



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, I ² C, IrDA, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	59
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	1.72V ~ 5.5V
Data Converters	A/D 8x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-QFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s912xet256j2caar

2.3.29 Port S Data Register (PTS)

Address 0x0248

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R	PTS7	PTST6	PTS5	PTS4	PTS3	PTS2	PTS1	PTS0
W	PTS7	PTST6	PTS5	PTS4	PTS3	PTS2	PTS1	PTS0
Altern. Function	$\overline{SS}0$	SCK0	MOSI0	MISO0	TXD1	RXD1	TXD0	RXD0
Reset	0	0	0	0	0	0	0	0

Figure 2-27. Port S Data Register (PTS)

1. Read: Anytime.
Write: Anytime.

Table 2-26. PTS Register Field Descriptions

Field	Description
7 PTS	Port S general purpose input/output data—Data Register Port S pin 7 is associated with the \overline{SS} signal of the SPI0 module. When not used with the alternative function, this pin can be used as general purpose I/O. If the associated data direction bit of this pin is set to 1, a read returns the value of the port register, otherwise the buffered pin input state is read.
6 PTS	Port S general purpose input/output data—Data Register Port S pin 6 is associated with the SCK signal of the SPI0 module. When not used with the alternative function, this pin can be used as general purpose I/O. If the associated data direction bit of this pin is set to 1, a read returns the value of the port register, otherwise the buffered pin input state is read.
5 PTS	Port S general purpose input/output data—Data Register Port S pin 5 is associated with the MOSI signal of the SPI0 module. When not used with the alternative function, this pin can be used as general purpose I/O. If the associated data direction bit of this pin is set to 1, a read returns the value of the port register, otherwise the buffered pin input state is read.
4 PTS	Port S general purpose input/output data—Data Register Port S pin 4 is associated with the MISO signal of the SPI0 module. When not used with the alternative function, this pin can be used as general purpose I/O. If the associated data direction bit of this pin is set to 1, a read returns the value of the port register, otherwise the buffered pin input state is read.
3 PTS	Port S general purpose input/output data—Data Register Port S pin 3 is associated with the TXD signal of the SCI1 module. When not used with the alternative function, this pin can be used as general purpose I/O. If the associated data direction bit of this pin is set to 1, a read returns the value of the port register, otherwise the buffered pin input state is read.
2 PTS	Port S general purpose input/output data—Data Register Port S bits 2 is associated with the RXD signal of the SCI1 module. When not used with the alternative function, this pin can be used as general purpose I/O. If the associated data direction bit of this pin is set to 1, a read returns the value of the port register, otherwise the buffered pin input state is read.

7.3.2.4 BDM Global Page Index Register (BDMGPR)

Register Global Address 0x7FFF08

	7	6	5	4	3	2	1	0
R	BGAE	BGP6	BGP5	BGP4	BGP3	BGP2	BGP1	BGP0
W								
Reset	0	0	0	0	0	0	0	0

Figure 7-6. BDM Global Page Register (BDMGPR)

Read: All modes through BDM operation when not secured

Write: All modes through BDM operation when not secured

Table 7-5. BDMGPR Field Descriptions

Field	Description
7 BGAE	BDM Global Page Access Enable Bit — BGAE enables global page access for BDM hardware and firmware read/write instructions. The BDM hardware commands used to access the BDM registers (READ_BD_ and WRITE_BD_) can not be used for global accesses even if the BGAE bit is set. 0 BDM Global Access disabled 1 BDM Global Access enabled
6–0 BGP[6:0]	BDM Global Page Index Bits 6–0 — These bits define the extended address bits from 22 to 16. For more detailed information regarding the global page window scheme, please refer to the S12X_MMC Block Guide.

7.3.3 Family ID Assignment

The family ID is a 8-bit value located in the firmware ROM (at global address: 0x7FFF0F). The read-only value is a unique family ID which is 0xC1 for S12X devices.

7.4 Functional Description

The BDM receives and executes commands from a host via a single wire serial interface. There are two types of BDM commands: hardware and firmware commands.

Hardware commands are used to read and write target system memory locations and to enter active background debug mode, see [Section 7.4.3, “BDM Hardware Commands”](#). Target system memory includes all memory that is accessible by the CPU.

Firmware commands are used to read and write CPU resources and to exit from active background debug mode, see [Section 7.4.4, “Standard BDM Firmware Commands”](#). The CPU resources referred to are the accumulator (D), X index register (X), Y index register (Y), stack pointer (SP), and program counter (PC).

Hardware commands can be executed at any time and in any mode excluding a few exceptions as highlighted (see [Section 7.4.3, “BDM Hardware Commands”](#)) and in secure mode (see [Section 7.4.1, “Security”](#)). Firmware commands can only be executed when the system is not secure and is in active background debug mode (BDM).

- **Single Stepping**
Writing a "1" to the XGSS bit will call the RISC core to execute a single instruction. All RISC core registers will be updated accordingly.
- **Write accesses to the XGCHID register and the XGCHPL register**
XGATE threads can be initiated and terminated through a 16 write access to the XGCHID and the XGCHPL register or through a 8 bit write access to the XGCHID register. Detailed operation is shown in Table 10-22. Once a thread has been initiated it's code can be either single stepped or it can be executed by leaving debug mode.

