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	I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	12
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 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/mc9s08qg4cffe

MC9S08QG8 Data Sheet

Covers MC9S08QG8
MC9S08QG4

MC9S08QG8
Rev. 5
11/2009

Section Number **Title** **Page**

Chapter 5
Resets, Interrupts, and General System Control

Appendix A
Electrical Characteristics

Appendix B
Ordering Information and Mechanical Drawings

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)

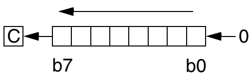
Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR	
						VH	I N Z C

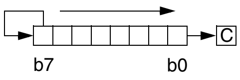
ADC #*opr8i*
ADC *opr8a*
ADC *opr16a*
ADC *oprx16*,X
ADC *oprx8*,X
ADC ,X
ADC *oprx16*,SP
ADC *oprx8*,SP
	Add with Carry
	A ← (A) + (M) + (C)
		IMM
		DIR
		EXT
		IX2
		IX1
		IX
		SP2
		SP1
			`A9 ii`
			`B9 dd`
			`C9 hh ll`
			`D9 ee ff`
			`E9 ff`
			`F9`
			`9E D9 ee ff`
			`9E E9 ff`
				2
				3
				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 ← (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`
						↕ ↕	– ↕ ↕ ↕

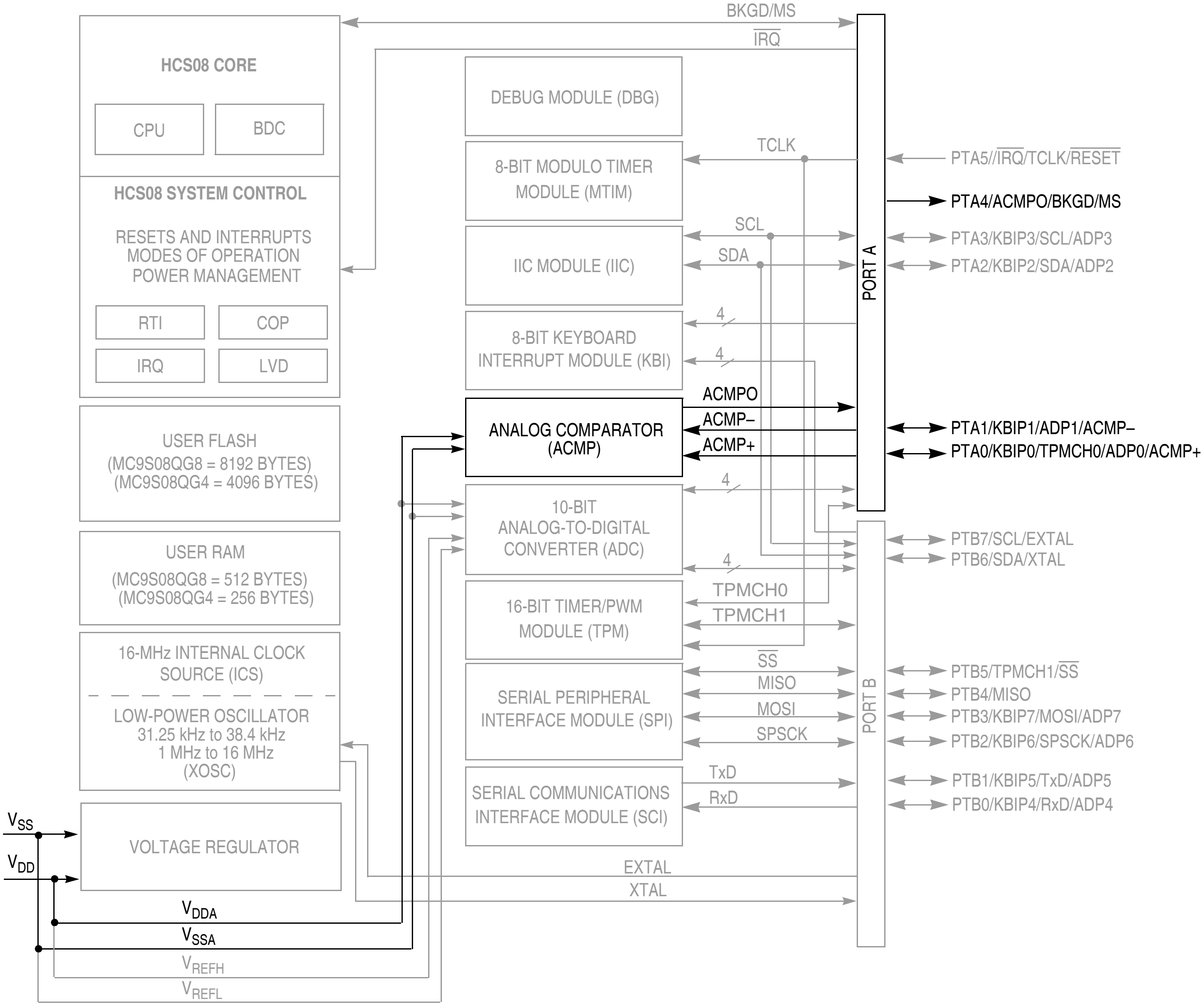
AIS #*opr8i*	Add Immediate Value (Signed) to Stack Pointer SP ← (SP) + (M)	IMM	`A7 ii`	2	`pp`	– –	– – – –
AIX #*opr8i*	Add Immediate Value (Signed) to Index Register (H:X) H:X ← (H:X) + (M)	IMM	`AF ii`	2	`pp`	– –	– – – –

