

Getting Started with the Student Learning Kit Application Module

Using the MC9S12DT256 Microcontroller Application Module

For use with the following part numbers:

**CSM12D with MC9S12DT256 chipset
APS12DT256SLK**

1 Introduction

This guide provides the steps necessary to begin development with a HCS12 family microcontroller (MCU) using ONLY your application module. *For a Quick Start Guide in using the application module in conjunction with the Freescale project board (PBMCUSLK) refer to the “Getting Started with Freescale Project Board using the MC9S12DT256 Application Module”.*

This guide is separated into four 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. Fourth, the demonstration application takes the knowledge learned in step 1-3 and provides a simple example that exercises the many user-accessible features on the project board.

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

Visit www.freescale.com/universityprograms for latest updates, current information and more resources.

Contents

1 Introduction	2
2 Unpack	3
3 Software Setup	3
4 Hardware Setup	4
4.1 Configure the Application Module	4
4.2 Cable Connections	5
5 Development	6
5.1 Creating and Building a Project in CodeWarrior Environment	6
5.2 Debugging your Application	11
6 Running the Demonstration Application	13
7 Revision History	15

2 Unpack

Open the shipping carton and remove the contents. Locate the packaging list document 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 the manufacturer at support@axman.com 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 HC(S)12 MCUs onto a host PC, complete the instructions outlined in sections A of the CodeWarrior Development Studio for Freescale 68HC12/HCS12/HCS12X/XGATE Microcontrollers quick start pamphlet found inside the supplied case labeled CodeWarrior Development Studio for HCS12(X) Microcontrollers Academic Edition.

Additional tools and training materials for CodeWarrior are available at the Freescale website.

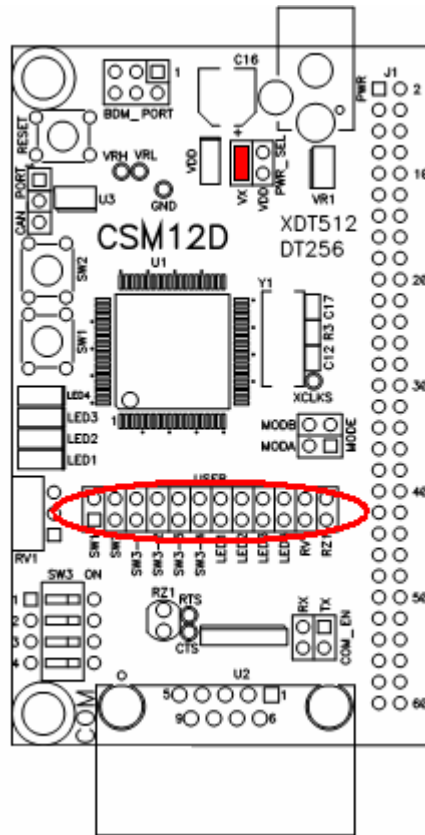
NOTE

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

















4.1 Configure the Application Module



1. Application Module Jumper Locations

- 1) Verify that all the jumper settings are configured properly on the application module as shown by **Error! Reference source not found..** Highlighted blocks indicate the on or installed position of jumpers. Table 1 summarizes the default state for the jumper settings. Please refer to application module user guide for more details and alternative configurations.

Table 1. Application Module Jumper Settings

Jumper Position	Jumper ID	Description
 Vdd  VX	PWR_SEL	Source power input from barrel connector.
 SW1  SW2  SW3_1  SW3_2  SW3_3  SW3_4  LED1  LED2  LED3  LED4  RV1  RZ1	USER	Install Jumpers (<i>default</i>) to 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.
 RX  TX	COM_EN	Install jumpers (<i>default</i>) to enable the application modules serial communication port.

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

4.2 Cable Connections

- 1) Plug your power supply into the wall socket, and the other end into the application module barrel connector.
- 1a) Verify that the VDD LED on the application module board is lit.
- 2) Plug your DB9 serial cable into the application module COM port and then to a unused COM port on the PC.

The application module is now ready for use and by default is running a program which will be explained in the following sections.

5 Development

5.1 Creating and Building a Project in CodeWarrior Environment

After installing and registering the CodeWarrior development tools as described in Section 2, “Software Setup,” and completing the project board and application module configuration as described in Section 3, “Hardware Setup,” you can now begin to develop your application for a target microcontroller. This section will create and build a project under the CodeWarrior for HCS12(X) development tool platform.

