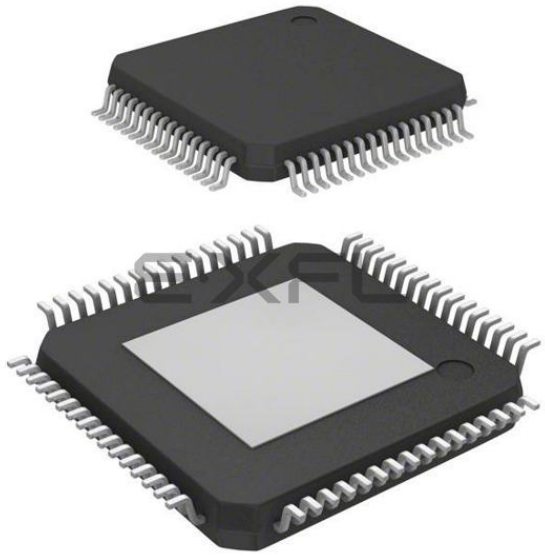

Welcome to E-XFL.COM

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	S12Z
Core Size	16-Bit
Speed	32MHz
Connectivity	CANbus, I²C, SCI, SPI
Peripherals	DMA, POR, PWM, WDT
Number of I/O	42
Program Memory Size	192KB (192K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	3.5V ~ 40V
Data Converters	A/D 16x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP Exposed Pad
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s912zvc19f0mkh

Table 1-6. MC9S12ZVC-Family Pin Summary

LQFP			Function					Power Supply	Internal Pull Resistor	
64	48	Pin	1st Func.	2nd Func.	3rd Func.	4th Func.	5th Func.		CTRL	Reset State
1	1	VSUP	—	—	—	—	—	V_{SUP}	—	—
2	2	PL1	HVI1	KWL1	—	—	—	V_{DDX}	—	—
3	3	PL0	HVI0	KWL0	—	—	—	V_{DDX}	—	—
4	4	BCTL			—	—	—	V_{DDX}		
5	5	BCTLC			—	—	—	V_{DDX}		
6	6	PE0	EXTAL		—	—	—	V_{DDX}	PERE/ PPSE	Down
7	7	PE1	XTAL	—	—	—	—	V_{DDX}	PERE/ PPSE	Down
8		PJ1	SCL0	TXD0	—	—	—	V_{DDX}	PERJ/ PPSJ	Up
9		PJ0	SDA0	RXD0	—	—	—	V_{DDX}	PERJ/ PPSJ	Up
10	—	PAD11	KWAD11	AN11	—	—	—	V_{DDA}	PERADL/ PPSADL	Off
11	—	PAD10	KWAD10	AN10	—	—	—	V_{DDA}	PERADL/ PPSADL	Off
12	8	PAD9	KWAD9	AN9	AMP	—	—	V_{DDA}	PERADL/ PPSADL	Off
13	9	PAD8	KWAD8	AN8	AMPM	—	—	V_{DDA}	PERADL/ PPSADL	Off
14	10	PAD7	KWAD7	AN7	AMPP	—	—	V_{DDA}	PERADH/ PPSADH	Off
15	11	PAD6	KWAD6	AN6	DACU	—	—	V_{DDA}	PERADH/ PPSADH	Off
16	12	PAD5	KWAD5	AN5	ACMPO1	—	—	V_{DDA}	PERADL/ PPSADL	Off
17	13	PAD4	KWAD4	AN4	ACMP1_1	—	—	V_{DDA}	PERADL/ PPSADL	Off
18	14	VSSA					—	V_{DDA}		
19	15	VDDA					—	V_{DDA}		
20	16	PAD3	KWAD3	AN3	VRH_0	ACMP1_0	—	V_{DDA}	PERADL/ PPSADL	Off
21	17	PAD2	KWAD2	AN2	ACMPO0	—	—	V_{DDA}	PERADL/ PPSADL	Off

The POR circuit monitors the VDD and VDDA domains, ensuring a reset assertion until an adequate voltage level is attained. The LVR circuit monitors the VDD, VDDF and VDDX domains, generating a reset when the voltage in any of these domains drops below the specified assert level. The VDDX LVR monitor is disabled when the VREG is in reduced power mode. A low voltage interrupt circuit monitors the VDDA domain.

Global Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0315–0x031E	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x031F	WOMJ	R	0	0	0	0	0	0	WOMJ1	WOMJ0
		W								
0x0320–0x032F	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0330	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0331	PTIL	R	0	0	0	0	0	0	PTIL1	PTIL0
		W								
0x0332	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0333	PTPSL	R	0	0	0	0	0	0	PTPSL1	PTPSL0
		W								
0x0334	PPSL	R	0	0	0	0	0	0	PPSL1	PPSL0
		W								
0x0335	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0336	PIEL	R	0	0	0	0	0	0	PIEL1	PIEL0
		W								
0x0337	PIFL	R	0	0	0	0	0	0	PIFL1	PIFL0
		W								
0x0338–0x0339	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x033A	PTABYPL	R	0	0	0	0	0	0	PTABYPL1	PTABYPL0
		W								
0x033B	PTADIRL	R	0	0	0	0	0	0	PTADIRL1	PTADIRL0
		W								
0x033C	DIENL	R	0	0	0	0	0	0	DIENL1	DIENL0
		W								
0x033D	PTAENL	R	0	0	0	0	0	0	PTAENL1	PTAENL0
		W								

