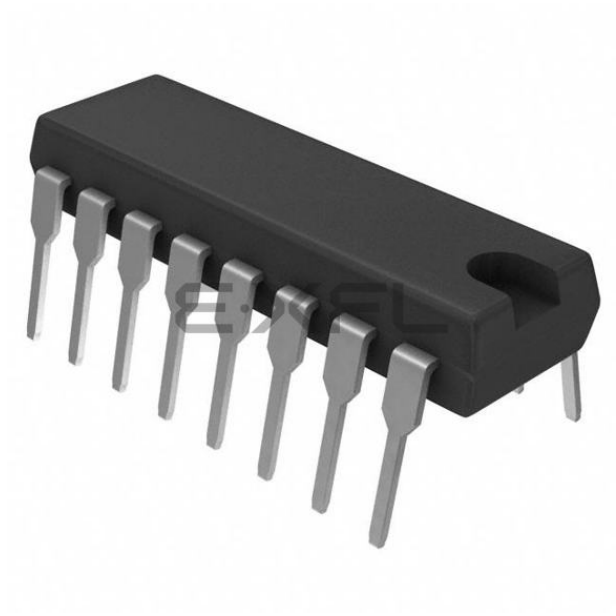


Welcome to E-XFL.COM

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	Obsolete
Core Processor	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	SCI
Peripherals	LVD, POR, PWM
Number of I/O	13
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 4x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	16-DIP (0.300", 7.62mm)
Supplier Device Package	16-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68hc908kx8cp

1.4 Pin Assignments

Figure 1-2 shows the pin assignments for MC68HC908KX8.

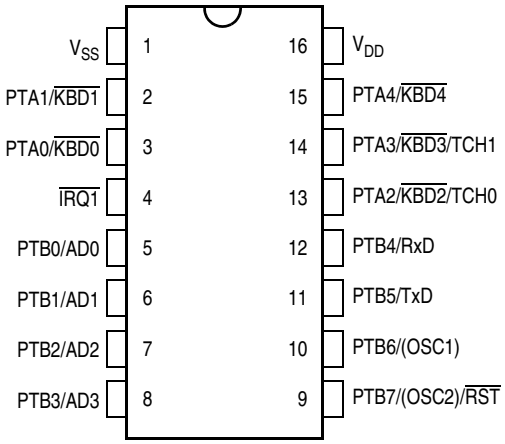
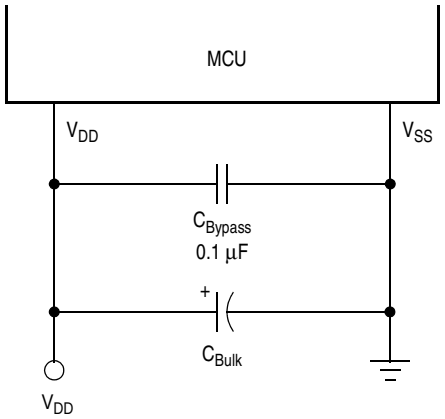


Figure 1-2. PDIP and SOIC Pin Assignments

1.4.1 Supply Pins (V_{DD} and V_{SS})

V_{DD} and V_{SS} are the power supply and ground pins. The MCU operates from a single power supply.

Fast signal transitions on MCU pins place high, short-duration current demands on the power supply. To prevent noise problems, take special care to provide power supply bypassing at the MCU as shown in Figure 1-3. Place the bypass capacitors as close to the MCU power pins as possible. Use high-frequency response ceramic capacitors for C_{Bypass}. C_{Bulk} are optional bulk current bypass capacitors for use in applications that require the port pins to source high-current levels.



Note: Component values shown represent typical applications.

Figure 1-3. Power Supply Bypassing

$0000 ↓ $003F	I/O REGISTERS (64 BYTES)
$0040 ↓ $00FF	RAM (192 BYTES)
$0100 ↓ $0FFF	UNIMPLEMENTED (3839 BYTES)
$1000 ↓ $13FF	FLASH BURN-IN ROM (1024 BYTES)
$1400 ↓ $DFFF	UNIMPLEMENTED (52,224 BYTES)
$E000 ↓ $FDFF	USER FLASH MEMORY (7680 BYTES)

$FE00	RESERVED
$FE01	SIM RESET STATUS REGISTER (SRSR)
$FE02	RESERVED
$FE03	RESERVED
$FE04	INTERRUPT STATUS REGISTER 1 (INT1)
$FE05	INTERRUPT STATUS REGISTER 2 (INT2)
$FE06	INTERRUPT STATUS REGISTER 3 (INT3)
$FE07	RESERVED
$FE08	FLASH CONTROL REGISTER (FLCR)
$FE09	BREAK ADDRESS REGISTER HIGH (BRKH)
$FE0A	BREAK ADDRESS REGISTER LOW (BRKL)
$FE0B	BREAK STATUS AND CONTROL REGISTER (BRKSCR)
$FE0C	LVI STATUS REGISTER (LVISR)
$FE0D ↓ $FE1F	UNIMPLEMENTED (18 BYTES)
$FE20 ↓ $FF46	MONITOR ROM (295 BYTES)
$FF47 ↓ $FF7D	UNIMPLEMENTED (57 BYTES)
$FF7E	FLASH BLOCK PROTECT REGISTER (FLBPR)
$FF7F ↓ $FFDB	UNIMPLEMENTED (90 BYTES)
$FFDC ↓ $FFFF	FLASH VECTORS (36 BYTES)

