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	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 ~ 125°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/s912xet256bmaa

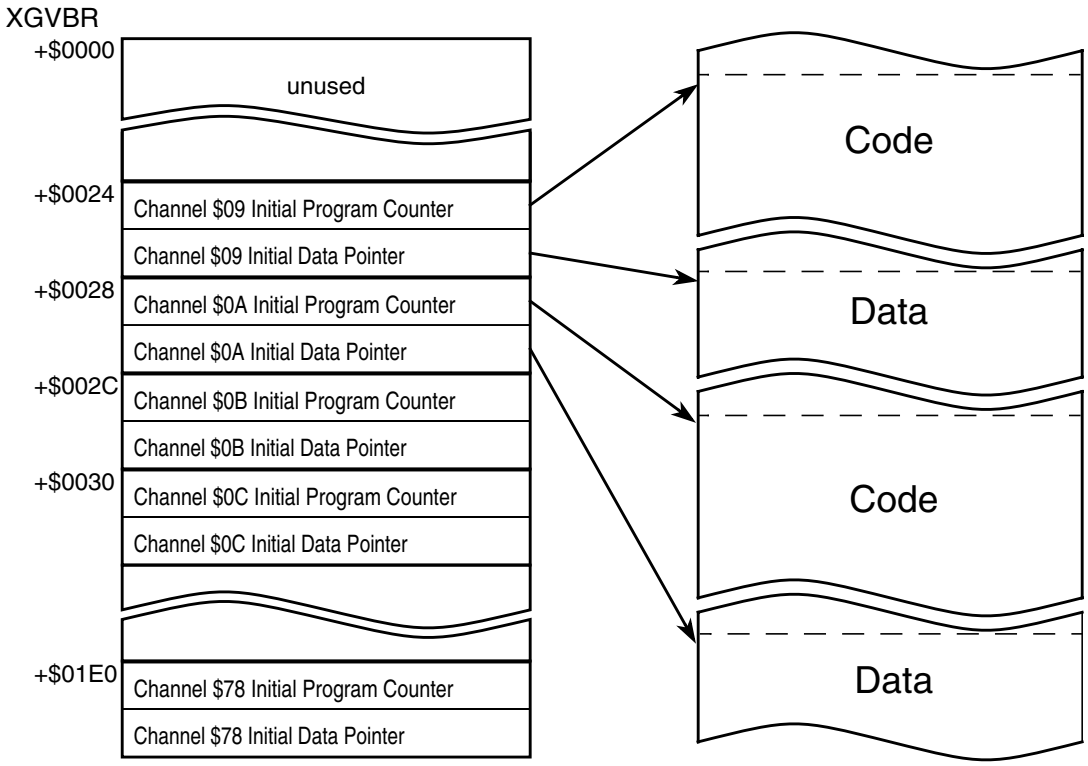

Figure 10-23. XGATE Vector Block

10.4.4 Semaphores

The XGATE module offers a set of eight hardware semaphores. These semaphores provide a mechanism to protect system resources that are shared between two concurrent threads of program execution; one thread running on the S12X_CPU and one running on the XGATE RISC core.

Each semaphore can only be in one of the three states: “Unlocked”, “Locked by S12X_CPU”, and “Locked by XGATE”. The S12X_CPU can check and change a semaphore’s state through the XGATE semaphore register (XGSEM, see Section 10.3.1.10, “XGATE Semaphore Register (XGSEM)”). The RISC core does this through its SSEM and CSEM instructions.

IFigure 10-24 illustrates the valid state transitions.

BITL — Bit Test Immediate 8 bit Constant (Low Byte) — BITL

Operation

RD.L & IMM8 ⇒ NONE

Performs a bit wise logical AND between the low byte of register RD and an immediate 8 bit constant. Only the condition code flags get updated, but no result is written back.

