

N I N T E N D O
NITRO-System

Build Procedures for Samples Using the IDE

How to Build the NITRO-System Samples Using CW IDE

Version 1.0.0

The contents in this document are highly
confidential and should be handled accordingly.

Confidential

These coded instructions, statements, and computer programs contain proprietary information of Nintendo of America Inc. and/or Nintendo Company Ltd. and are protected by Federal copyright law. They may not be disclosed to third parties or copied or duplicated in any form, in whole or in part, without the prior written consent of Nintendo.

Table of Contents

1	Introduction	6
2	CodeWarrior IDE Configuration	7
3	Building the Archive Sample	8
3.1	Creating a Project for a NITRO Application	8
3.2	Preparing the Source and Data of the Samples	9
3.3	Adding the Sample's Source to the Project	9
3.4	Adding NITRO-System to the Project	11
3.4.1	Configuration of the Access Path for the Debug Version of the Libraries	11
3.4.2	Adding the Debug Versions of the Libraries	14
3.5	Adding the Sample's Data to the Project	15
3.6	Building the Project	16
4	Building the MultiCellAnimation Sample	17
4.1	Building the g2d Demo Library	17
4.1.1	Creating the Demo Library Project	17
4.1.2	Preparing the Source and Data of the Samples	17
4.1.3	Adding the Demo Library's Source to the Project	18
4.1.4	Adding NITRO-System to the Project	18
4.1.5	Building the Library	18
4.2	Building the MultiCellAnimation Sample	18
4.2.1	Creating the MultiCellAnimation Sample Project	18
4.2.2	Adding the Demo Library	18
4.2.3	Building the Project	20
4.3	Subprojects	21

Figures

Figure 2-1 IDE Preferences.....	7
Figure 3-1 New Dialog Box	8
Figure 3-2 New Project Dialog Box	9
Figure 3-3 Add Files Dialog Box.....	10
Figure 3-4 Nitro TS Debug Settings Dialog Box.....	11
Figure 3-5 Browse for Folder Dialog Box	12
Figure 3-6 Nitro TS Debug Settings Dialog Box.....	13
Figure 3-7 Nitro TS Debug Target Postlinker Settings Panel	15
Figure 4-1 Select Project Stationery Selection Screen	17
Figure 4-2 IDEMultiCellAnimation Project Window	20
Figure 4-3 Configuration of the Targets.....	21

Revision History

Version	Revision Date	Description
1.0.0	2007/02/13	Initial Version.

1 Introduction

This document explains the procedure for building the NITRO-System samples using CodeWarrior integrated development environment (IDE). For specific details about CodeWarrior IDE, please refer to the *CodeWarrior™ Development Studio for Nintendo DS™ Targeting Manual Version 2.0*.

2 CodeWarrior IDE Configuration

You will have to configure the CodeWarrior IDE settings first.

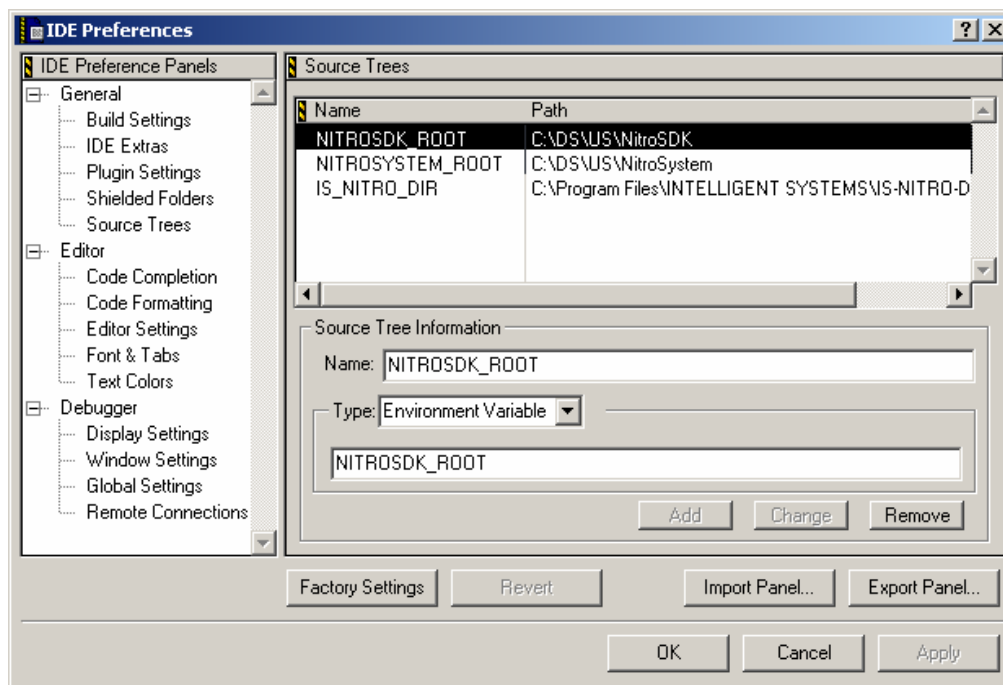
In order to build and run the NITRO-System samples, a NITRO-SDK, an IS-NITRO-DEBUGGER, and a NITRO-System are required, so in CodeWarrior IDE, configure the location where these programs are installed as the source tree. If the command line build environment has already been set up, the location where these programs are installed will have already been configured in the environment variables. Configure the source tree in CodeWarrior IDE using the values of these environment variables.