Figure 2-1. Memory Map

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0000	Port A Data Register (PTA) See page 106.	Read:	0	0	0	PTA4	PTA3	PTA2	PTA1	PTA0
		Write:								
		Reset:	Unaffected by reset							
$0001	Port B Data Register (PTB) See page 108.	Read:	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
		Write:								
		Reset:	Unaffected by reset							
$0002	Unimplemented									
$0003	Unimplemented									
$0004	Data Direction Register A (DDRA) See page 106.	Read:	0	0	0	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0005	Data Direction Register B (DDRB) See page 109.	Read:	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0006 ↓ $000C	Unimplemented									
$000D	Port A Input Pullup Enable Register (PTAPUE) See page 108.	Read:	0	0	0	PTAPUE4	PTAPUE3	PTAPUE2	PTAPUE1	PTAPUE0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$000E ↓ $0012	Unimplemented									
$0013	SCI Control Register 1 (SCC1) See page 125.	Read:	LOOPS	ENSCI	TXINV	M	WAKE	ILTY	PEN	PTY
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0014	SCI Control Register 2 (SCC2) See page 127.	Read:	SCTIE	TCIE	SCRIE	ILIE	TE	RE	RWU	SBK
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0015	SCI Control Register 3 (SCC3) See page 129.	Read:	R8	T8	R	R	ORIE	NEIE	FEIE	PEIE
		Write:								
		Reset:	U	U	0	0	0	0	0	0
$0016	SCI Status Register 1 (SCS1) See page 130.	Read:	SCTE	TC	SCRF	IDLE	OR	NF	FE	PE
		Write:								
		Reset:	1	1	0	0	0	0	0	0

= Unimplemented R = Reserved U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 1 of 5)

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0017	SCI Status Register 2 (SCS2) See page 132.	Read:	0	0	0	0	0	0	BKF	RPF
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$0018	SCI Data Register (SCDR) See page 133.	Read:	R7	R6	R5	R4	R3	R2	R1	R0
		Write:	T7	T6	T5	T4	T3	T2	T1	T0
		Reset:	Unaffected by reset							
$0019	SCI Baud Rate Register (SCBR) See page 133.	Read:	0	0	SCP1	SCP0	R	SCR2	SCR1	SCR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001A	Keyboard Status and Control Register (KBSCR) See page 99.	Read:	0	0	0	0	KEYF	0	IMASKK	MODEK
		Write:						ACKK		
		Reset:	0	0	0	0	0	0	0	0
$001B	Keyboard Interrupt Enable Register (KBIER) See page 100.	Read:	0	0	0	KBIE4	KBIE3	KBIE2	KBIE1	KBIE0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001C	Timebase Control Register (TBCR) See page 154.	Read:	TBIF	TBR2	TBR1	TBR0	0	TBIE	TBON	R
		Write:					TACK			
		Reset:	0	0	0	0	0	0	0	0
$001D	IRQ Status and Control Register (ISCR) See page 94.	Read:	0	0	0	0	IRQF1	0	IMASK1	MODE1
		Write:	R	R	R	R	R	ACK1		
		Reset:	0	0	0	0	0	0	0	0
$001E	Configuration Register 2[1] (CONFIG2) See page 48.	Read:	R	0	EXT-XTALEN	EXT-SLOW	EXT-CLKEN	0	OSCENIN-STOP	SCIBDSRC
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$001F	Configuration Register 1[1] (CONFIG1) See page 47.	Read:	COPRS	LVISTOP	LVIRSTD	LVIPWRD	LVI5OR3	SSREC	STOP	COPD
		Write:								
		POR Reset:	0	0	0	0	0	0	0	0
		Other Resets:	0	0	0	0	U	0	0	0

1. LVI5OR3 is only writable after a power-on reset (POR). Bit 6 of CONFIG1 is read-only and will read 0. All other bits in CONFIG1 and CONFIG2 are one-time writable after any reset.

$0020	Timer Status and Control Register (TSC) See page 163.	Read:	TOF	TOIE	TSTOP	0	0	PS2	PS1	PS0
		Write:	0			TRST				
		Reset:	0	0	1	0	0	0	0	0
$0021	Timer Counter Register High (TCNTH) See page 164.	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented R = Reserved U = Unaffected

Figure 2-2. Control, Status, and Data Registers (Sheet 2 of 5)

Table 2-1. Vector Locations

Low

Priority

High

Address	Vector
$FFDC	Timebase module vector (high)
$FFDD	Timebase module vector (low)
$FFDE	ADC conversion complete vector (high)
$FFDF	ADC conversion complete vector (low)
$FFE0	Keyboard vector (high)
$FFE1	Keyboard vector (low)
$FFE2	SCI transmit vector (high)
$FFE3	SCI transmit vector (low)
$FFE4	SCI receive vector (high)
$FFE5	SCI receive vector (low)
$FFE6	SCI receive error vector (high)
$FFE7	SCI receive error vector (low)
$FFE8	Reserved
$FFE9	Reserved
$FFEA	Reserved
$FFEB	Reserved
$FFEC	Reserved
$FFED	Reserved
$FFEE	Reserved
$FFEF	Reserved
$FFF0	Reserved
$FFF1	Reserved
$FFF2	TIM overflow vector (high)
$FFF3	TIM overflow vector (low)
$FFF4	TIM channel 1 vector (high)
$FFF5	TIM channel 1 vector (low)
$FFF6	TIM channel 0 vector (high)
$FFF7	TIM channel 0 vector (low)
$FFF8	CMIREQ vector (high)
$FFF9	CMIREQ vector (low)
$FFFA	IRQ1 vector (high)
$FFFB	IRQ1 vector (low)
$FFFC	SWI vector (high)
$FFFD	SWI vector (low)
$FFFE	Reset vector (high)
$FFFF	Reset vector (low)

7.4.4 Quantization Error in DCO Output

The digitally controlled oscillator (DCO) is comprised of three major sub-blocks:

- Binary weighted divider
- Variable-delay ring oscillator
- Ring oscillator fine-adjust circuit

Each of these blocks affects the clock period of the internal clock (ICLK). Since these blocks are controlled by the digital loop filter (DLF) outputs DDIV and DSTG, the output of the DCO can only change in quantized steps as the DLF increments or decrements its output. The following subsections describe how each block will affect the output frequency.

