



TIM-40 PEROM and Sysload Utility
Flash Programmable and Erasable ROM
Programming Guide

Document Number 500-00338

Revision 3.00

September 1999

Copyright © 1999 Spectrum Signal Processing Inc.

All rights reserved, including those to reproduce this document or parts thereof in any form without permission in writing from Spectrum Signal Processing Inc.

All trademarks are registered trademarks of their respective owners.

Spectrum Signal Processing reserves the right to change any of the information contained herein without notice.

Customer Feedback

At Spectrum, we recognize that product documentation that is both accurate and easy to use is important in aiding you in your new product development. We appreciate hearing your comments on how our product's documentation could be improved.

If you wish to comment on any Spectrum documentation then please fax or e-mail a completed copy of this page to us.

Full Name of Document: _____

Document Number: _____ Version Number: _____

If you have found a technical inaccuracy please describe it here:

If you particularly liked or disliked an aspect of the manual then please describe it here:

It may be helpful for us to call you to discuss your comments. If this would be acceptable please provide the following details:

Name: _____ Telephone #: _____

Organization: _____

Thank you for your time,

Spectrum Signal Processing Documentation Group

Fax: (604) 421-1764

Email: documentation@spectrumsignal.com

Contacting Spectrum...

Spectrum's team of dedicated Applications Engineers are available to provide technical support to you for this product. Our office hours are Monday to Friday, 8:00 AM to 5:00 PM, Pacific Standard Time.

Telephone 1-800-663-8986 or (604) 421-5422

Fax (604) 421-1764

Email support@spectrumsignal.com

Internet <http://www.spectrumsignal.com>

When you contact us, please have the following information on hand:

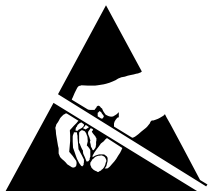
- A concise description of the problem
- The name of all Spectrum hardware components
- The name and version number of all Spectrum software components
- The minimum amount of code that demonstrates the problem
- The version number of all software packages, including compilers and operating systems

Preface

Spectrum Signal Processing offers a complete line of DSP hardware, software and I/O products for the DSP Systems market based on the latest DSP microprocessors, bus interface standards, I/O standards and software development environments. By delivering quality products, and DSP expertise tailored to specific application requirements, Spectrum can consistently exceed the expectations of our customers. We pride ourselves in providing unrivaled pre and post sales support from our team of application engineers. Spectrum has excellent relationships with third party vendors which allows us to provide our customers with a more diverse and top quality product offering.



Spectrum achieved ISO 9001 quality certification in 1994.



As Spectrum's hardware products are static sensitive, please take precautions when handling and make sure they are protected against static discharge.

Table of Contents

1 Introduction	1
1.1. Purpose Of This Manual.....	1
1.2. Overview.....	1
1.3. Referenced Documents.....	1
1.4. Definitions And Acronyms.....	2
<hr/>	
2 Installation	3
2.1. MS-DOS PEROM Utility Installation.....	3
2.2. Solaris 2.x PEROM Utility Installation.....	3
2.3. Setting Up The C4x Compiler Environment.....	4
<hr/>	
3 Programming PEROMs.....	5
3.1. Generating the Burn Program - Menu Interface.....	5
3.2. Generating the Burn Program - Command Line Interface.....	9
3.2.1. Programming the PEROM using Sysload Utility (WinNT Only).....	9
3.3. Programming The PEROM.....	12
3.4. PEROM Program Verification.....	13
<hr/>	
4 PEROMs, PEROM Utility, and the Sysload Utility	15
4.1. PEROM Memory Structure.....	15
4.2. PEROM Utility Files and Directory Structure.....	15
4.2.1. Program Files	17
4.2.2. Board Support Directory	17
4.2.3. TIM Support Directory.....	19
4.2.4. File Description	21
4.3. Sysload Utility Program Files and Directory Structure.....	22
<hr/>	
5 Special PEROM Programming Notes	25
5.1. LEMANS Carrier Board	25
5.2. CV6 Carrier Board	26
5.3. MDC40MD TIM-40 Module.....	26

List of Figures

Figure 1 PEROM Memory Structure 15

List of Tables

Table 1 Supported Carrier Boards 18
Table 2 Supported TIM-40 Modules 20
Table 3 Module Sub-Directory Contents 21

1 Introduction

1.1. Purpose Of This Manual

This manual provides the information you need to use the PEROM Utility to program PEROMs on Spectrum TIM-40 products. It will guide you through installation of the PEROM Utility software, Sysload Utility software (WinNT only), generation of the C4x burn program using both the menu and command line interface, and the final programming and verification of the PEROM. Further details are also given about TIM-40 IDROM memory structure used in the PEROM, and about the PEROM Utility itself.

Refer to the *Special PEROM Programming Notes* at the end of this guide if you are using non-standard amounts of DRAM on one of the following:

- LEMANS carrier board,
- CV6 carrier board, or
- MDC40MD TIM-40 module

1.2. Overview

Spectrum's PEROM Utility generates the *C4x burn programs* which are used to program the PEROMs of Spectrum's range of TIM-40 modules. This program is generated from two sources:

- The appropriate IDROM data for the particular TIM-40 module and carrier board combination as defined in Texas Instruments' TIM-40 TMS320C4x Module Specification Version 1.01.
- Optional user boot code in the form of a **.out** COFF0 format file

The IDROM data must be programmed into every PEROM and is provided by the PEROM Utility. The user boot code is optional and is supplied by the user.

After the C4x burn program is generated, a JTAG debugger or host application interface is used to run the C4x burn program on the TIM's C4x processor to program the PEROM.

