



Lattice Diamond 1.4 Installation Notice

Linux

Lattice Semiconductor Corporation
5555 NE Moore Court
Hillsboro, OR 97124
(503) 268-8001

December 2011

Copyright

Copyright © 2011 Lattice Semiconductor Corporation.

This document may not, in whole or part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Lattice Semiconductor Corporation.

Trademarks

Lattice Semiconductor Corporation, L Lattice Semiconductor Corporation (logo), L (stylized), L (design), Lattice (design), LSC, CleanClock, E²CMOS, Extreme Performance, FlashBAK, FlexiClock, flexiFlash, flexiMAC, flexiPCS, FreedomChip, GAL, GDX, Generic Array Logic, HDL Explorer, IPexpress, ISP, ispATE, ispClock, ispDOWNLOAD, ispGAL, ispGDS, ispGDX, ispGD XV, ispGDX2, ispGENERATOR, ispJTAG, ispLEVER, ispLeverCORE, ispLSI, ispMACH, ispPAC, ispTRACY, ispTURBO, ispVIRTUAL MACHINE, ispVM, ispXP, ispXPGA, ispXPLD, Lattice Diamond, LatticeCORE, LatticeEC, LatticeECP, LatticeECP-DSP, LatticeECP2, LatticeECP2M, LatticeECP3, LatticeMico8, LatticeMico32, LatticeSC, LatticeSCM, LatticeXP, LatticeXP2, MACH, MachXO, MachXO2, MACO, ORCA, PAC, PAC-Designer, PAL, Performance Analyst, Platform Manager, ProcessorPM, PURESPEED, Reveal, Silicon Forest, Speedlocked, Speed Locking, SuperBIG, SuperCOOL, SuperFAST, SuperWIDE, sysCLOCK, sysCONFIG, sysDSP, sysHSI, sysI/O, sysMEM, The Simple Machine for Complex Design, TransFR, UltraMOS, and specific product designations are either registered trademarks or trademarks of Lattice Semiconductor Corporation or its subsidiaries in the United States and/or other countries. ISP, Bringing the Best Together, and More of the Best are service marks of Lattice Semiconductor Corporation.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

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.
<i><Italic></i>	Variables in commands, code syntax, and path names.
Ctrl+L	Press the two keys at the same time.
<i>Courier</i>	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

Installing Diamond Tools	1
System Requirements	1
Memory Requirements	2
Required Support Libraries	3
Contacting Technical Support	6
Installer Contents	6
Diamond Installed Directory Structure	7
Installing Lattice Diamond	8
Installing Stand-Alone Reveal Analyzer	9
Installing Stand-Alone Programmer	10
Installing and Configuring USB Cable and Parallel Cable	11
Licensing Lattice Diamond	18
Obtaining a License	18
Editing the License File	19
Starting the License Manager	19
Stopping the License Manager	20
Setting Up a Floating License on Linux	20
Running the Lattice Diamond GUI	20
Using the Examples Directory	21
Finding the Installation History	21
Running Lattice Diamond from the Command Line	22
Running Standalone Tcl Console	22

- Running Using CSH or Bash Interpreters 22
- Running Synplify Pro for Lattice 22
- Running Stand-Alone Reveal Analyzer 23
- Running Stand-Alone Programmer 23
- Running ispVM System 23
 - Setting Environment Variables (C Shell) 23
 - Setting Environment Variables (BASH) 24
 - Invoking ispVM System 25
- Running Multiple Versions of Lattice Software 25
- Updating Lattice Diamond 25
 - Running the UPDATE Tool 26
 - Changing the Automatic Checking Settings 26
 - Changing the Network Setting 27
 - Downloading and Installing Software Updates 28
- Installing LatticeMico Development Tools 29**
 - System and Software Requirements 29
 - System Requirements 29
 - Software Requirements 29
 - Additional Notes for Linux 30
 - Installing LatticeMico Within Diamond Software 31
 - Installing LatticeMico as a Standalone Tool 32

Installing Diamond Tools

This document provides installation instructions for the Lattice Diamond[®] Linux software.

Diamond supports LatticeEC[™], LatticeECP[™], LatticeECP2[™], LatticeECP2M[™], LatticeECP3[™], LatticeSC[™], LatticeSCM[™], LatticeXP[™], LatticeXP2[™], MachXO[™], MachXO2[™], and Platform Manager[™] designs.

Note

The available devices vary depending on the type of license.

System Requirements

The following are the basic system requirements for Lattice Diamond on Linux:

- ◆ Intel Pentium or Pentium-compatible PC, or AMD Opteron system support
For Programmer and ispVM System, it contains a 32-bit driver to support the 32-bit system only.
- ◆ Red Hat Enterprise Linux version 4.X or 5.X, or Novell SUSE Linux Enterprise 10 SP1 or 11 operating system

The host operating system can be either 32-bit or 64-bit. Diamond is a 32-bit application requiring 32-bit support libraries in order run on a 64-bit host operating system.

- ◆ Approximately 5.75 GB free disk space
- ◆ Network adapter and network connectivity

Note

A floating license requires access to the license server, so both a network adapter and connectivity are required.

- ◆ 1024 x 768 graphics display
- ◆ JScript-capable Web browser
- ◆ Adobe Acrobat Reader, or equivalent PDF reader

Memory Requirements

Table 1 lists the minimum memory requirements and recommended memory for all the Lattice Semiconductor FPGA families.

