



Welcome to [E-XFL.COM](http://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	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	CANbus, EBI/EMI, I <sup>2</sup> C, SCI, SPI
Peripherals	LCD, Motor control PWM, POR, PWM, WDT
Number of I/O	85
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	6K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	<a href="https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s12hz128cal">https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s12hz128cal</a>

## 1.7 Modes of Operation

Eight possible modes determine the operating configuration of the MC9S12HZ256. Each mode has an associated default memory map and external bus configuration.

Three low power modes exist for the device.

The operating mode out of reset is determined by the states of the MODC, MODB, and MODA pins during reset (Table 1-10). The MODC, MODB, and MODA bits in the MODE register show the current operating mode and provide limited mode switching during operation. The states of the MODC, MODB, and MODA pins are latched into these bits on the rising edge of the reset signal.

**Table 1-10. Mode Selection**

MODC	MODB	MODA	Mode Description
0	0	0	Special Single Chip, BDM allowed and ACTIVE. BDM is allowed in all other modes but a serial command is required to make BDM active.
0	0	1	Emulation Expanded Narrow, BDM allowed
X	1	0	Reserved for factory test
0	1	1	Emulation Expanded Wide, BDM allowed
1	0	0	Normal Single Chip, BDM allowed
1	0	1	Normal Expanded Narrow, BDM allowed
1	1	1	Normal Expanded Wide, BDM allowed

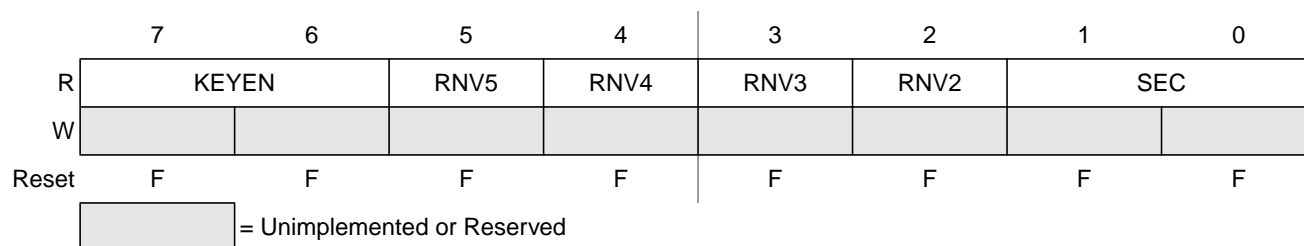
There are two basic types of operating modes:

1. **Normal** modes: Some registers and bits are protected against accidental changes.
2. **Special** modes: Allow greater access to protected control registers and bits for special purposes such as testing.

A system development and debug feature, background debug mode (BDM), is available in all modes. In special single-chip mode, BDM is active immediately after reset.

Some aspects of port E are not mode dependent. Bit 1 of port E is a general-purpose input or the  $\overline{\text{IRQ}}$  interrupt input.  $\overline{\text{IRQ}}$  can be enabled by bits in the CPU's condition codes register but it is inhibited at reset so this pin is initially configured as a simple input with a pull-up. Bit 0 of port E is a general-purpose input or the  $\overline{\text{XIRQ}}$  interrupt input.  $\overline{\text{XIRQ}}$  can be enabled by bits in the CPU's condition codes register but it is inhibited at reset so this pin is initially configured as a simple input with a pull-up. The ESTR bit in the EBICTL register is set to one by reset in any user mode. This assures that the reset vector can be fetched even if it is located in an external slow memory device. The PE6/MODB/IPIPE1 and PE5/MODA/IPIPE0 pins act as pull-down select inputs during reset and high-impedance select inputs after reset.

The following paragraphs discuss the default bus setup and describe which aspects of the bus can be changed after reset on a per mode basis.



**Figure 2-5. Flash Security Register (FSEC)**

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

The FSEC register is loaded from the Flash Configuration Field at address \$FF0F during the reset sequence, indicated by F in Figure 2-5.

**Table 2-5. FSEC Field Descriptions**

Field	Description
1-0 KEYEN[1:0]	<b>Backdoor Key Security Enable Bits</b> —The KEYEN[1:0] bits define the enabling of backdoor key access to the Flash module as shown in Table 2-6.
5-2 RNV[5:2]	<b>Reserved Nonvolatile Bits</b> — The RNV[5:2] bits must remain in the erased 1 state for future enhancements.
1-0 SEC[1:0]	<b>Flash Security Bits</b> — The SEC[1:0] bits define the security state of the MCU as shown in Table 2-7. If the Flash module is unsecured using backdoor key access, the SEC bits are forced to 10.

**Table 2-6. Flash KEYEN States**

KEYEN[1:0]	Status of Backdoor Key Access
00	DISABLED
01 <sup>1</sup>	DISABLED
10	<b>ENABLED</b>
11	DISABLED

<sup>1</sup> Preferred KEYEN state to disable Backdoor Key Access.

