



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	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	SCI
Peripherals	LVD, POR, PWM
Number of I/O	16
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 7x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mchc908mr8mfae

7.7.2	Stop Mode	106
7.7.3	SIM Break Status Register	106
7.7.4	SIM Reset Status Register	108
7.7.5	SIM Break Flag Control Register	109

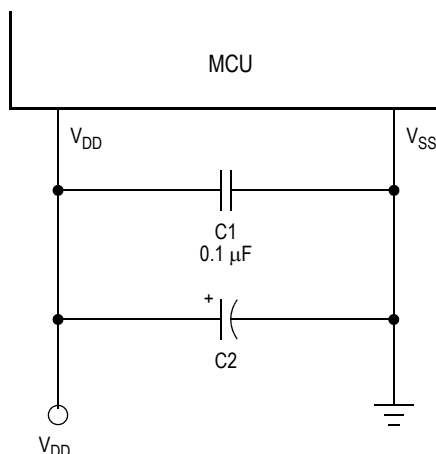
Section 8. Clock Generator Module (CGM)

8.1	Contents	111
8.2	Introduction	112
8.3	Features	112
8.4	Functional Description	113
8.4.1	Crystal Oscillator Circuit	113
8.4.2	Phase-Locked Loop Circuit (PLL)	115
8.4.2.1	PLL Circuits	115
8.4.2.2	Acquisition and Tracking Modes	118
8.4.2.3	Manual and Automatic PLL Bandwidth Modes	118
8.4.2.4	Programming the PLL	120
8.4.2.5	Special Programming Exceptions	121
8.4.3	Base Clock Selector Circuit	121
8.4.4	CGM External Connections	122
8.5	I/O Signals	123
8.5.1	Crystal Amplifier Input Pin (OSC1)	123
8.5.2	Crystal Amplifier Output Pin (OSC2)	123
8.5.3	External Filter Capacitor Pin (CGMXFC)	123
8.5.4	PLL Analog Power Pin (V_{DDA})	124
8.5.5	Oscillator Enable Signal (SIMOSCEN)	124
8.5.6	Crystal Output Frequency Signal (CGMXCLK)	124
8.5.7	CGM Base Clock Output (CGMOUT)	124
8.5.8	CGM CPU Interrupt (CGMINT)	124
8.6	CGM Registers	125
8.6.1	PLL Control Register	126
8.6.2	PLL Bandwidth Control Register	129
8.6.3	PLL Programming Register	131
8.7	Interrupts	132
8.8	Wait Mode	133
8.9	Stop Mode	133

1.5.1 Power Supply Pins (V_{DD} and V_{SS})

V_{DD} and V_{SS} are the power supply and ground pins. The MCU operates from a single power supply.

Fast signal transitions on MCU pins place high, short-duration current demands on the power supply. To prevent noise problems, take special care to provide power supply bypassing at the MCU as [Figure 1-3](#) shows. Place the C1 bypass capacitor as close to the MCU as possible. Use a high-frequency-response ceramic capacitor for C1. C2 is an optional bulk current bypass capacitor for use in applications that require the port pins to source high current levels.



Note: Component values shown represent typical applications.

Figure 1-3. Power Supply Bypassing

1.5.2 Oscillator Pins (OSC1 and OSC2)

The OSC1 and OSC2 pins are the connections for the on-chip oscillator circuit. See [Section 8. Clock Generator Module \(CGM\)](#).

1.5.3 External Reset Pin (\overline{RST})

A logic 0 on the \overline{RST} pin forces the MCU to a known startup state. \overline{RST} is bidirectional, allowing a reset of the entire system. It is driven low when any internal reset source is asserted. See [Section 7. System Integration Module \(SIM\)](#).

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$003F	IRQ Status/Control Register (ISCR) See page 303.	Read:	0	0	0	0	IRQF	0	IMASK1	MODE1
		Write:	R	R	R	R		ACK1		
		Reset:	0	0	0	0	0	0	0	0
\$0040	ADC Status and Control Register (ADSCR) See page 319.	Read:	COCO	AIEN	ADCO	ADCH4	ADCH3	ADCH2	ADCH1	ADCH0
		Write:								
		Reset:	0	0	0	1	1	1	1	1
\$0041	ADC Data Register High (ADRH) See page 322.	Read:	0	0	0	0	0	0	AD9	AD8
		Write:	R	R	R	R	R	R	R	R
		Reset:	Unaffected by reset							
\$0042	ADC Data Register Low (ADRL) See page 323.	Read:	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
		Write:	R	R	R	R	R	R	R	R
		Reset:	Unaffected by reset							
\$0043	ADC Clock Register (ADCLK) See page 324.	Read:	ADIV2	ADIV1	ADIV0	ADICLK	MODE1	MODE0	0	0
		Write:							0	R
		Reset:	0	0	0	0	0	1	0	0
\$0044			Unimplemented							
↓										
\$0050			Unimplemented							
\$0051	TIMB Status/Control Register (TBSC) See page 238.	Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
		Write:	0			TRST	R			
		Reset:	0	0	1	0	0	0	0	0
\$0052	TIMB Counter Register High (TBCNTH) See page 240.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
\$0053	TIMB Counter Register Low (TBCNTL) See page 240.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:	R	R	R	R	R	R	R	R
		Reset:	0	0	0	0	0	0	0	0
\$0054	TIMB Counter Modulo Register High (TBMODH) See page 241.	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1
U = Unaffected X = Indeterminate			R	= Reserved		Bold	= Buffered		= Unimplemented	

