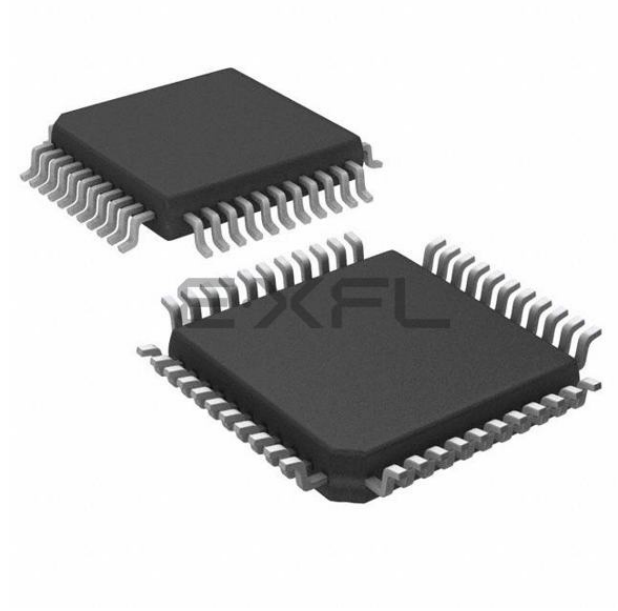

Welcome to E-XFL.COM

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	36
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K 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	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08gt16acfber

Appendix A
Electrical Characteristics

Appendix B
Ordering Information and Mechanical Drawings

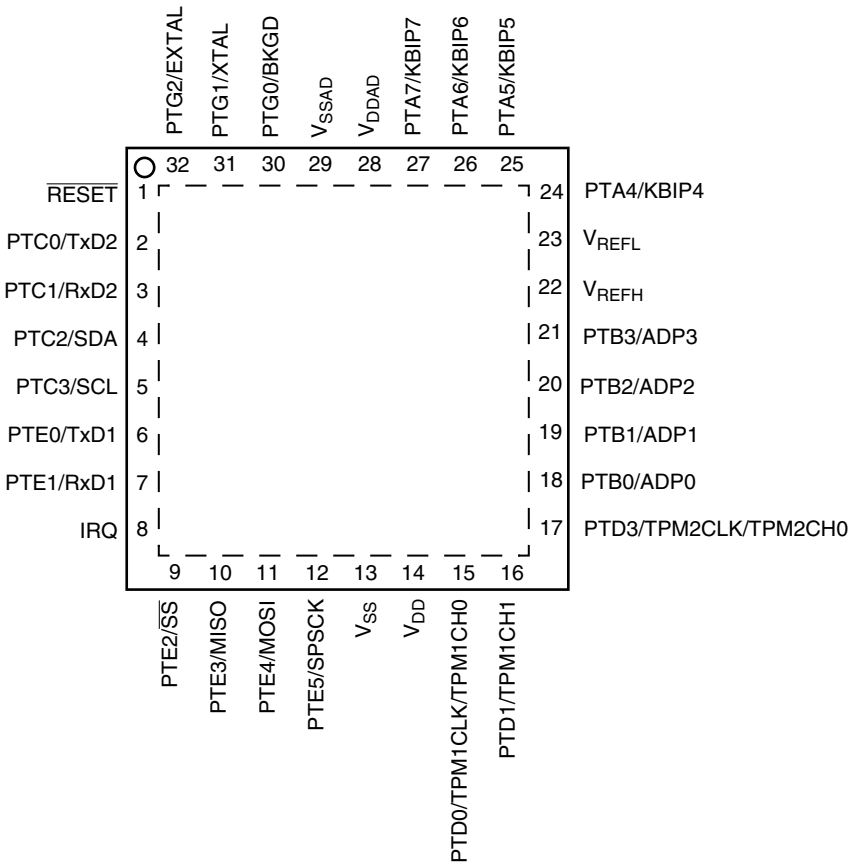

Figure 2-4. MC9S08GT16A/GT8A in 32-Pin QFN Package

Before entering stop2 mode, the user must save the contents of the I/O port registers, as well as any other memory-mapped registers they want to restore after exit of stop2, to locations in RAM. Upon exit of stop2, these values can be restored by user software before pin latches are opened.

