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	119
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 24x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12xet256mag

The state of the ROMCTL signal is latched into the ROMON bit in the MMCCTL1 register on the rising edge of $\overline{\text{RESET}}$. The state of the EROMCTL signal is latched into the EROMON bit in the MMCCTL1 register on the rising edge of $\overline{\text{RESET}}$.

Table 1-12. Chip Modes and Data Sources

Chip Modes	MODC	MODB	MODA	ROMCTL	EROMCTL	Data Source[1]
Normal single chip	1	0	0	X	X	Internal
Special single chip	0	0	0			
Emulation single chip	0	0	1	X	0	Emulation memory
				X	1	Internal Flash
Normal expanded	1	0	1	0	X	External application
				1	X	Internal Flash
Emulation expanded	0	1	1	0	X	External application
				1	0	Emulation memory
				1	1	Internal Flash
Special test	0	1	0	0	X	External application
				1	X	Internal Flash

1. Internal means resources inside the MCU are read/written.
Internal Flash means Flash resources inside the MCU are read/written.
Emulation memory means resources inside the emulator are read/written (PRU registers, Flash replacement, RAM, EEPROM, and register space are always considered internal).
External application means resources residing outside the MCU are read/written.

1.4.1.1 Normal Expanded Mode

Ports K, A, and B are configured as a 23-bit address bus, ports C and D are configured as a 16-bit data bus, and port E provides bus control and status signals. This mode allows 16-bit external memory and peripheral devices to be interfaced to the system. The fastest external bus rate is divide by 2 from the internal bus rate.

1.4.1.2 Normal Single-Chip Mode

There is no external bus in this mode. The processor program is executed from internal memory. Ports A, B,C,D, K, and most pins of port E are available as general-purpose I/O.

1.4.1.3 Special Single-Chip Mode

This mode is used for debugging single-chip operation, boot-strapping, or security related operations. The background debug module BDM is active in this mode. The CPU executes a monitor program located in an on-chip ROM. BDM firmware waits for additional serial commands through the BKGD pin. There is no external bus after reset in this mode.

1.4.1.4 Emulation of Expanded Mode

Developers use this mode for emulation systems in which the users target application is normal expanded mode. Code is executed from external memory or from internal memory depending on the state of ROMON and EROMON bit. In this mode the internal operation is visible on external bus interface.

2.3.71 Port AD0 Data Direction Register 0 (DDR0AD0)

Address 0x0272 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	DDR0AD07	DDR0AD06	DDR0AD05	DDR0AD04	DDR0AD03	DDR0AD02	DDR0AD01	DDR0AD00
Reset	0	0	0	0	0	0	0	0

Figure 2-69. Port AD0 Data Direction Register 0 (DDR0AD0)

1. Read: Anytime.
   Write: Anytime.

Table 2-67. DDR0AD0 Register Field Descriptions

Field	Description
7-0 DDR0AD0	**Port AD0 data direction**— This register controls the data direction of pins 15 through 8. 1 Associated pin is configured as output. 0 Associated pin is configured as input.

NOTE

Due to internal synchronization circuits, it can take up to 2 bus clock cycles until the correct value is read on PT0AD0 registers, when changing the DDR0AD0 register.

NOTE

To use the digital input function on Port AD0 the ATD Digital Input Enable Register (ATD0DIEN1) has to be set to logic level “1”.

2.3.72 Port AD0 Data Direction Register 1 (DDR1AD0)

Address 0x0273 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	DDR1AD07	DDR1AD06	DDR1AD05	DDR1AD04	DDR1AD03	DDR1AD02	DDR1AD01	DDR1AD00
Reset	0	0	0	0	0	0	0	0

Figure 2-70. Port AD0 Data Direction Register 1 (DDR1AD0)

1. Read: Anytime.
   Write: Anytime.

BVS Branch if Overflow Set BVS

Operation

If V = 1, then PC + $0002 + (REL9 << 1) ⇒ PC

Tests the Overflow flag and branches if V = 1.

CCR Effects

N	Z	V	C
—	—	—	—

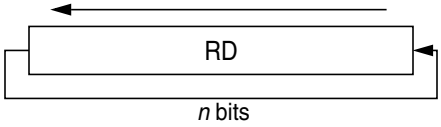
N: Not affected.
Z: Not affected.
V: Not affected.
C: Not affected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code								Cycles
BVS REL9	REL9	0	0	1	0	1	1	1	REL9	PP/P

ROL Rotate Left ROL

Operation



n = RS or IMM4

Rotates the bits in register RD *n* positions to the left. The lower *n* bits of the register RD are filled with the upper *n* bits. Two source forms are available. In the first form, the parameter *n* is contained in the instruction code as an immediate operand. In the second form, the parameter is contained in the lower bits of the source register RS[3:0]. All other bits in RS are ignored. If *n* is zero, no shift will take place and the register RD will be unaffected; however, the condition code flags will be updated.

