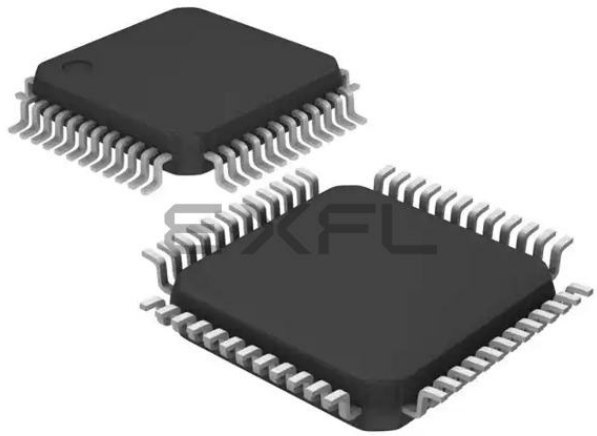


Welcome to E-XFL.COM

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	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	CANbus, EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	31
Program Memory Size	96KB (96K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s12c96vfae

1.2.2 Detailed Register Map

The detailed register map of the MC9S12C128 is listed in address order below.

0x0000–0x000F MEBI Map 1 of 3 (HCS12 Multiplexed External Bus Interface)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0000	PORTA	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
0x0001	PORTB	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
0x0002	DDRA	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
0x0003	DDRB	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
0x0004	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								
0x0005	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								
0x0006	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								
0x0007	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								
0x0008	PORTE	Read:	Bit 7	6	5	4	3	2	Bit 1	Bit 0
		Write:								
0x0009	DDRE	Read:	Bit 7	6	5	4	3	Bit 2	0	0
		Write:								
0x000A	PEAR	Read:	NOACCE	0	PIPOE	NECLK	LSTRE	RDWE	0	0
		Write:								
0x000B	MODE	Read:	MODC	MODB	MODA	0	IVIS	0	EMK	EME
		Write:								
0x000C	PUCR	Read:	PUPKE	0	0	PUPEE	0	0	PUPBE	PUPAE
		Write:								
0x000D	RDRIV	Read:	RDPK	0	0	RDPE	0	0	RDPB	RDPA
		Write:								
0x000E	EBICTL	Read:	0	0	0	0	0	0	0	ESTR
		Write:								
0x000F	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0006	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0007	MODRR	R	0	0	0	MODRR4	MODRR3	MODRR2	MODRR1	MODRR0
		W								
0x0008	PTS	R	0	0	0	0	PTS3	PTS2	PTS1	PTS0
		W								
		SCI	—	—	—	—	—	—	TXD	RXD
0x0009	PTIS	R	0	0	0	0	PTIS3	PTIS2	PTIS1	PTIS0
		W								
0x000A	DDRS	R	0	0	0	0	DDRS3	DDRS2	DDRS1	DDRS0
		W								
0x000B	RDRS	R	0	0	0	0	RDRS3	RDRS2	RDRS1	RDRS0
		W								
0x000C	PERS	R	0	0	0	0	PERS3	PERS2	PERS1	PERS0
		W								
0x000D	PPSS	R	0	0	0	0	PPSS3	PPSS2	PPSS1	PPSS0
		W								
0x000E	WOMS	R	0	0	0	0	WOMS3	WOMS2	WOMS1	WOMS0
		W								
0x000F	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0010	PTM	R	0	0	PTM5	PTM4	PTM3	PTM2	PTM1	PTM0
		W								
		MSCAN / SPI	—	—	SCK	MOSI	$\overline{SS}$	MISO	TXCAN	RXCAN
0x0011	PTIM	R	0	0	PTIM5	PTIM4	PTIM3	PTIM2	PTIM1	PTIM0
		W								
0x0012	DDRM	R	0	0	DDRM5	DDRM4	DDRM3	DDRM2	DDRM1	DDRM0
		W								
0x0013	RDRM	R	0	0	RDRM5	RDRM4	RDRM3	RDRM2	RDRM1	RDRM0
		W								
0x0014	PERM	R	0	0	PERM5	PERM4	PERM3	PERM2	PERM1	PERM0
		W								
0x0015	PPSM	R	0	0	PPSM5	PPSM4	PPSM3	PPSM2	PPSM1	PPSM0
		W								
0x0016	WOMM	R	0	0	WOMM5	WOMM4	WOMM3	WOMM2	WOMM1	WOMM0
		W								
0x0017	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0018	PTP	R	PTP7	PTP6	PTP5	PTP4	PTP3	PTP2	PTP1	PTP0
		W								
		PWM	—	—	PWM5	PWM4	PWM3	PWM2	PWM1	PWM0
0x0019	PTIP	R	PTIP7	PTIP6	PTIP5	PTIP4	PTIP3	PTIP2	PTIP1	PTIP0
		W								

= Unimplemented or Reserved

Figure 2-2. Quick Reference to PIM Registers (Sheet 2 of 3)

