



Welcome to [E-XFL.COM](#)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I ² C, LINbus, SCI, SPI
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)	2.7V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°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/product-detail/nxp-semiconductors/mc9s08sh16vwl

Contents

Section Number	Title	Page
Chapter 1		
Device Overview		
1.1	Devices in the MC9S08SH32 Series	19
1.2	MCU Block Diagram	20
1.3	System Clock Distribution	22
Chapter 2		
Pins and Connections		
2.1	Device Pin Assignment	23
2.2	Recommended System Connections	25
2.2.1	Power	26
2.2.2	Oscillator (XOSC)	26
2.2.3	RESET	27
2.2.4	Background / Mode Select (BKGD/MS)	27
2.2.5	General-Purpose I/O and Peripheral Ports	28
Chapter 3		
Modes of Operation		
3.1	Introduction	31
3.2	Features	31
3.3	Run Mode	31
3.4	Active Background Mode	31
3.5	Wait Mode	32
3.6	Stop Modes	32
3.6.1	Stop3 Mode	33
3.6.2	Stop2 Mode	34
3.6.3	On-Chip Peripheral Modules in Stop Modes	34
Chapter 4		
Memory		
4.1	MC9S08SH32 Series Memory Map	37
4.2	Reset and Interrupt Vector Assignments	38
4.3	Register Addresses and Bit Assignments	39
4.4	RAM	46
4.5	FLASH	46
4.5.1	Features	47
4.5.2	Program and Erase Times	47

Section Number	Title	Page
13.3	Register Definition	194
13.3.1	RTC Status and Control Register (RTCSC)	195
13.3.2	RTC Counter Register (RTCCNT)	196
13.3.3	RTC Modulo Register (RTCMOD)	196
13.4	Functional Description	196
13.4.1	RTC Operation Example	197
13.5	Initialization/Application Information	198

Chapter 14

Serial Communications Interface (S08SCIV4)

14.1	Introduction	201
14.1.1	Features	203
14.1.2	Modes of Operation	203
14.1.3	Block Diagram	204
14.2	Register Definition	206
14.2.1	SCI Baud Rate Registers (SCIxBDH, SCIxBDL)	206
14.2.2	SCI Control Register 1 (SCIxC1)	207
14.2.3	SCI Control Register 2 (SCIxC2)	208
14.2.4	SCI Status Register 1 (SCIxS1)	209
14.2.5	SCI Status Register 2 (SCIxS2)	211
14.2.6	SCI Control Register 3 (SCIxC3)	212
14.2.7	SCI Data Register (SCIxD)	213
14.3	Functional Description	213
14.3.1	Baud Rate Generation	213
14.3.2	Transmitter Functional Description	214
14.3.3	Receiver Functional Description	215
14.3.4	Interrupts and Status Flags	217
14.3.5	Additional SCI Functions	218

Chapter 15

Serial Peripheral Interface (S08SPIV3)

15.1	Introduction	221
15.1.1	Features	223
15.1.2	Block Diagrams	223
15.1.3	SPI Baud Rate Generation	225
15.2	External Signal Description	226
15.2.1	SPSCK — SPI Serial Clock	226
15.2.2	MOSI — Master Data Out, Slave Data In	226
15.2.3	MISO — Master Data In, Slave Data Out	226
15.2.4	\overline{SS} — Slave Select	226
15.3	Modes of Operation	227
15.3.1	SPI in Stop Modes	227

Section Number	Title	Page
15.4	Register Definition	227
15.4.1	SPI Control Register 1 (SPIxC1)	227
15.4.2	SPI Control Register 2 (SPIxC2)	228
15.4.3	SPI Baud Rate Register (SPIxBR)	229
15.4.4	SPI Status Register (SPIxS)	230
15.4.5	SPI Data Register (SPIxD)	231
15.5	Functional Description	232
15.5.1	SPI Clock Formats	232
15.5.2	SPI Interrupts	235
15.5.3	Mode Fault Detection	235

Chapter 16

Timer Pulse-Width Modulator (S08TPMV3)

16.1	Introduction	237
16.1.1	TPM Configuration Information	237
16.1.2	TPM Pin Repositioning	237
16.1.3	Features	239
16.1.4	Modes of Operation	239
16.1.5	Block Diagram	240
16.2	Signal Description	242
16.2.1	Detailed Signal Descriptions	242
16.3	Register Definition	246
16.3.1	TPM Status and Control Register (TPMxSC)	246
16.3.2	TPM-Counter Registers (TPMxCNTH:TPMxCNTL)	247
16.3.3	TPM Counter Modulo Registers (TPMxMODH:TPMxMODL)	248
16.3.4	TPM Channel n Status and Control Register (TPMxCnSC)	249
16.3.5	TPM Channel Value Registers (TPMxCnVH:TPMxCnVL)	251
16.4	Functional Description	252
16.4.1	Counter	253
16.4.2	Channel Mode Selection	255
16.5	Reset Overview	258
16.5.1	General	258
16.5.2	Description of Reset Operation	258
16.6	Interrupts	258
16.6.1	General	258
16.6.2	Description of Interrupt Operation	259

