



Getting Started with the Student Learning Kit

Featuring the Freescale HCS12 Microcontroller Application Module

1 Introduction

This guide provides the steps necessary to begin development with a HCS12 family microcontroller (MCU) using your application module and/or in conjunction with the Freescale project board (PBMCUSLK).

This guide is separated into three main sections. First, the software setup section will assist in installing the CodeWarrior™ development tools. Second, the hardware setup section will step through configuring the project board and application module. Third, the development section illustrates how to create, build, and debug an application.

Refer to the project board user guide, application module user guide, and CodeWarrior documentation for more details.

For demonstration applications, source code examples, current documentation and resources; visit www.freescale.com/universityprograms

Contents

1 Introduction	2
2 Unpack	3
3 Software Setup	3
4 Hardware Setup	3
4.1 Configuring the Application Module for independent operation	4
4.2.1 Configuring the Application Module for use with the Project Board.....	6
4.2.3 Installing the Application Module	11
4.3 Windows Operating System Hardware Detection.	12
5 Running the Demonstration Application.....	14
6 Custom Software Development	15
6.1 Creating and Building a Project in CodeWarrior Environment	15
6.2 Debugging your Application.....	19
7 Revision History	22
8 Support	22

2 Unpack

Open the shipping carton and remove the contents. Locate the packaging list and verify items have been received from the packaging list.

Inspect both the project board and the application module for damage that may have occurred during shipping. If damage is found, contact www.freescale.com/support for assistance.

3 Software Setup

The following software installation is a one-time, required procedure per host PC that you intend to use with the project board and/or application module. After completing the software setup on your host PC, you can skip the software section on subsequent uses of your project board and/or application module and begin the hardware setup or development sections.

To install and register CodeWarrior development tools for HCS12(X) Microcontrollers onto a host PC, complete the instructions found inside the CodeWarrior Development Studio for Freescale HCS12(X) Microcontrollers case.

Additional tools and training materials for CodeWarrior are available at www.freescale.com

NOTE

Academic Edition development tools are full-featured and not for commercial development. They may only be licensed to students, faculty, and staff of accredited institutions.

Periodically check <http://www.freescale.com/codewarrior> for updates and patches to your development tools. For questions on CodeWarrior development tools, visit <http://www.freescale.com/support>

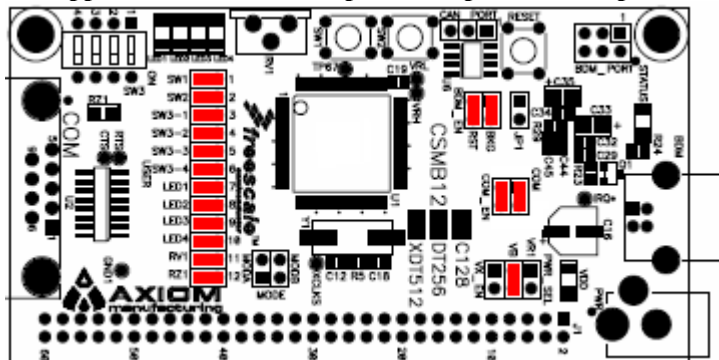
4 Hardware Setup

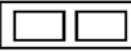



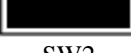
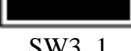

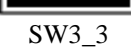
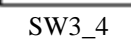
For more information on the Application Module operation, jumper descriptions and settings please refer to the development board User Guide or visit www.freescale.com/universityprograms for latest information.











4.1 Configuring the Application Module for independent operation

- 1) By default, the application module is configured to run independently out of the box. Verify that all the jumper settings are configured properly on the application module as shown by Figure 1.
- 2) Plug in the USB cable from the host PC (running CodeWarrior) to the Application Module.
- 3) Your Done! Proceed to Running your Demonstration application.

Figure 1. Application Module: Independent Operation Jumper Settings



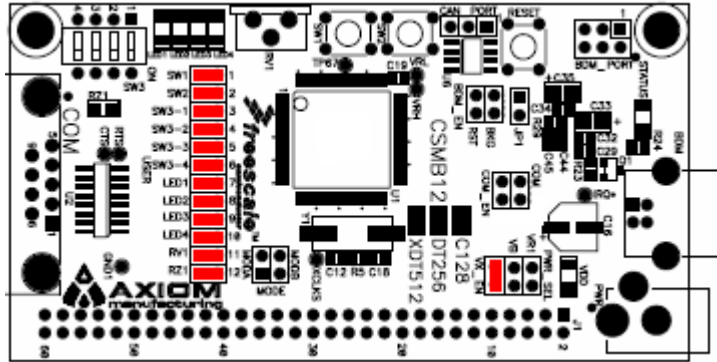
Jumper Position	Jumper ID	Description
 VR1	PWR_SEL	Source power input from connector on-board USB.
 VB		
 VX_EN		
 SW1	USER	Jumpers enable access to application modules features. For push button switch (SW1:2), 4-position DIP switch 3 (SW3_1:4), LED[1:4], and to simplify circuit connections and emphasize software development.
 SW2		
 SW3_1		
 SW3_2		
 SW3_3		
 SW3_4		

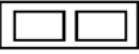
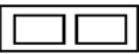







<p>LED1</p>  <p>LED2</p>  <p>LED3</p>  <p>LED4</p>  <p>RV1</p>  <p>RZ1</p> 		
<p>RX</p>  <p>TX</p> 	COM_EN	Jumpers enable the application modules serial communication port.
<p>BKG</p>  <p>RST</p> 	BDM_EN	Jumpers enable the application modules background debug mode (BDM) circuit.