6.3.2.1 BDM Status Register (BDMSTS)

0xFF01

	7	6	5	4	3	2	1	0
R	ENBDM	BDMACT	ENTAG	SDV	TRACE	CLKSW	UNSEC	0
W								
Reset:								
Special single-chip mode:	1(1)	1	0	0	0	0	0(2)	0
Special peripheral mode:	0	1	0	0	0	0	0	0
All other modes:	0	1	0	0	0	0	0	0
	0	0	0	0	0	0	0	0

= Unimplemented or Reserved = Implemented (do not alter)

Figure 6-3. BDM Status Register (BDMSTS)

Note:
1. ENBDM is read as "1" by a debugging environment in Special single-chip mode when the device is not secured or secured but fully erased (Flash and EEPROM).This is because the ENBDM bit is set by the standard firmware before a BDM command can be fully transmitted and executed.

2. UNSEC is read as "1" by a debugging environment in Special single-chip mode when the device is secured and fully erased, else it is "0" and can only be read if not secure (see also bit description).

Read: All modes through BDM operation

Write: All modes but subject to the following:

- BDMACT can only be set by BDM hardware upon entry into BDM. It can only be cleared by the standard BDM firmware lookup table upon exit from BDM active mode.
- CLKSW can only be written via BDM hardware or standard BDM firmware write commands.
- All other bits, while writable via BDM hardware or standard BDM firmware write commands, should only be altered by the BDM hardware or standard firmware lookup table as part of BDM command execution.
- ENBDM should only be set via a BDM hardware command if the BDM firmware commands are needed. (This does not apply in special single-chip mode).

Table 9-2. CRGFLG Field Descriptions (continued)

Field	Description
1 SCMIF	**Self-Clock Mode Interrupt Flag** — SCMIF is set to 1 when SCM status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (SCMIE=1), SCMIF causes an interrupt request. 0 No change in SCM bit. 1 SCM bit has changed.
0 SCM	**Self-Clock Mode Status Bit** — SCM reflects the current clocking mode. Writes have no effect. 0 MCU is operating normally with OSCCLK available. 1 MCU is operating in self-clock mode with OSCCLK in an unknown state. All clocks are derived from PLLCLK running at its minimum frequency f_{SCM}.

9.3.2.5 CRG Interrupt Enable Register (CRGINT)

This register enables CRG interrupt requests.

Module Base + 0x0004

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

= Unimplemented or Reserved

Figure 9-8. CRG Interrupt Enable Register (CRGINT)

Read: anytime

Write: anytime

Table 9-3. CRGINT 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	**Lock Interrupt Enable Bit** 0 LOCK interrupt requests are disabled. 1 Interrupt will be requested whenever LOCKIF is set.
1 SCMIE	**Self-Clock Mode Interrupt Enable Bit** 0 SCM interrupt requests are disabled. 1 Interrupt will be requested whenever SCMIF is set.

Module Base + 0x0001

	7	6	5	4	3	2	1	0
R	CANE	CLKSRC	LOOPB	LISTEN		WUPM	SLPAK	INITAK
W								
Reset:	0	0	0	1	0	0	0	1

= Unimplemented

Figure 10-5. MSCAN Control Register 1 (CANCTL1)

Read: Anytime
Write: Anytime when INITRQ = 1 and INITAK = 1, except CANE which is write once in normal and anytime in special system operation modes when the MSCAN is in initialization mode (INITRQ = 1 and INITAK = 1).

Table 10-2. CANCTL1 Register Field Descriptions

Field	Description
7 CANE	**MSCAN Enable** 0 MSCAN module is disabled 1 MSCAN module is enabled
6 CLKSRC	**MSCAN Clock Source** — This bit defines the clock source for the MSCAN module (only for systems with a clock generation module; Section 10.4.3.2, "Clock System," and Section Figure 10-42., "MSCAN Clocking Scheme,"). 0 MSCAN clock source is the oscillator clock 1 MSCAN clock source is the bus clock
5 LOOPB	**Loopback Self Test Mode** — When this bit is set, the MSCAN performs an internal loopback which can be used for self test operation. The bit stream output of the transmitter is fed back to the receiver internally. The RXCAN input pin is ignored and the TXCAN output goes to the recessive state (logic 1). The MSCAN behaves as it does normally when transmitting and treats its own transmitted message as a message received from a remote node. In this state, the MSCAN ignores the bit sent during the ACK slot in the CAN frame acknowledge field to ensure proper reception of its own message. Both transmit and receive interrupts are generated. 0 Loopback self test disabled 1 Loopback self test enabled
4 LISTEN	**Listen Only Mode** — This bit configures the MSCAN as a CAN bus monitor. When LISTEN is set, all valid CAN messages with matching ID are received, but no acknowledgement or error frames are sent out (see Section 10.4.4.4, "Listen-Only Mode"). In addition, the error counters are frozen. Listen only mode supports applications which require "hot plugging" or throughput analysis. The MSCAN is unable to transmit any messages when listen only mode is active. 0 Normal operation 1 Listen only mode activated
2 WUPM	**Wake-Up Mode** — If WUPE in CANCTL0 is enabled, this bit defines whether the integrated low-pass filter is applied to protect the MSCAN from spurious wake-up (see Section 10.4.5.4, "MSCAN Sleep Mode"). 0 MSCAN wakes up on any dominant level on the CAN bus 1 MSCAN wakes up only in case of a dominant pulse on the CAN bus that has a length of T_{wup}

