# Design Vision™
# User Guide

Version M-2016.12, December 2016

# Copyright Notice and Proprietary Information

## Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

## Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

## Trademarks

## Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
690 E. Middlefield Road
Mountain View, CA 94043
www.synopsys.com

## Copyright Notice for the Command-Line Editing Feature

## Copyright Notice for the Line-Editing Library

# Contents

# Preface

This preface includes the following sections:

- About This User Guide
- Customer Support

# About This User Guide

This section contains information about the target audience of this document, where to find other pertinent publications, and documentation conventions used in this manual.

## Audience

This user guide is for logic design engineers who have some experience using the Synopsys Design Compiler® or DC Explorer tool and who want to use the visualization features of the Design Vision tool for analysis. To use this user guide, you should be familiar with

- Synthesis using Design Compiler or DC Explorer
- VHDL or Verilog HDL
- The UNIX or Linux operating system

## Related Publications

For additional information about the Design Vision tool, see the documentation on the Synopsys SolvNet® online support site at the following address:

https://solvnet.synopsys.com/DocsOnWeb

You might also want to see the documentation for the following related Synopsys products:

- DC Explorer
- Design Compiler
- DFT Compiler/DFTMAX™
- Power Compiler™

## Release Notes

Information about new features, enhancements, changes, known limitations, and resolved Synopsys Technical Action Requests (STARs) is available in the *Design Vision Release Notes* on the SolvNet site.

To see the *Design Vision Release Notes*,

1. Go to the SolvNet Download Center located at the following address:

   https://solvnet.synopsys.com/DownloadCenter

2. Select Design Vision, and then select a release in the list that appears.

## Conventions

The following conventions are used in Synopsys documentation.

| Convention | Description |
|---|---|
| `Courier` | Indicates syntax, such as `write_file`. |
| *`Courier italic`* | Indicates a user-defined value in syntax, such as `write_file` *`design_list`*. |
| **`Courier bold`** | Indicates user input—text you type verbatim—in examples, such as<br>`prompt>` **`write_file top`** |
| [ ] | Denotes optional arguments in syntax, such as `write_file [-format` *`fmt`*`]` |
| ... | Indicates that arguments can be repeated as many times as needed, such as *`pin1 pin2 ... pinN`* |
| \| | Indicates a choice among alternatives, such as `low \| medium \| high` |
| Ctrl+C | Indicates a keyboard combination, such as holding down the Ctrl key and pressing C. |
| \ | Indicates a continuation of a command line. |
| / | Indicates levels of directory structure. |
| Edit > Copy | Indicates a path to a menu command, such as opening the Edit menu and choosing Copy. |

# Customer Support

Customer support is available through SolvNet online customer support and through contacting the Synopsys Technical Support Center.

## Accessing SolvNet

The SolvNet site includes a knowledge base of technical articles and answers to frequently asked questions about Synopsys tools. The SolvNet site also gives you access to a wide range of Synopsys online services including software downloads, documentation, and technical support.

To access the SolvNet site, go to the following address:

https://solvnet.synopsys.com

If prompted, enter your user name and password. If you do not have a Synopsys user name and password, follow the instructions to sign up for an account.

If you need help using the SolvNet site, click HELP in the top-right menu bar.

## Contacting the Synopsys Technical Support Center

If you have problems, questions, or suggestions, you can contact the Synopsys Technical Support Center in the following ways:

- Open a support case to your local support center online by signing in to the SolvNet site at https://solvnet.synopsys.com, clicking Support, and then clicking “Open A Support Case.”
- Send an e-mail message to your local support center.
  - E-mail support_center@synopsys.com from within North America.
  - Find other local support center e-mail addresses at

    http://www.synopsys.com/Support/GlobalSupportCenters/Pages
- Telephone your local support center.
  - Call (800) 245-8005 from within North America.
  - Find other local support center telephone numbers at

    http://www.synopsys.com/Support/GlobalSupportCenters/Pages

# 1

# Design Vision Introduction

The Design Vision product is the graphical user interface (GUI) of the Synopsys synthesis products. It provides tools for viewing and analyzing your design at the generic technology (GTECH) level and the gate level. It also provides all of the synthesis capabilities of the Design Compiler product. Menu commands and dialog boxes are available for the most commonly used synthesis features. In addition, you can enter any dc_shell command on the command line in the GUI or the shell.

For an overview of the Design Vision tool, see the following topics:

- About Design Vision
- Supported Platforms
- The Design Vision Documentation Set
- Design Vision and Other Synopsys Products

# About Design Vision

For an overview of the Design Vision tool, see the following sections:

- Features and Benefits
- User Interfaces
- Methodology
- Supported Formats

## Features and Benefits

The Design Vision GUI provides the following features:

- Window- and menu-driven interface for Synopsys synthesis tools (Design Compiler, DC Explorer, DFT Compiler/DFTMAX, and Power Compiler)
- Visualization and analysis capabilities that include
    - A hierarchy browser for navigating through the design hierarchy and exploring design structures

      You can view hierarchical cells and blocks and gather information about the objects (cells, pins, nets, and ports) in a design
    - Histograms for visualizing the overall timing performance of the design and examining trends in various metrics, for example slack and capacitance
    - Schematics for visually examining timing paths, including fanin and fanout logic, and both high-level and low-level design connectivity

      Schematics can help you to analyze timing-path problems by focusing on the critical paths in your design. You can create schematics to view the top-level design, hierarchical cells, selected timing paths, or selected design objects.
    - A path analyzer for performing custom trend analysis on collections of timing paths

      You categorize the paths by using predefined or custom category rules based on available timing attributes, and then examine them in a color-coded treemap view.
    - A timing analysis driver and path data tables for viewing timing path details, such as attribute values, and for accessing other timing analysis tools
    - A path inspector for detailed timing path analysis

      You can view delay profiles, a path summary, the clock path and datapath elements, and the slack details for an individual timing path.

    - Path profiles for visually examining the contributions of individual cells and nets to the total delay of a timing path

    - A properties viewer and list views for examining object information, such as attribute values

    - An RTL browser for finding and debugging RTL source problems for selected cells and timing paths

    - A layout view for analyzing and debugging floorplan elements and physical constraints in a design that you optimize by using Design Compiler topographical technology

    - A congestion map for visually examining highly congested areas in your floorplan

    - Visual modes for visually examining specific design information in the physical layout, such as hierarchical cell placement, cell placement in voltage areas, and the distribution of selected groups of cells, nets, ports, and pins

    - DFT analysis views (DRC violation browser, violation inspector, and hold time analysis windows) for examining static or dynamic DRC violations and DFT hold time violations

    - A UPF diagram view for visually examining a graphic representation of the power architecture as it is described in your multivoltage design database

    - A Visual UPF generator for designing and implementing the power architecture for a multivoltage design

      You can create power domains and define their supply networks, connections with other power domains, and relationships with elements in the design hierarchy.

    - A power state table viewer for performing always-on analysis and multivoltage level-shifter analysis in correlation with a UPF diagram

    - An MV Advisor violation browser that provides a visual analysis and debugging environment for design violations in a multivoltage design

- Reporting capabilities that correlate reported objects to graphical views, including a text report viewer in which you can save or open report files and an HTML report viewer for resource reports with links to RTL files

- An integrated command-line interface with scripting support for all Tool Command Language (Tcl) commands

Using the features of the Design Vision GUI in conjunction with the Design Compiler tool and other Synopsys synthesis tools, you can

- Navigate through the design hierarchy and explore design structures

- Obtain a high-level overview of the timing performance

- Perform timing analysis for blocks you are synthesizing
- Perform detailed visual analysis of selected timing paths and connected logic
- Validate physical constraints and visually examine the physical placement of critical timing path objects in your floorplan

  For example, a cell might be placed at a physical distance from the rest of the path because of its fanin or fanout nets.
- Visually examine the orientation and physical placement of objects such as macro cells, port locations, placement blockages, and the die area and core area outlines to avoid correlation issues that can result from incorrect or missing physical constraints
- Visually analyze floorplan-related congestion and identify the causes of congestion hotspots
- Analyze and debug congestion problems by cross-probing the RTL for selected cells or timing paths
- Visually examine static and dynamic DRC violations and DFT hold time violations for a test-ready design
- Create UPF power domains and define their supply networks for a multivoltage design
- Visualize the UPF power architecture currently defined in a multivoltage design
- Perform always-on analysis and multivoltage level-shifter analysis using power state tables in correlation with a UPF diagram
- Analyze and debug multivoltage design violations and multivoltage design connections

## User Interfaces

The Design Vision tool offers two interfaces for synthesis and analysis: the Design Vision graphical user interface (GUI) and a shell command-line interface.

- The Design Vision GUI is an advanced visualization and analysis tool set.

  The GUI can perform certain tasks, such as very accurately displaying your design, and it provides visual analysis tools that are available only in the GUI. The look and feel of the Design Vision GUI is consistent with the look and feel of other Synopsys GUI tools.
- The design_vision shell command-line interface is a text-only environment that is identical to the Design Compiler shell command-line interface (dc_shell).

  You enter commands at the command-line prompt the same way you enter them in dc_shell. For information about using the shell command-line interface, see the *Design Compiler User Guide*.

The Design Vision GUI offers menus and dialog boxes for important Design Compiler functions. The GUI also provides menus and dialog boxes for visual analysis features that you can use to visualize design data and analyze results. In addition, the GUI provides a command console with a Tcl command-line interface and views of the session log and the command history. You can perform any task in the GUI that you can perform in the shell.

The command-line interface provides access to all the capabilities of the Synopsys synthesis tools. You can execute Tcl commands in the following way

- By typing single commands interactively on the console command line in the Design Vision window
- By entering single commands interactively in the shell
- By running one or more command scripts, which are text files of commands

Using this approach allows you to supplement the subset of Design Compiler commands available through the menu interface. For information about Tcl, see the *Using Tcl With Synopsys Tools* manual.

The shell command-line interface is always available. You can open or close the GUI multiple times during a session. The GUI opens by default when you start the Design Vision tool. Help is available for both interfaces.

## Methodology

The Design Vision tool allows you to use the same design methodology and scripts you currently use and to extend your methodology with Design Vision visual analysis. Many Design Compiler commands are available on Design Vision menus. All Design Compiler functions are available through the Design Vision command-line interface.

## Supported Formats

The Design Vision tool stores design data in an internal database format. It supports two design database formats: the Synopsys logical database format (.ddc) and the Synopsys Milkyway™ format.

- .ddc format

  The .ddc format is a single-file, binary format. The .ddc format stores design data in an efficient manner than the .db format, enabling increased capacity. In addition, reading and writing files in .ddc format is faster than reading and writing files in .db format. The .ddc format stores only logical design information.

- Milkyway format

  The Milkyway format allows you to write a Milkyway database for use with other Synopsys Galaxy tools, such as the IC Compiler tool. The Milkyway format stores both logical and physical design information, but it requires a mapped design.

The Milkyway format is available only when you start the tool in topographical mode. Use the `write_milkyway` command to save netlist and physical design data in a Milkyway design library. You can use a single Milkyway library across the entire Galaxy flow. For more information, see the *Design Compiler User Guide*.

Note:
Design Vision does not support the `read_milkyway` command.

The Design Vision tool can access all the files supported by the Design Compiler tool. Table 1-1 shows the supported design file formats. All netlist formats except .db, equation, PLA, state table, Verilog, and VHDL require special license keys.

*Table 1-1 Supported File Formats*

| **Data** | **Formats** |
|---|---|
| Netlist | Milkyway |
| | Programmable logic array (PLA) |
| | Synopsys equation |
| | Synopsys state table |
| | Synopsys dc_shell database format (.ddc) |
| | Verilog |
| | VHDL |
| Timing | Standard Delay Format (SDF) |
| Command Script | Tcl |
| Library | Synopsys internal library format (.lib)<br>Synopsys database format (.db) |
| Parasitics | dc_shell command scripts |

# Supported Platforms

The Design Vision tool is supported on the same platforms that support the Design Compiler tool and the other Synopsys synthesis tools. Your hardware and operating system vendor has required patches available for your system. For information about the supported hardware and operating systems and the required operating system patches necessary to run the synthesis tools, see the *Installing Synopsys Tools* at the following address:

http://www.synopsys.com/install

From this Web page you can navigate to the *Synopsys Synthesis Tools Installation Notes* for your release.

# The Design Vision Documentation Set

You can find most of the information you need to know to run the Design Vision tool in the Design Vision documentation set.

The Design Vision documentation set is divided into these parts:

- *Design Vision User Guide*
- Design Vision Help

Other sources of information include man pages, the SolvNet knowledge base, and the Customer Support Center. For information about accessing these sources of information, see “Customer Support” on page xv.

## Design Vision User Guide

The *Design Vision User Guide* assumes you are familiar with basic Design Compiler concepts.

The user guide provides guidance in solving particular problems. For example, it presents short procedures that use the analysis visualization features of the GUI to locate and solve timing problems.

Sometimes steps in a procedure refer to actions without further explanation: for example, “Create a histogram” or “Create a schematic.” Such steps refer to features of the GUI that are explained in Design Vision Help.

The *Design Vision User Guide* does not contain specific information about individual menu items or dialog boxes. For such information, see Design Vision Help.

In Chapter 5, “Solving Timing Problems,” experienced Design Compiler users can learn how to do certain familiar synthesis tasks using the Design Vision tool. However, the user guide explains such topics only briefly.

## Design Vision Help

Design Vision Help is available in the Design Vision GUI. You can access the Design Vision Help from the Help menu in the Design Vision window or the Layout window. The Help system contains topics that explain the details of tasks that you can perform. For example, if you need help performing a step in a procedure presented in the user guide, you can find the information you need in Design Vision Help.

Information in Design Vision Help is grouped in the following categories:

- Feature topics

  Overviews of Design Vision window components and tools.

- How-to topics

  Procedures for accomplishing synthesis and analysis tasks.

- Reference topics

  Explanations of views, toolbar buttons, menu commands, and dialog box options.

Note:
> Before you can access Design Vision Help from within the GUI, the Web browser executable file must be listed in your UNIX or Linux path variable.

Design Vision Help is a browser-based HTML Help system designed for viewing in a Web browser.

To access online Design Vision Help,

1. Choose Help > Online Help.

   The Web browser appears and displays the Welcome topic for the Design Vision Help.

2. Use the navigation frame (leftmost frame) to find the information you need in one of the following ways:

   ❍ Find the topic in the hierarchical organization of the Help system by clicking Contents and expanding the appropriate books until you find the information you need.

   ❍ Find the topic by its subject by clicking Index and looking for the subject in the alphabetical listing.

- Search for keywords found in the topic by entering the keywords and clicking Search.

  If more than one topic has the words you are searching for, you must select the appropriate topic from a list of topics.

Design Vision Help makes extensive use of JavaScript and cascading style sheets (CSS). If your browser encounters problems displaying Design Vision Help, open the browser preferences and make sure that JavaScript and style sheets are enabled and that JavaScript is not blocked by your security preferences.

Note:

> If you reset preferences while the Help system is open, you might need to click the Reload button on the browser's navigation toolbar after you reset the preferences.

You can view Design Vision Help as a standalone Help system in your Web browser by opening the file named index.html in the online Help directory: $SYNOPSYS/doc/syn/html/dvoh/enhanced.

The default Help browser is Mozilla Firefox. If you prefer to use a different browser, note the following limitations:

- Online Help is designed to run in the Firefox browser.
- Online Help is not tested or supported in other browsers, such as Google Chrome, Chromium, SeaMonkey, or Internet Explorer.

**See Also**

- The “Using This Help System” topic in Design Vision Help

## Design Vision and Other Synopsys Products

As a visual analysis tool and the GUI for Synopsys synthesis, the Design Vision tool works with the Design Compiler, DFT Compiler/DFTMAX, and Power Compiler tools to synthesize and analyze your design. The Design Vision GUI is also available in the DC Explorer tool.

The Design Vision and Synopsys PrimeTime tools have similar timing visualization features; however, the tools have different timing engines and differ in their application to analysis. The Design Vision tool has the same static timing engine as the Design Compiler tool. Use the Design Vision tool to perform timing analysis and modification of blocks you are synthesizing. Use the PrimeTime tool for static timing sign-off or for analyzing the timing of a chip or of large portions of a chip.

# 2

# Working With Design Vision

The Design Vision tool offers two interfaces: the design_vision shell command-line interface (or shell) and the Design Vision graphical user interface (GUI). The shell command-line interface is a text-only environment in which you enter commands at the command-line prompt. The GUI provides menus with frequently-used synthesis commands and visual analysis tools for the Synopsys synthesis environment; use it for visualizing design data and analyzing results.

To learn how to operate the Design Vision GUI, see the following topics:

- Running Design Vision
- Graphical User Interface
- Getting Help in the GUI

# Running Design Vision

To learn how to run the Design Vision tool and use the GUI, see the following topics:

- Design Vision Modes
- License Requirements
- The Design Vision Setup Files
- Starting the Tool
- Entering Tcl Commands in the GUI
- Choosing Menu Commands in GUI Windows
- Opening and Closing the GUI
- Using Script Files
- Saving Designs and Exiting Design Vision

## Design Vision Modes

You can use the Design Vision tool in the following modes:

- Wire load mode
- Topographical mode
- Multimode
- UPF mode

Wire load mode and topographical mode are tool modes. When you start the Design Vision tool, you must choose either wire load mode or topographical mode.

Multimode and UPF mode are not tool modes; multimode allows you to operate the tool under multiple operating conditions and multiple modes, such as test mode and standby mode. UPF mode allows you to specify advanced low-power methodologies. Multimode and UPF mode are available only in topographical mode.

For more information about these modes, see the *Design Compiler User Guide*.

## License Requirements

To use the Design Vision tool, you need the Design-Vision license. To use the Design Vision tool in topographical mode, you need a Design-Vision license, a DesignWare license, and the DC Ultra package. To use the Design Compiler Graphical Layout window in topographical mode, you also need a DC-Extension license. If you use the Milkyway flow in topographical mode, you also need a Milkyway-Interface license; this license is included in the DC Ultra package.

Synopsys licensing software and the documentation describing it are separate from the tools that use it. You install, configure, and use a single copy of Synopsys Common Licensing (SCL) for all Synopsys tools. By providing a single, common licensing base for all Synopsys tools, SCL reduces license administration complexity and minimizes the effort you expend in installing, maintaining, and managing licensing software for Synopsys tools.

For complete Synopsys licensing information, see the *Synopsys Common Licensing Administration Guide*. This guide provides detailed information about SCL installation and configuration, including examples of license key files and troubleshooting guidelines.

### See Also

- The “Getting and Releasing Licenses” topic in Design Vision Help

  Provides information about checking out and releasing licenses using the GUI

- The *Design Compiler User Guide*

  Provides information about the licenses required for synthesis and about checking out and releasing licenses in shell

## The Design Vision Setup Files

Before starting the Design Vision tool, make sure your `$SYNOPSYS` variable is set, and the path to the bin directory is included in your `$PATH` variable. Be sure to specify the absolute path to indicate the Synopsys root that contains the Design Vision installation, as shown:

```
/tools/synopsys/2013.03/bin/
```

If you use a relative path (../), as shown, the tool cannot access the libraries that are located in the root directory:

```
../../2013.03/bin/
```

When you start the tool in wire load or topographical mode, it automatically executes commands in the three standard Design Compiler setup files that dc_shell uses. These files have the same file name, .synopsys_dc.setup, but they reside in different directories. The same sourcing rules apply for both the design_vision shell and dc_shell. For more information about the .synopsys_dc.setup files and the initialization settings for synthesis, see the *Design Compiler User Guide*.

In addition, the tool reads another set of setup files when you open the GUI, named .synopsys_dv_gui.tcl. You can use these files to perform GUI-specific setup tasks. Use the .synopsys_dc.setup files to perform non-GUI application setup tasks. Settings from the .synopsys_dv_gui.tcl files override settings from the .synopsys_dc.setup files.

The tool reads the .synopsys_dc.setup and .synopsys_dv_gui.tcl files from three directories in the following order:

- The Synopsys root directory

  These system-wide setup files contain system variables defined by Synopsys and general Design Compiler and Design Vision setup information for all users at your site. Only the system administrator can modify these files.

- Your home directory

  These user-defined setup files can contain variables that define your preferences for the Design Compiler and Design Vision working environment. The variables in these files override the corresponding variables in the system-wide setup files.

- The current working directory (the directory from which you start the tool)

  These design-specific setup files can contain project-specific or design-specific variables that affect all of the designs in this project directory. To use these files, you must invoke the tool from this directory. Variables defined in these files override the corresponding variables in the user-defined and system-wide setup files.

You can use the setup file in your home or design directory to define Tcl scripts that you need to run during a Design Vision or Design Compiler session. For more information, see “Using Script Files” on page 2-10.

In addition to reading the setup files, the tool loads GUI preferences and view settings from a file named .synopsys_dv_prefs.tcl in your home directory. You should not edit this file. For more information, see “Setting GUI Preferences” on page 2-23.

### See Also

- The “Using Setup Files” topic in Design Vision Help
- *Synopsys Synthesis Tools Installation Notes*

  Provides information about defining the `$SYNOPSYS` and `$PATH` variables

- The *Design Compiler User Guide*

  Provides information about the locations of setup files and initialization settings for synthesis

## Starting the Tool

The Design Vision tool operates in the X windows environment on UNIX or Linux. Before starting a Design Vision session, make sure your `$SYNOPSYS` variable is set and the path to the bin directory is included in your `$PATH` variable. Before opening the GUI, make sure your `$DISPLAY` environment variable is set to the name of your UNIX or Linux system display. You can optionally set the `$DISPLAY` variable when start the session.

The tool provides an option to enable the Design Compiler topographical technology for designs with physical constraints. You can start a Design Vision session in either wire load mode or topographical mode, but you cannot change the mode during a session.

Note:

> You can query the mode by running the `shell_is_in_topographical_mode` command. The command returns 1 if the tool is running in topographical mode; otherwise it returns 0.

To start the Design Vision tool in wire load mode, enter the `design_vision` command in a UNIX or Linux shell:

```
% design_vision
```

If you are using Design Compiler topographical technology or the Design Compiler Graphical tool, you must indicate this by specifying the `-topographical_mode` option with the `design_vision` command:

```
% design_vision -topographical_mode
```

You can abbreviate this option to as short as `-to`. Topographical mode requires a DC Ultra license and a DesignWare license. For information about additional license requirements, see “License Requirements” on page 2-3.

These commands start the tool and open the GUI by default. The Design Vision window appears on the screen, and the command-line prompt, which is design_vision> in wire load mode or design_vision-topo> in topographical mode, appears in the UNIX or Linux shell and on the console in the Design Vision window.

Be sure to specify the absolute path to indicate the Synopsys root that contains the Design Vision installation, as shown:

```
% /tools/synopsys/2014.09/bin/design_vision
```

If you use a relative path (../), as shown, the tool cannot access the libraries that are located in the root directory:

```
% ../../2013.03/bin/design_vision
```

You can start the tool in the shell command-line interface without opening the GUI by specifying the `-no_gui` option. For example, enter one of the following commands:

```
% design_vision -no_gui
% design_vision -topographical_mode -no_gui
```

When you want to open the GUI, enter the `gui_start` command. For more information, see “Opening and Closing the GUI” on page 2-9.

To set the `$DISPLAY` environment variable when you start a Design Vision session, specify the `-display` *`host_name`* option, where *`host_name`* is the name of your UNIX display terminal. For example, enter one of the following commands:

```
% design_vision -display 192.180.50.155:0.0
% design_vision -topographical_mode -display my_host:0.0
```

To see the complete list of available options without starting the tool, specify the `-help` option with the `design_vision` command:

```
% design_vision -help
```

For detailed information about the startup options, see the `design_vision` man page.

### See Also

- License Requirements
- The Design Vision Setup Files
- Opening and Closing the GUI

## Entering Tcl Commands in the GUI

You can interact with the design_vision shell by using dc_shell commands, which are based on the Tool Command Language (Tcl) and include certain command extensions needed to implement specific Design Compiler functionality. The dc_shell command language provides capabilities similar to UNIX command shells, including variables, conditional execution of commands, and control flow commands.

You can run dc_shell commands in the following ways:

- By entering single commands on the command line at the bottom of the console in the Design Vision window
- By entering single commands on the command line in the shell
- By running one or more command scripts, which are text files of commands

  For details about running command scripts in the GUI, see “Using Script Files” on page 2-10.

You can enter any dc_shell command on the console command line just as you would enter commands in the shell. When you enter a command, the tool echoes the command output (including processing messages and any warnings or error messages) in the console log view. For example, if you enter `get_selection`, the log view displays a list of the names of all selected objects.

To enter a command on the console command line,

1. Click the command line to give it the focus.
2. Type the command.
3. Click the prompt button or press Return.

When entering a command, option, or file name, you can minimize your typing by pressing the Tab key when you have typed enough characters to specify a unique name; the tool completes the remaining characters. If the characters you typed could be used for more than one name, the tool lists the qualifying names from which you can select by using the arrow keys and the Enter key.

You can find information about dc_shell commands by viewing man pages in the man page viewer. You can also use the man page and help utilities just as you would use them in dc_shell.

### See Also

- Console Command-Line Editing
- Getting Help on the Command Line
- Viewing Man Pages

## Choosing Menu Commands in GUI Windows

The Design Vision GUI provides menu commands and dialog boxes for most graphic features, such as generating histograms, displaying schematics, and highlighting design objects. In addition, the GUI provides menu and dialog box equivalents for many dc_shell commands. Menu commands are grouped by function on the menus in each GUI window.

To choose a command on a menu bar menu,

- Click the menu name to open the menu, and click the command name on the menu.

Some frequently used menu commands are also available on pop-up menus for individual views.

To choose a command on a pop-up menu,

- Move the pointer over the object of interest, right-click to display the menu, and click the command name.

A menu command can perform an immediate operation, display a submenu, or display a dialog box.

- Menu commands that display a submenu are followed by a right-pointing arrow.
- Menu commands that open a dialog box that requires a response before performing an operation are followed by an ellipsis (...).

  These commands open a dialog box to prompt you for the information. Dialog boxes that require a response before performing an operation contain OK and Cancel buttons and sometimes an Apply button. After selecting options or entering information in the dialog box, you respond by clicking OK or Apply.
- Menu commands without an arrow or ellipsis either perform an immediate operation or open a dialog box that performs immediate operations.

  Dialog boxes that perform immediate operations usually display options and contain a Close button. You select options that perform operations and click Close to close the dialog box.

The GUI displays command output, including processing messages and any warnings or error messages, in both the shell and the console log view.

The Design Vision documentation identifies commands with their menus in the following formats:

- *Menu* > *Command*
- *Menu* > *Submenu* > *Command*

where

- *Menu* represents a menu title on the menu bar
- *Submenu* represents a menu command that displays a submenu

  Some submenus contain commands that open other submenus.
- *Command* represents a command that performs an operation or displays a dialog box

Each menu command can also be activated by a shortcut key, which is indicated on the menu by an underscore (_) below a letter in the command and, if needed, the name of the modifier key (Shift or Ctrl) to the right of the command name. You can view a list of shortcut keys by choosing Help > Report Hotkey Bindings.

