

Welcome to [E-XFL.COM](http://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	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I <sup>2</sup> C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	22
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 6x10b
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	<a href="https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s08ac8mfje">https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s08ac8mfje</a>

# MC9S08AC16 Series Features

## MC9S08AC16 Series Devices

- Consumer & Industrial
  - MC9S08AC16
  - MC9S08AC8
- Automotive
  - MC9S08AW16A
  - MC9S08AW8A

## 8-Bit HCS08 Central Processor Unit (CPU)

- 40-MHz HCS08 CPU (central processor unit)
- 20-MHz internal bus frequency
- HC08 instruction set with added BGND instruction
- Background debugging system
- Breakpoint capability to allow single breakpoint setting during in-circuit debugging (plus two more breakpoints in on-chip debug module)
- Debug module containing two comparators and nine trigger modes. Eight deep FIFO for storing change-of-flow addresses and event-only data. Debug module supports both tag and force breakpoints.
- Support for up to 32 interrupt/reset sources

## Memory Options

- Up to 16 KB of on-chip in-circuit programmable FLASH memory with block protection and security options
- Up to 1 KB of on-chip RAM

## Clock Source Options

- Clock source options include crystal, resonator, external clock, or internally generated clock with precision NVM trimming

## System Protection

- Optional computer operating properly (COP) reset with option to run from independent internal clock source or bus clock
- Low-voltage detection with reset or interrupt
- Illegal opcode detection with reset
- Illegal address detection with reset

## Power-Saving Modes

- Wait plus two stops

## Peripherals

- **ADC** — 8-channel, 10-bit analog-to-digital converter with automatic compare function
- **SCI** — Two serial communications interface modules with optional 13-bit break
- **SPI** — Serial peripheral interface module
- **IIC** — Inter-integrated circuit bus module to operate at up to 100 kbps with maximum bus loading; capable of higher baud rates with reduced loading
- **Timers** — Three 16-bit timer/pulse-width modulator (TPM) modules — Two 2-channel and one 4-channel; each has selectable input capture, output compare, and edge-aligned PWM capability on each channel. Each timer module may be configured for buffered, centered PWM (CPWM) on all channels
- **KBI** — 7-pin keyboard interrupt module

## Input/Output

- Up to 38 general-purpose input/output (I/O) pins
- Software selectable pullups on ports when used as inputs
- Software selectable slew rate control on ports when used as outputs
- Software selectable drive strength on ports when used as outputs
- Master reset pin and power-on reset (POR)
- Internal pullup on RESET, IRQ, and BKGD/MS pins to reduce customer system cost

## Package Options

- 48-pin quad flat no-lead package (QFN)
- 44-pin low-profile quad flat package (LQFP)
- 42-pin shrink dual-in-line package (SDIP)
- 32-pin low-profile quad flat package (LQFP)

external reset circuitry unnecessary. This pin is normally connected to the standard 6-pin background debug connector so a development system can directly reset the MCU system. If desired, a manual external reset can be added by supplying a simple switch to ground (pull reset pin low to force a reset).

Whenever any reset is initiated (whether from an external signal or from an internal system), the reset pin is driven low for approximately 34 bus cycles. The reset circuitry decodes the cause of reset and records it by setting a corresponding bit in the system control reset status register (SRS).

In EMC-sensitive applications, an external RC filter is recommended on the reset pin. See Figure 2-5 for an example.

### 2.3.4 Background/Mode Select (BKGD/MS)

While in reset, the BKGD/MS pin functions as a mode select pin. Immediately after reset rises the pin functions as the background pin and can be used for background debug communication. While functioning as a background/mode select pin, the pin includes an internal pullup device, input hysteresis, a standard output driver, and no output slew rate control.

If nothing is connected to this pin, the MCU will enter normal operating mode at the rising edge of reset. If a debug system is connected to the 6-pin standard background debug header, it can hold BKGD/MS low during the rising edge of reset which forces the MCU to active background mode.

The BKGD pin is used primarily for background debug controller (BDC) communications using a custom protocol that uses 16 clock cycles of the target MCU's BDC clock per bit time. The target MCU's BDC clock could be as fast as the bus clock rate, so there should never be any significant capacitance connected to the BKGD/MS pin that could interfere with background serial communications.

Although the BKGD pin is a pseudo open-drain pin, the background debug communication protocol provides brief, actively driven, high speedup pulses to ensure fast rise times. Small capacitances from cables and the absolute value of the internal pullup device play almost no role in determining rise and fall times on the BKGD pin.

### 2.3.5 ADC Reference Pins ( $V_{REFH}$ , $V_{REFL}$ )

The  $V_{REFH}$  and  $V_{REFL}$  pins are the voltage reference high and voltage reference low inputs respectively for the ADC module.

### 2.3.6 External Interrupt Pin (IRQ)

The IRQ pin is the input source for the IRQ interrupt and is also the input for the BIH and BIL instructions. If the IRQ function is not enabled, this pin does not perform any function.

In EMC-sensitive applications, an external RC filter is recommended on the IRQ pin. See Figure 2-5 for an example.

**Table 4-13. FCMD Register Field Descriptions**

Field	Description
7:0 FCMD[7:0]	<b>FLASH Command Bits</b> — See Table 4-14

**Table 4-14. FLASH Commands**

Command	FCMD	Equate File Label
Blank check	0x05	mBlank
Byte program	0x20	mByteProg
Byte program — burst mode	0x25	mBurstProg
Page erase (512 bytes/page)	0x40	mPageErase
Mass erase (all FLASH)	0x41	mMassErase

All other command codes are illegal and generate an access error.

It is not necessary to perform a blank check command after a mass erase operation. Only blank check is required as part of the security unlocking mechanism.

stop mode and system clocks are shut down, a separate asynchronous path is used so the IRQ (if enabled) can wake the MCU.

