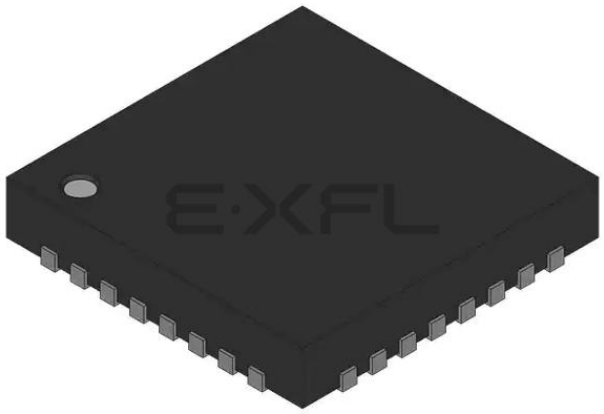

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-HVQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08gt8amfce

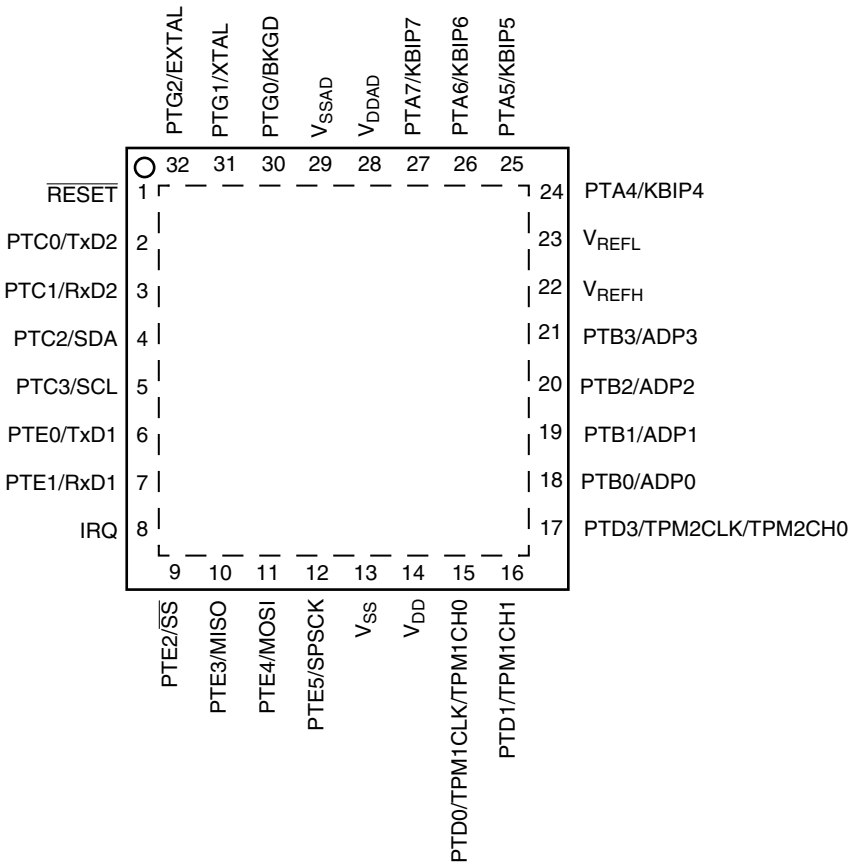

Figure 2-4. MC9S08GT16A/GT8A in 32-Pin QFN Package

2.3.5 IRQ — External Interrupt Request Pin

IRQ is a dedicated pin with both pullup and pulldown devices built in. This pin has no output capabilities. After a system reset, the IRQ pin is disabled and must be enabled before use. See Section 5.4.2, "IRQ — External Interrupt Request Pin" for more details.

For EMC-sensitive applications, an external RC filter is recommended on the IRQ pin. See Figure 2-5 for an example.

2.3.6 General-Purpose I/O and Peripheral Ports

The remaining 36 pins are shared among general-purpose I/O and on-chip peripheral functions such as timers and serial I/O systems. (Three of these pins are not bonded out on the 44-pin package, five are not bonded out on the 42-pin package, and 15 are not bonded out on the 32-pin package.) Immediately after reset, all 36 of these pins are configured as high-impedance general-purpose inputs with internal pullup devices disabled.

NOTE

To avoid extra current drain from floating input pins, the reset initialization routine in the application program should either enable on-chip pullup devices or change the direction of unused pins to outputs so the pins do not float.

For information about controlling these pins as general-purpose I/O pins, see Chapter 6, "Parallel Input/Output." For information about how and when on-chip peripheral systems use these pins, refer to the appropriate section from Table 2-1.

Table 2-1. Pin Sharing References

Port Pins	Alternate Function	Reference[1]
PTA7–PTA0	KBIP7–KBIP0	Chapter 7, "Keyboard Interrupt (S08KBIV1)"
PTB7–PTB0	ADP7–ADP0	Chapter 14, "Analog-to-Digital Converter (S08ATDV3)"
PTC7–PTC4		
PTC3–PTC2	SCL–SDA	Chapter 13, "Inter-Integrated Circuit (S08IICV1)"
PTC1–PTC0	RxD2–TxD2	Chapter 11, "Serial Communications Interface (S08SCIV1)"
PTD4–PTD3	TPM2CH1–TPM2CH0, TPM2CLK	Chapter 10, "Timer/PWM (S08TPMV2)"
PTD2–PTD0	TPM1CH2–TPM1CH0, TPM1CLK	Chapter 10, "Timer/PWM (S08TPMV2)"
PTE5 PTE4 PTE3 PTE2	SPSCK MISO MOSI $\overline{SS}$	Chapter 12, "Serial Peripheral Interface (S08SPIV3)"
PTE1–PTE0	RxD1–TxD1	Chapter 11, "Serial Communications Interface (S08SCIV1)"
PTG3		
PTG2–PTG1	EXTAL–XTAL	Chapter 9, "Internal Clock Generator (S08ICGV4)"
PTG0	BKGD/MS	Chapter 15, "Development Support"

[1] See this section for information about modules that share these pins.

makes the MCU secure. During development, whenever the FLASH is erased, it is good practice to immediately program the SEC00 bit to 0 in NVOPT so SEC01:SEC00 = 1:0. This would allow the MCU to remain unsecured after a subsequent reset.

