

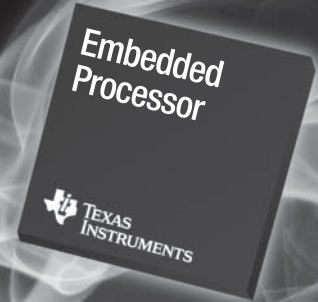


Code Composer Studio™

Integrated Development Environment (IDE) v5

Quick Start Guide

```
void ShapingFails
int i;
/* shift the delayLine array
for( i = 0; i < SIIIR_DELAYLINE_SIZE; i++)
{
    delayLine[i] = 0;
}
/* clear the delayLine array
for( i = 0; i < SIIIR_DELAYLINE_SIZE; i++)
{
    delayLine[i] = 0;
}
/* add new samples to the delayLine array
for( i = SIIIR_DELAYLINE_SIZE - 1; i >= 0; i--)
{
    #if defined(SIIIR_DELAYLINE_SIZE)
    delayLine[i] = 0;
    #endif
}
```



Code Composer Studio™ IDE Version 5

Quick Start Guide

Installation Requirements

These operating platform requirements are necessary to install the Code Composer Studio Integrated Development Environment (IDE):

Minimum

- 1.5-GHz Pentium-compatible CPU
- 1 GB of RAM (2 GB for Windows® 7)
- 300 MB of free hard disk space (minimal custom installation)
- Local DVD-ROM drive

Recommended

- Dual-core processor
- 4 GB of RAM
- 2 GB of free hard disk space (typical installation)

Supported operating systems

- Windows XP (32 and 64 bit)
- Windows 7 (32 and 64 bit)
- Linux™
 - Check the Wiki for more details on Linux Host support:
processors.wiki.ti.com/index.php/Linux_Host_Support

Installing the Code Composer Studio IDE

1. Insert the installation DVD in the DVD drive.
2. Go to Windows Explorer and run “**ccs_setup_5.x.x.xxxxx.exe**” from your DVD.
3. Respond to the dialog boxes as the installation program runs. You can customize the components that are installed. A standard evaluation license agreement will appear during installation. Note that when the Code Composer Studio IDE is activated using an activation code (see section 3 of “Launching the Code Composer Studio IDE”), another license agreement will appear that is specific to the license activated.

Note: Depending on your operating system, there are some recommended steps to take:

- **WINDOWS®:** By default, Microsoft Windows Vista and Windows 7 do not allow user level accounts to write files in the “Program Files” directory. This can prevent the Code Composer Studio™ IDE from starting when installed in the “Program Files” or “Program Files (x86)” directory, because the Code Composer Studio IDE stores configuration data by default in its install directory. Either of the following workarounds can be used to avoid this problem:
 - Install the Code Composer Studio IDE outside the “Program Files” or “Program Files (x86)” directory
 - Instruct the Code Composer Studio IDE to store configuration data outside the “Program Files” or “Program Files (x86)” directory using the -configuration command line argument. The problem will not occur when running the Code Composer Studio IDE as the Administrator user, but this is not recommended because it compromises the Windows Vista security model
- **LINUX™:** It is recommend to install as “root”. This avoids many issues with emulation device drivers.
 - From a terminal, run the installer as **sudo ccs_setup_5.x.x.xxxxx.bin &**
 - At the end of the install, DO NOT check the box Launch Code Composer Studio.
 - This option launches the Code Composer Studio IDE with root privileges, which will naturally create all the temporary cache files with such privileges. This will disrupt subsequent launches of the Code Composer Studio IDE since these files are non-writable by the regular user.

Launching the Code Composer Studio IDE

The following steps apply to launching the Code Composer Studio IDE for the first time.

1. Invoke the Code Composer Studio IDE by double clicking on the “Code Composer Studio” icon that appears on your desktop.

2. A prompt will appear asking for a workspace location. The workspace is the main working folder for the Code Composer Studio™ IDE. The default location will be (on Windows® XP) “C:\Documents and Settings\\My Documents\workspace_v5_1”. This path may vary on other operating systems. The Code Composer Studio IDE will ask for the location of the workspace each time on startup. To have the Code Composer Studio IDE automatically default to the specified workspace location, select the “Use this as the default and do not ask again” option.

3. A dialog box will appear, requesting one of the following licensing options be chosen:

- **ACTIVATE:** This will provide options to activate the Code Composer Studio IDE with an activation code, license file, or floating license. Select the option for “ACTIVATE” and hit the “Next” button at the bottom of the dialog box to continue to the activation dialog. Then follow the steps below:

i. **Step 1 – Generate License File:** Choose the license type:

Note: If you already have a valid license file, or have access to a floating license on a server, you can skip this step and go to Step 2.