Table 10-2. CANCTL1 Register Field Descriptions (continued)

Field	Description
1 SLPAK	**Sleep Mode Acknowledge** — This flag indicates whether the MSCAN module has entered sleep mode (see Section 10.4.5.4, "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. 0 Running — The MSCAN operates normally 1 Sleep mode active — The MSCAN has entered sleep mode
0 INITAK	**Initialization Mode Acknowledge** — This flag indicates whether the MSCAN module is in initialization mode (see Section 10.4.5.5, "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. 0 Running — The MSCAN operates normally 1 Initialization mode active — The MSCAN has entered initialization mode

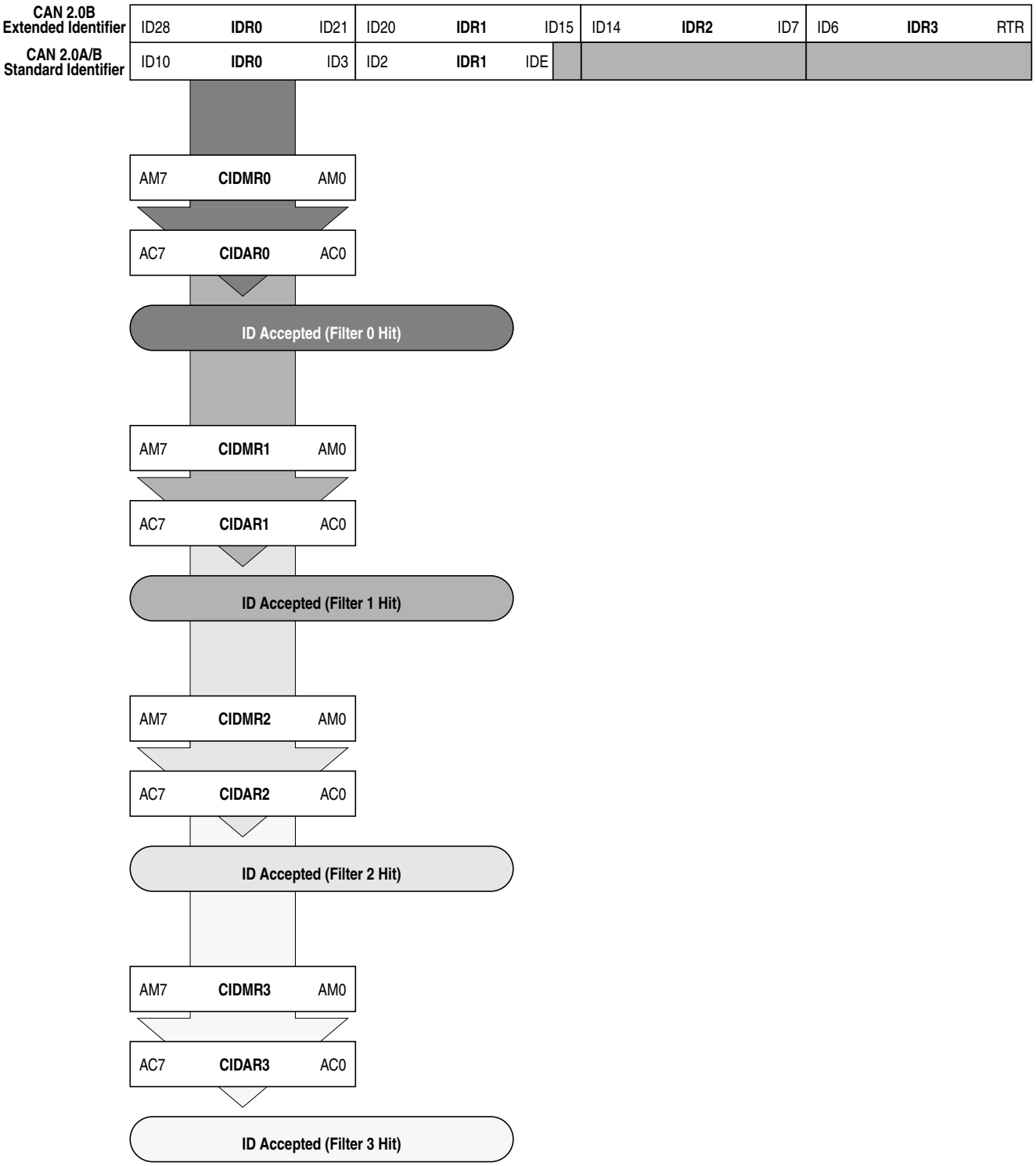


Figure 10-41. 8-bit Maskable Identifier Acceptance Filters

If the bus clock is generated from a PLL, it is recommended to select the oscillator clock rather than the bus clock due to jitter considerations, especially at the faster CAN bus rates.

