating the |EXTRACT NETLIST| or |LOAD TXT NETLIST| command, the system will update the most recently used revision of the net data base in the current project file. That is, it will update the net data base linked to the most recently opened board drawing. To avoid confusion, it is good practice to keep only one revision on a board drawing and a net data base file in the current project file. If earlier versions are needed, copy them to floppy disk or tape.

Whenever a board drawing is copied or deleted, the net data base file linked to it is also copied or deleted. If the revision label assigned to a board drawing is changed, the system also changes the net data base revision label to match the board drawing.

A net data base may be separated from its board drawing with the following commands:

| COPY NET-DB |
| DELETE NET-DB |
| CHG NET-DB RV |
| NET-DB FROM TAPE |
| NET-DB TO TAPE |

Use these commands only in cases when the net data base to board drawing link is no longer required.

Net data base files always have the name: NET DATA BASE. The name cannot be changed. The revision label can be changed, however.

CREATING A NET DATA BASE FROM A SCHEMATIC DRAWING

1. GENERAL INFORMATION
   - o PURPOSE
   - o WHERE THE SCHEMATIC FITS IN THE OVERALL DESIGN PROCESS

2. THE SCHEMATIC DRAWING AND THE NET DATA BASE
   - o GENERAL INFORMATION

3. LOGIC SYMBOLS AND DEVICE DESCRIPTION FILES
   - o LOGIC SYMBOLS
   - o DEVICE DESCRIPTION FILES

4. PLANNING, DRAWING AND ANNOTATING THE SCHEMATIC

5. USING AND CONTROLLING AUTOMATIC ASSIGNMENT

6. REQUIRED ANNOTATION OF LOGIC SYMBOLS FOR AUTOMATIC ASSIGNMENT
   - o DEVICE TYPE LABELS
   - o ATTACHING FUNCTION TYPES

7. CONTROLLING REFERENCE DESIGNATOR ASSIGNMENT
   - o CREATING THE REFDES-CONTROL FILE
   - o FORMAT OF THE REFDES-CON TEXT FILE
   - o INPUTTING THE REFDES-CON TEXT FILE

8. PRE-ASSIGNING LOGIC FUNCTIONS ON THE SCHEMATIC
   - o CREATING THE NET DATA BASE WITHOUT DEVICE DEVICE DESCRIPTION FILES
   - o COMMANDS USED TO MAKE ASSIGNMENTS TO LOGIC SYMBOLS ON THE SCHEMATIC
   - o REFERENCE DESIGNATORS AND PIN NUMBER LABELS

- o DEVICE TYPE LABELS
- o SIGNAL NAME LABELS
- o POWER AND GROUND PINS

9. THE AUTO-ASSIGNMENT / PRE-ASSIGNMENT STRATEGY

- o SUMMARY
- o COMPONENT, FUNCTION, AND PIN SWAPPING OVERVIEW

10. HOW TO USE THE EXTRACT NETLIST COMMAND

- o PREREQUISITES
- o HOW THE EXTRACT NETLIST COMMAND WORKS
- o HOW THE EXTRACT NETLIST COMMAND PERFORMS AUTOMATIC ASSIGNMENT
- o THE EXTRACT NETLIST SEQUENCE
- o THE EXTRACTION-LOG TEXT FILE
- o TYPES OF ERROR AND WARNING MESSAGES SHOWN IN THE EXTRACTION-LOG

1. GENERAL INFORMATION

PURPOSE

Drawing a schematic is the customary way to create a net data base on the Telesis system.

The schematic is also a form of permanent documentation when it it plotted or archived (on floppy disk or tape).

WHERE THE SCHEMATIC FITS IN THE OVERALL DESIGN PROCESS

PREREQUISTIES

Logic Symbol Library

Device File Library

or

Pin File Library

------->

CREATING A SCHEMATIC-GENERATED NET DATA BASE

STEP 1. Create schematic showing required netlist information.

STEP 2. Use the EXTRACT NETLIST command.

OPERATIONS THAT USE THE SCHEMATIC GENERATED NET-DATA-BASE

- O LOGICAL DESIGN RULES CHECKING
- O AUTOMATIC, INTERACTIVE, MANUAL PLACEMENT
- O RATSNEST
- O COMPONENT, GATE, AND PIN SWAPPING
- O AUTO-ROUTING
- O MANUAL INTERCONNECTION
- O NET COMPARE
- O PHYSICAL DESIGN RULES CHECKING
- O BACK ANNOTATION
- O ARTWORK
- O NC-DRILL
- O BILL OF MATERIALS GENERATION
- O CREATE TEXT NETLIST
- O CREATE NETLIST-REPORT AND COMPONENT-REPORT

2. THE SCHEMATIC DRAWING AND THE NET DATA BASE

GENERAL INFORMATION

The operator can draw a schematic which allows automatic assignment of schematic logic functions to physical board component symbols during the EXTRACT NETLIST command. With device description files present in the current project or the SYSTEM-LIBRARY, the resulting Net Data Base can then be used for selective swapping of board components, logic functions, and component pins on the schematic drawing and the board drawing. This automatic assignment feature improves the board drawing for automatic placement and auto routing.

NOTE

> If you are planning to use the ECL Toolbox option to route high frequency transmission lines, refer to the ECL Toolbox section in the Options section of this manual, subsection 2.1.A, prior to creating a schematic drawing.

During the creation of the schematic, the operator may annotate logic symbols using conventional methods, or the simplified methods described in this section. When the EXTRACT NETLIST command is used to create the Net Data Base, the system gathers the information from the schematic and device description files and creates a file that supports many automated features of the Telesis system.

This section describes the procedures to create a Net Data Base from a schematic drawing and device description files. Procedures to allow the system to assign physical board components to unassigned logic functions in the schematic drawing are also detailed.

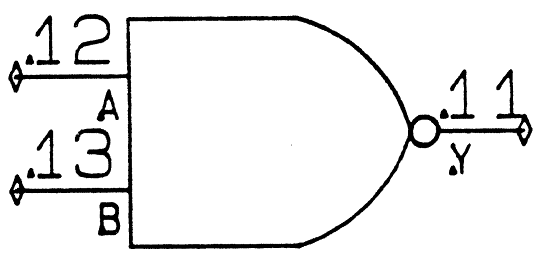
3. LOGIC SYMBOLS AND DEVICE DESCRIPTION FILES

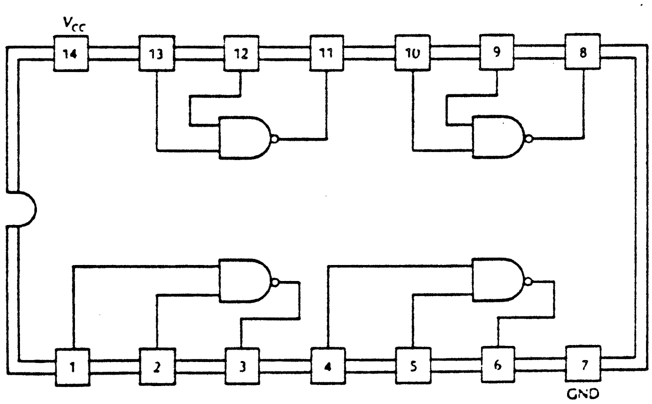
Logic Symbols

A logic symbol is a symbol on a schematic drawing that represents an electronic device to be placed on a board drawing. A logic symbol may also be part of the electronic device. For example, NAND2 symbols are logic symbols; four of them fit on one 7400 device. CAPACITOR symbols are logic symbols; usually one per physical device.

- o Each logic symbol represents an electronic FUNCTION, such as a NAND2 gate. With the 7400 device, for example, up to four logic functions (NAND2) may represent one 7400 device (PACKAGE DIP14) on a board.

EXAMPLE: THE NAND2 GATE AND THE 7400 DEVICE



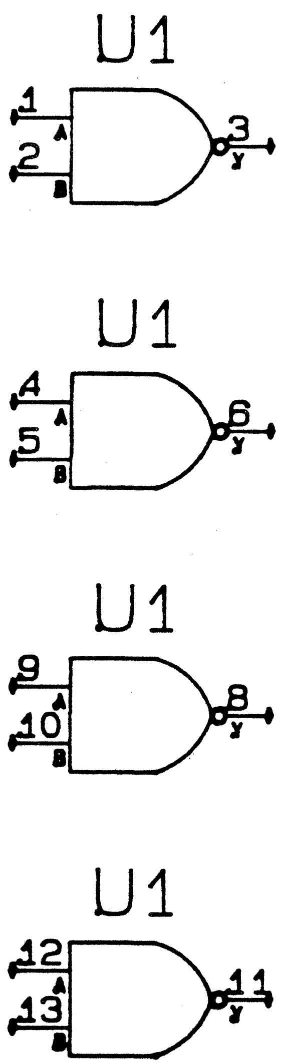


- o Every device must be assigned to a package symbol on the board drawing. For example, the 7400 device is assigned to the package DIP14. The package symbols on the board drawing represent the electronic COMPONENTS of the design, each with a REFERENCE DESIGNATOR as its unique name.

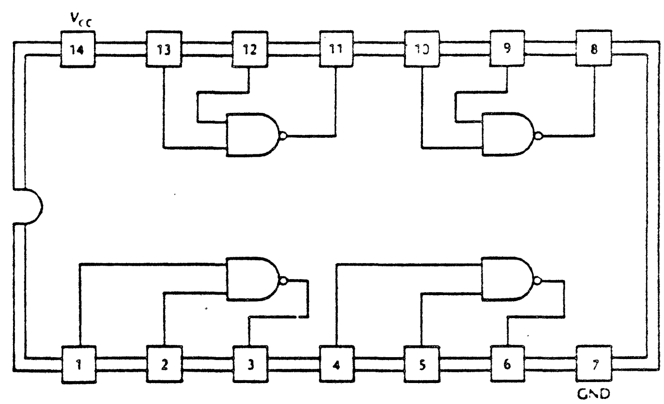
  The logic FUNCTIONS within a 7400, for example, may contain up to four identical REFERENCE DESIGNATORS on the schematic, representing each identical logic FUNCTION associated with the 7400. For example, each NAND2 gate may contain REFERENCE DESIGNATOR, U1, specifying that four NAND2 functions are used within one 7400 device, represented by one DIP14 symbol (U1) on the board.

EXAMPLE: PACKAGE DIP14 (DEVICE 7400) REPRESENTED BY REFERENCE DESIGNATOR U1 WITH THE FOUR ASSOCIATED NAND2 FUNCTIONS.

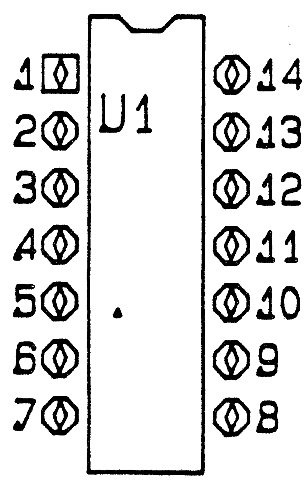
LOGIC SYMBOLS ------------>



DEVICE 74ØØ



PACKAGE DIP14



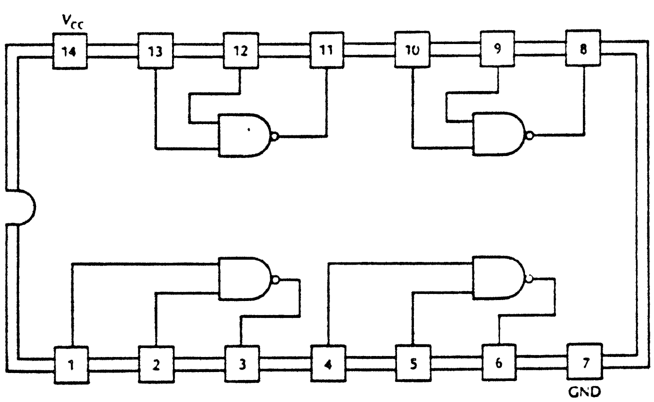
Device Description Files

Device description files provide information to the system about each device (i.e. 7400), its logic functions (NAND2), and all pin information. The device description file contains the following:

- o The physical PACKAGE. (DIP14)
- o The CLASS associated with the device. (IC, IO or DISCRETE)
- o The number of pins contained within the device, or the PINCOUNT.
- o The types of logic FUNCTIONS contained within the device, and how their logical pins correspond to the pin numbers of the device.
- o The use of each pin contained on the device. (IN, OUT, etc.)
- o The swappable pins for each type of function contained in the device.
- o The connection of power and ground pins.

The example below illustrates the 7400 device shown in the TTL Data Book for Design Engineers and the Telesis device description file.

DEVICE 7400

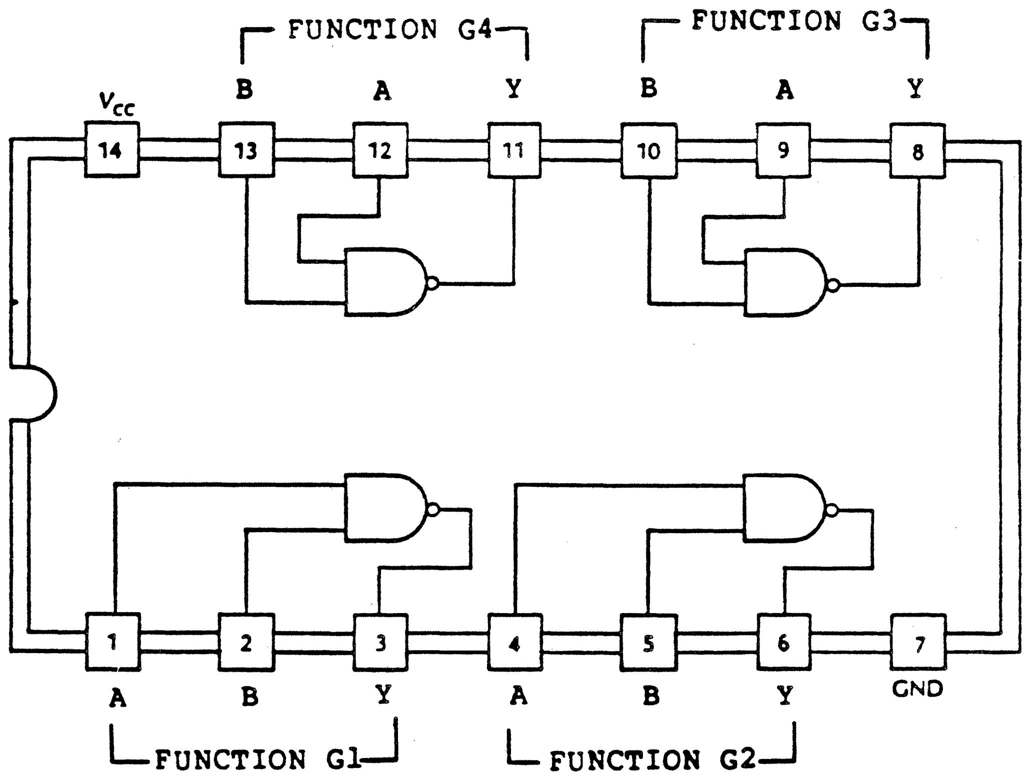


DEVICE DESCRIPTION FILE

```
(DEVICE DESCRIPTION FILE: 7400)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER 7400 A B Y
PINUSE 7400 IN IN OUT
PINSWAP 7400 A B
FUNCTION G1 7400 1 2 3
FUNCTION G2 7400 4 5 6
FUNCTION G3 7400 9 10 8
FUNCTION G4 7400 12 13 11
POWER +5V; 14
GROUND GND;7
END
```

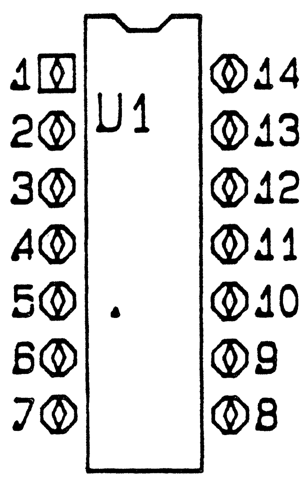
The information for many device description files can be acquired from the The TTL Data Book for Design Engineers. This book illustrates devices by "family" (i.e. TTL, ECL, etc.). For example, the 7400 device (TTL) pictured below, shows the four NAND2 gates and their appropriate function slots, the 7400 device file, and the board component symbol DIP14.

DEVICE 7400



DEVICE DESCRIPTION FILE -------> SYMBOL DIP14

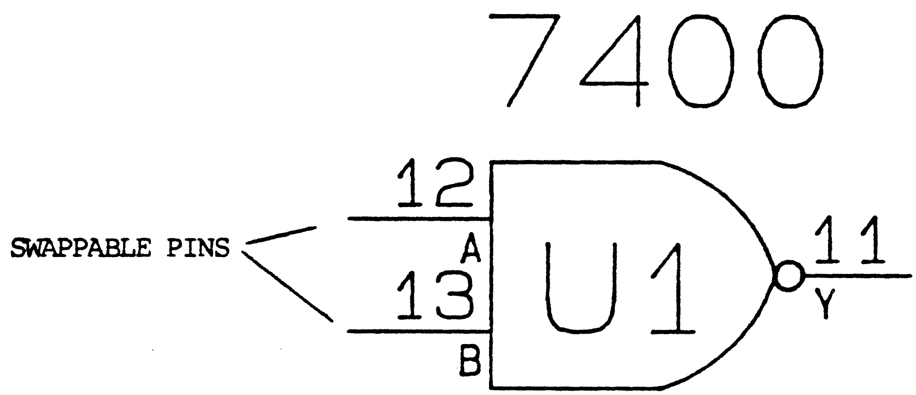
```
(DEVICE DESCRIPTION FILE: 7400)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER 7400 A B Y
PINUSE 7400 IN IN OUT
PINSWAP 7400 A B
FUNCTION G1 7400 1 2 3
FUNCTION G2 7400 4 5 6
FUNCTION G3 7400 9 10 8
FUNCTION G4 7400 12 13 11
POWER +5V; 14
GROUND GND;7
END
```



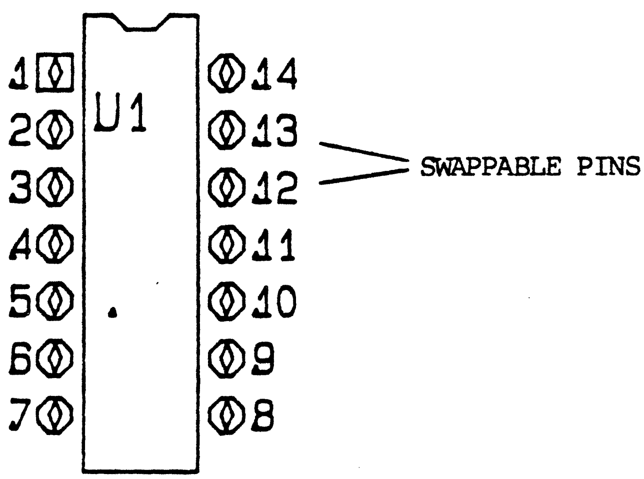
Device description files are required for automatic assignment of logic functions when the |EXTRACT NETLIST| command is used to create the net data base from the schematic drawing. A net data base created from device files allows swapping of components, logic functions, and swappable pins on the board during and after initial placement.

In the example shown on the previous page, the PINORDER line specifies the pin names "A" and "B", with PINUSE, "IN" "IN", respectively. Therefore, since the pin names "A" and "B" are both specified as "IN", swappable pins exist for this FUNCTION type specified in the file. For example, FUNCTION G1 contains the swappable pins 1 and 2; FUNCTION G2 contains the swappable pins 4 and 5, etc.

EXAMPLE: FUNCTION G4 (NAND2,DEVICE 7400) ON THE SCHEMATIC



PACKAGE DIP14 (DEVICE 7400) ON THE BOARD DRAWING



With device description files present in the current project, or the SYSTEM-LIBRARY, the operator does not need to fully annotate logic symbols on the schematic. During |EXTRACT NETLIST|, the system reads the information from the device files, and automatically assigns all unassigned functions to the required components in the net data base.

If the operator chooses to fully annotate each logic symbol on the schematic, the devices files, if present, are checked against the assignments on the schematic. If descrepancies do not exist, the assignments on the schematic will override the contents of the device description files. The system uses the assigned information on the schematic when |EXTRACT NETLIST| creates the net data base.

In addition, the swapping capability on the board drawing can only be used if device filesare present in the current project file or SYSTEM-LIBRARY during |EXTRACT NETLIST|. Pre-assigned logic functions on the schematic are FIXED when the operator places the board. That is, FIXED logic functions are unavailable for automatic swapping until FREED by the operator.

4. PLANNING, DRAWING AND ANNOTATING THE SCHEMATIC

The steps below outline the necessary steps in drawing and annotating the schematic.

STEP 1. CHECK THE SYSTEM LIBRARY

Check the SYSTEM-LIBRARY to ensure that all logic symbols and device description files needed for the schematic are present. If not, first create the logic symbols, or copy them to the system from tape or floppy diskette.

STEP 2. PLAN THE SCHEMATIC SHEETS

Plan the number of sheets required for the schematic. It is good practice to sketch each sheet when planning the logic symbol placement. This will allow the operator to work faster during the creation of the schematic.

STEP 3. LOGIC SYMBOL PLACEMENT AND INTERCONNECTION

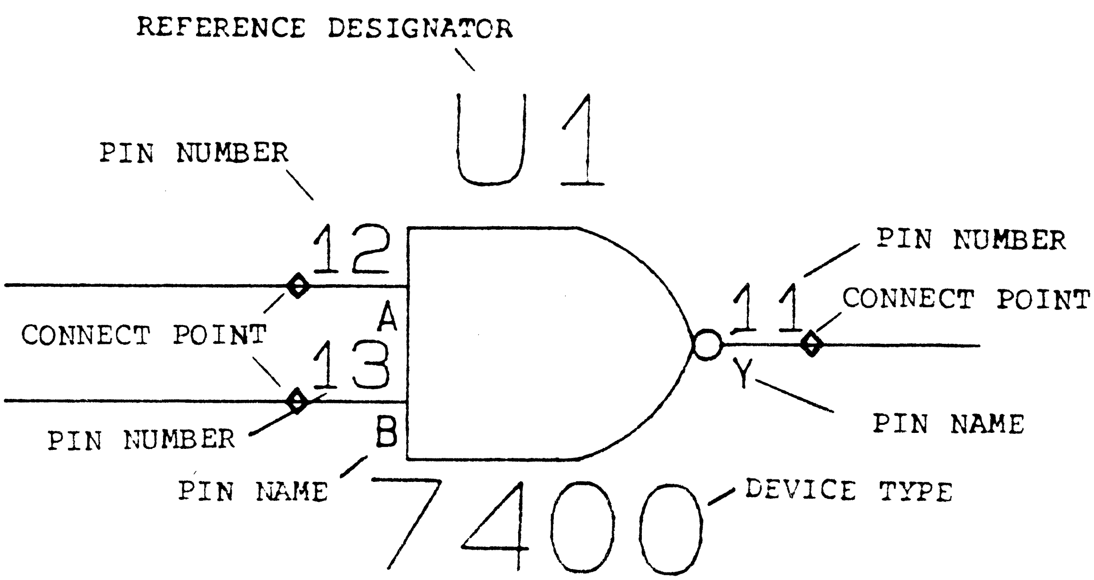
For each sheet of the schematic, place the logic symbols, then interconnect them according to the original engineering sketch. A good strategy is placement of the larger symbols first, then the smaller ones. After placement of the logic symbols, proceed to use the |ADD CONNECTION| command to interconnect the symbols.

STEP 4. ANNOTATING THE LOGIC SYMBOLS

The diagram below illustrates a fully annotated logic symbol. For example, to represent the NAND2 gate of a 7400, the symbol contains:

- o Three connect points, one for each pin of the gate;
- o One reference designator;
- o One device type attached;
- o Each pin may have one pin number attached;
- o Each pin may have one pin name attached.

THE ANATOMY OF A LOGIC SYMBOL



5. USING AND CONTROLLING AUTOMATIC ASSIGNMENT

The automatic assignment feature assigns reference designators and pin numbers to unassigned logic symbols in the schematic drawing. To use the automatic assignment feature, there must be a Device File for each device type used in the schematic drawing in the current project or in the SYSTEM-LIBRARY.

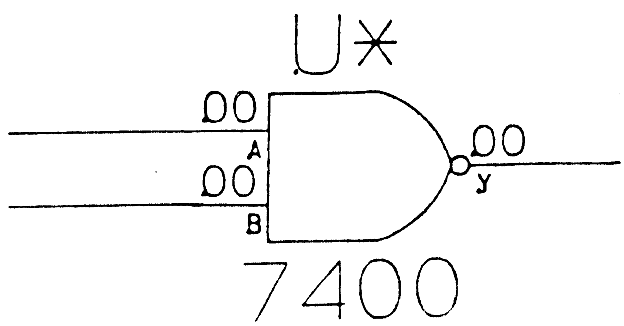
CAUTION

> The schematic logic symbol must have a device type label and pin names attached to it. If the symbol name is the same as the device type, the device type label can be omitted.

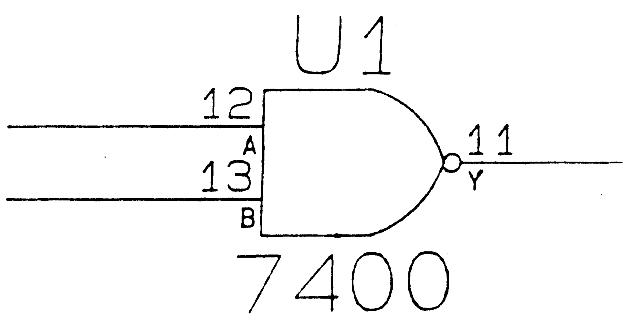
When the operator selects EXTRACT NETLIST, the system automatically links all unassigned logic symbols on the schematic drawing to the corresponding component in the Net Data Base by assigning a reference designator to the symbol. The system also assigns physical pins on the component to logical pins in the schematic drawing from information obtained in the Device File.

The following figures illustrate an unassigned and preassigned function of the NAND2 (device type 7400) logic function.

EXAMPLE OF UNASSIGNED LOGIC SYMBOL (device file mandatory)



EXAMPLE OF PREASSIGNED LOGIC SYMBOL (device file mandatory)



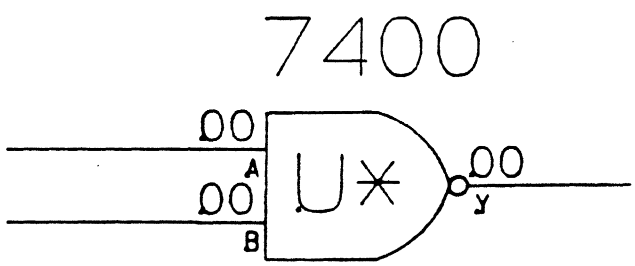
DEVICE TYPE LABELS

With device description files in the current project, or in the SYSTEM-LIBRARY, the operator only needs to annotate the device type label to each logic symbol on the schematic. When the |EXTRACT NETLIST| command creates the net data base, the system automatically assigns all unassigned functions to the required components in the net data base. Use the |UPDATE TEXT| command to update the ´DEV´ field on the logic symbol with the appropriate device type label (i.e. 7400).

NOTE: If the symbol name is the same as the device file name, the device type label on the schematic symbol may be omitted.

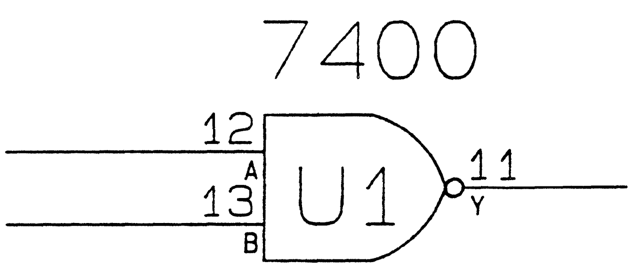
The NAND2 logic function shown below illustrates the the required annotation for automatic assignment with device description files.

On the schematic before |EXTRACT NETLIST| ----



The net data base after |EXTRACT NETLIST| ----

DEVICE TYPE ATTACHED



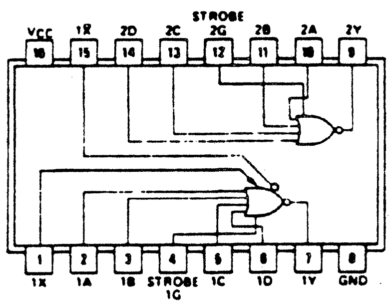

```
(DEVICE FILE: 7400)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER 7400 A B Y
PINUSE 7400 IN IN OUT
PINSWAP 7400 A B
FUNCTION G1 7400 1 2 3
FUNCTION G2 7400 4 5 6
FUNCTION G3 7400 9 10 8
FUNCTION G4 7400 12 13 11
POWER +5V; 14
GROUND GND; 7
END
```

This NAND2 gate represents one FUNCTION to |EXTRACT NETLIST|. It could fit in any of the four FUNCTION slots (G1, G2, G3, G4). The pin numbers show that |EXTRACT NETLIST| assigned this logic symbol to FUNCTION G4. For this gate, pin name "A" corresponds to pin number 12; pin name "B" corresponds to pin number 13; pin name "Y" corresponds to pin number 11.

The PINORDER line and the FUNCTION lines of the device file use the function type "7400", the same as the device type. This is most convenient when all the functions in a device are the same type. The operator must annotate the function type on the logic symbol, and in the device file, if there are two or more different types of functions on the device.

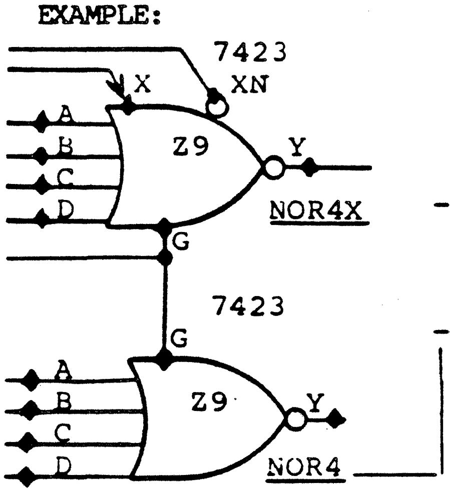
ATTACHING FUNCTION TYPES

If a device contains one or more different functions, and if the operator attaches the device type only (i.e. 7423), the |EXTRACT NETLIST| command will not know which logic symbol represents which function on the device. For example, the 7423 device contains both the NOR gate, and the expanded NORX gate, shown below.



By attaching the function type, NOR4, to the symbol on the schematic representing the 4-input NOR, and NOR4X to the symbol representing the extended 4-input NOR, the system will differentiate the multiple functions on the 7423 device during |EXTRACT NETLIST|. The system then automatically assigns all unassigned functions from the information in the device description file.

EXAMPLE:




```
(DEVICE FILE: 7423)
PACKAGE DIP16
CLASS IC
PINCOUNT 16
PINORDER NOR4X A B C D G X XN Y
PINUSE NOR4X IN IN IN IN IN IN IN OUT
PINSWAP NOR4X A B C D
FUNCTION FS1 NOR4X 2 3 4 5 6 4 1 15 7
PINORDER NOR4 A B C D G Y
PINUSE NOR4 IN IN IN IN IN OUT
PINSWAP NOR4 A B C D
FUNCTION FS2 NOR4 10 11 13 14 12 9
POWER +5V; 16
GROUND GND; 8
END
```

The operator may use the |ADD FUNCTION TYPE| command when creating a logic symbol, or the |ATTCH FNCTN TYPE| command during the creation of the schematic. Generally, it is more convenient to add the function type to the original symbol drawing. That is, the logic symbol will always maintain the required function type.

7. CONTROLLING REFERENCE DESIGNATOR ASSIGNMENT

Reference designators are OPTIONAL labels, and are not required if the net data base is to be created with device files. During the |EXTRACT NETLIST| command, the system automatically assigns reference designators to the required components in the net data base.

However, the operator may create a text file named REFDES-CON. This is a text file that specifies the reference designators to be used by the system when |EXTRACT NETLIST| creates the net data base. The REFDES-CON can be generated from a bill-of-materials, using specific reference designators.

Without a REFDES-CON file present in the project directory, the system automatically assigns reference designators using a system counter. For example, if the schematic reflects logic functions for ten IC components, the system assigns reference designators,

U1 , U2, U3, U4, U5, U6, U7, U8, U9, U10

to the net data base. The reference designators will appear on the IC component symbols during INTERACTIVE or AUTOMATIC placement.

However, if a REFDES-CON file exists in the project directory, the system will read the file during |EXTRACT NETLIST|, and assign the operator specified reference designators to the components in the net data base. The example below illustrates a typical REFDES-CON text file.

```
(REFDES-CON FILE FOR PROJECT: TEST)
7400; U1-U10
CONNECTOR; J6
RESISTOR; R1-25
CAPACITOR! 20PF; C1-8
CAPACITOR; C9-25
7410; U17
7404; Z12-14
7454; Z11
```

In the REFDES-CON file illustrated above, for example, the system assigns reference designator J6 to the CONNECTOR. Without a REFDES-CON text file, the system would assign J1 to the CONNECTOR during |EXTRACT NETLIST|.

CREATING THE REFDES-CONTROL FILE

To create the REFDES-CON text file, the operator may use the Telesis Function Screen editor, or the optional Telesis Keyboard Text Editor (VT101 or WYSE). Refer to the BASICS section of the manual for the required procedures for creating and editing text files on the function screen, or with the optional keyboard editor.

FORMAT OF THE REFDES-CON TEXT FILE

The format of the REFDES-CON is described below. The file must be named REFDES-CON. During |EXTRACT NETLIST|, the system looks for the the text file named REFDES-CON when assigning reference designators to the net data base. If the file does not exist, or if it is named other than REFDES-CON, the system automatically assigns the default reference designators from a programmed counter.

INPUTTING THE REFDES-CON TEXT FILE

STEP 1. Device type; list of reference designators

When inputting the contents of the file named REFDES-CON, list the device type and the reference designators to be used for that device type.

EXAMPLE: 7404; U17-22

Input Sequence:

- o Type-in the device type followed by a semi-colon.
- o Input a blank space, then type the reference designators to be used by the system for that device type.
- o A series of reference designators may be indicated with a hyphen (-). For example, U17-22 indicates the reference designators U17, U18, U19, U20, U21, and U22.
- o Separate non-series entries with a comma (,) or a blank space. For example,

  7404; U17, U19 U20

In the above example, reference designators U17 through U20 will be assigned to the three components with the device type label 7404.

NOTE: If the list of reference designators is longer than can fit on one line, the operator may continue the list by inputting a comma at the end of the first line, and continuing on the next.

STEP 2. Device type! value; list of reference designators

EXAMPLE: CAPACITOR! 20pf; C1-8

If a device type contains a component value, type an exclamation point after the device type, then the component value followed by a semi-colon, then the list of reference designators. A value is optional.

STEP 3. Open-series reference designators

EXAMPLE: 7454; Z*

The operator may indicate an "open-series" of reference designators by a reference designator ending with an asterisk (*). The system will assign as many reference designators in the series as it needs. For example, Z*, will cause reference designators Z1, Z2, Z3,...as needed.

If the reference designator RP* is specified in the REFDES-CON file, and if reference designators RP1 and RP3 were pre-assigned on the schematic drawing, the system will start the series with RP2, then proceed with RP4 as needed until all device types with the label RP* have been assigned.

STEP 4. Comment lines

(REFDES-CON FILE FOR DRAWING SCH REV 1)

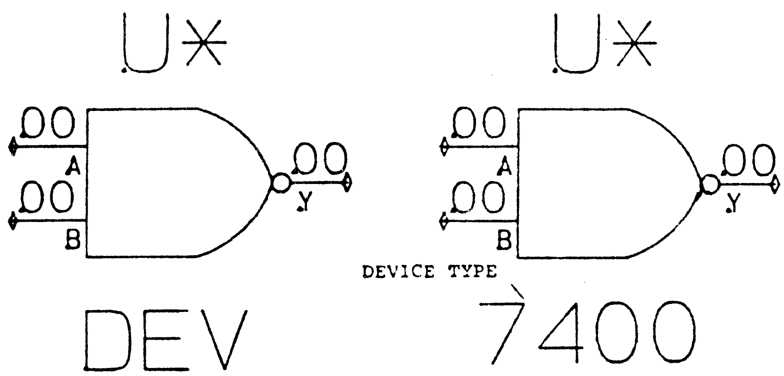
Enclose comment lines in parentheses. When the system reads the file, comment lines are ignored.

During |EXTRACT NETLIST|, if the system does not find the text file REFDES-CON, assignment of references designators will occur as "next in the series". That is, if references designator U2 was pre-assigned, the system will automatically assign U1, then U3, U4, U5, etc.

8. PRE-ASSIGNING LOGIC FUNCTIONS ON THE SCHEMATIC

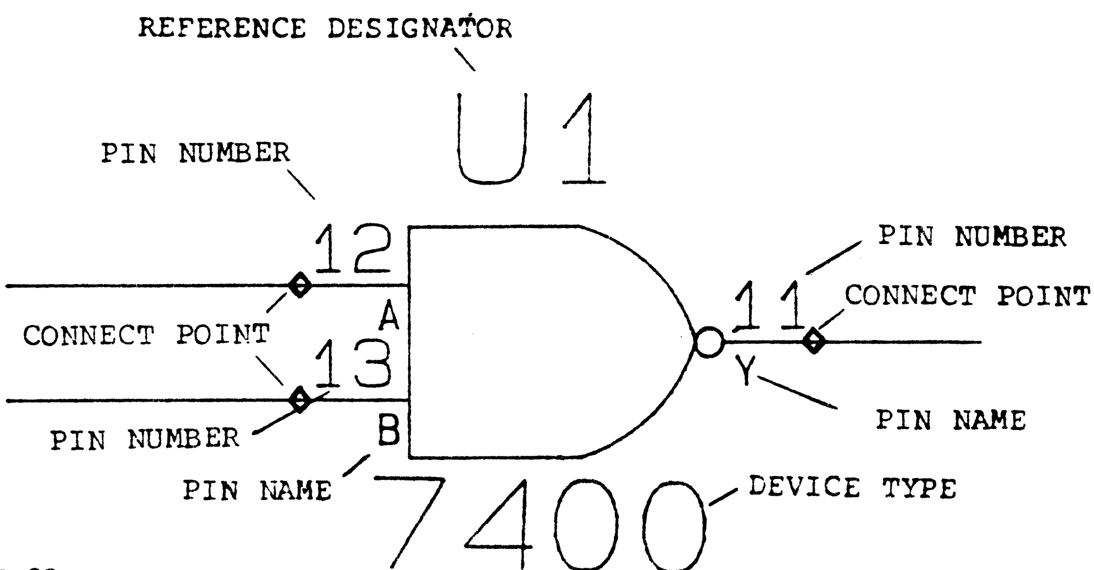
When the |ADD SYMBOL| command is used to place all logic symbols on the schematic drawing, the operator may simply use the |UPDATE TEXT| command to update the 'DEV' field on symbol with the appropriate device type. With device description files in the current project, or in the SYSTEM-LIBRARY, the system will automatically assign functions to components in the net data base during |EXTRACT NETLIST|.

EXAMPLE: ADD SYMBOL NAND2   NAND2 WITH DEVICE TYPE



However, the operator may selectively pre-assign logic functions on the schematic with device description files present. When the |EXTRACT NETLIST| command is picked, the system simply proceeds to perform assignments to those unassigned functions when the net data base is created.

To pre-assign logic functions on the schematic prior to using the |EXTRACT NETLIST| command, use |UPDATE TEXT| to make the assignments to the existing fields on the symbol. However, if a text field does not appear on an operator-created symbol, the ATTACH commands may be used to make the assignments to the logic symbol. The illustration below shows a fully assigned logic symbol (NAND2, DEVICE 7400).



CREATING THE NET DATA BASE WITHOUT DEVICE DESCRIPTION FILES

If device description files are not used in the creation of the net data base, logic symbols must be pre-assignedon the schematic drawing prior to using the |EXTRACT NETLIST| command.

COMMANDS USED TO MAKE ASSIGNMENTS TO LOGIC SYMBOLS ON THE SCHEMATIC

|UPDATE TEXT|

Use the |UPDATE TEXT| command to change existing text fields on on the logic symbol. This command changes the text only. The type of label remains the same. When |UPDATE TEXT| is used, the old text is replaced with the new text that the operator inputs on the keyboard menu.

THE ATTACH COMMANDS

Use the ATTACH commands to add assignment information to logic symbols on the schematic that do not contain the required text fields. For example, if a symbol does not have a device type label ('DEV'), use the |ATTCH DVICE TYPE| command to assign a device type label to that instance of the symbol. The commands used to attach text labels to a schematic symbol are listed below:

ON THE SCHEMATIC	EXTRACT WITHOUT DEVICE FILES
|ATCH SIG NAME LN|	SIGNAL NAME LINES (OPTIONAL)
|ATCH SIG NAME PN|	SIGNAL NAME PINS (OPTIONAL)
|ATTCH DVICE TYPE|	DEVICE TYPE LABEL (OPTIONAL)
|ATTCH PIN NAME|	PIN NAMES (OPTIONAL)
|ATTCH PIN NUMBER|	PIN NUMBERS (REQUIRED)
|ATTCH REF DES|	REFERENCE DESIGNATORS (REQUIRED)
|ATTCH VALUE|	VALUE LABEL (OPTIONAL)
|ATTCH FNCTN TYPE|	FUNCTION TYPE (OPTIONAL)
|ATTCH PINUSE CODE|	PIN USE CODE (OPTIONAL)

Refer to COMMANDS section of the manual for additional information on each command used for attaching text to symbols on a schematic drawing.

NOTE: When creating schematic symbols, use the following commands to create "dummy" text fields on the symbol drawing.

|ADD REF DES| |ADD FUNCTION TYPE|
|ADD DEVICE TYPE| |ADD VALUE|

|ADD REF DES| |ADD FUNCTION TYPE|
|ADD DEVICE TYPE| |ADD VALUE|

For additional "dummy" fields on the symbol drawing, use the ATTACH commands specified above.

REFERENCE DESIGNATOR AND PIN NUMBER LABELS (REQUIRED)

The operator may annotate the schematic symbols with reference designators and pin numbers at any time during the creation of each schematic sheet. It is good practice to place the symbols and connect lines prior to assigning reference designators and pin number labels.

The symbols placed in the schematic from the symbol library should already have dummy reference designators and pin number labels. Use the |UPDATE TEXT| command to assign the appropriate text labels.

Common Pins (Optional)

The operator may annotate two or more symbols with the same reference designator, and put the same pin number on a connect point in each of them. This capabililty is used to show common pins on each instance of a device, such as CLEAR LINES, and COMMON VOLTAGES. For example, a set of resistors in a resistor pack are all tied to a common ground by pin 1.

Note on Pin Number Labels

If device description files exist in the current project, or in the SYSTEM-LIBRARY, the operator may omit all but one pin number label from each logic symbol on the schematic. When the |EXTRACT NETLIST| command is picked, the system will collect the other pin number labels for each symbol from the device description files if each symbol is labeled with the DEVICE TYPE and PIN NAMES. Later, the operator may use the BACK ANNOTATE commands to automatically place the pin numbers on the appropriate schematic symbols.

DEVICE TYPE LABELS (OPTIONAL)

Device type labels are required if:

- power and ground pins are shown on a separate sheet.
- device description files are used to create the net data base, allowing automatic assignment of logic functions to components created in the Net Data Base.

SIGNAL NAME LABELS (OPTIONAL)

The operator may establish any set of connect-lines to be part of a particular signal net in one of two ways:

1. Attach a signal name label to any of the connect-lines dedicated to that net. Use the |ATCH SIG NAME LN| command.

2. Attach a signal name label to any of the pins (connect-points) of the net. Use the |ATCH SIG NAME PN| command.

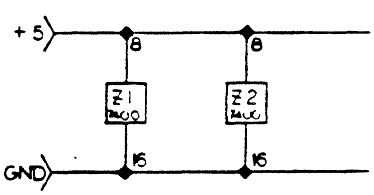
Signal names are not required, except in 2 cases:

1. If connect-lines are not placed in a schematic, the operator must show connections by attaching the same signal name to all the pins in the same net.

2. To show connections between sheets, the operator must attach the same signal name to the connect-line on each sheet.

POWER AND GROUND PINS

Since the operator may not want to show power and ground pins on the schematic, a separate sheet may be created. The example below illustrates the method of showing the information on a separate sheet.



Use the |ATCH SIG NAME LN| command to attach names to the power and ground connect lines, and connect the lines to the pins of dummy symbols having the correct reference designators, device type labels, and pin numbers.

Omitting Power and Ground Pins

Power and ground pins are not required on the schematic if the information is contained in the device description files. If the information is placed on the schematic and in the device description files, the system will "prefer" the information on the schematic when the net data base is created, and will ignore the power and ground data in the device description files.

NOTE: When placing elements on the schematic drawing, refer to the Telesis Layer Standards Table (BASIC-30) for the proper layer settings. It is not necessary to set the ACTIVE layer when using the |UPDATE TEXT| command. New text entries will automatically be placed on the same layer as the original text point.

9. THE AUTO-ASSIGNMENT/PRE-ASSIGNMENT STRATEGY

SUMMARY

After adding and interconnecting the logic symbols on the schematic, the operator may selectively pre-assign functions, or simply use the automatic assignment capability of the Telesis system.

With automatic assignment, device description files must be in the current project, or in the SYSTEM-LIBRARY. The DEVICE TYPE label must be annotated to each logic symbol on the schematic drawing. However, the operator may selectively pre-assign any number of logic functions on the drawing prior to creating the net data base with the |EXTRACT NETLIST| command. These pre-assigned, or "fixed" logic functions will be checked with the information in the device description files. In addition, assignment of reference designators may be controlled with the REFDES-CON text file, generated from a component bill-of materials, or assigned automatically by the system.

Without device description files, the operator must pre-assign logic functions on the schematic. Otherwise, the system will not not have the required information needed to create the net data base.

COMPONENT, FUNCTION AND PIN SWAPPING CAPABILITY

OVERVIEW

The Telesis automatic swapping capability permits the operator to selectively control components, functions, and pin assignment after initial component placement on the board drawing. After the net data base is created from the schematic and device files, and after the board component symbols are placed, the operator has the option to "fix" or "free" components, functions, and pins for automatic swapping on the board.

"Free" components, functions, and pins on the board drawing are those that are available for swapping, while "fixed" components, functions, and pins are unavailable for swapping until the operator uses the "free" commands. After initial placement of board components, those logic functions that are "fixed" were pre-assigned on the original schematic drawing with the device type and pin number labels.

The swapping capability simply improves the board layout for increased auto-routing completion percentage.

The commands that support automatic swapping are located on the |AUTOMATIC PLACEMENT| menus. After using these commands, the operator may back annotate the schematic drawings. Refer to the PLACEMENT section of the manual for information on the commands used to swap components, functions and pins on the board drawing.

10. HOW TO USE THE EXTRACT NETLIST COMMAND

The purpose of the |EXTRACT NETLIST| command is to create (extract) a net data base from a schematic drawing.

PREREQUISITES

- o The schematic must be active drawing. In the case of multi-sheet schematics, the |EXTRACT NETLIST| command must be executed for each sheet.
- o Device description files (optional) must be in the current project, or in the SYSTEM-LIBRARY. Device files are required if the operator wishes to create a net data base allowing component, logic function, and pin swapping on the board drawing. Device files are not required if the operator chooses to assign logic functions and pins on the schematic drawing.

HOW THE EXTRACT NETLIST COMMAND WORKS

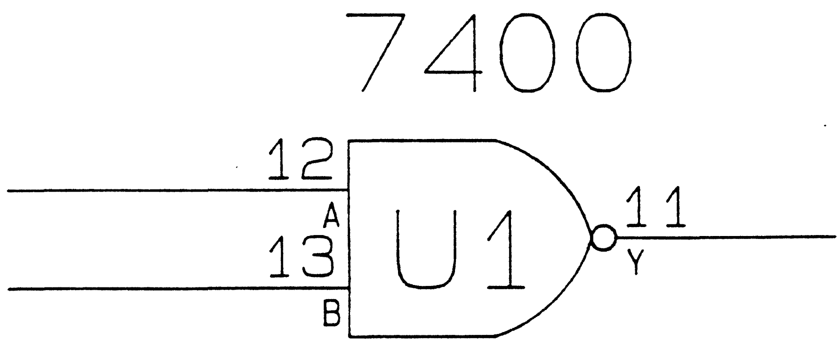
When the operator picks the |EXTRACT NETLIST| command, the system performs the following operations:

1. Reads the schematic drawing, device description files, and the REFDES-CON file, if present, and performs assignments to the net data base file.
2. Creates a text file named EXTRACTION-LOG. This text file lists the system-detected errors on the schematic and in the device description files.
3. Highlights schematic errors on the graphics screen.
4. Creates a net data base using all schematic and device file information. If there is a discrepancy between the information on the schematic and in the device files, the system issues a warning in the EXTRACTION-LOG.

The illustrations below show four possible assignment conditions of a logic symbol on the schematic drawing. A description of how |EXTRACT NETLIST| performs the assignment is provided with each condition.

CONDITION 1. COMPONENT AND FUNCTION PRE-ASSIGNED-

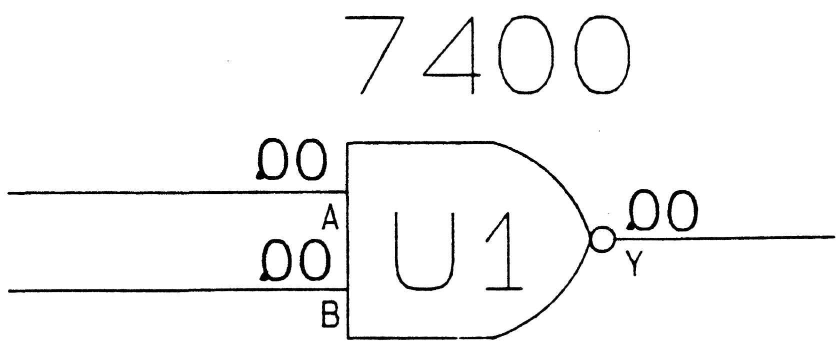
REFERENCE DESIGNATOR AND PIN NUMBERS ANNOTATED



When |EXTRACT NETLIST| is picked, the system assigns the logic function to the reference designator and pin number shown in the schematic.

CONDITION 2. COMPONENT PRE-ASSIGNED WITH REFERENCE DESIGNATOR-

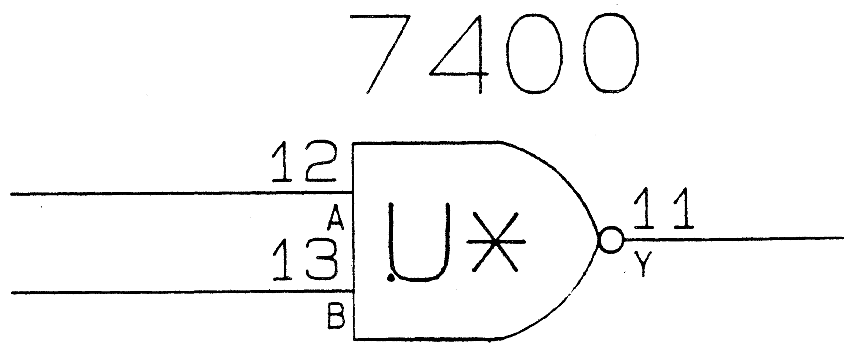
REFERENCE DESIGNATOR ANNOTATED, PIN NUMBERS NOT ANNOTATED



When |EXTRACT NETLIST| is picked, the system assigns the logic function to the first open function slot in the component with the pre-assigned reference designator U1.

CONDITION 3. NOT ASSIGNED TO A COMPONENT, PRE-ASSIGNED BY PIN NUMBERS-

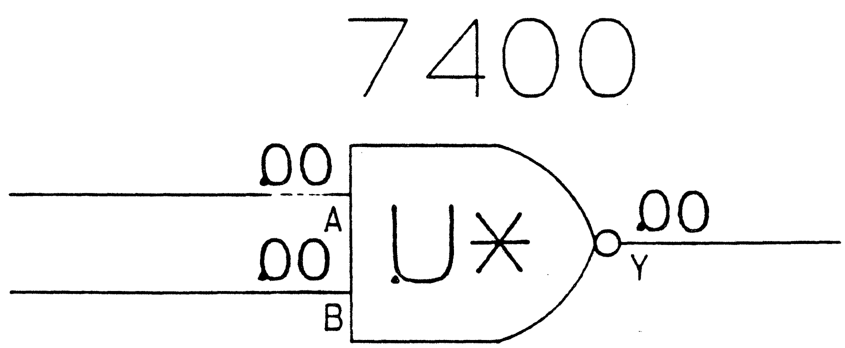
PIN NUMBERS ANNOTATED, NO REFERENCE DESIGNATOR ANNOTATED



When |EXTRACT NETLIST| is picked, the system assigns the function to the correct component already in the net data base. Components may already be there if a REFDES-CON file exists, or if they were created for other similar device functions. If no component is available, or if the required function slots are already filled, the system creates a new component of that device type (i.e. 7400), and assigns the function into the correct slot for that function type.

CONDITION 4. COMPLETELY UNASSIGNED

NO REFERENCE DESIGNATORS AND PIN NUMBERS



When |EXTRACT NETLIST| is picked, the system automatically assigns the function to any unassigned function slot in a component already in the net data base. Components may already be there if a REFDES-CON file exists, or if components were created for other similar device functions. If a component is unavailable, the system creates a new component of that device type, and assigns the function to the first available slot for that function type.

THE EXTRACT NETLIST SEQUENCE

When the operator picks |EXTRACT NETLIST|, the system displays the following message sequence on the function screen message line:

-STARTING SCHEMATIC PAGE DATA EXTRACTION-
-STARTING SORT OF EXTRACTED DATA-
-STARTING COMPARE WITH DEVICE FILES-
-STARTING BUILD OF NET DATA BASE-
-ADDING FUNCTIONS 10%, 20% 30% 40%......<<COMPLETED>>
-ADDING FUNCTIONS WITHOUT PIN NUMBERS-
-ADDING FUNCTIONS WITHOUT REFDES BUT WITH PIN NUMBERS-
-ADDING UNASSIGNED FUNCTIONS-
-ADDING COMPONENTS FROM REFDES-CON FILE-
-BUILD OF NET DATA BASE COMPLETE-
-STARTING NET DATA BASE CLEANUP-
-NET DATA BASE CLEANUP COMPLETED-

THE EXTRACTION-LOG TEXT FILE

The EXTRACTION-LOG is a text file generated by the system during |EXTRACT NETLIST|. This file lists warnings and errors that occurred during the creation of the net data base. The |EXTRACT NETLIST| command should be repeated until the EXTRACTION-LOG is error-free.

To ensure accuracy of the net data base created from a schematic, the operator should perform the following steps:

1. Use the |EXTRACT NETLIST| command.

2. Print out the EXTRACTION-LOG.

3. If warnings or error appear in the EXTRACTION-LOG, edit the schematic and/or device files to correct errors shown in the log.

4. Repeat the above steps until the EXTRACTION-LOG is error-free.

   Each time the |EXTRACT NETLIST| command is repeated, the system updates the existing net data base to reflect any changes made to the schematic and/or device files.

However, using the |EXTRACT NETLIST| command cannot remove or change a component in the net data base once the components are placed on the board with assigned reference designators. Therefore, it is advisable to complete the initial correction process prior to placing the board components on the drawing.

5. Create a NETLIST-REPORT or a COMPONENT-REPORT and check the original engineering sketch against it. This will provide some assurance that the net data base conforms to the original design.

6. The engineering sketch may also be checked against a pen plot of the schematic drawing.

EXAMPLE: EXTRACTION-LOG

```
NET EXTRACTION FOR SCHEMATIC SCH STARTED AT 4-OCT-84  8:35:04

SUMMARY OF DESIGN DRAFTING ERRORS

***TOTAL NUMBER OF DESIGN DRAFTING ERRORS =0  WARNINGS=0

SUMMARY OF DEVICE FILE VERIFICATION ERRORS

Error in parsing device file 7400     rev 1

PINORDER NAND2  A B Y

***Illegal Keyword
ERROR: Unassigned component at (4400, 4400, device 7400
       of function type NAND2 with pin name B and number 0
       has neither a REF DES nor a DEVICE FILE.
       Skipping all functions with same ref des and device
       type.

***TOTAL NUMBER OF DEVICE FILE VERIFICATION ERROR=1 WARNING=0
***GATE ASSIGNMENT begun 4-OCT-84 at 8:35:40***

Gate Assignment done at 8:35:58
*Net Data Base clean-up started at 8:36:09
*Net Data Base clean-up completed at 8:36:15
```

o WARNING: connect line going from (drawing coordinate) to (drawing coordinate) has floating end or ends.

(Line is highlighted in drawing.)

The connect line reported is not attached to a connect point at both ends. Use the |ADD CONNECTION| command to complete the required connections.

o WARNING: connect path has conflicting net names

(Names in conflict are reported in log.)
(Names are highlighted in the drawing.)

There are two signal name labels attached to the same net, but with different text values on them. Delete one of the labels or use the |UPDATE TEXT| command to change one of the signal names to be the same as the other.

o WARNING: connect point has two pin numbers

(Names in conflict are reported in log.)
(Names are highlighted in the drawing.)

There are two pin number labels attached to the same connect point, but with different text values on them. Delete one of the labels or use |UPDATE TEXT| to change one of the pin numbers to be the same as the other. (Normally, a connect point should have only one pin number label attached to it, if device description files are used to create the net data base.)

o WARNING: connect point has two pin names

(Names in conflict are reported on log.)
(Names are highlighted in the drawing.)

There are two pin name labels attached to the same connect point, but with different text values on them. Delete one of the labels or use |UPDATE TEXT| to change one of the pin names to be the same as the other. (Normally, a connect point should have only one pin name label attached to it.)

o WARNING: symbol has two reference designators

(Reference designators in conflict are reported in log, and are highlighted in the drawing.)

There are two reference designators labels attached to the same symbol, but with different text values on them. Delete one of the labels to resolve the conflict. A symbol must not have two reference designator labels.

o WARNING: symbol has two device types

(Device type labels in conflict are reported in log, and are highlighted in the drawing.)

There are two device type labels attached to the same symbol, but with different text values on them. Delete one of the labels to resolve the conflict. A symbol must not have two device type labels.

o WARNING: symbol has two values

(Values in conflict are reported in log, and are highlighted in the drawing.)

There are two value labels attached to the same symbol, but with different values on them. Delete one of the labels to resolve the conflict. A symbol must not have two value labels.

o ERROR: symbol has missing reference designator

(Symbol is highlighted.)
(|EXTRACT NETLIST| was done without a device description file associated with symbol.)

The symbol has a blank or missing reference designator. If blank, (text point with no label), use the UPDATE TEXT command to supply a reference designator. If missing, (no text point, no label) use the ATTCH REF DES command.

o ERROR: missing pin number or pin name on connect point at (drawing coordinate)

The connect point has a blank or missing pin label. (Each connect point must have either a pin name, or a pin number label if the net data base was created without device files.) If blank, use the |UPDATE TEXT| command to supply a pin name or number. If missing (no text point, no label) use the |ATTCH PIN NUMBER| or |ATTCH PIN NAME| command.

o WARNING: missing library data for device type (device type name)

The system found a device type on the schematic, but cannot find a device description file for it. Check the SYSTEM-LIBRARY index. Either the device file is missing, or the device type was misspelled on the schematic.

o WARNING: syntax error in library data for device type (device type name)

There is a formatting error in the device description file for device type name stated in the warning message. Reopen the device description file and correct the error.

CREATING A NET DATA BASE FROM A TEXT-INPUT NETLIST

1. GENERAL INFORMATION
   - PURPOSE
   - ADVANTAGES OF A TEXT-GENERATED NET DATA BASE
   - LIMITATION OF A TEXT-GENERATED NET DATA BASE
   - SUMMARY OF STEPS TO FOLLOW IN CREATING A TEXT-GENERATED NET DATA BASE

2. THE TEXT NETLIST FILE
   - EXAMPLE OF A TEXT-NETLIST FILE

3. THE NET DATA BASE CREATED FROM THE TEXT-INPUT NETLIST

4. REQUIRED CONTENTS AND FORMAT OF THE TEXT NETLIST FILE
   - $PACKAGES
   - $FUNCTIONS
   - $NETS
   - COMMENT LINES
   - END STATEMENT

5. THE TEXT-INPUT NETLIST STRATEGY
   - SUMMARY
   - COMPONENT, FUNCTION, AND PIN SWAPPING OVERVIEW

6. HOW TO USE THE LOAD TXT NETLIST COMMAND TO CREATE THE NET DATA BASE
   - PREREQUISITES
   - HOW THE LOAD TXT NETLIST COMMAND WORKS
   - EXAMPLE: NET-LOAD-LOG FILE
   - REPEATING THE LOAD TXT NETLIST COMMAND
   - ERROR AND WARNING MESSAGES

CREATING A NET DATA BASE FROM A TEXT-INPUT NETLIST

1. GENERAL INFORMATION

PURPOSE

The text-input netlist is a time-saving method of creating a net data base. It may be used as an alternative to drawing a schematic on the system.

This method is appropriate when the operator has the netlist information for a board, and when a schematic drawing is not essential.

The text-input netlist contains the net, pin, reference designator, and device type information to be used in the board design. Optionally, functions may be specified in the file if the operator is creating the netlist from an unannotated engineering sketch.

ADVANTAGES OF A TEXT-GENERATED NET DATA BASE

- o It takes considerably less time to input a text file containing netlist information than it does to draw a schematic on the system.
- o It also takes less time to correct errors in a text-input netlist than in a schematic drawing.

LIMITATION OF A TEXT-GENERATED NET DATA BASE

- o The completed design will not have an up-to-date set of schematic drawings unless the operator manually records any changes on the original engineering sketch.

SUMMARY OF STEPS TO FOLLOW IN CREATING A TEXT-GENERATED NET DATA BASE

STEP 1. Input a text file containing the netlist information for the board design.

STEP 2. Make certain that device files and/or pin files are in the current project or in the SYSTEM-LIBRARY.

Device files are required if the operator wishes to create a net data base that allows automatic swapping of components, logic functions, and pins on the board. A net data base created with device description files allows the operator to use additional system capabilities. For example, logical design rules checking (LDRC) can only be performed if the net data base was created with device files. Pin files are only needed if device files are not used to create the net data base.

2. THE TEXT NETLIST FILE

The text-input netlist is a single text file in three parts:

- o PART 1 is a list of PACKAGES and REFERENCE DESIGNATORS. (OPTIONAL)
- o PART 2 is a list of FUNCTIONS. (OPTIONAL-DEVICE FILES REQUIRED)
- o PART 3 is a list of NETS. (REQUIRED)

NOTE: The file must contain at least PART 1 or PART 2, with the required list of NETS.

EXAMPLE OF TEXT INPUT NETLIST

```
(NETLIST FOR PROJECT TEST, OCTOBER 1984)
$PACKAGES
CON15S/156 ; J
CAPRAD300 ; C1
RES400! RLR05 ; R2
RES400! RLR05 ; R1
CAPRAD300 ; C2
$FUNCTIONS
74LS00 ; AND[1-4]
74L74 ; FLIP[1-2]
$NETS
; J.8  AND2.B
; J.7  AND2.A
; J.4  AND1.B
; J.3  AND1.A
GND  ; J.1  C2.2  C1.2  R2.2
; J.11  R1.2
; R1.1  FLIP2.QN
; FLIP1.Q  AND3.A
; FLIP2.Q  AND3.B
; R2.1  FLIP1.QN
+5V  ; J.12  J.10  C2.1  C1.1  FLIP1.PRE  FLIP2.PRE  AND3.Y
; AND1.Y  FLIP1.D
; J.6  FLIP2.CLK  FLIP1.CLK
; J.5  FLIP2.CLR  FLIP1.CLR
$END
```

3. THE NET DATA BASE CREATED FROM THE TEXT-INPUT NETLIST AND DEVICE DESCRIPTION FILES.

After the net data base is created from the text-input netlist, the operator may proceed to place components on the board. If the net data base is created with device description files, the automatic swapping capability may be used to improve the placement of ratsnest connect-lines for increased completion percentage during automatic routing.

Like the schematic drawing, the operator may selectively pre-assign logic functions, using the text netlist file. Pre-assigned logic functions are associated with those devices defined by the $PACKAGES section (reference designator), with the $NETS section containing the reference designator and pin number information for each device. When the net data base is created with the |LOAD TXT NETLIST| command, a pre-assigned logic function is "fixed" on the board and unavailable for automatic swapping until "freed" by the operator.

In addition, the operator may optionally allow the system to perform reference designator and pin number assignments to the net data base during |LOAD TXT NETLIST| with the optional $FUNCTIONS section of the netlist file. The $FUNCTIONS section of the file may be used when inputting a text netlist from an original, unannotated engineering sketch. The information defined by the $FUNCTIONS section (device type; function designator) and the related function designators and pin names associated with a single logic function specified under the $NETS section will reflect a "free" logic function. The system uses the device, function and pin information in the device files to make assignments to the net data base.

During |LOAD TXT NETLIST|, the system automatically assigns a reference designator to each device (or with an operator-input REFDES-CON text file) specified by $FUNCTIONS. The system also assigns pin numbers to each available (unassigned) function slot using information from the device file. Logic functions created in the net data base associated with a device listed with $FUNCTIONS are "free" for automatic swapping on the placed board.

NOTE: Devices specified under $PACKAGES cannot be re-stated under $FUNCTIONS, and the reverse.

If the operator chooses to create the net data base using pin description files rather than device files, the operator must use the $PACKAGES section of the netlist. Without device description files, the function and pin swapping capability cannot be used on the placed board drawing simply because the net data base does not have required device, pin and function information. The reference designator and pin assignments specified in the text netlist under $NETS will be "fixed" to components created in the net data base.

The following section describes the formatting requirements of the text netlist file used to create the net data base.

$PACKAGES

PURPOSE: The $PACKAGES section of the netlist file allows pre-assignment of reference designators to components during creation of the net data base. During the |LOAD TXT NETLIST| command, the system first reads the $PACKAGES section, then the $NETS section of the netlist file.

With device description files present in the current project, or SYSTEM-LIBRARY, the system uses the device file information to build a net data base which allows swapping of functions and pins on the placed board drawing.

If a device contains swappable functions and pins, the operator may selectively "free" pre-assigned functions for automatic swapping on the board.

FORMAT: $PACKAGES

- o The $PACKAGES keyword is placed on the first line of the text file. A list of packages follows the keyword.

LIST OF PACKAGES

- o The list includes packages to be used in the board design, with a reference designator for each instance of a package. Packages are listed by component symbol name and device type. The operator may omit the component symbol name, however, the device type must be included if device description files are to be used to create the net data base. The component symbol name must be included if the system uses pin description files to create the net data base.

FORMAT: COMPONENT SYMBOL NAME ! DEVICE TYPE; REF DES

```
EXAMPLE:    $PACKAGES
            RES400 ! RLR05 ; R2
            CON15S/156 ; J
```

In the above example, the system will use the -PIN text file to define the CON15S/156 component symbol when |LOAD TXT NETLIST| creates the net data base. The device file RLR05 will define the component symbol RES400.

INPUT SEQUENCE:

COMPONENT SYMBOL NAME:

The component symbol name must be precisely the same as the name of component symbol to be placed on the board drawing.

A symbol name may contain up to 14 characters with no blank spaces allowed. (Allowed characters are: A-Z, 0-9, - + = % * ? / . ")

Each component symbol name must be placed at the beginning of a new line.

Repeat the symbol name on a new line for each different device type.

```
EXAMPLE:    RES400 ! RLR05; R2
            RES400 ! RLR06; R1
```

DEVICE TYPE:

Place the device type between the symbol name and the list of reference designators. It must be preceded by an exclamation point.

The device type must be the same name of the device file to be used with the package.

A device type may contain up to 18 alphanumeric characters with no blank spaces. (Allowed characters are: A-Z 0-9 - + = % * ? / . ")

REFERENCE DESIGNATORS

A semicolon (;) must be placed before the list of reference designators.

A reference designator may contain up to 8 characters with no blank spaces. (Allowed characters are A-Z 0-9 - + = % ? / ")

Reference designators must be separated from each other with a blank space, or with a comma (,).

To continue the list of reference designators on a new line, place a comma at the end of the line and continue on the line below:

```
EXAMPLE:  TO92 ; Q1 Q2 Q3 Q4 Q5 Q6 Q9 Q10 Q11 Q14 Q15,
                 Q16 Q17 Q25 Q26 Q27
```

To input a series of reference designators, brackets may be used:

```
EXAMPLE:   Z6 Z[8-10] Z12 Z15
```

- o The repeating character(s) precedes the first bracket.
- o A blank space before the first bracket is not allowed.
- o The characters in the series must be numeric.
- o The first bracket and the closing bracket must be on same line. (Blank spaces are allowed within the brackets.
- o Use a hyphen (-) between the first and last numbers in the series.
- o Leading zeros are allowed. (However, the system will ignore any leading zeros in the last number that cause the last number to have more digits than the first number. For example, if [0008 - 0100] is input, all leading zeros are used by the system; if [008 - 0100] is input, the system ignores the leading zero in the last number.

TO OMIT A COMPONENT SYMBOL NAME:

(Device files used to create the net data base.)

Begin the line with an exclamation point (!) followed by the device type.

```
EXAMPLE:  ! 7400 ; U1  U4  U6
```

! NEW ! VALUE ; 2Y —

TO OMIT THE DEVICE TYPE:

(Pin files used to create the net data base.)

Omit both the exclamation point and the device type name.

EXAMPLE: DIP14 ; U1 U4 U6

$FUNCTIONS (OPTIONAL - Must follow the $PACKAGES section, if used)
- Device files are required.
- Devices specified under $PACKAGES cannot be re-specified under $FUNCTIONS, and the reverse.

PURPOSE: The optional $FUNCTIONS section of the text netlist file is useful if the operator is creating a netlist from an unannotated engineering sketch. For example, if the sketch shows eight NAND2 gates without pin number and reference designator annotation, the operator may input entries under $FUNCTIONS. An entry must contain a device type, an operator-defined FUNCTION DESIGNATOR, as well as the number of functions to be created for the device specified. Based on the information in the device file, the system creates the required number of components needed to accommodate the number of functions.

This section of the netlist file can only be used if the net data base is to be created with device description files in the current project, or SYSTEM-LIBRARY.

Entries made to the $FUNCTIONS section are unassigned. That is, the system assigns available reference designators automatically to the devices during |LOAD TXT NETLIST|, or performs the assignment with an operator-input REFDES-CON file in the current project.

During the |LOAD TXT NETLIST| command, the system reads the entries within $FUNCTIONS, then reads the $NETS section of the file. The $NETS section must contain FUNCTION DESIGNATORs, and PIN NAMEs rather than pin numbers to specify nets. The system uses the information in the device files to build a net data base which allows swapping of functions and pins on the placed board drawing. Functions and pins are "free" on the board for automatic swapping.

FORMAT: $FUNCTIONS

- If used, the $FUNCTIONS keyword must follow the $PACKAGES section of the file. A list of device types follows the keyword. Devices specified under $PACKAGES cannot be re-stated in the $FUNCTIONS section of the netlist.

LIST OF DEVICE TYPES; FUNCTION DESIGNATORS

- The list includes each device type with an operator-defined FUNCTION DESIGNATOR, and the number of logic functions to be created in the net data base for the specified device type. Based on the information in the device description file, the system creates the proper number of components for the number of logic functions specified.

  Component symbol names are not required since device description files are required to support this optional section of the netlist file.

  FORMAT: DEVICE TYPE ; FUNCTION DESIGNATOR [NUMBER OF FUNCTIONS]

```
EXAMPLE:   $FUNCTIONS
           74LS00 ; AND[1-4]
           74L74 ; FLIP[1-2]
```

  In the above example, the device 74LS00 is stated under $FUNCTIONS, with the operator-defined function designator AND, requiring four logic functions to be created during |LOAD TXT NETLIST|. The system creates a single component in the net data base, since the device 74LS00 only contains four logic functions.

  The example below illustrates the typical entries to the $NETS section of the file, relating to the entries contained under $FUNCTIONS.

```
$NETS
; FLIP1.Q  AND2.A
; AND1.Y  FLIP2.D
```

  The line ;FLIP1.Q AND2.A identifies available functions and pin names associated with that net. FLIP1.Q identifies an available function on the device 74L74, with pin name Q connected to pin name A of the function, AND2, of the device 74LS00. The pin number assignments are made from the information in the device files when the |LOAD TXT NETLIST| command is used to create the net data base. However, because function designators and pin names are specified, logic functions are free for swapping with other functions of the same device type on the placed board drawing. For example, the AND2 function assignment in the net data base is swappable with other available functions on the same device (AND1, AND3, AND4).

NOTE ON THE NUMBER OF FUNCTIONS

Under the $FUNCTIONS section of the netlist file, the operator may specify a device type, the function designator, and the total number of functions to be used, exceeding the number of functions available on one device. For example, the 74LS00 device contains four functions. The operator, however, may specify any number of functions. When the net data base is created, the system creates the appropriate number of components needed for the number of functions specified.

```
EXAMPLE:  $FUNCTIONS
          74LS00 ; AND[1-8]
```

In the above example, eight functions will be created in the net data base for the device 74LS00. Since only four functions can fit on one physical device, the system creates the net data base with two 74LS00 devices with four available functions on each device.

However, if eight functions are specified for a device under $FUNCTIONS, with only four functions reflected under the $NETS section, the system will still create two 74LS00 components in the net data base. Five functions (AND[1-5]) reflected under $NETS will create two components, since one 74LS00 contains only four logic functions.

If a device contains only one function, the brackets must be omitted. In the example below the operator-defined function designator, "NAND", contains only one function. Simple omit the brackets and place the number "1" immediately after the function designator.

```
EXAMPLE:  $FUNCTIONS
          7400 ; NAND1
```

NOTE ON FUNCTION DESIGNATORS

If a single device contains different logic functions, the function type specified in the device description file must be stated under $FUNCTIONS after the device name, preceding the operator-defined function designator. The system requires the additional function type entry to make the distinction between different functions on a single device. For example, the device 4000A contains two NOR3 gates and the inverter (INV) logic function. The 4000A device file is shown below:

```
(DEVICE DESCRIPTION FILE: 4000A)
PACKAGE DIP14
CLASS IC
PINCOUNT 14
PINORDER NOR3 A B C Y
PINUSE NOR3 IN IN IN OUT
PINSWAP NOR3 A B C
FUNCTION G1 NOR3 3 4 5 6
FUNCTION G3 NOR3 11 12 13 10
PINORDER INV A Y
PINUSE INV IN OUT
FUNCTION G2 INV 8 9
POWER +5V; 14
GROUND GND; 7
NC 1 2
END
```

The format under $FUNCTIONS is shown below for single device types containing different functions.