**Table 2-7. Flash Security States**

SEC[1:0]	Status of Security
00	SECURED
01 <sup>1</sup>	SECURED
10	<b>UNSECURED</b>
11	SECURED

<sup>1</sup> Preferred SEC state to set MCU to secured state.

The security function in the Flash module is described in Section 2.6, “Flash Module Security”.

Except for the sector erase abort command, a buffered command will wait for the active operation to be completed before being launched. The sector erase abort command is launched when the CBEIF flag is cleared as part of a sector erase abort command write sequence. After a command is launched, the completion of the command operation is indicated by the setting of the CCIF flag. The CCIF flag only sets when all active and buffered commands have been completed.

### 2.4.1.3 Valid Flash Commands

Table 2-20 summarizes the valid Flash commands along with the effects of the commands on the Flash block.

**Table 2-20. Valid Flash Command Description**

FCMDB	NVM Command	Function on Flash Memory
0x05	Erase Verify	Verify all memory bytes in the Flash block are erased. If the Flash block is erased, the BLANK flag in the FSTAT register will set upon command completion.
0x06	Data Compress	Compress data from a selected portion of the Flash block. The resulting signature is stored in the FDATA register.
0x20	Program	Program a word (two bytes) in the Flash block.
0x40	Sector Erase	Erase all memory bytes in a sector of the Flash block.
0x41	Mass Erase	Erase all memory bytes in the Flash block. A mass erase of the full Flash block is only possible when FPLDIS, FPHDIS, and FOPEN bits in the FPROT register are set prior to launching the command.
0x47	Sector Erase Abort	Abort the sector erase operation. The sector erase operation will terminate according to a set procedure. The Flash sector must not be considered erased if the ACCERR flag is set upon command completion.

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

### 2.4.1.3.2 Data Compress Command

The data compress command is used to check Flash code integrity by compressing data from a selected portion of the Flash block into a signature analyzer. The starting address for the data compress operation is defined by the address written during the command write sequence. The number of consecutive word addresses compressed is defined by the data written during the command write sequence. If the data value written is 0x0000, 64K addresses or 128 Kbytes will be compressed. After launching the data compress command, the CCIF flag in the FSTAT register will set after the data compress operation has completed. The number of bus cycles required to execute the data compress operation is equal to two times the number of addresses read plus 20 bus cycles as measured from the time the CBEIF flag is cleared until the CCIF flag is set. After the CCIF flag is set, the signature generated by the data compress operation is available in the FDATA register. The signature in the FDATA register can be compared to the expected signature to determine the integrity of the selected data stored in the Flash block. If the last address of the Flash block is reached during the data compress operation, data compression will continue with the starting address of the Flash block.

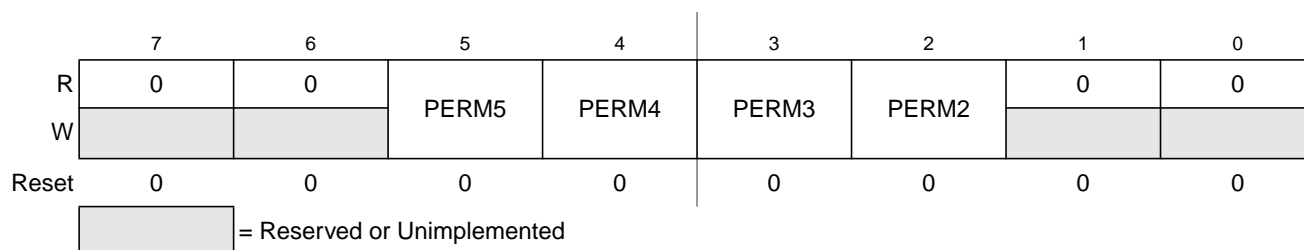
#### NOTE

Since the FDATA register (or data buffer) is written to as part of the data compress operation, a command write sequence is not allowed to be buffered behind a data compress command write sequence. The CBEIF flag will not set after launching the data compress command to indicate that a command must not be buffered behind it. If an attempt is made to start a new command write sequence with a data compress operation active, the ACCERR flag in the FSTAT register will be set. A new command write sequence must only be started after reading the signature stored in the FDATA register.

In order to take corrective action, it is recommended that the data compress command be executed on a Flash sector or subset of a Flash sector. If the data compress operation on a Flash sector returns an invalid signature, the Flash sector must be erased using the sector erase command and then reprogrammed using the program command.

The data compress command can be used to verify that a sector or sequential set of sectors are erased.

