

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	Obsolete
Core Processor	HCS08
Core Size	8-Bit
Speed	48MHz
Connectivity	I <sup>2</sup> C, LINbus, SCI, SPI, USB
Peripherals	LVD, POR, PWM, WDT
Number of I/O	37
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x12b SAR
Oscillator Type	External, Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08jm32cgt">https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08jm32cgt</a>

11.6	Interrupts .....	173
11.6.1	Byte Transfer Interrupt .....	173
11.6.2	Address Detect Interrupt .....	174
11.6.3	Arbitration Lost Interrupt .....	174
11.7	Initialization/Application Information .....	175

## Chapter 12 Multi-Purpose Clock Generator (S08MCGV1)

12.1	Introduction .....	179
12.1.1	Features .....	181
12.1.2	Modes of Operation .....	183
12.2	External Signal Description .....	183
12.3	Register Definition .....	184
12.3.1	MCG Control Register 1 (MCGC1) .....	184
12.3.2	MCG Control Register 2 (MCGC2) .....	185
12.3.3	MCG Trim Register (MCGTRM) .....	186
12.3.4	MCG Status and Control Register (MCGSC) .....	187
12.3.5	MCG Control Register 3 (MCGC3) .....	188
12.4	Functional Description .....	190
12.4.1	Operational Modes .....	190
12.4.2	Mode Switching .....	194
12.4.3	Bus Frequency Divider .....	194
12.4.4	Low Power Bit Usage .....	195
12.4.5	Internal Reference Clock .....	195
12.4.6	External Reference Clock .....	195
12.4.7	Fixed Frequency Clock .....	195
12.5	Initialization / Application Information .....	196
12.5.1	MCG Module Initialization Sequence .....	196
12.5.2	MCG Mode Switching .....	197
12.5.3	Calibrating the Internal Reference Clock (IRC) .....	208

## Chapter 13 Real-Time Counter (S08RTCV1)

13.1	Introduction .....	211
13.1.1	Features .....	213
13.1.2	Modes of Operation .....	213
13.1.3	Block Diagram .....	214
13.2	External Signal Description .....	214
13.3	Register Definition .....	214
13.3.1	RTC Status and Control Register (RTCSC) .....	215
13.3.2	RTC Counter Register (RTCCNT) .....	216
13.3.3	RTC Modulo Register (RTCMOD) .....	216
13.4	Functional Description .....	216
13.4.1	RTC Operation Example .....	217
13.5	Initialization/Application Information .....	218

## Appendix A Electrical Characteristics

A.1	Introduction .....	347
A.2	Parameter Classification .....	347
A.3	Absolute Maximum Ratings .....	347
A.4	Thermal Characteristics .....	348
A.5	ESD Protection and Latch-Up Immunity .....	349
A.6	DC Characteristics .....	350
A.7	Supply Current Characteristics .....	355
A.8	Analog Comparator (ACMP) Electricals .....	356
A.9	ADC Characteristics .....	356
A.10	External Oscillator (XOSC) Characteristics .....	360
A.11	MCG Specifications .....	361
A.12	AC Characteristics .....	362
	A.12.1 Control Timing .....	362
	A.12.2 Timer/PWM (TPM) Module Timing .....	363
	A.12.3 SPI Characteristics .....	364
A.13	Flash Specifications .....	367
A.14	USB Electricals .....	368
A.15	EMC Performance .....	368
	A.15.1 Radiated Emissions .....	368

## Appendix B Ordering Information and Mechanical Drawings

B.1	Ordering Information .....	371
B.2	Orderable Part Numbering System .....	371
B.3	Mechanical Drawings .....	371

Table 1-2 lists the functional versions of the on-chip modules.

**Table 1-2. Versions of On-Chip Modules**

Module	Version
Analog Comparator (ACMP)	2
Analog-to-Digital Converter (ADC)	1
Central Processing Unit (CPU)	2
IIC Module (IIC)	2
Keyboard Interrupt (KBI)	2
Multi-Purpose Clock Generator (MCG)	1
Real-Time Counter (RTC)	1
Serial Communications Interface (SCI)	4
16-bit Serial Peripheral Interface (SPI16)	1
Timer Pulse-Width Modulator (TPM)	3
Universal Serial Bus (USB)	1
Debug Module (DBG)	2

### 1.3 System Clock Distribution

Figure 1-2 shows a simplified clock connection diagram. Some modules in the MCU have selectable clock inputs as shown. The clock inputs to the modules indicate the clock(s) that are used to drive the module function. All memory mapped registers associated with the modules are clocked with BUSCLK.

Table 4-2. Direct-Page Register Summary (Sheet 3 of 4)