When the MCU is in stop2 mode, all internal circuits that are powered from the voltage regulator are turned off, except for the RAM. The voltage regulator is in a low-power standby state, as is the ATD. Upon entry into stop2, the states of the I/O pins are latched. The states are held while in stop2 mode and after exiting stop2 mode until a 1 is written to PPDACK in SPMSC2.

Exit from stop2 is performed by asserting either of the wake-up pins: $\overline{\text{RESET}}$ or IRQ, or by an RTI interrupt. IRQ is always an active low input when the MCU is in stop2, regardless of how it was configured before entering stop2.

Upon wake-up from stop2 mode, the MCU will start up as from a power-on reset (POR) except pin states remain latched. The CPU will take the reset vector. The system and all peripherals will be in their default reset states and must be initialized.

After waking up from stop2, the PPDF bit in SPMSC2 is set. This flag may be used to direct user code to go to a stop2 recovery routine. PPDF remains set and the I/O pin states remain latched until a 1 is written to PPDACK in SPMSC2.

To maintain I/O state for pins that were configured as general-purpose I/O, the user must restore the contents of the I/O port registers, which have been saved in RAM, to the port registers before writing to the PPDACK bit. If the port registers are not restored from RAM before writing to PPDACK, then the register bits will assume their reset states when the I/O pin latches are opened and the I/O pins will switch to their reset states.

For pins that were configured as peripheral I/O, the user must reconfigure the peripheral module that interfaces to the pin before writing to the PPDACK bit. If the peripheral module is not enabled before writing to PPDACK, the pins will be controlled by their associated port control registers when the I/O latches are opened.

3.5.3 Stop3 Mode

Upon entering the stop3 mode, all of the clocks in the MCU, including the oscillator itself, are halted. The ICG is turned off, the ATD is disabled, and the voltage regulator is put in standby. The states of all of the internal registers and logic, as well as the RAM content, are maintained. The I/O pin states are not latched at the pin as in stop2. Instead they are maintained by virtue of the states of the internal logic driving the pins being maintained.

Exit from stop3 is performed by asserting $\overline{\text{RESET}}$, an asynchronous interrupt pin, or through the real-time interrupt. The asynchronous interrupt pins are the IRQ or KBI pins.

If stop3 is exited by means of the $\overline{\text{RESET}}$ pin, then the MCU will be reset and operation will resume after taking the reset vector. Exit by means of an asynchronous interrupt or the real-time interrupt will result in the MCU taking the appropriate interrupt vector.

A separate self-clocked source (≈1 kHz) for the real-time interrupt allows a wakeup from stop2 or stop3 mode with no external components. When RTIS2:RTIS1:RTIS0 = 0:0:0, the real-time interrupt function

and this 1-kHz source are disabled. Power consumption is lower when the 1-kHz source is disabled, but in that case the real-time interrupt cannot wake the MCU from stop.

3.5.4 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
Stop3	Don't care	Don't care	Standby	Standby	Active	Disabled[1]	Active	States held	Optionally on

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

3.5.5 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[1]	Active	States held	Optionally on

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

Table 4-2. Direct-Page Register Summary (Sheet 2 of 3)

