



Welcome to [E-XFL.COM](https://www.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	20MHz
Connectivity	I ² C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	12
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 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	16-VQFN Exposed Pad
Supplier Device Package	16-QFN-EP (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08qg8cffe

Section Number	Title	Page
8.1.4	Modes of Operation	109
8.1.5	Block Diagram	109
8.2	External Signal Description	111
8.3	Register Definition	111
8.3.1	ACMP Status and Control Register (ACMPSC)	112
8.4	Functional Description	113

Chapter 9

Analog-to-Digital Converter (S08ADC10V1)

9.1	Introduction	115
9.1.1	Module Configurations	117
9.1.2	Features	119
9.1.3	Block Diagram	119
9.2	External Signal Description	120
9.2.1	Analog Power (V_{DDAD})	121
9.2.2	Analog Ground (V_{SSAD})	121
9.2.3	Voltage Reference High (V_{REFH})	121
9.2.4	Voltage Reference Low (V_{REFL})	121
9.2.5	Analog Channel Inputs (ADx)	121
9.3	Register Definition	121
9.3.1	Status and Control Register 1 (ADCSC1)	121
9.3.2	Status and Control Register 2 (ADCSC2)	123
9.3.3	Data Result High Register (ADCRH)	124
9.3.4	Data Result Low Register (ADCRL)	124
9.3.5	Compare Value High Register (ADCCVH)	125
9.3.6	Compare Value Low Register (ADCCVL)	125
9.3.7	Configuration Register (ADCCFG)	125
9.3.8	Pin Control 1 Register (APCTL1)	127
9.3.9	Pin Control 2 Register (APCTL2)	128
9.3.10	Pin Control 3 Register (APCTL3)	129
9.4	Functional Description	130
9.4.1	Clock Select and Divide Control	130
9.4.2	Input Select and Pin Control	131
9.4.3	Hardware Trigger	131
9.4.4	Conversion Control	131
9.4.5	Automatic Compare Function	134
9.4.6	MCU Wait Mode Operation	134
9.4.7	MCU Stop3 Mode Operation	134
9.4.8	MCU Stop1 and Stop2 Mode Operation	135
9.5	Initialization Information	135
9.5.1	ADC Module Initialization Example	135
9.6	Application Information	137

Section Number	Title	Page
9.6.1	External Pins and Routing	137
9.6.2	Sources of Error	139

Chapter 10 Internal Clock Source (S08ICSV1)

10.1	Introduction	143
10.1.1	Module Configuration	143
10.1.2	Factory Trim Value	143
10.1.3	Features	145
10.1.4	Modes of Operation	145
10.1.5	Block Diagram	146
10.2	External Signal Description	147
10.3	Register Definition	147
10.3.1	ICS Control Register 1 (ICSC1)	147
10.3.2	ICS Control Register 2 (ICSC2)	148
10.3.3	ICS Trim Register (ICSTRM)	149
10.3.4	ICS Status and Control (ICSSC)	149
10.4	Functional Description	150
10.4.1	Operational Modes	150
10.4.2	Mode Switching	152
10.4.3	Bus Frequency Divider	152
10.4.4	Low Power Bit Usage	153
10.4.5	Internal Reference Clock	153
10.4.6	Optional External Reference Clock	153
10.4.7	Fixed Frequency Clock	153

Chapter 11 Inter-Integrated Circuit (S08IICV1)

11.1	Introduction	155
11.1.1	Module Configuration	155
11.1.2	Features	157
11.1.3	Modes of Operation	157
11.1.4	Block Diagram	158
11.2	External Signal Description	158
11.2.1	SCL — Serial Clock Line	158
11.2.2	SDA — Serial Data Line	158
11.3	Register Definition	158
11.3.1	IIC Address Register (IICA)	159
11.3.2	IIC Frequency Divider Register (IICF)	159
11.3.3	IIC Control Register (IICC)	162
11.3.4	IIC Status Register (IICS)	163
11.3.5	IIC Data I/O Register (IICD)	164

Table 1-2 provides the functional versions of the on-chip modules.

Table 1-2. Versions of On-Chip Modules

Module	Version
Analog Comparator (ACMP)	2
Analog-to-Digital Converter (ADC)	1
Central Processing Unit (CPU)	2
IIC Module (IIC)	1
Internal Clock Source (ICS)	1
Keyboard Interrupt (KBI)	2
Modulo Timer (MTIM)	1
Serial Communications Interface (SCI)	3
Serial Peripheral Interface (SPI)	3
Timer Pulse-Width Modulator (TPM)	2
Low-Power Oscillator (XOSC)	1
Debug Module (DBG)	2

System Clock Distribution

Figure 1-2 shows a simplified clock connection diagram. Some modules in the MCU have selectable clock inputs as shown. The clock inputs to the modules indicate the clock(s) that are used to drive the module function. All memory mapped registers associated with the modules are clocked with BUSCLK.