CCR Effects

N	Z	V	C
Δ	Δ	0	—

N: Set if bit 7 of the result is set; cleared otherwise.
Z: Set if the 8 bit result is $00; cleared otherwise.
V: 0; cleared.
C: Not affected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code							Cycles
BITL RD, #IMM8	IMM8	1	0	0	1	0	RD	IMM8	P

Table 13-3. Multi-Channel Wrap Around Coding

WRAP3	WRAP2	WRAP1	WRAP0	Multiple Channel Conversions (MULT = 1) Wraparound to AN0 after Converting
0	0	0	1	AN1
0	0	1	0	AN2
0	0	1	1	AN3
0	1	0	0	AN4
0	1	0	1	AN5
0	1	1	0	AN6
0	1	1	1	AN7
1	0	0	0	AN8
1	0	0	1	AN9
1	0	1	0	AN10
1	0	1	1	AN11
1	1	0	0	AN12
1	1	0	1	AN13
1	1	1	0	AN14
1	1	1	1	AN15

1. If only AN0 should be converted use MULT=0.

13.3.2.2 ATD Control Register 1 (ATDCTL1)

Writes to this register will abort current conversion sequence.

Module Base + 0x0001

	7	6	5	4	3	2	1	0
R W	ETRIGSEL	SRES1	SRES0	SMP_DIS	ETRIGCH3	ETRIGCH2	ETRIGCH1	ETRIGCH0
Reset	0	0	1	0	1	1	1	1

Figure 13-4. ATD Control Register 1 (ATDCTL1)

Read: Anytime

Write: Anytime

Table 13-4. ATDCTL1 Field Descriptions

Field	Description
7 ETRIGSEL	**External Trigger Source Select** — This bit selects the external trigger source to be either one of the AD channels or one of the ETRIG3-0 inputs. See device specification for availability and connectivity of ETRIG3-0 inputs. If a particular ETRIG3-0 input option is not available, writing a 1 to ETRISEL only sets the bit but has not effect, this means that one of the AD channels (selected by ETRIGCH3-0) is configured as the source for external trigger. The coding is summarized in Table 13-6.
6–5 SRES[1:0]	**A/D Resolution Select** — These bits select the resolution of A/D conversion results. See Table 13-5 for coding.

15.4.1.5 Repeated START Signal

As shown in Figure 15-10, a repeated START signal is a START signal generated without first generating a STOP signal to terminate the communication. This is used by the master to communicate with another slave or with the same slave in different mode (transmit/receive mode) without releasing the bus.

15.4.1.6 Arbitration Procedure

The Inter-IC bus is a true multi-master bus that allows more than one master to be connected on it. If two or more masters try to control the bus at the same time, a clock synchronization procedure determines the bus clock, for which the low period is equal to the longest clock low period and the high is equal to the shortest one among the masters. The relative priority of the contending masters is determined by a data arbitration procedure, a bus master loses arbitration if it transmits logic 1 while another master transmits logic 0. The losing masters immediately switch over to slave receive mode and stop driving SDA output. In this case the transition from master to slave mode does not generate a STOP condition. Meanwhile, a status bit is set by hardware to indicate loss of arbitration.

15.4.1.7 Clock Synchronization

Because wire-AND logic is performed on SCL line, a high-to-low transition on SCL line affects all the devices connected on the bus. The devices start counting their low period and as soon as a device's clock has gone low, it holds the SCL line low until the clock high state is reached.However, the change of low to high in this device clock may not change the state of the SCL line if another device clock is within its low period. Therefore, synchronized clock SCL is held low by the device with the longest low period. Devices with shorter low periods enter a high wait state during this time (see Figure 15-11). When all devices concerned have counted off their low period, the synchronized clock SCL line is released and pulled high. There is then no difference between the device clocks and the state of the SCL line and all the devices start counting their high periods.The first device to complete its high period pulls the SCL line low again.