1. **Node-Locked License:** Record one of your MAC addresses (displayed above the “Register” button). A MAC address will be needed to generate a license. Then press the “Register” button. This will open a web page in your web browser to guide you through the steps of generating a license file with your activation code and MAC address. The generated license file will then be e-mailed to you.
2. **Floating License:** Select the “Register” button to generate a license for a multi-user floating license. This will open a web page in your web browser to guide you through the steps of generating a license file with your activation code. The generated license file will then be e-mailed to you along with instructions on how to setup the license on a server.

ii. **Step 2 – Install License File:** Choose the license type:

1. **Node-Locked License:** Select the “Specify a license file” option to install a node-locked license file that has been received by e-mail. Select the file using the “Browse...” button.
2. **Floating License:** Select the “Specify a license server” option and enter the “Address” and “Port Number” information for the floating license installed on a server.

Hit the “Finish” button at the bottom of the dialog box. This will activate the Code Composer Studio™ IDE.

- **EVALUATE:** This will activate the Code Composer Studio IDE for 90 days for free evaluation without any feature restriction. Simply select the option to “EVALUATE” and hit the “Finish” button at the bottom of the dialog box to continue.
- **FREE LICENSE:** This will activate the Code Composer Studio IDE with a FREE license with no time restriction. This option is limited to the following scenarios:
 - i. Using the XDS100 JTAG emulator
 - ii. Using onboard emulators in EVMs/DSKs/Stellaris/eZdsp/MAVRK development kits (eZ430 kits are NOT supported). Linux™ drivers are not available for many onboard emulators. Check the Wiki for more details on Linux Host support: processors.wiki.ti.com/index.php/Linux_Host_Support
 - iii. Using simulators
 - iv. Using the Code Composer Studio IDE for Linux/Android™ application development using GDBSimply select the option for “FREE LICENSE” and hit the “Finish” button at the bottom of the dialog box to continue.
- **CODE SIZE LIMITED:** This will activate the Code Composer Studio IDE with a FREE license for MSP430™ with no time restriction, but with a code size limit of 16 KB. Simply select the option for “CODE SIZE LIMITED” and hit the “Finish” button at the bottom of the dialog box to continue.

4. The “Welcome Page” will appear in the “TI Resource Explorer”, welcoming you to the Code Composer Studio™ IDE. On this page, there are several links that can be clicked to help get started with the Code Composer Studio IDE such as creating a new project, importing existing projects, links to examples, links to additional resources on the web, and so forth. Once you are finished with the “Welcome Page”, you can then proceed to the main Code Composer Studio IDE workbench by closing the “Welcome Page.”
5. The Code Composer Studio IDE workbench will appear and be in the “CCS Edit” Perspective.
6. The Code Composer Studio IDE will check for updates and will notify you if updates have been detected. You can then choose to have the Code Composer Studio IDE download and install the updates or decide to update the Code Composer Studio IDE at a later time.

Using the Code Composer Studio IDE to Create a Project

To create a new project:

1. Select “File → New → CCS Project” from the menu. This opens the “New CCS project” wizard.
2. In the “Project Name” field, type the new project name.
3. In the “Output type”, select “Executable” or “Static Library”, depending on the type of output you are creating.
4. The “Use default location” option, if enabled, will create the project in your workspace folder. The option is enabled by default. If desired, uncheck the option to select a new location (by using the “Browse...” button).
5. Specify the architecture in the “Family” field using the drop-down menu.
6. In the “Variant” field, select the device you are creating the project for. If your specific device is not present, select the “Custom” or “Generic” device option that is closest. Selecting an option in this field will apply some default settings for the fields in the “Advanced settings”, such as picking a runtime support library and linker command file to use.

- a. If a specific device was selected in the “Variant” field (as opposed to “Generic” or “Custom”), the “Connection:” field can be used to specify the connection type between the Code Composer Studio IDE and the target. Specifying this field will have the project wizard automatically generate a target configuration file in the project and the section “Creating your Target Configuration File” can be skipped.
7. The “Advanced settings” section is minimized by default. If expanded, settings for device endianness, compiler version, output format, linker command file and runtime support library can be specified. The default settings specified from Step 6 can be modified if desired.
8. A project template can be specified. Project templates will generate some example code which can be used to help get started. The types of templates available will vary per Device Family.
9. Your project has now been created and added to the workspace. The contents of the project will be visible from the “Project Explorer” view, even on re-launching the Code Composer Studio IDE, as long as the workspace folder is not changed.

