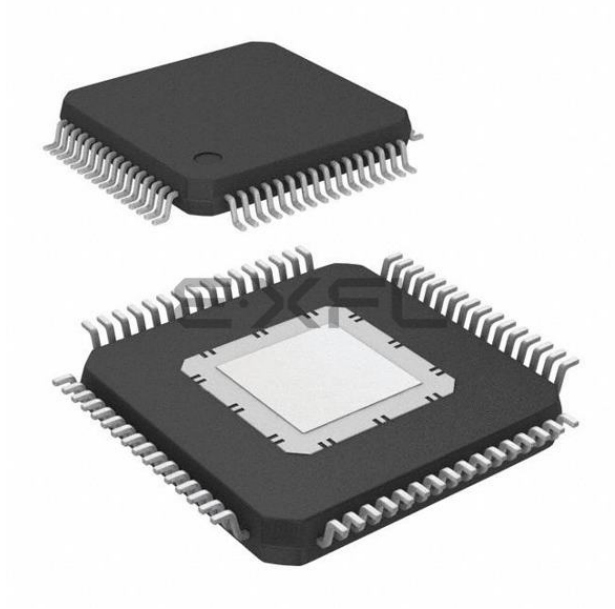


Welcome to E-XFL.COM

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"

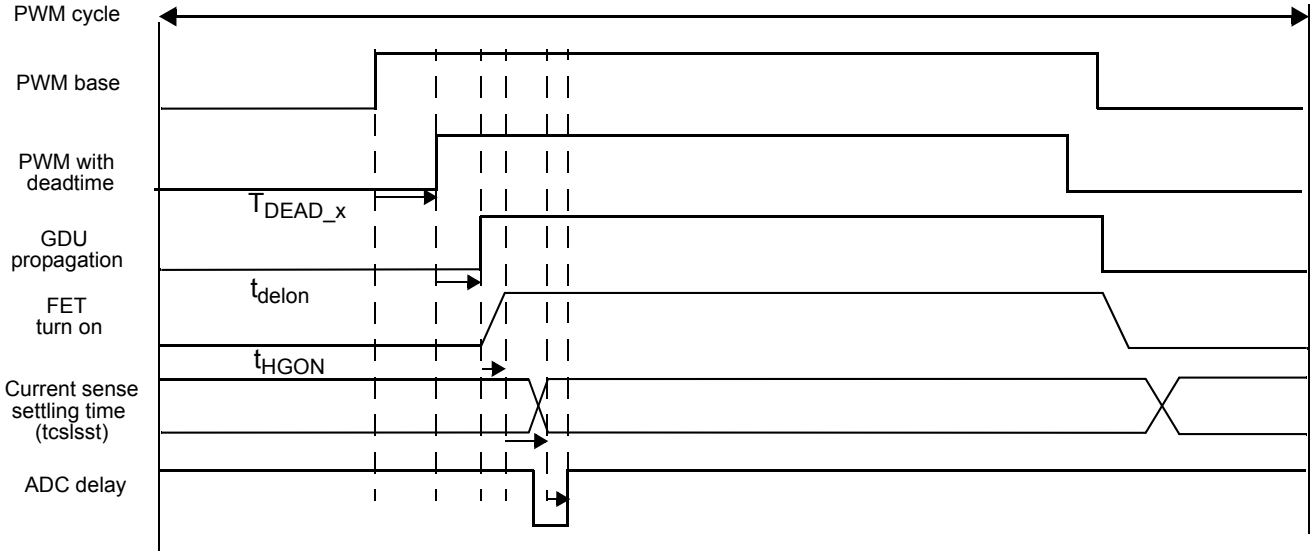
Details	
Product Status	Obsolete
Core Processor	S12Z
Core Size	16-Bit
Speed	50MHz
Connectivity	CANbus, LINbus, SCI, SPI
Peripherals	DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3.5V ~ 40V
Data Converters	A/D 9x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 150°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/s912zvml32f1wkhr

1.13.3.2 Control Loop Timing Considerations

Delays within the separate control loop elements require consideration to ensure correct synchronization.

Regarding the raw PWM0 signal as the starting point and stepping through the control loop stages, the factors shown in Figure 1-10 contribute to delays within the control loop, starting with the deadtime insertion, going through the external FETs and back into the internal ADC measurements of external voltages and currents.

Figure 1-10. Control Loop Delay Overview



The PWM deadtime (T_{DEAD_X}) is an integral number of bus clock cycles, configured by the PMF deadtime registers.

The GDU propagation delays (t_{delon}, t_{deloff}) are specified in the electrical parameter Table E-1.

The FET turn on times (t_{HGON}) are load dependent but are specified for particular loads in the electrical parameter Table E-1.

The current sense amplifier delay is highly dependent on external components.

The ADC delay until a result is available is specified as the conversion period N_{CONV} in Table C-1.

1.13.3.3 Static Timing Operation

The timing frame is static if it is the same in every control cycle (defined by reload frequency) and is relative to start of the control cycle. The only settings modified from one control cycle to the next one are the PWM duty cycle registers.

