Application Note

AN2571/D
Rev. 1, 9/2003

Converting Demos in Standard Software Drivers for HC908 Derivatives

By Chen He
Embedded Memory Center
Technology & Manufacturing, SPS
Austin, Texas

Introduction

The HC908 SGF NVM standard software driver (SSD) provides a set of standard driver functions for embedded FLASH and EEPROM on HC908 MCUs using 0.5 micron split-gate FLASH (SGF) non-volatile memory (NVM) technology.

The following file formats are provided for the SSD function set:

- Assembly source codes — Users can directly include this file format in their applications.
- S-record — This file format can be used as the only target resident code in a monitor-mode programming tool.

In the release of the standard software SGF driver for HC908 v3.0, demos are provided for each driver file format as below:

- ASM demo — Demonstrates how to use the SSD functions in embedded applications written in assembly language. The demos are developed and ready to use in CodeWarrior® development studio for HC08 v2.1.1.
- C demo — Demonstrates how to use the SSD functions in embedded applications written in C language where the SSD functions written in assembly language are called as standard C functions. The demos are developed and ready to use in CodeWarrior development studio for HC08 v2.1.1.
- S-record demo — Demonstrates how to use the S-record format SSD functions with a monitor mode controller. The demo is developed and ready to use with HiWave™ debugger v6.1.

is a registered trademark and HiWave™ is a trademark of Freescale Semiconductor, Inc.
This product incorporates SuperFlash® technology licensed from SST.

The driver functions are designed to run on any member of the HC08 Family with embedded 0.5 micron SGF NVM. However, the demo codes are part-dependent due to possible different memory mapping among HC908 derivatives. The demos included in the release package v3.0 are ready-to-run for HC908 Families of AZ60A, KX8, GR8, and SR12.

This application note describes the needed modifications of the above demo codes for HC908 derivatives other than AZ60A, KX8, GR8, and SR12.

NOTE: *With the exception of mask set errata documents, if any other Motorola document contains information that conflicts with the information in the device data sheet, the data sheet should be considered to have the most current and correct data.*

Summary of Potential Modifications to Demos

ASM Demo and C Demo

Some macros in the demo source files (*.asm for ASM demos and *.c for C demos) and sections in the link parameter file (*.prm) must be updated according to the memory map of a specific HC908 MCU.

ASM and C Demo for EEPROM

Modifications may be required to the following places for EEPROM demos for HC908 derivatives with embedded EEPROM using 0.5 micron SGF technology.

Table 1. Macros in Source File

Macro	File	Description	Remarks
EEDIVREGH	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Address of EEPROM clock divider register high	See data sheet for correct value
EEDIVREGL	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Address of EEPROM clock divider register low	See data sheet for correct value
EECR	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Address of EEPROM control register	See data sheet for correct value
initStack	ASM demo\AZ60A\EEPROM\source\main.asm	Stack top address used as initial value of SP	Depends on user application’s memory map

Table 1. Macros in Source File (Continued)

Macro	File	Description	Remarks
ERASESTRT	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Start address of EEPROM region to be erased	Equal to the start address of EEPROM in the demos
BLKEND	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	End address of EEPROM to be block erased	Equal to ERASESTRT plus EEPROM block size minus one
BYTEEND	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	End address of EEPROM to be byte erased	Equal to ERASESTRT
BULKEND	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	End address of EEPROM to be bulk erased	Equal to ERASESTRT plus EEPROM bulk size minus one
PRGSTRT	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Start address of EEPROM region to be programmed	Equal to the start address of EEPROM in the demos
PRGEND	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	End address of EEPROM region to be programmed	Equal to PRGSTRT plus PRGSIZE minus one
PRGSIZE	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Size of bytes to be programmed	Limited by RAM size for source buffer
SRCBUF	ASM demo\AZ60A\EEPROM\source\main.asm; C demo\AZ60A\EEPROM\source\main.c	Source data buffer start address	Resides in the direct page of RAM for demos

Table 2. Link Parameter File

Section	File	Description	Remark
PARA_RAM	ASM demo\AZ60A\EEPROM\prms\AZ60A-RAM.PRM; C demo\AZ60A\EEPROM\prms\mon08-ram.prm	Section storing global parameters	Must reside in the direct page of RAM. The size is at least 16 bytes.
Z_RAM	ASM demo\AZ60A\EEPROM\prms\AZ60A-RAM.PRM; C demo\AZ60A\EEPROM\prms\mon08-ram.prm	Section storing source data buffer	Resides in the direct page of RAM for demos
MY_RAM	ASM demo\AZ60A\EEPROM\prms\AZ60A-RAM.PRM; C demo\AZ60A\EEPROM\prms\mon08-ram.prm	Default RAM section	Resides in RAM space
MY_PSEUDO_ROM1	ASM demo\AZ60A\EEPROM\prms\AZ60A-RAM.PRM; C demo\AZ60A\EEPROM\prms\mon08-ram.prm	Code section 1	These two sections contain demo codes and driver codes. They will be in RAM space for FLASH operations.
MY_PSEUDO_ROM2	ASM demo\AZ60A\EEPROM\prms\AZ60A-RAM.PRM; C demo\AZ60A\EEPROM\prms\mon08-ram.prm	Code section 2	