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
0x0054	SPI1DH	Bit 15	14	13	12	11	10	9	Bit 8
0x0055	SPI1DL	Bit 7	6	5	4	3	2	1	Bit 0
0x0056	SPI1MH	Bit 15	14	13	12	11	10	9	Bit 8
0x0057	SPI1ML	Bit 7	6	5	4	3	2	1	Bit 0
0x0058	IICA	AD7	AD6	AD5	AD4	AD3	AD2	AD1	0
0x0059	IICF	MULT			ICR				
0x005A	IICC1	IICEN	IICIE	MST	TX	TXAK	RSTA	0	0
0x005B	IICS	TCF	IAAS	BUSY	ARBL	0	SRW	IICIF	RXAK
0x005C	IICD	DATA							
0x005D	IICC2	GCAEN	ADEXT	0	0	0	AD10	AD9	AD8
0x005E	Reserved	—	—	—	—	—	—	—	—
0x005F		—	—	—	—	—	—	—	—
0x0060	TPM2SC	TOF	TOIE	CPWMS	CLKSB	CLKSA	PS2	PS1	PS0
0x0061	TPM2CNTH	Bit 15	14	13	12	11	10	9	Bit 8
0x0062	TPM2CNTL	Bit 7	6	5	4	3	2	1	Bit 0
0x0063	TPM2MODH	Bit 15	14	13	12	11	10	9	Bit 8
0x0064	TPM2MODL	Bit 7	6	5	4	3	2	1	Bit 0
0x0065	TPM2C0SC	CH0F	CH0IE	MS0B	MS0A	ELS0B	ELS0A	0	0
0x0066	TPM2C0VH	Bit 15	14	13	12	11	10	9	Bit 8
0x0067	TPM2C0VL	Bit 7	6	5	4	3	2	1	Bit 0
0x0068	TPM2C1SC	CH1F	CH1IE	MS1B	MS1A	ELS1B	ELS1A	0	0
0x0069	TPM2C1VH	Bit 15	14	13	12	11	10	9	Bit 8
0x006A	TPM2C1VL	Bit 7	6	5	4	3	2	1	Bit 0
0x006B	Reserved	—	—	—	—	—	—	—	—
0x006C	RTCSC	RTIF	RTCLKS		RTIE	RTCPS			
0x006D	RTCCNT	RTCCNT							
0x006E	RTCMOD	RTCMOD							
0x006F	Reserved	—	—	—	—	—	—	—	—
0x0070	SPI2C1	SPIE	SPE	SPTIE	MSTR	CPOL	CPHA	SSOE	LSBFE
0x0071	SPI2C2	SPMIE	SPIMODE	0	MODFEN	BIDIROE	0	SPISWAI	SPC0
0x0072	SPI2BR	0	SPPR2	SPPR1	SPPR0	0	SPR2	SPR1	SPR0
0x0073	SPI2S	SPRF	SPMF	SPTEF	MODF	0	0	0	0
0x0074	SPI2DH	Bit 15	14	13	12	11	10	9	Bit 8
0x0075	SPI2DL	Bit 7	6	5	4	3	2	1	Bit 0
0x0076	SPI2MH	Bit 15	14	13	12	11	10	9	Bit 8
0x0077	SPI2ML	Bit 7	6	5	4	3	2	1	Bit 0
0x0078	Reserved	—	—	—	—	—	—	—	—
0x007F		—	—	—	—	—	—	—	—
0x0080	USBCTL0	USBRESET	USBPU	USBRESMEN	LPRESF	—	USBVREN	—	USBPHYEN
0x0081	Reserved	—	—	—	—	—	—	—	—
0x0087		—	—	—	—	—	—	—	—
0x0088	PERID	0	0	ID5	ID4	ID3	ID2	ID1	ID0

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

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR				
						VH	I	N	Z	C
BCLR <i>n,opr8a</i>	Clear Bit <i>n</i> in Memory ( $M_n \leftarrow 0$ )	DIR (b0)	11 dd	5	rfwpp	--	-	-	-	-
		DIR (b1)	13 dd	5	rfwpp					
		DIR (b2)	15 dd	5	rfwpp					
		DIR (b3)	17 dd	5	rfwpp					
		DIR (b4)	19 dd	5	rfwpp					
		DIR (b5)	1B dd	5	rfwpp					
		DIR (b6)	1D dd	5	rfwpp					
		DIR (b7)	1F dd	5	rfwpp					
BCS <i>rel</i>	Branch if Carry Bit Set (if C = 1) (Same as BLO)	REL	25 rr	3	ppp	--	-	-	-	-
BEQ <i>rel</i>	Branch if Equal (if Z = 1)	REL	27 rr	3	ppp	--	-	-	-	-
BGE <i>rel</i>	Branch if Greater Than or Equal To (if $N \oplus V = 0$ ) (Signed)	REL	90 rr	3	ppp	--	-	-	-	-
BGND	Enter active background if ENBDM=1 Waits for and processes BDM commands until GO, TRACE1, or TAGGO	INH	82	5+	fp...ppp	--	-	-	-	-
BGT <i>rel</i>	Branch if Greater Than (if $Z \mid (N \oplus V) = 0$ ) (Signed)	REL	92 rr	3	ppp	--	-	-	-	-
BHCC <i>rel</i>	Branch if Half Carry Bit Clear (if H = 0)	REL	28 rr	3	ppp	--	-	-	-	-
BHCS <i>rel</i>	Branch if Half Carry Bit Set (if H = 1)	REL	29 rr	3	ppp	--	-	-	-	-
BHI <i>rel</i>	Branch if Higher (if $C \mid Z = 0$ )	REL	22 rr	3	ppp	--	-	-	-	-
BHS <i>rel</i>	Branch if Higher or Same (if C = 0) (Same as BCC)	REL	24 rr	3	ppp	--	-	-	-	-
BIH <i>rel</i>	Branch if IRQ Pin High (if IRQ pin = 1)	REL	2F rr	3	ppp	--	-	-	-	-
BIL <i>rel</i>	Branch if IRQ Pin Low (if IRQ pin = 0)	REL	2E rr	3	ppp	--	-	-	-	-
BIT # <i>opr8i</i> BIT <i>opr8a</i> BIT <i>opr16a</i> BIT <i>opr16,X</i> BIT <i>opr8,X</i> BIT <i>,X</i> BIT <i>opr16,SP</i> BIT <i>opr8,SP</i>	Bit Test (A) & (M) (CCR Updated but Operands Not Changed)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A5 ii B5 dd C5 hh ll D5 ee ff E5 ff F5 9E D5 ee ff 9E E5 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	0-	-	↑	↓	-
BLE <i>rel</i>	Branch if Less Than or Equal To (if $Z \mid (N \oplus V) = 1$ ) (Signed)	REL	93 rr	3	ppp	--	-	-	-	-
BLO <i>rel</i>	Branch if Lower (if C = 1) (Same as BCS)	REL	25 rr	3	ppp	--	-	-	-	-
BLS <i>rel</i>	Branch if Lower or Same (if $C \mid Z = 1$ )	REL	23 rr	3	ppp	--	-	-	-	-
BLT <i>rel</i>	Branch if Less Than (if $N \oplus V = 1$ ) (Signed)	REL	91 rr	3	ppp	--	-	-	-	-
BMC <i>rel</i>	Branch if Interrupt Mask Clear (if I = 0)	REL	2C rr	3	ppp	--	-	-	-	-
BMI <i>rel</i>	Branch if Minus (if N = 1)	REL	2B rr	3	ppp	--	-	-	-	-
BMS <i>rel</i>	Branch if Interrupt Mask Set (if I = 1)	REL	2D rr	3	ppp	--	-	-	-	-
BNE <i>rel</i>	Branch if Not Equal (if Z = 0)	REL	26 rr	3	ppp	--	-	-	-	-
BPL <i>rel</i>	Branch if Plus (if N = 0)	REL	2A rr	3	ppp	--	-	-	-	-

## 9.1.1 Features

The KBI features include:

- Up to eight keyboard interrupt pins with individual pin enable bits.
- Each keyboard interrupt pin is programmable as falling edge (or rising edge) only, or both falling edge and low level (or both rising edge and high level) interrupt sensitivity.
- One software enabled keyboard interrupt.
- Exit from low-power modes.

