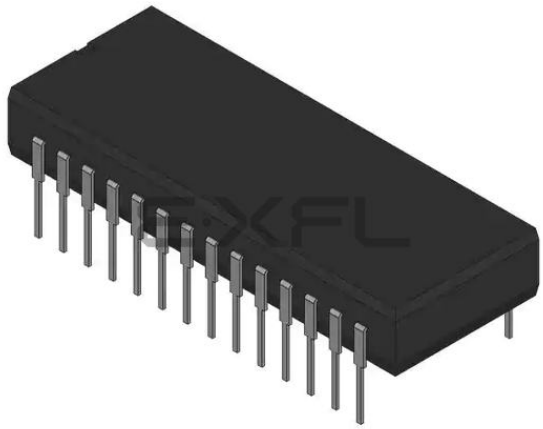


Welcome to E-XFL.COM

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	8MHz
Connectivity	SCI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	23
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s08rd16cdwe

Chapter 9
Keyboard Interrupt (S08KBIV1)

Chapter 10
Timer/PWM Module (S08TPMV1)

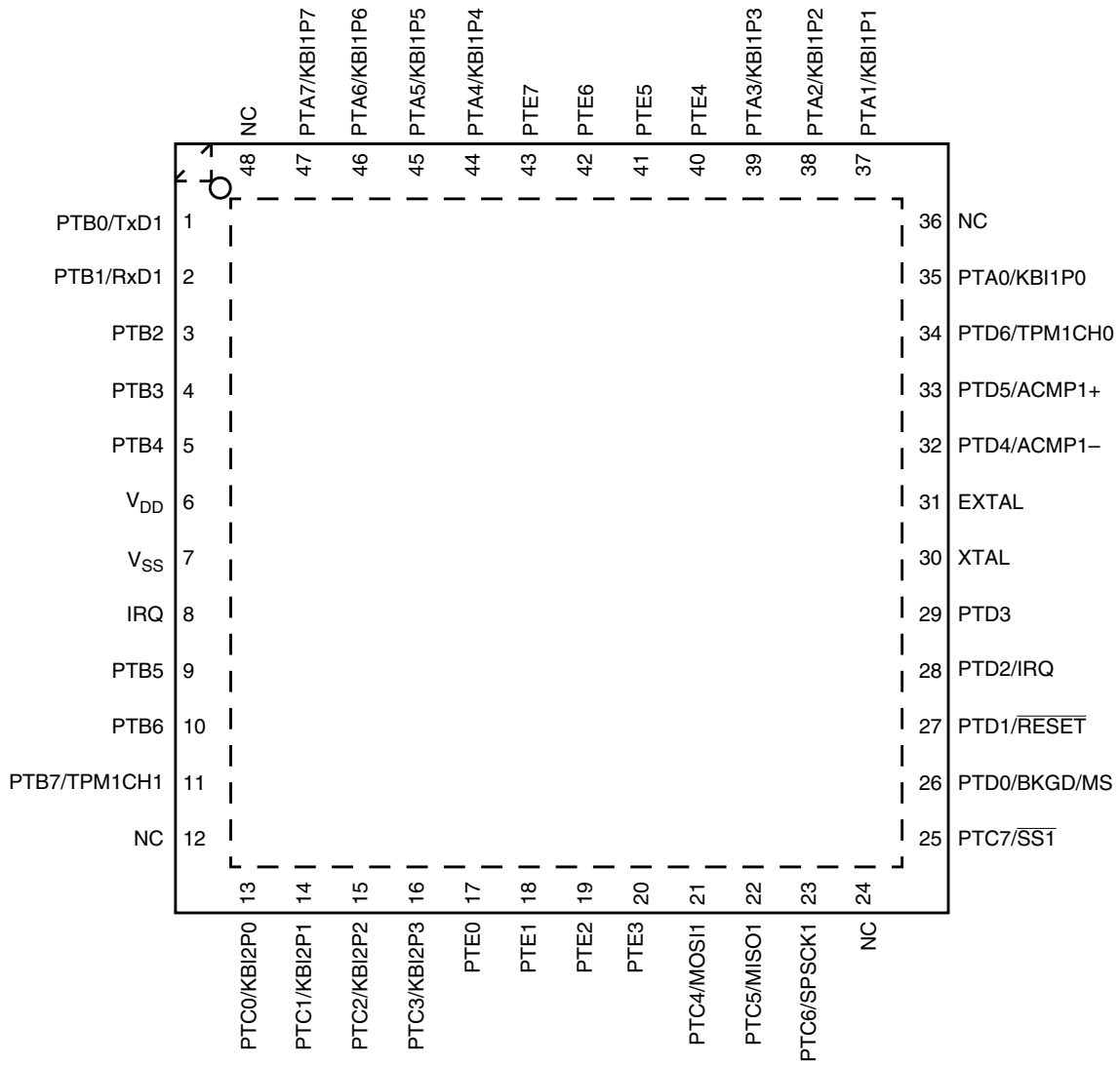


Figure 2-4. MC9S08RC/RD/RE/RG in 48-Pin QFN Package

2.3 Recommended System Connections

Figure 2-5 shows pin connections that are common to almost all MC9S08RC/RD/RE/RG application systems. A more detailed discussion of system connections follows.