1.3. Referenced Documents

This guide references the following documents:

- Texas Instruments TMS320C3x/C4x Code Generation Tools - Getting Started Guide, Release 5.00
- Texas Instruments TIM-40 TMS320C4x Module Specification Version 1.01

1.4. Definitions And Acronyms

The following terms are used throughout this document:

Board	In this document, a board capable of supporting one or more TIM-40 modules. Also known as a TIM-40 Carrier Board.
C4x	Texas Instruments' TMS320C40 or TMS320C44 Digital Signal Processor
COFF	Common Object File Format
DRAM	Dynamic Random Access Memory
DSP	Digital Signal Processor
IDROM	Identification ROM allowing storage of manufacture data, TIM-40 setup data, and user code
LSI	Loughborough Sound Images
PEROM	Flash Programmable and Erasable ROM
ROM	Read Only Memory
SIMM	Single Inline Memory Module
TIM	Within this document, the shortened form of "TIM-40 Module".
TIM-40	Texas Instruments Module (TIM) specification

2 Installation

In order to program PEROMs on your development system you must set up the C4x compiler environment and install the TIM-40 PEROM Utility software on your computer. The PEROM Utility can be run on the following platforms:

- 386 PC or higher with MS-DOS version 3 or later
- Solaris 2.x for the Sun SPARC4

The PEROM Utility is contained on two 3½” high density disks; one for MS-DOS-based systems and one for Solaris systems.

2.1. MS-DOS PEROM Utility Installation

Follow these steps to install the PEROM Utility software on a MS-DOS-based computer:

1. Insert the PEROM Utility (DOS) disk into your computer’s 3½” floppy drive.
2. Copy the contents of the disk to the working directory of your development system. At the MS-DOS prompt, type:

```
c:\>xcopy /s a:\*.* c:\working_directory
```

2.2. Solaris 2.x PEROM Utility Installation

Follow these steps to install the PEROM Utility software on a Solaris system:

Note: This installation assumes that the 3½” floppy drive on your system is configured as drive `/dev/diskette`. Refer to your Solaris documentation for further information if it is not.

1. Change the directory to the working directory of your development system:

```
% cd /working_directory
```

2. Make sure you have correct permissions to write to the installation directory:

```
% cd ..  
% chmod 777 working_directory  
% cd working_directory
```

3. Log into the designated machine as user **root**:

```
% su root
```

4. If Volume Management is running on your system, then turn it off by entering the following commands:

```
# /etc/init.d/volmgt stop
```

If you do not, you will not be able to access the floppy drive.

5. Insert the PEROM Utility (Solaris 2.x) disk into the 3 1/2" floppy drive (/dev/diskette).
6. Extract the files from disk using the following commands:

```
# tar xf /dev/diskette
# eject diskette
# /etc/init.d/volmgt start
# exit
% uncompress -v spectrum.tar.z
% tar xvof spectrum.tar
% rm spectrum.tar.z
```

2.3. Setting Up The C4x Compiler Environment

The PEROM Utility works in conjunction with **asm30** and **lnk30** to generate the C4x burn program. These files are supplied with the TMS320C4X Microprocessor Development System.

The path and environment variables of these files must be set up. Refer to the *Getting Started Guide* of Texas Instruments' TMS320C3x/C4x Code Generation Tools, Release 5.00, for information on setting up these software tools.

Note: The environment variable **C_DIR** must contain the path where **rts40.lib** can be found. This is usually the path containing **asm30** and **lnk30**.

3 Programming PEROMs

3.1. Generating the Burn Program - Menu Interface

Once the PEROM Utility is installed in the development directory, you can generate the PEROM C4x burn program file using the PEROM Utility's text-based menu interface.

1. Go to the directory where you installed the PEROM Utility.
2. Run the PEROM Utility by entering the command:

```
# perom
```

3. Select the type of carrier board that you are using from the following menu:

```
PEROM Utility - Version 2.3
(c) 1999 Spectrum Signal Processing Inc.

BOARD MENU
[a] maranello
[b] dakar
[c] lemans
[d] cv2
[e] cv4
[f] cv6
[g] qpc40b
[h] qpc40s
[i] Quit PEROM Utility

Enter Choice:
```

Example: Enter "a" to select the MARANELLO carrier board.

4. If different carrier board configurations are available the following menu is displayed. Select the configuration you are using from this menu:

```
MARANELLO MENU
[a] maranello : maranello_S1_60 Quad C4x Carrier Board for VME (512KB)
[b] maranello : maranello_S2_60 Quad C4x Carrier Board for VME (2 MB)
[c] maranello : maranello_S3_60 Quad C4x Carrier Board for VME (8 MB)
[d] Quit PEROM Utility

Enter Choice:
```

Example: Enter "b" to select a MARANELLO_S2-60 quad C4x VME carrier board equipped with 2 MB of memory.

5. Select the type of TIM you are using from the following menu:

```
TIM MENU
[a] dab
[b] ddc
[c] sdr
[d] md
[e] ss
[f] st
[g] ta
[h] lemans_emb
[i] dakar_emb
[j] Quit PEROM Utility

Enter Choice:
```

Example: Enter “f” to select an MDC44ST TIM-40 module.

6. If different TIM configurations are available the following menu is displayed. Select the configuration you are using from this menu:

```
ST MENU
[a] 44st3_50 : MDC44ST3_50 Super Twin Dual 50 MHz C44s (1 MB per C44)
[b] 44st3_60 : MDC44ST3_60 Super Twin Dual 60 MHz C44s (1 MB per C44)
[c] 44st3550 : MDC44ST3.5_50 Super Twin Dual 50 MHz C44s (2.5 MB per C44)
[d] 44st4_50 : MDC44ST4_50 Super Twin Dual 50 MHz C44s (4 MB per C44)
[e] 44st3560 : MDC44ST3.5_60 Super Twin Dual 60 MHz C44s (2.5 MB per C44)
[f] 44st4_60 : MDC44ST4_60 Super Twin Dual 60 MHz C44s (4 MB per C44)
[g] Quit PEROM Utility

Enter Choice:
```

Example: Enter “c” to select a 50 MHz MDC44ST equipped with 2.5 MB of memory per C44.

