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, SPI
Peripherals	LVD, POR, PWM
Number of I/O	24
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	28-TSSOP
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908qc16mdre

Table of Contents

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0024	TIM1 Counter Modulo Register Low (T1MODL) See page 200.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:	1	1	1	1	1	1	1	1
$0025	TIM1 Channel 0 Status and Control Register (T1SC0) See page 201.	Read: Write:	CH0F / 0	CH0IE	MS0B	MS0A	ELS0B	ELS0A	TOV0	CH0MAX
		Reset:	0	0	0	0	0	0	0	0
$0026	TIM1 Channel 0 Register High (T1CH0H) See page 204.	Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Reset:				Indeterminate after reset				
$0027	TIM1 Channel 0 Register Low (T1CH0L) See page 204.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:				Indeterminate after reset				
$0028	TIM1 Channel 1 Status and Control Register (T1SC1) See page 198.	Read: Write:	CH1F / 0	CH1IE	0	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
		Reset:	0	0	0	0	0	0	0	0
$0029	TIM1 Channel 1 Register High (T1CH1H) See page 204.	Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Reset:				Indeterminate after reset				
$002A	TIM1 Channel 1 Register Low (T1CH1L) See page 204.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:				Indeterminate after reset				
$002B ↓ $002F	Reserved									
$0030	TIM1 Channel 2 Status and Control Register (T1SC2) See page 201.	Read: Write:	CH2F / 0	CH2IE	0	MS2A	ELS2B	ELS2A	TOV2	CH2MAX
		Reset:	0	0	0	0	0	0	0	0
$0031	TIM1 Channel 2 Register High (T1CH2H) See page 204.	Read: Write:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Reset:				Indeterminate after reset				
$0032	TIM1 Channel 2 Register Low (T1CH2L) See page 204.	Read: Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Reset:				Indeterminate after reset				
$0033	TIM1 Channel 3 Status and Control Register (T1SC3) See page 201.	Read: Write:	CH3F / 0	CH3IE	0	MS3A	ELS3B	ELS3A	TOV3	CH3MAX
		Reset:	0	0	0	0	0	0	0	0

[ ] = Unimplemented   R = Reserved   U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 4 of 8)

**Algorithm for Programming
a Row (32 Bytes) of FLASH Memory**

1 SET PGM BIT

2 READ THE FLASH BLOCK PROTECT REGISTER

3 WRITE ANY DATA TO ANY FLASH ADDRESS WITHIN THE ROW ADDRESS RANGE DESIRED

4 WAIT FOR A TIME, t_{NVS}

5 SET HVEN BIT

6 WAIT FOR A TIME, t_{PGS}

7 WRITE DATA TO THE FLASH ADDRESS TO BE PROGRAMMED

8 WAIT FOR A TIME, t_{PROG}

9 COMPLETED PROGRAMMING THIS ROW?   Y   N

10 CLEAR PGM BIT

11 WAIT FOR A TIME, t_{NVH}

12 CLEAR HVEN BIT

13 WAIT FOR A TIME, t_{RCV}

END OF PROGRAMMING

NOTES:

The time between each FLASH address change (step 7 to step 7 loop),
or the time between the last FLASH address programmed
to clearing PGM bit (step 7 to step 10)
must not exceed the maximum programming
time, t_{PROG} max.

This row program algorithm assumes the row/s
to be programmed are initially erased.

