

Getting Started with CCS C in MPLAB® IDE

June 2005



Custom Computer Services, Inc.

P.O. Box 2452 • Brookfield, Wisconsin 53186 USA

Sales: 262-522-6500 ext. 35 • Tech Support: 262-522-6500 ext. 32

Copyright © 2005 Custom Computer Services, Inc.

All rights reserved worldwide. No part of this work may be reproduced or copied in any form by any means—electronic, graphic or mechanical, including photocopying, recording, taping or information retrieval systems—without written permission.

PIC®, PICmicro® and MPLAB® are registered trademarks of Microchip Technology Inc. in the USA and in other countries.



Custom Computer Services, Inc. proudly supports the Microchip brand with highly optimized C compilers and embedded software development tools.



MPLAB® IDE INTEGRATION

CCS Plug-In Installation

- ❑ This plug-in allows MPLAB® 6.x/7.x IDE users to use CCS C Compiler to build their projects. The plug-in can be downloaded and installed from:

<http://www.ccsinfo.com/mplab-ccs-plugin-install.exe>

Note: This plug-in works with CCS C Compiler versions 3.168 and up.

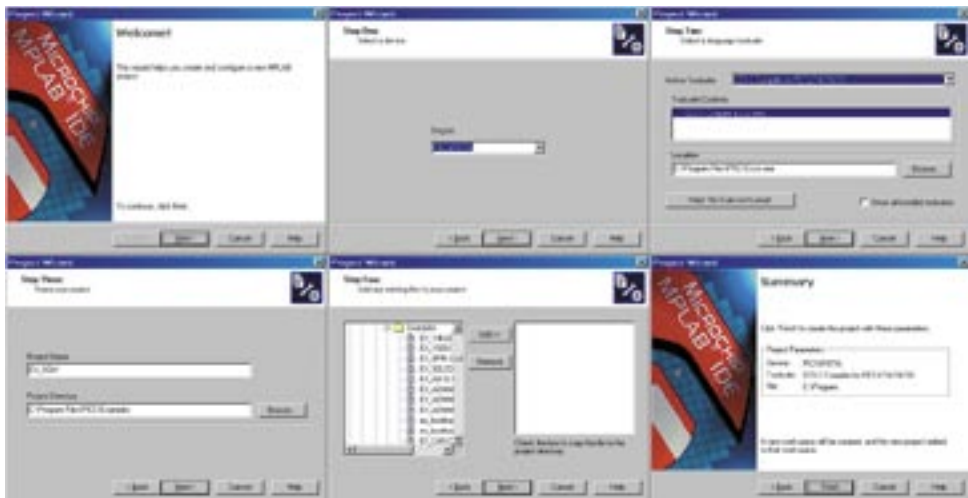
MPLAB® IDE Installation

- ❑ You can download and install MPLAB® IDE from Microchip's website (www.microchip.com). MPLAB® IDE and additional plug-ins may be installed in any order. Use only MPLAB® IDE versions 6.40 and up.

Getting Started

Starting a MPLAB® IDE Project From an Existing Source

- ❑ To start a new project, you can use either the Wizard or create it manually. To create a CCS project using the Wizard, follow these steps:



- ❑ Select the “Project” Menu.
- ❑ Choose “Project Wizard” to make the Project Wizard window appear.
- ❑ Click “Next” on the Welcome Page.
- ❑ Step One, choose “PIC16F877A” from the drop-down list and click “Next.”

- ❑ Step Two, select “CCS C Compiler for PIC12/14/16/18” from the drop-down list. Click “Next.”
Note: If the plug-in is not listed, make sure that “Show all installed tool suite” is checked. If it still doesn’t appear, then the CCS Plug-in may not be installed correctly. You will need to re-install.
- ❑ Step Three, type the project name “EX_SQW” and type or select the project directory “C:\Program Files\PICC\Examples.” Click “Next.”
- ❑ Step Four, select file “Ex_SQW.C.” Click “Add.” Click “Next.”
- ❑ View Summary, click “Finish.”

Start a New Project Manually

```
#include<16F877a.h>
#fuses HS, NOLVP, NOWDT, NOPROTECT
#use delay(clock=20000000)

void main() {
    int8 a, b, c;
    a = 2;
    b = 3;

    c = a + b;
    c = a * b;
    c = b - a;
}
```

- ❑ Enter the following program into MPLAB® IDE as “EX8.c” in the “Exercises Directory.”
- ❑ Go to Project>New.
- ❑ Name the project “EX8” and choose the “Exercises Directory.”
- ❑ Click “Add File to the Project” in the “Project” menu and select the “ex8.c” file.
- ❑ Click “Select Language Tool Suite” and make sure “CCS C Compiler for PIC12/14/16/18” is selected.
- ❑ Go to Configure>Select Device and select “PIC16F877A,” then click “OK.” MPLAB® IDE will then choose the right CCS C Compiler and options for the device.



