# Instructions for Using ICC11 for Windows v4.0 with the NMI M68HC11 Microprocessor Board

by
Mike Kearny
San José State University
Department of Mechanical and Aerospace Engineering
Mechatronic Engineering Laboratory
January 13, 1997

The New Micros Inc. (NMI) microprocessor board utilizes a custom 68HC11 series microprocessor made by Motorola. ICC11 is a software development tool for use with the 68HC11 series microprocessor. ICC11 allows a user to write the source code using a text editor, download the code to the processor, and communicate with the processor through a terminal window; making ICC11 an Integrated Development Environment. ICC11 will be used exclusively in the Foundations of Mechatronics course (ME 106). The software development package is user friendly and very efficient.

There are 6 steps to developing an application with ICC11:

1. Open or Create a source code file.
2. Edit the source code.
3. Save the source code.
4. Compile the source code into an executable file.
5. Download the executable file to the microprocessor.
6. Execute the program on the microprocessor using the terminal window.

1. Open or Create source code file.

Start ICC11 by clicking on the "ICC11 for Windows" icon in the ICC11 v4.0 program group. Once the program begins you will see a status window. The status window shows the compiler commands and the results of compiling the source code. You cannot type text in the status window, it is only used to give you compilation information. A new source code file is created by selecting File and then New. (Make sure that the name is a valid filename, i.e., 8 alphanumeric characters, no spaces, etc., and that you are working with only one window of ICC11 open). The program prompts you for a name for the new file. The source code files must have a ".c" extension on the file name for ICC11 to edit and compile the source code. An existing source code file is opened by selecting File and then Open.

2. Edit the source code.

When you open a source code file, an editor window will open with the name of the source code file in the title bar. You can then edit the code in this window as required.

3. Save the source code.

You must save the source code before you compile the program. If you do not, then ICC11 will prompt you to save the code before you compile. Make sure the file is saved in C:\ICC11\bin, or else the compiler won't find it. Remember to take home a copy of the source code on a floppy disk at the end of lab. Any files left on the computer after lab will be erased. (You have now been warned!)

4. Compile the source code into an executable file.

The source code is compiled to executable code by selecting Compile and then Compile-to-Executable. The executable file will have the same file name as the source code, but the filename extension will be changed to ".s19". Click in the Status window to check the status of the compiled code. If the code has no errors or "bugs", then you should see a prompt like this:

```
C:\ICC11
icc11 -IC:\ICC11\include -dheap_size:0x0 -btext:0x1400 -lfp -LC:\ICC11\lib -dinit_sp:0x7FFF
TEST.C
.......
Done
```

The word "Done" without any error messages signifies a successful compilation. Since there are no errors, you can proceed to the next step: downloading the executable file to the microprocessor.

If there are bugs in the code, then the prompt could look like this:

```
C:\ICC11
icc11 -IC:\ICC11\include -dheap_size:0x0 -btext:0x1400 -lfp -LC:\ICC11\lib -dinit_sp:0x7FFF
TEST.C
......
!E TEST.C(28): syntax error; found `for' expecting `;'
Done
```

The error message: "!E TEST.C(28): syntax error; found `for' expecting `;' " means there is an error on line number 28. Go back to the code in the editor window, and fix the bug. The number of the line, where the cursor is located, is listed in the lower right hand corner of the ICC11 window. Move the cursor to the line of the error, and fix the error. Keep compiling and fixing the bugs until the source code compiles with no error messages. When the source code is compiled with no errors, then the executable file can be downloaded to the microprocessor.

5. Download the executable file to the microprocessor.

Before downloading the executable file to the NMI board, connect the NMI board to serial port number 1 and apply 5 VDC to the NMI board. Open the terminal window, if it is not already opened, by selecting Target and then Terminal. Press the red reset button on the NMI board. You should see the following prompt:

```
BUFFALO 2.3 - Bit User Fast Friendly Aid to Logical Operation
```

This verifies that the microprocessor board can send data through the serial port to the computer. Press the return key and you should see this prompt:

```
BUFFALO 2.3 - Bit User Fast Friendly Aid to Logical Operation
>
```

This verifies that the microprocessor is receiving data through the serial port from the computer. Now type "load t ", and press the return key. This tells the microprocessor that you are going to download an ASCII text file, which is the executable file. Select the ASCII Download button at the bottom of the Terminal window. Select the executable file you wish to download. If you can see a Bootstrap Download button at the bottom of the screen, then the terminal is in Bootstrap Download mode. Turn the Bootstrap Download mode off by selecting Target and then

unchecking Bootstrap Download. Once you select a file to download a status bar will appear, which displays the progress of downloading the file.

6. Execute the program in the microprocessor using the terminal window.

After the executable file is downloaded to the microprocessor board the microprocessor must be instructed to execute the code. This is done by typing “go 1400”, and pressing the return key. Any text the program prints using the printf command will appear in the terminal window. Keep in mind that the terminal window only displays the text, which the microprocessor sends to the computer. When you type in the terminal window the characters are sent to the NMI board, which then echoes back the characters to the computer through the serial port. The computer then prints the received characters in the terminal window.