

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	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-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	16-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08qg8cdte

Contents

Section Number	Title	Page
Chapter 1		
Device Overview		
1.1	Introduction	19
1.1.1	Devices in the MC9S08QG8/4 Series	19
1.1.2	MCU Block Diagram	20
Chapter 2		
External Signal Description		
2.1	Device Pin Assignment	23
2.2	Recommended System Connections	25
2.2.1	Power	26
2.2.2	Oscillator (XOSC)	27
2.2.3	Reset (Input Only)	27
2.2.4	Background / Mode Select (BKGD/MS)	28
2.2.5	General-Purpose I/O and Peripheral Ports	28
Chapter 3		
Modes of Operation		
3.1	Introduction	33
3.2	Features	33
3.3	Run Mode	33
3.4	Active Background Mode	33
3.5	Wait Mode	34
3.6	Stop Modes	35
3.6.1	Stop3 Mode	35
3.6.2	Stop2 Mode	36
3.6.3	Stop1 Mode	37
3.6.4	On-Chip Peripheral Modules in Stop Modes	37
Chapter 4		
Memory Map and Register Definition		
4.1	MC9S08QG8/4 Memory Map	39
4.2	Reset and Interrupt Vector Assignments	40
4.3	Register Addresses and Bit Assignments	41
4.4	RAM	45
4.5	FLASH	46
4.5.1	Features	47

5.8.7 System Real-Time Interrupt Status and Control Register (SRTISC)

This high page register contains status and control bits for the RTI.

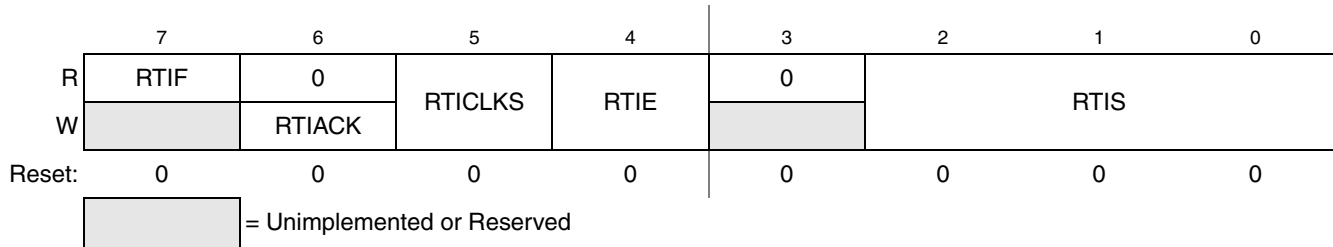


Figure 5-9. System RTI Status and Control Register (SRTISC)

Table 5-10. SRTISC Register Field Descriptions

Field	Description
7 RTIF	Real-Time Interrupt Flag — This read-only status bit indicates the periodic wakeup timer has timed out. 0 Periodic wakeup timer not timed out. 1 Periodic wakeup timer timed out.
6 RTIACK	Real-Time Interrupt Acknowledge — This write-only bit is used to acknowledge real-time interrupt request (write 1 to clear RTIF). Writing 0 has no meaning or effect. Reads always return 0.
5 RTICLKs	Real-Time Interrupt Clock Select — This read/write bit selects the clock source for the real-time interrupt. 0 Real-time interrupt request clock source is internal 1-kHz oscillator. 1 Real-time interrupt request clock source is external clock.
4 RTIE	Real-Time Interrupt Enable — This read-write bit enables real-time interrupts. 0 Real-time interrupts disabled. 1 Real-time interrupts enabled.
2:0 RTIS	Real-Time Interrupt Delay Selects — These read/write bits select the period for the RTI. See Table 5-11.

Table 5-11. Real-Time Interrupt Period

RTIS2:RTIS1:RTIS0	Using Internal 1-kHz Clock Source ^{1 2}	Using External Clock Source Period = t_{ext} ³
0:0:0	Disable RTI	Disable RTI
0:0:1	8 ms	$t_{ext} \times 256$
0:1:0	32 ms	$t_{ext} \times 1024$
0:1:1	64 ms	$t_{ext} \times 2048$
1:0:0	128 ms	$t_{ext} \times 4096$
1:0:1	256 ms	$t_{ext} \times 8192$
1:1:0	512 ms	$t_{ext} \times 16384$
1:1:1	1.024 s	$t_{ext} \times 32768$

¹ Values are shown in this column based on $t_{RTI} = 1$ ms. See t_{RTI} in the appendix Section A.8.1, “Control Timing,” for the tolerance of this value.

² The initial RTI timeout period will be up to one 1-kHz clock period less than the time specified.

³ t_{ext} is the period of the external crystal frequency.



7.3.5 Extended Addressing Mode (EXT)

In extended addressing mode, the full 16-bit address of the operand is located in the next two bytes of program memory after the opcode (high byte first).

7.3.6 Indexed Addressing Mode

Indexed addressing mode has seven variations including five that use the 16-bit H:X index register pair and two that use the stack pointer as the base reference.

7.3.6.1 Indexed, No Offset (IX)

This variation of indexed addressing uses the 16-bit value in the H:X index register pair as the address of the operand needed to complete the instruction.

7.3.6.2 Indexed, No Offset with Post Increment (IX+)

This variation of indexed addressing uses the 16-bit value in the H:X index register pair as the address of the operand needed to complete the instruction. The index register pair is then incremented ($H:X = H:X + 0x0001$) after the operand has been fetched. This addressing mode is only used for MOV and CBEQ instructions.

