

Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	35
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	52-LQFP
Supplier Device Package	52-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s12gc16mpbe

There are two basic types of operating modes:

1. **Normal** modes: Some registers and bits are protected against accidental changes.
2. **Special** modes: Allow greater access to protected control registers and bits for special purposes such as testing.

A system development and debug feature, background debug mode (BDM), is available in all modes. In special single-chip mode, BDM is active immediately after reset.

Some aspects of Port E are not mode dependent. Bit 1 of Port E is a general purpose input or the $\overline{\text{IRQ}}$ interrupt input. $\overline{\text{IRQ}}$ can be enabled by bits in the CPU's condition codes register but it is inhibited at reset so this pin is initially configured as a simple input with a pull-up. Bit 0 of Port E is a general purpose input or the $\overline{\text{XIRQ}}$ interrupt input. $\overline{\text{XIRQ}}$ can be enabled by bits in the CPU's condition codes register but it is inhibited at reset so this pin is initially configured as a simple input with a pull-up. The ESTR bit in the EBICtrl register is set to one by reset in any user mode. This assures that the reset vector can be fetched even if it is located in an external slow memory device. The PE6/MODB/IPIPE1 and PE5/MODA/IPIPE0 pins act as high-impedance mode select inputs during reset.

The following paragraphs discuss the default bus setup and describe which aspects of the bus can be changed after reset on a per mode basis.

4.4.3.1 Normal Operating Modes

These modes provide three operating configurations. Background debug is available in all three modes, but must first be enabled for some operations by means of a BDM background command, then activated.

4.4.3.1.1 Normal Single-Chip Mode

There is no external expansion bus in this mode. All pins of Ports A, B and E are configured as general purpose I/O pins. Port E bits 1 and 0 are available as general purpose input only pins with internal pull resistors enabled. All other pins of Port E are bidirectional I/O pins that are initially configured as high-impedance inputs with internal pull resistors enabled. Ports A and B are configured as high-impedance inputs with their internal pull resistors disabled.

The pins associated with Port E bits 6, 5, 3, and 2 cannot be configured for their alternate functions IPIPE1, IPIPE0, $\overline{\text{LSTRB}}$, and $\text{R}/\overline{\text{W}}$ while the MCU is in single chip modes. In single chip modes, the associated control bits PIPEOE, LSTRE, and RDWE are reset to zero. Writing the opposite state into them in single chip mode does not change the operation of the associated Port E pins.

In normal single chip mode, the MODE register is writable one time. This allows a user program to change the bus mode to narrow or wide expanded mode and/or turn on visibility of internal accesses.

Port E, bit 4 can be configured for a free-running E clock output by clearing NECLK=0. Typically the only use for an E clock output while the MCU is in single chip modes would be to get a constant speed clock for use in the external application system.

5.6.3 Interrupt Priority Decoder

The priority decoder evaluates all interrupts pending and determines their validity and priority. When the CPU requests an interrupt vector, the decoder will provide the vector for the highest priority interrupt request. Because the vector is not supplied until the CPU requests it, it is possible that a higher priority interrupt request could override the original exception that caused the CPU to request the vector. In this case, the CPU will receive the highest priority vector and the system will process this exception instead of the original request.

NOTE

Care must be taken to ensure that all exception requests remain active until the system begins execution of the applicable service routine; otherwise, the exception request may not be processed.

If for any reason the interrupt source is unknown (e.g., an interrupt request becomes inactive after the interrupt has been recognized but prior to the vector request), the vector address will default to that of the last valid interrupt that existed during the particular interrupt sequence. If the CPU requests an interrupt vector when there has never been a pending interrupt request, the INT will provide the software interrupt (SWI) vector address.

5.7 Exception Priority

The priority (from highest to lowest) and address of all exception vectors issued by the INT upon request by the CPU is shown in [Table 5-5](#).

Table 5-5. Exception Vector Map and Priority

Vector Address	Source
0xFFFFE–0xFFFF	System reset
0xFFFFC–0xFFFFD	Crystal monitor reset
0xFFFFA–0xFFFFB	COP reset
0xFFFF8–0xFFFF9	Unimplemented opcode trap
0xFFFF6–0xFFFF7	Software interrupt instruction (SWI) or BDM vector request
0xFFFF4–0xFFFF5	XIRQ signal
0xFFFF2–0xFFFF3	IRQ signal
0xFFFF0–0xFF00	Device-specific I-bit maskable interrupt sources (priority in descending order)

6.3.2.1 BDM Status Register (BDMSTS)

0xFF01

	7	6	5	4	3	2	1	0
R	ENBDM	BDMACT	ENTAG	SDV	TRACE	CLKSW	UNSEC	0
W								

Reset:

Special single-chip mode:	1 ⁽¹⁾	1	0	0	0	0	0 ⁽²⁾	0
Special peripheral mode:	0	1	0	0	0	0	0	0
All other modes:	0	1	0	0	0	0	0	0
	0	0	0	0	0	0	0	0

= Unimplemented or Reserved
 = Implemented (do not alter)

Figure 6-3. BDM Status Register (BDMSTS)

Note:

- ENBDM is read as "1" by a debugging environment in Special single-chip mode when the device is not secured or secured but fully erased (Flash and EEPROM). This is because the ENBDM bit is set by the standard firmware before a BDM command can be fully transmitted and executed.
- UNSEC is read as "1" by a debugging environment in Special single-chip mode when the device is secured and fully erased, else it is "0" and can only be read if not secure (see also bit description).