### 4.3.3.5 Port M Pull Device Enable Register (PERM)



**Figure 4-20. Port M Pull Device Enable Register (PERM)**

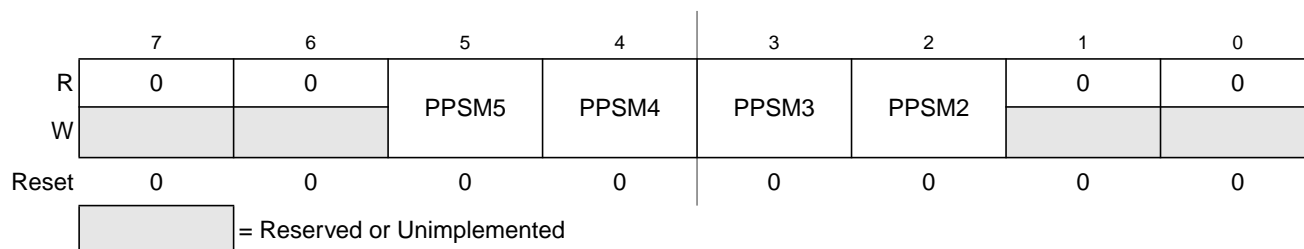
Read: Anytime. Write: Anytime.

This register configures whether a pull-up or a pull-down device is activated on configured input or wired-or output pins. If a pin is configured as push-pull output, the corresponding Pull Device Enable Register bit has no effect.

**Table 4-15. PERM Field Descriptions**

Field	Description
5:2 PERM[5:2]	<b>Pull Device Enable Port M</b> 0 Pull-up or pull-down device is disabled. 1 Pull-up or pull-down device is enabled.

### 4.3.3.6 Port M Polarity Select Register (PPSM)



**Figure 4-21. Port M Polarity Select Register (PPSM)**

Read: Anytime. Write: Anytime.

The Port M Polarity Select Register selects whether a pull-down or a pull-up device is connected to the pin. The Port M Polarity Select Register is effective only when the corresponding Data Direction Register bit is set to 0 (input) and the corresponding Pull Device Enable Register bit is set to 1.

If a CAN module is enabled, a pull-up device can be activated on the receiver pin, and on the transmitter pin if the corresponding wired-OR mode bit is set. Pull-down devices can not be activated on CAN pins.

### 4.3.6 Port T

Port T is associated with the 8-channel timer (TIM) and the liquid crystal display (LCD) driver. Each pin is assigned to these modules according to the following priority:  
 LCD Driver > timer > general-purpose I/O.

If the corresponding LCD frontplane drivers are enabled (and LCD module is enabled), the FP[27:24] outputs of the LCD module are available on port T pins PT[3:0].

If the corresponding LCD frontplane drivers are disabled (or LCD module is disabled) and the timer is enabled, the timer channels configured for output compare are available on port T pins PT[3:0].

Refer to the LCD block description chapter for information on enabling and disabling the LCD and its frontplane drivers. Refer to the TIM block description chapter for information on enabling and disabling the TIM module.

During reset, port T pins are configured as inputs with pull down.

#### 4.3.6.1 Port T I/O Register (PTT)

	7	6	5	4	3	2	1	0
R	PTT7	PTT6	PTT5	PTT4	PTT3	PTT2	PTT1	PTT0
W	PTT7	PTT6	PTT5	PTT4	PTT3	PTT2	PTT1	PTT0
TIM:	OC7	OC6	OC5	OC4	OC3	OC2	OC1	OC0
LCD:					1	1	1	1
Reset	0	0	0	0	0	0	0	0

Figure 4-37. Port T I/O Register (PTT)

Read: Anytime. Write: Anytime.

If the associated data direction bit (DDRTx) is set to 1 (output), a read returns the value of the I/O register bit.

If the associated data direction bit (DDRTx) is set to 0 (input) and the LCD frontplane driver is enabled (and LCD module is enabled), the associated I/O register bit (PTTx) reads “1”.

If the associated data direction bit (DDRTx) is set to 0 (input) and the LCD frontplane driver is disabled (or LCD module is disabled), a read returns the value of the pin.

## Chapter 5

# Clocks and Reset Generator (CRGV4)

### 5.1 Introduction

This specification describes the function of the clocks and reset generator (CRG).

#### 5.1.1 Features

The main features of this block are:

- Phase-locked loop (PLL) frequency multiplier
  - Reference divider
  - Automatic bandwidth control mode for low-jitter operation
  - Automatic frequency lock detector
  - CPU interrupt on entry or exit from locked condition
  - Self-clock mode in absence of reference clock
- System clock generator
  - Clock quality check
  - Clock switch for either oscillator- or PLL-based system clocks
  - User selectable disabling of clocks during wait mode for reduced power consumption
- Computer operating properly (COP) watchdog timer with time-out clear window
- System reset generation from the following possible sources:
  - Power-on reset
  - Low voltage reset
    - Refer to the device overview section for availability of this feature.
  - COP reset
  - Loss of clock reset
  - External pin reset
- Real-time interrupt (RTI)



If the MCU gets an external reset during pseudo-stop mode active, the CRG asynchronously restores all configuration bits in the register space to its default settings and starts the reset generator. After completing the reset sequence processing begins by fetching the normal reset vector. Pseudo-stop mode is exited and the MCU is in run mode again.

