



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	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I ² C, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	13
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	16-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	16-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08sh4ctgr

Chapter 1

Device Overview

The MC9S08SH8 members of the low-cost, high-performance HCS08 Family of 8-bit microcontroller units (MCUs). All MCUs in the family use the enhanced HCS08 core and are available with a variety of modules, memory sizes, memory types, and package types.

1.1 Devices in the MC9S08SH8 Series

Table 1-1 summarizes the feature set available in the MC9S08SH8 series of MCUs.

Table 1-1. MC9S08SH8 Features by MCU and Package

Feature	9S08SH8				9S08SH4			
FLASH size (bytes)	8192				4096			
RAM size (bytes)	512				256			
Pin quantity	24	20	16	8	24	20	16	8
ACMP	yes							
ADC channels	12	12	8	4	12	12	8	4
DBG	yes							
ICS	yes	yes	yes	yes ¹	yes	yes	yes	yes ¹
IIC	yes							
MTIM	yes							
Pin Interrupts	8	8	8	4	8	8	8	4
Pin I/O ²	17	17	13	5	17	17	13	5
RTC	yes							
SCI	yes	yes	yes	no	yes	yes	yes	no
SPI	yes	yes	yes	no	yes	yes	yes	no
TPM1 channels	2	2	2	1	2	2	2	1
TPM2 channels	2	2	2	1	2	2	2	1
XOSC	yes	yes	yes	no	yes	yes	yes	no

¹ FBE and FEE modes are not available in 8-pin packages.

² Port I/O count does not include the output-only PTA4/ACMPO/BKGD/MS.

The status flag corresponding to the interrupt source must be acknowledged (cleared) before returning from the ISR. Typically, the flag is cleared at the beginning of the ISR so that if another interrupt is generated by this same source, it will be registered so it can be serviced after completion of the current ISR.

5.5.2 External Interrupt Request Pin (IRQ)

External interrupts are managed by the IRQ status and control register, IRQSC. When the IRQ function is enabled, synchronous logic monitors the pin for edge-only or edge-and-level events. When the MCU is in 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 Pin Configuration Option

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 will not be pulled to V_{DD} . The internal gates connected to this pin are pulled to V_{DD} . If the IRQ pin is required to drive to a V_{DD} level an external pullup should be used.

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

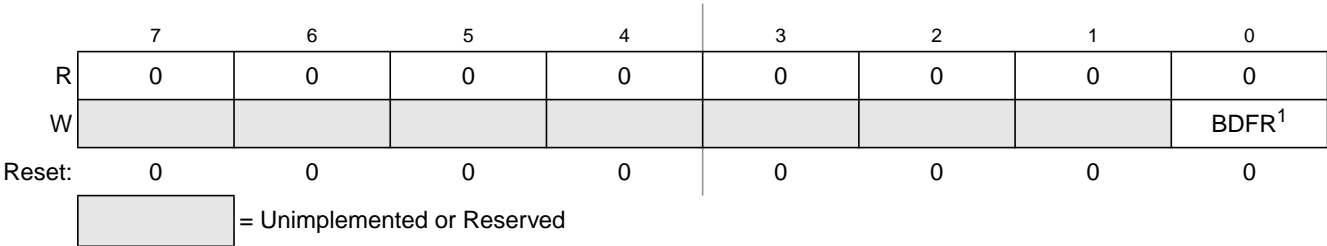
Table 5-2 provides a summary of all interrupt sources. Higher-priority sources are located toward the bottom of the table. The high-order byte of the address for the interrupt service routine is located at the first address in the vector address column, and the low-order byte of the address for the interrupt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

Table 5-4. SRS Register Field Descriptions

Field	Description
3 ILAD	Illegal Address — Reset was caused by an attempt to access either data or an instruction at an unimplemented memory address. 0 Reset not caused by an illegal address 1 Reset caused by an illegal address
1 LVD	Low Voltage Detect — If the LVDRE bit is set and the supply drops below the LVD trip voltage, an LVD reset will occur. This bit is also set by POR. 0 Reset not caused by LVD trip or POR. 1 Reset caused by LVD trip or POR.

5.7.3 System Background Debug Force Reset Register (SBDFR)

This high page register contains a single write-only control bit. A serial background command such as WRITE_BYTE must be used to write to SBDFR. Attempts to write this register from a user program are ignored. Reads always return 0x00.



¹ BDFR is writable only through serial background debug commands, not from user programs.

Figure 5-4. System Background Debug Force Reset Register (SBDFR)

Table 5-5. SBDFR Register Field Descriptions

Field	Description
0 BDFR	Background Debug Force Reset — A serial background command such as WRITE_BYTE can be used to allow an external debug host to force a target system reset. Writing 1 to this bit forces an MCU reset. This bit cannot be written from a user program.

5.7.8 System Power Management Status and Control 2 Register (SPMSC2)

This register is used to report the status of the low voltage warning function, and to configure the stop mode behavior of the MCU.

	7	6	5	4	3	2	1	0
R	0	0	LVDV ¹	LVWV	PPDF	0	0	PPDC ²
W						PPDACK		
Power-on Reset:	0	0	0	0	0	0	0	0
LVD Reset:	0	0	u	u	0	0	0	0
Any other Reset:	0	0	u	u	0	0	0	0

= Unimplemented or Reserved
 u = Unaffected by reset

¹ This bit can be written only one time after power-on reset. Additional writes are ignored.

