



nRF51 Development with GCC and Eclipse

nAN-29

Application Note v1.1

This application note is for developers who want to use Eclipse, the GNU C Compiler (GCC), and the GNU Debugger (GDB) to develop and debug programs on Nordic Semiconductor's nRF51 series devices.

The nRF51 series Software Development Kits (SDK) come with GCC compatible examples. This application note shows you how to import these examples into Eclipse allowing you to make use of the powerful Eclipse editor with C-language syntax highlighting and completion. You will also learn how to configure and use the GDB plug-in with Eclipse to debug the examples.

1 Required tools

The following tools were used in this application note. The version number of each tool is supplied for reference, but as software is updated frequently, it is not required that the specific version be used.

nRF51 SDK

Download from www.nordicsemi.com.

1. Select **Custom location install**.
2. Edit Makefile.windows at:
C:\Nordic Semiconductor\nRF51 SDK_v4.2.0.25045\Nordic\nrf51822\Source\templates\gcc
3. Set GNU_INSTALL_ROOT and GNU_VERSION to proper values.

This needs to be done for every device you plan to use (nRF51422, nRF51822, nRF51922 etc). GNU_VERSION can be determined by checking the version subfolder name at:

C:\Program Files (x86)\GNU Tools ARM Embedded\4.7 2013q1\lib\gcc\arm-none-eabi.

For example:

GNU_INSTALL_ROOT := \$(PROGFILES)/GNU Tools ARM Embedded/4.7 2013q1

GNU_VERSION := 4.7.3

GNU Tools for ARM Embedded Processors version 4.7 2013q1

Download and install from <https://launchpad.net/gcc-arm-embedded>.

Core Utility

Download and install the option "Complete package, except sources" from:

<http://gnuwin32.sourceforge.net/packages/coreutils.htm>

Make utility

Download and install from <http://gnuwin32.sourceforge.net/packages/make.htm>.

Add the make utility to your system path (typically **C:\Program Files (x86)\GnuWin32\bin**).

J-Link software version 4.66

Download and install from www.segger.com/jlink-software.html.

Eclipse IDE for C/C++ Developers version 1.5.2.20130211-1820 (Juno Service Release 2)

Download and install from www.eclipse.org/downloads/.

Additional tools used in this document:

- CDT Main Features version 8.1.2.201302132326
- GDB Hardware Debugging version 7.0.0.201302132326
- embsysregview version 0.2.2

Note: Instructions for adding these tools are found below in *Section 1.1 “Setting up Eclipse”*.

1.1 Setting up Eclipse

1. Start Eclipse.
2. Click **Help** and select **Install New Software**.
3. Add **<http://download.eclipse.org/tools/cdt/releases/juno>** to the list of repositories (replace “juno” if you have installed another release of Eclipse).
4. Install **CDT Main Features** and **GDB Hardware Debugging**.
5. Repeat step 2 and add **<http://embsysregview.sourceforge.net/update>** to the list of repositories.
6. Install **embsysregview**.
7. To install the register view file for Nordic Semiconductor devices copy file **nrf51.xml** from:
C:\Nordic Semiconductor\NRF51 SDK_v4.2.0.25045\Nordic\nrf51822\SVD\nrf51.xml
to the folder
C:\eclipse\plugins\org.eclipse.cdt.embsysregview_x.x.x\data\cortex-m0\Nordic.
(Create a Nordic subfolder if it does not exist.)

1.2 Import existing Eclipse project to Workspace

We will use the example project ble_app_hrs in this guide. The following is a set of instructions to help you import an existing Eclipse project to your workspace. An example setup can be seen in the steps below.

Create a new C project in Eclipse

1. On the File menu, click **Import** (or right click on Project Explorer and select **Import**).
2. Select **Existing Projects into Workspace**.