Table 10-22. Initiating and Terminating Threads in Debug Mode

Register Content		Single Cycle Write Access to...		Action
XGCHID	XGCHPL	XGCHID	XGCHPL	
0	0	1..127	_(1)	Set new XGCHID Set XGCHPL to 0x01 Initiate new thread
0	0	1..127	0..7	Set new XGCHID Set new XGCHPL Initiate new thread
1..127	0..3	1..127	4..7	Interrupt current thread Set new XGCHID Set new XGCHPL Initiate new thread
1..127	0..7	0	0..7	Terminate current thread. Resume interrupted thread or become idle if no interrupted thread is pending
			-1	
All other combinations				No action

1. 8 bit write access to XGCHID

NOTE

Even though zero is not a valid interrupt priority level of the S12X_INT module, a thread of priority level 0 can be initiated in debug mode. The XGATE handles requests of priority level 0 in the same way as it handles requests of priority levels 1 to 3.

NOTE

All channels 1 to 127 can be initiated by writing to the XGCHID register, even if they are not assigned to any peripheral module.

NOTE

In Debug Mode the XGATE will ignore all requests from peripheral modules.

10.6.1.0.1 Entering Debug Mode

Debug mode can be entered in four ways:

1. Setting XGDBG to "1"

Table 11-6. CLKSEL Field Descriptions

Field	Description
7 PLLSEL	PLL Select Bit Write: Anytime. Writing a one when LOCK=0 has no effect. This prevents the selection of an unstable PLLCLK as SYSCCLK. PLLSEL bit is cleared when the MCU enters Self Clock Mode, Stop Mode or Wait Mode with PLLWAI bit set. It is recommended to read back the PLLSEL bit to make sure PLLCLK has really been selected as SYSCCLK, as LOCK status bit could theoretically change at the very moment writing the PLLSEL bit. 0 System clocks are derived from OSCCLK ($f_{BUS} = f_{OSC} / 2$). 1 System clocks are derived from PLLCLK ($f_{BUS} = f_{PLL} / 2$).
6 PSTP	Pseudo Stop Bit Write: Anytime This bit controls the functionality of the oscillator during Stop Mode. 0 Oscillator is disabled in Stop Mode. 1 Oscillator continues to run in Stop Mode (Pseudo Stop). Note: Pseudo Stop Mode allows for faster STOP recovery and reduces the mechanical stress and aging of the resonator in case of frequent STOP conditions at the expense of a slightly increased power consumption.
5 XCLKS	Oscillator Configuration Status Bit — This read-only bit shows the oscillator configuration status. 0 Loop controlled Pierce Oscillator is selected. 1 External clock / full swing Pierce Oscillator is selected.
3 PLLWAI	PLL Stops in Wait Mode Bit Write: Anytime If PLLWAI is set, the S12XECRG will clear the PLLSEL bit before entering Wait Mode. The PLLON bit remains set during Wait Mode but the IPLL is powered down. Upon exiting Wait Mode, the PLLSEL bit has to be set manually if PLL clock is required. 0 IPLL keeps running in Wait Mode. 1 IPLL stops in Wait Mode.
1 RTIWAI	RTI Stops in Wait Mode Bit Write: Anytime 0 RTI keeps running in Wait Mode. 1 RTI stops and initializes the RTI dividers whenever the part goes into Wait Mode.
0 COPWAI	COP Stops in Wait Mode Bit Normal modes: Write once Special modes: Write anytime 0 COP keeps running in Wait Mode. 1 COP stops and initializes the COP counter whenever the part goes into Wait Mode.

11.3.2.7 S12XECRG IPLL Control Register (PLLCTL)

This register controls the IPLL functionality.

Module Base + 0x0006

	7	6	5	4	3	2	1	0
R								
W	CME	PLLON	FM1	FM0	FSTWKP	PRE	PCE	SCME
Reset	1	1	0	0	0	0	0	1

Figure 11-9. S12XECRG IPLL Control Register (PLLCTL)

Only analog input signals within the potential range of V_{RL} to V_{RH} (A/D reference potentials) will result in a non-railed digital output code.

13.4.2 Digital Sub-Block

This subsection explains some of the digital features in more detail. See [Section 13.3.2, “Register Descriptions”](#) for all details.

13.4.2.1 External Trigger Input

The external trigger feature allows the user to synchronize ATD conversions to the external environment events rather than relying on software to signal the ATD module when ATD conversions are to take place. The external trigger signal (out of reset ATD channel 15, configurable in ATDCTL1) is programmable to be edge or level sensitive with polarity control. [Table 13-23](#) gives a brief description of the different combinations of control bits and their effect on the external trigger function.

Table 13-23. External Trigger Control Bits

ETRIGLE	ETRIGP	ETRIGE	SCAN	Description
X	X	0	0	Ignores external trigger. Performs one conversion sequence and stops.
X	X	0	1	Ignores external trigger. Performs continuous conversion sequences.
0	0	1	X	Falling edge triggered. Performs one conversion sequence per trigger.
0	1	1	X	Rising edge triggered. Performs one conversion sequence per trigger.
1	0	1	X	Trigger active low. Performs continuous conversions while trigger is active.
1	1	1	X	Trigger active high. Performs continuous conversions while trigger is active.

During a conversion, if additional active edges are detected the overrun error flag ETORF is set.

In either level or edge triggered modes, the first conversion begins when the trigger is received.

