Lattice Radiant Software Tutorial for iCE40 UltraPlus

October 9, 2019

Copyright

Trademarks

Disclaimers

Type Conventions Used in This Document

Convention	Meaning or Use
Bold	Items in the user interface that you select or click. Text that you type into the user interface.
<*Italic*>	Variables in commands, code syntax, and path names.
Ctrl+L	Press the two keys at the same time.
`Courier`	Code examples. Messages, reports, and prompts from the software.
`...`	Omitted material in a line of code.
`.` `.` `.`	Omitted lines in code and report examples.
`[ ]`	Optional items in syntax descriptions. In bus specifications, the brackets are required.
`( )`	Grouped items in syntax descriptions.
`{ }`	Repeatable items in syntax descriptions.
`|`	A choice between items in syntax descriptions.

Contents

Lattice Radiant Software Tutorial for iCE40 UltraPlus

The next generation design tool for FPGA design, Lattice Radiant™ software is designed to address the needs of high-density FPGA designs.

This tutorial leads you through all the basic steps of creating, processing, and analyzing VHDL and Verilog designs targeted to the Lattice iCE40 UltraPlus device family. It shows you how to use several processes, tools, and reports from the Radiant software to import sources, run design analysis, and inspect strategy settings. The tutorial then proceeds to step through the processes of examining the device resources, setting timing and location assignments, and editing constraints to configure the settings to implement the design to the target device. Finally, you have the option of loading the design to a development board and observing its operation.

Learning Objectives

When you have completed this tutorial, you should be able to do the following:

- Create a new Radiant software project.
- Create a module using IP Catalog.
- Verify functionality with simulation.
- Inspect strategy settings.
- Examine resources.
- Set timing and location assignments.
- Run synthesis.
- Run map design and check reports.
- Run place and route.
- Examine post place and route results.

- Analyze power consumption.
- Run Export Utility programs.
- Download a bitstream to an FPGA.
- Use Reveal Inserter to add on-chip debug logic.
- Use Reveal Logic Analyzer to perform logic analysis.

Time to Complete This Tutorial

The time to complete this tutorial is approximately 2 hours.

System Requirements

The following is required to complete the tutorial:

- Radiant software
- (Optional) iCE40 UltraPlus Breakout Board to download a bitstream and to insert debug logic using Reveal Inserter. See Figure 23 on page 36.

  Note

  For more information on the iCE40 UltraPlus Breakout Board, go to: http://www.latticesemi.com/en/Products/DevelopmentBoardsAndKits/iCE40UltraPlusBreakoutBoard.aspx

- (Optional) Lattice HW-USBN-2B Download Cable to run Reveal Analyzer. See Figure 30 on page 43.

  Note

  - For more information on the Lattice HW-USBN-2B Download Cable, go to: Programming Cables User Guide.
  - To perform the Reveal Analyzer Task 16, connector pins must be soldered to iCE40 UltraPlus Breakout Board headers. See Figure 30 on page 43.

Accessing Online Help

You can find online help information on any tool included in the tutorial at any time by choosing **Help > Lattice Radiant Software Help**.

Another excellent resource is the *Lattice Radiant Software User Guide*.

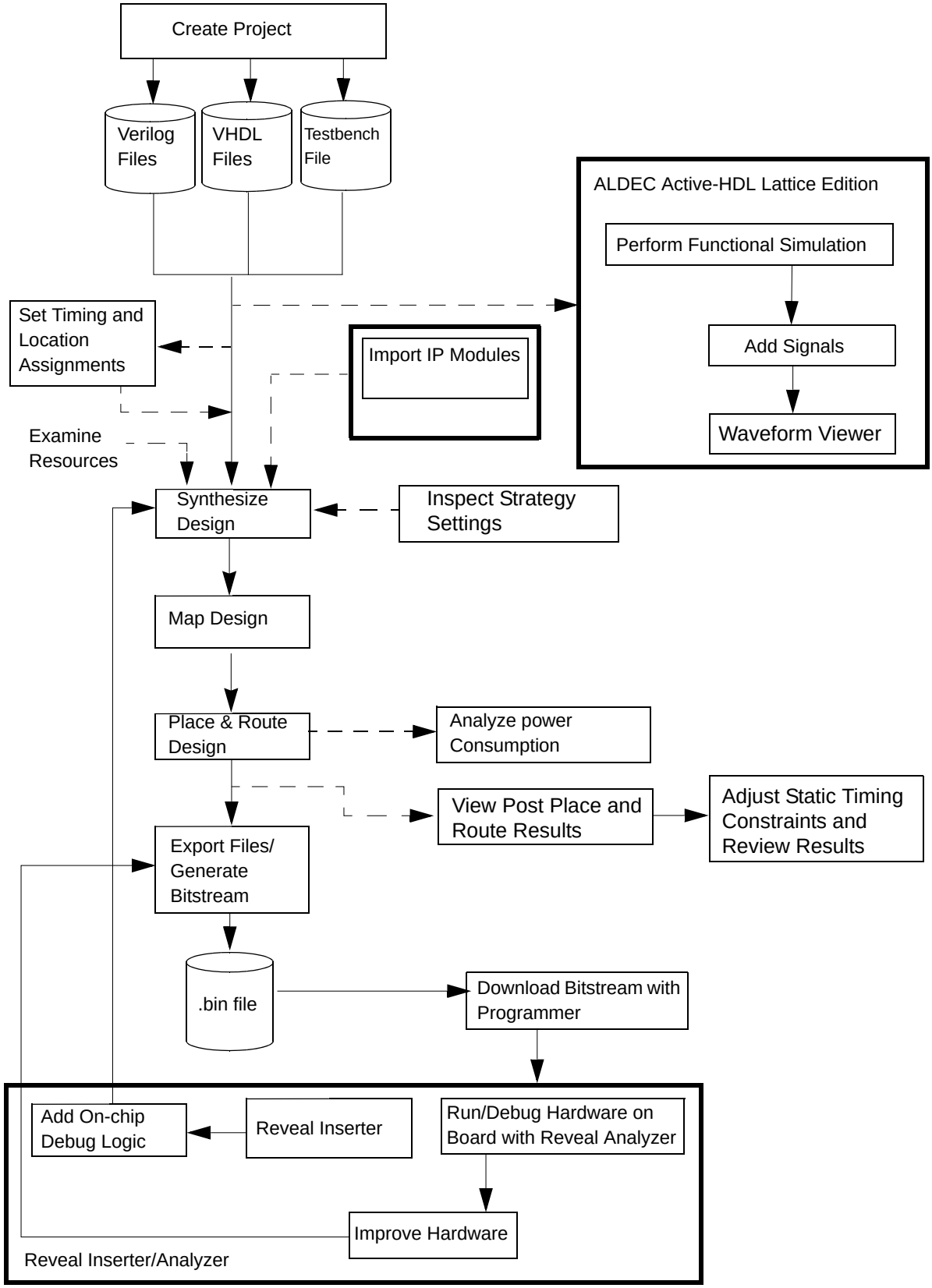
About the Tutorial Design

The design in this tutorial consists of a Verilog HDL module, and a VHDL module. The design that you create is targeted for the iCE40 UltraPlus device.

About the Tutorial Data Flow

Figure 1 illustrates the tutorial data flow through the FPGA design system. You may find it helpful to refer to this diagram as you move through the tutorial tasks.

Figure 1: Tutorial Data Flow



Task 1: Create a New Radiant Software Project

Projects are used to manage input files, constraints, and optimization options related to an FPGA implementation. While there are a number of tasks you can perform independent of a project, most designs start with creating a new project.

Note

Some of the screen captures in this tutorial may have been taken from a version of Radiant software that differs from the one you are using. There may be slight differences in the graphical user interface (GUI), but the software functions the same.

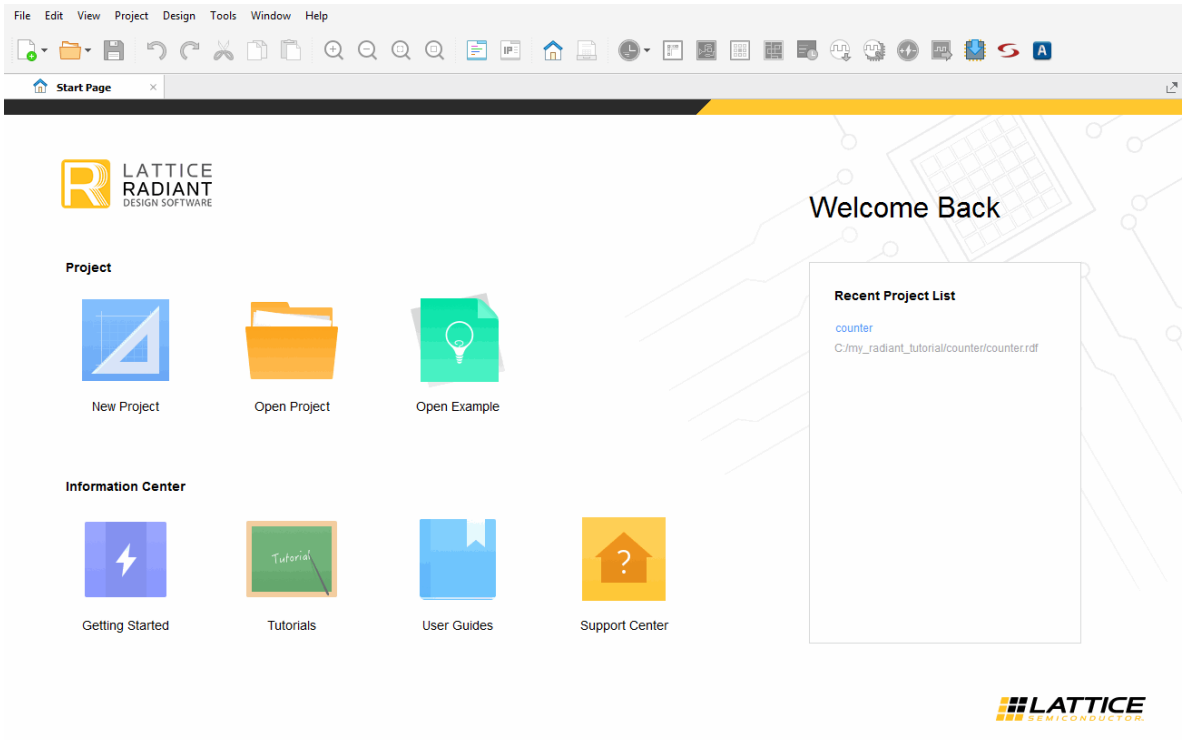
To create a new project:

1. Do one of the following depending on your operating system:
   - On Windows, go to the Start menu and choose **Lattice Radiant Software > Radiant Software**.
   - On Linux, enter the following on a command line:

     <install_path>/bin/lin64/**Radiant**

   The Radiant Software Design Environment appears, as shown in Figure 2.

Figure 2: Radiant Software Design Environment



The initial layout provides the Start Page, which provides a list of common project actions like New Project to run the New Project wizard, Open Project to open a pre-existing project, and Open Example.

For almost all questions, the place to start is the Radiant online Help. It describes the FPGA design flow using the Radiant software, the libraries of logic design elements, and the details of the Radiant software design tools. The Help also provides easy access to many other information sources. The Help can be accessed from **Help > Lattice Radiant Software Help**.

2. Open a new project in one of the following ways:
   - On the Start page, click the **New Project** button.
   - From the Radiant software main window choose **File > New > Project**.

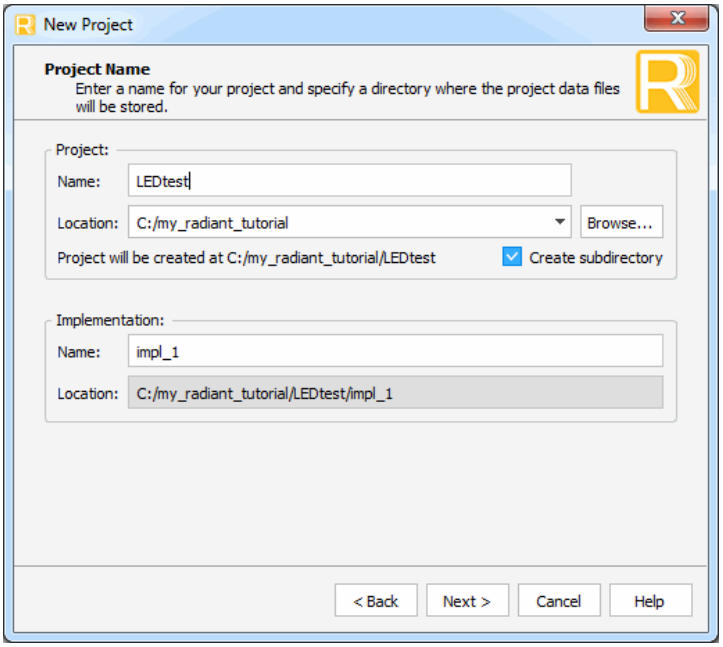
   The New Project wizard opens.
3. Click **Next**.
4. Specify the project name: **LEDtest.**

   Note

   File names for Radiant software projects and project source files must start with a letter (A-Z, a-z) and must contain only alphanumeric characters (A-Z, a-z, 0-9) and underscores (_).

5. Click **Browse**. In the Project Location dialog box, browse to where you want to store the project's files, such as **C:/my_radiant_tutorial**, as shown in Figure 3. Click **Select Folder**.
6. Leave the **Implementation Name** as the default: **impl_1.** The directory to store the implementation is automatically displayed in the Location box. We will talk about creating a new implementation later in this tutorial.

Figure 3: New Project Window



7. Click **Next**.

The Add Source dialog box appears.

8. Click **Add Source**.

   The Import File dialog box appears.

9. Navigate to the folder containing the source files, which are located in the <*radiant_install_directory*>/docs/tutorial/**Radiant_tutorial** directory (typically C:/lscc/radiant/<*version_number*>/docs/tutorial/Radiant_tutorial). Select the following files in the directory:
   - LED_control.v
   - rgb_led_top.vhd
   - testbench.v
10. Click **Open**.

    The Add Source step of the Wizard appears with all the selected source files added.

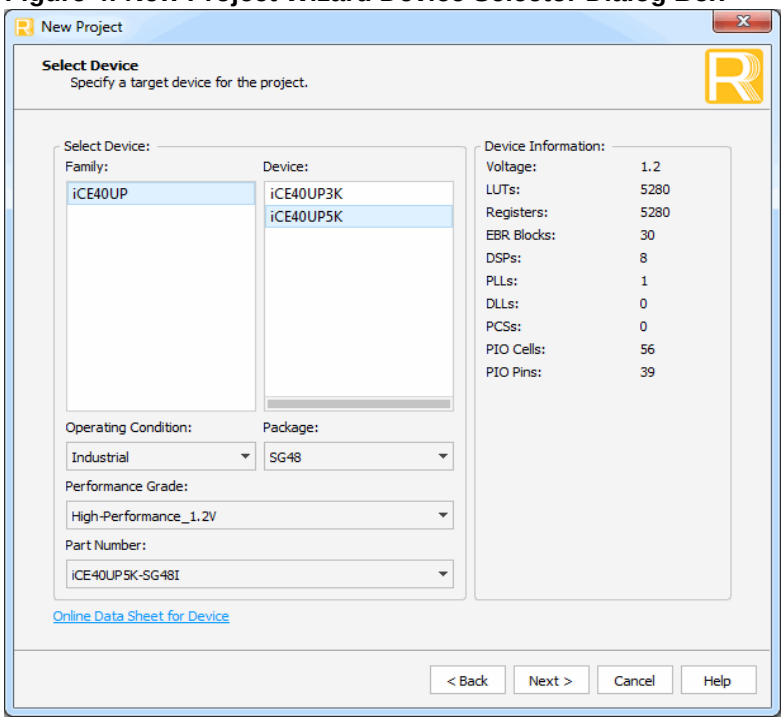
11. Select **Copy source to implementation directory.**
12. Clear **Create empty constraint files**. This option is not required for this tutorial.
13. Click **Next**.