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
0x0028	**SPIC1**	SPIE	SPE	SPTIE	MSTR	CPOL	CPHA	SSOE	LSBFE
0x0029	**SPIC2**	0	0	0	MODFEN	BIDIROE	0	SPISWAI	SPC0
0x002A	**SPIBR**	0	SPPR2	SPPR1	SPPR0	0	SPR2	SPR1	SPR0
0x002B	**SPIS**	SPRF	0	SPTEF	MODF	0	0	0	0
0x002C	Reserved	0	0	0	0	0	0	0	0
0x002D	**SPID**	Bit 7	6	5	4	3	2	1	Bit 0
0x002E	Reserved	0	0	0	0	0	0	0	0
0x002F	Reserved	0	0	0	0	0	0	0	0
0x0030	**TPM1SC**	TOF	TOIE	CPWMS	CLKSB	CLKSA	PS2	PS1	PS0
0x0031	**TPM1CNTH**	Bit 15	14	13	12	11	10	9	Bit 8
0x0032	**TPM1CNTL**	Bit 7	6	5	4	3	2	1	Bit 0
0x0033	**TPM1MODH**	Bit 15	14	13	12	11	10	9	Bit 8
0x0034	**TPM1MODL**	Bit 7	6	5	4	3	2	1	Bit 0
0x0035	**TPM1C0SC**	CH0F	CH0IE	MS0B	MS0A	ELS0B	ELS0A	0	0
0x0036	**TPM1C0VH**	Bit 15	14	13	12	11	10	9	Bit 8
0x0037	**TPM1C0VL**	Bit 7	6	5	4	3	2	1	Bit 0
0x0038	**TPM1C1SC**	CH1F	CH1IE	MS1B	MS1A	ELS1B	ELS1A	0	0
0x0039	**TPM1C1VH**	Bit 15	14	13	12	11	10	9	Bit 8
0x003A	**TPM1C1VL**	Bit 7	6	5	4	3	2	1	Bit 0
0x003B	**TPM1C2SC**	CH2F	CH2IE	MS2B	MS2A	ELS2B	ELS2A	0	0
0x003C	**TPM1C2VH**	Bit 15	14	13	12	11	10	9	Bit 8
0x003D	**TPM1C2VL**	Bit 7	6	5	4	3	2	1	Bit 0
0x003E– 0x0043	Reserved	— —	— —	— —	— —	— —	— —	— —	— —
0x0044	**PTGD**	0	0	0	0	PTGD3	PTGD2	PTGD1	PTGD0
0x0045	**PTGPE**	0	0	0	0	PTGPE3	PTGPE2	PTGPE1	PTGPE0
0x0046	**PTGSE**	0	0	0	0	PTGSE3	PTGSE2	PTGSE1	PTGSE0
0x0047	**PTGDD**	0	0	0	0	PTGDD3	PTGDD2	PTGDD1	PTGDD0
0x0048	**ICGC1**	HGO	RANGE	REFS	CLKS		OSCSTEN	LOCD	0
0x0049	**ICGC2**	LOLRE	MFD			LOCRE	RFD		
0x004A	**ICGS1**	CLKST		REFST	LOLS	LOCK	LOCS	ERCS	ICGIF
0x004B	**ICGS2**	0	0	0	0	0	0	0	DCOS
0x004C	**ICGFLTU**	0	0	0	0	FLT			
0x004D	**ICGFLTL**	FLT							
0x004E	**ICGTRM**	TRIM							
0x004F	Reserved	0	0	0	0	0	0	0	0
0x0050	**ATDC**	ATDPU	DJM	RES8	SGN	PRS			
0x0051	**ATDSC**	CCF	ATDIE	ATDCO	ATDCH				
0x0052	**ATDRH**	Bit 7	6	5	4	3	2	1	Bit 0
0x0053	**ATDRL**	Bit 7	6	5	4	3	2	1	Bit 0

- 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 $t_{FCLK} = 5$ µs. 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 µs
Byte program (burst)	4	20 µs[1]
Page erase	4000	20 ms
Mass erase	20,000	100 ms

[1] Excluding start/end overhead

4.4.3 Program and Erase Command Execution

The steps for executing any of the commands are listed below. The FCDIV register must be initialized and any error flags cleared before beginning command execution. The command execution steps are:

1. Write a data value to an address in the FLASH array. The address and data information from this write is latched into the FLASH interface. This write is a required first step in any command sequence. For erase and blank check commands, the value of the data is not important. For page erase commands, the address may be any address in the 512-byte page of FLASH to be erased. For mass erase and blank check commands, the address can be any address in the FLASH memory. Whole pages of 512 bytes are the smallest blocks of FLASH that may be erased.

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.