7. If the selected TIM has a global bus connector, the following menu is displayed. Select the site in which the TIM will be installed from this menu:

```
TIM SITE MENU
[a] Site A - global bus connector
[b] Site B - global bus connector
[c] Site C - global bus connector
[d] Site D - global bus connector
[e] Quit PEROM Utility

Enter Choice:
```

Example: Enter “a” to select TIM-40 site A on the LEMANS carrier board.

Note: The Utility needs to know whether the TIM will be placed on a site with a global bus connector in order to set up board shared resources. Each site option also contains a comment describing whether the particular site contains a global bus connector, no global bus connector, or an embedded TIM. If the TIM has no global bus connector or it is placed on a site without a global bus connector, board shared resources are automatically disabled.

8. If the selected TIM has more than one CPU the following menu is displayed. Select which CPU to use from this menu:

```
TIM CPU MENU
[a] CPU 1
[b] CPU 2
[c] Quit PEROM Utility

Enter Choice:
```

Example: Enter "a" to select CPU 1.

9. If the TIM has a global bus connector on the selected CPU and is placed on a site with a global bus connector, the following prompt is displayed. If the carrier board **does not have** shared resources, enter "n". If it **does have** shared resources, then you can enable access to them by entering "y", or disable access to them by entering "n".

```
Enable access to board shared resources on CPU 1? (y/n)
```

Example: Enter "y" to enable board shared resources for CPU 1.

10. If board shared resources were enabled in the previous step, the following prompt is displayed. Enter "y" to enable access to board shared memory, or "n" to disable access to board shared memory:

```
Enable access to board shared memory on CPU 1? (y/n)
```

Example: Enter "y" to enable access to board shared memory for CPU 1.

Note: On some multi-processor DSP kernels, such as 3L's Parallel C, only the root processor can have access to the board's shared memory; therefore, only the root TIM site can have board shared memory enabled.

11. If boot code is being programmed, specify the boot code COFF path and file name. If boot code is not being programmed, just press <Enter>:

```
Enter user boot code COFF filename [none]
:
```

Note: The boot code must be in a COFF0 format file.

12. Specify a name for the PEROM outfile and the path where you want it placed. Press <Enter>:

```
Enter perom outfile name [gsl/hardware/tim/st/44st3_50/44st3_50.out]
:
```

Example: The default path and filename is "gsl/hardware/tim/st/44st3_50/44st3_50.out".

If you do not specify a path and file name, the **.out** file will be placed in the default directory indicated within the square brackets of the displayed prompt. This is the PEROM Utility's TIM configuration file directory for the selected TIM.

The following type of messages will be displayed as the **.out** file is generated:

```
Building Perom outfile...
TMS320C3x/4x COFF Assembler      Version 5.00
Copyright (c) 1987-1997 Texas Instruments Incorporated
PASS 1
PASS 2

No Errors, No Warnings
TMS320C3x/4x COFF Assembler      Version 5.00
Copyright (c) 1987-1997 Texas Instruments Incorporated
PASS 1
PASS 2

No Errors, No Warnings
TMS320C3x/4x COFF Linker         Version 5.00
Copyright (c) 1987-1997 Texas Instruments Incorporated
```

Note: If a series of **Bad command or file name** error messages are displayed after pressing the <Enter> key, then your C4x compiler environment has not been set up properly. Go back to the Installation section in this guide.

13. If the TIM contains more than one DSP the following prompt is displayed.
Enter “y” if you want to build another outfile for the same board, TIM, and site using a different CPU.
Enter “n” to go to the next prompt.
Enter “q” to quit the PEROM Utility program.

```
Build PEROM outfile for another CPU? (y/n/q)
```

14. If the TIM has a global bus connector the following prompt is displayed.
Enter “y” if you want to build another outfile using a different site for the same board and TIM.
Enter “n” to go to the next prompt.
Enter “q” to quit the PEROM Utility program.

```
Build PEROM outfile for another Site? (y/n/q)
```

15. Enter “y” if you want to build another outfile using a different TIM for the same board.
Enter “n” to quit the PEROM Utility program.

```
Build PEROM outfile for another Tim? (y/n)
```

3.2. Generating the Burn Program - Command Line Interface

The PEROM Utility has a command line interface to allow other programs, such as System Definition Tool, make, batch, and script files to call the PEROM Utility.

Syntax `perom [-B board_subtype board_model -T tim_subtype tim_model
[-S site] [-P processor] [-G] [-M] [-U usercode] [-O outfile]] [-H]`

Switch	Description	Default Setting
-B	Carrier board subtype followed by model type	n/a
-T	TIM-40 module subtype followed by model type	n/a
-S	Site of TIM on board (A through H)	A
-P	processor on TIM (1 or 2)	1
-G	Enables global bus if available on selected site	global bus disabled
-M	Enable access to the board's shared memory if global bus is enabled.	Access to shared memory disabled
-U	User boot code path and filename (in COFF0 format)	none
-O	Out file path and filename	gsl/hardware/tim/tim_subtype/ tim_model/tim_model.out
-H	Help/usage information	

UNIX Example

```
perom -B maranello maranello_s1_60 -T ss 40ss2_60 -S c -P 1 -G -M -U  
/user/flash44.out -O /gsl/hardware/board1/siteA/procl/44ss2_60.out
```

MS-DOS Example

```
perom -B maranello maranello_s1_60 -T ss 40ss2_60 -S c -P 1 -G -M -U  
\user\flash44.out -O \gsl\hardware\board1\siteA\procl\44ss2_60.out
```

3.2.1. Programming the PEROM using Sysload Utility (WinNT Only)

The Sysload Utility further simplifies the process of programming the PEROM for Maranello (CV8), Dakar (F5) and Lemans (VX8) carrier boards. Like the PEROM Utility, the Sysload Utility has a command line interface to allow other programs, such as System Definition Tool, make, batch and script files to call the PEROM Utility.