The on-chip debug module cannot be enabled while the MCU is secure. The separate background debug controller can still be used for background memory access commands, but the MCU cannot enter active background mode except by holding BKGD/MS low at the rising edge of reset.

A user can choose to allow or disallow a security unlocking mechanism through an 8-byte backdoor security key. If the nonvolatile KEYEN bit in NVOPT/FOPT is 0, the backdoor key is disabled and there is no way to disengage security without completely erasing all FLASH locations. If KEYEN is 1, a secure user program can temporarily disengage security by:

1. Writing 1 to KEYACC in the FCNFG register. This makes the FLASH module interpret writes to the backdoor comparison key locations (NVBACKKEY through NVBACKKEY+7) as values to be compared against the key rather than as the first step in a FLASH program or erase command.
2. Writing the user-entered key values to the NVBACKKEY through NVBACKKEY+7 locations. These writes must be done in order, starting with the value for NVBACKKEY and ending with NVBACKKEY+7. STHX should not be used for these writes because these writes cannot be done on adjacent bus cycles. User software normally would get the key codes from outside the MCU system through a communication interface such as a serial I/O.
3. Writing 0 to KEYACC in the FCNFG register. If the 8-byte key that was just written matches the key stored in the FLASH locations, SEC01:SEC00 are automatically changed to 1:0 and security will be disengaged until the next reset.

The security key can be written only from RAM, so it cannot be entered through background commands without the cooperation of a secure user program.

The backdoor comparison key (NVBACKKEY through NVBACKKEY+7) is located in FLASH memory locations in the nonvolatile register space so users can program these locations exactly as they would program any other FLASH memory location. The nonvolatile registers are in the same 512-byte block of FLASH as the reset and interrupt vectors, so block protecting that space also block protects the backdoor comparison key. Block protects cannot be changed from user application programs, so if the vector space is block protected, the backdoor security key mechanism cannot permanently change the block protect, security settings, or the backdoor key.

Security can always be disengaged through the background debug interface by performing these steps:

1. Disable any block protections by writing FPROT. FPROT can be written only with background debug commands, not from application software.
2. Mass erase FLASH, if necessary.
3. Blank check FLASH. Provided FLASH is completely erased, security is disengaged until the next reset.

   To avoid returning to secure mode after the next reset, program NVOPT so SEC01:SEC00 = 1:0.

5.7.7 System Power Management Status and Control 1 Register (SPMSC1)

	7	6	5	4	3	2	1	0
R	LVDF	0	LVDIE	LVDRE Note (1)	LVDSE Note (1)	LVDE Note (1)	0	0
W		LVDACK						
Reset	0	0	0	1	1	1	0	0

= Unimplemented or Reserved

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

Figure 5-9. System Power Management Status and Control 1 Register (SPMSC1)

Table 5-10. SPMSC1 Field Descriptions

Field	Description
7 LVDF	**Low-Voltage Detect Flag** — Provided LVDE = 1, this read-only status bit indicates a low-voltage detect event.
6 LVDACK	**Low-Voltage Detect Acknowledge** — This write-only bit is used to acknowledge low voltage detection errors (write 1 to clear LVDF). Reads always return 0.
5 LVDIE	**Low-Voltage Detect Interrupt Enable** — This read/write bit enables hardware interrupt requests for LVDF. 0 Hardware interrupt disabled (use polling). 1 Request a hardware interrupt when LVDF = 1.
4 LVDRE	**Low-Voltage Detect Reset Enable** — This read/write bit enables LVDF events to generate a hardware reset (provided LVDE = 1). 0 LVDF does not generate hardware resets. 1 Force an MCU reset when LVDF = 1.
3 LVDSE	**Low-Voltage Detect Stop Enable** — Provided LVDE = 1, this read/write bit determines whether the low-voltage detect function operates when the MCU is in stop mode. 0 Low-voltage detect disabled during stop mode. 1 Low-voltage detect enabled during stop mode.
2 LVDE	**Low-Voltage Detect Enable** — This read/write bit enables low-voltage detect logic and qualifies the operation of other bits in this register. 0 LVD logic disabled. 1 LVD logic enabled.

Chapter 6
Parallel Input/Output

6.1 Introduction

This section explains software controls related to parallel input/output (I/O). The MC9S08GT16A/GT8A has six I/O ports which include a total of up to 39 general-purpose I/O pins (one pin, PTG0, is output only). See Chapter 2, "Pins and Connections," for more information about the logic and hardware aspects of these pins.

Many of these pins are shared with on-chip peripherals such as timer systems, external interrupts, or keyboard interrupts. When these other modules are not controlling the port pins, they revert to general-purpose I/O control. For each I/O pin, a port data bit provides access to input (read) and output (write) data, a data direction bit controls the direction of the pin, and a pullup enable bit enables an internal pullup device (provided the pin is configured as an input), and a slew rate control bit controls the rise and fall times of the pins.

