



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 <sup>2</sup> C, IrDA, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	119
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	64K 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	<a href="https://www.e-xfl.com/product-detail/nxp-semiconductors/s912xep100bmag">https://www.e-xfl.com/product-detail/nxp-semiconductors/s912xep100bmag</a>

Port	Pin Name	Pin Function & Priority <sup>(1)</sup>	I/O	Description	Pin Function after Reset
M	PM7	TXCAN3	O	MSCAN3 transmit pin	GPIO
		(TXCAN4)	O	MSCAN4 transmit pin	
		TXD3	O	Serial Communication Interface 3 transmit pin	
		GPIO	I/O	General-purpose I/O	
	PM6	RXCAN3	I	MSCAN3 receive pin	
		(RXCAN4)	I	MSCAN4 receive pin	
		RXD3	I	Serial Communication Interface 3 receive pin	
		GPIO	I/O	General-purpose I/O	
	PM5	TXCAN2	O	MSCAN2 transmit pin	
		(TXCAN0)	O	MSCAN0 transmit pin	
		(TXCAN4)	O	MSCAN4 transmit pin	
		(SCK0)	I/O	Serial Peripheral Interface 0 serial clock pin If CAN0 is routed to PM[3:2] the SPI0 can still be used in bidirectional master mode.	
		GPIO	I/O	General-purpose I/O	
	PM4	RXCAN2	I	MSCAN2 receive pin	
		(RXCAN0)	I	MSCAN0 receive pin	
		(RXCAN4)	I	MSCAN4 receive pin	
		(MOSI0)	I/O	Serial Peripheral Interface 0 master out/slave in pin If CAN0 is routed to PM[3:2] the SPI0 can still be used in bidirectional master mode.	
		GPIO	I/O	General-purpose I/O	
	PM3	TXCAN1	O	MSCAN1 transmit pin	
		(TXCAN0)	O	MSCAN0 transmit pin	
		(SS0)	I/O	Serial Peripheral Interface 0 slave select output in master mode, input for slave mode or master mode.	
		GPIO	I/O	General-purpose I/O	
	PM2	RXCAN1	I	MSCAN1 receive pin	
		(RXCAN0)	I	MSCAN0 receive pin	
		(MISO0)	I/O	Serial Peripheral Interface 0 master in/slave out pin	
		GPIO	I/O	General-purpose I/O	
	PM1	TXCAN0	O	MSCAN0 transmit pin	
		GPIO	I/O	General-purpose I/O	
	PM0	RXCAN0	I	MSCAN0 receive pin	
		GPIO	I/O	General-purpose I/O	

**Table 2-87. PTR Routing Register Field Descriptions (continued)**

Field	Description
5 PTRRR	<b>Port R routing—</b> This register configures the re-routing of the associated TIM channel. 1 TIMIOC5 is available on PP5 0 TIMIOC5 is available on PR5
4 PTRRR	<b>Port R routing—</b> This register configures the re-routing of the associated TIM channel. 1 TIMIOC4 is available on PP4 0 TIMIOC4 is available on PR4
3 PTRRR	<b>Port R routing—</b> This register configures the re-routing of the associated TIM channel. 1 TIMIOC3 is available on PP3 0 TIMIOC3 is available on PR3
2 PTRRR	<b>Port R routing—</b> This register configures the re-routing of the associated TIM channel. 1 TIMIOC2 is available on PP2 0 TIMIOC2 is available on PR2
1 PTRRR	<b>Port R routing—</b> This register configures the re-routing of the associated TIM channel. 1 TIMIOC1 is available on PP1 0 TIMIOC1 is available on PR1
0 PTRRR	<b>Port R routing—</b> This register configures the re-routing of the associated TIM channel. 1 TIMIOC0 is available on PP0 0 TIMIOC0 is available on PR0

### 2.3.93 Port L Data Register (PTL)

Address 0x0370

Access: User read/write<sup>(1)</sup>

	7	6	5	4	3	2	1	0
R	PTL7	PTLT6	PTL5	PTL4	PTL3	PTL2	PTL1	PTL0
W	(TXD7)	(RXD7)	(TXD6)	(RXD6)	(TXD5)	(RXD5)	(TXD4)	(RXD4)
Altern. Function	(TXD7)	(RXD7)	(TXD6)	(RXD6)	(TXD5)	(RXD5)	(TXD4)	(RXD4)
Reset	0	0	0	0	0	0	0	0

**Figure 2-91. Port L Data Register (PTL)**

1. Read: Anytime.  
Write: Anytime.

In emulation single-chip mode, accesses to global addresses which are not occupied by the on-chip resources (unimplemented areas) result in accesses to the external bus. CPU accesses to global addresses which are occupied by external memory space result in an illegal access reset (system reset) in case of no MPU error. BDM accesses to the external space are performed but the data will be undefined.