AND #*opr8i*
AND *opr8a*
AND *opr16a*
AND *oprx16*,X
AND *oprx8*,X
AND ,X
AND *oprx16*,SP
AND *oprx8*,SP
	Logical AND
	A ← (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
					`rfwpp`
					`p`
					`p`
					`rfwpp`
					`rfwp`
					`prfwpp`
						↕ –	– ↕ ↕ ↕

ASR *opr8a*
ASRA
ASRX
ASR *oprx8*,X
ASR ,X
ASR *oprx8*,SP
	Arithmetic Shift Right
	
		DIR
		INH
		INH
		IX1
		IX
		SP1
			`37 dd`
			`47`
			`57`
			`67 ff`
			`77`
			`9E 67 ff`
				5
				1
				1
				5
				4
				6
					`rfwpp`
					`p`
					`p`
					`rfwpp`
					`rfwp`
					`prfwpp`
						↕ –	– ↕ ↕ ↕

BCC *rel*	Branch if Carry Bit Clear (if C = 0)	REL	`24 rr`	3	`ppp`	– –	– – – –



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] IRQ contains a software configurable (IRQPDD) pullup device if PTA5 enabled as IRQ pin function (IRQPE = 1).

[6] 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 8-1. MC9S08QG8/4 Block Diagram Highlighting ACMP Block and Pins

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.

8.3.1 ACMP Status and Control Register (ACMPSC)

ACMPSC contains the status flag and control bits which are used to enable and configure the ACMP.

	7	6	5	4	3	2	1	0
R	ACME	ACBGS	ACF	ACIE	ACO	ACOPE	ACMOD	
W					▨			
Reset:	0	0	0	0	0	0	0	0

▨ = Unimplemented

Figure 8-3. ACMP Status and Control Register

Table 8-2. ACMP Status and Control Register Field Descriptions

Field	Description
7 ACME	**Analog Comparator Module Enable** — ACME enables the ACMP module. 0 ACMP not enabled 1 ACMP is enabled
6 ACBGS	**Analog Comparator Bandgap Select** — ACBGS is used to select between the bandgap reference voltage or the ACMP+ pin as the input to the non-inverting input of the analog comparatorr. 0 External pin ACMP+ selected as non-inverting input to comparator 1 Internal reference select as non-inverting input to comparator
5 ACF	**Analog Comparator Flag** — ACF is set when a compare event occurs. Compare events are defined by ACMOD. ACF is cleared by writing a one to ACF. 0 Compare event has not occurred 1 Compare event has occurred
4 ACIE	**Analog Comparator Interrupt Enable** — ACIE enables the interrupt from the ACMP. When ACIE is set, an interrupt will be asserted when ACF is set. 0 Interrupt disabled 1 Interrupt enabled
3 ACO	**Analog Comparator Output** — Reading ACO will return the current value of the analog comparator output. ACO is reset to a 0 and will read as a 0 when the ACMP is disabled (ACME = 0).
2 ACOPE	**Analog Comparator Output Pin Enable** — ACOPE is used to enable the comparator output to be placed onto the external pin, ACMPO. 0 Analog comparator output not available on ACMPO 1 Analog comparator output is driven out on ACMPO
1:0 ACMOD	**Analog Comparator Mode** — ACMOD selects the type of compare event which sets ACF. 00 Encoding 0 — Comparator output falling edge 01 Encoding 1 — Comparator output rising edge 10 Encoding 2 — Comparator output falling edge 11 Encoding 3 — Comparator output rising or falling edge

11.1.4 Block Diagram

Figure 11-2 is a block diagram of the IIC.