Table 1: Recommended Memory for Linux

Device	Size	Minimum	Recommended
LatticeEC, LatticeECP	Up to 20K LUT	512 MB	768 MB
	Up to 50K LUT	768 MB	1 GB
LatticeECP2/M	Up to 20K LUT	768 MB	1 GB
	UP to 50K LUT	1 GB	1.5 GB
	Up to 100K LUT	1 GB	2 GB
LatticeECP3	Up to 95K LUT	2 GB	3 GB
	Up to 150K LUT	3 GB	4 GB
LatticeSC/M	Up to 40K LUT	768 MB	1 GB
	Up to 115K LUT	1 GB	2.5 GB
LatticeXP, LatticeXP2	Up to 20K LUT	512 MB	768 MB
	Up to 50K LUT	768 MB	1 GB

Table 1: Recommended Memory for Linux (Continued)

Device	Size	Minimum	Recommended
MachXO, MachXO2	All	256MB	512MB
Platform Manager	All	256 MB	512 MB

Required Support Libraries

Diamond is packaged in Red Hat Package Manager format (RPM). Users must download and install specific operating system PERL RPMs. The exact file names are dependent upon the user's operating system. For example:

```
perl-XML-Parser
perl-XML-RegExp
perl-XML-DOM
```

In order to install Diamond, several support libraries must be acquired from the operating system vendor's repository. See the following tables.

Note

It is the responsibility of the person installing the software to ensure that all of the required libraries are present.

Table 2: Diamond Base

ld-linux.so.2	libjpeg.so.62	libX11.so.6
libc.so.6	libm.so.6	libXext.so.6
libdl.so.2	libpthread.so.0	libXft.so.2
libfontconfig.so.1	librt.so.1	libXi.so.6
libfreetype.so.6	libSM.so.6	libXmu.so.6
libglib-2.0.so.0	libtiff.so.3	libXrender.so.1
libgthread-2.0.so.0	libusb-0.1.so.4	libXt.so.6
libICE.so.6	libutil.so.1	

Table 3: Diamond Encryption (Optional)

diamond_1_4-base	libm.so.6
libc.so.6	libpthread.so.0

Note

If some required LatticeMico System support libraries are missing, you may receive the following error message:

```
An SWT error has occurred. You are recommended to exit the
workbench. Subsequent errors may happen and may terminate the
workbench without warning. See the .log file for more details.
Do you want to exit the workbench?
```

Ensure that the LatticeMico System support libraries listed in Table 4 are installed if this error is displayed.

Table 4: LatticeMico System

libasound.so.2	libnspr4.so	libXi.so.6
libatk-1.0.so.0	libnss3.so	libXp.so.6
libc.so.6	libpango-1.0.so.0	libXrender.so.1
libdl.so.2	libpangox-1.0.so.0	libXt.so.6
libexpat.so.0	libpangoxft-1.0.so.0	libXtst.so.6
libfontconfig.so.1	libplc4.so	libz.so.1
libfreetype.so.6	libplds4.so	perl(File::Copy)
libgdk_pixbuf-2.0.so.0	libpng12.so.0	perl(File::Glob)
libgdk-x11-2.0.so.0	libpthread.so.0	perl(File::Path)
libglib-2.0.so.0	librt.so.1	perl(Getopt::Std)
libgmodule-2.0.so.0	libsmime3.so	perl(POSIX)
libgmp.so.3	libSM.so.6	perl(Scalar::Util)
libgobject-2.0.so.0	libsoftokn3.so	perl(strict)
libgthread-2.0.so.0	libssl3.so	Perl(XML::DOM)
libgtk-x11-2.0.so.0	libstdc++.so.5	perl(XML::Parser)
libICE.so.6	libthread_db.so.1	perl(XML::RegExp)
libjpeg.so.62	libusb-0.1.so.4	/usr/bin/perl
libm.so.6	libX11.so.6	/usr/bin/python
libncurses.so.5	libXext.so.6	
libnsl.so.1	libXft.so.2	

Table 5: Diamond Programmer Stand-Alone

libc.so.6	libICE.so.6	libusb-0.1.so.4
libdl.so.2	libm.so.6	libX11.so.6
libfontconfig.so.1	libpng12.so.0	libXext.so.6
libfreetype.so.6	libpthread.so.0	libXft.so.2
libglib-2.0.so.0	librt.so.1	libXrender.so.1
libgthread-2.0.so.0	libSM.so.6	libz.so.1

Table 6: Diamond Programmer Encryption (Optional)

programmer_1_4	libgcc_s.so.1	libpthread.so.0
libc.so.6	libm.so.6	

Table 7: Reveal Analyzer Stand-Alone

libc.so.6	libICE.so.6	libusb-0.1.so.4
libdl.so.2	libm.so.6	libX11.so.6
libfontconfig.so.1	libpng12.so.0	libXext.so.6
libfreetype.so.6	libpthread.so.0	libXft.so.2
libgcc_s.so.1	librt.so.1	libXrender.so.1
libglib-2.0.so.0	libSM.so.6	

Contacting Technical Support

FAQs The first place to look. The [Lattice FAQs](#) (frequently asked questions) provide solutions to questions that many of our customers have already asked. Lattice Applications Engineers are continuously adding to the FAQs.

Online Forums [Lattice Forums](#) contain a wealth of knowledge and are actively monitored by Lattice Applications Engineers.

Telephone Support Hotline Receive direct technical support for all Lattice products by calling Lattice Applications from 5:30 a.m. to 6 p.m. Pacific Time.

- ◆ For USA & Canada: 1-800-LATTICE (528-8423)
- ◆ For other locations: +1 503 268 8001

In Asia, call Lattice Applications from 8:30 a.m. to 5:30 p.m. Beijing Time (CST), +0800 UTC. Chinese and English language only.

- ◆ For Asia: +86 21 52989090

E-mail Support

- ◆ techsupport@latticesemi.com
- ◆ techsupport-asia@latticesemi.com

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

Installer Contents

The Lattice Diamond RPM installation packages are available for download from the [Lattice Semiconductor Web site](#).

The following describes the contents of the Lattice Diamond installer.

- ◆ The Lattice Diamond “base” RPM installs Lattice Diamond and Synplify Pro for Lattice. See “Installing Lattice Diamond” on page 8 for more information.
- ◆ The Reveal installation file (.rpm) installs the stand-alone Reveal Analyzer. It is not necessary to install this package if the Diamond Base RPM is installed. See “Installing Stand-Alone Reveal Analyzer” on page 9 for more information.
- ◆ The Programmer installation file (.rpm) installs the stand-alone Diamond Programmer. It is not necessary to install this package if the Diamond

Base RPM is installed. See “Installing Stand-Alone Programmer” on page 10 for more information.

