

Efinity[®] Software Installation User Guide

UG-EFN-INSTALL-v3.8 May 2025 www.efinixinc.com



Copyright © 2025. All rights reserved. Efinix, the Efinix logo, the Titanium logo, the Topaz logo, Quantum, Trion, and Efinity are trademarks of Efinix, Inc. All other trademarks and service marks are the property of their respective owners. All specifications subject to change without notice.

Contents

Overview	3
Hardware and Software Requirements	3
Third-Party Simulator Support	6
Installing the Efinity Software	6
Installing Patches	6
Setting User and Project Directories	7
Proxy Settings	8
Efinity Quick Start	9
Appendix: Installing USB Drivers Installing the Linux USB Driver Installing the Windows USB Driver	10
Where to Learn More	
Troubleshooting	13
Revision History	14

Overview

The Efinity[®] software provides a complete tool flow for designing with Efinix[®] products. This document describes how to install the software.

Hardware and Software Requirements

General Requirements

- Efinity full release: 64-bit operating system, at least dual-core
- Your preferred text editor such as Notepad, gVim, Visual Studio
- Machine memory requirements (when compiling Efinity designs):

Table 1: Machine Memory Requirements

These requirements assume up to 16 threads, and include 4 GB for the operating system and background applications.

Product	Model	Memory
Trion	T4, T8, T13, T20, T35	8 GB
	T55, T85, T120	12 GB
Titanium	Ti35, Ti60, Ti85, Ti90, Ti120, Ti135, Ti180	8 GB
	Ti165, Ti240, Ti375	12 GB
Topaz	Tz50, Tz75, Tz100, Tz110, Tz170	8 GB
	Tz200, Tz325	12 GB

Windows Requirements

i

- Efinity full release or Windows Standalone Programmer: Windows 10 or later, 64-bit operating system
- Microsoft Visual C+ + 2022 x64 runtime library (or latest version) redistributable https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist? view=msvc-170
- Zadig software to install USB drivers see Installing the Windows USB Driver on page 11
- Java 64-bit runtime environment; required for configuring some IP cores in the IP Manager (e.g., Sapphire SoC); available from:
 - https://www.java.com/en/download/manual.jsp (Java 8)
 - https://developers.redhat.com/products/openjdk/download (OpenJDK 8 or 11)
 - http://jdk.java.net/16/ (OpenJDK 16)

Note: You may also use other Java software platforms that are available in the market.



Note: The path <*drive*>:\Windows\System32 must exist in %PATH% if you have a customized environment variable.

Linux Requirements

Supported operating systems:

- Ubuntu v20.04 or later
- Red Hat Enterprise v8.8 or later

Additional software you need to install:

- Libraries:
 - Ubuntu v20.04⁽¹⁾—apt install libxcb-cursor0
 - Red Hat-yum install xcb-util-cursor
- Linux X11 or Wayland windowing system (for Efinity[®] GUI)
- Java 64-bit runtime environment (8 or higher), required for configuring some IP cores in the IP Manager (e.g., Sapphire SoC). Follow the instructions on the Ubuntu web site or Red Hat web site to install it. Your path environment variable should include the Java executable.
- Udev device manager for Efinix USB programming cable see Installing the Linux USB Driver on page 10

Running the Efinity software on Ubuntu in a virtual machine or using Windows WSL requires these additional libraries:

Platform	Ubuntu Version	Required Libraries
VM	20.04	sudo apt install libxcb-cursor0 libnss3 libasound2 libxkbfile1 -y
	22.04	No additional libraries required. ⁽¹⁾
	24.04	
WSL v2.3,	20.04	sudo apt update
v2.4 ⁽²⁾		sudo apt install libxcb-cursor0 libnss3 libasound2 libxkbfile1
		sudo apt install libxcb-xinerama0 libxcb-icccm4 libxcb-image0 libxcb- keysyms1 libxcb-render-util0 libxcb-shape0 libxkbcommon-x11-0 libegl1 libxdamage1 ⁽³⁾
	22.04	sudo apt update apt install libxcb-cursor0 libnss3 libasound2 libxkbfile1
	24.04	sudo apt update sudo apt install libxcb-cursor0 libnss3 libasound2t64 libxkbfile1 sudo apt install libxcb-xinerama0 libxcb-icccm4 libxcb-image0 libxcb- keysyms1 libxcb-render-util0 libxcb-shape0 libxkbcommon-x11-0 ⁽³⁾

Table 2: Using Efinity and Ubuntu in VM or WSL

Note: Efinix recommends increasing the memory reservation for your WSL2 machine to avoid degraded performance or out of memory situations. Refer to https://learn.microsoft.com/en-us/windows/wsl/wsl-config#wslconfig.