Chapter 17

Development Support

17.1	Introduction	261
17.1.1	Forcing Active Background	261
17.1.2	Features	262

Table 2-1. Pin Availability by Package Pin-Count

Pin Number			<div> <div>←</div> <div>Priority</div> <div>→</div> <div>Lowest</div> <div>Highest</div> </div>					
28-pin	20-pin	16-pin	Port Pin	Alt 1	Alt 2	Alt 3	Alt 4	Alt5
1	—	—	PTC5					ADP13
2	—	—	PTC4					ADP12
3	1	1	PTA5	IRQ	TCLK			RESET ¹
4	2	2	PTA4	ACMPO			BKGD	MS
5	3	3						V _{DD}
6							V _{DDA}	V _{REFH}
7	4	4					V _{SSA}	V _{REFL}
8								V _{SS}
9	5	5	PTB7	SCL ²	EXTAL			
10	6	6	PTB6	SDA ²	XTAL			
11	7	7	PTB5	TPM1CH1 ³	$\overline{\text{SS}}$	PTC0 ⁴		
12	8	8	PTB4	TPM2CH1 ⁵	MISO	PTC0 ⁴		
13	9	—	PTC3			PTC0 ⁴	ADP11	
14	10	—	PTC2			PTC0 ⁴	ADP10	
15	11	—	PTC1	TPM1CH1 ³		PTC0 ⁴	ADP9	
16	12	—	PTC0	TPM1CH0 ³		PTC0 ⁴	ADP8	
17	13	9	PTB3	PIB3	MOSI	PTC0 ⁴	ADP7	
18	14	10	PTB2	PIB2	SPSCK	PTC0 ⁴	ADP6	
19	15	11	PTB1	PIB1	TxD		ADP5	
20	16	12	PTB0	PIB0	RxD		ADP4	
21	—	—	PTA7	TPM2CH1 ⁵				
22	—	—	PTA6	TPM2CH0 ⁵				
23	17	13	PTA3	PIA3	SCL ²		ADP3	
24	18	14	PTA2	PIA2	SDA ²		ADP2	
25	19	15	PTA1	PIA1	TPM2CH0 ⁵		ADP1 ⁶	ACMP- ⁶
26	20	16	PTA0	PIA0	TPM1CH0 ³		ADP0 ⁶	ACMP+ ⁶
27	—	—	PTC7					ADP15
28	—	—	PTC6					ADP14

¹ 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 $\overline{\text{RESET}}$ in will not be pulled to V_{DD} . The internal gates connected to this pin are pulled to V_{DD} .

² IIC pins can be repositioned using IICPS in SOPT2, default reset locations are PTA2, PTA3.

³ TPM1CHx pins can be repositioned using T1CHxPS bits in SOPT2, default reset locations are PTA0, PTB5.

⁴ This port pin is part of the ganged output feature. When pin is enabled for ganged output, it will have priority over all digital modules. The output data, drive strength and slew-rate control of this port pin will follow the configuration for the PTC0 pin, even in 16-pin packages where PTC0 doesn't bond out.

⁵ TPM2CHx pins can be repositioned using T2CHxPS bits in SOPT2, default reset locations are PTA1, PTB4.

⁶ If ACMP and ADC are both enabled, both will have access to the pin.

5.4 Computer Operating Properly (COP) Watchdog

The COP watchdog is intended to force a system reset when the application software fails to execute as expected. To prevent a system reset from the COP timer (when it is enabled), application software must reset the COP counter periodically. If the application program gets lost and fails to reset the COP counter before it times out, a system reset is generated to force the system back to a known starting point.

After any reset, the COP watchdog is enabled (see [Section 5.7.4, “System Options Register 1 \(SOPT1\),”](#) for additional information). If the COP watchdog is not used in an application, it can be disabled by clearing COPT bits in SOPT1.

The COP counter is reset by writing 0x0055 and 0x00AA (in this order) to the address of SRS during the selected timeout period. Writes do not affect the data in the read-only SRS. As soon as the write sequence is done, the COP timeout period is restarted. If the program fails to do this during the time-out period, the MCU will reset. Also, if any value other than 0x0055 or 0x00AA is written to SRS, the MCU is immediately reset.

The COPCLKS bit in SOPT2 (see [Section 5.7.5, “System Options Register 2 \(SOPT2\),”](#) for additional information) selects the clock source used for the COP timer. The clock source options are either the bus clock or an internal 1-kHz clock source. With each clock source, there are three associated time-outs controlled by the COPT bits in SOPT1. [Table 5-1](#) summarizes the control functions of the COPCLKS and COPT bits. The COP watchdog defaults to operation from the 1-kHz clock source and the longest time-out (2^{10} cycles).

Table 5-1. COP Configuration Options

Control Bits		Clock Source	COP Window ¹ Opens (COPW = 1)	COP Overflow Count
COPCLKS	COPT[1:0]			
N/A	0:0	N/A	N/A	COP is disabled
0	0:1	1 kHz	N/A	2^5 cycles (32 ms ²)
0	1:0	1 kHz	N/A	2^8 cycles (256 ms ¹)
0	1:1	1 kHz	N/A	2^{10} cycles (1.024 s ¹)
1	0:1	Bus	6144 cycles	2^{13} cycles
1	1:0	Bus	49,152 cycles	2^{16} cycles
1	1:1	Bus	196,608 cycles	2^{18} cycles

¹ Windowed COP operation requires the user to clear the COP timer in the last 25% of the selected timeout period. This column displays the minimum number of clock counts required before the COP timer can be reset when in windowed COP mode (COPW = 1).

² Values shown in milliseconds based on $t_{LPO} = 1$ ms. See t_{LPO} in the appendix [Section A.12.1, “Control Timing,”](#) for the tolerance of this value.

When the bus clock source is selected, windowed COP operation is available by setting COPW in the SOPT2 register. In this mode, writes to the SRS register to clear the COP timer must occur in the last 25% of the selected timeout period. A premature write immediately resets the MCU. When the 1-kHz clock source is selected, windowed COP operation is not available.

5.7.5 System Options Register 2 (SOPT2)

This high page register contains bits to configure MCU specific features on the MC9S08SH32 Series devices.