Figure 2-2. Control, Status, and Data Registers (Sheet 7 of 10)



Section 5. Configuration Register (CONFIG)

5.1 Contents

5.2	Introduction	67
5.3	CONFIG	68
5.4	CONFIG Bits	68

5.2 Introduction

This section describes the configuration register (CONFIG).

The CONFIG registers contain bits that configure these options:

- Resets caused by the low-voltage inhibit (LVI) module
- Power to the LVI module
- Computer operating properly (COP) module
- Top-side pulse-width modulator (PWM) polarity
- Bottom-side PWM polarity
- Edge-aligned versus center-aligned PWMs
- Six independent PWMs versus three complementary PWM pairs
- STOP instruction enable

6.3 Features

Features of the CPU include:

- Fully upward, object-code compatibility with M68HC05 family
- 16-bit stack pointer with stack manipulation instructions
- 16-bit index register with X-register manipulation instructions
- 8-MHz CPU internal bus frequency
- 64-Kbyte program/data memory space
- Sixteen addressing modes
- Memory-to-memory data moves without using the accumulator
- Fast 8-bit by 8-bit multiply and 16-bit by 8-bit divide instructions
- Enhanced binary-coded decimal (BCD) data handling
- Modular architecture with expandable internal bus definition for extension of addressing range beyond 64 Kbytes
- Low-power stop and wait modes

6.4 CPU Registers

Figure 6-1 shows the five CPU registers. CPU registers are not part of the memory map.

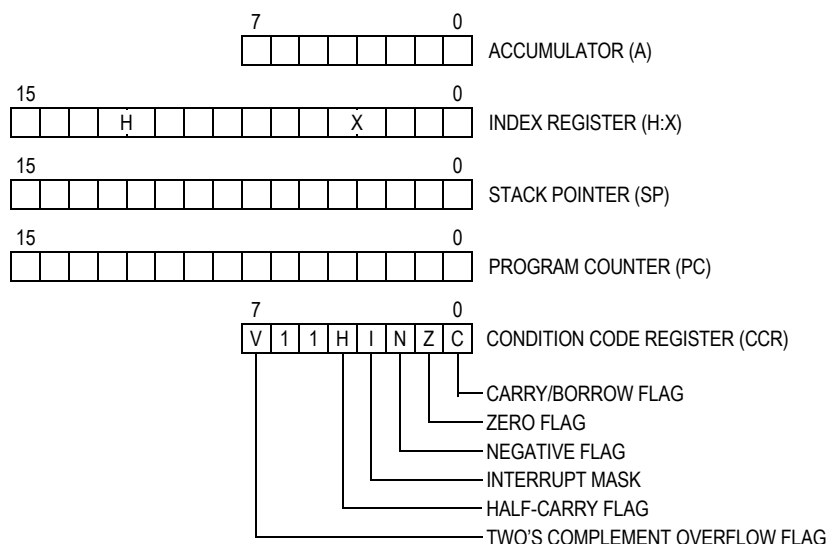


Figure 6-1. CPU Registers

6.4.1 Accumulator

The accumulator is a general-purpose 8-bit register. The CPU uses the accumulator to hold operands and the results of arithmetic/logic operations.

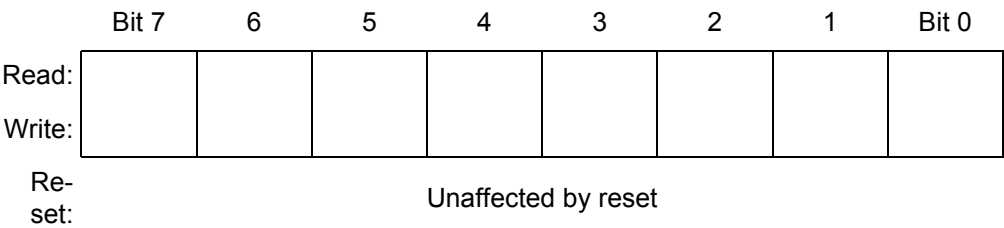


Figure 6-2. Accumulator (A)

6.4.2 Index Register

The 16-bit index register allows indexed addressing of a 64-Kbyte memory space. H is the upper byte of the index register, and X is the lower byte. H:X is the concatenated 16-bit index register.

