Release Notes for CrossCore Embedded Studio 2.1.0

Release Notes for CrossCore Embedded Studio 2.1.0
August 2015

http://www.analog.com
processor.tools.support@analog.com

Contents

1 Introduction

This document describes the changes for CrossCore Embedded Studio (CCES) 2.1.0. You can find the release notes for older releases in the `docs` sub-directory of your CCES installation.

1.1 Supported Operating Systems

This release of CCES is supported on the following operating systems:

- Windows Vista Business, Enterprise, or Ultimate SP2 (32-bit only)
- Windows 7 Professional, Enterprise, or Ultimate (32 and 64-bit)
- Windows 8.1 Pro or Enterprise (32 and 64-bit)

> **Note**
>
> Windows Vista, Windows 7, and Windows 8.1 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.

1.2 System 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
- 1 GB RAM; 4GB or more recommended
- 2 GB available disk space
- One open USB port

> **Note**
>
> - A faster disk drive decreases the build time, especially for a large amount of source files. 4GB of RAM or more will substantially increase the performance of the IDE.
> - For proper viewing of documentation under Windows Internet Explorer 9 or greater is recommended.

1.3 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

2 New and Noteworthy

2.1 DSP library optimizations for SHARC processors

A number of DSP library routines have been improved since the release of CCES 1.2.0, giving significant performance improvements. Although most of the improvements have been targeted at ADSP-SC58x (which has an 11-stage pipeline), there have been improvements for all SHARC processors, and the most significant gains are shown here. The actual improvements can vary according to the input values, so these figures represent average performance gains across a range of inputs for ADSP-21469 and ADSP-SC589 processors and are intended to be indicative only. All functions are 32-bit versions unless stated otherwise.

Library routine	Performance improvement for ADSP_21469 vs. CCES 1.2.0	Performance improvement for ADSP-SC589 vs. libraries in CCES 2.1.0
a_compress() (*vector version)*	20%	30%
alog()	30%	30%
arg()	20%	20%
atan()	30%	30%
atan2()	25%	20%
autocorr()		30%
biquad() (*scalar version)*	15%	40%
biquad() (*vector version)*		70%
cabs()	30%	30%
cartesian()	35%	35%

Library routine	Performance improvement for ADSP_21469 vs. CCES 1.2.0	Performance improvement for ADSP-SC589 vs. libraries in CCES 2.1.0
cdiv()	15%	20%
cfft()		15%
cfft_mag()		40%
cmatmadd()		20%
cmatmmlt()		40%
cmatmsub()		20%
cmatsadd()		25%
cmatsmlt()		25%
cmatssub()		25%
cos()		20%
cosh()	25%	25%
cot()	25%	15%
crosscorr()		25%
csub()	5%	10%
cvecsmlt()		15%
fft_magnitude()	15%	25%
fftf_magnitude()	15%	20%

Library routine	Performance improvement for ADSP_21469 vs. CCES 1.2.0	Performance improvement for ADSP-SC589 vs. libraries in CCES 2.1.0
fir() (*vector version*)		45%
fir() (*scalar version)*		15%
fir_decima()		30%
fir_interp()		35%
gen_bartlett()	50%	50%
gen_blackman()	40%	35%
gen_gaussian()	5%	25%
gen_hamming()	5%	15%
gen_hanning()	5%	15%
gen_harris()		20%
gen_triangle()	35%	20%
gen_vonhann()	5%	15%
gen_kaiser()	15%	20%
histogram()		35%
iir() (*scalar version*)		25%
iir() (*vector version*)		35%

Library routine	Performance improvement for ADSP_21469 vs. CCES 1.2.0	Performance improvement for ADSP-SC589 vs. libraries in CCES 2.1.0
ifft()		15%
ifftf()		20%
log()	15%	15%
log() (*simd version*)	50%	20%
matmmlt()		10%
matmsub()		10%
matsadd()		20%
matsmlt()		15%
matssub()		15%
rfft_mag()		35%
rfftf_2()		15%
rms()		30%
sin()		15%
sin() (*simd version*)	30%	10%
sinh()	30%	25%
sqrt()	25%	25%

Library routine	Performance improvement for ADSP_21469 vs. CCES 1.2.0	Performance improvement for ADSP-SC589 vs. libraries in CCES 2.1.0
sqrt() (*simd version*)	45%	5%
var()		30%
vecsadd()		15%
vecsmlt()		15%
vecssub()		15%

2.2 Support to connect to multiple ADSP-BF70x processors in the same JTAG scan chain

Support has been added for multiple ADSP-BF70x processors in the same scan chain. See section 'Configuring a Scan Chain' in the CCES online help for more information on JTAG scan chains.