### 5.5.2.1 IRQ Pin Configuration Options

The IRQ pin enable (IRQPE) control bit in IRQSC must be 1 in order for the IRQ pin to act as the interrupt request (IRQ) input. As an IRQ input, the user can choose the polarity of edges or levels detected (IRQEDG), whether the pin detects edges-only or edges and levels (IRQMOD), and whether an event causes an interrupt or only sets the IRQF flag which can be polled by software.

The IRQ pin, when enabled, defaults to use an internal pull device (IRQPDD = 0), the device is a pull-up or pull-down depending on the polarity chosen. If the user desires to use an external pull-up or pull-down, the IRQPDD can be written to a 1 to turn off the internal device.

BIH and BIL instructions may be used to detect the level on the IRQ pin when the pin is configured to act as the IRQ input.

#### NOTE

This pin does not contain a clamp diode to  $V_{DD}$  and should not be driven above  $V_{DD}$ . The voltage measured on the internally pulled up IRQ pin may be as low as  $V_{DD} - 0.7$  V. The internal gates connected to this pin are pulled all the way to  $V_{DD}$ .

#### NOTE

When enabling the IRQ pin for use, the IRQF will be set, and should be cleared prior to enabling the interrupt. When configuring the pin for falling edge and level sensitivity in a 5V system, it is necessary to wait at least 6 cycles between clearing the flag and enabling the interrupt.

### 5.5.2.2 Edge and Level Sensitivity

The IRQMOD control bit reconfigures the detection logic so it detects edge events and pin levels. In the edge and level detection mode, the IRQF status flag becomes set when an edge is detected (when the IRQ pin changes from the deasserted to the asserted level), but the flag is continuously set (and cannot be cleared) as long as the IRQ pin remains at the asserted level.

### 5.5.3 Interrupt Vectors, Sources, and Local Masks

Table 5-2 provides a summary of all interrupt sources. Higher-priority sources are located toward the bottom of the table. The high-order byte of the address for the interrupt service routine is located at the first address in the vector address column, and the low-order byte of the address for the interrupt service routine is located at the next higher address.

When an interrupt condition occurs, an associated flag bit becomes set. If the associated local interrupt enable is 1, an interrupt request is sent to the CPU. Within the CPU, if the global interrupt mask (I bit in the CCR) is 0, the CPU will finish the current instruction, stack the PCL, PCH, X, A, and CCR CPU registers, set the I bit, and then fetch the interrupt vector for the highest priority pending interrupt. Processing then continues in the interrupt service routine.

## 5.9.2 System Reset Status Register (SRS)

This register includes seven read-only status flags to indicate the source of the most recent reset. When a debug host forces reset by writing 1 to BDFR in the SBDPFR register, none of the status bits in SRS will be set. Writing any value to this register address clears the COP watchdog timer without affecting the contents of this register. The reset state of these bits depends on what caused the MCU to reset.

	7	6	5	4	3	2	1	0
R	POR	PIN	COP	ILOP	ILAD	ICG	LVD	0
W	Writing any value to SRS address clears COP watchdog timer.							
POR	1	0	0	0	0	0	1	0
LVR:	U	0	0	0	0	0	1	0
Any other reset:	0	Note <sup>1</sup>	Note <sup>1</sup>	Note <sup>1</sup>	0	Note <sup>1</sup>	0	0

U = Unaffected by reset

<sup>1</sup> Any of these reset sources that are active at the time of reset will cause the corresponding bit(s) to be set; bits corresponding to sources that are not active at the time of reset will be cleared.

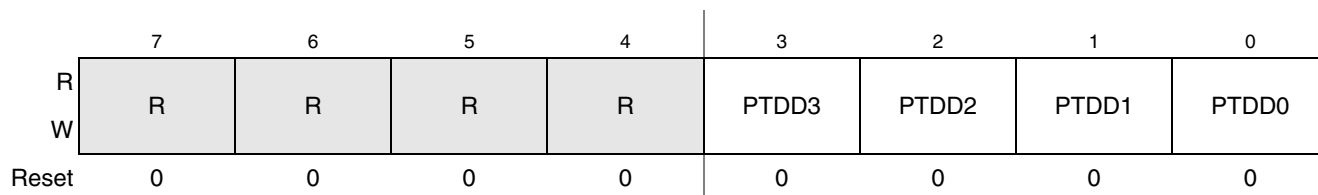
**Figure 5-3. System Reset Status (SRS)**

**Table 5-4. SRS Register Field Descriptions**

Field	Description
7 POR	<b>Power-On Reset</b> — Reset was caused by the power-on detection logic. Because the internal supply voltage was ramping up at the time, the low-voltage reset (LVR) status bit is also set to indicate that the reset occurred while the internal supply was below the LVR threshold. 0 Reset not caused by POR. 1 POR caused reset.
6 PIN	<b>External Reset Pin</b> — Reset was caused by an active-low level on the external reset pin. 0 Reset not caused by external reset pin. 1 Reset came from external reset pin.
5 COP	<b>Computer Operating Properly (COP) Watchdog</b> — Reset was caused by the COP watchdog timer timing out. This reset source may be blocked by COPE = 0. 0 Reset not caused by COP timeout. 1 Reset caused by COP timeout.
4 ILOP	<b>Illegal Opcode</b> — Reset was caused by an attempt to execute an unimplemented or illegal opcode. The STOP instruction is considered illegal if stop is disabled by STOPE = 0 in the SOPT register. The BGND instruction is considered illegal if active background mode is disabled by ENBDM = 0 in the BDCSC register. 0 Reset not caused by an illegal opcode. 1 Reset caused by an illegal opcode.

## 6.7.7 Port D I/O Registers (PTDD and PTDDD)

Port D parallel I/O function is controlled by the registers listed below.

