

CH10_001

EXERCISE

COMPILING TORQUE IN WINDOWS

Exercise Files

Starter – n/a

Answers – n/a

Exercise Mission

n/a

Special Setup

If you have not already done so, please install the the engine SDK (with source code). In this exercise, I will assume that the SDK is installed in "C:\Torque" and that the engine code is in the respective directory "C:\Torque\engine". If this is not true, please adjust my instructions to match your directory choices.

Synopsis

In this exercise, we will walk through the basic steps that are required to compile the engine using Visual Studio .NET 2003 (Visual Studio version 7).

These instructions are specific to the .NET 2003 version of Visual Studio, but can be modified to match version 8 also. I will attempt to give tips to help version 8 users as I progress, but you may need to search for some of the features I mention in this exercise on your own.

Prerequisites

1. [ch1_001.pdf "Using The Kit"](#)

Exercises

1. [Opening The Right Project \(pg 2\)](#)
2. [Selecting Debug/Release Builds \(pg 2\)](#)
3. [Setting Up The Tool Dock \(pg 4\)](#)
4. [Selecting a Build Target \(pg 5\)](#)
5. [Your First Build \(pg 5\)](#)
6. [Getting Your Files In Place \(pg 6\)](#)
7. [Adding New Folders/Files \(pg 7\)](#)

COMPILING TORQUE IN WINDOWS

1 Opening The Right Project

Visual Studio 7 (.NET 2003) Users

Locate the file "C:\Torque\vc7\Torque SDK.sln" and double-click it. This will open the solution file in Visual Studio 7.

Visual Studio 8 (.NET 2005) Users

Locate the file "C:\Torque\VS2005\Torque SDK.sln" and double-click it. This will open the solution file in Visual Studio 8.

2 Selecting Debug/Release Builds

When Visual Studio finishes loading, you should have something like figure 1 below. I use the word "like" very loosely, because your screen layout is likely to vary from mine. However, if you poke around a bit, you will discover a tabbed pane (on the left or right of the screen) called "Solution Explorer" (shown on left in figure 1 below).

