4. Hello World Using The IDE

This chapter is not meant to teach C or C++, but rather to provide a simple "Hello World" program example using the OpenWatcom IDE. The OpenWatcom *Getting Started Guide* contains a "Hands-on Introduction" which covers detailed examples using the IDE, Debugger, and Profiler.

4.1 Writing The Code

This example will utilize MED as the editor and it is assumed the IDE is configured to use MED as the default editor.
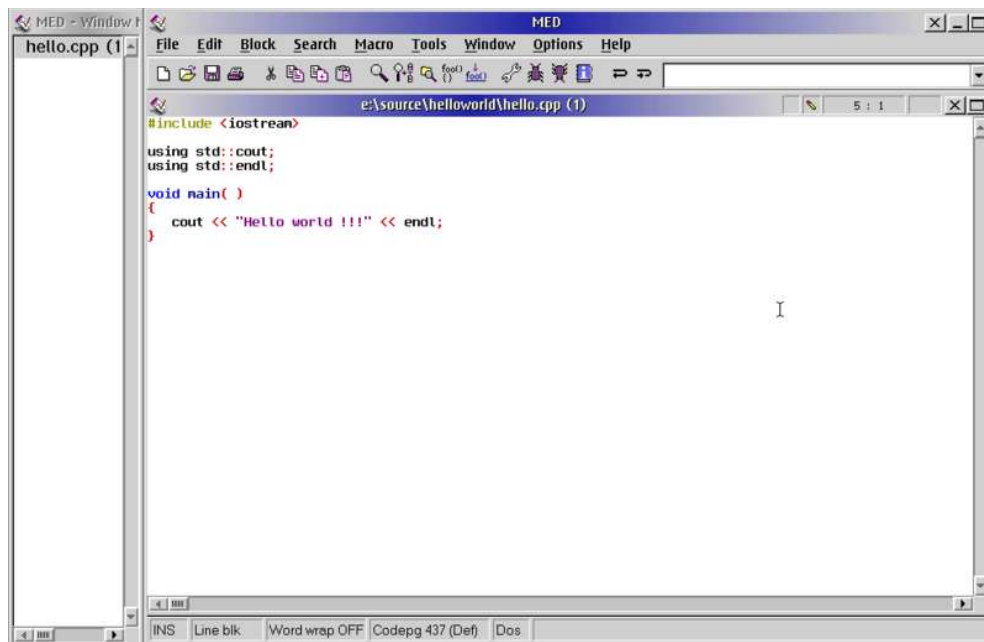
Enter the following in the editor and save as *hello.cpp:*

```
#include <iostream>

using std::cout;
using std::endl;

void main( )
{
   cout << "Hello world !!!" << endl;
}
```
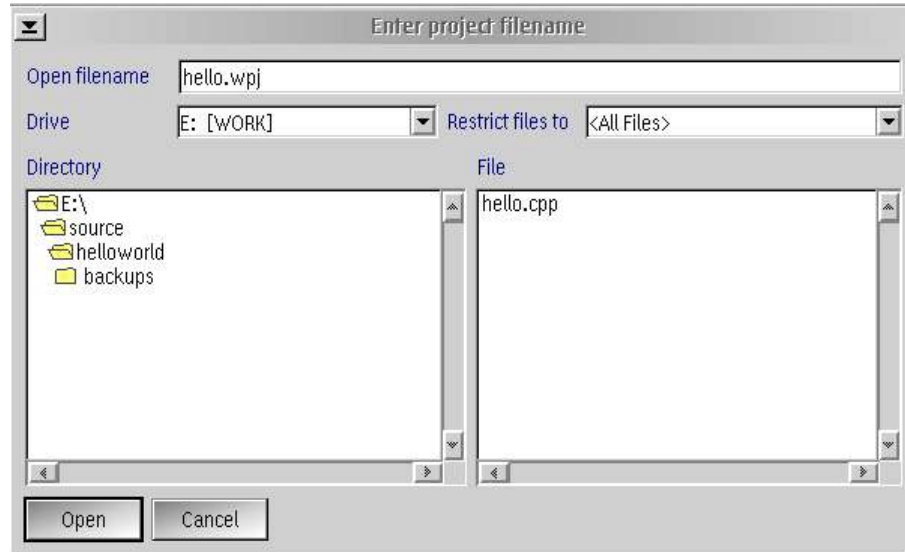
Illustration 4-1 shows the program typed into MED.
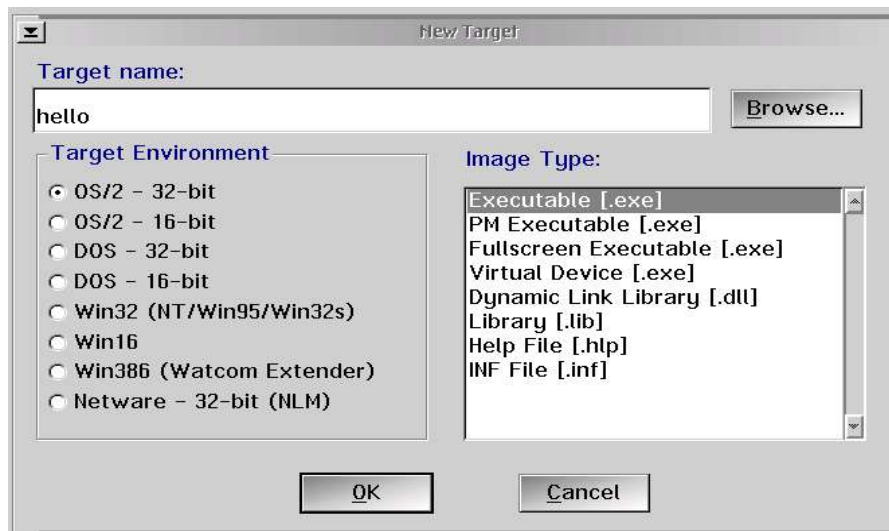