In single-chip modes accesses by the CPU (except for firmware commands) to any of the unimplemented areas (see [Figure 3-19](#)) will result in an illegal access reset (system reset) in case of no MPU error. BDM accesses to the unimplemented areas are allowed but the data will be undefined.

No misaligned word access from the BDM module will occur; these accesses are blocked in the BDM module (Refer to BDM Block Guide).

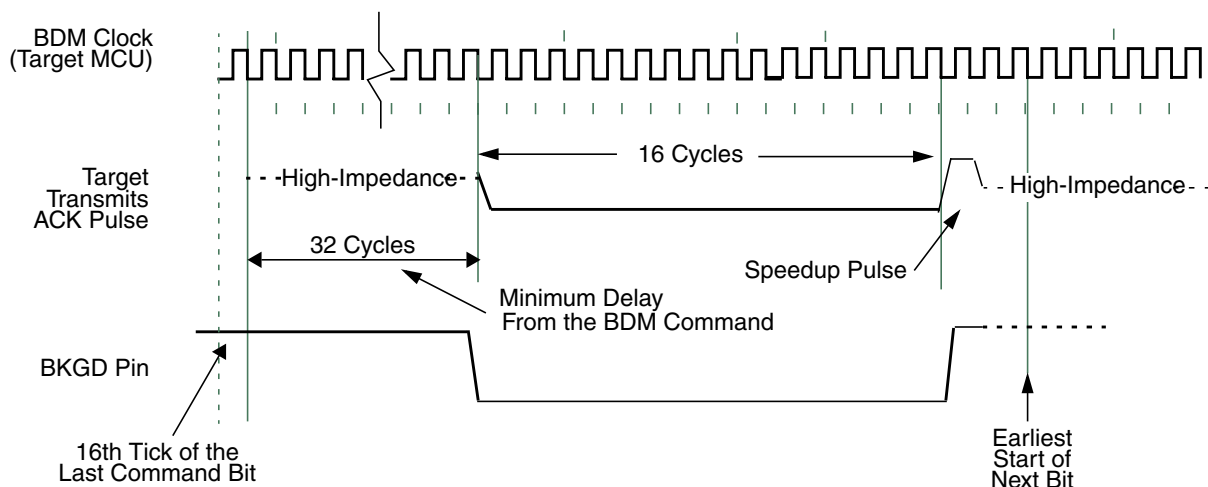
Misaligned word access to the last location of RAM is performed but the data will be undefined.

Misaligned word access to the last location of any global page (64 Kbyte) by any global instruction, is performed by accessing the last byte of the page and the first byte of the same page, considering the above mentioned misaligned access cases.

The non-internal resources (unimplemented areas or external space) are used to generate the chip selects (CS0,CS1,CS2 and CS3) (see [Figure 3-19](#)), which are only active in normal expanded, emulation expanded (see [Section 3.3.2.1, “MMC Control Register \(MMCCTL0\)”](#)).



compared to the serial communication rate. This protocol allows a great flexibility for the POD designers, since it does not rely on any accurate time measurement or short response time to any event in the serial communication.

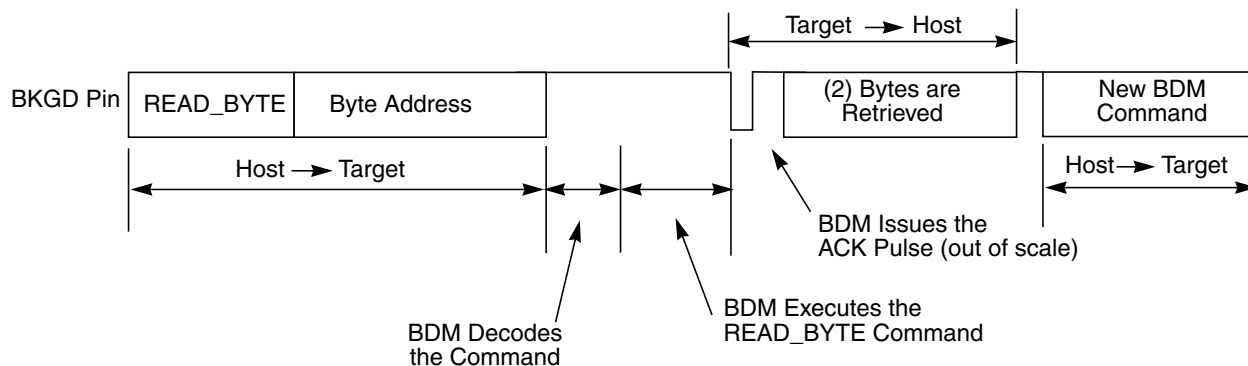


**Figure 7-11. Target Acknowledge Pulse (ACK)**

### NOTE

If the ACK pulse was issued by the target, the host assumes the previous command was executed. If the CPU enters wait or stop prior to executing a hardware command, the ACK pulse will not be issued meaning that the BDM command was not executed. After entering wait or stop mode, the BDM command is no longer pending.