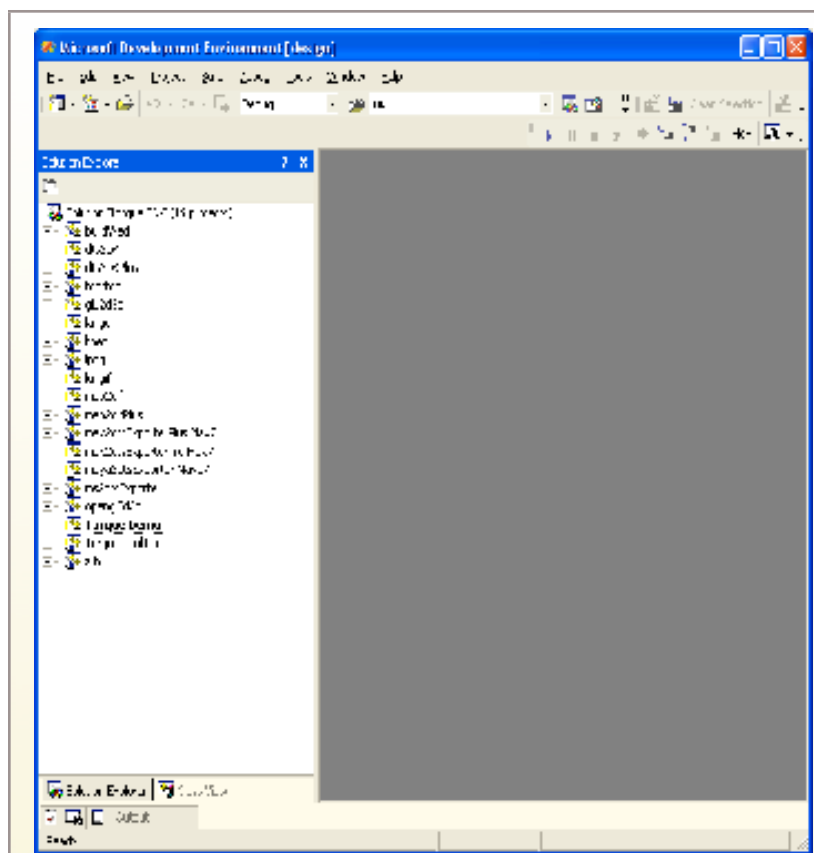


FIGURE 1. THE ENGINE SOLUTION IS OPEN

COMPILING TORQUE IN WINDOWS

I like my "Solution Explorer" on the left, you may like yours on the right. I like mine open all the time, you may want yours closed unless you are accessing it. That is fine, but for now please click the little button at the top of the "Solution Explorer" that looks like a push-pin and be sure the "Solution Explorer" stays open.

Now, please examine the list of projects in the "Solution Explorer" and be sure that "Torque Demo" is highlighted (bold). If it is not, please right click on it and select "Set as Startup Project" from the context-sensitive menu that pops up (figure 2 below).

