



Release Notes for Lattice Diamond 3.5

Welcome to Lattice Diamond[®], the complete design environment for Lattice Semiconductor FPGAs. Lattice Diamond design software offers leading-edge design and implementation tools optimized for cost-sensitive, low-power Lattice FPGA architectures.

Diamond is available for both the Windows and Linux operating systems. For details, see “System Requirements” on page 8.

This version of Diamond adds a variety of enhancements to make designing for Lattice Semiconductor programmable devices easier than ever. The design tools also include support for the latest Lattice Semiconductor devices. See “What’s New” on page 2.

Lattice Semiconductor offers a rich variety of information sources, including the Help system, PDF manuals, tutorials, and online discussions. The easiest way to reach them all is through the online Help. The first topic in the [Help](#) provides links to all the other sources of information.

You can also find extensive information about Diamond and its capabilities, tools, and workflow on the Lattice Semiconductor Web site under:

www.latticesemi.com/latticediamond

What's New

This release of Diamond provides a variety of new features in the following areas. See the online Help for details. Also see "Issues Fixed" on page 11 for known issues of the previous release that have been fixed.

New Device Support The following devices are available with device programming enabled:

- ▶ ECP5 without ES suffix
- ▶ MachXO2-4000HE, industrial grade, in MG184 package

Clarity Designer You can reset placement of individual ports in order to make more incremental changes to your designs. For more information, open the online Help and go to [Entering the Design > Creating Clarity Designer Modules > DDR I/O Planning > Resetting or Modifying Placement](#).

Deployment Tool A new "JEDEC to Hex File Conversion" function type has been added. For more information, open the online Help and go to: [Programming the FPGA > Deploying the Design with the Deployment Tool > Deployment Function Types > File Conversion Deployment Function Type](#).

ECP5 devices now support encrypted dual boot. For more information, open the online Help and go to [Command Line Reference Guide > Command Line Tool Usage > Running the Deployment Tool from the Command Line](#).

IPexpress IPexpress™ includes updates to many modules. For more information, open the online Help and go to [Lattice Module Reference Guide](#).

Platform Designer Lattice Synthesis Engine support has been added to Platform Designer. For more information, open the online Help and go to [Designing with Lattice Diamond Platform Designer > Launching Platform Designer > Creating a New Platform Designer Project](#).

Programmer You can now change the way Programmer writes the paths to data files in the .xcf file when using Programmer on Windows. You can specify either cross-platform compatible format (forward slashes - "/") or Native Delimiter format (backslashes - "\"). For more information, open the online Help and go to [Programming the FPGA > Using Programmer > Creating a New Programmer Project](#).

Programmer now supports programming Platform Manager 2 plus ASC devices and MachXO2 plus ASC devices in a single operation. For more information, open the online Help and go to [Programming the FPGA > Using Programmer > Programming Platform Manager 2 or MachXO2 Devices Using I2C with External ASC](#).

Previously, when the SPI Flash vendor was changed in an existing Programmer project (.xcf file), the Access Mode and Operation were reset to the default mode and operation. This was because vendors do not support the same operations. Now, Programmer checks if the new vendor supports the original operation. If the vendor does, Programmer does not reset the Operation. If the vendor does not, Programmer resets the Operation to the default operation for that SPI Flash vendor.

Simulation (Windows version) The Aldec® [Active-HDL™ Lattice Edition](#) simulator has been updated.

Synthesis Tools The Synopsys® [Synplify Pro®](#) for Lattice and Lattice Synthesis Engine (LSE) synthesis tools have been updated.

LSE now supports LatticeECP2/M, LatticeECP3, and LatticeXP2 in addition to ECP5, MachXO, MachXO2, MachXO3L, Platform Manager, and Platform Manager 2.