DEVICE NAME ! FUNCTION TYPE ; FUNCTION DESIGNATOR [NUMBER OF FUNCTIONS]

```
EXAMPLE:  $FUNCTIONS
          4000A! NOR3; XYZ[1-2]
          4000A! INV ; ABC1
```

- Line entries must be repeated for the different function types.
- In the above example, the NOR3 function type is specified by the operator-defined function designator XYZ, with two available functions. The function type INV is specified by the function designator ABC, with one available function.

INPUT SEQUENCE:

DEVICE TYPE NAME

The device type name must be the exact same name as its device description file

A device type may contain up to 18 alphanumeric characters with no blank spaces. (Allowed characters are: A-Z 0-9 - + = % * ? / . ")

The component symbol name does not need to be included since $FUNCTIONS section of the netlist file requires device description files to create the net data base.

Simply input the device name at the beginning of each line, following the $FUNCTIONS statement.

```
EXAMPLE:  $FUNCTIONS
          74LS00 ;
```

FUNCTION DESIGNATORS and [NUMBER OF FUNCTIONS]

A semicolon (;) must follow the device name, preceding the function designator and the number of functions.

An operator-defined function designator may contain up to eight characters with no blank spaces. (Allowed characters are A-Z 0-9 - + = % * ? / ")

Immediately following the function designator, input the number of available functions to be created in the net data base, using brackets to specify a range. Do not input a blank space between the function designator and the number of functions. Brackets may be omitted if the device contains only one function.

```
EXAMPLE:  $FUNCTIONS
          74LS00 ; AND[1-4]
          74L74  ;FLIP1
```

Refer to NOTE ON FUNCTION DESIGNATORS on the previous page for information on single devices containing different function types and an example of a typical entry under $FUNCTIONS.

PURPOSE: The $NETS section of the netlist file specifies the net information for the entries made under the $PACKAGES and $FUNCTIONS sections of the file.

When the operator picks the |LOAD TXT NETLIST| command, the system reads the contents of the $PACKAGES section and the $FUNCTIONS section of the netlist. The system then scans the $NETS section for the net information needed to create the net data base.

The net information specified must be entered in two different formats if the operator used the optional $FUNCTIONS section of the netlist. Nets specified for entries contained under $PACKAGES are entered by pre-assigned reference designators and pin numbers. Nets specified for entries under the $FUNCTIONS section are entered by function designator, function number, and pin name.

NOTE: The function number provides a unique name to to each function designator associated with a single logic function on a device. For example, AND1.A, AND1.B, AND1.Y are functions associated with a single function on the 74LS00, specified by the function designator AND listed under $FUNCTIONS.

The following example illustrates typical net entries. The first entry shown under $NETS uses reference designators and pin numbers, relating to $PACKAGES. The second net entry uses a function designator, function number, and pin name, relating to $FUNCTIONS. The third net entry under $NETS uses information pertaining to both the $PACKAGES and $FUNCTIONS sections.

```
EXAMPLE:  $PACKAGES
          RES400 ! RLR05 ; R2
          CAPRAD 300; C1
          $FUNCTIONS
          74LS00 ; AND[1-4]
          74L74 ; FLIP[1-2]
          $NETS
          ; R2.1 C1.2
          ; AND1.A FLIP2.QN
          ; R1.1 FLIP1.QN
```

NOTE: The function designator and function number, FLIP1 for example, under $NETS specifies an available function slot on the device. FLIP2 also specifies an available function on the device. The function number does not imply a selected function slot.

FORMAT: $NETS

- o The keyword is placed on the line following the list of packages, and/or list of functions. The keyword $NETS precedes the list of nets.

LIST OF NETS

- o This includes all nets that will be used in the board design, and provides a list of pins and/or function designator and pin names associated with each net.

For nets relating to devices and reference designators specified under $PACKAGES:

```
FORMAT:  ;RefDes.PinNumber RefDes.PinNumber RefDes.PinNumber

EXAMPLE:  ;R2.1 C2.1 U4.3
```

For nets relating to devices, function designators and number of functions specified under $FUNCTIONS:

```
FORMAT: ;FunctionDesignatorFunctionNumber.PinName,
         FunctionDesignatorFunctionNumber.PinName

EXAMPLE:  ;AND1.A  FLIP2.QN
```

For nets relating to entries pertaining to both $PACKAGES and $FUNCTIONS sections.

```
FORMAT: ;RefDes.PinNumber FunctionDesignatorFunctionNum.PinName

EXAMPLE: ;R1.1  FLIP1.QN
```

INPUT SEQUENCE:

- o Each net begins on a new line. A semicolon must precede the net information.

- o For nets specified by reference designator:

  -Show the reference designator and pin number for each pin. The format is --REFDES.PIN#--with no blank spaces allowed.

  -Reference designators must correspond to the ones shown in the list of packages.

  -A pin number may be any integer from 1 to 32767.

  -Put at least one blank space between pins in the list.

  -To continue the list of pins on a new line, place a comma at the end of the line and continue on the line below.

  -As in the list of packages, brackets may be used to input a series of reference designators, a series of pin numbers, or both:

```
Z[1-3].7 is equivalent to Z1.7  Z2.7  Z3.7
Z3.[2-5] is equivalent to Z3.2  Z3.3  Z3.4 Z3.5
RP[1-3].[1-2] is equivalent to RP1.1  RP1.2  RP2.1
          RP2.1  RP2.2  RP3.1  RP3.2
```

  There must be no blank spaces between the (.) and a bracket preceding it, or following it. Otherwise the the input requirements for a bracketed series are the same as in the $PACKAGES list.

```
EXAMPLE:  $NETS
          ;R2.2  C1.1
```

o For nets specified by function designator

-Show the function designator, the function number and the pin name. The format is --FunctionDesFunctionNum.PinName-- with no blank spaces allowed.

-Function designators must correspond to the ones shown in the list of functions.

-The function number (providing the unique name associated with a single logic function) immediately following the function designator must be within the range specified in brackets under the $FUNCTIONS section. Blank spaces are not allowed between the function designator and the function number.

-The pin name must be separated from the function number by by a (.). Blank spaces are not allowed.

-Put at least one blank space between pin name and the next entry on the line.

```
EXAMPLE:  $NETS
          ;FLIP1.Q  AND3.A
```

-To continue the net information on a new line, place a comma at the end of the line and continue on the line below.

NET NAME (OPTIONAL)

o The operator may optionally give each net a name.

FORMAT: NETNAME; RefDes.PinNum RefDes.PinNum

```
EXAMPLE:

$NETS
SERIAL CLOCK; J7.22  Z10.11
```

-The net name must be the same as the net name the operator uses on the board drawing.

-A net name may contain up to 18 alphanumeric characters with blank spaces allowed. (Allowed characters are; A-Z 0-9 - + = % * ? / . : ")

-Place the net name at the beginning of the line before the net information.

COMMENTS (OPTIONAL)

- The operator may include information comments in the text netlist file.

  FORMAT: (COMMENT LINE)

  EXAMPLE:

  (***NET DEFINITIONS ARE BELOW***)

  -Place comments in parentheses for informational use only. The system ingores text within parentheses.

  -A comment may not extend beyond the length of one line.

END STATEMENT (OPTIONAL)

- The END statement must be the last line of the text file if the operator chooses to use it.

FORMAT: $END

5. THE TEXT-INPUT NETLIST STRATEGY

SUMMARY

When inputting the text netlist, the operator may simply make entries to $PACKAGES, and $NETS. If device description files are used to create the net data base during |LOAD TXT NETLIST|, the system takes the information from the device files to create a net data base which allows swapping of logic functions and pins on the board drawing. The system uses the reference designator specified in the $PACKAGES section, and the reference designator and pin number entries contained in the $NETS section to make the initial assignments. That is, the reference designators and pin numbers specified under $NETS are associated with the reference designators specified under $PACKAGES. The resulting "fixed" logic functions in the net data base may be "freed" for automatic swapping on the board components if a device contains swappable functions. However, swappable pins are always "free" to swap until "fixed" by the operator.

Optionally, the operator may input the text netlist with a $FUNCTIONS section, following the entries to $PACKAGES (if used). The operator may use this section if inputting the netlist from an unannotated engineering sketch. Entries made under $FUNCTIONS are unassigned. That is, the system automatically assigns available reference designators to components created during |LOAD TXT NETLIST|. The operator may create an REFDES-CON text file, generated from a component bill-of-materials, prior to using the |LOAD TXT NETLIST| command, if specific operator-defined reference designators are to be used in the creation of the net data base.

The unique function designators and pin names under $NETS are related to the devices and function designators listed under $FUNCTIONS. The system makes the assignments to the net data base using the information in the device description files, creating the essential number of components for the required number of functions. The resulting net data base will contain "free" logic functions that are available for automatic swapping on board components.

If pin files are used to create the net data base during |LOAD TXT NETLIST|, the system requires that all assignment information be present under $PACKAGES and $NETS. In addition, the swapping capability cannot be used on the board drawing.

COMPONENT, LOGIC FUNCTION, AND PIN SWAPPING OVERVIEW

The Telesis automatic swapping capability permits the operator to selectively swap components, logic functions, and pin assignments on the board after initial component placement. After the net data base is created from the text netlist and device description files, and after the board component symbols are placed, the operator has the option to "fix" or "free" components, functions, and pins for swapping on the board.

The swapping capability simply improves the board layout for increased completion percentage during automatic routing.

The commands that support automatic swapping are located on the |AUTOMATIC PLACEMENT| menus. After using the swapping commands, the operator may update the text netlist to reflect the assignments on the board drawing, using the |CREATE TEXT NETLIST| command.

Refer to the PLACEMENT section of the manual for the commands used to swap components, functions, and pins on the board.

6. HOW TO USE THE |LOAD TXT NETLIST| COMMAND TO CREATE THE NET DATA BASE

The COMMANDS section of the manual defines the exact input sequence for the |LOAD TXT NETLIST| command. The information below explains the use of the command in broader context.

PREREQUISITES

- o A device file for every device type named in the text netlist must be in the current project, or in the SYSTEM-LIBRARY. The device file prerequisite may be omitted if pin files are used to create the net data base. However, a net data base created without device files cannot, for example, be used with logical design rules checking, interactive or auto-placement, or function and pin swapping.

- o A pin file for each symbol named in the text netlist must be in the current project, or in the SYSTEM-LIBRARY. Pin description files may be omitted if device description files are used to create the net data base.

- o The text input netlist must be in the current project file.

HOW THE |LOAD TXT NETLIST| COMMAND WORKS

When the |LOAD TXT NETLIST| command is picked, the operator must input the name of the text netlist file. The system then performs the following operations:

1. Creates a text file named NET-LOAD-LOG listing system-detected errors in the text netlist. The text netlist is repeated line--by-line in the NET-LOAD-LOG. Each error is listed directly below the line involved in the error.

2. Creates a net data base if there are no major errors.

3. Places the following messages on the function screen when they are appropriate.

   << STARTING LOAD TEXT NETLIST.>>
   << CHECKING FOR KEYWORDS.>>
   << KEYWORDS NOT FOUND. NETLOAD STOPPED.>>

   If the keywords are found, the system will display the following message sequence on the function screen message line.

```
<< STARTING LOAD TEXT NETLIST. >>
<< CHECKING FOR KEYWORDS.>>
<< REQUIRED KEYWORDS FOUND. CONTINUING.>>
<< PROCESSING LINE 10 OUT OF 27 LINES TOTAL.>>
<< PROCESSING LINE 20 OUT OF 27 LINES TOTAL.>>
<< STARTING BUILD OF NET DATA BASE.>>
<< ADDING FUNCTIONS>> << 100% COMPLETED>>
<< CREATING COMPONENT__ CREATING COMPONENT__....>>
<< ADDING FUNCITONS WITHOUT PIN NUMBERS.>>
<< ADDING FUNCTIONS WITHOUT REFDES BUT WITH PIN NUMBERS.>>
<< ADDING UNASSIGNED FUNCTIONS.>>
<< CREATING COMPONENT__ CREATING COMPONENT__...>>
<< ADDING COMPONENTS FROM REFDES-CON FILE.>>
<< BUILD OF NETDATA BASE COMPLETE.>>
```

After the message <<BUILD OF NETDATA BASE COMPLETE.>> appears on the function screen message line, the system displays the NET-LOAD-LOG.

EXAMPLE: NET-LOAD-LOG FILE

```
 NET-LOAD-LOG

        1  $PACKAGES
        2  CON15S/156
MESSAGE:     CON15S/156  -PIN (pin file) defines this package.
------------------------------------------------------------------
        3  CAPRAD300 ; C1
MESSAGE:     CAPRAD300   -PIN (pin file) defines this package.
------------------------------------------------------------------
        4  RES400 !  RLR05 ; R2
MESSAGE:     RLR05          (device file) defines this package.
------------------------------------------------------------------
        5  RES400 !  RLR05 ; R1
MESSAGE:     RLR05          (device file) defines this package.
------------------------------------------------------------------
        6  CAPRAD300  ;  C2
MESSAGE:     CAPRAD300   -PIN (pin file) defines this package.
------------------------------------------------------------------
        7  $FUNCTIONS
        8  74LS00 ; AND[1-4]
        9  74L74  ; FLIP[1-2]
       10  $NETS
       11  ; J.8  AND2.B
       12  ; J.7  AND2.A
       13  ; J.4  AND1.B
       14  ; J.3  AND1.A
       15  GND  ;  J.1  C2.2  C1.2  R2.2
       16  ; J.11  R1.2
```

NET-LOAD-LOG (continued)

```
      17  ; R1.1   FLIP2.QN
      18  ; FLIP1.Q  AND3.A
      19  ; FLIP2.Q  AND3.B
      20  ; R2.1   FLIP1.QN
      21  ; +5V; J.12 J.10 C2.1 C1.1 FLIP1.PRE FLIP2.PRE AND3.Y
      22  ; AND1.Y FLIP1.D
      23  ; J.6   FLIP2.CLK   FLIP1.CLK
      24  ; J.5   FLIP2.CLR   FLIP1.CLR
      25  $END
MESSAGE :  $END keyword found.
ACTION :  Normal termination at this line.
-----------------------------------------------------------------
***GATE ASSIGNMENT begun 03-OCT-84 at 09:21:15 ***

ADDING ASSIGNED FUNCTIONS
***NUMBER OF ERRORS = 0
ADDING UNASSIGNED FUNCTIONS
***NUMBER OF ERRORS = 0
ADDING PIN FROM DEVICE FILES
***NUMBER OF ERRORS***
Gate Assignment done at 09:24:00
```

REPEATING THE |LOAD TXT NETLIST| COMMAND UNTIL THE NET-LOAD-LOG IS ERROR-FREE

To ensure that an accurate net data base has been created, perform the following steps:

STEP 1. Use the |LOAD TXT NETLIST| command.

STEP 2. Print out the NET-LOAD-LOG text file.

STEP 3. Open the text netlist file and edit the file to correct the errors shown in the NET-LOAD-LOG.

STEP 4. Repeat STEPS 1-3 until the NET-LOAD-LOG is error-free.

Each time the |LOAD TXT NETLIST| command is repeated, the system clears any existing information in the net data base, and replaces it with the information in the text netlist.

STEP 5. If the original engineering sketch is available, generate a NETLIST-REPORT (|CREATE NETLIST REPORT|), or COMPONENT-REPORT (|CREATE COMPONENT REPORT|), and check it against the original drawing.

TYPES OF ERROR AND WARNING MESSAGES LISTED IN THE NET-LOAD-LOG

Error and warning messages that may appear in the NET-LOAD-LOG are listed below. A brief explanation is given here for messages that are not self-explanatory.

- "Error: Non-comment line(s) before $PACKAGES keyword.
  Action: Line(s) ignored."

- "Warning: Invalid text in keyword line.
  Action: Invalid text ignored."

  The system ignores any superfluous character appearing after a keyword in a keyword line.

- "Error: Keyword expected in line with $.
  Action: Line ignored."

- "Error: Input line too long. Limit of 132 characters."
  Action: Line ignored."

- "Error: Keywords ($PACKAGES, $END , or $NETS) are missing or out of order.
  Action: Load Text Netlist Terminated."

  The system will not create a net data base because it cannot find the required keywords.

- "Error: Invalid character(s) in package name '____'.
  Action: Processing for this package name terminated."

  The system ignores the information listed for this package.

- "Error: Package name '____(first 14 char)' too long.
  Action: Processing for this package name terminated."

  The system ignores the information listed for this package.

- "Error : Invalid character(s) in device name '___'..
  Action Processing for this device name terminated."

  The system ignores the information listed for this device.

- "Error: Device name '___' too long.
  Action: Processing for this device name terminated."

  The system ignores the information listed for this device.

- "Error: Cannot find device name and/or package name.Expecting a semicolon in input.
  Action: Line ignored."

- "Error: Semicolon (;) found in continuation line. Not allowed in continuation.
  Action: Line ignored.".

- "Error: Device file package name '___' and netlist package name ' ' do not match.
  Action: Processing for this device name terminated."

  The system ignores the information listed for this device.

- "Error: Errors found in device file '___'.
  Action: Processing for this device terminated."

  The system ignores the information listed for this device.

- "Error: Errors found in - PIN file '___'.
  Action: Processing for this device terminated."

  The system ignores the information listed for this device.

- "Error: Existing reference designator '___' already in NET-DATA-BASE, but has different package or device type than that specified by input text.
  Action: Existing reference designator not changed. No new entries."

- "Error: Reference designator for this pin '___' not in NET-DATA-BASE.
  Action: Pin not loaded into NET-DATA-BASE."

- "Error: Net name '___' too long.
  Action: Net ignored."

- "Error: Invalid character "___" in net name '___'.
  Action: Net ignored."

- "Error: Pin number '___' too large.
  Action: Pin ignored."

- "Error: Designated refdes. pin '___' not found in component.
  Action: Refdes. pin ignored."

- "Error: Pin number '___' has non-numeric character.
  Action: Pin ignored "

- "Error: Refdes. pin '___' already assigned to net #, name '___
  Action: Refdes. pin ignored in this net."

- "Error: Invalid or missing pin number for refdes. '___'.
  Action: Refdes. ignored."

- "Error: Unbalanced parenthesis found in line.
  Action: Line ignored."

- "Action Continuation line ignored because of previous error."

  The system ignores a continuation line whenever the previous line was ignored because of error.

- "Error: [] Range specification error. Unbalanced brackets.
  Action: Refdes ignored."

- "Error: [] Range specification error. Starting value less than ending value.
  Action: Refdes.ignored.

- "Error: [] Range specification error. Invalid character in range field. (Only numbers and - allowed.).
  Action: Refdes ignored."

- "Error: [] Range specification error. Number too large or too small or missing. Limit 1 to 31999.
  Action: Ref des ignored."

- "Error: [] Range specification error. Only one period (.) allowed.
  Action: Refdes. pin ignored."

- "Error Reference designator '_____' too long.
  Action: Reference designator ignored."

- "Error: Invalid characters in refdes.
  Action: Reference designator ignored."

- "Error: [] Range specification error. Invalid number of leading characters.
  Action: Reference designator ignored."

- "Error: [] Range specification error. Start or end refdes too long.
  Action: Reference designator ignored.

- "Error: Unbalanced brackets in input line.
  Action: Line ignored."

- "Error: Cannot find semicolon (;) while looking for the next net.
  Action: Processing for this net terminated."

- "Message: Netload terminated by operator CANCEL."

- "Message: $END keyword found.
  Action: Normal termination at this line."

- "Error: Cannot find device file '___'.
  Action: Processing for this device name terminated."

  The system ignores the information listed for this device.

- "Error: Cannot open device file '___'.
  Action: Processing for this device name terminated.".

  The system ignores the information listed for this device.

- "Error: Cannot find - PIN file '___'.
  Action: Processing for this package name terminated."

  The system ignores the information listed for this package.

- "Error: Cannot open - PIN file '___'.
  Action: Processing for this package name terminated."

  The system ignores the information listed for this package.

USING A NETLIST GENERATED BY THE EDA-1000

1. THE TELESIS NETLIST FORMAT GENERATED BY THE EDA-1000

2. SAMPLE EDA-1000 NETLIST

1. THE TELESIS NETLIST FORMAT GENERATED BY THE EDA-1000

The netlist generated by the EDA-1000 follows the conventions already described in this section.

However, an additional keyword called "$WASIS" is included in the netlist generated by the EDA-1000. When the netlist generated by the EDA-1000 is transferred to the EDA-3000, this $WASIS keyword informs the system creating the net data base that a "back annotatable" netlist is being compiled during the EDA-3000 LOAD TEXT NETLIST command. Back annotation is the process of updating the schematic drawing with the pin numbers and reference designators assigned during the creation of a PC board drawing. The keyword $WASIS is always the first line entry in an EDA-1000 generated netlist file.

Following the $WASIS keyword is the $FUNCTIONS section. Generally, each record under $FUNCTIONS has the following format:

Device Type! Function Type! Value; Function Des1. . . .Function Des2

When creating the schematic drawing, the user may assign device types, function types, and values as described earlier. The function designators, however, are created by the EDA-1000 and have the following format:

Reference number — Symbol Reference Number

The Reference number is extracted from the netlist generated by the Netlist Processor Program (NETXTR). If the operator created a multi-sheet schematic (as opposed to a hierarchical design), the reference number will always be the form 1-N, where N is the sheet number. The reference number format for a hierarchical design is of greater complexity. Refer to the section on the Netlist Processor for detailed information.

The Symbol Reference Number is the number generated by the Design Capture Editor, and is displayed in the upper left corner of the symbol cell.

Function Designators uniquely identify a symbol on a sheet of a set of schematic drawings.

EXAMPLE:
```
$WASIS
$FUNCTIONS
74LS02;1-1-51 1-1-101 1-1-122 1-1-123
CK05!!.-1UF;1-1-116 1-1-121
```

In the example, the first line under $FUNCTIONS contains the following information:

```
74LS02;1-1-51 1-1-101 1-1-122 1-1-123

Device type:    74LS02
Function type:  74LS02 (by default, no function type
                       for this device on the schematic)
Value:          no value assigned to symbol
Function Des:   1-1-51
                1-1-101
                1-1-116
                1-1-121
```

The symbol reference numbers are 51, 101, 116 and 121. They all appeared on sheet 1.

The second line under $FUNCTIONS contains similar information, except that a value is present (notice the two exlamation points following the device type):

```
CK05!!.01UF;1-1-116 1-1-121

Device type:    CK05
Function type:  CK05  (by default, no function type for
                       this device on the schematic)
Value:          .01UF (if the value was not present on the
                       schematic symbol, the exclamation
                       points would not be shown in the
                       entry)
Function Des    1-1-116
                1-1-121
```

The next record in the EDA-1000 generated netlist file is the $NETS section. This section contains the net information of the schematic sheet(s) in a drawing set. Generally each record will have the following format:

Signal Name; Component1....Component2....ComponentN

If the operator named the net, it will be included as a signal name followed by a semi-colon.

The format of each component may be either of the following:

;Reference Designator:Function Designator.Pin Number

or

;Reference Designator:Function Designator.Pin Name

If the reference designator is "open-series" (with an asterisk) or a pin name (rather than a pin number), the component reference designator and pin numbers will be assigned during the LOAD TEXT NETLIST command when the net data base is created on the EDA-3000 software.

EXAMPLE: Entries under the $NETS section of the netlist

```
$NETS
;U*:1-1-43.Y
+5V; C1:1-1-1.1
```

In the first line entry under $NETS, a net name (or a signal name) is not specified. The "open-series" reference designator, U*, is followed by a colon, then by the function designator.pin name. During the LOAD TEXT NETLIST command, the EDA-3000 software will assign an available reference designator to the component created by U*. The system will aslo assign an available pin number to the pin name, Y, to an available function slot on the component. The system will get this assignment information from the device description file for the component.

In the second line entry under $NETS, a net name (+5V;) is specified followed by the component reference designator, function designator, and pin number. This is a pre-assigned component; the reference designator (C1) and pin number (1) are assigned on the schematic drawing.

2. Sample EDA-1000 Netlist

```
$WASIS
$FUNCTIONS
74LS02;1-1-51 1-1-101 1-1-122 1-1-123
74LS32;1-1-17 1-1-35 1-1-43 1-1-58
74LS193;1-1-9 1-1-10 1-1-11
2716;1-1-36 1-1-112
3904;1-1-64 1-1-65
6116;1-1-37
CK05!!.01UF;1-1-116 1-1-121
CK05!!.01uF;1-1-1 1-1-79 1-1-80 1-1-100
CON50;1-1-104 1-1-105 1-1-106 1-1-107 1-1-113 1-1-114
CSR13!!.01uF;1-1-81 1-1-82
JMPR;1-1-52 1-1-72 1-1-120
RC07!!1K;1-1-88
RC07!!2K;1-1-92 1-1-91
RC07!!4.7K;1-1-75 1-1-89
RC07!!15K;1-1-87 1-1-90
$NETS
; U*:1-1-43.Y U1:1-1-9.5 U4:1-1-36.20
; U*:1-1-17.Y U1:1-1-9.11 U2:1-1-10.11
; J1:1-1-52.2 U4:1-1-36.21
; U*:1-1-101.Y U*:1-1-122.A
; U1:1-1-9.12 U2:1-1-10.5
; U*:1-1-122.Y U*:1-1-123.A U*:1-1-123.B
; U*:1-1-51.Y U5:1-1-37.21
; U*:1-1-58.Y U*:1-1-101.A U*:1-1-101.B U5:1-1-37.18
; R3:1-1-75.1 U*:1-1-123.Y
; Q1:1-1-64.2 R3:1-1-75.2 R4:1-1-87.1
; U2:1-1-10.12 U3:1-1-11.5
; Q1:1-1-64.1 Q2:1-1-65.1 R5:1-1-88.1
; J2:1-1-72.2 Q2:1-1-65.3 R2:1-1-91.2 U6:1-1-112.18
; U*:1-1-35.Y U3:1-1-11.11
; J3:1-1-120.2 U6:1-1-112.21
; Q1:1-1-64.3 R1:1-1-92.2 R6:1-1-90.1
; Q2:1-1-65.2 R6:1-1-90.2 R7:1-1-89.1
+5V; C1:1-1-1.1 C2:1-1-79.1 C3:1-1-80.1 C4:1-1-100.1 C5:1-1-116.1 C6:1-1-121.1,
C7:1-1-81.1 C8:1-1-82.1 J1:1-1-52.3 J3:1-1-120.3 P1:1-1-114.25 U1:1-1-9.4,
U2:1-1-10.4 U3:1-1-11.4
+11V; R1:1-1-92.1 R2:1-1-91.1
CS-; J2:1-1-72.3 P1:1-1-107.41 U*:1-1-58.B U*:1-1-122.B
GND; C1:1-1-1.2 C2:1-1-79.2 C3:1-1-80.2 C4:1-1-100.2 C5:1-1-116.2 C6:1-1-121.2,
C7:1-1-81.2 C8:1-1-82.2 P1:1-1-114.26 R4:1-1-87.2 R5:1-1-88.2 R7:1-1-89.2,
U1:1-1-9.14 U2:1-1-10.14 U3:1-1-11.14 U4:1-1-36.18 U5:1-1-37.20 U6:1-1-112.20
OD; J1:1-1-52.1 U3:1-1-11.7
R/W; P1:1-1-106.18 U*:1-1-51.A U*:1-1-51.B
XS; P1:1-1-106.1 U*:1-1-17.B U*:1-1-35.B U*:1-1-43.B
AB-0; P1:1-1-104.2 U*:1-1-43.A U5:1-1-37.8 U6:1-1-112.8
AB-1; P1:1-1-104.3 U*:1-1-35.A U5:1-1-37.7 U6:1-1-112.7
AB-2; P1:1-1-104.4 U*:1-1-17.A U5:1-1-37.6 U6:1-1-112.6
AB-3; P1:1-1-104.5 U5:1-1-37.5 U6:1-1-112.5
AB-4; P1:1-1-104.6 U5:1-1-37.4 U6:1-1-112.4
AB-5; P1:1-1-104.7 U5:1-1-37.3 U6:1-1-112.3
AB-6; P1:1-1-104.8 U5:1-1-37.2 U6:1-1-112.2
AB-7; P1:1-1-104.9 U5:1-1-37.1 U6:1-1-112.1
AB-8; P1:1-1-105.10 U5:1-1-37.23 U6:1-1-112.23
AB-9; P1:1-1-105.11 U5:1-1-37.22 U6:1-1-112.22
AB-10; P1:1-1-105.12 U5:1-1-37.19 U6:1-1-112.19
AB-11; J3:1-1-120.1 P1:1-1-105.13 U*:1-1-58.A
CB-0; U1:1-1-9.3 U4:1-1-36.8
CB-1; U1:1-1-9.2 U4:1-1-36.7
CB-2; U1:1-1-9.6 U4:1-1-36.6
CB-3; U1:1-1-9.7 U4:1-1-36.5
CB-4; U2:1-1-10.3 U4:1-1-36.4
CB-5; U2:1-1-10.2 U4:1-1-36.3
CB-6; U2:1-1-10.6 U4:1-1-36.2
CB-7; U2:1-1-10.7 U4:1-1-36.1
CB-8; U3:1-1-11.3 U4:1-1-36.23
CB-9; U3:1-1-11.2 U4:1-1-36.22
CB-10; U3:1-1-11.6 U4:1-1-36.19
DB-0; P1:1-1-113.49 U1:1-1-9.15 U3:1-1-11.15 U4:1-1-36.9 U5:1-1-37.9,
U6:1-1-112.9
DB-1; P1:1-1-113.48 U1:1-1-9.1 U3:1-1-11.1 U4:1-1-36.10 U5:1-1-37.10,
U6:1-1-112.10
DB-2; P1:1-1-113.47 U1:1-1-9.10 U3:1-1-11.10 U4:1-1-36.11 U5:1-1-37.11,
U6:1-1-112.11
DB-3; P1:1-1-113.46 U1:1-1-9.9 U3:1-1-11.9 U4:1-1-36.13 U5:1-1-37.13,
U6:1-1-112.13
DB-4; P1:1-1-113.45 U2:1-1-10.15 U4:1-1-36.14 U5:1-1-37.14 U6:1-1-112.14
DB-5; P1:1-1-113.44 U2:1-1-10.1 U4:1-1-36.15 U5:1-1-37.15 U6:1-1-112.15
DB-6; P1:1-1-113.43 U2:1-1-10.10 U4:1-1-36.16 U5:1-1-37.16 U6:1-1-112.16
DB-7; P1:1-1-113.42 U2:1-1-10.9 U4:1-1-36.17 U5:1-1-37.17 U6:1-1-112.17
$END
```

EDITING A NET DATA BASE WITH THE INCREMENTAL NET-LOAD TEXT FILE

1. THE INCREMENTAL NET-LOAD TEXT FILE
   - o PURPOSE
   - o PREREQUISITES
   - o RESTRICTION ON EDITING ASSIGNED COMPONENTS IN THE NET DATA BASE
2. STRUCTURE OF THE INCREMENTAL NET-LOAD TEXT FILE
3. NAMING THE INCREMENTAL NET-LOAD TEXT FILE
4. HOW THE SYSTEM USES THE INCREMENTAL NET-LOAD TEXT FILE
   - o NOTE ON THE FUNCTION AND PIN SWAPPING CAPABILITY
5. INPUTTING THE INCREMENTAL NET-LOAD TEXT FILE

EDITING A NET DATA BASE WITH THE INCREMENTAL NET-LOAD TEXT FILE

1. THE INCREMENTAL NET-LOAD TEXT FILE

PURPOSE

An existing net data base created from a schematic or text-input netlist may be updated with an incremental net-load text file. This text file is used to add and delete information in the current net data base file. The incremental net-load text file is a time-saving way to edit the net data base without re-creating the entire net data base file.

The format of the incremental net-load file is the same as the text-input netlist, with the addition of the two keywords, $DELETE and $ADD. The operator may specify deletions and additions to the net data base with these keywords. An example of an incremental net-load text file is shown below with arrows pointing to the $DELETE and $ADD keywords, identifying the file as an incremental net-load text file.

```
        (EXAMPLE: INCREMENTAL NET-LOAD TEXT FILE)
--->    $DELETE
        $PACKAGES
        C1 C2
        $FUNCTIONS
        AND[1-2]
        $NETS
        ;J.3
--->    $ADD
        $FUNCTIONS
        4000A ! INV ; INV1
        4000A ! NOR3; NOR3[1-2]
        $PINS
        R1.2; INV1.Y
        R1.1; NOR32.Y INV1.A
        J.6; NOR31.A NOR31.B NOR31.C
        J.5; NOR32.A NOR32.B NOR32.C
```

PREREQUISITES

The prerequisites for the incremental net-load text file are the same as those for the text-input netlist. Device files and/or pin files must be in the current project or in the SYSTEM-LIBRARY. In addition, the net data base to be updated must exist in the current project file. The operator should use the COMPONENT-REPORT, NETLIST-REPORT, and the BOM-REPORT for reference when inputting the incremental net-load text file.

RESTRICTION ON EDITING ASSIGNED COMPONENTS IN A NET DATA BASE

With the incremental net-load text file, there is one restriction; a component that has been placed in a board drawing with an assigned reference designator cannot be edited out of the net data base corresponding to that board drawing. The |DEASSIGN REF DES| and |DELETE COMPONENT| commands must be used on the board component in the drawing. The net data base may then be edited with the incremental net-load text file.

2. STRUCTURE OF THE INCREMENTAL NET-LOAD TEXT FILE

The format and contents of the incremental net-load text file are similar to that of the original text-input netlist. With the keywords $DELETE and $ADD, the operator may create the text file to delete and/or add information to the existing net data base. An additional keyword, $PINS, may be specified if the operator wishes to incrementally delete or add pin information to existing nets in the net data base.

If information is to be deleted from the net data base, the $DELETE section of the file must precede any additions to the net data base under $ADD. However, if only additions to the net data base are required, $DELETE is not required. The example below shows each keyword, and the flow of possible options that could follow:

```
        $DELETE                              $ADD
           |                                   |
 INFORMATION FROM LATEST          INFORMATION TO LATEST
       NET DATA BASE                    NET DATA BASE
           |                                   |
       $PACKAGES                          $PACKAGES
       $FUNCTIONS                         $FUNCTIONS
       $NETS                              $NETS
       $PINS                              $PINS
```

Under $DELETE, for example, the operator may specify packages, functions, nets and pin information for deletion from the current net data base. Under $ADD the operator may add package, function, net and pin information to the net data base.

3. NAMING THE INCREMENTAL NET-LOAD TEXT FILE

When inputting the incremental net-load text file, the operator must specify a text file name different from the original text-input netlist. Eighteen alphanumeric characters are allowed in the file name, with no blank spaces.

Like the text-input netlist, the |LOAD TXT NETLIST| command is used to update the net data base. When |LOAD TXT NETLIST| is picked, the the operator must specify the name of the incremental net-load text file. The system then updates the latest net data base with the deletions and additions identified in the incremental net-load file. With the $DELETE and $ADD keywords, the system is able to differentiate the incremental net-load text file from the original text-input netlist.

4. HOW THE SYSTEM USES THE INCREMENTAL NET-LOAD FILE

The examples below illustrate the original text-input netlist used to create the net data base, the incremental net-load text file used to update the net data base created from the original netlist, and the netlist file reflecting the updated net data base.

EXAMPLE: ORIGINAL TEXT-INPUT NETLIST USED TO CREATE THE NET DATA BASE

```
(TEXT NETLIST FILENAME: NETLIST)
$PACKAGES
CON15S/156 ; J
CAPRAD300 ; C1 C2
RES400 ! RLR05  ; R1 R2
$FUNCTIONS
74LS00 ; AND[1-4]
74L74 ; FLIP[1-2]
$NETS
; J.8 AND2.B
; J.7 AND2.A
; J.4 AND1.B
; J.3 AND1.A
GND; J.1  C2.2  C1.2  R2.2
; J.11  R1.2
; R1.1  FLIP2.QN
; FLIP1.Q  AND3.A
; FLIP2.Q  AND3.B
; R2.1  FLIP1.QN
+5V ; J.12  J.10  C2.1  C1.1  FLIP1.PRE  FLIP2.PRE  AND3.Y
; AND1.Y  FLIP1.D
; J.6  FLIP2.CLK  FLIP1.CLK
; J.5  FLIP2.CLR  FLIP1.CLR
```

EXAMPLE: INCREMENTAL NET-LOAD TEXT FILE USED TO UPDATE THE NET DATA BASE CREATED FROM THE ORIGINAL TEXT-INPUT NETLIST

```
(TEXT-INPUT NETLIST - FILENAME: UPDATE)
$DELETE
$PACKAGES
C1 C2
$FUNCTIONS
AND[1-2]
$NETS
;J.3
$ADD
$FUNCTIONS
4000A ! INV ; INV1
4000A ! NOR3; NOR3[1-2]
$PINS
R1.2; INV1.Y
R1.1; NOR32.Y INV1.A
J.6; NOR31.A NOR31.B NOR31.C
J.5; NOR32.A NOR32.B NOR32.C
```

After the incremental net-load file is input and saved in the current project file, the operator may then use the LOAD TXT NETLIST command to update the latest net data base.

EXAMPLE: |LOAD TXT NETLIST| (FILENAME) UPDATE |ENTER|

When ENTER is picked, the system proceeds to update the net data base with the information from the incremental net-load file. After the net data base has been updated, the operator may pick the CREATE TXT NETLIST command to generate an updated netlist, reflecting the contents of the net data base.

EXAMPLE: |CREATE TXT NETLIST| (FILENAME) NETLIST 2 |ENTER|

RESULTING NETLIST FILE SHOWING THE ASSIGNMENTS IN THE NET DATA BASE

```
$PACKAGES
CON15S/156 !  ;  J
DIP14 ! 4000A ;  U3
RES400 ! RLR05  ; R2
RES400 ! RLR05  ; R1
DIP14  ! 74LS00 ; U1
DIP14  ! 74L74  ; U2
$NETS
GND; U3.7  U2.11  U1.7  R2.2  J.1
; J.4
; U3.13  U3.12  U3.11  U2.5  U2.3  J.5
; U3.5  U3.4  U3.3  U2.7  U2.1  J.6
; J.7
; J.8
+5V; U3.14  U2.4  U1.14  U2.8  U2.14  U1.10  J.12  J.10
; U3.9  R1.2  J.11
; U2.12  R2.1
; U3.10  U3.8  U2.10  R1.1
; U2.2
; U2.13  U1.8
; U2.9  U1.9
$END
```

The netlist generated by the |CREATE TXT NETLIST| command shows the assignments made in the net data base. Like the original text netlist, the |LOAD TXT NETLIST| command performs automatic assignment of reference designators to component packages from the devices listed under the $ADD / $FUNCTIONS section of the incremental net-load text file. For example,

INCREMENTAL NET-LOAD FILE

```
$ADD
$FUNCTIONS
4000A !  INV  ; INV1
4000A !  NOR3 ; NOR3[1-2]
```

NETLIST FILE GENERATED BY CREATE TXT NETLIST REFLECTING THE THE CONTENTS OF THE NET DATA BASE

```
$PACKAGES
DIP14 ! 4000A ; U3
```

In the above example, the system automatically assigned reference designator U3 to the net data base for the device 4000A. Since reference designators U1 and U2 were already contained in the net data base, the system used the next available reference designator, U3.

Since device description files are required for additions to net data base specified under $FUNCTIONS, the system uses the information in each device file to make the initial pin assignments in net data base. For example,

INCREMENTAL NET-LOAD FILE

```
$ADD
|
|
$PINS
R1.2; INV1.Y
```

In the example, the function designator.pin name INV1.Y is to be added to an existing net in the net data base. Here, the operator has specified that INV1.Y be added to the net that contains reference designator.pin number, R1.2.

RESULTING NET INFORMATION IN THE NETLIST FILE GENERATED BY THE CREATE TXT NETLIST COMMAND.

```
$NETS
; U3.9  R1.2  J.11
```

In the example, the system used the reference designator U3 for for the function designator INV1. When the system read the 4000A device file, it assigned pin 9 to the reference designator U3, then added the information to the net containing R2.1.

; U3.9 R1.2 J.11

NOTE ON THE FUNCTION AND PIN SWAPPING CAPABILITY

Like the net data base created from original text-input netlist, the function and pin swapping capability may be used on the placed board drawing if device files are used with the incremental net-load text file to update the latest net data base.

5. INPUTTING THE INCREMENTAL NET-LOAD TEXT FILE

The format of the incremental net-load file is similar to that of the original text-input netlist. However, there are a few exceptions. This section describes the input format of the incremental net-load text file for deleting and adding information to the existing net data base.

$DELETE

The $DELETE keyword must be the first line of the text file if information is to be deleted from the net data base. That is, if deletions and additions are required, $DELETE must precede $ADD in the file.

o PACKAGES

When deleting packages from the net data base, the operator is only required to specify the reference designators of the packages to be deleted.

EXAMPLE:

```
$DELETE
$PACKAGES
C1  C2
```

INPUT SEQUENCE

- Simply input the names of the references designators to be deleted on the line following the $PACKAGES keyword.

- A semi-colon is not required before the list of reference designators to be deleted. Blank spaces or commas may be used to separate reference designators in the list.

- In the above example, components C1 and C2 will be deleted when the operator uses the |LOAD TXT NETLIST| command to update the net data base. The system also detaches the pins of components C1 and C2 from any existing nets.

---

o $FUNCTIONS

When specifying functions to be deleted, the operator must input the function designators to be deleted. Any number of functions within one device may be specified.

For example, the information below shows a function designator specified in the original netlist used to create the net data base.

ORIGINAL NETLIST

```
$FUNCTIONS
74LS00   ; AND[1-4]
```

In the above example, the device 74LS00 contains four functions. The operator may delete the function designator and all functions, or simply free one, two, or three functions associated with the device.

The example below illustrates a typical entry in the incremental net-load text file that frees two of the functions associated with the device 74LS00.

INCREMENTAL NET-LOAD FILE

```
$DELETE
$FUNCTIONS
AND[1-2]
```

The function designators specified under $FUNCTIONS will be freed when |LOAD TXT NETLIST| updates the net data base. That is, all assigned reference designators and pin numbers in the net data base associated with the functions AND1 and AND2 will be cleared from any nets they appear in. However, the reference designators and pin numbers that were assigned to the functions named AND3 and AND4 will remain in the net data base if net information (function des.pin name) was specified in the original netlist.

EXAMPLE: $FUNCTIONS AND $NETS SECTIONS OF THE ORIGINAL NETLIST

```
$FUNCTIONS
74LS00 ; AND[1-4]
$NETS
; J.3  AND1.A
```

NET INFORMATION IN THE NET DATA BASE AFTER
|LOAD TXT NETLIST|

```
;J.2  U1.1
```

(Reference Designator U1 was assigned by the system; pin number 1 was assigned from the information in the 74LS00 device file.)

INCREMENTAL NETLOAD FILE

```
$DELETE
$FUNCTIONS
AND1
```

RESULTING NET INFORMATION IN THE NET DATA BASE
AFTER |LOAD TXT NETLIST|

```
; J.2
```

(Pin 1 on component U1 is now unused.)

INPUT SEQUENCE

- Simply input the names of the function designators to be deleted, or freed, starting on the line following the $FUNCTIONS keyword.

- A semi-colon is not required before the list of function designators. A blank space or a comma may be used to separate function designators in the list. Use brackets to specify a range of functions.

EXAMPLES:

```
$DELETE              $DELETE
$FUNCTIONS           $FUNCTIONS
AND1 AND2 AND3       AND[1-3]
```

o NETS

Entire nets may be deleted from the net data base by specifying one reference designator.pin number within the net. A function designator may also be specified to delete entire nets.

For example, the information below shows one net from the original text netlist that exists in the net data base.

ORIGINAL NETLIST

```
$NETS
; J.3  AND1.A
```

To delete this net from the net data base, the operator may specify either the reference designator J.3 or the function designator AND1.A to delete the entire net from the net data base. For example,

INCREMENTAL NET-LOAD FILE

```
$DELETE              $DELETE
$NETS        OR      $NETS
; J.3                ; AND1.A
```

In the examples above, either net-load file will delete the the entire net from the net data base.

When |LOAD TXT NETLIST| is picked, the system searches for the reference designator.pin number, or the function designator.pin name, then deletes the contents of the entire net.

INPUT SEQUENCE

- Input a semi-colon at the start of the line following the the $NETS keyword.
- Input the reference designator.pin number contained within the net. The system will delete the contents of the entire net from the net data base. A function designator.pin name may be specified.
- Each net to be deleted must begin on a new line.
- Net names may be specified, if desired. Input the net name before the semi-colon. The contents of that net will be deleted.

EXAMPLE:

```
$DELETE
$NETS
; J.3
; FLIP1.Q
+5V;
```

o $PINS

The operator may specify pins to be deleted from a net with the $PINS section of the incremental net-load text file. Only the pin information specified is deleted from the net data base.

When the |LOAD TXT NETLIST| command is picked, the system searches the net data base for the information listed under $PINS. The system then deletes the pin information from the existing nets. For example, the existing net is contained in the net data base:

```
; U3.5  U3.4  U2.5
```

If the operator chooses to delete the connection to pin 4, assigned to reference designator U3, the entry in the incremental net-load file will look like this:

```
$DELETE
$PINS
U3.4
```

The resulting net in the net data base after LOAD TXT NETLIST would be,

```
; U3.5  U2.5
```

Function designators may also be specified, for example:

```
$DELETE
$PINS
AND3.A
```

The reference designator and pin number assigned in the net data base for the function AND3, pin name A, would be cleared from its existing net. The pin number is now unused.

INPUT SEQUENCE

- Input the list of pins at the start of the line following the $PINS keyword. Separate entries in the list with a blank space, or a comma.

- To specify a range of pin numbers within a device, brackets may be used. Example: Z1.[3-5] deletes the pins Z1.3, Z1.4 and Z1.5 from the net data base.

- Input a comma at the end of the line if continuing on the line below.

$ADD

The $ADD keyword must follow the $DELETE section of the incremental net-load text file. If $DELETE was not used, the $ADD keyword must be the first line of the text file.

For:

- o $PACKAGES
- o $FUNCTIONS
- o $NETS

The formats used for adding packages, functions and nets are the same as those for the text-input netlist. The example below illustrates entries under the $ADD keyword.

EXAMPLE:

```
$ADD
$PACKAGES
DIO400 ;  D1  D2
$FUNCTIONS
4000A ! INV ; INV1
4000A ! NOR3 ; NOR3[1-2]
$NETS
; D1.1  AND4.Y
```

Refer to the section on creating the text-input netlist for additional formatting information.

o $PINS

To add pins to existing nets in the net data base, the operator must first specify the existing net, then the pin information to be added.

For example, the following net is contained in the net data base.

```
; J.11  R1.2
```

To add pin information to the above net, the operator must input a pin currently in the net, then the pin information to be added. For example, if the function INV1.Y is to be added, the incremental net-load file may be input as follows:

```
$ADD
$PINS
R1.2; INV1.Y
```

When |LOAD TXT NETLIST| is picked, the system searches for the pin R1.2, then adds the function INV1.Y to that net. The system assigns a reference designator and a pin number to the function.

In the example below, the system used the reference designator U3 for the function designator INV1. When the system read the 4000A device file, it assigned pin 9 to the reference designator U3, then added the information to the net containing R2.1.

RESULTING NET IN THE NET DATA BASE

```
; U3.9  R1.2  J.11
```

INPUT SEQUENCE

- Input the keyword $PINS, then proceed to the next line of the file.

- Specify a current pin number contained in the net being added to. A semi-colon must follow the pin information.

```
EXAMPLE:    $PINS
            R1.2;
```

- Specify the pins to be added to the net following the the semi-colon. Use the pin name if a function designator is specified.

```
EXAMPLE:    $PINS
            R1.2;  INV1.Y
```

  A range of pins within a device may be specified by using brackets. For example,

```
            R1.2; Z1.[3-5]
```

  will add Z1.3, Z1.4, and Z1.5 to the existing net containing pin, R1.2.

- Each net specified must begin on a new line, followed by a semi-colon, then the pin information to be added.

ADDITIONAL METHODS USED TO EDIT A NET DATA BASE

DEFINITIONS

TEXT NETLIST - A text file containing net data base information that can be used with the |LOAD TXT NETLIST| command to create or edit a net data base.

TEXT-INPUT NETLIST - A text netlist keyed-in by the user, as opposed to one created by the system when the user picks the |CREATE TEXT NETLIST| command.

(Do not confuse these with NETLIST-REPORT which simply reports informaton and cannot be used to create or edit a net data base.)

PURPOSE AND METHODS

Once you have created a net data base, you may wish to change it because of an error, a design change, or an engineering change order. There are three ways to edit a net data base:

1. Change the original schematic drawing(s) and repeat the |EXTRACT NETLIST| command.

2. Change the original text-input netlist and repeat the |LOAD TXT NETLIST| command.

3. Use the |CREATE TEXT NETLIST| command to create a text netlist from the existing net data base, edit the resulting text netlist, then use the |LOAD TXT NETLIST| command.

Methods 2 and 3 are very similar except that with method 3 you use the |CREATE TX NETLST| command to create a text netlist that exactly and accurately reflects the contents of the net data base at the moment you are ready to edit. This insures much greater accuracy than method 2 because you cannot be sure that the original text-input netlist exactly reflects the net data base, particularly if some time has elapsed since you originally created the net data base.

Method 3 is useful for updating a net data base with changes from a small ECO. Refer to EDITTING A NET DATA BASE WITH THE INCREMENTAL NET-LOAD TEXT FILE in this section for procedures to update a net data base with changes from a large ECO.

All three methods are discussed in the tables below.

RESTRICTION ON EDITING ASSIGNED COMPONENTS IN A NET DATA BASE

With all three methods you will encounter a restriction: a component that has been placed in a board drawing and that has an assigned reference designator cannot be edited out of the net data base corresponding to that board drawing. You must first use the |DELETE COMPONENT| command or the |DEASSIGN REF DES| command on the component in the drawing. You may then edit it out of the net data base.

EDITING THE NET DATA BASE BY EDITING THE SCHEMATIC	
PREREQUISITES	- The net data base and the schematic drawing(s) must be in the current project file. - The net data base must have been created and edited only with the |EXTRACT NETLIST| command from a schematic, never with the |LOAD TEXT NETLIST| command. - The schematic drawing must have the same name as the one orginally used to create the net data base. It need not have the same revision label.
WHEN TO USE THIS METHOD	- Use this method when you want to keep your schematic drawings up to date, and when you want to be able to use the back annotate capability. (This capability can only be used with a net data base that has been created and edited exclusively from a schematic.)
HOW TO USE THIS METHOD	- Edit the schematic drawing(s). - Use the |EXTRACT NETLIST| command. - Check the EXTRACTION-LOG for errors. - Correct errors and repeat the process until the -LOG is satisfactory.

EDITING THE NET DATA BASE BY EDITING THE TEXT-INPUT NETLIST	
PREQUISITES	- The net data base and the text-input netlist must be in the current project file.
WHEN TO USE THIS METHOD	- This method is only recommended for correcting errors shown in the NET-LOAD-LOG immediately after the |LOAD TEXT NETLIST| command is used.
HOW TO USE THIS METHOD	- Edit the text-input netlist. - Use the |LOAD TEXT NETLIST| command. - Check the NET-LOAD-LOG for errors. - Correct errors and repeat the process until the -LOG is satisfactory.

EDITING THE NET DATA BASE WITH THE |CREATE TEXT NETLIST| COMMAND

PREREQUISITES	- The net data base must be in the current project file.
WHEN TO USE THIS METHOD	- This method is probably the most error-proof means of editing a net data base. - It can be used to edit a net data base whether the net data base was created from a schematic |EXTRACT NETLIST| or from a text-input netlist |LOAD TEXT NETLIST|. - As mentioned above, this method is more accurate than editing a text-input netlist. - It is also more accurate than editing a schematic for the same reason — you can never be sure that the schematic is an exact reflection of what is in the net data base, particularly if some time has elapsed since the creation of the net data base. - This method is often quicker and more convenient than editing a schematic. - Be aware, however, that you will not be able to back annotate your schematic once you have used the |LOAD TEXT NETLIST| command to edit the net data base. - However, if |CHANGE REF DES| is performed on the board drawing, the file created by |CREATE TEXT NETLIST| will reflect the new REF DES.
HOW TO USE THIS METHOD	- Use the |CREATE TEXT NETLIST| command. (See the Command Description section of the manual). - Edit the resulting text netlist just as you would edit any text file. Be sure to adhere to the format requirements for a text-input netlist. (See p. NETDB-23). - Use the |LOAD TEXT NETLIST| command. - Check the NET-LOAD-LOG for errors. - Correct errors and repeat the process until the -LOG is satisfactory.

B A C K   A N N O T A T I O N

1. BACK ANNOTATION
   - o PURPOSE
   - o BACK ANNOTATING REFERENCE DESIGNATORS
   - o BACK ANNOTATING PIN NUMBERS
2. THE BACK ANNOTATE COMMANDS
   - o DIFFERENCE BETWEEN BACK ANNOT ALL AND BACK ANNOT CUR
3. PREREQUISITES FOR THE BACK ANNOTATION COMMANDS
4. THE ANNOTATE-LOG

1. BACK ANNOTATION

PURPOSE

The back annotation capability accomplishes two things:

1. It brings reference designators on a schematic into conformity with reference designators in a board drawing.

2. It fills in blank or missing pin number labels on a schematic by using pin number information from device description files.

BACK ANNOTATING REFERENCE DESIGNATORS

When you use the |ASSIGN REF DES| command on a board drawing, you establish a link in the net data base between board components and schematic symbols having the same reference designators.

When you use the |CHANGE REF DES| command on the board drawing, this link remains, but the board components now have reference designators labels different from the schematic symbols to which they are linked.

Back annotation is the means by which you bring the schematic reference designator labels into conformity with the changed board reference designators.

BACK ANNOTATING PIN NUMBERS

If you have a device library, you may put one pin number label on each symbol in your schematic and leave the others blank. Then, when you use the |EXTRACT NETLIST| command, the system creates a net data base using the pin numbers on the schematic and filling in the blank numbers from the device files.

At this point, the net data base has a complete set of pin numbers, but the schematic does not. Back annotation is the means by which you label the schematic with pin numbers in the net data base.

2. THE BACK ANNOTATION COMMANDS

There are two back annotation commands:

|BACK ANNOT ALL|

|BACK ANNOT CUR|

The Command Description Section tells you how to use these two commands. The information below gives you the general points to bear in mind when using them.

DIFFERENCE BETWEEN |BACK ANNOT ALL| AND |BACK ANNOT CUR|

Use one or the other of these commands to back annotate a schematic; do not use both.

|BACK ANNOT CUR| - This command back annotates only the currently active schematic drawing. Use this command when you wish to back annotate only one sheet, or when you wish to back annotate one sheet at a time.

|BACK ANNOT ALL| - This command back annotates all schematic sheets related to a board in the net data base. Use this command when you wish to back annotate all sheets at the same time.

YOU MAY USE THE BACK ANNOTATE COMMANDS AS OFTEN AS YOU LIKE

You may want to wait to back annotate the schematic until you have finished annotating the board drawing. Or you may want to back annotate several times while working on the board drawing in order to keep your schematic fairly current with your board. You may even want to back annotate before you begin the board drawing in order fill in blank pin numbers.

3. PREREQUISITES FOR THE BACK ANNOTATION COMMANDS

1. There must be a net data base in the current project file. If you have changed the schematic drawing(s), you must repeat the |EXTRACT NETLIST| command in order to bring the net data base up to date with the current schematic. Otherwise, you may encounter errors when you use the back annotation commands. (If you have changed the name of the schematic(s), back annotation will not work properly.)

2. The schematic sheet(s) to be annotated must be in the current project file.

   For |BACK ANNOT CUR|, the schematic must be the active drawing.

   For |BACK ANNOT ALL|, no drawing should be active.

4. THE ANNOTATE-LOG

When you use either of the back annotation commands, the system creates a text file named ANNOTATE-LOG. This text file lists all drawing files that were back annotated, and it lists error conditions.

EXAMPLE:

```
BACK ANNOTATION STARTED AT 31-AUG-82 14:32:14

starting back annotation for schematic SCH/2 rev ANTD
function U5 has been deleted from the drawing but still exists...
...in the NET-DATA-BASE
function U4 has been deleted from the drawing but still exists...
...in the NET-DATA-BASE
function C9 has been deleted from the drawing but still exists...
...in the NET-DATA-BASE
function C25 has been deleted from the drawing but still exists...
...in the NET-DATA-BASE
back annotation of schematic complete
```

ERROR CONDITIONS

As you can see from the example above, error conditions are usually discrepancies between the schematic drawing and the net data base.

Such discrepancies are almost always the result of changing the schematic but failing to repeat the |EXTRACT NETLIST| command.

5. THE EDA-1000 BACK ANNOTATION CAPABILITY

A. INTRODUCTION

With the EDA-1000 Design Capture System, the operator may back annotate EDA-1000 schematic sheets from the resulting PC board design created on the EDA-3000 Printed Circuit Board Design System.

The back annotation process is summarized below:

EDA-1000

1. The operator creates a schematic on the EDA-1000 Design Capture Editor.

2. From the resulting schematic, the operator generates a pinlist for each schematic sheet using the Pinlist Processor.

3. Using the Netlist Processor, the operator generates a netlist for the entire schematic (all sheets).

4. The operator then runs the TELNET program to create a Telesis-fmroatted netlist file. The file must then be transferred to a project file on the EDA-300 or EDA-700 workstation.

EDA-300

1. When the Telesis-formatted netlist file is transferred to the EDA-300 or 700, the operator uses the LOAD TEXT NETLIST command to create the net data base file. The net data base contains the net, pin, reference designator, function, and device type information needed to create a PC board on the EDA-300. The operator then proceeds to create and complete the PC board design.

2. During the creation of the PC board, changes may have been made to the design. For example, if the pin and function swapping capability was used during board placement, the original EDA-1000 schematic will not reflect these changes. As a result, Back Annotation is necessary for the accurate updating of the EDA-1000 schematic sheets.

3. To begin the back annotation process from the EDA-300, the operator must select the CREATE TEXT BACK ANNOTATE FILE command. This command creates the text file to be used by the EDA-1000 software for back annotation of the schematic sheets.

4. When the back annotation text file is created, the file must be transferred to the EDA-1000 system. The operator then runs EDA-1000 back annotation program to update the schematic sheets.

B. LIMITATIONS OF EDA-1000 BACK ANNOTATION

Currently, hierarchical or structured designs created on the EDA-1000 cannot be back annotated.

C. STARTING THE BACK ANNOTATION PROCESS

Using the CREATE TEXT BACK ANNOTATE FILE command

When the operator completes the creation of the PC board design on the EDA-300, the CREATE BACK ANNOTATE FILE command may be used to create the text file needed to back annotate the EDA-1000 schematic sheets. This command is located on the TEXT NETLIST menu.

When CREATE BACK ANNOTATE FILE is selected, the system extracts the information from the latest net data base file in the current project. As the system creates the file, the system responds with a series of messages on the short message line:

```
10% OF BACK ANNOTATION FILE GENERATED
20% OF BACK ANNOTATION FILE GENERATED
30% OF BACK ANNOTATION FILE GENERATED
            |
            |
            |
BACK ANNOTATION FILE GENERATION COMPLETE
```

The resulting back annotation file is named BACK-FILE. This name will appear in the current project directory. If an existing file named BACK-FILE currently exists, the file will be overwritten upon generation of the new file when CREATE BACK ANNOTATE FILE is selected

EXAMPLE: Back Annotation File: BACK-FILE

```
BACK ANNOTATION,2
(FDES,1-1,U1
PART,74LS00
PIN,A,1
PIN,B,2
PIN,Y,3
)
(FDES,1-2,U2
PART,74LS04
PIN,A,1
PIN,Y,2
)
(FDES,103,R1
PART,RES1K
VAL,1K
PIN,1,1
PIN,2,2
)
```

The back annotation file is similar in concept to the Version 2 pinlist and netlist files generated by the EDA-1000 software.

The first record identifies the file as a back annotation file that follows Version 2 formatting conventions.

The remainder of the file consists of one or more groups of back annotation records. A back annotation record consists of a function designator by one or more pin records.

The ´FDES´ record type indicatese that a function designator record follows. The field order is: function designator, then reference designator. This record will always be the first record in the group.

The function designator has the following format:

File reference number ´-´ Symbol Reference Number

These fields are linked together without blank spaces. The function designator may contain up to eight characters. Examples of legal function designators include: 1-1-10, 10-9-9999, and 5-1-1000. Both the file reference number and the symbol reference number are taken from the EDA-1000 netlist [.NET]. The correct drawing is determined using the file reference number from the drawing header record in the EDA-1000 netlist [.NET].

The reference designator may be of variable length, with up to eight characters.

The 'PART' type indicates that a part number follows. All symbols are required to have a part number when used in conjunction with a Telesis system. The part number may also be the device type (ex. 7400).

The 'VAL' type indicates that a component value record follows. These are most often used for capacitors and resistors.

The 'PIN' type indicates that a pin record follows. The field order is pin name, pin number. Pin name is the old value field and pin number is new value. FDES, VAL, PART, and PIN appear on the same line as their corresponding fields. As a result, pins, device types, values, and reference designators may be back annotated.

File Transfer

Using the file transfer capability, transfer the file named BACK-FILE to the EDA-1000. Ensure that the file is transferred to the proper directory, if not the current directory. Refer to UTILITIES-154 for details on using the file transfer capability.

Back Annotating the EDA-1000 Schematic Sheets

After the back annotation file is transferred to the EDA-1000 the operator may begin back annotation of the schematic sheet(s). To invoke the back annotation program, from DOS, type:

```
C>BACKANNO
```

The editor then responds with the following message:

```
EDA-1000 Back Annotation Program Version 2
```

The following prompt is then displayed:

```
Do you wish to overwrite your drawing files (Y/N)???
```

If N (no) is specified, the back annotation files will be created in the default directory with a .BAK extension. Otherwise, the original files will be overwritten.

The program next prompts the operator for the original netlist file [.NET] that was used to input to the Telesis netlist conversion program, TELNET. The prompt is:

> Please enter the name of the EDA-1000 netlist file [.NET] or
> press ENTER only to terminate:

A directory path specification is required if the operator is not in the proper directory.

If the operator presses the ENTER key, the program is immediately terminated. If the file specified does not exist, the following error message is generated:

> "File not found or invalid file extension"

The operator is then prompted for the netlist file.

The program opens the netlist file [.NET] and scans it for all the drawing records that define the file reference number and the drawing file names. The program must attempt to open each drawing file to determine if it exists. If a drawing file does not exist, the following error message is generated for each file that could not be located:

> "The drawing file, <drawing file name>, could not be found"

The string enclosed in the angle brackets is the drawing file name that could not be located. As a result, the back annotation program is immediately terminated.

Next, the program prompts the operator for the back annotation file generated by the EDA-300 workstaion:

> "Please enter the name of the EDA-1000 back annotation file [.BAF]"

Enter the name of the back annotation file that was transferred to the EDA-1000. The system assumes the .BAF file extension if not specified. If the file name does not exist, the system will respond with an error message. The oeprator will then be reprompted for the EDA-1000 back annotation file.

The system then requests the name of the device type label attribute number. This is the device type attribute that was specified during the creation of the schematic sheet(s). The TELNET program also required this attribute number. The prompt is:

Device type label attribute number [NO LABEL]

Input the appropriate attribute, then press the <Return> key. If this attribute is not supplied, then only PART files will be changed. Simply press the <Return> key for the NO LABEL default.

The system then requests the name for the back annotation log file:

Logging file [<Backfile>LOG]

Specify the name to be used for the log file. If <Enter> is pressed without a filename, the system uses the back annotation file name with the .LOG extension. All warnings and error messages are recorded in the log file.

The back annotation process now begins. The following messages are displayed on the video screen:

"Beginning to back annotate file: <drawing file name>

As each function from that schematic sheet is being back annotated, the system returns the message:

"Beginning to back annotate symbol <symbol number>.

These messages continue until all schematic sheet(s) have been back annotated. When complete, the system displays the DOS prompt, C>.

D. ERROR MESSAGES

Introduction

The back annotation program generates messages to the log file using the same format as the TELNET processor. The log file contains a set of header records that indicates the back annotation program revision number, day, time and the .BAF file used as the back annotation file.

The back annotation program follows the header records with the message (if any). Each message contains two optional parts and one required part. If the message is caused by a particular input line, then the two optional parts are included. These parts are the line number of the .BAF record that caused the message and the line number of the .BAF record that caused the message and the actual text of the record. The required part is the message itself. The message consists of two parts: an indicator fo the seriousness of the message which is either 'WARNING' or 'ERROR' and the text of the message.

Messages containing the optional parts are called "line messages." The other message type is "symbol or sheet messages." The messages are described next.

Line Messages

This section describes messages that the back annotation program generates in response to a particular input line in the .BAF file.

"<func des> can not be located on drawing <drawing file>."

The user has probably added a function designator using the EDA-300 workstation that follows the format expected but can not be located on the current sheet. This is an error.

"0-0-0 is not associated with any of the drawing files in the directory."

The user has probably added a function designator using the EDA-300 workstation that does not follow the format expected. This is an error.

"<pin name> can not be located on symbol <symbol number> on drawing <drawing file>."

Somehow the pin number can not be matched to the symbol on the current sheet. The user might have edited the drawing after the netlists were generated.

"No circuit designator on symbol <symbol number> on sheet <drawing file>."

The circuit designator field has been deleted from the symbol or was never there. This is an error.

"No part number on symbol <symbol number> on sheet <drawing file>."

The part number field has been deleted from a symbol or was never there. This is an error.

"No component value on symbol <symbol number> on sheet <drawing file>."

The component value field has been deleted from a symbol or was never there. This is an error.

"Symbol <symbol number> on sheet <drawing file> has previously been back annotated."

Somehow, duplicate copies of a symbol got into the back annotation file. This is an error.

Pin name <pin name> on symbol <symbol number> on sheet <drawing file> has previously been back annotated."

Somehow, duplicate copies of the PART record got into the back annotation file. This first instance is used to back annotate the schematic. This is an error.

"Part number on symbol <symbol number> on sheet <drawing file> has previously been back annotated. Only one PART record is allowed per FDES."

Somehow, duplicate copies of the PART record got into the back annotation file. The first instance is used to back annotate the schematic. This is an error.

"Component value on symbol <symbol number> on sheet <drawing file> has previously been back annotated. One one VAL is allowed per FDES."

Somehow, duplicate copies of the VAL record got into the back annotation file. The first instance is used to back annotate the schematic. This is an error.

Symbol and Sheet Messages

Symbol messages are detected on a symbol by symbol basis. Sheet messages may only be detected when an entire sheet has been processed. Recall that there is no offending record so no record will appear in the logfile. This section desribes possible messages.

"The following pins on symbol <symbol number> on sheet <drawing file> were never back annotated. <pin1> ... <pinn>."

Either pins that were always unconnected or perhaps deleted by the PC board designer are not in the final board description. Pins with a signal name of N/C generate this message.

"The following symbols on sheet <drawing file> were never back annotated. <symbol number> ... <symbol numbern>."

These symbols were probably deleted by the PC board designer and were not described in the .BAF file. This is a warning.

"The part number field <part>, on symbol <symbol number> on sheet <drawing file> was never back annotated."

There should be a part number field in the .BAF file. Something is wrong. Please contact your Telesis representative if you see this message.

"The component value field <value>, on symbol <symbol number> on sheet <drawing file> was never back annotated."

The PCB designer deleted the component value field using the EDA-300 and as such, the appropriate information was not in the .BAF file.

There is a class of messages called "text update messages." They all may be described by the following template:

> The new <field type>, <field value>, which replaces <field value> on symbol <symbol number> on sheet <sheet number> <problem>.

Where:

> field type="reference designator," part number," "value" (or) "pin number"

and

> problem = "overlaps an existing alphanumeric field." (or)
> "overlaps the symbol boundary."

An example message is:

> the part number, 74LS00, which replaces 7400 on symbol 5 on sheet sch.dwg overlaps the symbol boundary.

The back annotation program generates these messages when it updates the text fields. Text fields may not overlap each other or extend outside the symbol boundary. The four types of fields that are back annotated are: circuit designator, part number, value and pin number.

THE NETLIST-REPORT, COMPONENT-REPORT, AND
BILL OF MATERIALS REPORT

1. GENERAL INFORMATION

2. THE NETLIST-REPORT

3. THE COMPONENT-REPORT

4. THE BILL OF MATERIALS REPORT

1. GENERAL INFORMATION

REPORTS CAN BE GENERATED FROM THE NET DATA BASE

After you have created a net data base, you can instruct the system to create a text file (or files) listing the information in the net data base. The three text files that you can generate contain similar information, but they present the information in different formats:

NETLIST-REPORT:	This lists the net data base information by net number.
COMPONENT REPORT:	This lists the net data base information by component symbol name.
BOM-REPORT:	This is a bill of materials.

PURPOSE

These text files show you the information in a net data base; the net data base file itself cannot be viewed or printed.

The system does not make any use of these reports; they are for your information only.

PREREQUISITES FOR CREATING THESE REPORTS

- A net data base file must be in the current project file. If there is more than one revision, the system takes the information from the most recently used net data base revision.

PRINTING AND VIEWING

These reports are ordinary text files. They are printed or displayed (opened) in the ordinary way except that there are three commands making it possible for you to print them without keying in their names:

PRINT NETLIST-REPORT
PRINT COMPONENT-REPORT
PRINT BOM-REPORT

Usually, you will want to print these files rather than view them on the graphics screen. However, if you do want to open these text files, be sure to input the file name exactly as shown here: NETLIST-REPORT, COMPONENT-REPORT, BOM-REPORT.

2. THE NETLIST-REPORT

PURPOSE

To be sure that a net data base contains the information you want, check the NETLIST-REPORT against your original engineering schematic.

HOW TO CREATE A NETLIST-REPORT

Use this command:

```
CREATE NETLIST-REPORT
```

EXAMPLE

```
ALPHA  REV 1                                                 13-MAY-82  09:47:22
NET LIST REPORT                                              PAGE       1
NET     NET                   SCHEMATIC          NET PINS
NUMBER  NAME                  SHEET              [* = duplicate pin]
 1                            WW                 R1.1  R2.1  R3.1  R4.1  R5.1  R6.1
                                                 R7.1
 2                            WW                 R1.2  Z1.6*  Z1.6*
 3                            WW                 R2.2  Z1.3*  Z1.3*
 4                            WW                 R3.2  Z1.4*  Z1.4*
 11                           WW                 R4.2  Z1.7*  Z1.7*
 12                           WW                 R5.2  Z1.12* Z1.12*
 13                           WW                 R6.2  Z1.11* Z1.11*
 14                           WW                 R7.2  Z1.10* Z1.10*
 15                           WW                 R8.1  Z3.1
 16                           WW                 R8.2  Z1.9*  Z1.9*
 17                           WW                 Z1.8*  Z1.8*  Z3.8
 18                           WW                 Z1.16* Z1.16* Z3.16
 19                           WW                 Z1.1*  Z1.1*
 20                           WW                 Z1.15* Z1.15*
 21                           WW                 Z1.14* Z1.14*
 22                           WW                 Z1.13* Z1.13*
 23                           WW                 Z1.5*  Z1.5*  Z3.5
 24                           WW                 Z3.3
 25                           WW                 Z3.4
 26                           WW                 Z3.7
 27                           WW                 Z3.12
 28                           WW                 Z3.11
 29                           WW                 Z3.10
 30                           WW                 Z3.9
 31                           WW                 Z3.2
 32                           WW                 Z3.6
```

3. THE COMPONENT-REPORT

PURPOSE

The COMPONENT-REPORT is most useful when you have an engineering change order (ECO). It provides you with a list of the nets associated with each component.

HOW TO CREATE A COMPONENT-REPORT

Use this command:

```
CREATE COMPONENT-REPORT
```

EXAMPLE

```
ALPHA   REV 1                                                    13-MAY-82   09:49:37
 COMPONENT REPORT                                                        PAGE        1
- - - - - - - - - - - - - - - - PIN - - NET - - NET - - - - - - - - -
          REF DES   DEVICE TYPE       NUMBER  NUMBER  NAME
- - - - - R1 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2       2
- - - - - R2 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2       3
- - - - - R3 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2       4
- - - - - R4 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2      11
- - - - - R5 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2      12
- - - - - R6 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2      13
- - - - - R7 - - RES1/4W - - - - - - 1 - - 1 - - - - - - - - - - - -
                                     2      14
- - - - - R8 - - RES1/4W - - - - - - 1 - - 15 - - - - - - - - - - - -
                                     2      16
- - - - - Z1 - - 4035 - - - - - - - 1 - - 19 - - - - - - - - - - - -
                                     2             ...unused
                                     3       3
                                     4       4
                                     5      23
                                     6       2
                                     7      11
                                     8      17
                                     9      16
                                    10      14
                                    11      13
                                    12      12
                                    13      22
                                    14      21
                                    15      20
                                    16      18
- - - - - Z3 - - 4035 - - - - - - - 1 - - 15 - - - - - - - - - - - -
                                     2      31
                                     3      24
                                     4      25
                                     5      23
                                     6      32
                                     7      26
                                     8      17
                                     9      30
                                    10      29
                                    11      28
                                    12      27
```

4. THE BILL OF MATERIALS REPORT

PURPOSE

The BOM-REPORT is a parts list.