Table 2-34. Effect of Enabled Features

Enabled Feature[1]	Related Signal(s)	Effect on I/O state	Effect on enabled pull device
SENT	SENT_TX	Forced output	Forced off if push-pull, pulldown forced off if open-drain
IRQ	$\overline{\text{IRQ}}$	Forced input	None[3]
XIRQ	$\overline{\text{XIRQ}}$	Forced input	None[3]
MSCANx	TXCANx	Forced output	Forced off
	RXCANx	Forced input	Pulldown forced off
CANPHYx	CPTXDx	Forced input	None[3]
	CPRXDx	Forced output	Forced off, pulldown forced off if open-drain

1 If applicable the appropriate routing configuration must be set for the signals to take effect on the pins.

2 DDR maintains control

3 PER/PPS maintain control

4 To use the digital input function the related bit in Digital Input Enable Register (DIENADH/L) must be set to logic level “1”.

2.4.3 Pin I/O Control

Figure 2-33 illustrates the data paths to and from an I/O pin. Input and output data can always be read via the input register (PTIx, Section 2.3.3.2, “Port Input Register”) independent if the pin is used as general-purpose I/O or with a shared peripheral function. If the pin is configured as input (DDRx=0, Section 2.3.3.3, “Data Direction Register”), the pin state can also be read through the data register (PTx, Section 2.3.3.1, “Port Data Register”).

The general-purpose data direction configuration can be overruled by an enabled peripheral function shared on the same pin (Table 2-34). If more than one peripheral function is available and enabled at the same time, the highest ranked module according the predefined priority scheme in Table 2-2 will take precedence on the pin.

Table 5-6. INT_CFDATA0–7 Field Descriptions

Field	Description
2–0 PRIOLVL[2:0]	**Interrupt Request Priority Level Bits** — The PRIOLVL[2:0] bits configure the interrupt request priority level of the associated interrupt request. Out of reset all interrupt requests are enabled at the lowest active level (“1”). Please also refer to Table 5-7 for available interrupt request priority levels. **Note:** Write accesses to configuration data registers of unused interrupt channels are ignored and read accesses return all 0s. For information about what interrupt channels are used in a specific MCU, please refer to the Device Reference Manual for that MCU. **Note:** When non I-bit maskable request vectors are selected, writes to the corresponding INT_CFDATA registers are ignored and read accesses return all 0s. The corresponding vectors do not have configuration data registers associated with them. **Note:** Write accesses to the configuration register for the spurious interrupt vector request (vector base + 0x0001DC) are ignored and read accesses return 0x07 (request is handled by the CPU, PRIOLVL = 7).

Table 5-7. Interrupt Priority Levels

Priority	PRIOLVL2	PRIOLVL1	PRIOLVL0	Meaning
	0	0	0	Interrupt request is disabled
low	0	0	1	Priority level 1
	0	1	0	Priority level 2
	0	1	1	Priority level 3
	1	0	0	Priority level 4
	1	0	1	Priority level 5
	1	1	0	Priority level 6
high	1	1	1	Priority level 7

5.4 Functional Description

The INT module processes all exception requests to be serviced by the CPU module. These exceptions include interrupt vector requests and reset vector requests. Each of these exception types and their overall priority level is discussed in the subsections below.

5.4.1 S12Z Exception Requests

The CPU handles both reset requests and interrupt requests. The INT module contains registers to configure the priority level of each I-bit maskable interrupt request which can be used to implement an interrupt priority scheme. This also includes the possibility to nest interrupt requests. A priority decoder is used to evaluate the relative priority of pending interrupt requests.

5.4.2 Interrupt Prioritization

After system reset all I-bit maskable interrupt requests are configured to be enabled, are set up to be handled by the CPU and have a pre-configured priority level of 1. Exceptions to this rule are the non-maskable interrupt requests and the spurious interrupt vector request at (vector base + 0x0001DC)