The main control cycle synchronization event is the PMF **reload** event. The PMF **reload** event can be generated every *n* PWM periods.
This mode can optionally be extended by a timer channel trigger to PMF to change the PWM channel operation (e.g. used for BLDCM commutation). In this case, the PMF configuration can propagate the

2.3.1 Register Map

Global Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0200	MODRR0	R	0	0	SPI0SSRR	SPI0RR	SCI1RR	S0L0RR2-0[1]		
		W								
0x0201	MODRR1	R	M0C0RR2-0[2]			PWMPRR1-0[3]		PWM54RR	PWM32RR	PWM10RR
		W								
0x0202	MODRR2	R	T0C2RR1-0[4]		T0C1RR[4]	T1IC0RR[2]	T0IC3RR1-0		T0IC1RR	T0IC1RR0[4]
		W								
0x0203–0x0207	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0208	ECLKCTL	R	NECLK	0	0	0	0	0	0	0
		W								
0x0209	IRQCR	R	IRQE	IRQEN	0	0	0	0	0	0
		W								
0x020A	PIMMISC	R	0	0	0	0	0	0	OCPE1	0
		W								
0x020B–0x020C	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x020D	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x020E	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x020F	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								
0x0210–0x025F	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0260	PTE	R	0	0	0	0	0	0	PTE1	PTE0
		W								

2.3.4.3 Port P Interrupt Flag Register (PIFP)

Address 0x02F7 PIFP Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	OCIF1	0	0	0	0	PIFP2	PIFP1	PIFP0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-25. Port P Interrupt Flag Register

1. Read: Anytime
   Write: Anytime, write 1 to clear

Table 2-29. Port P Interrupt Flag Register Field Descriptions

Field	Description
7 OCIF1	**Over-Current Interrupt Flag register —** This flag asserts if an over-current condition is detected on PP0 (Section 2.4.5, “Over-Current Interrupt”). Writing a logic “1” to the corresponding bit field clears the flag. 1 PP0 Over-current event occurred 0 No PP0 over-current event occurred
2-0 PIFP2-0	**See Section 2.3.3.7, “Port Interrupt Flag Register”**

Table 4-7. Interrupt Priority Levels

Priority	PRIOLVL2	PRIOLVL1	PRIOLVL0	Meaning
	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

4.4 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.

4.4.1 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.

4.4.2 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) which cannot be disabled, are always handled by the CPU and have a fixed priority levels. A priority level of 0 effectively disables the associated I-bit maskable interrupt request.

If more than one interrupt request is configured to the same interrupt priority level the interrupt request with the higher vector address wins the prioritization.

The following conditions must be met for an I-bit maskable interrupt request to be processed.

1. The local interrupt enabled bit in the peripheral module must be set.
2. The setup in the configuration register associated with the interrupt request channel must meet the following conditions:
   a) The priority level must be set to non zero.
   b) The priority level must be greater than the current interrupt processing level in the condition code register (CCW) of the CPU (PRIOLVL[2:0] > IPL[2:0]).
3. The I-bit in the condition code register (CCW) of the CPU must be cleared.
4. There is no access violation interrupt request pending.
5. There is no SYS, SWI, SPARE, TRAP, Machine Exception or $\overline{\text{XIRQ}}$ request pending.

If the ACK pulse handshake protocol is enabled and STEAL is cleared, then the BDC waits for the first free bus cycle to make a non-intrusive access. If no free bus cycle occurs within 512 core clock cycles then the BDC aborts the access, sets the NORESP bit and uses a long ACK pulse to indicate an error condition to the host.

Table 5-8 summarizes the BDC command set. The subsequent sections describe each command in detail and illustrate the command structure in a series of packets, each consisting of eight bit times starting with a falling edge. The bar across the top of the blocks indicates that the BKGD line idles in the high state. The time for an 8-bit command is 8 × 16 target BDCSI clock cycles.

The nomenclature below is used to describe the structure of the BDC commands. Commands begin with an 8-bit hexadecimal command code in the host-to-target direction (most significant bit first)

```
/        =        separates parts of the command
d        =        delay 16 target BDCSI clock cycles (DLY)
dack     =        delay (16 cycles) no ACK; or delay (=> 32 cycles) then ACK.(DACK)
ad24     =        24-bit memory address in the host-to-target direction
rd8      =        8 bits of read data in the target-to-host direction
rd16     =        16 bits of read data in the target-to-host direction
rd24     =        24 bits of read data in the target-to-host direction
rd32     =        32 bits of read data in the target-to-host direction
rd64     =        64 bits of read data in the target-to-host direction
rd.sz    =        read data, size defined by sz, in the target-to-host direction
wd8      =        8 bits of write data in the host-to-target direction
wd16     =        16 bits of write data in the host-to-target direction
wd32     =        32 bits of write data in the host-to-target direction
wd.sz    =        write data, size defined by sz, in the host-to-target direction
ss       =        the contents of BDCCSRL in the target-to-host direction
sz       =        memory operand size (00 = byte, 01 = word, 10 = long)
                  (sz = 11 is reserved and currently defaults to long)
crn      =        core register number, 32-bit data width
WS       =        command suffix signaling the operation is with status
```

Table 5-8. BDC Command Summary

Command Mnemonic	Command Classification	ACK	Command Structure	Description
SYNC	Always Available	N/A	N/A[1]	Request a timed reference pulse to determine the target BDC communication speed
ACK_DISABLE	Always Available	No	0x03/d	Disable the communication handshake. This command does not issue an ACK pulse.
ACK_ENABLE	Always Available	Yes	0x02/dack	Enable the communication handshake. Issues an ACK pulse after the command is executed.
BACKGROUND	Non-Intrusive	Yes	0x04/dack	Halt the CPU if ENBDC is set. Otherwise, ignore as illegal command.

Table 6-32 shows the effect for RWE and RW on the comparison conditions. These bits are ignored if INST is set, as matches based on instructions reaching the execution stage are data independent.