HOW TO CREATE A BOM-REPORT

Use this command:

```
CREATE BOM-REPORT
```

EXAMPLE

```
ROGER  REV A                                              08-JAN-82  15:04:36

BOM REPORT                                                       PAGE      1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
DESCRIPTION                    REFERENCE DESIGNATORS                    TOTAL
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
35VT .01UF              C1  C2  C3                                          3
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4019                    Z3                                                  1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4035                    Z2                                                  1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4069                    Z4                                                  1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4081                    Z1  Z5                                              2
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
IOPIN                   J1                                                  1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
RPACK                   RP1                                                 1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

L O G I C A L   D E S I G N   C H E C K I N G

1. GENERAL INFORMATION

2. HOW TO USE THE LOGIC RUL CHK COMMAND

1. GENERAL INFORMATION

PURPOSE

The purpose of the logical design rules checking capability is to check the signal nets in a net data base for the following faults or omissions:

- o Nets with only one pin
- o Nets with one source pin
- o Nets with no load pin
- o Nets with two or more source pins
- o Nets with unspecified pins
- o Nets with non-connecting pins
- o Nets with no pins

WHEN TO USE LOGICAL DESIGN RULES CHECKING

Use logical design rules checking after you have created a net data base.

We recommend that you use this capability only after you have finished correcting all the errors shown in the EXTRACTION-LOG or the NET-LOAD-LOG.

2. HOW TO USE THE |LOGIC RULE CHK| COMMAND

|LOGIC RULE CHK| is a one-pick command. No other input is required.

PREREQUISITES

- o The net data base to be checked must include information from device description files.
- o The net data base to be checked must be in the current project file. If there is more than one revision in the current project file, the system will check the most recently used net data base revision.

HOW THE |LOGIC RULE CHK| COMMAND WORKS

When you pick the |LOGIC RULE CHK| command, the system does 2 things:

1. CHECKS THE NET DATA BASE FOR LOGICAL ERRORS:

   The system checks the number of pins on each net, and examines the pinuse code of each pin to determine whether there are any logical errors. (Pinuse codes - IN, OUT, BI, TRI, OCA, OCL, POWER, GROUND, NC- were specified by you in the device file or with the |ATTACH PINUSE| command on the schematic).

2. CREATES A TEXT FILE NAMED LDRC-LOG LISTING THE ERRORS IT HAS FOUND.

EXAMPLE:

```
LOGICAL DRC                                                         28-DEC-82 15:55:22

ERROR: Mixed source (out) pins on net.
       Net #:        1, Signal name: N3                        Contains:
           5 OUT Pins: U50.10              U49.9                      U48.6
       U45.6                  U44.6

ERROR: No load (in) pins on Net #:         1, Signal name: N3

ERROR: Mixed source (out) pins on net.
       Net #:        2, Signal name: A1                        Contains:
          41 OUT Pins: U9.1                U8.1                       U7.3
       U6.3                   U5.3                U42.9
       U41.8                  U40.6               U39.12
       U38.6                  U37.6               U36.6
       U35.6                  U34.6               U33.6
       U32.6                  U31.6               U30.3
       U3.3                   U29.3               U28.3
       U27.3                  U26.2               U25.2
       U24.2                  U23.2               U22.2
       U21.2                  U20.2               U2.3
       U19.2                  U18.2               U17.2
       U16.2                  U15.3               U14.3
       U13.1                  U12.1               U11.1
       U10.1                  U1.3

ERROR: No load (in) pins on Net #:         2, Signal name. A1

ERROR: No source (out) pins on Net #:         5, Signal name: B1
```

TYPES OF WARNINGS AND ERRORS SHOWN IN THE LDRC-LOG

Along with each warning and error message, the system lists the number and the name of of the net involved in the error; and (if appropriate) the reference designator and pin number of the pin involved in the error.

WARNING: PINUSE NOT SPECIFIED

The system cannot find a pinuse code for this pin.

WARNING: NON-CONNECT PIN FOUND

This net includes a pin with the pinuse code: NC (non-connecting)

ERROR: ONE PIN NET

This net has only one pin.

ERROR: NO LOAD (IN) PINS ON NET

This net does not have a pin with the pinuse code: IN.

ERROR: NO LOAD (IN OR BI) PINS ON NET

This net has a TRI pin but lacks a corresponding load pin (IN or BI).

ERROR: ONE SOURCE (BI) PIN ON NET

NO LOAD (IN or other BI) PINS.

This net has one BI but lacks a corresponding load pin (IN or BI).

ERROR: NO SOURCE (OUT) PINS

This net has no pins with the pinuse code: OUT, POWER, GROUND, OCA, OCL, TRI, or BI.

ERROR: MIXED SOURCE (OUT) PINS ON NET

This net has more than one source pin. In other words, the net has 2 pins or more labeled with one of these pinuse codes: POWER, GROUND, OUT, OCA, OCL, or TRI.

ERROR: NO PINS FOUND FOR NET

The system cannot find any pins for this net.

ERROR: NO NETS FOUND IN NET DATA BASE

The system cannot find any nets in the net data base.

REPEAT THE |LOGIC RULE CHK| COMMAND TO INSURE ACCURACY

Before going on the board design, correct errors shown in the LDRC-LOG. To be sure of accuracy, we recommend that you follow these steps:

STEP 1 Use the |LOGIC RULE CHK| command.

STEP 2 Print out the LDRC-LOG.

STEP 3 Reopen the schematic drawing(s) or the text netlist file and correct errors reported in the LDRC-LOG. If necessary, reopen device description files and correct errors.

STEP 4 Repeat the |EXTRACT NETLIST| or the |LOAD TXT NETLIST| command until the EXTRACTION-LOG or NET-LOAD-LOG is error-free.

STEP 5 Repeat STEPS 1-4 until the LDRC-LOG shows no errors.

PLACEMENT

- o PREPARATION
- o PLACEMENT - GENERAL INFORMATION
- o INTERACTIVE PLACEMENT
- o AUTOMATIC PLACEMENT
- o MANUAL PLACEMENT
- o THE RATSNEST

P R E P A R A T I O N

1. THE BOARD DRAWING
2. THE LAYERSTD FILE
3. NAMING THE LAYER STANDARD FILE ON THE TELESIS SYSTEM
4. CREATING THE LAYER STANDARD FILE
5. EDITING THE LAYER STANDARD FILE
6. LAYER STANDARD FILE NOT CREATED WITH THE $\boxed{\text{EDIT LAYERSTD}}$ LEAD-THRU

1. THE BOARD DRAWING

Once you have created a net data base, you may begin a board drawing. The system uses the information in the net data base to aid you as you work on the drawing.

Be sure to pick the |BOARD DRAWINGS| command rather than the |OTHER DRAWINGS| command when opening the drawing. Picking |BOARD DRAWINGS| instructs the system to link the drawing to the net data base in the current project file.

Do not begin a board drawing until you have thoroughly checked your net data base for accuracy.

2. THE LAYER STANDARD FILE

Before you begin to place or route your board, you must create a LAYERSTD (LAYER STANDARD) file for your particular design. It is a prerequisite for many system capabilities.

PURPOSE

The LAYER STANDARD file is a text file required for the design rule checking process, the artwork process, ratsnest, NC Drill and the auto router. It specifies the Telesis data base layers you are using for the physical layers of your board.

When you input pin description text files in the design process, you assign pad shapes and sizes to layers using physical layer nomenclature. To generate design rules checking and artwork, for example, the system must know which data base layer numbers correspond to these physical layer assignments. The LAYER STANDARD file is the means by which you give the system this information.

EXAMPLE: LAYERSTD FILE FORMAT

```
                    LAYERSTD 1
 (LAYERSTD)
 (TELESIS STANDARD LIBRARY)
 (PHYSICAL LAYER COMPONENT SIDE)
DBLAYER  1  COMPONENT-SIDE
 (PHYSICAL LAYER SOLDER SIDE)
DBLAYER  2  SOLDER-SIDE
 (PHYSICAL LAYER INTERNAL SIGNAL)
DBLAYER  3  INTERNAL-SIGNAL
 (PHYSICAL LAYER INTERNAL SIGNAL)
DBLAYER  4  INTERNAL-SIGNAL
 (PHYSICAL LAYER IMBEDDED VOLTAGE -V)
DBLAYER  5  IMBEDDED-PLANE                  -V
 (PHYSICAL LAYER IMBEDDED VOLTAGE +V)
DBLAYER  6  IMBEDDED-PLANE                  +V
DBLAYER 15  COMPONENT-SOLDER-MASK
DBLAYER 16  SOLDER-SOLDER-MASK
DBLAYER 33  CARD-OUTLINE
DBLAYER 34  PLATING-BAR
DBLAYER 36  DRAWING-FORMAT
DBLAYER 51  COMPONENT-OUTLINE
DBLAYER 54  REFERENCE-DESIGNATOR
DBLAYER 88  BOARD-DIMENSIONS
DBLAYER 89  SILKSCREENS
DBLAYER 90  TOOLING-CORNERS
DBLAYER 99  DRILL-CODE
END
```

3. NAMING THE LAYER STANDARD FILE ON THE TELESIS SYSTEM

When you create a LAYERSTD file, you must name it LAYERSTD. You may use the |EDIT LAYERSTD| lead-thru command to assist you in naming and creating the LAYERSTD text file. If you pick |ENTER| when the system prompts you for a revision label, it automatically names the file LAYERSTD.

By allowing the system to name the file, you need not input the text file name when editing or adding to the LAYERSTD file.

EXAMPLE: |EDIT LAYERSTD| (rev) ---> |ENTER|

When the system prompts you for a revision label (rev), you may pick |ENTER|. The system then looks for the latest version of the LAYERSTD file. The system looks for the text file LAYERSTD in the current project, then in the SYSTEM-LIBRARY.

If you create a new LAYERSTD text file for your particular project, the |SAVE FILE| command automatically assigns a revision label of "1".

However, you may assign a revision label of your choice. A revision label may consist of four alphanumeric characters. Input the revision if you are creating a new LAYERSTD file (if desired), or if you are editing a version of the file that was not the latest.

The LAYERSTD file must be in the current project or in the SYSTEM-LIBRARY. If you have the same layer standard for all your PC board designs, we recommend that you keep the LAYERSTD text file in the SYSTEM-LIBRARY.

NOTE: If you attempt to perform a design rules check on your PC board design without a LAYERSTD text file, for example, the system reports "NO LAYER STANDARD FILE". Furthermore, you will not be able to perform the design rules check on the board drawing since the LAYERSTD file is a prerequisite.

4. CREATING THE LAYER STANDARD FILE

The |EDIT LAYERSTD| lead-thru allows you to create and edit the LAYERSTD text file. It reduces the possibility of numerous format errors that could be generated when a text file is created on a text editor keyboard.

When you begin creating the LAYERSTD file for your particular PC board design, the Telesis system assists you in the following ways:

1. Automatically names the file LAYERSTD and prompts you for a revision (REV) label.

2. Automatically positions the cursor at the proper location.

3. Prompts you for input.

4. Automatically inputs keywords required in the file.

5. Automatically places parentheses ( ) around comment lines.

6. Issues error messages if your input is improperly formatted.

NOTE: When opening an existing LAYERSTD file, error messages may appear on the graphics screen if the file was not created with the |EDIT LAYERSTD| lead-thru. (See PLACE-14)

The following steps will lead you through the creation of the LAYERSTD file.

STEP 1 |TEXT LEADTHRU|

Pick the |TEXT LEADTHRU| command to enter the selected menu page containing options for creating and editing text files.

STEP 2 |EDIT LAYERSTD|

Pick the |EDIT LAYERSTD| command to begin creating the LAYERSTD file. When the system prompts you for a revision (rev), you may pick |ENTER|. The system automatically names the file LAYERSTD.

EXAMPLE: |EDIT LAYERSTD|---> Keyboard menu ----> |ENTER|

The system displays the LAYERSTD file on the graphics screen as a new file. The display will have the information shown below.

```
        LAYERSTD

|_|   (LAYERSTD)
      END
```

When you open a new LAYERSTD file, you may use the following commands to add information between (LAYERSTD) and the END statement:

|ADD DBLAYER|
|ADD COMMENT|

STEP 3 |ADD DBLAYER|

Pick the |ADD DBLAYER| command to enter the DBLAYER and the name of the physical board layer that you wish to assign.

Each assignment line begins with the keyword DBLAYER, followed by the layer number, followed by the name of the physical board layer, followed by a signal name (if there is one).

The function screen message line issues the following prompts:

(LAYER) ---> (LAYER NAME) ---> (SIGNAL NAME) ---> ENTER

EXAMPLE:

DBLAYER 6 IMBEDDED-PLANE +V

Words comprising the layer name must be separated by hyphens (-).

If the DBLAYER does not require a signal name, simply pick |ENTER| to bypass the signal name. If the DBLAYER requires a signal name, your input will appear at the right margin on the LAYERSTD file.

OPTIONS:

1. Usually, you list DBLAYER numbers in sequence. However, the order does not matter.

2. You may use the same layer name for more than one DBLAYER. The layer name must be less than or equal to 24 alphanumeric characters with no blank spaces and must be EXACTLY the same as used in the PIN DESCRIPTION file.

3. The signal-name may contain up to 18 alphanumeric characters with blank spaces allowed.

STEP 4 |ADD COMMENT| PHYSICAL LAYER COMPONENT SIDE |ENTER|

Use the |ADD COMMENT| command if you wish to add comments to the LAYERSTD file. The comment line is automatically placed within parentheses ( ). This identifies the line of text as a comment.

You may use the |ADD COMMENT| command to specifically identify the DBLAYER as a physical-layer.

EXAMPLE:

```
                LAYERSTD  1

 (LAYERSTD)
 (TELESIS STANDARD LIBRARY)
 (PHYSICAL LAYER COMPONENT SIDE)
DBLAYER  1   COMPONENT-SIDE
 (PHYSICAL LAYER SOLDER SIDE)
DBLAYER  2   SOLDER-SIDE
```

In this example, the comment identifies DBLAYER 1 as the PHYSICAL LAYER COMPONENT SIDE. Since the comment line is optional and is ignored by the system, you may place a comment above the line it is related to. The comment line provides the additional information needed for easy interpretation of the LAYERSTD file.

NOTE: You can input a maximum of 59 alphanumeric and blank spaces on a comment line. If your input exceeds 59 characters, the system interrupts your input and inserts the 59 character comment line on the file.

STEP 6 |DONE|

Pick |DONE| when you are finished with creating LAYERSTD file. To close the file, pick one of the following commands:

|SAVE FILE| - causes the system to save the active version of the file. The file is saved under the same revision label that was named when the file was opened.

If you created a new file, the system saves the file with a revision label of "1".

EXAMPLE:

Active file (new)	LAYERSTD
Saved file:	LAYERSTD 1

|SAVE FILE NEW RV| - allows you to input a new revision label for the active file. The new revision label may contain four alphanumeric characters.

The system saves the file that you opened under the new revision WITHOUT deleting or changing the original and its revision.

EXAMPLE:

Original version (old REV)	LAYERSTD 1
New version (new REV)	LAYERSTD 2

|CANCEL ACTV FILE| - causes the system to delete the active version of the file. The original version is not deleted; it keeps the same revision label.

|EDIT FILE| - Allows you to "menu-back" and return to the active file for further editing.

5. EDITING THE LAYERSTD FILE

As you create the LAYERSTD file, you may wish to make additions or corrections. You may also edit the file at a later time to update the file you previously created. When reopening a file, pick |ENTER| to display the latest version.

You may use the following commands to edit the LAYERSTD file:

|CHANGE DBLAYER|
|CHANGE COMMENT|
|DELETE LINE|
|UNDELETE LINE|

However, you may also use the ADD commands previously used in creating the file to make additions to it. All commands used to create and edit the file are located on the same menu page.

POSITIONING THE CURSOR

To edit a file you must always move the cursor to the proper position. The cursor always remains in the left margin of the file, and it must be positioned at or above the line to be edited.

- For ADD command during editing:

  Position the cursor to the line above the desired location of the new line. Lines positioned below the cursor automatically shift down after the new line is inserted.

  If the cursor is positioned at the END statement, the new line is inserted one line above.

- For CHANGE or DELETE commands:

  Move the cursor to the line to be changed or deleted.

Use these commands to move the cursor:

|LINE UP| - This move the cursor up one line.

|LINE DWN| - This moves the cursor down one line.

EXAMPLE: |LINE DWN| (Picked 4 times)

```
                                         LAYERSTD  1
Current
Cursor                  (LAYERSTD)
Position --->           (TELESIS STANDARD LIBRARY)
                        (PHYSICAL LAYER COMPONENT SIDE)
                        DBLAYER  1   COMPONENT-SIDE
New                     (PHYSICAL LAYER SOLDER SIDE)
Cursor   -->            DBLAYER  2   SOLDER-SIDE
Position                (PHYSICAL LAYER INTERNAL SIGNAL)
                        DBLAYER  3   INTERNAL-SIGNAL
```

When you pick |LINE DWN|, the cursor moves from the current position to the start of the next line of the LAYERSTD file. Each |LINE DWN| pick moves the cursor down one line.

If you command |LINE UP|, the cursor moves up one line with each |LINE UP| command pick.

IF THE FILE IS LONGER THAN SHOWN ON THE GRAPHICS SCREEN

If you add lines to the file after the graphics screen is full, the file scrolls up so that the bottom lines are visible and the top lines are not. Use the |LINE UP| command to bring the top lines back into view. Use the |LINE DWN| command to bring the bottom lines back into view.

TO CHANGE INFORMATION IN THE FILE

Use the CHANGE commands to change or update the LAYERSTD file. For example, you can use the |CHANGE DBLAYER| and the |CHANGE COMMENT| commands to update lines of text. You must move the cursor to the line you wish to change.

NOTE: When using the |CHANGE DBLAYER| command, you may pick |ENTER| for any input that is unchanged. For example, if you wish to change the LAYER without changing the LAYER NAME, pick |ENTER| when the system issues the LAYER NAME prompt. The layer name displayed on the file remains the same.

EXAMPLE: USING THE |CHANGE COMMENT| COMMAND

```
                                      LAYERSTD  1

Current          (LAYERSTD)
Cursor --->  |_| (TELESIS STANDARD LIBRARY)
Position         (PHYSICAL LAYER COMPONENT SIDE)
                 DBLAYER  1    COMPONENT-SIDE
                 (PHYSICAL LAYER SOLDER SIDE)
```

PICK: |CHANGE COMMENT|

INPUT: CLOCK GENERATOR BOARD |ENTER|

RESULT: The comment line (TELESIS STANDARD LIBRARY) is now changed to (CLOCK GENERATOR BOARD).

```
                                      LAYERSTD  1

Current          (LAYERSTD)
Cursor --->  |_| (CLOCK GENERATOR BOARD)
Position         (PHYSICAL LAYER COMPONENT SIDE)
                 DBLAYER  1    COMPONENT-SIDE
                 (PHYSICAL LAYER SOLDER SIDE)
```

TO DELETE INFORMATION FROM THE FILE

Use these commands to delete information from the file:

|DELETE LINE| - This command deletes one line from the file. You must move the cursor to the line you wish to delete.

|UNDELETE LINE| - This command reinserts the last line deleted. You may use this command to re-position the line to another location on the file. The system restores the line below the new cursor position regardless of the line's original placement in the file. If you position the cursor to the END statement the line is restored one line above.

NOTE: You cannot delete the lines labeled LAYERSTD and the END statement. You can only delete lines that you input when creating the file.

DELETING A LINE FROM THE FILE

EXAMPLE: Deleting the comment line (CLOCK GENERATOR BOARD).

```
                                     LAYERSTD  1

Current                (LAYERSTD)
Cursor  --->   |_|     (CLOCK GENERATOR BOARD)
Position               (PHYSICAL LAYER COMPONENT SIDE)
                       DBLAYER  1    COMPONENT-SIDE
                       (PHYSICAL LAYER SOLDER SIDE)
```

PICK: |DELETE LINE|

In the example, when you pick |DELETE LINE| the comment line labeled (CLOCK GENERATOR BOARD) is deleted from the file. The contents below the deleted line will shift up and close the open line.

RESULT:

```
                                     LAYERSTD  1

New                    (LAYERSTD)
Cursor  --->   |_|     (PHYSICAL LAYER COMPONENT SIDE)
Position               DBLAYER  1    COMPONENT-SIDE
                       (PHYSICAL LAYER SOLDER SIDE)
```

UNDELETING A LINE ON THE LAYERSTD FILE

You may now use the |UNDELETE LINE| command to reinsert the deleted line back to the file at your desired location. Use the |LINE UP| and the |LINE DWN| commands to position the cursor to a new location on the file before reinserting the line.

EXAMPLE: Undeleting the (CLOCK GENERATOR BOARD) comment line.

```
                  |                      LAYERSTD  1                        |
                  |                                                         |
Current --->      |  |‾|  (LAYERSTD)                                        |
Cursor            |       (PHYSICAL LAYER COMPONENT SIDE)                   |
Position          |       DBLAYER  1   COMPONENT-SIDE                       |
                  |       (PHYSICAL LAYER SOLDER SIDE)                      |
```

PICK: |UNDELETE LINE|

RESULT:

```
                  |                      LAYERSTD  1                        |
                  |                                                         |
New               |       (LAYERSTD)                                        |
Cursor   --->     |  |‾|  (CLOCK GENERATOR BOARD)                           |
Position          |       (PHYSICAL LAYER COMPONENT SIDE)                   |
                  |       DBLAYER  1   COMPONENT-SIDE                       |
                  |       (PHYSICAL LAYER SOLDER SIDE)                      |
```

In the example, the (CLOCK GENERATOR BOARD) comment line was reinserted back to its original location on the file. After reinserting the line with the UNDELETE command, the cursor automatically moves to that line of the file.

6. LAYER STANDARD FILES NOT CREATED WITH THE |EDIT LAYERSTD| LEAD-THRU

You may currently have LAYERSTD files on your system that were not created with the |EDIT LAYERSTD| lead-thru command. You may edit these files using the |EDIT LAYERSTD| command.

When you open one of these files with |EDIT LAYERSTD| command, the system checks it for formatting errors. If errors are present, the system places a message on the function screen telling you how many faulty lines exist in the file and it displays the file on the graphics screen with all faulty lines bracketed between parentheses and asterisks.

EXAMPLE: (*DBLYER 1 COMPONENT SIDE*)

The line is incorrect because the keyword DBLAYER is mispelled. There should also be a hyphen (-) between the words COMPONENT and SIDE.

The lines in parentheses are now comments, and the system treats them as such. You may delete these lines with the |DELETE LINE| command. Use the ADD commands to input the correct information.

If you are using the optional VT101 terminal to create and edit your LAYERSTD text file, refer to the example at the beginning of this section for general formatting requirements. You may later use the |EDIT LAYERSTD| command to display the file on the Telesis system and to check for formatting errors.

NOTE: The spacing requirements, keyword spelling, individual line formats and the maximum character inputs are shown and described in this section. However, refer to the COMMANDS section of the manual for additional information on each command used to create and edit the LAYERSTD file.

P L A C E M E N T

o G E N E R A L   I N F O R M A T I O N

1. PLACEMENT

GENERAL INFORMATION

Placement is the process of positioning symbols that are linked to the Net Data Base on a board drawing. Placement can be performed in four ways :

- Placement by Text File
- Manual Placement
- INTERACTIVE PLACEMENT
- AUTOMATIC PLACEMENT

WARNING

> If you are planning to use the SPLIT PLANE feature of the PHOTOPLOT utility, refer to POST PROCESSING, subsection 2.4, before placing components.

The operator can use manual, interactive, and automatic placement to place symbols on a board drawing. For example, manual placement can be used to place symbols that do not require the ASSIGN REF DES command (i.e., mounting hole, card outline). INTERACTIVE PLACEMENT can then be used to place all unplaced I/O components, then AUTOMATIC PLACEMENT for all unplaced IC and DISCRETE components in the Net Data Base.

The operator can automatically place components on a board drawing using the PLACE BY-TXT command. This command places components on a board drawing at the locations specified in a special text file named Place-Txt.

Manual placement is the process of adding symbols with one of the ADD SYMBOL commands, then linking the symbol to the Net Data Base with the ASSIGN REF DES command. When beginning the placement of a board, the ADD SYMBOL command can be used to place the card outline and mounting holes; these symbols do not require the ASSIGN REF DES command.

INTERACTIVE PLACEMENT, however, does not require the ADD SYMBOL or the ASSIGN REF DES command when components are placed. The link from the drawing to the Net Data Base is automatic. The system issues prompts on the function screen for each reference designator, device type, and symbol name of each unplaced component in the Net Data Base. When prompted for this information, the operator must select the location in the drawing for each component. When components are placed, the operator can selectively swap components, logic functions, and pins on the drawing for improved ratsnest connect lines and board routability.

The AUTOMATIC PLACEMENT feature controls the positioning of all unplaced IC and DISCRETE components in the Net Data Base. This feature operates according to parameters and restrictions specified by the operator. AUTOMATIC PLACEMENT can be interrupted at any time; the operator can then use interactive or manual placement to edit the positioning of components on the board drawing. In addition, the automatic swapping capability (ICs, functions, pins) can be used during or after automatic placement, if specified in the parameters.

The following sections describe the commands used to set up and use interactive, automatic, and manual placement when positioning components in a PC board drawing.

I N T E R A C T I V E   P L A C E M E N T

1. GENERAL INFORMATION
   - o PREREQUISITES FOR INTERACTIVE PLACEMENT
2. INTERACTIVE PLACEMENT COMMANDS
3. COMMANDS USED TO ADD COMPONENTS
4. COMMANDS USED TO EDIT PLACED COMPONENTS
5. COMMANDS USED TO CHECK PLACEMENT STATUS
6. THE COMPONENT, FUNCTION, AND PIN SWAPPING CAPABILITY
   - o INTERACTIVE SWAPPING COMMANDS

INTERACTIVE PLACEMENT

1. GENERAL INFORMATION

|INTERACTIVE PLACEMENT| allows placement of components on a board drawing without the |ADD SYMBOL| or |ASSIGN REF DES| commands. The system takes the component information from the net data base, then prompts the operator with each reference designator, device type and symbol name to be placed in the board drawing. The operator may then select the position on the drawing for each component.

When the |INTERACTIVE PLACEMENT| command is picked, the system displays the |INTERACTIVE PLACEMENT| menu. This menu contains the commands used to interactively add components to the drawing, as well as commands used for editing placed components. An additional menu page is available for interactive swapping of components, functions, and pins on the drawing.

PREREQUISTIES FOR |INTERACTIVE PLACEMENT|

1. The net data base must be created with device files, containing package information.
2. A board drawing (with a net data base) must be active.
3. There must be a symbol for every component to be placed on the board. These symbols must be in the current project, or in the SYSTEM-LIBRARY.

2. |INTERACTIVE PLACEMENT COMMANDS|

There are three types of interactive placement commands:

1. Commands used to ADD components.
2. Commands used to EDIT component placement.
3. Commands used to CHECK placement status.
4. Commands used to interactively swap components, functions, and pins.

3. COMMANDS USED TO ADD COMPONENTS

When adding components to the board drawing, the operator may use the following commands:

PLACE ICS	PLACE DISCRETES	PLACE I/OS
PLACE BY REFDES	PLACE BY DEVICE	PLACE ALL
SET PLACE ANGLE		

When the PLACE commands are used to place components on the board drawing, the system prompts the operator for the location of each UNPLACED component in the net data base, one at a time. The prompt displayed on the function screen message line shows the reference designator, device type, and symbol (package) name of the component.

The operator may respond to each prompt by selecting a location on the drawing for component placement, or simply pick |NEXT| to skip the component named in the prompt to advance to the next one.

The table below describes each ADD command used during |INTERACTIVE PLACEMENT.|

COMMAND	DEFINITION
PLACE ICS	To place IC components on the board drawing.
PLACE DISCRETES	To place DISCRETE components on the board drawing.
PLACE I/OS	To place I/O components on the board drawing.
PLACE BY DEVICE	To place components of a specified device type on the board drawing.
PLACE BY REFDES	To place a component on the the board drawing by naming its reference designator.
PLACE ALL	To place all UNPLACED components stored in the net data base.
SET PLACE ANGLE	To specify the placement angle of ALL components added to the board drawing during |INTERACTIVE PLACEMENT|. Only multiples of 90 are valid. (e.g. 0, 90, 180, 270)

Refer to the COMMANDS section of the manual for additional information on the input sequence for each command.

EXAMPLE: |PLACE ICS|

The function screen message line displays the first UNPLACED IC component stored in the net data base. The prompt displays the reference designator, device type, and symbol (package) name of the IC.

PLACE U11 / 7400 /DIP14

When the tablet menu is displayed, the operator may pick the location on the board for component placement. Additional picks will move the symbol to new placement locations.

EXAMPLE: |PLACE BY REFDES|

When |PLACE BY REFDES| is picked, the operator must input the reference designator of the component to be placed (reference designator for IC, DISCRETE, I/Os).

Keyboard input: U11 |ENTER|

The function screen message line then displays the reference designator, device type and symbol (package) name of the component.

PLACE U11 / 7400 / DIP14

When the tablet menu is displayed on the function screen, the operator may select the board location for this component. After the board location is selected, pick |ENTER| to place the component and to complete the |PLACE BY REFDES| command.

However, the operator may pick |NEXT| to continue the command, inputting the reference designator of the next component to be placed.

NOTE: An error message will appear on the function screen message line if the net data base does not contain a component for the reference designator named by by the operator.

EXAMPLE: "NO COMPONENT WITH REFDES U15"

4. COMMANDS USED TO EDIT PLACED COMPONENTS

Use the following commands to EDIT the placement of components during |INTERACTIVE PLACEMENT|.

DELETE COMPONENT :

Pick |DELETE COMPONENT| to delete placed components on the board drawing. Simply pick the components displayed on the board drawing to delete them.

Deleting a component from the drawing "returns" that component to the net data base as an UNPLACED component. The operator may use the PLACE commands to add the component back to the board drawing, if desired.

EXAMPLE: |DELETE COMPONENT| --> Tablet Menu --> (P) |ENTER|

|MOVE COMPONENT| :

Pick the |MOVE COMPONENT| command to change the location of any components displayed on the board drawing.

EXAMPLE: |MOVE COMPONENT| --> Tablet Menu --> (P1) (P2)

(P1) - defines the component to be moved. The symbol attaches to the light pen cursor, with ratsnest lines, if present, rubberbanding to the symbol.

(P2) - defines the new location of the component.

|ROTATE COMPONENT| :

Pick |ROTATE COMPONENT| to change the angle of any component on the board drawing.

EXAMPLE: |ROTATE COMPONENT| --> Keyboard Input --> |ENTER| (P) |ENTER|

The keyboard input must be a multiple of 90 to be valid (e.g. 0, 90, 180, 270).

(P) - defines the component to be rotated. Additional picks will rotate additional components at the same angle.

5. COMMANDS USED TO CHECK PLACEMENT STATUS

|PLACED STATUS| :

|PLACED STATUS| is a one-pick command that lists the components placed on the board drawing. The component list is displayed on the function screen.

The list displays the reference designator, device type, symbol (package) name, board location (X,Y coordinates), and rotation for each PLACED component on the board drawing.

EXAMPLE: |PLACED STATUS|

Typical line display:

```
U11  /  7400  / DIP14   @( 6000,   5525)  rot=0
```

|UNPLACED STATUS| :

Use the |UNPLACED STATUS| command to list the UNPLACED components in the net data base.

The reference designator, device type and symbol (package) name are listed for each component.

EXAMPLE: |UNPLACED STATUS|

Typical line display:

```
U12  /  7410  / DIP14
```

6. THE COMPONENT, FUNCTION, AND PIN SWAPPING CAPABILITY

After placement of components on the board drawing, the operator may pick the |COMP/FUNC/PIN SWAPPING| command to display the interactive swapping menu. This menu contains commands which allow the operator to selectively swap components, functions, and pins on the board drawing. The operator may use the interactive swapping capability to shorten and straighten ratsnest connect-lines on the drawing, and to improve board layout for increased auto-routing completion.

The commands on the interactive swapping menu are listed below:

|SWAP COMPONENTS|	|SWAP COMPONENTS BY REFDES|
|SWAP FUNCTIONS|	|SWAP FUNCTIONS BY REFDES.PIN#|
|SWAP PINS|	|HILITE SWAPPABLE FUNCTIONS|
	|HILITE SWAPPABLE PINS|

Refer to COMMANDS section of the manual for the exact input sequence for each of the above commands.

INTERACTIVE SWAPPING COMMANDS

Regardless of the method used to place board components, the operator may interactively swap components, functions, and pins on the active drawing. During interactive swapping, the system reconstructs existing ratsnest connect-lines on the drawing, maintaining the correct net information. Each interactive swapping command is described below.

|SWAP COMPONENTS|

Use the |SWAP COMPONENTS| command to swap components on the active drawing.

EXAMPLE: |SWAP COMPONENTS| -Tablet-Menu (P1) (P2) |ENTER|

- o (P1) - defines and highlights the first component to be swapped.
- o (P2) - defines and highlights the second component to be swapped.
- o When |ENTER| is picked, the system dehighlights each component, then swaps the components. The system reconstructs the ratsnest connect-lines on the drawing, maintaining the correct net information for the swapped components.

- Swapped components are placed at the symbol origin locations maintained by each component prior to swapping. For example, if swapping an IC component with a DISCRETE component, the system places the IC symbol origin at the origin location previously held by the DISCRETE, and the reverse.

Use the |SWAP COMPONENTS BY REFDES| command to swap components on the active drawing by specifying the reference designator of each component to be swapped. Refer to the COMMANDS section of the manual for the exact input sequence of the |SWAP COMPONENTS BY REFDES| command.

|SWAP FUNCTIONS|

Use the |SWAP FUNCTIONS| command to selectively swap identical IC component logic functions within a device, or with other devices containing the same functions (i.e. multiple instances of the device 7400). That is, logic functions may be swapped between components of the same device type.

EXAMPLE: |SWAP FUNCTIONS| -Tablet-Menu (P1)(P2) |ENTER|

- (P1) - defines the first function to be swapped. Select one logic function pin on the component to highlight all pins associated with the function.

  If the pin selected is not associated with a swappable function, the function screen message line displays:

  -SELECTED POINT IS NOT A SWAPPABLE FUNCTION-

- (P2) - defines the second function to be swapped. Select one logic function pin to highlight the pins contained in that function. If the pin selected is not associated with a function that is swappable with the first function, the function screen message line displays:

  -SELECTED FUNCTION NOT SWAPPABLE WITH FIRST FUNCTION-

- Pick |ENTER| to swap the functions selected.

- During swapping, the system reconstructs ratsnest connect-lines, maintaining the correct net information for each function.

Use the |SWAP FUNCTIONS BY REFDES.PIN#| command to swap IC component logic functions on the active drawing by specifying the component reference designators, and pin numbers associated with the functions to be swapped. Refer to the COMMANDS section of the manual for the exact input sequence required for the |SWAP FUNCTIONS BY REFDES.PIN#| command.

Use the |HILITE SWAPPABLE FUNCTIONS| command to highlight swappable IC component logic functions on the graphics screen. This command may be used to view swappable functions prior to using the |SWAP FUNCTIONS|, or |SWAP FUNCTIONS BY REFDES.PIN#| command. Refer to the COMMANDS section of the manual for the required input sequence.

|SWAP PINS|

Use the |SWAP PINS| command to interactively swap pins within an IC component logic function. The swappable pins on a logic function are those defined in the device description file (PINSWAP) for the component.

EXAMPLE: |SWAP PINS| -Tablet-Menu (P1) (P2) |ENTER|

- (P1) - defines the first swappable pin within the logic function. The selected pin highlights on the component.
- (P2) - defines the second swappable pin within the logic function. The selected pin highlights on the component.
- If a pin selected is not a swappable pin, the function screen message line displays:

  -SELECTED POINT IS NOT A SWAPPABLE PIN-
- Pick |ENTER| to swap the selected pins. During swapping, the system reconstructs ratsnest connect-lines, maintaining the correct net information for the logic function.

Use the |HILITE SWAPPABLE PINS| commands to highlight all swappable pins associated with a single IC component logic function on the active drawing. This command may be used to view swappable pins prior to using the |SWAP PINS| command. Refer to COMMANDS section of the manual for the required input sequence of the |HILITE SWAPPABLE PINS| command.

AUTOMATIC PLACEMENT AND IMPROVEMENT

1. THE AUTOMATIC PLACEMENT CAPABILITY AND HOW TO USE IT
   - o THE AUTOMATIC PLACEMENT AND IMPROVEMENT STRATEGY
   - o AUTOMATICALLY PLACING THE COMPONENTS ON THE DRAWING
2. AUTOMATIC PLACEMENT PREREQUISITES AND SUMMARY OF STEPS
   - o PREPARING THE BOARD FOR AUTOMATIC PLACEMENT
3. USING THE AUTOMATIC PLACEMENT COMMAND
4. SETTING THE AUTOMATIC PLACEMENT PARAMETERS
   - o NOTE ON SETTING THE SWAPPING PARAMETERS PRIOR TO COMPONENT PLACEMENT
   - o THE AUTOMATIC PLACEMENT FIX AND FREE COMMANDS
   - o AUTOMATIC PLACEMENT PARAMETERS
5. THE AUTO-PLACE BOARD COMMAND AND THE PLACE-LOG TEXT FILE

1. THE AUTOMATIC PLACEMENT CAPABILITY AND HOW TO USE IT

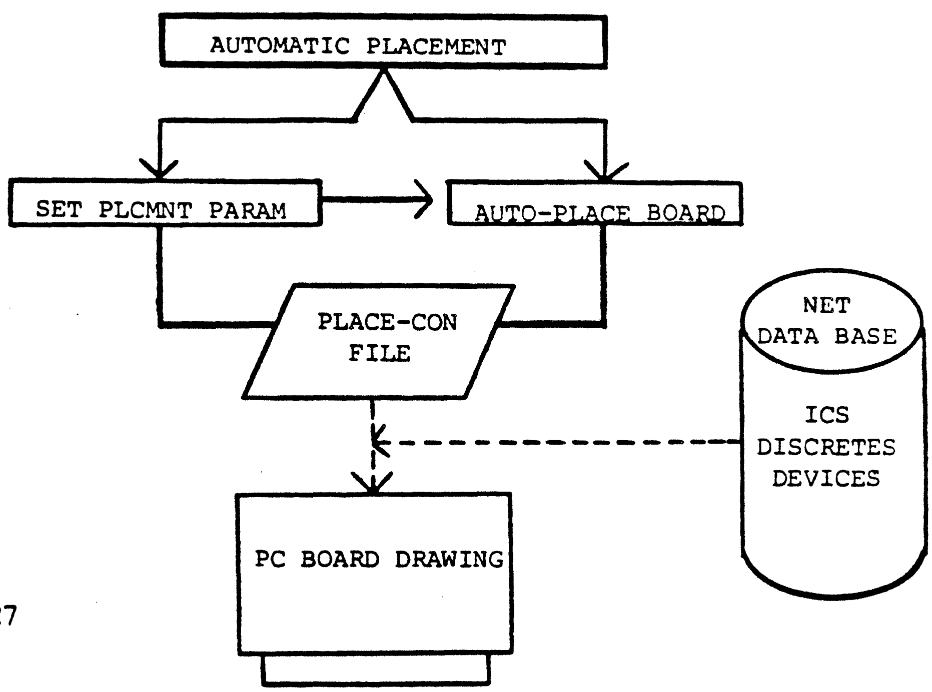
The automatic placement capability automatically positions all unplaced IC and DISCRETE components on the active board drawing. This capability operates according to the parameters and restrictions specified by the operator. Automatic placement may be interrupted at any time; the operator may then interactively edit the placement of existing components on the board drawing.

Like |INTERACTIVE PLACEMENT|, |AUTOMATIC PLACEMENT| does not require the |ADD SYMBOL| and |ASSIGN REF DES| commands.

During |AUTOMATIC PLACEMENT| the system scans the net data base for IC and DISCRETE component information. The system uses this information with the operator-defined parameters to efficiently place the components on the board. The |AUTOMATIC PLACEMENT| parameters may be set to allow placement of all IC and DISCRETE components in the net data base. If the operator chooses not to set the placement parameters, the system uses default values during execution of the |AUTO-PLACE BOARD| command.

When the placement parameters have been set, the system stores this information in a control file named PLACE-CON. The system uses this text file to control the automatic placement of components on the board.

The diagram below illustrates the |AUTOMATIC PLACEMENT| capability:



THE AUTOMATIC PLACEMENT AND IMPROVEMENT STRATEGY

Although the operator may choose many strategies for automatically placing a board, it is recommended that the |AUTOMATIC PLACEMENT| parameters be set so that the system first places ICs, DISCRETES and DEVICES. When the system completes the automatic placement of components, with the system executing the |RATSNEST| command, the operator may then use the commands on the |INTERACTIVE PLACEMENT| menu to edit the placed components on the board.

For example, the |MOVE COMPONENT| command may be used to interactively and selectively reposition components. In addition, the SWAP commands allow the operator to swap placed components, IC logic functions, and pins for improvement of ratsnest connect-lines.

When interactive editing is complete, the operator may proceed back to the |AUTOMATIC PLACEMENT| menus to FIX those components, logic functions, and pins previously edited. Fixed components, logic functions, and pins are those that are made unavailable for automatic swapping. For example, the operator may wish to fix selected components that are currently placed on the board at the desired locations. Use the available LIST and HILITE commands on the |AUTOMATIC PLACEMENT| menus to selectively show and verify the currently FIXED components, IC logic functions, and pins.

The FREE commands, however, may be used to free components, logic functions, and pins that were previously FIXED with the available interactive commands on the |AUTOMATIC PLACEMENT| menus. In addition, fixing of logic functions may have occurred when the net data base was originally created with the |EXTRACT NETLIST| or |LOAD TXT NETLIST| command. That is, pre-assigned logic functions on the schematic will be fixed and unavailable for automatic swapping on the board until the operator selectively frees a fixed function. In the text netlist, fixed functions are those pre-assigned by the information under $PACKAGES, with the $NETS section of file containing the reference designator and pin number information for the defined PACKAGES. Use the available LIST and HILITE commands to selectively show the currently FREE components, IC logic functions, and pins.

When the operator completes selective fixing and freeing on the placed board, the |AUTOMATIC PLACEMENT| parameters may then be reset to allow automatic swapping of FREE components, logic functions and pins. During |AUTO-PLACE BOARD| the command, with "yes" specified for each swapping parameter (|RUN IC SWAP|, |RUN FUNCTION SWAP| and |RUN PIN SWAP)|, the system swaps FREE components, IC logic functions, and pins. During swapping, ratsnest connect-lines reconstruct on the drawing, maintaining the correct net information. Swapping simply attempts to improve the board, creating shorter, straight horizontal/straight vertical ratsnest connect-lines, preparing the board for increased automatic routing completion percentage.

The following sections describe the |AUTOMATIC PLACEMENT| capability, using the strategy defined here.

AUTOMATICALLY PLACING THE COMPONENTS ON THE DRAWING

There are three ways to automatically place components on the active board drawing:

1. |RUN IC PLACEMENT|
2. |RUN DEVICE PLACEMENT|
3. |RUN DISCRETE PLACEMENT|

These commands are |AUTOMATIC PLACEMENT| parameters. The operator may specify "yes" for each of the above parameters to place ICs, DISCRETES, and DEVICES. When the |AUTO-PLACE BOARD| command is picked, the system scans the net data base, then proceeds to place the components according to the operator-defined parameters, or with the system default parameters. (See SETTING THE PLACEMENT PARAMETERS).

NOTE: The system requires one pre-placed component on board prior to using the |AUTO-PLACE BOARD| command. This pre-placed component may be placed using |INTERACTIVE PLACEMENT|.

In addition, |AUTOMATIC PLACEMENT| does not place I/O components on the board. If an I/O component exists in the net data base, the operator must interactively place the I/O. This also satisfies the single pre-placed component requirement.

2. |AUTOMATIC PLACEMENT| PREREQUISITES AND SUMMARY OF STEPS

The |AUTOMATIC PLACEMENT| prerequisites and summary of steps are outlined below. This information is described in greater detail in the sections that follow.

PREREQUISITES

1. A board drawing with a net data base must be active.

2. The net data base must be created with device description files present in the current project, or in the SYSTEM-LIBRARY.

   The automatic placement capability requires the PACKAGE (component symbol name belonging to each device type) and CLASS (I/O, IC or DISCRETE) information provided by device files.

3. There must be a symbol for every component to be used on the board. These symbols must be in the current project, or in the SYSTEM-LIBRARY. However, a pre-placed component on the board does not require a symbol in the current project, or in the SYSTEM-LIBRARY during |AUTOMATIC PLACEMENT|.

SUMMARY OF STEPS

STEP 1. Place rectangles on specific data base layers to indicate the following:

A. Placement keep-in area (one rectangle on data base layer 147). - REQUIRED

B. Placement keep-out areas of the board. -OPTIONAL

C. Component space rectangles. - OPTIONAL

STEP 2. |INTERACTIVE PLACEMENT|

Use the |INTERACTIVE PLACEMENT| capability to pre-place one component in the net data base. If I/O components exist in the net data base, place the I/O components on the board. This also satisfies the single pre-placed component requirement.

STEP 3. |AUTOMATIC PLACEMENT|

Pick the |AUTOMATIC PLACEMENT| command, then the |SET PLCMNT PARAM| command to open the menu set used to input the placement parameters.

STEP 4. |AUTO-PLACE BOARD|

Pick the |AUTO-PLACE BOARD| command to start automatic placement of components (IC, DISCRETE, DEVICE) on the board. After components have been placed, the system automatically executes the |RATSNEST| command.

STEP 5. |INTERACTIVE PLACEMENT| (OPTIONAL)

Use the |INTERACTIVE PLACEMENT| capability to edit the positioning of components on the board. The SWAP commands may also be used to swap placed components, IC logic functions, and pins for improvement of ratsnest connect-lines.

STEP 6. |AUTOMATIC PLACEMENT|

After interactive editing, the operator may proceed to the |AUTOMATIC PLACEMENT| menus to FIX those components, IC logic functions, and pins previously edited during STEP 5. For example, the operator may wish to fix selected components that are currently placed on the board at the desired locations. Use the available LIST and HILITE commands to selectively show and verify the currently FIXED components, IC logic functions, and pins. Use the FREE commands to free selected components, logic functions, and pins previously fixed interactively, or to free logic functions that were fixed (pre-assigned) on the schematic or text-netlist prior to creation of the net data base.

STEP 7. |SET PLCMNT PARAM|

When the operator completes selective fixing and freeing, the automatic placement parameters may be reset to allow automatic swapping of FREE components, IC logic functions, and pins. Specify "yes" for each swapping parameter: |RUN IC SWAP|, |RUN FUNCTION SWAP|, |RUN PIN SWAP|.

STEP 8. |AUTO-PLACE BOARD|

During the |AUTO-PLACE BOARD| command, with "yes" specified for each swapping parameter, the system swaps FREE components, logic functions, and pins on the placed board drawing. During swapping, ratsnest connect-lines reconstruct on the drawing, maintaining the correct net information. The system attempts to improve the board, creating shorter, straight horizontal/straight vertical ratsnest lines, preparing the board for increased automatic routing completion.

PREPARING THE BOARD FOR |AUTOMATIC PLACEMENT|

To prepare the board for |AUTOMATIC PLACEMENT|, the operator must place rectangles on specific data base layers to indicate the placement KEEP-IN area, optional KEEP-OUT areas, and component space requirements.

Use the |ADD RECTANGLE| command to place rectangles. Do not use the |ADD LINE| command; the operator cannot always be ensured of a perfect rectangle with |ADD LINE|.

KEEP-IN AREA: (required)	Place a rectangle on layer 147 to indicate the overall area of the board for component placement.

KEEP-OUT AREA: (optional)	Place rectangles on the board to indicate areas prohibiting component placement. The operator may use any unreserved layer for these rectangles. (See BASIC-30; LAYER RECOMMENDATIONS) Use the |SET PLCMNT PARAM| command to specify the layers where keep-out rectangles exist. Up to 10 data base layers may be specified.

COMPONENT SPACE REQUIREMENT: (optional)	A rectangle can be added to each package symbol to indicate the space requirements for the symbol when placed on the board. A component space rectangle must be added to the symbol drawing prior to using the |CREATE SYMBOL| command. The rectangle may be placed on any unreserved layer. Use the |COMPONENT OUTLINE LAYER| parameter to specify the data base layer where these rectangles exist. If a layer is not specified with this parameter, the system searches for the rectangles on data base layer 154 (default) during automatic placement. If rectangles are not found, the system determines the minimum space requirement for each package symbol to be placed by examining the graphics within the symbol, except text.

3. USING THE |AUTOMATIC PLACEMENT COMMAND|

When the |AUTOMATIC PLACEMENT| command is picked, the function screen message line displays the following information:

-LOOKING FOR KEEP-IN RECTANGLE:
MISSING KEEP-IN RECTANGLE ON LAYER 147-

The operator must place a rectangle on layer 147 to specify the KEEP-IN area on the board for automatic placement of components in the net data base. However, if a rectangle was added to layer 147 prior to picking the |AUTOMATIC PLACEMENT| command, the function screen message line displays:

-LOOKING FOR KEEP-IN RECTANGLE:-

When the system locates the KEEP-IN rectangle on layer 147, the |AUTOMATIC PLACEMENT| menu is displayed on the function screen:

|SET PLCMNT PARAM|	|AUTO-PLACE BOARD|
|FIX/FREE COMPONENTS|	|FIX/FREE FUNCTIONS|
|FIX/FREE PINS|	|FIX ALL|
|FREE ALL|	|RATSNEST|
|LIST PLACE-LOG|	|PRINT PLACE-LOG|
|CURRENT INDEX|	|PROJECT INDEX|

4. SETTING THE |AUTOMATIC PLACEMENT| PARAMETERS

Use the |SET PLCMNT PARAM | command to set the parameters to be used during exection of the |AUTO-PLACE BOARD| command. Parameters are simply tools which allow the operator to achieve optimum placement on each unique board design.

For example, when setting the |IC GRID| parameter, the operator must define a grid that accomodates all unplaced IC components in the net data base. This parameter, if properly set, specifies the grid points on the drawing for each IC to be placed. During |AUTO-PLACE BOARD|, the system places the origin of each IC at these grid points.

The commands below are |AUTOMATIC PLACEMENT| parameters:

|RUN IC PLACEMENT|
|RUN FUNCTION SWAP|
|RUN IC SWAP|
|IC ROTATION |
|RUN DEVICE PLACEMENT|
|DEVICE NAMES|
|DISCRETE GRID|
|RUN PIN SWAP|
|I/O WEIGHT|
|COMPONENT OUTLINE LAYER|

|IC GRID|
|MAX FUNCTION SWAP TIME|
|MAX IC SWAP TIME|
|DEVICE ROTATION|
|DEVICE GRID|
|RUN DISCRETE PLACEMENT|
|DISCRETE ROTATION|
|POSITION WEIGHTS|
|KEEPOUT LAYERS|

There are default values for each parameter. The operator must determine if the default value for any of the above parameters is acceptable for automatic placement on a unique board. This section defines the input sequence for each parameter and the system default value.

NOTE ON SETTING THE SWAPPING PARAMETERS PRIOR TO COMPONENT PLACEMENT

When setting the |AUTOMATIC PLACEMENT| parameters, the operator may specify "yes" for each swapping parameter. During component placement, the system will swap IC components, swappable IC logic functions, and swappable pins prior to executing the |RATSNEST| command, completing the |AUTO-PLACE BOARD| command. The following commands are swapping parameters:

|RUN IC SWAP|　　|RUN FUNCTION SWAP|　　|RUN PIN SWAP|

However, it is recommended that the operator first execute the the |AUTO-PLACE BOARD| without running IC, function and pin swapping. This will allow interactive editing of placed components, logic functions and pins. The FIX and FREE commands available on the |AUTOMATIC PLACEMENT| menu may then be used to FIX and/or FREE edited components, functions and pins prior to re-executing the |AUTO-PLACE BOARD| command to allow swapping of FREE ICs, logic functions, and pins, with "yes" specified for the above parameters.

If the operator chooses to automatically place the board with swapping specified ("yes") in the parameters, DO NOT use the |FIX ALL| commands on the |AUTOMATIC PLACMENT| menu prior to executing the |AUTO-PLACE BOARD| command. Fixing all components, swappable logic functions, and swappable pins will cause the |AUTO-PLACE BOARD| execution to stop, with incomplete board placement. A further discussion of the FIX and FREE commands is presented below.

THE |AUTOMATIC PLACEMENT| FIX AND FREE COMMANDS

The FIX and FREE commands are designed to be used on the placed board drawing. The FIX commands allow the operator to selectively FIX those FREE components, swappable logic functions, and swappable pins so that these do not swap during the execution of the |AUTO-PLACE BOARD| on the placed board.

The FREE commands, however, may be used to FREE components, swappable logic functions, and swappable pins that were previously FIXED. In addition, fixing of logic functions may have occurred when the net data base was originally created with the |EXTRACT NETLIST|, or |LOAD TXT NETLIST| command. That is, pre-assigned logic functions (pin numbers annotated) on the schematic will be fixed and unavailable for automatic swapping until the operator selectively frees a fixed function. In the text netlist, fixed functions are those pre-assigned by the information under $PACKAGES, with the $NETS section of the file containing the reference designator and pin number information for the defined PACKAGES. Free logic functions, however, are those defined by the $FUNCTIONS section, with the $NETS section of the file listing the function designators and pin names.
Use the HILITE and LIST commands to selectively show the currently fixed and free components, swappable logic functions, and swappable pins.

The FIX and FREE commands are listed below:

|FIX ALL| |FREE ALL|

|FIX/FREE COMPONENTS|

HILITE FIXED COMPONENTS	HILITE FREE COMPONENTS
FIX ALL COMPONENTS	FREE ALL COMPONENTS
FIX COMPONENT [SELECT]	FREE COMPONENT [SELECT]
LIST FIXED COMPONENTS	LIST FREE COMPONENTS
FIX COMPONENTS BY WINDOW	FREE COMPONENTS BY WINDOW
FIX COMPONENT BY REFDES	FREE COMPONENT BY REFDES

|FIX/FREE FUNCTIONS|

|HILITE FIXED FUNCTIONS|	|HILITE FREE FUNCTIONS|
|FIX ALL FUNCTIONS|	|FREE ALL FUNCTIONS|
|FIX FUNCTION [SELECT PIN]|	|FREE FUNCTON [SELECT PIN]|
|FIX FUNCTIONS [SELECT COMP]|	|FUNCTIONS [SELECT COMP]|
|LIST FIXED FUNCTIONS|	|LIST FREE FUNCTIONS|
|FIX FUNCTIONS BY WINDOW|	|FREE FUNCTIONS BY WINDOW|
|FIX FUNCTIONS BY REFDES.PIN#|	|FREE FUNCTION BY REFDES.PIN#|
|FIX BY DEVICE|	|FREE BY DEVICE|

|HILITE FIXED PINS ON COMP|	|HILITE FREE PINS ON COMP|
|FIX ALL PINS|	|FREE ALL PINS|
|FIX PIN [SELECT PIN]|	|FREE PIN [SELECT PIN]|
|FIX ALL PINS IN COMPONENT|	|FREE ALL PINS IN COMPONENT|
|FIX PINS BY WINDOW|	|FREE PINS BY WINDOW|
|FIX PINS BY REFDES.PIN#|	|FREE PIN BY REFDES.PIN#|
|FIX ALL PINS IN FUNCTION|	|FREE ALL PINS IN FUNCTION|

The COMMANDS section of the manual describes the use and input sequence of each FIX/FREE command. Refer to the COMMANDS section when using these commands.

|AUTOMATIC PLACEMENT| PARAMETERS

|RUN IC PLACEMENT|

Use the |RUN IC PLACEMENT| command to specify automatic placement of all unplaced ICs in the net data base.

|RUN IC PLACEMENT|	(RUN IC PLACEMENT [Y/N]	-Keyboard- Input	|ENTER|

DEFAULT: Y (yes)

|RUN DEVICE PLACEMENT|

Use the |RUN DEVICE PLACEMENT| parameter to specify automatic placement of all unplaced DEVICES in the net data base. This is a special mode for placing all ICs or DISCRETES of the same device type on a unique grid. This is commonly used for decoupling capacitors. See |DEVICE GRID| and |DEVICE NAMES|.

INPUT SEQUENCE:

|RUN DEVICE PLACEMENT| (RUN DEVICE PLACEMENT [Y/N] -Keyboard- |ENTER|
Input

DEFAULT: N (no)

|RUN DISCRETE PLACEMENT|

Use the |RUN DISCRETE PLACEMENT| command to specify automatic placement of all unplaced DISCRETE components in the net data base.

INPUT SEQUENCE:

|RUN DISCRETE PLACEMENT| (RUN DISCRETE PLACEMENT [Y/N])-Keyboard- |ENTER|
Input

DEFAULT: N (no)

|POSITION WEIGHTS|

Use |POSITION WEIGHTS| to specify the preferred positioning of components to be placed on the board. The command prompts the operator for horizontal and vertical weight values. Values must be numbers between 0 and 10, with decimal inputs allowed. (e.g. 1.5)

EXAMPLE:	(HORIZONTAL WEIGHT)	2
	(VERTICAL WEIGHT)	1

In the example, optimization of the horizontal routing channels is more desirable than the vertical channels based on the 2:1 ratio. The input ratio is significant; the actual numbers are not.

The operator-defined position weights should depend on the following:

1. The overall shape of the board.

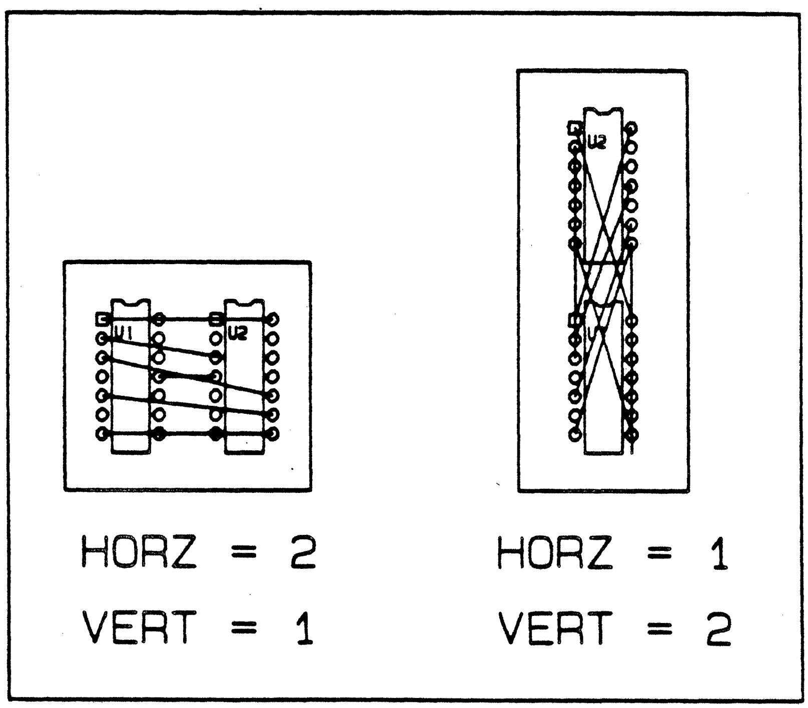
2. The rotation of components when placed.

INPUT SEQUENCE:

|POSITION WEIGHTS|	(HORIZONAL WEIGHT)	-Keyboard- Input	|ENTER|
	(VERTICAL WEIGHT)	-Keyboard- Input	|ENTER|

DEFAULT: HORIZONTAL: 1.0 VERTICAL: 1.0

EXAMPLE:



|IC GRID|

Use the |IC GRID| command to define a matrix of valid locations for placement of IC components.

The IC grid is a series of horizontal and vertical lines, displayed on layer 147. The matrix points formed by the intersecting lines are possible locations for the automatic placement of ICs. When a component is placed at one of these points, the symbol is placed with its origin at that point.

The operator should define a grid that accommodates the number of ICs to be placed on the board. For example, if the net data base contains ten IC components, the grid must be defined so that ten matrix points are available with areas large enough for the extents of each IC symbol to be placed.

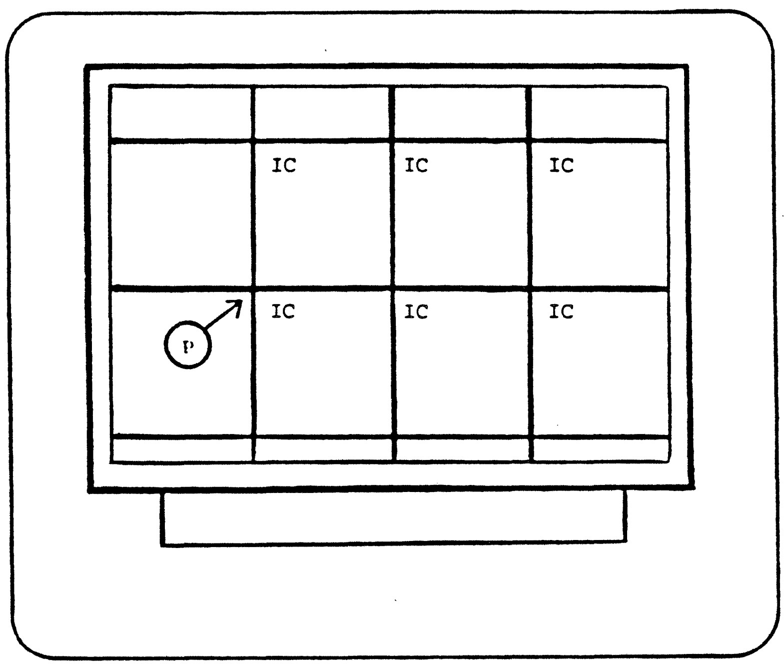
INPUT SEQUENCE:

|IC GRID| (IC GRID SIZE X) -Keyboard- Input |ENTER|

(IC GRID SIZE Y) -Keyboard- Input |ENTER| (P) |ENTER|

* (P) is the origin of the grid.

EXAMPLE: Matrix points to allow placement of six ICs.



(P) - defines the grid origin and displays the matrix points on the graphics screen.

NOTE: DO NOT define a grid size smaller than the space requirements for the smallest IC to be placed. Although |AUTOMATIC PLACEMENT| never overlaps components, small grid sizes will cause this feature to operate slower.

After the grid points are displayed on the graphics screen, the operator may respond with one of the following:

P - defines a new grid origin if the first grid display is unacceptable. The existing grid is erased and the new grid is drawn on the graphics screen.

The operater may also use the keypad to define explicit (X,Y) coordinates to define a new grid origin.

|ENTER| - This indicates that the grid is accepted, with the system saving the grid information. The grid displayed on the graphics screen is erased.

|CANCEL| - This causes the grid information to be ignored; the system erases the grid display.

DEFAULT: X 700 , Y 1400 Origin: 0,0

|DEVICE GRID|

Use the |DEVICE GRID| command to define a matrix of valid locations for DEVICES. The input sequence is the same as |IC GRID|.

The |DEVICE GRID| allows placement of all ICs or DISCRETES of the same device type on a unique grid. (See NOTE below.)

Do not define a grid size smaller than the space requirements for the smallest DEVICE to be placed. The grid must also accomodate the number of DEVICES to be placed on the board.

INPUT SEQUENCE:

|DEVICE GRID|	(DEVICE GRID SIZE X)	-Keyboard- Input	|ENTER|		
	(DEVICE GRID SIZE Y)	-Keyboard- Input	|ENTER|	P	|ENTER|

* P is the origin of the grid.

DEFAULT: X 700, Y 1400 Origin 0, 200

NOTE: The system default origin offsets the grid 200 mils in the Y-direction to allow placement of multiple "like-devices" near the components they are connected to. For example, bypass capacitors may be automatically placed near ICs, since bypass capacitors are connected to the power and ground pins of the IC. When the |IC GRID| and the |DEVICE GRID| parameters are set to the same X,Y coordinates (with the IC grid origin 0,0) bypass capacitors will be placed near the IC, within the 200-mil offset grid.

For example, if the IC GRID and the |DISCRETE GRID| parameters are set to the same coordinates, with the |IC GRID| origin at 0,400, the |DEVICE GRID| origin must then be set to 0,600 to establish the 200-mil offset in the Y-direction.

Use the |DEVICE NAMES| command to name the DEVICES (up to 5) to be placed on the DEVICE grid.

EXAMPLE: Using the |DEVICE GRID| parameter to create an offset grid for the automatic placement of capacitors near IC components.

CURRENT |IC GRID| PARAMETERS

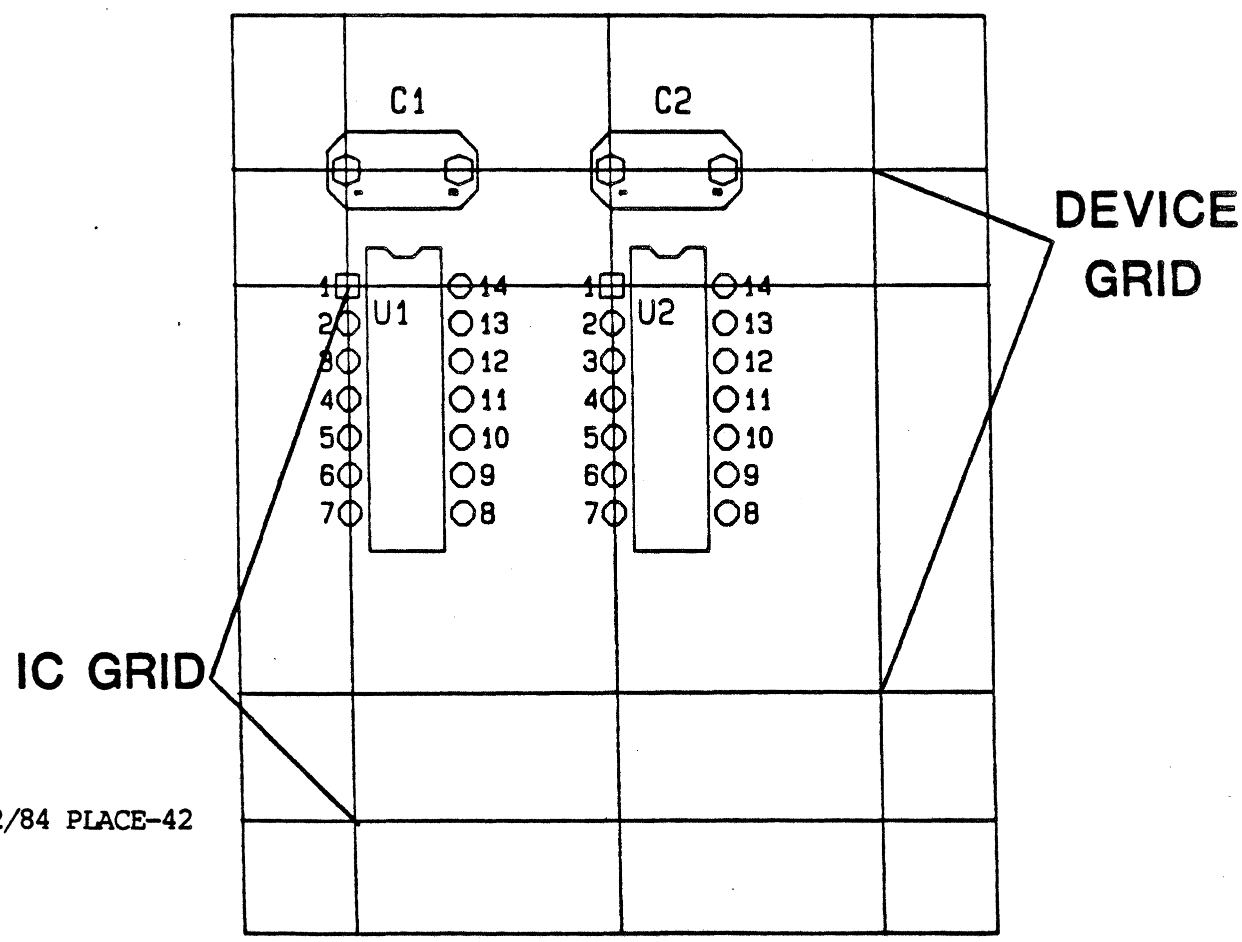
IC GRID SIZE:	X: 500	Y: 1200
IC GRID ORIGIN:	X: 450	Y: 1400

CURRENT |DEVICE GRID| PARAMETERS

DEVICE GRID SIZE:	X: 500	Y: 1200
DEVICE GRID ORIGIN:	X: 450	Y: 1800

NOTE: THE |DEVICE GRID| ORIGIN IS OFFSET 400 MILS IN THE Y-DIRECTION FROM THE IC GRID.

RESULT: Capacitors are placed at the DEVICE grid origin (offset 400 mils from the IC grid) during execution of the |AUTO-PLACE BOARD| command.



|DISCRETE GRID|

Use the |DISCRETE GRID| parameter to define a matrix of valid locations for DISCRETES. The input sequence is the same as |IC GRID|.

INPUT SEQUENCE:

|DISCRETE GRID| (DISCRETE GRID SIZE X) -Keyboard-Input |ENTER|

(DISCRETE GRID SIZE Y) -Keyboard-Input |ENTER| P |ENTER|

* P defines the grid origin.

DEFAULT: X: 100 Y: 100 Origin: 0,0

|IC ROTATION|

Use |IC ROTATION| to define the rotations of all ICs to be automatically placed. The operator may specify up to four rotation angles: 0, 90, 180, 270 (degrees). The system places each component at one of the rotations specified, selecting the best and most efficient rotation angle for each unplaced IC (if more than one angle is specified).

INPUT SEQUENCE:

|IC ROTATION|	(ALLOW 0 ROTATION [Y/N])	-Keyboard Input	|ENTER|
	(ALLOW 90 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 180 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 270 ROTATION [Y/N])	-Keyboard Input-	|ENTER|

The keyboard input is the character Y (yes) or N (no).

DEFAULT:

0 DEG:	Y
90 DEG:	N
180 DEG:	N
270 DEG:	N

|DEVICE ROTATION|

Use |DEVICE ROTATION| to define the rotations of unplaced user-defined DEVICES. The operator may specify up to four rotation angles. The system places each component at one of the rotations specified, selecting the best and most efficient rotation angle for each unplaced DEVICE (if more than one angle is specified).

INPUT SEQUENCE:

|DEVICE ROTATION|	(ALLOW 0 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 90 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 180 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 270 ROTATION [Y/N[)	-Keyboard Input-	|ENTER|

The keyboard input is the character Y (yes) or N (no).

DEFAULT:	0 DEG:	Y
	90 DEG:	N
	180 DEG:	N
	270 DEG:	N

|DISCRETE ROTATION|

Use |DISCRETE ROTATION| to define the rotation of unplaced DESCRETES. The operator may specify up to four rotation angles. The system places each component at one of the rotations specified, selecting the best and most efficient rotation angle for each unplaced DISCRETE (if more than one angle is specified).

INPUT SEQUENCE:

|DISCRETE ROTATION|	(ALLOW 0 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 90 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 180 ROTATION [Y/N])	-Keyboard Input-	|ENTER|
	(ALLOW 270 ROTATION [Y/N])	-Keyboard Input-	|ENTER|

The keyboard input is the character Y (yes) or N (no).

DEFAULT:	O DEG:	Y
	90 DEG:	N
	180 DEG:	N
	270 DEG:	N

|DEVICE NAMES|

Use |DEVICE NAMES| to specify the device types (up to 5) of IC or DISCRETE components to be placed on the |DEVICE GRID|. This command is useful in specifying multiple "like-devices", such a bypass capacitors and memory arrays, that are to be placed near components they are connected to.

INPUT SEQUENCE:

|DEVICE NAMES|	(LIST DEVICE(S) TO PLACE)	-Keyboard- Input	|ENTER|

The keyboard input is the device type (e.g. 7400). The operator may specify up to 5 device types; separate each device type with a blank space, or a comma during input.

DEFAULT: NONE

|I/O WEIGHT|

Use |I/O WEIGHT| to define to the system the relative importance and placement priority of IC-I/O connections and IC-IC connections.

Based on the weight factor specified, the system will prefer placement of an IC near an I/O to which it is connected, over placement near another IC to which it is connected.

INPUT SEQUENCE:

|I/O WEIGHT|	(I/O WEIGHT)	-Keyboard- Input	|ENTER|

The keyboard input must be a value between 0 and 10. However, decimal values are allowed. (e.g. 1.5)

If 1 is input, IC-I/O connections will be equal to IC-IC connections. If 2 is input, IC-I/O will be twice as important as IC-IC connections.

DEFAULT: 2.0

|KEEPOUT LAYERS|

Use |KEEPOUT LAYERS| to specify the data base layers where KEEPOUT rectangles exist. The operator may specify up to 10 unreserved layer assignments for this parameter. The system looks for KEEPOUT rectangles on the layers specified during execution of the |AUTO-PLACE BOARD| command.

INPUT SEQUENCE:

|KEEPOUT LAYERS| (ENTER KEEPOUT LAYER) -Keyboard Input |ENTER|

If more than one KEEPOUT layer exists, input a comma, or a blank space between layers during the keyboard input. If the operator does not use the KEEPOUT LAYERS parameter, the system looks for KEEPOUT rectangles on layer 40.

DEFAULT: LAYER 40

|COMPONENT OUTLINE LAYER|

Use |COMPONENT OUTLINE LAYER| to define the layer for any component space rectangles. The operator may input only one unreserved layer. This is the layer that was active when the rectangle was added to the symbol drawing, prior to |CREATE SYMBOL|.

The system examines this layer for any component rectangle added to a component symbol prior to placement. If component space rectangles do not exist, the system determines the minimum space requirement for each package symbol to be placed, looking at the extents of each graphic item in the symbol, except text.

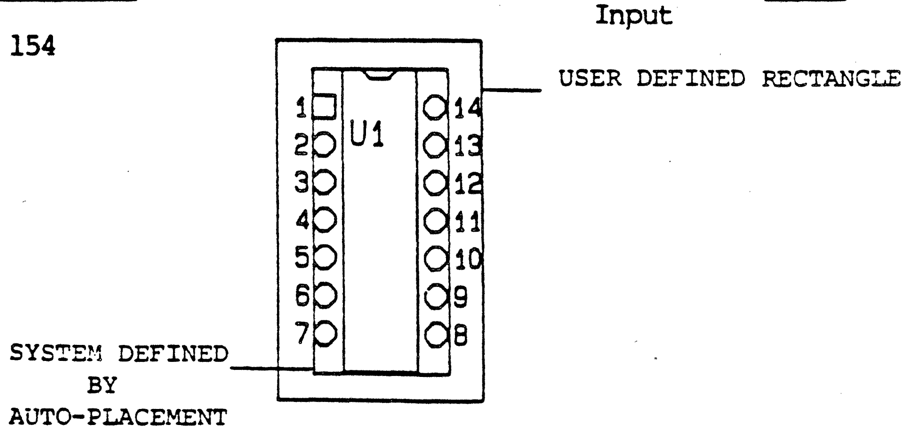
CAUTION: ENSURE THAT ANY RECTANGLES ADD TO PACKAGE SYMBOLS ARE NOT LARGER THAN THE GRID SIZE ESTABLISED FOR THE PLACEMENT OF COMPONENTS (ICs, DISCRETES, DEVICES). OTHERWISE, THE SYSTEM WILL NOT BE ABLE TO USE ALL AVAILABLE GRID POINTS. FOR EXAMPLE, A COMPONENT SPACE RECTANGLE ON AN IC PACKAGE SYMBOL THAT WOULD OVERLAP TWO AVAILABLE IC GRID POINTS MAY CAUSE INCOMPLETE PLACEMENT OF ICs WITH FEWER AVAILABLE GRID POINTS.

INPUT SEQUENCE:

|COMPONENT OUTLINE LAYER| (COMPONENT OUTLINE LAYER) -Keyboard- Input |ENTER|

DEFAULT: LAYER 154

EXAMPLE:



|RUN FUNCTION SWAP|

Use the |RUN FUNCTION SWAP| command to specify swapping of FREE IC logic functions during execution of the |AUTO-PLACE BOARD| command. Function swapping may occur between identical logic function types within a device, or with other devices containing the same functions (i.e. multiple instances of the device 7400). That is, functions will swap between components of the same device type.

INPUT SEQUENCE:

|RUN FUNCTION SWAP| (RUN FUNCTION SWAP [Y/N]) -Keyboard- Input |ENTER|

The keyboard input is the character Y (yes) or N (no).

DEFAULT: N (no)

|RUN IC SWAP|

Use the |RUN IC SWAP| parameter to specify swapping of FREE IC components during execution of the |AUTO-PLACE BOARD| command. Automatic swapping of components attempts to improve the board for shorter, straight horizontal/straight vertical ratsnest connect-lines, preparing the board for increased automatic routing completion percentage.

Any two free components will be considered for swapping as long as there is enough space at each component location for the other component. For example, a DIP40 may swap with a DIP14 if there is enough space at the DIP14 location for the DIP40.

INPUT SEQUENCE:

|RUN IC SWAP| (RUN IC SWAP [Y/N]) -Keyboard- Input |ENTER|

The keyboard input is the character Y (yes) or N (no).

DEFAULT: N (no)

|RUN PIN SWAP|

Use the |RUN PIN SWAP| parameter to specify swapping of FREE, swappable IC logic function pins during execution of the |AUTO-PLACE BOARD| command. Swappable pins are those specified in the device description file for each IC component in the net data base. Pin swapping allows improvement of ratsnest connect-lines for increased automatic routing completion percentage.

INPUT SEQUENCE:

|RUN PIN SWAP| (RUN PIN SWAP [Y/N]) -Keyboard- Input |ENTER|

The keyboard input is the character Y (yes), or N (no).

DEFAULT: N (no)

|MAX FUNCTION SWAP TIME|

Use the |MAX FUNCTION SWAP TIME| parameter to specify the maximum time (in minutes) to be allowed for swapping of IC logic functions. During execution of the |AUTO-PLACE BOARD| command, function swapping will terminate when the maximum time (if needed) is reached. Function swapping stops before the maximum time allowed if no further improvements can be made.

INPUT SEQUENCE:

|MAX FUNCTION SWAP TIME| (MAX FUNC SWAP TIME IN MINUTES)

-Keyboard Input- |ENTER|

The keyboard input specifies the maximum number of minutes. The input must be positive; decimal values are not allowed.

DEFAULT: 60 MINUTES

|MAX IC SWAP TIME|

Use the |MAX IC SWAP TIME| parameter to specify the maximum time (in minutes) to be allowed for swapping of IC components. During execution of the |AUTO-PLACE BOARD| command, component swapping will terminate when the maximum time (if needed) is reached. IC swapping stops before the maximum time allowed if no further improvements can be made.

INPUT SEQUENCE:

|MAX IC SWAP TIME| (MAX IC SWAP TIME IN MINUTES) -Keyboard- Input |ENTER|

The keyboard input specifies the maximum number of minutes. The input must be positive; decimal values are not allowed.

DEFAULT: 60 MINUTES

|LIST|

Use the |LIST| command to display the PLACE-CON text file of all |AUTOMATIC PLACEMENT| parameters. These parameter settings are displayed on the function screen.

The operator may use the |PRINT| softkey at the top of the function screen to print the file on the optional matrix printer.

|DONE|

Pick |DONE| when finished setting the |AUTOMATIC PLACEMENT| parameters. The current parameter settings are saved; the menu flips back to the |AUTOMATIC PLACEMENT| menu.

|CANCEL|

Pick |CANCEL| if the current parameter settings are to be ignored. |CANCEL| does not change parameters that were previously saved with the |DONE| command.

5. THE |AUTO-PLACE BOARD| COMMAND AND THE PLACE-LOG TEXT FILE

To start automatic placement of components, pick the |AUTO-PLACE BOARD| command. The system places components on the board in the following sequence:

1. Placement of all unplaced ICs (if specified in the parameters).

2. Placement of all unplaced DEVICES (if specified in the parameters).

3. Placement of all unplaced DISCRETES (if specified in the parameters).

The function screen message line displays the following information:

-AUTOMATIC PLACEMENT SET-UP STARTED...PLACEMENT SET-UP COMPLETE-

The system then proceeds with the placement of the board. The message line displays the component reference designator and the explicit drawing coordinates for each component being placed.

EXAMPLE: (PLACING U11 @ (600,1400)

When the system completes the |AUTO-PLACE BOARD| command, the system executes the |RATSNEST| command.

THE PLACE-LOG

A PLACE-LOG text file is created when the operator picks the |AUTOMATIC PLACEMENT| command. This is a text file of status information and error conditions: the system builds the log file during execution of the |AUTO-PLACE BOARD| command.

The PLACE-LOG may belisted at any time on the function screen with the |LIST PLACE-LOG| command. It may also be printed with the |PRINT PLACE-LOG| command. The following information is contained in the file:

GENERAL PLACEMENT PARAMETERS
PACKAGE SPACE REQUIREMENTS
KEEPIN/KEEPOUT AREAS
PREPLACED COMPONENTS
NUMBER OF UNPLACED COMP.
NUMBER OF PREPLACED COMP.
IC PLACEMENT PARAMETERS
DEVICE PLACEMENT PARAMETERS
DISCRETE PLACEMENT PARAMETERS
FUNCTION SWAP SUMMARY
IC SWAP SUMMARY
PIN SWAP SUMMARY
ERROR CONDITIONS

M A N U A L P L A C E M E N T

1. ADDING SYMBOLS
2. THE REFERENCE DESIGNATOR COMMANDS
3. HOW TO USE THE REFERENCE DESIGNATOR COMMANDS
4. USING THE ASSIGN ALL REF DES AND DEASSIGN ALL REF DES COMMANDS

1. ADDING SYMBOLS

You may place components in a board drawing by using the ADD SYMBOL commands and the |ASSIGN REF DES| command. The ADD SYMBOL commands place symbols from the component symbol library in the board drawing. The |ASSIGN REF DES| command makes the link between each symbol and the net data base that belong with this board drawing.

You may use any one of the ADD SYMBOL commands to place component symbols in a board drawing.

|ADD SYMBOL[MENU]|	Use this command followed by a pick from the Telesis-prepared symbol menus to add a symbol from the Telesis-prepared library.
|ADD SYMBOL[USER]|	Use this command followed by a pick from your own customized symbol menus to add a symbol from your own library.
|ADD SYMBOL[NAME]|	Use this command followed by a symbol file name to add a symbol from either library.

Refer to the Command Description Section of the manual for the exact input sequence to use with each command.

2. THE REFERENCE DESIGNATOR COMMANDS

There are 3 reference designator commands that can be used with board components:

COMMAND	PURPOSE
ASSIGN REF DES	1. To place reference designator labels on board components. 2. To identify components on the board drawing with components in the net data base. When you use ASSIGN REF DES to label a component, the system looks in the net data base for the same reference designator. All of the net information belonging to that reference designator in the net data base can now be used with the board component. This identification is required so that the system can perform operations such as RATSNEST, NET COMPARE, and NET LOCK.
CHANGE REF DES	The CHANGE REF DES command changes the text of the reference designator label you previously placed on a component with the ASSIGN REF DES command. It does not change the relation to the net data base established when you originally use the ASSIGN REF DES command. The net information belonging to a component before the name change still belongs to the component. Usually, you use the CHANGE REF DES command when your board design is nearly complete in order to bring the row and column reference designation into conformance with manufacturing standards.
DEASSIGN REF DES	This command has two purposes: 1. To delete the text of the reference designator label you previously placed on a component with the ASSIGN REF DES or CHANGE REF DES commands. 2. To delete the link formed by the ASSIGN REF DES command between a board compponment and the net data base. After using DEASSIGN REF DES on a component, you may use ASSIGN REF DES again to give the component a new label and a new link to the net data base.

PREREQUISITES

There are two prerequisities for the |ASSIGN REF DES|, |CHANGE REF DES|, and |DEASSIGN REF DES| commands:

1. The board must be the active drawing.

2. The net data base must be in the current project file.

DIFFERENCE BETWEEN |ASSIGN REF DES| AND |ATTACH REF DES|

Do not confuse the |ASSIGN REF DES| command with the |ATTCH REF DESIG| command or the |ADD REF DESIG| command. Only the |ASSIGN REF DES| command can associate a component on the board with a component in the net data base.

Once you have placed a reference designator on a component with the |ASSIGN REF DES| command, only the |CHANGE REF DES| or |DEASSIGN REF DES| commands can be used to change or delete it. Do not use the |UPDATE TEXT| or |DELETE TEXT| commands on these reference designators.

3. HOW TO USE THE REFERENCE DESIGNATOR COMMANDS

The Command Description Section of the manual tells you how to use the |ASSIGN REF DES|, |CHANGE REF DES|, and |DEASSIGN REF DES| commands.

Here are some general principles to keep in mind as you use them:

- You may assign as many (or as few) reference designators as you like before going on the RATSNEST and NET COMPARE operations. It is your option to work on the whole board, or to work on it by section.

- When you have assigned reference designators to all components on the board, the system will display this message on the function screen:

  ASSIGNMENT OF REF DES COMPLETE

- If you end the |ASSIGN REF DES| command before assigning all components, the system will display a long message on the function screen listing all components in the net data base that have not yet been assigned in the board drawing. For IO and IC components, the list will give reference designators and device types. For DISCRETE components, the list will give reference designators and component values.

- After using the |CHANGE REF DES| command, you may use the BACK ANNOTATE commands to bring the schematic reference designators into conformity with those on the board drawing. (See p. NETDB-15.)

- If the system detects an error when you use the |ASSIGN REF DES|, |CHANGE REF DES|, or |DEASSIGN REF DES| commands, the function screen will display an error message immediately after you pick the component involved in the error.

ERRORS DETECTED WHEN THE |ASSIGN REF DES| COMMAND IS USED

REF DES ALREADY USED; PLEASE ASSIGN NEXT REF DES!:

The reference designator you are trying to assign has already been assigned to another component. The system will skip this reference designator and assign the next one with your next (P).

NO SUCH REF DES ON SCHEMATIC; PLEASE ASSIGN NEXT REF DES!:

The system does not recognize the existence of this reference designator on your schematic. (Did you remember to use the |EXTRACT NETLIST| command on all sheets of your schematic?) The system will skip this reference designator and assign the next one with your next (P).

WRONG DEVICE OR VALUE; PLEASE PICK A DIFFERENT COMPONENT!:

The component you have picked is a device type or value inconsistent with the net data base information. This reference designator will be assigned to the next component of the correct type or value that you pick.

WRONG PACKAGE TYPE; PLEASE PICK A DIFFERENT COMPONENT!:

The component you have picked is a package type inconsistent with the net data base information. This reference designator will be assigned to the next component you pick.

COMPONENT ALREADY ASSIGNED; PLEASE PICK A DIFFERENT COMPONENT!:

The component you have picked has already been assigned a reference designator. Your reference designator will be assigned to the next component you pick.

ACCESS FAILURE!:

The system cannot relate your reference designator assignment to the net data base. If you receive this message with your first (P), it may be that there is something wrong with the net data base information. (Did you remember to use the |EXTRACT NETLIST| command on the schematic drawing?) If you receive this message only after several (P) s, the problem is probably limited to the component you last picked.

The |ASSIGN REF DES| command is effectively cancelled when you receive this message.

When you receive this message, you should use the |CANCL ACTV DRWNG| command or the |SAVE DRW NEW RV| command to close the drawing. Then, use the |OLD DRAWING| command to activate the previous revision of the drawing and begin working again.

BE CAREFUL!: Do not use the |SAVE DRW OLD RV| command after receiving the ACCESS FAILURE! message. If you do, you will be destroying the previous revision of the drawing.

ERRORS DETECTED WHEN THE |CHANGE REF DES| COMMAND IS USED

COMPONENT NOT ASSIGNED!:

The |ASSIGN REF DES| command has not yet been used on this component. Your reference designator text will be placed on the next component you pick. If you want to skip this text, pick |NEXT| before picking the next component.

DUPLICATE REF DES NAME!:

You are using a reference designator name that already exists on another component. The system warns you of this but allows you to do it on the assumption that you will go on to change the reference designator text on the other component involved in the duplication.

ACCESS FAILURE!:

This message means the same with this command as it means with the |ASSIGN REF DES| command: the system cannot relate your input to the net data base.

ERRORS DETECTED WHEN THE |DESASSIGN REF DES| COMMAND IS USED

COMPONENT LACKS ASSIGNED REFDES-PICK DIFFERENT COMPONENT

The |ASSIGN REF DES| command has not yet been used on this component. Go on to pick another component or pick |CANCEL|.

4. USING THE |ASSIGN ALL REF DES| AND THE |DEASSIGN ALL REF DES| COMMANDS

The |ASSIGN ALL REF DES| and |DEASSIGN ALL REF DES| commands allow you to batch assign and batch deassign reference designators on a board drawing. The system performs each assignment (or deassignment) automatically. These commands operate much faster than the current |ASSIGN REF DES| and |DEASSIGN REF DES| commands. However, you must use the |UPDATE TEXT| command to label the reference designators on each board component symbol prior to using the |ASSIGN ALL REF DES| command.

When you use the |ASSIGN ALL REF DES| command, the system scans the net data base for all reference designators not currently assigned to the component symbols in the board drawing. The system then proceeds with the assignment of each reference designator to its component symbol on the drawing.

With |DEASSIGN ALL REF DES|, the system scans the board drawing and deassigns all reference designators that were previously assigned with the |ASSIGN REF DES| command, or with the |ASSIGN ALL REF DES| command.

PREREQUISITES

1. The board drawing must be active.

2. The net data base must be in the current project file.

3. You must use the |UPDATE TEXT| command to label the reference designator on each component symbol on the drawing prior to using the |ASSIGN ALL REF DES|.

THE |ASSIGN ALL REF DES| COMMAND

Before using the |ASSIGN ALL REF DES| command, ensure that all board component symbols displayed on the drawing are labeled with each reference designator to be assigned. The |UPDATE TEXT| command links each board component symbol to the appropriate reference designator in the net data base when you pick the |ASSIGN ALL REF DES| command.

When you pick |ASSIGN ALL REF DES|, the system scans the net data base for all unassigned reference designators linked to the board component symbols on the drawing. The function screen message line displays:

BEGINNING TO ASSIGN REFERENCE DESIGNATORS

The system then proceeds with the assignment of each reference designator to its board component symbol, one at a time. The message line displays the reference designator during the assignment.

ASSIGNING REFERENCE DESIGNATOR U2

While the system is running the |ASSIGN ALL REF DES| command, the message line displays.

SCANNING THE DRAWING FOR UNASSIGNED REF DES´S

This message appears when the system scans every ten component symbols on the drawing.

When all the reference designators stored in the net data base are assigned, the system issues a prompt stating that the |ASSIGN ALL REF DES| command is complete; it also displays the number of reference designator assignments that occurred.

COMPLETED, 6 ASSIGNS OCCURRED

The system then displays the REF-DES-LOG file on the function screen. This is a text file that is created when you pick the |ASSIGN ALL REF DES| command. It lists the status information and error conditions occuring during the assignment of all reference designators.

NOTE: If you pick CANCEL during the |ASSIGN ALL REF DES| command, the REF-DES-LOG file is still created; it contains all status information that was filed prior to |CANCEL|.

EXAMPLE: REF-DES-LOG