7.3.6.3 Indexed, 8-Bit Offset (IX1)

This variation of indexed addressing uses the 16-bit value in the H:X index register pair plus an unsigned 8-bit offset included in the instruction as the address of the operand needed to complete the instruction.

7.3.6.4 Indexed, 8-Bit Offset with Post Increment (IX1+)

This variation of indexed addressing uses the 16-bit value in the H:X index register pair plus an unsigned 8-bit offset included in the instruction as the address of the operand needed to complete the instruction. The index register pair is then incremented ($H:X = H:X + 0x0001$) after the operand has been fetched. This addressing mode is used only for the CBEQ instruction.

7.3.6.5 Indexed, 16-Bit Offset (IX2)

This variation of indexed addressing uses the 16-bit value in the H:X index register pair plus a 16-bit offset included in the instruction as the address of the operand needed to complete the instruction.

7.3.6.6 SP-Relative, 8-Bit Offset (SP1)

This variation of indexed addressing uses the 16-bit value in the stack pointer (SP) plus an unsigned 8-bit offset included in the instruction as the address of the operand needed to complete the instruction.

Table 7-2. . Instruction Set Summary (Sheet 2 of 9)

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR				
						VH	I	N	Z	C
BCLR <i>n,opr8a</i>	Clear Bit <i>n</i> in Memory ($M_n \leftarrow 0$)	DIR (b0)	11 dd	5	r fwpp	--	--	--	--	--
		DIR (b1)	13 dd	5	r fwpp					
		DIR (b2)	15 dd	5	r fwpp					
		DIR (b3)	17 dd	5	r fwpp					
		DIR (b4)	19 dd	5	r fwpp					
		DIR (b5)	1B dd	5	r fwpp					
		DIR (b6)	1D dd	5	r fwpp					
		DIR (b7)	1F dd	5	r fwpp					
BCS <i>rel</i>	Branch if Carry Bit Set (if $C = 1$) (Same as BLO)	REL	25 rr	3	ppp	--	--	--	--	--
BEQ <i>rel</i>	Branch if Equal (if $Z = 1$)	REL	27 rr	3	ppp	--	--	--	--	--
BGE <i>rel</i>	Branch if Greater Than or Equal To (if $N \oplus V = 0$) (Signed)	REL	90 rr	3	ppp	--	--	--	--	--
BGND	Enter active background if ENBDM=1 Waits for and processes BDM commands until GO, TRACE1, or TAGGO	INH	82	5+	fp . . . ppp	--	--	--	--	--
BGT <i>rel</i>	Branch if Greater Than (if $Z \mid (N \oplus V) = 0$) (Signed)	REL	92 rr	3	ppp	--	--	--	--	--
BHCC <i>rel</i>	Branch if Half Carry Bit Clear (if $H = 0$)	REL	28 rr	3	ppp	--	--	--	--	--
BHCS <i>rel</i>	Branch if Half Carry Bit Set (if $H = 1$)	REL	29 rr	3	ppp	--	--	--	--	--
BHI <i>rel</i>	Branch if Higher (if $C \mid Z = 0$)	REL	22 rr	3	ppp	--	--	--	--	--
BHS <i>rel</i>	Branch if Higher or Same (if $C = 0$) (Same as BCC)	REL	24 rr	3	ppp	--	--	--	--	--
BIH <i>rel</i>	Branch if IRQ Pin High (if IRQ pin = 1)	REL	2F rr	3	ppp	--	--	--	--	--
BIL <i>rel</i>	Branch if IRQ Pin Low (if IRQ pin = 0)	REL	2E rr	3	ppp	--	--	--	--	--
BIT # <i>opr8i</i> BIT <i>opr8a</i> BIT <i>opr16a</i> BIT <i>opr16,X</i> BIT <i>opr8,X</i> BIT <i>,X</i> BIT <i>opr16,SP</i> BIT <i>opr8,SP</i>	Bit Test (A) & (M) (CCR Updated but Operands Not Changed)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A5 ii B5 dd C5 hh ll D5 ee ff E5 ff F5 9E D5 ee ff 9E E5 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	0-	-	↑	↓	-
BLE <i>rel</i>	Branch if Less Than or Equal To (if $Z \mid (N \oplus V) = 1$) (Signed)	REL	93 rr	3	ppp					
BLO <i>rel</i>	Branch if Lower (if $C = 1$) (Same as BCS)	REL	25 rr	3	ppp					
BLS <i>rel</i>	Branch if Lower or Same (if $C \mid Z = 1$)	REL	23 rr	3	ppp					
BLT <i>rel</i>	Branch if Less Than (if $N \oplus V = 1$) (Signed)	REL	91 rr	3	ppp					
BMC <i>rel</i>	Branch if Interrupt Mask Clear (if $I = 0$)	REL	2C rr	3	ppp					
BMI <i>rel</i>	Branch if Minus (if $N = 1$)	REL	2B rr	3	ppp					
BMS <i>rel</i>	Branch if Interrupt Mask Set (if $I = 1$)	REL	2D rr	3	ppp					
BNE <i>rel</i>	Branch if Not Equal (if $Z = 0$)	REL	26 rr	3	ppp	--	--	--	--	--
BPL <i>rel</i>	Branch if Plus (if $N = 0$)	REL	2A rr	3	ppp	--	--	--	--	--

Table 7-2. . Instruction Set Summary (Sheet 3 of 9)