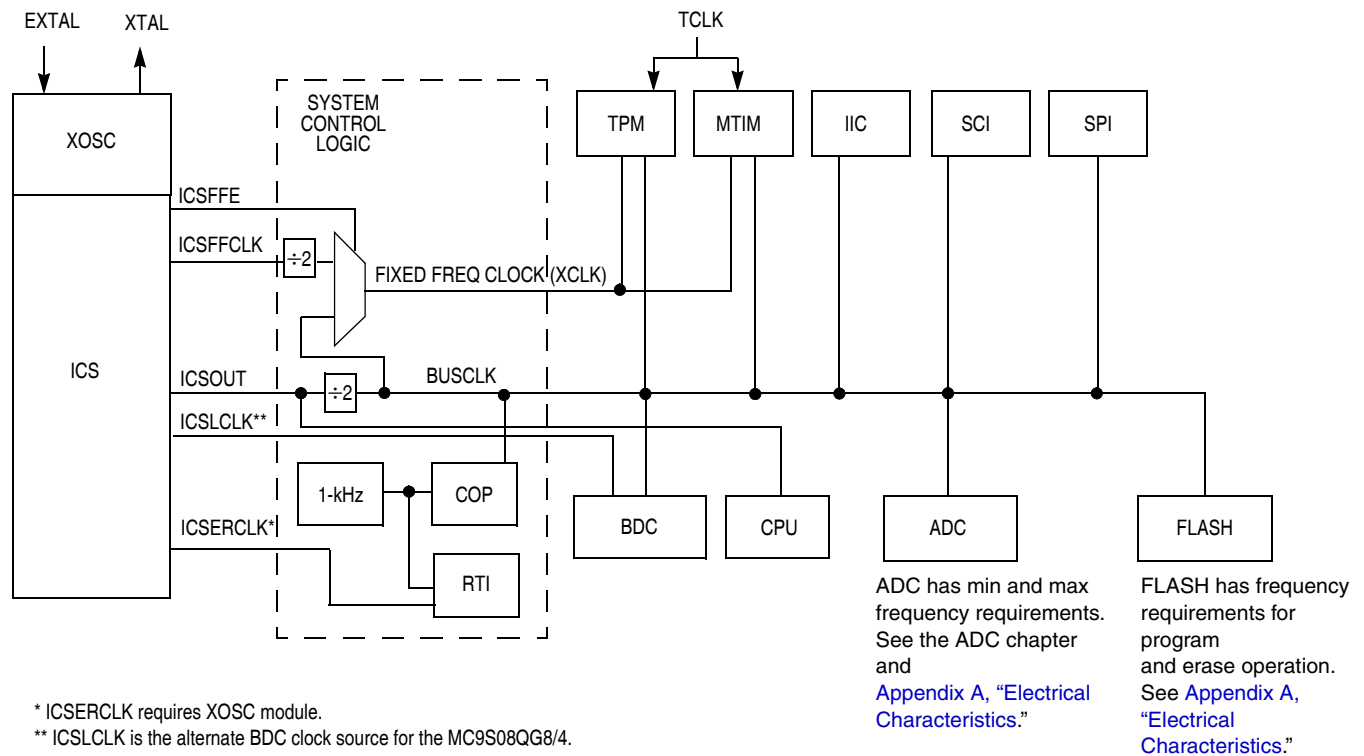


Figure 1-2. System Clock Distribution Diagram

Table 4-12. FSTAT Register Field Descriptions (continued)

Field	Description
4 FACCERR	<p>Access Error Flag — FACCERR is set automatically when the proper command sequence is not obeyed 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 4.5.5, “Access Errors.” FACCERR is cleared by writing a 1 to FACCERR. Writing a 0 to FACCERR has no meaning or effect.</p> <p>0 No access error. 1 An access error has occurred.</p>
2 FBLANK	<p>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.</p> <p>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).</p>

4.7.6 FLASH Command Register (FCMD)

Only five command codes are recognized in normal user modes as shown in [Table 4-13](#). Refer to [Section 4.5.3, “Program and Erase Command Execution,”](#) for a detailed discussion of FLASH programming and erase operations.

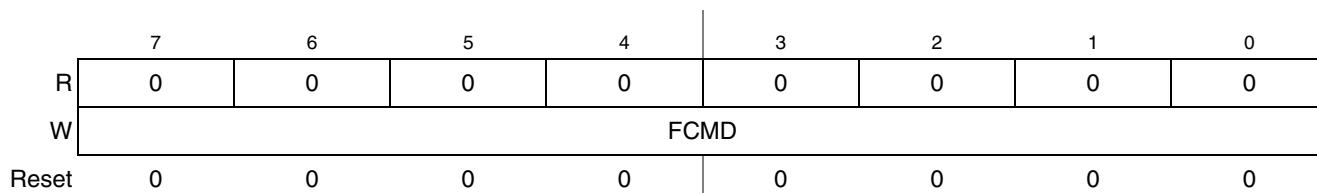


Figure 4-10. FLASH Command Register (FCMD)

Table 4-13. FLASH Commands

Command	FCMD	Equate File Label
Blank check	0x05	mBlank
Byte program	0x20	mByteProg
Byte program — burst mode	0x25	mBurstProg
Page erase (512 bytes/page)	0x40	mPageErase
Mass erase (all FLASH)	0x41	mMassErase

All other command codes are illegal and generate an access error.

It is not necessary to perform a blank check command after a mass erase operation. The blank check command is only required as part of the security unlocking mechanism.

When the 1-kHz clock source is selected, the COP counter is re-initialized to zero upon entry to stop mode. The COP counter begins from zero after the MCU exits stop mode.

5.5 Interrupts

Interrupts provide a way to save the current CPU status and registers, execute an interrupt service routine (ISR), and then restore the CPU status so processing resumes where it was before the interrupt. Other than the software interrupt (SWI), which is a program instruction, interrupts are caused by hardware events such as an edge on the $\overline{\text{IRQ}}$ pin or a timer-overflow event. The debug module can also generate an SWI under certain circumstances.

If an event occurs in an enabled interrupt source, an associated read-only status flag will become set. The CPU will not respond until and unless the local interrupt enable is a 1 to enable the interrupt. The I bit 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 obeys 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 can 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 from the stack.

NOTE

For compatibility with M68HC08 devices, 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 immediately 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 5-2](#)).

The $\overline{\text{IRQ}}$ pin, when enabled, defaults to use an internal pullup device ($\text{IRQPDD} = 0$). If the user desires to use an external pullup, the IRQPDD can be written to a 1 to turn off the internal device.

