56F801x
Serial Bootloader User Guide

56F8000
16-bit Digital Signal Controllers

56F801xBLUG
Rev. 1.0
12/2005

freescale.com

freescale™
semiconductor

TABLE OF CONTENTS

LIST OF FIGURES

About This Document

This manual describes the 56F801x Serial Bootloader application.

Audience

This manual targets software developers utilizing the 56F801x Bootloader applications.

Organization

This manual consists of the following sections:

- **Chapter 1, Bootloader** -- describes the Serial Bootloader.
- **Chapter 2, License** -- provides the license required to use this product.

Suggested Reading

We recommend that you have a copy of the following references:

- *Freescale DSP56800E Reference Manual,* DSP56800ERM
- *56F8000 Peripheral Reference Manual*, MC56F8000RM
- *Inside CodeWarrior: Core Tools*, Freescale Inc.

Conventions

This document uses the following notational conventions:

Typeface, Symbol or Term	Meaning	Examples
`Courier Monospaced Type`	Code examples	`//Process command for line flash`
Italic	Directory names, project names, calls, functions, statements, procedures, routines, arguments, file names, applications, variables, directives, code snippets in text	...and contains these core directories: *applications* contains applications software... ...CodeWarrior project, *3des.mcp* is... ...the *pConfig* argument.... ...defined in the C header file, *aec.h*....
Bold	Reference sources, paths, emphasis	...refer to the **Targeting** 56F801x **Platform** manual.... ...see: **C:\Program Files\Motorola\Embedded SDK\help\tutorials**
Blue Text	Linkable on-line	...refer to Chapter 7, License....
Number	Any number is considered a positive value, unless preceded by a minus symbol to signify a negative value	3V -10 DES^{-1}
ALL CAPITAL LETTERS	# defines/ defined constants	# define INCLUDE_STACK_CHECK
Brackets [...]	Function keys	...by pressing function key [F7]
Quotation marks, “...”	Returned messages	...the message, “Test Passed” is displayed.... ...if unsuccessful for any reason, it will return “NULL”...

Definitions, Acronyms, and Abbreviations

The following list defines the acronyms and abbreviations used in this document. As this template develops, this list will be generated from the document. As we develop more group resources, these acronyms will be easily defined from a common acronym dictionary. Please note that while the acronyms are in solid caps, terms in the definition should be initial capped ONLY IF they are trademarked names or proper nouns.

COP	Computer Operating Properly
DSC	Digital Signal Controller
EVM	Evaluation module
FM	Flash Module
IDE	Integrated Development Environment
PC	Personal Computer
PLL	Phase Locked Loop
RAM	Random Access (read/write) Memory
SCI	Serial Communications Interface

References

The following sources were used to produce this book:

1. *Freescale DSP56800E Reference Manual,* DSP56800ERM
2. *56F8000 Peripheral Reference Manual*, MC56F8000RM
3. Freescale website:
   www.freescale.com

Chapter 1
Bootloader

1.1 Serial Bootloader

All 56F801x family devices are preprogrammed with the Serial Bootloader application. The Serial Bootloader has been developed to load and run a proprietary user application presented as an S-Record file into the Program Memory. The Serial Bootloader is located in the dedicated Program Memory region, called Program Flash. The Serial Bootloader supports the simplest serial protocol, so any standard serial terminal program can be used on the host PC.

The Serial Bootloader application reads the S-Record file of a user application (for example, generated by CodeWarrior) via serial interface, parses this S-Record file, and stores needed data in Program Flash memory. When the processing of the S-Record file is finished, the Bootloader will remain in an infinite loop. Resetting the device will allow the downloaded application to be launched. If any error occurs while loading the S-Record file, the Bootloader outputs an error message via the serial line and will prompt the user to resend the S-Record file. See **Figure 1-1**.