Pins that are not used in the application must be terminated. This prevents excess current caused by floating inputs and enhances immunity during noise or transient events. Termination methods include:

- Configuring unused pins as outputs driving high or low
- Configuring unused pins as inputs and using internal or external pullups

Never connect unused pins to V_{DD} or V_{SS}.

NOTE

Not all general-purpose I/O pins are available on all packages. To avoid extra current drain from floating input pins, the user's reset initialization routine in the application program should either enable on-chip pullup devices or change the direction of unconnected pins to outputs so the pins do not float.

6.1.1 Features

Parallel I/O features, depending on package choice, include:

- A total of 39 general-purpose I/O pins in six ports (PTG0 is output only)
- High-current drivers on port C pins
- Hysteresis input buffers
- Software-controlled pullups on each input pin
- Software-controlled slew rate output buffers
- Eight port A pins shared with KBI

6.5.6 Port G Registers (PTGD, PTGPE, PTGSE, and PTGDD)

Port G includes four general-purpose I/O pins that are shared with BKGD/MS function and the oscillator or external clock pins. Port G pins used as general-purpose I/O pins are controlled by the port G data (PTGD), data direction (PTGDD), pullup enable (PTGPE), and slew rate control (PTGSE) registers.

Port pin PTG0, while in reset, defaults to the BKGD/MS pin. After the MCU is exits reset, PTG0 can be configured to be a general-purpose output pin. When BKGD/MS takes control of PTG0, the corresponding PTGDD, PTGPE, and PTGPSE bits are ignored.

Port pins PTG1 and PTG2 can be configured to be oscillator or external clock pins. When the oscillator takes control of a port G pin, the corresponding PTGD, PTGDD, PTGSE, and PTGPE bits are ignored.

Reads of PTGD will return the logic value of the corresponding pin, provided PTGDD is 0.

	7	6	5	4	3	2	1	0
R	0	0	0	0	PTGD3	PTGD2	PTGD1	PTGD0
W								
Reset	0	0	0	0	0	0	0	0

Figure 6-28. Port PTG Data Register (PTGD)

Table 6-21. PTGD Field Descriptions

Field	Description
3:0 PTGD[3:0]	**Port PTG Data Register Bits —** For port G pins that are inputs, reads return the logic level on the pin. For port G 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 G pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTGD 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	0	0	0	0	PTGPE3	PTGPE2	PTGPE1	PTGPE0
W								
Reset	0	0	0	0	0	0	0	0

Figure 6-29. Pullup Enable for Port G (PTGPE)

Table 6-22. PTGPE Field Descriptions

Field	Description
3:0 PTGPE[3:0]	**Pullup Enable for Port G Bits** — For port G pins that are inputs, these read/write control bits determine whether internal pullup devices are enabled. For port G 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.

7.2.1 KBI Status and Control Register (KBISC)

	7	6	5	4	3	2	1	0
R	KBEDG7	KBEDG6	KBEDG5	KBEDG4	KBF	0	KBIE	KBIMOD
W						KBACK		
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 7-4. KBI Status and Control Register (KBISC)

Table 7-1. KBISC Register Field Descriptions

Field	Description
7:4 KBEDG[7:4]	**Keyboard Edge Select for KBI Port Bits** — Each of these read/write bits selects the polarity of the edges and/or levels that are recognized as trigger events on the corresponding KBI port pin when it is configured as a keyboard interrupt input (KBIPEn = 1). Also see the KBIMOD control bit, which determines whether the pin is sensitive to edges-only or edges and levels. 0 Falling edges/low levels 1 Rising edges/high levels
3 KBF	**Keyboard Interrupt Flag** — This read-only status flag is set whenever the selected edge event has been detected on any of the enabled KBI port pins. This flag is cleared by writing a 1 to the KBACK control bit. The flag will remain set if KBIMOD = 1 to select edge-and-level operation and any enabled KBI port pin remains at the asserted level. KBF can be used as a software pollable flag (KBIE = 0) or it can generate a hardware interrupt request to the CPU (KBIE = 1). 0 No KBI interrupt pending 1 KBI interrupt pending
2 KBACK	**Keyboard Interrupt Acknowledge** — This write-only bit (reads always return 0) is used to clear the KBF status flag by writing a 1 to KBACK. When KBIMOD = 1 to select edge-and-level operation and any enabled KBI port pin remains at the asserted level, KBF is being continuously set so writing 1 to KBACK does not clear the KBF flag.
1 KBIE	**Keyboard Interrupt Enable** — This read/write control bit determines whether hardware interrupts are generated when the KBF status flag equals 1. When KBIE = 0, no hardware interrupts are generated, but KBF can still be used for software polling. 0 KBF does not generate hardware interrupts (use polling) 1 KBI hardware interrupt requested when KBF = 1
KBIMOD	**Keyboard Detection Mode** — This read/write control bit selects either edge-only detection or edge-and-level detection. KBI port bits 3 through 0 can detect falling edges-only or falling edges and low levels. KBI port bits 7 through 4 can be configured to detect either: • Rising edges-only or rising edges and high levels (KBEDGn = 1) • Falling edges-only or falling edges and low levels (KBEDGn = 0) 0 Edge-only detection 1 Edge-and-level detection

Chapter 9
Internal Clock Generator (S08ICGV4)

9.1 Introduction

The MC9S08GT16A/GT8A microcontroller provides one internal clock generation (ICG) module to create the system bus frequency. All functions described in this section are available on the MC9S08GT16A/GT8A microcontroller. The EXTAL and XTAL pins share port G bits 2 and 1, respectively. Analog supply lines V_{DDA} and V_{SSA} are internally derived from the MCU's V_{DD} and V_{SS} pins. Electrical parametric data for the ICG may be found in Appendix A, "Electrical Characteristics."