² This bit can be written only one time after reset. Additional writes are ignored.

Figure 5-10. System Power Management Status and Control 2 Register (SPMSC2)

Table 5-11. SPMSC2 Register Field Descriptions

Field	Description
5 LVDV	Low-Voltage Detect Voltage Select — This write-once bit selects the low voltage detect (LVD) trip point setting. It also selects the warning voltage range. See Table 5-12 .
4 LVWV	Low-Voltage Warning Voltage Select — This bit selects the low voltage warning (LVW) trip point voltage. See Table 5-12 .
3 PPDF	Partial Power Down Flag — This read-only status bit indicates that the MCU has recovered from stop2 mode. 0 MCU has not recovered from stop2 mode. 1 MCU recovered from stop2 mode.
2 PPDACK	Partial Power Down Acknowledge — Writing a 1 to PPDACK clears the PPDF bit
0 PPDC	Partial Power Down Control — This write-once bit controls whether stop2 or stop3 mode is selected. 0 Stop3 mode enabled. 1 Stop2, partial power down, mode enabled.

Table 5-12. LVD and LVW trip point typical values¹

LVDV:LVWV	LVW Trip Point	LVD Trip Point
0:0	$V_{LVW0} = 2.74 \text{ V}$	$V_{LVD0} = 2.56 \text{ V}$
0:1	$V_{LVW1} = 2.92 \text{ V}$	
1:0	$V_{LVW2} = 4.3 \text{ V}$	$V_{LVD1} = 4.0 \text{ V}$
1:1	$V_{LVW3} = 4.6 \text{ V}$	

¹ See Electrical Characteristics appendix for minimum and maximum values.

7.2.3 Stack Pointer (SP)

This 16-bit address pointer register points at the next available location on the automatic last-in-first-out (LIFO) stack. The stack may be located anywhere in the 64-Kbyte address space that has RAM and can be any size up to the amount of available RAM. The stack is used to automatically save the return address for subroutine calls, the return address and CPU registers during interrupts, and for local variables. The AIS (add immediate to stack pointer) instruction adds an 8-bit signed immediate value to SP. This is most often used to allocate or deallocate space for local variables on the stack.

SP is forced to 0x00FF at reset for compatibility with the earlier M68HC05 Family. HCS08 programs normally change the value in SP to the address of the last location (highest address) in on-chip RAM during reset initialization to free up direct page RAM (from the end of the on-chip registers to 0x00FF).

The RSP (reset stack pointer) instruction was included for compatibility with the M68HC05 Family and is seldom used in new HCS08 programs because it only affects the low-order half of the stack pointer.

7.2.4 Program Counter (PC)

The program counter is a 16-bit register that contains the address of the next instruction or operand to be fetched.

During normal program execution, the program counter automatically increments to the next sequential memory location every time an instruction or operand is fetched. Jump, branch, interrupt, and return operations load the program counter with an address other than that of the next sequential location. This is called a change-of-flow.

During reset, the program counter is loaded with the reset vector that is located at 0xFFFFE and 0xFFFF. The vector stored there is the address of the first instruction that will be executed after exiting the reset state.

7.2.5 Condition Code Register (CCR)

The 8-bit condition code register contains the interrupt mask (I) and five flags that indicate the results of the instruction just executed. Bits 6 and 5 are set permanently to 1. The following paragraphs describe the functions of the condition code bits in general terms. For a more detailed explanation of how each instruction sets the CCR bits, refer to the *HCS08 Family Reference Manual, volume 1*, Freescale Semiconductor document order number HCS08RMv1.

Table 7-2. Instruction Set Summary (Sheet 7 of 9)