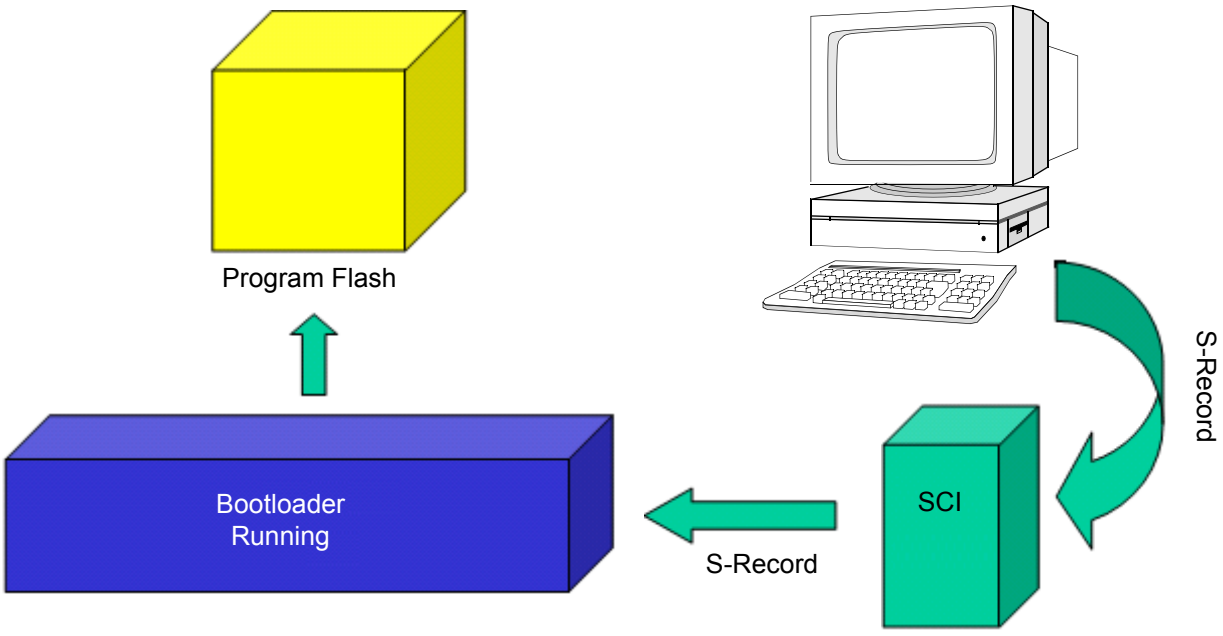

Figure 1-1. Programming Flash on a 56F801x Device

1.1.1 Start-up Sequence with Bootloader

The following steps describe the Bootloader's start-up sequence.

1.1.1.1 Step 1: Power-up/Reset

The 56800E core specifies two reset vectors: Hardware Reset and COP Reset. The reset vectors are each two words wide, and reside in consecutive addresses in the beginning of Program Flash. The Hardware Reset vector identifies the address the processor accesses after a power-on reset, external reset, or software reset. The COP Reset vector is accessed after a watchdog time-out. When the 56F801x Bootloader is present, both reset vectors will contain an entry point to the Serial Bootloader application. Once the Serial Bootloader has begun reprogramming the Flash, the entire Flash will be erased. Since this affects the reset vectors, it is critical that no resets are asserted until the Serial Bootloader application has concluded.

1.1.1.2 Step 2: Transfer Code from Program Flash to RAM

Before the Serial Bootloader application can execute, portions of the software must be moved from Program Flash to RAM. This is necessary because the Program Flash is to be erased and reprogrammed with new data. Only the portions of the Serial Bootloader application used to reprogram Flash with S-record data are transferred. Initialization code used by the Serial Bootloader application will be lost once the application has begun modifying Program Flash.

1.1.1.3 Step 3: Write New Application to Program Flash

Once the Serial Bootloader has processed a valid S0 S-record, it erases the entire Program Flash in anticipation of new data. Program Flash is written with new data each time a valid S3 S-record has been processed. The Serial Bootloader application will not reprogram any RAM locations. In the event of an error, a notification will be sent over the serial port. The application will flush all data associated with the current S-record file and wait for the user to send a new S-record file.

1.1.1.4 Step 4: Switch to New Application

Once the 56F801x Serial Bootloader has reprogrammed the Flash with the data from the S-record file, it enters an infinite loop to prevent further code execution. A hardware reset is necessary to begin executing the downloaded application. The reset causes the processor to begin executing code contained in the Hardware Reset vector (which should now contain a JMP instruction to an address used as the entry point for the new application). The reset also returns all peripheral registers to default values.

1.1.2 Bootloader History

The most current version of the Bootloader is 1.5. When running the Serial Bootloader application, the version number is contained in a message transmitted over the serial port. The Bootloader version history can be seen in the following table; see **Appendix B** for functionality differences between versions.

Known Issues:

a. A default terminal serial program line-ending character terminator is not supported. Work around through the serial application configuration. See **Appendix B** for the work-around.
b. XON/XOFF flow control requires 2ms line delays. Work around through the serial application configuration. See **Appendix B** for the work-around.
c. An odd number of data bytes in S3 Record will hang execution. A work-around PERL script is available on the Freescale FAQ website:
http://faqts.freescale.net
FAQ# 25759 contains the script, as well as a detailed explanation of the problem.

Table 1-1. Software Revision History

Version Number	Known Issues	Actions/Resolutions
1.2	a,b,c	
1.3	c	Corrected “a” and “b”
1.5		Corrected “c”

1.1.3 Files

The Serial Bootloader application was developed using CodeWarrior™ with Processor Expert™ (PE) and makes use of the standard files generated for PE projects. Some of the standard PE files were modified, however, to support special functionality needed by the application.