Read: All modes through BDM operation

Write: All modes but subject to the following:

- BDMACT can only be set by BDM hardware upon entry into BDM. It can only be cleared by the standard BDM firmware lookup table upon exit from BDM active mode.
- CLKSW can only be written via BDM hardware or standard BDM firmware write commands.
- All other bits, while writable via BDM hardware or standard BDM firmware write commands, should only be altered by the BDM hardware or standard firmware lookup table as part of BDM command execution.
- ENBDM should only be set via a BDM hardware command if the BDM firmware commands are needed. (This does not apply in special single-chip mode).

firmware. The standard BDM firmware watches for serial commands and executes them as they are received.

The firmware commands are shown in [Table 6-6](#).

Table 6-6. Firmware Commands

Command ⁽¹⁾	Opcode (hex)	Data	Description
READ_NEXT	62	16-bit data out	Increment X by 2 ($X = X + 2$), then read word X points to.
READ_PC	63	16-bit data out	Read program counter.
READ_D	64	16-bit data out	Read D accumulator.
READ_X	65	16-bit data out	Read X index register.
READ_Y	66	16-bit data out	Read Y index register.
READ_SP	67	16-bit data out	Read stack pointer.
WRITE_NEXT	42	16-bit data in	Increment X by 2 ($X = X + 2$), then write word to location pointed to by X.
WRITE_PC	43	16-bit data in	Write program counter.
WRITE_D	44	16-bit data in	Write D accumulator.
WRITE_X	45	16-bit data in	Write X index register.
WRITE_Y	46	16-bit data in	Write Y index register.
WRITE_SP	47	16-bit data in	Write stack pointer.
GO	08	None	Go to user program. If enabled, ACK will occur when leaving active background mode.
GO_UNTIL ⁽²⁾	0C	None	Go to user program. If enabled, ACK will occur upon returning to active background mode.
TRACE1	10	None	Execute one user instruction then return to active BDM. If enabled, ACK will occur upon returning to active background mode.
TAGGO	18	None	Enable tagging and go to user program. There is no ACK pulse related to this command.

1. If enabled, ACK will occur when data is ready for transmission for all BDM READ commands and will occur after the write is complete for all BDM WRITE commands.
2. Both WAIT (with clocks to the S12 CPU core disabled) and STOP disable the ACK function. The GO_UNTIL command will not get an Acknowledge if one of these two CPU instructions occurs before the "UNTIL" instruction. This can be a problem for any instruction that uses ACK, but GO_UNTIL is a lot more difficult for the development tool to time-out.

6.4.5 BDM Command Structure

Hardware and firmware BDM commands start with an 8-bit opcode followed by a 16-bit address and/or a 16-bit data word depending on the command. All the read commands return 16 bits of data despite the byte or word implication in the command name.

NOTE

8-bit reads return 16-bits of data, of which, only one byte will contain valid data. If reading an even address, the valid data will appear in the MSB. If reading an odd address, the valid data will appear in the LSB.

7.3.2.9 Debug Comparator A Extended Register (DBGCAx)

Module Base + 0x002A
Starting address location affected by INITRG register setting.

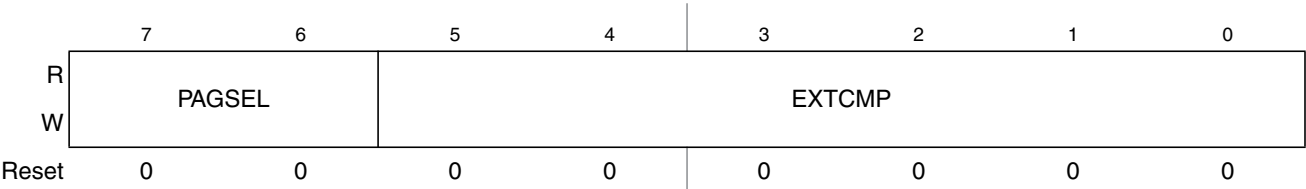


Figure 7-15. Debug Comparator A Extended Register (DBGCAx)

Table 7-19. DBGCAx Field Descriptions

Field	Description
7:6 PAGSEL	Page Selector Field — If DBGGEN is set in DBGCR1, then PAGSEL selects the type of paging as shown in Table 7-20 . DPAGE and EPAGE are not yet implemented so the value in bit 7 will be ignored (i.e., PAGSEL values of 10 and 11 will be interpreted as values of 00 and 01, respectively). In BKP mode, PAGSEL has no meaning and EXTCMP[5:0] are compared to address bits [19:14] if the address is in the FLASH/ROM memory space.
5:0 EXTCMP	Comparator A Extended Compare Bits — The EXTCMP bits are used as comparison address bits as shown in Table 7-20 along with the appropriate PPAGE, DPAGE, or EPAGE signal from the core.

Table 7-20. Comparator A or B Compares

Mode		EXTCMP Compare	High-Byte Compare
BKP ⁽¹⁾	Not FLASH/ROM access	No compare	DBGCRxH[7:0] = AB[15:8]
	FLASH/ROM access	EXTCMP[5:0] = XAB[19:14]	DBGCRxH[5:0] = AB[13:8]
DBG ⁽²⁾	PAGSEL = 00	No compare	DBGCRxH[7:0] = AB[15:8]
	PAGSEL = 01	EXTCMP[5:0] = XAB[21:16]	DBGCRxH[7:0] = XAB[15:14], AB[13:8]

1. See [Figure 7-16](#).
2. See [Figure 7-10](#) (note that while this figure provides extended comparisons for comparator C, the figure also pertains to comparators A and B in DBG mode only).