BIH and BIL instructions may be used to detect the level on the $\overline{\text{IRQ}}$ pin when the pin is configured to act as the IRQ input.

NOTE

This pin does not contain a clamp diode to V_{DD} and should not be driven above V_{DD} .

The voltage measured on the internally pulled-up $\overline{\text{IRQ}}$ pin will not be pulled to V_{DD} . The internal gates connected to this pin are pulled to V_{DD} . The $\overline{\text{IRQ}}$ pullup should not be used to pull up components external to the MCU. The internal gates connected to this pin are pulled all the way to V_{DD} .

5.5.2.2 Edge and Level Sensitivity

The IRQMOD control bit reconfigures 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 $\overline{\text{IRQ}}$ pin changes from the deasserted to the asserted level), but the flag is continuously set (and cannot be cleared) as long as the $\overline{\text{IRQ}}$ pin remains at the asserted level.

5.5.3 Interrupt Vectors, Sources, and Local Masks

Table 5-2 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.

5.6 Low-Voltage Detect (LVD) System

The MC9S08QG8/4 includes a system to protect against low voltage conditions to protect memory contents and control MCU system states during supply voltage variations. The system is comprised of a power-on reset (POR) circuit and a LVD circuit with a user selectable trip voltage, either high ($V_{LV\text{DH}}$) or low ($V_{LV\text{DL}}$). The LVD circuit is enabled when LVDE in SPMSC1 is high and the trip voltage is selected by LVDV in SPMSC3. The LVD is disabled upon entering any of the stop modes unless LVDSE is set in SPMSC1. If LVDSE and LVDE are both set, then the MCU cannot enter stop1 or stop2, and the current consumption in stop3 with the LVD enabled will be greater.

5.6.1 Power-On Reset Operation

When power is initially applied to the MCU, or when the supply voltage drops below the V_{POR} level, the POR circuit will cause a reset condition. As the supply voltage rises, the LVD circuit will hold the MCU in reset until the supply has risen above the $V_{LV\text{DL}}$ level. Both the POR bit and the LVD bit in SRS are set following a POR.

5.6.2 LVD Reset Operation

The LVD can be configured to generate a reset upon detection of a low voltage condition by setting LVDRE to 1. After an LVD reset has occurred, the LVD system will hold the MCU in reset until the supply voltage has risen above the level determined by LVDV. The LVD bit in the SRS register is set following either an LVD reset or POR.

5.6.3 LVD Interrupt Operation

When a low voltage condition is detected and the LVD circuit is configured using SPMSC1 for interrupt operation (LVDE set, LVDIE set, and LVDRE clear), then LVDF in SPMSC1 will be set and an LVD interrupt request will occur.

5.6.4 Low-Voltage Warning (LVW)

The LVD system has a low voltage warning flag to indicate to the user that the supply voltage is approaching, but is above, the LVD voltage. The LVW does not have an interrupt associated with it. There are two user selectable trip voltages for the LVW, one high ($V_{LV\text{WH}}$) and one low ($V_{LV\text{WL}}$). The trip voltage is selected by LVWV in SPMSC3.

5.7 Real-Time Interrupt (RTI)

The real-time interrupt function can be used to generate periodic interrupts. The RTI can accept two sources of clocks, the 1-kHz internal clock or an external clock if available. External clock input requires the XOSC module; consult [Table 1-1](#) to see if your MCU contains this module. The RTICLK bit in SRTISC is used to select the RTI clock source.

Either RTI clock source can be used when the MCU is in run, wait or stop3 mode. When using the external oscillator in stop3, it must be enabled in stop (EREFSTEN = 1) and configured for low frequency operation

Chapter 6

Parallel Input/Output Control

This section explains software controls related to parallel input/output (I/O) and pin control. The MC9S08QG8 has two parallel I/O ports which include a total of 12 I/O pins, one output-only pin and one input-only pin. See [Section Chapter 2, “External Signal Description,”](#) for more information about pin assignments and external hardware considerations of these pins. Not all pins are available on all devices of the MC9S08QG8/4 Family; see [Table 1-1](#) for the number of general-purpose pins available on your device.

All of these I/O pins are shared with on-chip peripheral functions as shown in [Table 2-2](#). The peripheral modules have priority over the I/Os so that when a peripheral is enabled, the I/O functions associated with the shared pins are disabled. After reset, the shared peripheral functions are disabled so that the pins are controlled by the I/O. All of the I/Os are configured as inputs ($PTxDDn = 0$) with pullup devices disabled ($PTxPEn = 0$), except for output-only pin PTA4 which defaults to the BKGD/MS pin.

NOTE

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

6.1 Port Data and Data Direction

Reading and writing of parallel I/Os is performed through the port data registers. The direction, either input or output, is controlled through the port data direction registers. The parallel I/O port function for an individual pin is illustrated in the block diagram shown in [Figure 6-1](#).

6.4.2.1 Port A Internal Pullup Enable (PTAPE)

An internal pullup device can be enabled for each port pin by setting the corresponding bit in the pullup enable register (PTAPEn). The pullup device is disabled if the pin is configured as an output by the parallel I/O control logic or any shared peripheral function regardless of the state of the corresponding pullup enable register bit. The pullup device is also disabled if the pin is controlled by an analog function.

	7	6	5	4	3	2	1	0
R	0	0	PTAPE5	PTAPE4 ¹	PTAPE3	PTAPE2	PTAPE1	PTAPE0
W								
Reset:	0	0	0	0	0	0	0	0

Figure 6-4. Internal Pullup Enable for Port A Register (PTAPE)

¹ PTAPE4 has no effect on the output-only PTA4 pin.

