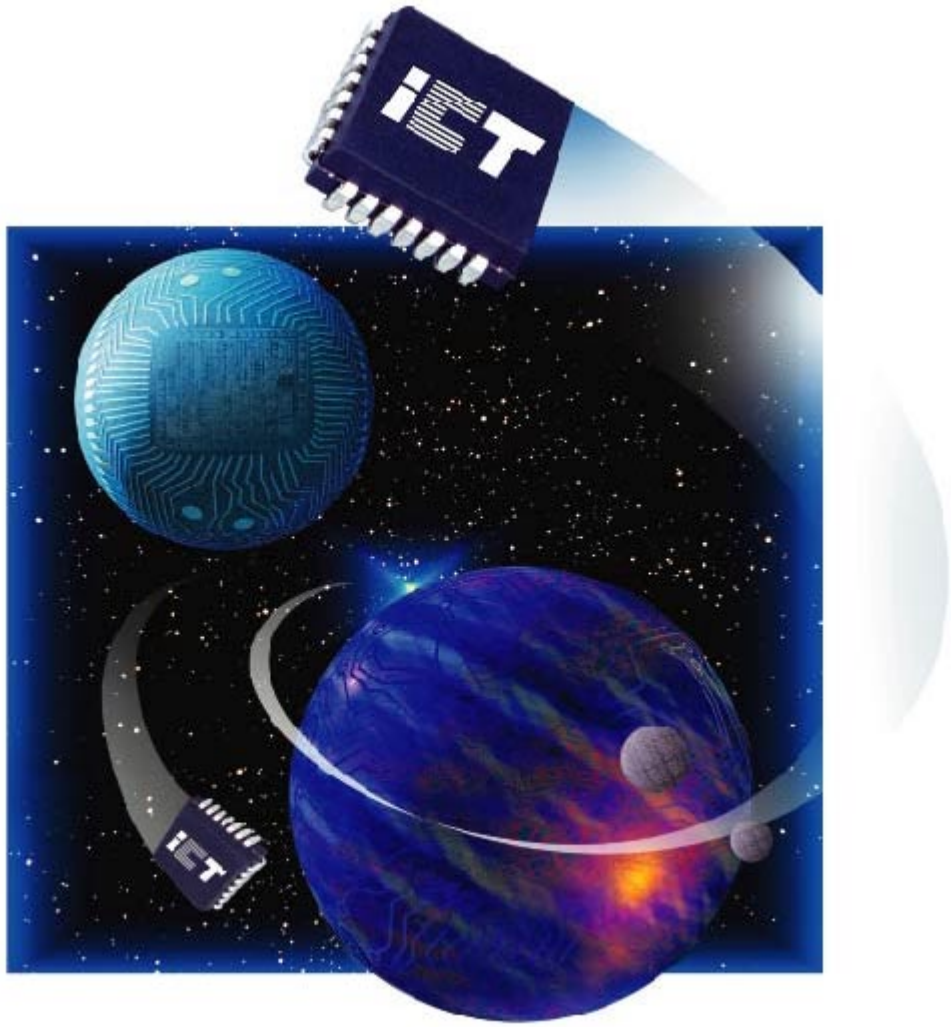
WinPLACE™
Software User's Manual



Integrated Circuit Technology Corp.

September 2000

Table of Contents

1 Introduction to WinPLACE™

1.1 WinPLACE™ Advanced Development Software

Welcome to the WinPLACE™ Advanced Development Software from Integrated Circuit Technology Corporation. WinPLACE™ (PEEL™ Logic Architectural Compiler and Editor) is an enhanced development package that offers complete support for ICT's family of PEEL™ (Programmable Electrically Erasable Logic) Arrays and Devices.

The WinPLACE™ Advanced Development Software is free to qualified PLD users. To obtain a copy of WinPLACE™ please contact your local ICT representative.

WinPLACE™ Features

Architectural Editor

WinPLACE™ incorporates an innovative architectural editor that graphically illustrates and controls the architectures, logic equations, state-diagram and truth-table entries, hence making the overall design easy to understand while allowing for optimum utilization.

Logic Compiler

The WinPLACE™ compiler performs logic transformation, allowing equations to be specified in a variety of formats. The compiler also features five levels of user-selectable logic reduction (including "auto-demorganization") making it possible to fit more logic into every design.

Logic Simulator

WinPLACE™ provides a multi-level logic simulator that lets the external and internal signals be fully simulated, analyzed and edited via a graphically illustrated waveform display.

1.2 WinPLACE™ Introduction

The WinPLACE™ Software portion of this description is organized in six sections. Before trying to design with the WinPLACE™ software, make sure you read through the three sections: "Introduction", "Installation" and "Getting Started". By doing so you will save yourself time.

After you have completed installing the WinPLACE™ software and have become familiar with the basic operations, you can refer to the "Operation Reference Guide" and "WinPLACE™ Design Language Reference Guide" section as you implement your first design. Documentation describing several application examples is provided in Chapter 5. These examples are also included on diskette and will automatically be loaded during installation.

While using this manual and the software you may need to reference the product specifications in the data book.

The software operations and features described in this manual are referenced to WinPLACE™ Version 0.5.4. For additional information on new features and manual corrections, please refer to the README.DOC file on the WinPLACE™ disk.

1.3 PEEL™ Device and PEEL™ Array support

The devices supported in the WinPLACE™ software include:

PEEL™ Arrays

PA7024	PA7128	PA7140
PA7536	PA7540	PA7572

PEEL™ Devices

PEEL™18CV8	PEEL™18CV8Z	PEEL™18LV8Z
PEEL™16CV8	PEEL™22CV10A	PEEL™22CV10AZ
PEEL™22LV10AZ		

Additional devices will be supported in future software versions.

1.4 Translation

Converting APEEL™ File ".APL" to WinPLACE™ File ".PSF"

ICT has discontinued further development of its original PEEL™ software named APEEL. WinPLACE™ now supports all products. If you still use APEEL for development your designs can be converted to the WinPLACE™ design language format . This allows the WinPLACE™ enhanced features available for the PEEL™ Arrays to be used to implement designs for the lower density PEEL™ devices.

Use Translators → APEEL to PSF under the Options menu in the design window to convert your designs.

The extension of the filename defaults to ".APL" if not specified.

JEDEC File Translation

The JEDEC-file translation utility of the WinPLACE™ Software translates JEDEC files created for programming other PLD's (PAL's, GAL's, EPLD's etc.,) to JEDEC files used for programming PEEL™ Devices. The translated JEDEC file will program a PEEL™ Device to be a pin-to-pin replacement for the original PLD.

The JEDEC file translation is available in Options → Translators → JEDEC to JEDEC in the design window.

The translated PEEL™ JEDEC file is given the name of the original file with the ".JED" extension modified to ".JEX". The ".JEX" file can then be used to program your PEEL™ devices.

Devices that translate to the PEEL™18CV8

PAL16L8	PAL16R8	PAL16R6	PAL16R4
PAL16P8	PAL16RP8	PAL16RP6	PAL16RP4
PAL10L8	PAL12L6	PAL14L4	PAL16L2
PAL10H8	PAL12H6	PAL14H4	PAL16H2
PAL16H8	PAL16LD8	PAL16HD8	PAL18P8
PAL18V8	GAL16V8	EP310	EP320
5C031	5C032	EP330	PAL18U8
PALCE16V8	85C220		

Devices that translate to the PEEL™22CV10A

PAL20L8	PAL20R8	PAL20R6	PAL20R4
PAL20L10	PAL20L2	PAL18L4	PAL16L6
PAL14L8	PAL12L10	PAL22V10	PAL20G10
PAL20AP10	PAL20RP8	PAL20RP6	PAL20RP4
PAL20ARP4	PAL20ARP6	PAL20ARP8	PAL20ARP10
GAL20V8	PALCE20V8	GAL22V10	

Note: For any devices on the following list, please contact ICT Technical Support for translations

Devices that translate to the PEEL™18CV8Z or the PEEL™18LV8Z

PLC18V8Z35/I	PLC18V8Z25/IA

Devices that translate to the PA7536

Lattice			
26CV12B-15LP	26CV12B-15LJ	26CV12B-20LP	26CV12B-20LJ
Vantis (AMD)			
24V10H-15P	24V10H-15J	24V10H-25P	24V10H-25J
26V10H-15P	26V10H-15J	26V10H-25P	26V10H-25J

Devices that translate to the PA7540

Altera			
EP312	EP600PC	EP600PC-45	EP600PC-3
EP600DC*	EP600DC-3*	EP610PC-35	EP610PC-30
EP610PC-25T	EP610PC-25	EP610PC-20T	EP610PC-20
EP610PC-15T	EP610PC-15	EP610PI-30	EP610DC-35*
EP610DC-30*	EP610DI-35*	EP610DI-30*	EP610SC-25
EP610DC-20	EP610DC-15		
Atmel			
ATV750-20DC*	ATV750-20GC*	ATV750-20JC	ATV750-20KC*
ATV750-20PC	ATV750-20SC	ATV750-25DC*	ATV750-25GC*
ATV750-25JC	ATV750-25KC*	ATV750-25PC	ATV750-25SC
ATV750-30DC*	ATV750-30GC*	ATV750-30JC	ATV750-30KC*
ATV750-30PC	ATV750-30SC	ATV750-35DC*	ATV750-35GC*
ATV750-35JC	ATV750-35KC	ATV750-35PC	ATV750-35SC
Cypress			
20RA10-15PC	20RA10-15WC*	20RA10-20PC	20RA10-20WC*
7C324-A15HC*	7C324-A15JC	7C324-A20HC*	7C324-A20JC
Intel			
PPLD610-15	PDLD610-15*	PPLD610-25	PDLD610-25*
D5C060-45*	P5C060-45	D5C060-55*	P5C060-55
D5AC312-25*	P5AC312-25	D5AC312-30*	P5AC312-30
Lattice			
20RA10-15LP	20RA10-15LJ	20RA10-20LP	20RA10-20LJ
20RA10-30LP	20RA10-30LJ	20XV10B-15LP	20XV10B-15LJ
20XV10B-20LP	20XV10B-20LJ	6001B-30LP	6001B-30LJ
6002B-15LP	6002B-15LJ	6002B-20LP	6002B-20LJ
National			
20RA10-15NC	20RA10-20NC	20RA10-25NC	20RA10-15VC
20RA10-20VC	20RA10-25VC	6001-30LNC	6001-30LVC

Philips			
PLS173N	PLS173BN	PLS179N	PLC42VA12FA*
PLC42VA12N	PLCVA12A		
TI			
EP630-15CNT	EP630-20CNT		
Vantis (AMD)			
29M16H-25P	29M16H-25J	29MA16H-25P	29MA16H-25J
610H-15P	610H-25P	20RA10H-20P	20RA10H-20J

*These devices use ceramic DIP or J-lead packages; the equivalent ICT device uses plastic devices.

Devices that translate to the PA7572

Altera			
EP910L	EP900PC	EP900PC-3	EP900PC-2
EP910PC-40	EP910PC-35	EP910PC-30	EP910PC-30T
EP910DC-40*	EP910DC-35*	EP910DC30*	
Intel			
PPLD910-15	DPLD910-15*	PPLD910-25	DPLD910-25*
D5C090-50*	P5C090-50	D5C090-60	P5C090-60

*These devices use ceramic DIP or J-lead packages; the equivalent ICT device uses plastic devices.

1.5 WinPLACE™ System Requirements

- IBM PC-AT, 386, 486, Pentium or compatible
- Minimum 32 MB of RAM
- A hard disk with at least 8.0 MB of free space
- Logitech or Microsoft mouse with driver
- Windows 95 or 98

1.6 Hard Disk Installation Procedure

- Please uninstall any earlier versions of WinPLACE software you may have on your computer before you install this version.
- Turn the computer on.
- Boot-up Windows 95 or 98.
- Insert the WinPLACE™ CD into your CD-ROM drive.

- Open your CD drive and click on the Setup program. This will automatically load the installation process.
- Click “Next” on the Welcome screen.
- Read and accept the software license agreement.
- Now you will be asked for the destination folder. The default is: C:\Program Files\WinPLACE\ If this is ok, click “Next”. If you would like to change the destination folder click “Browse”. Select the folder you want, then click “OK” followed with “Next”.
- Finally, add a program icon to the “Program Folder”, which will be placed in the “Start Menu”. The default directory is “WinPLACE”. Click “Next” when done.
- The WinPLACE™ installer will now copy the correct files onto your hard drive. When it is done, click “Finish”.
- Setup is now complete.

2 Getting Started with WinPLACE™

2.1 Running the WinPLACE™ Software

Once WinPLACE™ is properly installed; it can be invoked in one of three ways:

1) Click on the “Start” menu, and then click on “Programs”. Click on WinPLACE™

2) Double-click on the “My Computer” folder on the desktop. Then, open the drive that WinPLACE™ was installed in. Locate the WinPLACE folder (it is most likely in the Program Files folder). Open the folder and double-click on the WinPLACE™ program (place_w5.exe).

3) Create a shortcut for WinPLACE™ and place it on the desktop. Then, from now on, open the program by double-clicking on the shortcut.

Before using the WinPLACE™ software, please read through the following sections on "Using the Mouse" and "Getting HELP"

2.2 Using the Mouse

The WinPLACE™ software and manual will commonly refer to several mouse actions using the nomenclatures specified in Table 2-1.

Term	Mouse Action
Click	Press/release the left button of the mouse.
Click - R	Press/release the right button of the mouse.
Click - LH	Press/hold the left button of the mouse while moving the mouse.
Click - RH	Press/hold the right button of the mouse while moving the mouse.
Double - Click	Press/release the left button twice.
The Middle Button	The middle button is not used in WinPLACE™. However, if your mouse comes equipped with a scroll button in the middle, it may be used only in the Text Editor.

Table 2-1 Nomenclatures for WinPLACE mouse actions

"Click" - press/release left button

"Click" is used in all operations and modes to make a selection. Make a selection by moving the mouse cursor to the desired item and then press/release the left button of the mouse. In many cases the selected item

will be highlighted. Items that can be selected include pop-down and pop-up menu windows, architectural elements in the design operation such as Logic Control Cells (LCC), I/O Cells (IOC), Input Cells (INC), Global Cells (GBC), test vector waveforms (Simulate operation), etc.

"Click-R" press/release right button

"Click-R" in most cases is used to exit, complete or return from the current function being performed. It also has two different functions in the Pin Block Diagram and in the Text Editor. In the Pin Block Diagram, it displays a pop-up menu, which contains the commands: “Copy”, “Swap”, and “Clear”. In the Text Editor, it displays a different pop-up menu containing the commands: “Undo”, “Cut”, “Copy”, “Paste”, “Delete”, and “Select All”.

"Click-LH" or press/hold the left button while moving the mouse

Click-LH is used in the Design and Simulate operations and also in the Text Editor. In the Design operation, it is used to scroll from one LCC/IOC to another in the LCC/IOC screen. In the Simulate operation, it is used for panning in the waveform screen as well as block selection for the copy, move, and delete functions in the "Edit" mode.

"Click-RH" and "Click-MH" or press/hold the right and middle button respectively while moving the mouse

For 3-button mouse systems, click-MH (click-RH for 2-button mouse) is used to display the menu options in the WinPLACE™ text editor utilized in the Design operation. While holding the middle mouse button down (right button for 2-button mouse), move the mouse cursor and click at the menu option. Once the option is selected, the middle button can be released.

Mouse Support in the WinPLACE™ Text Editor

The mouse is supported in the WinPLACE™ Text Editor and Compile Operation. You can find the Text Editor at the bottom of the screen in the Design and Compile functions. Refer to Section 3.24 “WinPLACE™ Text Editor” for detailed information on the Text Editor. To initiate the mouse support, right click in the WinPLACE™ Text Editor. A pop-down menu will appear next to the cursor containing the following commands:

Undo……Reverses the last action that was made in the editor.

Copy……Copies the currently highlighted text into the computer’s clipboard.

Cut……Almost the same as the “Copy” function however, it clears the text as well as copying it onto a clipboard.

Paste……...Re-inserts the text that was either “Cut” or “Copied” from the document.

Delete……..This is simply to clear the highlighted text without using the keyboard.

Select All…This function will highlight all of the text in the WinPLACE™ Text Editor.

Mouse Support in the WinPLACE™ Design Operation

The mouse is also supported in the WinPLACE™ Design Operation. This is mainly for the Pin Block Diagram screen. For the Text Editor mouse support, refer to the above. To initiate the mouse support, right click in the WinPLACE™ Pin Block Diagram screen. A pop-down menu will appear next to the cursor containing the following commands:

Copy……Copies the selected cell and all of its configurations (except the Label) into the computer's clipboard.

Swap……Switches two I/O cells, two Input cells, or two logic control cells with each other. (i.e., I/O-to-I/O or Input-to-Input or LCC-to-LCC)

Clear……Deletes all the user-entered attributes on the cell, reverting it to a default cell.

2.3 Getting HELP

The WinPLACE™ software incorporates an on-line HELP feature, which provides information and procedures for most WinPLACE™ functions and modes. A general HELP menu is provided in the Help pop-down menu.

2.4 Guided Tour through the WinPLACE™ software

To quicken the learning process, this section discusses some basic procedures commonly used in the WinPLACE™ software. The device used in this guided tour is the PA7540. Some of the terms used in this section may only be applicable to the PA7540 device (or the PEEL™ Array family of devices). For instance, terms such as LCC (Logic Control Cell) and GBC (Global Cell) pertain only to devices in the PEEL™ Array family.

If you are a first-time WinPLACE™ user, ICT recommends that you run the WinPLACE™ software while reading this section. By actually performing the instructions (specified in italics), you will be able to get a more complete understanding of the features, modes or functions found in the WinPLACE™ software.

There are three main operations that can be performed with WinPLACE™: Design, Compile, and Simulate. When first entering WinPLACE™, it will default to the Design Operation. The display will show the PA7540 PEEL™ Array pin block diagram (Figure 2-1).

Note that WinPLACE™ automatically loads the ANEW template file upon initial boot-up. In Figure 2-1, the WinPLACE™ software has loaded the ANEW file (ANEWPA7540.PSF) for the PA7540 device. There is an ANEW template file for each device supported by the WinPLACE™ software. For instance, the ANEWPA7140.PSF file is the template file for the PA7140 device. Each ANEW file contains the device's default cell configurations.

At the top of the screen, there are five pop-down menu options: File, Design, Operation, Utilities, and Options (available only in the Design operation). Move the mouse cursor to the "Operation" menu. A pop-down window will appear showing the three main operations (Figure 2-2). Note the menu option titled "Design" to the left of the Operation menu. This menu is called the "command" menu. Each time a new operation is selected, this command menu will change to allow the selection of commands specific for that

operation. The command menu is also used as an indicator for the current operation.

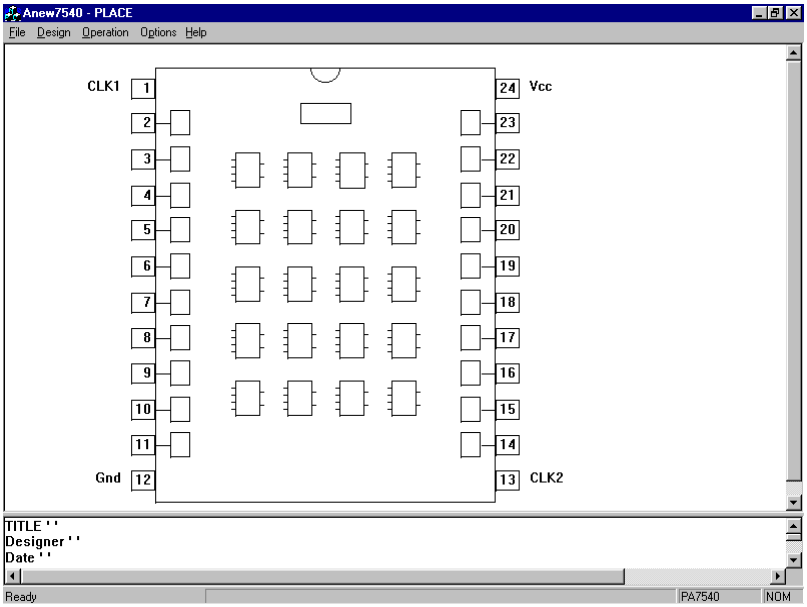


Figure 2-1 The WinPLACE™ startup default

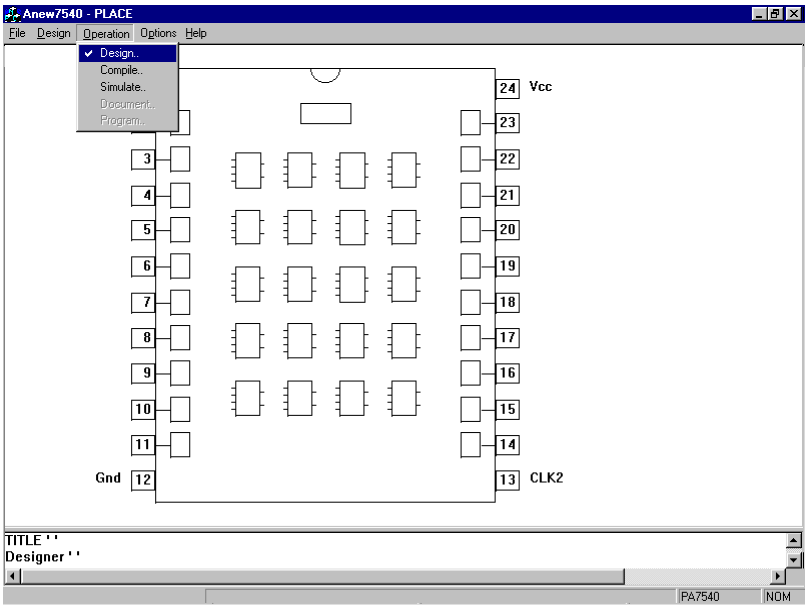


Figure 2-2 The Operations pop-down menu

Move the mouse cursor to the "File" menu option and Click the "Read" command. A list of example WinPLACE™ Source Files (.PSF) will appear (Figure 2-3). The design examples provided with the WinPLACE™ software include:

· Multiple-Application Design Example (PA7540, DEMO1A.PSF)

· 8-Bit Counter with Hold, Reset and Preset (PA7024, COUNTER1 .PSF)

· Bi-Directional I/O Port (PA7024, BI_PORT.PSF)

· Bus Programmable Clock Generator/Timer (PA7024, TIMER.PSF)

· Blackjack Machine Example (PA7024, JACK7024.PSF)

· Timer/Counter (PA7140, TC7140.PSF)

. Register Configurations (REG7536.PSF)

· Basic Gates (18CV8, V8GATES.PSF)

· Basic Registers and Latches (18CV8, V8REGS.PSF)

· Clock Divider and Address Decoder (18CV8, V8CLKADD.PSF)

· Bus Programmable 8-to-1 Multiplexer (18CV8, V8BUSMUX.PSF)

· 8-Bit Counter with Function Controls (18CV8, V8FCNTR.PSF)

· Change-of-state Input Port with Interrupt (18CV8, V8CPORT.PSF)

· Octal Synchronization Circuits (18CV8, V8SYNC.PSF)

· 8-Bit Up/Down Loadable Counter with Carry-out or Borrow-in (PEEL22CV10A, V10CNT8.PSF)

· 9-Bit Even/Odd Parity Generator/Checker (PEEL22CV10A, PARV10A.PSF)

· 8-Bit Change-of-State Input Port with Interrupt (PEEL22CV10A+, V10ZPORT.PSF)

As shown in Figure 2-3, there are two methods of making a selection from the file menu window:

1. Click to highlight a file or directory and then click the “Open” button.

2. "Double-Click" at the file or directory. The first click highlights the selection and makes the name appear in the “File Name” text box. The second click makes the selection.

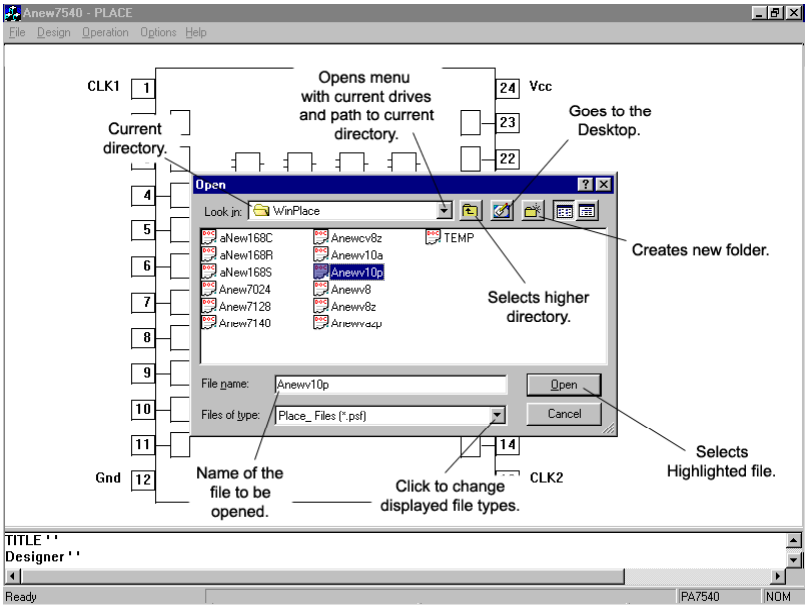


Figure 2-3 Reading a demonstration file from the File menu

Click the DEMO1A.PSF file. The WinPLACE™ Design operation will once again be displayed, but this time with the DEMO1A demonstration design tile. For more detailed information on the demonstration example files refer to Chapter 5 of this manual.

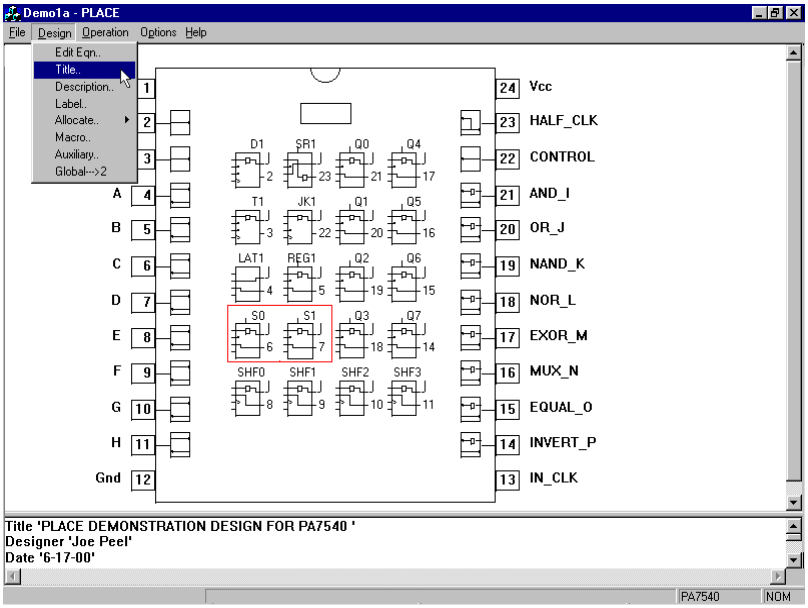


Figure 2-4 The "Design" command menu

Once the file is loaded, *move the mouse to the "Design" menu option at the top of the screen (*Figure 2-4*). Click at the "TITLE" command.* The cursor in the text edit area will automatically skip the "*TITLE*" section between the single quotation marks. The title of the design can now be entered with your keyboard (Figure 2-5). *Use the down/up arrows or click to move the cursor to the next Title field.*

Move the cursor to the next line labeled, "Designer." (Figure 2-5) *Click between the single quotation marks to edit the name of the current designer.* The program does not require this, but it is convenient for other user reference.

By clicking to the next line, labeled "Date," you can edit the last entered date. (Figure 2-5) This option is convenient to signify when the device was created, edited last, completed, etc. This option is liable to change at any time, without affecting the device.

Move the mouse cursor back to the "Design" menu option and click the command listed as "Description". (Figure 2-5) This procedure allows the design description to be entered or modified by having the WinPLACE™ software automatically move the text cursor to the "Description" area in the Text Editor.

The size of the Text Editor may be freely changed to add or subtract view space of both sections in the window. Click-LH (left-click and hold) on the line that separates the Text Editor and the Pin Block Diagram, then drag it up or down to increase/decrease the window of your choice. In Figure 2-5 the size of the Text Editor has been increased, blocking out some of the Diagram.

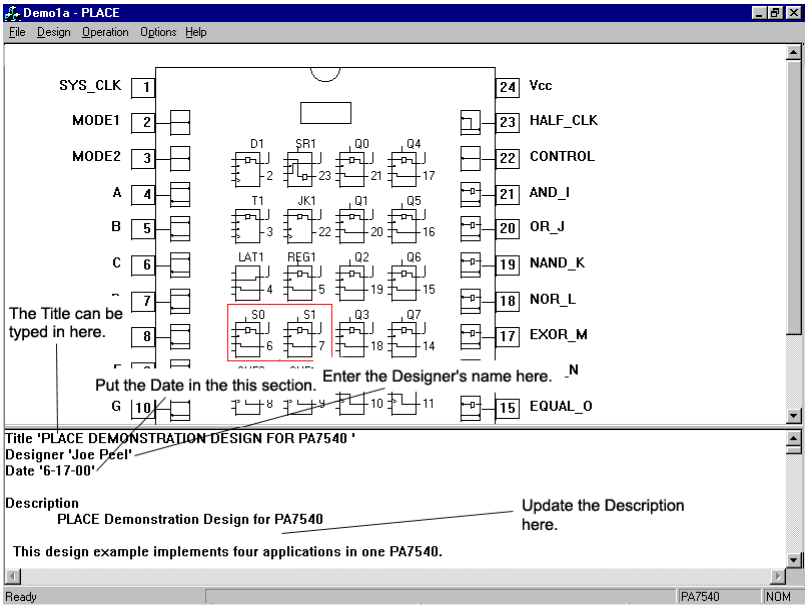


Figure 2-5 The "Description" window

From the pin block diagram screen move the mouse cursor to the "Design" menu option and click the "Label" command. A window will appear containing two parts. The top part should be gray and the bottom part should be white, both are empty. *Now move the cursor to one of the Logic Control Cells (LCC's), Input/Output Cells (IOC's), or Input Cells (INC's - PA7140 only) and click the mouse button.* The window will now display the current label (or name) for that Cell. The gray (top) half of the window will contain the name of the cell selected, and the white (bottom) half of the window will display the user-given label. Figure 2-6 shows that I/O cell #2 was selected in the Label mode. To change the label, use the [Back Space] key and type in the new label followed by the [Enter] key. If the label is

already blank then simply type in the new label followed by the [Enter] key. The Label command is used to define all IOC's, INC's and LCC's that are used in a design.

The architecture of the LCC's, IOC's, or INC's can be configured prior to labeling. However, labels must be specified before the equations, state-diagrams, or truth-tables can be entered. Click-R once to exit the "Label" mode. Please refer to Chapter 3 "Operation Reference Guide" for options on the "Label" command.

Now, move the cursor to one of the LCC's. Note that both the LCC and its interconnected IOC will be highlighted. Click the mouse to bring up the associated "LCC and IOC Screen". This screen displays a close-up view of the selected LCC and its associated IOC configuration.

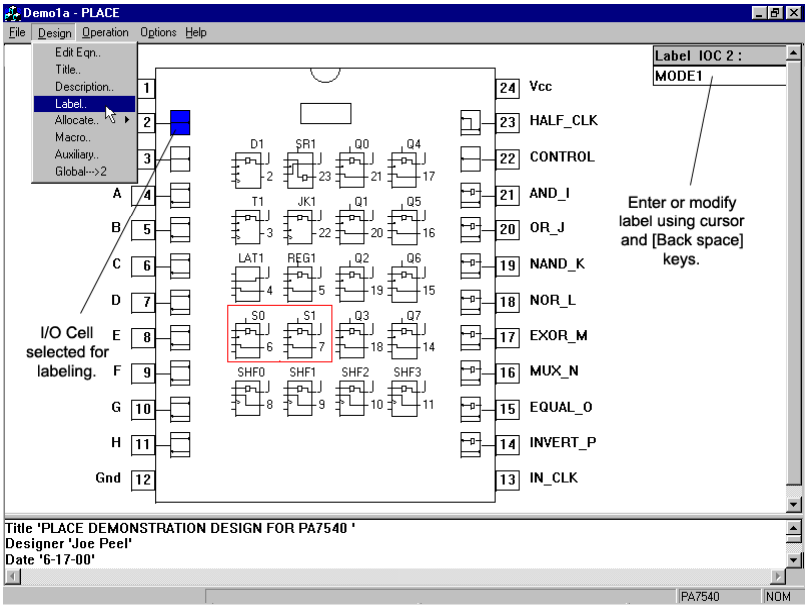


Figure 2-6 The "Label" command for Pin, INC, IOC and LCC

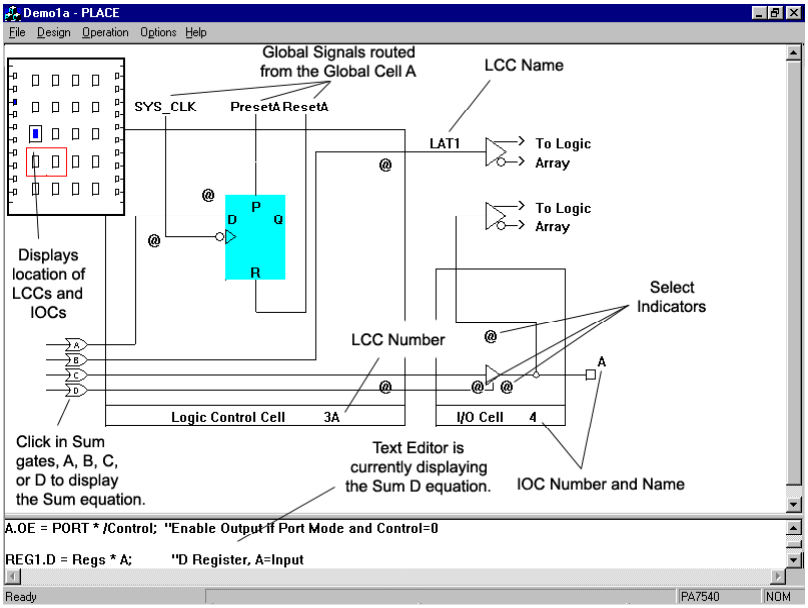


Figure 2-7 Configuring the LCC and IOC Architecture

Select the configuration of the cells by clicking at any of the "Select Indicators" (Figure 2-7). With the PA7540 device, over 4000 configurations can be selected by clicking at each of the select indicators. Any time the mouse cursor is moved away from a select indicator, the pop-up window will be cleared. *Click-LH (press/hold left mouse button) and move the cursor left and right or up and down.* This allows panning from one cell to another without returning to the pin block screen. Notice that the miniature pin block diagram in the upper left corner displays the current LCC/IOC location.