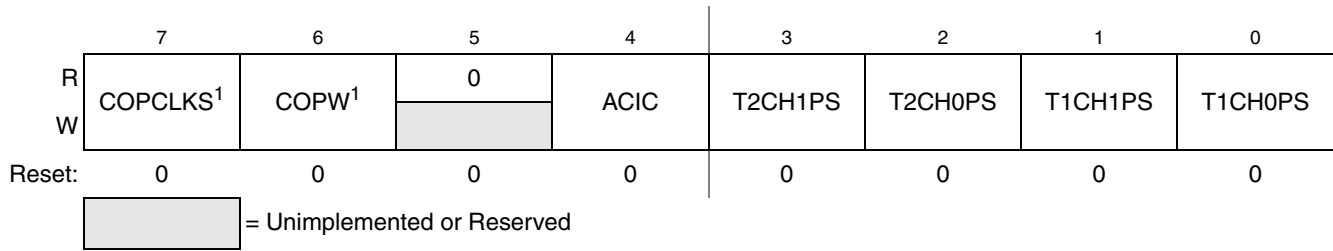


Figure 5-6. System Options Register 2 (SOPT2)

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

Table 5-7. SOPT2 Register Field Descriptions

Field	Description
7 COPCLKS	COP Watchdog Clock Select — This write-once bit selects the clock source of the COP watchdog. 0 Internal 1-kHz clock is source to COP. 1 Bus clock is source to COP.
6 COPW	COP Window — This write-once bit selects the COP operation mode. When set, the 0x55-0xAA write sequence to the SRS register must occur in the last 25% of the selected period. Any write to the SRS register during the first 75% of the selected period will reset the MCU. 0 Normal COP operation 1 Window COP operation (only if COPCLKS = 1)
4 ACIC	Analog Comparator to Input Capture Enable — This bit connects the output of ACMP to TPM1 input channel 0. 0 ACMP output not connected to TPM1 input channel 0. 1 ACMP output connected to TPM1 input channel 0.
3 T2CH1PS	TPM2CH1 Pin Select — This selects the location of the TPM2CH1 pin of the TPM2 module. 0 TPM2CH1 on PTB4. 1 TPM2CH1 on PTA7.
2 T2CH0PS	TPM2CH0 Pin Select — This bit selects the location of the TPM2CH0 pin of the TPM2 module. 0 TPM2CH0 on PTA1. 1 TPM2CH0 on PTA6.
1 T1CH1PS	TPM1CH1 Pin Select — This selects the location of the TPM1CH1 pin of the TPM1 module. 0 TPM1CH1 on PTB5. 1 TPM1CH1 on PTC1.
0 T1CH0PS	TPM1CH0 Pin Select — This bit selects the location of the TPM1CH0 pin of the TPM1 module. 0 TPM1CH0 on PTA0. 1 TPM1CH0 on PTC0.

Table 7-2. Instruction Set Summary (Sheet 8 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
SUB #opr8i SUB opr8a SUB opr16a SUB oprx16,X SUB oprx8,X SUB ,X SUB oprx16,SP SUB oprx8,SP	Subtract $A \leftarrow (A) - (M)$	IMM DIR EXT IX2 IX1 IX SP2 SP1	A0 ii B0 dd C0 hh ll D0 ee ff E0 ff F0 9E D0 ee ff 9E E0 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	\uparrow 1 1 -	- \uparrow \uparrow \uparrow
SWI	Software Interrupt $PC \leftarrow (PC) + \$0001$ Push (PCL); $SP \leftarrow (SP) - \$0001$ Push (PCH); $SP \leftarrow (SP) - \$0001$ Push (X); $SP \leftarrow (SP) - \$0001$ Push (A); $SP \leftarrow (SP) - \$0001$ Push (CCR); $SP \leftarrow (SP) - \$0001$ $I \leftarrow 1$; PCH \leftarrow Interrupt Vector High Byte PCL \leftarrow Interrupt Vector Low Byte	INH	83	11	sssssvvfppp	- 1 1 -	1 - - -
TAP	Transfer Accumulator to CCR $CCR \leftarrow (A)$	INH	84	1	p	\uparrow 1 1 \uparrow	\uparrow \uparrow \uparrow \uparrow
TAX	Transfer Accumulator to X (Index Register Low) $X \leftarrow (A)$	INH	97	1	p	- 1 1 -	- - - -
TPA	Transfer CCR to Accumulator $A \leftarrow (CCR)$	INH	85	1	p	- 1 1 -	- - - -
TST opr8a TSTA TSTX TST oprx8,X TST ,X TST oprx8,SP	Test for Negative or Zero (M) - \$00 (A) - \$00 (X) - \$00 (M) - \$00 (M) - \$00 (M) - \$00	DIR INH INH IX1 IX SP1	3D dd 4D 5D 6D ff 7D 9E 6D ff	4 1 1 4 3 5	rfpp p p rfpp rfp prfpp	0 1 1 -	- \uparrow \uparrow -
TSX	Transfer SP to Index Reg. $H:X \leftarrow (SP) + \$0001$	INH	95	2	fp	- 1 1 -	- - - -
TXA	Transfer X (Index Reg. Low) to Accumulator $A \leftarrow (X)$	INH	9F	1	p	- 1 1 -	- - - -