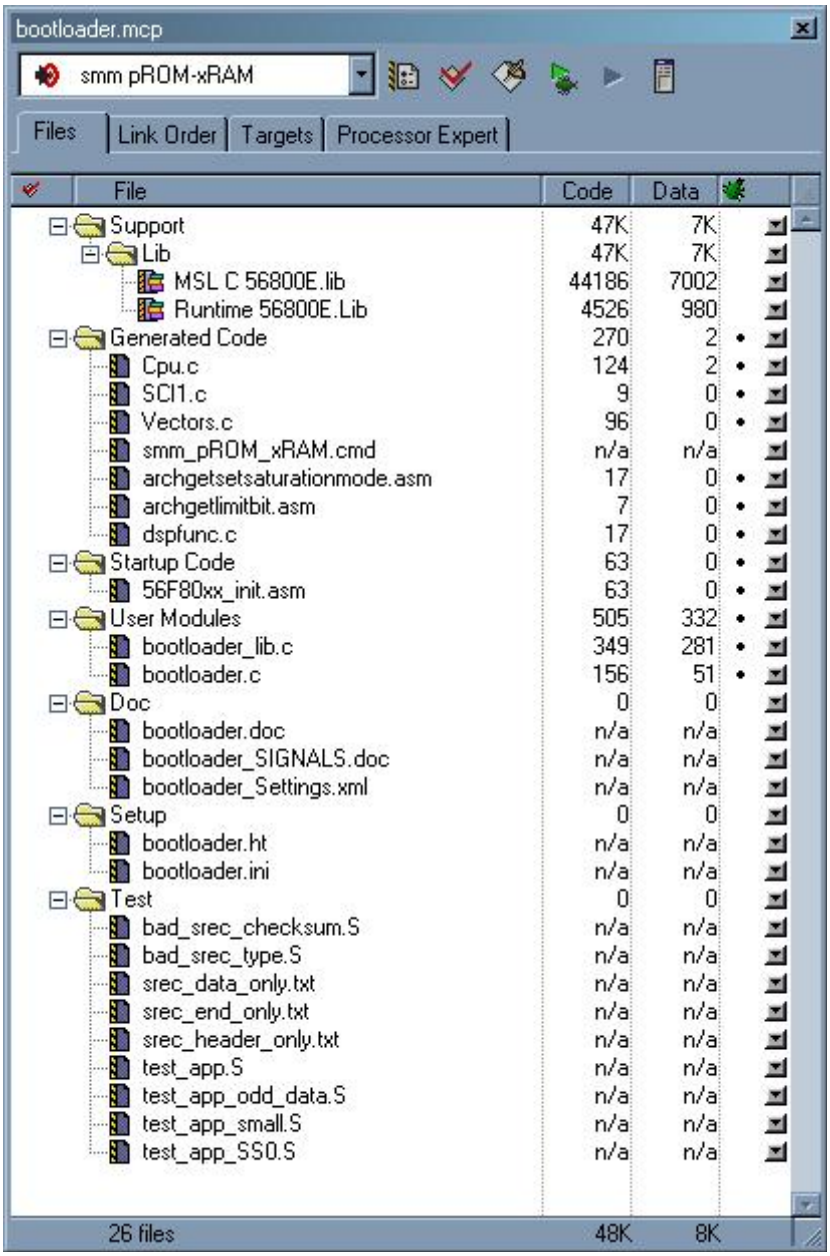

Figure 1-2. Serial Bootloader Directory Structure and Files

Support

This directory contains run time libraries imported when using the Codewarrior tool. The contents of this directory can not be modified.

Generated Code

This directory contains source code generated by the Processor Expert (PE) stationery and beans. Files within this directory may be modified if changes are made to project or bean settings. It should be noted, however, that the linker command file, *smm_pROM_pRAM.cmd*, has been customized for the Serial Bootloader application and, to ensure correct operation of the Bootloader software, should not be modified. When compiling the Bootloader software, do not enable the CPU bean property to build the linker command file, as this will result in overwriting the customized file.

Start-up Code

This directory contains the start-up assembly code, *56F80xx_init.asm*, responsible for initializing the core registers and internal memory. This includes the Operating Mode Register (OMR), Stack Pointer (SP), and Hardware Stack (HWS), as well as the on-chip RAM.

User Modules

All files within this directory are customized files, not generated by Processor Expert. The *bootloader.c* file contains the *main()* function, which controls the Serial Bootloader application flow. The *bootloader_lib.c* file contains various utilities for acheving the functionality required by the Serial Bootloader application.

Doc

This directory contains miscellaneous files generated by the Processor Expert tool, related to project settings and the stationery used.

Set-up

This directory contains set-up files used to configure serial communication applications, such as HyperTerminal and Tera Term Pro.

Test

This directory contains various S-record files used to test the Serial Bootloader application.

The Serial Bootloader application source code is available on the Freescale website (**http://www.freescale.com**) and contains the Serial Bootloader application source code and other associated files. To download the Serial Bootloader package, go to:

Products / Digital Signal Processors and Controllers / 56800/E / 56F8013 / Design Tools / Software / Application Software / 56F801x Bootloader

1.1.4 Bootloader Hardware Set-up

These requirements must be met to use the Bootloader:

- It must be programmed into Program Flash
- The RS-232 socket must be connected by serial cable with the host PC's COM serial port (See **Figure 1-3**)