In the indexed addressing modes, the CPU uses the contents of the index register to determine the conditional address of the operand.

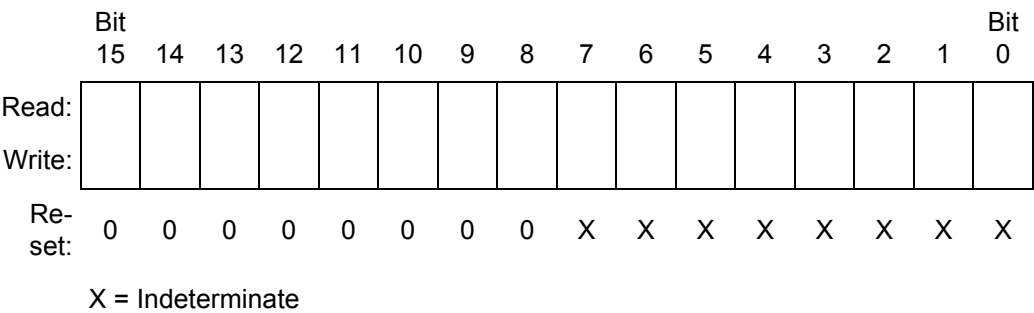


Figure 6-3. Index Register (H:X)

The index register can serve also as a temporary data storage location.