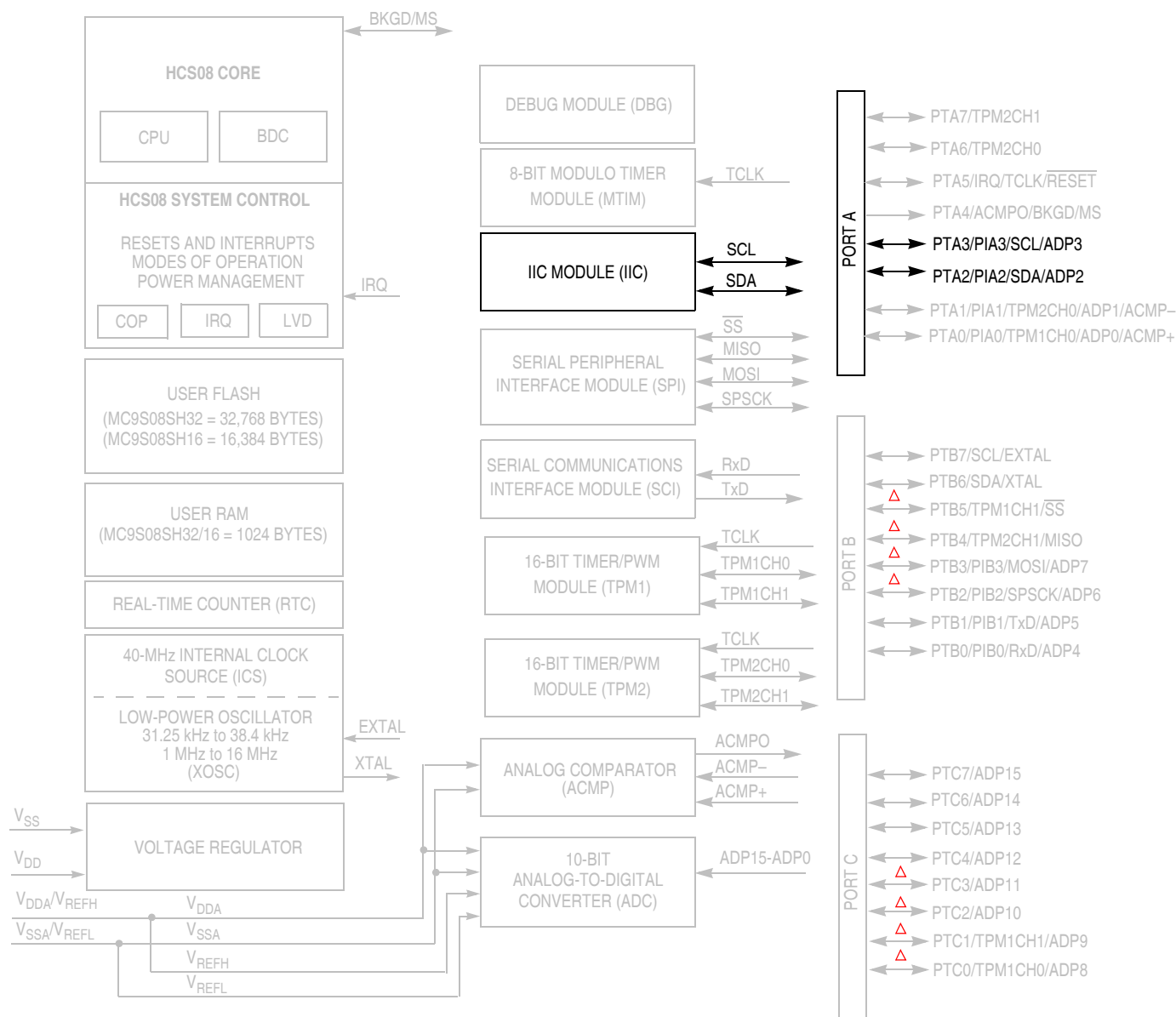
Table 7-3. Opcode Map (Sheet 2 of 2)

Bit-Manipulation	Branch	Read-Modify-Write				Control				Register/Memory							
					9E60 6 NEG 3 SP1						9ED0 5 SUB 4 SP2	9EE0 4 SUB 3 SP1					
					9E61 6 CBEQ 4 SP1						9ED1 5 CMP 4 SP2	9EE1 4 CMP 3 SP1					
											9ED2 5 SBC 4 SP2	9EE2 4 SBC 3 SP1					
					9E63 6 COM 3 SP1						9ED3 5 CPX 4 SP2	9EE3 4 CPX 3 SP1	9EF3 6 CPHX 3 SP1				
					9E64 6 LSR 3 SP1						9ED4 5 AND 4 SP2	9EE4 4 AND 3 SP1					
											9ED5 5 BIT 4 SP2	9EE5 4 BIT 3 SP1					
					9E66 6 ROR 3 SP1						9ED6 5 LDA 4 SP2	9EE6 4 LDA 3 SP1					
					9E67 6 ASR 3 SP1						9ED7 5 STA 4 SP2	9EE7 4 STA 3 SP1					
					9E68 6 LSL 3 SP1						9ED8 5 EOR 4 SP2	9EE8 4 EOR 3 SP1					
					9E69 6 ROL 3 SP1						9ED9 5 ADC 4 SP2	9EE9 4 ADC 3 SP1					
					9E6A 6 DEC 3 SP1						9EDA 5 ORA 4 SP2	9EEA 4 ORA 3 SP1					
					9E6B 8 DBNZ 4 SP1						9EDB 5 ADD 4 SP2	9EEB 4 ADD 3 SP1					
					9E6C 6 INC 3 SP1												
					9E6D 5 TST 3 SP1												
										9EAE 5 LDHX 2 IX	9EBE 6 LDHX 4 IX2	9ECE 5 LDHX 3 IX1	9EDE 5 LDX 4 SP2	9EEE 4 LDX 3 SP1	9EFE 5 LDHX 3 SP1		
					9E6F 6 CLR 3 SP1						9EDF 5 STX 4 SP2	9EEF 4 STX 3 SP1	9EFF 5 STHX 3 SP1				

INH Inherent REL Relative SP1 Stack Pointer, 8-Bit Offset
 IMM Immediate IX Indexed, No Offset SP2 Stack Pointer, 16-Bit Offset
 DIR Direct IX1 Indexed, 8-Bit Offset IX+ Indexed, No Offset with
 EXT Extended IX2 Indexed, 16-Bit Offset Post Increment
 DD DIR to DIR IMD IMM to DIR IX1+ Indexed, 1-Byte Offset with
 IX+D IX+ to DIR DIX+ DIR to IX+ Post Increment

Note: All Sheet 2 Opcodes are Preceded by the Page 2 Prebyte (9E)

Prebyte (9E) and Opcode in
 Hexadecimal 9E60 6
 NEG
 3 SP1
 Number of Bytes HCS08 Cycles
 Instruction Mnemonic
 Addressing Mode



NOTE:

- PTC7-PTC0 and PTA7-PTA6 not available on 16-pin Packages
- PTC7-PTC4 and PTA7-PTA6 not available on 20-pin Packages
- For the 16-pin and 20-pin packages: V_{DDA}/V_{REFH} and V_{SSA}/V_{REFL} are double bonded to V_{DD} and V_{SS} respectively.
- When PTA4 is configured as BKGD, pin becomes bi-directional.

