

A Step-by-Step Installation and Start-up Guide for CCS5 and the Biricha CSL/DPWR Libraries

Biricha Digital

1. Abstract

This document is a step by step guide for installing and setting up TI's C2000 development tools. It assumes no previous knowledge and it guides the reader from installation, to setting up various options to creating both legacy and new projects.

It is ideal for engineers who have little experience of embedded systems programming and who need to get up to speed with TI's MCUs and development tools very quickly. It is also very useful for engineers with experience of CCS3.x who need to quickly migrate to CCS5.x

2. Version History

Author	Changes	Version	Date
M Hallworth	Release	V1.1	21/06/2012
M Hallworth	Included DPWR library	V1.2	25/02/2013
M Hallworth	Included GEL files	V1.3	29/04/2013
M Hallworth	Updated include path for f2802x	V1.4	07/05/2013
M Hallworth	Updated for CCS5.5	V1.5	31/10/2013

Table of Contents

A Step-by-Step Installation and Start-up Guide for CCS5 and the Biricha CSL/DPWR Libraries	1
1. Abstract	1
2. Version History	1
3. Introduction.....	2
4. Installation	3
CCS5	3
Biricha Chip Support Library (CSL) Installation	9
Biricha Digital Power Library (DPWR) Installation	13
TI's controlSUITE header files.....	17
5. Using CCS5.....	20
Target Configuration	22
Creating a New CCS5 Project.....	25
IMPORTANT NOTICE	33

3. Introduction

This document describes how to install the following software:

1. Code Composer Studio (CCS) v5.x.
2. Biricha Digital Chip Support Library (CSL)
3. Biricha Digital Digital Power Library (DPWR)
4. TI controlSUITE

CCS5.x is TI's development environment for C2000 family of MCUs. This document is based on version 5.1 and there may be small differences between the steps given here and the ones you may need to follow if you are installing a different version.

The Biricha Digital Chip Support Library (CSL) is a comprehensive set of proprietary library functions that make programming intuitive and easy. By installing the CSL you no longer need to know the names of registers or bit fields. This library is specially useful for engineers new to embedded systems programming for example analogue engineers who need to program digital power supplies.

In the following sections we will describe how you can download and install the CSL. This download version is not limited in functionality but please note that the license agreement strictly prohibits you from using it for commercial purposes unless you have purchased it through attending one of our workshops.



**BIRICHA
DIGITAL**

Hands-On Digital Power Three Day Workshop

Learn to design and implement digital power supplies with Texas Instruments' TMS320C2000™ microcontrollers

- Reduce your development time and learning curve
- Aimed at analog power supply designers and embedded systems engineers
- Morning theory followed by extensive design labs in the afternoon
- Requires no/little knowledge of programming
- Complete syllabus available on www.biricha.com
- Presented at locations worldwide

www.biricha.com/workshops

 **TEXAS
INSTRUMENTS**

The TI logo and TMS320C2000 are trademarks of Texas Instruments.

Advertisement

4. Installation

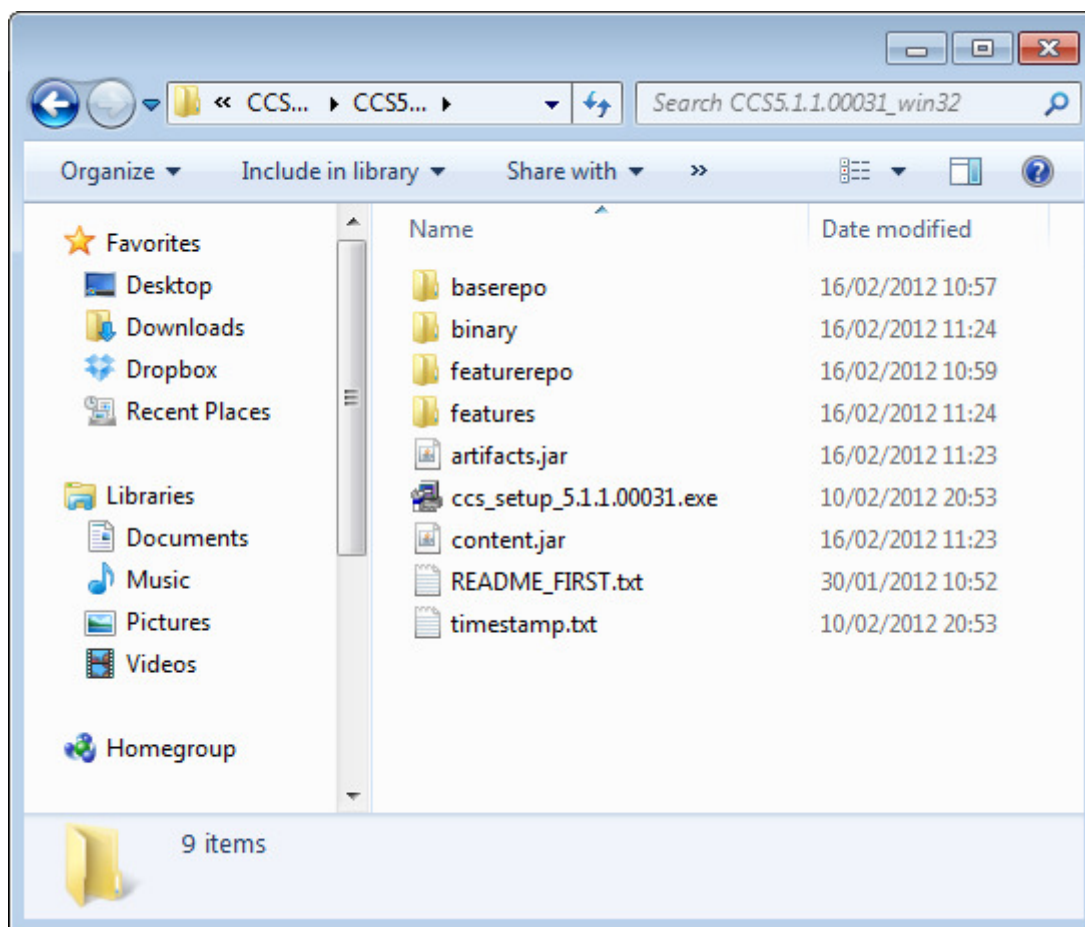
CCS5

You can download the latest version of CCS5 from the TI website using the link below:

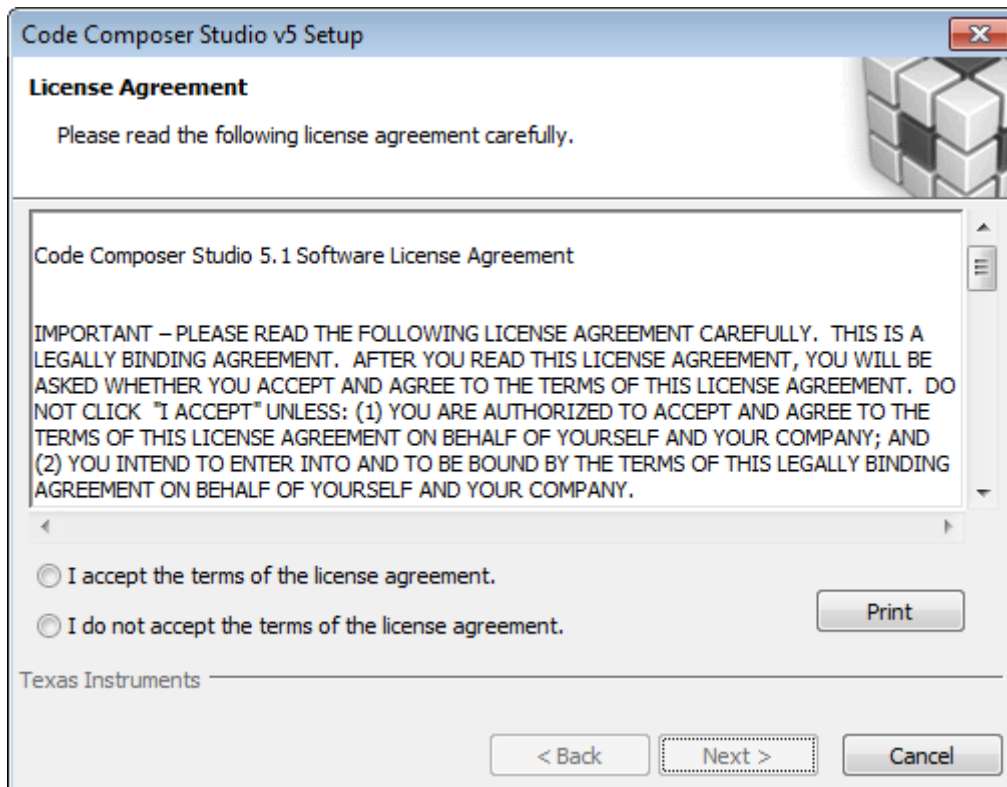
http://processors.wiki.ti.com/index.php/Download_CCS

Download or insert a CD with CCS5 on it. Extract the files and double Click on the Setup icon.

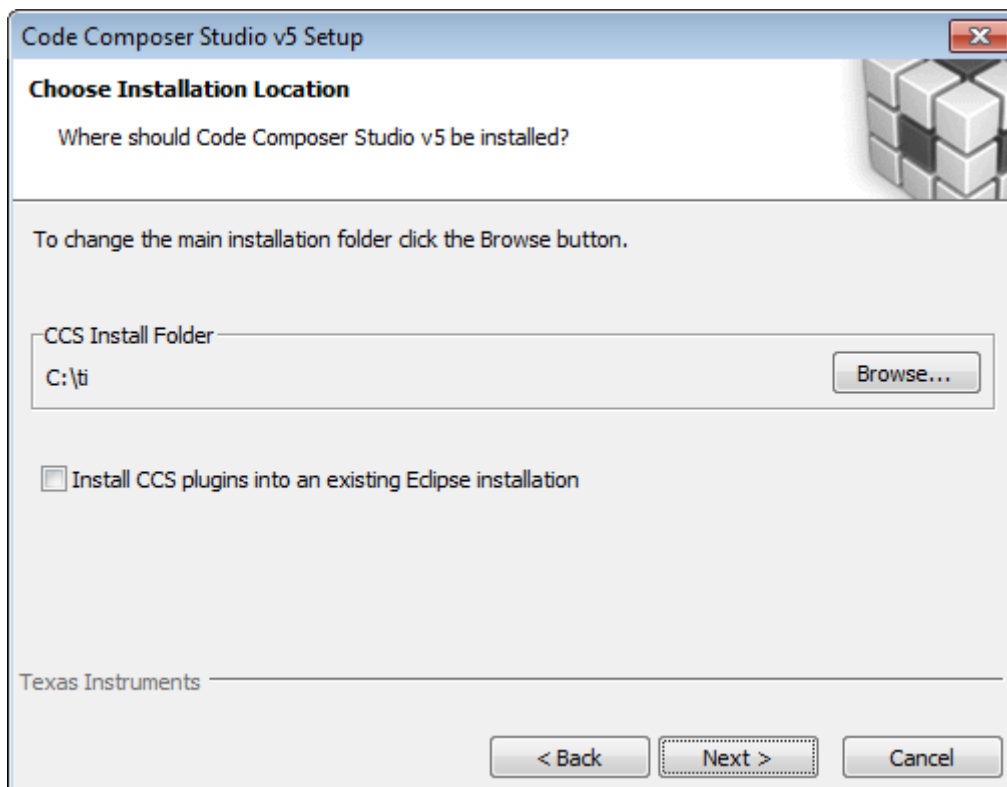
Please note: Antivirus applications have been known to cause problems with the installation. You may need to disable any sandbox/shields whilst the installation completes.



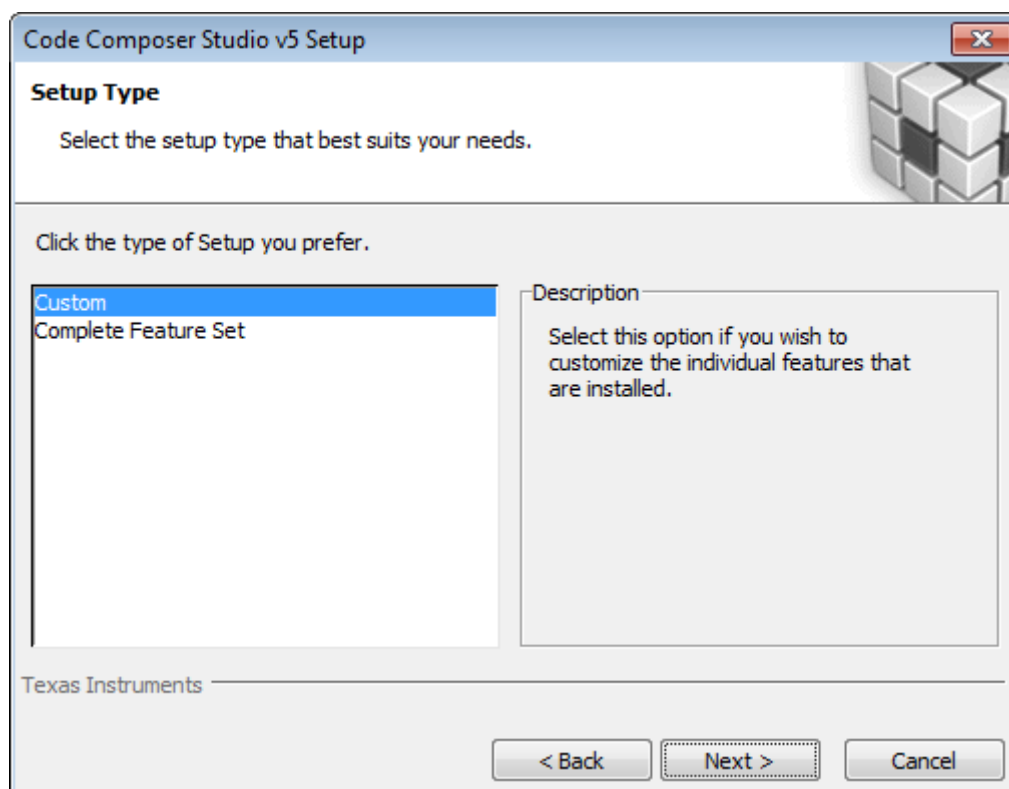
If you are using Windows Vista or above, you may see a message regarding "User Account Control". Click "OK" to continue.