For microcontrollers without a clock and reset generator (CRG), CANCLK is driven from the crystal oscillator (oscillator clock).

A programmable prescaler generates the time quanta (Tq) clock from CANCLK. A time quantum is the atomic unit of time handled by the MSCAN.

Eqn. 10-2

$$Tq = \frac{f_{CANCLK}}{(\text{Prescaler value})}$$

A bit time is subdivided into three segments as described in the Bosch CAN specification. (see Figure 10-43):

- SYNC_SEG: This segment has a fixed length of one time quantum. Signal edges are expected to happen within this section.
- Time Segment 1: This segment includes the PROP_SEG and the PHASE_SEG1 of the CAN standard. It can be programmed by setting the parameter TSEG1 to consist of 4 to 16 time quanta.
- Time Segment 2: This segment represents the PHASE_SEG2 of the CAN standard. It can be programmed by setting the TSEG2 parameter to be 2 to 8 time quanta long.

Eqn. 10-3

$$\text{Bit Rate} = \frac{f_{Tq}}{(\text{number of Time Quanta})}$$

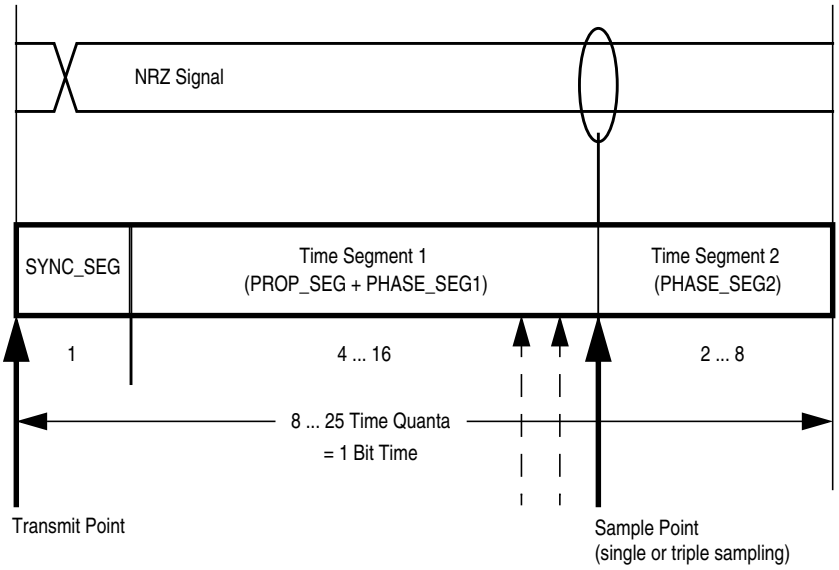


Figure 10-43. Segments within the Bit Time

Chapter 11
Oscillator (OSCV2) Block Description

11.1 Introduction

The OSCV2 module provides two alternative oscillator concepts:

- A low noise and low power Colpitts oscillator with amplitude limitation control (ALC)
- A robust full swing Pierce oscillator with the possibility to feed in an external square wave

11.1.1 Features

The Colpitts OSCV2 option provides the following features:

- Amplitude limitation control (ALC) loop:
  - Low power consumption and low current induced RF emission
  - Sinusoidal waveform with low RF emission
  - Low crystal stress (an external damping resistor is not required)
  - Normal and low amplitude mode for further reduction of power and emission
- An external biasing resistor is not required

The Pierce OSC option provides the following features:

- Wider high frequency operation range
- No DC voltage applied across the crystal
- Full rail-to-rail (2.5 V nominal) swing oscillation with low EM susceptibility
- Fast start up

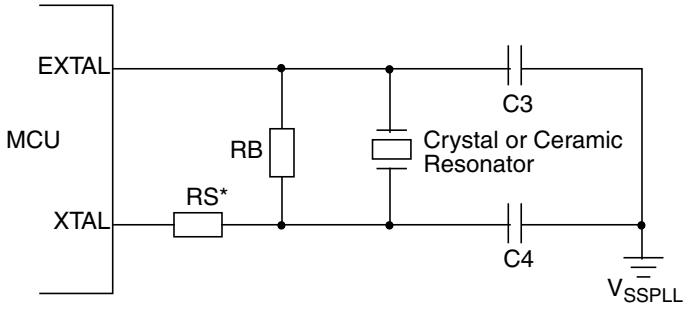
Common features:

- Clock monitor (CM)
- Operation from the V_{DDPLL} 2.5 V (nominal) supply rail

11.1.2 Modes of Operation

Two modes of operation exist:

- Amplitude limitation controlled Colpitts oscillator mode suitable for power and emission critical applications
- Full swing Pierce oscillator mode that can also be used to feed in an externally generated square wave suitable for high frequency operation and harsh environments



* Rs can be zero (shorted) when used with higher frequency crystals.
Refer to manufacturer's data.

