



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	Obsolete
Core Processor	S08
Core Size	8-Bit
Speed	16MHz
Connectivity	-
Peripherals	LVD, POR, PWM, WDT
Number of I/O	4
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	8-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08qd4cpc

6.2	Pin Control — Pullup, Slew Rate and Drive Strength	68
6.3	Pin Behavior in Stop Modes	68
6.4	Parallel I/O Registers	69
6.4.1	Port A Registers	69
6.4.2	Port A Control Registers	70

Chapter 7

Central Processor Unit (S08CPUV2)

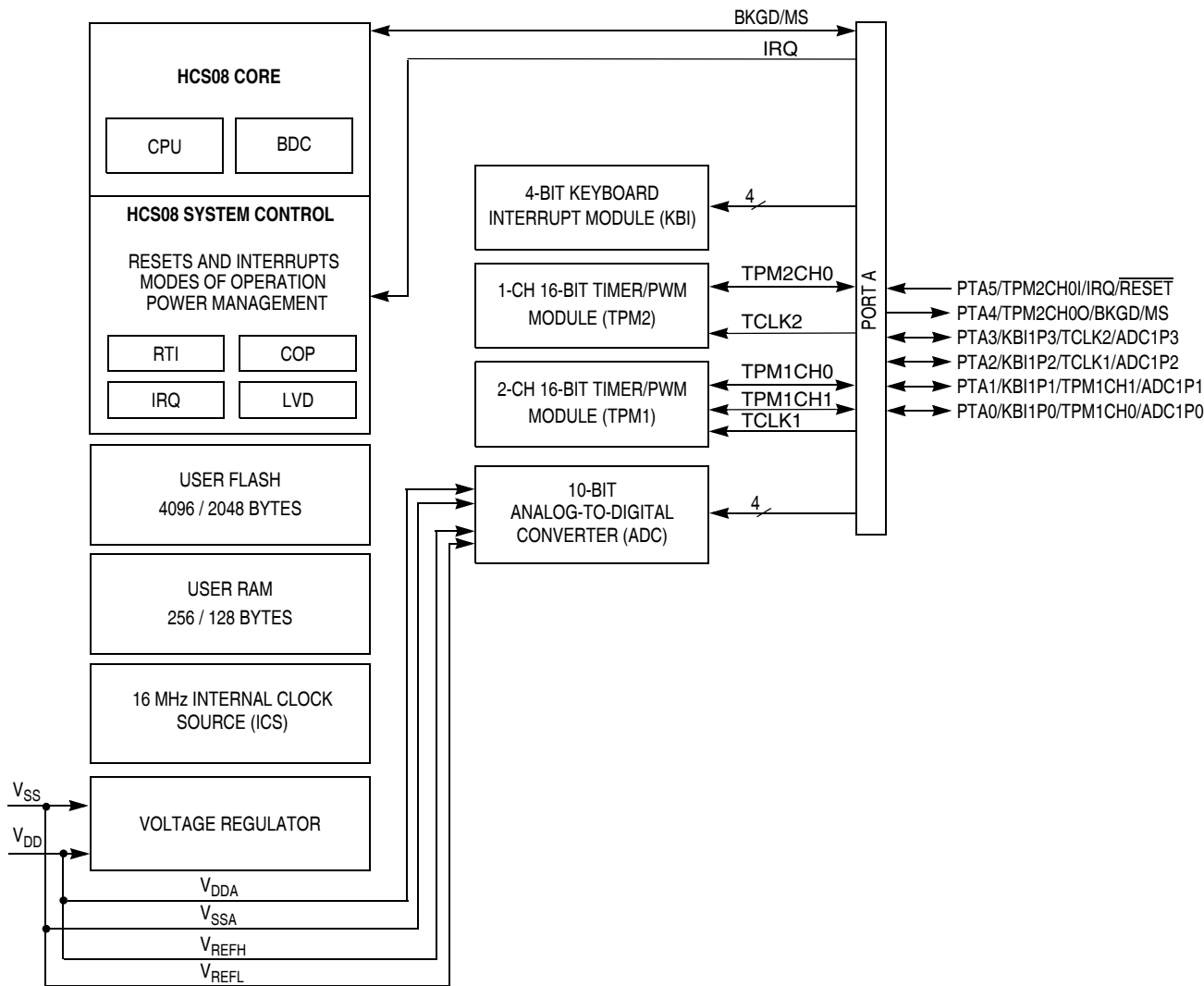
7.1	Introduction	73
7.1.1	Features	73
7.2	Programmer's Model and CPU Registers	74
7.2.1	Accumulator (A)	74
7.2.2	Index Register (H:X)	74
7.2.3	Stack Pointer (SP)	75
7.2.4	Program Counter (PC)	75
7.2.5	Condition Code Register (CCR)	75
7.3	Addressing Modes	77
7.3.1	Inherent Addressing Mode (INH)	77
7.3.2	Relative Addressing Mode (REL)	77
7.3.3	Immediate Addressing Mode (IMM)	77
7.3.4	Direct Addressing Mode (DIR)	77
7.3.5	Extended Addressing Mode (EXT)	78
7.3.6	Indexed Addressing Mode	78
7.4	Special Operations	79
7.4.1	Reset Sequence	79
7.4.2	Interrupt Sequence	79
7.4.3	Wait Mode Operation	80
7.4.4	Stop Mode Operation	80
7.4.5	BGND Instruction	81
7.5	HCS08 Instruction Set Summary	82

Chapter 8

Analog-to-Digital Converter (ADC10V1)

8.1	Introduction	93
8.1.1	Module Configurations	94
8.1.2	Features	97
8.1.3	Block Diagram	97
8.2	External Signal Description	98
8.2.1	Analog Power (V_{DDAD})	99
8.2.2	Analog Ground (V_{SSAD})	99
8.2.3	Voltage Reference High (V_{REFH})	99
8.2.4	Voltage Reference Low (V_{REFL})	99
8.2.5	Analog Channel Inputs (ADx)	99
8.3	Register Definition	99
8.3.1	Status and Control Register 1 (ADCSC1)	99

1.2.1 MCU Block Diagram



- NOTES:
- Port pins are software configurable with pullup device if input port.
 - Port pins are software configurable for output drive strength.
 - Port pins are software configurable for output slew rate control.
 - IRQ contains a software configurable (IRQPDD) pullup/pulldown device if PTA5 enabled as IRQ pin function (IRQPE = 1).
 - RESET contains integrated pullup device if PTA5 enabled as reset pin function (RSTPE = 1).
 - PTA5 does not contain a clamp diode to V_{DD} and must not be driven above V_{DD}. The voltage measured on this pin when internal pullup is enabled may be as low as V_{DD} - 0.7 V. The internal gates connected to this pin are pulled to V_{DD}.
 - PTA4 contains integrated pullup device if BKGD enabled (BKGDPE = 1).
 - 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 1-1. MC9S08QD4 Series Block Diagram

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

Chapter 2

External Signal Description

This chapter describes signals that connect to package pins. It includes pinout diagrams, table of signal properties, and detailed discussions of signals.

