E·XFL



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I ² C, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	12
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 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	16-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	16-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s08sg16e1mtg

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong





Revision Number	Revision Date	Description of Changes
8	5/2010	 In the A.9 ICS Characteristic table, changed row 9 parameter classification from a D to a P to indicate that these parameters are guaranteed during production testing on each individual device.
0	5/2010	 In the A.16 Flash Charateristic table, added the AEC temperature range to row 9. Revised Figure 2-1 so that the RESET pin shows the overbar.

© Freescale Semiconductor, Inc., 2007-2010. All rights reserved.

This product incorporates ${\sf SuperFlash}^{\tiny(\!6\!)}$ Technology licensed from SST.



Chapter 5 Resets, Interrupts, and General System Control

5.7.3 System Options Register 1 (SOPT1)

This high page register is a write-once register so only the first write after reset is honored. It can be read at any time. Any subsequent attempt to write to SOPT1 (intentionally or unintentionally) is ignored to avoid accidental changes to these sensitive settings. SOPT1 should 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.



= Unimplemented or Reserved

Figure 5-4. System Options Register 1 (SOPT1)

Table 5-5. SOPT1 Register Field Descriptions

Field	Description
7:6 COPT[1:0]	COP Watchdog Timeout — These write-once bits select the timeout period of the COP. COPT along with COPCLKS in SOPT2 defines the COP timeout period. See Table 5-1.
5 STOPE	 Stop Mode Enable — This write-once bit is used to enable 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.
2 IICPS	 IIC Pin Select — This bit selects the location of the SDA and SCL pins of the IIC module. 0 SDA on PTA2, SCL on PTA3. 1 SDA on PTB6, SCL on PTB7.