7.2.2.4 ECC Debug Pointer Register (ECCDPTRH, ECCDPTRM, ECCDPTRL)

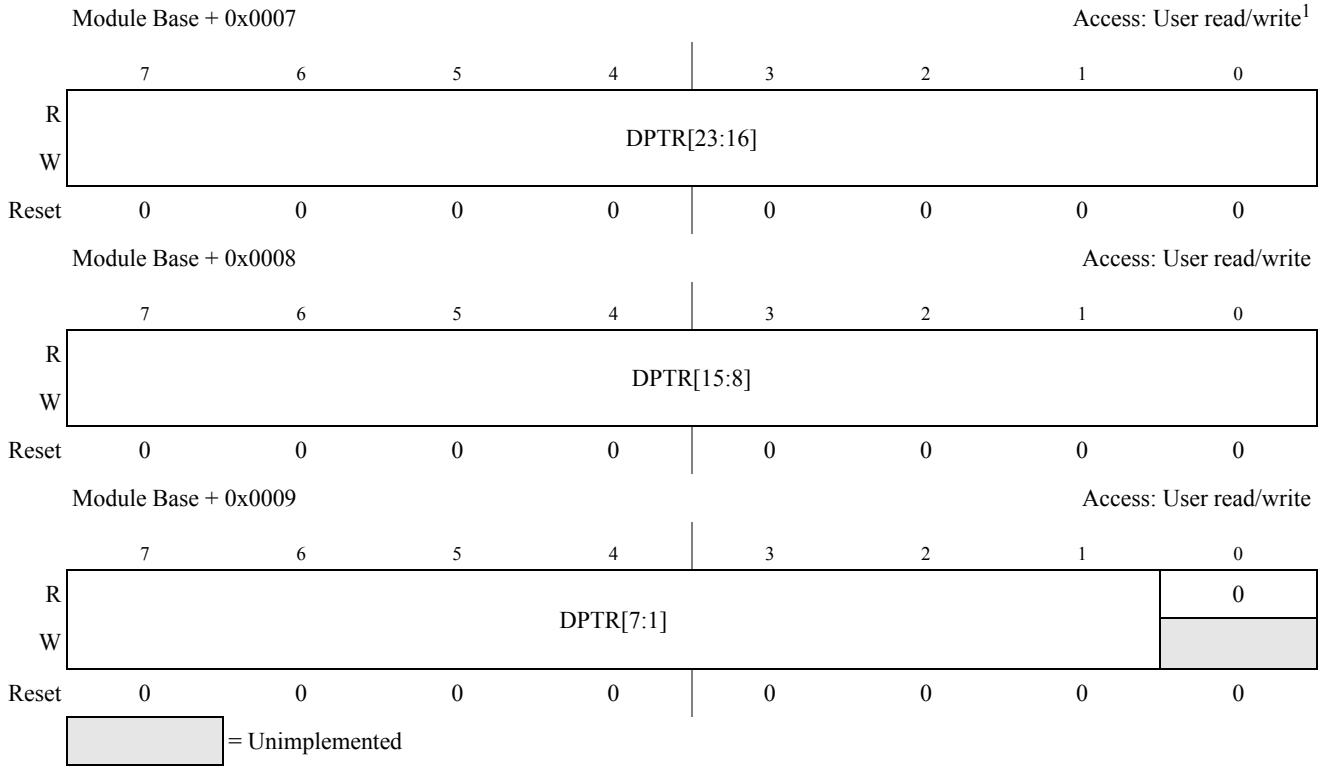

Figure 7-5. ECC Debug Pointer Register (ECCDPTRH, ECCDPTRM, ECCDPTRL)

1 Read: Anytime
Write: Anytime

Table 7-5. ECCDPTR Register Field Descriptions

Field	Description
DPTR [23:0]	**ECC Debug Pointer** — This register contains the system memory address which will be used for a debug access. Address bits not relevant for SRAM address space are not writeable, so the software should read back the pointer value to make sure the register contains the intended memory address. It is possible to write an address value to this register which points outside the system memory. There is no additional monitoring of the register content; therefore, the software must make sure that the address value points to the system memory space.

Table 8-1. CPMURFLG Field Descriptions (continued)

Field	Description
1 OMRF	**Oscillator Clock Monitor Reset Flag** — OMRF is set to 1 when a loss of oscillator (crystal) clock occurs. Refer to8.5.3, "Oscillator Clock Monitor Reset" for details.This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Loss of oscillator clock reset has not occurred. 1 Loss of oscillator clock reset has occurred.
0 PMRF	**PLL Clock Monitor Reset Flag** — PMRF is set to 1 when a loss of PLL clock occurs. This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Loss of PLL clock reset has not occurred. 1 Loss of PLL clock reset has occurred.

8.3.2.2 S12CPMU_UHV_V7 Synthesizer Register (CPMUSYNR)

The CPMUSYNR register controls the multiplication factor of the PLL and selects the VCO frequency range.

Module Base + 0x0004

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

Figure 8-5. S12CPMU_UHV_V7 Synthesizer Register (CPMUSYNR)

Read: Anytime

Write: If PROT=0 (CPMUPROT register) and PLLSEL=1 (CPMUCLKS register), then write anytime. Else write has no effect.

NOTE

Writing to this register clears the LOCK and UPOSC status bits.

If PLL has locked (LOCK=1) $f_{VCO} = 2 \times f_{REF} \times (SYNDIV + 1)$

NOTE

f_{VCO} must be within the specified VCO frequency lock range. Bus frequency f_{bus} must not exceed the specified maximum.

The VCOFRQ[1:0] bits are used to configure the VCO gain for optimal stability and lock time. For correct PLL operation the VCOFRQ[1:0] bits have to be selected according to the actual target VCOCLK

8.3.2.6 S12CPMU_UHV_V7 Interrupt Enable Register (CPMUINT)

This register enables S12CPMU_UHV_V7 interrupt requests.

Module Base + 0x0008

	7	6	5	4	3	2	1	0
R	RTIE	0	0	LOCKIE	0	0	OSCIE	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 8-9. S12CPMU_UHV_V7 Interrupt Enable Register (CPMUINT)

Read: Anytime

Write: Anytime

Table 8-5. CPMUINT Field Descriptions

Field	Description
7 RTIE	**Real Time Interrupt Enable Bit** 0 Interrupt requests from RTI are disabled. 1 Interrupt will be requested whenever RTIF is set.
4 LOCKIE	**PLL Lock Interrupt Enable Bit** 0 PLL LOCK interrupt requests are disabled. 1 Interrupt will be requested whenever LOCKIF is set.
1 OSCIE	**Oscillator Corrupt Interrupt Enable Bit** 0 Oscillator Corrupt interrupt requests are disabled. 1 Interrupt will be requested whenever OSCIF is set.

