cadence™

Conformal® Constraint Designer Known Problems and Solutions

Product Version 23.2
October 2023

Cadence Design Systems, Inc., 2655 Seely Avenue, San Jose, CA 95134, USA

Contents

Conformal Constraint Designer Known Problems and Solutions

Product Version 23.2
October 2023

This Known Problems and Solutions document describes important Cadence Management System Change Records (CCRs) for Encounter® Conformal® Constraint Designer and tells you how to solve or work around these problems.

Important: Only known problems and solutions available at release time are available in this document. An up-to-date list of known problems for the Conformal Constraint Designer software is published in SourceLink.

For information about CCRs that were fixed in this release, see the `README` file accompanying this release. You can read this file online at downloads.cadence.com. For information about new, changed, and deleted features, compatibility issues, and documentation changes, see *What's New in Conformal Constraint Designer*.

General

KornShell (ksh) Bug with RHEL 6.2

Description: When exiting a subshell after a command substitution, ksh could prematurely exit without any error. This bug exists with RHEL 6.2 users with ksh-20100621-12.el6.x86_64 rpm.

This bug can cause Cadence wrapper scripts to exit before setting the correct user environment and lead to tool crashes or run errors.

Solution: Use ksh-20100621-16.el6.x86_64 rpm instead. With this version, ksh no longer terminates under the above circumstances and all subsequent commands are processed correctly.

CCR 1073366: Tool Crashes During Elaborate
CCR 1074734: Error when Setting Mode to Verify

Description: The tool cannot handle unintended global net references in different design trees. When encountered, the tool can crash during flattening or netlist crossing checks.

For example, if there are wire declarations in the `port.vh` file:

```
wire clk;
wire rst;
```

and they are included in `design1.v`. However, at the same time, in `design2.v` and `design3.v` have implicit wire declarations in instantiation statements:

```
// design2.v
dut1 d1 (.rst(rst), .clk(clk), ...)
```

and

```
// design3.v
dut2 d2 (.reset_b(rst), .clk_b(clk), ...)
```

If the `read design` command includes `port.vh` (as shown in the following command), the tool will crash when switching from Setup mode to LEC/Verify mode:

```
read design -sv port.vh design1.v design2.v design3.v -root design2
```

Solution: Elaborate the design with the `-rootonly` option or remove the offending/ unnecessary file from the `read design` file list.

For instance, the crash in the example above can be avoided by changing the read design command to:

```
read design -sv design1.v design2.v design3.v -root design2
```

Error When Using a Verilog Library

Description: As with other Conformal tools, the default library format in the `READ LIBRARY` command is Verilog. However, for proper interpretation of the design's timing characteristics, the Conformal Constraint Designer relies on information it can only find in Liberty libraries. Cadence recommends using the Liberty library format (`read library -liberty <file>.lib`) within Conformal Constraint Designer.

If a Liberty description for certain library cells is not available, you can provide a Verilog library, making sure the cell's definition is surrounded by 'celldefine ... 'endcelldefine directives.

By default, when finding such Verilog library cells, Conformal Constraint Designer will issue an error such as the following:

```
// Error: DIR9.1: Verilog 'celldefine is for timing simulation.
// Use LIBERTY library instead.
// on line xxxx in file 'LIBS/yyy.v'
```

Solution: To avoid this kind of error message, use the following command before reading in the Verilog library:

```
set rule handling -ignore DIR9.1
```

Schematics Viewer Does Not Recognize Double Clicks

Description: When you double click in the Schematic Viewer, the tool recognizes this instead as a single click.

Solution: To work around this:

1. Run `/usr/bin/switchdesk`

2. If your current window manager is the K Desktop Environment (KDE), try switching to the GNOME or TWM window manager.

Conformal Constraint Designer

Clock Objects Not Supported in get_paths from/to Object List

Description: In the current release, the `get_paths from/to` objects list cannot contain clock objects if the command is used in Setup mode. Also, the `-timing` option of this command cannot be used in Setup mode.

Conformal Constraint Designer Might Adjust Clock Period or Waveform Internally for Sequential Proofs

Description: In some cases, the Conformal software might need to adjust the clock period or waveform to obtain a waveform using only integers.

If this happens, the Constraint Designer automatically alters the waveform internally and gives a warning when running the `VALIDATE` command. You can use `REPORT CLOCK` to view the adjusted waveforms.

Note: This only applies to sequential proofs.

Solution: Make sure that the waveform edges and the clock period are all distinct after being rounded to the closest integer. If necessary, manually adjust the clock waveforms. For example, scale the values of all or part of the clocks by a given factor.

Combinational Loops Might Cause Some Rule Checks To Give Inaccurate Results

Description: If a design has combinational loops, even after applying `set_case_analysis` and `set_disable_timing`, the Conformal Constraint Designer cuts them at arbitrarily selected points. If not disabled, the `CCD_EXC_SDT1` rule reports these cases.

When loops are cut, the results of the structural rules that deal with timing exceptions, such as `CCD_EXC_HIER*` and `CCD_EXC_OLP*` rules, can be affected.

Solution: Use `set_case_analysis` and `set_disable_timing` to avoid timing loops. Note, however, that these commands do not affect the structural rule `STRC1.1`.

In any case, Cadence recommends removing any timing loops before running static timing analysis or synthesis tools on the design, because these tools deal with loops by cutting them at arbitrary points, and the choice of cut-points can be tool-dependent.

set_clock_groups Replaces set_false_path Commands

Starting from this release, `set_clock_groups` commands replace `set_false_path` commands specified between clocks that are defined as belonging to different groups. `SDC_LINT_CMD9` reports this event. The `set_false_path` is then ignored everywhere in the tool, and `set_clock_groups` are used to determine which clocks are synchronous.

In this release, Conformal Constraint Design does not consider `set_clock_groups` commands in timing exception structural checks (such as overlap, hierarchical checks, and so on).

SDC Integration: Issue with Timing Exception Leak Check

Description: In SDC integration, if you do not specify SDC design || (glue), the timing exception leak check does not work correctly.

Solution: As a workaround, add the following arguments to the `READ HIERARCHICAL -sdc_design` command:

```
read hierarchical -sdc_design || top.sdc
```

where `top.sdc` is the same file applied to SDC design '/' (top).

Note:

- If SDC design || is already used with its own SDC, no change is needed.
- If SDC design || is not used, then it has to be added to the `READ HIERARCHICAL SDC` command.
- If the SDC design / has an SDC file, read the same file into `||`.
- If the SDC design / has no SDC file, use: `-sdc_design || /dev/null`