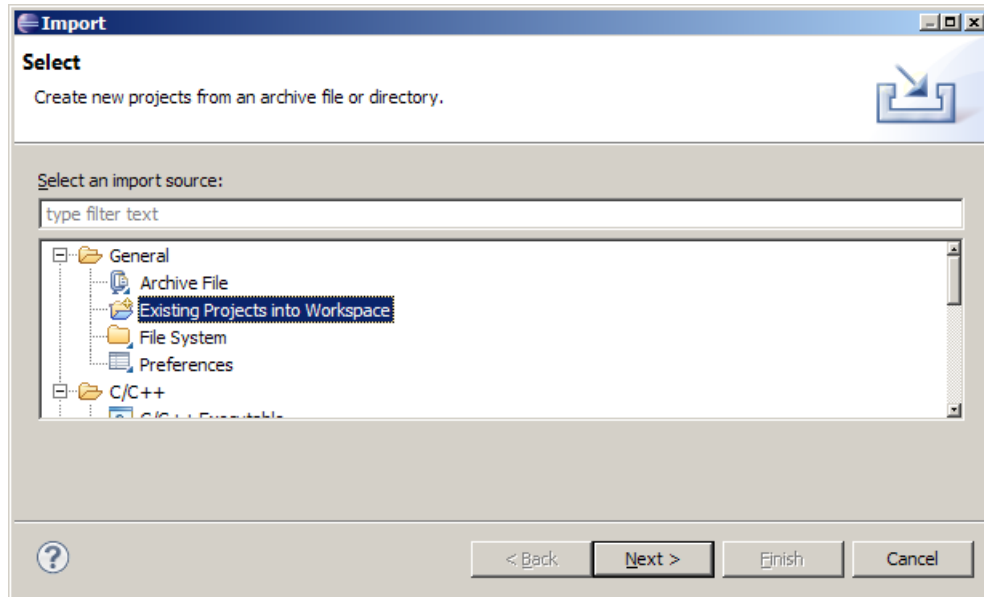


Figure 1 Import dialog

3. Choose **Select Root Directory** and browse to locate ble_app_hrs/gcc.
4. Click **Finish**.

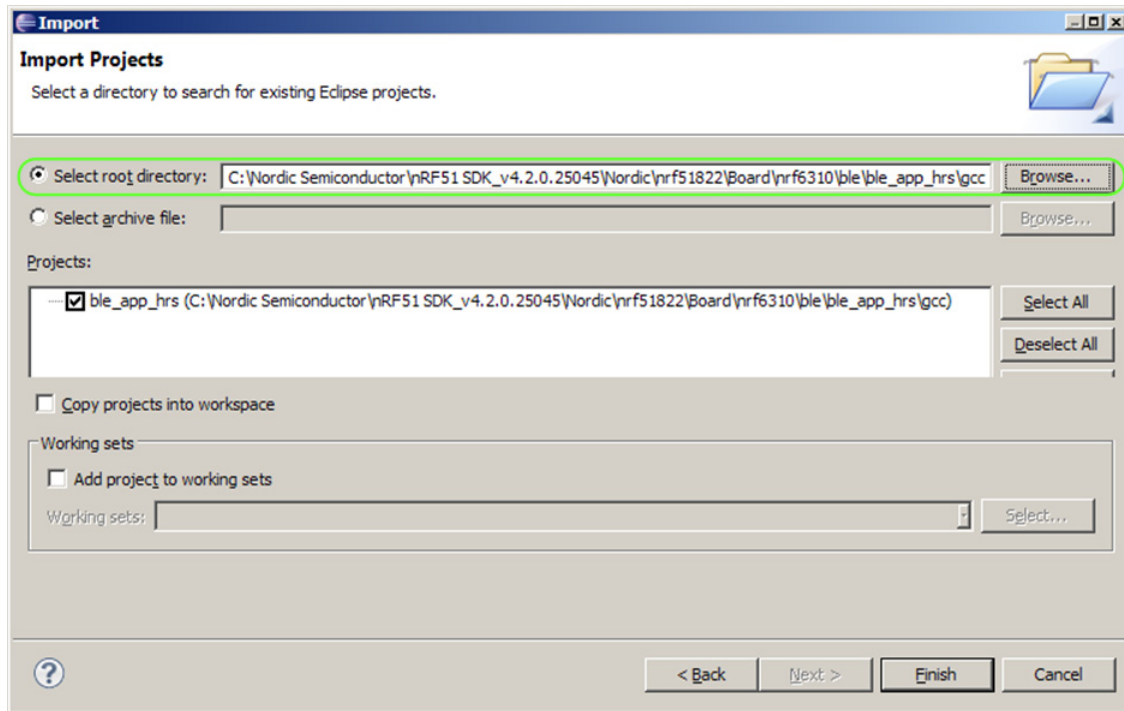


Figure 2 Root directory selection

2 Setting up a project for debugging in Eclipse

1. In the **Project Explorer** select the **ble_app_hrs** project.
2. Open the **Run** menu and click **Debug Configurations**.
3. Right click **GDB Hardware Debugging** and select **New**.
4. Change the name to ble_app_hrs.
5. Under C/C++ Application, click **Browse**.
6. Select the **_build\ble_hrs_gcc_s110_xxaa.out** file in the **_build** folder of the project.
7. Set the Build configuration to **Debug**.
8. Click **Select other** and change the preferred launcher to **GDB (DSF) Hardware Debugging Launcher**. See *Figure 3*.