## 9.1.2 Modes of Operation

This section defines the KBI operation in wait, stop, and background debug modes.

### 9.1.2.1 KBI in Wait Mode

The KBI continues to operate in wait mode if enabled before executing the WAIT instruction. Therefore, an enabled KBI pin ( $KBPEx = 1$ ) can be used to bring the MCU out of wait mode if the KBI interrupt is enabled ( $KBIE = 1$ ).

### 9.1.2.2 KBI in Stop Modes

The KBI operates asynchronously in stop3 mode if enabled before executing the STOP instruction. Therefore, an enabled KBI pin ( $KBPEx = 1$ ) can be used to bring the MCU out of stop3 mode if the KBI interrupt is enabled ( $KBIE = 1$ ).

During either stop1 or stop2 mode, the KBI is disabled. In some systems, the pins associated with the KBI may be sources of wakeup from stop1 or stop2, see the stop modes section in the [Modes of Operation](#) chapter. Upon wake-up from stop1 or stop2 mode, the KBI module will be in the reset state.

### 9.1.2.3 KBI in Active Background Mode

When the microcontroller is in active background mode, the KBI will continue to operate normally.

## 9.1.3 Block Diagram

The block diagram for the keyboard interrupt module is shown [Figure 9-2](#).

### 10.2.1 Analog Power ( $V_{DDAD}$ )

The ADC analog portion uses  $V_{DDAD}$  as its power connection. In some packages,  $V_{DDAD}$  is connected internally to  $V_{DD}$ . If externally available, connect the  $V_{DDAD}$  pin to the same voltage potential as  $V_{DD}$ . External filtering may be necessary to ensure clean  $V_{DDAD}$  for good results.

### 10.2.2 Analog Ground ( $V_{SSAD}$ )

The ADC analog portion uses  $V_{SSAD}$  as its ground connection. In some packages,  $V_{SSAD}$  is connected internally to  $V_{SS}$ . If externally available, connect the  $V_{SSAD}$  pin to the same voltage potential as  $V_{SS}$ .

### 10.2.3 Voltage Reference High ( $V_{REFH}$ )

$V_{REFH}$  is the high reference voltage for the converter. In some packages,  $V_{REFH}$  is connected internally to  $V_{DDAD}$ . If externally available,  $V_{REFH}$  may be connected to the same potential as  $V_{DDAD}$  or may be driven by an external source between the minimum  $V_{DDAD}$  spec and the  $V_{DDAD}$  potential ( $V_{REFH}$  must never exceed  $V_{DDAD}$ ).

### 10.2.4 Voltage Reference Low ( $V_{REFL}$ )

$V_{REFL}$  is the low-reference voltage for the converter. In some packages,  $V_{REFL}$  is connected internally to  $V_{SSAD}$ . If externally available, connect the  $V_{REFL}$  pin to the same voltage potential as  $V_{SSAD}$ .

### 10.2.5 Analog Channel Inputs (ADx)

The ADC module supports up to 28 separate analog inputs. An input is selected for conversion through the ADCH channel select bits.

## 10.3 Register Definition

These memory-mapped registers control and monitor operation of the ADC:

- Status and control register, ADCSC1
- Status and control register, ADCSC2
- Data result registers, ADCRH and ADCRL
- Compare value registers, ADCCVH and ADCCVL
- Configuration register, ADCCFG
- Pin control registers, APCTL1, APCTL2, APCTL3

### 10.3.1 Status and Control Register 1 (ADCSC1)

This section describes the function of the ADC status and control register (ADCSC1). Writing ADCSC1 aborts the current conversion and initiates a new conversion (if the ADCH bits are equal to a value other than all 1s).



## 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

### Register Model

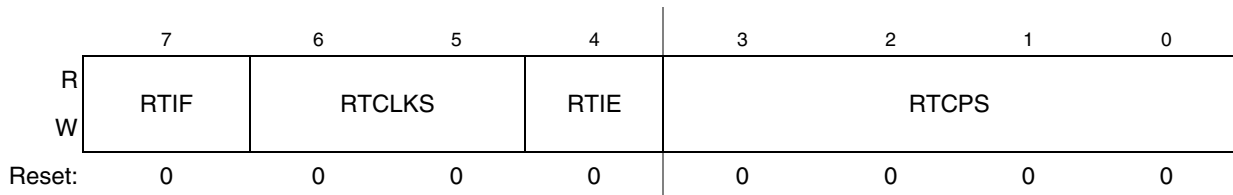
IICA	AD[7:1]							0
When addressed as a slave (in slave mode), the module responds to this address								
IICF	MULT				ICR			
Baud rate = BUSCLK / (2 x MULT x (SCL DIVIDER))								
IICC1	IICEN	IICIE	MST	TX	TXAK	RSTA	0	0
Module configuration								
IICS	TCF	IAAS	BUSY	ARBL	0	SRW	IICIF	RXAK
Module status 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**