Figure 10-1. MC9S08SH32 Series Block Diagram Highlighting IIC Block and Pins

10.4.1.5 Repeated Start Signal

As shown in Figure 10-9, a repeated start signal is a start signal generated without first generating a stop signal to terminate the communication. This is used by the master to communicate with another slave or with the same slave in different mode (transmit/receive mode) without releasing the bus.

10.4.1.6 Arbitration Procedure

The IIC bus is a true multi-master bus that allows more than one master to be connected on it. If two or more masters try to control the bus at the same time, a clock synchronization procedure determines the bus clock, for which the low period is equal to the longest clock low period and the high is equal to the shortest one among the masters. The relative priority of the contending masters is determined by a data arbitration procedure, a bus master loses arbitration if it transmits logic 1 while another master transmits logic 0. The losing masters immediately switch over to slave receive mode and stop driving SDA output. In this case, the transition from master to slave mode does not generate a stop condition. Meanwhile, a status bit is set by hardware to indicate loss of arbitration.

10.4.1.7 Clock Synchronization

Because wire-AND logic is performed on the SCL line, a high-to-low transition on the SCL line affects all the devices connected on the bus. The devices start counting their low period and after a device's clock has gone low, it holds the SCL line low until the clock high state is reached. However, the change of low to high in this device clock may not change the state of the SCL line if another device clock is still within its low period. Therefore, synchronized clock SCL is held low by the device with the longest low period. Devices with shorter low periods enter a high wait state during this time (see Figure 10-10). When all devices concerned have counted off their low period, the synchronized clock SCL line is released and pulled high. There is then no difference between the device clocks and the state of the SCL line and all the devices start counting their high periods. The first device to complete its high period pulls the SCL line low again.

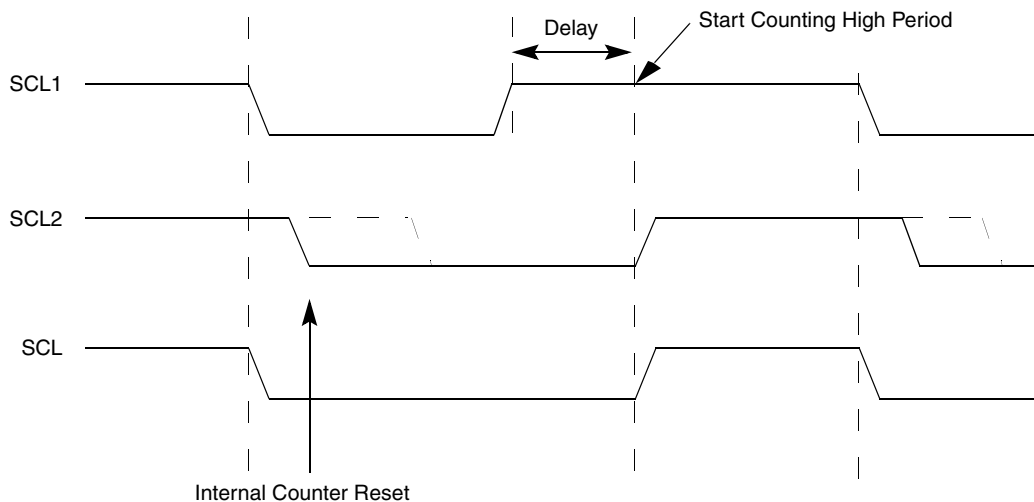


Figure 10-10. IIC Clock Synchronization

After a repeated start condition (Sr), all other slave devices also compare the first seven bits of the first byte of the slave address with their own addresses and test the eighth (R/\overline{W}) bit. However, none of them are addressed because $R/\overline{W} = 1$ (for 10-bit devices) or the 11110XX slave address (for 7-bit devices) does not match.

S	Slave Address 1st 7 bits 11110 + AD10 + AD9	R/W 0	A1	Slave Address 2nd byte AD[8:1]	A2	Sr	Slave Address 1st 7 bits 11110 + AD10 + AD9	R/W 1	A3	Data	A	...	Data	A	P
---	---	----------	----	--------------------------------------	----	----	---	----------	----	------	---	-----	------	---	---

Table 10-11. Master-Receiver Addresses a Slave-Transmitter with a 10-bit Address

After the master-receiver has sent the first byte of the 10-bit address, the slave-transmitter sees an IIC interrupt. Software must ensure the contents of IICD are ignored and not treated as valid data for this interrupt.

10.4.3 General Call Address

General calls can be requested in 7-bit address or 10-bit address. If the GCAEN bit is set, the IIC matches the general call address as well as its own slave address. When the IIC responds to a general call, it acts as a slave-receiver and the IAAS bit is set after the address cycle. Software must read the IICD register after the first byte transfer to determine whether the address matches its own slave address or a general call. If the value is 00, the match is a general call. If the GCAEN bit is clear, the IIC ignores any data supplied from a general call address by not issuing an acknowledgement.