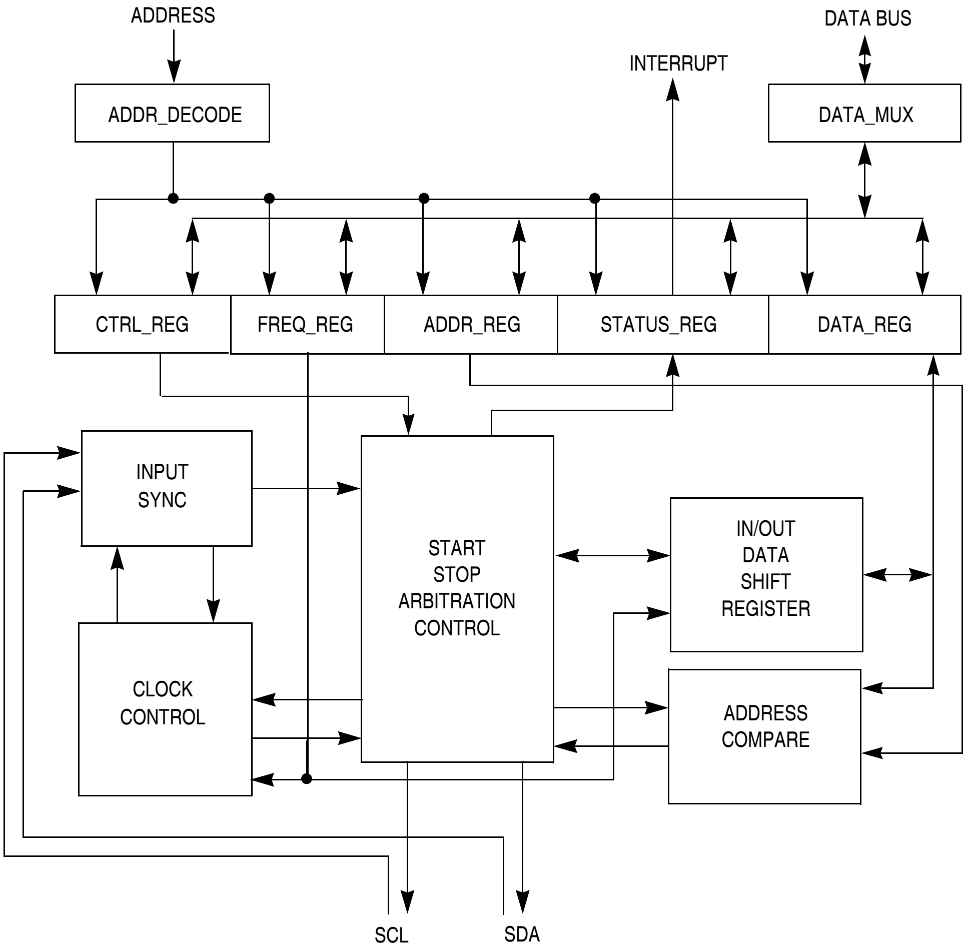


Figure 11-2. IIC Functional Block Diagram

11.2 External Signal Description

This section describes each user-accessible pin signal.

11.2.1 SCL — Serial Clock Line

The bidirectional SCL is the serial clock line of the IIC system.

11.2.2 SDA — Serial Data Line

The bidirectional SDA is the serial data line of the IIC system.

11.3 Register Definition

This section consists of the IIC register descriptions in address order.

13.4.1 MTIM Operation Example

This section shows an example of the MTIM operation as the counter reaches a matching value from the modulo register.

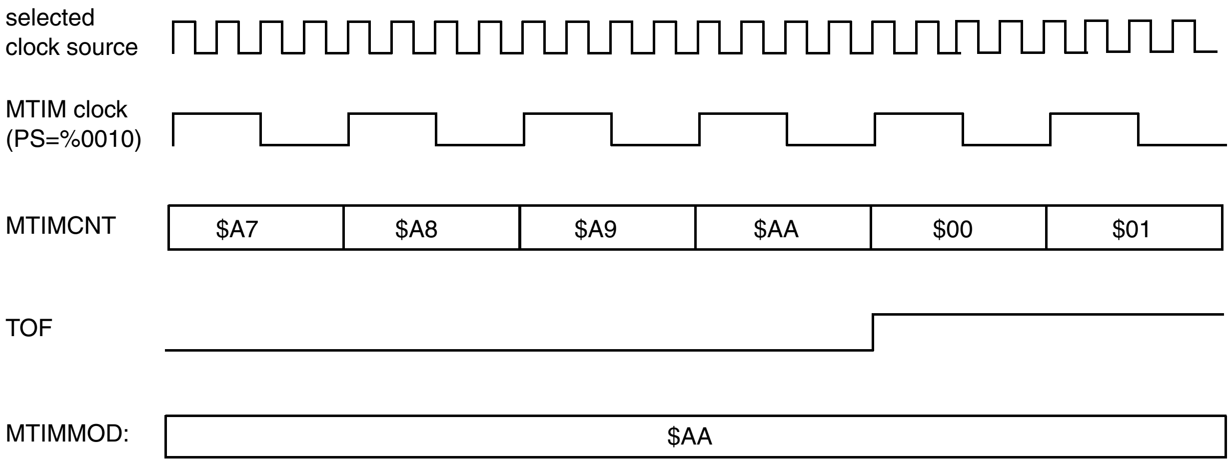


Figure 13-8. MTIM counter overflow example

In the example of Figure 13-8, the selected clock source could be any of the five possible choices. The prescaler is set to PS = %0010 or divide-by-4. The modulo value in the MTIMMOD register is set to $AA. When the counter, MTIMCNT, reaches the modulo value of $AA, the counter overflows to $00 and continues counting. The timer overflow flag, TOF, sets when the counter value changes from $AA to $00. An MTIM overflow interrupt is generated when TOF is set, if TOIE = 1.