**See Also**

- Menu Bar
- Displaying the List of Keyboard Shortcuts

## Opening and Closing the GUI

You can open or close the GUI at any time during a Design Vision session. For example, you can open the GUI to perform visual analysis tasks or close the GUI to perform time-consuming tasks or batch processes in the shell. When you close the GUI, your designs remain loaded in memory and the command-line prompt remains active in the shell. If you reopen the GUI, a new Design Vision window appears.

When you open the GUI, it reads the GUI setup and preferences files and opens a new Design Vision window.

- The setup files perform basic setup tasks, such as initializing variables and declaring design libraries.
- The preference files set schematic and abstract clock graph view properties and global application preferences.

The Design Vision window contains the menus, toolbars, view windows, and panels that you use to perform timing analysis and other visual analysis tasks.

To learn how to open and close the GUI, see the following topics:

- Opening the GUI
- Closing the GUI

### Opening the GUI

You can open the GUI at any time during a Design Vision session. If you start a session with the GUI closed, you can open the GUI from the shell command line. Before you open the GUI, make sure that your `$DISPLAY` environment variable is set to your UNIX display name.

To open or reopen the GUI from the design_vision shell, enter the following command:

```
prompt> gui_start
```

When you open the GUI, either at startup or from within the shell command-line interface, the tool performs the following tasks:

1. Reads and executes commands from the Design Vision GUI setup files.
2. Opens the Design Vision window.

You can specify a Tcl script that you want to run when you open the GUI by using the `-file` option with the `gui_start` command. For example, to run the script from a file named my_gui_script.tcl, enter the following command:

```
prompt> gui_start -file my_gui_script.tcl
```

**See Also**

- Closing the GUI
- Starting the Tool

## Closing the GUI

You can close the GUI without exiting the tool at any time during the session. For example, if you need to save system resources, you can close the GUI and leave the tool running as a command-line interface.

To close the GUI without exiting the tool,

- Choose File > Close GUI.

Alternatively, you can enter the following command:

```
prompt> gui_stop
```

**See Also**

- Opening the GUI
- Saving Designs and Exiting Design Vision

# Using Script Files

You can use scripts to accomplish routine or repetitive tasks, such as setting constraints or defining other design attributes. You can use your existing Tcl scripts in the Design Vision command-line interface and the GUI.

You can create a script file by placing a sequence of Tcl commands in a text file. You can also define scripts in your setup files. Any dc_shell command can be executed within a script file.

To run scripts in the GUI,

- Choose File > Execute Scripts.

  The Execute File dialog box opens. Use the dialog box to navigate to the appropriate directory and run your script.

Alternatively, you can run scripts from the command line by using the `source` command. For information about this command, see the man page.

**See Also**

- *Using Tcl With Synopsys Tools*

## Saving Designs and Exiting Design Vision

You can exit the Design Vision tool at any time and return to the operating system. By default, the tool saves the session information in the command.log file. However, if you change the name of the log file using the `sh_command_log_file` variable after you start the tool, session information might be lost.

The tool does not automatically save the designs loaded in memory before exiting. To save these designs before exiting, use the Save or Save As command on the File menu or the `write_file` command on the command line.

To save the current design and each of its subdesigns in separate .ddc format files named *design_name*.ddc, where *design_name* is the name of the design,

- Choose File > Save.

To save the current design and all of its subdesigns in a single file with a different file name or file format,

1. Choose File > Save As.
2. Enter or select a file name.
3. Select a file format.
4. Click OK.

For more information about how to save your design, see “Saving Designs” on page 4-12.

To exit the tool, you can do any of the following:

- Choose File > Exit, and then click OK in the message box that appears.
- Enter `exit` or `quit` on the command line.
- Press Ctrl+C three times in the UNIX or Linux shell.

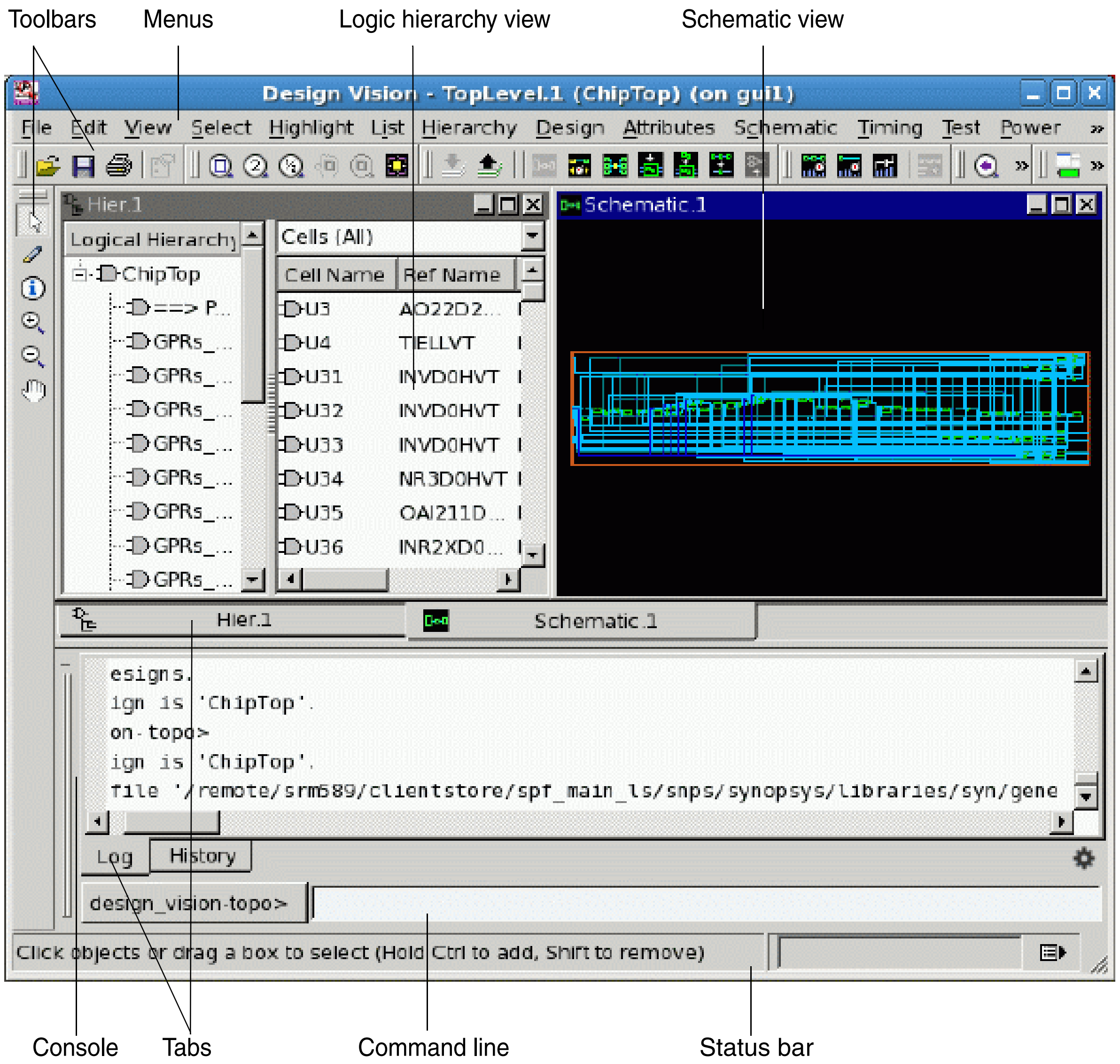
**See Also**

- Opening and Closing the GUI

## Graphical User Interface

The Design Vision window appears by default when you start the Design Vision tool. Figure 2-1 shows an example of the window you see when you start Design Vision, read in a design, and open a schematic view.

*Figure 2-1 The Design Vision Window*



| Cell Name | Ref Name |
|---|---|
| U3 | AO22D2... |
| U4 | TIELLVT |
| U31 | INVD0HVT |
| U32 | INVD0HVT |
| U33 | INVD0HVT |
| U34 | NR3D0HVT |
| U35 | OAI211D... |
| U36 | INR2XD0... |

```
esigns.
ign is 'ChipTop'.
on-topo>
ign is 'ChipTop'.
file '/remote/srm589/clientstore/spf_main_ls/snps/synopsys/libraries/syn/gene
```

design_vision-topo>

Click objects or drag a box to select (Hold Ctrl to add, Shift to remove)

The Design Vision window consists of a title bar, a menu bar, and several toolbars at the top of the window and a status bar at the bottom of the window. The title bar and menu bar are always visible. You can display or hide individual toolbars or the status bar.

The workspace area between the toolbars and the status bar displays view windows and panels. View windows provide graphic or textual views of design information. Panels provide interactive tools for setting options or performing often used tasks. View windows and panels can contain tabs with multiple views or pages. The “active view” is the view that has the mouse focus.

The hierarchy browser (logic hierarchy view) and the console appear in the Design Vision window by default. To visualize a design, you can open a schematic view. For information about these features, see the following topics:

- Design Vision Windows
- The Hierarchy Browser
- Schematic Views
- The Command Console
- Setting GUI Preferences

## Design Vision Windows

The Design Vision GUI displays information in application windows that you can move, resize, minimize, or maximize by using the window management tools on your UNIX or Linux desktop. Each window title bar lists the product name (Design Vision), the name of the window, and the name of the active view (the view window that has the mouse focus).

The GUI provides the following application windows:

- The Design Vision window appears automatically when you start a Design Vision session or open the GUI.
- The Layout window is available for visualizing the physical aspects of a design in the Design Compiler Graphical product when you start the tool in topographical mode.

You can open multiple instances of the Design Vision window or the Layout window and use them to compare views, or different design information within a view, side by side. The window name includes the unique instance number of the window.

All open application windows share the same designs in memory and the same current timing information. However, each window is independent of the other windows. You can configure the toolbars, status bar, view windows, and panels independently for each window. Design objects you select in one window are automatically selected in the other windows.

For more information about Design Vision windows, see the following sections:

- Menu Bar
- Toolbars
- Status Bar
- View Windows
- Panels

## Menu Bar

The menu bar contains menus with the commands you need to work in the window. Menu commands are grouped by function on the menus in each application window.

To choose a command on a menu bar menu, click the menu name to open the menu, and click the command name on the menu. You can display a brief message in the status bar about the action that a command performs by holding the pointer over the command name. For menu commands that can also be used by pressing a toolbar button or typing a keyboard shortcut, the menus show representations of those alternatives.

Note:

> If the window is not wide enough to display all the menu names on the menu bar, the window displays all the menu names that fit, from left to right, followed by an overflow button (»). To access the other menus, click the overflow button.

Some frequently used menu commands are also available on pop-up menus for individual views. To choose a command on a pop-up menu, move the pointer over the object of interest, right-click to display the menu, and click the command.

## Toolbars

Each application window provides toolbars with buttons you can use to quickly access frequently used operations or tasks. To determine the function of a toolbar button, hold the pointer over the button. A ToolTip displays the name of the button, and the status bar displays a brief description of its use. You cannot disable these messages.

Toolbars are always attached to a window edge. You can enhance your working environment by moving individual toolbars to different positions below the menu bar, or to the left, right, or bottom edge of the window. You can also disable a toolbar, hiding it from view.

If a window edge is not long enough to display all of the toolbars attached to it, the GUI displays the full toolbars that fit and shortened versions of the other toolbars. A shortened toolbar consists of a default toolbar button and an overflow button (»). To access the other toolbar buttons on a shortened toolbar, click the overflow button.

### See Also

- The “Toolbars” and “Configuring the GUI” topics in Design Vision Help

## Status Bar

Each application window displays a status bar at the bottom of the window. The status bar displays the information listed in Table 2-1.

*Table 2-1 Information Displayed by the Status Bar*

| When you do this | The status bar displays this information |
|---|---|
| Select one object | Object name |
| Select multiple objects | Number of selected objects |
| Hold the pointer over a menu command, toolbar button, or tab | Information about the action it performs |

You can quickly display the list of selected objects in the Selection List dialog box by clicking the ▤ button at the right end of the status bar.

You can hide or display the status bar in a window by choosing View > Status Bar.

### See Also

- The “Status Bar” and “Configuring the GUI” topics in Design Vision Help

## View Windows

View windows are child windows that display graphic or textual views of design information within the workspace area of an application window. When you click anywhere within a view window, the GUI highlights its title bar to indicate that it has the focus (that is, it is the active view) and can receive keyboard and mouse input.

View windows that contain multiple views provide a tab for each view. When you open a view window that has multiple views, it displays a default view. To change to a different view, you click its tab.

The GUI provides the following types of view windows:

- Graphic views (graphical descriptions of design information such as schematic, histogram, and layout views)

- Hierarchy views (for traversing hierarchical structures and gathering design information at different hierarchy levels)
- Text views (textual design information such as reports and object lists)

When you open the GUI, the logic hierarchy view appears in the workspace area of the Design Vision window. The analysis tasks that you perform during the session determine which other types of views you open. For information about the logic hierarchy view, see “Browsing the Design Hierarchy” on page 3-2.

You can adjust the sizes of view windows for viewing purposes, and you can move them to different locations within the workspace area. In addition, you can

- Arrange the open view windows by tiling or cascading them within the workspace area
- Minimize individual view windows, or maximize a view window to fill the workspace area

For more information, see the “Configuring the GUI” topic in Design Vision Help.

The GUI displays a tab at the bottom of the workspace area for each open view window. When you click a tab, the GUI displays its view window on top of the other view windows and makes it the active view. If a view window and a panel overlap on the screen, the panel appears on top of the view window.

### See Also

- The “View Windows” and “Configuring the GUI” topics in Design Vision Help

## Panels

Panels are enhanced toolbars that contain tools for setting options or performing frequent tasks while working with the design in view windows. Most panels are associated with a particular view and operate on the active view (the view that has the mouse focus). (An exception is the console, which contains a command line and its own views.) A tabbed panel contains tabs that you can click to access different tools or views. The first time you open a panel during a session, it displays the tools or view for its default tab.

When you open the GUI, the console is docked to the bottom edge of the Design Vision window and the other panels are hidden by default. For information about the console, see “The Command Console” on page 2-19.

You can adjust the sizes of panels for viewing purposes, and you can move them to different locations inside or outside the window. In addition, you can dock or undock individual panels by attaching them at edges of the window or separating them from the edge so they can float above or outside the window.

If a panel and a view window overlap on the screen, the panel appears above the view window.

### See Also

- The “Panels” and “Configuring the GUI” topics in Design Vision Help

## The Hierarchy Browser

The hierarchy browser (logic hierarchy view) appears by default when you open the GUI. Use it to navigate through your design, see the relationships among its levels, and gather object information. You can also select the designs or objects that you want to examine in graphic views or with other analysis tools.

The view window consists of an instance tree on the left and an object table on the right. When you read in a design, the instance name of the top-level design appears in the left pane. You can

- Click the expansion button (plus sign) for a hierarchical block (an instance that contains subblocks) to expand the instance tree, showing the names of the subblocks at the next level in the hierarchy
- Select an instance or hierarchical block to display information about the cells or other objects that it contains

The object table displays information about hierarchical cells by default. To facilitate your examination of the objects within an instance or hierarchical cell, you can select the type of objects that appear in the object table. You can display information about hierarchical cells, leaf cells, pins and ports, pins of child cells, and nets.

### See Also

- Browsing the Design Hierarchy
- The “Logic Hierarchy Views” topic in Design Vision Help

## Schematic Views

Schematic views are the primary tools for visualizing the design. You can use schematic views to analyze timing and logic in the optimized design and to gather information that can help you to guide later optimization operations. A schematic view shows graphic representations of design logic and timing paths in a design or subdesign.

You can create a schematic to

- Examine the logic elements (blocks and gates) and connectivity in the top-level design or a hierarchical cell

- Analyze timing problems by focusing on critical paths in your design
- Examine the power management cells in a multivoltage design

When you create a schematic that includes timing paths, the schematic shows the cells and nets on each path.

You can modify the viewing range and scale in a schematic view by using the interactive zoom and pan tools or the zoom and pan commands on the View menu. In addition, you can use the scroll arrows and scroll box in the schematic view window or the arrow keys on the keyboard to scroll vertically or horizontally through the schematic.

Note:

> Text does not appear in a schematic view when it is below a certain size in pixels. Use the zoom tools and zoom commands to magnify the view if necessary to see object names and annotations in a schematic.

You can select, highlight, and query objects in a schematic view. Objects that you select or highlight in a schematic view are automatically selected or highlighted in other views. This capability allows you to efficiently analyze the logic and timing aspects of your design.

By default, a schematic view displays timing paths and design logic in a flat, single-sheet schematic that can span multiple levels of hierarchy. Hierarchy crossing symbols (diamond shapes) indicate places where a path moves up or down a level in the hierarchy. Each timing path consists of the objects (cells, pins, and nets) that make up the path.

To focus on the area or objects that you need to examine in a schematic view, you can

- Hide or display buffer and inverter chains, buffer and inverter trees, or unconnected macro pins
- Hide or display the contents of hierarchical cells or blocks
- Reorganize the schematic hierarchically and display boundaries for the top-level design and each hierarchical cell or block

  In a multivoltage design with UPF power domains, you can color the boundaries based on the hierarchical power relationships of the design

In addition, you can

- Add or remove selected objects (cells, ports, or nets) in a schematic view
- Add fanin logic or fanout logic for selected objects
- Add the worst-case timing paths from, through, or to selected objects

The objects are added or removed only in the active schematic view. The netlist is not changed and other schematic views are not affected.

You can reverse and reapply changes that you make in a schematic view, such as adding logic, expanding a hierarchical cell, or displaying hierarchical boundaries. You can reverse the most recent action or sequentially reverse a series of actions. If you have reversed one or more actions, you can reapply the most recently reversed action or a series of actions.

You can customize a schematic view by setting options on the View Settings panel. You can change

- Object label or annotation visibility, text colors, or text sizes (click the Text tab)
- Object colors (click the Objects tab)
- The display style for highlighted timing paths (click the Settings tab)
- The color brightness

You can print the contents of the active schematic view or save an image of the view in a PDF or PostScript file for printing later from a UNIX or Linux shell.

If you change the netlist for a design (for example, by using netlist editing commands such as `change_link`) when a schematic view is open, the GUI immediately updates the schematic and maintains the current zoom level and pan position.

**See Also**

- Examining Hierarchical Cells
- Examining Synthetic Operators in GTECH Designs
- Examining Timing Paths and Selected Logic
- The “Using Schematic Views” topic in Design Vision Help

## The Command Console

When you start the tool and open the GUI, the command console is docked above the status bar by default. The console provides a command-line interface and two views; a log view that displays the session transcript (the default view) and a history view that displays the command history list.

You can use the console to

- Enter Design Compiler Tcl commands on the console command line
- View either the session transcript (log view) or the command history list (history view) by clicking the tabs above the command line
- Display an error message man page in the man page viewer by clicking the message number in the console log view

- Copy and edit or reuse commands
- Search for, select, and save commands or messages in the log view or the history view

You can open (or close) one console in each application window. When the console is open, you can dock it to the bottom or top of its window, or move it over or away from the window.

The console displays information about the commands you use during the session in the following views:

- Log view
- History view

To select a view, click the tab above the command line. The log view is displayed by default when you start the Design Vision tool or open the GUI.

You can copy text in the log view and paste it on the command line, the same way you would in a UNIX or Linux shell, by selecting the text with the left mouse button and pasting it with the middle button. You can also select commands in the history view and edit or reissue them on the command line.

You can display or hide the console, and you can increase or decrease its height. You can dock it to the top or bottom edge of the Design Vision window, or you can undock it and resize it or move it around or off its Design Vision window.

For information about the console, see the following sections:

- Console Log View
- Console History View
- Console Command-Line Editing

**See Also**

- Entering Tcl Commands in the GUI

## Console Log View

The console log view displays a transcript of session information that includes the commands you have entered and the Design Vision output and messages resulting from your commands. You can use this information to

- Check tool status after performing functions
- Troubleshoot problems that you encounter
- Look up information about past functions

You can reenter commands you have already used by copying them from the log view to the console command line.

The log view displays the commands you enter next to a boldfaced prompt. Warnings and error messages are noted with “Warning” or “Error” as the first word. If you need to see information not currently displayed in the display area of the log view, use the scroll bars to scroll through the session information.

You can choose commands on the Options menu (on the right side of the console above the command line) to

- Find text in the transcript
- Select and copy text in the transcript
- Search the transcript for commands or messages
- Save the transcript, selected text, or just the error and warning messages in a text file

### See Also

- The “Viewing the Session Log” topic in Design Vision Help

## Console History View

The console history view lists shell, menu, and dialog box commands you have used in the current session. You can use the history view in the following ways:

- See which commands you have used
- Find and reuse commands you have already used
- Copy commands in the list
- Save the list for later use

If you need to see information not currently displayed in the display area of the history view, use the scroll bars to scroll through the list of commands. Alternatively, you can enter the `history` command on the command line to display the list of commands in the log view.

You can set options in the Application Preferences dialog box to control which types of GUI commands are included in the history list. For information about GUI preferences, see “Setting GUI Preferences” on page 2-23.

### See Also

- The “Viewing the Command History” topic in Design Vision Help

## Console Command-Line Editing

You can display, edit, and reissue commands on the console command line by using the arrow keys to scroll up or down the command stack and to move the insertion point to the left or right on the command line. You can copy text in the log view and paste it on the command line the same way you would in a UNIX or Linux shell, by selecting the text with the left mouse button and pasting it with the middle button. You can also select commands in the history view and edit or reissue them on the command line.

If you need to enter a command or Tcl procedure that uses multiple lines, you can vertically expand the command line by either typing a backslash (\) at the end of a line and pressing Return or clicking in the command line and pressing Shift+Return. The command line automatically shrinks to a single line when you issue the command.

If you need to enter a command or Tcl procedure that uses multiple lines, you can expand the command line vertically by either typing a backslash (\) at the end of a line and pressing Return or clicking in the command line and pressing Shift+Return. The command line automatically shrinks to a single line when you issue the command.

To expand the command line to display multiple lines,

- Press Shift-Return or type a backslash (\) at the end of the line and press Return.

To shrink the command line to display a single line,

- Press Control-Return.

You can display, edit, and reuse commands on the console command line by using the arrow keys to scroll up or down the command stack and to move the insertion point to the left or right on the command line.

- To scroll up to the previous command in the command stack, press the Up Arrow key (or press Shift-Up Arrow when the command line displays multiple lines).
- To scroll down to the next command in the command stack, press the Down Arrow key (or press Shift-Down Arrow when the command line displays multiple lines).
- To move the insertion point to the left, press the Left Arrow key (or press the Home key to move the insertion point to the beginning of the line).
- To move the insertion point to the right, press the Right Arrow key (or press the End key to move the insertion point to the end of the line).
- To complete a partial command, option, or file name, press Tab.
- To display a list of command options, type the command name followed by a blank space and press Tab.
- To issue a command, press Return.

**See Also**

- The “Entering Commands in the Console” topic in Design Vision Help

### Setting GUI Preferences

At the beginning of the GUI session, the tool loads GUI preferences from your preferences file. The default system preferences are set for optimal tool operation and work well for most designs. However, if necessary, you can change GUI preferences during the session. You can set preferences that control how text appears in GUI windows and whether commands for selection or interactive operations appear in the session log. You can also set various global, schematic view, and layout view default controls.

To set GUI preferences,

1. Choose View > Preferences.

   The Application Preferences dialog box appears.

2. Select a category in the Categories tree.

   The page for that category appears.

3. Set options as needed.
4. Repeat steps 2 and 3 if you want to set options in a different category.
5. Click OK or Apply.

When you change preference settings during a GUI session, the tool automatically saves the new preference settings in the preferences file named .synopsys_dv_prefs.tcl in your home directory. The next time you start the Design Vision tool or open the GUI, it loads the preferences from this file.

For more information about GUI preferences, see the “Setting GUI Preferences” topic in Design Vision Help.

## Getting Help in the GUI

The GUI provides a variety of user-assistance tools. The following online information resources are available while you are using the Design Vision tool:

- Command help, which is a list of options and arguments used with a specified dc_shell command, displayed in the shell and in the console log view when the GUI is open
- Man pages displayed in the shell and in the console log view when the GUI is open

- A man page viewer in the GUI that displays command, variable, and error message man pages that you request while using the GUI
- A report that lists the keyboard shortcuts you can use in the GUI
- An online Help system in a Web browser that explains how to use the GUI

For information about using these tools, see the following topics:

- Getting Help on the Command Line
- Displaying the List of Keyboard Shortcuts
- Viewing Man Pages
- Viewing the Help System

## Getting Help on the Command Line

The GUI provides three levels of command help on the console command line:

- A list of commands
- Command usage help
- Topic help

To get a list of all Tcl commands, enter the command:

```
prompt> help
```

To get help about a particular Tcl command, enter the command name with the `-help` option. The syntax is

```
prompt> command_name -help
```

To get topic help for a Tcl command, variable, or variable group, enter

```
prompt> man topic
```

Replace *`topic`* with the name of a Tcl command, variable, or variable group. By using the `man` command, you can display the man pages for the topic while you are interactively running the tool. In the GUI, you can view topic help in the man page viewer by choosing Help > Man Pages. For information about viewing man pages, see “Viewing Man Pages” on page 2-25.

### See Also

- Entering Tcl Commands in the GUI

## Displaying the List of Keyboard Shortcuts

You can view a report of the keyboard shortcuts for Tcl commands and commands on menus in the active window. The tool displays the hotkeys report in a report view.

To display the report of keyboard shortcuts for the active window,

- Choose Help > Report Hotkey Bindings.

The report contains the following columns:

- Hot Key lists the shortcut keys or key combinations
- Type indicates whether the key is a shortcut for a menu command or a Tcl command
- Function lists the commands that the shortcut keys launch

### See Also

- Choosing Menu Commands in GUI Windows

## Viewing Man Pages

The Design Vision GUI provides an HTML-based browser window that lets you view, search, and print man pages for commands, variables, and error messages. You can use it to

- Display a man page
- Search for text on the man page you are viewing
- Print the man page you are viewing
- Browse back and forth between man pages you have already viewed

To view a man page in the man page viewer,

1. Choose Help > Man Pages.

   The man page viewer appears. The home page displays a list of links for the different man page categories.

2. Click the category link for the type of man page you want to view.

   The choices are Commands, Variables, and Messages. The contents page for the category displays a list of title links for the man pages in that category.

3. Click the title link for the man page you want to view.

You can also display man pages in the man page viewer by using the `man` command or the `gui_show_man_page` command on the console command line.

Note:

If you enter the `gui_show_man_page` command in the shell when the GUI is closed, the tool automatically opens the GUI and displays the man page in the man page viewer.

You can browse back and forth between pages you previously viewed the same way you browse Web pages in a Web browser.

