

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	Obsolete
Core Processor	S12Z
Core Size	16-Bit
Speed	50MHz
Connectivity	CANbus, SCI, SPI
Peripherals	DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	3.5V ~ 40V
Data Converters	A/D 9x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP Exposed Pad
Supplier Device Package	64-HLQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s912zvmc12f2vkh

The exposed pad on the package bottom must be connected to a grounded contact pad on the PCB.

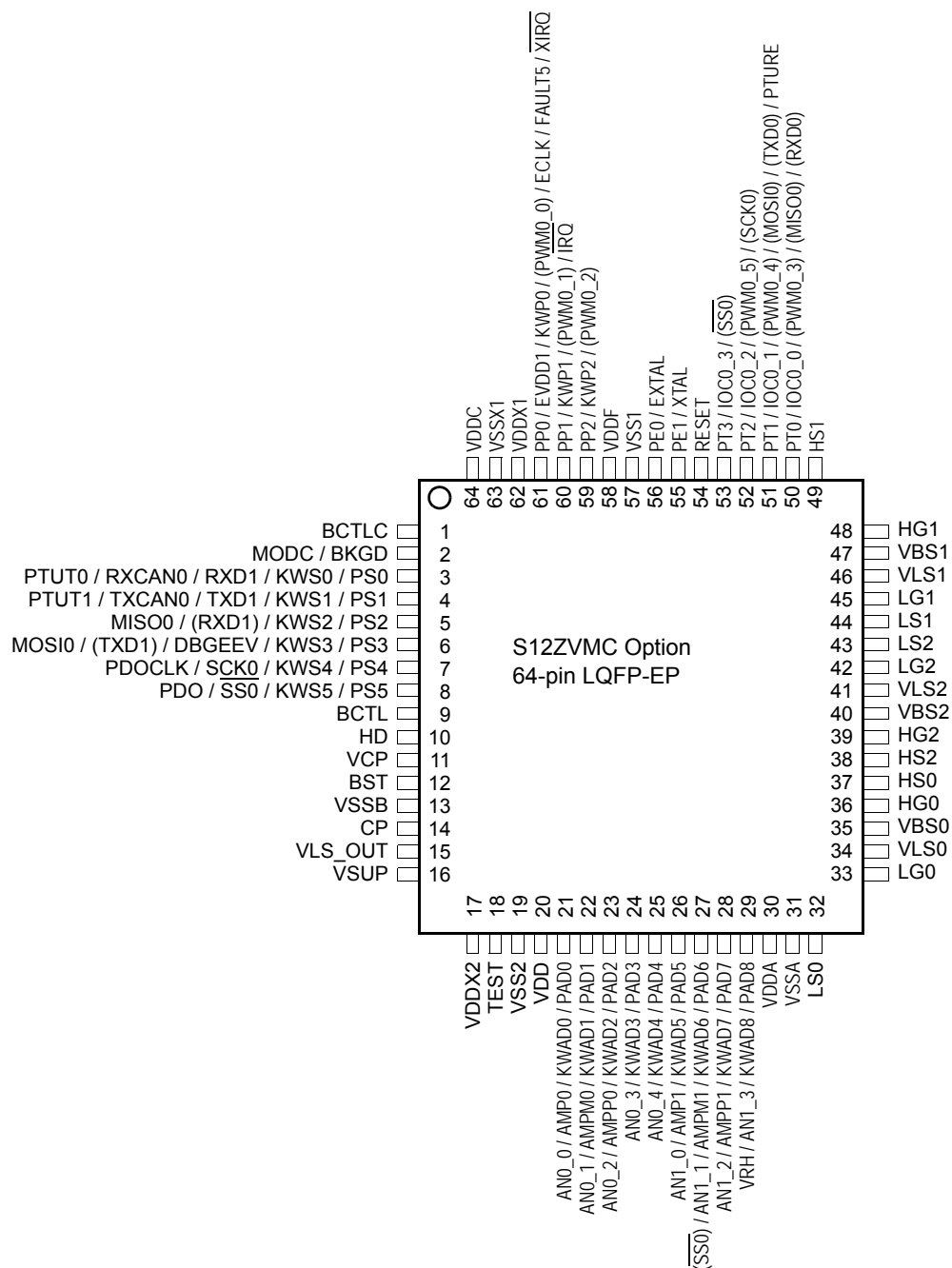


Figure 1-5. S12ZVMC Option 64-pin LQFP pin out

7.2.2 Register Descriptions

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

7.2.2.1 ECC Status Register (ECCSTAT)

Module Base + 0x00000				Access: User read only ⁽¹⁾				
	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	RDY
W								
Reset	0	0	0	0	0	0	0	0

1. Read: Anytime
Write: Never

Figure 7-2. ECC Status Register (ECCSTAT)

Table 7-2. ECCSTAT Field Description

Field	Description
0 RDY	ECC Ready— Shows the status of the ECC module. 0 Internal SRAM initialization is ongoing, access to the SRAM is disabled 1 Internal SRAM initialization is done, access to the SRAM is enabled

7.2.2.2 ECC Interrupt Enable Register (ECCIE)

Module Base + 0x00001				Access: User read/write ⁽¹⁾				
	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	SBEEIE
W								
Reset	0	0	0	0	0	0	0	0