Once ETRIGE is enabled, conversions cannot be started by a write to ATDCTL5, but rather must be triggered externally.

If the level mode is active and the external trigger both de-asserts and re-asserts itself during a conversion sequence, this does not constitute an overrun. Therefore, the flag is not set. If the trigger is left asserted in level mode while a sequence is completing, another sequence will be triggered immediately.

Table 16-4. CANCTL1 Register Field Descriptions (continued)

Field	Description
1 SLPAK	Sleep Mode Acknowledge — This flag indicates whether the MSCAN module has entered sleep mode (see Section 16.4.5.5, “MSCAN Sleep Mode”). It is used as a handshake flag for the SLPRQ sleep mode request. Sleep mode is active when SLPRQ = 1 and SLPAK = 1. Depending on the setting of WUPE, the MSCAN will clear the flag if it detects activity on the CAN bus while in sleep mode. 0 Running — The MSCAN operates normally 1 Sleep mode active — The MSCAN has entered sleep mode
0 INITAK	Initialization Mode Acknowledge — This flag indicates whether the MSCAN module is in initialization mode (see Section 16.4.4.5, “MSCAN Initialization Mode”). It is used as a handshake flag for the INITRQ initialization mode request. Initialization mode is active when INITRQ = 1 and INITAK = 1. The registers CANCTL1, CANBTR0, CANBTR1, CANIDAC, CANIDAR0–CANIDAR7, and CANIDMR0–CANIDMR7 can be written only by the CPU when the MSCAN is in initialization mode. 0 Running — The MSCAN operates normally 1 Initialization mode active — The MSCAN has entered initialization mode

16.3.2.3 MSCAN Bus Timing Register 0 (CANBTR0)

The CANBTR0 register configures various CAN bus timing parameters of the MSCAN module.

Module Base + 0x0002

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
W								
Reset:	0	0	0	0	0	0	0	0

Figure 16-6. MSCAN Bus Timing Register 0 (CANBTR0)

1. Read: Anytime

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

Table 16-5. CANBTR0 Register Field Descriptions

Field	Description
7-6 SJW[1:0]	Synchronization Jump Width — The synchronization jump width defines the maximum number of time quanta (Tq) clock cycles a bit can be shortened or lengthened to achieve resynchronization to data transitions on the CAN bus (see Table 16-6).
5-0 BRP[5:0]	Baud Rate Prescaler — These bits determine the time quanta (Tq) clock which is used to build up the bit timing (see Table 16-7).

Table 16-6. Synchronization Jump Width

SJW1	SJW0	Synchronization Jump Width
0	0	1 Tq clock cycle
0	1	2 Tq clock cycles
1	0	3 Tq clock cycles
1	1	4 Tq clock cycles

16.4 Functional Description

16.4.1 General

This section provides a complete functional description of the MSCAN.

