Welcome to 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	Active
Core Processor	HCS12X
Core Size	16-Bit
Speed	50MHz
Connectivity	CANbus, EBI/EMI, I²C, IrDA, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	91
Program Memory Size	384KB (384K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	24K x 8
Voltage - Supply (Vcc/Vdd)	1.72V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s12xeq384cal

2.3.67 Port J Interrupt Enable Register (PIEJ)

Address 0x026E — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	PIEJ7	PIEJ6	PIEJ5	PIEJ4	PIEJ3	PIEJ2	PIEJ1	PIEJ0
Reset	0	0	0	0	0	0	0	0

Figure 2-65. Port J Interrupt Enable Register (PIEJ)

1. Read: Anytime.
   Write: Anytime.

Table 2-63. PPSP Register Field Descriptions

Field	Description
7-0 PIEJ	**Port J interrupt enable**— This register disables or enables on a per-pin basis the edge sensitive external interrupt associated with Port J. 1 Interrupt is enabled. 0 Interrupt is disabled (interrupt flag masked).

2.3.68 Port J Interrupt Flag Register (PIFJ)

Address 0x026F — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	PIFJ7	PIFJ6	PIFJ5	PIFJ4	PIFJ3	PIFJ2	PIFJ1	PIFJ0
Reset	0	0	0	0	0	0	0	0

Figure 2-66. Port J Interrupt Flag Register (PIFJ)

1. Read: Anytime.
   Write: Anytime.

Table 2-64. PPSP Register Field Descriptions

Field	Description
7-0 PIFJ	**Port J interrupt flag**— Each flag is set by an active edge on the associated input pin. This could be a rising or a falling edge based on the state of the PPSJ register. To clear this flag, write logic level 1 to the corresponding bit in the PIFJ register. Writing a 0 has no effect. 1 Active edge on the associated bit has occurred (an interrupt will occur if the associated enable bit is set). 0 No active edge pending.

2.3.87 Port R Data Direction Register (DDRR)

Address 0x036A — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	DDRR7	DDRR6	DDRR5	DDRR4	DDRR3	DDRR2	DDRR1	DDRR0
Reset	0	0	0	0	0	0	0	0

Figure 2-85. Port R Data Direction Register (DDRR)

1. Read: Anytime.
   Write: Anytime.

Table 2-83. DDRR Register Field Descriptions

Field	Description
7-0 DDRR	**Port R data direction**— This register controls the data direction of pins 7 through 0. The TIM forces the I/O state to be an output for each timer port associated with an enabled output compare. In this case the data direction bits will not change. The data direction bits revert to controlling the I/O direction of a pin when the associated timer output compare is disabled. The timer Input Capture always monitors the state of the pin. 1 Associated pin is configured as output. 0 Associated pin is configured as high-impedance input.

NOTE

Due to internal synchronization circuits, it can take up to 2 bus clock cycles until the correct value is read on PTR or PTIR registers, when changing the DDRR register.

2.3.88 Port R Reduced Drive Register (RDRR)

Address 0x036B — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	RDRR7	RDRR6	RDRR5	RDRR4	RDRR3	RDRR2	RDRR1	RDRR0
Reset	0	0	0	0	0	0	0	0

Figure 2-86. Port R Reduced Drive Register (RDRR)

1. Read: Anytime.
   Write: Anytime.

2.3.98 Port L Polarity Select Register (PPSL)

Address 0x0375 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	PPSL7	PPSL6	PPSL5	PPSL4	PPSL3	PPSL2	PPSL1	PPSL0
Reset	0	0	0	0	0	0	0	0

Figure 2-96. Port L Polarity Select Register (PPSL)

1. Read: Anytime.
   Write: Anytime.

Table 2-93. PPSL Register Field Descriptions

Field	Description
7-0 PPSL	**Port L pull device select**—Determine pull device polarity on input pins This register selects whether a pull-down or a pull-up device is connected to the pin. 1 A pull-down device is connected to the associated pin, if enabled and if the pin is used as input. 0 A pull-up device is connected to the associated pin, if enabled and if the pin is used as input.

2.3.99 Port L Wired-Or Mode Register (WOML)

Address 0x0376 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	WOML7	WOML6	WOML5	WOML4	WOML3	WOML2	WOML1	WOML0
Reset	0	0	0	0	0	0	0	0

Figure 2-97. Port L Wired-Or Mode Register (WOML)

1. Read: Anytime.
   Write: Anytime.

Table 2-94. WOML Register Field Descriptions

Field	Description
7-0 WOML	**Port L wired-or mode**—Enable wired-or functionality This register configures the output pins as wired-or independent of the function used on the pins. If enabled the output is driven active low only (open-drain). A logic level of “1” is not driven.This allows a multipoint connection of several serial modules. These bits have no influence on pins used as inputs. 1 Output buffers operate as open-drain outputs. 0 Output buffers operate as push-pull outputs.