7.4.4.1 Digitally Controlled Oscillator

The digitally controlled oscillator (DCO) is an inaccurate oscillator which generates the internal clock (ICLK), whose clock period is dependent on the digital loop filter outputs (DSTG[7:0] and DDIV[3:0]). Because of the digital nature of the DCO, the clock period of ICLK will change in quantized steps. This will create a clock period difference, or quantization error (Q-ERR) from one cycle to the next. Over several cycles or for longer periods, this error is divided out until it reaches a minimum error of 0.202% to 0.368%. The dependence of this error on the DDIV[3:0] value and the number of cycles the error is measured over is shown in Table 7-3.

Table 7-3. Quantization Error in ICLK

DDIV[3:0]	ICLK Cycles	Bus Cycles	τ_{ICLK} Q-ERR
%0000 (min)	1	NA	6.45% – 11.8%
%0000 (min)	4	1	1.61% – 2.94%
%0000 (min)	≥ 32	≥ 8	0.202% – 0.368%
%0001	1	NA	3.23% – 5.88%
%0001	4	1	0.806% – 1.47%
%0001	≥ 16	≥ 4	0.202% – 0.368%
%0010	1	NA	1.61% – 2.94%
%0010	4	1	0.403% – 0.735%
%0010	≥ 8	≥ 2	0.202% – 0.368%
%0011	1	NA	0.806% – 1.47%
%0011	≥ 4	≥ 1	0.202% – 0.368%
%0100	1	NA	0.403% – 0.735%
%0100	≥ 2	≥ 1	0.202% – 0.368%
%0101 – %1001 (max)	≥ 1	≥ 1	0.202% – 0.368%

7.4.4.2 Binary Weighted Divider

The binary weighted divider divides the output of the ring oscillator by a power of 2, specified by the DCO divider control bits (DDIV[3:0]). DDIV maximizes at %1001 (values of %1010 through %1111 are interpreted as %1001), which corresponds to a divide by 512. When DDIV is %0000, the ring oscillator's output is divided by 1. Incrementing DDIV by one will double the period; decrementing DDIV will halve the period. The DLF cannot directly increment or decrement DDIV; DDIV is only incremented or decremented when an addition or subtraction to DSTG carries or borrows.

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0039	ICG DCO Divider Control Register (ICGDVR) See page 89.	Read:					DDIV3	DDIV2	DDIV1	DDIV0
		Write:								
		Reset:	0	0	0	0	U	U	U	U
$003A	ICG DCO Stage Control Register (ICGDSR) See page 89.	Read:	DSTG7	DSTG6	DSTG5	DSTG4	DSTG3	DSTG2	DSTG1	DSTG0
		Write:	R	R	R	R	R	R	R	R
		Reset:	U	U	U	U	U	U	U	U

= Unimplemented R = Reserved U = Unaffected

Figure 7-10. ICG Module I/O Register Summary (Continued)

Table 7-5. ICG Module Register Bit Interaction Summary

Condition	Register Bit Results for Given Condition											
	CMIE	CMF	CMON	CS	ICGON	ICGS	ECGON	ECGS	N[6:0]	TRIM[7:0]	DDIV[3:0]	DSTB[7:0]
Reset	0	0	0	0	1	0	0	0	$15	$80	—	—
OSCENINSTOP = 0, STOP = 1	0	0	0	—	—	0	—	0	—	—	—	—
EXTCLKEN = 0	0	0	0	0	1	—	0	0	—	—	uw	uw
CMF = 1	—	(1)	1	—	1	—	1	—	uw	uw	uw	uw
CMON = 0	0	0	(0)	—	—	—	—	—	—	—	—	—
CMON = 1	—	—	(1)	—	1	—	1	—	uw	uw	uw	uw
CS = 0	—	—	—	(0)	1	—	—	—	—	—	uw	uw
CS = 1	—	—	—	(1)	—	—	1	—	—	—	—	—
ICGON = 0	0	0	0	1	(0)	0	1	—	—	—	—	—
ICGON = 1	—	—	—	—	(1)	—	—	—	—	—	uw	uw
ICGS = 0	us	—	us	uc	—	(0)	—	—	—	—	—	—
ECGON = 0	0	0	0	0	1	—	(0)	0	—	—	uw	uw
ECGS = 0	us	—	us	us	—	—	—	(0)	—	—	—	—
IOFF = 1	—	1*	(1)	1	(1)	0	(1)	—	uw	uw	uw	uw
EOFF = 1	—	1*	(1)	0	(1)	—	(1)	0	uw	uw	uw	uw
N = written	(0)	(0)	(0)	—	—	0*	—	—	—	—	—	—
TRIM = written	(0)	(0)	(0)	—	—	0*	—	—	—	—	—	—

—Register bit is unaffected by the given condition.
0, 1Register bit is forced clear or set (respectively) in the given condition.
0*, 1*Register bit is temporarily forced clear or set (respectively) in the given condition.
(0), (1)Register bit must be clear or set (respectively) for the given condition to occur.
us, uc, uwRegister bit cannot be set, cleared, or written (respectively) in the given condition.