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR	
						V 1 1 H	I N Z C
RSP	Reset Stack Pointer (Low Byte) $SPL \leftarrow \$FF$ (High Byte Not Affected)	INH	9C	1	p	- 1 1 -	- - - -
RTI	Return from Interrupt $SP \leftarrow (SP) + \$0001$; Pull (CCR) $SP \leftarrow (SP) + \$0001$; Pull (A) $SP \leftarrow (SP) + \$0001$; Pull (X) $SP \leftarrow (SP) + \$0001$; Pull (PCH) $SP \leftarrow (SP) + \$0001$; Pull (PCL)	INH	80	9	uuuuufppp	\uparrow 1 1 \uparrow	\uparrow \uparrow \uparrow \uparrow
RTS	Return from Subroutine $SP \leftarrow SP + \$0001$; Pull (PCH) $SP \leftarrow SP + \$0001$; Pull (PCL)	INH	81	5	ufppp	- 1 1 -	- - - -
SBC #opr8i SBC opr8a SBC opr16a SBC oprx16,X SBC oprx8,X SBC ,X SBC oprx16,SP SBC oprx8,SP	Subtract with Carry $A \leftarrow (A) - (M) - (C)$	IMM DIR EXT IX2 IX1 IX SP2 SP1	A2 ii B2 dd C2 hh ll D2 ee ff E2 ff F2 9E D2 ee ff 9E E2 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rpp pprpp prpp	\uparrow 1 1 -	- \uparrow \uparrow \uparrow
SEC	Set Carry Bit ($C \leftarrow 1$)	INH	99	1	p	- 1 1 -	- - - 1
SEI	Set Interrupt Mask Bit ($I \leftarrow 1$)	INH	9B	1	p	- 1 1 -	1 - - -
STA opr8a STA opr16a STA oprx16,X STA oprx8,X STA ,X STA oprx16,SP STA oprx8,SP	Store Accumulator in Memory $M \leftarrow (A)$	DIR EXT IX2 IX1 IX SP2 SP1	B7 dd C7 hh ll D7 ee ff E7 ff F7 9E D7 ee ff 9E E7 ff	3 4 4 3 2 5 4	wpp pwpp pwpp wpp wp ppwpp pwpp	0 1 1 -	- \uparrow \uparrow -
STHX opr8a STHX opr16a STHX oprx8,SP	Store H:X (Index Reg.) $(M:M + \$0001) \leftarrow (H:X)$	DIR EXT SP1	35 dd 96 hh ll 9E FF ff	4 5 5	wwpp pwwpp pwwpp	0 1 1 -	- \uparrow \uparrow -
STOP	Enable Interrupts: Stop Processing Refer to MCU Documentation $I \text{ bit} \leftarrow 0$; Stop Processing	INH	8E	2	fp...	- 1 1 -	0 - - -
STX opr8a STX opr16a STX oprx16,X STX oprx8,X STX ,X STX oprx16,SP STX oprx8,SP	Store X (Low 8 Bits of Index Register) in Memory $M \leftarrow (X)$	DIR EXT IX2 IX1 IX SP2 SP1	BF dd CF hh ll DF ee ff EF ff FF 9E DF ee ff 9E EF ff	3 4 4 3 2 5 4	wpp pwpp pwpp wpp wp ppwpp pwpp	0 1 1 -	- \uparrow \uparrow -

11.3.4 IIC Status Register (IICS)

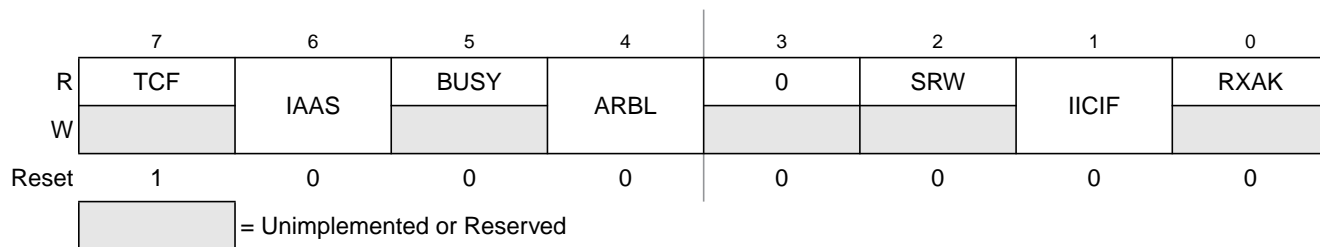


Figure 11-6. IIC Status Register (IICS)

Table 11-7. IICS Field Descriptions

Field	Description
7 TCF	Transfer Complete Flag. This bit is set on the completion of a byte transfer. This bit is only valid during or immediately following a transfer to the IIC module or from the IIC module. The TCF bit is cleared by reading the IICD register in receive mode or writing to the IICD in transmit mode. 0 Transfer in progress 1 Transfer complete
6 IAAS	Addressed as a Slave. The IAAS bit is set when the calling address matches the programmed slave address or when the GCAEN bit is set and a general call is received. Writing the IICC register clears this bit. 0 Not addressed 1 Addressed as a slave
5 BUSY	Bus Busy. The BUSY bit indicates the status of the bus regardless of slave or master mode. The BUSY bit is set when a start signal is detected and cleared when a stop signal is detected. 0 Bus is idle 1 Bus is busy
4 ARBL	Arbitration Lost. This bit is set by hardware when the arbitration procedure is lost. The ARBL bit must be cleared by software by writing a 1 to it. 0 Standard bus operation 1 Loss of arbitration
2 SRW	Slave Read/Write. When addressed as a slave, the SRW bit indicates the value of the R/W command bit of the calling address sent to the master. 0 Slave receive, master writing to slave 1 Slave transmit, master reading from slave
1 IICIF	IIC Interrupt Flag. The IICIF bit is set when an interrupt is pending. This bit must be cleared by software, by writing a 1 to it in the interrupt routine. One of the following events can set the IICIF bit: <ul style="list-style-type: none"> One byte transfer completes Match of slave address to calling address Arbitration lost 0 No interrupt pending 1 Interrupt pending
0 RXAK	Receive Acknowledge. When the RXAK bit is low, it indicates an acknowledge signal has been received after the completion of one byte of data transmission on the bus. If the RXAK bit is high it means that no acknowledge signal is detected. 0 Acknowledge received 1 No acknowledge received

11.7 Initialization/Application Information

Module Initialization (Slave)

1. Write: IICC2
 - to enable or disable general call
 - to select 10-bit or 7-bit addressing mode
2. Write: IICA
 - to set the slave address
3. Write: IICC1
 - to enable IIC and interrupts
4. Initialize RAM variables (IICEN = 1 and IICIE = 1) for transmit data
5. Initialize RAM variables used to achieve the routine shown in [Figure 11-12](#)

Module Initialization (Master)

1. Write: IICF
 - to set the IIC baud rate (example provided in this chapter)
2. Write: IICC1
 - to enable IIC and interrupts
3. Initialize RAM variables (IICEN = 1 and IICIE = 1) for transmit data
4. Initialize RAM variables used to achieve the routine shown in [Figure 11-12](#)
5. Write: IICC1
 - to enable TX
6. Write: IICC1
 - to enable MST (master mode)
7. Write: IICD
 - with the address of the target slave. (The lsb of this byte determines whether the communication is master receive or transmit.)

