CrossCore® Development Tools by Analog Devices

Installation Guide for CrossCore Embedded Studio

Installation Guide for CrossCore Embedded Studio
June 2024

http://www.analog.com

1 Table of Contents

2 Introduction

This document describes the installation procedures for CrossCore Embedded Studio under both Windows and Linux as well as minimum hardware requirements and supported versions of each operating system.

3 System Hardware Requirements

Verify that your PC has these minimum requirements for the CCES installation:

- 2 GHz single core processor; 3.3GHz dual core or better recommended
- 4 GB RAM; 8GB or more recommended
- 4 GB available disk space

> **Note**
>
> A faster disk drive or SSD decreases the build time, especially for a large amount of source files. 8GB of RAM or more will substantially increase the performance of the IDE.

4 Supported Operating Systems

Notes for Windows Users

The following versions of Windows are supported for this release of CCES:

- Windows 10 Pro or Enterprise (32-bit and 64-bit)
- Windows 11 Pro or Enterprise (64-bit)

Notes for Linux Users

The following versions of Windows are supported for this release of CCES:

- Ubuntu 20.04 32-bit and 64-bit
- Ubuntu 22.04 32-bit and 64-bit

5 Installing CrossCore Embedded Studio on Windows

Caution

- Windows users may experience User Access Control (UAC) related errors if the software is installed into a protected location, such as `Program Files` or `Program Files (x86)`. We recommend installing the software in a non-UAC-protected location.
- *Prior to installation:* Ensure your machine is up-to-date with relevant Windows updates from Microsoft. CrossCore Embedded Studio relies upon the Microsoft Universal C Runtime from VisualStudio 2015, and this can silently fail to install if your machine is out of date. For more details, refer to Update for Universal C Runtime in Windows on Microsoft's web site.

To install CrossCore Embedded Studio on Windows run the installer by double-clicking `ADI_CrossCoreEmbeddedStudio-Rel2.12.1.exe`.

To uninstall CrossCore Embedded Studio, open Control Panel / Programs and Features applet and select to uninstall CrossCore Embedded Studio 2.12.1. You may need to delete the installation directory to clean up any leftover files.

6 Installing CrossCore Embedded Studio on Linux

The following features are available and supported by this installation:

- Compilation using the GNU toolchain for the ARM Cortex-A core on ADSP-SC57x, ADSP-SC58x and ADSP-SC59x processors.
- Compilation using the GNU ARM toolchain for the ADuCM36x, ADuCM302x and ADuCM4x50 ARM Cortex-M cores.
- Debugging ADSP-SC5xx, ADuCM360, ADuCM302x and ADuCM4x50 via the IDE with GDB/OpenOCD.
- Development and debugging of Applications running under Linux on the ARM Cortex-A core on ADSP-SC57x, ADSP-SC58x and ADSP-SC59x processors.
- Development and debugging of bare-metal applications on the ADuCM360, ADuCM302x and ADuCM4x50 ARM Cortex-M cores.

The following features are available after installing *SHARC and Blackfin Linux Command-Line Tools* though *Help>Install New Software...*:

- Building and command-line execution via the functional simulator (but not cycle-accurate simulation, or debugging within the IDE) of Blackfin+ processors.
- Building and command-line execution via the functional simulator (but not cycle-accurate simulation, or debugging within the IDE) of SHARC+ processors.

The following features are *only* supported under Windows:

- Development, simulation and debug of Blackfin processors
- Development, simulation and debug of SHARC processors (excluding the ARM Cortex-A core on ADSP-SC57x, ADSP-SC58x and ADSP-SC59x processors)
- Use of CrossCore Embedded Studio Add-Ins
- Debugging an Application using the native CrossCore Debugger

6.1 32-bit Compatibility Libraries

To install and use CCES on Ubuntu 64-bit platforms, the following 32-bit compatibility libraries need to be installed:

```
sudo dpkg --add-architecture i386 && sudo apt-get update && sudo apt-get install -y \
  libc6:i386 libncurses5:i386 libstdc++6:i386 libgtk2.0-0:i386 libxtst6:i386 \
  gtk2-engines-murrine:i386 libcanberra-gtk-module:i386 gtk2-engines:i386
```

6.2 Installing the CrossCore Embedded Studio Package

> **Caution**
>
> It is strongly recommended to use the command prompt to install CrossCore Embedded Studio. The installation may not work properly when using Ubuntu Software and/or Ubuntu Software Center.