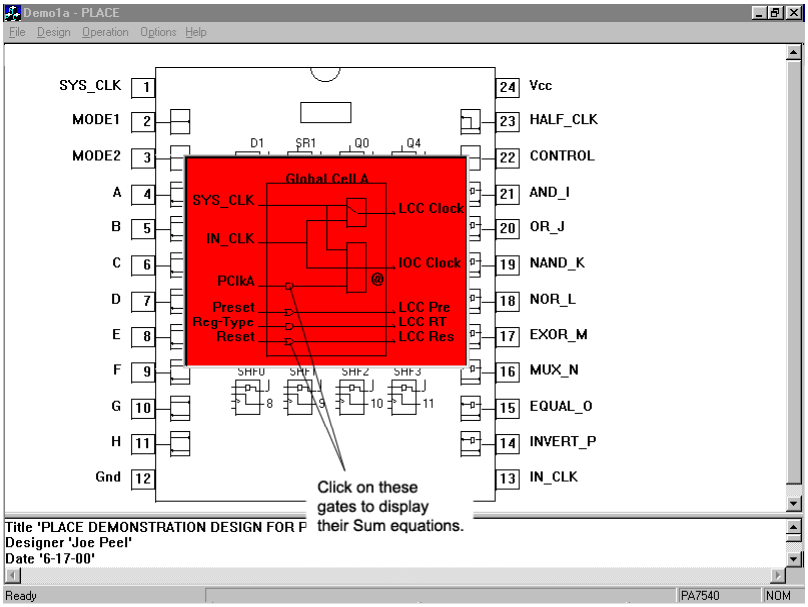


Figure 2-8 Configuring the Global Cell A (GBC) Architecture

Move the cursor into the "D" OR or Sum-D gate (shown in Figure 2-7*) and double-click the mouse button.* In the Text Editor, the cursor will

automatically jump to the current equation for the selected OR gate. *To return to the pin block diagram screen, simply click on the diagram.*

The same process can select the Global Cell (GBC) configuration, i.e. by clicking the select indicators in the cell (Figure 2-8).

There are multiple global cells in all PEEL™ Arrays. For instance, the PA7540 device has two global cells that are called Global Cell A and B. The default condition for the PA7540 (and all PEEL™ Arrays) is the one global cell mode. The two global cell mode can be selected by clicking at the "Global = 2" command found in the "Design" menu window. With two global cells, Global Cells A and B control global signals for all LCC's connected to the IOC's located on the left (pins 2 to 11) and right side (pins 14 to 23) of the pin block diagram respectively. Please refer to the PEEL™ Array data sheet for more information on the global cells.

In the pin block screen, move the mouse cursor to the "Design" Command pop-down menu again. Click the command listed as "Edit Eqn" (this stands for Edit Equation). You will notice that there is now a check mark next to the “Edit Eqn” mode name in the “Design” menu. Move the cursor to one of the four inputs of any LCC and the input will be highlighted. Click to select the input equation. In Figure 2-9, the cursor in the Text Editor, automatically jumps to the Sum equation for that cell.

The Text Editor, which is located at the bottom of the screen, is used for editing Boolean equations, state diagram and truth table syntax. Once inside the editor, most of the standard WordStar™ commands can be used. Also, the separation bar can be pulled all the way to the top of the window, so that solely text can be seen. Figure 2-10 shows this.

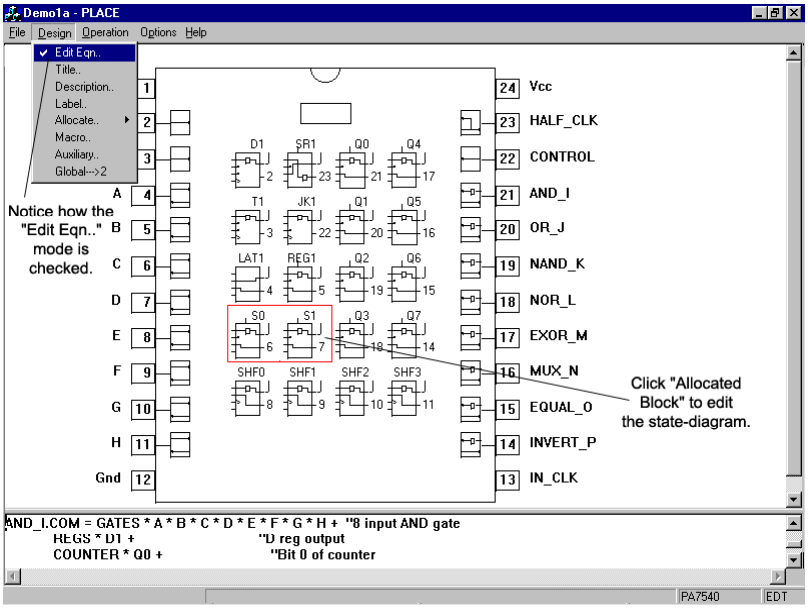


Figure 2-9 Selecting equations from the block diagram screen

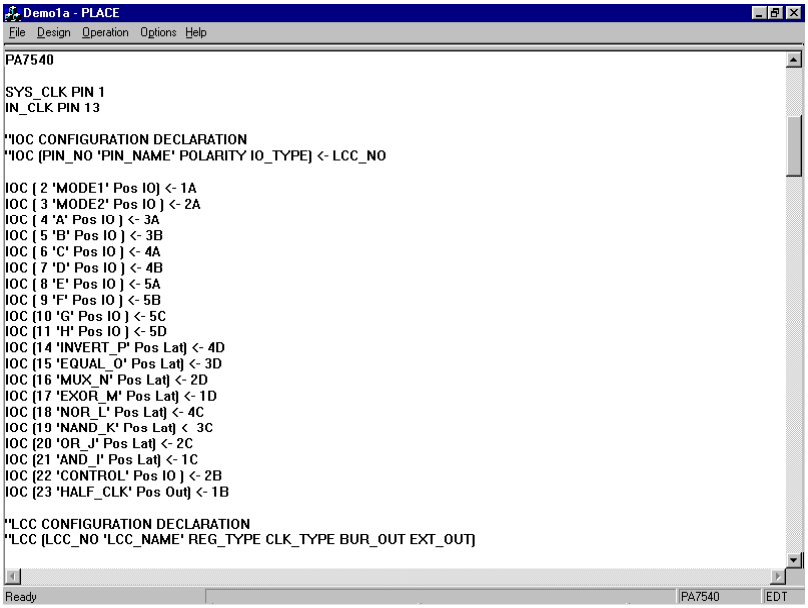


Figure 2-10 Inside the WinPLACE™ text editor

If you have followed the instructions up to this point you have now familiarized yourself with the basic functions of the Design operation in the WinPLACE™ software. Now, move the cursor to the "File" menu option and click the "Save As" command. The file window will appear (Figure 2-11). Move the cursor to the box named "File name:" and click the mouse button. Type in the name "TEMP", or any other new name, to save your modified file. If the file extension is omitted, then it will be defaulted to ".psf". You can change the folder where you want the file to be saved. To jump up multiple folders, click on the pull-down menu that has the name of the current folder. Once the menu is down, locate the folder you want and click on it. The contents of that folder will now be displayed in the big box. If you want to enter a folder in the big box, simply double-click on it. Once the correct location has been found, click on the button labeled “Save”.

Once your Design file is saved, select the “File” pop-down window again and read in the "GATES1.PSF" design example. Move the mouse cursor to the Operations pop-down window and try selecting the other operations starting with Compile, Simulate and back to Design. **Note that you cannot open the Simulate Operation without compiling your design file. To compile, simply go to the “Compile” pop-down window and select “Run.” Now the Simulate option can be entered without error.** The screens should look as displayed in Figure 2-12 and Figure 2-13. For more information on the commands and functions for the four main operations, please refer to the "Operation Reference Guide" in Chapter 3.

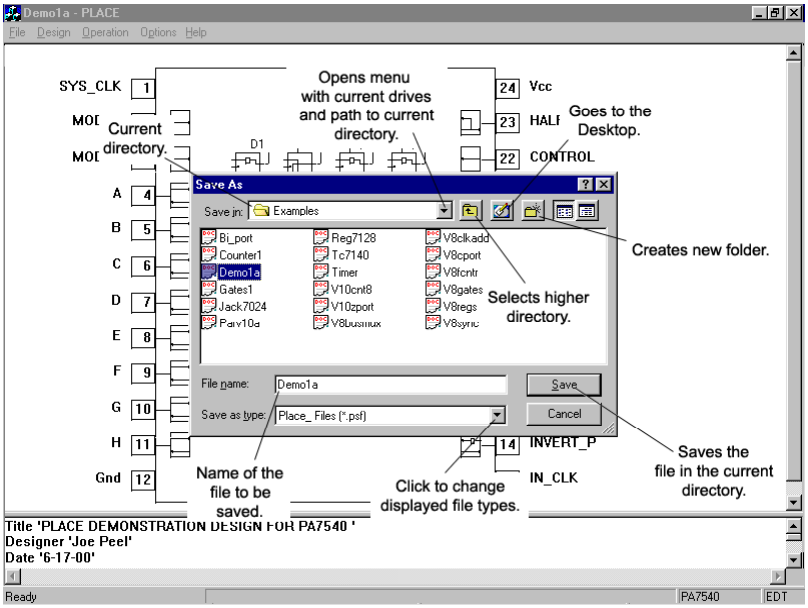


Figure 2-11 Using "Save As" to save a new WinPLACE™ source file

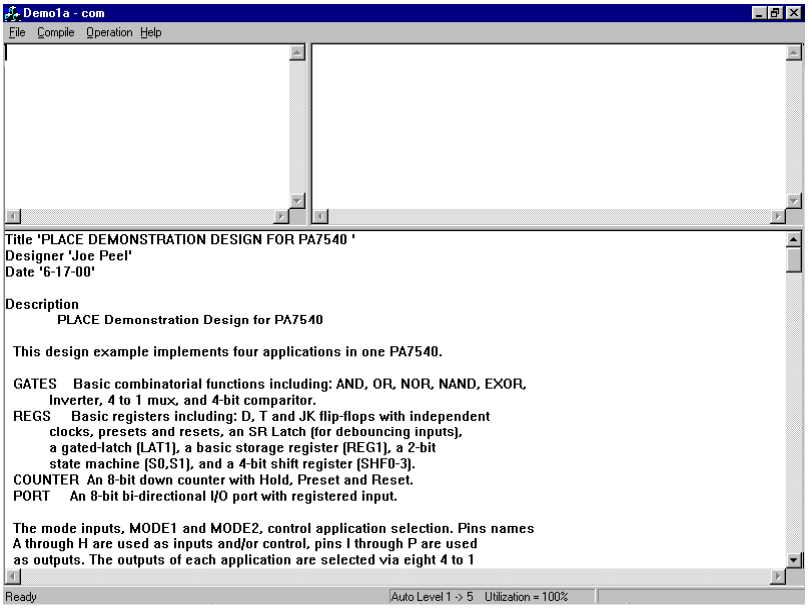


Figure 2-12 The Compile Operation Screen (Standard version only)

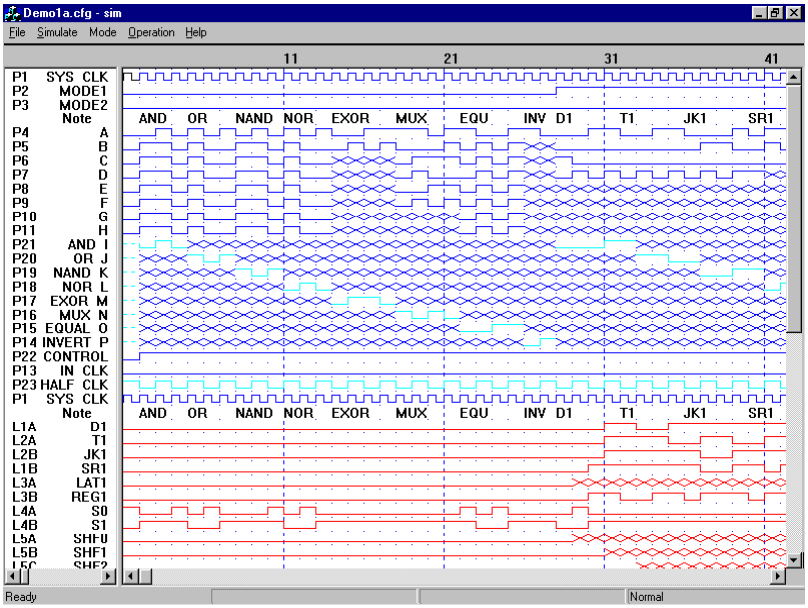


Figure 2-13 The Simulate Operation Main Screen

2.5 The WinPLACE™ Design Process

New designs can be started by selecting the "New" command in the "File" menu option while in the Design Operation. This will display a slide window containing all of the devices that WinPLACE™ can edit. Once a device type is selected it loads the ANEW file which clears all of the pin and cell (INC, IOC and LCC) names. Note that when a cell is cleared, it will be set to its default configuration. Once this is done a new design can be entered. The following lists the typical procedure for implementing a design using WinPLACE™. Please use this as a "road map" for implementing your WinPLACE™ designs while referring to the Operation and Language Reference Guides in Chapter 3 and Chapter 0 of this manual.

Design

1. Select "New" from the "File" menu option.

2. Enter "Title" and "Description" from the "Design" menu option.

3. Label all the pins (IOC' and INC's) and LCC (node) names to be used in the design with the "Label" command.

4. Configure the architectures of the INC's, IOC's and LCC's for the design. Use the "Copy" command to copy or duplicate the INC, IOC or LCC configurations if needed. Use the "Swap" command to move pins, INC's, IOC's and LCC's for desired positioning.

5. For state-diagrams and truth-tables, use the "Allocate" command to allocate the labeled IOC's and/or LCC's for the state machine and truth table designs.

6. For Boolean equation design entry, select the Edit Equation ("Edit Eqn") mode and enter the equations for each LCC input. The equation entry can be done via the pin block or LCC and IOC screen.

7. Edit or modify the architectures and equations until ready to compile. Make sure to save the design through the "File" menu option or by pressing (Ctrl-S).

8. If desired, use the "Swap" command again to reposition the pin and cell locations.

9. Pull down the "Operation" menu and select the Compile operation.

Throughout the design process, it is a good practice to periodically save your design. You can do this with the "Save" function in the "File" pop-down menu or hold the [Ctrl] key down and press "S".

Compile

1. If coming directly from the Design operation (i.e., with the design file loaded), select the "Compile" command menu and click the "Run" command. If the design file was not loaded prior to entering the Compile operation, then "Read" the appropriate source (.PSF) file from the "File" menu window.

2. If a compile error occurs, the compiler will indicate the error with a message and locate the error in the displayed source file. You may analyze the error and correct it with-in the compile operation (by clicking the "Editor" title bar to enter the editor). You may also return to the Design operation to correct the error. If the compilation is successful, proceed to the Simulate operation.

Simulate

1. Enter the input pin waveforms using the "Edit" command.

2. Enter expected output waveforms for test verification or use the "Capture" command in the "Simulate" menu window to automatically generate the output waveforms.

3. After Simulation click on the "Status" command to check if there are any simulation failures. Correct all simulation failures either by changing the vectors, or by returning to the Design operation and modifying the design.

4. Once properly simulated, append the vectors to the JEDEC file using the "Append test vectors" command from the "Simulate" menu window.

5. Save the ".CFG" simulation file using the "File" menu window. ICT recommends saving your first simulation vector file with the ".CFG" file extension. Any vector file can be saved with the extension ".CFx", where x is an alphanumeric character.

3 Operation Reference Guide

In this section, the features of the WinPLACE™ software are discussed in more detail. Sections 3.1 to 3.14 explain the details of the Design Operation. It also explains the main menus of the WinPLACE™ software. Sections 3.15 to 3.16 explain the Compiler Operation and the menus different from those in the Design Operation. Sections 3.17 to 3.23 explain the Simulate Operation and lastly Section 3.24 explains the Text Editor, which does not change throughout the Operations.

3.1 File Menu

The "File" menu option shown in Figure 3-1 provides options for file maintenance, printing, opening the four most recently accessed files, and exiting the program. This file menu is similar in all operations (e.g., Design, Compile and Simulate operations) with the exception of the type of file (see Table 3-1 for WinPLACE™ file types) that is read or saved.

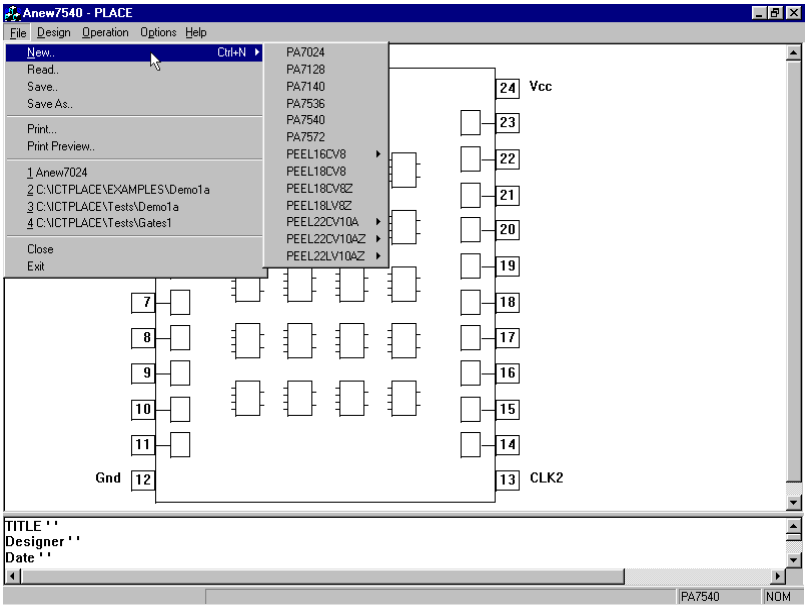


Figure 3-1 File Menu Option

New..... When in the Design operation, "New" clears the file in memory and allows device selection for starting a new design. It loads "ANEWxxxx.PSF" (xxxx = device number) as a template file. The WinPLACE™ software automatically prevents your edited ANEW file from overwriting the original file by prompting for a new file name during the "Save" command. When highlighted, a new menu pans out to reveal the list of devices available in the WinPLACE™ software (Figure 3-1).

Read..... Allows a PSF source, simulation, documentation or JEDEC file to be loaded from the current directory in the current operation (e.g. Loads PSF file in Design operation, CFG in

Simulate operation, etc. -- refer to Table 3-1). When selected, the File selection window will appear (Figure 3-2):

-- To read a file, simply click on the desired file name and then click the "Open" button or press the [Enter] key. For quick selection, you can double-click on the file. The first click highlights it and the second click activates it.

-- A new drive is selected by clicking on the down arrow next to the current directory. A pull-down menu will appear revealing all of the current drives and the path to the current directory.

-- A new directory is selected by doing the same as the step above or by clicking on the button next to pull-down menu. By clicking on that button, you will be brought to the directory directly above the current one.

-- Clicking in the "File name" text box and typing in a new name can change the file name. Use the [Backspace] key, the cursor keys, or the mouse to move the text cursor. Press the [Enter] key or click the mouse on the "Open" button when completed. The default viewing extension is ".psf". In other words, the only files you can view are ones with the extension of ".psf". To change this, click on the down arrow next to extension. A pull-down menu will appear containing the other optional extensions to view. After the file is read, the name will appear in the upper left corner of the program menu bar.

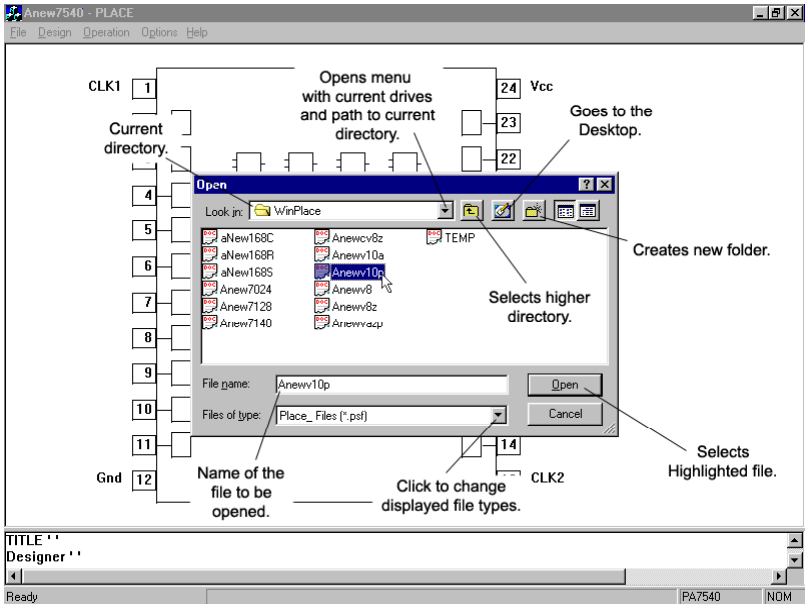


Figure 3-2 "Read" file selection screen

Save — Saves the current file to disk. If the file is a new design (i.e. an ANEW file), then a new file name will be prompted for.

Save As — Allows a new file to be saved to the current directory and disk. Click an existing file to over-write or write in the "File name" text box to enter a new file name (the "Save As" file selection screen is similar to that of the "Read" screen in Figure 3-2 with the exception of the "Save" button). **Note that in the Simulate operation, if the root name of the CFG simulation file is not changed, then the file will still reference the same PSF design file. An example is shown below.**

DEMO1A.CFG saved as DEMO1A.CF1 -- both files reference the DEMO1A.PSF design during simulation.
DEMO1A.CFG saved as DEMO1B.CFG -- these files reference DEMO1A.PSF and DEMO1B.PSF design respectively during simulation.

Print — Sends the current screen to the graphic printer. To print the Pin Block Diagram or the Text Editor, simply click in the general area of the one you want before selecting the "Print" option.

Print Preview — Opens a window containing a printing prediction of the document if you were to print at that moment. It contains some options for navigation, printing, and zooming. Click the "Close" button to exit.

The Four File Names — These file names are the four most recently accessed PSF files. You can access them without clicking on the Read function, but by simply clicking on their name. They are labeled, "1, 2, 3, and 4".

Close — Closes a current design file and returns the default cell configurations for the PA7540 device,

Exit — Quits the WinPLACE™ program and returns to Windows.

3.2 Design Menu

The "Design" menu contains commands used for implementing PEEL™ Array and PEEL™ Device designs. Note that some design commands, such as "LCC Re-assign" and "Global ---> 2" commands, are only applicable to PEEL™ Array designs.

File Extension	Function
PSF	WinPLACE™ Source File (PSF) is the design source file used by the Design and Compile operations.
PS	Back-up file for the PSF design file.
MAP	Output file from the WinPLACE™ Compiler. This file provides detailed information on how the design equations are mapped into the JEDEC file. This information may be useful for design debugging.

Table 3-1 WinPLACE™ File Types

RED	Output file from the WinPLACE™ Optimizer (prior to fuse mapping) in the Compile operation. This file contains the reduced or optimized equations in sum-of-products form. It maintains the WinPLACE™ design format so that it can be read into the Design operation for design verification and debug. Note that the unused equations are omitted, so you will get "Equations not found" errors in the Design operation.
JED	Output JEDEC file from the WinPLACE™ Compiler used by the Simulate and Program operations.
JE	Back-up for the JED file.
INT	Output file from the WinPLACE™ Compiler. This file contains the IOC and LCC interconnects which are used for the waveform display in the Simulate operation.
CFG	Primary vector source file for the Simulate operation. This file contains the data for vector simulation and waveform display. The WinPLACE™ Simulator
CF(n)	(Recommended) Alternate vector source file for the Simulate operation. The character "n" can be any alphanumeric character except of course "G". This file extension method is used for the convenience of displaying all the vector files in the directory popup window, i.e., with the file mask *.CF*. Remember that the vector source files with the same root file name reference the same PSF design file. Example the DEMO1A.CFG and DEMO1A.CF1 are vector files for the DEMO1A.PSF design file.

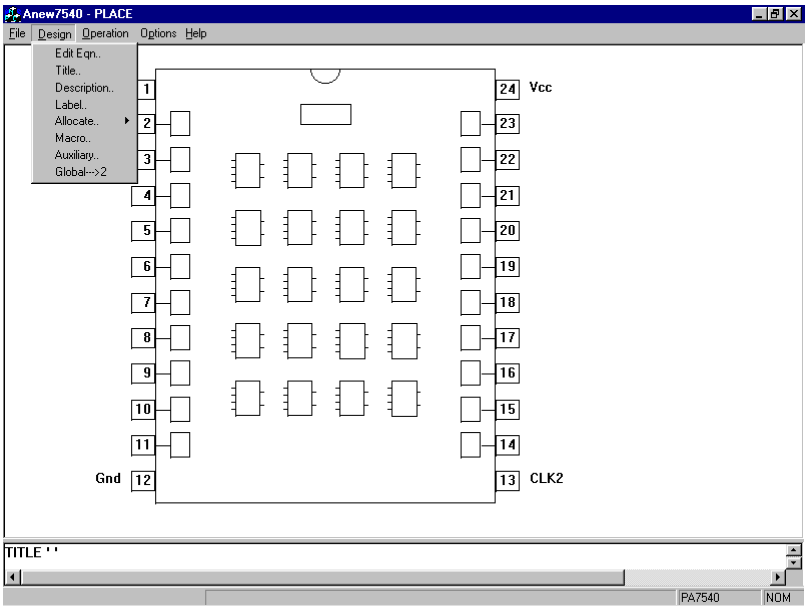


Figure 3-3 Design Menu Option

In the WinPLACE™ software, all commands that are not applicable to the selected device will be disabled.

Edit Eqn…. Selects Edit Equation mode. This mode is identified by a check next to the “Edit Eqn” command line in the “Design” menu. To disable or uncheck this mode, simply re-click it in the “Design” menu. In the "Edit Eqn" mode (also referred to as the "Select Sum Equation" mode), there are three ways of entering a logic description.

1. Equation: Move the cursor to highlight the OR (Sum) gate, and double-click to bring the sum equation out. The cursor in the Text Editor will automatically jump to reveal the selected equation. Note that the equations for each cell are created by the label command. Hence, all cells must be labeled prior to selecting their OR gates.

2. State diagram: Move the cursor into the state diagram box without highlighting any of the OR gates in the LCC selected for the state diagram design. Then, double-click to open the window with the state diagram syntax. The requirement for designing with the state diagram syntax is that the cells (LCC’s and IOC’s) must be allocated using the "Allocate" command.

3. Truth-Table: Double-click at the T(n) labels located at the bottom of the LCC or IOC to open the window for displaying the truth table design syntax. Like the state diagram design, cells used in the truth table design must first be allocated.

After accessing the portion of the Text Editor containing the design syntax, you can edit it freely with using the keyboard. To move the cursor, use your mouse.

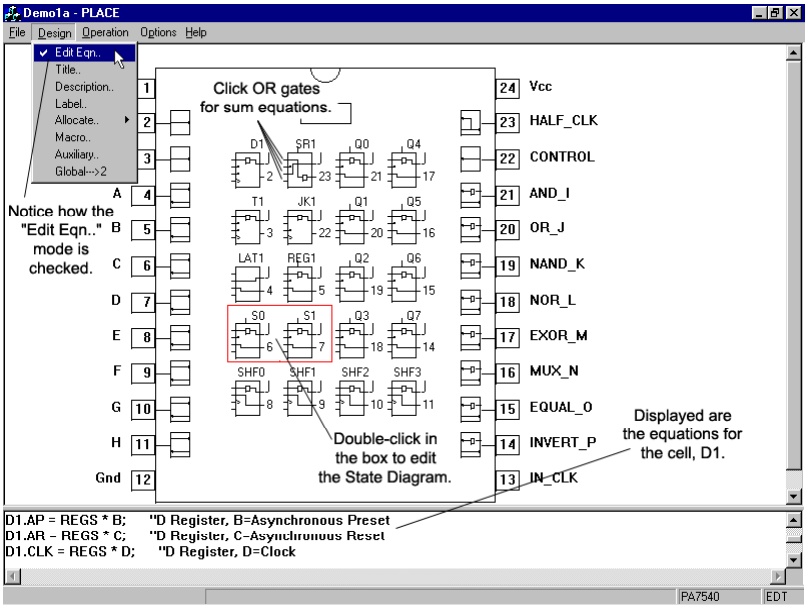


Figure 3-4 Edit Equation Mode

Title ……Automatically moves the text cursor in the Text Editor to the title section. This allows for a title, designer's name, and date to be entered into the source file.

Description Automatically moves the text cursor to the description portion of the source file. Paging can be accomplished by using the scroll bar, the scroll button on a mouse, or the PgUp or PgDn keys. There is no limitation in terms of the type of characters you can use for describing your design, just as long as your description is specified within the reserved words "DESCRIPTION" and "END_DESC;". There is a default text that should already occupy the space for the description, it should read, “Enter description here…” Delete this and put your description where it was. Note: you can use the reserved word "DESCRIPTION" but not the word "END_DESC".

Label……Enters the "Label" mode so that any LCC, IOC, INC or pin can be given a title. After an LCC or IOC is labeled, the equations related to the labeled cell are automatically generated in the source file (all equations equate to 0). The number and type (sum or product equations) of equations generated depends on the device type and the configuration of the cell (Table 3-3). **In PEEL™ Arrays, note that the equations are generated only if both the LCC and its associated IOC are previously unused or unlabeled.**

If the label for the LCC or IOC is deleted and its associated cell (IOC or LCC) is unused, then all the equations are automatically deleted.

Below is an example of the four equations generated for a labeled PA7540 LCC (assume the label is "!TEST").

Example:

Configuration	Equations generated
D-type flip-flop	!TEST.D = 0;
Async. Preset	!TEST.AP = 0
Async. Reset	!TEST.AR = 0;
Assigned IOC is an I/O type	!TEST.OE = 0;

PEEL™ Device	Number of equations per IOC
16CV8	1 sum equation, 1 product equation
18CV8	
18CV8Z, 18LV8Z	
22CV10A, 22CV10AZ	
22LV10AZ	

PEEL™ Array	Number of equations per LCC/IOC pair
PA7024, PA7128	4 sum equations
PA7140	
PA7572, PA7536	
PA7540	

Table 3-2. Number of Equations for IOC or LCC/IOC pair

Labeling Methods

Below are the three methods of labeling cells and pins.

1) **Click and Type method:** This is the normal method of labeling the cells or pins. Move the mouse cursor and click the mouse button to select the cell or pin. Type in the label and press the ENTER key or right button of the mouse to complete the labeling procedure. Repeat this procedure for all cells and pins.

2) **Group (Set) method:** The group or set method is designed for labeling a group of cells or pins with names that differ by a single alphabet character or a set of numbers. Some examples of the group names are: ModeA, ModeB, ModeC; AD0, AD1,..., AD15; and A0, A1,..., A12.

The first step in implementing this method is to click at a cell (IOC, INC, LCC or pin), which will be the starting cell, i.e. the first label in the group, will be assigned to this cell. If the starting cell is an IOC, INC or pin, then the labels will be assigned to the next cell or pin in ascending order. For PEEL™ Arrays, if the starting cell is an LCC, then the labels will be assigned to the subsequent LCC's in ascending order (i.e., 1A, 2A,..., 1B, 2B,..., 5D). After the starting cell has been selected, type the group name in the label window.

Format of the group name is:

prefix name	[A..Z] + suffix name
	[Z..A] +
	[1. . n] +
	[n...1] + (where n > 1)

Some examples are:

Group Name	Assigned Labels
Q[0...3]	Q0, Q1, Q2, Q3
ADDR[9...13]	ADDR9, ADDR10, ADDR10, ADDR12, ADDR13
D[99...102]_IN	D99_IN, D100_IN, D101_IN, D102_IN
OUT[1000...998]	OUT1000, OUT999, OUT998
IN[51...49]DATA	IN51DATA, IN50DATA, IN49DATA, IN48DATA
[A...C]10	A10, B10, C10
[Z....X]1	Z1, !Y1, !X1

3) **Keyboard method:** Like the Group method, the keyboard method also allows labels to be assigned quickly. However, this method is more suitable for assigning labels which are significantly different from each other, i.e. they differ by more than two alphabet characters. Some of the label examples

are INPUT, OUTPUT, ADDRAA, ADDRBB, OE, READ, !WRITE and etc. Normally to assign these labels you would need to implement the "Click and Type" method. But, performing this "click and type" task repeatedly for twenty cells is a very tedious job. The "Keyboard" method shortcuts this task by bypassing the mouse click procedure.