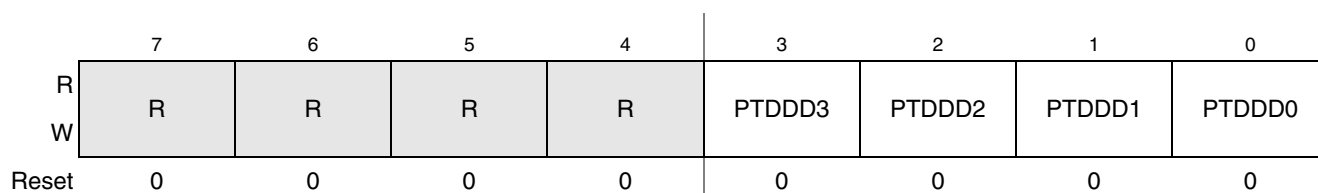


**Figure 6-25. Port D Data Register (PTDD)<sup>1</sup>**

<sup>1</sup> Bits 7 through 4 are reserved bits that must always be written to 0.

**Table 6-16. PTDD Register Field Descriptions**

Field	Description
3:0 PTDD[3:0]	<b>Port D Data Register Bits</b> — For port D pins that are inputs, reads return the logic level on the pin. For port D pins that are configured as outputs, reads return the last value written to this register. Writes are latched into all bits of this register. For port D pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTDD to all 0s, but these 0s are not driven out the corresponding pins because reset also configures all port pins as high-impedance inputs with pullups disabled.



**Figure 6-26. Data Direction for Port D (PTDDD)<sup>1</sup>**

<sup>1</sup> Bits 7 through 4 are reserved bits that must always be written to 0.

**Table 6-17. PTDDD Register Field Descriptions**

Field	Description
3:0 PTDDD[3:0]	<b>Data Direction for Port D Bits</b> — These read/write bits control the direction of port D pins and what is read for PTDD reads. 0 Input (output driver disabled) and reads return the pin value. 1 Output driver enabled for port D bit n and PTDD reads return the contents of PTDDn.

**Table 8-12. MFD and RFD Decode Table**

101	14	101	÷32
110	16	110	÷64
111	18	111	÷128

### 8.5.2 Example #1: External Crystal = 32 kHz, Bus Frequency = 4.19 MHz

In this example, the FLL will be used (in FEE mode) to multiply the external 32 kHz oscillator up to 8.38 MHz to achieve 4.19 MHz bus frequency.

After the MCU is released from reset, the ICG is in self-clocked mode (SCM) and supplies approximately 8 MHz on ICGOUT, which corresponds to a 4 MHz bus frequency ( $f_{Bus}$ ).

The clock scheme will be FLL engaged, external (FEE). So

$$f_{ICGOUT} = f_{ext} * P * N / R ; P = 64, f_{ext} = 32 \text{ kHz} \quad \text{Eqn. 8-1}$$

Solving for N / R gives:

$$N / R = 8.38 \text{ MHz} / (32 \text{ kHz} * 64) = 4 ; \text{ we can choose } N = 4 \text{ and } R = 1 \quad \text{Eqn. 8-2}$$

The values needed in each register to set up the desired operation are:

#### ICGC1 = \$38 (%00111000)

Bit 7	HGO	0	Configures oscillator for low power
Bit 6	RANGE	0	Configures oscillator for low-frequency range; FLL prescale factor is 64
Bit 5	REFS	1	Oscillator using crystal or resonator is requested
Bits 4:3	CLKS	11	FLL engaged, external reference clock mode
Bit 2	OSCSTEN	0	Oscillator disabled
Bit 1	LOCD	0	Loss-of-clock detection enabled
Bit 0		0	Unimplemented or reserved, always reads zero

#### ICGC2 = \$00 (%00000000)

Bit 7	LOLRE	0	Generates an interrupt request on loss of lock
Bits 6:4	MFD	000	Sets the MFD multiplication factor to 4
Bit 3	LOCRES	0	Generates an interrupt request on loss of clock
Bits 2:0	RFD	000	Sets the RFD division factor to ÷1

#### ICGS1 = \$xx

This is read only except for clearing interrupt flag

#### ICGS2 = \$xx

This is read only; should read DCOS = 1 before performing any time critical tasks

#### ICGFLTLU/L = \$xx

Only needed in self-clocked mode; FLT will be adjusted by loop to give 8.38 MHz DCO clock  
 Bits 15:12 unused 0000



## 10.3 TPMV3 Differences from Previous Versions

The TPMV3 is the latest version of the Timer/PWM module that addresses errata found in previous versions. The following section outlines the differences between TPMV3 and TPMV2 modules, and any considerations that should be taken when porting code.

**Table 10-1. TPMV2 and TPMV3 Porting Considerations**

Action	TPMV3	TPMV2
<b>Write to TPMxCnTH:L registers<sup>1</sup></b>		
Any write to TPMxCNTH or TPMxCNTL registers	Clears the TPM counter (TPMxCNTH:L) and the prescaler counter.	Clears the TPM counter (TPMxCNTH:L) only.
<b>Read of TPMxCNTH:L registers<sup>1</sup></b>		
In BDM mode, any read of TPMxCNTH:L registers	Returns the value of the TPM counter that is frozen.	If only one byte of the TPMxCNTH:L registers was read before the BDM mode became active, returns the latched value of TPMxCNTH:L from the read buffer (instead of the frozen TPM counter value).
In BDM mode, a write to TPMxSC, TPMxCNTH or TPMxCNTL	Clears this read coherency mechanism.	Does not clear this read coherency mechanism.
<b>Read of TPMxCnVH:L registers<sup>2</sup></b>		
In BDM mode, any read of TPMxCnVH:L registers	Returns the value of the TPMxCnVH:L register.	If only one byte of the TPMxCnVH:L registers was read before the BDM mode became active, returns the latched value of TPMxCNTH:L from the read buffer (instead of the value in the TPMxCnVH:L registers).
In BDM mode, a write to TPMxCnSC	Clears this read coherency mechanism.	Does not clear this read coherency mechanism.
<b>Write to TPMxCnVH:L registers</b>		
In Input Capture mode, writes to TPMxCnVH:L registers <sup>3</sup>	Not allowed.	Allowed.
In Output Compare mode, when (CLKSB:CLKSA not = 0:0), writes to TPMxCnVH:L registers <sup>3</sup>	Update the TPMxCnVH:L registers with the value of their write buffer at the next change of the TPM counter (end of the prescaler counting) after the second byte is written.	Always update these registers when their second byte is written.