To add files to a project:

Note: Most project template examples will already have generated source files added to the project. No additional source files are needed to build the examples and the following steps adding files to the project are not necessary.

- Create/add new source files to your project by right-clicking on the project in the “Project Explorer” view and selecting “New → Source File” in the context menu. This opens the “New Source File” dialog box. In the “Source File” field, type the new project name. Make source the name also has the file extension (ex.: main.c). Click “Finish” to exit the dialog. This will open a blank file of the specified name in the editor and add the file to the project.
 - (Optional): The Code Composer Studio IDE comes with some code templates that can be used from the editor. Code templates are templates of code that can be

referenced using the “Content Assist” feature of the editor. Custom code templates can also be created. It is a quick way to get started with the Code Composer Studio™ IDE.

1. MSP430™: Type the letter “b” in blank file in the editor. Then press “CTRL-SPACE” (press the “CTRL” key and “SPACE” key at the same time). This will add source for the MSP430 “blink LED” example.
 2. Other ISA: Type the letter “h” in blank file in the editor. Then press “CTRL-SPACE” (press the “CTRL” key and “SPACE” key at the same time). This will bring up a list of code template options for “hello world”. Selecting the “helloworld” option will add the source for a standard “hello world” example.
- To add existing source files, right-click on the project in the “Project Explorer” view, select “Add Files...”, and browse to the file you wish to add. When a file is selected, a prompt will appear, asking if the Code Composer Studio IDE should “Copy files” (make a physical copy of the file and put it in the project folder) or “Link to files” (create a reference to the file).
 - When “Link to files” is selected, a linked resource path variable may be specified in a drop down list to create the reference using a variable instead of an absolute path. This helps to make the project “portable”.

Creating Your Target Configuration File

Before a debug session can be launched, a target configuration file is needed to configure the Code Composer Studio IDE for the target board or simulator that you want to use. This file can be added to a project or exist standalone. The steps below will show how to add it to a project.

Note: *If a specific device variant and connection was specified in Step 6 of “To create a new project”, then a target configuration file has been automatically created and added to the project after project creation. The following steps below for creating a target configuration file are not needed.*


1. Right-click on the project in the “Project Explorer” view and select “New → Target Configuration File”. This opens the “New

Target Configuration” dialog box. In the “File name” field, type in a new file name. Give it a file extension of (*.ccxml). Click “Finish” to exit the dialog. This will open the target configuration file in the editor.

2. In the “Connection” field, select the connection type by enabling the checkbox for the desired device.
3. In the “Device” field, select the device to connect to. Use the text filter field above the list of devices to shorten the list.
4. Save the file by hitting the “Save” button under “Save Configuration”.

Note: *If your device does not appear in the list of devices, you will need to use the “Advanced” setup tool to create it. Check the documentation for more information on how to use the “Advanced” setup tool.*

Building and Running the Program

- **Setting Build Options:** Right-click on the project in the “Project Explorer” view and select “Properties...” from the context menu. You can change options for the compiler and linker under the “Build” properties.
- **Building and loading the program:** Press the “Debug” button  to build the project, switch to the “CCS Debug” perspective, start a debug session for the target specified in the target configuration file, load the executable file (*.out), and run to “main”. All of these steps are done automatically. Now you are ready to start debugging or running your application. Note that if any build errors are encountered and no executable file is created, a warning will appear and a debug session will not be started.

Resources

- **Welcome Page:** The Welcome Page has links to additional documentation, guides, examples and such. The Welcome Page can be opened by selecting “Help → Welcome to CCS” from the main menu.

- **Online Help:** The Code Composer Studio™ IDE is accompanied by searchable online documentation. Select “Help → Help Contents” to bring up the online help.
- **Code Composer Studio IDE v5 Wiki:** The Code Composer Studio IDE v5 Wiki site has the latest information regarding the usage of the Code Composer Studio IDE. It provides a wealth of information on the Code Composer Studio IDE, flash captures on how to use various features, information on how to obtain support and much more. A link to the Wiki site is available via “Help → CCSv5 Developer Site”. processors.wiki.ti.com/index.php/Category:Code_Composer_Studio_v5
- **TI E2E™ Community:** The TI E2E Community is a place for users to discuss and find answers to questions regarding the Code Composer Studio IDE, devices and software. A link to the forum site is available from the Welcome Page (under “Support”). www.ti.com/e2e

Notes

Notes

The platform bar, Code Composer Studio, e2e and MSP430 are trademarks of Texas Instruments.



Copyright © 2011 Texas Instruments

Printed in U.S.A.

SPRM382A