The PEROM Utility must be run before the SYSLOAD Utility to create the burn file.

Syntax `XXX_sysload [-P proc_or_ldf -D sdf_file -F out_file][-H]`
XXX = CV8, F5 or VX8

Switch	Description
-P	Complete path to the target DSP processor or complete path and filename to the LDF file
-F	Complete path and filename to the OUT file
-D	Complete path and filename to the SDF file
-H	Help/Usage Information

Example using OUT file.

```
f5_sysload -P Board1:NodeB:Proc0 -F burn.out -D f5.sdf
```

Example using LDF file.

```
f5_sysload -P load.ldf -D f5.sdf
```

Running the example batch files

Examples which use both the PEROM and SYSLOAD utility are included. Each batch file creates, then downloads a burn file to a specified module and carrier board (either Dakar, LeMans, or Maranello).

Before running each example make the following preliminary steps:

1. Make sure the host SDK's are installed for the Dakar (F5), Maranello (CV8), and LeMans (VX8) Spectrum DSP carrier boards.
2. Make sure the DSP carrier boards are configured correctly, (such as, default settings for the board numbers).

For example, the batchfile F5SS2_60 .BAT is located in the SSP\SYSLOAD\WinNT\F5\Examples\SS directory. It has been shown here to illustrate the most important sections. You may use the examples as a guide to create your own batch files or scripts.

All the batch files have the following basic usage:

Syntax *XXXXX.BAT TIM_Site Processor SDF_File*
Where XXXXX describes the carrier board type, TIM module type, memory and speed configuration.

Parameter	Description	Possible Values
TIM_Site	Site of SS TIM module	B, C or D.
Processor	target processor on ST TIM module ¹	0 or 1;
SDF_File	Filename and path to SDF file.	

The batch file first sets several environment variables for the EXAMPLE, PEROM, SYSLOAD directories, SDF File and TIM SITE location. If the user does not specify an SDF file in the command line, the batch file selects the default SDF located in the examples directory.

```

REM *** DIRECTORIES ***
SET EXAMPLE_DIR=EXAMPLES\SS\
SET PEROM_DIR=..\..\..\perom\BIN\
SET SYSLOAD_DIR=..\..\SYSLOAD\WinNT\f5\bin

REM *** CHECK PARAMETERS ***
if "%1" == "b" goto param2
if "%1" == "c" goto param2
if "%1" == "d" goto param2

if "%1" == "B" goto param2
if "%1" == "C" goto param2
if "%1" == "D" goto param2
goto error

:param2
if "%2" == "" goto defaultSDF

REM *** SET MISC. VARIABLES ***
SET SDF_FILE=%2
goto continue

:defaultSDF
SET SDF_FILE=..\EXAMPLE_DIR%f5ss2_60.sdf

:continue
SET TIM_SITE=%1
SET TARGET_PROC=Board1:Node%TIM_SITE%:Proc0

```

The batch/script calls the PEROM utility. Remove the REM statement for a specific carrier board.

¹ This parameter is only needed when using the ST TIM module

```

REM *** CREATE BURN CODE WITH PEROM UTILITY ***
cd ..\..\%PEROM_DIR%

REM < Trio C4x Carrier Board for PCI (512K) >
perom -b DAKAR DAKAR_M1S1 -t SS 40SS2_60 -s %TIM_SITE% -p 1 -g -o .\40ss2_60.out

REM < Trio C4x Carrier Board for PCI (2 MB) >
REM perom -b DAKAR DAKAR_M2S2 -t SS 40SS2_60 -s %TIM_SITE% -p 1 -g -o
.\40ss2_60.out
    
```

The batch/script calls the SYSLOAD utility for Dakar (F5) with the proper SDF file and target processor. The burn files are then deleted from disk.

```

REM *** DOWNLOAD BURN CODE WITH SYSLOAD UTIL ***
cd %SYSLOAD_DIR%
F5_SYSLOAD -p %TARGET_PROC% -f ..\%PEROM_DIR%40SS2_60.out -d %SDF_FILE%

REM *** DELETE TEMPORARY BURN FILES ***
cd ..\%PEROM_DIR%
if exist 40ss2_60.out del 40ss2_60.out
if exist 40ss2_60.map del 40ss2_60.map
    
```

3.3. Programming The PEROM

Once the C4x burn program has been generated for the required TIM by the PEROM Utility, it is used to program the IDROM/boot code into the TIM's PEROM. The C4x burn program filename has a **.out** extension and corresponds to the specific TIM-40 module type, such as **44ddc150.out**, **44ddc450.out**, or **40ta_50.out**. It is located in the corresponding TIM-40 Module sub-directory by default. For example, the C4x burn program for a TA TIM-40 module sited on a CV4 Carrier Board, **40ta_50.out**, would be found in the directory:

```

gsl\hardware\tim\ta\40ta_50          (under MS-DOS)
gsl/hardware/tim/ta/40ta_50         (under Solaris)
    
```

For example, the following command sequence uses Texas Instruments' Source Debugger to burn the PEROM on a TIM-40 module with the C4x burn program. Perform this procedure for each PEROM being programmed.