Figure 7-12 shows the ACK handshake protocol in a command level timing diagram. The READ\_BYTE instruction is used as an example. First, the 8-bit instruction opcode is sent by the host, followed by the address of the memory location to be read. The target BDM decodes the instruction. A bus cycle is grabbed (free or stolen) by the BDM and it executes the READ\_BYTE operation. Having retrieved the data, the BDM issues an ACK pulse to the host controller, indicating that the addressed byte is ready to be retrieved. After detecting the ACK pulse, the host initiates the byte retrieval process. Note that data is sent in the form of a word and the host needs to determine which is the appropriate byte based on whether the address was odd or even.



**Figure 7-12. Handshake Protocol at Command Level**

## 8.3.2 Register Descriptions

This section consists of the S12XDBG control and trace buffer register descriptions in address order. Each comparator has a bank of registers that are visible through an 8-byte window between 0x0028 and 0x002F in the S12XDBG module register address map. When ARM is set in DBG1, the only bits in the S12XDBG module registers that can be written are ARM, TRIG, and COMRV[1:0].

### 8.3.2.1 Debug Control Register 1 (DBG1)

Address: 0x0020

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

Figure 8-3. Debug Control Register (DBG1)

Read: Anytime

Write: Bits 7, 1, 0 anytime

Bit 6 can be written anytime but always reads back as 0.

Bits 5:2 anytime S12XDBG is not armed.

#### NOTE

If a write access to DBG1 with the ARM bit position set occurs simultaneously to a hardware disarm from an internal trigger event, then the ARM bit is cleared due to the hardware disarm.

#### NOTE

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

Table 8-5. DBG1 Field Descriptions

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

BFINSI

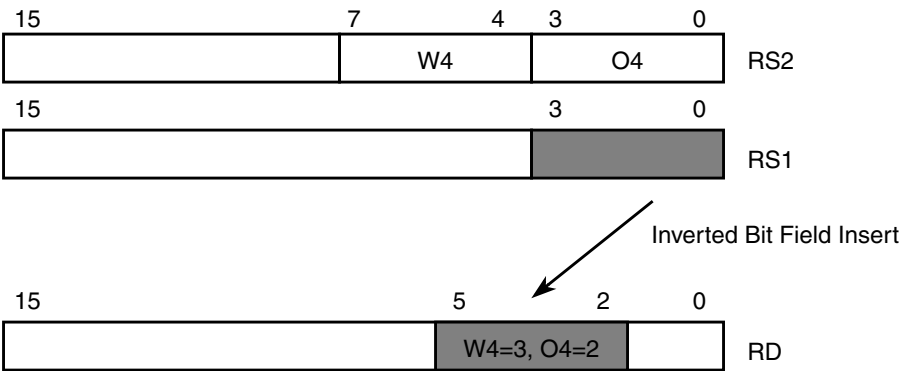
Bit Field Insert and Invert

BFINSI

Operation

```
!RS1[w:0] ⇒ RD[w+o:o];  
w = (RS2[7:4])  
o = (RS2[3:0])
```

Extracts  $w+1$  bits from register RS1 starting at position 0, inverts them and writes into register RD starting at position  $o$ . The remaining bits in RD are not affected. If  $(o+w) > 15$  the upper bits are ignored. Using R0 as a RS1, this command can be used to set bits.



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
BFINSI RD, RS1, RS2	TRI	0	1	1	1	0	RD	RS1	RS2	1	1	P

# BGE

Branch if Greater than or Equal to Zero

# BGE

## Operation

If  $N \wedge V = 0$ , then  $PC + \$0002 + (REL9 \ll 1) \Rightarrow PC$

Branch instruction to compare signed numbers.

Branch if  $RS1 \geq RS2$ :

SUB	R0, RS1, RS2
BGE	REL9

## 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
BGE REL9	REL9	0	0	1	1	0	1	0	REL9	PP/P

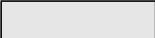
**Table 14-37. PBCTL Field Descriptions**

Field	Description
6 PBEN	<b>Pulse Accumulator B System Enable</b> — PBEN is independent from TEN. With timer disabled, the pulse accumulator can still function unless pulse accumulator is disabled. 0 16-bit pulse accumulator system disabled. 8-bit PAC1 and PAC0 can be enabled when their related enable bits in ICPAR are set. 1 Pulse accumulator B system enabled. The two 8-bit pulse accumulators PAC1 and PAC0 are cascaded to form the PACB 16-bit pulse accumulator B. When PACB is enabled, the PACN1 and PACN0 registers contents are respectively the high and low byte of the PACB. PA1EN and PA0EN control bits in ICPAR have no effect. The PACB shares the input pin with IC0.
1 PBOVI	<b>Pulse Accumulator B Overflow Interrupt Enable</b> 0 Interrupt inhibited 1 Interrupt requested if PBOVF is set