To begin with, configure the source tree for NITRO-SDK. The configuration procedure is as follows.

- Run the IDE for CodeWarrior for NINTENDO DS Version 2.0
- Selecting **Environment Settings** from the Edit menu will display the IDE Preferences dialog box
- Select **General – Source Tree** from the IDE Preferences panel
- Enter “NITROSDK_ROOT” in the **Name:** input section of the Source Tree Information, select **Environment Variable** as the Type, enter the environment variable name “NITROSDK_ROOT” below that, and click the **Add** button

You should also register the source trees for NITROSYSTEM_ROOT and IS_NITRO_DIR in the same way.

Figure 2-1 IDE Preferences



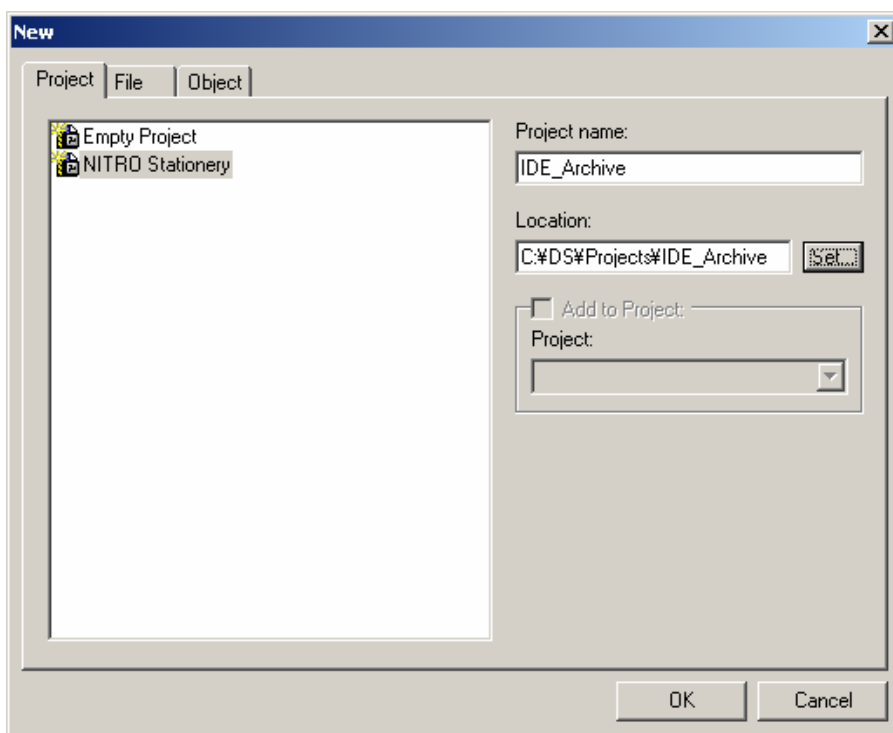
3 Building the Archive Sample

This section explains the procedure through the build of the Archive sample of NITRO-System's `fn` library using CodeWarrior IDE.