Table 4-2. High-Page Register Summary (continued)

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
$1809	SPMSC1	LVDF	LVDACK	LVDIE	SAFE	LVDRE	—	—	—
$180A	SPMSC2	LVWF	LVWACK	0	0	PPDF	PPDACK	PDC	PPDC
$180B–$180F	Reserved	— —	— —	— —	— —	— —	— —	— —	— —
$1810	DBGCAH	Bit 15	14	13	12	11	10	9	Bit 8
$1811	DBGCAL	Bit 7	6	5	4	3	2	1	Bit 0
$1812	DBGCBH	Bit 15	14	13	12	11	10	9	Bit 8
$1813	DBGCBL	Bit 7	6	5	4	3	2	1	Bit 0
$1814	DBGFH	Bit 15	14	13	12	11	10	9	Bit 8
$1815	DBGFL	Bit 7	6	5	4	3	2	1	Bit 0
$1816	DBGC	DBGEN	ARM	TAG	BRKEN	RWA	RWAEN	RWB	RWBEN
$1817	DBGT	TRGSEL	BEGIN	0	0	TRG3	TRG2	TRG1	TRG0
$1818	DBGS	AF	BF	ARMF	0	CNT3	CNT2	CNT1	CNT0
$1819–$181F	Reserved	— —	— —	— —	— —	— —	— —	— —	— —
$1820	FCDIV	DIVLD	PRDIV8	DIV5	DIV4	DIV3	DIV2	DIV1	DIV0
$1821	FOPT	KEYEN	FNORED	0	0	0	0	SEC01	SEC00
$1822	Reserved	—	—	—	—	—	—	—	—
$1823	FCNFG	0	0	KEYACC	0	0	0	0	0
$1824	FPROT	FPOPEN	FPDIS	FPS2	FPS1	FPS0	0	0	0
$1825	FSTAT	FCBEF	FCCF	FPVIOL	FACCERR	0	FBLANK	0	0
$1826	FCMD	FCMD7	FCMD6	FCMD5	FCMD4	FCMD3	FCMD2	FCMD1	FCMD0
$1827–$182B	Reserved	— —	— —	— —	— —	— —	— —	— —	— —

1. The ILAD bit is only present on 16K and 8K versions of the devices.

Nonvolatile FLASH registers, shown in Table 4-3, are located in the FLASH memory. These registers include an 8-byte backdoor key that optionally can be used to gain access to secure memory resources. During reset events, the contents of NVPROT and NVOPT in the nonvolatile register area of the FLASH memory are transferred into corresponding FPROT and FOPT working registers in the high-page registers to control security and block protection options.

Table 4-3. Nonvolatile Register Summary

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
$FFB0–$FFB7	NVBACKKEY	8-Byte Comparison Key							
$FFB8–$FFBC	Reserved	— —	— —	— —	— —	— —	— —	— —	— —
$FFBD	NVPROT	FPOPEN	FPDIS	FPS2	FPS1	FPS0	0	0	0
$FFBE	Reserved	—	—	—	—	—	—	—	—
$FFBF	NVOPT	KEYEN	FNORED	0	0	0	0	SEC01	SEC00

- Writing to a FLASH address before the internal FLASH clock frequency has been set by writing to the FCDIV register
- Writing to a FLASH address while FCBEF is not set (A new command cannot be started until the command buffer is empty.)
- Writing a second time to a FLASH address before launching the previous command (There is only one write to FLASH for every command.)
- Writing a second time to FCMD before launching the previous command (There is only one write to FCMD for every command.)
- Writing to any FLASH control register other than FCMD after writing to a FLASH address
- Writing any command code other than the five allowed codes ($05, $20, $25, $40, or $41) to FCMD
- Accessing (read or write) any FLASH control register other than the write to FSTAT (to clear FCBEF and launch the command) after writing the command to FCMD
- The MCU enters stop mode while a program or erase command is in progress (The command is aborted.)
- Writing the byte program, burst program, or page erase command code ($20, $25, or $40) with a background debug command while the MCU is secured (The background debug controller can only do blank check and mass erase commands when the MCU is secure.)
- Writing 0 to FCBEF to cancel a partial command

4.4.6 FLASH Block Protection

Block protection prevents program or erase changes for FLASH memory locations in a designated address range. Mass erase is disabled when any block of FLASH is protected. The MC9S08RC/RD/RE/RG allows a block of memory at the end of FLASH, and/or the entire FLASH memory to be block protected. A disable control bit and a 3-bit control field, for each of the blocks, allows the user to independently set the size of these blocks. A separate control bit allows block protection of the entire FLASH memory array. All seven of these control bits are located in the FPROT register (see Section 4.6.4, “FLASH Protection Register (FPROT and NVPROT)“).