1. Read: Anytime
Write: Anytime

Figure 7-3. ECC Interrupt Enable Register (ECCIE)

Table 7-3. ECCIE Field Description

Field	Description
0 SBEEIE	Single bit ECC Error Interrupt Enable — Enables Single ECC Error interrupt. 0 Interrupt request is disabled 1 Interrupt will be requested whenever SBEEIF is set

7.2.2.5 ECC Debug Data (ECCDDH, ECCDDL)

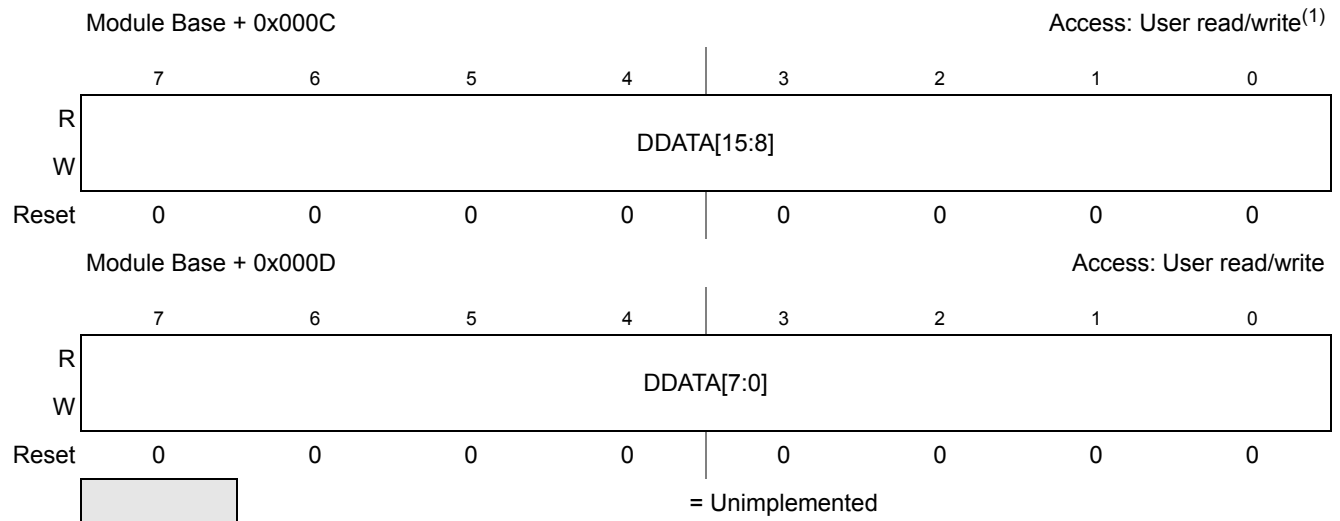


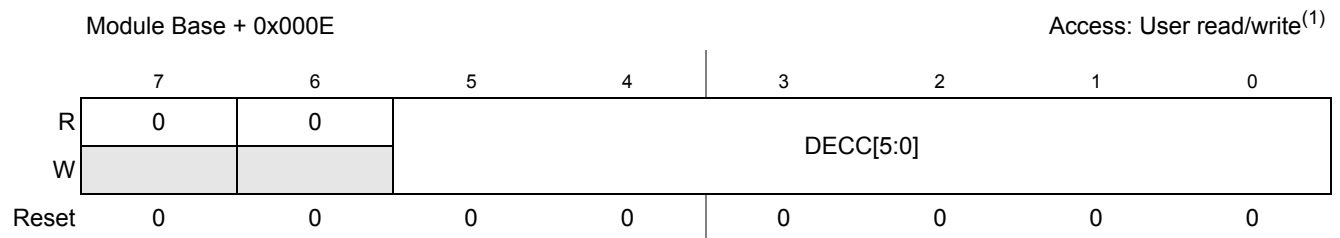
Figure 7-6. ECC Debug Data (ECCDDH, ECCDDL)

1. Read: Anytime
Write: Anytime

Table 7-6. ECCDD Register Field Descriptions

Field	Description
DDATA [23:0]	ECC Debug Raw Data — This register contains the raw data which will be written into the system memory during a debug write command or the read data from the debug read command.

7.2.2.6 ECC Debug ECC (ECCDE)



1. Read: Anytime
Write: Anytime

Figure 7-7. ECC Debug ECC (ECCDE)

Table 7-7. ECCDE Field Description

Field	Description
5:0 DECC[5:0]	ECC Debug ECC — This register contains the raw ECC value which will be written into the system memory during a debug write command or the ECC read value from the debug read command.

8.3.2.6 S12CPMU_UHV_V10_V6 Post Divider Register (CPMUPOSTDIV)

The POSTDIV register controls the frequency ratio between the VCOCLK and the PLLCLK.