**See Also**

- The “Viewing Man Pages” topic in Design Vision Help

## Viewing the Help System

Design Vision Help is a browser-based HTML Help system that provides detailed information and instructions for using the GUI. You can use Design Vision Help to

- Learn about tool features, including windows, views, toolbars, panels, menus, and dialog boxes
- Learn how to use the interactive visualization and analysis tools
- Learn how to use the Design Vision tool to perform synthesis tasks

You can open Design Vision Help from the GUI or standalone in your Web browser. When you open the Help system from the GUI, the browser executable file must be specified in your UNIX or Linux `$PATH` variable.

To open Design Vision Help from the GUI,

1. Choose Help > Online Help.

   The Help system appears in a Web browser window and displays the Welcome topic for Design Vision Help.

2. Use the Help system navigation tools to find the information you need in one of the following ways:

   ❍ Browse for a topic in the navigation pane on the left side of the browser window by expanding the appropriate books until you find the information you need.

   ❍ Find the topic by its subject by clicking the Index tab above the navigation pane and looking for the subject in the alphabetical listing.

   ❍ Search for text in Help topics by typing the text in the search box (labeled “Search Design Vision Help”) and clicking the Search button.

   The search results (list of topics found) appears in the topic pane on the right side of the browser window. Select the appropriate topic by clicking its link. The search terms are marked with color highlights.

Design Vision Help lets you search for text in any Help topic. By default, the search mechanism searches for one or more words you type in the search box. To search for an exact phrase containing two or more words, enclose the words within double quotation marks (" ").

Searches are not case-sensitive. The search results are ranked according to the location of the matched term and the number of matches. Matches in headings always rank near the top.

### See Also

- The “Using This Help System” topic in Design Vision Help

# 3

# Using Visual Analysis Tools

The Design Vision GUI provides a variety of tools that you can use to visualize design data and analyze results for the Design Compiler, DC Explorer, DFT Compiler, and Power Compiler tools. The GUI also supports the Design Compiler topographical technology. In the Design Compiler Graphical tool, the GUI provides visual analysis tools that can help you to analyze and debug physical problems related to Design Compiler topographical synthesis. Before you start using the GUI to analyze and troubleshoot a design, you should become familiar with the operation of the GUI and the various tools that it provides.

To learn the general and specific information you need to know before you use the Design Vision tool for the first time, see the following topics:

- Exploring the Design
- Analyzing Design Timing
- Viewing the Floorplan in Design Compiler Graphical
- Using DFT Analysis Tools
- Defining and Viewing the Power Intent for Multivoltage Designs
- Analyzing Multivoltage Design Problems

# Exploring the Design

You can use the hierarchy browser (logic hierarchy view) and schematic views to explore the design and examine design information. You can also select objects that you want to examine with other analysis tools. You can view information about design objects by

- Selecting an object and viewing its properties in the Properties dialog box
- Viewing object information in object list views
- Viewing object reports that you generate by choosing commands on the Design and Timing menus

You can view a list of the selected objects in the Selection List dialog box. It displays the names and object types of all the objects in the current selection. When you select objects in other views, their names automatically appear in the selection list.

For information about these tools, see the following sections:

- Browsing the Design Hierarchy
- Examining Hierarchical Cells
- Examining Synthetic Operators in GTECH Designs
- Viewing the Selection List
- Viewing Object Lists
- Viewing and Editing Object Properties
- Cross-Probing the RTL for Cells and Timing Paths
- Cross-Probing From RTL to Gates

## Browsing the Design Hierarchy

You can use the logic hierarchy view to browse the complete hierarchical structure of the current design and observe how many hierarchical blocks are present, the size of each block, and whether any DesignWare components have been inferred (used). If you are not familiar with a design, you can explore the design hierarchy to understand its structure and gather information about objects (cells, nets, or pins) in the design. You can also select the names of designs or objects you want to examine in graphic views or with other analysis tools.

To explore the design hierarchy, you can

- Select the names of hierarchical blocks (instances that contain subblocks) to display information about the cells in the block
- Click the expansion buttons next to the names of hierarchical blocks to further expand the instance tree
- Use the arrow keys to navigate the instance tree

  Press the Up Arrow and Down Arrow keys to move up or down the tree, the Right Arrow key to expand a cell and the Left Arrow key to collapse a cell.

You can select objects in the instance tree or the object table that you want to examine with other analysis tools. The objects you select are automatically selected in other views. For example, if you want to examine a schematic representation of a hierarchical cell, select the cell in the hierarchy browser and choose Schematic > New Schematic View. For details about using schematics, see “Examining Hierarchical Cells” on page 3-3.

By default, the object table contains cell information. You can select an object type in the list above the table to display information about hierarchical cells, all cells, pins, pins of child cells, or nets.

- Cell information includes the cell instance names, the reference names of the designs the cells reference, the paths from the top-level designs to the cells, and the values of the `dont_touch` attribute.
- Pin information includes the pin and port names and the paths from the top-level design to the pins and ports.
- Net information includes the net names and their paths from the top-level design.

You can sort the object table and resize columns in the table. You can also filter the table, limiting it to information based on a character string or regular expression that you define.

### See Also

- The “Browsing the Design Hierarchy” topic in Design Vision Help

## Examining Hierarchical Cells

You can use schematic views to examine logic connectivity and timing paths in the optimized design and to gather information that can help you to guide later optimization operations. Schematic views show graphic representations of design logic and timing paths in your design.

A schematic view displays the design objects and timing paths in a flat, single-sheet schematic that can span multiple hierarchy levels.

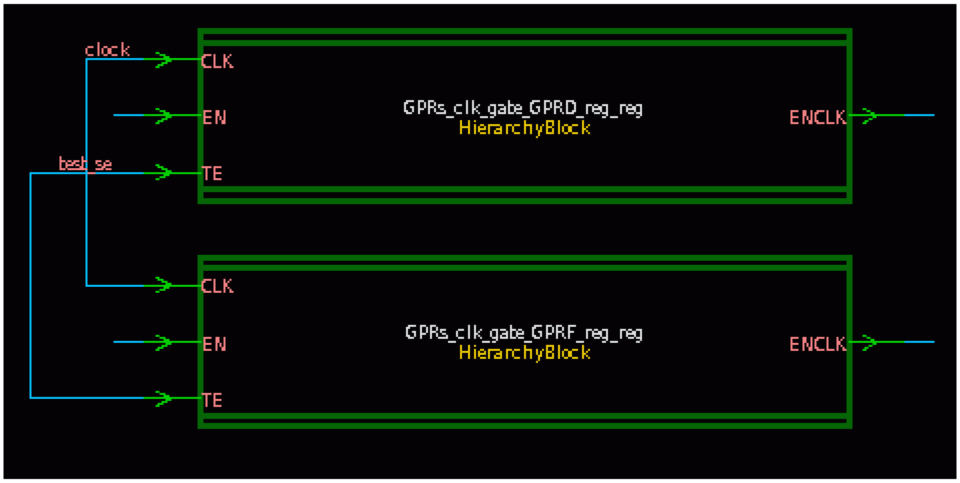
- A schematic can contain instances (cells), ports, pins, nets, buses, bus rippers, and hierarchy crossings.

  An instance can be a block (a hierarchical cell representing the top-level design or a subdesign) or a leaf cell.
- A hierarchy (diamond shape) crossing indicates a place where a timing path traverses the design hierarchy.
- A selected or highlighted timing path appears as a series of flylines between pins or a pin and a port.

When you create a schematic that contains hierarchical cells, the cells initially appear as collapsed metacells. For a set of objects that have a common hierarchical parent, the hierarchy metacell is similar to a design cell but has a thicker line width and a different color.

Figure 3-1 shows an example of a schematic with two hierarchical cells collapsed into metacells.

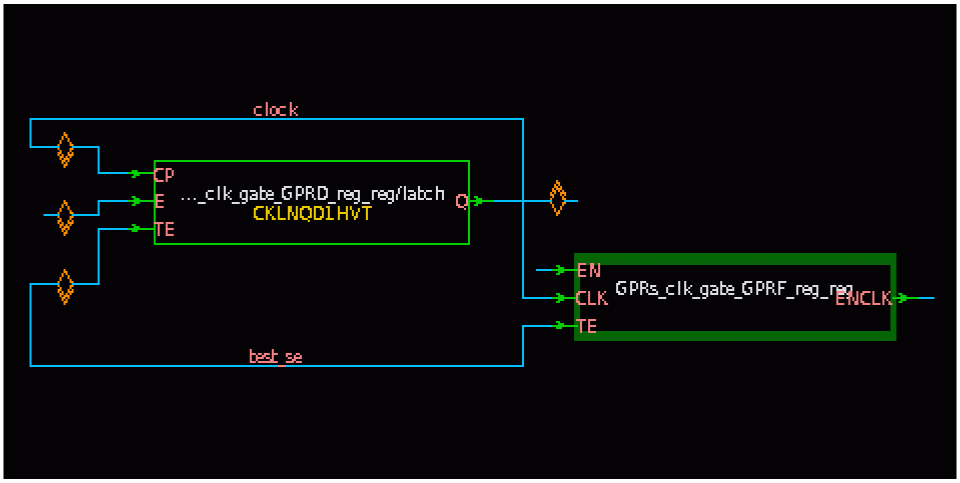
*Figure 3-1 Hierarchical Metacells*



You can expand a hierarchy metacell to display the objects in the next hierarchy level. If it includes further levels of hierarchy, they appear as metacells.

Figure 3-2 shows the same schematic with one hierarchical cell expanded to display its contents.

*Figure 3-2 Expanded Hierarchical Cell*

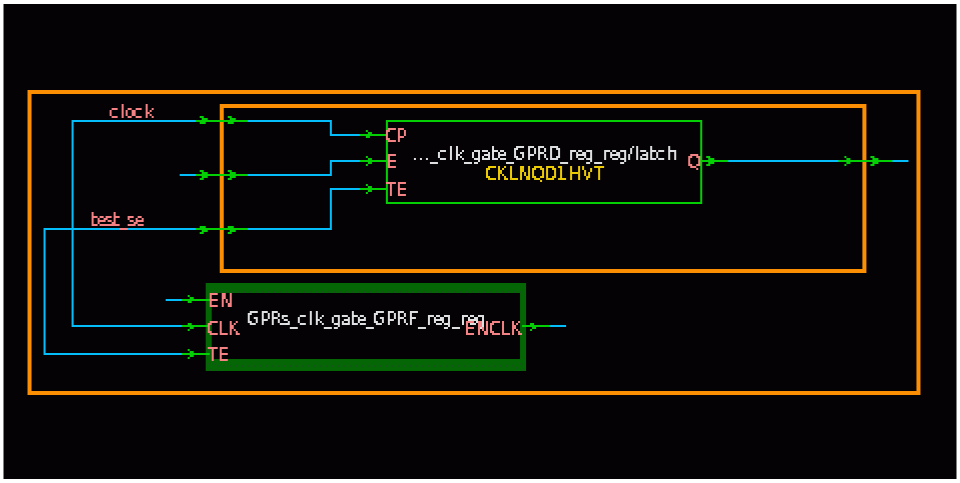


By default, a schematic displays the objects with input ports on the left and output ports on the right. You can reorganize the schematic hierarchically and display rectangular boundaries for the top-level design and each hierarchical set of objects.

You can reorganize the schematic hierarchically and display rectangular boundaries for the top-level design and each hierarchical set of objects.

Figure 3-3 shows an example of a schematic organized hierarchically with visible boundaries.

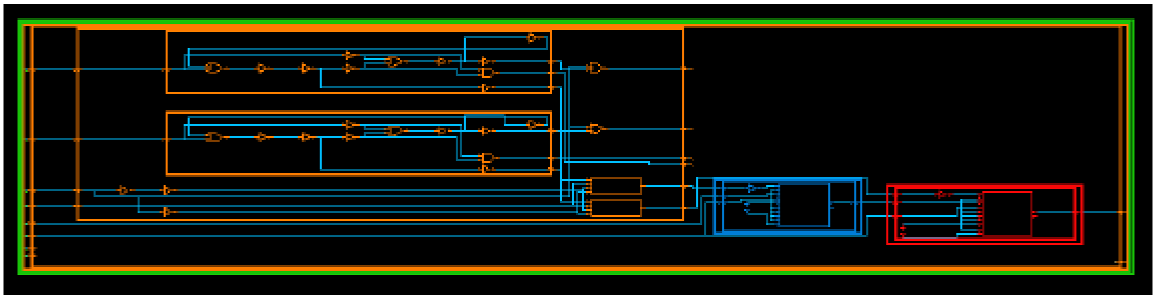
*Figure 3-3 Hierarchical Cell Boundaries*



The tool rearranges the schematic so that objects are placed hierarchically, which puts objects that share the same hierarchical parent near each other. The logic hierarchy boundaries are orange, and the power domain boundaries are yellow.

In a multivoltage design with UPF power domains, you can color the boundaries based on the hierarchical power relationships of the design. Figure 3-3 shows an example of a schematic organized hierarchically with colored boundaries.

*Figure 3-4 Colored Power Domain Boundaries*



You can also traverse the design hierarchy within the schematic view by moving down into the schematic for a block (subdesign) at the next lower level of the hierarchy or by moving up (from a subdesign) into the schematic for the hierarchical (parent) cell at the next higher level of the hierarchy.

In addition, you can

- Hide or display long chains or trees of buffers, inverters, and hierarchy crossings
- Hide or display unconnected macro pins
- Collapse or expand bus nets and pins
- Add or remove selected objects or add fanin logic, fanout logic, or timing paths for selected objects
- Trace fanin or fanout connections with highlighting for selected cells, pins, or ports

You can reverse and reapply changes that you make in a schematic view, such as adding logic, expanding a hierarchical cell, or displaying hierarchical boundaries. You can reverse the most recent action or sequentially reverse a series of actions. If you have reversed one or more actions, you can reapply the most recently reversed action or a series of actions.

**See Also**

- Schematic Views
- The “Examining Hierarchical Cells” topic in Design Vision Help

## Examining Synthetic Operators in GTECH Designs

Schematic views generate standard symbols to display synthetic operators in a GTECH netlist. These operator symbols can help you to identify the associated functions and to analyze and debug operators and datapaths. The same symbols are used for both signed and unsigned operators.

These symbols are generated automatically for the following operators, with the appropriate drawing geometry and pin sets to match the specific cell:

- Add (+), subtract (-), multiply (x), and divide (/) operators
- Less than (<), greater than (>), equal to (=), and not equal to (/=) operators
- Less than or equal to (<=) and greater than or equal to (=>) operators
- Multiplexor and selector operators

The symbols appear only in GTECH designs. They are not visible in synthesized designs even when the operator hierarchies are preserved and are not supported for instantiated DesignWare modules.

### See Also

- Schematic Views
- The "Synthetic Operators in Schematic Views" topic in Design Vision Help

## Viewing the Selection List

The selection list displays the names and object types of all selected objects in the current design. When you select objects in other views, their names automatically appear in the selection list.

The Selection List dialog box contains a table with a row for each selected object and columns that display the object name, the object type, and certain attribute values for each selected object.

You can control which attributes appear in the table by selecting an attribute group. By default, the table displays values for the attributes in the Basic attribute group. You can modify the contents of some predefined attribute groups and create or modify custom attribute groups by using the Attribute Group Manager dialog box.

You can deselect object names and remove their names from the selection list. You can also filter the selection list, limiting it to information based on a character string or regular expression that you define.

If you want to save the object or timing path information in the selection list, you can export the entire list to a text file. The file is formatted with each object or path on a separate line and the column data delimited by commas.

### See Also

- The “Viewing the Selection List” topic in Design Vision Help

## Viewing Object Lists

You can generate a list of objects (cells, nets, or ports and pins) and display information about them in a list view. You can display information about

- Selected objects (such as selected cells)
- Objects related to other selected objects (such as pins of selected cells)
- Objects with a common function or attribute (such as fixed cells)

You can select some or all of the objects in a list by clicking or dragging the pointer across their names in the list view. You can also use Shift-click or Control-click to select multiple objects. In addition, you can

- Sort the information alphabetically by clicking a column heading. Click the heading again to reverse the sort
- Scroll up and down in the table by pressing the Up Arrow and Down Arrow keys
- Filter the objects listed in the table, limiting it to objects based on a character string or regular expression that you define

If some of the text in a column is missing because the column is too narrow, you can hold the pointer over it to display the information in an InfoTip. You can also adjust the width of a column by dragging its right edge left or right.

You can filter an object list, limiting it to designs based on a character string or regular expression that you define, by using the Filter List dialog box.

### See Also

- The “Viewing Collections of Objects” topic in Design Vision Help
- The “Filtering Object Lists” topic in Design Vision Help

## Viewing and Editing Object Properties

You can view attributes and other object properties for selected designs, design objects, or timing paths by using the Properties dialog box. You can also set, change, or remove the attribute values for certain properties.

The Properties dialog box lists the object properties in a table with two columns (for property names and property values) and a row for each property.

Note:

> The Properties dialog box displays properties for the objects that are currently selected. If you change the current selection when the Properties dialog box is open, the dialog box changes to display the properties for the newly selected objects.

The properties you can view include object names, attribute values, and certain timing and placement values. The list of properties differs from one object type to another. To control which properties appear in the Properties dialog box, you select an option in the “Attribute group” list.

Note:

> Timing values do not appear until you perform an operation that updates timing information, such as generating a timing report or opening a histogram.

You can modify the contents of some attribute groups and create custom, user-defined attribute groups, by using the Attribute Group Manager dialog box. You can also view the contents of the predefined attribute groups.

If you select multiple objects, the property lists are displayed separately for each object. You can click the previous and next arrow buttons to navigate from one list of object properties to another. Alternatively, you can select an option to list the property values for all the selected objects together in a single table.

Some object properties are attributes that you can edit by changing or removing their values or by applying values if they are not already assigned. A bold border in the value column indicates an editable property value.

For more information about object properties and attribute groups, see the “Viewing and Editing Object Properties” and “Using Attribute Groups” topics in Design Vision Help.

### See Also

- the “Viewing and Editing Object Properties” topic in Design Vision Help
- the “Using Attribute Groups” and “Creating and Editing Attribute Groups” topics in Design Vision Help

## Cross-Probing the RTL for Cells and Timing Paths

You can analyze timing and congestion problems in an elaborated or compiled design by cross-probing the RTL for selected objects and timing paths. RTL cross-probing can help you to identify where certain cells originate in the RTL and to find and debug problems during timing analysis, congestion analysis, or datapath analysis.

Note:

> The design must be elaborated or compiled using Design Compiler version I-2013.12 or later, and the design files must remain unmodified and in the same locations.

You can cross-probe the RTL for cells and timing paths that you select in other views. You can also cross-probe cells in highly congested areas of the design, cells in a design resources report, and synthetic operators in a datapath extraction report. You can cross-probe the RTL in VHDL, Verilog, and SystemVerilog files.

Cross-probing cells can help you identify where certain gates originate in the RTL.

- During timing analysis, cross-probing a critical path can help you to debug worst-case timing problems.
- During congestion analysis, cross-probing cells in highly congested areas helps you identify the RTL code that is causing the congestion.
- During datapath analysis, RTL links in design resource reports can help you to debug datapaths during worst-case path analysis.

  This helps you understand the datapath extraction, which can help you to improve the coding style for better extraction.

You use the RTL browser to examine the RTL for cells and timing paths that you cross-probe from other views. You can view, select, and copy text in the RTL browser, but you cannot edit the RTL file. You can also use the RTL browser to cross-probe from the RTL to gates.

Note:

> The GUI does not support cross-probing from nets or cross-probing to UPF files.

The RTL browser window contains two panes. The RTL chooser appears in the top pane, and the RTL text view appears in the bottom pane. The RTL chooser lists the RTL file names, line numbers, and cell names for each cell in an expanding tree view. The RTL text view displays the content of the RTL files. The RTL text view is always visible; the RTL chooser appears only when you cross-probe cells or timing paths.

When you cross-probe a cell, the tool attempts to locate the RTL file in which the cell originates and opens the file in the RTL browser. When you cross-probe a timing path, the tool creates a collection of all the cells on the path, attempts to locate the RTL files in which the cells originate, and opens the files in the RTL browser.

If the tool cannot find the RTL file for a cell, the RTL browser displays the following message:

```
Unable to open RTL File:
Cross Probe has no RTL File Associated
```

If you cross-probe multiple cells and the tool cannot find the files for some of the cells, the RTL chooser displays “<No RTL Source>” instead of the file name for each file it could not find.

The RTL chooser displays file names, line numbers, and cell names in an expanding tree view. You can expand a file name or line number by double-clicking the name or number or by clicking its expansion button (plus sign). When a cell has multiple sources, the additional source files are listed below the cell name, preceded by the words “Alternate Source.”

The first file name is expanded by default to show all of the line numbers and cell names under it.

- If just one cell name appears under the expanded file name, the RTL text view displays the line for that cell.
- If multiple cell names appear under the expanded file name, the RTL text view displays the first line of the file.

The RTL text view displays the RTL for the file name, line number, or cell name that you select in the RTL chooser.

- To display a file, click the file name.

  The first time you clock a file name, the RTL text view displays the file beginning at line 1 and an icon with the file name appears at the top of the pane.
- To display the RTL for a cell, click its line number or cell name.

  The RTL text view displays the line where the cell originates in the RTL file.

In general, when you select a cell name in the RTL chooser, the RTL text view displays the line in the file where the cell is defined. The tool always attempts to maintain the accuracy of the cross probe. For cells that pass through complex optimization steps, such as ungrouping, boundary optimization, or mapping to complex cells, the tool points to the `always` block of the RTL file.

You can configure the RTL browser by setting options at the top of the RTL chooser to hide or display the RTL chooser, enable or disable the follow-selection mechanism, and reuse the RTL browser window. The columns in the RTL chooser provide the following information:

- The file names, line numbers, and cell instance names
- The number of cells in a file
- The line number, origin, and reference name for each cell

You can select and copy file names, line numbers, and cell names in the RTL chooser or text in the RTL text view. You can also add or remove markers on lines of text and display or hide blocks of code, such as a module, an `always` block, or an `if` statement. The RTL browser also provides tools that you can use to open a design resources report, to find text in the RTL file or a linked resources report, or to search for and select cells by name.

You can cross-probe cells in the current design from a line in the RTL file that you used to elaborate or compile the design. The tool selects the cells associated with the line and displays them in the Selection List dialog box. You can open the RTL file by choosing AnalyzeRTL > Open RTL Files and selecting the file in the Open RTL Files dialog box.

When you cross-probe cells from an RTL file, the tool opens the Selection List dialog box and colors the cells with the selection color in schematic and layout views, but it does not open a new schematic or layout view to display the cells. You can perform operations on the selected cells, such as creating a schematic of the selected logic.

### See Also

- Cross-Probing From RTL to Gates
- The “Analyzing RTL” topic in Design Vision Help

## Cross-Probing From RTL to Gates

You can cross-probe cells in the current design from a line in the RTL file that you used to elaborate or compile the design. Lines that you can cross-probe are highlighted with a green background. The tool selects the cells associated with the line in the RTL, opens the Selection List dialog box, and colors the cells with the selection color in schematic and layout views, but it does not open a new schematic or layout view to display the cells. You can perform operations on the selected cells, such as creating a schematic of the selected logic.

You can open an RTL file by choosing AnalyzeRTL > Open RTL Files and selecting the file in the Open RTL Files dialog box. The RTL browser window displays just the RTL text view and not the RTL chooser. The title bar displays the name and location of the RTL file.

When you open an RTL file that was used to elaborate or compile the design, you can cross-probe from lines in the RTL to cells in the current design. You can also perform other RTL browsing tasks, such as finding text in the RTL file, selecting and copying text, adding or removing markers, or finding text in an open design resources or datapath extraction report.

If the RTL files that were used to elaborate or compile the design have been moved or their directory has been renamed, you can use the `update_cross_probing_files` command to update the location or directory name. To learn the status of the RTL files, use the `report_cross_probing_files` command. For information about using these commands, see their man pages or the *Design Compiler User Guide*.

If you open an RTL file that has been moved or that was not used to compile the design, the RTL browser displays the text on a gray background. You can select and copy text and add or remove markers, but you cannot cross-probe cells from the RTL or perform other RTL browsing tasks in the file.

**See Also**

- Cross-Probing the RTL for Cells and Timing Paths
- The "Cross-Probing From RTL to Gates" and "Opening RTL Files" topics in Design Vision Help

## Analyzing Design Timing

The Design Vision GUI provides views you can use for both high-level analysis of overall timing in the design and detailed analysis of individual timing paths and object connectivity. For information about the different kinds of timing and design analysis views you can use, see the following sections:

- Viewing High-Level Timing Results
- Analyzing Timing Path Collections
- Examining Timing Paths and Selected Logic
- Profiling Path Delays
- Examining Timing Path Details
- Inspecting Timing Path Elements

### Viewing High-Level Timing Results

Histograms provide high-level views of design timing for visual timing analysis. You can use histograms to view the overall timing performance of your logic design and to select individual timing paths for further study in timing reports or other analysis views.

The GUI provides the following predefined histograms: endpoint slack, path slack, and net capacitance.

- Endpoint slack histograms show a distribution of timing slack values for all endpoints in the design.

  You can choose maximum or minimum delay (setup or hold). The slack distribution provides an overall picture of how close the design is to meeting requirements.

- Path slack histograms show a distribution of timing slack values for selected paths or for the paths with the worst slack in the design.

  You can select a maximum or minimum delay type (setup or hold), set the maximum number of paths and the number of worst paths per endpoint, and select a path group. You can also specify individual paths to, from, or through selected objects (similar to the way you specify paths for timing reports).

- Net capacitance histograms show a distribution of net capacitance values for selected nets or for all nets in the design.

When you use the timing analysis driver window to view timing path details, you can also generate histograms that show the distribution of values for certain types of path details listed in the window.

**See Also**

- The “Viewing Histograms” topic in Design Vision Help

## Analyzing Timing Path Collections

You can use the path analyzer to analyze a collection of timing paths and determine where timing failures occur in the design. The path analyzer analyzes the paths using a rule that you specify, categorizes them based on available attribute values, and displays the categories in a color-coded treemap view. You can select a predefined category rule or define a custom category rule. You can also add subcategories.

