Freescale Semiconductor
Application Note

Document Number: AN3260
Rev. 1, 05/2007

Interfacing HCS12 Microcontrollers to the MFR4310 FlexRay Controller

by: David Paterson
MCD Applications, East Kilbride

1 Introduction

Freescale Semiconductor offers a range of standalone FlexRay communication controllers (MFR4xxx) that can be interfaced with 16-bit and 32-bit microcontrollers (MCU). This application note describes the hardware, software and timing considerations necessary for reliable communication between the MFR4310 controller and the HCS12 family of MCUs.

2 Objective

The aim of the document is to demonstrate the simplicity of the hardware interface between the MFR4310 and the HCS12 and to provide an example of the software used to configure the HCS12 for operation. The information contained can help you quickly design a fully functional FlexRay node based on the HCS12 Family of MCUs. Evaluation boards with software are available from Freescale, to assist in the development of FlexRay applications. (See http://www.freescale.com/flexray.)

Contents

Note: In this document, active-low signals are indicated by a "#" at the end of the signal name, e.g. "IRQn#".

© Freescale Semiconductor, Inc., 2006, 2007. All rights reserved.

3 Hardware Design

The MFR4310 FlexRay controller has two separate controller host interfaces (CHI) on board - an HCS12 interface, for direct connection to Freescale's HCS12 family of microcontrollers, and an asynchronous memory interface (AMI) for asynchronous connection to all other microcontrollers. The HCS12 interface clock signal, used to synchronize the data transfer, can run at a maximum rate of 8 MHz.

Chip selection for the S12 interface is generated internally using the following signals:

- The input values of the expanded address signals XADDR[14:19] are compared with logical 0's (the S12 External Bus Interface (EBI) is in the Paged or Unpaged mode).
- The three most significant bits of the demultiplexed address bus, PA[5:7], are compared with the pattern set up externally on the address chip select pins ACS[0:2]; PA5 is compared with ACS0, PA6 with ACS1, PA7 with ACS2.

NOTE

The address decoding phase of a read/write operation is passed if all the comparisons described above are passed.

The devices can be connected together without additional glue logic, thereby simplifying the design and reducing the system cost.

Section 3.2 shows how to connect the FlexRay CC to an S12 MCU with EBI paged mode and unpaged mode support.

3.1 Selecting the HCS12 Mode

There are two modes of operation for the controller host interface (CHI), the AMI mode and the HCS12 mode. The HCS12 mode is used to interface to the HCS12 Family of microcontrollers.

To select the HCS12 mode, IF_SEL1 must be at the logic high level and IF_SEL0 must be at the logic low level. When using 5V VDDIO, for example, IF_SEL1 must be pulled high using a 10 kΩ pullup resistor, and IF_SEL0 must be pulled low using a 47 kΩ pulldown resistor. (Similarly, at 3.3 V VDDIO, INT_CC# must be held high by a 16 kΩ pullup resistor.)

See the MFR4310 data sheet (reference 2) for recommended pullup and pulldown resistor values for the IF_SEL[1:0] inputs.

3.2 Paged Mode and Unpaged Mode

There are two modes of operation that can be used when interfacing the HCS12 and MFR4310 microcontrollers — paged and unpaged mode.

Figure 1 and Figure 2 show block diagrams with all the EBI connections that are necessary for each mode of operation.

Table 1 provides a description of each signal and pin that must be connected to interface the MFR4310 to a typical HCS12 MCU, the MC9S12DP256B in a 112-pin LQFP package.