```
                    REF-DES-LOG          17-JAN-84  10:30:51

project ACC-TEST
drawing PCB

Beginning to assign reference designators
Completed.    6 assigns occurred

MESSAGE:  ASSIGNED
            ref des C1              symbol origin x 2000   y  1500
            package CAPRAD300       device

MESSAGE:  ASSIGNED
            ref des C2              symbol origin x  800   y  1800
            package CAPRAD300       device

MESSAGE:  ASSIGNED
            ref des J               symbol origin x  200   y   300
            package CON15S/156      device

MESSAGE:  ASSIGNED
            ref des R*              symbol origin x 1900   y  1300
            package RES400          device

MESSAGE:  ASSIGNED
            ref des R2              symbol origin x 1200   y  1800
            package                 device

MESSAGE:  ASSIGNED
            ref des U2              symbol origin x 1200   y  1500
            package                 device
```

POSSIBLE WARNINGS AND ERROR MESSAGES WHEN USING THE |ASSIGN ALL REF DES| COMMAND

WARNING: SYMBOL HAS NO REF DES LABEL, NO ASSIGNMENT MADE

A component symbol on the board drawing is not labeled with a reference designator or text point. Use the |UPDATE TEXT| command to label the symbol.

WARNING: BLANK REF DES TEXT ON PACKAGE SYMBOL, NO ASSIGNMENT MADE

This message appears in the REF-DES-LOG file if |UPDATE TEXT| was used without the input of a reference designator label.

WARNING: PACKAGE SYMBOL HAS EXTRA PINS NOT IN THE NET DATA BASE, ASSIGNMENT MADE

This indicates that a component symbol has extra or unused pins on ICs, edge connectors, etc.

ERROR: REF DES R3 NOT IN NET DATA BASE, NO ASSIGNMENT MADE

When using the |UPDATE TEXT| command to label reference designators on component symbols, ensure that you input the proper reference designator for each component.

ERROR: NET DATA BASE REQUIRES PACKAGE TYPE DIP14
DOES NOT MATCH PACKAGE SYMBOL NAME DIP16
NO ASSIGNMENT MADE

This error appears in the REF-DES-LOG if a reference designator is labeled to the wrong component symbol when using the |UPDATE TEXT| command.

ERROR: REF DES Q1 ALREADY ASSIGNED, NO ASSIGNMENT MADE

This occurs if the |ASSIGN REF DES| command was used to assign a reference designator to a component symbol prior to running the |ASSIGN ALL REF DES| command.

ERROR: NET-DATA-BASE REQUIRES VALUE 100
DOES NOT MATCH PACKAGE VALUE 1500
NO ASSIGNMENT MADE

This error occurs if the value assigned to the schematic symbol does not match the value assigned to the board component symbol.

NOTE: WHEN USING THE |UPDATE TEXT| COMMAND TO LABEL REFERENCE DESIGNATORS TO COMPONENT SYMBOLS, DO NOT USE THE KEYBOARD CARRIAGE RETURN TO CREATE MULTI-LINE ENTRIES. THE SYSTEM ONLY ASSIGNS A MULTI-LINE REFERENCE DESIGNATOR IF IT APPEARS IN THE SCHEMATIC SYMBOL, AND IS LABELED IDENTICALLY ON THE BOARD SYMBOL WITH THE |UPDATE TEXT| COMMAND.

USING THE |DEASSIGN ALL REF DES| COMMAND

After assigning reference designators to individual board component symbols with the |ASSIGN REF DES| command, or with the batch |ASSIGN ALL REF DES| command, you may use the |DEASSIGN ALL REF DES| command. This command simply deassigns all previously assigned reference designators.

Use |DEASSIGN ALL REF DES| on the board drawing prior to making engineering changes to the schematic drawing.

You may then,

1. Update the schematic drawing
2. |EXTRACT NETLIST|
3. Label the reference designator to each component symbol on the board drawing with |UPDATE TEXT|.
4. Use the |ASSIGN ALL REF DES| command to batch reassign the reference designators to the components on the board drawing.

When you pick |DEASSIGN ALL REF DES|, the system displays the following message on the function screen.

	-->PAGE	CANCEL

```
THIS COMMAND WILL DEASSIGN ALL REFERENCE DESIGNATORS
ON THIS BOARD DRAWING.

"--> PAGE" TO CONTINUE AND DEASSIGN
"CANCEL" TO TERMINATE WITH NO DEASSIGN
```

If you pick |CANCEL|, the system terminates the |DEASSIGN ALL REF DES| command. If you pick |-->PAGE|, the system then proceeds to deassign all assigned reference designators from the board drawing.

During the deassignment, the function screen message line displays the percentage of reference designator deasssignment.

EXAMPLE: DEASSIGNING ALL REFERENCE DESIGNATORS "BOARD 20% DEASSIGNED"
"BOARD 40% DEASSIGNED"
"BOARD 60% DEASSIGNED"
"BOARD 100% DEASSIGNED"

COMPLETED, 6 DEASSIGNS OCCURRED

PLACING COMPONENTS ON A BOARD DRAWING USING A PLACEMENT TEXT FILE

Components can be placed on a board drawing using placement information contained in the Place-Txt file. The Place-Txt file takes up less room on the system disk than a board drawing; therefore, it is useful for storing placement information that provided optimum results for other system utilities. Since the Place-Txt file is an ASCII file, placement information can be transferred between different CAD systems. The Place-Txt file can be created automatically using the CREATE PLACE-TXT command, or the file can be created manually using normal text editing procedures.

The Place-Txt file lists the reference designator, X and Y coordinates, and rotation angle of each component on a board drawing. The Place-Txt file can contain placement information for an entire board or the file can be edited to contain placement information for a particular section of a board (such as a memory array). Only one Place-Txt file can exist in any project; to store more than one placement file in the same project, rename additional placement files (Place-Txt is system default).

The Place-Txt file can be transferred between various CAD systems if the file is in ASCII, and the following format is maintained on both systems :

Ref Des	X and Y coordinate	Rotation Angle (in 90 degree increments)

Refer to CREATE PLACE-TXT in the COMMANDS DESCRIPTION section of this manual for procedures to create a Place-Txt file from a board drawing. Refer to the Telesis Keyboard Text Editor section within the BASICS section of the Users Manual for procedures to create a text file manually (name the file Place-Txt and ensure that the format shown above is maintained).

When the PLACE BY-TXT command is selected, the system places components in the Net Data Base on an active board drawing at the locations specified in the Place-Txt file. The system automatically assigns a Reference Designator to each component that it places on the board drawing. As the board or section of board is being placed, the system creates a PLACETXT-LOG file that lists placement status. The file lists the X and Y coordinate of each component listed in the Place-Txt file and indicates whether or not the component was placed successfully. Any errors or warning messages associated with the placement of the component are listed in the file. Refer to PLACE BY-TXT in the COMMANDS DESCRIPTION section of this manual for procedures to place components on a board drawing using the PLACE BY-TXT command.

THE RATSNEST

1. USING THE $\boxed{\text{RATSNEST}}$ COMMAND

2. HISTOGRAMS OF CHANNEL USAGE

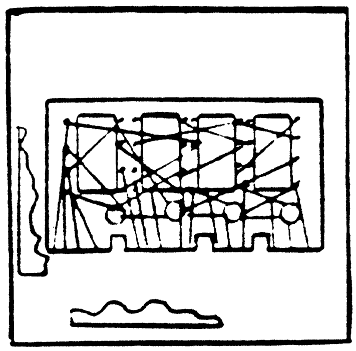
1. USING THE |RATSNEST| COMMAND

PURPOSE

The |RATSNEST| command instructs the system to display the shortest possible connections between the unconnected pins of each net on your board layout.

The resulting RATSNEST is a visual aid in the placement and routing of your board.

Below is an example of the graphics screen appearance after picking |RATSNEST|:



PREREQUISITES

1. The board must be the active drawing.

2. Net data base must be in the current project file.

3. You must have used the |ASSIGN REF DES| command on at least some of your board components. The RATSNEST will only include those nets connecting components to which you have assigned reference designators.

HOW TO USE THE COMMAND

The Command Description Section of the manual tells you how to use the |RATSNEST| command.

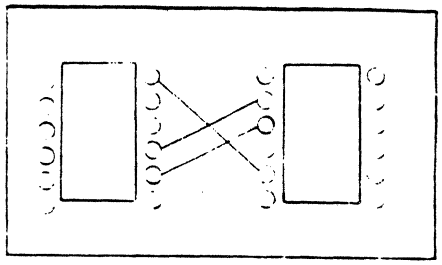
Here are some general principles to keep in mind as you use it:

. TIME REQUIRED FOR RATSNEST

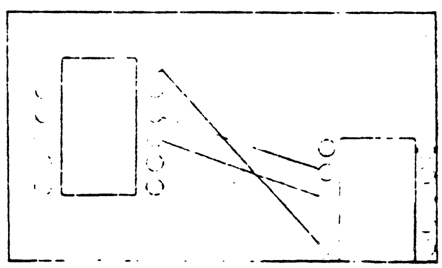
Ratsnest lines are drawn quickly on an unconnected board. On a partially routed board, the RATSNEST requires a considerably longer time; in some cases, more than an hour.

- RATSNEST LINES ARE RUBBERBANDED

  RATSNEST lines are rubberbanded as you make changes to your board:



GRAPHICS SCREEN BEFORE MOVING COMPONENT



GRAPHICS SCREEN AFTER MOVING COMPONENT

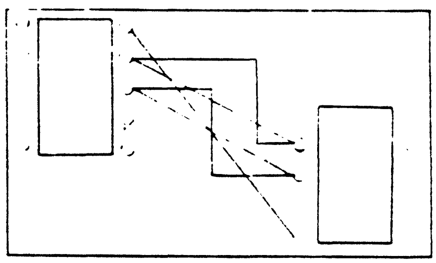
  As you edit and lines are rubberbanded, the resulting RATSNEST lines may no longer be the shortest possible interconnections. You may repeat the |RATSNEST| command to replace the rubberbanded lines with new RATSNEST lines.

- REPEATING THE RATSNEST COMMAND USES DISK SPACE

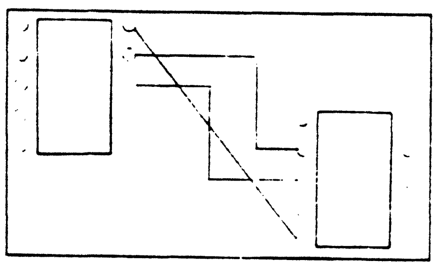
  You should be aware that if you repeat the |RATSNEST| command several times, you may be using as much as 1000 blocks of disk space.

- REMOVING RATSNEST LINES AFTER CONNECTIONS ARE ROUTED

  RATSNEST lines remain displayed as you rout your board unless you remove them. To remove RATSNEST lines for connections you have routed, repeat the |RATSNEST| command:



GRAPHICS SCREEN BEFORE REPEATING RATSNEST COMMAND



GRAPHICS SCREEN AFTER REPEATING RATSNEST COMMAND

- POWER AND GROUND SIGNALS

  Power and ground signals will not be ratsnested if you specifv power and ground layer in your LAYERSTD file.

- BLANKING THE RATSNEST DISPLAY

  The RATSNEST is on layer 101. You may use the |BLANK LAYER| command on layer 101 if you wish to remove the RATSNEST from your view.

  To display it again, use the |DISPLAY LAYER| command.

- CANCELLING THE RATSNEST BEFORE IT IS COMPLETED

  Use the |CANCEL| command if you want to cancel the RATSNEST before the system has finished drawing it. Lines already drawn will remain, but no new lines will be drawn.

2. HISTOGRAMS OF CHANNEL USAGE

In addition to displaying a RATSNEST, the system displays 2 histograms of channel usage when you use the |RATSNEST| command.

The histogram at the left of your drawing represents the connections passing through the horizontal channels of your drawing.

The histogram at the bottom of your drawing represents the connections passing through the vertical channels of your drawing.

If your board is partially routed, these histograms do not include routed lines as well as ratsnest lines.

The histograms are on layer 102. You may use the |BLANK LAYER| command to remove the histogram display. Use the |DISPLAY LAYER| command to display it again.

NOTE: Histograms of channel usage do not change as you edit the board unless you repeat the |RATSNEST| command. Doing so will give you updated histograms.

INTERCONNECTION

MANUAL INTERCONNECTION

1. TELESIS FEATURES USED WITH THE $\boxed{\text{ADD CONNECTION}}$ COMMAND

1. TELESIS FEATURES USED WITH THE |ADD CONNECTION| COMMAND

There are two ways to interconnect a board using the Telesis system:

1. MANUAL INTERCONNECTION: using the |ADD CONNECTION| command described in this section.

2. ROUTER: using the auto router described in the next section.

As you interconnect the board, there are a number of Telesis features that can be used in conjunction with the |ADD CONNECTION| command to assist you in your work:

NECK
DRILL
SET LAYER PAIR AND SWAP LAYERS
NET LOCKING
DELETING A NET
NET HIGHLIGHTING

NECK

The |NECK| command is used during the |ADD CONNECTION| command. It allows you to narrow the width of one segment of a connect line so that it can be placed in a space too narrow to accommodate its full width.

The Command Description Section of the manual tells you how to use this command.

DRILL

The |DRILL| command is used during the |ADD CONNECTION| command. It allows you to place a via at the end of a connect line segment, and continue the connect line on a different layer of your drawing.

The Command Description Section of the manual tells you how to use this command.

SET LAYER PAIR AND SWAP LAYERS

The |SET LAYER PAIR| command is used to establish the layers to be alternated (as active) when you use the DRILL command. The |SWAP| command is used to switch from one of these layers to the other without using the |DRILL| command. The Command Description Section of the manual tells you how to use these commands.

NET LOCKING

The Command Description Section of the manual tells you how to use the net lock commands. The table below tells you when to use them.

COMMAND	WHEN TO USE IT
|NETLOCK ON|	|NETLOCK ON| prevents you from making any connection on the board that is not also in the net data base.
|NETLOCK OFF|	This command removes the restriction placed by the |NETLOCK ON| command. Use |NETLOCK OFF| when you want to place a connection on the board that (because of error or engineering change) does not show up in your net data base. Use it also when you want to swap pins while interconnecting a board.

The net locking commands affect the results of the |NET COMPARE| command (see section on NET COMPARE).

If you have had net locking off, and have made board connections that differ from net data base connections, these differences will be listed in the NET-COMPARE-REPORT.

If you have had net locking on consistently while interconnecting the board, the NET-COMPARE-REPORT should show no differences. You should use the |NET COMPARE| command, nevertheless, to double-check for accuracy.

DELETING A NET

The |DEL NET SECTION| command allows you to delete a connected set of connect lines.

The Command Description Section of the manual tells you how to use this command.

NET HIGHLIGHTING

The Command Description Section of the manual tells you how to use the net highlighting commands. The table below tells you when to use them.

HIGHLIGHT NET	Use this command as a visual aid. It highlights an entire set of connect lines and pins when you pick any one of the lines or pins in the net.
HLIGHT NET/NUMBR	HIGHLIGHT NET BY NUMBER. Use this command as a visual aid when you know the net number but when you may not know the exact location of net on the graphics screen.
DEHIGHLIGHT NET	Use this command to return a highlighted net to its normal color.

The highlighting commands will highlight any members of the net displayed on the graphics screen--ratsnest lines, connect lines and/or connect points--even if the net has not yet been interconnected.

Only one net can be highlighted at a time. When you highlight a second net, the currently highlighted net will automatically be dehighlighted.

THE AUTOMATIC ROUTER

1. GENERAL

    o SUMMARY OF STEPS TO FOLLOW WHEN USING THE AUTO ROUTER
    o LIMITATIONS OF THE ROUTER.

2. PREREQUISITES TO OPERATING THE AUTO ROUTER

3. PREPARING THE BOARD DRAWING WITH THE ADD RECTANGLE COMMAND

    o OVERALL ROUTING AREA
    o VIA-INHIBIT AREAS
    o ETCH RESTRICTION AREAS

4. THE ROUTER COMMAND

    o SETTING THE AUTO ROUTER RESTRICTIONS

5. SETTING THE AUTO ROUTER PARAMETERS (OPTIONAL)

    o NOTES ON SETTING THE ROUTER PARAMETERS
    o DISPLAYING AND PRINTING THE ROUTER-CON TEXT FILE

6. STARTING THE AUTO ROUTER

    o NOTES ON SET-UP FILES
    o USING THE CANCEL COMMAND
    o REPEATED USE OF THE ROUTER START-UP COMMANDS
    o CHANGING PARAMETERS BETWEEN USE OF ROUTER START-UP COMMANDS
    o THE BOARD DRAWING DISPLAY DURING ROUTING

7. THE ROUTER-LOG FILE

8. VIA ELIMINATION

9. EDITING MANUALLY

10. AUTO ROUTER ERROR MESSAGES

11. ROUTING APPLICATIONS

    o GENEREAL APPROACH TO USING THE AUTO ROUTER
    o RECOMMENDED ROUTING PARAMETERS
    o STRATEGIES IN ROUTING SPECIFIC AREAS OF THE BOARD

        • ROUTING THE MEMORY ARRAY
        • ROUTING THE POWER BUSS

THE ROUTER CAPABILITY AND THE TELESIS SYSTEM

1. GENERAL INFORMATION

The Telesis automatic routing capability places connections on the board drawing. This feature operates according to physical design rules, parameters, and restrictions established by the operator. (The router capability is referred to as the "auto-router" in this manual.)

When the |ROUTER| command is picked, the system opens a control file called ROUTER-CON. The ROUTER-CON text file contains the parameter information needed to operate the auto-router. The operator may proceed to set the auto-router parameters, or start the router using default parameters set by the system.

In either case, the system uses the ROUTER-CON file and the essential drawing information in the net-data-base to route each connection.

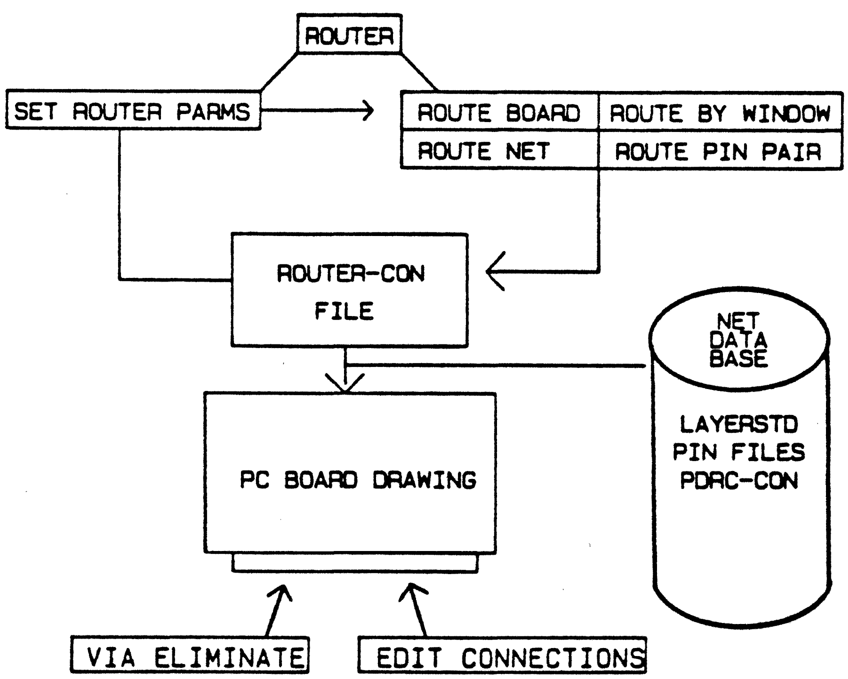
Presently, there are four commands that operate the auto-router in four different ways:

|ROUTE BOARD| |ROUTE BY WINDOW|

|ROUTE NET| |ROUTE PIN PAIR|

When the operator selects one of these commands, the system scans the net-data-base and assembles all the required information. The system then proceeds to route each connection.

The diagram below illustrates the automatic router capability.



SUMMARY OF STEPS TO FOLLOW WHEN USING THE AUTO-ROUTER

There are four preparatory steps to follow prior to starting the auto-router. Careful planning of these steps will achieve optimum PC board routing.

Each step is summarized below and treated in greater detail in the pages that follow.

STEP 1. PREREQUISITES

o A board drawing (with a net data base) must be active.

o All components should be placed. However, if an unplaced component is found, a warning is logged and the component is ignored.

o A pin description text file for every component in the board drawing must be in the current project or in the SYSTEM-LIBRARY.

NOTE: Edge connectors must have pin files. If a single edge connector has fingers defined on the drawing as lines-at-width or as rectangles, pad data does not need to exist in the pin file.

o A LAYERSTD text file must be in the current project or in the SYSTEM-LIBRARY.

o A PDRC-CON (physical design rules checking) text file must be in the current project or in the SYSTEM-LIBRARY.

o The net-data-base must be created with device files if floating connector pins are used during routing.

STEP 2. USING THE |ADD RECTANGLE| COMMAND

Use the |ADD RECTANGLE| command on specific data base layers to indicate: router keep-in, via-inhibit areas, and etch-restriction areas.

STEP 3. PICK THE |ROUTER| COMMAND

Pick the |ROUTER| command on the PC board menus to begin auto-router set-up and to display the first router menu page. Then, select and use the commands that place restrictions on the router. (e.g., Use |SET NO-ROUTE NET| to specify nets not to be routed.)

STEP 4. SETTING THE ROUTER PARAMETERS

Pick the |SET ROUTER PARMS| command to display the router parameters menu page. Then, select and input the parameters to be used during the router operation. Each parameter is a tool that controls the auto-router. The operator may use these tools to obtain the best routing results on a board drawing.

STEP 5. STARTING THE ROUTER

Use the |ROUTE BOARD|, |ROUTE BY WINDOW|, |ROUTE NET|, or the |ROUTE PIN PAIR| command to start the router. A considerable delay will occur while they system sets up the necessary files and information needed by the router. The system then proceeds to route the connections in the board drawing according to the command specified.

STEP 6. ELIMINATING VIAS (Optional)

When the system completes the router operation, use the |VIA ELIMINATE| command to instruct the system to try to eliminate any unnecessary vias placed in the drawing. The system does not re-route connections as vias are eliminated.

STEP 7. EDITING CONNECTIONS (Optional)

Use the |EDIT CONNECTIONS| command to display the command menu used to interactively add and edit connections on the board drawing.

NOTE: The operator will experience a slight delay when using |VIA ELIMINATE| and |EDIT CONNECTIONS| after running the auto-router. The system requires set-up time before these commands can be used. (See VIA ELIMINATION and EDITING MANUALLY for additional information on these commands.)

LIMITATIONS OF THE ROUTER

- A maximum of 800 pins in a routed net are allowed. If a net has more than 800 pins, the first 800 will be routed and the remainder will be ignored by the system.
- A net cannot have more than 1300 connect points, or 1300 connect-lines. This includes existing connections added manually, or with prior use of the auto-router.
- The system allows a maximum of 200 floating pins per connector.
- A connect-point may contain up to 50 attached connect-lines.
- A connect-line may contain up to 49 line-segments.

2. PREREQUISITES TO OPERATING THE AUTO-ROUTER

- o A board drawing (with a net-data-base) must be active.

- o All components should be placed. After the system sets up the required files needed by the router, components CANNOT be placed in the board drawing without exiting from the router menus.

  When the router is started, a message will appear on the function screen if unplaced components exist in the net-data-base, with all unplaced components listed in the ROUTER-LOG.

- o A pin description file for every component (except edge connectors) must be in the current project of in the SYSTEM-LIBRARY.

  Edge connector symbols in the board drawing must show fingers as connect-points surrounded by rectangles or a line-at-width. Each connect-point and its surrounding rectangle (or line) must be on the same layer. This layer cannot be LAYER 0.

- o A LAYERSTD file must be in the current project or in the SYSTEM-LIBRARY.

  NOTE: The system does not route or ratsnest power and ground nets if these nets are specified as imbedded-planes in the LAYERSTD file. Furthermore, those layers defined as imbedded-planes cannot be specified as valid routing layers.

- o A PDRC-CON (physical design rules checking) text file must be in the current project or in the SYSTEM-LIBRARY. This file contains the physical design rules needed by the router.

  NOTE: This file must be named PDRC-CON. PDRC-CON is the default name given to the rules file when the physical design rules check is run. Use the |CHG TEXT FILE NM| command to change the file to PDRC-CON if this file was created with pre-8301 software.

- o The net-data-base must be created with device files if connectors with floating pins are declared in the router. Device files must be in the current project or in the SYSTEM-LIBRARY when |EXTRACT NETLIST| or |LOAD TXT NETLIST| is used to create the net-data-base.

NOTE: If connections are added interactively prior to using the router, use |NETLOCK| or |NET COMPARE| to ensure proper connections. Otherwise, the router will not operate if conflicting nets exist on the board and in the net-data-base.

3. PREPARING THE BOARD DRAWING WITH THE |ADD RECTANGLE| COMMAND

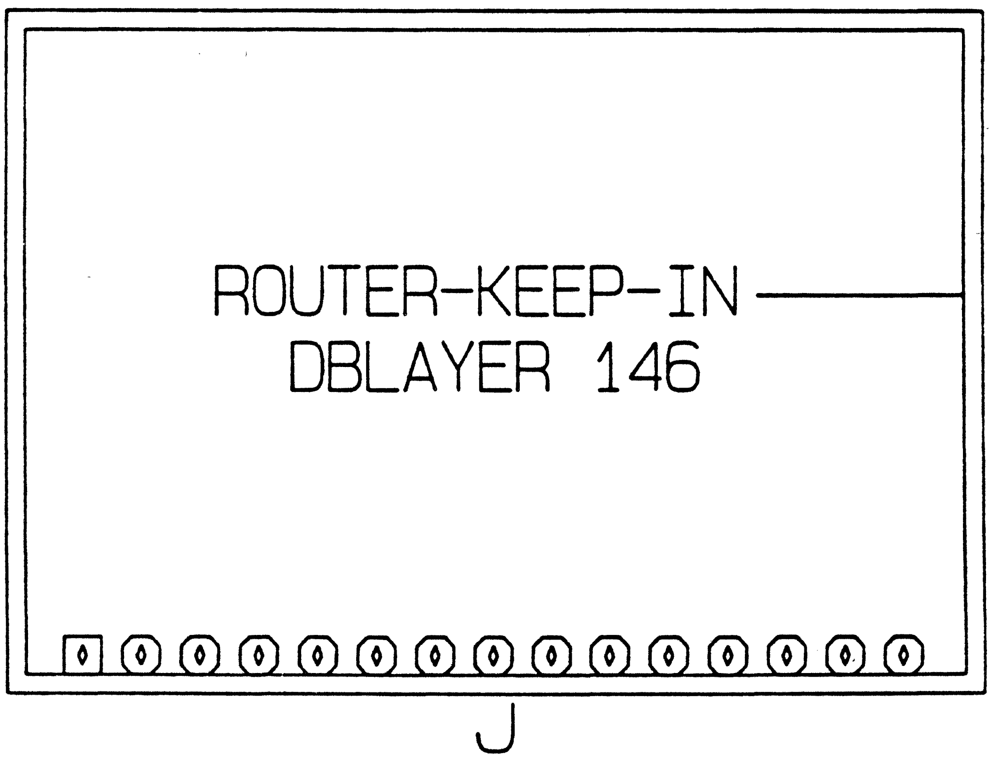
To prepare the board drawing for automatic routing: rectangles must be added to certain data base layers to indicate:

1. Overall routing area
2. Via inhibit areas
3. Etch restriction areas

The |ADD RECTANGLE| command must be used to place these rectangles. If the |ADD LINE| is used, the system does not ensure that a perfect rectangle has been placed.

OVERALL ROUTING AREA (DBLAYER 146)

The operator must place a rectangle on LAYER 146 to indicate the overall area of the board to be routed. Connections created by the auto-router will be inside this rectangle. (This is also called the "ROUTER KEEP-IN" area.) A rectangle on LAYER 146 is required before starting the auto-router, and must be placed on a grid equal to, or an even multiple of the routing grid. This will allow efficient "on-grid" connections during routing. (See INTERCON-17, ROUTER GRID SIZE.) All routed connect-lines will be created inside the rectangle on LAYER 146. A minimum line-to-line spacing (line-to-pad spacing for vias) will be maintained from the KEEP-IN rectangle to the nearest line.

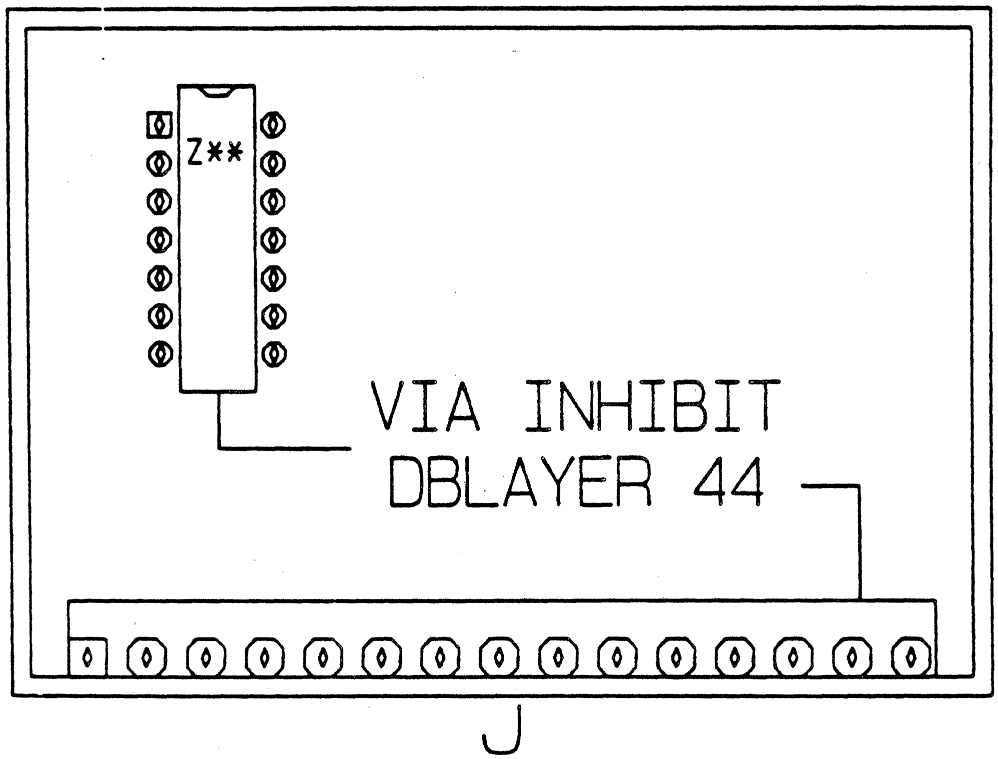


NOTE: IF THIS RECTANGLE IS NOT PLACED ON THE ROUTER GRID, OR AN EVEN MULTIPLE OF THE ROUTER GRID, THE ROUTER WILL NOT START.

VIA INHIBIT AREAS

Place rectangles on LAYER 44 to specify those areas on the board drawing where vias are disallowed. Via inhibit rectangles are optional.

EXAMPLE: Rectangle around edge connector and component symbol body



NOTE: Via inhibit rectangle may be placed on LAYER 44 when the symbol is created with the |CREATE SYMBOL| command.

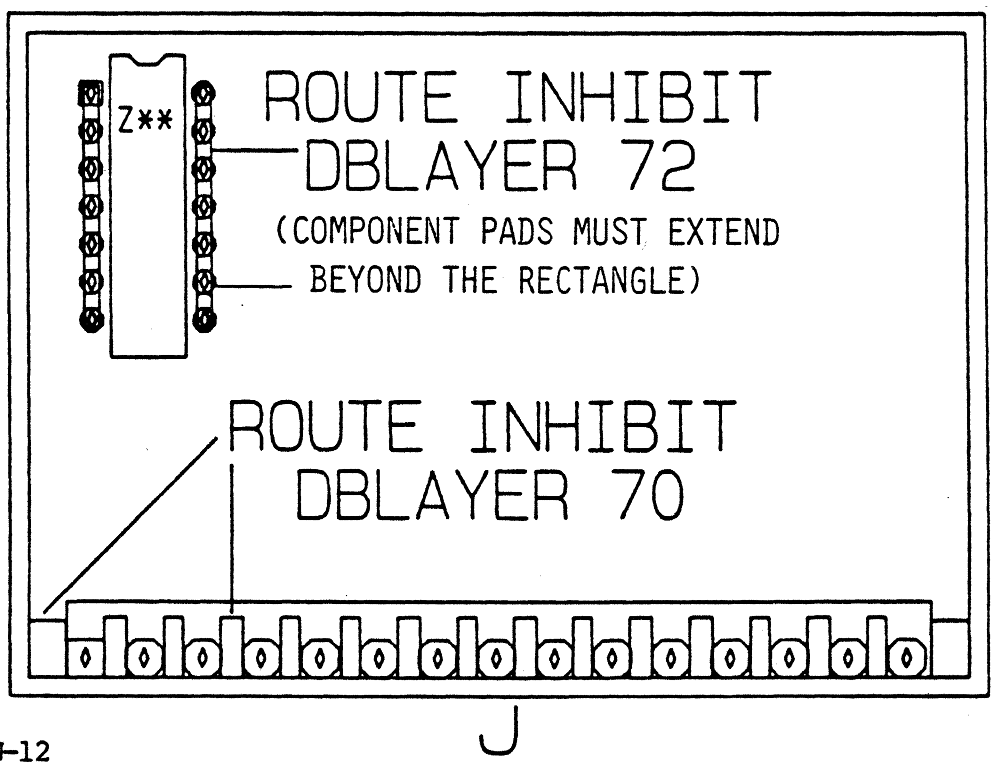
ETCH RESTRICTION AREAS

Place rectangles on the board drawing to specify areas of etch restriction during routing. Etch restrictions may be placed on LAYERS 70 through 84.

When a rectangle is placed on this layer:	Etch will be restricted on this layer:
70	all routing layers
71	1 - (drawing layer 1)
72	2 - (drawing layer 2)
73	3
74	4
75	5
76	6
77	7
78	8
79	9
80	10
81	11
82	12
83	13
84	14 (drawing layer 14)

EXAMPLE:

Etch restriction on DBLAYER 70 between connector pads.
Etch restriction on DBLAYER 72 over or between component pins (connect points) on board solder side.



THE |ROUTER| COMMAND

The |ROUTER| command allows the operator to open the router menus. When |ROUTER| is picked, there is a slight delay before the first router menu is displayed on the function screen. During this delay, the system reads the ROUTER-CON text file, creates log file called ROUTER-LOG, then extracts the design rules from the PDRC-CON file.

When the first router menu page is displayed, the operator may proceed to set the router parameters. However, the operator may start the auto-router using system default parameters.

The following menu page is displayed on the function screen when the |ROUTER| command is picked.

DONE	WORLD	ROAM	<- MENU	MENU ->	STATUS	DISPLAY	CANCEL

SET ROUTER PARAMS	EDIT CONNECTIONS		
RATSNEST			
LIST ROUTER-LOG		SAVE DRW SAME RV & CONTINUE	SAVE DRW NEW RV & CONTINUE
PRINT ROUTER-LOG			
ROUTE BOARD	ROUTE BY WINDOW	ROUTE NET	ROUTE PIN PAIR

Pick |MENU ->| to display the router restrictions.

	WORLD	ROAM	<- MENU	MENU ->	STATUS	ZOOM	CANCEL
MENU ->							

SET NO-ROUTE NET	SET NET LIN WIDTH	SET NO VIA ELIM	SET FLOATING CON
DEL NO ROUTE-NET	DEL NET LIN WIDTH	DEL NO VIA ELIM	DEL FLOATING CON
LST NO-ROUTE NET	LST NET LIN WIDTH	LST NO VIA ELIM	LST FLOATING CON
		VIA ELIMINATE	

SETTING THE AUTO-ROUTER RESTRICTIONS

There are three commands that allow the operator to set optional restrictions on the auto-router. The following optional commands must be set prior to starting the auto-router.

|SET NO-ROUTE NET|
|SET NET LIN WIDTH|
|SET FLOATING CON|

Refer to COMMAND DESCRIPTION section of this manual for the exact input sequence for each command discussed in this section.

NOTE: The via restriction commands on this menu page are discussed later in this section. (See VIA ELIMINATION)

|SET NO-ROUTE NET|

Use this command to specify nets not to be routed. When the auto-router is started, the system will not route the nets specified with this command. The operator must input the net name, or net number, when using this command. Use a comma (,) to separate inputs if more than one net is specified.

EXAMPLE: |SET NO-ROUTE NET| (NET NAME/NUMBER) 2, CLOCK+ |ENTER|

RESULT: Nets 2 and CLOCK+ will not be routed.

The |LST NO-ROUTE NET| command may be used to list the nets selected by the operator. The list is displayed on the function screen.

The |DEL NO-ROUTE NET| command may be used to delete any net from the list of NO-ROUTES.

|SET NET LIN WDTH|

Use this command to specify the nets to be routed at specific line widths. When the auto-router is started, the system routes these nets at the width specified, while unspecified nets are routed at the line width established by the |ROUTER LINE WIDTH| command.

When using this command, the operator must input the net names or net numbers, then input the line width [in mils]. Use a comma (,) to separate the nets during input. The operator may only input one line width per list of nets when this command is used. Repeat the command to list additional nets at other line widths.

EXAMPLE: |SET LIN WDTH| (NET NAME/NUMBER) 5,6 (VALUE) 10 |ENTER|

RESULT: Nets 5 and 6 will be routed at a 10 mil line width.

Use the |LST NET LIN WDTH| command to display the list of nets and widths specified with the |SET NET LIN WDTH| command. Use the |DEL NET LIN WDTH| command to delete any nets from this list of special line widths.

NOTE: The net line width plus the minimum line to line spacing (operator defined in PDRC-CON file) cannot exceed the router grid size. (See INTERCON-17, ROUTER GRID SIZE.) If it does, the net will not be routed.

|SET FLOATING CON|

Use this command to specify floating connector pins. When the auto-router is started, the system will treat the floating connect points as interchangable.

EXAMPLE: |SET FLOATING CON| (ENTER REF DES OF COMPONENT) J1 |ENTER|

(ENTER PINS) 1-12 |ENTER|

RESULT: Connect points with pin numbers 1 through 12 on component J1 will be treated as interchangable by the auto-router.

The list of pins may be separated by a comma (,) or a hyphen (-) to specify a series of pin numbers.

NOTE: Edge connectors with floating pins must be extracted with device files when the net data base is created. (See PRE-REQUISITES TO OPERATING THE AUTO-ROUTER, INTERCON-9.)

Use the |LST FLOATING CON| command to list the pins specified for the component reference designator. Use the |DEL FLOATING CON| command to delete a connect point, or connect points from the list of floating connectors for each component reference designator specified with the |SET FLOATING CON| command.

5. SETTING THE AUTO-ROUTER PARAMETERS (OPTIONAL)

The |SET ROUTER PARMS| command allows the operator to display the router parameters menu. The operator may select and specify each routing parameter to be used by the system during routing. If the routing parameters are not specified, the system assigns default values for each parameter.

When the |SET ROUTER PARMS| command is picked, the following menu is displayed on the function screen.

DONE				LIST		CANCEL
SET ROUTER PARMS						

ROUTER GRID SIZE	SKIP PASS 1	SKIP PASS 2	P5 WINDOW EXPAN
ROUTING LAYERS	P1 WINDOW EXPAN	P2 WINDOW EXPAN	P5 LAYER PAIRS
COMP ORIENTATION	P1 PIN KEEP AWAY	P2 PIN KEEP AWAY	P5 PIN KEEP AWAY
DIAGONAL ALLOWED	ROUTER LINE WIDTH		P5 JOG SIZES
STATUS MESSAGE FREQUENCY	P5 ROUTER TYPES	NUM P5 EXECUTES	P5 VIA ALLOWED

As the operator proceeds to set the auto-router parameters, the system stores each input in the ROUTER-CON text file. The ROUTER-CON may be listed on the function screen at any time while the router parameters menu is displayed. Any parameter not set by the operator is assigned the system default value.

NOTE: When |DONE| is picked, the ROUTER-CON text file is saved by the system, and is used when the auto-router is started. If the operator chooses not to start the auto-router after setting the parameters, the system deletes the old version of the file (default values or previously set values) when |DONE| is picked to exit the router menus. The function screen message line displays:

text file ROUTER-CON REV 1 deleted

However, if the operator picks |CANCEL| while setting the auto-router parameters the system deletes all the new entries. The system maintains the previous ROUTER-CON file (default values or previously set values).

When editing an existing ROUTER-CON file with the |SET ROUTER PARMS| command, the operator may update any number of parameters. When |DONE| is picked, the system creates a new version of the file.

|ROUTER GRID SIZE| (DEFAULT VALUE: 25 MILS)

Use the |ROUTER GRID SIZE| command to specify the grid to be used during routing. The following considerations must be noted before specifying the router grid size.

1. ROUTER LINE WIDTH and LINE TO LINE SPACING

The router line width specified with the |ROUTER LINE WIDTH| command, plus the line-to-line spacing (defined in PDRC-CON file) cannot exceed the router grid size. If the grid size and line-to-line spacing are set so that nets cannot be routed, an error will be written in the ROUTER-LOG file stopping execution. If the values are set so that at least one net can be routed, the system will route that net. All other nets will be skipped.

EXAMPLE: |ROUTER LINE WIDTH| = 12 mils
\+
LINE-TO-LINE SPACING = 13 mils (Operator-defined in PDRC-CON file)

TOTAL = 25 mils (acceptable grid size)

RESULT: The routing grid must be equal to, or greater than the |ROUTER LINE WIDTH| plus the LINE-TO-LINE SPACING.

If the |ROUTER LINE WIDTH| plus the line-to-line spacing exceeds the |ROUTER GRID SIZE|, the system will not route connections on the drawing. The following error message will appear on the function screen.

ERROR IN PROCESSING. CHECK LOG FILE

Use the |LIST ROUTER-LOG| command to display the violation on the function screen.

2. PLACEMENT GRID

The grid used during component placement should be an even multiple of the grid established for routing. This allows efficient "on-grid" connections to occur during routing. "On-grid" connections will leave additional channels open on the drawing for later routes, and allows a higher auto-route completion.

EXAMPLE: Grid used during component placement - 100 mils

Grid established for routing - 25 mils

RESULT: The routing grid can be evenly divided into the placement grid.

|ROUTING LAYERS| (DEFAULT VALUE: 1 (HORIZONTAL) 2 (VERTICAL))

Use the |ROUTING LAYERS| command to specify the routing layers to be used by the system. Drawing layers 1 through 14 are the only allowable routing layers.

EXAMPLE:

|ROUTING LAYERS| (ENTER HORIZONTAL LAYERS) 1 |ENTER|

(ENTER VERTICAL LAYERS) 2 |ENTER|

RESULT: Horizontal connect lines will be routed on drawing LAYER 1.
Vertical connect lines will be routed on drawing LAYER 2.

To list more than one drawing layer for each direction, input a comma (,) between layer numbers.

EXAMPLE:

|ROUTING LAYERS| (ENTER HORIZONTAL LAYERS) 1,3 |ENTER|

(ENTER VERTICAL LAYERS) 2,4 |ENTER|

NOTE: To route on a single layer, simply pick |ENTER| after one prompt, then input the layer number for the other. This will create one routing layer.

|COMP ORIENTATION| (DEFAULT VALUE: V (vertical))

Use |COMP ORIENTATION| to inform the system of the predominant orientation (horizontal or vertical) of the components in the board drawing. The system requires this information when the auto-router is started.

EXAMPLE

|COMP ORIENTATION| (SPECIFY MAJORITY COMP. H OR V) H |ENTER|

- o The keyboard input is H (horizontal), or V (vertical).

|ROUTER LINE WIDTH| (DEFAULT VALUE: 12)

Use the |ROUTER LINE WIDTH| command to specify the line width to be used by the system during routing. All nets not specified with the |SET NET LIN WDTH| command will be routed at the line width specified with the |ROUTER LINE WIDTH| command.

EXAMPLE:

|ROUTER LINE WIDTH| (DEFAULT ROUTING LINE WIDTH) 10 |ENTER|

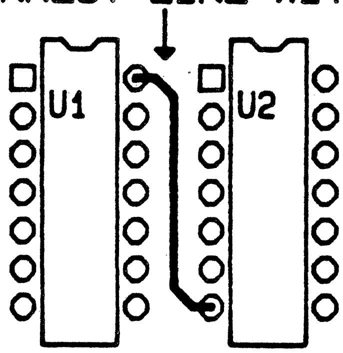
- o The keyboard input must be positive. Decimals and a 0-value are not allowed.

|DIAGONAL ALLOWED| (DEFAULT VALUE: Y [yes])

Use the |DIAGONAL ALLOWED| command to specify whether the auto-router may use diagonal instead of square corners for routed connect lines. Diagonal corners, if allowed, are 45 degree connect lines extending across one diagonal grid interval.

EXAMPLE: |DIAGONAL ALLOWED| (YES OR NO [Y or N]) Y |ENTER|

EXAMPLE: CONNECT-LINE WITH DIAGONALS



|SKIP PASS 1| (DEFAULT VALUE: N [NO])

|SKIP PASS 2| (DEFAULT VALUE: N [NO])

|NUM P5 EXECUTES| (DEFAULT VALUE: 2)

When the auto-router is started, the system makes up to three attempts to place the connections called for by the command. Each attempt executed by the router is called a "pass".

A "pass" places connections on the drawing according to parameters defined by the system. During a pass, the system completes the connections that adhere to these system parameters.

Currently, three "passes" are available:

PASS 1 PASS 2 PASS 5

When setting the auto-router parameters, the operator may specify the pass, or passes to be used with the following commands:

|SKIP PASS 1|
|SKIP PASS 2|
|NUM P5 EXECUTES|

Refer to the COMMAND DESCRIPTION of the manual for the exact input sequence for each of these commands.

o PASS 5 - This is the general two-layer pass. It attempts to complete all connections, including those that PASS 1 and PASS 2 failed to complete. When PASS 5 begins, it first attempts routing of the shortest connections, then proceeds to route longer connections.

There are two types of routers available in PASS 5. The first type is the line probe router; the second is the maze runner. The line probe router attempts to complete connections in the straightest possible path, while the maze runner considers a variety of directions before routing the connection.

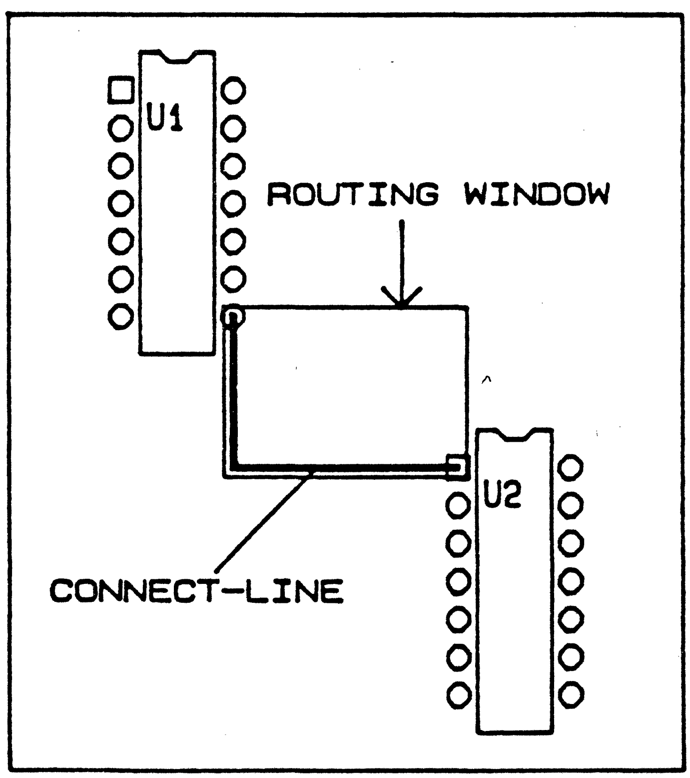
During PASS 5, the system scans the "ideal routing window",?*? then considers the connection with the line probe, then the maze runner. The system proceeds to route the connection when an available routing channel is located.

NOTE: A PASS 5 connection may route to an existing connect-line or via if it is part of the same net, and within the routing window. This allows shorter connections to occur during PASS 5.

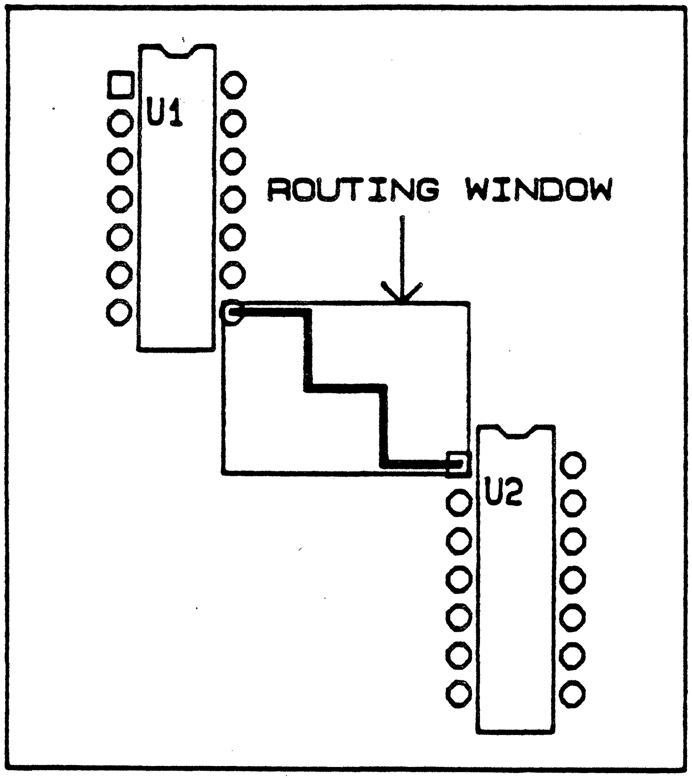
* The "ideal routing window" is a rectangle surrounding two pins (at opposite corners) that are to be connected. (See ROUTER WINDOW EXPANSION)

The line probe router is faster, but will not complete as many connections as the maze runner. The operator has the option of selecting one or both router types prior to starting the router.

EXAMPLE: LINE PROBE ROUTER



EXAMPLE: MAZE RUNNER



ROUTER WINDOW EXPANSION

|P1 WINDOW EXPAN| (DEFAULT VALUE: 5 [grid points])

|P2 WINDOW EXPAN| (DEFAULT VALUE: 6 [grid points])

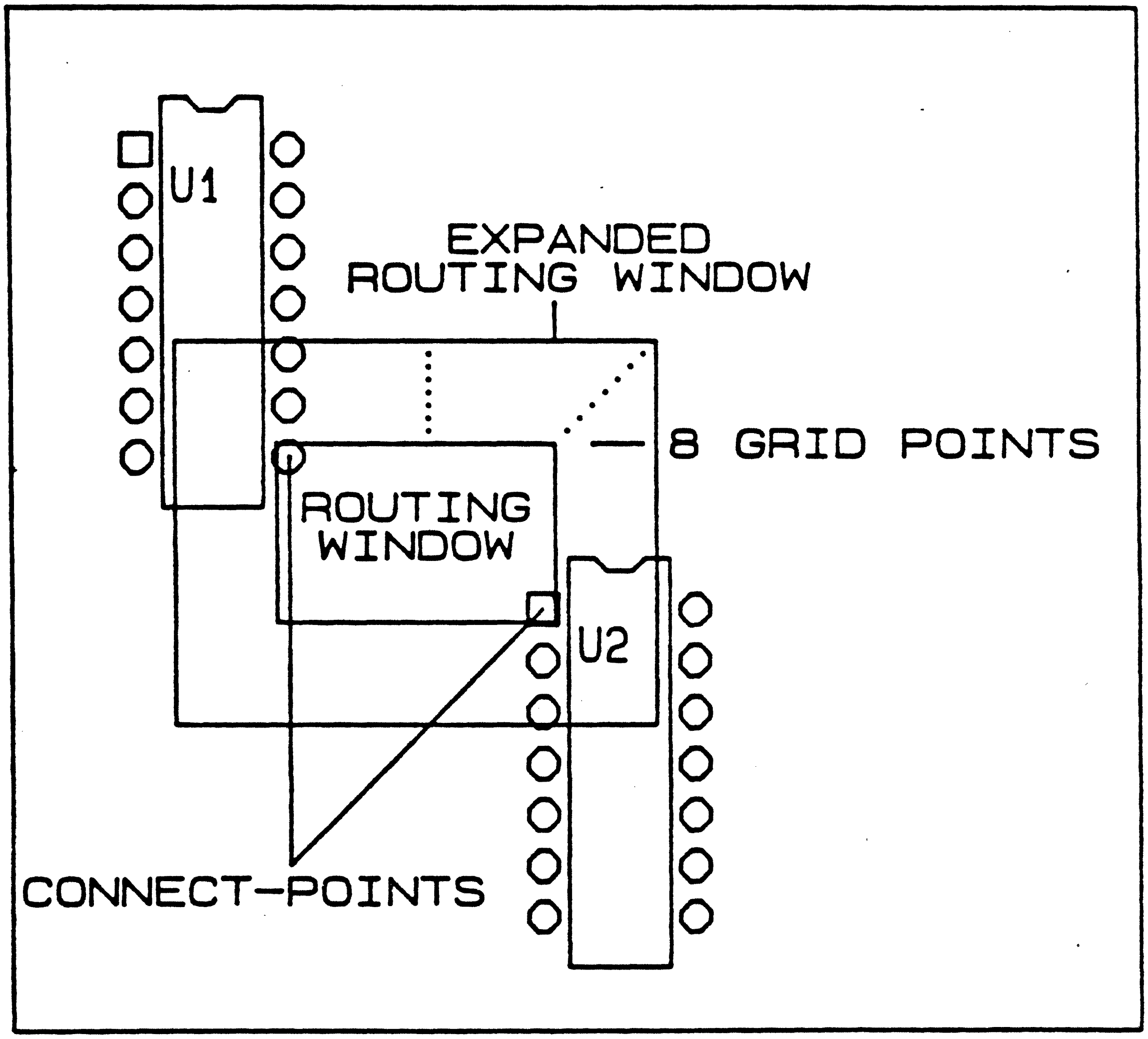
|P5 WINDOW EXPAN| (DEFAULT VALUE: 8, 16 [grid points])

Use the "window expansion" commands to specify the number of grid points outside the "ideal routing window" the system should consider when routing a connection. Window expansion is simply a method of defining the maximum area, or boundary, allowed when the auto-router considers a connect path between two connect points. The operator may define the window expansion for each pass, or use the system default values.

Window expansion allows the system to locate complicated connect line paths on a congested drawing. Congested drawings usually require large routing windows.

The example on the following page illustrates a system default window expansion during PASS 5.

EXAMPLE: TYPICAL PASS 5 WINDOW EXPANSION



SYSTEM DEFAULT VALUE OF 8

ROUTER PIN KEEP AWAYS

|P1 PIN KEEP AWAY| (DEFAULT VALUE: 2 [grid points])

|P2 PIN KEEP AWAY| (DEFAULT VALUE: 3 [grid points])

|P5 PIN KEEP AWAY| (DEFAULT VALUE: 0 [grid points])

Use these parameters to specify the distance (in grid points) a connect-line must travel before turning away from a component pin. Pin "keep-aways" will force connect-lines to be routed away from a row of pins; this allows routable channels to remain open for later routes.

When the auto-router is started, the system first establishes the routing window (per pin pair) and the pin keep away values for each pass to be executed. The system then proceeds to route each connection.

All connect-line segments running parallel to component pin rows (component orientation) must keep this distance away from the pins. Before routing to a pin, the connect-line must turn in a perpendicular direction.

The operator may set pin "keep-away" distances, or use the system default parameters.

SETTING PIN KEEP AWAY VALUES

The following examples illustrate various pin "keep-away" values and the resulting connect-line distances. Each example assumes a router grid size of 25 mils and a vertical component orientation.

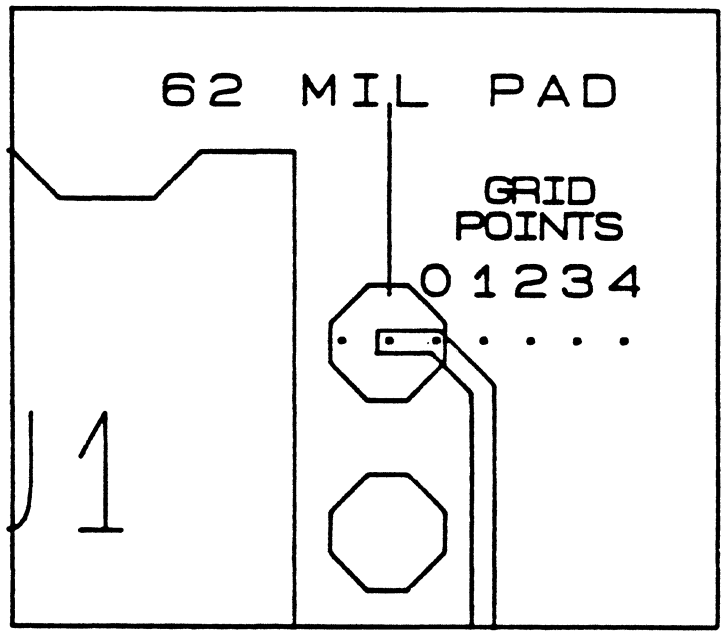
The system routes each connect-line to the specified pin keep away value, then in the direction that routes the pin pair.

NOTE: A connection occurring in PASS 1/ PASS 2 will ignore a pin keep away value if a straight routing channel exists between a pin pair.

PIN KEEP AWAY VALUE = 0 (VERTICAL COMPONENT ORIENTATION)
(25 MIL GRID SIZE)

This value allows the system to route the connect-line to the first grid point away from the center of the pad. The connect-line travels in the non-preferred direction (opposite to component orientation). The connect-line must travel to this grid point before turning to complete the pin pair connection.

EXAMPLE: PIN KEEP AWAY VALUE = 0

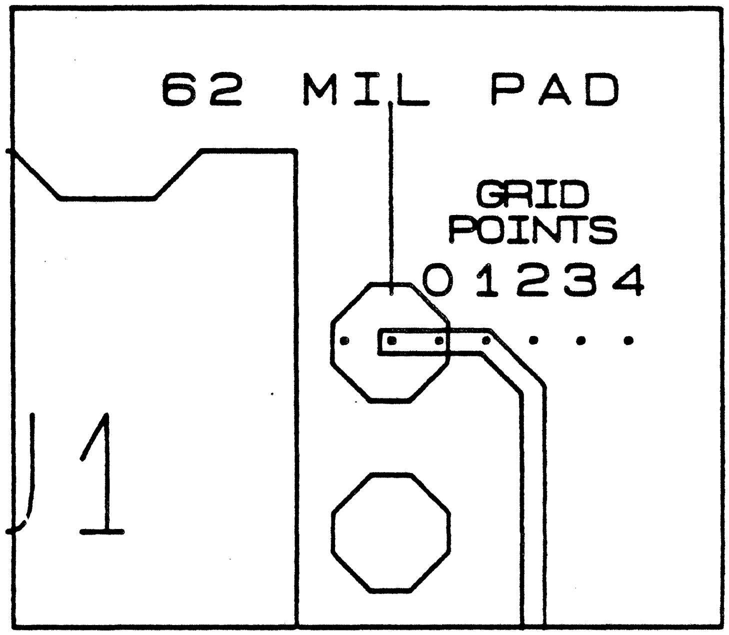


NON-PREFERRED CONNECT-LINE DIRECTION = ⟷

PIN KEEP AWAY VALUE = 1 (VERTICAL COMPONENT ORIENTATION)
(25 MIL GRID SIZE)

This value allows the system to route the connect-line to the second grid point away from the center of the pad. The connect-line travels in the non-preferred direction (opposite to component orientation). The connect-line must travel to this grid point before turning to complete the pin pair connection.

EXAMPLE: PIN KEEP AWAY VALUE = 1

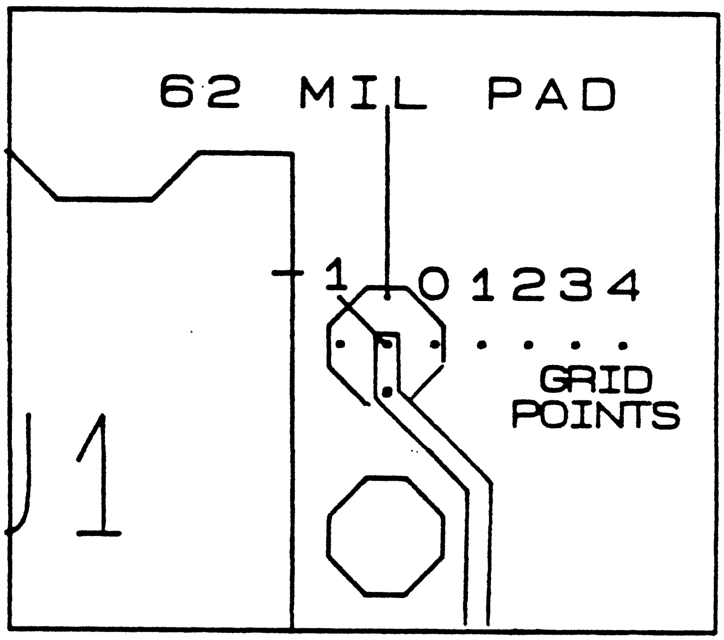


NON-PREFERRED CONNECT-LINE DIRECTION = ⟷

PIN KEEP AWAY VALUE = -1 (VERTICAL COMPONENT ORIENTATION)
(25 MIL GRID SIZE)

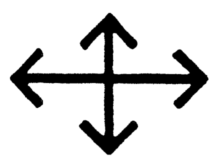
This value allows the system to route the connect-line to the first routable grid point outside the component-pad. The connect-line travels in either the preferred or non-preferred direction. The system considers four possible directions from the pad center. The connect-line must travel to one of four grid points before turning to complete the pin pair connection.

EXAMPLE: PIN KEEP AWAY VALUE = -1



PREFERRED / NON-PREFERRED CONNECT-LINE

DIRECTION =



|P5 LAYER PAIRS| (DEFAULT VALUE: 1,2 [for all executions])

Use this command to specify the routing layers to be used during PASS 5. These routing layers must be specified with the |ROUTING LAYERS| command.

EXAMPLE: |P5 LAYER PAIRS| (ENTER LAYER PAIRS) - Keyboard Input - |ENTER|

NOTE: If the PASS 5 routing layers are different than the layers specified with the |ROUTING LAYERS| command, the operator must reset the |ROUTING LAYERS| parameter before executing PASS 5.

For example, if the operator wishes to route LAYERS 1 and 2 during PASS 1 and PASS 2, and then use LAYERS 3 and 4 during PASS 5, the following procedure should be followed.

Step 1. Specify LAYERS 1 and 2 with the |ROUTING LAYERS| command.

Step 2. Use the |NUM P5 EXECUTES| command to specify 0-executions of PASS 5.

Step 3 Proceed to start the router with one of the router commands (i.e., |ROUTE BOARD|)

Step 4. After the router has completed PASS 1 and/or PASS 2, use the |ROUTING LAYERS| command to specify LAYERS 3,4. (i.e., HORIZONTAL LAYERS 3 VERTICAL 4). These layers must be valid routing layers.

Step 5. Use the |NUM P5 EXECUTES| command to specify the number of PASS 5 executions.

Step 6. Use the |P5 LAYER PAIRS| command to specify the PASS 5 routing layers. (i.e., LAYERS 3 and 4)

Step 7. Use the |SKIP PASS 1| and |SKIP PASS 2| commands to specify that PASS 1 and PASS 2 are to be skipped.

Step 8. Proceed to start the router.

RESULT: The system will route PASS 5 on layers 3 and 4.

NOTE ON SINGLE LAYER BOARDS

When specifying the PASS 5 layer pairs for a single layer board, the operator must specify the single layer number followed by a zero. For example, 1, 0 will allow the system to route on layer 1 only.

|P5 JOG SIZES| (DEFAULT VALUE: 5 [for all executions])

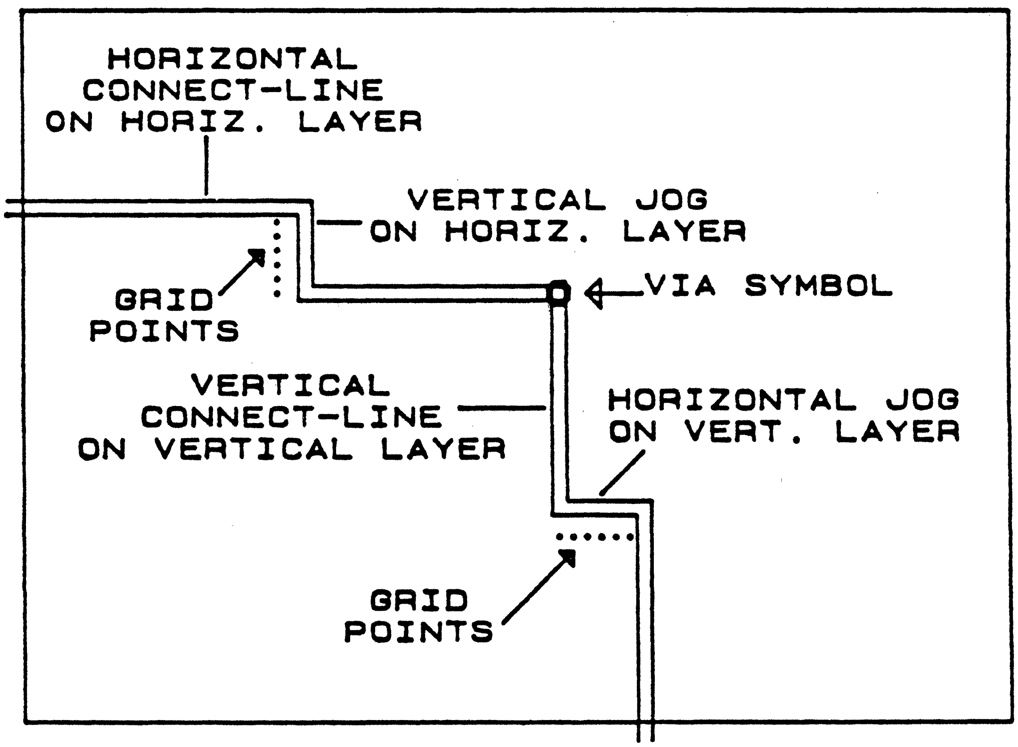
Use this command to specify the number of grid points a connect-line may turn vertically on a horizontal layer, and horizontally on a vertical layer. Routing "Jogs" allow the system to consider and use available routing channels (per layer) before inserting necessary vias to complete a pin pair connection.

EXAMPLE: |P5 JOG SIZES| (ENTER JOG SIZES) 6,20 |ENTER|

- o The keyboard input is the number of grid points.
- o The first execute of PASS 5 will jog connect-lines a maximum of 6 grid points.

  The second execute of PASS 5 will jog connect-lines a maximum of 20 grid points.

EXAMPLE: JOG SIZE OF 6



|P5 ROUTER TYPES| (DEFAULT VALUE:B [Both])

Use this command to specify the router type(s) to be used by the system during PASS 5. The operator may specify the line probe router, maze runner or both.

EXAMPLE: |P5 ROUTER TYPES| (TYPE OF ROUTER [L-M-B] -Keyboard- Input |ENTER|

L = line probe router

M = maze runner

B = both

The operator may specify a value for each execution of PASS 5. Separate each input with a blank space or comma.

|P5 VIA ALLOWED| (DEFAULT VALUE: Y [YES])

Use this command to specify if the router is to use vias during PASS 5.

EXAMPLE: |P5 VIA ALLOWED| (YES/NO PER PASS [Y/N]) -Keyboard- Input |ENTER|

The operator may specify a value for each execution of PASS 5. Separate each Y or N with a blank space or comma.

|STATUS MESSAGE FREQUENCY| (DEFAULT VALUE: 10)

Use this parameter to specify how frequently routing completion status is to be displayed on the function screen message line. The default value 10 indicates that the message is to be written after every 10 connection attempts. If the operator specifies a value of 0 for this parameter, no status messages will be written.

FORMAT: P5-2: COMPLETES=XX FAILS=XX SKIPS=XX REMAINS=XX

- o P5-2 indicates PASS 5 execution 2. The execution number is only shown during PASS 5 since PASS 1 and PASS 2 only have one execution.
- o COMPLETES= - specifies the number of completed connections per pass execution.

  FAILS= - specifies the number of failed connections per pass execution.

  SKIPS= - specifies the number of skipped connections.

  REMAINS= - specifies the number of remaining connections to be attempted in the pass executions.

EXAMPLE:

|STATUS MESSAGE FREQUENCY| (#ROUTE ATTEMPTS/STATUS MESSAG) 5 ENTER

THE STATUS MESSAGE DISPLAYED DURING ROUTING

During routing, the system displays a message listing the status of the connections being routed by the Auto Router. The status message pertains only to the pass presently being routed (i.e., the number of fails listed in the status message is the number of connections that could not be routed in this PASS out of the connections that were attempted during this PASS). The status message is updated after the number of attempted connections specified by the operator with the STATUS MSG FREQ command. The four categories listed in the status message are detailed below.

The following figure illustrates the status message line.