2.3 QEMU support for all ADSP-SC58x processors

In addition to ADSP-SC589, the ADSP-SC582, ADSP-SC584 and ADSP-SC587 ARM cores can now be simulated using GDB with QEMU Simulator.

2.4 Improved default debug configuration settings for the ADSP-SC589 /ADSP-21584 Functional Simulators

It is no longer necessary to modify the 'Reset' and 'Run Immediately After Load' settings in an ADSP-SC589/ADSP-21548 CrossCore Debugger Functional Simulator session prior to launching.

2.5 Ethernet controller driver for ADSP-SC589 is enhanced

- Precision Time Protocol (PTP) support is added to the driver.

- Supports Audio Video feature which allows time sensitive real time audio video data transmission.
- Added Pulse-Per-Second (PPS) support.

2.6 Additional documentation for Debugging on ADSP-SC58x/ADSP-2158x in EngineerZone

Please check EngineerZone for Debugging on ADSP-SC58x/ADSP-2158x Targets.

3 Known Issues

> **Viewing Issues on the Web**
>
> Note that some of these issues have identifiers associated with them (e.g. `CCES-12345`). You can visit our Software and Tools Anomalies Search page and enter the issue ID into the `Reference Number(s)` field to view additional details and the see the latest status of these issues.

3.1 Message about needing to run OpenOCD as sudo when debugging with GDB and OpenOCD (Emulator) on Windows (CCES-13604)

If you debug an Application with GDB and OpenOCD (Emulator) on Windows and OpenOCD is unable to connect to the hardware due to a USB error (libusb_open() failed with LIBUSB_ERROR_ACCESS), a misleading dialog box will appear.

The dialog displays a message that OpenOCD failed to run, because it needs root permission. OpenOCD did fail to run, but the reason is not that it required root permission. The most likely causes are that another debug session is already open in the Debug window, or that the wrong emulator or board is configured in the Debug Session Target tab. More details about the OpenOCD failure can be seen in the IDE Output Console View.

3.2 Some system services and device driver Add-in components may fail to upgrade automatically when opened in a newer version of CCES (CCES-13717)

If your CCES 2.0.0 project uses a System Services and Device Driver Add-in component, such as an Ethernet Driver, then opening your project with CCES 2.1.0 may result in an error (a popup dialog and an error in an IDE Console View).

```
2015-08-10 17:57:20, 261 [Thread: Worker-6] ERROR - Source doesn't
exist: C:\Analog Devices\CrossCore Embedded Studio 2.0.0\ARM\arm-
none-eabi\arm-none-eabi\lib\src\cortex-
a5\drivers\Source\ethernet\gemac\adi_gemac_av.c
```

`The System Services and Device Driver Add-in components are upgraded automatically`

The error can be resolved after your project is opened by removing and re-adding the problem System Services and Device Driver Add-in component(s).

3.3 Eclipse internal error dialog pops up when trying to delete a opened binary file (CCES-13455)

If a binary file is open in a CCES Editor and the file is deleted using Windows Explorer or the project in your Project Explorer is deleted, then you may encounter an error dialog telling your that there were "problems encountered while deleting resources".

Click OK to continue using CCES.

3.4 adi_core.h cannot be found if the IDE is installed to a non default location (CCES-12686)

If you have installed CCES in a non-standard directory (i.e. not in the default C:\Analog Devices\CrossCore Embedded Studio 2.1.0 directory), then you may encounter an error while debugging code in a system header file, such as adi_core.h.

The problem is that after stepping into the system header file code, the debugger cannot find the corresponding system header source. An error will appear in the CCES Editor telling you a source file could not be found. To continue debugging the system header file code, click the Locate File... button and browse for the source file which is located in your non-standard CCES installation directory.

3.5 SDRAM size cannot be changed from default (CCES-13581)

If you enable External memory in the Startup Code/LDF configuration UI for a non ADSP-SC58x project and you choose an SDRAM size other than the default, then the generated app.ldf will continue to use the default SDRAM size.

If you would like to use a non default SDRAM size, then you should manually modify the MEM_DMC_SDRAM_BANK3 memory segment END address in the generated app.ldf. Note that your change will be clobbered if app.ldf is regenerated.