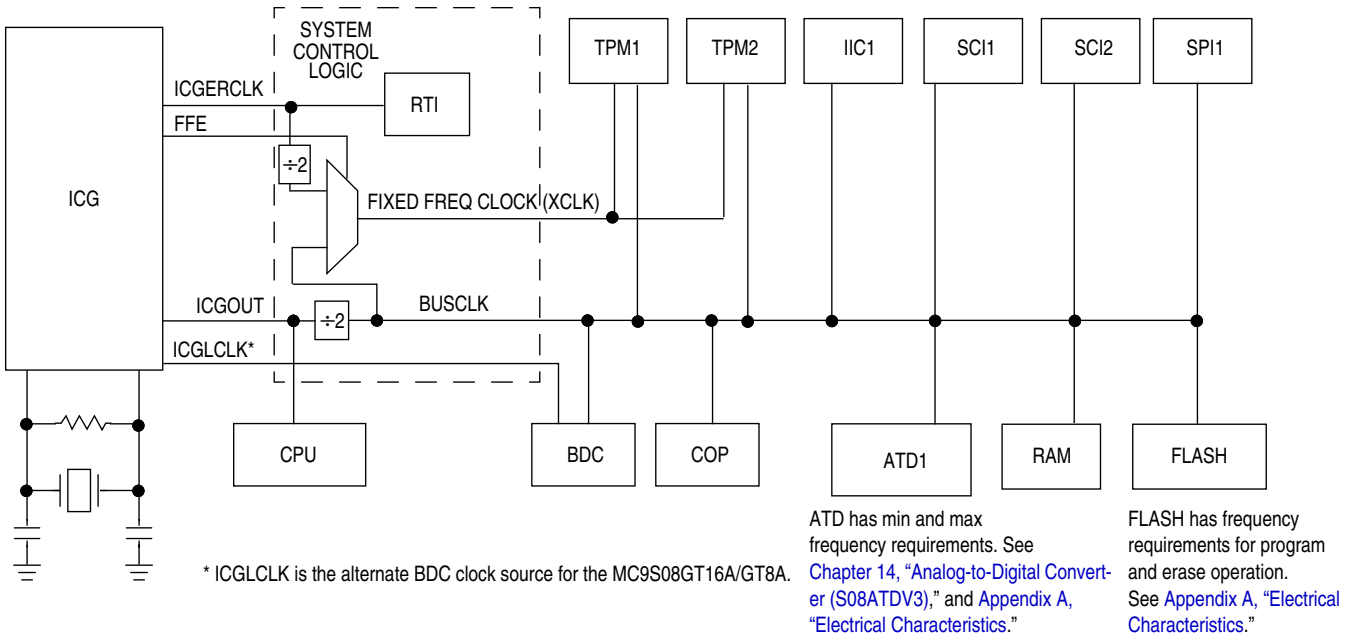

Figure 9-1. System Clock Distribution Diagram

NOTE

Freescale Semiconductor programs a factory trim value for ICGTRM into the FLASH location $FFBE (NVICGTRM). Leaving this address for the ICGTRM value also allows debugger and programmer vendors to perform a manual trim operation and store the resultant ICGTRM value into NVICGTRM for users to access at a later time. The value in NVICGTRM is not automatically loaded and therefore must be copied into ICGTTRM by user code.

- Digitally-controlled oscillator (DCO) preserves previous frequency settings, allowing fast frequency lock when recovering from stop3 mode
- DCO will maintain operating frequency during a loss or removal of reference clock
- Post-FLL divider selects 1 of 8 bus rate divisors (/1 through /128)
- Separate self-clocked source for real-time interrupt
- Trimmable internal clock source supports SCI communications without additional external components
- Automatic FLL engagement after lock is acquired
- External oscillator selectable for low power or high gain

9.1.2 Modes of Operation

This is a high-level description only. Detailed descriptions of operating modes are contained in Section 9.4, "Functional Description."

- Mode 1 — Off

  The output clock, ICGOUT, is static. This mode may be entered when the STOP instruction is executed.
- Mode 2 — Self-clocked (SCM)

  Default mode of operation that is entered immediately after reset. The ICG's FLL is open loop and the digitally controlled oscillator (DCO) is free running at a frequency set by the filter bits.
- Mode 3 — FLL engaged internal (FEI)

  In this mode, the ICG's FLL is used to create frequencies that are programmable multiples of the internal reference clock.
  - FLL engaged internal unlocked is a transition state that occurs while the FLL is attempting to lock. The FLL DCO frequency is off target and the FLL is adjusting the DCO to match the target frequency.
  - FLL engaged internal locked is a state that occurs when the FLL detects that the DCO is locked to a multiple of the internal reference.
- Mode 4 — FLL bypassed external (FBE)

  In this mode, the ICG is configured to bypass the FLL and use an external clock as the clock source.
- Mode 5 — FLL engaged external (FEE)

  The ICG's FLL is used to generate frequencies that are programmable multiples of the external clock reference.
  - FLL engaged external unlocked is a transition state that occurs while the FLL is attempting to lock. The FLL DCO frequency is off target and the FLL is adjusting the DCO to match the target frequency.
  - FLL engaged external locked is a state which occurs when the FLL detects that the DCO is locked to a multiple of the internal reference.