    The Select Device dialog box appears.

14. Select the following device options:
    - Family: **iCE40UP**
    - Device: **iCE40UP5K**
    - Operating Condition: **Industrial**
    - Package: **SG48**
    - Performance Grade: **High-Performance_1.2V**

    The dialog box should resemble Figure 4. The Device Information box on the right side shows (as applicable) available Voltage, LUTs, Registers, EBR Blocks, DSPs, PLLs, DLLs, PCSs, PIO Cells, and PIO Pins.

Figure 4: New Project Wizard Device Selector Dialog Box



15. Click **Next.**

    The Select Synthesis Tool dialog box opens.

16. Select **Lattice LSE**.

17. Click **Next.**

    The Project Information dialog box appears. The project information includes project name, location, implementation name, device, synthesis tool, and import source.

18. Click **Finish**.

    The File List view (left side of screen) is populated and the Reports view (right side of screen) appears.

    The File List view displays the components of the project. File List organizes project files by categories: Strategies, and Implementation including Input Files, Constraint Files, Debug Files, Script Files, Analysis Files, and Programming Files. You may adjust the file order by dragging and dropping file names in the list. Properties of each file are accessed by right-clicking the file and choosing Properties from the pop-up menu.

    Note

    You can also see Area and Timing listed in the Strategies folder in the File List view. These are predefined strategies supplied by Lattice Semiconductor that solve particular design requirements. For details of these predefined strategies, refer to the Radiant online Help.

19. In the File List view, right-click **Pre-Synthesis Constraint Files** and choose **Add > Existing File**.

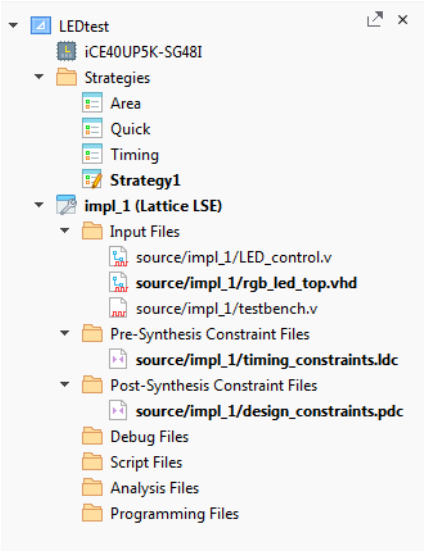
The Add Existing File dialog box appears.

20. Browse to <*radiant_install_directory*>/docs/tutorial/Radiant_tutorial and select the file **timing_constraints.ldc**. Select **Copy file to directory**, and click **Add**.
21. In the File List, right-click **Post-Synthesis Constraint Files** and choose **Add > Existing File**.

    Add Existing File dialog box appears.
22. Browse to <*radiant_install_directory*>/docs/tutorial/Radiant_tutorial and select the file **design_constraints.pdc**. Select **Copy file to directory**, and click **Add**.
23. In the File List, right-click **testbench.v** and choose **Include for** > **Simulation.**

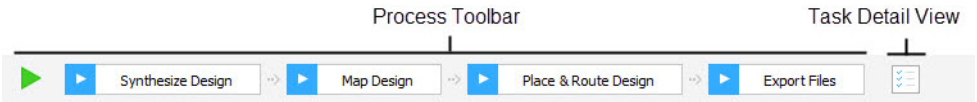
    The File List should appear as shown in Figure 5.

Figure 5: File List View



24. The Process Toolbar, shown in Figure 6, lists all the processes available, such as Synthesize Design, Map Design, Place & Route Design, and Export Files. A process is a specific task in the overall processing of a source or project. You can view the available processes for a design in the Process Toolbar.
25. Click Task Detail View to see detailed information of the processes.

Figure 6: Process Toolbar and Task Detail View



The Reports view allows you to examine and print process reports. There are two panes in the Reports view. The left pane lists the reports. The right pane displays the reports.

Log messages are displayed in the Output frame of the Radiant software's main window.

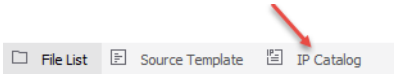
Task 2: Create a PLL Using IP Catalog

IP Catalog is an easy way to use a collection of modules from Lattice Semiconductor. With IP Catalog, these modules can be extensively customized. They can be created as part of a specific project or as a library for multiple projects.

In this task, you will generate a phase lock loop (PLL) module to import into your design.

To generate a PLL:

1. Open IP Catalog. IP Catalog is accessed via a tab at the lower left of the Radiant software. Click the tab to view the list of available modules and IP.

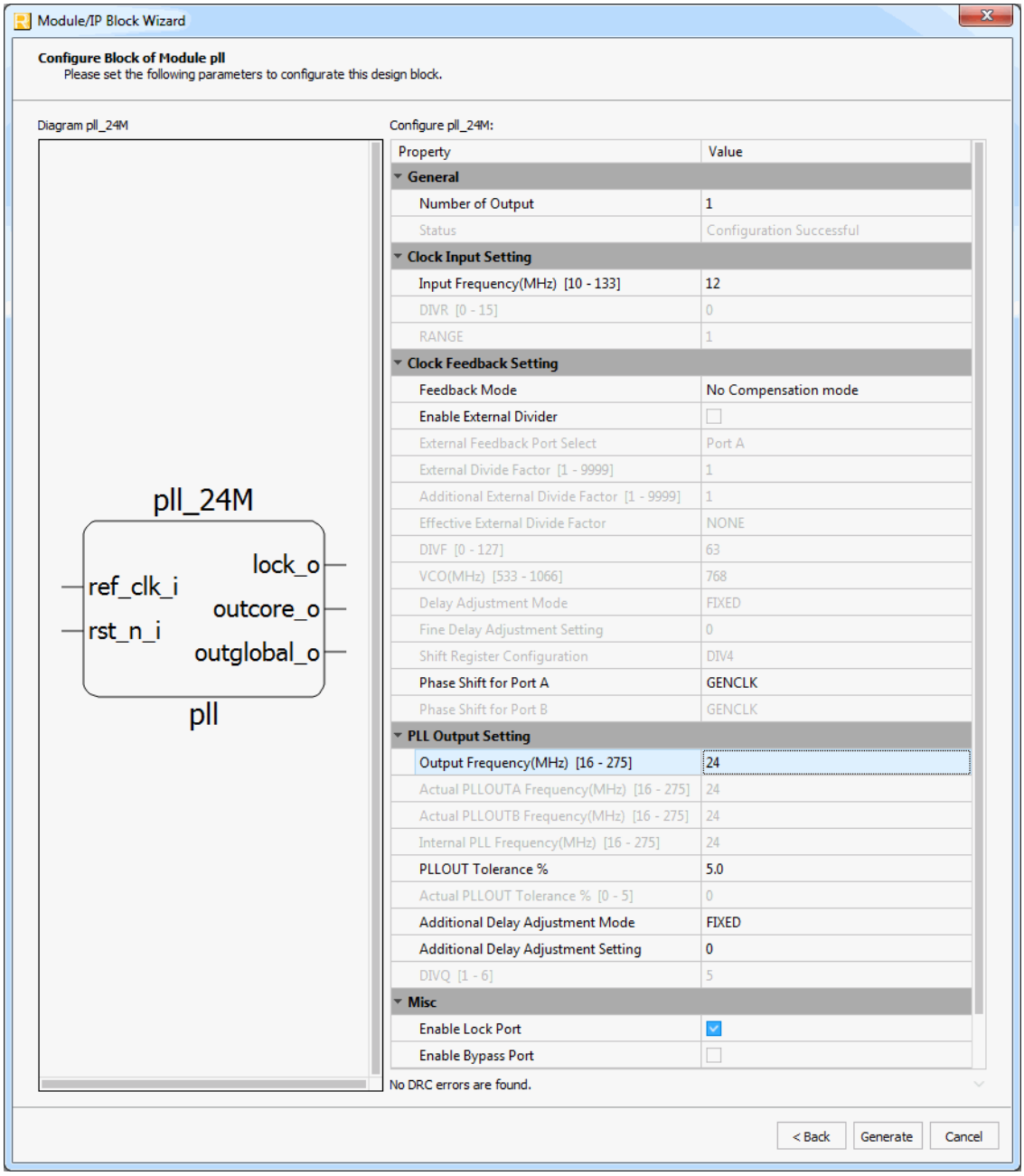


2. In the Module/IP on Local tree, open **Module > Architecture_Modules**, and double-click **PLL**.
3. In the Module/IP Block Wizard, specify general project information and the base file name for the module or IP.
   - **Instance Name** is the base name for the module's files (that is, with no extension). For this tutorial enter the name **pll_24M**.
   - The **Create In** box is the location for the customized module's files. For this tutorial, use the default directory.
4. Click **Next.**

   The Module/IP Block Wizard dialog box opens.
5. In the dialog box, set the following options:
   - Input Frequency: **12**
   - Desired Frequency: **24**
   - Check **Enable Lock Port**

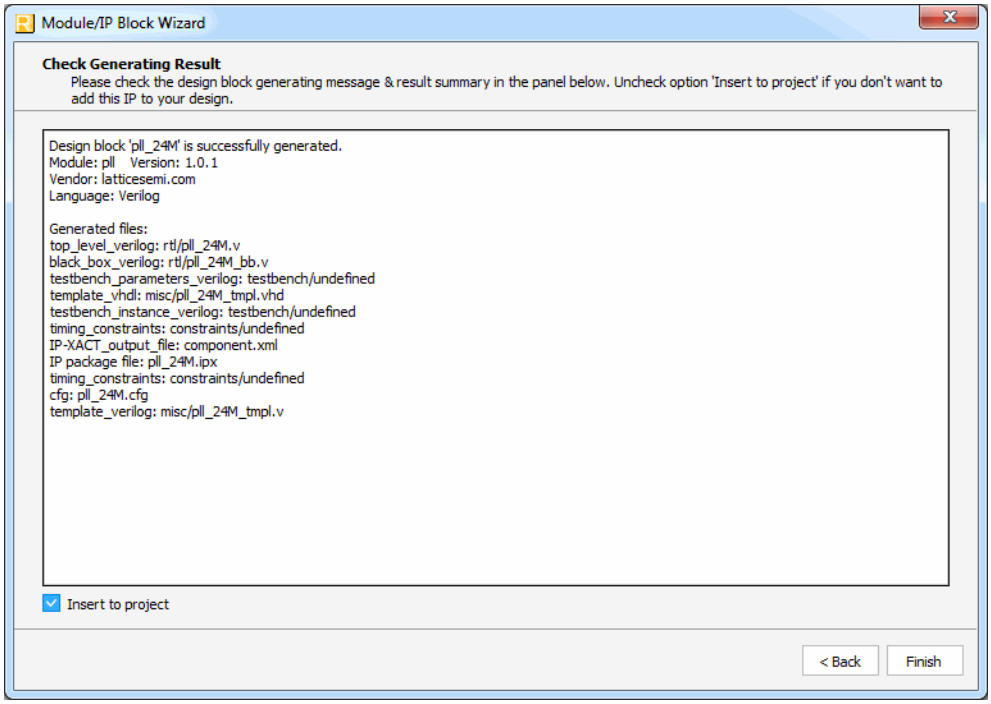
   Leave all other values as default. The module dialog box should appear as shown in Figure 7**.**

Figure 7: Configuration Block of Module PLL



6. Click **Generate**. The Check Generating Result dialog box should appear as shown in Figure 8.

Figure 8: Check Generating Result Dialog Box



7. Ensure that **Insert to project** is selected.
8. Click **Finish**.

Task 3: Verify Functionality with Simulation

The Radiant software provides an interface to create a new simulation project file that can be imported into a standalone simulator. The Radiant software can export Active-HDL and ModelSim® simulation files.

Aldec® Active-HDL™ is an integrated environment designed for simulation of VHDL, Verilog/SystemVerilog, EDIF, and SystemC designs.

In this task, you will simulate the design using Active-HDL and analyze the resulting waveforms.

To simulate the design:

1. Make sure all the Source files are included in the simulation.
   a. In the File List pane, under **Input Files**, select source files LED_control.v and rgb_led_top.vhd using the **Shift** or **Ctrl** keys. Right-click and select **Include For > Synthesis and Simulation**.
   b. In the File List pane, under **Input Files**, right-click on source file pll_24M.ipx, and select **Include For > Synthesis and Simulation**.

Note

The testbench.v file should be included for **Simulation** only.

2. Choose **Tools** > **Simulation Wizard** or click on the Radiant software toolbar.

   The Simulation Wizard dialog box appears.

3. Click **Next.**

   The Simulator Project Name dialog box appears.

4. Perform the following:

   - Specify Project name: **simulationfile**.
   - Click “Browse...” button to browse to where you want to store the project’s file. For this tutorial use default directory C:/my_radiant_tutorial/LEDtest.
   - Select **Active-HDL** is the Simulator dialog box.
   - Select **RTL** in the Process Stage dialog box.

5. Click **Next**.

6. If you left the default for the project location, a dialog box opens saying, “simulationfile does not exist. Do you want to create it?” Click **Yes**.

   The Add and Reorder Source dialog box appears. Make sure all source files are present in the Source Files list. Leave the **Automatically set simulation compilation file order** option selected.

7. Click **Next**.

   The Parse HDL Files for Simulation dialog box appears.

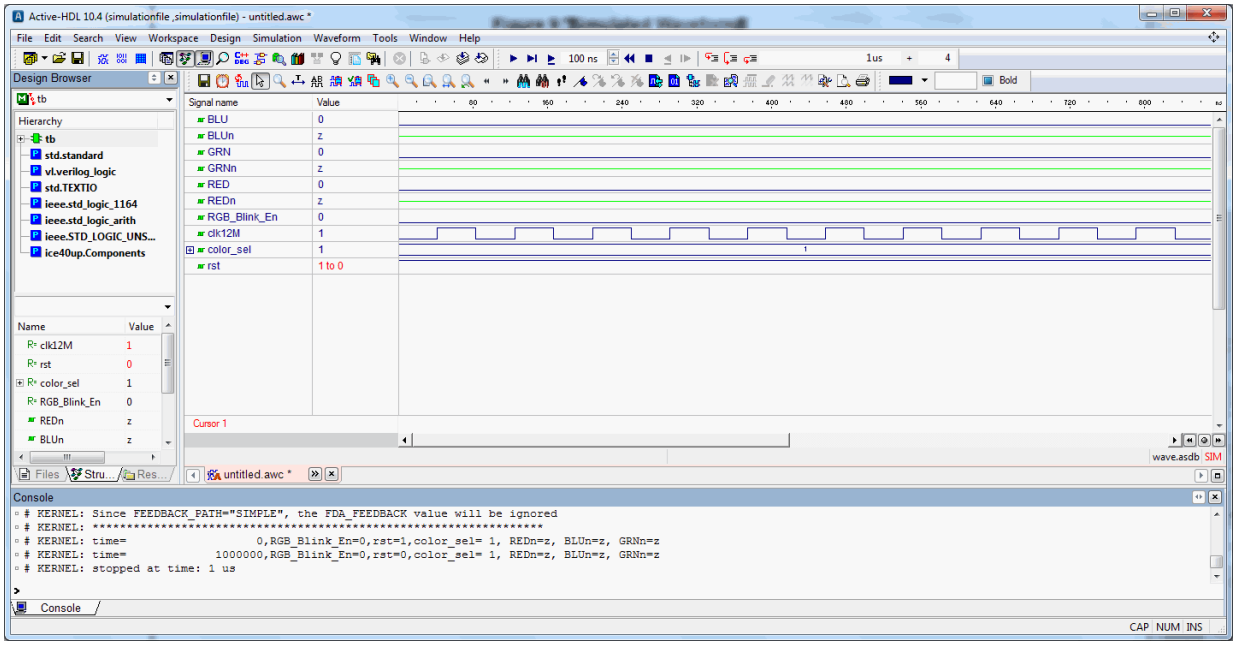
8. Click **Next.**

   The Summary dialog box appears. Make sure that **Run simulator**, **Add top-level signals to waveform display**, and **Run simulation** are all selected.