Module Base + 0x0006

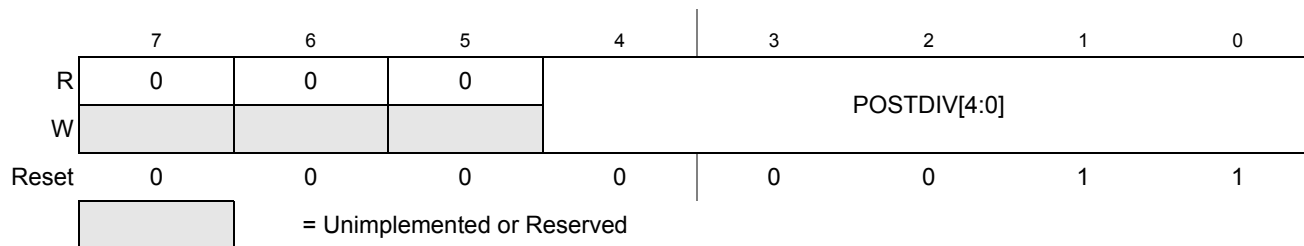


Figure 8-11. S12CPMU_UHV_V10_V6 Post Divider Register (CPMUPOSTDIV)

Read: Anytime

Write: If PLLSEL=1 write anytime, else write has no effect

$$\text{If PLL is locked (LOCK=1)} \quad f_{\text{PLL}} = \frac{f_{\text{VCO}}}{(\text{POSTDIV} + 1)}$$

$$\text{If PLL is not locked (LOCK=0)} \quad f_{\text{PLL}} = \frac{f_{\text{VCO}}}{4}$$

$$\text{If PLL is selected (PLLSEL=1)} \quad f_{\text{bus}} = \frac{f_{\text{PLL}}}{2}$$

When changing the POSTDIV[4:0] value or PLL transitions to locked stated (lock=1), it takes up to 32 Bus Clock cycles until f_{PLL} is at the desired target frequency. This is because the post divider gradually changes (increases or decreases) f_{PLL} in order to avoid sudden load changes for the on-chip voltage regulator.

8.3.2.7 S12CPMU_UHV_V10_V6 Interrupt Flags Register (CPMUIFLG)

This register provides S12CPMU_UHV_V10_V6 status bits and interrupt flags.

Module Base + 0x0007

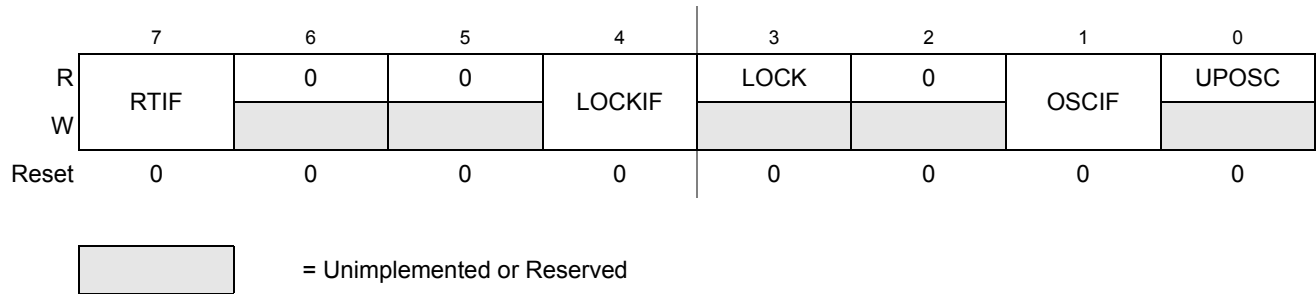


Figure 8-12. S12CPMU_UHV_V10_V6 Flags Register (CPMUIFLG)

Read: Anytime

Write: Refer to each bit for individual write conditions

Table 8-5. CPMUIFLG Field Descriptions

Field	Description
7 RTIF	Real Time Interrupt Flag — RTIF is set to 1 at the end of the RTI period. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (RTIE=1), RTIF causes an interrupt request. 0 RTI time-out has not yet occurred. 1 RTI time-out has occurred.
4 LOCKIF	PLL Lock Interrupt Flag — LOCKIF is set to 1 when LOCK status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (LOCKIE=1), LOCKIF causes an interrupt request. 0 No change in LOCK bit. 1 LOCK bit has changed.
3 LOCK	Lock Status Bit — LOCK reflects the current state of PLL lock condition. Writes have no effect. While PLL is unlocked (LOCK=0) f_{PLL} is $f_{VCO} / 4$ to protect the system from high core clock frequencies during the PLL stabilization time t_{lock} . 0 VCOCLK is not within the desired tolerance of the target frequency. $f_{PLL} = f_{VCO}/4$. 1 VCOCLK is within the desired tolerance of the target frequency. $f_{PLL} = f_{VCO}/(POSTDIV+1)$.
1 OSCIF	Oscillator Interrupt Flag — OSCIF is set to 1 when UPOSC status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (OSCIE=1), OSCIF causes an interrupt request. 0 No change in UPOSC bit. 1 UPOSC bit has changed.
0 UPOSC	Oscillator Status Bit — UPOSC reflects the status of the oscillator. Writes have no effect. Entering Full Stop Mode UPOSC is cleared. 0 The oscillator is off or oscillation is not qualified by the PLL. 1 The oscillator is qualified by the PLL.