Table 4-11. FSTAT Field Descriptions (continued)

Field	Description
5 FPVIOL	**Protection Violation Flag** — FPVIOL is set automatically when FCBEF is cleared to register a command that attempts to erase or program a location in a protected block (the erroneous command is ignored). FPVIOL is cleared by writing a 1 to FPVIOL. 0 No protection violation. 1 An attempt was made to erase or program a protected location.
4 FACCERR	**Access Error Flag** — FACCERR is set automatically when the proper command sequence is not followed exactly (the erroneous command is ignored), if a program or erase operation is attempted before the FCDIV register has been initialized, or if the MCU enters stop while a command was in progress. For a more detailed discussion of the exact actions that are considered access errors, see Section 4.4.5, "Access Errors." FACCERR is cleared by writing a 1 to FACCERR. Writing a 0 to FACCERR has no meaning or effect. 0 No access error has occurred. 1 An access error has occurred.
2 FBLANK	**FLASH Verified as All Blank (Erased) Flag** — FBLANK is set automatically at the conclusion of a blank check command if the entire FLASH array was verified to be erased. FBLANK is cleared by clearing FCBEF to write a new valid command. Writing to FBLANK has no meaning or effect. 0 After a blank check command is completed and FCCF = 1, FBLANK = 0 indicates the FLASH array is not completely erased. 1 After a blank check command is completed and FCCF = 1, FBLANK = 1 indicates the FLASH array is completely erased (all 0xFF).

4.6.6 FLASH Command Register (FCMD)

Only five command codes are recognized in normal user modes as shown in Table 4-13. Refer to Section 4.4.3, "Program and Erase Command Execution" for a detailed discussion of FLASH programming and erase operations.

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W	FCMD7	FCMD6	FCMD5	FCMD4	FCMD3	FCMD2	FCMD1	FCMD0
Reset	0	0	0	0	0	0	0	0

Figure 4-10. FLASH Command Register (FCMD)

Table 4-12. FCMD Field Descriptions

Field	Description
7:0 FCMD[7:0]	**FLASH Command Bits --** See Table 4-13 for a description of FCMD[7:0].

5.4.2.2 Edge and Level Sensitivity

The IRQMOD control bit re-configures the detection logic so it detects edge events and pin levels. In this edge detection mode, the IRQF status flag becomes set when an edge is detected (when the IRQ pin changes from the deasserted to the asserted level), but the flag is continuously set (and cannot be cleared) as long as the IRQ pin remains at the asserted level.

5.4.3 Interrupt Vectors, Sources, and Local Masks

Table 5-1 provides a summary of all interrupt sources. Higher-priority sources are located toward the bottom of the table. The high-order byte of the address for the interrupt service routine is located at the first address in the vector address column, and the low-order byte of the address for the interrupt service routine is located at the next higher address.

When an interrupt condition occurs, an associated flag bit becomes set. If the associated local interrupt enable is 1, an interrupt request is sent to the CPU. Within the CPU, if the global interrupt mask (I bit in the CCR) is 0, the CPU will finish the current instruction, stack the PCL, PCH, X, A, and CCR CPU registers, set the I bit, and then fetch the interrupt vector for the highest priority pending interrupt. Processing then continues in the interrupt service routine.

5.7.2 System Reset Status Register (SRS)

This register includes six read-only status flags to indicate the source of the most recent reset. When a debug host forces reset by writing 1 to BDFR in the SBDFR register, none of the status bits in SRS will be set. Writing any value to this register address clears the COP watchdog timer without affecting the contents of this register. The reset state of these bits depends on what caused the MCU to reset.

	7	6	5	4	3	2	1	0
R	POR	PIN	COP	ILOP	ILAD	ICG	LVD	0
W	Writing any value to SIMRS address clears COP watchdog timer.							
Power-on reset:	1	0	0	0	0	0	1	0
Low-voltage reset:	U	0	0	0	0	0	1	0
Any other reset:	0	Note (1)	Note (1)	Note (1)	0	Note (1)	0	0

U = Unaffected by reset

1 Any of these reset sources that are active at the time of reset will cause the corresponding bit(s) to be set; bits corresponding to sources that are not active at the time of reset will be cleared.