At reset, the high-page register (FPROT) is loaded with the contents of the NVPROT location that is in the nonvolatile register block of the FLASH memory. The value in FPROT cannot be changed directly from application software so a runaway program cannot alter the block protection settings. If the last 512 bytes of FLASH (which includes the NVPROT register) is protected, the application program cannot alter the block protection settings (intentionally or unintentionally). The FPROT control bits can be written by background debug commands to allow a way to erase a protected FLASH memory.

One use for block protection is to block protect an area of FLASH memory for a bootloader program. This bootloader program then can be used to erase the rest of the FLASH memory and reprogram it. Because the bootloader is protected, it remains intact even if MCU power is lost during an erase and reprogram operation.

5.7 Real-Time Interrupt (RTI)

The real-time interrupt function can be used to generate periodic interrupts based on a multiple of the source clock's period. The RTI has two source clock choices, the external clock input or the RTI's own internal clock. The RTI can be used in run, wait, stop2, and stop3 modes. It is not available in stop1 mode.

In run and wait modes, only the external clock can be used as the RTI's clock source. In stop2 mode, only the internal RTI clock can be used. In stop3, either the external clock or internal RTI clock can be used. When using the external oscillator in stop3 mode, it must be enabled in stop (OSCSTEN = 1) and configured for low bandwidth operation (RANGE = 0).

The SRTISC register includes a read-only status flag, a write-only acknowledge bit, and a 3-bit control value (RTIS2:RTIS1:RTIS0) used to select one of seven RTI periods. The RTI has a local interrupt enable, RTIE, to allow masking of the real-time interrupt. The module can be disabled by writing 0:0:0 to RTIS2:RTIS1:RTIS0 in which case the clock source input is disabled and no interrupts will be generated. See Section 5.8.6, "System Real-Time Interrupt Status and Control Register (SRTISC)," for detailed information about this register.

5.8 Reset, Interrupt, and System Control Registers and Control Bits

One 8-bit register in the direct page register space and five 8-bit registers in the high-page register space are related to reset and interrupt systems.

Refer to the direct-page register summary in the Memory chapter of this data sheet for the absolute address assignments for all registers. This section refers to registers and control bits only by their names. A Freescale-provided equate or header file is used to translate these names into the appropriate absolute addresses.

Some control bits in the SOPT and SPMSC2 registers are related to modes of operation. Although brief descriptions of these bits are provided here, the related functions are discussed in greater detail in Chapter 3, "Modes of Operation."

5.8.1 Interrupt Pin Request Status and Control Register (IRQSC)

This direct page register includes two unimplemented bits that always read 0, four read/write bits, one read-only status bit, and one write-only bit. These bits are used to configure the IRQ function, report status, and acknowledge IRQ events.

5.8.4 System Options Register (SOPT)

This register may be read at any time. Bits 3 and 2 are unimplemented and always read 0. This is a write-once register so only the first write after reset is honored. Any subsequent attempt to write to SOPT (intentionally or unintentionally) is ignored to avoid accidental changes to these sensitive settings. SOPT must be written during the user's reset initialization program to set the desired controls even if the desired settings are the same as the reset settings.

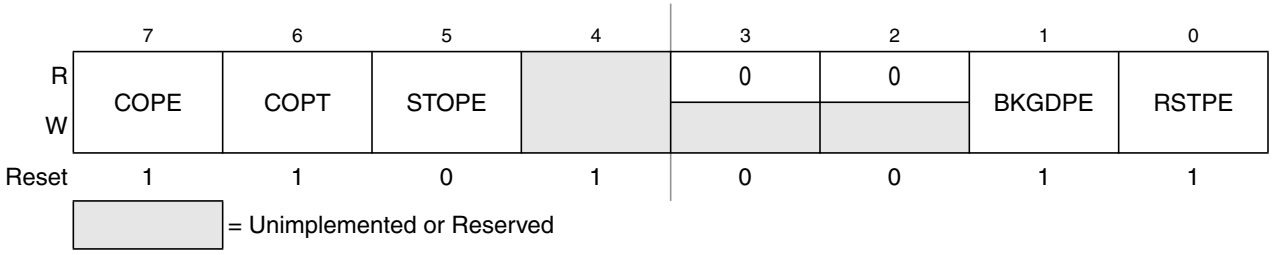


Figure 5-5. System Options Register (SOPT)