Table 8-7. CPMUCLKS Descriptions (continued)

Field	Description
1 RTIOSCSEL	RTI Clock Select — RTIOSCSEL selects the clock source to the RTI. Either IRCCLK or OSCCLK. Changing the RTIOSCSEL bit re-starts the RTI time-out period. RTIOSCSEL can only be set to 1, if UPOSC=1. UPOSC= 0 clears the RTIOSCSEL bit. 0 RTI clock source is IRCCLK. 1 RTI clock source is OSCCLK.
0 COP OSCSEL0	COP Clock Select 0 — COPOSCSEL0 and COPOSCSEL1 combined determine the clock source to the COP (see also Table 8-8) If COPOSCSEL1 = 1, COPOSCSEL0 has no effect regarding clock select and changing the COPOSCSEL0 bit does not re-start the COP time-out period. When COPOSCSEL1=0,COPOSCSEL0 selects the clock source to the COP to be either IRCCLK or OSCCLK. Changing the COPOSCSEL0 bit re-starts the COP time-out period. COPOSCSEL0 can only be set to 1, if UPOSC=1. UPOSC= 0 clears the COPOSCSEL0 bit. 0 COP clock source is IRCCLK. 1 COP clock source is OSCCLK

Table 8-8. COPOSCSEL1, COPOSCSEL0 clock source select description

COPOSCSEL1	COPOSCSEL0	COP clock source
0	0	IRCCLK
0	1	OSCCLK
1	x	ACLK

Table 8-33. CPMUVDDS Field Descriptions (continued)

Field	Description
3 SCS2IF	Short circuit VDDS2 Interrupt Flag — SCS2IF is set to 1 when SCS2 status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (VDDSIIE = 1), SCS2IF causes an interrupt request. 0 No change in SCS2 bit. 1 SCS2 bit has changed.
2 SCS1IF	Short circuit VDDS1 Interrupt Flag — SCS1IF is set to 1 when SCS1 status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (VDDSIIE = 1), SCS1IF causes an interrupt request. 0 No change in SCS1 bit. 1 SCS1 bit has changed.
1 LVS2IF	Low-Voltage VDDS2 Interrupt Flag — LVS2IF is set to 1 when LVDS2 status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (VDDSIIE = 1), LVS2IF causes an interrupt request. 0 No change in LVDS2 bit. 1 LVDS2 bit has changed.
0 LVS1IF	Low-Voltage VDDS1 Interrupt Flag — LVS1IF is set to 1 when LVDS1 status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (VDDSIIE = 1), LVS1IF causes an interrupt request. 0 No change in LVDS1 bit. 1 LVDS1 bit has changed.

NOTE

The register ADCIMDRI is updated and simultaneously a conversion interrupt flag CON_IF[15:1] occurs when the corresponding conversion command (conversion command with INTFLG_SEL[3:0] set) has been processed and related data has been stored to RAM.

Chapter 13

Scalable Controller Area Network (S12MSCANV3)

Revision History

Revision Number	Revision Date	Sections Affected	Description of Changes
V03.14	12 Nov 2012	Table 13-10	<ul style="list-style-type: none">• Corrected RxWRN and TxWRN threshold values
V03.15	12 Jan 2013	Table 13-2 Table 13-25 Figure 13-37 13.1/13-477 13.3.2.15/13-499	<ul style="list-style-type: none">• Updated TIME bit description• Added register names to buffer map• Updated TSRH and TSRL read conditions• Updated introduction• Updated CANTXERR and CANRXERR register notes
V03.16	08 Aug 2013		<ul style="list-style-type: none">• Corrected typos

13.1 Introduction

Scalable controller area network (S12MSCANV3) definition is based on the MSCAN12 definition, which is the specific implementation of the MSCAN concept targeted for the S12, S12X and S12Z microcontroller families.

The module is a communication controller implementing the CAN 2.0A/B protocol as defined in the Bosch specification dated September 1991. For users to fully understand the MSCAN specification, it is recommended that the Bosch specification be read first to familiarize the reader with the terms and concepts contained within this document.

Though not exclusively intended for automotive applications, CAN protocol is designed to meet the specific requirements of a vehicle serial data bus: real-time processing, reliable operation in the EMI environment of a vehicle, cost-effectiveness, and required bandwidth.

MSCAN uses an advanced buffer arrangement resulting in predictable real-time behavior and simplified application software.

Table 13-5. Synchronization Jump Width

SJW1	SJW0	Synchronization Jump Width
0	0	1 Tq clock cycle
0	1	2 Tq clock cycles
1	0	3 Tq clock cycles
1	1	4 Tq clock cycles

Table 13-6. Baud Rate Prescaler

BRP5	BRP4	BRP3	BRP2	BRP1	BRP0	Prescaler value (P)
0	0	0	0	0	0	1
0	0	0	0	0	1	2
0	0	0	0	1	0	3
0	0	0	0	1	1	4
:	:	:	:	:	:	:
1	1	1	1	1	1	64

13.3.2.4 MSCAN Bus Timing Register 1 (CANBTR1)

The CANBTR1 register configures various CAN bus timing parameters of the MSCAN module.

