

Application Note

AN2322/D

Rev. 2, 10/2002

Reprogramming the
M68DEMO908QT4

By: John Suchyta
Applications Engineering
8/16-Bit Products Division

Introduction

The M68DEMO908QT4 demo board was developed as a promotional tool for the introduction of the 8-pin MC68HC908QT4 microcontroller (MCU). The demo board consists of the MCU, an RS232 level shifter for serial communications, a pushbutton switch and potentiometer for inputs, and an LED for output. The board is powered by a 9 V battery and has a simple “blinker” program and a “user monitor” program.

The 4K MCU FLASH memory can be reprogrammed through the user mode monitor program. The user monitor program itself however cannot be programmed or erased on the board as it was constructed. This application note describes how to reprogram the user program space with a serial cable through the user monitor, and how to reprogram the user monitor program through a MON08 interface.

This application note requires that a registered version of CodeWarrior® Development Studio for 68HC08, Special Edition, version 2.1 or later, be loaded on your PC.

Demo Board Memory Map

The M68DEMO908QT4 demo board comes with three sections of code in the FLASH memory. The first two sections, in the 4K user array (**Figure 1**), contain a “blinker” demo program and some test code. The third section, in the vector array, contains a short “user monitor” program that is described in the application note *User Mode Monitor Access for MC68HC908QY/QT Series MCUs*, Motorola document order number AN2305/D. The user monitor is used in the first part of this document to perform a simple debug step and to reprogram the user FLASH with a different program. The user monitor is invoked by holding the pushbutton switch down during power up.

CodeWarrior is a registered trademark of Metrowerks, a Motorola company.

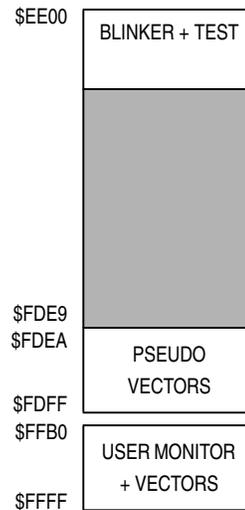


Figure 1. M68DEMO908QT4 Demo Board FLASH Memory Configurations

Reprogramming the User FLASH with the User Monitor

First, Get the Trim Value

Before programming the FLASH with any new code, we strongly advise you to copy the Internal Oscillator trim value from memory location \$FFC0 and write it on this document or on the demo board. This will save time later in the case that the user monitor program gets corrupted.

To do this, slide switch S1 to the closest edge of the board (OFF position). Connect a 9 V battery to the terminals. Connect a serial cable between the board and your PC. Do not switch the power on yet.

1. Launch CodeWarrior IDE (CW08 V2.1 or later).
2. Under File menu select New.
3. Select HC08 Stationery ([Figure 2](#)).
4. Type GetTrim as Project name.
5. Click OK.

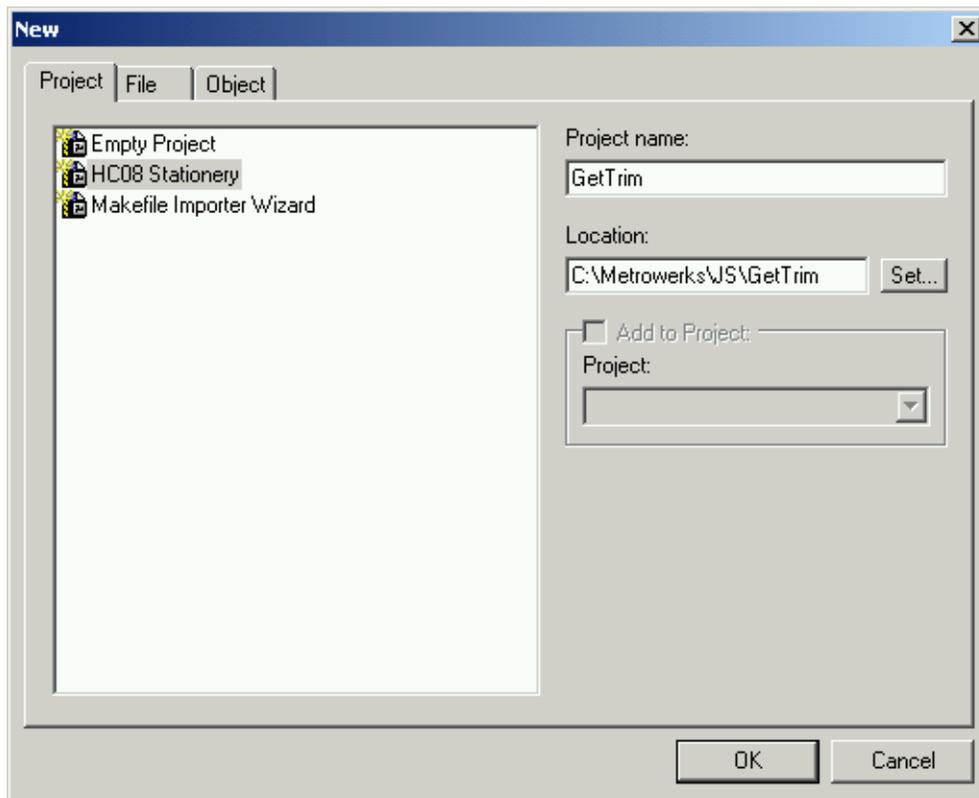


Figure 2. CodeWarrior New Project Window

6. In the New Project Window, click on + by QT_QY to expand the selections. (Figure 3)
7. Click on + by QT4 to expand the selections.
8. Click on Absolute Assembly.
9. Click OK.

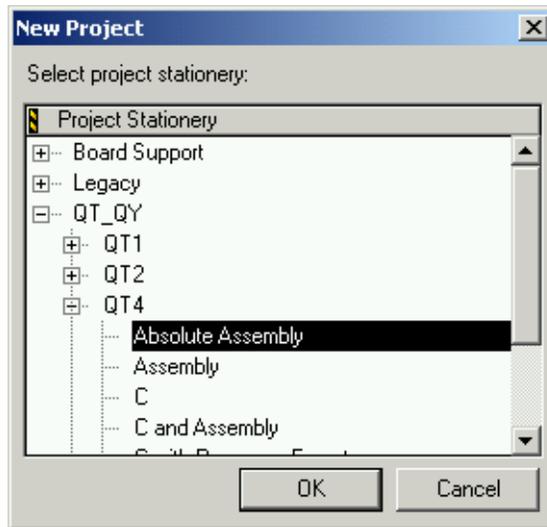


Figure 3. New Project Stationery Window

The GetTrim project window appears. (Figure 4)