Chapter 8

Analog-to-Digital Converter (ATD10B8C)

Block Description

8.1 Introduction

The ATD10B8C is an 8-channel, 10-bit, multiplexed input successive approximation analog-to-digital converter. Refer to device electrical specifications for ATD accuracy.

The block is designed to be upwards compatible with the 68HC11 standard 8-bit A/D converter. In addition, there are new operating modes that are unique to the HC12 design.

8.1.1 Features

- 8/10-bit resolution.
- 7 μ sec, 10-bit single conversion time.
- Sample buffer amplifier.
- Programmable sample time.
- Left/right justified, signed/unsigned result data.
- External trigger control.
- Conversion completion interrupt generation.
- Analog input multiplexer for eight analog input channels.
- Analog/digital input pin multiplexing.
- 1-to-8 conversion sequence lengths.
- Continuous conversion mode.
- Multiple channel scans.

8.1.2 Modes of Operation

8.1.2.1 Conversion Modes

There is software programmable selection between performing **single** or **continuous conversion** on a **single channel** or **multiple channels**.

10.3.2.16 MSCAN Identifier Acceptance Registers (CANIDAR0-7)

On reception, each message is written into the background receive buffer. The CPU is only signalled to read the message if it passes the criteria in the identifier acceptance and identifier mask registers (accepted); otherwise, the message is overwritten by the next message (dropped).

The acceptance registers of the MSCAN are applied on the IDR0–IDR3 registers (see [Section 10.3.3.1, “Identifier Registers \(IDR0–IDR3\)”](#)) of incoming messages in a bit by bit manner (see [Section 10.4.3, “Identifier Acceptance Filter”](#)).

For extended identifiers, all four acceptance and mask registers are applied. For standard identifiers, only the first two (CANIDAR0/1, CANIDMR0/1) are applied.

Module Base + 0x0010 (CANIDAR0)
 0x0011 (CANIDAR1)
 0x0012 (CANIDAR2)
 0x0013 (CANIDAR3)

	7	6	5	4	3	2	1	0
R								
W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R								
W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R								
W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R								
W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

Figure 10-19. MSCAN Identifier Acceptance Registers (First Bank) — CANIDAR0–CANIDAR3

Read: Anytime

Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1)

Table 10-19. CANIDAR0–CANIDAR3 Register Field Descriptions

Field	Description
7:0 AC[7:0]	Acceptance Code Bits — AC[7:0] comprise a user-defined sequence of bits with which the corresponding bits of the related identifier register (IDRn) of the receive message buffer are compared. The result of this comparison is then masked with the corresponding identifier mask register.

- a) the 14 most significant bits of the extended identifier plus the SRR and IDE bits of CAN 2.0B messages or
 - b) the 11 bits of the standard identifier, the RTR and IDE bits of CAN 2.0A/B messages.
- Figure 10-40 shows how the first 32-bit filter bank (CANIDAR0–CANIDA3, CANIDMR0–3CANIDMR) produces filter 0 and 1 hits. Similarly, the second filter bank (CANIDAR4–CANIDAR7, CANIDMR4–CANIDMR7) produces filter 2 and 3 hits.
- Eight identifier acceptance filters, each to be applied to the first 8 bits of the identifier. This mode implements eight independent filters for the first 8 bits of a CAN 2.0A/B compliant standard identifier or a CAN 2.0B compliant extended identifier. Figure 10-41 shows how the first 32-bit filter bank (CANIDAR0–CANIDAR3, CANIDMR0–CANIDMR3) produces filter 0 to 3 hits. Similarly, the second filter bank (CANIDAR4–CANIDAR7, CANIDMR4–CANIDMR7) produces filter 4 to 7 hits.
 - Closed filter. No CAN message is copied into the foreground buffer RxFG, and the RXF flag is never set.

