

# **Command-Line IP Selector (CLIPS) User Guide**

**Product Version ICADVM20.1  
October 2020**

© 2020 Cadence Design Systems, Inc. All rights reserved.

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

**Trademarks:** Trademarks and service marks of Cadence Design Systems, Inc. contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522. All other trademarks marks are the property of their respective owners.

**Restricted Permission:** This publication is protected by copyright law and international treaties and contains trade secrets and proprietary information owned by Cadence. Unauthorized reproduction or distribution of this publication, or any portion of it, may result in civil and criminal penalties. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. Unless otherwise agreed to by Cadence in writing, this statement grants Cadence customers permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used only in accordance with a written agreement between Cadence and its customer.
2. The publication may not be modified in any way.
3. Any authorized copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement.
4. The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration.

**Disclaimer:** Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information.

**Restricted Rights:** Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor

---

# Contents

---

<b><u>Preface</u></b> .....	5
<b><u>Scope</u></b> .....	6
<b><u>Licensing Requirements</u></b> .....	6
<b><u>Related Documentation</u></b> .....	6
<u>What's New and KPNS</u> .....	6
<u>Installation, Environment, and Infrastructure</u> .....	6
<b><u>Additional Learning Resources</u></b> .....	7
<u>Video Library</u> .....	7
<u>Virtuoso Videos Book</u> .....	7
<u>Rapid Adoption Kits</u> .....	7
<u>Help and Support Facilities</u> .....	7
<b><u>Customer Support</u></b> .....	8
<b><u>Feedback about Documentation</u></b> .....	8
<b><u>Typographic and Syntax Conventions</u></b> .....	9
<b><u>1</u></b>	
<b><u>Introduction to Command-Line IP Selector</u></b> .....	11
<u>Benefits of Using CLIPS</u> .....	11
<u>The CLIPS Flow</u> .....	11
<b><u>2</u></b>	
<b><u>Getting Started with CLIPS</u></b> .....	13
<u>Prerequisite</u> .....	14
<u>Launching CLIPS</u> .....	14
<u>Running CLIPS in GUI Mode</u> .....	14
<u>The CLIPS GUI Components</u> .....	16
<u>Saving and Loading the CLIPS Setup</u> .....	22
<u>Running CLIPS in Command-Line Mode</u> .....	23

### 3

<u>Working with CLIPS</u>	27
<u>CLIPS workflow</u>	28
<u>Managing xrun Files</u>	28
<u>Managing Virtuoso Directories</u>	30
<u>Managing Instance Hierarchies and Bindings</u>	32
<u>Controlling the Hierarchy View</u>	33
<u>Overriding or Switching Instance Bindings</u>	35
<u>Configuring Settings and Options</u>	42
<u>Creating Groups</u>	43
<u>Applying Settings to Specific Groups</u>	46
<u>Configuring Global Options</u>	48
<u>Netlisting the Virtuoso Config or Schematic Views</u>	49
<u>Performing a Binding Check</u>	49
<u>Viewing the Netlist Output</u>	51
<u>Accessing Logs</u>	53

### 4

<u>Understanding the Difference Between CLIPS and Unified Netlisting (UNL)</u>	55
--	----

### 5

<u>CLIPS Environment Variables</u>	57
.....	58

---

# Preface

---

This user guide describes how to use Command-Line IP Selector (CLIPS) to make Virtuoso<sup>®</sup> mixed-signal IP reuse easier. It also explains how the netlisting procedure in CLIPS is different from the netlisting in Analog Design Environment (ADE) with Unified Netlister (UNL).

This document is intended for developers and designers of integrated circuits and assumes that you are familiar with the application infrastructure mechanisms designed to support consistent operations between all the Cadence<sup>®</sup> tools.

The preface discusses the following:

- [Scope](#)
- [Licensing Requirements](#)
- [Related Documentation](#)
- [Additional Learning Resources](#)
- [Customer Support](#)
- [Feedback about Documentation](#)
- [Typographic and Syntax Conventions](#)

## Scope

Unless otherwise noted, the functionality described in this guide can be used in both the mature node (for example, IC6.1.8) and advanced node and methodologies (for example, ICADVM20.1) releases.

Label	Meaning
(ICADVM20.1 Only)	Features supported only in the ICADVM20.1 advanced nodes and advanced methodologies release.
(IC6.1.8 Only)	Features supported only in the mature node releases.

## Licensing Requirements

You require the 70000, the AMS\_environment license, to run CLIPS.

**Note:** If you are using an ADE state for netlisting, you would also require a license of the ADE product in which that state was saved.

For more details, refer to the [Virtuoso Software Licensing and Configuration User Guide](#)

## Related Documentation

### What's New and KPNS

- [Command-Line IP Selector What's New](#)
- [Command-Line IP Selector KPNS](#)

### Installation, Environment, and Infrastructure

- [Cadence Installation Guide](#)
- [Cadence Application Infrastructure User Guide](#)

## Additional Learning Resources

### Video Library

The [Video Library](#) on the Cadence Online Support website provides a comprehensive list of videos on various Cadence products.

To view a list of videos related to a specific product, you can use the *Filter Results* feature available in the pane on the left. For example, click the *Virtuoso Layout Suite* product link to view a list of the videos available for the product.

You can also save your product preferences in the Product Selection form, which opens when you click the *Edit* icon located next to *My Products*.

### Virtuoso Videos Book

You can access certain videos directly from Cadence Help. To learn more about this feature and to access the list of available videos, see [Virtuoso Videos](#).

### Rapid Adoption Kits

Cadence provides a number of [Rapid Adoption Kits](#) that demonstrate how to use Virtuoso applications in your design flows. These kits contain design databases and instructions on how to run the design flow.

To explore the full range of training courses provided by Cadence in your region, visit [Cadence Training](#), or write to [training\\_enroll@cadence.com](mailto:training_enroll@cadence.com).

**Note:** The links in this section open in a separate web browser window when clicked in Cadence Help.

### Help and Support Facilities

Virtuoso offers several built-in features to let you access help and support directly from the software.

- The Virtuoso *Help* menu provides consistent access to help across Virtuoso tools and applications. The standard Virtuoso *Help* menu lets you access the most useful help and support resources from the Cadence support and corporate websites directly from the CIW or any Virtuoso application.

- The Virtuoso Welcome Page is a self-help launch pad offering access to a host of useful knowledge resources, including quick links to content available within the Virtuoso installation as well as to other popular online content.

The Welcome Page is displayed by default when you open Cadence Help in standalone mode from a Virtuoso installation. You can also access it at any time by selecting *Help – Virtuoso Documentation Library* from any application window, or by clicking the *Home* button on the Cadence Help toolbar (provided you have not set a custom home page).

For more information, see [Getting Help](#) in *Virtuoso Design Environment User Guide*.

## Customer Support

For assistance with Cadence products:

- Contact Cadence Customer Support

Cadence is committed to keeping your design teams productive by providing answers to technical questions and any queries about the latest software updates and training needs. For more information, visit <https://www.cadence.com/support>.

- Log on to Cadence Online Support

Customers with a maintenance contract with Cadence can obtain the latest information about various tools at <https://support.cadence.com>.

## Feedback about Documentation

You can contact Cadence Customer Support to open a service request if you:

- Find erroneous information in a product manual
- Cannot find the information you are looking for in a product manual
- Face an issue while accessing documentation by using Cadence Help

You can also submit feedback by using the following methods:

- In the Cadence Help window, click the *Feedback* button and follow instructions.
- On the Cadence Online Support [Product Manuals](#) page, select the required product and submit your feedback by using the *Provide Feedback* box.



## Typographic and Syntax Conventions

The following typographic and syntax conventions are used in this manual.

<i>text</i>	Indicates names of manuals, menu commands, buttons, and fields.
text	Indicates text that you must type as presented. Typically used to denote command, function, routine, or argument names that must be typed literally.
<i>z_argument</i>	Indicates text that you must replace with an appropriate argument value. The prefix (in this example, <i>z_</i> ) indicates the data type the argument can accept and must not be typed.
	Separates a choice of options.
{ }	Encloses a list of choices, separated by vertical bars, from which you <b>must</b> choose one.
[ ]	Encloses an optional argument or a list of choices separated by vertical bars, from which you <b>may</b> choose one.
[ ?argName t_arg ]	Denotes a <i>key argument</i> . The question mark and argument name must be typed as they appear in the syntax and must be followed by the required value for that argument.
...	Indicates that you can repeat the previous argument.
	Used with brackets to indicate that you can specify zero or more arguments.
	Used without brackets to indicate that you must specify at least one argument.
, ...	Indicates that multiple arguments must be separated by commas.
=>	Indicates the values returned by a Cadence® SKILL® language function.
/	Separates the values that can be returned by a Cadence SKILL language function.

If a command-line or SKILL expression is too long to fit within the paragraph margins of this document, the remainder of the expression is moved to the next line and indented. In code excerpts, a backslash ( \ ) indicates that the current line continues on to the next line.

# **Command-Line IP Selector (CLIPS) User Guide**

## Preface

---

---

# Introduction to Command-Line IP Selector

---

In an advanced mixed-signal design flow, a mixed-signal block is commonly used as a part of the System on Chip (SoC) simulation with the Digital on Top (DoT) methodology. Spectre AMS Designer, which is a powerful tool used to simulate mixed-signal designs, uses a config or schematic view from the Virtuoso database to manage a complex binding configuration in a mixed-signal block. However, when this mixed-signal block or mixed-signal IP is reused in the DoT flow, manual export and reconfiguration of these blocks become challenging for a designer. Command-Line IP Selector (CLIPS) is a utility that simplifies the Virtuoso mixed-signal IP reuse.

## Benefits of Using CLIPS

CLIPS provides a bridge between Virtuoso, which is a UI-based, analog, and mixed-signal design environment, and other command-line, digital, and mixed-signal simulation tools and flows that use text-based tests. The design verification team can use CLIPS to verify the digital representation of an IP with their corresponding analog representation. CLIPS has several benefits for mixed-signal designers:

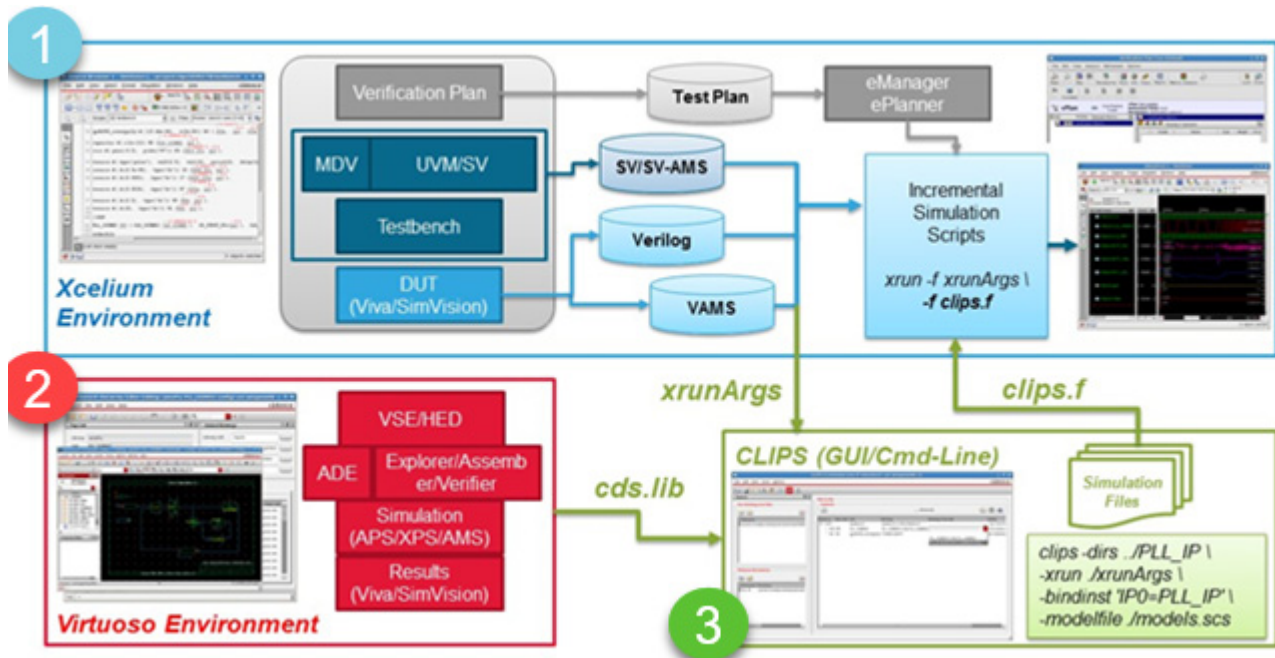
- Leverages an existing testbench setup
- Provides a powerful digital verification mechanism
- Eliminates the need to import large and complex digital designs in Virtuoso
- Provides an integrated design management system

## The CLIPS Flow

The illustration given below shows how CLIPS works with Virtuoso and Xcelium:

# Command-Line IP Selector (CLIPS) User Guide

## Introduction to Command-Line IP Selector



1. In Xcelium, a high-level simulation and verification is run after the top-down flow sets up the SoC-level verification methodology by using Metric-Driven Verification (MDV) with Universal Verification Methodology (UVM) or SystemVerilog (SV).
2. In Virtuoso Analog Design Environment (ADE), a bottom-up mixed-signal IP is designed and verified.
3. CLIPS bridges the gap between the top-down flow of the Xcelium environment and the bottom-up flow of the Virtuoso environment. When you import the Xcelium simulation setup and the Virtuoso AMS IP configuration in CLIPS with the help of the `xrunArgs` files and the `cds.lib` files respectively, it automatically generates netlist and packages the IP config into an independent directory, and generates an incremental file, `clips.f`, on top of the existing `xrunArgs` file. When both these files, `xrunArgs` and `clips.f`, are added to the `xrun` command, the AMS IP automatically replaces its digital counterpart in the SoC simulation setup.

---

## Getting Started with CLIPS

---

This chapter provides the information required to start and set up CLIPS. It includes the following topics:

- Prerequisite
- Launching CLIPS
  - Running CLIPS in GUI Mode
  - The CLIPS GUI Components
  - Running CLIPS in Command-Line Mode

To know about the features and functionality of CLIPS, see the chapter "Introduction to Command-Line IP Selector".

## Prerequisite

Ensure that the path of Xcelium 17.10 is set and added to the PATH environment variable.

