Migrating Designs from AMD ISE Design Suite to Lattice Diamond Software

Application Note

FPGA-AN-02083-1.0

February 2024

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
CPLD	Complex Programmable Logic Device
ECO	Engineering Change Order
FPGA	Field Programmable Gate Array
GUI	Graphical User Interface
HDL	Hardware Description Language
I/O	Input/Output
IP	Intellectual Property
LDC	Lattice Design Constraints
LPF	Lattice Preference File
LSE	Lattice Synthesis Engine
NGD	Native Generic Database
PACE	Pinout and Area Constraints Editor
PAR	Place and Route
PCB	Printed Circuit Board
PRF	Preference file
RTL	Register Transfer Level
SDC	Synopsys Design Constraints
TCL	Tool Command Language
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
XCF	XST Constraint File
XST	Xilinx Synthesis Technology

1. Introduction

The Lattice Semiconductor Field Programmable Gate Array (FPGA)/Complex Programmable Logic Device (CPLD) design flow is similar in conception and implementation to the AMD FPGA design flow. At its core, a hardware description language (HDL) code of the register transfer level (RTL) can be imported into one of Lattice's design software and then configured to one of Lattice's FPGA.

This document guides FPGA designers familiar with AMD ISE® Design Suite software, specifically version 14.7, to migrate existing designs to Lattice Diamond® software and highlights some of the differences and similarities between the design flows of AMD ISE Design Suite software and Lattice Diamond software.

This migration guide starts with an overview and comparison of the design software and a mapping of the tools and file extensions between the two. The next section compares the design flows of the design software. The succeeding sections provide step-by-step walkthroughs on the following:

- Project creation and management
- Design entry
- Implementation flow
- Programming/Configuration

2. Design Software Overview

A design software is required to develop and implement FPGA/CPLD designs for market usage. The design software must have the following characteristics to reduce the design's time-to-market:

- Is full-featured which means it offers all necessary tools for design development.
- Has high performance which means it can perform powerful optimizations and analyses.
- Has intuitive user interface which provides the best user experience with a graphical user interface (GUI) that is both modular and wizard driven.

Diamond software has these characteristics, making it one of the best industry solutions for low-end to mid-range FPGA designs. Diamond software integrated tool environment provides a modern and comprehensive user interface to control Lattice Semiconductor FPGA implementation process. Diamond software supports older Lattice FPGA device families when compared to Lattice Radiant™ software, which supports relatively newer FPGA device families.

Diamond software uses an expanded project-based design flow and integrated tool views to create and analyze design alternatives and what-if scenarios easily. The Implementations and Strategies concepts provide a convenient way for you to try alternate design structures and manage multiple tool settings.

Key features of the Diamond software are as follows:

- Has system-level information, integrated HDL code checking, and consolidated reporting features.
- Includes a Timing Analysis View that saves time by allowing interactive changes to design constraints and viewing the results without disturbing your design.
- Has ECO Editor and Programmer tools that are tailored to make individual task easier.
- Has a cross-probing feature and a shared memory architecture to ensure fast performance and better memory utilization.
- Is highly customizable and provides Tool Command Language (TCL) scripting capabilities from either its built-in console or from an external shell.

2.1. Design Software Comparison

Most of the capabilities available in the AMD ISE Design Suite software are also available in the Diamond software.

However, the terminology of the individual tools, features, and file formats may differ from one software to the other. Table 2.1 lists some of the tools, features, and file formats in AMD ISE Design Suite software and their corresponding names in Diamond software.

Table 2.1. Tools, Features, and File Extensions Mapping between AMD ISE Design Suite and Diamond Software

Comparison Items	AMD ISE Design Suite software	Lattice Diamond software
Optimization or Implementation Settings	Design Strategies	Design Strategies
Design Entry	HDL file AMD Core Generator	HDL file IPexpress or Clarity Designer Schematic Editor
Design Constraints	Constraints Editor ISE Text Editor PlanAhead™ tool for FPGA Pinout and Area Constraints Editor (PACE) for CPLD Commercially available text editors HDL source file using a text editor XST Constraint File (XCF) using a text editor	Spreadsheet View LDC Editor Source Editor
Synthesis Tools	Xilinx Synthesis Technology (XST) Synplify Pro Precision	Lattice Synthesis Engine (LSE) Synplify Pro
Schematic Tools	Schematic Viewer for RTL and Technology View	Netlist Analyzer
Place & Route	Xilinx Implementation Tool	Diamond Place & Route

Comparison Items	AMD ISE Design Suite software	Lattice Diamond software
Simulation	Isim ModelSim QuestaSim	ModelSim
Power	Xpower Analyzer	Power Calculator
On-Chip Debug Tool	ChipScope™ Pro Tool	Reveal Inserter Reveal Analyzer
Project File Extension	.xise	.ldf

3. Design Entry

HDL such as Verilog and VHSIC Hardware Description Language (VHDL) are fully supported in both Lattice Diamond and AMD ISE Design Suite software. This section describes various design HDL methodologies used in both Lattice Diamond and the ISE Design Suite software, and highlights the similarities and differences.

3.1. Using HDL in Diamond Software

You can reuse your Verilog and VHDL source files when migrating designs from AMD ISE Design Suite to Lattice Diamond software.

In ISE Design Suite software, new source files are added using the New Source wizard by clicking **File** > **New** or by right-clicking anywhere on the hierarchy window and choosing **New Source** or **Add Source** to add to existing sources.

In the Lattice Diamond software, you can do this by:

1. Clicking **File** > **New** or by right-clicking the **Input Files** folder on the **File List** tab.
2. From the dropdown menu click on **Add** > **New File**. A **New File** dialog box as shown in Figure 3.1 opens and you can choose the HDL file type to add to the design. You can also add an existing source by selecting **Add** > **Existing File**.