Figure 5-3. System Reset Status (SRS)

Table 5-3. SRS Field Descriptions

Field	Description
7 POR	**Power-On Reset** — Reset was caused by the power-on detection logic. Because the internal supply voltage was ramping up at the time, the low-voltage reset (LVD) status bit is also set to indicate that the reset occurred while the internal supply was below the LVD threshold. 0 Reset not caused by POR. 1 POR caused reset.
6 PIN	**External Reset Pin** — Reset was caused by an active-low level on the external reset pin. 0 Reset not caused by external reset pin. 1 Reset came from external reset pin.
5 COP	**Computer Operating Properly (COP) Watchdog** — Reset was caused by the COP watchdog timer timing out. This reset source may be blocked by COPE = 0. 0 Reset not caused by COP timeout. 1 Reset caused by COP timeout.
4 ILOP	**Illegal Opcode** — Reset was caused by an attempt to execute an unimplemented or illegal opcode. The STOP instruction is considered illegal if stop is disabled by STOPE = 0 in the SOPT register. The BGND instruction is considered illegal if active background mode is disabled by ENBDM = 0 in the BDCSC register. 0 Reset not caused by an illegal opcode. 1 Reset caused by an illegal opcode.

Table 8-3. Opcode Map (Sheet 2 of 2)

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

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

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

Prebyte (9E) and Opcode in Hexadecimal
Number of Bytes

9E60 6
NEG
3 SP1

HCS08 Cycles
Instruction Mnemonic
Addressing Mode

9.3 Register Definition

Refer to the direct-page register summary in the Memory chapter of this data sheet for the absolute address assignments for all ICG 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.

9.3.1 ICG Control Register 1 (ICGC1)

	7	6	5	4	3	2	1	0
R	HGO[1]	RANGE	REFS	CLKS		OSCSTEN	LOCD	0
W								
Reset	0	1	0	0	0	1	0	0

= Unimplemented or Reserved

Figure 9-6. ICG Control Register 1 (ICGC1)

[1] This bit can be written only once after reset. Additional writes are ignored.

Table 9-1. ICGC1 Register Field Descriptions

Field	Description
7 HGO	**High Gain Oscillator Select** — The HGO bit is used to select between low power operation and high gain operation for improved noise immunity. This bit is write-once after reset. 0 Oscillator configured for low power operation. 1 Oscillator configured for high gain operation.
6 RANGE	**Frequency Range Select** — The RANGE bit controls the oscillator, reference divider, and FLL loop prescaler multiplication factor (P). It selects one of two reference frequency ranges for the ICG. The RANGE bit is write-once after a reset. The RANGE bit only has an effect in FLL engaged external and FLL bypassed external modes. 0 Oscillator configured for low frequency range. FLL loop prescale factor P is 64. 1 Oscillator configured for high frequency range. FLL loop prescale factor P is 1.
5 REFS	**External Reference Select** — The REFS bit controls the external reference clock source for ICGERCLK. The REFS bit is write-once after a reset. 0 External clock requested. 1 Oscillator using crystal or resonator requested.
4:3 CLKS	**Clock Mode Select** — The CLKS bits control the clock mode as described below. If FLL bypassed external is requested, it will not be selected until ERCS = 1. If the ICG enters off mode, the CLKS bits will remain unchanged. Writes to the CLKS bits will not take effect if a previous write is not complete. 00 Self-clocked 01 FLL engaged, internal reference 10 FLL bypassed, external reference 11 FLL engaged, external reference The CLKS bits are writable at any time, unless the first write after a reset was CLKS = 0X, the CLKS bits cannot be written to 1X until after the next reset (because the EXTAL pin was not reserved).

Table 9-12. MFD and RFD Decode Table

101	14	101	÷32
110	16	110	÷64
111	18	111	÷128

9.5.2 Example #1: External Crystal = 32 kHz, Bus Frequency = 4.19 MHz

In this example, the FLL will be used (in FEE mode) to multiply the external 32 kHz oscillator up to 8.38 MHz to achieve 4.19 MHz bus frequency.

After the MCU is released from reset, the ICG is in self-clocked mode (SCM) and supplies approximately 8 MHz on ICGOUT, which corresponds to a 4 MHz bus frequency (f_{Bus}).