10. Click the Debug icon (green arrow).

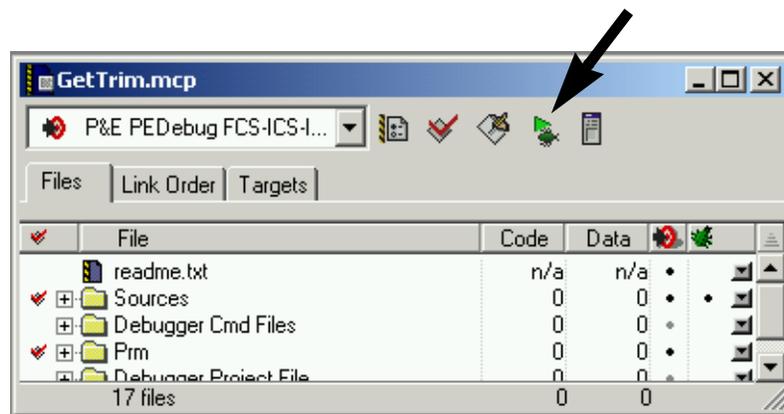


Figure 4. GetTrim Project Window

The True-Time Simulator & Real-Time Debugger window automatically loads. (**Figure 5**)

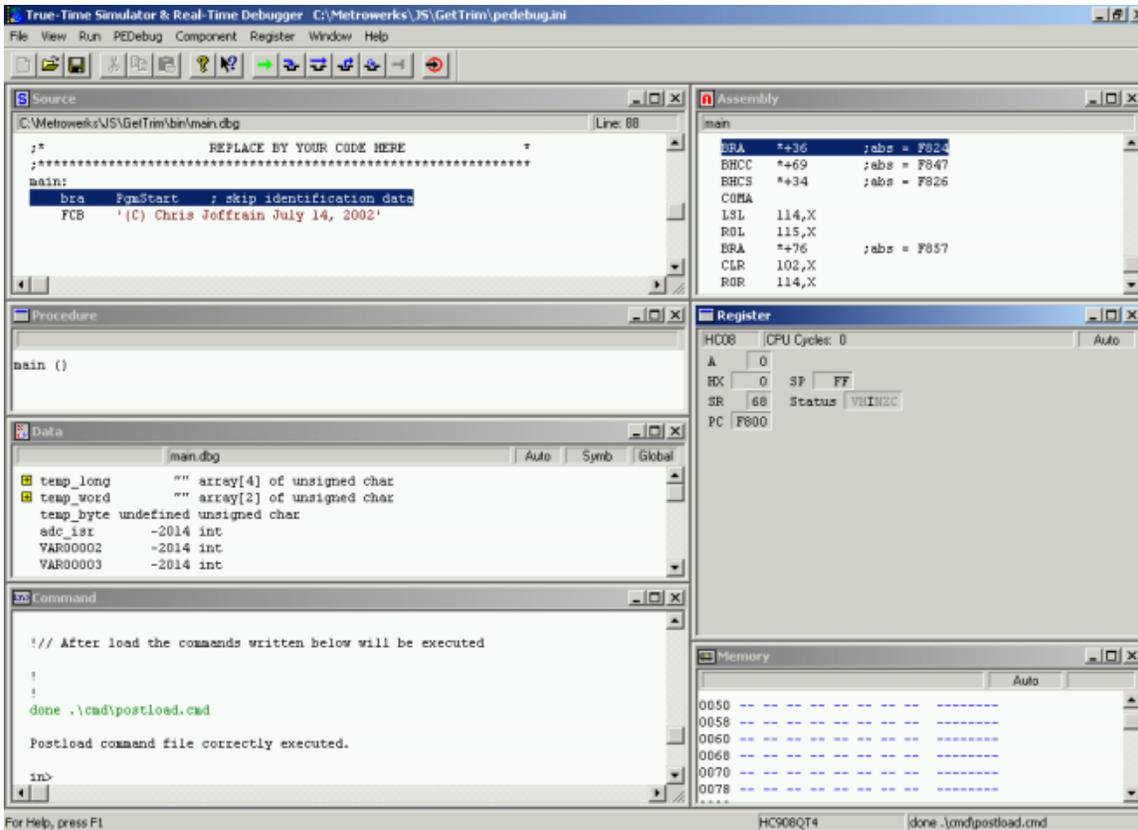


Figure 5. True-Time Simulator and Real-Time Debugger Window

11. From the PEDebug pulldown menu select Device: M68DEMOQTY. (Figure 6)
12. Likewise, select Mode: In-Circuit Debug/Programming.

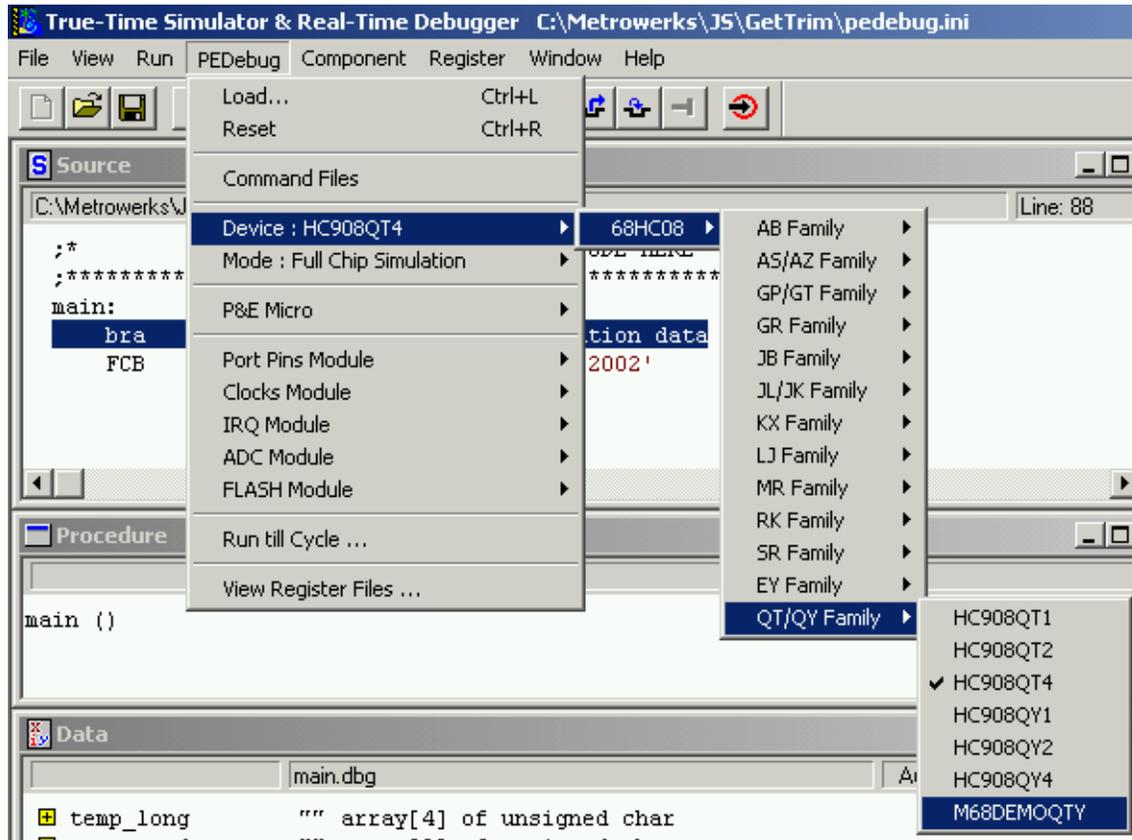


Figure 6. PEDebug Pulldown Menu for Device and Mode Selection

The PROG08SZ programmer launches and the “Attempting to contact target” window appears. (Figure 7)

13. Select Class III in Target Hardware Type.
14. Select Serial Port 1 or other port.
15. Select Baud = 9600 Baud.
16. Check the IGNORE security failure... box.
17. Click Contact target with these settings...