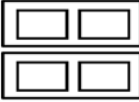
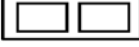
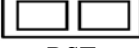
4.2.1 Configuring the Application Module for use with the Project Board

- 1) Move the PWR_SEL jumper FROM “VB” TO “VX_EN”
- 2) Remove all COM_EN jumpers from the Application Module as shown by Figure 2.
- 3) Remove all BDM_EN jumpers from the Application Module as shown by Figure 2.

Figure 2 Application Module: with Project Board Operation Jumper Settings



Jumper Position	Jumper ID	Description
 VR1  VB  VX_EN	PWR_SEL	Source power input from connector J1
 SW1  SW2  SW3_1  SW3_2  SW3_3  SW3_4	USER	Jumpers enable access to application modules features. For push button switch (SW1:2), 4-position DIP switch 3 (SW3_1:4), LED[1:4], and to simplify circuit connections and emphasize software development.

 LED1  LED2  LED3  LED4  RV1 		
	COM_EN	<p>Remove jumpers to disable the application module serial communication port.</p> <p><i>*Note: The COM pins are shared when using the PBMCUSLK with the application module. Conflicts may occur if both are in use simultaneously.</i></p>
 BKG  RST	BDM_EN	<p>Remove jumpers to disable the application modules background debug mode (BDM) circuit.</p> <p><i>*Note: The BDM pins are shared when using the PBMCUSLK with the application module. Conflicts may occur if both are in use simultaneously.</i></p>

4.2.2 Configure the Project Board

Place the project board on a flat and sturdy surface. Ensure sufficient space is available around the project board to safely construct and test prototyped circuits.

Verify that all the jumper settings are configured properly on the project board as shown by Figure 3. Highlighted blocks indicate the on or installed position of jumpers. Circles indicate the location on the project board where the jumpers are located.

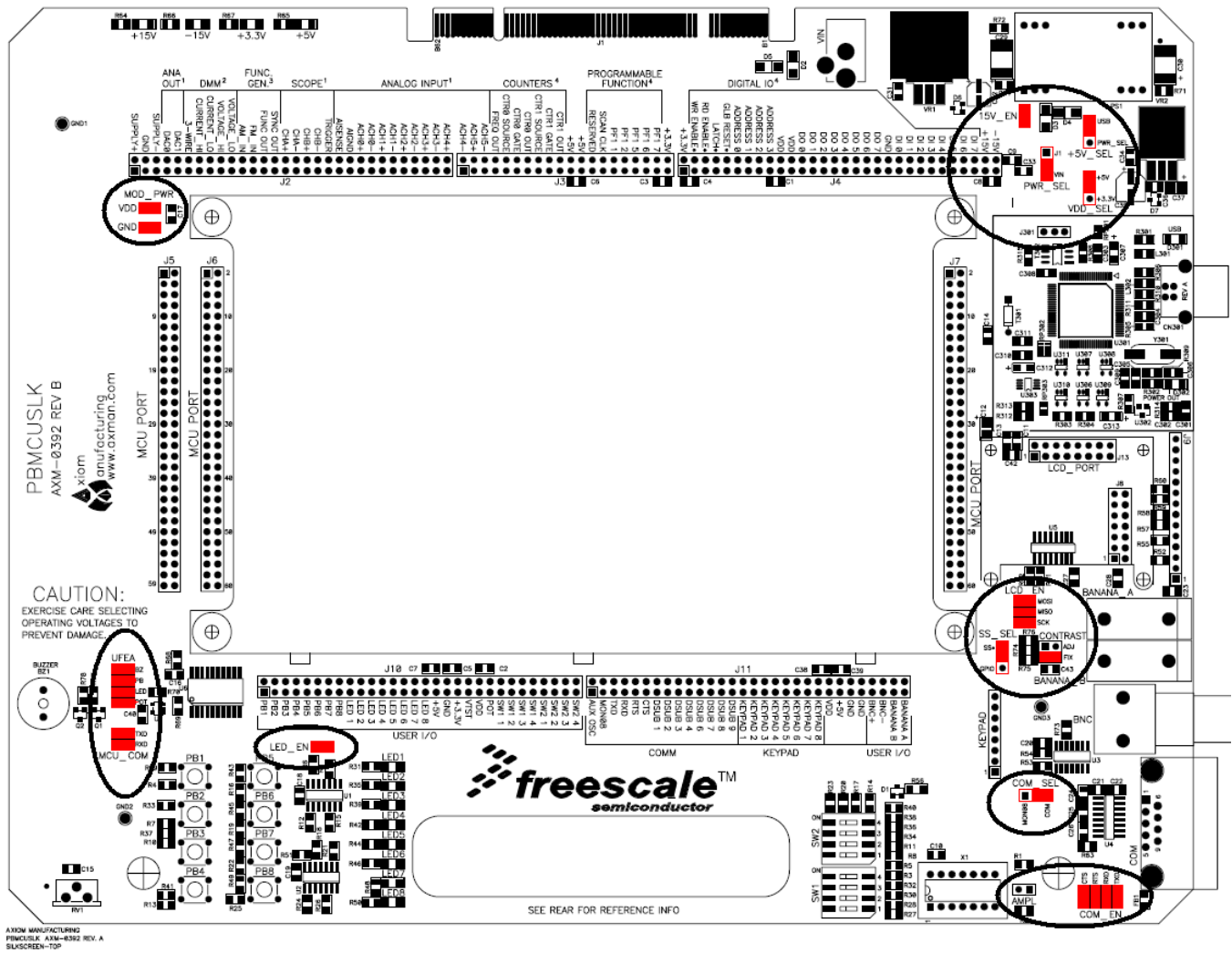




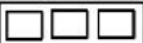






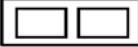
















Figure 3. Project Board Jumper Settings

The table below summarizes the required jumper settings for use with the S12 Application Modules. Please refer to project board user guide for more details and alternative configurations.