9. Click **Finish**.

   The Aldec Active-HDL software launches and the simulation starts automatically. After completing the simulation, the waveform appears, as shown in Figure 9.

Figure 9: Simulated Waveform



10. In Active-HDL, click **Simulation > Run** or click ▶ to run the simulator for some more time. Simulate the design until it stops automatically.
11. In Active-HDL, use the zoom tools (**View > Zoom**) to see how the waveform is changing.
12. Choose **File > Exit** to close Active-HDL.

    The Design Browser dialog box appears.
13. Click **OK** to stop the simulation.

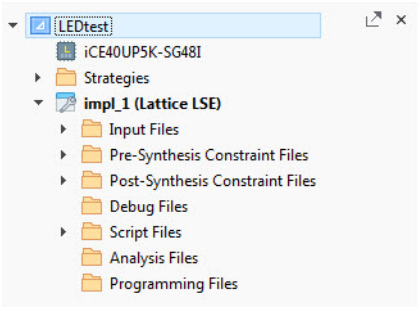
    The Save File dialog box appears.
14. Click **No**.

Task 4: Inspect Strategy Settings

Implementations define the design structural elements for a project, including source code, constraint files, and debug insertion. Implementation contains all source files, constraint files, debug files, scripts, and analysis files.

A strategy is a collection of settings for controlling the different stages of the implementation process (synthesis, map, place & route, and so on). Strategies can control whether the design is optimized for area or speed, how long place and route takes, and many other factors. The Radiant software provides a default strategy, which may be a good collection to start with, and some variations that you can try. You can modify Strategy1, shown in the Strategies section in Figure 10, and create other strategies to experiment with or to use in different circumstances. Predefined strategies can also be cloned and then modified.

Figure 10: File List View, including Implementations and Strategies



- LEDtest: Project Name
- iCE40UP5K-SG48I: Device Part Number
- Strategies: Strategy Types
- impl_1 (Lattice LSE): Implementation (Synthesis Tool)
- Input Files: Input File Library
- Pre-Synthesis Constraint Files: Pre-Synthesis Constraint File Library
- Post-Synthesis Constraint Files: Post-Synthesis Constraint File Library
- Debug Files: Used for Reveal debugging
- Script Files: Used for Simulation Wizard
- Analysis Files: Used for Power Calculator and Timing Wizard
- Programming Files: Used for programming a device

To view synthesis settings:

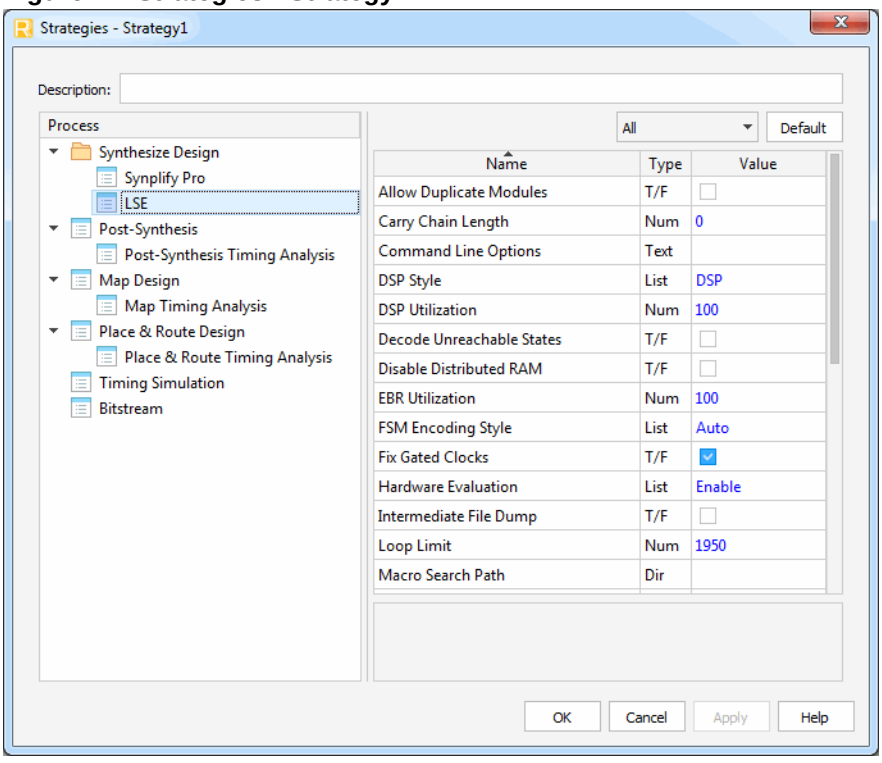
1. In the File List view, double-click **Strategy1**.

   The Strategies - Strategy1 dialog box**,** shown in Figure 11**,** appears.

2. Click **Synthesize Design > LSE**.

   A set of default global synthesis timing constraints and optimization settings appears in the panel. LSE settings are displayed as the default in the dialog box.

Figure 11: Strategies - Strategy 1



   When each option is selected, descriptive text appears in the lower panel of the dialog box. Default values in the strategies dialog box are shown in blue while changed values are shown in black.

3. Click **OK** to close the Strategies dialog box.

Task 5: Set Timing and Location Assignments

Timing and location assignments constrain logic synthesis, as well as backend map, and place-and-route programs to help meet your design requirements. A well-constrained design helps optimization algorithms work as efficiently as possible. In this section you'll set default timing constraints for the operating frequency and I/O timing then assign package pins to specific I/O signals.

To set timing constraints:

1. From the Tools menu, choose **Timing Constraints Editor > Pre-Synthesis Timing Constraint Editor**, or click the arrow in the icon on the Radiant software toolbar, and from the pull-down menu, choose **Pre-Synthesis Timing Constraint Editor.**

   The Pre-Synthesis Timing Constraint Editor appears. The top half is a spreadsheet with a row of tabs beneath it. The bottom half is a box where the constraint text appears.

2. If you see a yellow bar with a message saying the “Design database in memory is outdated,” click **Reset Database**, which is to the right of the message.

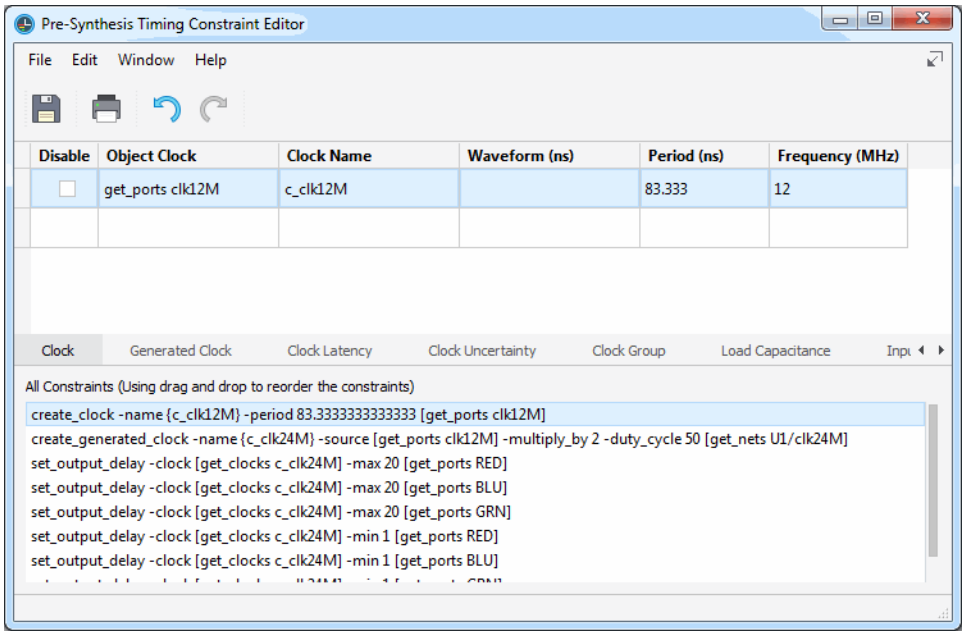
3. Click **Detach Tool** icon at the upper-right corner to detach the tool.

4. Click the **Clock** tab.

5. Click in the **Frequency** cell, and enter **12** as shown in Figure 12.

   The Period column is recalculated and changes to 83.333. Note that in the text part of the view, the period also changes in the create_clock constraint.

Figure 12: Pre-Synthesis Timing Constraint Editor



6. Choose **File > Save timing_constraints.ldc.**
7. Close the Pre-Synthesis Constraint Editor.
8. In the File List view of the Radiant software main window, double-click the **timing_constraints.ldc** file under Synthesis Constraint Files.

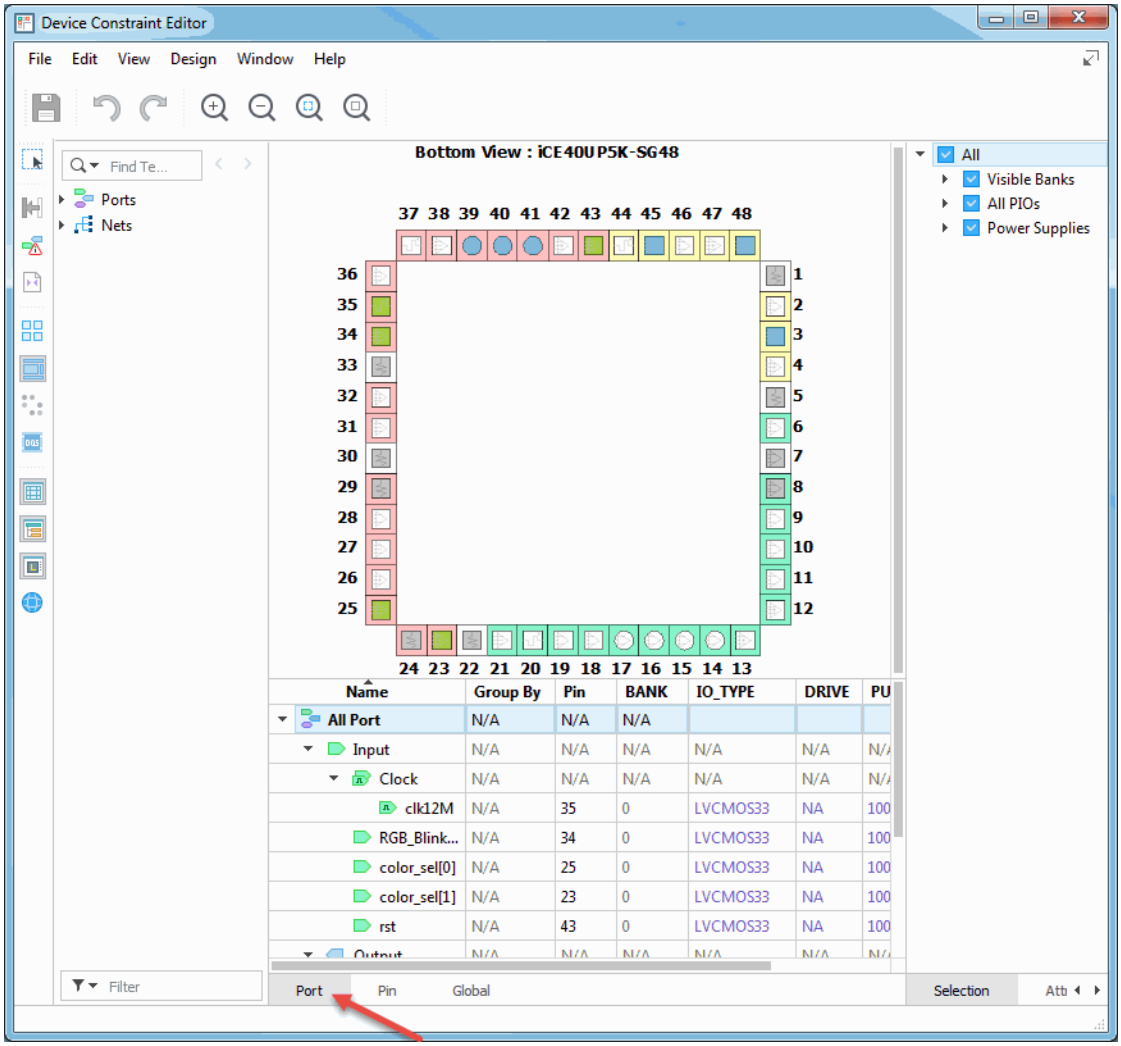
   Source Editor appears with the .ldc file. Note the timing constraints defined so far.
9. Close the Source Editor.

To set location constraints:

1. From Tools menu, choose **Tools > Device Constraint Editor** or click on the Radiant software toolbar.

   Device Constraint Editor appears.
2. Click **Detach Tool** icon at the upper-right corner to detach the tool as shown in Figure 13.

Figure 13: Device Constraint Editor



3. Select the **Port** tab, in the lower-left of Device Constraint Editor.

   Cell entries in Device Constraint Editor are color-coded to indicate the source of a constraint setting:

   - Black - User-defined setting.
   - Blue - Default.
   - Green - Implied by the synthesis-defined constraint setting.
   - Yellow - Implied by another user-defined setting.

4. Close Device Constraint Editor.

Task 6: Run Synthesis Process

Synthesis is the process of translating a register-transfer-level design into a process-specific, gate-level netlist that is optimized for Lattice Semiconductor FPGAs. The Radiant software can be used with almost any synthesis tool. The Radiant software comes with two tools fully integrated: Synopsys Synplify Pro for Lattice and Lattice Synthesis Engine (LSE). “Fully integrated” means that you can set options and run synthesis entirely from within the Radiant software.

You will use LSE to synthesize your design for the iCE40 UltraPlus FPGA.

To synthesize the design and examine resource utilization:

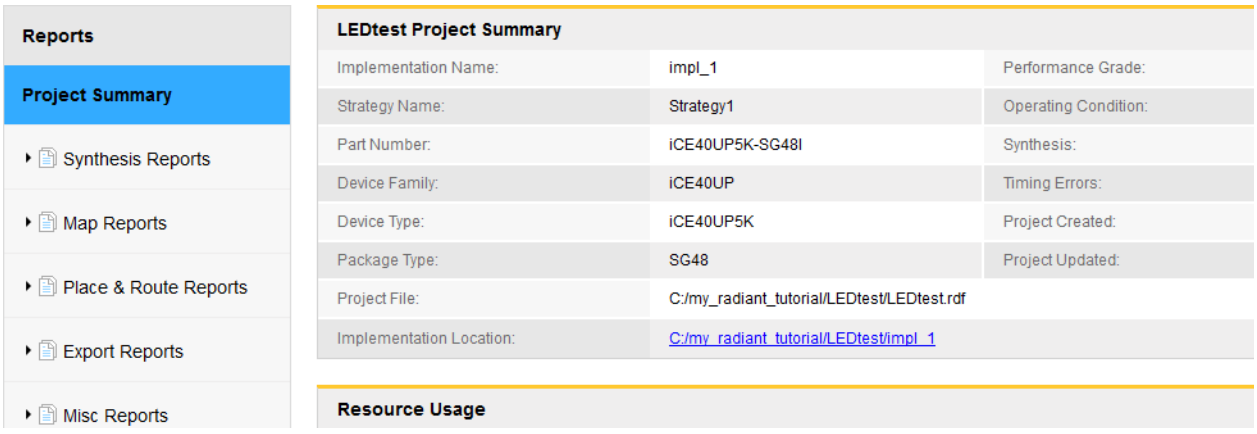
1. From the Process Toolbar, click **Synthesize Design**.