In this method, you should first click on a cell (or pin) that you want the label to start from (defaults to the first pin if no pin or cell is selected). After typing in the label, press the [ENTER] key to implement the assignment (like in the "Click and Type" method). If the [ENTER] key is pressed the second time, the hand mouse cursor automatically advances to the next cell **in ascending order**. You can now type in the label for the current selected cell, and then press the [ENTER] key twice to advance to the next cell. Repeat the procedure until all the cells are labeled. **Please note that if you move your mouse cursor at any time during this mode. The "Keyboard" labeling will be aborted.**

Note that all three methods of labeling cells and pins can be used in conjunction with each other. The "Click and Type" method can be used to select a new starting cell for the "Group" and "Keyboard" methods.

Renaming labels

The Label function can also be used for renaming the pins, IOC's, INC's or LCC's. During the renaming process, the previous labels used in the IOC, INC and LCC configuration, DEFINE, STATE_DIAGRAM, TRUTH_TABLE and EQUATIONS sections will automatically be replaced by the new label. This replacement process allows the user to change the pin or cell labels with ease so that the labels that are used throughout the PSF file need not manually changed. An example of label replacement is:

	Before Change	**After Change**
Pin Label	TEST	/TEST1
Equations	OUT.COM = A & TEST	OUT.COM = A & TEST1
	OUT.OE = /TEST	OUT.OE = /TEST1

As seen n the previous example, the replacement process changes the label and the input signal active level. However, the logic of the equation remains unchanged. For instance, the OUT output, which is controlled by the OUT.OE equation, is enabled on a FALSE or OFF condition. With the TEST input the FALSE condition is a HIGH signal. On the other hand, the FALSE condition for the /TEST1 input is a LOW signal.

Allocate……Assigns the LCC or IOC for state-machine or truth table design. Before the cell can be used for allocation, it must first be labeled. This mode also allows a state-machine or truth table design to be created, deleted or updated. For state-machine design, a border will surround the allocated cells. For truth table design, the allocated cells will be indicated by "in" and "Tn" (inputs and outputs respectively), where n ranges from 1 to 10. See Sections 3.12 and 3.13 for detailed descriptions of the state diagram and truth table designs.

Macro……. Displays the macro definitions in the Text Editor. Macro definitions are text statements, which succeed the keyword "DEFINE" but precede one of the following reserved words: STATE_DIAGRAM; TRUTH_TABLE; or EQUATIONS.

Auxiliary…. Additional functions, such as Security Bit, Signature and Zero-Power options, which are found in the PEEL™ Arrays and in some PEEL™ Devices. Below is a brief description of each function; refer to the ICT data book for more information.

Security Bit--Setting this feature ON enables the security bit to be programmed on the device (inserts the "G1" field in the JEDEC file). Once the security bit has been programmed, the design programmed into the device cannot be read back (except for the Signature Word). All PEEL™ products provide the security bit feature.

Signature Word--The Signature Word of the device allows a user ID to be stored in the device so that it can be read back for design verification even after the security bit has been programmed. Devices with the Signature Word (number of 8-bit bytes in parenthesis) are PA7024 (8 bytes), PA7540 (8bytes), PA7140 (2 bytes), PA7572 (2 bytes), PA7128 (1 byte), PA7536 (1 byte), PEEL™22CV10A+ (3 bytes) and PEEL™22CV10A++ (8 bytes).

Example Signature = ABC [ENTER]

(Converts the characters A, B, C to the ASCII values 65,66 and 67 respectively. Each character requires an 8-bit byte.)

Clock Polarity -- For PEEL™22CV10A++ only, this feature allows clock polarity to be set. The default setting is "NON-INVERTED" which results in positive-edge clocking.

Setting the polarity to "INVERTED" will result in negative-edge clocking.

Clock Select -- For the PEEL™22CV10A++ only, this feature allows global clock selection between "PIN 1" (external) and "P-TERM" (internal product-term) clocks.

Global……..Toggles between the single and double global cell modes (having one or two). If two global cells are used, swapping is not allowed with IOC's that are located on the left and right sides of the device, or with LCC's which are associated with IOC's that are located on the left and right sides of the device. This feature allows the device to be separated into two parts with each part containing its own high-speed clock. Please refer to the ICT data book for more information on the benefits of the one and two global cell modes.

3.3 Operation Menu

When an operation is selected from the "Operation" menu (Figure 3-5), the WinPLACE™ software automatically loads the proper input file for the

selected operation. For instance, the PSF file is loaded for the Design and Compile operations, and the CFG file for the Simulate operation.

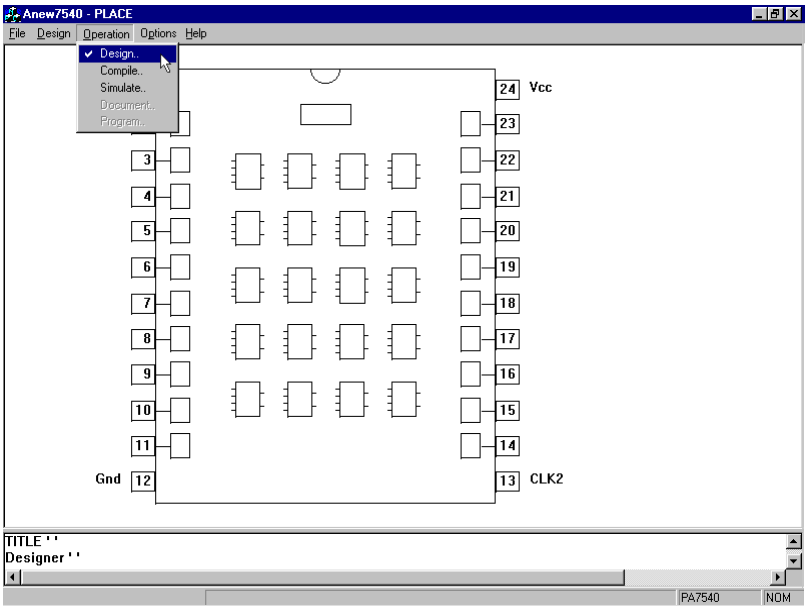


Figure 3-5 Operation Menu Option

Design…….Selects the Design operation for creating a PSF design source file. The PSF file (if it exists) will automatically be loaded.

Compile…...Selects the Compile operation for compiling the PSF design source file to create a JED (JEDEC) file for simulation and programming use. The PSF file (if it exists) will automatically be loaded upon entering the Compile operation.

Simulate…..Selects the Simulate operation for creating a CFG vector source file, which can be edited and simulated to the JED file. The CFG file (if it exists) will automatically be loaded.

3.4 Options Menu

Set Pinout to DIP/PLCC…..Allows the pin numbers in the pin block diagram and LCC/IOC screen to show the pinout of DIP or PLCC package type. Figure 3-7 and Figure 3-8 show the pinouts for DIP and PLCC packages respectively. **The default is the DIP pinout for all devices with the exception of the PA7140 device, which defaults to the PLCC pinout configuration.** The pinout for the SOIC package is the same as the DIP package.

PLCC Configuration..Displays the design in the actual PLCC package form (Figure 3-9). Press the [Esc] key or click-R to return to the previous screen.

Translators…

JEDEC-to-JEDEC…See Section 1.4 for details.

APEEL-to-PSF…….Also See Section 1.4 for more information.

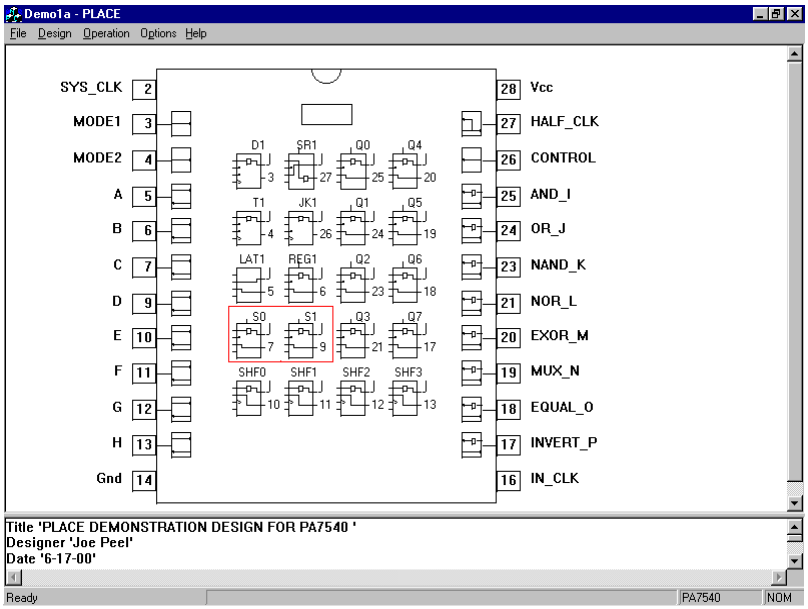


Figure 3-6 Utilities Menu Option

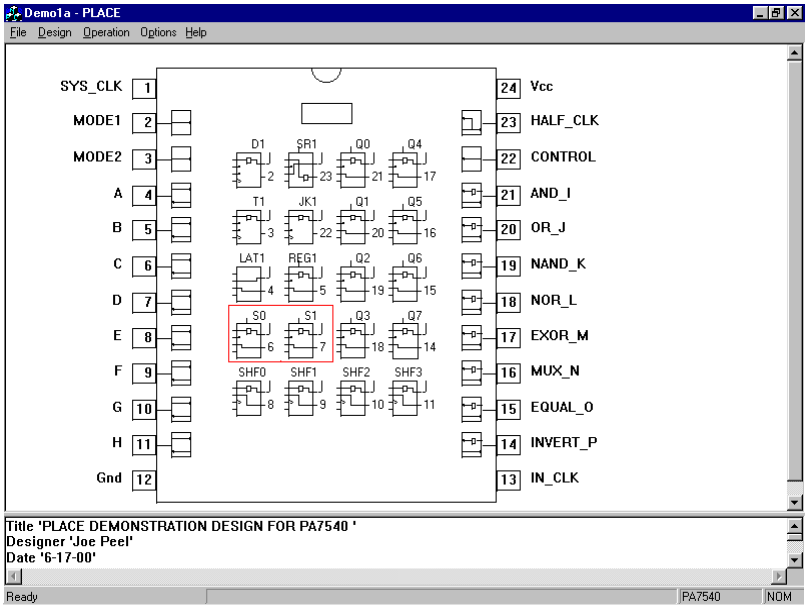


Figure 3-7 The DIP pinout for the PA7540

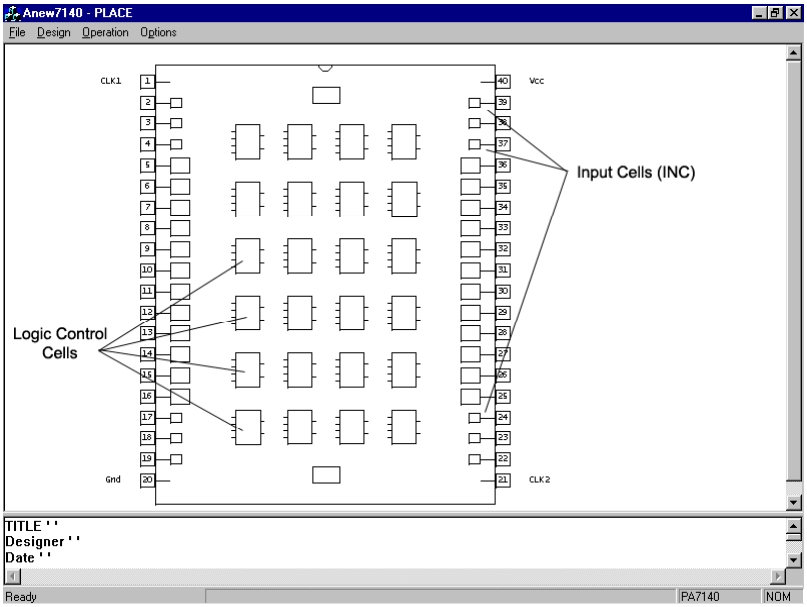


Figure 3-8 PLCC pinout for the PA7540

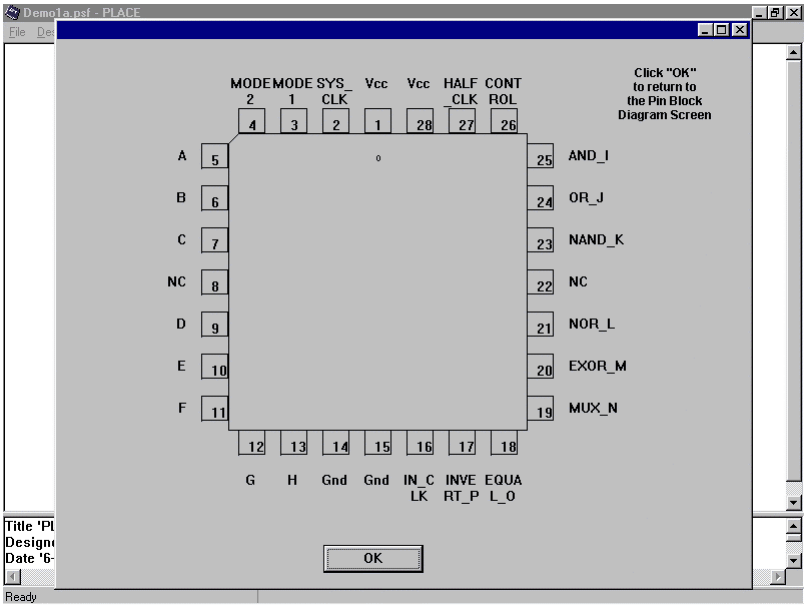


Figure 3-9 PLCC package configuration for the PA7540

3.5 Help Menu

WinPLACE™ Help….Provides information about the operation of WinPLACE™.

About place_w5……..Provides version number, copyright date, and official WinPLACE™ icon.

3.6 Design Operation -- Pin Block Diagram Right-Click Options

Copy…

From….Copies the selected LCC's, IOC's or INC's configuration into the computer's clipboard, but does not copy the equations. Right click over one of the cells and select the "From" option under "Copy" (Figure 3-10). The only restriction in the "Copy" mode is that the "From" and "To" cells must be of the same type, i.e., LCC-to-LCC and IOC-to-IOC.

To…….This function pastes the most recently copied LCC, IOC, or INC configuration over the selected cell. This function can be used multiple times, without re-copying the original cell.

Swap…This mode allows switching of LCC's, IOC's, INC's or pins to reorganize your design. The functions of the cells do not change. There are two restrictions when swapping the two cells.

1. Both the source and target cells must be of the same type, i.e., LCC-to-LCC, IOC-to-IOC, and INC-to-INC.
2. For PEEL™ Arrays, the swapping of IOC's or LCC's are not permitted if the double global cell mode is used and if one of the following is true:

   a) The cells that are being swapped are IOC's located on the opposite sides of the pin block diagram.

   b) The cells that are being swapped are LCC's connected to IOC's that are located on the opposite sides of the diagram.

Clear……Clears the LCC, IOC, or INC user-entered configurations. This sets a default configuration to the cells and renames the sum outputs according to the default. All of the sum equations associated with this cell are deleted as well.

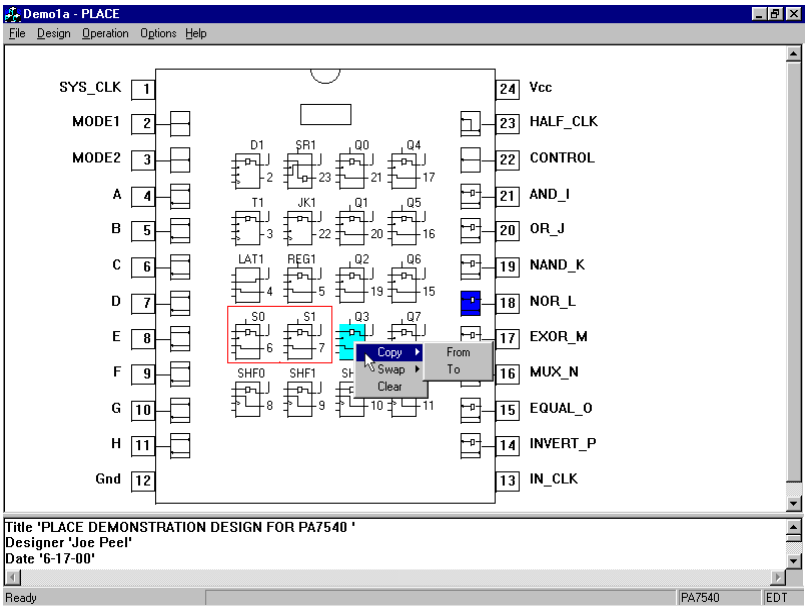


Figure 3-10 The Right-Click Options

3.7 Design Operation – Pin Block Diagram Screen

The default function for the pin block diagram screen is the Edit Architecture mode (Edit Arch). This mode is automatically set when the WinPLACE™ software is initially entered. It allows selection of the Logic Control Cells (LCC's), I/O Cells (IOC's), Input Cells (INC's) or Global Cells (GBC's) for controlling cell configurations.

The PA7140 pin block diagram shown in Figure 3-11 illustrates the DIP pinout configuration, which is the default for this device. Other devices, such as the PEEL™22CV10A shown in Figure 3-12, also default to the DIP configuration. The PLCC configuration can be set in the "Options" menu, refer to Section 3.4 for more information.

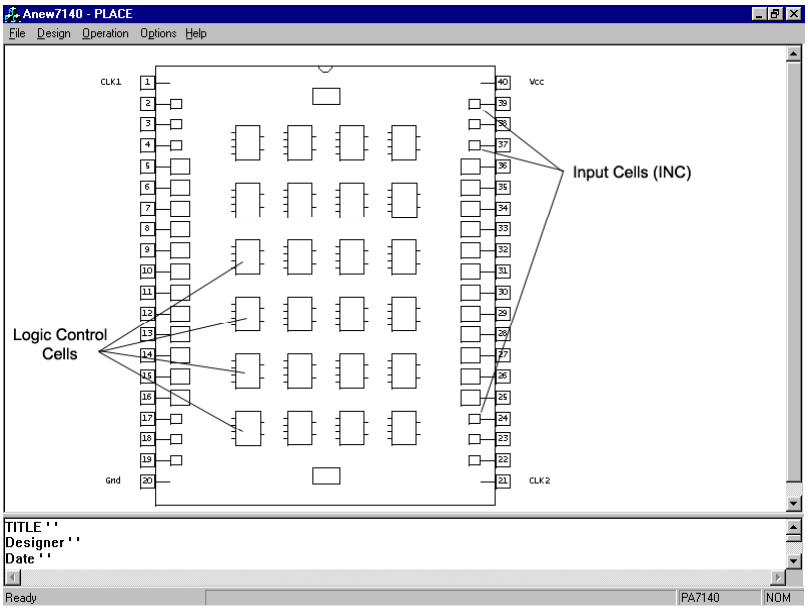


Figure 3-11 Pin Block Diagram of the PA7140

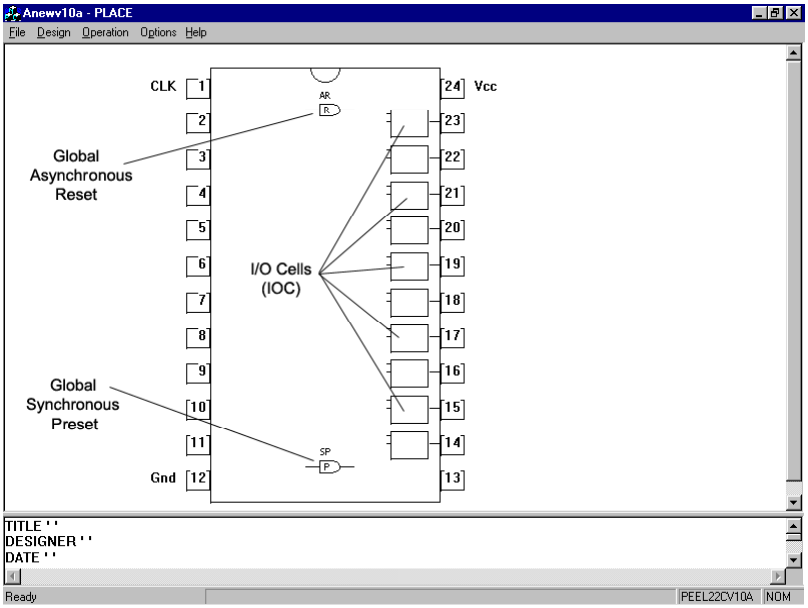


Figure 3-12 Pin Block Diagram of the PEEL™22CV10A

3.8 Design Operation – LCC and IOC Screen

If a Logic Control Cell (LCC) or I/O Cell (IOC) is selected from the pin block diagram screen, the screen zooms into the selected cell for a close-up view of the cell configurations (Figure 3-10). Note that both the LCC and its currently connected IOC are displayed. In this screen, selections can be accomplished by moving the mouse cursor to the "selectors" such as the register's "rectangle", polarity "bubble", OR gates, or the "@ symbol" selectors. Except for the OR gate, all of the above selectors are used for controlling the cell configurations.

Selecting any of the four OR gates will display its associated equation in the Text Editor, located at the bottom of the screen. The cursor will automatically move itself to the equation related with that OR gate.

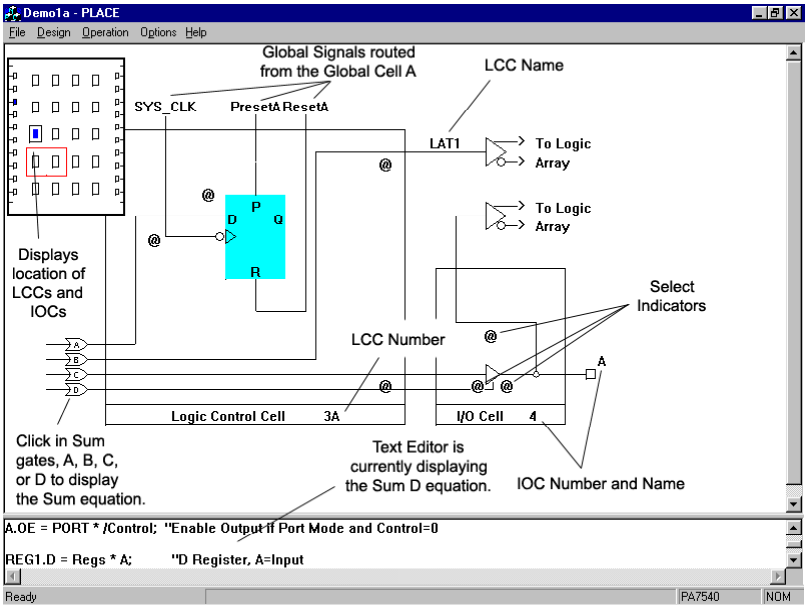


Figure 3-13 LCC and IOC screen of the PA7540

Select Indicators: PEEL™ Arrays (7024, 7540, 7128, 7536, 7140, and 7572)

Below are the descriptions of the "select indicators" found in all PEEL™ Arrays. Unless specified otherwise, all "select indicators" are applicable for all the PEEL™ Array devices.

Clk (Clock) Select

/Global Inverted clock signal from the Global Cell A or B. If the two global cell mode is used, then LCC's connected to IOC's on the left and right sides of the pin block diagram are control led by Global Cells A and B respectively. For one global cell mode, all global signals are routed from Global Cell A.

Global Default configuration. Non-inverted clock signal from the Global Cell A or B.

Sum-C Clock signal from the Sum-C gate. The Reset signal for the LCC register will automatically be routed from the Global Cell A or B if it was originally connected to the Sum-C gate.

Sum-D Clock signal from the Sum-D gate. If the IOC connected to the LCC is an I/O type, then the output will be disabled. If the IOC is an output type, then it will remain as an output type with only combinatorial feedback from the pin.

RT (Register Off Type) Control

Off **Default configuration.** Sets the dynamic register control to Off. This means that any signal on the RT line (RTA or RTB equation in the PSF file) will not implement a dynamic register change.

On Sets the dynamic register control to On. TRUE logic on the RTA or RTB equation (in the PSF file) changes the type of register in the LCC during normal operation. For instance, if RT is "On" and Register Type is "D -> T", then the D-type register will be changed to a T-type register when the logic on the RTA equation is TRUE. For the one global cell mode, the RTA equation controls the RT signals in all LCC's. For the double global cell mode, the RTA and RTB equations control the RT signals of the LCC's, which are connected to IOC's on the left and right sides of the pin block diagram respectively.

Register Type

If RT is Off:

D **Default configuration.** D-type register. Reset and preset of the register can be locally (through the sum B or C gate) or globally (ResetA or PresetA equation) controlled.

T T-type register. Reset and preset of the register can be locally (through the sum B or C gate) or globally (ResetA or PresetA equation) controlled.

JK JK register. Sum-A for the J-input, and Sum-B for the K-input. Reset can be controlled locally or globally, but the preset will automatically be set to global preset.

If RT is On:

D -> T **Default configuration.** D-type register when the RT signal is FALSE, and T-type register when it is TRUE. Reset and preset can be locally or globally controlled.

D -> JK D-type register when the RT signal is FALSE, and JK-type register when it is TRUE. For the JK-type register, Sum-A is used for the J-input, and Sum-B for the K-input. Reset can be controlled locally or globally, but the preset will automatically be set to global preset.

T -> D T-type register when the RT signal is FALSE, and D-type register when it is TRUE. Reset and preset can be locally or globally controlled.

JK -> D JK-type register when the RT signal is FALSE, and D-type register when it is TRUE. For the JK-type register, Sum-A is used for the J-input, and Sum-B for the K-input. Reset can be controlled locally or globally, but the preset will automatically be set to global preset.

Note that the Sum-B gate cannot be used for both K-input and preset for the register.

Preset and Reset for the LCC Register

The output of the LCC register is set to a HIGH signal when the preset signal is TRUE. On the other hand, the output of the register goes LOW if the reset signal is TRUE. **If both the preset and reset signals are TRUE, then the preset signal takes precedence over the reset signal.**

There is no dedicated MUX for controlling the preset or reset of the LCC register. Both of these signals are indirectly controlled by the "Clk Select", "Register Type", "Buried Output" and "Ext Output" selections. **The same Sum (OR) gate allocated to any of the above configurations cannot be used for presetting or resetting the LCC register.** So, the WinPLACE™ software automatically switches the preset or reset to the global signal when the local sum gate is used.

Reg-Q **Default configuration.** Connects the output of the LCC register to the internal or buried output of the LCC.

Sum-A Connects the Sum-A gate to the internal or buried output of the LCC.

Sum-B Connects the Sum-B gate to the internal or buried output of the LCC. If the preset of the register is locally controlled (through Sum-B), it will automatically be set to global preset.

Sum-C Connects the Sum-C gate to the internal or buried output of the LCC. If the reset of the register is locally controlled (through Sum-C), it will automatically be set to global preset.

Buried Output (Internal Output of the LCC)

Ext Output (External Output of the LCC to the IOC)

Reg-Q **Default configuration.** Connects the output of the LCC register to the internal or buried output of the LCC.

Sum-A Connects the Sum-A gate to the external output of the LCC.

Sum-B Connects the Sum-B gate to the external output of the LCC. If the preset of the register is locally controlled (through Sum-B), it will automatically be set to global preset.

Sum-C Connects the Sum-C gate to the external output of the LCC. If the reset of the register is locally controlled (through SumC), it will automatically be set to global preset.

OE (Output Enable) Select

I/O **Default configuration.** Sets the IOC to I/O type. Sum-D is used for the output enable control. If the LCC Clk Select is set to Sum-D, the IOC changes from I/O to input type automatically.

Input Sets the IOC to input type. If Sum-D is not used for the LCC clock, then it will be disabled.

Output Sets the IOC to output type. If Sum-D is used for the LCC clock, then the Feedback Type will automatically be set to combinatorial.

Feedback Type

In the normal configuration, whether the IOC is an I/O, input or output type, this multiplexer controls the path from the pin. However, in the PA7536, PA7572, PA7128 and PA7140 devices, the option "FB Mux" allows the path to come from the Sum-D gate.

Com **Default configuration.** Combinatorial path from the pin or Sum-D.

Reg Registered path from the pin or Sum-D. The clock for the register can be directly from the CLK1 or CLK2 pin, or PCLK product term.

Lat Latched path from the pin or Sum-D. The latch trigger can come directly from the CLK1 or CLK2 pin, or PCLK product term.

Out (Output) Polarity

Invert Inverts the output to the pin for active Low output.

Non-invert **Default configuration.** Non-inverted output for active High output.

FB Mux (PA7128, PA7536, PA7140, and PA7572 only)

Pin **Default configuration.** Sets the feedback path to come from the pin.

Sum-D Sets the feedback path to come from the Sum-D gate. With this configuration, the IOC automatically becomes an output pin with no feedback from the pin. Therefore, the Sum-D signal will be buried. See Figure 3-14.

Select Indicators in the PEEL™ Devices

Output Select Com

Com **Default configuration.** Sets the IOC to combinatorial output. In the 22CV10A devices, the feedback path is automatically set to come from the pin with this configuration. For other devices such as PEEL™18CV8, the feedback path is control led by the "Feedback Type" MUX.

Reg Sets the IOC to registered output. In the 22CV10A devices, the feedback path is automatically set to come from the Q of the register with this configuration. See Figure 3-15. The register is triggered on the rising-edge of the clock.

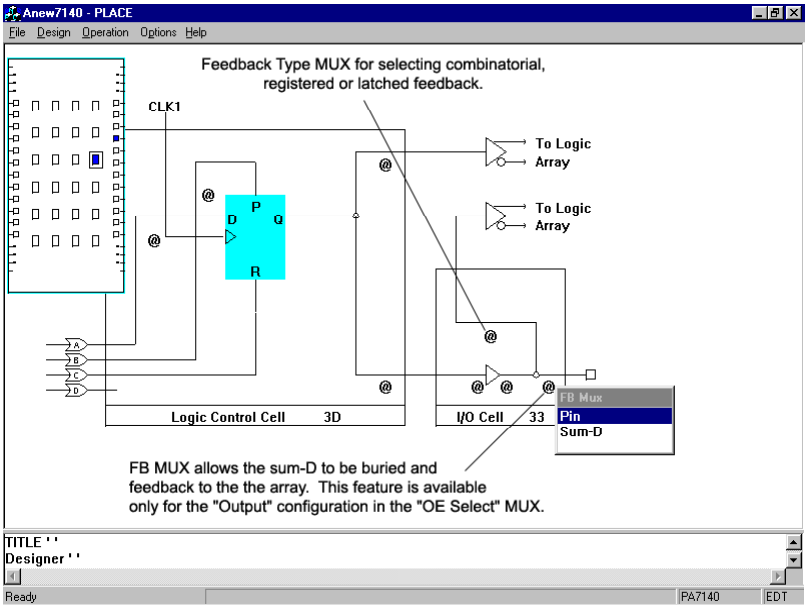


Figure 3-14 LCC and IOC screen of the PA7140

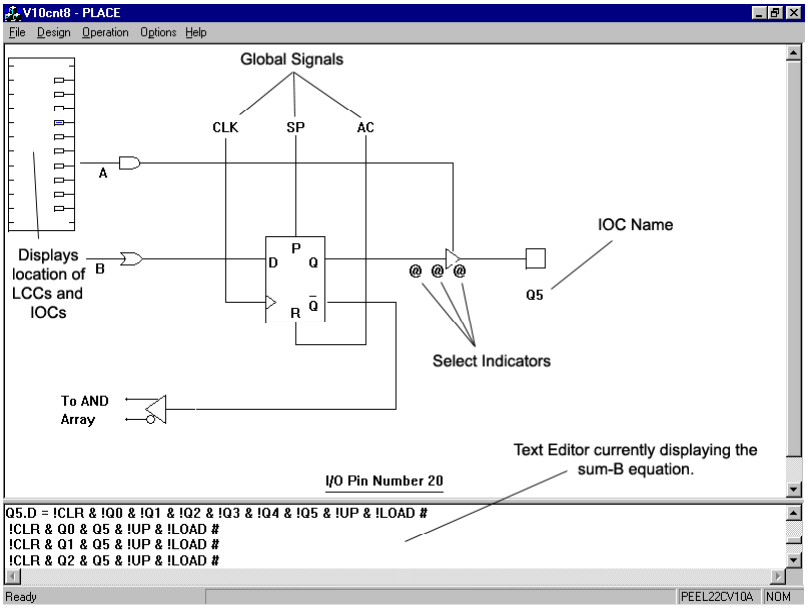


Figure 3-15 IOC screen of the PEEL™22CV10A

OE (Output Enable) Select

I/O — **Default configuration.** Sets the IOC to I/O type. Depending on the device, a sum or product term controls the output enable term.

Output Enabled — Enables the output in the IOC.

Output Disabled — Disables the output in the IOC.

Out (Output) Polarity

Invert — **Default configuration.** Inverts the output to get an active Low output.

Non-invert — Buffers the active High output.

Feedback Select (18CV8, 18CV8Z, 18LV8Z, 22CV10A+ and 22CV10A++)

Pin — Feedback path from the pin.

Reg — Feedback path from the Q (PEEL™18CV8) or Qn (all other devices) of the register.

Or — Feedback from the OR gate, i.e., prior to the register and output buffer.

Global Asynchronous Reset and Synchronous Preset in PEEL™ Devices

In all registered PEEL™ devices, the IOC (or macrocell) register can be reset or preset using an internal reset or preset product term. The WinPLACE™ software automatically assigns node numbers for both product terms.

When the reset product term is TRUE, the output of the register in the IOC (or macrocell) is set to a LOW signal asynchronously. **For 20-pin devices, the asynchronous reset node number is 21, and for 24-pin devices the node number is 25.**

When the preset product term is TRUE, the output of the register in the IOC changes to a HIGH signal (synchronously) with the rising-edge of the clock signal. **The node number for the preset product term is 22 and 26 for the 20 and 24-pin devices respectively.**

When both the reset and preset signals are TRUE, then the reset signal takes precedence over the preset signal.

3.9 Design Operation – Input Cell (INC) for PA7128, 7572, 7140, and 7536

In addition to the lOC's and LCC's, the PA7536, PA7572, PA7128 and PA7140 have Input Cells (INC's). Each INC allows the input to be configured as combinatorial, registered or latched input (Figure 3-16).