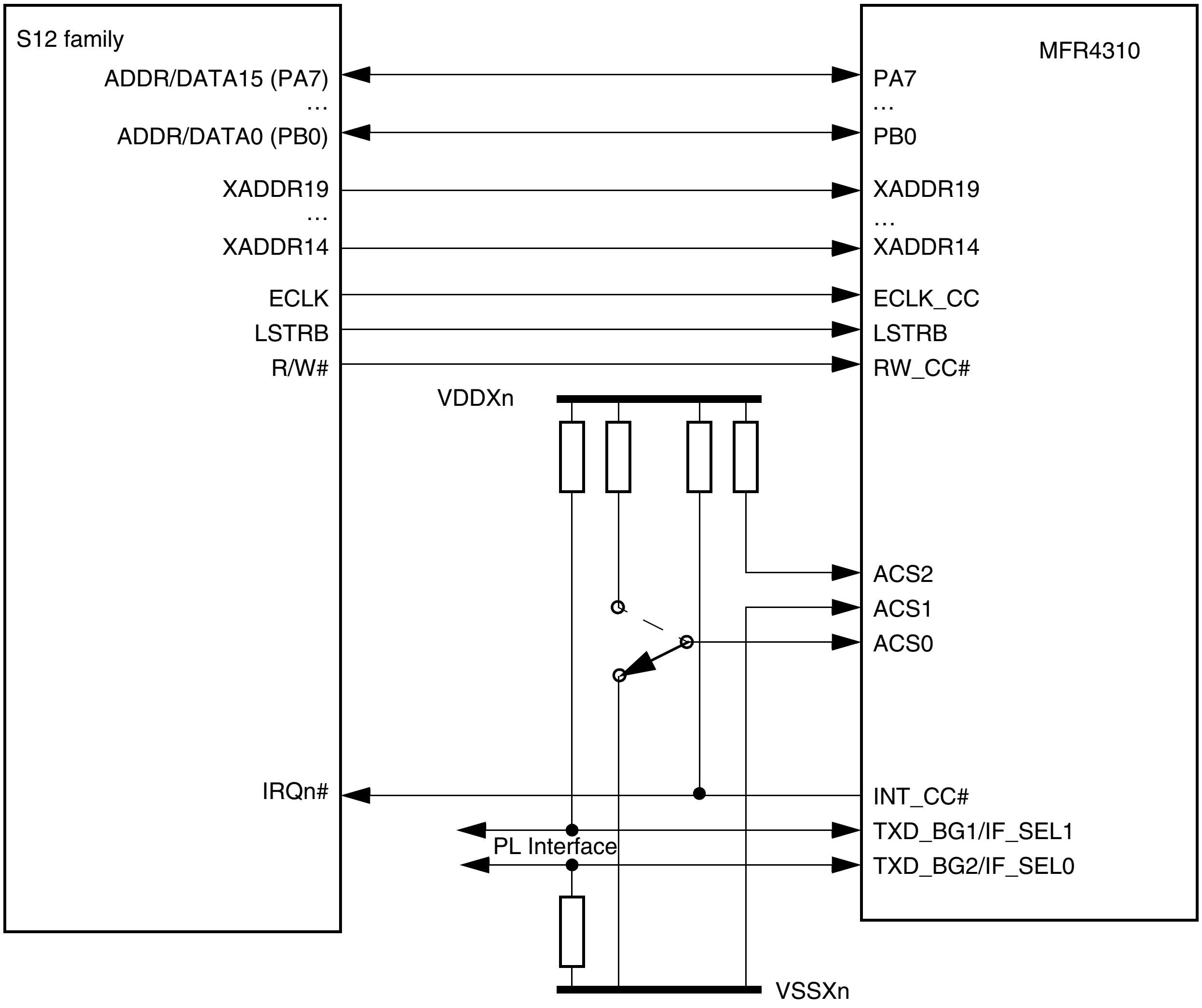

Figure 1. Connecting MFR4310 to HCS12 with Page Mode Support

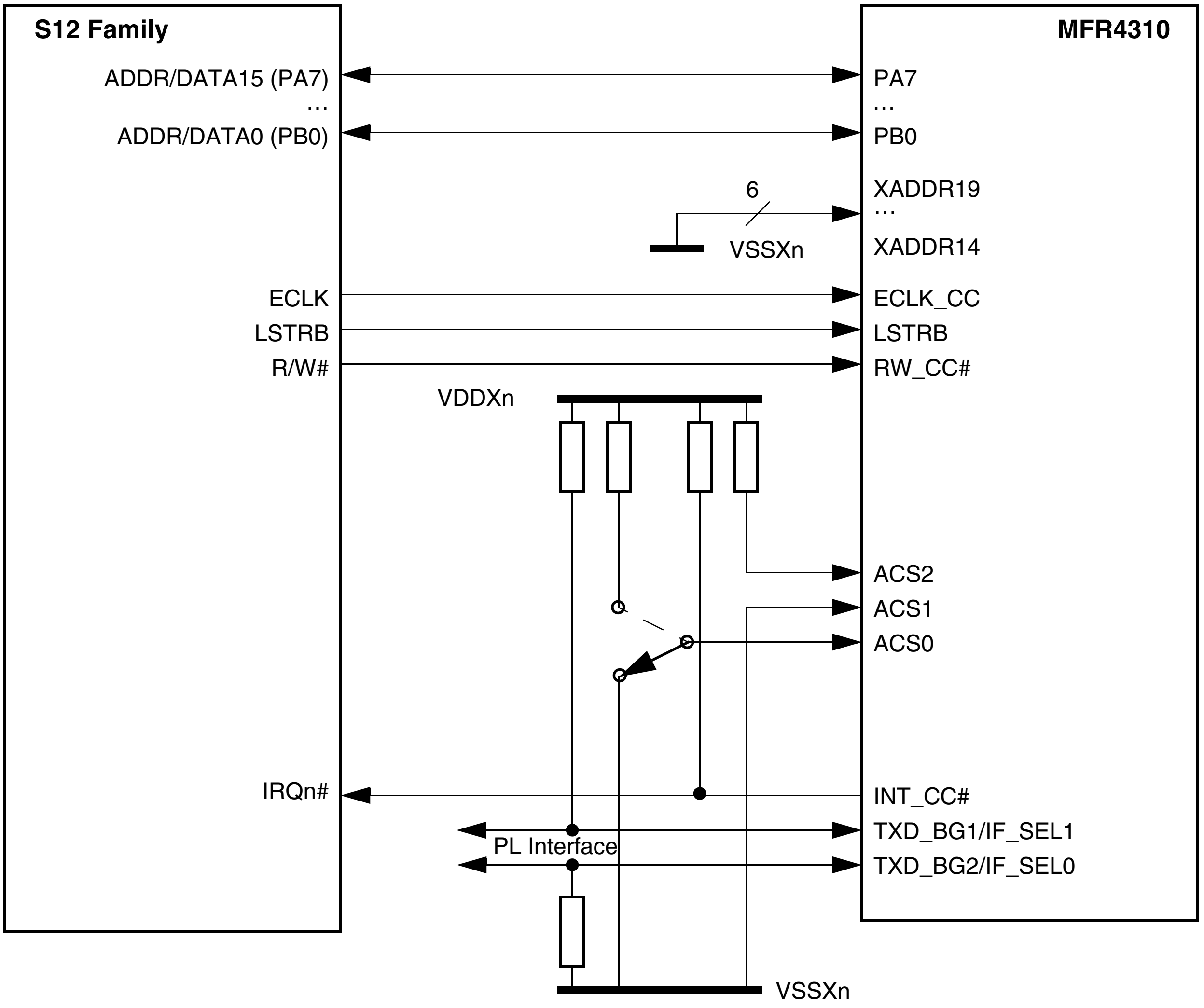

Figure 2. Connecting MFR4310 to HCS12 with Unpaged Mode Support

Table 1. Interface Signal Description

HCS12 to MFR4310 Connection				
MC9S12DP256B (112-Pin LQFP)		MFR300		Comments
Signal	Pin	Signal	Pin	
ADDR0/DATA0/PB0	24	D15/PB0	10	HCS12 data/address bus (LSB)
ADDR1/DATA1/PB1	25	D14/PB1	7	HCS12 data/address bus
ADDR2/DATA2/PB2	26	D13/PB2	6	HCS12 data/address bus
ADDR3/DATA3/PB3	27	D12/PB3	5	HCS12 data/address bus
ADDR4/DATA4/PB4	28	D11/PB4	4	HCS12 data/address bus
ADDR5/DATA5/PB5	29	D10/PB5	3	HCS12 data/address bus
ADDR6/DATA6/PB6	30	D9/PB6	2	HCS12 data/address bus
ADDR7/DATA7/PB7	31	D8/PB7	62	HCS12 data/address bus
PA0/ADDR8/DATA8	57	D7/PA0	61	HCS12 data/address bus
PA1/ADDR9/DATA9	58	D6/PA1	58	HCS12 data/address bus
PA2/ADDR10/DATA10	59	D5/PA2	57	HCS12 data/address bus