If the clock monitor is enabled ( $CME = 1$ ) the MCU is able to leave pseudo-stop mode when loss of oscillator/external clock is detected by a clock monitor fail. If the SCME bit is not asserted the CRG generates a clock monitor fail reset (CMRESET). The CRG's behavior for CMRESET is the same compared to external reset, but another reset vector is fetched after completion of the reset sequence. If the SCME bit is asserted the CRG generates a SCM interrupt if enabled ( $SCMIE=1$ ). After generating the interrupt the CRG enters self-clock mode and starts the clock quality checker (see Section 5.4.4, "Clock Quality Checker"). Then the MCU continues with normal operation. If the SCM interrupt is blocked by  $SCMIE = 0$ , the SCMIF flag will be asserted but the CRG will not wake-up from pseudo-stop mode.

If any other interrupt source (e.g. RTI) triggers exit from pseudo-stop mode the MCU immediately continues with normal operation. Because the PLL has been powered-down during stop mode the PLLSEL bit is cleared and the MCU runs on OSCCLK after leaving stop mode. The software must set the PLLSEL bit again, in order to switch system and core clocks to the PLLCLK.

Table 5-12 summarizes the outcome of a clock loss while in pseudo-stop mode.

In either level or edge triggered modes, the first conversion begins when the trigger is received. In both cases, the maximum latency time is one bus clock cycle plus any skew or delay introduced by the trigger circuitry.

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

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

### 7.4.2.2 General-Purpose Digital Input Port Operation

The input channel pins can be multiplexed between analog and digital data. As analog inputs, they are multiplexed and sampled to supply signals to the A/D converter. As digital inputs, they supply external input data that can be accessed through the digital port registers (PORTAD0 & PORTAD1) (input-only).

The analog/digital multiplex operation is performed in the input pads. The input pad is always connected to the analog inputs of the ATD10B16C. The input pad signal is buffered to the digital port registers. This buffer can be turned on or off with the ATDDIEN0 & ATDDIEN1 register. This is important so that the buffer does not draw excess current when analog potentials are presented at its input.

### 7.4.3 Operation in Low Power Modes

The ATD10B16C can be configured for lower MCU power consumption in three different ways:

- **Stop Mode**  
 Stop Mode: This halts A/D conversion. Exit from Stop mode will resume A/D conversion, But due to the recovery time the result of this conversion should be ignored.  
 Entering stop mode causes all clocks to halt and thus the system is placed in a minimum power standby mode. This halts any conversion sequence in progress. During recovery from stop mode, there must be a minimum delay for the stop recovery time  $t_{SR}$  before initiating a new ATD conversion sequence.
- **Wait Mode**  
 Wait Mode with AWAI = 1: This halts A/D conversion. Exit from Wait mode will resume A/D conversion, but due to the recovery time the result of this conversion should be ignored.  
 Entering wait mode, the ATD conversion either continues or halts for low power depending on the logical value of the AWAIT bit.
- **Freeze Mode**  
 Writing ADPU = 0 (Note that all ATD registers remain accessible.): This aborts any A/D conversion in progress.  
 In freeze mode, the ATD10B16C will behave according to the logical values of the FRZ1 and FRZ0 bits. This is useful for debugging and emulation.

#### NOTE

The reset value for the ADPU bit is zero. Therefore, when this module is reset, it is reset into the power down state.

**Table 12-4. CANCTL1 Register Field Descriptions (continued)**

Field	Description
1 SLPAK	<p><b>Sleep Mode Acknowledge</b> — This flag indicates whether the MSCAN module has entered sleep mode (see Section 12.4.6.4, “MSCAN Sleep Mode”). It is used as a handshake flag for the SLPRQ sleep mode request. Sleep mode is active when SLPRQ = 1 and SLPAK = 1. Depending on the setting of WUPE, the MSCAN will clear the flag if it detects activity on the CAN bus while in sleep mode.</p> <p>0 Running — The MSCAN operates normally                      1 Sleep mode active — The MSCAN has entered sleep mode</p>
0 INITAK	<p><b>Initialization Mode Acknowledge</b> — This flag indicates whether the MSCAN module is in initialization mode (see Section 12.4.6.5, “MSCAN Initialization Mode”). It is used as a handshake flag for the INITRQ initialization mode request. Initialization mode is active when INITRQ = 1 and INITAK = 1. The registers CANCTL1, CANBTR0, CANBTR1, CANIDAC, CANIDAR0–CANIDAR7, and CANIDMR0–CANIDMR7 can be written only by the CPU when the MSCAN is in initialization mode.</p> <p>0 Running — The MSCAN operates normally                      1 Initialization mode active — The MSCAN has entered initialization mode</p>

## 14.4.2 Slave Mode

The SPI operates in slave mode when the MSTR bit in SPI Control Register1 is clear.

- SCK Clock

In slave mode, SCK is the SPI clock input from the master.