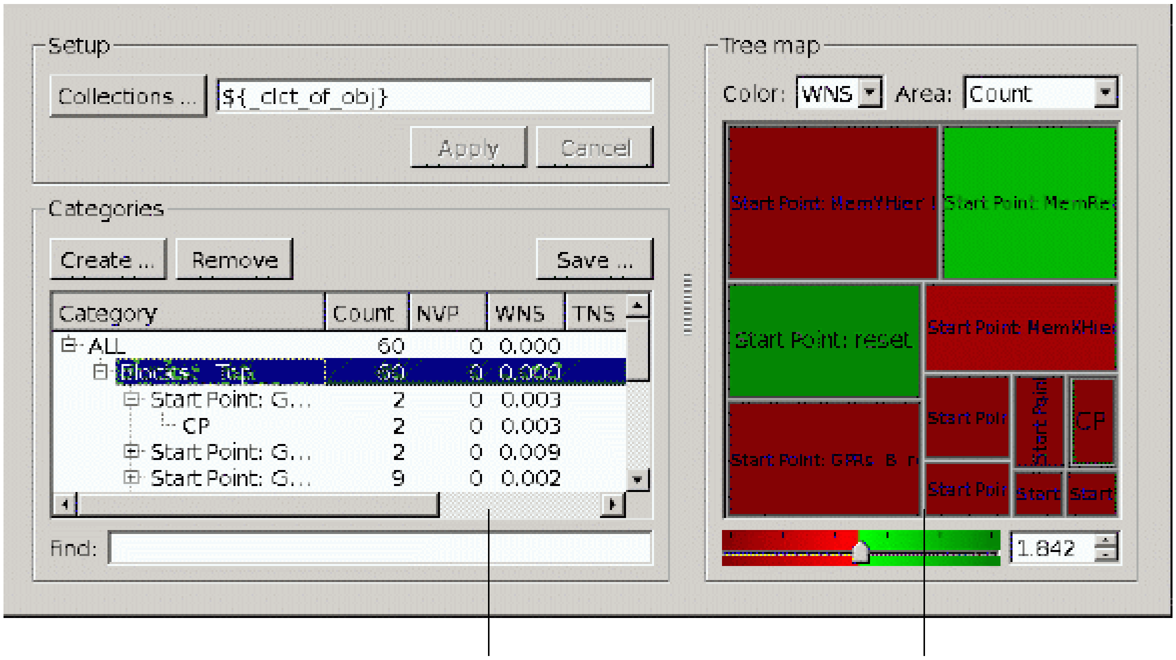
The path analyzer allows you to perform custom trend analysis on collections of timing paths. For example, you can

- Categorize paths by path group to see if timing violations are specific to a particular path group
- Create a custom category rule for paths with slack violations of more than one clock cycle to determine whether you need to specify multicycle path constraints on the paths
- Categorize the paths by their start clock and end clock values to identify cross-domain paths, which are the paths where these values are not the same

You can also tag appropriate blocks with block marks, and then categorize the paths by using the block mark attribute. This allows you to see if timing path failures result from a certain block in the design.

Figure 3-5 shows an example of the path analyzer after the path collections have been loaded and categorized.

*Figure 3-5 Path Analyzer Window*



${_dct_of_obj}

| Category | Count | NVP | WNS | TNS |
|---|---|---|---|---|
| ALL | 60 | 0 | 0.000 | |
| Blocks: Top | 60 | 0 | 0.000 | |
| Start Point: G... | 2 | 0 | 0.003 | |
| CP | 2 | 0 | 0.003 | |
| Start Point: G... | 2 | 0 | 0.009 | |
| Start Point: G... | 9 | 0 | 0.002 | |

Find:

Color: WNS Area: Count

1.842

Category Tree Treemap View

The category tree on the left side of the window displays the categories and subcategories. Each row represents a category or subcategory. For each category, the columns display the following information:

- Category displays the category name.
- Count displays the number of paths in the category.
- NVP displays the number of violating paths.
- WNS displays the worst negative slack value.
- TNS displays the total negative slack value.
- TPS displays the total positive slack value.

To view the categories in the treemap, select All in the category tree. To view the subcategories for a category, select the category.

The treemap view on the right side of the window displays hierarchical data as a set of nested rectangles. Each category has its own rectangle, which can be tiled with smaller rectangles that represent subcategories.

- The area of a rectangle represents one dimension of the data, such as the total number of paths or the number of violating paths.
- The color of a rectangle represents the worst negative slack in the category. Red means the slack value is lower than the threshold value set at the bottom of the treemap view, and green means the slack value is higher than the threshold value. These colors also change from light to dark to indicate how far the WNS value is from the threshold value.

You can save the categorized paths in a Tcl script file that you can use later to reload the paths.

**See Also**

- The “Analyzing Timing Paths” topic in Design Vision Help

## Examining Timing Paths and Selected Logic

You can visually examine timing paths in your design by creating a schematic to display selected timing paths, design objects, or both. You can select the paths in another timing analysis tool, such as a histogram, the timing analysis driver, a path data table, or the path inspector.

A schematic view displays individual timing paths or design logic in a flat, single-sheet schematic that can span multiple levels of hierarchy. Each timing path consists of the objects (cells, pins, and nets) that make up the path. A selected path appears as a series of flylines between pins or a pin and a port. Hierarchy crossings indicate places where a path traverses the design hierarchy.

You can view information about an object by holding the pointer over the object. The information appears in an InfoTip. The information content depends on the object type.

- Cell information can include the cell instance name and the full (hierarchical) cell name plus information about each input or output pin on the cell.
- Pin information can include the full (hierarchical) pin name, the pin direction, and the arrival time, transition time, and slack time values.
- Net information can include the full (hierarchical) net name, the total net capacitance value, the number of local fanouts, and the total number of fanouts.

  Local fanouts are the loads that directly connect to the net when pin directions are considered. Total fanouts are all the loads on the net that are driven by the same source or driver, regardless of the directions of hierarchical pins. Total fanout is the same as the fanout number provided in the net report.

- Port information can include the full (hierarchical) port name, the port direction, and the arrival time, transition time, and slack time values.
- Hierarchy crossing information can include the direction of the crossing (down a level into a subdesign or up a level to the parent design) and the full (hierarchical) name of the subdesign or parent design.

Note:

InfoTips are disabled by default. To enable InfoTips, choose View > InfoTips.

A timing path can include long chains of buffers or inverters and multiple hierarchy crossings. To avoid examining the progression of a signal across buffer and inverter chains or through unimportant blocks, you can hide some objects by collapsing them into abstract metacells.

You can collapse buffer and inverter chains, buffer and inverter trees, or the objects in hierarchical blocks. You can also collapse unconnected pins into metapins.

- For a buffer or inverter chain or a buffer or inverter tree that results in a noninverted output, the metacell is similar to a buffer but has a thicker line width and darker color.
- For a buffer or inverter chain or a buffer or inverter tree that results in an inverted output, the metacell is similar to an inverter but has a thicker line width and darker color.
- For a buffer or inverter tree that results in both noninverted and inverted outputs, the metacell combines the appearance of both an inverter and a buffer.

  The symbol has both inverted and noninverted outputs with the loads of the chain connected to the appropriate polarity output.

You can add or remove selected logic (cells, ports, or nets) in a schematic. You can also add fanin logic, fanout logic, or worst-case timing paths for objects that you select in a schematic. The tool makes these changes only in the active schematic view. The netlist is not changed and other schematic views are not affected.

When you add fanin logic, fanout logic, or timing paths, you can control whether the additions appear in the active schematic view or in a new schematic view. You can

- Add the logic or paths to the schematic in the active schematic view
- Display only the selected objects and the additional logic or paths in the active schematic view
- Add the logic or paths to the schematic and display it in a new schematic view
- Display only the selected objects and the additional logic or paths in a new schematic view

You can reorganize a schematic hierarchically and display rectangular boundaries for the top-level design and each hierarchical set of objects. In a multivoltage design with UPF power domains, you can color the boundaries based on the hierarchical power relationships of the design.

You can reverse and reapply changes that you make in a schematic view, such as adding logic, expanding a hierarchical cell, or displaying hierarchical boundaries. You can reverse the most recent action or sequentially reverse a series of actions. If you have reversed one or more actions, you can reapply the most recently reversed action or a series of actions.

**See Also**

- Schematic Views
- The “Viewing Timing Paths and Connected Logic” topic in Design Vision Help

## Profiling Path Delays

Path profile views help you examine the contributions of individual cells and nets to the total delay of a timing path.

The view window contains a table that shows path and pin data.

- For each path, the table displays the path name, total delay time, relative pin delay contributions, and full path name.
- For each pin on each path, the table displays the pin name, individual delay time, relative contribution, clock edge (rising or falling), and full pin name.

The combined delays for each path and the relative delay contribution for each pin appear graphically in bar graphs that represent the percentages of the total path delay.

**See Also**

- The “Viewing Path Profiles” topic in Design Vision Help

## Examining Timing Path Details

The timing analysis driver and path data tables provide detailed information about timing paths in your design. You can examine path details, such as attribute values, generate histograms based on path attributes, and select paths for in other tools, such as a schematic view or the path inspector.

Both the timing analysis driver and a path data table contain a timing path table and a button bar.

- The timing path table displays a list of paths that you specify. The table columns show the startpoint name, endpoint name, and other details about each path.
- The button bar below the table provides buttons that you can use to
  - Select and display one or more paths in a schematic view
  - Generate a histogram for the value distribution of a specific attribute
  - Select and display an individual path in a path inspector window
  - Select one or more paths and generate a timing report (available only in the timing analysis driver)

  You can configure the table by hiding or displaying individual table columns, and you can save the path details in a text file. in the timing analysis driver, you can also reload the table with a different collection of timing paths.

The timing analysis driver contains a command display box that shows the command and options used to find the paths. You can copy text in this box and paste it somewhere else, such as on the console command line or in a text file.

You can select paths in the timing path table and view or highlight in a schematic or layout view. The selected paths appear in the selection color, which is white by default. If you want to view the cells connected to the selected paths, choose Select > Cells > Of Selected Paths. You can cross-probe selected paths by choosing AnalyzeRTL > Cross Probe to Source.

When you open the timing analysis driver, or reload the paths if it is already open, you use the Select Paths dialog box to load a collection of paths into the timing path table. You can

- Set options in the dialog box and run the `get_timing_paths` command
- Select and run a predefined collection command

  You can load all selected paths or all highlighted paths.
- Enter a command to define and load a custom collection

  The tool adds the command to the list of predefined commands. If you define a custom collection in a variable, you can use the `get` command to load the paths from the collection.

The dialog box options are set by default to select the 20 timing paths with the worst slack times in the design. You can reset the dialog box options to their default values by clicking the Default button.

When you open a path data table, it displays information about the paths that you selected in the path analyzer.

### See Also

- The “Analyzing Timing Path Details” topic in Design Vision Help
- The “Viewing Path Data” topic in Design Vision Help

## Inspecting Timing Path Elements

The path inspector provides tools you can use to examine various aspects of a timing path. When you select a path and open the path inspector, you can view path delay profiles and timing report information about the path.

The path delay profiles display the relative contributions of various clock launch and capture components to the delay and slack calculations of the timing path. You can hold the pointer over a path profile component to view its contribution to the total path delay.

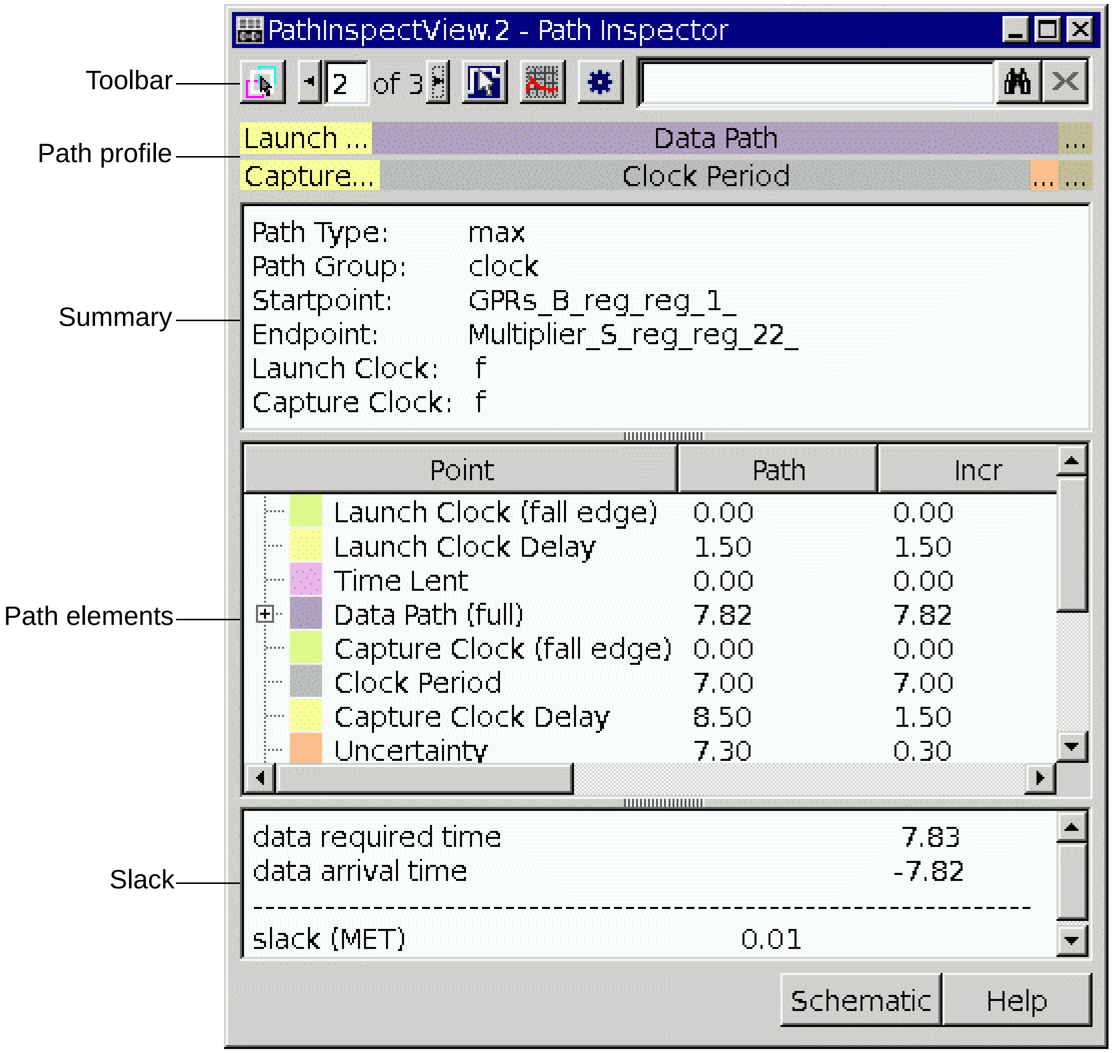
The timing report information appears in three parts: a path summary, a path elements table, and path slack details.

- The path summary provides information about the path attributes that identify the path, such as its path group, startpoint and endpoint, and delay type.
- The path elements table displays information about the elements of the path and their contributions to the path delay and slack calculation.
- The path slack details include information about the required and actual arrival times and the slack values. You can also copy the summary and paste them into text files, export the tables to text files, customize the element tables, and generate delay calculation reports for selected cells or nets.

You can load multiple paths and display the information for each path sequentially by changing from one path to another. You can also select and load different paths at any time when the path inspector is open.

Figure 3-6 shows an example of a path inspector window.

*Figure 3-6 Path Inspector Window*



```
Path Type:       max
Path Group:      clock
Startpoint:      GPRs_B_reg_reg_1_
Endpoint:        Multiplier_S_reg_reg_22_
Launch Clock:    f
Capture Clock:   f
```

| Point | Path | Incr |
|---|---|---|
| Launch Clock (fall edge) | 0.00 | 0.00 |
| Launch Clock Delay | 1.50 | 1.50 |
| Time Lent | 0.00 | 0.00 |
| Data Path (full) | 7.82 | 7.82 |
| Capture Clock (fall edge) | 0.00 | 0.00 |
| Clock Period | 7.00 | 7.00 |
| Capture Clock Delay | 8.50 | 1.50 |
| Uncertainty | 7.30 | 0.30 |

```
data required time                                     7.83
data arrival time                                     -7.82
-------------------------------------------------------------
slack (MET)                             0.01
```

You can search for text in the timing report sections. You can also select and copy text from the summary and slack sections.

By default, the path elements pane displays a table of timing path elements, such as datapath, clock period, and clock uncertainty. You can expand some elements, such as the datapath, by clicking the expansion button (plus sign) beside the element name. Use the path elements table to view

- A clock, pin, or net design object associated with a path element
- The launch clock path

- The capture clock path
- The timing path

You can view information about a path element by holding the pointer over its row in the path elements table. You can also configure how the path elements table displays the data.

**See Also**

- The “Using the Path Inspector” topic in Design Vision Help

## Viewing the Floorplan in Design Compiler Graphical

The Design Vision layout view lets you analyze physical constraints, timing, and congestion in your floorplan. A layout view displays floorplan constraints, critical timing paths, and congested areas in a single, flat view of the physical design.

When you start the tool in topographical mode, you use the layout view to visually examine block and hierarchical cell placement, floorplan constraints, critical timing paths, and global routing congestion, and gather information that can help you to guide later optimization operations in the Design Compiler tool and other Synopsys tools.

You can view the floorplan before or after you optimize the design. To examine the floorplan in a layout view, you can

- Change your view of the design in a layout view, or change between layout views, by using the Overview panel
- Select, highlight, and query objects and magnify or shrink the view by using the interactive mouse tools
- Measure distances by drawing rulers
- View cell-to-cell or pin-to-pin connectivity by displaying flylines
- Visually analyze floorplan-related congestion by viewing the congestion map
- Visually analyze block and cell placement by coloring objects with visual modes

You can display or hide objects, control object selection, and customize object appearance by changing options on the View Settings panel. Objects you select or highlight in other views, such as a schematic view, are automatically cross-selected or cross-highlighted in the layout view. This capability allows you to efficiently examine both the layout and timing aspects of your floorplan.

**See Also**

- Solving Floorplan and Congestion Problems
- The “Viewing the Floorplan” and “Analyzing the Floorplan” topics in Design Vision Help

# Using DFT Analysis Tools

If you are using the DFT Compiler tool, you can access tools from the Test menu to DRC violations and hold time violations. For information about the test analysis views you can use, see the following sections:

- Checking Scan Test Design Rules
- DRC Violation Browser
- Violation Inspector
- DFT Hold Time Analysis Window

**See Also**

- The *DFT Compiler User Guide*

## Checking Scan Test Design Rules

You should check the current design for DRC violations, fix any problems in the RTL, and resynthesize the design before performing other DFT Compiler operations.

To check the current design for DRC violations,

- Choose Test > Run DFT DRC.

The DFT Compiler tool checks the design for DRC violations and displays messages in the console log view. You can click the underlined message numbers in the console log view to display man pages for the messages in the man page viewer.

If violations exist, the GUI automatically opens a new Design Vision window and displays the violation messages in the violation browser view window.

The DFT DRC command requires a valid test protocol. You can generate a test protocol by using the `create_test_protocol` command or view an existing test protocol by using the `read_test_protocol` command. For details about these commands, see their man pages.

**See Also**

- The “Checking Scan Test Design Rules” topic in Design Vision Help

## DRC Violation Browser

If you are using the DFT Compiler tool, you can use the DRC violation browser to search for and view information about DFT unified DRC violations in the current design. The violation browser can display both static and dynamic violation messages. Static violations occur as a result of the design topology. For dynamic violations, you can view waveforms for pins on the violated path.

The violation browser window consists of two panes. A violation category tree appears in the left pane. When you select a violation category, a list of violations appears in the right pane.

- The violation category tree groups warning and error messages into categories that help you find the problems you are interested in.
- The violation list displays the violation ID and pin name for each violation in the list.

You can select pin names and view information about the pins. You can also display man pages (in the man page viewer) for warning and error messages. If you want to visually inspect violations, you can display them in the violation inspector window.

You can filter the pin names in the violation list by specifying the names or name patterns for the pins you want to include or exclude. You can use the ? and * wildcard characters to create name patterns. Separate multiple names or name patterns with blank spaces.

The following section describes the violation inspector. For more information about the DRC violation browser, see the *DFT Compiler User Guide* and the “Examining DRC Violations” topic in Design Vision Help.

### See Also

- The “Examining DRC Violations” topic in Design Vision Help

## Violation Inspector

If you are using the DFT Compiler tool, you can analyze and debug DFT unified DRC violations by inspecting them in a violation inspector window. You can inspect one or more violations of the same type. The violation inspector provides both a violation schematic for inspecting static violations and a coordinated waveform view for inspecting dynamic violations.

The violation inspector displays the pin data that corresponds to the most suitable pin data type for debugging the violation. If you need to, you can change to a different pin data type.

- The pin data for static violations is constant; the simulation values do not change over time.

- The pin data for dynamic violations represents simulation values for a series of initialization cycles.

For more information about pin data types, see the TetraMAX documentation.

You can analyze and debug static violations by inspecting the design topology. Use the violation schematic to view and probe the signal and clock pins where the violations occur.

The violation schematic is an enhanced schematic. You can perform any schematic view operation in a violation schematic, including selecting objects, viewing object information, highlighting objects or timing paths, and magnifying and traversing the view.

To debug dynamic violations, you can select pins in the violation schematic and view their simulation values in the waveform view. Simulation values can be constant or they can vary over time in a series of simulation “events.” The violation inspector displays the pin data that corresponds to the most suitable pin data type for debugging the violation. To simulate pin data for a dynamic violation, you must use a pin data type that supports simulation values.

For more information about violation inspector, see the *DFT Compiler User Guide* and the “Inspecting Static DRC Violations” and “Inspecting Dynamic DRC Violations” topics in Design Vision Help.

### See Also

- The “Inspecting Static DRC Violations” and “Inspecting Dynamic DRC Violations” topics in Design Vision Help

## DFT Hold Time Analysis Window

If you are using the DFT Compiler tool, you can use the hold time analysis window to view information about scan cells that have hold time violations. You can also select scan cells in the window for further examination with other analysis tools.

The hold time analysis window contains a scan cell table that displays a list of scan cells with hold time violations. The table columns show the scan cell names and other details about each cell. A button bar below the table lets you load a different list of scan cells, save the scan cell details in a text file, customize the table columns, and access other analysis tools.

### See Also

- The “Analyzing Hold Time Violations” topic in Design Vision Help

## Defining and Viewing the Power Intent for Multivoltage Designs

The Design Vision tool supports IEEE 1801 power domains in multivoltage designs. IEEE 1801 is also known as Unified Power Format (UPF). In multivoltage design, the subdesign instances (blocks) operate at different voltages. In multisupply designs, the voltages of the various subdesigns are the same, but the blocks can be powered on and off independently. Except when stated otherwise, the term multivoltage as used here includes multisupply and mixed multisupply-multivoltage designs.

To reduce power consumption, multivoltage designs typically make use of power domains that are independently powered up and down, including domains that are defined to have always-on relationships relative to each other. By definition, a power domain is a logical grouping of one or more hierarchical blocks in the design that share the same power characteristics.

A power domain has the following characteristics:

- The domain name
- A scope, which is the hierarchy level in the logic design where the domain is defined
- The design elements that comprise the domain
- An associated set of supply nets that can be used within the domain
- The primary power supply and ground nets
- Synthesis strategies for isolation, level-shifters, always-on cells, retention registers, and secondary power supply and ground nets

When used together, the power domain and supply network objects allow you to specify the power management intentions of the design. For more details, see the *Power Compiler User Guide*.

In the Design Vision GUI, you can generate the UPF commands that create power domains and define their supply networks. By using the Visual UPF dialog box, you can

- Define the initial power architecture
- Edit an existing power architecture
- Review an existing power architecture

You can examine a UPF diagram that can help you determine whether the domains you have defined match your power intent for the design. If you have defined power state tables in your design, you can use the Power State Table panel to perform always-on analysis and multivoltage level shifter analysis.

For information about these subjects, see the following sections:

- Visually Defining the UPF Power Intent
- Viewing the UPF Power Design
- Visualizing Power State Tables

### See Also

- The *Power Compiler User Guide*
- The *Synopsys Multivoltage Flow User Guide*

## Visually Defining the UPF Power Intent

You can use the Visual UPF dialog box to create UPF power domains and define their supply networks, connections with other power domains, and relationships with elements in the design hierarchy. You can also review and edit an existing power design.

The Visual UPF dialog box contains the work environment for power domain generation. It displays the design hierarchy and provides tools you can use to define power domains for the top-level design and its subdesigns (hierarchical cells).

You can use the Visual UPF dialog box to

- Define the initial power design architecture

  You can create power domains, define their possible states, and specify their elements. Most of this work requires an overview of the design hierarchy but does not require details, such as those provided by the UPF format.

- Edit an existing power design architecture

  You can examine and modify an existing power design in which power domains and supply networks have already been defined. This allows you to define a power design incrementally or to make specific changes to resolve problems found in the initial design.

- Review an existing power design architecture

  You can review the existing power domain structures and modify them if necessary to meet the requirements of your UPF specifications.

You can use the Visual UPF dialog box any time before you compile the design. After compiling the design, you can open the dialog box to view information about the power domains and their properties, but you cannot make any changes.

The general use flow for generating UPF power domains includes these steps:

1. Define the power design architecture.

   Start by creating power domains and supply sets, and then define their properties. You can add or edit property values as needed, such as primary and secondary supply nets, power switches, and design elements for power domains. You can also create a power state table.

2. Review the generated UPF script.

   After defining the power design architecture, review the UPF commands that the tool generates to verify that the power domain definitions they implement meet your power intent. If the commands are not satisfactory, return to step 1.

3. Create the power objects in your design.

   You can either save the commands in a UPF file and run the `load_upf` command, by clicking OK in the Visual UPF dialog box, or save the commands in a Tcl script file by clicking the Save Script button.

You can examine information about the current power domains in your design and their power and ground supply networks by viewing a UPF diagram.

### See Also

- Viewing the UPF Power Design
- The “Generating and Running a UPF Script” topic in Design Vision Help
- The *Power Compiler User Guide*

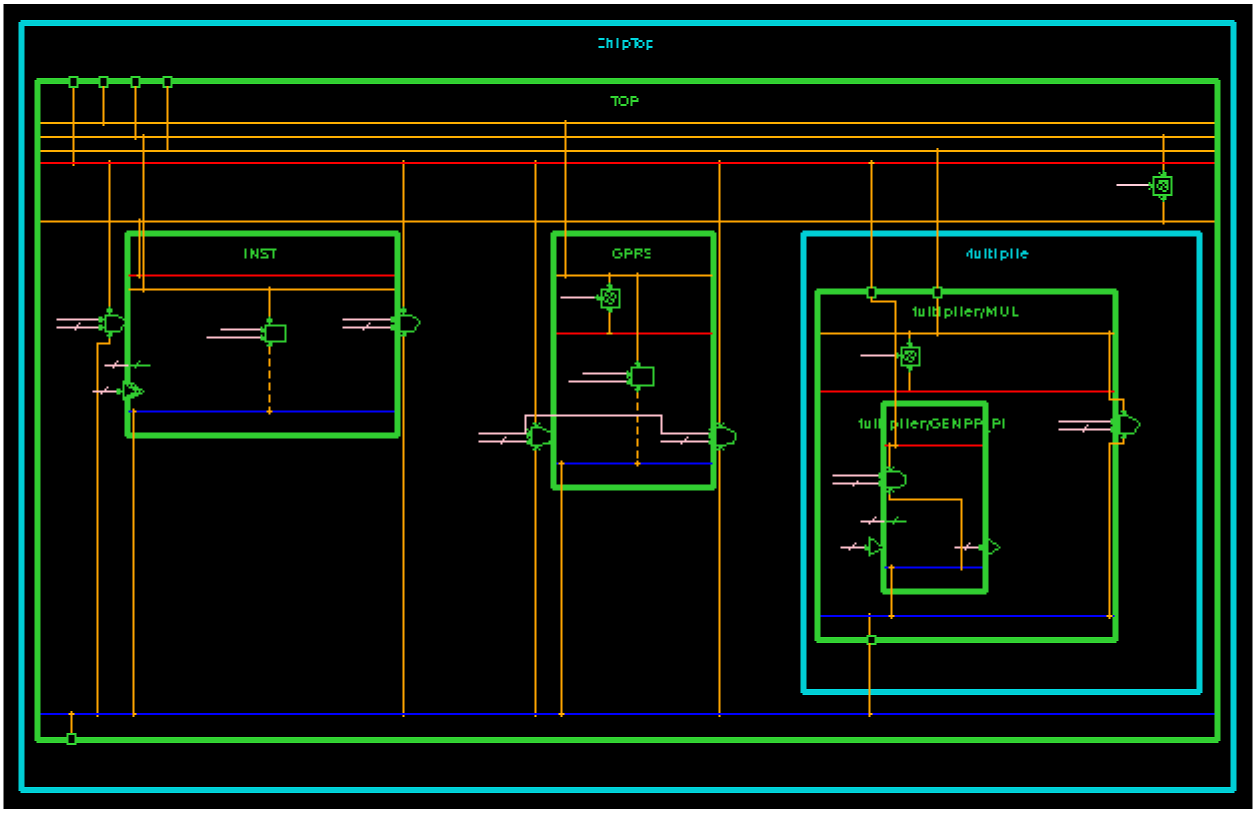
## Viewing the UPF Power Design

You can visually examine a graphic representation of the UPF power architecture in your design by using the UPF diagram view. The diagram displays graphic representations of UPF power domains and the supply network, switches, isolation, retention, and other power-management elements of your design. In this view, you can magnify and traverse the diagram, select objects, and view object information. You can also print the diagram and change the colors used for objects in the diagram.