Table 3-14. RPAGE Field Descriptions

Field	Description
7–0 RP[7:0]	**RAM Page Index Bits 7–0** — These page index bits are used to select which of the 256 RAM array pages is to be accessed in the RAM Page Window.

The reset value of 0xFD ensures that there is a linear RAM space available between addresses 0x1000 and 0x3FFF out of reset.

The fixed 4K page from 0x2000–0x2FFF of RAM is equivalent to page 254 (page number 0xFE).

The fixed 4K page from 0x3000–0x3FFF of RAM is equivalent to page 255 (page number 0xFF).

3.3.2.8 EEPROM Page Index Register (EPAGE)

Address: 0x0017

	7	6	5	4	3	2	1	0
R W	EP7	EP6	EP5	EP4	EP3	EP2	EP1	EP0
Reset	1	1	1	1	1	1	1	0

Figure 3-15. EEPROM Page Index Register (EPAGE)

Read: Anytime

Write: Anytime

These eight index bits are used to page 1 KByte blocks into the EEPROM page window located in the local (CPU or BDM) memory map from address 0x0800 to address 0x0BFF (see Figure 3-16). This supports accessing up to 256 KByte of EEPROM (in the Global map) within the 64 KByte Local map. The EEPROM page index register is effectively used to construct paged EEPROM addresses in the Local map format.

CAUTION

XGATE write access to this register during an CPU access which makes use of this register could lead to unexpected results.

6.5.3.2 XGATE Wake Up from Stop or Wait Mode

Interrupt request channels which are configured to be handled by the XGATE module are capable of waking up the XGATE module. Interrupt request channels handled by the XGATE module do not affect the state of the CPU.

8.3.2 Register Descriptions

This section consists of the S12XDBG control and trace buffer register descriptions in address order. Each comparator has a bank of registers that are visible through an 8-byte window between 0x0028 and 0x002F in the S12XDBG module register address map. When ARM is set in DBGC1, the only bits in the S12XDBG module registers that can be written are ARM, TRIG, and COMRV[1:0].

8.3.2.1 Debug Control Register 1 (DBGC1)

Address: 0x0020

	7	6	5	4	3	2	1	0
R	ARM	0	XGSBPE	BDM	DBGBRK		COMRV	
W		TRIG						
Reset	0	0	0	0	0	0	0	0

Figure 8-3. Debug Control Register (DBGC1)

Read: Anytime

Write: Bits 7, 1, 0 anytime
Bit 6 can be written anytime but always reads back as 0.
Bits 5:2 anytime S12XDBG is not armed.

NOTE

If a write access to DBGC1 with the ARM bit position set occurs simultaneously to a hardware disarm from an internal trigger event, then the ARM bit is cleared due to the hardware disarm.

NOTE

When disarming the S12XDBG by clearing ARM with software, the contents of bits[5:2] are not affected by the write, since up until the write operation, ARM = 1 preventing these bits from being written. These bits must be cleared using a second write if required.

Table 8-5. DBGC1 Field Descriptions

Field	Description
7 ARM	**Arm Bit** — The ARM bit controls whether the S12XDBG module is armed. This bit can be set and cleared by user software and is automatically cleared on completion of a tracing session, or if a breakpoint is generated with tracing not enabled. On setting this bit the state sequencer enters State1. 0 Debugger disarmed 1 Debugger armed
6 TRIG	**Immediate Trigger Request Bit** — This bit when written to 1 requests an immediate trigger independent of comparator or external tag signal status. When tracing is complete a forced breakpoint may be generated depending upon DBGBRK and BDM bit settings. This bit always reads back a 0. Writing a 0 to this bit has no effect. If TSOURCE are clear no tracing is carried out. If tracing has already commenced using BEGIN- or MID trigger alignment, it continues until the end of the tracing session as defined by the TALIGN bit settings, thus TRIG has no affect. In secure mode tracing is disabled and writing to this bit has no effect. 0 Do not trigger until the state sequencer enters the Final State. 1 Trigger immediately .

LDB — Load Byte from Memory (Low Byte) — LDB

Operation

M[RB, #OFFS5] ⇒ RD.L; $00 ⇒ RD.H
M[RB, RI] ⇒ RD.L; $00 ⇒ RD.H
M[RB, RI] ⇒ RD.L; $00 ⇒ RD.H; RI+1 ⇒ RI;[1]
RI-1 ⇒ RI; M[RS, RI] ⇒ RD.L; $00 ⇒ RD.H

Loads a byte from memory into the low byte of register RD. The high byte is cleared.