## Launching CLIPS

A CLIPS session can be launched in two modes:

- GUI Mode
- Command-line Mode

### Running CLIPS in GUI Mode

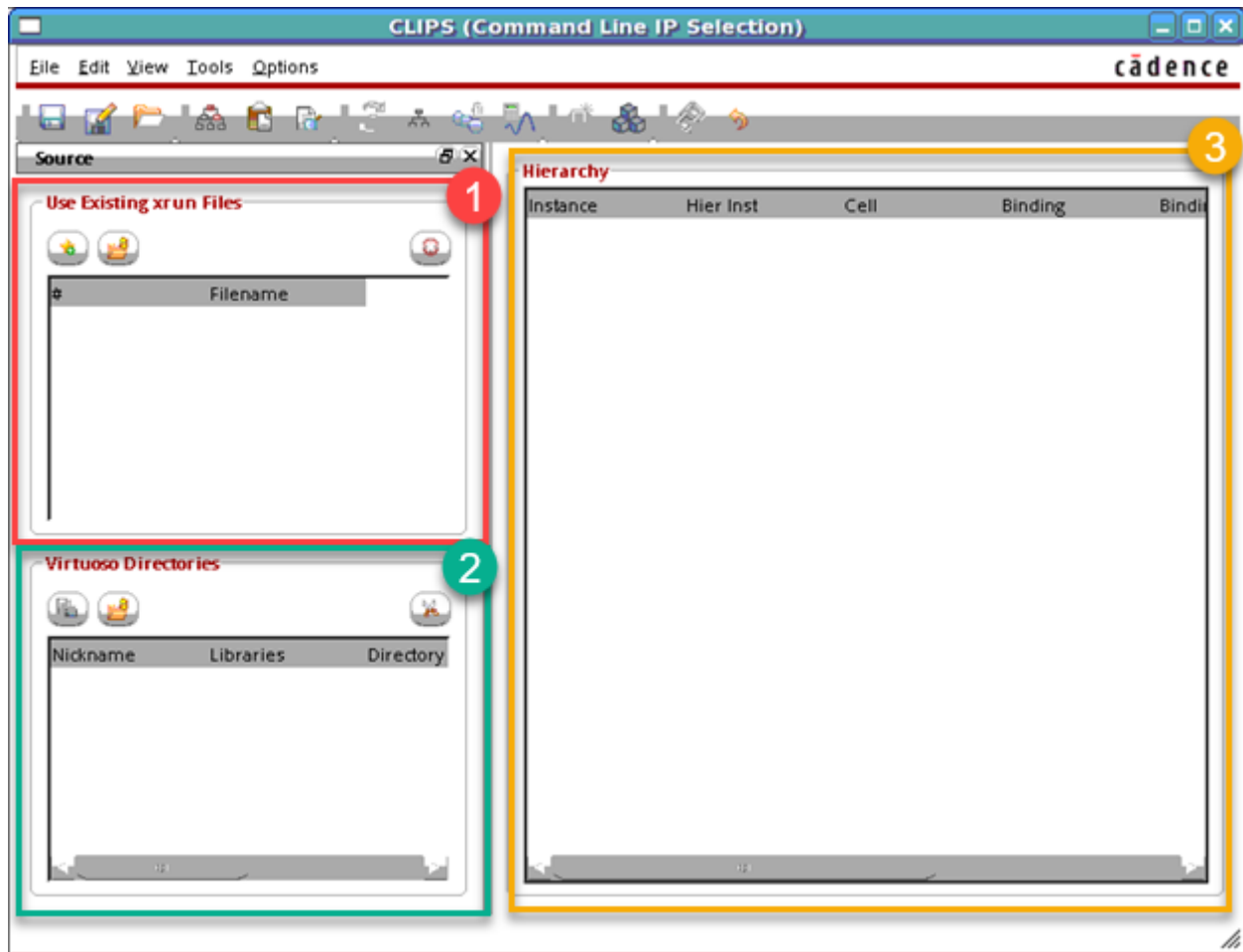
Use the following command to launch CLIPS in GUI mode:

```
clips &
```




# Command-Line IP Selector (CLIPS) User Guide

## Getting Started with CLIPS

The CLIPS window is displayed. It includes three sections, as shown in the figure given below:



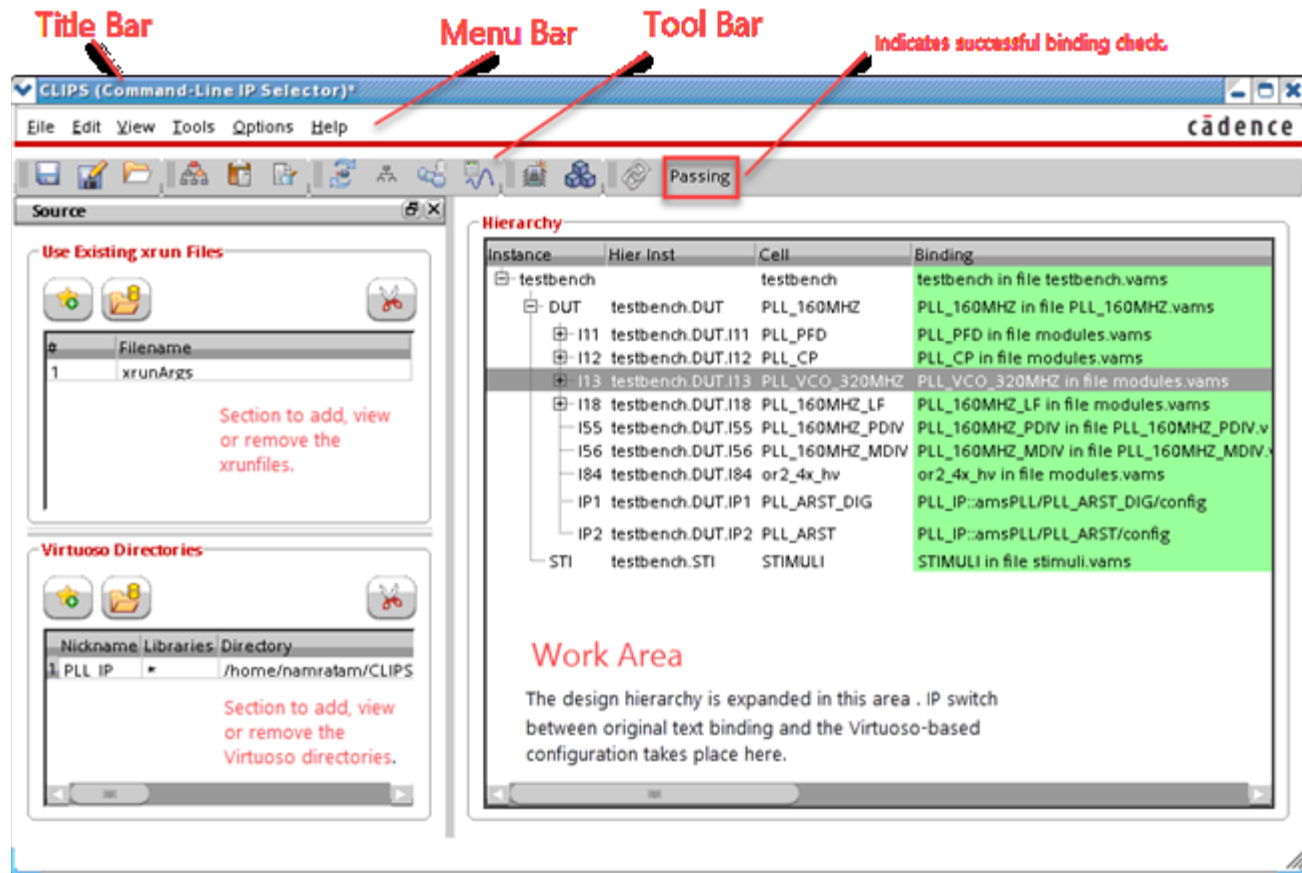
The table given below describes the sections available in the CLIPS user interface :

Section	Used for
 <i>Use Existing xrun Files</i>	<u>Managing xrun Files</u>
 <i>Virtuoso Directories</i>	<u>Managing Virtuoso Directories</u>
 <i>Hierarchy</i>	<u>Managing Instance Hierarchies and Bindings</u>

# Command-Line IP Selector (CLIPS) User Guide

## Getting Started with CLIPS

### The CLIPS GUI Components



The following tables describe the main form components.

- Toolbar
- Menu bar

### Toolbar

#### Icon



#### Description

Saves the current CLIPS setup at the current location and with the existing name.



# Command-Line IP Selector (CLIPS) User Guide

## Getting Started with CLIPS

---



Saves the current CLIPS setup to a file with suffix `.clips`, with a specific name and at a specific location. This file can be loaded later.



Loads the `.clips` file, which contains a state or settings.



Shows or hides the source section for including the xrun files and Virtuoso directories.



Shows or hides the Settings and Options section.



Shows or hides the Status log section.



Elaborates the xrun files. For any IP switch and binding override changes, it updates and descends through complete instance hierarchy and bindings.



Expands the hierarchy tree to show which instances have switchable binding overrides. Trees with no available binding overrides are collapsed.



Shows the advanced hierarchy tree control panel on top of the hierarchy window. You can search instance or cell or binding in the hierarchy and control expand level.



Creates the netlist of Virtuoso hierarchy bindings in the output directory.



Opens the default or user-defined output directory to view the netlisted Virtuoso cellviews and the related xrun files created/exported by CLIPS.



Checks the binding overrides by elaborating the original xrun files, the netlisted Virtuoso cellview bindings, and the CLIPS-generated xrun files.

## Menu bar

### Menu

#### *File*

### Types of options

Access the following options:

- Create, Save and Load a CLIPS setup.
- Exit the tool.

## Command-Line IP Selector (CLIPS) User Guide

### Getting Started with CLIPS

---

#### *Edit*

Access the following options:

- Clear the log file.
- Specify a directory to be used as the netlist output directory.

#### *View*

Access the following options:

- Set your preferred general options to show or hide xrun files and Virtuoso setup directories. You can also show or hide the settings and options pane, the status log pane, and the advanced hierarchy tree.
- Show the output directory where the netlisted Virtuoso cellviews and the related xrun files created or exported by CLIPS are saved.
- Show the netlisted Virtuoso cellviews.

#### *Tools*

Access the following options:

- Create a netlist for Virtuoso hierarchy bindings in the output directory.
- Check the binding overrides by elaborating the original xrun files, the netlisted Virtuoso cellview bindings, and the CLIPS-generated xrun files.
- Add, update or remove xrun files.
- Add a Virtuoso session (directory with cds.lib).
- Set up the model files.
- Remove the manually added binding overrides from all the instances in the hierarchy.
- Refresh the hierarchy to elaborate xrun files, if changed.
- Expand the hierarchy tree to show the instances that have switchable binding overrides.

# Command-Line IP Selector (CLIPS) User Guide

## Getting Started with CLIPS

---

### *Options*

Access the following options:

- Control whether the model files will be copied or linked to the netlist output directory.
- Cache the available views in the library cellviews.
- Hide the *Hierarchy instance*, *Binding override*, and *Notes* column in the hierarchy tree.
- Show details of file parsing and hierarchy processing in the log file.
- Add or remove groups of instance related settings.
- Enable the group setting panel. If unchecked, only global setting panel is visible.
- Automatically run the elaboration/binding check after creating the netlist.

### *Help*

Access help on using CLIPS and other useful resources.

### ***Help***

The options in the Help menu are described below.

<b>Help Menu options</b>	<b>Description</b>
<i>Search</i>	A text field that lets you enter a search string. Press <code>Enter</code> to view the search results.  <b>Note:</b> Do not enclose the search string in double quotes.
<i>User Guide</i>	Opens CLIPS User Guide (at the section that provides information about using CIW) in Cadence Help.
<i>What's New</i>	Opens the CLIPS What's New document in Cadence Help.
<i>Known Problems and Solutions</i>	Opens the Virtuoso Known Problems and Solutions document in Cadence Help.

## Command-Line IP Selector (CLIPS) User Guide

### Getting Started with CLIPS

---

<i>Virtuoso Documentation Library</i>	<p>Opens the Cadence Help home page, which provides quick access links to the following local and online resources:</p> <ul style="list-style-type: none"><li>What's New</li><li>Video Demos and Tutorials</li><li>Featured Content</li><li>Known Problems and Solutions</li><li>Other web resources</li></ul>
<i>Virtuoso Video Library</i>	<p>Opens the Video Library page available on Cadence Online Support (COS). This page lists the videos available for various Virtuoso products.</p> <p>You must have a COS account to access the content available on COS.</p> <p>Contact your IT support to ensure that the Internet ports required for video playback are enabled.</p>
<i>Virtuoso Rapid Adoption Kits</i>	<p>Opens the Rapid Adoption Kits page on COS. This page lists the Rapid Adoption Kits (RAKs) available for various Virtuoso products. A RAK contains design databases and instructions on how to run the design flow.</p>
<i>Virtuoso Learning Map</i>	<p>Lists the domain-specific training available on Cadence Training Services.</p> <p>Cadence Training Services learning maps provide a comprehensive visual overview of the learning opportunities for Cadence customers. They provide recommended course flows as well as tool experience and knowledge levels to guide customers through a complete learning plan.</p>
<i>Virtuoso Custom IC Community</i>	<p>Opens the Virtuoso Custom IC Community web page. This page provides access to the latest blogs and discussion threads on various Virtuoso products and design topics, information about software downloads and support and training, and other related information. You too can contribute to the community forum by creating a Cadence account. This gives you additional benefits such as alerts about topics of interest and access to online webinars.</p>

## Command-Line IP Selector (CLIPS) User Guide

### Getting Started with CLIPS

---

<i>Cadence Online Support</i>	<p>Opens COS, which you can use to access information about Cadence products, documentation, videos, RAKs, application notes, troubleshooting information, alerts, and so on. Improvements are regularly made to COS to provide the best and latest information. We recommend that you bookmark this web site and use it as your first point of reference for any Virtuoso-related information.</p> <p>You can also access COS by clicking the Cadence logo on the upper-right banner in each Virtuoso window.</p>
<i>Cadence Training</i>	<p>Opens the Cadence training web page. You can find on this page information about the training courses available in different regions. Information is available about both classroom and online courses.</p>
<i>Cadence Community</i>	<p>Opens the Cadence Community web page. This page provides access to the latest blogs and discussion threads on various Cadence products and solutions, and EDA Industry Insights. You too can contribute to the community forum by creating a Cadence account. This gives you additional benefits such as alerts about topics of interest and access to online webinars.</p>
<i>Cadence OS Platform Support</i>	<p>Provides information about the current Cadence software releases and the supported platforms.</p>
<i>Contact Us</i>	<p>Opens the Cadence Customer Support web page, which provides the customer support contact information for different regions.</p>
<i>Cadence Home</i>	<p>Opens the Cadence corporate website.</p>
<i>About CLIPS</i>	<p>Displays the version information for CLIPS.</p>

## Saving and Loading the CLIPS Setup

To save the current CLIPS setup in GUI mode:

1. Select *File – Save As*.

The CLIPS Save Filename form appears.

2. Specify a name for the clips setup and click *OK*.

The setup is saved in a `.clips` file.