Table 5-5. SOPT Field Descriptions

Field	Description
7 COPE	**COP Watchdog Enable** — This write-once bit defaults to 1 after reset. 0 COP watchdog timer disabled. 1 COP watchdog timer enabled (force reset on timeout).
6 COPT	**COP Watchdog Timeout** — This write-once bit defaults to 1 after reset. 0 Short timeout period selected (2^{18} cycles of BUSCLK). 1 Long timeout period selected (2^{20} cycles of BUSCLK).
5 STOPE	**Stop Mode Enable** — This write-once bit defaults to 0 after reset, which disables stop mode. If stop mode is disabled and a user program attempts to execute a STOP instruction, an illegal opcode reset is forced. 0 Stop mode disabled. 1 Stop mode enabled.
1 BKGDPE	**Background Debug Mode Pin Enable** — The BKGDPE bit enables the PTD0/BKGD/MS pin to function as BKGD/MS. When the bit is clear, the pin will function as PTD0, which is an output only general purpose I/O. This pin always defaults to BKGD/MS function after any reset. 0 BKGD pin disabled. 1 BKGD pin enabled.
0 RSTPE	**$\overline{\text{RESET}}$ Pin Enable** — The RSTPE bit enables the PTD1/$\overline{\text{RESET}}$ pin to function as $\overline{\text{RESET}}$. When the bit is clear, the pin will function as PTD1, which is an output only general purpose I/O. This pin always defaults to $\overline{\text{RESET}}$ function after any reset. 0 $\overline{\text{RESET}}$ pin disabled. 1 $\overline{\text{RESET}}$ pin enabled.

6.6.1 Port A Registers (PTAD, PTAPE, and PTADD)

Port A pins used as general-purpose I/O pins are controlled by the port A data (PTAD), data direction (PTADD), and pullup enable (PTAPE) registers.

	7	6	5	4	3	2	1	0
R W	PTAD7	PTAD6	PTAD5	PTAD4	PTAD3	PTAD2	PTAD1	PTAD0
Reset	0	0	0	0	0	0	0	0

Figure 6-6. Port A Data Register (PTAD)

Table 6-1. PTAD Field Descriptions

Field	Description
7:0 PTAD[7:0]	**Port A Data Register Bits** — For port A pins that are inputs, reads of this register return the logic level on the pin. For port A pins that are configured as outputs, reads of this register return the last value written to this register. Writes are latched into all bits of this register. For port A pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTAD 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.

	7	6	5	4	3	2	1	0
R W	PTAPE7	PTAPE6	PTAPE5	PTAPE4	PTAPE3	PTAPE2	PTAPE1	PTAPE0
Reset	0	0	0	0	0	0	0	0

Figure 6-7. Pullup Enable for Port A (PTAPE)

Table 6-2. PTAPE Field Descriptions

Field	Description
7:0 PTAPE[7:0]	**Pullup Enable for Port A Bits** — For port A pins that are inputs, these read/write control bits determine whether internal pullup devices are enabled provided the corresponding PTADDn is a logic 0. For port A pins that are configured as outputs, these bits are ignored and the internal pullup devices are disabled. When any of bits 7 through 4 of port A are enabled as KBI inputs and are configured to detect rising edges/high levels, the pullup enable bits enable pulldown rather than pullup devices. 0 Internal pullup device disabled. 1 Internal pullup device enabled.

6.6.5 Port E Registers (PTED, PTEPE, and PTEDD)

Port E pins used as general-purpose I/O pins are controlled by the port E data (PTED), data direction (PTEDD), and pullup enable (PTEPE) registers.

	7	6	5	4	3	2	1	0
R W	PTED7	PTED6	PTED5	PTED4	PTED3	PTED2	PTED1	PTED0
Reset	0	0	0	0	0	0	0	0

Figure 6-18. Port E Data Register (PTED)

Table 6-13. PTED Field Descriptions

Field	Description
7:0 PTED[7:0]	**Port E Data Register Bits** — For port E pins that are inputs, reads return the logic level on the pin. For port E 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 E pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTED 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.

	7	6	5	4	3	2	1	0
R W	PTEPE7	PTEPE6	PTEPE5	PTEPE4	PTEPE3	PTEPE2	PTEPE1	PTEPE0
Reset	0	0	0	0	0	0	0	0

Figure 6-19. Pullup Enable for Port E (PTED)

Table 6-14. PTED Field Descriptions

Field	Description
7:0 PTED[7:0]	**Pullup Enable for Port E Bits** — For port E pins that are inputs, these read/write control bits determine whether internal pullup devices are enabled. For port E pins that are configured as outputs, these bits are ignored and the internal pullup devices are disabled. 0 Internal pullup device disabled. 1 Internal pullup device enabled.