MPLAB® INTEGRATION (CONT.)

Compiling

- ☐ Go to Project>Compile.
- ☐ If the output window is open, the status of the compiler is displayed here.
- ☐ You can view the CCS C Compiler options that MPLAB® IDE is using by going to Project>Build Options>Project. If you wish to change these options, click on the “Use Alternative Settings” checkbox to edit the options that you want.
- ☐ The output window will show if the code contains any warnings or errors. The line number, which document, and a brief description of the error will be listed for each error or warning. To correct an error or warning go to the line of the specified document and check where the error or warning description occurred on that line. Go through each error and warning to get the project to successfully compile.

Debugging



- ☐ Follow the steps of the MPLAB® ICD 2 USB Device Driver First Time Installation document found in “C:\Program Files\Microchip\MPLAB IDE\ICD2\Drivers\ddicd2.htm.” This will install the MPLAB® ICD 2 device drivers. Be sure to install the drivers for the MPLAB® ICD 2 before connecting the device to the PC via a USB cable.
- ☐ Connect the MPLAB® ICD 2 the PC to the target board, and power up the target board.
- ☐ Go to Debugger>Select Tool>MPLAB® ICD 2.

- ☐ The MPLAB® ICD 2 Setup Wizard will open or can be opened by going to Debugger>MPLAB® ICD 2 Setup Wizard. Click “Next” in the Welcome Page.
- ☐ Step One, select “USB.”
- ☐ Step Two, check “Target Has Own Power Supply.” Click “Next.”
- ☐ Step Three, select “MPLAB® IDE Automatically Connects to the MPLAB® ICD 2.” Click “Next.”
- ☐ Step Four, select “MPLAB® ICD 2 Automatically Downloads the Required Operating System.” Click “Next.”
- ☐ View Summary, click “Finish.”
- ☐ Go to Debugger>Connect. This makes sure that the MPLAB® ICD 2 is ready.
- ☐ Click Debugger>Program. This will download the program to the target board.
- ☐ Once the program is downloaded you can control the operation of the program with the buttons on the toolbar. You can just run the program or step through the program one line of code at a time. You can also step into function calls.
- ☐ Go to View>Watch.



- ☐ In the Watch Window use the drop-down box on the right to choose “a” and click “Add Symbol.” Do this for both “b” and “c.”
- ☐ Now step through the program using the “Step Over” button on the toolbar and watch the values change for “a”, “b,” and “c.”
- ☐ A breakpoint can be added to a line of code by double clicking on it.
- ☐ The “View Menu” has several other windows, which can be useful.





APPENDIX A: BUILD OPTIONS

General

List File: Determines the format of the output list file from the compiler. There are three different formats:

Normal CCS Format:	Generate the list file in regular CCS format. This format displays PC address and Assembly instructions.
MPASM™ Assembler Format:	Generate the list file in MPASM™ Assembler list file formats. This format displays PC address, Opcode, line number, and Assembly instructions.
Symbolic Format:	This is the same as Normal CCS Format, but displays symbols instead of addresses in the list file. This format is more readable than the others.

Debug: Determines the debugging output file.

None:	Generates a regular COD file without any additional debugging information.
Expanded COD format:	Generates a COD file with expanded CCS debugging information. The expanded information is used inside the CCS PCW IDE Compiler, but not in MPLAB® IDE.
IDE. COFF format:	Generates a COFF file. This format is the preferable format to use in MPLAB® IDE.

Other Files: Specify which files that the compiler generates.

Call Tree:	Create call tree. This file contains a tree of the calls in the code.
Statistics File:	Create a file with statistics information about the memory usage and segments.
Symbol File:	Create a file with all the symbols and addresses. It also includes some compiler settings.
Show Warnings:	Force the compiler to generate warnings when compiling the source code.

Device Family: To select the device family, use [Configure->Select Device](#).

Compile for use with ICD: Generates code that is ready to run with ICD debugger. This is similar to putting #device ICD=TRUE in the source code.

Optimization

Set the optimization level for the compiler. A higher optimization level produces more efficient code.

Extra Defines

This category allows defining additional global defines that are going to be used when compiling the code.

- | | |
|---------------------------|--|
| Adding a Define: | Put the define name in the ID field. Put the value in the Value field. Press the “+” button to add the define. |
| Modifying a Define: | Select a definition from the list box. Change the “ID” field and/or the “Value” field, and click “Modify.” |
| Removing a Define: | Select a definition from the list box. Press the “-” button to remove it. |
| Removing all Definitions: | Click “Remove All” to remove all the definitions in the list box. |