Table 6-32. Read or Write Comparison Logic Table

RWE Bit	RW Bit	RW Signal	Comment
0	x	0	RW not used in comparison
0	x	1	RW not used in comparison
1	0	0	Write match
1	0	1	No match
1	1	0	No match
1	1	1	Read match

6.3.2.17 Debug Comparator B Address Register (DBGBAH, DBGBAM, DBGBAL)

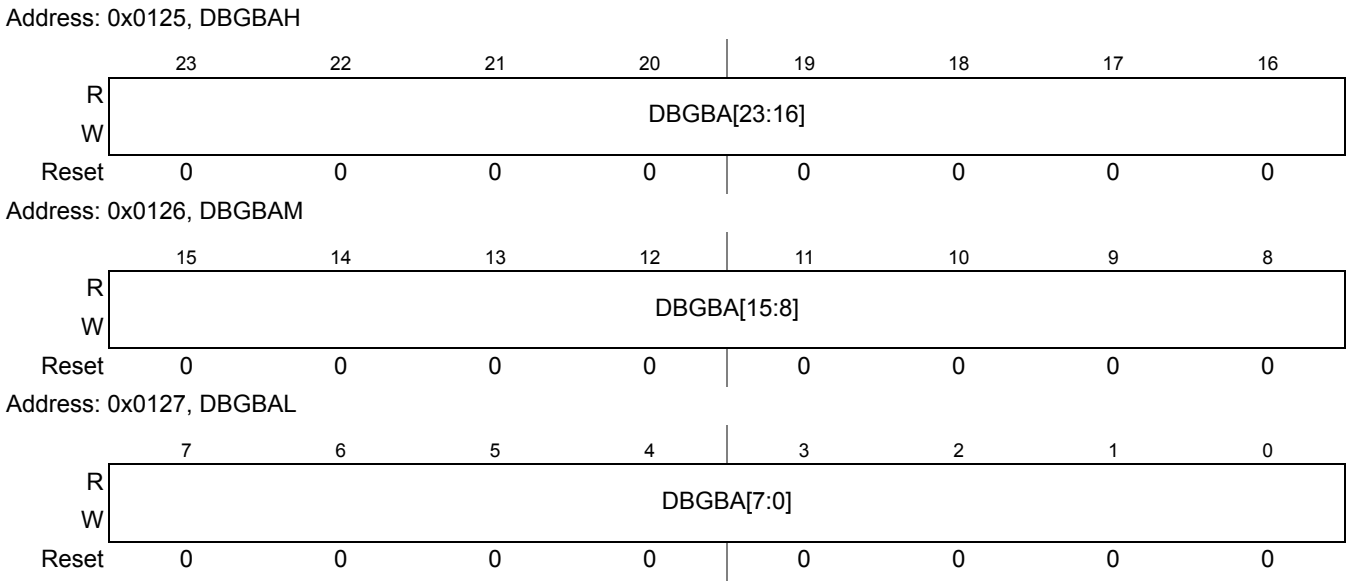


Figure 6-19. Debug Comparator B Address Register

Read: Anytime.

Write: If DBG not armed and PTACT is clear.

Table 6-33. DBGBAH, DBGBAM, DBGBAL Field Descriptions

Field	Description
23–16 DBGBA [23:16]	**Comparator Address Bits [23:16]**— These comparator address bits control whether the comparator compares the address bus bits [23:16] to a logic one or logic zero. 0 Compare corresponding address bit to a logic zero 1 Compare corresponding address bit to a logic one
15–0 DBGBA [15:0]	**Comparator Address Bits**[15:0]— These comparator address bits control whether the comparator compares the address bus bits [15:0] to a logic one or logic zero. 0 Compare corresponding address bit to a logic zero 1 Compare corresponding address bit to a logic one

- Make sure the PLL configuration is valid for the selected oscillator frequency.
- Enable the external oscillator (OSCE bit).
- Wait for oscillator to start up (UPOSC=1).
- Select the Oscillator Clock (OSCCLK) as source of the Bus Clock (PLLSEL=0).

— The PLLCLK is on and used to qualify the external oscillator clock.

8.1.3.2 Wait Mode

For S12CPMU_UHV_V10_V6 Wait Mode is the same as Run Mode.

8.1.3.3 Stop Mode

This mode is entered by executing the CPU STOP instruction.

The voltage regulator is in Reduced Performance Mode (RPM).

NOTE

The voltage regulator output voltage may degrade to a lower value than in Full Performance Mode (FPM), additionally the current sourcing capability is substantially reduced (see also Appendix for VREG electrical parameters). Only clock source ACLK is available and the Power On Reset (POR) circuitry is functional. The Low Voltage Interrupt (LVI) and Low Voltage Reset (LVR) are disabled.

The API is available.

The Phase Locked Loop (PLL) is off.

The Internal Reference Clock (IRC1M) is off.

Core Clock and Bus Clock are stopped.

Depending on the setting of the PSTP and the OSCE bit, Stop Mode can be differentiated between Full Stop Mode (PSTP = 0 or OSCE=0) and Pseudo Stop Mode (PSTP = 1 and OSCE=1). In addition, the behavior of the COP in each mode will change based on the clocking method selected by COPOSCSEL[1:0].

- **Full Stop Mode (PSTP = 0 or OSCE=0)**
  External oscillator (XOSCLCP) is disabled.
  — If COPOSCSEL1=0:
  The COP and RTI counters halt during Full Stop Mode.
  After wake-up from Full Stop Mode the Core Clock and Bus Clock are running on PLLCLK (PLLSEL=1). COP and RTI are running on IRCCLK (COPOSCSEL0=0, RTIOSCSEL=0).
  — If COPOSCSEL1=1:
  The clock for the COP is derived from ACLK (trimmable internal RC-Oscillator clock). During Full Stop Mode the ACLK for the COP can be stopped (COP static) or running (COP active) depending on the setting of bit CSAD. When bit CSAD is set the ACLK clock source for the