Table 6-3. PTAPE Register Field Descriptions

Field	Description
5:0 PTAPE[5:0]	Internal Pullup Enable for Port A Bits — Each of these control bits determines if the internal pullup device is enabled for the associated PTA pin. For port A pins that are configured as outputs, these bits have no effect and the internal pullup devices are disabled. 0 Internal pullup device disabled for port A bit n. 1 Internal pullup device enabled for port A bit n.

6.4.2.2 Port A Slew Rate Enable (PTASE)

Slew rate control can be enabled for each port pin by setting the corresponding bit in the slew rate control register (PTASEn). When enabled, slew control limits the rate at which an output can transition to reduce EMC emissions. Slew rate control has no effect on pins which are configured as inputs.

	7	6	5	4	3	2	1	0
R	0	0	PTASE5 ¹	PTASE4	PTASE3	PTASE2	PTASE1	PTASE0
W								
Reset:	0	0	1	1	1	1	1	1

Figure 6-6. Slew Rate Enable for Port A Register (PTASE)

¹ PTASE5 has no effect on the input-only PTA5 pin.

Table 6-4. PTASE Register Field Descriptions

Field	Description
5:0 PTASE[5:0]	Output Slew Rate Enable for Port A Bits — Each of these control bits determines if the output slew rate control is enabled for the associated PTA pin. For port A pins that are configured as inputs, these bits have no effect. 0 Output slew rate control disabled for port A bit n. 1 Output slew rate control enabled for port A bit n.

6.4.4 Port B Control Registers

The pins associated with port B are controlled by the registers in this section. These registers control the pin pullup, slew rate, and drive strength of the port B pins independent of the parallel I/O register.

6.4.4.1 Port B Internal Pullup Enable (PTBPE)

An internal pullup device can be enabled for each port pin by setting the corresponding bit in the pullup enable register (PTBPE_n). The pullup device is disabled if the pin is configured as an output by the parallel I/O control logic or any shared peripheral function regardless of the state of the corresponding pullup enable register bit. The pullup device is also disabled if the pin is controlled by an analog function.

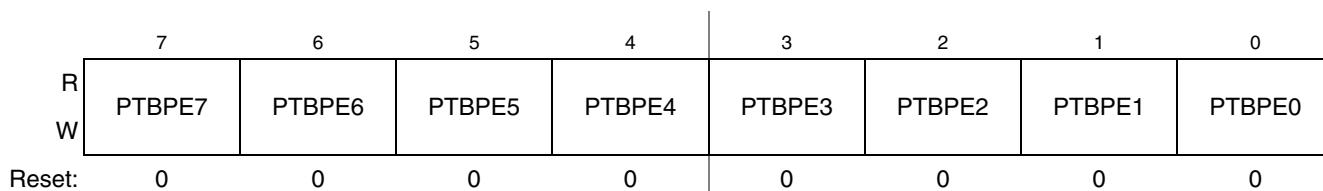


Figure 6-12. Internal Pullup Enable for Port B Register (PTBPE)

Table 6-8. PTBPE Register Field Descriptions

Field	Description
7:0 PTBPE[7:0]	Internal Pullup Enable for Port B Bits — Each of these control bits determines if the internal pullup device is enabled for the associated PTB pin. For port B pins that are configured as outputs, these bits have no effect and the internal pullup devices are disabled. 0 Internal pullup device disabled for port B bit n. 1 Internal pullup device enabled for port B bit n.

6.4.4.2 Port B Slew Rate Enable (PTBSE)

Slew rate control can be enabled for each port pin by setting the corresponding bit in the slew rate control register (PTBSE_n). When enabled, slew control limits the rate at which an output can transition in order to reduce EMC emissions. Slew rate control has no effect on pins which are configured as input.

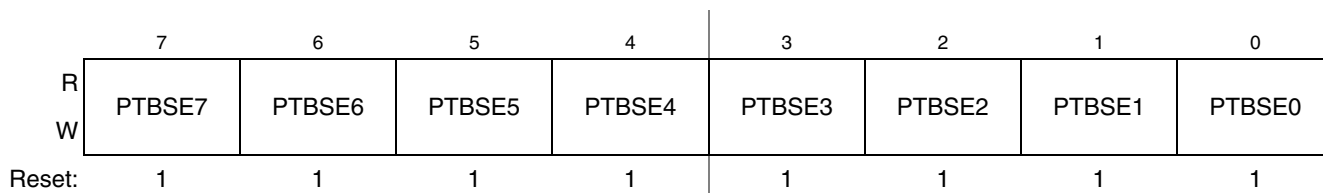


Figure 6-14. Slew Rate Enable for Port B Register (PTBSE)

Table 6-9. PTBSE Register Field Descriptions

Field	Description
7:0 PTBSE[7:0]	Output Slew Rate Enable for Port B Bits — Each of these control bits determines if the output slew rate control is enabled for the associated PTB pin. For port B pins that are configured as inputs, these bits have no effect. 0 Output slew rate control disabled for port B bit n. 1 Output slew rate control enabled for port B bit n.

9.4.4.2 Completing Conversions

A conversion is completed when the result of the conversion is transferred into the data result registers, ADCRH and ADCRL. This is indicated by the setting of COCO. An interrupt is generated if AIEN is high at the time that COCO is set.

A blocking mechanism prevents a new result from overwriting previous data in ADCRH and ADCRL if the previous data is in the process of being read while in 10-bit MODE (the ADCRH register has been read but the ADCRL register has not). When blocking is active, the data transfer is blocked, COCO is not set, and the new result is lost. In the case of single conversions with the compare function enabled and the compare condition false, blocking has no effect and ADC operation is terminated. In all other cases of operation, when a data transfer is blocked, another conversion is initiated regardless of the state of ADCO (single or continuous conversions enabled).

If single conversions are enabled, the blocking mechanism could result in several discarded conversions and excess power consumption. To avoid this issue, the data registers must not be read after initiating a single conversion until the conversion completes.