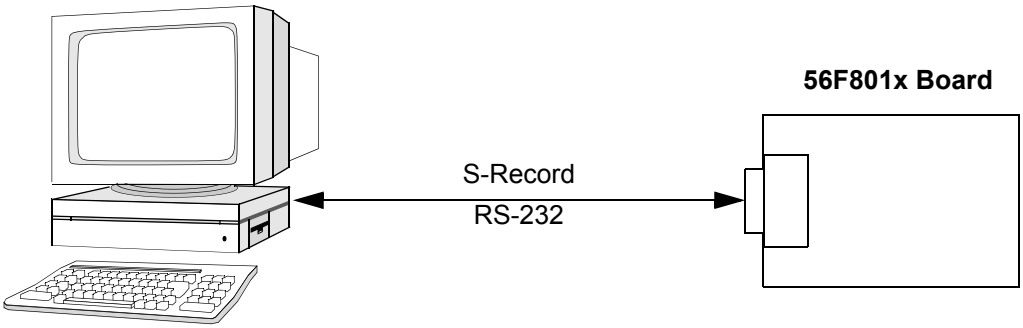

Figure 1-3. Loading a User's Application via Serial Bootloader

1.1.5 Host Terminal Set-up

A host terminal program is used to communicate with the Bootloader. The terminal must be configured to the following mode:

Table 1-2. PC Serial Communication Settings

Baud Rate	115200bps
Data Format	8N1 (8 data bits, no parity, 1 stop bit character format)
Flow Control Protocol	Xon / Xoff

Note: See **Appendix B** for additional configuration options required by Serial Bootloader versions prior to 1.3.

Once your serial port has been properly configured, the serial communications application should be capable of displaying any 56F801x Serial Bootloader messages. An example of how to obtain this configuration using HyperTerminal follows.

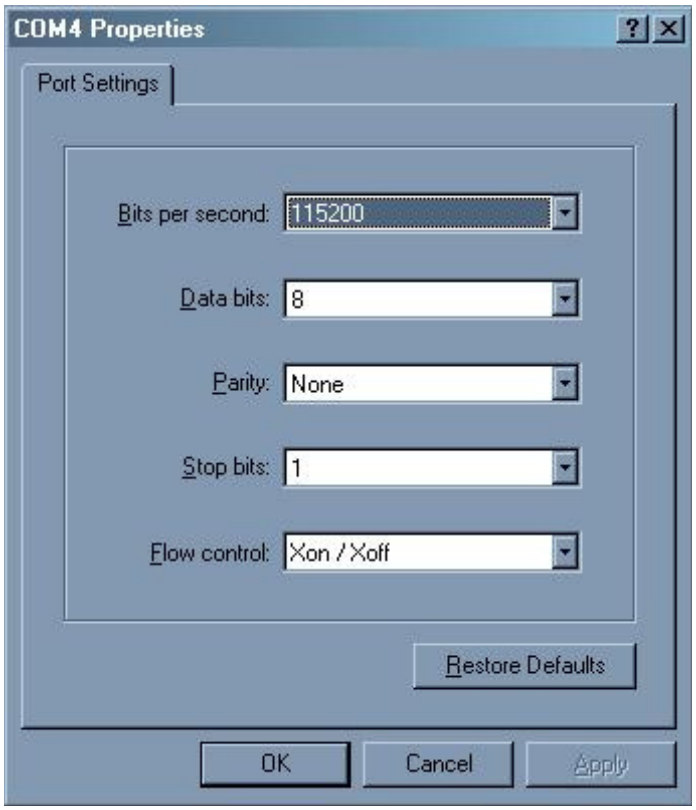

Figure 1-4. HyperTerminal Configuration Example

1.1.6 Build

All 56F801x devices should be preprogrammed with the Serial Bootloader application. However, source code is provided in the event it is necessary to reprogram a 56F801x device with the Serial Bootloader application via CodeWarrior (see **Section 1.1.3**).

To build the Serial Bootloader, first open the *bootloader.mcp* project file using the CodeWarrior IDE. Although this project uses the 56F8013 CPU bean, it is not necessary to change the target processor when building for other 56F801x devices. This single project generates code executable on all 56F801x devices. Next, select the *Project/Make* command, which builds and links the Serial Bootloader application.

1.1.7 Download into Program Flash

Load the software onto the 56F801x device by choosing the *Project/Debug* command in the CodeWarrior IDE. Make sure all hardware has been properly configured before attempting to download the software.

1.1.8 Execute

Once the 56F801x Serial Bootloader has been downloaded, the Bootloader will run automatically after a reset has been asserted on the device. If the terminal program is properly set up and the EVM and the Host PC are properly connected, the terminal program will display the following version number and wait for S-record download:

```
"MC56F801x Serial Bootloader v1.5"
"Waiting for S-record file ..."
```

To load the S-Record file using HyperTerminal, use the *"Transfer/Send Text File"* menu option to select the desired S-Record file. If the download attempt is successful, the following message will be displayed:

```
"Download complete"
```

When an error is detected while downloading the S-Record file, the 56F801x Serial Bootloader displays an error message and waits for the transmission of another S-record file. The following example of this scenario appears:

```
"Error: 3001"
"Waiting for S-record file ..."
```

Note: Consult Troubleshooting, **Section 1.1.13**, for information on decoding error messages.

1.1.9 Requirements for a Loaded Program

If the application is to be loaded via the Bootloader, it must meet the following requirements:

- **Application Entry Point** - The hardware reset vector must be present at the beginning of Program Flash
- **Initialized Data** - All variables must be initialized via program code. Although it is possible to generate S-records that contain data used to initialize RAM, this data will be ignored by the Serial Bootloader application.
- **Restricted Resources** - The S-record used for download should not contain any data targeted for RAM. The Serial Bootloader uses this region of memory to execute code used to erase and program Flash, as well as to download and parse S-record data. If data is needed in RAM, it should be loaded into Program Flash and copied into RAM at run time (this is handled automatically when using the Processor Expert stationery).