2.1 Device Pin Assignment

Figure 2-1 shows the pin assignments for the 8-pin packages.

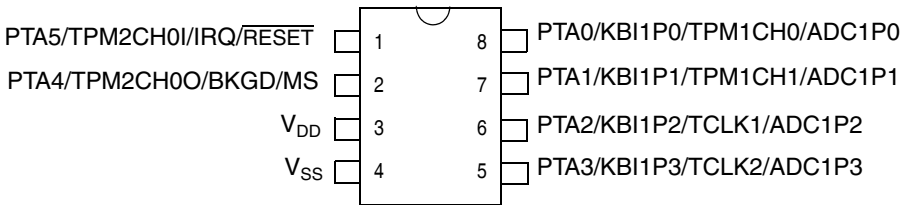


Figure 2-1. 8-Pin Packages

2.2 Recommended System Connections

Figure 2-2 shows pin connections that are common to almost all MC9S08QD4 series application systems.

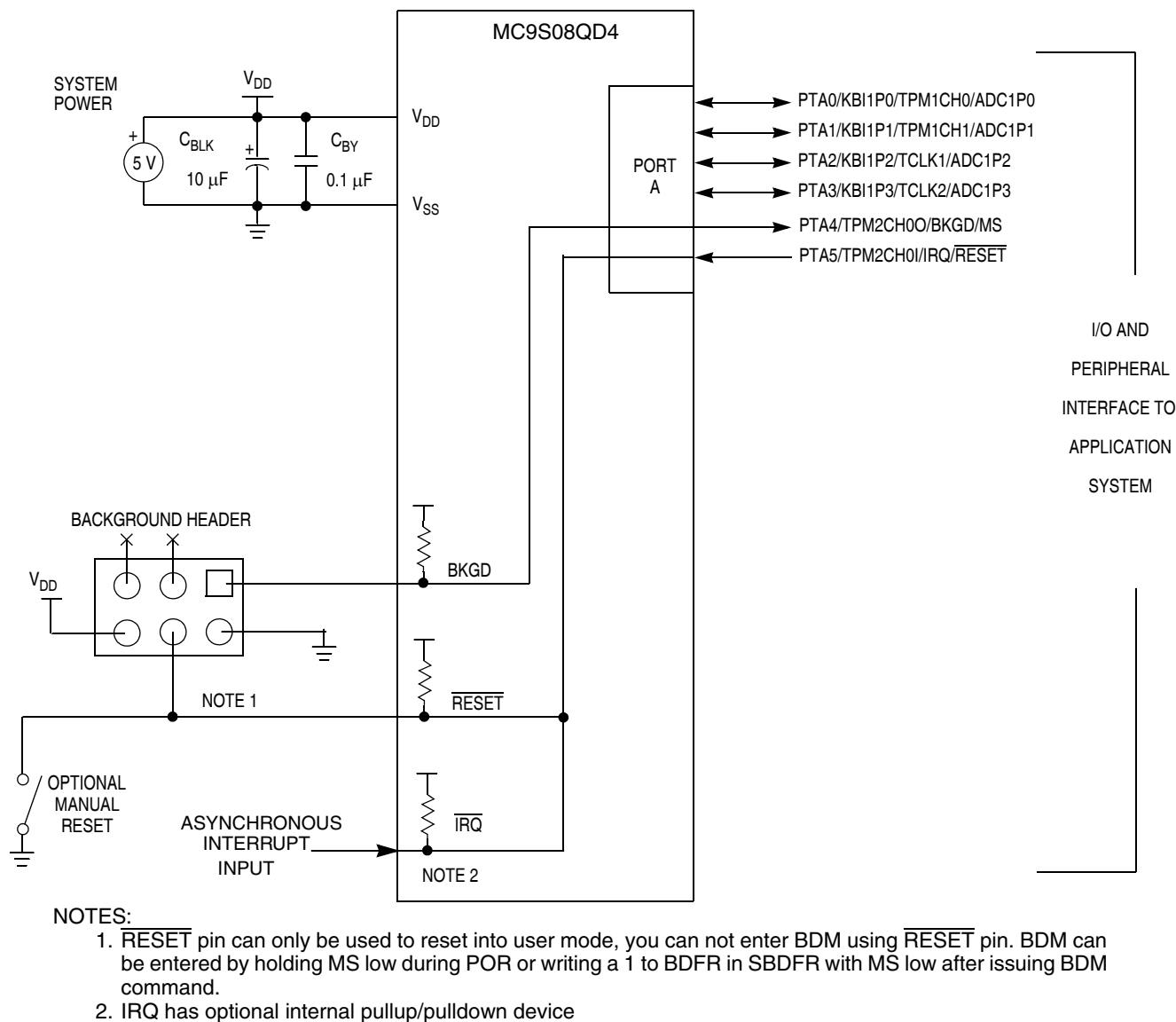


Figure 2-2. Basic System Connections

2.2.1 Power

V_{DD} and V_{SS} are the primary power supply pins for the MCU. This voltage source supplies power to all I/O buffer circuitry, the ADC module, and to an internal voltage regulator. The internal voltage regulator provides regulated lower-voltage source to the CPU and other internal circuitry of the MCU.

Typically, application systems have two separate capacitors across the power pins: a bulk electrolytic capacitor, such as a 10µF tantalum capacitor, to provide bulk charge storage for the overall system, and a bypass capacitor, such as a 0.1µF ceramic capacitor, located as near to the MCU power pins as practical to suppress high-frequency noise.



Table 4-4. Nonvolatile Register Summary

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
0xFFAA – 0xFFAC	Reserved	—	—	—	—	—	—	—	—
0xFFAD	Reserved for ADCRL of AD26 value during ICS trim	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0
0xFFAE	Reserved for ADCRH of AD26 value during ICS trim and ICS Trim value “FTRIM”	ADR9	ADR8	—	—	—	—	—	FTRIM
0xFFAF	Reserved for ICS Trim value “TRIM”	TRIM							
0xFFB0 – 0xFFB7	NVBACKKEY	8-Byte Comparison Key							
0xFFB8 – 0xFFBC	Reserved	—	—	—	—	—	—	—	—
0xFFBD	NVPROT	FPS							
0xFFBE	Unused	—	—	—	—	—	—	—	—
0xFFBF	NVOPT	KEYEN	FNORED	0	0	0	0	SEC01	SEC00

Provided the key enable (KEYEN) bit is 1, the 8-byte comparison key can be used to temporarily disengage memory security. This key mechanism can be accessed only through user code running in secure memory. (A security key cannot be entered directly through background debug commands.) This security key can be disabled completely by programming the KEYEN bit to 0. If the security key is disabled, the only way to disengage security is by mass erasing the flash if needed (normally through the background debug interface) and verifying that flash is blank. To avoid returning to secure mode after the next reset, program the security bits (SEC01:SEC00) to the unsecured state (1:0).