Module Use

The routine shown in [Figure 11-12](#) can handle both master and slave IIC operations. For slave operation, an incoming IIC message that contains the proper address begins IIC communication. For master operation, communication must be initiated by writing to the IICD register.

Register Model

IICA	AD[7:1]							0
When addressed as a slave (in slave mode), the module responds to this address								
IICF	MULT		ICR					
Baud rate = BUSCLK / (2 x MULT x (SCL DIVIDER))								
IICC1	IICEN	IICIE	MST	TX	TXAK	RSTA	0	0
Module configuration								
IICS	TCF	IAAS	BUSY	ARBL	0	SRW	IICIF	RXAK
Module status flags								
IICD	DATA							
Data register; Write to transmit IIC data read to read IIC data								
IICC2	GCAEN	ADEXT	0	0	0	AD10	AD9	AD8
Address configuration								

Figure 11-11. IIC Module Quick Start

Chapter 12

Modulo Timer (S08MTIMV1)

12.1 Introduction

The MTIM is a simple 8-bit timer with several software selectable clock sources and a programmable interrupt.

The central component of the MTIM is the 8-bit counter, which can operate as a free-running counter or a modulo counter. A timer overflow interrupt can be enabled to generate periodic interrupts for time-based software loops.

Figure 12-1 shows the MC9S08SH8 block diagram with the MTIM module highlighted.

12.1.1 MTIM Configuration Information

The external clock for the MTIM module, TCLK, is selected by setting $CLKS = 1:1$ or $1:0$ in MTIMCLK, which selects the TCLK pin input. The TCLK input on PTA0 can be enabled as external clock inputs to both MTIM and TPM modules simultaneously.

```
#pragma TRAP_PROC
void RTC_ISR(void)
{
    /* Clear the interrupt flag */
    RTCSC.byte = RTCSC.byte | 0x80;
    /* RTC interrupts every 1 Second */
    Seconds++;
    /* 60 seconds in a minute */
    if (Seconds > 59){
        Minutes++;
        Seconds = 0;
    }
    /* 60 minutes in an hour */
    if (Minutes > 59){
        Hours++;
        Minutes = 0;
    }
    /* 24 hours in a day */
    if (Hours > 23){
        Days ++;
        Hours = 0;
    }
}
```

Chapter 15

Serial Peripheral Interface (S08SPIV3)

15.1 Introduction

Figure 15-1 shows the MC9S08SH8 block diagram with the SPI module highlighted.

Table 15-1. SPIC1 Field Descriptions (continued)

Field	Description
4 MSTR	Master/Slave Mode Select 0 SPI module configured as a slave SPI device 1 SPI module configured as a master SPI device
3 CPOL	Clock Polarity — This bit effectively places an inverter in series with the clock signal from a master SPI or to a slave SPI device. Refer to Section 15.5.1, “SPI Clock Formats” for more details. 0 Active-high SPI clock (idles low) 1 Active-low SPI clock (idles high)
2 CPHA	Clock Phase — This bit selects one of two clock formats for different kinds of synchronous serial peripheral devices. Refer to Section 15.5.1, “SPI Clock Formats” for more details. 0 First edge on SPSCCK occurs at the middle of the first cycle of an 8-cycle data transfer 1 First edge on SPSCCK occurs at the start of the first cycle of an 8-cycle data transfer
1 SSOE	Slave Select Output Enable — This bit is used in combination with the mode fault enable (MODFEN) bit in SPCR2 and the master/slave (MSTR) control bit to determine the function of the \overline{SS} pin as shown in Table 15-2 .
0 LSBFE	LSB First (Shifter Direction) 0 SPI serial data transfers start with most significant bit 1 SPI serial data transfers start with least significant bit

Table 15-2. \overline{SS} Pin Function

MODFEN	SSOE	Master Mode	Slave Mode
0	0	General-purpose I/O (not SPI)	Slave select input
0	1	General-purpose I/O (not SPI)	Slave select input
1	0	\overline{SS} input for mode fault	Slave select input
1	1	Automatic \overline{SS} output	Slave select input

NOTE

Ensure that the SPI should not be disabled (SPE=0) at the same time as a bit change to the CPHA bit. These changes should be performed as separate operations or unexpected behavior may occur.

15.4.2 SPI Control Register 2 (SPIC2)

This read/write register is used to control optional features of the SPI system. Bits 7, 6, 5, and 2 are not implemented and always read 0.