Jumper Position	Jumper ID	Description
 J1 VIN	PWR_SEL	Selects V _{in} (wall power transformer) as the default power source for the project board.
 USB PWR_SEL	+5V_SEL	Selects USB (power via USB cable from host PC) as the default power source for the project board. This jumper overrides PWR_SEL in USB position.
 +5V +3.3V	VDD_SEL	Sets V _{DD} to +5V for project board.
 15V_EN	15V_EN	Enables PS1 output (± 15 V supply from dc-dc converter) to the project board. Selects if BDM Multilink pod on the project board has a generic or unique ID associated with the host PC.
 J301	J301	Selects if BDM Multilink pod on the project board has a generic or unique ID associated with the host PC.
 VDD  GND	MOD_PWR	Provides PWR & GND to Application Module
 \overline{SS} GPIO	$\overline{SS_SEL}$	Selects dedicated SS input source to MCU ₂₃ (SS) on MCU PORT connector to transfer data to LCD panel.
 MOSI  MISO	LCD_EN	Connects dedicated SPI signals to LCD port.

 SCK		
 ADJ  FIX	CONTRAST	Selects fixed LCD contrast voltage
 MONO8 COM	COM_SEL	Selects RS-232 communications for project board COM port.
 CTS  RTS  RXD  TXD	COM_EN	Enables all RS-232 communication signals individually for project board COM port.
 AMPL	AMPL	Oscillator output at reduced amplitude – 3.3 Vpp
 LED_EN	LED_EN	Enables all LED outputs on project board.
 BZ  PB 	UFEA	Enables access to direct connect features for buzzer, four push buttons, four LEDs, and potentiometer to MCU PORT connector to simplify circuit connections and emphasize software development.

LED  POT		
 RXD  TXD	MCU_COM	Connects dedicated SCI signals to COM port.

4.2.3 Installing the Application Module

1) Install the application module in the MCU PORT connector (J5) on the project board. Align pin 1 on the application module with pin 1 of the MCU PORT connector on the project board as marked by Figure 4.

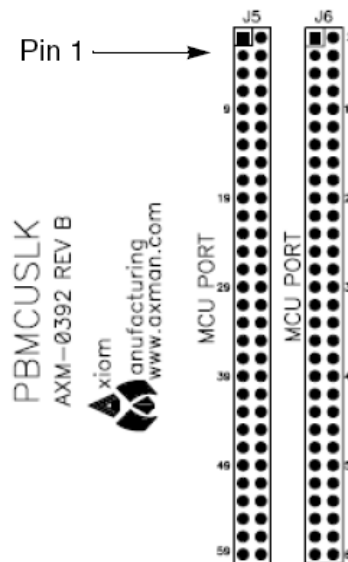


Figure 4. Installation of Application Module

- 2) Power up your host PC and connect one end of the supplied USB cable between an available USB port on your host PC.
- 3) Connect the other end of the USB cable to the USB connector on the project board. Figure 5 illustrates the USB connector on the project board for the BDM Multilink pod interface to power, program, erase, and debug the application module board.

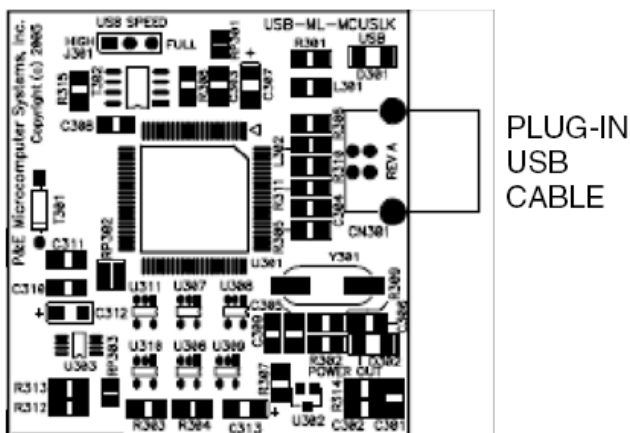


Figure 5. Installation of USB Cable

4.3 Windows Operating System Hardware Detection

NOTE

Steps 1 through 7 are a one-time, required procedure per host PC that you intend to use with the project board and can be skipped on subsequent uses of your project board.

- 1) Once you plug in the USB cable to the hardware Windows will detect the new hardware and launch the “Found New Hardware Wizard”.
- 2) If you have installed the latest version of CodeWarrior prior to this step, simply follow the prompts provided by Windows to automatically detect new hardware as illustrated by Figure 6 and prompt you to install software for USB Multilink 2.0 or Turbo BDM.

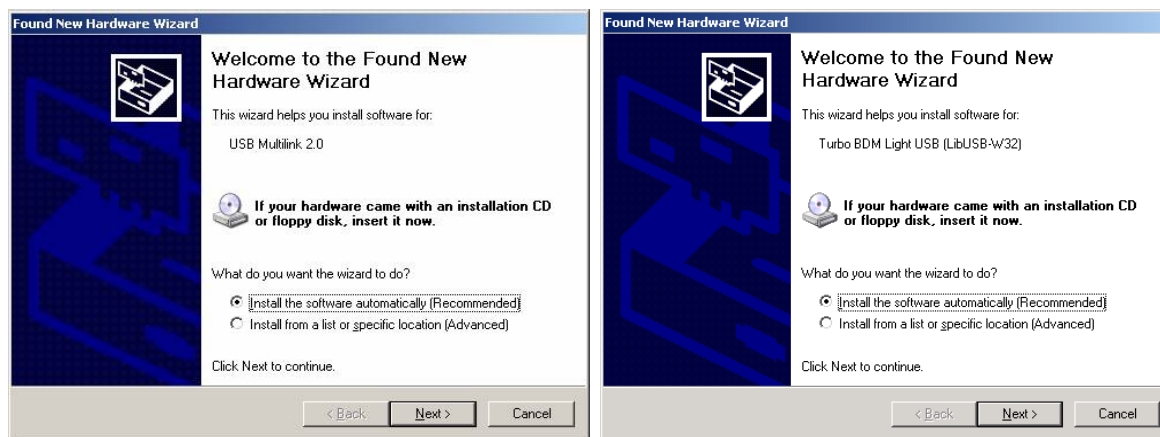


Figure 6. Found New Hardware Wizard (Screen 1)
 Left: Project Board USB BDM Right: Application Module USB BDM