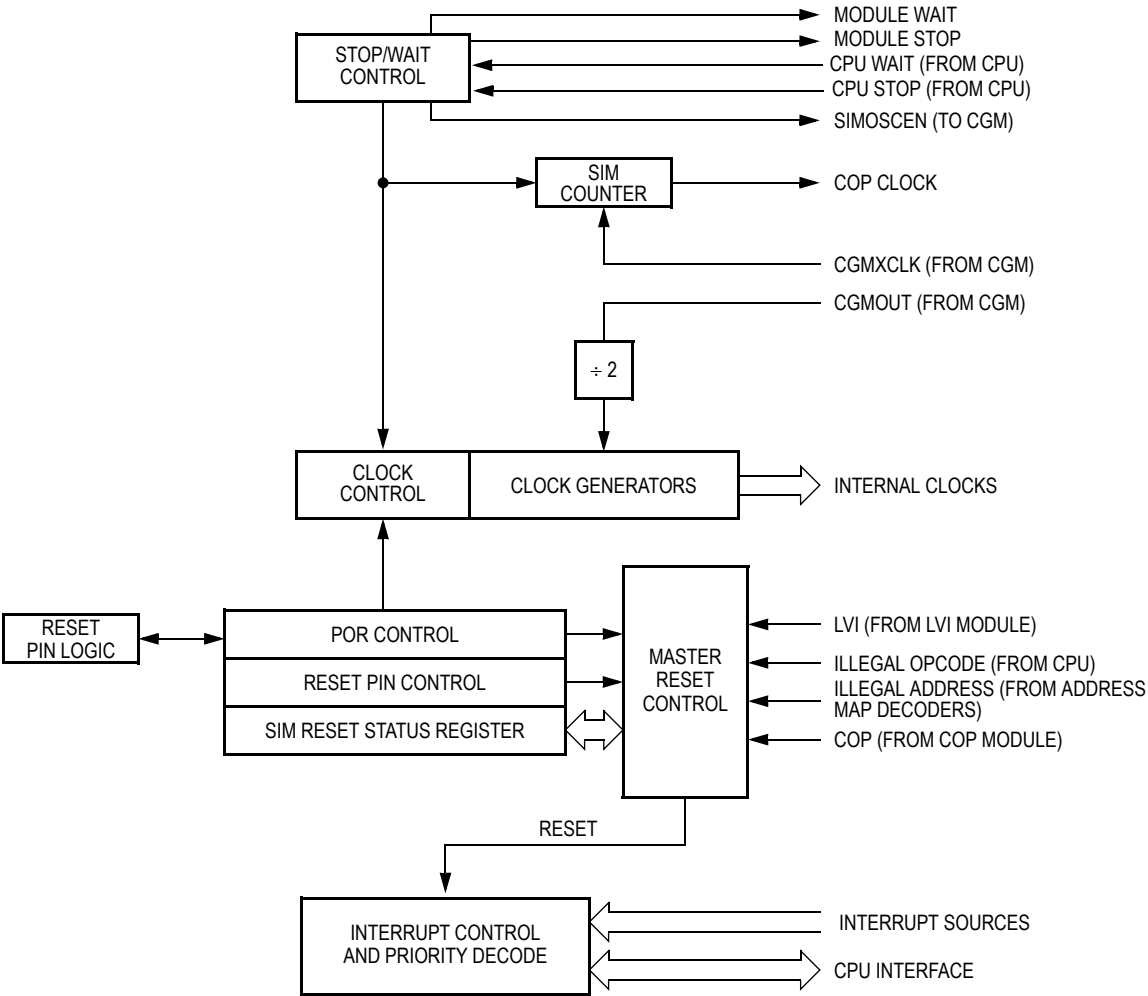


Figure 7-1. SIM Block Diagram

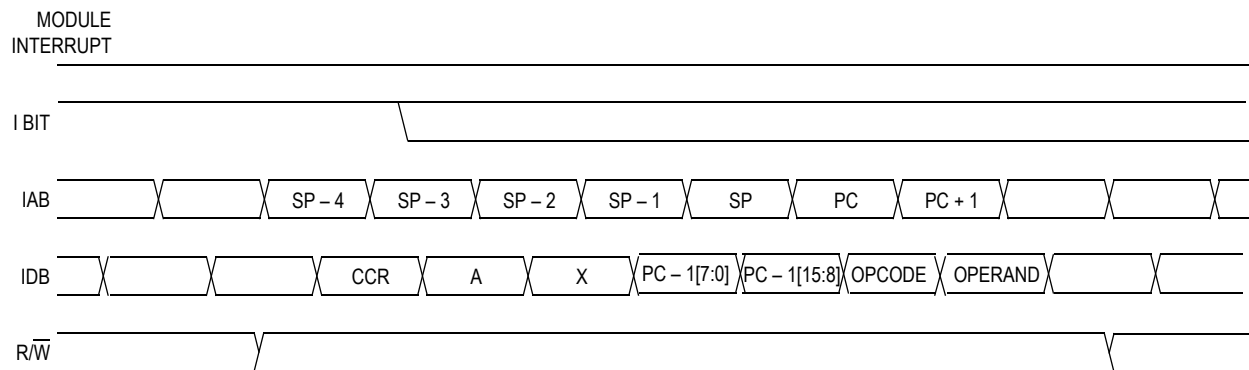


Figure 7-10. Interrupt Recovery

7.6.1.1 Hardware Interrupts

A hardware interrupt does not stop the current instruction. Processing of a hardware interrupt begins after completion of the current instruction. When the current instruction is complete, the SIM checks all pending hardware interrupts. If interrupts are not masked (I bit clear in the condition code register), and if the corresponding interrupt enable bit is set, the SIM proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If more than one interrupt is pending at the end of an instruction execution, the highest priority interrupt is serviced first. [Figure 7-11](#) demonstrates what happens when two interrupts are pending. If an interrupt is pending upon exit from the original interrupt service routine, the pending interrupt is serviced before the LDA instruction is executed.

The LDA opcode is prefetched by both the INT1 and INT2 RTI instructions. However, in the case of the INT1 RTI prefetch, this is a redundant operation.

NOTE: *To maintain compatibility with the M6805 Family, the H register is not pushed on the stack during interrupt entry. If the interrupt service routine modifies the H register or uses the indexed addressing mode, software should save the H register and then restore it prior to exiting the routine.*

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
\$005C	PLL Control Register (PCTL) See page 126.	Read :		PLL F			1	1	1	1
		Write :	PLL IE	R	PLL ON	BCS	R	R	R	R
		Re-set:	0	0	1	0	1	1	1	1
\$005D	PLL Bandwidth Control Register (PBWC) See page 129.	Read :		LOCK			0	0	0	0
		Write :	AUTO	R	ACQ	XLD	R	R	R	R
		Re-set:	0	0	0	0	0	0	0	0
\$005E	PLL Programming Register (PPG) See page 131.	Read :								
		Write :	MUL 7	MUL 6	MUL 5	MUL 4	VRS 7	VRS 6	VRS 5	VRS 4
		Re-set:	0	1	1	0	0	1	1	0

R = Reserved

Figure 8-2. CGM I/O Register Summary

8.4.2 Phase-Locked Loop Circuit (PLL)

The PLL is a frequency generator that can operate in either acquisition mode or tracking mode, depending on the accuracy of the output frequency. The PLL can change between acquisition and tracking modes either automatically or manually.