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR				
						VH	I	N	Z	C
BRA <i>rel</i>	Branch Always (if I = 1)	REL	20 rr	3	ppp	--	--	--	--	--
BRCLR <i>n,opr8a,rel</i>	Branch if Bit <i>n</i> in Memory Clear (if (Mn) = 0)	DIR (b0)	01 dd rr	5	rpppp	--	--	--	--	↑
		DIR (b1)	03 dd rr	5	rpppp					
		DIR (b2)	05 dd rr	5	rpppp					
		DIR (b3)	07 dd rr	5	rpppp					
		DIR (b4)	09 dd rr	5	rpppp					
		DIR (b5)	0B dd rr	5	rpppp					
		DIR (b6)	0D dd rr	5	rpppp					
		DIR (b7)	0F dd rr	5	rpppp					
BRN <i>rel</i>	Branch Never (if I = 0)	REL	21 rr	3	ppp	--	--	--	--	--
BRSET <i>n,opr8a,rel</i>	Branch if Bit <i>n</i> in Memory Set (if (Mn) = 1)	DIR (b0)	00 dd rr	5	rpppp	--	--	--	--	↑
		DIR (b1)	02 dd rr	5	rpppp					
		DIR (b2)	04 dd rr	5	rpppp					
		DIR (b3)	06 dd rr	5	rpppp					
		DIR (b4)	08 dd rr	5	rpppp					
		DIR (b5)	0A dd rr	5	rpppp					
		DIR (b6)	0C dd rr	5	rpppp					
		DIR (b7)	0E dd rr	5	rpppp					
BSET <i>n,opr8a</i>	Set Bit <i>n</i> in Memory (Mn ← 1)	DIR (b0)	10 dd	5	rfwpp	--	--	--	--	--
		DIR (b1)	12 dd	5	rfwpp					
		DIR (b2)	14 dd	5	rfwpp					
		DIR (b3)	16 dd	5	rfwpp					
		DIR (b4)	18 dd	5	rfwpp					
		DIR (b5)	1A dd	5	rfwpp					
		DIR (b6)	1C dd	5	rfwpp					
		DIR (b7)	1E dd	5	rfwpp					
BSR <i>rel</i>	Branch to Subroutine PC ← (PC) + \$0002 push (PCL); SP ← (SP) – \$0001 push (PCH); SP ← (SP) – \$0001 PC ← (PC) + <i>rel</i>	REL	AD rr	5	ssppp	--	--	--	--	--
CBEQ <i>opr8a,rel</i>	Compare and... Branch if (A) = (M)	DIR	31 dd rr	5	rpppp	--	--	--	--	--
CBEQA # <i>opr8i,rel</i>	Branch if (A) = (M)	IMM	41 ii rr	4	pppp					
CBEQX # <i>opr8i,rel</i>	Branch if (X) = (M)	IMM	51 ii rr	4	pppp					
CBEQ <i>opr8,X+,rel</i>	Branch if (A) = (M)	IX1+	61 ff rr	5	rpppp					
CBEQ <i>,X+,rel</i>	Branch if (A) = (M)	IX+	71 rr	5	rpppp					
CBEQ <i>opr8,SP,rel</i>	Branch if (A) = (M)	SP1	9E 61 ff rr	6	prpppp					
CLC	Clear Carry Bit (C ← 0)	INH	98	1	p	--	--	--	--	0
CLI	Clear Interrupt Mask Bit (I ← 0)	INH	9A	1	p	--	0	--	--	--
CLR <i>opr8a</i>	Clear M ← \$00	DIR	3F dd	5	rfwpp	0	--	0	1	--
CLRA	A ← \$00	INH	4F	1	p					
CLR X	X ← \$00	INH	5F	1	p					
CLR H	H ← \$00	INH	8C	1	p					
CLR <i>opr8,X</i>	M ← \$00	IX1	6F ff	5	rfwpp					
CLR <i>,X</i>	M ← \$00	IX	7F	4	rwp					
CLR <i>opr8,SP</i>	M ← \$00	SP1	9E 6F ff	6	prfwpp					

8.2 External Signal Description

The ACMP has two analog input pins, ACMP+ and ACMP– and one digital output pin ACMPO. Each of these pins can accept an input voltage that varies across the full operating voltage range of the MCU. As shown in [Figure 8-2](#), the ACMP– pin is connected to the inverting input of the comparator, and the ACMP+ pin is connected to the comparator non-inverting input if ACBGS is a 0. As shown in [Figure 8-2](#), the ACMPO pin can be enabled to drive an external pin.

The signal properties of ACMP are shown in [Table 8-1](#).

Table 8-1. Signal Properties

Signal	Function	I/O
ACMP–	Inverting analog input to the ACMP. (Minus input)	I
ACMP+	Non-inverting analog input to the ACMP. (Positive input)	I
ACMPO	Digital output of the ACMP.	O

8.3 Register Definition

The ACMP includes one register:

- An 8-bit status and control register

Refer to the direct-page register summary in the memory section of this data sheet for the absolute address assignments for all ACMP registers. This section refers to registers and control bits only by their names and relative address offsets.

Some MCUs may have more than one ACMP, so register names include placeholder characters to identify which ACMP is being referenced.

are too fast, then the clock must be divided to the appropriate frequency. This divider is specified by the ADIV bits and can be divide-by 1, 2, 4, or 8.

9.4.2 Input Select and Pin Control

The pin control registers (APCTL3, APCTL2, and APCTL1) are used to disable the I/O port control of the pins used as analog inputs. When a pin control register bit is set, the following conditions are forced for the associated MCU pin:

- The output buffer is forced to its high impedance state.
- The input buffer is disabled. A read of the I/O port returns a zero for any pin with its input buffer disabled.
- The pullup is disabled.