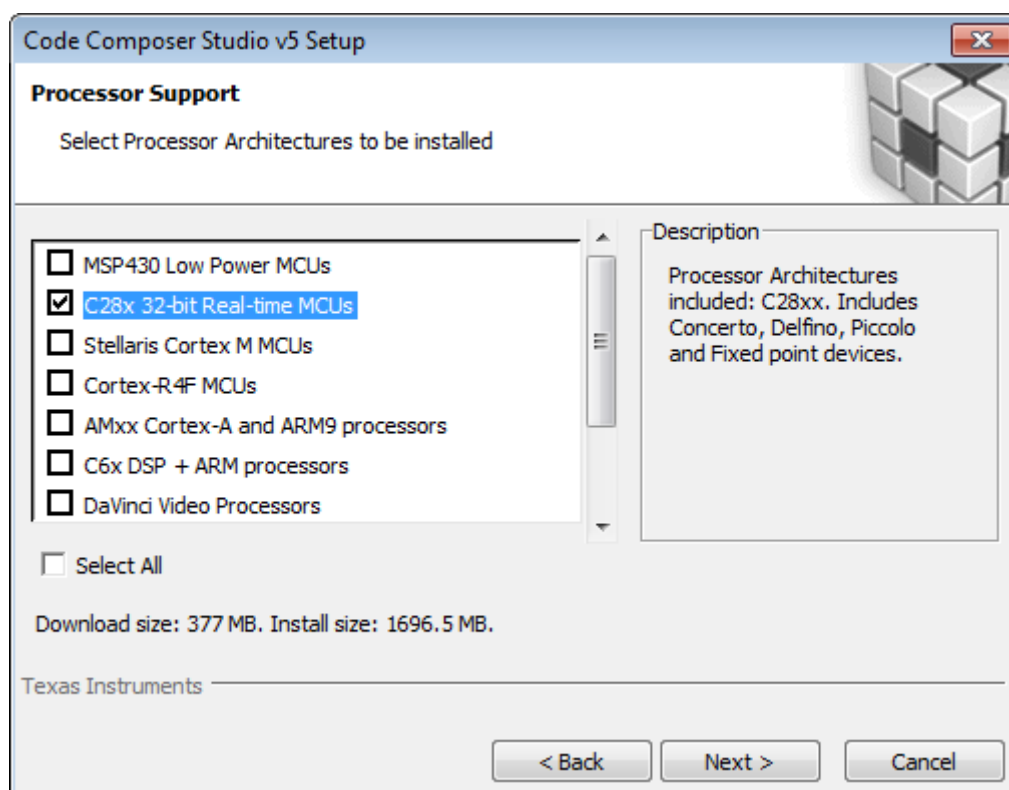
The CCS5 installation start dialogue will appear. Read the licence agreement carefully and decide whether or not you are going to accept the agreement. If you wish to continue click “I accept the License Agreement” to continue with the installation. Click “Next”.



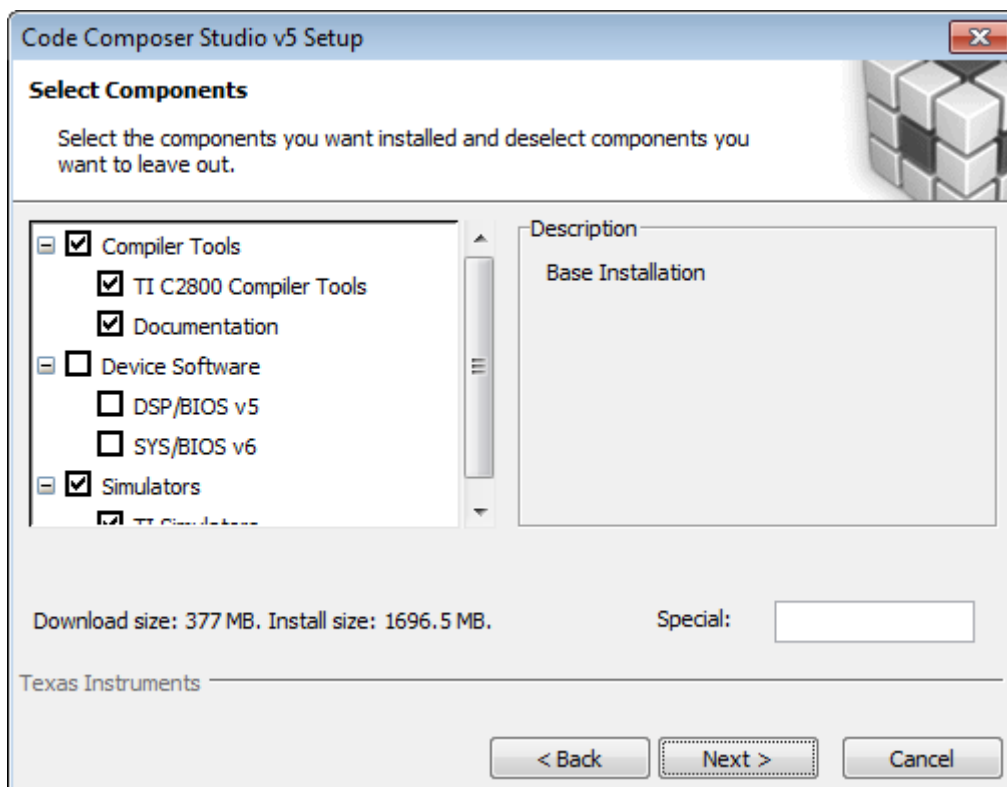
All paths are relative to C:\TI and thus we highly recommend installing to this default directory. Click “Next”.



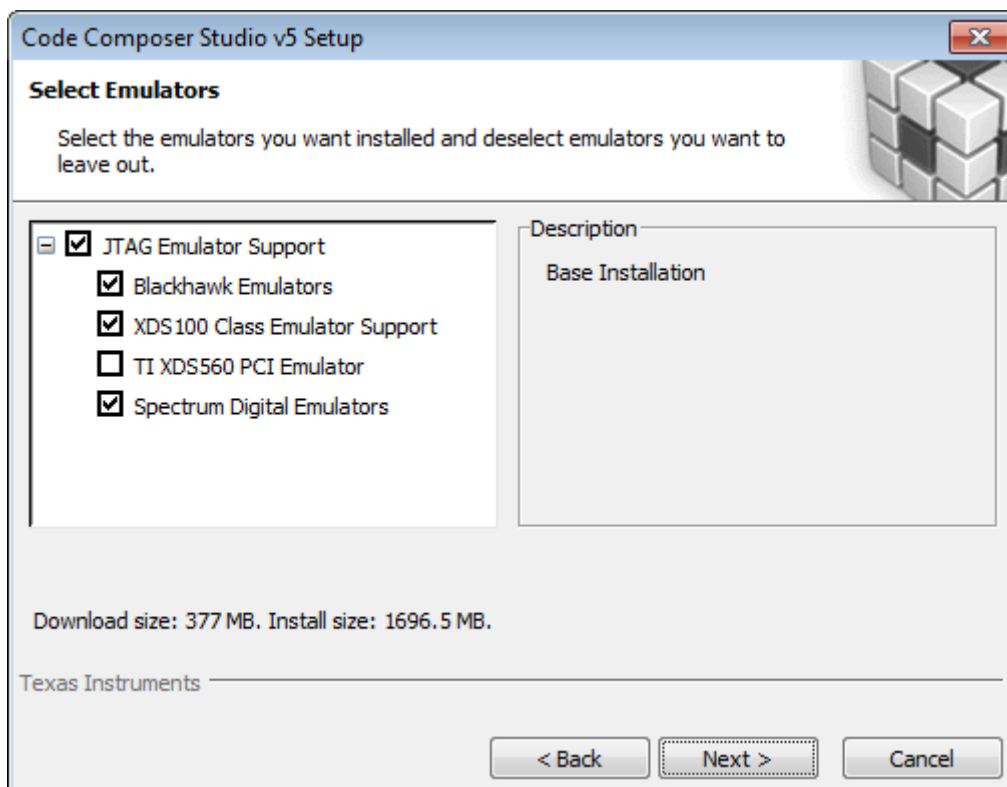
Select "Custom" and click "Next".



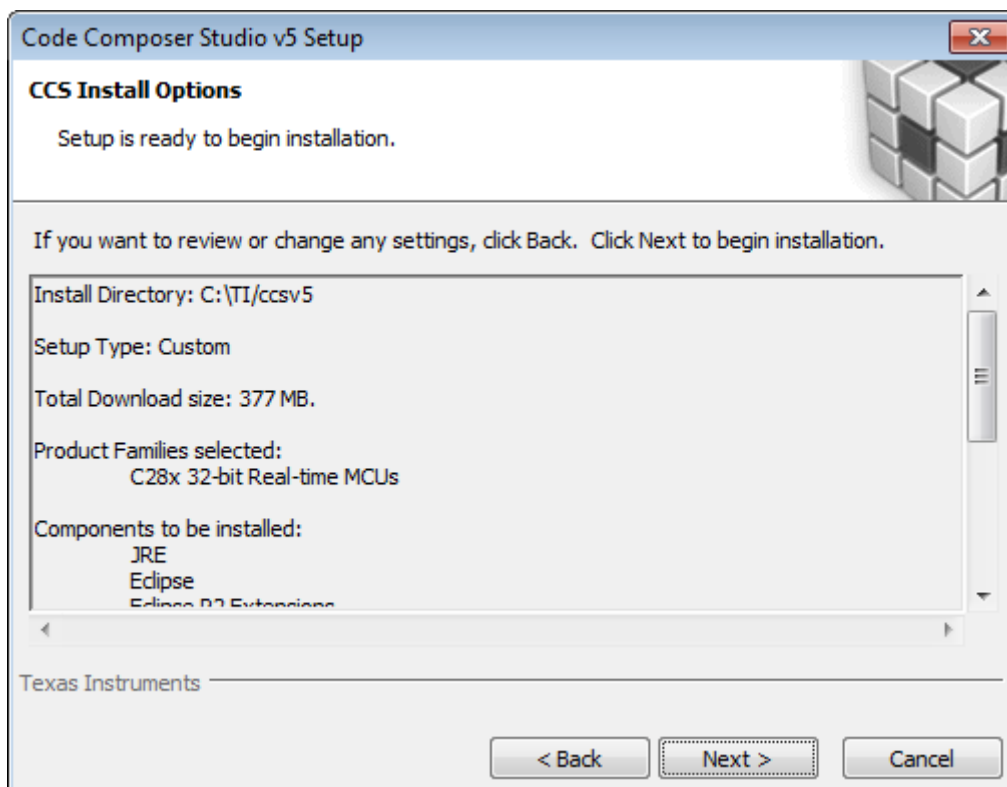
Select "C28x 32-bit Real-time MCUs" and click "Next".



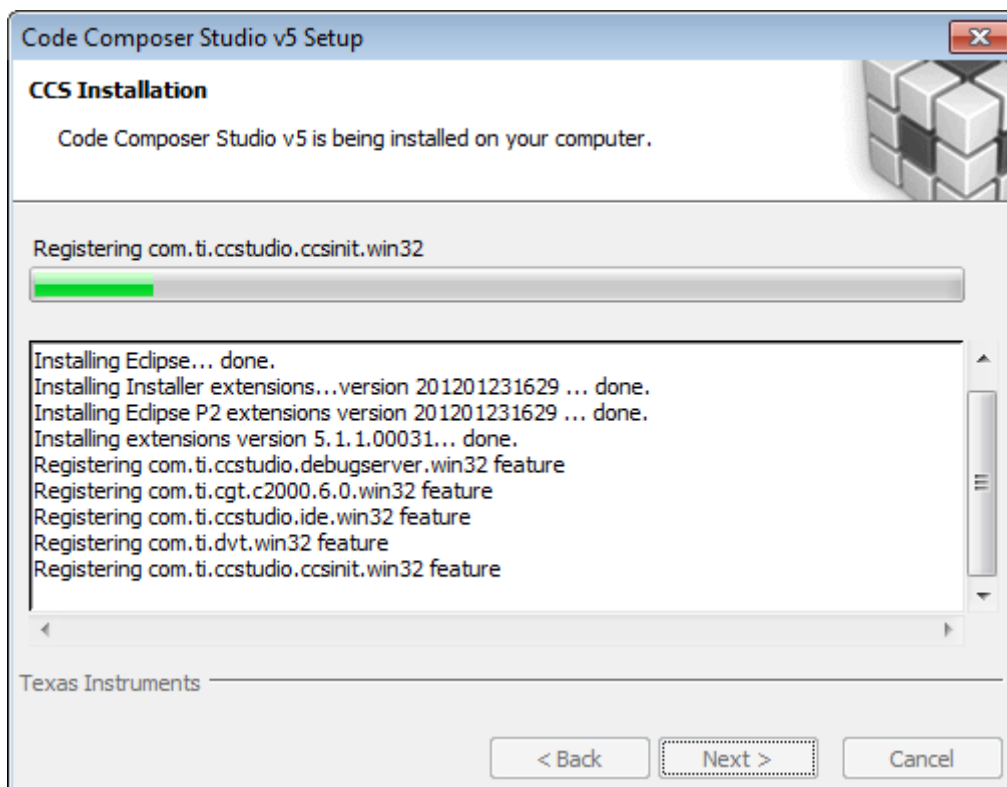
Keep the default options on this screen. Click "Next".