- MISO and MOSI Pins

In slave mode, the function of the serial data output pin (MISO) and serial data input pin (MOSI) is determined by the SPC0 bit and BIDIROE bit in SPI Control Register 2.

- $\overline{SS}$  Pin

The  $\overline{SS}$  pin is the slave select input. Before a data transmission occurs, the  $\overline{SS}$  pin of the slave SPI must be low.  $\overline{SS}$  must remain low until the transmission is complete. If  $\overline{SS}$  goes high, the SPI is forced into idle state.

The  $\overline{SS}$  input also controls the serial data output pin, if  $\overline{SS}$  is high (not selected), the serial data output pin is high impedance, and, if  $\overline{SS}$  is low the first bit in the SPI Data Register is driven out of the serial data output pin. Also, if the slave is not selected ( $\overline{SS}$  is high), then the SCK input is ignored and no internal shifting of the SPI shift register takes place.

Although the SPI is capable of duplex operation, some SPI peripherals are capable of only receiving SPI data in a slave mode. For these simpler devices, there is no serial data out pin.

### NOTE

When peripherals with duplex capability are used, take care not to simultaneously enable two receivers whose serial outputs drive the same system slave's serial data output line.

As long as no more than one slave device drives the system slave's serial data output line, it is possible for several slaves to receive the same transmission from a master, although the master would not receive return information from all of the receiving slaves.

If the CPHA bit in SPI Control Register 1 is clear, odd numbered edges on the SCK input cause the data at the serial data input pin to be latched. Even numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

If the CPHA bit is set, even numbered edges on the SCK input cause the data at the serial data input pin to be latched. Odd numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

When CPHA is set, the first edge is used to get the first data bit onto the serial data output pin. When CPHA is clear and the  $\overline{SS}$  input is low (slave selected), the first bit of the SPI data is driven out of the serial data output pin. After the eighth shift, the transfer is considered complete and the received data is transferred into the SPI Data Register. To indicate transfer is complete, the SPIF flag in the SPI Status Register is set.

### NOTE

A change of the bits CPOL, CPHA, SSOE, LSBFE, MODFEN, SPC0 and BIDIROE with SPC0 set in slave mode will corrupt a transmission in progress and has to be avoided.

## 15.3.2 Register Descriptions

The following paragraphs describe in detail all the registers and register bits in the PWM8B6C module.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
PWME	R	0	0	PWME5	PWME4	PWME3	PWME2	PWME1	PWME0
	W								
PWMPOL	R	0	0	PPOL5	PPOL4	PPOL3	PPOL2	PPOL1	PPOL0
	W								
PWMCLK	R	0	0	PCLK5	PCLK4	PCLK3	PCLK2	PCLK1	PCLK0
	W								
PWMPRCLK	R	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
	W								
PWMCAE	R	0	0	CAE5	CAE4	CAE2	CAE2	CAE1	CAE0
	W								
PWMCTL	R	0	CON45	CON23	CON01	PSWAI	PFRZ	0	0
	W								
PWMTST	R	0	0	0	0	0	0	0	0
	W								
PWMPRSC	R	0	0	0	0	0	0	0	0
	W								
PWMSCLA	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMSCLB	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMSCNTA	R	0	0	0	0	0	0	0	0
	W								
PWMSCNTB	R	0	0	0	0	0	0	0	0
	W								
PWMCNT0	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
PWMCNT1	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
PWMCNT2	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 15-2. PWM Register Summary

### 15.4.2.1 PWM Enable

Each PWM channel has an enable bit (PWME<sub>x</sub>) to start its waveform output. When any of the PWME<sub>x</sub> bits are set (PWME<sub>x</sub> = 1), the associated PWM output signal 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 PWME<sub>x</sub> and the clock source. An exception to this is when channels are concatenated. Refer to Section 15.4.2.7, “PWM 16-Bit Functions,” for more detail.

#### NOTE

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

On the front end of the PWM timer, the clock is enabled to the PWM circuit by the PWME<sub>x</sub> bit being high. There is an edge-synchronizing circuit to guarantee that the clock will only be enabled or disabled at an edge. When the channel is disabled (PWME<sub>x</sub> = 0), the counter for the channel does not count.

### 15.4.2.2 PWM Polarity

Each channel has a polarity bit to allow starting a waveform cycle with a high or low signal. This is shown on the block diagram as a mux select of either the Q output or the  $\bar{Q}$  output of the PWM output flip-flop. When one of the bits in the PWMPOL register is set, the associated PWM channel output is high at the beginning of the waveform, then goes low when the duty count is reached. Conversely, if the polarity bit is 0, the output starts low and then goes high when the duty count is reached.

### 15.4.2.3 PWM Period and Duty

Dedicated period and duty registers exist for each channel and are double buffered so that if they change while the channel is enabled, the change will NOT take effect until one of the following occurs:

- The effective period ends
- The counter is written (counter resets to 0x0000)
- The channel is disabled

In this way, the output of the PWM will always be either the old waveform or the new waveform, not some variation in between. If the channel is not enabled, then writes to the period and duty registers will go directly to the latches as well as the buffer.