16.4.2 Message Storage

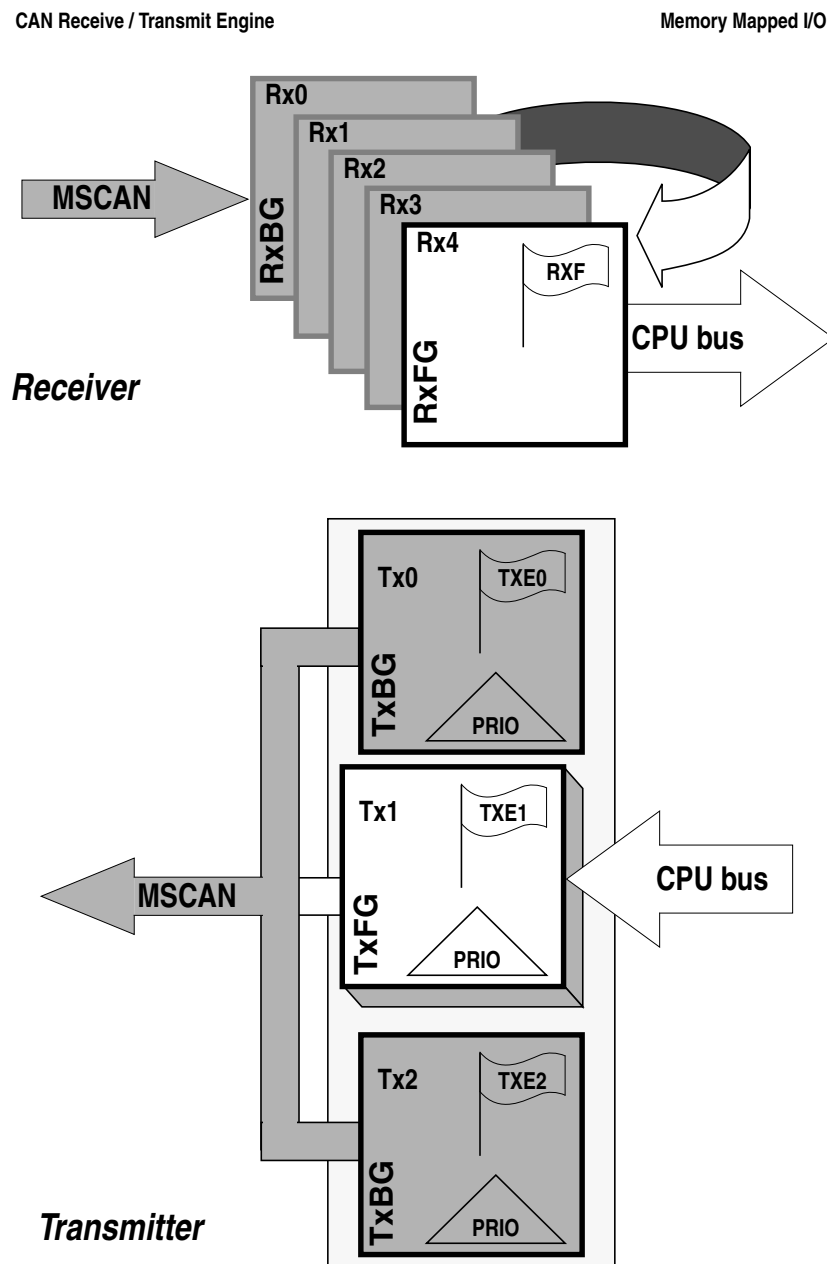


Figure 16-39. User Model for Message Buffer Organization

Chapter 18

Periodic Interrupt Timer (S12PIT24B4CV2)

Table 18-1. Revision History

Revision Number	Revision Date	Sections Affected	Description of Changes
V01.00	28 Apr 2005		- Initial Release
V01.01	05 Jul 2005	18.6/18-690	- Added application section. - Removed table 1-1

18.1 Introduction

The period interrupt timer (PIT) is an array of 24-bit timers that can be used to trigger peripheral modules or raise periodic interrupts. Refer to [Figure 18-1](#) for a simplified block diagram.

18.1.1 Glossary

Acronyms and Abbreviations	
PIT	Periodic Interrupt Timer
ISR	Interrupt Service Routine
CCR	Condition Code Register
SoC	System on Chip
micro time bases	clock periods of the 16-bit timer modulus down-counters, which are generated by the 8-bit modulus down-counters.

18.1.2 Features

The PIT includes these features:

- Four timers implemented as modulus down-counters with independent time-out periods.
- Time-out periods selectable between 1 and 2^{24} bus clock cycles. Time-out equals $m \cdot n$ bus clock cycles with $1 \leq m \leq 256$ and $1 \leq n \leq 65536$.
- Timers that can be enabled individually.
- Four time-out interrupts.
- Four time-out trigger output signals available to trigger peripheral modules.
- Start of timer channels can be aligned to each other.

18.1.3 Modes of Operation

Refer to the device overview for a detailed explanation of the chip modes.

2 registers become the high order bytes of the double byte channel. When channels 0 and 1 are concatenated, channel 0 registers become the high order bytes of the double byte channel.

See [Section 19.4.2.7, “PWM 16-Bit Functions”](#) for a more detailed description of the concatenation PWM Function.

NOTE

Change these bits only when both corresponding channels are disabled.

Table 19-9. PWMCTL Field Descriptions

Field	Description
7 CON67	Concatenate Channels 6 and 7 0 Channels 6 and 7 are separate 8-bit PWMs. 1 Channels 6 and 7 are concatenated to create one 16-bit PWM channel. Channel 6 becomes the high order byte and channel 7 becomes the low order byte. Channel 7 output pin is used as the output for this 16-bit PWM (bit 7 of port PWMP). Channel 7 clock select control-bit determines the clock source, channel 7 polarity bit determines the polarity, channel 7 enable bit enables the output and channel 7 center aligned enable bit determines the output mode.
6 CON45	Concatenate Channels 4 and 5 0 Channels 4 and 5 are separate 8-bit PWMs. 1 Channels 4 and 5 are concatenated to create one 16-bit PWM channel. Channel 4 becomes the high order byte and channel 5 becomes the low order byte. Channel 5 output pin is used as the output for this 16-bit PWM (bit 5 of port PWMP). Channel 5 clock select control-bit determines the clock source, channel 5 polarity bit determines the polarity, channel 5 enable bit enables the output and channel 5 center aligned enable bit determines the output mode.
5 CON23	Concatenate Channels 2 and 3 0 Channels 2 and 3 are separate 8-bit PWMs. 1 Channels 2 and 3 are concatenated to create one 16-bit PWM channel. Channel 2 becomes the high order byte and channel 3 becomes the low order byte. Channel 3 output pin is used as the output for this 16-bit PWM (bit 3 of port PWMP). Channel 3 clock select control-bit determines the clock source, channel 3 polarity bit determines the polarity, channel 3 enable bit enables the output and channel 3 center aligned enable bit determines the output mode.
4 CON01	Concatenate Channels 0 and 1 0 Channels 0 and 1 are separate 8-bit PWMs. 1 Channels 0 and 1 are concatenated to create one 16-bit PWM channel. Channel 0 becomes the high order byte and channel 1 becomes the low order byte. Channel 1 output pin is used as the output for this 16-bit PWM (bit 1 of port PWMP). Channel 1 clock select control-bit determines the clock source, channel 1 polarity bit determines the polarity, channel 1 enable bit enables the output and channel 1 center aligned enable bit determines the output mode.
3 PSWAI	PWM Stops in Wait Mode — Enabling this bit allows for lower power consumption in wait mode by disabling the input clock to the prescaler. 0 Allow the clock to the prescaler to continue while in wait mode. 1 Stop the input clock to the prescaler whenever the MCU is in wait mode.
2 PFRZ	PWM Counters Stop in Freeze Mode — In freeze mode, there is an option to disable the input clock to the prescaler by setting the PFRZ bit in the PWMCTL register. If this bit is set, whenever the MCU is in freeze mode, the input clock to the prescaler is disabled. This feature is useful during emulation as it allows the PWM function to be suspended. In this way, the counters of the PWM can be stopped while in freeze mode so that once normal program flow is continued, the counters are re-enabled to simulate real-time operations. Since the registers can still be accessed in this mode, to re-enable the prescaler clock, either disable the PFRZ bit or exit freeze mode. 0 Allow PWM to continue while in freeze mode. 1 Disable PWM input clock to the prescaler whenever the part is in freeze mode. This is useful for emulation.