- ◆ The LatticeMico installation file (.rpm) installs LatticeMico Development Tools and GNU-based Compiler Tools. See “Installing LatticeMico Development Tools” on page 29 for more information.
- ◆ The Installation Notice contains installation and licensing information.

Diamond Installed Directory Structure

Table 8 shows the Lattice Diamond software directory structure after installation:

Table 8: The Diamond Software Directory Structure

File or Directory	Description
bin/lin/	This directory contains files for running the Lattice Diamond GUI.
cae_library/	This directory contains synthesis header libraries for Synplify Pro in both Verilog and VHDL formats. It also contains the libraries for interface kits (ispLSI [®] design and simulation libraries).
data/	This directory contains files for the Lattice Diamond software.
docs/	This directory contains Lattice Diamond documentation, including manuals, tutorials, and the online Help.
examples/	This directory contains Lattice design examples.
ispfpga/	This directory contains files for the Lattice Diamond software.
license/	This directory contains the license agreement.
module/	This directory contains Lattice module files.
synpbase/	This directory contains files for the Synplify Pro for Lattice software.
tcltk/	This directory contains tcl/tk related files.

Installing Lattice Diamond

Before installing the Lattice Diamond software, download the installation files from the Lattice Semiconductor Web site. For more information on how to download the Diamond software, go to www.latticesemi.com/latticediamond.

RPM installation files are provided to simplify the installation process. The RPM file has a companion MD5 checksum file. The MD5 file permits you to validate the integrity of the RPM file.

Note

- ◆ You need root privilege to install an RPM package.
 - ◆ In the Diamond installation, blank space characters are not allowed in the installation directory path name.
-

Lattice Diamond is installed with the **diamond_1_4-base-xx-i386-linux.rpm** installation file. The file installs all Lattice Semiconductor FPGA devices on your system. It also installs the Synplify Pro for Lattice software. The following section guides you through the installation procedure step by step.

To install the Diamond software:

1. Verify the integrity of the first RPM file. Make sure the companion MD5 file is located in the same directory as the RPM file.

```
cd <directory_with_RPM>
rpm -K diamond_1_4-base-xx-i386-linux.rpm
```

2. Install the Diamond software. The default installation directory for Diamond is `/usr/local/diamond`. You have the option of changing the installation directory.

- ◆ To install to the default location (`/usr/local/diamond`):

```
sudo rpm -Uvh <path_to_package>/diamond_1_4-base-xx-i386-
linux.rpm
```

- ◆ To override the default location:

```
sudo rpm -Uvh --prefix <install_path> <path_to_package>/
diamond_1_4-base-xx-i386-linux.rpm
```

Note

The `--prefix` can not be used to install directly to Network Attached Storage. The `--prefix` must point to a location writable by the `sudo` command.

If networked installation is desired, install Diamond locally, and then transfer the file structure to your Network Attached Storage system.

Important!

The RPM installation process for Diamond verifies the Linux operating system has all of the required support libraries installed. If library files required for running Diamond are not installed on the system Diamond will not be installed.

If you want to force the installation to continue, you can use “`--nodeps`” on the `rpm` command to ignore the dependency information. Diamond may not function correctly if the “`--nodeps`” switch is used to force installation of Diamond to proceed.

Installing Stand-Alone Reveal Analyzer

Reveal Analyzer is included in the Diamond installation. If you want to use the tool without installing Lattice Diamond, use the **reveal_1_4-xx-i386-linux.rpm** file to install the stand-alone Reveal Analyzer.

To install the stand-alone Reveal Analyzer:

1. Verify the integrity of the RPM file. Make sure the companion MD5 file is located in the same directory as the RPM file.

```
cd <directory_with_RPM>
rpm -K reveal_1_4-xx-i386-linux.rpm
```

2. Install the Reveal Analyzer software. The default installation directory for Reveal Analyzer is **/usr/local/reveal**. You have the option of changing the installation directory.

- ◆ To install to the default location (`/usr/local/reveal`):

```
sudo rpm -Uvh <path_to_package>/reveal_1_4-xx-i386-
linux.rpm
```

- ◆ To override the default location:

```
sudo rpm -Uvh --prefix <install_path> <path_to_package>/
reveal_1_4-xx-i386-linux.rpm
```

Note

The `--prefix` can not be used to install directly to Network Attached Storage. The `--prefix` must point to a location writable by the `sudo` command.

If networked installation is desired, install Diamond locally, and then transfer the file structure to your Network Attached Storage system.

Installing Stand-Alone Programmer

Programmer is included in the Diamond installation. If you want to use the tool without installing Lattice Diamond, use the **programmer_1_4-xx-i386-linux.rpm** file to install the stand-alone Programmer.

To install the stand-alone Programmer:

1. Verify the integrity of the RPM file.

```
cd <directory_with_RPM>
rpm -K programmer_1_4-xx-i386-linux.rpm
```

2. Install the Programmer software. The default installation directory for Programmer is **/usr/local/programmer**. You have the option of changing the installation directory.

- ◆ To install to the default location (`/usr/local/programmer`):

```
sudo rpm -Uvh <path_to_package>/programmer_1_4-xx-i386-
linux.rpm
```

- ◆ To override the default location:

```
sudo rpm -Uvh --prefix <install_path> <path_to_package>/
programmer_1_4-xx-i386-linux.rpm
```

Note

The `--prefix` can not be used to install directly to Network Attached Storage. The `--prefix` must point to a location writable by the `sudo` command.

If networked installation is desired, install Diamond locally, and then transfer the file structure to your Network Attached Storage system.

Installing and Configuring USB Cable and Parallel Cable

This section provides information on USB cable and parallel cable installation and configuration.

Parallel Port Cable Configuration

Before using the ispDOWNLOAD[®] parallel port cable, the computer's parallel port must be enabled and have the correct read/write permission. Contact the system administrator if the parallel port is not enabled or does not have the correct read/write permission.