Bit-Mani	pulation	Branch		Rea	d-Modify-W	/rite	•	Coi	ntrol	· /		Register	/Memorv		
00 5	10 5	20 3	30 5	40 1	50 1	60 5	70 4	80 9	90 3	A0 2	B0 3	C0 4	D0 4	E0 3	F0 3
BRSET0	BSET0	BRA	NEG	NEGA	NEGX	NEG	NEG	RTI	BGE	SUB	SUB	SUB	SUB	SUB	SUB
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	2 REL	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
01 5	11 5	21 3	31 5	41 4	51 4	61 5	71 5	81 6	91 3	A1 2	B1 3	C1 4	D1 4	E1 3	F1 3
BRCLR0	BCLR0	BRN	CBEQ	CBEQA	CBEQX	CBEQ	CBEQ	RTS	BLT	CMP	CMP	CMP	CMP	CMP	CMP
3 DIR	2 DIR	2 REL	3 DIR	3 IMM	3 IMM	3 IX1+	2 IX+	1 INH	2 REL	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
02 5	12 5	22 3	32 5	42 5	52 6	62 1	72 1	82 5+	92 3	A2 2	B2 3	C2 4	D2 4	E2 3	F2 3
BRSET1	BSET1	BHI	LDHX	MUL	DIV	NSA	DAA	BGND	BGT	SBC	SBC	SBC	SBC	SBC	SBC
3 DIR	2 DIR	2 REL	3 EXT	1 INH	1 INH	1 INH	1 INH	1 INH	2 REL	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
03 5	13 5	23 3	33 5	43 1	53 1	63 5	73 4	83 11	93 3	A3 2	B3 3	C3 4	D3 4	E3 3	F3 3
BRCLR1	BCLR1	BLS	COM	COMA	COMX	COM	COM	SWI	BLE	CPX	CPX	CPX	CPX	CPX	CPX
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	2 REL	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
04 5 BRSET2 3 DIR	14 5 BSET2 2 DIR	BCC 2 REL	34 5 LSR 2 DIR	44 1 LSRA 1 INH	54 1 LSRX 1 INH	64 5 LSR 2 IX1	⁷⁴ 4 LSR 1 IX	84 1 TAP 1 INH	94 2 TXS 1 INH	A4 2 AND 2 IMM	AND 2 DIR	AND 3 EXT	D4 4 AND 3 IX2	E4 3 AND 2 IX1	F4 3 AND 1 IX
05 5 BRCLR2 3 DIR	15 5 BCLR2 2 DIR	25 3 BCS 2 REL	35 4 STHX 2 DIR	45 3 LDHX 3 IMM	LDHX 2 DIR	65 3 CPHX 3 IMM	75 5 CPHX 2 DIR	85 1 TPA 1 INH	95 2 TSX 1 INH	A5 2 BIT 2 IMM	B5 3 BIT 2 DIR	BIT 3 EXT	BIT 3 IX2	BIT 2 IX1	BIT 1 IX
06 5	16 5	26 3	36 5	46 1	56 1	66 5	76 4	86 3	96 5	A6 2	B6 3	C6 4	D6 4	E6 3	F6 3
BRSET3	BSET3	BNE	ROR	RORA	RORX	ROR	ROR	PULA	STHX	LDA	LDA	LDA	LDA	LDA	LDA
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	3 EXT	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
07 5	17 5	27 3	37 5	47 1	57 1	67 5	77 4	87 2	97 1	A7 2	B7 3	C7 4	D7 4	E7 3	F7 2
BRCLR3	BCLR3	BEQ	ASR	ASRA	ASRX	ASR	ASR	PSHA	TAX	AIS	STA	STA	STA	STA	STA
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	1 INH	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
08 5	18 5	28 3	38 5	48 1	58 1	68 5	78 4	88 3	98 1	A8 2	B8 3	C8 4	D8 4	E8 3	F8 3
BRSET4	BSET4	BHCC	LSL	LSLA	LSLX	LSL	LSL	PULX	CLC	EOR	EOR	EOR	EOR	EOR	EOR
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	1 INH	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
09 5	19 5	29 3	39 5	49 1	59 1	69 5	79 4	89 2	99 1	A9 2	B9 3	C9 4	D9 4	E9 3	F9 3
BRCLR4	BCLR4	BHCS	ROL	ROLA	ROLX	ROL	ROL	PSHX	SEC	ADC	ADC	ADC	ADC	ADC	ADC
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	1 INH	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
0A 5	1A 5	2A 3	3A 5	4A 1	5A 1	6A 5	7A 4	8A 3	9A 1	AA 2	BA 3	CA 4	DA 4	EA 3	FA 3
BRSET5	BSET5	BPL	DEC	DECA	DECX	DEC	DEC	PULH	CLI	ORA	ORA	ORA	ORA	ORA	ORA
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	1 INH	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
0B 5	1B 5	2B 3	3B 7	4B 4	5B 4	6B 7	7B 6	8B 2	9B 1	AB 2	BB 3	CB 4	DB 4	EB 3	FB 3
BRCLR5	BCLR5	BMI	DBNZ	DBNZA	DBNZX	DBNZ	DBNZ	PSHH	SEI	ADD	ADD	ADD	ADD	ADD	ADD
3 DIR	2 DIR	2 REL	3 DIR	2 INH	2 INH	3 IX1	2 IX	1 INH	1 INH	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
0C 5	1C 5	2C 3	3C 5	4C 1	5C 1	6C 5	7C 4	8C 1	9C 1		BC 3	CC 4	DC 4	EC 3	FC 3
BRSET6	BSET6	BMC	INC	INCA	INCX	INC	INC	CLRH	RSP		JMP	JMP	JMP	JMP	JMP
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	1 INH		2 DIR	3 EXT	3 IX2	2 IX1	1 IX
0D 5	1D 5	2D 3	3D 4	4D 1	5D 1	6D 4	7D 3		9D 1	AD 5	BD 5	CD 6	DD 6	ED 5	FD 5
BRCLR6	BCLR6	BMS	TST	TSTA	TSTX	TST	TST		NOP	BSR	JSR	JSR	JSR	JSR	JSR
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX		1 INH	2 REL	2 DIR	3 EXT	3 IX2	2 IX1	1 IX
0E 5 BRSET7 3 DIR	1E 5 BSET7 2 DIR	2E 3 BIL 2 REL	3E 6 CPHX 3 EXT	4E 5 MOV 3 DD	5E 5 MOV 2 DIX+	6E 4 MOV 3 IMD	7E 5 MOV 2 IX+D	8E 2+ STOP 1 INH	9E Page 2	AE 2 LDX 2 IMM	BE 3 LDX 2 DIR	CE 4 LDX 3 EXT	DE 4 LDX 3 IX2	EE 3 LDX 2 IX1	FE 3 LDX 1 IX
0F 5	1F 5	2F 3	3F 5	4F 1	5F 1	6F 5	7F 4	8F 2+	9F 1	AF 2	BF 3	CF 4	DF 4	EF 3	FF 2
BRCLR7	BCLR7	BIH	CLR	CLRA	CLRX	CLR	CLR	WAIT	TXA	AIX	STX	STX	STX	STX	STX
3 DIR	2 DIR	2 REL	2 DIR	1 INH	1 INH	2 IX1	1 IX	1 INH	1 INH	2 IMM	2 DIR	3 EXT	3 IX2	2 IX1	1 IX

