

Tiva™ C Series Development and Evaluation Kits for Code Composer Studio™

Tiva C Series Development and Evaluation Kits provide a low-cost way to start designing with Tiva microcontrollers using Texas Instruments' Code Composer Studio development tools. The evaluation boards can function as either a complete evaluation target or with minor modifications as a debugger interface to any external Stellaris device.

1 Requirements

- PC with a USB interface, running Microsoft Windows® 2000, Windows XP, Windows 7, or Windows 8 operating systems (OSs).
- Tiva C Series Development Kit Software USB flash drive or the downloaded and extracted kit software (found on www.ti.com/tool/sw-<kit_name>)
- ICDI drivers installed following the instructions in *Stellaris® In-Circuit Debug Interface (ICDI) and Virtual COM Port* ([SPMU287](#)).

2 Code Composer Studio

This quickstart shows how to install the Code Composer Studio development tool and how to use it to build and run an example application on your Tiva Evaluation or Development Board.

2.1 Installation

2.1.1 Install Code Composer Studio

1. Insert the Tiva C Series Development Kit Software USB flash drive and click on the index.html. For kits that don't include a USB flash drive, go to the evaluation board software webpage (www.ti.com/tool/sw-<kit_name>).
2. With the Evaluation Kit USB flash drive, click the Code Composer Studio logo to start the setup program. For the downloaded software, use Windows Explorer to navigate to Tools > CCS and double-click on ccs_setup_<version number>.exe.

NOTE: The CCS installer executable relies on additional installer executables on the USB flash drive. Your web browser must support running files without first saving them to your hard drive to successfully run the CCS installer by clicking the CCS logo. If your web browser does not support this feature, Windows Explorer must be used to browse to and run the CCS installer (setup_CCS_n.n.n.nnnnn.exe) from the \Tools\CCS\ directory on the USB flash drive.

3. Follow the instructions in the Code Composer Studio installation program. Select a *Complete Feature Set* install or a *Custom* install with the *Stellaris® (or Tiva C Series in the future) ARM® Cortex™-M MCUs* selected at a minimum. For all of the other options, keep the default values.

Tiva, Code Composer Studio, TivaWare are trademarks of Texas Instruments.

Stellaris is a registered trademark of Texas Instruments.

Cortex is a trademark of ARM Limited.

ARM is a registered trademark of ARM Limited.

Microsoft Windows is a registered trademark of Microsoft Corporation.

All other trademarks are the property of their respective owners.

2.1.2 Install the TivaWare™ Package

A full set of C-based peripheral drivers is provided, covering all peripherals and functionality of the Tiva devices. The TivaWare for C Series package includes various example applications with project files for all major tool vendors that support Tiva C Series, including Code Composer Studio. To install TivaWare components, follow these steps:

1. Navigate to the *Tools* tab on the html index of the Development Kit Software USB flash drive, click on the Firmware Development Package. Depending on your web browser, you have the option to run the TivaWare installer or save it to your drive.
NOTE: If you are navigating the USB flash drive using Windows Explorer (or a similar application), go to the *Tools/TivaWare*/directory.
2. Run the TivaWare installer. If manual installation of TivaWare is preferred, the installer is a self-extracting zip file that is located in the *Tools/TivaWare* directory. A zip file extraction utility such as WinZip can be used to manually extract the contents.
3. To view the TivaWare documentation, navigate to the *Tools/TivaWare/docs* directory and open the *Tiva for C Series Peripheral Driver Library User's Guide* PDF.

NOTE: For the most recent version of TivaWare, check the www.ti.com/tiva-c web site.

2.2 Start Code Composer Studio and Open a Workspace

1. Start the Code Composer Studio IDE by selecting it from the Windows *Start* menu or double-clicking the icon installed on the desktop. When the IDE loads, it asks where to open the workspace folder:

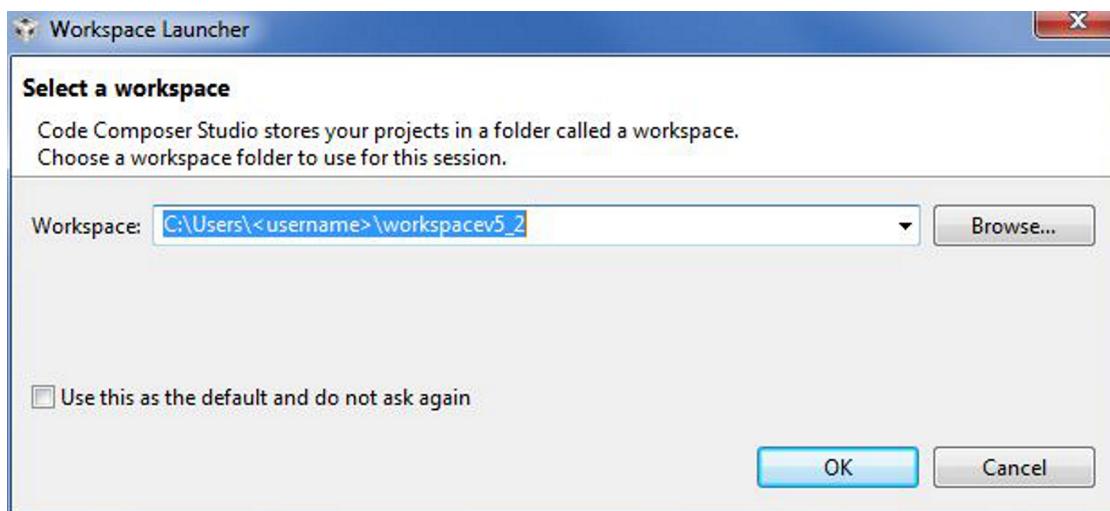


Figure 1.

2. The Workspace Launcher defaults to the following path: *C:\Users\<username>\workspace_v5_2*. Click OK to use this default workspace location.
NOTE: If the Code Composer Studio IDE is installed from the USB flash drive, the license was automatically installed. If the Code Composer Studio IDE is downloaded, an extra licensing step may be necessary.
3. The Code Composer Studio IDE may now open with the welcome page. If so, close the welcome page by clicking the X on the tab. You should now have an empty workspace.

2.3 Import Libraries

1. Select the *Import Existing CCS Eclipse Project* option from the *Project* menu in the IDE.

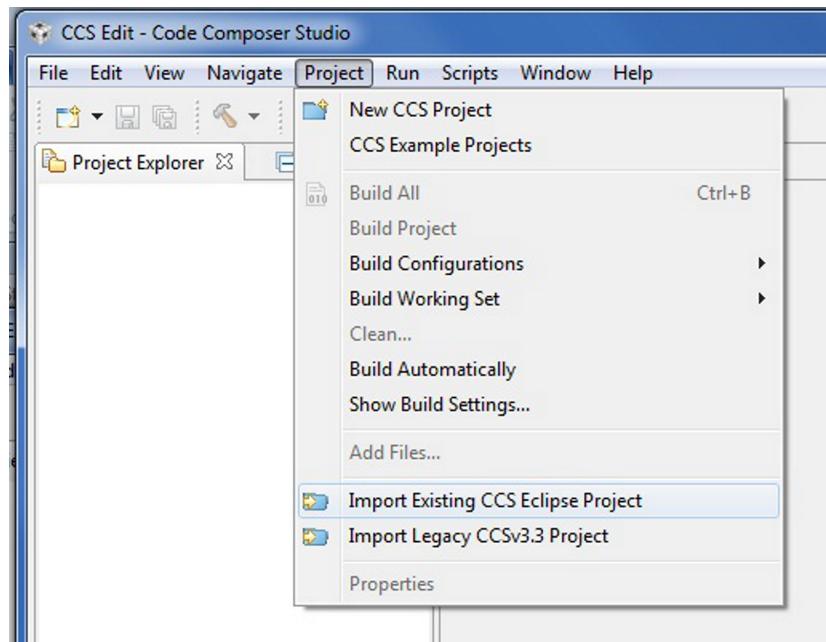


Figure 2.

2. In the *Import CCS Eclipse Projects* window, browse to the root directory for the Tiva peripheral driver library (`t\TivaWare_C_Series-n.n\driverlib`). Select the *driverlib* option displayed in the *Discovered project* field. Click *Finish* after the selection has been made.

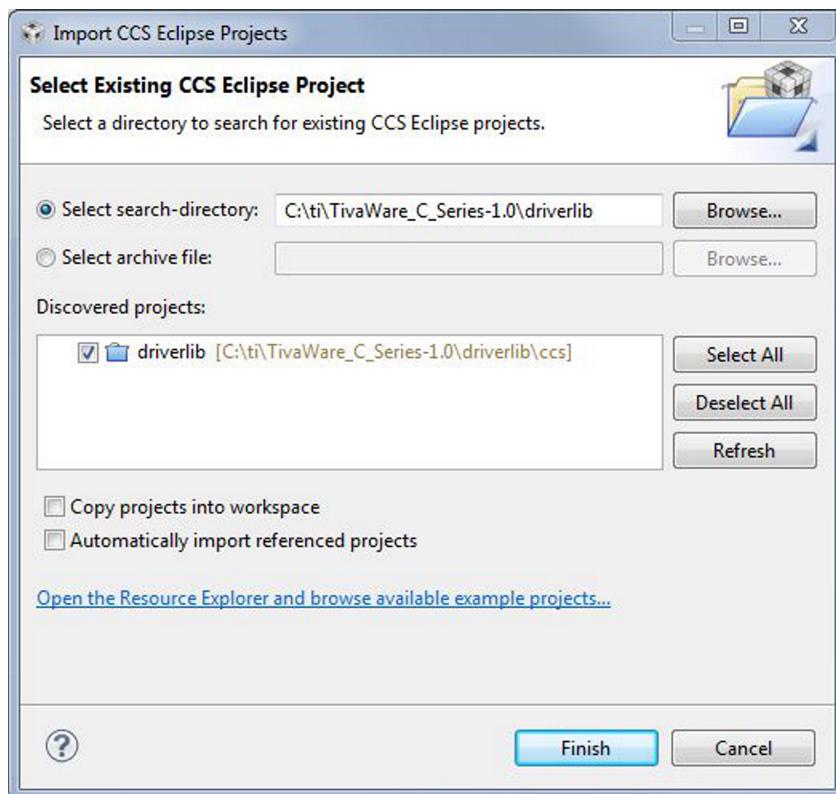


Figure 3.