The ICS factory-trimmed value will be stored in 0xFFAE (bit-0) and 0xFFAF. Development tools, such as programmers can trim the ICS and the internal temperature sensor (via the ADC) and store the values in 0xFFAD–0xFFAF.

4.4 RAM

The MC9S08QD4 series includes static RAM. The locations in RAM below 0x0100 can be accessed using the more efficient direct addressing mode, and any single bit in this area can be accessed with the bit manipulation instructions (BCLR, BSET, BRCLR, and BRSET). Locating the most frequently accessed program variables in this area of RAM is preferred.

The RAM retains data when the MCU is in low-power wait, stop2, or stop3 mode. At power-on or after wakeup from stop1, the contents of RAM are uninitialized. RAM data is unaffected by any reset provided that the supply voltage does not drop below the minimum value for RAM retention (V_{RAM}).

- Writing a second time to a flash address before launching the previous command (There is only one write to flash for every command.)
- Writing a second time to FCMD before launching the previous command (There is only one write to FCMD for every command.)
- Writing to any flash control register other than FCMD after writing to a flash address
- Writing any command code other than the five allowed codes (0x05, 0x20, 0x25, 0x40, or 0x41) to FCMD
- Writing any flash control register other than to write to FSTAT (to clear FCBEF and launch the command) after writing the command to FCMD.
- The MCU enters stop mode while a program or erase command is in progress (The command is aborted.)
- Writing the byte program, burst program, or page erase command code (0x20, 0x25, or 0x40) with a background debug command while the MCU is secured (The background debug controller can do blank check and mass erase commands only when the MCU is secure.)
- Writing 0 to FCBEF to cancel a partial command

4.5.6 Flash Block Protection

The block protection feature prevents the protected region of flash from program or erase changes. Block protection is controlled through the flash protection register (FPROT). When enabled, block protection begins at any 512 byte boundary below the last address of flash, 0xFFFF. (see [Section 4.7.4, “Flash Protection Register \(FPROT and NVPROT\).”](#))

After exit from reset, FPROT is loaded with the contents of the NVPROT location which is in the nonvolatile register block of the flash memory. FPROT cannot be changed directly from application software so a runaway program cannot alter the block protection settings. Because NVPROT is within the last 512 bytes of flash, if any amount of memory is protected, NVPROT is itself protected and cannot be altered (intentionally or unintentionally) by the application software. FPROT can be written through background debug commands, which allows a way to erase and reprogram a protected flash memory.

The block protection mechanism is illustrated in [Figure 4-4](#). The FPS bits are used as the upper bits of the last address of unprotected memory. This address is formed by concatenating FPS7:FPS1 with logic 1 bits as shown. For example, in order to protect the last 8192 bytes of memory (addresses 0xE000 through 0xFFFF), the FPS bits must be set to 1101 111, which results in the value 0xDFFF as the last address of unprotected memory. In addition to programming the FPS bits to the appropriate value, FPDIS (bit 0 of NVPROT) must be programmed to logic 0 to enable block protection. Therefore the value 0xDE must be programmed into NVPROT to protect addresses 0xE000 through 0xFFFF.

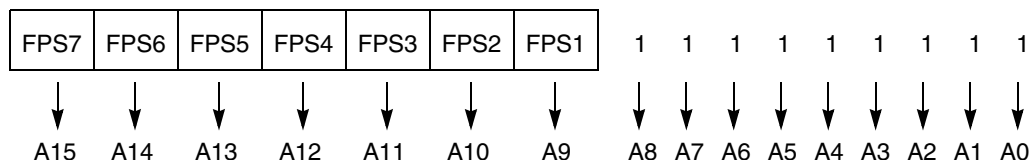


Figure 4-4. Block Protection Mechanism

7.5 HCS08 Instruction Set Summary

Table 7-2 provides a summary of the HCS08 instruction set in all possible addressing modes. The table shows operand construction, execution time in internal bus clock cycles, and cycle-by-cycle details for each addressing mode variation of each instruction.

Table 7-2. Instruction Set Summary (Sheet 1 of 9)

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR				
						VH	I	N	Z	C
ADC #opr8i ADC opr8a ADC opr16a ADC oprx16,X ADC oprx8,X ADC ,X ADC oprx16,SP ADC oprx8,SP	Add with Carry $A \leftarrow (A) + (M) + (C)$	IMM DIR EXT IX2 IX1 IX SP2 SP1	A9 ii B9 dd C9 hh ll D9 ee ff E9 ff F9 9E D9 ee ff 9E E9 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	↑↑	-	↑	↑	↑
ADD #opr8i ADD opr8a ADD opr16a ADD oprx16,X ADD oprx8,X ADD ,X ADD oprx16,SP ADD oprx8,SP	Add without Carry $A \leftarrow (A) + (M)$	IMM DIR EXT IX2 IX1 IX SP2 SP1	AB ii BB dd CB hh ll DB ee ff EB ff FB 9E DB ee ff 9E EB ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	↑↑	-	↑	↑	↑
AIS #opr8i	Add Immediate Value (Signed) to Stack Pointer $SP \leftarrow (SP) + (M)$	IMM	A7 ii	2	pp	--	-	-	-	-
AIX #opr8i	Add Immediate Value (Signed) to Index Register (H:X) $H:X \leftarrow (H:X) + (M)$	IMM	AF ii	2	pp	--	-	-	-	-
AND #opr8i AND opr8a AND opr16a AND oprx16,X AND oprx8,X AND ,X AND oprx16,SP AND oprx8,SP	Logical AND $A \leftarrow (A) \& (M)$	IMM DIR EXT IX2 IX1 IX SP2 SP1	A4 ii B4 dd C4 hh ll D4 ee ff E4 ff F4 9E D4 ee ff 9E E4 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	0-	-	↑	↑	-
ASL opr8a ASLA ASLX ASL oprx8,X ASL ,X ASL oprx8,SP	Arithmetic Shift Left (Same as LSL)	DIR INH INH IX1 IX SP1	38 dd 48 58 68 ff 78 9E 68 ff	5 1 1 5 4 6	rffwpp p p rffwpp rffw prffwpp	↑-	-	↑	↑	↑
ASR opr8a ASRA ASRX ASR oprx8,X ASR ,X ASR oprx8,SP	Arithmetic Shift Right (Same as LSR)	DIR INH INH IX1 IX SP1	37 dd 47 57 67 ff 77 9E 67 ff	5 1 1 5 4 6	rffwpp p p rffwpp rffw prffwpp	↑-	-	↑	↑	↑
BCC rel	Branch if Carry Bit Clear (if C = 0)	REL	24 rr	3	ppp	--	-	-	-	-