10.5 Resets

The IIC is disabled after reset. The IIC cannot cause an MCU reset.

10.6 Interrupts

The IIC generates a single interrupt.

An interrupt from the IIC is generated when any of the events in [Table 10-12](#) occur, provided the IICIE bit is set. The interrupt is driven by bit IICIF (of the IIC status register) and masked with bit IICIE (of the IIC control register). The IICIF bit must be cleared by software by writing a 1 to it in the interrupt routine. You can determine the interrupt type by reading the status register.

Table 10-12. Interrupt Summary

Interrupt Source	Status	Flag	Local Enable
Complete 1-byte transfer	TCF	IICIF	IICIE
Match of received calling address	IAAS	IICIF	IICIE
Arbitration Lost	ARBL	IICIF	IICIE

10.6.1 Byte Transfer Interrupt

The TCF (transfer complete flag) bit is set at the falling edge of the ninth clock to indicate the completion of byte transfer.

12.1.2 Features

Timer system features include:

- 8-bit up-counter
 - Free-running or 8-bit modulo limit
 - Software controllable interrupt on overflow
 - Counter reset bit (TRST)
 - Counter stop bit (TSTP)
- Four software selectable clock sources for input to prescaler:
 - System bus clock — rising edge
 - Fixed frequency clock (XCLK) — rising edge
 - External clock source on the TCLK pin — rising edge
 - External clock source on the TCLK pin — falling edge
- Nine selectable clock prescale values:
 - Clock source divide by 1, 2, 4, 8, 16, 32, 64, 128, or 256

12.1.3 Modes of Operation

This section defines the MTIM's operation in stop, wait and background debug modes.

12.1.3.1 MTIM in Wait Mode

The MTIM continues to run in wait mode if enabled before executing the WAIT instruction. Therefore, the MTIM can be used to bring the MCU out of wait mode if the timer overflow interrupt is enabled. For lowest possible current consumption, the MTIM should be stopped by software if not needed as an interrupt source during wait mode.

12.1.3.2 MTIM in Stop Modes

The MTIM is disabled in all stop modes, regardless of the settings before executing the STOP instruction. Therefore, the MTIM cannot be used as a wake up source from stop modes.

Waking from stop1 and stop2 modes, the MTIM will be put into its reset state. If stop3 is exited with a reset, the MTIM will be put into its reset state. If stop3 is exited with an interrupt, the MTIM continues from the state it was in when stop3 was entered. If the counter was active upon entering stop3, the count will resume from the current value.

12.1.3.3 MTIM in Active Background Mode

The MTIM suspends all counting until the microcontroller returns to normal user operating mode. Counting resumes from the suspended value as long as an MTIM reset did not occur (TRST written to a 1 or MTIMMOD written).

13.3.2 RTC Counter Register (RTCCNT)

RTCCNT is the read-only value of the current RTC count of the 8-bit counter.

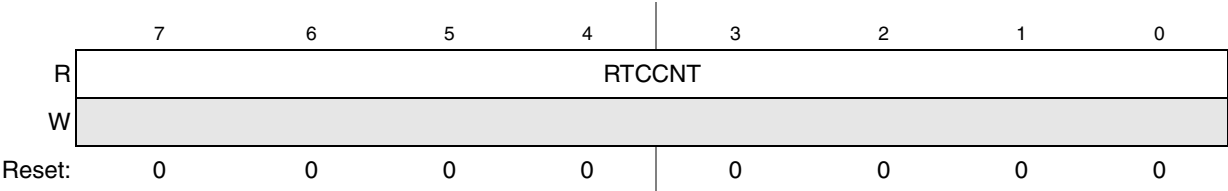


Figure 13-4. RTC Counter Register (RTCCNT)

Table 13-4. RTCCNT Field Descriptions

Field	Description
7:0 RTCCNT	RTC Count. These eight read-only bits contain the current value of the 8-bit counter. Writes have no effect to this register. Reset, writing to RTCMOD, or writing different values to RTCLKS and RTCPS clear the count to 0x00.

13.3.3 RTC Modulo Register (RTCMOD)

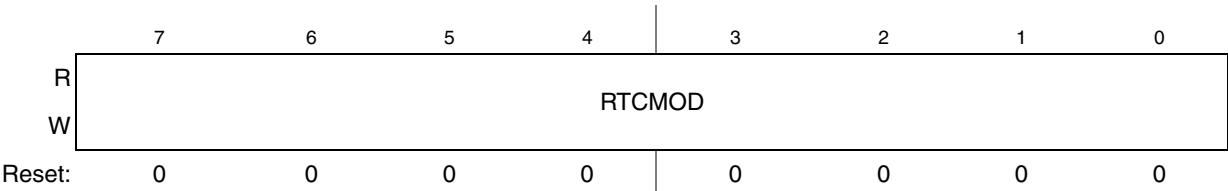


Figure 13-5. RTC Modulo Register (RTCMOD)

Table 13-5. RTCMOD Field Descriptions

Field	Description
7:0 RTCMOD	RTC Modulo. These eight read/write bits contain the modulo value used to reset the count to 0x00 upon a compare match and set the RTIF status bit. A value of 0x00 sets the RTIF bit on each rising edge of the prescaler output. Writing to RTCMOD resets the prescaler and the RTCCNT counters to 0x00. Reset sets the modulo to 0x00.

13.4 Functional Description

The RTC is composed of a main 8-bit up-counter with an 8-bit modulo register, a clock source selector, and a prescaler block with binary-based and decimal-based selectable values. The module also contains software selectable interrupt logic.

After any MCU reset, the counter is stopped and reset to 0x00, the modulus register is set to 0x00, and the prescaler is off. The 1-kHz internal oscillator clock is selected as the default clock source. To start the prescaler, write any value other than zero to the prescaler select bits (RTCPS).