To load an existing CLIPS setup in the UI mode:

1. Select *File – Load*.

The CLIPS Load Filename form appears.

2. Select the `.clips` file from which you want to load the setup, and click *OK*.

Alternatively, to load an existing CLIPS setup in the command-line mode, run the following command:

```
clips -load <xxx.clips file>
```

## Running CLIPS in Command-Line Mode

Use the following command to launch CLIPS in the command-line mode:

```
clips <optional command-line arguments>
```

A few examples are given below.

```
clips -load myTextPll.clips
clips -dirs ../PLL_IP -xrun xrunArgs &
clips -load myTextPll.clips -batch -log myBatchRun.log
clips -dirs ../PLL_IP -xrun xrunArgs -switch "inst=testbench.IP0.I11.I14
config=PLL_IP::amsPLL/dffnr_2x_hv/config ade=PLL_IP::amsPLL/pll_top/
maestro:pll_top:1" -switch "inst=testbench.IP0.I11.I13 config=PLL_IP::amsPLL/
nor2_2x_nv/config ade=PLL_IP::amsPLL/pll_top/ams_state" -batch
clips -export "config=amsPLL/dffnr_2x_hv/config ade=PLL_IP::amsPLL/pll_top/
ams_state outdir=clips_export_dffnr cdsdir=../test/IP_dir" -batch
```

The command-line arguments can be saved in a text file and used while launching CLIPS with the help of the following command:

```
clips -f <command-line arguments file>
```

## Introducing Command-line Arguments to Reuse Testbenches and IP

You can use the command-line option `-export` to export an IP block from a remote Virtuoso config or schematic view, without top-level SoC design access to a package directory. Use the command-line option `-switch` to import multiple packages into an SoC top-level design to assemble them together.

CLIPS makes the export and switch IP procedures seamless because it manages the files and design hierarchy paths along with maintaining the relationship between them.

These options are supported in batch mode. Use the following command in the command-line mode:

### ■ Export (for IP provider):

```
clips -batch
-export "config=<lib/cell/view>
ade=<stateLib/stateCell/stateView[/testName]>
outdir=<ip_export_dir>
[cdsdir=<dir where cds.lib exists>]"
```

### ■ Switch (for IP user):

```
clips -batch
-xrun <top soc xrunArgs file>\
-output <output directory after import>\
-switch "inst=<inst hierarchy in top design> import=<ip1_export_dir>"\
-switch "inst=<inst hierarchy in top design> import=<ip2_export_dir>"
```

# Command-Line IP Selector (CLIPS) User Guide

## Getting Started with CLIPS

### Other Command-line Arguments

Depending on your requirements, you can use one or more of the following optional command-line arguments:

Command-line Argument	Purpose
<code>-load &lt;file&gt;</code>	Loads the setup from previously saved CLIPS file.
<code>-batch</code>	Runs (netlists) a loaded CLIPS file and exits with status (0==pass, 1==fail).
<code>-netlist</code>	Runs (netlists) a loaded CLIPS file on startup before interactive use.
<code>-local</code>	Automatically connects to Virtuoso using a local cds.lib when started without loading.
<code>-continue</code>	Continues after load/setup error, if not in batch. Default behavior is to exit, which does not allow any changes.
<code>-postrun</code>	Enables running of any batch, post netlisting run commands.
<code>-xrun &lt;file&gt; [,&lt;file&gt;]</code>	Populates the hierarchy from one or more existing <code>xrun -f</code> files. Runs and analyzes the created .pak file.
<code>-switch &lt;value&gt;</code>	Switches the settings of specified instances to an HED configuration. For multiple instances, add an option and value for each instance.
<code>-export &lt;value&gt;</code>	Exports an IP remotely to assemble with a test bench in AMS UNL. It supports both, ADE and Maestro states. For multiple instances, add an option and value for each instance.
<code>-savebatch &lt;file&gt;</code>	When in batch mode, saves the setup to <file> before running netlisting. It includes all command-line changes.
<code>-log &lt;file&gt;</code>	Sets the status log file. The Default value is <code>CLIPS.log</code> . Set as <code>no</code> to disable log write.
<code>-dirs &lt;dirs&gt;</code>	Loads Virtuoso directories: <code>&lt;dir&gt;[:&lt;nickname&gt;][,&lt;dir&gt;[:&lt;nickname&gt;]]...</code>
<code>-details &lt;value&gt;</code>	Shows parsing and hierarchy processing in the log area / file. Default values are <code>yes</code> or <code>no</code> .
<code>-checkbind &lt;value&gt;</code>	Automatically runs elaboration/binding check after netlisting, when CLIPS is run in batch mode. Default values are <code>yes</code> or <code>no</code> .



## Command-Line IP Selector (CLIPS) User Guide

### Getting Started with CLIPS

Command-line Argument	Purpose
<code>-depth &lt;depth&gt;</code>	Sets the depth of hierarchy to show. 0 means all levels.
<code>-modelFile &lt;value&gt;</code>	Used for netlisting configs. Its syntax: <code>&lt;model1[:model2(sectionName)]&gt;</code> . Its default value is the value in the environment variable <code>\$CLIPS_MODELFILE</code> .
<code>-cdsPre &lt;value&gt;</code>	Sets the full path to a UNIX script file to set up Virtuoso-specific environment variables. This C Shell script is sourced before Virtuoso sessions are started. Enter <code>setenv CLIPS_VIRTUOSO_BOURNE_SHELL</code> to use the Bourne shell format.  <b>Note:</b> This is the command-line equivalent of the field <i>Virtuoso Setup Script</i> .
<code>-output &lt;value&gt;</code>	Netlists and exports the hierarchy to a new <code>CLIPSOUTPUT</code> (or <code>-subDir &lt;name&gt;</code> ) under this directory (this directory must exist). It defaults to <code>'.'</code>
<code>-subDir &lt;value&gt;</code>	Creates subdirectory under <code>-output &lt;directory&gt;</code> during netlisting. We can include <code>&lt;date&gt;</code> or <code>\$envvarnam</code> , which will be processed when the directory is created.
<code>-hideCell &lt;value&gt;</code>	Shows the list of cell names for which instances will be hidden in the hierarchy. Its default value is the value of the environment variable <code>\$CLIPS_HIDE_CELLS</code> .
<code>-preRun &lt;value&gt;</code>	It is a UNIX command to run before running netlisting, when in batch mode, if <code>-postrun</code> is set. If it returns a non-zero value then netlisting will not run.
<code>-postRun &lt;value&gt;</code>	It is a UNIX command to always run (pass or fail) after running netlisting, when in batch mode, if <code>-postrun</code> is set. Example, <code>mv ./CLIPS_OUTPUT ~/simFiles</code>
<code>-postRunPass &lt;value&gt;</code>	It is a UNIX command to run after running netlisting when in batch mode, if <code>-postrun</code> is set. It is used only if netlisting has passed. Example, <code>xrun -f xrunArgs -f ./CLIPS_OUTPUT/clips.f</code>
<code>-postRunFail &lt;value&gt;</code>	It is a UNIX command to run after running netlisting, when in batch mode. It is used if netlisting has failed. Example, <code>sendFailMail projectLeaders &lt;logfile&gt;</code>

You can also use the `clips -help` command to view the help information.

# **Command-Line IP Selector (CLIPS) User Guide**

## Getting Started with CLIPS

---

---

## Working with CLIPS

---

This chapter includes the following topics that will help you get started with CLIPS:

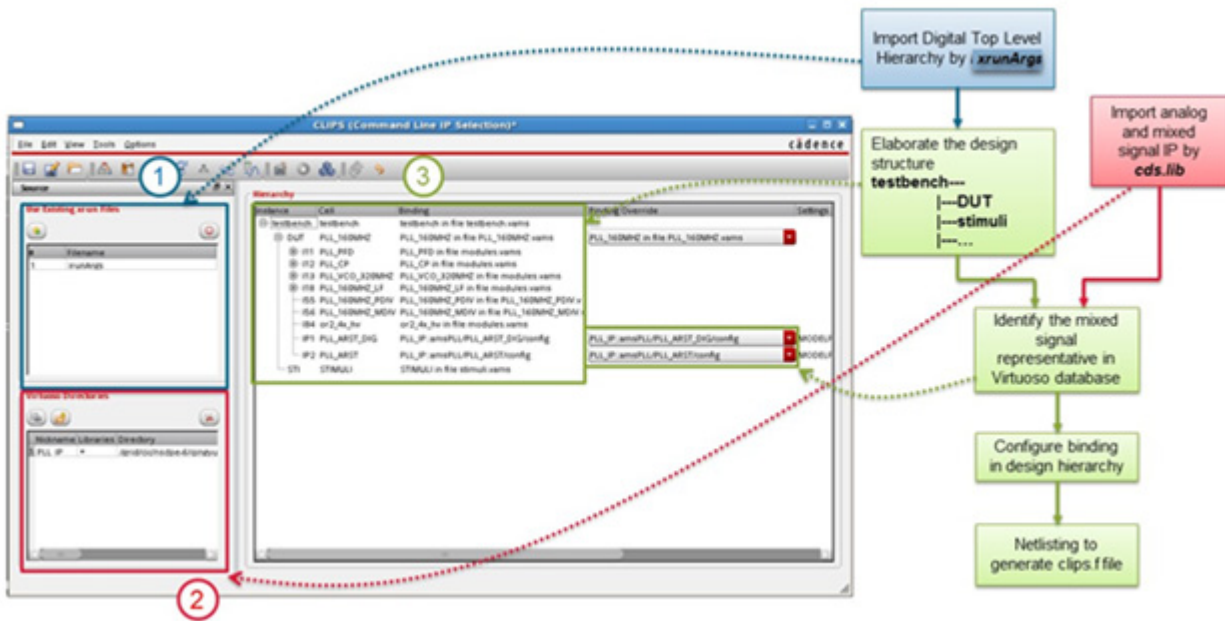
- [CLIPS workflow](#)
- [Managing xrun Files](#)
- [Managing Virtuoso Directories](#)
- [Managing Instance Hierarchies and Bindings](#)
- [Configuring Settings and Options](#)
- [Netlisting the Virtuoso Config or Schematic Views](#)
- [Performing a Binding Check](#)
- [Viewing the Netlist Output](#)
- [Accessing Logs](#)

# Command-Line IP Selector (CLIPS) User Guide

## Working with CLIPS

### CLIPS workflow

The following figure explains the sequence of steps to be followed in CLIPS.



### Managing xrun Files

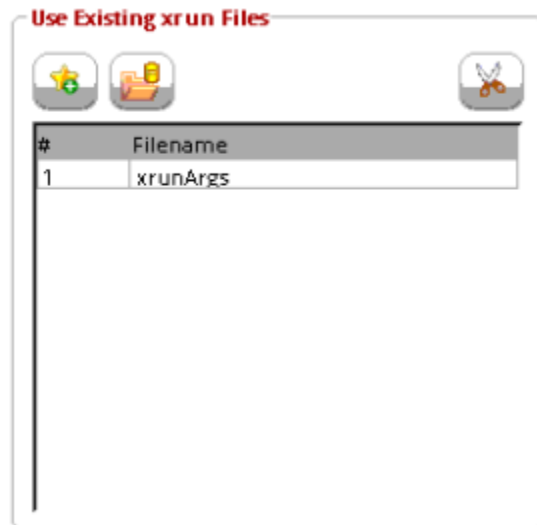
CLIPS internally executes the xrun command to run simulations using Xcelium. You can specify the design files, input files and the command-line options in the argument files that are used by the xrun command. You can manage these xrun argument files in the 'Use

## Command-Line IP Selector (CLIPS) User Guide




### Working with CLIPS


---

Existing `xrun` Files' section of the CLIPS window. The following figure shows the Use Existing `xrun` Files section, which can be used to manage the `xrun` files in CLIPS:



The table given below describes the command buttons available in this section:

Command	Description
<i>Add new <code>xrun</code> file</i> 	Opens the CLIPS Add <code>xrun -f</code> Files form that you can use to select the <code>xrun</code> files from where you need to import the SoC simulation setup. The selected files are added to the table in this section.
<i>View <code>xrun</code> file</i> 	Opens the selected <code>xrun</code> file to show its content.
<i>Delete <code>xrun</code> file</i> 	Deletes the selected <code>xrun</code> script file.

When you add or remove an `xrun` script file, CLIPS prompts you to update the hierarchy. Click **Update**  on the toolbar to update the hierarchy. An elaboration runs in the background, if needed, to generate the latest hierarchy structure of the SoC simulation setup. The **Update** command also checks the time stamp of the existing `xrunArgs` files and re-runs the elaboration.



## Managing Virtuoso Directories

If you have used Virtuoso to define analog or mixed-signal IP blocks corresponding to the digital blocks in your design, you can override the digital blocks with the blocks from Virtuoso. These overridden blocks are then used to create netlists that contain a mixed-signal representation of the complete hierarchy. The *Virtuoso Directories* section in the CLIPS window is used to specify the location of the directories that contain the Virtuoso library files, `cds.lib`. These `cds.lib` files contain the paths to the libraries where design files are saved. When you specify a Virtuoso directory, CLIPS uses a nickname in the `<Virtuoso directory nickname>:<library>/<cell>/<view>` format to identify that directory.

The following figure shows the *Virtuoso Directories* section that you can use to manage Virtuoso directories in CLIPS.



The table given below describes the command buttons available in this section:

Command	Description
<i>Add new Virtuoso Directory</i> 	Opens a file browser in which you can select the Virtuoso library file, <code>cds.lib</code> , to import a Virtuoso directory that contains the mixed-signal IP configurations (Virtuoso config or schematic views).
<i>View Virtuoso Directory</i> 	Opens the selected <code>cds.lib</code> file in the Virtuoso Library Manager.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

---

**Delete Virtuoso Directory** Deletes the selected `cds.lib` file.



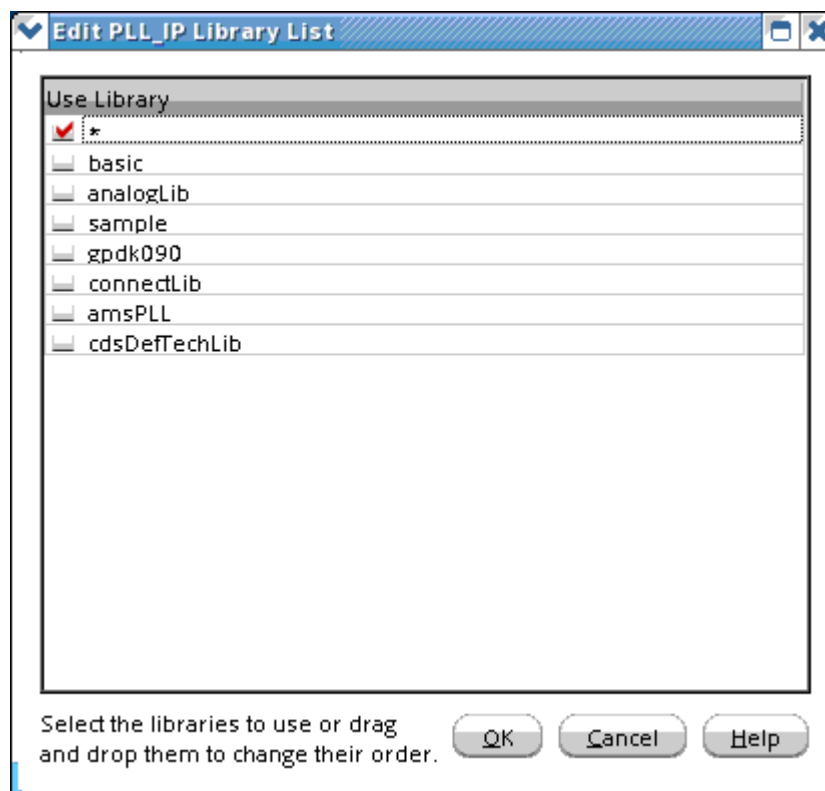
---

CLIPS searches the specified Virtuoso directories to look for the Virtuoso config files that can override the corresponding digital block in your design. By default, all the directories specified in the *Virtuoso Directories* section can be searched for config files. This is indicated by the \* symbol shown in the Libraries column of this section. However, you can choose the libraries to be searched while finding the schematic or config files that can be overridden, that is, switchable IP configurations. Other Virtuoso directories are excluded from the search. This helps improve the searching and matching performance.