To install CrossCore Embedded Studio run the following command from the command prompt (substituting `x.y.z` for the version of CCES that you have downloaded):

```
sudo apt-get install ./adi-CrossCoreEmbeddedStudio-linux-x86-x.y.z.deb
```

6.3 Uninstalling CrossCore Embedded Studio

To uninstall CrossCore Embedded Studio run the following commands from the command prompt (substituting `x.y.z` for the version of CCES that you have downloaded):

```
sudo apt-get purge adi-cces-x.y.z
sudo rm -rf /opt/analog/cces/x.y.z  # to clean up any leftover files
```

6.4 Different users sharing the same CCES license on Linux

Many users can share a single valid `license.dat` file on a system by creating a symbolic link to the valid license.dat in their own home directory (`~/.analog/cces`).
The user who installed license should ensure that the appropriate directory and file permissions are set-up to allow other users to access the valid license.dat.

6.5 OpenOCD needs to be run as sudo on Linux

In order to debug an Application with GDB and OpenOCD (Emulator) on Linux, OpenOCD needs to have permissions to access your USB device. You can set-up the necessary permissions when installing CCES on Linux by selecting 'Configure OpenOCD permissions' option on the installation dialog or afterwards by running `sudo sh /opt/analog/cces/{version}/Setup/setup_openocd_permissions.sh`.

If you debug an Application with GDB and OpenOCD (Emulator) using the IDE and OpenOCD fails because it cannot access your USB device, a dialog will appear with a message telling you that you can run the `setup_openocd_permissions.sh` script.

If you start CCES with sudo permission, then there should be no problems with OpenOCD accessing your USB device.

7 Installing SHARC+ and Blackfin+ Linux Command-Line Tools

The SHARC+ and Blackfin+ toolchains are available as an optional extra installation, through *Help>Install New Software...*

Please note:

- These toolchains are made available in order to support Linux-hosted Continuous Integration/Continuous Deployment environments.
- These toolchains do *not* contain support for cycle-accurate simulation, or debugging SHARC or Blackfin applications within the IDE on Linux hosts.
- The appropriate Linux-hosted version of CrossCore Embedded Studio must be installed first, then the *SHARC and Blackfin Linux Command-line Tools* can be installed.
- There is typically a delay of 1–2 weeks between the release of the Windows-hosted CCES product and the corresponding *SHARC and Blackfin Linux Command-line Tools*.

There are two ways to install the SHARC and Blackfin toolchains, described below.

7.1 Supported Parts

The supported processors are ADSP-BF707, and all ADSP-2156x, ADSP-2157x, ADSP-2158x, ADSP-2159x, ADSP-SC57x, ADSP-SC58x parts and ADSP-SC59x.

Earlier SHARC and Blackfin processors are not supported.

7.2 Installing Using the CrossCore Embedded Studio IDE

In these steps, replace "x.y.z" with the appropriate version number for your CrossCore Embedded Studio installation.

1. If you are using docker, make sure you create a container that exports its display to your local machine.
2. Open the IDE by executing the following command:
   `/opt/analog/cces/x.y.z/Eclipse/cces &`
3. Go to *Help > Install New Software...*
4. Click the down arrow to the right of the *Work with:* entry box
5. Choose *CrossCore Embedded Studio Software and Documentation - http://www.analog.com/static/ccesupdatesite*
6. From the resulting list of Available Software, expand the *CrossCore Toolchain Linux Support* category
7. Choose *SHARC and Blackfin Linux Command-Line Tools*, version x.y.z.<*timestamp*>
8. Click *Next>*
9. Click *Finish*
10. Close CrossCore Embedded Studio

7.3 Installing Manually From the Command Line

In these steps, replace "x.y.z" with the appropriate version number for your CrossCore Embedded Studio installation.

1. Download the JAR package from http://www.analog.com/static/ccesupdatesite/blackfin_sharc_linux/x.y.z-SNAPSHOT/plugins/com.analog.crosscore.incubation.blackfin_sharc_linux.stage_x.y.z.<timestamp>.jar
    - The <timestamp> can be obtained by finding the installation first using CCES following the steps above to step (7). You can also do this using CCES on a Windows machine.
2. Make sure the unzip utility is available:
   ```
   sudo apt install unzip
   ```
3. Extract the payload tarball from the JAR:
   ```
   unzip com.analog.crosscore.*.jar support_files.tar.gz
   ```
4. Unpack the tarball into the CCES install:
   ```
   sudo tar xvf support_files.tar.gz -C /opt/analog/cces/x.y.z
   ```

8 Obtaining Technical Support

You can reach Analog Devices software and tools technical support in the following ways:

- Post your questions in the software and development tools support community at EngineerZone®
- E-mail your questions about software and development tools directly from CrossCore Embedded Studio by choosing Help > Email Support or directly to processor.tools.support@analog.com
- E-mail your questions about processors and processor applications to processor.support@analog.com
- Submit your questions to technical support directly via http://www.analog.com/support
- Contact your Analog Devices sales office or authorized distributor