CCR Effects

N	Z	V	C
Δ	Δ	0	—

N: Set if bit 15 of the result is set; cleared otherwise.
Z: Set if the result is $0000; cleared otherwise.
V: 0; cleared.
C: Not affected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code											Cycles
ROL RD, #IMM4	IMM4	0	0	0	0	1	RD	IMM4		1	1	1	0	P
ROL RD, RS	DYA	0	0	0	0	1	RD	RS	1	0	1	1	0	P

SBC — Subtract with Carry — SBC

Operation

RS1 - RS2 - C ⇒ RD

Subtracts the content of register RS2 and the value of the Carry bit from the content of register RS1 using binary subtraction and stores the result in the destination register RD. Also the zero flag is carried forward from the previous operation allowing 32 and more bit subtractions.

Example:

```
SUB     R6,R4,R2
SBC     R7,R5,R3          ; R7:R6 = R5:R4 - R3:R2
BCC                       ; conditional branch on 32 bit subtraction
```

CCR Effects

N	Z	V	C
Δ	Δ	Δ	Δ

N: Set if bit 15 of the result is set; cleared otherwise.

Z: Set if the result is $0000 and Z was set before this operation; cleared otherwise.

V: Set if a two´s complement overflow resulted from the operation; cleared otherwise.
$RS1[15] \& \overline{RS2[15]} \& \overline{RD[15]}_{new} \mid \overline{RS1[15]} \& RS2[15] \& RD[15]_{new}$

C: Set if there is a carry from bit 15 of the result; cleared otherwise.
$\overline{RS1[15]} \& RS2[15] \mid \overline{RS1[15]} \& RD[15]_{new} \mid RS2[15] \& RD[15]_{new}$

Code and CPU Cycles

Source Form	Address Mode	Machine Code										Cycles
SBC RD, RS1, RS2	TRI	0	0	0	1	1	RD	RS1	RS2	0	1	P

14.4.1 Enhanced Capture Timer Modes of Operation

The enhanced capture timer has 8 input capture, output compare (IC/OC) channels, same as on the HC12 standard timer (timer channels TC0 to TC7). When channels are selected as input capture by selecting the IOSx bit in TIOS register, they are called input capture (IC) channels.

Four IC channels (channels 7–4) are the same as on the standard timer with one capture register each that memorizes the timer value captured by an action on the associated input pin.

Four other IC channels (channels 3–0), in addition to the capture register, also have one buffer each called a holding register. This allows two different timer values to be saved without generating any interrupts.

Four 8-bit pulse accumulators are associated with the four buffered IC channels (channels 3–0). Each pulse accumulator has a holding register to memorize their value by an action on its external input. Each pair of pulse accumulators can be used as a 16-bit pulse accumulator.

The 16-bit modulus down-counter can control the transfer of the IC registers and the pulse accumulators contents to the respective holding registers for a given period, every time the count reaches zero.

The modulus down-counter can also be used as a stand-alone time base with periodic interrupt capability.

14.4.1.1 IC Channels

The IC channels are composed of four standard IC registers and four buffered IC channels.

- An IC register is empty when it has been read or latched into the holding register.
- A holding register is empty when it has been read.

14.4.1.1.1 Non-Buffered IC Channels

The main timer value is memorized in the IC register by a valid input pin transition. If the corresponding NOVWx bit of the ICOVW register is cleared, with a new occurrence of a capture, the contents of IC register are overwritten by the new value. If the corresponding NOVWx bit of the ICOVW register is set, the capture register cannot be written unless it is empty. This will prevent the captured value from being overwritten until it is read.

14.4.1.1.2 Buffered IC Channels

There are two modes of operations for the buffered IC channels:

1. IC latch mode (LATQ = 1)

   The main timer value is memorized in the IC register by a valid input pin transition (see Figure 14-67 and Figure 14-68).

   The value of the buffered IC register is latched to its holding register by the modulus counter for a given period when the count reaches zero, by a write 0x0000 to the modulus counter or by a write to ICLAT in the MCCTL register.