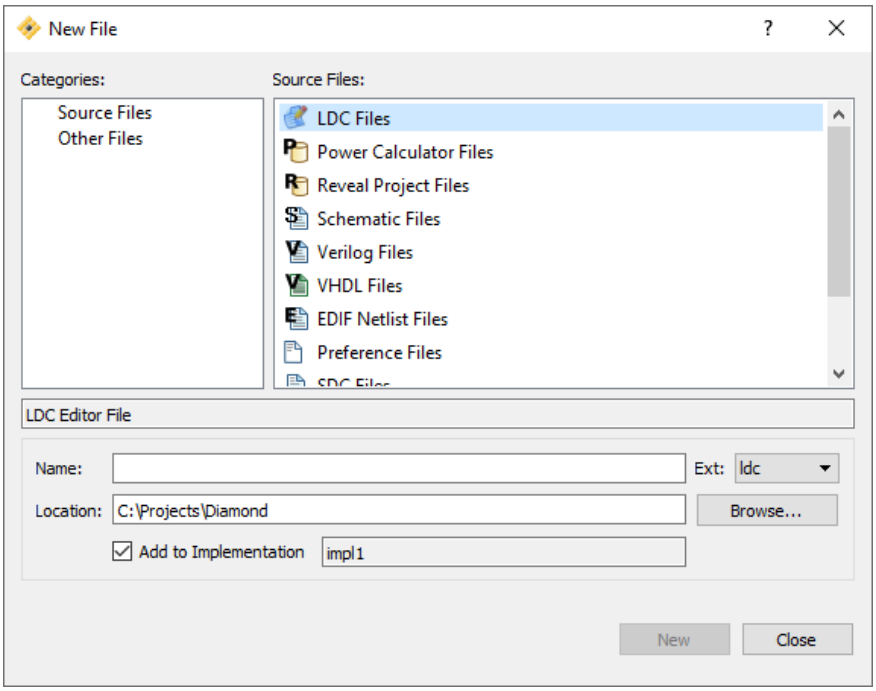

Figure 3.1. New File Wizard in Diamond Software

3.2. Using Diamond Primitives

If your design contains any ISE primitives, you may need to replace them with Diamond primitives. Refer to *Libraries Reference Guide* in Diamond software's Help section for a complete list of primitives available for Diamond devices.

3.3. Using an IP Core in Diamond Software

In ISE Design Suite software, Core Generator is used to import an IP core into a project. Whereas in Diamond software, IPexpress or Clarity tool is used to configure and import an IP core into a project.

In ISE Design Suite software, you can open the Core Generator under **Tools** > **Core Generator**. The Core Generator window opens and you can select which IP core to customize and add to the design. Figure 3.2 shows the ISE Core Generator flow.

In Lattice Diamond software, an IP core can be accessed either through the IPexpress or the Clarity Designer. This document focuses on the IPexpress Designer. Clarity Designer is only available for the ECP5™ and CrossLink™ device families. Figure 3.3 shows the Diamond IPexpress or Clarity Designer flow.

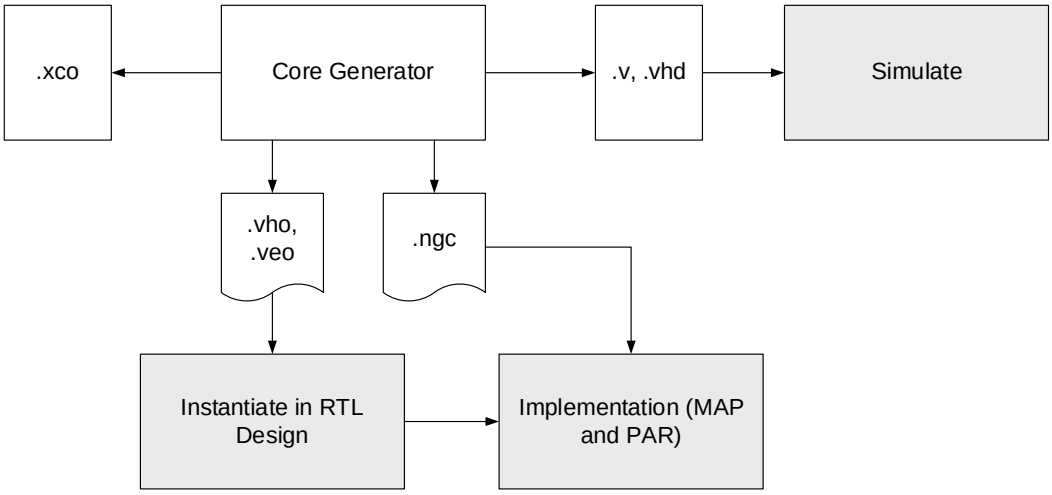

Figure 3.2. ISE Core Generator Flow

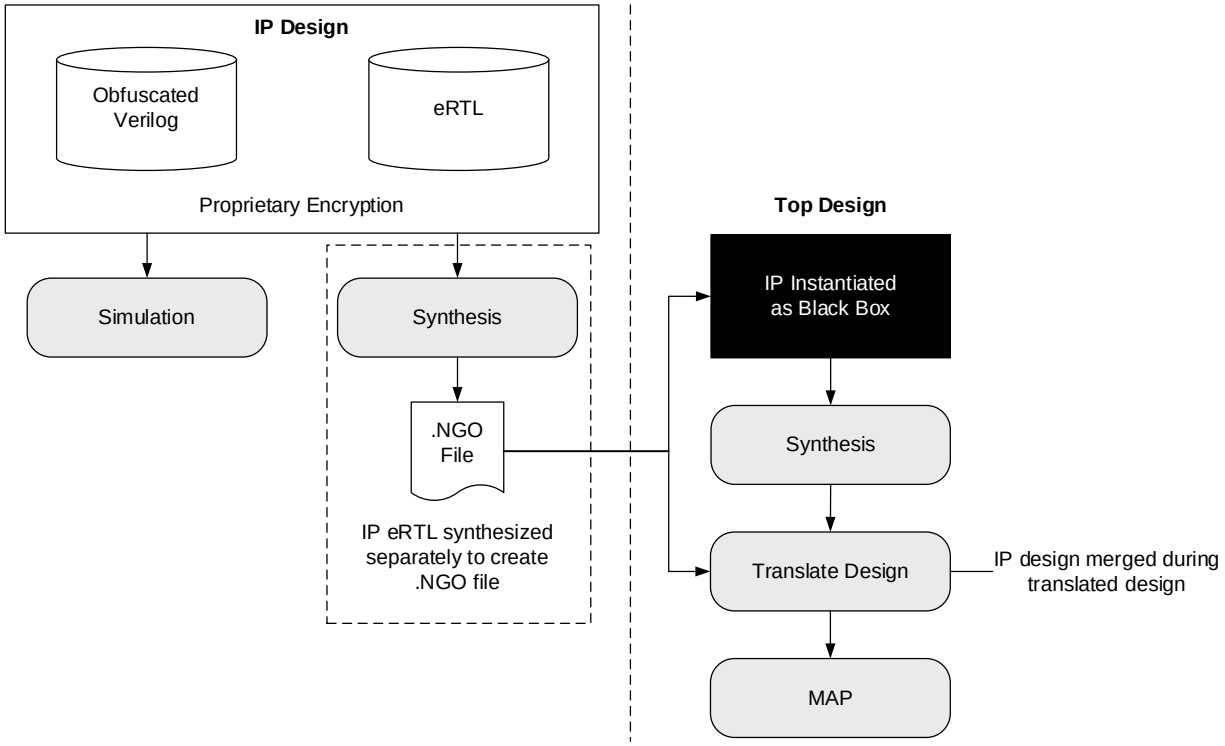

Figure 3.3. Diamond IPexpress or Clarity Designer Flow

Refer to the following steps to configure and import an IP core using Diamond IPexpress Designer:

1. Click the IPexpress icon on the Diamond software toolbar as shown in Figure 3.4 to open the IPexpress tab.

Figure 3.4. IPexpress Icon on Diamond Software Toolbar

2. On the IPexpress tab as shown in Figure 3.5, you can select between an installed IP core on a local folder or to download and install additional IP core from the IP Server.

