



Welcome to [E-XFL.COM](http://E-XFL.COM)

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

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

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

#### Details

Product Status	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I <sup>2</sup> C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-QFP (10x10)
Purchase URL	<a href="https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s08gt60acfber">https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s08gt60acfber</a>

Section Number	Title	Page
5.8.2	System Reset Status Register (SRS) .....	74
5.8.3	System Background Debug Force Reset Register (SBDFR) .....	75
5.8.4	System Options Register (SOPT) .....	76
5.8.5	System Device Identification Register (SDIDH, SDIDL) .....	77
5.8.6	System Real-Time Interrupt Status and Control Register (SRTISC) .....	78
5.8.7	System Power Management Status and Control 1 Register (SPMSC1) .....	79
5.8.8	System Power Management Status and Control 2 Register (SPMSC2) .....	80

## Chapter 6 Parallel Input/Output

6.1	Introduction .....	81
6.2	Features .....	83
6.3	Pin Descriptions .....	83
6.3.1	Port A and Keyboard Interrupts .....	83
6.3.2	Port B and Analog to Digital Converter Inputs .....	84
6.3.3	Port C and SCI2, IIC, and High-Current Drivers .....	84
6.3.4	Port D, TPM1 and TPM2 .....	85
6.3.5	Port E, SCI1, and SPI .....	85
6.3.6	Port F and High-Current Drivers .....	86
6.3.7	Port G, BKGD/MS, and Oscillator .....	86
6.4	Parallel I/O Controls .....	87
6.4.1	Data Direction Control .....	87
6.4.2	Internal Pullup Control .....	87
6.4.3	Slew Rate Control .....	87
6.5	Stop Modes .....	88
6.6	Parallel I/O Registers and Control Bits .....	88
6.6.1	Port A Registers (PTAD, PTAPE, PTASE, and PTADD) .....	88
6.6.2	Port B Registers (PTBD, PTBPE, PTBSE, and PTBDD) .....	91
6.6.3	Port C Registers (PTCD, PTCPE, PTCSE, and PTCDD) .....	93
6.6.4	Port D Registers (PTDD, PTDPE, PTDSE, and PTDDD) .....	95
6.6.5	Port E Registers (PTED, PTEPE, PTESE, and PTEDD) .....	97
6.6.6	Port F Registers (PTFD, PTFPE, PTFSE, and PTFDD) .....	99
6.6.7	Port G Registers (PTGD, PTGPE, PTGSE, and PTGDD) .....	100

## Chapter 7 Internal Clock Generator (S08ICGV2)

7.1	Introduction .....	105
7.1.1	Features .....	106
7.1.2	Modes of Operation .....	107
7.2	Oscillator Pins .....	107
7.2.1	EXTAL— External Reference Clock / Oscillator Input .....	107
7.2.2	XTAL— Oscillator Output .....	107

### 3.6.4 Active BDM Enabled in Stop Mode

Entry into the active background mode from run mode is enabled if the ENBDM bit in BDCSCR is set. This register is described in the [Chapter 15, “Development Support,”](#) section of this data sheet. If ENBDM is set when the CPU executes a STOP instruction, the system clocks to the background debug logic remain active when the MCU enters stop mode so background debug communication is still possible. In addition, the voltage regulator does not enter its low-power standby state but maintains full internal regulation. If the user attempts to enter either stop1 or stop2 with ENBDM set, the MCU will instead enter stop3.

Most background commands are not available in stop mode. The memory-access-with-status commands do not allow memory access, but they report an error indicating that the MCU is in either stop or wait mode. The BACKGROUND command can be used to wake the MCU from stop and enter active background mode if the ENBDM bit is set. After the device enters background debug mode, all background commands are available. The table below summarizes the behavior of the MCU in stop when entry into the background debug mode is enabled.

**Table 3-2. BDM Enabled Stop Mode Behavior**

Mode	PDC	PPDC	CPU, Digital Peripherals, Flash	RAM	ICG	ATD	Regulator	I/O Pins	RTI <sup>1</sup>
Stop3	Don't care	Don't care	Standby	Standby	Active	Disabled <sup>2</sup>	Active	States held	Optionally on

<sup>1</sup> The 1 kHz internal RTI clock is not available in stop3 with active BDM enabled.

<sup>2</sup> Either ATD stop mode or power-down mode depending on the state of ATDPU.

### 3.6.5 LVD Enabled in Stop Mode

The LVD system is capable of generating either an interrupt or a reset when the supply voltage drops below the LVD voltage. If the LVD is enabled in stop by setting the LVDE and the LVDSE bits in SPMSC1 when the CPU executes a STOP instruction, then the voltage regulator remains active during stop mode. If the user attempts to enter either stop1 or stop2 with the LVD enabled for stop (LVDSE = 1), the MCU will instead enter stop3. The table below summarizes the behavior of the MCU in stop when the LVD is enabled.

**Table 3-3. LVD Enabled Stop Mode Behavior**

Mode	PDC	PPDC	CPU, Digital Peripherals, Flash	RAM	ICG	ATD	Regulator	I/O Pins	RTI
Stop3	Don't care	Don't care	Standby	Standby	Standby	Disabled <sup>1</sup>	Active	States held	Optionally on

<sup>1</sup> Either ATD stop mode or power-down mode depending on the state of ATDPU.

### 3.6.6 On-Chip Peripheral Modules in Stop Modes

When the MCU enters any stop mode, system clocks to the internal peripheral modules are stopped. Even in the exception case (ENBDM = 1), where clocks to the background debug logic continue to operate,

### 4.1.1 Reset and Interrupt Vector Assignments

Table 4-1 shows address assignments for reset and interrupt vectors. The vector names shown in this table are the labels used in the Freescale-provided equate file for the MC9S08GBxxA/GTxxA. For more details about resets, interrupts, interrupt priority, and local interrupt mask controls, refer to Chapter 5, “Resets, Interrupts, and System Configuration.”