   If the corresponding NOVWx bit of the ICOVW register is cleared, with a new occurrence of a capture, the contents of IC register are overwritten by the new value. In case of latching, the contents of its holding register are overwritten.

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.

17.4.1 Timer

As shown in Figure 17-1 and Figure 17-27, the 24-bit timers are built in a two-stage architecture with eight 16-bit modulus down-counters and two 8-bit modulus down-counters. The 16-bit timers are clocked with two selectable micro time bases which are generated with 8-bit modulus down-counters. Each 16-bit timer is connected to micro time base 0 or 1 via the PMUX[7:0] bit setting in the PIT Multiplex (PITMUX) register.

A timer channel is enabled if the module enable bit PITE in the PIT control and force load micro timer (PITCFLMT) register is set and if the corresponding PCE bit in the PIT channel enable (PITCE) register is set. Two 8-bit modulus down-counters are used to generate two micro time bases. As soon as a micro time base is selected for an enabled timer channel, the corresponding micro timer modulus down-counter will load its start value as specified in the PITMTLD0 or PITMTLD1 register and will start down-counting. Whenever the micro timer down-counter has counted to zero the PITMTLD register is reloaded and the connected 16-bit modulus down-counters count one cycle.

Whenever a 16-bit timer counter and the connected 8-bit micro timer counter have counted to zero, the PITLD register is reloaded and the corresponding time-out flag PTF in the PIT time-out flag (PITTF) register is set, as shown in Figure 17-28. The time-out period is a function of the timer load (PITLD) and micro timer load (PITMTLD) registers and the bus clock f_{BUS}:

time-out period = (PITMTLD + 1) * (PITLD + 1) / f_{BUS}.

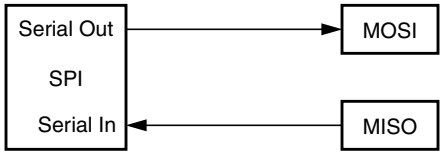
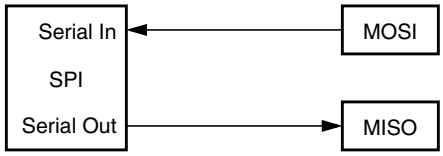
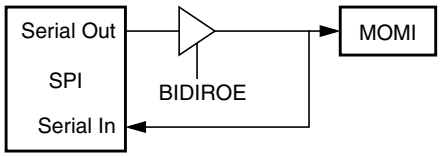
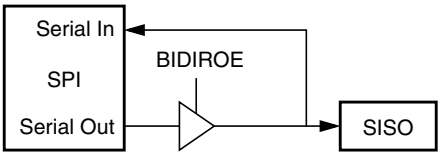
For example, for a 40 MHz bus clock, the maximum time-out period equals:

256 * 65536 * 25 ns = 419.43 ms.

The current 16-bit modulus down-counter value can be read via the PITCNT register. The micro timer down-counter values cannot be read.

The 8-bit micro timers can individually be restarted by writing a one to the corresponding force load micro timer PFLMT bits in the PIT control and force load micro timer (PITCFLMT) register. The 16-bit timers can individually be restarted by writing a one to the corresponding force load timer PFLT bits in the PIT forceload timer (PITFLT) register. If desired, any group of timers and micro timers can be restarted at the same time by using one 16-bit write to the adjacent PITCFLMT and PITFLT registers with the relevant bits set, as shown in Figure 17-28.

Table 21-11. Normal Mode and Bidirectional Mode

When SPE = 1	Master Mode MSTR = 1	Slave Mode MSTR = 0
Normal Mode SPC0 = 0		
Bidirectional Mode SPC0 = 1		

The direction of each serial I/O pin depends on the BIDIROE bit. If the pin is configured as an output, serial data from the shift register is driven out on the pin. The same pin is also the serial input to the shift register.

- The SCK is output for the master mode and input for the slave mode.
- The $\overline{SS}$ is the input or output for the master mode, and it is always the input for the slave mode.
- The bidirectional mode does not affect SCK and $\overline{SS}$ functions.

NOTE

In bidirectional master mode, with mode fault enabled, both data pins MISO and MOSI can be occupied by the SPI, though MOSI is normally used for transmissions in bidirectional mode and MISO is not used by the SPI. If a mode fault occurs, the SPI is automatically switched to slave mode. In this case MISO becomes occupied by the SPI and MOSI is not used. This must be considered, if the MISO pin is used for another purpose.