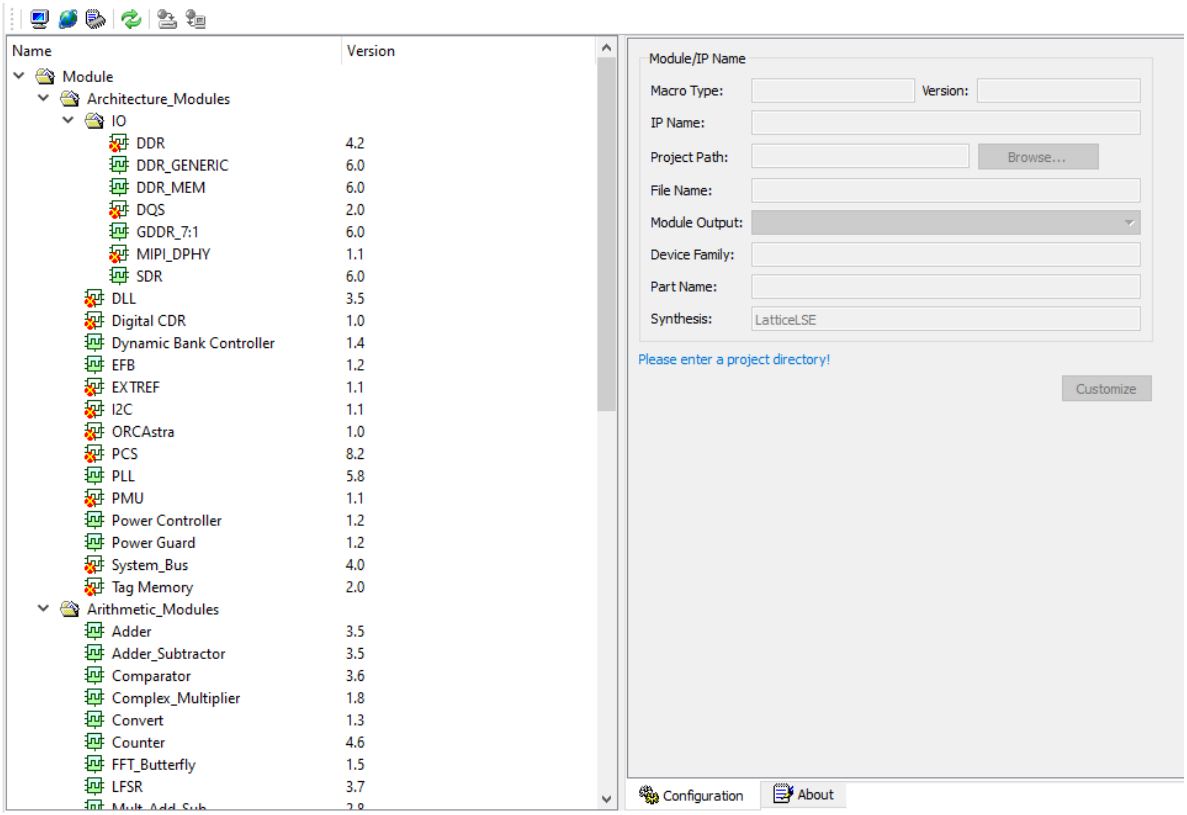

Figure 3.5. Diamond IPexpress GUI

IPexpress uses scuba engine to generate an IP core. Output files generated by Diamond IPexpress and their descriptions are listed in Table 3.1.

Table 3.1. IP Express Output Files and their Descriptions

File Types	Descriptions
<ip_name>.ipx	The IPexpress IP file which can be imported to Diamond project
<ip_name>.v or <ip_name>.vhd	RTL code in Verilog or VHDL file format which must be chosen during IP creation
<ip_name>.edn	Edif file for the IP Core
<ip_name>.lpc	IP configuration file
<ip_name>.naf	IP I/O list
<ip_name>.srp	Scuba engine log file
<ip_name>.sym	IP symbol for schematic

Some of the common IP cores used in AMD Spartan 3E and Lattice ECP3 are listed in Table 3.2. For the detailed descriptions of the IP, refer to Diamond software's Help section.

Table 3.2. Comparison of Common IP Cores Used between AMD Spartan 3E and Lattice ECP3 Devices

Purpose	Spartan 3E IP Cores	ECP3 IP Cores
Clock Manager	Single DCM_SP	PLL IP
FIFO Operations	FIFO Generator	FIFO IP, FIFO_DC IP
Memory Blocks	RAM IP (SP, Dual Port, True Dual Port, and Dual Port)	RAM_DP, RAM_DP_TRUE, RAM_DQ
Distributed Memory	Distributed Memory (ROM, Single Port, Dual Port, and SR16 based memory)	Distributed_DPRAM, Distributed_ROM, Distributed_SPRAM
Shift Register	RAM-based Shift Register	RAM_Based_Shift_Register
Arithmetic Operations	Adder Subtractor	Adder, Adder_Subtractor
	Multiplier	Multiplier, Multiply_Accumulate
Trigonometric Operations	CORDIG	Sin-Cos_Table

4. Design Preferences

In Diamond software, all preferences are specified post synthesis. The synthesis flow has its own separate constraining system decoupled from the back-end. Any changes in preferences require the synthesis flow to rerun starting from Map Design. Figure 4.1 shows the Lattice Diamond Constraints flow.