- c) MCGC1 = 0xB8 (%10111000)
    - CLKS (bits 7 and 6) set to %10 in order to select external reference clock as system clock source
    - RDIV (bits 5-3) set to %111, or divide-by-128 because  $4 \text{ MHz} / 128 = 31.25 \text{ kHz}$  which is in the 31.25 kHz to 39.0625 kHz range required by the FLL
    - IREFS (bit 2) cleared to 0, selecting the external reference clock
  - d) Loop until IREFST (bit 4) in MCGSC is 0, indicating the external reference is the current source for the reference clock
  - e) Loop until CLKST (bits 3 and 2) in MCGSC are %10, indicating that the external reference clock is selected to feed MCGOUT
2. Then, FBE must transition either directly to PBE mode or first through BLPE mode and then to PBE mode:
    - a) BLPE: If a transition through BLPE mode is desired, first set LP (bit 3) in MCGC2 to 1.
    - b) BLPE/PBE: MCGC1 = 0x90 (%10010000)
      - RDIV (bits 5-3) set to %010, or divide-by-4 because  $4 \text{ MHz} / 4 = 1 \text{ MHz}$  which is in the 1 MHz to 2 MHz range required by the PLL. In BLPE mode, the configuration of the RDIV does not matter because both the FLL and PLL are disabled. Changing them only sets up the the dividers for PLL usage in PBE mode
    - c) BLPE/PBE: MCGC3 = 0x44 (%01000100)
      - PLLS (bit 6) set to 1, selects the PLL. In BLPE mode, changing this bit only prepares the MCG for PLL usage in PBE mode
      - VDIV (bits 3-0) set to %0100, or multiply-by-16 because  $1 \text{ MHz reference} * 16 = 16 \text{ MHz}$ . In BLPE mode, the configuration of the VDIV bits does not matter because the PLL is disabled. Changing them only sets up the multiply value for PLL usage in PBE mode
    - d) BLPE: If transitioning through BLPE mode, clear LP (bit 3) in MCGC2 to 0 here to switch to PBE mode
    - e) PBE: Loop until PLLST (bit 5) in MCGSC is set, indicating that the current source for the PLLS clock is the PLL
    - f) PBE: Then loop until LOCK (bit 6) in MCGSC is set, indicating that the PLL has acquired lock
  3. Last, PBE mode transitions into PEE mode:
    - a) MCGC1 = 0x10 (%00010000)
      - CLKS (bits 7 and 6) in MCGSC1 set to %00 in order to select the output of the PLL as the system clock source
      - Loop until CLKST (bits 3 and 2) in MCGSC are %11, indicating that the PLL output is selected to feed MCGOUT in the current clock mode
    - b) Now, With an RDIV of divide-by-4, a BDIV of divide-by-1, and a VDIV of multiply-by-16,  $\text{MCGOUT} = [(4 \text{ MHz} / 4) * 16] / 1 = 16 \text{ MHz}$ , and the bus frequency is  $\text{MCGOUT} / 2$ , or 8 MHz

### 13.3.1 RTC Status and Control Register (RTCSC)

RTCSC contains the real-time interrupt status flag (RTIF), the clock select bits (RTCLKS), the real-time interrupt enable bit (RTIE), and the prescaler select bits (RTCPS).



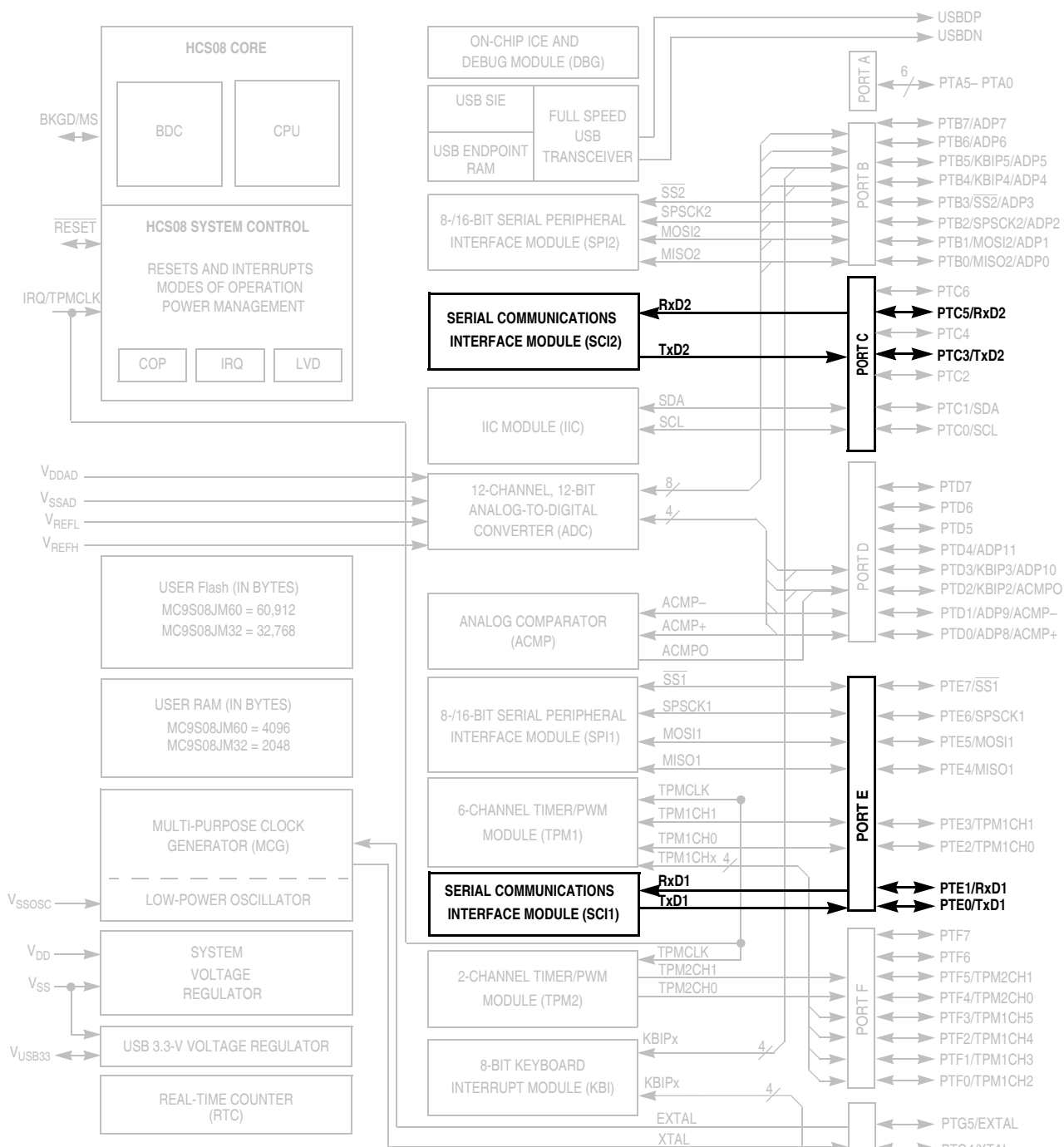
**Figure 13-3. RTC Status and Control Register (RTCSC)**