ASM and C Demo for FLASH

Modifications may be required to the following places for FLASH demos for HC908 derivatives with embedded FLASH memory using 0.5 micron SGF technology.

Table 3. Macros in Source Files

Macro	File[(1)]	Description	Remark

FLCR	ASMdemo\{MCU}\Flash\source\main.asm;
	ASMdemo\{MCU}\Flash\source\main_masserase.asm;
	C demo\{MCU}\Flash\source\main.c;
	C demo\{MCU}\Flash\source\mainmasserase.c
		Address of FLASH control register	See data sheet for correct value

FLBPR	ASMdemo\{MCU}\Flash\source\main.asm;
	ASMdemo\{MCU}\Flash\source\main_masserase.asm;
	C demo\{MCU}\Flash\source\main.c;
	C demo\{MCU}\Flash\source\mainmasserase.c
		Address of FLASH protection register	See data sheet for correct value

initStack	ASMdemo\{MCU}\Flash\source\main.asm; ASMdemo\{MCU}\Flash\source\main_masserase.asm	Stack top address used as initial value of SP	Depends on user application's memory map

FESTRT	ASMdemo\{MCU}\Flash\source\main.asm;
	ASMdemo\{MCU}\Flash\source\main_masserase.asm;
	C demo\{MCU}\Flash\source\main.c;
	C demo\{MCU}\Flash\source\mainmasserase.c
		Start address of FLASH region to be erased	Equal to start address of FLASH in demos

Table 3. Macros in Source Files (Continued)

Macro	File[1]	Description	Remark

FEEND	ASMdemo\{MCU}\Flash\source\main.asm;
	ASMdemo\{MCU}\Flash\source\main_masserase.asm;
	C demo\{MCU}\Flash\source\main.c;
	C demo\{MCU}\Flash\source\mainmasserase.c
		End address of FLASH region to be erased	Equal to FESTRT plus erase size minus one

PRGSTRT	ASMdemo\{MCU}\Flash\source\main.asm; C demo\{MCU}\Flash\source\main.c	Start address of FLASH region to be programmed	Equal to the start address of FLASH in demos
PRGEND	ASMdemo\{MCU}\Flash\source\main.asm; C demo\{MCU}\Flash\source\main.c	End address of FLASH region to be programmed	Equal to PRGSTRT plus program size minus one
SSD_BASE	ASMdemo\{MCU}\Flash\source\main.asm; C demo\{MCU}\Flash\source\main.c	Base address in RAM of driver codes copied from FLASH	Do not overlap the source buffer in RAM
SRCBUF	ASMdemo\{MCU}\Flash\source\main.asm; C demo\{MCU}\Flash\source\main.c	Source data buffer start address in RAM	Will be greater than SSD_BASE plus the size of FlashProgram (0x48) to avoid overlapping
BUFSIZE	ASMdemo\{MCU}\Flash\source\main.asm	Source buffer size in bytes	Limited by RAM size
ROWSIZE	C demo\{MCU}\Flash\source\main.c	FLASH row size in bytes	Can be 64 bytes or 32 bytes depending on the HC908 MCU

[1] {MCU} can be AZ60A, KX8, GR8, or SR12.

Table 4. Linking Parameter File

Section	File[1]	Description	Remark
Flash_FECOP	ASMdemo\{MCU}\Flash\prms\{MCU}-Flash.PRM	Section reserved for FlashEraseCOP	Can be located in anywhere in FLASH
Flash_FP	ASMdemo\{MCU}\Flash\prms\{MCU}-Flash.PRM; C demo\{MCU}\Flash\prms\mon08-flash.prm	Section reserved for FlashProgram	Can be located in anywhere in FLASH
Flash_FE	ASMdemo\{MCU}\Flash\prms\{MCU}-Flash.PRM; C demo\{MCU}\Flash\prms\mon08-flash.prm	Section reserved for FlashErase	Can be located in anywhere in FLASH
Flash_MISC	ASMdemo\{MCU}\Flash\prms\{MCU}-Flash.PRM; C demo\{MCU}\Flash\prms\mon08-flash.prm	Section reserved for other driver codes	Can be located in anywhere in FLASH

PARA_RAM	ASMdemo\{MCU}\Flash\prms\{MCU}-Flash.PRM;
	ASMdemo\{MCU}\Flash\prms\{MCU}-RAM.PRM;
	C demo\{MCU}\Flash\prms\mon08-flash.prm;
	C demo\{MCU}\Flash\prms\mon08-ram.prm
		Section storing global parameters	Must reside in the direct page of RAM. The size is at least 16 bytes.