Table 7-3. Opcode Map (Sheet 1 of 2)

Inherent
Immediate
Direct
Extended
DIR to DIR
IX+ to DIR

REL IX IX1 IX2 IMD DIX+ Relative Indexed, No Offset Indexed, 8-Bit Offset Indexed, 16-Bit Offset IMM to DIR DIR to IX+

Stack Pointer, 8-Bit Offset Stack Pointer, 16-Bit Offset Indexed, No Offset with Post Increment Indexed, 1-Byte Offset with Post Increment

SP1 SP2 IX+

IX1+

Opcode in Hexadecimal F0 3 SUB Number of Bytes 1 IX HCS08 Cycles Instruction Mnemonic Addressing Mode



Chapter 8 Analog Comparator 5-V (S08ACMPV3)

8.1 Introduction

The analog comparator module (ACMP) provides a circuit for comparing two analog input voltages or for comparing one analog input voltage to an internal reference voltage. The comparator circuit is designed to operate across the full range of the supply voltage (rail-to-rail operation).

Figure 8-1 shows the MC9S08SG32 Series block diagram with the ACMP highlighted.

8.1.1 ACMP Configuration Information

When using the bandgap reference voltage for input to ACMP+, the user must enable the bandgap buffer by setting BGBE =1 in SPMSC1 see Section 5.7.6, "System Power Management Status and Control 1 Register (SPMSC1)". For value of bandgap voltage reference see Section A.6, "DC Characteristics".

8.1.2 ACMP/TPM Configuration Information

The ACMP module can be configured to connect the output of the analog comparator to TPM1 input capture channel 0 by setting ACIC in SOPT2. With ACIC set, the TPM1CH0 pin is not available externally regardless of the configuration of the TPM1 module for channel 0.



Chapter 8 Analog Comparator 5-V (S08ACMPV3)

8.6.1.1 ACMP Status and Control Register (ACMPSC)

ACMPSC contains the status flag and control bits which are used to enable and configure the ACMP.



Figure 8-3. ACMP Status and Control Register

Field	Description
7 ACME	Analog Comparator Module Enable — ACME enables the ACMP module. 0 ACMP not enabled 1 ACMP is enabled
6 ACBGS	 Analog Comparator Bandgap Select — ACBGS is used to select between the bandgap reference voltage or the ACMP+ pin as the input to the non-inverting input of the analog comparatorr. 0 External pin ACMP+ selected as non-inverting input to comparator 1 Internal reference select as non-inverting input to comparator Note: refer to this chapter introduction to verify if any other config bits are necessary to enable the bandgap reference in the chip level.
5 ACF	 Analog Comparator Flag — ACF is set when a compare event occurs. Compare events are defined by ACMOD. ACF is cleared by writing a one to ACF. 0 Compare event has not occurred 1 Compare event has occurred
4 ACIE	 Analog Comparator Interrupt Enable — ACIE enables the interrupt from the ACMP. When ACIE is set, an interrupt will be asserted when ACF is set. 0 Interrupt disabled 1 Interrupt enabled
3 ACO	Analog Comparator Output — Reading ACO will return the current value of the analog comparator output. ACO is reset to a 0 and will read as a 0 when the ACMP is disabled (ACME = 0).
2 ACOPE	 Analog Comparator Output Pin Enable — ACOPE is used to enable the comparator output to be placed onto the external pin, ACMPO. 0 Analog comparator output not available on ACMPO 1 Analog comparator output is driven out on ACMPO
1:0 ACMOD	 Analog Comparator Mode — ACMOD selects the type of compare event which sets ACF. 00 Encoding 0 — Comparator output falling edge 01 Encoding 1 — Comparator output rising edge 10 Encoding 2 — Comparator output falling edge 11 Encoding 3 — Comparator output rising or falling edge