**Table 13-2. RTCSC Field Descriptions**

Field	Description
7 RTIF	Real-Time Interrupt Flag This status bit indicates the RTC counter register reached the value in the RTC modulo register. Writing a logic 0 has no effect. Writing a logic 1 clears the bit and the real-time interrupt request. Reset clears RTIF. 0 RTC counter has not reached the value in the RTC modulo register. 1 RTC counter has reached the value in the RTC modulo register.
6–5 RTCLKS	Real-Time Clock Source Select. These two read/write bits select the clock source input to the RTC prescaler. Changing the clock source clears the prescaler and RTCCNT counters. When selecting a clock source, ensure that the clock source is properly enabled (if applicable) to ensure correct operation of the RTC. Reset clears RTCLKS. 00 Real-time clock source is the 1-kHz low power oscillator (LPO) 01 Real-time clock source is the external clock (ERCLK) 1x Real-time clock source is the internal clock (IRCLK)
4 RTIE	Real-Time Interrupt Enable. This read/write bit enables real-time interrupts. If RTIE is set, then an interrupt is generated when RTIF is set. Reset clears RTIE. 0 Real-time interrupt requests are disabled. Use software polling. 1 Real-time interrupt requests are enabled.
3–0 RTCPS	Real-Time Clock Prescaler Select. These four read/write bits select binary-based or decimal-based divide-by values for the clock source. See <a href="#">Table 13-3</a> . Changing the prescaler value clears the prescaler and RTCCNT counters. Reset clears RTCPS.

**Table 13-3. RTC Prescaler Divide-by values**

RTCLKS[0]	RTCPS															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
<b>0</b>	Off	2 <sup>3</sup>	2 <sup>5</sup>	2 <sup>6</sup>	2 <sup>7</sup>	2 <sup>8</sup>	2 <sup>9</sup>	2 <sup>10</sup>	1	2	2 <sup>2</sup>	10	2 <sup>4</sup>	10 <sup>2</sup>	5x10 <sup>2</sup>	10 <sup>3</sup>
<b>1</b>	Off	2 <sup>10</sup>	2 <sup>11</sup>	2 <sup>12</sup>	2 <sup>13</sup>	2 <sup>14</sup>	2 <sup>15</sup>	2 <sup>16</sup>	10 <sup>3</sup>	2x10 <sup>3</sup>	5x10 <sup>3</sup>	10 <sup>4</sup>	2x10 <sup>4</sup>	5x10 <sup>4</sup>	10 <sup>5</sup>	2x10 <sup>5</sup>



NOTES:

1. Port pins are software configurable with pullup device if input port.
2. Pin contains software configurable pullup/pull-down device if IRQ is enabled (IRQPE = 1). Pull-down is enabled if rising edge detect is selected (IRQEDG = 1)
3. IRQ does not have a clamp diode to  $V_{DD}$ . IRQ must not be driven above  $V_{DD}$ .
4. Pin contains integrated pullup device.
5. When pin functions as KBI (KBIPEn = 1) and associated pin is configured to enable the pullup device, KBEDGn can be used to reconfigure the pullup as a pull-down device.

Figure 14-1. MC9S08JM60 Series Block Diagram Highlighting the SCI Blocks and Pins

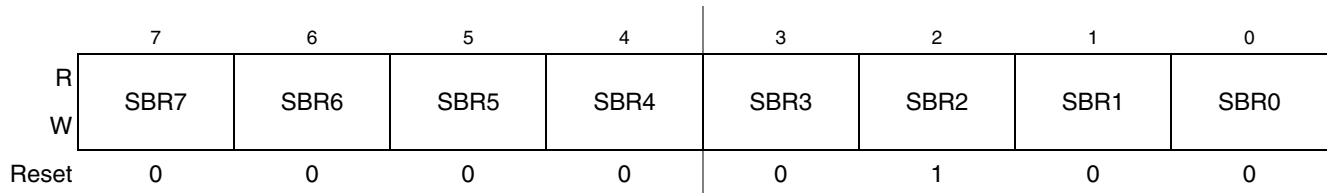


Figure 14-5. SCI Baud Rate Register (SCIxBDL)

Table 14-2. 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×BR). See also BR bits in <a href="#">Table 14-1</a> .

### 14.2.2 SCI Control Register 1 (SCIxC1)

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

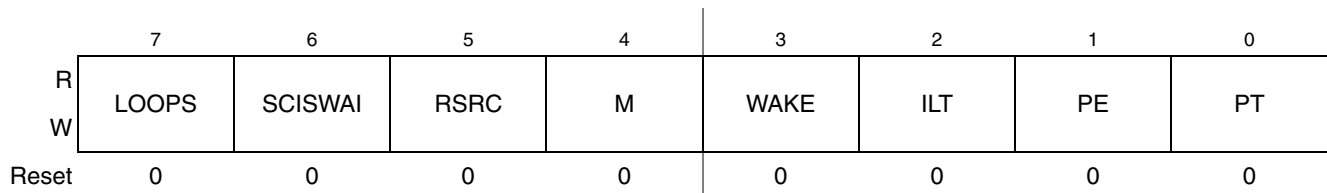


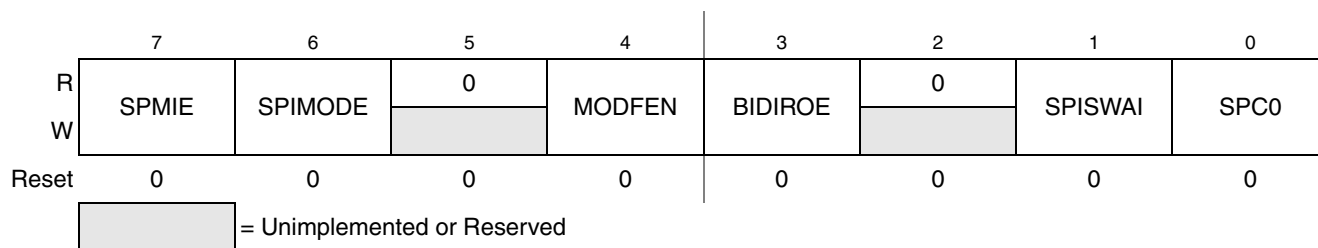
Figure 14-6. SCI Control Register 1 (SCIxC1)

Table 14-3. 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 <a href="#">RSRC</a> 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.