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

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR			
						VH	I	N	Z C
RSP	Reset Stack Pointer (Low Byte) $SPL \leftarrow \$FF$ (High Byte Not Affected)	INH	9C	1	p	--	--	--	--
RTI	Return from Interrupt $SP \leftarrow (SP) + \$0001$; Pull (CCR) $SP \leftarrow (SP) + \$0001$; Pull (A) $SP \leftarrow (SP) + \$0001$; Pull (X) $SP \leftarrow (SP) + \$0001$; Pull (PCH) $SP \leftarrow (SP) + \$0001$; Pull (PCL)	INH	80	9	uuuuufppp	↑↑	↑↑↑↑	↑	
RTS	Return from Subroutine $SP \leftarrow SP + \$0001$; Pull (PCH) $SP \leftarrow SP + \$0001$; Pull (PCL)	INH	81	5	ufppp	--	--	--	--
SBC #opr8i SBC opr8a SBC opr16a SBC oprx16,X SBC oprx8,X SBC ,X SBC oprx16,SP SBC oprx8,SP	Subtract with Carry $A \leftarrow (A) - (M) - (C)$	IMM DIR EXT IX2 IX1 IX SP2 SP1	A2 ii B2 dd C2 hh ll D2 ee ff E2 ff F2 9E D2 ee ff 9E E2 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rpp pprpp prpp	↑-	-	↑↑↑↑	
SEC	Set Carry Bit ($C \leftarrow 1$)	INH	99	1	p	--	--	--	1
SEI	Set Interrupt Mask Bit ($I \leftarrow 1$)	INH	9B	1	p	--	1	--	--
STA opr8a STA opr16a STA oprx16,X STA oprx8,X STA ,X STA oprx16,SP STA oprx8,SP	Store Accumulator in Memory $M \leftarrow (A)$	DIR EXT IX2 IX1 IX SP2 SP1	B7 dd C7 hh ll D7 ee ff E7 ff F7 9E D7 ee ff 9E E7 ff	3 4 4 3 2 5 4	wpp pwpp pwpp wpp wp ppwpp pwpp	0-	-	↑↑↑-	
STHX opr8a STHX opr16a STHX oprx8,SP	Store H:X (Index Reg.) $(M:M + \$0001) \leftarrow (H:X)$	DIR EXT SP1	35 dd 96 hh ll 9E FF ff	4 5 5	wwpp pwwpp pwwpp	0-	-	↑↑↑-	
STOP	Enable Interrupts: Stop Processing Refer to MCU Documentation $I \text{ bit} \leftarrow 0$; Stop Processing	INH	8E	2	fp...	--	0	--	--
STX opr8a STX opr16a STX oprx16,X STX oprx8,X STX ,X STX oprx16,SP STX oprx8,SP	Store X (Low 8 Bits of Index Register) in Memory $M \leftarrow (X)$	DIR EXT IX2 IX1 IX SP2 SP1	BF dd CF hh ll DF ee ff EF ff FF 9E DF ee ff 9E EF ff	3 4 4 3 2 5 4	wpp pwpp pwpp wpp wp ppwpp pwpp	0-	-	↑↑↑-	

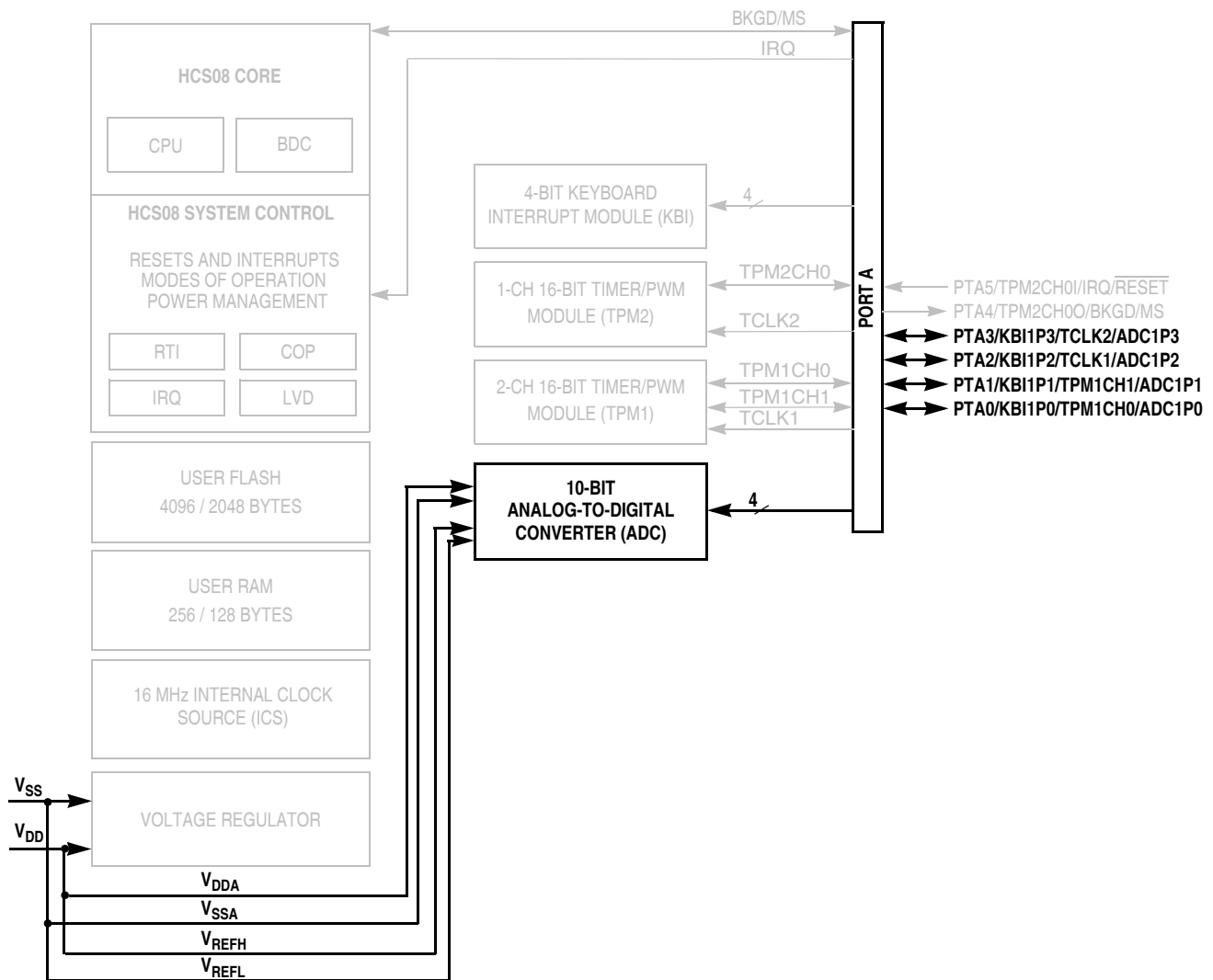
Chapter 8

Analog-to-Digital Converter (ADC10V1)