The UPF diagram view displays the UPF power intent as it is defined in the design database. When you change the database, for example by entering a UPF command, the tool immediately updates the diagram. You can view the UPF diagram at any point in the design flow.

Figure 3-7 illustrates many of the aspects of a UPF diagram.

*Figure 3-7 UPF Diagram Example*



To facilitate your analysis, you can collapse or expand individual power domains or scopes. Initially, all the power domains and scopes are expanded. By displaying or hiding the contents of particular power domains or scopes, you can visually inspect just the power design data that you are interested in viewing while ignoring unrelated data.

Initially, the full diagram is visible in the view. You can magnify and traverse the view by using the zoom and pan tools and commands. You can also use the arrow keys to scroll vertically or horizontally through the view. To examine the UPF diagram, you can

- Select individual objects or objects in a rectangular area by using the Selection tool

  Note that object selection is local in the UPF diagram view and does not change the global selection list in the tool.

- Preview object information in an InfoTip by holding the pointer over the object
- Display object information on the Query panel by using the Query tool

The View Settings panel provides options you can use to customize the appearance of the UPF diagram. You can adjust the brightness, change the colors for individual object types, change the background color, or apply a predefined or user-defined color theme.

You can print the UPF diagram displayed in the UPF diagram view or the Visual UPF dialog box. Make sure that a default printer is set in your .cshrc file. You can also save an image of the UPF diagram in a file for printing later from a UNIX or Linux shell.

**See Also**

- The “Viewing the UPF Power Design” topic in Design Vision Help
- The *Power Compiler User Guide*

## Visualizing Power State Tables

You can use the Power State Table panel with the UPF diagram view to analyze and debug your isolation and level-shifter strategies in a UPF multivoltage design. You can view the states for each supply in a power state table and visually examine their relationships in the UPF diagram view.

You can perform the following types of analysis:

- Always-on analysis compares the on-off states between power and ground supplies
- Multivoltage level-shifter analysis compares the voltage relationships between power supplies

Always-on analysis compares the on-off states between supplies, including both power and ground supplies. This analysis produces one of the following states: More AO, Less AO, Equally AO, and Unrelated AO.

Multivoltage level-shifter analysis compares the voltage relationships between supplies. This analysis produces one of the following states: LH, HL, HL_LH, or None.

**See Also**

- The “Visualizing Power State Tables” topic in Design Vision Help

# Analyzing Multivoltage Design Problems

The Design Vision GUI provides tools that can help you to analyze and debug multivoltage designs. In the GUI, you can

- Analyze multivoltage design problems by checking the design for errors and viewing the violation report in the MV Advisor violation browser
- Analyze path-based multivoltage design connections by generating and examining a report of level-shifter drive and load pins or a report of always-on nets

For information about these subjects, see the following sections:

- Examining and Debugging Multivoltage Design Violations
- Analyzing Multivoltage Design Connections

### See Also

- The *Power Compiler User Guide*
- The *Synopsys Multivoltage Flow User Guide*

## Examining and Debugging Multivoltage Design Violations

The MV Advisor violation browser provides a visual analysis and debugging environment for multivoltage design violations. You can check the design for problems such as multivoltage constraint violations, electrical isolation violations, connection rule violations, and operating condition mismatches. After checking a design, you can use the violation browser to examine the violation report.

The violation browser groups violations based on specific properties, displays detailed information about the violations, and guidance for investigating and fixing them. When you select a violation, the violation browser displays details such as an explanation of the warning or error message and suggestions for fixing the violation.

The violation browser also provides access to context-aware reports and other analysis tools. You can

- Select pin names and view information about the pins
- Display man pages (in the man page viewer) for warning and error messages
- Visually inspect a violation by displaying it in a schematic view

You can also display the report for an individual violation in a new Design Vision window that serves as a debugging work environment.

You can check the design for violations before or after you open the violation browser. To check the design before opening the violation browser, use the `check_mv_design` command. When the violation browser is open, you can use the Check MV Design dialog box to check the design.

### See Also

- The “Examining Multivoltage Design Violations” topic in Design Vision Help
- The *Power Compiler User Guide*

## Analyzing Multivoltage Design Connections

You can analyze path-based multivoltage design connections by generating a report of level-shifter drive and load pins and a report of always-on nets. These reports allow you to gather design details that can help you to understand multivoltage-related design problems. They contain details about the variable settings for level-shifter insertion and always-on buffering, relevant power state tables, the driver-to-load pin connections, the pin-to-pin information for specified paths, the target libraries used for insertion of power management cells, and other useful debugging information.

Each report appears in a new analysis view. You can select and copy text in the view that you want to paste into another application, such as a text editor. You can click a hyperlink in the analysis view to generate a schematic containing the objects in the report. By using the schematic view, you can

- View power information for pins and cells
- Create collections of the power and ground supply nets connected to selected pins
- Generate reports of power pin information for selected cells

For more details about multivoltage analysis reports, see the *Power Compiler User Guide* and the "Analyzing Path-Based Design Details" topic in Design Vision Help.

### See Also

- The "Analyzing Path-Based Design Details" topic in Design Vision Help
- The *Power Compiler User Guide*

# 4

# Performing Basic Tasks

The Design Vision GUI is a menu-driven interface. if you are an experienced user of Synopsys synthesis tools, you can accomplish familiar pre-synthesis and synthesis tasks by using Design Vision windows and menus. If you are new to the Synopsys synthesis tools, see the *Design Compiler User Guide* or the *DC Explorer User Guide* to learn the standard tasks for working in the synthesis environment and running a synthesis flow.

For information about performing these tasks, see the following sections:

- Specifying Logic Libraries
- Using a Milkyway Database
- Working With Designs in Memory
- Defining the Design Environment
- Setting Design Constraints
- Compiling the Design
- Working With Reports
- Selecting Objects by Name
- Printing Schematic Views
- Changing the Appearance of Schematics
- Saving an Image of a Window or View

## Specifying Logic Libraries

Before you start work on a design, specify the location of your libraries. You can define your library locations directly in the .synopsys_dc.setup file or indirectly by entering the locations in the Application Setup dialog box. Either method is acceptable—they both accomplish the same thing. (You can also specify library locations by running a script when you start the tool or by using the Execute Script dialog box. For details, see "Starting the Tool" on page 2-5 and "Using Script Files" on page 2-10.)

The link and target libraries are logic libraries that define the semiconductor vendor's set of cells and related information, such as cell names, cell pin names, delay arcs, pin loading, design rules, and operating conditions. The symbol library defines the symbols for schematic viewing of the design.

Synopsys provides a standard DesignWare library with components that implement many of the built-in HDL operators. You do not need to specify the standard DesignWare library; however, you must specify any additional, specially licensed, DesignWare libraries with the `synthetic_library` variable (you do not need to specify the standard DesignWare library).

To specify the library location in the Application Setup dialog box,

1. Choose File > Setup.

   The Application Setup dialog box opens.

2. Select the Defaults category if the Defaults page is not displayed.
3. Enter the appropriate path in the Search Path box.
4. Enter the library file names for the link, target, and symbol libraries you need to use.
5. (Optional) Enter the library file names for any specially licensed Synopsys or third-party DesignWare libraries you need to use.
6. Click OK.

For information about defining libraries in the .synopsys_dc.setup file, see the *Design Compiler User Guide*. For information about the options on the Variables page in the Application Setup dialog box, see the "Setting Variables" topic in Design Vision Help.

### See Also

- The "Setting Library Locations" topic in Design Vision Help

  Provides information about the options on the Defaults page

- The *Design Compiler User Guide*

  Provides information about the function of link libraries, target libraries, symbol libraries, and DesignWare libraries

## Using a Milkyway Database

Design Compiler topographical technology provides the capability to accurately predict post-layout timing, area, and power during RTL synthesis without the need for wire load model-based timing approximations. It uses Synopsys' placement and optimization technologies to drive accurate timing prediction within synthesis, ensuring better correlation to the final physical design. This new technology is a part of the DC Ultra feature set and is available only by using the `compile_ultra` command in topographical mode.

To use the Design Compiler topographical features in the Design Vision tool, you must run the tool in topographical mode. In this mode, the command-line prompt appears as design_vision-topo> on the console and in the shell. For more information about starting the tool, see "Starting the Tool" on page 2-5.

Topographical technology leverages the Synopsys physical implementation solution to derive the "virtual layout" of the design so that the tool can accurately predict and use real net capacitances instead of wire load model-based statistical net approximations. If wire load models are present, they are ignored. In addition, the tool updates capacitances as synthesis progresses by adjusting placement-derived net delays based on an updated "virtual layout" at multiple points during synthesis.

This approach eliminates the need for over constraining the design or using optimistic wire load models in synthesis. The accurate prediction of net capacitances drives the tool to generate a netlist that is optimized for all design goals including area, timing, test, and power. It also results in a better starting point for physical implementation.

Topographical technology supports all synthesis flows, including

- Test-ready compile flow (basic scan and DFT MAX adaptive scan)
- Clock-gating flow
- Register retiming

When you use the Compile Ultra command in topographical mode, the Design Compiler topographical features are automatically used. All `compile_ultra` command options are supported. In addition, the Compile Design command (Design menu), the Report Wire Load command (Timing menu), and the Wire Load command (Attributes > Operating Environment menu) are not available in topographical mode.

For more information about using Design Compiler topographical technology, see the *Design Compiler User Guide*.

For information about working with Milkyway design libraries in topographical mode, see the following sections:

- Creating a Milkyway Design Library
- Opening or Closing a Milkyway Design Library
- Setting the TLUPlus Extraction Files

## Creating a Milkyway Design Library

You need a Milkyway design library to save a design in Milkyway format for use in other Synopsys Galaxy platform tools, such as the IC Compiler tool. You can use a single Milkyway design library across the entire Galaxy flow.

The Milkyway tool stores design data in the Milkyway design library and physical library data in the Milkyway reference library. Before creating a Milkyway design library, you must prepare the design and reference libraries.

- The Milkyway directory structure used to store design data (the uniquified, mapped netlist and constraints) is referred to as the Milkyway design library. You can specify a design library for the current session by setting the `mw_design_library` variable to the root directory path.
- The Milkyway directory structure used to store physical library data is referred to as the Milkyway reference library. Reference libraries contain standard cells, macro cells, and pad cells. the Design Vision tool uses the FRAM view of the reference libraries as the default physical model for your design. You can specify a reference library for the current session by setting the `mw_reference_library` variable to the root directory path.

When you create a Milkyway design library, the Design Vision tool sets the reference libraries for the design. For more information about using Milkyway design libraries in topographical mode, see the *Design Compiler User Guide*.

To create a Milkyway design library,

1. Choose File > Create MW Library.

   The Create Library dialog box opens.

2. Specify the following design library information:

   ❍ The path to the library root directory

   ❍ The library name

   ❍ The technology and physical library file names

3. Select a reference library option and specify the reference library information.
4. Set other options as needed.
5. Click OK.

You can specify the Milkyway reference library files directly or by using a reference control file. For information about using these options, see the “Creating a Milkyway Design Library” topic in Design Vision Help and the *Design Compiler User Guide*.

Using the Create Library dialog box is equivalent to using the `create_mw_lib` command.

**See Also**

- Using a Milkyway Database

## Opening or Closing a Milkyway Design Library

After you create a Milkyway design library, you must open it before you can read in the design. You cannot have more than one library open at the same time. If another library is already open, you must close it before opening a different library.

To open a Milkyway design library,

1. Choose File > Open MW Library.

   The Open Library dialog box opens.
2. Specify the library name and select a permission option.
3. Click OK.

Using the Open Library dialog box is equivalent to using the `open_mw_lib` command.

To close a Milkyway design library,

- Choose File > Close MW Library.

Using the Close MW Library command is equivalent to using the `close_mw_lib` command.

**See Also**

- Using a Milkyway Database

## Setting the TLUPlus Extraction Files

TLUPlus is a binary table format in the Milkyway library for RC coefficients. Although you do not need to specify TLUPlus files if resistance and capacitance models are present in your vendor technology physical library, these files provide more accurate capacitance and resistance data, thereby improving correlation with back-end results.

To use TLUPlus, you must specify the maximum TLUPlus model files. You can also specify minimum TLUPlus model files and a map file that maps layer names between the Milkyway technology file and the process Interconnect Technology Format (ITF) file.

To select the TLUPlus files,

1. Choose File > Set TLU+.

   The Set TLU+ dialog box opens.
2. Specify the TLUPlus file names.
3. Click OK.

Using the Set TLU+ dialog box is equivalent to using the `set_tlu_plus_files` command. The Design Vision tool stores the names and locations of the TLUPlus files in Milkyway, but it does not store the TLUPlus information found in these files. For each Design Vision session, you must specify the TLUPlus files that you need to use during the session.

### See Also

- Using a Milkyway Database

## Working With Designs in Memory

The Design Vision tool reads designs into memory from design files. Many designs can be in memory at any time. After reading in a design, you can change it in numerous ways, such as grouping or ungrouping its subdesigns or changing subdesign references.

To learn how to work with designs in memory, see the following topics:

- Reading Designs
- Viewing the List of Designs in Memory
- Setting the Current Design
- Linking Designs

- Removing Designs From Memory
- Saving Designs

## Reading Designs

To begin working on your design, read the design from disk into the tool's active memory. This is where all changes in the design take place before you save the design by writing it back to disk.

The File menu contains the commands for reading in a design:

- Analyze and Elaborate

  Use Analyze and Elaborate to read HDL designs and convert them to .ddc format. These commands open dialog boxes in which you can set options that are equivalent to the `analyze` and `elaborate` command-line options. For information, see the "Analyzing Files" and "Elaborating a Design" topics in Design Vision Help.

- Read

  Use Read (`read_file` is the command-line equivalent) to read designs that are already in .ddc format. This command opens a dialog box in which you can set options that are equivalent to the `read_file` command-line options. For more information, see the "Reading in a Design" topic in Design Vision Help.

The Analyze command checks the HDL designs for proper syntax and synthesizable logic, translates the design files into an intermediate format, and stores the intermediate files in the directory you specify. The Elaborate command first checks the intermediate format files before building a .ddc design. During this process, Elaborate determines whether it has the necessary synthetic operators to replace the HDL operators, and it also determines correct bus size.

If you use the Read command to read in HDL files, the Analyze and Elaborate read functions are combined. However, Read does not perform certain design checks that Analyze and Elaborate perform.

The GUI can access all of the files supported by the Design Compiler tool. Table 4-1 shows the supported design file input formats. All netlist formats except .db, equation, PLA, state table, Verilog, and VHDL require special license keys.

*Table 4-1 Supported Design File Input Formats*

| Format | Description |
|---|---|
| .ddc | Synopsys internal database format (recommended) |
| .db | Synopsys internal database format |

*Table 4-1 Supported Design File Input Formats (Continued)*

| Format | Description |
|---|---|
| Verilog | IEEE Standard Verilog (see the HDL Compiler documentation) |
| VHDL | IEEE Standard VHDL (see the HDL Compiler documentation) |
| SystemVerilog | IEEE Standard SystemVerilog (see the HDL Compiler documentation) |
| equation | Synopsis equation format |
| pla | Berkeley (Espresso) PLA format |

For information about using the Analyze and Elaborate commands, see the “Analyzing Files” and “Elaborating a Design” topics in Design Vision Help. For information about using the Read command, see the “Reading in a Design” topic in Design Vision Help.

**See Also**

- Viewing the List of Designs in Memory
- Setting the Current Design
- Linking Designs
- Removing Designs From Memory
- Saving Designs

## Viewing the List of Designs in Memory

You can generate a list of the designs loaded in memory and display information about them in a list view. The list includes the name, the area, the design directory path, the DesignWare implementation, and the states of the structure, flatten, and dont_touch attributes for each the design.

To open the design list view,

- Choose List > Designs View.

You can select some or all of the designs in the list by clicking or dragging the pointer across their names in the list view. You can also use Shift-click or Control-click to select multiple designs. In addition, you can

- Sort the information alphabetically by clicking a column heading. Click the heading again to reverse the sort

- Scroll up and down in the table by pressing the Up Arrow and Down Arrow keys
- Filter the objects listed in the table, limiting it to objects based on a character string or regular expression that you define

If some of the text in a column is missing because the column is too narrow, you can hold the pointer over it to display the information in an InfoTip. You can also adjust the width of a column by dragging its right edge left or right.

For more information about the design list, see the “Viewing a List of Designs” topic in Design Vision Help.

You can filter the design list, limiting it to designs based on a character string or regular expression that you define, by using the Filter List dialog box. For more information, see the “Filtering Object Lists” topic in Design Vision Help.

### See Also

- Reading Designs
- Setting the Current Design
- Linking Designs
- Removing Designs From Memory

## Setting the Current Design

When you start a Design Vision session and read a design, the current design is automatically set to the top-level design. Some commands require you to set the current design to a subdesign before you issue them (the man pages provide such information).

To set the current design,

1. Click the drop-down list on the Design List toolbar to display the design names.
2. Select a design name.

Alternatively, you can open a design list view (by choosing List > Designs View), select a design name in the list, right-click, and choose Set Current Design. The command-line equivalent is `set current_design`.

### See Also

- Reading Designs
- Viewing the List of Designs in Memory

- Linking Designs
- Removing Designs From Memory

## Linking Designs

For a design to be complete, it must be connected to all of the designs and library components that it references. For a subdesign to be complete, there must be a reference that links the subdesign or component to the link libraries. This process is called linking the design or resolving references.

When you link a design, the tool locates and connects all of the designs and library components referenced in the current design and connects them to the current design.

Note:

If you read in a linked design from a .ddc file, you do not need to relink the design.

Designs can be linked either automatically or manually.

Automatic linking occurs when you optimize the design, open a schematic view of the design, group cells into subdesigns, check design consistency, or generate reports. During automatic linking, the tool links only unlinked components and subdesigns and does not remove existing links.

When you manually link a design, the tool removes existing links before starting the link process.

To manually link the current design,

1. Choose File > Link Design.

   The Link Design dialog box appears.

2. Specify the design file search paths.

   You can either enter the paths in the “Search path” box (use blank spaces to separate individual paths) or click the Browse button and select directories in the file browser.

3. Enter the library paths and file names in the “Link library” box.
4. If you want the compiler to search for referenced subdesigns in memory before searching files in the specified search path directories, select the “Search memory first” option.
5. Click OK.

The GUI displays the names of the linked designs and libraries in the console log view.

### See Also

- Reading Designs
- Viewing the List of Designs in Memory
- Setting the Current Design
- Removing Designs From Memory
- Saving Designs

## Removing Designs From Memory

You can remove designs from memory without exiting the tool. You can either remove all the designs or one or more individual designs.

To remove all designs from memory,

- Choose File > Remove All Designs.

To remove an individual design,

1. Select the design name in the design list view.
2. Click right and choose Remove Design.

For information about opening a design list view, see "Viewing the List of Designs in Memory" on page 4-8.

After removing designs, you can load different designs or reload the same designs by using either the Read command or the Analyze and Elaborate commands on the File menu. For information about these commands, see "Reading Designs" on page 4-7.

### See Also

- Reading Designs
- Setting the Current Design
- Linking Designs
- Saving Designs

## Saving Designs

You can save (write to disk) the designs and subdesigns of the design hierarchy at any time, using different names or formats. After modifying a design, you should save it manually. the Design Vision tool does not automatically save designs before it exits.

For information about saving designs, see the following topics:

- Supported Design Output Formats
- Writing a Design Netlist
- Writing to a Milkyway Database

### See Also

- Reading Designs
- Viewing the List of Designs in Memory
- Removing Designs From Memory

## Supported Design Output Formats

The tool stores design data in an internal database format. It supports two design database formats: .ddc and Milkyway.

- .ddc format

  The .ddc format is a single-file, binary format. The .ddc format stores design data in an efficient manner than the .db format, enabling increased capacity. In addition, reading and writing files in .ddc format is faster than reading and writing files in .db format. The .ddc format stores only logical design information.

- Milkyway format

  The Milkyway format allows you to write a Milkyway database for use with other Synopsys Galaxy tools, such as the IC Compiler tool. The Milkyway format stores both logical and physical design information, but it requires a mapped design.

The Milkyway format is available only when you start the Design Vision tool in topographical mode. Use the `write_milkyway` command to save netlist and physical design data in a Milkyway design library. You can use a single Milkyway library across the entire Galaxy flow. For more information, see the *Design Compiler User Guide*.

Note:
The Design Vision tool does not support the `read_milkyway` command.

The GUI can access all of the design file formats supported by the Design Compiler tool. Table 4-2 shows the supported design file output formats.

*Table 4-2 Supported Design File Output Formats*

| Format | Description |
|---|---|
| .ddc | Synopsys internal database format |
| Verilog | IEEE Standard Verilog (see the HDL Compiler documentation) |
| svsim | SystemVerilog netlist wrapper<br>Note: Specifying this format causes the tool to write out only the netlist wrapper, not the gate-level design under test (DUT). To write out the gate-level DUT, use the `write_file -format verilog` command. For more information, see the *HDL Compiler for SystemVerilog User Guide*. |
| VHDL | IEEE Standard VHDL (see the HDL Compiler documentation) |
| Milkyway | Format for writing a Milkyway database |

**See Also**

- Writing a Design Netlist
- Writing to a Milkyway Database

## Writing a Design Netlist

The Design Vision tool does not automatically save designs before exiting. When you make changes to a design, save it in a file before you exit the tool.

To save the current design and its subdesigns,

- Click the 💾 button on the File toolbar or choose File > Save.

By default, the files are saved in .ddc format files named *design_name*.ddc, where *design_name* is the name of the design. You can save a design in any supported design file output format.

To save the current design and all of its subdesigns in a single file or with a different file name or file format,

1. Choose File > Save As.

   The Save Design As dialog box appears.

2. (Optional) Select a file format in the Format list.

   By default, the tool automatically selects the format based on the file name suffix. If the file name you specify does not contain a suffix or the suffix does not indicate the appropriate format, you must select a format option.

3. Navigate to the directory where you want to save the design, or enter the path name in the “File name” box.

4. Select a file name, or enter a file name in the “File name” box.

   If you select or enter the name of an existing file, the tool overwrites the file.

5. (Optional) If you want to save all the designs in the hierarchy instead of just the current design, make sure the “Save all designs in hierarchy” option is selected.

6. Click OK.

### See Also

- Supported Design Output Formats
- Writing to a Milkyway Database
- The “Saving the Design” topic in Design Vision Help

## Writing to a Milkyway Database

The Milkyway format allows you to write a Milkyway database for use with other Synopsys Galaxy tools, such as the IC Compiler tool. The Milkyway format stores both logic design and physical design information, but it requires a mapped design.

The Milkyway format is available only when you start the tool in topographical mode. Use the `write_milkyway` command to save netlist and physical design data in a Milkyway design library. You can use a single Milkyway library across the entire Galaxy flow. For more information, see the *Design Compiler User Guide*.

Note:
The tool does not support the `read_milkyway` command.

### See Also

- Supported Design Output Formats
- Writing a Design Netlist

## Defining the Design Environment

The Design Compiler tool requires that you model the environment of the design to be synthesized. This model comprises the external operating conditions (manufacturing process, temperature, and voltage), loads, drive characteristics, fanout loads, and wire loads. It directly influences design synthesis and optimization results.

Defining the design environment can involve using a large number of commands. Many designers find it convenient to define the design environment by using the default target library settings and by running scripts to define differences or additions. To define the design environment by using Design Vision menus, choose commands on the Operating Environment submenu under the Attributes menu. Design Vision Help has more information about particular commands and submenus under the Attributes menu.

### See Also

- The “Defining the Design Environment” topic in Design Vision Help
- The *Design Compiler User Guide*

## Setting Design Constraints

Setting design constraints can involve using a large number of commands. Most designers find it convenient to use scripts to set design constraints.

The Design Compiler tool uses design rule and optimization constraints to control the synthesis of the design. For information about these constraints, see the following topics:

- Setting Design Rule Constraints
- Setting Optimization Constraints
- Reporting Constraint Violations

### See Also

- The “Setting Constraints” topic in Design Vision Help
- The *Design Compiler User Guide*

## Setting Design Rule Constraints

Design rules are provided in the vendor logic library to ensure that the product meets specifications and works as intended. Typical design rules constrain transition times, fanout loads, and capacitances. These rules specify technology requirements that you cannot violate. (You can, however, specify stricter constraints.)

To set design rule constraints for the current design by using the GUI,

- Choose Attributes > Optimization Constraints > Design Constraints, set options as needed, and click OK.

To set design rule constraints for certain input ports by using the GUI,

- Select the ports, choose Attributes > Optimization Directives > Input Port, set options as needed, and click OK.

### See Also

- The “Setting Design Rule Constraints” topic in Design Vision Help
- The *Design Compiler User Guide*

## Setting Optimization Constraints

Optimization constraints define the design goals for timing (clocks, clock skews, input delays, and output delays) and area (maximum area). During optimization, the tool attempts to meet these goals; however, it does not violate your design rules. To optimize a design correctly, you must set realistic optimization constraints.

To set optimization constraints by using the GUI,

1. Click Attributes in the menu bar to open the Attributes menu.
2. Choose the command for the constraints you want to set.