You can check the current permissions of the parallel port by navigating to the `/dev` directory and doing a lookup on the port, as follows

```
cd /dev ls -l parport0
```

There are two main ways to give read/write permission to the parallel port. The first is to give read/write permission to the parallel port to anyone that has access to that machine. The second is more secure, where only selected individuals will have read/write permission to the parallel port that has access to that machine. Below are the descriptions on how to setup parallel port zero to have read/write permission.

Giving Parallel Port Zero Read/Write Permission to All Users

1. Log in as root.
2. Change directory to the `/dev` directory as follows:

```
cd /dev <return>
```

3. Give read/write permission to parallel port zero (parport0) as follows:

```
chmod 666 parport0 <return>
```

Giving Parallel Port Zero Read/Write Permission to Selected Individuals

There are two files to change on the machine to give selected individuals read/write permission to parallel port zero. Both files are located in the `/etc` directory. The two files are "group" and "nsswitch.conf".

1. Log in as root.
2. Change directory to the `/dev` directory as follows:

```
cd /dev <return>
```

3. Give read/write permission to parallel port zero (parport0) to owner and group as follows:

```
chmod 666 parport0 <return>
```

4. Change directory to the `/etc` directory as follows:

```
cd /etc <return>
```

5. Using your favorite text editor, edit the file “group” as follows:

Add the login name for all the users you want to give read/write permission for parallel port zero by adding the login names to the “lp:x:7:” line. This line may vary from system to system. Two examples are shown below.

Example 1: Change:

```
lp:x:7:
```

to:

```
lp:x:7:user_name1,user_name2,etc
```

Example 2: Change:

```
lp:x:7:daemon,lp
```

to:

```
lp:x:7:daemon,lp,user_name1,user_name2,etc
```

where `user_name1, etc.`, are the login names for all the users you want to give read/write permission for parallel port zero. Save the changes and close the “group” file.

6. Using your favorite text editor, edit the file “nsswitch.conf” as follows:

Look at the group line. Ensure “files” is there.

For example:

```
group: files nis
```

7. Reboot for the changes to take effect.

Lattice USB2A USB Download Cable Configuration

Programmer software can directly access USB devices using libusb, which accesses the USB file system. To verify that you have the USB library installed, do the following:

1. Plug in the USB cable.
2. On the command line, type:

```
/sbin/lsusb
```

You should see an entry similar to the following:

```
> Bus 002 Device 002: ID 1134:8001
```

3. If you receive an error when you type `/sbin/lssusb`, you probably do not have `libusb` installed. To install the `libusb`, go to:

<http://libusb.sourceforge.net/>

4. Unplug the cable.

Note

The USB cable must be disconnected from the PC when performing the installations. You must have root access and be able to log in as a super user.

Driver Setup for Lattice USB2A Download Cable on RedHat Linux Enterprise 5 and Newer

Manual UDEV Method Configuration

1. Create a working file called `10-local.rules`
2. Add the following information to the `10-local.rules` file

```
#Lattice
BUS=="usb",ACTION=="add",SYSFS{idVendor}=="1134",SYSFS{idProduct}=="8001", \
MODE=="0660",OWNER=="root",SYMLINK+="lattice-%n"
```

Note

Make sure everything after `BUS==` is on one line.

3. Copy the file into the `/etc/udev/rules.d/10-local.rules` directory. If a file by this name already exists, simply append the working file to it. For example:

```
sudo cp 10-local.rules /etc/udev/rules.d/.
```

Or if the file already exists

```
sudo cat 10-local.rules >>/etc/udev/rules.d/10-local.rules
```

4. Create a working file called `"lattice.perms"`.
5. Add the following information to the `lattice.perms` file:

```
<lattice>=/dev/lattice*
```

```
<console>0660 <lattice>0660 root
```

Note

Make sure to enter a new line after lattice*

6. Copy the file into `/etc/security/console.perms.d` directory

```
sudo cp lattice.perms /etc/security/console.perms.d/.
```

Or if the file already exists

```
sudo cat lattice.perms >>/etc/security/console.perms.d/  
lattice.perms
```

7. Make the new UDEV settings active

```
sudo skill -HUP udevd
```

Correctly configured UDEV will create new entries into the `/dev` directory each time a Lattice programming cable is attached to the system. The programming cable will be owned by the user currently logged in at the console. When you edit the file, pay attention to the spacing since some statements are sensitive to the omission of spaces.

For more information on how to setup UDEV rules for Red Hat, go to:

<http://www.redhat.com/magazine/002dec04/features/udev/>

Driver Setup for Lattice USB2A Download Cable on RedHat Linux Enterprise 4 and Older

Manual Hot Plug Method Configuration

Use the following manual configuration instructions. You can also use this method for a customized setup.

1. Edit the `/etc/hotplug/usb.usermap` file as follows:

```
sudo vi /etc/hotplug/usb.usermap
```

2. Append the following line to the end of the file:

```
usbcable 0x0003 0x1134 0x8001 0x0000 0x0000 0x00 0x00 0x00  
0x00 0x00 0x00 0x00000000
```

3. Use one of the following methods to create the script that will be executed each time the USB device is inserted:

- ◆ Manually create the `/etc/hotplug/usb/usbcable` file.

In the following example, the device is owned by “root,” the group is assigned to “usbusers,” and the device is readable and writable by both “root” and “usbusers.”

```
#!/bin/bash if [ "${ACTION}" = "add" ] && [ -f
"${DEVICE}" ] then
chown root "${DEVICE}" chgrp usbusers "${DEVICE}" chmod
660 "${DEVICE}"
fi
```

When you create the usbcable file, pay attention to the spacing since some statements are sensitive to the omission of spaces.

- ◆ Copy the Linux usbcam file and save it as usbcable.

On most versions of Linux, a usbcam script is available that changes the device ownership to the user who inserts the cable and logs into the console. The usbcam script is located in the same `/etc/hotplug/` directory.

If you prefer to use this script, copy the usbcam file and save it as usbcable as follows:

```
sudo cp /etc/hotplug/usbcam /etc/hotplug/usbcable
```

4. Grant the script execute permission as follows:

```
sudo chmod 755 /etc/hotplug/usbcable
```

Note

The USB cable must be removed and reinserted whenever one user logs out and another user logs in. Otherwise, the system will still recognize the previous user as the owner of the cable, and an error message will be generated.