1.1.10 S-Record Generation

Figure 1-5 illustrates how a user's application can be set up within Code Warrior to generate an S-Record needed by the Bootloader to program the user's application into the 56F801x device's Flash. The options for S-Record generation are circled in blue.

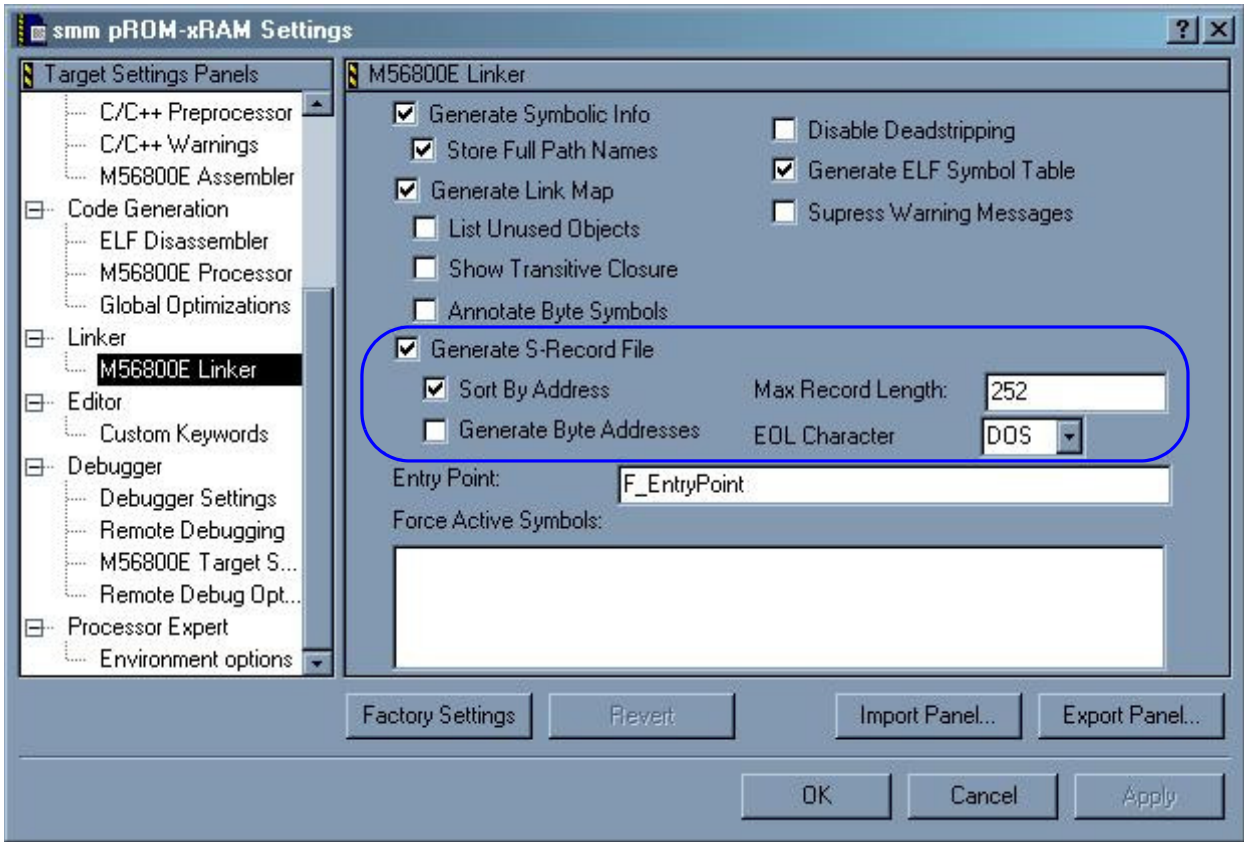

Figure 1-5. S-Record Settings in the User Application's Project Settings

The "Max Record Length" option gives the maximum number of bytes that can be used to comprise the address and data fields for a given S-Record. To reduce S-Record parsing overhead, this option should be set with a large number. Although the maximum value for this option is "255", CodeWarrior will use "252" by default.

The "EOL Character" option determines what type of line ending will be used for the S-Record file. To ensure proper parsing of the S-Record file, this option must be set to "DOS".

1.1.11 Peripheral Usage

The 56F801x Serial Bootloader uses the Flash Module (FM), Interrupt Controller (ITCN), On-Chip Clock Synthesis (OCCS), Phase-Locked Loop (PLL), and Serial Communications Interface (SCI) peripherals. The PLL clock is enabled to allow a 32MHz system clock, and assumes the Internal Relaxation Oscillator input frequency is 8MHz. The FM Clock Divide value used by the application also assumes an 8MHz Internal Relaxation Oscillator input. The SCI Baud Rate value used was calculated to achieve 115,200 baud communication, assuming a

32MHz system clock. After downloading a new application with the Serial Bootloader, a hardware reset is required to run the application. The hardware reset will restore all peripheral registers to default values before running the new application.