- 3) Select Install the software automatically (recommended) from the Found New Hardware Wizard and click Next to continue with installation.
- 4) Installation will begin with the Found New Hardware Wizard (Screen 2) illustrated by Figure 7. No user action necessary until software is completed installing.

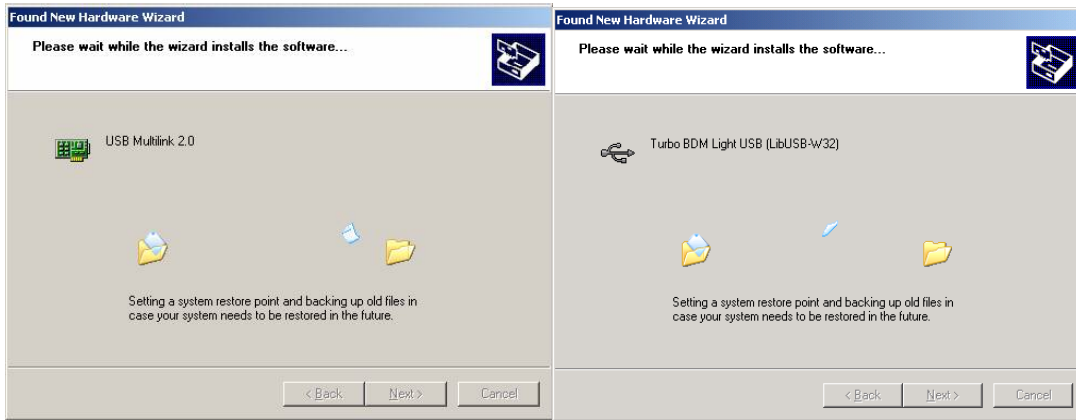


Figure 7. Found New Hardware Wizard (Screen 2)
Left: Project Board USB BDM Right: Application Module USB BDM

5) Successful installation is verified by Found New Hardware (Screen 3) illustrated by Figure 8. Click Finish to complete installation and close the wizard.

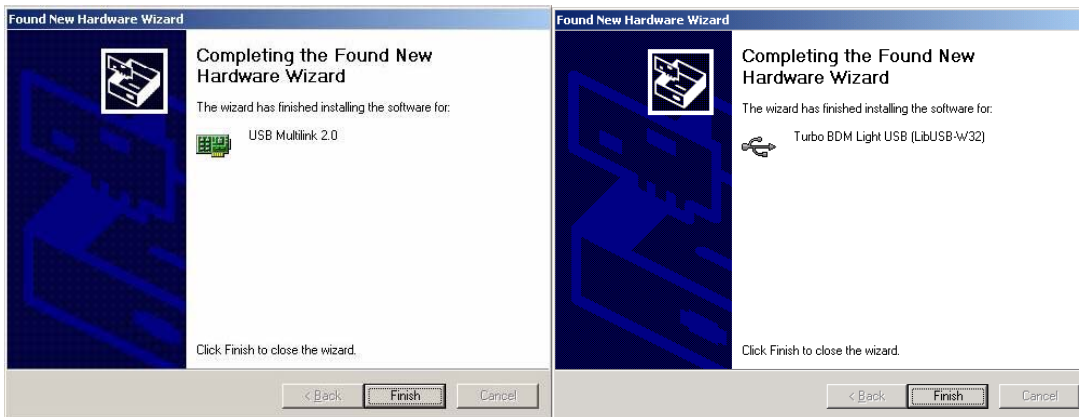


Figure 8. Found New Hardware Wizard (Screen 3)
Left: Project Board USB BDM Right: Application Module USB BDM

6) Validation that the USB is properly installed can be done by verifying that the VDD LEDs, and USB status lights are lit.

7) The hardware is now ready for development and use.

5 Running the Demonstration Application

By correctly following the previous chapters you should now be able to.

- I) Observe LED4 is illuminated
- II) Press PB1 on the application module (or Project Board) to illuminate LED1.
 - i) Project Board users will notice a new message on the LCD Display.
- III) Press PB2 on the application module (or Project Board) to illuminate LED1.
 - i) Project Board users will notice a new message on the LCD Display.
- IV) Observe LED3 blinking at a periodic rate.
- V) Connect a DB9 cable between HOST PC and Application Module (or Project Board) COM connectors to observe serial communications.
 - i) Host PC terminal settings
Baud: 9600, Data Bits: 8, Parity: None, Stop Bits: 1, Flow Control: None
- VI) Available on with Project Board ONLY:
 - i) Press PB3 on the project board and adjust the potentiometer. The buzzer is enabled with a preset frequency and the potentiometer value is sampled to control the volume of the buzzer while the button is pressed. Also, note that while pressed the LCD display is now a new message.

NOTE

This demonstration application in its entirety is available for free at www.freescale.com/universityprograms in the student learning kits section.