   Task Detail View opens and tracks completion of the processes.

2. When the system is finished, take a look at the icon next to Synthesize Design. A green check mark indicates success; a red X indicates failure.

3. To view post-synthesis resource usage, click **View > Reports**, or click the **Reports** tab. The Synthesis report is displayed, as shown in Figure 14. Note that your Synthesis report results may differ slightly from what is shown in the example.

Figure 14: Synthesis Report



The Resource Usage table displays the number of logical resources within each level of the design.

- LUT4 represents the total LUT4 count utilization throughout the design.
- PFU Registers represents the total PFU register utilization throughout the design.

Task 7: Run Map Design and Check Reports

Timing and location assignments constrain logic synthesis, map, and place-and-route programs to help meet your design requirements.

A well-constrained design helps optimization algorithms work as efficiently as possible. In this section, you'll set default timing constraints for the operating frequency and then assign package pins to specific I/O signals.

To run the Map process:

1. From the Process Toolbar, open Task Detail View and double-click **Map Design**.

   The batch interface to logic synthesis and the design mapper run. When finished, check the icon next to Map Design. A green check mark indicates success; a red X indicates failure.

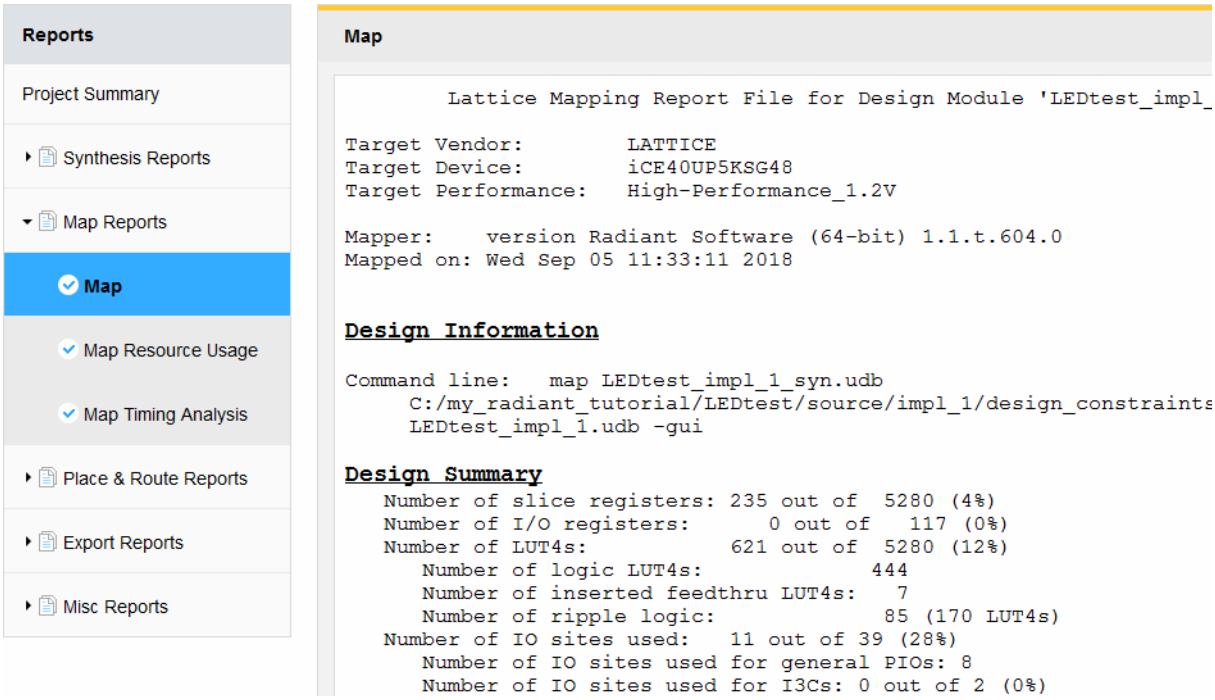
   Report files appear in the Reports view. To view each process report, select the process in the **Project Summary** pane. Reports include Synthesis Reports, Map Reports, Place & Route Reports, Export Reports, and Misc Reports. Additional reports are available within each category.

2. From **Project Summary**, choose **Map Reports > Map.**

   The Map Report, appears in the right panel as shown in Figure 15, Hover your cursor over **Contents** to display the MAP Report table of contents.

   In Project Summary, if a report has been generated, the icon appears as . If the report has not yet been generated, the icon appears as . Use the scroll bar to navigate through the report. Some of the reports are divided into sections.

Figure 15: MAP Report



Task 8: Running Place and Route

Use the Process Toolbar to run the Place & Route Design process stages.

To run place and route:

1. From the Process Toolbar click **Place & Route Design**.

   The place and route tools run. Intermediate results appear in the Output frame of the Radiant main window.

2. From the Reports tab, you can directly click on Place & Route Report. You will find a Blue check mark appears before the reports generated successfully. Expand the **Place & Route Report** section. Select **Place & Route**.

   Details about Place & Route appear in the right pane of the Reports view.

3. From **Place & Route Report**, select **Place & Route Timing Analysis**.

   The Place & Route Timing Analysis Report appears in the right pane of the Reports view.

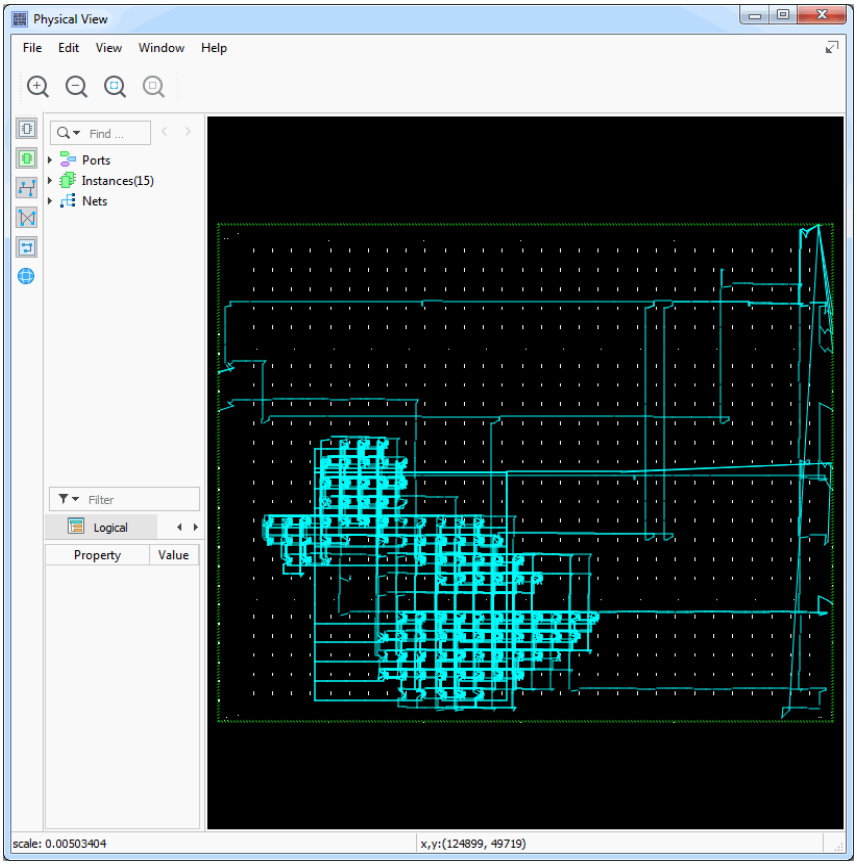
4. Choose **Tools > Physical View** or click on the Radiant software toolbar.

   The Physical View appears.

5. Click **Detach Tool** icon at the upper-right corner to detach the tool as shown in Figure 16.

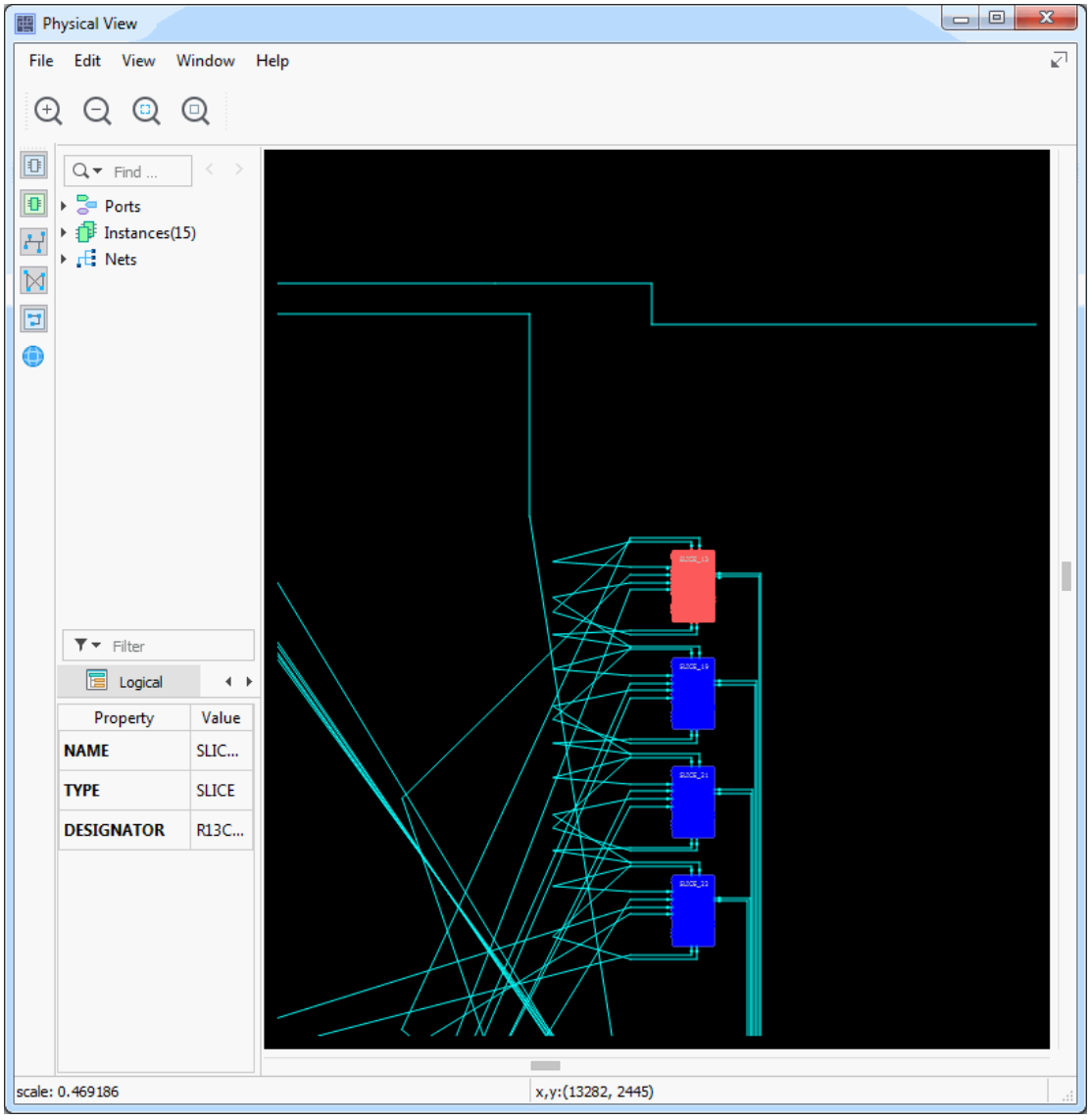
Physical View provides a read-only detailed layout of your design that includes switch boxes and physical wire connections.

Figure 16: Physical View



6. To zoom into a component:
   - Click the name of the component in the list to the left of the diagram.
   - Magnify the surrounding area by clicking and dragging a box around it from left to right.
   - Click the button.
7. In the diagram, right-click on the component and choose **Show in > Floorplan View**, as shown in Figure 17, to display the Floorplan View.

Figure 17: Physical View - Right-Click - Show in > Floorplan View

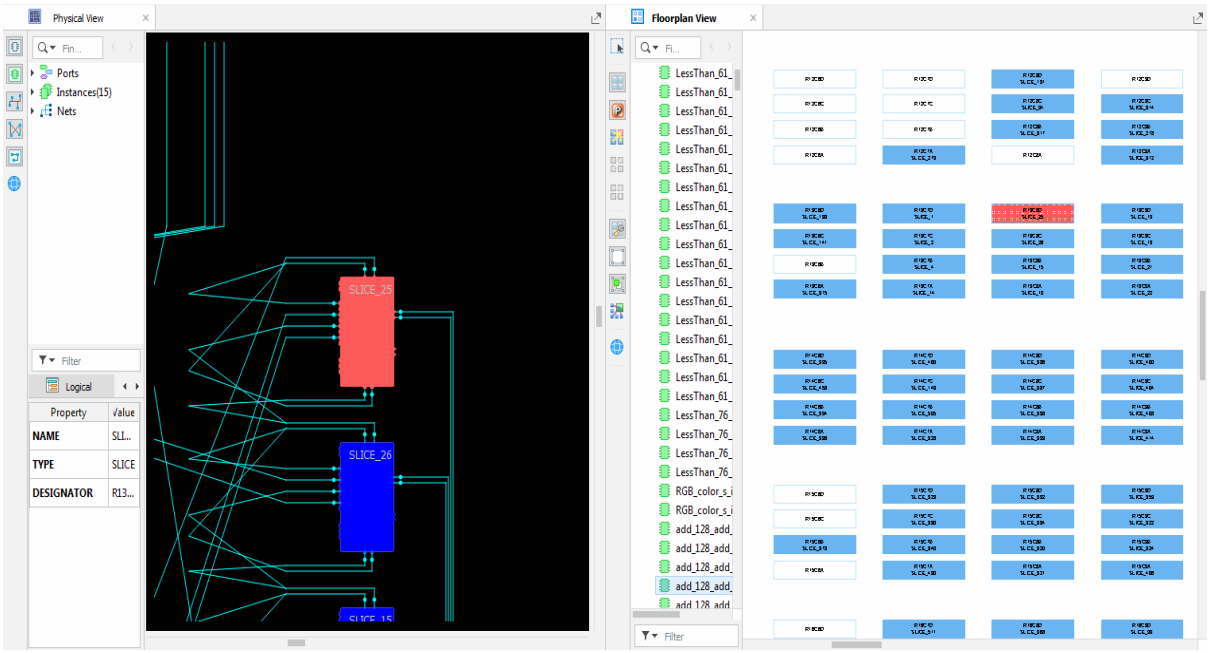


8. To easily see cross-probing between Floorplan View and Physical View, ensure both views are attached to the Radiant main window and then right-click on the Floorplan View tab and select **Split Tab Group**.

   The two views display in parallel, as shown in Figure 18.

   When both Floorplan View and Physical View are open, an item that you select in one of these views is automatically selected in the other. Cross-probing is especially useful for immediate examination of connections in both views.

Figure 18: Cross Probing



9. Right-click on the Floorplan View tab and choose **Move to Another Tab Group**.

   Now both tabs are merged into a single group as before.