Input Type

Com	**Default configuration.** Sets the input to be combinatorial.
Reg	Sets the input to be registered. The clock for the register is controlled by the Global Cell C.
Lat	Sets the input to be latched. The trigger for the latch is controlled by the Global Cell C.

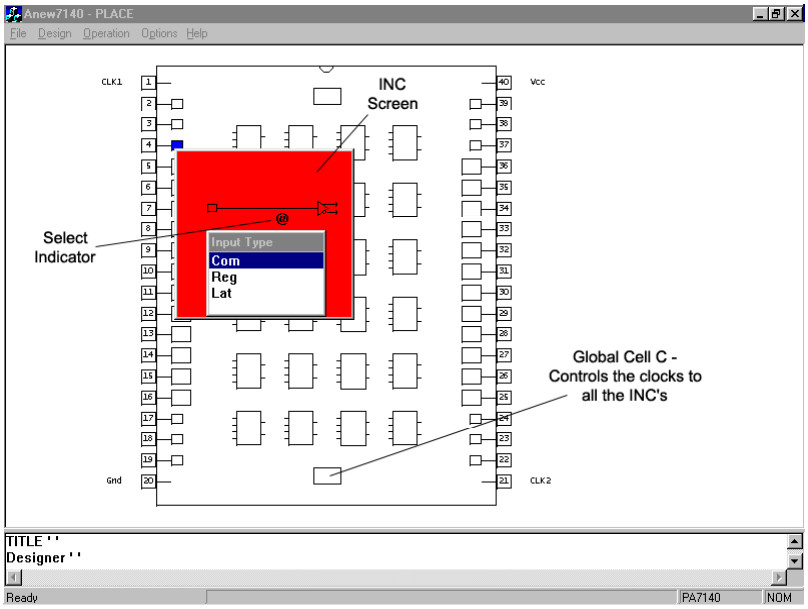


Figure 3-16 INC screen of the PA7140

3.10 Design Operation – Global Cell (GBC) for PEEL™ Arrays

Global Cells A and B

In PEEL™ Arrays, the Global Cells (GBC's) A and B which are located at the top of the pin block diagram are used to control the global signals for the LCC and IOC. See Figure 3-17. These global signals include the clock for the LCC and IOC, and preset, reset and register type control for the LCC.

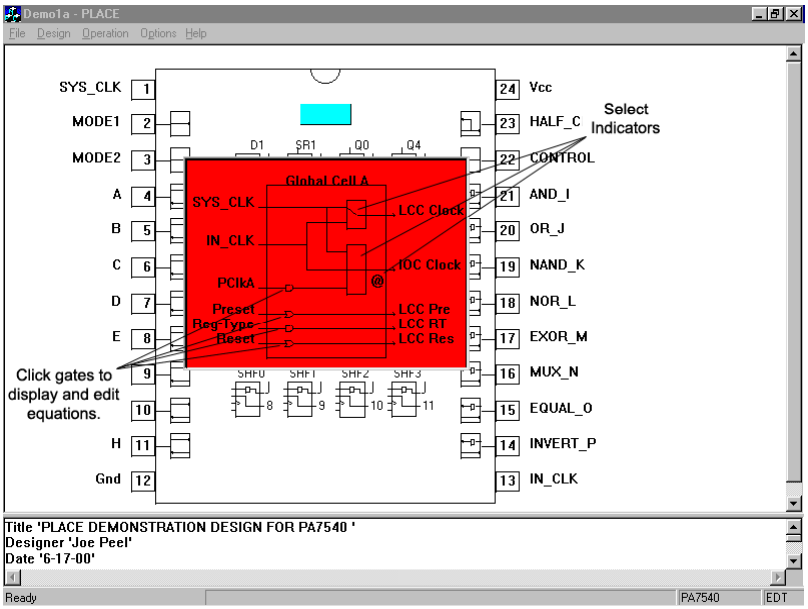


Figure 3-17 Global Cell A screen of the PA7540

After clicking the GBC to bring the Global Cell window up for selection, click at the MUX "rectangle" to set the desired clock selection for the LCC's and IOC's. The "Clock Polarity" select indicator controls the IOC clock polarity.

Click at the AND or OR (Sum) gates to display the global equations in the Text Editor for register-type, PCLK, global reset or preset. The cursor will be flashing next to the equation and you can edit it freely. The same procedure can be used in the single or double global cell mode.

In the single global cell mode, the Global Cell A controls the global signals to all the LCC's and IOC's in the device (Global Cell B is ignored). In the double global cell mode, Global Cell A controls the global signals to the LCC's and its associated IOC's that are located on the left side of the pin block diagram. Global Cell B controls the global signals to the remaining LCC's and IOC's, which are located on the right side of the pin block diagram.

Global Signals

LCC Clock This signal clocks the register in the LCC's. The signal comes from one of the two dedicated clock pins, CLK1 or CLK2.

IOC Clock This signal clocks the register in the IOC's. In addition to the two dedicated clock pins CLK1 and CLK2, the signal can come from a product term PCLKA (GBC A) or PCLKB (GBC B). The select indicator "Clock Polarity" allows the IOC clock polarity to be changed. The default configuration is "Pos" which means that the IOC register or latch is triggered on the rising-edge or HIGH signal. If the configuration is set to "Neg", then the register or latch is triggered on the falling-edge or LOW signal.

LCC Pre This sum (OR) term is the global preset for the LCC register. The LCC register is preset to a HIGH signal when this sum term is TRUE. This term takes precedence over the reset term for the LCC register.

LCC RT This product term is the global register-type change for the LCC. Each LCC has the option ("RT Control") of enabling the dynamic register-type change when this term is TRUE.

LCC Res This sum (OR) term is the global reset for the LCC register. The LCC register is reset to a LOW signal when this sum term is TRUE.

LCC PLD (PA7140 & PA7572 only) This sum (OR) term is the global preload for the LCC register. The LCC registers controlled by this global cell are loaded from the associated IOC pin when this sum term is true and the registers are clocked.

LCC ULD (PA7140 & PA7572 only) This sum (OR) term is the global unload for the LCC register. The content of the LCC registers controlled by the global cell will be unloaded to the IOC pin when true.

Global Cell C (PA7536, PA7572, PA7128 and PA7140 only)

In addition to Global Cells A and B, the PA7128, PA7536, PA7140, and PA7572 have Global Cell C. This cell, which is located at the bottom of the pin block diagram, controls the global clock signal for the Input Cells (INC's). See Figure 3-18.

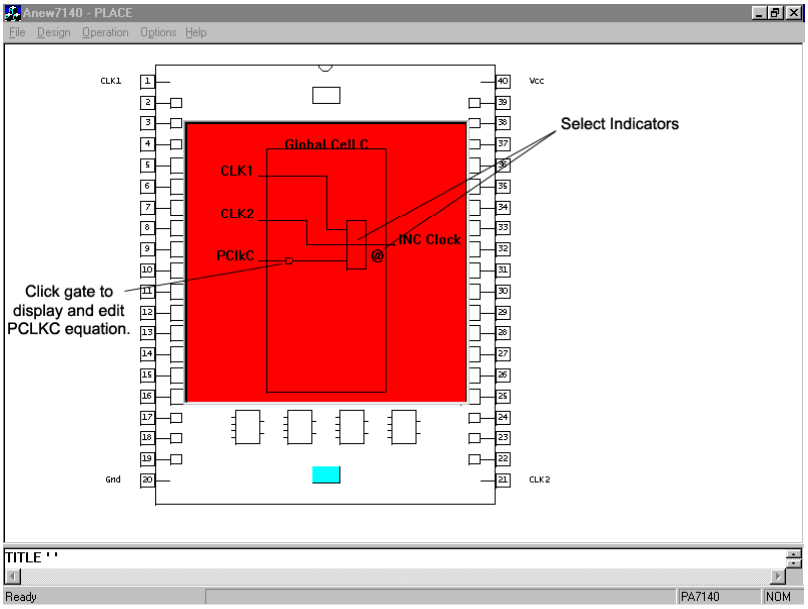


Figure 3-18 Global Cell C screen of the PA7140

3.11 Design Operation – Entering Equations

One of the primary methods of entering the design equations is via the "Edit Equation" mode, which was discussed in Section 3.2. Figure 3-19 and Figure 3-20 show the "Edit Equation" mode for the PA7540 and PEEL™22CV10A devices respectively. In addition to the "Edit Equation" mode, equations can also be edited via the macrocell screen (or just the IOC screen for the PEEL™ devices). Refer to Figure 3-21 and Figure 3-22.

For editing equations for the global signals in the PEEL™ Arrays, please refer to Section 3.10 on "Global Cells for PEEL™ Arrays".

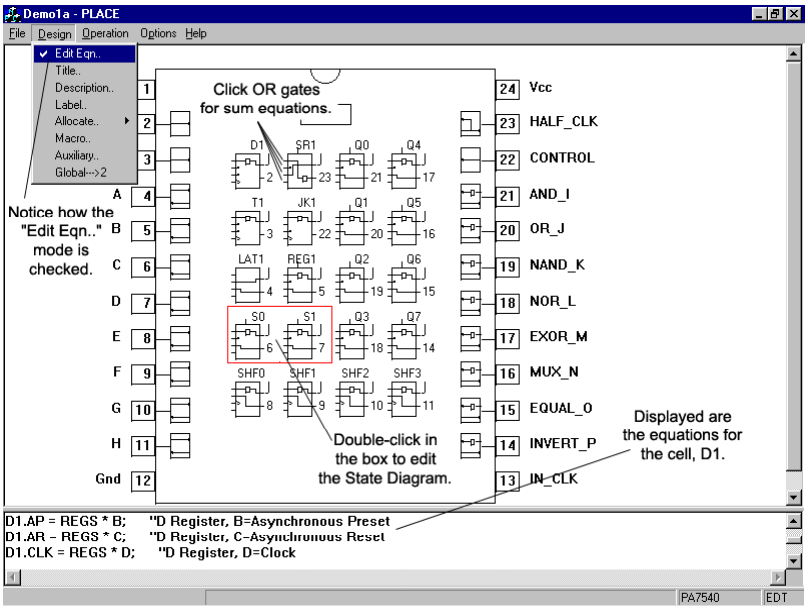


Figure 3-19 "Edit Eqn" mode of the PA7540

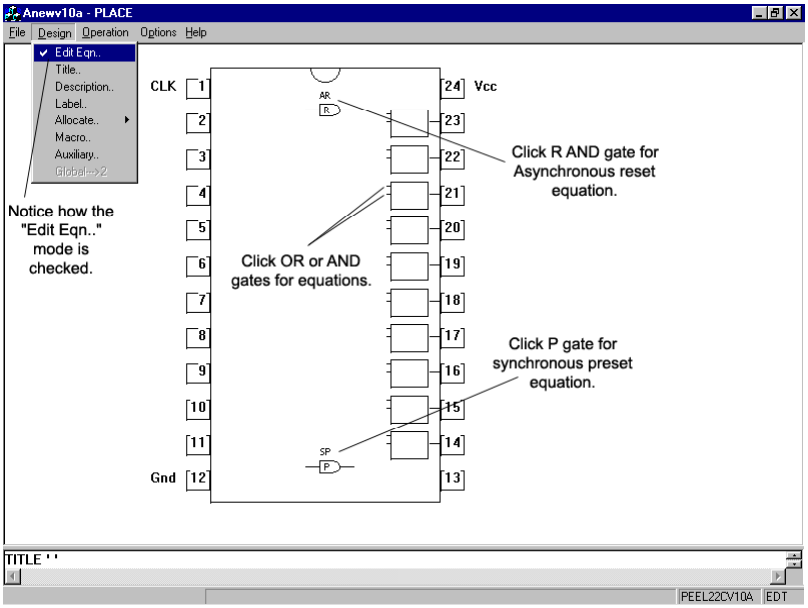


Figure 3-20 "Edit Eqn" mode of the PEEL™22CV10A

After the cell (LCC or IOC) is labeled using the "Label" command in the "Design" menu window, the WinPLACE™ software automatically generates the equations. To edit or modify these equations, click at the desired OR (sum) or AND (product) gate and it will bring the Text Editor cursor to that equation. Whether the device is a PA7540, PEEL™22CV10A or any other device, the procedure for entering the design equations remains the same.

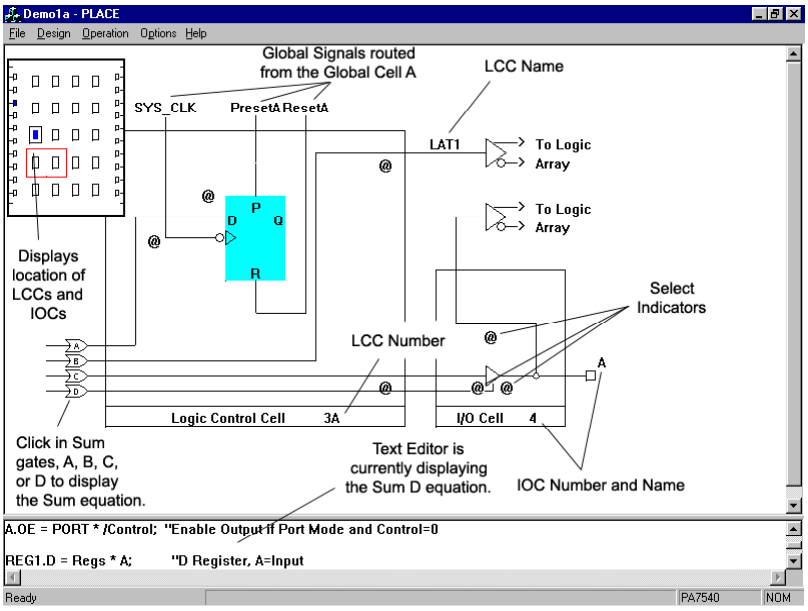


Figure 3-21 Editing equations in the LCC and IOC screen of the PA7540

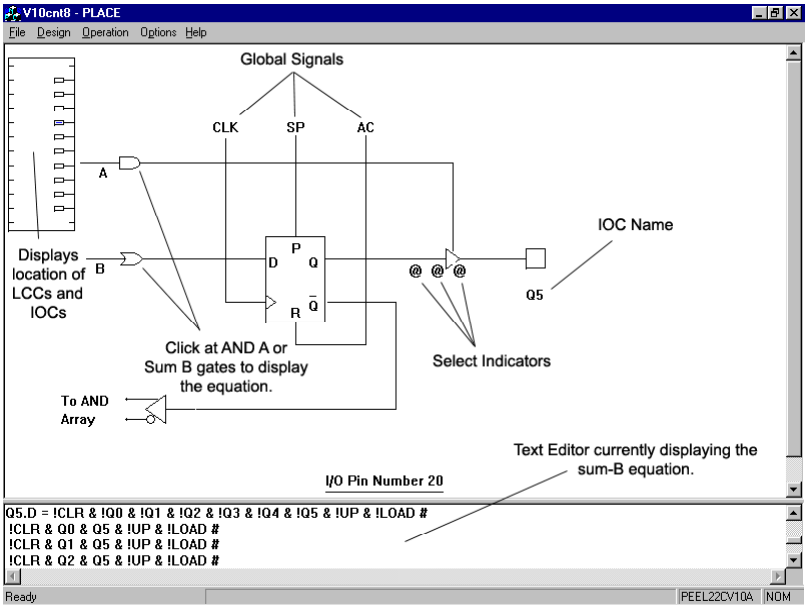


Figure 3-22 Editing equations in the IOC screen of the PEEL™22CV10A

3.12 Design Operation – State Diagram Designs

An alternate method of describing a logic design is the state diagram design implementation. In this section, the procedure to implement the state diagram designs is discussed. The syntax for these state diagram designs is discussed in detail in Chapter 4 on "WinPLACE™ Design Language".

The first step in the state diagram design method is to label the cells to be used as state variables for the state diagram. These cells will be referred to as "State Cells". Then, configure each state cell and use the "Copy" command to duplicate the configurations to other state cells. Next, select the "Allocate" command (Section 3.2) in the "Design" menu window to implement the state cell assignments then choose the "State Diagram" option. A window pops up allowing the options of adding or erasing state diagrams (Figure 3-23).

Note that the PEEL™22CV10A device is used in the example described in this section. The procedure of implementing the state diagram design remains the same for all devices. In addition to the IOC's, LCC's in the PEEL™ Arrays can also be allocated as state cells.

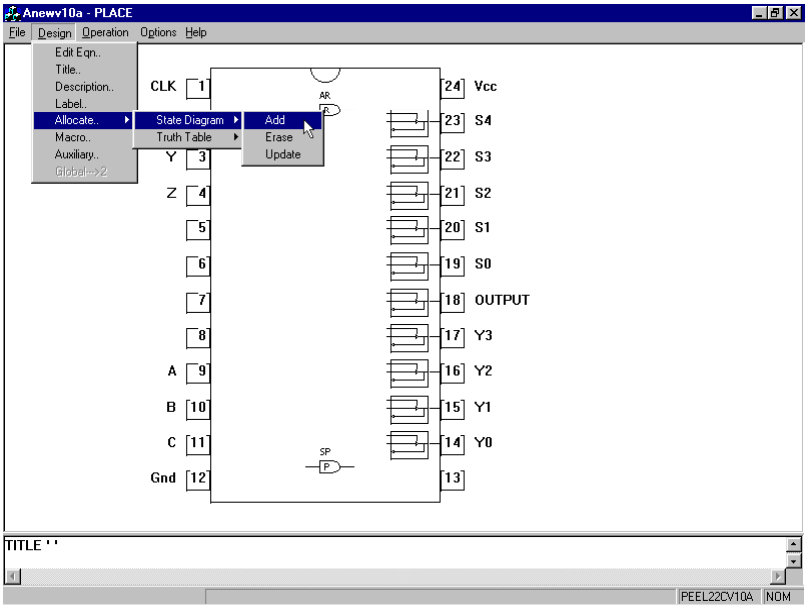


Figure 3-23 Adding a state diagram design

Insert the label for the new state diagram. The syntax of the label is similar to those used for labeling pins or cells (refer to Chapter 4 on "WinPLACE™ Design Language"). The next step is to allocate the state cells as illustrated in Figure 3-24.

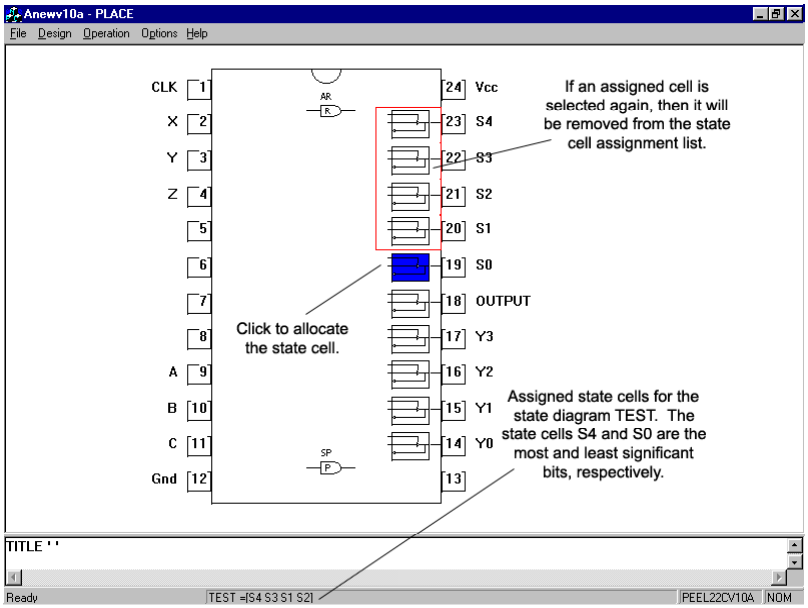


Figure 3-24 Allocating state cells for a state-diagram design

After completing the state cell assignments (by pressing the [Esc] key or click-R), a border surrounds the assigned state cells to indicate the state diagram. See Figure 3-25. Multiple state diagrams are differentiated between the line types in the borders.

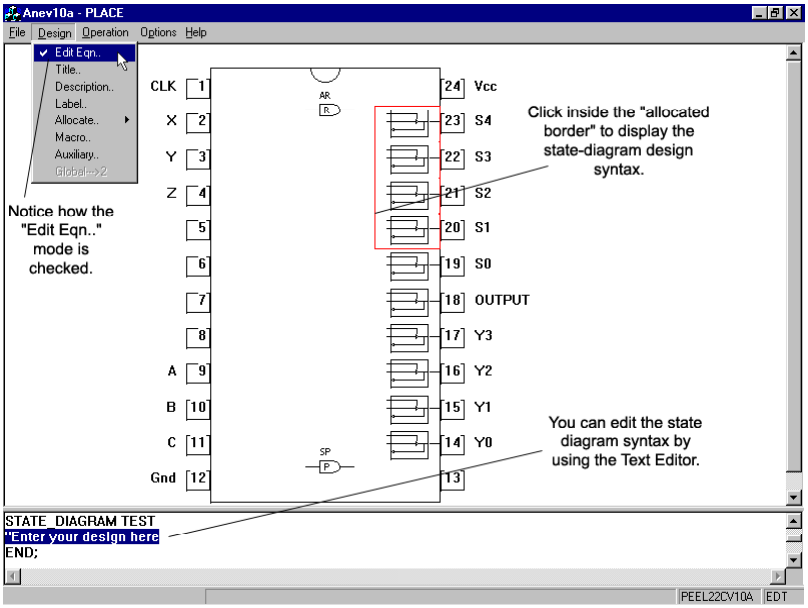


Figure 3-25 Entering design description for the state-diagram

3.13 Design Operation – Truth Table Designs

In truth table designs, the description of the logic design is in the form of a truth table. This design method is most suitable for random combinatorial logic applications.

Like the state diagram design procedure discussed in the above section, the truth table design begins with allocating labeled pins and cells. Figure 3-26 shows a new truth table being labeled "DECODE" at the beginning of the truth table "Allocate" command.

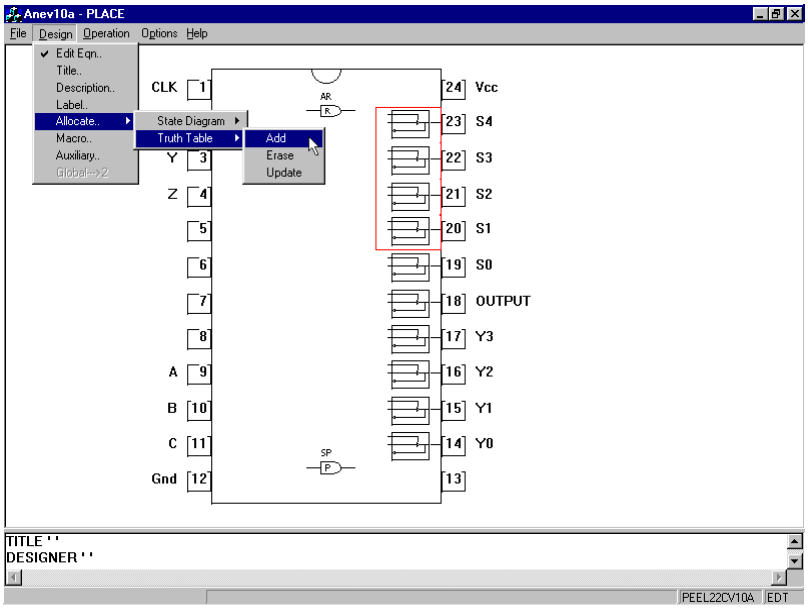


Figure 3-26 Adding a Truth Table Design

After typing in the label, the next step is to select the inputs of the truth table (Figure 3-27); the truth table feature automatically starts with this function. A

truth table input can be a pin, INC, IOC, or LCC. Click at the "Output" option in the pop-up menu bar to complete the input selection and to start the output selection. Press the [Esc] key or click-R to complete the output selection. When completed a window will appear reading, “Implement Changes?” When this appears, click the desired option to move on.

Note that during the input or output selection process, if you click on a selected pin or cell, then it will be removed from the input or output selection list.

Press the [Esc] key or click-R during the input or output selection process to abort or complete the "Allocate" command. A window will pop up to confirm implementing the changes made. Pressing the [Esc] key or clicking-R when the "Implement changes?" window is popped up will return you to the previous mode (input or output selection).

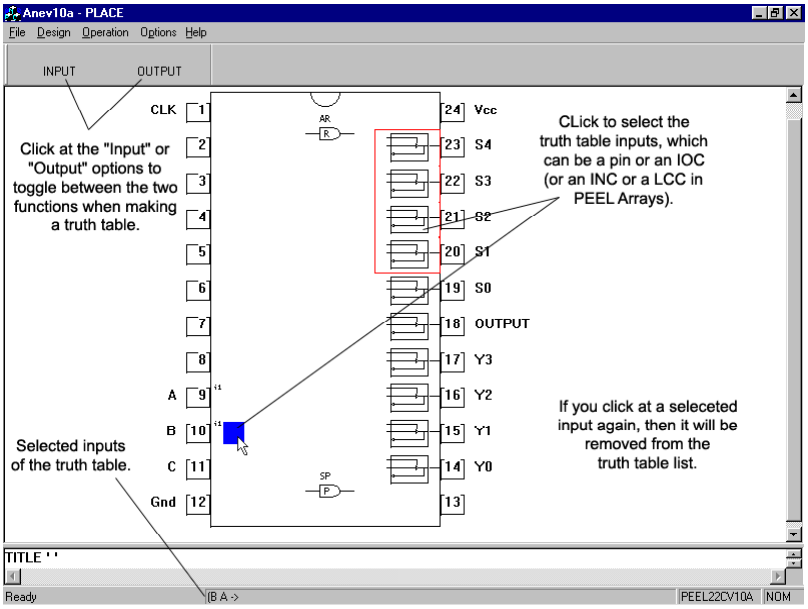


Figure 3-27 Selecting truth-table inputs

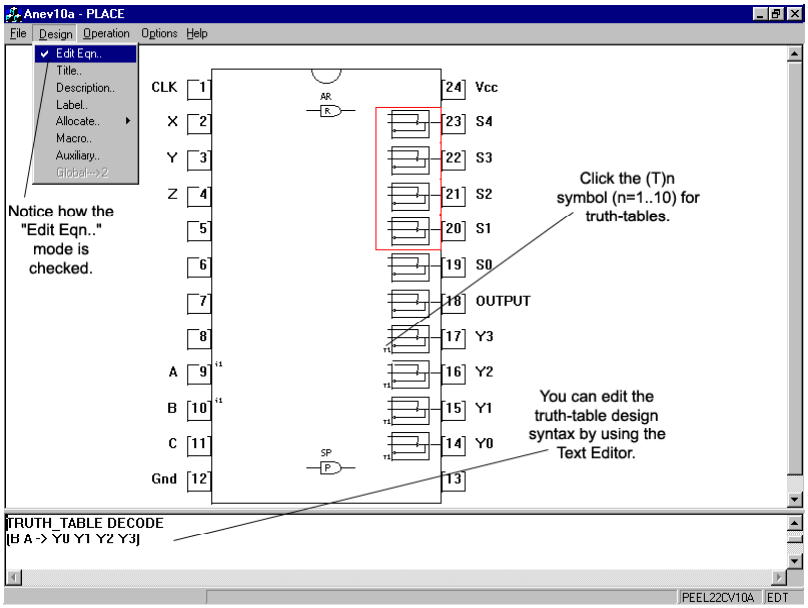


Figure 3-28 To enter the truth-table design description

Figure 3-28 shows the truth table window for entering the truth table design syntax. For more information on the syntax of the truth table design description, please refer to Chapter 4 on "WinPLACE™ Design Language".

3.14 Design Operation – The Text Editor

The Text Editor is always open in WinPLACE™ Design operation and is located at the bottom of the screen. Whenever an equation is selected instead of a window popping up, the cursor within the Text Editor automatically moves to that portion of the source file. An equation is opened by double-clicking the left mouse button on any SUM or AND node (gate), state diagram block, and truth table marker (indicated by Tn where n = 1 to 10). See Figure 3-28. Also view Section 3.24 for more information on the operations of the Text Editor.

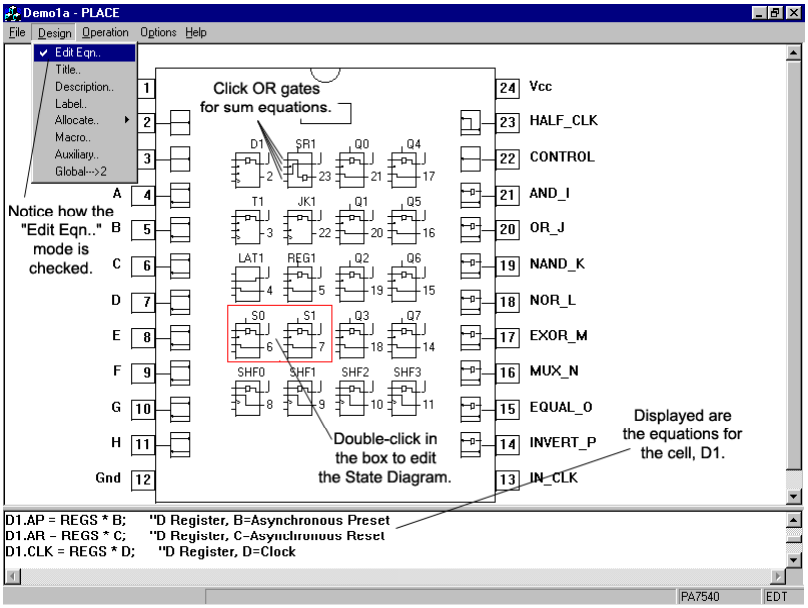


Figure 3-29 Equation text of the Global Cell A

3.15 Compile Operation – Main Screen

There are three windows in the main screen of the Compile Operation (Figure 3-30). Each of these windows provides a specific function for the compilation of WinPLACE™ Source Files (PSF).

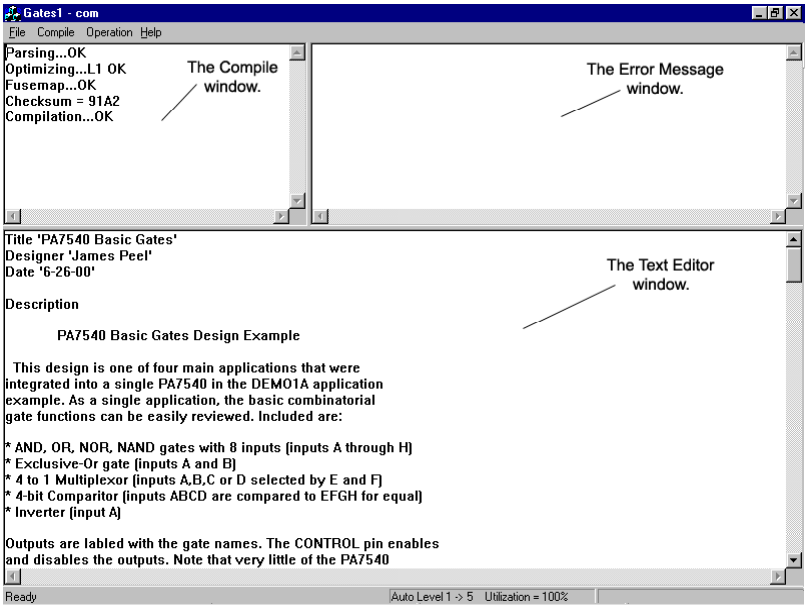


Figure 3-30 Compile Operation screen

Compile…..This window is located in the upper left hand corner of the program. It displays any information regarding the status of the compilation. The compilation of a PSF file is a three-step process: Parsing, Optimizing, and Fuse-mapping. Also, Checksum and Compilation time will be displayed.

Parsing - Checks the syntax of the PSF file and displays the error in the Error Message window.
Optimizing - Implements logic transformation (converting complex equations into sum-of-product form) and reduction to the parsed PSF file, and outputs the results to a file with extension ".RED". The RED file is in the PSF format; hence it can be read into the Design Operation for analysis of the reduced equations. After optimization is completed, the reduction level implemented is displayed. For instance in Figure 3-30, the "L1" term in "Optimizing ...L1 OK" indicates that the reduction level L1 was used for the GATES1.PSF design, and the optimization process was successful (as indicated by the "OK").
Fuse-mapping - This is the final step in the compilation process. After successful optimization, the reduced equations are mapped into the device and a JEDEC programming file is created. In addition, the fuse-mapper creates a ".MAP" file that contains information on how each equation is mapped in the JEDEC file.

Once the compilation is completed and successful, the checksum of the JEDEC file and compilation time will be displayed.

Error Message….This window is in the upper left hand corner of the program display. It shows any errors that were encountered during the compilation process. Figure 3-31 shows an example of a syntax error in which an unknown signal "MODE2" was encountered in the macro definition (the correct signal is "Mode2"). Once an error is encountered the compilation is aborted, and the text editor is opened automatically to allow the highlighted error to be investigated or analyzed. Note that the highlighted line sometimes does not contain the error. This may be due to an incorrect format for comments in previous lines. Please refer to Chapter 4 for more information on the Comment format.

Editor……...The Editor window is shown on the bottom and spans across the entire screen. It allows for direct design edits and/or modifications. If a syntax error is found during the compiling process, the editor automatically places the cursor next to the error line. To change the size of the Editor, simply place the mouse over the separation bar between the Text Editor and the Compile windows. The cursor should change to an up and down double sided arrow. Click-LH and drag the bar up or down depending on how big or small you want the Editor.