Table 9-10. ADCFLWCTL Field Descriptions

Field	Description
7 SEQA	**Conversion Sequence Abort Event** — This bit indicates that a conversion sequence abort event is in progress. When this bit is set the ongoing conversion sequence and current CSL will be aborted at the next conversion boundary. This bit gets cleared when the ongoing conversion sequence is aborted and ADC is idle. This bit can only be set if bit ADC_EN is set. This bit is cleared if bit ADC_EN is clear. Data Bus Control: This bit can be controlled via the data bus if access control is configured accordingly via ACC_CFG[1:0]. Writing a value of 1'b0 does not clear the flag. Writing a one to this bit does not clear it but causes an overrun if the bit has already been set. See Section 9.6.3.2.6, "Conversion flow control in case of conversion sequence control bit overrun scenarios for more details. Internal Interface Control: This bit can be controlled via the internal interface Signal "Seq_Abort" if access control is configured accordingly via ACC_CFG[1:0]. After being set an additional request via the internal interface Signal "Seq_Abort" causes an overrun. See also conversion flow control in case of overrun situations. General: In both conversion flow control modes (Restart Mode and Trigger Mode) when bit RSTA gets set automatically bit SEQA gets set when the ADC has not reached one of the following scenarios: - A Sequence Abort request is about to be executed or has been executed. - "End Of List" command type has been executed or is about to be executed In case bit SEQA is set automatically the Restart error flag RSTA_EIF is set to indicate an unexpected Restart Request. 0 No conversion sequence abort request. 1 Conversion sequence abort request.
6 TRIG	**Conversion Sequence Trigger Bit** — This bit starts a conversion sequence if set and no conversion or conversion sequence is ongoing. This bit is cleared when the first conversion of a sequence starts to sample. This bit can only be set if bit ADC_EN is set. This bit is cleared if bit ADC_EN is clear. Data Bus Control: This bit can be controlled via the data bus if access control is configured accordingly via ACC_CFG[1:0]. Writing a value of 1'b0 does not clear the flag. After being set this bit can not be cleared by writing a value of 1'b1 instead the error flag TRIG_EIF is set. See also Section 9.6.3.2.6, "Conversion flow control in case of conversion sequence control bit overrun scenarios for more details. Internal Interface Control: This bit can be controlled via the internal interface Signal "Trigger" if access control is configured accordingly via ACC_CFG[1:0]. After being set an additional request via internal interface Signal "Trigger" causes the flag TRIG_EIF to be set. 0 No conversion sequence trigger. 1 Trigger to start conversion sequence.

Table 9-11. Summary of Conversion Flow Control Bit Scenarios

RSTA	TRIG	SEQA	LDOK	Conversion Flow Control Mode	Conversion Flow Control Scenario
0	0	0	0	Both Modes	Valid
0	0	0	1	Both Modes	Can Not Occur
0	0	1	0	Both Modes	Valid[5.]
0	0	1	1	Both Modes	Can Not Occur
0	1	0	0	Both Modes	Valid[2.]
0	1	0	1	Both Modes	Can Not Occur
0	1	1	0	Both Modes	Can Not Occur
0	1	1	1	Both Modes	Can Not Occur
1	0	0	0	Both Modes	Valid[4.]
1	0	0	1	Both Modes	Valid[1. 4.]
1	0	1	0	Both Modes	Valid[3. 4. 5.]
1	0	1	1	Both Modes	Valid[1. 3. 4. 5.]
1	1	0	0	“Restart Mode”	Error flag TRIG_EIF set
				“Trigger Mode”	Valid[2. 4. 6.]
1	1	0	1	“Restart Mode”	Error flag TRIG_EIF set
				“Trigger Mode”	Valid[1. 2. 4. 6.]
1	1	1	0	“Restart Mode”	Error flag TRIG_EIF set
				“Trigger Mode”	Valid[2. 3. 4. 5. 6.]
1	1	1	1	“Restart Mode”	Error flag TRIG_EIF set
				“Trigger Mode”	Valid[(1) (2) (3) (4) (5) (6)]

1. Swap CSL buffer

2. Start conversion sequence

3. Prevent RSTA_EIF and LDOK_EIF

4. Load conversion command from top of CSL

5. Abort any ongoing conversion, conversion sequence and CSL

6. Bit TRIG set automatically in Trigger Mode

For a detailed description of all conversion flow control bit scenarios please see also Section 9.6.3.2.4, “The two conversion flow control Mode Configurations, Section 9.6.3.2.5, “The four ADC conversion flow control bits and Section 9.6.3.2.6, “Conversion flow control in case of conversion sequence control bit overrun scenarios

Table 9-24. Analog Input Channel Select

CH_SEL[5]	CH_SEL[4]	CH_SEL[3]	CH_SEL[2]	CH_SEL[1]	CH_SEL[0]	Analog Input Channel
0	0	0	0	0	0	VRL_0/1 (V1, V2, see Table 9-2) VRL_0 (V3, see Table 9-2)
0	0	0	0	0	1	VRH_0/1 (V1, V2, see Table 9-2) VRH_0/1/2 (V3, see Table 9-2)
0	0	0	0	1	0	(VRH_0/1 + VRL_0/1) / 2 (V1, V2, see Table 9-2) (VRH_0/1/2 + VRL_0) / 2 (V3, see Table 9-2)
0	0	0	0	1	1	Reserved
0	0	0	1	0	0	Reserved
0	0	0	1	0	1	Reserved
0	0	0	1	1	0	Reserved
0	0	0	1	1	1	Reserved
0	0	1	0	0	0	Internal_0 (ADC temperature sense)
0	0	1	0	0	1	Internal_1
0	0	1	0	1	0	Internal_2
0	0	1	0	1	1	Internal_3
0	0	1	1	0	0	Internal_4
0	0	1	1	0	1	Internal_5
0	0	1	1	1	0	Internal_6
0	0	1	1	1	1	Internal_7
0	1	0	0	0	0	AN0
0	1	0	0	0	1	AN1
0	1	0	0	1	0	AN2
0	1	0	0	1	1	AN3
0	1	0	1	0	0	AN4
0	1	x	x	x	x	ANx
1	x	x	x	x	x	Reserved

NOTE

ANx in Table 9-24 is the maximum number of implemented analog input channels on the device. Please refer to the device overview of the reference manual for details regarding number of analog input channels.