<sup>8</sup> For more information, refer to Section 10.6.2.4, “Center-Aligned PWM Mode.” [SE110-TPM case 4]

### 10.3.1 Migrating from TPMV1

In addition to Section 10.3, “TPMV3 Differences from Previous Versions,” keep in mind the following considerations when migrating from a device that uses TPMV1.

- You can write to the Channel Value register (TPMxCnV) when the timer is not in input capture mode for TPMV2, not TPMV3.
- In edge- or center- aligned modes, the Channel Value register (TPMxCnV) registers only update when the timer changes from TPMMOD-1 to TPMMOD, or in the case of a free running timer from 0xFFFE to 0xFFFF.
- Also, when configuring the TPM modules, it is best to write to TPMxSC before TPMxCnV as a write to TPMxSC resets the coherency mechanism on the TPMxCnV registers.

**Table 10-2. Migrating to TPMV3 Considerations**

When...	Action / Best Practice
Writing to the Channel Value Register (TPMxCnV) register...	Timer must be in Input Capture mode.
Updating the Channel Value Register (TPMxCnV) register in edge-aligned or center-aligned modes...	Only occurs when the timer changes from TPMMOD-1 to TPMMOD (or in the case of a free running timer, from 0xFFFE to 0xFFFF).
Resetting the coherency mechanism for the Channel Value Register (TPMxCnV) register...	Write to TPMxSC.
Configuring the TPM modules...	Write first to TPMxSC and then to TPMxCnV register.

to avoid such settings.) The TPM channel could still be used in output compare mode for software timing functions (pin controls set not to affect the TPM channel pin).

### 10.6.1.2 Counter Overflow and Modulo Reset

An interrupt flag and enable are associated with the 16-bit main counter. The flag (TOF) is a software-accessible indication that the timer counter has overflowed. The enable signal selects between software polling (TOIE=0) where no hardware interrupt is generated, or interrupt-driven operation (TOIE=1) where a static hardware interrupt is generated whenever the TOF flag is equal to one.

The conditions causing TOF to become set depend on whether the TPM is configured for center-aligned PWM (CPWMS=1). In the simplest mode, there is no modulus limit and the TPM is not in CPWMS=1 mode. In this case, the 16-bit timer counter counts from 0x0000 through 0xFFFF and overflows to 0x0000 on the next counting clock. TOF becomes set at the transition from 0xFFFF to 0x0000. When a modulus limit is set, TOF becomes set at the transition from the value set in the modulus register to 0x0000. When the TPM is in center-aligned PWM mode (CPWMS=1), the TOF flag gets set as the counter changes direction at the end of the count value set in the modulus register (that is, at the transition from the value set in the modulus register to the next lower count value). This corresponds to the end of a PWM period (the 0x0000 count value corresponds to the center of a period).

### 10.6.1.3 Counting Modes

The main timer counter has two counting modes. When center-aligned PWM is selected (CPWMS=1), the counter operates in up/down counting mode. Otherwise, the counter operates as a simple up counter. As an up counter, the timer counter counts from 0x0000 through its terminal count and then continues with 0x0000. The terminal count is 0xFFFF or a modulus value in TPMxMODH:TPMxMODL.

When center-aligned PWM operation is specified, the counter counts up from 0x0000 through its terminal count and then down to 0x0000 where it changes back to up counting. Both 0x0000 and the terminal count value are normal length counts (one timer clock period long). In this mode, the timer overflow flag (TOF) becomes set at the end of the terminal-count period (as the count changes to the next lower count value).

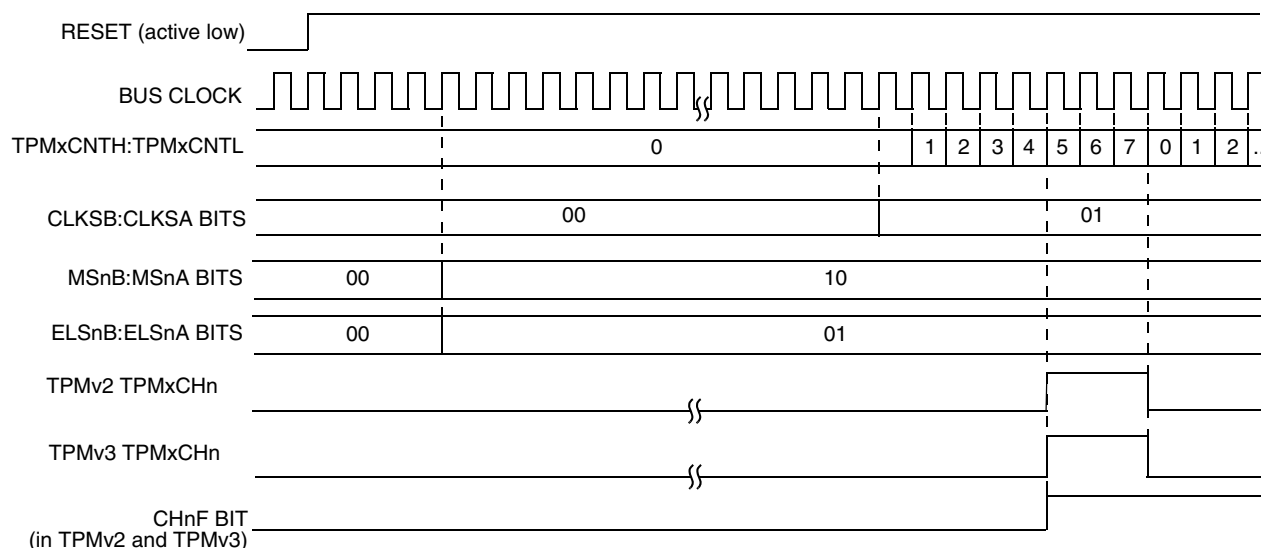
### 10.6.1.4 Manual Counter Reset

The main timer counter can be manually reset at any time by writing any value to either half of TPMxCNTH or TPMxCNTL. Resetting the counter in this manner also resets the coherency mechanism in case only half of the counter was read before resetting the count.