Figure 11-2. Pierce Oscillator Connections (XCLKS = 1)

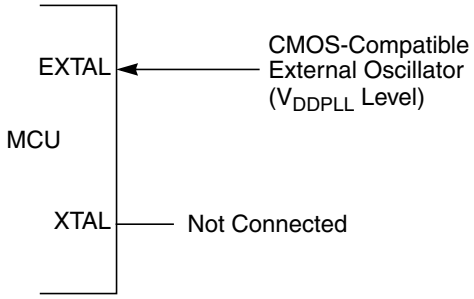


Figure 11-3. External Clock Connections (XCLKS = 1)

11.2.3 XCLKS — Colpitts/Pierce Oscillator Selection Signal

The XCLKS is an input signal which controls whether a crystal in combination with the internal Colpitts (low power) oscillator is used or whether the Pierce oscillator/external clock circuitry is used. The XCLKS signal is sampled during reset with the rising edge of $\overline{\text{RESET}}$. Table 11-1 lists the state coding of the sampled XCLKS signal. Refer to the device overview chapter for polarity of the XCLKS pin.

Table 11-1. Clock Selection Based on XCLKS

XCLKS	Description
0	Colpitts oscillator selected
1	Pierce oscillator/external clock selected

12.3.2 Register Descriptions

The following paragraphs describe in detail all the registers and register bits in the PWM8B6CV1 module.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 PWME	R	0	0	PWME5	PWME4	PWME3	PWME2	PWME1	PWME0
	W								
0x0001 PWMPOL	R	0	0	PPOL5	PPOL4	PPOL3	PPOL2	PPOL1	PPOL0
	W								
0x0002 PWMCLK	R	0	0	PCLK5	PCLK4	PCLK3	PCLK2	PCLK1	PCLK0
	W								
0x0003 PWMPRCLK	R	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
	W								
0x0004 PWMCAE	R	0	0	CAE5	CAE4	CAE2	CAE2	CAE1	CAE0
	W								
0x0005 PWMCTL	R	0	CON45	CON23	CON01	PSWAI	PFRZ	0	0
	W								
0x0006 PWMTST	R	0	0	0	0	0	0	0	0
	W								
0x0007 PWMPRSC	R	0	0	0	0	0	0	0	0
	W								
0x0008 PWMSCLA	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x0009 PWMSCLB	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x000A PWMSCNTA	R	0	0	0	0	0	0	0	0
	W								
0x000B PWMSCNTB	R	0	0	0	0	0	0	0	0
	W								
0x000C PWMCNT0	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x000D PWMCNT1	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x000E 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 12-2. PWM Register Summary

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000F PWMCNT3	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x0010 PWMCNT4	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x0011 PWMCNT5	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x0012 PWMPER0	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0013 PWMPER1	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0014 PWMPER2	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0015 PWMPER3	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0016 PWMPER4	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0017 PWMPER5	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0018 PWMDTY0	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x0019 PWMPER1	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001A PWMPER2	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001B PWMPER3	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001C PWMPER4	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001D PWMPER5	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x001E PWMSDB	R	PWMIF	PWMIE	0	PWMLVL	0	PWM5IN	PWM5INL	PWM5ENA
	W			PWMRSTRT					

= Unimplemented or Reserved

Figure 12-2. PWM Register Summary (continued)

Table 13-5. SCISR1 Field Descriptions (continued)

Field	Description
5 RDRF	**Receive Data Register Full Flag** — RDRF is set when the data in the receive shift register transfers to the SCI data register. Clear RDRF by reading SCI status register 1 (SCISR1) with RDRF set and then reading SCI data register low (SCIDRL). 0 Data not available in SCI data register 1 Received data available in SCI data register
4 IDLE	**Idle Line Flag** — IDLE is set when 10 consecutive logic 1s (if M=0) or 11 consecutive logic 1s (if M=1) appear on the receiver input. Once the IDLE flag is cleared, a valid frame must again set the RDRF flag before an idle condition can set the IDLE flag.Clear IDLE by reading SCI status register 1 (SCISR1) with IDLE set and then reading SCI data register low (SCIDRL). 0 Receiver input is either active now or has never become active since the IDLE flag was last cleared 1 Receiver input has become idle **Note:** When the receiver wakeup bit (RWU) is set, an idle line condition does not set the IDLE flag.
3 OR	**Overrun Flag** — OR is set when software fails to read the SCI data register before the receive shift register receives the next frame. The OR bit is set immediately after the stop bit has been completely received for the second frame. The data in the shift register is lost, but the data already in the SCI data registers is not affected. Clear OR by reading SCI status register 1 (SCISR1) with OR set and then reading SCI data register low (SCIDRL). 0 No overrun 1 Overrun **Note:** OR flag may read back as set when RDRF flag is clear. This may happen if the following sequence of events occurs: 1. After the first frame is received, read status register SCISR1 (returns RDRF set and OR flag clear); 2. Receive second frame without reading the first frame in the data register (the second frame is not received and OR flag is set); 3. Read data register SCIDRL (returns first frame and clears RDRF flag in the status register); 4. Read status register SCISR1 (returns RDRF clear and OR set). Event 3 may be at exactly the same time as event 2 or any time after. When this happens, a dummy SCIDRL read following event 4 will be required to clear the OR flag if further frames are to be received.
2 NF	**Noise Flag** — NF is set when the SCI detects noise on the receiver input. NF bit is set during the same cycle as the RDRF flag but does not get set in the case of an overrun. Clear NF by reading SCI status register 1(SCISR1), and then reading SCI data register low (SCIDRL). 0 No noise 1 Noise
1 FE	**Framing Error Flag** — FE is set when a logic 0 is accepted as the stop bit. FE bit is set during the same cycle as the RDRF flag but does not get set in the case of an overrun. FE inhibits further data reception until it is cleared. Clear FE by reading SCI status register 1 (SCISR1) with FE set and then reading the SCI data register low (SCIDRL). 0 No framing error 1 Framing error
0 PF	**Parity Error Flag** — PF is set when the parity enable bit (PE) is set and the parity of the received data does not match the parity type bit (PT). PF bit is set during the same cycle as the RDRF flag but does not get set in the case of an overrun. Clear PF by reading SCI status register 1 (SCISR1), and then reading SCI data register low (SCIDRL). 0 No parity error 1 Parity error