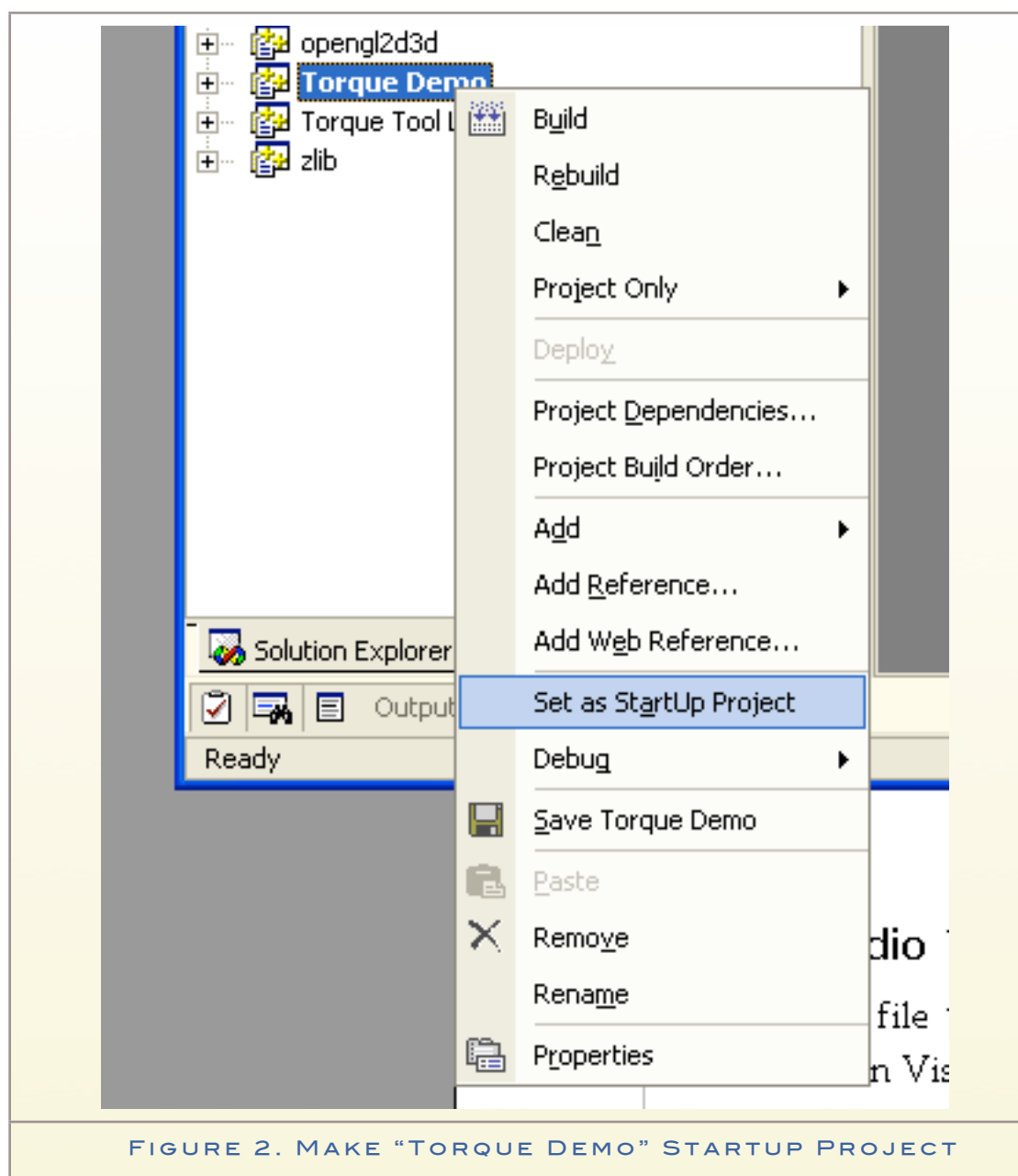


FIGURE 2. MAKE "TORQUE DEMO" STARTUP PROJECT

COMPILING TORQUE IN WINDOWS

3 Setting Up The Tool Dock

When you start Visual Studio for the first time it probably won't have all the toolbars you need. To remedy this, right-click anywhere on the top tool dock and be sure that the following options are checked (see figure 3 below).

- Build – This toolbar allows us to switch between build targets (Debug and Release).
- Debug – This toolbar provides quick access buttons to allow us to run, stop, step, step-over, etc.. That is, it provides debugger buttons for when we are using Visual Studio to debug the engine while running.
- Other? – You can add any other toolbars you want and leaving the default options in place is fine for now.

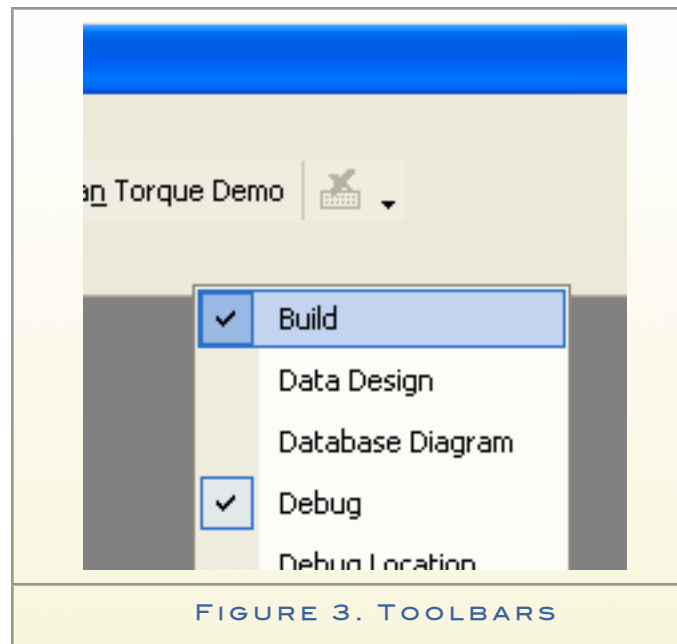


FIGURE 3. TOOLBARS

COMPILING TORQUE IN WINDOWS

4 Selecting a Build Target

On the top tool dock, you should have a drop-down list control like the one in figure 4 below. Be sure that the build target "Debug" is selected from this Solutions Configurations list. This tells Visual Studio to build the "Debug" version of Torque.