### 14.3.2.29 Pulse Accumulator B Flag Register (PBFLG)

Module Base + 0x0031

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

 = Unimplemented or Reserved

**Figure 14-52. Pulse Accumulator B Flag Register (PBFLG)**

Read: Anytime

Write used in the flag clearing mechanism. Writing a one to the flag clears the flag. Writing a zero will not affect the current status of the bit.

#### NOTE

When TFFCA = 1, the flag cannot be cleared via the normal flag clearing mechanism (writing a one to the flag). Reference [Section 14.3.2.6, “Timer System Control Register 1 \(TSCR1\)”](#).

All bits reset to zero.

PBFLG indicates when interrupt conditions have occurred. The flag can be cleared via the normal flag clearing mechanism (writing a one to the flag) or via the fast flag clearing mechanism (Reference TFFCA bit in [Section 14.3.2.6, “Timer System Control Register 1 \(TSCR1\)”](#)).

**Table 14-38. PBFLG Field Descriptions**

Field	Description
1 PBOVF	<b>Pulse Accumulator B Overflow Flag</b> — This bit is set when the 16-bit pulse accumulator B overflows from 0xFFFF to 0x0000, or when 8-bit pulse accumulator 1 (PAC1) overflows from 0x00FF to 0x0000. When PACMX = 1, PBOVF bit can also be set if 8-bit pulse accumulator 1 (PAC1) reaches 0x00FF and an active edge follows on IC1.

Module Base + 0x000E

Access: User read/write<sup>(1)</sup>

	7	6	5	4	3	2	1	0
R	RXERR7	RXERR6	RXERR5	RXERR4	RXERR3	RXERR2	RXERR1	RXERR0
W								
Reset:	0	0	0	0	0	0	0	0
	<div style="border: 1px solid black; width: 100px; height: 20px; display: inline-block;"></div> = Unimplemented							

**Figure 16-18. MSCAN Receive Error Counter (CANRXERR)**

1. Read: Only when in sleep mode (SLPRQ = 1 and SLPK = 1) or initialization mode (INITRQ = 1 and INITAK = 1)  
Write: Unimplemented

### NOTE

Reading this register when in any other mode other than sleep or initialization mode may return an incorrect value. For MCUs with dual CPUs, this may result in a CPU fault condition.

## 16.3.2.16 MSCAN Transmit Error Counter (CANTXERR)

This register reflects the status of the MSCAN transmit error counter.

Module Base + 0x000F

Access: User read/write<sup>(1)</sup>

	7	6	5	4	3	2	1	0
R	TXERR7	TXERR6	TXERR5	TXERR4	TXERR3	TXERR2	TXERR1	TXERR0
W								
Reset:	0	0	0	0	0	0	0	0
	<div style="border: 1px solid black; width: 100px; height: 20px; display: inline-block;"></div> = Unimplemented							

**Figure 16-19. MSCAN Transmit Error Counter (CANTXERR)**

1. Read: Only when in sleep mode (SLPRQ = 1 and SLPK = 1) or initialization mode (INITRQ = 1 and INITAK = 1)  
Write: Unimplemented

### NOTE

Reading this register when in any other mode other than sleep or initialization mode, may return an incorrect value. For MCUs with dual CPUs, this may result in a CPU fault condition.