DDRA4–DDRA0 — Data Direction Register A Bits

These read/write bits control port A data direction. Reset clears DDRA4–DDRA0, configuring all port A pins as inputs.

1 = Corresponding port A pin configured as output
0 = Corresponding port A pin configured as input

NOTE

Avoid glitches on port A pins by writing to the port A data register before changing data direction register A bits from 0 to 1.

Figure 11-4 shows the port A I/O logic.

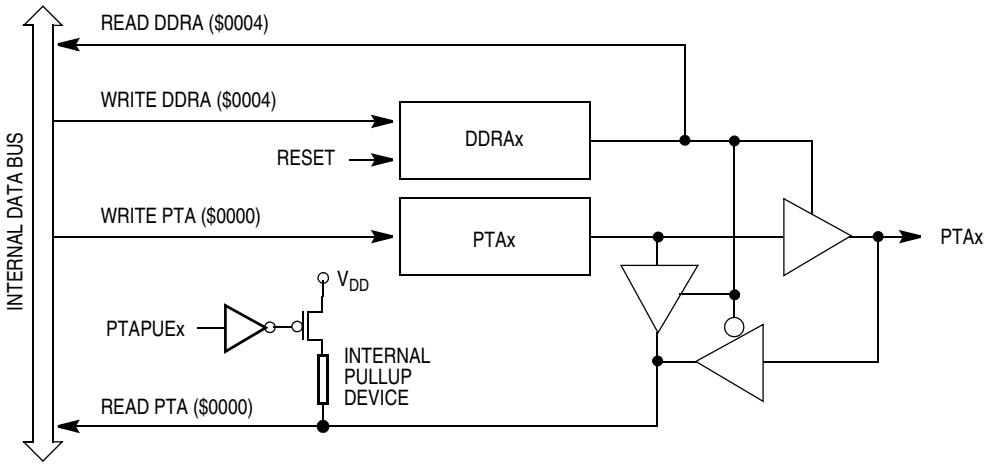


Figure 11-4. Port A I/O Circuit

When bit DDRAx is a 1, reading address $0000 reads the PTAx data latch. When bit DDRAx is a 0, reading address $0000 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. Table 11-1 summarizes the operation of the port A pins.

Table 11-1. Port A Pin Functions

PTAPUE Bit	DDRA Bit	PTA Bit	I/O Pin Mode	Accesses to DDRA	Accesses to PTA	
				Read/Write	Read	Write
1	0	X	Input, V_{DD}[1]	DDRA4–DDRA0	Pin	PTA4–PTA0[2]
0	0	X	Input, Hi-Z	DDRA4–DDRA0	Pin	PTA4–PTA0[3]
X	1	X	Output	DDRA4–DDRA0	PTA4–PTA0	PTA4–PTA0

X = Don't care
Hi-Z = High impedance

1. I/O pin pulled up to V_{DD} by internal pulllup device
2. Writing affects data register, but does not affect input.

11.2.3 Port A Input Pullup Enable Register

The port A input pullup enable register (PTAPUE) contains a software configurable pullup device for each of the five port A pins. Each bit is individually configurable and requires that the data direction register, DDRA, bit be configured as an input. Each pullup is automatically disabled when a port bit's DDRA is configured for output mode.

12.4.3.1 Character Length

The receiver can accommodate either 8-bit or 9-bit data. The state of the M bit in SCI control register 1 (SCC1) determines character length. When receiving 9-bit data, bit R8 in SCI control register 2 (SCC2) is the ninth bit (bit 8). When receiving 8-bit data, bit R8 is a copy of the eighth bit (bit 7).

12.4.3.2 Character Reception

During an SCI reception, the receive shift register shifts characters in from the RxD pin. The SCI data register (SCDR) is the read-only buffer between the internal data bus and the receive shift register.

After a complete character shifts into the receive shift register, the data portion of the character transfers to the SCDR. The SCI receiver full bit, SCRF, in SCI status register 1 (SCS1) becomes set, indicating that the received byte can be read. If the SCI receive interrupt enable bit, SCRIE, in SCC2 is also set, the SCRF bit generates a receiver CPU interrupt request.

12.4.3.3 Data Sampling

The receiver samples the RxD pin at the RT clock rate. The RT clock is an internal signal with a frequency 16 times the baud rate. To adjust for baud rate mismatch, the RT clock is resynchronized at these times (see Figure 12-7):

- After every start bit
- After the receiver detects a data bit change from 1 to 0 (after the majority of data bit samples at RT8, RT9, and RT10 returns a valid 1 and the majority of the next RT8, RT9, and RT10 samples returns a valid 0)

To locate the start bit, data recovery logic does an asynchronous search for a 0 preceded by three 1s. When the falling edge of a possible start bit occurs, the RT clock begins to count to 16.