Chapter 12
Timer Module (TIM16B2CV3) Block Description

Table 12-1. Revision History

V03.03	Jan,14,2013		-single source generate different channel guide

12.1 Introduction

The basic scalable timer consists of a 16-bit, software-programmable counter driven by a flexible programmable prescaler.

This timer can be used for many purposes, including input waveform measurements while simultaneously generating an output waveform.

This timer could contain up to 2 input capture/output compare channels . The input capture function is used to detect a selected transition edge and record the time. The output compare function is used for generating output signals or for timer software delays.

A full access for the counter registers or the input capture/output compare registers should take place in one clock cycle. Accessing high byte and low byte separately for all of these registers may not yield the same result as accessing them in one word.

12.1.1 Features

The TIM16B2CV3 includes these distinctive features:

- Up to 2 channels available. (refer to device specification for exact number)
- All channels have same input capture/output compare functionality.
- Clock prescaling.
- 16-bit counter.

12.1.2 Modes of Operation

Stop: Timer is off because clocks are stopped.

Freeze: Timer counter keeps on running, unless TSFRZ in TSCR1 is set to 1.

Wait: Counters keeps on running, unless TSWAI in TSCR1 is set to 1.

Normal: Timer counter keep on running, unless TEN in TSCR1 is cleared to 0.

1. Read: Anytime
   Write: This register cannot be modified after the WP bit is set.

Address: Module Base + 0x0009 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	QSMP3		QSMP2		QSMP1		QSMP0	
Reset	0	0	0	0	0	0	0	0

Figure 15-12. PMF Fault Qualifying Samples Register (PMFQSMP1)

1. Read: Anytime
   Write: This register cannot be modified after the WP bit is set.

Table 15-14. PMFQSMP0-1 Field Descriptions

Field	Description
7–0 QSMP*m*[1:0]	**Fault *m* Qualifying Samples** — This field indicates the number of consecutive samples taken at the FAULT*m* input to determine if a fault is detected. The first sample is qualified after two bus cycles from the time the fault is present and each sample after that is taken every four core clock cycles. See Table 15-15. This register cannot be modified after the WP bit is set. *m* is 0, 1, 2, 3, 4 and 5.

Table 15-15. Qualifying Samples

QSMP*m*[1:0]	Number of Samples
00	1 sample(1)
01	5 samples
10	10 samples
11	15 samples

1. There is an asynchronous path from fault inputs FAULT3-0, FAULT4 if DMP*n*4=b10, and FAULT5 if DMP*n*5=b10 to disable PWMs immediately but the fault is qualified in two bus cycles.

15.3.2.10 PMF Output Control Register (PMFOUTC)

Address: Module Base + 0x000C Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	OUTCTL5	OUTCTL4	OUTCTL3	OUTCTL2	OUTCTL1	OUTCTL0
W								
Reset	0	0	0	0	0	0	0	0

Figure 15-13. PMF Output Control Register (PMFOUTC)

1. Read: Anytime
   Write: Anytime

Table 15-34. PMFFQCC Field Descriptions

Field	Description
7–4 LDFQC[3:0]	**Load Frequency C** — This field selects the PWM load frequency according to Table 15-35. See Section 15.4.12.3, “Load Frequency” for more details. **Note:** The LDFQC field takes effect when the current load cycle is complete, regardless of the state of the LDOKC bit or global load OK. Reading the LDFQC field reads the buffered value and not necessarily the value currently in effect.
3 HALFC	**Half Cycle Reload C** — This bit enables half-cycle reloads in center-aligned PWM mode. This bit has no effect on edge-aligned PWMs. It takes effect immediately. When set, reload opportunities occur also when the counter matches the modulus in addition to the start of the PWM period at count zero. See Section 15.4.12.3, “Load Frequency” for more details. 0 Half-cycle reloads disabled 1 Half-cycle reloads enabled
2–1 PRSCC[1:0]	**Prescaler C** — This buffered field selects the PWM clock frequency illustrated in Table 15-36. **Note:** Reading the PRSCC field reads the buffered value and not necessarily the value currently in effect. The PRSCC field takes effect at the beginning of the next PWM cycle and only when the LDOKC bit or global load OK is set.
0 PWMRFC	**PWM Reload Flag C** — This flag is set at the beginning of every reload cycle regardless of the state of the LDOKC bit or global load OK. Clear PWMRFC by reading PMFFQCC with PWMRFC set and then writing a logic one to the PWMRFC bit. If another reload occurs before the clearing sequence is complete, writing logic one to PWMRFC has no effect. 0 No new reload cycle since last PWMRFC clearing 1 New reload cycle since last PWMRFC clearing **Note:** Clearing PWMRFC satisfies pending PWMRFC CPU interrupt requests.