IMPORTANT: If the TivaWare Firmware Development package has been installed for a board that does not use the USB library or Graphics library, the following steps do not apply. If the directories mentioned below do not exist in your TivaWare installation, then click *Cancel* to close the *Import CCS Eclipse Projects* dialog box.

Repeat the [Install the TivaWare Package](#) section to import the USB Library (usblib) and the Graphics Library (grlib) as follows:

3. Select the *Import Existing CCS Eclipse Project* option from the *Project* menu in the IDE again. Browse to the root directory of USB library (*ti\TivaWare_C_Series-n.n*). Select the usblib option. Click *Finish* after you have made your selection.
4. Select the *Import Existing CCS Eclipse Project* option from the *Project* menu in the IDE again. Browse to the root directory of the graphics library (*ti\TivaWare_C_Series-n.n*). Select the grlib option. Click *Finish* after you have made your selection.

2.4 Import Board Examples

1. Select the *Import Existing CCS Eclipse Project* option from the *Project* menu in the main menu bar.
2. The *Import* dialog box appears. Browse to the root directory for your board. The image below shows the EK-TM4C123GXL board as an example (*TivaWare_C_Series-n.n\examples\boards\ek-tm4c123gxl*). Make sure that the examples are selected in the *Discovered projects* field, then click *Finish*.

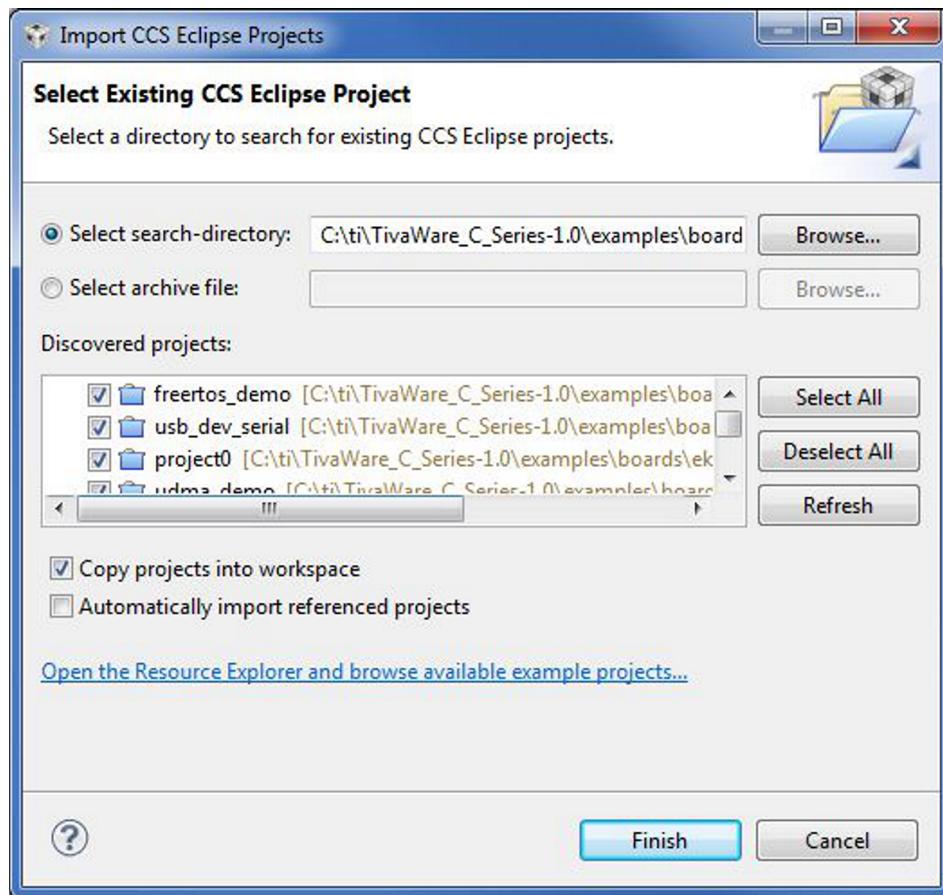


Figure 4.

NOTE: It is not required to copy projects into the workspace, however, for board examples, it is recommended that projects be copied into the workspace.