Under your Student Learning Kit page you will find:

Complete, commented source code in C
 Detailed function explanation
 Latest up-to-date function additions, features, and updates.

6 Custom Software Development

The sections below walk you through the steps to create a blank project to begin your own programming with.

6.1 Creating and Building a Project in CodeWarrior Environment

- 1) Launch the CodeWarrior IDE
 - a) Select: Start > Programs > Freescale CodeWarrior > CW for HC12 V4.6 ; a menu will appear
 - b) Select: CodeWarrior IDE. IDE will start and a CodeWarrior window will appear

Create a new project

- 2) From the IDE main menu bar, select: File > New; a new window will appear as seen in Figure 9.

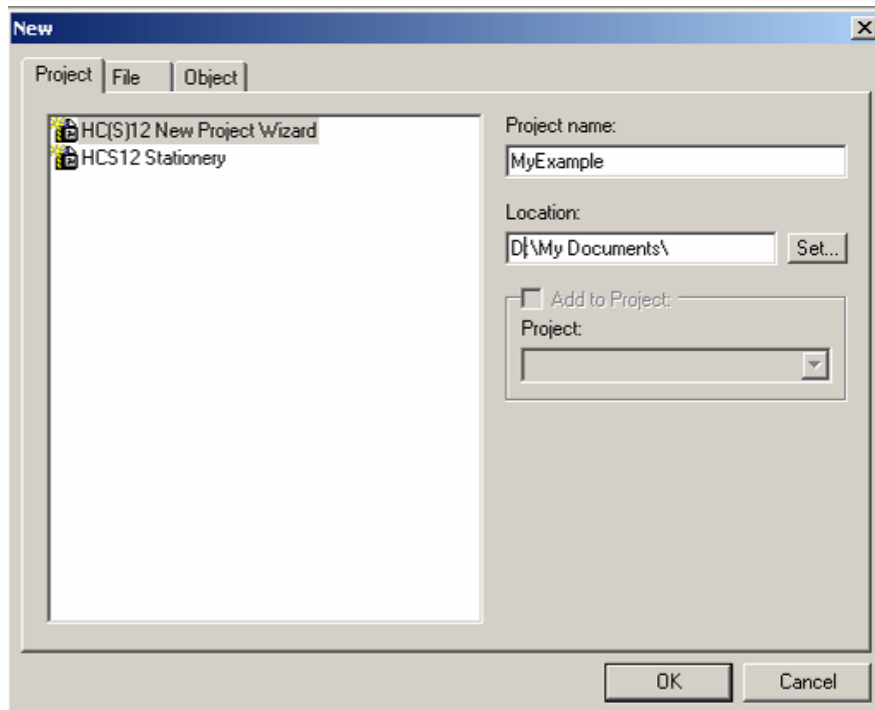


Figure 9. CodeWarrior New Project Window

- a) Select: HC(S)12 New Project Wizard.
- b) In the Project name text box, type the name you want to give the project. IDE automatically adds .mcp extension when it creates project.
- c) In the Location text box, set location where you want the project to be created.
- d) Click OK — the first page of the new project wizard will appear explaining to you how the project wizard works, proceed to the second page by clicking OK again (see Figure 10)

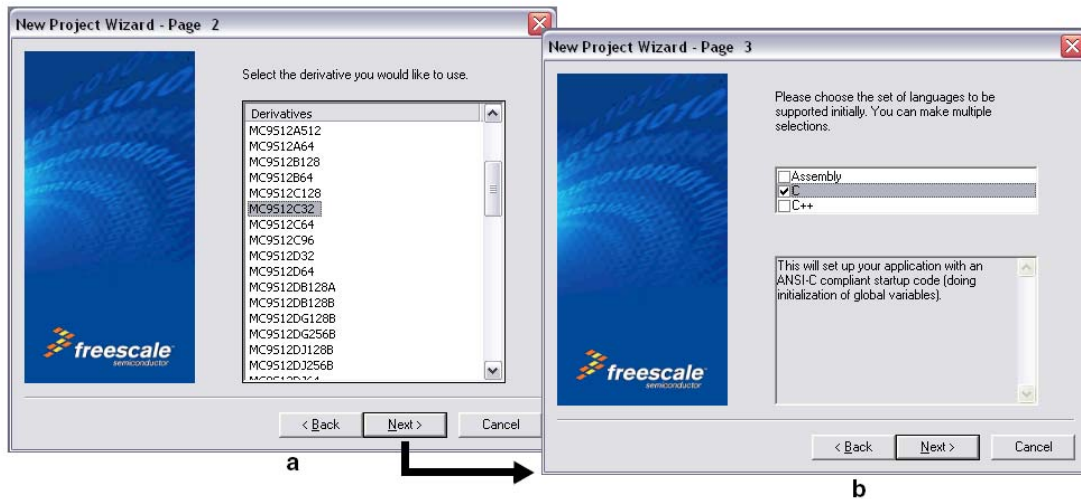


Figure 10. CodeWarrior New Project Wizard - Page 2 and Page 3

- e) Select your microcontroller.
- f) Click Next — page 3 of the new project wizard will appear (see Figure 10).
- g) Make sure C checkbox is marked. If you are using assembly or C++, mark the respective selection.
- h) Click Next — page 4 of the new project wizard will appear (see Figure 11).