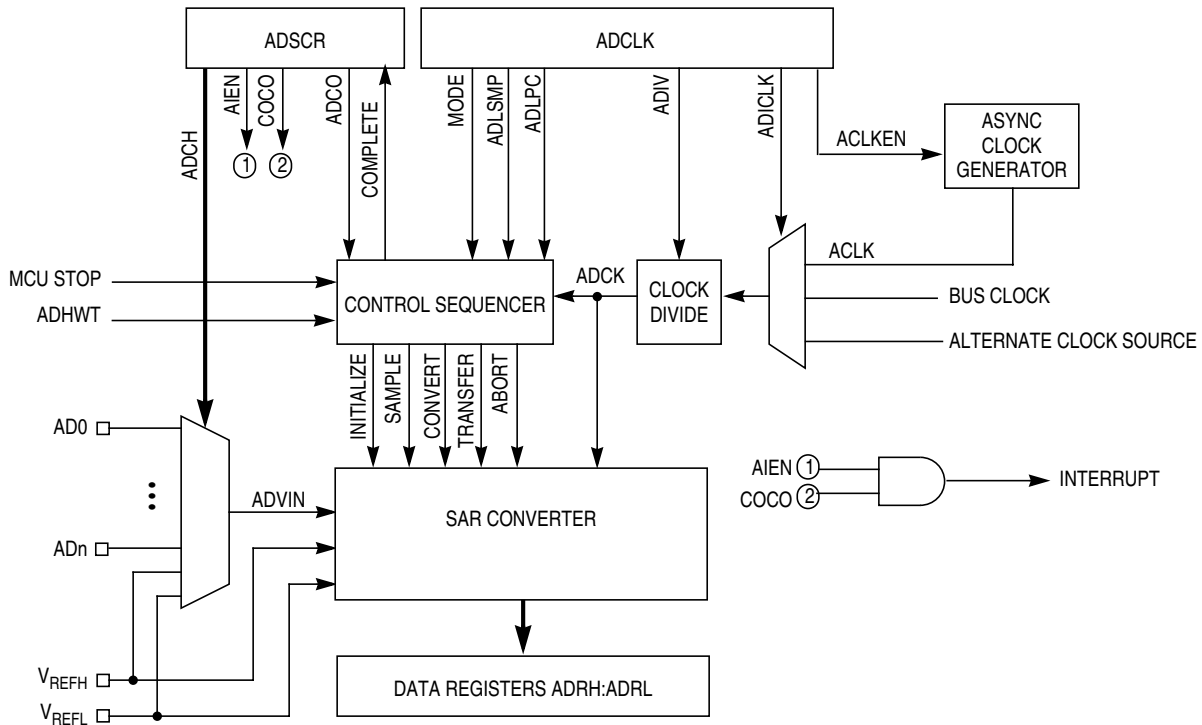**Figure 2-4. FLASH Programming Flowchart**

Figure 3-2. ADC10 Block Diagram

The ADC10 can perform an analog-to-digital conversion on one of the software selectable channels. The output of the input multiplexer (ADVIN) is converted by a successive approximation algorithm into a 10-bit digital result. When the conversion is completed, the result is placed in the data registers (ADRH and ADRL). In 8-bit mode, the result is rounded to 8 bits and placed in ADRL. The conversion complete flag is then set and an interrupt request is generated if AIEN has been set.

3.3.1 Clock Select and Divide Circuit

The clock select and divide circuit selects one of three clock sources and divides it by a configurable value to generate the input clock to the converter (ADCK). The clock can be selected from one of the following sources:

- The asynchronous clock source (ACLK) — This clock source is generated from a dedicated clock source which is enabled when the ADC10 is converting and the clock source is selected by setting ACLKEN. When ADLPC is clear, this clock operates from 1–2 MHz; when ADLPC is set, it operates at 0.5–1 MHz. This clock is not disabled in STOP and allows conversions in stop mode for lower noise operation.

- Alternate Clock Source — This clock source is equal to the external oscillator clock or four times the bus clock. The alternate clock source is MCU specific, see 3.1 Introduction to determine source and availability of this clock source option. This clock is selected when ADICLK and ACLKEN are both clear.

- The bus clock — This clock source is equal to the bus frequency. This clock is selected when ADICLK is set and ACLKEN is clear.

Whichever clock is selected, its frequency must fall within the acceptable frequency range for ADCK. If the available clocks are too slow, the ADC10 will not perform according to specifications. If the available

The COP counter is a free-running 6-bit counter preceded by the 12-bit system integration module (SIM) counter. If not cleared by software, the COP counter overflows and generates an asynchronous reset after 262,128 or 8176 BUSCLKX4 cycles; depending on the state of the COP rate select bit, COPRS, in configuration register 1. With a 262,128 BUSCLKX4 cycle overflow option, the internal 12.8-MHz oscillator gives a COP timeout period of 20.48 ms. Writing any value to location $FFFF before an overflow occurs prevents a COP reset by clearing the COP counter and stages 12–5 of the SIM counter.

> **NOTE**
> Service the COP immediately after reset and before entering or after exiting stop mode to guarantee the maximum time before the first COP counter overflow.

A COP reset pulls the $\overline{RST}$ pin low (if the RSTEN bit is set in the CONFIG1 register) for 32 × BUSCLKX4 cycles and sets the COP bit in the reset status register (RSR). See 14.8.1 SIM Reset Status Register.

> **NOTE**
> Place COP clearing instructions in the main program and not in an interrupt subroutine. Such an interrupt subroutine could keep the COP from generating a reset even while the main program is not working properly.

5.3 I/O Signals

The following paragraphs describe the signals shown in Figure 5-1.

5.3.1 BUSCLKX4

BUSCLKX4 is the oscillator output signal. BUSCLKX4 frequency is equal to the crystal frequency, internal oscillator frequency, or the RC-oscillator frequency.