## 10.6.2 Channel Mode Selection

Provided CPWMS=0, the MSnB and MSnA control bits in the channel n status and control registers determine the basic mode of operation for the corresponding channel. Choices include input capture, output compare, and edge-aligned PWM.

EPWM mode  
 TPMxMODH:TPMxMODL = 0x0007  
 TPMxCnVH:TPMxCnVL = 0x0005



**Figure 10-18. Generation of low-true EPWM signal by TPM v2 and v3 after the reset**

The following procedure can be used in TPM v3 (when the channel pin is also a port pin) to emulate the high-true EPWM generated by TPM v2 after the reset.

- ...
- configure the channel pin as output port pin and set the output pin;
- configure the channel to generate the EPWM signal but keep ELSnB:ELSnA as 00;
- configure the other registers (TPMxMODH, TPMxMODL, TPMxCnVH, TPMxCnVL, ...);
- configure CLKSb:CLKSA bits (TPM v3 starts to generate the high-true EPWM signal, however TPM does not control the channel pin, so the EPWM signal is not available);
- wait until the TOF is set (or use the TOF interrupt);
- enable the channel output by configuring ELSnB:ELSnA bits (now EPWM signal is available);
- ...

### 11.1.1 Features

Features of SCI module include:

- Full-duplex, standard non-return-to-zero (NRZ) format
- Double-buffered transmitter and receiver with separate enables
- Programmable baud rates (13-bit modulo divider)
- Interrupt-driven or polled operation:
  - Transmit data register empty and transmission complete
  - Receive data register full
  - Receive overrun, parity error, framing error, and noise error
  - Idle receiver detect
  - Active edge on receive pin
  - Break detect supporting LIN
- Hardware parity generation and checking
- Programmable 8-bit or 9-bit character length
- Receiver wakeup by idle-line or address-mark
- Optional 13-bit break character generation / 11-bit break character detection
- Selectable transmitter output polarity

### 11.1.2 Modes of Operation

See Section 11.3, “Functional Description,” For details concerning SCI operation in these modes:

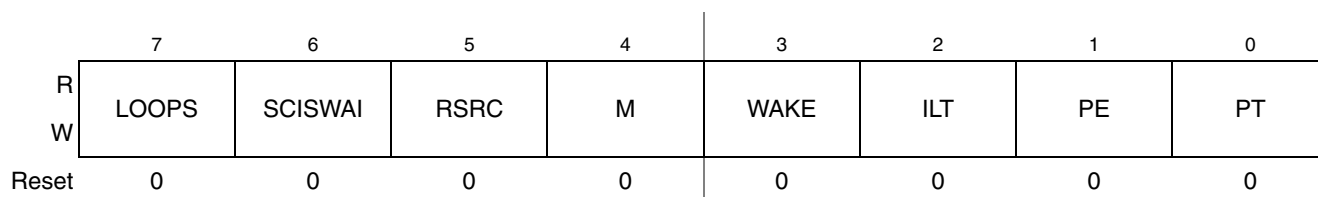
- 8- and 9-bit data modes
- Stop mode operation
- Loop mode
- Single-wire mode

**Table 11-2. SCixBDL Field Descriptions**

Field	Description
7:0 SBR[7:0]	<b>Baud Rate Modulo Divisor</b> — These 13 bits in SBR[12:0] are referred to collectively as BR, and they set the modulo divide rate for the SCI baud rate generator. When BR = 0, the SCI baud rate generator is disabled to reduce supply current. When BR = 1 to 8191, the SCI baud rate = $BUSCLK/(16 \times BR)$ . See also BR bits in Table 11-1.

## 11.2.2 SCI Control Register 1 (SCixC1)

This read/write register is used to control various optional features of the SCI system.


**Figure 11-6. SCI Control Register 1 (SCixC1)**
**Table 11-3. SCixC1 Field Descriptions**

Field	Description
7 LOOPS	<b>Loop Mode Select</b> — Selects between loop back modes and normal 2-pin full-duplex modes. When LOOPS = 1, the transmitter output is internally connected to the receiver input. 0 Normal operation — RxD and TxD use separate pins. 1 Loop mode or single-wire mode where transmitter outputs are internally connected to receiver input. (See RSRC bit.) RxD pin is not used by SCI.
6 SCISWAI	<b>SCI Stops in Wait Mode</b> 0 SCI clocks continue to run in wait mode so the SCI can be the source of an interrupt that wakes up the CPU. 1 SCI clocks freeze while CPU is in wait mode.
5 RSRC	<b>Receiver Source Select</b> — This bit has no meaning or effect unless the LOOPS bit is set to 1. When LOOPS = 1, the receiver input is internally connected to the TxD pin and RSRC determines whether this connection is also connected to the transmitter output. 0 Provided LOOPS = 1, RSRC = 0 selects internal loop back mode and the SCI does not use the RxD pins. 1 Single-wire SCI mode where the TxD pin is connected to the transmitter output and receiver input.
4 M	<b>9-Bit or 8-Bit Mode Select</b> 0 Normal — start + 8 data bits (LSB first) + stop. 1 Receiver and transmitter use 9-bit data characters start + 8 data bits (LSB first) + 9th data bit + stop.
3 WAKE	<b>Receiver Wakeup Method Select</b> — Refer to Section 11.3.3.2, “Receiver Wakeup Operation” for more information. 0 Idle-line wakeup. 1 Address-mark wakeup.
2 ILT	<b>Idle Line Type Select</b> — Setting this bit to 1 ensures that the stop bit and logic 1 bits at the end of a character do not count toward the 10 or 11 bit times of logic high level needed by the idle line detection logic. Refer to Section 11.3.3.2.1, “Idle-Line Wakeup” for more information. 0 Idle character bit count starts after start bit. 1 Idle character bit count starts after stop bit.