3.1 Creating a Project for a NITRO Application

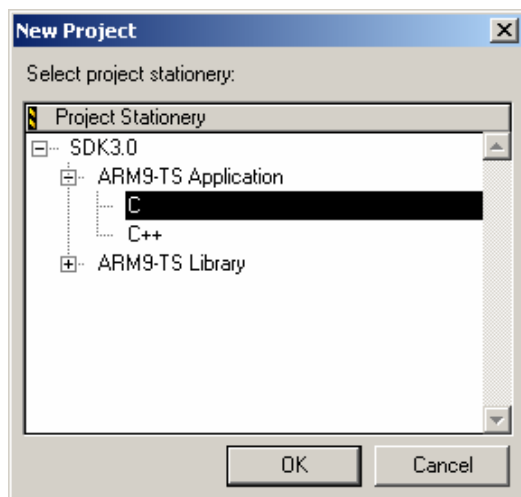
Selecting **Create New** from the File menu of the IDE will display the New dialog box. Select **NITRO Stationery** in the Project tab of the New dialog box, enter the Project name and Location of the project in the right section, and click the **OK** button.

Figure 3-1 New Dialog Box



Clicking the **OK** button will display the New Project dialog box. Select **SDK3.0-ARM9-TS Application-C** in this dialog box's Project Stationery section and click the **OK** button to create the project.

Figure 3-2 New Project Dialog Box



3.2 Preparing the Source and Data of the Samples

Having created a project for the Archive sample, we will now prepare the source and data of the fnd library's Archive sample for the project.

First, the main.c file that has been added to the project in advance by the stationery will not be used, so delete it from the project directory. Next, copy the three directories "include," "src," and "data" from the NitroSystem/build/demos/fnd/archive directory of NITRO-System to the project directory.

At this point, the preparations required for building and executing the samples have been completed. Next, these files will be registered in the project.

3.3 Adding the Sample's Source to the Project

In the Sources group of the project, add the source of the sample that was copied into the project directory.

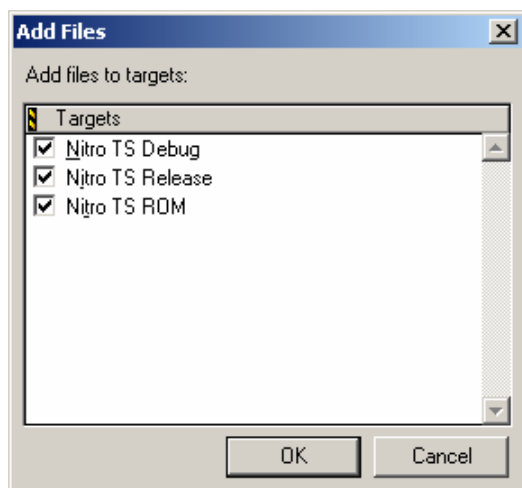
Adding files to a project requires two operations, configuring the access path to the files being added and adding the files. However, under the project directory (the directory where the .mcp file is), the access path is configured when a project is created. So there is no need to configure the access path again.

First, delete the main.c that was added in advance by the stationery, as it is not required. Right-clicking on main.c in the Sources group of the CodeWarrior IDE project window and selecting **Delete** from the menu that is displayed will delete that file.

Next, right-clicking the Sources group in the project window and selecting **Add Files** from the menu that is displayed will bring up the **Select files to add...** dialog box. In this dialog, select `main.c`, `sdk_init.c`, and `nns_util.c` from the `src` directory that was just copied and click the **Open** button.

Clicking the **Open** button will then display the **Add Files** dialog box. The sample's source will be added to the three targets **Nitro TS Debug**, **Nitro TS Release**, and **Nitro TS ROM**, so verify that there is a check by all three targets and click the **OK** button.