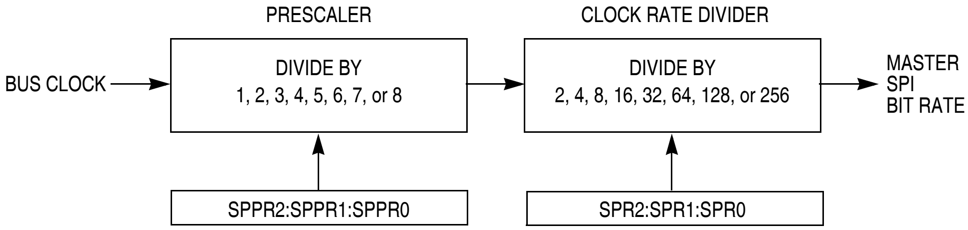


Figure 15-4. SPI Baud Rate Generation

15.2 External Signal Description

The SPI optionally shares four port pins. The function of these pins depends on the settings of SPI control bits. When the SPI is disabled (SPE = 0), these four pins revert to being general-purpose port I/O pins that are not controlled by the SPI.

15.2.1 SPSCK — SPI Serial Clock

When the SPI is enabled as a slave, this pin is the serial clock input. When the SPI is enabled as a master, this pin is the serial clock output.

15.2.2 MOSI — Master Data Out, Slave Data In

When the SPI is enabled as a master and SPI pin control zero (SPC0) is 0 (not bidirectional mode), this pin is the serial data output. When the SPI is enabled as a slave and SPC0 = 0, this pin is the serial data input. If SPC0 = 1 to select single-wire bidirectional mode, and master mode is selected, this pin becomes the bidirectional data I/O pin (MOMI). Also, the bidirectional mode output enable bit determines whether the pin acts as an input (BIDIROE = 0) or an output (BIDIROE = 1). If SPC0 = 1 and slave mode is selected, this pin is not used by the SPI and reverts to being a general-purpose port I/O pin.

15.2.3 MISO — Master Data In, Slave Data Out

When the SPI is enabled as a master and SPI pin control zero (SPC0) is 0 (not bidirectional mode), this pin is the serial data input. When the SPI is enabled as a slave and SPC0 = 0, this pin is the serial data output. If SPC0 = 1 to select single-wire bidirectional mode, and slave mode is selected, this pin becomes the bidirectional data I/O pin (SISO) and the bidirectional mode output enable bit determines whether the pin acts as an input (BIDIROE = 0) or an output (BIDIROE = 1). If SPC0 = 1 and master mode is selected, this pin is not used by the SPI and reverts to being a general-purpose port I/O pin.

15.2.4 $\overline{\text{SS}}$ — Slave Select

When the SPI is enabled as a slave, this pin is the low-true slave select input. When the SPI is enabled as a master and mode fault enable is off (MODFEN = 0), this pin is not used by the SPI and reverts to being a general-purpose port I/O pin. When the SPI is enabled as a master and MODFEN = 1, the slave select output enable bit determines whether this pin acts as the mode fault input (SSOE = 0) or as the slave select output (SSOE = 1).

MOSI output pin from a master and the MISO waveform applies to the MISO output from a slave. The $\overline{SS}$ OUT waveform applies to the slave select output from a master (provided MODFEN and SSOE = 1). The master $\overline{SS}$ output goes to active low one-half SPSCK cycle before the start of the transfer and goes back high at the end of the eighth bit time of the transfer. The $\overline{SS}$ IN waveform applies to the slave select input of a slave.