⁽¹⁾ The official LTS images for v22.04 and v24.04 include the **libxcb-cursor0** library by default.

⁽²⁾ For the Bitstream Security Key Generator and JTAG SVF Player, you need to set an environment variable. See **Table 4:** Efinity GUI Does Not Open on page 13.

⁽³⁾ You can also use the the command sudo apt install qtwayland5, however, it installs more libraries than you need, which may not be desired.

Installing iVerilog

Icarus Verilog (iVerilog) is a free Verilog simulation tool you can use to compile and simulate Verilog HDL source code. The software is available as source code or as pre-compiled binaries.

Windows installation:

To download the simulator: bleyer.org/icarus

 (\mathbf{i})

i

i.

Note: The latest versions of iVerilog are bundled with the GTKWave software, so you only need to download 1 file to get both tools. Refer to the **bleyer.org/icarus** website for more information.

To download the simulator source code: github.com/steveicarus/iverilog

Linux installation:

Refer to the Installation Guide for steps to obtain, compile and install Icarus Verilog: steveicarus.github.io/iverilog/

Note: Efinix recommends iVerilog version 11.0 or later.

Installing GTKWave

GTKWave is an open-source tool that analyzes post-simulation dumpfiles and displays the results in a graphical interface. It includes a waveform viewer and RTL source code navigator. You can use GTKWave with the iVerilog simulator to analyze and debug your simulation model, or to view any VCD waveform.

Windows installation:

You can read more at gtkwave.sourceforge.net.

Note: If you have downloaded and installed the iverilog setup file (bundled with GTKWave), you do not need to install a separate standalone GTKWave.

To download and run the latest Windows version, follow these steps:

- 1. You can browse for the software files at gtkwave Browse Files at Sourceforge.net. The Windows files are situated lower down the page.
- 2. Unzip the downloaded file.
- **3.** Optional:

You may need to add the path to GTKWave (\$GTKWave_folder\$\bin\) to your System Variables path for the software to launch correctly.

4. Run the program by executing **gtkwave.exe** in the <install dir>/bin directory.

Linux installation:

Linux users can use the following commands:

sudo apt-get update sudo apt-get install gtkwave

Third-Party Simulator Support

The Efinity tools do not include or explicitly integrate third-party simulators. However, Efinix has verified that the following simulators work with Efinity-generated Verilog HDL netlist files:

- Cadence Xcelium Logic Simulator
- Mentor Graphics QuestaSim Simulator
- Aldec Active HDL and Riviera-PRO Simulators
- Free Icarus Verilog (iVerilog) Simulator

To simulate an Efinity post-synthesis (or later compiler stage) Verilog HDL netlist, include the following library path as a resource in your third-party simulator:

<Efinity top-level path>/sim_models/verilog

Installing the Efinity Software

Windows installation:

Double-click the **efinity**-<*version*>.msi installer and follow the on-screen instructions.

Optional:

Run the following script to install a shortcut in your Desktop directory:

> <installation directory>/bin/install desktop.sh

Linux installation:

Unzip or untar the Efinity package into your user directory:

> tar -xjvf efinity-<version>.tar.bz2

Installing Patches

You download Efinity[®] patches separately from the software and then install them into your existing Efinity[®] installation directory.

Windows

- 1. Download the patch from the Efinity[®] page in the Support Center.
- 2. Unzip the patch into any temporary directory by double-clicking the patch filename in the Windows Explorer and choosing Extract all or by using the command unzip efinity-<version>-patch.zip at a command prompt.
- 3. Setup the environment variables by typing these commands at a command prompt:

> <path to Efinity>\<version>\bin\setup.bat

4. Run the patch installer by typing these commands at a command prompt:

```
> cd efinity-<version>-patch
```

> run.bat



Note: The path <*drive*>:**Windows****System32** must exist in %PATH% if you have a customized environment variable.

Linux

- 1. Download the patch from the Efinity[®] page in the Support Center.
- 2. Open a terminal window.
- 3. Unzip the patch into any temporary directory:

```
> unzip efinity-<version>-patch.zip
```

4. Setup the environment variables:

```
> source /path/to/efinity/<version>/bin/setup.sh
```

5. Run the patch installer:

```
> cd efinity-<version>-patch
> ./run.sh
```

Setting User and Project Directories

Historically, the Efinity software saved all files, including project files, in the Efinity installation directory by default. Beginning with the Efinity software v2025.1, the default location is in your user directory:

- Linux—/home/
 user name>/.efinity
- Windows—C:\Users\<user name>\.efinity

This change makes it easier to manage projects if you have multiple versions of Efinity software installed. The **.efinity** directory is your default project directory and also stores log files and **.ini** files for various tools (IP Manager, IP Packager, Interface Designer, Debugger, Programmer, etc.).