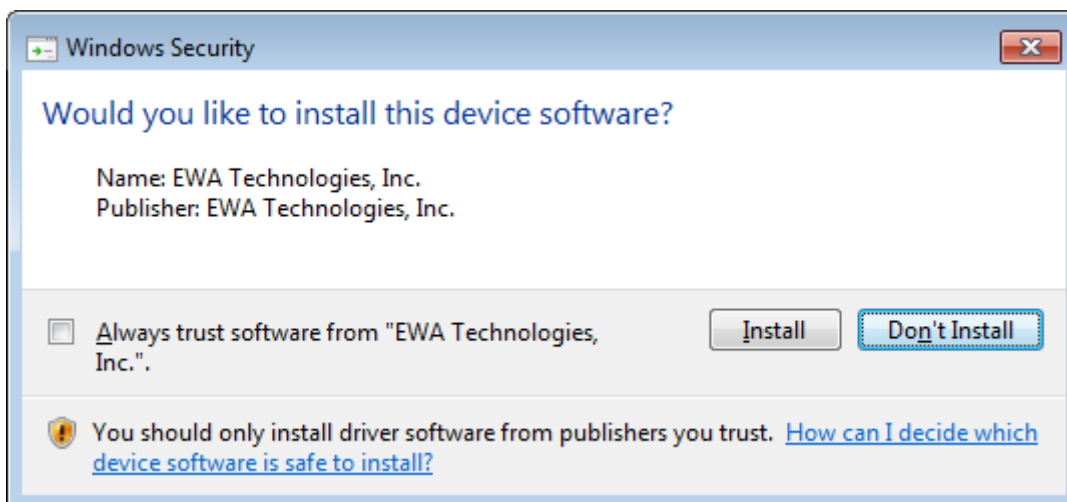
Keep the default options on this screen. Click "Next".



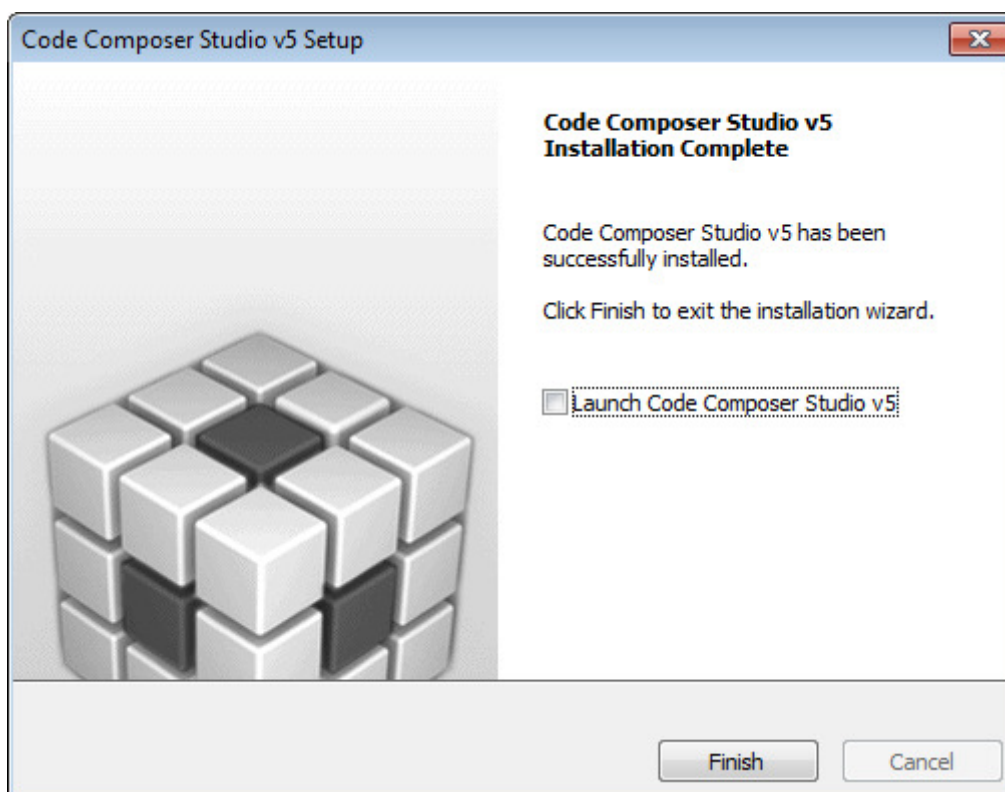
Click "Next" to begin copying the files.



The setup will install the components you have selected. This will take some time.



You may be asked for permission to install drivers several times during this setup process. This is for the JTAG emulators. Click "Install".



At the end of the installation uncheck "Launch Code Composer Studio v5" as there are other components yet to be installed. Click "Finish".

Biricha Chip Support Library (CSL) Installation

You can download the latest version of the Biricha CSL from:

http://www.biricha.com/resources/download_resource.php?id=27

You will need to create an account, if you do not already have one, in order to download the CSL.

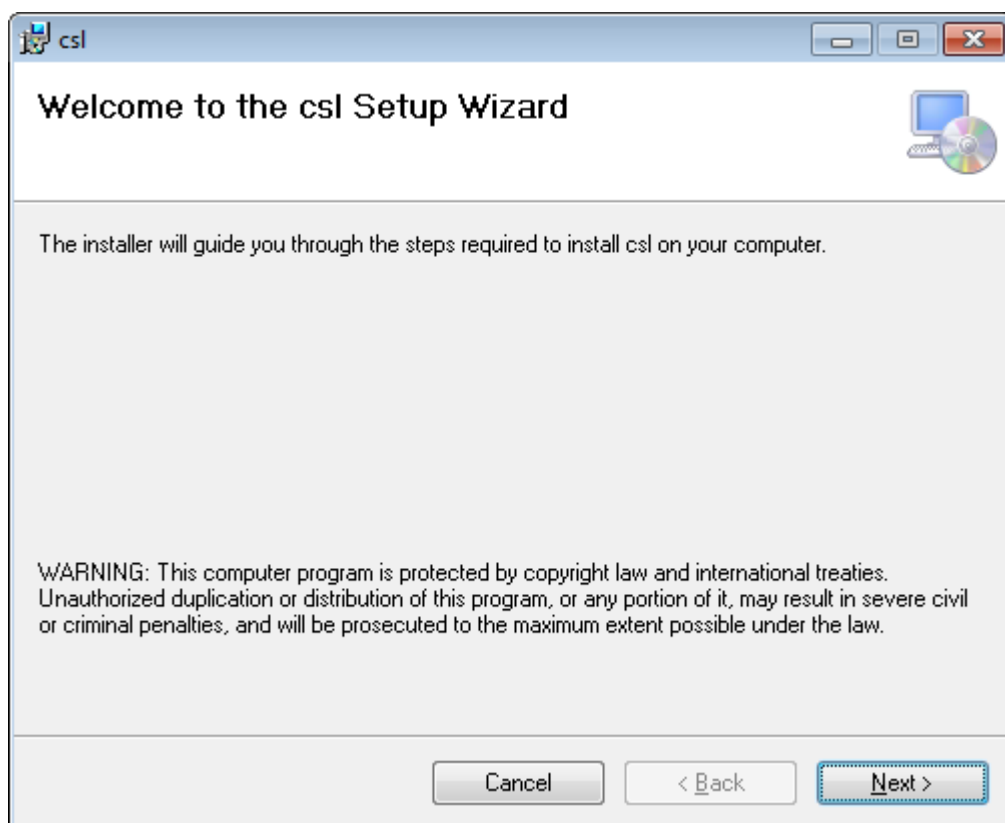
Please read the license agreement carefully. The library available on our website is limited in functionality. Also, you will not be able to use it for commercial purposes. The full and commercial version of this library can be through attending one of our workshops.

The CSL installs the following files on to your "C" drive.

- C:\TI\controlSUITE\libs\cs\latest (CSL core files)
 - \include
 - \lib
 - \doc
 - \cmd
- C:\TI\controlSUITE\libs\cs\examples (example projects)
 - \c280x
 - \c2802x
 - \c2803x
 - \c2806x
 - \c2833x

To use the library, include the header file "csl.h" from the include directory.

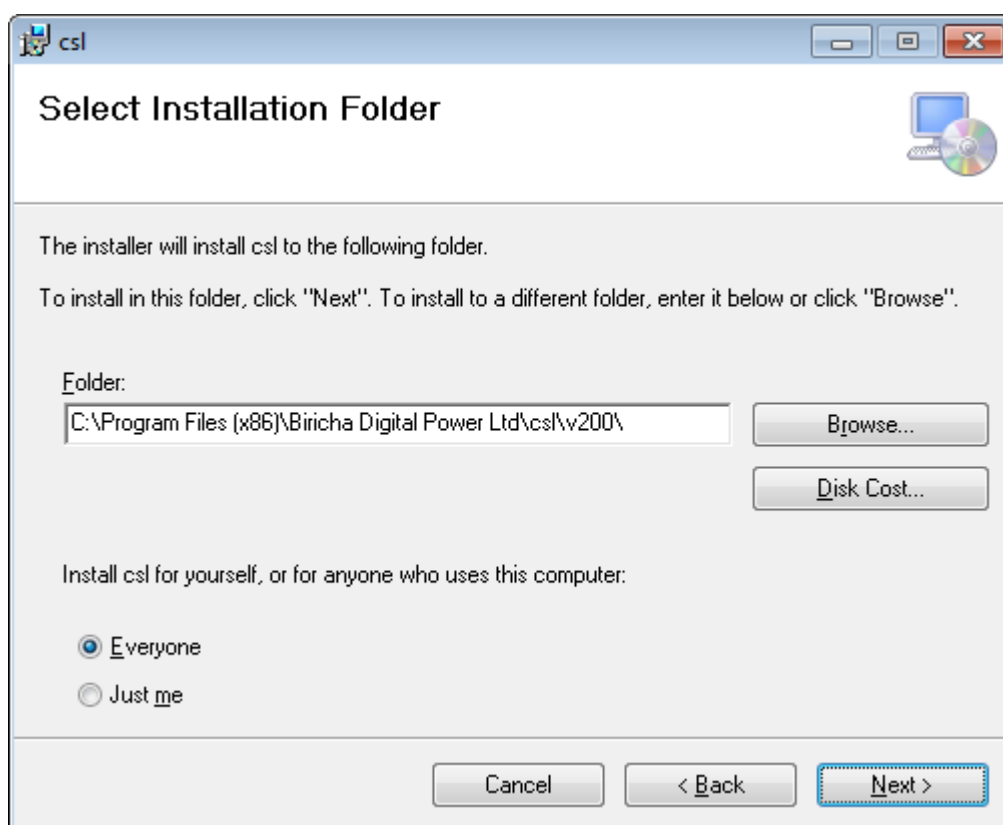
To start the installation of the CSL, double click on "BirichaCsl_vXXX_X.msi".



Click "Next" to continue.

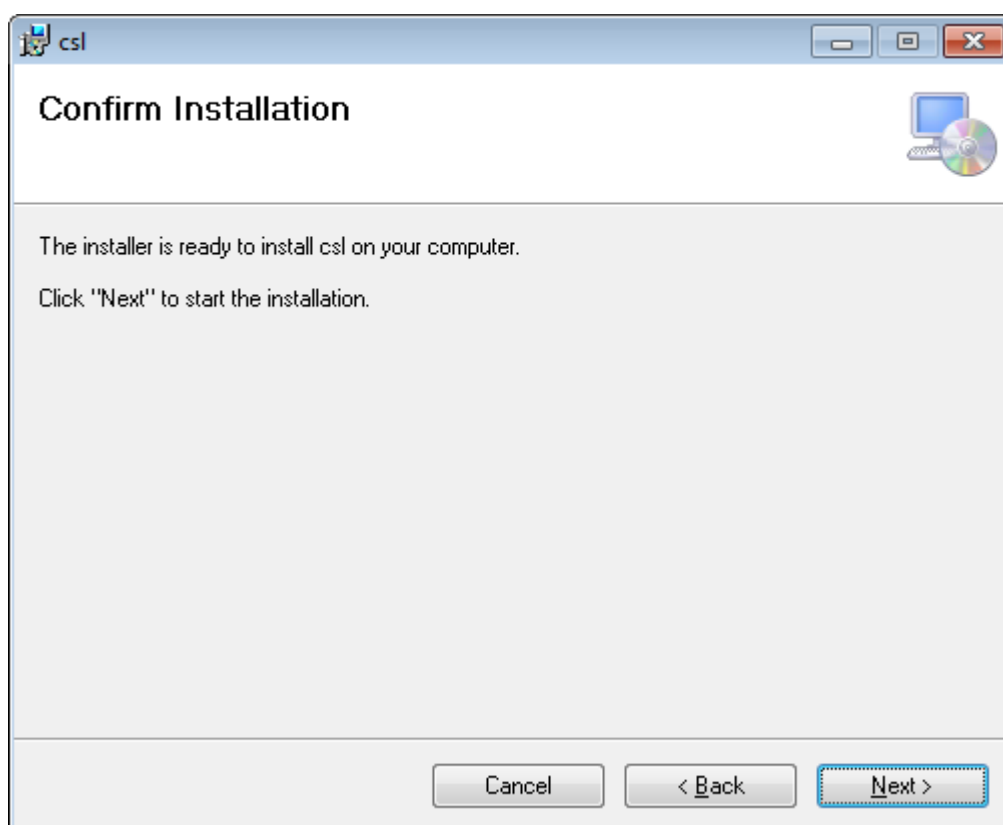


Read the license agreement and if you wish to proceed select "I Agree". Click "Next".