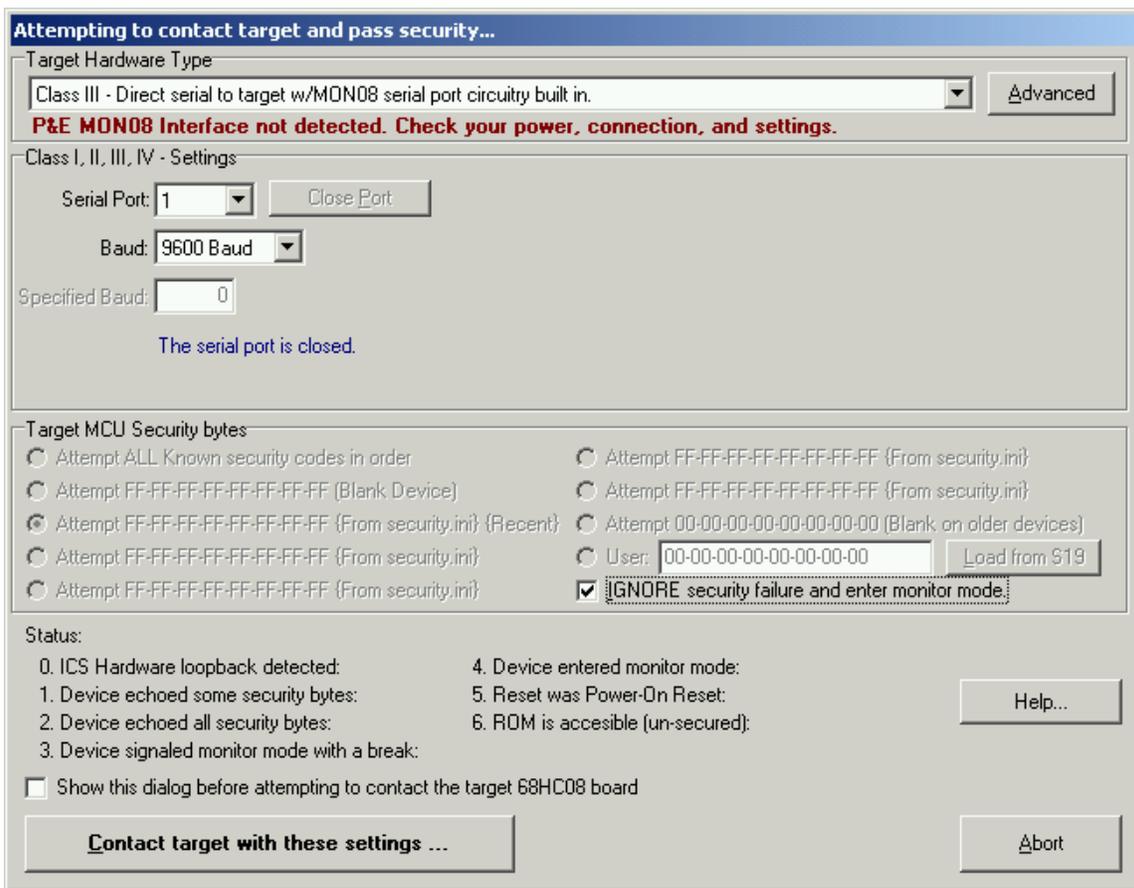


Figure 7. Attempting to Contact Target Window

The M68DEMOQTY Board Reset Sequence window appears. (Figure 8)

18. Press and hold the S1 pushbutton switch on the demo board while sliding S2 to the ON position. (Note: LED will blink if you did not hold the pushbutton down while cycling power.)
19. Click OK.

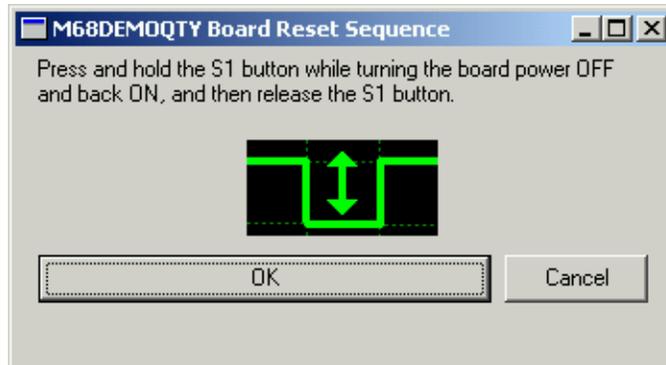


Figure 8. M68DEMOQTY Board Reset Sequence Window

The programmer Confirm window appears. We do not want to reload the object file at this time.

20. Click **No** in Confirm window. (Figure 9) **Important: do not click Yes.** We do not want to program the FLASH at this time since we haven't selected valid code to replace the user program.

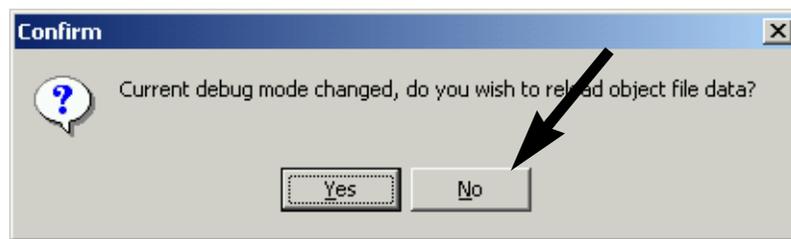


Figure 9. Programmer Confirm Window

The above actions let the debugger enter monitor mode through the user monitor program that resides in the 80 byte block of FLASH (\$FFB0–FFFF). At this time we can examine the contents of the FLASH.