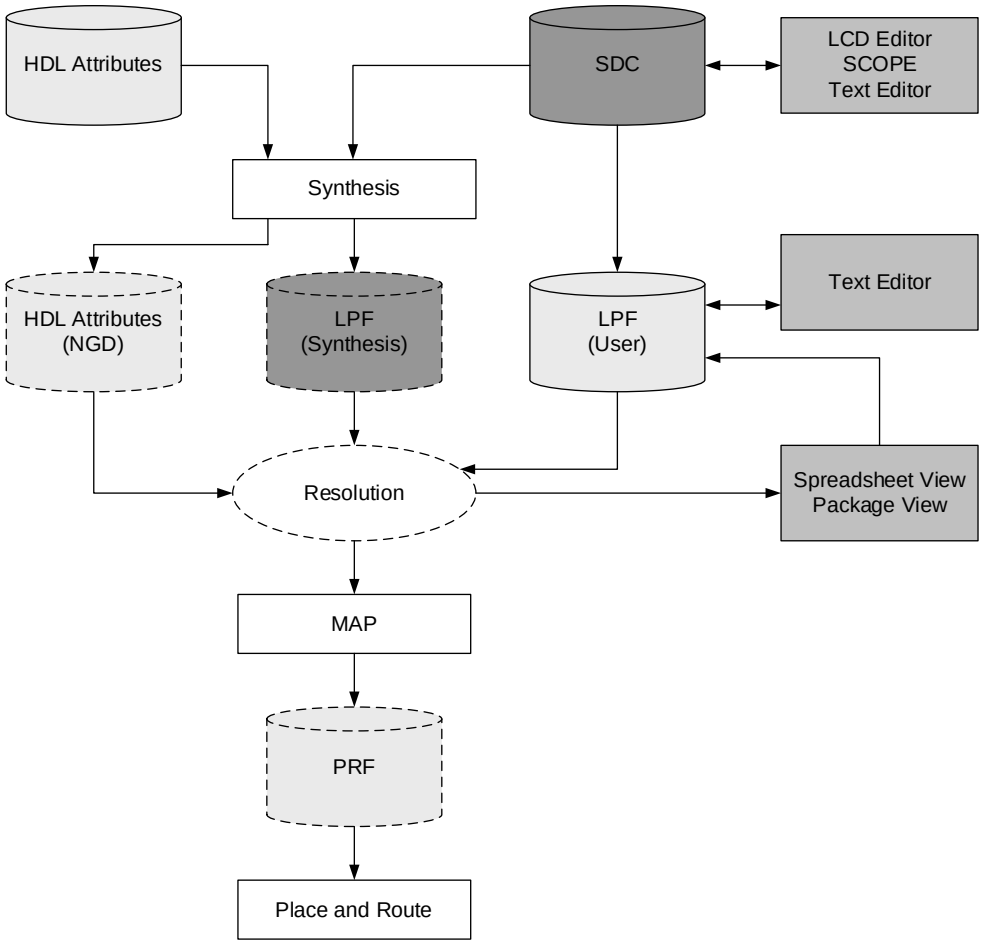

Figure 4.1. Lattice Diamond Constraints Flow

4.1. Timing Preferences

Table 4.1 below shows commonly used timing preferences in ISE Design Suite software and their equivalent preferences in Diamond software. For a detailed description of each of these preferences, refer to Diamond software's Help section.

Table 4.1. ISE Design Suite Timing Constraints to Diamond Software Preferences Conversions

ISE Constraints	Constraint Functions	Diamond Equivalent Preferences
FAST	To turn on Fast Slew Rate Control	A part of the SLEWRATE preference
OFFSET IN	To specify input delay	INPUT_SETUP
OFFSET OUT	To specify output delay	CLOCK_TO_OUT
KEEP	To prevent a net from either being absorbed by a block or synthesized out	Can be controlled using the synthesis attribute /* synthesis syn_keep = 1 */
MAXDELAY	To specify the maximum delay in a net	MAXDELAY
MAXSKEW	To specify the maximum skew in a net	MAXSKEW

ISE Constraints	Constraint Functions	Diamond Equivalent Preferences
PERIOD	To specify the timing relationship of a global clock such as an *fMAX* requirement	PERIOD/ FREQUENCY
TIG	Timing Ignore or false path	BLOCK
FROM/THRU/TO	Multicycle path constraints constrained using *FROM:* and *TO:* in ISE Design Suite software	MULTICYCLE
SYSTEM JITTER	To specify system jitter of the design	SYSTEM JITTER

Here are some examples of timing constraints conversions:

- FREQUENCY/PERIOD preference:
  - ISE Design Suite software:
    ```
    NET "clk" TNM_NET = clk_in;
    TIMESPEC TS_clk_in = PERIOD "clk_in" 5 ns HIGH 50%;
    ```
  - Diamond software:
    ```
    PERIOD NET “clk” 5 NS HIGH 2.5 NS;
    ```
- INPUT_SETUP preference for an input port:
  - ISE Design Suite software:
    ```
    NET in_a OFFSET = IN 2 BEFORE clk;
    ```
  - Diamond software:
    ```
    INPUT_SETUP PORT “in_a” 2 ns CLKNET ”clk”;
    ```
    INPUT_SETUP preference can also be written as:
    ```
    INPUT_SETUP PORT “in_a” (X-2) ns CLKNET ”clk”; where X = Clock Period
    ```
    Note: If the OFFSET numbers must be used directly, use INPUT_DELAY option along with the INPUT_SETUP preference.
- INPUT_SETUP preference for ALLPORTS:
  - ISE Design Suite software:
    ```
    OFFSET = IN 2 AFTER clka;
    ```
  - Diamond software:
    ```
    INPUT_SETUP ALLPORTS INPUT_DELAY 2 ns CLKNET ”clk”;
    ```
    Note: If the OFFSET numbers must be used directly, use INPUT_DELAY option along with the INPUT_SETUP preference.

    INPUT_SETUP preference can also be written as:
    ```
    INPUT_SETUP ALLPORTS (X-2) ns CLKNET ”clk”; where X = Clock Period
    ```
    Note: In Diamond software, INPUT_SETUP preference is used to specify input delay from the fabric input port to the first register. Hence the OFFSET numbers cannot be directly copied over. From the example, 2 ns delay must be subtracted from the clock period to capture the delay from the input port to the first register.
- CLOCK_TO_OUT preference for an output port:
  - ISE Design Suite software:
    ```
    NET out1 OFFSET = OUT 8 AFTER clk2;
    ```

- Diamond software:
```
CLOCK_TO_OUT PORT “out1” OUTPUT_DELAY 8 ns CLKPORT=”clk2”;
```

  CLOCK_TO_OUT preference can also be written as:
```
CLOCK_TO_OUT PORT “out1” (X-8) ns CLKPORT=”clk2”; where X= Clock Period
```

  Notes:
  - In the ISE Design Suite software, OFFSET = OUT performs only maximum delay analysis. In Diamond software, CLOCK_TO_OUT preference can be used with MAX and/or MIN options to perform max and min delay analysis.
  - If the OFFSET numbers must be used directly, use OUTPUT_DELAY option along with the CLOCK_TO_OUT preference.