A change in duty or period can be forced into effect “immediately” by writing the new value to the duty and/or period registers and then writing to the counter. This forces the counter to reset and the new duty and/or period values to be latched. In addition, because the counter is readable it is possible to know where the count is with respect to the duty value and software can be used to make adjustments.

#### NOTE

When forcing a new period or duty into effect immediately, an irregular PWM cycle can occur.

Depending on the polarity bit, the duty registers will contain the count of either the high time or the low time.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000C TIE	R W	C7I	C6I	C5I	C4I	C3I	C2I	C1I	C0I
0x000D TSCR2	R W	TOI	0	0	0	TCRE	PR2	PR1	PR0
0x000E TFLG1	R W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
0x000F TFLG2	R W	TOF	0	0	0	0	0	0	0
0x0010–0x001F TCxH–TCxL	R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
	R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0020 PACTL	R W	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
0x0021 PAFLG	R W	0	0	0	0	0	0	PAOVF	PAIF
0x0022 PACNTH	R W	PACNT15	PACNT14	PACNT13	PACNT12	PACNT11	PACNT10	PACNT9	PACNT8
0x0023 PACNTL	R W	PACNT7	PACNT6	PACNT5	PACNT4	PACNT3	PACNT2	PACNT1	PACNT0
0x0024–0x002F Reserved	R W								

= Unimplemented or Reserved

Figure 16-5. TIM16B8C Register Summary (continued)

### 16.3.2.1 Timer Input Capture/Output Compare Select (TIOS)

	7	6	5	4	3	2	1	0
R W	IOS7	IOS6	IOS5	IOS4	IOS3	IOS2	IOS1	IOS0
Reset	0	0	0	0	0	0	0	0

Figure 16-6. Timer Input Capture/Output Compare Select (TIOS)





# Chapter 20 Interrupt (INTV1)

## 20.1 Introduction

This section describes the functionality of the interrupt (INT) sub-block of the S12 core platform.

A block diagram of the interrupt sub-block is shown in Figure 20-1.