To change the default location, choose **File > Preferences**. In the **Preferences** dialog box, change the **Top-level project path** and **User directory** fields for the new path. The Efinity software prompts you to restart the software after you change the path(s).

Proxy Settings

Depending upon your circumstances, you may need to modify your network proxy settings to enable the Efinity software to run correctly.

Important: The default network proxy settings work for most users. You should only modify your network proxy settings if your existing network settings block access to the ports used by the Efinity software, which causes the Efinity software to fail to start.

Open the network proxy settings for your operating system, then copy and paste the following into the **Ignore Hosts** box:

```
localhost, 127.0.0.0/8, ::1, 127.0.0.1
```

Figure 1: Proxy Settings Example

	Network Proxy			×
AutomaticManualDisabled				
HTTP Proxy	www.proxy.example.com	8080	-	+
HTTPS Proxy		0		+
FTP Proxy		0	-	+
Socks Host		0	-	+
Ignore Hosts	localhost, 127.0.0.0/8, ::1	, 127.0.0.	1	

Efinity Quick Start

To launch the Efinity graphical user interface (GUI), double-click the Efinity desktop icon. To launch and use the Efinity tool from the command line, refer to the following sections.



Warning: Do not use non-English characters in the Efinity project paths.

Windows

Set up your environment and PATH:

bin\setup.bat

Launch the Efinity GUI from the command line:

bin\setup.bat --run

Run Efinity from the command line:

cd %EFINITY_HOME%\project \<project name> // Cl
efx_run.bat <project name>.xml // Ru
// Ru

// Change to project directory
// Run Efinity

For command-line help:

efx_run.bat --help

Linux

Set up your environment and PATH:

source bin/setup.sh

Launch the Efinity GUI from the command line:

efinity

Run Efinity from the command line:

cd \$EFINITY HOME/project/<project name> // Change to project directory efx_run.py <project name>.xml // Run Efinity

For command-line help:

efx_run.py --help

Appendix: Installing USB Drivers

To program Trion[®], Topaz, and Titanium FPGAs using the Efinity[®] software and programming cables, you need to install drivers.

Efinix development boards have FTDI chips (FT232H, FT2232H, or FT4232H) to communicate with the USB port and other interfaces such as SPI, JTAG, or UART. Refer to the Efinix development kit user guide for details on installing drivers for the development board.



Note: If you are using more than one Efinix development board, you must manage drivers accordingly. Refer to AN 050: Managing Windows Drivers for more information.

Notice: The Trion T8 BGA81 Development Boards do not have FTDI chip for USB communication. Refer to the T8 BGA81 Development Kit User Guide for more information about installing its Windows USB driver.

For your own development board, Efinix suggests using the FTDI Chip FT2232H or FT4232H Mini Modules for JTAG programming Trion[®], Topaz, and Titanium FPGAs. (You can use any JTAG cable for JTAG functions other than programming.)



Note: Efinix does not recommend the FTDI Chip C232HM-DDHSL-0 programming cable due to the possibility of the FPGA not being recognized or the potential for programming failures.

Board	Connect to Computer with
Efinix development boards	USB cable
Your own board	FTDI x232H programming kit. For example:
	• FT2232H Mini Module
	• FT4232H Mini Module

Table 3: USB Programming Connections



Note: The FTDI Chip Mini Module supports 3.3 V I/O voltage only. Refer to the FTDI Chip website for more information about the modules.

Installing the Linux USB Driver

The following instructions explain how to install a USB driver for Linux operating systems.

- 1. Disconnect your board from your computer.
- 2. In a terminal, use these commands:

```
> sudo <installation directory>/bin/install_usb_driver.sh
> sudo udevadm control --reload-rules
```

```
> sudo udevadm trigger
```



Note: If your board was connected to your computer before you executed these commands, you need to disconnect it, then re-connect it.

Installing the Windows USB Driver

On Windows, you use software from Zadig to install drivers. Download the Zadig software (version 2.7 or later) from zadig.akeo.ie. (You do not need to install it; simply run the downloaded executable.)