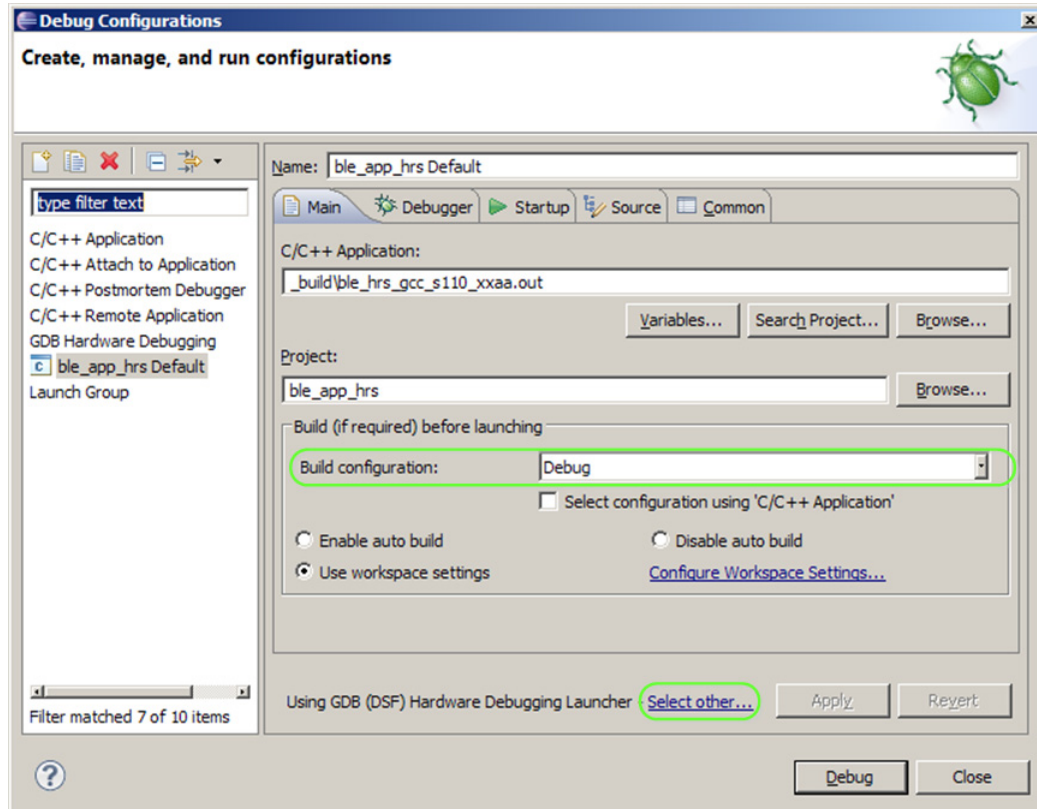


Figure 3 Debug configuration

9. Select the **Debugger** tab. Change the GDB Command to **arm-none-eabi-gdb**.
10. In the Remote Target box, change the Port number to **2331** as shown in *Figure 4*.

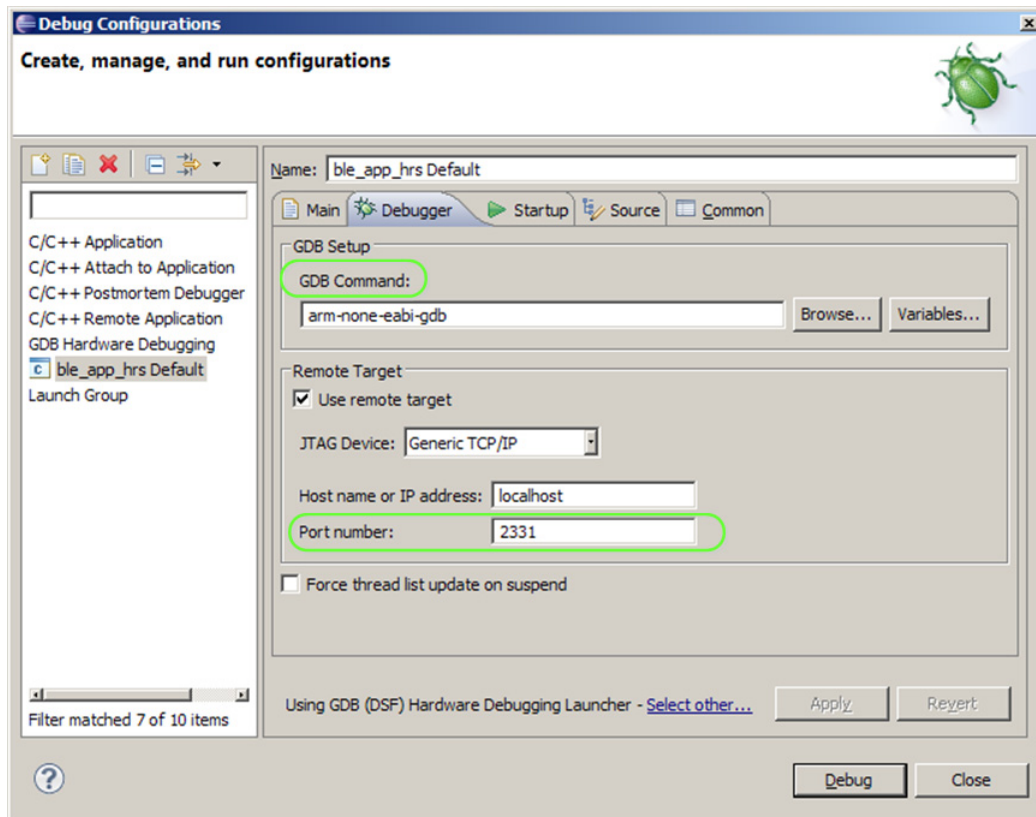


Figure 4 Entering GDB command and TCP/IP port number

11. Select the **Startup** tab.
12. Enter the following commands in the **Initialization Commands** box:
 - **mon speed 10000** (Sets the clock speed on the programming pins to 10 MHz)
 - **mon endian little** (Selects little endian format)
 - **mon flash download = 1** (Tells the debugger that our device supports flash download)
 - **mon flash device = NRF51822** (Sets the device to NRF51822)
 - **mon reset 0** (Selects reset type 0)