**Table 11-5. SC1xS1 Field Descriptions**

Field	Description
7 TDRE	<p><b>Transmit Data Register Empty Flag</b> — TDRE is set out of reset and when a transmit data value transfers from the transmit data buffer to the transmit shifter, leaving room for a new character in the buffer. To clear TDRE, read SC1xS1 with TDRE = 1 and then write to the SCI data register (SC1xD).</p> <p>0 Transmit data register (buffer) full. 1 Transmit data register (buffer) empty.</p>
6 TC	<p><b>Transmission Complete Flag</b> — TC is set out of reset and when TDRE = 1 and no data, preamble, or break character is being transmitted.</p> <p>0 Transmitter active (sending data, a preamble, or a break). 1 Transmitter idle (transmission activity complete).</p> <p>TC is cleared automatically by reading SC1xS1 with TC = 1 and then doing one of the following three things:</p> <ul style="list-style-type: none"> <li>• Write to the SCI data register (SC1xD) to transmit new data</li> <li>• Queue a preamble by changing TE from 0 to 1</li> <li>• Queue a break character by writing 1 to SBK in SC1xC2</li> </ul>
5 RDRF	<p><b>Receive Data Register Full Flag</b> — RDRF becomes set when a character transfers from the receive shifter into the receive data register (SC1xD). To clear RDRF, read SC1xS1 with RDRF = 1 and then read the SCI data register (SC1xD).</p> <p>0 Receive data register empty. 1 Receive data register full.</p>
4 IDLE	<p><b>Idle Line Flag</b> — IDLE is set when the SCI receive line becomes idle for a full character time after a period of activity. When ILT = 0, the receiver starts counting idle bit times after the start bit. So if the receive character is all 1s, these bit times and the stop bit time count toward the full character time of logic high (10 or 11 bit times depending on the M control bit) needed for the receiver to detect an idle line. When ILT = 1, the receiver doesn't start counting idle bit times until after the stop bit. So the stop bit and any logic high bit times at the end of the previous character do not count toward the full character time of logic high needed for the receiver to detect an idle line.</p> <p>To clear IDLE, read SC1xS1 with IDLE = 1 and then read the SCI data register (SC1xD). After IDLE has been cleared, it cannot become set again until after a new character has been received and RDRF has been set. IDLE will get set only once even if the receive line remains idle for an extended period.</p> <p>0 No idle line detected. 1 Idle line was detected.</p>
3 OR	<p><b>Receiver Overrun Flag</b> — OR is set when a new serial character is ready to be transferred to the receive data register (buffer), but the previously received character has not been read from SC1xD yet. In this case, the new character (and all associated error information) is lost because there is no room to move it into SC1xD. To clear OR, read SC1xS1 with OR = 1 and then read the SCI data register (SC1xD).</p> <p>0 No overrun. 1 Receive overrun (new SCI data lost).</p>
2 NF	<p><b>Noise Flag</b> — The advanced sampling technique used in the receiver takes seven samples during the start bit and three samples in each data bit and the stop bit. If any of these samples disagrees with the rest of the samples within any bit time in the frame, the flag NF will be set at the same time as the flag RDRF gets set for the character. To clear NF, read SC1xS1 and then read the SCI data register (SC1xD).</p> <p>0 No noise detected. 1 Noise detected in the received character in SC1xD.</p>

Table 13-4. IIC Divider and Hold Values

ICR (hex)	SCL Divider	SDA Hold Value	SCL Hold (Start) Value	SDA Hold (Stop) Value
00	20	7	6	11
01	22	7	7	12
02	24	8	8	13
03	26	8	9	14
04	28	9	10	15
05	30	9	11	16
06	34	10	13	18
07	40	10	16	21
08	28	7	10	15
09	32	7	12	17
0A	36	9	14	19
0B	40	9	16	21
0C	44	11	18	23
0D	48	11	20	25
0E	56	13	24	29
0F	68	13	30	35
10	48	9	18	25
11	56	9	22	29
12	64	13	26	33
13	72	13	30	37
14	80	17	34	41
15	88	17	38	45
16	104	21	46	53
17	128	21	58	65
<b>18</b>	80	9	38	41
<b>19</b>	96	9	46	49
<b>1A</b>	112	17	54	57
<b>1B</b>	128	17	62	65
<b>1C</b>	144	25	70	73
<b>1D</b>	160	25	78	81
<b>1E</b>	192	33	94	97
<b>1F</b>	240	33	118	121

ICR (hex)	SCL Divider	SDA Hold Value	SCL Hold (Start) Value	SCL Hold (Stop) Value
20	160	17	78	81
21	192	17	94	97
22	224	33	110	113
23	256	33	126	129
24	288	49	142	145
25	320	49	158	161
26	384	65	190	193
27	480	65	238	241
28	320	33	158	161
29	384	33	190	193
2A	448	65	222	225
2B	512	65	254	257
2C	576	97	286	289
2D	640	97	318	321
2E	768	129	382	385
2F	960	129	478	481
30	640	65	318	321
31	768	65	382	385
32	896	129	446	449
33	1024	129	510	513
34	1152	193	574	577
35	1280	193	638	641
36	1536	257	766	769
37	1920	257	958	961
<b>38</b>	1280	129	638	641
<b>39</b>	1536	129	766	769
<b>3A</b>	1792	257	894	897
<b>3B</b>	2048	257	1022	1025
<b>3C</b>	2304	385	1150	1153
<b>3D</b>	2560	385	1278	1281
<b>3E</b>	3072	513	1534	1537
<b>3F</b>	3840	513	1918	1921