RAM	ASMdemo\{MCU}\Flash\prms\{MCU}-Flash.PRM;
	ASMdemo\{MCU}\Flash\prms\{MCU}-RAM.PRM;
	C demo\{MCU}\Flash\prms\mon08-flash.prm;
	C demo\{MCU}\Flash\prms\mon08-ram.prm
		Default RAM section	In RAM space

ROM/ MY_PSEUDO_ROM/ MY_PSEUDO_ROM1	ASMdemo\{MCU}\Flash\prms\{MCU}-RAM.PRM; C demo\{MCU}\Flash\prms\mon08-ram.prm	Code sections	Contain demo codes and driver codes. They will be in RAM space for FLASH operations.

[1] {MCU} can be AZ60A, KX8, GR8, or SR12.

S-Record Demo

Some macros in the initialization script file init.scp must be updated according to the memory map of a specific HC908 MCU.

S-Record Demo for EEPROM

Modifications may be required to the following macros in init.scp for EEPROM demos for HC908 derivatives with embedded EEPROM using 0.5 micron SGF technology.

Table 5. Macros in Initialization Script File

Macro[1]	File	Description	Remark
EEnDIVREGH	S-record demo\AZ60A\EEPROM\init.scp	Address of EEPROM-[n] clock divider register high	See data sheet for correct value
EEnDIVREGL	S-record demo\AZ60A\EEPROM\init.scp	Address of EEPROM-[n] clock divider register low	See data sheet for correct value
EEnCONTROLREG	S-record demo\AZ60A\EEPROM\init.scp	Address of EEPROM-[n] control register	See data sheet for correct value
EEPROMn_BASE	S-record demo\AZ60A\EEPROM\init.scp	Start address of EEPROM-[n] region to be erased;	Typically equal to the start address of EEPROMn
RAM_BASE	S-record demo\AZ60A\EEPROM\init.scp	Base address of RAM	Depends on chip configuration
RAM_SIZE	S-record demo\AZ60A\EEPROM\init.scp	Size of RAM	Depends on chip configuration
DRIVER_DATA_ BASE	S-record demo\AZ60A\EEPROM\init.scp	Base address of global parameters	Must reside in the direct page of RAM
SSD_BASE	S-record demo\AZ60A\EEPROM\init.scp	Base address in RAM of driver functions	Do not overlap the source buffer in RAM
BUFFER_BASE	S-record demo\AZ60A\EEPROM\init.scp	Base address of source data buffer	Will be greater than SSD_BASE plus the size of FlashProgram (0x48) to avoid overlapping
BUFFER_SIZE	S-record demo\AZ60A\EEPROM\init.scp	Source data buffer size	Limited by RAM size
STACK_BASE	S-record demo\AZ60A\EEPROM\init.scp	Address of stack bottom	Stack must reside in RAM space
STACK_SIZE	S-record demo\AZ60A\EEPROM\init.scp	Size of stack in bytes	
Addr_StackTop	S-record demo\AZ60A\EEPROM\init.scp	Address of stack top used as initial value of SP	

[1] [n] is 1 for EEPROM1 or 2 for EEPROM2.

S-Record Demo for FLASH

Modifications may be required to the following places for FLASH demos for HC908 derivatives with embedded FLASH memory using 0.5 micron SGF technology.