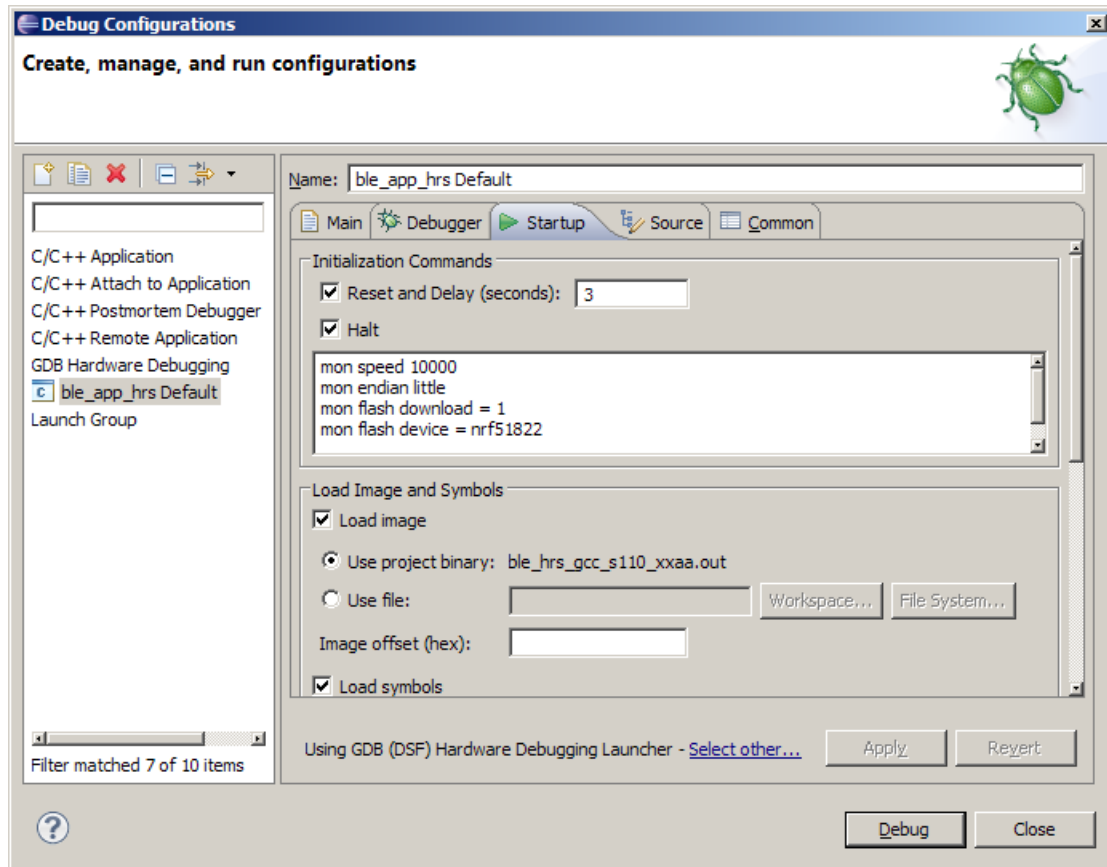


Figure 5 Configuring initialization commands

13. Check **Set breakpoint at** and type in **main**. (You may have to scroll down to see this option.)
14. Check **Resume**.
15. Confirm that the J-Link GDB server is running. A shortcut named **J-Link GDB Server via SWD** can be found under the SEGGER entry on the Start menu in Windows.
16. Click **Debug** to begin debugging.

3 Flash download

Flash downloading and running a program on your device without debugging is possible by using the **nrfjprog.exe** tool directly from Eclipse. Follow these steps for flash downloading.