21.4.6 Error Conditions

The SPI has one error condition:

- Mode fault error

21.4.6.1 Mode Fault Error

If the $\overline{SS}$ input becomes low while the SPI is configured as a master, it indicates a system error where more than one master may be trying to drive the MOSI and SCK lines simultaneously. This condition is not permitted in normal operation, the MODF bit in the SPI status register is set automatically, provided the MODFEN bit is set.

In the special case where the SPI is in master mode and MODFEN bit is cleared, the $\overline{SS}$ pin is not used by the SPI. In this special case, the mode fault error function is inhibited and MODF remains cleared. In case

Table 24-4. Program IFR Fields

Global Address (PGMIFRON)	Size (Bytes)	Field Description
0x40_0000 – 0x40_0007	8	Device ID
0x40_0008 – 0x40_00E7	224	Reserved
0x40_00E8 – 0x40_00E9	2	Version ID
0x40_00EA – 0x40_00FF	22	Reserved
0x40_0100 – 0x40_013F	64	Program Once Field Refer to Section 24.4.2.6, “Program Once Command”
0x40_0140 – 0x40_01FF	192	Reserved

Table 24-5. P-Flash IFR Accessibility

Global Address (PGMIFRON)	Size (Bytes)	Accessed From
0x40_0000 – 0x40_01FF	512	XBUS0 (PBLK0)(1)
0x40_0200 – 0x40_03FF	512	Unimplemented
0x40_0400 – 0x40_05FF	512	Unimplemented
0x40_0600 – 0x40_07FF	512	XBUS1 (PBLK1)

1. Refer to Table 24-4 for more details.

Table 24-6. EEE Resource Fields

Global Address	Size (Bytes)	Description
0x10_0000 – 0x10_7FFF	32,768	D-Flash Memory (User and EEE)
0x10_8000 – 0x11_FFFF	98,304	Reserved
0x12_0000 – 0x12_007F	128	EEE Nonvolatile Information Register (EEEIFRON(1) = 1)
0x12_0080 – 0x12_0FFF	3,968	Reserved
0x12_1000 – 0x12_1F7F	3,968	Reserved
0x12_1F80 – 0x12_1FFF	128	EEE Tag RAM (TMGRAMON[1] = 1)
0x12_2000 – 0x12_3BFF	7,168	Reserved
0x12_3C00 – 0x12_3FFF	1,024	Memory Controller Scratch RAM (TMGRAMON[1] = 1)
0x12_4000 – 0x12_DFFF	40,960	Reserved
0x12_E000 – 0x12_FFFF	8,192	Reserved
0x13_0000 – 0x13_F7FF	63,488	Reserved
0x13_F800 – 0x13_FFFF	2,048	Buffer RAM (User and EEE)

1. MMCCTL1 register bit

24.4.1.4 P-Flash Commands

Table 24-31 summarizes the valid P-Flash commands along with the effects of the commands on the P-Flash block and other resources within the Flash module.

Table 24-31. P-Flash Commands

FCMD	Command	Function on P-Flash Memory
0x01	Erase Verify All Blocks	Verify that all P-Flash (and D-Flash) blocks are erased.
0x02	Erase Verify Block	Verify that a P-Flash block is erased.
0x03	Erase Verify P-Flash Section	Verify that a given number of words starting at the address provided are erased.
0x04	Read Once	Read a dedicated 64 byte field in the nonvolatile information register in P-Flash block 0 that was previously programmed using the Program Once command.
0x06	Program P-Flash	Program a phrase in a P-Flash block.
0x07	Program Once	Program a dedicated 64 byte field in the nonvolatile information register in P-Flash block 0 that is allowed to be programmed only once.
0x08	Erase All Blocks	Erase all P-Flash (and D-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.
0x09	Erase P-Flash Block	Erase a single P-Flash block. An erase of the full P-Flash block is only possible when FPLDIS, FPHDIS and FPOPEN bits in the FPROT register are set prior to launching the command.
0x0A	Erase P-Flash Sector	Erase all bytes in a P-Flash sector.
0x0B	Unsecure Flash	Supports a method of releasing MCU security by erasing all P-Flash (and D-Flash) blocks and verifying that all P-Flash (and D-Flash) blocks are erased.
0x0C	Verify Backdoor Access Key	Supports a method of releasing MCU security by verifying a set of security keys.
0x0D	Set User Margin Level	Specifies a user margin read level for all P-Flash blocks.
0x0E	Set Field Margin Level	Specifies a field margin read level for all P-Flash blocks (special modes only).