**Table 14-5. 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>


**Figure 15-6. SPI Control Register 2 (SPIx2)**
**Table 15-3. SPIx2 Register Field Descriptions**

Field	Description
7 SPMIE	<b>SPI Match Interrupt Enable</b> — This is the interrupt enable for the SPI receive data buffer hardware match (SPMF) function. 0 Interrupts from SPMF inhibited (use polling). 1 When SPMF = 1, requests a hardware interrupt.
6 SPIMODE	<b>SPI 8- or 16-bit Mode</b> — This bit allows the user to select either an 8-bit or 16-bit SPI data transmission length. In master mode, a change of this bit will abort a transmission in progress, force the SPI system into idle state, and reset all status bits in the SPIxS register. Refer to section <a href="#">Section 15.4.4, “Data Transmission Length,”</a> for details. 0 8-bit SPI shift register, match register, and buffers. 1 16-bit SPI shift register, match register, and buffers.
4 MODFEN	<b>Master Mode-Fault Function Enable</b> — 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 <a href="#">Table 15-2</a> for 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	<b>Bidirectional Mode Output Enable</b> — 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	<b>SPI Stop in Wait Mode</b> — This bit is used for power conservation while in wait. 0 SPI clocks continue to operate in wait mode 1 SPI clocks stop when the MCU enters wait mode
0 SPC0	<b>SPI Pin Control 0</b> — This bit enables bidirectional pin configurations as shown in <a href="#">Table 15-4</a> . 0 SPI uses separate pins for data input and data output. 1 SPI configured for single-wire bidirectional operation.

In the special case where the SPI is in master mode and MODFEN bit is cleared, the  $\overline{SS}$  pin is not used by the SPI. In this special case, the mode fault error function is inhibited and MODF remains cleared. In case the SPI system is configured as a slave, the  $\overline{SS}$  pin is a dedicated input pin. Mode fault error doesn't occur in slave mode.

If a mode fault error occurs the SPI is switched to slave mode, with the exception that the slave output buffer is disabled. So SPSCCK, MISO and MOSI pins are forced to be high impedance inputs to avoid any possibility of conflict with another output driver. A transmission in progress is aborted and the SPI is forced into idle state.

If the mode fault error occurs in the bidirectional mode for a SPI system configured in master mode, output enable of the MOMI (MOSI in bidirectional mode) is cleared if it was set. No mode fault error occurs in the bidirectional mode for the SPI system configured in slave mode.

The mode fault flag is cleared automatically by a read of the SPI Status Register (with MODF set) followed by a write to SPI Control Register 1. If the mode fault flag is cleared, the SPI becomes a normal master or slave again.

## 15.4.9 Low Power Mode Options

### 15.4.9.1 SPI in Run Mode

In run mode with the SPI system enable (SPE) bit in the SPI control register clear, the SPI system is in a low-power, disabled state. SPI registers can still be accessed, but clocks to the core of this module are disabled.

### 15.4.9.2 SPI in Wait Mode

SPI operation in wait mode depends upon the state of the SPISWAI bit in SPI Control Register 2.

- If SPISWAI is clear, the SPI operates normally when the CPU is in wait mode
- If SPISWAI is set, SPI clock generation ceases and the SPI module enters a power conservation state when the CPU is in wait mode.
  - If SPISWAI is set and the SPI is configured for master, any transmission and reception in progress stops at wait mode entry. The transmission and reception resumes when the SPI exits wait mode.
  - If SPISWAI is set and the SPI is configured as a slave, any transmission and reception in progress continues if the SPSCCK continues to be driven from the master. This keeps the slave synchronized to the master and the SPSCCK.

If the master transmits data while the slave is in wait mode, the slave will continue to send out data consistent with the operation mode at the start of wait mode (i.e., if the slave is currently sending its SPIxDH:SPIxDL to the master, it will continue to send the same byte. Otherwise, if the slave is currently sending the last data received byte from the master, it will continue to send each previously receive data from the master byte).



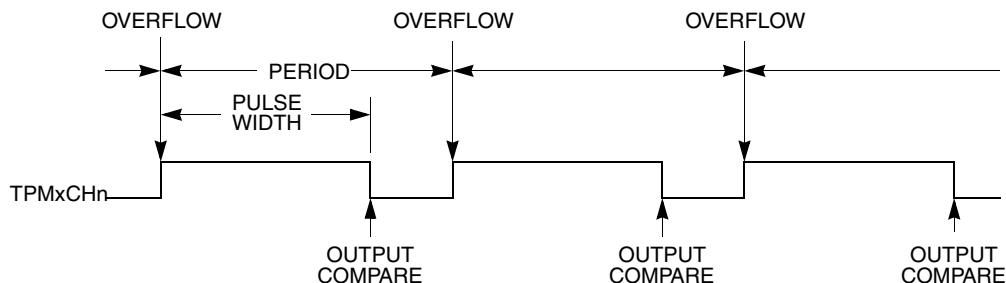
- Edge-aligned PWM mode  
The value of a 16-bit modulo register plus 1 sets the period of the PWM output signal. The channel value register sets the duty cycle of the PWM output signal. The user may also choose the polarity of the PWM output signal. Interrupts are available at the end of the period and at the duty-cycle transition point. This type of PWM signal is called edge-aligned because the leading edges of all PWM signals are aligned with the beginning of the period, which is the same for all channels within a TPM.
- Center-aligned PWM mode  
Twice the value of a 16-bit modulo register sets the period of the PWM output, and the channel-value register sets the half-duty-cycle duration. The timer counter counts up until it reaches the modulo value and then counts down until it reaches zero. As the count matches the channel value register while counting down, the PWM output becomes active. When the count matches the channel value register while counting up, the PWM output becomes inactive. This type of PWM signal is called center-aligned because the centers of the active duty cycle periods for all channels are aligned with a count value of zero. This type of PWM is required for types of motors used in small appliances.