CCR Effects

N	Z	V	C
—	—	—	—

N: Not affected.
Z: Not affected.
V: Not affected.
C: Not affected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code										Cycles
LDB RD, (RB, #OFFS5)	IDO5	0	1	0	0	0	RD	RB	OFFS5			Pr
LDB RD, (RS, RI)	IDR	0	1	1	0	0	RD	RB	RI	0	0	Pr
LDB RD, (RS, RI+)	IDR+	0	1	1	0	0	RD	RB	RI	0	1	Pr
LDB RD, (RS, -RI)	-IDR	0	1	1	0	0	RD	RB	RI	1	0	Pr

1. If the same general purpose register is used as index (RI) and destination register (RD), the content of the register will not be incremented after the data move: M[RB, RI] ⇒ RD.L; $00 ⇒ RD.H

14.3.2.11 Timer System Control Register 2 (TSCR2)

Module Base + 0x000D

	7	6	5	4	3	2	1	0
R	TOI	0	0	0	TCRE	PR2	PR1	PR0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 14-16. Timer System Control Register 2 (TSCR2)

Read or write: Anytime

All bits reset to zero.

Table 17-5. PITMUX Field Descriptions

Field	Description
7:0 PMUX[7:0]	**PIT Multiplex Bits for Timer Channel 7:0** — These bits select if the corresponding 16-bit timer is connected to micro time base 1 or 0. If PMUX is modified, the corresponding 16-bit timer is immediately switched to the other micro time base. 0 The corresponding 16-bit timer counts with micro time base 0. 1 The corresponding 16-bit timer counts with micro time base 1.

17.3.0.5 PIT Interrupt Enable Register (PITINTE)

Module Base + 0x0004

	7	6	5	4	3	2	1	0
R W	PINTE7	PINTE6	PINTE5	PINTE4	PINTE3	PINTE2	PINTE1	PINTE0
Reset	0	0	0	0	0	0	0	0

Figure 17-7. PIT Interrupt Enable Register (PITINTE)

Read: Anytime

Write: Anytime

Table 17-6. PITINTE Field Descriptions

Field	Description
7:0 PINTE[7:0]	**PIT Time-out Interrupt Enable Bits for Timer Channel 7:0** — These bits enable an interrupt service request whenever the time-out flag PTF of the corresponding PIT channel is set. When an interrupt is pending (PTF set) enabling the interrupt will immediately cause an interrupt. To avoid this, the corresponding PTF flag has to be cleared first. 0 Interrupt of the corresponding PIT channel is disabled. 1 Interrupt of the corresponding PIT channel is enabled.

17.3.0.6 PIT Time-Out Flag Register (PITTF)

Module Base + 0x0005

	7	6	5	4	3	2	1	0
R W	PTF7	PTF6	PTF5	PTF4	PTF3	PTF2	PTF1	PTF0
Reset	0	0	0	0	0	0	0	0

Figure 17-8. PIT Time-Out Flag Register (PITTF)

Read: Anytime

Write: Anytime (write to clear)

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000F PITCNT1 (Low)	R	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
	W								
0x0010 PITLD2 (High)	R	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
	W								
0x0011 PITLD2 (Low)	R	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
	W								
0x0012 PITCNT2 (High)	R	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
	W								
0x0013 PITCNT2 (Low)	R	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
	W								
0x0014 PITLD3 (High)	R	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
	W								
0x0015 PITLD3 (Low)	R	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
	W								
0x0016 PITCNT3 (High)	R	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
	W								
0x0017 PITCNT3 (Low)	R	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
	W								
0x0018–0x0027 RESERVED	R	0	0	0	0	0	0	0	0
	W								

= Unimplemented or Reserved

Figure 18-2. PIT Register Summary (Sheet 2 of 2)

18.3.0.1 PIT Control and Force Load Micro Timer Register (PITCFLMT)

Module Base + 0x0000

	7	6	5	4	3	2	1	0
R	PITE	PITSWAI	PITFRZ	0	0	0	0	0
W							PFLMT1	PFLMT0
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 18-3. PIT Control and Force Load Micro Timer Register (PITCFLMT)

Read: Anytime

Write: Anytime; writes to the reserved bits have no effect

18.3.0.3 PIT Channel Enable Register (PITCE)

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R	0	0	0	0	PCE3	PCE2	PCE1	PCE0
W								
Reset	0	0	0	0	0	0	0	0

Figure 18-5. PIT Channel Enable Register (PITCE)

Read: Anytime

Write: Anytime

Table 18-4. PITCE Field Descriptions

Field	Description
3:0 PCE[3:0]	**PIT Enable Bits for Timer Channel 3:0** — These bits enable the PIT channels 3-0. If PCE is cleared, the PIT channel is disabled and the corresponding flag bit in the PITTF register is cleared. When PCE is set, and if the PIT module is enabled (PITE = 1) the 16-bit timer counter is loaded with the start count value and starts down-counting. 0 The corresponding PIT channel is disabled. 1 The corresponding PIT channel is enabled.

20.4.5.5 LIN Transmit Collision Detection

This module allows to check for collisions on the LIN bus.