Module Base + 0x0003

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R								
W								
	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
Reset:	0	0	0	0	0	0	0	0

Figure 13-7. MSCAN Bus Timing Register 1 (CANBTR1)

1. Read: Anytime

Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1)

Table 13-7. CANBTR1 Register Field Descriptions

Field	Description
7 SAMP	<p>Sampling — This bit determines the number of CAN bus samples taken per bit time.</p> <p>0 One sample per bit.</p> <p>1 Three samples per bit⁽¹⁾.</p> <p>If SAMP = 0, the resulting bit value is equal to the value of the single bit positioned at the sample point. If SAMP = 1, the resulting bit value is determined by using majority rule on the three total samples. For higher bit rates, it is recommended that only one sample is taken per bit time (SAMP = 0).</p>

13.4 Functional Description

13.4.1 General

This section provides a complete functional description of the MSCAN.

13.4.2 Message Storage

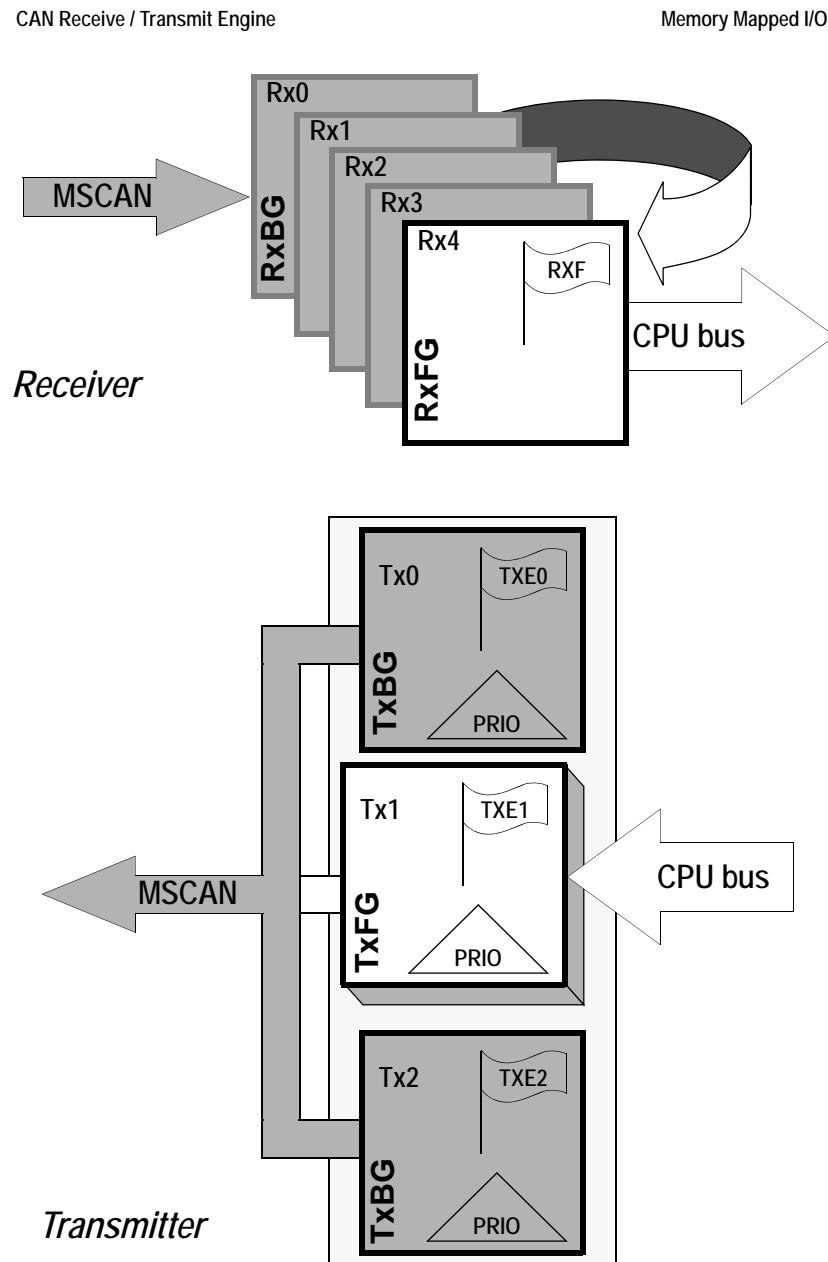


Figure 13-39. User Model for Message Buffer Organization

To generate a reload event or trigger event independent from the PWM status the debug register bits PTUFRE or TGxFTE can be used. A write one to this bits will generate the associated event. This behavior is not available during stop or freeze mode.