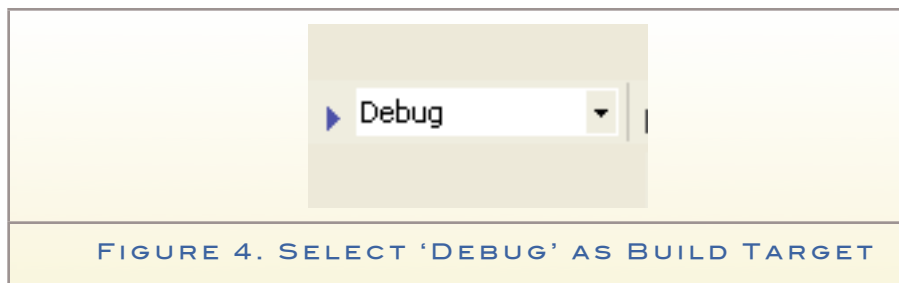


FIGURE 4. SELECT 'DEBUG' AS BUILD TARGET

5 Your First Build

Now that we have the right build-target selected, let's build the engine for the first time. To do this, right-click on "Torque Demo" in the "Solution Explorer" and select "Build" from the context-sensitive menu that pops up (see figure 5 below).

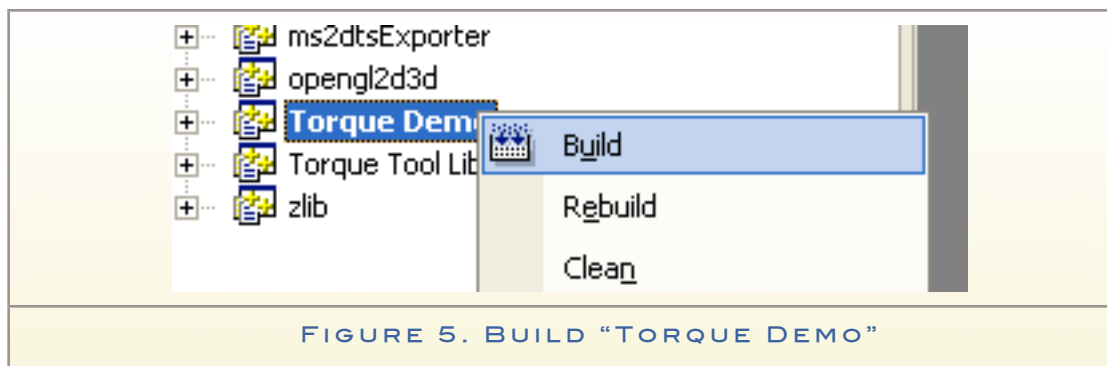


FIGURE 5. BUILD "TORQUE DEMO"

This build will take some time depending on your machine, so you may want to take a break, get some coffee, watch a short show.... Ah! It's done!

COMPILING TORQUE IN WINDOWS

6 Getting Your Files In Place

Before we can add any new files to the "Torque Demo" project, we should first copy them into an appropriate location.

For the purpose of this exercise, I suggest that you locate the "Torque/engine/game" directory and create a new subdirectory named "myFiles".

Now, copy the file "gpgt/engine/answers/chapter10/exer_004.cc" (from the location you installed the accompanying disk) into the newly created directory.

When you are done, you should have a new directory and file:

```
"Torque/engine/game/myFiles/exer_004.c"
```

COMPILING TORQUE IN WINDOWS

7 Adding New Folders/Files

Now that we have a new file, let's add it to our project.

To add new files to a project ("Torque Demo" is a project), you need to decide where to add them.

Generally, I like to have my directories match my project folders. So, we'll follow that guideline in this example.

In this example, please expand "Torque Demo" so it looks like figure 6 below and right-click on the game directory. Finally, select the "Add -> New Folder" option from the context-sensitive menu that pops up (also shown in figure 6).