9.3.4 ICG Status Register 2 (ICGS2)

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	DCOS
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 9-9. ICG Status Register 2 (ICGS2)

Table 9-4. ICGS2 Register Field Descriptions

Field	Description
0 DCOS	**DCO Clock Stable** — The DCOS bit is set when the DCO clock (ICG2DCLK) is stable, meaning the count error has not changed by more than n_{unlock} for two consecutive samples and the DCO clock is not static. This bit is used when exiting off state if CLKS = X1 to determine when to switch to the requested clock mode. It is also used in self-clocked mode to determine when to start monitoring the DCO clock. This bit is cleared upon entering the off state. 0 DCO clock is unstable. 1 DCO clock is stable.

9.3.5 ICG Filter Registers (ICGFLTU, ICGFLTL)

	7	6	5	4	3	2	1	0
R	0	0	0	0	FLT			
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 9-10. ICG Upper Filter Register (ICGFLTU)

Table 9-5. ICGFLTU Register Field Descriptions

Field	Description
3:0 FLT	**Filter Value** — The FLT bits indicate the current filter value, which controls the DCO frequency. The FLT bits are read only except when the CLKS bits are programmed to self-clocked mode (CLKS = 00). In self-clocked mode, any write to ICGFLTU updates the current 12-bit filter value. Writes to the ICGFLTU register will not affect FLT if a previous latch sequence is not complete.

Table 11-4. SCIxC2 Register Field Descriptions (continued)

Field	Description
1 RWU	**Receiver Wakeup Control** — This bit can be written to 1 to place the SCI receiver in a standby state where it waits for automatic hardware detection of a selected wakeup condition. The wakeup condition is either an idle line between messages (WAKE = 0, idle-line wakeup), or a logic 1 in the most significant data bit in a character (WAKE = 1, address-mark wakeup). Application software sets RWU and (normally) a selected hardware condition automatically clears RWU. Refer to Section 11.3.3.2, "Receiver Wakeup Operation," for more details. 0 Normal SCI receiver operation. 1 SCI receiver in standby waiting for wakeup condition.
0 SBK	**Send Break** — Writing a 1 and then a 0 to SBK queues a break character in the transmit data stream. Additional break characters of 10 or 11 bit times of logic 0 are queued as long as SBK = 1. Depending on the timing of the set and clear of SBK relative to the information currently being transmitted, a second break character may be queued before software clears SBK. Refer to Section 11.3.2.1, "Send Break and Queued Idle," for more details. 0 Normal transmitter operation. 1 Queue break character(s) to be sent.

11.2.4 SCI Status Register 1 (SCIxS1)

This register has eight read-only status flags. Writes have no effect. Special software sequences (which do not involve writing to this register) are used to clear these status flags.

	7	6	5	4	3	2	1	0
R	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
W								
Reset	1	1	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 11-8. SCI Status Register 1 (SCIxS1)

Table 11-5. SCIxS1 Register Field Descriptions

Field	Description
7 TDRE	**Transmit Data Register Empty Flag** — TDRE is set immediately after 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 SCIxS1 with TDRE = 1 and then write to the SCI data register (SCIxD). 0 Transmit data register (buffer) full. 1 Transmit data register (buffer) empty.
6 TC	**Transmission Complete Flag** — TC is set immediately after 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 SCIxS1 with TC = 1 and then doing one of the following three things: • Write to the SCI data register (SCIxD) to transmit new data • Queue a preamble by changing TE from 0 to 1 • Queue a break character by writing 1 to SBK in SCIxC2

Table 11-5. SCIxS1 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 (SCIxD). To clear RDRF, read SCIxS1 with RDRF = 1 and then read the SCI data register (SCIxD). 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 SCIxS1 with IDLE = 1 and then read the SCI data register (SCIxD). 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 SCIxD yet. In this case, the new character (and all associated error information) is lost because there is no room to move it into SCIxD. To clear OR, read SCIxS1 with OR = 1 and then read the SCI data register (SCIxD). 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 SCIxS1 and then read the SCI data register (SCIxD). 0 No noise detected. 1 Noise detected in the received character in SCIxD.
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 SCIxS1 with FE = 1 and then read the SCI data register (SCIxD). 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 SCIxS1 and then read the SCI data register (SCIxD). 0 No parity error. 1 Parity error.

13.1.3 Block Diagram

Figure 13-2 is a block diagram of the IIC.