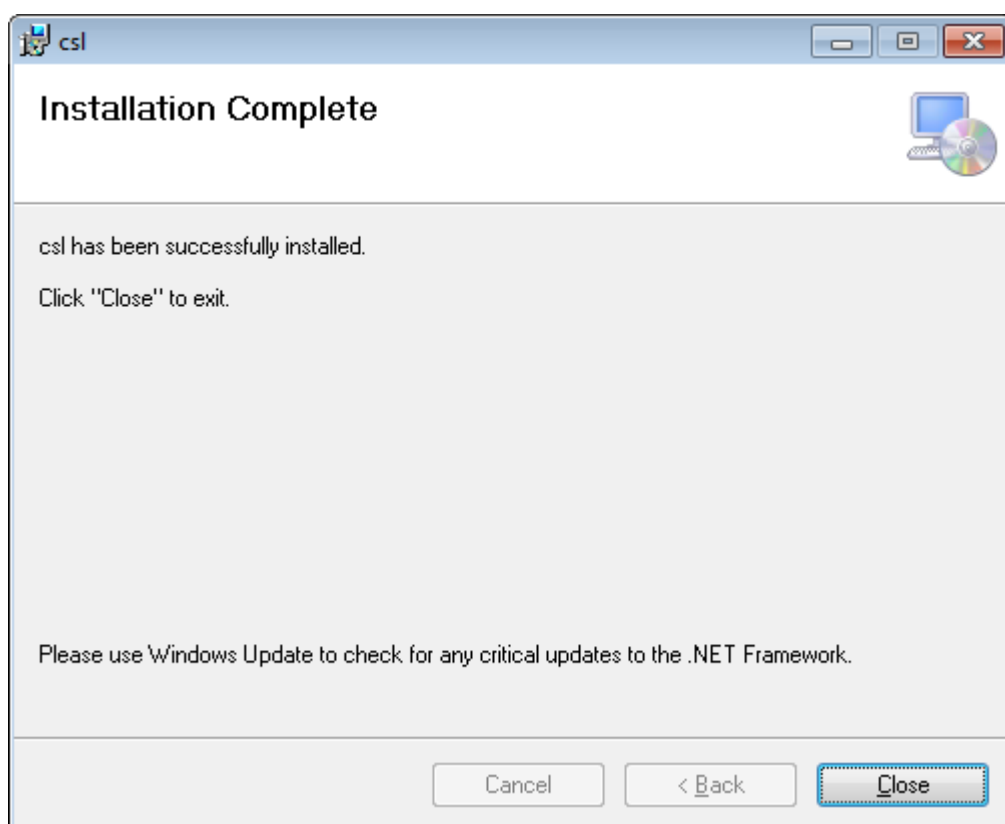
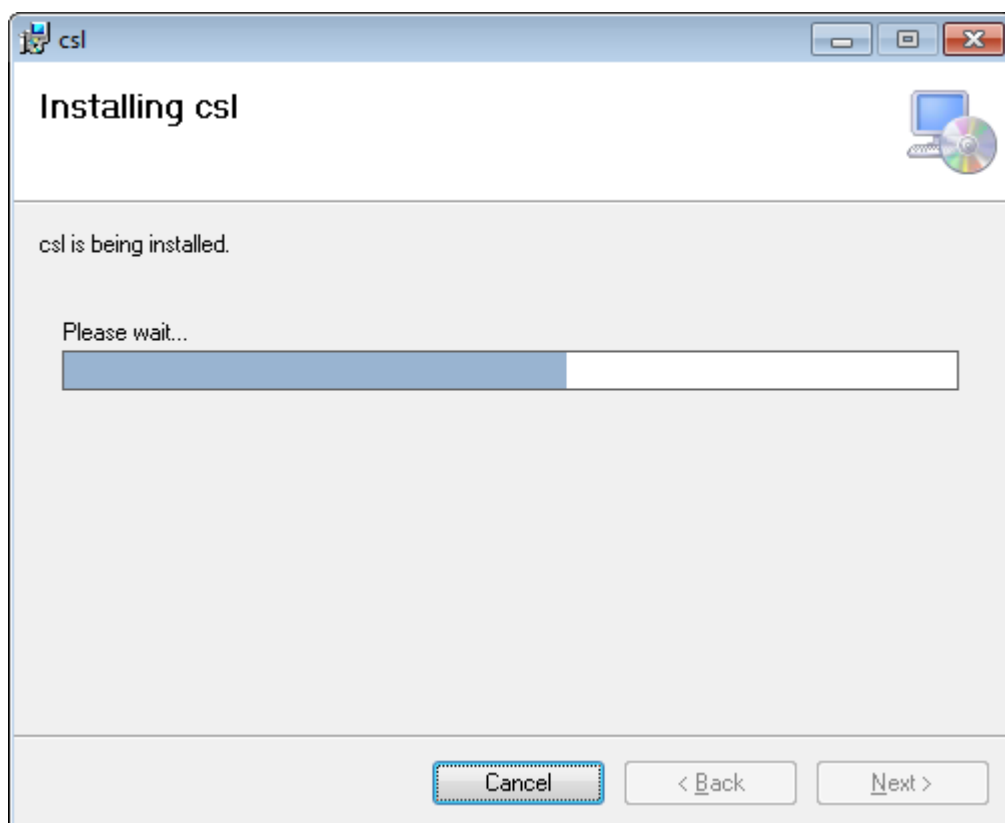


The path you select here is for temporary installation files and the location of the uninstaller. The actual paths to the CSL source files and library has been outlined at the beginning of this section.

Please note: you cannot install to a network drive at this time. Click "Next" to continue.



Click "Next".



Once the installation is complete click "Close".

All documentation can be found under: Start → Programs → Biricha Digital Power

Biricha Digital Power Library (DPWR) Installation

You can download the latest version of the Biricha DPWR from:

http://www.biricha.com/resources/download_resource.php?id=31

You will need to create an account, if you do not already have one, in order to download the DPWR.

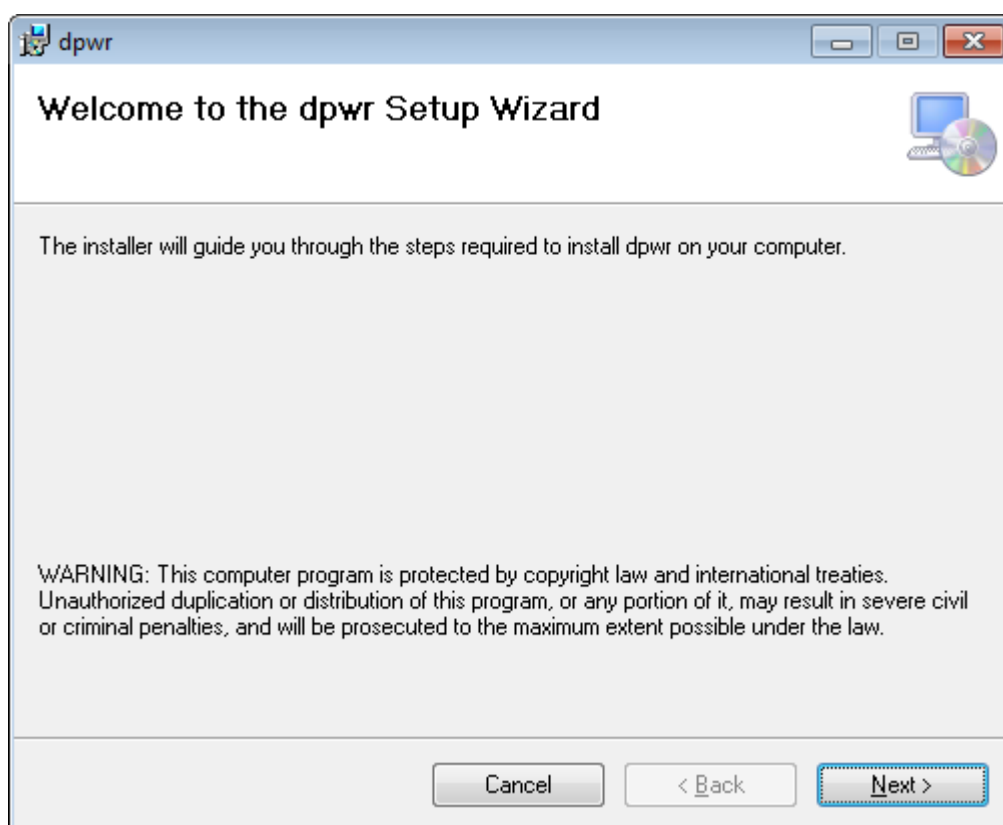
Please read the license agreement carefully. Although this library is not limited in functionality, you will not be able to use it for commercial purposes unless you purchased it through attending one of our workshops.

The DPWR installs the following files on to your "C" drive.

- C:\TI\controlSUITE\libs\dpwr\latest (DPWR core files)
 - \include
 - \lib
 - \doc

To use the library, include the header file "dpwr.h" from the include directory.

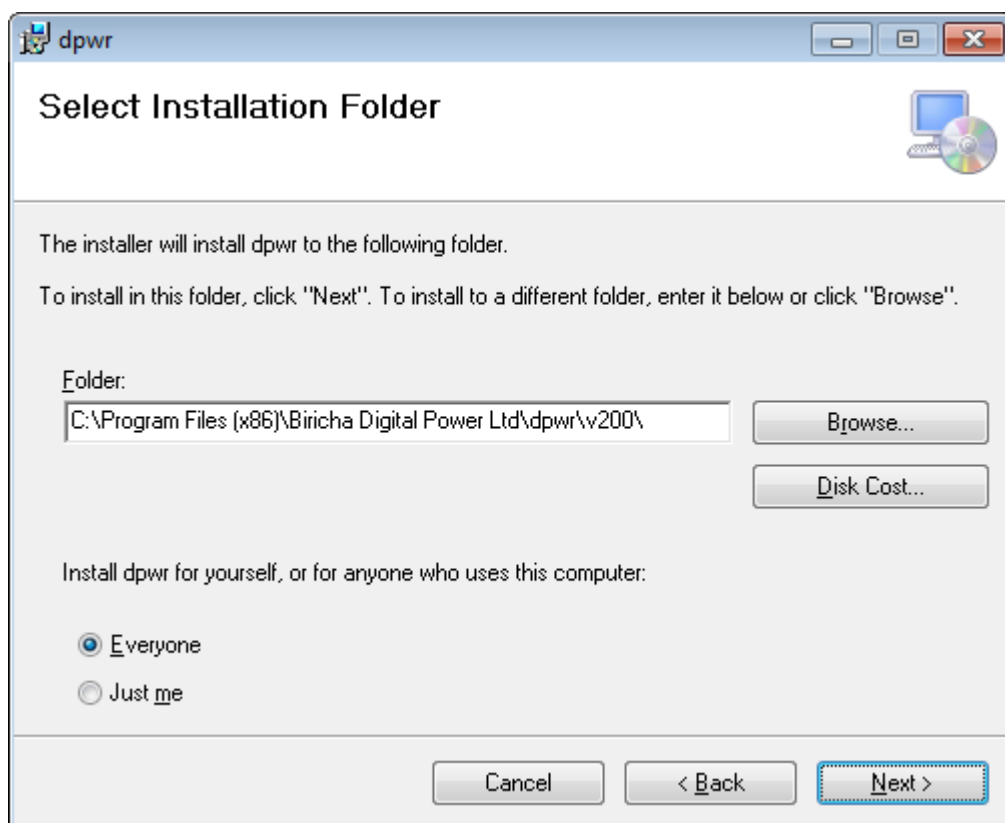
To start the installation of the DPWR, double click on "BirichaDwpr_vXXX_X.msi".



Click "Next" to continue.