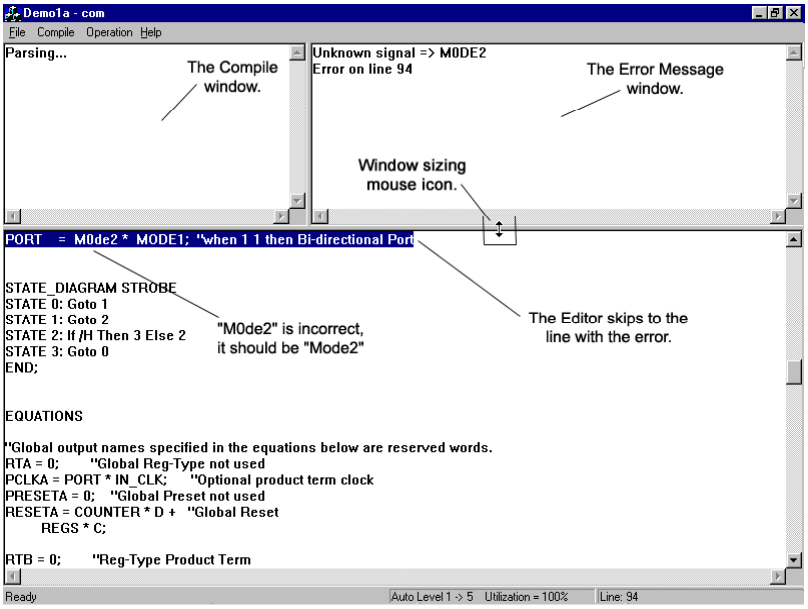


Figure 3-31 Encountering a compilation syntax error

3.16 Compiler Operation – Compiler Menu

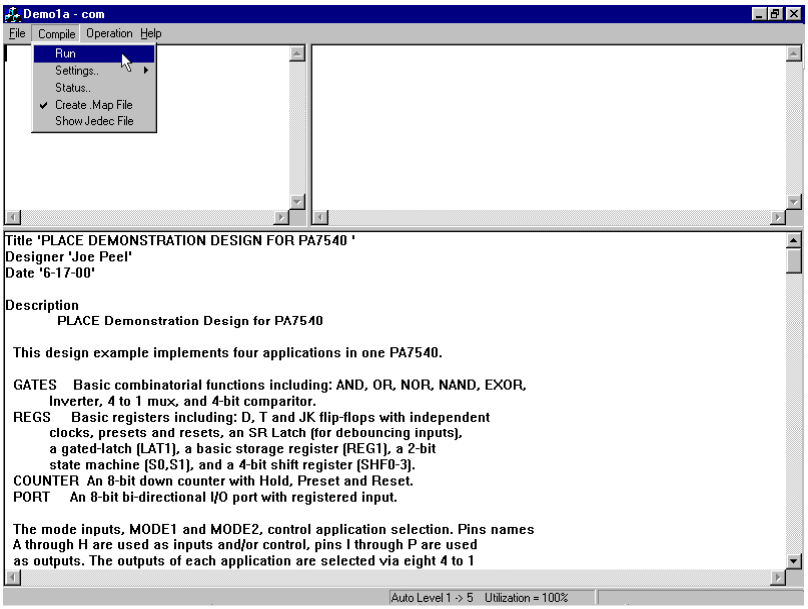


Figure 3-32 Compile Menu

Run…….Compiles the currently selected SOURCE FILE design file. If the compilation process is successful, then a JEDEC ".JED" file will be created. If a compilation error occurs, an error message will be displayed in the Error Message window. In addition, the editor will be opened and the line containing the error highlighted automatically. You can then make the edits here or go back to the Design operation to correct the source file. Please refer back to Section 3.15 for more information on the compilation process. Note: The

compiler can be executed by clicking on the "Compile" window heading.

Settings……Allows the selection of logic optimization level and product term utilization as listed in Table 3-4.

-- **The default option for the reduction level selection is "Auto 1-5".** The "Auto" reduction levels refer to the automatic increments of the reduction levels. For in stance, the "Auto 1-5" means that the optimization process starts with reduction level 1. If the design does not fit the device after the completion of level 1, then it proceeds to level 2. If the design still does not fit the device, it proceeds to level 3 and so on until a fit occurs. Once the design achieves a device fit, the reduction level will be displayed in the Compile window (Figure 3-32).

-- There are five reduction levels, ranging from no logic reduction to group reduction with demorganization of outputs. The higher the level the better the utilization but the optimization time will also be longer. When compiling a design for the first time, it is recommended to use the "Auto 1-5" option. Note the level needed (displayed in the Compile window) to successfully compile the design. For subsequent compilations, select the single reduction level.

-- **"Utilization":** This command sets the maximum number of product terms (as a percentage) to be used during the logic optimization-process. For instance in the PA7540 device, if the product utilization is set as "Utilization = 60%", then 60% of 80 product terms (80 product terms is the maximum for the PA7540) will be used during logic optimization. Hence, this command allows you to estimate whether additional logic can be implemented in the selected device. **The default product term utilization is "Utilization = 100%".**

Status……..Displays the device utilization and use of architecture after the design is compiled.

Create
MAP File….During the fuse-mapping process, a ".MAP" file can be created in addition to the JEDEC file. This MAP file contains the detailed information regarding how each equation is mapped in the JEDEC file. This option can be turned ON or OFF in the "Compile" menu. The default is to have this feature ON and so you will see a check next to the command "Create .MAP File". Simply re-click this option to take the check away and turn the option OFF.

Optimization Function Level	
Level 1	No reduction. Transforms equations to sum-of-products.
Level 2	Simple reduction. Combines duplicate product terms.
Level 3	Pin reduction. Optimizes terms per individual equation.
Level 4	Group reduction. Optimizes terms over all equations that can be shared.
Level 5	Group with output polarity inversion reduction. deMorganizes Outputs (automatically inverts polarity) to achieve best optimization of terms.
Auto 1-2	Optimizes from Level 1 to 2 until logic fits.
Auto 1-3	Optimizes from Level 1 to 3 until logic fits.
Auto 1-4	Optimizes from Level 1 to 4 until logic fits.
Auto 1-5	Optimizes from Level 1 to 5 until logic fits.
Utilization	Sets the maximum product terms that can be used in the logic reduction or optimization process. The number is set in percentage terms.

Table 3-2 "Settings" menu in the Compile menu window

Simulate Operation – Waveform Screen

After successful compilation of a PSF design file, test-vectors can be used to verify the design. In the Simulate operation, these test-vectors are displayed as waveform signals (Figure 3-33). The vectors are created using the "Edit" command. The vectors are then used to simulate (logically only) the function of the design by retrieving the design's logic from the JEDEC file that was created during the compilation process. These vectors can also be appended to the JEDEC file so that they can be used to exercise the device after programming.

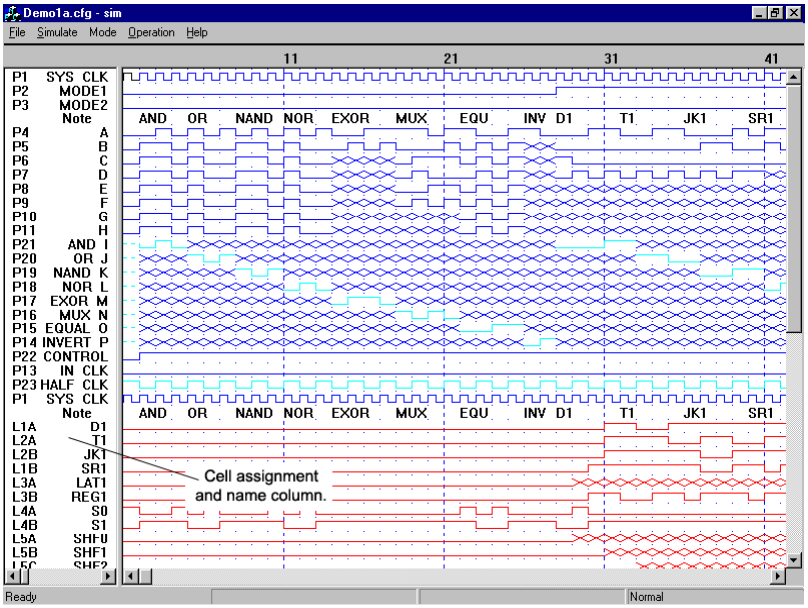


Figure 3-33 Simulate waveform screen

In the waveform display, each waveform row represents a signal from a pin or an internal node (e.g., a LCC, IOC or INC). A "P" followed by the pin number indicates the external pins. LCC internal output is indicated by an "L" with the cell assignment coordinates. The prefix for the IOC or INC node (i.e. the output of the IOC or INC register) is an "I" followed by the assigned pin number. The IOC and INC nodes are differentiated by the assigned pin numbers. For instance, in the PA7140 PLCC device, pins 3-5, 13-21, 25-27 and 41-43 are assigned to INC's. The rest of the pins except pins 2 and 24 (CLK pins) are assigned to IOC's. For more information, please refer to the ICT data book.

Some examples of the waveform pin and cell assignments and labels are found in the TC7140.PSF example design file for the PLCC-packaged PA7140 (the PA7140 device has all the cells, i.e., LCC's, IOC's and INC's). These examples include:

P2	CLK	=> pin 2 with label CLK
L3A	P2	=> LCC 3A with label P2
I15	C2	=> IOC 15 with label C2
I4	SELECT	=> INC 4 with label SELECT

Scrolling in the waveform display

As illustrated in Figure 3-34, scrolling can be accomplished by clicking at the scroll arrow markers located on the right side and the bottom of the screen. An additional method of scrolling is to Click-LH on the scroll bar, then, drag it up and down. Also, if you have a 3-button mouse where the middle button is a scroll button it can be used to go up and down.

Alternately, you can click-LH (click and hold the left mouse button) at the "position block" to jump to another waveform display section. Once you have selected the location, release the left button to return to the normal screen mode.

Functions of the waveform signals

In the Simulate operation, the graphical image of each waveform signal represents a specific function. See Figure 3-35. For instance, the waveform signal image , which is represented by the test-vector C in the JEDEC file, indicates a Low-High-Low input pulse. This signal functions as a clock for triggering a register on the rising or falling edge of the signal. Please refer to Table 3-3 and Table 3-4 for additional information on other waveform signals.

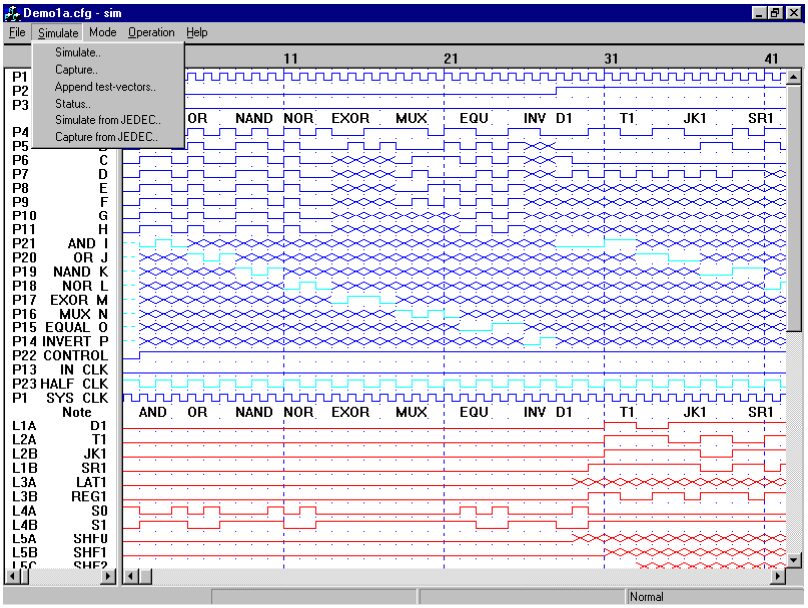


Figure 3-34 Simulate waveform screen

Symbol	Color	Function	JEDEC Vector Symbol
	Blue	System Clock	C
	Blue	Input High	1
	Blue	Input Low	0
	Blue	High Voltage Preload	P
	Blue	Input or Output Don't Care	X
	Red	Output High	H
	Red	Output Low	L
	Red	Output High Impedance	Z
	Green	Buried or Internal Signals which cannot be modified	N/A

Table 3-3 Waveform signal symbol table for a color monitor

Symbol	Color	Function	JEDEC Vector Symbol
	Normal	System Clock	C
	Normal	Input High	1
	Normal	Input Low	0
	Normal	High Voltage Preload	P
	Normal	Input or Output Don't Care	X
	Thick	Output High	H
	Thick	Output Low	L
	Dotted	Output High Impedance	Z
	Center	Buried or Internal Signals which cannot be modified	N/A

Table 3-4 Waveform signal symbols for a monochrome monitor

Figure 3-33 through Figure 3-47 illustrated in this section were captured on a color monitor. Refer to Table 3-3 for the functions of the waveform signals in these figures.

3.17 Simulate Operation – Simulate Menu

Simulate…..Performs logic simulation of waveform vectors on external signals, i.e., on the "P" waveform rows only. The simulator compares the simulated signals with the current signals on the pins, and then marks the locations with signals that do not match. These marked locations are vector simulation errors (Figure 3-35). The special symbols used indicate the type of simulation errors (Table 3-5). On the other hand, the signals for all internal nodes are not checked but are automatically captured during logic simulation.

During simulation, the design's logic is retrieved from the JEDEC file that has the same root name. For example, the JEDEC file DEMO1A.JED will be used during the vector simulation of DEMO1A.CFG.

Capture……Unlike the "Simulate" command, this command "captures" the signal on all external outputs. With this command, signals on the output pins need not be generated because the simulator automatically inserts the simulated signals. In addition, the simulated signals are also inserted into vector locations, which contain the output Low "L", output High "H" and Don't Care "X" signals. This means that if you have simulation errors on any of these vector locations, they will be replaced by the simulated signals. See Figure 3-35.

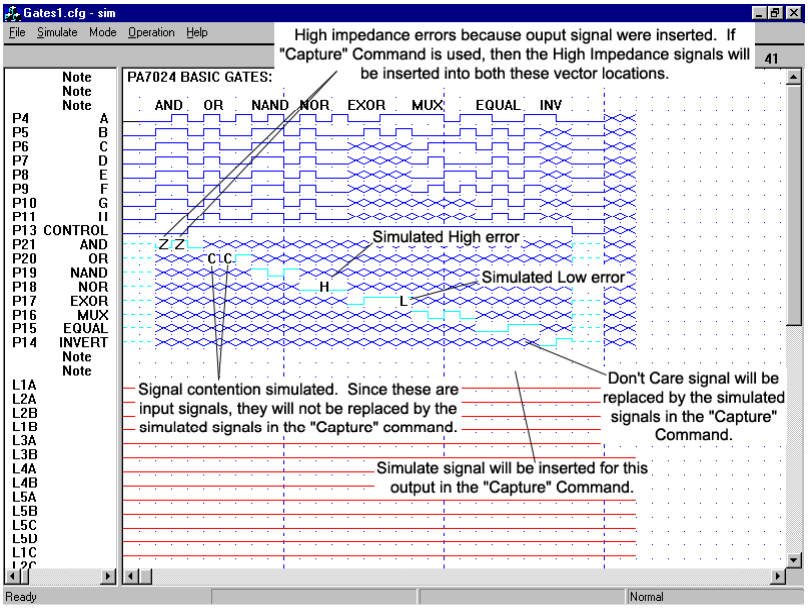


Figure 3-35 Vector simulate errors

Error Symbol	Function
L	Indicates that a LOW signal is simulated on the pin at the current vector location.
H	Indicates that a HIGH signal is simulated on the pin at the current vector location.
Z	Indicates that the pin is in a High Impedance condition at the current vector location. An example is the insertion of output signals ("L" or "H") on inputs.
C	Indicates that a signal contention condition exists at the current vector location. An example is the insertion of input signals ("0" or "1") on outputs.
U	Indicates that the signal at the current vector location is unstable or in an indeterminate state. An example of an application that causes this error is an oscillator. When an unstable error occurs, the simulation process is aborted Note that the WinPLACE™ Simulator will not flag an error if the unstable output has a Don't Care symbol.

Table 3-5 Vector Simulation Error symbols and their functions

Please refer to Figure 3-36 for information on the "Simulate" and "Capture" commands for asynchronous clock designs.

Append test
Vectors……This command converts the waveform vectors into the JEDEC file test-vectors and appends them to the ".JED" file. This allows the vectors to exercise the device on a PLD programmer.

Status……..Provides status information such as the maximum vector columns available with the current system configuration (more system RAM memory, more vector columns available), number of total vectors used, number of simulation errors, and the previous simulation time.

Simulate from

JEDEC…….With this function, the WinPLACE™ Simulate waveforms can be generated from the test vectors specified in the JEDEC file. If an ICT PDS-3 programmer is present, this function together with the "Capture" function in the "Test" menu of the Program operation allows viewing of device vector results via the waveform screen.

Capture from JEDEC……This command is similar to that of the "Simulate from JEDEC" command except that the simulated signal levels will replace the output signal levels.

3.18 Test Vectors for Asynchronous Clock – Sum or Product term clock

Asynchronous clock refers to the triggering of the LCC or IOC register via a sum or product term. Devices with the asynchronous clock capability include PA7540, PA7536, PA7572, PA7024, PA7128 and PA7140. If the simulated waveform vectors are appended to the JEDEC file for exercising the device, then special attention for the asynchronous clock designs is needed. This is because of possible data set-up time violations due to how the input signals are applied on the PLD programmer.

Note: On some programmers, the input signals are applied serially starting from pin 1 after the device has been powered-up (Vcc pin set to 5V). First, input signals "0" and "1" are applied, then preset signals "P" (this is a high voltage preset which is not supported in many PEEL™ devices), then clock signals ("C"), and then the output pins are sensed and compared with the vectors from the JEDEC file. With this method, data set-up time violations for asynchronous clock designs are very possible, especially if the input signals "0" and "1" are used to emulate the clock signals.

There are two methods to ensure proper test vectors for testing asynchronous clock designs.

1. The first method is to use the dedicated clock signal ⊓ (JEDEC "C") in the sum or product term equation. Since this signal exhibits a Low-High-Low voltage level in a given vector period, the standard logic operation of the AND and OR operators may be applied only with some modifications.

Clock	Operator	Input	Result
C	& (AND)	1	C
C	&	0	0
C	# (OR)	1	C
C	#	0	C

By using the dedicated clock signal, the programmer applies the signal to the clock only after all input signals are applied. Note that if the result "C" is routed to an external output, then a signal contention error will be flagged.

2. The second method is to add dummy or Wait states prior to all clock edges which are generated by the "0" and

"1" input signals. The advantage with this method is that the same clock signal can be routed to an external output without encountering a signal contention simulation error as in the preceding method. However, the disadvantage is that a minimum of two vectors are required to generate a clock cycle, "0" and "1" signals on successive vectors.

3.19 Preloading the LCC registers (in logic simulation only)

In the Simulate operation for the PA7540, PA7536, PA7572, PA7024, PA7128 and PA7140 devices, all the LCC registers can be preloaded with user-specified data. Note that this feature is available only in software and does not exist in the device physically. If the preload vectors are applied to the device on a PLD programmer, these vectors will fail.

In Figure 3-36, the PA7540 application example has two sets of registered output pins P0..P7. At the preload vector column 11 and 22, the LCC registers for these outputs are preloaded with data 0 and FF HEX respectively. The preload condition is activated by the waveform symbol \ (JEDEC "P") on the dedicated preload pin 13 (each device type has a specific pin dedicated for the preload function). The clock symbol ⌐¬ on pin 1 is not necessary because the preload symbol automatically loads the data asynchronously. Unload/Preload Select must be set to appropriate polarity.

Below is a list of the preload pins and the assigned pins for the LCC registers

By adding the preload condition at the beginning of each CFG file (except for the first one), then up to 36 CFG files (CFA, CFB,..., CFZ and CF0, CF1,..., CF's) can be linked together for simulating a large PEEL™ Array design. With approximately 700 vectors per CFG file for a 512K system, about 25,000 "continuous" vectors can be simulated for a design. Note: Preload feature is not available for JK registers.

Device	Preload Pin	Unload/Preload Select	Preload Data Pins	LCC Registers
PA7024 (DIP)/ PA7540	13		2-6	LCC 1A-5A
		1 = "0"	7-11	LCC 1B-5B
			14-18	LCC 1C-5C
			13-23	LCC 1D-5D
PA7128/ PA7536	1	2 = "1"	15-17	LCC 1A-3A
		13 = "1"	18-20	LCC 1B-3B
			21-23	LCC 1C-3C
			24-27	LCC 1D-3D
PA7140 (PLCC)/ PA7572	24	2 = "0"	6-11	LCC 1A-6A

12-16, 18	LCC 1B-6B
28-33	LCC 1C-6C
34-38, 40	LCC 1D-6D

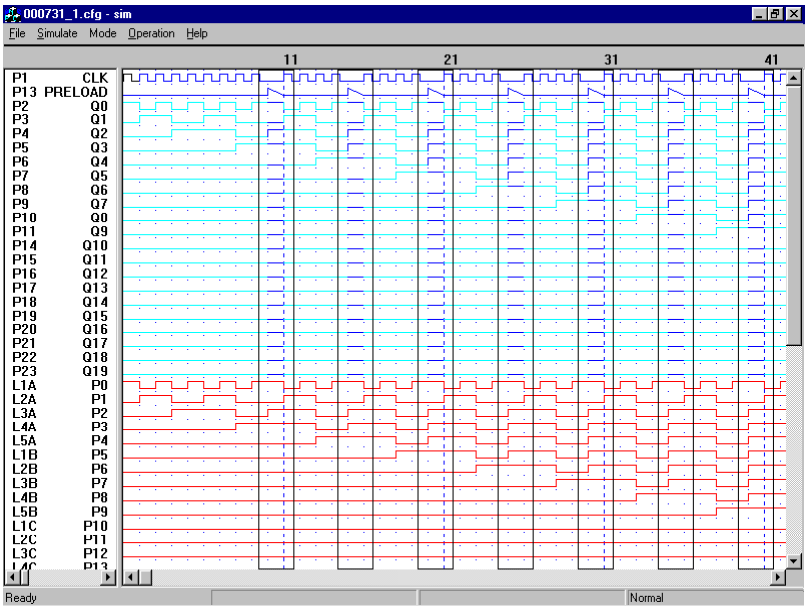


Figure 3-36 PA7540 LCC Register Preload, and Product or Sum Clock application

3.20 Simulation Operation – Entering or Editing Waveforms

In the Simulate operation, there are two ways to access the modification commands. The first is through the pull down menu labeled, "Mode". This menu contains seven organization commands (Col Note, Row Note, Copy, Dele, Move, Swap, and PSF). The "edit" command is used for modifications. To access the rest of the commands, right-click on the simulation screen (Figure 3-37). This will open a pull-down menu next to the mouse revealing six commands (Copy, Edit, Delete, Move, Swap, and Zoom). For text purposes this menu will be known as the “Right-Click” menu. The two command menus contain some repeated commands and some different ones. The ones repeated are identical in terms of their operation however; they are not identical in the way that they are presented.

In the Simulate operation, the test vectors are entered or modified via the "Edit" command. All other commands in the "Mode" menu (such as Col Note, Row Note, Copy, etc.) are merely used for organizing the waveform screen. This means that these commands do not affect the results of the vector simulation. Please refer to Section 3.21 for more information on the waveform organization commands.

Note that only waveform signals for the external pins (inputs or outputs) can be entered or edited. All internal node signals are captured by the simulator and displayed for analysis.

Once the “Edit” command is highlighted, the advanced commands for editing test vectors will pan out from the menu. See Figure 3-38. In

addition, the previous block of vectors selected via the BBegin and BEnd commands will be displayed. **Click-R to exit any of the "Edit" commands.**

The “Edit” commands in the "Right-Click" and “Mode pull-down menus:

Edit……..Edits input or output vectors. There are three methods of entering or editing the waveform vectors.

1. Move the edit “box” cursor to a vector location and click the mouse button. Continue clicking until the desired waveform signal is displayed. Since there are eight different signals possible, each vector type is selected again after every eight clicks.
2. Move the edit cursor to a vector location and press the vector symbol keys such as C, 1, 0, P, H, L, X or Z to select the type of waveform signal. For instance, pressing the key "C" will select the clock signal for the current vector location. Refer to Table 3-3 or Table 3-4 for the description of each waveform signal. Note that the vector symbols are actually the standard symbols used in the test vector section of the JEDEC file.
3. Use the "Drag" command, which will be described later in this section.

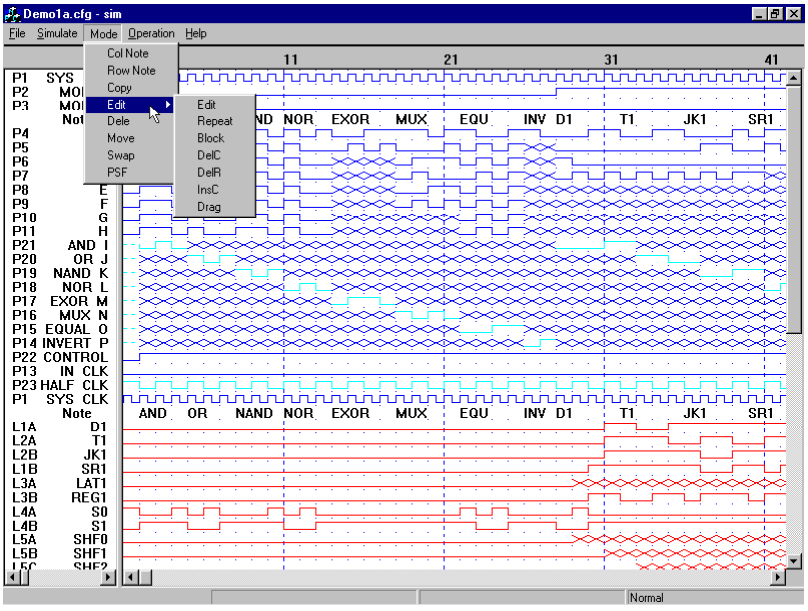


Figure 3-37 The "Edit" commands in the “Mode” menu

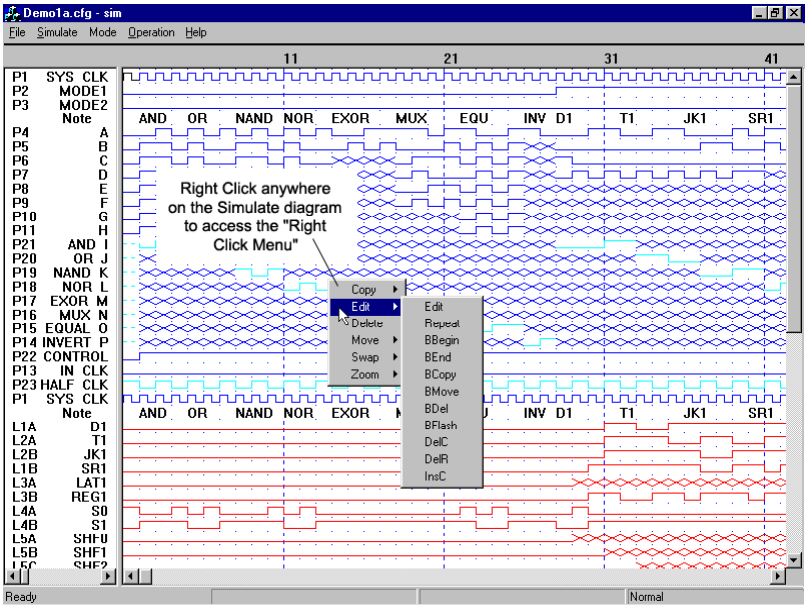


Figure 3-38 The “Edit” commands under the “Right-Click” option

Repeat…….”Repeat” allows a single vector to be repeated by a specified number. Aim at a vector location with the mouse and right-click on that spot to select the "Repeat" command in the “Edit” menu. This vector location will be the starting point. Then, enter the number of vectors to repeat and press Enter. All existing vectors following the starting vector will be overwritten. See Figure 3-39.

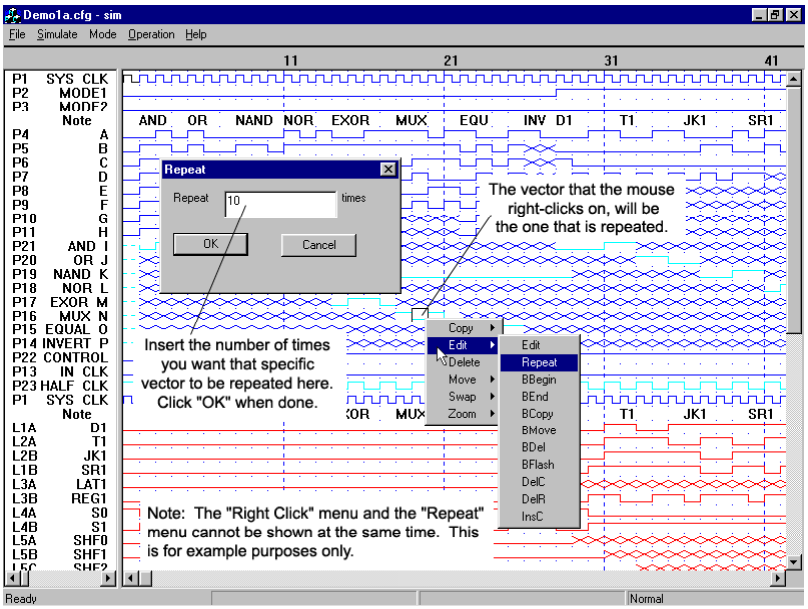


Figure 3-39 "Repeat" command

The “Edit” commands in only the "Right-Click" pull-down menu:

The commands that begin with the letter “B” indicate that they are block commands. Block commands are commands

that manipulate a block of test vectors referred to as a "vector block" (Figure 3-40).

BBegin……This command allows a vector location to be selected as the beginning vector of the "vector block".

BEnd…….Allows the selection of an ending vector for the "vector block". Like the "BBegin" command, a new ending vector can be selected for a current "vector block".

The beginning and ending vectors of the "vector block" must always be located on the upper-left and bottom-right of the block respectively. A block can have a single row or column. In this case, the beginning vectors are the most left or top vectors, and the ending vectors are the most right or bottom vectors. In addition, beginning and ending vectors can be located on separate waveform screens.

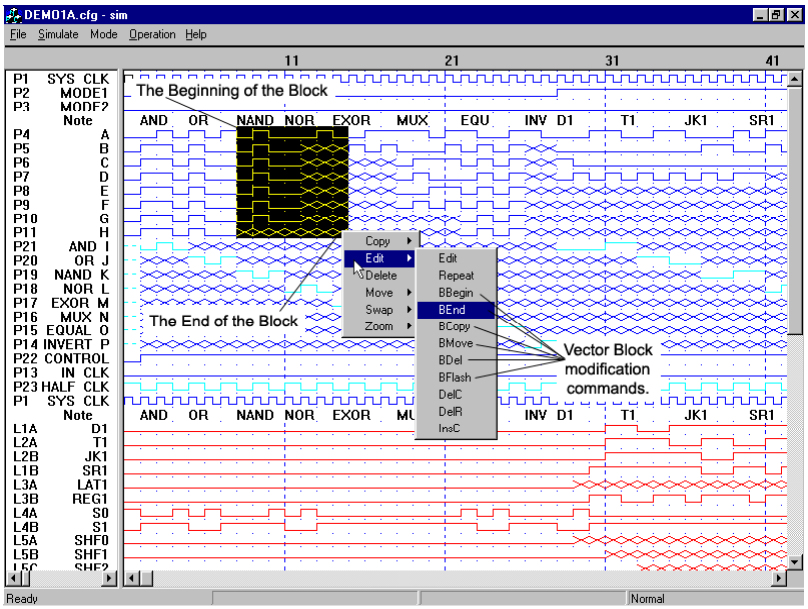


Figure 3-40 Vector block selected

BCopy……..Copies the currently selected block into another waveform area after the selection of a starting vector location.

BMove…….Moves the currently selected block into another waveform area after the selection of a starting vector location.

BDel…….Deletes the currently selected block of vectors. A pop-up window to confirm the deletion will appear.

The following lists the row or column commands. These commands are also repeated in the "Mode" menu.

DelC…….Enters a mode that allows the current waveform vector column to be deleted.

DelR……Enters a mode that allows all the vectors on the selected waveform vector row to be deleted.

InsC…….Enters a mode where vector columns can be inserted. The vector column is inserted prior to the selected vector column (Figure 3-41). Additionally, the signals on the new column follow the selected vector column.

The following are commands that are only in the “Mode” menu:

Block……Allows you to create a Block vector by simply clicking on the beginning vector and dragging the mouse to the location of the end vector. This skips the process of selecting the “BBegin” command and the “Bend” command.

Drag……Enters a mode which allows input or output signals to be entered or edited. Only two signals are available in the drag mode: input "0" and input "1".

The “Drag” command is also a row or column command.

Select the waveform vector row and column after you have entered the "Drag" command. The drag edits will start from this location. Move the cursor horizontally across from left to right and click at the selected vector location to end the current signal. Click again to select another vector location. To exit the drag mode, click-R or press the [Esc] key. Note that the drag edits will overwrite the previous vectors.

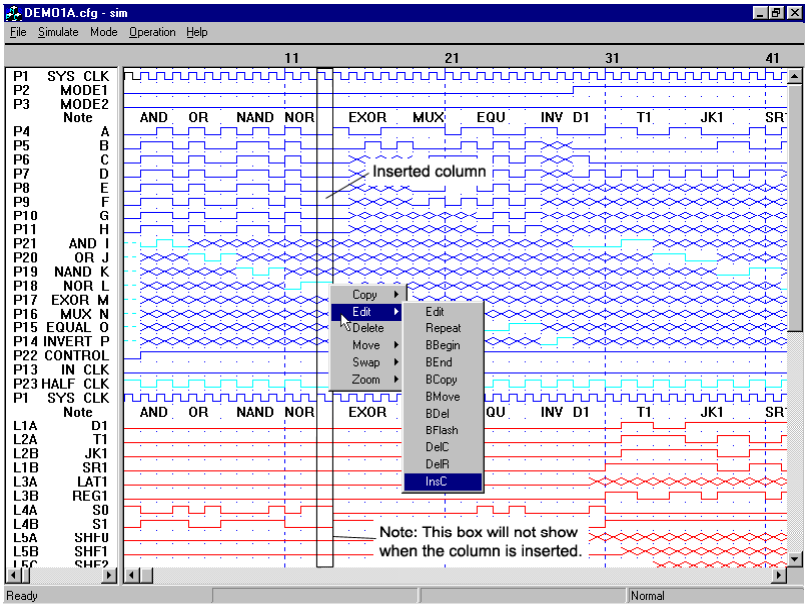


Figure 3-41 Inserting a vector column

3.21 Simulation Operation – Organizing Waveforms

All the commands in this section are used to organize the display of the waveform screen for a better understanding of WinPLACE™ designs. Therefore these commands do not affect the results of the simulation or the actual generation of the test-vectors for the JEDEC file. These commands, as discussed in Section3.20, can be found both in the “Mode” menu and the “Right-Click” menu (found by right-clicking on the Simulation display see Figure 3-42).