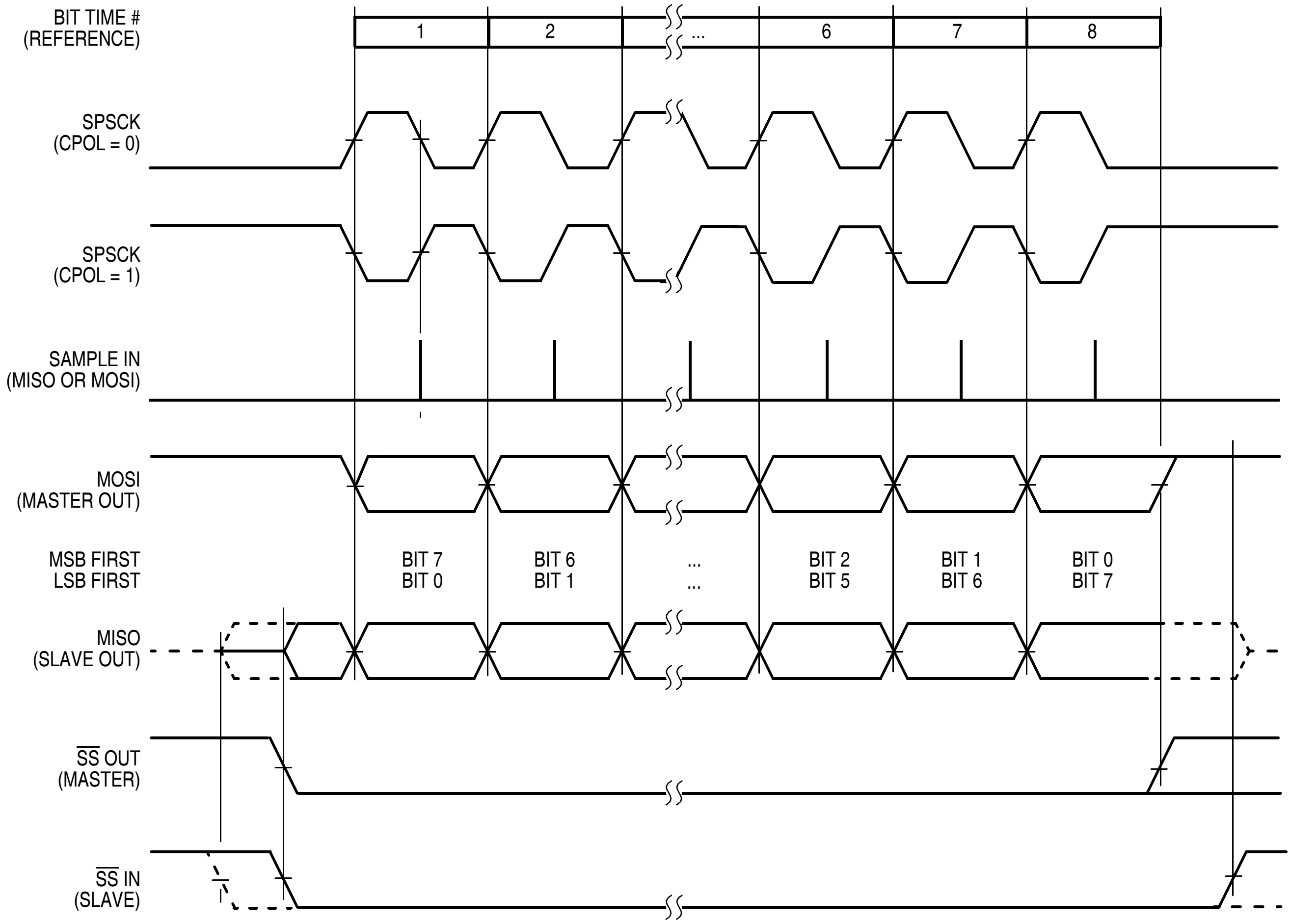


Figure 15-10. SPI Clock Formats (CPHA = 1)

When CPHA = 1, the slave begins to drive its MISO output when $\overline{SS}$ goes to active low, but the data is not defined until the first SPSCK edge. The first SPSCK edge shifts the first bit of data from the shifter onto the MOSI output of the master and the MISO output of the slave. The next SPSCK edge causes both the master and the slave to sample the data bit values on their MISO and MOSI inputs, respectively. At the third SPSCK edge, the SPI shifter shifts one bit position which shifts in the bit value that was just sampled, and shifts the second data bit value out the other end of the shifter to the MOSI and MISO outputs of the master and slave, respectively. When CHPA = 1, the slave's $\overline{SS}$ input is not required to go to its inactive high level between transfers.

Figure 15-11 shows the clock formats when CPHA = 0. At the top of the figure, the eight bit times are shown for reference with bit 1 starting as the slave is selected ($\overline{SS}$ IN goes low), and bit 8 ends at the last SPSCK edge. The MSB first and LSB first lines show the order of SPI data bits depending on the setting

Figure 17-4 shows the host receiving a logic 0 from the target HCS08 MCU. Because the host is asynchronous to the target MCU, there is a 0-to-1 cycle delay from the host-generated falling edge on BKGD to the start of the bit time as perceived by the target MCU. The host initiates the bit time but the target HCS08 finishes it. Because the target wants the host to receive a logic 0, it drives the BKGD pin low for 13 BDC clock cycles, then briefly drives it high to speed up the rising edge. The host samples the bit level about 10 cycles after starting the bit time.