Figure 3-3 Add Files Dialog Box



3.4 Adding NITRO-System to the Project

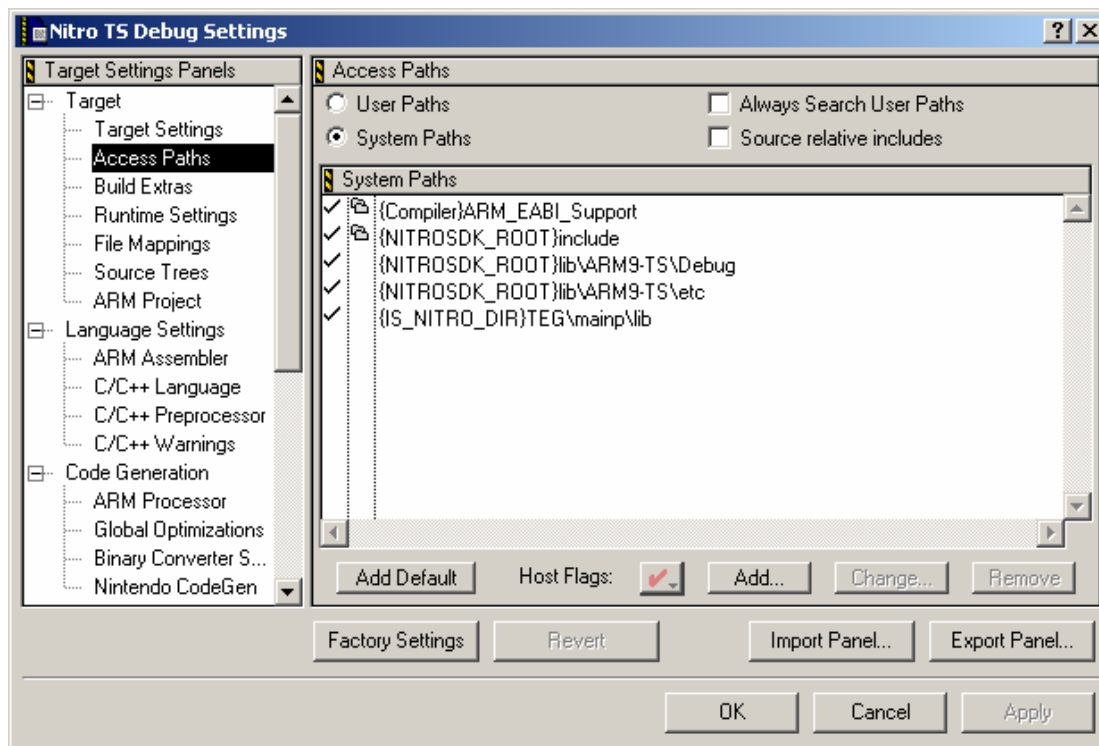
We will now add NITRO-System to the project. This section explains how to configure the **Nitro TS Debug** target. The configuration method is the same for the other targets, so they should be configured in the same way.

3.4.1 Configuration of the Access Path for the Debug Version of the Libraries

After verifying that the project's target is **Nitro TS Debug**, selecting **Nitro TS Debug Settings** from the Edit menu will display the **Nitro TS Debug Settings** dialog box.

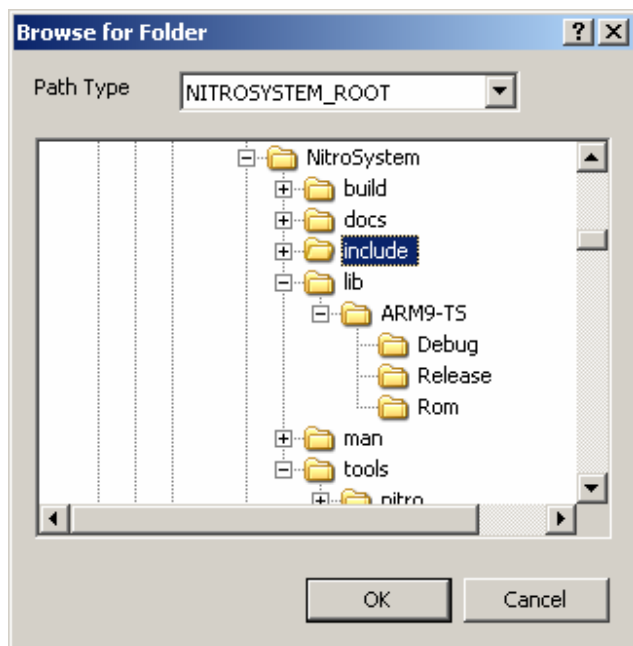
Selecting **Target – Access Paths** from the target settings panel of this dialog will display the access path panel on the right. The NITRO-System access path must be added to the system path, so select the [System Paths] radio button at the top of the panel.

Figure 3-4 Nitro TS Debug Settings Dialog Box



Clicking the [Add...] button of this panel will display the [Browse for Folder] dialog box. Selecting [NITROSYSTEM_ROOT] as the path type here, selecting the directory that contains the NITRO-System header files (NitroSystem/include), and clicking the **OK** button will add "{NITROSYSTEM_ROOT}include" to the system path.