3. All of the projects now display in the Projects Explorer window.

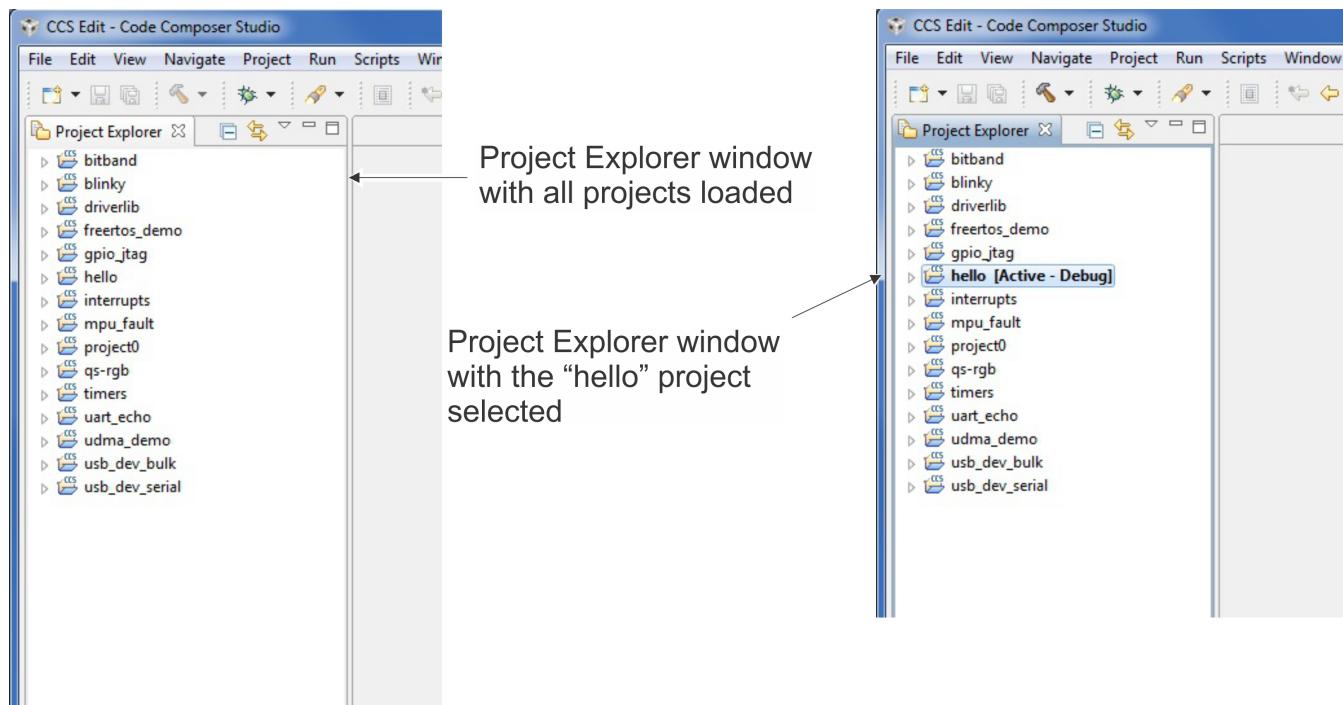


Figure 5.

2.5 Build a Debug Project

1. Click the *hello* project in the Project Explorer window to set as the Active project (see [Figure 5](#)).
2. Select *Build Project* from the Project menu in the main menu bar. The build may take a few minutes. As the project builds, messages scroll in the console window. When the build is complete, the words *Build Finished* appear in the console window.
3. Select *Debug* from the Run menu in the main menu bar. The Debug option can also be selected by clicking the *Debug Launch* icon on the toolbar, or pressing the F11 function key on your keyboard.
4. The Code Composer Studio debugger automatically connects to your development or evaluation board, programs the Flash memory, and runs to the beginning of the *main()* function. From here, it is possible to examine and modify memory, program variables and processor registers, set breakpoints, step, and perform other typical debugging activities. To run the program, select *Resume* from the *Run* menu.

2.6 Build and Run Additional Example Programs

There are several additional example projects listed in the Project Explorer window. Follow the instructions in [Build a Debug Project](#) to build additional example programs. The quickstart application that came preloaded on the evaluation board is the *qs-xxxxx* project listed with the examples.

3 Creating a New Project

After you are finished exploring and building the TivaWare example applications, you might want to create your own project to start development. Although you can start with an existing project, you can also create a new project.

The example shown below creates a new project, copies code from an existing project, and then builds the new project.

1. To add a new project to your workspace, select File→New→CCS Project from the main menu bar.
2. The *New CCS Project* window appears which allows the creation of a new CCS project. Update the following fields (shown in [Figure 6](#)): Project name, Location, Family, Variant (device), Connection, Device endianness, Compiler version, and Output format. Leave the Linker command file and Runtime support library with their default settings (<automatic>).