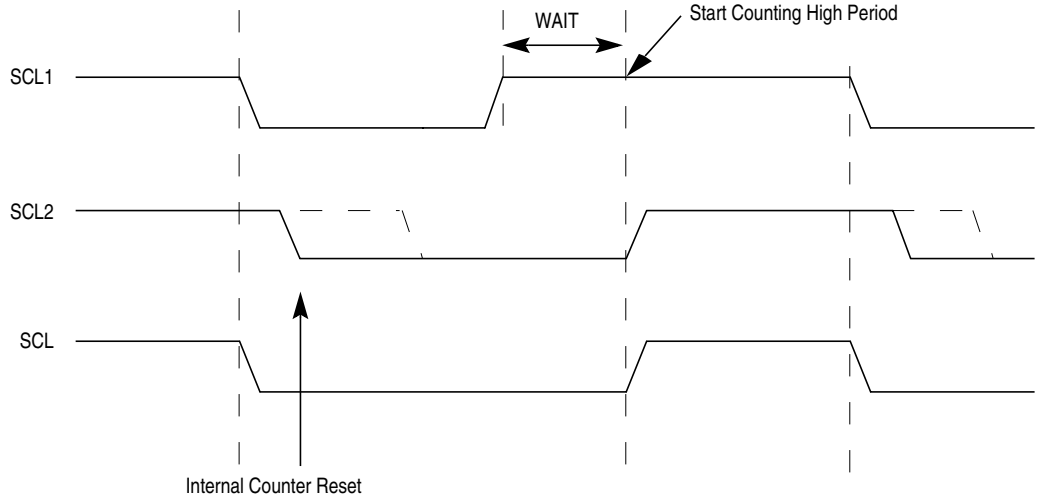

Figure 15-12. IIC-Bus Clock Synchronization

software simpler because only one address area is applicable for the transmit process, and the required address space is minimized.

The CPU then stores the identifier, the control bits, and the data content into one of the transmit buffers. Finally, the buffer is flagged as ready for transmission by clearing the associated TXE flag.

The MSCAN then schedules the message for transmission and signals the successful transmission of the buffer by setting the associated TXE flag. A transmit interrupt (see Section 16.4.7.2, "Transmit Interrupt") is generated[1] when TXEx is set and can be used to drive the application software to re-load the buffer.

If more than one buffer is scheduled for transmission when the CAN bus becomes available for arbitration, the MSCAN uses the local priority setting of the three buffers to determine the prioritization. For this purpose, every transmit buffer has an 8-bit local priority field (PRIO). The application software programs this field when the message is set up. The local priority reflects the priority of this particular message relative to the set of messages being transmitted from this node. The lowest binary value of the PRIO field is defined to be the highest priority. The internal scheduling process takes place whenever the MSCAN arbitrates for the CAN bus. This is also the case after the occurrence of a transmission error.

When a high priority message is scheduled by the application software, it may become necessary to abort a lower priority message in one of the three transmit buffers. Because messages that are already in transmission cannot be aborted, the user must request the abort by setting the corresponding abort request bit (ABTRQ) (see Section 16.3.2.9, "MSCAN Transmitter Message Abort Request Register (CANTARQ)".) The MSCAN then grants the request, if possible, by:

1. Setting the corresponding abort acknowledge flag (ABTAK) in the CANTAAK register.
2. Setting the associated TXE flag to release the buffer.
3. Generating a transmit interrupt. The transmit interrupt handler software can determine from the setting of the ABTAK flag whether the message was aborted (ABTAK = 1) or sent (ABTAK = 0).

16.4.2.3 Receive Structures

The received messages are stored in a five stage input FIFO. The five message buffers are alternately mapped into a single memory area (see Figure 16-39). The background receive buffer (RxBG) is exclusively associated with the MSCAN, but the foreground receive buffer (RxFG) is addressable by the CPU (see Figure 16-39). This scheme simplifies the handler software because only one address area is applicable for the receive process.

All receive buffers have a size of 15 bytes to store the CAN control bits, the identifier (standard or extended), the data contents, and a time stamp, if enabled (see Section 16.3.3, "Programmer's Model of Message Storage").

The receiver full flag (RXF) (see Section 16.3.2.5, "MSCAN Receiver Flag Register (CANRFLG)") signals the status of the foreground receive buffer. When the buffer contains a correctly received message with a matching identifier, this flag is set.

On reception, each message is checked to see whether it passes the filter (see Section 16.4.3, "Identifier Acceptance Filter") and simultaneously is written into the active RxBG. After successful reception of a valid message, the MSCAN shifts the content of RxBG into the receiver FIFO, sets the RXF flag, and

1. The transmit interrupt occurs only if not masked. A polling scheme can be applied on TXEx also.

If the WUPE bit in CANCTL0 is not asserted, the MSCAN will mask any activity it detects on CAN. RXCAN is therefore held internally in a recessive state. This locks the MSCAN in sleep mode. WUPE must be set before entering sleep mode to take effect.