Table 15-35. PWM Reload Frequency C

LDFQC[3:0]	PWM Reload Frequency	LDFQ[3:0]	PWM Reload Frequency
0000	Every PWM opportunity	1000	Every 9 PWM opportunities
0001	Every 2 PWM opportunities	1001	Every 10 PWM opportunities
0010	Every 3 PWM opportunities	1010	Every 11 PWM opportunities
0011	Every 4 PWM opportunities	1011	Every 12 PWM opportunities
0100	Every 5 PWM opportunities	1100	Every 13 PWM opportunities
0101	Every 6 PWM opportunities	1101	Every 14 PWM opportunities
0110	Every 7 PWM opportunities	1110	Every 15 PWM opportunities
0111	Every 8 PWM opportunities	1111	Every 16 PWM opportunities

Table 15-36. PWM Prescaler C

PRSCC[1:0]	Prescaler Value P_C	PWM Clock Frequency f_{PWM_C}
00	1	f_{core}
01	2	$f_{core}/2$
10	4	$f_{core}/4$
11	8	$f_{core}/8$

Table 15-40. PWM Value and Underflow Conditions

PMFVALn	Condition	PWM Value Used
0x0000–0x7FFF	Normal	Value in registers
0x8000–0xFFFF	Underflow	0x0000

Center-aligned operation is illustrated in Figure 15-46.

$$\text{PWM pulse width} = (\text{PWM value}) \times (\text{PWM clock period}) \times 2 \qquad \textit{Eqn. 15-6}$$

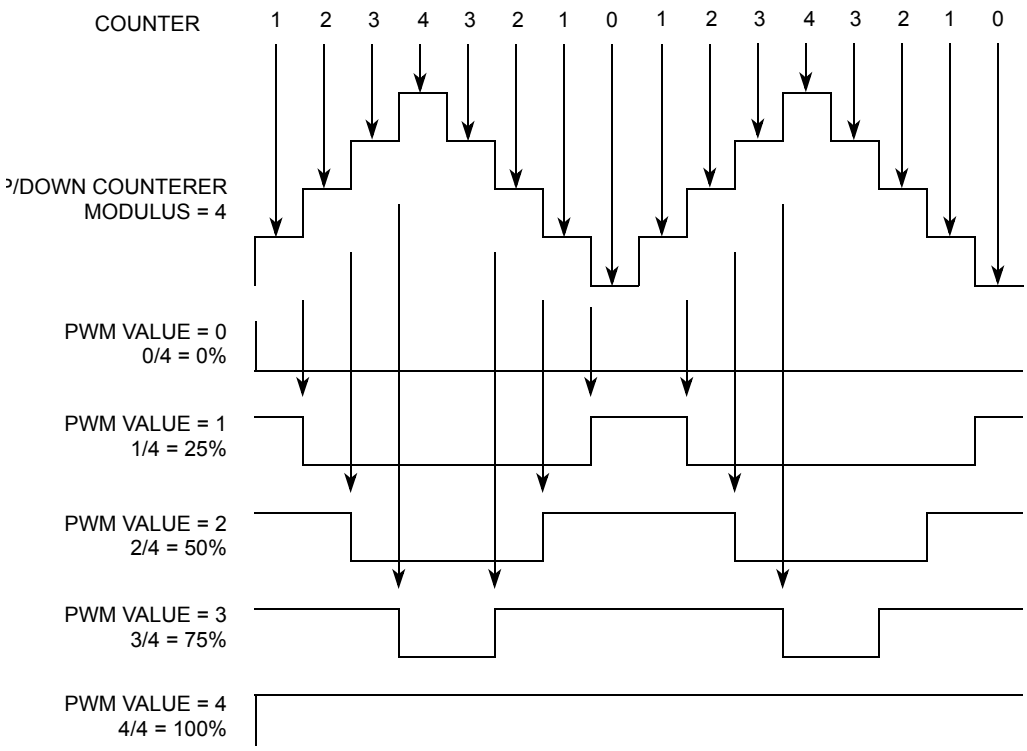


Figure 15-46. Center-Aligned PWM Pulse Width

Edge-aligned operation is illustrated in Figure 15-47.

$$\text{PWM pulse width} = (\text{PWM value}) \times (\text{PWM clock period}) \qquad \textit{Eqn. 15-7}$$

in center-aligned operation and at the end of cycle in edge-aligned operation. Using this mode requires external circuitry to sense current direction.

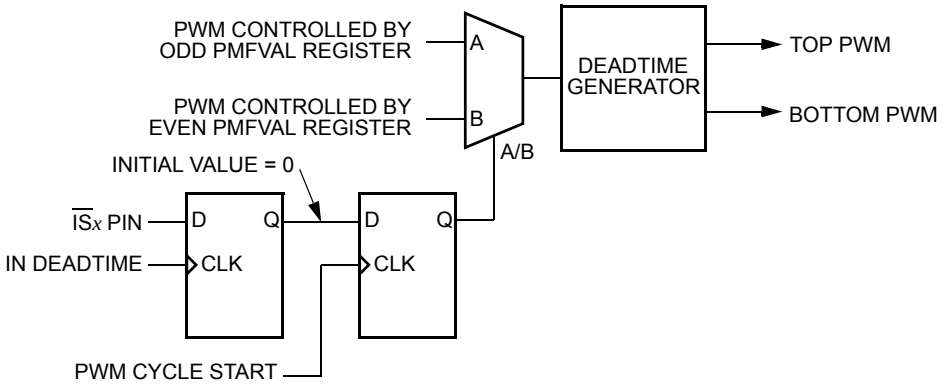


Figure 15-58. Internal Correction Logic when ISENS = 10

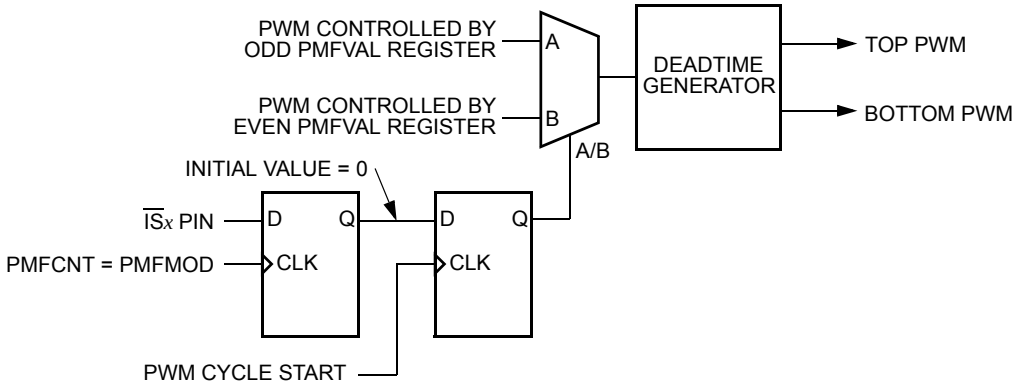


Figure 15-59. Internal Correction Logic when ISENS = 11

NOTE

Values latched on the $\overline{IS}x$ inputs are buffered so only one PWM register is used per PWM cycle. If a current status changes during a PWM period, the new value does not take effect until the next PWM period.