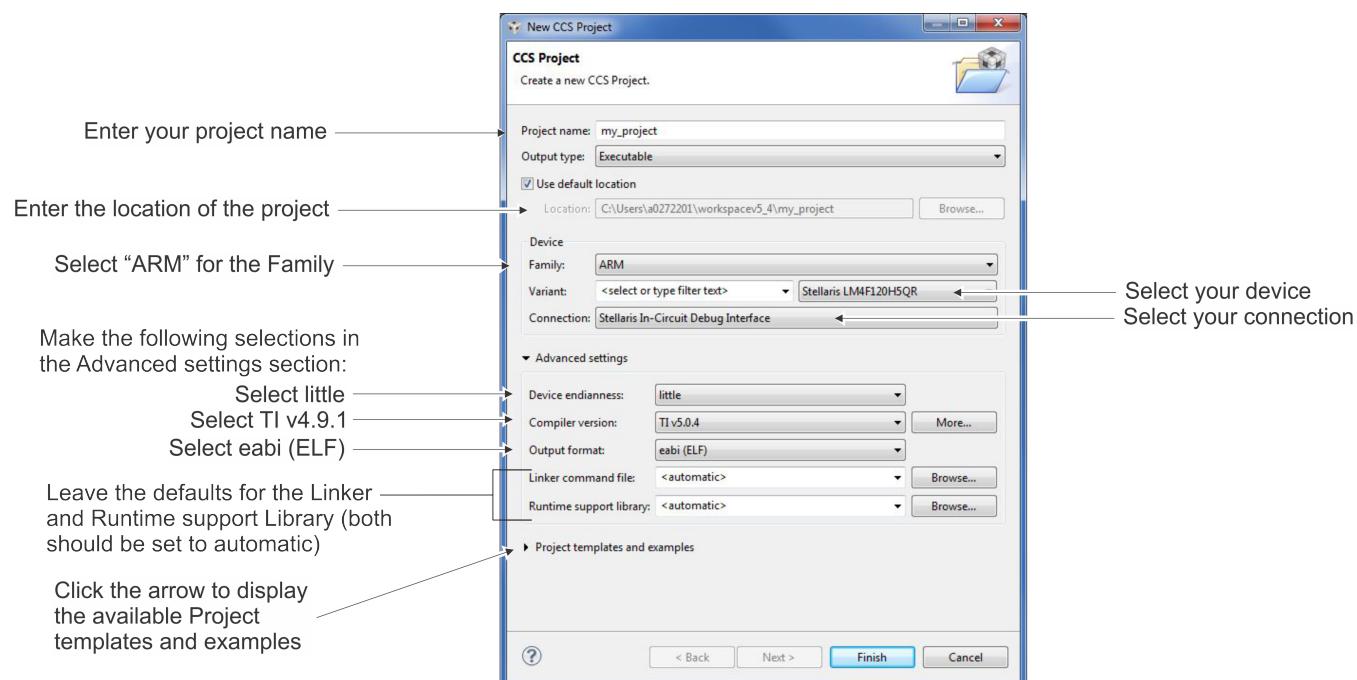


Figure 6.

3. Then click the arrow to display the options under *Project templates and examples*. Select *Empty project*, then click **Finish**.

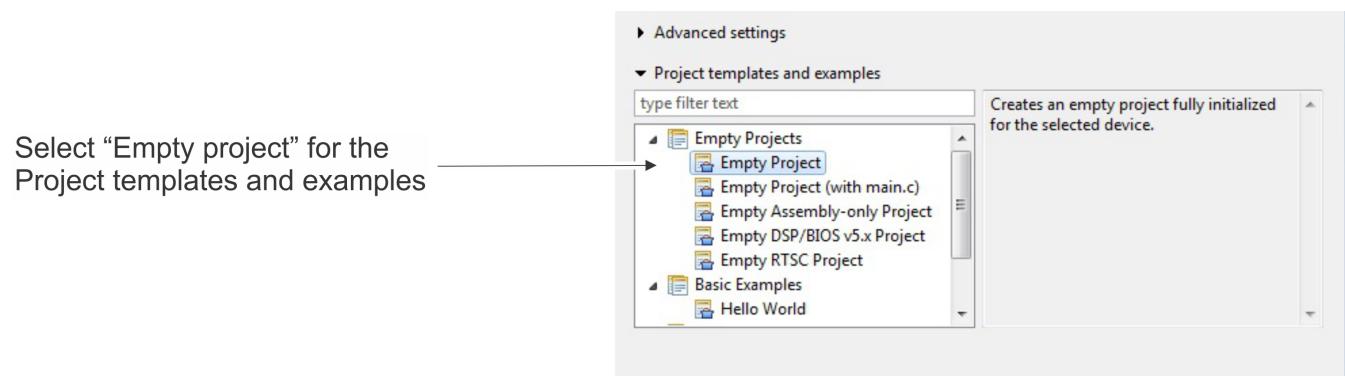


Figure 7.

4. A new project has now been created. The project includes the main.c source file, but does not contain any source code. Add a loop forever function to the main.c file and then save the main.c file.