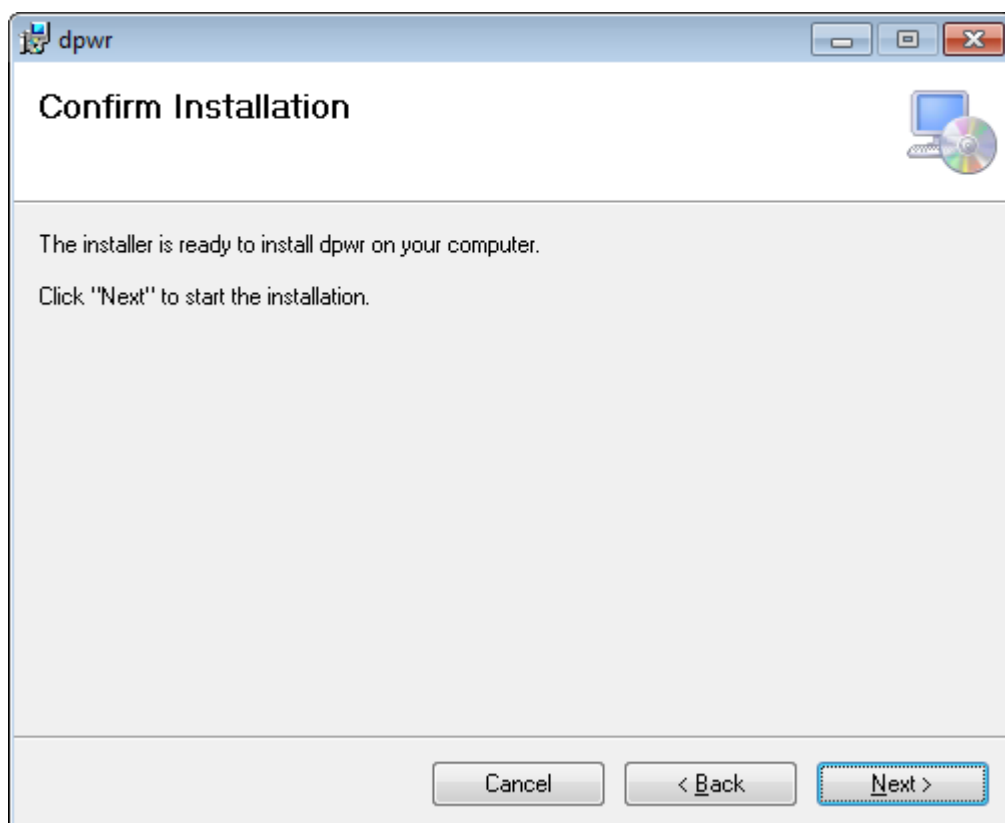


Read the license agreement and if you wish to proceed select "I Agree". Click "Next".

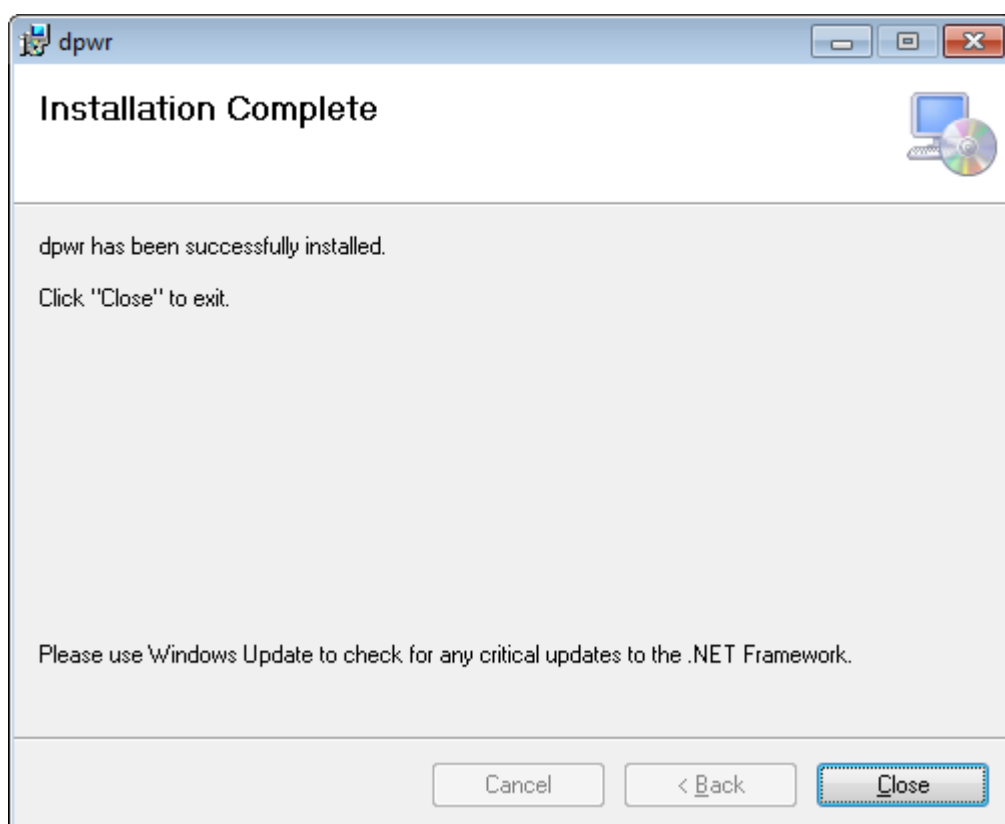
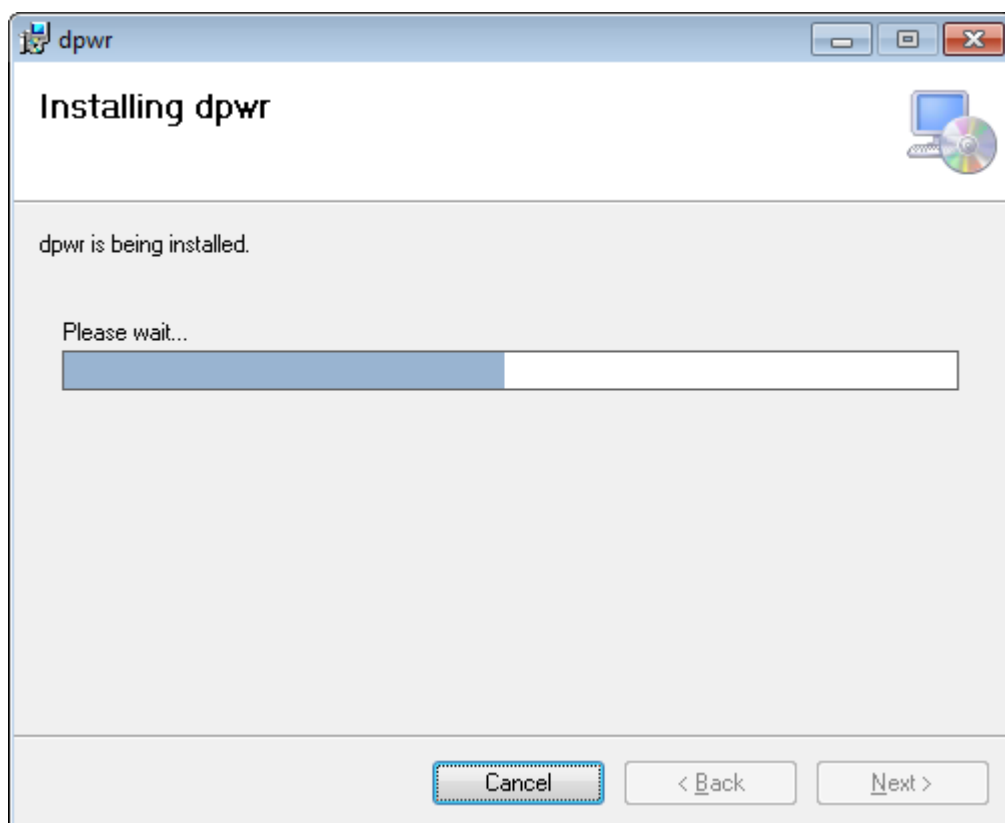


The path you select here is for temporary installation files and the location of the uninstaller. The actual paths to the DPWR source files and library has been outlined at the beginning of this section.

Please note: you cannot install to a network drive at this time. Click "Next" to continue.



Click "Next".



Once the installation is complete click "Close".

TI's controlSUITE header files

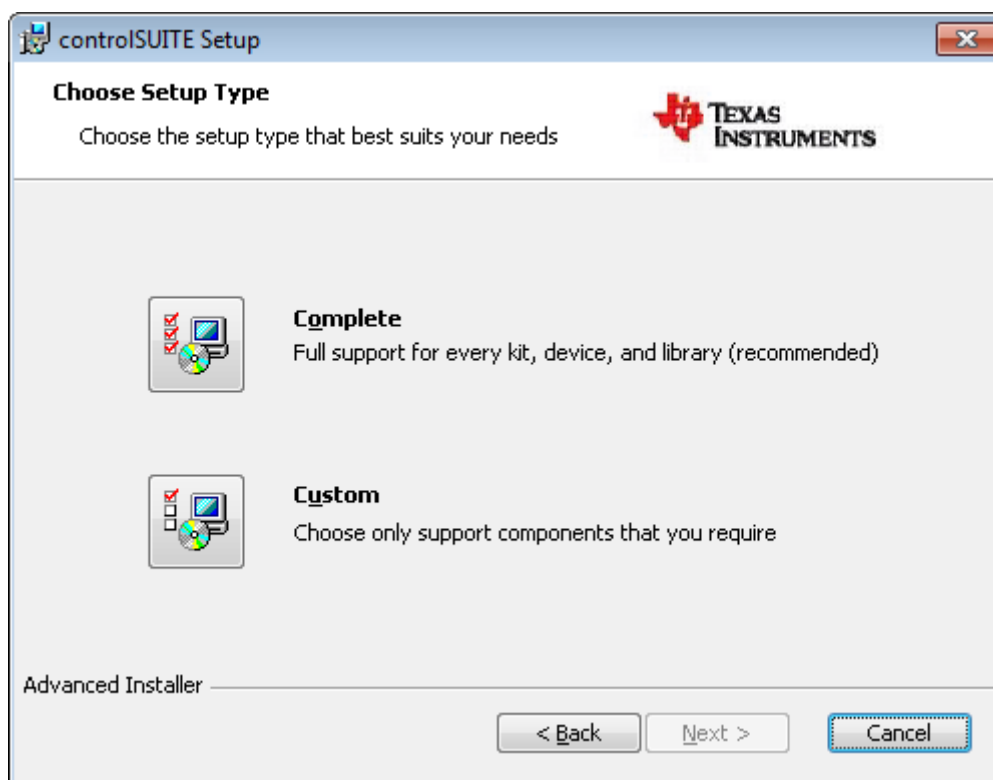
You must install controlSUITE to use the CSL. TI's controlSUITE can be downloaded from <http://www.ti.com/tool/controlsuite>. Download the "Web (EXE) Installer". This will require your to sign in to your TI account (free to create).



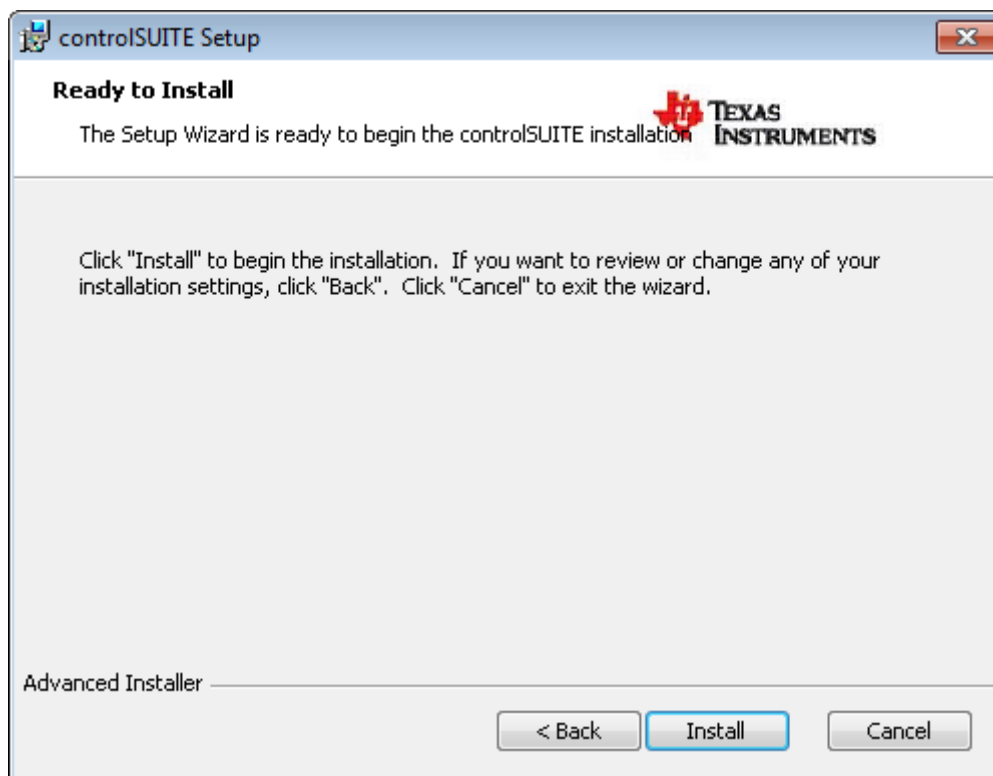
Click "Next" to continue.



Read the License Agreement carefully. If you agree, proceed by clicking "Next".



Click "Complete" to include all device options (recommended).