10. Close Floorplan View and Physical View tabs.

Task 9: Examine Post Place and Route Results

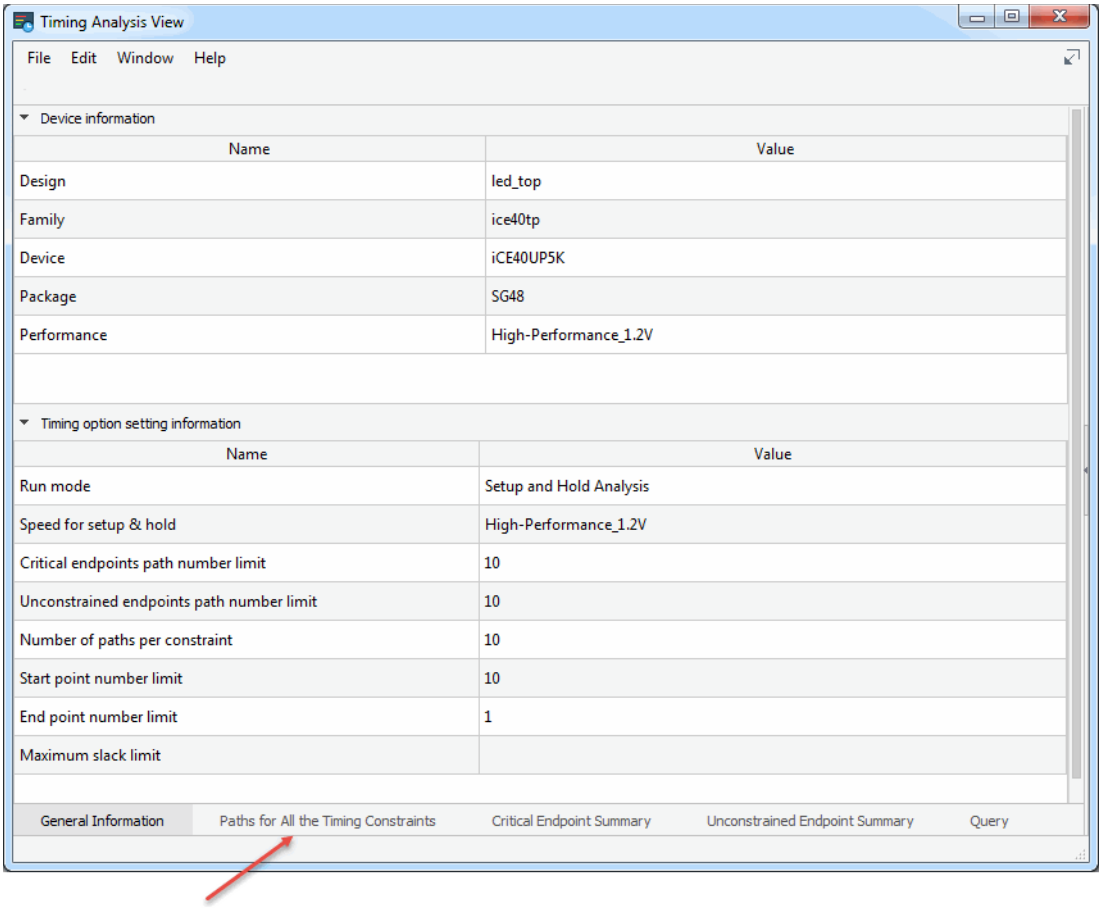
Static timing analysis can determine if your circuit design meets timing constraints. Rather than simulation, it employs conservative modeling of gate and interconnect delays that reflect different ranges of operating conditions on various dies, providing complete verification coverage.

To examine timing analysis results:

1. Choose **Tools > Timing Analysis View**, or click on the Radiant software toolbar.

   Timing Analysis View appears as shown in Figure 19.

Figure 19: Timing Analysis View



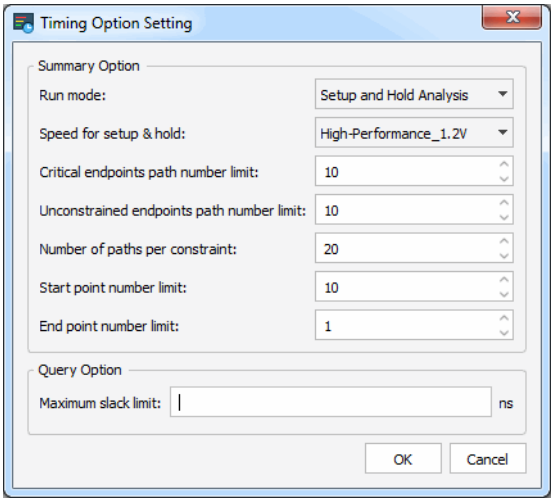
2. Click the **Paths for All the Timing Constraints** tab (on the lower-left of the Timing Analysis view).
3. Select **create_generated_clock**.

   The Paths summary in the center of Timing Analysis View is populated with Start Point, End Point, Setup/Hold Constraint, Slack, Delay, Source Clock, Destination Clock, and Analysis Type.
4. Select Row 1 of the Paths summary.

   The Path Detail tab in the right pane is populated with details.
5. Choose **Edit > Timing Option Setting**.

   The Timing Option Setting dialog box appears, as shown in Figure 20.

Figure 20: Timing Setting Dialog Box



6. Enter **20** into the “Number of paths per constraint” field.
7. Click **OK**.

   Timing Analysis View is cleared.
8. Select **create_generated_clock** again.

   The Paths summary is populated with the additional path data.
9. Close Timing Analysis View.

Task 10: Analyze Power Consumption

Included with the Radiant software is Power Calculator, which estimates the power dissipation for a given design. Power Calculator uses parameters such as voltage, temperature, process variations, air flow, heat sink, resource utilization, activity, and frequency to calculate the device’s static and dynamic power consumption.

To analyze power consumption:

1. Choose **Tools > Power Calculator** or click on the Radiant software toolbar.

   Power Calculator opens in calculation mode.

   Power Calculator provides two modes for reporting power consumption:

   - Estimation Mode:

     In estimation mode, Power Calculator provides estimates of power consumption based on the device resources or template that you provide. This mode enables you to estimate the power consumption for your design before the design is complete or even started.
   - Calculation Mode:

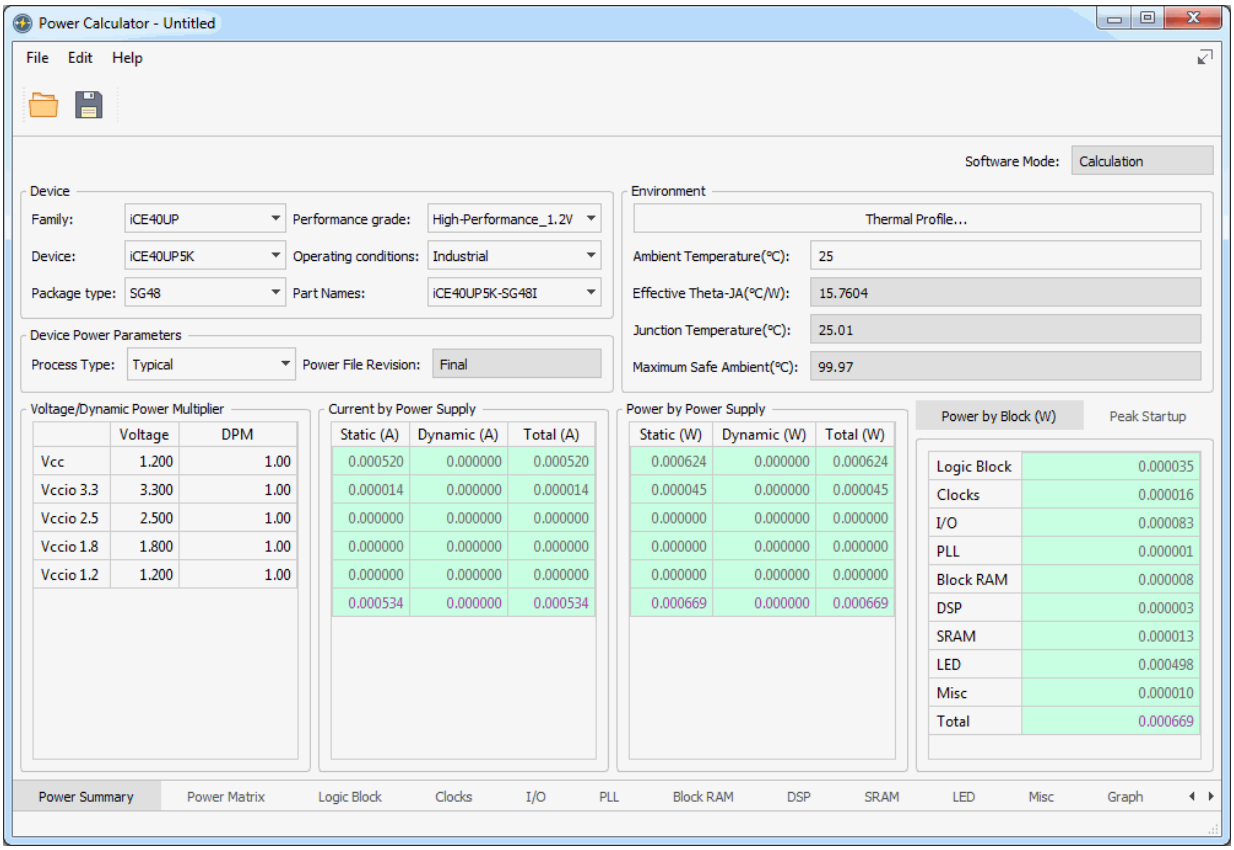
     In calculation mode, Power Calculator calculates power consumption on the basis of device resources taken from a design’s .udb file, or

from an external file such as a value change dump (.vcd) file, after placement and routing. This mode is intended for accurate calculation of power consumption, because it is based on the actual device utilization.

Editing data in white cells, such as voltage, frequency, activity factor, and thermal data, does not change mode. Editing data in yellow cells, such as design data, will change calculation mode to estimation mode.

2. Click **Detach Tool** icon at the upper-right corner to detach Power Calculator from the Radiant software main window, as shown in Figure 21.

Figure 21: Power Calculator



3. In the Device Power Parameters section select the following parameter:

   Process Type: **Worst**.

4. Click the **Thermal Profile** button in the Environment section.

   The Power Calculator – Thermal Profile dialog box appears.

5. In the Board Selection section, choose the following parameter: **Small board**.

6. Click **OK**.

   After a short while the new temperature results become available in the Environment section.

In the title bar of Power Calculator, “Untitled” appears with an asterisk, which indicates an in-memory change to the timing constraints. You can save the change to a Power Calculator File (.pcf) by choosing **File > Save File As** and giving it a name and location. The .pcf file will then appear in the Analysis Files folder of the File List Pane.

7. Close Power Calculator. If you chose not to save in the previous step, a Save dialog box will now appear. Click **No** to discard the change.

Task 11: Run Export Utility Programs

Use the Process Toolbar to generate files for exporting. One of the files exported will be a bitstream file (.bin) which will be used to program an iCE40 UltraPlus device in the next task.

1. Click the Task Detail View to see detailed information of the processes.
2. Under **Export Files**, ensure the following are selected:

   IBIS Model

   Gate-Level Simulation File
3. Click the **Run** button on the Process Toolbar.

   The Radiant software generates the selected files and saves them in your project directory.

Task 12: Download a Bitstream to an FPGA

This task requires that you have an iCE40 UltraPlus Breakout Board. In the previous section, you generated export files including a bitstream file (.bin). In this section, you will use the Radiant software Programmer to download a bitstream to a iCE40 UltraPlus FPGA mounted on a board.

This tutorial design displays RGB colors on iCE40 UltraPlus Breakout Board with an iCE40UP device. The colors are selected using the switches, controlled by the color_sel [1:0] signal. Switches allow you to control the Brightness, Blink Rate, Breathiness and Blink Enable signals and display RGB color on the board LED.

For this design we have selected 50 percent Brightness, 2x Breath Ramp and 1sec Blink Rate.

The top level design is provided with a 12-MHz clock from an on-board oscillator which is given to a PLL to be multiplied to a 24-MHz clock. The design is running at 24 MHz.

The LED_control module generates Pulse Width Modulated (PWM) waves and these modulated waves are driven by an RGB driver to keep the RGB LED continuously on or blinking. This blinking capability depends on

RGB_Blink_En signal. If the RGB_Blink_En signal is enabled, the RGB LED blinks. Otherwise it is continuously on. RGB_Blink_En is active high.

The following block diagram shows how the RGB LED is controlled.

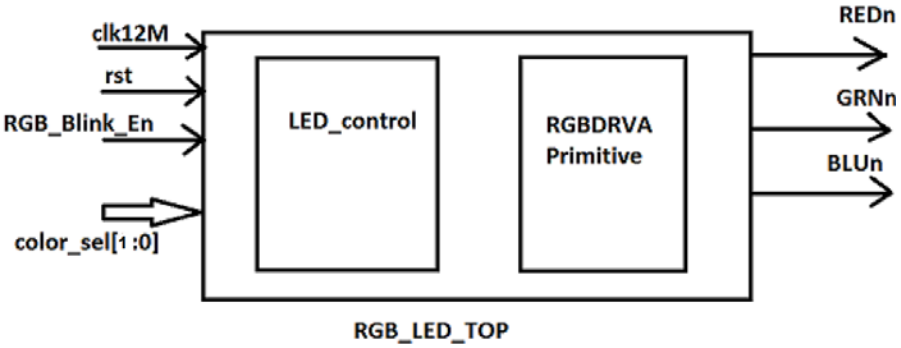


Table 1 describes the ports and signals of the design.

Table 1: Ports and Signals

Port	Signal	Description
Input Ports	clk12M	12-MHz on-board oscillator given to the top module.
	rst	Reset
	RGB_Blink_En	Enables blinking of the RGB LED.
	color_sel[1:0]	Selects the color to be displayed.
Output Ports	REDn	Drives RED color.
	GRNn	Drives GREEN color.
	BLUn	Drives BLUE color.

Ensure shunts on the board are configured as shown in Figure 22. For more information, refer to the *iCE40 UltraPlus Breakout Board User Guide*.