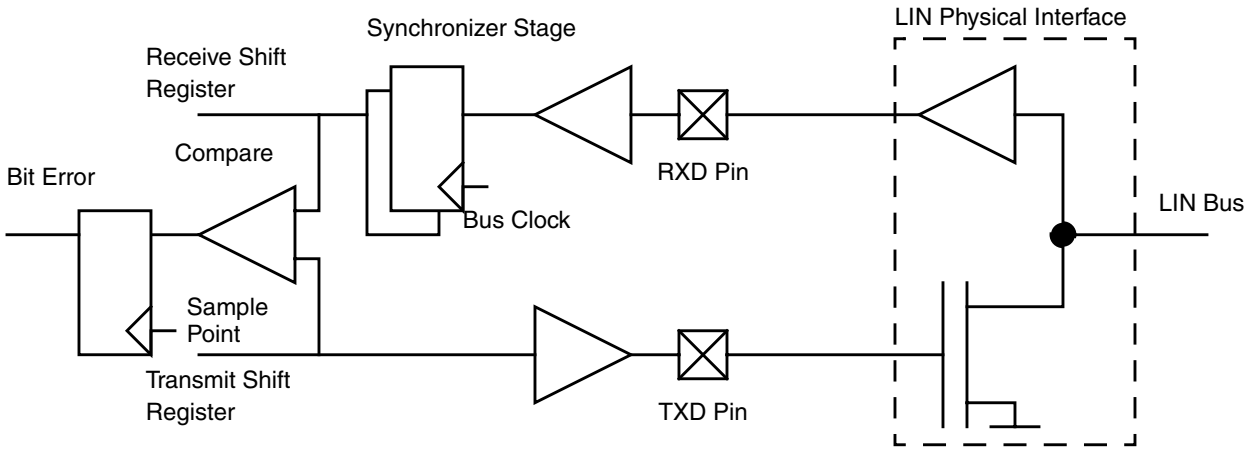

Figure 20-18. Collision Detect Principle

If the bit error circuit is enabled (BERRM[1:0] = 0:1 or = 1:0]), the error detect circuit will compare the transmitted and the received data stream at a point in time and flag any mismatch. The timing checks run when transmitter is active (not idle). As soon as a mismatch between the transmitted data and the received data is detected the following happens:

- The next bit transmitted will have a high level (TXPOL = 0) or low level (TXPOL = 1)
- The transmission is aborted and the byte in transmit buffer is discarded.
- the transmit data register empty and the transmission complete flag will be set
- The bit error interrupt flag, BERRIF, will be set.
- No further transmissions will take place until the BERRIF is cleared.

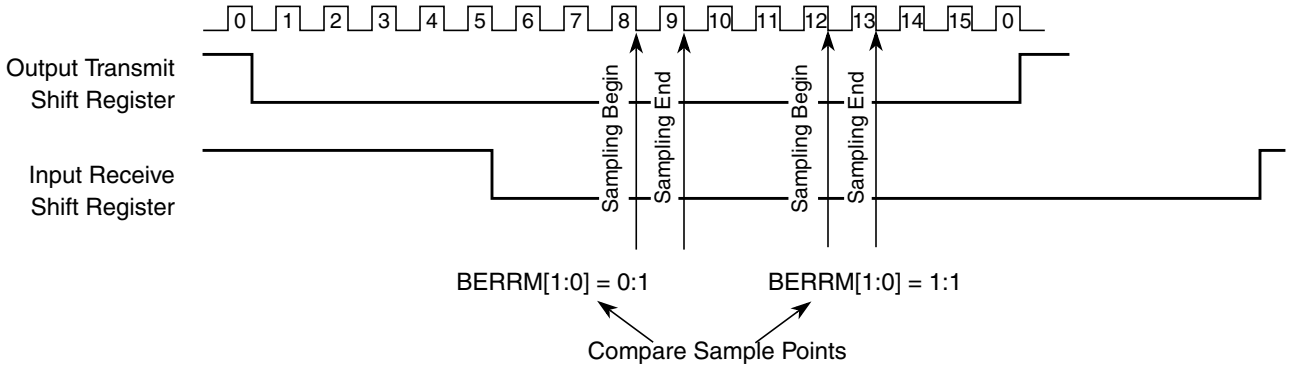

Figure 20-19. Timing Diagram Bit Error Detection

If the bit error detect feature is disabled, the bit error interrupt flag is cleared.

NOTE

The RXPOL and TXPOL bit should be set the same when transmission collision detect feature is enabled, otherwise the bit error interrupt flag may be set incorrectly.