PA3/ADDR11/DATA11	60	D4/PA3	56	HCS12 data/address bus
PA4/ADDR12/DATA12	61	D3/PA4	55	HCS12 data/address bus
PA5/ADDR13/DATA13	62	D2/PA5	51	HCS12 data/address bus
PA6/ADDR14/DATA14	63	D1/PA6	40	HCS12 data/address bus
PA7/ADDR15/DATA15	64	D0/PA7	39	HCS12 data/address bus
XADDR14/PK0	8	A6/XADDR14	17	Extended addresses (page mode)
XADDR15/PK1	7	A5/XADDR15	15	Extended addresses (page mode)
XADDR16/PK2	6	A4/XADDR16	14	Extended addresses (page mode)
XADDR17/PK3	5	A3/XADDR17	13	Extended addresses (page mode)
XADDR18/PK4	20	A2/XADDR18	12	Extended addresses (page mode)
XADDR19/PK5	19	A1/XADDR19	11	Extended addresses (page mode)
ECLK/PE4	39	A10/ECLK_CC	52	HCS12 clock input
R/W#/PE2	54	WE#/RW_CC#	30	HCS12 read/write select signal
LSTRB/TAGL0/PE3	53	CE#/LSTRB	29	HCS12 low-byte strobe signal
GPIO (or PU/PD)		OE#/ASC0, A11/ASC1, A12/ACS2	27, 28, 34	HCS12 address select inputs
GPIO		RESET#	16	Hardware reset input
PE1/IRQ#	55	INT_CC#	64	Controller interrupt output
		BGT/DBG2/IF_SEL0	32	Host interface selection = LOW (HCS12 mode)
		TXD_BG1/TXD1_485/IF_SEL1	41	Host interface selection = HIGH (HCS12 mode)

4 Timing Considerations

For the HCS12 and the MFR4310 communication controller to communicate reliably, the timing between the HCS12 and the MFR4310 must be matched. There is a requirement for the addition of stretch cycles[1] to match the timing characteristics. Refer to the MFR4310 data sheet (reference 2) for specific timing diagrams. A minimum of one stretch cycle must be inserted when running with an 8 MHz external clock. Table 3 summarizes the read and write timing for one stretch cycle, which is required for successful operation. The number of stretch cycles that can be added is controlled in the Miscellaneous System Control Register (MISC). Refer to the Module Mapping Control (MMC) section of the user manual. The available options are shown in Table 2.