FTDI USB Host Chip Configuration

libftd2xx-A library (using libusb) communicates with the FTDI UART/FIFO devices, including the popular bitbang mode. Programmer uses the libftd2xx.so.0.4.13 version which included in the `<install_path>\data\vmdata\drivers\ftdiusbdriver` directory.

Installation

1. Go to the `<install_path>\data\vmdata\drivers\ftdiusbdriver` directory.
2. Copy the libftd2xx.so.0.4.13 file to `/usr/local/lib` as follows:

```
sudo cp libftd2xx.so.0.4.13 /usr/local/lib/
```

3. Change directory to `/usr/local/lib` as follows:

```
cd /usr/local/lib
```

4. Make a symbolic link to the file in `/usr/local/lib` as follows:

```
sudo ln -s libftd2xx.so.0.4.13 libftd2xx.so
```

5. Change directory to `/usr/lib` as follows:

```
cd /usr/lib
```

6. Make a symbolic link to the driver as follows

```
sudo ln -s /usr/local/lib/libftd2xx.so.0.4.13 libftd2xx.so
```

7. Add the following line to `/etc/fstab`:

```
none /proc/bus/usb usbdevfs defaults,devmode=0666 0 0
```

You may need to use the following command for some distributions:

```
none /proc/bus/usb usbdevfs defaults,mode=0666 0 0
```

(use `usbfs` in 2.6 kernels)

8. Mount all filesystems described in `/etc/fstab` as follows:

```
sudo mount -a
```

If you have problems with these procedures, check with `usbview` (search on the internet for application or it can be sent to you by FTDI) to check the USB file system is mounted properly.

FTDI High Speed USB Host Chip (FT232H) Configuration (Linux 32 Bits operating system only)

`libftd2xx-A` library (using `libusb`) communicates to the FTDI UART/FIFO devices including the popular bitbang mode. Programmer uses the `libftd2xx.so.1.0.4` version which included in the

`<install_path>\data\vmdata\drivers\ftdiusbdriver` directory.

Installation

1. Go to the

```
<install_path>\data\vmdata\drivers\ftdiusbdriver  
directory.
```

2. Copy the `libftd2xx.so.1.0.4` file to `/usr/local/lib` as follows:

```
sudo cp libftd2xx.so.1.0.4 /usr/local/lib/.
```

3. Change directory to `/usr/local/lib` as follows:

```
cd /usr/local/lib
```

4. Make a symbolic link in `/usr/lib` to the file in `/usr/local/lib` as follows:

```
sudo ln -s lib/libftd2xx.so.1.0.4.so.1.0.4 libftd2xx.so
```

5. Change directory to `/usr/lib` as follows:

```
cd /usr/lib.
```

6. Make symbolic links to these files using the following commands:

```
sudo ln -s /usr/local/lib/ libftd2xx.so.1.0.4 libftd2xx.so
```

7. Plug in the cable.

8. Unload the `ftdi_sio` driver if it is attached to your device.

```
sudo /sbin/rmmod ftdi_sio
```

9. Unload the `usbserial` driver if it is attached to your device.

```
sudo /sbin/rmmod usbserial
```

Update Programmer Driver

1. Go to the `<install_path>\1.4\bin\lin` directory.
2. Remove the `libftd2xx.so.0.4.13`, `libftd2xx.so.0` and `libftd2xx.so` files as follows:

```
sudo rm libftd2xx.so.0.4.13 libstd2xx.so.0 libftd2xx.so
```

3. Make symbolic links to these files using the following commands:

```
sudo ln -s libftd2xx.so.1.0.4 libftd2xx.so.0  
sudo ln -s libftd2xx.so.0 libftd2xx.so
```

If you have problems with these procedures, check with `usbview` (search on the internet for application or it can be sent to you by FTDI) to check the USB file system is mounted properly.

Licensing Lattice Diamond

The Lattice Diamond development tool is licensed software. In order for you to launch the design tools you have to configure a Flexera license. The license can be either node-locked to the local machine, or acquired from a license server accessible from a LAN connection.

Note

Diamond permits the creation of configuration bitstreams for all of Lattice Semiconductor's FPGAs and CPLDs. However, a license is required for some devices.

The default location of the license file is `<install_path>/diamond/1.4/license/license.dat`. If this location is changed, you must set the `LM_LICENSE_FILE` environment variable to include the new path name.

Note

In order to change the `LM_LICENSE_FILE` variable, you may need to edit the `.cshrc` or `.bashrc` configuration file located in your home directory.

Obtaining a License

To register and license your Diamond software:

1. Obtain the host ID of your license server with the following command:

```
<install_path>/diamond/1.4/ispfpga/bin/lin/lmutil lmhostid
```
2. Go to the Licensing section of the Lattice Semiconductor Web site (www.latticesemi.com/license) and follow the on-screen instructions.

Note

Lattice Semiconductor supports licensing for a single server or three redundant servers. If you are using three redundant servers, enter all three server host IDs on the License File/Registration Form.

Lattice Semiconductor will send your Diamond license file (`license.dat`) to you by e-mail within one working day. After you receive the license file from Lattice Semiconductor, copy the `license.dat` file to the Diamond license directory as follows:

```
<install_path>/diamond/1.4/license/license.dat
```

Editing the License File

You must edit a floating license file to specify the server name and the path to the Lattice daemon.

The following example shows part of a floating license file:

```
SERVER nodename 1234abcd 1710
DAEMON lattice daemon_path
FEATURE LSC_DIAMOND_A lattice 10.0 01-jan-9999 100 \
85B686493C86
VENDOR_STRING=LSC_DIAMOND_A
```

Edit the SERVER line by replacing the *nodename* with the host name and the port ID (1710). The port ID, 1710 in this example, must be assigned a TCP/IP port number that is not already in use on the server, so you might need to change it.