5.3.2 STOP Instruction

The STOP instruction clears the SIM counter.

5.3.3 COPCTL Write

Writing any value to the COP control register (COPCTL) (see 5.4 COP Control Register) clears the COP counter and clears stages 12–5 of the SIM counter. Reading the COP control register returns the low byte of the reset vector.

5.3.4 Power-On Reset

The power-on reset (POR) circuit in the SIM clears the SIM counter 4096 × BUSCLKX4 cycles after power up.

5.3.5 Internal Reset

An internal reset clears the SIM counter and the COP counter.

5.3.6 COPD (COP Disable)

The COPD signal reflects the state of the COP disable bit (COPD) in the configuration register (CONFIG). See Chapter 4 Configuration Registers (CONFIG1 and CONFIG2).

Table 6-1. Instruction Set Summary (Sheet 2 of 6)

Source Form	Operation	Description	V	H	I	N	Z	C	Address Mode	Opcode	Operand	Cycles
BHS rel	Branch if Higher or Same (Same as BCC)	PC ← (PC) + 2 + rel ? (C) = 0	–	–	–	–	–	–	REL	24	rr	3
BIH rel	Branch if IRQ Pin High	PC ← (PC) + 2 + rel ? $\overline{IRQ}$ = 1	–	–	–	–	–	–	REL	2F	rr	3
BIL rel	Branch if IRQ Pin Low	PC ← (PC) + 2 + rel ? $\overline{IRQ}$ = 0	–	–	–	–	–	–	REL	2E	rr	3
BIT #opr BIT opr BIT opr BIT opr,X BIT opr,X BIT ,X BIT opr,SP BIT opr,SP	Bit Test	(A) & (M)	0	–	–	↕	↕	–	IMM DIR EXT IX2 IX1 IX SP1 SP2	A5 B5 C5 D5 E5 F5 9EE5 9ED5	ii dd hh ll ee ff ff ff ee ff	2 3 4 4 3 2 4 5
BLE opr	Branch if Less Than or Equal To (Signed Operands)	PC ← (PC) + 2 + rel ? (Z) | (N ⊕ V) = 1	–	–	–	–	–	–	REL	93	rr	3
BLO rel	Branch if Lower (Same as BCS)	PC ← (PC) + 2 + rel ? (C) = 1	–	–	–	–	–	–	REL	25	rr	3
BLS rel	Branch if Lower or Same	PC ← (PC) + 2 + rel ? (C) | (Z) = 1	–	–	–	–	–	–	REL	23	rr	3
BLT opr	Branch if Less Than (Signed Operands)	PC ← (PC) + 2 + rel ? (N ⊕ V) =1	–	–	–	–	–	–	REL	91	rr	3
BMC rel	Branch if Interrupt Mask Clear	PC ← (PC) + 2 + rel ? (I) = 0	–	–	–	–	–	–	REL	2C	rr	3
BMI rel	Branch if Minus	PC ← (PC) + 2 + rel ? (N) = 1	–	–	–	–	–	–	REL	2B	rr	3
BMS rel	Branch if Interrupt Mask Set	PC ← (PC) + 2 + rel ? (I) = 1	–	–	–	–	–	–	REL	2D	rr	3
BNE rel	Branch if Not Equal	PC ← (PC) + 2 + rel ? (Z) = 0	–	–	–	–	–	–	REL	26	rr	3
BPL rel	Branch if Plus	PC ← (PC) + 2 + rel ? (N) = 0	–	–	–	–	–	–	REL	2A	rr	3
BRA rel	Branch Always	PC ← (PC) + 2 + rel	–	–	–	–	–	–	REL	20	rr	3
BRCLR n,opr,rel	Branch if Bit n in M Clear	PC ← (PC) + 3 + rel ? (Mn) = 0	–	–	–	–	–	↕	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	01 03 05 07 09 0B 0D 0F	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BRN rel	Branch Never	PC ← (PC) + 2	–	–	–	–	–	–	REL	21	rr	3
BRSET n,opr,rel	Branch if Bit n in M Set	PC ← (PC) + 3 + rel ? (Mn) = 1	–	–	–	–	–	↕	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	00 02 04 06 08 0A 0C 0E	dd rr dd rr dd rr dd rr dd rr dd rr dd rr dd rr	5 5 5 5 5 5 5 5
BSET n,opr	Set Bit n in M	Mn ← 1	–	–	–	–	–	–	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	10 12 14 16 18 1A 1C 1E	dd dd dd dd dd dd dd dd	4 4 4 4 4 4 4 4
BSR rel	Branch to Subroutine	PC ← (PC) + 2; push (PCL) SP ← (SP) − 1; push (PCH) SP ← (SP) − 1 PC ← (PC) + rel	–	–	–	–	–	–	REL	AD	rr	4
CBEQ opr,rel CBEQA #opr,rel CBEQX #opr,rel CBEQ opr,X+,rel CBEQ X+,rel CBEQ opr,SP,rel	Compare and Branch if Equal	PC ← (PC) + 3 + rel ? (A) − (M) = $00 PC ← (PC) + 3 + rel ? (A) − (M) = $00 PC ← (PC) + 3 + rel ? (X) − (M) = $00 PC ← (PC) + 3 + rel ? (A) − (M) = $00 PC ← (PC) + 2 + rel ? (A) − (M) = $00 PC ← (PC) + 4 + rel ? (A) − (M) = $00	–	–	–	–	–	–	DIR IMM IMM IX1+ IX+ SP1	31 41 51 61 71 9E61	dd rr ii rr ii rr ff rr rr ff rr	5 4 4 5 4 6
CLC	Clear Carry Bit	C ← 0	–	–	–	–	–	0	INH	98		1
CLI	Clear Interrupt Mask	I ← 0	–	–	0	–	–	–	INH	9A		2