The MSCAN is able to leave sleep mode (wake up) only when:

- CAN bus activity occurs and WUPE = 1

  or
- the CPU clears the SLPRQ bit

NOTE

The CPU cannot clear the SLPRQ bit before sleep mode (SLPRQ = 1 and SLPAK = 1) is active.

After wake-up, the MSCAN waits for 11 consecutive recessive bits to synchronize to the CAN bus. As a consequence, if the MSCAN is woken-up by a CAN frame, this frame is not received.

The receive message buffers (RxFG and RxBG) contain messages if they were received before sleep mode was entered. All pending actions will be executed upon wake-up; copying of RxBG into RxFG, message aborts and message transmissions. If the MSCAN remains in bus-off state after sleep mode was exited, it continues counting the 128 occurrences of 11 consecutive recessive bits.

16.4.5.6 MSCAN Power Down Mode

The MSCAN is in power down mode (Table 16-38) when

- CPU is in stop mode

  or
- CPU is in wait mode and the CSWAI bit is set

When entering the power down mode, the MSCAN immediately stops all ongoing transmissions and receptions, potentially causing CAN protocol violations. To protect the CAN bus system from fatal consequences of violations to the above rule, the MSCAN immediately drives TXCAN into a recessive state.

NOTE

The user is responsible for ensuring that the MSCAN is not active when power down mode is entered. The recommended procedure is to bring the MSCAN into Sleep mode before the STOP or WAI instruction (if CSWAI is set) is executed. Otherwise, the abort of an ongoing message can cause an error condition and impact other CAN bus devices.

In power down mode, all clocks are stopped and no registers can be accessed. If the MSCAN was not in sleep mode before power down mode became active, the module performs an internal recovery cycle after powering up. This causes some fixed delay before the module enters normal mode again.

17.3 Register Definition

This section consists of register descriptions in address order of the PIT. Each description includes a standard register diagram with an associated figure number. Details of register bit and field function follow the register diagrams, in bit order.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 PITCFLMT	R	PITE	PITSWAI	PITFRZ	0	0	0	0	0
	W							PFLMT1	PFLMT0
0x0001 PITFLT	R	0	0	0	0	0	0	0	0
	W	PFLT7	PFLT6	PFLT5	PFLT4	PFLT3	PFLT2	PFLT1	PFLT0
0x0002 PITCE	R	PCE7	PCE6	PCE5	PCE4	PCE3	PCE2	PCE1	PCE0
	W								
0x0003 PITMUX	R	PMUX7	PMUX6	PMUX5	PMUX4	PMUX3	PMUX2	PMUX1	PMUX0
	W								
0x0004 PITINTE	R	PINTE7	PINTE6	PINTE5	PINTE4	PINTE3	PINTE2	PINTE1	PINTE0
	W								
0x0005 PITTF	R	PTF7	PTF6	PTF5	PTF4	PTF3	PTF2	PTF1	PTF0
	W								
0x0006 PITMTLD0	R	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
	W								
0x0007 PITMTLD1	R	PMTLD7	PMTLD6	PMTLD5	PMTLD4	PMTLD3	PMTLD2	PMTLD1	PMTLD0
	W								
0x0008 PITLD0 (High)	R	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
	W								
0x0009 PITLD0 (Low)	R	PLD7	PLD6	PLD5	PLD4	PLD3	PLD2	PLD1	PLD0
	W								
0x000A PITCNT0 (High)	R	PCNT15	PCNT14	PCNT13	PCNT12	PCNT11	PCNT10	PCNT9	PCNT8
	W								
0x000B PITCNT0 (Low)	R	PCNT7	PCNT6	PCNT5	PCNT4	PCNT3	PCNT2	PCNT1	PCNT0
	W								
0x000C PITLD1 (High)	R	PLD15	PLD14	PLD13	PLD12	PLD11	PLD10	PLD9	PLD8
	W								

= Unimplemented or Reserved

Figure 17-2. PIT Register Summary (Sheet 1 of 3)