Table 2. Stretch Cycle Selection

External Access Stretch Bit Definition	
EXSTR[1:0]	**Number of E Clocks Stretched**
00	0 cycles
01	1 cycle
10	2 cycles
11	3 cycles

Table 3. HCS12 to MFR4310 Timing Parameters with One Stretch Cycle

Rating	Symbol	Specified		Measured	Comment
		Min	Max		
Pulse width, ECLK Low	tLEC	30		560	PASS
Pulse width, ECLK High	tHEC	99		190	PASS
Address valid time to ECLK rise	tSA	11		56	PASS
Write data delay time	tDDW		70	5	PASS
Write data hold time	tHDW	80		190	PASS
RW delay time	tDRW		7	0	PASS
RW valid time to ECLK rise	tSRW	14		560	PASS
RW hold time	tHRW	2		8	PASS
Data hold to address	tHDA	2		8	PASS
Multiplexed address hold time	tHA	2		12	PASS
ECLK high access time (ECLK high to Read Data valid)	tDEC	50	90	56	PASS
Read data setup time	tSDR	13		135	PASS
Read data hold time	tHDR	0		12	PASS

[1] When accessing on-chip peripherals and memories, the HCS12 performs 8-bit and 16-bit core accesses in a single cycle. However, when the core accesses locations on the external bus using the expanded modes, the accesses are stretched and take more than a single cycle to complete. The minimum amount of stretching is one additional bus cycle, but can be increased. See Table 2 for details.

The S12 interface can run only with the 40 MHz EXTAL_CC clock. Therefore, the timing on the S12 bus is as described in the MFR4310 reference manual. Here, the minimum low is 30 ns and the minimum high 99 ns. That would give us a maximum ECLK frequency of 7.75 MHz (on the MFR4200 the maximum ECLK frequency is 25 ns + 114ns = 7.19 MHz).

The S12 interface accepts only aligned 16-bit read and 8-bit or 16-bit write transactions. The S12 interface does not support 8-bit read accesses.

- The internal chip select, CS, the low byte strobe, LSTRB, the least significant bit of the address, ADDR[0], and the read/write select, RW#, are used to determine the type of access.
- RW_CC# indicates the direction of data transfer for a transaction.

Table 4. HCS12 Access Types

CS	RW	LSTRB	ADR[0]	Type of Access
0	X	X	X	No access
1	0	0	0	16-bit write to word address[1]
1	0	0	1	8-bit write to an odd address[2]
1	0	1	0	8-bit write to an even address[2]
1	0	1	1	Not supported
1	1	0	0	16-bit read from an even address[3]
1	1	0	1	Not supported
1	1	1	0	Not supported
1	1	1	1	Not supported

[1] Write data from PA to even byte address and from PB to odd byte address.

[2] Write data from PB.

[3] Read data from even byte address at PA and from odd byte address at PB.

5 Software

The software setup is concerned mainly with selection of the correct MCU operating mode, configuration of the HCS12 for communication with the MFR4310, and linking into the correct area of the memory map.

5.1 Mode Configuration and EBI Operation

As already stated, the HCS12 must be configured into normal expanded mode to enable the bus interface. In this mode, ports A and B are configured for the data/address bus, and port E is configured for the control signals. During initialization, the EBI is configured, and the mode is set to normal expanded wide mode by writing MODC = 1, MODB = 1, and MODA = 1 in the MODE register. The EBI must be initialized to use the correct chip select and to select the correct number of stretch cycles. See Section 4, "Timing Considerations" for specific details.

Figure 3 shows the steps required to configure the HCS12 MCU for communication with the MFR4310 controller.

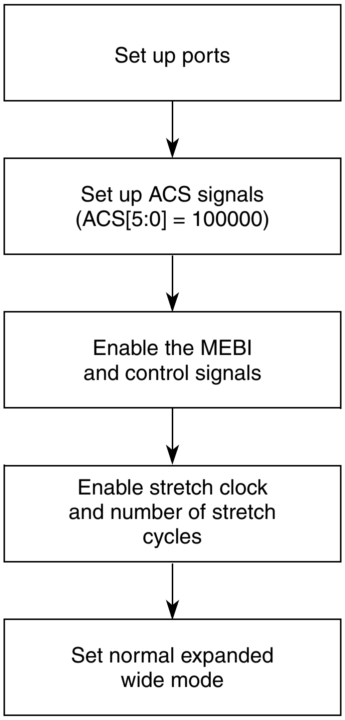
Set up ports
Set up ACS signals (ACS[5:0] = 100000)
Enable the MEBI and control signals
Enable stretch clock and number of stretch cycles
Set normal expanded wide mode