The clock scheme will be FLL engaged, external (FEE). So

$$f_{ICGOUT} = f_{ext} * P * N / R ; P = 64, f_{ext} = 32 \text{ kHz} \quad \textit{Eqn. 9-1}$$

Solving for N / R gives:

$$N / R = 8.38 \text{ MHz} /(32 \text{ kHz} * 64) = 4 ; \text{we can choose N = 4 and R =1} \quad \textit{Eqn. 9-2}$$

The values needed in each register to set up the desired operation are:

ICGC1 = $38 (%00111000)

Bit 7	HGO	0	Configures oscillator for low power
Bit 6	RANGE	0	Configures oscillator for low-frequency range; FLL prescale factor is 64
Bit 5	REFS	1	Oscillator using crystal or resonator is requested
Bits 4:3	CLKS	11	FLL engaged, external reference clock mode
Bit 2	OSCSTEN	0	Oscillator disabled
Bit 1	LOCD	0	Loss-of-clock detection enabled
Bit 0		0	Unimplemented or reserved, always reads zero

ICGC2 = $00 (%00000000)

Bit 7	LOLRE	0	Generates an interrupt request on loss of lock
Bits 6:4	MFD	000	Sets the MFD multiplication factor to 4
Bit 3	LOCRE	0	Generates an interrupt request on loss of clock
Bits 2:0	RFD	000	Sets the RFD division factor to ÷1

ICGS1 = $xx

This is read only except for clearing interrupt flag

ICGS2 = $xx

This is read only; should read DCOS = 1 before performing any time critical tasks

ICGFLTLU/L = $xx

Only needed in self-clocked mode; FLT will be adjusted by loop to give 8.38 MHz DCO clock

Bits 15:12	unused	0000

12.6.1.2 Pseudo—Code Example

In this example, the SPI module will be set up for master mode with only transmit interrupts enabled to run at a maximum baud rate of bus clock divided by 2. Clock phase and polarity will be set for an active-high SPI clock where the first edge on SPSCK occurs at the start of the first cycle of a data transfer.

SPIC1 = 0x74(%01110100)

Bit 7	SPIE	= 0	Disables receive and mode fault interrupts
Bit 6	SPE	= 1	Enables the SPI system
Bit 5	SPTIE	= 1	Enables SPI transmit interrupts
Bit 4	MSTR	= 1	Sets the SPI module as a master SPI device
Bit 3	CPOL	= 0	Configures SPI clock as active-high
Bit 2	CPHA	= 1	First edge on SPSCK at start of first data transfer cycle
Bit 1	SSOE	= 0	Determines $\overline{SS}$ pin function when mode fault enabled
Bit 0	LSBFE	= 0	SPI serial data transfers start with most significant bit

SPIC2 = 0x00(%00000000)

Bit 7:5		= 000	Unimplemented
Bit 4	MODFEN	= 0	Disables mode fault function
Bit 3	BIDIROE	= 0	SPI data I/O pin acts as input
Bit 2		= 0	Unimplemented
Bit 1	SPISWAI	= 0	SPI clocks operate in wait mode
Bit 0	SPC0	= 0	SPI uses separate pins for data input and output

SPIBR = 0x00(%00000000)

Bit 7	= 0	Unimplemented
Bit 6:4	= 000	Sets prescale divisor to 1
Bit 3	= 0	Unimplemented
Bit 2:0	= 000	Sets baud rate divisor to 2

SPIS = 0x00(%00000000)

Bit 7	SPRF	= 0	Flag is set when receive data buffer is full
Bit 6		= 0	Unimplemented
Bit 5	SPTEF	= 0	Flag is set when transmit data buffer is empty
Bit 4	MODF	= 0	Mode fault flag for master mode
Bit 3:0		= 0	Unimplemented

SPID = 0xxx

Holds data to be transmitted by transmit buffer and data received by receive buffer.

13.1.1 Features

The IIC includes these distinctive features:

- Compatible with IIC bus standard
- Multi-master operation
- Software programmable for one of 64 different serial clock frequencies
- Software selectable acknowledge bit
- Interrupt driven byte-by-byte data transfer
- Arbitration lost interrupt with automatic mode switching from master to slave
- Calling address identification interrupt
- START and STOP signal generation/detection
- Repeated START signal generation
- Acknowledge bit generation/detection
- Bus busy detection

13.1.2 Modes of Operation

The IIC functions the same in normal and monitor modes. A brief description of the IIC in the various MCU modes is given here.