   Choose Specify Clock if you want to set clock periods and waveforms. Other optimization constraints and settings are in the submenus under the Attributes menu:

   ❍ Operating Environment (input and output delays)

   ❍ Optimization Constraints (maximum and minimum delays and maximum area)

   ❍ Optimization Directives (design attributes, object attributes, and timing exceptions)

   Explore these submenus to find the settings you need. For more information about menu items in the Attributes menu, see the “Attributes Menu” topic in Design Vision Help.

**See Also**

- The “Setting Optimization Constraints” topic in Design Vision Help
- The *Design Compiler User Guide*

## Reporting Constraint Violations

You can generate a report that provides information about design rule and optimization constraint violations in the current design. For each constraint, the report includes information about

- Whether the constraint was violated or met, and by how much
- Which design object was the worst violator

By default, the report includes information about all design rule and optimization constraints in the design. You can limit the report to one or more types of constraints by selecting the options for those constraint types.

If you are using the Power Compiler tool, you can also limit the information about power constraints to dynamic power, leakage, or both.

Note:

> Queries for power constraint information require a Power-Optimization license and supporting libraries characterized for power. For details, see the *Power Compiler User Guide*.

To generate a report about constraint violations,

1. Choose Design > Report Constraints.

   The Report Constraints dialog box appears. The name of the current design appears in the “Current design” box.

2. Set options as needed to limit the report to information about certain types of design rule and optimization constraints.

   You can select one or more options for the types of constraint information you want to include.

3. (Optional) To include constraint information for specific scenarios in a multi-scenario design, enter the scenario names in the Scenarios box.

   The report lists the constraint information separately for each scenario and does not include constraint information for inactive scenarios.

   By default, the report includes constraint information for all active scenarios in the design. However, if you select the Verbose option or the “Show all violators” option and do not specify any scenarios, the report includes constraint information for the current scenario only.

4. Set other report content options as needed.
5. Set output options as needed.

   By default, the GUI displays the report in the report view and does not save it in a file.
6. Click OK.

For more information about the constraint report and the report options, see the man page for the `report_constraint` command.

### See Also

- The “Reporting Constraint Violations” topic in Design Vision Help
- The *Design Compiler User Guide*

## Compiling the Design

You can use the Design Vision GUI to initiate Design Compiler synthesis and optimization, thus compiling your high-level design description to your target technology. The Design Vision tool supports standard synthesis methodology: either a top-down compile or a bottom-up compile.

For information about compiling the design in the Design Vision GUI, see the following topics:

- Using the Compile Command
- Using the Compile Ultra Command

For information about compile methodologies, see the *Design Compiler User Guide*.

### Using the Compile Command

To compile the current design,

1. Choose Design > Compile Design.

   The Compile Design dialog box opens.
2. Select or deselect options as you require.
3. Click OK to begin compiling.

For information about using the Compile Design dialog box, see the “Optimizing the Design” topic in Design Vision Help. Use the default settings for your first-pass compile. For most designs, the default settings provide good initial results. For more information about `compile` command options, see the man page and the *Design Compiler User Guide*.

After compiling the design, save the design as described in “Saving Designs” on page 4-12.

**See Also**

- Using the Compile Ultra Command

## Using the Compile Ultra Command

For high-performance designs that have significantly tight timing constraints, you can use the Design Vision GUI to initiate the DC Ultra solution for better quality of results (QoR). The Compile Ultra command is a push-button solution that allows you to apply the best possible set of timing-centric variables or commands during compile for critical delay optimization as well as improvement in area QoR.

Note:

> Because Compile Ultra includes all compile options and starts the entire compile process, no separate Compile command is necessary.

To use the Compile Ultra command, you need a DC Ultra license and a DesignWare Foundation license.

To compile the current design by using Compile Ultra,

1. Choose Design > Compile Ultra.

   The Compile Ultra dialog box opens.

2. Select or deselect options as you require.

3. Click OK to begin compiling.

For information about using the Compile Ultra dialog box, see the “Optimizing Critical Delays” topic in Design Vision Help. Select options according to the requirements of your design. To perform a second-pass incremental compile, select the Incremental option. For more information about `compile_ultra` command options, see the man page and the *Design Compiler User Guide*.

When you run the Design Vision tool in topographical mode, the Compile Ultra command automatically uses the Design Compiler topographical features. All Compile Ultra command options are supported in this mode. Note that using the Incremental option with a topographical netlist results in placement-based optimization only. This compile should not be thought of as an incremental mapping. For more information about running the tool in topographical mode, see “Using a Milkyway Database” on page 4-3.

After compiling the design, save the design as described in “Saving Designs” on page 4-12.

### See Also

- Using the Compile Command

## Working With Reports

Textual reports are available from the Design menu and the Timing menu. Use Design menu commands to generate design information and design object reports. Use Timing menu commands to generate timing and constraint reports. If these menus do not have the report you need, you can generate any Design Compiler report by issuing dc_shell report commands on the command line.

- When you generate a report by choosing a menu command, the GUI opens a new report view and displays the report in both the report view and the console log view.

  If an empty report view is open, the GUI displays the report in that report view instead of opening a new report view.

- When you generate a report by entering the report command on the command line or by running a script, the GUI displays the report in the console log view.

  If an empty report view is open, the GUI also displays the report in the report view.

You use the report view to view report information and select reported objects. You can search for text in a report view. You can also save or append a report in a file or load a report from a file. Use the buttons at the top of the report view window to clear the view (remove the report text), save the report in a text file, display a report saved in a text file, and find text in a report.

In reports that list object names, such as design, cell, net, port, and worst-path timing reports, you can select an object by clicking its name (blue text) in the report view. The GUI displays the schematic for the design in which the object is located and magnifies the schematic to fit the selected object in the view. The name of the selected object also appears in the selection list, and the object is displayed in the selection color, which is white by default, in all schematic and layout views.

For more information about reports and the report view, see the following topics:

- Generating Object Reports
- Opening a Report View

### See Also

- The “Viewing Reports” topic in Design Vision Help

## Generating Object Reports

You can generate a report for one or more selected cells, ports, or nets. The GUI displays the report in a report view. You can also select objects in a report view that you want to examine in a schematic view.

To generate a report for one or more objects of the same type,

1. Select the objects.
2. Choose the appropriate report command in the Design menu.

   The associated report dialog box appears.
3. Click Selection in the report dialog box.

   The names of the selected objects appear.
4. Set other report options as needed.
5. Click OK.

If you click an object name (blue text) in the report view, the GUI selects the object in a schematic view and magnifies the schematic to fit the object in the view window.

### See Also

- Working With Reports

## Opening a Report View

You can open a blank report view and use it to display reports saved in text files.

To open a blank report view,

- Choose Window > New Report View.

To display a report from a file,

1. Click the 📂 button at the top of the report view window.

   The Open Report to File dialog box appears.
2. Select the file or enter its name in the “File name” box.
3. Click Open.

### See Also

- Working With Reports

## Selecting Objects by Name

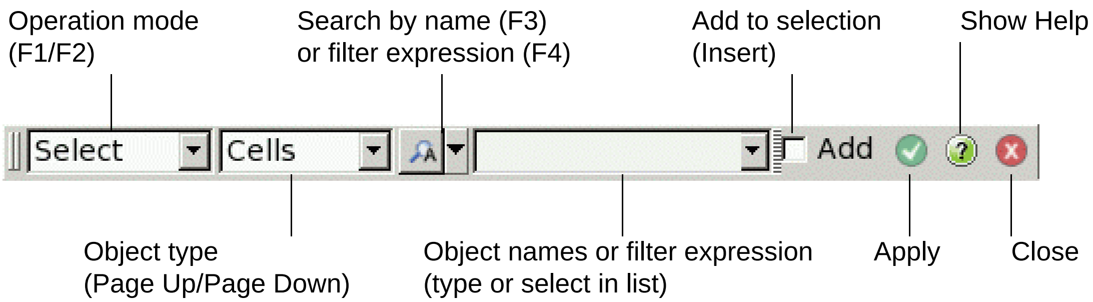
When you work in a schematic or layout view, you can quickly find objects of interest by using the Select By Name toolbar. You can select or highlight objects in the active view by pressing keys on the keyboard or by setting options on the toolbar. To specify the objects, you can type their names or a filter expression. You can also select object names or filter expression parts from a list.

To display the Select By Name toolbar and set the keyboard focus,

- Choose View > Toolbars > Select By Name.

The Select By Name toolbar appears above the status bar by default, as shown in Figure 4-1. For help using the Select By Name toolbar, click the ? button.

*Figure 4-1 Select By Name Toolbar*


Operation mode (F1/F2)
Search by name (F3) or filter expression (F4)
Add to selection (Insert)
Show Help
Select
Cells
Add
Object type (Page Up/Page Down)
Object names or filter expression (type or select in list)
Apply
Close

The Select By Name toolbar options are set by default to select cells by name and replace the current selection. To select one or more cells, you can type the cell names and press Return. You can type multiple names by separating them with blank spaces. You can also use wildcard characters (? or *) to specify a name pattern for multiple names.

You can set options to

- Change the operation that you want to perform

  The options are Select and Highlight. When you select objects, you can control whether the tool replaces or adds objects to the current selection.

- Change the type of objects that you want to find

  The options are Cells, Nets, Pins, or Ports.

- Change the search type

  The options are “Search by name” and “Search by filter.”

When you begin typing an object name or filter expression, the characters appear in the text box on the toolbar. You can allow the tool to complete a name or filter expression part that you are typing by pressing the Tab key. The tool completes the name or filter part to its longest match.

If the tool finds multiple objects or values that match the text you have typed, the object name or filter part list appears and the keyboard focus changes to the list. You can continue typing or select a name and close the list.

For more sophisticated object searches using reusable filters or regular expressions, use the Select By Name dialog box. For information about this dialog box, see the “Searching for Objects by Name or Type” topic in Design Vision Help.

**See Also**

- The “Selecting Objects by Name” topic in Design Vision Help

## Printing Schematic Views

You can print the schematic displayed in the active schematic view. Before printing a schematic view, make sure that a default printer is set in your .cshrc file.

To print the active schematic view,

1. Generate the schematic.
2. Make sure the schematic you want to print is in the active view.

   Click the corresponding tab at the bottom of the workspace if you need to make the view active.
3. Click the 🖨 button on the File toolbar or choose File > Print Schematic.

   The Print dialog box appears.
4. Select a print destination.

   You can send the schematic to a printer or save it in a file.
5. (Optional) Click the Properties button and set printer properties as needed in the printer properties dialog box that appears.
6. (Optional) Click the Options button and set the print options you require.
7. Click Print.

**See Also**

- The “Printing a Schematic View” topic in Design Vision Help

## Changing the Appearance of Schematics

For display purposes only, when the GUI generates a schematic, it applies the same display characteristics to all the objects or text of a given object type. The default display characteristics work well for most designs. However, if you need to customize schematics for a specific design or environment, you can use the View Settings panel to

- Change the colors for different types of design objects (cells, ports, pins, nets, buses, bus rippers, or hierarchy crossings)
- Change the colors or text sizes for different types of object names or object annotations
- Hide or display for different types of object names or object annotations

The changes apply only to the active view. However, you can save the new settings for use with in new schematic views that you open or in future Design Vision sessions.

To change visual display settings in the active schematic view:

1. Choose View > View Settings to open the View Settings panel if it is not already open.
2. Set the desired options.
3. Click Apply.

To save the new settings in your preferences file,

- On the View Settings panel, choose Options > Save to Preferences.

To load schematic view settings from your preferences file,

- On the View Settings panel, choose Options > Set from Preferences.

You can use the View Settings panel with any schematic view or DRC violation schematic. Each new schematic you open reads in the default display characteristics from your preferences file.

### See Also

- The “Changing Schematic Display Options” topic in Design Vision Help

## Saving an Image of a Window or View

You can save an image of a top-level GUI window or view window in an image file. The image format can be PNG (the default), BMP, JPEG, or XPM. The image shows the window exactly as it appears on the screen but without the window border or title bar. For example, if you save an image of the active schematic view, the image shows the visible portion of the schematic at the current zoom level and pan position.

- To save an image of the current top-level window or the active view, use the Save Screenshot As dialog box.
- To save an image of any open GUI window or view window, use the `gui_write_window_image` command.

You cannot save images of dialog boxes or other GUI elements such as toolbars or panels.

To save an image of the current window or active view,

1. Choose View > Save Screenshot As.

   The Save Screenshot As dialog box appears.

2. Select the file, or enter the path and file name in the “File name” box.

   The default format is PNG. you can specify a different format by using its extension to the file name.

3. (Optional) To save an image of the active view window instead of the top-level GUI window in which you are working, select the “Grab screenshot of active view only” option.

4. Click Save.

To save an image of any open GUI or view window, use the `gui_write_window_image` command to specify the file name, image format, and window name. Window instance names appear in the window title bars and on the Window menu.

Use the `-file` option to specify the file name. This option is required. For example, to save a PNG image of the active schematic view in a file named my_schematic.png, you can enter

```
prompt> gui_write_window_image -file my_schematic
```

You can use a file name extension or the `-format` option to specify the image format. The default image format is PNG. For example, to save an XPM image of the active layout view in a file named my_layout.xpm, enter either of the following commands:

```
prompt> gui_write_window_image -file my_layout.xpm
prompt> gui_write_window_image -file my_layout -format xpm
```

Use the `-window` option to specify the window, For example, to save a PNG image of the Layout window named Layout.1 in a file named mux_1.png, enter the following command:

```
prompt> gui_write_window_image -file mux_1.png -window Layout.1
```

You can use the `gui_write_window_image` command in a Tcl script if you want to save an image or a window or view when running the tool with a batch script. The following script example shows the commands you use to open the GUI, open a Layout window, save a PNG image of the Layout window, and close the GUI:

```
## Set the DISPLAY environment variable before opening the GUI.
## Replace “my_display_name” with the host name of your display terminal.
setenv DISPLAY my_display_name

## Open the GUI.
gui_start

## Create a new Layout window and store its name in a Tcl variable.
## Replace “window_name” with the name of your variable.
set window_name [gui_create_window -type LayoutWindow]

## Save an image of the window in a file named my_layout.png.
## Replace “window_name” with the name of your variable.
gui_write_window_image -file my_layout.png -window $window_name

## Remove the comment (#) from the next line to close the GUI here.
#gui_stop
```

The following script example includes the commands you need to use if you want to save a JPEG image of the congestion map:

```
## Set the DISPLAY environment variable before opening the GUI.
setenv DISPLAY my_display_name

## Open the GUI.
gui_start

## Create a new Layout window and store its name in a Tcl variable.
set window_name [gui_create_window -type LayoutWindow]

## Hide preroutes in the layout view.
gui_set_setting -window [gui_get_current_window -types Layout -mru] \
                 -setting showRoute -value false

## Display the congestion map.
gui_show_map -window [gui_get_current_window -types Layout -mru] \
             -map {Global Route Congestion} -show true
```

```
## Save an image of the window in a file named my_congestion.jpg.
gui_write_window_image -format jpg -window $window_name \
                        -file my_congestion.jpg

## Remove the comment (#) from the next line to close the GUI here.
#gui_stop
```

Similarly, you can save an image of a visual mode after displaying it in the Layout window.

### See Also

- The “Saving an Image of a Window” topic in Design Vision Help
- The `gui_write_window_image` command man page

# 5

# Solving Timing Problems

The Design Vision GUI provides tools for both high-level timing analysis and detailed path analysis. This chapter presents basic procedures and suggestions for solving timing problems by using the GUI. The chapter does not provide details about exercising particular features of the GUI, such as how to create a histogram or how to create a schematic. For detailed information about Design Vision features, see Design Vision Help.

For information about these procedures, see the following sections:

- Before You Analyze
- Creating a Timing Overview
- Choosing a Strategy for Timing Closure

## Before You Analyze

Before you analyze your design with the Design Vision GUI, follow your normal compile methodology to create a constrained gate-level design. A constrained gate-level design is a prerequisite to any timing analysis.

For more information about using the Design Vision tool to create a gate-level design, see Chapter 4, “Performing Basic Tasks.”

## Creating a Timing Overview

Creating an overview of the timing of your design is a valuable way to start any analysis of your design’s timing problems. A timing overview can help you decide what strategy to follow in gaining timing closure.

For example, a timing overview can help answer such questions as

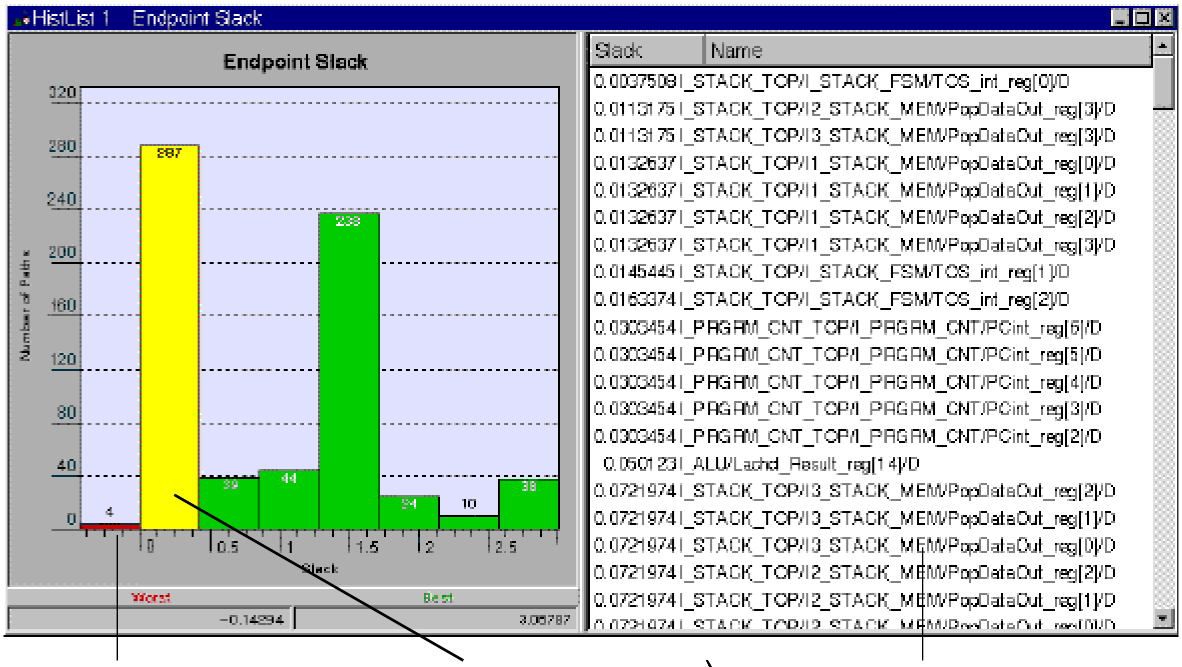
- Do I have many failing paths or just a few?
- Can I apply a local strategy for gaining timing closure?
- Do I need a global strategy for gaining timing closure?

To create a timing overview of your design,

1. Start with a constrained gate-level design.
2. Generate an endpoint slack histogram.

Figure 5-1 is a typical endpoint slack histogram for a design with a 4-ns clock cycle.

*Figure 5-1 Endpoint Slack Histogram*



| Slack | Name |
|---|---|
| 0.0037509 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[0]/D |
| 0.0113175 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0113175 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0145445 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[1]/D |
| 0.0163374 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[2]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[6]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[5]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[4]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[3]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[2]/D |
| 0.050123 | I_ALU/Lachd_Result_reg[14]/D |
| 0.0721974 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0721974 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0721974 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0721974 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0721974 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[1]/D |

Endpoints of failing paths (red)

Selected bin (yellow)

Endpoint names for selected bin

Using information such as that in Figure 5-1, you might decide on a local strategy if just a few paths are failing by a small margin (failing path endpoints are in one or more red bins to the left of 0 on the horizontal axis). Conversely, if you find that many paths are failing, or that the design is failing your timing goals by a large margin, you might choose a higher-level, or global, strategy for problem solving.

## Choosing a Strategy for Timing Closure

There is no single strategy that ensures quick and easy timing closure; however, a strategy based on the size and number of timing violations can be useful.

### Assessing the Relative Size of Your Timing Violations

This section suggests guidelines for describing the relative size of timing violations in your design. After you create an endpoint slack histogram, you can use these size guidelines to help you judge what strategy to use for timing closure.

What you consider to be small or large violations depends on the requirements of your design and your design process; however, assessing violation size as a percentage of clock cycle can be useful.

- Small violations

  Some designers consider small violations to be about 10 percent of the clock cycle or less.

- Large violations

  Some designers consider large violations to be about 20 percent of the clock cycle or greater.

- Medium violations

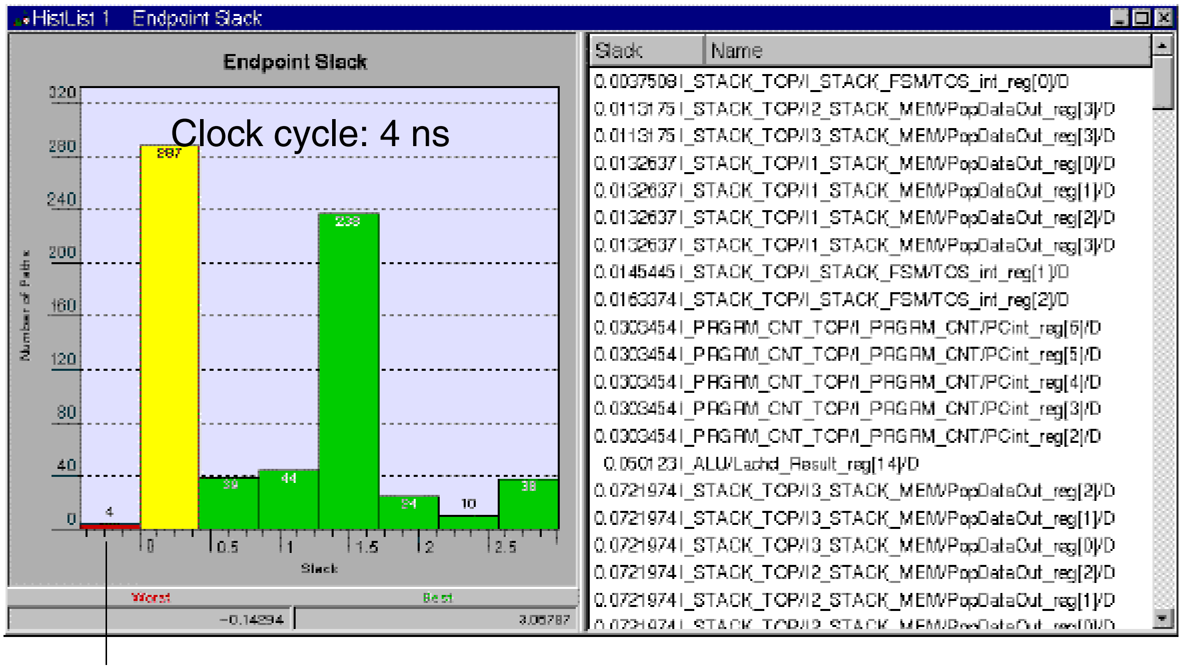
  Medium-size timing failures fall between the limits you set for large and small failures in your design or design process.

Whether your design is failing timing goals by large or small margins, the best strategy for timing closure is one that uses the least amount of runtime or number of design iterations to achieve timing goals. This principle underlies the methodology suggestions in this chapter. For more information about creating a timing overview, see “Creating a Timing Overview” on page 5-2.

## When Timing Violations Are Small

Figure 5-2 is a histogram of a design that is failing timing goals by a small margin. For example, no path is failing by more than 0.14 ns—that is, less than 10 percent of the 4-ns clock cycle (ignoring input and output delay). You can click any bin to see the endpoint names for the paths in the bin.

*Figure 5-2 Design With Small Timing Violations*



| Slack | Name |
|---|---|
| 0.0037508 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[0]/D |
| 0.0113175 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0113175 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0132637 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0145445 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[1]/D |
| 0.0163374 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[2]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[6]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[5]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[4]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[3]/D |
| 0.0303454 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[2]/D |
| 0.050123 | I_ALU/Lachd_Result_reg[14]/D |
| 0.0721974 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0721974 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0721974 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0721974 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0721974 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0721974 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[0]/D |

Endpoints of
failing paths (red)

Designs that fail by a small margin can have many failing paths or just a few. The endpoint slack histogram helps you to recognize quickly which case you have. Whether you have just a few failing paths or many, you can follow a global or local strategy in fixing the violations.

If suggestions for fixing small violations (either globally or locally) do not meet your timing goals, try applying the suggestions in "When Timing Violations Are Medium" on page 5-7 or "When Timing Violations Are Large" on page 5-9.

## Working Globally to Fix Small Violations

To apply a global methodology for fixing small violations, consider recompiling your design using the incremental option and a higher map effort. The incremental option saves runtime by using the current netlist as the startpoint for design improvements.

The incremental compile with higher map effort has the advantage of simplicity—that is, it requires little or no time spent in analyzing the source of timing problems. However, this method can change much of the logic in the design.

## Working Locally to Fix Small Violations

If you have a small number of paths with small violations, or if your violations seem to come from a limited set of problems on a few paths, a local strategy can be effective.

To use a local strategy for fixing small violations,

- Check hierarchy on failing paths

  The tool does not optimize across hierarchical boundaries. Thus, snake paths limit the tool's ability to solve timing problems on such paths.

- Look for excessive fanout on failing paths

  Because higher fanout causes higher transition times, excessive fanout can worsen negative slack on failing paths.

To check for hierarchy problems on failing paths,

1. Generate an endpoint slack histogram.
2. Click a bin that contains a failing path.

   A list of endpoints for failing paths is displayed.
3. Select the endpoint for the path you are interested in.
4. Generate a schematic to see which leaf cells are in which levels of hierarchy.

   If your critical path, for example, crosses multiple subblocks of a level of hierarchy, consider ungrouping these subblocks. The tool does not optimize across hierarchy boundaries. Thus, a subsequent compile has further opportunity to optimize the critical path when you ungroup such blocks.

To look for excessive fanout on failing paths,

1. Generate an endpoint slack histogram.
2. Select the endpoints for failing paths.

   Select the failing bin to see the endpoints.
3. Generate a timing report with the following options:

   ❍ net

   ❍ trans

   Send the report output to the report view. For more information about report generation, see “Working With Reports” on page 4-20 and the “Viewing Reports” topic in Design Vision Help.

4. Examine the report for pins with high transition times and nets with high fanout.

   Such paths are candidates for buffering or drive-cell resizing.

5. Create schematics of any paths you would like to see.

   A schematic view provides contextual information and details about the path and its components. Such information is often a prerequisite to understanding problems on the path.