- BLOCK preference:
  - ISE Design Suite software:
```
NET reset TIG;
```
  - Diamond software:
```
BLOCK RESETPATHS
```

- MAXSKEW preference:
  - ISE Design Suite software:
```
NET NetB MAXSKEW = 5ns;
```
  - Diamond software:
```
MAXSKEW NET "NetB" 5 NS;
```

- MULTICYCLE preference:
  - ISE Design Suite software:
```
TIMESPEC TS_01 = FROM "R_REG1_GROUP" TO "R_REG3_GROUP" TS_clk*3;
```
  - Diamond software:
```
MULTICYCLE FROM GROUP R_REG1_GROUP TO GROUP R_REG3_GROUP 3 X;
```

4.2. Physical Preferences

Table 4.2 shows commonly used physical preferences in ISE Design Suite software and their equivalent preferences in Diamond software.

Table 4.2. ISE Design Suite Physical Constraints to Diamond Software Preferences Conversions

ISE Constraint	Constraint Function	Diamond Equivalent Preference
LOC	The Location (LOC) constraint	LOCATE
IOB	To specify whether a register should be placed within the IOB of the device	Can be controlled using HDL attribute /* synthesis syn_useioff = 1|0 */
IOSTANDARD	To specify the I/O standard for an I/O pin.	IO_TYPE
DRIVE	To control the output pin current value	DRIVE
VREF	To configure listed pins as *VREF* supply pins that is used with other I/O pins	LOCATE VREF
PROHIBIT	To prohibit a placement on a site	PROHIBIT

4.3. Design Conversion Recommendations for Clock Resources using Preferences

Software tools assign clocks based on signal load and connectivity defined in the HDL code. When converting the design, check for the following:

- Replace the clock buffer with a single net and verify that Primary clocks are allocated properly. If not, use design constraint: USE_PRIMARY to force the software to implement a specific clock on the primary clock routing.

  Example: USE PRIMARY NET “clk_c”;
- Make sure you are using the PCLK pins for your clock input. They have the most optimal path to the clock network, PLLs or edge clocks.

  Note: Refer to the respective device datasheet for details about clocking structure of each device.

4.4. Adding and Modifying Preferences

You can create and modify logical preferences in the .lpf file using the Diamond preference-editing views or modify directly using a text editor. ISE constraints editor provides a spreadsheet-like interface for assigning all instance-specific settings and constraints.

Diamond software provides similar tools to view and define new constraints, called preferences. The tools can also modify existing constraints from the source files and save them as preferences. The preference editor tools available in Diamond software is shown in Figure 4.2.

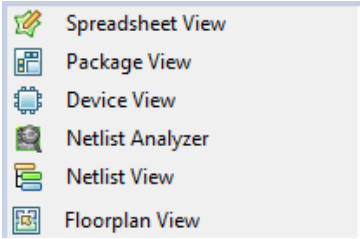

Figure 4.2. Preference Editor Tools in Diamond Software

Table 4.3 lists the functions of each preference editor tool.

Table 4.3. Preference Editor Tools in Diamond Software and their Functions

Preference Editor Tools	Functions
Spreadsheet View	• Modify timing constraints that are defined in the synthesis tool and save them as logical preferences to the active .lpf file. • Set timing objectives such as fMAX and I/O timing. • Define signaling standards and make pin assignments. • Assign clocks to primary or secondary routing resources. • Set parameters for simultaneous switching outputs and perform SSO analysis. • Define groups of ports, cells, or ASIC blocks. • Create UGROUPs from selected instances to guide placement and routing. • Establish REGIONs for UGROUPs or to reserve areas of the floorplan. • Run PIO design rule checking.
Package View	• View the pin layout of the design. • Modify signal assignments and reserve pin sites that should be excluded from placement and routing. • Examine the status of SSO pins. Run PIO design rule checking.
Device View	• Examine FPGA device resources. • Reserve sites that should be excluded from placement and routing.
Netlist View	• View the design tree by ports, instances, and nets. • Assign pins for selected signals.

Preference Editor Tools	Functions
	• Set timing constraints. • Define groups from selected ports or registers. • Create UGROUPs from selected instances to guide placement and routing.
Floorplan View	• View the device layout. • Draw bounding boxes for UGROUPs. • Draw REGIONs for the assignment of groups or to reserve areas. • Reserve sites and REGIONs that should be excluded from placement and routing. • Run PIO design rule checking.

For more information, refer to Diamond software's Help section and Lattice Diamond 3.12 User Guide. You can find more information on Synthesis Constraint Files and LPF Constraint Files in Chapters 5 and 6 of the user guide, and more information on the preference editor tools in Chapter 7 of the user guide.

5. Synthesis

AMD ISE Design Suite software has the XST (Xilinx Synthesis Tool) to perform synthesis on designs with user logic and AMD IP cores. The software also supports Synplify Pro for synthesis.

Lattice Diamond software has a proprietary synthesis tool in Lattice Synthesis Engine (LSE) and supports a third-party synthesis tool in Synplify Pro. Both synthesis tools support synthesis of user logic and Lattice IP cores.

Table 5.1 shows the Synthesis tools supported by both ISE Design Suite and Diamond software.

Table 5.1. Comparison between ISE Design Suite and Diamond Synthesis Tools

ISE Design Suite Software	Lattice Diamond Software
Xilinx Synthesis Tool (XST)	Lattice Synthesis Engine (LSE)
Synplify Pro	Synplify Pro

You can select available synthesis tools in ISE Design Suite software by going to **Project** > **Design Properties**.

In Diamond software, you can do this by right-clicking the current implementation and choosing **Select Synthesis Tool** from the dropdown menu as shown in Figure 5.1.