```
 ------------------------------------------------------------------
| PASS : EXECUTION     COMPLETES     FAILS     SKIPS     REMAINS |
 ------------------------------------------------------------------
```

The Pass and Execution category lists the Pass being executed. The Execution category is only used for PASS 5 to specify which execution of PASS 5 is being executed.

The Completes category lists the number of connections that the Auto Router successfully routed in this PASS.

The Fails category lists the number of connections that the Auto Router could not route in this PASS.

The Skips category lists the number of connections that the Auto Router did not attempt to route in this PASS because one of the component pins :

- is outside of the Router Keepin area
- is part of a net with a line width greater than the maximum line width being routed
- causes a Physical Design Rules Checking violation

NOTE

A pad-to-pad or line-to-pad violation will cause a skip for each component pin in violation.

- is not available on the layer being routed (i.e., during PASS 5 the Auto Router will not attempt to complete a connection to a component pin that can only be accessed from layer 2 if the routing layer pairs are layers 1 and 3)

NOTE

> The Auto Router can route a connection to a via or connect line that is connected to a component pin if the component pin itself is not available.

- is outside of the router window (if the operator has selected ROUTE BY WINDOW)
- is not part of the nets being routed (if the operator has selected ROUTE BY NET)
- is in a specified NO ROUTE NET (if the operator has selected ROUTE BY NET).

The Auto Router will not attempt to route a connection if the routing area required to route the connection is larger than the amount of memory available. The amount of memory required to route a connection depends on the distance between the component pins, the routing grid size, and the number of routing layers.

The Remains category lists the number of connections left to be routed in this PASS.

The following example illustrates the status message displayed during the second execution of PASS 5. The total number of connections that will be attempted during this execution of PASS 5 is 300. The Auto Router has completed 200 connections, failed to complete 20 connections, did not attempt to route 20 connections, and has 60 connections left to route.

```
 ---------------------------------------------------------------------
| P5-2:     COMPLETES=200     FAILS=20     SKIPS=20     REMAINS=60   |
 ---------------------------------------------------------------------
```

P1, P2, AND P5 JOG LIMITS

P1 JOG LIMIT (DEFAULT VALUE : -1)
P2 JOG LIMIT (DEFAULT VALUE : -1)
P5 JOG LIMIT (DEFAULT VALUE : -1)

The Auto-Router uses jogs, when necessary, to route connect-lines. This parameter specifies the maximum number of jogs that the Auto-Router can use on a connect-line.

The route a connect-line travels perpendicular to the routing direction of the board layer is a jog. On a horizontal routing layer, a jog is the route that the connect-line travels in the vertical direction.

This parameter specifies the maximum number of jogs that the Auto-Router is allowed to use on a connect-line. If the connect-line requires more jogs than the maximum limit specified with this parameter, the connect-line will not be routed.

If the operator specifies a jog limit of five, the Auto-Router will route only the connect-lines with less than five jogs. A value of -1 specifies that there is no limit to the number of jogs that the Auto-Router can use on a connect-line.

P5 VIA LIMITS (DEFAULT VALUE : -1)

The Auto-Router uses vias in PASS 5 to complete connections on the board that were not routed in PASS 1 or PASS 2. This parameter specifies the maximum number of vias that the Auto-Router can use on a connect-line.

A via is a through-hole on the board that the Auto-Router uses to route a connect-line from one board layer to another.

This parameter specifies the maximum number of vias that the Auto-Router can use on a connect-line. If the connect-line requires more vias than the maximum number specified with this parameter, the Auto-Router will not route the connect-line.

A value of Ø specifies that the Auto-Router cannot use vias. A value of -1 specifies that the Auto-Router can use any number of vias on a connect-line.

P5 VIA GRIDS (DEFAULT VALUE : X=Ø, Y=Ø)

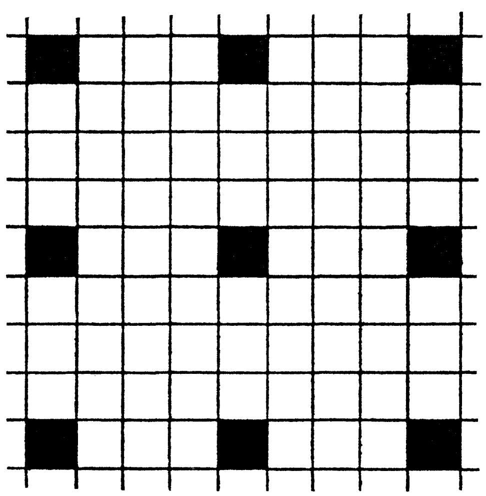
The Auto-Router uses via grids to determine the areas on the board where vias can be placed. This parameter specifies the distance between via grids in both the X and Y directions.

A via is a through-hole on the board that the Auto-Router uses to route a connect-line from one board layer to another. Vias are used by the Auto-Router in PASS 5 to route connect-lines that were not routed in PASS 1 or PASS 2. A via grid is the area on a board where the Auto-Router can place vias. The Auto-Router will place vias on a connect-line only where via and routing grids line up.

This parameter specifies the distance between via grids in both the X and Y directions. The Auto-Router will place via grids uniformly across the board at the distances specified by the operator. The operator can specify a distance between via grids in the X direction that is different than the distance between via grids in the Y direction.

A list of via grids in both the X and Y directions must be specified for each PASS 5 execution. A value of Ø specifies that there are no via grids and allows the Auto-Router to place vias on any routing grid.

The following figure illustrates the placement of via grids on the board when the operator specifies an X and Y via grid of 1ØØ mils. The grid size shown is 25 mils.



NOTES ON SETTING THE ROUTER PARAMETERS

PASS 5 EXECUTIONS

The operator may specify up to 10 executions of PASS 5 with the |NUM P5 EXECUTES| command. Each PASS 5 routing parameter may contain up to 10 values, one for each execution. When setting a PASS 5 parameter, input a comma or blank space between each value.

EXAMPLE: |NUMBER P5 EXECUTES| 3 |ENTER|

|P5 PIN KEEP AWAY| 0, -1 |ENTER|

Result: During the first execution of PASS 5, the system uses a pin keep away value of 0. During the second execution, the system uses a pin keep away value of -1.

NOTE: If the operator specifies 10 executions of PASS 5, and inputs only two values, the system will use the last value for the remaining executions. In the above example, the system would use a pin keep away value of-1 for execution 3.

Refer to the COMMAND DESCRIPTION section of the manual for the exact input sequence for each router parameter.

DISPLAYING AND PRINTING THE ROUTER-CON TEXT FILE

The operator may display and print the ROUTER-CON file. The ROUTER-CON file contains the latest auto-router parameters settings. Use the |LIST| command on the router parameters menu to list the file on the function screen. When the ROUTER-CON file is displayed, the operator may output the file to the matrix printer. Use the |PRINT| command at the top of the function screen to print the ROUTER-CON file.

EXAMPLE: ROUTER-CON TEXT FILE (ROUTER DEFAULT PARAMETERS)

```
GRID SIZE                          :25
HORIZONTAL LAYERS                  : 1
VERTICAL LAYERS                    : 2
VERTICAL COMPONENT ROTATION
DIAGONALS ALLOWED
DEFAULT LINE WIDTH                 :12
STATUS MESSAGE FREQUENCY           :10

ROUTE PASS 1                          ROUTE PASS 2
PASS 1 WINDOW EXPANSION:  5           PASS 2 WINDOW EXPANSION: 6
PASS 1 PIN KEEP AWAY    :  2          PASS 2 PIN KEEP AWAY    : 3

PASS 5 LISTINGS
# EXECUTIONS     :2
```

```
WINDOW EXPAND   :  8     16
PIN KEEP AWAY   :  0     -1
LAYER PAIRS     :  1:2   1:2
JOG SIZES       :  5     5
VIAS ALLOWED    :  YES   YES
ROUTER TYPE     :  BOTH  BOTH
```

6. STARTING THE AUTO-ROUTER

There are four commands used to start the auto-router. These commands allow the operator to selectively route a board drawing.

Each command is described below. However, refer to COMMAND DESCRIPTION section of the manual for the exact input sequence for each command.

|ROUTE BOARD|

This command routes the actively displayed board drawing. |ROUTE BOARD| will attempt to route all non-interconnected nets on the drawing. This command has no effect on interactively placed connections, or connections placed with prior use of the router.

1. Which router will you be using (RSX or CoRouter)?

    Type : : System type (RSX or CoRouter)
    Press : : RETURN or ENTER

2. What are the board dimensions?

    NOTE

    Board dimension is Router Keepin Area dimension.

    Please enter board X value (in INCHES) :

        Type : : Board width (in inches)
        Press : : RETURN or ENTER

    Please enter board Y value (in INCHES) :

        Type : : Board height (in inches)
        Press : : RETURN or ENTER

3. Would you like to specify all the routing layer directions? (Y or N)

    Type : : Y or N
    Press : : RETURN or ENTER

    NOTE

    If Y is specified, the system prompts with questions 4 and 5. If N is specified, the system prompts with question 6.

4. Which are your VERTICAL routing layers? (Use 1-14 only)
   (Please use a space to separate each layer number)

    Vertical routing layers :

        Type : : Layer numbers of vertical routing layers
        Press : : RETURN or ENTER

5. Which are your HORIZONTAL routing layers? (Use 1-14 only)
   (Please use a space to separate each layer number)

    Horizontal routing layers :

        Type : : Layer numbers of vertical routing layers
        Press : : RETURN or ENTER

6. What are your routing layers (use 1-14 only) (please use a space to separate each layer number)?

Routing layers :

Type	: :	Layer numbers of board layers to be routed
Press	: :	RETURN or ENTER

7. Which layer is the component side?

Type	: :	Layer number of the component side
Press	: :	RETURN or ENTER

8. Which direction is the etch going on the component side (H, V, or N)?

Type	: :	H, V, or N (for etch direction)
Press	: :	RETURN or ENTER

9. Which layer is the solder side?

Type	: :	Layer number of the solder side
Press	: :	RETURN or ENTER

10. Which direction is the etch going on the solder side (H, V, or N)?

Type	: :	H, V, or N (for etch direction on solder side)
Press	: :	RETURN or ENTER

11. Are there finger connectors on the edge of the board (Y or N)?

Type	: :	Y or N
Press	: :	RETURN or ENTER

If the operator specifies Y to question 11, the following USER ADVISORY will be displayed after all of the questions have been answered :

```
*************************************************
                 USER ADVISORY

If board contains CONNECTOR FINGERS without
attached through-holes, then
A rectangle (or multiple rectangles, if applicable)
should be placed on layer RL, completely enclosing
the connector and extending to the first row or
column of component pins.

To determine layer RL, use the following logic :

IF the connector is on the TOP or BOTTOM then
      RL = 220 + the horizontal layer that the
      connector is on.
IF the connector is on the LEFT or RIGHT side then
      RL = 220 + the vertical layer that the
      connector is on.
*************************************************
```

- This USER ADVISORY describes Reverse Routing Direction rectangles that can be placed on the board drawing. For more information about Reverse Routing Direction rectangles, refer to subsection 3.

12. What is the primary component orientation (H, V, or N)?

    Type : : H, V, or N (for component orientation)
    Press : : RETURN or ENTER

13. Are the components mostly IC types or discretes (I or D)?

    Type : : I or D (for majority of component type)
    Press : : RETURN or ENTER

14. Do any of the components have pins on 4 sides (Y or N)?

    Type : : Y or N
    Press : : RETURN or ENTER

NOTE

If Y is specified for question 14, the system prompts with question 15 and displays the following USER ADVISORY after all of the questions have been answered. If N is specified for question 14, the system prompts with question 16.

****
USER ADVISORY

If your board contains SMDs without through-holes, then PIN ESCAPES should be attached to each used pin.
****

- A PIN ESCAPE is a via that the operator connects to the Surface Mounted Device (SMD) component-pin when the SMD device is placed on the board.

15. Do most components have pins on 4 sides (Y or N)?

    Type : : Y or N
    Press : : RETURN or ENTER

16. What is the default line width (in MILS)?

    Type : : Value (in mils) of default line width
    Press : : RETURN or ENTER

17. What is the typical IC pad Y and X value?

    Please enter the pad Y value (in MILS)?

    Type : : Value (in mils) of pad height
    Press : : RETURN or ENTER

    Please enter the pad X value (in MILS)?

    Type : : Value (in mils) of pad width
    Press : : RETURN or ENTER

18. What is the distance between PIN 1 and PIN 2 (in MILS) of a typical IC (if there are no ICs, enter 100)?

    Type : : Value (in mils) of the typical distance between pins 1 and 2 on an IC.
    Press : : RETURN or ENTER

    If there are no ICs on the board :

    Type : : 100 mils
    Press : : RETURN or ENTER

19. What is the line-to-line spacing requirement (in MILS)?

Type : : Value (in mils) of the line-to-line spacing specified in PDRC-CON text file
Press : : RETURN or ENTER

NOTE

If the operator has specified a line to line spacing that is small and a linewidth that is small, the system will prompt with the following :

System recommends the following odd grid for routing :

X Grids : X, X, X, X, X
Y Grids : X, X, X, X, X

If odd grid is undesirable, system will route with 25 mil uniform grid. Do you want uniform grid ? (Y/N) :

Enter : : Y
Press : : RETURN or ENTER

The system deletes the odd multiple grid that was recommended and uses a 25 mil uniform grid.

Enter : : N
Press : : RETURN or ENTER

The system will use the recommended odd multiple grid.

CAUTION

Using the odd multiple grid recommended by the system will make it extremely difficult to edit connect-lines routed by the Auto-Router.

20. Do you wish to SAVE_DWG_NEW_REV_AUTO between multiple executions of PASS 5 (Y or N)?

Type : : Y or N
Press : : RETURN or ENTER

After a few minutes, the system prompts :

Analyzing your input data, Please standby!

USER ADVISORIES associated with operator input will be displayed at this time. The following USER ADVISORY will be displayed if the system will use Channel Preference to route the board.

```
**************************************************
                USER ADVISORY

CHANNEL PREFERENCE is used in this route.  Please
make sure that the Router Keepin rectangle should
have its lower left coordinate at a multiple of
100 in both X and Y.
**************************************************
```

All USER ADVISORIES are followed by the prompt :

PLEASE HIT RETURN TO CONTINUE [S] :

Press : : RETURN

After a few minutes, the system will display the following message on the terminal screen :

The system will now replace the old ROUTER-CON or COROUTER-CON text file (if any) with a new one.

The system then prompts :

DO YOU WANT TO CONTINUE [Y/N] :

Enter : : Y
Press : : RETURN or ENTER

The system will create a new version of the ROUTER-CON or COROUTER-CON text file that will replace the existing version (if any). When the new version of the ROUTER-CON or COROUTER-CON text file has been created, the system prompts :

Command completed!

The operator is now able to make selections from the menu displayed on the function screen.

Enter : : N
Press : : RETURN or ENTER

The system will delete the new version of the ROUTER-CON or COROUTER-CON text file and save the existing version (if any). The system prompts :

Existing ROUTER-CON or COROUTER-CON file (if any) is not affected.

The operator is now able to make selections from the menu displayed on the function screen.

5.2.2 ERROR AND WARNING MESSAGES DISPLAYED BY THE INSIGHT PROGRAM

The following error and WARNING messages might be displayed in response to operator input while running the INSIGHT program.

```
*** Your response is not recognized.  Please try again !

*** Layer out of range.  Please try again !

*** Layer must be an integer.  Please try again !

*** Layer cannot be duplicated.  Please try again !

*** Board size must be >0 and <=65 in.  Please try again !

*** Max. 8 routing layers allowed.
    If you wish to automatically route on more than 8 layers,
    please call TELESIS CUSTOMER SUPPORT.
    If not, please try again !

*** Even number of routing layers.  The number of horizontal and
    vertical layers must be the same.
    Please try again !

*** COMPONENT SIDE is not a routing layer.  Please try again !

*** Solder Side and Component Side must be on different routing layers.
    Please try again !

*** SOLDER SIDE is not a routing layer.  Please try again !

*** Warning: Component and Solder side has the same routing
    direction.

*** Line width must be >= 1 mil and <= 250 mils.  Please try again !

*** Pad Y value must be >= 1.  Please try again !

*** Pad X value must be >= 1.  Please try again !

*** Spacing must be >= 1 mil and <= 250 mils.  Please try again !

*** Board has edge connector :
    Both component and solder side must be on routing or
    non-routing layers.  Mixing is not allowed.

*** WARNING :
    Board has edge connector, both component and solder side are
    on non-routing layers.
    Make sure each of the connector fingers has a through-hole on
    it.
```

5.2.3 DISPLAYING AND MODIFYING AUTO-ROUTER PARAMETERS ASSIGNED BY THE INSIGHT PROGRAM

Since the INSIGHT program automatically assigns Auto-Router parameters, check the ROUTER-CON or COROUTER-CON text file to ensure that Auto-Router parameters assigned by the INSIGHT program are acceptable for auto-routing the board.

To call up the ROUTER-CON or COROUTER-CON text file, perform the following steps.

Type : : @TEXT on the terminal keyboard
Press : : RETURN or ENTER

The system prompts :

enter name of text file to be editted :

Type : : ROUTER-CON or COROUTER-CON
Press : : RETURN or ENTER

The system prompts :

you will be editting an existing text file named
ROUTER-CON rev 1
enter RETURN to continue :

Press : : RETURN

The ROUTER-CON text file is displayed on the terminal screen. The example of a ROUTER-CON text file illustrated on the following page lists the default parameter values used by the Auto-Router.

NOTE

Check the routing directions of the board layers to ensure that the horizontal and vertical routing directions of the board layers assigned by the INSIGHT program are acceptable. Ensure that the Auto-Router restrictions detailed in section 4 are set.

The operator can modify Auto-Router parameters by editing the ROUTER-CON or COROUTER-CON text file from the terminal.

CAUTION

If Auto-Router parameter values are changed manually by selecting the parameter command from the SET ROUTER PARMS menu, the ROUTER-CON or COROUTER-CON text file is returned to the standard format and cannot be used by the INSIGHT ROUTER.

The following example shows a ROUTER-CON or COROUTER-CON text file created by the INSIGHT program. Parentheses enclose operator comments to questions asked by the INSIGHT program. Information within parentheses is not visible to the Telesis system, only to the operator.

```
 ( ; INSIGHT Version 1.Ø  Date : May 12, 1986 )
 ( Telesis_router_type  RSXROUTER )
 ( Project_Name ACC-TEST )
 ( board_dimension 4.5Ø   4.3Ø )
 ( routing_layers  1  2 )
 ( edge_connector NO )
 ( component_side_layer  1 )
 ( component_side_direction  HORIZONTAL )
 ( solder_side_layer  2 )
 ( solder_side_direction  VERTICAL )
 ( component_rotation  VERTICAL )
 ( component_mostly_discrete  IC )
 ( any_smd_component  NO )
 ( component_mostly_four_sided  NO )
 ( default_line_width  12 )
 ( typical_pin_size  58   58 )
 ( typical_pin_spacing  1ØØ )
 ( line_line_spacing  12 )
 ( grid_size  25 )
 ( save_dwg_new_rev  NO )
GRID SIZE : 25
HORIZONTAL LAYERS : 1
VERTICAL LAYERS : 2
COMPONENT ROTATION : VERTICAL
DIAGONAL ALLOWED : YES
DEFAULT LINE WIDTH : 12

SKIP PASS 1 : NO
P1 WINDOW EXPANSION : 4
P1 PIN KEEP AWAY : Ø

SKIP PASS 2 : NO
P2 WINDOW EXPANSION : 4
P2 PIN KEEP AWAY : 1

NUMBER P5 EXECUTIONS : 4
P5 PIN KEEP AWAYS : -1  -1  -1  -1
P5 WINDOW EXPANSIONS : 8  2Ø  4Ø  1ØØ
P5 LAYER PAIRS : 1  2  1  2  1  2  1  2
P5 VIAS ALLOWED : YES  YES  YES  YES
P5 VIA LIMITS : 2  4  6  -1
P5 VIA GRID SIZES : Ø  Ø  Ø  Ø  Ø  Ø  Ø  Ø
P5 JOG SIZES : 6  12  2Ø  4Ø
P5 JOG LIMITS : -1  -1  -1  -1
P5 ROUTE TYPE : M  M  M  M
ROUTE_BOARD
 ( To activate the command on next line, remove the parentheses )
 ( SAVE_DRAWING_NEW_REV_AUTO )
```

NOTE

If the operator specified NO to question 2Ø, the last two lines in the ROUTER-CON or COROUTER-CON text file would appear as shown in the previous example. If the operator specified YES to question 2Ø, the parentheses around the last line would have been removed by the INSIGHT program.

To auto-route the board after running the INSIGHT program, perform the following step.

Select : : INSIGHT ROUTER from the ROUTER menu

The system will retrieve the ROUTER-CON or COROUTER-CON text file and read the Auto-Router parameters. When the system encounters the keyword ROUTE_BOARD imbedded in the ROUTER-CON or COROUTER-CON text file, it will automatically route the board using the Auto-Router parameters assigned by the INSIGHT program.

6. STARTING THE AUTO-ROUTER

There are four commands used to start the Auto-Router. These commands allow the operator to selectively route a board drawing. Each command is described below. However, refer to the COMMAND DESCRIPTION section of the manual for the exact input sequence for each command.

ROUTE BOARD

This command routes the actively displayed board drawing and attempts to route all non-interconnected nets on the drawing. This command has no effect on interactively placed connections, or connections placed with prior use of the Auto-Router.

|ROUTE BY WINDOW|

This command routes within a specified window on the active board drawing. Any two-pin connections with either or both pins outside this window will not be routed.

EXAMPLE: |ROUTE BY WINDOW| (P1) (P2) |ENTER|

- o (P 1) is one corner of the rectangular window.
- o (P 2) is the opposite corner of the rectangular window.

|ROUTE BY WINDOW| is useful in routing connections in areas of the board drawing where components are rotated against the majority component orientation. For example, if vertical component orientation is specified in the router parameters, the operator may change the |COMP ORIENTATION| parameter to horizontal, then window-in the area of the drawing to route those connections.

|ROUTE NET|

This command routes operator-specified nets. |ROUTE NET| is useful in routing crucial connections on the drawing prior to routing the entire drawing with the |ROUTE BOARD| command.

EXAMPLE: |ROUTE NET| 4,5,6,7,10 |ENTER|

Result: The system will route nets 4, 5, 6, 7, and 10.

- o The operator may specify the net name and/or the net number. Input a comma between each input.

|ROUTE PIN PAIR|

This command allows the operator to specify and route pin pairs. Crucial pin pair connections may be routed prior to using the |ROUTE BOARD| command to route the entire drawing.

EXAMPLE: |ROUTE PIN PAIR| (P 1) (P 2) |ENTER|

- o (P 1) is one component pin.
- o (P 2) is the second component pin. Use |NEXT| after (P 2) to continue specifying pin pairs.

NOTES : An error message will be displayed on the function screen if a pin pair does not belong to the same net, or if the connection has already been made.

Pin pairs are only routed during PASS 5 using PASS 5 parameters. The system will run PASS 5 the number of times specified by the NUM P5 EXECUTES parameter.

NOTE ON SET-UP FILES

When the operator first uses one of these commands to start the auto-router, a long delay will occur while the system assembles the information needed to operate the router. The system collects this information from the board drawing, net-data-base, pin files, LAYERSTD file, and other prerequisite files.

The system uses this information with the router parameter settings to build set-up files,and a routing map of the board drawing.

Once the set-up files have been created, the system preserves this information. Router start-up commands may be picked from this menu without an additional set-up delay. However, if |DONE| is picked to exit the router menus, the set-up files are deleted and must be rebuilt if the router is to be restarted.

USING THE |CANCEL| COMMAND

The operator may pick |CANCEL| when working on the router menus. If |CANCEL| is picked while the set-up files are being built, the system will delete the set-up files. However, if the set-up files have been built and the auto-router has started, the system will preserve the set-up files. Any completed connections on the drawing will be preserved.

REPEATED USE OF THE ROUTER COMMANDS

The operator may use one of the router start-up commands, repeat it, or use others as needed. This allows the operator to selectively route difficult connections first, then proceed to route the less difficult connections.

CHANGING PARAMETERS BETWEEN USE OF THE ROUTER START-UP COMMANDS

The operator may change parameters settings after using a router start-up command. However, there are three parameters that rebuild set-up files, and the routing map of the board drawing:

|ROUTER GRID SIZE|
|COMP ORIENTATION|
|ROUTING LAYERS|

If any one parameter is reset after using a router start-up command, all or part of the set up files will be rebuilt.

THE BOARD DRAWING DISPLAY DURING ROUTING

The following examples illustrate the routed connections occurring during each PASS. These examples adhere to the system default parameter settings.

GRID SIZE: 25
HORIZONTAL LAYERS : 1
VERTICAL LAYERS : 2
VERTICAL COMPONENT ROTATION
DIAGONALS ALLOWED
DEFAULT LINE WIDTH: 12
STATUS MESSAGE FREQUENCY: 10

ROUTE PASS 1
PASS 1 WINDOW EXPANSION :5
PASS 1 PIN KEEP AWAY :2

NUM P5 EXECUTES: 2
P5 WINDOW EXPAN: 8 16
P5 PIN KEEP AWAY: 0 -1
P5 LAYER PAIRS: 1:2 1:2
P5 JOG SIZES: 5 5
P5 VIAS ALLOWED: YES YES

ROUTE PASS 2
PASS 2 WINDOW EXPANSION :6
PASS 2 PIN KEEP AWAY :3

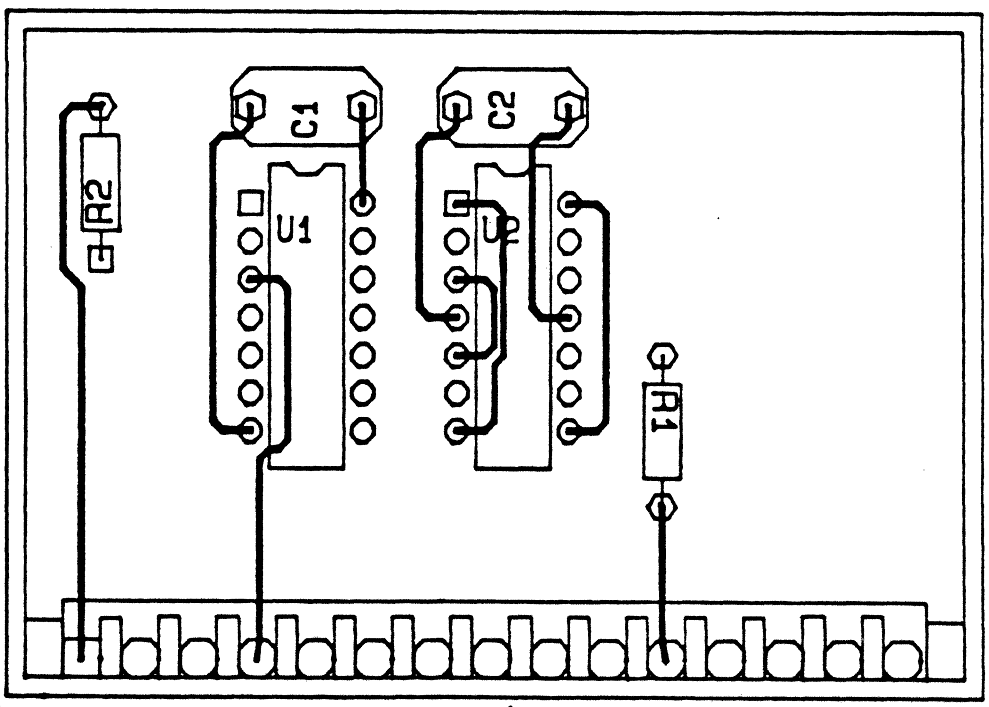
During each PASS, the system does not change existing connections on the board drawing that were routed manually or with prior use of the auto-router.

As the auto-router is running, the system builds a text file called ROUTER-LOG. This file summarizes routing statistics and failed connections during routing.

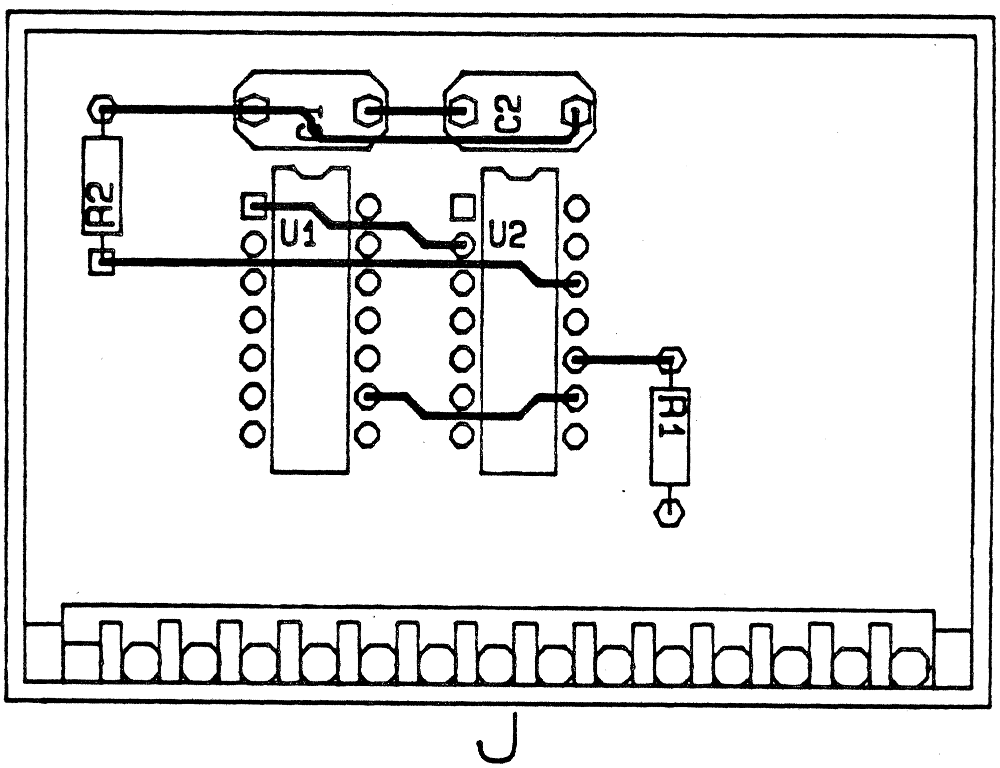
When all routing passes are complete, the system executes the |RATSNEST| command. Ratsnest connect-lines flag the connections that the router failed to complete.

Pick: |ROUTE BOARD|

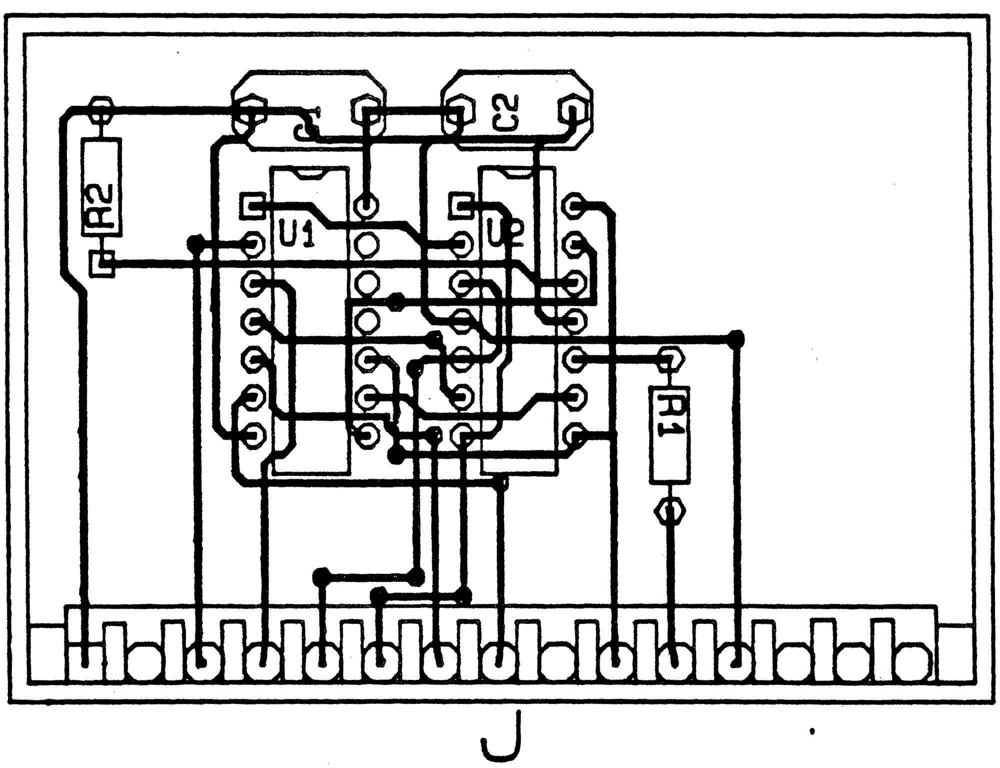
PASS 1 EXECUTION



PASS 2 EXECUTION



PASS 1, PASS 2, PASS 5 EXECUTED



7. THE ROUTER-LOG FILE

The ROUTER-LOG is a text file that summarizes the results of the autorouter. This file opens when the operator picks the |ROUTER| command. It continues to gather information until the operator exits from the router menus.

If the system encounters an error when the router is started, the function screen message line displays:

ERROR IN PROCESSING: CHECK-LOG FILE

Use the |LIST ROUTER-LOG| command to display the file on the function screen. Use the |PRINT ROUTER-LOG| command to print the file on the optional matrix printer.

When |DONE| is picked to exit from the router menus, the system does not delete the ROUTER-LOG. However, when the operator picks |ROUTER| to re-open the router menus, the system deletes the existing ROUTER-LOG, then opens a new ROUTER-LOG file.

EXAMPLE: ROUTER-LOG FILE

```
ROUTER-LOG for project XXX drawing XXX started at 10-Oct-84 10:27:17
elapsed time CMPNET:  0 hr  0 min  11 sec   (0.00 hr)

approximate equivalent IC count = 4

approximate board density = 1.13 sq. in per equivalent IC
elapsed time for PADSIZ:  0 hr  3 min  34 sec   (0.06 hr)

number missing connections:  27 out of 27
connections completion:  0.00%
total length of etch on the board:  0.00 inches
total number of vias on the board:  0
elapsed time for TRCNET:  0 hr  0 min  25 sec   (0.01 hr)

GLOBAL ROUTING PARAMETERS
grid size = 25
routing layers = 1 HORIZONTAL
                 2 VERTICAL
minimum line/line spacing = 13
minimum line/pad spacing  = 13
minimum pad/pad spacing   = 13
via diameter = 55
diagonals allowed
no net with a line width greater than 12 mils will be routed
```

```
ROUTING PARAMETERS FOR PASS 1
window expansion = 5
pin keep away    = 2

pass 1 statistics:  completes = 10   fails = 0
elapsed time for pass 1:  0 hr  0 min  48 sec   (0.01 hr)

ROUTING PARAMETERS FOR PASS 2
window expansion = 6
pin keep away    = 3

pass 2 statistics:  completes = 7   fails = 0
elapsed time for pass 2:  0 hr  0 min  26 sec   (0.01 hr)

ROUTING PARAMETERS FOR EXECUTION 1 OF PASS 5
window expansion = 8
pin keep away    = 0
horizontal routing layer = 1
vertical routing layer   = 2
jog size = 5
vias allowed
router type = BOTH

pass 5 statistics:  completes = 10  fails = 0
elapsed time for pass 5:  0 hr  3 min 8 sec  (0.05 hr)
number missing connections:  0 out of 27
connections completed:  100.00%

total length of etch added by this route:  23.64 inches
total number of vias added by this route:  12
total length of etch on the board:  23.64 inches
total number of vias on the board:  12
number of vias/connections:  0.44
```

8. VIA ELIMINATION (Optional)

When the board drawing has been routed, the operator may eliminate unnecessary vias with the |VIA ELIMINATE| command. When |VIA ELIMINATE| is picked the system attempts to discard as many vias as possible. This command may also be used to eliminate vias on a manually routed drawing.

Use the |SET NO VIA ELIM| command to specify nets (name ornumber) where vias are not to be eliminated. Use the |LIST NO VIA ELIM| command to list the nets specified. |DEL NO VIA ELIM| may be used to delete nets from the list.

Refer to COMMAND DESCRIPTION section of the manual for the exact input sequence for each command.

NOTE: If the operator picks |VIA ELIMINATE| upon entering the router menus, the system will build the set-up files with the usual delay.

9. EDITING MANUALLY

Use the EDIT CONNECTIONS command to interactively add and edit connections. This command allows editing without returning to the DESIGN BOARD menus. (Returning to the DESIGN BOARD menus would delete the router set-up files.)

EXAMPLE : Select EDIT CONNECTIONS from the AUTO ROUTER menu page

The following menu is displayed :

DONE	WORLD	ROAM			STATUS	DISPLAY	CANCEL
EDIT CONNECTIONS							

SET LAYER PAIR	HIGHLIGHT NET	DEHIGHLIGHT NET	ADD CONNECTION
DELETE ELEMENT	HLIGHT NET NUMB	SET TRAP SIZE	SET LINE WIDTH
DELETE VERTEX	MOVE VERTEX	INSERT VERTEX	SET NECK WIDTH
DELETE SEGMENT	MOVE SEGMENT	ZOOM RATIO	CHANGE
DEL NET SECTION	MOVE SYMBOL	ROTATE SYMBOL	LIST ELEMENT

Select : : CHANGE

The following menu is displayed :

DONE	WORLD	ROAM			STATUS	DISPLAY	CANCEL
CHANGE							

CHANGE LAYER	CHANGE WIDTH	CHG LAYR & WIDTH	ADD CONNECTION
CHANGE SEG LAYER	DELETE ELEMENT	SET LAYER PAIR	SET LINE WIDTH
MOVE SECTION	MOVE VERTEX	INSERT VERTEX	SET NECK WIDTH
DELETE SEGMENT	MOVE SEGMENT	ADD RECTANGLE	SET ACTIVE LAYER
DEL NET SECTION	MOVE SYMBOL		LIST ELEMENT

The commands on these menu pages operate identically to those on the DESIGN BOARD menus. However, the NETLOCK parameter is always on and cannot be turned off.

When the operator selects DONE from either menu page, there is a slight delay while the system incorporates all of the editing into the set-up files.

10. AUTO-ROUTER ERROR MESSAGES

ERROR IN PROCESSING, CHECK LOG FILE

This message will occur if the system locates anerror when the router is started. The operator must pick the |LIST ROUTER-LOG| command to display the error on the function screen.

UNABLE TO FIND PDRC-CON

The physical design rules checking file does not exist in the current directory, or in the SYSTEM-LIBRARY. If the file does exist, ensure the file is named PDRC-CON. (See PREREQUISITES TO OPERATING THE AUTO-ROUTER.)

UNABLE TO CREATE ROUTER-CON TEXT FILE
UNABLE TO CREATE NEW ROUTER-LOG FILE
UNABLE TO OPEN PDRC-CON TEXT FILE
UNABLE TO OPEN ROUTER-CON FILE

If any of these messages occur, the operator should check the amount of available disk space on the system. If the available disk space is low, the system will not create or open router files.

ERROR: SAME LAYER SPECIFIED FOR HORIZ & VERT

The operator specified the same layer for horizontal and vertical routing when using the |ROUTING LAYERS| command.

UNABLE TO FIND THE VIA SYMBOL FILE

A via symbol file must exist in the current project, or in the SYSTEM-LIBRARY.

PINS ARE DIFFERENT NETS

When using the |ROUTE PIN PAIR| command, the operator must ensure that the pins to be routed belong to the same net.

PINS ALREADY CONNECTED

This message may appear if the operator selects |ROUTE PIN PAIR| or |ROUTE BY NET|. The pins/nets specified were previously routed.

NO CONNECT FOUND, CONNECT-LINE NOT ADDED

The system will not connect invalid nets, or pin pairs specified with |ROUTE NET| or |ROUTE PIN PAIR|.

UNABLE TO FIND BOTH PINS

When the |ROUTE PIN PAIR| command is selected, the operator must pick both pins.

GENERAL APPROACH TO USING THE AUTO ROUTER

When using the auto router, the operator may select and use any number of routing strategies; routing the entire board, routing by windows, routing by nets, and routing by pin pairs. Since the Telesis system allows the operator to choose routing strategies, auto routing can be selectively controlled by "window", net, or pin pair prior to routing the entire board with the |ROUTE BOARD| command.

For example, the operator may wish to route long I/O connections manually, then route the memory array with |ROUTE BY WINDOW|, then execute the |ROUTE BOARD| command to connect all remaining nets on the drawing.

The operator may selectively change parameter settings prior to executing a particular routing strategy, thus controlling the types of connections occurring on the board during each route (i.e., diagonal corners, jog sizes, etc.).

The PASS 1/PASS 2 routing algorithms are designed to route straight connections in the preferred and non-referred directions to the IC orientation with a 100-mil pad-to-pad spacing limitation. These algorithms operate very quickly due to the straightness of the connections, and the limited routing paths. Any net that cannot be auto routed with a PASS 1/PASS 2 algorithm is a PASS 5 type connection.

The Telesis system contains two cost efficient routing algorithms available in PASS 5, the line probe and the maze runner. Cost is based upon connect line length and the number of vertices needed to route a net within the routing window. If both routing algorithms are specified in the routing parameters, the system first runs the line probe algorithm. This algorithm offers fast high quality routing with the least number of jogs, resulting in the lowest cost.

The maze runner algorithm will attempt to route any connection that the line probe failed to complete. The maze runner maintains the higher completion percentage, however, cost is slightly greater due to the number of jogs required to route the connection. It is recommended that both algorithms be specified with the |P5 ROUTER TYPES|.

After determining a strategy, the operator should select parameters that route connections on the board efficiently, thus allowing channels to remain open for long, complicated PASS 5 type connections. Although this may be difficult on a dense drawing, use the commands on the |EDIT CONNECTIONS| menu, if necessary. Editing connections prior to restarting the router will allow a high completion percentage during PASS 5 connections.

RECOMMENDED ROUTING PARAMETERS

The parameters outlined below are recommended for successful auto routing on many designs. While the parameters affect completion percentage, improper placement has a much greater influence on routing results. Board symbols should be placed so that the connections to be routed (ratsnest lines) are straight horizontal or straight vertical.

The table below lists each recommended parameter based on a router grid size of 25 mils. The router line width plus the line-to-line spacing must be less than or equal to 25 mils.

EXAMPLE:

ROUTER LINE WIDTH	= 12 MILS	
LINE TO LINE SPACING	= 13 MILS	
	25 MILS	(equal to router grid size)

PASS 1 PARAMETERS

WINDOW EXPANSION = 5

KEEP AWAY - 0

PASS 2 PARAMETERS

WINDOW EXPANSION = 3

KEEP AWAY = 3 - if IC's are 300 mils or more apart

2 - if IC's are 200 to 250 mils apart

1 - if IC's are 150 mils apart

0 - if IC's are 100 mils apart or less

PASS 5 PARAMETERS - 2 EXECUTIONS

	EXECUTE 1	EXECUTE 2
WINDOW EXPANSION	10	30
KEEP AWAY	0	-1
JOG SIZE	5	20

The routing parameters outlined above will assist the operator in achieving successful routing. However, the "fine tuning" of these parameters (mostly PASS 5) may accomplish better results on unique boards.

STRATEGIES IN ROUTING SPECIFIC AREAS OF THE BOARD

ROUTING THE MEMORY ARRAY

The memory array below illustrates the versatility of the auto router in controlling the types of connections. By selecting the proper strategy and parameters, the memory array can be easily routed.

Routing Strategy - |ROUTE BY WINDOW|

Since the memory array is typically confined to one area of the board, the |ROUTE BY WINDOW| command may be used. This will keep memory type connections limited to the "windowed-in" area.

The example assumes a router grid of 25 mils, 200-mil component spacing, and vertical component component orientation. PASS 2 routing will accomplish the results illustrated below. However, if the components are greater or less than 200 mils apart, with a grid size other than 25 mils, specific experiments should be run to determine the best parameters.

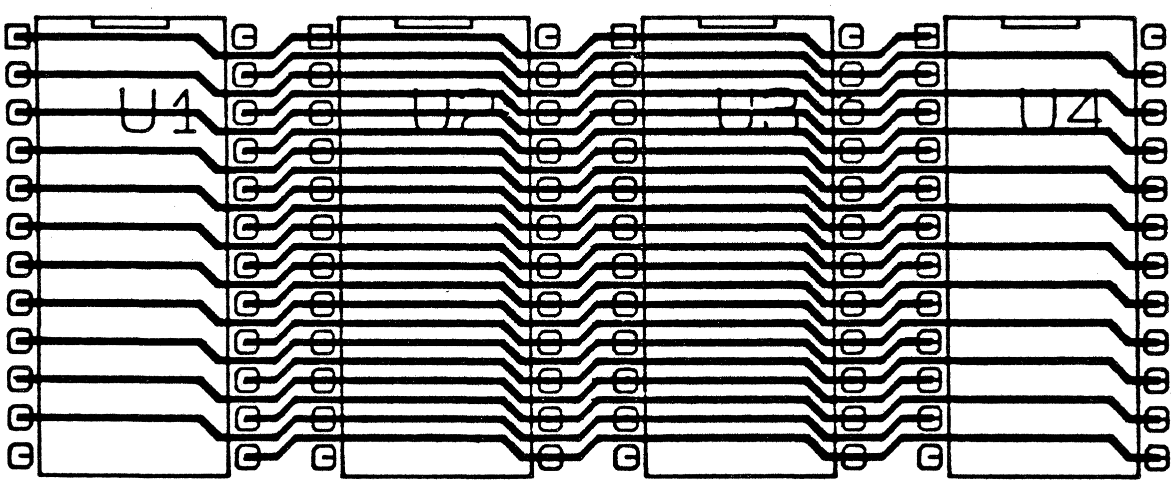
Routing Parameters

DIAGONALS ALLOWED	-	YES
SKIP PASS 1	-	YES
SKIP PASS 2	-	NO
NUM P5 EXECUTES	-	0
P2 WINDOW EXPANSION	-	4
P2 PIN KEEP AWAY	-	2 (based on 200-mil spacing)

Starting the Router

Pick: |ROUTE BY WINDOW| P 1 P 2 |ENTER|

RESULTING MEMORY ARRAY AFTER ROUTE



ROUTING CONNECTIONS TO THE POWER BUSS

It is recommended that power buss routing be accomplished interactively with the |ADD CONNECTION| command prior to starting the auto-router. This allows the operator to add straight connect lines, if desired, with the least number of jogs. Manual routing of the power buss also allows the operator to control the number of available routing channels prior to starting PASS 1/PASS 2/PASS 5 routing.

SHAPES

1. INTRODUCTION TO SHAPES

Shapes represent ground planes, solder masks, component pads, and areas of copper on specific layers of a board drawing. Adding shapes to a board drawing is one of the last steps in the process of designing a board and preparing it for photoplotting.

Certain requirements must be met before the system will display the SHAPES menu and allow the operator to add a shape to a board drawing. Subsection 2 details the prerequisites of adding a a shape to a board drawing.

A shape can only contain a certain number of lines depending on the number of void shapes and void circles within the shape. Physical Design Rules Checking (PDRC) can check shapes for certain violations. Subsection 3 details the limitations of shapes and of PDRC to check shapes for violations.

Creating and filling a shape simulates the photoplotting process. An unfilled shape cannot be photoplotted. After a shape has been added to a board drawing, void shapes or void circles can be added to the shape. Void shapes or circles are areas within the shape that will not fill when a FILL command is selected. Subsection 4 details procedures to create a shape, add void shapes or void circles, and fill the shape.

The system cannot fill any shape that contains errors. When the system attempts to fill a shape, an error log is created. This error log lists any errors that prohibited the system from filling the shape. Subsection 5 describes the error log and error messages that may be listed in the log.

2. PREREQUISITES TO CREATING OR EDITING SHAPES

Before the system will open the SHAPES menu, there must be an APERTURE-TAB text file in the current project or in the SYSTEM-LIBRARY. The APERTURE-TAB text file should contain a variety of line apertures to fill the shape efficiently.

Before creating or editing a shape, it is recommended that :

- The active board drawing be placed and routed
- An unused or unreserved layer be used as a shapes layer so that shapes can be blanked or unblanked from the board drawing without affecting the graphical display of other elements in the board drawing.

3. LIMITATIONS OF SHAPES

The following list summarizes the limitations of shapes used on a board drawing :

- A shape, including void shapes and void circles, cannot contain more than a specific amount of lines. The number of lines allowed in a shape depends on the size of the shape and the number of voids used within the shape. If the operator exceeds this limit while creating the shape, the system displays an error message and the operator must correct the shape before continuing.
- The system will add a shape to the board drawing even if the size or placement of the shape violates a Physical Design Rule.
- The system does not automatically add void shapes or void circles to a shape; the operator must add these elements by using the ADD VOID SHAPE and ADD VOID CIRCLE commands.
- PDRC checks only void shapes for violations, void circles are not checked for violations.
- Shapes can be checked against elements on other board layers for PDRC violations. Refer to POST PROCESSING subsection 3.3, for the procedures to specify PDRC checking between layers.
- A connect line that intersects a shape boundary line will not be flagged as a PDRC violation if the shape has the same net number as the connect line. To assign a net number to a shape refer to POST PROCESSING subsection 3.4.

4. CREATING A SHAPE

The operator adds shapes to a board drawing using commands in the SHAPES menu. Before the system will display the SHAPES menu, there must be an APERTURE-TAB text file in the SYSTEM-LIBRARY or in the current project. Once the shape has been added to a board drawing, the system displays the second page of the SHAPES menu. The second menu page contains commands to add void shapes or void circles to the shape, to edit the shape, or to fill the shape.

The following list summarizes the steps involved in creating and filling a shape.

1. Create the shape on the desired layer of the board drawing.

2. Add void shapes or void circles to the shape (optional).

3. Fill the shape (mandatory for photoplotting).

4. Correct any errors (optional).

The following sections detail the procedures to create a shape, to add void shapes or void circles to a shape, and finally to fill a shape.

4.1 PROCEDURES TO CREATE A SHAPE

To create a shape and add it to an active board drawing, perform the following steps.

NOTES

There must be an APERTURE-TAB text file in the current project or in the SYSTEM-LIBRARY or the system will not open the SHAPES menu. Procedures to create an APERTURE-TAB text file are detailed in the POST PROCESSING section of this manual, subsection 4.

Shapes can only be added to an active board drawing from within the DESIGN BOARD menu.

Select : : SHAPES from the DESIGN BOARD menu

The following system prompt is displayed below the command line of the menu :

PROCESSING APERTURE-TAB TEXT FILE

After a brief interval, the system displays the following menu :

DONE	WORLD	ROAM			STATUS	DISPLAY	

	SET ACTIVE LAYER	SET TRAP SIZE	
ADD SHAPE			EDIT SHAPE
			ADD SHAPE TO NET
LINE LOCK OFF		DELETE ELEMENT	
LINE LOCK 45/ON	LINE LOCK 90/ON		LIST ELEMENT

Select : : SET ACTIVE LAYER

The following system prompt is displayed below the command line of the menu :

Layer :

Enter : : A number to specify which layer of the board drawing the shape will be added to

Press or
Select : : ENTER

Select : : ADD SHAPE

The system flips to graphics mode :

Select : : Points on the graphics screen to specify the shape outline (the shape outline must begin and end at the same point)

The system highlights the shape outline as it is being drawn and rubberbands the cursor. When the shape outline is completed :

Select : : ENTER from the command line of the menu

The system adds the shape to the active layer of the board drawing and displays the following menu :

	WORLD	ROAM			STATUS	DISPLAY	CANCEL

ADD VOID SHAPE	ADD VOID CIRCLE		SET TRAP SIZE
DELETE ELEMENT		XHATCH PARAMS	
DELETE VERTEX	MOVE VERTEX	INSERT VERTEX	FILL CROSSHATCH
DELETE SEGMENT	MOVE SEGMENT		FILL SOLID
	MOVE SECTION		LIST ELEMENT

To exit the SHAPES menu without filling the shape :

Select : : CANCEL from the command line of the menu

The system displays the first menu page and the following warning message is displayed below the command line of the menu :

WARNING : UNFILLED SHAPES WILL NOT BE PHOTOPLOTTED.

Select : : DONE from the command line of the menu

The shape is added to the specified layer of the board drawing. The DESIGN BOARD menu is displayed and the operator can continue to design the board by making additional selections from the menu.

To add a void shape or void circle to the shape, proceed to subsection 4.2 or 4.3, respectively.

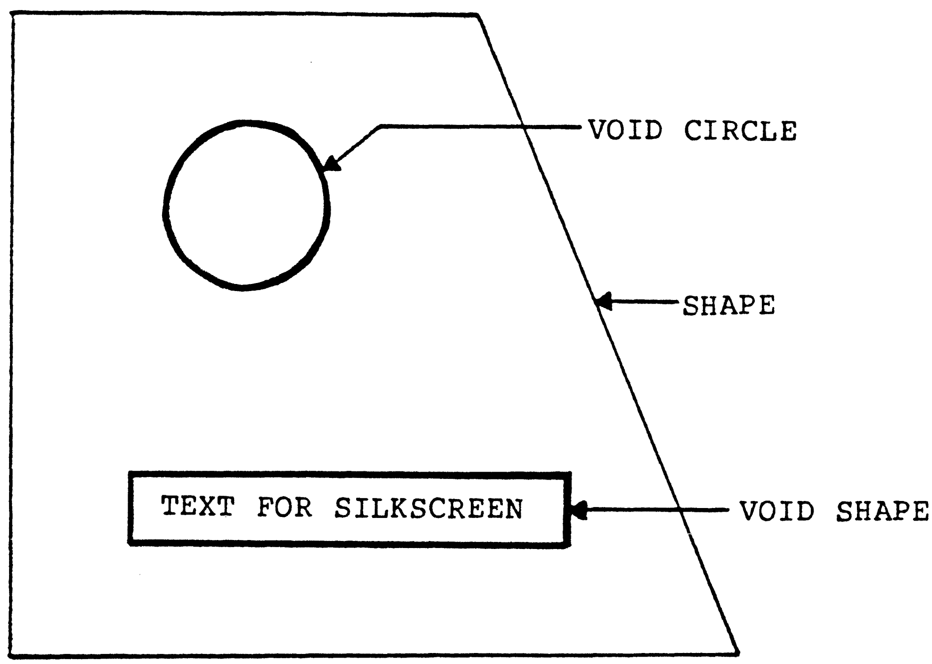
To fill the shape, proceed to subsection 4.4.

To edit an existing shape, proceed to subsection 4.5.

4.2 ADDING VOID SHAPES TO SHAPES

A void shape is an area within a shape that will not fill when the shape is filled. Void shapes provide clearance for component pads, text strings, connect-lines, or any other type of element within a shape.

The following figure illustrates a void shape within a shape.



To add a void shape to a shape, perform the following steps.

From the second page of the SHAPES menu :

Select : : ADD VOID SHAPE

The system flips to graphics mode :

Select : : Points on the graphics screen to specify the void shape outline (the void shape outline must begin and end at the same point)

The system highlights the void shape outline as it is being drawn and rubberbands the cursor. When the void shape outline is complete :

Select : : ENTER from the command line of the menu

The system adds the void shape to the shape and redisplays the menu. The operator can make additional selections from the second menu page or select DONE from the command line to display the first menu page of the SHAPES menu.

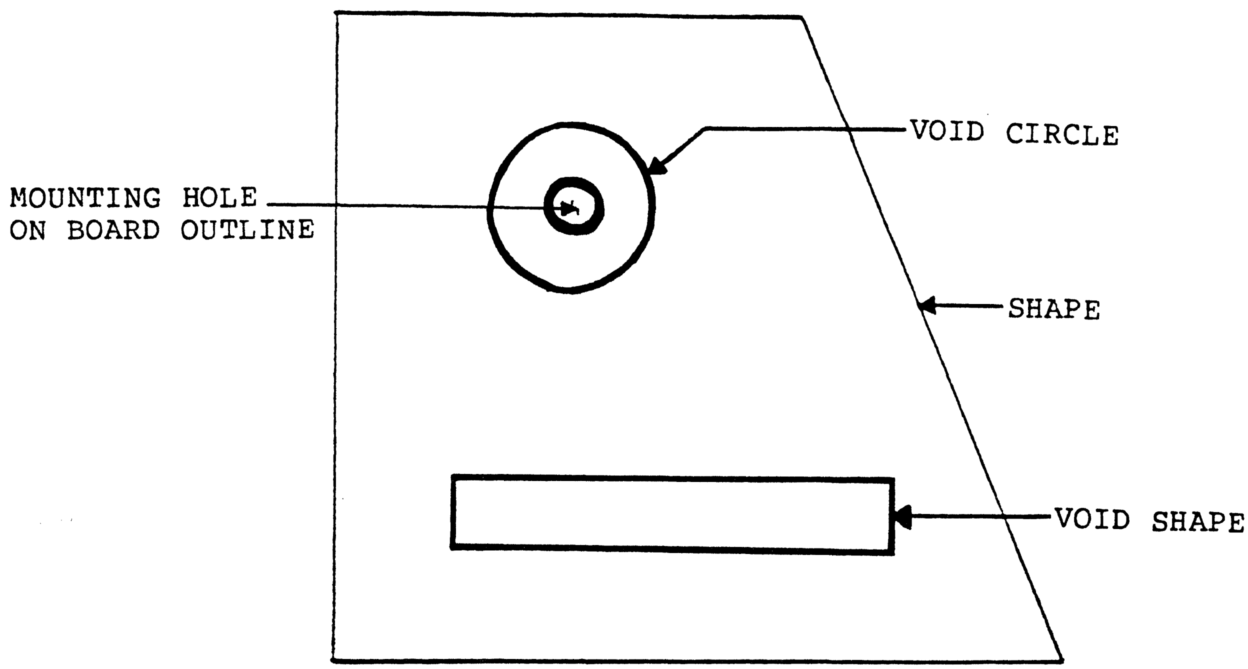
4.3 ADDING A VOID CIRCLE TO A SHAPE

A void circle is a circular area within a shape that will not fill when the shape is filled. Void circles provide clearance for component pads, mounting holes, or any other type of element within a shape.

NOTE

Void circles are not checked for PDRC violations.

The following figure illustrates a void circle within a shape.



To add a void circle to a shape, perform the following steps.

From the second page of the SHAPES menu :

Select : : ADD VOID CIRCLE

The system flips to graphics mode :

Select : : Two points on the graphics screen to specify the center and the radius of the circle, respectively

The system displays a highlighted circle at the location specified by the operator :

Select : : ENTER from the command line of the menu

The system adds the void circle to the shape and redisplays the menu. The operator can make additional selections from the second menu page or select DONE from the command line to display the first menu page of the SHAPES menu.

4.4 FILLING A SHAPE

Filling a shape simulates the photoplotting process and is a prerequisite to photoplotting a shape. If a shape can be filled, it can be photoplotted. If the operator exits the SHAPES menu without filling a shape, a warning message is displayed on the screen and the unfilled shape will not be photoplotted.

The second page of the SHAPES menu contains two FILL commands. FILL SOLID fills the shape completely, excluding void shapes or void circles. FILL XHATCH fills the shape, excluding void shapes or void circles, with a crosshatch pattern defined by crosshatch parameters.

NOTE

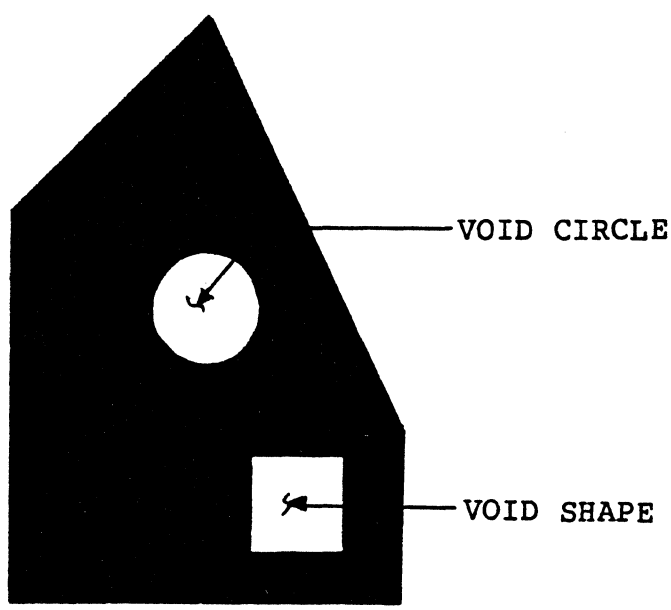
> If the shape boundary intersects itself or a void shape or void circle within the shape, the shape will not be filled and an error message will be displayed in the error log.

FILL SOLID

To fill a shape completely, perform the following.

Select : : FILL SOLID from the second page of the SHAPES menu

The shape is completely filled as illustrated below.



SOLID FILL

FILL CROSSHATCH

A crosshatch fill is a series of parallel lines that are an equal distance from each other. Crosshatch parameters specify the linewidth, distance between, and direction of crosshatch lines. Before selecting crosshatch fill, the operator can set crosshatch parameters to specify the desired crosshatch pattern.

To set the Crosshatch parameters, perform the following steps.

NOTE

Default crosshatch linewidth and spacing is Ø. The default crosshatch pattern is horizontal and vertical.

Select : : XHATCH PARAMS from the SHAPES menu

The following menu is displayed :

```
-----------------------------------------------------------------------
| DONE   | WORLD | ROAM  |        | LIST   | STATUS | DISPLAY |        |
-----------------------------------------------------------------------
|                                                                      |
-----------------------------------------------------------------------
|                  |               |                  |                |
| XHATCH LIN WIDTH |               | XHATCH LIN SPACE |                |
-----------------------------------------------------------------------
|                  |               |                  |                |
|                  |               |                  |                |
-----------------------------------------------------------------------
|                  |               |                  |                |
| HORZ XHATCH      | VERT XHATCH   | HRZ & VERT XHATCH|                |
-----------------------------------------------------------------------
|                  |               |                  |                |
| /// XHATCH       | \\\ XHATCH    | XXX XHATCH       |                |
-----------------------------------------------------------------------
|                  |               |                  |                |
|                  |               |                  |                |
-----------------------------------------------------------------------
```

NOTE

The APERTURE-TAB text file must contain an aperture that is the same size as the crosshatch line width specified or the system will not accept the entry and the following error message will be displayed below the command line of the menu :

CROSSHATCH LINEWIDTH NOT FOUND IN APERTURE-TAB.

Select : : XHATCH LIN WIDTH

The system prompts :

WIDTH :

Enter : : A number to specify the line width of crosshatch lines
Press or
Select : : ENTER

Select : : XHATCH LIN SPACE

The system prompts :

SPACING :

Enter : : A number to specify the spacing between crosshatch lines
Press or
Select : : ENTER

Select : : The desired crosshatch pattern command

No system prompt appears :

Select : : DONE from the command line of the menu

The system displays the second page of the SHAPES menu.

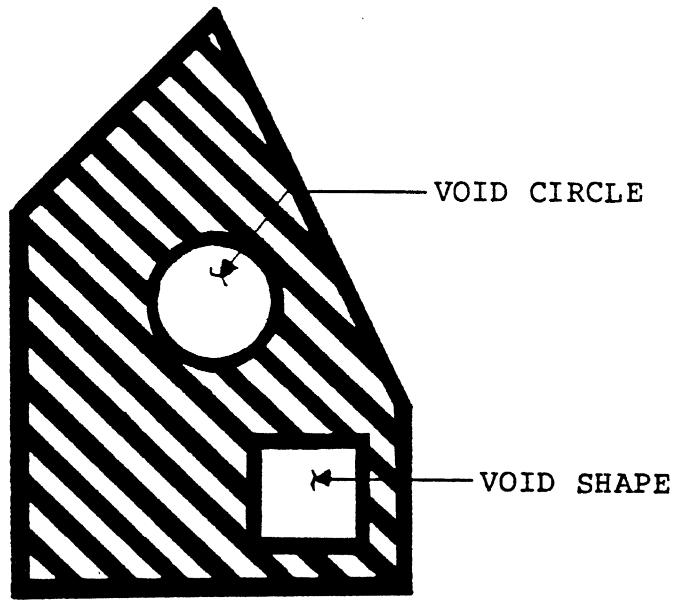
Select : : FILL CROSSHATCH

The shape is filled with the crosshatch pattern defined by the operator or the default pattern assigned by the system.

NOTE

Lines representing the shape outline or the outline of voids within the shape are filled at the crosshatch line width.

The following figure illustrates a shape containing a void shape and void circle that has been filled with a crosshatch fill.



CROSSHATCH FILL

4.5 EDIT SHAPE

To edit a shape that has previously been added to a board drawing, perform the following steps from within the DESIGN BOARD menu.

Select : : SET ACTIVE LAYER from the first page of the SHAPES menu

The system prompts :

Layer :

Enter : : Number to specify the layer that the shape is on

Press or
Select : : ENTER

Select : : EDIT SHAPE

The system flips to graphics mode :

Select : : The shape to be edited on the graphics screen

The shape and any voids within the shape are highlighted on the graphics screen :

Select : : ENTER from the command line of the menu

The shape and any voids within the shape are dehighlighted on the graphics screen and any fill within the shape is removed. The second page of the SHAPES menu is displayed.

Use the commands on the second menu page to change the shape or any voids within the shape. The COMMAND DESCRIPTION section of this manual details the procedures to use each command.

NOTE

To specify a void circle within a shape, you must use the FIND ARC command on the KEYPAD menu.

5. ERROR MESSAGES

Errors that prohibit the system from filling a shape are listed in the shapefill error log. The shapefill error log is displayed on the screen when the system attempts to fill a shape that contains an error.

The following figure illustrates the format of the shapefill error log.