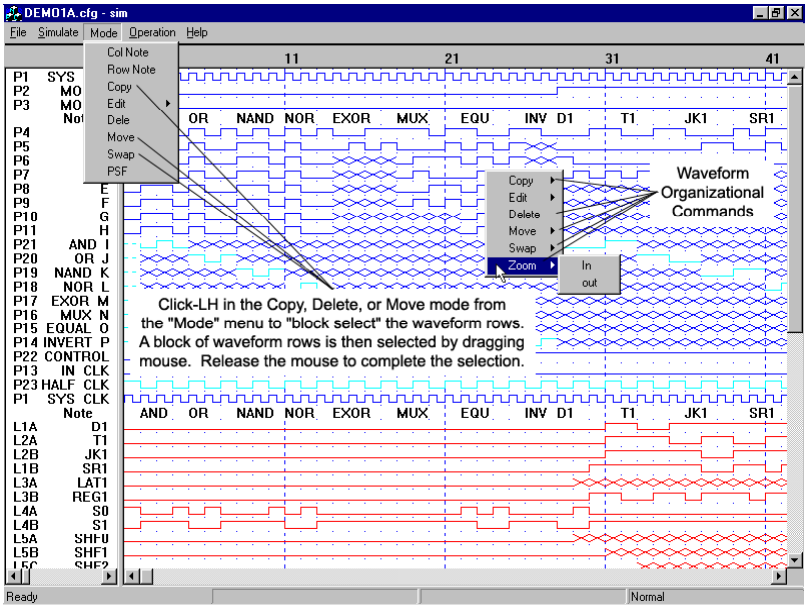


Figure 3-42 Commands for organizing the waveform screen

Col Note…..Allows a "Note" column to be inserted for adding comments to the waveform display. This feature allows for multiple notes in one row, but only one note per column. To add the note line, select the function in the "Mode" menu and click at the desired line, click-R to exit the note command. To edit the notes for each column, simply move the mouse cursor to the desired column and double-click in the empty space. A maximum of 60 characters can be entered per note.

Row Note…Allows a "Note Row" to be inserted for adding comments to the waveform display. This row can only have one note for the entire row. Add the note row by clicking at the desired line, Click-R to exit the note command. To edit the row, move the mouse cursor to the note line and double-click in the empty space. A maximum of 60 characters can be entered into this note.

Copy……Copies a waveform vector row or a "block" of selected waveform rows to another location. This command will prompt for a source and target selection. Click-R to exit this mode.

Dele or Delete……..The "Dele" function is found in the "Mode" menu and is explained here. It removes a waveform vector row or a "block" of selected waveform rows from the current waveform screen. Note that the unduplicated waveform rows for external pins and internal nodes will be appended to the bottom of the CFG file (last screen of the waveform display). All other waveform rows such as Notes and duplicated rows (i.e. rows that were copied using the "Copy" command) can be removed from the CFG file. The CFG file has a minimum number of waveform rows allocated for each device type that cannot be removed. For instance, the PA7540 device has 62 waveform signals (22 inputs/outputs + 20 LCC internal outputs + 20 IOC registered nodes) that are always present in the CFG file. Click-R to exit. The

"Delete" function is essentially the same thing, only it is found in the "Right-Click" menu and it will delete the row that the cursor is on when you right-click. This mode stops after one time.

Move……This function can be found in both menus, the "Mode" menu will be discussed first. It allows a waveform vector row or a "block" of selected waveform rows to be moved from one location to another location. To select the line to be moved click on it and it will become highlighted. To select multiple lines simply click on the starting line and drag your mouse down or up to highlight them all. The next row that you click will signal the highlighted line(s) to move one row above it. Click-R to exit.

Swap……This function can also be found in both the "Mode" and "Right-Click" menus. In the "Mode" menu it allows a waveform vector row to be swapped with another vector row. This command will highlight the first selection and then swap with the next. Click-R to exit. No "block" swapping is available. In the "Right-Click" menu, the operation is the same, however, when you select the "Swap" function a window will pan out to the right containing the commands "Source" and "Target". When swapping two lines first select a source and then select the target line that you want it to change with.

3.22 Simulate Operation – Zoom Command

The "Zoom" command is located in the "Right-Click" pull-down menu. Refer to the first paragraph in Section 3.22for more information. When the "Zoom" command is highlighted, another window will pan out to reveal the options, "In" and "Out". The default view in the Simulate operation is 35 rows by 42 columns and is the farthest out that the program can go. The "Zoom" allows for two other closer views. With less vectors being displayed, less waveform vectors can be viewed on a single screen, and hence there can be less of a cluttered feeling and more precise work can be done.

Within the Zoom mode, there are actually three separate modes that will be referred to as Zoom Modes A, B, and C. The default mode is the A mode. From the normal waveform screen mode, right-click to access the "Zoom" menu and select the "In" command to enter Zoom Mode B (Figure 3-43), do it again to enter Zoom Mode C (Figure 3-44). The screen in the first mode consists of 18 rows and 21 columns and the second has 12 rows and 14 columns. The only difference amongst the modes is how many rows and columns that can be viewed. All of the Simulate commands can be used in each mode.

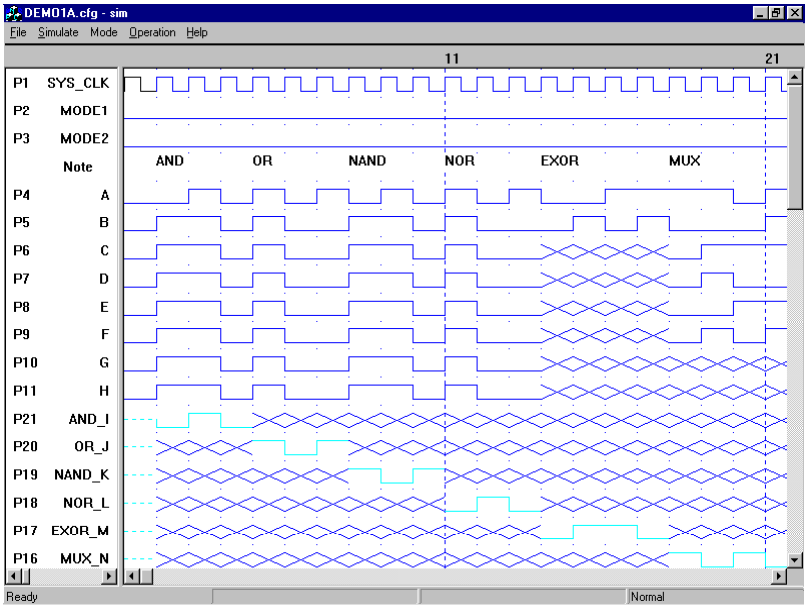


Figure 3-43 Mode B of the "Zoom" waveform screen

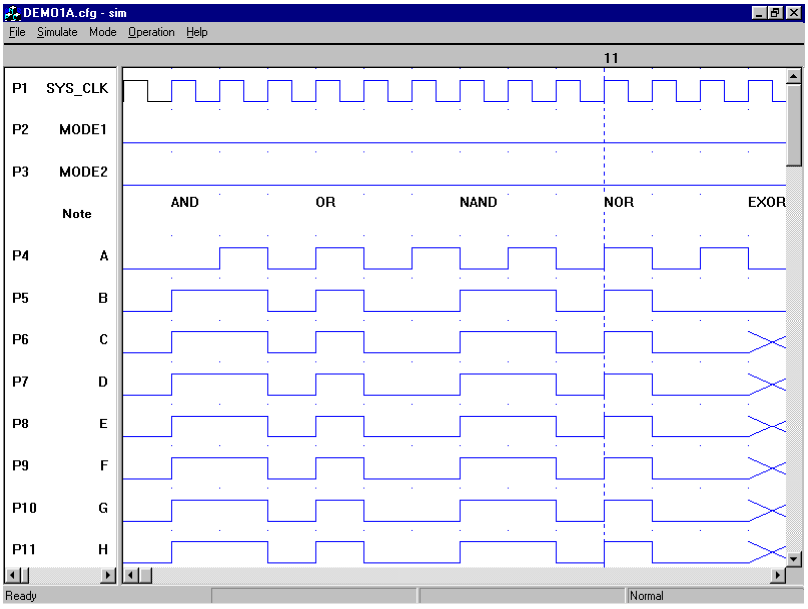


Figure 3-44 Mode C of the "Zoom" waveform screen

3.23 Simulate Operation – PSF Command

The PSF design file can be displayed on the waveform screen by selecting the "PSF" command located in the “Mode” menu. See Figure 3-45. When selected the WinPLACE™ Text Editor will not open, but rather a plain text file will open to display the source. With the design source file displayed, you can compare the simulation results with the design logic. If a simulation waveform vector error is detected, then the modification can be done using the "Edit" command. If the error is in the design logic, then return to the Design operation to correct the error and recompile the PSF file.

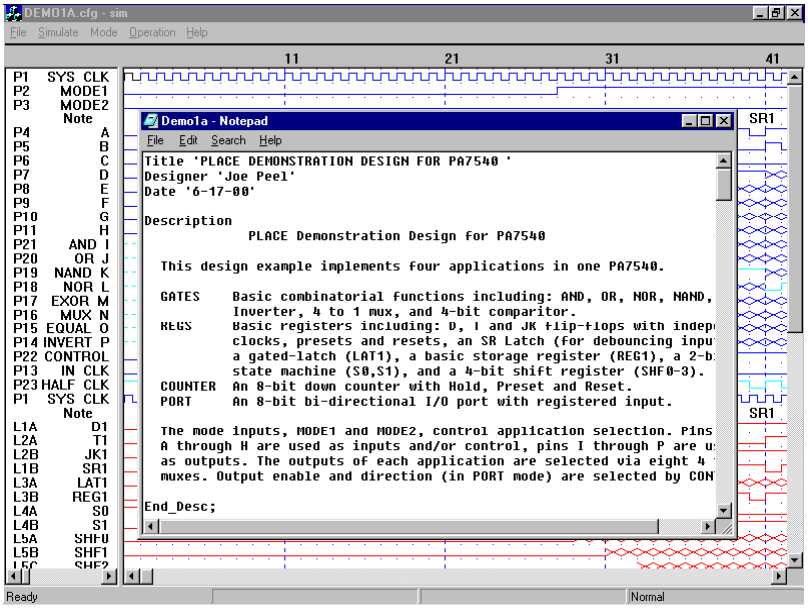


Figure 3-45 Displaying the PSF file on the waveform screen

As shown in Figure 3-45, a window will pop-up displaying the WinPLACE™ design source file. The window that appears allows for edits to be applied to the PSF file, however if you edit the file the program will not recognize those changes. Any changes that you want to be read will have to be done via the WinPLACE™ Text Editor. The features of this window consist of paging the display screen up and down, window sizing, and the ability to move the display window to another location (Figure 3-46). To close the PSF window go to the “File” menu and select “Exit” or simply click on the “X” box in the upper right hand corner of the window.

Paging Up and Down

As in the Text Editor, paging can be done within the PSF window by clicking at the scroll arrows or bars located at the right and bottom edges of the window, by pressing the PgUp and PgDn keys, or by using the scroll roller on a 3-button scroll mouse.

Sizing the Display Window

The size of the window can be adjusted by dragging one of the corners in the direction that you want it. Release the mouse to set the new size. See Figure 3-46. To immediately make it the size of your screen click on the button to the left of the “X” button on the upper right hand edge of the window.

Moving the Display Window

To move the window, click-LH on the top bar of the window. Drag the window to the desired location, then release the mouse button. See Figure 3-47.

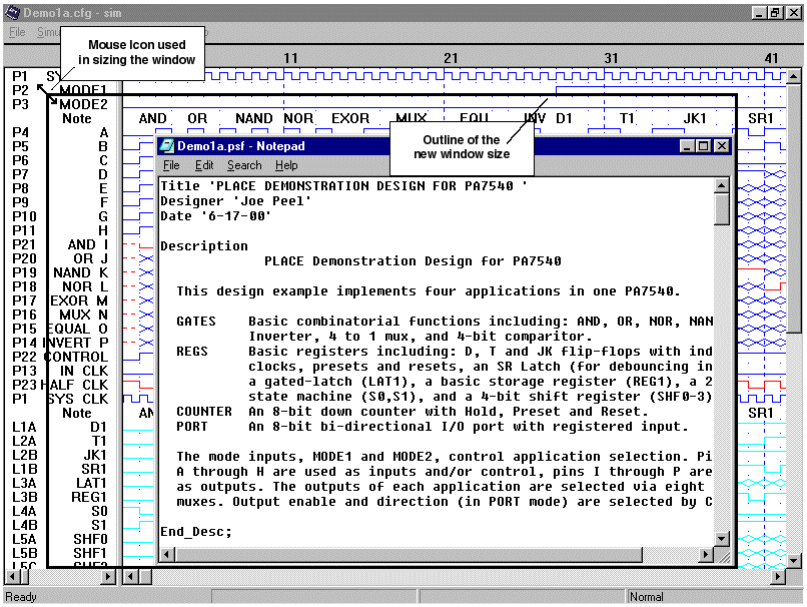


Figure 3-46 Sizing the PSF display window

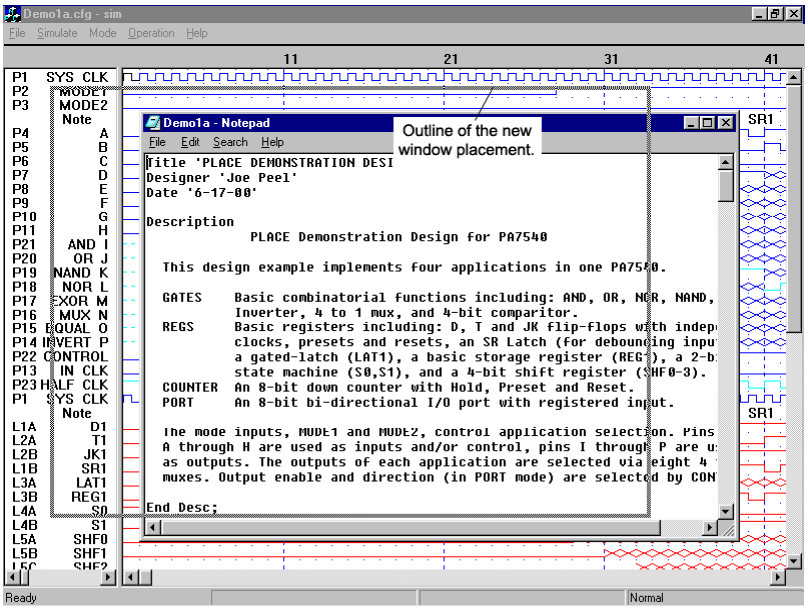


Figure 3-47 Moving the PSF display window

3.24 WinPLACE™ Text Editor

The WinPLACE™ text editor (a Wordstar ™ -- like editor) is used in the Design and Compile operations. In the Design operation, the text editor is primarily used for entering or modifying the logic descriptions of the design. In the Compile operation, the editor is interfaced closely with the WinPLACE™ compiler. If a compilation syntax error is encountered, the editor automatically displays the line with the error. If possible, this error can then be analyzed and modified without returning to the Design operation.

Using a mouse in the editor

The WinPLACE™ text editor supports some of the editor commands via the mouse. Right-Click the mouse once in the Text Editor to reveal a window containing six functions. The only way to have access to these functions is by using a mouse. Besides these six functions, the mouse can be used to quickly move the editor cursor to another location. The six functions are explained as follows:

Undo……Reverses the last action that was made in the editor. Whether it be typing or erasing, it will undo it. If you undo an action and then select the undo function again, it will redo the action that you just undid.

Copy……This function will copy the currently highlighted text. This function will not work unless something is highlighted. This option is convenient for quickly writing multiple text of basically the same content.

Cut……This function is almost the same as the “Copy” function however, it clears the text as well as copying it onto a clipboard. This is mainly used for relocating a portion of text without retyping it.

Paste……...This is used to insert the text that was either “Cut” or “Copied” last from the document. This function can be used repeatedly without re-copying or cutting the text. However, when a new portion of text is copied or cut, the last portion is erased from the clipboard.

Delete……..This is simply to clear a portion of highlighted text with using the mouse instead of the keyboard.

Select All….This function will highlight all of the text in the WinPLACE™ Text Editor. This is convenient to either do mass deleting or to copy it to another document.

Sizing the Text Editor

The Text Editor viewer can be changed depending how much needs to be seen of the text. If this function is used in the Design operation, then more will be seen of the Pin-Block Diagram, and if it is used in the Compile operation, then more will be seen of the Compile and Error windows. To perform the sizing, simply place the mouse over the separation bar between the Text Editor and the Pin-Block Diagram. The cursor should change to an up and down double sided arrow. Click-LH and drag the bar up or down depending on how big or small you want the Editor. (SeeFigure 3-48)

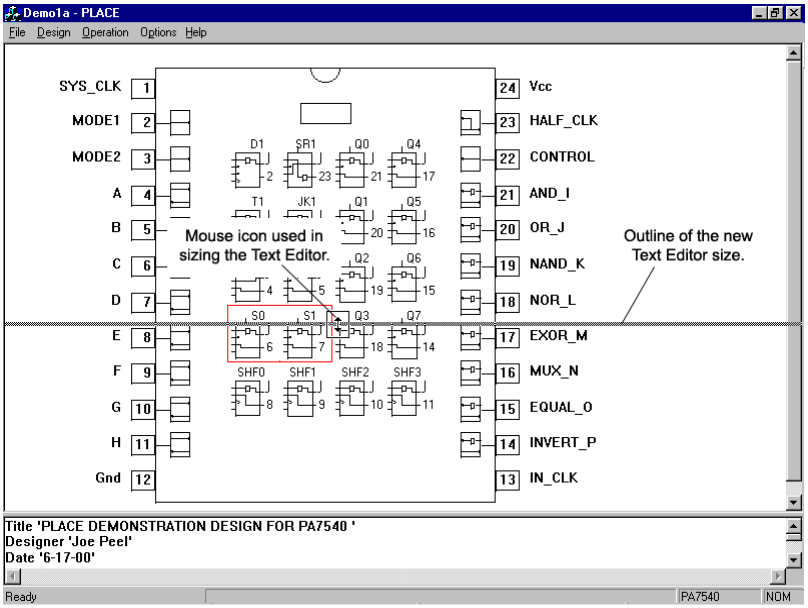


Figure 3-48 Sizing the Text Editor

Paging Up and Down

The contents of the window can be paged up and down by pressing the PgUp and PgDn keys respectively. Paging up and down can also be accomplished by clicking at the scroll bar on the right side of the Editor. Also, if you own a scroll mouse, you can use the scroll button in the middle to go up and down

4 WinPLACE™ Design Language

4.1 Introduction

To simplify the design entry process, the WinPLACE™ software allows control of the architectures graphically. This capability allows the designer to better utilize his or her time on the actual design implementation and not on architectural syntax as in most other PLD software tools. Underneath the graphics however, the WinPLACE™ software incorporates a powerful design language that provides standard behavioral design methods such as State Diagrams, Truth Tables and Equations.

WinPLACE™ Source File Format

Figure 4-1 shows the format of the WinPLACE™ Source File (PSF). Figure 4-2 and Figure 4-3 illustrate the differences between the PSF formats for the PA7540 and PEEL™18CV8 devices. **Note: All format categories in** Figure 4-1 **that are shaded are automatically set-up via the WinPLACE™ architectural editor.**

Design Description:	*Title of the design, name of designer, date and detailed description of the design.*
Device type:	*PA7540, PEEL™18CV8, PEEL™22CV10A, etc.*
Special features:	*Sets Security-bit, Zero-Power or Signature Word.*
Input or Clock Pins:	*Assigns names to the clock and dedicated input pins.*
Cell Configurations:	*PEEL™ Arrays: IOC and LCC Configuration. PEEL™ Devices: IOC (Macro Cell) Configurations.*
Global Configurations:	*PEEL™ Arrays: Group A and B Global Cell Configurations. PEEL™ Devices: Asynchronous and Synchronous node definitions for registered PEEL™ Devices (e.g., PEEL™18CV8, PEEL™22CV10A, etc.).*
Comments	*Details the description of the design.*
Macro Definitions:	*State assignment, equation and constant declarations.*
State-diagrams	
Truth-tables	
Equations	

Figure 4-1 WinPLACE™ Source File Format

While reading the PSF file, the WinPLACE™ software checks the file format for incompatibilities. If any format incompatibilities are found for the selected device type, the WinPLACE™ software will display error messages.

```
TITLE ‘ ‘

Designer ‘ ‘

Date ‘ ‘



Description
   Enter description here…

End_Desc;



PA7540                                  “Device Type



"Optional Special Features
Identifiers AUTO_SECURE                 "Programs the security bit. If
                                        unspecified,
                                        "defaults to Security-bit OFF

SIGNATURE 'ABCDEFGH'                    "Programs Signature Word
                                        'ABCDEFGH'



CLK1 PIN 1                              "Input or Clock pin declaration

CLK2 PIN 13



IOC ( 2 " POS IO ) <- 1A                "IOC Declaration

IOC ( 3 " POS IO ) <- 2A

IOC ( 4 " POS IO ) <- 3A

IOC ( 5 " POS IO ) <- 4A

IOC ( 6 " POS IO ) <- 5A

IOC ( 7 " POS IO ) <- 1B

IOC ( 8 " POS IO ) <- 2B

IOC ( 9 " POS IO ) <- 3B

IOC ( 10 " POS IO ) <- 4B

IOC ( 11 " POS IO ) <- 5B

IOC ( 14 " POS IO ) <- 1C
```

```
IOC ( 15 " POS IO ) <- 2C
IOC ( 16 " POS IO ) <- 3C
IOC ( 17 " POS IO ) <- 4C
IOC ( 18 " POS IO ) <- 5C
IOC ( 19 " POS IO ) <- 1D
IOC ( 20 " POS IO ) <- 2D
IOC ( 21 " POS IO ) <- 3D
IOC ( 22 " POS IO ) <- 4D
IOC ( 23 " POS IO ) <- 5D
LCC (1A " D POS REG REG )          "LCC Declaration
LCC (2A " D POS REG REG )
LCC (3A " D POS REG REG )
LCC (4A " D POS REG REG )
LCC (5A " D POS REG REG )
LCC (1B " D POS REG REG )
LCC (2B " D POS REG REG )
LCC (3B " D POS REG REG )
LCC (4B " D POS REG REG )
LCC (5B " D POS REG REG )
LCC (1C " D POS REG REG )
LCC (2C " D POS REG REG )
LCC (3C " D POS REG REG )
LCC (4C " D POS REG REG )
LCC (5C " D POS REG REG )
LCC (1D " D POS REG REG )
LCC (2D " D POS REG REG )
LCC (3D " D POS REG REG )
LCC (4D " D POS REG REG )
LCC (5D " D POS REG REG )
```

```
Global = 1                          "Number of global cells used

GBC (A Clk1 Clk1)                   "Global Cell A configuration

GBC (B Clk1 Clk1)                   "Global Cell B configuration

DEFINE                              "Macro Definitions

STATE_DIAGRAM SD_name               "State diagram design syntax
END;                                "Ends current State diagram syntax


TRUTH TABLE TT_name                 "Truth table design syntax
END;                                "Ends current State diagram syntax

EQUATIONS                           "Logic equation syntax


"Equations for the Global Cell A

RTA = 0;                            "Reg-Type Product Term

PCLKA = 0;                          "I/O Clock Product Term

PRESETA = 0;                        "Preset Sum Term

RESETA = 0                          "Reset Sum Term


"Equations for the Global Cell B

RTB = 0;                            "Reg-Type Product Term

PCLKB = 0;                          "I/O Clock Product Term

PRESETB = 0;                        "Preset Sum Term

RESETB = 0;                         "Reset Sum Term
```

Figure 4-2 PA7540 "ANEW7540.PSF" File Template

```
TITLE ' '

DESIGNER ' '

DATE ' '


Description
   Enter description here...

End_Desc;
```

```
PEEL™18CV8                          "Device type
```

"Optional Special Features Identifiers AUTO_SECURE	"Programs Security-bit. If unspecified, "defaults to security-bit OFF
CLK1 PIN 1	"Input or Clock pin declaration
IOC (12 " POS COM FEED_PIN)	"I/O or Macro Cell Configuration
IOC (13 " POS COM FEED_PIN)	
IOC (14 " POS COM FEED_PIN)	
IOC (15 " POS COM FEED_PIN)	
IOC (16 " POS COM FEED_PIN)	
IOC (17 " POS COM FEED_PIN)	
IOC (18 " POS COM FEED_PIN)	
IOC (19 " POS COM FEED_PIN)	
AR NODE 21	"Global Asynchronous Reset Node
SP NODE 22	"Global Synchronous Preset Node
DEFINE	"Macro Definitions
STATE_DIAGRAM *SD_name* END;	"State diagram design syntax "Ends current State diagram syntax
TRUTH TABLE *TT_name* END;	"Truth table design syntax "Ends current State diagram syntax
EQUATIONS	"Logic equation syntax
"Equations for the global nodes	
AR = 0;	"Global Asynchronous Reset Equation
SP = 0;	"Global Synchronous Preset Equation

Equation

Figure 4-3 PEEL™18CV8 "ANEWV8.PSF" Template

4.2 Design Description

The design description section of the PSF format is made up of four fields. The fields include: Title of the design, Designer's name, Date of the design, and a detailed description of the design.

In describing the PSF formats for the following sections (including this one), italics will be used to identify fields in which the user would enter identifiers, such as title and date of the design, name of the designer, pin names, etc. The reserved identifiers will be specified in **bold**. Most of the examples used for illustrating the formats (except for the PEEL™ device formats) are taken from the Blackjack Machine Application Example (JACK7024.PSF) illustrated in Section 5.4. **All reserved identifiers and labels are not case sensitive.**

Title

Format: **Title** *'title of design'*

Example: Title 'Blackjack Machine Example'

Designer

Format: **Designer** *'name of the designer'*

Example: Designer 'Joe Peel'

Date

Format: **Date** 'date of design'

Example: Date 'May 10th, 1991'

Description

The Description identifier allows the user to specify in detail the description of the design. The user specifies his or her description within the reserved identifiers "**Description**" and "**End_Desc;**". The WinPLACE™ software automatically inserts these identifiers.

Format:
Description
enter description of design here ...
End_Desc;

Example:
Description
Blackjack Machine Example

This design example was based on C.R. Clare's design in Designing Logic Systems Using State Machines (McGraw Hill, 1972). The blackjack machine plays....

All ASCII characters can be used here.

End_Desc;

All WinPLACE™ reserved words except "End_Desc" can be used within the "Description" and "End_Desc" identifiers. Each line does not need to begin with a double quotation mark, as required in the Comments field (Section 4.9).

4.3 Device Type

The target device of the design is declared by simply entering the ICT PEEL™ device name.

Format: *device_type*

PA7024	**PA7540**	**PA7128**	**PA7536**
PA7140	**PA7572**	**PEEL™16CV8**	**PEEL™18CV8**
PEEL™18CV8Z	**PEEL™18LV8Z**	**PEEL™22CV10A**	**PEEL™22CV10AZ**
PEEL™22LV10AZ			

4.4 Special Features

Special features such as enabling the Security Bit, programming the Signature Word, and setting the Zero Power Bit are available for some of the PEEL™ devices. These features are optional, meaning that they are not required to be specified in the PSF file. If not specified, the default conditions will be implemented. Refer to the description of each of these features for their default conditions.

Security Bit

Once the security bit feature is enabled, the programmed data in the device (except for the Signature Word) is prevented from being loaded or read, and hence prevents any unauthorized copying of the design in the PEEL™ device.

The security bit feature is available for the following devices.

PA7024	**PA7540**	**PA7128**
PA7536	**PA7140**	**PA7572**
PEEL™16CV8	**PEEL™18CV8**	**PEEL™18CV8Z**
PEEL™18LV8Z	**PEEL™22CV10A**	**PEEL™22CV10AZ**
PEEL™22LV10AZ		

Format: **AUTO_SECURE**

The security bit of the device is enabled via the reserved identifier AUTO_SECURE. If this identifier is specified in the PSF file, the WinPLACE™ Compiler will create a JEDEC file with the security bit enabled (sets the "G1" field). In most PLD programmers, the "G1" field automatically enables security bit programming.

Default condition: The AUTO_SECURE identifier is unspecified. The JEDEC file generated will not have the "G1" field. In most PLD programmers, the user can enable or disable security bit programming.

Signature Word

The signature word of the device allows a user to enter a design revision number so that the design can be identified after the security bit of the PEEL™ device is enabled. Hence, the signature word data can still be loaded even after the security bit of the device is enabled.

The signature word feature is supported in the following devices. Note that the number of 8-bit bytes in the signature word is specified within the parenthesis.

PA7024 (8 bytes)
PA7540 (8 bytes)
PA7128 (1 byte)
PA7536 (1 byte)
PA7140 (2 bytes)
PA7572 (2 bytes)
PEEL™18CV8Z (8 bytes)
PEEL™22CV10A+ (3 bytes)
PEEL™22CV10AZ+ (8 bytes)
PEEL™16CV8 (8 bytes)
PEEL™18LV8Z (8 bytes)

Format: **SIGNATURE** 'signature str'

Example: Signature 'REV. A'

Default condition: SIGNATURE identifier is unspecified, which means that the signature word in the device JEDEC file is unused.

4.5 Clock and Input Pins

After labeling a clock or a dedicated input pin (a pin that is not associated with an Input Cell or INC) of the device using the "**Label**" command in the Design operation, the WinPLACE™ software automatically creates the pin assignment statement.

Format: pin_label **PIN** pin_number

Example: CLK1 pin 13

Please refer to Section 4.6 for the pin label format.

Default condition: An unlabeled pin (no pin assignment statement) signifies that the pin is unused.

4.6 Pin and Cell Labels

Format: First character: **A...Z, a...z, ~, /, !**
Body of the label: **A...Z, a...z, 0...9, ~, _**

Examples: Valid labels: Addr10, ~10, /OUT
Invalid labels: _Add, 25MHz, /15IN

The label is not case sensitive. The maximum length of the label is 8 characters (including the / or ! character). **When a / or ! character is added at the beginning of the label, the pin, cell or node becomes an active Low signal path.** Hence, a TRUE logic (logic "1") results when a Low signal is applied.

Example:

```
/A pin 1          "Active Low Input
B pin 2  "Active High Input

IOC (12 'C' Pos COM Feed_pin)        "Output Polarity = Pos
IOC (13 'D' Neg COM Feed_pin)        "Output Polarity = Neg
IOC (14 '/E' Pos COM Feed_pin)       "Output Polarity = Pos
IOC (15 '/F' Neg COM Feed_pin)       "Output Polarity = Neg

EQUATIONS
C.COM = A;              "C=TRUE or 1 when A=LOW
D.COM = B;              "D=TRUE or 1 when B=HIGH
```

The / or ! on the pin or cell labels only affect the active level of the inputs or feedback paths (i.e. variables on the right side of the equal sign in the equations). The polarity of the outputs (i.e. outputs routed to the external pins) is not affected because they are controlled by the IOC configuration statements. In the above example, the feedback active levels and output polarities of cells C, D, E, and F are:

Cell	Feedback Active Level	Output Polarity
C	High	High
D	High	Low
/E	Low	High
/F	Low	Low

4.7 Cell Configurations

The cell configuration format statements are used to specify the type of configuration of each cell in the selected device. In most cases, knowledge of the cell configuration formats is not necessary because the configurations of the IOC and LCC are automatically modified by the WinPLACE™ architectural software.

Note that all the configuration statements are necessary for the operation of the WinPLACE™ software. This means that **you should not delete any of these configuration statements** including the configuration statements for unused cells.

Format:

Input Cell in PA7128, PA7536, PA7140, and PA7572:

INC (pin_number *'pin_label'* input_type)

Example: INC (3 'A1' Reg)

I/O Cell in PEEL™ Arrays:

IOC (pin number *'pin_label'* output_pol pin_type) ← Assigned_LCC

Example: IOC (4 'V4' Pos IO) ← 3A

Logic Control Cell in PEEL™ Arrays only:

LCC(cell_number 'cell_label' flip-flop_type clock buried_out ext_out)

Example: LCC (1A 'ADD10' D SumC Reg Reg)

I/O Cell (or Macro Cell) in PEEL™ devices:

IOC (pin_number 'pin_label' output pol pin_type feedback_type)

Example: IOC (12 'OUT' POS COM FEED_PIN)

Default condition: The cell configuration statements in the "ANEWxxxx.PSF" files set the default cell configurations. If the "**New**" function under the File menu command in the Design operation is selected, the WinPLACE™ software reads the ANEW file for the selected device (see Table 4-1) and sets the default configurations found in the file.