9.4.4.3 Aborting Conversions

Any conversion in progress will be aborted when:

- A write to ADCSC1 occurs (the current conversion will be aborted and a new conversion will be initiated, if ADCH are not all 1s).
- A write to ADCSC2, ADCCFG, ADCCVH, or ADCCVL occurs. This indicates a mode of operation change has occurred and the current conversion is therefore invalid.
- The MCU is reset.
- The MCU enters stop mode with ADACK not enabled.

When a conversion is aborted, the contents of the data registers, ADCRH and ADCRL, are not altered but continue to be the values transferred after the completion of the last successful conversion. In the case that the conversion was aborted by a reset, ADCRH and ADCRL return to their reset states.

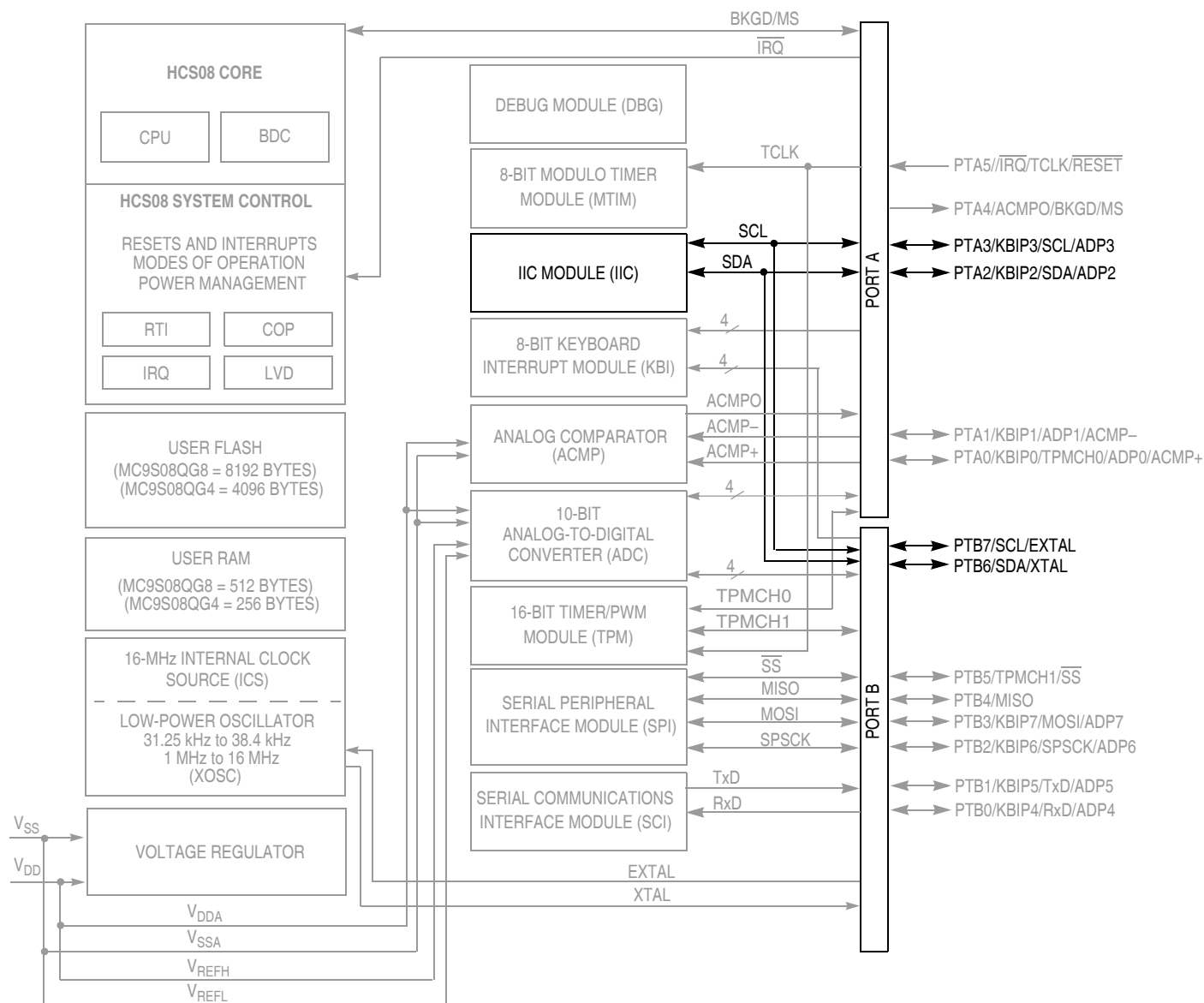
9.4.4.4 Power Control

The ADC module remains in its idle state until a conversion is initiated. If ADACK is selected as the conversion clock source, the ADACK clock generator is also enabled.

Power consumption when active can be reduced by setting ADLPC. This results in a lower maximum value for f_{ADCK} (see the electrical specifications).

9.4.4.5 Total Conversion Time

The total conversion time depends on the sample time (as determined by ADLSMP), the MCU bus frequency, the conversion mode (8-bit or 10-bit), and the frequency of the conversion clock (f_{ADCK}). After the module becomes active, sampling of the input begins. ADLSMP is used to select between short and long sample times. When sampling is complete, the converter is isolated from the input channel and a successive approximation algorithm is performed to determine the digital value of the analog signal. The



NOTES:

- 1 Not all pins or pin functions are available on all devices, see Table 1-1 for available functions on each device.
- 2 Port pins are software configurable with pullup device if input port.
- 3 Port pins are software configurable for output drive strength.
- 4 Port pins are software configurable for output slew rate control.
- 5 $\overline{\text{IRQ}}$ contains a software configurable (IRQPDD) pullup device if PTA5 enabled as $\overline{\text{IRQ}}$ pin function (IRQPE = 1).
- 6 $\overline{\text{RESET}}$ contains integrated pullup device if PTA5 enabled as reset pin function (RSTPE = 1).
- 7 PTA4 contains integrated pullup device if BKGD enabled (BKGDPPE = 1).
- 8 SDA and SCL pin locations can be repositioned under software control (IICPS), defaults on PTA2 and PTA3.
- 9 When pin functions as KBI (KBIPEn = 1) and associated pin is configured to enable the pullup device, KBEDGn can be used to reconfigure the pullup as a pulldown device.