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
0x001A PWMPER6	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001B PWMPER7	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001C PWMDTY0	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001D PWMDTY1	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001E PWMDTY2	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001F PWMDTY3	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0020 PWMDTY4	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0021 PWMDTY5	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0022 PWMDTY6	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0023 PWMDTY7	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0024 PWMSDN	R W	PWMIF	PWMIE	0 PWMRSTRT	PWMLVL	0	PWM7IN	PWM7INL	PWM7ENA

= Unimplemented or Reserved

Figure 19-2. PWM Register Summary (Sheet 3 of 3)

1. Intended for factory test purposes only.

19.3.2.1 PWM Enable Register (PWME)

Each PWM channel has an enable bit (PWMEx) to start its waveform output. When any of the PWMEx bits are set (PWMEx = 1), the associated PWM output is enabled immediately. However, the actual PWM waveform is not available on the associated PWM output until its clock source begins its next cycle due to the synchronization of PWMEx and the clock source.

NOTE

The first PWM cycle after enabling the channel can be irregular.

Table 24-62. Full Partition D-Flash Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 010 at command launch
		Set if command not available in current mode (see Table 24-30)
		Set if an invalid DFPART or ERPART selection is supplied(1)
	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

1. As defined by the maximum ERPART for FTM256K2.

24.4.2.15 Erase Verify D-Flash Section Command

The Erase Verify D-Flash Section command will verify that a section of code in the D-Flash user partition is erased. The Erase Verify D-Flash Section command defines the starting point of the data to be verified and the number of words.

Table 24-63. Erase Verify D-Flash Section Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x10	Global address [22:16] to identify the D-Flash block
001	Global address [15:0] of the first word to be verified	
010	Number of words to be verified	