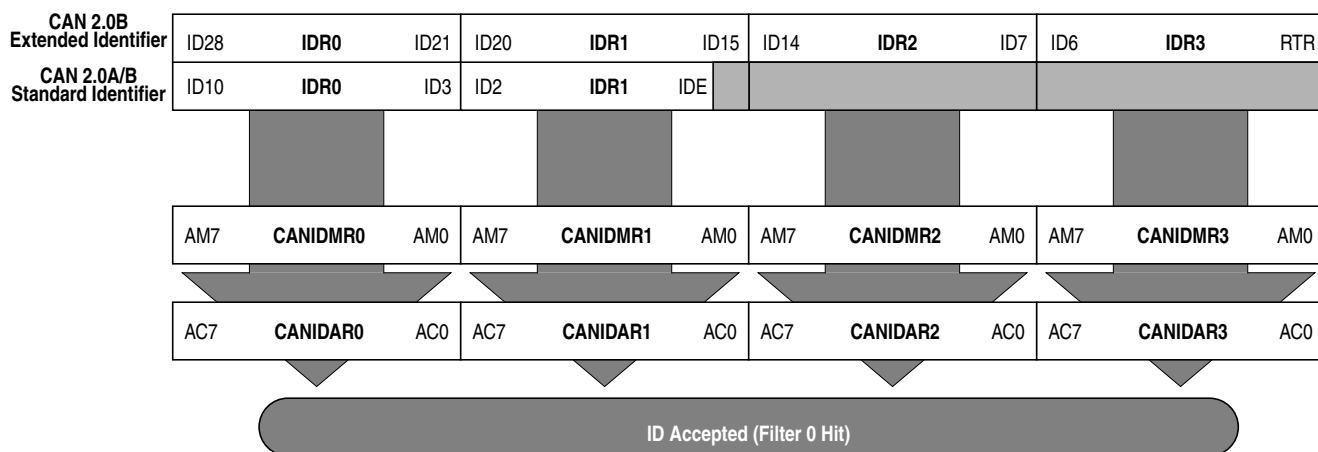


Figure 10-39. 32-bit Maskable Identifier Acceptance Filter

13.3.2.1 SCI Baud Rate Registers (SCIBDH and SCHBDL)

Module Base + 0x_0000

	7	6	5	4	3	2	1	0
R	0	0	0	SBR12	SBR11	SBR10	SBR9	SBR8
W								
Reset	0	0	0	0	0	0	0	0

Module Base + 0x_0001

	7	6	5	4	3	2	1	0
R	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
W								
Reset	0	0	0	0	0	1	0	0

 = Unimplemented or Reserved

Figure 13-3. SCI Baud Rate Registers (SCIBDH and SCIBDL)

The SCI Baud Rate Register is used by the counter to determine the baud rate of the SCI. The formula for calculating the baud rate is:

$$\text{SCI baud rate} = \text{SCI module clock} / (16 \times \text{BR})$$

where:

BR is the content of the SCI baud rate registers, bits SBR12 through SBR0. The baud rate registers can contain a value from 1 to 8191.

Read: Anytime. If only SCIBDH is written to, a read will not return the correct data until SCIBDL is written to as well, following a write to SCIBDH.

Write: Anytime

Table 13-1. SCIBDH AND SCIBDL Field Descriptions

Field	Description
4–0 7–0 SBR[12:0]	<p>SCI Baud Rate Bits — The baud rate for the SCI is determined by these 13 bits.</p> <p>Note: The baud rate generator is disabled until the TE bit or the RE bit is set for the first time after reset. The baud rate generator is disabled when BR = 0.</p> <p>Writing to SCIBDH has no effect without writing to SCIBDL, since writing to SCIBDH puts the data in a temporary location until SCIBDL is written to.</p>