Enter...	Action.
Db40 -n <i>processor_id</i>	Runs the TI debugger Utility. <i>processor_id</i> is the C4x processor of the PEROM you are programming.
load <i>burn_program</i>	Downloads the appropriate C4x burn program to the C4x processor where <i>burn_program</i> is the name of the C4x burn program. For example, the <i>burn_program</i> for the M40ta_50DM TIM, would be: <i>Under MS-DOS:</i> gsl\hardware\tim\ta\40ta_50\40ta_50.out <i>Under Solaris:</i> gsl/hardware/tim/ta/40ta_50/40ta_50.out

Enter...	Action.
run or press the <F5> key press the <ESC> key	Executes the loaded C4x burn program. Wait 5 seconds to allow the data to be burnt into the PEROM.
quit	Quits the Texas Instruments' Source Debugger. The C4x PEROM is now programmed.

3.4. PEROM Program Verification

The PEROM Utility's verification function verifies that the PEROM was programmed successfully. The verification result is stored in the DSP's memory at address 0x002F FFFF. The lower byte contain the comparison results as shown in the following table:

Binary	Hex	Description
0000 0001	0x01	IDROM data successfully written to PEROM
0000 0010	0x02	IDROM data unsuccessfully written to PEROM
0000 0100	0x04	User code data successfully written to PEROM
0000 1000	0x08	User code data unsuccessfully written to PEROM

A combination of these verification results is also possible. For example, a result of 0x05 (0000 0101 in binary) shows that IDROM data was successfully written to the PEROM and that user code data was successfully written to the PEROM.

4 PEROMs, PEROM Utility, and the Sysload Utility

4.1. PEROM Memory Structure

Two memory structures for the PEROM are possible. The PEROM can have the IDROM data only, or the user code and the IDROM data. Normally IDROM data starts at the base address of the PEROM. If user boot code is included, it appears first at the PEROM base address and is followed by the IDROM data. This allows the C4x to boot from PEROM when the TIM is configured to boot from the PEROM. See Figure 1.

Note: The base address for IDROM data as defined in the TIM-40 specifications is 0x7000 0000. The base address for user boot code is 0x4000 0000. Therefore, the PEROM address space is often decoded to both base addresses on TIM-40 modules.

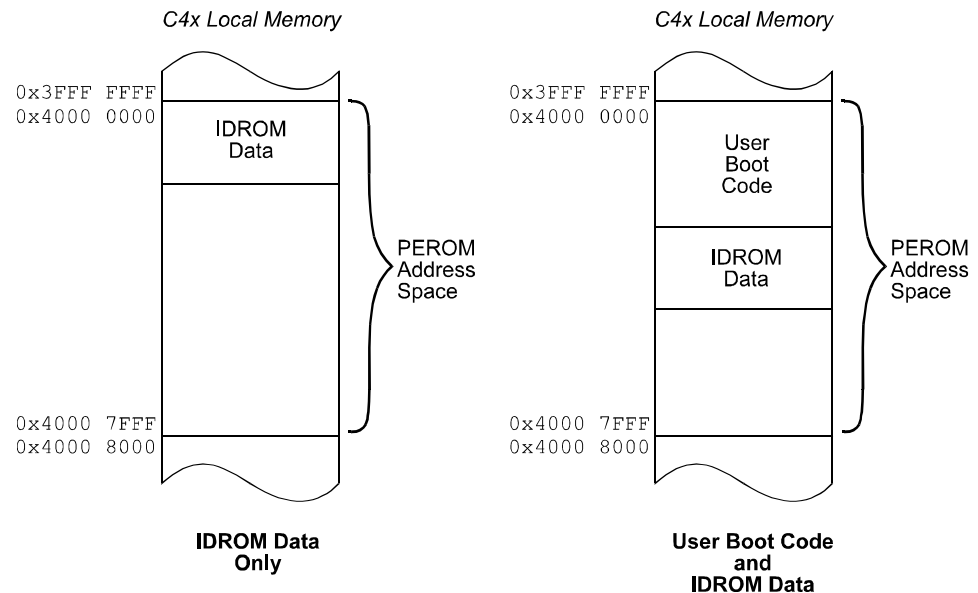


Figure 1 PEROM Memory Structure

4.2. PEROM Utility Files and Directory Structure

When installed on your system, the PEROM Utility program file resides in the `/ssp` directory as shown in the following directory listing. Program files are located in the `/ssp/bin` directory; board and TIM support files are located in the `/ssp/gsl/hardware`

directory (Global System Library hardware); the SYSLOAD utility and example batch files are located in the `/ssp/sysload` directory.

```
/ssp
  /perom
    /bin
      perom.exe <DOS>
      perom.ini

    /gsl
      /hardware
        /board
          /board_subtype
            /board_modeltype
              board.cfg
          /board_modeltype
            board.cfg

                                and so on for each board...

      /tim
        /tim_subtype
          boot.asm
          boot.obj
          burn.obj
          idrom.cmd
          /tim_modeltype
            tim.cfg
          /tim_modeltype
            tim.cfg

                                and so on for each tim_subtype...

      /sspboot
        /dakar
          sspboota.out
        /maranello
          ssboot40.out
          ssboot44.out

    /sysload
      /WinNT
        /CV8
          /bin
            cv8_sysload.exe
            blink.out
            sspboot.out
          /examples
            /dab
              cv8dab_6.bat
              cv8dab_6.sdf
            /sdr
            /ss
            /st

                                more batch files and SDF

      /F5
                                similar to CV8 structure

      /VX8
                                similar to CV8 structure
```

4.2.1. Program Files

The following PEROM Utility program files are placed in the **/ssp/bin** directory.

perom.exe	Program file for DOS-based development systems
perom	Program file for UNIX-based development systems
perom.ini	Initialization file for the PEROM Utility. It contains a database of board and TIMs used in creating the various menus and paths to board and TIM configuration files. As new boards and TIMs are added to Spectrum's product line, only the PEROM initialization file (along with the addition of appropriate board and TIM directories) needs to be updated.