```
                    SHAPEFILL ERROR LOG

   PROJECT:   TEST        1
   DRAWING:   PCB         1

************************************************************
            Butterflies on layer 155 show fill errors
------------------------------------------------------------
            Illegal Shape:  shape intersects with itself
                        at X 25    ,    Y 8
            ------------------------------------------
   ***        PICK                                      ***
   ***          "PAGE->"  to edit this shape.           ***
   ***          "CANCEL"  to exit without filling shape.***
   *********************************************************
```

Error messages that may appear in the shapefill error log are listed below.

ILLEGAL SHAPE : SHAPE INTERSECTS WITH ITSELF AT X, Y

The shape outline intersects itself or a void outline. A void outline that intersects itself will also generate this error. Edit the shape so that there are no intersections between void outlines or shape outlines.

SMALLEST APERTURE WILL NOT FIT IN SHAPE AT X, Y

The smallest line width in the APERTURE-TAB text file is too large for the X and Y coordinate listed. Increase the size of the smallest area of the shape or edit the APERTURE-TAB text file so that it contains an aperture that is smaller than the smallest aperture presently in the file.

THERE ARE TOO MANY EDGES IN THIS SHAPE. ONLY XXX EDGES ARE ALLOWED. THE NUMBER OF EDGES MUST BE REDUCED.

Too many lines were used to create the shape and any voids within the shape. Reduce the number of lines in the shape or the number of voids in the shape.

CANCEL DETECTED. SHAPEFILL ABORTED!

The system started to fill the shape and the operator selected CANCEL before the fill was completed. Select a FILL command and do not select CANCEL before the system has filled the shape.

THE SHAPE HAS TOO MANY FILL LINES TO PUT INTO THE DATABASE. REDUCE EITHER THE NUMBER OF EDGES OR THE NUMBER OF VOIDS.

There are too many lines required to fill the shape. The system does not have that much room in the database. Reduce the size of the shape or edit the APERTURE-TAB text file so that it contains an aperture that is larger than the largest aperture presently in the file.

CROSSHATCH LINEWIDTH EXCEEDS NARROWEST PART OF SHAPE.

The linewidth specified for the crosshatch fill is too large to fill the narrowest part of the shape. Reduce the size of the crosshatch linewidth or increase the size of the smallest area of the shape.

ILLEGAL CROSSHATCH LINE SPACING (Ø - 32767 ONLY).

The line spacing specified for the crosshatch fill is either less than Ø or more than 32767. Change the crosshatch line spacing parameter so that is an allowable value.

POST PROCESSING

- THE NET COMPARE REPORT
- PHYSICAL DESIGN RULES CHECKING
- ARTWORK
- NC DRILL

THE NET COMPARE REPORT

1. USING THE ¶NET COMPARE¶ COMMAND
2. LIST OF DESIGN DRAFTING ERRORS
3. LIST OF EXTRA CONNECTIONS
4. LIST OF MISSING CONNECTIONS
5. LIST OF UNASSIGNED COMPONENTS
6. REPORT SUMMARY

1. USING THE |NET COMPARE| COMMAND

PURPOSE

The |NET COMPARE| command is an aid in routing your board.

As you place connections on the board, you may pick |NET COMPARE|. to compare the nets on the board to the net data base.

When you pick |NET COMPARE|, the system prepares a 5 part report:

1. SUMMARY OF DESIGN DRAFTING ERRORS
2. EXTRA CONNECTIONS ON PCB
3. MISSING CONNECTIONS ON PCB
4. COMPONENTS NOT ASSIGNED ON PCB
5. SUMMARY OF THE NET COMPARE REPORT

You may use the |NET COMPARE| command as often as you like while routing your board. Each time you repeat the command, the existing NET COMPARE REPORT is deleted and an up-to-date one is created.

PREREQUISITES

1. The board must be the active drawing.

This is the only prerequisite for |NET COMPARE|. However, each part of the report has its own prerequisites if it is to be useful to you. (See below under the description of each part of the report.)

HOW TO USE THE COMMAND

The Command Description Section of the Manual tells you how to use the |NET COMPARE| command.

2. LIST OF DESIGN DRAFTING ERRORS

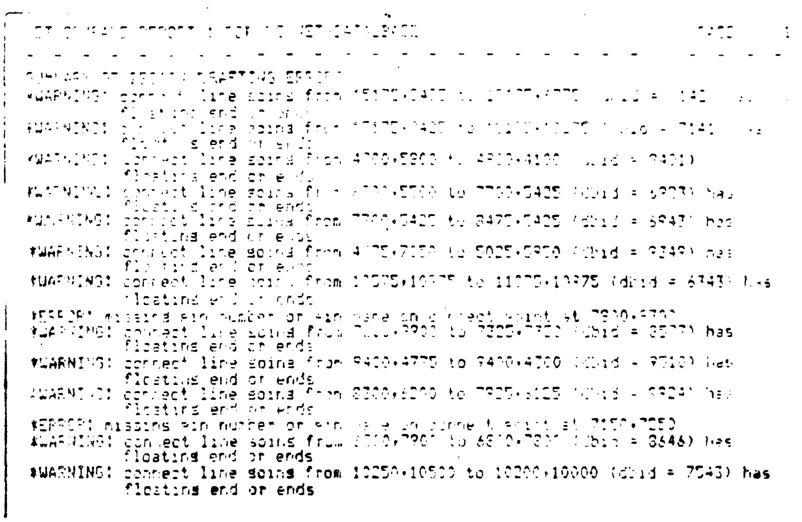
PURPOSE

When you pick the |NET COMPARE| command, the system checks your board in the same way that it checks your schematic when you pick |EXTRACT NETLIST|.

The results of this checking are listed in this part of the NET COMPARE REPORT.

As when you pick |EXTRACT NETLIST|, the system highlights design drafting errors on the graphics screen.

EXAMPLE OF THE LIST



```
*WARNING: connect line going from 4700,5900 to 4900,4100 (dbid = 9401)
          floating end or ends
*WARNING: connect line going from 7700,5400 to 8470,5425 (dbid = 6947) has
          floating end or ends
*WARNING: connect line going from 4175,7050 to 5025,5900 (dbid = 9349) has
          floating end or ends
*WARNING: connect line going from 10575,10775 to 11075,10975 (dbid = 6743) has
          floating end or ends
*ERROR: missing pin number or pin name on connect point at 7930,4700
*WARNING: connect line going from [illegible] to 7325,7700 (dbid = 3577) has
          floating end or ends
*WARNING: connect line going from 9400,4775 to 9400,4700 (dbid = 9712) has
          floating end or ends
*WARNING: connect line going from 8300,6200 to 7925,6125 (dbid = 9924) has
          floating end or ends
*ERROR: missing pin number or pin name on connect point at 7150,7250
*WARNING: connect line going from [illegible] to 6810,7900 (dbid = 3646) has
          floating end or ends
*WARNING: connect line going from 10250,10500 to 10200,10000 (dbid = 7543) has
          floating end or ends
```

PREREQUISITES

1. The board must be the active drawing.

Otherwise, there are no special prerequisites for this part of the report. You need not have completed the annotation or routing of your board. But, of course, you will receive a long list of missing reference designators and missing pin numbers if you have not yet placed these annotations on your board.

3. LIST OF EXTRA CONNECTIONS

PURPOSE

For this part of the report, the system compares your board with the net data base to see if there are any connections on your board that do not exist in the net data base. If there are, they are listed in this part of the NET COMPARE REPORT.

The system looks only at board connections you have made to components with assigned reference designators (ones on which you have used the |ASSIGN REF DES| command). Any other connections you have made on the board are ignored by the system in this part of the report.

EXAMPLE OF THE LIST

```
                                                       17-JAN-82  1?:
NET COMPARE REPORT : PCB TO NET-DATA_BASE                         PAGE
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 EXTRA CONNECTIONS ON PCB
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
           NET: #18 AT Z1.3
        TO NET: #20 AT RP1.5
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
           NET: #19 AT Z1.4
        TO NET: #?????? AT RP1.4
 SCHEMATIC DOES NOT REFERENCE RP1.4 IN  NET: #19
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
           NET: #17 AT Z1.10
        TO NET: #19 AT RP1.3
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

PREREQUISITES

1. The board must be the active drawing.
2. There must be a net data base file in the current project file.
3. You must have used the |ASSIGN REF DES| command on some of the board components.
4. You must have placed some connections on the board.

4. LIST OF MISSING CONNECTIONS

PURPOSE

For this part of the report, the system compares your board with the net data base to see if there are any connections in the net data base that have not yet been made on the board. If there are, they are listed in this part of the NET COMPARE REPORT.

The system looks only at net data base connections to components with assigned reference designators (ones on which you have used the |ASSIGN REF DES| command). Any other connections in the net data base are ignored by the system in this part of the report.

The system does not include TESTPOINTS in the list of unconnected pins.

EXAMPLE OF THE LIST

```
                                                        [illegible]
NET COMPARE REPORT : PCB TO NET-DATA BASE                     PAGE   1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 MISSING CONNECTIONS ON PCB
 NET#      NET NAME        PIN
- - - - - - - - - - - - - Z23.4 - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
95                        TC6.4
                          Z20.1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
98                        C15.1
                          R27.1
                          Z23.11
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
99                        C14.1
                          Z23.5
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
100        X2             C12.2
                          J3.2
                          R20.2
                          RP2.4
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
101        X0             C10.2
                          J3.4
                          R22.2
                          RP2.2
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
102        Y2             J3.6
                          RP2.8
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
103        Y0             J3.8
                          R26.2
                          RP2.6
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
104                       C8.2
                          R21.1
                          Z22.11
                          Z22.12
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
105                       C6.2
                          R19.1
                          Z22.1
                          Z22.4
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
107        BEEPER         R29.1
                          TC6.20
```

PREREQUISITES

1. The board must be the active drawing.
2. There must be a net data base file in the current project file.
3. You must have used the |ASSIGN REF DES| command on some of the board components.

5. LIST OF UNASSIGNED COMPONENTS

PURPOSE

For this part of the report, the system compares your board to the net data base to see if there are any reference designators in the net data base that you have not yet linked to board components using the |ASSIGN REF DES| command. If there are, they are listed in this part of the report.

EXAMPLE OF THE LIST

```
                                                                  1[illegible]-JAN-82  16:
NET COMPARE REPORT : PCB TO NET-DATA_BASE                                       PAGE
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
COMPONENTS NOT ASSIGNED ON PCB => MISSING CONNECTIONS
PARTNUMBER              REF DES    PIN       NET#     NET NAME
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
IOPIN                   J1         16        1        GND2
IOPIN                   J1         15        2        -10V
IOPIN                   J1         14        16
IOPIN                   J1         9         15
IOPIN                   J1         13        14
IOPIN                   J1         12        13
IOPIN                   J1         11        12
IOPIN                   J1         10        11
IOPIN                   J1         8         10
IOPIN                   J1         7         9
IOPIN                   J1         6         8
IOPIN                   J1         5         7
IOPIN                   J1         4         6
IOPIN                   J1         3         5
IOPIN                   J1         2         4
IOPIN                   J1         1         3
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
RPACK                   RP         4         17
RPACK                   RP         1         2        -10V
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

PREREQUISITES

1. The board must be the active drawing.
2. There must be a net data base file in the current project file.

6. REPORT SUMMARY

PURPOSE

At the end of the NET COMPARE REPORT, the system lists a summary of:

- EXTRA CONNECTIONS
- MISSING CONNECTIONS
- UNASSIGNED COMPONENTS

EXAMPLE

```
3 EXTRA CONNECTIONS ON PCB DETECTED
55 MISSING CONNECTIONS ON PCB DETECTED
2 MISSING/UNASSIGNED COMPONENTS ON PCB
```

PHYSICAL DESIGN RULES CHECKING (DRC)

1. OVERVIEW OF PHYSICAL DRC

2. USING THE PHYSICAL DRC COMMANDS

3. THE VIOLATIONS REPORT

1. OVERVIEW OF PHYSICAL DRC

PURPOSE

The physical design rules checking capability aids you in routing your board.

When you use the DRC commands, the system checks your board for these design rule violations:

- Line-to-line spacing violations
- Line-to-pad spacing violations
- Pad-to-pad spacing violations
- Tracewidths that are too narrow

STEPS IN THE PHYSICAL DRC PROCESS

STEP 1 Use the |INPUT DRC RULE| command to input the rules you want the system to use when it checks your board.

STEP 2 Use the |DESIGN RULE CHK| command to instruct the system to check your board and create a violations report text file, and to flag violations on the graphics screen.

PREREQUISITES FOR PHYSICAL DRC

1. The board drawing file must be the active drawing file.
2. There must be a pin description text file for every symbol on the board. These pin files must be in the current project file or in the SYSTEM-LIBRARY.
3. There must be a LAYERSTD (LAYER STANDARD) file for this board in the current project file or in the SYSTEM-LIBRARY.

2. USING THE PHYSICAL DRC COMMANDS

|INPUT DRC RULE|

|DESIGN RULE CHK|

The Command Description Section of the manual tells you how to use these commands. Below are some general principles to keep in mind as you use them.

|INPUT DRC RULE| When you use this command, the system prompts you for the layers you want checked, and the minimum allowable widths you want to establish as rules.

LAYERS - Usually you input all the layers on which you have placed connect lines. NOTE that the system checks each layer individually. It will not check for violations between lines on one layer and lines on another layer. This means that if you have placed connect lines on two different system layers that will eventually be plotted on the same physical layer, violations between these lines will not be checked.

VALUES - Input a value for each system prompt: line-to-line space, line-to-pad space, pad-to-pad space, minimum tracewidth.

DEFAULT VALUES - If you respond to a prompt by picking |ENTER| instead of a value, the system will use zero for that value.

|DESIGN RULE CK| You must use the |INPUT DRC RULE| command before using this command. If you do not, the system will display an error message that the rules file is missing.

The system will not perform design rules checking if any pin files are missing; instead, it will display an error message.

The system flags violations in the drawing display with an "X" mark. When you save the drawing, these marks are saved with it.

When you pick |DESIGN RULE CHK|, the system takes approximately 1/2 hour to 1 hour to check your board depending on its size.

REPEATING THE DRC COMMANDS

It is your option to use the DRC commands after you have finished routing your board, or to use them more than once as you work on routing your board.

You may repeat the DRC commands as often as you like.

Each time you repeat the $\boxed{\text{INPUT DESIGN RULE}}$ command, the previous rules file will be deleted unless you give the new file a different name.

Each time you repeat the $\boxed{\text{DESIGN RULE CHK}}$ command, the previous violations report file will be deleted unless you give the new file a different name. The previous "X" marks will also be deleted.

3. THE VIOLATIONS REPORT

The violations report is a text file created by the system listing the following information:

- RULES
- VIOLATIONS
- NUMBER OF VIAS ON THE BOARD
- TOTAL NUMBER OF VIOLATIONS ON EACH LAYER

EXAMPLE

```
PHYSICAL DRC VIOLATIONS                      date=04-MAY-82  time=20:20:27

Spacing checks will be performed
     min LIN  to LIN  spacing=     10
     min LIN  to PAD  spacing=     10
     min PAD  to PAD  spacing=     10

Minimum trace width will be checked.  Min value= 12


      HORIZONTAL TO VERTICAL LINE FAULTS

line(width= 12   ) at  7225 ,  8850  to  9100 ,  8850  in violation
     with line(width= 12   ) at  9100 ,  8800  to  9100 ,  8850
          actual spacing = 0        (min=  10)
None


      HORIZONTAL TO 45 DEGREE LINE FAULTS

None


      VERTICAL TO 45 DEGREE LINE FAULTS

None


         PAD TO PAD FAULTS

None


      HORIZONTAL LINE TO PAD FAULTS

line(width= 12   ) at  9675 .  7500  to  9725 ,  7500   in violation
     with via pad at  9675 ,  7500
          actual spacing = 0        (min=  10)
line(width= 12   ) at 10250 ,  8650  to 10425 ,  8650   in violation
     with component pad on Q3        pin 2
          actual spacing = 0        (min=  10)
line(width= 12   ) at 10325 ,  8350  to 10350 ,  8350   in violation
     with component pad on Q4        pin 1
          actual spacing = 0        (min=  10)
line(width= 12   ) at 10250 ,  8650  to 10425 ,  8650   in violation
     with component pad on Q3        pin 1
          actual spacing =     4    (min=  10)
line(width= 12   ) at 11000 ,  6600  to 11050 ,  6600   in violation
     with via pad at 11050 ,  6600
          actual spacing = 0        (min=  10)
line(width= 12   ) at 12825 .  5950  to 12875 ,  5950   in violation
     with via pad at 12875 ,  5950
          actual spacing = 0        (min=  10)
```

HORIZONTAL LINE TO PAD FAULTS

```
line(width= 12    ) at  7225 ,  8850  to   9100    8850    in violation
     with via pad at  7100 ,  8850
          actual spacing = 0        (min=  10)
line(width= 12    ) at  7150 ,  6700  to   9850 ,  6700    in violation
     with component pad on S3          pin 1
          actual spacing = 0        (min=  10)
line(width= 12    ) at  9950 ,  8800  to  10150 ,  8800    in violation
     with component pad on D14         pin 1
          actual spacing = 0        (min=  10)
line(width= 12    ) at 10250 ,  7075  to  10400 ,  7075    in violation
     with component pad on R3          pin 2
          actual spacing = 0        (min=  10)
line(width= 12    ) at  9800 ,  6525  to  10825 ,  6525    in violation
     with component pad on C3          pin 2
          actual spacing =      4   (min=  10)
line(width= 12    ) at 10850 ,  6525  to  11050 ,  6525    in violation
     with via pad at 11050 ,  6500
          actual spacing = 0        (min=  10)
line(width= 12    ) at 11050 ,  6525  to  11100 ,  6525    in violation
     with via pad at 11050 ,  6500
          actual spacing = 0        (min=  10)
line(width= 12    ) at 12700 ,  7350  to  13150 ,  7350    in violation
     with component pad on R12         pin 1
          actual spacing =      4   (min=  10)
line(width= 31    ) at 13300 ,  5875  to  13350 ,  5875    in violation
     with via pad at 13300 ,  5875
          actual spacing = 0        (min=  10)
line(width= 12    ) at 13850 ,  6000  to  13950 ,  6000    in violation
     with component pad on J4          pin 3
```

VERTICAL LINE TO PAD FAULTS

```
line(width= 12    ) at  8550 ,  7600  to   8550 ,  7650    in violation
     with via pad at  8550 ,  7650
          actual spacing = 0        (min=  10)
line(width= 12    ) at  9350 ,  8525  to   9350 ,  8850    in violation
     with via pad at  9350 ,  8850
          actual spacing = 0        (min=  10)
line(width= 31    ) at  9775 ,  5675  to   9775 ,  6075    in violation
     with component pad on S4          pin 1
          actual spacing =      7   (min=  10)
line(width= 31    ) at  9775 ,  5675  to   9775 ,  6075    in violation
     with component pad on S4.         pin 4
          actual spacing =      7   (min=  10)
line(width= 12    ) at  9850 ,  6600  to   9850 ,  6700    in violation
     with component pad on S3          pin 1
          actual spacing =      7   (min=  10)
line(width= 12    ) at  9950 ,  8650  to   9950 ,  8800    in violation
     with component pad on D14         pin 1
          actual spacing =      4   (min=  10)
line(width= 12    ) at 10250 ,  7075  to  10250 ,  7150    in violation
     with component pad on R3          pin 2
          actual spacing =      4   (min=  10)
line(width= 12    ) at 11050 ,  6500  to  11050 ,  6525    in violation
     with via pad at 11050 ,  6500
          actual spacing = 0        (min=  10)
line(width= 12    ) at 11400 ,  8225  to  11400 ,  8600    in violation
     with component pad on Z3          pin 12
          actual spacing = 0        (min=  10)
line(width= 12    ) at 11700 ,  8150  to  11700 ,  8600    in violation
     with component pad on Z3          pin 9
          actual spacing = 0        (min=  10)
line(width= 12    ) at 11800 ,  8125  to  11800 ,  8600    in violation
     with component pad on Z3          pin 8
          actual spacing = 0        (min=  10)
line(width= 12    ) at 11900 ,  6400  to  11900 ,  6475    in violation
     with component pad on Z12         pin 6
          actual spacing = 0        (min=  10)
line(width= 12    ) at 11900 ,  6750  to  11900 ,  6800    in violation
     with component pad on Z10         pin 9
          actual spacing = 0        (min=  10)
line(width= 12    ) at 12850 ,  5050  to  12850 ,  5325    in violation
     with component pad on R1          pin 2
          actual spacing =      4   (min=  10)
line(width= 12    ) at 13150 ,  7350  to  13150 ,  7400    in violation
     with component pad on R12         pin 1
          actual spacing = 0        (min=  10)
line(width= 31    ) at 13350 ,  5500  to  13350 ,  5875    in violation
     with via pad at 13300 ,  5875
          actual spacing =      4   (min=  10)
```

45 DEGREE LINE TO PAD FAULTS

```
line(width= 12    ) at 13450 ,  6000  to  13479 ,  5987    in violation
     with component pad on J4          pin 3
          actual spacing = 0        (min=  10)
```

TRACEWIDTH FAULTS FOR HORIZONTAL LINES

None

TRACEWIDTH FAULTS FOR VERTICAL LINES

None

TRACEWIDTH FAULTS FOR 45 DEGREE LINES

None

There are 27 faults on layer 1

ARTWORK

1. GENERAL INFORMATION

PURPOSE OF ARTWORK CAPABILITY

The Telesis artwork capability enables you to photoplot your board design using a Gerber photoplotter in the 6200 series.

You use the artwork capability, first, to produce artwork files (an artwork file is a text file containing all the information needed to photoplot a single filmsheet; such as, coordinates, line widths, pad shapes and sizes, and parameter values). You then transfer these artwork files to a magnetic tape that is used as direct input to the photoplotter. The system formats both the artwork files and the tape for use with the Gerber photoplotters in the 6200 series.

SUMMARY OF STEPS INVOLVED IN THE ARTWORK PROCESS

Below is a brief explanation of the steps you should follow in completing the artwork process on the Telesis system. Each step is detailed in the following sections.

NOTE

> Before you begin the artwork process, you should already have created pin description files for all components on the board and a LAYERSTD file (refer to the LIBRARY section, subsection PIN FILES and the PLACEMENT section, subsection PREPARATION).

1. INPUT A PHOTOPLOT-PAR TEXT FILE

This is the photoplot parameter file. It specifies the parameter settings required by the Gerber photoplotter. You can omit this file if you wish to use the system default parameter values.

2. INPUT A PHOTOPLOT-CON TEXT FILE

This is the photoplot control file. It specifies the drawing layers to be photoplotted on any given filmsheet. Typically, you will list several filmsheets in one PHOTOPLOT-CON text file.

3. INPUT AN APERTURE-TAB TEXT FILE

This is the aperture table. It lists all of the line sizes and pad shapes and sizes in your board design, and specifies the aperture station to be used for each size.

STEP 4 USE THE |CREATE PHOTOPLOT| COMMAND

This command does the following:

- Creates a PHOTOPLOT-LOG text file listing errors.
- Creates an artwork text file for each filmsheet you specified in the PHOTOPLOT-CON file.

STEP 5 USE THE |PENPLOT ARTWORK| COMMAND

This step is optional. Use this command to check your artwork files before photoplotting them.

NOTE: PROCEED TO STEPS 6 AND 7 ONLY IF YOU HAVE - - MAGNETIC TAPE WITH YOUR SYSTEM

STEP 6 USE THE |GERBER TO TAPE| COMMAND

This command prepares the magnetic tape to receive your artwork files.

STEP 7 USE THE |ARTFILE TO TAPE| COMMAND

Use this command to specify the artwork files you wish to transfer to tape, and to transfer them.

2. HOW TO INPUT THE PHOTOPLOT-PAR TEXT FILE

2.1 PURPOSE

The PHOTOPLOT-PAR text file specifies parameter values that the Gerber photoplotter will use. The PHOTOPLOT-PAR text file is not required if you wish to use the system parameter default values.

EXAMPLE

```
( TYPICAL PHOTOPLOT PARAMETER FILE )
( FILE NAME :  PHOTOPLOT-PAR )
DEVICE-TYPE GERBER6240
CODE ASCII
NUMBER-OF-WHEELS  1
END-BLOCK-CHAR  *
COORDINATES  INCREMENTAL
ENGLISH
FORMAT  5.3
LEADING-ZEROS  NOT-SUPPRESS
TRAILING-ZEROS  NOT-SUPPRESS
TEXT-THICKNESS  .012
UNDEF-LINE-WIDTH  .012
MACHINE-OFFSET  0.0.0.0
SPLIT-PLANE  5 1 2 3 4 5
END
```

2.2 NAME AND LOCATION OF THE FILE

You must name this file PHOTOPLOT-PAR (no spaces between any of the characters). The PHOTOPLOT-PAR file must be in the SYSTEM-LIBRARY or the current project when you use the CREATE PHOTOPLOT command.

2.3 REQUIRED CONTENTS AND FORMAT OF THE FILE

Contact the photoplotting service you will use to find out the proper values for the parameters. You may omit any parameter (both keyword and value) from the file if you wish to use the default value shown in the following table. Many of the parameters only have one allowable value. For convenience, you can omit these parameters from your file and use the default value. Each parameter must be placed at the beginning of a new line.

CONTENTS	FORMAT
DEVICE TYPE OF THE GERBER MACHINE BEING USED	FORMAT: DEVICE-TYPE GERBER62xx EXAMPLE: DEVICE-TYPE GERBER6240 Input a hyphen (-) and no blank spaces in the keyword: DEVICE-TYPE. Input at least one blank space after the keyword: DEVICE-TYPE. Input no blank spaces in the text: GERBER62xx. xx is the number of the particular device in the 6200 serios. (Actually, your art-work tape will be usable with any Gerber photoplotter in the 6200 series regardless of the xx you specify here.) DEFAULT: GERBER6200
CODE	FORMAT: CODE ASCII EXAMPLE: CODE ASCII Input at least one blank space after the keyword: CODE. ASCII is the only allowable value for this parameter. Input no blank spaces in the word: ASCII. DEFAULT: ASCII
NUMBER OF APERTURE WHEELS USED	FORMAT: NUMBER-OF-WHEELS 1 EXAMPLE: NUMBER OF WHEELS 1 Input a hyphen (-) and no blank spaces in the keyword: NUMBER-OF-WHEELS. Input at least one blank space after the keyword: NUMBER-OF-WHEELS.

(table continued on next page)

(table continued from previous page)

CONTENTS	FORMAT
NUMBER-OF-WHEELS	The only allowable value for this parameter is 1. DEFAULT: NUMBER-OF-WHEELS 1
CHARACTER TO BE USED AT THE END OF EACH BLOCK OF COORDINATES	FORMAT: END-BLOCK-CHAR * EXAMPLE: END-BLOCK-CHAR * Input a hyphen (-) and no blank spaces in the keyword: END-BLOCK-CHAR. Input at least one blank space after the keyword: END-BLOCK-CHAR. The only allowable value for this paramater is the asterisk symbol (*). DEFAULT: END-BLOCK-CHAR *
ABSOLUTE OR INCREMENTAL COORDINATES	FORMAT: COORDINATES ABSOLUTE or INCREMENTAL EXAMPLE: COORDINATES ABSOLUTE DEFAULT: COORDINATES ASBOLUTE
INPUT UNITS (for machine offset, text thickness, and undefined line width)	FORMAT: UNITS:ENGLISH or UNITS:METRIC EXAMPLE: UNITS:METRIC DEFAULT: UNITS:ENGLISH Note that when you choose English of metric units, you are choosing an input value for this file only. <u>THIS INPUT VALUE DOES NOT AFFECT ANY OTHER FILES IN YOUR PROJECT.</u> You may change the unit of measurement at any time. However, note that when you choose a new unit of measurement, ONLY THOSE VALUES FOR INPUT UNITS FOLLOWING THAT UNIT OF MEASUREMENT ARE IN THAT UNIT.

(table continued on following page)

(table continued from previous page)

CONTENTS	FORMAT
ENGLISH OR METRIC UNITS FOR OUTPUT	FORMAT: ENGLISH OR METRIC EXAMPLE: ENGLISH DEFAULT: ENGLISH
FORMAT OF COORDINATES	FORMAT: FORMAT m.n EXAMPLE: FORMAT 5.3 Input at least one blank space after the keyword: FORMAT. m indicates the number of spaces before the decimal point. n indicates the number of spaces after the decimal point. Input a period (.) between m and n with no blank spaces. 4.4 and 5.3 are the only allowable values with ENGLISH coordinates (above). 5.2 and 5.3 are the only allowable values with METRIC coordinates (above). We recommend 5.3 because it provides greater accuracy. DEFAULT: FORMAT 5.3
SUPPRESS OR NOT-SUPPRESS LEADING ZEROS IN COORDINATE VALUES	FORMAT: LEADING-ZEROS SUPPRESS or LEADING-ZEROS NOT-SUPPRESS EXAMPLE: LEADING-ZEROS, NOT-SUPPRESS Input a hyphen (-) and no blank spaces in the keyword: LEADING-ZEROS. Input at least one blank space after the keyword: LEADING-ZEROS. Input a hyphen and no blank spaces in the word: NOT-SUPPRESS. DEFAULT: LEADING-ZEROS NOT-SUPPRESS

(table continued on next page)

(table continued from previous page)

NOT-SUPPRESS TRAILING ZEROS IN COORDINATE VALUES	FORMAT: TRAILING-ZEROS NOT-SUPPRESS EXAMPLE: TRAILING-ZEROS NOT-SUPPRESS Input a hyphen (-) and no blank spaces in the keyword: TRAILING-ZEROS.
(TRAILING ZEROS parameter is ignored by CREATE PHOTOPLOT program.)	Input at least one blank space after the keyword: TRAILING-ZEROS. NOT-SUPPRESS is the only allowable value for this parameter. Input a hyphen (-) and no blank spaces in the word: NOT-SUPPRESS. DEFAULT: TRAILING-ZEROS NOT-SUPPRESS
LINE THICKNESS TO BE USED FOR PHOTO-PLOTTING TEXT CHARACTERS	FORMAT: TEXT-THICKNESS n EXAMPLE: TEXT-THICKNESS .012 Input a hyphen (-) and no blank spaces in the keyword: TEXT-THICKNESS. Input at least one blank space after the keyword: TEXT-THICKNESS. n is a value in inches or millimeters (depending on which input unit you chose) DEFAULT: Smallest line width shown in the APERTURE-TAB file.
LINE THICKNESS TO BE USED FOR PHOTO-PLOTTING LINES OF UNDEFINED (0) WIDTH IN THE BOARD DRAWING	FORMAT: UNDEF-LINE-WIDTH n EXAMPLE: UNDEF-LINE-WIDTH .012 Input two hyphens (-) and no blank spaces in the keyword: UNDEF-LINE-WIDTH. Input at least one blank space after the keyword: UNDEF-LINE-WIDTH. n is a value in inches or millimeters (depending on which input unit you chose.) DEFAULT: Smallest line width shown in the APERTURE-TAB file.

(table continued on next page)

(table continued from previous page)

CONTENTS	FORMAT
OFFSET OF THE 0,0 OF YOUR BOARD FROM THE 0,0 (lower left corner) OF THE PHOTOPLOT FILM	FORMAT: MACHINE-OFFSET x.y EXAMPLE: MACHINE-OFFSET 2.0, 2.0 Input a hyphen (-) and no blank spaces in the keyword: MACHINE-OFFSET. Input at least one blank space after the keyword: MACHINE-OFFSET. x is the x offset; y is the y offset. Input a comma (,) between x and y with no blank spaces. If you placed a decimal point in the number the system assumes either inches or milli-meters, depending on which input unit you chose. If you did not place a decimal pt. in the number, then the unit for the number is mils if you chose "English" as the input unit and board database units if you chose "metric" as the input unit. The x and y values may be no longer than 5 digits. The upper and lower limits for x and y values are $\pm$32000 mils, expressed either as 32000 or 32.0. DEFAULT: MACHINE-OFFSET 0,0
COMMENTS You may optionally include information-al comments in this file	FORMAT: (text of comment) EXAMPLE: (PHOTOPLOT-PAR FILE) Place comments in parentheses. The comment text may be anything you wish. The system ignores all information placed in parentheses in a text file.

(table continued on next page)

(Table continued from previous page)

CONTENTS	FORMAT
SPLIT-PLANE (optional)	SPLIT-PLANE layer number net numbers SPLIT-PLANE is the keyword (must be entered as the first word on a new line). Layer number specifies the imbedded plane layer that will be used as a split plane. Net numbers specify up to nine nets that will be included on the split plane layer. EXAMPLE : SPLIT-PLANE 5 1 2 3 4 5 Layer 5, which is an imbedded plane, will be used as a split plane layer. Nets numbered 1, 2, 3, 4, and 5 will be included on layer 5, as well as the net number of layer 5.
END	END The keyword END must be entered as the last line of the PHOTOPLOT-PAR text file.

2.4 RSX SPLIT PLANES SPECIFIED IN THE FILE

An imbedded plane is an internal board layer that is made up entirely of copper. A split plane is an imbedded plane that is divided into more than one area of copper.

An imbedded plane connects component pins in an extensive net such as VCC or GND. Since copper is conductive, component pins connected to the copper will also be connected to each other. When the board is photoplotted, the system determines the net number of a component pin from information in the Device File. If the component pin net number matches an imbedded plane layer net number, the photoplotter flashes a thermal relief pad on the imbedded plane layer to connect the component pin to the copper on the layer. If the component pin net number does not match an imbedded plane layer net number, the photoplotter flashes an anti pad on the imbedded plane layer to isolate the component pin from the copper on the layer.

The operator creates a split plane layer by adding a SPLIT-PLANE line to the PHOTOPLOT-PAR text file and adding lines to the corresponding layer of the board drawing. The SPLIT-PLANE entry in the PHOTOPLOT-PAR text file specifies the imbedded plane layer that will be used as a split plane layer and the net numbers that will be included on the layer. The lines on the corresponding board drawing layer represent the outlines of the different copper areas that will be photoplotted on the physical board layer.

The system does not check to ensure that net numbers of component pins within a copper area match; the operator must ensure this by correct component placement. The system only knows that all component pins with a net number listed in the Split Plane entry will be connected to the copper on the Split Plane layer. The operator must divide the board drawing into different copper areas and place components in the correct copper area to ensure that component pins with different net numbers do not end up in the same copper area.

To specify an imbedded plane layer as a split plane, add the following line to the PHOTOPLOT-PAR text file :

```
SPLIT-PLANE   layer number   net numbers contained on the layer
```

```
EXAMPLE :  SPLIT-PLANE  5 1 2 3 4 5
```

When the CREATE PHOTOPLOT command is selected, the system will flash a thermal relief pad on layer 5 for ALL component pins with a net number of 1, 2, 3, 4, and 5, or the net name of the net specified for layer 5 in the LAYERSTD file. Anti pads will be flashed on layer 5 for all other component pins.

WARNING

Ensure that all components with component pins in the same net are placed in the same copper area of the board. The system will not flag a warning or error message if component pins of different nets are located in the same copper area.

NOTES

Separate the keyword from the layer number and the layer number from the net number(s) with at least one space.

Separate different net numbers with a space.

The split plane entry in the PHOTOPLOT-PAR text file can be used a maximum of five times.

The LAYERSTD file entry for the layer is included on the split plane, allowing a maximum of ten net numbers to be specified on one layer.

The order in which the nets are listed is not important; but if more than nine nets are listed, the system will display a Warning message and only the first nine net numbers will be used.

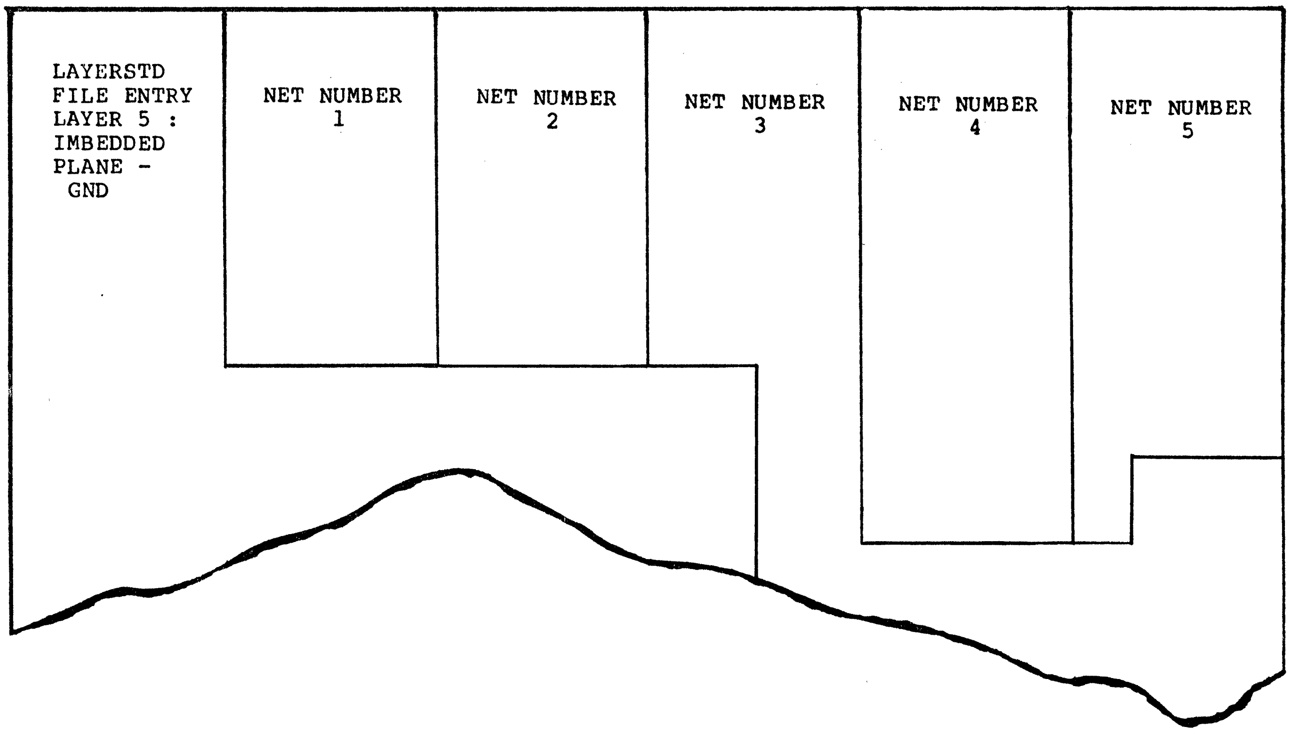
EXAMPLE

In the following example :

- The LAYERSTD entry for layer 5 specifies an imbedded GND plane
- The operator has divided layer 5 of the board drawing into 6 equal areas
- The operator has added the following line to the PHOTOPLOT-PAR text file :

  SPLIT-PLANE 5 1 2 3 4 5

The following figure illustrates the board drawing using the above information.



Net numbers are not assigned to specific copper areas by the system. It is up to the operator to place components with similar component pin net numbers in the same copper area. The copper areas illustrated in the figure above, are shown with net numbers to clarify the example only.

3. HOW TO INPUT THE PHOTOPLOT-CON TEXT FILE

PURPOSE

The PHOTOPLOT-CON (photoplot control) file is a list of filmsheets. It specifies the drawing layers to be photoplotted on each filmsheet. (When you use the |CREATE PHOTOPLOT| command, the system will create an artwork text file for each filmsheet you specify here.)

EXAMPLE

```
(TYPICAL PHOTOPLOT CONTROL FILE)
(FILE NAME:  PHOTOPLOT-CON)
(COMPONENT SIDE FILM)
ARTWORK FILM-1
COMPONENT-SIDE
PLATING-BAR
TOOLING-CORNERS
DRAWING-FORMAT
(SOLDER SIDE FILM)
ARTWORK FILM-2
SOLDER-SIDE
PLATING-BAR
TOOLING-CORNERS
DRAWING-FORMAT
(INTERNAL SIGNAL #1 FILM)
ARTWORK FILM=3
DBLAYER 3
TOOLING-CORNERS
DRAWING-FORMAT
(INTERNAL SIGNAL #2 FILM)
ARTWORK FILM-4
DBLAYER 4
TOOLING-CORNERS
DRAWING-FORMAT
(IMBEDDED GROUND PLANE FILM)
ARTWORK FILM-5
DBLAYER 5
TOOLING-CORNERS
DRAWING-FORMAT
(IMBEDDED VOLTAGE PLANE FILM)
ARTWORK FILM-6
DBLAYER 6
TOOLING-CORNERS
DRAWING-FORMAT
(COMPONENT SIDE SOLDER MASK FILM)
ARTWORK FILM-7
COMPONENT-SOLDER-MASK
TOOLING-CORNERS
DRAWING-FORMAT
(SOLDER SIDE SOLDER MASK FILM)
ARTWORK FILM-8
SOLDER-SOLDER-MASK
TOOLING-CORNERS
DRAWING-FORMAT
(SILKSCREEN FILM)
ARTWORK FILM-9
SILKSCREEN
COMPONENT-OUTLINE
REFERENCE-DESIGNATOR
TOOLING-CORNERS
DRAWING-FORMAT
(DRILL DRAWING FILM)
ARTWORK FILM-10
DRILL-CODE
CARD-OUTLINE
BOARD-DIMENSIONS
TOOLING-CORNERS
DRAWING-FORMAT
END
```

NAME AND LOCATION OF THE FILE

You must name the file PHOTOPLOT-CON. It must be in the current project file or in the SYSTEM-LIBRARY when you use the |CREATE PHOTOPLOT| command.

REQUIRED CONTENTS AND FORMAT OF THE FILE

The PHOTOPLOT-CON file must contain one section for each filmsheet you wish to describe. The table below describes the required contents and format to be used for each section.

You may input as many filmsheet sections as you wish. Input each section immediately below the previous one: there should be no blank lines in the file.

CONTENTS	FORMAT
FILMSHEET NAME	FORMAT: ARTWORK (filmsheet name) EXAMPLE: ARTWORK FILM-1 Input at least one blank space after the keyword: ARTWORK. The filmsheet name may be up to 14 keyboard characters with no blank spaces allowed.
LIST OF ALL DRAWING LAYERS TO BE PLOTTED ON THE FILMSHEET NAMED ABOVE	FORMAT: (layer name or layer number) EXAMPLE 1: COMPONENT-SIDE EXAMPLE 2: DBLAYER 6 Input each layer at the beginning of a new line. You may input either the layer name or the layer number. (However, you must input the layer number rather than the name if you have assigned the same layer name to more than one layer in the LAYERSTD file.) LAYER NAME: If you input a layer name, it should correspond exactly to a layer name in the LAYERSTD file. The name must have no

(table continued on next page)

(table continued from previous page)

CONTENTS	FORMAT
	blank spaces. LAYER NUMBER: If you input a layer number, it must have this format: DBLAYER n. The word DBLAYER must have no blank spaces, and it must be followed by at least one blank space. n may be any drawing layer (0-255).
COMMENTS You may optionally include informational comments in this file	FORMAT: (text of comment) EXAMPLE: (Component-Side Film) Place the comments in parentheses. The comment text may be anything you wish. The system ignores all information placed in parentheses in a text file.
END STATEMENT	FORMAT: END EXAMPLE: END Input the word "END" as the last line of the file.

4. HOW TO INPUT THE APERTURE-TAB TEXT FILE

PURPOSE

The APERTURE-TAB (aperture table) text file is a list of all line sizes and pad shapes occurring in a PC board design. The APERTURE-TAB text file assigns each pad shape and size to an aperture station on the photoplotter´s aperture wheel. The APERTURE-TAB is a necessary prerequisite for producing the artwork text file for the Gerber photoplotter. The example below illustrates a typical aperture table text file.

EXAMPLE: APERTURE-TAB text file

```
                 APERTURE-TAB  1
(APERTURE-TAB)
WHEEL 1
1 CIRCLE=.055               D10
2 CIRCLE=.062               D11
3 CIRCLE=.065               D12
4 CIRCLE=.075               D13
5 CIRCLE=.080               D14
6 CIRCLE=.095               D15
7 CIRCLE=.100               D16
8 CIRCLE=.125               D17
9 CIRCLE=.150               D18
10 LINE=.010                D19
11 LINE=.012                D70
12 LINE=.015                D71
13 LINE=.020                D20
14 LINE=.025                D21
15 LINE=.050                D22
16 LINE=.100                D23
17 SQUARE=.062              D24
18 SQUARE=.075              D25
19 SQUARE=.080              D26
20 SQUARE=.095              D27
21 FLASH=AB00               D28
(.050 THERMAL-RELIEF)
22 FLASH=AB01               D29
(.060 THERMAL-RELIEF)
23 FLASH=AB02               D72
(.080 THERMAL-RELIEF)
24 FLASH=AB05               D73
(MOIRE PATTERN)
END
```

NAMING THE APERTURE-TAB TEXT FILE

When creating the APERTURE-TAB text file, the operator MUST name the file APERTURE-TAB. The file may be created with the |EDIT APERTURE-TAB| text leadthru, Telesis Keyboard Text Editor (VT101 or WYSE), or the Telesis Function Screen Editor.

Aftr creating, editing and saving the file, and when the operator uses the |CREATE PHOTOPLOT| command, the system looks for the APERTURE-TAB text file in the current project, then in the SYSTEM-LIBRARY.

NOTE: If the operator picks |CREATE PHOTOPLOT| without the APERTURE-TAB text file the system will report,

-NO APERTURE-TAB FILE-

The APERTURE-TAB is a prerequisite to artwork generation. The system does not assign default line sizes and pad shapes to a PC board design, and will not proceed with |CREATE PHOTOPLOT| without the APERTURE-TAB file.

USING THE APERTURE-TAB TEXT LEADTHRU.

The text leadthru capability allows the operator to create the text file on the function screen, with the system maintaining the proper formatting requirements of the file. The operator simply uses the interactive commands on the |EDIT APERTURE-TAB| leadthru to input and edit the information contained in the file. The leadthru reduces the possibility of numerous formatting errors that could result when the text file is created on a text editor keyboard.

When using the |EDIT APERTURE-TAB| leadthru, the system assists the operator in the following ways:

1. Automatically names the text file APERTURE-TAB.
2. Prompts the operator for a revision label.
3. Automatically positions the cursor when the operator inputs information.
4. Automatically inputs the keywords required in the file.
5. Places parentheses ( ) around comment lines.
6. Issues error messages if the operator-defined input is improperly formatted.

USING THE TEXT LEADTHRU TO NAME AND CREATE THE APERTURE-TAB FILE

When the operator picks the |EDIT APERTURE-TAB| command, the system issues a prompt for a text file revision label (up to four alphanumeric characters). The revision label specified opens an existing APERTURE-TAB text file (of that revision) or opens a new file named APERTURE-TAB. If |ENTER| is picked without specifying a revision, the system simply opens the latest version of the file, or creates a new file named APERTURE-TAB.

The operator may use the steps described below to create the APERTURE-TAB text file using the leadthru capability.

STEP 1. |TEXT LEADTHRU|

Pick the |TEXT LEADTHRU| command to display the menu of leadthru options:

|EDIT APERTURE-TAB|
|EDIT PIN FILE|
|EDIT LAYERSTD|
|EDIT PENPLOT-CON|
|EDIT DEVICE FILE|

STEP 2. |EDIT APERTURE-TAB|

Pick the |EDIT APERTURE-TAB| command to open an existing APERTURE-TAB text file, or to create a new version of the file. The system first prompts for a revision label, then opens the file specified by the revision. If an existing file is not found, the system creates a new file of the revision specified. The example below illustrates the input sequence when creating a new file:

|EDIT APERTURE-TAB| (REV) 1 |ENTER|

When |ENTER| is picked, the system displays the following information on the graphics screen:

```
                    APERTURE-TAB  1

     _  (APERTURE-TAB)
        END
```

(The cursor is positioned at the start of the comment line (APERTURE-TAB).)

The operator may now use the following commands to add information between (APERTURE-TAB) and the END statement:

|ADD WHEEL|
|ADD APERTURE|
|ADD COMMENT|

STEP 3. |ADD WHEEL|

Pick the |ADD WHEEL| command to specify the photoplotter aperture wheel number. |ADD WHEEL| opens the wheel section of the file, indicated by an asterisk (*) in the left margin. Currently, WHEEL 1 is the only allowable value.

EXAMPLE: |ADD WHEEL| (WHEEL) 1 |ENTER|

When ENTER is picked, the system displays the WHEEL line of the file, preceded by an asterisk. The asterisk indicates that the WHEEL section of the file is open for editing.

```
                    APERTURE-TAB  1

        (APERTURE-TAB)
     *  _WHEEL 1
        END
```

The operator may now use the |ADD APERTURE| and |ADD COMMENT| commands to add information to the WHEEL section.

If using the Telesis Keyboard or Function Screen Editor, input the keyword WHEEL, a blank space, then the number 1.

STEP 4. (Optional) Choose English or Metric Units

The system gives you a choice of English or metric units for Photoplot-PAR values.

|UNITS:ENGLISH| |UNITS:METRIC|

If you do not make a choice, the default is English units. Note that when you choose English or metric units, you are choosing an input value for this file only. **THIS INPUT VALUE DOES NOT AFFECT ANY OTHER FILES IN YOUR PROJECT.**

When you input a number for size, if a decimal point is part of the number, the system assumes either inches or millimeters (depending on the input unit you chose). If this number has no decimal point, the system assumes mils for English units input and board database units (.01mm) for metric input.

You may change the unit of measurement at any time. However, note that wehn you choose a new unit of measurement, **ONLY THOSE LINE SIZES AND PAD SHAPES FOLLOWING THAT UNIT OF MEASUREMENT ARE IN THAT UNIT.** Line sizes and pad shapes following the previously chosen unit are in that previously chosen unit of measurement.

For example, in the file below, the circle is in metric units, while the square is in English units.

```
            APERTURE-TAB 1

        (APERTURE-TAB)
    *   WHEEL 1
    *   UNITS:METRIC
    *   1 CIRCLE=.55
    *   UNITS:ENGLISH
    *   1 SQUARE=.055
```

STEP 5. |ADD COMMENT|

The operator may use the optional |ADD COMMENT| command to supply informational text into the file. A comment is always enclosed within parentheses ( ), and is ignored by the system when CREATE PHOTOPLOT processes the APERTURE-TAB file.

EXAMPLE: |ADD COMMENT| (TEXT) STA APERTURE CODE |ENTER|

RESULT:

```
                    APERTURE-TAB  1

      (APERTURE-TAB)
  *   WHEEL 1
  * _ (STA    APERTURE    CODE)
      END
```

In the above example, the comment line provides the header for the 24 station lines in the file. Comments may be conveniently placed above the related section or line entry, providing the additional information needed for easy interpretation of the file's contents.

NOTE: A maximum of 59 alphanumeric and/or blank spaces may be included in a comment entry. If the keyboard input exceeds 59 characters, the system interrupts, then inserts the first 59 characters into the file. The system automatically inserts parentheses around operator-defined text.

If using the Telesis Keyboard or Function Screen Editor, input parentheses around line entries to be treated as comments.

STEP 6. ADD APERTURE

Pick the |ADD APERTURE| command to input the aperture wheel station numbers, aperture shapes, and aperture sizes occurring in the board design (i.e. each line width occurring in the board drawing, and each pad shape/size occurring in the pin files for components placed on the board).

NOTE: There are 24 allowable stations on a photoplotter aperture wheel. Contact the photoplotting service for aperture station assignments for the various aperture shapes/sizes.

EXAMPLE: |ADD APERTURE| (STATION NUMBER) 1 |ENTER|

When ENTER is picked after specifying the aperture wheel station number, the system displays a menu of apertures:

|LINE|	|CIRCLE|
|SQUARE|	|RECTANGLE|
|OBLONG|	|FLASH|

Select the desired aperture for the station number entered. The following prompts (in parentheses) are issued for each aperture listed below:

LINE (WIDTH)	CIRCLE (WIDTH)
SQUARE (WIDTH)	RECTANGLE (WIDTH, HEIGHT)
OBLONG (WIDTH, HEIGHT)	FLASH (FLASH NAME)

EXAMPLE: |CIRCLE| (WIDTH) 55 |ENTER|

Input the aperture size, in mils or millimeters (depending upon which input unit you chose) for the selected shape. When |ENTER| is picked, the system displays the line in the file with the appropriate aperture station D-code in the right margin. For example, STATION 1 on the aperture wheel is always assigned the code D10 on the Gerber 6240 Photoplotter.

RESULT:

```
                    APERTURE-TAB  1

      (APERTURE-TAB)
  *   WHEEL 1
  *   (STA    APERTURE      CODE)
  * _ 1 CIRCLE=.055          D10
      END
```

NOTE: When specifying a |FLASH|, ten alphanumeric characters may be used immediately after the equal (=) sign. The operator must be sure that the photoplotting service is able to interpret the FLASH name specified in the file.

When using the Telesis Keyboard or Function Screen Editor to create the file, use the following format specification when adding aperture information to the file:

FORMAT: (STATION NUMBER) (SHAPE)=(SIZE) (APERTURE STATION CODE)

EXAMPLE 1: 1 LINE=.010 D10

EXAMPLE 2: 2 CIRCLE=.062 D11

FOR LINE SIZE:

Input the station number followed by a blank space, then the aperture shape immediately followed by an equal sign (=), then the line width in mils. There must be no blank spaces before or after the equals sign.

FOR PAD SHAPE/SIZE:

Use exactly the same format used in the pin files for pad shapes and sizes. For shapes with two size dimensions (OBLONGS and RECTANGLES), the operator must make two entries in the text file, one showing width x height first, then one showing the height x width. For example:

1 OBLONG=.150x.100 D10
2 OBLONG=.100X.150 D11

This is required because the photoplotter will use two different aperture sizes to plot this shape/size depending on the vertical or horizontal rotation of the symbol containing this pad.

APERTURE STATION CODE

Input the aperture station D-code following the line or pad size. Input at lease one blank space before the station code. There must be no blank spaces in the code itself.

There are 24 allowable station codes (one for each of the 24 stations on an aperture wheel).

(STA	D-CODE)	(STA	D-CODE)	(STA	D-CODE)
1	D10	13	D20	11	D70
2	D11	14	D21	12	D71
3	D12	15	D22	23	D72
4	D13	16	D23	24	D73
5	D14	17	D24		
6	D15	18	D25		
7	D16	19	D26		
8	D17	20	D27		
9	D18	21	D28		
10	D19	22	D29		

STEP 7. END STATEMENT

When using the EDIT APERTURE-TAB leadthru, the system automatically includes the END statement at the last line position. The cursor cannot be moved beyond the END statement.

If using the Telesis Keyboard or Function Screen Editor, include the END statement as the last line of the file.

STEP 8. |DONE|

Pick |DONE| when finished creating the APERTURE-TAB text file. To close the file, select one of the following commands:

|SAVE FILE| - allows the operator to save the active version of the file, and to delete the previous version of the same revision. If the file is newly created, the system saves the file under the revision label specified when the file was opened, or with revision "1" if a (REV) was was not specified.

Example:

New file: APERTURE-TAB

|SAVE FILE| : APERTURE-TAB 1

|SAVE FILE NEW RV| - allows the operator to save the active file under a revision other than the one originally specified when the file was opened. The system does not delete the old version.

If the active file is newly created without a revision originally specified, the system saves the file with the revision label input with the |SAVE FILE NEW RV| command.

|CANCEL ACTV FILE| - allows the operator to cancel the actively displayed file on the graphics screen without deleting the original (if it exists). The original file maintains its original revision label.

|EDIT FILE| - allows the operator to return to the active file for additional editing.

EDITING THE APERTURE-TAB TEXT FILE WITH THE TEXT LEADTHRU

When creating the file, the operator may make additions or corrections. The file may also be re-opened at a later time to update any version of the file previously created. When re-opening the file, the operator must specify the revision label if the file is not the latest revision. When the system issues the (REV) prompt on the function screen, simply pick |ENTER| to open the latest revision.

The following commands may be used to edit the APERTURE-TAB text file with the leadthru capability:

|CHANGE WHEEL|	|DELETE SECTION|
|CHANGE COMMENT|	|UNDELETE SECTION|
|CHANGE APERTURE|	|DELETE LINE|
	|UNDELETE LINE|

POSITIONING THE CURSOR

To edit the file, the operator must always position the cursor to the line to be edited. The cursor always remains in the left margin of the file and must be properly positioned .

For ADD commands during editing:

Position the cursor to the line above the desired location of the addition. Lines positioned below the cursor will shift down when the new line is added. If the cursor is positioned at the END statement, the new line will be placed one line above the END statement.

For CHANGE or DELETE commands:

Move the cursor to the line to be changed or deleted.

Use the following commands to move the cursor:

|SEC UP| - moves the cursor up one section.

|SEC DWN|- moves the cursor down one section.

|LINE UP|- move the cursor up one line.

|LINE DWN|- moves the cursor down one line.

THE DIFFERENCE BETWEEN A SECTION AND A LINE

The system groups the information in the APERTURE-TAB file into sections of interrelated data. For example, the WHEEL section contains all information pertaining to the 24 stations on the aperture wheel. When editing lines within the WHEEL section, use the |LINE UP| and |LINE DWN| commands.

While editing the WHEEL section, asterisks(*) are displayed at the left margin of the file. The asterisks highlight all lines within the WHEEL section of the file. When |DONE| is picked, the system repositions the cursor to the WHEEL line of the file, then cancels the asterisk display.

When the cursor is positioned at the WHEEL line, use the |SEC DWN| command to move the cursor to the END statement or the next WHEEL section of the file. Currently, WHEEL 1 is the only allowable wheel in the aperture table. If the cursor is positioned at the END statement, SEC UP will move the cursor to the WHEEL 1 line of the file.

EXAMPLE: |LINE DWN| (PICKED 3 TIMES)

```
                 *    21 FLASH=AB00        D28
                 *    (.050 THERMAL RELIEF)
Current          *    22 FLASH=AB01        D29
 cursor-->       *  _ (.060 THERMAL RELIEF)
                 *    23 FLASH=AB02        D72
New              *    (.080 THERMAL RELIEF)
cursor-->        *  _ 24 FLASH-AB05        D73
                 *    (MOIRE PATTERN)
                      END
```

EXAMPLE: |SEC DWN|

```
                          APERTURE-TAB

Current              (APERTURE-TAB)
 Cursor-->         _ WHEEL 1
                     (STA      APERTURE       CODE)
New Cursor           1 CIRCLE=.055             D10
 Position -->      _ END
```

IF THE FILE IS LONGER THAN SHOWN ON THE GRAPHICS SCREEN

If lines are added to the file after the graphics screen is full, the file scrolls up so that the bottom lines are visible and the top lines are not. Use the LINE and SEC softkeys to scroll the file to the lines to be edited.

CHANGING INFORMATION IN THE FILE

|CHANGE WHEEL|

The |CHANGE WHEEL| command allows the operator to open the WHEEL section of the APERTURE-TAB file for editing. Before using the |CHANGE WHEEL| command, the operator must position the cursor to the WHEEL line. When the system prompts the operator for the WHEEL NUMBER, simply pick |ENTER| to open the WHEEL section, with the asterisk display highlighting in the left margin.

Currently, WHEEL 1 is the only allowable value.

|CHANGE APERTURE|

The |CHANGE APERTURE| command allows editing of existing aperture station line entries. The input sequence for |CHANGE APERTURE| is the same as the input sequence for the |ADD APERTURE| command. However, the operator may skip any prompt, except shape, for any information that remains unchanged on the line being edited.

EXAMPLE: Existing line entry:

1 SQUARE=.062 D10

|CHANGE APERTURE| (STATION NUMBER) |ENTER| |SQUARE| 75 |ENTER|

Resulting line entry:

1 SQUARE=.075 D10

|CHANGE COMMENT|

EXAMPLE: |CHANGE COMMENT|

```
                              APERTURE-TAB 1
             (APERTURE-TAB)
Cursor->   _ (APERTURE TABLE FOR GERBER PHOTOPLOTTER)
             WHEEL 1
             (STA     APERTURE        CODE)
             1 CIRCLE=.055            D10
             2 CIRCLE=.062            D11
```

INPUT: CLOCK GENERATOR BOARD |ENTER|

Result: The comment line is now changed to (CLOCK GENERATOR BOARD).

```
                              APERTURE-TAB 1
             (APERTURE-TAB)
Cursor->   _ (CLOCK GENERATOR BOARD)
             WHEEL 1
             (STA     APERTURE        CODE)
             1 CIRCLE=.055            D10
             2 CIRCLE=.062            D11
```

DELETING INFORMATION FROM THE FILE

Use the following commands to DELETE and reinsert information on the file:

|DELETE LINE| - allows deletion of a single line entry specified by the cursor position.

|UNDELETE LINE| - allows reinsertion of the line deleted with |DELETE LINE| command. |UNDELETE LINE| may be used to reposition the line to another area of the file. The system restores the line below the new cursor position regardless of the line's original placement.

|DELETE SECTION| - allows deletion of a single section specified by the cursor position. The cursor must be positioned at the first line of the section to be deleted.

|UNDELETE SECTION| - allows restoration of the last file section deleted. The deleted section may be restored to another location in the file.

NOTE: The END statement cannot be deleted.

USING THE |DELETE LINE| AND |DELETE SECTION| COMMANDS

EXAMPLE: Deleting a comment line from the WHEEL 1 section:

```
                         APERTURE-TAB 1
             (APERTURE-TAB)
             (CLOCK GENERATOR BOARD)
           *  WHEEL 1
Cursor->   * _(STA    APERTURE       CODE)
           *  1 CIRCLE=.055          D10
           *  2 CIRCLE=.062          D11
```

Pick: DELETE LINE

When |DELETE LINE| is picked, the comment line (STA APERTURE CODE) is deleted from the file. The contents below the deleted line shift up, closing the open line.

RESULT:

```
                         APERTURE-TAB 1
             (APERTURE-TAB)
             (CLOCK GENERATOR BOARD)
           *  WHEEL 1
Cursor->   * _1 CIRCLE=.055          D10
           *  2 CIRCLE=.062          D11
```

The operator may now use the |UNDELETE LINE| command to reinsert the line (STA APERTURE CODE) back into the file. Use the |LINE UP| command to position the cursor back the WHEEL 1 line, then pick |UNDELETE LINE| to reinsert it to its original position in the file.

EXAMPLE: |DELETE SECTION|

(Deleting the WHEEL 1 section of the file.)