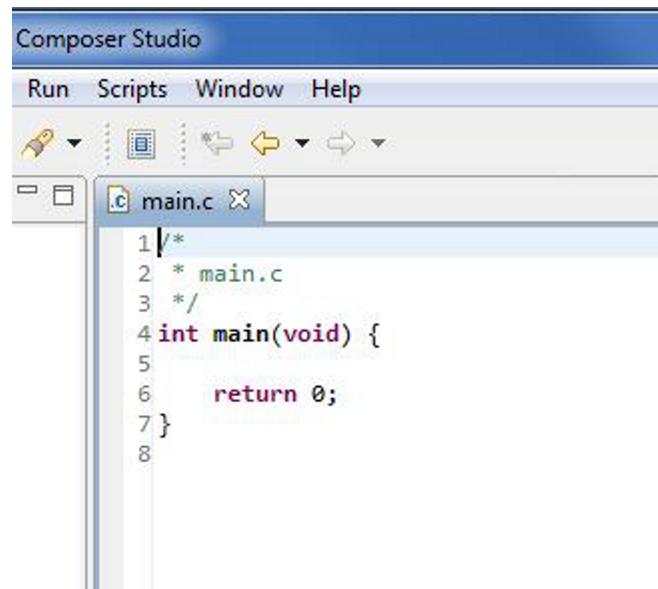


Figure 8.

5. Add some startup code to the project by copying the startup_ccs.c file from an existing example directory in TivaWare (such as *ti\TivaWare_c_Series-n.n\examples*) to your new project directory created in the [Install the TivaWare Package](#) section above.
6. The main.c and startup_ccs.c files have automatically been added to the project. If the files were not automatically added, right-click the project, select *Add Files...* from the drop-down menu, and then browse to the files in the new project folder to select.
7. Build the new project by selecting *Project→Build Project* from the main menu. Your new project has been created. Now add your own code using the existing TivaWare board examples as a reference.
8. To create a new C source file, select *File→New→Source File* from the main menu.
9. In the New Source File dialog box that appears, enter a file name and click *Finish*.

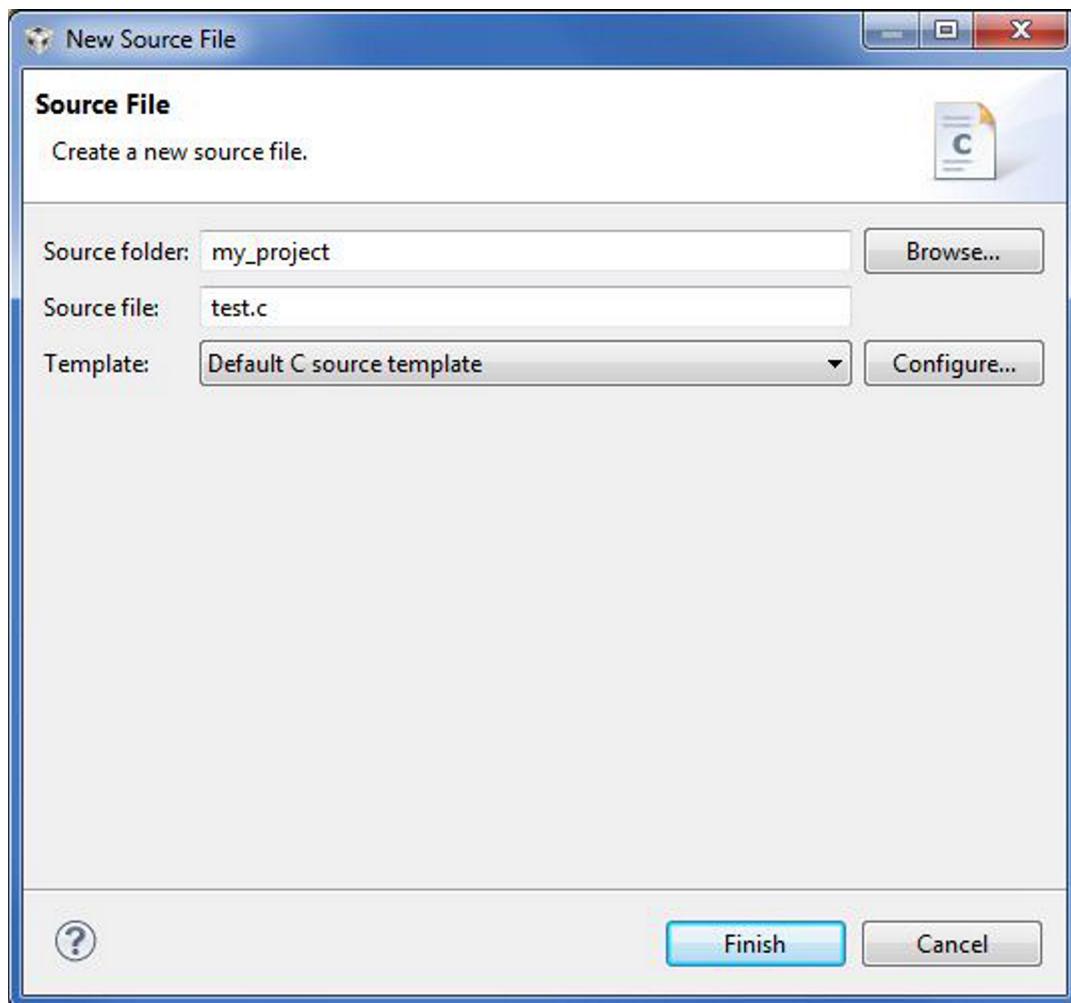


Figure 9.