8.1 Introduction

The 10-bit analog-to-digital converter (ADC) is a successive approximation ADC designed for operation within an integrated microcontroller system-on-chip.

The ADC module design supports up to 28 separate analog inputs (AD0–AD27). Only four (ADC1P0–ADC1P3) of the possible inputs are implemented on the MC9S08QD4 series MCU. These inputs are selected by the ADCH bits.



NOTES:

- ¹ Port pins are software configurable with pullup device if input port.
- ² Port pins are software configurable for output drive strength.
- ³ Port pins are software configurable for output slew rate control.
- ⁴ IRQ contains a software configurable (IRQPDD) pullup/pulldown device if PTA5 enabled as IRQ pin function (IRQPE = 1).
- ⁵ RESET contains integrated pullup device if PTA5 enabled as reset pin function (RSTPE = 1).
- ⁶ PTA5 does not contain a clamp diode to V_{DD} and must not be driven above V_{DD} . The voltage measured on this pin when internal pullup is enabled may be as low as $V_{DD} - 0.7$ V. The internal gates connected to this pin are pulled to V_{DD} .
- ⁷ PTA4 contains integrated pullup device if BKGD enabled (BKGDPE = 1).
- ⁸ 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 8-1. MC9S08QD4 Series Block Diagram Highlighting ADC Block and Pins

8.1.1 Module Configurations

This section provides device-specific information for configuring the ADC on MC9S08QD4 series.

In cases where separate power supplies are used for analog and digital power, the ground connection between these supplies must be at the V_{SSAD} pin. This must be the only ground connection between these supplies if possible. The V_{SSAD} pin makes a good single point ground location.

8.6.1.2 Analog Reference Pins

In addition to the analog supplies, the ADC module has connections for two reference voltage inputs. The high reference is V_{REFH} , which may be shared on the same pin as V_{DDAD} on some devices. The low reference is V_{REFL} , which may be shared on the same pin as V_{SSAD} on some devices.

When available on a separate pin, V_{REFH} may be connected to the same potential as V_{DDAD} , or may be driven by an external source that is between the minimum V_{DDAD} spec and the V_{DDAD} potential (V_{REFH} must never exceed V_{DDAD}). When available on a separate pin, V_{REFL} must be connected to the same voltage potential as V_{SSAD} . Both V_{REFH} and V_{REFL} must be routed carefully for maximum noise immunity and bypass capacitors placed as near as possible to the package.

AC current in the form of current spikes required to supply charge to the capacitor array at each successive approximation step is drawn through the V_{REFH} and V_{REFL} loop. The best external component to meet this current demand is a 0.1 μF capacitor with good high frequency characteristics. This capacitor is connected between V_{REFH} and V_{REFL} and must be placed as near as possible to the package pins. Resistance in the path is not recommended because the current will cause a voltage drop which could result in conversion errors. Inductance in this path must be minimum (parasitic only).

8.6.1.3 Analog Input Pins

The external analog inputs are typically shared with digital I/O pins on MCU devices. The pin I/O control is disabled by setting the appropriate control bit in one of the pin control registers. Conversions can be performed on inputs without the associated pin control register bit set. It is recommended that the pin control register bit always be set when using a pin as an analog input. This avoids problems with contention because the output buffer will be in its high impedance state and the pullup is disabled. Also, the input buffer draws dc current when its input is not at either V_{DD} or V_{SS} . Setting the pin control register bits for all pins used as analog inputs must be done to achieve lowest operating current.

Empirical data shows that capacitors on the analog inputs improve performance in the presence of noise or when the source impedance is high. Use of 0.01 μF capacitors with good high-frequency characteristics is sufficient. These capacitors are not necessary in all cases, but when used they must be placed as near as possible to the package pins and be referenced to V_{SSA} .

For proper conversion, the input voltage must fall between V_{REFH} and V_{REFL} . If the input is equal to or exceeds V_{REFH} , the converter circuit converts the signal to \$3FF (full scale 10-bit representation) or \$FF (full scale 8-bit representation). If the input is equal to or less than V_{REFL} , the converter circuit converts it to \$000. Input voltages between V_{REFH} and V_{REFL} are straight-line linear conversions. There will be a brief current associated with V_{REFL} when the sampling capacitor is charging. The input is sampled for 3.5 cycles of the ADCK source when ADLSMP is low, or 23.5 cycles when ADLSMP is high.

For minimal loss of accuracy due to current injection, pins adjacent to the analog input pins must not be transitioning during conversions.

9.4.7 Fixed Frequency Clock

The ICS presents the divided FLL reference clock as ICSFFCLK for use as an additional clock source for peripheral modules. The ICS provides an output signal (ICSFFE) which indicates when the ICS is providing ICSOUT frequencies four times or greater than the divided FLL reference clock (ICSFFCLK). In FLL engaged mode (FEI and FEE) this is always true and ICSFFE is always high. In ICS bypass modes, ICSFFE will get asserted for the following combinations of BDIV and RDIV values:

- BDIV=00 (divide by 1), RDIV \geq 010
- BDIV=01 (divide by 2), RDIV \geq 011
- BDIV=10 (divide by 4), RDIV \geq 100
- BDIV=11 (divide by 8), RDIV \geq 101

9.5 Module Initialization

This section describes how to initialize and configure the ICS module. The following sections contain two initialization examples.

9.5.1 ICS Module Initialization Sequence

The ICS comes out of POR configured for FEI mode with the BDIV set for divide-by 2. The internal reference will stabilize in t_{IRST} microseconds before the FLL can acquire lock. As soon as the internal reference is stable, the FLL will acquire lock in $t_{Acquire}$ milliseconds.