MED Editor Illustration 1

4.2 Setting Up The Project

Start the IDE and select from the menubar **FILE** and then **NEW PROJECT.** Select the directory that *hello.cpp* was saved in, enter *hello.wpj* in the Open Filename, and then select the **OPEN** button. See illustration 4-2 for an example.
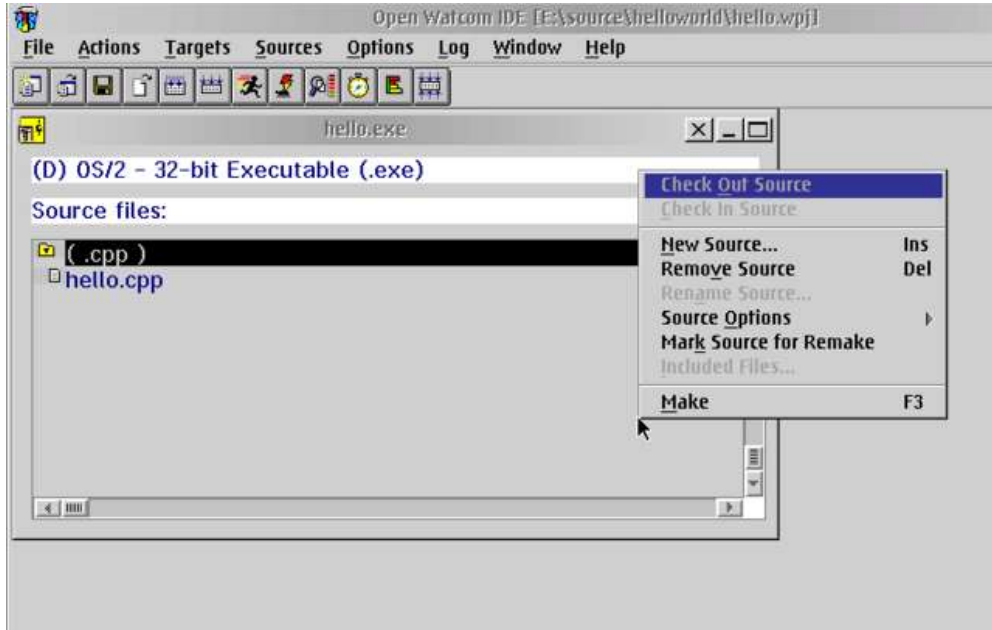
New Project Illustration 2

Do not worry about the next dialog, the default are fine for this simple project. Select **OK**, see illustration 4-3.

New Target Illustration 3

New add the source *hello.cpp* to the project. Right click in the HELLO.EXE window and select **NEW SOURCE.** Browse to *hello.cpp* and select **OK.** If you have compete all the steps correctly, the IDE should look similar to illustration 4-4.
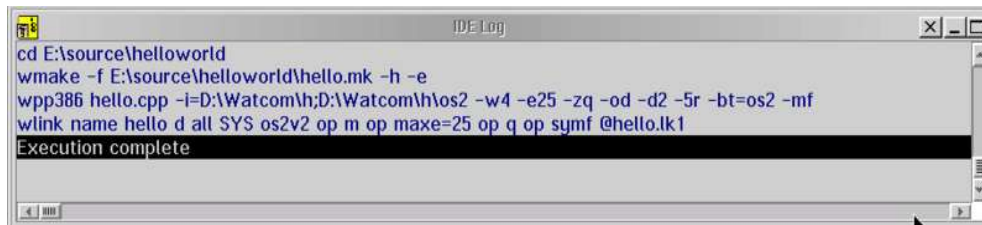
IDE with source file added Illustration 4

Finally, select **SAVE PROJECT** from the **FILE** menu.

4.3 Building The Project

There are several ways to start the make process, but for this exercise right click on the HELLO.EXE window and select **MAKE**. As the project is build the IDE LOG window will should build status as in illustration 4-5.

IDE LOG showing build status Illustration 5

When the build process is complete, you can open a command window and execute hello.exe. As an experiment, load the *hello.cpp* in the editor and change:

    cout << "Hello world !!!" << endl;

to:

    cout < "Hello world !!!" << endl;

Make the project again. In the IDE LOG there will be several error messages. If you have MED configured correctly, click on the error message in the IDE LOG. The MED cursor will reposition to the line in *hello.hpp* where the error originated.