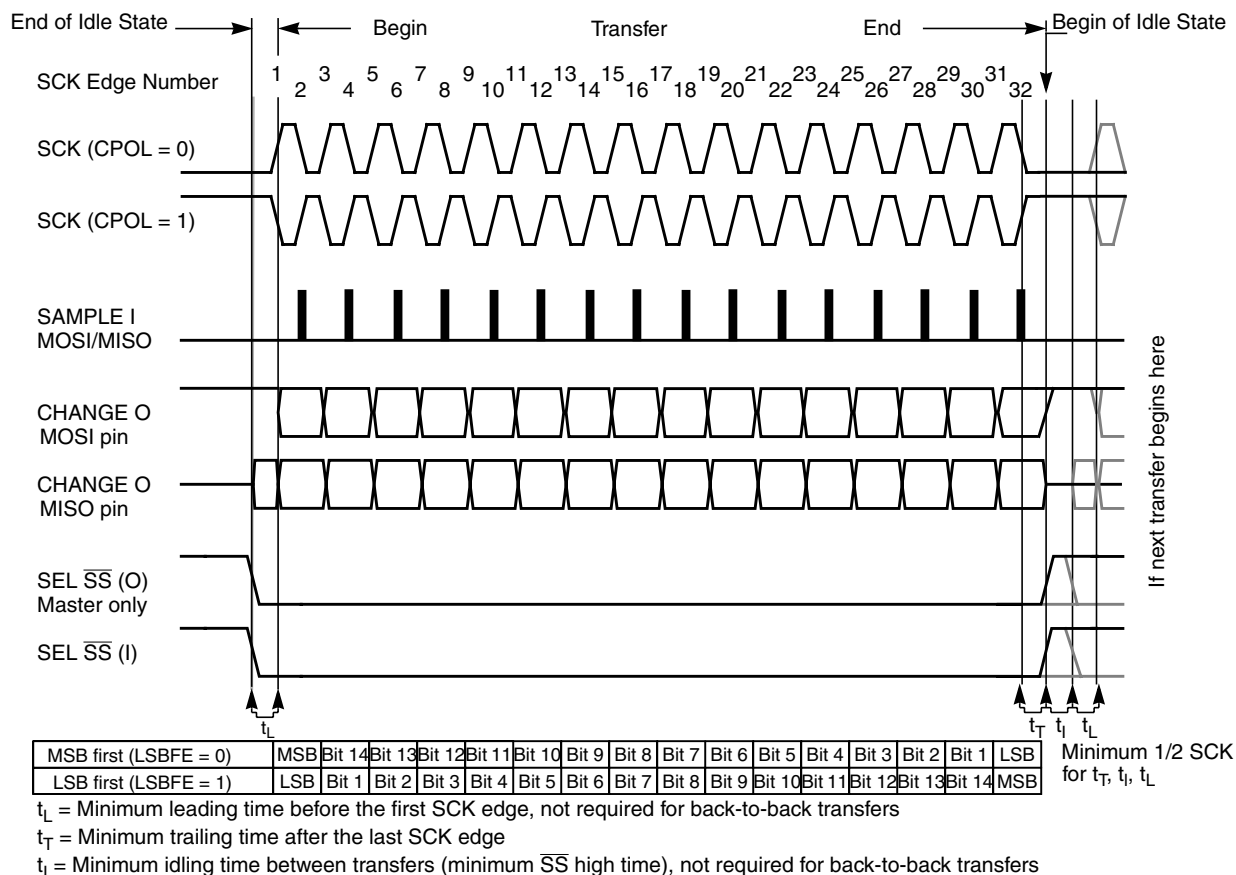


Figure 21-15. SPI Clock Format 1 (CPHA = 1), with 16-Bit Transfer Width selected (XFRW = 1)

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.

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

Table 25-16. FERCNFG Field Descriptions (continued)

Field	Description
3 ERSVIE1	EEE Error Type 1 Interrupt Enable — The ERSVIE1 bit controls interrupt generation when a change state error is detected during an EEE operation. 0 ERSVIF1 interrupt disabled 1 An interrupt will be requested whenever the ERSVIF1 flag is set (see Section 25.3.2.8)
2 ERSVIE0	EEE Error Type 0 Interrupt Enable — The ERSVIE0 bit controls interrupt generation when a sector format error is detected during an EEE operation. 0 ERSVIF0 interrupt disabled 1 An interrupt will be requested whenever the ERSVIF0 flag is set (see Section 25.3.2.8)
1 DFDIE	Double Bit Fault Detect Interrupt Enable — The DFDIE bit controls interrupt generation when a double bit fault is detected during a Flash block read operation. 0 DFDIF interrupt disabled 1 An interrupt will be requested whenever the DFDIF flag is set (see Section 25.3.2.8)
0 SFDIE	Single Bit Fault Detect Interrupt Enable — The SFDIE bit controls interrupt generation when a single bit fault is detected during a Flash block read operation. 0 SFDIF interrupt disabled whenever the SFDIF flag is set (see Section 25.3.2.8) 1 An interrupt will be requested whenever the SFDIF flag is set (see Section 25.3.2.8)

25.3.2.7 Flash Status Register (FSTAT)

The FSTAT register reports the operational status of the Flash module.