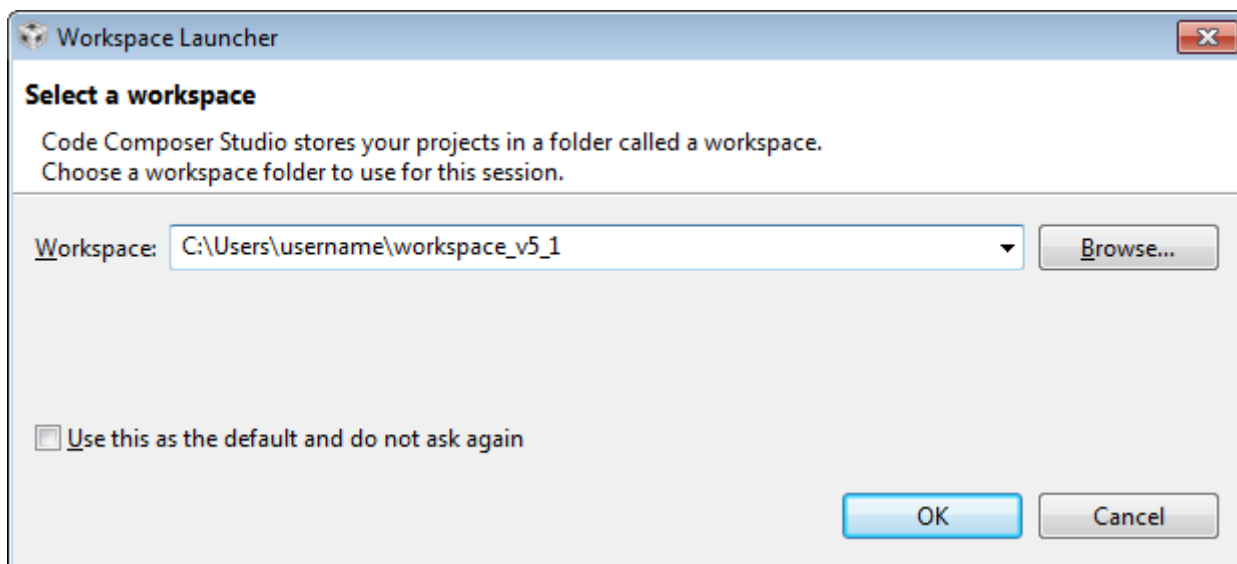
Click "Install". The installation process will take a few minutes as the required data will be downloaded during installation.



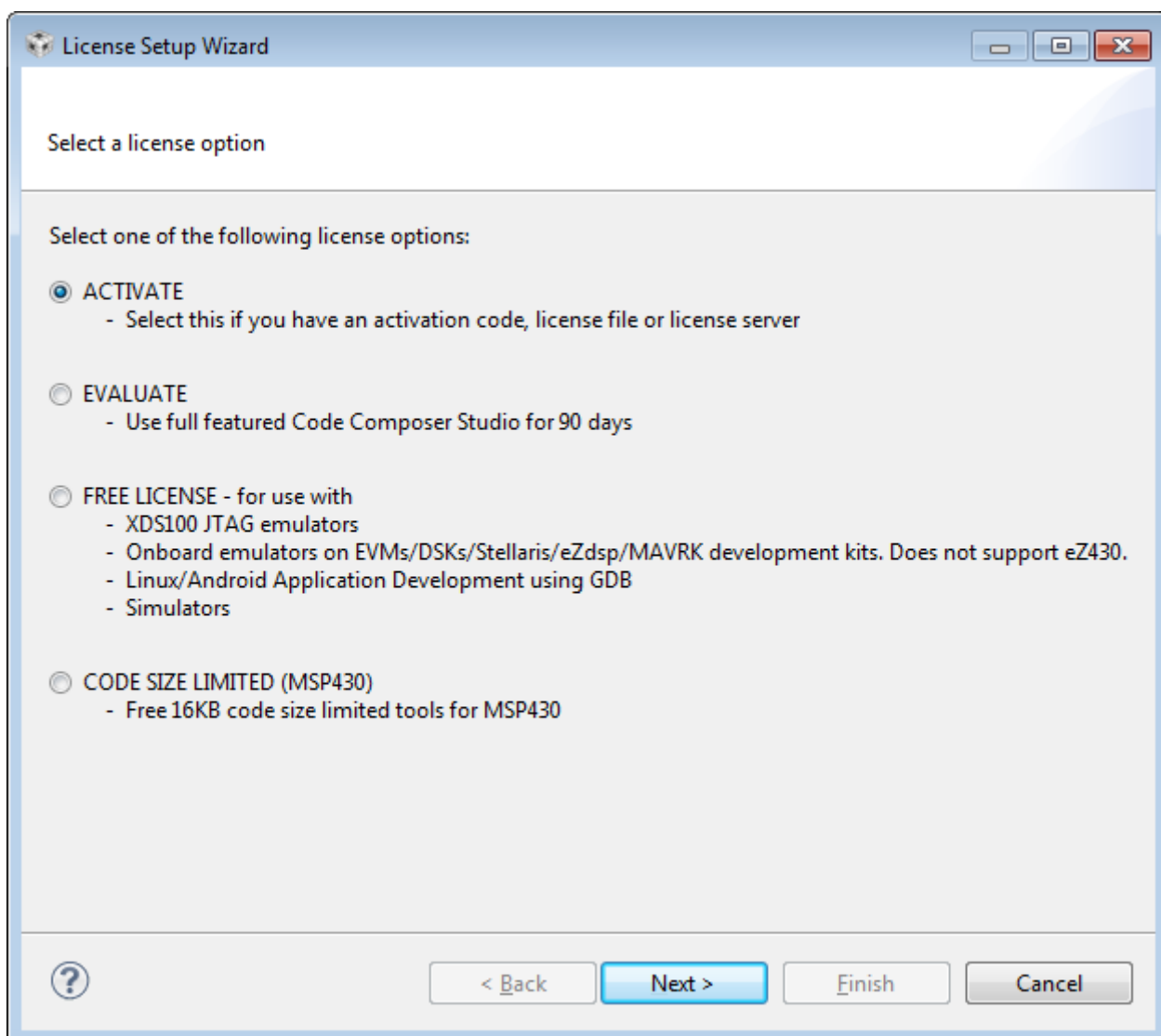
Click "Finish" to close the installer.

5. Using CCS5

Load Code Composer Studio from your start menu: Texas Instruments > Code Composer Studio v5.

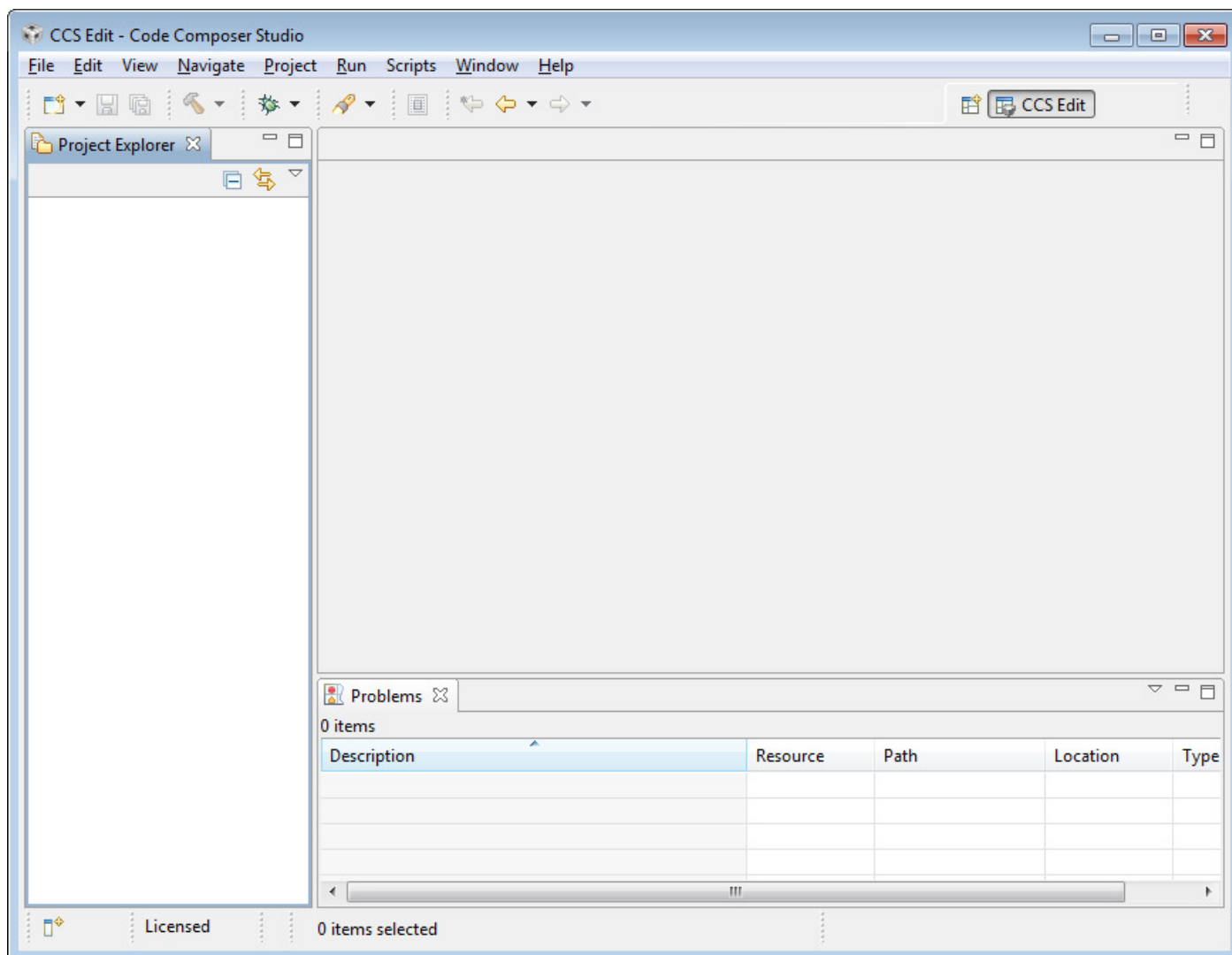


This opens up a dialog box that prompts you for a workspace. You can use the default value. Your projects do not have to be saved within your workspace folder.



The next screen will prompt you to activate the software. Users who have attended a Biricha Digital multi-day workshop will have been provided with a license free of charge - worth over \$500. To activate your license please visit <http://www.ti.com/activatesoftware>

For other users, you are able to use a free limited license with XDS100 JTAG emulators.



After activation, the main window of CCS5 will appear.

Target Configuration

We need to set up a target configuration for our platform. Select "View -> Target Configurations".

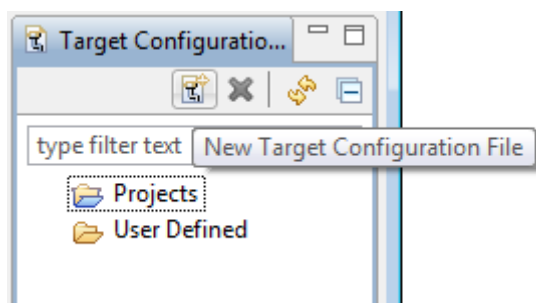
In this example we will set up a target configuration for a C2803x using a TI docking station. This uses the XDS100 emulation.

A quick note on XDS100: XDS100 is a JTAG emulator. This is simply the means by which the development tool, such as CCS5, communicates with the MCU. There are several different types of emulators the purpose of which is the same: they allow the development tools to talk to the processor.

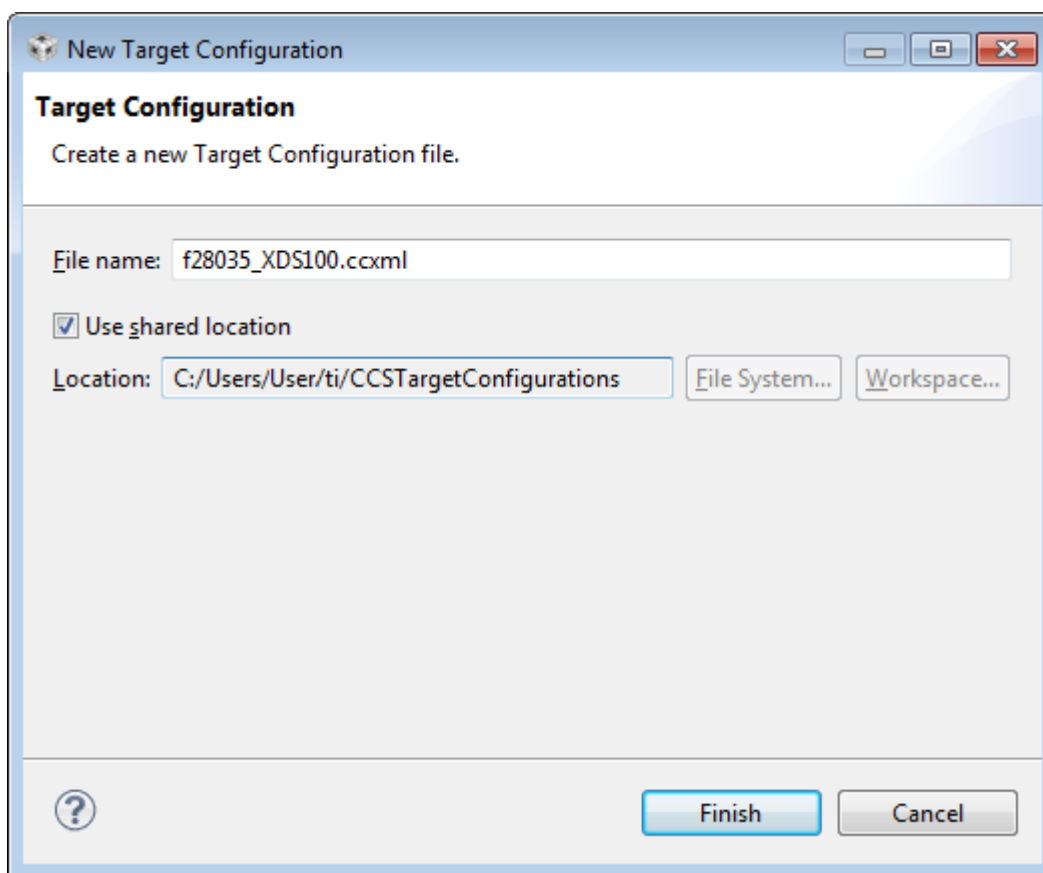
The one that we use here is called XDS100 and is the one that is most commonly found on TI's hardware development kits (but not always). XDS100 does not require an extra connector as it allows CCS5 to communicate with the device directly via the USB cable.