Table 8-2. ACMP Status and Control Register Field Descriptions



Field	Description
5 ACFE	 Compare Function Enable — Enables the compare function. 0 Compare function disabled 1 Compare function enabled
4 ACFGT	 Compare Function Greater Than Enable — Configures the compare function to trigger when the result of the conversion of the input being monitored is greater than or equal to the compare level. The compare function defaults to triggering when the result of the compare of the input being monitored is less than the compare level. Compare triggers when input is less than compare level Compare triggers when input is greater than or equal to compare level

Table 9-5. ADCSC2 Register Field Descriptions (continued)

9.3.3 Data Result High Register (ADCRH)

In 10-bit operation, ADCRH contains the upper two bits of 10-bit conversion data. In 10-bit mode, ADCRH is updated each time a conversion completes except when automatic compare is enabled and the compare condition is not met. When configured for 8-bit mode, ADR[9:8] are cleared.

When automatic compare is not enabled, the value stored in ADCRH are the upper bits of the conversion result. When automatic compare is enabled, the conversion result is manipulated as described in Section 9.4.5, "Automatic Compare Function" prior to storage in ADCRH:ADCRL registers.

In 10-bit mode, reading ADCRH prevents the ADC from transferring subsequent conversion data into the result registers until ADCRL is read. If ADCRL is not read until after the next conversion is completed, the intermediate conversion data is lost. In 8-bit mode, there is no interlocking with ADCRL. If the MODE bits are changed, any data in ADCRH becomes invalid.



Figure 9-5. Data Result High Register (ADCRH)

9.3.4 Data Result Low Register (ADCRL)

ADCRL contains the lower eight bits of a 10-bit conversion data, and all eight bits of 8-bit conversion data. ADCRL is updated each time a conversion completes except when automatic compare is enabled and the compare condition is not met.

When automatic compare is not enabled, the value stored in ADCRL is the lower eight bits of the conversion result. When automatic compare is enabled, the conversion result is manipulated as described in Section 9.4.5, "Automatic Compare Function" prior to storage in ADCRH:ADCRL registers.

In 10-bit mode, reading ADCRH prevents the ADC from transferring subsequent conversion data into the result registers until ADCRL is read. If ADCRL is not read until the after next conversion is completed,



Chapter 10 Inter-Integrated Circuit (S08IICV2)



NOTE

- PTC7-PTC0 and PTA7-PTA6 are not available on 16-pin Packages
- PTC7-PTC4 and PTA7-PTA6 are not available on 20-pin Packages
- + For the 16-pin and 20-pin packages: $V_{\mbox{DDA}}/V_{\mbox{REFH}}$ and $V_{\mbox{SSA}}/V_{\mbox{REFL}}$ are
- double bonded to V_{DD} and V_{SS} respectively.

 Δ = Pin can be enabled as part of the ganged output drive feature

Figure 10-1. MC9S08SG32 Series Block Diagram Highlighting IIC Block and Pins



Chapter 10 Inter-Integrated Circuit (S08IICV2)



Figure 10-9. IIC Bus Transmission Signals

10.4.1.1 Start Signal

When the bus is free, no master device is engaging the bus (SCL and SDA lines are at logical high), a master may initiate communication by sending a start signal. As shown in Figure 10-9, a start signal is defined as a high-to-low transition of SDA while SCL is high. This signal denotes the beginning of a new data transfer (each data transfer may contain several bytes of data) and brings all slaves out of their idle states.

10.4.1.2 Slave Address Transmission

The first byte of data transferred immediately after the start signal is the slave address transmitted by the master. This is a seven-bit calling address followed by a R/\overline{W} bit. The R/\overline{W} bit tells the slave the desired direction of data transfer.

- 1 =Read transfer, the slave transmits data to the master.
- 0 = Write transfer, the master transmits data to the slave.

Only the slave with a calling address that matches the one transmitted by the master responds by sending back an acknowledge bit. This is done by pulling the SDA low at the ninth clock (see Figure 10-9).