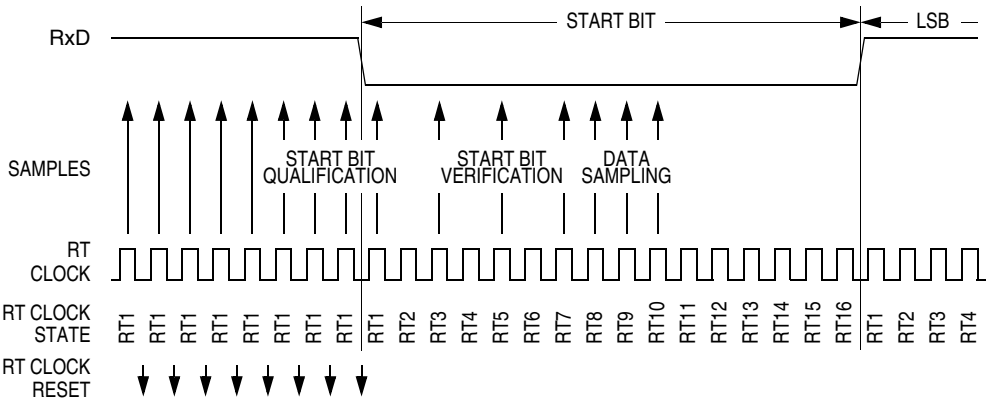


Figure 12-7. Receiver Data Sampling

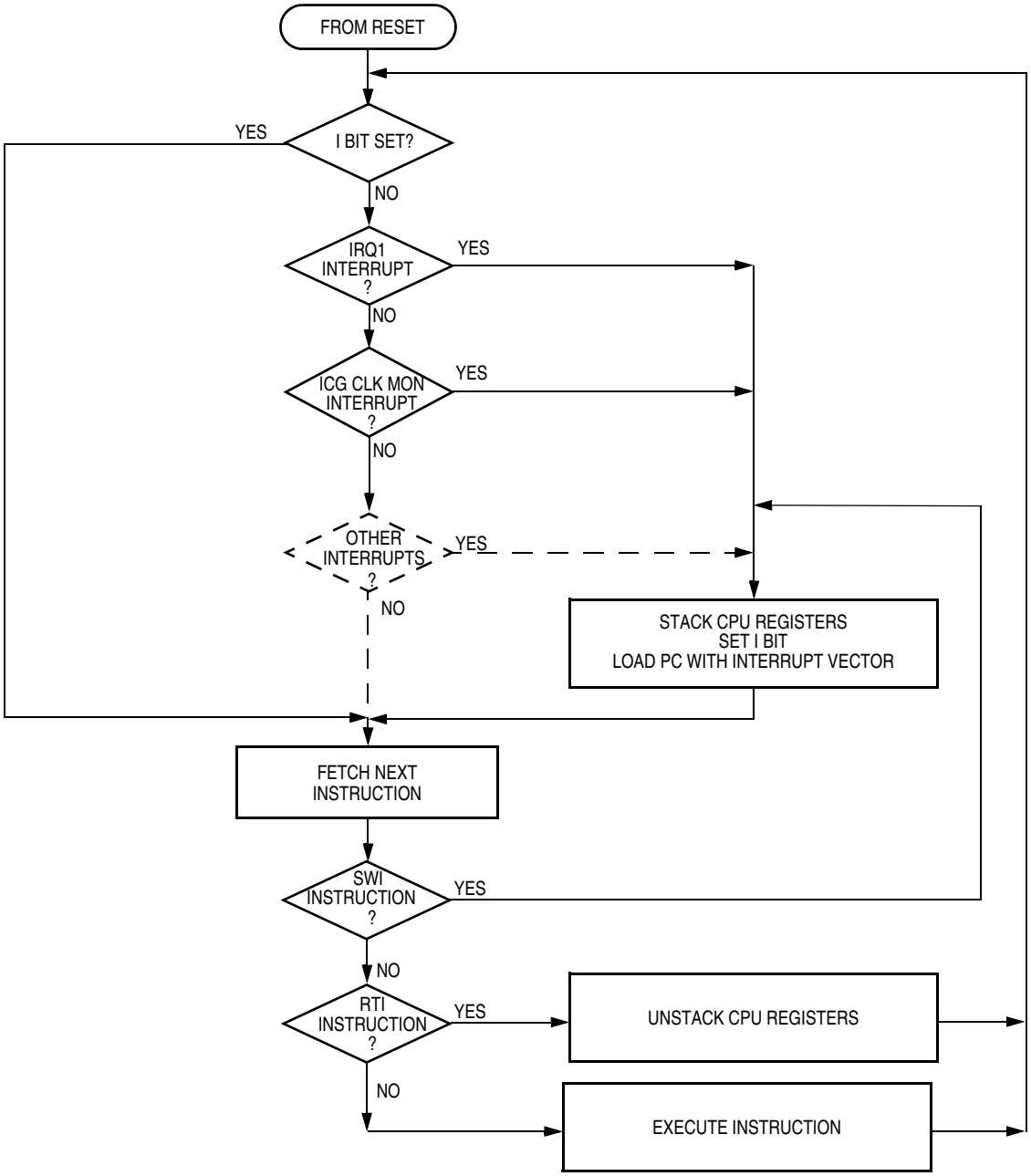


Figure 13-9. Interrupt Processing

13.5.1.1 Hardware Interrupts

A hardware interrupt does not stop the current instruction. Processing of a hardware interrupt begins after completion of the current instruction. When the current instruction is complete, the SIM checks all pending hardware interrupts. If interrupts are not masked (I bit clear in the condition code register), and if the corresponding interrupt enable bit is set, the SIM proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If more than one interrupt is pending at the end of an instruction execution, the highest priority interrupt is serviced first. Figure 13-10 demonstrates what happens when two interrupts are pending. If an interrupt

IF6 — Interrupt Flag 6

Since the MC68HC908KX8 parts do not use this interrupt flag, this bit will always read 0.