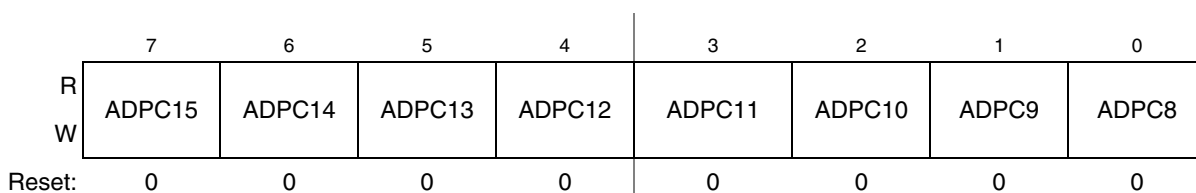


**Table 14-9. APCTL1 Register Field Descriptions (continued)**

Field	Description
1 ADPC1	<b>ADC Pin Control 1</b> — ADPC1 is used to control the pin associated with channel AD1. 0 AD1 pin I/O control enabled 1 AD1 pin I/O control disabled
0 ADPC0	<b>ADC Pin Control 0</b> — ADPC0 is used to control the pin associated with channel AD0. 0 AD0 pin I/O control enabled 1 AD0 pin I/O control disabled

### 14.4.9 Pin Control 2 Register (APCTL2)

APCTL2 is used to control channels 8–15 of the ADC module.



**Figure 14-12. Pin Control 2 Register (APCTL2)**

**Table 14-10. APCTL2 Register Field Descriptions**

Field	Description
7 ADPC15	<b>ADC Pin Control 15</b> — ADPC15 is used to control the pin associated with channel AD15. 0 AD15 pin I/O control enabled 1 AD15 pin I/O control disabled
6 ADPC14	<b>ADC Pin Control 14</b> — ADPC14 is used to control the pin associated with channel AD14. 0 AD14 pin I/O control enabled 1 AD14 pin I/O control disabled
5 ADPC13	<b>ADC Pin Control 13</b> — ADPC13 is used to control the pin associated with channel AD13. 0 AD13 pin I/O control enabled 1 AD13 pin I/O control disabled
4 ADPC12	<b>ADC Pin Control 12</b> — ADPC12 is used to control the pin associated with channel AD12. 0 AD12 pin I/O control enabled 1 AD12 pin I/O control disabled
3 ADPC11	<b>ADC Pin Control 11</b> — ADPC11 is used to control the pin associated with channel AD11. 0 AD11 pin I/O control enabled 1 AD11 pin I/O control disabled
2 ADPC10	<b>ADC Pin Control 10</b> — ADPC10 is used to control the pin associated with channel AD10. 0 AD10 pin I/O control enabled 1 AD10 pin I/O control disabled

- Non-intrusive commands can be executed at any time even while the user's program is running. Non-intrusive commands allow a user to read or write MCU memory locations or access status and control registers within the background debug controller.

Typically, a relatively simple interface pod is used to translate commands from a host computer into commands for the custom serial interface to the single-wire background debug system. Depending on the development tool vendor, this interface pod may use a standard RS-232 serial port, a parallel printer port, or some other type of communications such as a universal serial bus (USB) to communicate between the host PC and the pod. The pod typically connects to the target system with ground, the BKGD pin,  $\overline{\text{RESET}}$ , and sometimes  $V_{\text{DD}}$ . An open-drain connection to reset allows the host to force a target system reset, which is useful to regain control of a lost target system or to control startup of a target system before the on-chip nonvolatile memory has been programmed. Sometimes  $V_{\text{DD}}$  can be used to allow the pod to use power from the target system to avoid the need for a separate power supply. However, if the pod is powered separately, it can be connected to a running target system without forcing a target system reset or otherwise disturbing the running application program.

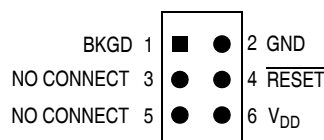


Figure 15-1. BDM Tool Connector

## 15.2.1 BKGD Pin Description

BKGD is the single-wire background debug interface pin. The primary function of this pin is for bidirectional serial communication of active background mode commands and data. During reset, this pin is used to select between starting in active background mode or starting the user's application program. This pin is also used to request a timed sync response pulse to allow a host development tool to determine the correct clock frequency for background debug serial communications.

BDC serial communications use a custom serial protocol first introduced on the M68HC12 Family of microcontrollers. This protocol assumes the host knows the communication clock rate that is determined by the target BDC clock rate. All communication is initiated and controlled by the host that drives a high-to-low edge to signal the beginning of each bit time. Commands and data are sent most significant bit first (MSB first). For a detailed description of the communications protocol, refer to Section 15.2.2, "Communication Details."

If a host is attempting to communicate with a target MCU that has an unknown BDC clock rate, a SYNC command may be sent to the target MCU to request a timed sync response signal from which the host can determine the correct communication speed.

BKGD is a pseudo-open-drain pin and there is an on-chip pullup so no external pullup resistor is required. Unlike typical open-drain pins, the external RC time constant on this pin, which is influenced by external capacitance, plays almost no role in signal rise time. The custom protocol provides for brief, actively driven speedup pulses to force rapid rise times on this pin without risking harmful drive level conflicts. Refer to Section 15.2.2, "Communication Details," for more detail.

## A.6 DC Characteristics

This section includes information about power supply requirements, I/O pin characteristics, and power supply current in various operating modes.