4.2.2. Board Support Directory

The **/board** directory contains the board configuration files, **board.cfg**, for the supported carrier boards. They are kept in the directory:

```
/ssp/gsl/hardware/board/sub_type /model_type /board.cfg
```

sub_type is the name of a board supported by this utility, such as CV4 or LEMANS.

model_type is the particular configuration of the board, such as LEMANS_S_60 or LEMANS_SA_60.

Table 1 Supported Carrier Boards

<i>sub_type</i>	<i>model_type</i>	Description
MARANELLO	MARANELLO_S1_60	Quad C4x Carrier Board for VME (512K)
	MARANELLO_S2_60	Quad C4x Carrier Board for VME (2 MB)
	MARANELLOS3-60	Quad C4x Carrier Board for VME (8 MB)
DAKAR	DAKAR_M1S1	Trio C4x Carrier Board for PCI (512K)
	DAKAR_M2S2	Trio C4x Carrier Board for PCI (2 MB)
LEMANS	LEMANS_S_60	Octal Embedded Board for VXI (no DRAM)
	LEMANS_SA_60	Octal Embedded Board for VXI (4 MB DRAM)
	LEMANS_SB_60	Octal Embedded Board for VXI (8 MB DRAM)
	LEMANS_SC_60	Octal Embedded Board for VXI (32 MB DRAM)
	LEMANS_SD_60	Octal Embedded Board for VXI (64 MB DRAM)
CV2	CV2	CV2S1_[40 50] Dual C4x Carrier Board for VME (512K)
CV4	CV4	CV4_[50 60] Quad C4x Carrier Board for VME (no RAM)
CV6	CV6_A	CV6_S[1 2]D0 Dual C4x Carrier Board for VME (no DRAM)
	CV6_B	CV6_S[1 2]D1 Dual C4x Carrier Board for VME (16 MB DRAM)
	CV6_C	CV6_S[1 2]D2 Dual C4x Carrier Board for VME (64 MB DRAM)
QPC40	QPC40B	QPC40B Quad C4x TIM Carrier Board for AT (no RAM)
QPC40S	QPC40S_0	QPC40S_0[50 60] Quad C4x Carrier Board for AT (no RAM)
	QPC40S_1	QPC40S_1[50 60] Quad C4x Carrier Board for AT (512K)

Note: Use the LSI programming tool to program the embedded PEROM on site A of the CV6 carrier board and the PEROM on site A of the CV2 carrier board.

4.2.3. TIM Support Directory

The **/tim** directory contains PEROM support files for TIM-40 modules. The **boot.asm**, **boot.obj**, **burn.obj**, and **idrom.cmd** files for each TIM subtype are kept in */ssp/gsl/hardware/tim/sub_type /*

The configuration files for each different configuration of a TIM (**tim.cfg**) are kept in *gsl/hardware/tim/sub_type/model_type/*

sub_type is the name of a TIM supported by this utility, such as DDC or ST

model_type is the particular configuration of the TIM, such as 44DDC450 or 44ST3-60.

Table 2 Supported TIM-40 Modules

<i>sub_type</i>	<i>model_type</i>	Description
DAB	44DAB_60	MDC44DAB_60 Data Acquisition Board 60 MHz C44 (32.5 MB)
DDC	44DDC150	MDC44DDC1_50 Digital Down Converter Dual (512K per C44)
	44DDC450	MDC44DDC4_50 Digital Down Converter Dual (2 MB per C44)
SDR	44SDR60	MDC44SDR_60 Single Digital Drop Receiver 60 MHz C44 (512K)
MD	40MD_50	MDC40MD_50 50 MHz C40 (16K SRAM + DRAM)
SS	40SS2_60	MDC40SS2_60 Super SRAM 60 MHz C40 (1.5 MB)
	40SS4_60	MDC40SS4_60 Super SRAM 60 MHz C40 (2 MB)
	40SS5_50	MDC40SS5_50 Super SRAM 50 MHz C40 (4 MB)
	40SS6_50	MDC40SS6_50 Super SRAM 50 MHz C40 (8 MB)
	40SS5_60	MDC40SS5_60 Super SRAM 60 MHz C40 (4 MB)
	40SS6_60	MDC40SS6_60 Super SRAM 60 MHz C40 (8 MB)
ST	44ST3_50	MDC44ST3_50 Super Twin Dual 50 MHz C44s (1 MB per C44)
	44ST3_60	MDC44ST3_60 Super Twin Dual 60 MHz C44s (1 MB per C44)
	44ST3550	MDC44ST3.5_50 Super Twin Dual 50 MHz C44s (2.5 MB per C44)
	44ST4_50	MDC44ST4_50 Super Twin Dual 50 MHz C44s (4 MB per C44)
	44ST3560	MDC44ST3.5_60 Super Twin Dual 60 MHz C44s (2.5 MB per C44)
	44ST4_60	MDC44ST4_60 Super Twin Dual 60 MHz C44s (4 MB per C44)
TA	40TA_50	MDC40TA_50 ADC Dual 50 MHz C40s (256K per C40)
	40TA1_60	MDC40TA1_60 ADC Dual 60 MHz C40s (64K per C40)
	40TA2_60	MDC40TA1_60 ADC Dual 60 MHz C40s (128K per C40)
	40TA3_60	MDC40TA1_60 ADC Dual 60 MHz C40s (256K per C40)
	40TA4_60	MDC40TA1_60 ADC Dual 60 MHz C40s (512K per C40)
LEMANS_EMB	40LEMANS_A_1	LEMANS_S1_60 Embedded Node A 60 MHz C40 (1 MB)
	40LEMANS_A_2	LEMANS_S2_60 Embedded Node A 60 MHz C40 (4 MB)
	40LEMANS_B_1	LEMANS_S1_60 Embedded Node B 60 MHz C40 (1 MB)
	40LEMANS_B_2	LEMANS_S2_60 Embedded Node B 60 MHz C40 (4 MB)
DAKAR_EMB	44DAKAR_A_1	DAKAR_M1S1 Embedded Node A 60 MHz C44 (1 MB)
	44DAKAR_A_2	DAKAR_M2S2 Embedded Node A 60 MHz C44 (4 MB)
	44DAKAR_A_3	DAKAR_M2S3 Embedded Node A 60 MHz C44 (10 MB)