21. Right-click in the Memory window in the lower right corner of the debugger window.

22. Select Address.
23. Type in FFC0 and click OK. (Figure 10)

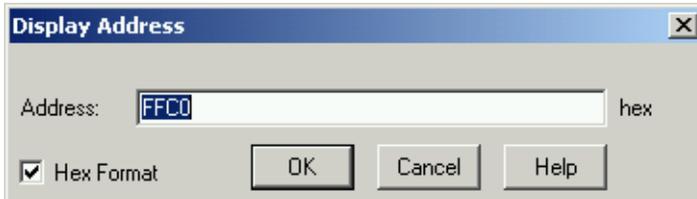


Figure 10. Debugger Display Address Window

24. Record the value at location FFC0 as the Internal Oscillator trim value. (Figure 11) Write it here in this document or, better yet, write it on the bottom of the board with a permanent marker:

MY TRIM VALUE =

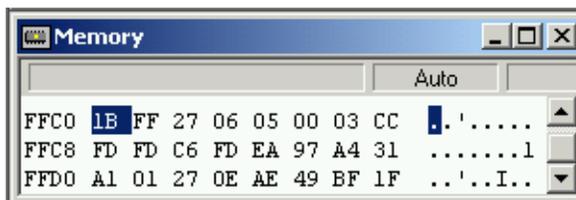
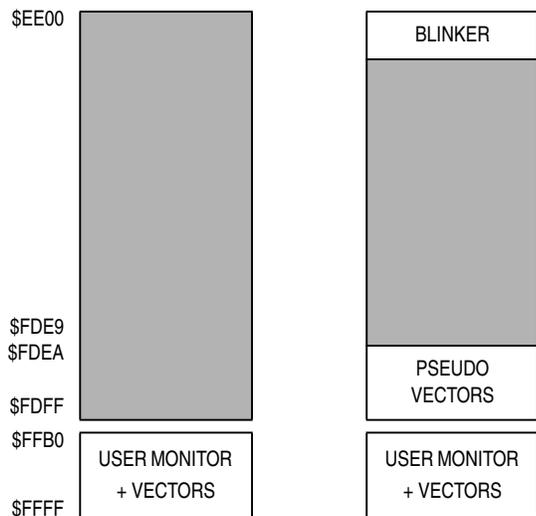


Figure 11. Debugger Memory Window

25. Close the Debugger and CodeWarrior windows.
26. Power down demo board.

Reprogram the User FLASH

The M68DEMO908QT4 demo board comes from the vendor with extra (test) code in the FLASH. The following is a sample session that uses the CodeWarrior IDE to erase the user contents of the 4K FLASH module (see Figure 12a) and reprogram with a sample project (Figure 12b). The only hardware interface needed to do this is a serial cable connected between the board and your PC. Turn off power to the board at this time.



a. User Monitor Erase b. Reprogram
Figure 12

1. Launch the CodeWarrior IDE. (CW08 V2.1 or later)
2. Under File Menu select Open.
3. Select (CodeWarrior_Examples) in CodeWarrior CW08_V2.1 folder and click Open. (Figure 13)

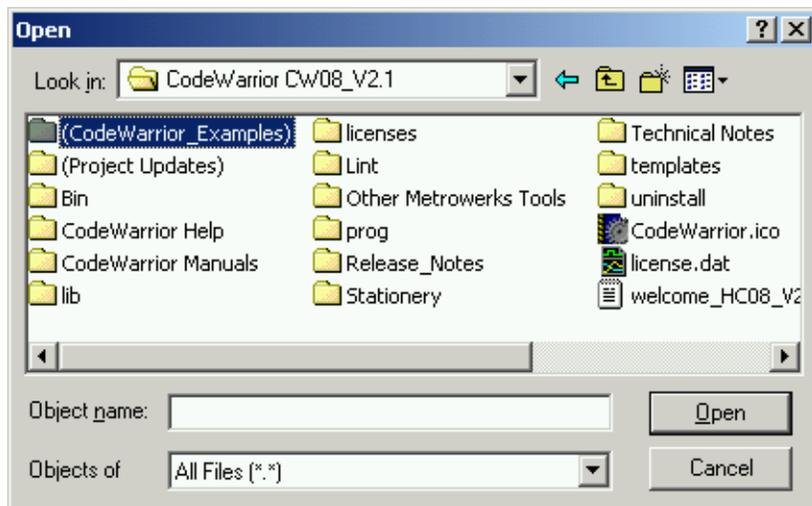


Figure 13. Find the Demo Project

4. Continue selecting and opening the following folders in sequence:
 - a. HC08
 - b. HC08 PEDebug
 - c. QT_QY Demos
 - d. 68HC908QT4 Demonstration Board
 - e. 68HC908QT4 Demo- User Monitor Mode

- Double-click on the demo_user_monitor.mcp file name. The CodeWarrior project will open with the following window. (Figure 14)

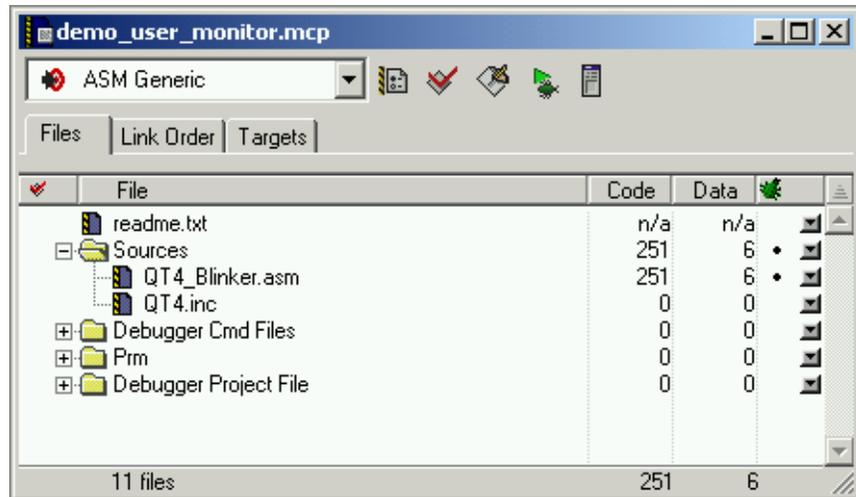


Figure 14. Project Window for Demo Software

- Click the Debug icon (green arrow) in either the CodeWarrior menu bar or in the project window to start the debugger.
The True-Time Simulator & Real-Time Debugger Window opens up. (Figure 5)
- From the PEDebug pulldown menu select Device: M68DEMOQTY. (Figure 6)
- Likewise, select Mode: In-Circuit Debug/Programming.
The PROG08SZ programmer launches and the M68DEMOQTY Board Reset Sequence window appears (Figure 15). If the Attempting to contact target window appears (Figure 7), select Class III, serial port, 9600 baud, IGNORE security failure, and click Contact target.
- Press and hold the S1 pushbutton switch on the demo board while sliding S2 to the ON position. (Note: the LED will blink if you did not hold the pushbutton down while turning power on.)
- Click OK.