**Table 4-1. Reset and Interrupt Vectors**

Address (High/Low)	Vector	Vector Name
0xFFC0:FFC1 ↕ 0xFFCA:FFCB	Unused Vector Space (available for user program)	
0xFFCC:FFCD	RTI	Vrti
0xFFCE:FFCF	IIC	Viic1
0xFFD0:FFD1	ATD Conversion	Vatd1
0xFFD2:FFD3	Keyboard	Vkeyboard1
0xFFD4:FFD5	SCI2 Transmit	Vsci2tx
0xFFD6:FFD7	SCI2 Receive	Vsci2rx
0xFFD8:FFD9	SCI2 Error	Vsci2err
0xFFDA:FFDB	SCI1 Transmit	Vsci1tx
0xFFDC:FFDD	SCI1 Receive	Vsci1rx
0xFFDE:FFDF	SCI1 Error	Vsci1err
0xFFE0:FFE1	SPI	Vspi1
0xFFE2:FFE3	TPM2 Overflow	Vtpm2ovf
0xFFE4:FFE5	TPM2 Channel 4	Vtpm2ch4
0xFFE6:FFE7	TPM2 Channel 3	Vtpm2ch3
0xFFE8:FFE9	TPM2 Channel 2	Vtpm2ch2
0xFFEA:FFEB	TPM2 Channel 1	Vtpm2ch1
0xFFEC:FFED	TPM2 Channel 0	Vtpm2ch0
0xFFEE:FFEF	TPM1 Overflow	Vtpm1ovf
0xFFF0:FFF1	TPM1 Channel 2	Vtpm1ch2
0xFFF2:FFF3	TPM1 Channel 1	Vtpm1ch1
0xFFF4:FFF5	TPM1 Channel 0	Vtpm1ch0
0xFFF6:FFF7	ICG	Vicg
0xFFF8:FFF9	Low Voltage Detect	Vlvd
0xFFFA:FFFB	IRQ	Virq
0xFFFC:FFFD	SWI	Vswi
0xFFFE:FFFF	Reset	Vreset

to program the entire array through the single-wire background debug interface. Because no special voltages are needed for flash erase and programming operations, in-application programming is also possible through other software-controlled communication paths. For a more detailed discussion of in-circuit and in-application programming, refer to the *HCS08 Family Reference Manual, Volume I*, Freescale Semiconductor document order number HCS08RMv1/D.

### 4.4.1 Features

Features of the flash memory include:

- Flash Size
  - MC9S08GB60A/MC9S08GT60A — 61268 bytes (120 pages of 512 bytes each)
  - MC9S08GB32A/MC9S08GT32A— 32768 bytes (64 pages of 512 bytes each)
- Single power supply program and erase
- Command interface for fast program and erase operation
- Up to 100,000 program/erase cycles at typical voltage and temperature
- Flexible block protection
- Security feature for flash and RAM
- Auto power-down for low-frequency read accesses

### 4.4.2 Program and Erase Times

Before any program or erase command can be accepted, the flash clock divider register (FCDIV) must be written to set the internal clock for the flash module to a frequency ( $f_{FCLK}$ ) between 150 kHz and 200 kHz (see Table 4.6.1). This register can be written only once, so normally this write is done during reset initialization. FCDIV cannot be written if the access error flag, FACCERR in FSTAT, is set. The user must ensure that FACCERR is not set before writing to the FCDIV register. One period of the resulting clock ( $1/f_{FCLK}$ ) is used by the command processor to time program and erase pulses. An integer number of these timing pulses is used by the command processor to complete a program or erase command.

Table 4-5 shows program and erase times. The bus clock frequency and FCDIV determine the frequency of FCLK ( $f_{FCLK}$ ). The time for one cycle of FCLK is  $t_{FCLK} = 1/f_{FCLK}$ . The times are shown as a number of cycles of FCLK and as an absolute time for the case where  $t_{FCLK} = 5 \mu\text{s}$ . Program and erase times shown include overhead for the command state machine and enabling and disabling of program and erase voltages.

**Table 4-5. Program and Erase Times**

Parameter	Cycles of FCLK	Time if FCLK = 200 kHz
Byte program	9	45 $\mu\text{s}$
Byte program (burst)	4	20 $\mu\text{s}$ <sup>1</sup>
Page erase	4000	20 ms
Mass erase	20,000	100 ms

<sup>1</sup> Excluding start/end overhead

### 4.4.7 Vector Redirection

Whenever any block protection is enabled, the reset and interrupt vectors will be protected. Vector redirection allows users to modify interrupt vector information without unprotecting bootloader and reset vector space. Vector redirection is enabled by programming the FNORED bit in the NVOPT register located at address 0xFFBF to zero. For redirection to occur, at least some portion but not all of the flash memory must be block protected by programming the NVPROT register located at address 0xFFBD. All of the interrupt vectors (memory locations 0xFFC0–0xFFFFD) are redirected, while the reset vector (0xFFFFE:FFFF) is not. When more than 32K is protected, vector redirection must not be enabled.

For example, if 512 bytes of flash are protected, the protected address region is from 0xFE00 through 0xFFFF. The interrupt vectors (0xFFC0–0xFFFFD) are redirected to the locations 0xFDC0–0xFDFD. Now, if an SPI interrupt is taken for instance, the values in the locations 0xFDE0:FDE1 are used for the vector instead of the values in the locations 0xFFE0:FFE1. This allows the user to reprogram the unprotected portion of the flash with new program code including new interrupt vector values while leaving the protected area, which includes the default vector locations, unchanged.

## 4.5 Security

The MC9S08GBxxA/GTxxA includes circuitry to prevent unauthorized access to the contents of flash and RAM memory. When security is engaged, flash and RAM are considered secure resources. Direct-page registers, high-page registers, and the background debug controller are considered unsecured resources. Programs executing within secure memory have normal access to any MCU memory locations and resources. Attempts to access a secure memory location with a program executing from an unsecured memory space or through the background debug interface are blocked (writes are ignored and reads return all 0s).

Security is engaged or disengaged based on the state of two nonvolatile register bits (SEC01:SEC00) in the FOPT register. During reset, the contents of the nonvolatile location NVOPT are copied from flash into the working FOPT register in high-page register space. A user engages security by programming the NVOPT location which can be done at the same time the flash memory is programmed. The 1:0 state disengages security while the other three combinations engage security. Notice the erased state (1:1) 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.

in the CCR is 0 to allow interrupts. The global interrupt mask (I bit) in the CCR is initially set after reset which masks (prevents) all maskable interrupt sources. The user program initializes the stack pointer and performs other system setup before clearing the I bit to allow the CPU to respond to interrupts.