Table 6. Macros in Initialization Script File

Macro[1]	File	Description	Remark
FLCONTROLREG[n]	S-record demo\{MCU}\Flash\init.scp	Address of FLASH[n] control register	See data sheet for correct value
FLPROTECTREG[n]	S-record demo\{MCU}\Flash\init.scp	Address of FLASH[n] protection register	See data sheet for correct value
Flash[n]_BASE	S-record demo\{MCU}\Flash\init.scp	Start address of FLASH[n] region for demos	In FLASH[n] space
Flash[n]_SIZE	S-record demo\{MCU}\Flash\init.scp	Size of FLASH[n] in bytes for demos	In FLASH[n] space
RAM_BASE	S-record demo\{MCU}\Flash\init.scp	Base address of RAM	Depends on chip configuration
RAM_SIZE	S-record demo\{MCU}\Flash\init.scp	Size of RAM in bytes	Depends on chip configuration
DRIVER_DATA_BASE	S-record demo\{MCU}\Flash\init.scp	Base address of global parameters	Must reside in the direct page of RAM
SSD_BASE	S-record demo\{MCU}\Flash\init.scp	Base address in RAM of driver functions	Do not overlap the source buffer in RAM
BUFFER_BASE	S-record demo\{MCU}\Flash\init.scp	Base address of source data buffer	Will be greater than SSD_BASE plus the size of FlashProgram (0x48) to avoid overlapping
BUFFER_SIZE	S-record demo\{MCU}\Flash\init.scp	Source data buffer size	Limited by RAM size
STACK_BASE	S-record demo\{MCU}\Flash\init.scp	Address of stack bottom	Stack must reside in RAM space
STACK_SIZE	S-record demo\{MCU}\Flash\init.scp	Size of stack in bytes	
Addr_StackTop	S-record demo\{MCU}\Flash\init.scp	Address of stack top used as initial value of SP	

[1] For AZ60A, [n] is 1 for FLASH1 or 2 for FLASH2. For MCUs with only one FLASH module (such as KX8, GR8, and SR12), [n] is omitted.

Examples of Converting Demos for LJ12, AS32A, and QY4

LJ12 (Converting from Demos for SR12)

Compared to SR12, LJ12 has the same memory map of resources such as RAM and FLASH. Therefore, the demos for SR12 can be used as is for LJ12.

AS32A (Converting from Demos for AZ60A)

AS32A is most similar to AZ60A. Compared to AZ60A, AS32A is slightly different in resource sizes and memory map. For instance, AS32A has only one block of RAM (0x0050–0x044F). The modifications required are listed below (only the new values for AS32A are shown here).

- ASM demo for EEPROM
  - – Changes in Demos\ASM Demo\AZ60A\EEPROM\source\main.asm:
    ```
    initStack   EQU $450 ; initial stack top for AS32A
    ```
  - – Changes in Demos\ASMDemo\AZ60A\EEPROM\prms\AZ60A-RAM.PRM:
    ```
    MY_PSEUDO_ROM1 = READ_ONLY   0x0200 to 0x03FF;
    MY_PSEUDO_ROM2 = READ_ONLY   0x0400 to 0x044F;
    ```
- ASM demo for FLASH — (No changes required)
- C demo for EEPROM
  - – Changes in Demos\C Demo\AZ60A\EEPROM\prms\mon08_ram.prm:
    ```
    MY_PSEUDO_ROM1 = READ_ONLY   0x0200 to 0x03FF;
    MY_PSEUDO_ROM2 = READ_ONLY   0x0400 to 0x044F;
    ```
- C demo for FLASH — (No changes required)
- S-record demo for EEPROM — (No changes required)
- S-record demo for FLASH — (No changes required)

QY4 (Converting from Demos for SR12)

Compared to AZ60A, KX8, GR8, and SR12, QY4 has very different resource sizes and memory map. Therefore, the demos for QY4 require many changes to the existing demos, e.g. demos for SR12. The modifications required are listed as below (only the new values for QY4 are shown here).

- ASM demo for FLASH
  - – Changes in Demos\ASM Demo\SR12\Flash\source\main.asm:
    ```
    PRGSTRT     EQU $EE00  ; program start address
    PRGEND      EQU $EE1F  ; program end address
    FESTRT      EQU $EE00  ; erase start address
    FEEND       EQU $EE3F  ; erase end address
    SRCBUF      EQU $00D4  ; source buffer start address
    BUFSIZE     EQU $20    ; source buffer size
    FLCR        EQU $FE08  ; QY4 flash control register address
    FLBPR       EQU $FFBE  ; QY4 flash block protection register address

    FP_START    EQU $F700  ; location of FlashProgram in flash
    ```

```
FP_SIZE      EQU $48    ; FlashProgram size
FE_START     EQU $F800  ; location of FlashErase in flash
FE_SIZE      EQU $46    ; FlashErase size
FECOP_START  EQU $F600  ; location of FlashEraseCOP in flash
FECOP_SIZE   EQU $59    ; FlashEraseCOP size

SSD_BASE:    EQU $8C    ; SSD driver base address in RAM
```

- Changes in Demos\ASM Demo\SR12\Flash\source\main_masserase.asm:

  ```
  ESTRT       EQU $EE00  ; erase start address
  FLCR        EQU $FE08  ; QY4 flash control register address
  FLBPR       EQU $FFBE  ; QY4 flash block protection register address
  ```

- The code for BlankCheck in main_masserase.asm must be removed due to RAM size limit in QY4.