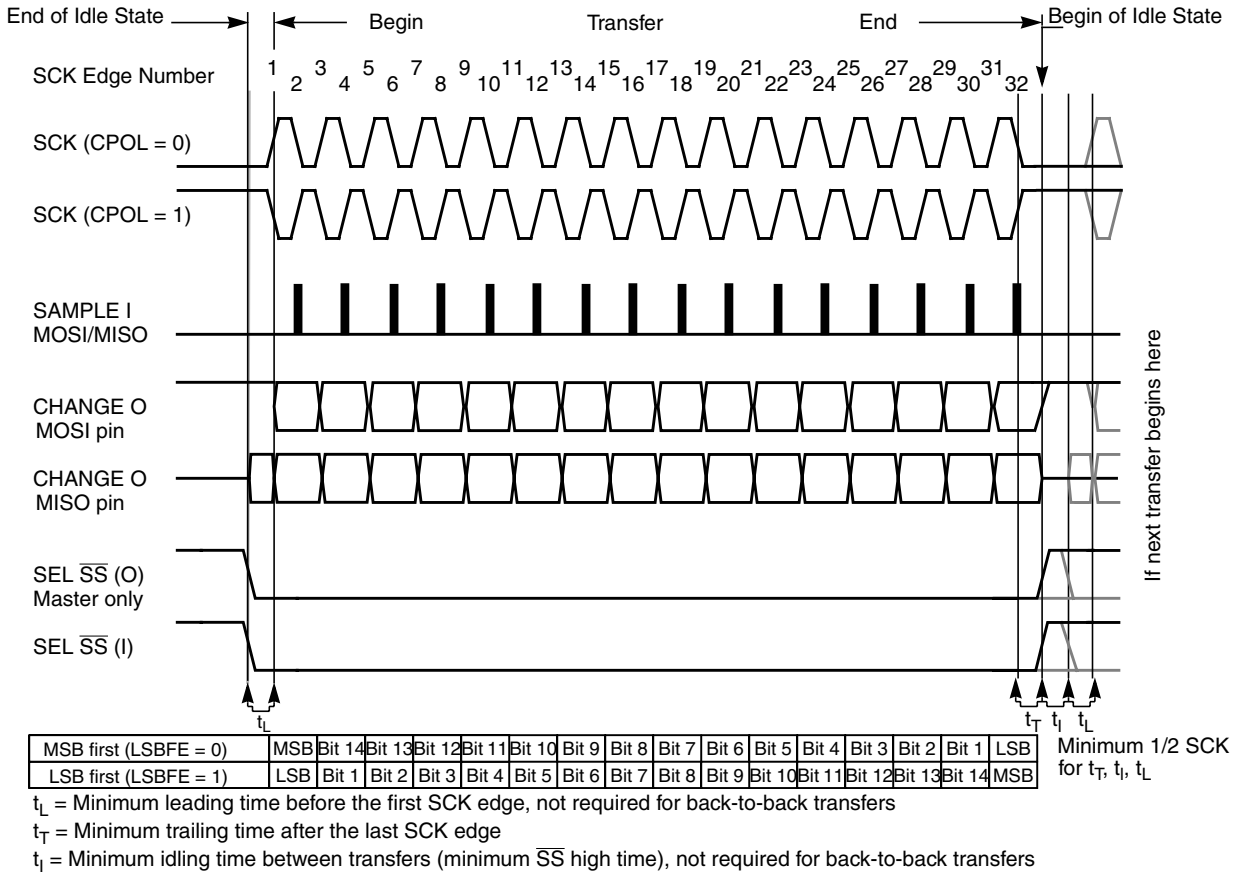

Figure 21-15. SPI Clock Format 1 (CPHA = 1), with 16-Bit Transfer Width selected (XFRW = 1)

The $\overline{SS}$ line can remain active low between successive transfers (can be tied low at all times). This format is sometimes preferred in systems having a single fixed master and a single slave that drive the MISO data line.

- Back-to-back transfers in master mode

  In master mode, if a transmission has completed and new data is available in the SPI data register, this data is sent out immediately without a trailing and minimum idle time.

The SPI interrupt request flag (SPIF) is common to both the master and slave modes. SPIF gets set one half SCK cycle after the last SCK edge.

21.4.4 SPI Baud Rate Generation

Baud rate generation consists of a series of divider stages. Six bits in the SPI baud rate register (SPPR2, SPPR1, SPPR0, SPR2, SPR1, and SPR0) determine the divisor to the SPI module clock which results in the SPI baud rate.

The SPI clock rate is determined by the product of the value in the baud rate preselection bits (SPPR2–SPPR0) and the value in the baud rate selection bits (SPR2–SPR0). The module clock divisor equation is shown in Equation 21-3.

$$\text{BaudRateDivisor} = (\text{SPPR} + 1) \bullet 2^{(\text{SPR} + 1)} \qquad \textit{Eqn. 21-3}$$