When the CPU receives a qualified interrupt request, it completes the current instruction before responding to the interrupt. The interrupt sequence follows the same cycle-by-cycle sequence as the SWI instruction and consists of:

- Saving the CPU registers on the stack
- Setting the I bit in the CCR to mask further interrupts
- Fetching the interrupt vector for the highest-priority interrupt that is currently pending
- Filling the instruction queue with the first three bytes of program information starting from the address fetched from the interrupt vector locations

While the CPU is responding to the interrupt, the I bit is automatically set to avoid the possibility of another interrupt interrupting the ISR itself (this is called nesting of interrupts). Normally, the I bit is restored to 0 when the CCR is restored from the value stacked on entry to the ISR. In rare cases, the I bit may be cleared inside an ISR (after clearing the status flag that generated the interrupt) so that other interrupts can be serviced without waiting for the first service routine to finish. This practice is not recommended for anyone other than the most experienced programmers because it can lead to subtle program errors that are difficult to debug.

The interrupt service routine ends with a return-from-interrupt (RTI) instruction which restores the CCR, A, X, and PC registers to their pre-interrupt values by reading the previously saved information off the stack.

#### NOTE

For compatibility with the M68HC08, the H register is not automatically saved and restored. It is good programming practice to push H onto the stack at the start of the interrupt service routine (ISR) and restore it just before the RTI that is used to return from the ISR.

When two or more interrupts are pending when the I bit is cleared, the highest priority source is serviced first (see [Table 5-1](#)).

### 5.5.1 Interrupt Stack Frame

[Figure 5-1](#) shows the contents and organization of a stack frame. Before the interrupt, the stack pointer (SP) points at the next available byte location on the stack. The current values of CPU registers are stored on the stack starting with the low-order byte of the program counter (PCL) and ending with the CCR. After stacking, the SP points at the next available location on the stack which is the address that is one less than the address where the CCR was saved. The PC value that is stacked is the address of the instruction in the main program that would have executed next if the interrupt had not occurred.

configured for low bandwidth operation (RANGE = 0). If active BDM mode is enabled in stop3, the internal RTI clock is not available.

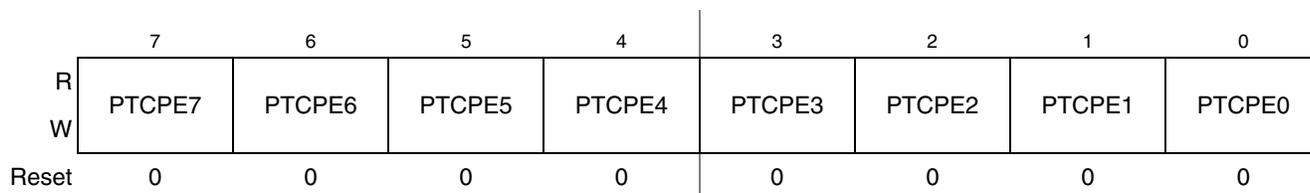
The SRTISC register includes a read-only status flag, a write-only acknowledge bit, and a 3-bit control value (RTIS2:RTIS1:RTIS0) used to select one of seven RTI periods. The RTI has a local interrupt enable, RTIE, to allow masking of the real-time interrupt. The module can be disabled by writing 0:0:0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

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

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

Refer to the direct-page register summary in [Chapter 4, “Memory”](#) of this data sheet for the absolute address assignments for all registers. This section refers to registers and control bits only by their names. A Freescale-provided equate or header file is used to translate these names into the appropriate absolute addresses.

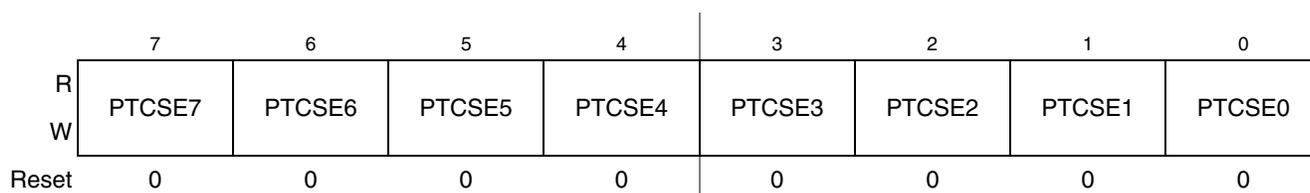
Some control bits in the SOPT and SPMSC2 registers are related to modes of operation. Although brief descriptions of these bits are provided here, the related functions are discussed in greater detail in [Chapter 3, “Modes of Operation.”](#)



**Figure 6-18. Pullup Enable for Port C (PTCPE)**

**Table 6-10. PTCPE Field Descriptions**

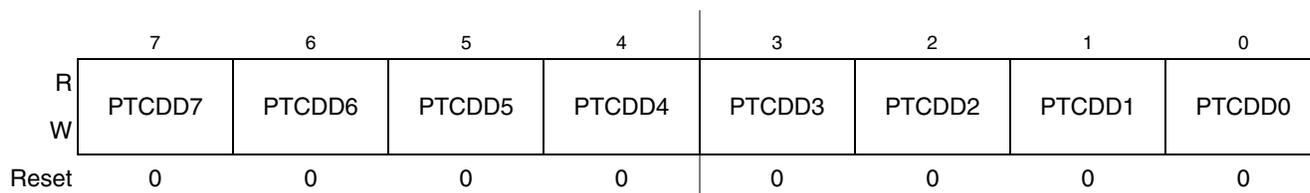
Field	Description
7:0 PTCPE[7:0]	<p><b>Pullup Enable for Port C Bits</b> — For port C pins that are inputs, these read/write control bits determine whether internal pullup devices are enabled. For port C pins that are configured as outputs, these bits are ignored and the internal pullup devices are disabled.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



**Figure 6-19. Slew Rate Control Enable for Port C (PTCSE)**

**Table 6-11. PTCSE Field Descriptions**

Field	Description
7:0 PTCSE[7:0]	<p><b>Slew Rate Control Enable for Port C Bits</b> — For port C pins that are outputs, these read/write control bits determine whether the slew rate controlled outputs are enabled. For port B pins that are configured as inputs, these bits are ignored.</p> <p>0 Slew rate control disabled. 1 Slew rate control enabled.</p>



**Figure 6-20. Data Direction for Port C (PTCDD)**

**Table 6-12. PTCDD Field Descriptions**

Field	Description
7:0 PTCDD[7:0]	<p><b>Data Direction for Port C Bits</b> — These read/write bits control the direction of port C pins and what is read for PTCDD reads.</p> <p>0 Input (output driver disabled) and reads return the pin value. 1 Output driver enabled for port C bit n and PTCDD reads return the contents of PTCDDn.</p>