Edit the Lattice DAEMON line, replacing *daemon_path* with the path to Lattice Diamond. For lmgrd V11, the path is:

```
<install_path>/diamond/1.4/ispfpga/bin/lin
```

When editing these lines, type them exactly as you received them. All entries are case-sensitive.

Note

The encryption codes are in hexadecimal format (digits 0-9, and lower-case letters a-f or upper-case letters A-F).

Starting the License Manager

Type the following command on one line to start the license manager daemon:

```
<install_path>/diamond/1.4/ispfpga/bin/lin/lmgrd
-l <install_path>/diamond/1.4/license/license.log
-c <install_path>/diamond/1.4/license/license.dat
```

Redirecting output to a log file is helpful when you debug licensing problems. The -l switch tells the license manager to send its output to a log file (license.log), and -c tells it which license to serve. The log file contains information on the status of the server and the daemon and TCP port in use. It also shows which users have checked out the license and the checkout time.

Stopping the License Manager

If it is necessary to stop the FLEXlm license manager, follow this procedure:

1. Confirm that the daemon is running by typing the following command:

```
ps -ef | grep lmgrd
```

2. If lmgrd.exe is running, type the following command on one line to stop the daemon:

```
<install_path>/diamond/1.4/ispfpga/bin/lin/lmutil lmdown -c  
<install_path>/diamond/1.4/license/license.dat
```

The following prompt appears:

```
Shutting down FLEXlm on nodes: hostname  
Are you sure? [y/n]:
```

3. Type **Y** and press **Enter** to shut down the license daemon.

Setting Up a Floating License on Linux

The licensing steps just outlined can be performed on a centralized license server. Then each client points to the license file on that machine. To gain access to the licenses on the remote license server, you need to set the environment variable LM_LICENSE_FILE value to *license_port_number@linux_host_name*.

Running the Lattice Diamond GUI

The Lattice Diamond software has a graphical user interface (GUI). When you run the diamond executable file, the software automatically performs the environment setup.

To start the Lattice Diamond GUI:

1. Set up the license as follows:

If your license file (license.dat) is not under *<install_path>/1.4/license*, you must set the LM_LICENSE_FILE variable to the location of your license file. For example:

For CSH users:

```
setenv LM_LICENSE_FILE (/<license_directory>/license.dat  
$LM_LICENSE_FILE)
```

For BASH users:

```
export LM_LICENSE_FILE=$LM_LICENSE_FILE:/  
<license_directory>/license.dat
```

2. Run the diamond executable file in the command line as follows:

```
<install_path>/diamond/1.4/bin/linux/diamond &
```

With the `diamond` script, you can also run the following tools in stand-alone mode.

To invoke stand-alone Power Calculator, run:

```
diamond -pwc
```

To invoke stand-alone Reveal Analyzer, run:

```
diamond -rva
```

To invoke stand-alone IPexpress, run:

```
diamond -ipx
```

To invoke stand-alone Epic, run:

```
diamond -epic
```

Refer to the online Help for more information about the Diamond GUI.

Using the Examples Directory

In order to use the design examples directory, you must copy the files from the server to your local system and change the write permissions. Copy the files from the server examples directory (`<install_path>/diamond/1.4/examples`) to the equivalent path and directory on your local system.

Finding the Installation History

The Diamond software records a log of your installation history. You can find the history in the Diamond main window.

To view the installation history:

1. Open the Diamond GUI.
2. Choose **Help > About Lattice Diamond**.

Running Lattice Diamond from the Command Line

There are two ways to run from the command line: through Diamond's Tcl Console or by running executable files directly.

Running Standalone Tcl Console

The Lattice Diamond development environment includes Tcl Console, which allows you to run scripts for automating common tasks. Tcl Console is also available outside of the user interface in order to run custom scripts. To launch the stand-alone Tcl Console, enter the following on a command line:

```
<install_path>/diamond/1.4/bin/lin/pnmainc
```

This command configures the environment allowing all of the underlying design tools to be run.

Refer to the online Help for more information about the command line.

Running Using CSH or Bash Interpreters

Before running any other commands, you need to run the following command:

```
cd <install_path>/diamond/1.4/bin/lin source setupenv
```

BASH users must run the following commands:

```
export bindir=<install_path>/diamond/1.4/bin/lin  
source $bindir/diamond_env
```

After that, you can run the executable files directly. For example, you can invoke the Diamond GUI by:

```
pnmain &
```

Or, you can run Power Calculator by running:

```
pwcmmain &
```

Running Synplify Pro for Lattice

To run Synplify Pro for Lattice, do the following:

1. Open the Diamond GUI.
2. Open any project to enable the commands on the Tools menu.

3. Choose **Tools > Synplify Pro for Lattice**.

Running Stand-Alone Reveal Analyzer

After the installation, you can invoke the stand-alone Reveal Analyzer in the command line:

```
% <install_path>/reveal/1.4/bin/linux/revealrva &
```

Running Stand-Alone Programmer

After the installation, you can invoke the stand-alone Programmer in the command line:

```
% <install_path>/programmer/1.4/bin/linux/programmer &
```

Running ispVM System

Before running the ispVM System software, you need to set the environment variables.

Setting Environment Variables (C Shell)

Set the following environment variables in the user profile (.cshrc) in the user's home directory.

- ◆ **ISPVM_DIR** – the directory in which ispVM System is installed.
- ◆ **LD_LIBRARY_PATH** – the directory in which the ispVM System's shared libraries are installed.
- ◆ **path** – to locate ispVM System and other ispVM System tools.

For example, in the .cshrc file, add the following lines:

```
setenv ISPVM_DIR <diamond_install_prefix>/diamond/1.3/  
ispvmsystem  
setenv LD_LIBRARY_PATH $ISPVM_DIR/lib:$LD_LIBRARY_PATH  
set path=($ISPVM_DIR $path)
```

If the **LD_LIBRARY_PATH** and the **path** variables are not already defined, add the following lines instead:

```
setenv ISPVM_DIR <diamond_install_prefix>/diamond/1.3/ispvmsystem
setenv LD_LIBRARY_PATH $ISPVM_DIR/lib
set path=($ISPVM_DIR)
```

Note

When setting an **LD_LIBRARY_PATH**, do not use a tilde (~) anywhere in the designated path.