	7	6	5	4	3	2	1	0
R W	PTEDD7	PTEDD6	PTEDD5	PTEDD4	PTEDD3	PTEDD2	PTEDD1	PTEDD0
Reset	0	0	0	0	0	0	0	0

Figure 6-20. Data Direction for Port E (PTEDD)

Table 6-15. PTEDD Field Descriptions

Field	Description
7:0 PTEDD[7:0]	**Data Direction for Port E Bits** — These read/write bits control the direction of port E pins and what is read for PTED reads. 0 Input (output driver disabled) and reads return the pin value. 1 Output driver enabled for port E bit n and PTED reads return the contents of PTEDn.

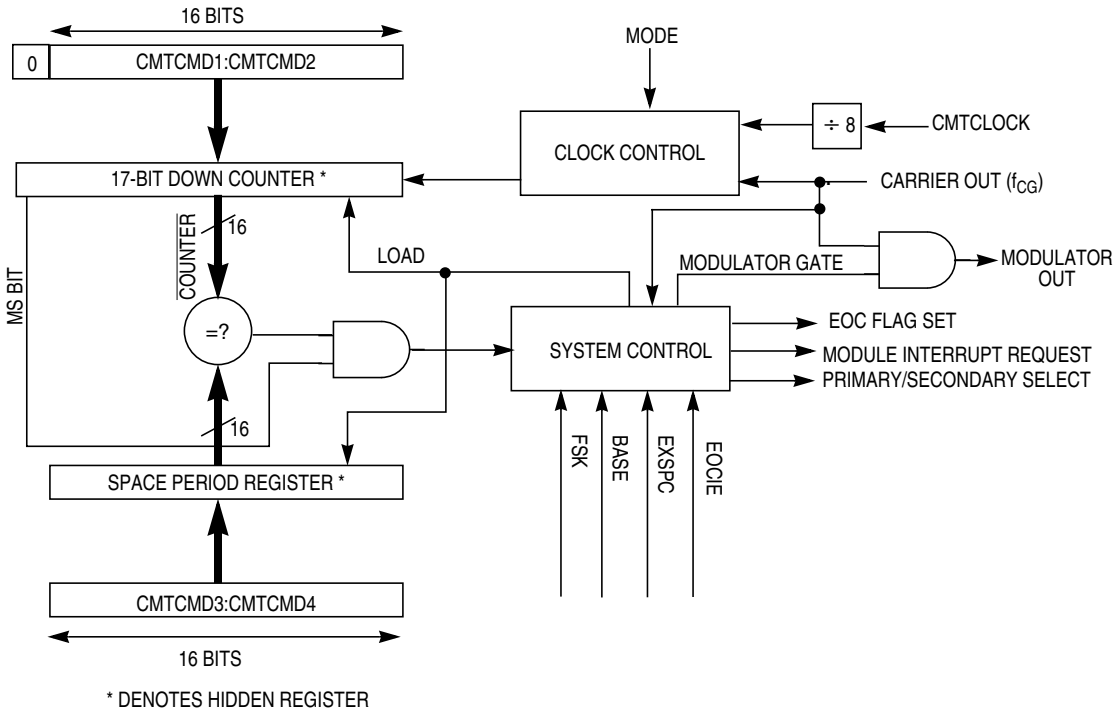


Figure 8-4. Modulator Block Diagram

8.5.2.1 Time Mode

When the modulator operates in time mode (MCGEN bit is set, BASE bit is clear, and FSK bit is clear), the modulation mark period consists of an integer number of CMTCLK ÷ 8 clock periods. The modulation space period consists of zero or an integer number of CMTCLK ÷ 8 clock periods. With an 8 MHz bus and CMTDIV1:CMTDIV0 = 00, the modulator resolution is 1 µs and has a maximum mark and space period of about 65.535 ms each. See Figure 8-5 for an example of the time mode and baseband mode outputs.

The mark and space time equations for time and baseband mode are:

$$t_{mark} = (CMTCMD1{:}CMTCMD2 + 1) \div (f_{CMTCLK} \div 8) \qquad \textit{Eqn. 8-5}$$

$$t_{space} = CMTCMD3{:}CMTCMD4 \div (f_{CMTCLK} \div 8) \qquad \textit{Eqn. 8-6}$$

where CMTCMD1:CMTCMD2 and CMTCMD3:CMTCMD4 are the decimal values of the concatenated registers.