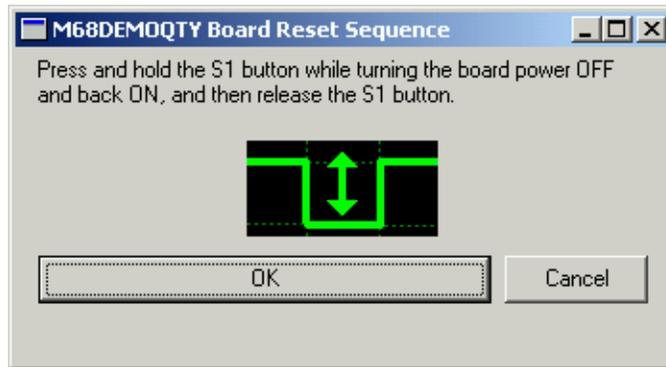


Figure 15. M68DEMOQTY Board Reset Sequence Window

The programmer Confirm window appears. (Figure 16)

11. Click Yes.

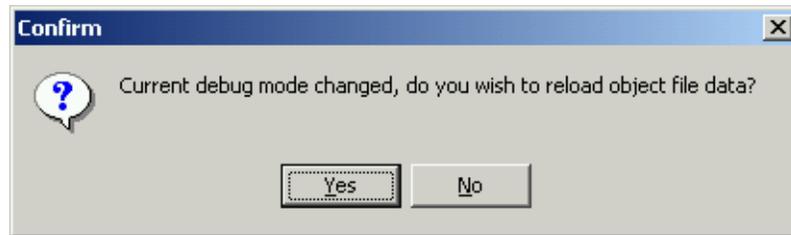


Figure 16. Programmer Confirm Window

The Erase and Program FLASH window appears. (Figure 17)

12. Click Yes.

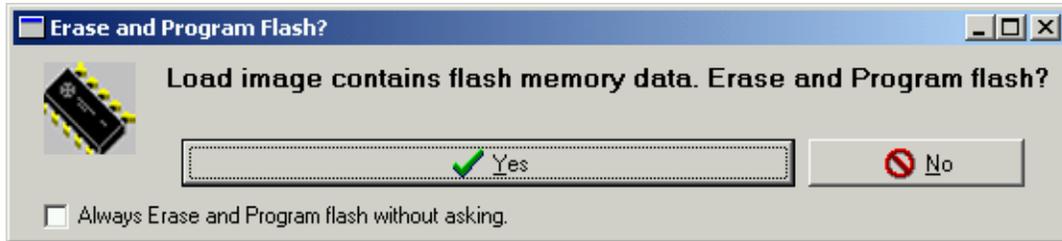


Figure 17. Programmer Erase and Program FLASH Window

The programmer begins an automated sequence that erases the FLASH then reprograms it with the project object file.

The M68DEMOQTY Board Reset Sequence window appears twice more.

13. Turn off switch S2.
14. Hold S1 depressed while turning S1 back on.
15. Click OK.
16. Repeat steps 13 – 15 as prompted.
The Debugger window reappears.
17. Right-click in the Memory window.
18. Select Address in menu.
19. Type in address EE00 and click OK. The new memory contents are shown (Figure 18). (Contents will be FF if not programmed.)

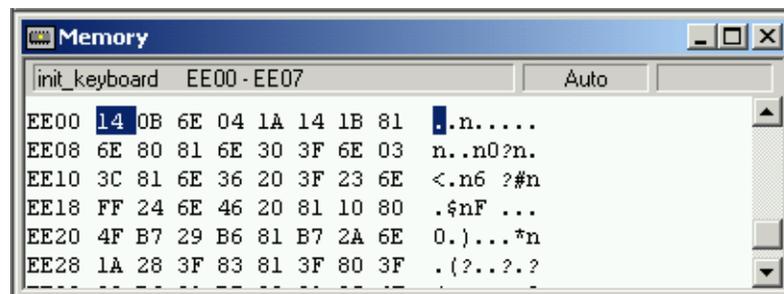


Figure 18. Verify New Code in FLASH

20. Close the Debugger and CodeWarrior windows.
21. Power down the demo board.
22. Disconnect the serial cable from the demo board.
23. Power up the demo board. The new demo program operates similarly to the original program, but the blink and brightness rates are slightly slower.

Reprogram the User Monitor

In the event that you accidentally erase the entire FLASH contents of the demo board, for example, by using the wrong Device type in the debug session or by clicking Yes when No was the correct response, you can reprogram the user monitor. Normal monitor mode will be needed to do this. Other than the CodeWarrior tool, the following are required.

- Normal monitor mode interface
 - External
 - On-board
- Internal oscillator trim value
- Software from AN2305

This procedure can also be followed, for the most part, to load a completely new program into the FLASH memory on the demo board.

Normal Monitor Mode Interface

The normal monitor mode interface must be used to reprogram the entire FLASH memory. The interface brings serial communications, high voltage, an external clock, and mode settings to the board.

External Requirements

Refer to the application note entitled *Low-Cost Programming and Debugging Options for M68HC08 MCUs*, Motorola document order number AN2317/D, for one of several monitor mode circuits that can serve as the interface between the PC and the board. Any of the three circuits, or a MON08-Cyclone or MON08-Multilink from P&E Microcomputer Systems, can be used.

On-Board Requirements

The simplest MON08 interface on the board can be built by soldering a dual row header in the J6 location on the demo board. This 16-pin header¹ (2 rows by 8 pins) can also be made by mounting 2 8-pin single row pin strip headers² side by side in J6.

1. DigiKey S2012-36-ND; Mouser 517-836-01-36
 2. DigiKey S1012-36-ND; Mouser 517-834-01-36

Connect the following signals from the monitor mode interface to the J6 pins on the demo board.

1. Connect the ground line to pin 2 (V_{SS}).
2. Connect the \overline{RST} signal to pin 4 (PTA3/ \overline{RST}).
3. Connect the V_{TST} signal to pin 6 (PTA2/ \overline{IRQ}).
4. Connect the COM signal to pin 8 (PTA0).
5. Connect the MOD1 signal to pin 10 (PTA4).
6. Connect the MOD0 signal to pin 12 (PTA1).
7. Connect the OSC signal to pin 13 (OSC1).
8. Connect the V_{DD} line to pin 15 (V_{DD}).

Note that the V_{DD} connection is only necessary if the MON08 interface is target powered. The MOD0 signal could be left off since it is pulled up by an LED on the demo board.

Internal Oscillator Trim Value

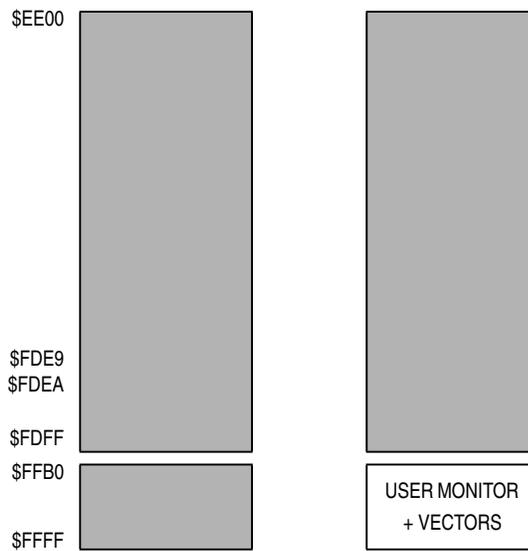
The user monitor program will need the internal oscillator trim value that we recommended copying in the first debug session. If this one-byte value is not written on the bottom of the board or available, please refer to application note entitled *MC68HC908QY4 Internal Oscillator Usage Notes*, Motorola document order number AN2312/D, for some methods to find this value.