NOTE

The instructions below are slightly different from the steps described in sections B and D of the quick start pamphlet included in the CodeWarrior for HCS12(X) development tools CD case. The instructions below are adjusted for the MCUSLK and MC9S12DT256 application module.

- 1) Launch the CodeWarrior IDE
 - a) Select: Start > Programs > Freescale CodeWarrior > CW for HC12 V4.5; 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 2.

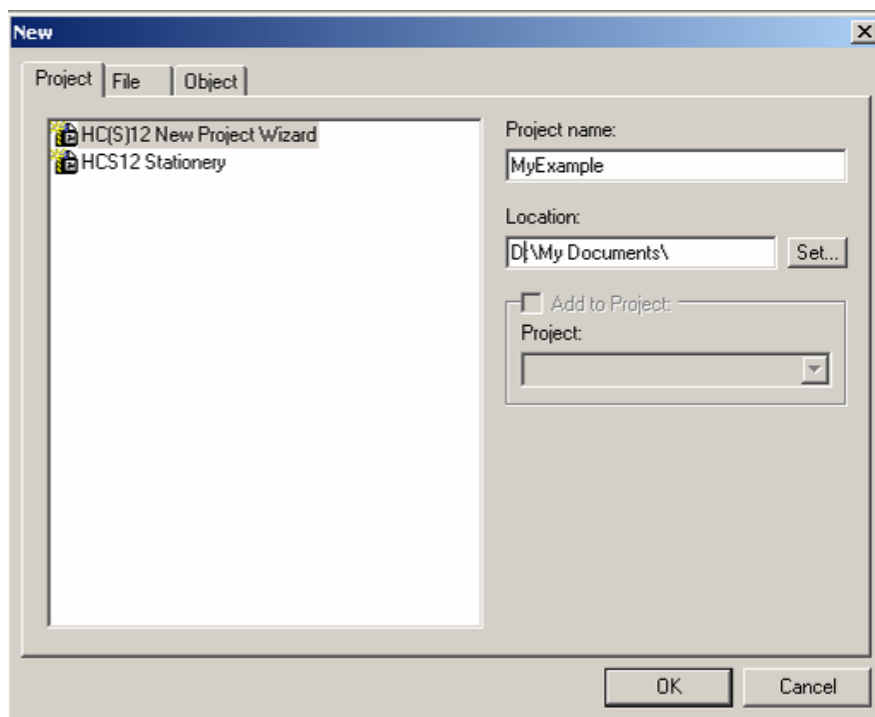


Figure 2. CodeWarrior New Project Window

- 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 3a)

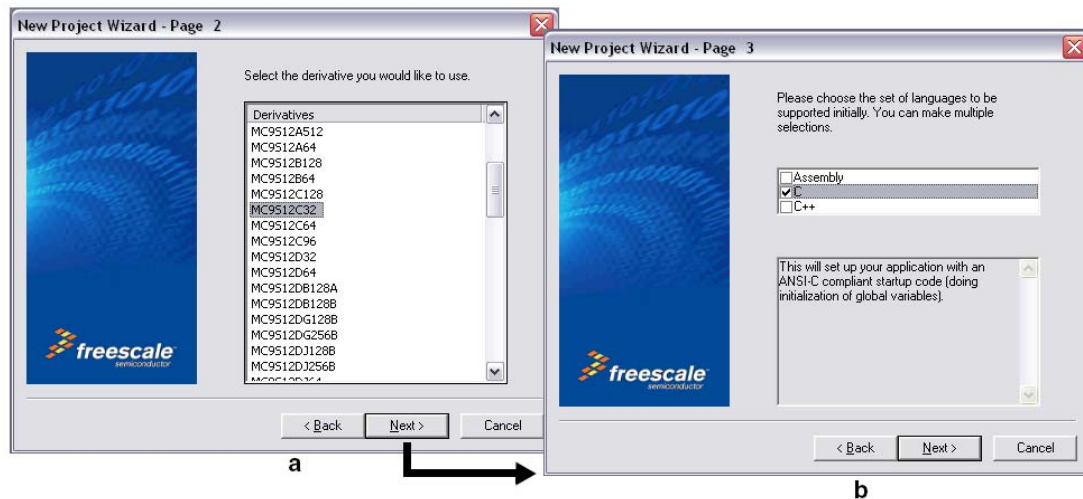


Figure 3. CodeWarrior New Project Wizard - Page 2 and Page 3

- e) Select MC9S12DT256.
- f) Click Next — page 3 of the new project wizard will appear (see Figure 3b).
- 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 10a).