Other propriety emulators can be bought (e.g. Blackhawk) which can then be connected to the development hardware via a JTAG connector. You will usually find a 14 pin JTAG connector on most of TI's development hardware for this purpose.

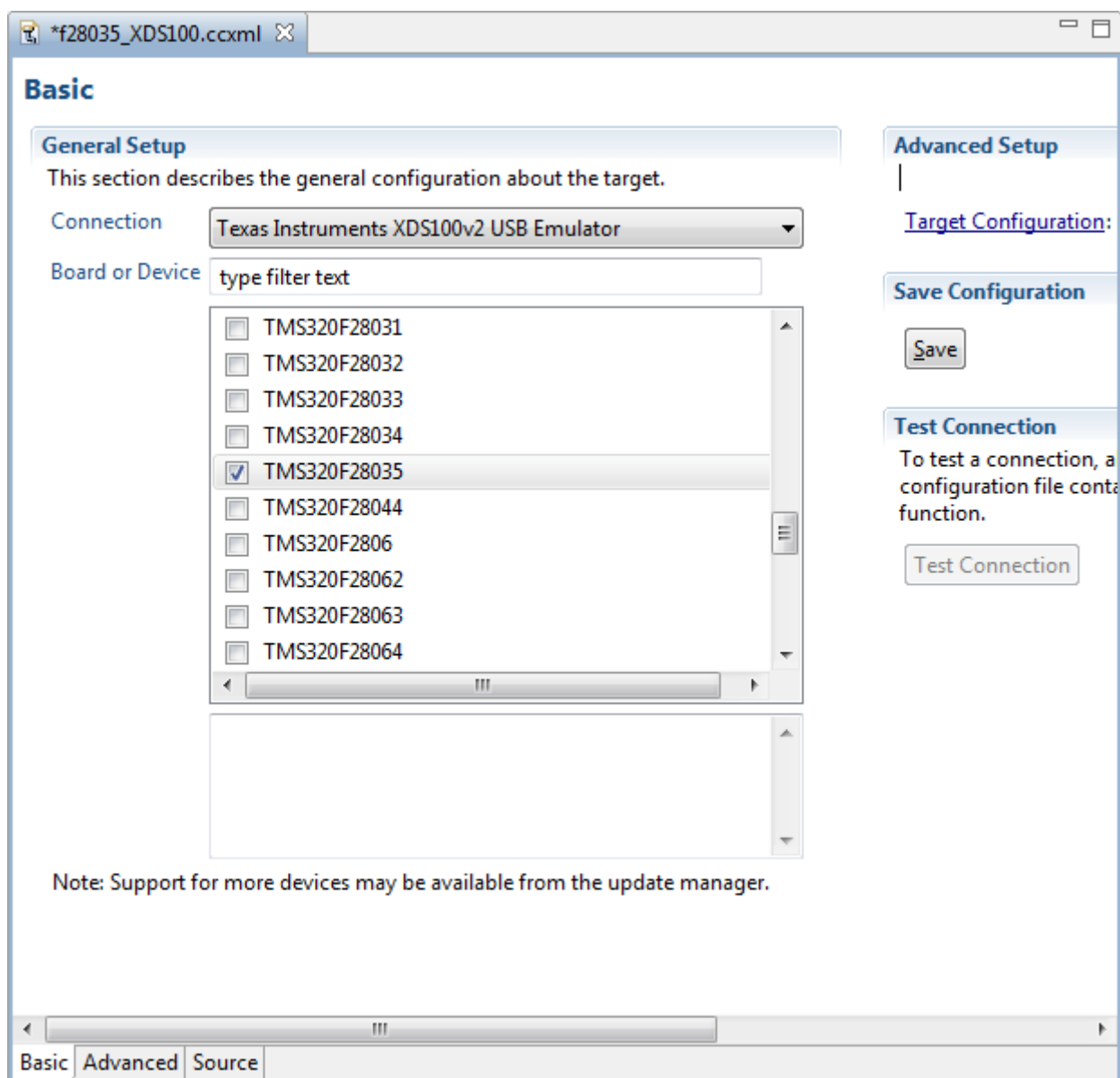
In this document we will use an F28035 MCU card with an XDS100 JTAG emulator.



Click on "New Target Configuration File".



Give the file a descriptive name such as "f28035_XDS100.ccxml". Tick "Use shared location". This allows all of your workspaces to use this target configuration. Click "Finish" when done.



A tab will have appeared in the centre of your main window. From the drop down box next to "Connect" find and select "Texas Instruments XDS100v2 USB Emulator" - assuming this is the emulator you are using.

From the now populated list, select the processor you are using. In this case "TMS320F28035".

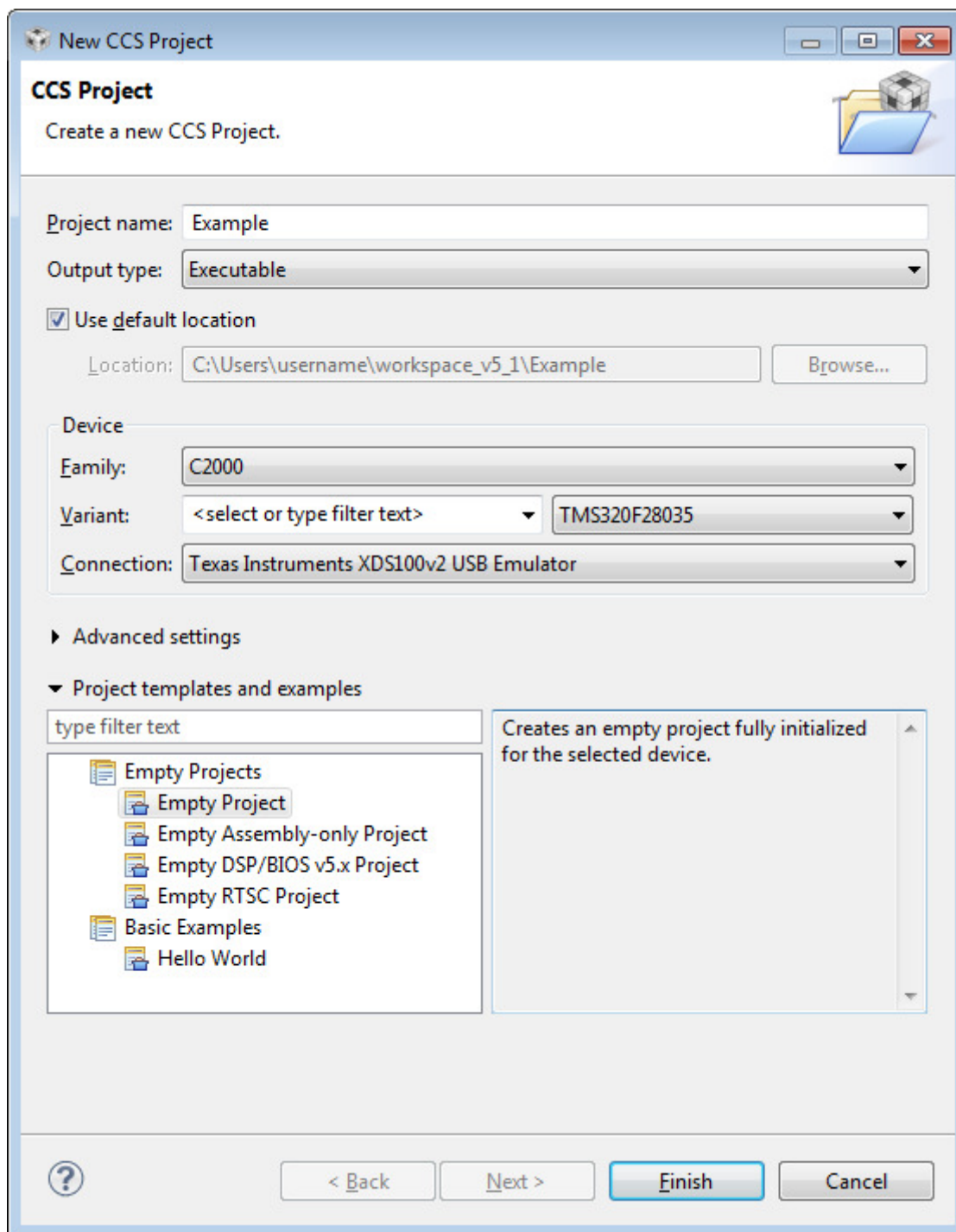
If you are using a different processor then you need to change this setting accordingly.

Finally, click "Save" to the right of this tab. Close the tab by clicking the "x" at the top next to the name of the target configuration file.

Your target configuration file will now be stored under the "User Defined" section of the "Target Configurations" tab and should be marked as "[Default]". You may now close the "Target Configurations" tab.

Creating a New CCS5 Project

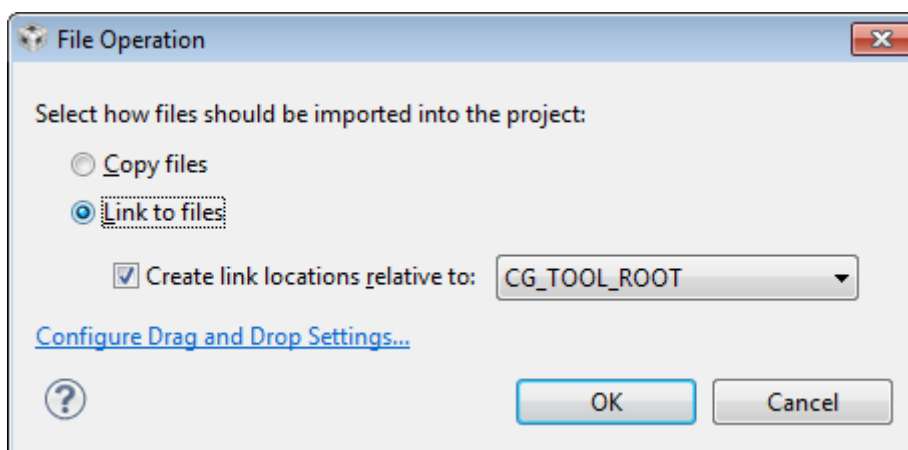
To create a new CCS5 project select "File -> New -> CCS Project".



Change the project name to "Example" and select the device type from the drop down box.

Click "Finish".

We need to add several files to every project we create. Right click on your project in the left hand "Project Explorer" tab. Select "Add Files...".



Always "Link to files" for the files described in this section. Create the link relative to "CG_TOOL_ROOT" as you may wish to move the workspace or project directory at a later date.

Add the following files depending on the processor you are using (versions correct as of date published):

C2802x	C:\TI\controlSUITE\libs\cs\latest\cmd\csl_28027_RAM_Ink.cmd
	C:\TI\controlSUITE\libs\cs\latest\lib\csl2802x_ml.lib
	C:\TI\controlSUITE\libs\dpwr\latest\lib\dpwr_ml.lib
	C:\TI\controlSUITE\device_support\2802x\v129\DSP2802x_headers\cmd\DSP2802x_Headers_nonBIOS.cmd
	C:\TI\controlSUITE\libs\math\IQmath\v160\lib\IQmath.lib
...	
C2803x	C:\TI\controlSUITE\libs\cs\latest\cmd\csl_28035_RAM_Ink.cmd
	C:\TI\controlSUITE\libs\cs\latest\lib\csl2803x_ml.lib
	C:\TI\controlSUITE\libs\dpwr\latest\lib\dpwr_ml.lib
	C:\TI\controlSUITE\device_support\2803x\v125\DSP2803x_headers\cmd\DSP2803x_Headers_nonBIOS.cmd
	C:\TI\controlSUITE\libs\math\IQmath\v160\lib\IQmath.lib
...	