9.4.3 Hardware Trigger

The ADC module has a selectable asynchronous hardware conversion trigger, ADHWT, that is enabled when the ADTRG bit is set. This source is not available on all MCUs. Consult the module introduction for information on the ADHWT source specific to this MCU.

When ADHWT source is available and hardware trigger is enabled (ADTRG=1), a conversion is initiated on the rising edge of ADHWT. If a conversion is in progress when a rising edge occurs, the rising edge is ignored. In continuous convert configuration, only the initial rising edge to launch continuous conversions is observed. The hardware trigger function operates in conjunction with any of the conversion modes and configurations.

9.4.4 Conversion Control

Conversions can be performed in either 10-bit mode or 8-bit mode as determined by the MODE bits. Conversions can be initiated by either a software or hardware trigger. In addition, the ADC module can be configured for low power operation, long sample time, continuous conversion, and automatic compare of the conversion result to a software determined compare value.

9.4.4.1 Initiating Conversions

A conversion is initiated:

- Following a write to ADCSC1 (with ADCH bits not all 1s) if software triggered operation is selected.
- Following a hardware trigger (ADHWT) event if hardware triggered operation is selected.
- Following the transfer of the result to the data registers when continuous conversion is enabled.

If continuous conversions are enabled a new conversion is automatically initiated after the completion of the current conversion. In software triggered operation, continuous conversions begin after ADCSC1 is written and continue until aborted. In hardware triggered operation, continuous conversions begin after a hardware trigger event and continue until aborted.

9.4.5 Automatic Compare Function

The compare function can be configured to check for either an upper limit or lower limit. After the input is sampled and converted, the result is added to the two's complement of the compare value (ADCCVH and ADCCVL). When comparing to an upper limit (ACFGT = 1), if the result is greater-than or equal-to the compare value, COCO is set. When comparing to a lower limit (ACFGT = 0), if the result is less than the compare value, COCO is set. The value generated by the addition of the conversion result and the two's complement of the compare value is transferred to ADCRH and ADCRL.

Upon completion of a conversion while the compare function is enabled, if the compare condition is not true, COCO is not set and no data is transferred to the result registers. An ADC interrupt is generated upon the setting of COCO if the ADC interrupt is enabled (AIEN = 1).

NOTE

The compare function can be used to monitor the voltage on a channel while the MCU is in either wait or stop3 mode. The ADC interrupt will wake the MCU when the compare condition is met.

9.4.6 MCU Wait Mode Operation

The WAIT instruction puts the MCU in a lower power-consumption standby mode from which recovery is very fast because the clock sources remain active. If a conversion is in progress when the MCU enters wait mode, it continues until completion. Conversions can be initiated while the MCU is in wait mode by means of the hardware trigger or if continuous conversions are enabled.

The bus clock, bus clock divided by two, and ADACK are available as conversion clock sources while in wait mode. The use of ALTCLK as the conversion clock source in wait is dependent on the definition of ALTCLK for this MCU. Consult the module introduction for information on ALTCLK specific to this MCU.

A conversion complete event sets the COCO and generates an ADC interrupt to wake the MCU from wait mode if the ADC interrupt is enabled (AIEN = 1).

9.4.7 MCU Stop3 Mode Operation

The STOP instruction is used to put the MCU in a low power-consumption standby mode during which most or all clock sources on the MCU are disabled.

9.4.7.1 Stop3 Mode With ADACK Disabled

If the asynchronous clock, ADACK, is not selected as the conversion clock, executing a STOP instruction aborts the current conversion and places the ADC in its idle state. The contents of ADCRH and ADCRL are unaffected by stop3 mode. After exiting from stop3 mode, a software or hardware trigger is required to resume conversions.

13.1.2 Features

Timer system features include:

- 8-bit up-counter
 - Free-running or 8-bit modulo limit
 - Software controllable interrupt on overflow
 - Counter reset bit (TRST)
 - Counter stop bit (TSTP)
- Four software selectable clock sources for input to prescaler:
 - System bus clock — rising edge
 - Fixed frequency clock (XCLK) — rising edge
 - External clock source on the TCLK pin — rising edge
 - External clock source on the TCLK pin — falling edge
- Nine selectable clock prescale values:
 - Clock source divide by 1, 2, 4, 8, 16, 32, 64, 128, or 256

13.1.3 Modes of Operation

This section defines the MTIM's operation in stop, wait and background debug modes.

13.1.3.1 MTIM in Wait Mode

The MTIM continues to run in wait mode if enabled before executing the WAIT instruction. Therefore, the MTIM can be used to bring the MCU out of wait mode if the timer overflow interrupt is enabled. For lowest possible current consumption, the MTIM should be stopped by software if not needed as an interrupt source during wait mode.