4 Optional: Set Up Project to Output a Binary File

The project can be set up to output a binary file (.bin) to be used with tools such as LM Flash Programmer by copying the post-build step from an existing StellarisWare example project. Right click the project and select *Show Build Settings...*

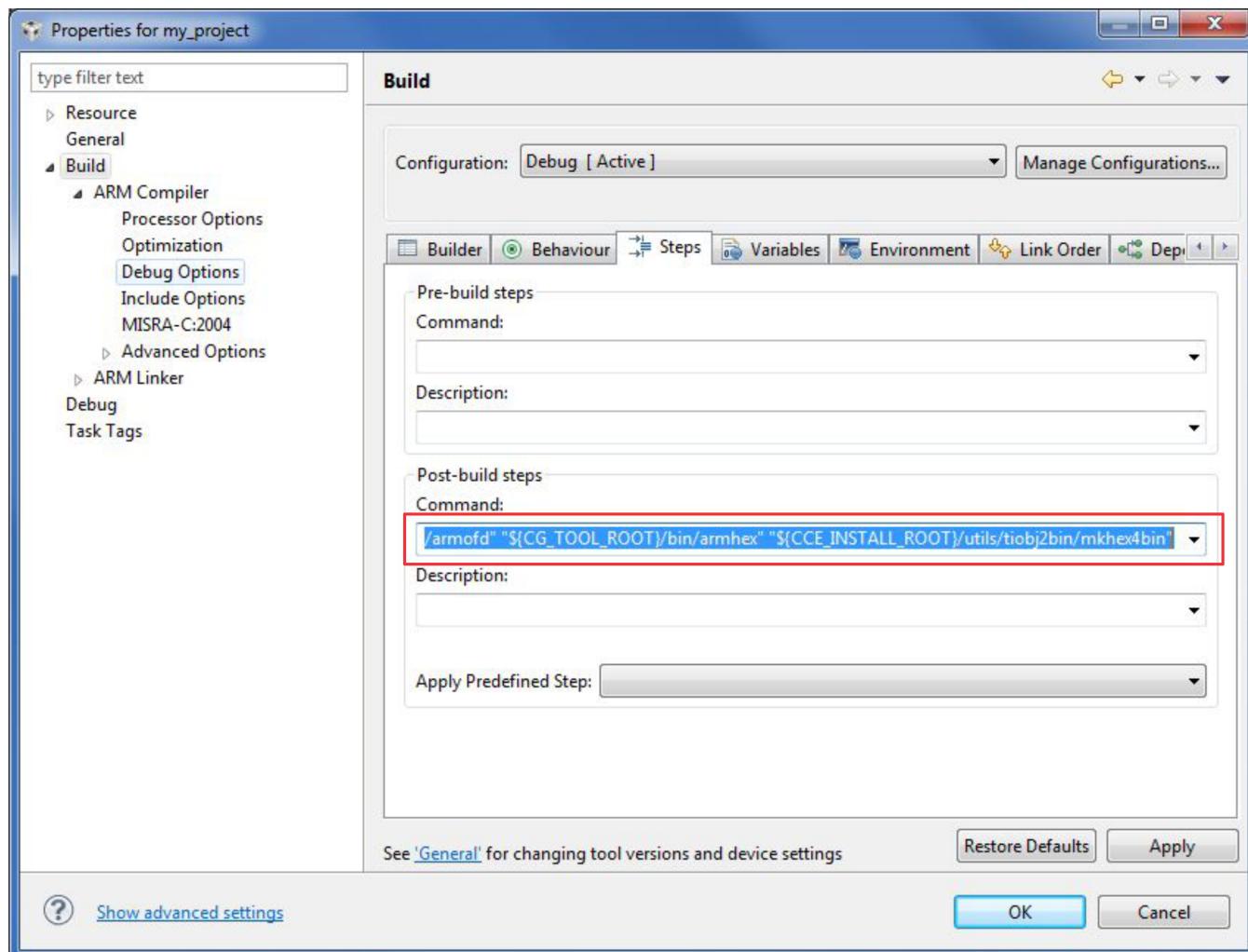


Figure 10.

If the standard C I/O functions such as printf or fread are used, the stack and heap allocations must be increased for your project as follows:

1. Right-click your project and then select *Show Build Settings...* to open the Properties dialog box.
2. Select *Basic Options* under *ARM Linker* in the pane on the left side of the dialog box.
3. Enter **0x800** for both the system stack and the heap.