15.4.6 Gated Time Accumulation Mode

Setting the PAMOD bit configures the pulse accumulator for gated time accumulation operation. An active level on the PACNT input pin enables a divided-by-64 clock to drive the pulse accumulator. The PEDGE bit selects low levels or high levels to enable the divided-by-64 clock.

The trailing edge of the active level at the IOC7 pin sets the PAIF. The PAI bit enables the PAIF flag to generate interrupt requests.

The pulse accumulator counter register reflect the number of pulses from the divided-by-64 clock since the last reset.

NOTE

The timer prescaler generates the divided-by-64 clock. If the timer is not active, there is no divided-by-64 clock.

15.5 Resets

The reset state of each individual bit is listed within Section 15.3, “Memory Map and Register Definition” which details the registers and their bit fields.

15.6 Interrupts

This section describes interrupts originated by the TIM16B8CV1 block. Table 15-23 lists the interrupts generated by the TIM16B8CV1 to communicate with the MCU.

Table 15-23. TIM16B8CV1 Interrupts

Interrupt	Offset (1)	Vector[1]	Priority[1]	Source	Description
C[7:0]F	—	—	—	Timer Channel 7–0	Active high timer channel interrupts 7–0
PAOVI	—	—	—	Pulse Accumulator Input	Active high pulse accumulator input interrupt
PAOVF	—	—	—	Pulse Accumulator Overflow	Pulse accumulator overflow interrupt
TOF	—	—	—	Timer Overflow	Timer Overflow interrupt

1. Chip Dependent.

The TIM16B8CV1 uses a total of 11 interrupt vectors. The interrupt vector offsets and interrupt numbers are chip dependent.

15.6.1 Channel [7:0] Interrupt (C[7:0]F)

This active high outputs will be asserted by the module to request a timer channel 7 – 0 interrupt to be serviced by the system controller.