1.1.12 Performance Data

The Serial Bootloader application consumes 838 words (approximately 1.64 kB) of Program Flash. The Serial Bootloader is capable of programming the entire flash array in under 5 seconds in most cases. Of course, the actual performance will depend on the PC, Operating System, and serial application used. Below are some sample download times for programming the 56F8013 device using HyperTerminal and Tera Term Pro.

Table 1-3. Sample Download Times for 16KB Memory

Serial Application	PC Type	Windows 2000	Windows XP
HyperTerminal	Pentium-4 1.7 GHz	4.46 seconds	4.75 seconds
Tera Term Pro	Pentium-3 1.0 GHz	3.75 seconds	4.29 seconds

1.1.13 Troubleshooting

The Serial Bootloader application checks for the following error conditions:

- SCI Receiver Error
- Flash Programming Error (Access Error, Protection Violation)
- Flash Verify Error
- Invalid S-Record Checksum
- Invalid S-Record Type

If any of these errors are encountered, the Serial Bootloader application will generate an error message and prompt the user to download the S-Record file again. The following text illustrates what is seen on the PC host terminal in the event of a download error:

```
"MC56F801x Serial Bootloader v1.5"
"Waiting for S-record file ..."
"PROGRAM&DATA***"
"Error: 3002"
"Waiting for S-record file ..."
```

In the progress message, *PROGRAM&DATA****, each asterisk indicates a succesfully processed S-record. This information can can be used with the status code, *Error: 3002*, to determine whether or not an S-record file contains bad data. The status code is a 16-bit hexadecimal

number, containing a variety of status and error flags controlled by the Serial Bootloader application; see **Figure 1-6**.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RX EOF	RX HEADER	TX XOFF						ERR SCI_RX		ERR FM_PROG	ERR FM_PVIOL	ERR FM_ACCERR	ERR TYPE	ERR CKSUM

Figure 1-6. Serial Bootloader Status Code

A description of each bit used in the Serial Bootloader status code is provided in **Table 1-4**. Please consult this table for possible errors and resolutions when troubleshooting the Serial Bootloader application.

Table 1-4. Bootloader Troubleshooting Guide

Status Code	Description	Possible Cause(s)	Solution(s)
ERR_CKSUM	The checksum calculated by the Bootloader does not match the S-record checksum field	S-record file corrupted	Regenerate the S-record by rebuilding the application
		Noisy serial cable	Ensure the serial cable is properly connected and/or replace it with a new cable
ERR_TYPE	An invalid S-record type was received. Only S0, S3 and S7 records are supported by the Serial Bootloader application.	S-record file corrupted	Regenerate the S-record by rebuilding the application
		Noisy serial cable	Ensure the serial cable is properly connected and/or replace it with a new cable
ERR FM_ACCERR	An access error was encountered when programming the Flash	On-chip Flash Module failure	Contact Freescale Technical Support
ERR FM_PVIOL	A protection violation was encountered when programming the Flash	On-chip Flash Module failure	Contact Freescale Technical Support
ERR FM_PROG	The Flash program / read verify failed for a word in the Flash array	On-chip Flash Module failure	Contact Freescale Technical Support
ERR SCI_RX	An SCI receive error was detected when downloading the S-record (e.g., noise, frame, and /or receiver overrun	Noisy serial cable	Ensure the serial cable is properly connected and/or replace it with a new cable
		Incorrect PC host terminal serial program baud rate selected	See **Section 1.1.5** for correct configuration
		PC host terminal serial program transmitted break character	Do not send break characters
TX XOFF	APPLICATION STATUS FLAG	The ASCII “XOFF” character (0x13) was sent to the PC terminal to halt transmission	N/A
RX HEADER	APPLICATION STATUS FLAG	The SO record was received and processed	N/A
RX EOF	APPLICATION STATUS FLAG	The S7 record was received and processed	N/A

Unshaded rows = Error Indicators
Shaded rows = Status Indicators

Chapter 2
License

2.1 Limited Use License Agreement

LIMITED USE LICENSE AGREEMENT

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THIS SOFTWARE. BY USING OR COPYING THE SOFTWARE, YOU AGREE TO THE TERMS OF THIS AGREEMENT.

The software in either source code form ("Source") or object code form ("Object") (cumulatively hereinafter "Software") is provided under a license agreement ("Agreement") as described herein. Any use of the Software including copying, modifying, or installing the Software so that it is usable by or accessible by a central processing unit constitutes acceptance of the terms of the Agreement by the person or persons making such use or, if employed, the employer thereof ("Licensee") and if employed, the person(s) making such use hereby warrants that they have the authority of their employer to enter this license agreement. If Licensee does not agree with and accept the terms of this Agreement, Licensee must return or destroy any media containing the Software or materials related thereto, and destroy all copies of the Software.

The Software is licensed to Licensee by Freescale Semiconductor Inc. ("Freescale") for use under the terms of this Agreement. Freescale retains ownership of the Software. Freescale grants only the rights specifically granted in this Agreement and grants no other rights. Title to the Software, all copies thereof and all rights therein, including all rights in any intellectual property including patents, copyrights, and trade secrets applicable thereto, shall remain vested in Freescale.