Upon POR, the internal reference will require trimming to guarantee an accurate clock. Freescale recommends using FLASH location 0xFFAE for storing the fine trim bit, FTRIM in the ICSSC register, and 0xFFAF for storing the 8-bit trim value for the ICSTRM register. The MCU will not automatically copy the values in these FLASH locations to the respective registers. Therefore, user code must copy these values from FLASH to the registers.

NOTE

The BDIV value must not be changed to divide-by 1 without first trimming the internal reference. Failure to do so could result in the MCU running out of specification.

9.5.1.1 Initialization Sequence, Internal Clock Mode to External Clock Mode

To change from FEI or FBI clock modes to FEE or FBE clock modes, follow this procedure:

1. Enable the external clock source by setting the appropriate bits in ICSC2.
 - If FBE will be the selected mode, also set the LP bit at this time to minimize power consumption.
2. If necessary, wait for the external clock source to stabilize. Typical crystal startup times are given in Electrical Characteristics appendix. If EREFS is set in step 1, then the OSCINIT bit will set as soon as the oscillator has completed the initialization cycles.
3. Write to ICSC1 to select the clock mode.

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

11.2 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.

11.2.1 External TPM Clock Sources

When control bits CLKS_B:CLKS_A in the timer status and control register are set to 1:1, the prescaler and consequently the 16-bit counter for TPM_x are driven by an external clock source, TPM_xCLK, 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 ELS_{nB}:ELS_{nA} control bits must be set to 0:0 so the channel is not trying to use the same pin.

11.2.2 TPM_xCH_n — TPM_x 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.

11.3 Register Definition

The TPM includes:

- An 8-bit status and control register (TPM_xSC)
- A 16-bit counter (TPM_xCNTH:TPM_xCNTL)
- A 16-bit modulo register (TPM_xMODH:TPM_xMODL)

Each timer channel has:

- An 8-bit status and control register (TPM_xCnSC)
- A 16-bit channel value register (TPM_xCnVH:TPM_xCnVL)

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

Table 11-2. TPM Clock Source Selection

CLKSB:CLKSA	TPM Clock Source to Prescaler Input
0:0	No clock selected (TPMx disabled)
0:1	Bus rate clock (BUSCLK)
1:0	Fixed system clock (XCLK)
1:1	External source (TPMxCLK) ^{1,2}

¹ The maximum frequency that is allowed as an external clock is one-fourth of the bus frequency.

² If the external clock input is shared with channel n and is selected as the TPM clock source, the corresponding ELSnB:ELSnA control bits must be set to 0:0 so channel n does not try to use the same pin for a conflicting function.

Table 11-3. Prescale Divisor Selection

PS2:PS1:PS0	TPM Clock Source Divided-By
0:0:0	1
0:0:1	2
0:1:0	4
0:1:1	8
1:0:0	16
1:0:1	32
1:1:0	64
1:1:1	128

11.3.2 Timer Counter Registers (TPMxCNTH:TPMxCNTL)

The two read-only TPM counter registers contain the high and low bytes of the value in the TPM counter. Reading either byte (TPMxCNTH or TPMxCNTL) latches the contents of both bytes into a buffer where they remain latched until the other byte is read. This allows coherent 16-bit reads in either order. The coherency mechanism is automatically restarted by an MCU reset, a write of any value to TPMxCNTH or TPMxCNTL, or any write to the timer status/control register (TPMxSC).

Reset clears the TPM counter registers.

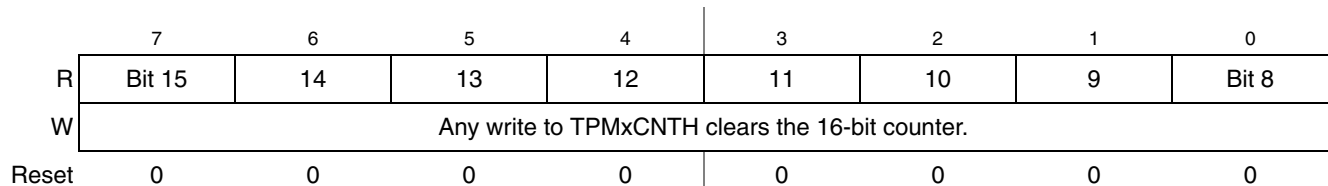


Figure 11-4. Timer Counter Register High (TPMxCNTH)

Table 11-5. Mode, Edge, and Level Selection

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

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

11.3.5 Timer Channel Value Registers (TPMxCnVH:TPMxCnVL)

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

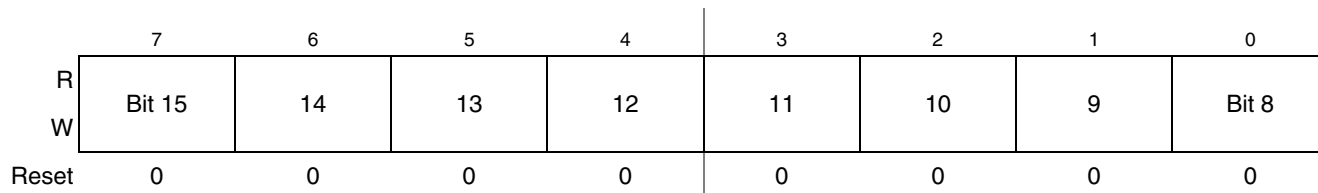


Figure 11-9. Timer Channel Value Register High (TPMxCnVH)

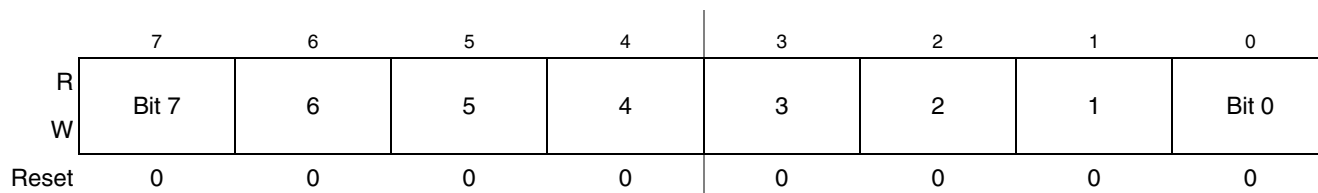


Figure 11-10. Timer Channel Value Register Low (TPMxCnVL)

In input capture mode, reading either byte (TPMxCnVH or TPMxCnVL) latches the contents of both bytes into a buffer where they remain latched until the other byte is read. This latching mechanism also resets (becomes unlatched) when the TPMxCnSC register is written.