3.6 Failed to buid ldr file when the project name contains a space (CCES-13578)

If your project contains a space in its name and your project's artifact is set up to create a Loader File, then your build will fail, because the elfloader tool cannot open the DXE file. The tool is not recognizing the escaped path to your DXE.

3.7 Load Complete, but nothing showing on the Debug Window, on some PC's (CCES-13783)

On some systems, when using 'Debug' or 'Debug As' to launch a debug session with the CrossCore Debugger, the IDDE console will show 'Load Complete', but there will be no Debug information in the Debug Window.

To work around this issue, use 'Debug Configurations' and click on the 'Debug' button to launch.

3.8 The default heap on ARM is too small, and increasing it can result in very large executables (CCES-13603, CCES-13728)

The default heap for the ARM processor on ADSP-SC589 is 0x2000 bytes, which might be too small. The section "Managing the Heap for ADSP-SC58x Processors" in the Help describes how to increase it, but doing so can result in very large executable (as the heap is zero-initialized), so it might be desirable to instruct the linker to leave the heap memory uninitialized. This can be done by applying the "NOLOAD" attribute to the default LD file. To do this, see CCES-13728 in the Tools and Anomalies Search Page.

3.9 OpenOCD/QEMU Step Over adi_core_enable() functions seems to hang (CCES-13197)

Stepping over the adi_core_enable() macro calls that are added to new ADSP-SC58x projects takes a long time and appears to hang. Run past the macros for faster debugging.

3.10 Projects which uses Lwip 2.0.0 may not work with CCES 2.1.0 (LWIP-142).

LwIP 2.0.0 examples and any projects which uses the precompiled library may fail to run in CCES 2.1.0. ADI_ETHER_BUFFER structure in CCES 2.1.0 is different from CCES 2.0.0.

Please recompile lwip and lwip wrapper libraries for the required processor in CCES 2.1.0

3.11 No System Reset

Currently only a core reset is supported on the ADSP-SC58x/ADSP-2158x (Rev 0.1 silicon or older) and the ADSP-BF70x (Rev 0.0 silicon) processors, which has shown limitations when peripherals are running at the time of a core reset. There may be cases where you run an example and then reload to run the example a second time and you get exceptions. This could be due to the peripheral interrupt being serviced at the wrong time causing an exception. In order to fix this, identify the peripheral that is causing the issue and reset that peripheral in a pre-load file. The other option would be to use the hard reset on the target board in between running examples. Using Engineer Zone or sending a private support request is also a good option so that we are aware of the issue and can add it to the default pre-load files for the upcoming release.

The nature of the core-only resets on the ADSP-SC58x and ADSP-2158x processors means that a core has to be running in order for the reset operation to take effect, when that reset is triggered by another core - if a core is halted by the emulator during the reset operation, the reset operation has no effect on the halted core. For this reason, the emulator sets all cores running after an application is downloaded to the processor from the IDE.

Beginning with Rev 1.0 silicon, the ADSP-BF70x supports system reset so the issues associated with a core reset on this processor will no longer be present.

3.12 "Invalid op code: Address 0" is output to the IDE Errors console and Error Log

While debugging on the ADSP-SC58x processors, you may see "invalid op code: Address 0" logged to the IDE Errors console and Error Log. You can safely ignore this error.

3.13 No support to connect to multiple ADSP-SC58x/ADSP-2158x processors in the same JTAG scan chain

There is currently no support for multiple ADSP-SC58x/ADSP-2158x processors in the same JTAG scan chain.

3.14 Debugging F1 – active mode loops on ADSP-SC58x/ADSP-2158x (CCES-12421)

Users cannot debug F1 – active mode loops unless the SLOWLOOP(Bit 7) bit in the Mode Control 2 Register is set. Some of the issues seen when this bit is not set properly are illegal opcode exceptions and incorrect updating of the Current Loop Counter Register(CURLCNTR).

More information can be found in the SHARC+ Core Programming Reference that is found in the CCES online help system. Sections of interest are the Loop Categories and Restrictions on Ending Loops sections where more information can be found about the Loop Sequencer.

3.15 Issues with pre-load file for 21587 (CCES-13693)

When an ADSP-21587 session is created, the session wizard will automatically add a pre-load file to Core 1. This file does not exist and should be removed from the session.

3.16 Unable to connect to target boards with multiple SHARC processors (CCES-13791)

Users will not be able to connect to target boards with more than one SHARC processor using CCES 2.1.0. Connecting to and debugging target boards with only 1 SHARC processor will still function as expected.