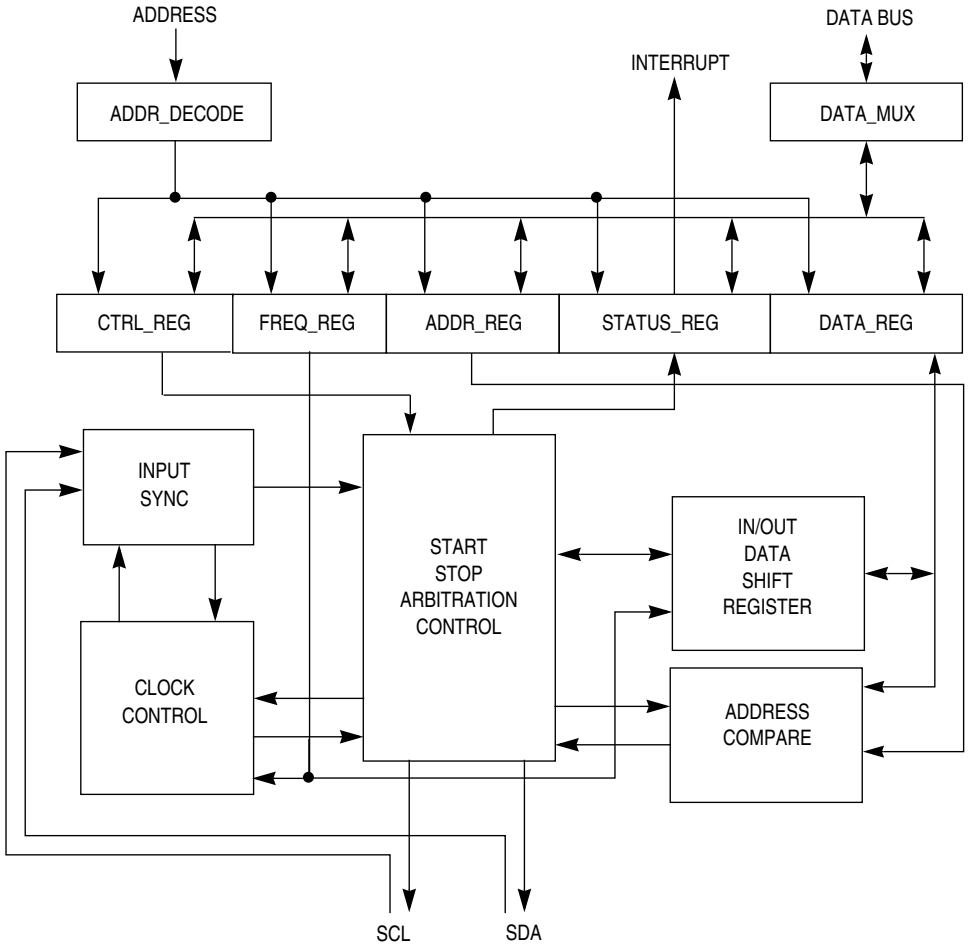

Figure 13-2. IIC Functional Block Diagram

13.2 External Signal Description

This section describes each user-accessible pin signal.

13.2.1 SCL — Serial Clock Line

The bidirectional SCL is the serial clock line of the IIC system.

13.2.2 SDA — Serial Data Line

The bidirectional SDA is the serial data line of the IIC system.

13.3 Register Definition

This section consists of the IIC register descriptions in address order.

13.3.4 IIC Status Register (IICS)

	7	6	5	4	3	2	1	0
R	TCF	IAAS	BUSY	ARBL	0	SRW	IICIF	RXAK
W								
Reset	1	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 13-6. IIC Status Register (IICS)

Table 13-5. IICS Register Field Descriptions

Field	Description
7 TCF	**Transfer Complete Flag** — This bit is set on the completion of a byte transfer. Note that this bit is only valid during or immediately following a transfer to the IIC module or from the IIC module.The TCF bit is cleared by reading the IICD register in receive mode or writing to the IICD in transmit mode. 0 Transfer in progress. 1 Transfer complete.
6 IAAS	**Addressed as a Slave** — The IAAS bit is set when the calling address matches the programmed slave address. Writing the IICC register clears this bit. 0 Not addressed. 1 Addressed as a slave.
5 BUSY	**Bus Busy** — The BUSY bit indicates the status of the bus regardless of slave or master mode. The BUSY bit is set when a START signal is detected and cleared when a STOP signal is detected. 0 Bus is idle. 1 Bus is busy.
4 ARBL	**Arbitration Lost** — This bit is set by hardware when the arbitration procedure is lost. The ARBL bit must be cleared by software, by writing a one to it. 0 Standard bus operation. 1 Loss of arbitration.
2 SRW	**Slave Read/Write** — When addressed as a slave the SRW bit indicates the value of the R/W command bit of the calling address sent to the master. 0 Slave receive, master writing to slave. 1 Slave transmit, master reading from slave.
1 IICIF	**IIC Interrupt Flag** — The IICIF bit is set when an interrupt is pending. This bit must be cleared by software, by writing a one to it in the interrupt routine. One of the following events can set the IICIF bit: • One byte transfer completes • Match of slave address to calling address • Arbitration lost 0 No interrupt pending. 1 Interrupt pending.
0 RXAK	**Receive Acknowledge** — When the RXAK bit is low, it indicates an acknowledge signal has been received after the completion of one byte of data transmission on the bus. If the RXAK bit is high it means that no acknowledge signal is detected. 0 Acknowledge received. 1 No acknowledge received.

14.5 Resets

The ATD module is reset on system reset. If the system reset signal is activated, the ATD registers are initialized back to their reset state and the ATD module is powered down. This occurs as a function of the register file initialization; the reset definition of the ATDPU bit (power down bit) is zero or disabled.

The MCU places the module back into an initialized state. If the module is performing a conversion, the current conversion is terminated, the conversion complete flag is cleared, and the SAR register bits are cleared. Any pending interrupts are also cancelled. Note that the control, test, and status registers are initialized on reset; the initialized register state is defined in the register description section of this specification.

Enabling the module (using the ATDPU bit) does not cause the module to reset since the register file is not initialized. Finally, writing to control register ATDC does not cause the module to reset; the current conversion will be terminated.

14.6 Interrupts

The ATD module originates interrupt requests and the MCU handles or services these requests. Details on how the ATD interrupt requests are handled can be found in the resets and interrupts chapter of this data sheet.

The ATD interrupt function is enabled by setting the ATDIE bit in the ATDSC register. When the ATDIE bit is set, an interrupt is generated at the end of an ATD conversion and the ATD result registers (ATDRH and ATDRL) contain the result data generated by the conversion. If the interrupt function is disabled (ATDIE = 0), then the CCF flag must be polled to determine when a conversion is complete.

The interrupt will remain pending as long as the CCF flag is set. The CCF bit is cleared whenever the ATD status and control (ATDSC) register is written. The CCF bit is also cleared whenever the ATD result registers (ATDRH or ATDRL) are read.