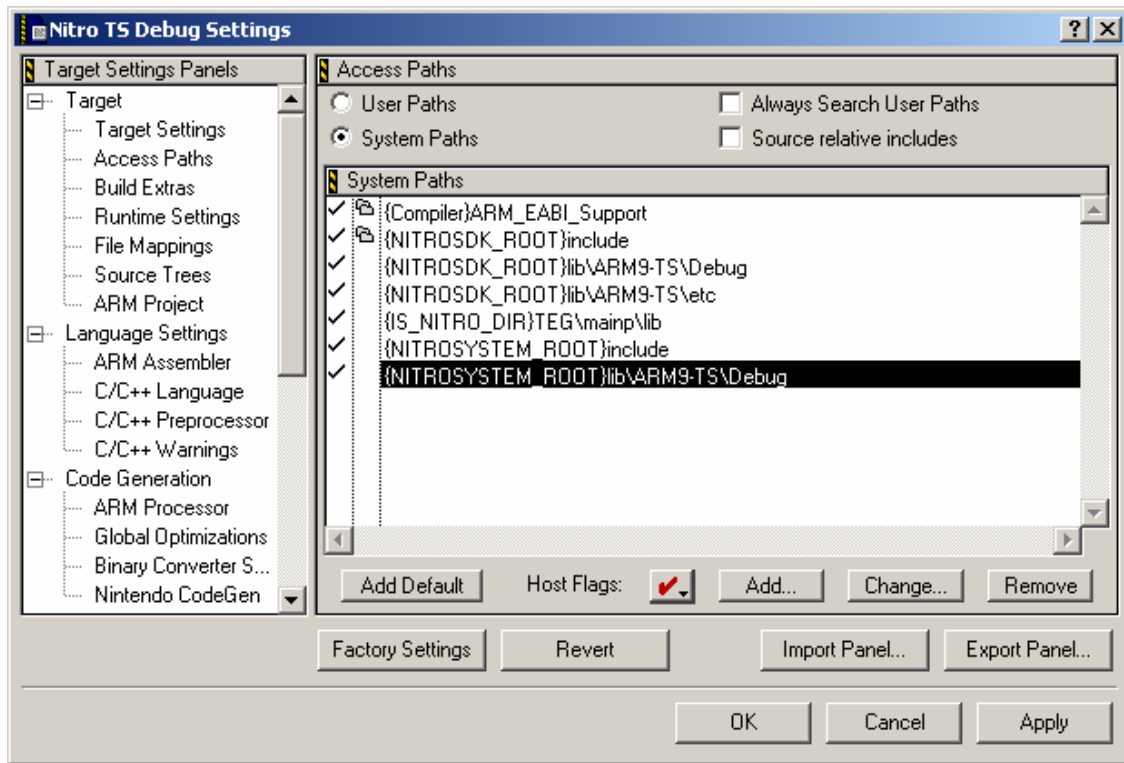
Figure 3-5 Browse for Folder Dialog Box



A folder icon is displayed at the beginning of the access path that was added. This icon indicates that the path is going through all directories under the access path that has been configured. In the command line build environment, the path only goes through the include directory. So in the event that you want to make the environment match it strictly, click on the icon and delete it.

Add the `NitroSystem/lib/ARM9-TS/Debug` directory, which stores the debug version of the library, to the system path in the same way.

Figure 3-6 Nitro TS Debug Settings Dialog Box



3.4.2 Adding the Debug Versions of the Libraries

Create a group with the same structure as the TS Libs group that has been added to the NITRO-SDK libraries, and add the NITRO-System libraries inside this group.

Selecting **Create Group** from the Project menu will display the group creation dialog box. Entering "NitroSystem" in this dialog and clicking the **OK** button will create a group named **NitroSystem** in the project file list. Create a **TS Debug** group inside the **NitroSystem** group in the same way.

Right-clicking the **NitroSystem/TS Debug** group will display a menu; select **Add Files** from this menu and add the following six libraries that are in `NitroSystem/lib/TS/Debug`.