24.4.1.5 D-Flash and EEE Commands

Table 24-32 summarizes the valid D-Flash and EEE commands along with the effects of the commands on the D-Flash block and EEE operation.

Table 24-32. D-Flash Commands

FCMD	Command	Function on D-Flash Memory
0x01	Erase Verify All Blocks	Verify that all D-Flash (and P-Flash) blocks are erased.
0x02	Erase Verify Block	Verify that the D-Flash block is erased.
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.

Table 26-17. FSTAT Field Descriptions

Field	Description
7 CCIF	**Command Complete Interrupt Flag** — The CCIF flag indicates that a Flash command has completed. The CCIF flag is cleared by writing a 1 to CCIF to launch a command and CCIF will stay low until command completion or command violation. 0 Flash command in progress 1 Flash command has completed
5 ACCERR	**Flash Access Error Flag** — The ACCERR bit indicates an illegal access has occurred to the Flash memory caused by either a violation of the command write sequence (see Section 26.4.1.2) or issuing an illegal Flash command or when errors are encountered while initializing the EEE buffer ram during the reset sequence. While ACCERR is set, the CCIF flag cannot be cleared to launch a command. The ACCERR bit is cleared by writing a 1 to ACCERR. Writing a 0 to the ACCERR bit has no effect on ACCERR. 0 No access error detected 1 Access error detected
4 FPVIOL	**Flash Protection Violation Flag** —The FPVIOL bit indicates an attempt was made to program or erase an address in a protected area of P-Flash memory during a command write sequence. The FPVIOL bit is cleared by writing a 1 to FPVIOL. Writing a 0 to the FPVIOL bit has no effect on FPVIOL. While FPVIOL is set, it is not possible to launch a command or start a command write sequence. 0 No protection violation detected 1 Protection violation detected
3 MGBUSY	**Memory Controller Busy Flag** — The MGBUSY flag reflects the active state of the Memory Controller. 0 Memory Controller is idle 1 Memory Controller is busy executing a Flash command (CCIF = 0) or is handling internal EEE operations
2 RSVD	**Reserved Bit** — This bit is reserved and always reads 0.
1–0 MGSTAT[1:0]	**Memory Controller Command Completion Status Flag** — One or more MGSTAT flag bits are set if an error is detected during execution of a Flash command or during the Flash reset sequence. See Section 26.4.2, “Flash Command Description,” and Section 26.6, “Initialization” for details.

26.3.2.8 Flash Error Status Register (FERSTAT)

The FERSTAT register reflects the error status of internal Flash operations.

Offset Module Base + 0x0007

	7	6	5	4	3	2	1	0
R	ERSERIF	PGMERIF	0	EPVIOLIF	ERSVIF1	ERSVIF0	DFDIF	SFDIF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 26-12. Flash Error Status Register (FERSTAT)

All flags in the FERSTAT register are readable and only writable to clear the flag.

Table 26-28. FECCR Index=000 Bit Descriptions

Field	Description
15:8 PAR[7:0]	**ECC Parity Bits** — Contains the 8 parity bits from the 72 bit wide P-Flash data word or the 6 parity bits, allocated to PAR[5:0], from the 22 bit wide D-Flash word with PAR[7:6]=00.
7 XBUS01	**Bus Source Identifier** — The XBUS01 bit determines whether the ECC error was caused by a read access from the CPU or XGATE. 0 ECC Error happened on the CPU access 1 ECC Error happened on the XGATE access
6–0 GADDR[22:16]	**Global Address** — The GADDR[22:16] field contains the upper seven bits of the global address having caused the error.

The P-Flash word addressed by ECCRIX = 001 contains the lower 16 bits of the global address. The following four words addressed by ECCRIX = 010 to 101 contain the 64-bit wide data phrase. The four data words and the parity byte are the uncorrected data read from the P-Flash block.

The D-Flash word addressed by ECCRIX = 001 contains the lower 16 bits of the global address. The uncorrected 16-bit data word is addressed by ECCRIX = 010.

26.3.2.14 Flash Option Register (FOPT)

The FOPT register is the Flash option register.

Offset Module Base + 0x0010

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

= Unimplemented or Reserved

Figure 26-22. Flash Option Register (FOPT)

All bits in the FOPT register are readable but are not writable.

During the reset sequence, the FOPT register is loaded from the Flash nonvolatile byte in the Flash configuration field at global address 0x7F_FF0E located in P-Flash memory (see Table 26-3) as indicated by reset condition F in Figure 26-22. If a double bit fault is detected while reading the P-Flash phrase containing the Flash nonvolatile byte during the reset sequence, all bits in the FOPT register will be set.