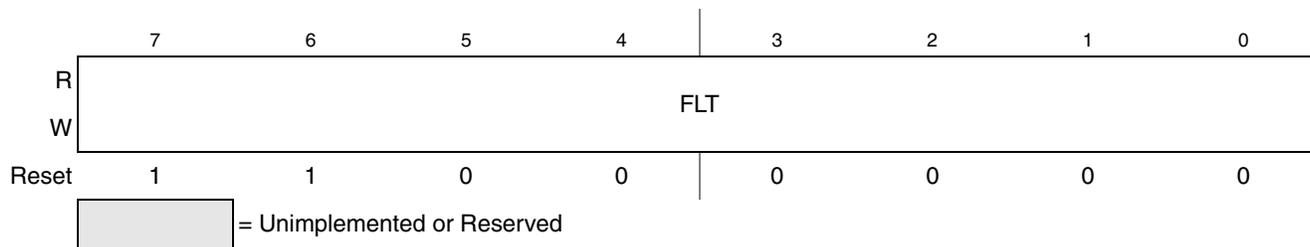


Figure 7-17. ICG Upper Filter Register (ICGFLTU)

Table 7-11. ICGFLTU Field Descriptions

Field	Description
7:0 FLT	<b>Filter Value</b> — 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.

### 7.5.6 ICG Trim Register (ICGTRM)

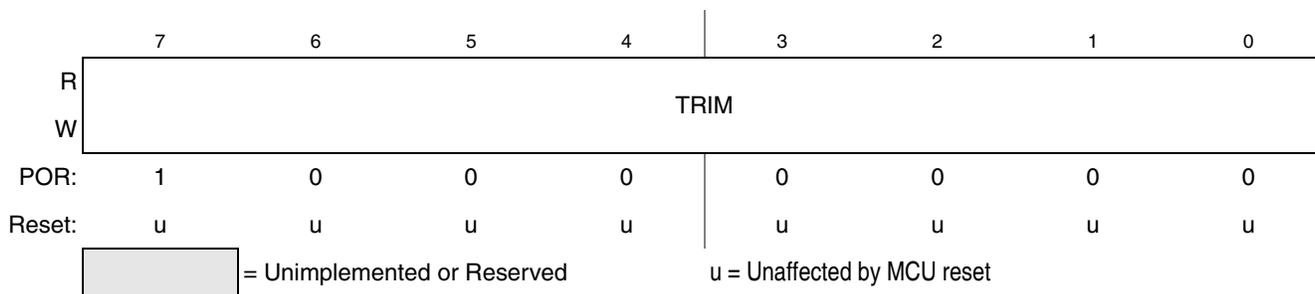


Figure 7-18. ICG Trim Register (ICGTRM)

Table 7-12. ICGTRM Field Descriptions

Field	Description
7:0 TRIM	<b>ICG Trim Setting</b> — The TRIM bits control the internal reference generator frequency. They allow a $\pm 25\%$ adjustment of the nominal (POR) period. The bit's effect on period is binary weighted (i.e., bit 1 will adjust twice as much as changing bit 0). Increasing the binary value in TRIM will increase the period and decreasing the value will decrease the period.

### 8.3.6.7 SP-Relative, 16-Bit Offset (SP2)

This variation of indexed addressing uses the 16-bit value in the stack pointer (SP) plus a 16-bit offset included in the instruction as the address of the operand needed to complete the instruction.

## 8.4 Special Operations

The CPU performs a few special operations that are similar to instructions but do not have opcodes like other CPU instructions. In addition, a few instructions such as STOP and WAIT directly affect other MCU circuitry. This section provides additional information about these operations.

### 8.4.1 Reset Sequence

Reset can be caused by a power-on-reset (POR) event, internal conditions such as the COP (computer operating properly) watchdog, or by assertion of an external active-low reset pin. When a reset event occurs, the CPU immediately stops whatever it is doing (the MCU does not wait for an instruction boundary before responding to a reset event). For a more detailed discussion about how the MCU recognizes resets and determines the source, refer to the [Resets, Interrupts, and System Configuration](#) chapter.

The reset event is considered concluded when the sequence to determine whether the reset came from an internal source is done and when the reset pin is no longer asserted. At the conclusion of a reset event, the CPU performs a 6-cycle sequence to fetch the reset vector from 0xFFFFE and 0xFFFF and to fill the instruction queue in preparation for execution of the first program instruction.