13.4.3 Transmitter

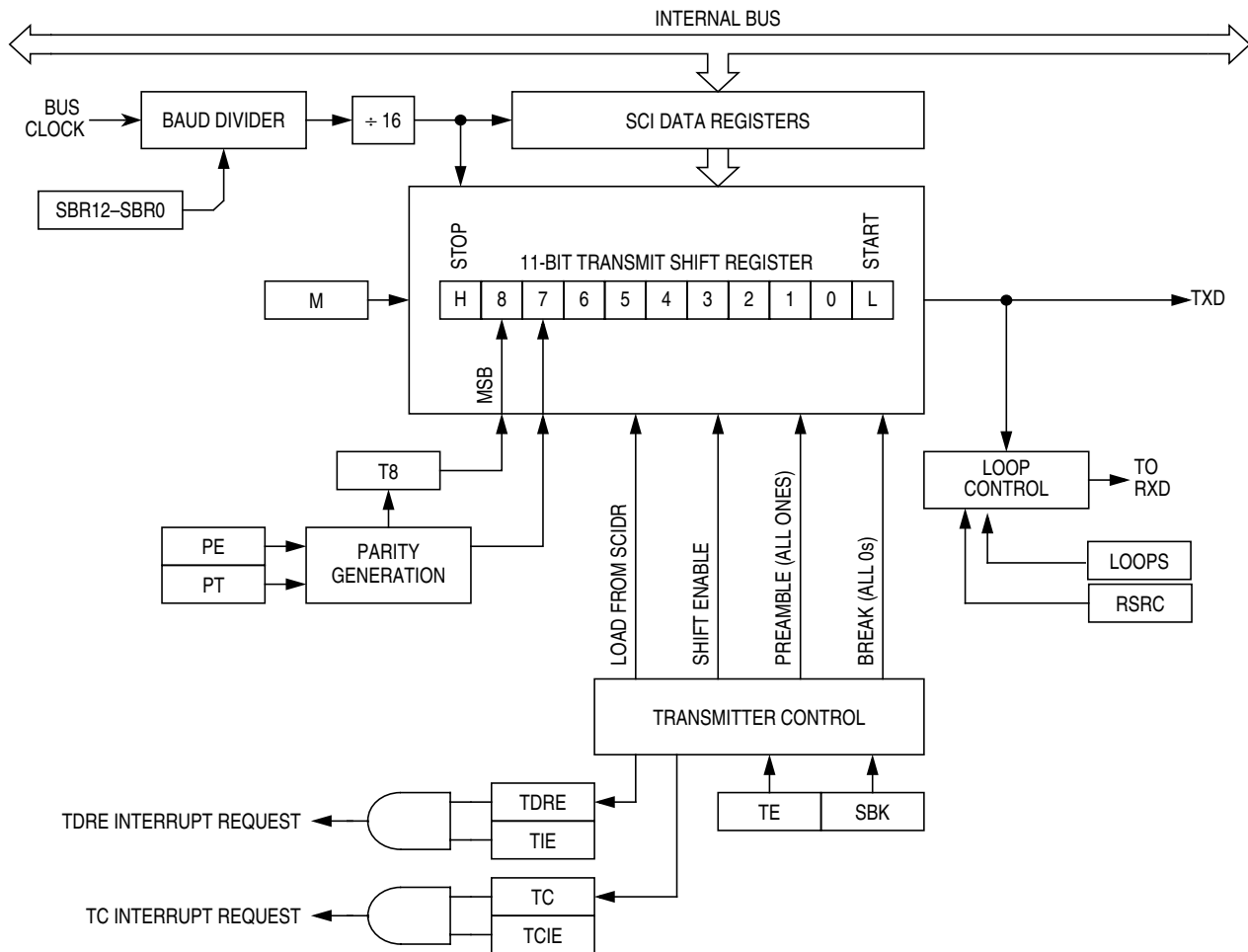


Figure 13-11. Transmitter Block Diagram

13.4.3.1 Transmitter Character Length

The SCI transmitter can accommodate either 8-bit or 9-bit data characters. The state of the M bit in SCI control register 1 (SCICR1) determines the length of data characters. When transmitting 9-bit data, bit T8 in SCI data register high (SCIDRH) is the ninth bit (bit 8).

13.4.3.2 Character Transmission

To transmit data, the MCU writes the data bits to the SCI data registers (SCIDRH/SCIDRL), which in turn are transferred to the transmitter shift register. The transmit shift register then shifts a frame out through the **Tx output** signal, after it has prefaced them with a start bit and appended them with a stop bit. The SCI data registers (SCIDRH and SCIDRL) are the write-only buffers between the internal data bus and the transmit shift register.

The SCI also sets a flag, the transmit data register empty flag (TDRE), every time it transfers data from the buffer (SCIDRH/L) to the transmitter shift register. The transmit driver routine may respond to this flag by

Table 14-2. SPICR1 Field Descriptions

Field	Description
7 SPIE	SPI Interrupt Enable Bit — This bit enables SPI interrupt requests, if SPIF or MODF status flag is set. 0 SPI interrupts disabled. 1 SPI interrupts enabled.
6 SPE	SPI System Enable Bit — This bit enables the SPI system and dedicates the SPI port pins to SPI system functions. If SPE is cleared, SPI is disabled and forced into idle state, status bits in SPISR register are reset. 0 SPI disabled (lower power consumption). 1 SPI enabled, port pins are dedicated to SPI functions.
5 SPTIE	SPI Transmit Interrupt Enable — This bit enables SPI interrupt requests, if SPTEF flag is set. 0 SPTEF interrupt disabled. 1 SPTEF interrupt enabled.
4 MSTR	SPI Master/Slave Mode Select Bit — This bit selects, if the SPI operates in master or slave mode. Switching the SPI from master to slave or vice versa forces the SPI system into idle state. 0 SPI is in slave mode 1 SPI is in master mode
3 CPOL	SPI Clock Polarity Bit — This bit selects an inverted or non-inverted SPI clock. To transmit data between SPI modules, the SPI modules must have identical CPOL values. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 Active-high clocks selected. In idle state SCK is low. 1 Active-low clocks selected. In idle state SCK is high.
2 CPHA	SPI Clock Phase Bit — This bit is used to select the SPI clock format. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 Sampling of data occurs at odd edges (1,3,5,...,15) of the SCK clock 1 Sampling of data occurs at even edges (2,4,6,...,16) of the SCK clock
1 SSOE	Slave Select Output Enable — The \overline{SS} output feature is enabled only in master mode, if MODFEN is set, by asserting the SSOE as shown in Table 14-3. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state.
0 LSBFE	LSB-First Enable — This bit does not affect the position of the MSB and LSB in the data register. Reads and writes of the data register always have the MSB in bit 7. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 Data is transferred most significant bit first. 1 Data is transferred least significant bit first.

Table 14-3. \overline{SS} Input / Output Selection

MODFEN	SSOE	Master Mode	Slave Mode
0	0	\overline{SS} not used by SPI	\overline{SS} input
0	1	\overline{SS} not used by SPI	\overline{SS} input
1	0	\overline{SS} input with MODF feature	\overline{SS} input
1	1	\overline{SS} is slave select output	\overline{SS} input

15.3.2.14 Timer Input Capture/Output Compare Registers High and Low 0–7 (TCxH and TCxL)

Module Base + 0x0010 = TC0H 0x0018 = TC4H
 0x0012 = TC1H 0x001A = TC5H
 0x0014 = TC2H 0x001C = TC6H
 0x0016 = TC3H 0x001E = TC7H

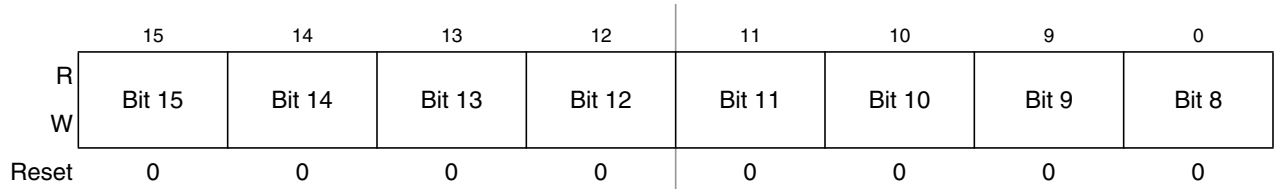


Figure 15-22. Timer Input Capture/Output Compare Register x High (TCxH)

Module Base + 0x0011 = TC0L 0x0019 = TC4L
 0x0013 = TC1L 0x001B = TC5L
 0x0015 = TC2L 0x001D = TC6L
 0x0017 = TC3L 0x001F = TC7L