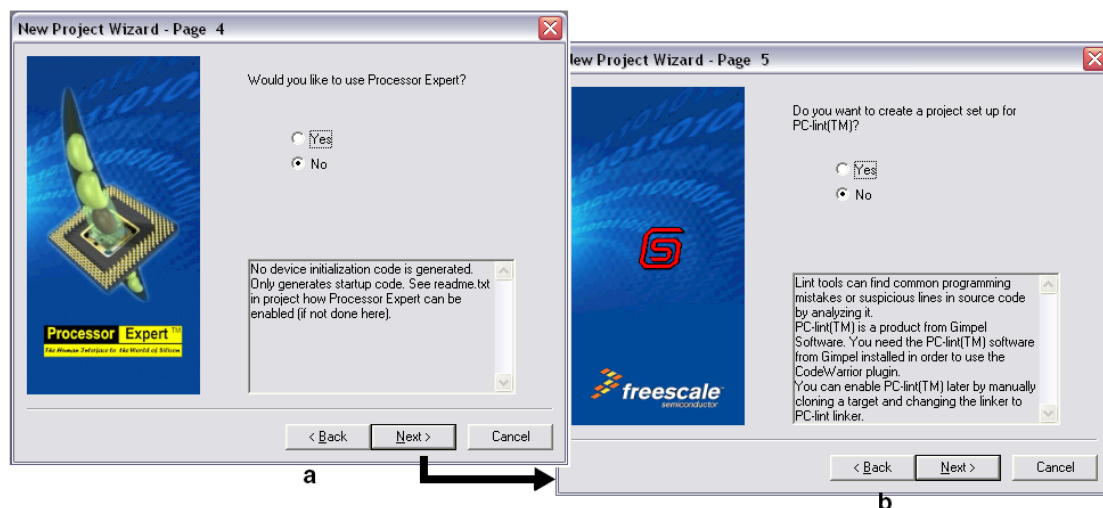


Figure 4. CodeWarrior New Project Wizard Page 4 and Page 5

- 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 4b).
- 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 5a).

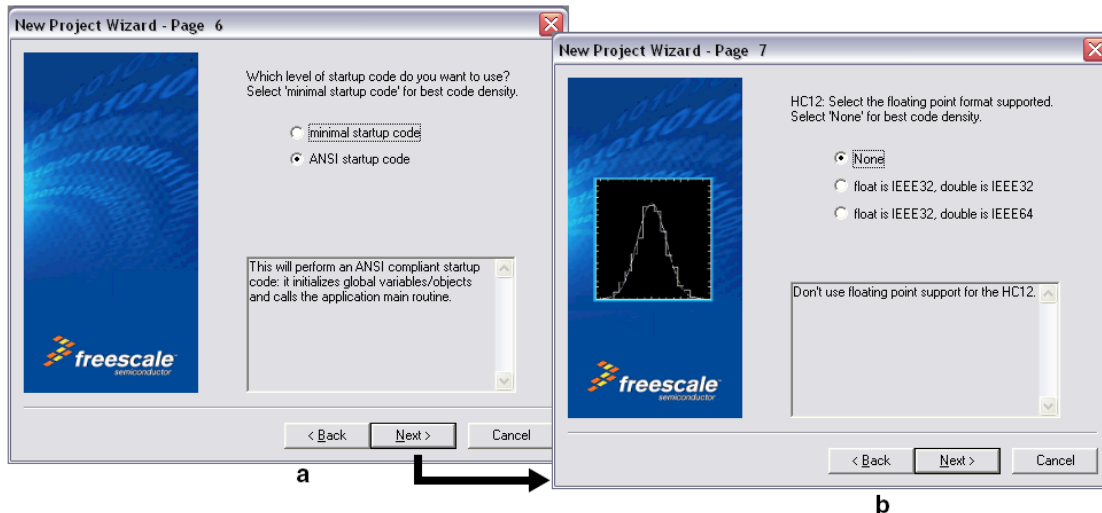


Figure 5. CodeWarrior New Project Wizard Page 6 and Page 7

- m) Select ANSI startup code.
- n) Click Next — page 7 of the new project wizard will appear (see Figure 5b).
- 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 6a).
- q) Select Small memory model
- r) Click Next — page 9 of the new project wizard will appear (see Figure 6b)