	7	6	5	4	3	2	1	0
R	0	0	0	MODFEN	BIDIROE	0	SPISWAI	SPC0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 15-6. SPI Control Register 2 (SPIC2)

Table 15-3. SPIC2 Register Field Descriptions

Field	Description
4 MODFEN	Master Mode-Fault Function Enable — When the SPI is configured for slave mode, this bit has no meaning or effect. (The \overline{SS} pin is the slave select input.) In master mode, this bit determines how the \overline{SS} pin is used (refer to Table 15-2 for more details). 0 Mode fault function disabled, master \overline{SS} pin reverts to general-purpose I/O not controlled by SPI 1 Mode fault function enabled, master \overline{SS} pin acts as the mode fault input or the slave select output
3 BIDIROE	Bidirectional Mode Output Enable — When bidirectional mode is enabled by SPI pin control 0 (SPC0) = 1, BIDIROE determines whether the SPI data output driver is enabled to the single bidirectional SPI I/O pin. Depending on whether the SPI is configured as a master or a slave, it uses either the MOSI (MOMI) or MISO (SISO) pin, respectively, as the single SPI data I/O pin. When SPC0 = 0, BIDIROE has no meaning or effect. 0 Output driver disabled so SPI data I/O pin acts as an input 1 SPI I/O pin enabled as an output
1 SPISWAI	SPI Stop in Wait Mode 0 SPI clocks continue to operate in wait mode 1 SPI clocks stop when the MCU enters wait mode
0 SPC0	SPI Pin Control 0 — The SPC0 bit chooses single-wire bidirectional mode. If MSTR = 0 (slave mode), the SPI uses the MISO (SISO) pin for bidirectional SPI data transfers. If MSTR = 1 (master mode), the SPI uses the MOSI (MOMI) pin for bidirectional SPI data transfers. When SPC0 = 1, BIDIROE is used to enable or disable the output driver for the single bidirectional SPI I/O pin. 0 SPI uses separate pins for data input and data output 1 SPI configured for single-wire bidirectional operation

15.4.3 SPI Baud Rate Register (SPIBR)

This register is used to set the prescaler and bit rate divisor for an SPI master. This register may be read or written at any time.

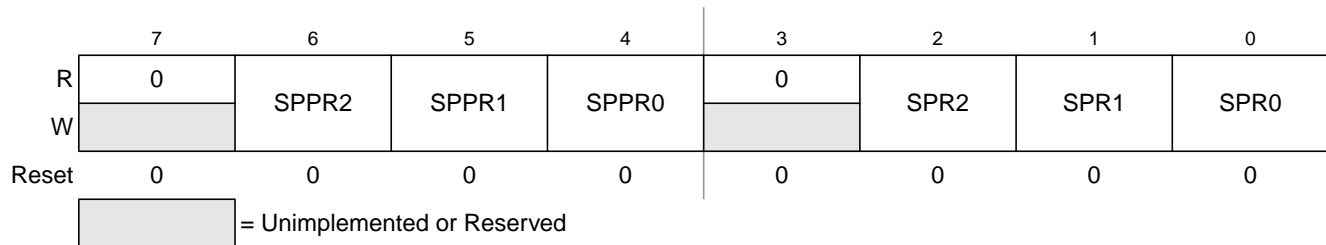


Figure 15-7. SPI Baud Rate Register (SPIBR)

Table 15-4. SPIBR Register Field Descriptions

Field	Description
6:4 SPPR[2:0]	SPI Baud Rate Prescale Divisor — This 3-bit field selects one of eight divisors for the SPI baud rate prescaler as shown in Table 15-5. The input to this prescaler is the bus rate clock (BUSCLK). The output of this prescaler drives the input of the SPI baud rate divider (see Figure 15-4).
2:0 SPR[2:0]	SPI Baud Rate Divisor — This 3-bit field selects one of eight divisors for the SPI baud rate divider as shown in Table 15-6. The input to this divider comes from the SPI baud rate prescaler (see Figure 15-4). The output of this divider is the SPI bit rate clock for master mode.

The TPM channels are programmable independently as input capture, output compare, or edge-aligned PWM channels. Alternately, the TPM can be configured to produce CPWM outputs on all channels. When the TPM is configured for CPWMs, the counter operates as an up/down counter; input capture, output compare, and EPWM functions are not practical.

If a channel is configured as input capture, an internal pullup device may be enabled for that channel. The details of how a module interacts with pin controls depends upon the chip implementation because the I/O pins and associated general purpose I/O controls are not part of the module. Refer to the discussion of the I/O port logic in a full-chip specification.

Because center-aligned PWMs are usually used to drive 3-phase AC-induction motors and brushless DC motors, they are typically used in sets of three or six channels.

16.2 Signal Description

Table 16-2 shows the user-accessible signals for the TPM. The number of channels may be varied from one to eight. When an external clock is included, it can be shared with the same pin as any TPM channel; however, it could be connected to a separate input pin. Refer to the I/O pin descriptions in full-chip specification for the specific chip implementation.

Table 16-2. Signal Properties

Name	Function
EXTCLK ¹	External clock source which may be selected to drive the TPM counter.
TPMxCHn ²	I/O pin associated with TPM channel n

¹ When preset, this signal can share any channel pin; however depending upon full-chip implementation, this signal could be connected to a separate external pin.

² n=channel number (1 to 8)

Refer to documentation for the full-chip for details about reset states, port connections, and whether there is any pullup device on these pins.