Offset Module Base + 0x0006

	7	6	5	4	3	2	1	0
R	CCIF	0	ACCERR	FPVIOL	MGBUSY	RSVD	MGSTAT[1:0]	
W								
Reset	1	0	0	0	0	0	0 ⁽¹⁾	0 ¹


 = Unimplemented or Reserved

Figure 25-11. Flash Status Register (FSTAT)

1. Reset value can deviate from the value shown if a double bit fault is detected during the reset sequence (see [Section 25.6](#)).

CCIF, ACCERR, and FPVIOL bits are readable and writable, MGBUSY and MGSTAT bits are readable but not writable, while remaining bits read 0 and are not writable.

Table 25-36. Erase Verify Block 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
		Set if an invalid global address [22:16] is supplied
	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

25.4.2.3 Erase Verify P-Flash Section Command

The Erase Verify P-Flash Section command will verify that a section of code in the P-Flash memory is erased. The Erase Verify P-Flash Section command defines the starting point of the code to be verified and the number of phrases. The section to be verified cannot cross a 128 Kbyte boundary in the P-Flash memory space.

Table 25-37. Erase Verify P-Flash Section Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x03	Global address [22:16] of a P-Flash block
001	Global address [15:0] of the first phrase to be verified	
010	Number of phrases to be verified	

Upon clearing CCIF to launch the Erase Verify P-Flash Section command, the Memory Controller will verify the selected section of Flash memory is erased. The CCIF flag will set after the Erase Verify P-Flash Section operation has completed.

Table 25-38. Erase Verify P-Flash Section Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 010 at command launch
		Set if a Load Data Field command sequence is currently active
		Set if command not available in current mode (see Table 25-30)
		Set if an invalid global address [22:0] is supplied
		Set if a misaligned phrase address is supplied (global address [2:0] != 000)
		Set if the requested section crosses a 128 Kbyte boundary
	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

25.4.2.11 Unsecure Flash Command

The Unsecure Flash command will erase the entire P-Flash and D-Flash memory space and, if the erase is successful, will release security.

Table 25-53. Unsecure Flash Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0B	Not required

Upon clearing CCIF to launch the Unsecure Flash command, the Memory Controller will erase the entire P-Flash and D-Flash memory space and verify that it is erased. If the Memory Controller verifies that the entire Flash memory space was properly erased, security will be released. If the erase verify is not successful, the Unsecure Flash operation sets MGSTAT1 and terminates without changing the security state. During the execution of this command (CCIF=0) the user must not write to any Flash module register. The CCIF flag is set after the Unsecure Flash operation has completed.

Table 25-54. Unsecure Flash 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
		Set if command not available in current mode (see Table 25-30)
	FPVIOL	Set if any area of the P-Flash memory is protected
	MGSTAT1	Set if any errors have been encountered during the verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation
FERSTAT	EPVIOLIF	Set if any area of the buffer RAM EEE partition is protected

25.4.2.12 Verify Backdoor Access Key Command

The Verify Backdoor Access Key command will only execute if it is enabled by the KEYEN bits in the FSEC register (see [Table 25-11](#)). The Verify Backdoor Access Key command releases security if user-supplied keys match those stored in the Flash security bytes of the Flash configuration field (see [Table 25-3](#)). The Verify Backdoor Access Key command must not be executed from the Flash block containing the backdoor comparison key to avoid code runaway.

Table 25-55. Verify Backdoor Access Key Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0C	Not required
001	Key 0	
010	Key 1	
011	Key 2	
100	Key 3	

Table 27-26. FCCOB - NVM Command Mode (Typical Usage)

CCOBIX[2:0]	Byte	FCCOB Parameter Fields (NVM Command Mode)
011	HI	Data 1 [15:8]
	LO	Data 1 [7:0]
100	HI	Data 2 [15:8]
	LO	Data 2 [7:0]
101	HI	Data 3 [15:8]
	LO	Data 3 [7:0]

27.3.2.12 EEE Tag Counter Register (ETAG)

The ETAG register contains the number of outstanding words in the buffer RAM EEE partition that need to be programmed into the D-Flash EEE partition. The ETAG register is decremented prior to the related tagged word being programmed into the D-Flash EEE partition. All tagged words have been programmed into the D-Flash EEE partition once all bits in the ETAG register read 0 and the MGBUSY flag in the FSTAT register reads 0.