11.4.3 Center-Aligned PWM Mode

This type of PWM output uses the up-/down-counting mode of the timer counter (CPWMS = 1). The output compare value in TPMxCnVH:TPMxCnVL determines the pulse width (duty cycle) of the PWM signal and the period is determined by the value in TPMxMODH:TPMxMODL.

TPMxMODH:TPMxMODL must be kept in the range of 0x0001 to 0x7FFF because values outside this range can produce ambiguous results. ELSnA will determine the polarity of the CPWM output.

$$\text{pulse width} = 2 \times (\text{TPMxCnVH:TPMxCnVL}) \quad \text{Eqn. 11-1}$$

$$\begin{aligned} \text{period} &= 2 \times (\text{TPMxMODH:TPMxMODL}); \\ &\text{for TPMxMODH:TPMxMODL} = 0x0001\text{--}0x7FFF \end{aligned} \quad \text{Eqn. 11-2}$$

If the channel value register TPMxCnVH:TPMxCnVL is zero or negative (bit 15 set), the duty cycle will be 0%. If TPMxCnVH:TPMxCnVL is a positive value (bit 15 clear) and is greater than the (nonzero) modulus setting, the duty cycle will be 100% because the duty cycle compare will never occur. This implies the usable range of periods set by the modulus register is 0x0001 through 0x7FFE (0x7FFF if generation of 100% duty cycle is not necessary). This is not a significant limitation because the resulting period is much longer than required for normal applications.

TPMxMODH:TPMxMODL = 0x0000 is a special case that must not be used with center-aligned PWM mode. When CPWMS = 0, this case corresponds to the counter running free from 0x0000 through 0xFFFF, but when CPWMS = 1 the counter needs a valid match to the modulus register somewhere other than at 0x0000 in order to change directions from up-counting to down-counting.

Figure 11-12 shows the output compare value in the TPM channel registers (multiplied by 2), which determines the pulse width (duty cycle) of the CPWM signal. If ELSnA = 0, the compare match while counting up forces the CPWM output signal low and a compare match while counting down forces the output high. The counter counts up until it reaches the modulo setting in TPMxMODH:TPMxMODL, then counts down until it reaches zero. This sets the period equal to two times TPMxMODH:TPMxMODL.

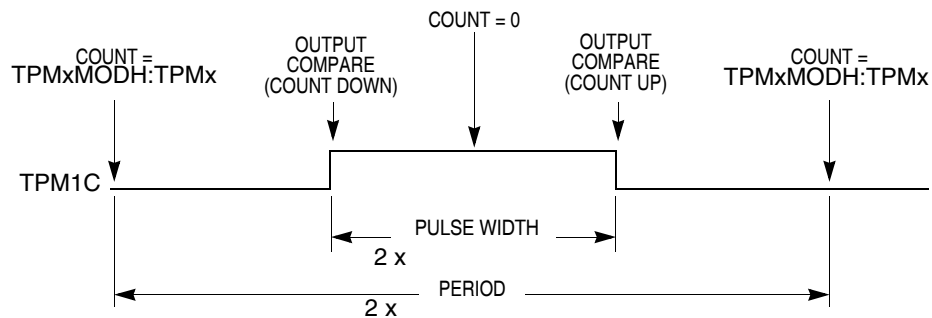


Figure 11-12. CPWM Period and Pulse Width (ELSnA = 0)

Center-aligned PWM outputs typically produce less noise than edge-aligned PWMs because fewer I/O pin transitions are lined up at the same system clock edge. This type of PWM is also required for some types of motor drives.

Because the HCS08 is a family of 8-bit MCUs, the settings in the timer channel registers are buffered to ensure coherent 16-bit updates and to avoid unexpected PWM pulse widths. Writes to any of the registers, TPMxMODH, TPMxMODL, TPMxCnVH, and TPMxCnVL, actually write to buffer registers. Values are

Chapter 12

Development Support

12.1 Introduction

Development support systems in the HCS08 include the background debug controller (BDC). The BDC provides a single-wire debug interface to the target MCU that provides a convenient interface for programming the on-chip flash and other nonvolatile memories. The BDC is also the primary debug interface for development and allows non-intrusive access to memory data and traditional debug features such as CPU register modify, breakpoints, and single instruction trace commands.

In the HCS08 Family, address and data bus signals are not available on external pins (not even in test modes). Debug is done through commands fed into the target MCU via the single-wire background debug interface. The debug module provides a means to selectively trigger and capture bus information so an external development system can reconstruct what happened inside the MCU on a cycle-by-cycle basis without having external access to the address and data signals.

12.1.1 Forcing Active Background

The method for forcing active background mode depends on the specific HCS08 derivative. For the MC9S08QD4 series, you can force active background mode by holding the BKGD pin low as the MCU exits the reset condition independent of what caused the reset. If no debug pod is connected to the BKGD pin, the MCU will always reset into normal operating mode.

12.1.2 Module Configuration

The alternative BDC clock source for MC9S08QD4 series is the ICGCLK. See [Chapter 9, “Internal Clock Source \(S08ICSV1\)”](#), for more information about ICGCLK and how to select clock sources.

Appendix A Electrical Characteristics

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

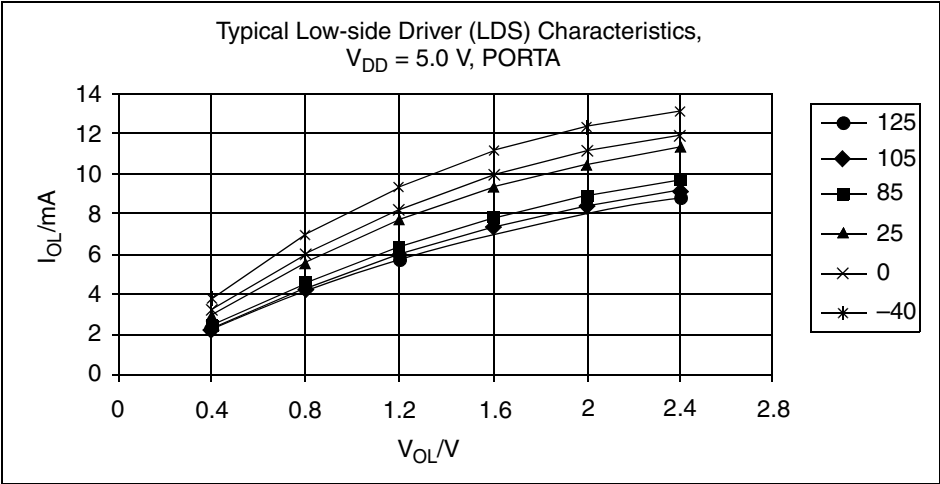


Figure A-1. Typical Low-Side Driver (Sink) Characteristics
Low Drive ($PTxDSn = 0$), $V_{DD} = 5.0\text{V}$, V_{OL} vs. I_{OL}