Address Offset	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x001B	PMFVAL5	R W	PMFVAL5							
0x001C	PMFROIE	R W	0	0	0	0	0	PMFROIE C	PMFROIE B	PMFROIE A
0x001D	PMFROIF	R W	0	0	0	0	0	PMFROIF C	PMFROIF B	PMFROIF A
0x001E	PMFICCTL	R W	0	0	PECC	PECB	PECA	ICCC	ICCB	ICCA
0x001F	PMFCINV	R W	0	0	CINV5	CINV4	CINV3	CINV2	CINV1	CINV0
0x0020	PMFENCA	R W	PWMENA	GLDOKA	0	0	0	RSTRTA	LDOKA	PWMRIEA
0x0021	PMFFQCA	R W	LDFQA				HALFA	PRSCA		PWMRFA
0x0022	PMFCNTA	R W	0	PMFCNTA						
0x0023	PMFCNTA	R W	PMFCNTA							
0x0024	PMFMODA	R W	0	PMFMODA						
0x0025	PMFMODA	R W	PMFMODA							
0x0026	PMFDTMA	R W	0	0	0	0	PMFDTMA			
0x0027	PMFDTMA	R W	PMFDTMA							
0x0028	PMFENCB	R W	PWMENB	GLDOKB	0	0	0	RSTRTB	LDOKB	PWMRIEB
0x0029	PMFFQCB	R W	LDFQB				HALFB	PRSCB		PWMRFB
				= Unimplemented or Reserved						

Figure 15-2. Quick Reference to PMF Registers (Sheet 3 of 5)

Table 15-16. PMFOUTC Field Descriptions

Field	Description
5–0 OUTCTL[5:0]	<p>OUTCTLn Bits — These bits enable software control of their corresponding PWM output. When OUTCTLn is set, the OUTn bit takes over the directly controls the level of the PWMn output.</p> <p>Note: OUTCTLn is buffered if ENCE is set. If ENCE is set, then the value written does not take effect until the next commutation cycle begins. Reading OUTCTLn returns the value in the buffer and not necessarily the value the output control is currently using. If ENCE is not set, then the OUTn bits take immediately effect when OUTCTLn bit is set. If the OUTCTLn bit is cleared then the OUTn control is disabled at the next PMF cycle start.</p> <p>When operating the PWM in complementary mode, these bits must be switched in pairs for proper operation. That is OUTCTL0 and OUTCTL1 must have the same value; OUTCTL2 and OUTCTL3 must have the same value; and OUTCTL4 and OUTCTL5 must have the same value. Otherwise see the behavior described on chapter Section 15.8.2, “BLDC 6-Step Commutation”.</p> <p>0 Software control disabled 1 Software control enabled n is 0, 1, 2, 3, 4 and 5.</p>

15.3.2.11 PMF Output Control Bit Register (PMFOUTB)

Address: Module Base + 0x000D

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R	0	0						
W			OUT5	OUT4	OUT3	OUT2	OUT1	OUT0
Reset	0	0	0	0	0	0	0	0

Figure 15-14. PMF Output Control Bit Register (PMFOUTB)

1. Read: Anytime
Write: Anytime

Table 15-17. PMFOUTB Field Descriptions

Field	Description
5–0 OUT[5:0]	<p>OUTn Bits — If the corresponding OUTCTLn bit is set, these bits control the PWM outputs, illustrated in Table 15-18.</p> <p>If the related OUTCTLn=1 a read returns the register contents OUTn else the current PWM output states are returned⁽¹⁾ On module version V3 the read returns always the register value.</p> <p>Note: OUTn is buffered if ENCE is set. The value written does not take effect until the next commutation cycle begins. Reading OUTn (with OUTCTLn=1) returns the value in the buffer and not necessarily the value the output control is currently using.</p> <p>n is 0, 1, 2, 3, 4 and 5.</p>