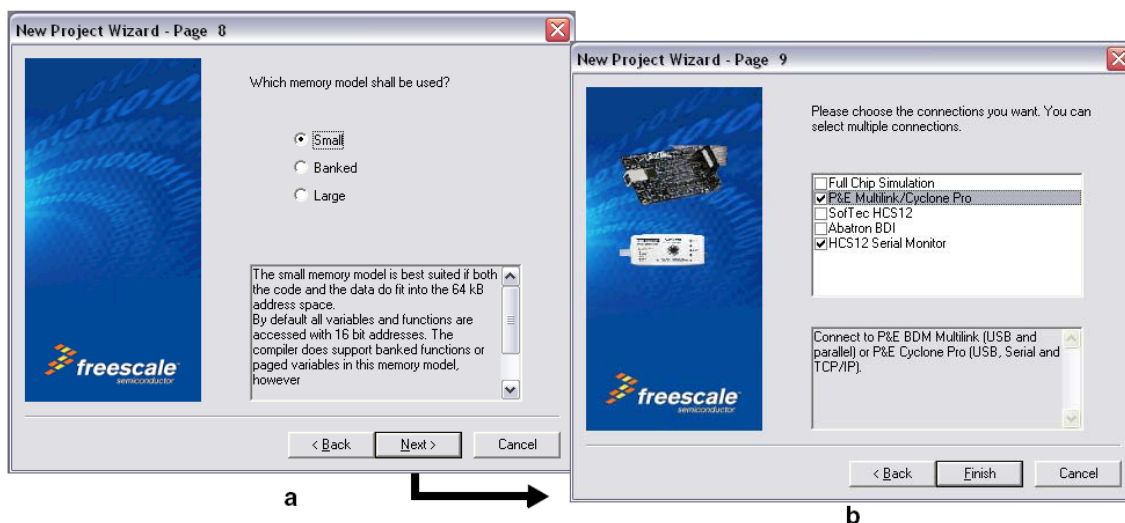


Figure 6. CodeWarrior New Project Wizard Page 8 and Page 9

- s) Page 9 allows you to specify connections that the project should be configured to support. Select HCS12 Serial Monitor, Full-Chip Simulation and P&E Multilink/Cyclone Pro.

NOTE

When choosing connections to specify more is better. This allows you to move to different targets without having to recreate your project. The only catch is that you must be aware which one you have selected, which we will show you how to do in the next few points.

- t) 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 7).

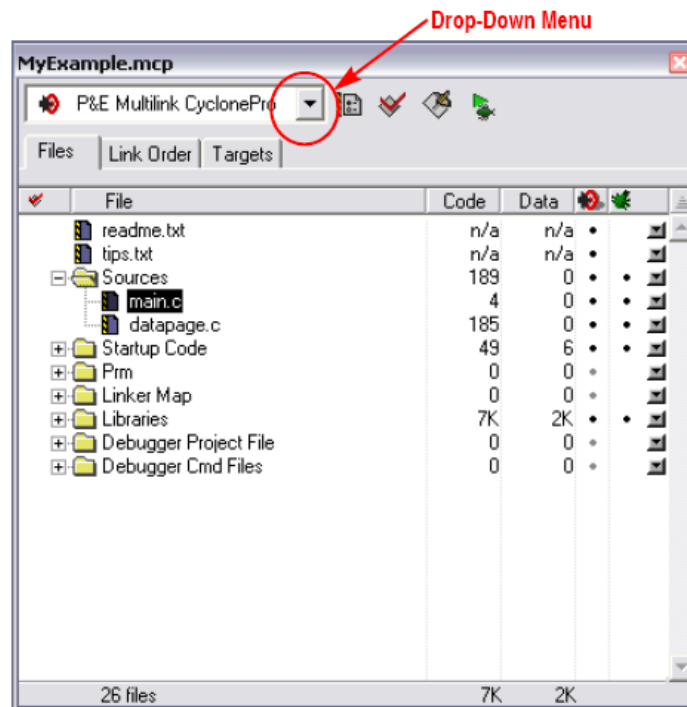


Figure 7. CodeWarrior Project Window

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.

3) Select Build Target.