LSE has the following new options:

- ▶ Allow Duplicate Modules allows the design to keep duplicate modules.
- ▶ Decode Unreachable States forces synthesis to infer safe recovery logic from unreachable states in all the state machines of the design.
- ▶ DSP Style specifies whether DSP modules should be implemented with DSP resources or with logic.
- ▶ DSP Utilization specifies the percentage of DSP sites that LSE should try to use.

Also, the following defaults have changed:

- ▶ Use IO Registers defaults to Auto. Auto enables register packing if Optimization Goal is set to Area. Otherwise, Auto disables packing.
- ▶ Optimization Goal defaults to Timing for larger devices: ECP5, LatticeECP2, LatticeECP3, and LatticeXP2. It continues to default to Balanced for smaller devices, such as MachXO and Platform Manager.

For more information, open the online Help and go to [Strategy Reference Guide > LSE Options](#).

Supported Devices

Lattice Diamond can be used with either a free license or a subscription license. The two licenses provide access to different device families.

Device Family	Free License	Subscription License
ASC	◀	◀
ECP5U	◀	◀
ECP5UM		◀
LatticeEC™	◀	◀
LatticeECP™	◀	◀
LatticeECP2™	◀	◀
LatticeECP2M™		◀
LatticeECP2S		◀

Device Family	Free License	Subscription License
LatticeECP2MS		◀
LatticeECP3™		◀
LatticeSC™		◀
LatticeSCM™		◀
LatticeXP™	◀	◀
LatticeXP2™	◀	◀
MachXO™	◀	◀
MachXO2™	◀	◀
MachXO3L	◀	◀
Platform Manager™	◀	◀
Platform Manager 2	◀	◀

Updating Projects from an Earlier Version

If you want to work on a design project created with an earlier version of Diamond, start with the following procedures. These procedures adapt the project for the changes in Diamond.

Find out which version of Diamond your project was created with. Then work through the changes for that and every later version, starting with the earliest and going to the most recent. For example, if your project was created with Diamond 1.1, you would start with the changes for 1.1. After completing those changes, you would work on the changes for 1.2, then 1.4, and so on.

When you open a project from Diamond 1.2 or earlier, Diamond opens a dialog box warning that Diamond will automatically move all SDC files to the Synthesis Constraint Files folder in File List view and remove the “Input SDC Constraint File” options from the strategies. If the project is using LSE, the file names will be changed to use an .lsc extension.

Once saved, the project will not be compatible with earlier Diamond versions.

2.2 Projects

ECP5 does not use the CIN port of the CCU2C Carry Chain primitive. This port should not be connected to anything. If the port is connected, the Design Map stage will fail with an error message. If you see such a failure, correct the design in one of the following ways:

- ▶ Rerun synthesis. This should correct the problem if the CCU2C primitive is part of an IPexpress module.

- ▶ If the CCU2C primitive was added to your HDL manually, edit the code to remove the connection. See the following examples:

In Verilog:

```
CCU2C addsub_0 (.A0(scuba_vlo), .A1(DataA[0]),
               .B0(scuba_vlo), .B1(DataB[0]), .C0(scuba_vhi),
               .C1(scuba_vhi), .D0(scuba_vhi), .D1(scuba_vhi),
               .CIN(), .S0(), .S1(Result[0]), .COUT(co0));
```

In VHDL:

```
signal tmp: std_logic := 'X';
cnt_cia: CCU2C
  generic map (INJECT1_1=> "NO", INJECT1_0=> "NO",
              INIT1=> X"0000", INIT0=> X"0000")
  port map (A0=>scuba_vhi, A1=>scuba_vhi, B0=>scuba_vhi,
           B1=>scuba_vhi, C0=>scuba_vhi, C1=>scuba_vhi,
           D0=>scuba_vhi, D1=>scuba_vhi,
           CIN=>tmp, S0=>open, S1=>open, COUT=>cnt_ci);
```

2.0.1 Projects

The default values of several strategy options were changed. If you are using Synplify Pro in integrated mode (running synthesis automatically in Diamond), check that the following settings are still as you want them. Also, check the setting of the Auto Hold-Time Correction option under Place & Route Design. Its default changed to On for all devices.