Figure 11-1. MC9S08QG8/4 Block Diagram Highlighting IIC Block and Pins

11.1.2 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

11.1.3 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. Stop2 will reset the register contents.

13.3.3 MTIM Counter Register (MTIMCNT)

MTIMCNT is the read-only value of the current MTIM count of the 8-bit counter.

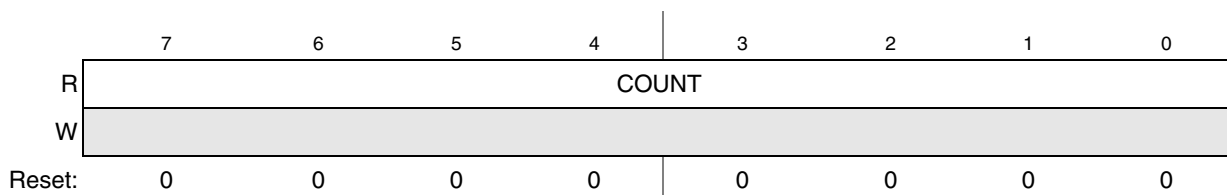


Figure 13-6. MTIM Counter Register

Table 13-4. MTIM Counter Register Field Description

Field	Description
7:0 COUNT	MTIM Count — These eight read-only bits contain the current value of the 8-bit counter. Writes have no effect to this register. Reset clears the count to \$00.

13.3.4 MTIM Modulo Register (MTIMMOD)

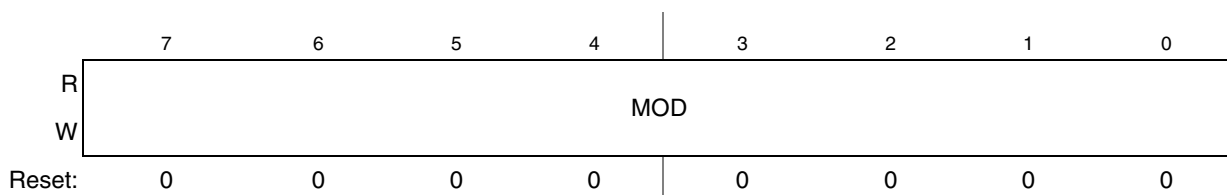


Figure 13-7. MTIM Modulo Register

Table 13-5. MTIM Modulo Register Field Descriptions

Field	Description
7:0 MOD	MTIM Modulo — These eight read/write bits contain the modulo value used to reset the count and set TOF. A value of \$00 puts the MTIM in free-running mode. Writing to MTIMMOD resets the COUNT to \$00 and clears TOF. Reset sets the modulo to \$00.

Figure 14-4 shows the receiver portion of the SCI.