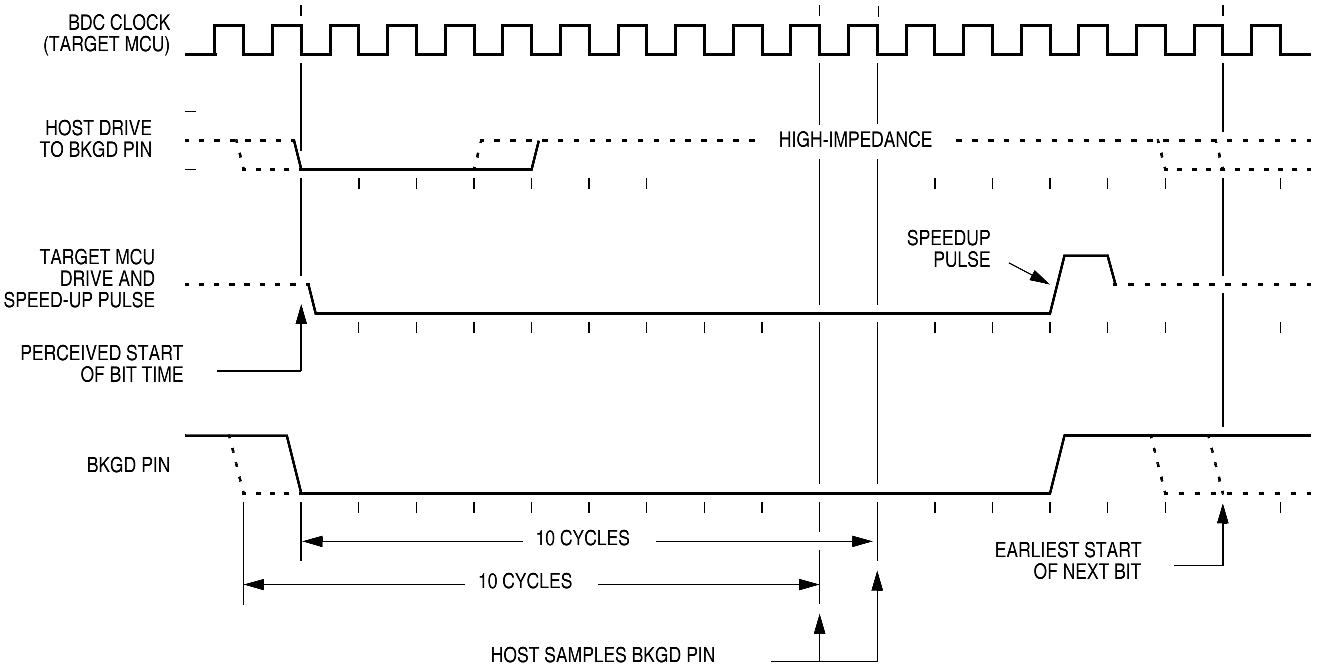


Figure 17-4. BDM Target-to-Host Serial Bit Timing (Logic 0)

17.2.3 BDC Commands

BDC commands are sent serially from a host computer to the BKGD pin of the target HCS08 MCU. All commands and data are sent MSB-first using a custom BDC communications protocol. Active background mode commands require that the target MCU is currently in the active background mode while non-intrusive commands may be issued at any time whether the target MCU is in active background mode or running a user application program.

Table 17-1 shows all HCS08 BDC commands, a shorthand description of their coding structure, and the meaning of each command.

Coding Structure Nomenclature

This nomenclature is used in Table 17-1 to describe the coding structure of the BDC commands.

Table 17-1. BDC Command Summary

Command Mnemonic	Active BDM/ Non-intrusive	Coding Structure	Description
SYNC	Non-intrusive	n/a[1]	Request a timed reference pulse to determine target BDC communication speed
ACK_ENABLE	Non-intrusive	D5/d	Enable acknowledge protocol. Refer to Freescale document order no. HCS08RMv1/D.
ACK_DISABLE	Non-intrusive	D6/d	Disable acknowledge protocol. Refer to Freescale document order no. HCS08RMv1/D.
BACKGROUND	Non-intrusive	90/d	Enter active background mode if enabled (ignore if ENBDM bit equals 0)
READ_STATUS	Non-intrusive	E4/SS	Read BDC status from BDCSCR
WRITE_CONTROL	Non-intrusive	C4/CC	Write BDC controls in BDCSCR
READ_BYTE	Non-intrusive	E0/AAAA/d/RD	Read a byte from target memory
READ_BYTE_WS	Non-intrusive	E1/AAAA/d/SS/RD	Read a byte and report status
READ_LAST	Non-intrusive	E8/SS/RD	Re-read byte from address just read and report status
WRITE_BYTE	Non-intrusive	C0/AAAA/WD/d	Write a byte to target memory
WRITE_BYTE_WS	Non-intrusive	C1/AAAA/WD/d/SS	Write a byte and report status
READ_BKPT	Non-intrusive	E2/RBKP	Read BDCBKPT breakpoint register
WRITE_BKPT	Non-intrusive	C2/WBKP	Write BDCBKPT breakpoint register
GO	Active BDM	08/d	Go to execute the user application program starting at the address currently in the PC
TRACE1	Active BDM	10/d	Trace 1 user instruction at the address in the PC, then return to active background mode
TAGGO	Active BDM	18/d	Same as GO but enable external tagging (HCS08 devices have no external tagging pin)
READ_A	Active BDM	68/d/RD	Read accumulator (A)
READ_CCR	Active BDM	69/d/RD	Read condition code register (CCR)
READ_PC	Active BDM	6B/d/RD16	Read program counter (PC)
READ_HX	Active BDM	6C/d/RD16	Read H and X register pair (H:X)
READ_SP	Active BDM	6F/d/RD16	Read stack pointer (SP)
READ_NEXT	Active BDM	70/d/RD	Increment H:X by one then read memory byte located at H:X
READ_NEXT_WS	Active BDM	71/d/SS/RD	Increment H:X by one then read memory byte located at H:X. Report status and data.
WRITE_A	Active BDM	48/WD/d	Write accumulator (A)
WRITE_CCR	Active BDM	49/WD/d	Write condition code register (CCR)
WRITE_PC	Active BDM	4B/WD16/d	Write program counter (PC)
WRITE_HX	Active BDM	4C/WD16/d	Write H and X register pair (H:X)
WRITE_SP	Active BDM	4F/WD16/d	Write stack pointer (SP)
WRITE_NEXT	Active BDM	50/WD/d	Increment H:X by one, then write memory byte located at H:X
WRITE_NEXT_WS	Active BDM	51/WD/d/SS	Increment H:X by one, then write memory byte located at H:X. Also report status.

[1] The SYNC command is a special operation that does not have a command code.