Figure 22: iCE40 UltraPlus Breakout Board Shut Positions

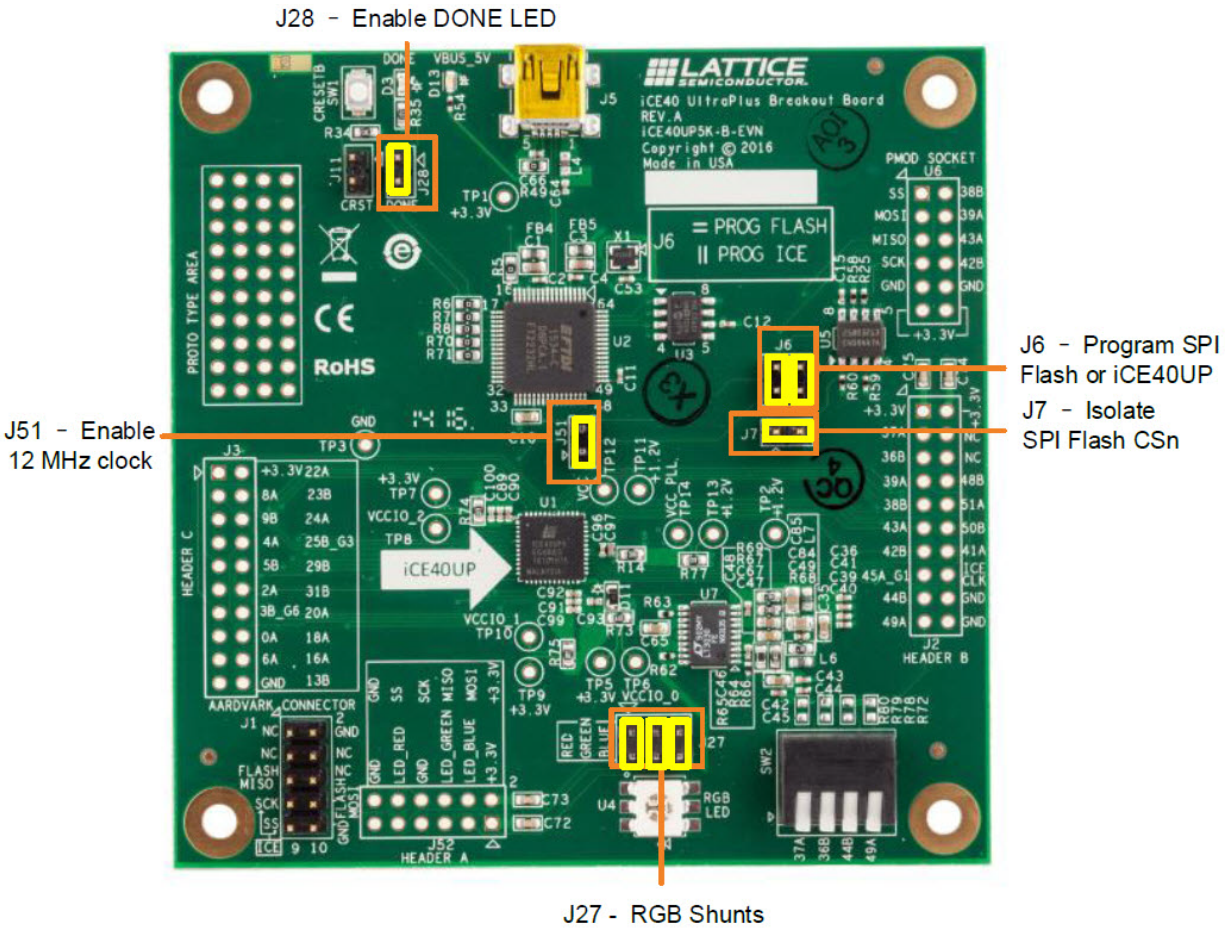
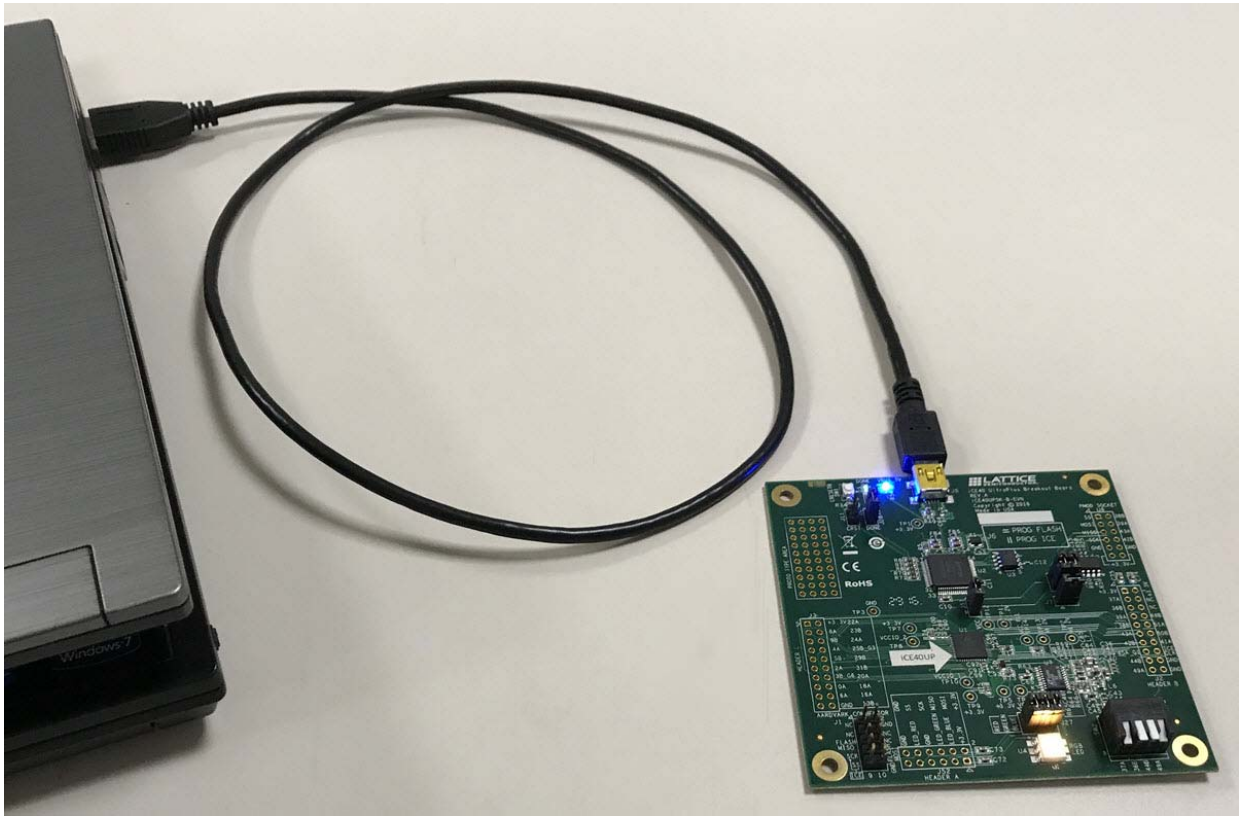


Figure 23 shows the iCE40 UltraPlus Breakout Board connected to the computer via USB cable.

Figure 23: iCE40 UltraPlus Breakout Board Connected to Computer with USB Cable

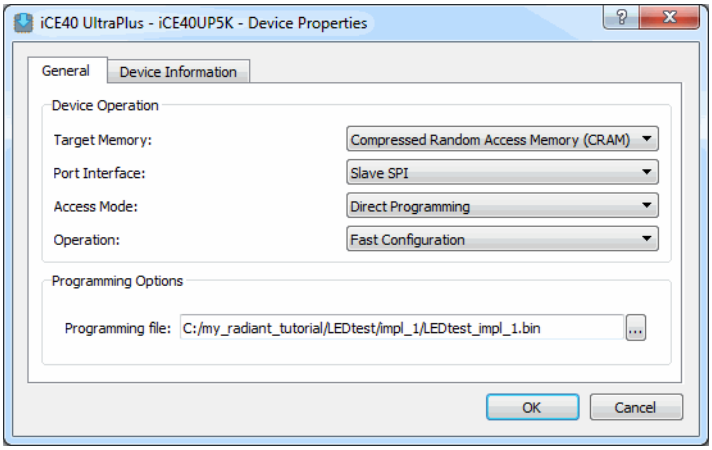


To download the bitstream to the FPGA on the board:

1. Remove any Lattice USB Programming cables from your system.
2. Connect a USB cable from your computer to the iCE40 UltraPlus Breakout board. Give the computer a few seconds to detect the USB device before moving to step 4.
3. Choose **Tools > Programmer**, or click on the Radiant software toolbar.
4. In the Cable Setup box, click **Detect Cable.**
   a. Cable should be set at **HW-USBN-2B (FTDI)**.
   b. Port should be set at **FTUSB-0**.
5. Ensure that the proper device is selected by doing the following:
   a. Click in the Device Family column and choose **iCE40 UltraPlus** from the pull-down menu.
   b. Click in the Device column and choose **iCE40UP5K** from the pull-down menu.
6. Right-click the cell labeled iCE40UP5K and choose **Device Properties** to open the Device Properties dialog box.

7. Ensure the settings are as follows:
   - For Target Memory, choose **Compressed Random Access Memory (CRAM)** from the pull-down menu.
   - For Port Interface, choose **Slave SPI** from the pull-down menu.
   - For Access Mode, choose **Direct Programming** from the pull-down menu.
   - For Operation, choose **Fast Configuration** from the pull-down menu.
   - For Programming file, browse to LEDTest/impl_1/**LEDtest_impl1.bin**, as shown in Figure 24.

Figure 24: Device Properties Dialog Box



8. Click **OK**.
9. In Programmer, choose **Run > Program Device** or click on the Programmer toolbar to initiate the download.
10. If the programming process succeeded, you will see a green-shaded PASS in Programmer's Status column. Check the Programmer output console to see if the download passed.

    The RGB LED on the iCE40 UltraPlus Breakout Board may glow and slowly blink on and off depending on the switch settings. Table 2 lists how the Breakout Board switches affect the RGB LED using this tutorial design.

Table 2: iCE40 UltraPlus Breakout Board Switch Description

Switch	Signal	Description
37A	color_sel[1]	Used to select the color.
36B	color_sel[0]	Used to select the color.
44B	RGB_Blink_En	▶ In UP position, RGB LED blinks. ▶ In DOWN position, RGB LED stops blinking.
49A	rst	▶ In UP position RGB LED will not glow. ▶ In DOWN position, RGB LED glows.

Task 13: Use Reveal Inserter to Add On-Chip Debug Logic

In this task, you will use Reveal Inserter to configure a Reveal core based on triggering conditions and the desired trace buffer. The primary output of Reveal Inserter is a modified version of your design with one or more cores instantiated and the core logic ready for mapping, placement, and routing.

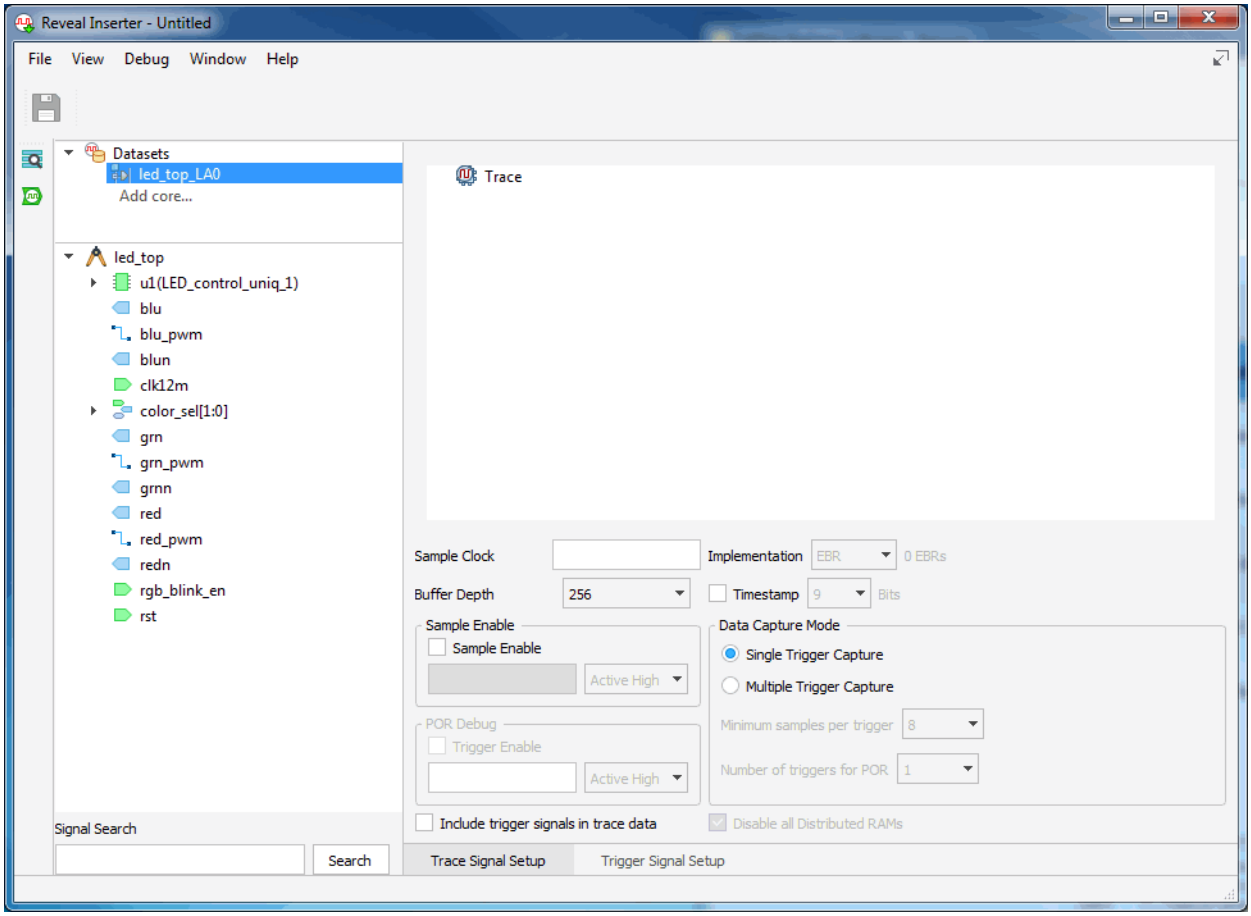
Note that the current version of the Radiant software support for UltraPlus devices does not support multi-core debug.

To generate and add a Reveal core:

1. Choose **Tools > Reveal Inserter** or click on the Radiant software toolbar.
2. Click **Detach Tool** icon at the upper-right corner to detach Reveal Inserter.

   Reveal Inserter is detached from the Radiant software main window, as shown in Figure 25.