- Changes in Demos\ASM Demo\SR12\Flash\prms\sr12-Flash.PRM:

  ```
  Flash_FECOP  = READ_ONLY 0xF600 to 0xF658;
  Flash_FP     = READ_ONLY 0xF700 to 0xF747;
  Flash_FE     = READ_ONLY 0xF800 to 0xF845;
  Flash_MISC   = READ_ONLY 0xF900 to 0xFDFF;
  RAM  = READ_WRITE 0x1000 to 0x1FFF; /* dummy area */
  PARA_RAM  = READ_WRITE  0x80 to 0x8B;
  ```

- Changes in file Demos\ASM Demo\SR12\Flash\prms\sr12-RAM.PRM:

  ```
  ENTRIES
  FlashErase // BlankCheck shall be removed because of RAM size limit
  END
  ROM  = READ_ONLY  0x8C to 0xF1;
  PARA_RAM  = READ_WRITE  0x80 to 0x8B;
  ```

- C demo for FLASH
  - Changes in Demos\C Demo\SR12\Flash\source\main.c:

    ```
    #define PRGSTRT      0xEE00 /* program start address */
    #define PRGEND       0xEE1F /* program end address */
    #define FESTRT       0xEE00 /* erase start address */
    #define FEEND        0xEE3F /* erase end address */
    #define SRCBUF       0x00D4 /* source buffer start address */
    #define FLCR         0xFE08 /* QY4 flash control register address */
    #define FLBPR        0xFFBE /* QY4 flash block protection register
    address */

    #define FP_START   0xF700 /* location of FlashProgram in flash */
    #define FP_SIZE    0x48   /* FlashProgram size */
    #define FE_START   0xF800 /* location of FlashErase in flash */
    #define FE_SIZE    0x46   /* FlashErase size */

    #define SSD_BASE     0x8C   /* SSD driver base address in RAM */
    ```

  - Changes in Demos\C Demo\SR12\Flash\source\mainmasserase.c:

    ```
    #define  FESTRT    0xEE00 /* erase start address */
    #define  FEEND     0xFDFF /* erase end address */
    #define  FLCR      0xFE08 /* QY4 flash control register address */
    ```

```
#define  FLBPR    0xFFBE /* QY4 flash block protection register
address*/
```

- The code for BlankCheck in mainmasserase.c need to be removed due to RAM size limit in QY4.
- Changes in Demos\C Demo\SR12\Flash\prms\mon08_flash.prm:

```
Flash_FP     = READ_ONLY 0xF700 to 0xF747;
Flash_FE     = READ_ONLY 0xF800 to 0xF845;
Flash_MISC   = READ_ONLY 0xF900 to 0xFDFF;
PARA_RAM  = READ_WRITE  0x80 to 0x8B;
```

- Changes in Demos\C Demo\SR12\Flash\prms\mon08_ram.prm:

```
ROM  = READ_ONLY  0x8C to 0xF1;
PARA_RAM  = READ_WRITE  0x80 to 0x8B;
```

- S-record demo for FLASH
  - Changes in Demos\S-record Demo\SR12\Flash\init.scp:

```
DEFINE DRIVER_DATA_BASE     0x0080
DEFINE SSD_BASE             0x008C
DEFINE BUFFER_BASE          0x00D4
DEFINE BUFFER_SIZE          0x0020
DEFINE STACK_BASE           0x00F4
DEFINE STACK_SIZE           0x000C
DEFINE FLPROTECTREG         0xFFBE
DEFINE Flash_BASE           0xEE00
DEFINE Flash_SIZE           0x1000
DEFINE PAGE_SIZE            0x0040
```

Constraints and General Considerations

The standard software SGF driver for HC908 is designed to work for HC908 MCUs using 0.5 micron SGF NVM technology. It does NOT support HC908 parts using different NVM technologies. For instance, MC68HC908AS60 uses a different NVM technology and thus is not supported by the driver.

For QY/QT Family, the current mass erase demo cannot fit into 128 bytes RAM. The BlankCheck function call may be removed to reduce the demo size.

The monitor ROM code will use several bytes in the RAM while debugging with CodeWarrior tools. Therefore, addresses from 0x00F3 to 0x00F8 are not recommended to use for the demos.

Users should be careful with the system clock settings. If the system clock frequency is changed, users must update the macro for the system clock in the driver accordingly and then rebuild the driver.

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.

freescale™
semiconductor

AN2571/D
Rev. 1, 9/2003