Fast Data Tolerance

Figure 13-8 shows how much a fast received character can be misaligned without causing a noise error or a framing error. The fast stop bit ends at RT10 instead of RT16 but is still there for the stop bit data samples at RT8, RT9, and RT10.
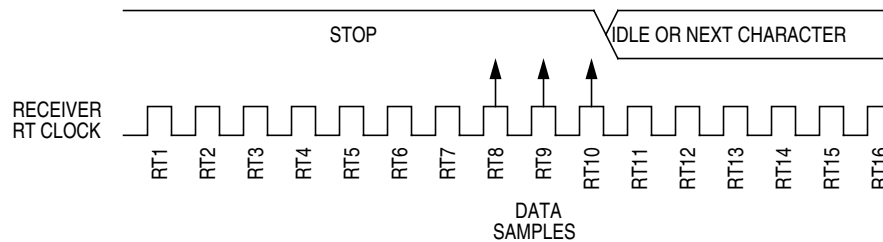

Figure 13-8. Fast Data

For an 8-bit character, data sampling of the stop bit takes the receiver
9 bit times × 16 RT cycles + 10 RT cycles = 154 RT cycles.

With the misaligned character shown in Figure 13-8, the receiver counts 154 RT cycles at the point when the count of the transmitting device is 10 bit times × 16 RT cycles = 160 RT cycles.

The maximum percent difference between the receiver count and the transmitter count of a fast 8-bit character with no errors is

$$\left| \frac{154 - 160}{154} \right| \times 100 = 3.90\%.$$

For a 9-bit character, data sampling of the stop bit takes the receiver
10 bit times × 16 RT cycles + 10 RT cycles = 170 RT cycles.

With the misaligned character shown in Figure 13-8, the receiver counts 170 RT cycles at the point when the count of the transmitting device is 11 bit times × 16 RT cycles = 176 RT cycles.

The maximum percent difference between the receiver count and the transmitter count of a fast 9-bit character with no errors is:

$$\left| \frac{170 - 176}{170} \right| \times 100 = 3.53\%.$$

13.3.3.6 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 RxD pin can bring the receiver out of the standby state:

1. Address mark — An address mark is a 1 in the MSB 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 ESCI 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.

13.7.2 ESCI Receive Data (RxD)

The RxD pin is the serial data input to the ESCI receiver. When the ESCI is enabled, the RxD pin becomes an input.

13.8 Registers

The following registers control and monitor operation of the ESCI:

- ESCI control register 1, SCC1
- ESCI control register 2, SCC2
- ESCI control register 3, SCC3
- ESCI status register 1, SCS1
- ESCI status register 2, SCS2
- ESCI data register, SCDR
- ESCI baud rate register, SCBR
- ESCI prescaler register, SCPSC
- ESCI arbiter control register, SCIACTL
- ESCI arbiter data register, SCIADAT

13.8.1 ESCI Control Register 1

ESCI control register 1 (SCC1):

- Enables loop mode operation
- Enables the ESCI
- Controls output polarity
- Controls character length
- Controls ESCI wakeup method
- Controls idle character detection
- Enables parity function
- Controls parity type

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	LOOPS	ENSCI	TXINV	M	WAKE	ILTY	PEN	PTY
Reset:	0	0	0	0	0	0	0	0

Figure 13-9. ESCI Control Register 1 (SCC1)

LOOPS — Loop Mode Select Bit
This read/write bit enables loop mode operation. In loop mode the RxD pin is disconnected from the ESCI, and the transmitter output goes into the receiver input. Both the transmitter and the receiver must be enabled to use loop mode.
   1 = Loop mode enabled
   0 = Normal operation enabled

ILIE — Idle Line Interrupt Enable Bit

This read/write bit enables the IDLE bit to generate ESCI receiver interrupt requests.