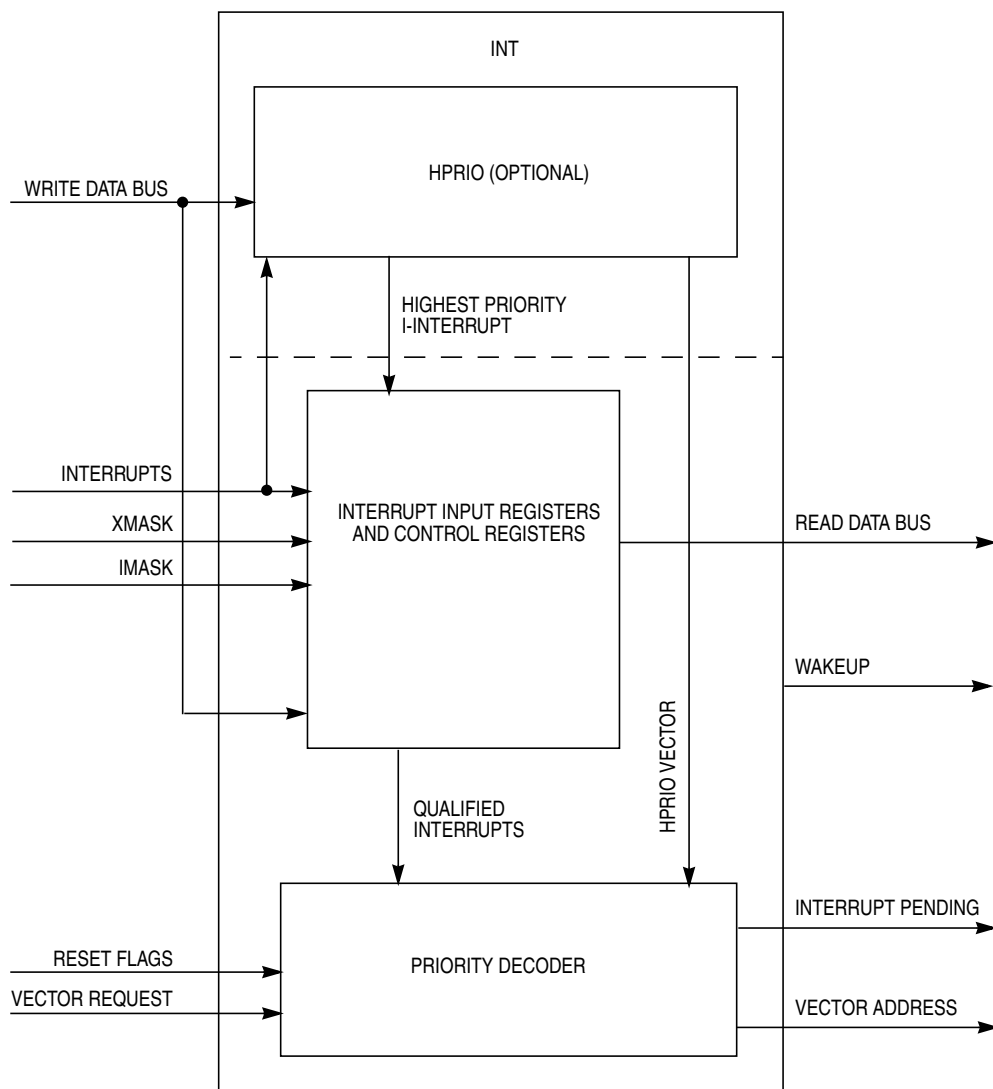


Figure 20-1. INT Block Diagram

## 20.2 External Signal Description

Most interfacing with the interrupt sub-block is done within the core. However, the interrupt does receive direct input from the multiplexed external bus interface (MEBI) sub-block of the core for the  $\overline{\text{IRQ}}$  and  $\overline{\text{XIRQ}}$  pin data.

## 20.3 Memory Map and Register Definition

Detailed descriptions of the registers and associated bits are given in the subsections that follow.

### 20.3.1 Module Memory Map

Table 20-1. INT Memory Map

Address Offset	Use	Access
0x0015	Interrupt Test Control Register (ITCR)	R/W
0x0016	Interrupt Test Registers (ITEST)	R/W
0x001F	Highest Priority Interrupt (Optional) (HPRIO)	R/W

### 20.3.2 Register Descriptions

#### 20.3.2.1 Interrupt Test Control Register

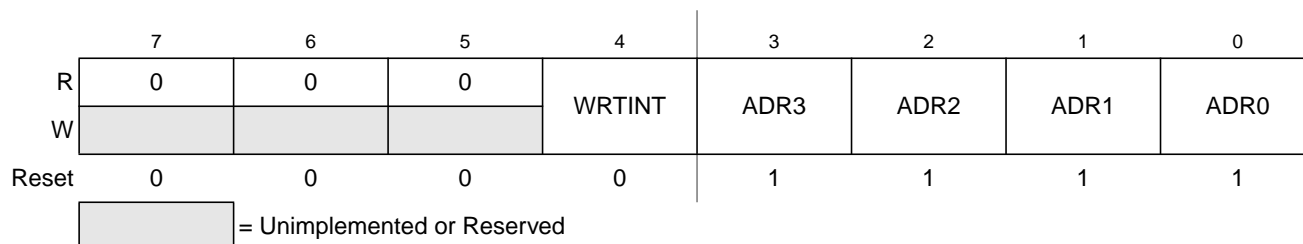


Figure 20-2. Interrupt Test Control Register (ITCR)

Read: See individual bit descriptions

Write: See individual bit descriptions

**Table 21-1. External System Pins Associated With MEBI (continued)**

Pin Name	Pin Functions	Description
PE4/ECLK	PE4	General-purpose I/O pin, see PORTE and DDRE registers.
	ECLK	Bus timing reference clock, can operate as a free-running clock at the system clock rate or to produce one low-high clock per visible access, with the high period stretched for slow accesses. ECLK is controlled by the NECLK bit in PEAR, the IVIS bit in MODE, and the ESTR bit in EBICTL.
PE3/ $\overline{\text{LSTRB}}$ / $\overline{\text{TAGLO}}$	PE3	General-purpose I/O pin, see PORTE and DDRE registers.
	$\overline{\text{LSTRB}}$	Low strobe bar, 0 indicates valid data on D7–D0.
	SZ8	In special peripheral mode, this pin is an input indicating the size of the data transfer (0 = 16-bit; 1 = 8-bit).
	$\overline{\text{TAGLO}}$	In expanded wide mode or emulation narrow modes, when instruction tagging is on and low strobe is enabled, a 0 at the falling edge of E tags the low half of the instruction word being read into the instruction queue.
PE2/ $\overline{\text{R}}/\overline{\text{W}}$	PE2	General-purpose I/O pin, see PORTE and DDRE registers.
	$\overline{\text{R}}/\overline{\text{W}}$	Read/write, indicates the direction of internal data transfers. This is an output except in special peripheral mode where it is an input.
PE1/ $\overline{\text{IRQ}}$	PE1	General-purpose input-only pin, can be read even if $\overline{\text{IRQ}}$ enabled.
	$\overline{\text{IRQ}}$	Maskable interrupt request, can be level sensitive or edge sensitive.
PE0/ $\overline{\text{XIRQ}}$	PE0	General-purpose input-only pin.
	$\overline{\text{XIRQ}}$	Non-maskable interrupt input.
PK7/ $\overline{\text{ECS}}$	PK7	General-purpose I/O pin, see PORTK and DDRK registers.
	$\overline{\text{ECS}}$	Emulation chip select
PK6/ $\overline{\text{XCS}}$	PK6	General-purpose I/O pin, see PORTK and DDRK registers.
	$\overline{\text{XCS}}$	External data chip select
PK5/X19 thru PK0/X14	PK5–PK0	General-purpose I/O pins, see PORTK and DDRK registers.
	X19–X14	Memory expansion addresses

Detailed descriptions of these pins can be found in the device overview chapter.