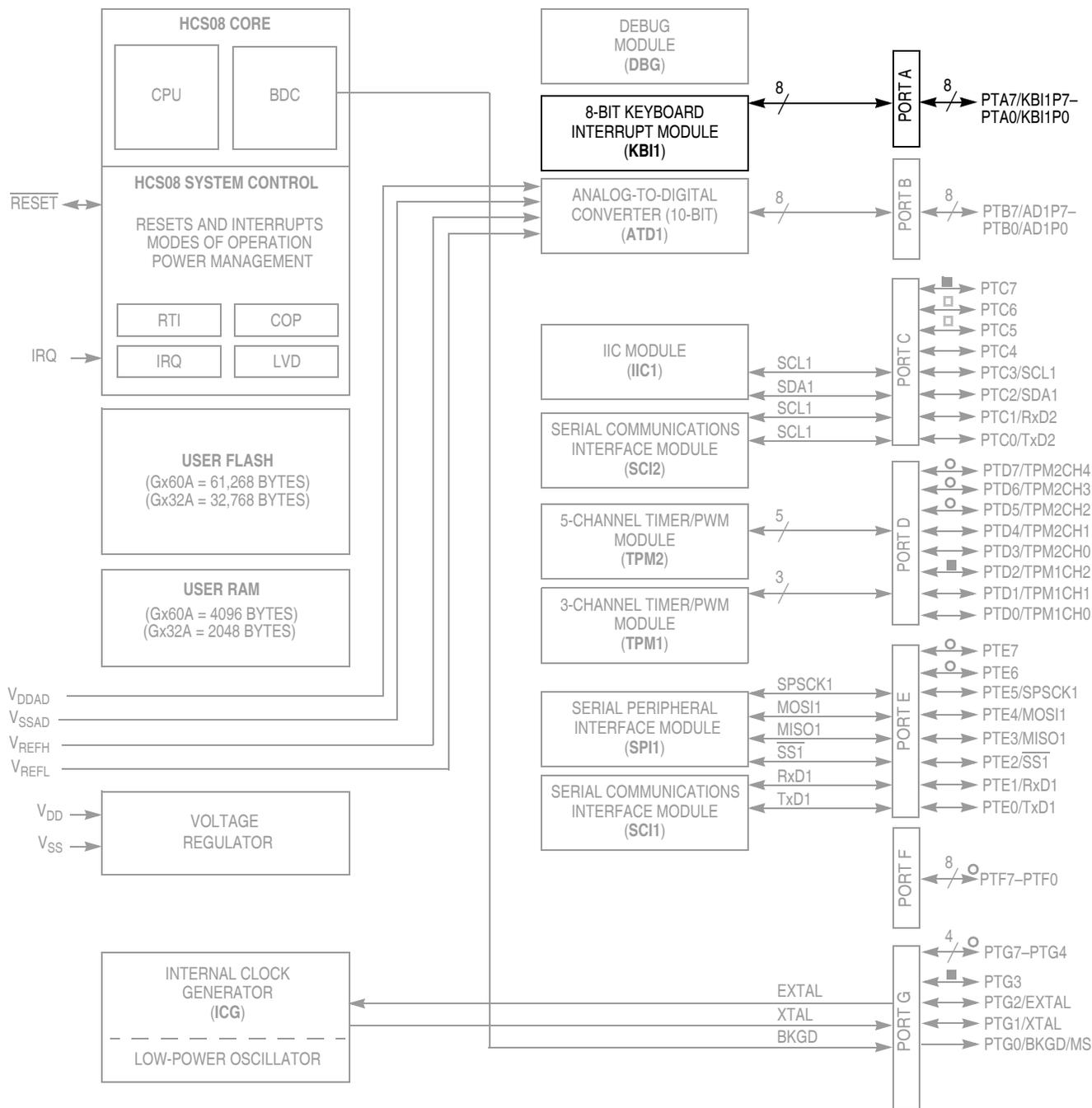
### 8.4.2 Interrupt Sequence

When an interrupt is requested, the CPU completes the current instruction before responding to the interrupt. At this point, the program counter is pointing at the start of the next instruction, which is where the CPU should return after servicing the interrupt. The CPU responds to an interrupt by performing the same sequence of operations as for a software interrupt (SWI) instruction, except the address used for the vector fetch is determined by the highest priority interrupt that is pending when the interrupt sequence started.

The CPU sequence for an interrupt is:

1. Store the contents of PCL, PCH, X, A, and CCR on the stack, in that order.
2. Set the I bit in the CCR.
3. Fetch the high-order half of the interrupt vector.
4. Fetch the low-order half of the interrupt vector.
5. Delay for one free bus cycle.
6. Fetch three bytes of program information starting at the address indicated by the interrupt vector to fill the instruction queue in preparation for execution of the first instruction in the interrupt service routine.

After the CCR contents are pushed onto the stack, the I bit in the CCR is set to prevent other interrupts while in the interrupt service routine. Although it is possible to clear the I bit with an instruction in the



**Note:** Not all pins are bonded out in all packages. See Table 2-2 for complete details.

Block Diagram Symbol Key:	
○	Not connected in 48-, 44-, and 42-pin packages
■	Not connected in 44- and 42-pin packages
□	Not connected in 42-pin packages

**Figure 9-2. Block Diagram Highlighting KBI Module**

### 9.4.3 KBI Interrupt Controls

The KBF status flag becomes set (1) when an edge event has been detected on any KBI input pin. If  $KBIE = 1$  in the KBI1SC register, a hardware interrupt will be requested whenever  $KBF = 1$ . The KBF flag is cleared by writing a 1 to the keyboard acknowledge (KBACK) bit.

When  $KBIMOD = 0$  (selecting edge-only operation), KBF is always cleared by writing 1 to KBACK. When  $KBIMOD = 1$  (selecting edge-and-level operation), KBF cannot be cleared as long as any keyboard input is at its asserted level.

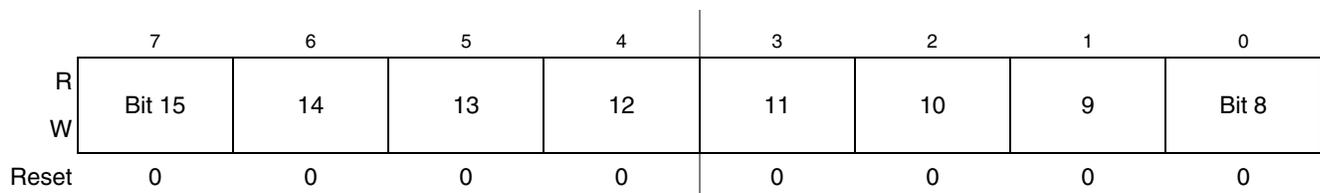
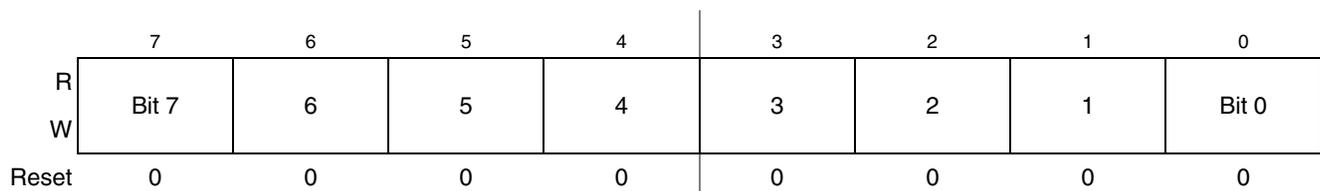
**Table 10-5. Mode, Edge, and Level Selection**

CPWMS	MSnB:MSnA	ELSnB:ELSnA	Mode	Configuration
X	XX	00	Pin not used for TPM channel; use as an external clock for the TPM or revert to general-purpose I/O	
0	00	01	Input capture	Capture on rising edge only
		10		Capture on falling edge only
		11		Capture on rising or falling edge
	01	00	Output compare	Software compare only
		01		Toggle output on compare
		10		Clear output on compare
		11		Set output on compare
	1X	10	Edge-aligned PWM	High-true pulses (clear output on compare)
		X1		Low-true pulses (set output on compare)
1	XX	10	Center-aligned PWM	High-true pulses (clear output on compare-up)
		X1		Low-true pulses (set output on compare-up)

If the associated port pin is not stable for at least two bus clock cycles before changing to input capture mode, it is possible to get an unexpected indication of an edge trigger. Typically, a program would clear status flags after changing channel configuration bits and before enabling channel interrupts or using the status flags to avoid any unexpected behavior.