1 = IDLE enabled to generate interrupt requests
0 = IDLE not enabled to generate interrupt requests

TE — Transmitter Enable Bit

Setting this read/write bit begins the transmission by sending a preamble of 10 or 11 1s from the transmit shift register to the TxD pin. If software clears the TE bit, the transmitter completes any transmission in progress before the TxD returns to the idle condition (high). Clearing and then setting TE during a transmission queues an idle character to be sent after the character currently being transmitted.

1 = Transmitter enabled
0 = Transmitter disabled

NOTE
Writing to the TE bit is not allowed when the enable ESCI bit (ENSCI) is clear. ENSCI is in ESCI control register 1.

RE — Receiver Enable Bit

Setting this read/write bit enables the receiver. Clearing the RE bit disables the receiver but does not affect receiver interrupt flag bits.

1 = Receiver enabled
0 = Receiver disabled

NOTE
Writing to the RE bit is not allowed when the enable ESCI bit (ENSCI) is clear. ENSCI is in ESCI control register 1.

RWU — Receiver Wakeup Bit

This read/write bit puts the receiver in a standby state during which receiver interrupts are disabled. The WAKE bit in SCC1 determines whether an idle input or an address mark brings the receiver out of the standby state and clears the RWU bit.

1 = Standby state
0 = Normal operation

SBK — Send Break Bit

Setting and then clearing this read/write bit transmits a break character followed by a 1. The 1 after the break character guarantees recognition of a valid start bit. If SBK remains set, the transmitter continuously transmits break characters with no 1s between them.

1 = Transmit break characters
0 = No break characters being transmitted

NOTE
Do not toggle the SBK bit immediately after setting the SCTE bit. Toggling SBK before the preamble begins causes the ESCI to send a break character instead of a preamble.

function, and TIM1 channel 1 status and control register (T1SC1) is unused. While the MS0B bit is set, the channel 1 pin, T1CH1, is available as a general-purpose I/O pin.

Channels 2 and 3 can be linked to form a buffered PWM channel whose output appears on the T1CH2 pin. The TIM1 channel registers of the linked pair alternately control the output.

Setting the MS2B bit in TIM1 channel 2 status and control register (T1SC2) links channel 2 and channel 3. The TIM1 channel 2 registers initially control the pulse width on the T1CH2 pin. Writing to the TIM1 channel 3 registers enables the TIM1 channel 3 registers to synchronously control the pulse width at the beginning of the next PWM period. At each subsequent overflow, the TIM1 channel registers (2 or 3) that control the pulse width are the ones written to last. T1SC2 controls and monitors the buffered PWM function, and TIM1 channel 3 status and control register (T1SC3) is unused. While the MS2B bit is set, the channel 3 pin, T1CH3, is available as a general-purpose I/O pin.

> **NOTE**
> *In buffered PWM signal generation, do not write new pulse width values to the currently active channel registers. User software should track the currently active channel to prevent writing a new value to the active channel. Writing to the active channel registers is the same as generating unbuffered PWM signals.*

16.3.4.3 PWM Initialization

To ensure correct operation when generating unbuffered or buffered PWM signals, use the following initialization procedure:

1. In the TIM1 status and control register (T1SC):
   a. Stop the counter by setting the TIM1 stop bit, TSTOP.
   b. Reset the counter and prescaler by setting the TIM1 reset bit, TRST.
2. In the TIM1 counter modulo registers (T1MODH:T1MODL), write the value for the required PWM period.
3. In the TIM1 channel x registers (T1CHxH:T1CHxL), write the value for the required pulse width.
4. In TIM1 channel x status and control register (T1SCx):
   a. Write 0:1 (for unbuffered output compare or PWM signals) or 1:0 (for buffered output compare or PWM signals) to the mode select bits, MSxB:MSxA. See Table 16-2.
   b. Write 1 to the toggle-on-overflow bit, TOVx.
   c. Write 1:0 (polarity 1 — to clear output on compare) or 1:1 (polarity 0 — to set output on compare) to the edge/level select bits, ELSxB:ELSxA. The output action on compare must force the output to the complement of the pulse width level. See Table 16-2.