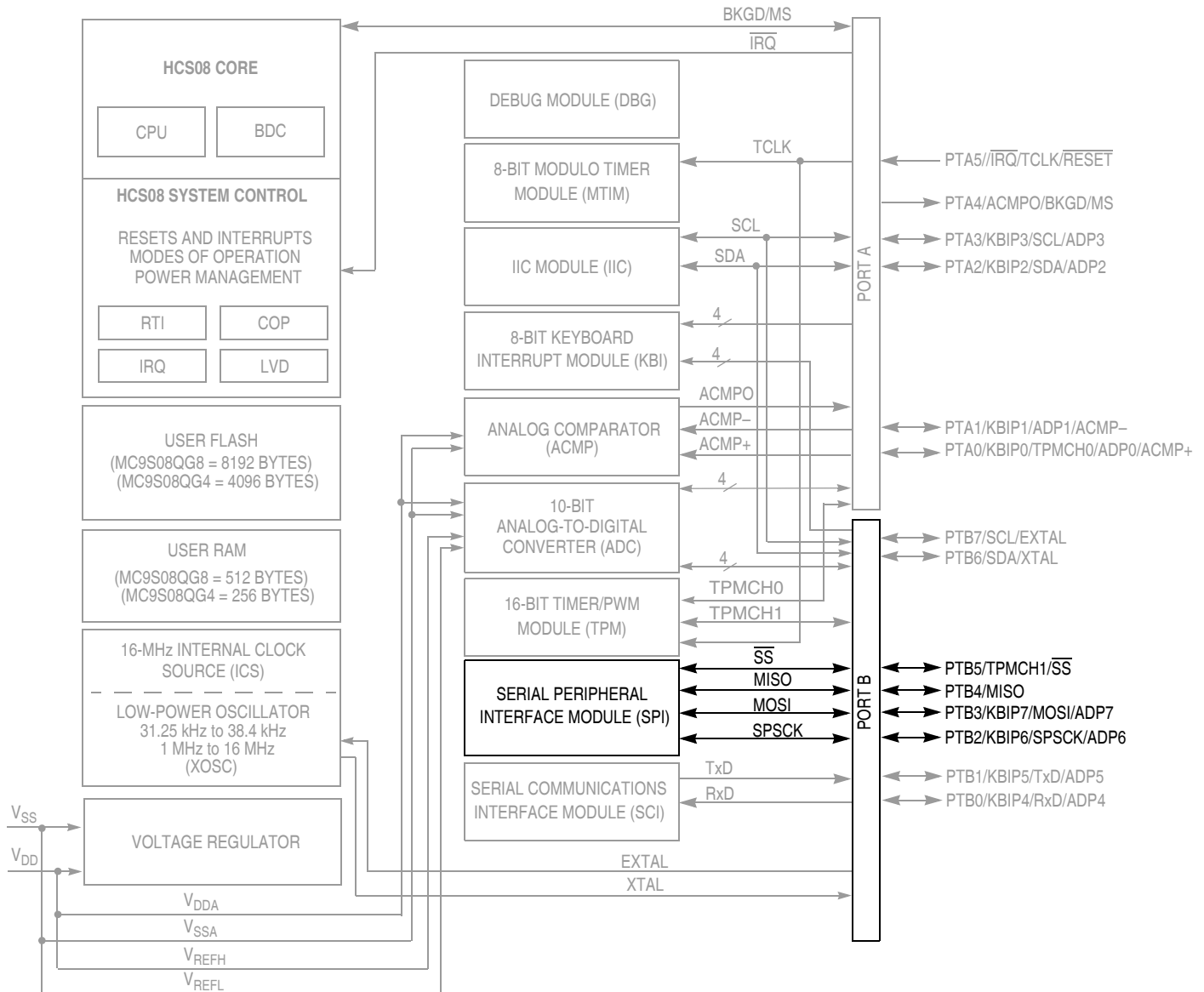
13.1.3.2 MTIM in Stop Modes

The MTIM is disabled in all stop modes, regardless of the settings before executing the STOP instruction. Therefore, the MTIM cannot be used as a wake up source from stop modes.

Waking from stop1 and stop2 modes, the MTIM will be put into its reset state. If stop3 is exited with a reset, the MTIM will be put into its reset state. If stop3 is exited with an interrupt, the MTIM continues from the state it was in when stop3 was entered. If the counter was active upon entering stop3, the count will resume from the current value.

13.1.3.3 MTIM in Active Background Mode

The MTIM suspends all counting until the microcontroller returns to normal user operating mode. Counting resumes from the suspended value as long as an MTIM reset did not occur (TRST written to a 1 or MTIMMOD written).



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 (BKGDPE = 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 15-1. MC9S08QG8/4 Block Diagram Highlighting SPI Block and Pins

Chapter 16

Timer/Pulse-Width Modulator (S08TPMV2)