To select the libraries to be searched, perform the following steps:

1. Double-click the *Libraries* column in the Virtuoso Directories section.

The Edit <Library\_name> Library List form appears.



2. Select the check box next to the libraries you want to use in search.

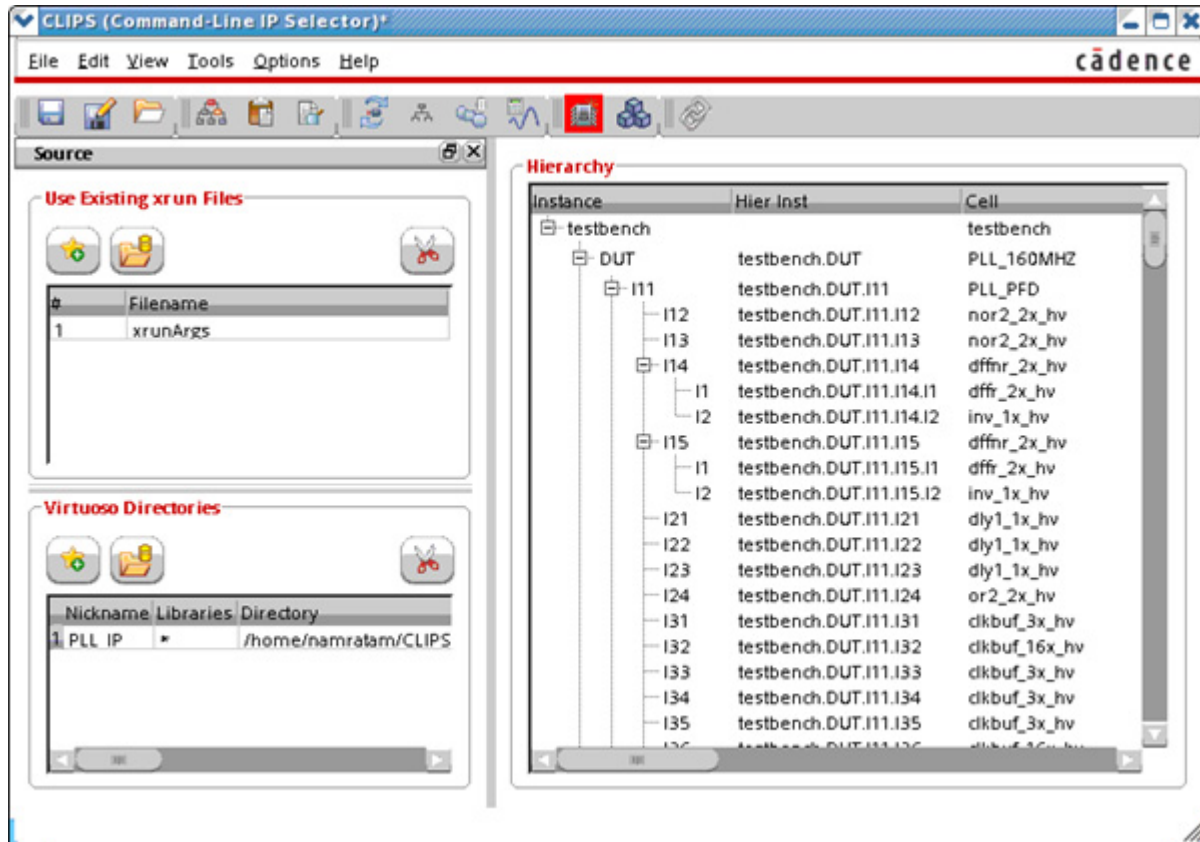
## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

Alternatively, you can drag and drop the library names in this list to specify the search order.

3. Click **OK** to close the form.

The `xrunArgs` files are elaborated and the digital hierarchy is represented in a tree in the *Hierarchy* section.



## Managing Instance Hierarchies and Bindings

In the *Hierarchy* section, you can switch or override the cellview bindings for selected blocks or cells to use config or schematic views from Virtuoso instead of the text cellviews (current bindings). If you switch the binding for a cellview to use a config or schematic view, you also need to either associate that cell with an ADE state, which provides the information required for netlisting, or directly add model files.

**Note:** For netlisting, it is not necessary to have model files for schematic view. It is generally required while simulating.

The following topics provide more details:



- Controlling the Hierarchy View
- Overriding or Switching Instance Bindings
- Associating an Instance with an ADE State
  - Configuring Settings and Options

## **Controlling the Hierarchy View**

For a large design hierarchy, you might need to scroll to look for a specific cell or view. Instead, you can use the controls provided in the *Hierarchy* section to view the design components of interest.

To access the controls, select *View – Hierarchy Controls*.

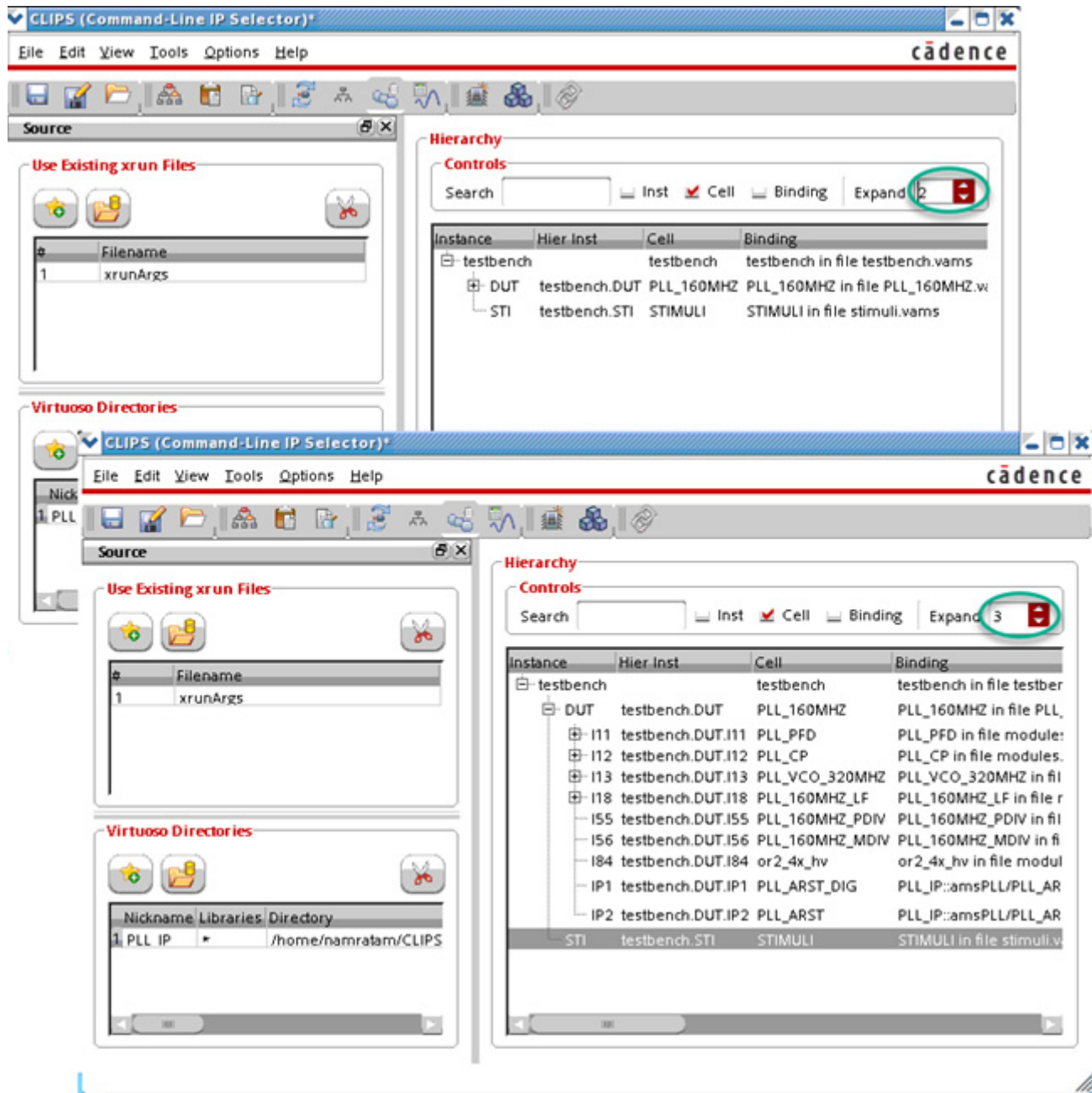
The *Controls* group box is displayed in the *Hierarchy* section. You can use the controls in this box to do the following:

- Search for a specific instance, or cells or bindings. For this, you can type in the search box.
- Limit the view to display only instance or cells or bindings. For this, you can select the corresponding check box.

## Command-Line IP Selector (CLIPS) User Guide


### Working with CLIPS

- Expand the instances in the hierarchy to higher levels to get a more detailed tree structure. For this you can select the value from the drop-down list box, as shown in the figure below.

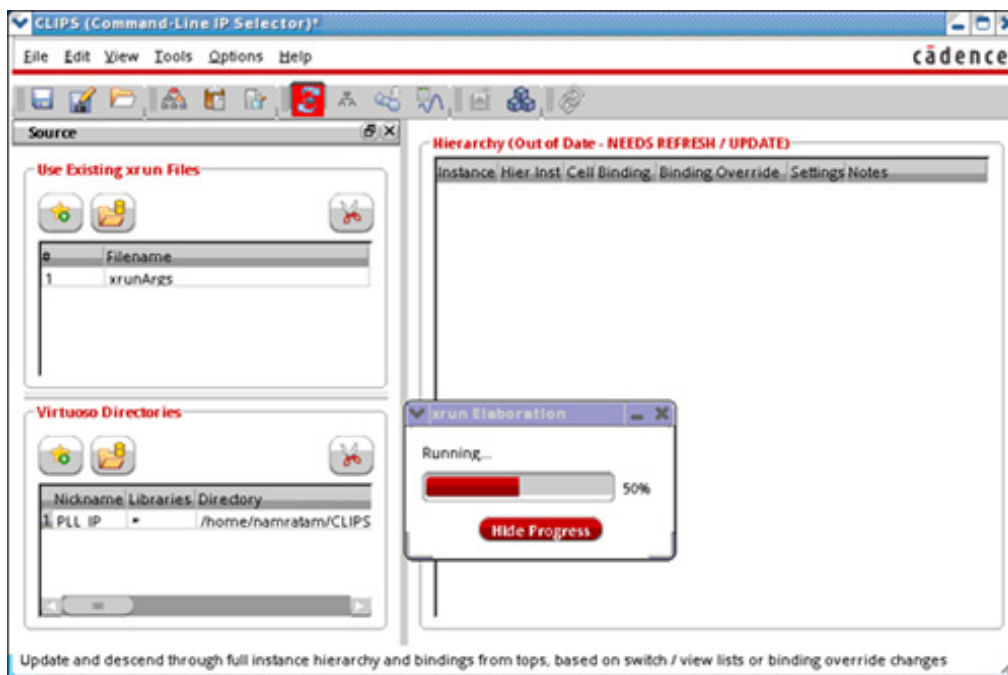


## Overriding or Switching Instance Bindings

To switch instance bindings:

1. Add xrun files in the *Use Existing xrun Files* section.
2. Add Virtuoso directories in the *Virtuoso Directories* section.
3. Click *Update*  on the toolbar to update the hierarchy after adding all the xrun scripts and Virtuoso directories.

CLIPS elaborates the design to identify the Virtuoso cellviews available for the cells listed in the *Cell* column.