Software from AN2305

The user monitor program is contained in the AN2305SW.zip file from application note AN2305. This source code is on the CD that came with the demo board, or can be downloaded from the documentation library on the MCU Web site at www.motorola.com/semiconductors/mcu. Un-zip the source file (UserMonQT4.asm) and remember the location for the following steps.

Reprogramming the FLASH

The following procedure will erase and reprogram the FLASH modules as shown in **Figure 19a** and **Figure 19b**.



a. Erase FLASH b. Reprogram Entire FLASH

Figure 19

1. Launch CodeWarrior IDE (CW08 V2.1 or later).
2. Under File menu select New.
3. Select HC08 Stationery.
4. Type in UserMonitor as project name.
5. Click OK.
6. In the New Project Stationery Window, click on + by QT_QY to expand selections. (**Figure 3**)
7. Click on + by QT4 to expand the selections.
8. Click on Absolute Assembly.
9. Click OK.

The UserMonitor project window is displayed. (**Figure 20**)

10. Click on the + to expand the Sources folder.
11. Click on the Sources folder to highlight it.
12. Right-click on the Sources folder and select Add Files...

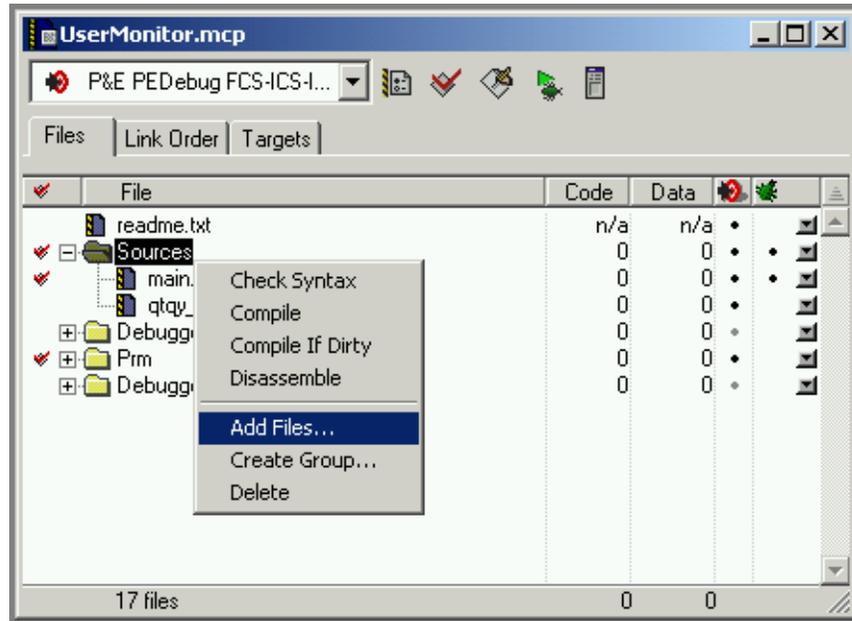


Figure 20. User Monitor Project Window — Add Files

The “Select files to add...” window appears.

13. Navigate to the UserMonQT4.asm file and select it.
14. Click Open.

The Add Files window appears. (Figure 21)

15. Uncheck MMDS-MMEVS and Cyclone.
16. Click OK.
17. Close the Project Messages window if it appears.

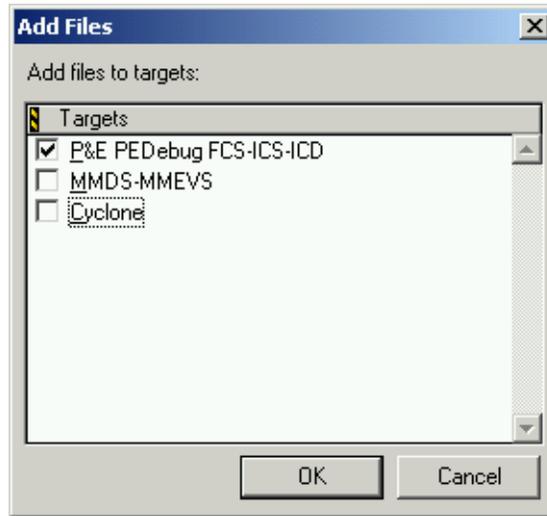


Figure 21. Add Files to Targets Window

18. In the UserMonitor project window, right-click on the main.asm file. (Figure 22)
19. Select Delete.

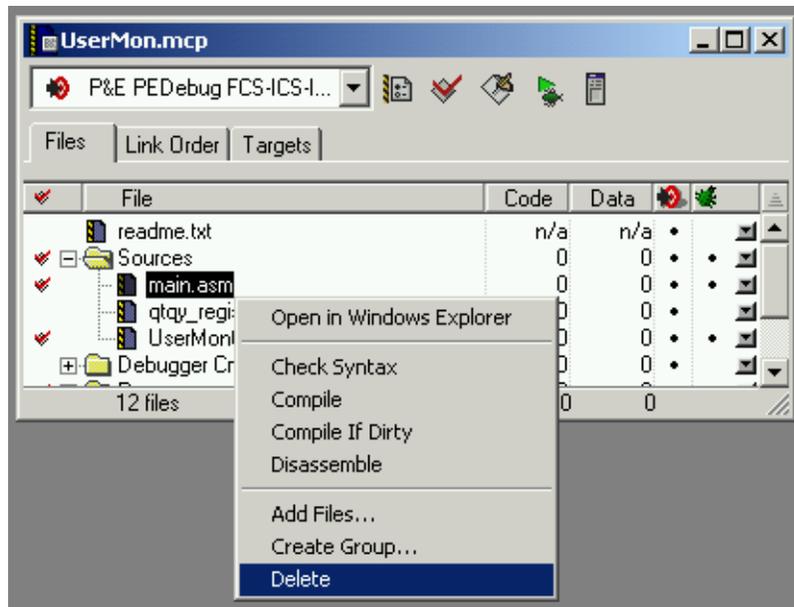


Figure 22. Delete Files from Project Window

20. Click OK in the warning window.

The qtqy_registers.inc file is not needed in this project and can be left in the project or deleted.