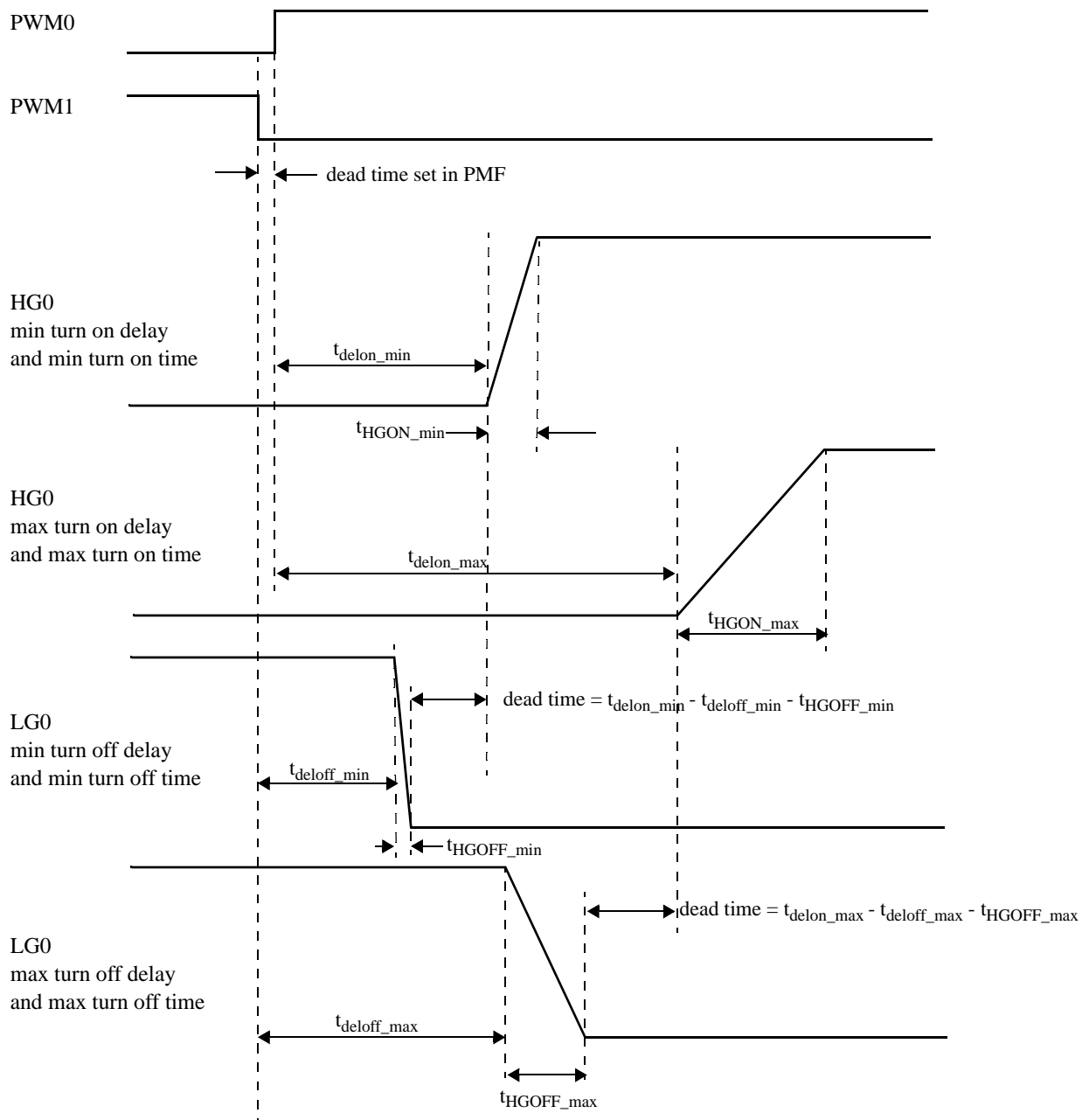
1. only valid for module version V4

1. Read: Anytime
Write: Only if GWP=0

Table 18-8. GDU Slew Rate Control Register Field Descriptions

Field	Description
6:4 GSRCHS[2:0]	<p>GDU Slew Rate Control Bits High-Side FET Pre-Drivers — These bits control the slew rate on the HG[2:0] pins (see FET Pre-Driver Details) .These bits cannot be modified after GWP bit is set.</p> <p>000 : slowest</p> <p>.</p> <p>.</p> <p>111 : fastest</p>
3:0 GSRCLS[2:0]	<p>GDU Slew Rate Control Bits Low-Side FET Pre-Drivers — These bits control the slew rate on the LG[2:0] pins (see FET Pre-Driver Details). These bits cannot be modified after GWP bit is set.</p> <p>000 : slowest</p> <p>.</p> <p>.</p> <p>111 : fastest</p>

Figure 18-31. Examples of Intrinsic Dead Time



The Flash memory may be read as bytes and aligned words. Read access time is one bus cycle for bytes and aligned words. For misaligned words access, the CPU has to perform twice the byte read access command. For Flash memory, an erased bit reads 1 and a programmed bit reads 0.

It is possible to read from P-Flash memory while some commands are executing on EEPROM memory. It is not possible to read from EEPROM memory while a command is executing on P-Flash memory . Simultaneous P-Flash and EEPROM operations are discussed in Section 20.4.6.

Both P-Flash and EEPROM memories are implemented with Error Correction Codes (ECC) that can resolve single bit faults and detect double bit faults. For P-Flash memory, the ECC implementation requires that programming be done on an aligned 8 byte basis (a Flash phrase). Since P-Flash memory is always read by half-phrase, only one single bit fault in an aligned 4 byte half-phrase containing the byte or word accessed will be corrected.

20.1.1 Glossary

Command Write Sequence — An MCU instruction sequence to execute built-in algorithms (including program and erase) on the Flash memory.

EEPROM Memory — The EEPROM memory constitutes the nonvolatile memory store for data.

EEPROM Sector — The EEPROM sector is the smallest portion of the EEPROM memory that can be erased. The EEPROM sector consists of 4 bytes.

NVM Command Mode — An NVM mode using the CPU to setup the FCCOB register to pass parameters required for Flash command execution.

Phrase — An aligned group of four 16-bit words within the P-Flash memory. Each phrase includes two sets of aligned double words with each set including 7 ECC bits for single bit fault correction and double bit fault detection within each double word.

P-Flash Memory — The P-Flash memory constitutes the main nonvolatile memory store for applications.

P-Flash Sector — The P-Flash sector is the smallest portion of the P-Flash memory that can be erased. Each P-Flash sector contains 512 bytes.

Program IFR — Nonvolatile information register located in the P-Flash block that contains the Version ID, and the Program Once field.

20.1.2 Features

20.1.2.1 P-Flash Features

- Derivatives featuring up to and including 128 KB of P-Flash include one P-Flash block
- Derivatives featuring more than 128 KB of P-Flash include two Flash blocks

21.4.2.5 Reserved Register

Module Base + 0x0004

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
W	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Reset	x	x	x	x	x	x	x	x

Figure 21-6. Reserved Register

1. Read: Anytime
Write: Only in special mode

NOTE

This reserved register is designed for factory test purposes only and is not intended for general user access. Writing to this register when in special modes can alter the modules functionality.