16.1 Introduction

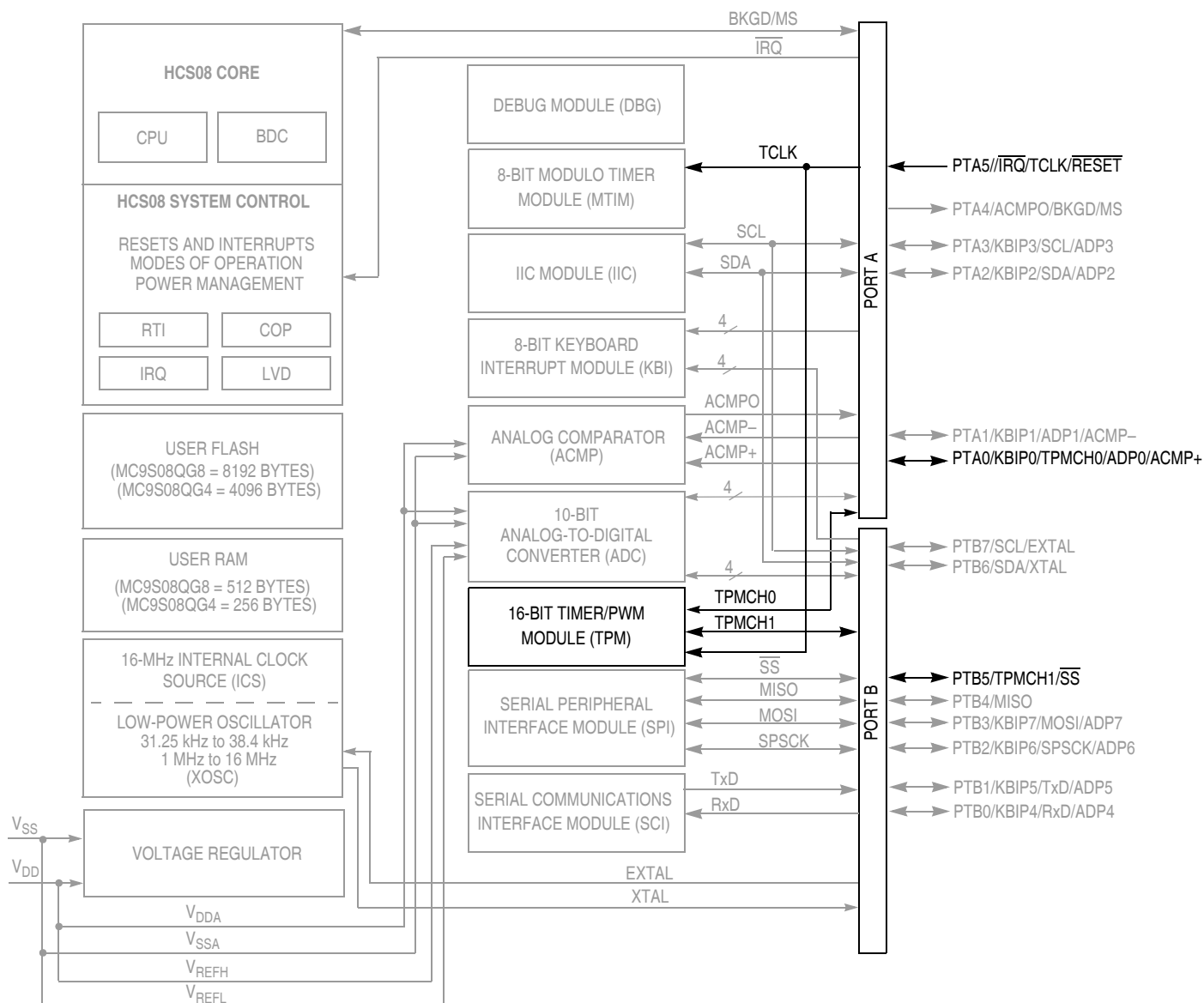
Figure 16-1 shows the MC9S08QG8/4 block diagram with the TPM highlighted.

16.1.1 ACMP/TPM Configuration Information

The ACMP module can be configured to connect the output of the analog comparator to TPM input capture channel 0 by setting ACIC in SOPT2. With ACIC set, the TPMCH0 pin is not available externally regardless of the configuration of the TPM module.

16.1.2 MTIM/TPM Configuration Information

The external clock for the TPM module, TPMCLK, is selected by setting $CLKS[B:A] = 1:1$ in TPMSC, which selects the TCLK pin input. The TCLK input on PTA5 can be enabled as external clock inputs to both the MTIM and TPM modules simultaneously.



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 (BKGDPE = 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 16-1. MC9S08QG8/4 Block Diagram Highlighting TPM Block and Pins

The SYNC command is unlike other BDC commands because the host does not necessarily know the correct communications speed to use for BDC communications until after it has analyzed the response to the SYNC command.

To issue a SYNC command, the host:

- Drives the BKGD pin low for at least 128 cycles of the slowest possible BDC clock (The slowest clock is normally the reference oscillator/64 or the self-clocked rate/64.)
- Drives BKGD high for a brief speedup pulse to get a fast rise time (This speedup pulse is typically one cycle of the fastest clock in the system.)
- Removes all drive to the BKGD pin so it reverts to high impedance
- Monitors the BKGD pin for the sync response pulse

The target, upon detecting the SYNC request from the host (which is a much longer low time than would ever occur during normal BDC communications):

- Waits for BKGD to return to a logic high
- Delays 16 cycles to allow the host to stop driving the high speedup pulse
- Drives BKGD low for 128 BDC clock cycles
- Drives a 1-cycle high speedup pulse to force a fast rise time on BKGD
- Removes all drive to the BKGD pin so it reverts to high impedance

The host measures the low time of this 128-cycle sync response pulse and determines the correct speed for subsequent BDC communications. Typically, the host can determine the correct communication speed within a few percent of the actual target speed and the communication protocol can easily tolerate speed errors of several percent.

17.2.4 BDC Hardware Breakpoint

The BDC includes one relatively simple hardware breakpoint that compares the CPU address bus to a 16-bit match value in the BDCBKPT register. This breakpoint can generate a forced breakpoint or a tagged breakpoint. A forced breakpoint causes the CPU to enter active background mode at the first instruction boundary following any access to the breakpoint address. The tagged breakpoint causes the instruction opcode at the breakpoint address to be tagged so that the CPU will enter active background mode rather than executing that instruction if and when it reaches the end of the instruction queue. This implies that tagged breakpoints can only be placed at the address of an instruction opcode while forced breakpoints can be set at any address.

The breakpoint enable (BKPTEN) control bit in the BDC status and control register (BDCSCR) is used to enable the breakpoint logic (BKPTEN = 1). When BKPTEN = 0, its default value after reset, the breakpoint logic is disabled and no BDC breakpoints are requested regardless of the values in other BDC breakpoint registers and control bits. The force/tag select (FTS) control bit in BDCSCR is used to select forced (FTS = 1) or tagged (FTS = 0) type breakpoints.