1. On the Run menu, click **External Tools**, click **External Tools Configurations**, select **Program**, and then click **New**.
2. In the Name field type **Download and run ble_app_hrs** as the name for the configuration. See *Figure 6*.
3. In the Location field type the full path to **nrfjprog.exe** (default **C:\Program Files (x86)\Nordic Semiconductor\nrf51\bin\nrfjprog.exe**).
4. In the Working Directory field select the output folder for ble_app_hrs (**\${workspace_loc:/ble_app_hrs/_build}**).
5. In the **Arguments** field enter **--reset --program "\${workspace_loc:/ble_app_hrs/_build}/ble_hrs_gcc_s110_xxaa.hex"**.
6. If you have more than one board connected, you can also select the board by defining the serial number with the **-s** option.
7. When you are finished, click **Apply** and close the dialog box. The command can now be found on the Run menu.

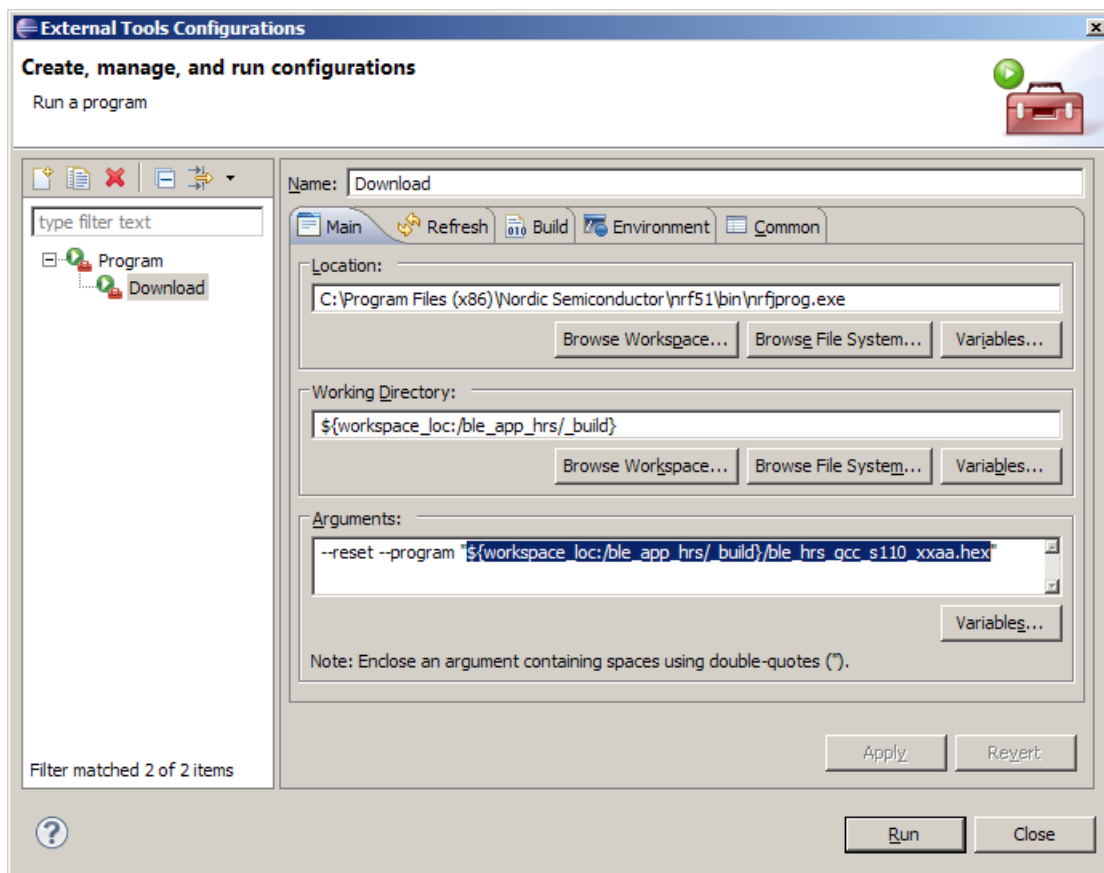


Figure 6 External tool configuration

4 Enabling the EmbSys Register Viewer

EmbSys Register Viewer is a plug-in for Eclipse that enables you to monitor and modify the peripheral registers on the chip you are debugging. For more information see <http://embsysregview.sourceforge.net/>.