Table 14-8. Interrupt Summary

Interrupt	Local Enable	Description
CCF	ATDIE	Conversion complete

Figure 15-4 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.

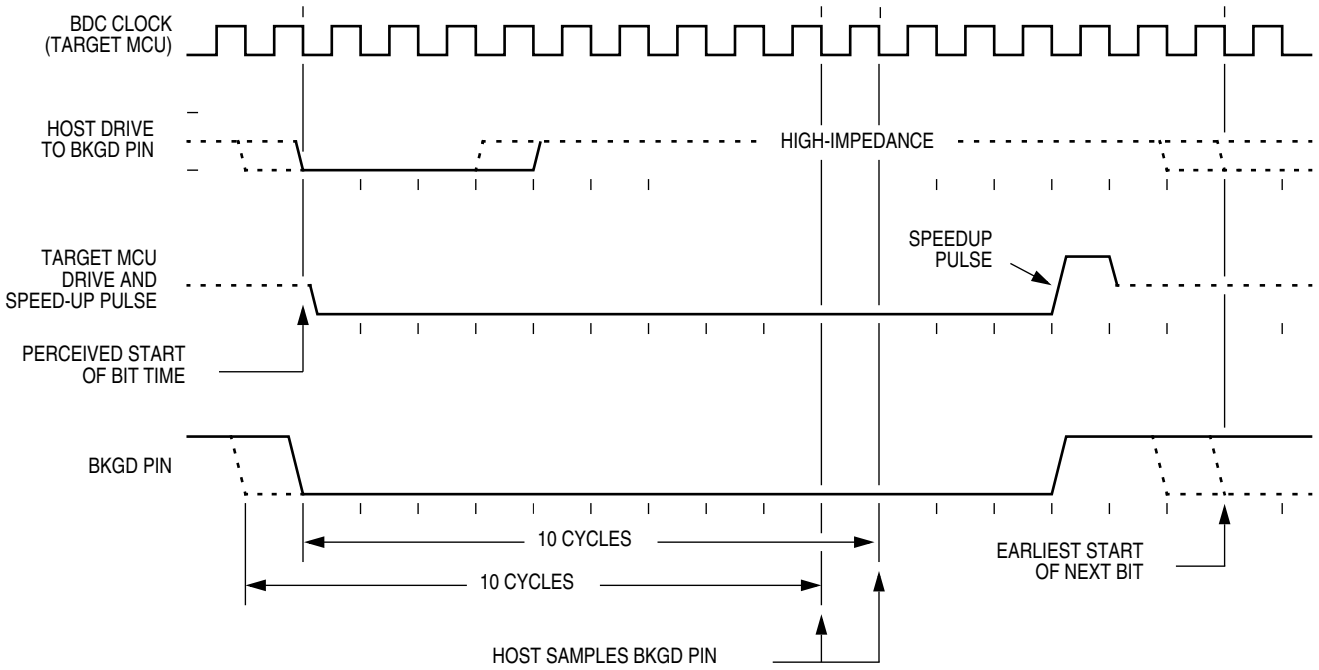

Figure 15-4. BDM Target-to-Host Serial Bit Timing (Logic 0)

15.3.6 Hardware Breakpoints

The BRKEN control bit in the DBGC register may be set to 1 to allow any of the trigger conditions described in Section 15.3.5, “Trigger Modes,” to be used to generate a hardware breakpoint request to the CPU. TAG in DBGC controls whether the breakpoint request will be treated as a tag-type breakpoint or a force-type breakpoint. A tag breakpoint causes the current opcode to be marked as it enters the instruction queue. If a tagged opcode reaches the end of the pipe, the CPU executes a BGND instruction to go to active background mode rather than executing the tagged opcode. A force-type breakpoint causes the CPU to finish the current instruction and then go to active background mode.

If the background mode has not been enabled (ENBDM = 1) by a serial WRITE_CONTROL command through the BKGD pin, the CPU will execute an SWI instruction instead of going to active background mode.

15.4 Register Definition

This section contains the descriptions of the BDC and DBG registers and control bits.

Refer to the high-page register summary in the device overview chapter of this data sheet for the absolute address assignments for all DBG 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.

15.4.1 BDC Registers and Control Bits

The BDC has two registers:

- The BDC status and control register (BDCSCR) is an 8-bit register containing control and status bits for the background debug controller.
- The BDC breakpoint match register (BDCBKPT) holds a 16-bit breakpoint match address.

These registers are accessed with dedicated serial BDC commands and are not located in the memory space of the target MCU (so they do not have addresses and cannot be accessed by user programs).

Some of the bits in the BDCSCR have write limitations; otherwise, these registers may be read or written at any time. For example, the ENBDM control bit may not be written while the MCU is in active background mode. (This prevents the ambiguous condition of the control bit forbidding active background mode while the MCU is already in active background mode.) Also, the four status bits (BDMACT, WS, WSF, and DVF) are read-only status indicators and can never be written by the WRITE_CONTROL serial BDC command. The clock switch (CLKSW) control bit may be read or written at any time.