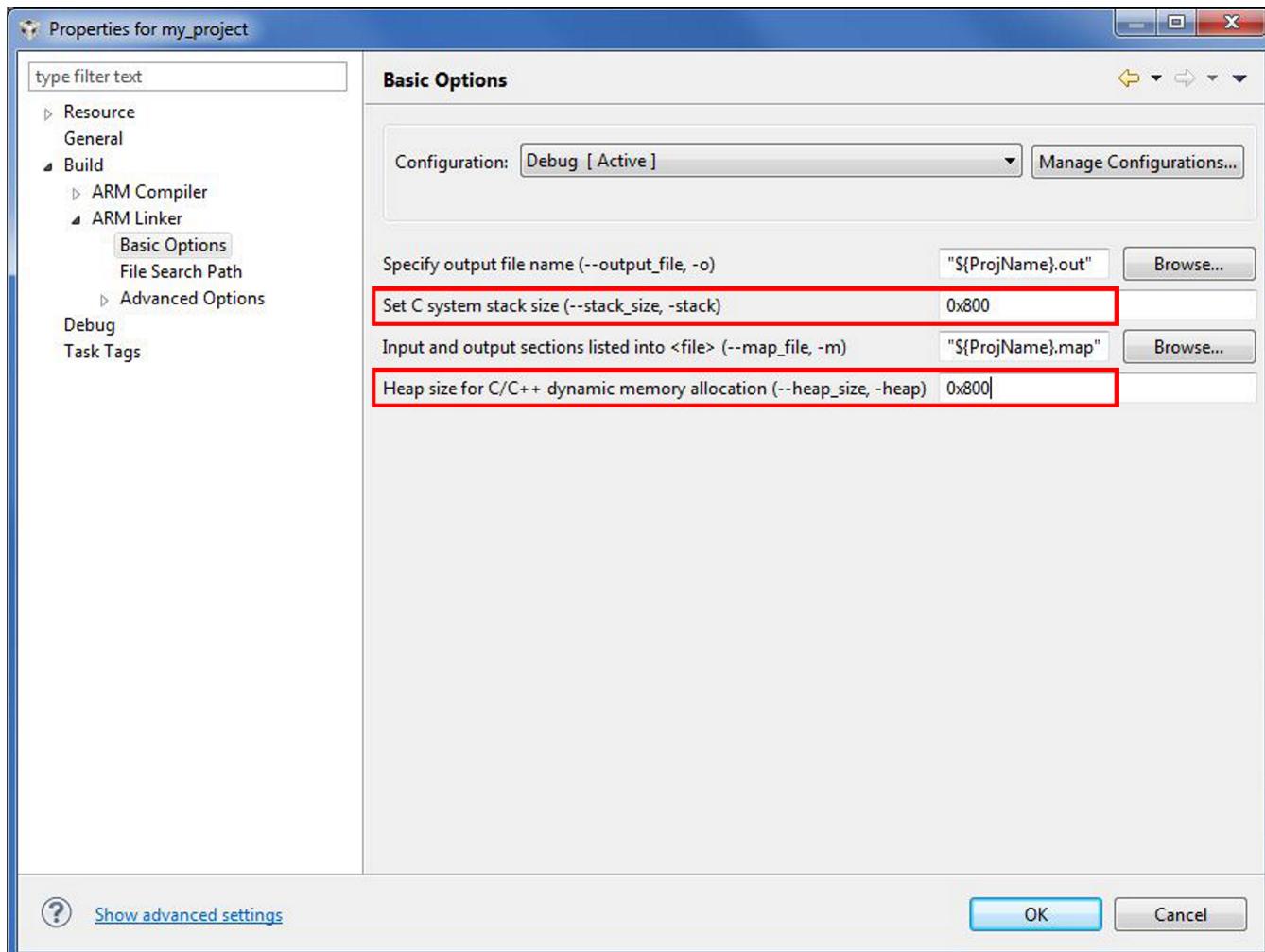


Figure 11.

5 Conclusion

You have now installed the Code Composer Studio development tools and used them to build and load an example application on your Tiva development or evaluation board. You can now experiment with the debugger or start creating your own application using the example projects for reference. For more information on Code Composer Studio, go to the CCS Developer Site by clicking Help on the main menu and selecting the CCSv5 Developer Site option.

6 References

The following references are included on the Tiva C Series Development Kit Software USB flash drive and are also available for download at www.ti.com/tiva-c:

- *Tiva C Series Evaluation Kit User's Manual*
 - *TivaWare for C Series Software*, Order Number SW-TM4C (<http://www.ti.com/tool/sw-tm4c>)
 - *TivaWare Peripheral Driver Library User's Guide*, Document Number SW-DRL-UG ([SPMU298](#))
- In addition, the following website may be useful:
- Code Composer Studio website at www.ti.com/ccstudio

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products	Applications		
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com	TI E2E Community	
OMAP Applications Processors	www.ti.com/omap	e2e.ti.com	
Wireless Connectivity	www.ti.com/wirelessconnectivity		