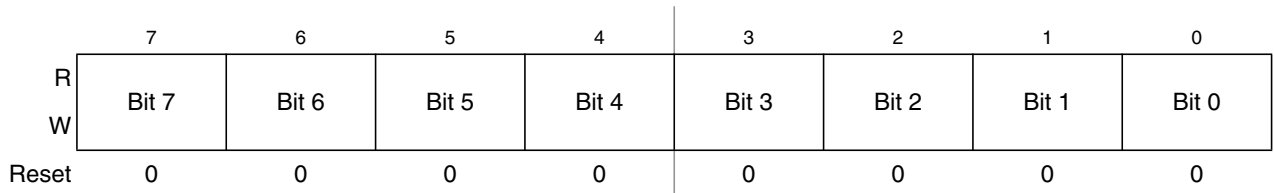


Figure 15-23. Timer Input Capture/Output Compare Register x Low (TCxL)

Depending on the TIOS bit for the corresponding channel, these registers are used to latch the value of the free-running counter when a defined transition is sensed by the corresponding input capture edge detector or to trigger an output action for output compare.

Read: Anytime

Write: Anytime for output compare function. Writes to these registers have no meaning or effect during input capture. All timer input capture/output compare registers are reset to 0x0000.

NOTE

Read/Write access in byte mode for high byte should takes place before low byte otherwise it will give a different result.

15.4.2 Input Capture

Clearing the I/O (input/output) select bit, IOSx, configures channel x as an input capture channel. The input capture function captures the time at which an external event occurs. When an active edge occurs on the pin of an input capture channel, the timer transfers the value in the timer counter into the timer channel registers, TCx.

The minimum pulse width for the input capture input is greater than two bus clocks.

An input capture on channel x sets the CxF flag. The CxI bit enables the CxF flag to generate interrupt requests. Timer module must stay enabled (TEN bit of TSCR1 must be set to one) while clearing CxF (writing one to CxF).

15.4.3 Output Compare

Setting the I/O select bit, IOSx, configures channel x as an output compare channel. The output compare function can generate a periodic pulse with a programmable polarity, duration, and frequency. When the timer counter reaches the value in the channel registers of an output compare channel, the timer can set, clear, or toggle the channel pin. An output compare on channel x sets the CxF flag. The CxI bit enables the CxF flag to generate interrupt requests. Timer module must stay enabled (TEN bit of TSCR1 register must be set to one) while clearing CxF (writing one to CxF).

The output mode and level bits, OMx and OLx, select set, clear, toggle on output compare. Clearing both OMx and OLx disconnects the pin from the output logic.

Setting a force output compare bit, FOCx, causes an output compare on channel x. A forced output compare does not set the channel flag.

A successful output compare on channel 7 overrides output compares on all other output compare channels. The output compare 7 mask register masks the bits in the output compare 7 data register. The timer counter reset enable bit, TCRE, enables channel 7 output compares to reset the timer counter. A channel 7 output compare can reset the timer counter even if the IOC7 pin is being used as the pulse accumulator input.

Writing to the timer port bit of an output compare pin does not affect the pin state. The value written is stored in an internal latch. When the pin becomes available for general-purpose output, the last value written to the bit appears at the pin.

When TCRE is set and TC7 is not equal to 0, then TCNT will cycle from 0 to TC7. When TCNT reaches TC7 value, it will last only one bus cycle then reset to 0.

16.2.3 V_{DD} , V_{SS} — Regulator Output1 (Core Logic)

Signals V_{DD}/V_{SS} are the primary outputs of VREG3V3V2 that provide the power supply for the core logic. These signals are connected to device pins to allow external decoupling capacitors (100 nF...220 nF, X7R ceramic).

In Shutdown Mode an external supply at V_{DD}/V_{SS} can replace the voltage regulator.

16.2.4 V_{DDPLL} , V_{SSPLL} — Regulator Output2 (PLL)

Signals V_{DDPLL}/V_{SSPLL} are the secondary outputs of VREG3V3V2 that provide the power supply for the PLL and oscillator. These signals are connected to device pins to allow external decoupling capacitors (100 nF...220 nF, X7R ceramic).

In Shutdown Mode an external supply at V_{DDPLL}/V_{SSPLL} can replace the voltage regulator.

16.2.5 V_{REGEN} — Optional Regulator Enable

This optional signal is used to shutdown VREG3V3V2. In that case V_{DD}/V_{SS} and V_{DDPLL}/V_{SSPLL} must be provided externally. Shutdown Mode is entered with V_{REGEN} being low. If V_{REGEN} is high, the VREG3V3V2 is either in Full Performance Mode or in Reduced Power Mode.

For the connectivity of V_{REGEN} see device overview chapter.

NOTE

Switching from FPM or RPM to shutdown of VREG3V3V2 and vice versa is not supported while the MCU is powered.

16.3 Memory Map and Register Definition

This subsection provides a detailed description of all registers accessible in VREG3V3V2.

16.3.1 Module Memory Map

Figure 16-2 provides an overview of all used registers.

