



# Efinity<sup>®</sup> Software Installation User Guide

---

UG-EFN-INSTALL-v3.5  
November 2024  
[www.efinixinc.com](http://www.efinixinc.com)



# Contents

- Overview.....3**
- Hardware and Software Requirements..... 3**
- Third-Party Simulator Support.....5**
- Installing the Efinity Software..... 6**
- Installing Patches..... 6**
- Proxy Settings..... 7**
- Efinity Quick Start..... 8**
- Appendix: Installing USB Drivers..... 9**
  - Installing the Linux USB Driver..... 9
  - Installing the Windows USB Driver..... 10
- Where to Learn More..... 10**
- Troubleshooting..... 11**
- Revision History..... 11**

# 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, x86 instruction set architecture
- Windows Standalone Programmer.
  - Windows 10: 64-bit x86 instruction set architecture.
  - Windows 11: 64-bit x86 instruction set architecture.
- Computer with a 64-bit operating system.
  - A 64-bit Windows system is required for the Efinity standalone programmer.
  - A 64-bit Windows system is required for using the security tools in the Efinity standalone programmer.
- Your preferred text editor such as Notepad, gVim, Visual Studio
- Machine memory requirements (when running Efinity design compilations):

*Table 1: Machine Memory Requirements*

Product	Model	Memory
Trion	T4, T8, T13, T20, T35	8 GB
	T55, T85, T120	16 GB
Titanium	Ti35, Ti60	8 GB
	Ti85, Ti90, Ti120, Ti135, Ti180, Ti165, Ti240, Ti375	16 GB
Topaz	Tz50	8 GB
	Tz110, Tz170	16 GB

### Windows Requirements

- Windows 10 or later, 64 bit operating system
- Microsoft Visual C++ 2019 x64 runtime library (or latest version) redistributable  
<https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170>



**Note:** If you want to use the stand-alone Programmer, you also need to install the x86 and x64 libraries (for 64-bit systems).

- Zadig software to install USB drivers  
see [Installing the Windows USB Driver](#) on page 10
- Java 64-bit runtime environment; required for configuring the Sapphire RISC-V SoC and DMA Controller in the IP Manager; 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

Table 2: Linux Operating System

Operating System	Note
Ubuntu v18.04 or later	<p>You may need to install some graphics related libraries before running the Efinity tools. An example list of libraries (not comprehensive) could be:  <b>apt install libxcb-xinerama0 libxcb-iccmm4 libxcb-image0 libxcb-keysyms1 libxcb-render-util0</b></p> <p>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 <a href="#">Ubuntu web site</a> to install it. Your path environment variable should include the Java executable.</p>
Red Hat Enterprise v8.0 or later	<p>You may need to install some graphics related libraries before running the Efinity tools. An example list of libraries (not comprehensive) could be:  <b>yum install libxkbcommon-x11 xcb-util-renderutil xcb-util-keysyms xcb-util-image xcb-util-wm xcb-util</b></p> <p> <b>Important:</b> If the GUI is not working, set an environment variable, then launch the tool again using the following commands (applicable for all Linux operating systems):</p> <pre>source/path/to/efinity/&lt;version&gt;/bin/setup.sh export QT_DEBUG_PLUGINS=1 efinity</pre> <p>The Qt verbose command line output can provide clues about which system libraries are missing and need to be installed via apt/yum.</p> <p>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 <a href="#">Red Hat web site</a> to install it. Your path environment variable should include the Java executable.</p>

- Linux X11 windowing system (for Efinity® GUI)
- Udev device manager for Efnix USB programming cable  
see [Installing the Linux USB Driver](#) on page 9

## 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](http://bleyer.org/icarus)



**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](http://bleyer.org/icarus) website for more information.

To download the simulator source code: [github.com/steveicarus/iverilog](https://github.com/steveicarus/iverilog)

### Linux installation:

Refer to the Installation Guide for steps to obtain, compile and install Icarus Verilog:  
[steveicarus.github.io/iverilog/](https://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](https://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 gtwave
```

## 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
- 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
```

# 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*

Proxy Type	Host	Port
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> // Change to project directory
efx_run.bat <project name>.xml // 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<sup>®</sup>, Topaz, and Titanium FPGAs using the Efinity<sup>®</sup> software and programming cables, you need to install drivers.

Efinity 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 Efinity development kit user guide for details on installing drivers for the development board.



**Note:** If you are using more than one Efinity development board, you must manage drivers accordingly. Refer to [AN 050: Managing Windows Drivers](#) for more information.



**Note:** 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, Efinity suggests using the FTDI Chip FT2232H or FT4232H Mini Modules for JTAG programming Trion<sup>®</sup>, Topaz, and Titanium FPGAs. (You can use any JTAG cable for JTAG functions other than programming.)



**Note:** Efinity 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.

*Table 3: USB Programming Connections*

Board	Connect to Computer with
Efinity development boards	USB cable
Your own board	FTDI x232H programming kit. For example: <ul style="list-style-type: none"> <li>• FT2232H Mini Module</li> <li>• FT4232H Mini Module</li> </ul>



**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
```



**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](http://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

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](#)
- [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 common 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. <a href="#">See related forum topic.</a>
	Your proxy settings may cause the software to fail to run. See <a href="#">Proxy Settings</a> on page 7.
Linux	You can have this problem if your system has <code>/usr/lib64/libffi.so.8</code> installed. Instead, use <code>/usr/lib64/libffi.so.6</code> . <a href="#">See related forum topic.</a>
	Your proxy settings may cause the software to fail to run. See <a href="#">Proxy Settings</a> on page 7.

## 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

**Table 5: Document Revision History**

Date	Version	Description
November 2024	3.5	Updated memory requirements in <a href="#">Table 1: Machine Memory Requirements</a> on page 3. (DOC-2052) Added new topic: <a href="#">Proxy Settings</a> on page 7. (DOC-2037) Corrected link to latest Microsoft Visual C++ Redistributable downloads. (DOC-2045) Linux users should ensure that the JRE is installed. (DOC-2056)

Date	Version	Description
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.
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 <b>Hardware and Software Requirements</b> on page 3. (DOC-1102) Added note for Java in Window requirement. Updated table Linux Operating System. Updated information for Installing iVerilog. Updated steps regarding browsing and downloading Windows file from Sourceforge.net in Installing GTKWave topic. (DOC-1122)
August 2022	2.8	Updated Efinity 2022.1 platform in <b>Hardware and Software Requirements</b> 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.

<b>Date</b>	<b>Version</b>	<b>Description</b>
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).
May 2017	1.0	Initial release.