TPM channel pins can be associated with general purpose I/O pins and have passive pullup devices which can be enabled with a control bit when the TPM or general purpose I/O controls have configured the associated pin as an input. When no TPM function is enabled to use a corresponding pin, the pin reverts to being controlled by general purpose I/O controls, including the port-data and data-direction registers. Immediately after reset, no TPM functions are enabled, so all associated pins revert to general purpose I/O control.

16.2.1 Detailed Signal Descriptions

This section describes each user-accessible pin signal in detail. Although Table 16-2 grouped all channel pins together, any TPM pin can be shared with the external clock source signal. Since I/O pin logic is not part of the TPM, refer to full-chip documentation for a specific derivative for more details about the interaction of TPM pin functions and general purpose I/O controls including port data, data direction, and pullup controls.

Table 16-7. Mode, Edge, and Level Selection

CPWMS	MSnB:MSnA	ELSnB:ELSnA	Mode	Configuratio
0	00	01	Input capture	Capture on rising edge only
		10		Capture on falling edge only
		11		Capture on rising or falling edge
	01	01	Output compare	Toggle output on compare
		10		Clear output on compare
		11		Set output on compare
	1X	10	Edge-aligned PWM	High-true pulses (clear output on compare)
		X1		Low-true pulses (set output on compare)
1	XX	10	Center-aligned PWM	High-true pulses (clear output on compare-up)
		X1		Low-true pulses (set output on compare-up)

16.3.5 TPM Channel Value Registers (TPMxCnVH:TPMxCnVL)

These read/write registers contain the captured TPM counter value of the input capture function or the output compare value for the output compare or PWM functions. The channel registers are cleared by reset.

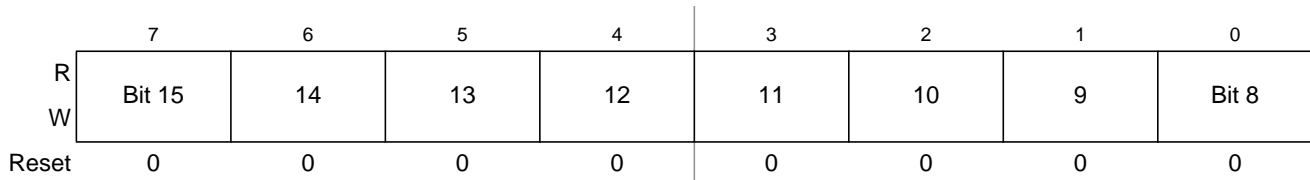


Figure 16-13. TPM Channel Value Register High (TPMxCnVH)

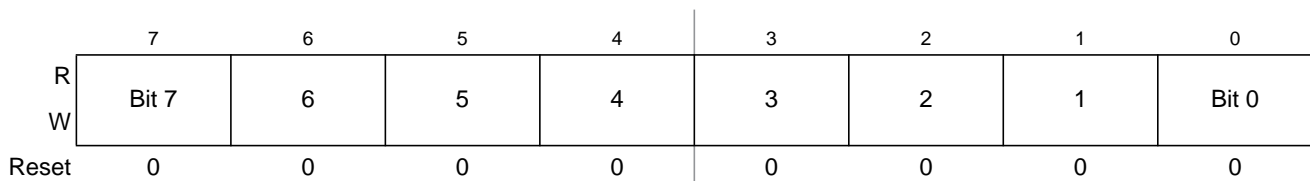


Figure 16-14. TPM Channel Value Register Low (TPMxCnVL)

In input capture mode, reading either byte (TPMxCnVH or TPMxCnVL) latches the contents of both bytes into a buffer where they remain latched until the other half is read. This latching mechanism also resets

The following sections describe the main counter and each of the timer operating modes (input capture, output compare, edge-aligned PWM, and center-aligned PWM). Because details of pin operation and interrupt activity depend upon the operating mode, these topics will be covered in the associated mode explanation sections.

16.4.1 Counter

All timer functions are based on the main 16-bit counter (TPMxCNTH:TPMxCNTL). This section discusses selection of the clock source, end-of-count overflow, up-counting vs. up/down counting, and manual counter reset.

16.4.1.1 Counter Clock Source

The 2-bit field, CLKSB:CLKSA, in the timer status and control register (TPMxSC) selects one of three possible clock sources or OFF (which effectively disables the TPM). See [Table 16-4](#). After any MCU reset, CLKSB:CLKSA=0:0 so no clock source is selected, and the TPM is in a very low power state. These control bits may be read or written at any time and disabling the timer (writing 00 to the CLKSB:CLKSA field) does not affect the values in the counter or other timer registers.

Table 16-8. TPM Clock Source Selection

CLKSB:CLKSA	TPM Clock Source to Prescaler Input
00	No clock selected (TPM counter disabled)
01	Bus rate clock
10	Fixed system clock
11	External source

The bus rate clock is the main system bus clock for the MCU. This clock source requires no synchronization because it is the clock that is used for all internal MCU activities including operation of the CPU and buses.

In MCUs that have no PLL and FLL or the PLL and FLL are not engaged, the fixed system clock source is the same as the bus-rate-clock source, and it does not go through a synchronizer. When a PLL or FLL is present and engaged, a synchronizer is required between the crystal divided-by two clock source and the timer counter so counter transitions will be properly aligned to bus-clock transitions. A synchronizer will be used at chip level to synchronize the crystal-related source clock to the bus clock.