Notes: Boot initialization code **must** be programmed into the embedded PEROMs on site A of LEMANS and DAKAR carrier boards. Refer to the relevant carrier board manuals.

Use the LSI programming tool to program the embedded PEROM on site A of the CV6 carrier board and the PEROM on site A of the CV2 carrier board.

4.2.4. File Description

Table 3 describes the files required or generated when the PEROM Utility is run. The location under the `/ssp/gsl/hardware` directory for each of these files is also given.

Table 3 Module Sub-Directory Contents

File Name	Location	Description
board.cfg	../board /sub_type /model_type/	Carrier board information file required by the utility. This file contains board-dependent IDROM information and site status information. Each site on a board has an entry in the configuration file indicating if the site: <ul style="list-style-type: none"> • Exists • Has a global bus connector • Contains an embedded TIM. This information is used in constructing the menu of selectable TIM sites as well as the option of enabling the board shared resources. The remaining board.cfg information describes board-dependent IDROM fields and board-dependent GMCR values.
boot.asm	../tim /sub_type/	C4x assembly code needed to produce boot.obj which is used by the PEROM Utility. It contains the C4x boot routine program used to call the user boot code program and/or the PEROM burn and verify program. Use the following command from a UNIX or MS-DOS shell to produce boot.obj . asm30 boot.asm -o boot.obj -v40
boot.obj	../tim /sub_type/	Boots the C4x on the TIM-40 module in order to program the PEROM. If any changes are made to the boot.asm , it must be re-assembled to produce the new boot.obj object file.
burn.obj	../tim /sub_type/	Burn and verify program C4x object file containing the code to write IDROM data and/or user code into the PEROM, and to verify that the data was successfully written.
idrom.cmd	../tim /sub_type/	C4x Linker command file required by the utility to compile the user boot code and IDROM information for a TIM-40 module. It contains linker options, memory maps, and sections required to link all object files and produce the executable PEROM outfile, tim.out .
tim.cfg	../tim /sub_type/	TIM-40 module information file required by the utility. It contains TIM-dependent IDROM information and global bus connector status information.

File Name	Location	Description
<i>tim_model.out</i> *	../tim	C4x executable COFF file containing all necessary code to program the PEROM of a chosen board and TIM. This file is used to burn the PEROM on the TIM-40 module.
<i>tim_model.map</i> *	/sub_type/ /model_type	
<i>tim_model.asm</i> *	(default) or user specified location	
		By default, this file is called tim_model.out and is located in the chosen TIM's <i>model_type</i> directory (<i>tim_model</i> corresponds to the TIM's modeltype directory name). The utility allows you to enter an alternate path and file name for this output file.
		Along with the out file, tim_model.map and tim_model.asm are also created in the TIM's <i>tim_model</i> directory. These are respectively, the linker map file and IDROM assembly file produced from the selected board and TIM configuration files.
		For example, 44ddc450.out , 44ddc450.map , and 44ddc450.asm would be produced for the 44DDC450 TIM.

*Files generated by PEROM Utility, assembler, or linker; not included with installation software.

If an error occurs while executing the PEROM Utility, one or more of the following temporary files may remain in the PEROM working directory. These files are removed upon the next successful run of the PEROM Utility.

_boot.obj	Boot routine object file copied from selected TIM subtype directory
_burn.obj	Burn program object file copied from selected TIM subtype directory
coff2asm.asm	COFF parsed assembly file produced if user code was specified
_nocode.asm	Assembly file produced if no user code was specified
_rom.obj	IDROM object file produced from selected board and TIM configuration files
_user.obj	User code object file produced by assembling <i>coff2asm.asm</i> (user code specified) or <i>_nocode.asm</i> (no user code specified)

4.3. Sysload Utility Program Files and Directory Structure

The Sysload utility and example batch files are located in the **/ssp/sysload** directory.

The following Sysload utility program and example files for WinNT are placed in the **/ssp/WinNT/board_type/** directory.

board_type is the name of a board supported by this utility, such as F5, CV8 and VX8.

Within each board type directory is a **/bin** and **/examples** directory which contains the program and example files for that particular board.

As an example, the following describes the Sysload Utility files for the CV8 carrier board.

In **cv8/bin** directory:

cv8_sysload.exe	Sysload program file for CV8 carrier board. See section 3.2.1 for details.
blink.out	DSP file for Flash LED example
sspboot.out	DSP boot file for CV8 carrier board. See CV8 SDK documentation for details.