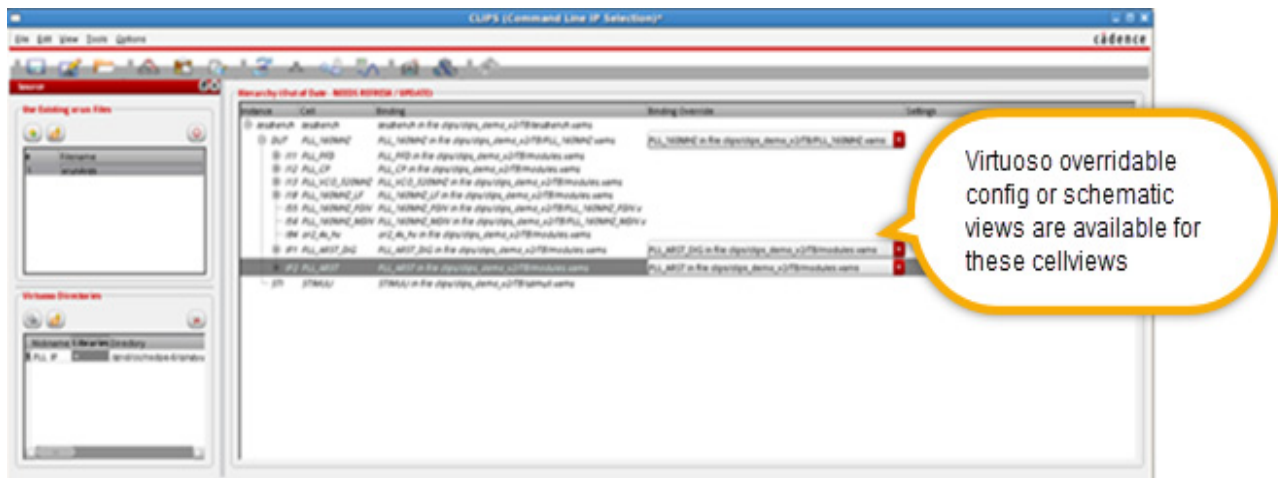


Only Virtuoso cellviews for which either of the following conditions is satisfied are displayed in the *Binding Override* column:

- ☐ a config view is available and the design sub-top (schematic or text) cell name is the same as the cell or block name
- ☐ a schematic view is available and the cell name is the same as the cell or block name

# Command-Line IP Selector (CLIPS) User Guide

## Working with CLIPS



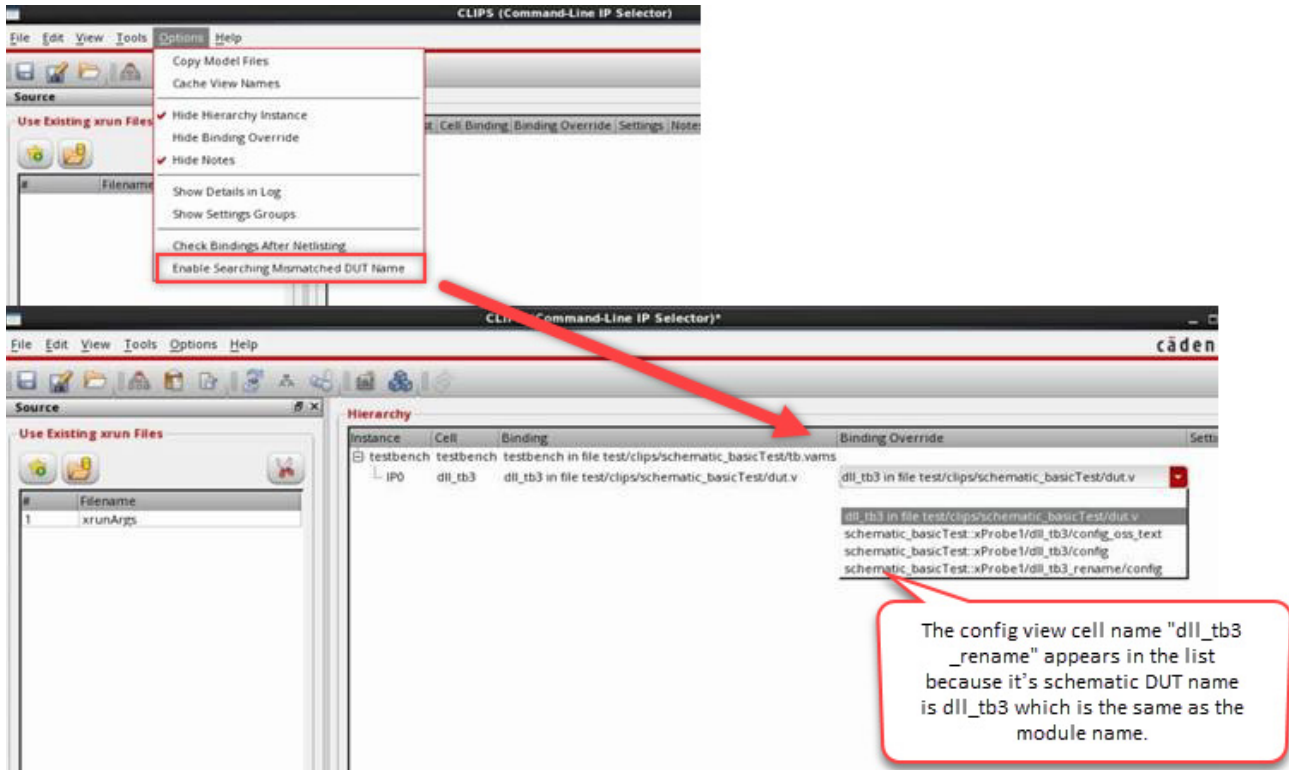
If the original text cellview is used for a cell, the drop-down list in the *Binding Override* column continues to show the directory path to that text view. If you override the cellview binding to use a Virtuoso config or schematic view instead of the text cellview, the drop-down list shows the name of the corresponding library, cell, and config or schematic view.


**Note:** You can find all the Virtuoso config or schematic views whose schematic DUT name matches the module name of digital text file by selecting *Enable Searching*

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

*Mismatched DUT Name in the Options menu.*



For a large design, if you need to scroll down to view the details of all instances and cells, you can click the *Smart Hierarchy Tree* button  on the toolbar to collapse all instances other than the ones for which switchable bindings are available. This makes it more convenient to locate switchable bindings in a large design with a long hierarchy tree.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

4. For each cell for which you need to override the cellview binding, select the config or schematic cellview name from the drop-down list in the *Binding Override* column.



**Note:** You can specify a testbench and it shows the design hierarchy. From the hierarchy, you can switch any instance with available binding override choices to an HED configuration. Then, AMS-UNL is called to export the HED configuration as an IP and assemble the IP together with the testbench.

After overriding the instance, the next step is Associating an Instance with an ADE State. And then you have to Add or edit model files in CLIPS.

### Associating an Instance with an ADE State

If you switch the original binding of an instance with a Virtuoso config or schematic view, you need to associate the instance with an ADE state for providing the information required for the netlisting. However, in case the ADE state provides only model files for netlisting and simulation, you can directly add the model files in CLIPS.

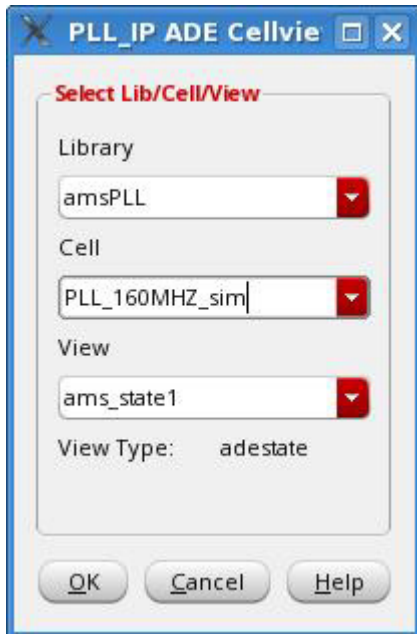
To associate an instance with an ADE state:

## Command-Line IP Selector (CLIPS) User Guide


### Working with CLIPS

---

1. Right-click the row of the instance for which you have overridden the binding and choose *Add/Edit ADE Cellview*.



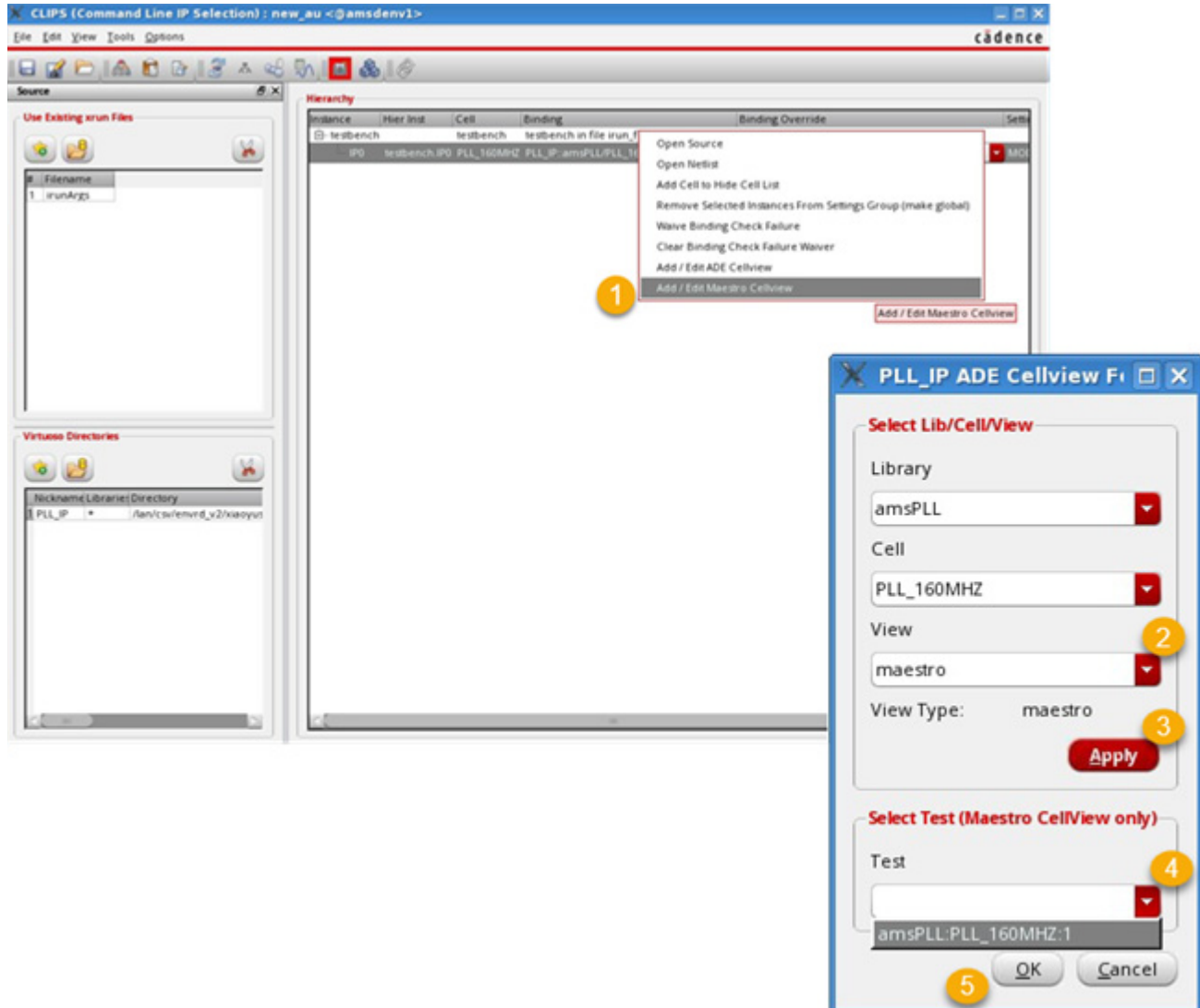
**Note:** When an ADE state is added to an instance in the digital hierarchy, a Virtuoso nickname is added as a prefix to the ADE cellview lib/cell/view in the settings column for the instances with which it is associated.

2. In the *View* drop-down list, select the name of the ADE state to be associated with the instance and click *OK*.
3. Click *Update*  on the toolbar to update the hierarchy.

# Command-Line IP Selector (CLIPS) User Guide

## Working with CLIPS

**Note:** You can also attach the Maestro cellview state in CLIPS. Follow the steps as highlighted in the figures below.





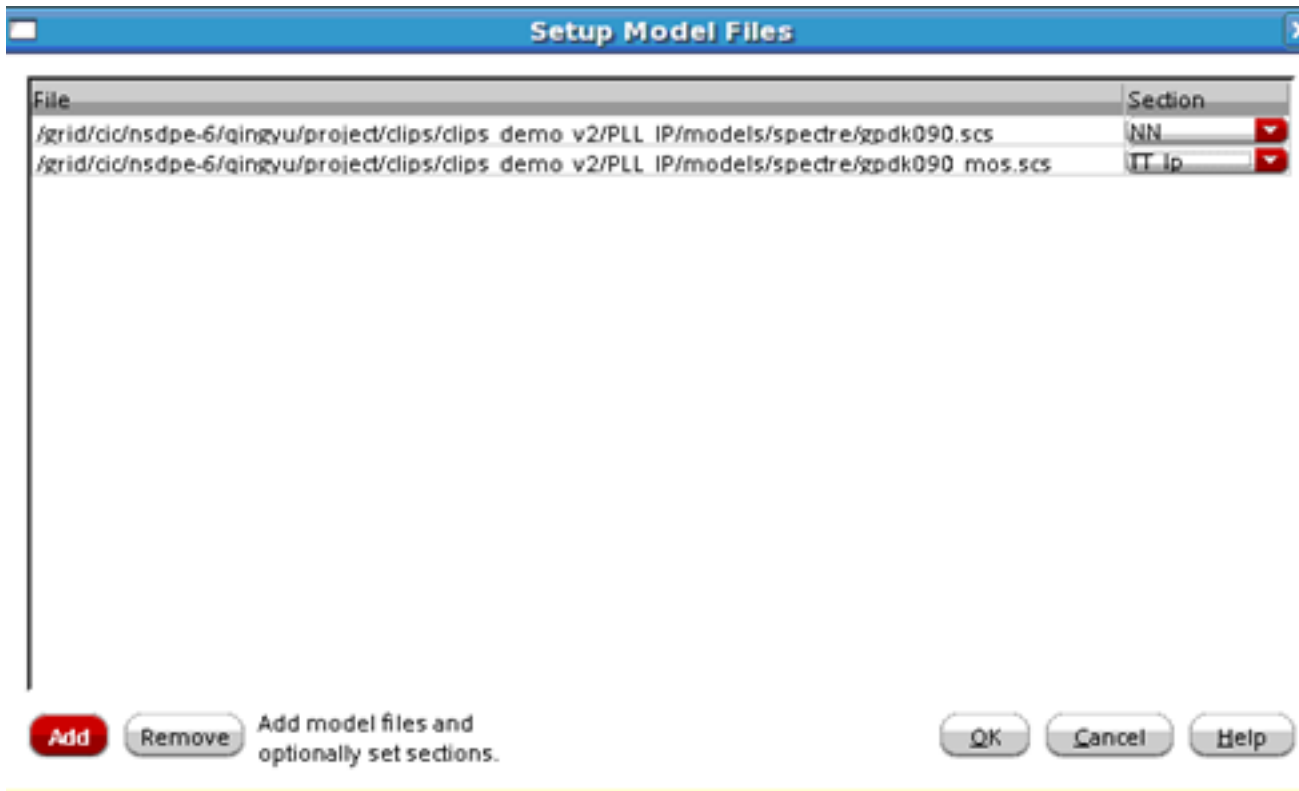
## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