Table 1: New Default Values for Synplify Pro for Lattice

Option	Before	Now
Fanout Limit is now Fanout Guide	100	1000
Export Diamond Settings to Synplify Pro GUI (new in 2.2)	Not available	No
Fix Gated Clocks and Fix Generated Clocks combined into new Clock Conversion	3 (converts and reports all sequential elements)	True (converts with no report)
Frequency	200	auto (blank means "auto")
Number of Critical Paths	3	blank (unspecified)
Number of Start/End Points	0	blank (unspecified)
Output Preference File	False	True
Pipelining and Retiming	False	Pipelining Only
Resolved Mixed Drivers	True	False
Use Clock Period for Unconstrained I/O	True	False

1.4 Projects

For Diamond 1.4 and earlier, there might be some constraints that are not honored because of the Synplify Pro cross-probing feature. This EDIF renaming is usually related to bus names.

If such a problem occurs, you can turn off the renaming feature by placing the following line in the “Command line Options” text box of the Synplify Pro section of the active strategy:

```
set_option -syn_edif_array_rename 0
```

1.2 Projects

There were several enhancements for IP and MachXO2.

IP Incompatibilities

SPI4.2 2.7 is not compatible with Diamond 1.3 or later. If you are using this IP, check the Lattice Semiconductor Web site for a more recent version.

MachXO2 Changes

See if your design involves any of the following features:

- ▶ For EFB modules with user flash memory (UFM), regenerate the module.
- ▶ For IO_TYPE=PCI33 on a MachXO2-1200 or larger device, check if the CLAMP is using the default setting. With Diamond 1.3 the CLAMP default changes from ON to PCI and the I/O will be placed in bank 2. If you were using the default and still want the setting to be ON, you need to set it explicitly.
- ▶ For PCI33 MT 6.5 and PCI33 T 6.4 IP, either set the CLAMP to ON explicitly or choose a bigger package (256 or more).

1.1 or 1.0 Projects

There were several enhancements for IP and MachXO2.

IP Incompatibilities

The following IP versions are not compatible with Diamond 1.2 or later. If you are using any of these IP, check the Lattice Semiconductor Web site for a more recent version.

- ▶ Convolution Block Encoder 3.6
- ▶ Interleaver Deinterleaver 3.5
- ▶ DDR1 6.9
- ▶ PCI_MT_33 6.4
- ▶ DDR2 7.1
- ▶ PCIe RC Lite 1.2
- ▶ DDR3 1.2.1
- ▶ Tri-Speed MAC 3.4
- ▶ DDR1_CP 1.1 with MachXO2
- ▶ Viterbi Block Decoder 4.6
- ▶ DDR2_CP 1.1 with MachXO2

MachXO2 Support

Some aspects of the software support for MachXO2 designs have been improved. See if your design involves any of the following features:

- ▶ The 4K/7K design with PLL has a CIB-to-PLL jump change. If you are using this design, recompile it.
- ▶ The EFB simulation model has changed. If you are using the EFB module, rerun your simulation tests to see more accurate results.
- ▶ In the DDR_GENERIC module of IPexpress, the GDDR1_RX.Aligned with PLL interface is no longer supported. If you are using such a module, use IPexpress to regenerate it without the PLL option.

Also, MachXO2 has IP evaluation capability and TransFR mode for all I/Os.

Migrating ispLEVER Projects

Diamond uses a different project structure than ispLEVER and cannot directly open an ispLEVER project. However, design projects created in ispLEVER can easily be imported into Diamond. The process is automatic except for the ispLEVER process properties, which are similar to the Diamond strategy settings, and some modules and IPs. All of your ispLEVER project source will be automatically handled.

Projects created using ispLEVER can be imported into Lattice Diamond through two different paths:

- ▶ On the Start Page, click **Import ispLEVER Project** (in the upper-left corner).
- ▶ From the File menu, choose **Open > Import ispLEVER Project**.