In **cv8/examples/dab** directory:

cv8dab_6.bat	Batch file creates, then downloads, a DAB (Data Acquisition Board) burn file to a DAB module on a Maranello (CV8) Spectrum DSP Carrier Board.
cv8dab_6.sdf	System Definition file for the DAB module on the CV8 carrier board.

In **cv8/examples/sdr** directory:

cv8sdr_6.bat	Batch file creates, then downloads, a SDR (Signal Digital Drop Receiver) burn file to a SDR module on a Maranello (CV8) Spectrum DSP Carrier Board.
cv8sdr_6.sdf	System Definition file for the SDR module on the CV8 carrier board.

In **cv8/examples/ss** directory:

cv8ss2_6.bat	Batch file creates, then downloads, a SS (Super SRAM 60 MHz C40, 1.5 MB) burn file to a SS module on a Maranello (CV8) Spectrum DSP Carrier Board.
cv8ss2_6.sdf	System Definition file for the SS module on the CV8 carrier board.

plus several other batch and SDF files for different SS module configurations

In **cv8/examples/st** directory:

cv8st3_5.bat	Batch file creates, then downloads, a ST (Super Twin Dual 50 MHz C44s, 1 MB per C44) burn file to a ST module on a Maranello (CV8) Spectrum DSP Carrier Board.
cv8st3_5.sdf	System Definition file for the ST module on the CV8 carrier board.

plus several other batch and SDF files for different SS module configurations

The directory structure and files for the Dakar (F5) and LeMans (VX8) Sysload Utility are similar to that of the CV8.

5 Special PEROM Programming Notes

The board and TIM configuration files provided with the PEROM Utility should generally not be modified as they contain standard data specific to the hardware. However, boards and TIMs using DRAM memory can be configured with many combinations of SIMM chips. The configuration files included with the PEROM Utility contain the standard memory configurations sold by Spectrum. The following procedures describe how to modify board and TIM configuration files for use with non-standard amounts of DRAM (those not already listed in the PEROM initialization file).

5.1. LEMANS Carrier Board

The LEMANS Carrier Board memory configurations include 0, 4, 8, 32, and 64 MB of DRAM. If your board contains another amount of DRAM, use the following procedure to create a new board configuration file.

1. Create a new TIM configuration directory by copying the **gsl\hardware\board\lemans\lemans_sa_60** directory and its contents to a new directory such as **gsl\hardware\board\lemans\lemans_ua_60** (if **lemans_ua_60** exists, use **lemans_ub_60** and so on).
2. Use an ASCII text-editor to open the new board configuration file (**..\lemans_ua_60\board.cfg**)
3. Modify the **GSize1** field which specifies the size of the board shared DRAM in words. It should currently contain **0x00100000** which is the hexadecimal value for 4 MB (megabytes) or 1 MW (megawords). For example, if your board contained a non-standard amount of DRAM such as 12 MB, **GSize1** would be set to **0x00300000**.
4. Save the new board configuration file.
5. Use an ASCII text-editor to open the PEROM initialization file, **perom.ini**, in the PEROM working directory.
6. Add a line for the newly created board configuration file using the **LEMANS_SA_60** line as an example. The Model Type field should contain the newly created board model type directory (**LEMANS_UA-60**) and the Description field should contain the new amount of memory.

5.2. CV6 Carrier Board

The CV6 Carrier Board memory configurations include 0, 16, and 64 MB of DRAM. If your board contains another amount of DRAM, use the following procedure to create a new board configuration file.

1. Create a new TIM configuration directory by copying the **gsl\hardware\board\cv6\cv6_b** directory and its contents to a new directory such as **gsl\hardware\board\cv6\cv6u_a** (if **cv6u_a** exists, use **cv6u_b** and so on).
2. Use an ASCII text-editor to open the copied board configuration file (**..\cv6u_a\board.cfg**)
3. Modify the **GSize1** field that specifies the size of the board shared DRAM in words. It should currently contain **0x00400000** which is the hexadecimal value for 16 MB (megabytes) or 4 MW (megawords). For example, if your board contained a non-standard amount of DRAM such as 12 MB, **GSize1** would be set to **0x00300000**.
4. Save the new board configuration file.
5. Use an ASCII text-editor to open the PEROM initialization file, **perom.ini**, in the PEROM working directory.
6. Add a line for the newly created board configuration file using the **CV6_B** line as an example. The Model Type field should contain the newly created board model type directory (**CV6U_A**) and the Description field should contain the new amount of memory.

5.3. MDC40MD TIM-40 Module

The MDC40MD TIM is configured for the maximum amount of DRAM it can hold, 128 MB. If your TIM contains another amount of DRAM, use the following procedure to create a new TIM configuration file.

1. Create a new TIM configuration directory by copying the **gsl\hardware\tim\md\40md_50** directory and its contents to a new directory such as **gsl\hardware\tim\md\mdua_50** (if **mdua_50** exists, use **mdub_50** and so on).
2. Use an ASCII text-editor to open the copied TIM configuration file (**..\mdua_50\tim.cfg**)
3. Modify the **LSize1** and **GSize0** fields which specify the size of the TIM local and global DRAM in words. **LSize1** and **GSize0** should currently contain **0x01000000** which is the hexadecimal value for 64 MB (megabytes) or 16 MW (megawords). For example, if your TIM contains 32 MB of DRAM on the local bus and 16 MB on the global bus, **LSize1** and **GSize0** would be set to **0x00800000** and **0x00400000** respectively.
4. Save the new TIM configuration file.

5. Use an ASCII text-editor to open the PEROM initialization file, **perom.ini** in the PEROM working directory.
6. Add a line for the newly created TIM configuration file using the **40MD_50** line as an example. The Model Type field should contain the newly created TIM model type directory (**MDUA_50**) and the Description field should contain the new amount of memory.