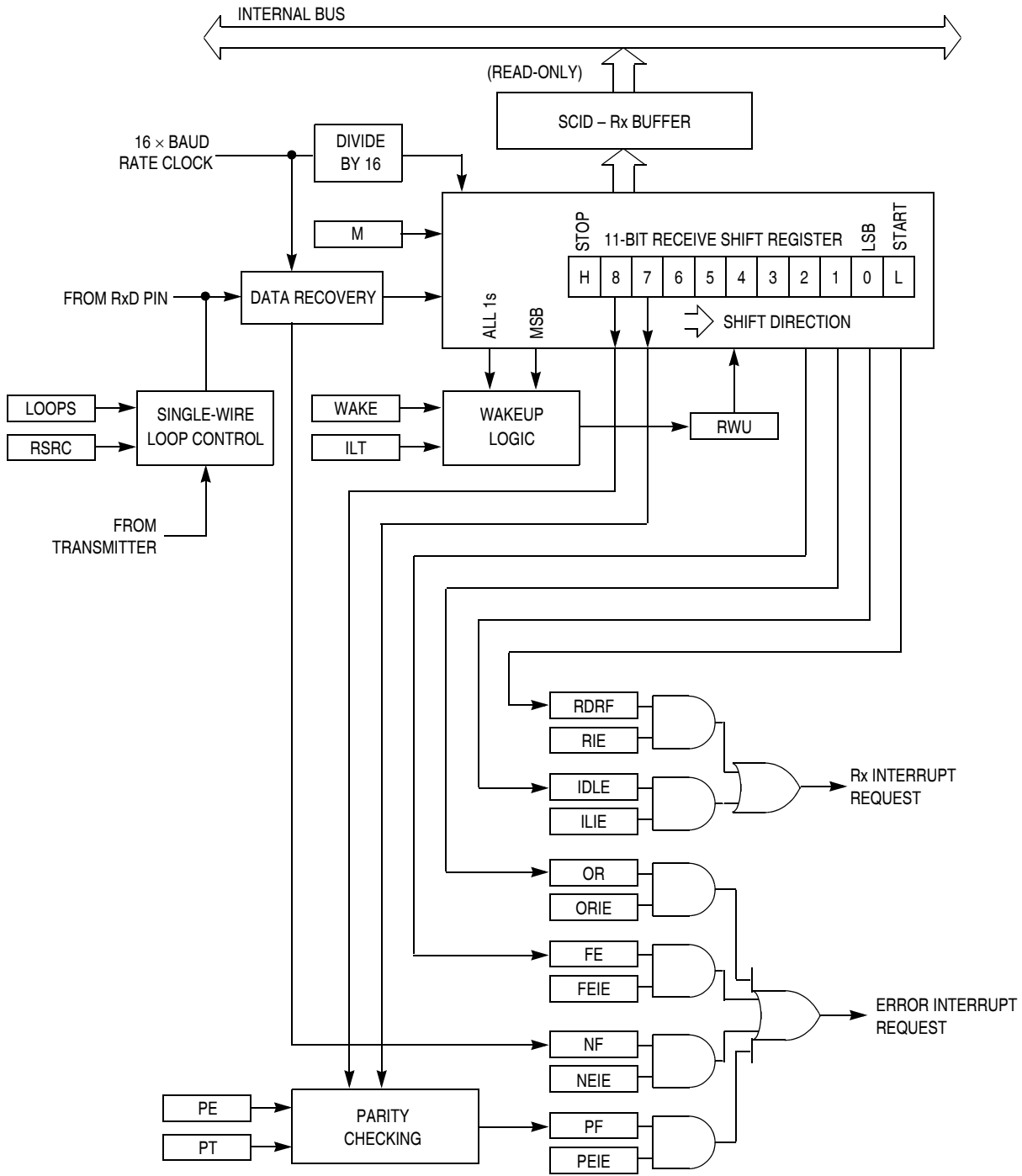


Figure 14-4. SCI Receiver Block Diagram

All TPM channels are programmable independently as input capture, output compare, or buffered edge-aligned PWM channels.

16.2 External Signal Description

When any pin associated with the timer is configured as a timer input, a passive pullup can be enabled. After reset, the TPM modules are disabled and all pins default to general-purpose inputs with the passive pullups disabled.

16.2.1 External TPM Clock Sources

When control bits CLKSB:CLKSA in the timer status and control register are set to 1:1, the prescaler and consequently the 16-bit counter for TPM are driven by an external clock source, TPMxCLK, connected to an I/O pin. A synchronizer is needed between the external clock and the rest of the TPM. This synchronizer is clocked by the bus clock so the frequency of the external source must be less than one-half the frequency of the bus rate clock. The upper frequency limit for this external clock source is specified to be one-fourth the bus frequency to conservatively accommodate duty cycle and phase-locked loop (PLL) or frequency-locked loop (FLL) frequency jitter effects.

On some devices the external clock input is shared with one of the TPM channels. When a TPM channel is shared as the external clock input, the associated TPM channel cannot use the pin. (The channel can still be used in output compare mode as a software timer.) Also, if one of the TPM channels is used as the external clock input, the corresponding ELSnB:ELSnA control bits must be set to 0:0 so the channel is not trying to use the same pin.

16.2.2 TPMCHn — TPM Channel n I/O Pins

Each TPM channel is associated with an I/O pin on the MCU. The function of this pin depends on the configuration of the channel. In some cases, no pin function is needed so the pin reverts to being controlled by general-purpose I/O controls. When a timer has control of a port pin, the port data and data direction registers do not affect the related pin(s). See the [Pins and Connections](#) chapter for additional information about shared pin functions.

16.3 Register Definition

The TPM includes:

- An 8-bit status and control register (TPMSC)
- A 16-bit counter (TPMCNTH:TPMCNTL)
- A 16-bit modulo register (TPMMODH:TPMMODL)

Each timer channel has:

- An 8-bit status and control register (TPMCnSC)
- A 16-bit channel value register (TPMCnVH:TPMCnVL)

Refer to the direct-page register summary in the [Memory](#) chapter of this data sheet for the absolute address assignments for all TPM registers. This section refers to registers and control bits only by their names. A

17.3 On-Chip Debug System (DBG)

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

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

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

17.3.1 Comparators A and B

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

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

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

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

17.3.2 Bus Capture Information and FIFO Operation

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

A.11 FLASH Specifications

This section provides details about program/erase times and program-erase endurance for the FLASH memory.

Program and erase operations do not require any special power sources other than the normal V_{DD} supply. For more detailed information about program/erase operations, see the Memory section.

Table A-15. FLASH Characteristics

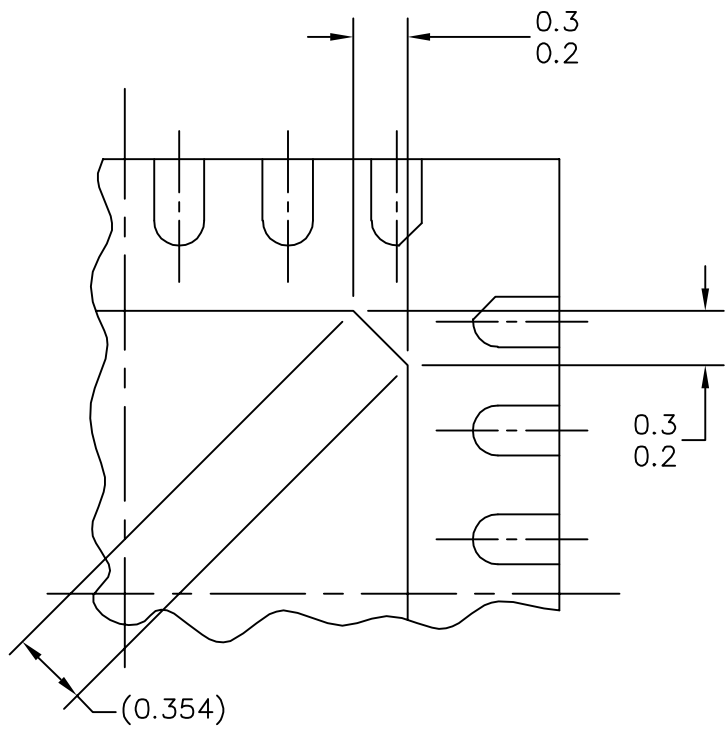
Characteristic	Symbol	Min	Typical	Max	Unit
Supply voltage for program/erase: $T \leq 85^{\circ}\text{C}$ $T > 85^{\circ}\text{C}$	$V_{\text{prog/erase}}$	1.8 2.1	— —	3.6 3.6	V
Supply voltage for read operation	V_{Read}	1.8	—	3.6	V
Internal FCLK frequency ¹	f_{FCLK}	150	—	200	kHz
Internal FCLK period (1/FCLK)	t_{FcyC}	5	—	6.67	μs
Byte program time (random location) ⁽²⁾	t_{prog}	9			t_{FcyC}
Byte program time (burst mode) ⁽²⁾	t_{Burst}	4			t_{FcyC}
Page erase time ²	t_{Page}	4000			t_{FcyC}
Mass erase time ⁽²⁾	t_{Mass}	20,000			t_{FcyC}
Program/erase endurance ³ T_L to $T_H = -40^{\circ}\text{C}$ to $+125^{\circ}\text{C}$ $T = 25^{\circ}\text{C}$		10,000	— 100,000	— —	cycles
Data retention ⁴	$t_{\text{D_ret}}$	15	100	—	years