ANEW File	Device
ANEW7024.PSF	PA7024
ANEW7540.PSF	PA7540
ANEW7140.PSF	PA7140
ANEW7572.PSF	PA7572
ANEW7128.PSF	PA7128
ANEW7536.PSF	PA7536
ANEW168S.PSF	PEEL™16CV8S
ANEW168C.PSF	PEEL™16CV8C
ANEW168R.PSF	PEEL™16CV8R
ANEWV8.PSF	PEEL™18CV8
ANEWCV8Z.PSF	PEEL™18CV8Z
ANEWV8Z.PSF	PEEL™18LV8Z

ANEWV10A.PSF	PEEL™22CV10A
ANEWV10P.PSF	PEEL™22CV10A+
ANEWVAZP.PSF	PEEL™22CV10AZ+

Table 4-1 WinPLACE™ ANEW Template Files

Parameters for the INC Format (PA7536, 7572, 7128 and 7140 only)

Pin_Number — The pin number that is assigned to the current Input cell.

Device	INC Pin Numbers
PA7128, PA7536	2-6, 8-14
PA7140 (PLCC), PA7572	3-5,19-21, 25-27, 41-43
PA7140 (DIP), PA7572	2-4,17-19, 22-24, 37-39

Pin_Label — See Section 4.6 for the format of the pin label.

Input_type — The identifiers for the pin type parameter are:

Identifier	Function
COM	Combinatorial input
REG	D-type registered input
LAT	D-type latched input

Parameters for the IOC Format (PEEL™ Arrays only)

Pin_Number — The pin number that is assigned to the current I/O cell.

Device	IOC Pin Numbers
PA7024, PA7540	2-11, 14-23
PA7128, PA7536	15-20, 22-27
PA7140 (PLCC), PA7572	6-16, 18, 28-38, 40
PA7140 (DIP), PA7572	5-16, 25-36

Pin_Label — See Section 4.6 for the format of the pin label.

Output_pol — This parameter, which refers to the output polarity of the pin is controlled by the following identifiers:

The output polarity "bubble" in the Design operation of the WinPLACE™ software controls this parameter. Inserting the / or ! character in the pin_label does not affect the output (Section 4.6).

Identifier	Function
POS	Positive Polarity for the Output
NEG	Negative Polarity for the Output

Pin_type

The identifiers for the pin type parameter are:

Identifier	Function
I/O	I/O
REG	I/O with D-type registered input
LAT	I/O with D-type latched input
OUT	Output only
INCOM	Input only
INREG	Input only with D-type register
INLAT	Input only with D-type latch
OUTREG	Output only with D-type registered feedback
OUTLAT	Output only with D-type latched feedback
DCOM	Output only with feedback from Sum-D
DREG	Output only with D-type registered feedback from Sum-D
DLAT	Output only with D-type latched feedback from Sum-D

The parameters DCOM, DREG and DLAT are only applicable for PA7128, PA7536, PA7140, and PA7572 devices.

Assigned_LCC

The LCC that is connected to the current IOC.

Parameters for the LCC Format (PEEL™ Arrays only)

Cell_number

The cell number that is assigned to the current Logic Control cell. It ranges from 1A-6A, 1B-6B, 1C-6C and 1D-6D. Figure 4-4 illustrates the cell number organization for the PA7540 device.

Device	LCC Assignments
PA7024, PA7540	1A-5A, 1B-5B, 1C-5C, 1D-5D
PA7128, PA7536	1A-3A, 1B-3B, 1C-3C, 1D-3D
PA7140 (PLCC), PA7572	1A-6A, 1B-6B, 1C-6C, 1D-6D

Refer to the IOC statement described in the above for the assigned IOC for each LCC.

Cell_label

Refer to Section 4.6 for the cell label format.

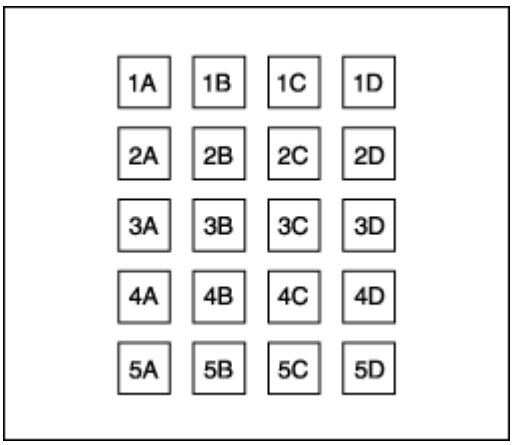


Figure 4-4 LCC numbering system in PA7024 & PA7540

Flip-flop type

Specifies the type of register in the LCC. The dynamic register type setting (RT signal) is also specified in this parameter. The RT signal which comes from the Global Cell dynamically changes the register type in the LCC during normal (5V) operation.

Identifier	Function
D	D-type register (RT mode is disabled)
T	T-type register (RT mode is disabled)
JK	JK-type register (RT mode is disabled)
DT	D-type register when RT = FALSE, T-type register when RT = TRUE
DJK	D-type register when RT = FALSE, JK-type register when RT = TRUE
TD	T-type register when RT = FALSE, D-type register when RT = TRUE
JKD	JK-type register when RT = FALSE, D-type register when RT = TRUE

Clock

Controls the type of clock for the current LCC.

Identifier	Function
POS	High speed (system) clock from an Input/Clock pin that triggers the register on the rising edge. Note: Global cell controls, which pin to use for system clock.
NEG	Register is triggered on the falling edge of the system clock.
SumC	Local clock coming from the Sum C term. Register is triggered on the rising edge of the clock signal.
SumD	Local clock coming from the Sum D term. Register is triggered on the falling edge of the clock signal.

Buried_Out — Output of the LCC that is fed back internally to the array.

Identifier	Function
REG	Internal output from the register
SumA	Internal output from the Sum A term (i.e. the input of the register)
SumB	Internal output from the Sum B term
SumC	Internal output from the Sum C term

Ext_Out — Output of the LCC to the I/O cell. This output sends the signal from inside the device to the outside world. The final output signal depends on the output polarity of its assigned IOC.

Parameters for the IOC Format (PEEL™ Devices)

Pin_Number — The pin number that is assigned to the current I/O cell.

Pin_Label — See Section 4.6 for the format of the I/O pin label.

Output_pol — This parameter which refers to the output polarity of the pin is controlled by the following identifiers:

Identifier	Function
POS	Positive Polarity for the Output
NEG	Negative Polarity for the Output

The output polarity "bubble" in the WinPLACE™ architectural software controls this parameter. Inserting the / or ! character in the pin_label does not affect the output (Section 4.6). Hence, only this parameter controls the polarity of the output.

Pin_type

The identifiers for the pin type parameter are:

Identifier	Function
COM	I/O with combinatorial output
REG	I/O with D-type registered output
OUTCOM	Combinatorial output only
OUTREG	Registered output only
IN	Output disabled (may or may not be an input depending on the feedback type)

Feedback type

Specifies the type of feedback for the selected device. This parameter is applicable for all PEEL™ Devices except the 22CV10A, which follow the industry-standard architectures. Note that the feedback selection is available on the 22CV10A+. The feedback types of the 22CV10A are automatically set to FEED_PIN and FEED_REG for COM and REG output types respectively.

Identifier	Function
FEED_PIN	Feedback from the pin
FEED_REG	Feedback from the output of the register
FEED_OR	Feedback directly from sum term (i.e. prior to the register)

4.8 Global Configurations

PEEL™ Arrays

The global configurations for PEEL™ Arrays are used to set signals for the LCC's, IOC's and INC's.

Format:

```
Global = n                          "n = 1 or 2 cells
GBC (A LCC_clock IOC_clock)         "Global Cell A
GBC (B LCC_clock IOC_clock)         "Global Cell B
GBC (C ___ INC_clock)               "Global Cell C
                                    "(PA7536, PA7140
                                    "PA752 and PA7128)
```

Example:

```
Global = 1
GBC (A Clk1 Clk2)
```

GBC (B Clk1 Clk2)
GBC(C ___ Clk1)

Table 4-2 shows the definitions of the Clk1, Clk2 and PClk terms used in the GBC configuration statements.

Clk1 and Clk2 terms -- specify the system clock pin to use.

Device	Clk1 pin	Clk2 pin
PA7024, PA7540 (DIP,	1	13
PA7024, PA7540 (PLCC)	2	16
PA7128, PA7536	1	28
PA7140, PA7572 (PLCC)	2	24
PA7140, PA7572 (DIP)	1	21

PClk terms -- specify to use the product term for the global clock.

Device	IOCs by PClkA	IOCs by PClkB	INCs by PClkC
PA7024, PA7540 (DIP, SOIC)	2-11	14-23	None
PA7024, PA7540 (PLCC)	3-7, 9-13	17-21, 23-27	
PA7128, PA7536	15-20, 22-27		2-6, 8-14
PA7140, PA7572 (PLCC)	6-16, 18	28-38, 40	3-5, 19-21, 25-27, 41-44
PA7140, PA7572 (DIP)	5-16	25-36	2-4, 17-19, 22-24, 37-39

Table 4-2 Definitions of global clock terms

Global = n — The n parameter equals 1 or 2. This parameter sets the number of global cells to be used for the LCC's and IOC's. If one global cell is used, then global cell A controls the global signals for all LCC's and IOC's in the device and the global cell B configuration statement is ignored. If "Global = 2" is specified, then Global Cell A controls the global signals for the IOC's and LCC's in which the IOC's are on the left side of the device. Global Cell B then controls the global signals for the IOC's and LCC's in which the IOCs are on the right side of the device. See Table 4-2.

Below are the descriptions of each parameter in the GBC (GBC A, B or C) configuration statements.

LCC_clock — Sets the system clock pin for the LCC global clock. The two options available are the Clk1 and Clk2 pins. Refer to Table 4-2 for the Clk1 and Clk2 pin numbers for each PEEL™ Array.

IOC_clock — Sets the system clock pin or product term for the IOC global clock. The options available are Clk1, Clk2, PCLKA and PCLKB. See Table 4-2.

INC_clock — Sets the system clock pin or product term for the INC global clock. The options available are Clk1, Clk2 and PCLKC. See Table 4-2.

PEEL™ Devices

In the registered PEEL™ devices such as the PEEL™18CV8, 22CV10A, and 22CV10A+, the global configurations are represented by the global node assignments. These global nodes control the asynchronous reset and synchronous preset product terms.

Format: node_label **NODE** node_number

Example: AR node 21 Asynchronous reset for PEEL™18CV8 device

node_label — Please refer to Section 4.6 for the node label format.

node_number — The node assignment numbers for the selected device are:

Device	Node #	Function
PEEL™18CV8	21	Asynchronous Reset (AR)
	22	Synchronous Preset (SP)
PEEL™22CV10A PEEL™22CV10A+	25 26	Asynchronous Reset (AR) Synchronous Preset (SP)
PEEL™22CV10AZ+	25	Asynchronous Reset (AR)
	26	Synchronous Preset (SP)
	27	P-Term Clock (CS)

4.9 Comments

In the WinPLACE™ software, comments are available so that each component of the design, which may not be readily apparent from the source file, is explained. Comments do not affect the design itself. Liberal use of comments can make a PSF design file easy to understand.

Format: "*Insert comments here…*

Example: "Enable security bit programming

A comment begins with a double quotation mark (") and ends with the end of line. A comment can be specified anywhere in the PSF design file.

Note that the double quotation marks are not required if the comments are specified within the "DESCRIPTION" and "END_DESC" reserved identifiers.

4.10 Macro Definitions

The macro definitions are used for:

-- declaring constants which make the design easier to understand

-- declaring commonly used equations so they don't have to be repeated throughout the design file

-- assigning the state cells and set variables for state diagram designs

-- assigning the pins or cells for truth table designs

The macro definitions in the PSF design file are located after the reserved identifier **DEFINE** but prior to one of the following reserved identifiers: **STATE_DIAGRAMS**; **TRUTH_TABLE**; or **EQUATIONS** (whichever is specified first). Macro definitions that are specified via the "Macro" function in the Design operation are automatically inserted into this location.

Format: **DEFINE**
specify macro definitions here...

STATE_DIAGRAMS, TRUTH_TABLE or **EQUATIONS**

Example:

```
DEFINE
QSTATE = [ADD10 SUB10 Q2 Q1 Q0]     "State_Diagram Assignment
Clear       = ^B00000               "Constant Declaration
ShowHit     = ^B00001
AddCard     = ^B11011
Add_10      = ^B10010
Wait        = ^B00010
Test_17     = ^B00110
Test_22     = ^B00111
ShowStand   = ^B00101
ShowBust    = ^B00100

is_Ace = !V4 & !V3 & !V2 & !V1 & V0;    "Equation Declaration

SCORE = [S4 S3 S2 S1 S0]            "Truth table input assignment
BCD2 = [D5 D4]                      "Truth table output assignment
BCD1 = [D3 D2 D1 D0]                "Truth table output assignment
STATE_DIAGRAM QSTATE                "Ends the Macro Definitions
```

Macro Constants

Format: Const_label = constant

Examples: Clear = ^B00000
ShowHit = ^B00001

Const_label The format for the label of the constant is similar to that of the pin or cell label (see Section 4.6) with two exceptions, and they are:

1. The length of the label can be up to 20 characters long instead of 8.
2. 2. The / or ! character cannot be used at the beginning of the label.

Constant Specifies the value of the constant in decimal, hexadecimal, octal or binary numbering system.

The format for the constant is:
symbol + number

The symbols for the numbering systems are:

Numbering system		Symbol
Decimal		none (default)
Hexadecimal		^H or ^h
Octal		^O or ^o
Binary		^B or ^b
Examples:	15	(decimal)
	^HF	(hexadecimal)
	^O17	(octal)
	^B1111	(binary)

Macro Equations

Format: Eqn_label = complex_eqn

Example: Is_ACS = !V4 & !V3 & !V2 & !V1 & !V0;

Eqn_label The label for the equation macro is similar to that for the label for the macro constant. See above section.

Complex_eqn Macro equation can be specified using the logic operators (), !, &, # and $ (refer to Section 4.13).

The input side of a macro equation (i.e. the right side of the "=" symbol) is made up of pin or cell labels, or labels from other macro equations. Below is an example of a macro equation, which is a function of other macro equations.

```
DEFINE

Mac1 = A & B;          "Macro level 1
Mac2 = C # D;          "Macro level 1
Mac3 = Mac1 $ Mac2;    "Macro level 2
```

If the macro equation uses only pin and/or cell labels, then it has one macro level. If it uses **previously defined** macro equation labels in addition to the pin and cell labels, then it has multiple macro levels. The number of macro levels depends on whether the macro equation labels used in the equation are functions of more macro equation labels themselves.

Macro State Cell Assignments for STATE DIAGRAMS

The state cell assignment defines the pin or cell labels to be used by the state diagram design syntax.

The "**Allocate**" command in the "Design" menu window of the Design operation automatically generates the state cell assignment definition and STATE_DIAGRAM design syntax. An example is shown below.

```
DEFINE
QSTATE = [ADD10 SUB10 Q2 Q1 Q0]
```

```
STATE_DIAGRAM QSTATE
              "enter design here
END;
```

Format: state_label = [Cell1 Cell2 Cell3 ... Celln]

Example: QSTATE = [ADD10 SUB10 Q2 Q1 Q0]

State_label The state label format is similar to that of the pin or cell label (Section 4.6). The only exception is that the state label does not allow the use of the / or ! character at the beginning of the label.

Cell(n) Specifies the cell labels to be used as state cells by the state diagram. A maximum of 24 cells can be allocated as state cells. Each state cell label must be separated by a space. The most and least significant bit of the state cells are the first and last allocated cells respectively in the cell assignment definition, i.e. Cell1 is the Most Significant Bit and Celln is the Least Significant Bit.

Macro Set Variables in State Diagrams

Outputs in a state diagram can be assigned to a set variable so that the logic of these outputs can be specified with a numeric constant. The constant can be specified in a binary, octal, hexadecimal or decimal (default) numbering system.

Format: *set_var = [Cell1 Cell2 Cell3 ... Celln]*

Example:

```
DEFINE
  TEST = [T2  T1]                 "State cell assignment
  OUT = [Y3  Y2  Y1  Y0]          "Set variable assignment


STATE_DIAGRAM TEST
  State 0:      OUT = 0;          "Macro Set variable OUT
                                  "Y3-0= 0000
                Goto 1;
  State 1:      OUT = ^HB;        "Y3-0= 1011
                Goto 2;
  State 2:      OUT = ^B1100;     "Y3-0= 1100
                Goto 3;
  State 3:      Goto 0
END;
```

Set_var The format is similar to the state label format used for the state cell assignments.

Celln Specifies the assigned pin or cell label.

The macro set variable equation feature is only available within the state diagram design syntax. Also, **only one macro level is available in the macro set variable assignment.**

Macro Cell Allocation for Truth Tables

The "**Allocate**" command in the WinPLACE™ Design operation can also be used to allocate the pins and cells for the truth table design. The pins and cells can be allocated as truth table inputs, truth table outputs or both. If a

cell or pin is allocated as the truth table input and output, then it is an I/O. The output of the I/O is then enabled or disabled via the .OE equation.

The following example shows the allocation of the pins and cells for the truth table design "TABLE1".

Example:
```
TRUTH_TABLE TABLE1
(I4  I3  I2  I1  I0  ->  Y5  Y4  Y3  Y2  Y1  Y0)
END;
```

Another method of allocating the pins and cells is through the macro set variable method. The inputs and outputs of the truth table can be assigned to the macro set variables in the DEFINE section. The labels of these macro set variables are then used in the truth table design syntax instead of the pin and cell labels.

Format: *table_label = [Cell1 Cell2 Cell3 ... Celln]*

Example:
```
DEFINE
Input = [I4  I3  I2  I1  I0];
Y_HiBit = [Y5  Y4];
Y_LoBit = [Y3 Y2  Y1  Y0];

TRUTH_TABLE TABLE1
(Input -> Y_HiBit Y_LoBit)
^H15   ->   1   ^HA   "I4-0= 10101,   Y5-0=011010
^H16   ->   1   ^HB   "I4-0= 10110,   Y5-0 = 011011
END
```

Table_label The format is similar to the state label used for the state cell assignments.

Celln Specifies the assigned pin or cell label.

Please refer to Section 4.12 on "Truth Table Design Syntax" for more information.

Macro Counter Function

COUNTERF is the macro function for designing an up-down, loadable counter. It must be specified in the DEFINE section.

Format COUNTERF *(Load, Updown, cBits, Type, iBits)*

Example:

```
DEFINE

UPDN_BITS = [Q4 Q3 Q2 Q1 Q0];      "Counting Bits. Q[4-0] can be
                                   “pin or cell labels.

LOAD_EN = A & B;                   "Load Control Equation. Loads
                                   “when A = TRUE & B = TRUE.

LOAD_BITS = [L4 L3 L2 L1 L0];      "Load Inputs, either pin or cell
                                   “labels
UPDN_CON = C & !D;                 "Updown Control. If C=TRUE,
                                   “D=FALSE, it is DOWN
Counter
                                   “else, it is Up Counter.
```

COUNTERF(LOAD_EN, UPDN_CON, UPDN_BITS, T, LOAD_BITS);

EQUATIONS

Load: 0 = No load function
Pin_Name, Node_Name (Cell_Names),
Macro_defined_label = load control

Updown: 0 = Up Count
1 = Down Count
Pin_Name, Node_Name (Cell_Names),
Macro_defined_label = updown control
(True logic for Up Count)

cBits: A Macro-Set-Variable with at least 2 elements. This set contains the counting bits which include Pin_Names or Node_Names (Cell_Names).

Type: D = D-type counter
T = T-type counter (uses fewest # of product terms)
(The pins or cells assigned to the macro label in the "cBits" parameter must have the same register type.)

iBits: 0 = Non-loadable counter
Macro-Set_Variable = Data bits for loading the counter. This set must have the same number of elements as the cBits. The elements include Pin_Names or Node_Names (Cell_Names). If load=0, then iBits=0.

4.11 State Diagrams

The state diagram language is used to implement state machine designs. In the WinPLACE™ software, state machine designs are specified between the "**STATE_DIAGRAM *state_label***" and "**END**" identifiers. These identifiers together with the state cell allocation definition (refer to Section 4.10) are automatically created when the LCC's or IOC's are allocated for the state machine via the "**Allocate**" command in the Design operation.

Format: **DEFINE**
State_label = [Cell1 Cell2 ... Celln]

STATE_DIAGRAM *State_label*
STATE *state_0:* "usually Reset state
STATE *state_1:*
.
.
.
STATE *state_n:* "last state
END;

State_label The state label format is similar to that of the pin or cell label (See Section 4.6). The only exception is that the state label does not allow the use of the / or ! character at the beginning of the label.

Celln Specifies the cell labels to be used as state cells by the state diagram. A maximum of 24 cells can be allocated as state cells. Each state cell label must be separated by a space. **The most and least significant bit of the state**

cells are the first and last allocated cells respectively, i.e., Cell0 and CelIn are MSB and LSB respectively.

State_n Specifies the state number which can be in the form of a numeric value or a constant label defined in the DEFINE section. Refer to "Macro Constants" in Section 4.10.

The WinPLACE™ state diagram syntax includes:

-- **GOTO**

-- **IF-THEN-ELSE**

-- **WITH-ENDWITH;**
(used in conjunction with GOTO and IF-THEN-ELSE)

GOTO

Format: **GOTO** *state_num;*

Examples: GOTO ShowHit; or
GOTO ^B00001;

State_num Specifies the state number for the unconditional jump. A numeric representation of the state or a constant label defining the numeric value in the macro definition section can be used to indicate the state number.

The GOTO statement is used to unconditionally jump to a different state on the next clock edge.

IF-THEN-ELSE

Unlike the GOTO statement, the IF statement provides a conditional jump to the next state. If the condition is satisfied, the logic jumps to the state specified after the THEN identifier. If the condition is not satisfied, the ELSE state will be the next state.

Format: **IF** condition **THEN** state_num1
ELSE state_num2;

Example: if (!CARDIN) then AddCard
else ShowHit;" ^B00001 can be used
"instead of ShowHit.

Condition A boolean expression condition which can be in the form of a macro equation label (see "Macro Equations" in Section 4.10).

State_num1 If condition is satisfied, then jump to this state on the next clock edge.

State_num2 If condition is not satisfied, then jump to the alternate state.

WITH-ENDWITH

The WITH statement is used in conjunction with the GOTO or IF-THEN-ELSE statements. It allows outputs to be specified so that they use the same clock edge (rising or falling clock edge dependent on the

configuration) that triggers the next state. **ICT recommends using only registered outputs with the WITH statement.**

Format: **WITH**
registered output equations ...
ENDWITH;

Example:
```
if (A & B) then 1 with
                    C = IN; "C is a D-type registered output
                    D = 1; "D is a T-type registered output
                      endwith;
else 2;
```

In the above example, when the expression (A & B) is true jump to state 1 on the next clock edge. Using the same clock edge, the output C latches the data from the input IN. Also, output D will toggle (since it is a T-type register) on same clock edge. If the condition (A & B) is not satisfied, the logic jumps to state 2 without changing the signals on outputs C and D.

Register Types of the "allocated" state cells

Prior to entering your state diagram design syntax, the outputs of the allocated LCCs and lOCs must be configured as registered outputs. The type of registers used, whether they are D, T or JK type registers, affect the behavior of the state diagram. An example is shown in Figure 4-5.

Outputs of the State Diagram

The two types of outputs in the WinPLACE™ state diagram are the synchronous and asynchronous outputs.

```
DEFINE
ST_TEST = [S2 S1 S0]

STATE_DIAGRAM ST_TEST
     State 0: goto 1;
     State 1: case
                A&B&C: 0;
                A&B&/C: 3;
                /A&/B&/C: 4;
                END;
State 2: Goto 5;
.
.
.
State 7: Goto 0;
END;
```

All conditions in this case statement of state 1 failed. Hence, the equations for the state cells S0, S1 and S2 are: S0=0; S1=0; S2=0;

Question:
What is the next state?

If D-type registers are used for S0, S1, and S3 state cells:	**If T-type registers are used for S0, S1, and S3 state cells:**
Answer: Output of D-type registers follow the input on the next clock edge. Hence after clocking the S0, S1, and S2 registers, the outputs equate to 0 which is the condition of state 0. **So, state 0 is the next state.**	Answer: Outputs of T-type registers follow the previous state on the next clock edge if the inputs are FALSE or “0”. Hence after clocking the S0, S1, and S2 registers, the outputs follow the previous state of each register. **So, state 1 is the next state.**

Figure 4-5 State diagrams with D and T type registers

-- **Synchronous Outputs:** These are registered outputs which use the same clock as the state machine. The outputs follow the input data on the next clock edge.

-- **Asynchronous Outputs:** These are combinatorial outputs. The outputs follow the input data immediately.

Note that the state diagram outputs must first be configured using the WinPLACE™ architectural software. For instance if a registered output is required, the type of flip-flop (D, Tor JK) and clock (pin or sum term) must be configured in the LCC/IOC or IOC screen in the Design operation.

The two classes of state machine designs that can be created using the WinPLACE™ software are the Mealy and Moore machines. Both of these state machine designs can utilize synchronous and asynchronous outputs.

Mealy Machine

A Mealy state machine is defined as having outputs which are a function of two sets of variables:

-- the present input conditions

-- the present state of the machine

Examples:	Me_Reg = INPUT;	"Registered output
	Me_Com = INPUT;	"Combinatorial output

Moore machine

A Moore state machine is defined as having outputs which are strictly a function of the state of the machine.

Examples:	Mo_Reg = 0;	"Registered output
	Mo_Com = 0;	"Combinatorial output

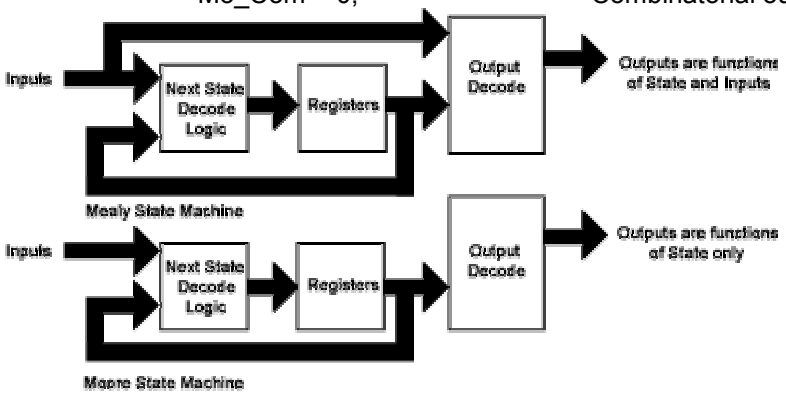


Figure 4-6 Mealy and Moore State Machines

How the WinPLACE™ State Diagram works

Figure 4-7shows a state diagram example (SDEXAMPL.PSF) using the WinPLACE™ state diagram language. This example does not implement any specific application except to illustrate the usage of the state diagram language. The features that are illustrated in the example are:

-- GOTO, IF-THEN-ELSE and WITH-END WITH syntax.

-- Synchronous and asynchronous outputs in Mealy and Moore state machines

-- Set Equations for Moore Machine Applications (refer to Section 4.10) on "Macro Definitions"

4.12 Truth Tables

In addition to state diagrams and equations, truth tables can be used to describe the logic designs.

Format 1: **TRUTH_TABLE** *table_label*
(In1 In2 ... InN -> Out1 Out2 ... OutN)

END;

Format 2: **TRUTH_TABLE** *table label*
(Input -> Output)

END;

```
DEFINE
EXAMPLE = [S1 S0]                    "State cell assignment definition
Grp_Out = [OUT3 OUT2 OUT1 OUT0]      "Group outputs assignment
St0 = 0;                             "default - Decimal
St1 =^H1;                            " ^H - Hexadecimal
St2 = ^O2;                           " ^O - Octal
St3 = ^B11;                          " ^B - Binary

In0 = /I2 & /I1 & /I0                "Input conditions for the State Diagram
In1 = /I2 & /I1 & I0
In2 = /I2 & I1 & /I0
In3 = /I2 & I1 &I0
In4 = I2 & /I1 & /I0
In5 = I2 & /I1 & I0
In6 = I2 & I1 & /I0
In7= I2& I1 & I0
In4T06 = In4 # In5 # In6
```

Figure 4-7 WinPLACE™ State Diagram Language for SDEXAMPL Design Example

```
STATE_DIAGRAM EXAMPLE


DEFINE
"Goes to STATE St0 upon device power-up (all registers reset on power-up)
  STATE St0:
```

```
Mo_Reg = 0;                    "Moore registered output
Mo_Com = 0;                    "Moore combinatorial output
Grp_Out = 0;                   "Moore group combinatorial output,
                               Out3-0 = 0000
Me_Reg = INPUT;                "Mealy register output
Me_Com = INPUT;                "Mealy combinatorial output


IF In1 THEN St1                "If In1=true, then go to STATE St1,
ELSE St0                       "else remain at STATE St0.

  STATE St1:
      Grp_Out = ^B0110;        "Out3-0 = 0110. These combinatorial
                               outputs will be valid after the present
                               state occurs with a single
                               propagation delay (tpd).
      Me_Reg = INPUT;          "This registered output will be valid
                               on the next clock edge, i.e. clock
                               edge for the NEXT STATE (St0, St1
                               or St2 depending on which CASE
                               condition is satisfied).
      Me_Com = INPUT;          "This combinatorial output will be
                               valid after the present
                               "state occurs with a single
                               propagation delay (tpd).
  CASE                         "Go to St0 if (/I2&I1&/I0)=In2=true
      /I2 & I1 & /I0: St0
          WITH                 "This output equals to 1 on the next
              Mo_Reg = 1;      clock edge only if the NEXT STATE
                               is St0.
          ENDWITH;
      In3: ST2;                "Go to St2 if In3=true
      ELSE
      St1                      "If no condition in the CASE list is
                               satisfied, remain at STATE St1.
  ENDCASE;
  STATE St2:
```

Figure 4-8 WinPLACE™ State Diagram Language for SDEXAMPL Design Example

```
Grp_Out = ^HA; "Out3-0 = 1010. These combinatorial outputs will be valid
after the "present state occurs with a single propagation delay (tpd).
IF (In4To6) THEN St3           “If (In4To6)=true, then go to STATE
                               St3
          WITH
              Me_Reg = INPUT;  "This registered output will be valid
                               on the next clock edge only if the
                               NEXT STATE is St3.
```

```
            ENDWITH;
  ELSE
        St2;                          "If (In4To6)=false, then remain at
                                      STATE St2.
  STATE St3:                          "This output equals 1 on the next
      Mo_Reg = 1;                     clock edge.  This combinatorial
      Mo_Com = 1;                     output will be valid after the present
  Grp_Out = 0;                        Out 3-0 = 0000.
  Me_Reg = 0;                         "Reset all Mealy outputs
  Me Com = 0;


GOTO St0;                             "Go to STATE St0 unconditionally.


END;                                  "End of STATE DIAGRAM
                                      EXAMPLE
```

Figure 4-9 WinPLACE™ State Diagram Language for SDEXAMPL Design

Example 1:

```
TRUTH_TABLE DECODE "3-to-8 Decoder

(C   B   A   ->   Y0   Y1   Y2   Y3   Y4   Y5   Y6   Y7
0    0   0   ->   0    0    0    0    0    0    0    0
0    0   1   ->   0    1    0    0    0    0    0    0
0    1   0   ->   0    0    1    0    0    0    0    0
.
END;
```

Example 2:

```
DEFINE
SCORE = [S4 S3 S2 S1 S0]
BCD2 = [D5 D4]
BCD1 = [D3 D2 D1 D0]

TRUTH_TABLE BIN2BCD "From JACK7024.PSF

 ( SCORE        ->        BCD2        BCD1)
     0          ->          0           0;
     1          ->          0           1;
     2          ->          0           2;
.
END;
```

Table_label — The format for the label is similar to that for the pin or cell label (Section 4.6) with one exception, and that is the / or ! character is not allowed at the beginning of the label.

InN (Format 1) — Specifies the pin or cell labels to be used as truth table inputs. The inputs can be either registered or combinatorial. Input data must be in binary (0 or 1) format.

OutN (Format 1) — Specifies the pin or cell labels to be used as truth table outputs. Like the inputs, truth table outputs can be

either registered or combinatorial. Output data must be in binary (0 or 1) format.

Inputs (Format 2) Specifies a macro defined group of registered or combinatorial pins or cells to be used as truth table inputs. Input data can be in decimal (default), hexadecimal (^H or ^h), octal (^O or ^o) or binary (^B or ^b) numbering system. Only one macro level is available for the macro input set.

Outputs (Format 2) Specifies a macro define group of registered or combinatorial pins or cells to be used as truth table outputs. Input data can be in a decimal (default), hexadecimal (^H or ^h), octal (^O or ^o) or binary (^B or ^b) numbering system. Only one macro level is available for the macro output set.

An additional feature is that both the truth table formats can be used in a single truth table design.

Alternate truth table description, which uses both formats for Example 2 is:

```
TRUTH_TABLE BIN2BCD

    ( SCORE      ->        D5        D4        BCD1)
        0        ->        0         0         0;
        1        ->        0         0         1;
        2        ->        0         0         2;
.
.
END;
```

4.13 Equations

The boolean logic equations are the primary methods for specifying logic functions in the WinPLACE™ software.

The WinPLACE™ architectural software automatically creates the equation for each sum or product term in the cell when it is labeled via the "Label" command in the "Design" menu of the Design operation. An example of a newly labeled LCC is shown below.

```
A.D = 0;
A.AP = 0;
A.AR = 0;
A.CLK = 0;
```

Format: Output_label.EXT = logic_equation;

(The semicolon at the end of the equation is used by the WinPLACE™ software to mark the end of the equation when displaying the equation in the Equation Display window. Refer to "PSF Text Display Windows" in Section 4.12.)

Examples:

```
C1.COM = (V0 & Add10 & Sub10 & S0);
S0.T = (V0 & Add10 & Sub10);
XOR1.COM = A $ B;
XOR2.COM = (!A & B) + (A & !B);
```

Output_label This is the pin or cell label that has been entered via the "Label" command. Refer to Section 4.6 for the format of the label.