Figure 25: Reveal Inserter Main Window

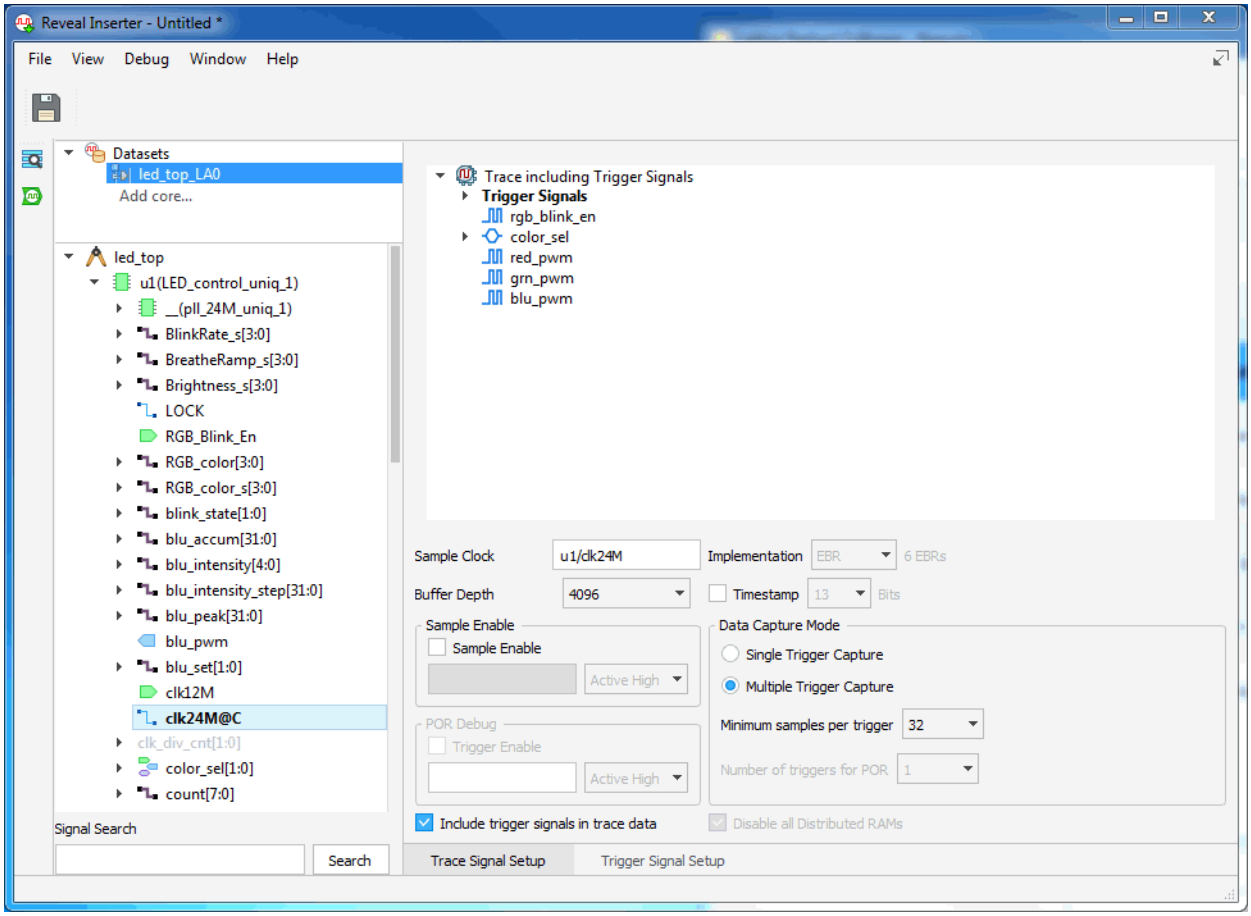


3. Click on the **Trace Signal Setup** tab, if it is not already selected.

4. From the Design Tree tab, expand the **led_top** module.
5. Drag the following signals from the Design Tree pane to Trace Data pane.
   - rgb_blink_en
   - color_sel
   - red_pwm
   - grn_pwm
   - blu_pwm
6. Select the **Include trigger signals in trace data** option.
7. Expand **u1(LED_control_uniq_1)**.
8. Drag the **clk24M** signal from the Design Tree pane to the Sample Clock box or type **u1/clk24M** in the Sample Clock box.
9. From the pull-down menu in the Buffer Depth box, select **4096**.
10. Set Data Capture Mode to **Multiple Trigger Capture** and Minimum Samples Per Trigger to **32**.

    The Trace Signal Setup tab should now resemble Figure 26.

Figure 26: Trace Signal Setup Tap



Setting Up the Trigger Units

Here you will set up the trigger units in the Trigger Unit section of the Trigger Signal Setup tab.

To set up the trigger units:

1. Click on the **Trigger Signal Setup** tab.
2. Minimize the **u1(LED_control_uniq_1)** module if still expanded.
3. Drag the **rst** signal from the Design Tree pane to the Signals (MSB:LSB) box in the Trigger Unit pane.
4. Click **Add** to add a second trigger unit.
5. Drag the **rgb_blink_en** signal from the Design Tree pane to the Signals (MSB:LSB) box in row 2 of the Trigger Unit pane.
6. Leave all other values as default.

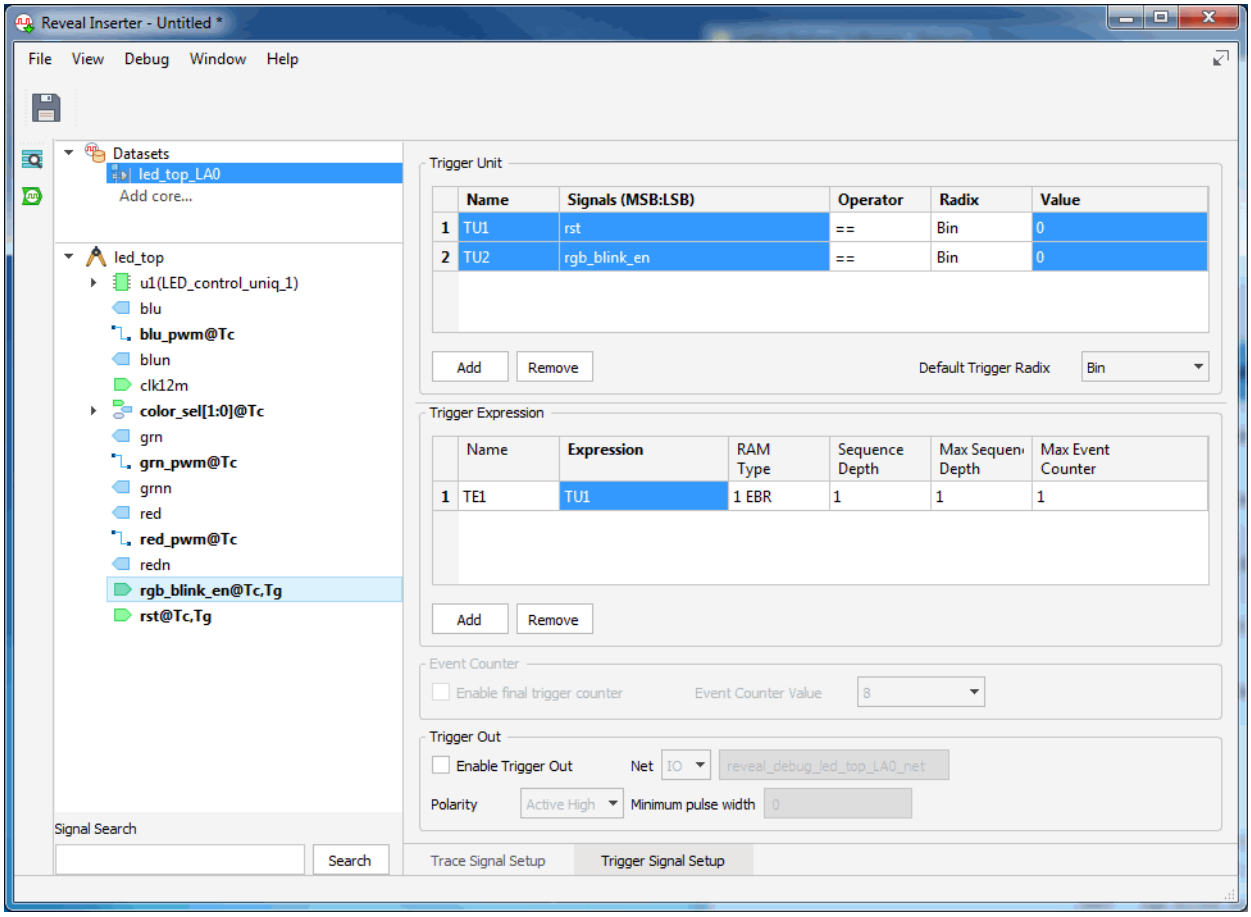
Setting Up the Trigger Expressions

Now you will set up the trigger expressions in the Trigger Expression section of the tab.

To set up the trigger expressions:

1. In the Name box in the Trigger Expressions section, use the default name of TE1.
2. In the Expression box, type **TU2**, and then **Enter** on your keyboard. The RAM Type automatically changes to **1 EBR**.
3. The Trigger Signal Setup tab should now should now resemble Figure 27.

Figure 27: Trigger Signal Setup Tap



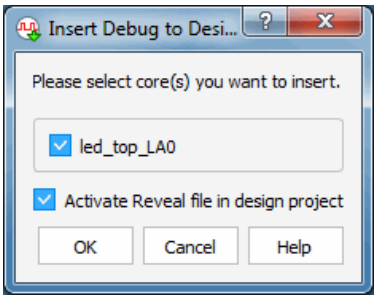
Inserting the Debug Logic

Now you will insert the debug logic into the design project.

To insert the debug logic:

1. Choose **Debug > Insert Debug** or click .
2. In the Insert Debug to Design dialog box, shown in Figure 28, be sure that the **Activate Reveal File in Design Project** option is selected.

Figure 28: Insert Debug to Design Dialog Box



3. Click **OK**.

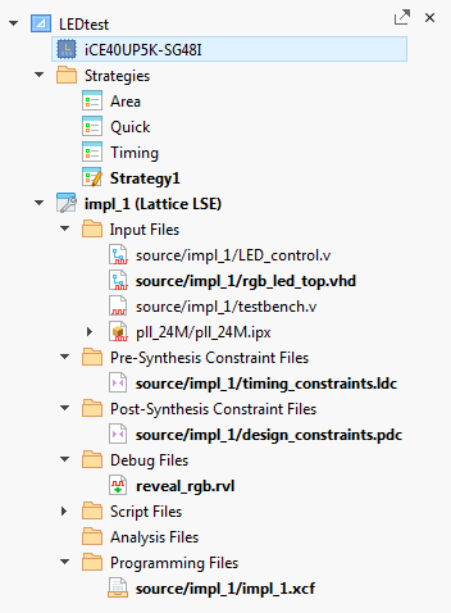
   The Save Reveal Project dialog box opens.

4. Name the file **reveal_rgb**.

5. Click **Save**.

   The .rvl file is added to the Debug Files folder in the File List view, as shown in Figure 29.

Figure 29: File List with Debug File



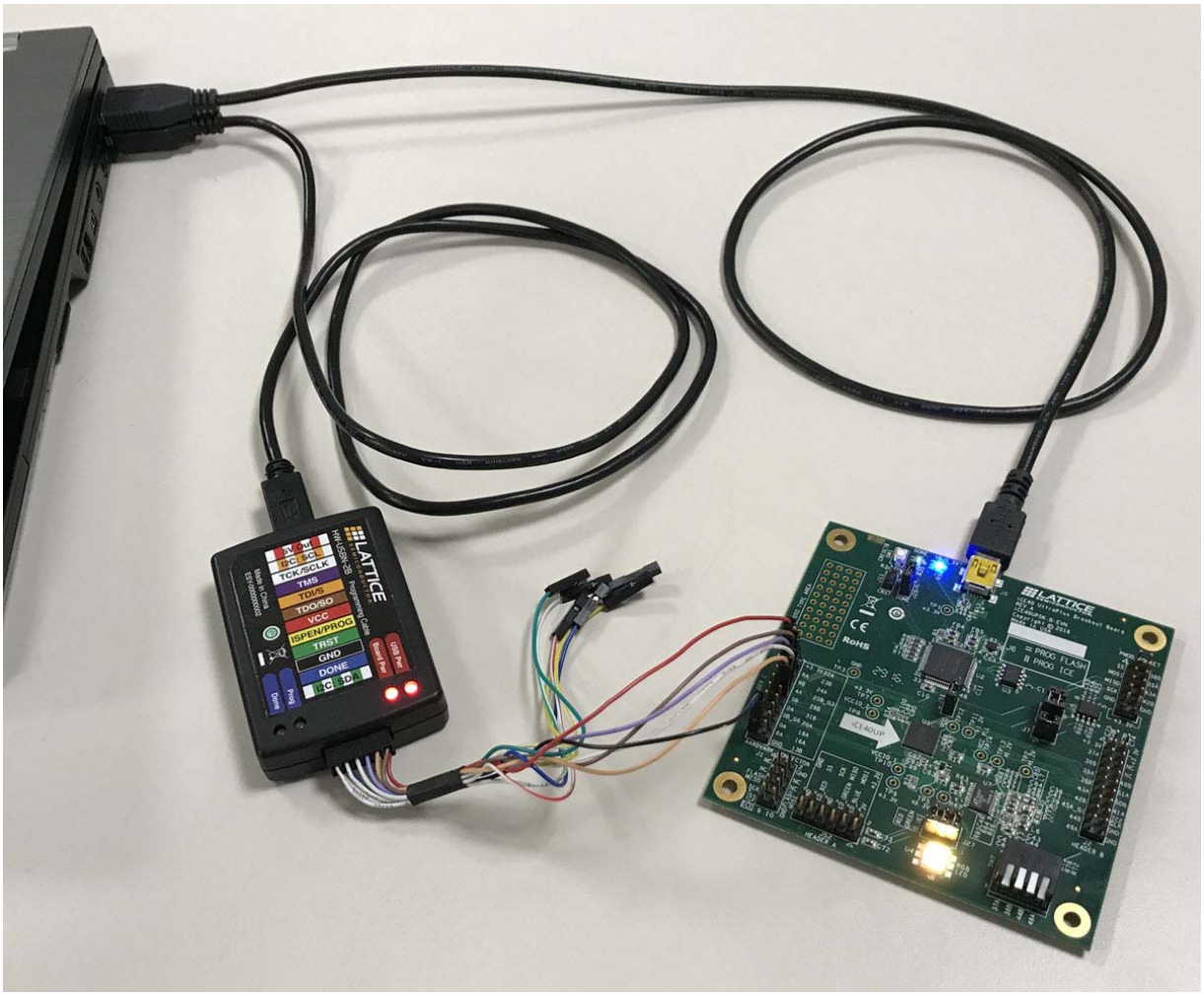
6. Close Reveal Inserter.

Generating a Bitstream and Programming the FPGA

Use the Process Toolbar to generate files for exporting and use Programmer to download the bitstream to the FPGA. This task assumes that the iCE40 UltraPlus Breakout Board is connected to your computer with a download cable.

This task requires that, at minimum, connector pins be soldered to Header C of the iCE40 UltraPlus Breakout Board. Also, in addition to the USB cable, the board must be connected to the computer with a Lattice HW-USBN-2B Download Cable. The following photograph shows the iCE40 UltraPlus Breakout Board connected to the computer with a USB cable and a Lattice HW-USBN-2B Download Cable.

Figure 30: iCE40 UltraPlus Breakout Board with Header Pins Installed and Connected to Computer with USB Cable and Lattice HW-USBN-2B Download Cable



The Lattice HW-USBN-2B Download Cable should be connected to Header C of the iCE40 UltraPlus Breakout Board according to Table 3.

Table 3: Lattice HW-USBN-2B Download Cable to iCE40 UltraPlus Breakout Board Connections

Lattice HW-USBN-2B Download Cable Connector	Breakout Board Header C Pin
VCC (Red)	3.3V
TDO/SO (Brown)	23B
TMS (Purple)	24A
TCK/SCLK (White)	25B_G3
TDI/S (Orange)	29B
GND (Black)	GND

To generate a bitstream:

1. Click the **Run** button on the Radiant software Process Toolbar.

   The Radiant software generates the selected files and saves them in your project directory.
2. If Programmer is not already open, choose **Tools > Programmer**, or click on the Radiant software toolbar.
3. In the Cable Setup box, click **Detect Cable.**
4. In the dialog box, select the cable labeled **Lattice iCE40UP5K Breakout**.
5. Click **OK**.
6. Ensure that the proper device is selected by clicking the cell in the Device column and selecting **iCE40UP5K** from the pull-down menu.
7. Right-click the cell labeled iCE40UP5K and choose **Device Properties** to open the Device Properties dialog box.
8. Ensure the settings are as follows:
   - For Target Memory, choose **Compressed Random Access Memory (CRAM)** from the pull-down menu.
   - For Port Interface, choose **Slave SPI** from the pull-down menu.
   - For Access Mode, choose **Direct Programming** from the pull-down menu.
   - For Operation, choose **Fast Configuration** from the pull-down menu.
9. Ensure that the bitstream file named **LEDtest_impl1.bin** is selected as the programming file.
10. Click **OK**.
11. In Programmer, choose **Run > Program Device** or click on the Programmer toolbar to initiate the download.
12. If the programming process succeeded, you will see a green-shaded PASS in the Programmer Status column. Check the Programmer output console to see if the download passed.
13. Close Programmer.

    A dialog box opens asking if you want to save your changes.
14. Click **No**.

Task 14: Use Reveal Logic Analyzer to Perform Logic Analysis

In this task, you will use Reveal Logic Analyzer to set up trigger conditions and view trace buffer data from the on-chip Reveal core operating within the device on the iCE40 UltraPlus Breakout Board.