Three clock sources are software selectable: the low power oscillator clock (LPO), the external clock (ERCLK), and the internal clock (IRCLK). The RTC clock select bits (RTCLKS) select the desired clock source. If a different value is written to RTCLKS, the prescaler and RTCCNT counters are reset to 0x00.

Table 14-5. SCIS1 Field Descriptions

Field	Description
7 TDRE	Transmit Data Register Empty Flag — 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 SCIS1 with TDRE = 1 and then write to the SCI data register (SCID). 0 Transmit data register (buffer) full. 1 Transmit data register (buffer) empty.
6 TC	Transmission Complete Flag — TC is set out of reset and when TDRE = 1 and no data, preamble, or break character is being transmitted. 0 Transmitter active (sending data, a preamble, or a break). 1 Transmitter idle (transmission activity complete). TC is cleared automatically by reading SCIS1 with TC = 1 and then doing one of the following three things: <ul style="list-style-type: none"> Write to the SCI data register (SCID) to transmit new data Queue a preamble by changing TE from 0 to 1 Queue a break character by writing 1 to SBK in SCIC2
5 RDRF	Receive Data Register Full Flag — RDRF becomes set when a character transfers from the receive shifter into the receive data register (SCID). To clear RDRF, read SCIS1 with RDRF = 1 and then read the SCI data register (SCID). 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 SCIS1 with IDLE = 1 and then read the SCI data register (SCID). 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 SCID yet. In this case, the new character (and all associated error information) is lost because there is no room to move it into SCID. To clear OR, read SCIS1 with OR = 1 and then read the SCI data register (SCID). 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 SCIS1 and then read the SCI data register (SCID). 0 No noise detected. 1 Noise detected in the received character in SCID.

Writing 0 to TE does not immediately release the pin to be a general-purpose I/O pin. Any transmit activity that is in progress must first be completed. This includes data characters in progress, queued idle characters, and queued break characters.

14.3.2.1 Send Break and Queued Idle

The SBK control bit in SCIx_C2 is used to send break characters which were originally used to gain the attention of old teletype receivers. Break characters are a full character time of logic 0 (10 bit times including the start and stop bits). A longer break of 13 bit times can be enabled by setting BRK13 = 1. Normally, a program would wait for TDRE to become set to indicate the last character of a message has moved to the transmit shifter, then write 1 and then write 0 to the SBK bit. This action queues a break character to be sent as soon as the shifter is available. If SBK is still 1 when the queued break moves into the shifter (synchronized to the baud rate clock), an additional break character is queued. If the receiving device is another Freescale Semiconductor SCI, the break characters will be received as 0s in all eight data bits and a framing error (FE = 1) occurs.

When idle-line wakeup is used, a full character time of idle (logic 1) is needed between messages to wake up any sleeping receivers. Normally, a program would wait for TDRE to become set to indicate the last character of a message has moved to the transmit shifter, then write 0 and then write 1 to the TE bit. This action queues an idle character to be sent as soon as the shifter is available. As long as the character in the shifter does not finish while TE = 0, the SCI transmitter never actually releases control of the TxD pin. If there is a possibility of the shifter finishing while TE = 0, set the general-purpose I/O controls so the pin that is shared with TxD is an output driving a logic 1. This ensures that the TxD line will look like a normal idle line even if the SCI loses control of the port pin between writing 0 and then 1 to TE.

The length of the break character is affected by the BRK13 and M bits as shown below.

Table 14-8. Break Character Length

BRK13	M	Break Character Length
0	0	10 bit times
0	1	11 bit times
1	0	13 bit times
1	1	14 bit times

14.3.3 Receiver Functional Description

In this section, the receiver block diagram (Figure 14-3) is used as a guide for the overall receiver functional description. Next, the data sampling technique used to reconstruct receiver data is described in more detail. Finally, two variations of the receiver wakeup function are explained.

The receiver input is inverted by setting RXINV = 1. The receiver is enabled by setting the RE bit in SCIx_C2. Character frames consist of a start bit of logic 0, eight (or nine) data bits (LSB first), and a stop bit of logic 1. For information about 9-bit data mode, refer to Section 14.3.5.1, “8- and 9-Bit Data Modes.” For the remainder of this discussion, we assume the SCI is configured for normal 8-bit data mode.

After receiving the stop bit into the receive shifter, and provided the receive data register is not already full, the data character is transferred to the receive data register and the receive data register full (RDRF)

status flag is set. If RDRF was already set indicating the receive data register (buffer) was already full, the overrun (OR) status flag is set and the new data is lost. Because the SCI receiver is double-buffered, the program has one full character time after RDRF is set before the data in the receive data buffer must be read to avoid a receiver overrun.

When a program detects that the receive data register is full ($RDRF = 1$), it gets the data from the receive data register by reading SCIxD. The RDRF flag is cleared automatically by a 2-step sequence which is normally satisfied in the course of the user's program that handles receive data. Refer to [Section 14.3.4, "Interrupts and Status Flags"](#) for more details about flag clearing.

14.3.3.1 Data Sampling Technique

The SCI receiver uses a $16\times$ baud rate clock for sampling. The receiver starts by taking logic level samples at 16 times the baud rate to search for a falling edge on the RxD serial data input pin. A falling edge is defined as a logic 0 sample after three consecutive logic 1 samples. The $16\times$ baud rate clock is used to divide the bit time into 16 segments labeled RT1 through RT16. When a falling edge is located, three more samples are taken at RT3, RT5, and RT7 to make sure this was a real start bit and not merely noise. If at least two of these three samples are 0, the receiver assumes it is synchronized to a receive character.