- libnssnd.a
- libnssfnd.a
- libnsg2d.a
- libnsg3d.a
- libnsgfd.a
- libnsmcs.a

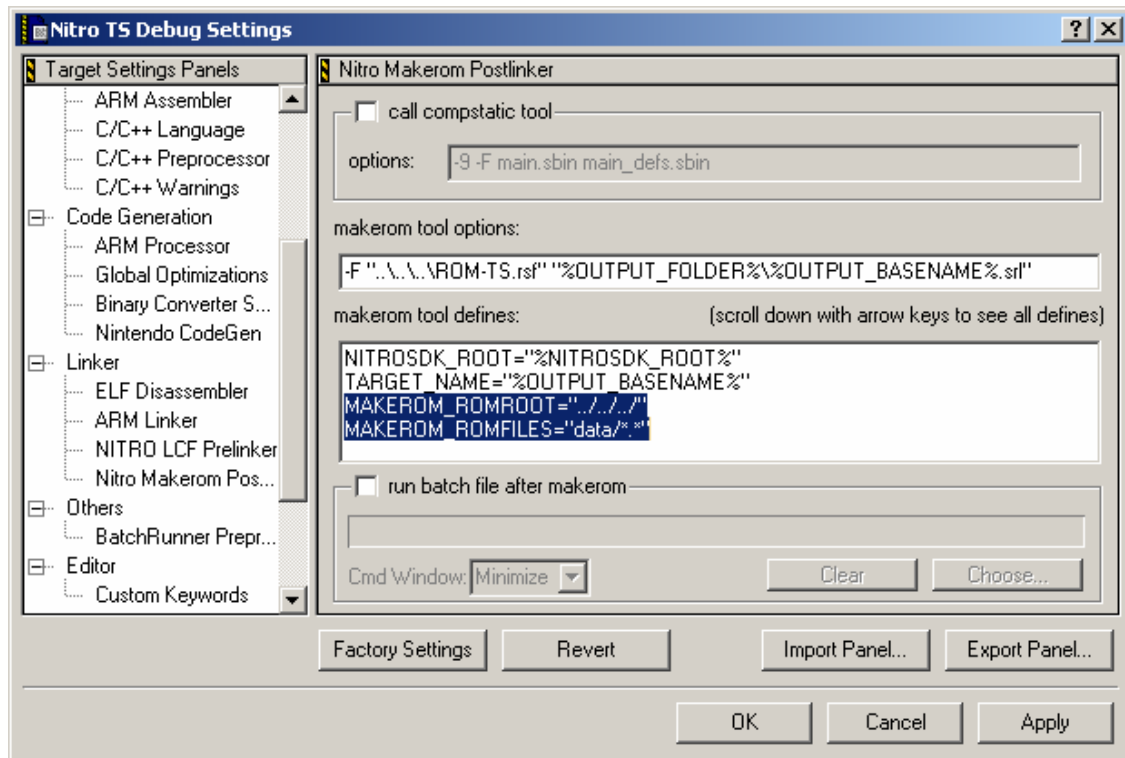
In the **Add Files** dialog box, place a check only by the **Nitro TS Debug** target and click the **OK** button.

3.5 Adding the Sample's Data to the Project

Register the data used by the sample program in the project.

Verify that the project target is **Nitro TS Debug**, and select **Nitro TS Debug Settings** from the Edit menu. Selecting [Linker] – [Nitro Makerom Postlinker] from the target settings panel will display the [Nitro Makerom Postlinker] panel.

Figure 3-7 Nitro TS Debug Target Postlinker Settings Panel



Change the settings of MAKEROM_ROMROOT and MAKEROM_ROMFILES in the **makerom tool defines:** text box of this dialog as follows.

```
MAKEROM_ROMROOT= " ../.. "
MAKEROM_ROMFILES= " data/*.x "
```

In MAKEROM_ROMROOT, configure the directory where there is data using the relative path from the directory in which ".srl" files are output. Also, in MAKEROM_ROMFILES, specify the files to include in the ROM.

3.6 Building the Project

At this point the preparations for building the Archive sample have been completed. Verify that the project target is **Nitro TS Debug**, select **Make** from the **Project** menu (or press the F7 key), and build the target.

The method for configuring the **Nitro TS Release** target and the **Nitro TS ROM** target is the same. Configure the projects in the same way and try building them.

4 Building the MultiCellAnimation Sample

This section explains the procedure up through the building of the MultiCellAnimation sample of the NITRO-System g2d library. This is an example of using a private library.

4.1 Building the g2d Demo Library

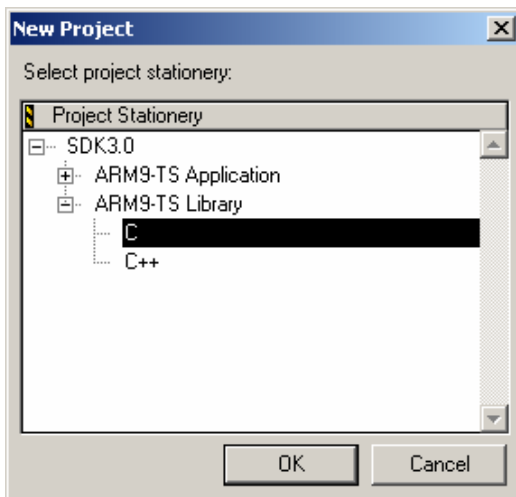
The g2d library samples use a private demo library unique to g2d samples called `libnmsg2d_demo.a`. First, create a project for this demo library.