Figure 3. Initialization Flow Diagram

An example of typical HCS12 MCU initialization code is shown below.

```
// Set up ports
PORTB = 0xFF;             // Port B = 0xFF
DDRB = 0xFF;              // Set port B as output

PTS = 0x00;               // Port S = 0x00
DDRS = 0x04;              // Set port PS2 as output

                          // Enable PULL UPS on PTE
PUCR_PUPEE = 1;           // Pull ups enabled on bits 7, 4-0

PTH = 0x04;               // ASC[2, 1, 0] = 100match 0x8000 on ADR bus

DDRH = 0x07;              // make PortH an Output

PPAGE = PAGE0;            // Pages $00 - $2F = External
                          // Pages $30 - $3F = Internal

                          // Enable EBI
PEAR = 0x0C;              // NOACCE = 0; PTE7 (GPIO)
                          // PIPOE = 0;  PTE[6:5](GPIO)
                          // NECLK = 0;  PTE4 (External ECLK)
                          // LSTRE = 1;  PTE3 (LSRTB#)
                          // RDWE = 1;   PTE2 (R/W#)

EBICTL = 0x01;            // ESTR = 1; Stretch Clock enabled

INTCR = 0xC0;             // IRQE = 0; low level
                          // IRQEN = 1; Interrupt enabled

MISC = 0x05;              // ESTR1:0 = 01; 1 clock stretches enabled
                          // ROMHM = 0; Flash in low 1/2 of map can be accessed
                          // ROMON = 1; Flash enabled in Memory map

MODE = 0xE3;              // MODC:MODB:MODA = 111; Normal Expanded Wide
                          // IVIS = 0; No visibility of internal bus signals
                          // EMK = 1; PORTK & DDRK removed from memory map
                          // EME = 1; PORTE & DDRE removed from memory map
```

NOTE

This is example code only; the port setup is dependent on specific schematic connections.

5.2 Placement of MFR4310 in Memory Map

Access to the external space is usually done by accessing locations in the page window (0x8000–0xC000) with PPAGE set to point to a page in external space.

In CodeWarrior, the FAR qualifier is used to tell the compiler to use the global instructions included in the HCS12 CPU. An example of how to place the MFR4310 registers into the memory map is shown below.

Example:

```
#define  BASE_ADDRESS 0x8000

#define  MVR      (*(volatile unsigned int* far)(0x000 + BASE_ADDRESS))

#define  MCR      (*(volatile unsigned int* far)(0x002 + BASE_ADDRESS))
```

In this option, the placement of the memory map can be changed by modifying the BASE_ADDRESS. The selected BASE_ADDRESS of 8000 is in the external address range.[1]

6 Conclusions

The FlexRay controller can be connected to the HCS12 Family of MCUs, with a maximum interface clock frequency (to synchronize data transfer) of 8 MHz.

Due to this specific interface between the HCS12 microcontrollers and MFR4310 FlexRay controller, no glue logic required. Software configuration is also straightforward, the peripheral being simply memory-mapped into the address space.

7 References

1. MC9S12DP256B Data Sheet (MC9S12DP256B)
2. MFR4310 FlexRay Communication Controller Reference Manual (MFR4310RM)

These documents are available on the Freescale Semiconductor web site at http://www.freescale.com.

More information on FlexRay and FlexRay products can be found at http://www.freescale.com/flexray

[1] Many alternative methods of defining the MFR4310 register exist.

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

Document Number: AN3260
Rev. 1
05/2007

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. The described product is a PowerPC microprocessor. The PowerPC name is a trademark of IBM Corp. and is used under license.

© Freescale Semiconductor, Inc. 2006, 2007. All rights reserved.

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics as 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.

freescale™
semiconductor