21. Double-click on the UserMonQT4.asm file.
22. After the file opens, scroll down to the bottom of the file.
23. Add 2 lines starting in column 15 with the following code. (Figure 23) (Replace trim value with the value you recorded earlier in this application note.)

```
org $FFC0
fcb $(trim value)
```

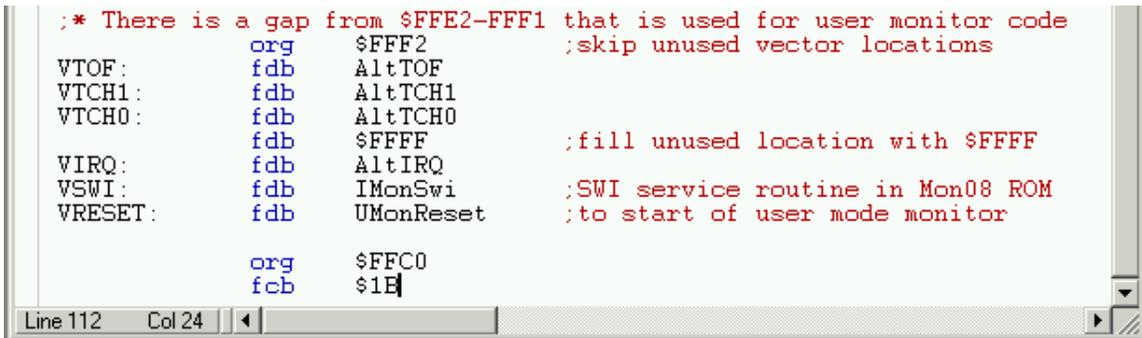


Figure 23. New Code Added at End of UserMonQT4.asm

24. Click the Debug icon in the CodeWarrior toolbar.
The True-Time Simulator & Real-Time Debugger launches.
A error window will also appear (Figure 24). This error can be ignored because the postload.cmd file could not find an entry point named “main” in our code.
25. Click OK in error window.

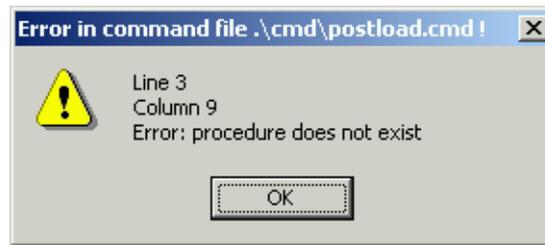


Figure 24. Error in Command File Window

26. From the PEDebug pulldown menu select Device: HC908QT4.
27. Likewise, select Mode: In-Circuit Debug/Programming.
The PROG08SZ programmer launches and the Attempting to contact target window appears. (Figure 25)
28. Select Class III in Target Hardware Type. Also select serial port and baud rate.
29. Check the IGNORE security failure... box.
30. Click Contact target with these settings...

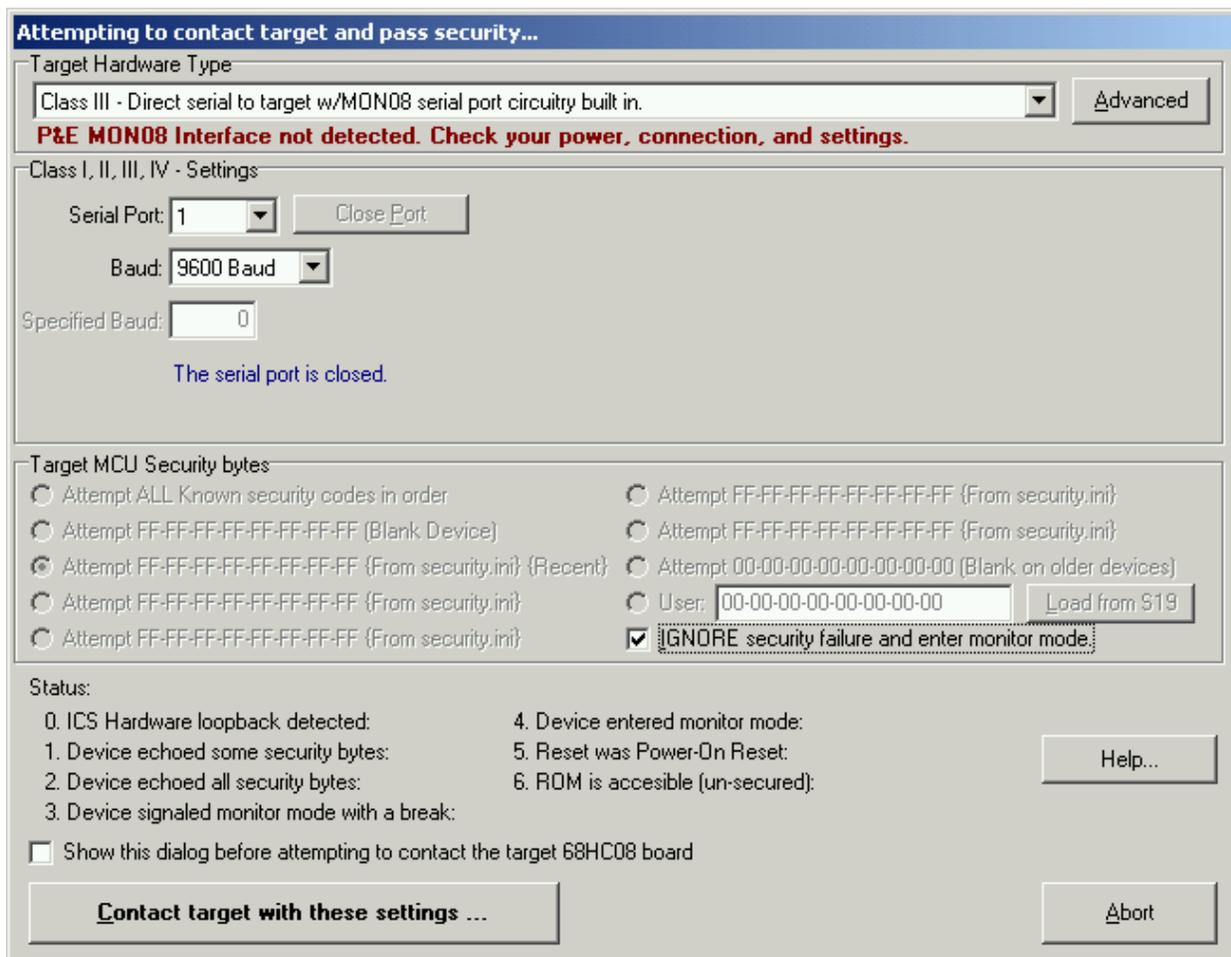


Figure 25. Attempting to Contact Target Window

The Power Cycle Dialog window appears. (Figure 26)

31. Slide the S2 power switch to the ON position. The green LED will light. Since we are in normal monitor mode instead of user monitor mode in this example, we do not press S1 while turning on power.
32. Click OK.