Module Base + 0x00X2

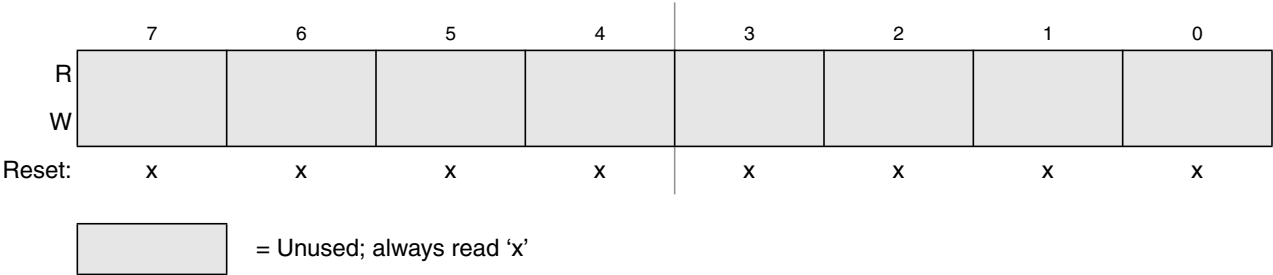


Figure 16-32. Identifier Register 2 — Standard Mapping

Module Base + 0x00X3

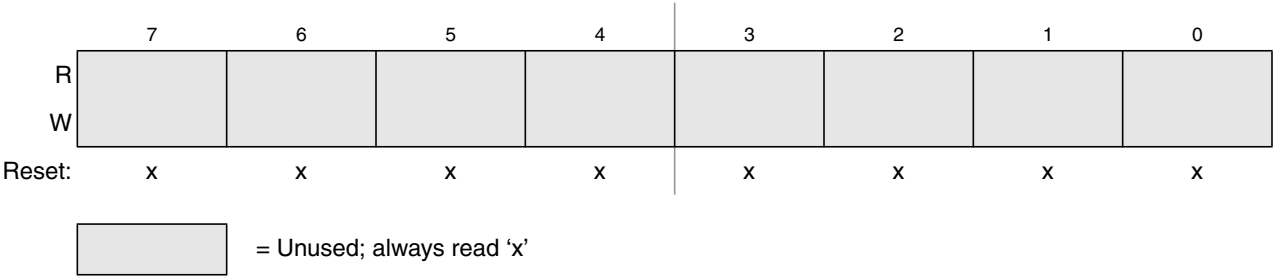


Figure 16-33. Identifier Register 3 — Standard Mapping

### 16.3.3.2 Data Segment Registers (DSR0-7)

The eight data segment registers, each with bits DB[7:0], contain the data to be transmitted or received. The number of bytes to be transmitted or received is determined by the data length code in the corresponding DLR register.

Module Base + 0x00X4 to Module Base + 0x00XB

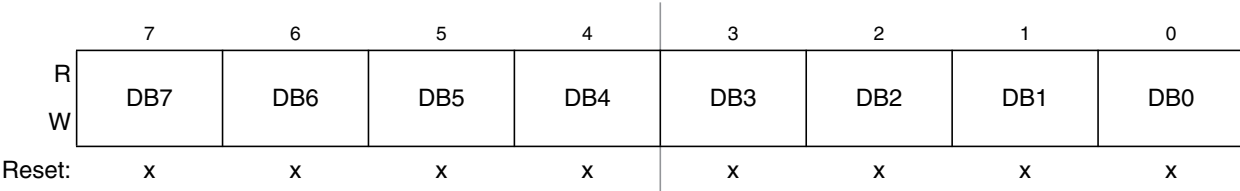


Figure 16-34. Data Segment Registers (DSR0–DSR7) — Extended Identifier Mapping

Table 16-33. DSR0–DSR7 Register Field Descriptions

Field	Description
7-0 DB[7:0]	Data bits 7-0

### 16.4.7.3 Receive Interrupt

A message is successfully received and shifted into the foreground buffer (RxFG) of the receiver FIFO. This interrupt is generated immediately after receiving the EOF symbol. The RXF flag is set. If there are multiple messages in the receiver FIFO, the RXF flag is set as soon as the next message is shifted to the foreground buffer.

### 16.4.7.4 Wake-Up Interrupt

A wake-up interrupt is generated if activity on the CAN bus occurs during MSCAN sleep or power-down mode.

#### NOTE

This interrupt can only occur if the MSCAN was in sleep mode (SLPRQ = 1 and SLPK = 1) before entering power down mode, the wake-up option is enabled (WUPE = 1), and the wake-up interrupt is enabled (WUPIE = 1).

### 16.4.7.5 Error Interrupt

An error interrupt is generated if an overrun of the receiver FIFO, error, warning, or bus-off condition occurs. MSCAN Receiver Flag Register (CANRFLG) indicates one of the following conditions:

- **Overrun** — An overrun condition of the receiver FIFO as described in [Section 16.4.2.3, “Receive Structures,”](#) occurred.
- **CAN Status Change** — The actual value of the transmit and receive error counters control the CAN bus state of the MSCAN. As soon as the error counters skip into a critical range (Tx/Rx-warning, Tx/Rx-error, bus-off) the MSCAN flags an error condition. The status change, which caused the error condition, is indicated by the TSTAT and RSTAT flags (see [Section 16.3.2.5, “MSCAN Receiver Flag Register \(CANRFLG\)”](#) and [Section 16.3.2.6, “MSCAN Receiver Interrupt Enable Register \(CANRIER\)”](#)).

### 16.4.7.6 Interrupt Acknowledge

Interrupts are directly associated with one or more status flags in either the MSCAN Receiver Flag Register (CANRFLG) or the MSCAN Transmitter Flag Register (CANTFLG). Interrupts are pending as long as one of the corresponding flags is set. The flags in CANRFLG and CANTFLG must be reset within the interrupt handler to handshake the interrupt. The flags are reset by writing a 1 to the corresponding bit position. A flag cannot be cleared if the respective condition prevails.