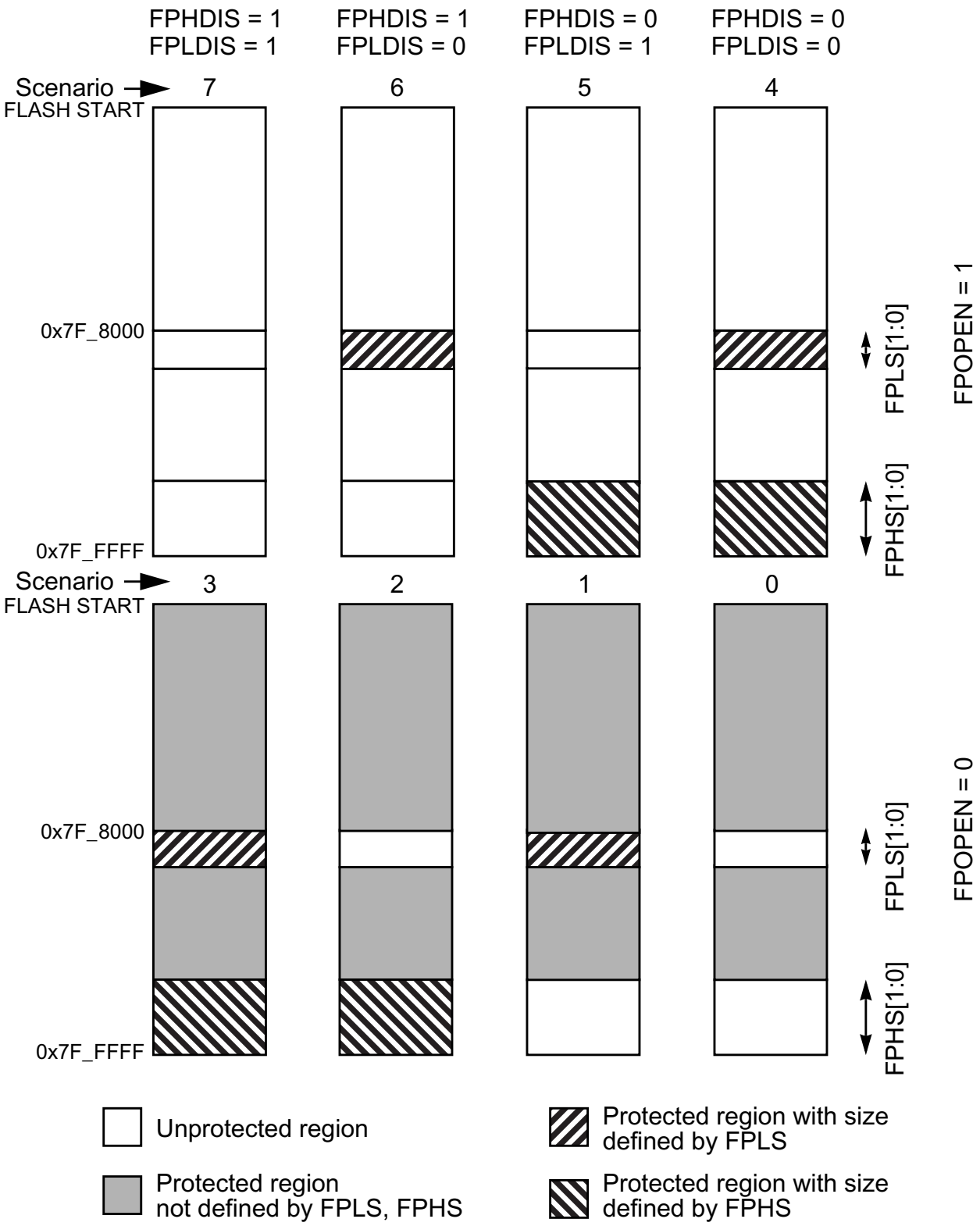

Figure 24-14. P-Flash Protection Scenarios

25.1.2.3 Emulated EEPROM Features

- Up to 4 Kbytes of emulated EEPROM (EEE) accessible as 4 Kbytes of RAM
- Flexible protection scheme to prevent accidental program or erase of data
- Automatic EEE file handling using an internal Memory Controller
- Automatic transfer of valid EEE data from D-Flash memory to buffer RAM on reset
- Ability to monitor the number of outstanding EEE related buffer RAM words left to be programmed into D-Flash memory
- Ability to disable EEE operation and allow priority access to the D-Flash memory
- Ability to cancel all pending EEE operations and allow priority access to the D-Flash memory

25.1.2.4 User Buffer RAM Features

- Up to 4 Kbytes of RAM for user access

25.1.2.5 Other Flash Module Features

- No external high-voltage power supply required for Flash memory program and erase operations
- Interrupt generation on Flash command completion and Flash error detection
- Security mechanism to prevent unauthorized access to the Flash memory

25.1.3 Block Diagram

The block diagram of the Flash module is shown in Figure 25-1.

Table 25-20. P-Flash Protection Function

FPOPEN	FPHDIS	FPLDIS	Function[1]
1	1	1	No P-Flash Protection
1	1	0	Protected Low Range
1	0	1	Protected High Range
1	0	0	Protected High and Low Ranges
0	1	1	Full P-Flash Memory Protected
0	1	0	Unprotected Low Range
0	0	1	Unprotected High Range
0	0	0	Unprotected High and Low Ranges

1. For range sizes, refer to Table 25-21 and Table 25-22.

Table 25-21. P-Flash Protection Higher Address Range

FPHS[1:0]	Global Address Range	Protected Size
00	0x7F_F800–0x7F_FFFF	2 Kbytes
01	0x7F_F000–0x7F_FFFF	4 Kbytes
10	0x7F_E000–0x7F_FFFF	8 Kbytes
11	0x7F_C000–0x7F_FFFF	16 Kbytes