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

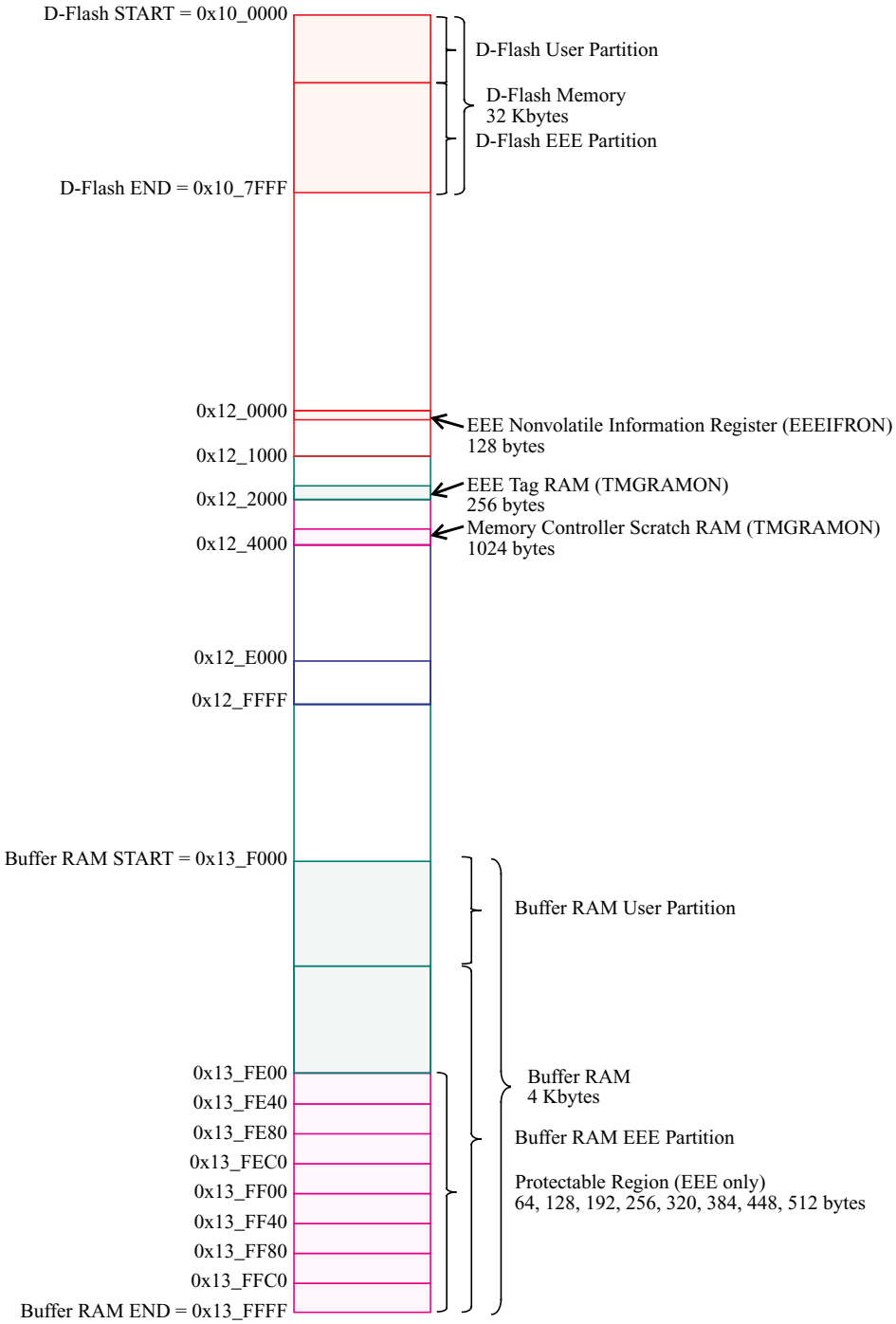

Figure 25-3. EEE Resource Memory Map

The Full Partition D-Flash command (see Section 25.4.2.15) is used to program the EEE nonvolatile information register fields where address 0x12_0000 defines the D-Flash partition for user access and address 0x12_0004 defines the buffer RAM partition for EEE operations.

Table 26-36. 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(1)
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the read(2)
	MGSTAT0	Set if any non-correctable errors have been encountered during the read2
FERSTAT	EPVIOLIF	None

1. As defined by the memory map for FTM512K3.
2. As found in the memory map for FTM512K3.

26.4.2.3 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 256 Kbyte boundary in the P-Flash memory space.

Table 26-37. 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 27-20. 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 27-21 and Table 27-22.

Table 27-21. 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 27-22. 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 27-14. 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 27-47. Erase All Blocks Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x08	Not required

Upon clearing CCIF to launch the Erase All Blocks command, the Memory Controller will erase the entire 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. During the execution of this command (CCIF=0) the user must not write to any Flash module register. The CCIF flag will set after the Erase All Blocks operation has completed.

Table 27-48. Erase 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
		Set if command not available in current mode (see Table 27-30)
	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

27.4.2.9 Erase P-Flash Block Command

The Erase P-Flash Block operation will erase all addresses in a P-Flash block.

Table 27-49. Erase P-Flash Block Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x09	Global address [22:16] to identify P-Flash block
001	Global address [15:0] in P-Flash block to be erased	

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

Table 27-63. Full Partition D-Flash Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0F	Not required
001	Number of 256 byte sectors for the D-Flash user partition (DFPART)	
010	Number of 256 byte sectors for buffer RAM EEE partition (ERPART)	

Upon clearing CCIF to launch the Full Partition D-Flash command, the following actions are taken to define a partition within the D-Flash block for direct access (DFPART) and a partition within the buffer RAM for EEE use (ERPART):

- Validate the DFPART and ERPART values provided:
  - DFPART <= 128 (maximum number of 256 byte sectors in D-Flash block)
  - ERPART <= 16 (maximum number of 256 byte sectors in buffer RAM)
  - If ERPART > 0, 128 - DFPART >= 12 (minimum number of 256 byte sectors in the D-Flash block required to support EEE)
  - If ERPART > 0, ((128-DFPART)/ERPART) >= 8 (minimum ratio of D-Flash EEE space to buffer RAM EEE space to support EEE)
- Erase the D-Flash block and the EEE nonvolatile information register
- Program DFPART to the EEE nonvolatile information register at global address 0x12_0000 (see Table 27-7)
- Program a duplicate DFPART to the EEE nonvolatile information register at global address 0x12_0002 (see Table 27-7)
- Program ERPART to the EEE nonvolatile information register at global address 0x12_0004 (see Table 27-7)
- Program a duplicate ERPART to the EEE nonvolatile information register at global address 0x12_0006 (see Table 27-7)

The D-Flash user partition will start at global address 0x10_0000. The buffer RAM EEE partition will end at global address 0x13_FFFF. After the Full Partition D-Flash operation has completed, the CCIF flag will set.