### 10.7.5 Timer x Channel Value Registers (TPMxCnVH:TPMxCnVL)

These read/write registers contain the captured TPM counter value of the input capture function or the output compare value for the output compare or PWM functions. The channel value registers are cleared by reset.


**Figure 10-11. Timer x Channel Value Register High (TPMxCnVH)**

**Figure 10-12. Timer Channel Value Register Low (TPMxCnVL)**

# Chapter 11

## Serial Communications Interface (S08SCIV1)

### 11.1 Introduction

The MC9S08GBxxA/GTxxA includes two independent serial communications interface (SCI) modules — sometimes called universal asynchronous receiver/transmitters (UARTs). Typically, these systems are used to connect to the RS232 serial input/output (I/O) port of a personal computer or workstation, and they can also be used to communicate with other embedded controllers.

A flexible, 13-bit, modulo-based baud rate generator supports a broad range of standard baud rates beyond 115.2 kbaud. Transmit and receive within the same SCI use a common baud rate, and each SCI module has a separate baud rate generator.

This SCI system offers many advanced features not commonly found on other asynchronous serial I/O peripherals on other embedded controllers. The receiver employs an advanced data sampling technique that ensures reliable communication and noise detection. Hardware parity, receiver wakeup, and double buffering on transmit and receive are also included.

### 13.1.3 Block Diagram

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

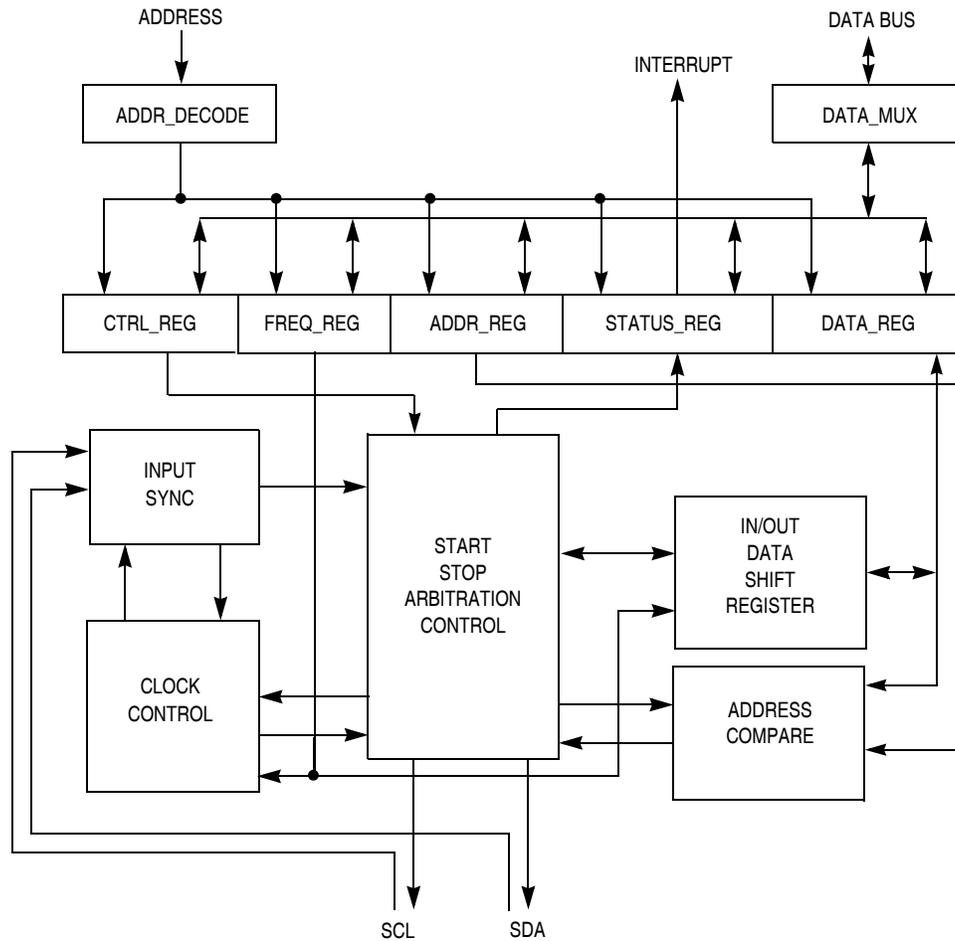


Figure 13-2. IIC Functional Block Diagram

## 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.

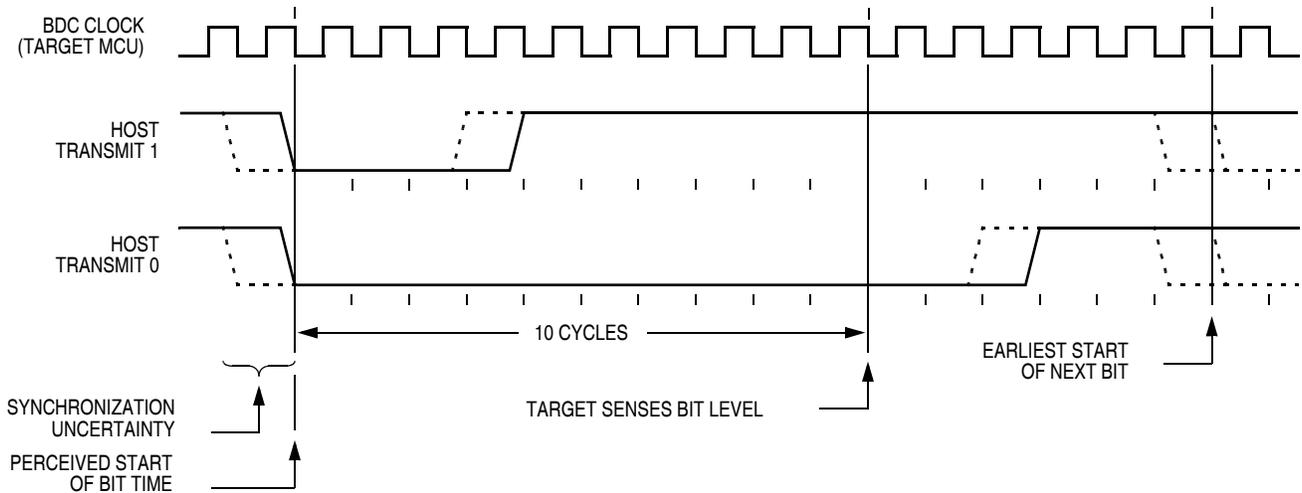
### 13.2.2 SDA — Serial Data Line

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.