Bit 0 and Bit 1 — Always read 0

13.7.2.2 Interrupt Status Register 2

Address: $FE05

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IF14	IF13	IF12	IF11	IF10	IF9	IF8	IF7
Write:	R	R	R	R	R	R	R	R
Reset:	0	0	0	0	0	0	0	0

R = Reserved

Figure 13-18. Interrupt Status Register 2 (INT2)

IF14–IF11 — Interrupt Flags 14–11

These flags indicate the presence of interrupt requests from the sources shown in Table 13-2.

1 = Interrupt request present
0 = No interrupt request present

IF10–IF7 — Interrupt Flags 10–7

Since the MC68HC908KX8 parts do not use these interrupt flags, these bits will always read 0.

13.7.2.3 Interrupt Status Register 3

Address: $FE06

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IF22	IF21	IF20	IF19	IF18	IF17	IF16	IF15
Write:	R	R	R	R	R	R	R	R
Reset:	0	0	0	0	0	0	0	0

R = Reserved

Figure 13-19. Interrupt Status Register 3 (INT3)

IF22–IF17 — Interrupt Flags 22–17

Since the MC68HC908KX8 parts do not use these interrupt flags, these bits will always read 0.

IF16–IF15 — Interrupt Flags 16–15

These flags indicate the presence of interrupt requests from the sources shown in Table 13-2.

1 = Interrupt request present
0 = No interrupt request present

14.5 TBM Interrupt Rate

The interrupt rate is determined by the equation:

$$t_{TBMRATE} = \frac{1}{f_{TBMRATE}} = \frac{Divider}{f_{TBMCLK}}$$

where:

f_{TBMCLK} = Frequency supplied from the internal clock generator (ICG) module
Divider = Divider value as determined by TBR2–TBR0 settings. See Table 14-1.

As an example, a clock source of 4.9152 MHz and the TBR2–TBR0 set to {011}, the divider tap is 128 and the interrupt rate calculates to $128/4.9152 \times 10^6 = 26$ µs.

Table 14-1. Timebase Divider Selection

TBR2	TBR1	TBR0	Divider Tap
0	0	0	32768
0	0	1	8192
0	1	0	2048
0	1	1	128
1	0	0	64
1	0	1	32
1	1	0	16
1	1	1	8

NOTE

Do not change TBR2–TBR0 bits while the timebase is enabled (TBON = 1).

14.6 Low-Power Modes

The WAIT and STOP instructions put the MCU in low power-consumption standby modes.

14.6.1 Wait Mode

The timebase module remains active after execution of the WAIT instruction. In wait mode the timebase register is not accessible by the CPU.

If the timebase functions are not required during wait mode, reduce the power consumption by stopping the timebase before executing the WAIT instruction.

14.6.2 Stop Mode

The timebase module may remain active after execution of the STOP instruction if the internal clock generator has been enabled to operate during stop mode through the OSCENINSTOP bit in the configuration register. The timebase module can be used in this mode to generate a periodic wake up from stop mode.

If the internal clock generator has not been enabled to operate in stop mode, the timebase module will not be active during stop mode. In stop mode, the timebase register is not accessible by the CPU.

If the timebase functions are not required during stop mode, reduce power consumption by disabling the timebase module before executing the STOP instruction.

15.4.4 Unbuffered Output Compare

Any output compare channel can generate unbuffered output compare pulses as described in 15.4.3 Output Compare. The pulses are unbuffered because changing the output compare value requires writing the new value over the old value currently in the TIM channel registers.

An unsynchronized write to the TIM channel registers to change an output compare value could cause incorrect operation for up to two counter overflow periods. For example, writing a new value before the counter reaches the old value but after the counter reaches the new value prevents any compare during that counter overflow period. Also, using a TIM overflow interrupt routine to write a new, smaller output compare value may cause the compare to be missed. The TIM may pass the new value before it is written.

Use these methods to synchronize unbuffered changes in the output compare value on channel x:

- When changing to a smaller value, enable channel x output compare interrupts and write the new value in the output compare interrupt routine. The output compare interrupt occurs at the end of the current output compare pulse. The interrupt routine has until the end of the counter overflow period to write the new value.
- When changing to a larger output compare value, enable TIM overflow interrupts and write the new value in the TIM overflow interrupt routine. The TIM overflow interrupt occurs at the end of the current counter overflow period. Writing a larger value in an output compare interrupt routine (at the end of the current pulse) could cause two output compares to occur in the same counter overflow period.

15.4.5 Buffered Output Compare

Channels 0 and 1 can be linked to form a buffered output compare channel whose output appears on the TCH0 pin. The TIM channel registers of the linked pair alternately control the output.

Setting the MS0B bit in TIM channel 0 status and control register (TSC0) links channel 0 and channel 1. The output compare value in the TIM channel 0 registers initially controls the output on the TCH0 pin. Writing to the TIM channel 1 registers enables the TIM channel 1 registers to synchronously control the output after the TIM overflows. At each subsequent overflow, the TIM channel registers (0 or 1) that control the output are the 1s written to last. TSC0 controls and monitors the buffered output compare function, and TIM channel 1 status and control register (TSC1) is unused. While the MS0B bit is set, the channel 1 pin, TCH1, is available as a general-purpose I/O pin.

NOTE

In buffered output compare operation, do not write new output compare values to the currently active channel registers. User software should track the currently active channel to prevent writing a new value to the active channel. Writing to the active channel registers is the same as generating unbuffered output compares.

15.4.6 Pulse-Width Modulation (PWM)

By using the toggle-on-overflow feature with an output compare channel, the TIM can generate a PWM signal. The value in the TIM counter modulo registers determines the period of the PWM signal. The channel pin toggles when the counter reaches the value in the TIM counter modulo registers. The time between overflows is the period of the PWM signal.