Running the Full Partition D-Flash command a second time will result in the previous partition values and the entire D-Flash memory being erased. The data value written corresponds to the number of 256 byte sectors allocated for either direct D-Flash access (DFPART) or buffer RAM EEE access (ERPART).

29.3 Memory Map and Registers

This section describes the memory map and registers for the Flash module. Read data from unimplemented memory space in the Flash module is undefined. Write access to unimplemented or reserved memory space in the Flash module will be ignored by the Flash module.

29.3.1 Module Memory Map

The S12X architecture places the P-Flash memory between global addresses 0x70_0000 and 0x7F_FFFF as shown in Table 29-2. The P-Flash memory map is shown in Figure 29-2.

Table 29-2. P-Flash Memory Addressing

Global Address	Size (Bytes)	Description
0x7C_0000 – 0x7F_FFFF	256 K	P-Flash Block 0 Contains Flash Configuration Field (see Table 29-3)
0x7A_0000 – 0x7B_FFFF	128 K	P-Flash Block 1N
0x78_0000 – 0x79_FFFF	128 K	P-Flash Block 1S
0x74_0000 – 0x77_FFFF	256 K	P-Flash Block 2
0x70_0000 – 0x73_FFFF	256 K	P-Flash Block 3

The FPROT register, described in Section 29.3.2.9, can be set to protect regions in the Flash memory from accidental program or erase. Three separate memory regions, one growing upward from global address 0x7F_8000 in the Flash memory (called the lower region), one growing downward from global address 0x7F_FFFF in the Flash memory (called the higher region), and the remaining addresses in the Flash memory, can be activated for protection. The Flash memory addresses covered by these protectable regions are shown in the P-Flash memory map. The higher address region is mainly targeted to hold the boot loader code since it covers the vector space. Default protection settings as well as security information that allows the MCU to restrict access to the Flash module are stored in the Flash configuration field as described in Table 29-3.

Table 29-3. Flash Configuration Field(1)

Global Address	Size (Bytes)	Description
0x7F_FF00 – 0x7F_FF07	8	Backdoor Comparison Key Refer to Section 29.4.2.12, “Verify Backdoor Access Key Command,” and Section 29.5.1, “Unsecuring the MCU using Backdoor Key Access”
0x7F_FF08 – 0x7F_FF0B(2)	4	Reserved
0x7F_FF0C[2]	1	P-Flash Protection byte. Refer to Section 29.3.2.9, “P-Flash Protection Register (FPROT)”
0x7F_FF0D[2]	1	EEE Protection byte Refer to Section 29.3.2.10, “EEE Protection Register (EPROT)”

Table 29-12. 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 29.5.

29.3.2.3 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 29-7. FCCOB Index Register (FCCOBIX)

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

Table 29-13. 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 29.3.2.11, “Flash Common Command Object Register (FCCOB),” for more details.

29.3.2.4 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 29-8. FECCR Index Register (FECCRIX)

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

Table 29-70. 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 29-30)
		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

29.4.2.19 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 29-71. 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 29-72. 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

0x0080–0x00AF Analog-to-Digital Converter 12-bit 16-Channels (ATD1) Map (Sheet 2 of 3)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0096	ATD1DR3H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x0097	ATD1DR3L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x0098	ATD1DR4H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x0099	ATD1DR4L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x009A	ATD1DR5H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x009B	ATD1DR5L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x009C	ATD1DR6H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x009D	ATD1DR6L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x009E	ATD1DR7H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x009F	ATD1DR7L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x00A0	ATD1DR8H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x00A1	ATD1DR8L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x00A2	ATD1DR9H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x00A3	ATD1DR9L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x00A4	ATD1DR10H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x00A5	ATD1DR10L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x00A6	ATD1DR11H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x00A7	ATD1DR11L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x00A8	ATD1DR12H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x00A9	ATD1DR12L	R	Bit7	Bit6	0	0	0	0	0	0
		W								
0x00AA	ATD1DR13H	R	Bit15	14	13	12	11	10	9	Bit8
		W								
0x00AB	ATD1DR13L	R	Bit7	Bit6	0	0	0	0	0	0
		W								