No two slaves in the system may have the same address. If the IIC module is the master, it must not transmit an address equal to its own slave address. The IIC cannot be master and slave at the same time. However, if arbitration is lost during an address cycle, the IIC reverts to slave mode and operates correctly even if it is being addressed by another master.



Chapter 10 Inter-Integrated Circuit (S08IICV2)



Chapter 13 Real-Time Counter (S08RTCV1)

13.1 Introduction

The RTC module consists of one 8-bit counter, one 8-bit comparator, several binary-based and decimal-based prescaler dividers, two clock sources, and one programmable periodic interrupt. This module can be used for time-of-day, calendar or any task scheduling functions. It can also serve as a cyclic wake up from low power modes without the need of external components.

Chapter 13 Real-Time	Counter (S08RTC)	(1)				
Internal 1-kHz Clock Source						
RTC Clock (RTCPS = 0xA)						
RTCCNT	0x52	0x53	0x54	0x55	0x00	0x01
RTIF						
RTCMOD			0x	55		

Figure 13-6. RTC Counter Overflow Example

In the example of Figure 13-6, the selected clock source is the 1-kHz internal oscillator clock source. The prescaler (RTCPS) is set to 0xA or divide-by-4. The modulo value in the RTCMOD register is set to 0x55. When the counter, RTCCNT, reaches the modulo value of 0x55, the counter overflows to 0x00 and continues counting. The real-time interrupt flag, RTIF, sets when the counter value changes from 0x55 to 0x00. A real-time interrupt is generated when RTIF is set, if RTIE is set.

13.5 Initialization/Application Information

This section provides example code to give some basic direction to a user on how to initialize and configure the RTC module. The example software is implemented in C language.

The example below shows how to implement time of day with the RTC using the 1-kHz clock source to achieve the lowest possible power consumption. Because the 1-kHz clock source is not as accurate as a crystal, software can be added for any adjustments. For accuracy without adjustments at the expense of additional power consumption, the external clock (ERCLK) or the internal clock (IRCLK) can be selected with appropriate prescaler and modulo values.

```
/* Initialize the elapsed time counters */
Seconds = 0;
Minutes = 0;
Hours = 0;
Days=0;
/* Configure RTC to interrupt every 1 second from 1-kHz clock source */
RTCMOD.byte = 0 \times 00;
RTCSC.byte = 0x1F;
Function Name : RTC_ISR
Notes : Interrupt service routine for RTC module.
                                              * * * * * * * * * * * * * * * * * /
#pragma TRAP_PROC
void RTC_ISR(void)
{
        /* Clear the interrupt flag */
```



Chapter 16 Timer/PWM Module (S08TPMV3)

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.

pulse width = 2 x (TPMxCnVH:TPMxCnVL)
period = 2 x (TPMxMODH:TPMxMODL); TPMxMODH:TPMxMODL=0x0001-0x7FFF

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.

The output compare value in the TPM channel registers (times 2) determines the pulse width (duty cycle) of the CPWM signal (Figure 16-16). If ELSnA=0, a compare occurred while counting up forces the CPWM output signal low and a compare occurred while counting down forces the output high. The counter counts up until it reaches the modulo setting in TPMxMODH:TPMxMODL, then counts down until it reaches zero. This sets the period equal to two times TPMxMODH:TPMxMODL.



Center-aligned PWM outputs typically produce less noise than edge-aligned PWMs because fewer I/O pin transitions are lined up at the same system clock edge. This type of PWM is also required for some types of motor drives.



Chapter 17 Development Support

17.1 Introduction

Development support systems in the HCS08 include the background debug controller (BDC) and the on-chip debug module (DBG). The BDC provides a single-wire debug interface to the target MCU that provides a convenient interface for programming the on-chip FLASH and other nonvolatile memories. The BDC is also the primary debug interface for development and allows non-intrusive access to memory data and traditional debug features such as CPU register modify, breakpoints, and single instruction trace commands.

In the HCS08 Family, address and data bus signals are not available on external pins (not even in test modes). Debug is done through commands fed into the target MCU via the single-wire background debug interface. The debug module provides a means to selectively trigger and capture bus information so an external development system can reconstruct what happened inside the MCU on a cycle-by-cycle basis without having external access to the address and data signals.