9.4.2.11 ADC Conversion Interrupt Enable Register (ADCCONIE)

Module Base + 0x000A

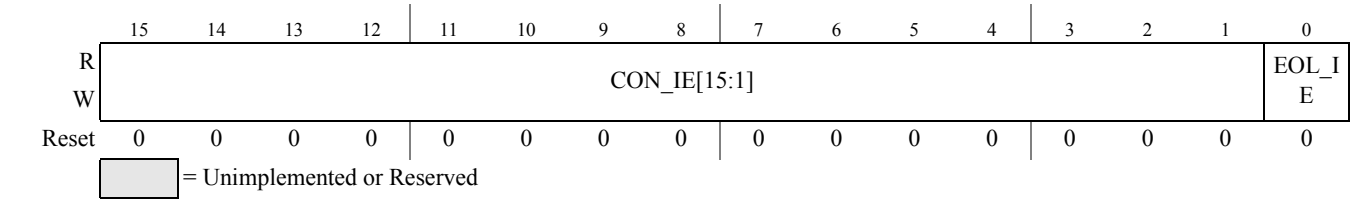

Figure 9-14. ADC Conversion Interrupt Enable Register (ADCCONIE)

Read: Anytime

Write: Anytime

Table 9-15. ADCCONIE Field Descriptions

Field	Description
15-1 CON_IE[15:1]	**Conversion Interrupt Enable Bits —** These bits enable the individual interrupts which can be triggered via interrupt flags CON_IF[15:1]. 0 ADC conversion interrupt disabled. 1 ADC conversion interrupt enabled.
0 EOL_IE	**End Of List Interrupt Enable Bit —** This bit enables the end of conversion sequence list interrupt. 0 End of list interrupt disabled. 1 End of list interrupt enabled.

9.6 Resets

At reset the ADC12B_LBA is disabled and in a power down state. The reset state of each individual bit is listed within Section 9.4.2, “Register Descriptions” which details the registers and their bit-fields.

9.7 Interrupts

The ADC supports three types of interrupts:

- Conversion Interrupt
- Sequence Abort Interrupt
- Error and Conversion Flow Control Issue Interrupt

Each of the interrupt types is associated with individual interrupt enable bits and interrupt flags.

9.7.1 ADC Conversion Interrupt

The ADC provides one conversion interrupt associated to 16 interrupt enable bits with dedicated interrupt flags. The 16 interrupt flags consist of:

- 15 conversion interrupt flags which can be associated to any conversion completion.
- One additional interrupt flag which is fixed to the “End Of List” conversion command type within the active CSL.

The association of the conversion number with the interrupt flag number is done in the conversion command.

9.7.2 ADC Sequence Abort Done Interrupt

The ADC provides one sequence abort done interrupt associated with the sequence abort request for conversion flow control. Hence, there is only one dedicated interrupt flag and interrupt enable bit for conversion sequence abort and it occurs when the sequence abort is done.

Table 11-11. TCTL3/TCTL4 Field Descriptions

Note: Writing to unavailable bits has no effect. Reading from unavailable bits return a zero.

Field	Description
7:0 EDGnB EDGnA	**Input Capture Edge Control —** These eight pairs of control bits configure the input capture edge detector circuits.

Table 11-12. Edge Detector Circuit Configuration

EDGnB	EDGnA	Configuration
0	0	Capture disabled
0	1	Capture on rising edges only
1	0	Capture on falling edges only
1	1	Capture on any edge (rising or falling)

11.3.2.10 Timer Interrupt Enable Register (TIE)

Module Base + 0x000C

	7	6	5	4	3	2	1	0
R W	C7I	C6I	C5I	C4I	C3I	C2I	C1I	C0I
Reset	0	0	0	0	0	0	0	0

Figure 11-18. Timer Interrupt Enable Register (TIE)

Read: Anytime

Write: Anytime.

Table 11-13. TIE Field Descriptions

Note: Writing to unavailable bits has no effect. Reading from unavailable bits return a zero

Field	Description
7:0 C7I:C0I	**Input Capture/Output Compare "x" Interrupt Enable —** The bits in TIE correspond bit-for-bit with the bits in the TFLG1 status register. If cleared, the corresponding flag is disabled from causing a hardware interrupt. If set, the corresponding flag is enabled to cause a interrupt.

NOTE

The newly selected prescale factor will not take effect until the next synchronized edge where all prescale counter stages equal zero.

11.3.2.12 Main Timer Interrupt Flag 1 (TFLG1)

Module Base + 0x000E

	7	6	5	4	3	2	1	0
R W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
Reset	0	0	0	0	0	0	0	0

Figure 11-20. Main Timer Interrupt Flag 1 (TFLG1)

Read: Anytime

Write: Used in the clearing mechanism (set bits cause corresponding bits to be cleared). Writing a zero will not affect current status of the bit.

Table 11-16. TRLG1 Field Descriptions

Note: Writing to unavailable bits has no effect. Reading from unavailable bits return a zero.