The external clock source may be connected to any TPM channel pin. This clock source always has to pass through a synchronizer to assure that counter transitions are properly aligned to bus clock transitions. The bus-rate clock drives the synchronizer; therefore, to meet Nyquist criteria even with jitter, the frequency of the external clock source must not be faster than the bus rate divided-by four. With ideal clocks the external clock can be as fast as bus clock divided by four.

When the external clock source shares the TPM channel pin, this pin should not be used for other channel timing functions. For example, it would be ambiguous to configure channel 0 for input capture when the TPM channel 0 pin was also being used as the timer external clock source. (It is the user's responsibility

In this case, the TPM v3 waits for the start of a new PWM period to begin using the new duty cycle setting. Instead, the TPM v2 changes the channel output at the middle of the current PWM period (when the count reaches 0x0000).

- TPMxCnVH:L is changed from a non-zero value to 0x0000 [SE110-TPM case 4]

In this case, the TPM v3 finishes the current PWM period using the old duty cycle setting. Instead, the TPM v2 finishes the current PWM period using the new duty cycle setting.

6. Write to TPMxMODH:L registers in BDM mode (Section 16.3.3, “TPM Counter Modulo Registers (TPMxMODH:TPMxMODL)”)

In the TPM v3 a write to TPMxSC register in BDM mode clears the write coherency mechanism of TPMxMODH:L registers. Instead, in the TPM v2 this coherency mechanism is not cleared when there is a write to TPMxSC register.

7. Update of EPWM signal when CLKSb:CLKSA = 00

In the TPM v3 if CLKSb:CLKSA = 00, then the EPWM signal in the channel output is not update (it is frozen while CLKSb:CLKSA = 00). Instead, in the TPM v2 the EPWM signal is updated at the next rising edge of bus clock after a write to TPMxCnSC register.

The Figure 0-1 and Figure 0-2 show when the EPWM signals generated by TPM v2 and TPM v3 after the reset (CLKSb:CLKSA = 00) and if there is a write to TPMxCnSC register.

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

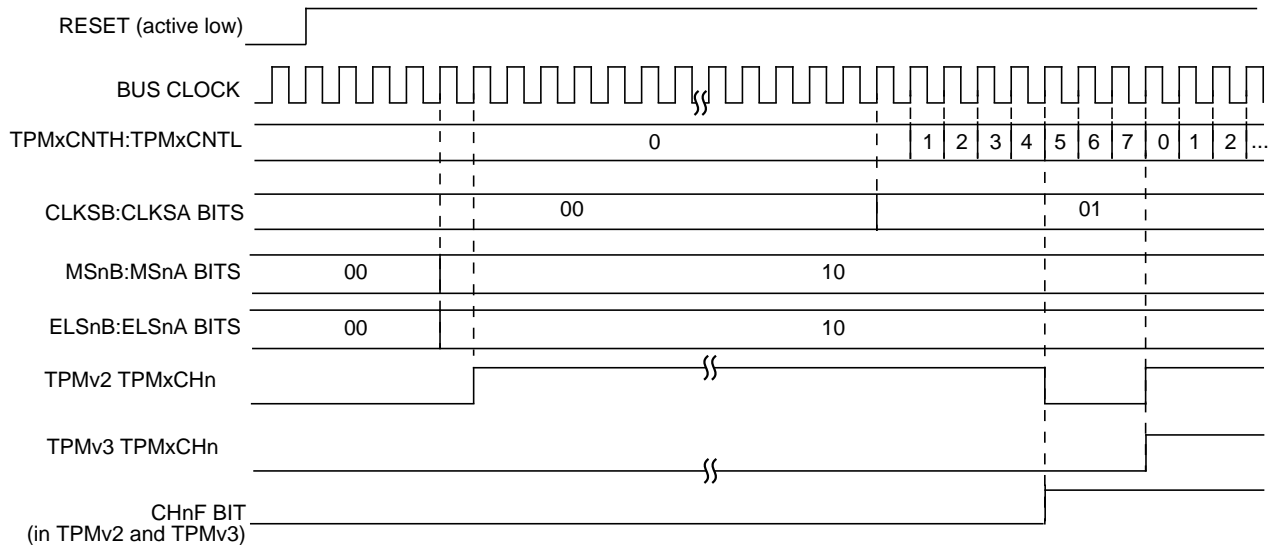


Figure 0-1. Generation of high-true EPWM signal by TPM v2 and v3 after the reset

Table 17-2. BDCSCR Register Field Descriptions (continued)

Field	Description
2 WS	Wait or Stop Status — When the target CPU is in wait or stop mode, most BDC commands cannot function. However, the BACKGROUND command can be used to force the target CPU out of wait or stop and into active background mode where all BDC commands work. Whenever the host forces the target MCU into active background mode, the host should issue a READ_STATUS command to check that BDMACT = 1 before attempting other BDC commands. 0 Target CPU is running user application code or in active background mode (was not in wait or stop mode when background became active) 1 Target CPU is in wait or stop mode, or a BACKGROUND command was used to change from wait or stop to active background mode
1 WSF	Wait or Stop Failure Status — This status bit is set if a memory access command failed due to the target CPU executing a wait or stop instruction at or about the same time. The usual recovery strategy is to issue a BACKGROUND command to get out of wait or stop mode into active background mode, repeat the command that failed, then return to the user program. (Typically, the host would restore CPU registers and stack values and re-execute the wait or stop instruction.) 0 Memory access did not conflict with a wait or stop instruction 1 Memory access command failed because the CPU entered wait or stop mode
0 DVF	Data Valid Failure Status — This status bit is not used in the MC9S08SH8 because it does not have any slow access memory. 0 Memory access did not conflict with a slow memory access 1 Memory access command failed because CPU was not finished with a slow memory access