This is a high-level description only. Detailed descriptions of operating modes are in later sections.

### 16.1.3 Block Diagram

The TPM uses one input/output (I/O) pin per channel, TPMxCHn (timer channel n) where n is the channel number (1-8). The TPM shares its I/O pins with general purpose I/O port pins (refer to I/O pin descriptions in full-chip specification for the specific chip implementation).

Figure 16-2 shows the TPM structure. The central component of the TPM is the 16-bit counter that can operate as a free-running counter or a modulo up/down counter. The TPM counter (when operating in normal up-counting mode) provides the timing reference for the input capture, output compare, and edge-aligned PWM functions. The timer counter modulo registers, TPMxMODH:TPMxMODL, control the modulo value of the counter (the values 0x0000 or 0xFFFF effectively make the counter free running). Software can read the counter value at any time without affecting the counting sequence. Any write to either half of the TPMxCNT counter resets the counter, regardless of the data value written.



**Figure 16-15. PWM Period and Pulse Width (ELSnA=0)**

When the channel value register is set to 0x0000, the duty cycle is 0%. 100% duty cycle can be achieved by setting the timer-channel register (TPMxCnVH:TPMxCnVL) to a value greater than the modulus setting. This implies that the modulus setting must be less than 0xFFFF in order to get 100% duty cycle.

Because the TPM may be used in an 8-bit MCU, the settings in the timer channel registers are buffered to ensure coherent 16-bit updates and to avoid unexpected PWM pulse widths. Writes to any of the registers TPMxCnVH and TPMxCnVL, actually write to buffer registers. In edge-aligned PWM mode, values are transferred to the corresponding timer-channel registers according to the value of CLKSb:CLKSA bits, so:

- If (CLKSB:CLKSA = 0:0), the registers are updated when the second byte is written
- If (CLKSB:CLKSA not = 0:0), the registers are updated after the both bytes were written, and the TPM counter changes from (TPMxMODH:TPMxMODL - 1) to (TPMxMODH:TPMxMODL). If the TPM counter is a free-running counter then the update is made when the TPM counter changes from 0xFFFE to 0xFFFF.

#### 16.4.2.4 Center-Aligned PWM Mode

This type of PWM output uses the up/down counting mode of the timer counter (CPWMS=1). The output compare value in TPMxCnVH:TPMxCnVL determines the pulse width (duty cycle) of the PWM signal while the period is determined by the value in TPMxMODH:TPMxMODL. TPMxMODH:TPMxMODL should be kept in the range of 0x0001 to 0x7FFF because values outside this range can produce ambiguous results. ELSnA will determine the polarity of the CPWM output.

$$\text{pulse width} = 2 \times (\text{TPMxCnVH:TPMxCnVL})$$

$$\text{period} = 2 \times (\text{TPMxMODH:TPMxMODL}); \text{TPMxMODH:TPMxMODL}=0\text{x}0001\text{-}0\text{x}7\text{FFF}$$

If the channel-value register TPMxCnVH:TPMxCnVL is zero or negative (bit 15 set), the duty cycle will be 0%. If TPMxCnVH:TPMxCnVL is a positive value (bit 15 clear) and is greater than the (non-zero) modulus setting, the duty cycle will be 100% because the duty cycle compare will never occur. This implies the usable range of periods set by the modulus register is 0x0001 through 0x7FFE (0x7FFF if you do not need to generate 100% duty cycle). This is not a significant limitation. The resulting period would be much longer than required for normal applications.

TPMxMODH:TPMxMODL=0x0000 is a special case that should not be used with center-aligned PWM mode. When CPWMS=0, this case corresponds to the counter running free from 0x0000 through 0xFFFF, but when CPWMS=1 the counter needs a valid match to the modulus register somewhere other than at 0x0000 in order to change directions from up-counting to down-counting.

### 17.3.11 Address Register (ADDR)

The ADDR register contains the unique 7-bit address the device will be recognized as through USB. The register is reset to 0x00 after the reset input has gone active or the USB module has decoded USB reset signaling. That will initialize the address register to decode address 0x00 as required by the USB specification. Firmware will change the value when it processes a SET\_ADDRESS request.

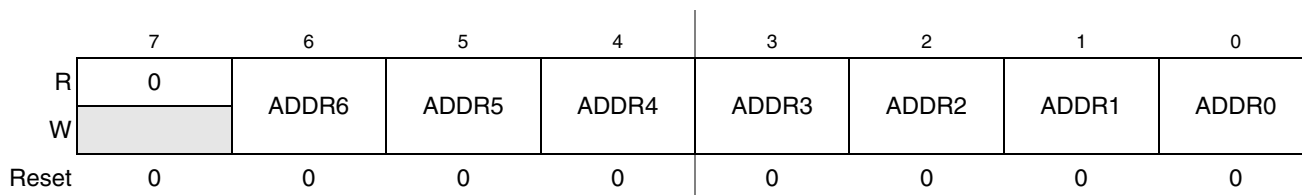


Figure 17-14. Address Register (ADDR)

Table 17-15. ADDR Field Descriptions

Field	Description
6–0 ADDR[6:0]	<b>USB Address</b> — This 7-bit value defines the USB address that the USB module will decode

### 17.3.12 Frame Number Register (FRMNUML, FRMNUMH)

The frame number registers contains the 11-bit frame number. The frame number registers require two 8-bit registers to implement. The low order byte is contained in FRMNUML, and the high order byte is contained in FRMNUMH. These registers are updated with the current frame number whenever a SOF TOKEN is received.