.EXT The dot extension of the output is automatically appended to the output label by the WinPLACE™ software. The type of extension that is appended on each output label depends on the configuration of the pin or cell, the specific function of the product or sum term, and the device type. Refer to Figure 4-8.

Logic_equation A boolean logic expression that consists of inputs, feedbacks and logic operators. Table 4-3 shows the functions and priorities of the logic operators available in the WinPLACE™ software.

Operator	Logic Function	Priority
( )	Logical organization	1
! or /	NOT	2
& or *	AND	3
# or +	OR	4
$	Exclusive-OR	5

Table 4-3 Priorities of logic operators in WinPLACE™

The output equations can be moved to other locations of the PSF design file as long as they are specified after the reserved identifier EQUATIONS. However, the WinPLACE™ software operation will be affected if the unused or any other equations generated by the WinPLACE™ software are deleted. This means that all equations, whether they are used or unused, are continuously referenced by the WinPLACE™ software.

Functions of the Dot Extensions in the equation labels

In the WinPLACE™ source file, each of the Dot extensions represents a specific function. Figure 4-8details the functions of all Dot extensions.

Logic Reduction Compiler Directive (For Equations Only)

A compiler directive is available to prevent redundant terms in the equations from being removed during the logic optimization process. Sometimes redundant terms are intentionally added to avoid race or hazard conditions, especially in asynchronous applications.

Format: **@REDUCE ON** or **@R+**
@REDUCE OFF or **@R-**

```
Example:    @R-                          "same as @Reduce Off
            G_Latch.COM =                "Gated Latch Application
                 LAT_EN & LAT_IN
                 !LAT_EN & G_Latch
                 LAT_IN & G_Latch;       "redundant term to fix hazard
            @R+                          "same as @Reduce On
```

@Reduce On	Equations following this directive will be optimized. Redundant terms will be removed from the equations. This is the default condition.
@Reduce Off	All equations specified after this directive will be flattened (i.e. converted to Sum-of-Product equations from complex equations) but not optimized. Redundant terms will be left in the equations.

PEEL™ Arrays

Sum Term	Dot Extension	LCC Function (unless stated otherwise)
Sum A	.COM	Combinatorial Internal/External Output
	.D	D Input of the Register
	.T	T Input of the Register
	.J	J Input of the Register
	.SumA	Sum A term is unused
Sum B	.COM	Combinatorial Internal/External Output
	.K	K Input of the Register
	.AP	Asynchronous Preset for Register
	.SumB	Sum B term is unused
Sum C	.COM	Combinatorial Internal/External Output
	.CLK	Asynchronous Clock for Register
	.AR	Asynchronous Reset for Register
	.SumC	Sum C term is unused
Sum D	.CLK	Asynchronous Clock for Register
	.OE	External Output Enable Control (IOC)
	.FB	Buried feedback (PA7128, PA7536,
	.SumD	PA7140, and PA7572)
		Sum D term is unused

PEEL™ Devices

Prod Term	Dot Extension	Function
And A	.OE	External Output Enable Control
	.AndA or .And	Product (AND) term A is unused

Figure 4-10 Functions of the Dot Extension in the Equations Labels

Equations of the Outputs used for State Diagrams or Truth Tables

Boolean equations are generated for all pins and cells that are labeled via the "Label" command, including those that are specifically used for state diagram or truth table designs. **These equations if they are unmodified do not affect the logic of the state diagrams or truth tables because they always equate to zero.** However, if the equations are modified and they do not equate to zero, then they will be logically ORed with the boolean equations that are transformed from the state diagram or truth table design syntax by the WinPLACE™ compiler.

5 WinPLACE™ Application Examples

5.1 Overview

There are several WinPLACE™ application examples provided with the WinPLACE™ software; these are listed in Table 5-1 The following pages provide descriptions and block diagrams for the most important of the design examples. The design source (PSF) files with equation, state machine, and truth table descriptions are not shown. Please use the WinPLACE™ software to review the design source files. The WinPLACE™ document operation can also be useful for printing out the PSF design files or other applicable graphics and text files.

File Name	Device	Description
*DEMO1A.PSF	PA7540	Demonstration design for the PA7540 which includes Basic Gates and Registers, 8-bit Down Counter, 2-bit State Machine, 4-bit Shift Register and 8-bit Bidirectional I/O Port
REG7536. PSF	PA7536	Basic Register Configurations
BI_PORT.PSF	PA7024	8-bit Bidirectional I/O Port (part of DEMO1A)
COUNTER1.PSF	PA7024	8-bit Down Counter with Preset, Reset and Hold (part of DEMO1A)
*TIMER.PSF	PA7024	16-Bit Programmable Clock Generator/Timer
*JACK7024.PSF	PA7024	Blackjack Machine Example
*TC7140.PSF	PA7140	8-bit Time/Counter
*V8GATES.PSF	18CV8	Basic Logic Gates
*V8REGS.PSF	18CV8	Basic Registers and Latches
*V8CLKADD.PSF	18CV8	Clock Divider Address Decoder
V8BUSMUX.PSF	18CV8	Bus Programmable 8-to-1 Multiplexer
V8FCNTR.PSF	18CV8	8-bit Counter with Function Controls
V8CPORT.PSF	18CV8	Change-of-State Input Port with Interrupt
V8SYNC.PSF	18CV8	Synchronization Circuit
V10CNT8.PSF	22CV10	8-bit Up/Down Loadable Counter with Carry-out or Borrow-in
PARV10A.PSF	22CV10A	9-bit Even/Odd Parity Generator/Checker
V10ZPORT.PSF	22CV10A+	Change-of-State Input Port with Interrupt

*Design descriptions are provided in the following pages.

Table 5-1 WinPLACE™ application examples

5.2 DEMO1 A.PSF -- PA7540

The WinPLACE™ design file DEMO1A.PSF incorporates several applications within one design, including: Basic Gates, Basic Registers and Latches, 8-bit Counter, Bi-Directional I/O Port and a Divide-by-2 Clock design. Figure 5-1 and Figure 5-2 show the WinPLACE™ pin block and equivalent schematic diagrams.

-- **GATES** Basic combinatorial functions including AND, OR, NOR, NAND, EXOR, Inverter, 4-to-1 mux, and 4-bit comparator.

-- **REGS** Basic registers including D, T and JK flip-flops with independent clocks, presets and resets, an SR Latch (for debouncing inputs), a gated-latch (LAT1), a basic storage register (REG1), a 2-bit state machine (S0,S1), and a 4-bit shift register (SHF0-3).

-- **COUNTER** An 8-bit down counter with Hold, Preset and Reset.

-- **PORT** An 8-bit bi-directional I/O port with registered input.

The mode inputs, MODE1 and MODE2, control application selection. Pins A through H are used as inputs and/or control, pins I through P are used as outputs. The outputs of each application are selected via eight 4-to-1 muxes.

Output enable and direction (in PORT mode) are selected by CONTROL.

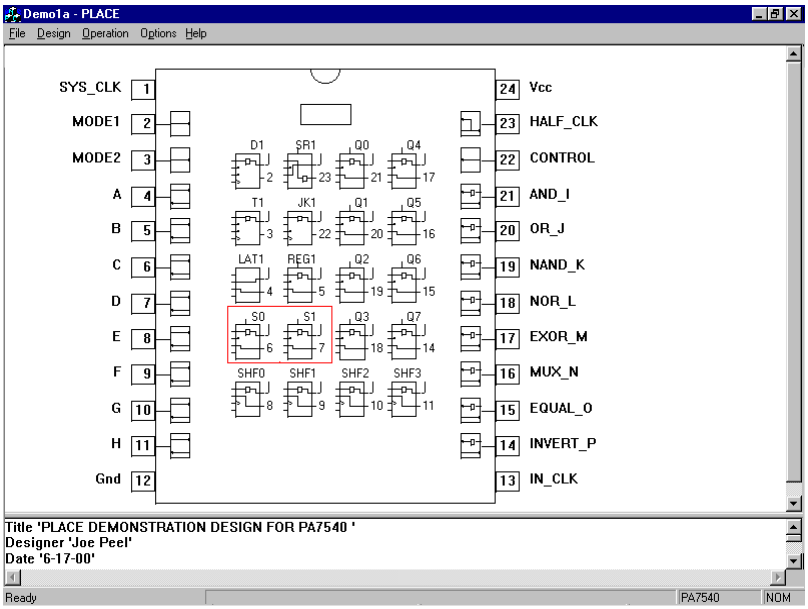


Figure 5-1 WinPLACE™ pin block diagram of DEMO1A.PSF

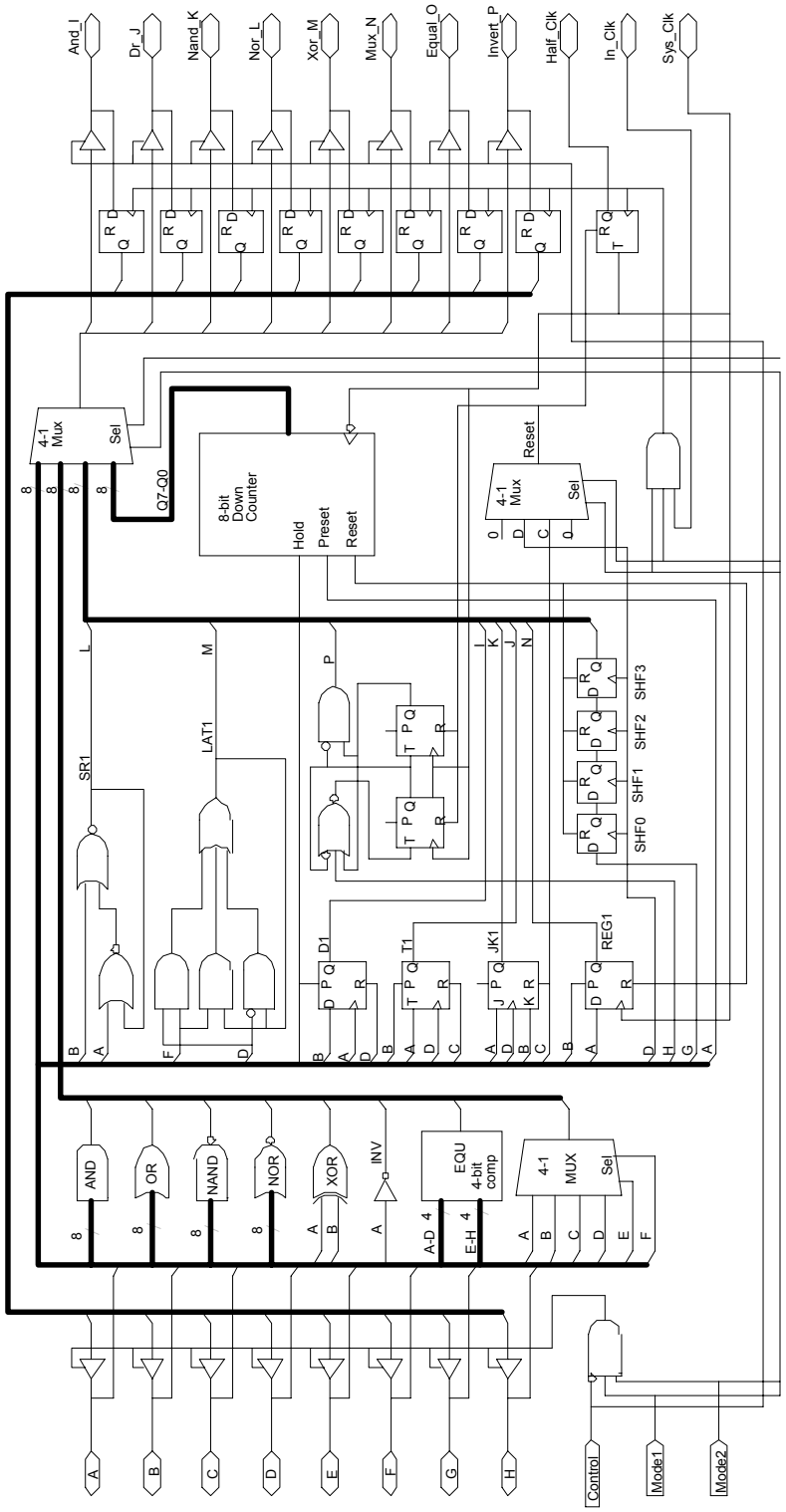


Figure 5-2 Logic schematic of DEMO1A.PSF

5.3 TIMER.PSF -- PA7024

This application uses the PA7024 to implement a 16-bit programmable clock-generator/interrupt-timer that can be interfaced to a 16-bit microprocessor bus. The CLK input can operate over 50MHz (five times that of conventional programmable counter/timer ICs). A buried 16-bit reloadable down counter is used to divide the high-speed input clock.

Upon power-up the counter is disabled. To program the counter, a value must be written into a 16-bit count register from the D0-D15 I/O pins and the counter must be enabled (see Table 5-2 for control). The value in the count register will be loaded into the counter, which will count down to 0000 Hex. After reaching 0000 Hex, it will automatically reload the value from the count register. This allows the counter to be free running for clock generation if the value in the count register is maintained. Note that the "register-type change" feature in the PA7024 global cell is used to dynamically switch the T registers to D registers for loading when the counter reaches the count 0000 Hex.

If the count register is changed, the new count will be loaded after the count reaches 0000 Hex. One-shot operation for timer controlled interrupts can be implemented by setting the count register to 0000 Hex after the count has been loaded. When this is done, the counter will stop and stay at 0000 Hex. The OUT pin will toggle (initially low then high and so on) each time the counter reaches 0000 Hex. The counter and OUT pin can be reset, disabled or enabled via a bus command (see Table 5-2). The count can be read "on the fly" via the D0-D15 pins, temporarily held stable until the read is completed.

/CS	/RD	/WR	/A0	Function
1	X	X	X	Not selected (Don't Care)
0	1	0	0	Write Count Register from D0-D15
0	0	1	0	Read Count Register onto D0-D15
0	1	0	1	Reset/Stop Counter and OUT
0	0	1	1	Enable and Read Counter onto D0-D15

Table 5-2 Command table for TIMER.PSF

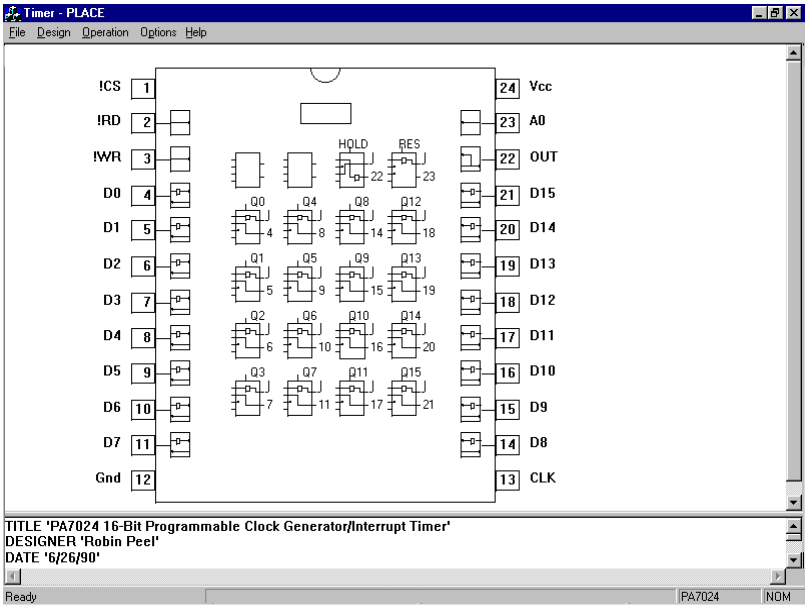


Figure 5-3 WinPLACE™ pin block diagram of TIMER.PSF

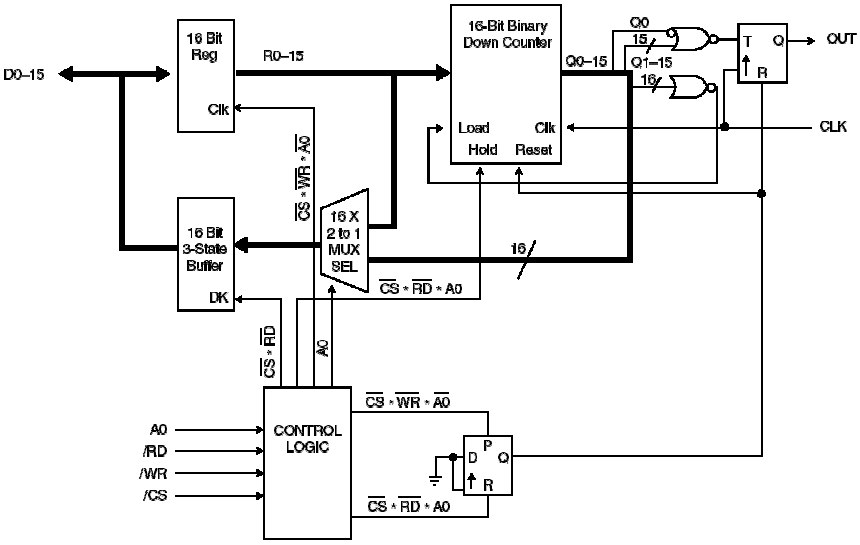


Figure 5-4 Logic schematic of TIMER.PSF

5.4 JACK7024.PSF -- PA7024

This design example is based on C.R. Clare's design in Designing Logic Systems Using State Machines (McGraw Hill, 1972). The blackjack machine plays the dealer's hand, using typical dealer strategies to decide whether to draw another card (hit) or stand after each round.

The example contains the following logic designs:

· A state machine that controls the game logic, which includes:

-- checking the status of the card reader.

-- making the decision of what action to take for a hit, stand or a bust. An example is to draw a card if the hit signal is true.

-- making the decision of when to use the value 1 or 11 for an ace card.

· A Multiplexer/Comparator, which compares the point total and sends the hit, stand or bust signal to the state machine. If the point total is greater than 21, it's a bust. If it is equal or less than 16, then hit, else stand.

· A 5-Bit Adder that adds the value of the drawn card.

· A Binary-to-BCD converter for converting the 5-bit binary score and converts it to 2-digit BCD for the digital display.

This design example can also be implemented by using three PLDs, which include a PAL22V10 for the Multiplexer, Comparator and Adder, a PAL16L8 for the Binary-to-BCD converter, and a PAL16R6 for the state machine.

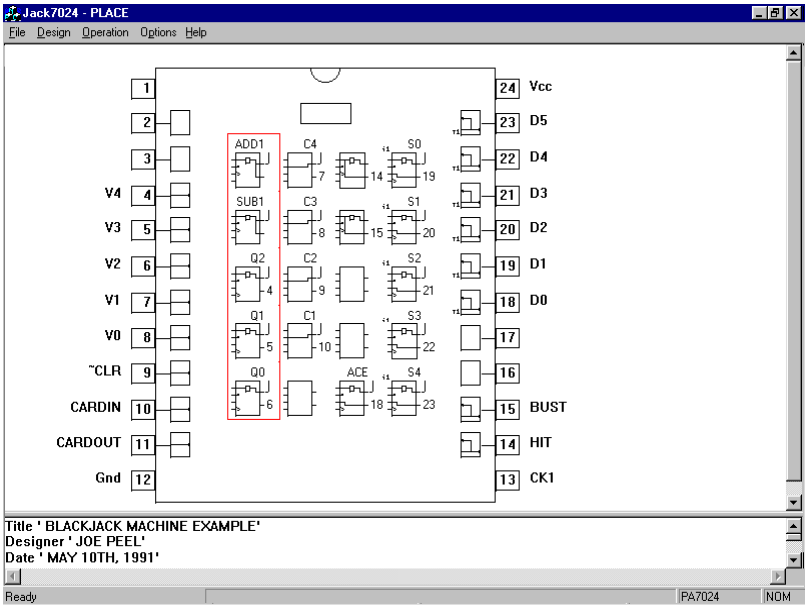


Figure 5-5 WinPLACE™ pin block diagram of JACK7024.PSF

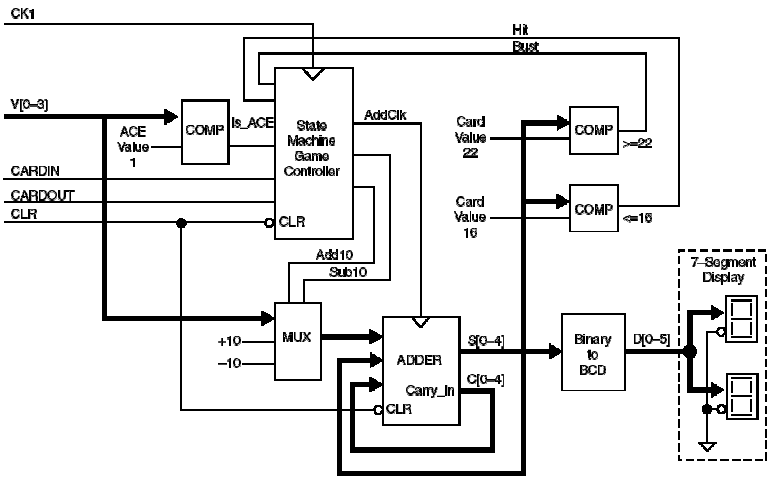


Figure 5-6 Logic diagram of JACK7024.PSF

5.5 TC7140.PSF -- PA7140

This example uses a PA7140 device to implement a timer/counter application, which is typically used in a microprocessor-based computer system. The circuit employs a multiplexer to allow either the incoming or latched data to be loaded into the counter. The desired data is then loaded into the counter either by resetting the counter and the compare register, or by a match between the counter's state and the value in the compare register.

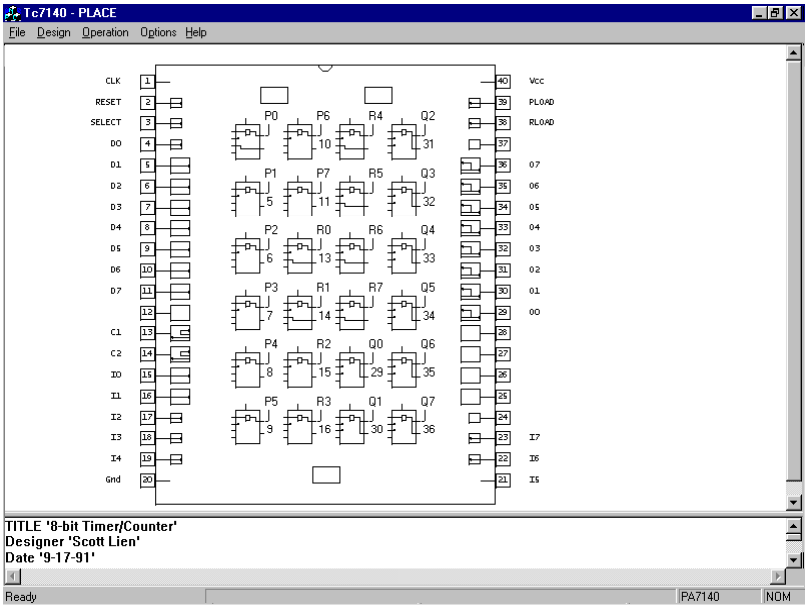


Figure 5-7 WinPLACE™ pin block diagram of TC7140.PSF

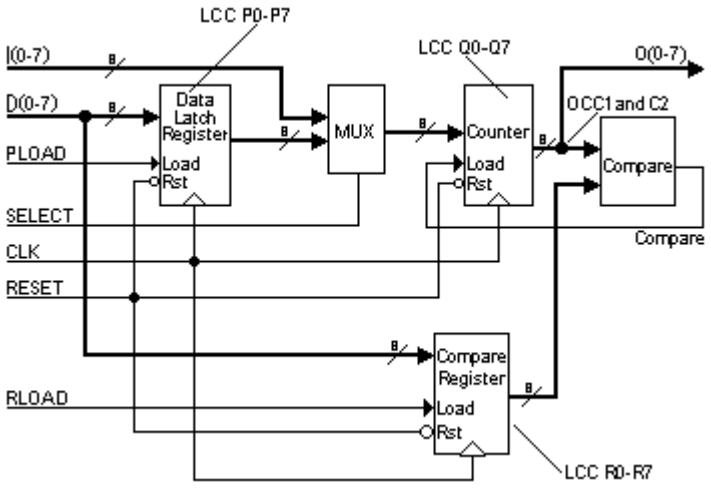


Figure 5-8 Logic diagram of TC7140.PSF

5.6 V8GATES.PSF -- PEEL18CV8

This PEEL18CV8 application example implements several basic logic gates. The logic gates include:

-- an inverter
-- four-input AND, OR, NAND, and NOR gates
-- a four-input AND-OR-INVERT gate o a two-input XOR gate
-- a high-impedance buffer.

Each gate uses one or more of the (A,B,C and D) inputs. Additionally, the high-impedance buffer uses the /OE input for impedance control. The truth table for these gates can be examined in the test vectors. Note, the remaining unused input pins can be used as additional inputs into the gates.

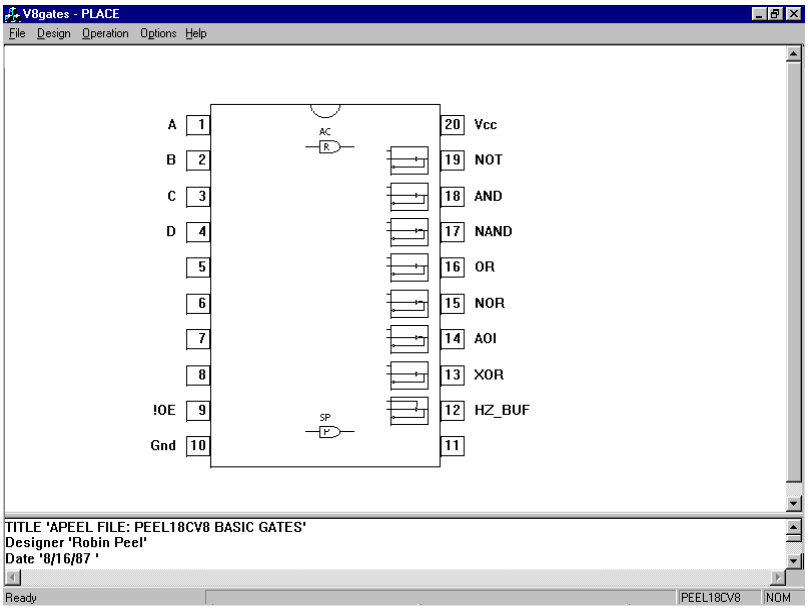


Figure 5-9 WinPLACE™ block diagram of V8GATES.PSF

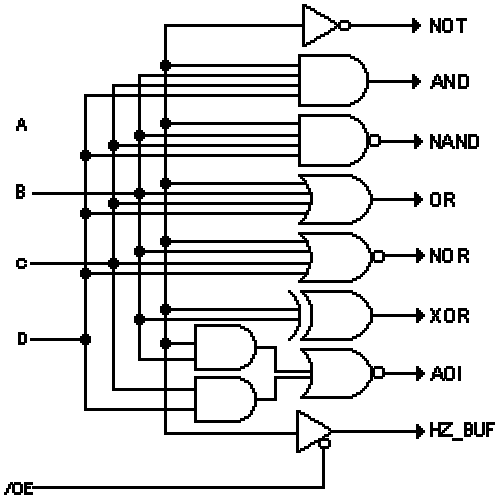


Figure 5-10 Logic schematic of V8GATES.PSF

5.7 V8REGS -- PEEL18CV8

This application example demonstrates the implementation of several basic registers and latches within a PEEL18CV8. Four register types included are the D, T, JK, and SR, all of which are clocked by the CLK input. All registers can be synchronously reset, set, and asynchronously reset using the SRES, SSET and ARES inputs respectively. Besides the registers, a SR latch and a Gated Latch circuit show how independent asynchronous storage elements can be implemented. Only the Q outputs of these registers and latches are provided at the output pins. The /Q outputs could easily be accessed by inverting the macrocell output polarity. Truth table operation can be referenced via the test vectors.

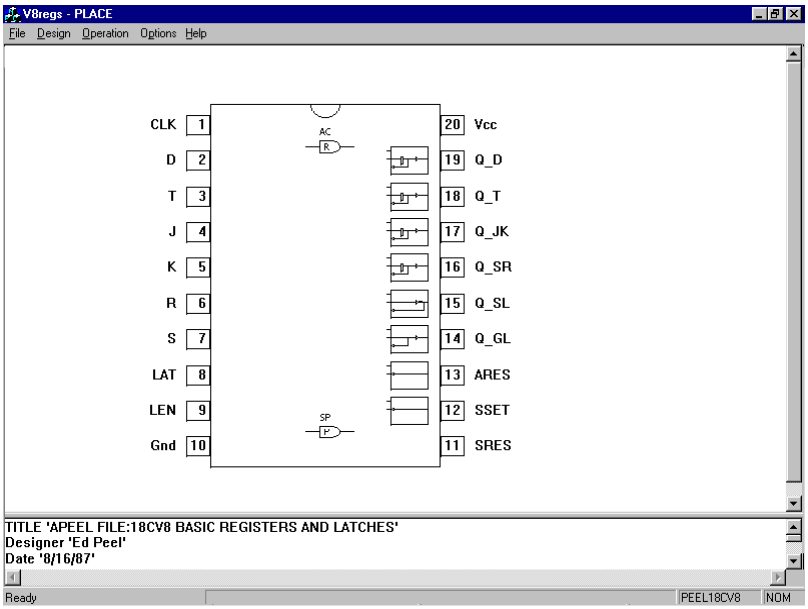


Figure 5-11 WinPLACE™ block diagram of V8REGS.PSF

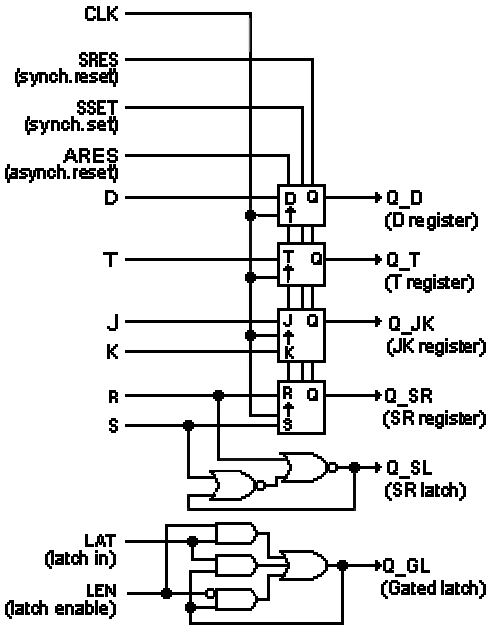


Figure 5-12 Logic schematic of V8REGS.PSF

5.8 V8CLKADD.PSF -- PEEL18CV8

This application uses the PEEL18CV8 for two common microprocessor system functions: a clock divider and a memory mapped address decoder. The clock divider provides +2, +4 and +8 clock outputs. The SET input sets all clock outputs high. The address decoder decodes the processor address lines to select one of five memory or I/O devices. The chip select for these devices are active low. The memory map over a 64K boundary is shown below.

Memory Map for Address Decoder Function

Function	Address
EPROM (32K X 8)	8000-FFFF Hex
EEPROM (2K X 8)	5000-5FFF Hex
UART	4100-41FF Hex
PORT	4000-40FF Hex
SRAM (8K X 8)	0000-1 FFF Hex

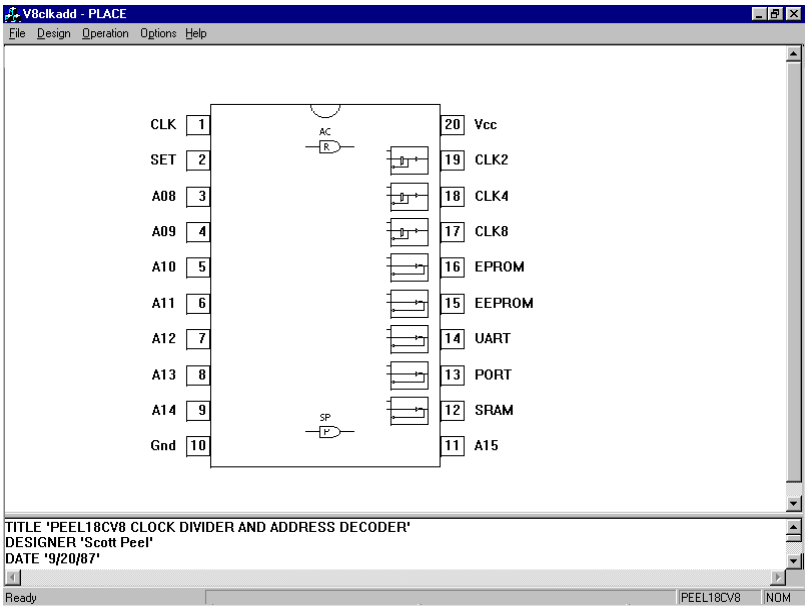


Figure 5-13 WinPLACE™ block diagram of V8CLKADD.PSF

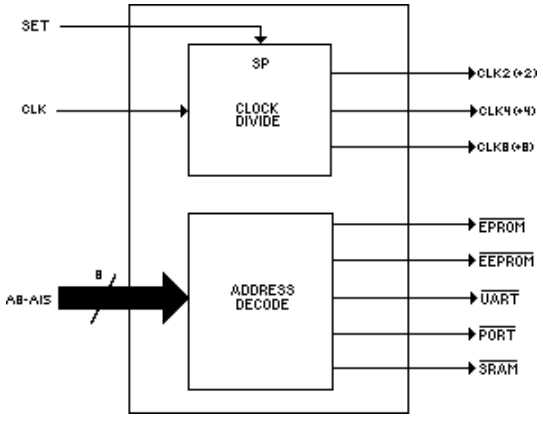


Figure 5-14 Block diagram of V8CLKADD.PSF