#### NOTE

It must be guaranteed that the CPU clears only the bit causing the current interrupt. For this reason, bit manipulation instructions (BSET) must not be used to clear interrupt flags. These instructions may cause accidental clearing of interrupt flags which are set after entering the current interrupt service routine.



channel 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.

When using the 16-bit concatenated mode, the clock source is determined by the low order 8-bit channel clock select control bits. That is channel 7 when channels 6 and 7 are concatenated, channel 5 when channels 4 and 5 are concatenated, channel 3 when channels 2 and 3 are concatenated, and channel 1 when channels 0 and 1 are concatenated. The resulting PWM is output to the pins of the corresponding low order 8-bit channel as also shown in [Figure 19-24](#). The polarity of the resulting PWM output is controlled by the PPOLx bit of the corresponding low order 8-bit channel as well.

**Table 21-9. SPIF Interrupt Flag Clearing Sequence**

XFRW Bit	SPIF Interrupt Flag Clearing Sequence		
0	Read SPISR with SPIF == 1	then	Read SPIDRL
1	Read SPISR with SPIF == 1	then	Byte Read SPIDRL <sup>(1)</sup>
			or
			Byte Read SPIDRH <sup>(2)</sup>   Byte Read SPIDRL
			or
			Word Read (SPIDRH:SPIDRL)

1. Data in SPIDRH is lost in this case.

2. SPIDRH can be read repeatedly without any effect on SPIF. SPIF Flag is cleared only by the read of SPIDRL after reading SPISR with SPIF == 1.

**Table 21-10. SPTEF Interrupt Flag Clearing Sequence**

XFRW Bit	SPTEF Interrupt Flag Clearing Sequence		
0	Read SPISR with SPTEF == 1	then	Write to SPIDRL <sup>(1)</sup>
1	Read SPISR with SPTEF == 1	then	Byte Write to SPIDRL <sup>1(2)</sup>
			or
			Byte Write to SPIDRH <sup>1(3)</sup>   Byte Write to SPIDRL <sup>1</sup>
			or
			Word Write to (SPIDRH:SPIDRL) <sup>1</sup>

1. Any write to SPIDRH or SPIDRL with SPTEF == 0 is effectively ignored.

2. Data in SPIDRH is undefined in this case.

3. SPIDRH can be written repeatedly without any effect on SPTEF. SPTEF Flag is cleared only by writing to SPIDRL after reading SPISR with SPTEF == 1.

**Table 24-64. Erase Verify D-Flash Section 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 <a href="#">Table 24-30</a> )
		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 an area of the D-Flash EEE partition
		Set if the requested section breaches the end of the D-Flash block or goes into the D-Flash EEE partition
	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

#### 24.4.2.16 Program D-Flash Command

The Program D-Flash operation programs one to four previously erased words in the D-Flash user partition. The Program D-Flash operation will confirm that the targeted location(s) were successfully programmed upon completion.

#### CAUTION

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

**Table 24-65. Program D-Flash Command FCCOB Requirements**

CCOBIX[2:0]	FCCOB Parameters	
000	0x11	Global address [22:16] to identify the D-Flash block
001	Global address [15:0] of word to be programmed	
010	Word 0 program value	
011	Word 1 program value, if desired	
100	Word 2 program value, if desired	
101	Word 3 program value, if desired	

Upon clearing CCIF to launch the Program D-Flash command, the user-supplied words will be transferred to the Memory Controller and be programmed. The CCOBIX index value at Program D-Flash command launch determines how many words will be programmed in the D-Flash block. No protection checks are made in the Program D-Flash operation on the D-Flash block, only access error checks. The CCIF flag is set when the operation has completed.

**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 <sup>(1)</sup>
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the read <sup>(2)</sup>
	MGSTAT0	Set if any non-correctable errors have been encountered during the read <sup>2</sup>
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.

### CAUTION

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

**Table 27-43. Program P-Flash Command FCCOB Requirements**

CCOBIX[2:0]	FCCOB Parameters	
000	0x06	Global address [22:16] to identify P-Flash block
001	Global address [15:0] of phrase location to be programmed <sup>(1)</sup>	
010	Word 0 program value	
011	Word 1 program value	
100	Word 2 program value	
101	Word 3 program value	

1. Global address [2:0] must be 000

Upon clearing CCIF to launch the Program P-Flash command, the Memory Controller will program the data words to the supplied global address and will then proceed to verify the data words read back as expected. The CCIF flag will set after the Program P-Flash operation has completed.