**Table A-6. DC Characteristics**

Num	C	Parameter	Symbol	Min	Typ <sup>1</sup>	Max	Unit
1	P	Output high voltage — Low Drive (PTxDSn = 0) 5 V, I <sub>Load</sub> = -2 mA 3 V, I <sub>Load</sub> = -0.6 mA 5 V, I <sub>Load</sub> = -0.4 mA 3 V, I <sub>Load</sub> = -0.24 mA	V <sub>OH</sub>	V <sub>DD</sub> - 1.5	—	—	V
		V <sub>DD</sub> - 1.5 V <sub>DD</sub> - 0.8 V <sub>DD</sub> - 0.8		— — —	— — —		
2	P	Output high voltage — High Drive (PTxDSn = 1) 5 V, I <sub>Load</sub> = -10 mA 3 V, I <sub>Load</sub> = -3 mA 5 V, I <sub>Load</sub> = -2 mA 3 V, I <sub>Load</sub> = -0.4 mA	V <sub>OH</sub>	V <sub>DD</sub> - 1.5	—	—	V
		V <sub>DD</sub> - 1.5 V <sub>DD</sub> - 0.8 V <sub>DD</sub> - 0.8		— — —	— — —		
2	P	Output low voltage — Low Drive (PTxDSn = 0) 5 V, I <sub>Load</sub> = 2 mA 3 V, I <sub>Load</sub> = 0.6 mA 5 V, I <sub>Load</sub> = 0.4 mA 3 V, I <sub>Load</sub> = 0.24 mA	V <sub>OL</sub>	—	—	1.5	V
		— — — —		— — — —	1.5 0.8 0.8		
2	P	Output low voltage — High Drive (PTxDSn = 1) 5 V, I <sub>Load</sub> = 10 mA 3 V, I <sub>Load</sub> = 3 mA 5 V, I <sub>Load</sub> = 2 mA 3 V, I <sub>Load</sub> = 0.4 mA	V <sub>OL</sub>	—	—	1.5	V
		— — — —		— — — —	1.5 0.8 0.8		
3	P	Output high current — Max total I <sub>OH</sub> for all ports 5V 3V	I <sub>OHT</sub>	— —	— —	100 60	mA
4	P	Output low current — Max total I <sub>OL</sub> for all ports 5V 3V	I <sub>OLT</sub>	— —	— —	100 60	mA
5	P	Input high voltage; all digital inputs 2.7v ≤ V <sub>DD</sub> 4.5v	V <sub>IH</sub>	0.70xV <sub>DD</sub>	—	—	V
		4.5v ≤ V <sub>DD</sub> ≤ 5.5v	V <sub>IH</sub>	0.65xV <sub>DD</sub>	—	—	
6	P	Input low voltage; all digital inputs	V <sub>IL</sub>	—	—	0.35 x V <sub>DD</sub>	
7	P	Input hysteresis; all digital inputs	V <sub>hys</sub>	0.06 x V <sub>DD</sub>			V
8	P	Input leakage current; input only pins <sup>2</sup>	I <sub>In</sub>	—	0.1	1	μA
9	P	High Impedance (off-state) leakage current <sup>2</sup>	I <sub>OZ</sub>	—	0.1	1	μA
10	P	Internal pullup resistors <sup>3</sup>	R <sub>PU</sub>	20	45	65	kΩ
11	P	Internal pulldown resistors <sup>4</sup>	R <sub>PD</sub>	20	45	65	kΩ
12	C	Input Capacitance; all non-supply pins	C <sub>In</sub>	—	—	8	pF
13	P	POR rearm voltage	V <sub>POR</sub>	0.9	1.4	2.0	V
14	D	POR rearm time	t <sub>POR</sub>	10	—	—	μs

## A.11 SPI Characteristics

Table A-14 and Figure A-15 through Figure A-18 describe the timing requirements for the SPI system.

**Table A-14. SPI Electrical Characteristic**

Num	C	Characteristic <sup>1</sup>	Symbol	Min	Max	Unit
		Operating frequency Master Slave	$f_{op}$ $f_{op}$	$f_{Bus}/2048$ dc	$f_{Bus}/2$ $f_{Bus}/4$	Hz
1		Cycle time Master Slave	$t_{SCK}$ $t_{SCK}$	2 4	2048 —	$t_{cyc}$ $t_{cyc}$
2		Enable lead time Master Slave	$t_{Lead}$ $t_{Lead}$	— 1/2	1/2 —	$t_{SCK}$ $t_{SCK}$
3		Enable lag time Master Slave	$t_{Lag}$ $t_{Lag}$	— 1/2	1/2 —	$t_{SCK}$ $t_{SCK}$
4		Clock (SPSCK) high time Master and Slave	$t_{SCKH}$	$1/2 t_{SCK} - 25$	—	ns
5		Clock (SPSCK) low time Master and Slave	$t_{SCKL}$	$1/2 t_{SCK} - 25$	—	ns
6		Data setup time (inputs) Master Slave	$t_{SI(M)}$ $t_{SI(S)}$	30 30	— —	ns ns
7		Data hold time (inputs) Master Slave	$t_{HI(M)}$ $t_{HI(S)}$	30 30	— —	ns ns
8		Access time, slave <sup>2</sup>	$t_A$	0	40	ns
9		Disable time, slave <sup>3</sup>	$t_{dis}$	—	40	ns
10		Data setup time (outputs) Master Slave	$t_{SO}$ $t_{SO}$	25 25	— —	ns ns
11		Data hold time (outputs) Master Slave	$t_{HO}$ $t_{HO}$	-10 -10	— —	ns ns

<sup>1</sup> All timing is shown with respect to 20%  $V_{DD}$  and 70%  $V_{DD}$ , unless noted; 100 pF load on all SPI pins. All timing assumes slew rate control disabled and high drive strength enabled for SPI output pins.

<sup>2</sup> Time to data active from high-impedance state.

<sup>3</sup> Hold time to high-impedance state.



NOTES:

1. DIMENSIONS AND TOLERANCING PER ASME Y14.5M-1994.
2. CONTROLLING DIMENSION: MILLIMETER
3. DATUM PLANE H IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS L, M AND N TO BE DETERMINED AT DATUM PLANE H.

5. DIMENSIONS TO BE DETERMINED AT SEATING PLANE T.

6. DIMENSIONS DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 PER SIDE. DIMENSIONS DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE H.

7. DIMENSION DOES NOT INCLUDE DAMBAR PROTRUSION. DAMBAR PROTRUSION SHALL NOT CAUSE THE DIMENSION TO EXCEED 0.53. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD OR PROTRUSION 0.07.

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE	
TITLE: 44 LD LQFP, 10 X 10 PKG, 0.8 PITCH, 1.4 THICK	DOCUMENT NO: 98ASS23225W	REV: D	
	CASE NUMBER: 824D-02	26 FEB 2007	
	STANDARD: JEDEC MS-026 BCB		