The receiver then samples each bit time, including the start and stop bits, at RT8, RT9, and RT10 to determine the logic level for that bit. The logic level is interpreted to be that of the majority of the samples taken during the bit time. In the case of the start bit, the bit is assumed to be 0 if at least two of the samples at RT3, RT5, and RT7 are 0 even if one or all of the samples taken at RT8, RT9, and RT10 are 1s. If any sample in any bit time (including the start and stop bits) in a character frame fails to agree with the logic level for that bit, the noise flag (NF) will be set when the received character is transferred to the receive data buffer.

The falling edge detection logic continuously looks for falling edges, and if an edge is detected, the sample clock is resynchronized to bit times. This improves the reliability of the receiver in the presence of noise or mismatched baud rates. It does not improve worst case analysis because some characters do not have any extra falling edges anywhere in the character frame.

In the case of a framing error, provided the received character was not a break character, the sampling logic that searches for a falling edge is filled with three logic 1 samples so that a new start bit can be detected almost immediately.

In the case of a framing error, the receiver is inhibited from receiving any new characters until the framing error flag is cleared. The receive shift register continues to function, but a complete character cannot transfer to the receive data buffer if FE is still set.

14.3.3.2 Receiver Wakeup Operation

Receiver wakeup is a hardware mechanism that allows an SCI receiver to ignore the characters in a message that is intended for a different SCI receiver. In such a system, all receivers evaluate the first character(s) of each message, and as soon as they determine the message is intended for a different receiver, they write logic 1 to the receiver wake up (RWU) control bit in SCIxC2. When RWU bit is set, the status flags associated with the receiver (with the exception of the idle bit, IDLE, when RWUID bit is set) are inhibited from setting, thus eliminating the software overhead for handling the unimportant

Figure 17-4 shows the host receiving a logic 0 from the target HCS08 MCU. Because the host is asynchronous to the target MCU, there is a 0-to-1 cycle delay from the host-generated falling edge on BKGD to the start of the bit time as perceived by the target MCU. The host initiates the bit time but the target HCS08 finishes it. Because the target wants the host to receive a logic 0, it drives the BKGD pin low for 13 BDC clock cycles, then briefly drives it high to speed up the rising edge. The host samples the bit level about 10 cycles after starting the bit time.

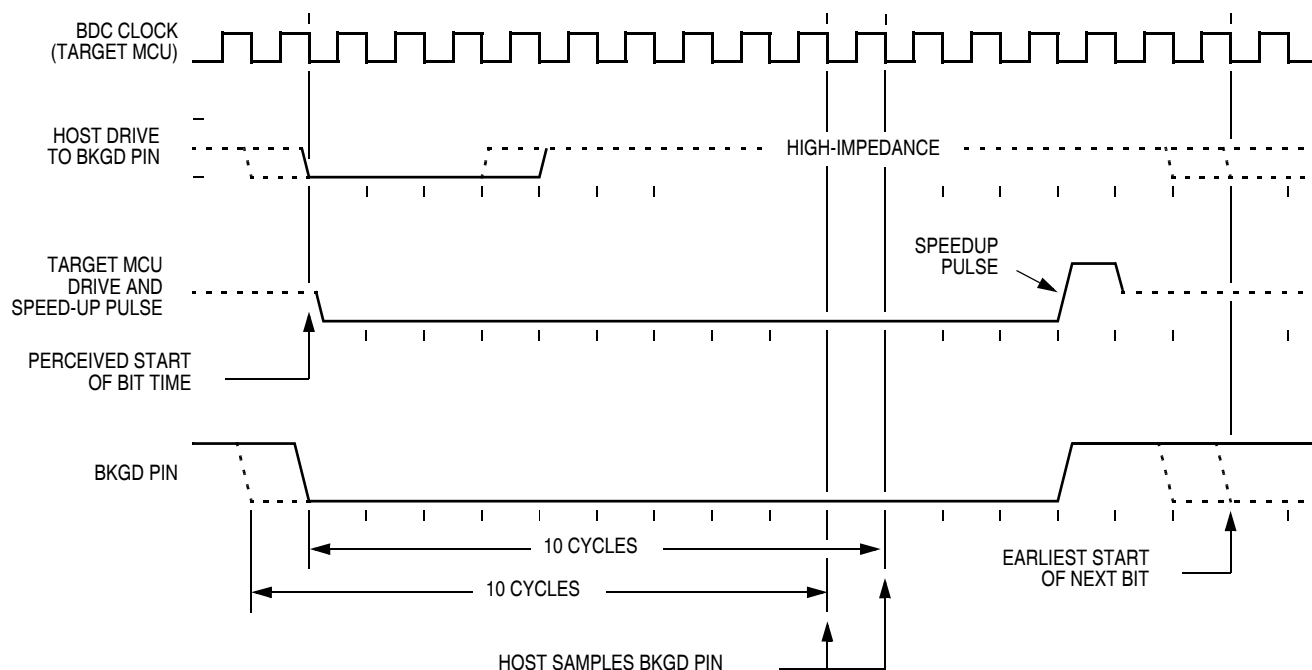


Figure 17-4. BDM Target-to-Host Serial Bit Timing (Logic 0)

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 ($A \leq \text{Address} \leq B$) — 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 ($\text{Address} < A$ or $\text{Address} > B$) — A trigger occurs when the address is either less than the value in comparator A or greater than the value in comparator B.

A.12.3 SPI

Table A-15 and Figure A-14 through Figure A-17 describe the timing requirements for the SPI system.

Table A-15. SPI Electrical Characteristic

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

¹ Refer to Figure A-14 through Figure A-17.

² All timing is shown with respect to 20% V_{DD} and 70% V_{DD} , unless noted; 100 pF load on all SPI pins. All timing assumes slew rate control disabled and high drive strength enabled for SPI output pins.

³ Time to data active from high-impedance state.

⁴ Hold time to high-impedance state.

⁵ Maximum baud rate must be limited to 5 MHz due to input filter characteristics.