Follow the directions in the dialog box that opens to convert your ispLEVER project into a Lattice Diamond project.

Limitations to the import/conversion process include:

- ▶ NGO files in ispLEVER projects need to be manually copied into the Lattice Diamond project if the NGO files were originally copied into the ispLEVER project. For example, NGO files that were copied from Lattice IP generation.
- ▶ The .lpc files are replaced with .ipx files in Lattice Diamond. You need to regenerate your IP by double-clicking on the .lpc file. The resultant wizard will help you generate the new .ipx file, replacing the old .lpc file.

More information on importing ispLEVER projects can be found in the *Lattice Diamond User Guide*, online Help (see [Managing Projects > Importing ispLEVER Projects](#)), and training videos on the Lattice Web site.

Other Information Resources

Other available information resources for the Diamond software include the following.

- ▶ General Information: General information on Lattice Diamond can be found on the Lattice Web site at:
www.latticesemi.com/latticediamond
- ▶ Online Help: Start Lattice Diamond and choose **Help > Lattice Diamond Help**.
- ▶ *Lattice Diamond User Guide*: This document can be found from a link on the Start Page view.
- ▶ Training Videos: Several short videos are available on different aspects of the Lattice Diamond software. These can be viewed online at:
www.latticesemi.com/latticediamond
Click the **Videos** tab.

System Requirements

The basic system requirements for Lattice Diamond are:

- ▶ Intel Pentium or Pentium-compatible PC, or AMD Opteron system support (Linux only)
- ▶ CPU with the SSE3 instruction set to run the Aldec Active-HDL Lattice Edition simulator
- ▶ One of the following operating systems:
 - ▶ Windows Vista (32-bit), Windows 7 (32-bit or 64-bit), or Windows 8 (32-bit or 64-bit).

- ▶ Red Hat Enterprise Linux 4.X, 5.3, or 6. The host operating system can be either 32-bit or 64-bit.
Version 5.3 of Red Hat Enterprise Linux has some extra installation requirements. See “Configuring Red Hat 5.3” on page 10.
- ▶ Novell SUSE Linux Enterprise 10 SP1 or 11 operating system. Novell SUSE Linux supports 32-bit only.
- ▶ Approximately 5.75 GB free disk space
- ▶ RAM adequate for your FPGA design. For guidelines see “Memory Requirements” on page 9.
- ▶ Network adapter and, for a floating license, network connectivity
A node-locked license is based on the physical (hard-coded) address provided by the network adapter. Network connectivity is not required for a node-locked license. In the absence of a network connection, you can install the NWLink IPX/SPX protocol to force recognition of your NIC card ID (see the installation notice).
A floating license requires access to the license server, so both a network adapter and connectivity are required.
- ▶ JavaScript-capable Web browser
- ▶ Microsoft Internet Explorer 8 or higher if using the included Aldec Active-HDL Lattice Edition simulator
- ▶ Acrobat Reader 5.0 or later

Memory Requirements

Table 2 lists the minimum memory requirements and the recommended memory for the Lattice Semiconductor devices supported by Diamond.

On Windows, designing for the largest FPGAs may require more than the usual 2 GB of memory found in 32-bit computers. For help in extending your memory to 3 GB, see “Extending Memory on Windows” on page 10. Designing for LatticeECP3 with more than 95K LUT on a Windows system requires a 64-bit operating system.