> **NOTE**
> *In PWM signal generation, do not program the PWM channel to toggle on output compare. Toggling on output compare prevents reliable 0% duty cycle generation and removes the ability of the channel to self-correct in the event of software error or noise. Toggling on output compare can also cause incorrect PWM signal generation when changing the PWM pulse width to a new, much larger value.*

5. In the TIM1 status control register (T1SC), clear the TIM1 stop bit, TSTOP.

Setting MS0B links channels 0 and 1 and configures them for buffered PWM operation. The TIM1 channel 0 registers (T1CH0H:T1CH0L) initially control the buffered PWM output. TIM1 status control register 0 (T1SC0) controls and monitors the PWM signal from the linked channels. MS0B takes priority over MS0A.

Setting MS2B links channels 2 and 3 and configures them for buffered PWM operation. The TIM1 channel 2 registers (T1CH2H:T1CH2L) initially control the buffered PWM output. TIM1 status control register 2 (T1SC2) controls and monitors the PWM signal from the linked channels. MS2B takes priority over MS2A.

Clearing the toggle-on-overflow bit, TOVx, inhibits output toggles on TIM1 overflows. Subsequent output compares try to force the output to a state it is already in and have no effect. The result is a 0% duty cycle output.

Setting the channel x maximum duty cycle bit (CHxMAX) and setting the TOVx bit generates a 100% duty cycle output. See 16.8.4 TIM1 Channel Status and Control Registers.

16.4 Interrupts

The following TIM1 sources can generate interrupt requests:

- TIM1 overflow flag (TOF) — The TOF bit is set when the counter reaches the modulo value programmed in the TIM1 counter modulo registers. The TIM1 overflow interrupt enable bit, TOIE, enables TIM1 overflow interrupt requests. TOF and TOIE are in the T1SC register.

- TIM1 channel flags (CH3F:CH0F) — The CHxF bit is set when an input capture or output compare occurs on channel x. Channel x TIM1 interrupt requests are controlled by the channel x interrupt enable bit, CHxIE. Channel x TIM1 interrupt requests are enabled when CHxIE =1. CHxF and CHxIE are in the T1SCx register.

16.5 Low-Power Modes

The WAIT and STOP instructions put the MCU in low power-consumption standby modes.

16.5.1 Wait Mode

The TIM1 remains active after the execution of a WAIT instruction. In wait mode the TIM1 registers are not accessible by the CPU. Any enabled interrupt request from the TIM1 can bring the MCU out of wait mode.

If TIM1 functions are not required during wait mode, reduce power consumption by stopping the TIM1 before executing the WAIT instruction.

16.5.2 Stop Mode

The TIM1 module is inactive after the execution of a STOP instruction. The STOP instruction does not affect register conditions. TIM1 operation resumes after an external interrupt. If stop mode is exited by reset, the TIM1 is reset.

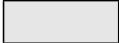|	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
Write:	0			TRST				
Reset:	0	0	1	0	0	0	0	0

☐ = Unimplemented

Figure 16-4. TIM1 Status and Control Register (T1SC)

TOF — TIM1 Overflow Flag Bit

This read/write flag is set when the counter reaches the modulo value programmed in the TIM1 counter modulo registers. Clear TOF by reading the T1SC register when TOF is set and then writing a 0 to TOF. If another TIM1 overflow occurs before the clearing sequence is complete, then writing 0 to TOF has no effect. Therefore, a TOF interrupt request cannot be lost due to inadvertent clearing of TOF. Writing a 1 to TOF has no effect.

    1 = Counter has reached modulo value
    0 = Counter has not reached modulo value

TOIE — TIM1 Overflow Interrupt Enable Bit

This read/write bit enables TIM1 overflow interrupts when the TOF bit becomes set.

    1 = TIM1 overflow interrupts enabled
    0 = TIM1 overflow interrupts disabled

TSTOP — TIM1 Stop Bit

This read/write bit stops the counter. Counting resumes when TSTOP is cleared. Reset sets the TSTOP bit, stopping the counter until software clears the TSTOP bit.

    1 = Counter stopped
    0 = Counter active

NOTE

Do not set the TSTOP bit before entering wait mode if the TIM1 is required to exit wait mode. Also, when the TSTOP bit is set and the timer is configured for input capture operation, input captures are inhibited until the TSTOP bit is cleared.

When using TSTOP to stop the timer counter, see if any timer flags are set. If a timer flag is set, it must be cleared by clearing TSTOP, then clearing the flag, then setting TSTOP again.

TRST — TIM1 Reset Bit

Setting this write-only bit resets the counter and the TIM1 prescaler. Setting TRST has no effect on any other timer registers. Counting resumes from $0000. TRST is cleared automatically after the counter is reset and always reads as 0.

    1 = Prescaler and counter cleared
    0 = No effect

NOTE

Setting the TSTOP and TRST bits simultaneously stops the counter at a value of $0000.