NOTE

If the modulator is disabled while the t_{mark} time is less than the programmed carrier high time ($t_{mark} < CMTCGH1/f_{CMTCLK}$), the modulator can enter into an illegal state and end the curent cycle before the programmed value. Make sure to program t_{mark} greater than the carrier high time to avoid this illegal state.

	7	6	5	4	3	2	1	0
R W	SL7	SL6	SL5	SL4	SL3	SL2	SL1	SL0
Reset	u	u	u	u	u	u	u	u

u = Unaffected

Figure 8-11. Carrier Generator Data Register Low 2 (CMTCGL2)

Table 8-6. CMTCGL2 Field Descriptions

Field	Description
7:0 SL[7:0]	**Secondary Carrier Low Time Data Values** — When selected, these bits contain the number of input clocks required to generate the carrier high and low time periods. When operating in time mode (see Section 8.5.2.1, “Time Mode"), this register pair is never selected. When operating in FSK mode (see Section 8.5.2.3, “FSK Mode"), this register pair and the primary register pair are alternatively selected under control of the modulator. The secondary carrier high and low time values are unaffected out of reset. These bits must be written to nonzero values before the carrier generator is enabled when operating in FSK mode.

8.6.2 CMT Output Control Register (CMTOC)

This register is used to control the IRO output of the CMT module.

	7	6	5	4	3	2	1	0
R	IROL	CMTPOL	IROPEN	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 8-12. CMT Output Control Register (CMTOC)

Table 8-7. CMTOC Field Descriptions

Field	Description
7 IROL	**IRO Latch Control** — Reading IROL reads the state of the IRO latch. Writing IROL changes the state of the IRO pin when the MCGEN bit is clear in the CMTMSC register and the IROPEN bit is set.
6 CMTPOL	**CMT Output Polarity** — The CMTPOL bit controls the polarity of the IRO pin output of the CMT. 0 IRO pin is active low 1 IRO pin is active high
5 IROPEN	**IRO Pin Enable** — The IROPEN bit is used to enable and disable the IRO pin. When the pin is enabled, it is an output that drives out either the CMT transmitter output or the state of the IROL bit depending on whether the MCGEN bit in the CMTMSC register is set. Also, the state of the output is either inverted or not depending on the state of the CMTPOL bit. When the pin is disabled, it is in a high impedance state so it doesn’t draw any current. The pin is disabled during reset. 0 IRO pin disabled 1 IRO pin enabled as output

Table 12-5. SCI1S1 Register Field Descriptions (continued)

Field	Description
5 RDRF	**Receive Data Register Full Flag** — RDRF becomes set when a character transfers from the receive shifter into the receive data register (SCI1D). To clear RDRF, read SCI1S1 with RDRF = 1 and then read the SCI data register (SCI1D). 0 Receive data register empty. 1 Receive data register full.
4 IDLE	**Idle Line Flag** — 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. To clear IDLE, read SCI1S1 with IDLE = 1 and then read the SCI data register (SCI1D). 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. 0 No idle line detected. 1 Idle line was detected.
3 OR	**Receiver Overrun Flag** — 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 SCI1D yet. In this case, the new character (and all associated error information) is lost because there is no room to move it into SCI1D. To clear OR, read SCI1S1 with OR = 1 and then read the SCI data register (SCI1D). 0 No overrun. 1 Receive overrun (new SCI data lost).
2 NF	**Noise Flag** — 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 SCI1S1 and then read the SCI data register (SCI1D). 0 No noise detected. 1 Noise detected in the received character in SCI1D.
1 FE	**Framing Error Flag** — FE is set at the same time as RDRF when the receiver detects a logic 0 where the stop bit was expected. This suggests the receiver was not properly aligned to a character frame. To clear FE, read SCI1S1 with FE = 1 and then read the SCI data register (SCI1D). 0 No framing error detected. This does not guarantee the framing is correct. 1 Framing error.
0 PF	**Parity Error Flag** — PF is set at the same time as RDRF when parity is enabled (PE = 1) and the parity bit in the received character does not agree with the expected parity value. To clear PF, read SCI1S1 and then read the SCI data register (SCI1D). 0 No parity error. 1 Parity error.

Table 12-7. SCI1C3 Register Field Descriptions (continued)

Field	Description
3 ORIE	**Overrun Interrupt Enable** — This bit enables the overrun flag (OR) to generate hardware interrupt requests. 0 OR interrupts disabled (use polling). 1 Hardware interrupt requested when OR = 1.
2 NEIE	**Noise Error Interrupt Enable** — This bit enables the noise flag (NF) to generate hardware interrupt requests. 0 NF interrupts disabled (use polling). 1 Hardware interrupt requested when NF = 1.
1 FEIE	**Framing Error Interrupt Enable** — This bit enables the framing error flag (FE) to generate hardware interrupt requests. 0 FE interrupts disabled (use polling). 1 Hardware interrupt requested when FE = 1.
0 PEIE	**Parity Error Interrupt Enable** — This bit enables the parity error flag (PF) to generate hardware interrupt requests. 0 PF interrupts disabled (use polling). 1 Hardware interrupt requested when PF = 1.