Table 2: Recommended Memory

Device	Size	32-Bit Operating Systems		64-Bit Operating Systems	
		Minimum	Recommended	Minimum	Recommended
ECP5	All	2 GB	3 GB	4 GB	6 GB
LatticeEC, LatticeECP	Up to 20K LUT	512 MB	768 MB	1 GB	1.5 GB
	Up to 50K LUT	768 MB	1 GB	1.5 GB	2 GB
LatticeECP2/M	Up to 20K LUT	768 MB	1 GB	1.5 GB	2 GB
	Up to 50K LUT	1 GB	1.5 GB	2 GB	3 GB
	Up to 100K LUT	1 GB	2 GB	2 GB	4 GB

Table 2: Recommended Memory (Continued)

Device	Size	32-Bit Operating Systems		64-Bit Operating Systems	
		Minimum	Recommended	Minimum	Recommended
LatticeECP3	Up to 95K LUT	2 GB	3 GB	4 GB	6 GB
	Up to 150K LUT	3 GB	4 GB	6 GB	8 GB
LatticeSC/M	Up to 40K LUT	768 MB	1 GB	1.5 GB	2 GB
	Up to 115K LUT	1 GB	2.5 GB	2 GB	5 GB
LatticeXP, LatticeXP2	Up to 20K LUT	512 MB	768 MB	1 GB	1.5 GB
	Up to 50K LUT	768 MB	1 GB	1.5 GB	2 GB
MachXO, MachXO2, MachXO3L	All	256 MB	512 MB	512 MB	1 GB
Platform Manager, Platform Manager 2	All	256 MB	512 MB	512 MB	1 GB

Extending Memory on Windows

Designing for LatticeECP3 or ECP5 may require more than the 2 GB normally available with 32-bit Windows systems. But you can configure Windows to use up to 3 GB of memory.

Note that increasing the amount of memory available to applications decreases the amount available for the file cache, paged pool, and nonpaged pool, which can affect applications with heavy networking or I/O.

Use the **BCDEdit /set increaseuserva 3072** command to set the boot entry option to 3 GB. For details, see Microsoft article “BCDEdit /set”:
msdn.microsoft.com/en-us/library/ff542202.aspx

Configuring Red Hat 5.3

Red Hat Enterprise Linux 5.3 has some extra requirements for Diamond:

- ▶ In addition to the basic installation of Red Hat 5.3, under Development/ Legacy Software Development, select:

```
1:gtk+-1.2.10-56.el5.i386 - GIMP Toolkit (GTK+) sb:(9 of 9)
```

Under Base System/Legacy Software Support, add the following to the default items:

```
Openmotif22-2.2.3-18.i386 - Open Motif runtime
```

Proper Diamond operation depends upon these libraries being installed.

- ▶ When installing the Red Hat Enterprise Linux version, be sure to install the PERL modules XML::Parser, XML::DOM, and XML::RegExp. These PERL modules are available at www.cpan.org.

Issues Fixed

The following known issues are fixed with this release. Their workarounds are no longer needed. For the complete list of known issues, see: www.latticesemi.com/view_document?document_id=50676

Deployment Tool

SVF-Flash Refresh option fails to generate the SVF File if the Rev-D Standard was selected

If the Flash Refresh operation and the Rev-D Standard option are selected, the Deployment Tool fails to generate the SVF file.

Devices affected: MachXO2
CR124160

Embedded I2C files generated from Diamond Deployment Tool fails for Verify CFG

The Embedded I2C files generated for the MachXO2 family fails to program and verify the Configuration memory block.

Devices affected: MachXO2
CR123587

Model 300

Model 300 reports an error when programming a LatticeXP2 with an encrypted JEDEC file

Model 300 reports an error message when trying to program a LatticeXP2 with an encrypted JEDEC file.

Devices affected: LatticeXP2
CR123941

Programmer

Programing status is not updated when programming a chain of devices

When programming a chain of devices with three or devices, the status column for the third device, for example, shows NA until that row is selected.

Devices affected: All
CR124200

Programmer does not save the file name in the XCF file when a different JEDEC file is selected

When selecting a different JEDEC file in an existing XCF file with the XFLASH Verify Feature Rows operation, Programmer does not save the new JEDEC file name in the XCF file.

Devices affected: MachXO2
CR124146

Programmer changes the selected operation to the default operation

When opening an existing XCF file with the XSRAM Bypass operation, Programmer resets the operation to the default operation, SRAM Fast Program.