16.8.3 TIM1 Counter Modulo Registers

The read/write TIM1 modulo registers contain the modulo value for the counter. When the counter reaches the modulo value, the overflow flag (TOF) becomes set, and the counter resumes counting from $0000 at the next timer clock. Writing to the high byte (T1MODH) inhibits the TOF bit and overflow interrupts until the low byte (T1MODL) is written. Reset sets the TIM1 counter modulo registers.
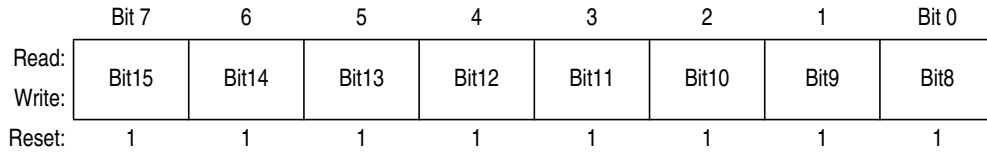
	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
Reset:	1	1	1	1	1	1	1	1

Figure 16-7. TIM1 Counter Modulo High Register (T1MODH)

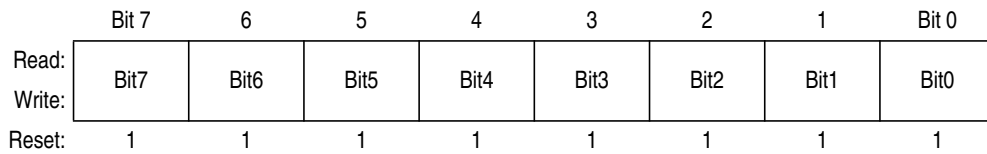|	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Reset:	1	1	1	1	1	1	1	1

Figure 16-8. TIM1 Counter Modulo Low Register (T1MODL)

NOTE
Reset the counter before writing to the TIM1 counter modulo registers.

16.8.4 TIM1 Channel Status and Control Registers

Each of the TIM1 channel status and control registers does the following:

- Flags input captures and output compares
- Enables input capture and output compare interrupts
- Selects input capture, output compare, or PWM operation
- Selects high, low, or toggling output on output compare
- Selects rising edge, falling edge, or any edge as the active input capture trigger
- Selects output toggling on TIM1 overflow
- Selects 0% and 100% PWM duty cycle
- Selects buffered or unbuffered output compare/PWM operation

Setting MS0B links channels 0 and 1 and configures them for buffered PWM operation. The TIM2 channel 0 registers (T2CH0H:T2CH0L) initially control the buffered PWM output. TIM2 status control register 0 (T2SCR0) controls and monitors the PWM signal from the linked channels. MS0B takes priority over MS0A.

Clearing the toggle-on-overflow bit, TOVx, inhibits output toggles on TIM2 overflows. Subsequent output compares try to force the output to a state it is already in and have no effect. The result is a 0% duty cycle output.

Setting the channel x maximum duty cycle bit (CHxMAX) and setting the TOVx bit generates a 100% duty cycle output. See 17.8.1 TIM2 Status and Control Register.

17.4 Interrupts

The following TIM2 sources can generate interrupt requests:
- TIM2 overflow flag (TOF) — The TOF bit is set when the counter reaches the modulo value programmed in the TIM2 counter modulo registers. The TIM2 overflow interrupt enable bit, TOIE, enables TIM2 overflow interrupt requests. TOF and TOIE are in the T2SC register.
- TIM2 channel flags (CH1F:CH0F) — The CHxF bit is set when an input capture or output compare occurs on channel x. Channel x TIM2 interrupt requests are controlled by the channel x interrupt enable bit, CHxIE. Channel x TIM2 interrupt requests are enabled when CHxIE =1. CHxF and CHxIE are in the T2SCx register.

17.5 Low-Power Modes

The WAIT and STOP instructions put the MCU in low power-consumption standby modes.

17.5.1 Wait Mode

The TIM2 remains active after the execution of a WAIT instruction. In wait mode the TIM2 registers are not accessible by the CPU. Any enabled interrupt request from the TIM2 can bring the MCU out of wait mode.

If TIM2 functions are not required during wait mode, reduce power consumption by stopping the TIM2 before executing the WAIT instruction.

17.5.2 Stop Mode

The TIM2 module is inactive after the execution of a STOP instruction. The STOP instruction does not affect register conditions. TIM2 operation resumes after an external interrupt. If stop mode is exited by reset, the TIM2 is reset.

17.6 TIM2 During Break Interrupts

A break interrupt stops the counter and inhibits input captures.