12.2.7 SCI Data Register (SCI1D)

This register is actually two separate registers. Reads return the contents of the read-only receive data buffer and writes go to the write-only transmit data buffer. Reads and writes of this register are also involved in the automatic flag clearing mechanisms for the SCI status flags.

	7	6	5	4	3	2	1	0
R	R7	R6	R5	R4	R3	R2	R1	R0
W	T7	T6	T5	T4	T3	T2	T1	T0
Reset	0	0	0	0	0	0	0	0

Figure 12-10. SCI Data Register (SCI1D)

A-Only — Trigger when the address matches the value in comparator A

A OR B — Trigger when the address matches either the value in comparator A or the value in comparator B

A Then B — Trigger when the address matches the value in comparator B but only after the address for another cycle matched the value in comparator A. There can be any number of cycles after the A match and before the B match.

A AND B Data (Full Mode) — This is called a full mode because address, data, and R/W (optionally) must match within the same bus cycle to cause a trigger event. Comparator A checks address, the low byte of comparator B checks data, and R/W is checked against RWA if RWAEN = 1. The high-order half of comparator B is not used.

In full trigger modes it is not useful to specify a tag-type CPU breakpoint (BRKEN = TAG = 1), but if you do, the comparator B data match is ignored for the purpose of issuing the tag request to the CPU and the CPU breakpoint is issued when the comparator A address matches.

A AND NOT B Data (Full Mode) — Address must match comparator A, data must not match the low half of comparator B, and R/W must match RWA if RWAEN = 1. All three conditions must be met within the same bus cycle to cause a trigger.

In full trigger modes it is not useful to specify a tag-type CPU breakpoint (BRKEN = TAG = 1), but if you do, the comparator B data match is ignored for the purpose of issuing the tag request to the CPU and the CPU breakpoint is issued when the comparator A address matches.

Event-Only B (Store Data) — Trigger events occur each time the address matches the value in comparator B. Trigger events cause the data to be captured into the FIFO. The debug run ends when the FIFO becomes full.

A Then Event-Only B (Store Data) — After the address has matched the value in comparator A, a trigger event occurs each time the address matches the value in comparator B. Trigger events cause the data to be captured into the FIFO. The debug run ends when the FIFO becomes full.

Inside Range (A ≤ Address ≤ B) — A trigger occurs when the address is greater than or equal to the value in comparator A and less than or equal to the value in comparator B at the same time.

Outside Range (Address < A or Address > B) — A trigger occurs when the address is either less than the value in comparator A or greater than the value in comparator B.