When initially enabled by setting the PWMEN bit, no current status has previously been sampled. PWM value registers one, three, and five initially control the three PWM pairs when configured for current status correction.

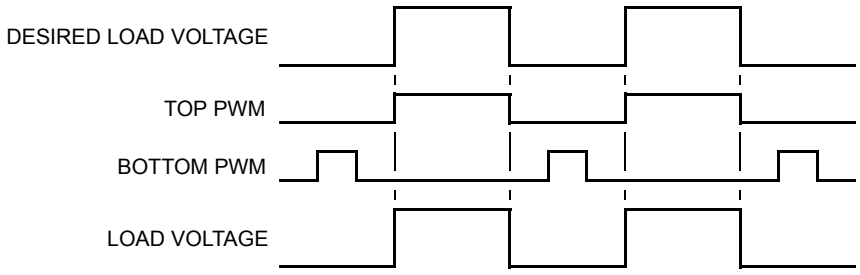


Figure 15-60. Correction with Positive Current

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

PPOLx = 0

PWMPERx = 4

PWMDTYx = 1

PWMx Frequency = 10 MHz/8 = 1.25 MHz

PWMx Period = 800 ns

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

Shown in Figure 22-20 is the output waveform generated.

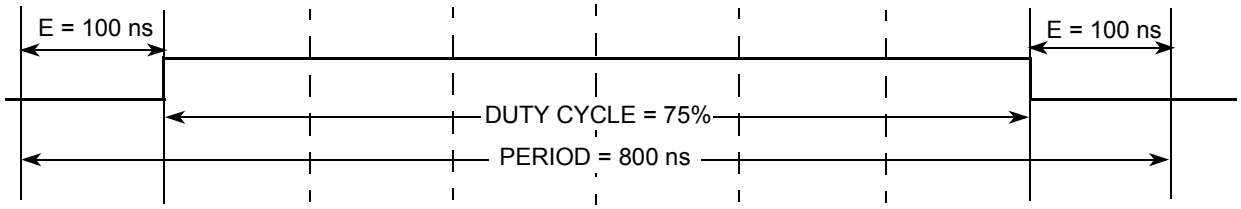


Figure 22-20. PWM Center Aligned Output Example Waveform

22.4.2.7 PWM 16-Bit Functions

The scalable PWM timer also has the option of generating up to 8-channels of 8-bits or 4-channels of 16-bits for greater PWM resolution. This 16-bit channel option is achieved through the concatenation of two 8-bit channels.

The PWMCTL register contains four control bits, each of which is used to concatenate a pair of PWM channels into one 16-bit channel. Channels 6 and 7 are concatenated with the CON67 bit, channels 4 and 5 are concatenated with the CON45 bit, channels 2 and 3 are concatenated with the CON23 bit, and channels 0 and 1 are concatenated with the CON01 bit.

NOTE

Change these bits only when both corresponding channels are disabled.

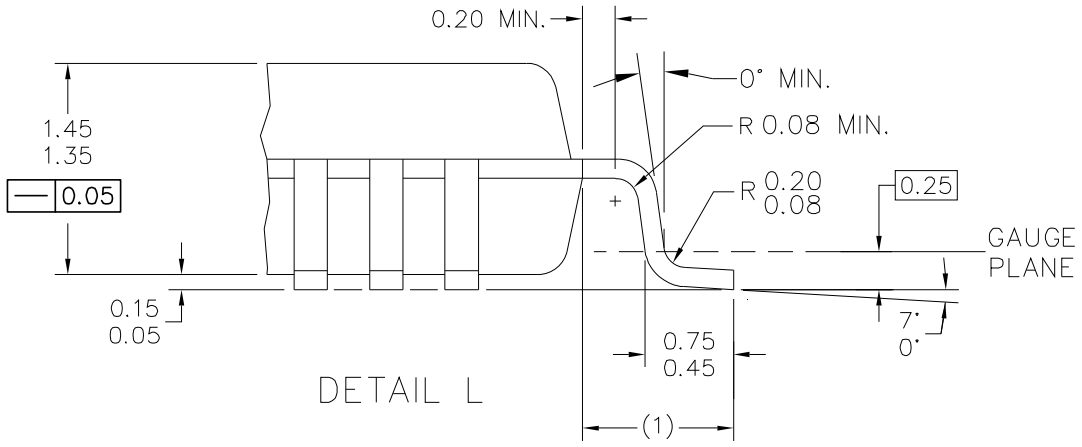
When channels 6 and 7 are concatenated, channel 6 registers become the high order bytes of the double byte channel, as shown in Figure 22-21. Similarly, when channels 4 and 5 are concatenated, channel 4 registers become the high order bytes of the double byte channel. When channels 2 and 3 are concatenated, 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 22-21. The polarity of the resulting PWM output is controlled by the PPOLx bit of the corresponding low order 8-bit channel as well.