---

#### Add or edit model files in CLIPS

1. Choose *Tools – Setup Model Files*. The Add/Edit Model Files form appears.



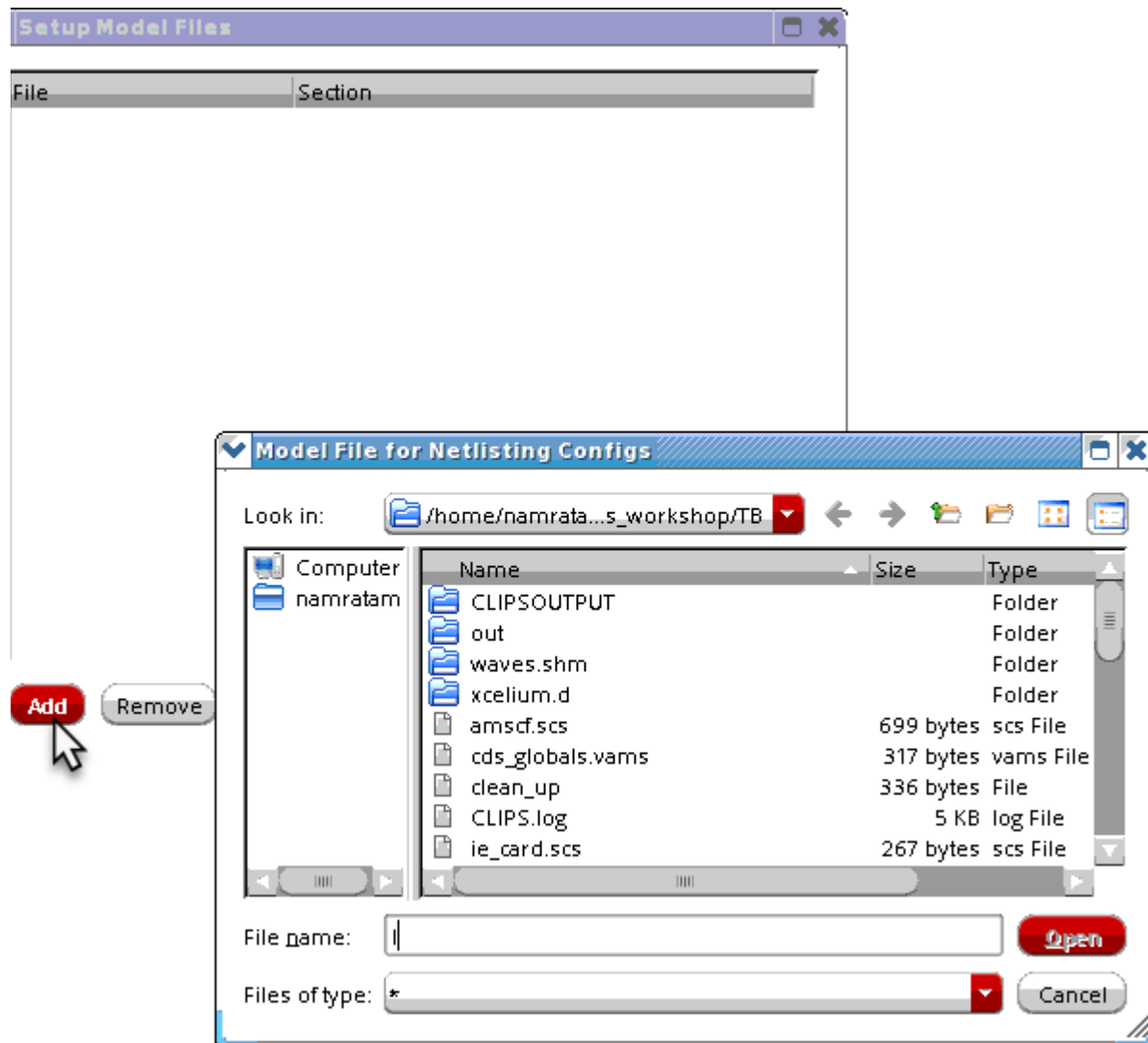
In this form,

- ❑ Click *Add* to open the file browser and select model files.
- ❑ Select a file and click *Remove* to remove it.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

- The following figure shows how to add Model files.



2. Click OK.

## Configuring Settings and Options

By default, all the instances in the hierarchy table inherit the global design settings, such as model files, design information, and netlisting options. If required, you can specify different values to these settings for one or more hierarchical instances. For that, you need to create a group that contains one or more hierarchical instances, and then, specify settings for the group.

# Command-Line IP Selector (CLIPS) User Guide

## Working with CLIPS

In addition to the design settings mentioned above, you can define options to specify the pre-processing and post-processing options for a CLIPS run and directories to be used. The following sections provide details about how to configure these settings and options:

- [Creating Groups](#)
- [Applying Settings to Specific Groups](#)
- [Configuring Global Options](#)

## Creating Groups

To create a group to include specific hierarchical instances, perform the following steps:

1. Choose *View — Show/Hide Settings* to open the Settings and Options form.

**Settings and Options**

**Settings Groups**

Name	Hier Inst
1 Global	All Others not in Groups
2 testbench.DUT.IP1	testbench.DUT.IP1
3 testbench.DUT.IP2	testbench.DUT.IP2

**Settings (Global)**

Description	Value
ADE Cellview	
Config Netlisting Model Files	

**Global Options**

Description	Value
Virtuoso Setup Script	
Netlisting Output Directory	.

The available groups are displayed in the *Settings Groups* section.

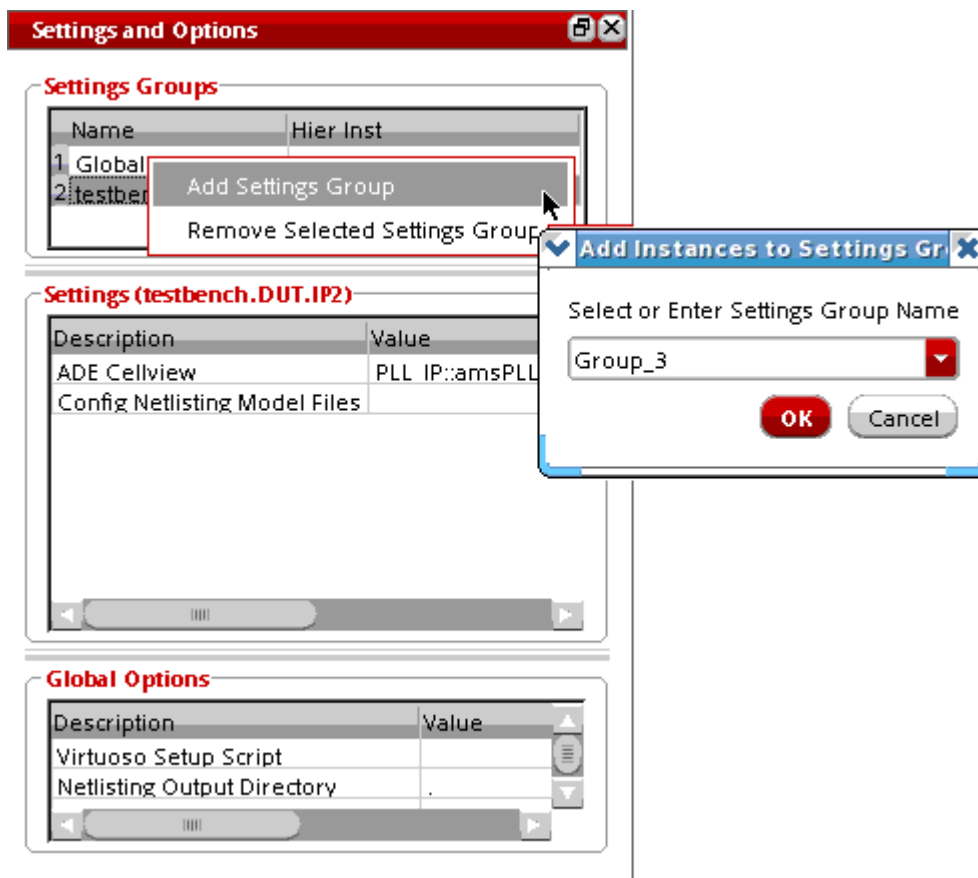
**Note:** By default, only one group named *Global* is created. All the hierarchical instances belong to this group and use the common settings.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

2. To create a new group, right-click in the *Settings and Groups* section and choose *Add Settings Group* from the context-sensitive menu. The Add instances to Settings Group form is displayed.
3. Specify a name for the new group in the *Select or Enter Settings Group Name* field.
4. Click OK to close the form.

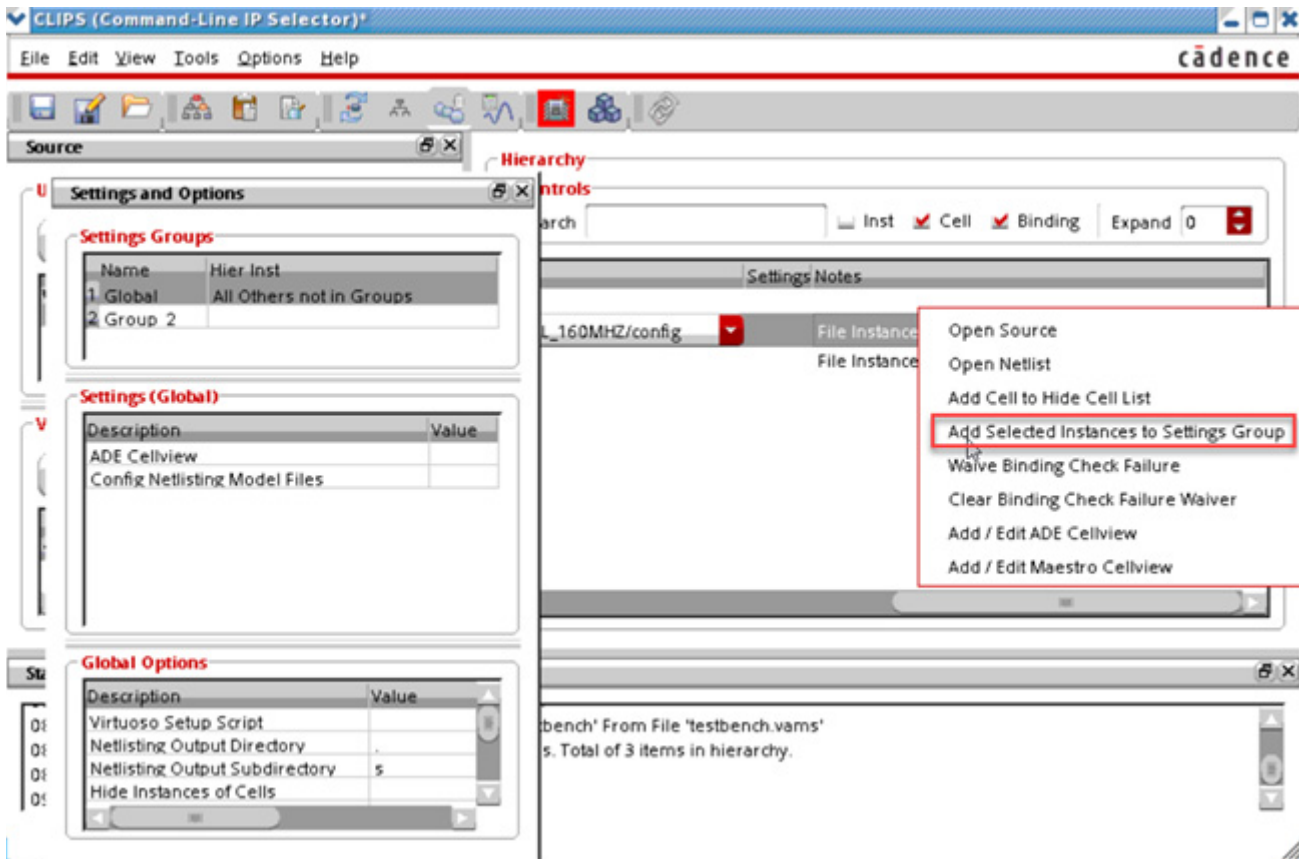
A new group is displayed in the *Settings Groups* section.



## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

5. To add an instance to this group, right-click an instance in the *Hierarchy* sections of the CLIPS window and choose *Add Selected Instances to the Settings Group*.



The Add Instances to Settings Group form is displayed.



6. In the *Select or Enter Settings Group Name* drop-down list, select the name of the group to which you want to add the instance.

**Note:** You can add more than one instance to the group.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

---

The instance name is displayed in the *Hier Inst* column of the *Settings Groups* section, as shown below.

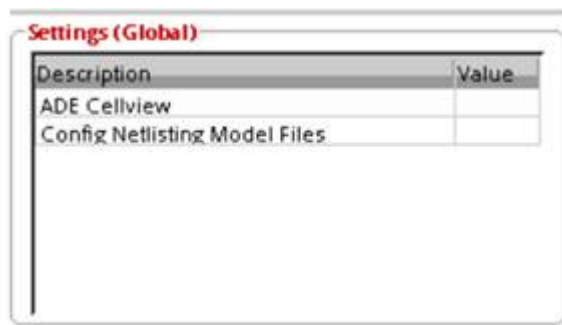


## Applying Settings to Specific Groups

Settings specified for the Global group are applied to all the instances that are not a part of any other group.

To apply different settings to some specific instances, perform the following steps:

1. In the *Settings Groups* section, select the group name that contains the hierarchical instances for which you need to modify the settings. The properties of the selected groups are shown in the *Settings (<group-name>)* section.



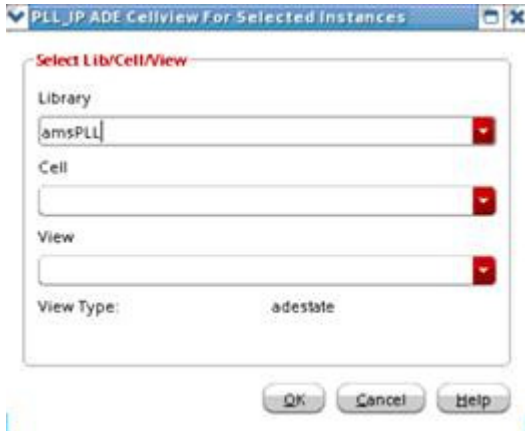
## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

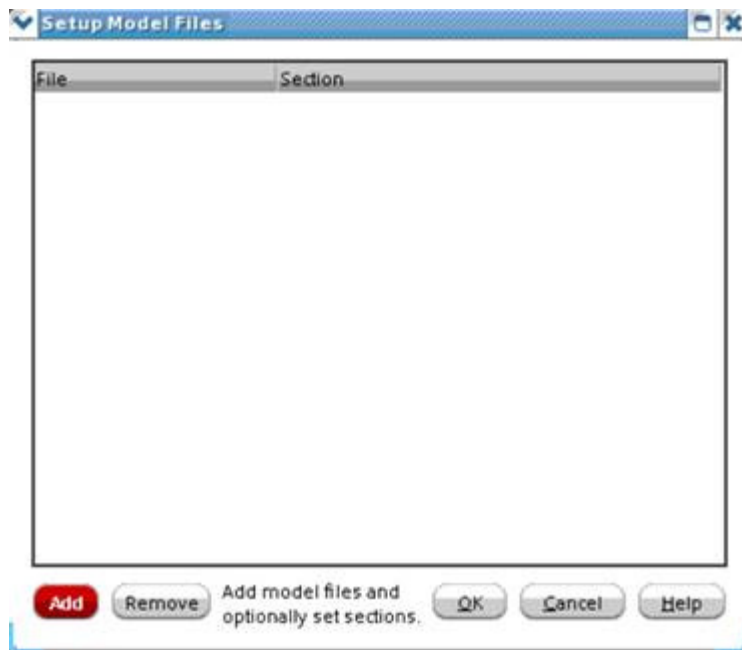
---

2. Double-click the *Value* cell corresponding to the setting you want to modify.
3. Enter the value to the form that is displayed.