After the **LD_LIBRARY_PATH** and **path** variables are updated, you are required to login again, or resource the profile before continuing. This will set the environment variables. To resource the user profile, type “source ~/.cshrc” at the prompt. Type **env** and look for the **ISPVM_DIR** and **LD_LIBRARY_PATH** lines to ensure they are pointing to the appropriate directories in which ispVM System was installed.

Setting Environment Variables (BASH)

Set the following environment variables in the user profile (.bashrc) in the user’s home directory.

- ◆ **ISPVM_DIR** – the directory in which ispVM System is installed.
- ◆ **LD_LIBRARY_PATH** – the directory in which the ispVM System’s shared libraries are installed.
- ◆ **path** – to locate ispVM System and other ispVM System tools.

For example, in the .bashrc file, add the following lines:

```
ISPVM_DIR=<diamond_install_prefix>/diamond/1.3/ispvmsystem
export ISPVM_DIR
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ISPVM_DIR/lib
export LD_LIBRARY_PATH
PATH=$PATH:$ISPVM_DIR
export PATH
```

If the **LD_LIBRARY_PATH** and the **path** variables are not already defined, add the following lines instead:

```
ISPVM_DIR=<diamond_install_prefix>/diamond/1.3/ispvmsystem
export ISPVM_DIR
LD_LIBRARY_PATH=$ISPVM_DIR/lib
export LD_LIBRARY_PATH
PATH=$ISPVM_DIR
```

```
export PATH
```

Note

When setting an **LD_LIBRARY_PATH**, do not use a tilde (~) anywhere in the designated path.

After the **LD_LIBRARY_PATH** and **path** variables are updated, you are required to login again, or resource the profile before continuing. This will set the environment variables. To resource the user profile, type “`source ~/.bashrc`” at the prompt. Type **export** and look for the **ISPVM_DIR** and **LD_LIBRARY_PATH** lines to ensure they are pointing to the appropriate directories in which ispVM System was installed.

Invoking ispVM System

To invoke the ispVM System software, do the following:

- ◆ Run the `ispvm` executable file in the command line:

```
ispvm
```

For more information on installing and setting up the ispVM System software, refer to the ispVM System Linux Installation and Setup Manual (`ispVMLinuxInstallation.pdf`) in the ispVM System installation directory.

Running Multiple Versions of Lattice Software

You can run multiple versions of ispLEVER and Lattice Diamond at the same time from either the command line or the graphical user interface. You can run current and previous versions.

Updating Lattice Diamond

After you have registered and licensed your installation, check the Lattice Semiconductor Web site for new software updates, device support, and enhancements. Make sure that you have the latest software by checking for updates regularly.

This process will only find updates to the version you have installed. For upgrades to a new version, check the Lattice Web site.

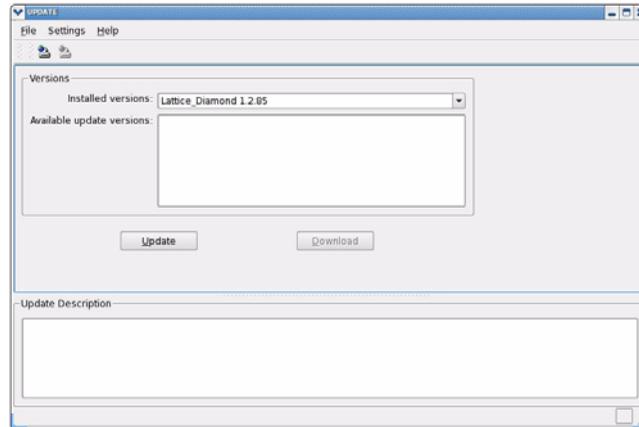
Running the UPDATE Tool

To run the UPDATE tool:

- ◆ Run the update executable file in the command line:

```
update
```

The UPDATE window appears, as shown below.

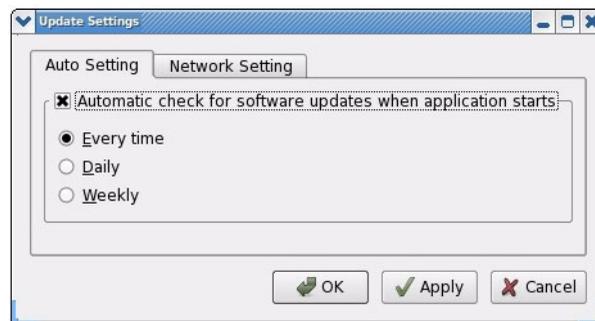


Changing the Automatic Checking Settings

The Diamond software can automatically check for updates and open a window showing the Diamond updates that are available. If you want to change or turn off the automatic checking, change the settings in UPDATE as follows.

To change the automatic checking settings:

1. In the UPDATE window, choose **Settings > Update Settings**.
2. In the Update Settings dialog box, select the **Auto Setting** tab.



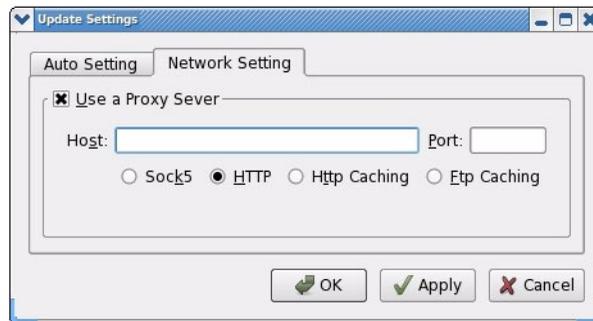
3. In the Auto Setting tab, do either of the following:
 - ◆ To enable automatic checking for software updates, make sure that the **Automatic check for software updates when application starts** option is selected (it is selected by default). Then specify the frequency with which you want the checking to be performed: every time when Diamond is started, daily, or weekly. The default is **Every time**.
 - ◆ To disable the automatic checking, clear the **Automatic check for software updates when application starts** option.
4. Click **OK**.