Important: For some Efinix development boards, Windows automatically installs drivers for some interfaces when you connect the board to your computer. You do not need to install another driver for these interfaces. Refer to the user guide for your development board for specific driver installation requirements.

To install the driver:

- 1. Connect the board to your computer with the appropriate cable and power it up.
- 2. Run the Zadig software.



Note: To ensure that the USB driver is persistent across user sessions, run the Zadig software as administrator.

- 3. Choose Options > List All Devices.
- 4. Repeat the following steps for each interface. The interface names end with *(Interface N)*, where *N* is the channel number.
 - Select libusb-win32 in the Driver drop-down list.
 - Click **Replace Driver**.
- 5. Close the Zadig software.

Note: This section describes how to install the libusb-win32 driver for each interface separately. If you have previously installed a composite driver or installed using libusbK drivers, you do not need to update or reinstall the driver. They should continue to work correctly.

Where to Learn More

i

The Efinity[®] software includes documentation as PDF user guides and on-line HTML help. This documentation is provided with the software. You can also access the latest versions of PDF documentation in the Support Center:

- Efinity Software User Guide
- Efinity Synthesis User Guide
- Efinity Timing Closure User Guide
- Efinity Software Installation User Guide
- Efinity Trion Tutorial
- Efinity Debugger Tutorial
- Topaz Interfaces User Guide
- Titanium Interfaces User Guide
- Trion Interfaces User Guide
- Efinity Interface Designer Python API
- Quantum[®] Trion Primitives User Guide
- Quantum[®] Titanium Primitives User Guide
- Quantum[®] Topaz Primitives User Guide

In addition to documentation, Efinix field application engineers have created a series of videos to help you learn about aspects of the software. You can view these videos in the Support Center.

Troubleshooting

Although it is rare, sometimes you may encounter issues running the Efinity software. This topic describes some issues and how to debug the problem.

Table 4: Efinity GUI Does Not Open

Platform	Solution
Windows	The Efinity software GUI uses QT, which has a known bug for non-English language fonts. If a font with a name containing trailing spaces (e.g., "SDC-Sadeh Fat " or "BC C39 2 to 1 HD ") is installed on your computer, the Efinity software will not open. Uninstall the font to fix this problem. See related forum topic.
	Your proxy settings may cause the software to fail to run. See Proxy Settings on page 8.
Linux	You can have this problem if your system has /usr/lib64/libffi.so.8 installed. Instead, use /usr/lib64/libffi.so.6 . See related forum topic.
	Your proxy settings may cause the software to fail to run. See Proxy Settings on page 8.
	If you are using the Wayland display server and have problems with the Efinity GUI, set this environment variable to help troubleshoot: export QT_QPA_PLATFORM=xcb.

Table 5: Efinity Crashes During Synthesis

Platform	Solution
Windows	Observed in 2024.2 (See related forum topic.)
	If you have the Sophos antivirus software on your computer, it may prevent Efinity synthesis from running. To solve this issue, disable the antivirus software, add Efinity to the Sophos exclusion list, or set the environment variable TCMALLOC_DISABLE_REPLACEMENT to 1 before starting the Efinity GUI or the flow from the command line. (Setting this environment variable comes at a cost of reducing the performance advantages in 2024.2 vs. 2024.1). The command-line commands are:
	<efinity path="">\2024.2\bin\setup.bat set TCMALLOC_DISABLE_REPLACEMENT=1 efinity</efinity>
	If the TCMALLOC_DISABLE_REPLACEMENT setting does not fix the issue, try running the software in Windows 8 compatibility mode. This mode increases the compile time.

Table 6: Cannot Connect to Board after Installing the USB Driver

Platform	Solution
Linux	If, after installing the USB driver, the USB permissions appear to be wrong or do not work, verify that the file 80-efx-pgm.rules exists in the /etc/udev/rules.d/ directory. If the file is not there, try re-running the install_usb_driver.sh command, or, if required, manually copy the <i><efinity directory<="" i="">/bin/80-efx-pgm.rules file to the /etc/udev/rules.d/ directory.</efinity></i>

Table 7:	IP Manager	RPC Server	Not Connected
----------	------------	-------------------	---------------

Platform	Solution
Windows	The IP Manager creates and uses server instances for inter-process communication. If no ports are available for use, it shows the message "IP Manager RPC Server Not Connected" in the Console. Refer to "Resolving IP Manager Issues" in the Efinity Software User Guide for details on how to resolve the problem.

Turn on Debug Messages

To help diagnose an issue, turn on Efinity debug messages and send them to Efinix support in the forum.