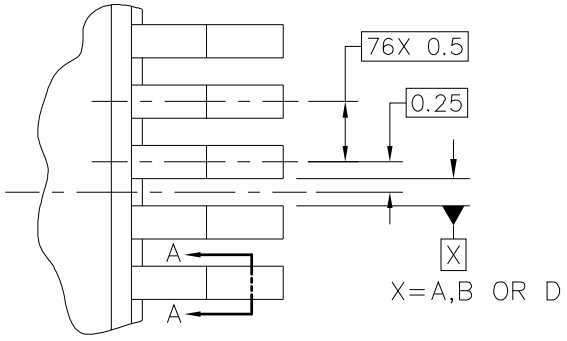
freescale

MECHANICAL OUTLINE

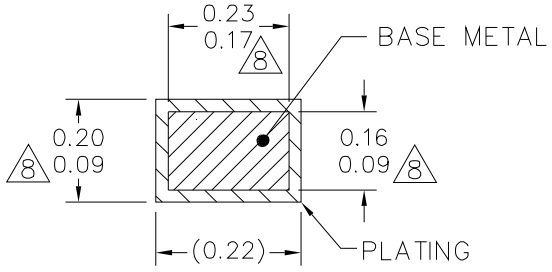
DO NOT SCALE THIS DRAWING



DETAIL L



DETAIL K



SECTION A–A
100 PLACES
ROTATED 90° CLOCKWISE

TITLE: LQFP, 12 X 12 X 1.4 PKG, 0.5 PITCH, 80LD, 5.6 X 5.6 EXPOSED PAD	DOCUMENT NO: 98ASA00505D	REV: X2
	STANDARD: NON–JEDEC	
	SHEET:	2

M.10 0x0500-x053F PMF15B6C

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0529	PMFFQCB	R	LDFQB				HALFB	PRSCB		PWMRFB
		W								
0x052A	PMFCNTB	R	0	PMFCNTB						
		W								
0x052B	PMFCNTB	R	PMFCNTB							
		W								
0x052C	PMFMODB	R	0	PMFMODB						
		W								
0x052D	PMFMODB	R	PMFMODB							
		W								
0x052E	PMFDTMB	R	0	0	0	0	PMFDTMB			
		W								
0x052F	PMFDTMB	R	PMFDTMB							
		W								
0x0530	PMFENCC	R	PWMENC	GLDOKC	0	0	0	RSTRTC	LDOKC	PWMRIEC
		W								
0x0531	PMFFQCC	R	LDFQC				HALFC	PRSCC		PWMRFC
		W								
0x0532	PMFCNTC	R	0	PMFCNTC						
		W								
0x0533	PMFCNTC	R	PMFCNTC							
		W								
0x0534	PMFMODC	R	0	PMFMODC						
		W								
0x0535	PMFMODC	R	PMFMODC							
		W								
0x0536	PMFDTMC	R	0	0	0	0	PMFDTMC			
		W								
0x0537	PMFDTMC	R	PMFDTMC							
		W								
0x0538	PMFDMP0	R	DMP05		DMP04		DMP03	DMP02	DMP01	DMP00
		W								
0x0539	PMFDMP1	R	DMP15		DMP14		DMP13	DMP12	DMP11	DMP10
		W								

M.12 0x05C0-0x05FF TIM0

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x05D2	TIM0TC1H	R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
0x05D3	TIM0TC1L	R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x05D4	TIM0TC2H	R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
0x05D5	TIM0TC2L	R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x05D6	TIM0TC3H	R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
0x05D7	TIM0TC3L	R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x05D8– 0x05DF	Reserved	R W								
0x05E0	Reserved	R W								
0x05E1	Reserved	R W								
0x05E2	Reserved	R W								
0x05E3	Reserved	R W								
0x05E4– 0x05EB	Reserved	R W								
0x05EC	TIM0OCPD	R W					OCPD3	OCPD2	OCPD1	OCPD0
0x05ED	Reserved	R W								
0x05EE	TIM0PTPSR	R W	PTPS7	PTPS6	PTPS5	PTPS4	PTPS3	PTPS2	PTPS1	PTPS0
0x05EF	Reserved	R W								

M.17 0x06F0-0x06F7 BATS

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x06F0	BATE	R	0	BVHS	BVLS[1:0]		BSUAE	BSUSE	0	0
		W								
0x06F1	BATSR	R	0	0	0	0	0	0	BVHC	BVLC
		W								
0x06F2	BATIE	R	0	0	0	0	0	0	BVHIE	BVLIE
		W								
0x06F3	BATIF	R	0	0	0	0	0	0	BVHIF	BVLIF
		W								
0x06F4 - 0x06F5	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x06F6 - 0x06F7	Reserved	R	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
		W								

M.18 0x0700-0x0707 SCI0

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0700	SCI0BDH[1]	R	SBR15	SBR14	SBR13	SBR12	SBR11	SBR10	SBR9	SBR8
		W								
0x0701	SCI0BDL[1]	R	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
		W								
0x0702	SCI0CR1[1]	R	LOOPS	SCISWAI	RSRC	M	WAKE	ILT	PE	PT
		W								
0x0700	SCI0ASR1[2]	R	RXEDGIF	0	0	0	0	BERRV	BERRIF	BKDIF
		W								
0x0701	SCI0ACR1[2]	R	RXEDGIE	0	0	0	0	0	BERRIE	BKDIE
		W								
0x0702	SCI0ACR2[2]	R	IREN	TNP1	TNP0	0	0	BERRM1	BERRM0	BKDFE
		W								
0x0703	SCI0CR2	R	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
		W								
0x0704	SCI0SR1	R	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
		W								
0x0705	SCI0SR2	R	AMAP	0	0	TXPOL	RXPOL	BRK13	TXDIR	RAF
		W								