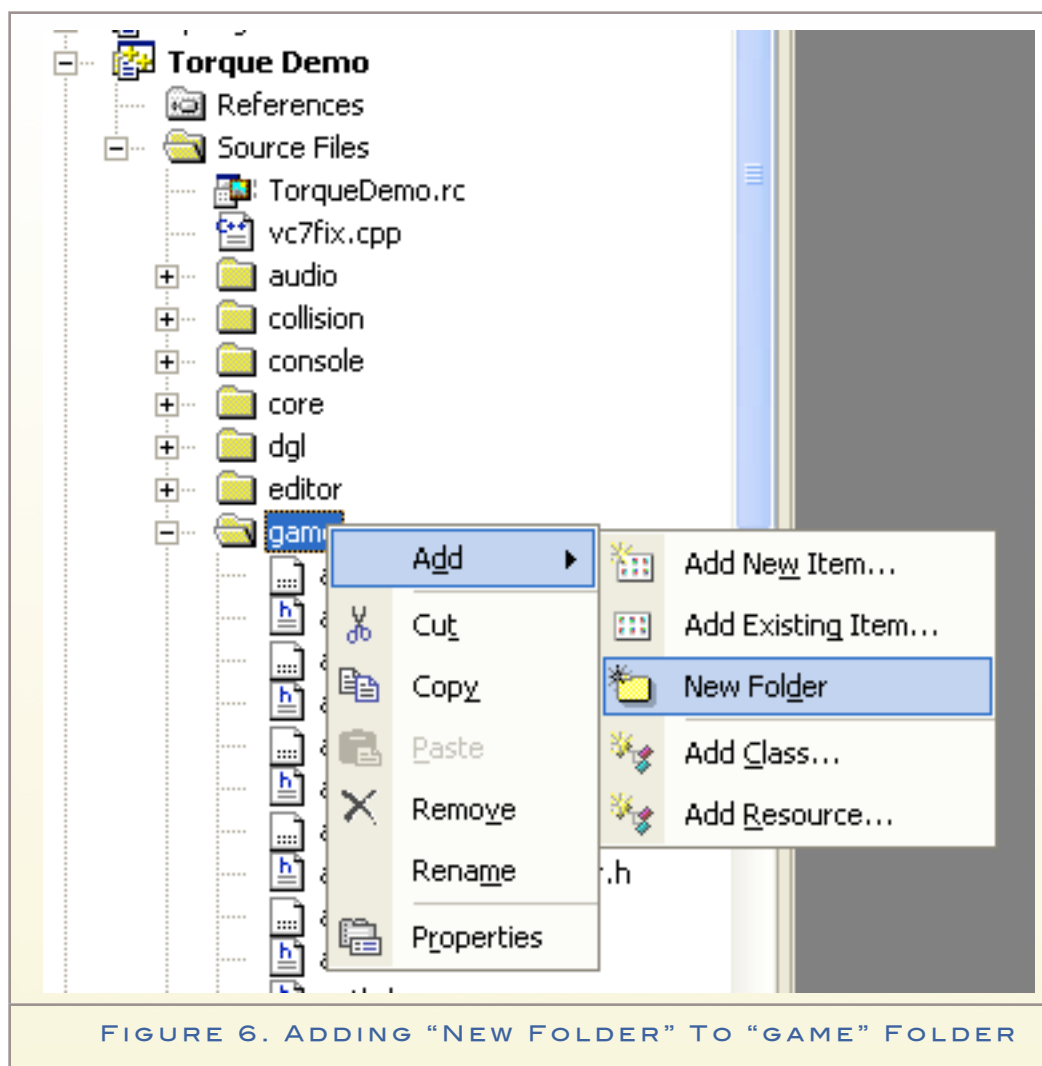


FIGURE 6. ADDING "NEW FOLDER" TO "GAME" FOLDER

COMPILING TORQUE IN WINDOWS

Now, name the folder "myFiles". It may be a bit confusing, but this new folder will show up all the way at the bottom of the list of files and folders for "game".

To add new files to this folder, we simply locate them on the hard drive. Then drag them using Windows Explorer and drop them directly onto the new folder.

For example, I could locate the file "Torque/engine/game/myFiles/exer_004.c", select it, and then drag-and-drop it into "myFiles" (see figure 7 below).