Figure 15-2 shows an external host transmitting a logic 1 or 0 to the BKGD pin of a target HCS08 MCU. The host is asynchronous to the target so there is a 0-to-1 cycle delay from the host-generated falling edge to where the target perceives the beginning of the bit time. Ten target BDC clock cycles later, the target senses the bit level on the BKGD pin. Typically, the host actively drives the pseudo-open-drain BKGD pin during host-to-target transmissions to speed up rising edges. Because the target does not drive the BKGD pin during the host-to-target transmission period, there is no need to treat the line as an open-drain signal during this period.



**Figure 15-2. BDC Host-to-Target Serial Bit Timing**

### 15.4.3.7 Debug Control Register (DBGC)

This register can be read or written at any time.

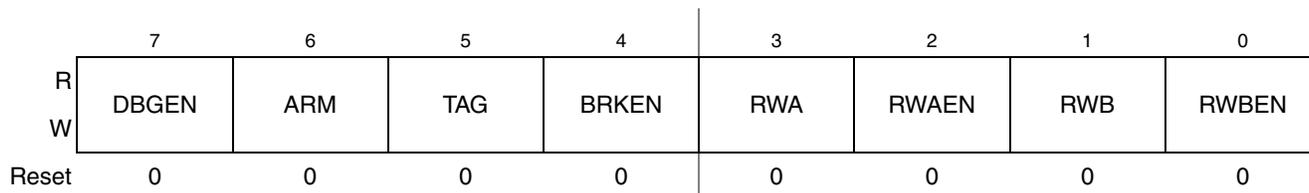


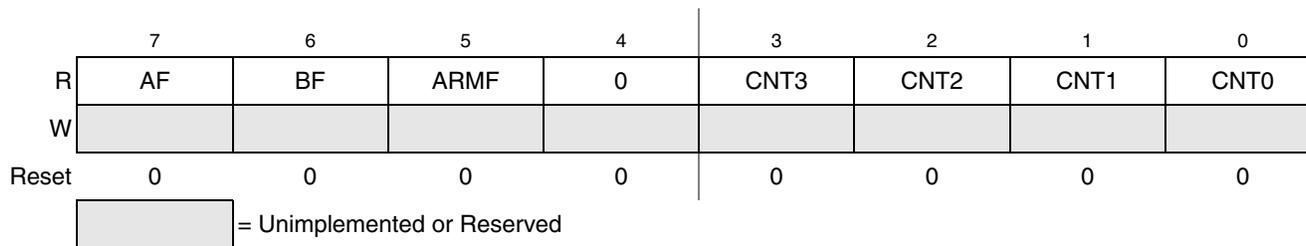
Figure 15-7. Debug Control Register (DBGC)

Table 15-4. DBGC Register Field Descriptions

Field	Description
7 DBGEN	<b>Debug Module Enable</b> — Used to enable the debug module. DBGEN cannot be set to 1 if the MCU is secure. 0 DBG disabled 1 DBG enabled
6 ARM	<b>Arm Control</b> — Controls whether the debugger is comparing and storing information in the FIFO. A write is used to set this bit (and ARMF) and completion of a debug run automatically clears it. Any debug run can be manually stopped by writing 0 to ARM or to DBGEN. 0 Debugger not armed 1 Debugger armed
5 TAG	<b>Tag/Force Select</b> — Controls whether break requests to the CPU will be tag or force type requests. If BRKEN = 0, this bit has no meaning or effect. 0 CPU breaks requested as force type requests 1 CPU breaks requested as tag type requests
4 BRKEN	<b>Break Enable</b> — Controls whether a trigger event will generate a break request to the CPU. Trigger events can cause information to be stored in the FIFO without generating a break request to the CPU. For an end trace, CPU break requests are issued to the CPU when the comparator(s) and R/W meet the trigger requirements. For a begin trace, CPU break requests are issued when the FIFO becomes full. TRGSEL does not affect the timing of CPU break requests. 0 CPU break requests not enabled 1 Triggers cause a break request to the CPU
3 RWA	<b>R/W Comparison Value for Comparator A</b> — When RWAEN = 1, this bit determines whether a read or a write access qualifies comparator A. When RWAEN = 0, RWA and the R/W signal do not affect comparator A. 0 Comparator A can only match on a write cycle 1 Comparator A can only match on a read cycle
2 RWAEN	<b>Enable R/W for Comparator A</b> — Controls whether the level of R/W is considered for a comparator A match. 0 R/W is not used in comparison A 1 R/W is used in comparison A
1 RWB	<b>R/W Comparison Value for Comparator B</b> — When RWBEN = 1, this bit determines whether a read or a write access qualifies comparator B. When RWBEN = 0, RWB and the R/W signal do not affect comparator B. 0 Comparator B can match only on a write cycle 1 Comparator B can match only on a read cycle
0 RWBEN	<b>Enable R/W for Comparator B</b> — Controls whether the level of R/W is considered for a comparator B match. 0 R/W is not used in comparison B 1 R/W is used in comparison B

### 15.4.3.9 Debug Status Register (DBGS)

This is a read-only status register.



**Figure 15-9. Debug Status Register (DBGS)**

**Table 15-6. DBGS Register Field Descriptions**

Field	Description
7 AF	<b>Trigger Match A Flag</b> — AF is cleared at the start of a debug run and indicates whether a trigger match A condition was met since arming. 0 Comparator A has not matched 1 Comparator A match
6 BF	<b>Trigger Match B Flag</b> — BF is cleared at the start of a debug run and indicates whether a trigger match B condition was met since arming. 0 Comparator B has not matched 1 Comparator B match
5 ARMF	<b>Arm Flag</b> — While DBGEN = 1, this status bit is a read-only image of ARM in DBG. This bit is set by writing 1 to the ARM control bit in DBG (while DBGEN = 1) and is automatically cleared at the end of a debug run. A debug run is completed when the FIFO is full (begin trace) or when a trigger event is detected (end trace). A debug run can also be ended manually by writing 0 to ARM or DBGEN in DBG. 0 Debugger not armed 1 Debugger armed
3:0 CNT[3:0]	<b>FIFO Valid Count</b> — These bits are cleared at the start of a debug run and indicate the number of words of valid data in the FIFO at the end of a debug run. The value in CNT does not decrement as data is read out of the FIFO. The external debug host is responsible for keeping track of the count as information is read out of the FIFO. 0000 Number of valid words in FIFO = No valid data 0001 Number of valid words in FIFO = 1 0010 Number of valid words in FIFO = 2 0011 Number of valid words in FIFO = 3 0100 Number of valid words in FIFO = 4 0101 Number of valid words in FIFO = 5 0110 Number of valid words in FIFO = 6 0111 Number of valid words in FIFO = 7 1000 Number of valid words in FIFO = 8

**Table A-4. DC Characteristics (Sheet 3 of 3)**  
**(Temperature Range = -40 to 85°C Ambient)**

Parameter	Symbol	Min	Typical <sup>1</sup>	Max	Unit
dc injection current <sup>4, 5, 6, 7, 8</sup> $V_{IN} < V_{SS}, V_{IN} > V_{DD}$ Single pin limit Total MCU limit, includes sum of all stressed pins	$ I_{IC} $	—		0.2 5	mA mA
Input capacitance (all non-supply pins) <sup>(2)</sup>	$C_{In}$	—		7	pF

<sup>1</sup> Typicals are measured at 25°C.