Table A-6. DC Characteristics (Sheet 1 of 2)
(Temperature Range = –40 to 125°C Ambient)

C	Parameter	Symbol	Min	Typical[1]	Max	Unit
	Minimum RAM retention supply voltage applied to V_{DD}	V_{RAM}	1.0[2]		—	V
P	Low-voltage detection threshold — high range (V_{DD} falling) (V_{DD} rising)	V_{LVDH}	 2.08 2.16	 2.1 2.19	 2.2 2.27	V
P	Low-voltage detection threshold — low range (V_{DD} falling) (V_{DD} rising)	V_{LVDL}	 1.80 1.88	 1.82 1.90	 1.91 1.99	V
P	Low-voltage warning threshold — high range (V_{DD} falling) (V_{DD} rising)	V_{LVWH}	 2.35 2.35	 2.40 2.40	 2.5 2.5	V
P	Low-voltage warning threshold — low range (V_{DD} falling) (V_{DD} rising)	V_{LVWL}	 2.08 2.16	 2.1 2.19	 2.2 2.27	V
P	Power on reset (POR) re-arm voltage[(2)] Mode = stop Mode = run and Wait	V_{Rearm}	 0.20 0.50	 0.30 0.80	 0.40 1.2	V
P	Input high voltage (V_{DD} > 2.3 V) (all digital inputs)	V_{IH}	$0.70 \times V_{DD}$		—	V
P	Input high voltage (1.8 V ≤ V_{DD} ≤ 2.3 V) (all digital inputs)	V_{IH}	$0.85 \times V_{DD}$		—	V
P	Input low voltage (V_{DD} > 2.3 V) (all digital inputs)	V_{IL}	—		$0.35 \times V_{DD}$	V
P	Input low voltage (1.8 V ≤ V_{DD} ≤ 2.3 V) (all digital inputs)	V_{IL}	—		$0.30 \times V_{DD}$	V
Y	Input hysteresis (all digital inputs)	V_{hys}	$0.06 \times V_{DD}$		—	V
P	Input leakage current (per pin) V_{In} = V_{DD} or V_{SS}, all input only pins	$\lvert I_{In} \rvert$	—	0.025	1.0	µA
P	High impedance (off-state) leakage current (per pin) V_{In} = V_{DD} or V_{SS}, all input/output	$\lvert I_{OZ} \rvert$	—	0.025	1.0	µA
P	Internal pullup and pulldown resistors[3] (all port pins and IRQ)	R_{PU}	17.5		52.5	kΩ
p	Internal pulldown resistors (Port A4–A7 and IRQ)	R_{PD}	17.5		52.5	kΩ
P	Output high voltage (V_{DD} ≥ 1.8 V) I_{OH} = –2 mA (ports A, B, D, E, and G)	V_{OH}	V_{DD} – 0.5		—	V
P	Output high voltage (port C) I_{OH} = –10 mA (V_{DD} ≥ 2.7 V) I_{OH} = –6 mA (V_{DD} ≥ 2.3 V) I_{OH} = –3 mA (V_{DD} ≥ 1.8 V)		V_{DD} – 0.5		 — —	
D	Maximum total I_{OH} for all port pins	$\lvert I_{OHT} \rvert$	—		60	mA

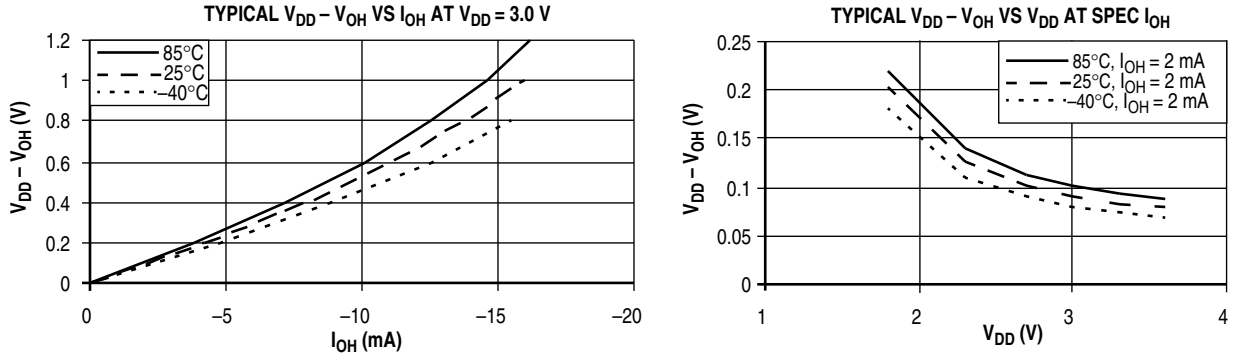

Figure A-5. Typical High-Side (Source) Characteristics (Ports A, B, D, E, and G)

A.7 Supply Current Characteristics

Table A-7. Supply Current Characteristics

Parameter	Symbol	V_{DD} (V)	Typical[1]	Max[2]	Temp. (°C)
Run supply current[3] measured at (CPU clock = 2 MHz, f_{Bus} = 1 MHz)	RI_{DD}	3	0.8 mA	1.3 mA[4]	125
		2	0.66 mA	1.0 mA(4)	125
Run supply current (3) measured at (CPU clock = 16 MHz, f_{Bus} = 8 MHz)	RI_{DD}	3	4.3 mA	7.0 mA[5]	125
		2	3.3 mA	4.5 mA(4)	125
Stop1 mode supply current	$S1I_{DD}$	3	25 nA	0.6 µA(4) 1.8 µA(4) 4.0 µA(5) 13 µA(5)	55 70 85 125
		2	20 nA	500 nA(4) 1.5 µA(4) 3.3 µA(4) 10 µA(4)	55 70 85 125
Stop2 mode supply current	$S2I_{DD}$	3	550 nA	3.0 µA(4) 5.5 µA(4) 11 µA(5) 20 µA(5)	55 70 85 125
		2	400 nA	2.4 µA(4) 5.0 µA(4) 9.5 µA(4) 17 µA(4)	55 70 85 125