For the Source, Freescale grants Licensee a personal, non-exclusive, non-assignable, revocable, royalty-free right to use, copy, and make derivatives of the Source solely in a development system environment in order to produce object code solely for operating on a Freescale semiconductor device having a central processing unit ("Derivative Object").

For the Object and Derivative Object, Freescale grants Licensee a personal, non-exclusive, non-assignable, revocable, royalty-free right to copy, use, and distribute the Object and the Derivative Object solely for operating on a Freescale semiconductor device having a central processing unit.

Licensee agrees to: (a) not use, modify, or copy the Software except as expressly provided herein, (b) not distribute, disclose, transfer, sell, assign, rent, lease, or otherwise make available the

Software, any derivatives thereof, or this license to a third party except as expressly provided herein, (c) not remove obliterate, or otherwise defeat any copyright, trademark, patent or proprietary notices, related to the Software (d) not in any form export, re-export, resell, ship or divert or cause to be exported, re-exported, resold, shipped, or diverted, directly or indirectly, the Software or a direct product thereof to any country which the United States government or any agency thereof at the time of export or re-export requires an export license or other government approval without first obtaining such license or approval.

THE SOFTWARE IS PROVIDED ON AN "AS IS" BASIS AND WITHOUT WARRANTY OF ANY KIND INCLUDING (WITHOUT LIMITATION) ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL FREESCALE BE LIABLE FOR ANY LIABILITY OR DAMAGES OF ANY KIND INCLUDING, WITHOUT LIMITATION, DIRECT OR INDIRECT OR INCIDENTAL OR CONSEQUENTIAL OR PUNITIVE DAMAGES OR LOST PROFITS OR LOSS OF USE ARISING FROM USE OF THE SOFTWARE OR THE PRODUCT REGARDLESS OF THE FORM OF ACTION OR THEORY OF LIABILITY (INCLUDING WITHOUT LIMITATION, ACTION IN CONTRACT, NEGLIGENCE, OR PRODUCT LIABILITY) EVEN IF FREESCALE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THIS DISCLAIMER OF WARRANTY EXTENDS TO LICENSEE OR USERS OF PRODUCTS AND IS IN LIEU OF ALL WARRANTIES WHETHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR PARTICULAR PURPOSE.

Freescale does not represent or warrant that the Software is free of infringement of any third party patents, copyrights, trade secrets, or other intellectual property rights or that Freescale has the right to grant the licenses contained herein. Freescale does not represent or warrant that the Software is free of defect, or that it meets any particular requirements or need of the Licensee, or that it conforms to any documentation, or that it meets any standards.

Freescale shall not be responsible to maintain the Software, provide upgrades to the Software, or provide any field service of the Software. Freescale reserves the right to make changes to the Software without further notice to Licensee.

The Software is 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 Software could create a situation where personal injury or death may occur. Should Licensee purchase or use the Software for any such unintended or unauthorized application, Licensee shall indemnify and hold Freescale 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 was negligent regarding the design or manufacture of the Software.

The term of this Agreement is for as long as Licensee uses the Software for its intended purpose and is not in default of any provisions of this Agreement. Freescale may terminate this Agreement if Licensee is in default of any of the terms and conditions of this Agreement.

This Agreement shall be governed by and construed in accordance with the laws of the State of Arizona and can only be modified in a writing signed by both parties. Licensee agrees to jurisdiction and venue in the State of Arizona.

By using, modifying, installing, compiling, or copying the Software, Licensee acknowledges that this Agreement has been read and understood and agrees to be bound by its terms and conditions. Licensee agrees that this Agreement is the complete and exclusive statement of the agreement between Licensee and Freescale and supersedes any earlier proposal or prior arrangement, whether oral or written, and any other communications relative to the subject matter of this Agreement.

Appendix A
Serial Bootloader Test Application

A simple test application is provided for the Serial Bootloader application. This application fills the entire Program Flash array with functional code, along with a sequential data pattern. When executed, the application checks for the expected pattern of data in the Program Flash locations not used for storing code. If the data pattern verification was successful, the application blinks a series of LEDs connected to GPIOA[2:0] at 1Hz. Otherwise, the LEDs will remain off to indicate failure.

Versions of the test application are available in S-record format for each 56F801x device (i.e., *test_app_56F8013*, *test_app_56F8014*) on the Freescale website (**http://www.freescale.com**) as part of the 56F801x Serial Bootloader software package. The test application source code is also available on the website. To obtain any desired files, go to:

Products / Digital Signal Processors and Controllers / 56800/E / 56F8013 / Design Tools / Software / Application Software

Appendix B
Serial Bootloader Revision History

This document was targeted for the most recent version of the 56F801x Serial Bootloader. You should reference this section if you are using an older version of the Bootloader software, where some of the differences in functionality of the older revisions will be covered in detail.

- **Bootloader Messages**

While newer versions of the Bootloader (1.3 and higher) show an error code containing the types of errors encountered, previous versions do not send such a descriptive message. The following block of text is an example of what is seen on a terminal program when an error occurs on a device running Bootloader versions 1.2 and lower:

```
"MC56F801x Serial Bootloader v1.1"
"Waiting for S-record file ..."
"PROGRAM&DATA***"
"Download Aborted!"
"Waiting for S-record file ..."
```

- **Host Terminal Set-up**

Bootloader versions 1.2 and lower require additional terminal configuration options; additional options are highlighted in bold in **Table B-1** .