8.4.2.1 PLL Circuits

The PLL consists of these circuits:

- Voltage-controlled oscillator (VCO)
- Modulo VCO frequency divider

VRS[7:4] — VCO Range Select Bits

These read/write bits control the hardware center-of-range linear multiplier L , which controls the hardware center-of-range frequency f_{VRS} . See [8.4.2.1 PLL Circuits](#), [8.4.2.4 Programming the PLL](#), and [8.6.1 PLL Control Register](#). VRS[7:4] cannot be written when the PLLON bit in the PLL control register (PCTL) is set. See [8.4.2.5 Special Programming Exceptions](#). A value of \$0 in the VCO range select bits disables the PLL and clears the BCS bit in the PCTL. See [8.4.3 Base Clock Selector Circuit](#) and [8.4.2.5 Special Programming Exceptions](#) for more information.

Reset initializes the bits to \$6 to give a default range multiply value of 6.

NOTE: *The VCO range select bits have built-in protection that prevents them from being written when the PLL is on (PLLON = 1) and prevents selection of the VCO clock as the source of the base clock (BCS = 1) if the VCO range select bits are all clear.*

The VCO range select bits must be programmed correctly. Incorrect programming may result in failure of the PLL to achieve lock.

8.7 Interrupts

When the AUTO bit is set in the PLL bandwidth control register (PBWC), the PLL can generate a CPU interrupt request every time the LOCK bit changes state. The PLLIE bit in the PLL control register (PCTL) enables CPU interrupts from the PLL. PLLF, the interrupt flag in the PCTL, becomes set whether interrupts are enabled or not. When the AUTO bit is clear, CPU interrupts from the PLL are disabled and PLLF reads as logic 0.

Software should read the LOCK bit after a PLL interrupt request to see if the request was due to an entry into lock or an exit from lock. When the PLL enters lock, the VCO clock, CGMVCLK, divided by two can be selected as the CGMOUT source by setting BCS in the PCTL. When the PLL exits lock, the VCO clock frequency is corrupt, and appropriate precautions should be taken. If the application is not

frequency-sensitive, interrupts should be disabled to prevent PLL interrupt service routines from impeding software performance or from exceeding stack limitations.

NOTE: *Software can select the CGMVCLK divided by two as the CGMOUT source even if the PLL is not locked (LOCK = 0). Therefore, software should make sure the PLL is locked before setting the BCS bit.*

8.8 Wait Mode

The WAIT instruction puts the MCU in low power-consumption standby mode.

The WAIT instruction does not affect the CGM. Before entering wait mode, software can disengage and turn off the PLL by clearing the BCS and PLLON bits in the PLL control register (PCTL). Less power-sensitive applications can disengage the PLL without turning it off. Applications that require the PLL to wake the MCU from wait mode also can deselect the PLL output without turning off the PLL.

8.9 Stop Mode

The STOP instruction puts the MCU in low power-consumption standby mode.

The STOP instruction disables the CGMC (oscillator and phase-lock loop) and holds the CGM outputs low.

8.10 CGM During Break Mode

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the SIM break flag control register (SBFCR) enables software to clear status bits during the break state. See [7.7.5 SIM Break Flag Control Register](#).

To allow software to clear status bits during a break interrupt, write a logic 1 to the BCFE bit. If a status bit is cleared during the break state, it remains cleared when the MCU exits the break state.