As Figure 15-4 shows, the output compare value in the TIM channel registers determines the pulse width of the PWM signal. The time between overflow and output compare is the pulse width. Program the TIM

5. In the TIM status control register (TSC), clear the TIM stop bit, TSTOP.

Setting MS0B links channels 0 and 1 and configures them for buffered PWM operation. The TIM channel 0 registers (TCH0H and TCH0L) initially control the buffered PWM output. TIM status control register 0 (TSCR0) controls and monitors the PWM signal from the linked channels.

Clearing the toggle-on-overflow bit, TOVx, inhibits output toggles on TIM overflows. Subsequent output compares try to force the output to a state it is already in and have no effect. The result is a 0 percent duty cycle output.

Setting the channel x maximum duty cycle bit (CHxMAX) and setting the TOVx bit generates a 100 percent duty cycle output. See 15.8.4 TIM Channel Status and Control Registers.

15.5 Interrupts

These TIM sources can generate interrupt requests:

- TIM overflow flag (TOF) — The timer overflow flag (TOF) bit is set when the TIM counter reaches the modulo value programmed in the TIM counter modulo registers. The TIM overflow interrupt enable bit, TOIE, enables TIM overflow interrupt requests. TOF and TOIE are in the TIM status and control registers.
- TIM channel flags (CH1F and CH0F) — The CHxF bit is set when an input capture or output compare occurs on channel x. Channel x TIM CPU interrupt requests are controlled by the channel x interrupt enable bit, CHxIE. Channel x TIM CPU interrupt requests are enabled when CHxIE = 1. CHxF and CHxIE are in the TIM channel x status and control register.

15.6 Low-Power Modes

The WAIT and STOP instructions put the MCU in low power-consumption standby modes.

15.6.1 Wait Mode

The TIM remains active after the execution of a WAIT instruction. In wait mode the TIM registers are not accessible by the CPU. Any enabled CPU interrupt request from the TIM can bring the MCU out of wait mode.

If TIM functions are not required during wait mode, reduce power consumption by stopping the TIM before executing the WAIT instruction.

15.6.2 Stop Mode

The TIM is inactive after the execution of a STOP instruction. The STOP instruction does not affect register conditions or the state of the TIM counter. TIM operation resumes when the MCU exits stop mode after an external interrupt.

15.7 I/O Signals

Port A shares two of its pins with the TIM, PTA3/$\overline{\text{KBD3}}$/TCH1 and PTA2/$\overline{\text{KBD2}}$/TCH0. Each channel input/output (I/O) pin is programmable independently as an input capture pin or an output compare pin. TCH0 can be configured as buffered output compare or buffered PWM pins.

16.3.1.5 Data Format

The MCU waits for the host to send eight security bytes (see 16.3.2 Security). After the security bytes, the MCU sends a break signal (10 consecutive 0s) to the host computer, indicating that it is ready to receive a command.

Communication with the monitor ROM is in standard non-return-to-zero (NRZ) mark/space data format. Transmit and receive baud rates must be identical.

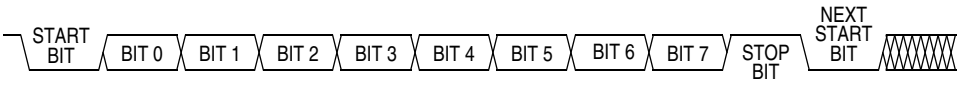


Figure 16-10. Monitor Data Format

16.3.1.6 Break Signal

A start bit (0) followed by nine 0 bits is a break signal. When the monitor receives a break signal, it drives the PTA0 pin high for the duration of two bits and then echoes back the break signal.

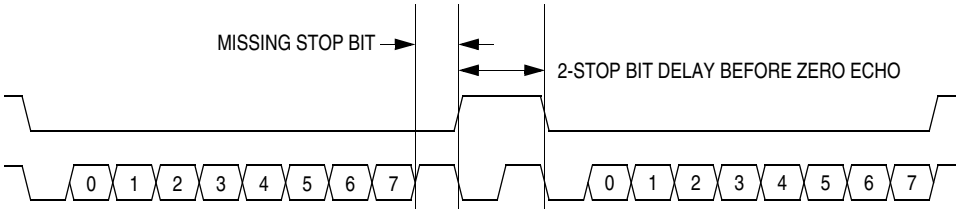


Figure 16-11. Break Transaction

16.3.1.7 Baud Rate

The communication baud rate is controlled by the CGMXCLK frequency output of the internal clock generator module.

16.3.1.8 Force Monitor Mode

In forced monitor mode, the baud rate is fixed at CGMXCLK/1024. A CMGXCLK frequency of 4.9152 MHz results in a 4800 baud rate. A 9.8304-MHz frequency produces a 9600 baud rate.

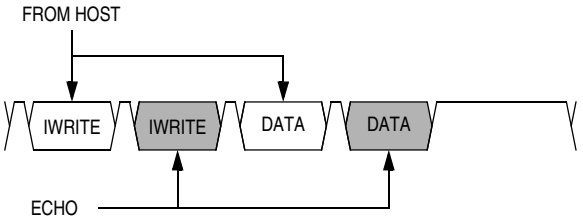
16.3.1.9 Normal Monitor Mode

In normal monitor mode, the communication baud rate is controlled by the CGMXCLK frequency output of the internal clock generator module. Table 16-3 lists CGMXCLK frequencies required to achieve standard baud rates. Other standard baud rates can be accomplished using other clock frequencies. The internal clock can be used as the clock source by programming the internal clock generator registers however, monitor mode will always be entered using the external clock as the clock source.