1. Open the Window menu in Eclipse and click **Preferences**.
2. In the left navigation bar expand **C/C++**, and then expand **Debug**.
3. Select **EmbSys Register View**.
4. In the Architecture drop down, select **cortex-m0**.
5. In the Vendor drop down, select **Nordic**.
6. In the Chip drop down, select **nrf51**. See *Figure 7*.

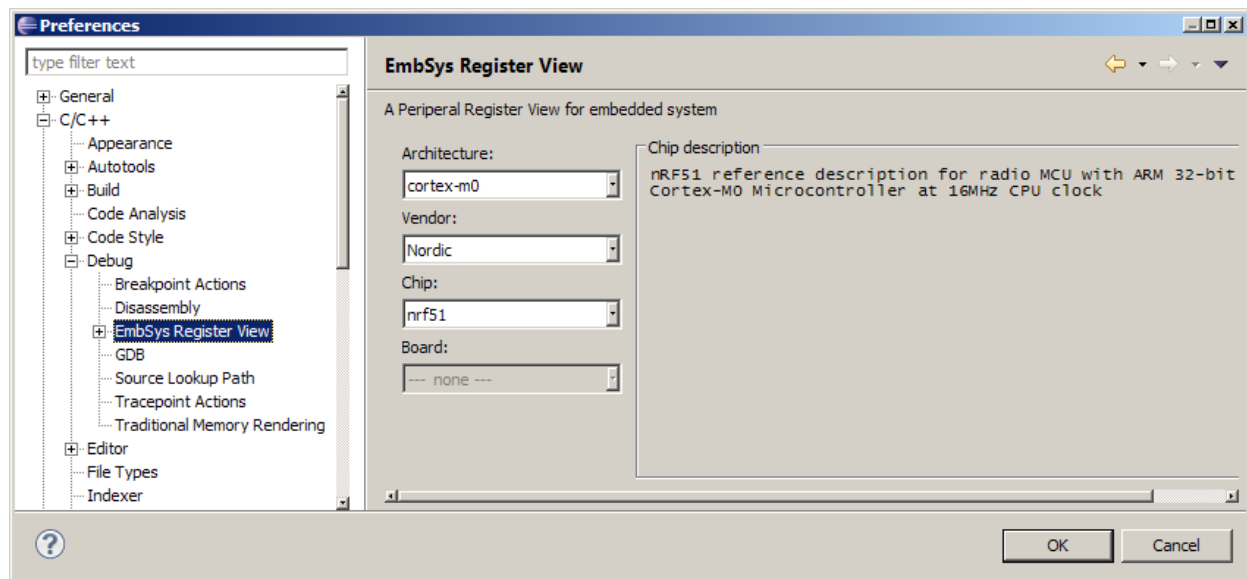


Figure 7 Configuring Emb Sys Register view

Note: To show the EmbSys Register View when debugging, from the Window menu select **Show View**, click **Other Debug**, then select **EmSys Registers**.

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Life support applications

Nordic Semiconductor's products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

Contact details

For your nearest dealer, please see <http://www.nordicsemi.com>.

Information regarding product updates, downloads, and technical support can be accessed through your My Page account on our homepage.

Main office: Otto Nielsens veg 12
7052 Trondheim
Norway

Phone: +47 72 89 89 00

Fax: +47 72 89 89 89

Mailing address: Nordic Semiconductor
P.O. Box 2336
7004 Trondheim
Norway



Revision History

Date	Version	Description
June 2013	1.1	Updated <i>Section 1 "Required tools"</i> on page 2 and <i>Section 3 "Flash download"</i> on page 9.
November 2012	1.0	First release