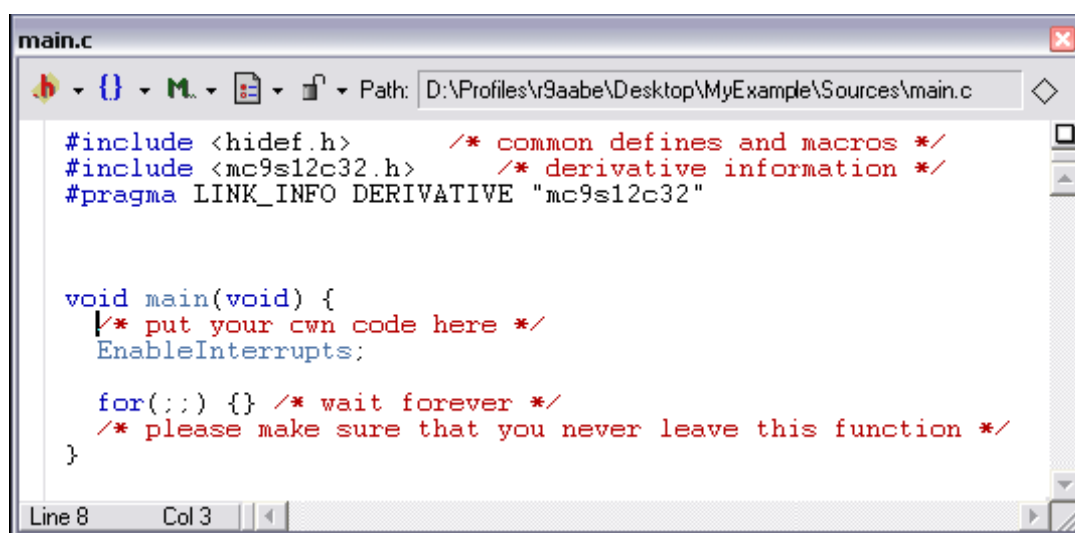
Your project can contain multiple build targets. For this example, we use the HCS12 Serial Monitor target, which uses a program resident in the microcontroller memory to use the serial port interface for device program and debug.

- a) Click the drop-down menu of the project window (see Figure 7).

- b) Select HCS12 Serial Monitor.

NOTE
Alternative, and more widely used, methods of device program and debug uses the Background Debug Mode (BDM). This will require additional circuitry often called the POD. You can purchase BDM programmers online, or most of the latest development boards have included this hardware.
BDM does not occupy memory space or constrain the Serial Port for debug like the Serial Monitor. <i>If you use the BDM you can write-over the factory preloaded Serial Monitor program. You will be required to re-flash this program into memory before you can use this mode again.</i>