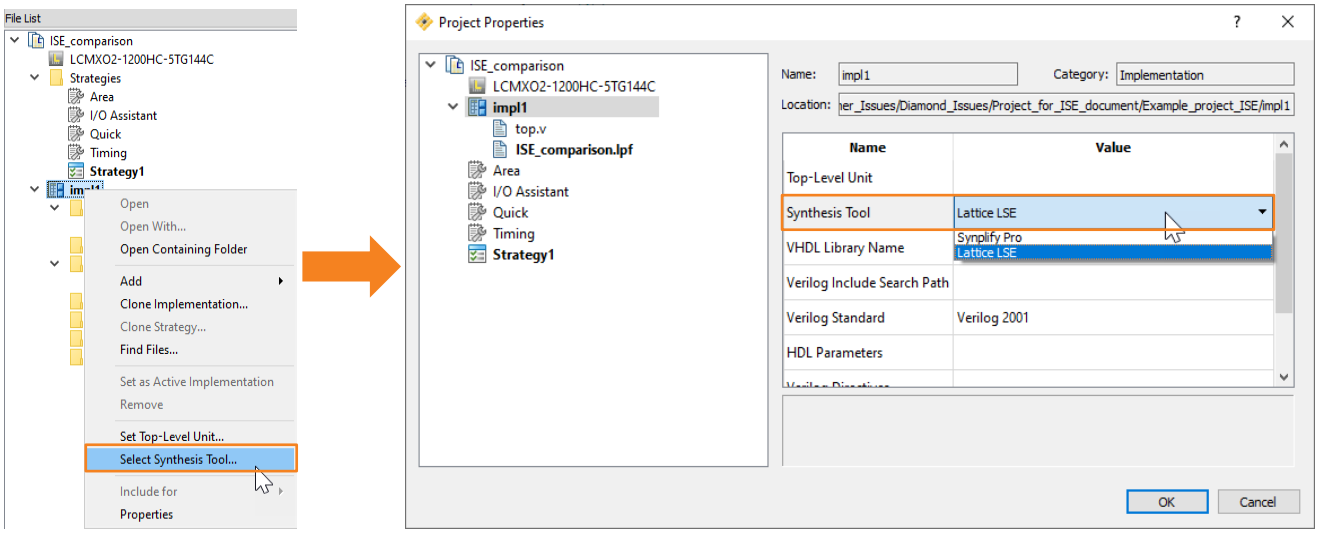

Figure 5.1. Choosing Synthesis Tool in Diamond Software

In ISE Design Suite software, synthesis is performed by double-clicking on **Synthesize -XST process** under **Processes**.

In Diamond software, synthesis can be performed by double-clicking on the **Synthesize Design** on the **Process** tab or by right-clicking **Synthesize Design** and then clicking **Run** on the dropdown menu. It is important to note that when *Synplify Pro* is selected, synthesis is divided into two tasks which are *Synthesize Design* and *Translate* as compared to *Lattice LSE* which only have the *Synthesize Design* task. Figure 5.2 shows the Process window in Diamond software.

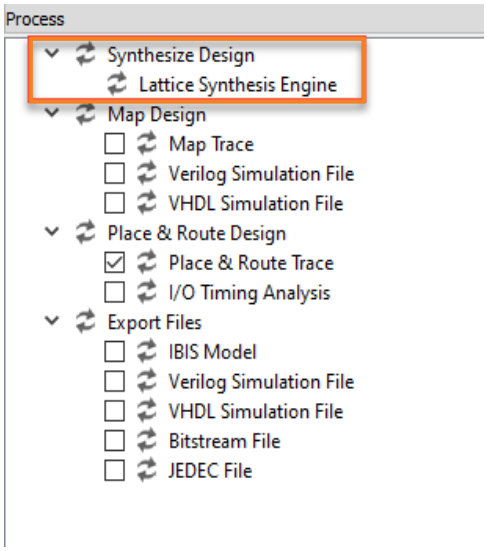

Figure 5.2. Process Window in Diamond Software

6. Netlist Analyzer

A generated netlist can be viewed through one or more schematic views and a browser that shows lists of modules, instances, ports, and nets.

In ISE Design Suite software, this can be performed by going to **Tools** > **Schematic Viewer**.

In Diamond software, this can be performed using Netlist Analyzer of LSE or HDL Analyst of Synplify Pro.

Table 6.1 compares the Schematic Viewer between ISE Design Suite and Diamond software.

Table 6.1. Comparison between ISE Design Suite and Diamond Software Schematic Viewer

Xilinx ISE	Lattice Diamond
RTL Viewer and Technology Map Viewer	Netlist Analyzer and HDL Analyst

To open Netlist Analyzer, click on the **Netlist Analyzer** icon on the Diamond software toolbar as shown in Figure 6.1.

Figure 6.1. Tool bar in Diamond Software

For detailed information, refer to Diamond software's Help section and Lattice Diamond 3.12 User Guide. You can find more information on Netlist Analyzer in Chapter 7 of the user guide.

7. Design Mapping

Design mapping converts the logical design into a network of physical components or configurable logic blocks.

In ISE Design Suite software, this process is combined with the Implementation process. Double-click the **MAP process** to run MAP. You can click on **Implement Design** to automatically run Translate, MAP and Place & Route in a sequence.

In Diamond software, this is a separate process in the design flow that can be optimized through the design strategy settings. To map a design, right-click **Map Design** and select **Run** from the dropdown menu or double-click the **Map Design** as shown in Figure 7.1.

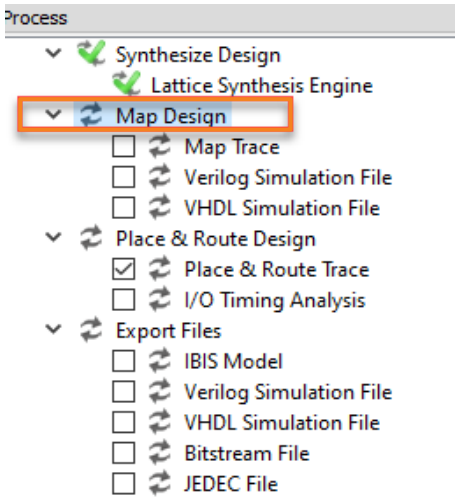

Figure 7.1. Diamond Process bar for MAP

8. Place & Route

After a design has undergone the necessary translation to bring it into the physical design format during mapping, it is ready for placement and routing. Placement is the process of assigning the device-specific components produced by the mapping process to specific locations on the device floorplan.

After placement is complete, the route phase establishes physical connections to join components in an electrical network. The place and route process takes a mapped physical design and places and routes the design. Placement and routing of a design can be cost-based or timing driven.

Placement and routing options can influence the performance and utilization of the design implementation and ease incremental design changes. Some options affect the way the results are reported. Experimenting with place and route settings in the **Strategies** dialog box can help to improve the placement and routing results. Figure 8.1 shows an example of place and route settings.