Table A-7. Supply Current Characteristics

Parameter	Symbol	V_{DD} (V)[1]	Typical[2]	Max	T (°C)
RTI adder to stop1, stop2 or stop3 [4]	—	3	300 nA	—	85
		2	300 nA	—	85
LVD adder to stop3 (LVDE = LVDSE = 1)	—	3	70 µA	—	85
		2	60 µA	—	85
Adder to stop3 for oscillator enabled [5] (EREFSTEN =1)	—	3	5 µA	—	85
		2	4 µA	—	85

[1] 3-V values are 100% tested; 2-V values are characterized but not tested.

[2] Typicals are measured at 25°C.

[3] Does not include any DC loads on port pins.

[4] Most customers are expected to find that auto-wakeup from a stop mode can be used instead of the higher current wait mode.

[5] Values given under the following conditions: low range operation (RANGE = 0), Loss-of-clock disabled (LOCD = 1), low-power oscillator (HGO = 0).

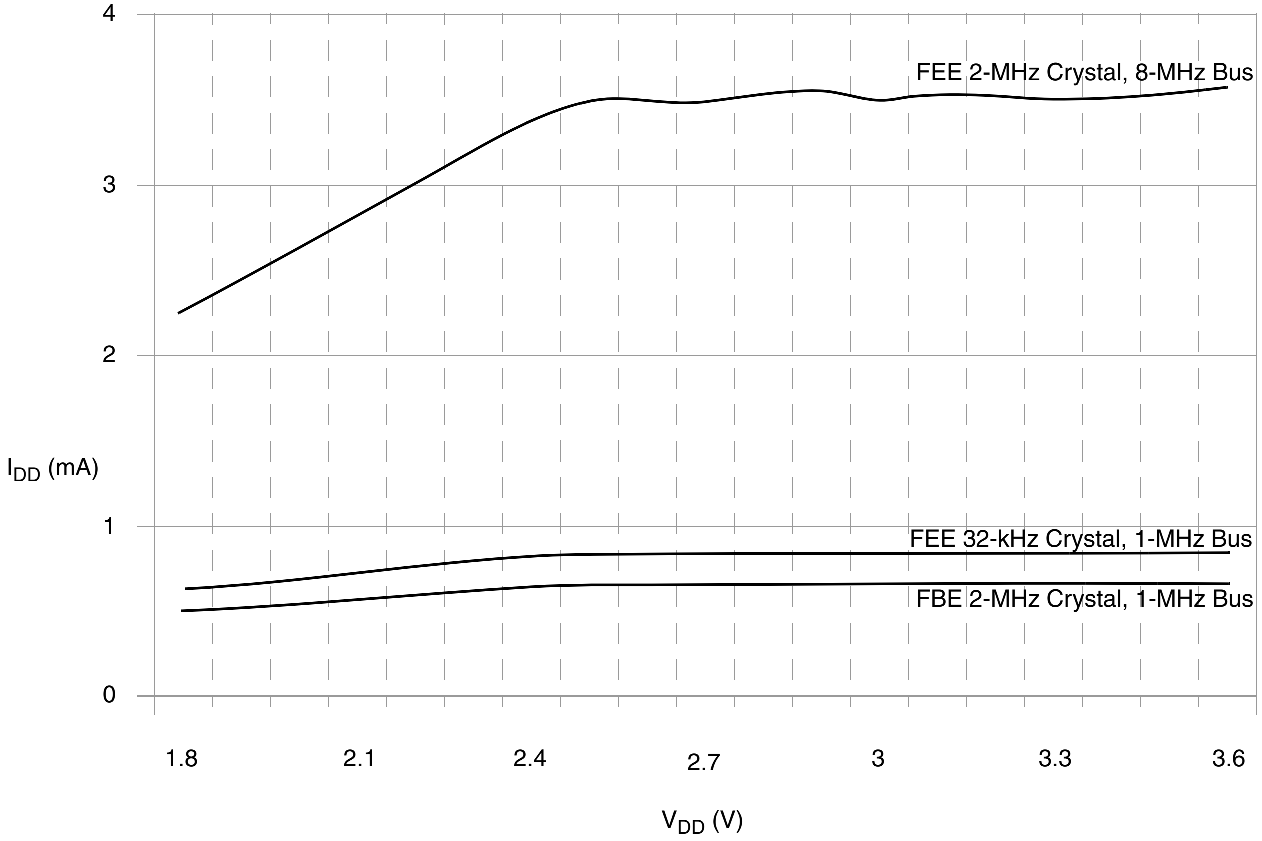


Figure A-6. Typical Run I_{DD} for FBE and FEE, I_{DD} vs. V_{DD} (ACMP and ADC off, All Other Modules Enabled)