A.6 Supply Current Characteristics

Table A-6. Supply Current Characteristics

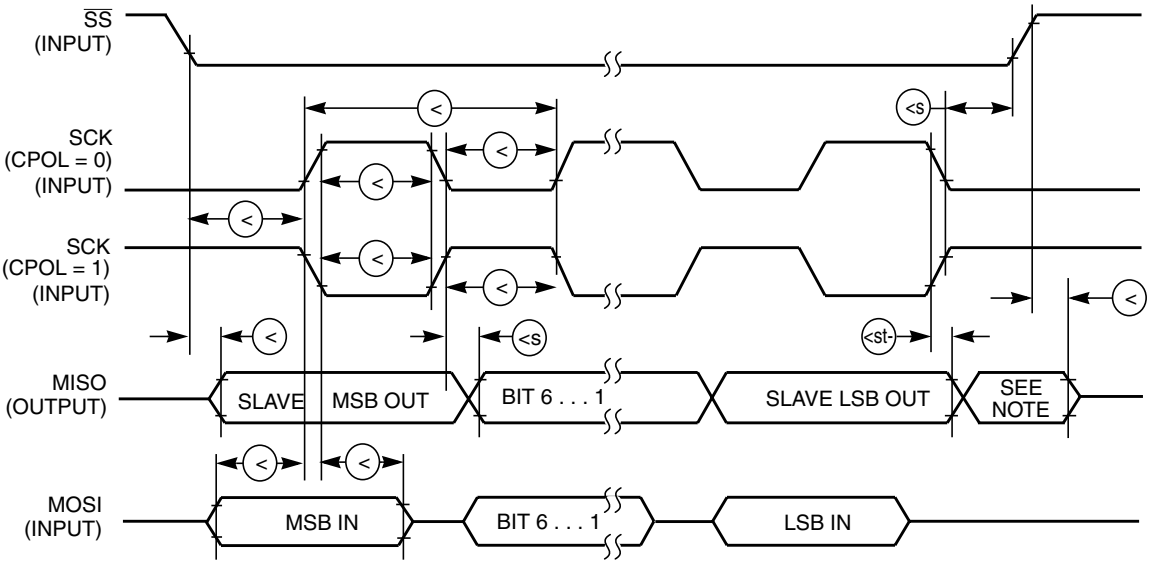
Parameter	Symbol	V_{DD} (V)[1]	Typical[2]	Max	Temp. (°C)
Run supply current[3] measured at (CPU clock = 2 MHz, f_{Bus} = 1 MHz)	RI_{DD}	3	500 µA	1.525 mA 1.525 mA	70 85
		2	450 µA	1.475 mA 1.475 mA	70 85
Run supply current[3] measured at (CPU clock = 16 MHz, f_{Bus} = 8 MHz)	RI_{DD}	3	3.8 mA	4.8 mA 4.8 mA	70 85
		2	2.6 mA	3.6 mA 3.6 mA	70 85
Stop1 mode supply current	$S1I_{DD}$	3	100 nA	350 nA 736 nA	70 85
		2	100 nA	150 nA 450 nA	70 85
Stop2 mode supply current	$S2I_{DD}$	3	500 nA	1.20 µA 1.90 µA	70 85
		2	500 nA	1.00 µA 1.70 µA	70 85
Stop3 mode supply current	$S3I_{DD}$	3	600 nA	2.65 µA 4.65 µA	70 85
		2	500 nA	2.30 µA 4.30 µA	70 85
RTI adder from stop2 or stop3		3	300 nA		
		2	300 nA		
Adder for LVD reset enabled in stop3		3	70 µA		
		2	60 µA		

1. 3 V values are 100% tested; 2 V values are characterized but not tested.
2. Typicals are measured at 25°C.
3. Does not include any dc loads on port pins

A.7 Analog Comparator (ACMP) Electricals

Table A-7. ACMP Electrical Specifications (Temp Range = -40 to 85° C Ambient)

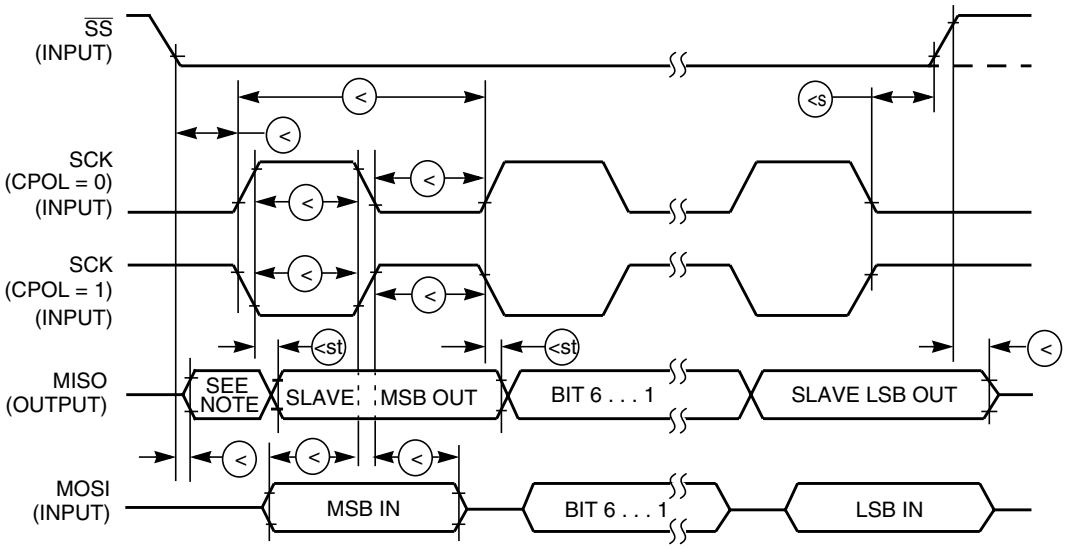
Characteristic	Symbol	Min	Typical	Max	Unit
Analog input voltage	VAIN	V_{SS} – 0.3	—	V_{DD}	V
Analog input offset voltage	VAIO		—	40	mV
Analog Comparator initialization delay	t_{AINIT}		—	1	µs
Analog Comparator bandgap reference voltage	V_{BG}	1.208	1.218	1.228	V



NOTE:

1. Not defined but normally MSB of character just received

Figure A-13. SPI Slave Timing (CPHA = 0)



NOTE:

1. Not defined but normally LSB of character just received

Figure A-14. SPI Slave Timing (CPHA = 1)