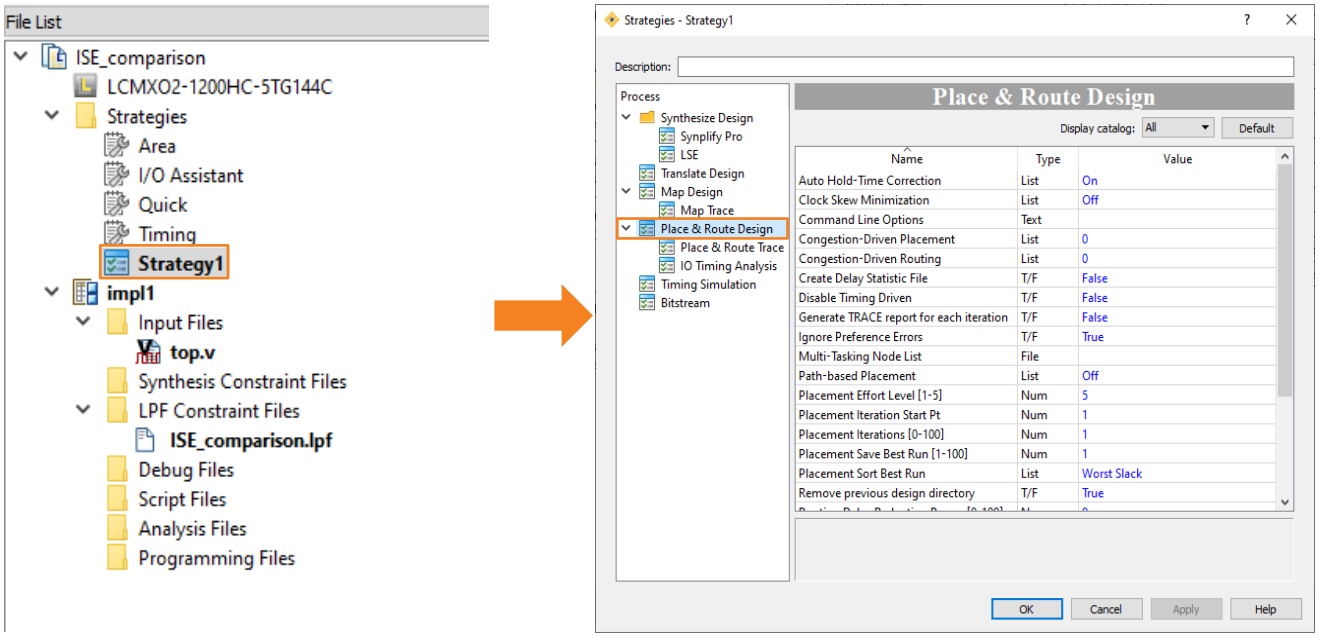

Figure 8.1. Diamond Strategy Settings for Place & Route Design

In most cases, your design requires timing-driven placement and routing, where the timing criteria specified influences the implementation of the design. Static timing analysis results show how constrained nets meet or do not meet the timing preferences.

In Diamond Process view and under **Place & Route Design**, a **Place & Route Trace** process is available as shown in Figure 8.2. This process runs static timing analysis on the place and routed .ncd file. It also generates a report on any timing errors associated with preferences which is called the Place & Route Trace and I/O Timing Analysis report.

The report can be accessed from the **Analysis Reports** folder in the **Reports** window. These reports help to ensure that the I/O plan meets the I/O standards and power integrity requirements of the PCB design.

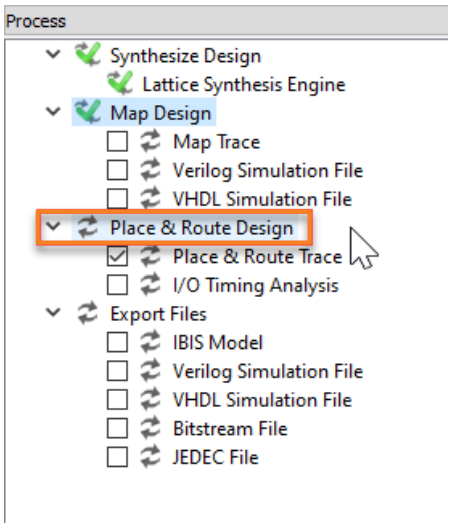

Figure 8.2. Diamond Process Window for Place & Route

9. Cross Probing

Cross-probing is a feature that allows the software to seamlessly have bi-directional communication between different tools within the software itself. You can select a design element from one tool and locate them in another tool. Both ISE Design Suite and Diamond software have cross-probing ability.

The Diamond preference-editing views allow you to select an element in one view and quickly display the corresponding logical or physical element in a different view. For example, you can cross-probe a component in Floorplan View to view the logical elements in Netlist View or you can cross-probe a signal in Spreadsheet View to examine its placement on the pin layout in Package View.

To cross-probe an element from one view to another, follow below steps:

1. Right-click an element in any view.
2. Choose **Show in** and choose the desired view from the pop-up menu. Table 9.1 shows the available options for cross-probing in Diamond software.

Table 9.1. Diamond Software Cross-Probing Options

Cross Probe From	Element	Cross Probe To
Spreadsheet View (Port Assignments)	Pin	Package View Device View Netlist View NCD View Floorplan View
Spreadsheet View (Pin Assignments)	Signal	Package View
	Port Group	Package View
Spreadsheet View (Group)	Anchored UGROUP	Floorplan View
Spreadsheet View (Misc)	REGION	Floorplan View
Package View	Assigned pin	Netlist View Spreadsheet View Floorplan View Device View
	Site	Floorplan View Device View
Device View	DQS, IOL, PFF, PFU, PLL/DLL, DCC/DCS, sysDSP, and sysMEM	Floorplan View Physical View
	Others: GSR, JTAG, Oscillator, etc.	Floorplan View Physical View
	Assignable PIO Cell	Package View Floorplan View Physical View
	Unassignable PIO Cell	Floorplan View Physical View
	DDR Support with bonded DQS	Package View Floorplan View Physical View
Netlist View	Assigned port	Package View
	Instance	Floorplan View Physical View NCD View
	Net	Floorplan View Physical View
NCD View	IOL instance, PCS block, PFF slice, PFU slice, PLL/DLL, sysDSP block, and sysMEM block	Netlist View Floorplan View Physical View

Cross Probe From	Element	Cross Probe To
	User PIO	Netlist View Package View Floorplan View Physical View
	Others: GSR, JTAG, Oscillator, Net, etc.	Floorplan View Physical View
Floorplan View	Assignable PIO site	Device View Physical View Package View
	Placed PIO	Device View Netlist View NCD View Physical View Package View
	Placed IOL	Device View Netlist View NCD View Physical View
	Any non-PIO site	Device View Physical View
	Any non-PIO placed component*	Device View Netlist View NCD View Physical View
	Port	Physical View
	UGROUP, REGION	Spreadsheet View
Physical View	Site	Device View Floorplan View
	Placed component	Device View Netlist View NCD View Floorplan View
	Net	NCD View Floorplan View

***Note:** To cross-probe PFF or PFU components, select individual slices.

10. Programming Files

After you have created and verified your design, you can use the final output data file to download or upload a bitstream to or from an FPGA device using the Diamond Programmer.

The bitstream file contains all configuration information from the physical design that define the internal logic and interconnections of the FPGA, as well as device-specific information from other files associated with the target device. Figure 10.1 shows the Export Files options in Diamond Process view and Table 10.1 lists all the final output data file formats and their descriptions.