Offset Module Base + 0x000C

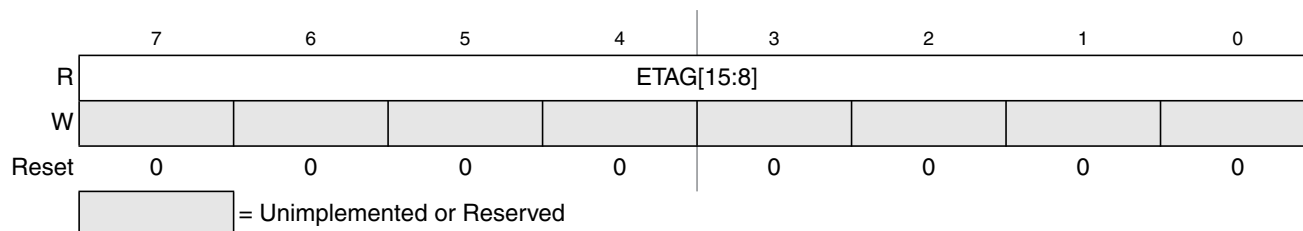


Figure 27-18. EEE Tag Counter High Register (ETAGHI)

Offset Module Base + 0x000D

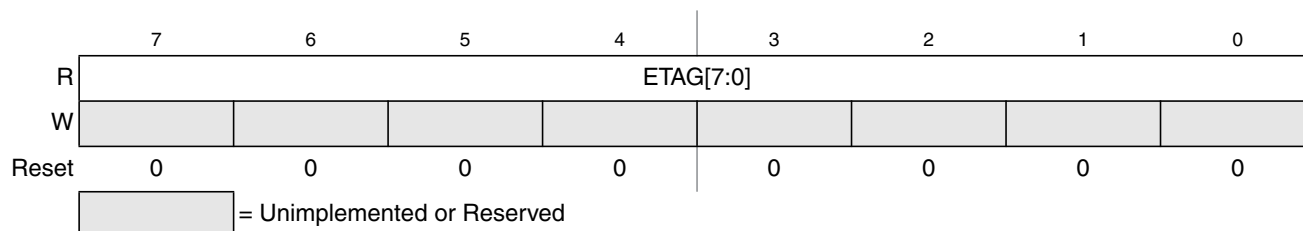


Figure 27-19. EEE Tag Counter Low Register (ETAGLO)

All ETAG bits are readable but not writable and are cleared by the Memory Controller.

27.3.2.13 Flash ECC Error Results Register (FECCR)

The FECCR registers contain the result of a detected ECC fault for both single bit and double bit faults. The FECCR register provides access to several ECC related fields as defined by the ECCRIX index bits in the FECCRIX register (see [Section 27.3.2.4](#)). Once ECC fault information has been stored, no other

Table 28-12. Flash Security States

SEC[1:0]	Status of Security
00	SECURED
01	SECURED ⁽¹⁾
10	UNSECURED
11	SECURED

1. Preferred SEC state to set MCU to secured state.

The security function in the Flash module is described in [Section 28.5](#).

28.3.2.3 Flash CCOB Index Register (FCCOBIX)

The FCCOBIX register is used to index the FCCOB register for Flash memory operations.

Offset Module Base + 0x0002

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

= Unimplemented or Reserved

Figure 28-7. FCCOB Index Register (FCCOBIX)

CCOBIX bits are readable and writable while remaining bits read 0 and are not writable.

Table 28-13. FCCOBIX Field Descriptions

Field	Description
2–0 CCOBIX[1:0]	Common Command Register Index — The CCOBIX bits are used to select which word of the FCCOB register array is being read or written to. See Section 28.3.2.11, “Flash Common Command Object Register (FCCOB),” for more details.

28.3.2.4 Flash ECCR Index Register (FECCRIX)

The FECCRIX register is used to index the FECCR register for ECC fault reporting.

Offset Module Base + 0x0003

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

= Unimplemented or Reserved

Figure 28-8. FECCR Index Register (FECCRIX)

ECCRIX bits are readable and writable while remaining bits read 0 and are not writable.

Table 28-20. 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 28-21](#) and [Table 28-22](#).

Table 28-21. 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 28-22. 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 28-14](#). 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.

Table 28-70. Erase D-Flash Sector Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 001 at command launch
		Set if a Load Data Field command sequence is currently active
		Set if command not available in current mode (see Table 28-30)
		Set if an invalid global address [22:0] is supplied
		Set if a misaligned word address is supplied (global address [0] != 0)
		Set if the global address [22:0] points to the D-Flash EEE partition
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation
FERSTAT	EPVIOLIF	None

28.4.2.19 Enable EEPROM Emulation Command

The Enable EEPROM Emulation command causes the Memory Controller to enable EEE activity. EEE activity is disabled after any reset.

Table 28-71. Enable EEPROM Emulation Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x13	Not required

Upon clearing CCIF to launch the Enable EEPROM Emulation command, the CCIF flag will set after the Memory Controller enables EEE operations using the contents of the EEE tag RAM and tag counter. The Full Partition D-Flash or the Partition D-Flash command must be run prior to launching the Enable EEPROM Emulation command.