Changing the Network Setting

To enable online checking for software updates, you must indicate how your computer accesses the Internet.

To change the Internet connection settings:

1. In the UPDATE window, choose **Settings > Update Settings**.
2. In the Update Settings dialog box, select the **Network Setting** tab.



3. In the Network Setting tab, do either of the following:
 - ◆ If you must go through a proxy server before connecting to the Internet, select **Use a Proxy Server** (it is selected by default). The proxy server prevents outsiders from breaking into your organization's private network. Ask your system administrator for the URL address and port assignment.
 - ◆ If you use direct Internet access, clear the **Use a Proxy Server** option.
4. Click **OK**.

Downloading and Installing Software Updates

When you use the automatic checking feature, the Diamond software notifies you whenever an update version becomes available.

You can also manually check for software updates.

To manually check for and download a Diamond update version:

1. In the UPDATE window, click **Update**.

The software goes online to check for updates, and lists the available Diamond update versions (if any) in the Available update versions box.

2. Select the update version that you want to download, and click **Download**.

The software starts downloading the installer of the update version to your computer. You may install it later.

Note

You can also use the **Help > Check for Updates** command in the Diamond window to check for and download software update.

To install a downloaded Diamond update version:

1. Close all Diamond tools.
2. Uncompress the downloaded installer.
3. Go to the location where you put the uncompressed installer.
4. From the command line, execute the RPM installation file.

Note

You need the root privilege to install an RPM package.

Installing LatticeMico Development Tools

This document explains how to install LatticeMico Development Tools on the Linux operating system.

System and Software Requirements

Your system must meet the following minimum system and software requirements to run LatticeMico Development Tools on Linux.

System Requirements

- ◆ Intel Pentium or Pentium-compatible PC with USB port
- ◆ 32-bit Red Hat Enterprise Linux version 4.X or 5.X operating system
- ◆ 2 GB memory recommended for FPGAs
- ◆ Approximately 436 megabytes of free disk space
- ◆ 1024 x 768 graphics display

Software Requirements

- ◆ Perl version 5.8.0 or later. The following Perl modules are required:

- ◆ XML::DOM
- ◆ XML::Parser
- ◆ XML::RegExp
- ◆ Getop::Std
- ◆ Scalar::Util
- ◆ FindBin::Bin

These modules are normally found in the native version of Perl that comes with Red Hat Linux.

- ◆ PDF browser, such as Adobe Acrobat
- ◆ Mozilla or Netscape browser (optional)

Additional Notes for Linux

You may need the following information to run LatticeMico on Linux:

- ◆ If you want to configure your system to use an external browser instead of the Eclipse help browser, you can change this in the LatticeMico main window by choosing the **Window > Preferences** command.
- ◆ Eclipse requires shared libraries from the Mozilla (also known as Seamonkey) Web browser. The Eclipse software expects to find a set of IA32 shared libraries. It may be necessary for you to manually configure the MOZILLA_FIVE_HOME environment variable to point to a valid Mozilla browser library installation.
- ◆ If you cannot access the PDF files referenced in the LatticeMico online Help, you can configure LatticeMico to use an external browser by following these steps:
 - a. In the LatticeMico software, choose **Window > Preferences**.
 - b. Click on **Help**.
 - c. Select **Use external browser**.
 - d. Click on **Web Browser**.
 - e. Select **Use external Web browser**.
 - f. Select **Firefox** or **Netscape**.
 - g. Click **OK**.

Installing LatticeMico Within Diamond Software

To take advantage of the full features and functionality of the LatticeMico Development Tools, Lattice Semiconductor recommends that you install the 1.4 version of Diamond before installing the LatticeMico Development Tools.

This section describes how to install the LatticeMico Development Tools on top of the Diamond software. These procedures assume that you have already installed Diamond 1.4.

Note

You need the root privilege to install an RPM package.

To install LatticeMico Development Tools:

1. Install Lattice Diamond 1.4 first, for example, in `<install_path>/diamond`.
2. Verify the integrity of the RPM file.

```
cd <directory_with_RPM>
rpm -K diamond_1_4-lm-xx-i386-linux.rpm
```

3. Execute the RPM installation file as follows:

To install to the default location (`/usr/local/diamond`), use:

```
sudo rpm -Uvh <path_to_package>/diamond_1_4-lm-xx-i386-
linux.rpm
```

To override the default location use:

```
sudo rpm -Uvh --prefix=<install_path> <path_to_package>/
diamond/1.4/micosystem
```

4. After the installation is completed, verify that the `/micosystem` directory is created in the `<install_path>/diamond/1.4` directory, for example, `/usr/local/diamond/1.4/micosystem`.
5. You can then run LatticeMico System as follows:

```
<install_path>/diamond/1.4/micosystem/LatticeMicoLauncher &
```

Installing LatticeMico as a Standalone Tool

If you do not have the current version of Diamond installed, you can still install the LatticeMico Development Tools, but their functionality will be limited. Mico System Builder (MSB) will not be fully functional. You can create platforms, but the platform generator will not be fully functional because of missing Diamond executables and functions. The design-rule checker will also not be fully functional. However, the graphical user interfaces of the C/C++ Software Project Environment (C/C++ SPE) and the debug environment will be fully functional.

Note

You need the root privilege to install an RPM package.

To install LatticeMico Development Tools:

1. Verify the integrity of the RPM file.

```
cd <directory_with_RPM>
```

2. Execute the RPM installation file as follows:

To install to the default location, use:

```
sudo rpm -Uvh <path_to_package>/diamond_1_4-lm-xx-i386-  
linux.rpm
```

To override the default location use:

```
sudo rpm -Uvh --prefix=<install_path> <path_to_package>/  
diamond/1.4/micosystem
```

3. After the installation is completed, verify that the /micosystem directory is created in the <install_path>/latticemicosystem/1.4 directory, for example, /usr/local/latticemicosystem/1.4/micosystem.
4. You can then run LatticeMico System as follows:

```
<install_path>/latticemicosystem/1.4/LatticeMicoLauncher &
```