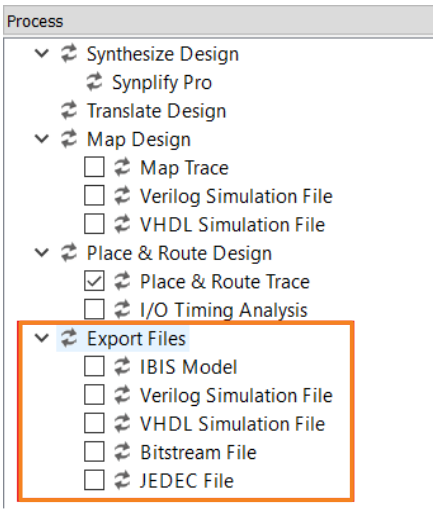

Figure 10.1. Export Files Process Options

Table 10.1. Final Output Data File Formats and their Descriptions

File Formats	Descriptions
Bit File (binary) (.bit)	Binary bitstream files are the default output of the bitstream process and contain the configuration information in bitstream (zeroes and ones) that is represented in the physical design (.ncd) file
Raw Bit File (ASCII) (.rbt)	The Raw Bit File is a text file containing ASCII ones and zeros representing the bits in the bitstream file. If you are using a microprocessor to configure a single FPGA, you can include the Raw Bit file in the source code as a text file to represent the configuration data. The sequence of characters in the Raw Bit file is the same as the bit sequence that is written into the FPGA. The .rbt file differs from the .bit file in that it contains design information in the first six lines
Hex Mask File (.msk)	Used to compare relevant bit locations for executing a read back of configuration data contained in an operating FPGA.
Bit Generation Report File (.bgn)	Outputs information on a bit generation (bitgen) run and displays information on options that are set. This file is output by default and given the name, design_name>.bgn
JEDEC	The programming standard from the Joint Electron Design Engineering Committee, a committee of programmer and semiconductor manufacturers that provide common standards for programmable issues. Examples include acceptable test characters for PLDs and standard data transfer/programming formats for PLDs. The JEDEC Standard is the industry standard for PLD formats. In Diamond Programmer, JEDEC usually refers to the JEDEC fuse map of your design for the device that you have selected.

For more information, you can refer to Diamond software's Help section or Lattice Diamond 3.12 Programming Tools User Guide. You can find more information on Programming Files in Chapter 5 and Programmer in Chapter 7 of the user guide.

11. Reports

In ISE Design Suite software, the reports created during synthesis or implementation can be accessed under **Design Summary**.

In Diamond software, you can click on the **Reports** tab, which is open by default on Diamond workspace. Alternately, you can go to **View** > **Reports**.

Each step on the design flow has its own set of reports from resource usage to timing analysis as shown in Figure 11.1.

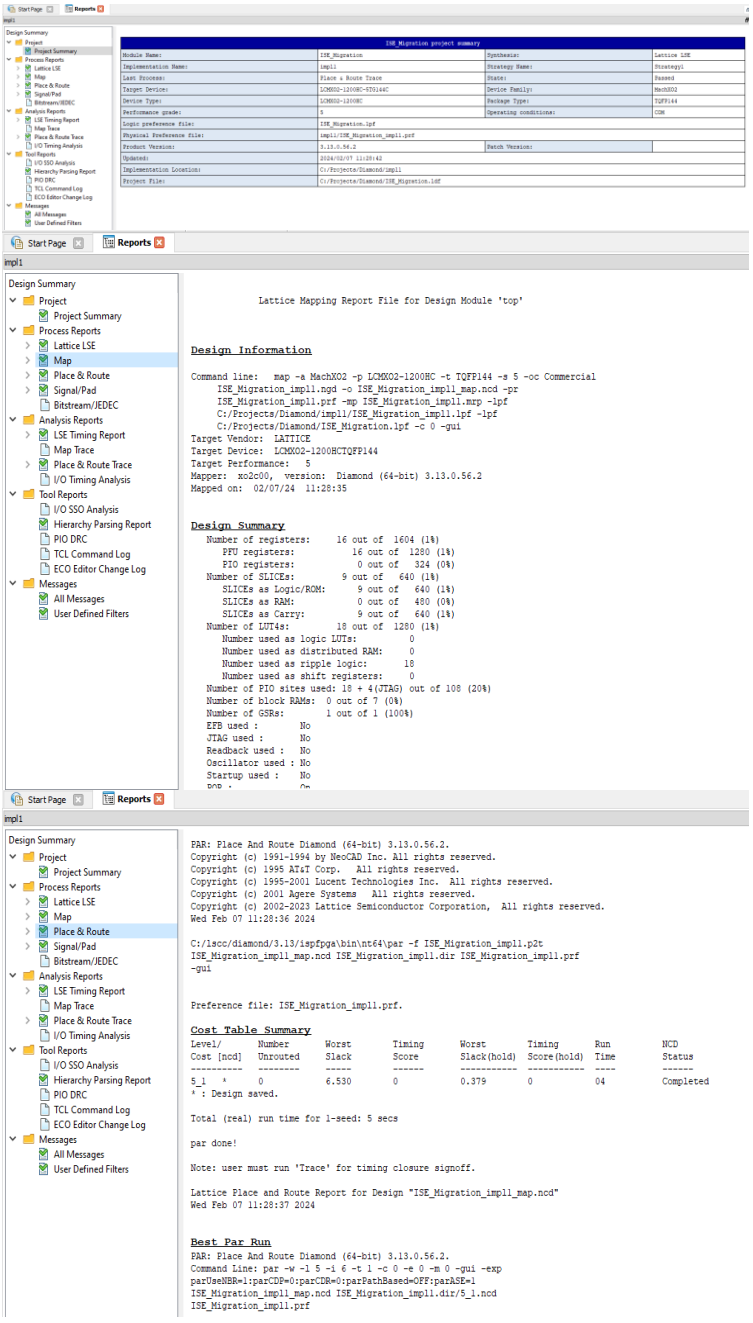

Figure 11.1. Reports in Diamond Software

For more information you can refer to Diamond software's Help section or Lattice Diamond 3.12 User Guide. You can find more information on Reports in Chapters 4 and 7 of the user guide.

References

- Lattice Diamond 3.12 User Guide
- Lattice Diamond 3.12 Programming Tools User Guide
- Lattice Diamond Software web page
- Lattice Solutions IP Cores web page
- LatticeECP3 web page
- Lattice Insights web page for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/en/Support/AnswerDatabase.

Revision History

Revision 1.0, February 2024

Section	Change Summary
All	Initial release.

www.latticesemi.com