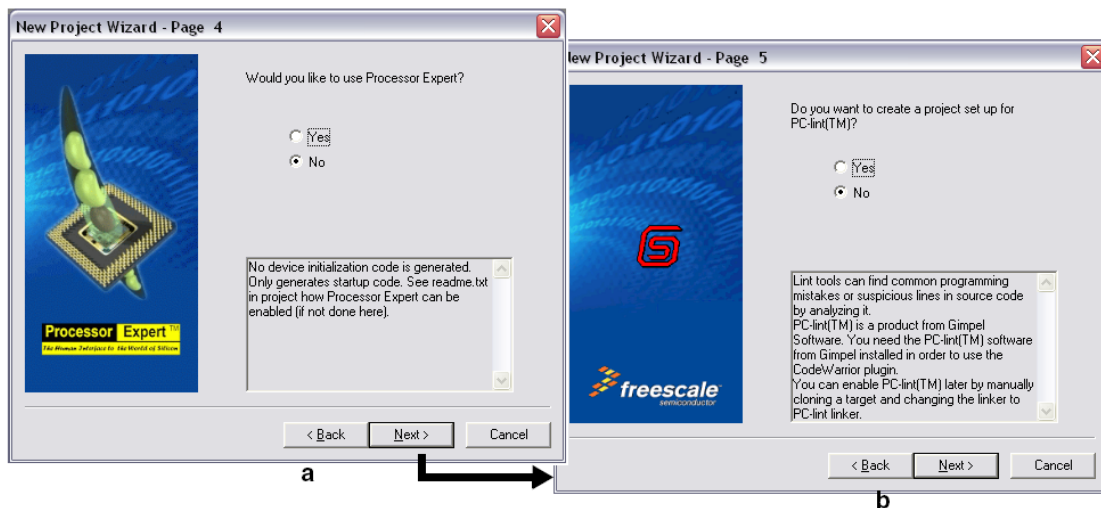


Figure 11. CodeWarrior New Project Wizard Page 4 and Page 5

- i) Select No, you do not want your project configured with Processor Expert.
- j) Click Next — page 5 of the new project wizard will appear (see Figure 11).
- k) Select No, you do not want your project configured to work with PC-lint.
- l) Click Next — page 6 of the new project wizard will appear (see Figure 12).

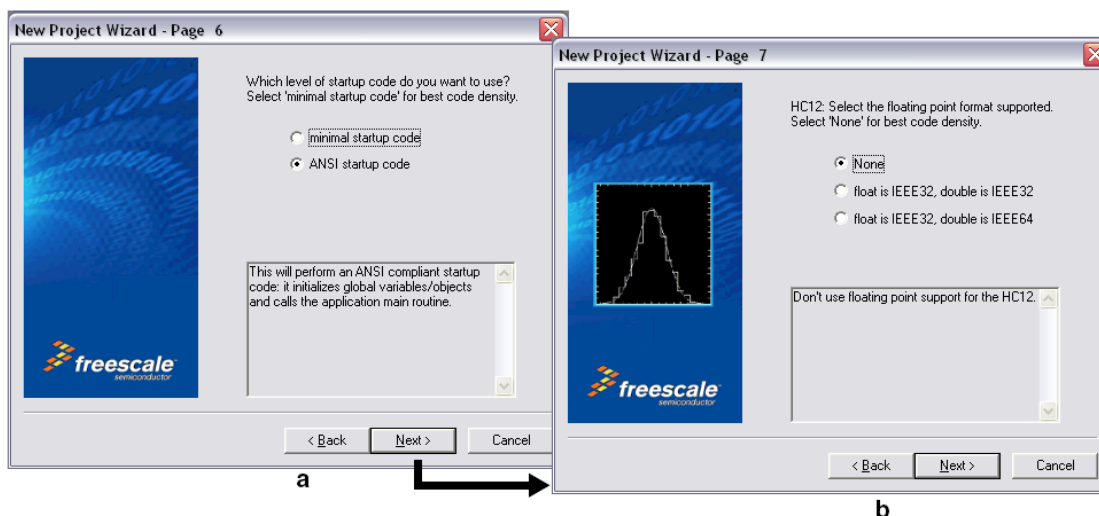


Figure 12. CodeWarrior New Project Wizard Page 6 and Page 7

- m) Select ANSI startup code.
- n) Click Next — page 7 of the new project wizard will appear (see Figure 12).
- o) Select None, you do not want to specify a floating point format.
- p) Click Next — page 8 of the new project wizard will appear (see Figure 13).
- q) Select Small memory model
- r) Click Next — page 9 of the new project wizard will appear (see Figure 13)

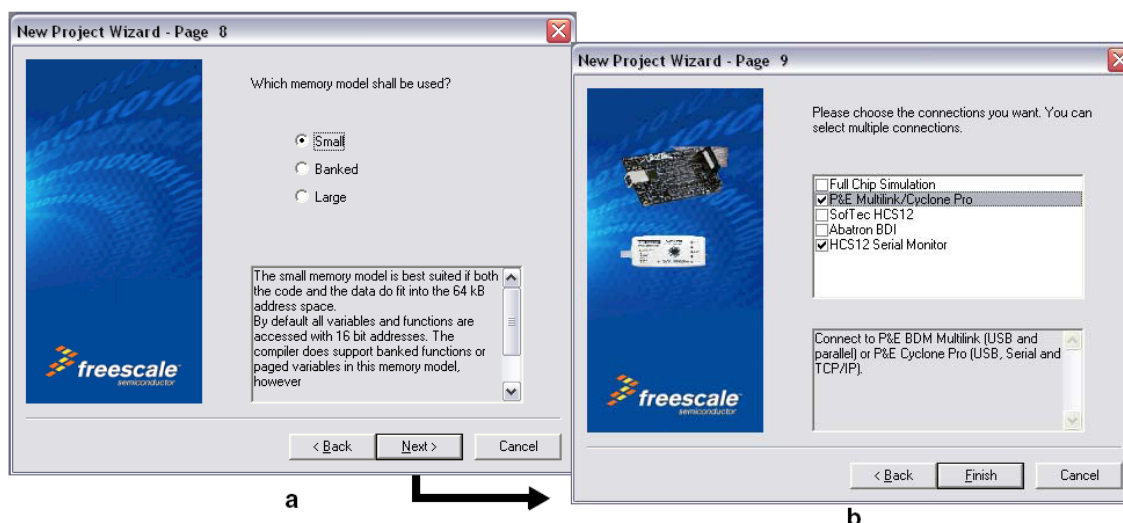


Figure 13. CodeWarrior New Project Wizard Page 8 and Page 9

- s) Page 9 allows you to specify connections that the project should be configured to support.