Pulse-Width Modulator for Motor Control

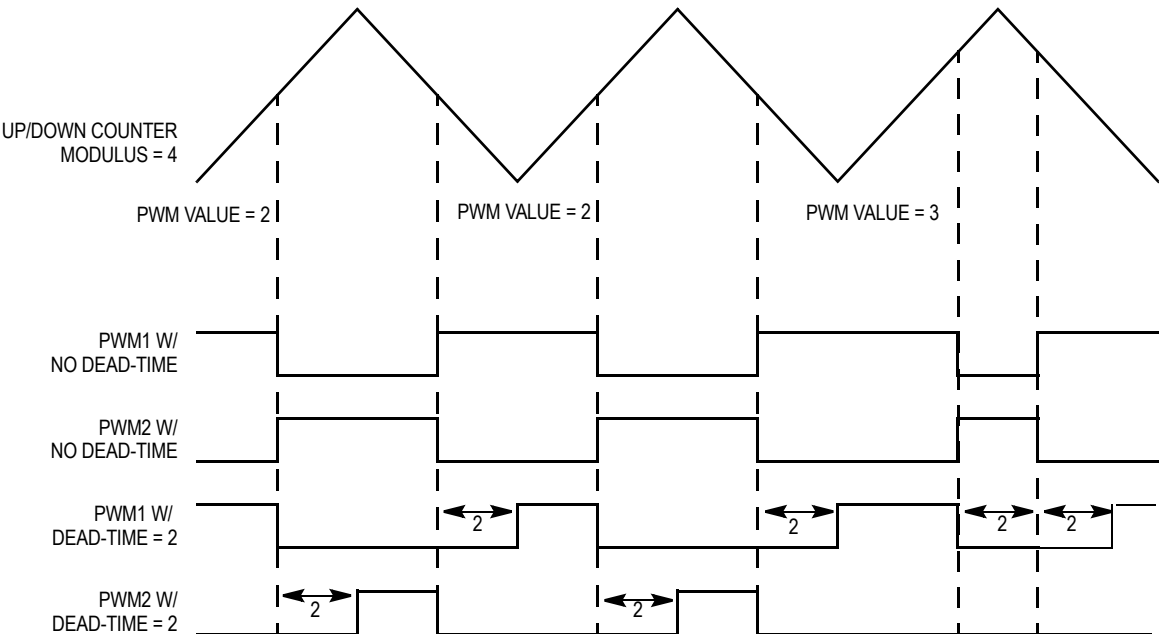


Figure 9-14. Effects of Dead-Time Insertion

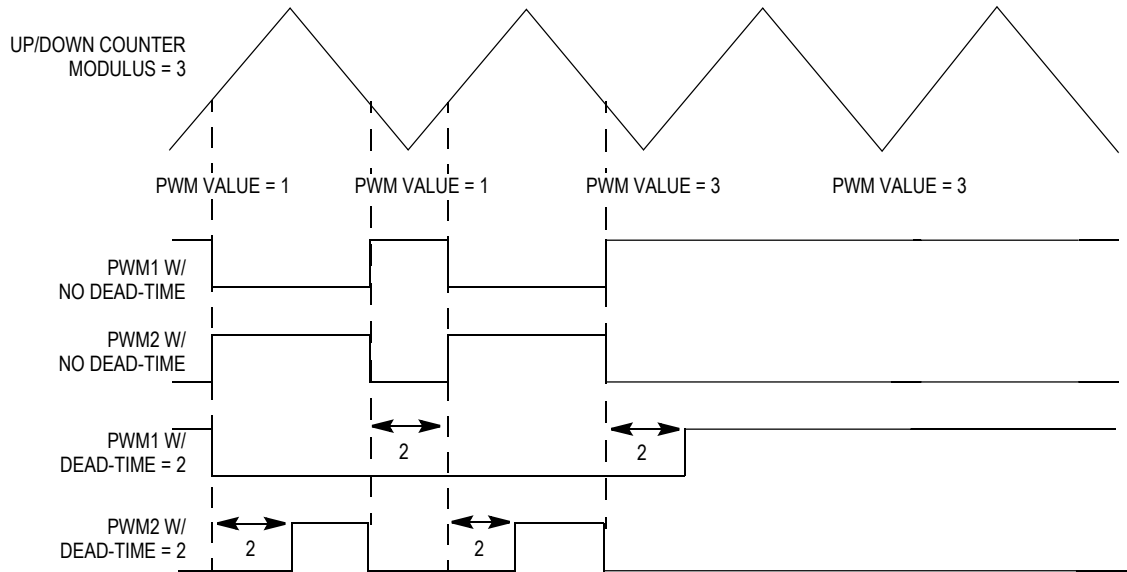


Figure 9-15. Dead-Time at Duty Cycle Boundaries

9.11 PWM Operation in Break Mode

If the microcontroller goes into break mode (or background mode), the clocks to the PWM generator and output control blocks will freeze. This allows the user to set a breakpoint on a development system and examine the register contents and PWM outputs at that point. It also allows the user to single-step through the code.

The clocks to the fault block will continue to run. Therefore, if a fault occurs while the microcontroller is in break mode, the PWM outputs will immediately be driven to their inactive state(s).

During break mode, the system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the SIM break flag control register (SBFCR) enables software to clear status bits during the break state. Refer to [7.7.5 SIM Break Flag Control Register](#).

To allow software to clear status bits during a break interrupt, write a logic 1 to the BCFE bit. If a status bit is cleared during the break state, it remains cleared when the MCU exits the break state.

To protect the PWMF and FFLAGx bits during the break state, make sure BCFE is a logic 0. With BCFE at logic 0 (its default state), software can read and write the status and control registers during the break state without affecting the PWMF and FFLAGx bits.

NOTE: When PWMINT is cleared, pending CPU interrupts are inhibited.

PWMF— PWM Reload Flag

This read/write bit is set at the beginning of every reload cycle regardless of the state of the LDOK bit. This bit is cleared by reading PWM control register 1 with the PWMF flag set, then writing a logic 0 to PWMF. If another reload occurs before the clearing sequence is complete, then writing logic 0 to PWMF has no effect.

1 = New reload cycle began.

0 = New reload cycle has not begun.

NOTE: When PWMF is cleared, pending PWM CPU interrupts are cleared (excluding fault interrupts).