Field	Description
7:0 C[7:0]F	**Input Capture/Output Compare Channel "x" Flag** — These flags are set when an input capture or output compare event occurs. Clearing requires writing a one to the corresponding flag bit while TEN or PAEN is set to one. **Note:** When TFFCA bit in TSCR register is set, a read from an input capture or a write into an output compare channel (0x0010–0x001F) will cause the corresponding channel flag CxF to be cleared.

11.3.2.13 Main Timer Interrupt Flag 2 (TFLG2)

Module Base + 0x000F

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

Unimplemented or Reserved

Figure 11-21. Main Timer Interrupt Flag 2 (TFLG2)

TFLG2 indicates when interrupt conditions have occurred. To clear a bit in the flag register, write the bit to one while TEN bit of TSCR1 or PAEN bit of PACTL is set to one.

Read: Anytime

Write: Used in clearing mechanism (set bits cause corresponding bits to be cleared).

Any access to TCNT will clear TFLG2 register if the TFFCA bit in TSCR register is set.

11.3.2.15 16-Bit Pulse Accumulator Control Register (PACTL)

Module Base + 0x0020

	7	6	5	4	3	2	1	0
R	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
W								
Reset	0	0	0	0	0	0	0	0

Unimplemented or Reserved

Figure 11-24. 16-Bit Pulse Accumulator Control Register (PACTL)

Read: Any time

Write: Any time

When PAEN is set, the Pulse Accumulator counter is enabled. The Pulse Accumulator counter shares the input pin with IOC7.

Table 11-18. PACTL Field Descriptions

Field	Description
6 PAEN	**Pulse Accumulator System Enable** — PAEN is independent from TEN. With timer disabled, the pulse accumulator can function unless pulse accumulator is disabled. 0 16-Bit Pulse Accumulator system disabled. 1 Pulse Accumulator system enabled.
5 PAMOD	**Pulse Accumulator Mode** — This bit is active only when the Pulse Accumulator is enabled (PAEN = 1). See Table 11-19. 0 Event counter mode. 1 Gated time accumulation mode.
4 PEDGE	**Pulse Accumulator Edge Control** — This bit is active only when the Pulse Accumulator is enabled (PAEN = 1). For PAMOD bit = 0 (event counter mode). See Table 11-19. 0 Falling edges on IOC7 pin cause the count to be increased. 1 Rising edges on IOC7 pin cause the count to be increased. For PAMOD bit = 1 (gated time accumulation mode). 0 IOC7 input pin high enables M (Bus clock) divided by 64 clock to Pulse Accumulator and the trailing falling edge on IOC7 sets the PAIF flag. 1 IOC7 input pin low enables M (Bus clock) divided by 64 clock to Pulse Accumulator and the trailing rising edge on IOC7 sets the PAIF flag.
3:2 CLK[1:0]	**Clock Select Bits —** Refer to Table 11-20.
1 PAOVI	**Pulse Accumulator Overflow Interrupt Enable** 0 Interrupt inhibited. 1 Interrupt requested if PAOVF is set.
0 PAI	**Pulse Accumulator Input Interrupt Enable** 0 Interrupt inhibited. 1 Interrupt requested if PAIF is set.

Table 12-4. TSCR1 Field Descriptions

Field	Description
7 TEN	**Timer Enable** 0 Disables the main timer, including the counter. Can be used for reducing power consumption. 1 Allows the timer to function normally. If for any reason the timer is not active, there is no ÷64 clock for the pulse accumulator because the ÷64 is generated by the timer prescaler.
6 TSWAI	**Timer Module Stops While in Wait** 0 Allows the timer module to continue running during wait. 1 Disables the timer module when the MCU is in the wait mode. Timer interrupts cannot be used to get the MCU out of wait. TSWAI also affects pulse accumulator.
5 TSFRZ	**Timer Stops While in Freeze Mode** 0 Allows the timer counter to continue running while in freeze mode. 1 Disables the timer counter whenever the MCU is in freeze mode. This is useful for emulation. TSFRZ does not stop the pulse accumulator.
4 TFFCA	**Timer Fast Flag Clear All** 0 Allows the timer flag clearing to function normally. 1 For TFLG1(0x000E), a read from an input capture or a write to the output compare channel (0x0010–0x001F) causes the corresponding channel flag, CnF, to be cleared. For TFLG2 (0x000F), any access to the TCNT register (0x0004, 0x0005) clears the TOF flag. This has the advantage of eliminating software overhead in a separate clear sequence. Extra care is required to avoid accidental flag clearing due to unintended accesses.
3 PRNT	**Precision Timer** 0 Enables legacy timer. PR0, PR1, and PR2 bits of the TSCR2 register are used for timer counter prescaler selection. 1 Enables precision timer. All bits of the PTPSR register are used for Precision Timer Prescaler Selection, and all bits. This bit is writable only once out of reset.