17.4.1.2 BDC Breakpoint Match Register (BDCBKPT)

This 16-bit register holds the address for the hardware breakpoint in the BDC. The BKPTEN and FTS control bits in BDCSCR are used to enable and configure the breakpoint logic. Dedicated serial BDC commands (READ_BKPT and WRITE_BKPT) are used to read and write the BDCBKPT register but is not accessible to user programs because it is not located in the normal memory map of the MCU. Breakpoints are normally set while the target MCU is in active background mode before running the user application program. For additional information about setup and use of the hardware breakpoint logic in the BDC, refer to [Section 17.2.4, “BDC Hardware Breakpoint.”](#)

17.4.2 System Background Debug Force Reset Register (SBDFFR)

This register contains a single write-only control bit. A serial background mode command such as WRITE_BYTE must be used to write to SBDFFR. Attempts to write this register from a user program are ignored. Reads always return 0x00.

A.8 External Oscillator (XOSC) Characteristics

Table A-8. Oscillator Electrical Specifications (Temperature Range = -40 to 125°C Ambient)

Num	C	Rating	Symbol	Min	Typ ¹	Max	Unit
1	C	Oscillator crystal or resonator (EREFS = 1, ERCLKEN = 1)					
		Low range (RANGE = 0)	f_{lo}	32	—	38.4	kHz
		High range (RANGE = 1) FEE or FBE mode ²	f_{hi}	1	—	5	MHz
		High range (RANGE = 1, HGO = 1) FBELP mode	f_{hi-hgo}	1	—	16	MHz
		High range (RANGE = 1, HGO = 0) FBELP mode	f_{hi-lp}	1	—	8	MHz
2	—	Load capacitors	C_1, C_2	See crystal or resonator manufacturer's recommendation.			
3	—	Feedback resistor	R_F				
		Low range (32 kHz to 100 kHz)		—	10	—	MΩ
		High range (1 MHz to 16 MHz)		—	1	—	
4	—	Series resistor	R_S				
		Low range, low gain (RANGE = 0, HGO = 0)		—	0	—	
		Low range, high gain (RANGE = 0, HGO = 1)		—	100	—	
		High range, low gain (RANGE = 1, HGO = 0)		—	0	—	
		High range, high gain (RANGE = 1, HGO = 1)					
		≥ 8 MHz		—	0	0	
		4 MHz		—	0	10	
		1 MHz		—	0	20	
5	T	Crystal start-up time ³					
		Low range, low gain (RANGE = 0, HGO = 0)	$t_{CSTL-LP}$	—	200	—	
		Low range, high gain (RANGE = 0, HGO = 1)	$t_{CSTL-HGO}$	—	400	—	
		High range, low gain (RANGE = 1, HGO = 0) ⁴	$t_{CSTH-LP}$	—	5	—	
		High range, high gain (RANGE = 1, HGO = 1) ⁴	$t_{CSTH-HGO}$	—	20	—	
6	T	Square wave input clock frequency (EREFS = 0, ERCLKEN = 1)					
		FEE or FBE mode ²	f_{extal}	0.03125	—	5	MHz
		FBELP mode		0	—	40	MHz

¹ Typical data was characterized at 5.0 V, 25°C or is recommended value.

² The input clock source must be divided using RDIV to within the range of 31.25 kHz to 39.0625 kHz.

³ This parameter is characterized and not tested on each device. Proper PC board layout procedures must be followed to achieve specifications.

⁴ 4 MHz crystal



NOTES:

- 1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M 1994.
- 2. CONTROLLING DIMENSION: INCH.
- 3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.
- 4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

DIM	MILLIMETERS		DIM	INCHES		DIM	MILLIMETERS		DIM	INCHES	
	MIN	MAX		MIN	MAX		MIN	MAX		MIN	MAX
A	24.39	24.99		0.960	0.984						
B	6.96	7.49		0.274	0.295						
C	3.56	5.08		0.140	0.200						
D	0.38	0.56		0.015	0.022						
E	1.27 BSC			0.050 BSC							
F	1.14	1.52		0.045	0.060						
G	2.54 BSC			0.100 BSC							
J	0.20	0.38		0.008	0.015						
K	2.79	3.76		0.110	0.148						
L	7.62 BSC			0.300 BSC							
M	0°	15°		0°	15°						
N	0.50	1.01		0.020	0.040						
R	1.29		0.051						
© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.				MECHANICAL OUTLINE			PRINT VERSION NOT TO SCALE				
TITLE: 20LD .300 PDIP						DOCUMENT NO: 98ASB42899B			REV: B		
						CASE NUMBER: 738C-01			24 MAY 2005		
						STANDARD: NON-JEDEC					