The on-chip debug module (DBG) includes circuitry for two additional hardware breakpoints that are more flexible than the simple breakpoint in the BDC module.

17.4.3.7 Debug Control Register (DBGC)

This register can be read or written at any time.

	7	6	5	4	3	2	1	0
R	DBGEN	ARM	TAG	BRKEN	RWA	RWAEN	RWB	RWBEN
W								
Reset	0	0	0	0	0	0	0	0

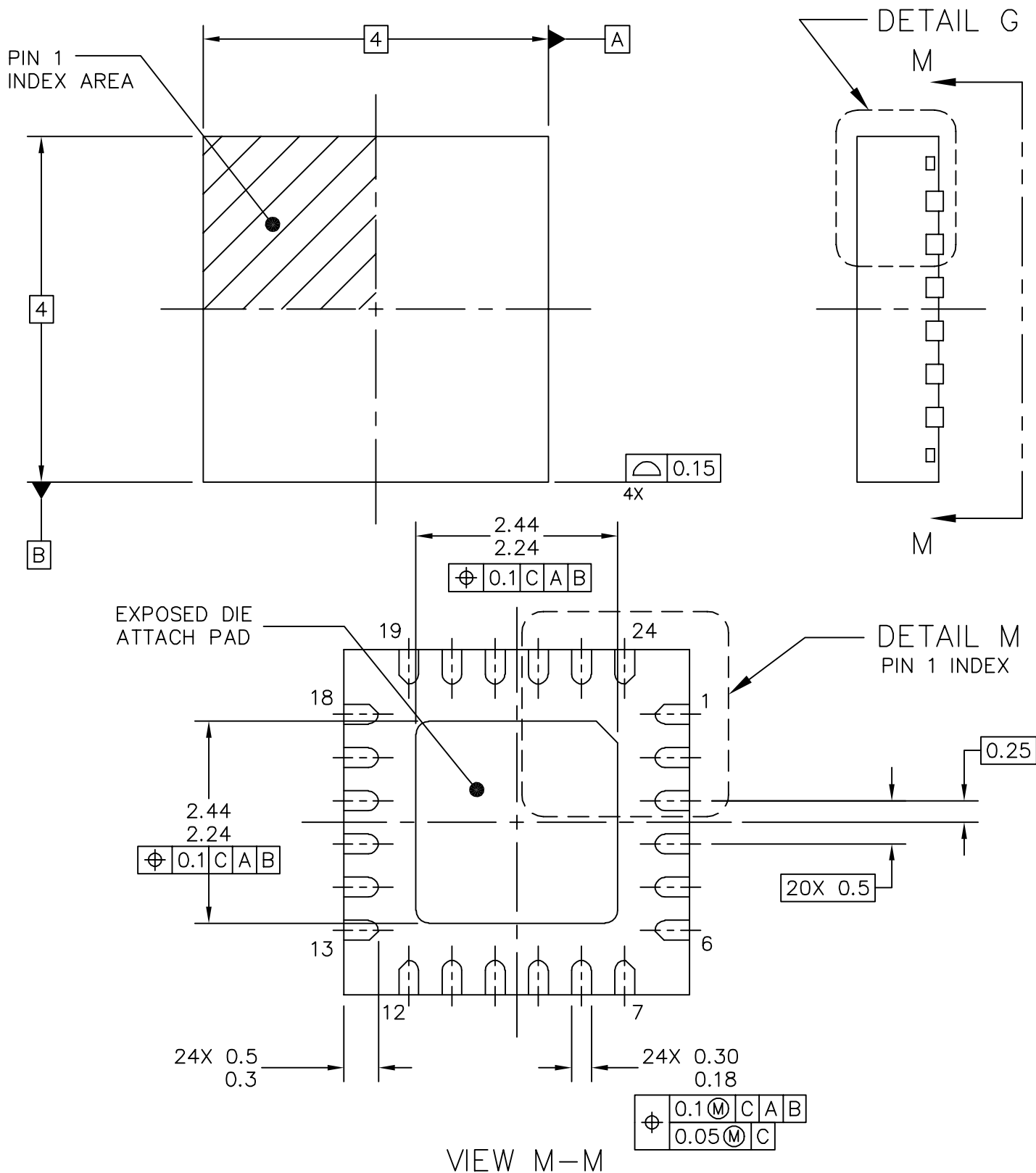
Figure 17-7. Debug Control Register (DBGC)

Table 17-4. DBGC Register Field Descriptions

Field	Description
7 DBGEN	Debug Module Enable — 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	Arm Control — 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	Tag/Force Select — 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	Break Enable — 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	R/W Comparison Value for Comparator A — 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	Enable R/W for Comparator A — 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	R/W Comparison Value for Comparator 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	Enable R/W for Comparator 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

Table B-2. Package Information (continued)

Pin Count	Type	Designator	Document No.
8	DFN	FQ	98ARL10557D
8	PDIP	PA	98ASB42420B
8	NB SOIC	DN	98ASB42564B