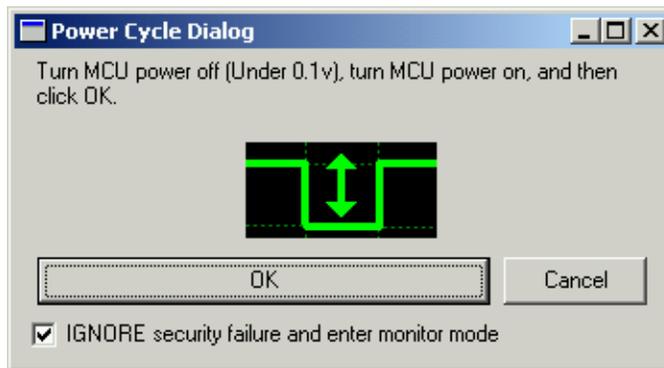


Figure 26. Power Cycle Dialog Window

The programmer Confirm Window appears. (Figure 27)

33. Click Yes.

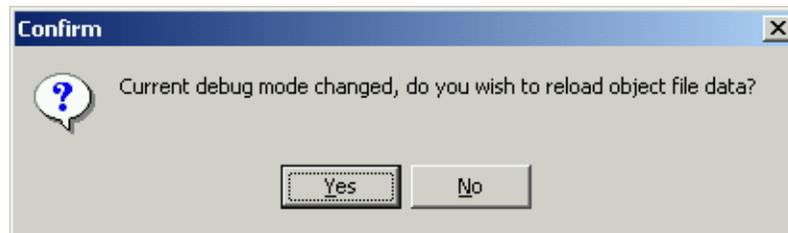


Figure 27. Programmer Confirm Window

34. Click Yes in the Erase and Program FLASH window. (Figure 28)

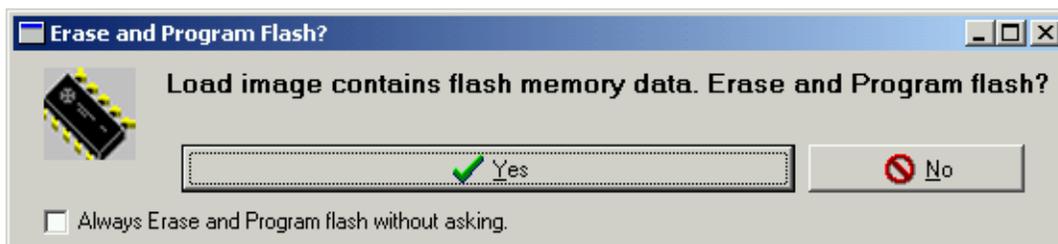


Figure 28. Programmer Erase and Program FLASH Window

The Power Cycle Dialog window appears a few more times as the programming script erases and reprograms the FLASH memory.

35. Follow the instructions in the window and click OK as requested.

The Command window in the True-Time Debugger indicates that the FLASH programming was successful. The FLASH contents can be checked by right-clicking in the Memory window and entering FF80 as the address. (Figure 29)

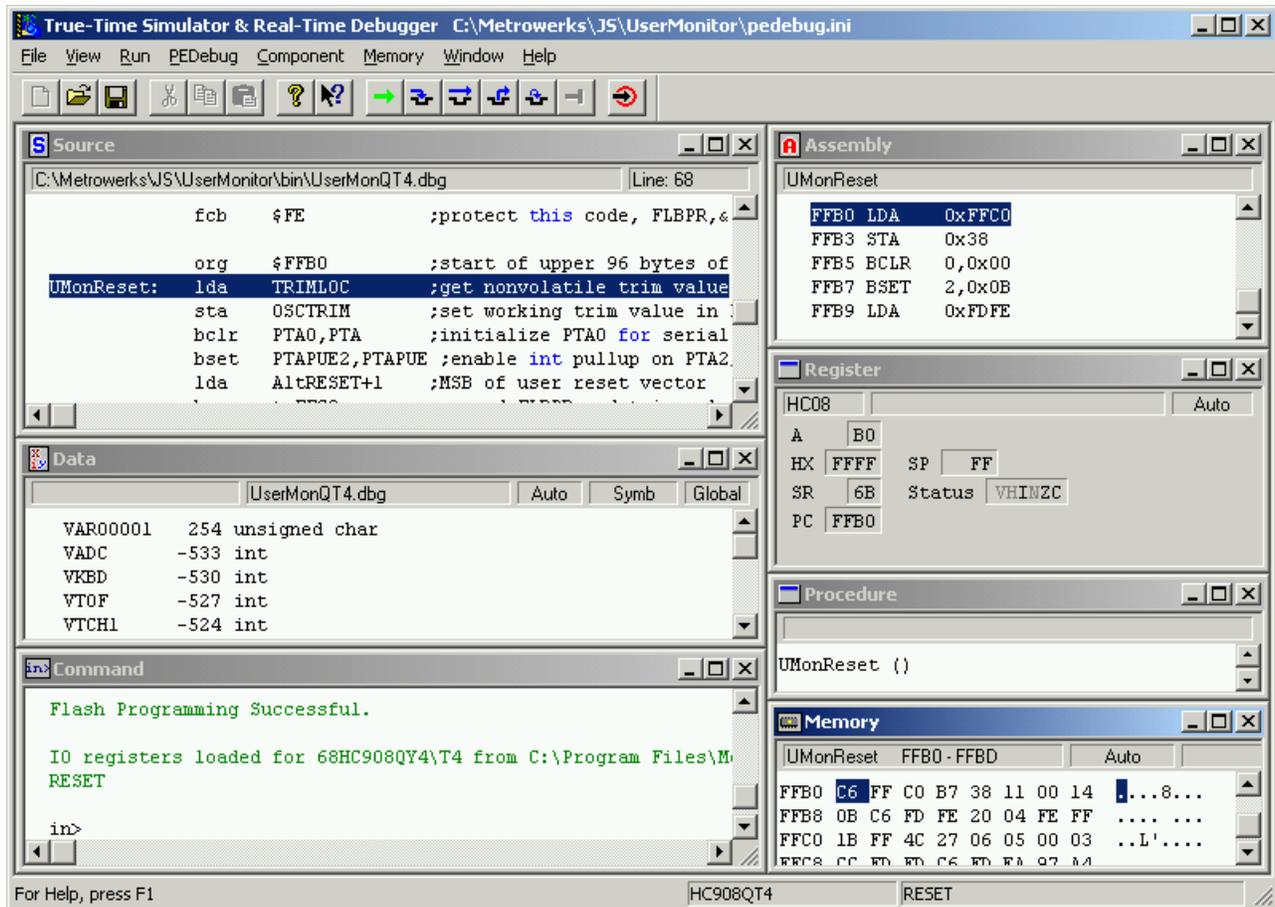


Figure 29. True-Time Debugger Window — Successful Programming

36. Close the Debugger and CodeWarrior windows.
37. Power down the demo board.
38. Disconnect the MON08 interface from the demo board.

The user monitor program is now loaded as shown in Figure 19b. The example program can now be loaded as shown in the Reprogram the User FLASH section.

Conclusion

In this application note, we copied the internal oscillator trim value and reprogrammed the demo program in the FLASH using the user monitor mode. Using these procedures, we can reprogram the user FLASH area of the demo board for other evaluation purposes.

This document also describes how to reprogram the user monitor program into the FLASH in the event that the user monitor FLASH is inadvertently erased. Refer to the following documents for more information on subjects in this application note.

- AN2305/D — *User Mode Monitor Access for MC68HC908QY/QT Series MCUs*
- AN2312/D — *MC68HC908QY4 Internal Oscillator Usage Notes*
- AN2317/D — *Low Cost Programming and Debugging Options for M68HC08 MCUs*

This page intentionally left blank.

This page intentionally left blank.

This page intentionally left blank.

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

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 which 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.