**Table 27-44. Program P-Flash Command Error Handling**

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 101 at command launch
		Set if command not available in current mode (see <a href="#">Table 27-30</a> )
		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 a Load Data Field command sequence is currently active and the selected block has previously been selected in the same command sequence
		Set if a Load Data Field command sequence is currently active and global address [17:0] does not match that previously supplied in the same command sequence
	FPVIOL	Set if the global address [22:0] points to a protected area
FERSTAT	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
	EPVIOLIF	None

#### 27.4.2.7 Program Once Command

The Program Once command restricts programming to a reserved 64 byte field (8 phrases) in the nonvolatile information register located in P-Flash block 0. The Program Once reserved field can be read using the Read Once command as described in [Section 27.4.2.4](#). The Program Once command must only be issued once since the nonvolatile information register in P-Flash block 0 cannot be erased. The Program

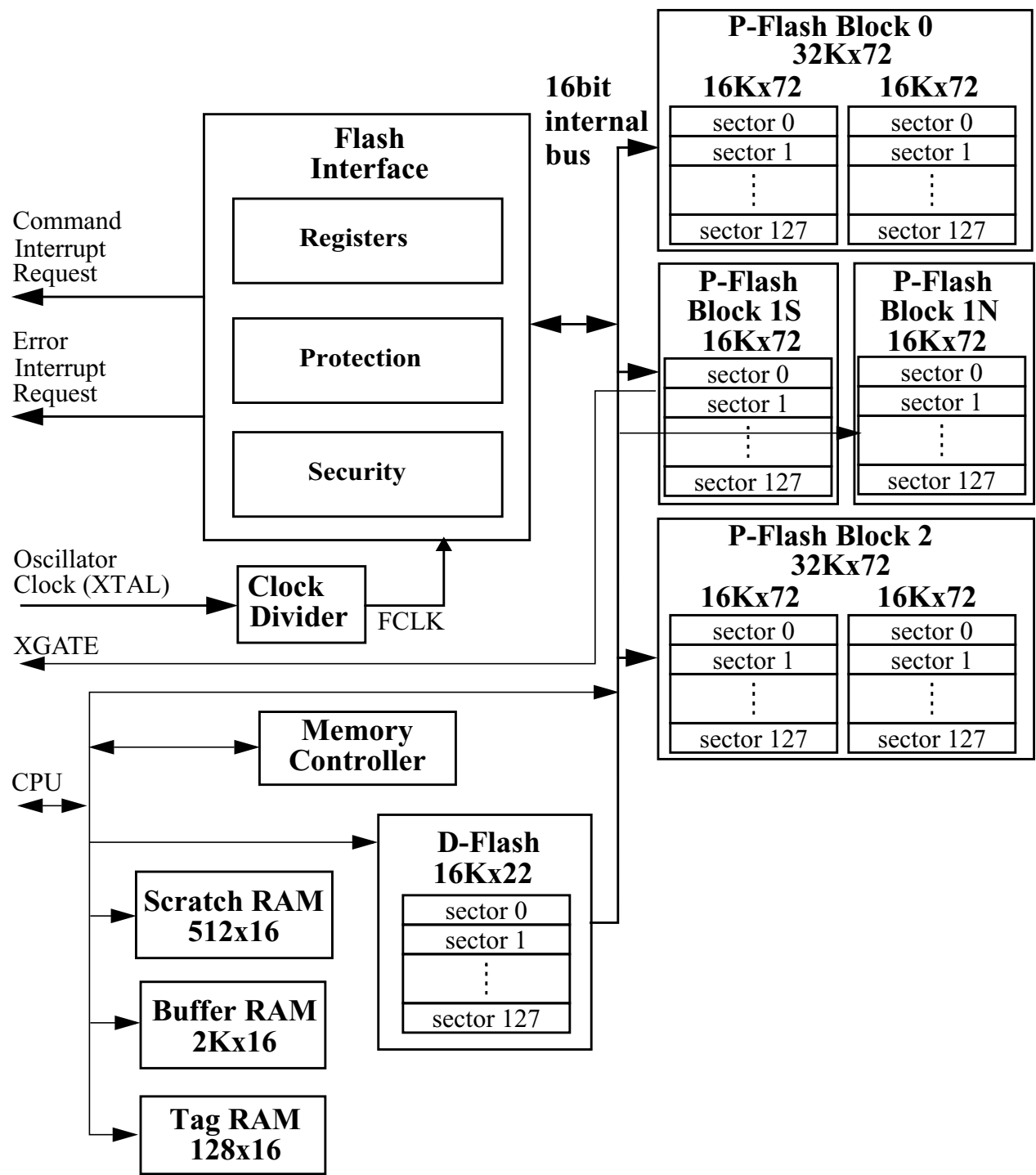


Figure 28-1. FTM768K4 Block Diagram

## 28.2 External Signal Description

The Flash module contains no signals that connect off-chip.