17.1.1 Forcing Active Background

The method for forcing active background mode depends on the specific HCS08 derivative. For the MC9S08SG32 Series, you can force active background after a power-on reset by holding the BKGD pin low as the device exits the reset condition. You can also force active background by driving BKGD low immediately after a serial background command that writes a one to the BDFR bit in the SBDFR register. Other causes of reset including an external pin reset or an internally generated error reset ignore the state of the BKGD pin and reset into normal user mode. If no debug pod is connected to the BKGD pin, the MCU will always reset into normal operating mode.



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)



17.3 On-Chip Debug System (DBG)

Because HCS08 devices do not have external address and data buses, the most important functions of an in-circuit emulator have been built onto the chip with the MCU. The debug system consists of an 8-stage FIFO that can store address or data bus information, and a flexible trigger system to decide when to capture bus information and what information to capture. The system relies on the single-wire background debug system to access debug control registers and to read results out of the eight stage FIFO.

The debug module includes control and status registers that are accessible in the user's memory map. These registers are located in the high register space to avoid using valuable direct page memory space.

Most of the debug module's functions are used during development, and user programs rarely access any of the control and status registers for the debug module. The one exception is that the debug system can provide the means to implement a form of ROM patching. This topic is discussed in greater detail in Section 17.3.6, "Hardware Breakpoints."

17.3.1 Comparators A and B

Two 16-bit comparators (A and B) can optionally be qualified with the R/W signal and an opcode tracking circuit. Separate control bits allow you to ignore R/W for each comparator. The opcode tracking circuitry optionally allows you to specify that a trigger will occur only if the opcode at the specified address is actually executed as opposed to only being read from memory into the instruction queue. The comparators are also capable of magnitude comparisons to support the inside range and outside range trigger modes. Comparators are disabled temporarily during all BDC accesses.

The A comparator is always associated with the 16-bit CPU address. The B comparator compares to the CPU address or the 8-bit CPU data bus, depending on the trigger mode selected. Because the CPU data bus is separated into a read data bus and a write data bus, the RWAEN and RWA control bits have an additional purpose, in full address plus data comparisons they are used to decide which of these buses to use in the comparator B data bus comparisons. If RWAEN = 1 (enabled) and RWA = 0 (write), the CPU's write data bus is used. Otherwise, the CPU's read data bus is used.

The currently selected trigger mode determines what the debugger logic does when a comparator detects a qualified match condition. A match can cause:

- Generation of a breakpoint to the CPU
- Storage of data bus values into the FIFO
- Starting to store change-of-flow addresses into the FIFO (begin type trace)
- Stopping the storage of change-of-flow addresses into the FIFO (end type trace)

17.3.2 Bus Capture Information and FIFO Operation

The usual way to use the FIFO is to setup the trigger mode and other control options, then arm the debugger. When the FIFO has filled or the debugger has stopped storing data into the FIFO, you would read the information out of it in the order it was stored into the FIFO. Status bits indicate the number of words of valid information that are in the FIFO as data is stored into it. If a trace run is manually halted by writing 0 to ARM before the FIFO is full (CNT = 1:0:0:0), the information is shifted by one position and



Field	Description
2 WS	 Wait or Stop Status — When the target CPU is in wait or stop mode, most BDC commands cannot function. However, the BACKGROUND command can be used to force the target CPU out of wait or stop and into active background mode where all BDC commands work. Whenever the host forces the target MCU into active background mode, the host should issue a READ_STATUS command to check that BDMACT = 1 before attempting other BDC commands. 0 Target CPU is running user application code or in active background mode (was not in wait or stop mode when background became active) 1 Target CPU is in wait or stop mode, or a BACKGROUND command was used to change from wait or stop to active background mode
1 WSF	 Wait or Stop Failure Status — This status bit is set if a memory access command failed due to the target CPU executing a wait or stop instruction at or about the same time. The usual recovery strategy is to issue a BACKGROUND command to get out of wait or stop mode into active background mode, repeat the command that failed, then return to the user program. (Typically, the host would restore CPU registers and stack values and re-execute the wait or stop instruction.) 0 Memory access did not conflict with a wait or stop instruction 1 Memory access command failed because the CPU entered wait or stop mode
0 DVF	 Data Valid Failure Status — This status bit is not used in the MC9S08SG32 Series because it does not have any slow access memory. 0 Memory access did not conflict with a slow memory access 1 Memory access command failed because CPU was not finished with a slow memory access