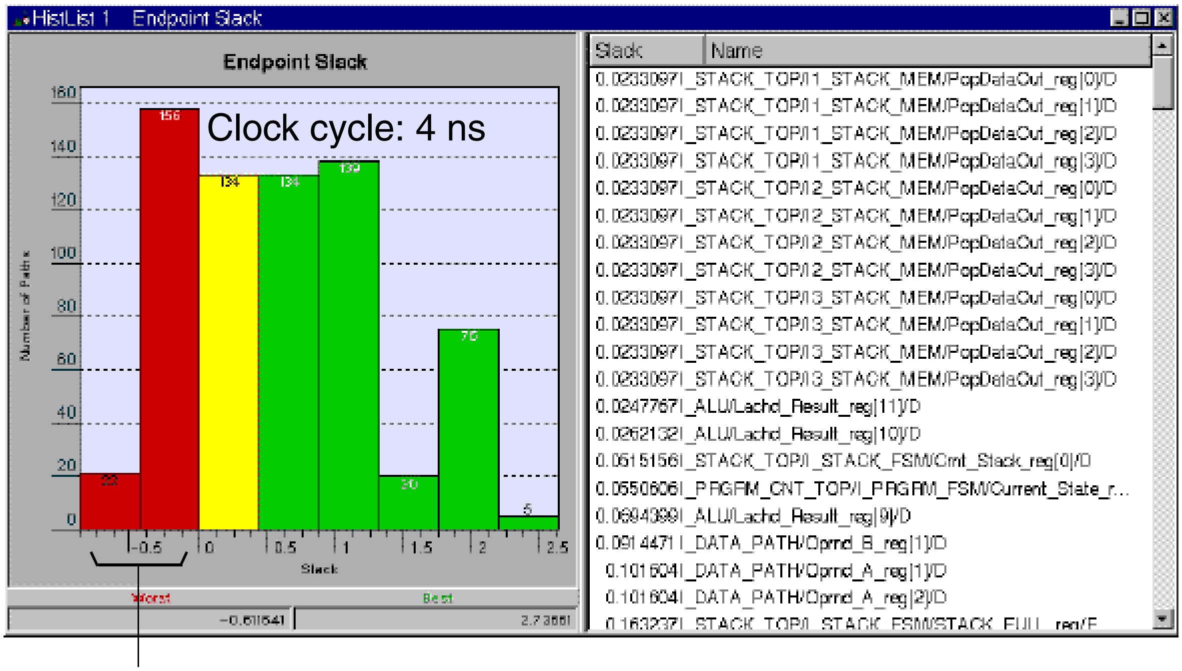
6. View fanin and fanout logic in schematics.

   This step can provide useful information about the logic that drives, or is driven by, the problem path. For example, after viewing fanin or fanout, you might choose to resize cells in those logic cones.

## When Timing Violations Are Medium

Figure 5-3 is a histogram of a design that is failing timing goals by margins that are between the large and small limits that are appropriate to your design methodology (for example, between 10 and 20 percent of the clock cycle). You can click a bin to see the endpoint names for paths the bin contains. A bin is yellow when selected. In Figure 5-3, one of the four bins containing endpoints of failing paths is selected.

*Figure 5-3 Design With Medium Timing Violations*



| Slack | Name |
|---|---|
| 0.0233097 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0233097 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0233097 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0233097 | I_STACK_TOP/I1_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0233097 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0233097 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0233097 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0233097 | I_STACK_TOP/I2_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0233097 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[0]/D |
| 0.0233097 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[1]/D |
| 0.0233097 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[2]/D |
| 0.0233097 | I_STACK_TOP/I3_STACK_MEM/PopDataOut_reg[3]/D |
| 0.0247767 | I_ALU/Lachd_Result_reg[11]/D |
| 0.0262132 | I_ALU/Lachd_Result_reg[10]/D |
| 0.0515156 | I_STACK_TOP/I_STACK_FSM/Crnt_Stack_reg[0]/D |
| 0.0550606 | I_PRGRM_CNT_TOP/I_PRGRM_FSM/Current_State_r... |
| 0.0694399 | I_ALU/Lachd_Result_reg[9]/D |
| 0.0914471 | I_DATA_PATH/Oprnd_B_reg[1]/D |
| 0.101604 | I_DATA_PATH/Oprnd_A_reg[1]/D |
| 0.101604 | I_DATA_PATH/Oprnd_A_reg[2]/D |
| 0.163237 | I_STACK_TOP/I_STACK_FSM/STACK_FULL_reg/E |

Endpoints of failing paths

When negative slack values are medium, you can use the tool to investigate further and focus your recompile on a critical range of negative slack values for path groups. Focusing your compile effort on a critical range can improve worst negative slack and total negative slack.

Defining a critical range for path groups offers the advantage of concentrating compile effort and runtime on those areas that most need it.

To investigate and focus a recompile by defining a critical negative slack range for path groups,

1. Create a path slack histogram for each path group in your design.

   Start with an arbitrary value of 1000 for the number of paths to include in each histogram. Raise or lower this value depending on the number of failing paths. The goal is to choose a value that shows you all or nearly all of the failing paths.

2. Decide on a critical range for each path group (note the values for use in step 3).

   When deciding on a critical range, choose a range that allows the tool to focus on the worst endpoint violations without too large an increase in runtime.

3. For example, some designers apply one of the following guidelines to decide on a critical range:

   ❍ Use a range that includes the worst 50 paths in a group.

   ❍ Use a range equal to one generic cell delay in your technology.

   These are rough guidelines; for subsequent compiles you can adjust your critical range as necessary.

4. Set a critical range for each path group.

   Using the values you decided on in step 2, set the critical ranges for each path group with the `group_path` command. For example,

   ```
   prompt> group_path -name my_clock -critical_range 0.25
   ```

5. Recompile the design.

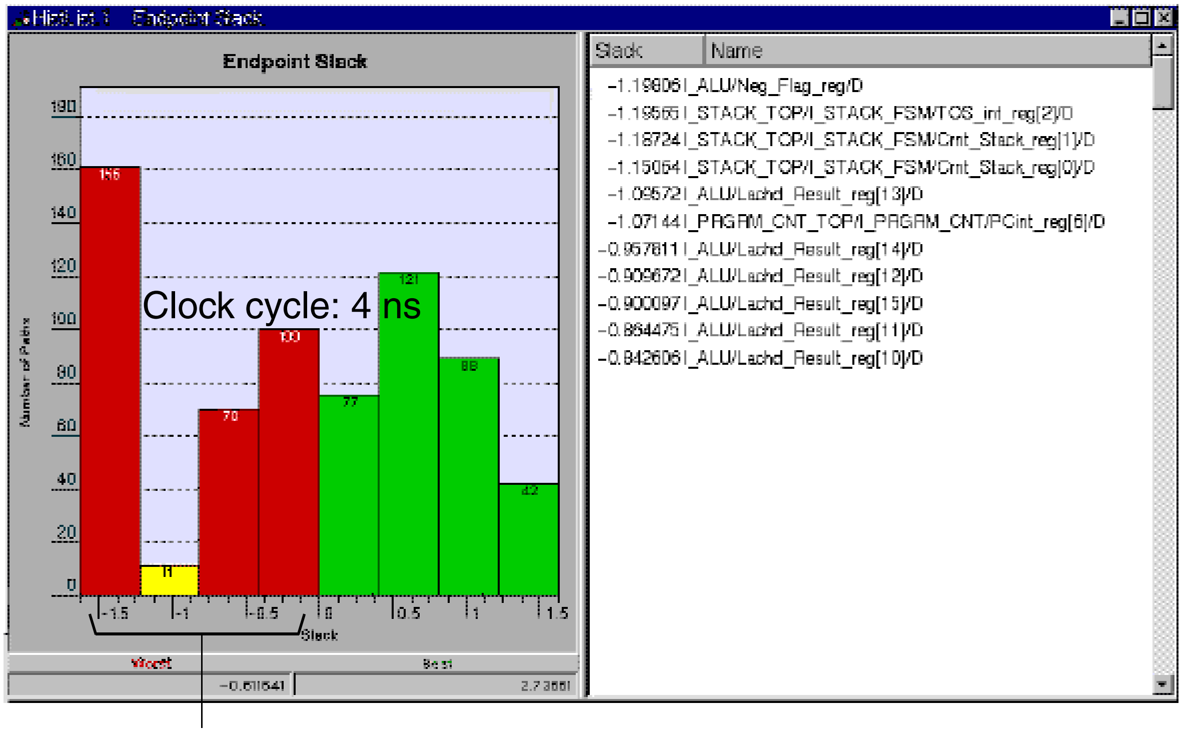
   With a critical range defined, the compile effort is now focused. However, you can also choose to increase the compile effort over your previous compile. In the Compile Design dialog box, select medium or high effort. Select the “Incremental mapping” option to direct the tool to use the current netlist as a starting point for design improvements. An incremental compile can save runtime when your design does not require fundamental changes.

If suggestions in this section don’t meet your timing goals, try applying the suggestions in “When Timing Violations Are Large” on page 5-9.

## When Timing Violations Are Large

Figure 5-4 shows a design that is failing timing goals by a large margin. You can click a bin to see the endpoint names for the paths it contains.

*Figure 5-4 Design With Large Timing Violations*



| Slack | Name |
|---|---|
| −1.19806 | I_ALU/Neg_Flag_reg/D |
| −1.19565 | I_STACK_TOP/I_STACK_FSM/TOS_int_reg[2]/D |
| −1.18724 | I_STACK_TOP/I_STACK_FSM/Crnt_Stack_reg[1]/D |
| −1.15064 | I_STACK_TOP/I_STACK_FSM/Crnt_Stack_reg[0]/D |
| −1.09572 | I_ALU/Lachd_Result_reg[13]/D |
| −1.07144 | I_PRGRM_CNT_TOP/I_PRGRM_CNT/PCint_reg[6]/D |
| −0.957811 | I_ALU/Lachd_Result_reg[14]/D |
| −0.909672 | I_ALU/Lachd_Result_reg[12]/D |
| −0.900097 | I_ALU/Lachd_Result_reg[15]/D |
| −0.864475 | I_ALU/Lachd_Result_reg[11]/D |
| −0.842606 | I_ALU/Lachd_Result_reg[10]/D |

Endpoints of failing paths

Fixing large violations can require a high-level strategy to improve your design's performance. To fix large timing problems, consider any of the following changes:

- Modify your constraints.

  For example, increase the clock cycle or adjust time budgeting for the block or chip.

- Change the target technology.

  For example, target a higher performance technology.

- Modify the RTL.

  For example, you can move late-arriving signals such that you minimize their path length.

# 6

# Solving Floorplan and Congestion Problems

The Design Compiler Graphical Layout window in Design Vision provides tools that can help you to analyze and debug physical problems related to Design Compiler topographical synthesis. This chapter provides general and specific information about analyzing the physical placement of critical timing path objects, avoiding correlation issues that can result from incorrect or missing physical constraints, and visually examining floorplan-related congestion and identifying the causes of congestion hotspots.

For information about performing visual analysis in the Layout window, see the following sections:

- Physical View Advantage
- Before You Start
- Preparing for Physical Analysis
- Using the Layout Window
- Visualizing the Physical Layout
- Validating Physical Constraints
- Debugging QoR Issues Related to the Floorplan and Placement
- Visually Analyzing Congestion

## Physical View Advantage

As part of the Design Compiler Graphical package, the Design Vision GUI provides the Layout window for viewing and analyzing the physical aspects of a design that you are optimizing by using the Design Compiler topographical technology. The Layout window contains a layout view that displays physical design information such as

- Die area and core placement area
- Ports
- Cells, including standard cells, hard and soft macro cells, I/O cells, block abstractions, physical hierarchy blocks, black boxes, physical-only cells, cell keepout margins, and cell orientations
- Pins, including macro pins and I/O pads
- Physical constraints, including placement blockages, site rows, bounds, pin guides, preroutes (net shapes, vias, and user shapes), tracks, and wiring keepouts
- Relative placement groups
- Voltage areas

The Layout window is the physical design working environment for the GUI. Layout views provide the focal points for viewing and analyzing the physical layout of your design. The Layout window provides visually customizable layout views with the following tools:

- An Overview panel for quickly magnifying and traversing the active layout view or changing from one layout view to another
- A View Settings panel for controlling object visibility and selection and customizing object appearance in the active layout view
- Interactive left-button mouse tools that you can use to select, highlight, and query objects, magnify and pan your view of the design, and draw rulers to measure distances
- Lithographic and user grids that you can display or hide in the layout view
- Flylines for examining connections between cells or pins in your floorplan
- A congestion map for identifying areas of high congestion in your floorplan
- Cell and pin density maps for identifying areas of high cell or pin density in your floorplan
- Visual modes for examining specific floorplan data in color overlays on the layout view

Note that standard cells are not visible by default in the Layout window. To view and select standard cells in the layout view, you must change the standard cell visibility and selection options on the View Settings panel. For more information, see the “Controlling Object Visibility” and “Controlling Object Selection” topics in Design Vision Help.

## Before You Start

Before you can view the physical layout of a design, you must

1. Provide any necessary physical design setup information, such as libraries, TLUPlus files, preferred routing layer directions, and ignored layer settings
2. Link the design without any errors

For information about performing these steps, see the *Design Compiler User Guide*.

You can view a design and analyze the physical aspects of a design in the Layout window before or after you optimize the design. Before optimization, the Layout window can display an elaborated GTECH design or a partially-synthesized design. Use the Layout window to

- Validate the physical constraints for your floorplan
- View the locations for block abstractions, physical hierarchy blocks, and preplaced macro cells
- View cross-selected standard cells that have been mapped to specific locations by either the `set_cell_location` command or topographical technology virtual placement
- Select the cells in a logic design view such as the hierarchy browser or a schematic view

If you select unmapped GTECH cells or mapped standard cells that have not been assigned a location, they appear at the layout view origin (0,0). Note that bounds, relative placement groups, and the map and visual modes are not available in the Layout window until you optimize the design.

To view a design after optimization, you must optimize the design using the Design Compiler topographical technology. You can either optimize the design during the current session or load the optimized design from a .ddc file.

After optimization, the Layout window displays the optimized floorplan. You can

- Debug QoR Issues related to the physical aspects of your design, including
  - ❍ Why particular cells have a given drive strength
  - ❍ Why particular timing paths contain long buffer chains that are not related to high fanout

  - Why particular I/O paths contain high concentrations of buffers
  - What causes the huge transition or capacitance on particular pins

- Validate any user-defined physical constraints that you have applied to the design
- Analyze congested areas in the physical design

After performing QoR analysis, you can identify the next step, which might be one of the following:

- If there is a simple way to fix or eliminate the QoR issues, you might decide to continue with the back-end flow.
- If you identify problems in the design source, such as your RTL, timing constraints, or physical constraints, you might need to rerun synthesis with updated source files.

**See Also**

- Preparing for Physical Analysis
- Using the Layout Window

## Preparing for Physical Analysis

The following steps illustrate the typical setup tasks in the Design Vision tool before you can analyze a design in the Layout window:

1. Start the tool in topographical mode without the GUI by entering the following command:

   ```
   % design_vision -topographical_mode -no_gui
   ```

2. Set up the logic and physical libraries required for topographical mode.
3. Read in the .ddc netlist synthesized in topographical mode, and make sure that the design links correctly.
4. Open the GUI by entering the following command:

   ```
   design_vision-topo> start_gui
   ```

5. Open the Layout window.

Alternatively, you can perform steps 2 through 4 by running a Tcl script. The following example shows a basic setup script:

```
source echo dct.setup.tcl   # Sourcing DC Ultra Topographical setup
read_ddc dct.opt.ddc   # Reading DC Ultra Topographical-synthesized .ddc
current_design top
link
```

```
start_gui
```

Design Compiler Graphical features are enabled with the DC-Extension license, in addition to any other licenses for your current design configuration. These features are available in topographical shell (dc_shell-topo). If the DC-Extension license is not available, the tool issues the following error message:

```
Error: This site is not licensed for 'DC-Extension'. (SEC-51)
```

If you see this message, contact your local Synopsys representative.

### See Also

- Before You Start
- Using the Layout Window

## Using the Layout Window

The Design Vision Layout window in Design Compiler Graphical has a similar user interface and the same look and feel as the IC Compiler Layout window. However, the Layout window in Design Compiler Graphical is designed to provide the features that you need to analyze and debug synthesis-related problems.

The Layout view provides the focal point for viewing and analyzing your physical floorplan constraints and congested areas in your design. Use the Overview panel and the View Settings panel to adjust the layout view display when you examine objects and validate the applied constraints.

- You can magnify and traverse the layout view by clicking or dragging the pointer on the Overview panel. If multiple layout views are open, you can change the active view from one view to another.
- You can change layout view display properties by setting options on the View Settings panel, including object visibility, object selection, and object display styles.

The Layout window is not designed to be used for the following applications:

- Floorplan exploration

  You cannot use the Layout window as a floorplan exploration tool because it does not allow you to view user-supplied physical constraints until after you have performed topographical-based synthesis.
- Floorplan or physical constraint editing

  The Layout window does not allow you to change any physical constraints by using the layout view. You must apply all required physical constraints before running the `compile_ultra` command. The tool does not support physical constraint changes between multipass synthesis runs.

If you significantly change or update the .ddc data, the Layout window closes automatically.

To learn more about the Layout window, see the following topics:

- Opening the Layout Window
- Performing Floorplan Exploration

**See Also**

- Before You Start
- Preparing for Physical Analysis

## Opening the Layout Window

To open the Layout window,

- In the Design Vision window, click the button on the Layout toolbar or choose Windows > New Layout Window.

The layout view, Overview panel, and View Settings panel are opened by default when you open the Layout window. When you open the Layout window, all selected objects are deselected.

**See Also**

- Using the Layout Window
- The “Opening the Layout Window” topic in Design Vision Help

## Performing Floorplan Exploration

Design Compiler Graphical allows you to perform floorplan exploration within the synthesis environment by using the IC Compiler floorplanning tools in the IC Compiler Layout window. Although you use the IC Compiler Layout window, the interface between floorplan exploration in Design Compiler Graphical and the IC Compiler Layout window is transparent, allowing you to move seamlessly between the Design Vision and IC Compiler Layout windows.

For information about performing floorplan exploration in Design Compiler Graphical, see the *Design Compiler User Guide*.

**See Also**

- Using the Layout Window
- The “Using Floorplan Exploration Tools” topic in Design Vision Help

## Visualizing the Physical Layout

The Design Vision layout view, like the IC Compiler layout view, displays a flat representation of the physical design, and it can display only one top-level design at a time. You do not need to set floorplan constraints before viewing a design in the layout view.

You can cross-select design objects between schematic and layout views. This helps you to understand the functions of the selected cells. When you select logic design objects in the hierarchy browser or a schematic view, the objects are automatically cross-selected in the layout view.

Layout data can be densely packed with overlapping objects. You can control the visibility (display or hide) and selection (enable or disable) of individual object types by setting options on the View Settings panel. You can also customize object properties such as color and fill pattern, and set other layout and object display options. If you open multiple layout views, you can set different options for each view.

You can open multiple layout views in the Layout window. If you change settings in the active layout view and want to use the same settings in another layout view or during a future session, you can save them in your preferences file, .synopsys_dv_prefs.tcl. You can also restore previously saved display properties by loading them from your preferences file.

Note:

> The Design Vision Layout window is not designed for floorplan or physical constraint editing. If you change the netlist or physical constraint data for a design when the Layout window is open (for example, by using netlist editing commands such as `change_link`), the GUI immediately closes the Layout window.

You can visualize the physical layout as explained in the following topics:

- Opening a New Layout View
- Navigating Through Layout Views
- Displaying Grid Lines
- Displaying Cell Orientations
- Drawing Rulers
- Examining Block Abstractions and Physical Hierarchy Blocks
- Expanding Hierarchical Cells

- Examining Relative Placement Groups
- Examining Voltage Areas
- Analyzing Cell Connectivity
- Analyzing Cell Placement
- Analyzing Cell and Pin Density
- Changing the Appearance of the Layout View

**See Also**

- The “Viewing the Floorplan” topic in Design Vision Help

## Opening a New Layout View

You can open multiple layout views in the same Layout window and work simultaneously with different areas of the design side by side.

To open a layout view,

- Choose View > New Layout View.

## Navigating Through Layout Views

In the Layout window, the Overview panel shows you what portion of the design is visible in each open layout view.

- The portion of the design displayed in the active layout view is shown as a solid yellow rectangle.
- The portions of the design displayed in other layout views are shown as solid gray rectangles.

You can use the Overview panel to quickly magnify or traverse the design in the active layout view. When multiple layout views are open, you can change to a different layout view. For more information about the Overview panel, see Design Vision Help.

## Displaying Grid Lines

You can display or hide the lithography grid and the user grid. Both grids are hidden by default.

To display or hide the lithography grid,

- Choose View > Grid > Show Litho Grid.

To display or hide the user grid,

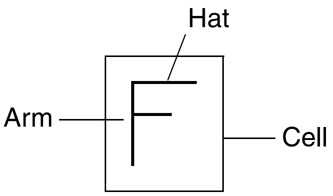
- Choose View > Grid > Show User Grid.

To switch between the default grid spacing and ten times the default grid spacing,

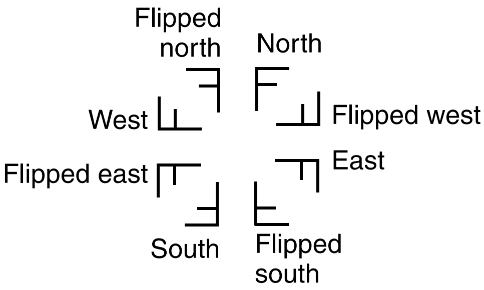
- Choose View > Grid > Cycle Grid Spacing.

## Displaying Cell Orientations

The GUI represents cell orientation in the layout view as an F. The position of the “hat” of the F indicates the first direction; the “arm” indicates the second direction.



Cells can be oriented in any of the following ways:



To display cell orientations in the active layout view,

1. On the View Settings panel, make sure the Cell visibility (Vis) option is selected.
2. Click the Cell expansion button (plus sign).
3. Select the Orientation visibility (Vis) option.

   A check mark on the option indicates that the cell orientations are visible.

4. Click Apply.

## Drawing Rulers

You draw rulers to measure distances in a layout view. For example, you can measure the distance between two cells or between two pins on a net.

A ruler can be composed of one or more horizontal, vertical, or diagonal segments. The distances from the beginning of the ruler are labeled at the ends of each segment and at every tenth tick mark within a segment.

While you draw a ruler segment, the GUI displays a preview image of the ruler in the layout view and displays the coordinates for the current pointer position on the status bar. The preview image indicates the distance between the initial point and the current pointer position.

To draw a ruler,

1. Click the Ruler tool button on the Mouse Tools toolbar or choose View > Mouse Tools > Ruler Tool.
2. Click the location in the layout view where you want to begin the ruler.
3. Move the pointer in the direction that you want to draw the ruler segment, and click to define the segment.

   To draw a diagonal segment, press the Shift key when you move the pointer.
4. Click where you want to end the ruler segment.

   The ruler segment appears.
5. Repeat steps 2 through 4 to draw additional ruler segments.
6. When you finish the last segment, press the Esc key or right-click and choose End Ruler.

### See Also

- The “Drawing Rulers” topic in Design Vision Help

## Examining Block Abstractions and Physical Hierarchy Blocks

If your design contains block abstractions or physical hierarchy blocks, you should visually check them for correct site locations when you validate the floorplan. You should also visually check for block abstractions or physical hierarchy blocks in highly congested areas when you examine the congestion map.

Block abstractions are an extension to interface logic models. However, when you use block abstractions, the tool saves both the full design and the block abstraction information in the same .ddc file. When reading the design in as a block abstraction at the top level, the tool loads only the interface logic from the .ddc file.

Note:
Design Vision does not support interface logic models. Use block abstractions instead.

A physical hierarchy is a block that you create from a hierarchical cell by using the `set_physical_hierarchy` command. For information about this command, see the man page. The layout view displays a physical hierarchy as a rectangular block that indicates the area in which the leaf cells have been placed. You can view cell placement, pin placement, or net connections within the block by selecting the cells, pins, or nets in the hierarchy browser or a schematic view.

You can set options on the View Settings panel to control the visibility and selection of block abstractions and physical hierarchy blocks in the layout view and to change their color and fill pattern.

You can distinguish block abstractions and physical hierarchy blocks by the value of the `cell_type` attribute, which you can view by using the Query tool or the Properties dialog box.

- The cell type value for a block abstraction is Block Abstraction.
- The cell type value for a physical hierarchy block is Physical Hierarchy.

**See Also**

- Expanding Hierarchical Cells
- The “Examining Block Abstractions” and “Examining Physical Hierarchy Blocks” topics in Design Vision Help
- The *Design Compiler User Guide*

  Provides in-depth information about block abstractions and physical hierarchy models

## Expanding Hierarchical Cells

You can expand hierarchical cells such as soft macros and block abstractions to view the logic (cells, ports, pins, nets, and so forth) inside them. When hierarchical cells are expanded, you can use Layout window analysis tools to analyze their internal cell placement. You can control the visibility and selection of these objects by using the same View Settings panel options you use for other objects in the layout view.

To expand hierarchical cells,

1. Select or enter a value greater than 0 in the Level box on the View Settings panel.
2. Click Apply.

To collapse (close) hierarchical cells,

1. Select or enter 0 in the Level box on the View Settings panel.
2. Click Apply.

**See Also**

- Examining Block Abstractions and Physical Hierarchy Blocks
- The “Examining Block Abstractions” topic in Design Vision Help

## Examining Relative Placement Groups

If your design contains relative placement groups, you should visually check the groups for correct site locations when you validate the floorplan. You should also visually check for relative placement groups in highly congested areas when you examine the congestion map.

You can examine relative placement group structures in your floorplan by viewing them in the layout view. You can control the visibility, selection, and display properties of relative placement groups and the visibility of relative placement group labels in the active layout view by setting options on the View Settings panel.

If you have defined relative placement groups in your design, you can use the layout view to

- Determine how the DC Ultra topographical technology placed the relative placement groups, based on the constraints that you provided.

  You should examine the size and location of each group.

- Determine how the DC Ultra topographical technology placed relative placement groups that are not constrained.

  This allows you to find answers to such questions as what is the best topographical technology-derived location for a relative placement group.

- Examine the timing paths that pass through a relative placement group relative to other relative placement groups that they pass through.
- Debug the relative placement group constraints based on the visual feedback in the layout view.

### See Also

- The “Examining Relative Placement Groups” topic in Design Vision Help

## Examining Voltage Areas

If your design contains voltage areas, you should visually check the areas for correct site locations when you validate the floorplan. You can examine the voltage areas in your floorplan by viewing and probing them in the layout view. You can control the visibility and display properties of voltage areas and the visibility of voltage area labels in the active layout view by setting options on the View Settings panel.

If you have defined voltage areas in your design, you can use the layout view to

- Determine how the DC Ultra topographical technology placed the voltage areas. You should examine the size and location of each area.
- Select the standard cells in the hierarchical block related to a voltage area and examine them in the layout view to make sure they are all placed within the area outline.

You can also examine voltage areas by coloring them in a visual mode overlay on the layout view. For more information, see “Analyzing Cell Placement” on page 6-14.

### See Also

- The “Examining Voltage Areas” topic in Design Vision Help

## Analyzing Cell Connectivity

You can select an object and display flylines in the layout view to see the locations of the objects that have net connections to the selected object. Flylines represent unrouted straight-line pin-to-pin connections.