- 1. Open the file *<path>/efinity/<version>/bin/lc.ini*.
- 2. In the *Disable debug logging in all categories* section, change all of the settings from false to true.
- **3.** Save.
- 4. Run the Efinity software. Debug messages are saved to log files (efinity.log and efinity_console.log). The files are saved in:
 - Linux: /home/<username>/.local/share/efinity/log
 - Windows: C:\Users\<username>\AppData\Local\efinity\log
- 5. Send the messages to Efinix support in the forum https://forum.efinix.net.

Note: The **AppData** directory is a hidden item. To view it, you need to select the **Show hidden files**, folders, and drives option (Folder Options > View tab > Hidden files and folders).

Revision History

i

Date	Version	Description
May 2025	3.8	Updated machine memory requirements. (DOC-2286)
		Updated Hardware and Software Requirements. (DOC-2343)
		Updated the troubleshooting topic for the Sophos anti-virus software. (DOC-2409)
		Aldec Active HDL and Riviera-PRO simulators now supported. (DOC-2463)
		Added topic on Setting User and Project Directories on page 7.
January 2025	3.7	Updated instructions for installing Linux USB drivers. (DOC-2279)
December 2024	3.6	Added toubleshooting tip for users who have the Sophos antivirus software installed (Windows). (DOC-2267)
November 2024	3.5	Updated memory requirements in Table 1: Machine Memory Requirements on page 3. (DOC-2052)
		Added new topic: Proxy Settings on page 8. (DOC-2037)
		Corrected link to latest Microsoft Visual C++ Redistributable downloads. (DOC-2045)
		Linux users should ensure that the JRE is installed. (DOC-2056)
August 2024	3.4	Added Troubleshooting topic. (DOC-1956)
June 2024	3.3	Red Hat support is v8.0 and higher. Removed support for v7.4. (DOC-1648)
May 2024	3.2	Added Ti165 and Ti240 FPGAs, replacing the Ti135 and Ti200, respectively.

Table 8: Document Revision History

Date	Version	Description
January 2024	3.1	Added note about Windows %PATH% variable when installing patches. (DOC-1687)
December 2023	3.0	Updated machine memory requirements.
		For Windows, a 64-bit operating system is required. 32-bit systems are not supported.
March 2023	2.9	Updated x86 architecture info under General topic of Hardware and Software Requirements on page 3. (DOC-1102)
		Added note for Java in Window requirement.
		Updated table Linux Operating System.
		Updated information for Installing iVerilog.
		Sourceforge.net in Installing GTKWave topic. (DOC-1122)
August 2022	2.8	Updated Efinity 2022.1 platform in Hardware and Software Requirements on page 3.
		Updated Installing USB Drivers topics.
June 2022	2.7	Pointed to new sourceforge location for GTKWave download. (DOC-797)
December 2021	2.6	Updated machine memory requirements (RAM).
October 2021	2.5	When using the stand-alone Programmer on 64-bit Windows, install both the x86 and x64 libraries. (DOC-576)
September 2021	2.4	JRE required for running the DMA Controller in the IP Manager. (DOC-549)
June 2021	2.3	Supported Ubuntu version is v18.04 or higher. v16.04 is end of life. (DOC-433)
		Added the Java runtime environment as a software requirement for configuring the Sapphire SoC in the IP Manager.
		Updated the Windows USB driver installation topic.
December 2020	2.2	Added the requirement to install the Microsoft Visual C++ 2015 x64 and x86 runtime libraries for the standalone Programmer.
November 2020	2.1	Updated instructions on installing USB drivers for Windows.
June 2020	2.0	Added instructions on how to install software patches.
		Windows 7, Red Hat v6, and CentOS v6 no longer supported. Provided new driver when installing USB drivers on Windows with Zadig software.
		Added FTDI Dual RS232 HS mini module in steps to install the USB driver.
December 2019	1.7	Updated Zadig USB driver information for Windows.
August 2019	1.6	Updated Quick Start command-line instructions.
January 2019	1.5	Added instructions on installing the USB driver for Windows.
October 2018	1.4	Added Python 3 to the software requirements list. For Windows, if you do not have a full version of Python, the .py extension may not be correctly associated with Python.
June 2018	1.3	Removed Python requirement; as of this release, Python is included with the software.
		Added the requirement that Windows users install the Microsoft Visual C++ 2015 x64 runtime library.
April 2018	1.2	No changes.
November 2017	1.1	Removed references to OPM family.
		Removed instructions for setting external code editor (this version embeds a Code Editor).

Date	Version	Description
May 2017	1.0	Initial release.