**Table 29-32. D-Flash Commands**

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

## 29.4.2 Flash Command Description

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

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

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

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

**NOTE**

Connecting VDDR to VSS disables the internal voltage regulator.

The VDDF, VSS1 pin pair supplies the internal NVM logic.

The VDD, VSS2 are the supply pins for the internal digital logic.

VDDPLL, VSSPLL pin pair supply the oscillator and the PLL.

VSS1, VSS2 and VSS3 are internally connected by metal.

VDDA1, and VDDA2 are internally connected by metal.

All VDDX pins are internally connected by metal.

All VSSX pins are internally connected by metal.

VDDA is connected to all VDDX pins by diodes for ESD protection such that VDDX must not exceed VDDA by more than a diode voltage drop. VDDA can exceed VDDX by more than a diode drop in order to support applications with a 5V A/D converter range and 3.3V I/O pin range.

VSSA and VSSX are connected by anti-parallel diodes for ESD protection.

**NOTE**

In the following context  $V_{DD35}$  is used for either VDDA, VDDR, and VDDX;  $V_{SS35}$  is used for either VSSA and VSSX unless otherwise noted.

$I_{DD35}$  denotes the sum of the currents flowing into the VDDA and VDDR pins. The Run mode current in the VDDX domain is external load dependent.

$V_{DD}$  is used for VDD,  $V_{SS}$  is used for VSS1, VSS2 and VSS3.

$V_{DDPLL}$  is used for VDDPLL,  $V_{SSPLL}$  is used for VSSPLL

$I_{DD}$  is used for the sum of the currents flowing into VDD, VDDF and VDDPLL.

**A.1.3 Pins**

There are four groups of functional pins.

**A.1.3.1 I/O Pins**

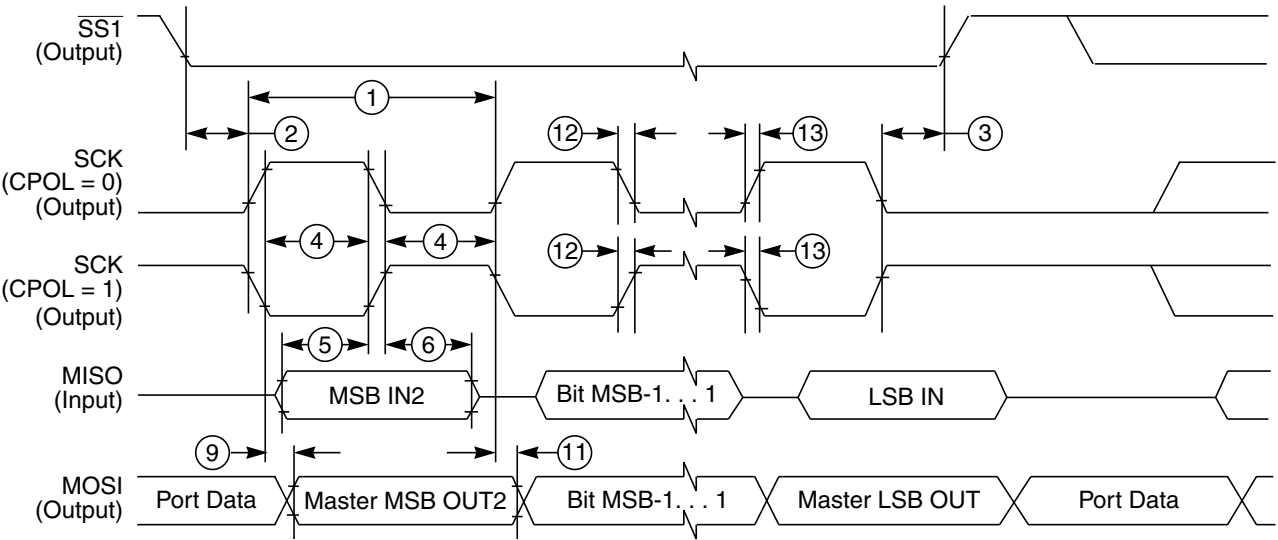
Standard I/O pins have a level in the range of 3.13V to 5.5 V. This class of pins is comprised of all port I/O pins (including PortAD), BKGD and the RESET pins. The internal structure of all those pins is identical; however, some of the functionality may be disabled. For example the BKGD pin pull up is always enabled.

**A.1.3.2 Analog Reference**

This group is made up by the  $V_{RH}$  and  $V_{RL}$  pins.



In Figure A-8 the timing diagram for master mode with transmission format CPHA=1 is depicted.



1. If configured as output
2. LSBF = 0. For LSBF = 1, bit order is LSB, bit 1, bit 2... MSB.

**Figure A-8. SPI Master Timing (CPHA = 1)**