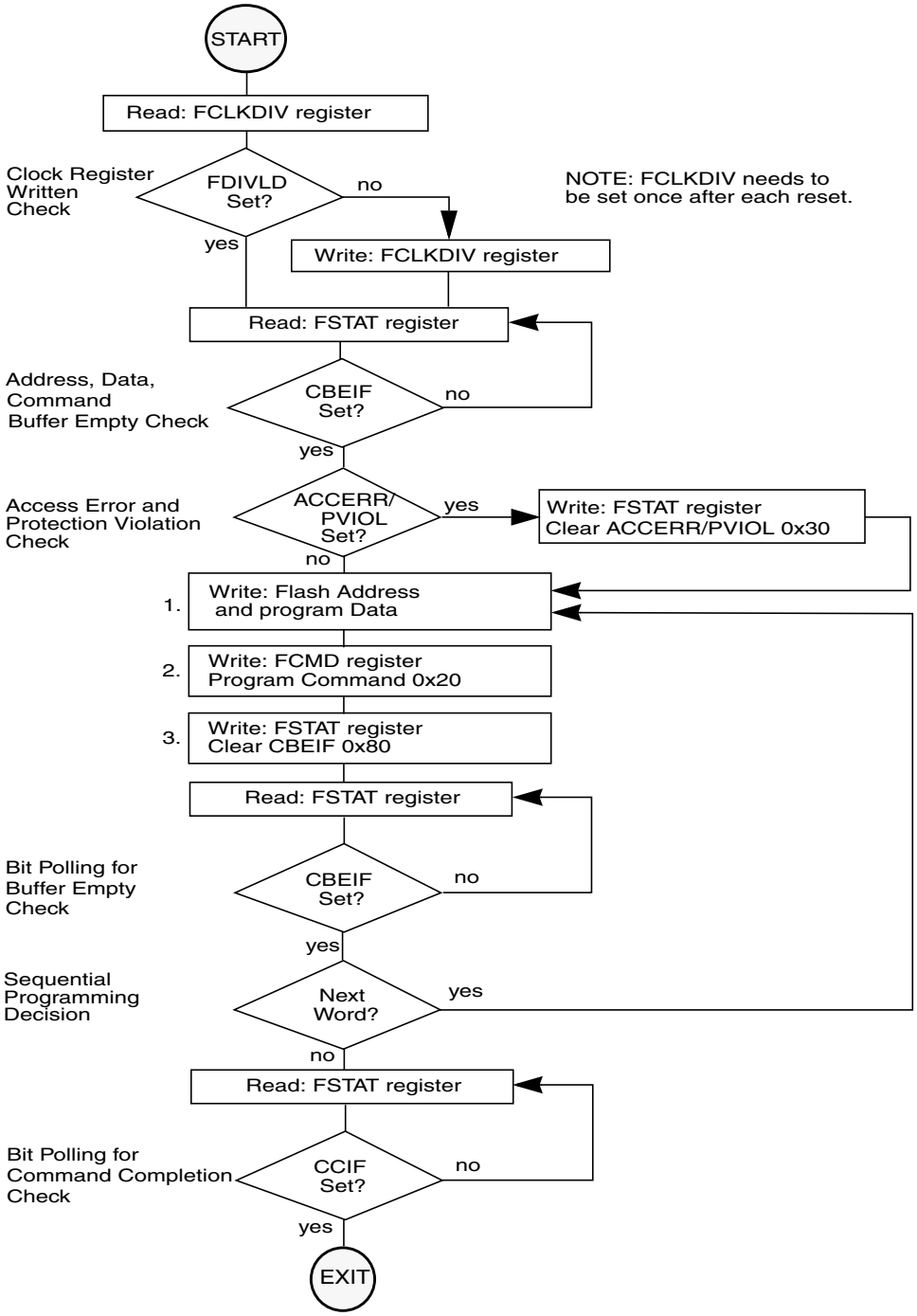


Figure 18-23. Example Program Command Flow

Table 19-10. Flash Protection Function

FPOPEN	FPHDIS	FPHS[1]	FPHS[0]	FPLDIS	FPLS[1]	FPLS[0]	Function(1)
1	1	x	x	1	x	x	No protection
1	1	x	x	0	x	x	Protect low range
1	0	x	x	1	x	x	Protect high range
1	0	x	x	0	x	x	Protect high and low ranges
0	1	x	x	1	x	x	Full Flash array protected
0	0	x	x	1	x	x	Unprotected high range
0	1	x	x	0	x	x	Unprotected low range
0	0	x	x	0	x	x	Unprotected high and low ranges

1. For range sizes refer to Table 19-11 and Table 19-12 or .

Table 19-11. Flash Protection Higher Address Range

FPHS[1:0]	Address Range	Range Size
00	0xF800–0xFFFF	2 Kbytes
01	0xF000–0xFFFF	4 Kbytes
10	0xE000–0xFFFF	8 Kbytes
11	0xC000–0xFFFF	16 Kbytes

Table 19-12. Flash Protection Lower Address Range

FPLS[1:0]	Address Range	Range Size
00	0x4000–0x43FF	1 Kbyte
01	0x4000–0x47FF	2 Kbytes
10	0x4000–0x4FFF	4 Kbytes
11	0x4000–0x5FFF	8 Kbytes

Figure 19-11 illustrates all possible protection scenarios. Although the protection scheme is loaded from the Flash array after reset, it is allowed to change in normal modes. This protection scheme can be used by applications requiring re-programming in single chip mode while providing as much protection as possible if no re-programming is required.