- Run mode — This is the basic mode of operation. To conserve power in this mode, disable the module.
- Wait mode — The module will continue to operate while the MCU is in wait mode and can provide a wake-up interrupt.
- Stop mode — The IIC is inactive in stop3 mode for reduced power consumption. The STOP instruction does not affect IIC register states. Stop1 and stop2 will reset the register contents.

Table 14-1. Signal Properties

Name	Function
AD7–AD0	Channel input pins
V_{REFH}	High reference voltage for ATD converter
V_{REFL}	Low reference voltage for ATD converter
V_{DDAD}	ATD power supply voltage
V_{SSAD}	ATD ground supply voltage

14.2.1 ADP7–ADP0 — Channel Input Pins

The channel pins are used as the analog input pins of the ATD. Each pin is connected to an analog switch which serves as the signal gate into the sample submodule.

14.2.2 V_{REFH}, V_{REFL} — ATD Reference Pins

These pins serve as the source for the high and low reference potentials for the converter. Separation from the power supply pins accommodates the filtering necessary to achieve the accuracy of which the system is capable.

14.2.3 V_{DDAD}, V_{SSAD} — ATD Supply Pins

These two pins are used to supply power and ground to the analog section of the ATD. Dedicated power is required to isolate the sensitive analog circuitry from the normal levels of noise present on digital power supplies.

NOTE

V_{DDAD1} and V_{DD} must be at the same potential. Likewise, V_{SSAD1} and V_{SS} must be at the same potential.

14.3 Register Definition

The ATD has seven registers that control ATD functions.

Refer to the direct-page register summary in the memory chapter of this data sheet for the absolute address assignments for all ATD 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.

14.3.1 ATD Control (ATDC)

Writes to the ATD control register will abort the current conversion, but will not start a new conversion.

Table 14-3. Available Result Data Formats

RES8	DJM	SGN	Data Formats of Result	Analog Input $V_{REFH} = V_{DDA}$, $V_{REFL} = V_{SSA}$ ATDRH:ATDRL	
				V_{DDA}	V_{SSA}
1	0	0	8-bit : left justified : unsigned	$FF:$00	$00:$00
1	0	1	8-bit : left justified : signed	$7F:$00	$80:$00
1	1	X[1]	8-bit : left justified[2] : unsigned	$FF:$00	$00:$00
0	0	0	10-bit : left justified : unsigned	$FF:$C0	$00:$00
0	0	1	10-bit : left justified : signed	$7F:$C0	$80:$00
0	1	X[1]	10-bit : right justified : unsigned	$03:$FF	$00:$00

1 The SGN bit is only effective when DJM = 0. When DJM = 1, SGN is ignored.
2 8-bit results are always in ATDRH.

Table 14-4. Clock Prescaler Values

PRS	Factor = (PRS +1) × 2	Max Bus Clock MHz (2 MHz max ATD Clock)[1]	Max Bus Clock MHz (1 MHz max ATD Clock)[2]	Min Bus Clock[3] MHz (500 kHz min ATD Clock)
0000	2	4	2	1
0001	4	8	4	2
0010	6	12	6	3
0011	8	16	8	4
0100	10	20	10	5
0101	12	20	12	6
0110	14	20	14	7
0111	16	20	16	8
1000	18	20	18	9
1001	20	20	20	10
1010	22	20	20	11
1011	24	20	20	12
1100	26	20	20	13
1101	28	20	20	14
1110	30	20	20	15
1111	32	20	20	16

1 Maximum ATD conversion clock frequency is 2 MHz. The max bus clock frequency is computed from the max ATD conversion clock frequency times the indicated prescaler setting; i.e., for a PRS of 0, max bus clock = 2 (max ATD conversion clock frequency) × 2 (Factor) = 4 MHz.

2 Use these settings if the maximum desired ATD conversion clock frequency is 1 MHz. The max bus clock frequency is computed from the max ATD conversion clock frequency times the indicated prescaler setting; i.e., for a PRS of 0, max bus clock = 1 (max ATD conversion clock frequency) × 2 (Factor) = 2 MHz.