Table 26-29. FOPT Field Descriptions

Field	Description
7–0 NV[7:0]	**Nonvolatile Bits** — The NV[7:0] bits are available as nonvolatile bits. Refer to the device user guide for proper use of the NV bits.

26.3.2.15 Flash Reserved0 Register (FRSV0)

This Flash register is reserved for factory testing.

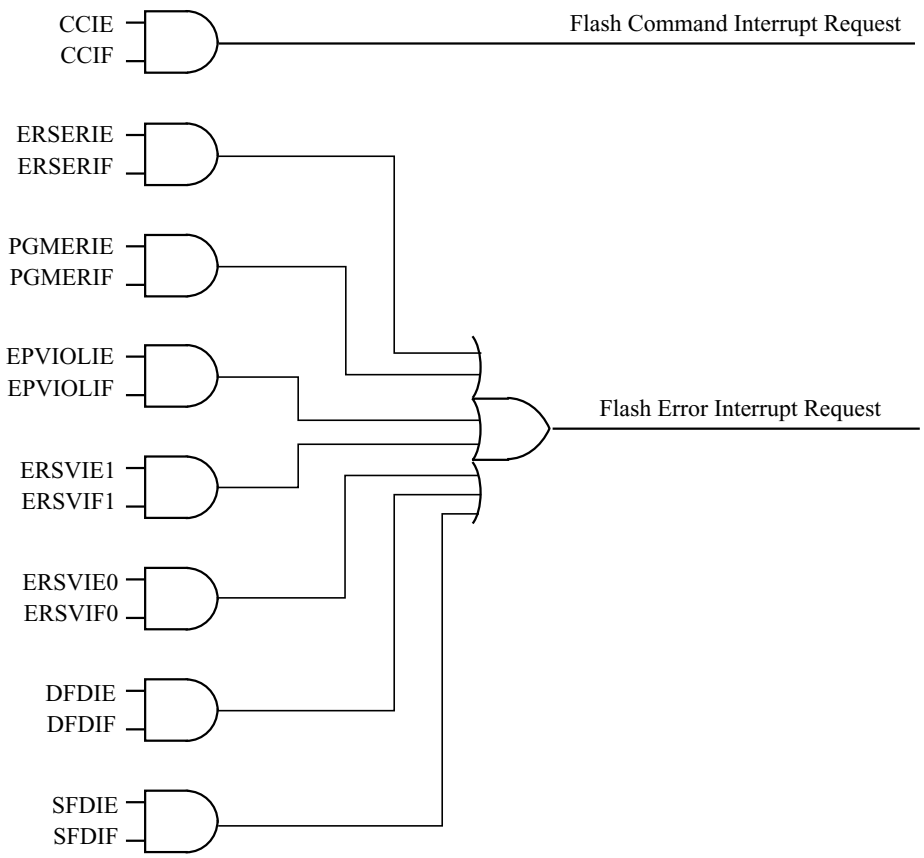


Figure 26-27. Flash Module Interrupts Implementation

26.4.4 Wait Mode

The Flash module is not affected if the MCU enters wait mode. The Flash module can recover the MCU from wait via the CCIF interrupt (see Section 26.4.3, “Interrupts”).

26.4.5 Stop Mode

If a Flash command is active (CCIF = 0) or an EE-Emulation operation is pending when the MCU requests stop mode, the current Flash operation will be completed before the CPU is allowed to enter stop mode.

26.5 Security

The Flash module provides security information to the MCU. The Flash security state is defined by the SEC bits of the FSEC register (see Table 26-12). During reset, the Flash module initializes the FSEC register using data read from the security byte of the Flash configuration field at global address 0x7F_FF0F.

27.4.1.2.1 Define FCCOB Contents

The FCCOB parameter fields must be loaded with all required parameters for the Flash command being executed. Access to the FCCOB parameter fields is controlled via the CCOBIX bits in the FCCOBIX register (see Section 27.3.2.3).

The contents of the FCCOB parameter fields are transferred to the Memory Controller when the user clears the CCIF command completion flag in the FSTAT register (writing 1 clears the CCIF to 0). The CCIF flag will remain clear until the Flash command has completed. Upon completion, the Memory Controller will return CCIF to 1 and the FCCOB register will be used to communicate any results. The flow for a generic command write sequence is shown in Figure 27-26.