Table 16-3. Normal Monitor Mode Baud Rate Selection

CGMXCLK Frequency (MHz)	Baud Rate
9.8304	9600

Table 16-7. IWRITE (Indexed Write) Command

Description	Write to last address accessed + 1
Operand	Single data byte
Data Returned	None
Opcode	$19
Command Sequence	

A sequence of IREAD or IWRITE commands can access a block of memory sequentially over the full 64-Kbyte memory map.

Table 16-8. READSP (Read Stack Pointer) Command

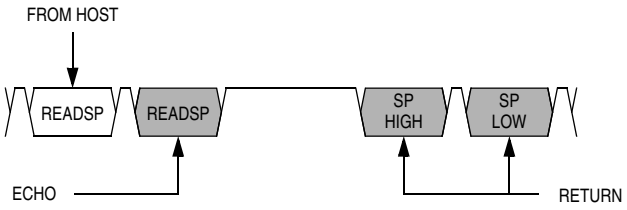
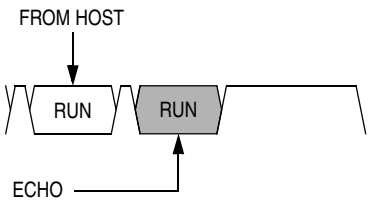
Description	Reads stack pointer
Operand	None
Data Returned	Returns incremented stack pointer value (SP + 1) in high-byte:low-byte order
Opcode	$0C
Command Sequence	

Table 16-9. RUN (Run User Program) Command

Description	Executes PULH and RTI instructions
Operand	None
Data Returned	None
Opcode	$28
Command Sequence	

Appendix A
MC68HC908KX2

A.1 Introduction

This appendix describes the differences between the MC68HC908KX8 and the MC68HC908KX2.

A.2 Functional Description

The MC68HC908KX2 FLASH memory is an array of 2,048 bytes with an additional 36 bytes of user vectors and one byte used for block protection. See Figure A-1.

NOTE

An erased bit reads as a 1 and a programmed bit reads as a 0.

The program and erase operations are facilitated through control bits in the FLASH control register (FLCR). See 2.6 FLASH Control Register.

The FLASH is organized internally as an 8-word by 8-bit complementary metal-oxide semiconductor (CMOS) page erase, byte (8-bit) program embedded FLASH memory. Each page consists of 64 bytes. The page erase operation erases all words within a page. A page is composed of two adjacent rows.

A security feature prevents viewing of the FLASH contents.[1]

See 2.6 FLASH Control Register for a complete description of FLASH operation.

1. No security feature is absolutely secure. However, Freescale's strategy is to make reading or copying the FLASH difficult for unauthorized users.

B.2.2 Partial Use of FLASH-Related Module

16.3 Monitor ROM (MON) was written having FLASH as user memory and user vector space.

MON functions are maintained for the ROM version. MON will allow execution of code in random-access memory (RAM) or ROM and provide ROM memory security[1]. The memory programming interface, though, will have no effect in ROM version.

An assumption that must be made for the ROM version is that the reset vector will always have a value different from $0000, corresponding to the user code start address. For this reason, force entry into monitor mode, described in 16.3.1.1 Monitor Mode Entry and in 16.3.1.3 Forced Monitor Mode, is not applicable to the ROM version. The MENRST module has been eliminated from the ROM version.

The security function described in 16.3.2 Security also applies to the user ROM memory for the ROM version.

B.3 Configuration Register Programming

Functionally, the terms MOR (mask option register) and CONFIG (configuration register) can be used interchangeably. MOR and CONFIG are equivalent since both define the same module functionality options through the registers bits. As a naming convention, though, configuration registers are named MOR for a ROM version and CONFIG for a FLASH version.

Some modules affected by the configuration register bits make reference to default values of these bits and have recommendation notes on programming them.

For specific information see:

- FLASH — 2.5 FLASH Memory (FLASH)
- ICG — 7.6 CONFIG (or MOR) Register Options
- LVI — 10.3 Functional Description
- PORT — 11.3.1 Port B Data Register
- COP — 5.4.7 COPD (COP Disable) and 5.4.8 COPRS (COP Rate Select)
- CONFIG — 4.2 Functional Description

NOTE

The user must keep in mind that these notes are not entirely applicable to the MOR found in the ROM version. The MOR bits can neither assume the described CONFIG default values after reset nor can they be modified later under user code control. While the MOR is mask defined, and consequently unwritable, CONFIG can be written once after each reset.

1. No security feature is absolutely secure. However, Freescale's strategy is to make reading or copying the ROM difficult for unauthorized users.

Address: $001E

	Bit 7	6	5	4	3	2	1	Bit 0
Read: / Write:	R	LVI2	EXTXTALEN	EXTSLOW	EXTCLKEN	0	OSCEINSTOP	SCIBDSRC
Reset:	Unaffected by reset							

R = Reserved

Figure B-2. Mask Option Register 2 (MOR2)

Address: $001F

	Bit 7	6	5	4	3	2	1	Bit 0
Read: / Write:	COPRS	LVISTOP	LVIRSTD	LVIPWRD	LVI5OR3	SSREC	STOP	COPD
Reset:	Unaffected by reset							

Figure B-3. Mask Option Register 1 (MOR1)

NOTE

With the FLASH charge pump eliminated, MOR2 bit 2 (originally PMPREGD in CONFIG) has no effect. Reading this bit will return 0. For a complete description of other configuration bits, refer to 4.2 Functional Description.

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

MC68HC908KX8
Rev. 2.1, 07/2005

freescale™
semiconductor