Table 25-22. P-Flash Protection Lower Address Range

FPLS[1:0]	Global Address Range	Protected Size
00	0x7F_8000–0x7F_83FF	1 Kbyte
01	0x7F_8000–0x7F_87FF	2 Kbytes
10	0x7F_8000–0x7F_8FFF	4 Kbytes
11	0x7F_8000–0x7F_9FFF	8 Kbytes

All possible P-Flash protection scenarios are shown in Figure 25-14. Although the protection scheme is loaded from the Flash memory at global address 0x7F_FF0C during the reset sequence, it can be changed by the user. The P-Flash protection scheme can be used by applications requiring reprogramming in single chip mode while providing as much protection as possible if reprogramming is not required.

Offset Module Base + 0x000A

	7	6	5	4	3	2	1	0
R W	CCOB[15:8]							
Reset	0	0	0	0	0	0	0	0

Figure 26-16. Flash Common Command Object High Register (FCCOBHI)

Offset Module Base + 0x000B

	7	6	5	4	3	2	1	0
R W	CCOB[7:0]							
Reset	0	0	0	0	0	0	0	0

Figure 26-17. Flash Common Command Object Low Register (FCCOBLO)

26.3.2.11.1 FCCOB - NVM Command Mode

NVM command mode uses the indexed FCCOB register to provide a command code and its relevant parameters to the Memory Controller. The user first sets up all required FCCOB fields and then initiates the command's execution by writing a 1 to the CCIF bit in the FSTAT register (a 1 written by the user clears the CCIF command completion flag to 0). When the user clears the CCIF bit in the FSTAT register all FCCOB parameter fields are locked and cannot be changed by the user until the command completes (as evidenced by the Memory Controller returning CCIF to 1). Some commands return information to the FCCOB register array.

The generic format for the FCCOB parameter fields in NVM command mode is shown in Table 26-26. The return values are available for reading after the CCIF flag in the FSTAT register has been returned to 1 by the Memory Controller. Writes to the unimplemented parameter fields (CCOBIX = 110 and CCOBIX = 111) are ignored with reads from these fields returning 0x0000.

Table 26-26 shows the generic Flash command format. The high byte of the first word in the CCOB array contains the command code, followed by the parameters for this specific Flash command. For details on the FCCOB settings required by each command, see the Flash command descriptions in Section 26.4.2.

Table 26-26. FCCOB - NVM Command Mode (Typical Usage)

CCOBIX[2:0]	Byte	FCCOB Parameter Fields (NVM Command Mode)
000	HI	FCMD[7:0] defining Flash command
	LO	0, Global address [22:16]
001	HI	Global address [15:8]
	LO	Global address [7:0]
010	HI	Data 0 [15:8]
	LO	Data 0 [7:0]

Table 29-32. D-Flash Commands

FCMD	Command	Function on D-Flash Memory
0x08	Erase All Blocks	Erase all D-Flash (and P-Flash) blocks. An erase of all Flash blocks is only possible when the FPLDIS, FPHDIS, and FPOPEN bits in the FPROT register and the EPDIS and EPOPEN bits in the EPROT register are set prior to launching the command.
0x0B	Unsecure Flash	Supports a method of releasing MCU security by erasing all D-Flash (and P-Flash) blocks and verifying that all D-Flash (and P-Flash) blocks are erased.
0x0D	Set User Margin Level	Specifies a user margin read level for the D-Flash block.
0x0E	Set Field Margin Level	Specifies a field margin read level for the D-Flash block (special modes only).
0x0F	Full Partition D-Flash	Erase the D-Flash block and partition an area of the D-Flash block for user access.
0x10	Erase Verify D-Flash Section	Verify that a given number of words starting at the address provided are erased.
0x11	Program D-Flash	Program up to four words in the D-Flash block.
0x12	Erase D-Flash Sector	Erase all bytes in a sector of the D-Flash block.
0x13	Enable EEPROM Emulation	Enable EEPROM emulation where writes to the buffer RAM EEE partition will be copied to the D-Flash EEE partition.
0x14	Disable EEPROM Emulation	Suspend all current erase and program activity related to EEPROM emulation but leave current EEE tags set.
0x15	EEPROM Emulation Query	Returns EEE partition and status variables.
0x20	Partition D-Flash	Partition an area of the D-Flash block for user access.

29.4.2 Flash Command Description

This section provides details of all available Flash commands launched by a command write sequence. The ACCERR bit in the FSTAT register will be set during the command write sequence if any of the following illegal steps are performed, causing the command not to be processed by the Memory Controller:

- Starting any command write sequence that programs or erases Flash memory before initializing the FCLKDIV register
- Writing an invalid command as part of the command write sequence
- For additional possible errors, refer to the error handling table provided for each command

If a Flash block is read during execution of an algorithm (CCIF = 0) on that same block, the read operation will return invalid data. If the SFDIF or DFDIF flags were not previously set when the invalid read operation occurred, both the SFDIF and DFDIF flags will be set and the FECCR registers will be loaded with the global address used in the invalid read operation with the data and parity fields set to all 0.