Table 28-14. FECCRIX Field Descriptions

Field	Description
2-0 ECCRIX[2:0]	**ECC Error Register Index**— The ECCRIX bits are used to select which word of the FECCR register array is being read. See Section 28.3.2.13, “Flash ECC Error Results Register (FECCR),” for more details.

28.3.2.5 Flash Configuration Register (FCNFG)

The FCNFG register enables the Flash command complete interrupt and forces ECC faults on Flash array read access from the CPU or XGATE.

Offset Module Base + 0x0004

	7	6	5	4	3	2	1	0
R	CCIE	0	0	IGNSF	0	0	FDFD	FSFD
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 28-9. Flash Configuration Register (FCNFG)

CCIE, IGNSF, FDFD, and FSFD bits are readable and writable while remaining bits read 0 and are not writable.

Table 28-15. FCNFG Field Descriptions

Field	Description
7 CCIE	**Command Complete Interrupt Enable** — The CCIE bit controls interrupt generation when a Flash command has completed. 0 Command complete interrupt disabled 1 An interrupt will be requested whenever the CCIF flag in the FSTAT register is set (see Section 28.3.2.7)
4 IGNSF	**Ignore Single Bit Fault** — The IGNSF controls single bit fault reporting in the FERSTAT register (see Section 28.3.2.8). 0 All single bit faults detected during array reads are reported 1 Single bit faults detected during array reads are not reported and the single bit fault interrupt will not be generated

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:

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

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]

$$P_{IO} = \sum_i R_{DSON} \cdot I_{IO_i}^2$$

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

$$R_{DSON} = \frac{V_{OL}}{I_{OL}}; \text{for outputs driven low}$$

$$R_{DSON} = \frac{V_{DD35} - V_{OH}}{I_{OH}}; \text{for outputs driven high}$$

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-6. Thermal Package Characteristics (9S12XEQ512)(1)

Num	C	Rating	Symbol	Min	Typ	Max	Unit
LQFP144							
1a	D	Thermal resistance single sided PCB, natural convection	θ_{JA}	—	—	49	°C/W
1b	D	Thermal resistance single sided PCB @ 200 ft/min	θ_{JA}	—	—	40	°C/W
2a	D	Thermal resistance double sided PCB with 2 internal planes, natural convection	θ_{JA}	—	—	40	°C/W
2b	D	Thermal resistance double sided PCB with 2 internal planes @ 200 ft/min	θ_{JA}	—	—	34	°C/W
3	D	Junction to Board LQFP 144	θ_{JB}	—	—	28	°C/W
4	D	Junction to Case LQFP 144[2.]	θ_{JC}	—	—	9	°C/W
5	D	Junction to Package Top LQFP144[3]	Ψ_{JT}	—	—	2	°C/W
LQFP112							
6a	D	Thermal resistance single sided PCB, natural convection	θ_{JA}	—	—	50	°C/W
6b	D	Thermal resistance single sided PCB @ 200 ft/min	θ_{JA}	—	—	40	°C/W
7a	D	Thermal resistance double sided PCB with 2 internal planes, natural convection	θ_{JA}	—	—	40	°C/W
7b	D	Thermal resistance double sided PCB with 2 internal planes @ 200 ft/min	θ_{JA}	—	—	34	°C/W
8	D	Junction to Board LQFP112	θ_{JB}	—	—	28	°C/W
9	D	Junction to Case LQFP112[2.]	θ_{JC}	—	—	9	°C/W
10	D	Junction to Package Top LQFP112[3]	Ψ_{JT}	—	—	2	°C/W
QFP80							
11a	D	Thermal resistance single sided PCB, natural convection	θ_{JA}	—	—	50	°C/W
11b	D	Thermal resistance single sided PCB @ 200 ft/min	θ_{JA}	—	—	40	°C/W
12a	D	Thermal resistance double sided PCB with 2 internal planes, natural convection	θ_{JA}	—	—	37	°C/W
12b	D	Thermal resistance double sided PCB with 2 internal planes @ 200 ft/min	θ_{JA}	—	—	31	°C/W
13	D	Junction to Board QFP 80	θ_{JB}	—	—	23	°C/W
14	D	Junction to Case QFP 80(2)	θ_{JC}	—	—	13	°C/W
15	D	Junction to Package Top QFP 80(3)	Ψ_{JT}	—	—	3	°C/W

1. The values for thermal resistance are achieved by package simulations for the 9S12XEQ512 die.
2. Junction to case thermal resistance was simulated to be equivalent to the measured values using the cold plate technique with the cold plate temperature used as the “case” temperature. This basic cold plate measurement technique is described by MIL-STD 883D, Method 1012.1. This is the correct thermal metric to use to calculate thermal performance when the package is being used with a heat sink.
3. Thermal characterization parameter Ψ_{JT} is the “resistance” from junction to reference point thermocouple on top center of the case as defined in JESD51-2. Ψ_{JT} is a useful value to use to estimate junction temperature in a steady state customer enviroment.