12.3.2.5 Timer Toggle On Overflow Register 1 (TTOV)

Module Base + 0x0007

	7	6	5	4	3	2	1	0
R W	RESERVED	RESERVED	RESERVED	RESERVED	TOV3	TOV2	TOV1	TOV0
Reset	0	0	0	0	0	0	0	0

Figure 12-9. Timer Toggle On Overflow Register 1 (TTOV)

Read: Anytime

Write: Anytime

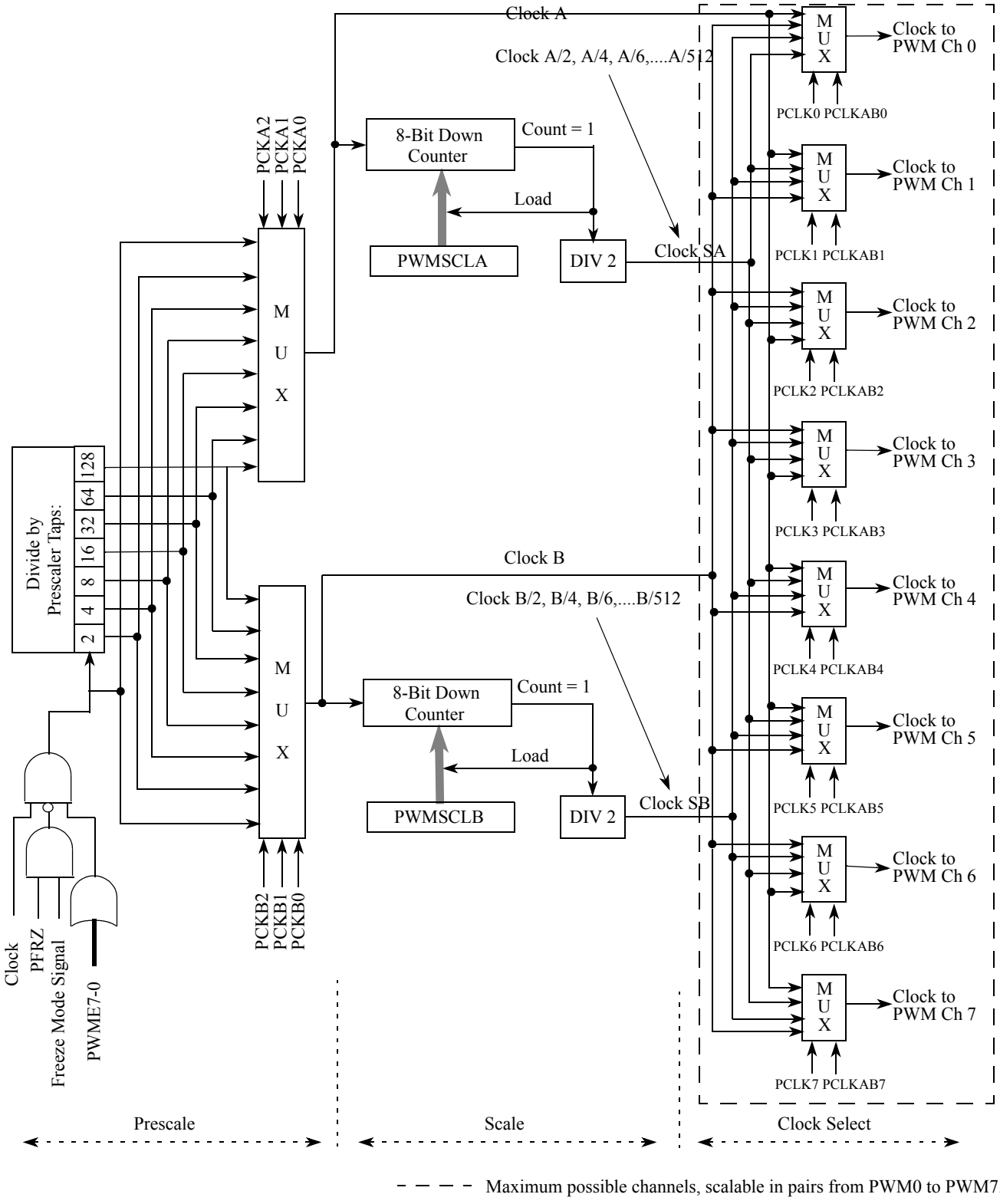

Figure 13-15. PWM Clock Select Block Diagram

14.3.2.9 SCI Data Registers (SCIDRH, SCIDRL)

Module Base + 0x0006

	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 14-12. SCI Data Registers (SCIDRH)

Module Base + 0x0007

	7	6	5	4	3	2	1	0
R	R7	R6	R5	R4	R3	R2	R1	R0
W	T7	T6	T5	T4	T3	T2	T1	T0
Reset	0	0	0	0	0	0	0	0

Figure 14-13. SCI Data Registers (SCIDRL)

Read: Anytime; reading accesses SCI receive data register

Write: Anytime; writing accesses SCI transmit data register; writing to R8 has no effect

NOTE

The reserved bit SCIDRH[2:0] are designed for factory test purposes only, and are not intended for general user access. Writing to these bit is possible when in special mode and can alter the modules functionality.

Table 14-13. SCIDRH and SCIDRL Field Descriptions

Field	Description
SCIDRH 7 R8	**Received Bit 8** — R8 is the ninth data bit received when the SCI is configured for 9-bit data format (M = 1).
SCIDRH 6 T8	**Transmit Bit 8** — T8 is the ninth data bit transmitted when the SCI is configured for 9-bit data format (M = 1).
SCIDRL 7:0 R[7:0] T[7:0]	**R7:R0** — Received bits seven through zero for 9-bit or 8-bit data formats **T7:T0** — Transmit bits seven through zero for 9-bit or 8-bit formats

NOTE

If the value of T8 is the same as in the previous transmission, T8 does not have to be rewritten.The same value is transmitted until T8 is rewritten

In 8-bit data format, only SCI data register low (SCIDRL) needs to be accessed.