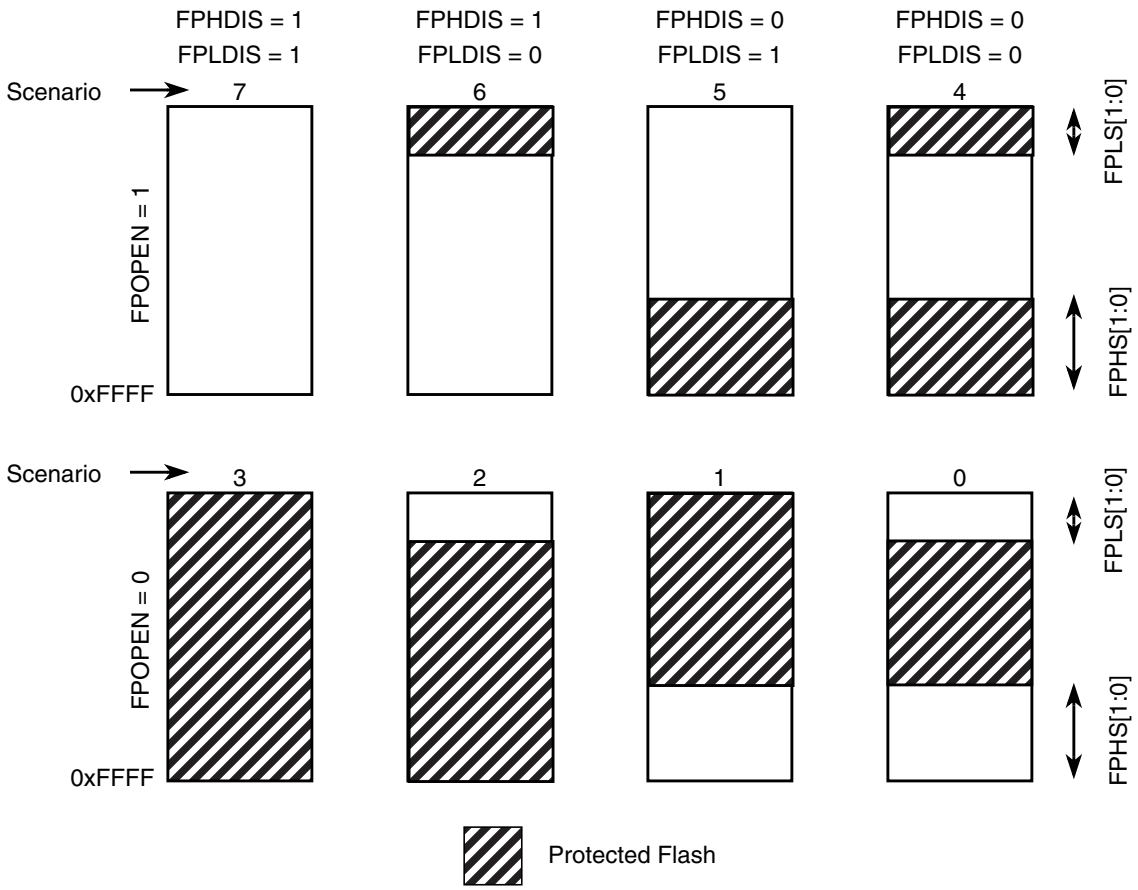


Figure 19-11. Flash Protection Scenarios

19.3.2.5.1 Flash Protection Restrictions

The general guideline is that protection can only be added, not removed. All valid transitions between Flash protection scenarios are specified in Table 19-13. Any attempt to write an invalid scenario to the FPROT register will be ignored and the FPROT register will remain unchanged. The contents of the FPROT register reflect the active protection scenario.

Table 19-13. Flash Protection Scenario Transitions

From Protection Scenario	To Protection Scenario[1]							
	0	1	2	3	4	5	6	7
0	X	X	X	X				
1		X		X				
2			X	X				
3				X				
4				X	X			
5			X	X	X	X		

Module Base + 0x000F

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

= Unimplemented or Reserved

Figure 20-22. RESERVED6

All bits read 0 and are not writable.

20.4 Functional Description

20.4.1 Flash Command Operations

Write operations are used for the program, erase, and erase verify algorithms described in this section. The program and erase algorithms are controlled by a state machine whose timebase FCLK is derived from the oscillator clock via a programmable divider. The FCMD register as well as the associated FADDR and FDATA registers operate as a buffer and a register (2-stage FIFO) so that a new command along with the necessary data and address can be stored to the buffer while the previous command is still in progress. This pipelined operation allows a time optimization when programming more than one word on a specific row, as the high voltage generation can be kept active in between two programming commands. The pipelined operation also allows a simplification of command launching. Buffer empty as well as command completion are signalled by flags in the FSTAT register with corresponding interrupts generated, if enabled.

The next sections describe:

- How to write the FCLKDIV register
- Command write sequence used to program, erase or erase verify the Flash array
- Valid Flash commands
- Errors resulting from illegal Flash operations

20.4.1.1 Writing the FCLKDIV Register

Prior to issuing any Flash command after a reset, it is first necessary to write the FCLKDIV register to divide the oscillator clock down to within the 150-kHz to 200-kHz range. Since the program and erase timings are also a function of the bus clock, the FCLKDIV determination must take this information into account.

If we define:

- FCLK as the clock of the Flash timing control block
- Tbus as the period of the bus clock
- INT(x) as taking the integer part of x (e.g., INT(4.323) = 4),

20.4.1.3.4 Mass Erase Command

The mass erase operation will erase all addresses in a Flash array using an embedded algorithm.

An example flow to execute the mass erase operation is shown in Figure 20-27. The mass erase command write sequence is as follows:

1. Write to a Flash array address to start the command write sequence for the mass erase command. The address and data written will be ignored.
2. Write the mass erase command, 0x41, to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the mass erase command.

If a Flash array to be erased contains any protected area, the PVIOL flag in the FSTAT register will set and the mass erase command will not launch. Once the mass erase command has successfully launched, the CCIF flag in the FSTAT register will set after the mass erase operation has completed unless a new command write sequence has been buffered.