If the ACCERR or FPVIOL bits are set in the FSTAT register, the user must clear these bits before starting any command write sequence (see Section 29.3.2.7).

CAUTION

A Flash word or phrase must be in the erased state before being programmed. Cumulative programming of bits within a Flash word or phrase is not allowed.

29.4.2.1 Erase Verify All Blocks Command

The Erase Verify All Blocks command will verify that all P-Flash and D-Flash blocks have been erased.

Table 29-33. Erase Verify All Blocks Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x01	Not required

Upon clearing CCIF to launch the Erase Verify All Blocks command, the Memory Controller will verify that the entire Flash memory space is erased. The CCIF flag will set after the Erase Verify All Blocks operation has completed.

Table 29-34. Erase Verify All Blocks Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 000 at command launch
		Set if a Load Data Field command sequence is currently active
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the read
	MGSTAT0	Set if any non-correctable errors have been encountered during the read
FERSTAT	EPVIOLIF	None

29.4.2.2 Erase Verify Block Command

The Erase Verify Block command allows the user to verify that an entire P-Flash or D-Flash block has been erased. The FCCOB upper global address bits determine which block must be verified.

Table 29-35. Erase Verify Block Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x02	Global address [22:16] of the Flash block to be verified.

Upon clearing CCIF to launch the Erase Verify Block command, the Memory Controller will verify that the selected P-Flash or D-Flash block is erased. The CCIF flag will set after the Erase Verify Block operation has completed.

0x0240–0x027F Port Integration Module (PIM) Map 5 of 6 (continued)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0278	PT0AD1	R	PT0AD17	PT0AD16	PT0AD15	PT0AD14	PT0AD13	PT0AD12	PT0AD11	PT0AD10
		W								
0x0279	PT1AD1	R	PT1AD17	PT1AD16	PT1AD15	PT1AD14	PT1AD13	PT1AD12	PT1AD11	PT1AD10
		W								
0x027A	DDR0AD1	R	DDR0AD17	DDR0AD16	DDR0AD15	DDR0AD14	DDR0AD13	DDR0AD12	DDR0AD11	DDR0AD10
		W								
0x027B	DDR1AD1	R	DDR1AD17	DDR1AD16	DDR1AD15	DDR1AD14	DDR1AD13	DDR1AD12	DDR1AD11	DDR1AD10
		W								
0x027C	RDR0AD1	R	RDR0AD17	RDR0AD16	RDR0AD15	RDR0AD14	RDR0AD13	RDR0AD12	RDR0AD11	RDR0AD10
		W								
0x027D	RDR1AD1	R	RDR1AD17	RDR1AD16	RDR1AD15	RDR1AD14	RDR1AD13	RDR1AD12	RDR1AD11	RDR1AD10
		W								
0x027E	PER0AD1	R	PER0AD17	PER0AD16	PER0AD15	PER0AD14	PER0AD13	PER0AD12	PER0AD11	PER0AD10
		W								
0x027F	PER1AD1	R	PER1AD17	PER1AD16	PER1AD15	PER1AD14	PER1AD13	PER1A1D2	PER1AD11	PER1AD10
		W								

0x0280–0x02BF MSCAN (CAN4) Map

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0280	CAN4CTL0	R	RXFRM	RXACT	CSWAI	SYNCH	TIME	WUPE	SLPRQ	INITRQ
		W								
0x0281	CAN4CTL1	R	CANE	CLKSRC	LOOPB	LISTEN	BORM	WUPM	SLPAK	INITAK
		W								
0x0282	CAN4BTR0	R	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
		W								
0x0283	CAN4BTR1	R	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
		W								
0x0284	CAN4RFLG	R	WUPIF	CSCIF	RSTAT1	RSTAT0	TSTAT1	TSTAT0	OVRIF	RXF
		W								
0x0285	CAN4RIER	R	WUPIE	CSCIE	RSTATE1	RSTATE0	TSTATE1	TSTATE0	OVRIE	RXFIE
		W								
0x0286	CAN4TFLG	R	0	0	0	0	0	TXE2	TXE1	TXE0
		W								
0x0287	CAN4TIER	R	0	0	0	0	0	TXEIE2	TXEIE1	TXEIE0
		W								
0x0288	CAN4TARQ	R	0	0	0	0	0	ABTRQ2	ABTRQ1	ABTRQ0
		W								
0x0289	CAN4TAAK	R	0	0	0	0	0	ABTAK2	ABTAK1	ABTAK0
		W								
0x028A	CAN4TBSEL	R	0	0	0	0	0	TX2	TX1	TX0
		W								
0x028B	CAN4IDAC	R	0	0	IDAM1	IDAM0	0	IDHIT2	IDHIT1	IDHIT0
		W								
0x028C	Reserved	R	0	0	0	0	0	0	0	0
		W								