4.1.1 Creating the Demo Library Project

Selecting **Create New** from the File menu of the CodeWarrior IDE will display the New dialog box. Select **NITRO Stationery** in the Project tab of this Create New dialog, enter the name and location of the project in the right section, and click the **OK** button.

Clicking the **OK** button will display the New Project dialog box. Select **SDK3.0-ARM9-TS Library-C** in this dialog box's project stationery and click the **OK** button to create the project.

Figure 4-1 Select Project Stationery Selection Screen



4.1.2 Preparing the Source and Data of the Samples

Having created a project for the demo library, we will now prepare the source of the g2d samples' demo library.

First, the `main.c` file that has been added to the project in advance by the stationery will not be used, so delete it from the project directory. Next, copy the two directories "include" and "src" from the `NitroSystem/build/demos/g2d/demolib` directory of NITRO-System to the project directory.

4.1.3 Adding the Demo Library's Source to the Project

Add the demo library's source to the **Sources** group in the project. The method for adding the source is the same as for the Archive sample.

First, delete the `main.c` file added by the stationery from the project, as it is not required. Next, add the four files `fontData.c`, `loader.c`, `print.c`, and `system.c` in the `src` directory that was just copied to the project directory. At the **Add Files** dialog that is displayed at this time, do not forget to add a check for all three targets.

4.1.4 Adding NITRO-System to the Project

We will now add NITRO-System to the project. This section explains how to configure the **Nitro TS Debug** target. The configuration method is the same for the other targets, so they should be configured in the same way.

There is no need to add the NitroSystem library binaries (.a files) when building the demo library. Only configure the access path to the NitroSystem include directory. The configuration method is the same as for the Archive sample.

Select **Targets – Access Paths** from the target settings panel of the **Nitro TS Debug Settings** dialog box, and add the directory that contains the NITRO-System header files (NitroSystem/include) to the system path.

4.1.5 Building the Library

At this point the preparations for building the demo library have been completed. Verify that the project target is **Nitro TS Debug**, select **Make** from the Project menu (or press the F7 key), and build the target.

Build the **Nitro TS Release** target and the **Nitro TS ROM** target by performing the same configuration.

4.2 Building the MultiCellAnimation Sample

This section explains how to build the g2d library sample, MultiCellAnimation.

4.2.1 Creating the MultiCellAnimation Sample Project

The method for creating the MultiCellAnimation sample project is the same as the method for creating the Archive sample project. Create the project following the creation method for the Archive sample project, and add the sample source, data, and NITRO-System libraries.

4.2.2 Adding the Demo Library

The g2d library samples use the g2d demo library, so the demo library must also be added to the project. This section will explain the configuration method for the **Nitro TS Debug** target. The configuration method is the same for the other targets, so configure them the same way.

4.2.2.1 Configuration of the Access Path for the Demo Library

We will now register the demo library in the user access paths. Select **Targets – Access Paths** from the target settings panel of the **Nitro TS Debug Settings** dialog box, and add the include directory of the g2d demo library and the directory where the debug version of the library is stored (bin/ARM9-TS/Debug in the demo library project) to the user paths.

4.2.2.2 Registering the Debug Version of the Demo Library

Create a group with the same structure as the **TS Libs** group that has been added to the NITRO-SDK libraries, and add the g2d demo library inside this group.

Selecting **Create Group** from the Project menu will display the group creation dialog box. Entering “G2dDemoLib” in this dialog and clicking the **OK** button will create a group named **G2dDemoLib** in the project file list. Create a **TS Debug** group inside the **G2dDemoLib** group in the same way.

Right-clicking the **G2dDemoLib/TS Debug** group will display a menu. Select **Add Files** from this menu and add the file `bin/ARM9-TS/Debug/libg2d_demo.a` in the project directory of the demo library.

In the **Add Files** dialog box, place a check only by the **Nitro TS Debug** target and click the **OK** button.