```
                    APERTURE-TAB 1
            (APERTURE-TAB)
            (CLOCK GENERATOR BOARD)
Cursor ->   _ WHEEL 1
            (STA    APERTURE      CODE)
            1 CIRCLE=.055         D10
            2 CIRCLE=.062         D11
            END
```

Pick: |DELETE SECTION|

RESULT:

```
                    APERTURE-TAB 1
            (APERTURE-TAB)
            (CLOCK GENERATOR BOARD)
Cursor->    _ END
```

In the example above, the |DELETE SECTION| command deleted all lines in the WHEEL 1 section. Pick the |UNDELETE SECTION| command to reinsert the entire WHEEL 1 section back to the file.

NOTE: When the cursor is positioned at the END statement, any line or section of text reinserted will be placed one line above END.

APERTURE TABLES NOT CREATED WITH THE EDIT APERTURE-TAB LEADTHRU

Aperture table text files that were previously created with the Telesis Keyboard or Function Screen Editor may be edited using the |EDIT APERTURE-TAB| text leadthru.

When the operator opens the file with the |EDIT APERTURE-TAB| command, the system checks it for formatting errors. If errors exist, the system issues a message on the function screen. This message states the number of existing faults. The file is then displayed on the graphics screen will all faulty lines bracketed between parentheses ( ) and asterisks *.

EXAMPLE: (* 1 CIRCLE= .055 D10*)

In the above example, the line is incorrect due to a blank space between the equal sign (=) and the circle diameter (.055). Lines in error are displayed as comments and must be deleted prior to using the ADD commands to specify the correct information.

APERTURE LIMITATIONS

1. LIMITATION OF APERTURE SIZES AVAILABLE AT THE PHOTOPLOTTING SERVICE.

Before completing the board drawing, contact the photoplotting service to be sure that all the apertures sizes are available. For example, if 1 mil lines are used in the board drawing, the photoplotting service may not have an aperture small enough to plot it.

2. LIMITATION OF 24 STATIONS ON THE APERTURE WHEEL

Only 24 different aperture sizes can be placed on the aperture wheel at one time, and only one wheel can be used to plot a filmsheet. Therefore, each filmsheet cannot require more than 24 different apertures. If possible, design the board so that the number of line sizes and pad shape/sizes does not exceed 24.

5. HOW TO USE THE |CREATE PHOTOPLOT| COMMAND

PURPOSE

The purpose of the |CREATE PHOTOPLOT| command is to create an artwork text file for each filmsheet you specified in the PHOTOPLOT-CON file. Each artwork file contains the information needed by the photoplotter to plot a filmsheet.

PREREQUISITES

- o The board drawing must be active.
- o Pin description files for every component on the board must be in the current project file or the SYSTEM-LIBRARY.
- o The LAYERSTD text file must be in the current project file or the SYSTEM-LIBRARY.
- o The PHOTOPLOT-PAR file (if you use one) must be in the current project file or the SYSTEM-LIBRARY. The PHOTOPLOT-PAR file is optional. (See p. BOARD-37.)
- o The PHOTOPLOT-CON file must be in the current project file or the SYSTEM-LIBRARY.
- o The APERTURE-TAB file must be in the current project file or the SYSTEM-LIBRARY.

  NOTE: Currently, any circle or arc at-width will be plotted as a single line using undefined line aperture from the PHOTOPLOT-PAR file.

HOW THE |CREATE PHOTOPLOT| COMMAND WORKS

When you pick |CREATE PHOTOPLOT|, the system creates two types of text files (the process takes from 1 to several hours depending on the size of the board and the number of filmsheets involved):

1. ARTWORK FILES

   The system gathers information from the board drawing, pin files, LAYERSTD, PHOTOPLOT-PAR, PHOTOPLOT-CON, and APERTURE-TAB files to create an artwork file for each filmsheet specified in the PHOTOPLOT-CON file. The system will not create an artwork file if it finds major erros in the information for that filmsheet. (See p. BOARD 54 for a list of possible error messages.)

   The system names these artwork files by combining the filmsheet name you specified in the PHOTOPLOT-CON file with the suffix -ART. For example, if you input the filmsheet name "FILM-1" in the PHOTOPLOT-CON file, the system will create an artwork text file named "FILM-1-ART".

2. THE PHOTOPLOT-LOG

The system also creates a text file named PHOTOPLOT-LOG. The PHOTOPLOT-LOG informs you of:

- Errors
- Whether or not an artwork file was created for each filmsheet
- Aperture station codes required for plotting each filmsheet.

EXAMPLE OF THE PHOTOPLOT-LOG:

```
    M E S S A G E S   from   P H O T O P L O T T E R  04-APR-83

  - Process PHOTOPLOT-PAR file
    --------------------------



INCREMENTAL
  */ Incorrect keyword in the line

  - Process LAYERSTD file
    ---------------------



  - Process PIN files
    -----------------

PROCESSING PIN FILE FOR SYMBOL VIA  REV :

PROCESSING PIN FILE FOR SYMBOL RES400  REV 1

PROCESSING PIN FILE FOR SYMBOL CCON24/154  REV :

PROCESSING PIN FILE FOR SYMBOL DIP.4  REV :

PROCESSING PIN FILE FOR SYMBOL DIP14  REV :
```

(example continued on next page)

(example continued from previous page)

```
        Photoplot file     FI-ART





WARNING -- PADS for the following pins are not defined in their PIN F
      1  Symbol ECON24/156         Pin     13 on COMPONENT-SIDE
      2  Symbol ECON24/156         Pin     14 on COMPONENT-SIDE
      3  Symbol ECON24/156         Pin     15 on COMPONENT-SIDE
      4  Symbol ECON24/156         Pin     16 on COMPONENT-SIDE
      5. Symbol ECON24/156         Pin     17 on COMPONENT-SIDE
      6  Symbol ECON24/156         Pin     18 on COMPONENT-SIDE
      7  Symbol ECON24/156         Pin     19 on COMPONENT-SIDE
      8  Symbol ECON24/156         Pin     20 on COMPONENT-SIDE
      9  Symbol ECON24/156         Pin     21 on COMPONENT-SIDE
     10  Symbol ECON24/156         Pin     22 on COMPONENT-SIDE
     11. Symbol ECON24/156         Pin     23 on COMPONENT-SIDE
     12. Symbol ECON24/156         Pin     24 on COMPONENT-SIDE


  LIST OF USED APERTURES

  1     D:2
  2     D:4
  3     D:6
  4.    D:8
  5     D:9
  6     D23
  7     D26
  8     D71

MESSAGE  Your photoplotfile is completed
```

TYPES OF ERROR MESSAGES YOU MAY RECEIVE IN THE PHOTOPLOT-LOG

- FILE DOES NOT EXIST: LAYERSTD

  No artwork files will be created because there is no LAYERSTD file in the current project or in the SYSTEM-LIBRARY.

- FILE DOES NOT EXIST: APERTURE-TAB

  No artwork files will be created because there is no APERTURE-TAB in the current project or in the SYSTEM-LIBRARY.

- MISSING PIN FILE FOR SYMBOL: (symbol name)

  No artwork files will be created because there is a component symbol in your board drawing for which there is no pin file in the current project or in the SYSTEM-LIBRARY.

- MISSING ")" IN APERTURE-TAB

  No artwork files will be created because there is a comment in the APERTURE-TAB for which there is no close-parentheses ")" sign.

- INCORRECT DEVICE TYPE. NOW GERBER6200 ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: GERBER6200.

- INCORRECT CODE. NOW ASCII ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: ASCII.

- INCORRECT COORDINATES. ABSOLUTE OR INCREMENTAL ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: INCREMENTAL.

- INCORRECT REQUEST FOR ZEROS. SUPPRESS/NOT-SUPPRESS ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: NOT-SUPPRESS.

- INCORRECT FORMAT. 4.4, 5.3 or 5.2 ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: 5.3.

- CONTRADICTION: WITH ENGLISH 4.4 or 5.3, WITH METRIC 5.2 or 5.3

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: 5.3

- INCORRECT END-OF-BLOCK REQUEST. *ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: *.

- INCORRECT NUMBER-OF-WHEELS. NOW 1 ONLY

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: 1.

- INCORRECT REQUEST MACHINE-OFFSET

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: 0,0.

- UNDEF-LINE-WIDTH NOT DEFINED

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: smallest line aperture in APERTURE-TAB.

- TEXT-THICKNESS NOT DEFINED

  Artwork files will be created. The system will overlook the error in the PHOTOPLOT-PAR file and supply the default value: smallest line aperture in APERTURE-TAB.

- MISSING KEYWORK "ARTWORK"

  The keyword "ARTWORK" is missing in the PHOTOPLOT-CON file. Some artwork files may be created, but the information in them may be incorrect. Correct the PHOTOPLOT-CON file, and repeat the |CREATE PHOTOPLOT| command.

- MISSING ARTWORK ATTRIBUTE-NAME OF PHOTOPLOT FILE

  The artwork file will be created. The system will overlook the error in the PHOTOPLOT-CON file and supply a default name for the filmsheet: "ARTWORK1". The artwork file will, therefore, be named "ARTWORK1-ART". If more than one filmsheet name is missing in the PHOTOPLOT-CON file, the system will supply the default names: "ARTWORK1", "ARTWORK2", etc. (up to "ARTWORK99").

- INCORRECT DBLAYER NUMBER

  There is an incorrect DBLAYER number in the PHOTOPLOT-CON file. The artwork file for the filmsheet containing that layer will not be created.

- DUPLICATE LAYER NAME

  The same layer is listed twice within a single filmsheet specification in the PHOTOPLOT-CON file. The artwork file for this filmsheet will be created, and the system will ignore the second listing of the layer.

- TOO MANY INTERNAL VOLTAGE LAYERS. NOT MORE THAN 4

  More than 4 internal-voltage layers are listed within a single filmsheet specification in the PHOTOPLOT-CON file. The artwork file for this filmsheet will not be created.

- INCORRECT SHAPE IN THE FILE APERTURE-TAB

There is an incorrect shape entry in the APERTURE-TAB file. No artwork file for a filmsheet requiring this shape will be created.

- MISSING APERTURE REQUEST: SHAPE, COORDINATES (x,y)

The system will not create an artwork file for a filmsheet containing a shape for which there is no entry in the APERTURE-TAB.

- WARNING PADS FOR THE FOLLOWING PINS ARE NOT DEFINED IN THEIR PIN FILES:
  (SYMBOL NAME) (PIN #) (LAYER NAME)

Artwork files will be created, but all undefined pads will be omitted from the files.

- WARNING: PADS FOR THE FOLLOWING PINS ARE DUPLICATED:
  (SYMBOL NAME) (PIN #) (LAYER NAME)

The system has found two pads in the same place on a single filmsheet. The artwork file for this filmsheet will be created, but the pad will only be included once in the artwork file to prevent blooming when the image is photoplotted.

- WARNING: DUPLICATE PINS:
  (x,y coordinates)

The system has found two pins in the same place on a single filmsheet. The artwork file for this filmsheet will be created, but only one pad will be included at this location in the artwork file to prevent blooming when the image is photoplotted.

- NO INFORMATION ON GIVEN LAYERS

The PHOTOPLOT-CON file contains a filmsheet specification for which all of the listed layers are empty of information. No artwork file will be created for this filmsheet.

- MISSING KEYWORD WHEEL IN FILE APERTURE-TAB

Artwork files will be created. The system will overlook the error and supply the default value: "WHEEL 1".

CORRECTING ERRORS AND REPEATING THE |CREATE PHOTOPLOT| COMMAND

Follow the steps below to correct errors shown in the PHOTOPLOT-LOG.

STEP 1 Print the PHOTOPLOT-LOG.

STEP 2 Read the PHOTOPLOT-LOG carefully to find out which artwork files (if any) were not created; and to find out which artwork files were created but show errors or warnings that you wish to correct. Make a list of all corrections that you need or wish to make.

STEP 3 Reopen the board drawing, the PHOTOPLOT-PAR, PHOTOPLOT-CON, APERTURE-TAB, LAYERSTD and/or pin files to correct the errors on your list.

STEP 4 Repeat the |CREATE PHOTOPLOT| command, making sure that you have met all prerequisites. (When you repeat the |CREATE PHOTOPLOT| command, new artwork files and a new PHOTOPLOT-LOG will be created. The old PHOTOPLOT-LOG and any of the old artwork files having the same name as the new ones will be deleted.)

STEP 5 Repeat STEPS 1-4 until the PHOTOPLOT-LOG shows no errors or warnings that you find unacceptable.

NOTE: To avoid the dealy involved while the system recreates artwork files that were acceptable the first time, you may reopen the PHOTOPLOT-CON file and delete the filmsheet sections corresponding to the acceptable artwork files. Then repeat the |CREATE PHOTOPLOT| command. (Alternatively, you may put parentheses around filmsheet sections instead of deleting them.)

6. CHECKING ARTWORK BY USING THE |PENPLOT ARTWORK| COMMAND

You may use the |PENPLOT ARTWORK| command to check the accuracy of your artwork files. With some exceptions (noted below), the appearance of a penplot created from an artwork file will correspond to the appearance of a photoplot created from the same file.

PREREQUISITES FOR PENPLOT ARTWORK

- o The artwork file to be pen plotted must be in the current project file or in the SYSTEM-LIBRARY.
- o The APERTURE-TAB text file and the PHOTOPLOT-PAR text file must be in the current project file or in the SYSTEM LIBRARY.

DEFINING THE ORIGIN OF THE ARTWORK FILE

You can set the origin (0,0 coordinate) when penplotting the art file with the |PENPLOT ARTWORK| command. If you do not wish to set the origin manually, you can allow the system to use the plotter default. The plotter default is set at 15mm, which is about one-half inch inside the hard-clip limits.

The procedure for setting the origin manually is as follows:

1. Use the joystick to move the penholder to the position where you want your 0,0 coordinate to be. If you set 0,0 as close to the edge of the paper as possible, this is setting 0,0 at the plotter's hard-clip limits. For more information on this, see the section on hard-clip limits in the Hewlett-Packard Manual.

2. Press "ENTER" and "P1" on the front panel of the plotter.

3. If you wish to mirror a plot of the art file or wish to change from a mirrored to a non-mirrored plot, set P1 and P2 in different corners of the paper. (For more details on this, see the Hewlett-Packard Manual). P2 can be any distance from P1 in the direction you want. Note that mirroring the plot does not change the art file; it only mirrors the art file on the Hewlett Packard plotter.

Every subsequent plot will have the same P1 or 0,0 if you do not change it and if you use the same size paper. When you are loading a different size paper between plots of art files, use one of the following three procedures to reset P1:

1. Hit "ENTER" and "RESET" on the front panel. This resets to a default setting of P1.

2. Manually reset P1.

3. Shut the plotter off and turn it on again. This will give you the default setting of P1.

Note that if you do not reset P1 and if you change to a smaller size paper, this might put 0,0 outside the hard-clip limits, in which case your plot will be clipped. On the other hand, if you do not reset P1 and if you change to a larger paper, this might cause 0,0 to lie too close to the center of the paper.

The Command Description Section of the manual tells you how to use the |PENPLOT ARTWORK| command. These are some general points to bear in mind when you use it:

- You may only pen plot one filmsheet at a time. Repeat the |PENPLOT ARTWORK| command to plot additional filmsheets.
- |PENPLOT ARTWORK| can only be used to plot artwork files created with 8203 (or later) software releases. (You may, however, make new artwork files for a board that was created with pre-8203 software. To do this, input a new PHOTOPLOT-CON file and a new PHOTOPLOT-PAR file, and use the |CREATE PHOTOPLOT| command as usual.)

PEN COLOR AND PEN POSITION

- Only one pen color can be used with the |PENPLOT ARTWORK| command. You do not create a PENPLOT-CON text file with the |PENPLOT ARTWORK| command (as you do with the |PENPLOT| command) because the plotter uses only pen number 1.
- The pen plotter will use the machine offset you specified in the PHOTOPLOT-PAR file to offset the 0,0 point of your drawing from the physical position of the pen.
- To reset the pen position, refer to the HEWLETT-PACKARD Manual.

DIFFERENCES IN PEN PLOT AND PHOTOPLOT APPEARANCE

- Unless you choose a scale of 1 when you pick |PENPLOT ARTWORK|, the pen plot will not be the same as the photoplot.
- Lines that are ≤ 12 mils width in the original drawing will be shown in the pen plot as a single line, that is, with no width.
- Lines that are >12 mils width will be drawn to scale in the pen plot, but they will be shown as unfilled double lines.
- Pad shapes that are to be flashed in the photoplot are shown as butterfly shapes (X) in the pen plot.

7. COPYING ARTWORK FILES TO AND FROM A GERBER-FORMATTED TAPE

7.1 HOW TO TRANSFER ARTWORK FILES TO A GERBER-FORMATTED MAGNETIC TAPE

If you have a magnetic tape capability with your Telesis system, you may use the |GERBER TO TAPE| and the |ARTFILE TO TAPE| commands to transfer your artwork files to a magnetic tape that can be used as direct input to a Gerber photoplotter in the 6200 series.

PREREQUISITES

- Artwork files to be transferred must be in the current project file or the SYSTEM-LIBRARY.
- The tape drive must be loaded and switched on.

  WARNING: Check with the photoplotting service you intend to use to find out if their photoplotter can use an 800 BPI tape, a 1600 BPI tape, or both. Set the HI DEN control on your tape drive accordingly. (See p. FILE/ARCH-29.)

STEPS TO FOLLOW

The Command Description Section of the manual tells you how to use the |GERBER TO TAPE| and |ARTFILE TO TAPE| commands. The steps to follow are described below.

STEP 1 Pick |GERBER TO TAPE|.

STEP 2 Pick |ARTFILE TO TAPE|, then input the name of the first artwork file you want to copy.

Pick |ARTFILE TO TAPE| again, and input the name of the second artwork file you want on this tape.

Repeat this procedure until you have input the names of all the artwork file names you have input.

STEP 3 Use the |SHOW COPY LIST| command if you want to display the list of artwork file names you have input.

STEP 4 Pick a final |ENTER| command after the |ENTER| that follows the last artwork file name you wish to list. The system will now copy and verify the files.

NOTE: When you send the tape to a vendor for photoplotting, send a copy of the APERTURE-TAB, the PHOTOPLOT-PAR, and the "LIST OF USED APERTURES" from the PHOTOPLOT-LOG. The vendor will need this information as well as the tape itself.

7.2 HOW TO TRANSFER ARTWORK FILES FROM TAPE BACK TO THE SYSTEM

PURPOSE

The Telesis system provides you with the ability to copy artwork files from a Gerber-formatted tape back to the system so that you can check the artwork files that are on the tape by using the |PENPLOT ARTWORK| command.

We recommend that you do this before sending the tape to a vendor for photoplotting. This capability can also be helpful if a vendor has difficulty in photoplotting a tape and returns it to you.

PROCEDURE

Pick the |GERBER FROM TAPE| command. (Refer to the Command Description Section of the manual for the exact input sequence to use.)

The system will copy the file (or files) to the current project and will give each file a name based on the file's sequence on the tape. For example, the first file on the tape will be named "TAPE1-ART" in the current project file, the second file on the tape will be named "TAPE2-ART", etc.

8. USING THE |FILL PENPLOT| COMMAND (OPTIONAL)

The |FILL PENPLOT| command allows filling of the various line widths and pad shapes in your PC board design. |FILL PENPLOT| reads the artwork text file (-ART) and plots it on the HP Pen Plotter.

Drawings that are plotted with the |FILL PENPLOT| command may be used for your PC board fabrication. (See |FILL PENPLOT| LIMITATIONS)

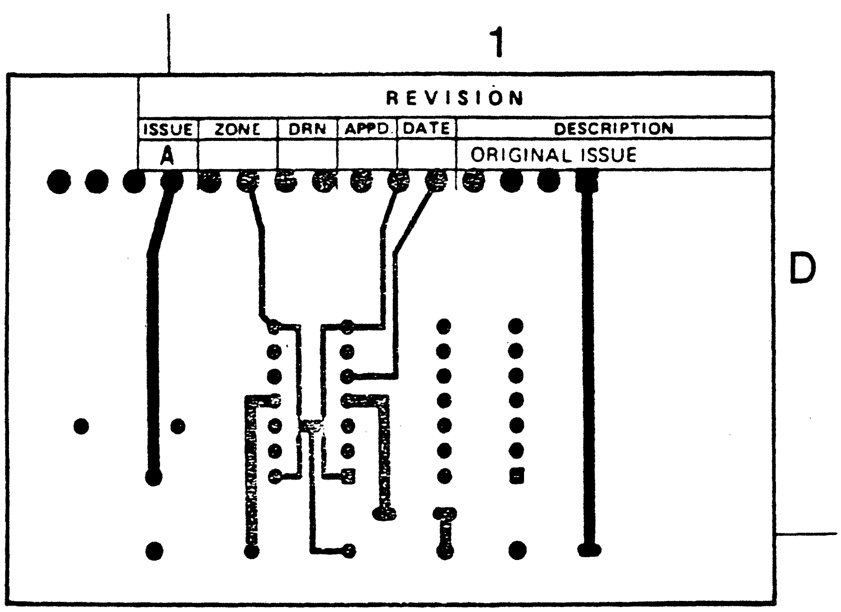
PREREQUISITES FOR THE |FILL PENPLOT| COMMAND

The prerequisites for the |FILL PENPLOT| command are the same as those for PENPLOT ARTWORK.

The APERTURE-TAB text file, the PHOTOPLOT-PAR text file and the artwork file (-ART) must be in the current project or in the SYSTEM LIBRARY.

|FILL PENPLOT| allows you to use any number of pens (up to 8) from the HP plotter carousel. However, you must input at least one pen width when using the |FILL PENPLOT| capability.

EXAMPLE OF A FILLED PEN PLOT



INPUT SEQUENCE

STEP 1. Pick |FILL PENPLOT|

The system prompts you for the artwork file name (-ART), and the plotting scale you wish to use. The artwork file was created when you executed the |CREATE PHOTOPLOT| command.

EXAMPLE: |FILL PENPLOT|

(ENTER ARTWORK FILENAME) FILM-1-ART |ENTER| (SCALE) 1 |ENTER|

STEP 2. You must now input a width for each pen in the HP Plotter carousel.

EXAMPLE: (width of pen 1 [in mils] 31 |ENTER|

(width of pen 2 [in mils] 100 |ENTER|

If there are positions on the carousel that do not have pen assignmnts, pick |ENTER| when the system prompts you for those particular pens. |ENTER| automatically assigns a zero plotting width and overrides those pen positions.

STEP 3. |ENTER|

After entering the eight pen widths for the HP Plotter carousel, pick |ENTER| to start the pen plot.

IMPORTANT CONSIDERATIONS WHEN USING THE |FILL PENPLOT| COMMAND

1. You should check the PHOTOPLOT-LOG file before selecting the pen widths to be used in the plot. The PHOTOPLOT-LOG file lists all the aperture sizes contained in the artwork (-ART) file. Selecting a variety of pen widths that best fit the apertures will allow faster "filling" of the plot.

2. You must input at least one pen width when using |FILL PENPLOT|. The error message "MUST INPUT AT LEAST ONE PEN WIDTH" will appear on the function screen if you do not define at least one pen width. Try to use a variety of pen widths in your pen plot.

3. If you enter a negative (-) pen width, the systems displays the error message "CANNOT HAVE A NEGATIVE PEN WIDTH".

4. The smallest pen width must be less than or equal to the smallest aperture size. You will receive an error message if you have any apertures that are smaller than any of your pen widths.

   "APERTURE SIZE TOO SMALL FOR PEN WIDTH"

   Check the PHOTOPLOT-LOG file to examine the apertures that are used in the artwork file (-ART). Your APERTURE-TAB file may have an aperture that is too small for the pen widths that you input.

   NOTE: If you plot the artwork file at a scale other than 1, you must adjust your pen widths accordingly. For example, if your plot scale is 2, you must multiply the smallest aperture in your APERTURE-TAB file by a factor of 2. You must then compare the result with the smallest pen width. It may be necessary to change the pen widths in the HP plotter carousel to suit the plot scale you have selected.

5. If you have a number of different line and pad sizes in the artwork file that you wish to plot at one size, you may assign the same station and "D" code for those apertures. It is not necessary to change the sizes in the drawing. You may change the APERTURE-TAB file so that apertures of different sizes are assigned to the same station.

   EXAMPLE: APERTURE-TAB file

   ```
   (STA   APER        CODE)
   (1)  LINE=.100     D10
   (1)  LINE=.200     D10
   (1)  LINE=.250     D10
   ```

   When you use the |FILL PENPLOT| command, the system reads the first aperture (LINE=.100 D10). All apertures assigned to D10 will be pen plotted at a width of .100.

|FILL PENPLOT| LIMITATIONS

You may use the |FILL PENPLOT| command to create pen plots that could be used for your PC board fabrication. The |FILL PENPLOT| capability allows you to bypass the Gerber photoplotting requirement of your production process. We recommend that you use this capability with board designs of "low density". "Low density" drawings allow the highest degree of plotting accuracy with the Telesis system and the HP Pen Plotter. We recommend a plotting scale of 2:1.

However, the following limitations must be considered if the "filled" pen plot is to be used for the PC board fabrication:

1. The actual ink photoplotting widths may be DIFFERENT from the pen widths that you input. Plotting widths may also vary with the type of plotting surface and the type of pens (ink, felt tip) used in the plotter carousel.

   For example, an ink pen width of .030 may pen plot a width of .025 on a mylar surface. You must determine if the variance (+ OR -) in plotting width is acceptable for your board fabrication.

   It is advisable that you "test plot" the various pen widths to determine your exact plotting results. You may use an optical comparator to check and compare plotting widths.

   If variances are unacceptable, and you wish to use the filled pen plot capability, load the HP Plotter carousel with your "tested" pen widths to suit your plotting specifications.

2. The system recognizes lines, circles, rectangles, squares, and oblongs when you use |FILL PENPLOT| .

NUMERICAL CONTROL DRILL

1. GENERAL INFORMATION

2. HOW TO INPUT THE NCDRILL-PAR TEXT FILE

3. HOW TO INPUT THE NCDRILL-FIG TEXT FILE

4. HOW TO USE THE ADD DRILL LEGEND COMMAND

5. HOW TO USE THE CREATE NC DRILL COMMAND

6. PEN PLOTTING OR PHOTOPLOTTING

7. HOW TO USE THE PUNCH DRILL TP COMMAND

1. GENERAL INFORMATION

PURPOSE

The Telesis NC DRILL capability accomplishes two things:

1. IT AIDS IN PRODUCING A FABRICATION DRAWING

   The NC DRILL capability automatically places a drill legend table on your board drawing, and places user-defined holesize figures at each drill coordinate in the drawing. These can be pen plotted or photoplotted in a fabrication drawing.

2. IT PUNCHES AN NC DRILL TAPE

   If you have an optional PAPER TAPE READER/PUNCH with your Telesis system, you can use the NC DRILL capability to punch an NC DRILL tape. This tape can be used with EXCELLON FORMAT 2 NC drill machines.

SUMMARY OF THE STEPS INVOLVED IN THE NC DRILL PROCESS

Below is a brief explanation of the steps you should follow in completing the NC DRILL process on the Telesis system. Details are given on the following pages.

STEP 1 INPUT AN NCDRILL-PAR TEXT FILE

This is the NC DRILL parameter file. It specifies the parameter settings required by the PAPER TAPE READER/PUNCH machine and the NC DRILL machine.

STEP 2 INPUT AN NCDRILL-FIG TEXT FILE

This file defines the size and shape of the holesize figures to be added to the board drawing.

STEP 3 USE THE |ADD DRILL LEGEND| COMMAND (optional)

This command places a drill legend header at a location that you specify in your board drawing. When you use the |CREATE NC DRILL| command, the system places the drill legend table under this header.

STEP 4 SELECT THE NC DRILL COMMAND

This command opens up the NC DRILL menu pages.

STEP 5 SELECT THE CREATE NC DRILL COMMAND

This command does the following :

Constructs a drill legend table in your drawing.

Places holesize figures in your drawing.

Produces a text file named NCDRILL-LOG listing errors.

Produces a text file named NCDRILL-TAPE that can be used to punch an NC DRILL tape.

STEP 6 PEN PLOT OR PHOTOPLOT THE NC DRILL DRAWING

STEP 7 SELECT THE PUNCH DRILL TP COMMAND

This command punches the NC DRILL tape.

2. HOW TO INPUT THE NCDRILL-PAR TEXT FILE

PURPOSE

This text file specifies parameters, some of which will be used by the PAPER TAPE READER/PUNCH machine (e.g., leader length) and some of which will be used by the NC DRILL machine (e.g., machine offset).

EXAMPLE

```
CODE ASCII
UNITS:ENGLISH
ENGLISH
ABSOLUTE
FORMAT 2.3
TRAILING-ZEROS NOT-SUPPRESS
LEADER 10 (ten inches of leader between all blocks)
MACHINE-OFFSSET 2.25,1.5
HEADER AMPLIFIER BOARD REV 1.0   10-NOV-82
END
```

NAME AND LOCATION OF THE FILE

This file must be named NCDRILL-PAR, and it must be in the current project file or the SYSTEM LIBRARY. NCDRILL-PAR is optional. If omitted, default parameter values will be used.

REQUIRED CONTENTS AND FORMAT OF THE FILE

Each parameter must be placed at the beginning of a new line.

NOTE: To know the proper values to input for many of these parameters, you may need to contact the drill operator.

You may omit any parameter from the file if you wish to use the default value shown below.

CONTENTS	FORMAT
CODE	FORMAT: CODE EIA or CODE ASCII EXAMPLE: CODE ASCII Input at least one blank space after the keyword: CODE; and no blank space in the words: EIA or ASCII. DEFAULT: EIA

(table continued from previous page)

ENGLISH or METRIC UNITS (set the unit of measure for the machine offset or leader length until the next UNITS keyword occurs)	FORMAT: UNITS:ENGLISH or UNITS:METRIC EXAMPLE: UNITS:METRIC DEFAULT: UNITS:ENGLISH
ENGLISH or METRIC UNITS (for output)	FORMAT: ENGLISH or METRIC EXAMPLE: ENGLISH DEFAULT: ENGLISH
FORMAT OF COORDINATES	FORMAT: FORMAT m.n EXAMPLE: FORMAT 2.4 - Input at leasts one blank space after the keyword: FORMAT. - m indicates the number of places before the decimal point. n indicates the number of places after the decimal point. - Input a period (.) between m and n with no blank spaces. - 2.3 and 2.4 are the only allowable values with ENGLISH coordinates. - 3.2 and 4.2 are the only allowable values with METRIC coordinates. DEFAULT: 2.3
ABSOLUTE or INCREMENTAL COORDINATES	FORMAT: ABSOLUTE or INCREMENTAL EXAMPLE: ABSOLUTE DEFAULT: ABSOLUTE
SUPPRESS OR NOT-SUPPRESS TRAILING ZEROS IN COORDINATE VALUES	FORMAT: TRAILING-ZEROS SUPRESS OR TRAILING-ZEROS NON-SUPPRESS EXAMPLE: TRAILING-ZEROS SUPPRESS - Input a hyphen (-) and no blank spaces in the keyword: TRAILING-ZEROS. - Input at least one blank space after the keyword: TRAILING-ZEROS. - Input a hyphen (-) and no blank spaces in the word: NOT-SUPRESS. DEFAULT: SUPPRESS

(table continued from previous page)

OFFSET OF THE 0,0 OF YOUR BOARD FROM THE 0,0 OF THE NC DRILL MACHINE	FORMAT: MACHINE-OFFSET x,y EXAMPLE: MACHINE-OFFSET 2.25,1.5 Input a hyphen (-) and no blank spaces in the keyword: MACHINE-OFFSET. Input at least one blank space after the keyword: MACHINE-OFFSET. x is the x offset. y is the y offset. Input a comma (,) between x and y with no blank spaces. If you placed a decimal point in the number, the system assumes either inches or millimeters depending on which input unit you chose. If you did not place a decimal point in the number then the unit for the number is mils if you chose "English" as the input unit and board database units if you chose "metric" as the input unit. DEFAULT: 0,0
HEADER TEXT: TEXT TO BE PUNCHED IN MAN-READABLE CHARACTERS ON THE LEADER OF THE PUNCHED TAPE	FORMAT: HEADER (text) EXAMPLE: HEADER AMPLIFIER BOARD REV 1.0 10-NOV-82 Input at least one blank space after the keyword: HEADER. We recommend that the text include board name and revision, part number, and date. You may use up to 70 characters including blank spaces. DEFAULT: 70 blank characters.

(table continued from previous page)

LEADER LENGTH: Length of the paper tape leader, and length of the blank space between blocks of punched tape	FORMAT: LEADER n EXAMPLE: LEADER 10 Input at least one blank space after the keyword: LEADER. n is a number in inches or millimeters (depending upon which input unit you chose.) DEFAULT: 12
COMMENTS (optional) YOU MAY INCLUDE INFORMATIONAL COMMENTS IN THIS FILE	FORMAT: (text of comment) EXAMPLE: (NDCRILL-PAR FILE FOR AMPLIFIER BOARD) Place comments in parentheses. The comment text may be anything you wish. The system ignores all information placed in parentheses in text file.
END STATEMENT	FORMAT: END EXAMPLE: END Input the word END as the last line of the file.

3. HOW TO INPUT THE NCDRILL-FIG TEXT FILE

PURPOSE

This text file tells the system what figure to place in your drawing for each holesize/plating-type in your design.

EXAMPLE 1

```
HEXAGON=.070
DIAMOND=.090
TRIANGLE=.100
RECTANGLE=.075X.060
TEXT=.080X.100
.039-P  HEXAGON
.040-N  DIAMOND
.052-0  TEXT  A
.060-N  DIAMOND=.125
END
```

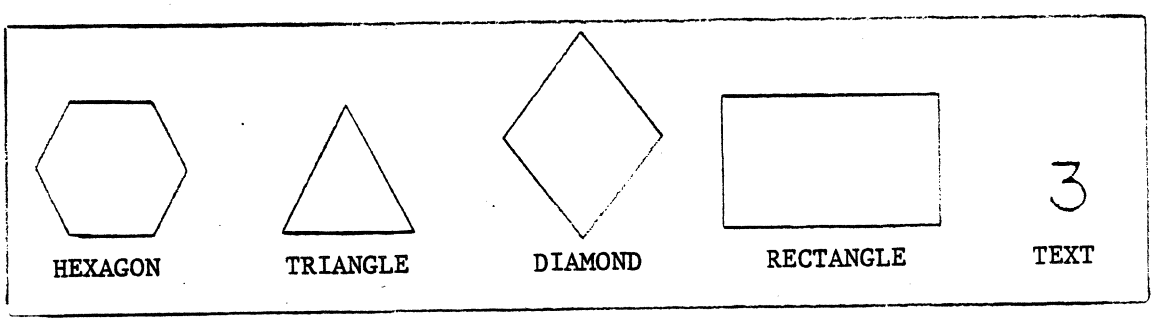
EXAMPLE 2

```
.039-P  HEXAGON=.100           P
.060-N  RECTANGLE=.150X.100    X
.039-N  DIAMOND=.080           A
.020-P  TEXT=.095X.095         8
.075-N  TRIANGLE=.120          Q
END
```

NAME AND LOCATION OF THE FILE

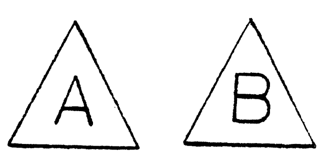
This file must be named NCDRILL-FIG, and it must be in the current project file or the SYSTEM-LIBRARY.

HOLESIZE FIGURES YOU MAY USE



To increase the variety of available figures, you may place a text character inside any geometric shape.

EXAMPLES:



REQUIRED CONTENTS AND FORMAT OF THE FILE

CONTENTS	FORMAT
INPUT UNITS FOR FIGURE and DRILL SIZES	FORMAT: UNITS:ENGLISH EXAMPLE: UNITS:ENGLISH DEFAULT: UNITS:ENGLISH When you input a number for size, if a decimal points is part of the number, the system assumes either inches or millimeters (depending on the input unit you chose). If this number has no decimal point, the system assumes mils for "English" units input and board database units (.01mm) for "metric" input.
LIST OF FIGURE SHAPES and SIZES (optional): This list specifies the size you wish to use for each figure shape when it is placed on the board. These values will only be used by the system if you do not specify figure sizes in the hole-size list below.	FORMAT: (figure)=(size) EXAMPLE: HEXAGON=.070 Each figure must be placed at the beginning of a new line. Figure may be one of the following words: HEXAGON, DIAMOND, TRIANGLE, RECTANGLE, TEXT. Figure size follows the figure separated by an equals (=) sign and no blank spaces. Figure size is in inches. Input one value (height) for HEXAGON, DIAMOND, or TRIANGLE. Input two values (width and height) for RECTANGLE or TEXT. Use this format: (figure)=(width)(height). Ex. RECTANGLE=.150X.100 TEXT=.075X.100 Use no blank spaces before or after the X.
LIST OF HOLESIZES AND FIGURES: This list includes the holesize/plating-type combinations you have used in your drawing. For each you input the figure shape and size you want to be placed in the board drawing. You may omit any	FORMAT: (holesize)-(plating-type) (figure)=(size) (character) EXAMPLE: .039-P HEXAGON=150H HOLESIZE: Each holesize must be placed at the beginning of a new line. Place a hyphen (-) between holesize and plating-type with no blank spaces. Holesize/plating-type must be the same as shown in the pin description file.

holesize/plating-type combination from this list if you do not want a figure to appear in the drawing for it. (The omitted holesize/plating-type will still be shown in the drill legend table and included in the NCDRILL-TAPE)	Input at least one blank space after plating-type. FIGURE: Input a figure shape and size for each holesize/plating-type combination. Use the same format as in the list above. The figure size is optional in this list. If you input a figure size here, it will take precedence over the figure size (if any) you specified in the list above. (This allows you to vary the figure size used for any individual holesize/plating-type.) DEFAULT FIGURE SIZES: If you do not input a figure size here, or in the list above, the system will use these default figure sizes (given in inches): HEXAGON=.060 DIAMOND=.060 TRIANGLE=.060 RECTANGLE=.060X.060 TEXT=.030x.045 CHARACTER: You must input a single keyboard character at the end of the line if you have used the figure: TEXT. If you have used any of the geometric figures, this character is optional. (The system will place the character inside the geometric shape.) Input at least one blank space before the character.
END STATEMENT	FORMAT: END EXAMPLE: END Input the word END as the last line of the file.
COMMENTS: (optional) You may include informational comments in the file	FORMAT: (text of the comment) EXAMPLE: (NCDRILL-FIG FILE FOR AMPLIFIER BOARD) Place comments in parentheses. The comment text may be anything you wish. The system ignores all information placed in parentheses in text files.

4. HOW TO USE THE |ADD DRILL LEGEND| COMMAND

PURPOSE

This command places the text of a drill legend header at a point that you specify in your drawing. When you subsequently use the |CREATE NC DRILLL| command, the system places the drill legend table under this header.

This command is optional. If you do not use it, the system will add the header and table simultaneously at the time that you use the |CREATE NC DRILL| command.

PREREQUISITE

- The board drawing must be active.

HOW THE COMMAND WORKS

(Refer to the Command Description Section of the manual for the precise input sequence you must use.)

When you pick |ADD DRILL LEGEND|, and pick a location, the system places this text in your drawing:

/FIGURE/HOLESIZE/ QTY/

The text is placed on the active layer, and it has the currently set text parameters. We recommend that you set active layer to 98. Do not place the header on layer 99. (The system deletes information on layer 99 when you pick the |CREATE NC DRILL| command.).

YOU MAY CHANGE THE HEADER TEXT

The text of the drill legend header might not conform to your company's standard, but we recommend that you do not change the text until after you have used the |CREATE NC DRILL| command. If you do use the |UPDATE TEXT| command before using the |CREATE NC DRILL| command, you may change only the order of the keywords, for example: / QTY/HOLESIZE/ FIGURE/; you may not change the keywords, the slashes, or the spacing.

After using the |CREATE NC DRILL| command to place the drill legend table and holesize figures in your drawing, you may change the header and table to any text or format you wish.

5. HOW TO USE THE |CREATE NC DRILL| COMMAND

PURPOSE

When you use the |CREATE NC DRILL| command, the system places a drill legend table and holesize figures in your drawing; creates a text file that can be used with the |PUNCH DRILL TP| command; and creates a text file NCDRILL-LOG listing errors detected by the system.

PREREQUISITE

- o Pin description files for every component on the board must be in the current project file or in the SYSTEM-LIBRARY.

- o The board drawing must be complete. All connect points, that have drill information specified in a pin file, must have pin numbers in the board drawing. We recommend that you add title blocks and dimensioning for a fabrication drawing to the board drawing before you use the |CREATE NC DRILL| command. (You may add these after using the |CREATE NC DRILL| command, but the system will be slowed down by the presence of holesize figures in the database.)

- o The board drawing must be active. We recommend that you complete the board drawing, save it, and then reopen it with the |OLD DRAWING| command. (There will then be an identical, saved version of your drawing that can be used if you must repeat the |CREATE NC DRILL| command.)

- o The NCDRILL-PAR text file must be in the current project file or in the SYSTEM-LIBRARY. If omitted, default parameter values will be used.

- o The NCDRILL-FIG text file must be in the current project file or in the SYSTEM-LIBRARY.

HOW THE |CREATE NC DRILL| COMMAND WORKS

When you pick |CREATE NC DRILL|, there are three results:

1. DRILL LEGEND TABLE AND HOLESIZE FIGURES

The system places a drill legend table and holesize figures on layer 99 of your active board drawing.

EXAMPLE OF A DRILL LEGEND TABLE:

/FIGURE/	HOLESIZE/	QTY/
D	.062-P	3
⬡	.049-P	4
C	.188-N	6
B	.106-P	14
A	.112-N	18
□	.055-P	225
◇	.040-P	360
△	.039-P	420
☒	.028-N	421

The system places the drill legend table under the drill legend header that you added with the |ADD DRILL LEGEND| command. If you did not use the |ADD DRILL LEGEND| command, the header and table are placed with the upper right corner at a point 500 mils in the x and y axes from the upper right corner of the drawing.

2. NCDRILL-TAPE TEXT FILE

The system creates a text file named NCDRILL-TAPE. This is used to punch an NC DRILL paper tape when you pick the |PUNCH DRILL TP| command (see POST-77 for further discussion of the NCDRILL-TAPE file and the |PUNCH DRILL TP| command.)

3. NCDRILL-LOG TEXT FILE

The system creates a text file named NCDRILL-LOG. This file lists:

- o The parameters you input in the NCDRILL-PAR file, or the default parameters used by the system.
- o Error messages.
- o A drill set-up table.

EXAMPLE OF THE NCDRILL-LOG (example is abbreviated):

```
        M E S S A G E S     from     N C D R I L      13-DEC-82    10 17 34

EIA
   */ Incorrect key word in the line

   NCDRILL  PARAMETERS
   ------------------

HEADER :    TESTXXXXXXXXXXXXXXXXXXXX

EIA

ENGLISH

ABSOLUTE

TRAILING-ZEROS  SUPPRESS

MACHINE-OFFSET   X:      0. Y:      0

FORMAT               2.3

LEADER               -1 inch.
```

(example cont. on next page)

```
LEGEND-LABEL does not exist in drawing

   - Process LAYERSTD file
     --------------------



   - Process PIN files
     ----------------

PROCESSING VIA                  1


PROCESSING DIP14                1


PROCESSING DIP8                 1

ERROR(S) - Drill hole(s) not defined in the pin file(s) for
      1TO220             pin #      4      X =    6300     Y =     800


     1.  Duplicate pad - X :    3800 ;  Y :   6900
     2.  Duplicate pad - X :   10250 ,  Y :   6900


    PROCESSING NCDRILL-FIG FILE
    ---------------------------

Undefined drill symbol name. .038-P

     1   Duplicate pad - X :    3800 ;  Y :   6900
     2.  Duplicate pad - X :   10250 ;  Y :   6900


Undefined drill symbol name: .028-P

     1.  Duplicate pad - X :   13150 ,  Y :   1550
     2.  Duplicate pad - X :   13150 ,  Y :   2250
     3   Duplicate pad - X :    7650 ,  Y :   5050
     4   Duplicate pad - X :    7650 ;  Y     4350




     5.  Duplicate pad - X     1000 ,  Y     6675
     6   Duplicate pad - X :    1500 ,  Y     6675
     7   Duplicate pad - X :    3800 ,  Y .   6900
     8.  Duplicate pad - X :   10250 ;  Y :   6900


Undefined drill symbol name: .034-P


    DRILL  LEGEND  INFORMATION
    --------------------------

         HOLESIZE              QTY

     1.   125-N                  4

     2.  .056-P                  4

     3   .042-P                 13

     4   .038-P                 86

     5.  .028-P                785

     6.   034-P               2312
     -----------------------------
     TOTAL: . . . . . .       3204
```

TYPES OF ERROR MESSAGES YOU MAY RECEIVE IN THE NCDRILL-LOG

"INCORRECT CODE, EIA OR ASCII ONLY"

The system will use the default, EIA.

"INCORRECT KEYWORD IN LEGEND-LABEL. DEFAULT: FIGURE/HOLESIAE/ QTY"

The system will insert the default legend label and continue.

"NEED MORE SPACE FOR DRILL LEGEND. MOVED UP ON THE DRAWING"

The system will move the position of the label so it can fit all the drill legend entries.

"INCORRECT REQUEST FOR ZEROS. SUPPRESS/NOT-SUPPRESS ONLY"

The system will use the default, TRAILING-ZEROS SUPPRESS.

"INCORRECT FORMAT REQUEST"

The system will use the default, FORMAT 2.3

"INCORRECT LEADER REQUEST"

The system will use the default, LEADER 12 (twelve inches)

"INCORRECT MACHINE-OFFSET REQUEST"

The system will use the default, MACHINE-OFFSET 0,0.

"NOT ENOUGH ROOM FOR DRILL LEGEND"

The system will omit the entire drill legend if it will not fit. (NOTE: It is unlikely the drill legend would not fit on the PCB drawing unless there were unusual values for text size, or the drawing were very small compared to the text size set.)

"LEGEND-LABEL DOES NOT EXIST IN DRAWING"

The message is a warning. The system will put a drill legend label in the drawing 500 database units in x and y from the upper right corner of the drawing.

"NCDRILL-TAPE FILE NOT CREATED DUE TO ERRORS"

Correct the errors found earlier in NCDRILL-LOG file, and retry |CREATE NCDRILL|

"FILE DOES NOT EXIST: NCDRILL-PAR"

The system will use defaults for all values, and continue processing.

"FILE DOES NOT EXIST: NCDRILL-FIG"

The system will continue processing to find any other detectable errors, but it will not create NCDRILL-TAPE, nor put any drill figures or drill legend into the active PCB drawing.

"INCORRECT KEYWORD IN THE LINE"

This message is printed immediately below the line the system was unable to interpret. The line will be ignored.

"ERROR-DRILL HOLE NOT DEFINED FOR (symbol name) PIN # (number)

X = (x drawing coordinate) Y = (y drawing coordinate)

Either:

1) The pin file does not show a pin number for that connect point. (Check the pin file for correct spelling, and be sure the pin number is defined in the pin file.)

or

2) The DRILL keyword for that pin number is in the pin file, but the type is not -P, -N, or -0.

"UNDEFINED DRILL FIGURE NAME"

The system has found a holesize/plating-type for which you did not specify a figure shape in the NCDRILL-FIG file. The system will not place a figure in the drawing for this holesize/plating-type, but it will include the holesize/plating-type in the drill legend table and in the NCDRILL-TAPE as usual.

YOU SHOULD REPEAT THE |CREATE NC DRILL| COMMAND UNTIL THE NCDRILL-LOG IS ERROR-FREE

To produce an accurate punched tape and fabrication drawing, you should correct the errors shown in the NCDRILL-LOG, and repeat the |CREATE NC DRILL| command. (Some errors may not need to be corrected if the system has used an acceptable default value.)

For best results, follow these steps when using the |CREATE NC DRILL| command:

STEP 1 Pick |CREATE NC DRILL|

STEP 2 Print out the NCDRILL-LOG text file.

STEP 3 Cancel the active drawing if the NCDRILL-LOG shows errors that must be corrected. (We recommend cancelling the drawing because the holesize figures on it will slow down the system in any subsequent operation.)

STEP 4 Use the |OLD DRAWING| command to activate a saved version of the board drawing that has not yet been used with the |CREATE NC DRILL| command.

STEP 5 Correct errors in the drawing and in the prerequisite text files: Pin files, NCDRILL-PAR, NCDRILL-FIG.

STEP 6 Repeat the |ADD DRILL LEGEND| command.

STEP 7 Repeat STEPS 1-6 until the NCDRILL-LOG is free of unacceptable errors.

6. PEN PLOTTING OR PHOTOPLOTTING THE NC DRILL LAYERS

You may pen plot or photoplot the NC DRILL layers of your drawing in the same way that you would any other layers. (See p. PRINT/PLOT-7 and p. BOARD-33). The system placed the drill legend table and the holesize figures on layer 99. If you used the |ADD DRILL LEGEND| command, you placed the drill legend header on another layer (recommended: 98).

YOU MAY ADD TO THE DRAWING OR UPDATE IT BEFORE PLOTTING

We recommended that you add title blocks, dimensioning, and any other fabrication graphics before using the |CREATE NC DRILL| command. If you did not do so, you may do so before plotting. However, the system will respond more slowly than usual.

If the drill legend table does not conform to your company standard, you may use the |UPDATE TEXT| command to alter it before plotting.

WHEN FINISHED, CANCEL THE ACTIVE DRAWING OR ARCHIVE IT

We recommend that you use the |CANCL ACTV DRWNG| command when you have finished pen plotting or creating photoplot text files; or you may archive the drawing on floppy disk or magnetic tape.

7. HOW TO USE THE |PUNCH DRILL TP| COMMAND

PURPOSE

Use this command to punch an NC DRILL paper tape.

PREREQUISITES

- The NCDRILL-TAPE text file must be in the current project file or the SYSTEM-LIBRARY.
- The NCDRILL-PAR text file must be in the current project file or the SYSTEM-LIBRARY.
- The PAPER TAPE READER/PUNCH must be on line and not busy.

THE NCDRILL-TAPE FILE

When you picked |CREATE NC DRILL|, the system created the NCDRILL-TAPE text file containing information needed by the punch machine.

EXAMPLE (example is abbreviated)

```
X03800Y04600
X03950Y04850
X04100Y05050
X04200Y05350
X04450Y04500
M00

X03500Y03150
X03650Y03300
X03800Y03450
X03900Y03150
X03950Y03600
X04050Y03300
X04100Y03750
X04200Y03450
X04300Y03900
X04350Y03600
X04450Y04050
X04500Y03750
X04600Y04200
X04700Y03900
X04850Y04050
X05000Y04200
X04750Y06400
R07Y-00100
X04450Y05700
R06Y00100
X03050Y06300
X04450Y05200
R06Y-00100
X04750Y04600
R06Y00100
X03050Y05200
R06Y-00100
X03350Y04600
R06Y00100
M00
```

The file contains a block of coordinate values for each holesize. (The R (repeat) code is used where appropriate.) The first block corresponds to the first holesize line in the drill legend table; the second block corresponds to the second holesize line; and so on. There will be a separate block for each holesize/plating-type combination, even if the holesizes are the same. For example, the block for .039-P will be separate from the block for .039-N.

At the end of each block is the tool code: MOO. The NCDRILL-TAPE file may be edited so as to replace the MOO code with the actual tool code required. Edit it as you would any other text file.

HOW THE |PUNCH DRILL TP| COMMAND WORKS

When you pick |PUNCH DRILL TP|, the system punches an NC DRILL paper tape for your board design. The tape may be used with EXCELLON FORMAT 2 drill machines.

PRINTER, PLOTTER, DIGITIZER

- USING THE MATRIX PRINTER
- USING THE PEN PLOTTER
- OPERATING THE PEN PLOTTER IN BACKGROUND
- DIGITIZER

USING THE MATRIX PRINTER

1. GENERAL INFORMATION

The matrix printer is optional equipment with the Telesis system. It has two uses:

1. To print text files and long messages.
2. To plot a drawing.

When using the matrix printer option to print text files and long messages, the system allows you to continue your work during the printing operation. When the system "beeps" while the printer is operating, you may pick a new menu box and continue your work on the Telesis system.

NOTE: If you are using the matrix printer to plot a drawing, wait until the printer operation has STOPPED before continuing your work.

USING THE |CANCEL PRINT| COMMAND

You may use the |CANCEL PRINT| command to cancel the printing of a text file. When |CANCEL PRINT| is picked, the printer may continue to operate for a short period of time before stopping. When the system "beeps", you may then continue using the system interactively. However, |CANCEL PRINT| does not cancel long messages being output to the matrix printer.

When |CANCEL PRINT| is picked, the function screen message line displays:

<BACKGROUND MATRIX PRINT CANCELLED>

2. PRINTING TEXT FILES

You must open a project file to use the matrix printer.

You can print any text file in the current project file. This includes both text files created by you, and text files created by the system (such as the bill of materials report).

GETTING TO MENUS WITH PRINT-TEXT-FILE COMMANDS

IF YOU HAVE AN OPEN DRAWING: when you open a drawing, the first menu to appear is the DRAWING TASK MENU. Pick the |PRINT OR PLOT| command on this menu to flip to the PRINT OR PLOT MENU where the PRINT commands are located.

IF YOU DO NOT HAVE AN OPEN DRAWING: when you open a project file, the first menu to appear is the MENU OF MENUS. Pick the |TEXT FILES| command on this menu to flip to the TEXT TASK MENU where the PRINT commands are located.

COMMANDS USED TO PRINT TEXT FILES

COMMAND	WHEN TO USE IT
|PRINT TEXT FILE|	Use this command to print any text file in the current project file. You must follow this command with the exact name of the text file. You may specify a revision of the text file as well as the name. If you do not, the most recently used revision will be printed. Use the |CURRENT INDEX| command to display the contents of the current project file, if you are not sure of the exact name or revision you want.
|PR NETLIST-RPT|	PRINT NETLIST REPORT: The command is a short cut for printing the text file named NETLIST-REPORT. It is not necessary to follow this command with the name of the text file.

PR COMPONENT-RPT	PRINT COMPONENT REPORT: This command is a short cut for printing the text file named COMPONENT-REPORT. It is not necessary to follow this command with the name of the text file.
PR BOM-REPORT	PRINT BOM REPORT: This command is a short cut for printing the text file named BOM-REPORT. It is not necessary to follow this command with the name of the text file.
PR PHOTOPLOT-LOG	PRINT PHOTOPLOT LOG: This command is a short cut for printing the text file named PHOTOPLOT-LOG. It is not necessary to follow this command with the name of the text file.
PR EXTRCTION-LOG	PRINT EXTRACTION LOG: This command is a short cut for printing the text file named EXTRACTION-LOG. It is not necessary to follow this command with the name of the text file.
PR NET-COMPR-RPT	PRINT NET COMPARE REPORT: This command is a short cut for printing the text file named NET-COMPARE-REPORT It is not necessary to follow this command with the name of the text file.

3. PRINTING A MATRIX PLOT OF A DRAWING

You must open a drawing to make a matrix plot of it.

Ordinarily, you use a matrix plot as a check plot of your work.

GETTING TO THE MENU WITH THE |MATRIX PLOT| COMMAND

When you open a drawing, the first menu to appear is the DRAWING TASK MENU. Pick the |PRINT OR PLOT| command on this menu to flip to the PRINT OR PLOT MENU where the |MATRIX PLOT| command is located.

THE |MATRIX PLOT| COMMAND

The |MATRIX PLOT| command is described in the Command Description Section of the manual.

Here are some general principles to keep in mind when you use this command.

CONTENTS OF THE PLOT

The contents of the plot will be the same as the contents of the graphics screen. If you want to plot the whole drawing, use the |DRAWING| command to display the whole drawing on the graphics screen. If you want to plot a smaller area of the drawing, |WINDOW|-in until the graphics screen displays exactly what you want to plot.

SOLID LINES AND DASHED LINES

Because the plot is in black and white, you lose the color distinctions shown on the graphics screen. You can, however, retain some of this distinction by specifying that some colors be plotted as dashed lines. The colors you do not specify will be plotted as solid lines.

SPECIFYING SCALE

Remember that the scale you specify is in relation to the actual size of your drawing (SIZE A,B,C,D or OTHER). The scale you specify has no relation to the size of your drawing as it appears on the graphics screen.

CUT-LINES

Down the left and right sides of the plot are dashed lines. These are cut-lines to help you if you are piecing together a drawing from more than one matrix plot.

CONSIDERABLE FREE DISK SPACE IS NEEDED FOR A MATRIX PLOT

The system requires considerable free disk space to create a matrix plot: approximately 2000 free blocks for a D SIZE drawing. You may temporarily copy files to floppy disks and then delete them from the system in order to free disk space for matrix plotting. When the plot is finished, you may copy these files back to the system.

4. PRINTING MESSAGES DISPLAYED ON THE FUNCTION SCREEN

The Telesis system allows you to print long messages that are displayed on the function screen. For example, you may output a CURRENT INDEX to the matrix printer if you wish to produce a hardcopy version.

You may use the |PRINT| command located at the top of the function screen to print a long message or an index.

PREREQUISITES

1. The matrix printer must be on-line.
2. The long message or index to be printed must be displayed on the function screen.

Even if the message is longer than can be shown on the function screen, the message is printed in its entirety. For example, if the CURRENT INDEX comprises three function screen pages, the entire index is printed regardless of the page currently displayed.

A message is printed EXACTLY as it appears on the function screen. If the displayed message is longer than one page, the printed output identifies the pages in sections rather than printer form feed. However, the printer form feeds after one printer page is full.

EXAMPLE: Function screen display of a CURRENT INDEX

			PRINT	<- PAGE	PAGE ->	CANCEL

Contents of project ACC-TEST rev 1

text file PENPLOT-CON rev 1	created	09-SEP-82	08:10:48
text file PDRC-FILE rev 1	created	09-SEP-82	08:10:56
text file LAYERSTD rev 1	created	09-SEP-82	08:11:00
text file CAP rev 1	created	02-FEB-83	09:13:05
drawing SCH rev 1	created	03-FEB-83	14:25:09
drawing PCB rev 1	created	08-FEB-83	12:02:50
file NET-DATA-BASE rev 1	created	08-FEB-83	12:09:06
symbol PC/RES rev 1	created	10 MAR-83	13:06:42
drawing AND3B rev 1	created	12-MAR-83	09:20:30
file NET-DATA-BASE rev 2	created	14-MAR-83	10:50:34
drawing PCB rev 2	created	18-MAR-83	09:32:40

When you pick the |PRINT| command at the top of the function screen, the CURRENT INDEX is printed EXACTLY as it appears on the function screen.

USING THE PEN PLOTTER

1. INTRODUCTION
2. HOW THE |PEN PLOT| COMMAND OPERATES
3. USING THE PEN PLOT CONTROL FILE
4. CREATING THE PEN PLOT CONTROL FILE
5. EDITING THE PEN PLOT CONTROL FILE
   - o PENPLOT-CON FILES NOT CREATED WITH THE EDIT PENPLOT-CON LEAD-THRU
6. PLOT SCALE
7. OPERATING THE |PEN PLOT| COMMAND
8. TROUBLE SHOOTING
9. PLOTTER LIMITATIONS

1. INTRODUCTION

The Telesis |PEN PLOT| command plots the presently displayed area (or window) of the active drawing. The commands on the world menu allow the operator to selectively control the view of the drawing to be plotted. The system plots the area of the drawing actively displayed on the graphics screen, plotting all visible layers, "skip" layers, and those layers blanked by color from the display menu.

The Telesis pen plot feature operates with the Hewlett-Packard HP 7580 series pen plotters. Before using the |PEN PLOT| command, the operator should be familiar with the following Hewlett-Packard Manuals:

"HP7580 Drafting Plotter Operator's Manual"

or

"HP7585 Drafting Plotter Operator's Manual"

These manuals explain how to power-on the plotter, pen and plotting media set-up, and how to activate the REMOTE state. This is all that is required for the Telesis PEN PLOT feature to work.

An in-depth understanding of the plotter can be acquired from the following document:

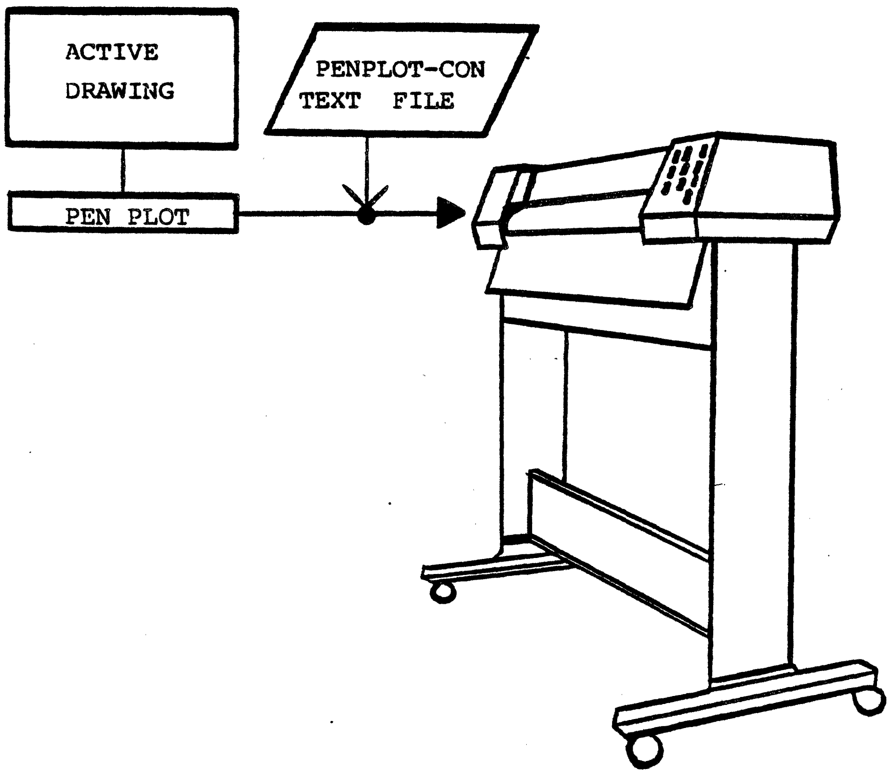
"Hewlett Packard 7580 (or 7585) Drafting Plotter Option 001 and 002 Operating and Programming Manual Using HP-GL Instructions."

However, this is not necessary for operation on the Telesis system.

This section of the manual describes the creation and editing of the pen plot control file, which is a requirement for plotting specific data base layers of a drawing with selected pens in the HP pen plotter carousel.

2. HOW THE PEN PLOT COMMAND OPERATES

The diagram below shows the flow of data during the PEN PLOT command. The Telesis system plots the actively displayed area of the drawing. The pen plot control file (if present) directs the HP7580 (or 7585) in plotting specific data base layers with specific pens. If a visible layer is not assigned to any pen in the control file, the system plots that layer with PEN 1 of the plotter carousel. Without a pen plot control file, all drawing layers will be plotted with this pen.



HP PEN PLOTTER

3. USING THE PEN PLOT CONTROL FILE

PURPOSE

The penplot control file instructs the plotter during the PEN PLOT operation. The plotter receives drawing information from the Telesis system and plots each drawing layer with the specified pen. The operator must create the penplot control file to plot specific layers of a drawing with a specific pen and pen color.

The HP plotter carousel holds a maximum of eight pens. The operator may use any number of pens to plot a drawing with a variety of colors.

NAMING THE PEN PLOT CONTROL TEXT FILE

When the operator picks PEN PLOT, the system prompts the operator for input of the penplot control file name. The system will then look for a text file of that name, first in the current project file, then in the SYSTEM-LIBRARY.

When creating a pen plot control file using the EDIT PENPLOT-CON text lead-thru, the operator may specify any name with up to 18 alphanumeric characters. However, if |ENTER| is picked without specifying a file name, the system automatically names the pen plot control file PENPLOT-CON.

By allowing the system to automatically name the control file, the operator does not need to specify a text file name when the PEN PLOT command is picked. The system will first look for the text file named PENPLOT-CON in the current project, then in the SYSTEM-LIBRARY.

NOTE: If the system cannot find the text file, it reports "NO PENPLOT CONTROL FILE" on the function screen message line. The system will draw with the pen plot with all drawing layers assigned to PEN 1 of the plotter carousel. The operator may terminate the command by picking CANCEL if desired.

EXAMPLE OF A PEN PLOT CONTROL FILE

The text file shown below is a typical pen plot control file:

```
                 PENPLOT-CON     1

      (PENPLOT-CON)
      PEN 1  DBLAYERS 0-3, 7, 8, 10
      PEN 2  DBLAYERS 4:6
      PEN 3  DBLAYERS 5  64
      PEN 4  DBLAYERS 15, 17, 33-35
      PEN 5  DBLAYERS 110-115  104
      PEN 6  DBLAYERS 40:44   88
      PEN 7  DBLAYERS 89-91
      PEN 8  DBLAYERS 146-147
      CONNECT-POINTS
      TEXT-POINTS
      BULLETSIZE 100 PENWIDTH  39
      END
```

CONTENTS OF THE PENPLOT-CON TEXT FILE

The PENPLOT-CON file directs the Hewlett-Packard system during execution of the |PEN PLOT| command. Each pen specified in the file is dedicated to the data base layer(s) listed after the pen number. Connect points and text points will be plotted if the keywords are included in the text file. In addition, the operator may specify bullets, that is, circles in place of connect-points when plotting a schematic drawing. Bullets may also be filled during plotting with the presence of the keyword PENWIDTH following the BULLETSIZE specification.

4. CREATING A PENPLOT CONTROL FILE WITH THE TEXT LEADTHRU

The text leadthru capability allows the operator to create a penplot control file on the function screen, with the system maintaining the proper formatting requirements for each line entry. The operator simply uses the interactive commands on the |EDIT PENPLOT-CON| leadthru to input and edit the information contained in the file. The leadthru reduces the possibility of numerous formatting errors that could result when the text file is created on a text editor keyboard.

When using the |EDIT PENPLOT-CON| leadthru, the system assists the operator in the following ways:

1. Automatically names the file PENPLOT-CON.

2. Automatically positions the graphics screen cursor when the operator inputs information.

3. Automatically inputs the keywords required in the file and prompts the operator for input.

4. Automatically places parentheses ( ) around comment line entries.

5. Issues error messages if the operator-defined input is improperly formatted.

   NOTE: Error messages may appear on the function screen if the penplot control file was not created with the |EDIT PENPLOT-CON| command. (See PRINT/PLOT-25).

The operator may use the steps described below to create a penplot control file using the leadthru capability:

STEP 1. TEXT LEADTHRU

Pick the |TEXT LEADTHRU| command to display the menu of leadthru options:

|EDIT PENPLOT-CON|

|EDIT DEVICE FILE|

|EDIT APERTURE-TAB|

|EDIT PIN FILE|

|EDIT LAYERSTD|

STEP 2. EDIT PENPLOT-CON

Pick the |EDIT PENPLOT-CON| command to open an existing penplot control file, or to create a new file. When the system issues the FILE NAME prompt, the operator may input the name of the penplot control file, or simply pick |ENTER|. Skipping the keyboard input will open the latest existing file named PENPLOT-CON or open a new file of the same name.

However, if an existing file is named other than PENPLOT-CON, the operator must specify the name of the file when |EDIT PENPLOT-CON| is used. If a new file is to be created with a name other than PENPLOT-CON, the operator may input up to 18 alphanumeric characters with no blank spaces. A revision label with up to four alphanumeric characters may be specfied; separate with the file name from the revision label with a blank space.

EXAMPLE: Creating a new file --

|EDIT PENPLOT-CON| (FILE NAME) ENTER

The system displays the new PENPLOT-CON file on the graphics screen with the information shown below:

```
PENPLOT - CON

___ (PENPLOT-CON)
    END
```

NOTE: If additional penplot control files are essential in the same project file, the operator may specify the PENPLOT-CON file name with a revision label different from existing PENPLOT-CON files.

The operator may use the following commands to add information between (PENPLOT-CON) and the END statement:

|ADD PEN LINE|

|ADD BULLETSIZE LINE|

|TEXT-POINTS|

|CONNECT POINTS|

|ADD COMMENT|

STEP 3. ADD PEN LINE

Use the |ADD PEN LINE| command to add pen and dblayer information to the file. The dblayers specified after each pen will be plotted with the pen specified during the |PEN PLOT| command. Continue the |ADD PEN LINE| command until all pens to be used for plotting have been specified.

Each pen assignment line starts with the keyword "PEN," followed by a pen number, followed by the keyword "DBLAYERS," followed by a list of layers assigned to that pen.

EXAMPLE:

ADD PEN LINE (PEN 1-8) -Keyboard- Input ENTER (DBLAYERS) -Keyboard- Input ENTER

The pen number must be a number from 1 through 8. Each layer in the list of layers must be a number between 0 and 255 inclusive. Each layer number in the list may be separated by any number of spaces.

An entire range of layers may be specified by inserting a hyphen (-) or a colon (:) between the selected range.

EXAMPLE: PEN 1 DBLAYERS 0-3
PEN 2 DBLAYERS 4:6

In the above example, layers 0 through 3 will be plotted with PEN 1. Layers 4 through 6 of the drawing will be plotted with PEN 2. The operator may input spaces or a comma (,) between specific layers and specific range values.

EXAMPLE:

PEN 3	DBLAYERS	5 64
PEN 4	DBLAYERS	15, 17, 33-35
PEN 5	DBLAYERS	110-115 104
PEN 6	DBLAYERS	40:44 88

This control file will create the following:

PEN 3 assigned to layers 5 and 64
PEN 4 assigned to layers 15, 17, 33, 34, 35
PEN 5 assigned to layers 104, 110, 111, 112, 113, 114, 115
PEN 6 assigned to layers 40, 41, 42, 43, 44 and 88

If the operator uses the Telesis Keyboard Editor, (VT101 or Wyse) to create the file, type in the keyword PEN followed by a blank space, then the pen number. Input a blank space, then input the keyword DBLAYERS followed by the list of layers as shown in the above example.

STEP 4. ADD BULLETSIZE LINE

Use the |ADD BULLET SIZE LINE| command to specify plotting of software generated connect-points as circles on schematic drawings. For example, connect-points added to the schematic drawing during |ADD CONNECTION| are software generated. Connect-points within symbols on the schematic are non-software generated.

Filling of bullets occurs during the |PEN PLOT| command if the operator specifies the PENWIDTH keyword following the BULLETSIZE specification.

EXAMPLE:

ADD BULLETSIZE LINE	(SIZE OF BULLET)	-Keyboard Input-	ENTER
	(WIDTH OF PEN)	-Keyboard Input-	ENTER

- o The first keyboard input is the size of the bullet to be plotted. If the schematic drawing is english units, specify a value in mils or inches. For example, a keyboard input of 12 specifies a bulletsize of 12 mils. To specify inches, use a decimal; for example, .12 denotes .12 inches.

  If the schematic is drawn in metric units, the value specified is in millimeters. For example, a value of 5 specifies a 5 millimeter bulletsize.

- o The second keyboard input is the penwidth to be used if the operator wishes to fill the bullet during plotting. The rules stated above for english units and metric units schematics apply when specifying bulletsize and penwidth.

  The operator may skip the WIDTH OF PEN prompt and simply pick |ENTER| if filling of the bullet is not required. The keyword PENWIDTH will not appear on the line entry.

General Rules

1. Only software-generated connect-points will be plotted as bullets,i.e., circles . These connect-points are plotted with the pen assigned to the DBLAYER where the software-generated connect-points exist on layer 0. The pen number specified for DBLAYER 0 will be used.

2. All bullets in a drawing must be the same size. There should only be one BULLETSIZE line in a penplot control file. If more than one BULLETSIZE line exists, the system uses the latest, or last entry.

3. In the penplot control file, the penwidth must be less than or equal to the bulletsize, with one exception. It is valid to input a smaller bulletsize if the plot scale (greater than 1 at a time of PEN PLOT) multiplied by the bulletsize makes the bulletsize larger than the penwidth. If the penwidth is still greater than the BULLETSIZE, an error message will result and the program will terminate.

4. Bullets will only be plotted from english or metric schematic drawings. The bulletsize line will be ignored when plotting a board drawing. For metric drawings, the penwidth specification in the file must not be less than 0.1 millimeters; the bulletsize must not be greater than 50 millimeters. For english drawings, the bulletsize must no be greater than 2000 mils, or 2.0 inches.

EXAMPLE: english schematic drawing to be plotted:

ADD BULLETSIZE LINE	(SIZE OF BULLET)	12	ENTER
	(WIDTH OF PEN)	5	ENTER

RESULT:

```
PEN 5     DBLAYERS     110-115    104
PEN 6     DBLAYERS     40:44      88
PEN 7     DBLAYERS     89-91
PEN 8     DBLAYERS     146-147
BULLETSIZE    12     PENWIDTH     5
END
```

If using the Telesis Keyboard Editor (VT101 or Wyse) to create the penplot control file, input the keyword BULLETSIZE followed by a blank space, then the size of the bullet. Input a blank space, then input the keyword PENWIDTH if the bullet is to be filled during plotting. Omit the PENWIDTH keyword if filling is not required. Separate the keyword PENWIDTH from the pen size with a blank space as shown in the above example.

STEP 5. CONNECT-POINTS

Pick connect-points to include plotting of connect points. The keyword CONNECT-POINTS will appear as a line entry in the PENPLOT-CON file. Otherwise, the connect points displayed on the drawing will be omitted.

If plotting a schematic with a BULLETSIZE/PENWIDTH specification in the penplot control file, the CONNECT POINTS keyword will inform the system to plot the non-software generated connect-points on the drawing. For example, connect-points within symbols will be plotted; software-generated connect-points will be plotted as bullets. If the CONNECT-POINTS keyword is omitted, the system will not plot the non-software generated connect-points.

STEP 6. TEXT-POINTS

Pick text-points to include text points in the pen plot. The keyword TEXT-POINTS will appear as a line entry in the PENPLOT-CON file. Otherwise, the text points displayed on the drawing will be omitted from the plot.

NOTE: If including connect points and/or text points in the pen plot, the system will plot them at the sizes last set by the |SET CNCT PT SIZE| and |SET TXT PNT SIZE| commands. Size variations of connect points and text points on the active drawing will not be shown in the pen plot.

STEP 7. ADD COMMENT

Pick the |ADD COMMENT| command to add informational text to the file. The comment line entry will automatically be placed within parentheses (). This identifies the line entry as a comment. Comments are optional operator-defined line entries which allow additional information for easy interpretation of the file.

NOTE: The operator may input a maximum of 59 alphanumeric and blank spaces in a comment line entry. If the keyboard input exceeds the 59 character-limit, the system interrupts and inserts the first 59 characters into the file.

When using the Telesis Keyboard Editor (VT101 or Wyse), input parentheses ( ) around operator-defined comment entries in the file.

EXAMPLE: Adding the comment line (CLOCK GENERATION BOARD).

|ADD COMMENT| (ENTER TEXT) CLOCK GENERATION BOARD |ENTER|

CURRENT DISPLAY ON THE GRAPHICS SCREEN

```
                 PENPLOT-CON  1

      (PENPLOT-CON)
 ___  PEN 1  DBLAYERS 0-3, 7, 8, 10
      PEN 2  DBLAYERS 4:6
```

RESULT: The comment line (CLOCK GENERATOR BOARD) is inserted below the line (PENPLOT-CON).

```
                 PENPLOT-CON  1

      (PENPLOT-CON)
      (CLOCK GENERATOR BOARD)
 ___  PEN 1  DBLAYERS 0-3, 7, 8, 10
      PEN 2  DBLAYERS 4:6
```

STEP 8. DONE

When finished creating the penplot control file, pick |DONE|. To close the file, select one of the following commands from the menu displayed on the function screen:

|SAVE FILE| - allows the operator to save the active version of the file, and to delete the previous version of the same revision. If the file is newly created, the system saves the file under the revision label specified when the file was opened, or with revision "1" (default).

EXAMPLE:

Active file (new)	PENPLOT-CON	
Saved file:	PENPLOT-CON	1

|SAVE FILE NEW REV| - allows the operator to save the active file under a revision label other than the one originally specified when the file was opened. The system does not delete the old version. The revision label may contain up to four alphanumeric characters.

EXAMPLE:

Original version (old REV)	PENPLOT-CON	1
New version (new REV)	PENPLOT-COn	2

|CANCEL ACTV FILE| - allows the operator to cancel the actively displayed file on the graphics screen without deleting the original (if it exists). The original file maintains its original revision label.

|EDIT FILE| - allows the operator to "menu-back" and return to the active file for additional editing.

IF THE FILE IS LONGER THAN CAN BE SHOWN ON THE GRAPHICS SCREEN

If lines are added to the file after the graphics screen is full, the file will scroll up so that the bottom lines are visible and the top lines are not. Use the |LINE UP| command to bring the top lines back into view. Use the |LINE DN| command to bring up the bottom lines back into view.

5. EDITING THE PEN PLOT CONTROL FILE

When creating a penplot control file, the operator may make additions or corrections. The file may also be re-opened at a later time to update any version of the file previously created. When re-opening the file, the operator must specify the filename and the revision label if the file is not the latest version.

The operator may use the following commands to edit the penplot control file:

|CHANGE PEN LINE|

|CHANGE BULLETSIZE LINE|

|DELETE LINE|

|UNDELETE LINE|

|CHANGE COMMENT|

However, the add commands previously used in creating the PENPLOT-CON file may be used to make additions to the file. All commands used to create and edit the file are located on the same menu page.

POSITIONING THE CURSOR

To edit the file, the cursor must always be moved to the proper line position. The cursor remains in the left margin of the file, and can only be moved up and down.

For ADD commands during editing:

Position the cursor to the line above the desired location of addition. Lines positioned below the cursor will shift down when the new line is added. If the cursor is positioned at the END statement, the new line will be placed one line above.

For CHANGE or DELETE commands:

Move the cursor to the line to be changed or deleted.

Use the following commands to move the cursor:

|LINE UP| - moves the cursor up one line.

|LINE DN| - moves the cursor down one line.

EXAMPLE: LINE DN (Picked 3 times)

```
                 ---------------------------------------------
Current         |                                             |
Cursor   ----->|   ____ PEN 6    DBLAYERS  49:44   88         |
Position        |        PEN 7    DBLAYERS  89-91             |
                |        PEN 8    DBLAYERS  146-147           |
New      ----->|   ____ CONNECT POINTS                        |
Position        |        TEXT POINTS                          |
                 ---------------------------------------------
```

When LINE DN is picked, the cursor moves from the current position to the start of the next line of the file. Each pick moves the cursor down one line.

CHANGING THE INFORMATION IN THE FILE

Use the CHANGE commands to change or update the PENPLOT-CON file. For example, use the |CHANGE PEN LINE| command to redefine dblayers to be plotted with a particular pen. Move the cursor to the line to be edited.

EXAMPLE: CHANGE PEN LINE

```
Current       |                                                    |
Cursor  ----->|   ____ PEN 6    DBLAYERS  49:44   88               |
Position      |        PEN 7    DBLAYERS  89-91                    |
              |        PEN 8    DBLAYERS  146-147                  |
              |        CONNECT POINTS                              |
              |        TEXT POINTS                                 |
```

|CHANGE PEN LINE| (PEN 1-8) 6 |ENTER| (DBLAYERS) 40:44 88 103 ENTER

RESULT: Dblayer 103 is now assigned to pen 6 of the file.

```
Current       |                                                    |
Cursor  ----->|   ____ PEN 6    DBLAYERS  49:44   88  103          |
Position
```

NOTE: When using the |CHANGE PEN LINE| command, the operator may skip any prompt if the information for that entry in the file is unchanged. For example, if the dblayers require editing, the operator may simply pick |ENTER| after the (PEN 1-8) prompt, then proceed to change the dblayers for that pen. The current pen number for that pen line will remain unchanged.

Use the |CHANGE COMMENT| command to change or edit existing comment lines in the PENPLOT-CON file. Use the |CHANGE BULLET SIZE| command to update the current bullet size and penwidth specifications.

DELETING AND RESTORING INFORMATION IN THE FILE

Use the following commands to delete and restore information in the file:

|DELETE LINE| - deletes a single line specified by the current cursor position.

|UNDELETE LINE| - restores the last line entry deleted from the file. The system restores the deleted entry directly below the new cursor position, regardless of the line's original placement. This command may be used to reposition a line to another location with the file.

If the cursor is positioned at the END statement, the line will be restored one line above it.

NOTE: The line PENPLOT-CON at the top of the file, as well as the END statement cannot be deleted. The operator can only delete line entries that were input during the creation of the file.

DELETING A LINE FROM THE FILE

EXAMPLE: Deleting the CONNECT POINTS line

```
                          |        PEN 5  DBLAYERS    110-115    104  |
Current ---->             |    ____CONNECT POINTS                     |
Cursor                    |        PEN  6  DBLAYERS  40:44  88         |
Position                  |        PEN  7  DBLAYERS  89-91             |
                          |        PEN  8  DBLAYERS 146-147            |
                          |        TEXT-POINTS                         |
```

PICK: |DELETE LINE|

In the above example, DELETE LINE removes the CONNECT-POINTS line from the file. The contents of the file shift up, closing the open line resulting from the deletion.