© 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			



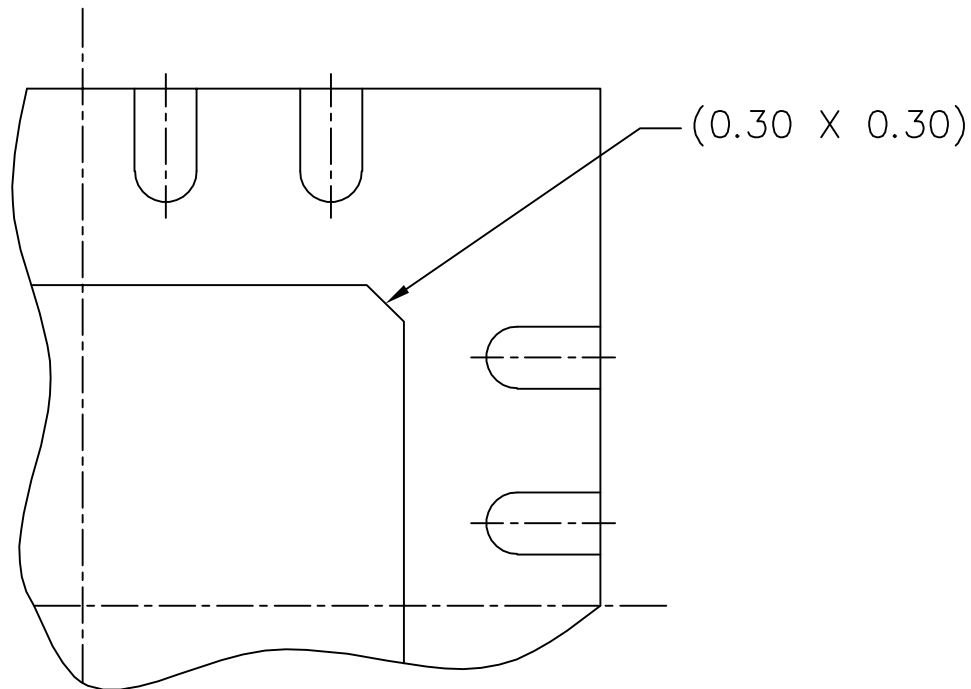
STYLE 1:

- PIN 1. CATHODE
- 2. CATHODE
- 3. CATHODE
- 4. CATHODE
- 5. CATHODE
- 6. CATHODE
- 7. CATHODE
- 8. CATHODE
- 9. ANODE
- 10. ANODE
- 11. ANODE
- 12. ANODE
- 13. ANODE
- 14. ANODE
- 15. ANODE
- 16. ANODE

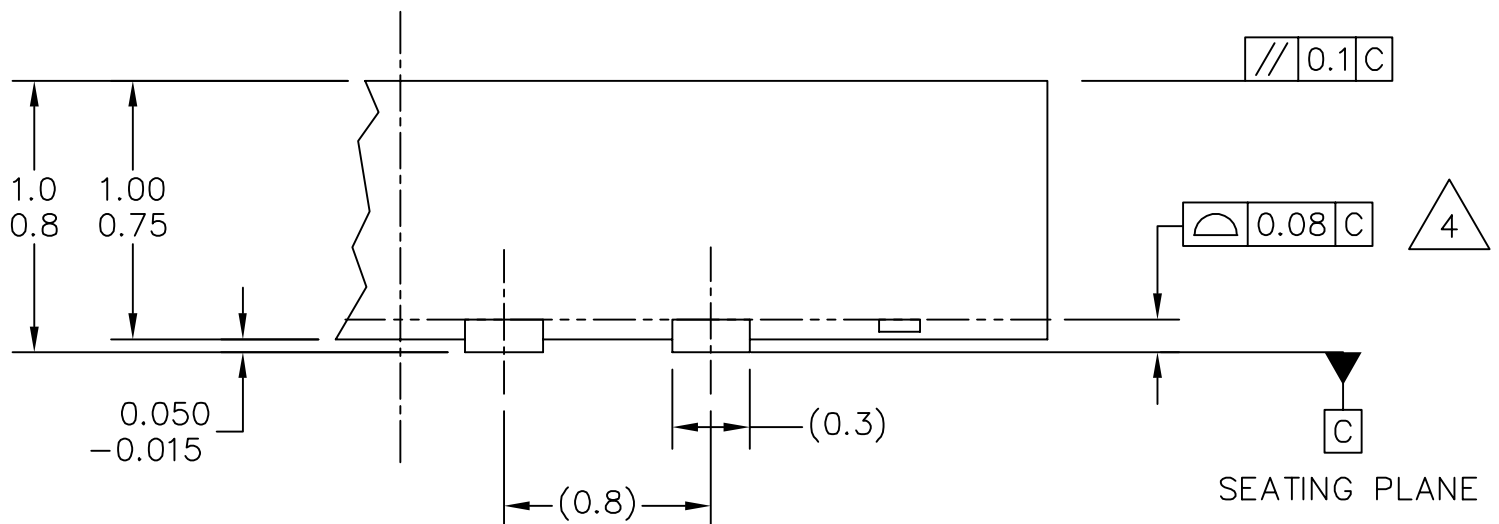
STYLE 2:

- PIN 1. COMMON DRAIN
- 2. COMMON DRAIN
- 3. COMMON DRAIN
- 4. COMMON DRAIN
- 5. COMMON DRAIN
- 6. COMMON DRAIN
- 7. COMMON DRAIN
- 8. COMMON DRAIN
- 9. GATE
- 10. SOURCE
- 11. GATE
- 12. SOURCE
- 13. GATE
- 14. SOURCE
- 15. GATE
- 16. SOURCE

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.		MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE	
TITLE: 16 LD PDIP		DOCUMENT NO: 98ASB42431B		REV: T
		CASE NUMBER: 648-08		19 MAY 2005
		STANDARD: NON-JEDEC		



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) 16 TERMINAL, 0.8 PITCH (5 X 5 X 1) CASE OUTLINE			DOCUMENT NO: 98ARE10614D		REV: 0
			CASE NUMBER: 1679-01		23 MAR 2005
			STANDARD: FREESCALE		