NOTE

Select as many connections as you would like! Having more than one selection does not impact your compiled code size.

- t) Select P&E Multilink/Cyclone Pro and TBDML. (also, you may want to recreate the pic of page 9 to show the current selections.)
- u) Click Finish — the wizard creates a new project based on information you specified. A project window will appear, docked at left side of main window (see Figure 14).

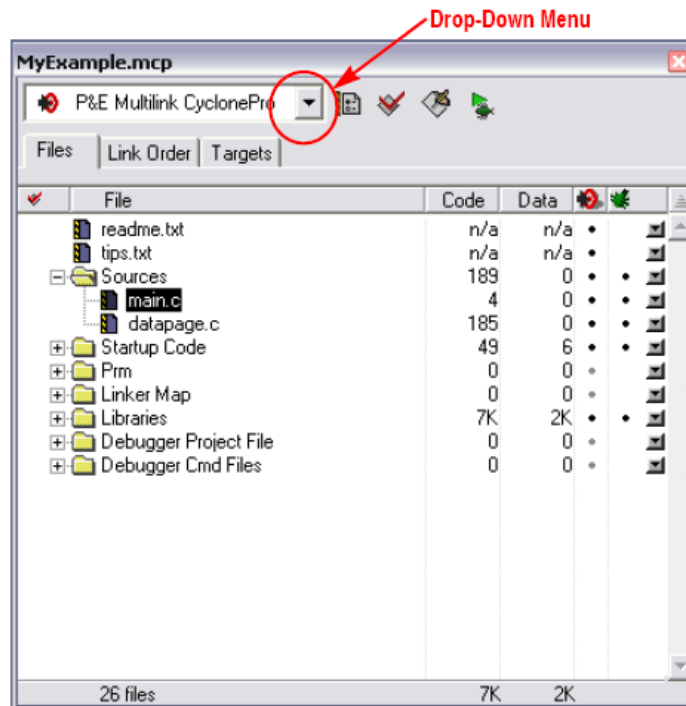


Figure 14. CodeWarrior Project Window

NOTE

To undock the project window, double-click the docking handle (double gray lines at top of the project window). To re-dock window, right click in title bar of project window, and select Docked.

1) Select Build Target.

Your project can contain multiple build targets. For this example, we use the P&E Multilink CyclonePro target, which uses the USB BDM pod on the project board. This BDM pod provides bidirectional communication between the microcontroller and the debugger.

- a) Click the drop-down menu of the project window (see Figure 14).
- b) Select P&E Multilink CyclonePro.

2) Edit source code.

- a) Click the + sign next to the sources folder to reveal files (see Figure 14).
- b) Double-click the main.c file. An editor window will open and display the contents of main.c similar to that shown (see Figure 15)

```

main.c
Path: D:\Profiles\r9aabe\Desktop\MyExample\Sources\main.c

#include <hidef.h>      /* common defines and macros */
#include <mc9s12c32.h>  /* derivative information */
#pragma LINK_INFO DERIVATIVE "mc9s12c32"

void main(void) {
    /* put your cwn code here */
    EnableInterrupts;

    for(;;) {} /* wait forever */
    /* please make sure that you never leave this function */
}

Line 8 Col 3

```

Figure 15. Example Editor Window - main.c

- c) Make changes to the contents of main.c file, if desired.
 - d) If you make changes to main.c, from IDE main menu bar, select File > Save.
- 3) Add files (if appropriate).
- a) Highlight the Sources folder.
 - b) From the IDE main menu bar, select Project; a menu will appear.
 - c) Select Add Files. A dialog box will appear.
 - d) Navigate to the directory that contains the file you want to add.
 - e) Highlight the filename of the file you want to add to your project.
 - f) Click Open — the Add Files dialog box will appear.
 - g) Check the checkbox for each build target to which the file applies.
 - h) Click OK — the Add Files dialog box closes. In the Project window, the filename of added file will appear under the Sources folder.
- 4) Build project.
- a) From the IDE main menu bar, select Project.
 - b) Select Make. IDE builds (assembles, compiles, and links) project. Error and warnings window will open and show error messages and warning messages, if appropriate.

6.2 Debugging your Application

The following steps explain how to establish communication and upload your application software to the application module. This will allow you to debug your application through CodeWarrior True-Time Simulator & Real-Time Debugger using the BDM pod interface on the project board. “Software Setup” and “Hardware Setup” sections described earlier must be completed before executing the steps in this section.

- 1) Make sure power is applied to the project board. If not, go back and complete hardware setup section instructions.
- 2) Start debugger.
 - a) From the main menu bar in CodeWarrior IDE, select Project.
 - b) Select Debug — the Debugger window will open in MultilinkCyclonePro target.
 - c) You will be prompted by a Loader Warning window that the debugger will mass-erase and program the microcontroller with your application. Click OK to continue.

NOTE

If prompted by debugger with a communication setup window, select the appropriate host PC USB port being used by the BDM pod and click OK.