CAUTION: Bits 2 and/or 3 of PCTL1 are reserved and must never be set to a 1. Setting these bits to a 1 will affect the active PWM value registers. Undesirable results will occur.

LDOK— Load OK Bit

This read/write bit allows the counter modulus, counter prescaler, and PWM values in the buffered registers to be used by the PWM generator. These values will not be used until the LDOK bit is set and a new PWM load cycle begins. LDOK may be cleared, if it is set, by writing a logic 0 to it prior to the beginning of a new PWM load cycle. Internally this bit is automatically cleared after the new values are loaded.

1 = Okay to load new modulus, prescaler, and PWM values at beginning of next PWM load cycle

0 = Not okay to load new modulus, prescaler, and PWM values

NOTE: The user should initialize the PWM registers and set the LDOK bit before enabling the PWM.

PWMEN — PWM Module Enable Bit

This read/write bit enables and disables the PWM generator and the PWM pins. When PWMEN is clear, the PWM generator is disabled and the PWM pins are in the high-impedance state (unless OUTCTL = 1). When the PWMEN bit is set, the PWM generator and PWM pins are activated. For more information, see [9.8 Initialization and the PWMEN Bit](#).

1 = PWM generator and PWM pins enabled

0 = PWM generator and PWM pins disabled

FTACK1 — Fault Acknowledge 1 Bit

The FTACK1 bit is used to acknowledge and clear FFLAG1. This bit will always read 0. Writing a 1 to this bit will clear FFLAG1. Writing a 0 will have no effect.

FTACK4 — Fault Acknowledge 4 Bit

The FTACK4 bit is used to acknowledge and clear FFLAG4. This bit will always read 0. Writing a 1 to this bit will clear FFLAG4. Writing a 0 will have no effect.

Table 10-7. READSP (Read Stack Pointer) Command

Description	Reads stack pointer
Operand	None
Data returned	Returns stack pointer in high byte:low byte order
Opcode	\$0C
Command sequence	

SENT TO MONITOR

ECHO

RESULT

Table 10-8. RUN (Run User Program) Command

Description	Executes RTI instruction
Operand	None
Data returned	None
Opcode	\$28
Command sequence	

SENT TO MONITOR

ECHO

CHxF — Channel x Flag Bit

When channel x is an input capture channel, this read/write bit is set when an active edge occurs on the channel x pin. When channel x is an output compare channel, CHxF is set when the value in the TIMA counter registers matches the value in the TIMA channel x registers.

When CHxIE = 1, clear CHxF by reading TIMA channel x status and control register with CHxF set, and then writing a logic 0 to CHxF. If another interrupt request occurs before the clearing sequence is complete, then writing logic 0 to CHxF has no effect. Therefore, an interrupt request cannot be lost due to inadvertent clearing of CHxF.

Reset clears the CHxF bit. Writing a logic 1 to CHxF has no effect.

- 1 = Input capture or output compare on channel x
- 0 = No input capture or output compare on channel x

CHxIE — Channel x Interrupt Enable Bit

This read/write bit enables TIMA CPU interrupts on channel x. Reset clears the CHxIE bit.

- 1 = Channel x CPU interrupt requests enabled
- 0 = Channel x CPU interrupt requests disabled

MSxB — Mode Select Bit B

This read/write bit selects buffered output compare/PWM operation. MSxB exists only in the TIMA channel 0. Setting MS0B disables the channel 1 status and control register and reverts TCH1A to general-purpose I/O. Reset clears the MSxB bit.

- 1 = Buffered output compare/PWM operation enabled
- 0 = Buffered output compare/PWM operation disabled

MSxA — Mode Select Bit A

When ELSxB:A ≠ 00, this read/write bit selects either input capture operation or unbuffered output compare/PWM operation. See [Table 11-2](#).

- 1 = Unbuffered output compare/PWM operation
- 0 = Input capture operation

When ELSxB:A = 00, this read/write bit selects the initial output level of the TCHx pin once PWM, input capture, or output compare operation is enabled. Reset clears the MSxA bit. See [Table 11-2](#).