Table 16-2. VREG3V3V2 Memory Map

Address Offset	Use	Access
0x0000	VREG3V3V2 Control Register (VREGCTRL)	R/W

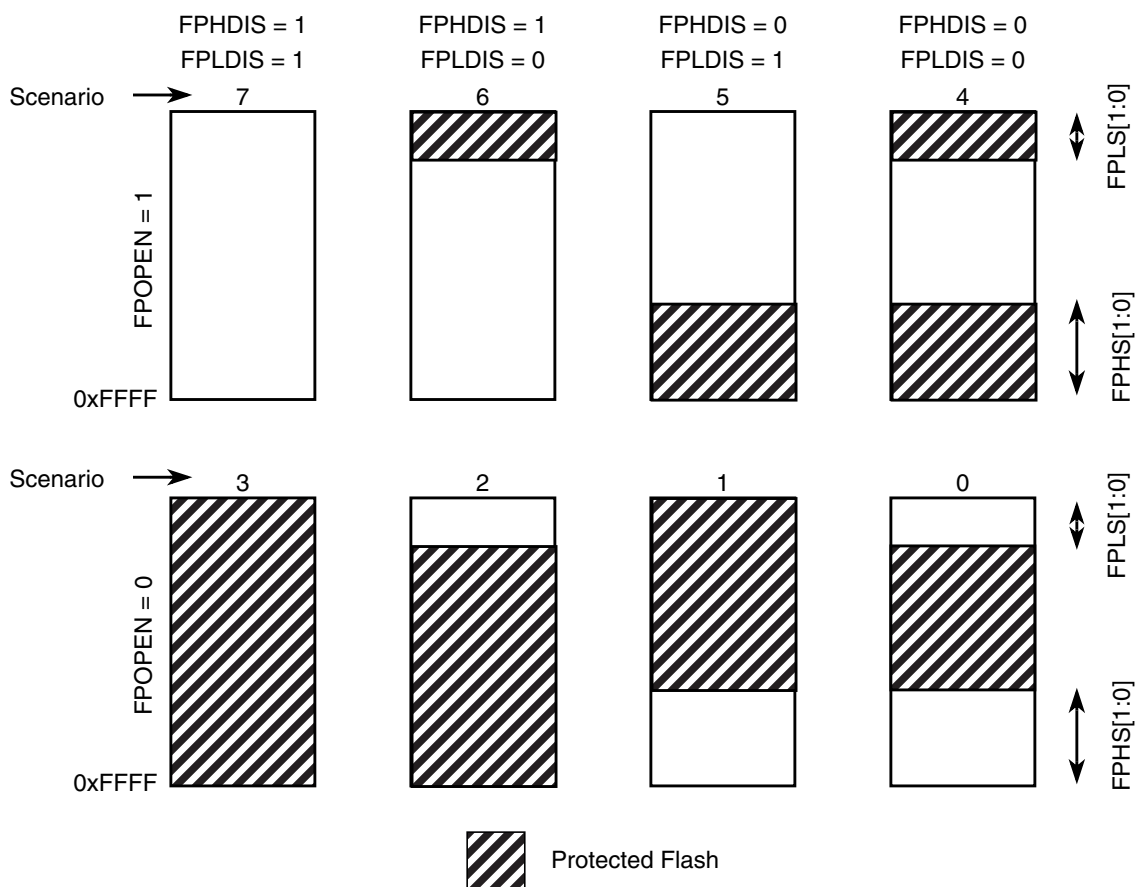


Figure 18-9. Flash Protection Scenarios

18.3.2.5.1 Flash Protection Restrictions

The general guideline is that protection can only be added, not removed. All valid transitions between Flash protection scenarios are specified in [Table 18-12](#). Any attempt to write an invalid scenario to the FPROT register will be ignored and the FPROT register will remain unchanged. The contents of the FPROT register reflect the active protection scenario.

Table 18-12. Flash Protection Scenario Transitions

From Protection Scenario	To Protection Scenario ⁽¹⁾							
	0	1	2	3	4	5	6	7
0	X	X	X	X				
1		X		X				
2			X	X				
3				X				
4				X	X			
5			X	X	X	X		

Module Base + 0x000C

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

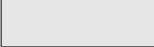
 = Unimplemented or Reserved

Figure 18-17. RESERVED3

All bits read 0 and are not writable.

18.3.2.12 RESERVED4

This register is reserved for factory testing and is not accessible to the user.

Module Base + 0x000D

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0


 = Unimplemented or Reserved

Figure 18-18. RESERVED4

All bits read 0 and are not writable.

18.3.2.13 RESERVED5

This register is reserved for factory testing and is not accessible to the user.

Module Base + 0x000E

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0


 = Unimplemented or Reserved

Figure 18-19. RESERVED5

All bits read 0 and are not writable.

18.3.2.14 RESERVED6

This register is reserved for factory testing and is not accessible to the user.

then FCLKDIV register bits PRDIV8 and FDIV[5:0] are to be set as described in [Figure 20-23](#).

For example, if the oscillator clock frequency is 950 kHz and the bus clock is 10 MHz, FCLKDIV bits FDIV[5:0] should be set to 4 (000100) and bit PRDIV8 set to 0. The resulting FCLK is then 190 kHz. As a result, the Flash algorithm timings are increased over optimum target by:

$$(200 - 190) / 200 \times 100 = 5\%$$

Command execution time will increase proportionally with the period of FCLK.

CAUTION

Because of the impact of clock synchronization on the accuracy of the functional timings, programming or erasing the Flash array cannot be performed if the bus clock runs at less than 1 MHz. Programming or erasing the Flash array with an input clock < 150 kHz should be avoided. Setting FCLKDIV to a value such that FCLK < 150 kHz can destroy the Flash array due to overstress. Setting FCLKDIV to a value such that $(1/\text{FCLK} + T_{\text{bus}}) < 5\mu\text{s}$ can result in incomplete programming or erasure of the Flash array cells.