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

17.4.1.2 BDC Breakpoint Match Register (BDCBKPT)

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

17.4.2 System Background Debug Force Reset Register (SBDFR)

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



Chapter 17 Development Support



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

Figure 17-6. System Background Debug Force Reset Register (SBDFR)

Table 17-3. SBDFR Register Field Description

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

17.4.3 DBG Registers and Control Bits

The debug module includes nine bytes of register space for three 16-bit registers and three 8-bit control and status registers. These registers are located in the high register space of the normal memory map so they are accessible to normal application programs. These registers are rarely if ever accessed by normal user application programs with the possible exception of a ROM patching mechanism that uses the breakpoint logic.

17.4.3.1 Debug Comparator A High Register (DBGCAH)

This register contains compare value bits for the high-order eight bits of comparator A. This register is forced to 0x00 at reset and can be read at any time or written at any time unless ARM = 1.

17.4.3.2 Debug Comparator A Low Register (DBGCAL)

This register contains compare value bits for the low-order eight bits of comparator A. This register is forced to 0x00 at reset and can be read at any time or written at any time unless ARM = 1.

17.4.3.3 Debug Comparator B High Register (DBGCBH)

This register contains compare value bits for the high-order eight bits of comparator B. This register is forced to 0x00 at reset and can be read at any time or written at any time unless ARM = 1.

17.4.3.4 Debug Comparator B Low Register (DBGCBL)

This register contains compare value bits for the low-order eight bits of comparator B. This register is forced to 0x00 at reset and can be read at any time or written at any time unless ARM = 1.



17.4.3.8 Debug Trigger Register (DBGT)

This register can be read any time, but may be written only if ARM = 0, except bits 4 and 5 are hard-wired to 0s.



Figure 17-8. Debug Trigger Register (DBGT)

Table 17-5. DBGT Register Field Descriptions

Field	Description
7 TRGSEL	 Trigger Type — Controls whether the match outputs from comparators A and B are qualified with the opcode tracking logic in the debug module. If TRGSEL is set, a match signal from comparator A or B must propagate through the opcode tracking logic and a trigger event is only signalled to the FIFO logic if the opcode at the match address is actually executed. 0 Trigger on access to compare address (force) 1 Trigger if opcode at compare address is executed (tag)
6 BEGIN	 Begin/End Trigger Select — Controls whether the FIFO starts filling at a trigger or fills in a circular manner until a trigger ends the capture of information. In event-only trigger modes, this bit is ignored and all debug runs are assumed to be begin traces. 0 Data stored in FIFO until trigger (end trace) 1 Trigger initiates data storage (begin trace)
3:0 TRG[3:0]	Select Trigger Mode — Selects one of nine triggering modes, as described below. 0000 A-only 0001 A OR B 0010 A Then B 0011 Event-only B (store data) 0100 A then event-only B (store data) 0101 A AND B data (full mode) 0110 A AND NOT B data (full mode) 0111 Inside range: A ≤ address ≤ B 1000 Outside range: address < A or address > B $1001 - 1111$ (No trigger)



How to Reach Us:

Home Page: www.freescale.com

E-mail: support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor Technical Information Center, CH370 1300 N. Alma School Road Chandler, Arizona 85224 +1-800-521-6274 or +1-480-768-2130 support@freescale.com

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH Technical Information Center Schatzbogen 7 81829 Muenchen, Germany +44 1296 380 456 (English) +46 8 52200080 (English) +49 89 92103 559 (German) +33 1 69 35 48 48 (French) support@freescale.com

Japan:

Freescale Semiconductor Japan Ltd. Headquarters ARCO Tower 15F 1-8-1, Shimo-Meguro, Meguro-ku, Tokyo 153-0064 Japan 0120 191014 or +81 3 5437 9125 support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor China Ltd. Exchange Building 23F No. 118 Jianguo Road Chaoyang District Beijing 100022 China +86 10 5879 8000 support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center 1-800-441-2447 or 1-303-675-2140 Fax: 1-303-675-2150 LDCForFreescaleSemiconductor@hibbertgroup.com Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale[™] and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007-2010. All rights reserved.