¹ The frequency of this clock is controlled by a software setting.

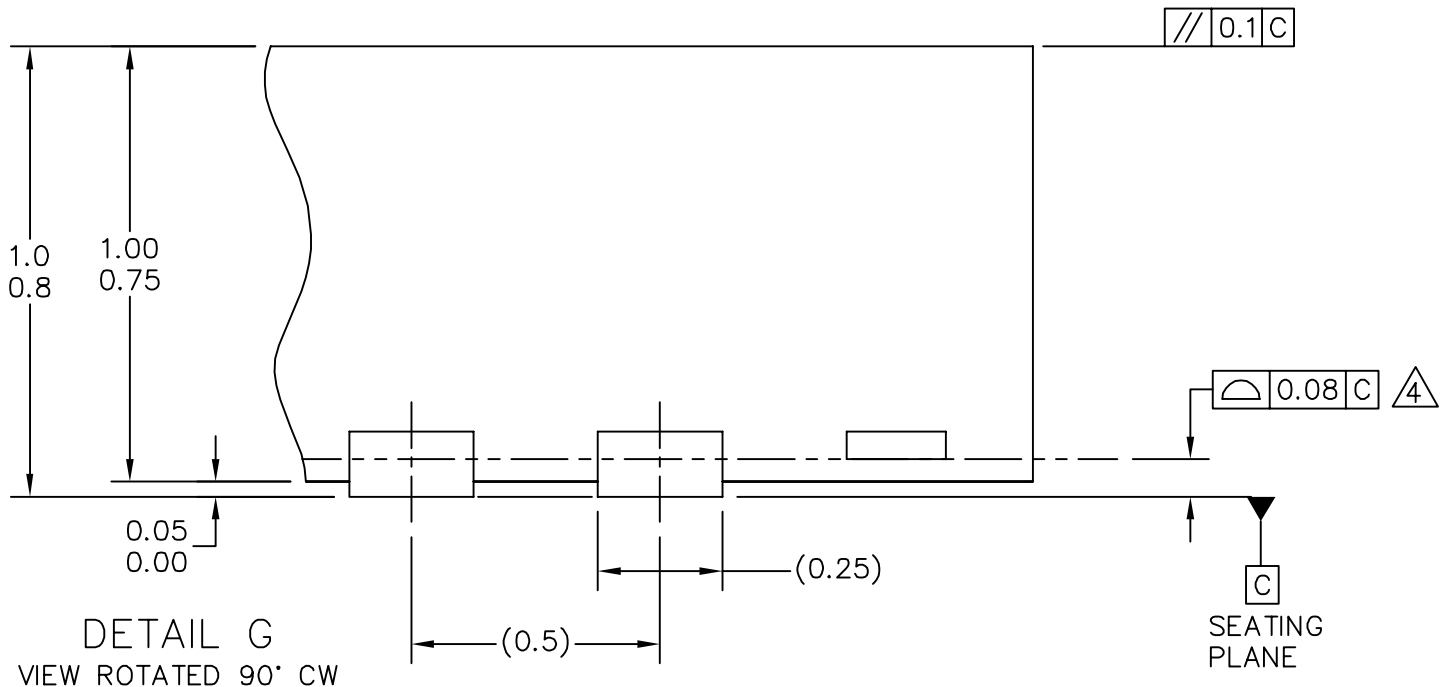
² These values are hardware state machine controlled. User code does not need to count cycles. This information supplied for calculating approximate time to program and erase.

³ **Typical endurance for FLASH** was evaluated for this product family on the 9S12Dx64. For additional information on how Motorola defines typical endurance, please refer to Engineering Bulletin EB619/D, *Typical Endurance for Nonvolatile Memory*.

⁴ **Typical data retention** values are based on intrinsic capability of the technology measured at high temperature and de-rated to 25°C using the Arrhenius equation. For additional information on how Motorola defines typical data retention, please refer to Engineering Bulletin EB618/D, *Typical Data Retention for Nonvolatile Memory*.



DETAIL M
PIN 1 BACKSIDE IDENTIFIER



DETAIL G
VIEW ROTATED 90° CW

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE	
TITLE: THERMALLY ENHANCED QUAD FLAT NON-LEADED PACKAGE (QFN) 24 TERMINAL, 0.5 PITCH (4 X 4 X 1)	DOCUMENT NO: 98ARL10605D	REV: 0	
	CASE NUMBER: 1897-01	08 SEP 2006	
	STANDARD: JEDEC MO-220 VGGD-8		