Table B-1. PC Serial Communication Settings for Versions 1.2 and Lower

Baud Rate	115200bps
Data Format	8NI (8 data bits, no parity, 1 stop bit character format)
Flow Control Protocol	Xon / Xoff
Line Delay	2ms
Line Ending	“\r” (carriage return), followed by “\n” (newline)

Figure B-1 shows how to configure HyperTerminal for use with 56F801x Serial Bootloader Versions 1.2 and lower.

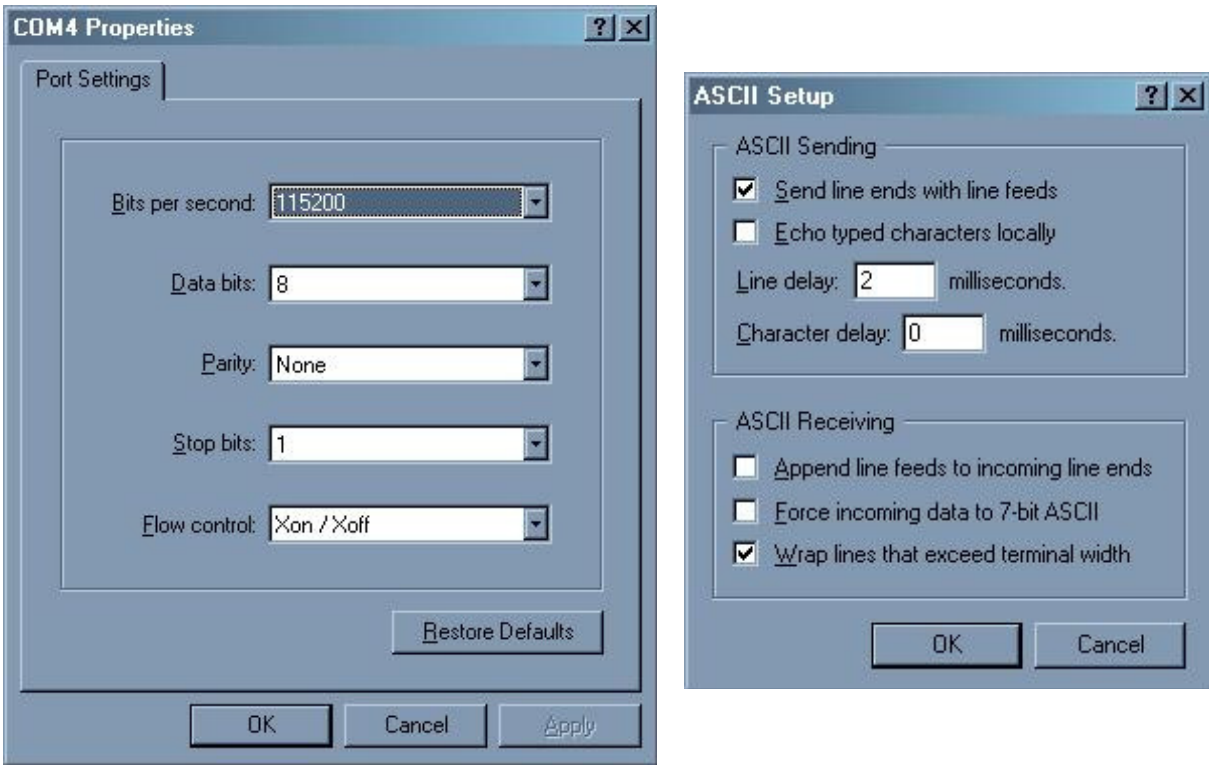

Figure B-1. HyperTerminal Configuration Example for Versions 1.2 and Lower

- **S-Record Formatting**

Bootloader versions 1.3 and lower are unable to fully process data in S3 records if the S-record data field contains an odd number of bytes. These versions of software assumed that all S3 records would contain an even number of actual data bytes. This assumption was based on the fact the 56800/E is a 16-bit architecture with 16-bit program opcodes.

A PERL script is available to modify S-record files containing odd S3 data. After using the script to convert an S-record file, it should be safe to use on all versions of the Bootloader software. Instructions for use are embedded within the script itself, and can be referenced by executing the script with an "*-h*" or "*-help*" option in the command line. Please reference the Freescale FAQ website, **http://faqts.freescale.net**, to obtain this script; see FAQ #25759.

It is recommended, however, that users take advantage of the 1.5 release. Version 1.5 of the Bootloader handles the odd S3 data field issue in the Bootloader application. It accomplishes this by appending 0xFF to the S3 data if the record contains an odd number of data field bytes. This fix does not affect the S-record checksum calculation. To obtain version 1.5, go to:

Products / Digital Signal Processors and Controllers / 56800/E / 56F8013 / Design Tools / Software / Application Software / 56F801x Bootloader

INDEX

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

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb-free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to http://www.freescale.com/epp.

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.

56F801xBLUG
Rev. 0
12/2005