If the FCLKDIV register is written, the bit FDIVLD is set automatically. If the FDIVLD bit is 0, the FCLKDIV register has not been written since the last reset. If the FCLKDIV register has not been written to, the Flash command loaded during a command write sequence will not execute and the ACCERR flag in the FSTAT register will set.

then FCLKDIV register bits PRDIV8 and FDIV[5:0] are to be set as described in [Figure 21-21](#).

For example, if the oscillator clock frequency is 950 kHz and the bus clock is 10 MHz, FCLKDIV bits FDIV[5:0] should be set to 4 (000100) and bit PRDIV8 set to 0. The resulting FCLK is then 190 kHz. As a result, the Flash algorithm timings are increased over optimum target by:

$$(200 - 190) / 200 \times 100 = 5\%$$

Command execution time will increase proportionally with the period of FCLK.

CAUTION

Because of the impact of clock synchronization on the accuracy of the functional timings, programming or erasing the Flash array cannot be performed if the bus clock runs at less than 1 MHz. Programming or erasing the Flash array with an input clock < 150 kHz should be avoided. Setting FCLKDIV to a value such that FCLK < 150 kHz can destroy the Flash array due to overstress. Setting FCLKDIV to a value such that $(1/\text{FCLK} + T_{\text{bus}}) < 5\mu\text{s}$ can result in incomplete programming or erasure of the Flash array cells.

If the FCLKDIV register is written, the bit FDIVLD is set automatically. If the FDIVLD bit is 0, the FCLKDIV register has not been written since the last reset. If the FCLKDIV register has not been written to, the Flash command loaded during a command write sequence will not execute and the ACCERR flag in the FSTAT register will set.

A.4.1.3 SRAM Data Retention

Provided an appropriate external reset signal is applied to the MCU, preventing the CPU from executing code when V_{DD5} is out of specification limits, the SRAM contents integrity is guaranteed if after the reset the PORF bit in the CRG Flags Register has not been set.

A.4.1.4 External Reset

When external reset is asserted for a time greater than PW_{RSTL} the CRG module generates an internal reset, and the CPU starts fetching the reset vector without doing a clock quality check, if there was an oscillation before reset.

A.4.1.5 Stop Recovery

Out of STOP the controller can be woken up by an external interrupt. A clock quality check as after POR is performed before releasing the clocks to the system.

A.4.1.6 Pseudo Stop and Wait Recovery

The recovery from Pseudo STOP and Wait are essentially the same since the oscillator was not stopped in both modes. In Pseudo Stop Mode the voltage regulator is switched to reduced performance mode to reduce power consumption. The returning out of pseudo stop to full performance takes t_{vup} . The controller can be woken up by internal or external interrupts. After t_{wrs} in Wait or $t_{vup} + t_{wrs}$ in Pseudo Stop the CPU starts fetching the interrupt vector.

A.4.2 Oscillator

The device features an internal Colpitts and Pierce oscillator. The selection of Colpitts oscillator or Pierce oscillator/external clock depends on the XCLKS signal which is sampled during reset. Pierce oscillator/external clock mode allows the input of a square wave. Before asserting the oscillator to the internal system clocks the quality of the oscillation is checked for each start from either power-on, STOP or oscillator fail. t_{CQOUT} specifies the maximum time before switching to the internal self clock mode after POR or STOP if a proper oscillation is not detected. The quality check also determines the minimum oscillator start-up time t_{UPOSC} . The device also features a clock monitor. A Clock Monitor Failure is asserted if the frequency of the incoming clock signal is below the Assert Frequency f_{CMFA} .

A.5.2 NVM Reliability

The reliability of the NVM blocks is guaranteed by stress test during qualification, constant process monitors and burn-in to screen early life failures. The program/erase cycle count on the sector is incremented every time a sector or mass erase event is executed.

Table A-19. NVM Reliability Characteristics⁽¹⁾

Conditions are shown in Table A-4. unless otherwise noted							
Num	C	Rating	Symbol	Min	Typ	Max	Unit
Flash Reliability Characteristics							
1	C	Data retention after 10,000 program/erase cycles at an average junction temperature of $T_{Javg} \leq 85^{\circ}\text{C}$	t_{FLRET}	15	$100^{(2)}$	—	Years
2	C	Data retention with <100 program/erase cycles at an average junction temperature $T_{Javg} \leq 85^{\circ}\text{C}$		20	100^2	—	
3	C	Number of program/erase cycles ($-40^{\circ}\text{C} \leq T_J \leq 0^{\circ}\text{C}$)	n_{FL}	10,000	—	—	Cycles
4	C	Number of program/erase cycles ($0^{\circ}\text{C} \leq T_J \leq 140^{\circ}\text{C}$)		10,000	$100,000^{(3)}$	—	

- ¹ T_{Javg} will not exceed 85°C considering a typical temperature profile over the lifetime of a consumer, industrial or automotive application.
- Typical data retention values are based on intrinsic capability of the technology measured at high temperature and de-rated to 25°C using the Arrhenius equation. For additional information on how Freescale defines Typical Data Retention, please refer to Engineering Bulletin EB618.
- Spec table quotes typical endurance evaluated at 25°C for this product family, typical endurance at various temperature can be estimated using the graph below. For additional information on how Freescale defines Typical Endurance, please refer to Engineering Bulletin EB619.