A.1.9 I/O Characteristics

Table A-13. Run and Wait Current Characteristics

Conditions are shown in Table A-4 unless otherwise noted							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
Run supply current (No external load, Peripheral Configuration see Table A-11.)							
1	P	Peripheral Set[1] f_{osc}=4MHz, f_{bus}=50MHz	I_{DD35}	—	—	100	mA
Run supply current (No external load, Peripheral Configuration see Table A-10.)							
2		Peripheral Set[(1)] Devices S12XEP100, S12XEP768	I_{DD35}				mA
	C	f_{osc}=4MHz, f_{bus}=50MHz		—	84	—	
	T	f_{osc}=4MHz, f_{bus}=20MHz		—	43	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	24	—	
2a		Peripheral Set[1] All other devices					
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	72	—	mA
3		Peripheral Set[(2)]					mA
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	63	—	
	T	f_{osc}=4MHz, f_{bus}=20MHz		—	35	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	21	—	
4		Peripheral Set[(3)]					mA
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	62	—	
	T	f_{osc}=4MHz, f_{bus}=20MHz		—	34	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	21	—	
5		Peripheral Set[(4)]					mA
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	60	—	
	T	f_{osc}=4MHz, f_{bus}=20MHz		—	33	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	20	—	
6		Peripheral Set[(5)]					mA
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	59	—	
	T	f_{osc}=4MHz, f_{bus}=20MHz		—	33	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	20	—	
7		Peripheral Set[(6)]					mA
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	57	—	
	T	f_{osc}=4MHz, f_{bus}=20MHz		—	33	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	20	—	
Wait supply current							
8	C	Peripheral Set[1] ,PLL on XGATE executing code from RAM	I_{DDW}	—	—	85	mA
9		Peripheral Set[2]					
	T	f_{osc}=4MHz, f_{bus}=50MHz		—	50	—	
	T	f_{osc}=4MHz, f_{bus}=8MHz		—	12	—	
10	P	All modules disabled, RTI enabled, PLL off		—	—	10	

1. The following peripherals are on: ATD0/ATD1/ECT/IIC1/PWM/SPI0-SPI2/SCI0-SCI7/CAN0-CAN4/XGATE
2. The following peripherals are on: ATD0/ATD1/ECT/IIC1/PWM/SPI0-SPI2/SCI0-SCI7/CAN0-CAN4
3. The following peripherals are on: ATD0/ATD1/ECT/IIC1/PWM/SPI0-SPI2/SCI0-SCI7
4. The following peripherals are on: ATD0/ATD1/ECT/IIC1/PWM/SPI0-SPI2
5. The following peripherals are on: ATD0/ATD1/ECT/IIC1/PWM
6. The following peripherals are on: ATD0/ATD1/ECT/IIC1

Defining the jitter as:

$$J(N) = \max\left(\left|1 - \frac{t_{max}(N)}{N \cdot t_{nom}}\right|, \left|1 - \frac{t_{min}(N)}{N \cdot t_{nom}}\right|\right)$$

The following equation is a good fit for the maximum jitter:

$$J(N) = \frac{j_1}{\sqrt{N}} + j_2$$

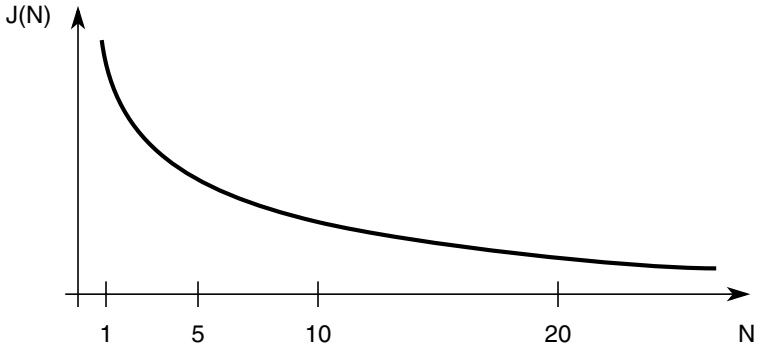


Figure A-6. Maximum bus clock jitter approximation