The trigger setup influences under what specific conditions and how the Reveal core trace signal states are displayed in Reveal Logic Analyzer's

graphical user interface. You will explore just a few of the many ways to trigger and trace the system.

Creating a New Reveal Logic Analyzer Project

You must first create a Reveal Analyzer project.

To create a new Reveal Logic Analyzer project:

1. In the Radiant software main window, choose **Tools > Reveal Analyzer/ Controller** or click on the Radiant software toolbar.

   The Reveal Analyzer Startup Wizard dialog box appears, as shown in Figure 31.

2. In the upper-left of the dialog box, select **Create a new file**.
3. Type **reveal_rgb** in the box to name the file.

   The .rva extension is added automatically.

4. In the pull-down menu on the top row next to the file name, choose **HW-USBN-2B (FTDI)**, if it is not already selected.
5. Click **Detect**.

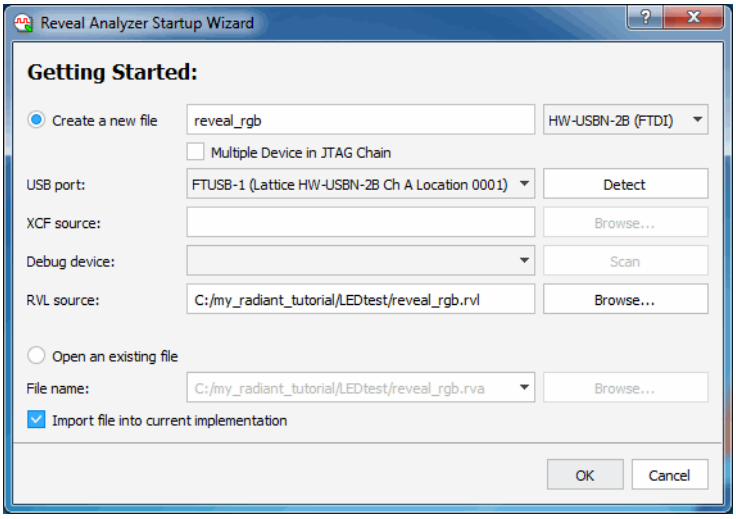
   The cables connected to the PC are listed in the USB Port box. Choose **Lattice HW-USBN-2B Ch A** from the drop-down menu.

6. In the RVL Source box, browse to <*project_directory*>/**reveal_rgb.rvl**.

   Note

   Do not click the Scan button. The device is soft JTAG.

Figure 31: Reveal Analyzer Startup Wizard



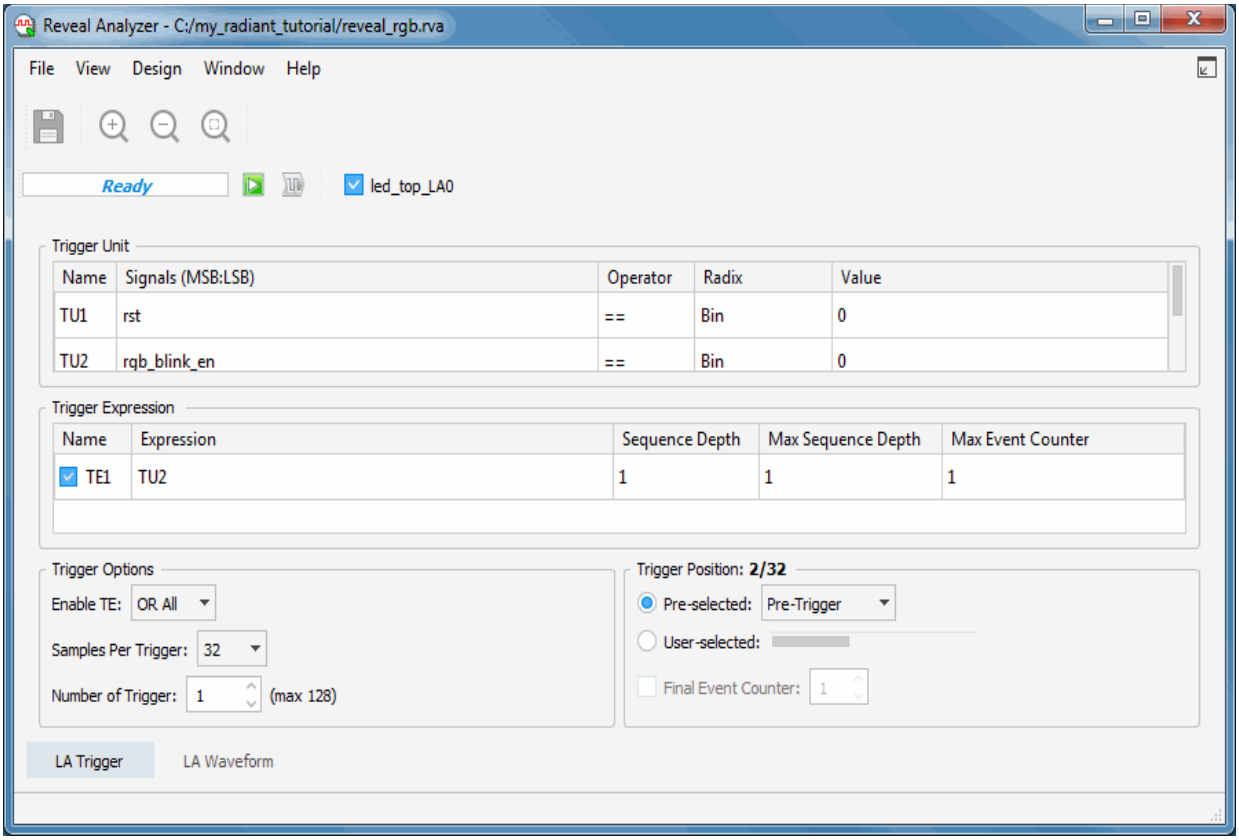
7. Click **OK**.

Reveal Logic Analyzer appears

8. Click **Detach Tool** icon at the upper-right corner to detach the tool.

   The Reveal Logic Analyzer window now appears with the LA Trigger tab selected, as shown in Figure 32. It contains the same trigger units and trigger expressions that you set up in Reveal Inserter.

Figure 32: Reveal Analyzer



9. Choose Trigger Options:

   Samples Per Trigger: **512**.

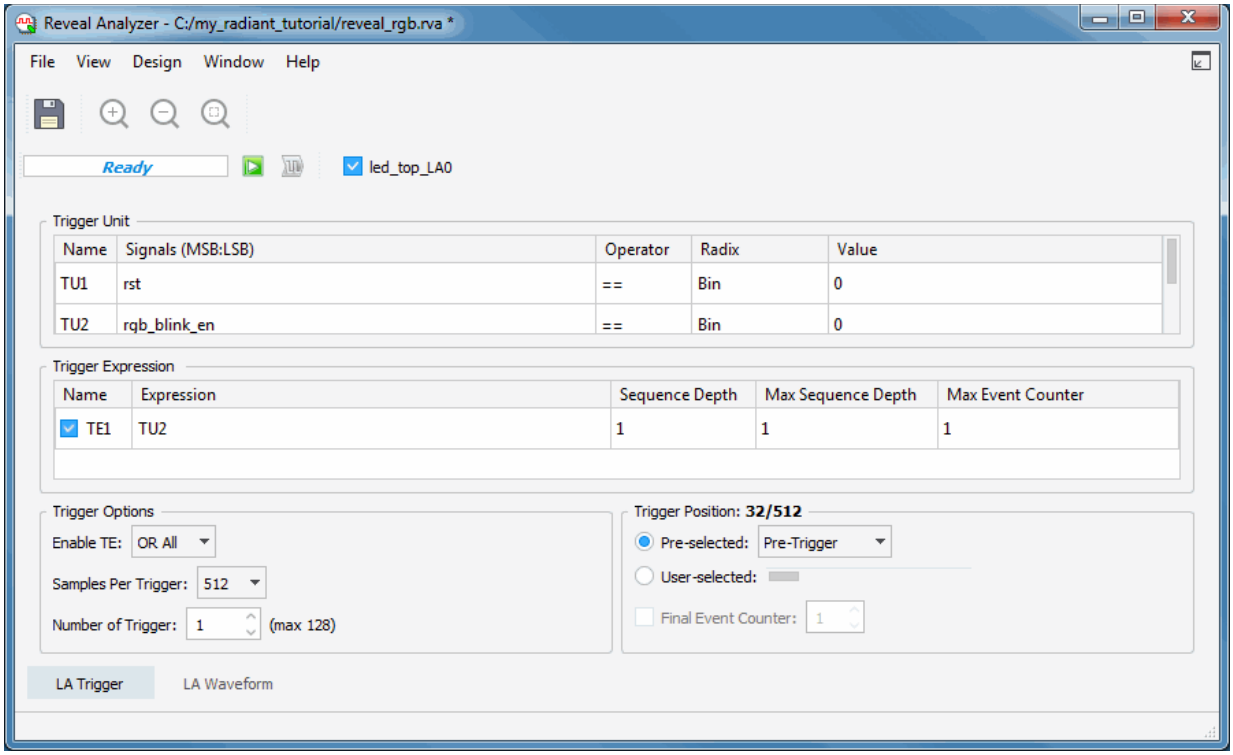
10. The default Trigger Position is: **Pre-selected: Pre-Trigger.**

    In the Trigger position section, you can specify the trigger position relative to the trace data. The numbers in the section title show the current position.The two options to choose from include:

    - **Pre-selected** allows you to choose one of the standard positions.
      - Pre-Trigger: 32/512 of the way from the beginning of the samples.
      - Center-Trigger: 256/512 of the way from the beginning of the samples.
      - Post-Trigger: 480/512 of the way from the beginning of the samples.
    - **User-selected** allows you to choose a position with the slider.

The Reveal Analyzer LA Trigger tab should now appear as shown in Figure 33.

Figure 33: Reveal Analyzer LA Trigger Tab



Running Logic Analyzer

Now that Reveal Logic Analyzer is set up, you can run Logic Analyzer.

To capture data:

1. Click the LA Waveform tab.
2. Click the Run button in the Reveal Analyzer toolbar.

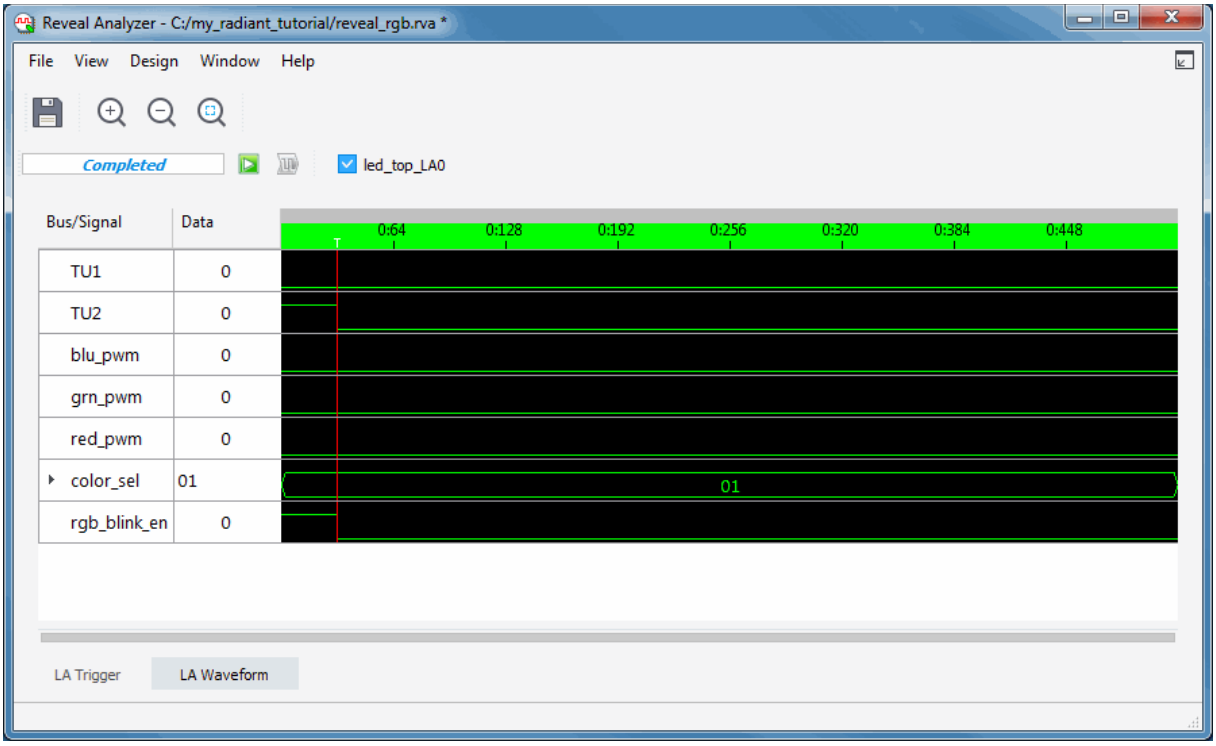
   The Run button changes into the Stop button and the status bar next to the button shows the progress.

   Reveal Analyzer first configures the modules selected for the correct trigger condition, then waits for the trigger conditions to occur. When a trigger occurs, the data is uploaded to your computer. The resulting waveforms appear in the LA Waveform tab.
3. The second DIP switch from the right (44B) may immediately set off a trigger. The trigger expression can now evaluate the next trigger unit and generate a trigger for data to be captured.

   If no trigger occurs click the **Manual Trigger** button.

You now see the waveforms displayed, but the data may be varied as shown in Figure 34.

Figure 34: Reveal Analyzer Waveform



4. Close Reveal Analyzer.

   A dialog box opens asking if you want to save your changes.

5. Click **No**.

6. To close the design project, choose **File > Close Project**.

   The Save Modified Files dialog box opens.

7. To save the files, click **OK**. To not save the changes, click **Deselect All** and then click **OK**.

   The design project and associated tools close. The Radiant window returns to the Start Page.

Summary of Accomplishments

You have completed the *Lattice Radiant Software Tutorial for iCE40 UltraPlus*. In this tutorial, you have learned how to:

- Create a new Radiant software project.
- Create a new module using IP Catalog.
- Verify functionality with simulation.
- Inspect strategy settings.

- Examine resources.
- Set timing and location assignments.
- Run synthesis process.
- Run map design and check reports.
- Run place and route.
- Examine post place and route results.
- Analyze power consumption.
- Run Export Utility programs.
- Download a bitstream to an FPGA.
- Use Reveal Inserter to add on-chip debug logic.
- Use Reveal Logic Analyzer to perform logic analysis.

Recommended References

You can find additional information on the subjects covered by this tutorial in the Radiant software online Help, and in the *Lattice Radiant Software User Guide*.

Revision History

The following table gives the revision history for this document.

Date	Version	Description
10/09/2019	2.0	Updated for version 2.0 of the Radiant software.
03/25/2019	1.1	Updated Detach tool icon and Reveal Analyzer icon. Updated the following images: Task 2, Fig 2, Fig 5, Fig 6, Fig 7, Fig 8, Fig 10, Fig 11, Fig 12, Fig 13, Fig 14, Fig 15, Fig 24, Fig 25, Fig 26, Fig 27, Fig 29, and Fig 30. Updated description in Task 5. Task2: Step 3 - removed statement regarding selecting Verilog as a language option. Replaced **u1(led_control_uniq_1)** with **u1(LED_control_uniq_1).** Updated links to Lattice Radiant Software User Guide. Corrected link to Programming Cables User Guide.
02/07/2018	1.0	Initial Release.