A flyline shows the connection between the pins on two cells or a pin and a port. You can display flylines to all types of objects or just to macro cells, I/O cells, or other selected objects.

To display or hide the flylines,

1. In the Layout window, click the button on the Analysis toolbar or choose View > Flylines.

   The Flylines Settings panel appears.

2. Select a cell.

   Flylines appear between the selected cell and each cell to which it has a net connection.

To facilitate your analysis, you can adjust the flyline display and style characteristics in the active layout view by setting options on the Flylines Settings panel. You can set options to

- Select the type of cell connections you need to display

  You can display flylines to all cells, macro cells, I/O cells, or selected cells.
- Combine the individual flylines into a minimum span tree
- Skip one or more logic levels
- Set the maximum number of fanouts to display for a net
- Filter nets by type or name

  You can display or hide flylines for signal nets, clock nets, power nets, and ground nets. You can include or exclude individual flylines by specifying net names or regular expressions.
- Change the flyline color

You can also set an option to display information about the selected cell on the Query panel.

### See Also

- The “Displaying Flylines” topic in Design Vision Help

## Analyzing Cell Placement

You can analyze cell placement in your floorplan by using visual modes to display design objects or other data in a color overlay on the layout view. A visual mode allows you to focus on the objects of interest while dimming other visible objects.

A visual mode groups cells or other objects into categories called bins. The layout view displays the contents of each bin in a different color. You can set visibility options on the Visual Mode panel to display or hide the objects in each bin.

The Layout window provides the following visual modes:

- Snapshot visual mode

  This is the default visual mode. You can analyze the placement quality of cells and other objects in your design by using snapshot visual mode to examine hierarchical cells and design logic in the layout view.

  You can identify logic blocks and hierarchical cells, leaf cells, and macro cells by selecting them in the hierarchy browser or a schematic view and coloring their physical locations in the layout view. By using different colors for each cell or logic block, you can identify problems with the distribution of placed cells that can result in areas with poor timing or high congestion.

- Hierarchy visual mode

  Use the hierarchy visual mode to display a high-level view of the placement quality of logic blocks and hierarchical cells in your physical design. You can color all the cells on a particular hierarchy level or just the hierarchical cells that you select. Each color represents a different hierarchical cell.

- Voltage areas visual mode

  Use the voltage areas visual mode to display a high-level view of the placement quality of cells in the voltage areas of a multivoltage design. A voltage area is a placement area for core cells in a logic block that operates under a single voltage level. Each voltage area corresponds to one or more hierarchical cells in the logic design.

  Voltage areas visual mode provides a separate bins for the cells in each voltage area and a bin for each of the following types of power management cells: regular level shifters, enable level shifters, always-on cells, and isolation cells.

- Highlight visual mode

  Use the highlight visual mode to group highlighted objects by color. You can focus on the objects that you highlight with particular colors by displaying or hiding individual bins. You can also select all the objects in a bin.

For more information about visual modes, see the following topic:

- Using Visual Modes

## Using Visual Modes

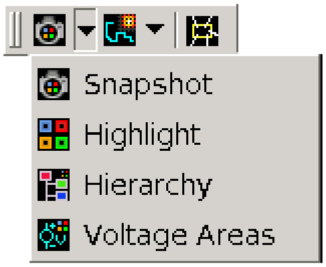
To display or hide the current or default visual mode,

- Click the visual mode button on the Analysis toolbar, or choose View > Visual Mode.

The visual mode button that appears on the Analysis toolbar changes to show the active visual mode (snapshot mode by default). After you disable visual mode, you can click the button to quickly redisplay the most recently active visual mode.

To display a different visual mode, you can

- Click the arrow button and choose a command from the Visual Mode menu on the Analysis toolbar.



- Select the visual mode name in the list on the Visual Mode panel.

You can view information about the active visual mode in the legend on the Visual Mode panel. Each bin displays the color and fill pattern, the data count (total number of objects in the category or values in the range), and optionally the color exaggeration value (hidden by default). The colored histogram bars on the right side of the legend represent the relative distribution of the objects or values.

In a visual mode that colors design objects, you can select or deselect the objects in each bin. In a visual mode that colors discrete, unrelated sets of objects or other information, you can reorder the bars in the legend.

Only one visual mode can be active at a time in the active layout view. If you need to examine more than one visual mode at the same time, you can either switch to a different visual mode or open multiple layout views and activate a different visual mode in each view.

**See Also**

- Analyzing Cell Placement
- The “Using Visual Modes” topic in Design Vision Help

## Analyzing Cell and Pin Density

You can analyze cell density and pin density in your floorplan by using map modes to display a cell density map or pin density map in a color overlay on the layout view. A map mode allows you to focus on areas of high cell or pin density in your design while dimming other visible objects.

A cell or pin density map divides the core area into a grid of colored boxes. The boxes are colored and labeled to show the cell or pin density levels. Each map color represents a range of density values called a bin. The ranges are calculated between minimum and maximum thresholds.

The cell or pin density map legend displays the color, the data count, and optionally, the color exaggeration level for each bin. The colored histogram bars on the right side of the legend represent the relative distribution of density values in the bins. You can set visibility options on the Map Mode panel to display or hide the densities in each bin.

For more information about map modes, see the following topics:

- Analyzing Cell Placement
- Visually Analyzing Congestion

## Changing the Appearance of the Layout View

The View Settings panel provides options you can use to set display properties in the active layout view. You can also save the current settings in your preferences file, or load settings from the preferences file. If you open multiple layout views, you can set different options for each view.

You can set options on the View Settings panel to

- Control object visibility and selection
- Display or hide object labels
- Change object display styles such as colors or fill patterns
- Set layout view display options for cell orientations and cell keepout margins (display or hide), the brightness level, and the hierarchy level for block abstractions

You can set visibility or selection options or change style properties for object types or subtypes. Object subtypes are categories of objects by property or attribute. For example, when cells are visible, you can display core cells and hide macro cells.

By displaying or hiding particular object types, you can visually inspect just the physical layout data that you are interested in viewing while ignoring unrelated data. By enabling or disabling the selection of particular object types, you can control which types of objects are selected when you click or drag the pointer in a layout view.

For more information, see the following topic:

- Setting Layout View Properties

## Setting Layout View Properties

To display or hide the View Settings panel,

- Choose View > Toolbars > View Settings.

  A check mark beside the command on the Toolbars menu indicates that the View Settings panel is visible.

To change layout view display properties in the active layout view,

1. Set options as needed on the View Settings panel.
2. Click Apply.

By default, when you change settings on the View Settings panel, you must click Apply before the changes take effect in the active view. If you prefer, you can set the panel to automatically apply your changes as soon as you make them.

To enable or disable the automatic apply mechanism,

- Choose Options > Auto Apply.

  A check mark beside the command on the Options menu indicates that the auto apply mechanism is enabled.

Alternatively, when the automatic apply mechanism is not enabled, you can reverse changes that you have not already applied.

To reverse unapplied changes,

- Choose Options > Cancel Changes.

You can customize how objects appear in the layout view by changing their style properties. Object styles set the appearance of objects in the active layout view. You can set the color, fill pattern, outline style, outline width, or exaggeration value for individual object types or layers.

The tool does not automatically save view settings when you close the GUI exit the session. If you change layout view settings during a session and want to use the same settings in a future session, you can save them in your preferences file. You can also restore previously saved view settings by loading them from your preferences file.

To save the current settings for the active layout view,

- Choose Options > Preferences > Save to Preferences.

To restore the most recently saved layout view settings,

- Choose Options > Preferences > Set from Preferences.

**See Also**

- Changing the Appearance of the Layout View
- The “Changing Layout Display Properties” topic in Design Vision Help

## Validating Physical Constraints

The layout view provides visual feedback about the orientation and physical placement of physical design objects and constraints, such as

- Die area and core placement area
- Port locations
- Macro cell and pin locations
- Cell orientations and keepout margins
- Site rows with cell sites
- Bounds
- Pin guides
- Placement blockages
- Preroute net shapes, vias, and user shapes
- Relative placement groups
- Routing tracks
- Voltage areas
- Wiring keepouts

By visually examining these objects, you can avoid the correlation problems that can occur due to incorrect or missing physical constraints.

You can examine the physical constraints in your floorplan by viewing and probing them in the layout view in the following ways:

- Display or hide the core area, ports, cells, cell orientations and keepout margins, pins, site rows, bounds, placement blockages, preroute shapes and vias, relative placement groups, routing tracks, voltage areas, and wiring keepouts
- Select or highlight the die area, the core area, ports, cells, pins, bounds, placement blockages, relative placement groups, and wiring keepouts

- Query (display information about) the die area, the core area, ports, cells, pins, bounds, placement blockages, preroute shapes and vias, relative placement groups, voltage areas, and wiring keepouts

In addition, you can view object properties for the die area, the core area, ports, cells, pins, bounds, placement blockages, relative placement groups, and wiring keepouts by using the Properties dialog box (choose Edit > Properties).

By default, the core area, ports, cells, and preroutes are visible, and the other physical constraint object types are hidden when you open the Layout window. The die area is always visible. If your design contains preroutes, you should examine them to make sure the tool honors the other physical constraints when it creates the prerouted net shapes.

Physical constraint validation provides the following benefits:

- Helps you improve the physical constraints and achieve better results.

  For example, you can identify the need for placement blockages to plug gaps between macros that the synthesis tool might consider free to use but your physical implementation tool does not use.

- Helps you identify mismatched results between Design Compiler topographical technology and the IC Compiler tool.

  For example, incorrect application of physical constraints during synthesis can lead to ignored placement blockages in the physical implementation tool.

**See Also**

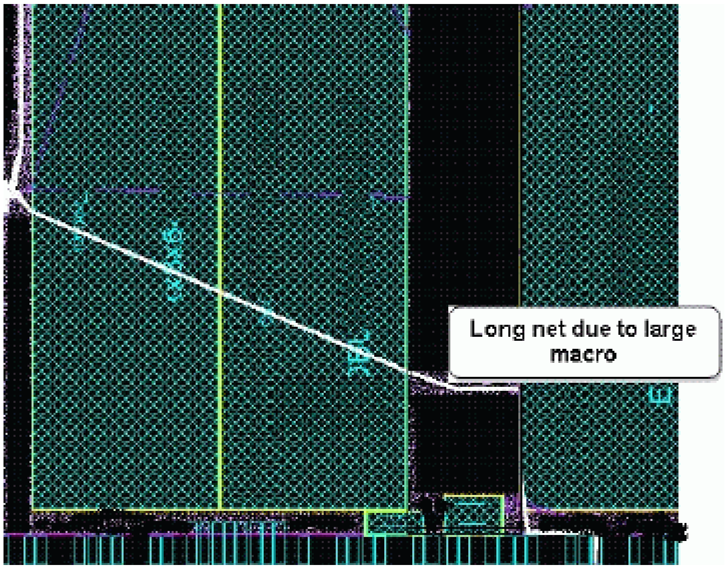
- The “Viewing the Floorplan” topic in Design Vision Help
- The “Examining Physical Constraints” topic in Design Vision Help

## Debugging QoR Issues Related to the Floorplan and Placement

The Design Vision layout view allows you to debug the physical design problems that cause QoR degradation, especially timing degradation. The layout view provides visual feedback about the physical placement of timing path objects. By visually examining the critical path in the layout view, you can find answers to such questions as

- Why are certain cells of a given drive strength?
- Why does a path contain long buffer chains that are not related to high fanout?
- Why are certain cells placed at a physical distance from the rest?
- Why are there high concentrations of buffers on I/O path?
- Why is there high transition or capacitance on pins?

You can query and highlight design objects on the critical path to find answers to these questions that help you understand Design Compiler topographical placement and the problems that can cause timing degradation. For example, you can perform critical path analysis in the layout view to identify the kinds of physical problems that can cause QoR degradation, such as why a long net is on the critical path.



For more information, see the following topic:

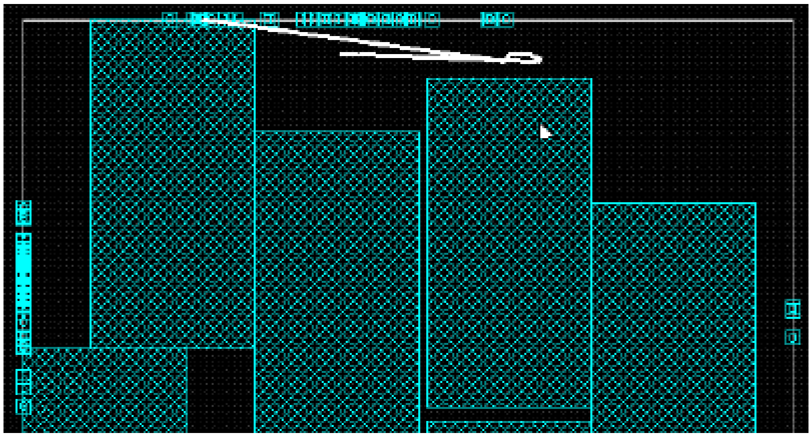
- Viewing the Critical Path

## Viewing the Critical Path

To view the critical path,

1. In the Design Vision window, choose Timing > Timing Analysis Driver.
2. In the timing analysis driver, select the path with the worst negative slack.

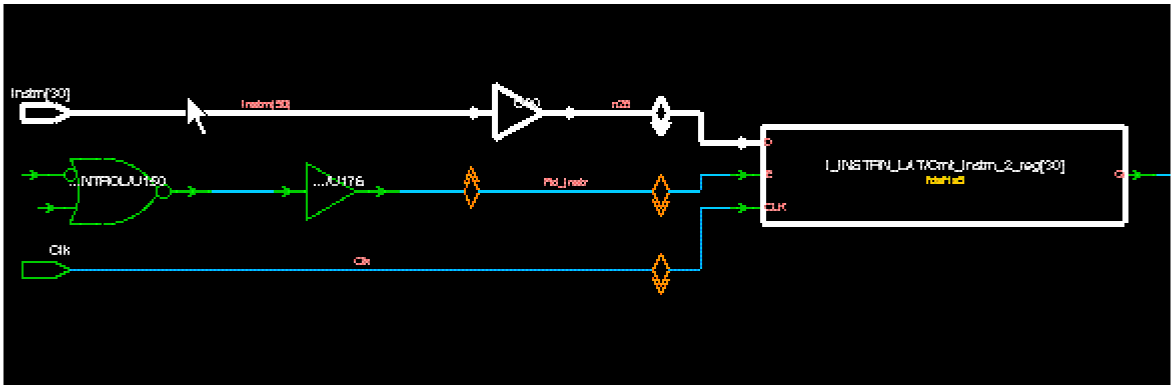
The selected path is automatically cross-selected in the layout view.



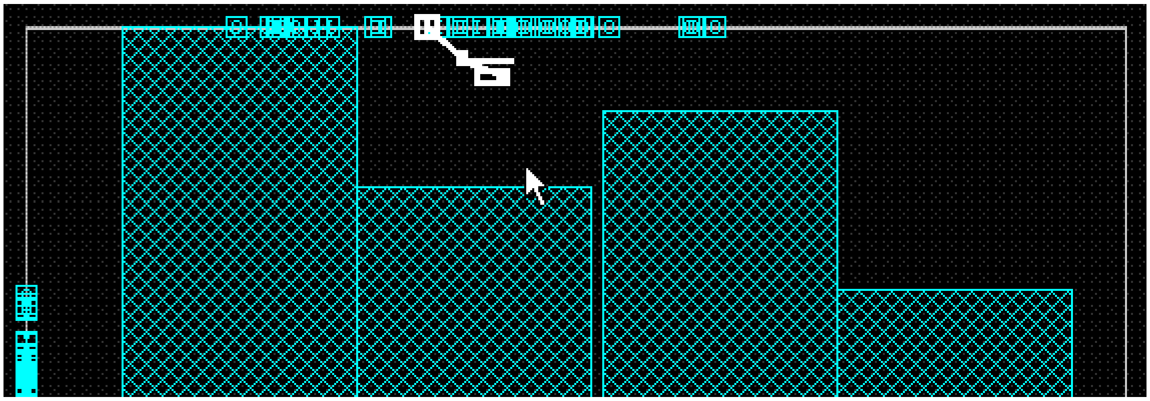
You can analyze the critical path to understand the Design Compiler topographical placement. The following example demonstrates one way to identify why the path startpoint is at a particular location:

1. Select the path startpoint in a schematic view.
2. Choose Select > Fanin/Fanout to open the Select Fanin/Fanout dialog box.
3. Select the Fanin option, set other options as needed, and click OK.
4. Select the startpoint and its input path in the schematic view.

The schematic view displays the startpoint and its input path in the selection color.



The selected logic is automatically cross-selected in the layout view.



**See Also**

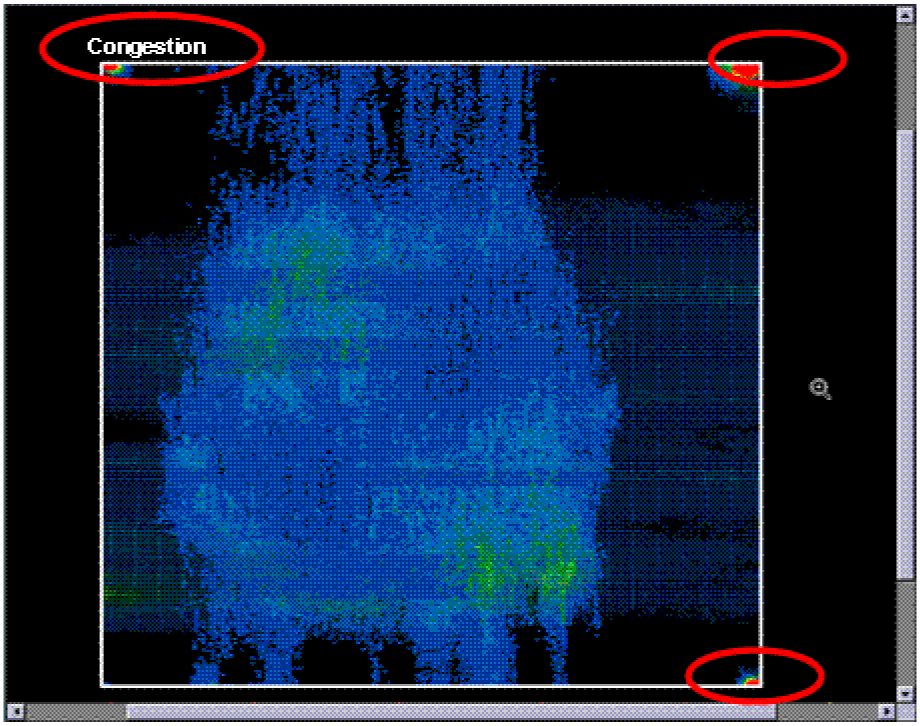
- Debugging QoR Issues Related to the Floorplan and Placement
- The "Examining Timing Path Details" topic in Design Vision Help
- The "Selecting Fanin or Fanout Logic" topic in Design Vision Help

## Visually Analyzing Congestion

You can identify areas of high congestion in your design by viewing the congestion map. By visually examining congested areas in your design, you can determine whether the design is routable and identify the causes of the congestion if the design is not routable. You can display or hide the congestion map at any time when a layout view is open in the Layout window.

Figure 6-1 shows an example of congestion resulting from the layout of the floorplan.

*Figure 6-1 Floorplan Congestion in Layout View*



For information about viewing the congestion in the layout view, see the following sections:

- Displaying the Congestion Map
- Viewing the Congestion Map
- Examining Cells in Congested Areas

**See Also**

- The “Analyzing Congestion” topic in Design Vision Help
- The *Design Compiler User Guide*

  Provides in-depth information about analyzing congestion in Design Compiler Graphical

## Displaying the Congestion Map

To display or hide the congestion map,

- In the Layout window, click the ▨ button on the Analysis toolbar or choose View > Map Mode.

The GUI dims the visible objects in the layout view and displays the Map Mode panel. If you have already generated congestion data during the session, the congestion map grid appears on top of the design in the layout view. If the map does not appear, you must load the congestion data. You can reload the data if it changes during the session.

To load or reload the congestion data,

1. Click the Reload button on the Map Mode panel.
2. Click OK in the dialog box that appears.

The congestion map divides the core area into a grid of colored boxes. Each box represents a vertical plane and a horizontal plane through which routes can pass. The left and bottom box edges are colored and labeled to show the usage-to-capacity ratios of routing tracks through the planes. Each map color represents a range of congestion values called a bin. The ranges are calculated by using a linear interpolation of the congestion data between minimum and maximum thresholds.

### See Also

- The “Analyzing Congestion” topic in Design Vision Help

## Viewing the Congestion Map

You can view map information in the legend on the Map Mode panel, and you can display or hide individual map colors (bins). The legend displays the color, the data count, and optionally, the color exaggeration level for each bin. The colored histogram bars on the right side of the legend represent the relative distribution of congestion values in the bins. You can use the visibility options on the left side of the legend to display or hide map colors for individual bins.

You can control how the tool calculates global route congestion (GRC) by selecting a congestion calculation option.

- To calculate congestion as the sum of the overflow for each layer, select the “Sum of overflow for each layer” option and click Apply.

  This option is selected by default.

- To calculate congestion as the total demand minus the total capacity, select the “Total demand minus total supply” option and click Apply.

For information about these global route congestion (GRC) calculations, see the *Design Compiler User Guide*.

You can facilitate your congestion analysis by

- Displaying or hiding map details map colors in the layout view

- Displaying or hiding map details (box edges or labels)
- Adjusting the congestion ranges

To display or hide map colors in the layout view,

1. Select the visibility options for the colors you want to display, and deselect the visibility options for the colors you want to hide.
2. Click Apply.

To display or hide map details,

1. Set the map display options as needed.
    - ❍ To display or hide the horizontal box edges, select or deselect the Horizontal edges option.
    - ❍ To display or hide the vertical box edges, select or deselect the Vertical option.
    - ❍ To display or hide congestion labels, select or deselect the Text option.
2. (Optional) To display just the congestion in the current design, select the “Current design only” option.

    By default, the congestion map displays congestion in the current design and its subdesigns.
3. Click Apply.

To adjust the congestion ranges,

1. Set the congestion thresholds as needed.
    - ❍ To change the number of congestion ranges (bins), select or type a value in the Bins box.
    - ❍ To change the lower and upper congestion bounds, type values in the From and To boxes.
2. Click Apply.

### See Also

- The “Analyzing Congestion” topic in Design Vision Help

## Examining Cells in Congested Areas

When the congestion map is visible, you can view and select cells in congested areas of the design by using the List by Congested Region dialog box to define a rectangular region in the layout view. The dialog box lists the cells in congested areas within the region. This list includes only the cells in areas with a congestion threshold that is within the range specified on the Map Mode panel and is equal to or greater than 1.

You can select cells in the cell list to view or highlight them in the layout view, and you can filter the cell list by setting a minimum global routing (GRC) congestion threshold. You can also save the cell list in a text file. In addition, you can cross-probe the RTL for cells in congested areas to identify the RTL code that could be causing the congestion.

To select and view cells in a congested region,

1. Click the “List cells in congested region” button on the Map Mode panel.

   The “List by Congested Region” dialog box appears. You can move this dialog box to a location on the screen where you can work with both it and the layout view at the same time.

2. Define the shape and location for the region by doing one of the following:

   - Drag the pointer in the layout view to form the rectangle where you need it.
   - Click the ⬚ button in the “List by Congested Region” dialog box and enter the x- and y-coordinates for the upper left and lower right corners of the rectangle in the Coordinates box.

3. Click Apply.

   The names of the cells in the region appear in the Cell Name list. Only cells in highly congested areas are listed.

4. (Optional) To filter the cells by their global route congestion (GRC) thresholds, enter a value in the GRC Threshold Minimum box.

   Only cells that have a global route threshold value equal to or greater than this value appear in the dialog box.

5. Repeat steps 2 through 4 if you need to list cells in a different area of the design.

The List by Congested Region dialog box displays the cell list in either a tree view or a list view. The tree view appears by default. You select the view in the list box at the bottom of the dialog box. Use the tree view to cross-probe congested areas. Use the list view to examine the cells in the congested areas.

- The tree view displays rows for the RTL file names, line numbers, and cell instance names and columns for the RTL line number, maximum global route congestion (GRC) value, RTL origin, and cell reference name.

You can expand a file name or line number by double-clicking the name or number or by clicking the expansion button (plus sign).

- The list view displays a row for each cell and columns for the cell instance name, cell reference name, cell path, dont_touch attribute value, is_mapped attribute value, cell_library attribute value, RTL file name, RTL origin (such as RTL, DATA_PATH, DFT, CLKGT, and so forth), and RTL line number.

To select and view congested cells in the layout view,

- Select the cell names in the cell list in the List by Congested Region dialog box.

  The layout view displays the selected cells in the selection color, which is white by default.

When you select a cell name, the List by Congested Region dialog box displays the RTL for the cell in an RTL text view below the cell list. You can also view the RTL for one or more cells in the RTL browser by selecting the cell names and clicking the Cross Probe button. Cross-probing cells in highly congested areas can help you to identify the RTL code that is causing the congestion. For more information, see Cross-Probing Cells in Congested Areas.

You can also save the cell list data in a file. For more information, see Saving the Cell List.

**See Also**

- Visually Analyzing Congestion
- The “Viewing Cells in Congested Areas” topic in Design Vision Help

## Cross-Probing Cells in Congested Areas

During congestion analysis, cross-probing cells in highly congested areas can help you to identify the RTL code that is causing the congestion. When you cross-probe cells in a congested area, you can view the RTL in the List by Congested Region dialog box or the RTL browser.

To quickly view the RTL for a cell in the List by Congested Region dialog box,

- Select the cell name in the List by Congested Region dialog box.

  You can Shift-click or Ctrl-click to select multiple cell names.

Note:
  To select a cell name in the tree view, you might need to expand its file name and line number by clicking their expansion buttons.

The first time you select a cell name, the tool locates the RTL files in which the cells originate, opens a new pane below the cell list in the List by Congested Region dialog box, and displays the RTL for the selected cell in an RTL text view.

Each time you select a cell name from a different file, a tab with the file name appears above the RTL text view. You can click this tab to return to the file after viewing other files. For information about working in the RTL text view, see the “Viewing RTL Files” topic in Design Vision Help.

To close a file in the RTL text view,

- Right-click its tab and choose Close.

To cross-probe cells in congested areas and view their RTL in the RTL browser,

1. Select one or more cells in the List by Congested Region dialog box.
2. Click the Cross Probe button, or choose AnalyzeRTL > Cross Probe to Source.

The tool locates the RTL files in which the cells originate and opens the files in a new RTL browser window.

### See Also

- Examining Cells in Congested Areas
- The “Cross-Probing Cells in Congested Areas” topic in Design Vision Help

## Saving the Cell List

To save the cell list data in a file,

1. Click the Save List As button.

   The Save Cell List As dialog box appears.
2. Select a file or type a file name in the “File name” box.
3. Click Save.

The tool saves the cell list data in a text file with a row for each cell and the column data delimited by commas.

### See Also

- Examining Cells in Congested Areas

# Index