When transmitting in 9-bit data format and using 8-bit write instructions, write first to SCI data register high (SCIDRH), then SCIDRL.

Table 16-8. IBCR Field Descriptions

Field	Description
7 IBEN	**I-Bus Enable** — This bit controls the software reset of the entire IIC bus module. 0 The module is reset and disabled. This is the power-on reset situation. When low the interface is held in reset but registers can be accessed 1 The IIC bus module is enabled.This bit must be set before any other IBCR bits have any effect If the IIC bus module is enabled in the middle of a byte transfer the interface behaves as follows: slave mode ignores the current transfer on the bus and starts operating whenever a subsequent start condition is detected. Master mode will not be aware that the bus is busy, hence if a start cycle is initiated then the current bus cycle may become corrupt. This would ultimately result in either the current bus master or the IIC bus module losing arbitration, after which bus operation would return to normal.
6 IBIE	**I-Bus Interrupt Enable** 0 Interrupts from the IIC bus module are disabled. Note that this does not clear any currently pending interrupt condition 1 Interrupts from the IIC bus module are enabled. An IIC bus interrupt occurs provided the IBIF bit in the status register is also set.
5 MS/SL	**Master/Slave Mode Select Bit** — Upon reset, this bit is cleared. When this bit is changed from 0 to 1, a START signal is generated on the bus, and the master mode is selected. When this bit is changed from 1 to 0, a STOP signal is generated and the operation mode changes from master to slave.A STOP signal should only be generated if the IBIF flag is set. MS/$\overline{SL}$ is cleared without generating a STOP signal when the master loses arbitration. 0 Slave Mode 1 Master Mode
4 Tx/Rx	**Transmit/Receive Mode Select Bit** — This bit selects the direction of master and slave transfers. When addressed as a slave this bit should be set by software according to the SRW bit in the status register. In master mode this bit should be set according to the type of transfer required. Therefore, for address cycles, this bit will always be high. 0 Receive 1 Transmit
3 TXAK	**Transmit Acknowledge Enable** — This bit specifies the value driven onto SDA during data acknowledge cycles for both master and slave receivers. The IIC module will always acknowledge address matches, provided it is enabled, regardless of the value of TXAK. Note that values written to this bit are only used when the IIC bus is a receiver, not a transmitter. 0 An acknowledge signal will be sent out to the bus at the 9th clock bit after receiving one byte data 1 No acknowledge signal response is sent (i.e., acknowledge bit = 1)
2 RSTA	**Repeat Start** — Writing a 1 to this bit will generate a repeated START condition on the bus, provided it is the current bus master. This bit will always be read as a low. Attempting a repeated start at the wrong time, if the bus is owned by another master, will result in loss of arbitration. 1 Generate repeat start cycle
1 RESERVED	**Reserved** — Bit 1 of the IBCR is reserved for future compatibility. This bit will always read 0.
0 IBSWAI	**I Bus Interface Stop in Wait Mode** 0 IIC bus module clock operates normally 1 Halt IIC bus module clock generation in wait mode

Wait mode is entered via execution of a CPU WAI instruction. In the event that the IBSWAI bit is set, all clocks internal to the IIC will be stopped and any transmission currently in progress will halt.If the CPU were woken up by a source other than the IIC module, then clocks would restart and the IIC would resume from where was during the previous transmission. It is not possible for the IIC to wake up the CPU when its internal clocks are stopped.

If it were the case that the IBSWAI bit was cleared when the WAI instruction was executed, the IIC internal clocks and interface would remain alive, continuing the operation which was currently underway. It is also

[1] Read: Anytime
Write: Anytime when not in initialization mode, except RSTAT[1:0] and TSTAT[1:0] flags which are read-only; write of 1 clears flag; write of 0 is ignored

NOTE

The CANRFLG register is held in the reset state[1] when the initialization mode is active (INITRQ = 1 and INITAK = 1). This register is writable again as soon as the initialization mode is exited (INITRQ = 0 and INITAK = 0).