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the break flag control register (BFCR) enables software to clear status bits during the break state. See BFCR in the SIM section of this data sheet.
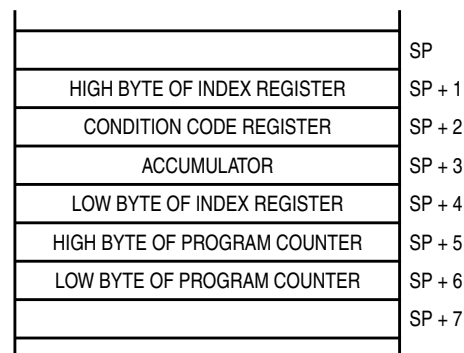
Table 18-7. READSP (Read Stack Pointer) Command

Description	Reads stack pointer
Operand	None
Data Returned	Returns incremented stack pointer value (SP + 1) in high-byte:low-byte order
Opcode	$0C
Command Sequence	

Table 18-8. RUN (Run User Program) Command

Description	Executes PULH and RTI instructions
Operand	None
Data Returned	None
Opcode	$28
Command Sequence	

The MCU executes the SWI and PSHH instructions when it enters monitor mode. The RUN command tells the MCU to execute the PULH and RTI instructions. Before sending the RUN command, the host can modify the stacked CPU registers to prepare to run the host program. The READSP command returns the incremented stack pointer value, SP + 1. The high and low bytes of the program counter are at addresses SP + 5 and SP + 6.

	SP
HIGH BYTE OF INDEX REGISTER	SP + 1
CONDITION CODE REGISTER	SP + 2
ACCUMULATOR	SP + 3
LOW BYTE OF INDEX REGISTER	SP + 4
HIGH BYTE OF PROGRAM COUNTER	SP + 5
LOW BYTE OF PROGRAM COUNTER	SP + 6
	SP + 7

Figure 18-17. Stack Pointer at Monitor Mode Entry

MC68HC908QC16 • MC68HC908QC8 • MC68HC908QC4 Data Sheet, Rev. 5

Characteristic	Conditions	Symbol	Min	Typ[1]	Max	Unit	Comment
Differential non-linearity	10-bit mode	DNL	0	±0.5	—	LSB	
	8-bit mode		0	±0.3	—		
	Monotonicity and no-missing-codes guaranteed						
Integral non-linearity	10-bit mode	INL	0	±0.5	—	LSB	
	8-bit mode		0	±0.3	—		
Zero-scale error	10-bit mode	E_{ZS}	0	±0.5	—	LSB	$V_{ADIN} = V_{SS}$
	8-bit mode		0	±0.3	—		
Full-scale error	10-bit mode	E_{FS}	0	±0.5	—	LSB	$V_{ADIN} = V_{DD}$
	8-bit mode		0	±0.3	—		
Quantization error	10-bit mode	E_Q	—	—	±0.5	LSB	8-bit mode is not truncated
	8-bit mode		—	—	±0.5		
Input leakage error	10-bit mode	E_{IL}	0	±0.2	±5	LSB	Pad leakage[5] * R_{AS}
	8-bit mode		0	±0.1	±1.2		
Bandgap voltage input[3][6]		V_{BG}	1.17	1.245	1.32	V	

1. Typical values assume V_{DD} = 5.0 V, temperature = 25 C, f_{ADCK} = 1.0 MHz unless otherwise stated. Typical values are for reference only and are not tested in production.
2. Incremental I_{DD} added to MCU mode current.
3. Values are based on characterization results, not tested in production.
4. Reference the ADC module specification for more information on calculating conversion times.
5. Based on typical input pad leakage current.
6. LVI must be enabled, (LVIPWRD = 0, in CONFIG1). Voltage input to ADCH4:0 = $1A, an ADC conversion on this channel allows user to determine supply voltage.

NOTES:

1. DIMENSIONS AND TOLERANCING PER ASME Y14.5M—1994.

2. DIMENSIONS IN MILLIMETERS.

△3. DIMENSION DOES NOT INCLUDE MOLD PROTRUSION. ALLOWABLE MOLD PROTRUSION IS 0.15 PER END.

△4. DIMENSION DOES NOT INCLUDE INTERLEAD FLASH OR MOLD PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 PER SIDE.

△5. DIMENSION DOES NOT INCLUDE DAM BAR PROTRUSIONS. DAM BAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED 0.38.

	MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE	
TITLE: 28 LEAD 0.65 PITCH TSSOP 4.4 WIDE BODY, 1.2 MAX HEIGHT	DOCUMENT NO: 98ARS23923W		REV: C
	CASE NUMBER: 1168-02		18 AUG 2006
	STANDARD: JEDEC MO-153 AE		