C2806x	C:\TI\controlSUITE\libs\cs\latest\cmd\csl_28069_RAM_Ink.cmd
	C:\TI\controlSUITE\libs\cs\latest\lib\csl2806x_fpu.lib
	C:\TI\controlSUITE\libs\dpwr\latest\lib\dpwr_fpu.lib
	C:\TI\controlSUITE\device_support\2806x\v130\F2806x_headers\cmd\F2806x_Headers_nonBIOS.cmd
	C:\TI\controlSUITE\device_support\2806x\v130\F2806x_common\lib\SFO_TI_Build_V6b_FPU.lib
	C:\TI\controlSUITE\libs\math\IQmath\v160\lib\IQmath_fpu32.lib
...	
C2833x	C:\TI\controlSUITE\libs\cs\latest\cmd\csl_28335_RAM_Ink.cmd
	C:\TI\controlSUITE\libs\cs\latest\lib\csl2833x_fpu.lib
	C:\TI\controlSUITE\libs\dpwr\latest\lib\dpwr_fpu.lib
	C:\TI\controlSUITE\device_support\2833x\v132\DSP2833x_headers\cmd\DSP2833x_Headers_nonBIOS.cmd
	C:\TI\controlSUITE\device_support\2833x\v132\DSP2833x_common\lib\SFO_TI_Build_V5B_fpu.lib
	C:\TI\controlSUITE\libs\math\IQmath\v160\lib\IQmath_fpu32.lib

The .cmd files are linker command files and determine where your code is placed in the MCU's memory. csl28*.lib files are the library files required to use the CSL functions.

IMPORTANT: By default, CCS5 includes a standard .cmd file for use with controlSUITE in the project when it is first created. This must be removed as we have now included the CSL version. Please remove: "280X_RAM_Ink.cmd" from the project by right clicking on it and selecting "Delete".

We now need to set up include paths for our header files. To change the include paths right click on the Project and select "Properties".

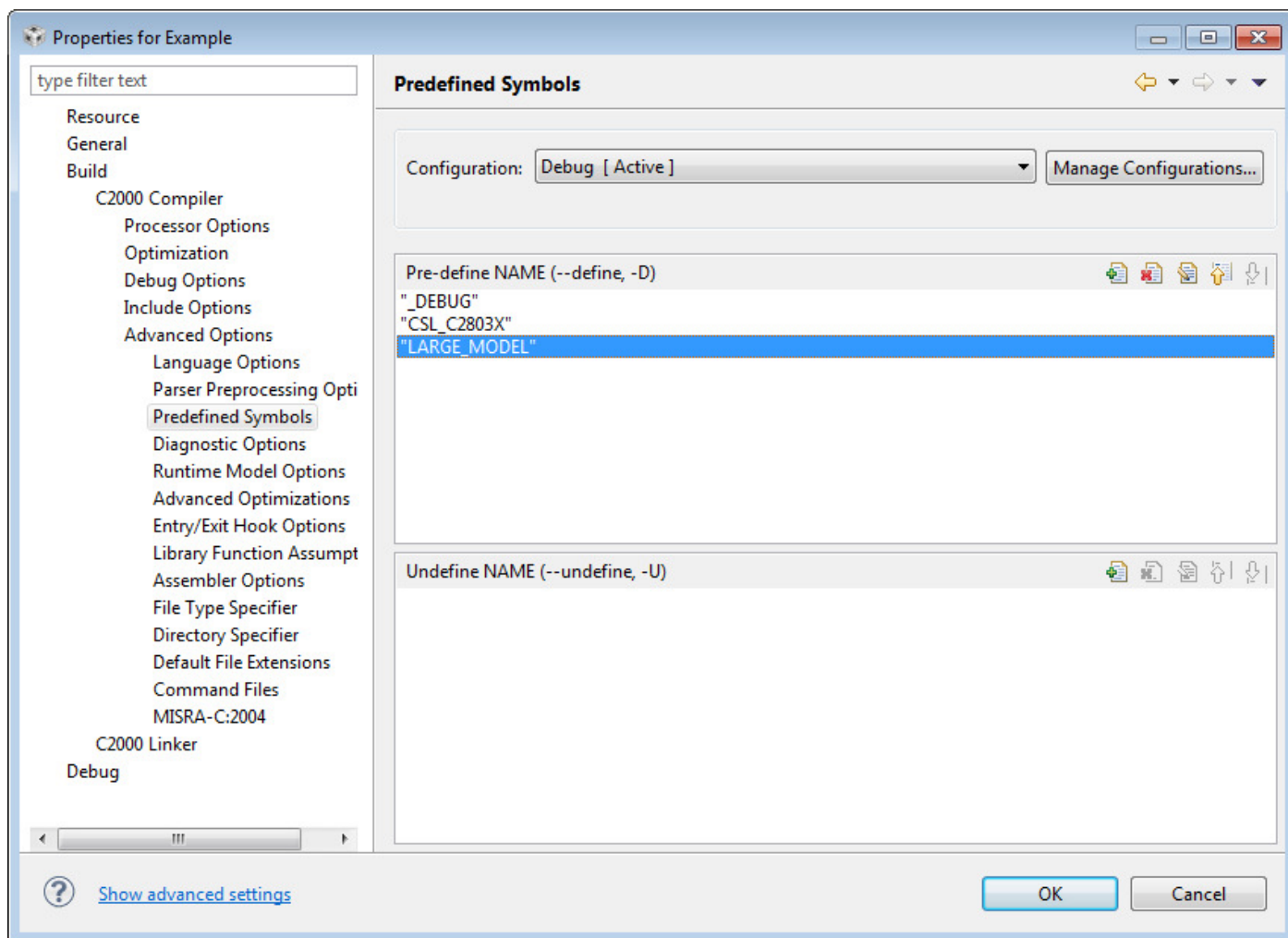
Expand "Advanced Options" and select "Predefined Symbols". Under "Pre-define NAME" click "Add"

Type "_DEBUG" (including quotes), and then click "OK".

Repeat for "LARGE_MODEL" (including quotes)

and repeat for the following depending on your microcontroller:

C2808	"CSL_C280X"
C2802x	"CSL_C2802X"
C2803x	"CSL_C2803X"
C2806x	"CSL_C2806X"
C2833x	"CSL_C2833X"

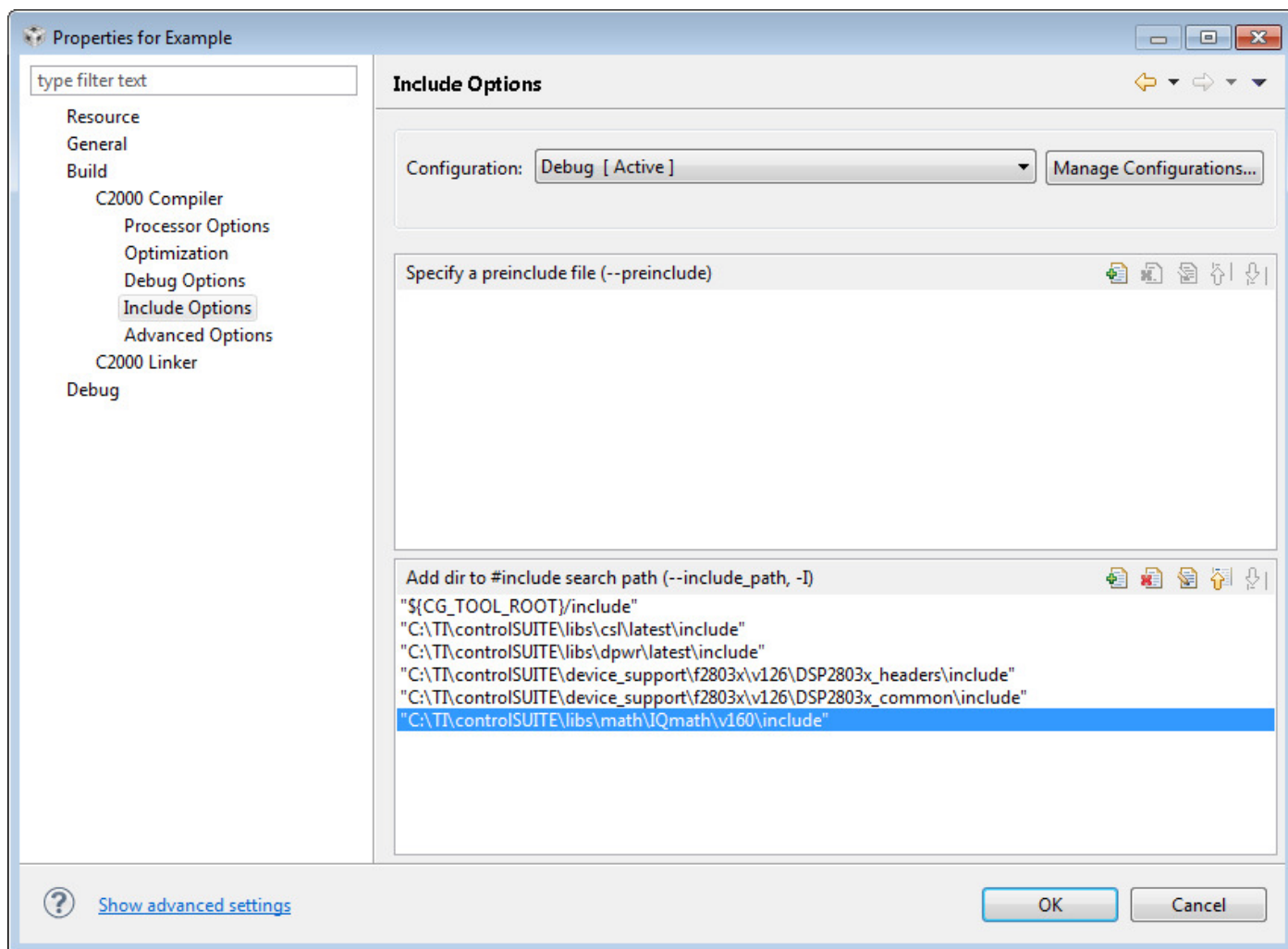


The predefined symbols window should look like the above.

Now select "Include Options" from the left hand side and add the following includes paths, according to the microcontroller that you are using, one at a time.

C2802x	"C:\TI\controlSUITE\libs\cs\latest\include"
	"C:\TI\controlSUITE\libs\dpwr\latest\include"
	"C:\TI\controlSUITE\device_support\f2802x\v200\f2802x_headers\include"
	"C:\TI\controlSUITE\device_support\f2802x\v200\f2802x_common\include"
	"C:\TI\controlSUITE\device_support\f2802x\v200"
	"C:\TI\controlSUITE\device_support\f2803x\v126" (note: this is to include the DSP2803x_Comp.h header file which enables use of the DAC ramp sub-module unavailable in F2802x_Comp.h from f2802x\v200)
	"C:\TI\controlSUITE\libs\math\IQmath\v160\include"
...	
C2803x	"C:\TI\controlSUITE\libs\cs\latest\include"
	"C:\TI\controlSUITE\libs\dpwr\latest\include"
	"C:\TI\controlSUITE\device_support\f2803x\v126\DSP2803x_headers\include"
	"C:\TI\controlSUITE\device_support\f2803x\v126\DSP2803x_common\include"
	"C:\TI\controlSUITE\libs\math\IQmath\v160\include"
...	
C2806x	"C:\TI\controlSUITE\libs\cs\latest\include"
	"C:\TI\controlSUITE\libs\dpwr\latest\include"
	"C:\TI\controlSUITE\device_support\f2806x\v130\F2806x_headers\include"
	"C:\TI\controlSUITE\device_support\f2806x\v130\F2806x_common\include"
	"C:\TI\controlSUITE\libs\math\IQmath\v160\include"
...	

C2833x	"C:\TI\controlSUITE\libs\cs\latest\include"
	"C:\TI\controlSUITE\libs\dpwr\latest\include"
	"C:\TI\controlSUITE\device_support\f2833x\v133\DSP2833x_headers\include"
	"C:\TI\controlSUITE\device_support\f2833x\v133\DSP2833x_common\include"
	"C:\TI\controlSUITE\libs\math\IQmath\v160\include"



These include the TI header files that we need for setting up the chip and programming the peripherals.

Click "OK" to apply all changes.

A "main.c" source file should have been added to your project when you created it. Expand your project in the "Project Explorer" tab. Double click on the file "main.c".

You can add a source file by: "File->New->Source File" if required.

Add the following code to main.c and save the file.

```
#include "cs1.h"
#include "dpwr.h"

void main( void )
{
    SYS_init();
    GPIO_config( GPIO_34, GPIO_DIR_OUT, false );
    while ( 1 )
    {
        SYS_msDelay(1000);
        GPIO_tog( GPIO_34 );
    }
}
```

Then build your new project by selecting "Project->Build Project". The project should compile and link without any errors. To load and debug your code press the debug icon and then press F8 to run.

IMPORTANT NOTICE

All material presented in this application note, associated software, exercises, appendices and any oral presentations made by or on behalf of Biricha Digital Power Ltd are for educational purposes only, are not intended for production or commercial applications and may contain irregularities and defects or errors. Furthermore, none of the material has been tested or certified by any government agency or industry regulatory organization or any other third party organization.

Biricha Digital Power Ltd makes no warranties or representations, express, implied or statutory, including any implied warranties of merchantability, fitness for a particular purpose, lack of viruses, lack of patent infringements, accuracy or completeness of responses and lack of negligence. Prior to using, incorporating or distributing any of the material that has been included in this application note in any commercial product, you must thoroughly test the product and the functionality in that product and be solely responsible for any problems or failures.

In no event shall Biricha Digital Power Ltd, or any applicable licensor, be liable for any special, indirect, incidental, or punitive damages, however caused, arising in any way out of your use of the material presented in the DPS Workshop. These excluded damages include, but are not limited to, cost of removal or reinstallation, procurement of substitute goods or services, outside computer time, labour costs, loss of data, loss of goodwill, loss of profits, loss of savings, loss of use or interruption of business or other economic loss.