Table 18-10. CANRFLG Register Field Descriptions

Field	Description
7 WUPIF	**Wake-Up Interrupt Flag** — If the MSCAN detects CAN bus activity while in sleep mode (see Section 18.4.5.5, "MSCAN Sleep Mode,") and WUPE = 1 in CANTCTL0 (see Section 18.3.2.1, "MSCAN Control Register 0 (CANCTL0)"), the module will set WUPIF. If not masked, a wake-up interrupt is pending while this flag is set. 0 No wake-up activity observed while in sleep mode 1 MSCAN detected activity on the CAN bus and requested wake-up
6 CSCIF	**CAN Status Change Interrupt Flag** — This flag is set when the MSCAN changes its current CAN bus status due to the actual value of the transmit error counter (TEC) and the receive error counter (REC). An additional 4-bit (RSTAT[1:0], TSTAT[1:0]) status register, which is split into separate sections for TEC/REC, informs the system on the actual CAN bus status (see Section 18.3.2.6, "MSCAN Receiver Interrupt Enable Register (CANRIER)"). If not masked, an error interrupt is pending while this flag is set. CSCIF provides a blocking interrupt. That guarantees that the receiver/transmitter status bits (RSTAT/TSTAT) are only updated when no CAN status change interrupt is pending. If the TECs/RECs change their current value after the CSCIF is asserted, which would cause an additional state change in the RSTAT/TSTAT bits, these bits keep their status until the current CSCIF interrupt is cleared again. 0 No change in CAN bus status occurred since last interrupt 1 MSCAN changed current CAN bus status
5-4 RSTAT[1:0]	**Receiver Status Bits** — The values of the error counters control the actual CAN bus status of the MSCAN. As soon as the status change interrupt flag (CSCIF) is set, these bits indicate the appropriate receiver related CAN bus status of the MSCAN. The coding for the bits RSTAT1, RSTAT0 is: 00 RxOK: 0 ≤ receive error counter < 96 01 RxWRN: 96 ≤ receive error counter < 128 10 RxERR: 128 ≤ receive error counter *11 Bus-off*[1]*: 256 ≤ transmit error counter*
3-2 TSTAT[1:0]	**Transmitter Status Bits** — The values of the error counters control the actual CAN bus status of the MSCAN. As soon as the status change interrupt flag (CSCIF) is set, these bits indicate the appropriate transmitter related CAN bus status of the MSCAN. The coding for the bits TSTAT1, TSTAT0 is: 00 TxOK: 0 ≤ transmit error counter < 96 01 TxWRN: 96 ≤ transmit error counter < 128 10 TxERR: 128 ≤ transmit error counter < 256 11 Bus-Off: 256 ≤ transmit error counter

1. The RSTAT[1:0], TSTAT[1:0] bits are not affected by initialization mode.

P-Flash memory (see Table 22-3) as indicated by reset condition F in Table 22-24.. To change the EEPROM protection that will be loaded during the reset sequence, the P-Flash sector containing the EEPROM protection byte must be unprotected, then the EEPROM protection byte must be programmed. If a double bit fault is detected while reading the P-Flash phrase containing the EEPROM protection byte during the reset sequence, the DPOPEN bit will be cleared and DPS bits will be set to leave the EEPROM memory fully protected.

Trying to alter data in any protected area in the EEPROM memory will result in a protection violation error and the FPVIOL bit will be set in the FSTAT register. Block erase of the EEPROM memory is not possible if any of the EEPROM sectors are protected.

Table 22-23. DFPROT Field Descriptions

Field	Description
7 DPOPEN	**EEPROM Protection Control** 0 Enables EEPROM memory protection from program and erase with protected address range defined by DPS bits 1 Disables EEPROM memory protection from program and erase
5–0 DPS[5:0]	**EEPROM Protection Size** — The DPS[5:0] bits determine the size of the protected area in the EEPROM memory, this size increase in step of 32 bytes, as shown in Table 22-24. .

Table 22-24. EEPROM Protection Address Range

DPS[5:0]	Global Address Range	Protected Size
000000	0x10_0000 – 0x10_001F	32 bytes
000001	0x10_0000 – 0x10_003F	64 bytes
000010	0x10_0000 – 0x10_005F	96 bytes
000011	0x10_0000 – 0x10_007F	128 bytes
000100	0x10_0000 – 0x10_009F	160 bytes
000101	0x10_0000 – 0x10_00BF	192 bytes
The Protection Size goes on enlarging in step of 32 bytes, for each DPS value increasing of one. . . .		
111111	0x10_0000 – 0x10_07FF	2,048 bytes

22.3.2.11 Flash Option Register (FOPT)

The FOPT register is the Flash option register.

N.23 0x0800-0x083F CAN0

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0800 CANCTL0	R	RXFRM	RXACT	CSWAI	SYNCH	TIME	WUPE	SLPRQ	INITRQ
	W								
0x0001 CANCTL1	R	CANE	CLKSRC	LOOPB	LISTEN	BORM	WUPM	SLPAK	INITAK
	W								
0x0802 CANBTR0	R	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
	W								
0x0803 CANBTR1	R	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
	W								
0x0804 CANRFLG	R	WUPIF	CSCIF	RSTAT1	RSTAT0	TSTAT1	TSTAT0	OVRIF	RXF
	W								
0x0805 CANRIER	R	WUPIE	CSCIE	RSTATE1	RSTATE0	TSTATE1	TSTATE0	OVRIE	RXFIE
	W								
0x0806 CANTFLG	R	0	0	0	0	0	TXE2	TXE1	TXE0
	W								
0x0807 CANTIER	R	0	0	0	0	0	TXEIE2	TXEIE1	TXEIE0
	W								
0x0808 CANTARQ	R	0	0	0	0	0	ABTRQ2	ABTRQ1	ABTRQ0
	W								
0x0809 CANTAAK	R	0	0	0	0	0	ABTAK2	ABTAK1	ABTAK0
	W								
0x080A CANTBSEL	R	0	0	0	0	0	TX2	TX1	TX0
	W								
0x080B CANIDAC	R	0	0	IDAM1	IDAM0	0	IDHIT2	IDHIT1	IDHIT0
	W								
0x080C Reserved	R	0	0	0	0	0	0	0	0
	W								
0x000D CANMISC	R	0	0	0	0	0	0	0	BOHOLD
	W								
0x080E CANRXERR	R	RXERR7	RXERR6	RXERR5	RXERR4	RXERR3	RXERR2	RXERR1	RXERR0
	W								

= Unimplemented or Reserved