**Note:** Different forms are displayed to specify the ADE cellview or model files. For ADE cellviews, the *<inst-name> ADE Cellview for Selected Instances* form is displayed, as shown below.



For *Config Netlisting Model files*, the *Setup Model* form is displayed, as shown below.



## Configuring Global Options


In the *Global Options* section on the *Settings and Options* form, you can specify the global pre- and post-processing options for simulations and the directories to be used for simulation runs.

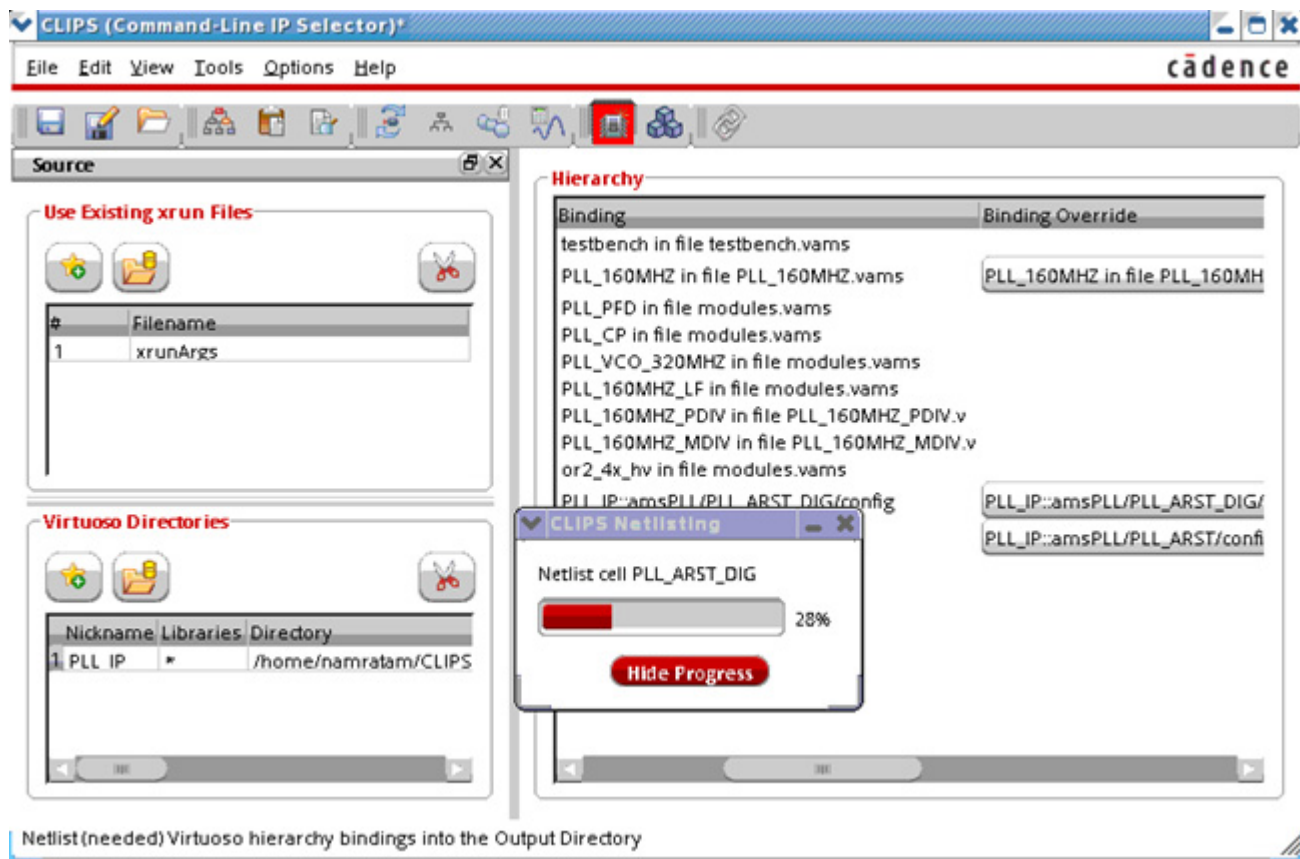
The options you can configure in this section are listed below.

- **Virtuoso Setup Script:** Path to the UNIX script to set up Virtuoso Environment Variables.
- **Netlisting Output Directory:** Name of the output directory.
- **Netlisting Output Subdirectory:** Name of the subdirectory inside the output directory where the netlist is saved.
- **Hide Instances of Cells:** List of cellnames for which instances are hidden from the hierarchy.
- **Batch Pre-Netlist Command:** UNIX command to run before generating a netlist in the batch mode.
- **Batch Post-Netlist Command:** UNIX command to run after generating a netlist in batch mode. This command is always run, irrespective of the pass or fail status of netlist creation.
- **Batch Post-Netlist Pass Command:** UNIX command to run after successful creation of a netlist in batch mode.
- **Batch Post-Nestlist Fail Command:** UNIX command to run after creation of a netlist fails in the batch mode.
- **Liblist in the Verilog2001 Configuration File:** A new or modified list of libraries. If this field is empty, the default library list, `worklib` is used.



## Netlisting the Virtuoso Config or Schematic Views

Once you have updated the hierarchy after switching the instance bindings, click *Netlist*  on the toolbar to netlist the Virtuoso config or schematic views.



It generates an incremental file, `clips.f`, along with the other necessary files.

When netlisting completes, messages are displayed in the terminal, as shown below:

```
Finished Netlisting 'PLL_IP::amsPLL/PLL_ARST_DIG/config' (testbench.DUT.  
IP1)  
Finished Netlisting 'PLL_IP::amsPLL/PLL_ARST/config' (testbench.DUT.IP2)  
Netlisting Complete
```

## Performing a Binding Check

Mostly, the top-level bindings (outside the IP block) are the native bindings by the Verilog language, which means that while searching for Virtuoso cellviews, it follows the search order

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

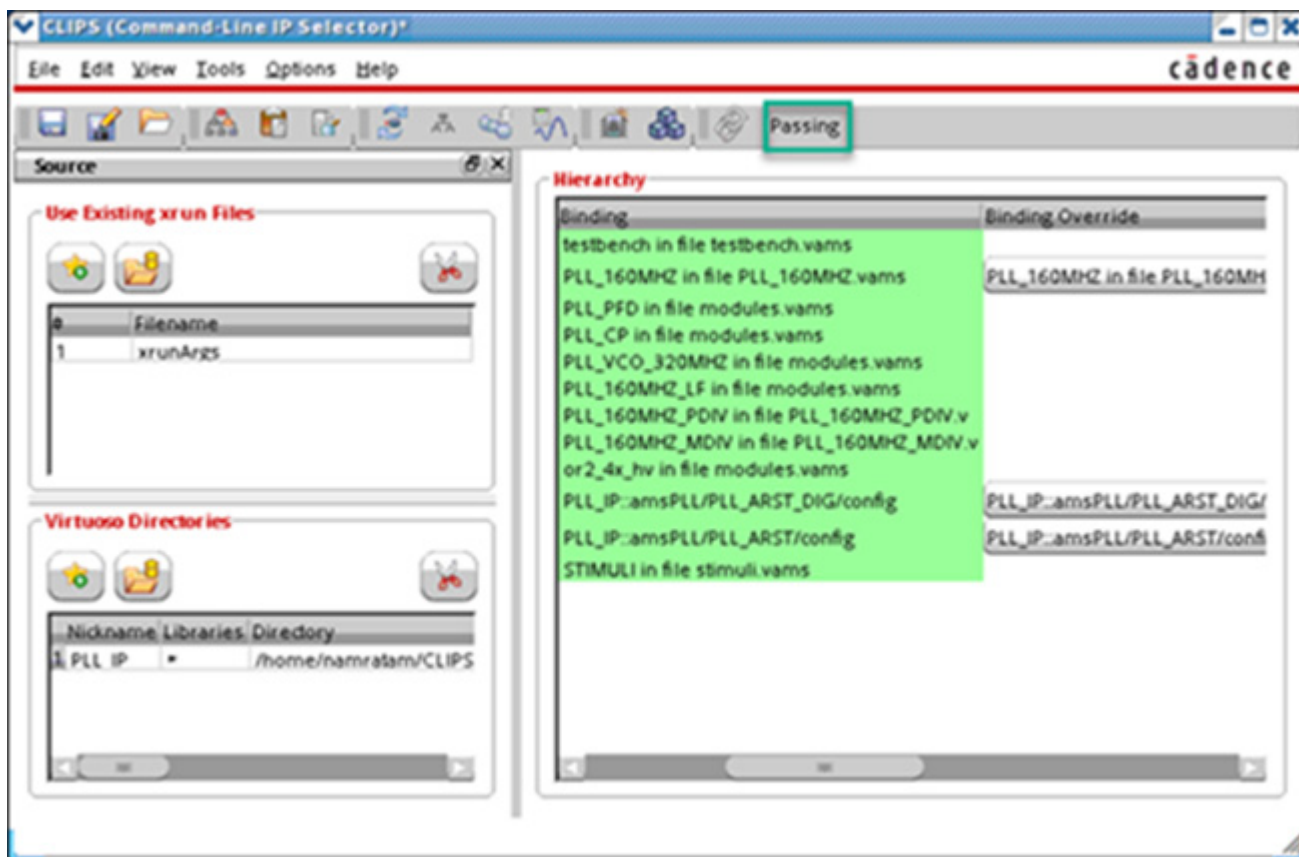
set in the Xcelium simulation. This kind of native binding gets changed during a binding switch because new libraries are introduced.

Therefore, it is highly recommended to check whether the switched bindings are correctly used by the elaborator before running the post-CLIPS simulation. The existing bindings for all the remaining instances must remain intact.

To perform a binding check, follow the steps:

- ➔ Click *Check Bindings*  on the toolbar.

The pre-CLIPS and post-CLIPS elaboration results are compared, and any mismatch found is highlighted. The correct bindings are highlighted in green. If an incorrect binding is found, it is highlighted in red.




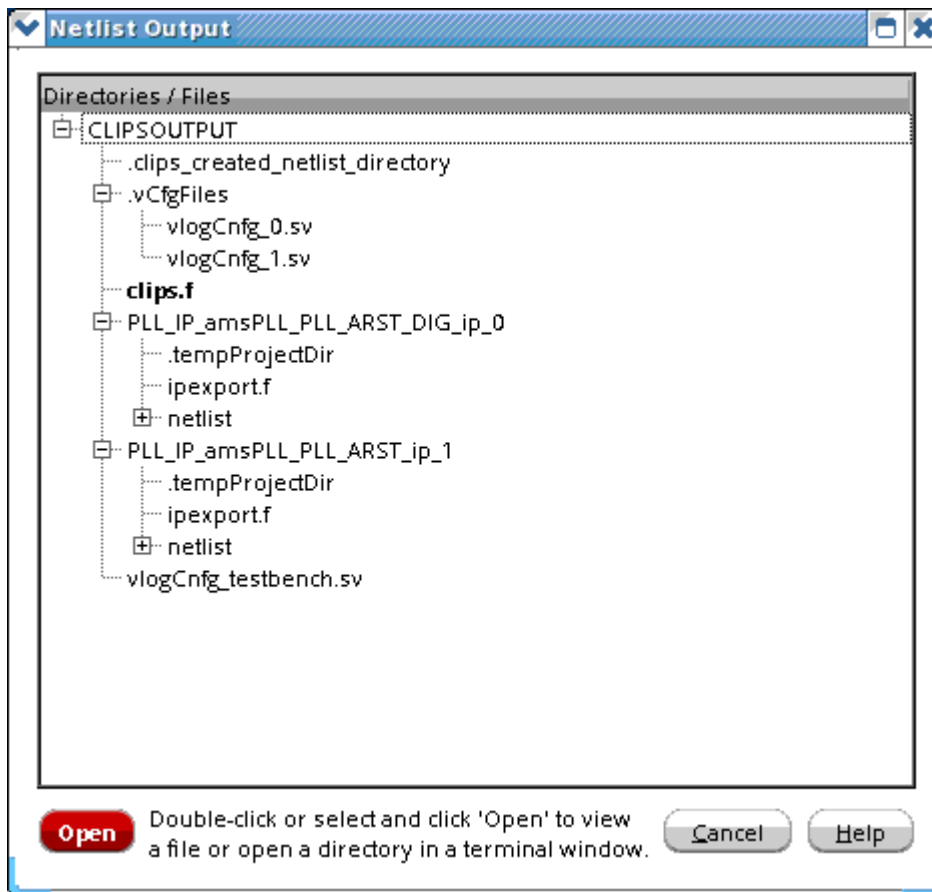
The result of binding check is displayed in the toolbar, where “Passing” indicates that correct bindings are confirmed.

**Note:** If you have enabled binding check in the GUI and saved it to the `.clips` file, the binding check will be performed when CLIPS is run in batch mode, unless `-checkbind` no is passed on the command line.

## Viewing the Netlist Output

By default, all switched config or schematic views are netlisted in the `./CLIPSOUPUT` directory.

- ➔ To view the contents of this directory, click *View Output*  on the toolbar. The directory structure is displayed in the Netlist Output window, as shown below.