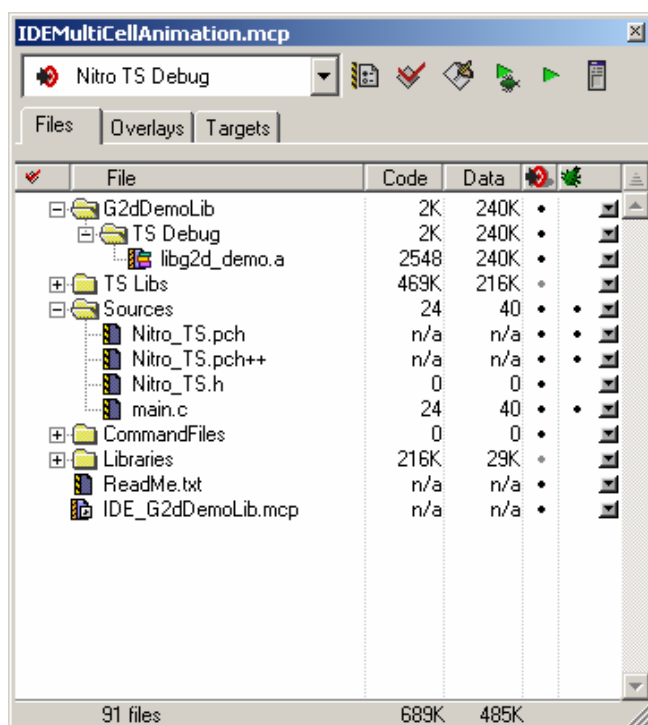
4.2.3 Building the Project

At this point the preparations for building the MultiCellAnimation sample have been completed. Verify that the project target is **Nitro TS Debug**, select **Make** from the Project menu (or press the F7 key), and build the target.

Build the **Nitro TS Release** target and the **Nitro TS ROM** target by performing the same configuration.

The contents of the IDEMultiCellAnimation project window are shown in Figure 4-2.

Figure 4-2 IDEMultiCellAnimation Project Window



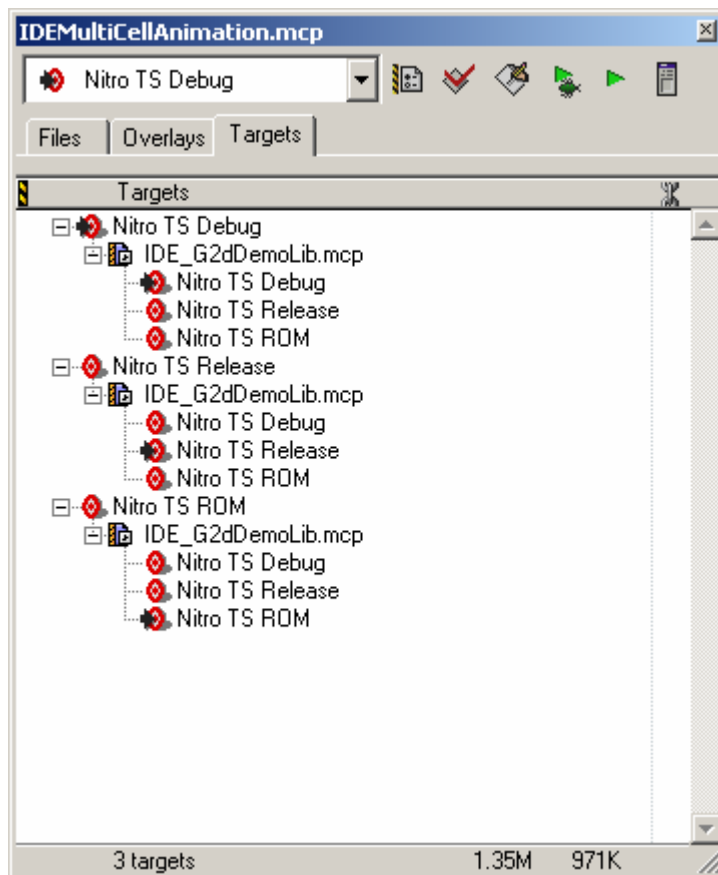
4.3 Subprojects

In Figure 4-2, the g2d demo library project is being added to the MultiCellAnimation project as a subproject. Doing so this way will trigger the demo library to be built first when building the MultiCellAnimation project if the g2d demo library must be built.

Adding the g2d demo library project is possible with the same method as adding files to a project. In order to link and build, it will be necessary to configure the target after adding it. The target settings are configured in the project's target tab.

When building each target of the sample, configure which targets of the demo library to build in the target tab. Normally, when building the **Nitro TS Debug** target of the sample, it is okay to build the **Nitro TS Debug** target of the demo library, so click on the target icon before the **Nitro TS Debug** target of the demo library and an arrow will be added.

Figure 4-3 Configuration of the Targets



CodeWarrior is a registered trademark of FreeScale Semiconductor, Inc.

All other company names and product names in this document are the registered trademarks or trademarks of those companies.

© 2007 Nintendo

The contents of this document cannot be duplicated, copied, reprinted, transferred, distributed or loaned in whole or in part without the prior approval of Nintendo.