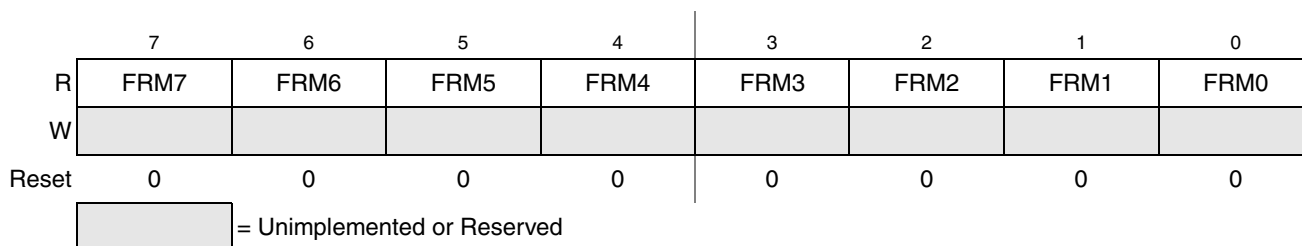


Figure 17-15. Frame Number Register Low (FRMNUML)

Table 17-16. FRMNUML Field Descriptions

Field	Description
7–0 FRM[7:0]	<b>Frame Number</b> — These bits represent the low order bits of the 11 bit frame number.

Table A-6. DC Characteristics (continued)

Num	C	Parameter	Symbol	Min	Typical <sup>1</sup>	Max.	Unit
5	P	Output low current — Max. total $I_{OL}$ for all ports 5 V 3 V	$I_{OLT}$	— —	— —	100 60	mA
6	C	Input high voltage; all digital inputs 5 V 3 V	$V_{IH}$	$0.65 \times V_{DD}$ $0.70 \times V_{DD}$	—	—	V
7	C	Input low voltage; all digital inputs	$V_{IL}$	—	—	$0.35 \times V_{DD}$	
8	C	Input hysteresis; all digital inputs	$V_{hys}$	$0.06 \times V_{DD}$			mV
9	C	Input leakage current (per pin); input only pins	$ I_{In} $	—	0.1	1	$\mu A$
10	P	Hi-Z (off-state) leakage current (per pin)	$ I_{OZ} $	—	0.1	1	$\mu A$
11	P	Internal pullup resistors <sup>3</sup>	$R_{PU}$	20	45	65	k $\Omega$
12	P	Internal pulldown resistors <sup>4</sup>	$R_{PD}$	20	45	65	k $\Omega$
13	T	Internal pullup resistor to USBDP (to $V_{USB33}$ ) Idle Transmit	$R_{PUPD}$	900 1425	1300 2400	1575 3090	k $\Omega$
14	D	DC injection current <sup>5 6 7 8</sup> (single pin limit) $V_{IN} > V_{DD}$ $V_{IN} < V_{SS}$	$I_{IC}$	0 0	— —	2 -0.2	mA
		DC injection current (Total MCU limit, includes sum of all stressed pins) $V_{IN} > V_{DD}$ $V_{IN} < V_{SS}$		0 0	— —	25 -5	
15	D	Input capacitance; all non-supply pins	$C_{In}$	—	—	8	pF
16	D	RAM retention voltage	$V_{RAM}$	—	0.6	1.0	V
17	D	POR re-arm voltage	$V_{POR}$	0.9	1.4	2.0	V
18	D	POR re-arm time	$t_{POR}$	10	—	—	$\mu s$

## A.10 External Oscillator (XOSC) Characteristics

**Table A-11. Oscillator Electrical Specifications (Temperature Range = -40 to 85°C Ambient)**

Num	C	Rating	Symbol	Min	Typ <sup>1</sup>	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 <sup>2</sup>	$f_{hi-ll}$	1	—	5	MHz
		High range (RANGE = 1) PEE or PBE mode <sup>3</sup>	$f_{hi-pll}$	1	—	16	MHz
		High range (RANGE = 1, HGO = 1) BLPE mode	$f_{hi-hgo}$	1	—	16	MHz
High range (RANGE = 1, HGO = 0) BLPE mode	$f_{hi-lp}$	1	—	8	MHz		
2	—	Load capacitors	$C_1, C_2$	See crystal or resonator manufacturer's recommendation.			
3	—	Feedback resistor	$R_F$		10		$M\Omega$
		Low range (32 kHz to 38.4 kHz)			1		$M\Omega$
4	—	Series resistor	$R_S$		0		$k\Omega$
		Low range, low gain (RANGE = 0, HGO = 0)		—	100	—	
		Low range, high gain (RANGE = 0, HGO = 1)		—	0	—	
		High range, low gain (RANGE = 1, HGO = 0)		—	0	—	
		High range, high gain (RANGE = 1, HGO = 1)		—	0	—	
≥ 8 MHz	—	0	0				
4 MHz	—	0	10				
1 MHz	—	0	20				
5	T	Crystal start-up time <sup>4</sup>					
		Low range, low gain (RANGE = 0, HGO = 0)	$t_{CSTL-LP}$	—	200	—	ms
		Low range, high gain (RANGE = 0, HGO = 1)	$t_{CSTL-HGO}$	—	400	—	
		High range, low gain (RANGE = 1, HGO = 0) <sup>5</sup>	$t_{CSTH-LP}$	—	5	—	
High range, high gain (RANGE = 1, HGO = 1) <sup>5</sup>	$t_{CSTH-HGO}$	—	15	—			
6	T	Square wave input clock frequency (EREFS = 0, ERCLKEN = 1)	$f_{extal}$	0.03125	—	5	MHz
		FEE or FBE mode <sup>2</sup>		1	—	16	MHz
		PEE or PBE mode <sup>3</sup>		0	—	40	MHz
BLPE mode							

<sup>1</sup> Typical data was characterized at 3.0 V, 25°C or is recommended value.

<sup>2</sup> When MCG is configured for FEE or FBE mode, input clock source must be divided using RDIV to within the range of 31.25 kHz to 39.0625 kHz.

<sup>3</sup> When MCG is configured for PEE or PBE mode, input clock source must be divided using RDIV to within the range of 1 MHz to 2 MHz.

<sup>4</sup> This parameter is characterized and not tested on each device. Proper PC board layout procedures must be followed to achieve specifications.

<sup>5</sup> 4 MHz crystal