## 21.3 Memory Map and Register Definition

A summary of the registers associated with the MEBI sub-block is shown in Table 21-2. Detailed descriptions of the registers and bits are given in the subsections that follow. On most chips the registers are mappable. Therefore, the upper bits may not be all 0s as shown in the table and descriptions.

### 22.3.2.4 Miscellaneous System Control Register (MISC)

	7	6	5	4	3	2	1	0
R	0	0	0	0	EXSTR1	EXSTR0	ROMHM	ROMON
W								
Reset: Expanded or Emulation	0	0	0	0	1	1	0	— <sup>1</sup>
Reset: Peripheral or Single Chip	0	0	0	0	1	1	0	1
Reset: Special Test	0	0	0	0	1	1	0	0

1. The reset state of this bit is determined at the chip integration level.

= Unimplemented or Reserved

**Figure 22-6. Miscellaneous System Control Register (MISC)**

Read: Anytime

Write: As stated in each bit description

**NOTE**

Writes to this register take one cycle to go into effect.

This register initializes miscellaneous control functions.

**Table 22-5. INITEE Field Descriptions**

Field	Description
3:2 EXSTR[1:0]	<b>External Access Stretch Bits 1 and 0</b> Write: once in normal and emulation modes and anytime in special modes This two-bit field determines the amount of clock stretch on accesses to the external address space as shown in Table 22-6. In single chip and peripheral modes these bits have no meaning or effect.
1 ROMHM	<b>FLASH EEPROM or ROM Only in Second Half of Memory Map</b> Write: once in normal and emulation modes and anytime in special modes 0 The fixed page(s) of FLASH EEPROM or ROM in the lower half of the memory map can be accessed. 1 Disables direct access to the FLASH EEPROM or ROM in the lower half of the memory map. These physical locations of the FLASH EEPROM or ROM remain accessible through the program page window.
0 ROMON	<b>ROMON — Enable FLASH EEPROM or ROM</b> Write: once in normal and emulation modes and anytime in special modes This bit is used to enable the FLASH EEPROM or ROM memory in the memory map. 0 Disables the FLASH EEPROM or ROM from the memory map. 1 Enables the FLASH EEPROM or ROM in the memory map.

**Table 22-6. External Stretch Bit Definition**

Stretch Bit EXSTR1	Stretch Bit EXSTR0	Number of E Clocks Stretched
0	0	0
0	1	1
1	0	2
1	1	3

## A.4 Voltage Regulator

### A.4.1 Operating Conditions

**Table A-13. Voltage Regulator - Operating Conditions**

Conditions are shown in Table A-4 unless otherwise noted							
Num	C	Characteristic	Symbol	Min	Typical	Max	Unit
1	P	Input Voltages	$V_{VDDR,A}$	4.5	—	5.5	V
2	P	Regulator Current Shutdown Mode	$I_{REG}$	—	12	40	$\mu$ A
3	P	Output Voltage Core Full Performance Mode Shutdown Mode	$V_{DD}$	2.35 —	2.5 — <sup>1</sup>	2.75 —	V V
4	P	Output Voltage PLL Full Performance Mode Shutdown Mode	$V_{DDPLL}$	2.35 —	2.5 — <sup>2</sup>	2.75 —	V V
7	P	Low Voltage Interrupt <sup>3</sup> Assert Level Deassert Level	$V_{LVIA}$ $V_{LVID}$	4.0 4.15	4.37 4.52	4.66 4.77	V V
8	P	Low Voltage Reset <sup>4</sup> Assert Level	$V_{LVRA}$	2.25	—	—	V
9	C	Power-on Reset <sup>5</sup> Assert Level Deassert Level	$V_{PORA}$ $V_{PORD}$	0.97 —	— —	— 2.05	V V

<sup>1</sup> High Impedance Output

<sup>2</sup> High Impedance Output

<sup>3</sup> Monitors  $V_{DDA}$ , active only in Full Performance Mode. Indicates I/O & ADC performance degradation due to low supply voltage.

<sup>4</sup> Monitors  $V_{DD}$ , active only in Full Performance Mode. MCU is monitored by the POR in RPM (see Figure A-2)

<sup>5</sup> Monitors  $V_{DD}$ . Active in all modes.

### A.4.2 Chip Power-up and Voltage Drops

The voltage regulator sub modules LVI (low voltage interrupt), POR (power-on reset) and LVR (low voltage reset) handle chip power-up or drops of the supply voltage. Their function is described in Figure A-2.