- d) Once the debugger is launched a window similar to Figure 16 will appear. The operation of the various sub windows is listed in Table 1 on the subsequent page. The configuration of sub-windows can be user configured and saved using File->Save Workspace

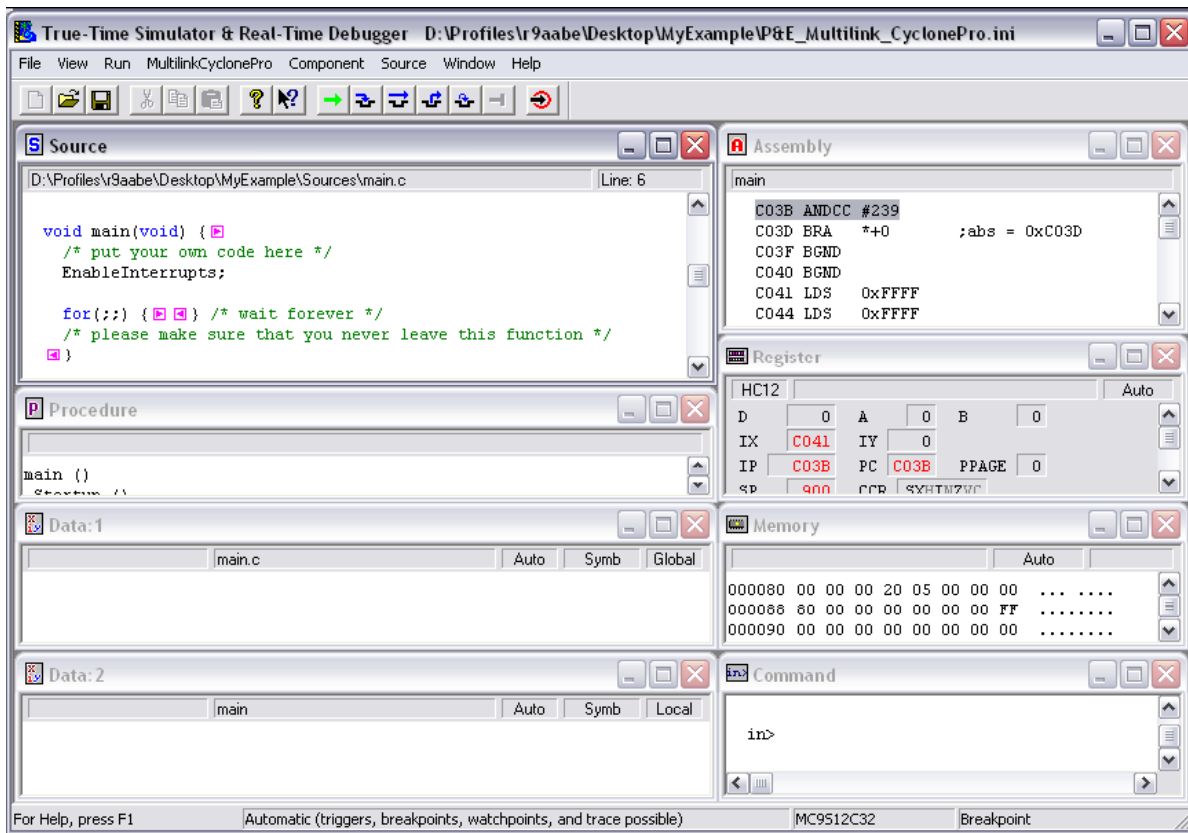



Figure 16. CodeWarrior Debugger Window


Table 1. Debugger Window Descriptions



Window Name	Description
Source and Assembly	Display programmed application software in C and assembly programming languages, respectively
Register	Displays the MCU internal registers
Memory	Displays the active value at each memory address in the MCU internal memory map

Procedure	Displays the active procedure being evaluated
Data 1 and Data 2	Display and global or local variables in your application software and their active value.
Command	Allows user control and log command execution






- 3) Reset the target by clicking on the reset target icon. 
- 4) Right-click anywhere in the Source window and select Open Source File. The Source Files menu will appear. To view between source files in your application software, select the appropriate file in the menu, then click OK.
- 5) Right-click on executable line of source code in Source window to set breakpoints or triggers in program code.
- 6) Run application.
 - a) From debugger main menu, select Run — the Run menu will appear.
 - b) Select Start/continue; program will execute until encountering first breakpoint, if breakpoint is present. Command pane displays program status.

NOTE

Alternatively, you can click on start/continue icon  in the debugger taskbar

- 7) Click start/continue icon, , to resume program code execution (if breakpoint occurred) or click halt icon, , to stop program execution.

NOTE

The debugger taskbar includes other icons for debugging which allow single step , step over , step out , assembly step , and reset target  functions.

- 8) In the debugger window tool bar, select: File > Exit (to exit debugger).
- 9) In the IDE main window tool bar, select: File > Exit (to exit CodeWarrior IDE).

Congratulations — You have successfully developed a project using CodeWarrior development tools with your project board and application module.

7 Revision History

Version	Date	Revised By	Description of Changes
0	9/2007	r2aakl	Initial Revision

8 Support

For easy answers to your questions, and to get the most out of your support experience, Freescale recommends that you use these resources in the following order:

For frequently asked questions (FAQs), latest updates, and listing of known bugs and solutions.	www.freescale.com/universityprograms OR www.axman.com/support
Ask your questions to our online developer community	www.freescale.com/forums
Ask your questions directly to Freescale representative	www.freescale.com/support

How to Reach Us:

Home Page:

www.freescale.com

E-mail:

support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:

Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Design and/or Manufacturing services for this product provided by:



Axiom Manufacturing
2813 Industrial Lane
Garland, Tx. 75041
Phone: 972-926-9303
Web: www.axman.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc.

All other product or service names are the property of their respective owners.