21.4.2.6 Reserved Register

Module Base + 0x0005

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
W	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Reset	x	x	x	x	x	x	x	x

Figure 21-7. Reserved Register

1. Read: Anytime
Write: Only in special mode

NOTE

This reserved register is designed for factory test purposes only and is not intended for general user access. Writing to this register when in special modes can alter the modules functionality.

21.4.2.7 CAN Physical Layer Interrupt Enable Register (CPIE)

Module Base + 0x0006

Access: User read/write⁽¹⁾

	7	6	5	4	3	2	1	0
R	0	0	0	CPVFIE	CPDTIE	0	0	CPOCIE
W								
Reset	0	0	0	0	0	0	0	0

Figure 21-8. CAN Physical Layer Interrupt Enable Register (CPIE)

1. Read: Anytime
Write: Anytime

NOTE

Changing the PWM output mode from left aligned to center aligned output (or vice versa) while channels are operating can cause irregularities in the PWM output. It is recommended to program the output mode before enabling the PWM channel.

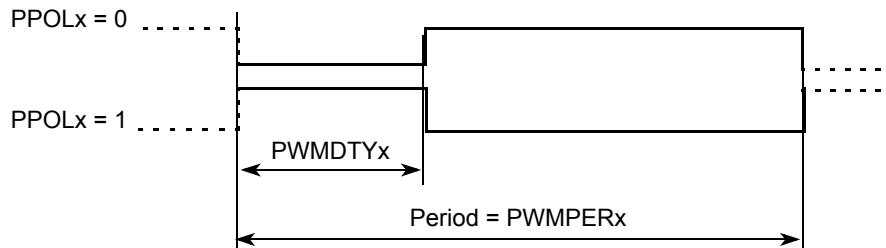


Figure 22-17. PWM Left Aligned Output Waveform

To calculate the output frequency in left aligned output mode for a particular channel, take the selected clock source frequency for the channel (A, B, SA, or SB) and divide it by the value in the period register for that channel.

- PWMx Frequency = Clock (A, B, SA, or SB) / PWMPERx
- PWMx Duty Cycle (high time as a% of period):
 - Polarity = 0 (PPOLx = 0)

$$\text{Duty Cycle} = [(PWMPERx - PWMDTYx) / PWMPERx] * 100\%$$
 - Polarity = 1 (PPOLx = 1)

$$\text{Duty Cycle} = [PWMDTYx / PWMPERx] * 100\%$$

As an example of a left aligned output, consider the following case:

Clock Source = bus clock, where bus clock = 10 MHz (100 ns period)

PPOLx = 0

PWMPERx = 4

PWMDTYx = 1

PWMx Frequency = 10 MHz / 4 = 2.5 MHz

PWMx Period = 400 ns

PWMx Duty Cycle = 3/4 * 100% = 75%

The output waveform generated is shown in Figure 22-18.

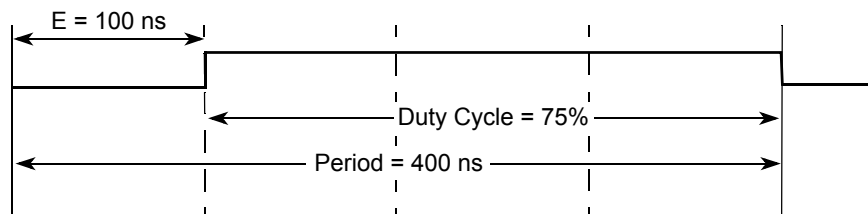


Figure 22-18. PWM Left Aligned Output Example Waveform

M.22 0x0980-0x0987 LINPHY0

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0980	LP0DR	R	0	0	0	0	0	0	LPDR1	LPDR0
		W								
0x0981	LP0CR	R	0	0	0	0	LPE	RXONLY	LPWUE	LPPUE
		W								
0x0982	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x0983	LP0SLRM	R	LPDTPDIS	0	0	0	0	0	LPSLR1	LPSLR0
		W								
0x0984	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x0985	LP0SR	R	LPDT	0	0	0	0	0	0	0
		W								
0x0986	LP0IE	R	LPDTIE	LPOCIE	0	0	0	0	0	0
		W								
0x0987	LP0IF	R	LPDTIF	LPOCIF	0	0	0	0	0	0
		W								

M.23 0x0990-0x0997 CANPHY

Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0990	CPDR	R	CPDR7	0	0	0	0	0	CPDR1	CPDR0
		W								
0x0991	CPCR	R	CPE	SPE	WUPE1-0	0	SLR2-0			
		W								
0x0992	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x0993	CPSR	R	CPCHVH	CPCHVL	CPCLVH	CPCLVL	CPDT	0	0	0
		W								
0x0994	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x0995	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x0996	CPIE	R	0	0	0	CPVFIE	CPDTIE	0	0	CPOCIE
		W								
0x0997	CPIF	R	CHVHIF	CHVLIF	CLVHIF	CLVLIF	CPDTIF	0	CHOCIF	CLOCIF
		W								

 = Unimplemented or Reserved