- 1 = Initial output level low
- 0 = Initial output level high

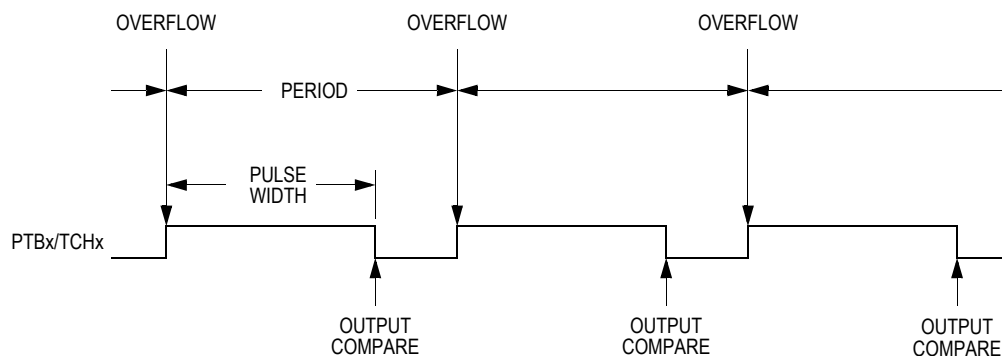


Figure 12-3. PWM Period and Pulse Width

The value in the TIMB counter modulo registers and the selected prescaler output determines the frequency of the PWM output. The frequency of an 8-bit PWM signal is variable in 256 increments. Writing \$00FF (255) to the TIMB counter modulo registers produces a PWM period of 256 times the internal bus clock period if the prescaler select value is \$000 (see [12.10.1 TIMB Status and Control Register](#)).

The value in the TIMB channel registers determines the pulse width of the PWM output. The pulse width of an 8-bit PWM signal is variable in 256 increments. Writing \$0080 (128) to the TIMB channel registers produces a duty cycle of 128/256 or 50 percent.

12.4.4.1 Unbuffered PWM Signal Generation

Any output compare channel can generate unbuffered PWM pulses as described in [12.4.4 Pulse-Width Modulation \(PWM\)](#). The pulses are unbuffered because changing the pulse width requires writing the new pulse width value over the value currently in the TIMB channel registers.

An unsynchronized write to the TIMB channel registers to change a pulse width value could cause incorrect operation for up to two PWM periods. For example, writing a new value before the counter reaches the old value but after the counter reaches the new value prevents any compare during that PWM period. Also, using a TIMB overflow interrupt routine to write a new, smaller pulse width value may cause the compare to be missed. The TIMB may pass the new value before it is written to the TIMB channel registers.

13.4.3.4 Framing Errors

If the data recovery logic does not detect a logic 1 where the stop bit should be in an incoming character, it sets the framing error bit, FE, in SCS1. The FE flag is set at the same time that the SCRF bit is set. A break character that has no stop bit also sets the FE bit.

13.4.3.5 Receiver Wakeup

So that the MCU can ignore transmissions intended only for other receivers in multiple-receiver systems, the receiver can be put into a standby state. Setting the receiver wakeup bit, RWU, in SCC2 puts the receiver into a standby state during which receiver interrupts are disabled.

Depending on the state of the WAKE bit in SCC1, either of two conditions on the PTB0/RxD pin can bring the receiver out of the standby state:

- **Address mark** — An address mark is a logic 1 in the most significant bit position of a received character. When the WAKE bit is set, an address mark wakes the receiver from the standby state by clearing the RWU bit. The address mark also sets the SCI receiver full bit, SCRF. Software can then compare the character containing the address mark to the user-defined address of the receiver. If they are the same, the receiver remains awake and processes the characters that follow. If they are not the same, software can set the RWU bit and put the receiver back into the standby state.
- **Idle input line condition** — When the WAKE bit is clear, an idle character on the PTB0/RxD pin wakes the receiver from the standby state by clearing the RWU bit. The idle character that wakes the receiver does not set the receiver idle bit, IDLE, or the SCI receiver full bit, SCRF. The idle line type bit, ILTY, determines whether the receiver begins counting logic 1s as idle character bits after the start bit or after the stop bit.

NOTE: *Clearing the WAKE bit after the PTB0/RxD pin has been idle can cause the receiver to wake up immediately.*

Serial Communications Interface (SCI)

13.9.6 SCI Data Register

The SCI data register (SCDR) is the buffer between the internal data bus and the receive and transmit shift registers. Reset has no effect on data in the SCI data register.

Ad- dress:	\$003D							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	R7	R6	R5	R4	R3	R2	R1	R0
Write:	T7	T6	T5	T4	T3	T2	T1	T0
Reset:	Unaffected by reset							

Figure 13-13. SCI Data Register (SCDR)

R7/T7:R0/T0 — Receive/Transmit Data Bits

Reading address \$003D accesses the read-only received data bits, R7:R0. Writing to address \$003D writes the data to be transmitted, T7:T0. Reset has no effect on the SCI data register.

13.9.7 SCI Baud Rate Register

The SCI baud rate register (SCBR) selects the baud rate for both the receiver and the transmitter.

Ad- dress:	\$003E							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	SCP1	SCP0	0	SCR2	SCR1	SCR0
Write:	R	R			R			
Reset:	0	0	0	0	0	0	0	0
	<div style="border: 1px solid black; display: inline-block; padding: 2px;">R</div> = Reserved							

Figure 13-14. SCI Baud Rate Register (SCBR)