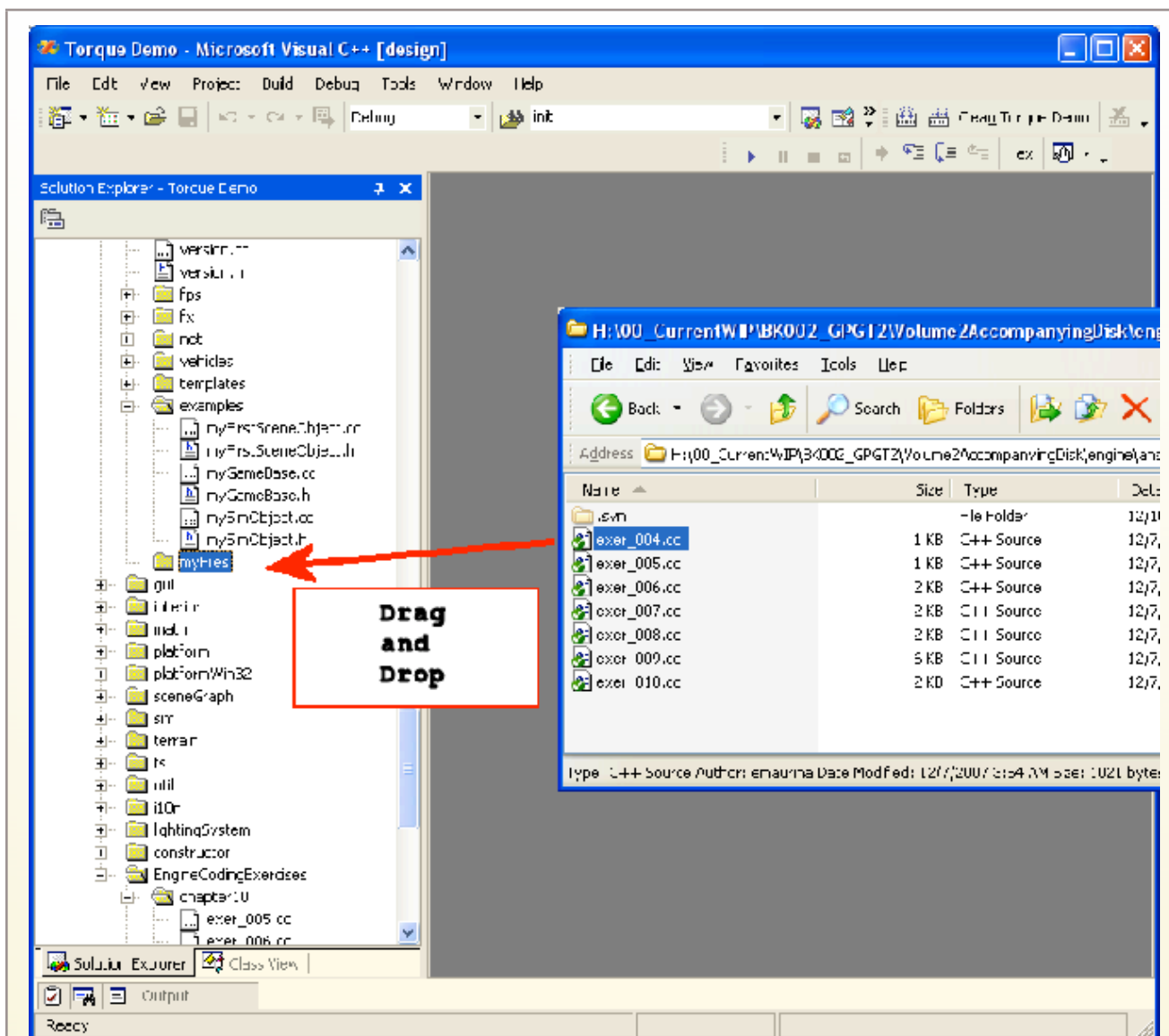


FIGURE 7. DRAG AND DROP FILES DIRECTLY INTO FOLDERS

COMPILING TORQUE IN WINDOWS

Yes, there are other ways to add files, but I find this to be the easiest. So, for now, please try it my way.

When you have done this, the file "exer_004.cc" will appear under "myFiles". Now, you can compile again and the new file should be compiled into the engine.

As you can see, adding new files to the engine and compiling is pretty easy with Visual Studio.