- 4) Edit source code.
- Click the + sign next to the sources folder to reveal files (see Figure 7).
 - Double-click the main.c file. An editor window will open and display the contents of main.c similar to that shown (see Figure 8)



```
#include <hcs12.h>          /* common defines and macros */
#include <mc9s12c32.h>       /* derivative information */
#pragma LINK_INFO DERIVATIVE "mc9s12c32"

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

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

Figure 8. Example Editor Window - main.c

- Make changes to the contents of main.c file, if desired.
 - If you make changes to main.c, from IDE main menu bar, select File > Save.
- 5) Add files (if appropriate).
- Highlight the Sources folder.
 - From the IDE main menu bar, select Project; a menu will appear.
 - Select Add Files. A dialog box will appear.
 - Navigate to the directory that contains the file you want to add.
 - Highlight the filename of the file you want to add to your project.
 - 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.
- 6) 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.

5.2 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 HCS12 Serial Monitor on the application module. Section 2, “Software Setup” and Section 3, “Hardware Setup” must be completed before executing the steps in this section.

- 1) Make sure power is applied to the application module. If not, go back and complete hardware setup section instructions.
- 2) Place the Microcontroller in DEBUG mode. This can be done by holding the RESET button and the SW1 button. Continue to hold SW1 while releasing RESET, wait 3 seconds, and then release SW1.
- 3) Start CodeWarrior debugger.
 - a) From the main menu bar in CodeWarrior IDE, select Project.
 - b) Select Debug — the Debugger window will open in HCS12 Serial Monitor 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 communication port being used by the connected serial cable and click OK.

- d) Once the debugger is launched a window similar to Figure 9 will appear. The operation of the various sub windows is listed in Table 2 on the subsequent page. The configuration of sub-windows can be user configured and saved using File->Save Workspace

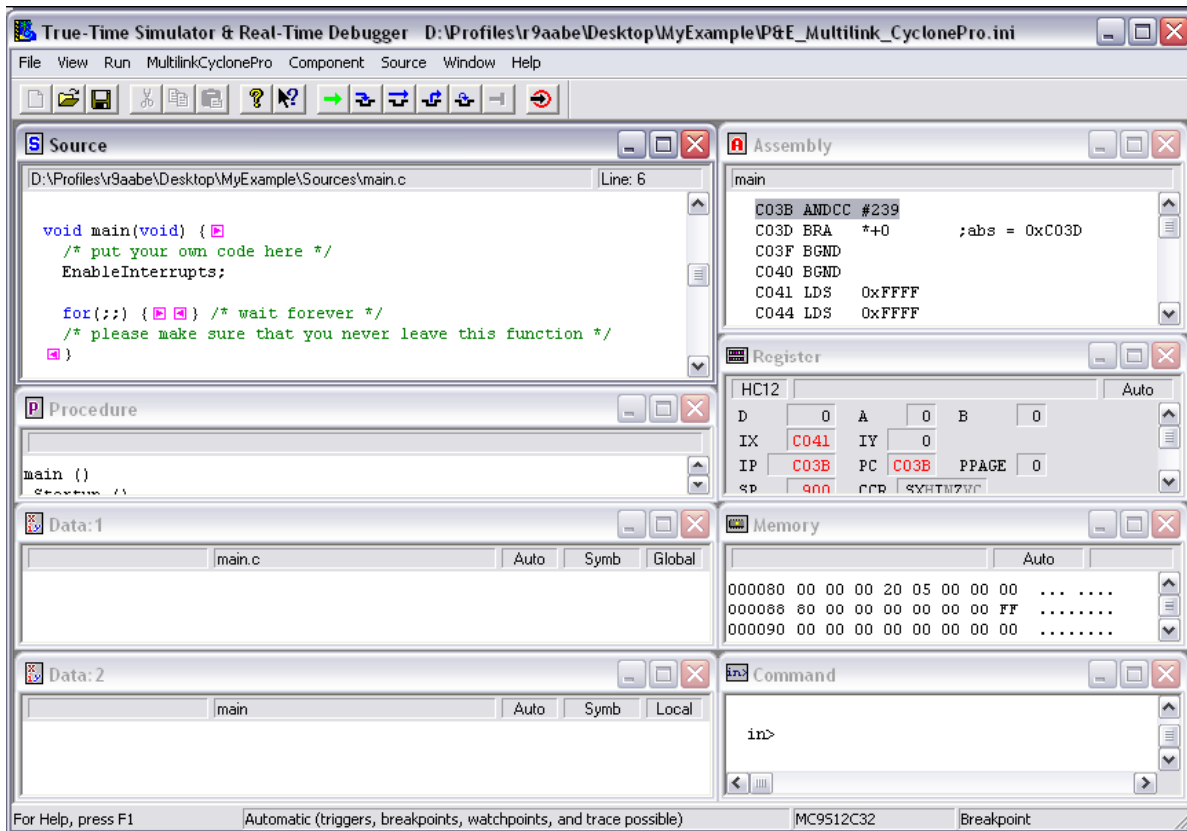



Figure 9. CodeWarrior Debugger Window

Table 2. 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

- 4) Reset the target by clicking on the reset target icon. 
- 5) 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.
- 6) Right-click on executable line of source code in Source window to set breakpoints or triggers in program code.
- 7) 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

- 8) 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.

- 9) In the debugger window tool bar, select: File > Exit (to exit debugger).
10) 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.


6 Running the Demonstration Application



The following example demonstrates the S12DT256SLK Application Module hardware and provides example code using the CodeWarrior software tools.

- 1) Configure the project board to the default jumper positions. Refer to Section 3.2 steps 1-15 for more information.
- 2) This demonstration comes preloaded into the Application module, and once powered the device begins running the demonstration. You can download the source code at www.freescale.com and search for keyword APS12DT256_DEMO.
 - a) If downloaded from web, unzip and extract the software to a directory on your PC.

NOTE

There is also a software example for use with the Freescale Project Board (PBMCUSLK). To find go online to www.freescale.com and search for PBS12DT256_DEMO.

- b) To view and run the code from within CodeWarrior, find the location that you extracted the files to, open the APS12DT256SLK_Demo folder. Double-Click on APS12DT256SLK_Demo.mcp
- c) Click the Run Debug  button to launch the real-time debugger.

- 
- d) Click the Run button  to run the application in the debugger.
 - 3) This is a very simplistic demonstration, to illustrate that the device is operational.
 - 4) Press PB1 on the application module. The microcontroller will acknowledge the button press by illuminating LED 1.
 - 5) Press PB2 on the application module. The microcontroller will acknowledge the button press by illuminating LED 2.

For more features check out the Freescale Project Board, at www.freescale.com/universityprograms

7 Revision History

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

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
Email: sales@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.