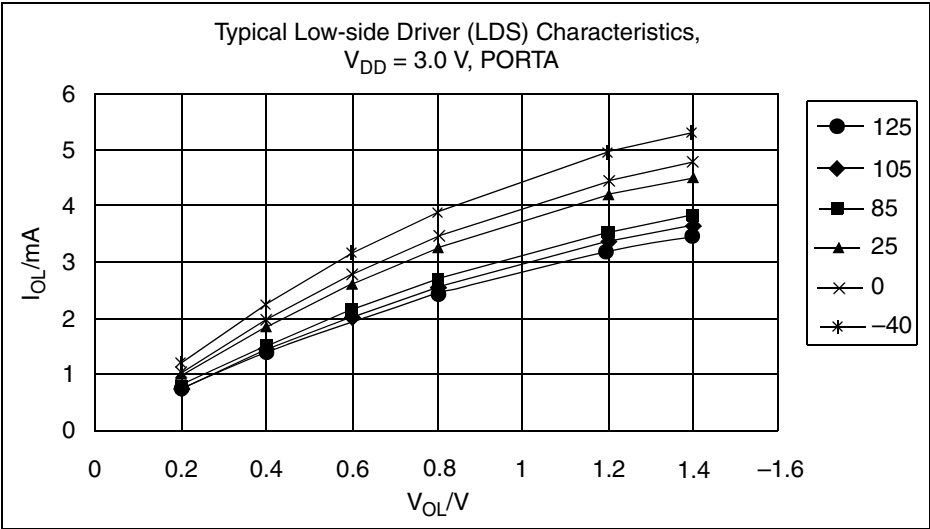


Figure A-2. Typical Low-Side Driver (Sink) Characteristics
Low Drive ($PTxDSn = 0$), $V_{DD} = 3.0 \text{ V}$, V_{OL} vs. I_{OL}

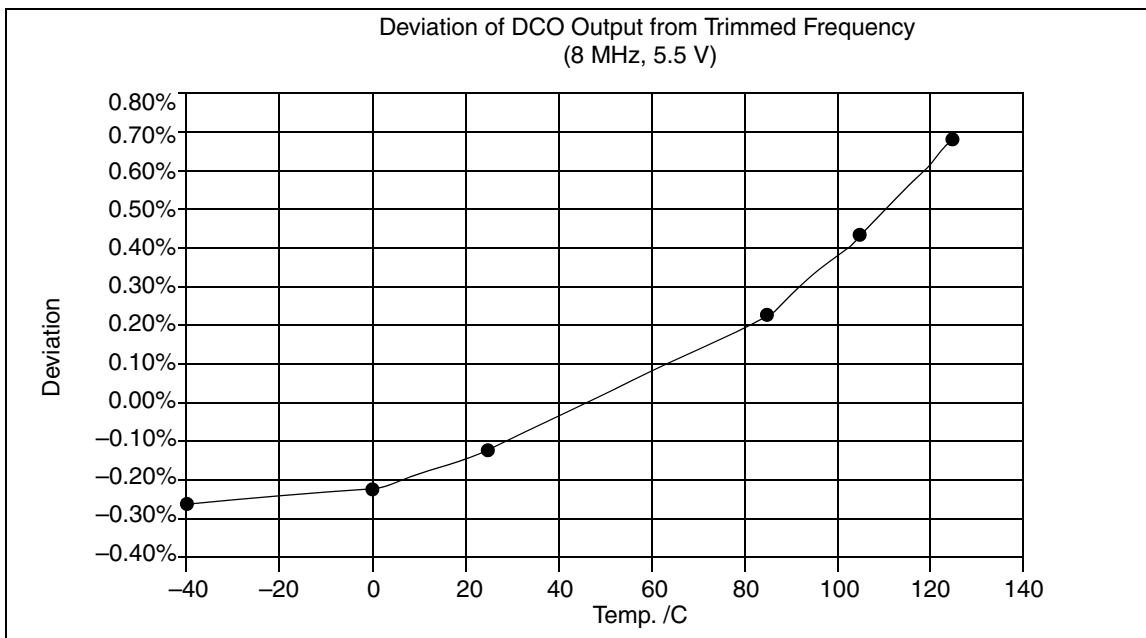


Figure A-10. Typical Deviation of DCO Output vs. Temperature

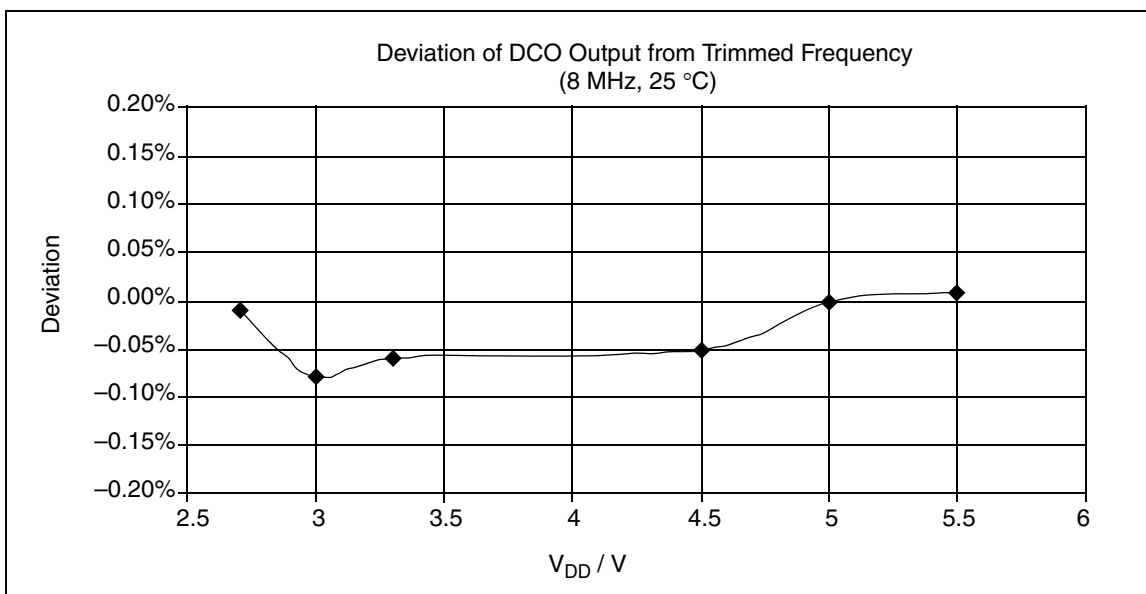


Figure A-11. Typical Deviation of DCO Output vs. Operating Voltage