<sup>2</sup> This parameter is characterized and not tested on each device.

<sup>3</sup> Measurement condition for pull resistors:  $V_{IN} = V_{SS}$  for pullup and  $V_{IN} = V_{DD}$  for pulldown.

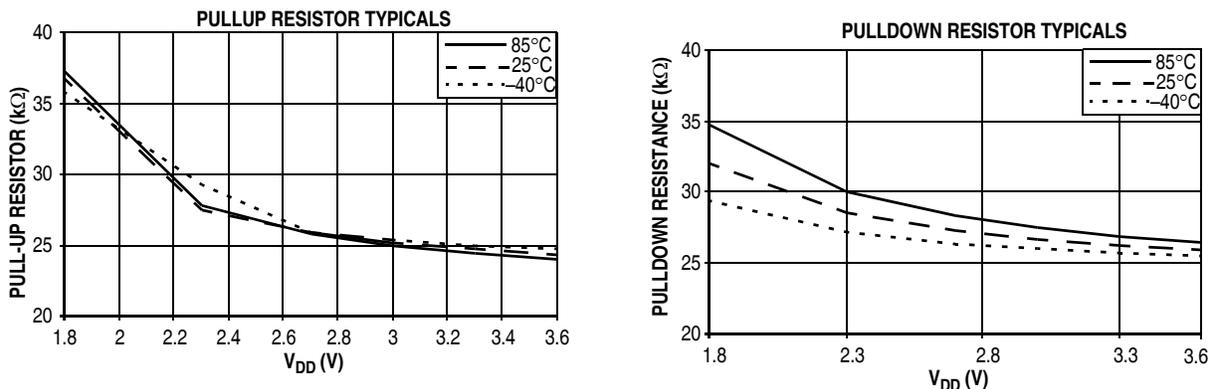
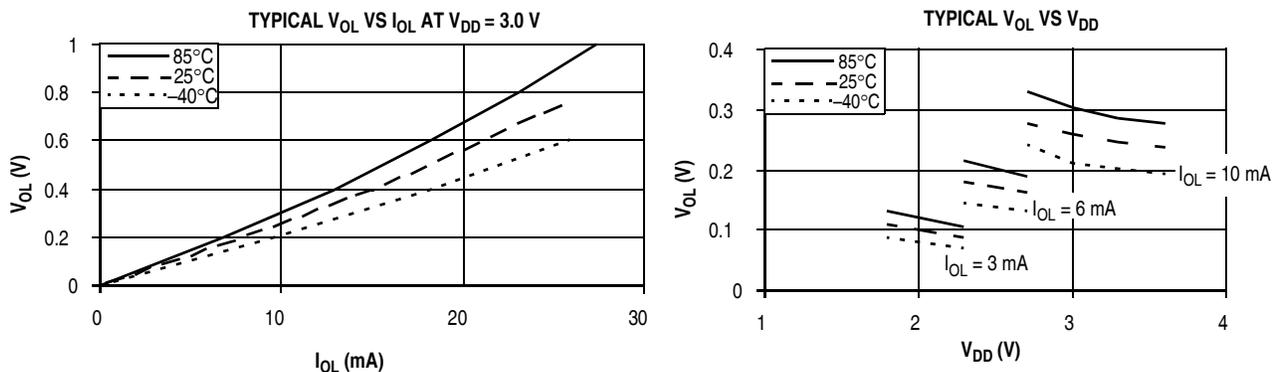
<sup>4</sup> Power supply must maintain regulation within operating  $V_{DD}$  range during instantaneous and operating maximum current conditions. If positive injection current ( $V_{IN} > V_{DD}$ ) is greater than  $I_{DD}$ , the injection current may flow out of  $V_{DD}$  and could result in external power supply going out of regulation. Ensure external  $V_{DD}$  load will shunt current greater than maximum injection current. This will be the greatest risk when the MCU is not consuming power. Examples are: if no system clock is present, or if clock rate is very low which would reduce overall power consumption.

<sup>5</sup> All functional non-supply pins are internally clamped to  $V_{SS}$  and  $V_{DD}$ .

<sup>6</sup> Input must be current limited to the value specified. To determine the value of the required current-limiting resistor, calculate resistance values for positive and negative clamp voltages, then use the larger of the two values.

<sup>7</sup> This parameter is characterized and not tested on each device.

<sup>8</sup> IRQ does not have a clamp diode to  $V_{DD}$ . Do not drive IRQ above  $V_{DD}$ .


**Figure A-1. Pullup and Pulldown Typical Resistor Values ( $V_{DD} = 3.0$  V)**

**Figure A-2. Typical Low-Side Driver (Sink) Characteristics (Ports C and F)**



NOTES:

1. DIMENSIONS AND TOLERANCING PER ASME Y14.5M, 1994.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE -H- IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS A-B AND -D- TO BE DETERMINED AT DATUM PLANE -H-.

5. THIS DIMENSION TO BE DETERMINED AT SEATING PLANE -C-.

6. THIS DIMENSION DO NOT INCLUDE MOLD PROTRUSION, ALLOWABLE PROTRUSION IS 0.25 PER SIDE, DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE -H-.

7. THIS DIMENSION DOES NOT INCLUDE DAMBAR PROTRUSION, ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	<b>MECHANICAL OUTLINE</b>	PRINT VERSION NOT TO SCALE	
TITLE: 44LD QFP, 10X10X2.0 PKG, 0.8 PITCH	DOCUMENT NO: 98ASB42839B	REV: B	
	CASE NUMBER: 824A-01	06 APR 2005	
	STANDARD: NON-JEDEC		