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

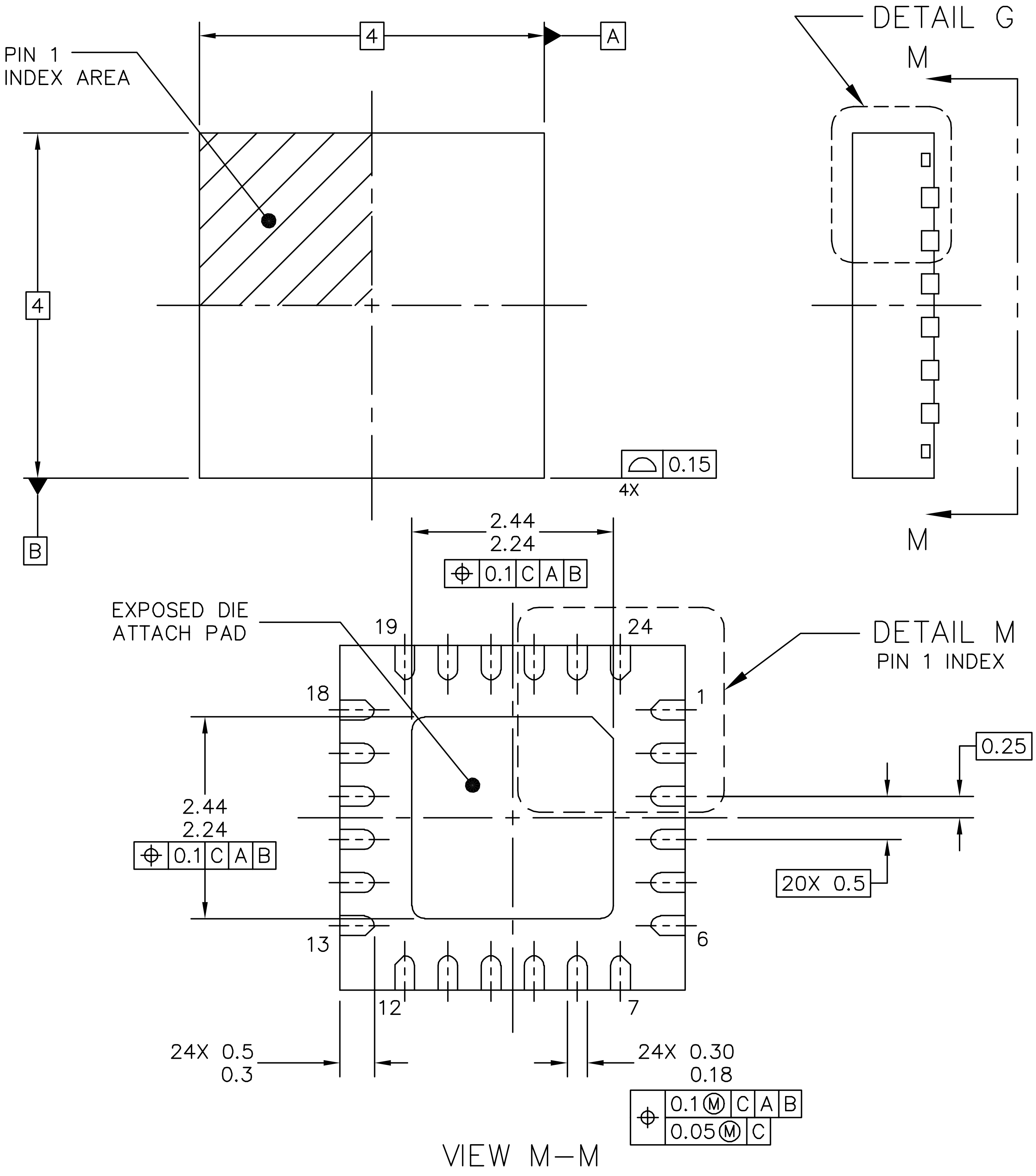
Characteristic	Symbol	Min	Typical	Max	Unit
Supply voltage for program/erase: T ≤ 85°C	$V_{prog/erase}$	1.8	—	3.6	V
T > 85 °C		2.1	—	3.6	
Supply voltage for read operation	V_{Read}	1.8	—	3.6	V
Internal FCLK frequency[1]	f_{FCLK}	150	—	200	kHz
Internal FCLK period (1/FCLK)	t_{Fcyc}	5	—	6.67	μs
Byte program 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°C to + 125°C T = 25°C		10,000	— 100,000	— —	cycles
Data retention[4]	t_{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.*



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: O

CASE NUMBER: 1897–01

08 SEP 2006

STANDARD: JEDEC MO–220 VGGD–8

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS.

2. DIMENSIONING AND TOLERANCING PER ASME Y14.5M–1994.

3. THE COMPLETE JEDEC DESIGNATOR FOR THIS PACKAGE IS: HF–PQFN.

△4. COPLANARITY APPLIES TO LEADS AND DIE ATTACH PAD.

5. MIN. METAL GAP SHOULD BE 0.2 MM.

© 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		

NOTES:

1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M – 1994.
2. ALL DIMENSIONS ARE IN INCHES.
3. 626-03 TO 626-06 OBSOLETE. NEW STANDARD 626-07.
4. DIMENSION TO CENTER OF LEAD WHEN FORMED PARALLEL.
5. PACKAGE CONTOUR OPTIONAL (ROUND OR SQUARE CONERS).

STYLE 1:

PIN			
1.	AC IN	5.	GROUND
2.	DC + IN	6.	OUTPUT
3.	DC – IN	7.	AUXILIARY
4.	AC IN	8.	VCC

MECHANICAL OUTLINE

PRINT VERSION NOT TO SCALE

TITLE:

8 LD PDIP

DOCUMENT NO: 98ASB42420B

REV: N

CASE NUMBER: 626-06

19 MAY 2005

STANDARD: NON-JEDEC