```
New                       |        PEN 5  DBLAYERS    110-115    104  |
Cursor  ---->             |    ____PEN  6  DBLAYERS  40:44  88         |
Position                  |        PEN  7  DBLAYERS  89-91             |
                          |        PEN  8  DBLAYERS 146-147            |
                          |        TEXT-POINTS                         |
```

UNDELETING A LINE ON THE FILE

Use the |UNDELETE LINE| command restore the last deleted line entry back to the file at the line below the cursor position. Use the |LINE UP| and |LINE DN| commands to position the cursor to the line above the location where the line is to be restored.

EXAMPLE: Undeleting the CONNECT POINTS line

```
                          PEN  5  DBLAYERS   110-115   104
                          PEN  6  DBLAYERS  40:44  88
      New                 PEN  7  DBLAYERS  89-91
      Cursor  ---->    ___PEN  8  DBLAYERS 146-147
                          TEXT-POINTS
```

PICK: |UNDELETE LINE|

```
                          PEN  5  DBLAYERS   110-115   104
                          PEN  6  DBLAYERS  40:44  88
                          PEN  7  DBLAYERS  89-91
                          PEN  8  DBLAYERS 146-147
                       ___CONNECT POINTS
                          TEXT-POINTS
```

After reinserting the CONNECT-POINTS line with the |UNDELETE LINE| command, the cursor moves to that line of the file.

PENPLOT-CON FILES NOT CREATED WITH THE EDIT PENPLOT-CON LEADTHRU

Penplot control files that were not created with the |EDIT PENPLOT-CON| command may exist on the system. If a penplot control file was created with the Telesis Keyboard Editor, for example, the operator may edit the file using the EDIT PENPLOT-CON lead-thru.

When the operator opens one of these files with the |EDIT PENPLOT-CON| command, the system checks it for formatting errors. If errors exist, the system places a message on the function screen stating the number of faulty lines existing in the file; the system displays the file on the graphics screen with all faulty lines bracketed between parentheses and asterisks.

EXAMPLE: (* PNE 2 DBLYERS 4:6*)

The line is incorrect because the keywords PEN and DBLAYERS are misspelled.

The lines in parentheses are now comments, and the system treats them as such. These lines may be deleted with the |DELETE LINE| command. Use the |ADD| commands to input the correct information.

6. PLOT SCALE

All dimensions in the Telesis systems drawings have coordinates expressed in english units (mils) or in metric units (millimeters). For example, with an english drawing, a line one inch long is 1000 units long. Unless otherwise specified, the drawing will be plotted at a scale of one-to-one. That is, a line 1000 units long in a drawing will plot 1.000 inches long on the plot paper. However, the operator may plot a drawing much larger or smaller depending on the scale specified when executing the |PEN PLOT| command. Scale is a number greater than zero. A scale greater than one causes the system to plot the drawing larger than its stored coordinates, while a number of less than zero causes the plot to be smaller.

EXAMPLE: SCALE 2. - causes the plotted elements to be twice the size of the original drawing.

SCALE .5 - causes the plotted elements to be one-half the size of the original drawing.

7. USING THE PEN PLOT COMMAND

Prior to using the |PEN PLOT| command, select the view of the drawing to be plotted. For example, on Telesis 2.0 systems use the |WINDOW| command to select an area of the drawing, or the |DRAWING| command to plot the entire drawing. On Telesis 2.1 systems, use |NEW WINDOW| command to select the desired area, or the |WINDOW=WORLD| to plot the entire drawing during execution of the |PEN PLOT| command.

Make sure the penplot control file is available in the current project file, or in the SYSTEM-LIBRARY so the specific drawing layers of the active drawing will be plotted with specific pens in plotter carousel. If a penplot control file does not exist, all drawing layers will be plotted with PEN 1 of the plotter carousel.

The pen plotter must be properly connected and the paper and pens correctly loaded. The plotter must be in the remote state; that is, the REMOTE light on the front of the plotter must be "on."

The operator may now use the |PEN PLOT| command. The input sequence of the command is shown below:

PEN PLOT	(CONTROL FILENAME)	-Keyboard- Input	ENTER	(SCALE)	-Keyboard- Input	ENTER

- o The first keyboard input is the name of the penplot control file. If the file is named PENPLOT-CON, skip the keyboard input and pick |ENTER|. The system will look for the name PENPLOT-CON first in the current project, then in the SYSTEM-LIBRARY.

- o The second keyboard input is plot scale to be used. Skip the keyboard input and pick |ENTER| if plotting at a scale of "1."

  If the system cannot find the penplot control file specified by the first keyboard input, the system reports the following on the function screen message line:

  "NO PENPLOT CONTROL FILE"

  The system will then proceed with the pen plot, plotting all drawing layers with pen 1 of the plotter carousel.

- o If the system reports, "PEN PLOTTER NOT ON-LINE," check the plotter to ensure that it is properly connected and powered-on. If REMOTE PERIPHERAL OPERATION is installed for networking of peripheral devices and if the system reports the following, refer to UTILITIES section of this manual for additional information on networking and deallocating of peripheral devices.

  "PLOTTER ALREADY IN USE. COMMAND CANCELLED."

  There will be a slight delay before the plotter begins to operate. If only a small area of the drawing is to be plotted, there may be a delay while the system "windows" the drawing elements to be plotted. The "OUT OF LIMIT" light may appear on the plotter console. This is a normal condition if plotting a "windowed" portion of the drawing. This condition indicates that the plotter is clipping those drawing elements in the window that extend outside the window limits.

The |PEN PLOT| command plots all drawing elements on the layers assigned to pen 1 first, then those assigned to pen 2, and so on. As a result, there may be a delay in plotting if there are very few or no elements on the layers assigned to a pen. The plot will proceed faster as layers with more data are encountered, assigned to pens with higher numbers.

8. TROUBLE SHOOTING

WHAT TO CHECK FOR IF THE PLOTTER DOES NOT OPERATE:

- o Is the plotter properly connected and powered-on?
- o Is the plotter loaded with paper or mylar?
- o Is the plotter REMOTE button lit?
- o Is the drawing to be plotted actively displayed on the graphics screen?
- o Is the cable from the Telesis system to the plotter attached properly? If REMOTE PERIPHERAL OPERATION is installed for networking of peripherals, refer to the UTILITIES section of the manual for additional information.

THE WRONG PEN IS PLOTTING AN ELEMENT

Check the penplot control text file for pen assignments. If two different pen assignment lines are assigned the same layer, the latest pen number assigned overrides the first pen assignment.

THE PLOT IS LARGER THAN THE PLOTTING SURFACE

Check to be sure that the SCALE input is not too large for the plot paper. If the plot is larger than the plot paper, the PEN PLOT command will only plot that part of the drawing that will fit on the plot paper.

It will not plot the drawing information at the dimension edges of the drawing.

9. PLOTTER LIMITATIONS

If the operator attempts to plot a drawing on paper equal to the size of the drawing, elements near the edges may not be plotted. This is a limitation of the Hewlett-Packard HP7580 and HP7585 plotters.

For example:

A SIZE drawing plotted on A SIZE paper

B SIZE drawing plotted on B SIZE paper

C SIZE drawing plotted on C SIZE paper

D SIZE drawing plotted on D SIZE paper

It is recommended that the operator plot the drawing on paper one size larger than the drawing size. For example, plot an A SIZE drawing on B SIZE paper, a B SIZE drawing C SIZE paper, etc. This will ensure the plotting of all drawing information.

When plotting a D SIZE drawing, it is recommended that roll paper or military specification paper be used. Military specification (MIL-SPEC) paper is one inch larger in all drawing dimensions.

OPERATING THE PEN PLOTTER IN BACKGROUND

1. PENPLOT BACKGROUND
   - o General Information
2. THE ACTIVE DRAWING AND THE PENPLOT CONTROL FILE
3. CREATING THE -PLOT FILE WITH THE $\boxed{\text{CREATE PENPLOT FILE}}$ COMMAND
4. STARTING THE PEN PLOT IN "BACKGROUND"

1. PENPLOT BACKGROUND

GENERAL INFORMATION

The Telesis system allows the operator to plot the ACTIVE or INACTIVE drawing in a "background" mode. "Background" mode allows the operator to interact with the Telesis system while the plotter is operating.

There are two essential commands needed to operate the plotter in "background".

CREATE PENPLOT FILE
PENPLOT BACKGROUND

When the |CREATE PENPLOT FILE| command is used, the system reads the actively displayed drawing information and the penplot control data in the PENPLOT-CON file. The system then assembles and stores this information into a special -PLOT file.

When the -PLOT file has been created, the operator may use the |PENPLOT BACKGROUND| command to plot the ACTIVE or INACTIVE drawing. The system first reads the -PLOT file, then creates a temporary "background" copy. When the plotter begins to operate, and when the function screen "beeps", the operator may continue working interactively on the system.

2. THE ACTIVE DRAWING AND THE PENPLOT CONTROL FILE

When the active drawing is displayed on the graphics screen, the |CREATE PENPLOT FILE| command must be used to create the -PLOT file needed for "background "plotting. The -PLOT file contains the displayed drawing information, and the penplot control data listed in the PENPLOT-CON text file. The actively displayed drawing may be the entire drawing, or a "windowed-in" area.

NOTE: A PENPLOT-CON text file is optional. If a penplot control file does not exist in the current project file, or in the SYSTEM-LIBRARY, the -PLOT file is still created. However, all drawing layers are assigned to PEN 1 of the plotter carousel when the plot is started.

The text file created with the |CREATE PENPLOT FILE| command must not be edited with the function screen editor or the optional Telesis keyboard editor.

If this file is plotted on the IDS matrix printer, erroneous characters will result. The printer/plotter must be powered-off then powered-on to clear it.

3. CREATING THE -PLOT FILE WITH THE |CREATE PENPLOT FILE| COMMAND

Pick the |CREATE PENPLOT FILE| command. The operator must enter the name of the PENPLOT-CON text file, and the name of the -PLOT file to be created.

EXAMPLE:

|CREATE PENPLOT FILE| (ENTER PENPLOT CONTROL FILE NAME) -Keyboard- Input |ENTER|

(ENTER FILE NAME TO BE CREATED) -Keyboard- Input |ENTER|

- o Enter the name of the existing penplot control file. If the file is named PENPLOT-CON, simply pick |ENTER|. If a penplot control file does not exist, pick |ENTER| to assign all drawing layers to PEN 1 of the plotter carousel.

- o Enter the -PLOT file name. If a file name is not input, the system names the -PLOT file from the existing drawing name when |ENTER| is picked. The system also assigns revision "1" to the file.

  EXAMPLE:

  Existing drawing name: TEST

  -PLOT file created by system default: TEST-PLOT rev 1

However, the operator may input a file name that contains up to 13 alphanumeric characters. A 13-character file name results in 18 characters when the system inserts the -PLOT suffix. If the operator wishes to assign a revision label to the file, the file name and the revision must be separated by a blank space when it is input. A revision may contain up to four alphanumeric characters.

EXAMPLE: Inputting a file name and revision label.

File name Revision

TEST ABCD

Resulting file name:

TEST-PLOT ABCD

NOTE: If more than one -PLOT file is to exist in the same project file, the operator may name the first file with the drawing name default, then proceed to name the additional -PLOT files with revision labels.

When the system completes the creation of the -PLOT file, the function screen message line displays:

PLOT FILE COMPLETED

4. STARTING THE PEN PLOT IN "BACKGROUND"

After the -PLOT file is created with the |CREATE PENPLOT FILE| command, the plot can be started with the |PENPLOT BACKGROUND| command. Since the -PLOT file was created with the active drawing displayed, the drawing does not need to be active to start the plot. However, the operator must open the project file prior to picking the |PENPLOT BACKGROUND| command.

INPUT SEQUENCE:

|PENPLOT BACKGROUND| (NAME OF FILE TO BE PLOTTED) Keyboard-Input |ENTER|

(PLOT SCALE ENTER ONLY =1) -Keyboard-Input |ENTER|

- Enter the name of the -PLOT file. If the drawing is actively displayed, simply pick |ENTER| to locate the latest version of the -PLOT file.

  If the drawing is inactive, enter the name of the -PLOT file to be used. The -PLOT suffix does not need to be input.

  NOTE: If more than one -PLOT file exists in the current project file, ensure that the proper file name is selected. Otherwise, the plotter may plot an undesired portion of the drawing.

- Enter the plot scale. If |ENTER| is picked without a plot scale, the system defaults to SCALE 1.

  The system first checks that the plotter is on-line and not presently operating. If the plotter is not on-line, the function screen message line displays:

  PEN PLOTTER NOT ON-LINE

  If the plotter is presently operating, the message line displays:

  PEN PLOTTER BUSY, PLOT NOT STARTED

  When the plotter is on-line, and available, the system proceeds to create a "background" copy of the -PLOT file. As the system copies the file, the message line displays:

  COPYING FILE TO BACKGROUND

  When the plotter begins to operate, and when the system "beeps", the operator may interactively continue working on the Telesis system.

DIGITIZER

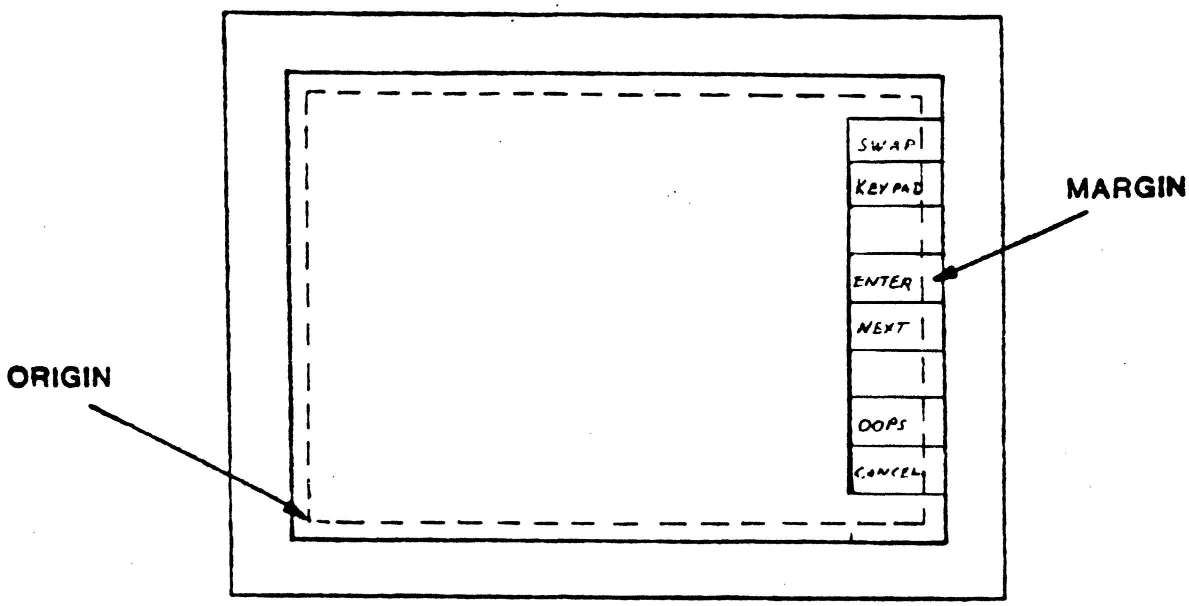
1. GENERAL INFORMATION

A CALCOMP 9000 series digitizer is optional equipment with the Telesis system. It can be used instead of the tablet and light pen to input a drawing.

Typically, you will use the digitizer to transfer an existing board drawing from hard-copy to the system. However, the digitizer may also be used for schematics or other drawings.

The digitizer consists of a table and a puck. When the digitizer is turned on, the table is used as the equivalent of the function screen tablet, and the puck is used as the equivalent of the light pen.

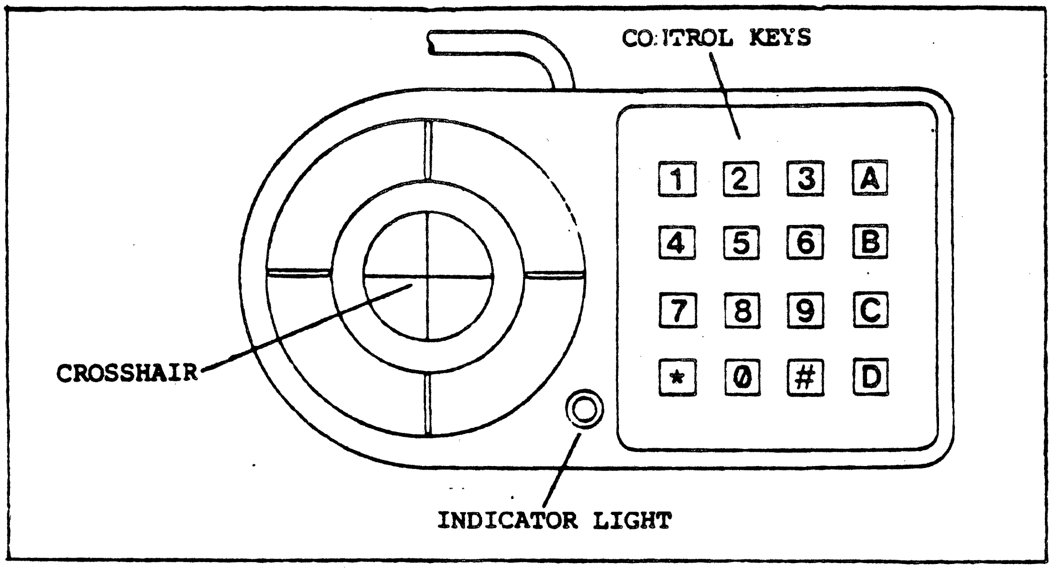
DIGITIZER TABLE



The digitizer table has a 48" by 36" active surface. It contains a network of wires in the horizontal and vertical directions, which are used when locating coordinate points relative to the origin (point 0,0) at the lower left corner of the active surface. There is an inactive margin around the perimeter of the surface. It is shown in the illustration above, but it is not marked on the table itself. (The margin is approximately 1-1/2 inches wide.)

At the right of the table are most of the same commands that appear in the top row of the function screen tablet.

DIGITIZER PUCK



The digitizer puck is used to input coordinate point locations, and to input the commands at the right of the digitizer table. The puck has these components:

INDICATOR LIGHT	The red indicator light is lit when the digitizer is ready to accept input. Any input attempted when this light is off will be lost.
CROSSHAIR	The crosshair is used to select the point to be digitized.
CONTROL KEYS: 1,2,3,4,5,6,7, 8,9,0 and #	Any one of these keys is pressed to digitize the point location shown under the crosshair or the command shown under the crosshair.
A,B,C,D	Any one of these keys is pressed to implement the |NEXT| command. The puck may be anywhere on the active surface. (For the sake of convenience the |NEXT| command can be implemented in two ways: by pressing the A,B,C or D control keys; or by positioning the crosshair over the |NEXT| command and pressing any control key except *.)
*	This key is not used.

SUMMARY OF STEPS TO FOLLOW IN USING THE DIGITIZER

STEP 1 Tape a hard-copy drawing to the digitizer table.

STEP 2 Open a drawing file.

STEP 3 Use the |DIGITIZER MENU| command to flip to the menu page containing these commands: |DIGITIZER ON|,|DIGITIZER OFF|, and |CALIBRATE DIG|.

STEP 4 Use the |CALIBRATE DIG| command to calibrate the digitizer. This step is optional; you are not required to calibrate the digitizer. (See p. PRINTPLOT-20 for calibration instructions.)

STEP 5 Use the |DIGITIZER ON| command to turn on the digitizer as an input device.

STEP 6 Digitize the drawing. (See p. PRINTPLOT-24 for instructions.)

STEP 7 Use the |DIGITIZER OFF| command to return to the tablet method of input.

NOTE: You may turn the digitizer on and off as often as you like to switch back and forth from digitizer input to tablet input.

LIMITATIONS OF THE DIGITIZER

- The |NECK| and |DRILL| commands cannot be used with the digitizer.
- Typically, you will use the digitizer to input a completed board drawing, and you will not create a net data base. (A net data base can only be created from a schematic drawing or a text-input netlist.) When this is the case, you will not be able to use commands and capabilities that require a net data base such as: |ASSIGN REF DES|, |RATSNEST|, |NET COMPARE|.

  You will be able to use all other system capabilities, such as: physical design rules checking, the artwork capability, and the NC drill capability.

2. HOW TO CALIBRATE THE DIGITIZER

PURPOSE

Digitizer calibration is optional. By calibrating the digitizer, you inform the system where and at what scale to place the data you digitize. Calibration has several benefits:

- It compensates for shrinkage and distortion in the hard-copy drawing.
- It compensates for any misalignment in the placement of the hard-copy drawing on the digitizer table.
- It allows you to scale the system version of your drawing to be larger or smaller than the hard copy version.
- It allows you to specify where your drawing will be placed relative to the system drawing extents.

DEFAULT CALIBRATION

If you do not calibrate the digitizer, the system will use the following default calibration:

ALIGNMENT	The system version of the drawing will be aligned to the system drawing extents exactly as the hard-copy drawing is aligned to the digitizer table. (If the hard-copy is crooked, the system version may be crooked as well.)
SCALE	The system version of the drawing will be created at a 1:1 scale with the hard copy drawing.
REFERENCE POINT	The system takes the lower left corner of the digitizer's active surface as the reference point for the hard copy drawing; and it takes the lower left corner of the system drawing extents as the reference point for the system version of the drawing. Any point you digitize will have the same relation to the system reference point that it has to the hard-copy reference point.

WHEN TO CALIBRATE

Typically, you use the |CALIBRATE DIG| command when you are preparing to digitize a new drawing. (If you do not, the system will use your previous calibration or, if there is no previous calibration, the default calibration.) If the hard-copy drawing is moved (or slips) on the digitizer surface, you should repeat the |CALIBRATE DIG| command and calibrate exactly as you did previously.

Drawings may be opened and closed, and the digitizer may be turned on and off without affecting the calibration.

Calibration is cancelled when the system is turned off or rebooted. To regain your previous calibration after turning the system back on or rebooting, you must use the |CALIBRATE DIG| command again and recalibrate exactly as you calibrated previously.

PREREQUISITES

- A drawing file must be active.
- The full extents of the current drawing must be displayed on the graphics screen. (New drawings wil look blank whether they are fully displayed, zoomed in, or shifted, so it is a good idea to be sure the drawing is fully displayed by using the |DRAWING| command before calibrating.)

STEPS TO FOLLOW IN THE CALIBRATION PROCESS

STEP 1 Carefully mark 3 points on the hard copy drawing:

Point 1 is the lower right corner
Point 2 is the lower left corner
Point 3 is the upper left corner

STEP 2 Pick the |CALIBRATE DIG| command.
(Do not pick the |DIGITIZER ON| command.)

STEP 3 Respond to each of the prompts displayed on the function screen message line. The table below gives an example of each prompt with instructions on how to respond:

EXAMPLE OF PROMPT	HOW TO RESPOND
ENTER IN X SIZE OF DRAWING (IN MILS)	Input the X size at which you wish the system to create your drawing (not necessarily the actual size of the hard-copy drawing). The size you specify here must fit within the drawing extents of the currently active drawing. For example, if you picked |B SIZE| when you opened the current drawing, your X size here must be $\leq$ 17000 mils.

(table continued on next page)

(Table continued from previous page)

EXAMPLE OF PROMPT	HOW TO RESPOND
ENTER IN Y SIZE OF DRAWING (IN MILS)	Input the Y size in the same way that you input the X size above.
PLEASE ENTER LOWER RIGHT HAND CORNER	Use the digitizer puck to digitize the lower right corner point that you marked on your drawing in STEP 1 above.
PLEASE ENTER LOWER LEFT HAND CORNER)	Digitize the lower left corner in the same way that you digitized the lower right hand corner.
PLEASE ENTER UPPER LEFT HAND CORNER	Digitize the upper left corner in the same way that you digitized the lower corners.
SCALE X:2:1(2.003) Y:2:1(2:007) HIT ENTER IF CORRECT	The system reports the scale of the hard-copy drawing to the system version of the drawing as you have specified it in the 5 previous responses. (The message shows the nominal scale with the exact scale enclosed in parentheses. The system will use the exact scale. Use the digitizer puck to pick |ENTER| if the scale is acceptable. If not, pick |CANCEL| and begin the calibration process again.
ENTER X OFFSET (IN MILS)	Input a value if you wish to offset (in the X axis) the drawing area you defined in this calibration from the drawing extents of the currently active drawing. For no X offset, just pick |ENTER|. (If you enter no X or Y offset, the system places the lower left corner you digitized at the lower left corner of the system drawing extents.)

(Table continued on next page)

(Table continued from previous page)

	NOTE: Do not input an offset so large that it shifts a large part of the drawing area you defined in this calibration outside the system drawing extents. You will not be able to digitize outside the system drawing extents.
ENTER Y OFFSET (IN MILS)	Input an offset in the Y direction in the same way that you input an X offset above.

3. HOW TO DIGITIZE A DRAWING

PREREQUISITE

- A drawing file must be active.

TURNING THE DIGITIZER ON AND OFF

Use these commands to turn the digitizer on and off.

|DIGITIZER ON| Use the |DIGITIZER ON| command to activate the digitizer.

Once the digitizer is on, you may use it whenever the tablet maenu appears. The word "DIGITIZER" will be displayed on the tablet menu indicating that the digitizer can be used and the tablet cannot.

The digitizer remains on until you turn it off with the |DIGITIZER OFF| command, or until you close the active drawing.

|DIGITIZER OFF| Use the |DIGITIZER OFF| command when you wish to switch from the digitizer method of input to the tablet method.

You may turn the digitizer off and on as often as you like, to switch back and forth from tablet input to digitizer input.

DIGITIZER INPUT

Use the digitizer just as you would use the tablet and light pen to input coordinate point locations. The table below shows your action and the system´s response.

ACTION	RESPONSE
MENU COMMANDS: Use the light pen and function screen to pick the command you want. For example: |ADD CONNECTION| (You may use any commands with the digitizer that you can use with the tablet except the |NECK| and |DRILL| commands.)	Whenever a command requires coordinate input, the system will display the tablet with the word "DIGITIZER" on it. As you work, the command echo, input echo, prompts and error messages will appear as usual on the function screen.

(Table continued on next page)

(Table continued from previous page)

ACTION	RESPONSE
COORDINATE INPUT: Aim the crosshair on the digitizer puck at the desired location on the hard-copy drawing, and press a control kdy (0-9,#) to digitize the point.	The graphics screen will display a small stationary crosshair at each point you digitize. The graphics screen will not display a dynamic cursor or the large cursor-crosshair.
DIGITIZER TABLE COMMANDS: Use the puck to pick the |SWAP|, |NEXT| |ENTER|, |KEYPAD|, |OOPS|, and |CANCEL|. commands on the digitizer table. Aim for the center of each box with the puck crosshair. (For convenience, you may also implement the |NEXT| command by pressing the A,B,C, or D control keys. In this case, the puck may be anywhere on the active surface.)	These commands operate just as they do on the tablet.

RECOMMENDED STEPS IN DIGITIZING A DRAWING

STEP 1	Add symbols
STEP 2	Add vias
STEP 3	Add lines (one layer at a time)
STEP 4	Add text labels. We recommend that to input text, you turn off the digitizer and use the the tablet and light pen. You will find this more convenient than using the digitizer since your attention will have to be on the function screen for keyboard input and for messages.

U T I L I T I E S

- o DISK SPACE
- o THE PERIPHERAL SWITCH
- o MENU FILES: USER DEFINABLE SYMBOL MENUS
- o CREATING AND EDITING EXECUTE FILES
- o REMOTE PERIPHERAL OPERATION
- o CONVERTING LINE GRAPHICS

DISK SPACE

1. QUANTITY OF DISK SPACE AVAILABLE ON THE SYSTEM

2. DRAWING COMPRESSION

1. QUANTITY OF DISK SPACE AVAILABLE ON THE SYSTEM

Drawing files, net data base files and all other files on the system use up disk space. So do many system operations. The Telesis system has approximately 45,000 blocks of disk space available for your use.

To put this in perspective, the list below gives you an idea of how much disk space is commonly used by different types of files (bear in mind that disk space used depends on the size and complexity of the file):

Type of File	Approximate Disk Space Used
Symbol file	25 - 150 blocks
Schematic drawing	300 - 1200 blocks
Board drawing	1000 - 4000 blocks
Text file	Very few blocks
Net data base file	100 - 200 blocks

Listed below are some of the system operations that require considerable free disk space:

- Using the matrix printer
- Using the |EXTRACT NETLIST| command
- Using the |NET COMPARE| command
- Physical design rules checking
- Creating artwork files
- Using the |OLD DRAWING| command
- Using the |CREATE NC DRILL| command

In general, 3000 blocks of free disk space will usually be enough for operations such as these.

USING THE FREE DISK SPACE COMMAND

When you pick the |FREE DISK SPACE| command, the system reports on the function screen the number of blocks of disk space currently available.

Use this command to check whether you have enough free disk space whenever you are about to begin a new drawing or a space-consuming operation.

TRY TO KEEP AS MUCH DISK SPACE AVAILABLE AS POSSIBLE

To avoid running out of available disk space, copy files from the system to floppy disks or magnetic tape whenever you do not need the files for your current work. You may copy them back to the system when you do need them.

You may also create more free disk space by using the |COMPRESS DRAWING| command. See the next section for instructions.

2. DRAWING COMPRESSION

PURPOSE OF THE |COMPRESS DRAWING| COMMAND

The |COMPRESS DRAWING| command compresses the drawing information on the system so that it takes up less disk space. This compression also causes the system to operate faster.

WHEN TO USE THE |COMPRESS DRAWING| COMMAND

Use the |COMPRESS DRAWING| command after you have done substantial editing or deleting on a drawing; or after you have used the |RATSNEST| command repeatedly. |COMPRESS DRAWING| frees disk space that was being occupied by deleted elements and deleted ratsnest lines.

This command does not alter the drawing in any way; it affects disk space only. It will have little or no effect if there has not been repeated ratsnesting or substantial editing on a drawing.

PREREQUISITES

1. You must be in the design project where the drawing exists.
2. The drawing must be INACTIVE.
3. When doing a |COMPRESS DRAWING| on a schematic, be certain that the latest net data base exists in the current project.

COMPRESSING THE DRAWING

When you pick the |COMPRESS DRAWING| command, the system prompts you for a drawing name. When you have input the name of the drawing to be compressed, and the REV (if the drawing is not the latest version), pick |ENTER| to begin the compression.

The message line at the top of the command menu displays:

"COMPRESS DRAWING STARTED"

When drawing compression begins, the drawing becomes ACTIVE and is displayed on the graphics screen. When drawing compression is completed, the drawing becomes INACTIVE. However, the drawing remains displayed on the graphics screen.

The message line at the top of the command menu displays:

"COMPRESS DRAWING COMPLETED"

The system automatically executes |SAME DRW SAME RV|. The compressed drawing is saved under the same name and revision label specified when compression was started.

THE PERIPHERAL SWITCH

THE PERIPHERAL SWITCH

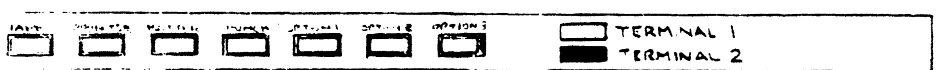
PURPOSE

If you have two Telesis terminals, you will be provided with a peripheral switch that allows you to use peripheral equipment (such as a magnetic tape drive) with either terminal.

LOCATION

The peripheral switch is located in the center of the magnetic tape drive cabinet.

It looks like this:



USE THE PERIPHERAL SWITCH BEFORE USING THE PERIPHERAL COMMANDS

Before using the function screen commands that operate a peripheral, you must be sure that the peripheral is switched to your terminal. For example, before picking [PEN PLOT], press the PLOTTER switch if it is not already set for use with your terminal.

HOW TO USE THE PERIPHERAL SWITCH

The four switches on the left are labelled with the names of the equipment they switch. The switches labelled OPTION 1, OPTION 2, and OPTION 3 can be connected to any additional Telesis peripherals (such as a VT100) that you have purchased and wish to share between the two terminals.

Press each switch to change it from the depressed position (white) for terminal 1, to the extended position (black) for terminal 2. (Which is terminal 1 and which is terminal 2 will be determined at the time that the peripheral switch is installed with your system.)

IMPORTANT! Check with the operator of the other Telesis terminal before using the peripheral switch.

IMPORTANT! Do not switch a peripheral that is currently in operation.

(This page intentionally left blank)

MENU FILES: USER-DEFINABLE SYMBOL MENUS

1. GENERAL INFORMATOIN

2. CREATING A MENU FILE

3. USING MENU FILES

1. GENERAL INFORMATION

The Telesis menu file capability allows you to create and use customized symbol menus.

A menu file is a file in which you may create one set of schematic symbol menus, one set of board symbol menus, or both.

Once you have created these menu sets, you may use them to add your own symbols to a drawing in the same way that you use the Telesis symbol menus to add Telesis-prepared symbols to a drawing.

You may create as many menu files as you wish, but you can only use one menu file at a time when adding symbols to a drawing.

2. CREATING A MENU FILE

STEPS TO FOLLOW IN CREATING A MENU FILE

STEP 1 Pick |USER MENUS| to flip to the menu page used for opening, editing and saving menu files.

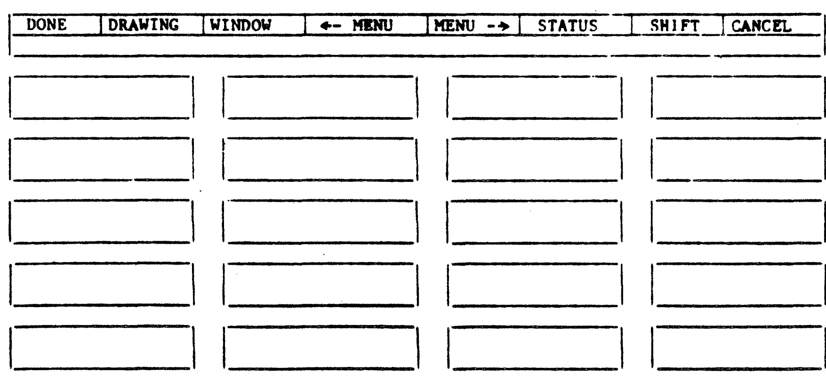
STEP 2 Pick |NEW USER MENUS| and input the name you wish to give the new menu file. (Refer to Command Description Section of the manual for required input sequence.)

STEP 3 Pick |EDIT SCHEM MENU| or |EDIT BOARD MENU| (these two commands work in exactly the same way).

The system will display the first page of a set of ten blank menu pages. (There is one set of 10 menu pages for schematic symbols and another set of 10 for component symbols.)

The menu boxes are blank except for the top row which contains the standard top-row commands appropriate to symbol menus.

EXAMPLE:



STEP 4 Pick any of the menu boxes, except those in the top row, to add or edit text. (You may not edit the commands in the top row.).

STEP 5 Input the symbol file name that you want to appear in the box.

EXAMPLE: |D|I|O|D|E| |ENTER|

The symbol file name may be up to 16 alphanumeric characters with no blank spaces allowed. (You may input blank spaces before the symbol file name in order to center the text in the menu box.)

No revision labels are allowed in the file name.

The name must be exactly the same as the name of the symbol file that you want to associate with this box.

When you pick |ENTER| the symbol file name will appear in the box you picked.

If you wish to change a symbol name after you have entered it, pick the menu box again and input a new name. The new name will replace the old one when you pick |ENTER|.

To delete a symbol name from a menu box, pick the box and input at least two blank spaces. When you pick |ENTER|, the box will become blank.

STEP 6 Repeat STEPS 4 and 5 until you have placed as many symbol names as you wish in the menu set.

Use the |<-MENU| and |MENU->| commands as usual to flip from one menu page to another within the 5 menu set.

STEP 7 Pick |DONE|. If you picked the |EDIT SCHEM MENU| command first, you may now pick |EDIT BOARD MENU| or vice versa.

After picking |DONE|, you may repeat the |EDIT SCHEM MENU| or |EDIT BOARD MENU| commands as often as you like, to make changes or additions to your menus.

STEP 8 Use the |SAVE USER MENUS| command when you have finished creating the menu set (or sets) within the menu file.

NOTE: The menu file must be open when you add symbols to a drawing. However, it is important to save a back-up copy of the menu file before using it to add symbols to a drawing. Therefore, be sure to use the |SAVE USER MENUS| command or the |SAVE NEW RV| command whenever you finish creating or editing a menu file. You may then reopen the file with the |OLD USER MENUS| command.

3. USING MENU FILES

ADDING SYMBOLS TO A DRAWING

STEP 1 Use the |OLD USER MENUS| command to open the menu file you want to use.

STEP 2 Open your drawing file.

STEP 3 Use the |ADD SYMBOL[USER]| command just as you would use the |ADD SYMBOL[MENU]| command: pick |ADD SYMBOl[USER]| to access your symbol menu set, then pick the menu box containing the name of the symbol you want to add to a drawing.

NOTE: Whenver you attempt to add a symbol, the symbol file must be in the current project file or the SYSTEM-LIBRARY.

If your symbol menus duplicate any of the names on the Telesis symbol menus, make sure that only the symbol files you want to use are in the current project file and/or SYSTEM-LIBRARY. (Symbol files in the current project file take precedence over symbol files with the same name in the SYSTEM-LIBRARY).

STEP 4 When you have finished using the menu file, you may use the |CANCL USER MENUS| command as long as you have a back-up copy of the menu file.

ARCHIVING MENU FILES

Menu files can be copied, deleted, and archived on tape or floppy disk in essentially the same way as other files. You will find the appropriate commands (such as, |COPY MENU FILE|, |DELETE MENU FILE|, |MENU TO TAPE|) on the file management and archive menus. These commands are all described in the Command Description Section of the manual.

Menu files can only be created and used at the system level. That is: they cannot be created (or edited) within a project file, and they cannot be used to add symbols to a drawing if they are filed in a project file.

However, it is possible to copy a menu file into a project file if you wish to do so for archival purposes. (For instance, if you created a menu file to be used only with one board project, you might want to archive that menu file within that project file.) If you do this, you must remember that you will have to copy the menu file back to the system level when you wish to use it again.

CREATING AND EDITING EXECUTE FILES

1. GENERAL INFORMATION
2. RECORDING THE EXECUTE FILE WITH THE TELESIS FUNCTION SCREEN.
3. CREATING AND RUNNING THE EXECUTE FILE AT VARIOUS SYSTEM LEVELS.
4. USING THE EXECUTE MENU COMMAND TO CREATE THE EXECUTE FILE.
5. EDITING THE EXECUTE FILE
6. PRINTING THE EXECUTE FILE
7. RUNNING THE EXECUTE FILE
8. IMPORTANT CONSIDERATIONS
9. EXECUTE FILE ERROR MESSAGES

CREATING AND EDITING EXECUTE FILES

1. GENERAL INFORMATION

The Telesis system provides the capability of recording a series of menu picks into a file. The file may then be used to operate the system overnight in a non-interactive mode. This record of interactive commands for overnight processing is called an "execute" file. Once the file has been created and started with the |RUN EXEC FILE| command, the system reads the file and "executes" each command non-interactively.

The execute file is simply a method of storing a sequence of commands to be processed overnight. When the execute file is started, the system operates non-interactively; the operator does not need to be present during processing. Productivity on the Telesis system can be greatly increased during those periods when the system is not used interactively by an operator.

An execute file may be created for any operation that can be performed interactively on the Telesis system. Because the execute file can be created at various menu locations, the system can be operated non-interactively for one, or many system capabilities.

Execute files may be printed, copied, deleted, or stored on floppies or magnetic tape. |CURRENT INDEX|, |SYSTEM INDEX| or |PROJECT INDEX| may be used to display the names of execute files created on the system.

HOW THE SYSTEM PROCESSES EACH EXECUTE FILE

Once an execute file is created and stored, the operator may run the file with the |RUN EXEC FILE| command. When the file name is entered, the system reads the execute file and begins to process each command in the sequence recorded on the file. As the system operates non-interactively, it echoes and displays each command being processed on the function screen message line. After a command is executed, the command name is re-displayed on the function screen in reverse video. This process continues until the system completes the last command recorded in the execute file.

The operator may then use the system interactively when the execute file completes processing.

NOTE: During the processing of an execute file, the operator may pick |CANCEL|. The command currently being processed (displayed on the function screen message line) continues until completed. Then, the execute file processing is terminated.

2. RECORDING THE EXECUTE FILE WITH THE TELESIS FUNCTION SCREEN

When creating an execute file, you must interact with the Telesis function screen to record the sequence of commands to be executed. When a command is picked and recorded in the execute file, the system does not process the command. The system simply records the command in the execute file. However, if you are creating an execute file to operate the auto-router, for example, you may set the auto-router parameters during the creation of the execute file. The system will process the parameter information when the execute file is started with the |RUN EXEC FILE| command.

NOTE: If an active drawing is displayed, the various display commands on the Telesis system will operate interactively during the creation of the execute file. For example, DISPLAY GRID and SET GRID SIZE will cause the system to paint the grid specification on the graphics screen during the recording of the execute file.

USING THE TELESIS KEYPAD AND TABLET MENU TO RECORD INPUTS

The Telesis keypad and the tablet menu may be used to record inputs into the execute file. For example, if the ADD SYMBOL [MENU] command is used during the creation of the file, the keypad may be used to input the explicit drawing coordinates. If an active drawing is displayed, the tablet menu may be used to pick the drawing location where the symbol is to be added during processing of the execute file.

3. CREATING AND RUNNING THE EXECUTE FILE AT VARIOUS SYSTEM LEVELS

The execute file may be created and processed at various levels on the Telesis system.

1. OLD PROJECT/NEW PROJECT level
2. Within a project (Inactive drawing)
3. Within a project (Active drawing displayed)

An execute file can be created at the NEW PROJECT/OLD PROJECT page (boot-up menu), or created within a project that performs a particular operation. (EX. ROUTER = DESIGN BOARD).

For example, the operator may create one execute file that opens an existing project (OLD PROJECT) and existing board drawing (OLD DRAWING) non-interactively. The same execute file then proceeds to start the auto-router (ROUTE BOARD) and save the drawing (SAVE DRW SAME REV). However, the operator may open a project and a drawing interactively, then proceed to create the execute file that performs an operation non-interactively.

The operator MUST ALWAYS run the execute file from the same level used to create the file. The system will not process an execute file if it is started from an incorrect menu location. (See RUNNING THE EXECUTE FILE).

4. USING THE |EXECUTE MENU| COMMAND TO CREATE THE EXECUTE FILE

The |EXECUTE MENU| command is located at the first page of various levels of the Telesis system. The Telesis system recognizes each level used during the creation, editing and running of execute files.

The |EXECUTE MENU| command is located at the following system levels:

1. OLD PROJECT/NEW PROJECT
2. BOARD DRAWINGS and DESIGN BOARD
3. OTHER DRAWINGS and DRAW SCHEMATIC
4. GROUPS MENU
5. TEXT MENU

When |EXECUTE MENU| is picked, the system displays the following menu page:

DONE								
EXECUTE MENU								

	CREATE EXEC FILE	EDIT EXEC FILE	PRINT EXEC FILE
	RUN EXEC FILE		
CURRENT INDEX		PROJECT INDEX	SYSTEM INDEX

Pick the |CREATE EXEC FILE| command to begin the creation of the execute file. When this command is picked, the system issues a prompt for the execute file name. After the file name is entered, the system then re-displays the previous menu page.

EXAMPLE: |CREATE EXEC FILE| (EXECUTE FILE NAME) -Keyboard- Input |ENTER|

- The keyboard input is the execute file name. The file name may contain up to 18 alphanumeric characters with no blank spaces. The system does not assign a default name if the file name is omitted.

- A revision label is optional. It may contain up to four alphanumeric characters. The file name and the revision label must be separated by a blank space. If a revision is not input, the system assigns the revision label "1" when the file is saved.

EXAMPLE: Typical execute file name

File Name → OVERNIGHT; Revision → TEST

OVERNIGHT TEST

When the |CREATE EXEC FILE| command sequence is terminated, the system re-displays the previous menu page containing the |EXECUTE MENU| command.

The system is now in the execute file "record" mode. Each command picked from the function screen is now recorded into the execute file in the sequence selected by the operator. Each menu pick must be selected in the sequence typically used to "interactively" operate the system.

The following example demonstrates a typical execute file created on the Telesis system.

EXAMPLE: CREATING THE EXECUTE FILE AT THE OLD PROJECT/NEW PROJECT MENU

STEP 1 Pick |EXECUTE MENU| to display the execute file menu page.

STEP 2 Pick |CREATE EXEC FILE|, input the execute file name, pick |ENTER|. The system now re-displays the OLD PROJECT/NEW PROJECT menu.

STEP 3 The system is now in the execute file "record" mode. Menu picks are recorded into the file in the sequence selected by the operator.

MENU BOX PICKED	MENU PICK SEQUENCE
OLD PROJECT	1
OLD DRAWING	2
DRAW SCHEMATIC	3
EXTRACT NETLIST	4
LOGIC RULE CHK	5
CREATE BOM FILE	6
CREATE NETLIST	7
CREATE CMPNT-RPT	8
PR BOM-REPORT	9
PR NETLIST-RPT	10
PR EXTRACTION-LOG	11
PR COMPONENT-RPT	12
SAVE DRW SAME RV	13

STEP 4 When the execute file has been created, pick |DONE| to return to the |EXECUTE MENU| command.

STEP 5 Pick |EXECUTE MENU| to display the menu boxes selected. The function screen display appears in the "edit mode" format:

DONE	LIST	DELETE	<- PAGE	PAGE ->	CHANGE	INSERT	

CREATE EXECUTE FILE

(1) OLD PROJECT	(2) OLD DRAWING	(3) DRAW SCHEMATIC	(4) EXTRACT NETLIST
(5) LOGIC RULE CHK	(6) CREATE BOM FILE	(7) CREATE NETLIST	(8) CREATE CMPNT RPT
(9) PR BOM-REPORT	(10) PR NETLIST-RPT	(11) PR EXTRACTION-LOG	(12) PR COMPONENT-RPT
(13) SAVE DRW SAME RV			

Each menu box displayed on the function screen is labeled with a number. This number identifies the sequence, or position, in the execute file where the command was entered. The operator may now edit the file, or pick |DONE|. (See EDITING THE EXECUTE FILE.)

STEP 6 Pick |DONE| to display the following command options:

|SAV EXEC SAM RV| - saves the file under the same revision label that was named when the file was opened. If a new file was created without a revision label, the system assigns revision "1" to the file.

EXAMPLE: Active file name - EXECUTE

|SAV EXEC SAM RV| - EXECUTE 1

|SAV EXEC NEW RV| - allows the operator to save the file with a new revision label. The revision label may contain up to four alphanumeric characters.

If the active file was previously saved, the system does not delete or change the original version and its revision label.

EXAMPLE: Original version - EXECUTE 1

|SAV EXEC NEW REV| (REV) ABC |ENTER|

Result - EXECUTE ABC

|CANCEL ACTV EXEC| - causes the system to delete the active version of the file.

If the active file was previously saved, the system does not delete the original version and its assigned revision label.

|EDIT FILE| - allows the operator to "menu-back" and return to the active file for further editing.

5. EDITING THE EXECUTE FILE

Existing execute files on the Telesis system may be edited to insert additional menu picks. Menu picks may also be deleted or changed to update the file.

The file may be edited when it is created with the |CREATE EXEC FILE| command, while existing files may be edited with the |EDIT EXEC FILE| command. When the execute file is displayed on the function screen, the softkey editing commands can be used to edit the file.

The system does not allow invalid editing of the execute file. Deletions and insertions at improper EXECUTE FILE locations will not be recorded. The file must be properly sequenced to operate the system non-interactively. (See EXECUTE FILE ERROR MESSAGES.)

USING THE |EDIT EXEC FILE| COMMAND TO OPEN AND EDIT AN EXISTING FILE

The |EDIT EXEC FILE| command must be used to open and edit existing execute files.

EXAMPLE: |EDIT EXEC FILE| (EXECUTE FILE NAME) - Keyboard Input - |ENTER|

- The keyboard input is the execute file name. Input the file name and the revision label to open the version to be edited. Omit the revision label if you wish to open the latest version.

When the above |EDIT EXEC FILE| command sequence is terminated with |ENTER|, the system displays the file on the function screen.

The following editing commands are available on the softkey line of the execute file display:

|DELETE|
|CHANGE|
|INSERT|
|LIST|

When a softkey editing command is picked, it highlights in reverse video. This signals that the command is active and is awaiting a menu pick selection from the execute file display.

|DELETE|

|DELETE| may be used to eliminate up to 18 menu boxes from the execute file. When |DELETE| is picked, it highlights in reverse-video. The first menu box to delete may be selected; the system then repaints the execute file display. The first selection to delete appears at the second menu box with its text displayed in reverse video. All menu boxes surrounding this selection are displayed.

EXAMPLE: Pick |DELETE| (Highlights in reverse-video)
Pick |PR NETLIST-RPT|

DONE	LIST	DELETE	<- PAGE	PAGE ->	CHANGE	INSERT	
EDIT EXECUTE FILE							

(1) OLD PROJECT	(2) OLD DRAWING	(3) DRAW SCHEMATIC	(4) EXTRACT NETLIST
(5) LOGIC RULE CHK	(6) CREATE BOM FILE	(7) CREATE NETLIST	(8) CREATE CMPNT RPT
(9) PR BOM-REPORT	(10) PR NETLIST-RPT	(11) PR EXTRACTION-LOG	(12) PR COMPONENT-RPT
(13) SAVE DRW SAME RV			

RESULT

Menu boxes highlighted in reverse-video

		DELETE			OOPS!	ENTER	

(9) PR BOM-REPORT	(10) PR NETLIST-RPT	(11) PR EXTRACTION-LOG	(12) PR COMPONENT-RPT
(13) SAVE DRW SAME RV			

When the execute file display is repainted, the |DELETE| command remains highlighted. The remaining menu boxes (up to 18) on this display may be highlighted for deletion if the operator chooses to do so. When all deletions have been selected, the operator may pick |OOPS!| or |ENTER|.

NOTE: The system does not allow random deletions from the execute file display. Random selections will not highlight in reverse-video.

EXAMPLE: |DELETE| (P)(P)(P) |ENTER|

|OOPS!| - Pick |OOPS!| to remove the last highlighted box from the group to be deleted. Each |OOPS!| pick will de-highlight a menu box from the "delete" group.

|ENTER| - Pick |ENTER| to delete the valid menu box selections in the execute file. The function screen message will display the following message:

DELETING...DELETING...DELETING...DELETION COMPLETED

When all valid deletions have occurred, the system repaints that page of the execute file display. All valid deletions do not re-appear in the file. The system simply resequences the remaining menu boxes in the file where the deletions occurred.

If the file is longer than one page, pick |PAGE -->| to display the next page of the file and to verify resequencing of the file.

NOTE: The system will not execute the deletion of menu boxes if an imcompatible execute file results. (See EXECUTE FILE ERROR MESSAGES)

|CHANGE|

|CHANGE| may be used to update only those menu picks which require keyboard or tablet input. A command such as |SET GRID SIZE| requires input on the function screen keyboard. Commands requiring input specifications may be updated in the execute file with the |CHANGE| command.

EXAMPLE: INPUT SEQUENCE

1. Pick |CHANGE|. (Highlights in reverse-video)
2. Pick |SET GRID SIZE| from the execute file display.
3. Input the new grid size specifications from the function screen keyboard.
4. Pick |ENTER|.

After the |CHANGE| command is used, the execute file display repaints on the function screen.

|INSERT|

|INSERT| may be used to add menu picks (up to 18) to the existing execute file. When |INSERT| is picked, the operator must select a menu box on the execute file display. This menu box is the starting point for the new menu pick insertions. The system is now in the execute file "record" mode; the system reads the selected menu box, then displays the command menu where valid insertions can be entered to the file.

EXAMPLE:

Step 1. |INSERT| (Highlights in reverse-video)

Step 2. |CREATE CMPNT RPT|. (System displays command menu at the appropriate level to allow insertions.)

DONE	LIST	DELETE	<-- PAGE	PAGE -->	CHANGE	INSERT	

(1) OLD PROJECT	(2) PR NETLIST-RPT	(3) DRAW SCHEMATIC	(4) EXTRACT NETLIST
(5) LOGIC RULE CHK	(6) CREATE BOM FILE	(7) CREATE NETLIST	(8) CREATE CMPNT RPT
(9) PR BOM-REPORT	(10) PR NETLIST-RPT	(11) PR EXTRACTION-LOG	(12) PR COMPONENT-RPT
(13) SAVE DRW SAME RV			

RESULT: The command menu is displayed; |MATRIX PLOT| command to be inserted.

					STATUS		

CREATE CMPNT-RPT

DRAW SCHEMATIC	PRINT OR PLOT		CANCEL ACTV DRW
DRAW SYMBOL			DIGITIZER MENU
EXECUTE MENU		SAVE DRW SAME RV	SAVE DRW NEW RV
REMOTE DIAGNOSTICS			FREE DISK SPACE
CURRENT INDEX		PROJECT INDEX	SYSTEM INDEX

Step 3. |PRINT OR PLOT| (The next menu page is displayed.)

Step 4. |MATRIX PLOT| (Input the plotting specifications.) |ENTER|

Step 5. |DONE|

Step 6. |EXECUTE MENU|

The function screen message line displays the following:

INSERTING...INSERTING...INSERTING...INSERTION COMPLETED

The system then displays the execute file with the |MATRIX PLOT| command inserted.

Resulting execute file

DONE	LIST	DELETE	<--PAGE	PAGE-->	CHANGE	INSERT	
CREATE CMPNT-RPT							

(1) OLD PROJECT	(2) OLD DRAWING	(3) DRAW SCHEMATIC	(4) EXTRACT NETLIST
(5) LOGIC RULE CHK	(6) CREATE BOM FILE	(7) CREATE NETLIST	(8) CREATE CMPNT-RPT
(9) MATRIX PLOT	(10) PR BOM-REPORT	(11) PR NETLIST-RPT	(12) PR EXTRACTION-LOG
(13) PR COMPONENT-RPT	(14) SAVE DRW SAME RV		

Step 7. |DONE|

Step 8. Select one of the following options:

|SAV EXEC SAM RV|

|SAV EXEC NEW RV|

|CANCEL ACTV EXEC|

|EDIT FILE|

NOTE: The system will not execute the insertion of any menu box if an incompatible execute file results. (See EXECUTE FILE ERROR MESSAGES.)

|LIST|

|LIST| may be used to display operator supplied inputs contained within an execute file menu pick. |LIST| is simply a method of viewing the latest inputs for each menu box in the execute file. All information is displayed on the function screen.

EXAMPLE: Pick |LIST|. (Highlights in reverse-video)

Pick |OLD PROJECT| from the execute file display.

RESULT:

DONE	LIST	DELETE	<--PAGE	PAGE-->	CHANGE	INSERT

OLD PROJECT

***Execute file TEST *** 01-MAR-84 11:38:41

(1) OLD PROJECT

: ACC-TEST

Pick |DONE| or any softkey command at the top of the function screen to re-display the execute file.

6. PRINTING THE EXECUTE FILE

When the execute file has been created and edited, the file may be printed. Use the |PRINT EXEC FILE| command to print the file with the matrix printer option.

EXAMPLE: |PRINT EXEC FILE| (EXECUTE FILE NAME) - Keyboard Input - |ENTER|

RESULT: Printed execute file

***Execute file TEST	*** 21-MAR-84 12:30:05
(1) OLD PROJECT :ACC-TEST	
(2) OLD DRAWING :SCH	
(3) DRAW.SCHEMATIC	
(4) EXTRACT NETLIST	
(5) LOGIC RULE CHK	
(6) CREATE BOM FILE	
(7) CREATE NETLIST	
(8) CREATE CMPNT-RPT	
(9) MATRIX PLOT :R :5.0000	
(10) PR BOM-REPORT	
(11) PR NETLIST-RPT	
(12) PR EXTRCTION-LOG	
(13) PR COMPONENT-RPT	
(14) SAVE DRW SAME RV	

End of EXECUTE file TEST

7. RUNNING THE EXECUTE FILE

The |RUN EXEC FILE| command is used to start the execute file. The operator must always run the execute file from the same menu location used to create the file. For example, if the execute file opens a project and proceeds to run a series of operations, the file must be started from the OLD PROJECT/NEW PROJECT menu. However, if the file is created at the DRAW SCHEMATIC menu, the operator must first interactively open the project and proceed to the DRAW SCHEMATIC menu before running the file.

To run the execute file:

Step 1. Pick the |EXECUTE MENU| command at the proper menu location.

Step 2. Pick |RUN EXEC FILE|.

- o Input the execute file name, then pick |ENTER|.

The system is now running non-interactively with the execute file displayed on the function screen. As each command is executing in the sequence recorded, it is echoed at the left side of the message line. When the command is complete, it is re-displayed in reverse-video on the function screen. This process continues until the system completes the last command recorded in the execute file.

NOTE: During the processing of an execute file, the operator may pick |CANCEL|. The command currently being processed (displayed on the function screen message line) continues until completed. Then, the execute file processing is terminated.

When the file is running, the system may encounter an error. An error may occur if an execute file menu box contains an improperly supplied input that would typically stop the system interactively, or non-interactively. For example, an execute file created to operate the auto-router that contains an improperly set parameter will stop the system. The execute file will terminate at the menu box where the error is detected. The following error message will appear on the function screen.

ERROR ENCOUNTERED WHILE PROCESSING EXECUTE FILE
PICK |DONE| TO CANCEL

The operator must always be certain that inputs contained within each execute file menu box are correct. Use |LIST| or |PRINT EXEC FILE| to check the file before running it.

8. IMPORTANT CONSIDERATIONS

1. Execute files must be created and edited from the |EXECUTE MENU|.

2. The keypad INCREMENT X,Y commands are not supported within an execute file if the operator attempts to increment an element added to the drawing with the execute file.

9. EXECUTE FILE ERROR MESSAGES

The following error messages may be encountered during the creation and editing of execute files. This section describes each error message.

EXECUTE FILE XXX WILL NOT RUN FROM THIS LEVEL

The operator tried to run the execute file from an incorrect menu location. The operator must ensure that the execute file is run from the same menu location used to create the file.

INVALID DELETION - PICK OOPS!

The operator selected a menu box for deletion that would result in an incompatible execute file. The menu boxes before and after the selected deletion would improperly sequence the file.

MAXIMUM DELETION LIMIT REACHED - PICK "ENTER".

The operator attempted to delete more than 18 menu boxes from the existing execute file display. |ENTER| may be picked to delete the 18 menu boxes before proceeding.

INVALID DELETION - FILE MUST CONTAIN AT LEAST ONE MENU BOX

The operator proceeded to delete all menu boxes displayed in the execute file. Deleting all menu boxes does not delete the file. File management must be used to delete the file from the current project file, or system index.

PLEASE SELECT EDITING OPTION AT TOP OF MENU

This error message occurs if the operator picks a menu box from the execute file display without first selecting an editing option (DELETE, CHANGE, INSERT).

ERROR: DISK SPACE LOW, PLEASE CHECK!

The system did not allow the operator to perform an execute file insertion due to low disk space. |CANCEL ACTV EXEC| must be used prior to checking the available disk space on the system.

INVALID INSERTION

The operator selected menu boxes for insertion that were incompatible with the surrounding menu boxes.

PARTIAL INSERT TO OCCUR - INVALID MENU PICKS NOT INSERTED.

Some of the menu boxes selected for insertion were not compatible with the surrounding menu boxes. The system proceeds to insert those menu picks that are compatible with the surrounding boxes.

EXECUTE FILE EMPTY!

The operator attempted to run an execute file that contained no menu boxes. File management must be used to delete empty execute files.

NOTE: An empty execute file will result if the system is re-booted while in the "create" mode.

EXECUTE FILE CANCELLED. NO PROCESSING OCCURRED.

The operator picked |RUN EXEC FILE|, then picked |CANCEL| before the first menu box processed.

PLEASE PICK "ENTER" OR "CANCEL".

This message may occur during the "create" or "insert" mode. There is not enough room for additional picks within the selected menu box.

NO INSERTION MADE.

This message will occur in the "insert" mode if the operator picks |EXECUTE MENU|.

CANNOT CREATE EXECUTE FILE -- FILE NAME ALREADY EXISTS!

The operator attempted to create an execute file with an existing file name. PROJECT INDEX, SYSTEM INDEX, or CURRENT INDEX may be used to check file names.

EXECUTE FILE NOT FOUND IN THIS PROJECT.

The operator attempted to edit or run an execute file that does not exist in the current system directory.

CANNOT OPEN EXECUTE FILE -- PLEASE TRY AGAIN!

The system did not allow the operator to open an existing execute file. The |FREE DISK SPACE| command may be used to check the amount of available space on the system.

MAXIMUM INSERT LIMIT REACHED!

This message informs the operator that 18 menu picks have been selected. The operator may proceed with additional menu picks after the first 18 selections have been inserted.

CANNOT CREATE EMPTY EXECUTE FILE.

This error occurs if the operator picks |CREATE EXEC FILE|, then proceeds to the "edit" mode with an empty execute file display. After this message appears, the system deletes the empty file.

REMOTE PERIPHERAL OPERATION

1. GENERAL INFORMATION
2. THE LOCAL AREA NETWORK
   - o HOST SYSTEM
   - o NETWORK SOFTWARE
   - o REMOTE SYSTEM
   - o REMOTE SOFTWARE
   - o VT101 TERMINAL
3. ESTABLISHING THE LOCAL AREA NETWORK
4. ACCESSING PERIPHERALS THROUGH THE LOCAL AREA NETWORK
5. CHECKING PERIPHERAL STATUS
6. REBOOTING SYSTEMS TIED TO THE NETWORK
7. RPO LIMITATIONS

REMOTE PERIPHERAL OPERATION (RPO)

1. GENERAL INFORMATION

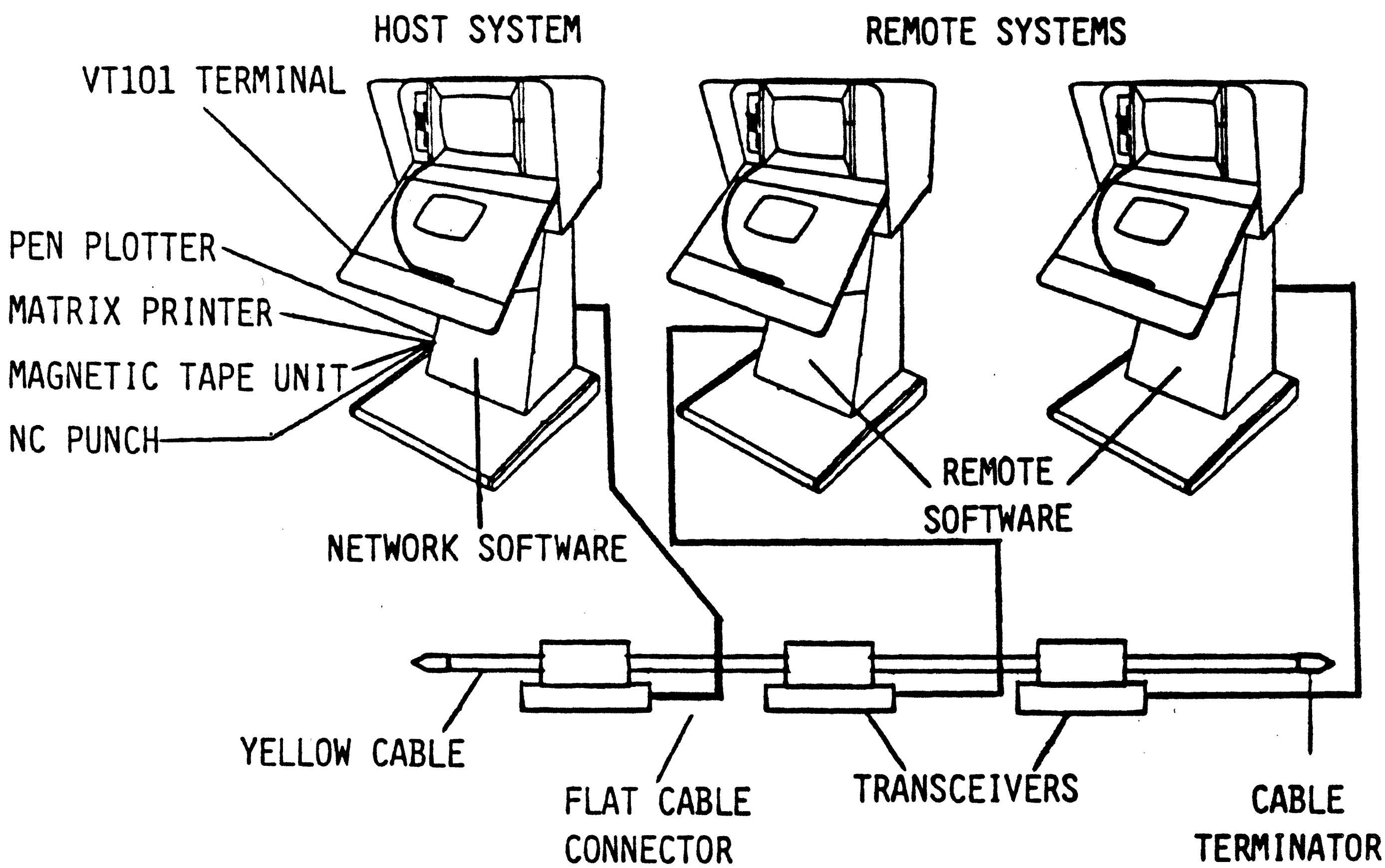
The Telesis Remote Peripheral Operation allows sharing of peripheral devices among several Telesis systems through the Ethernet® local area network. This interface provides an economical method of sharing a single set of common peripherals.

The networking capability replaces the peripheral switch used to access devices shared between two localized Telesis systems. Remote Peripheral Operation does not require "peripheral switching" once the network is established. An available peripheral (not busy) can be used from any Telesis system tied to the network.

Presently, the following peripheral devices are supported:

- o Magnetic tape unit
- o HP Pen Plotter
- o NC Tape Punch
- o Matrix Printer

The diagram below illustrates the hardware used to network Telesis systems to one set of peripheral devices.



Ethernet® is a registered trademark of Xerox Corporation.

2. THE LOCAL AREA NETWORK (LAN)

The LOCAL AREA NETWORK (LAN) allows data transfer between a REMOTE system and the HOST. This transfer of data is limited to a small geographical area.

When the network is established, the operator may use any available peripheral attached to the HOST system. For example, if the |PEN PLOT| command is picked from a REMOTE workstation, the data is then transferred through the network to the HOST system. The HOST system contains the NETWORK SOFTWARE needed to connect each peripheral device. This software then checks the status of the pen plotter. If the pen plotter is not currently operating, or allocated by another system in the network, the HOST connects the plotter, then allows the drawing information and the penplot control data to be transferred to the plotter.

The essential elements of the LOCAL AREA NETWORK are described below.

HOST SYSTEM

This is the Telesis system that has the peripheral devices attached to it. This system contains the NETWORK SOFTWARE needed to check and connect each peripheral device over the network.

NETWORK SOFTWARE (HOST only)

This is the Telesis system program that checks and reports the status of each peripheral. If a peripheral is available, the NETWORK SOFTWARE connects the device to the requesting system. However, if the peripheral is currently allocated, a message is issued to the system attempting to use the device.

EXAMPLE: PLOTTER ALREADY IN USE. COMMAND CANCELLED.

REMOTE SYSTEMS

These are the Telesis systems that share the peripherals attached to the HOST SYSTEM.

REMOTE SOFTWARE

This is the Telesis program that enables data transfer between a REMOTE SYSTEM and the HOST SYSTEM.

VT101 TERMINAL

A VT101 TERMINAL must be attached to the HOST SYSTEM. This terminal is essential if the operator wishes to check the status of all connected peripherals (See CHECKING PERIPHERAL STATUS). It is also essential for "freeing" or deallocating a connected peripheral should a system problem occur.

NOTE: VT101 terminals may be attached to any system for creating and editing text files with the |KEY EDIT TEXT| command. However, Remote Peripheral Operation DOES NOT allow transfer of text files and drawings over the network. (See RPO LIMITATIONS)

3. ESTABLISHING THE LOCAL AREA NETWORK

The LOCAL AREA NETWORK is established during loading of Telesis software on each system in the network. All essential hardware elements must be properly connected before peripherals can be accessed by each system.

NOTE: Each peripheral device must be attached to the appropriate PORT on the HOST SYSTEM. Each system tied to the network must contain the same Telesis configuration file. Configuration files allow data transfer to the proper PORT when the operator selects a peripheral start-up command, such as |PRINT| or |PEN PLOT|.

Peripheral devices are attached to the following PORTS on the HOST SYSTEM:

Matrix Plotter/Printer = PORT 1
HP Pen Plotter = PORT 4
Magnetic Tape Unit = -1 (MTØ:)
NC Tape Punch = PORT 5

During software loading, the operator (or Telesis Field Engineer) must specify the type of system being loaded; HOST or REMOTE. The system specified as the HOST will contain the NETWORK SOFTWARE; systems specified as REMOTE will contain REMOTE SOFTWARE.

NOTE: EACH WORKSTATION SHOULD HAVE A LABEL ATTACHED SPECIFYING THE TYPE OF SYSTEM; HOST OR REMOTE. THIS WILL PROPERLY IDENTIFY EACH SYSTEM SHOULD A NETWORK PROBLEM OCCUR.

4. ACCESSING PERIPHERALS THROUGH THE LOCAL AREA NETWORK

Peripheral start-up commands will automatically access available, or unallocated devices connected to the HOST SYSTEM. The operator simply selects the appropriate command on a networked system to start the device. When the following commands are picked, the NETWORK SOFTWARE will allocate an available peripheral (not busy) to a system in the NETWORK.

|PEN PLOT|, |PENPLOT BACKGROUND| - allocates the HP pen plotter

|FILE MGMT ARCHIV|, |ARCHIVE TAPE| - allocates the magnetic tape unit

|PUNCH DRILL TAPE| - allocates the NC tape punch

|MATRIX PLOT| or any PRINT command - allocates the IDS matrix plotter

When the NETWORK SOFTWARE checks the status of a device, then allocates it to a system, the device is now "busy" and unavailable to other systems. However, if the operator picks |CANCEL| or |DONE| (depending on the device allocated), the device is then cleared and available to the entire network. The network automatically deallocates a peripheral device when it stops, and when the operator picks |DONE|; the device is then available to other systems in the network.

A message will appear on the function screen if an operator attempts to access a device that is allocated to another system. One of the following messages will appear.

PRINTER ALREADY IN USE. COMMAND CANCELLED.
TAPE ALREADY IN USE. COMMAND CANCELLED.
PLOTTER ALREADY IN USE. COMMAND CANCELLED.
PUNCH ALREADY IN USE. COMMAND CANCELLED.

5. CHECKING PERIPHERAL STATUS

A later release will allow the operator to check network and peripheral status with the VT101 terminal connected to the HOST SYSTEM. The status information displayed on the terminal will include peripherals currently allocated, and the status of the network.

Currently, the operator may check the status of the network with the softkey |PRINT| command on each system. The matrix plotter/printer must be connected to the HOST SYSTEM (at the appropriate PORT), and must exist in the configuration file for each networked system. The following procedure may be used:

From each system in the network (HOST and REMOTE) -

Step 1. Pick |CURRENT INDEX|

Step 2. Pick the softkey |PRINT| command.

Follow this procedure for each system, one at a time. If the matrix printer operates, the network is functioning.

6. REBOOTING SYSTEMS TIED TO THE NETWORK

This section describes the results of rebooting any system tied to the network with allocated and deallocated peripherals. Should a system failure occur, the operator must be aware of the results when rebooting each system in the network.

GENERAL INFORMATION ON REBOOTING THE HOST SYSTEM

Whenever the operator reboots the HOST SYSTEM, recovery of the NETWORK SOFTWARE occurs first. This allows the network to become active so that peripherals can be accessed by REMOTE SYSTEMS without waiting for the entire rebooting process to complete.

REBOOTING THE HOST SYSTEM WHILE A PERIPHERAL IS ALLOCATED BY A REMOTE

If the operator reboots the HOST SYSTEM, the peripheral device may continue to operate for a short time before stopping. The peripheral device request will then "time-out" for approximately one minute. If the HOST SYSTEM is rebooted, with the NETWORK SOFTWARE recovering within this "time-out" period, the peripheral will continue.

However, if the HOST SYSTEM fails and is not rebooted, or if the NETWORK SOFTWARE does not recover within the "time-out" period, one of the following messages will appear on the REMOTE SYSTEM:

CONNECTION TO PRINTER LOST. PRINT CANCELLED.
CONNECTION TO TAPE LOST. ARCHIVE CANCELLED.
CONNECTION TO PLOTTER LOST. PLOT CANCELLED.
CONNECTION TO PUNCH LOST. PUNCH CANCELLED.

NOTE: If the HOST SYSTEM "hangs" within an application task, the operator should wait until all allocated peripherals have STOPPED before rebooting the HOST.

REBOOTING A HOST OR REMOTE SYSTEM WITH NO PERIPHERALS CURRENTLY ALLOCATED

If a failure occurs on any system tied to the network, with no peripherals currently allocated, the operator may simply reboot the system. This has no effect on other systems in the network.

HOST OR REMOTE SYSTEM "HANGS" AFTER ALLOCATING A PERIPHERAL

If this occurs, the operator should wait until the peripheral device finishes before rebooting the system allocating the device.

However, if the peripheral is allocated and not operating, the peripheral should be deallocated prior to rebooting. This will allow other systems in the network accessibility to that peripheral. Otherwise, the peripheral will be declared "busy" when a system in the network attempts to access it.

The peripheral device may be deallocated prior to rebooting with the VT101 terminal attached to the HOST SYSTEM. Type in the following information at the terminal:

@DEALLOCAT TTX

X = the PORT number of the peripheral to be deallocated.

Peripheral devices are attached to the following PORTS:

Matrix Plotter/Printer = PORT 1
HP Pen Plotter = PORT 4
Magnetic Tape Unit = -1 (Type @DEALLOCATE MTO:)
NC Tape Punch = PORT 5

The following message will appear on the VT101:

DEAL - NETWORK DEVICE DEALLOCATED

After rebooting the system, the operator does not need to "re-allocate" the peripheral with the VT101 terminal. The software is automatically re-established after rebooting; the operator may then use the "deallocated" peripheral device with a peripheral start-up command.

7. RPO LIMITATIONS

Remote Peripheral Operation is replacement for the peripheral switch only. Please note the current limitations:

1. RPO does not support text file transfer between systems.
2. RPO does not support transfer of drawings between systems.
3. RPO does not support sharing of the floppy disk drive.
4. RPO does not support the CALCOMP digitizer.

CONVERTING LINE GRAPHICS

- o USING THE CONVERT LINE GRAPHICS COMMAND ON TELESIS 2.0 AND TELESIS 2.1 SYSTEMS
- o WHEN TO USE THE CONVERT LINE GRAPHICS COMMAND
- o THE GRAPHICS SCREEN DISPLAY DURING CONVERT LINE GRAPHICS

USING THE |CONVERT LINE GRAPHICS| COMMAND ON TELESIS 2.0 AND TELESIS 2.1 SYSTEMS

The operator must use the |CONVERT LINE GRAPHICS| command to update all drawings generated prior to the 8304-E software release. This one-pick command allows the operator to update the all existing lines (greater than 0-width) to the new "filled line" graphics now on Telesis 2.1 * systems only.

This command also updates the line graphics of any drawing transferred to and from Telesis systems of different display configurations.

* Telesis 2.1 - includes the graphics processor for the WORLD, ROAM and ZOOM softkeys.

  Telesis 2.0 - systems that do not contain the graphics processor.

WHEN TO USE THE |CONVERT LINE GRAPHICS| COMMAND

The |CONVERT LINE GRAPHICS| command is located on the SYMBOL, SCHEMATIC and BOARD menus on Telesis 2.0 and 2.1 systems. Use |CONVERT LINE GRAPHICS| on the actively displayed drawing if any of the following conditions exist.

ON TELESIS 2.1 SYSTEMS (GRAPHICS PROCESSOR)

1. Use |CONVERT LINE GRAPHICS| on drawings created on Telesis 2.1 systems prior to 8304-E software release. All lines-at-width reconstructed during |CONVERT LINE GRAPHICS| will be displayed as filled lines on the graphics screen.

   NOTE: If the operator does not pick the |CONVERT LINE GRAPHICS| command after opening a 2.1 drawing created prior to 8304-E, existing lines-at-width on the drawing will remain "unfilled", while all new lines and connect-lines at-width added to the drawing will be displayed as "filled" on the drawing.

2. Use |CONVERT LINE GRAPHICS| on all drawings generated on Telesis 2.0 that now exist on a Telesis 2.1 system. All lines-at-width reconstructed during |CONVERT LINE GRAPHICS| will be displayed as filled lines on the graphics screen.

ON TELESIS 2.0 SYSTEMS

1. If a drawing is created on a Telesis 2.1 system, then transferred to a Telesis 2.0, the system will display ALL lines and connect-lines at 0-width when the operator opens the drawing.

   When |CONVERT LINE GRAPHICS| is picked, all lines at-width that appeared in the original drawing (2.1 system) will be reconstructed as "unfilled" double lines (at-width) on Telesis 2.0.

   NOTE: If the operator does not pick |CONVERT LINE GRAPHICS| after opening the drawing, the 0-width lines and connect lines will remain, while new lines and connect lines at-width added to the drawing will appear as "unfilled" double lines.

2. The operator does not need to use |CONVERT LINE GRAPHICS| on drawings transferred to and from Telesis 2.0 systems.

THE GRAPHICS SCREEN DISPLAY DURING |CONVERT LINE GRAPHICS|

When |CONVERT LINE GRAPHICS| is picked, the function screen displays the following message:

THIS MENU SELECTION IS USED TO CONVERT ALL LINES (WITH A WIDTH GREATER THAN ZERO) TO THE CURRENT GRAPHIC CONFIGURATION.

PLEASE PICK |PAGE ->| TO CONTINUE

OR

PLEASE PICK |CANCEL|

If the operator picks |PAGE ->|, the system will issue the following statement on the function screen message line:

"CONVERSION OF LINES STARTED"

The drawing will then repaint on the graphics screen, with all lines (greater than 0-width) converted to the existing graphics display configuration on the system. When |CONVERT LINE GRAPHICS| is complete, the function screen message line displays:

"CONVERSION COMPLETE"

NOTE: Once the operator picks |PAGE->|, the |CANCEL| command will not terminate the |CONVERT LINE GRAPHICS| command.

The drawing conversion process must complete before the system can be used interactively.

OPERATOR'S MANUAL INDEX

KEY TO INDEX:

Bas	= BASIC	Lib	= LIBRARY	Post	= POST
Graph/Proc	= GRAPH/PROC	NetDB	= NETDB	PP	= PRINTPLOT
F/A	= FILE/ARCH	Place	= PLACE	Util	= UTILITIES
Int	= INTERCON	GRAPH/PRO	= TELESIS GRAPHICS PROCESSOR		