Table 28-72. Enable EEPROM Emulation 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
		Set if Full Partition D-Flash or Partition D-Flash command not previously run
	FPVIOL	None
	MGSTAT1	None
	MGSTAT0	None
FERSTAT	EPVIOLIF	None

Table 29-7. EEE Nonvolatile Information Register Fields

Global Address (EEEIFRON)	Size (Bytes)	Description
0x12_0000 – 0x12_0001	2	D-Flash User Partition (DFPART) Refer to Section 29.4.2.15, “Full Partition D-Flash Command”
0x12_0002 – 0x12_0003	2	D-Flash User Partition (duplicate ⁽¹⁾)
0x12_0004 – 0x12_0005	2	Buffer RAM EEE Partition (ERPART) Refer to Section 29.4.2.15, “Full Partition D-Flash Command”
0x12_0006 – 0x12_0007	2	Buffer RAM EEE Partition (duplicate ⁽¹⁾)
0x12_0008 – 0x12_007F	120	Reserved

1. Duplicate value used if primary value generates a double bit fault when read during the reset sequence.

29.3.2 Register Descriptions

The Flash module contains a set of 20 control and status registers located between Flash module base + 0x0000 and 0x0013. A summary of the Flash module registers is given in [Figure 29-4](#) with detailed descriptions in the following subsections.

CAUTION

Writes to any Flash register must be avoided while a Flash command is active (CCIF=0) to prevent corruption of Flash register contents and Memory Controller behavior.

Address & Name		7	6	5	4	3	2	1	0
0x0000 FCLKDIV	R	FDIVLD	FDIV6	FDIV5	FDIV4	FDIV3	FDIV2	FDIV1	FDIV0
	W								
0x0001 FSEC	R	KEYEN1	KEYEN0	RNV5	RNV4	RNV3	RNV2	SEC1	SEC0
	W								
0x0002 FCCOBIX	R	0	0	0	0	0	CCOBIX2	CCOBIX1	CCOBIX0
	W								
0x0003 FECCRIX	R	0	0	0	0	0	ECCRIX2	ECCRIX1	ECCRIX0
	W								
0x0004 FCNFG	R	CCIE	0	0	IGNSF	0	0	FDFD	FSFD
	W								
0x0005 FERCNFG	R	ERSERIE	PGMERIE	0	EPVIOLE	ERSVIE1	ERSVIE0	DFDIE	SFDIE
	W								

Figure 29-4. FTM1024K5 Register Summary

A.1.8 Power Dissipation and Thermal Characteristics

Power dissipation and thermal characteristics are closely related. The user must assure that the maximum operating junction temperature is not exceeded. The average chip-junction temperature (T_J) in °C can be obtained from:

$$T_J = T_A + (P_D \cdot \Theta_{JA})$$

T_J = Junction Temperature, [°C]

T_A = Ambient Temperature, [°C]

P_D = Total Chip Power Dissipation, [W]

Θ_{JA} = Package Thermal Resistance, [°C/W]

The total power dissipation can be calculated from:

$$P_D = P_{INT} + P_{IO}$$

P_{INT} = Chip Internal Power Dissipation, [W]

$$P_{IO} = \sum_i R_{DS(on)} \cdot I_{IO_i}^2$$

P_{IO} is the sum of all output currents on I/O ports associated with V_{DDX} , whereby

$$R_{DS(on)} = \frac{V_{OL}}{I_{OL}}; \text{for outputs driven low}$$

$$R_{DS(on)} = \frac{V_{DD35} - V_{OH}}{I_{OH}}; \text{for outputs driven high}$$

Two cases with internal voltage regulator enabled and disabled must be considered:

1. Internal voltage regulator disabled

$$P_{INT} = I_{DD} \cdot V_{DD} + I_{DDPLL} \cdot V_{DDPLL} + I_{DDA} \cdot V_{DDA}$$

2. Internal voltage regulator enabled

$$P_{INT} = I_{DDR} \cdot V_{DDR} + I_{DDA} \cdot V_{DDA}$$

Table A-30. Example 1a: Normal Expanded Mode Timing 50 MHz bus (EWAIT disabled)

No.	Characteristic	Symbol	V _{DD5} =5.0V			V _{DD5} =3.3V			Unit
			C	Min	Max	C	Min	Max	
-	Frequency of internal bus	f _i	-	D.C.	50.0	-	D.C.	25.0	MHz
-	Internal cycle time	t _{cyc}	-	20	∞	-	40	∞	ns
-	Frequency of external bus	f _o	-	D.C.	25.0	-	D.C.	12.5	MHz
1	External cycle time (selected by EXSTR)	t _{cyce}	-	40	∞	-	80	∞	ns
2	Address ⁽¹⁾ valid to \overline{RE} fall	t _{ADRE}	D	4	-	D	13	-	ns
3	Pulse width, \overline{RE}	PW _{RE}	D	28	-	D	58	-	ns
4	Address valid to \overline{WE} fall	t _{ADWE}	D	4	-	D	15	-	ns
5	Pulse width, \overline{WE}	PW _{WE}	D	18	-	D	38	-	ns
6	Read data setup time (if ITHRS = 0)	t _{DSR}	D	19	-	D	38	-	ns
	Read data setup time (if ITHRS = 1)	t _{DSR}	D	23	-	D	N/A		ns
7	Read data hold time	t _{DHR}	D	0	-	D	0	-	ns
8	Read enable access time	t _{ACCR}	D	4	-	D	4	-	ns
9	Write data valid to \overline{WE} fall	t _{WDWE}	D	5	-	D	5	-	ns
10	Write data setup time	t _{DSW}	D	23	-	D	43	-	ns
11	Write data hold time	t _{DHW}	D	6	-	D	4	-	ns

1. Includes the following signals: ADDR_x, UDS, LDS, and CS_x.