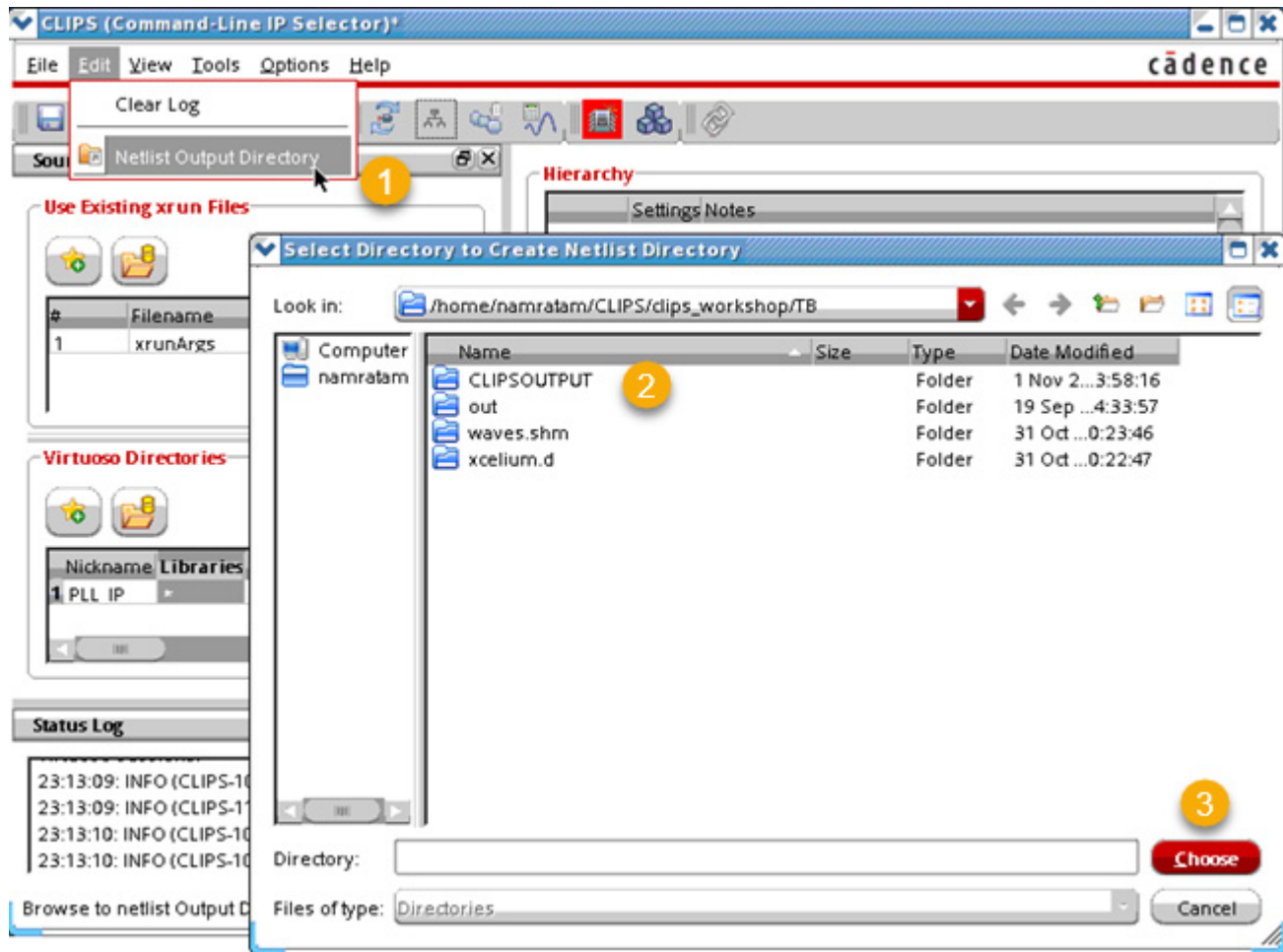
In this window,

- ❑ To view any file, double-click the file, or select the file and click *Open*.
- ❑ To open the terminal in a directory, double-click the directory, or select the directory and click *Open*.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

**Note:** You can set your own output directory, by following the steps highlighted in the figure below.



- ➔ To view the list of contents of this directory on the command line, enter the following command in the terminal:

```
% ls ./CLIPSOUTPUT/
```

In the CLIPSOUTPUT directory, separate directories are generated for all the config or schematic views bound to any of the instances. The `clips.f` file, which contains information about all the Virtuoso configurations, is also placed in this directory.

The following example shows the content available in the `clips.f` file:

```
# 'PLL_IP::amsPLL/PLL_ARST_DIG/config' first used for 'testbench.DUT.IP1'
-f /grid/cic/nsdpe-6/qingyu/project/clips/clips_demo_v2/TB/CLIPSOUTPUT/
PLL_IP_amsPLL_PLL_ARST_DIG_ip_0/ipexport.f
# 'PLL_IP::amsPLL/PLL_ARST/config' first used for 'testbench.DUT.IP2'
```

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

---

```
-f /grid/cic/nsdpe-6/qingyu/project/clips/clips_demo_v2/TB/CLIPSOUTPUT/  
PLL_IP_amsPLL_PLL_ARST_ip_1/ipexport.f  
-top vlogCnfg_testbench  
/grid/cic/nsdpe-6/qingyu/project/clips/clips_demo_v2/TB/CLIPSOUTPUT/  
vlogCnfg_testbench.sv  
-compcnfg
```

Additionally, a SystemVerilog configuration file, `vlogCnfg_testbench.sv`, is generated to bind all the configurations under the top-level testbench.

```
config vlogCnfg_testbench;  
    design testbench;  
    default liblist worklib;  
    //verilog 2001 binding for this IP  
    cell PLL_ARST_DIG use cds_amsconfiglib.PLL_ARST_DIG:ip_0;  
    cell PLL_ARST use cds_amsconfiglib.PLL_ARST:ip_1;  
endconfig
```

In each Virtuoso configuration directory, the config or schematic view is netlisted in the same way as it is netlisted in the UNL flow. However, it is better packaged with an `ipexport.f` file, which contains the details of the Virtuoso config or schematic view netlisted by CLIPS.

## Accessing Logs

CLIPS logs show useful information about everything happening in the current session.

You can show and hide the log area of CLIPS.

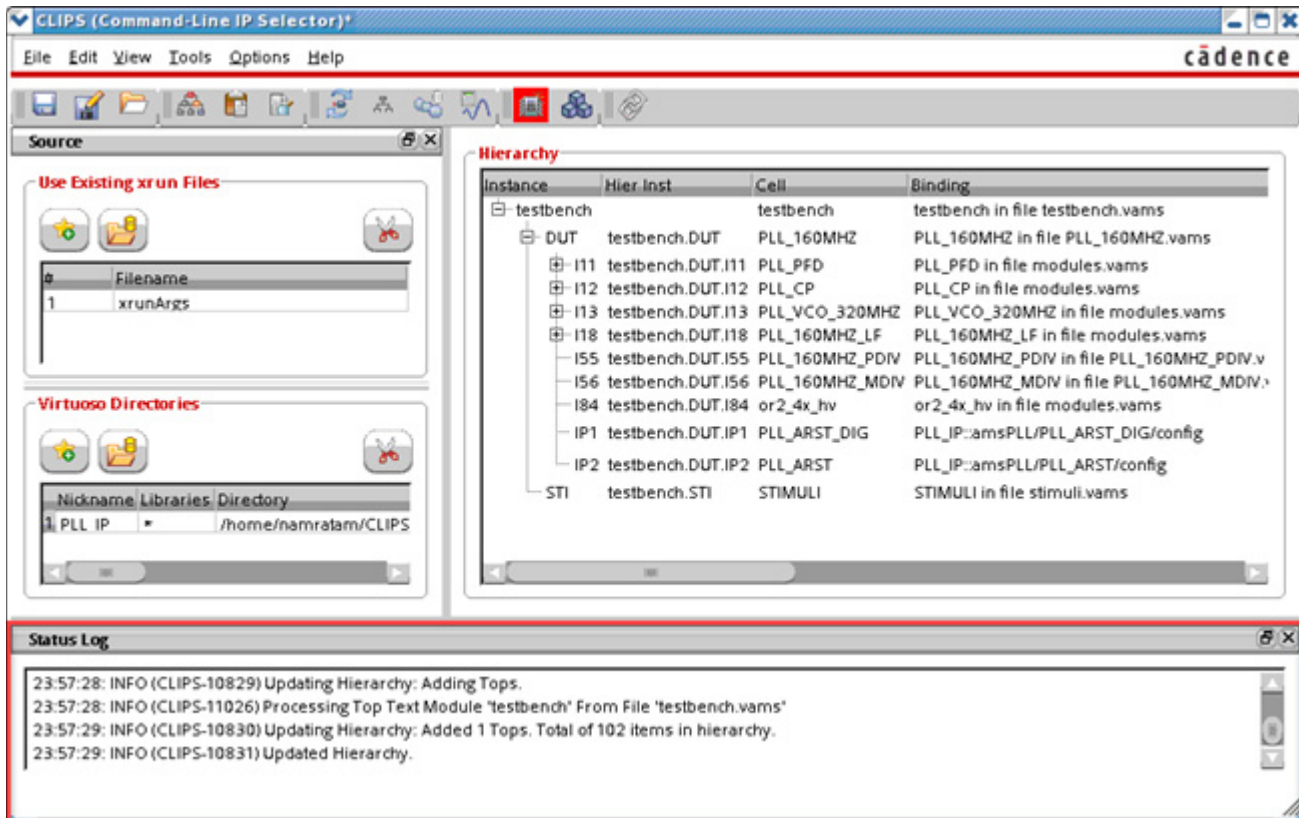
To view the log pane, do one of the following

1. Choose *View — Show / Hide Log*.

## Command-Line IP Selector (CLIPS) User Guide

### Working with CLIPS

2. Click  in the toolbar.



To capture the log files and intermediate files in a specified directory, perform the following steps:

1. Set the environment variable `CLIPS_DEBUG="dbg"`.
2. Create a local directory. For example, create a directory `local_tmp` using the command:

```
mkdir local_tmp
```

3. Use the following option in CLIPS command-line:

```
-tmpworkpath ./local_tmp
```

4. The content of `virtuoso_<tempID>/directory` is saved in the locally-defined directory `local_tmp`.

---

## Understanding the Difference Between CLIPS and Unified Netlisting (UNL)

---

CLIPS netlisting procedure works in a different manner as compared to the UNL in ADE. Following are the differences between the two netlisting procedures:

- Files generated and included in Virtuoso IP netlisting:
  - All Whitebox Design Units (WDU) are netlisted in the same way as they are during UNL in ADE.
- Files generated but modified in Virtuoso IP netlisting:
  - `spiceModels.scs` and `scopeSpiceModels.scs`:
    - Path in the `spiceModels.scs` file is an absolute path.
    - `modelincdir` is not applied.
    - The scope-setting in the `scopeSpiceModels.scs` file is for the analog IP, which means that models inside this IP only search for the model files in this file (and its included files).
  - `cds_globals`
    - For the original `cds_globals.vams` from the SoC simulation setup (possibly from the UNL result), if there is no `-top` specified in the original `xrunArgs` file, CLIPS will add `-top cds_globals` in the `clips.f` file.
    - For the CLIPS netlisted `cds_globals`, a unique ID is generated and added to the name of each global module file (for example, `cds_globals_ip_0`). The name of the referencing parameter/signal is also added to the name of the module (for example, `cds_globals_ip_0.res`, `cds_globals.\VDD!`).
  - Files generated but not included in Virtuoso IP netlisting:
    - `svPkgTextInputs/vhdlPkgTextInputs` – It is suggested that you set this in the SoC simulation setup configuration, `xrunArgs`.

## Command-Line IP Selector (CLIPS) User Guide

### Understanding the Difference Between CLIPS and Unified Netlisting (UNL)

---

- `ie_card.scs` – Since Interface Element (IE) cards are not supported in CLIPS, `ie_card.scs` is netlisted but commented out from the `ipexport.f`. If needed, you can manually modify the `ipexport.f` to include the `ie_card.scs` file.
- `probe.tcl` – It is suggested that you set this in the SoC simulation setup configuration, `xrunArgs`.
- `amsControlSpectre.scs` – This file contains only the analog simulation options. It is suggested that you set this in the SoC simulation setup configuration with an additional `xrunArgs` file.

#### ■ Options generated from an ADE L state but not included in the Virtuoso IP netlisting:

##### □ `-modelincdir`

- You can set this only once. For adding multiple paths, use a colon ':' to separate the paths.
- Global setting changes the search order of model files.
- You can set this in the SoC simulation setup (outside the testbench), if needed.
- You can set the full path or the relative path from the Virtuoso invoking directory (where the `cds.lib` file is placed) for all the model files.

##### □ `-indir` or `+incdir` – Can be set sparsely, but all of the settings are applied globally.

##### □ `-reflib` and `-makelib`

- These options are not printed in the `ipexport.f` file.
- The `runtimeCompileFiles` file is generated in the netlist directory, but you need to add it in the `ipexport.f` file manually.



---

## CLIPS Environment Variables

---

This appendix describes the public shell environment variables that control the characteristics of CLIPS. You can customize the operation and behavior of the tool by changing the value of a particular environment variable.

- LOGNAME
- USER
- CLIPS\_MODELFILE
- CLIPS\_HIDE\_CELLS
- CLIPS\_NETLISTING\_TIMEOUT
- CLIPS\_VIRTUOSO\_TIMEOUT
- CLIPS\_CDS\_BROWSE\_MODEL
- CLIPS\_INST\_CNFG
- CLIPS\_VIRTUOSO\_BOURNE\_SHELL
- CLIPS\_DEBUG

## Command-Line IP Selector (CLIPS) User Guide

### CLIPS Environment Variables

---

Environment Variable	Description	Default Value	Example
LOGNAME	Returns or prints the user's login name. It is a standard UNIX shell variable.	--	LOGNAME
USER	Specifies the user name. It is a standard UNIX shell variable. This user name is used in case the LOGNAME environment variable is not set.	--	if ["\$USER" ="testuser"] then ...
CLIPS_MODELFILE	Predefines model files to be used within CLIPS in the same format as the <code>-modelfile</code> command-line argument. These model files are used for netlisting configurations.  The following syntax is used for defining model files:  <model1[:model2(sectionName)]>	Blank list	setenv CLIPS_MODELFILE "/path1/ path2/ models.scs"
CLIPS_HIDE_CELLS	Specifies a comma or space-separated list of cell or module names to be excluded from the hierarchy tree.	Blank list	setenv CLIPS_HIDE_CELLS "cell1, cell2"
CLIPS_NETLISTING_TIMEOUT	Specifies the maximum duration (in seconds) allowed for netlisting before timing out.	300 (seconds)	setenv CLIPS_NETLISTING_TIMEOUT "150"
CLIPS_VIRTUOSO_TIMEOUT	Specifies the maximum duration for which CLIPS can attempt to connect to a Virtuoso session.  Possible values are: Any integer between 4 and 999 (seconds)	90 (seconds)	setenv CLIPS_VIRTUOSO_TIMEOUT "50"

## Command-Line IP Selector (CLIPS) User Guide

### CLIPS Environment Variables

---

CLIPS_CDS_BROWSE _MODEL	Prompts to add model files to be used with Virtuoso for netlisting. This would happen when a Virtuoso session is added.	--	setenv CLIPS_CDS_BROWSE_MODEL "xxx"
CLIPS_INST_CNFG	Generates all the switched bindings in explicitly instance-based configurations in Verilog-2001 config file.	NO	setenv CLIPS_INST_CONFIG "yes"
CLIPS_VIRTUOSO_BOURNE_SHELL	It is a standard UNIX shell variable. Set this variable if cdsPre script uses Bourne shell syntax; it ensures the parsing or analysis of a .clips file containing Bourne shell commands. By default, the cdsPre script uses C shell syntax.	--	setenv CLIPS_VIRTUOSO_BOURNE_SHELL "yes"
CLIPS_DEBUG	Saves the debugging logs locally on a user-defined log file.	--	setenv CLIPS_DEBUG "dbg"