3 Minimum ATD conversion clock frequency is 500 kHz. The min bus clock frequency is computed from the min ATD conversion clock frequency times the indicated prescaler setting; i.e., for a PRS of 1, min bus clock = 0.5 (min ATD conversion clock frequency) × 2 (Factor) = 1 MHz.

A.8 ATD Characteristics

Table A-8. ATD Electrical Characteristics (Operating)

No.	Characteristic	Condition	Symbol	Min	Typ	Max	Unit
1	ATD supply[1]		V_{DDAD}	1.80	—	3.6	V
2	ATD supply current	Enabled	$I_{DDADrun}$	—	0.7	1.2	mA
		Disabled (ATDPU = 0 or STOP)	$I_{DDADstop}$	—	0.02	0.6	µA
3	Differential supply voltage	V_{DD}–V_{DDAD}	$\lvert V_{DDLT}\rvert$	—	—	100	mV
4	Differential ground voltage	V_{SS}–V_{SSAD}	$\lvert V_{SDLT}\rvert$	—	—	100	mV
5	Reference potential, low		$\lvert V_{REFL}\rvert$	—	—	V_{SSAD}	V
	Reference potential, high	$2.08V \le V_{DDAD} \le 3.6V$	V_{REFH}	2.08	—	V_{DDAD}	V
		$1.80V \le V_{DDAD} < 2.08V$		V_{DDAD}	—	V_{DDAD}	
6	Reference supply current (V_{REFH} to V_{REFL})	Enabled	I_{REF}	—	200	300	µA
		Disabled (ATDPU = 0 or STOP)	I_{REF}	—	<0.01	0.02	
7	Analog input voltage[2]		V_{INDC}	V_{SSAD} – 0.3	—	V_{DDAD} + 0.3	V

1 V_{DDAD} must be at same potential as V_{DD}.
2 Maximum electrical operating range, not valid conversion range.

Table A-9. ATD Timing/Performance Characteristics[1]

No.	Characteristic	Condition	Symbol	Min	Typ	Max	Unit
1	ATD conversion clock frequency	$2.08V \le V_{DDAD} \le 3.6V$	f_{ATDCLK}	0.5	—	2.0	MHz
		$1.80V \le V_{DDAD} < 2.08V$		0.5	—	1.0	
2	Conversion cycles (continuous convert)[2]		CC	28	28	<30	ATDCLK cycles
3	Conversion time (Including sample time)	$2.08V \le V_{DDAD} \le 3.6V$	T_{conv}	14.0	—	60.0	µs
		$1.80V \le V_{DDAD} < 2.08V$		28.0	—	60.0	
4	ATD sample time	t_{ADS}	t_{ADS}	—	14	—	ATDCLK cycles
5	Source impedance at input[3]		R_{AS}	—	—	10	kΩ
6	Analog Input Voltage[4]		V_{AIN}	V_{REFL}		V_{REFH}	V

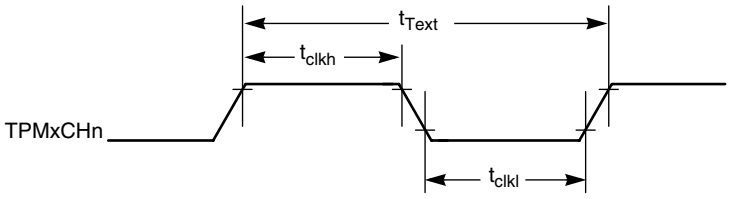

Figure A-15. Timer External Clock

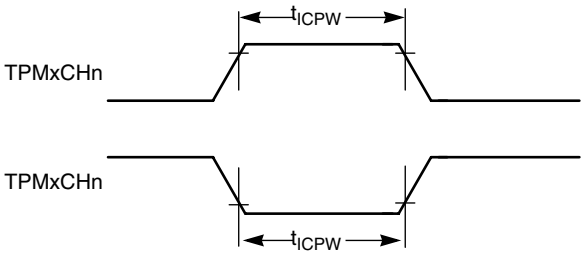

Figure A-16. Timer Input Capture Pulse