Devices affected: MachXO2
CR124125

Programmer reports an error message when creating an XCF file from Diamond

After generating a MachXO3LF JEDEC file using Diamond, if a new Integrated Programmer XCF file is created, Programmer reports an error message saying the XCF file was invalid.

Devices affected: MachXO3LF
CR123935

Diamond Programmer incorrectly generates error message for XO3L-4300C JEDEC file

Programmer incorrectly reports an error message when the JEDEC file for the LCMXO3L-4300C is selected.

Devices affected: MachXO3L
CR123820

Programmer fails to load USB driver for the HW-USB-2A USB cable on Windows 8.1

Diamond Programmer occasionally reports the error message "failed to load usb driver" when using the HW-USB-2A USB cable on Windows 8.1.

Devices affected: All
CR123791

Diamond Programmer command line reports “Cannot connect to X Server” on Linux

Diamond Programmer reports “Cannot connect to X Server” error message when running command line on Linux.

Devices affected: All
CR123720

Not able to program a chain of MachXO3L devices in Turbo mode

Diamond Programmer crashes when programming the NVCM of a chain of MachXO3L devices in Turbo mode.

Devices affected: MachXO3L
CR123716

Data File Size of SPI Flash device resets to zero

When the Operation for an SPI Flash device is changed in an existing Programmer project (.xcf file), the Data File Size occasionally resets to zero.

Devices affected: All

Clicking Load from File resets verification file

When the Verification file for an SPI Flash device is changed and the Load from File button is clicked, the Verification file occasionally resets to the original file.

Devices affected: All

Known Issues

Following are known issues with this release and workarounds for them. For the complete list, see:

www.latticesemi.com/view_document?document_id=50676

Design Entry

Diamond Project Navigator does not stop process flow when SBX files are set to a different synthesis tool

If your project's synthesis tool is set to Lattice Synthesis Engine (LSE), but the Clarity Designer .sbx file was generated using Synplify Pro, then when opening the file you will receive a message stating that all components need to be reconfigured.

If there is a Clarity Designer .sbx file in your Diamond project, and if you changed synthesis tool in the Diamond software, double-click the .sbx file to open Clarity Designer, reconfigure all the modules, and regenerate the design.

Devices affected: ECP5U
CR124372

PERIOD constraint set by Lattice Synthesis Engine LPF may over-write FREQUENCY constraint set by user LPF

LSE always generates a logical preference file (LPF). This LPF is named as *<project name>_<implementation name>.lpf*. This LPF is passed into MAP if the "Use LPF Created from SDC in Project" strategy setting is set to TRUE.

If you see a clock in PERIOD constraint that was set in the LSE LPF and a FREQUENCY constraint for the same clock set from the user LPF, then the tool may pick up the unwanted PERIOD constraint from LSE. The workaround is to write the user LPF with PERIOD on the affected clock net.

Devices affected: All devices supported by LSE
CR124293

Global FSM Binary encoding does not work in Lattice Synthesis Engine

LSE infers a finite state machine (FSM) coded in Binary and encodes it to One-Hot/Gray based on the user directive. But if the user codes the FSM in One-Hot in the RTL, LSE does not re-target the FSM to Binary/Gray encodings.

Devices affected: All
CR124331

Contacting Technical Support

FAQs The first place to look. The [Answer Database](#) on the Lattice Semiconductor Web site provides solutions to questions that many of our customers have already asked. Lattice Applications Engineers are continuously adding to the Database.

Technical Support Assistance Submit a technical support case via www.latticesemi.com/techsupport.

For Local Support Contact your nearest [Lattice Sales Office](#).

Trademarks

All Lattice trademarks are as listed at www.latticesemi.com/legal. Synopsys and Synplify Pro are trademarks of Synopsys, Inc. Aldec and Active-HDL are trademarks of Aldec, Inc. All other trademarks are the property of their respective owners.